

**PREDIKSI HARGA SAHAM HARIAN DENGAN METODE  
*GENETIC POLYNOMIAL NEURAL NETWORK (GPNN)*  
(Studi Kasus PT Telekomunikasi Indonesia, Tbk)**

**SKRIPSI**

Oleh:  
**RENKY ANGGRAINI MARTHA LINDA**  
0710940042-94



**PROGRAM STUDI MATEMATIKA  
JURUSAN MATEMATIKA  
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS BRAWIJAYA  
MALANG  
2011**

**PREDIKSI HARGA SAHAM HARIAN DENGAN METODE  
*GENETIC POLYNOMIAL NEURAL NETWORK* (GPNN)  
(Studi Kasus PT Telekomunikasi Indonesia, Tbk)**

**SKRIPSI**

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

**Oleh:**

**RENKY ANGGRAINI MARTHA LINDA**

**0710940042-94**



**PROGRAM STUDI MATEMATIKA  
JURUSAN MATEMATIKA  
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS BRAWIJAYA  
MALANG  
2011**

**LEMBAR PENGESAHAN SKRIPSI**

**PREDIKSI HARGA SAHAM HARIAN DENGAN METODE  
*GENETIC POLYNOMIAL NEURAL NETWORK (GPNN)*  
(Studi Kasus PT Telekomunikasi Indonesia, Tbk)**

Oleh:  
**RENKY ANGGRAINI MARTHA LINDA**  
**0710940042-94**

Setelah dipertahankan di depan Majelis Penguji  
pada tanggal 11 Agustus 2011  
dan dinyatakan memenuhi syarat untuk memperoleh gelar  
Sarjana Sains dalam bidang Matematika

**Pembimbing I**

**Pembimbing II**

**Prof. Dr. Marjono, M.Phil**  
**NIP. 196211161988031004**

**Syaiful Anam, S.Si, M.T.**  
**NIP. 197801152002121003**

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

**Dr. Abdul Rouf Alghofari, M.Sc**  
**NIP. 196709071992031001**

## LEMBAR PERNYATAAN

Saya yang bertanda tangan di bawah ini :

Nama : Renky Anggraini Martha Linda  
NIM : 0710940042-94  
Jurusan : Matematika  
Penulis skripsi berjudul : Prediksi Harga Saham Harian  
dengan Metode *Genetic Polynomial  
Neural Network* (GPNN)  
(Studi Kasus PT Telekomunikasi  
Indonesia, Tbk)

Dengan ini menyatakan bahwa:

1. skripsi ini adalah benar-benar karya saya sendiri, dan bukan hasil plagiat dari karya orang lain. Karya-karya yang tercantum dalam Daftar Pustaka skripsi ini, semata-mata digunakan sebagai acuan/referensi,
  2. apabila kemudian hari diketahui bahwa isi skripsi saya merupakan hasil plagiat, maka saya bersedia menanggung akibat hukum dari keadaan tersebut.
- Demikian pernyataan ini dibuat dengan segala kesadaran.

Malang, 11 Agustus 2011  
Yang menyatakan,

(Renky Anggraini Martha Linda)  
NIM. 0710940042-94

**PREDIKSI HARGA SAHAM HARIAN DENGAN METODE  
GENETIC POLYNOMIAL NEURAL NETWORK (GPNN)  
(Studi Kasus PT Telekomunikasi Indonesia, Tbk)**

**ABSTRAK**

*Polynomial Neural Network* (PNN) adalah Jaringan Saraf Tiruan (JST) yang setiap neuronnya merupakan suatu fungsi polinomial. Metode yang sering digunakan dalam proses pelatihan PNN adalah *backpropagation*. *Backpropagation* merupakan salah satu metode optimasi berdasarkan *gradient*, sehingga ketepatan *backpropagation* dalam meminimumkan galat dipengaruhi oleh pengambilan titik awal. Pengambilan titik awal yang kurang tepat menyebabkan galat tidak minimum secara global. Oleh karena itu, diperlukan metode yang dapat meminimumkan galat secara global, salah satunya yaitu Algoritma Genetika (AG).

Dalam skripsi ini diperkenalkan *Genetic Polynomial Neural Network* (GPNN) yaitu PNN yang menggunakan AG dalam meminimumkan galat. Arsitektur hasil kombinasi yang menghasilkan nilai galat minimum merupakan arsitektur terbaik yang akan digunakan untuk memprediksi harga penutupan saham harian PT Telekomunikasi Indonesia, Tbk. Setelah dilakukan beberapa kombinasi, diperoleh sebuah kombinasi terbaik yang menghasilkan galat minimum sebesar 0,01.

**Kata kunci** : Algoritma Genetika, JST, GPNN, galat, harga penutupan saham harian.

# **PREDICTION OF DAILY STOCK PRICE USING GENETIC POLYNOMIAL NEURAL NETWORK (GPNN) METHOD (Case Studies of PT. Telekomunikasi Indonesia,Tbk)**

## **ABSTRACT**

Polynomial Neural Network (PNN) is an Artificial Neural Network (ANN) which its neurons is a polynomial function. Method that normally used in PNN training process is backpropagation. Backpropagation is one of the optimization method based on gradient, so the accuration of backpropagation in minimizing error is depends on the initial point. The inappropriate initial point may not yield global minimum error. Therefore, it needs a method that can minimize the error globally. One of available methods is Genetic Algorithm (GA).

In this final project, it is introduced Genetic Polynomial Neural Network (GPNN) namely a PNN which uses GA in minimizing error. The combination architecture that produces the minimum error is the best architecture that can be used to predict the daily closing stock price of PT Telekomunikasi Indonesia, Tbk. Performing some combinations gives the best combination that gains minimum error of 0,01.

**Keywords** : Genetic Algorithm, ANN, GPNN, error, daily stock price.



## KATA PENGANTAR

Segala puji dan syukur penulis panjatkan ke hadirat Allah SWT atas segala rahmat serta hidayahNya, sehingga penulis dapat menyelesaikan skripsi ini dengan baik. Shalawat dan salam selalu tercurahkan kepada Nabi Muhammad SAW sebagai suri tauladan bagi penulis.

Banyak pihak yang telah memberikan dukungan baik moral maupun spiritual secara langsung maupun tidak langsung dalam penyelesaian skripsi ini. Penulis mengucapkan terima kasih kepada :

1. Prof. Dr. Marjono, M.Phil selaku pembimbing I dan Syaiful Anam, S.Si, M.T. selaku pembimbing II atas segala bimbingan dan motivasi yang telah diberikan selama penulisan skripsi ini.
2. Dr. Wuryansari Muharini K., M.Si., Drs. Imam Nurhadi Purwanto, M.T., dan Dra. Endang Wahyu Handamari, M.Si. selaku dosen penguji atas koreksi serta motivasinya sehingga skripsi ini dapat terselesaikan.
3. Drs. Hery Subagio, M.Kes selaku dosen pembimbing akademik atas segala bimbingan dan motivasi yang telah diberikan.
4. Seluruh dosen FMIPA khususnya Jurusan Matematika Program Studi Matematika atas segala bimbingan dan pengetahuan yang telah diberikan, dan segenap staf serta karyawan TU Jurusan Matematika atas segala bantuannya
5. Kedua orang tuaku tercinta, saudara-saudaraku tersayang, atas doa, kasih sayang dan dukungan yang telah diberikan.
6. Teman-teman seperjuangan mahasiswa Matematika 2007 atas kebersamaannya selama ini.
7. Sahabat-sahabatku Fadly, Laila, Hana, Silpi, Nina, Selly, Dwi, Candra, dan Hendra atas masukan dan kritik secara langsung.
8. Serta semua pihak yang tidak bisa disebutkan satu persatu.

Penulis menyadari bahwa masih banyak kekurangan dalam penulisan skripsi ini. Oleh karena itu, penulis dengan senang hati menerima masukan, kritik dan saran yang membangun. Akhir kata, semoga skripsi ini dapat memberi sumbangan bagi dunia sains Indonesia, khususnya di bidang Matematika Terapan.

Malang, 11 Agustus 2011

Penulis

## DAFTAR ISI

	Halaman
<b>HALAMAN JUDUL</b> .....	i
<b>LEMBAR PENGESAHAN</b> .....	iii
<b>LEMBAR PERNYATAAN</b> .....	v
<b>ABSTRAK</b> .....	vii
<b>ABSTRACT</b> .....	ix
<b>KATA PENGANTAR</b> .....	xi
<b>DAFTAR ISI</b> .....	xiii
<b>DAFTAR GAMBAR</b> .....	xv
<b>DAFTAR TABEL</b> .....	xvii
<b>DAFTAR LAMPIRAN</b> .....	xix
<b>BAB I PENDAHULUAN</b> .....	1
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	2
1.3 Batasan Masalah .....	2
1.4 Tujuan .....	2
1.5 Manfaat .....	2
<b>BAB II TINJAUAN PUSTAKA</b> .....	3
2.1 Jaringan Saraf Tiruan (JST) .....	3
2.1.1 Jaringan Lapis Tunggal .....	3
2.1.2 Jaringan Lapis Jamak .....	4
2.1.3 Jaringan Berulang .....	5
2.1.4 Jaringan Berlarik .....	5
2.2 Pelatihan JST .....	6
2.3 Fungsi Aktivasi .....	7
2.4 Ukuran Kesalahan Peramalan .....	10
2.5 <i>Polynomial Neural Network</i> (PNN) .....	12
2.6 Algoritma Genetika (AG) .....	13
2.6.1 Skema Pengkodean .....	14
2.6.2 Nilai <i>Fitness</i> .....	15
2.6.3 Seleksi Orang Tua .....	16
2.6.4 <i>Crossover</i> .....	16
2.6.5 Mutasi .....	17
2.6.6 Elitisme .....	18
2.6.7 Penggantian Populasi .....	18
2.7 Saham .....	18



<b>BAB III HASIL DAN PEMBAHASAN</b> .....	21
3.1 Pengumpulan Data .....	21
3.2 Normalisasi Data .....	22
3.3 Model PNN .....	23
3.4 Proses Perhitungan Mencari Nilai <i>Fitness</i> .....	25
3.4.1 Inisialisasi Populasi .....	29
3.4.2 Dekodekan Kromosom .....	29
3.4.3 Perhitungan <i>Fitness</i> .....	30
3.4.4 Penggantian Populasi .....	31
3.5 Proses Pemilihan Arsitektur Terbaik .....	33
3.6 Peramalan dengan Menggunakan GPNN .....	36
<b>BAB IV PENUTUP</b> .....	41
4.1 Kesimpulan .....	41
4.2 Saran .....	41
<b>DAFTAR PUSTAKA</b> .....	43
<b>LAMPIRAN</b> .....	45



## DAFTAR GAMBAR

	Halaman
<b>Gambar 2.1</b>	Jaringan lapis tunggal..... 4
<b>Gambar 2.2</b>	Jaringan lapis jamak..... 4
<b>Gambar 2.3</b>	Jaringan berulang..... 5
<b>Gambar 2.4</b>	Jaringan berlarik..... 6
<b>Gambar 2.5</b>	Fungsi identitas..... 8
<b>Gambar 2.6</b>	Fungsi <i>threshold</i> ..... 8
<b>Gambar 2.7</b>	Fungsi sigmoid biner..... 9
<b>Gambar 2.8</b>	Fungsi sigmoid bipolar..... 10
<b>Gambar 2.9</b>	Arsitektur PNN..... 12
<b>Gambar 2.10</b>	Ilustrasi <i>crossover</i> dua titik..... 17
<b>Gambar 2.11</b>	Proses mutasi..... 17
<b>Gambar 3.1</b>	Arsitektur PNN..... 23
<b>Gambar 3.2</b>	Diagram alir GPNN..... 29
<b>Gambar 3.3</b>	Diagram alir menghitung <i>fitness</i> pada PNN.... 26
<b>Gambar 3.4</b>	Proses pindah silang..... 32
<b>Gambar 3.5</b>	Proses pelatihan percobaan (2) Tabel 3.4 ..... 35
<b>Gambar 3.7</b>	Grafik keluaran jaringan dibandingkan dengan data asli ..... 36

## DAFTAR TABEL

	Halaman
<b>Tabel 2.1</b> Skema pengkodean <i>real number encoding</i> .....	14
<b>Tabel 2.2</b> Skema pengkodean <i>discrete decimal encoding</i> ....	15
<b>Tabel 2.3</b> Skema pengkodean <i>binary encoding</i> .....	15
<b>Tabel 3.1</b> Data harga saham harian PT Telkom.....	21
<b>Tabel 3.2</b> Data bulan Januari 2007 setelah dinormalisasi.....	22
<b>Tabel 3.3</b> Pemilihan arsitektur terbaik dengan GPNN linier	33
<b>Tabel 3.4</b> Pemilihan arsitektur terbaik dengan GPNN orde dua.....	34
<b>Tabel 3.5</b> Pemilihan arsitektur terbaik dengan GPNN orde tiga .....	34
<b>Tabel 3.6</b> Selisih harga keluaran jaringan dengan data sebenarnya .....	37



## DAFTAR LAMPIRAN

	Halaman
<b>Lampiran 1</b> Data harga saham harian PT Telkom yang diperdagangkan di Bursa Efek Indonesia pada bulan Januari 2007 sampai Maret 2011 .....	45
<b>Lampiran 2</b> Pengkodean dengan <i>binary encoding</i> .....	73
<b>Lampiran 3</b> Perhitungan dekode kromosom yang berisi <i>binary encoding</i> menjadi individu $x$ yang bernilai riil .....	75
<b>Lampiran 4</b> Langkah pelatihan data .....	83
<b>Lampiran 5</b> Program GPNN .....	85



# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Peran pasar saham dalam perekonomian Indonesia saat ini mulai melembaga. Saat ini pembelian saham menjadi salah satu pilihan modal yang sah, selain bentuk modal lainnya seperti uang, tanah, dan emas. Oleh karena itu, prediksi harga saham bermanfaat bagi *investor* untuk melihat prospek investasi saham pada suatu perusahaan di masa yang akan datang. Terdapat banyak metode kuantitatif dalam prediksi yaitu metode sederhana dan metode yang kompleks. Termasuk dalam metode kompleks adalah *Polynomial Neural Network* (PNN). PNN adalah jaringan saraf tiruan (JST) yang setiap lapisannya merupakan suatu fungsi polinomial. Keluaran dari masing-masing lapisan diperoleh dengan menggunakan beberapa tipe polinomial seperti linier, kuadrat, dan kubik. Setiap fungsi polinomial memiliki koefisien yang dapat diubah.

Metode yang sering digunakan dalam proses pelatihan PNN adalah *backpropagation*. *Backpropagation* merupakan salah satu metode optimasi berdasarkan *gradient*. Ketepatan *backpropagation* dalam meminimumkan galat dipengaruhi oleh pengambilan titik awal. Pengambilan titik awal yang kurang tepat menyebabkan galat tidak minimum secara global. Oleh karena itu, diperlukan metode yang dapat meminimumkan galat secara global, salah satunya yaitu Algoritma Genetika (AG).

Dalam skripsi ini diperkenalkan *Genetic Polynomial Neural Network* (GPNN) yaitu PNN yang menggunakan AG dalam meminimumkan galat. Optimasi bobot dengan AG akan membuat galat yang diperoleh semakin kecil, sehingga nilai peramalan semakin baik merepresentasikan nilai aktualnya.



## 1.2 Rumusan Masalah

Berdasarkan latar belakang tersebut dapat dirumuskan pokok permasalahan dalam skripsi ini, yaitu

- a. bagaimana menerapkan AG untuk mengoptimasi bobot pada PNN,
- b. bagaimana hasil prediksi harga penutupan saham harian PT Telekomunikasi Indonesia Tbk. menggunakan GPNN.

## 1.3 Batasan Masalah

Agar dalam penyusunan skripsi ini lebih fokus, diperlukan batasan-batasan masalah berikut ini

- a. arsitektur jaringan yang digunakan adalah jaringan lapis jamak dengan satu lapisan tersembunyi,
- b. skema pengkodean menggunakan *binary encoding*.

## 1.4 Tujuan

Tujuan penulisan skripsi ini antara lain untuk

- a. menerapkan AG untuk mengoptimasi bobot pada PNN,
- b. memprediksi harga penutupan saham harian PT Telekomunikasi Indonesia Tbk. menggunakan GPNN.

## 1.5 Manfaat

Studi ini bermanfaat bagi para pembaca untuk lebih mengetahui dan memahami PNN yang diaplikasikan untuk peramalan. Selain itu juga diperkenalkan AG untuk meningkatkan kinerja PNN. Manfaat bagi pihak *investor* yaitu hasil dari penelitian ini diharapkan dapat menjadi tambahan bahan evaluasi untuk mengambil keputusan investasi saham dan memberikan pandangan mengenai prediksi pergerakan harga saham.

## BAB II

### TINJAUAN PUSTAKA

#### 2.1 Jaringan Saraf Tiruan (JST)

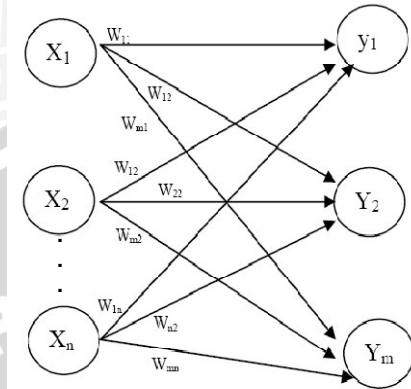
Jaringan Saraf Tiruan (JST) merupakan salah satu representasi buatan dari otak manusia yang mencoba untuk mensimulasikan proses pembelajaran otak manusia. Istilah buatan digunakan karena JST ini diimplementasikan dengan menggunakan program komputer yang mampu menyelesaikan sejumlah proses perhitungan selama proses pelatihan. Seperti halnya otak manusia, JST juga terdiri dari beberapa neuron, dan terdapat hubungan antara neuron-neuron tersebut. Neuron-neuron mentransformasikan informasi yang diterima menuju neuron-neuron yang lain. Pada JST, hubungan ini dikenal dengan bobot. Neuron-neuron buatan tersebut bekerja dengan cara yang sama pula dengan neuron-neuron biologis. Masukan akan dikirim ke neuron dengan bobot kedatangan tertentu. Masukan akan diproses oleh suatu fungsi perambatan yang menjumlahkan semua bobot yang datang. Selanjutnya, hasil penjumlahan ini dibandingkan dengan suatu nilai ambang tertentu melalui fungsi aktivasi setiap neuron. Apabila masukan melewati suatu nilai ambang tertentu, maka neuron tersebut akan diaktifkan sehingga neuron tersebut akan mengirimkan keluaran. (Kusumadewi, 2004).

Berdasarkan arsitekturnya, model JST digolongkan menjadi:

- Jaringan lapis tunggal (*single layer network*)
- Jaringan lapis jamak (*multiple layer network*)
- Jaringan berulang (*recurrent network*)
- Jaringan berlarik (*lattice network*)

##### 2.1.1 Jaringan Lapis Tunggal

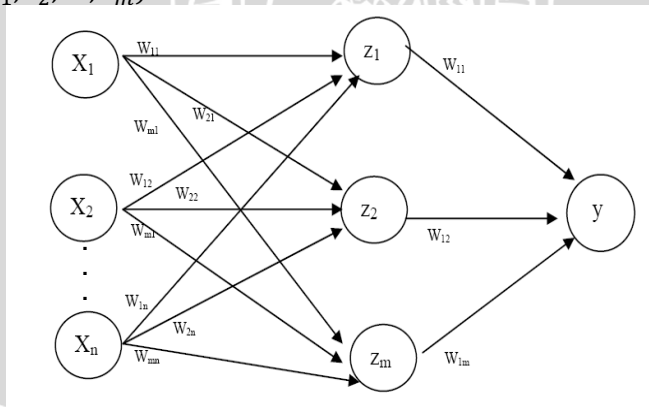
Jaringan dengan lapisan tunggal hanya memiliki satu lapisan dengan bobot-bobot terhubung. Jaringan ini hanya menerima masukan dan secara langsung akan mengelolanya menjadi keluaran tanpa harus melalui lapisan tersembunyi. Lapisan masukan terdiri dari  $n$  neuron, yaitu  $(x_1, x_2, \dots, x_n)$ . Sedangkan lapisan keluaran memiliki  $m$  neuron yaitu  $(y_1, y_2, \dots, y_m)$ . Neuron dari kedua lapisan saling berhubungan dengan bobot tertentu.



**Gambar 2.1** Jaringan lapis tunggal

### 2.1.2 Jaringan Lapis Jamak

Jaringan dengan banyak lapisan memiliki satu atau lebih lapisan yang terletak di antara lapisan masukan dan lapisan keluaran. Umumnya, ada lapisan bobot-bobot yang terletak antara dua lapisan yang bersebelahan. Jaringan dengan banyak lapisan ini dapat menyelesaikan permasalahan yang lebih sulit daripada lapisan tunggal, tentu saja dengan pelatihan yang lebih rumit. Gambar 2.2 memperlihatkan jaringan dengan  $n$  buah unit masukan ( $x_1, x_2, \dots, x_n$ ), sebuah layar tersembunyi yang terdiri dari  $m$  buah unit ( $z_1, z_2, \dots, z_m$ ) dan satu buah unit keluaran.

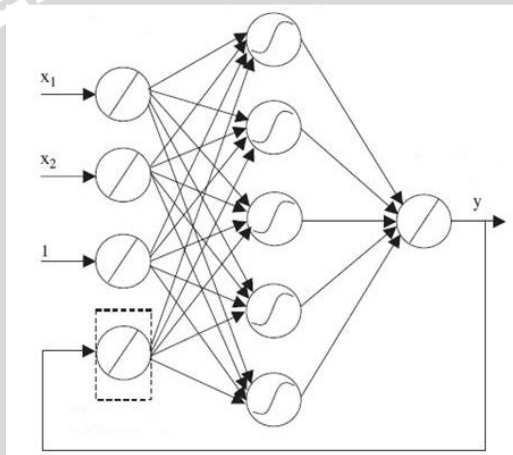


**Gambar 2.2** Jaringan lapis jamak

### 2.1.3 Jaringan Berulang

Model jaringan berulang mirip dengan jaringan layar tunggal ataupun jamak. Hanya saja, ada simpul keluaran yang memberikan sinyal pada unit masukan. Dengan kata lain sinyal mengalir dua arah, yaitu maju dan mundur.

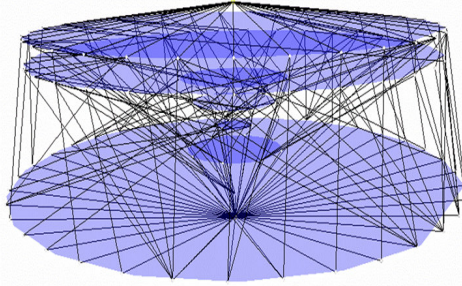
Berbeda dari jaringan layar tunggal maupun jamak yang hanya mempunyai satu arah perhitungan, jaringan berulang mempunyai paling sedikit satu kali umpan-balik. Adanya umpan-balik dimaksudkan untuk meningkatkan kemampuan belajar jaringan sehingga akan meningkatkan kinerjanya.



**Gambar 2.3** Jaringan berulang

### 2.1.4 Jaringan Berlarik

Suatu jaringan berlarik terdiri atas beberapa deret neuron yang terhubung pada suatu himpunan masukan. Ukuran jaringan berlarik menunjukkan jumlah neuron yang tersusun dalam deret yang bersangkutan. Seperti yang ditunjukkan Gambar 2.4, setiap simpul masukan dihubungkan dengan semua neuron dalam jaringan.



**Gambar 2.4** Jaringan berlirik

(Hidayatno, 2010)

## 2.2 Pelatihan JST

JST akan mencoba mensimulasikan kemampuan otak manusia untuk belajar. Tidak seperti model biologis, JST memiliki struktur yang tidak dapat diubah, dibangun oleh sejumlah neuron, dan memiliki nilai tertentu yang menunjukkan seberapa besar koneksi antara neuron, yang dikenal dengan nama bobot. Perubahan yang terjadi selama proses pelatihan adalah perubahan nilai bobot. Nilai bobot akan bertambah jika informasi yang diberikan oleh neuron yang bersangkutan tersampaikan, sebaliknya jika informasi tidak disampaikan oleh suatu neuron ke neuron yang lain maka, nilai bobot yang menghubungkan keduanya akan berkurang.

Ada dua jenis pelatihan dalam sistem JST, yaitu proses pelatihan terawasi (*supervised learning*) dan proses pelatihan tak terawasi (*unsupervised learning*). Pelatihan terawasi merupakan metode pelatihan pada JST di mana keluaran yang diharapkan telah diketahui sebelumnya, seolah-olah ada “guru” yang mengajarkan jaringan. Pada proses pelatihan, satu pola masukan akan diberikan ke satu neuron pada lapisan masukan. Pola tersebut akan dirambatkan ke setiap JST hingga sampai ke neuron pada lapisan keluaran. Lapisan keluaran membangkitkan pola keluaran yang akan dicocokkan dengan pola target keluarannya. Jika terjadi perbedaan antara pola keluaran hasil pelatihan dengan pola target, maka akan terjadi galat. Jika galat yang terjadi masih cukup besar, maka masih diperlukan lagi pelatihan.



Dalam proses belajar tak terawasi, tidak ada “guru” yang mengajari jaringan. Tidak diperlukan target keluaran pada metode pelatihan yang tak terawasi. Jaringan akan memodifikasi bobot sehingga untuk masukan yang hampir sama, keluaran yang dihasilkan sama. Dalam pelatihan jaringan, selain ada data pelatihan, juga ada data untuk pengujian. Data pelatihan, selain digunakan untuk pelatihan, juga digunakan untuk memantau besarnya galat yang terjadi antara keluaran yang dihasilkan jaringan dengan keluaran yang diharapkan. Data pelatihan akan mempengaruhi proses pelatihan. Sedangkan, data untuk pengujian dipakai dalam perhitungan untuk mengetahui keluaran yang dihasilkan jaringan karena adanya data masukan. Dalam data untuk pengujian tersebut dapat diketahui seberapa banyak jaringan telah belajar dari pelatihan dan apakah jaringan telah mampu untuk menghasilkan keluaran yang benar dengan bobot yang sudah ada. Data keluaran pada data untuk pengujian hanya digunakan untuk menentukan besarnya galat dengan cara membandingkan keluaran jaringan dengan keluaran yang diharapkan dan tidak mempengaruhi proses pelatihan. Tidak ada aturan khusus tentang banyaknya data pelatihan dan data untuk pengujian. Jadi, data pelatihan dan data untuk pengujian diambil secukupnya sesuai dengan kebutuhan. (Setiawan, 2003)

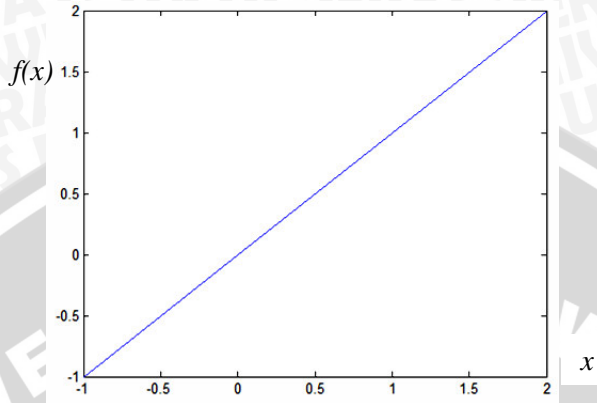
### 2.3 Fungsi Aktivasi

Fungsi aktivasi ada yang merupakan fungsi linier dan ada juga yang nonlinier. Untuk jaringan *multilayer*, fungsi aktivasi yang digunakan adalah fungsi aktivasi nonlinier, karena jika digunakan fungsi yang linier, hasilnya tidak akan berbeda dengan hasil pada jaringan *single layer*. Berikut fungsi-fungsi aktivasi yang biasa digunakan dalam sistem JST.

#### 1. Fungsi Identitas

$$f(x) = x, \text{ untuk semua } x \quad (2.1)$$

Fungsi identitas merupakan fungsi aktivasi untuk semua unit masukan. Bentuk fungsi identitas diberikan pada Gambar 2.5.

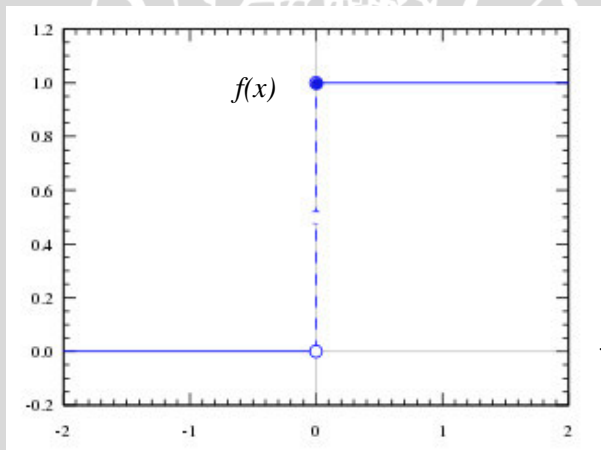


**Gambar 2.5** Fungsi identitas

2. Fungsi *Threshold*

$$f(x) = \begin{cases} 1 & \text{jika } x \geq \theta \\ 0 & \text{jika } x < \theta \end{cases} \quad (2.2)$$

Fungsi *threshold* sering dipakai pada jaringan *single layer*. Bentuk fungsi *threshold* dengan  $\theta = 0$  dapat dilihat pada Gambar 2.6.



**Gambar 2.6** Fungsi *Threshold*

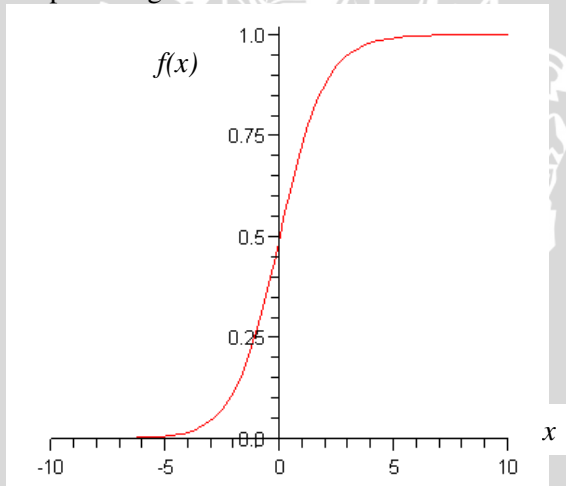
### 3. Fungsi Sigmoid Biner

$$f(x) = \frac{1}{1 + \exp(-\sigma x)} \quad (2.3)$$

$$f'(x) = \sigma f(x)[1 - f(x)] \quad (2.4)$$

dimana  $\sigma$  adalah parameter kecuraman fungsi.

Fungsi sigmoid biner merupakan fungsi yang sering digunakan dalam jaringan dengan metode pelatihan *backpropagation* karena bentuk fungsi aktivasi dan turunan fungsinya sederhana sehingga mudah dihitung. Fungsi sigmoid biner mempunyai nilai pada interval di antara 0 dan 1. Fungsi tersebut digunakan jika keluaran yang diinginkan terletak di antara 0 dan 1. Bentuk fungsi sigmoid biner yang sering digunakan adalah  $\sigma = 1$  untuk mempermudah perhitungan.



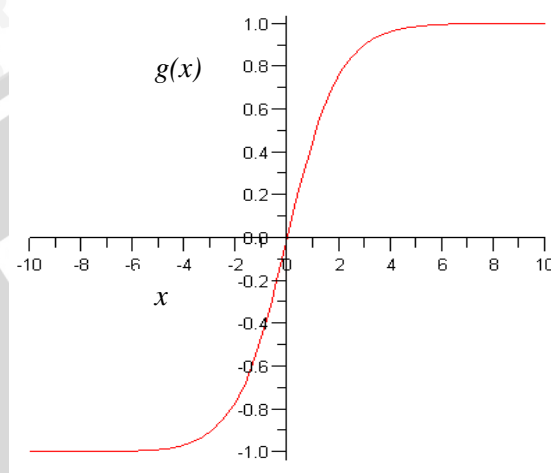
**Gambar 2.7** Fungsi sigmoid biner

### 4. Fungsi Sigmoid Bipolar

$$g(x) = \frac{1 - \exp(-\sigma x)}{1 + \exp(-\sigma x)} \quad (2.5)$$

$$g'(x) = \frac{\sigma}{2} [1 + g(x)][1 - g(x)] \quad (2.6)$$

Fungsi sigmoid bipolar adalah fungsi sigmoid biner yang nilai fungsinya berada pada interval -1 dan 1. Bentuk fungsi sigmoid bipolar diberikan pada Gambar 2.8.



**Gambar 2.8** Fungsi sigmoid bipolar

(Setiawan, 2003)

## 2.4 Ukuran Kesalahan Peramalan

Untuk mengevaluasi nilai parameter peramalan, digunakan ukuran kesalahan peramalan. Harga parameter peramalan yang terbaik adalah harga yang memberikan nilai kesalahan peramalan yang terkecil. Terdapat berbagai macam ukuran kesalahan yang dapat diklasifikasikan menjadi ukuran standar dalam statistik dan ukuran relatif. Ukuran kesalahan yang termasuk ukuran standar statistik adalah nilai rata-rata kesalahan (*mean error*), nilai rata-rata kesalahan absolut (*mean absolute error*), dan nilai rata-rata kesalahan kuadrat (*mean squared error*). Ukuran kesalahan yang termasuk ukuran relatif adalah nilai rata-rata kesalahan persentase (*mean percentage error*) dan nilai rata-rata kesalahan persentase absolut (*mean absolute percentage error*).

1. Nilai Rata-rata Kesalahan (*Mean Error*)

$$ME = \frac{\sum_{i=1}^n e_i}{n} \quad (2.7)$$

$$e_i = X_i - F_i \quad (2.8)$$

dengan,

$ME$  : nilai rata-rata kesalahan

$n$  : jumlah periode waktu data

$e_i$  : kesalahan pada periode waktu  $i$

$X_i$  : data pada periode waktu  $i$

$F_i$  : ramalan untuk periode waktu  $i$

2. Nilai Rata-rata Kesalahan Absolut (*Mean Absolute Error*)

$$MAE = \frac{\sum_{i=1}^n |e_i|}{n} \quad (2.9)$$

dengan,  $MAE$  : nilai rata-rata kesalahan absolut

3. Nilai Rata-rata Kesalahan Kuadrat (*Mean Squared Error*)

$$MSE = \frac{\sum_{i=1}^n e_i^2}{n} \quad (2.10)$$

dengan,  $MSE$ : nilai rata-rata kesalahan kuadrat

4. Nilai Rata-rata Kesalahan Persentase (*Mean Percentage Error*)

$$MPE = \frac{\sum_{i=1}^n PE_i}{n} \quad (2.11)$$

$$PE_i = \frac{X_i - F_i}{X_i} 100\% \quad (2.12)$$

dengan,

$PE_i$  : kesalahan persentase pada periode  $i$

$MPE$  : nilai rata-rata kesalahan persentase

5. Nilai Rata-rata Kesalahan Persentase Absolut (*Mean Absolute Percentage Error*)

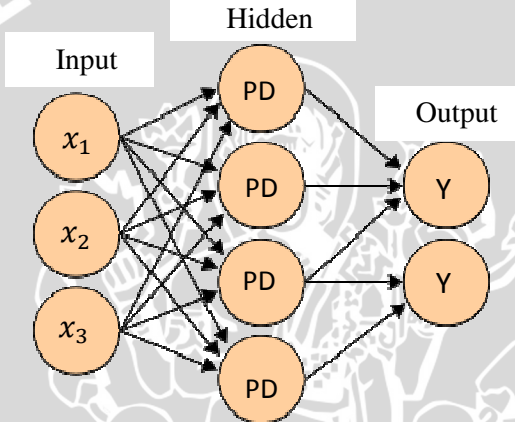
$$MAPE = \frac{\sum_{i=1}^n |PE_i|}{n} \quad (2.13)$$

dengan,  $MAPE$  : nilai rata-rata kesalahan persentase absolut



Dalam mengevaluasi harga parameter peramalan dapat dipilih satu ukuran kesalahan peramalan tersebut. Akan tetapi, ukuran nilai rata-rata kesalahan dan nilai rata-rata kesalahan persentase jarang digunakan, karena pada kedua ukuran tersebut dimungkinkan diperoleh nilai kesalahan negatif pada suatu periode peramalan. Hal ini dapat menyebabkan harga kedua ukuran kesalahan tersebut kecil karena kesalahan negatif tersebut dapat meniadakan kesalahan positif yang ada, meskipun sebenarnya harga kesalahan yang terjadi cukup besar. (Ai, 1999)

## 2.5 Polynomial Neural Network (PNN)



**Gambar 2.9** Arsitektur PNN

*Polynomial Neural Network* (PNN) adalah JST yang setiap neuronnya pada lapisan tersembunyi dan lapisan keluaran merupakan suatu fungsi polinomial. Gambar 2.9 merupakan arsitektur PNN, di mana  $x_1, x_2, \dots, x_n$  merupakan masukan dari fungsi polinomial. Fungsi polinomial ini disebut *partial description* (PD).

Berikut merupakan langkah-langkah yang dilakukan dalam membuat PNN.

Langkah 1. Menentukan variabel masukan ( $x_1, x_2, \dots, x_n$ ).

Langkah 2. Menyiapkan data pelatihan.

Langkah 3. Menentukan struktur dan orde PNN.

Struktur PNN terdiri atas beberapa macam.

a. *Basic* : nilai dari variabel masukan dari setiap *partial description* (PD) sama pada setiap lapisan.

- Kasus 1 : orde polinomial dari PD sama untuk masing-masing lapisan pada jaringan.
- Kasus 2 : orde polinomial dari PD pada lapisan kedua atau lapisan yang lebih tinggi mempunyai orde yang berbeda.
- b. *Modified* : nilai dari variabel masukan dari PD berbeda dari lapisan ke lapisan.
- Kasus 1 : orde polinomial dari PD sama pada masing-masing lapisan.
- Kasus 2 : orde polinomial dari PD pada lapisan ke-dua atau lapisan yang lebih tinggi berbeda.

Beberapa contoh fungsi polinomial yang digunakan adalah sebagai berikut.

Polinomial orde dua untuk dua masukan:

$$y = A + Bx_1 + Cx_2 + Dx_1^2 + Ex_2^2 + Fx_1x_2.$$

Polinomial orde tiga untuk dua masukan:

$$y = A + Bx_1 + Cx_2 + Dx_1^2 + Ex_2^2 + Fx_1x_2 + Gx_1^3 + Hx_2^3.$$

Polinomial orde tiga untuk tiga masukan:

$$y = A + Bx_1 + Cx_2 + Dx_3 + Ex_1^2 + Fx_2^2 + Gx_3^2 + Hx_1x_2 + Ix_1x_3 + Jx_2x_3 + Kx_1x_2x_3 + Lx_1^3 + Mx_2^3 + Nx_3^3.$$

Langkah 4. Melakukan pelatihan data. Untuk perhitungan lapisan tersembunyi dan keluaran, digunakan *mean sigmoid* biner dengan persamaan (2.3)

Langkah 5. Menghitung galat yang ada pada masing-masing keluaran. Untuk menghitung galat digunakan persamaan (2.10). Setelah melakukan perhitungan galat, dilakukan pelatihan sampai iterasi yang diinginkan dan berhenti jika sampai pada iterasi maksimum. (Setiawan, 2003)

## 2.6 Algoritma Genetika (AG)

AG adalah algoritma pencarian solusi dari suatu persoalan berdasarkan cara kerja melalui mekanisme seleksi alam dan genetika. Tujuannya untuk menentukan struktur-struktur yang disebut individu berkualitas tinggi di dalam suatu domain yang disebut populasi untuk mendapatkan solusi suatu persoalan tersebut. John Holland

mengembangkan AG melalui prosedur iteratif untuk mengatur populasi individu yang merupakan calon solusi.

Evolusi merupakan proses atau perubahan yang terus-menerus untuk menuju suatu kesempurnaan atau keseimbangan. Dengan adanya spesies yang mempunyai nilai *fitness*, maka seleksi dilakukan berdasarkan nilai *fitness*-nya. Gen adalah sifat penting yang akan dibawa oleh generasi atau keturunan berikutnya. Gen dapat dioperasikan secara genetik untuk memperoleh keanekaragaman melalui proses reproduksi, *crossover*, dan mutasi. Seleksi terjadi pada setiap generasi, sedangkan peningkatan keanekaragaman terjadi pada setiap generasi baru. Jadi, ide dasar AG adalah mempertahankan struktur populasi atau kumpulan spesies yang mewakili calon solusi suatu persoalan. Dengan memanfaatkan sifat-sifat sistem biologis, AG dapat memberikan hasil yang memuaskan dalam proses optimasi. (Pandjaitan, 2007)

Pada dasarnya AG memiliki tujuh komponen yang harus dipenuhi, yaitu skema pengkodean, nilai *fitness*, seleksi, pindah silang, mutasi, elitisme, penggantian populasi.

### 2.6.1 Skema Pengkodean

Terdapat tiga skema yang paling umum digunakan dalam pengkodean, yaitu *real number encoding*, *discrete decimal encoding*, dan *binary encoding*. Pada skema pengkodean dimisalkan terdapat tiga variabel, yaitu  $x_1$ ,  $x_2$ , dan  $x_3$  yang dikodekan ke dalam sebuah kromosom yang terdiri dari tiga gen untuk *real number encoding*. Sedangkan pada *discrete decimal encoding* maupun *binary encoding*, ketiga variabel dikodekan ke dalam kromosom yang terdiri dari sembilan gen, dimana masing-masing variabel dikodekan ke dalam tiga gen.

#### a. *Real number encoding*

Pada skema ini, nilai gen berada pada interval  $[0,R]$ , di mana  $R$  adalah bilangan riil positif dan biasanya  $R=1$ . Contoh *real number encoding* dapat dilihat pada Tabel 2.1.

**Tabel 2.1** Skema pengkodean *real number encoding*

$x_1$	$x_2$	$x_3$
0,2390	1,0000	0,0131
$g_1$	$g_2$	$g_3$

b. *Discrete decimal encoding*

Pada Tabel 2.2 berikut diberikan skema pengkodean *discrete decimal encoding* di mana untuk setiap gen dapat bernilai salah satu dari bilangan bulat dalam interval [0,9].

**Tabel 2.2** Skema pengkodean *discrete decimal encoding*

$x_1$			$x_2$			$x_3$		
2	3	9	9	9	9	0	1	3
$g_1$	$g_2$	$g_3$	$g_4$	$g_5$	$g_6$	$g_7$	$g_8$	$g_9$

c. *Binary encoding*

Pada Tabel 2.3 berikut diberikan contoh skema pengkodean *binary encoding* dimana untuk setiap gen hanya dapat bernilai nol dan satu.

**Tabel 2.3** Skema pengkodean *binary encoding*

$x_1$			$x_2$			$x_3$		
0	1	0	1	1	1	0	0	0
$g_1$	$g_2$	$g_3$	$g_4$	$g_5$	$g_6$	$g_7$	$g_8$	$g_9$

Setelah mendapatkan skema pengkodean, dilakukan dekode kromosom yang berisi *real number encoding*, *discrete decimal encoding*, dan *binary encoding* menjadi individu  $x$  yang bernilai riil dalam interval yang diinginkan untuk dievaluasi ke dalam fungsi yang dicari nilai optimalnya.

### 2.6.2 Nilai *Fitness*

Suatu individu dievaluasi berdasarkan suatu fungsi tertentu sebagai ukuran performansinya. Di dalam evolusi alam, individu yang bernilai *fitness* tinggi yang akan bertahan hidup. Sedangkan individu yang bernilai rendah akan mati. Pada masalah optimasi, nilai *fitness* yang digunakan antara lain:

- a. untuk masalah maksimasi, nilai *fitness* ditentukan oleh fungsi berikut:

$$f = h \quad (2.14)$$

- b. untuk masalah minimasi, nilai *fitness* ditentukan oleh fungsi berikut:

$$f = \frac{1}{h + a} \quad , \quad h \neq a \quad (2.15)$$

dimana,



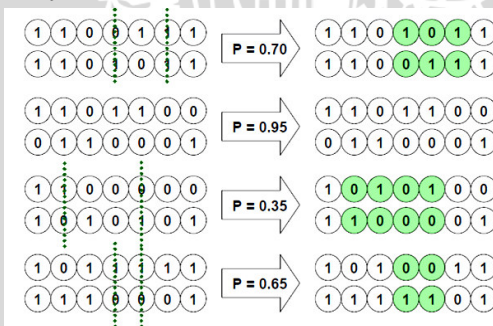
- $h$  = nilai fungsi objektif
- $a$  = bilangan sangat kecil, untuk menghindari pembagian dengan nol.

### 2.6.3 Seleksi Orang Tua

Pemilihan dua buah kromosom sebagai orang tua, yang akan dipindah-silangkan, biasanya dilakukan secara proporsional sesuai dengan nilai *fitness*-nya. Suatu metode seleksi yang umum digunakan adalah *roulette wheel* (roda *roulette*). Sesuai dengan namanya, metode ini menirukan permainan *roulette wheel* di mana masing-masing kromosom menempati potongan lingkaran pada roda *roulette* secara proporsional sesuai dengan nilai *fitness*-nya. Kromosom yang memiliki nilai *fitness* lebih besar menempati potongan lingkaran yang lebih besar dibandingkan dengan kromosom bernilai *fitness* rendah.

### 2.6.4 Crossover

*Crossover* adalah operator genetik buah *string* menurut nilai kecocokan yang terbaik. Fungsinya adalah menghasilkan *string* baru yang serupa dengan menggabungkan bagian dari kedua *string* ciri yang utama. Lebih lanjut, *crossover* merupakan perkawinan dua orangtua utama dan disebut pula proses rekombinasi. Proses rekombinasi bertujuan menambah keanekaragaman individu dalam populasi dengan mengawinkan individu-individu pada populasi. Pada Gambar 2.10 merupakan contoh proses *crossover* yang menghasilkan keturunan berupa individu-individu baru untuk ditempatkan pada populasi berikutnya. Individu baru akan memiliki bagian struktur kedua orang tuanya.

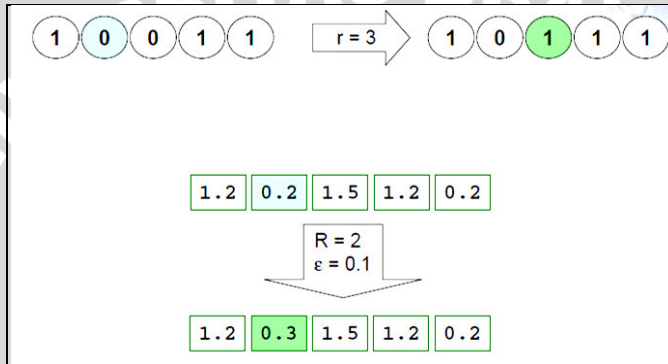


**Gambar 2.10** Ilustrasi *crossover* dua titik



### 2.6.5 Mutasi

Mutasi menyebabkan terciptanya individu baru melalui modifikasi terhadap satu atau lebih nilai gen pada individu yang sama. Mutasi berguna pula untuk menghindarkan kehilangan total materi yang tak dapat dikembalikan setelah reproduksi dan pindah silang dan agar probabilitas pencarian dalam sebarang daerah tidak pernah nol. Cara mutasi pada Gambar 2.11 adalah dengan mengubah suatu *bit* dari suatu representasi biner gen tertentu.



**Gambar 2.11** Proses mutasi

### 2.6.6 Elitisme

Proses ini bertujuan untuk menjaga agar individu bernilai *fitness* tertinggi tidak hilang selama evolusi dengan membuat satu atau beberapa kopi individu. Karena selama proses seleksi dilakukan secara acak, maka tidak ada jaminan bahwa suatu individu bernilai *fitness* tertinggi akan selalu dipilih.

### 2.6.7 Penggantian Populasi

Ukuran populasi menyatakan banyaknya individu dalam satu generasi. Setelah dilakukan proses seleksi, pindah silang dan mutasi, kemudian dilakukan penggantian populasi. Skema penggantian populasi ini disebut *generational replacement*, yang berarti semua individu (misal  $n$  individu dalam satu populasi) dari suatu generasi digantikan sekaligus oleh  $n$  individu baru hasil pindah silang dan mutasi. (Suyanto, 2005)

## 2.7 Saham

Saham adalah sebuah surat berharga yang dikeluarkan oleh sebuah perusahaan yang berbentuk Perseroan Terbatas (emiten) yang menyatakan bahwa pemilik saham tersebut adalah juga pemilik sebagian dari perusahaan itu. Saham merupakan seceruk kertas yang menunjukkan hak pemodal untuk memperoleh bagian dari prospek atau kekayaan organisasi yang menerbitkan sekuritas tersebut dan berbagai kondisi yang memungkinkan pemodal tersebut menjalankan haknya.

Dalam transaksi saham, investor mengenal beberapa bentuk harga, yaitu harga pasar, harga pembukaan dan harga penutupan. Harga pasar merupakan harga yang dibentuk pasar, sebuah harga yang pembentukannya melalui sistem lelang secara terus menerus sehingga akan diperoleh harga jual dan harga beli dengan harga yang sama. Pertemuan harga jual dan beli yang sama disebut sebagai harga pasar dari saham yang terjadi. Proses tawar-menawar akan terus terjadi hingga berakhirnya jam perdagangan. Jadi transaksi yang terjadi terus berulang, dalam satu hari bursa akan terjadi puluhan bahkan ratusan kali harga yang disepakati oleh masing-masing investor untuk saham yang sama. Harga yang terbentuk pada jam berakhir disebut sebagai harga penutupan dan harga penutupan sebuah saham tidak selalu menjadi patokan bagi harga pembukaan pasar. Harga pembukaan merupakan penentuan harga yang paling pantas bagi saham tertentu setelah penutupan sehari sebelumnya itu. Selanjutnya, didapatkan harga saham baru yang menjadi patokan investor bagi harga pembukaan saham pada perdagangan hari ini. Jadi, harga pembukaan saham bisa sama atau berbeda dengan harga penutupan sehari sebelumnya. (Anonymous, 2011)

Secara umum semakin baik keuangan perusahaan dan semakin banyak keuntungan yang dinikmati oleh pemegang saham, kemungkinan harga saham akan naik. Tetapi saham yang memiliki tingkat keuntungan yang baik juga dapat mengalami penurunan harga. Hal ini dapat disebabkan oleh keadaan pasar saham. Hal seperti ini tidak akan hilang jika kepercayaan pemodal belum pulih, kondisi ekonomi belum membaik ataupun hal-hal lain yang membaik. Salah satu resiko dari pemegang saham adalah menurunnya harga saham. Hal ini dapat diatasi dengan cara menahan saham tersebut sampai keadaan pasar membaik. (Ardiani, 2007)

## BAB III HASIL DAN PEMBAHASAN

### 3.1 Pengumpulan Data

Data yang digunakan dalam skripsi ini adalah data harga saham harian PT Telekomunikasi Indonesia Tbk yang diperdagangkan di Bursa Efek Indonesia. Data yang diperoleh adalah data saham harian pada bulan Januari 2007 sampai dengan bulan Maret 2011 berjumlah 874 data. Data tersebut dibagi menjadi tiga bagian, yaitu 733 data akan digunakan untuk pelatihan, 79 data untuk pengujian, dan 62 data baru yaitu harga saham harian pada bulan Januari sampai bulan Maret 2011 untuk proses peramalan.

**Tabel 3.1** Data harga saham harian PT Telkom

Data ke-	Tanggal	Masukan			Keluaran
		<i>open</i>	<i>high</i>	<i>low</i>	<i>close</i>
1	02-01-07	10.200	10.350	10.200	10.350
2	03-01-07	10.350	10.400	10.250	10.350
3	04-01-07	10.400	10.450	10.150	10.150
4	05-01-07	10.150	10.150	10.050	10.150
5	08-01-07	10.050	10.100	10.000	10.000
6	09-01-07	10.000	10.150	9.950	9.950
7	10-01-07	9.850	9.900	9.600	9.700
8	11-01-07	9.750	9.800	9.550	9.650
9	12-01-07	9.750	9.800	9.300	9.600
10	15-01-07	9.750	9.900	9.700	9.850
...	...	...	...	...	...
...	...	...	...	...	...
873	30-03-11	7.000	7.250	7.000	7.200
874	31-03-11	7.250	7.350	7.150	7.350

*sumber: www.telkom.co.id*

Keterangan:

- open* = harga pembukaan
- high* = harga tertinggi
- low* = harga terendah
- close* = harga penutupan

Data harga saham harian lengkap dapat dilihat pada Lampiran 1

### 3.2 Normalisasi Data

Normalisasi data sangat dibutuhkan dalam JST yang menggunakan fungsi aktivasi sigmoid biner. Tujuan normalisasi yaitu untuk menghindari perhitungan dengan angka-angka besar yang menyebabkan proses perhitungan membutuhkan waktu lebih lama. Penskalaan dilakukan dengan menggunakan rumus.

$$skala = \frac{\text{nilai yang diskala} - \text{nilai minimum data}}{\text{nilai maksimum data} - \text{nilai minimum data}} \quad (3.1)$$

Tabel 3.2 adalah hasil normalisasi bulan Januari 2007 yang digunakan untuk perhitungan langsung. Kolom  $x_1$  adalah harga pembukaan,  $x_2$  adalah harga tertinggi,  $x_3$  adalah harga terendah, dan  $y$  adalah harga penutupan.

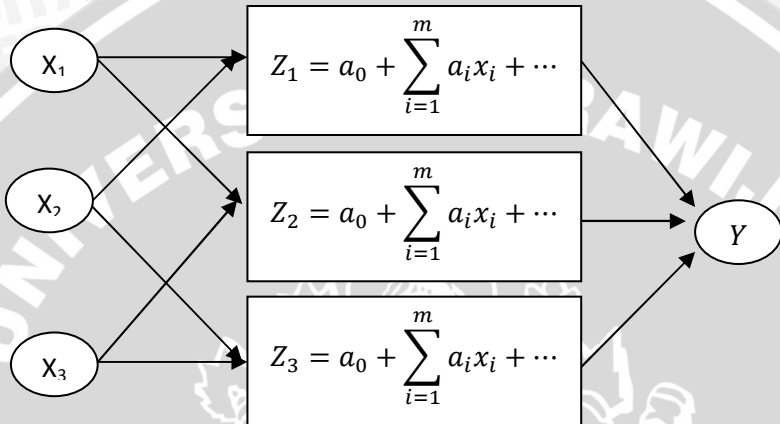
**Tabel 3.2** Data bulan Januari 2007 setelah dinormalisasi

Data ke-	Tanggal	Input			Target
		$x_1$	$x_2$	$x_3$	$y$
1	02-01-07	0,5778	0,5944	0,5778	0,5944
2	03-01-07	0,5944	0,6000	0,5833	0,5944
3	04-01-07	0,6000	0,6056	0,5722	0,5722
4	05-01-07	0,5722	0,5722	0,5611	0,5722
5	08-01-07	0,5611	0,5667	0,5556	0,5556
6	09-01-07	0,5556	0,5722	0,5500	0,5500
7	10-01-07	0,5389	0,5444	0,5111	0,5222
8	11-01-07	0,5278	0,5333	0,5056	0,5167
9	12-01-07	0,5278	0,5333	0,4778	0,5111
10	15-01-07	0,5278	0,5444	0,5222	0,5389
11	16-01-07	0,5444	0,5500	0,5389	0,5444
12	17-01-07	0,5444	0,5556	0,5444	0,5444
13	18-01-07	0,5444	0,5556	0,5444	0,5444
14	19-01-07	0,5444	0,5722	0,5444	0,5722
15	22-01-07	0,5722	0,5833	0,5611	0,5722
16	23-01-07	0,5722	0,5722	0,5500	0,5556
17	24-01-07	0,5556	0,5611	0,5389	0,5500
18	25-01-07	0,5444	0,5444	0,5056	0,5278
19	26-01-07	0,5111	0,5111	0,4889	0,5056
20	29-01-07	0,5111	0,5222	0,5056	0,5167
21	30-01-07	0,5167	0,5222	0,4944	0,5056
22	31-01-07	0,5111	0,5111	0,4889	0,4944

### 3.3 Model PNN

Pada penelitian ini digunakan PNN dengan langkah-langkah sebagai berikut

Langkah 1. Membuat arsitektur PNN seperti pada Gambar 3.1



**Gambar 3.1** Arsitektur PNN

Notasi:

$x_1$  = *open* (harga pembukaan)

$x_2$  = *high* (harga tertinggi)

$x_3$  = *low* (harga terendah)

$(z_1, z_2, z_3)$  = nilai pada lapisan tersembunyi

$(a_0, a_i, a_{ij}, a_{ijk}, \dots)$  adalah nilai bobot pada lapisan tersembunyi.

$Y$  = keluaran (harga penutupan)

Arsitektur PNN ini menggunakan tiga masukan, satu keluaran dan satu lapisan tersembunyi dengan tiga neuron pada lapisan tersembunyi tersebut, di mana  $x_1$ ,  $x_2$ , dan  $x_3$  merupakan masukan yang mempunyai koefisien dari fungsi polinomial. Hubungan masukan-keluaran diberikan oleh fungsi berikut

$$y = a_0 + \sum_{i=1}^m a_i x_i + \sum_{i=1}^m \sum_{j=i}^m a_{ij} x_i x_j + \sum_{i=1}^m \sum_{j=i}^m \sum_{k=j}^m a_{ijk} x_i x_j x_k + \dots \quad (3.2)$$

Langkah 2. Menyiapkan data pelatihan.



Bentuk pasangan masukan-keluaran yang akan digunakan sebagai data untuk melakukan pelatihan pada PNN. (lampiran 1)

Langkah 3. Menentukan struktur dan orde PNN.

Dalam menentukan struktur PNN digunakan rumus:

$3C_2 = 3 PD$  (tiga kombinasi dua)

Karena terdapat tiga masukan, maka diperoleh tiga neuron pada lapisan tersembunyi dan dua kombinasi pada setiap neuron.

Langkah 4. Menentukan inisialisasi nilai bobot.

Nilai bobot pada PNN didapatkan dari inisialisasi populasi pada AG.

Langkah 5. Menetapkan kondisi yang ingin dicapai

Kondisi yang ingin dicapai dalam menggunakan PNN adalah alat untuk memecahkan suatu permasalahan. Langkah ke-6 sampai dengan langkah ke-7 akan terus dilakukan sampai kondisi yang diinginkan terpenuhi. Jika sudah terpenuhi maka iterasi berhenti.

Langkah 6. Melakukan pelatihan data.

Untuk perhitungan  $z$  dan  $y$ , dapat digunakan *mean sigmoid biner* dengan persamaan 2.3.

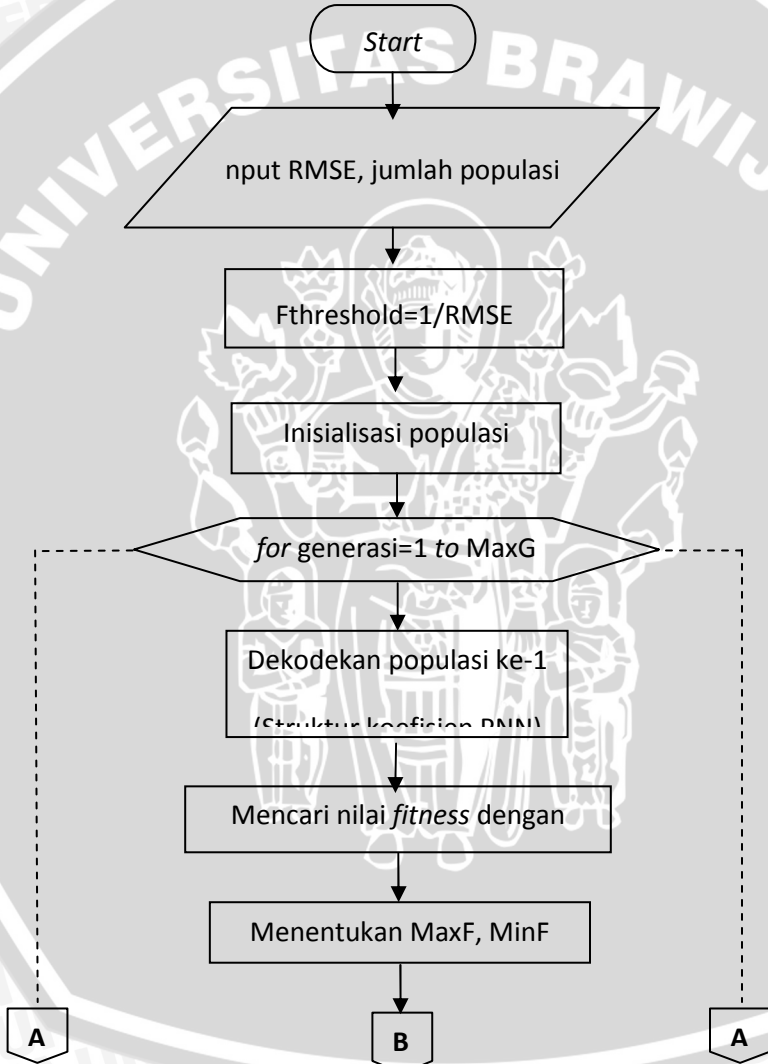
Langkah 7. Menghitung galat yang ada pada keluaran.

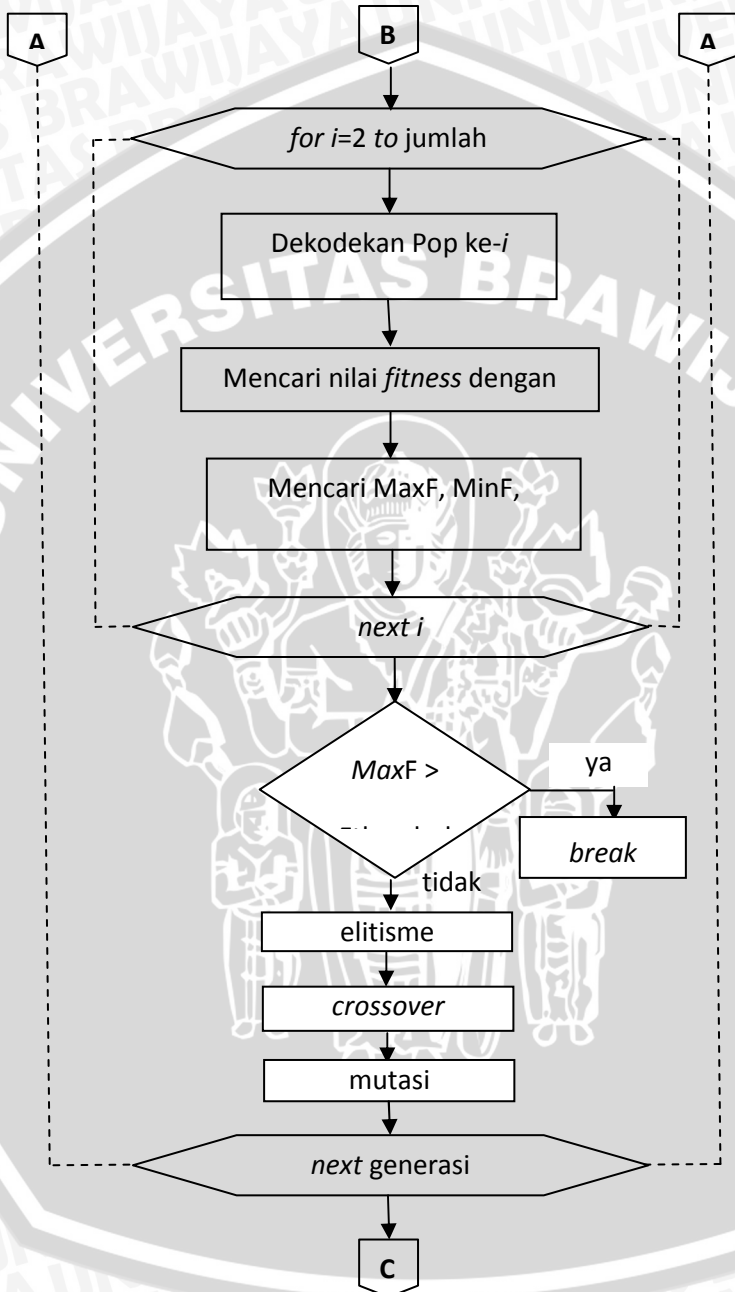
Arsitektur PNN dalam skripsi ini adalah jaringan lapis ganda. Arsitektur jaringan dibangun berdasarkan variabel-variabel yang mempengaruhi data harga saham harian. Variabel-variabel tersebut antara lain: harga pembukaan, harga tertinggi, dan harga terendah. Variabel yang digunakan sebagai keluaran jaringan adalah harga penutupan. Model arsitektur 3-3-1 berarti arsitektur jaringan menggunakan tiga neuron lapisan masukan, tiga neuron lapisan tersembunyi, dan satu neuron lapisan keluaran.

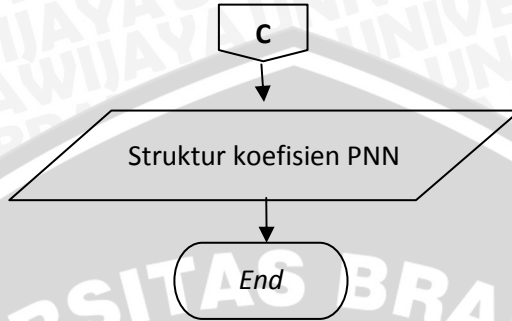
Fungsi aktivasi yang digunakan adalah fungsi sigmoid biner karena penskalaan data berada pada interval  $[0,1]$ . Sedangkan, dalam mengevaluasi harga parameter peramalan dipilih MSE agar tidak terhitung nilai kesalahan negatif, selain itu MSE memiliki keakuratan yang lebih baik dalam mendekati harga sebenarnya.

### 3.4 Proses Perhitungan Mencari Nilai *Fitness*

*Genetic Polynomial Neural Network* (GPNN) yaitu PNN yang menggunakan AG dalam meminimumkan galat. Pada arsitektur GPNN, AG digunakan untuk menentukan koefisien-koefisien PNN agar diperoleh galat yang minimum atau *fitness* maksimum, tahap-tahapnya yaitu





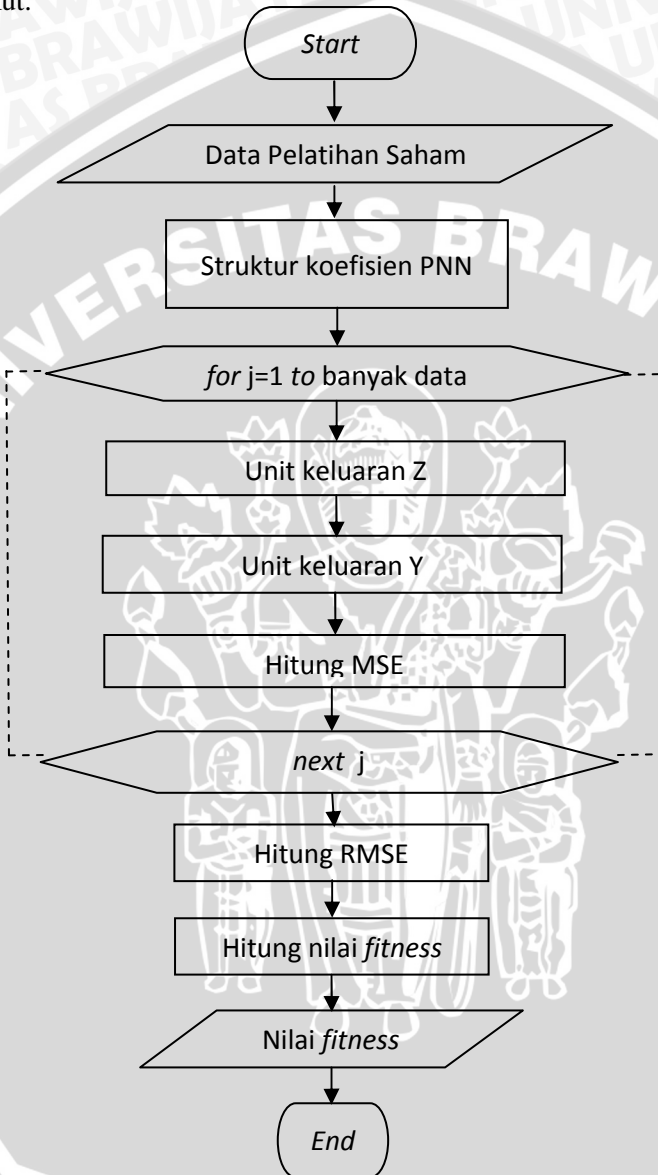


**Gambar 3.2** Diagram alir GPNN

maka keluaran yang diperoleh dalam pelatihan GPNN merupakan struktur koefisien dari PNN.



Sedangkan tahap menghitung nilai *fitness* dengan PNN sebagai berikut.



**Gambar 3.3** Diagram alir menghitung *fitness* pada PNN



Sebelum melakukan proses GPNN, dimasukkan data pelatihan dalam bentuk matriks dan dilakukan normalisasi data dengan persamaan 3.1. Selanjutnya dilakukan inisialisasi populasi, decode kromosom, penghitungan *fitness*, dan penggantian populasi.

### 3.4.1 Inisialisasi Populasi

Inisialisasi populasi pada GPNN ditentukan oleh jumlah gen dan ukuran populasi. Skema pengkodean yang digunakan adalah *binary encoding* sehingga setiap gen hanya dapat bernilai nol atau satu. Jumlah variabel (*nvar*) dihitung menggunakan persamaan (3.3).

$$Nvar = 3 * \sum_{i=1}^{d+1} i + \sum_{i=1}^{d+1} \sum_{j=1}^i j \quad (3.3)$$

di mana  $d$  adalah derajat polinomial.

Jika digunakan  $d = 2$ , maka  $Nvar$  yang didapat yaitu 28 variabel mewakili koefisien pada lapisan tersembunyi dan keluaran ( $a_0, a_1, a_2, a_3, \dots$ ) dan  $Nbit$  yaitu jumlah *bit* yang mengkodekan satu variabel, maka jumlah gen adalah  $Nvar$  dikalikan dengan  $Nbit$ . Sedangkan populasi adalah kumpulan kromosom berupa matriks berukuran ukuran populasi dikalikan jumlah gen. Setelah skema pengkodean ditentukan, AG diinisialisasi untuk sebuah populasi dengan  $n$  kromosom. Gen-gen yang mengisi masing-masing kromosom dibangkitkan secara *random*. Pada lampiran 2 dilakukan pengkodean dengan *binary encoding* dengan  $d = 2$  dan jumlah bit = 15.

### 3.4.2 Dekodekan Kromosom

Fungsi ini bertujuan untuk mengkodekan sebuah kromosom yang berisi bilangan biner menjadi individu  $x$  yang bernilai riil dalam interval yang diinginkan. Proses pengkodean dilakukan berdasarkan persamaan 3.4 berikut:

$$x = r_b + (r_a - r_b)(g_1 \times 2^{-1} + g_2 \times 2^{-2} + \dots + g_n \times 10^{-n}) \quad (3.4)$$

Pada fungsi ini istilah kromosom mengacu pada vektor baris yang berisi bilangan biner, sedangkan individu mengacu pada variabel  $x$  yang berisi bilangan riil.  $Nvar$  adalah jumlah

variabel yang terdapat pada fungsi yang dioptimasi. Sedangkan  $Nbit$  adalah jumlah  $bit$  yang digunakan untuk mengkodekan satu variabel.  $Ra$  adalah batas atas interval, sedangkan  $Rb$  adalah batas bawah interval. Keluaran dari fungsi ini adalah  $x$ , yaitu sebuah individu yang bernilai riil dalam interval  $[Ra, Rb]$ . Jika  $Nvar$  sama dengan 28 dan  $Nbit$  sama dengan 15, maka individu  $x$  terdiri dari 28 kolom, yaitu  $a_0, a_1, a_1$  sampai dengan  $a_{27}$  dihitung menggunakan dekode kromosom sehingga didapatkan 28 nilai riil (lampiran 3).

### 3.4.3 Perhitungan *Fitness*

Setelah dilakukan pengkodean kromosom, dihitung nilai *fitness* suatu individu menggunakan PNN dengan rumusan

$$fitness = \frac{1}{RMSE} \quad (3.5)$$

Berikut merupakan langkah-langkah dalam mencari *fitness* suatu individu menggunakan PNN. Sebagai contoh, dalam perhitungan menggunakan polinomial dengan orde dua.

- Langkah 1. Menentukan struktur koefisien dari nilai riil yang sudah didekodekan. Hasil pengkodean kromosom dapat dilihat pada lampiran 3.
- Langkah 2. Melakukan perulangan langkah 3 sampai langkah 5 sebanyak  $n$ -populasi.
- Langkah 3. Melakukan perulangan langkah 4 sampai langkah 5 untuk setiap pelatihan data.
- Langkah 4. Melakukan pelatihan data (lampiran 4).

Data 1

$$x_1 = 0,5778, \quad x_2 = 0,5944, \quad x_3 = 0,5778$$

$$y = 0,5944(\text{target})$$

maka didapat,

$$Z_{1out} = 0,001523904804$$

$$Z_{2out} = 0,0042509218417$$

$$Z_{3out} = 0,001958858707$$

$$Y_{out} = 0,0004529880222$$

Selanjutnya, data ke-2 sampai data ke- $n$  ( $n$  = banyak data) digunakan untuk melakukan pelatihan.

Langkah 5. Hitung galat yang ada pada masing-masing keluaran dengan persamaan:

$$MSE = \frac{(Y_{out\ data\ 1} - y)^2 + (Y_{out\ data\ 2} - y)^2 + \dots + (Y_{out\ data\ n} - y)^2}{n}$$

$$RMSE = \sqrt{MSE} \quad (3.6)$$

Setelah melakukan perhitungan RMSE, penghitungan nilai *fitness* dilakukan pada tiap populasi yang ada menggunakan persamaan (3.5).

### 3.4.4 Penggantian Populasi

Dalam AG dikenal skema penggantian populasi untuk mendapatkan nilai *fitness* maksimum, sehingga didapatkan sebuah populasi baru dengan mengulang langkah sebagai berikut:

Langkah 1. Mengurutkan *fitness*

Untuk menghindari kecenderungan konvergen pada optimum lokal, maka digunakan penskalaan nilai *fitness* seperti pada persamaan (3.7) berikut

$$Fitness(i) = maxF - (maxF - minF) \left( \frac{R(i) - 1}{n - 1} \right) \quad (3.7)$$

dengan

$n$  = jumlah kromosom dalam populasi

$maxF$  = nilai *fitness* tertinggi dalam populasi

$minF$  = nilai *fitness* terendah dalam populasi

Dengan menggunakan persamaan (3.7), diperoleh nilai *fitness* yang berada dalam interval [ $maxF$ ,  $minF$ ].

Langkah 2. *Roulette Wheel*

*Roulette wheel* merupakan pemilihan dua buah kromosom sebagai orang tua yang akan dipindah-silangkan. Proses pemilihan dilakukan sesuai dengan nilai *fitness*-nya dan dilakukan secara acak.

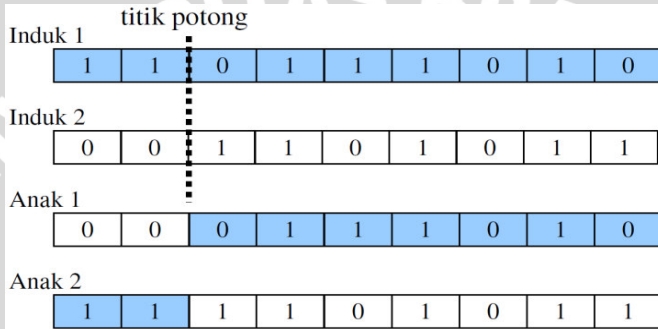
Langkah 3. Pindah silang

Titik potong pindah silang antara Induk 1 dan Induk 2 dilakukan secara *random*

$$Titik\ potong = 1 + fix(rand * (JumGen - 1)) \quad (3.8)$$

Dengan membangkitkan satu titik potong (titik potong bernilai antara 1 sampai  $JumGen-1$ ) menggunakan persamaan 3.8, maka didapatkan

- Anak 1 berisi potongan kromosom Induk 2 bagian depan dan potongan kromosom Induk 1 bagian belakang
- Anak 2 berisi potongan kromosom Induk 1 bagian depan dan potongan kromosom Induk 2 bagian belakang



**Gambar 3.4** Proses pindah silang

#### Langkah 4 Mutasi

Lakukan mutasi berdasarkan nilai peluang mutasi ( $P_m$ ). Gen-gen yang terpilih diubah nilai (0 menjadi 1 dan 1 menjadi 0).

#### Langkah 5 Elitisme

Karena seleksi dilakukan secara *random*, tidak ada jaminan bahwa suatu individu bernilai *fitness* tertinggi akan selalu terpilih, maka perlu dibuat satu atau beberapa kopinya. Jika ukuran populasi genap maka iterasi dimulai pada populasi ke-3 karena populasi (1) dan (2) merupakan individu terbaik. Jika ukuran populasi ganjil maka iterasi dimulai pada populasi ke-2 karena populasi (1) merupakan individu terbaik.

#### Langkah 6 Pergantian Populasi

Letakkan individu baru dalam populasi yang baru dan gunakan populasi baru untuk menjalankan algoritma lebih lanjut.

Penghitungan nilai *fitness* pada tiap populasi dilakukan sampai generasi yang diinginkan atau jika nilai *fitness* maksimum terpenuhi.

### 3.5 Proses Pemilihan Arsitektur Terbaik

Dalam menjalankan program GPNN dengan *binary encoding*, perlu dilakukan pemilihan arsitektur terbaik. Dalam AG, akan dilakukan kombinasi antara jumlah bit setiap variabel, batas atas, batas bawah, galat yang diinginkan (RMSE), ukuran populasi, peluang pindah silang, peluang mutasi, dan jumlah generasi untuk mengoptimasi bobot dalam PNN. Arsitektur hasil kombinasi yang menghasilkan nilai *fitness* yang besar pada proses pelatihan merupakan arsitektur terbaik yang akan digunakan untuk memprediksi harga penutupan saham harian. Dalam hal ini dilakukan beberapa percobaan untuk menghasilkan arsitektur terbaik. Tabel 3.3 menggunakan fungsi polinomial linier, Tabel 3.4 menggunakan fungsi polinomial orde dua, dan Tabel 3.5 menggunakan fungsi polinomial orde tiga. Dalam ketiga tabel tersebut menggunakan nilai  $[Rb, Ra] = [-10, 10]$ ,  $nbit = 4$ ,  $RMSE = 0,01$ , dan jumlah generasi=500, sedangkan ukuran populasi, peluang pindah silang, dan peluang mutasi dikombinasi agar menghasilkan nilai *fitness* terbaik.

- nbit* : Jumlah bit yang mengkodekan variabel
- Rb* : batas bawah interval
- Ra* : batas atas interval
- RMSE : galat minimum yang diharapkan
- Jumlah populasi : jumlah kromosom dalam suatu populasi
- Psilang : peluang pindah silang
- Pmutasi : peluang mutasi

**Tabel 3.3** Pemilihan arsitektur terbaik dengan GPNN linier

Perco- -baan	Jumlah populasi	Psilang	Pmutasi	<i>fitness</i>	Waktu Komputasi (detik)
1	10	0,95	0,01	16,38	152,93
2	250			85,23	2561,14
3	10	0,50	0,1	35,21	162,74
4	250			41,25	2937,45

Dalam Tabel 3.3 di atas, nilai *fitness* terbaik diperoleh saat menggunakan peluang pindah silang 0,95 dan peluang mutasi 0,01.



Ini menunjukkan peluang pindah silang besar (mendekati satu) dan peluang mutasi kecil maka diperoleh nilai *fitness* yang baik.

**Tabel 3.4** Pemilihan arsitektur terbaik dengan GPNN orde dua

Perco-baan	Jumlah populasi	Psilang	Pmutasi	<i>fitness</i>	Waktu Komputasi (detik)
1	10	0,95	0,01	62,40	228,85
2	250			100,03	2101,74
3	10	0,50	0,1	20,16	305,52
4	250			41,45	4538,11

Dalam Tabel 3.4, nilai *fitness* terbaik juga diperoleh saat menggunakan peluang pindah silang 0,95 dan peluang mutasi 0,01. Nilai *fitness* yang diperoleh dalam percobaan (2) adalah nilai maksimum yang diharapkan dalam penyelesaian persoalan.

**Tabel 3.5** Pemilihan arsitektur terbaik dengan GPNN orde tiga

Perco-baan	Jumlah populasi	Psilang	Pmutasi	<i>fitness</i>	Waktu Komputasi (detik)
1	10	0,95	0,01	65,60	592,98
2	250			90,68	6931,80
3	10	0,50	0,1	40,35	468,55
4	250			39,23	6793,71

Nilai *fitness* terbaik dalam Tabel 3.5 juga diperoleh saat menggunakan peluang pindah silang 0,95 dan peluang mutasi 0,01. Namun nilai *fitness* yang diperoleh tidak maksimum karena jumlah variabel dalam orde tiga lebih banyak dibandingkan variabel dalam linier dan orde dua, sehingga menyebabkan sulitnya pencarian koefisien yang menghasilkan *fitness* maksimum.

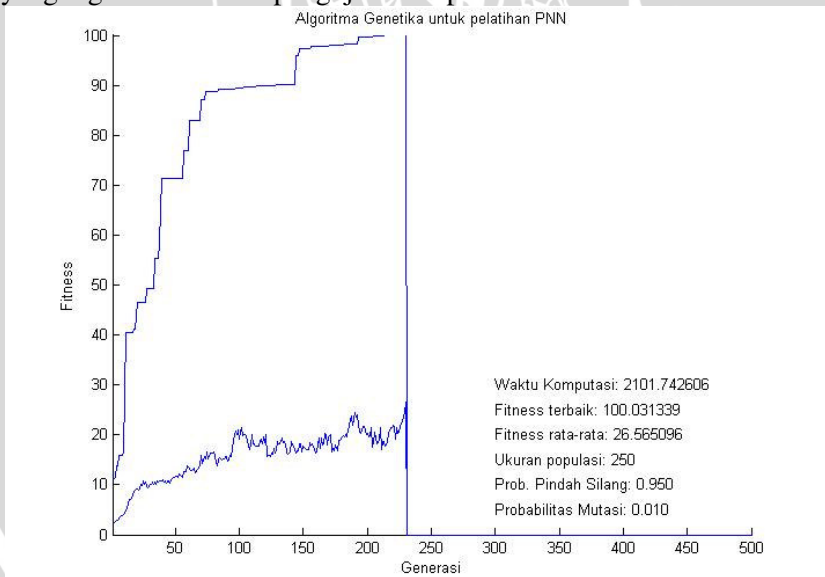
Dalam pencarian *fitness* dari ketiga tabel tersebut, jenis polinomial berpengaruh pada waktu komputasi. Semakin tinggi derajat polinomial maka semakin lama waktu komputasi. Hal ini disebabkan semakin tinggi polinomial maka semakin banyak pula koefisien PNN yang akan dioptimasi menggunakan AG yang dirumuskan pada persamaan (3.3). Jika persamaan linier, terdapat 13 variabel yang harus dioptimasi, orde 2 terdapat 28 variabel, dan orde tiga terdapat 50 variabel.

Jumlah populasi dalam suatu generasi juga berpengaruh terhadap lamanya komputasi. Semakin banyak populasi maka semakin lama juga waktu komputasi. Dalam suatu populasi yang sangat kecil, suatu kromosom dengan gen-gen yang mengarah ke solusi akan sangat cepat menyebar ke kromosom-kromosom lainnya, maka menggunakan nilai peluang pindah silang. Tabel 3.3, Tabel 3.4, dan Tabel 3.5 menggunakan dua jenis nilai peluang pindah silang yaitu 0,95 dan 0,50. Artinya, pindah silang dapat dilakukan hanya jika suatu bilangan acak [0,1) yang dibangkitkan kurang dari peluang pindah silang yang digunakan. Bilangan acak  $\neq 1$  karena membuat kromosom tidak mempunyai peluang dipindahsilangkan.

Pada Tabel 3.4 percobaan (2) yaitu saat digunakan GPNN orde dua, didapatkan nilai *fitness* maksimum, yaitu 100,03 atau

$$\text{galat} = \frac{1}{100,03} = 0,01$$

Jadi, struktur koefisien yang diperoleh dalam percobaan tersebut yang digunakan dalam pengujian dan peramalan.



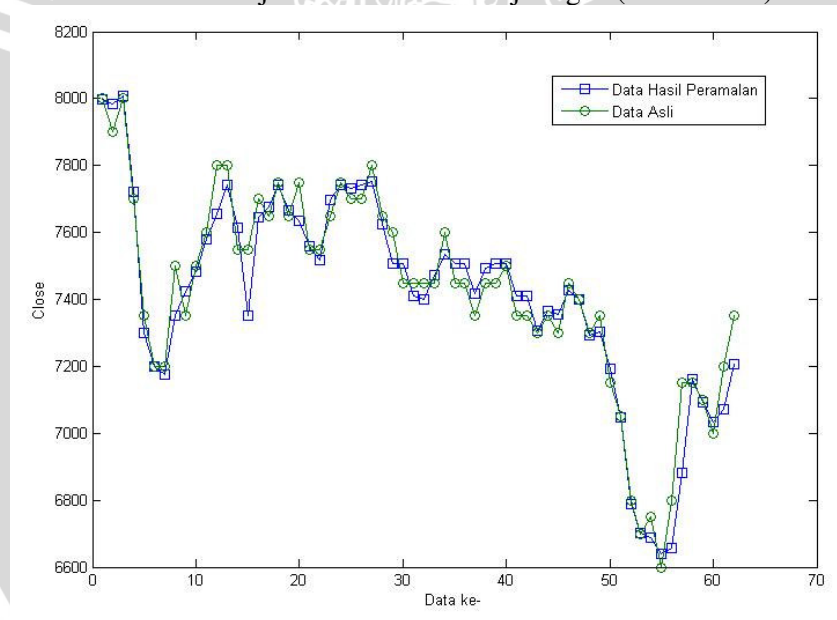
**Gambar 3.5** Proses pelatihan percobaan (2) Tabel 3.4

Pada Gambar 3.5, terlihat bahwa AG berhasil mencapai *fitness* maksimum pada generasi ke-230. Sehingga iterasi berhenti

sebelum mencapai generasi ke-500 karena *fitness* maksimum sudah terpenuhi. Selanjutnya akan dibandingkan hasil maksimum *fitness* pelatihan dengan hasil maksimum *fitness* pengujian. Data yang digunakan dalam pengujian adalah data harga saham harian PT Telkom pada bulan September 2010 sampai Desember 2010. Setelah data diuji maka didapatkan *nilai fitness* = 132,66 atau *RMSE* = 0,00754.

### 3.6 Peramalan dengan Menggunakan GPNN

Dari arsitektur terbaik yang diperoleh dari pelatihan dan setelah diuji, struktur koefisien-koefisien PNN yang didapat akan digunakan untuk memprediksi harga penutupan saham harian PT. Telekomunikasi Indonesia, Tbk. pada bulan Januari 2011 sampai Maret 2011. Dengan memasukkan data harga pembukaan, harga tertinggi, dan harga terendah, didapatkan prediksi harga penutupan. Grafik berikut menunjukkan data keluaran jaringan (Gambar 3.6).



**Gambar 3.6** Grafik keluaran jaringan dibandingkan dengan data asli. Selisih antara nilai keluaran jaringan dengan data yang sebenarnya (harga penutupan) pada Gambar 3.6 ditunjukkan dalam Tabel 3.6.

**Tabel 3.6** Selisih harga keluaran jaringan dengan data sebenarnya

<b>Data ke-</b>	<b>Tanggal</b>	<b>Harga Sebenarnya</b>	<b>Keluaran Jaringan</b>	<b>Selisih Mutlak</b>
<b>1</b>	03-01-11	8.000	7.997	3
<b>2</b>	04-01-11	7.900	7.985	85
<b>3</b>	05-01-11	8.000	8.009	9
<b>4</b>	06-01-11	7.700	7.722	22
<b>5</b>	07-01-11	7.350	7.301	49
<b>6</b>	10-01-11	7.200	7.198	2
<b>7</b>	11-01-11	7.200	7.175	25
<b>8</b>	12-01-11	7.500	7.350	150
<b>9</b>	13-01-11	7.350	7.423	73
<b>10</b>	14-01-11	7.500	7.482	18
<b>11</b>	17-01-11	7.600	7.580	20
<b>12</b>	18-01-11	7.800	7.656	144
<b>13</b>	19-01-11	7.800	7.742	58
<b>14</b>	20-01-11	7.550	7.614	64
<b>15</b>	21-01-11	7.550	7.350	200
<b>16</b>	24-01-11	7.700	7.644	56
<b>17</b>	25-01-11	7.650	7.678	28
<b>18</b>	26-01-11	7.750	7.741	9
<b>19</b>	27-01-11	7.650	7.667	17
<b>20</b>	28-01-11	7.750	7.634	116
<b>21</b>	31-01-11	7.550	7.560	10
<b>22</b>	01-02-11	7.550	7.516	34
<b>23</b>	02-02-11	7.650	7.698	48
<b>24</b>	04-02-11	7.750	7.741	9
<b>25</b>	07-02-11	7.700	7.731	31
<b>26</b>	08-02-11	7.700	7.741	41
<b>27</b>	09-02-11	7.800	7.753	47
<b>28</b>	10-02-11	7.650	7.624	26
<b>29</b>	11-02-11	7.600	7.507	93
<b>30</b>	14-02-11	7.450	7.507	57
<b>31</b>	16-02-11	7.450	7.410	40

**Lanjutan Tabel 3.6**

<b>Data ke-</b>	<b>Tanggal</b>	<b>Harga Sebenarnya</b>	<b>Keluaran Jaringan</b>	<b>Selisih Mutlak</b>
32	17-02-11	7.450	7.400	50
33	18-02-11	7.450	7.472	22
34	21-02-11	7.600	7.536	64
35	22-02-11	7.450	7.507	57
36	23-02-11	7.450	7.507	57
37	24-02-11	7.350	7.419	69
38	25-02-11	7.450	7.492	42
39	28-02-11	7.450	7.507	57
40	01-03-11	7.500	7.507	7
41	02-03-11	7.350	7.410	60
42	03-03-11	7.350	7.409	59
43	04-03-11	7.300	7.305	5
44	07-03-11	7.350	7.364	14
45	08-03-11	7.300	7.356	56
46	09-03-11	7.450	7.428	22
47	10-03-11	7.400	7.400	0
48	11-03-11	7.300	7.294	6
49	14-03-11	7.350	7.302	48
50	15-03-11	7.150	7.191	41
51	16-03-11	7.050	7.046	4
52	17-03-11	6.800	6.790	10
53	18-03-11	6.700	6.701	1
54	21-03-11	6.750	6.689	61
55	22-03-11	6.600	6.641	41
56	23-03-11	6.800	6.656	144
57	24-03-11	7.150	6.883	267
58	25-03-11	7.150	7.162	12
59	28-03-11	7.100	7.092	8
60	29-03-11	7.000	7.032	32
61	30-03-11	7.200	7.073	127
62	31-03-11	7.350	7.207	143



Setelah dijumlahkan, total selisih harga keluaran jaringan dengan data sebenarnya yaitu 3.170 rupiah, maka didapatkan rata-rata selisih nilai keluaran jaringan dengan data sebenarnya yaitu:

$$x = \frac{3.170}{62} = 51,13$$

Dari perhitungan, dapat dilihat bahwa rata-rata selisih nilai keluaran jaringan dengan data sebenarnya yaitu 51,13 rupiah, artinya jaringan telah mampu menghasilkan keluaran yang benar dengan bobot yang diperoleh karena telah memenuhi galat yang diinginkan pada proses pelatihan, yaitu 0,01. Dengan hasil galat 0,01 menunjukkan AG dapat diterapkan dengan baik pada PNN.



## BAB IV PENUTUP

### 4.1 Kesimpulan

Berdasarkan pembahasan skripsi ini, dapat diambil beberapa kesimpulan

1. Pada arsitektur GPNN, AG digunakan untuk menentukan koefisien-koefisien PNN agar diperoleh galat yang minimum. Dengan hasil galat 0,01 menunjukkan AG dapat diterapkan dengan baik pada PNN.
2. Dalam prediksi harga penutupan saham harian PT Telekomunikasi Indonesia, Tbk. bulan Januari sampai bulan Maret 2011 menggunakan GPNN, diperoleh rata-rata selisih nilai keluaran jaringan dengan data sebenarnya yaitu 51,13 rupiah, artinya jaringan telah mampu menghasilkan keluaran yang benar dengan bobot yang diperoleh pada proses pelatihan.

### 4.2 Saran

Pada penelitian selanjutnya ada beberapa hal yang dapat dikembangkan dari skripsi ini, yaitu meminimumkan waktu komputasi dalam pelatihan agar lebih efisien, dengan cara mengurangi jumlah generasi pada AG.

## DAFTAR PUSTAKA

- Anonymous, 2011. <http://www.telkom.co.id>. (Akses 10 April 2011).
- Anonymous, 2011. <http://economy.okezone.com>. (Akses 12 Agustus 2011)
- Ai, T.J. 1999. *Optimasi Peramalan Pemulusan Eksponensial Satu Parameter dengan Menggunakan Algoritma Non-Linear Programming*. Jurnal Teknologi Industri Vol. 3 (hal. 139-148)
- Ardiani, A. 2007. *Analisis Pengaruh Kinerja Keuangan terhadap Perubahan Harga Saham pada Perusahaan Perbankan di Bursa Efek Jakarta (BEJ)*. Fakultas Ekonomi, Universitas Negeri Semarang. Semarang.
- Farzi, S. 2008. *A New Approach to Polynomial Neural Network Based on Genetic Algorithm*. Department of Computer Engineering, Islamic Azad University. Iran. (hal. 480-486).
- Hidayatno, A. *Pengantar Jaringan Saraf*. Semarang: UNDIP (Akses 5 September 2010).
- Kusumadewi, S. 2004. *Membangun Jaringan Syaraf Tiruan Menggunakan MATLAB & Excel Link*. Yogyakarta : Graha Ilmu.
- Pandjaitan, L.W. 2007. *Dasar-dasar Komputasi Cerdas*. Yogyakarta: Andi.
- Setiawan, K. 2003. *Paradigma Sistem Cerdas*. Malang: Bayumedia Publishing.
- Suyanto. 2005. *Algoritma Genetika dalam MATLAB*. Yogyakarta: Andi.

## Lampiran 1

Data harga saham harian PT. Telkom yang diperdagangkan di Bursa Efek Indonesia pada bulan Januari 2007 sampai Maret 2011.

Data ke-	Tanggal	Input			Output
		Open	High	Low	Close
1	02-01-07	10.200	10.350	10.200	10.350
2	03-01-07	10.350	10.400	10.250	10.350
3	04-01-07	10.400	10.450	10.150	10.150
4	05-01-07	10.150	10.150	10.050	10.150
5	08-01-07	10.050	10.100	10.000	10.000
6	09-01-07	10.000	10.150	9.950	9.950
7	10-01-07	9.850	9.900	9.600	9.700
8	11-01-07	9.750	9.800	9.550	9.650
9	12-01-07	9.750	9.800	9.300	9.600
10	15-01-07	9.750	9.900	9.700	9.850
11	16-01-07	9.900	9.950	9.850	9.900
12	17-01-07	9.900	10.000	9.900	9.900
13	18-01-07	9.900	10.000	9.900	9.900
14	19-01-07	9.900	10.150	9.900	10.150
15	22-01-07	10.150	10.250	10.050	10.150
16	23-01-07	10.150	10.150	9.950	10.000
17	24-01-07	10.000	10.050	9.850	9.950
18	25-01-07	9.900	9.900	9.550	9.750
19	26-01-07	9.600	9.600	9.400	9.550
20	29-01-07	9.600	9.700	9.550	9.650
21	30-01-07	9.650	9.700	9.450	9.550
22	31-01-07	9.600	9.600	9.400	9.450
23	01-02-07	9.550	9.600	9.450	9.550
24	02-02-07	9.600	9.700	9.450	9.550
25	05-02-07	9.550	9.550	9.450	9.550
26	06-02-07	9.450	9.500	9.400	9.450
27	07-02-07	9.450	9.450	9.300	9.350
28	08-02-07	9.300	9.300	9.050	9.300
29	09-02-07	9.200	9.200	9.050	9.100
30	12-02-07	9.000	9.150	8.900	9.050

## Lanjutan Lampiran 1

Data ke-	Tanggal	Input			Output
		Open	High	Low	Close
31	13-02-07	9.050	9.100	8.900	8.950
32	14-02-07	9.150	9.350	9.150	9.250
33	15-02-07	9.500	9.600	9.450	9.500
34	16-02-07	9.600	9.750	9.550	9.650
35	19-02-07	9.650	9.700	9.600	9.650
36	20-02-07	9.650	9.650	9.550	9.650
37	22-02-07	9.650	9.750	9.500	9.600
38	23-02-07	9.600	9.600	9.300	9.300
39	26-02-07	9.300	9.300	9.100	9.200
40	27-02-07	9.200	9.300	9.100	9.100
41	28-02-07	8.600	9.000	8.550	8.900
42	01-03-07	9.150	9.350	9.150	9.250
43	02-03-07	9.150	9.350	9.150	9.350
44	05-03-07	9.100	9.150	8.950	9.000
45	06-03-07	9.000	9.300	8.950	9.250
46	07-03-07	9.300	9.400	9.200	9.250
47	08-03-07	9.200	9.300	9.200	9.300
48	09-03-07	9.350	9.350	9.150	9.200
49	12-03-07	9.250	9.450	9.250	9.400
50	13-03-07	9.400	9.450	9.350	9.400
51	14-03-07	9.200	9.350	9.150	9.200
52	15-03-07	9.300	9.350	9.200	9.350
53	16-03-07	9.350	9.400	9.250	9.300
54	22-03-07	9.400	9.550	9.400	9.550
55	23-03-07	9.600	9.650	9.500	9.600
56	26-03-07	9.600	9.700	9.550	9.700
57	27-03-07	9.750	9.850	9.700	9.800
58	28-03-07	9.700	9.700	9.650	9.700
59	29-03-07	9.600	9.750	9.550	9.650
60	30-03-07	9.750	9.900	9.750	9.850
61	02-04-07	9.850	10.000	9.850	9.900
62	03-04-07	10.000	10.100	9.950	10.000



## Lanjutan Lampiran 1

Data ke-	Tanggal	Input			Output
		Open	High	Low	Close
63	04-04-07	10.150	10.250	10.100	10.200
64	05-04-07	10.250	10.250	10.050	10.100
65	09-04-07	10.150	10.250	10.100	10.200
66	10-04-07	10.250	10.350	10.200	10.300
67	11-04-07	10.350	10.500	10.350	10.450
68	12-04-07	10.300	10.300	10.200	10.250
69	13-04-07	10.350	10.400	10.300	10.400
70	16-04-07	10.450	10.450	10.300	10.400
71	17-04-07	10.450	10.450	10.300	10.450
72	18-04-07	10.450	10.450	10.200	10.250
73	19-04-07	10.200	10.200	10.000	10.000
74	20-04-07	10.100	10.400	10.100	10.400
75	23-04-07	10.500	10.600	10.450	10.550
76	24-04-07	10.450	10.550	10.400	10.500
77	26-04-07	10.600	10.850	10.600	10.800
78	27-04-07	10.700	10.800	10.650	10.750
79	30-04-07	10.700	10.700	10.450	10.500
80	01-05-07	10.400	10.450	10.300	10.350
81	02-05-07	10.400	10.500	10.100	10.250
82	03-05-07	10.350	10.350	10.250	10.350
83	04-05-07	10.350	10.400	10.200	10.300
84	07-05-07	10.300	10.300	10.000	10.200
85	08-05-07	10.200	10.200	9.900	9.950
86	09-05-07	9.900	9.950	9.750	9.900
87	10-05-07	9.950	9.950	9.850	9.900
88	11-05-07	9.700	9.750	9.550	9.600
89	14-05-07	9.800	9.800	9.600	9.750
90	15-05-07	9.750	9.750	9.550	9.600
91	16-05-07	9.550	9.600	9.500	9.600
92	21-05-07	9.700	9.700	9.400	9.500
93	22-05-07	9.500	9.500	9.300	9.400
94	23-05-07	9.450	9.750	9.450	9.700

## Lanjutan Lampiran 1

Data ke-	Tanggal	Input			Output
		Open	High	Low	Close
95	24-05-07	9.700	9.800	9.600	9.700
96	25-05-07	9.550	9.600	9.500	9.600
97	28-05-07	9.650	9.700	9.550	9.650
98	29-05-07	9.650	9.650	9.450	9.500
99	30-05-07	9.400	9.550	9.350	9.550
100	31-05-07	9.650	9.650	9.450	9.550
101	04-06-07	9.750	9.800	9.600	9.750
102	05-06-07	9.750	9.750	9.600	9.700
103	06-06-07	9.650	9.700	9.600	9.650
104	07-06-07	9.600	9.800	9.550	9.700
105	08-06-07	9.600	9.650	9.550	9.650
106	11-06-07	9.700	9.700	9.600	9.700
107	12-06-07	9.700	9.900	9.700	9.850
108	13-06-07	9.700	9.750	9.600	9.700
109	14-06-07	9.850	9.850	9.700	9.800
110	15-06-07	9.800	9.800	9.700	9.750
111	18-06-07	9.800	9.850	9.650	9.700
112	19-06-07	9.600	9.750	9.600	9.700
113	20-06-07	9.650	9.850	9.600	9.800
114	21-06-07	9.750	9.750	9.650	9.750
115	22-06-07	9.700	9.750	9.600	9.700
116	25-06-07	9.650	9.750	9.650	9.650
117	26-06-07	9.600	9.800	9.600	9.750
118	27-06-07	9.750	9.800	9.650	9.750
119	28-06-07	9.750	9.750	9.600	9.700
120	29-06-07	9.750	9.900	9.600	9.850
121	02-07-07	9.850	9.950	9.800	9.900
122	03-07-07	9.900	10.100	9.900	10.000
123	04-07-07	10.050	10.400	10.000	10.350
124	05-07-07	10.400	10.600	10.400	10.550
125	06-07-07	10.550	10.650	10.350	10.400
126	09-07-07	10.400	10.550	10.400	10.500

## Lanjutan Lampiran 1

Data ke-	Tanggal	Input			Output
		Open	High	Low	Close
127	10-07-07	10.500	10.650	10.450	10.500
128	11-07-07	10.450	10.650	10.400	10.600
129	12-07-07	10.650	11.100	10.650	10.900
130	13-07-07	11.100	11.450	11.100	11.450
131	17-07-07	11.400	11.400	11.250	11.300
132	18-07-07	11.200	11.200	10.900	11.000
133	19-07-07	11.000	11.300	10.950	11.300
134	20-07-07	11.400	11.500	11.150	11.350
135	23-07-07	11.150	11.150	11.000	11.050
136	24-07-07	11.100	11.150	10.950	11.000
137	25-07-07	11.000	11.000	10.800	10.950
138	26-07-07	10.950	11.100	10.650	10.750
139	27-07-07	10.450	10.650	10.400	10.450
140	31-07-07	10.900	11.200	10.900	11.200
141	01-08-07	11.000	11.000	10.600	10.700
142	02-08-07	10.800	10.950	10.650	10.850
143	03-08-07	10.950	11.000	10.900	10.900
144	06-08-07	10.650	10.650	10.450	10.600
145	07-08-07	10.600	10.700	10.550	10.650
146	08-08-07	10.700	11.200	10.700	11.050
147	09-08-07	11.250	11.250	10.850	10.900
148	10-08-07	10.500	10.900	10.400	10.900
149	14-08-07	10.750	10.850	10.700	10.850
150	15-08-07	10.600	10.600	10.050	10.150
151	16-08-07	9.900	9.900	9.550	9.850
152	20-08-07	10.150	10.700	10.100	10.500
153	21-08-07	10.400	10.450	10.100	10.150
154	22-08-07	10.200	10.500	10.200	10.500
155	23-08-07	10.700	10.750	10.500	10.500
156	24-08-07	10.500	10.900	10.400	10.850
157	27-08-07	11.000	11.150	10.950	10.950
158	28-08-07	10.950	10.950	10.750	10.800

## Lanjutan Lampiran 1

Data ke-	Tanggal	Input			Output
		Open	High	Low	Close
159	29-08-07	10.500	10.650	10.450	10.600
160	30-08-07	10.750	10.850	10.650	10.700
161	31-08-07	10.700	10.850	10.550	10.850
162	03-09-07	10.900	11.200	10.900	11.150
163	04-09-07	11.150	11.300	11.150	11.200
164	05-09-07	11.250	11.300	10.950	11.000
165	06-09-07	10.950	11.050	10.850	10.950
166	07-09-07	11.100	11.250	11.050	11.150
167	10-09-07	10.900	11.000	10.850	10.900
168	11-09-07	10.900	10.950	10.800	10.800
169	12-09-07	10.850	10.900	10.600	10.850
170	13-09-07	10.850	10.850	10.700	10.800
171	14-09-07	10.800	10.900	10.650	10.650
172	17-09-07	10.750	10.750	10.600	10.700
173	18-09-07	10.700	10.700	10.600	10.700
174	19-09-07	11.100	11.300	11.000	11.250
175	20-09-07	11.300	11.350	11.000	11.000
176	21-09-07	10.950	11.150	10.900	11.150
177	24-09-07	11.100	11.100	10.900	10.950
178	25-09-07	10.900	10.900	10.750	10.800
179	26-09-07	10.800	10.950	10.750	10.850
180	27-09-07	10.950	11.000	10.850	10.900
181	28-09-07	10.950	11.050	10.900	11.000
182	01-10-07	11.050	11.350	11.050	11.350
183	02-10-07	11.500	12.150	11.500	12.000
184	03-10-07	12.100	12.400	11.850	11.950
185	04-10-07	11.800	12.100	11.750	12.100
186	05-10-07	12.150	12.550	12.050	12.450
187	08-10-07	12.550	12.650	12.400	12.500
188	09-10-07	12.300	12.400	12.150	12.300
189	10-10-07	12.400	12.550	12.300	12.350
190	11-10-07	12.400	12.750	12.400	12.650

## Lanjutan Lampiran 1

Data ke-	Tanggal	Input			Output
		Open	High	Low	Close
191	17-10-07	12.400	12.400	12.050	12.150
192	18-10-07	12.200	12.400	12.150	12.150
193	19-10-07	12.000	12.000	11.600	11.800
194	22-10-07	11.500	11.550	11.200	11.300
195	23-10-07	11.500	12.000	11.450	11.900
196	24-10-07	11.900	12.050	11.550	11.550
197	25-10-07	11.600	11.800	11.500	11.750
198	26-10-07	11.750	12.000	11.750	11.900
199	29-10-07	12.000	12.050	11.850	11.850
200	30-10-07	11.850	11.950	11.350	11.350
201	31-10-07	11.200	11.200	10.550	10.750
202	01-11-07	11.000	11.250	10.950	11.150
203	02-11-07	10.850	11.400	10.800	11.400
204	05-11-07	11.300	11.450	11.050	11.200
205	06-11-07	11.300	11.450	11.300	11.350
206	07-11-07	11.500	11.600	11.400	11.600
207	08-11-07	11.250	11.450	11.250	11.400
208	09-11-07	11.300	11.350	11.200	11.300
209	12-11-07	11.000	11.100	10.900	11.000
210	13-11-07	10.800	10.900	10.700	10.850
211	14-11-07	10.950	11.000	10.600	10.700
212	15-11-07	10.600	10.750	10.550	10.650
213	16-11-07	10.500	10.600	10.400	10.500
214	19-11-07	10.500	10.700	10.400	10.450
215	20-11-07	10.100	10.250	9.850	10.100
216	21-11-07	10.100	10.250	10.000	10.100
217	22-11-07	9.950	10.400	9.950	10.200
218	23-11-07	10.300	10.300	10.050	10.050
219	26-11-07	10.100	10.300	10.050	10.200
220	27-11-07	10.000	10.100	9.950	10.000
221	28-11-07	10.100	10.200	10.050	10.100
222	29-11-07	10.400	10.600	10.400	10.500



## Lanjutan Lampiran 1

Data ke-	Tanggal	Input			Output
		Open	High	Low	Close
223	30-11-07	10.400	10.400	10.100	10.150
224	04-12-07	10.450	10.550	10.350	10.500
225	05-12-07	10.500	10.900	10.450	10.900
226	06-12-07	11.050	11.250	10.950	11.100
227	07-12-07	11.250	11.250	10.900	10.950
228	10-12-07	11.000	11.050	10.900	11.000
229	11-12-07	11.100	11.150	10.900	11.100
230	12-12-07	10.900	11.000	10.850	10.950
231	13-12-07	11.000	11.000	10.550	10.600
232	14-12-07	10.650	10.650	10.300	10.400
233	17-12-07	10.400	10.400	9.900	10.050
234	18-12-07	9.850	10.250	9.800	10.050
235	19-12-07	10.150	10.200	10.000	10.000
236	26-12-07	10.200	10.300	10.000	10.050
237	27-12-07	10.100	10.100	9.900	10.000
238	28-12-07	9.950	10.150	9.900	10.150
239	02-01-08	10.050	10.100	9.950	10.000
240	03-01-08	9.900	10.000	9.850	9.900
241	04-01-08	9.900	10.100	9.800	10.000
242	07-01-08	9.900	10.250	9.850	10.200
243	08-01-08	10.200	10.300	10.150	10.250
244	09-01-08	10.050	10.200	10.050	10.050
245	14-01-08	10.000	10.050	9.700	9.850
246	15-01-08	9.850	9.850	9.250	9.300
247	16-01-08	9.000	9.000	8.750	8.950
248	17-01-08	9.050	9.300	8.950	9.200
249	18-01-08	8.900	9.400	8.800	9.200
250	21-01-08	9.100	9.150	8.750	8.900
251	22-01-08	8.500	8.600	8.250	8.400
252	23-01-08	8.800	9.050	8.600	8.850
253	24-01-08	8.800	9.250	8.800	9.050
254	25-01-08	9.200	9.450	9.150	9.350

## Lanjutan Lampiran 1

Data ke-	Tanggal	Input			Output
		Open	High	Low	Close
255	28-01-08	9.150	9.200	9.050	9.100
256	29-01-08	9.200	9.200	8.950	8.950
257	30-01-08	9.000	9.050	8.850	8.900
258	31-01-08	8.850	9.250	8.750	9.250
259	01-02-08	9.300	9.400	9.100	9.300
260	04-02-08	9.450	9.800	9.400	9.700
261	05-02-08	9.600	10.100	9.550	10.000
262	06-02-08	9.550	9.900	9.500	9.750
263	11-02-08	9.550	9.800	9.500	9.550
264	12-02-08	9.700	9.850	9.600	9.750
265	13-02-08	9.950	10.100	9.850	9.900
266	14-02-08	10.000	10.100	9.850	9.950
267	15-02-08	9.700	10.000	9.700	10.000
268	18-02-08	10.000	10.050	9.950	9.950
269	19-02-08	10.000	10.200	10.000	10.100
270	20-02-08	10.000	10.100	9.950	10.050
271	21-02-08	10.200	10.250	10.100	10.150
272	22-02-08	10.050	10.200	10.000	10.100
273	25-02-08	10.200	10.250	10.000	10.000
274	26-02-08	10.100	10.100	9.900	9.950
275	27-02-08	10.150	10.250	10.050	10.100
276	28-02-08	10.200	10.250	10.150	10.250
277	29-02-08	10.200	10.200	9.750	9.800
278	03-03-08	9.600	9.650	9.350	9.450
279	04-03-08	9.600	9.650	9.400	9.450
280	05-03-08	9.450	9.500	9.400	9.400
281	06-03-08	9.600	9.900	9.550	9.700
282	10-03-08	9.450	9.450	9.250	9.250
283	11-03-08	9.200	9.250	8.850	9.250
284	12-03-08	9.600	9.700	9.350	9.500
285	13-03-08	9.400	9.450	9.200	9.250
286	14-03-08	9.300	9.400	9.200	9.200

## Lanjutan Lampiran 1

Data ke-	Tanggal	Input			Output
		Open	High	Low	Close
287	18-03-08	9.350	9.550	9.200	9.400
288	19-03-08	9.550	9.700	9.500	9.600
289	25-03-08	9.900	10.000	9.750	9.800
290	26-03-08	9.800	9.850	9.600	9.800
291	27-03-08	9.700	9.750	9.550	9.600
292	28-03-08	9.650	9.800	9.600	9.750
293	31-03-08	9.650	9.750	9.600	9.650
294	01-04-08	9.650	9.700	9.400	9.700
295	02-04-08	9.850	9.850	9.550	9.700
296	03-04-08	9.650	9.650	9.300	9.400
297	04-04-08	9.600	9.600	9.200	9.400
298	07-04-08	9.450	9.450	9.200	9.450
299	08-04-08	9.300	9.350	9.200	9.250
300	09-04-08	9.250	9.300	8.950	9.000
301	10-04-08	8.900	9.100	8.850	9.050
302	11-04-08	9.150	9.200	9.050	9.150
303	14-04-08	9.000	9.200	9.000	9.000
304	15-04-08	9.000	9.100	9.000	9.100
305	16-04-08	9.100	9.150	9.050	9.100
306	17-04-08	9.200	9.200	9.100	9.200
307	18-04-08	9.100	9.250	9.050	9.100
308	21-04-08	9.200	9.200	9.000	9.000
309	22-04-08	9.000	9.050	8.750	8.850
310	23-04-08	8.850	8.850	8.600	8.750
311	24-04-08	8.800	8.850	8.650	8.650
312	25-04-08	8.700	8.750	8.600	8.650
313	28-04-08	8.700	8.900	8700	8.900
314	29-04-08	8.950	9.000	8.800	8.950
315	30-04-08	9.000	9.000	8.850	8.850
316	02-05-08	9.050	9.100	8.950	8.950
317	05-05-08	9.100	9.100	8.900	9.000
318	06-05-08	8.950	9.050	8.900	8.900

## Lanjutan Lampiran 1

Data ke-	Tanggal	Input			Output
		Open	High	Low	Close
319	07-05-08	8.950	8.950	8.650	8.750
320	08-05-08	8.600	8.700	8.400	8.500
321	09-05-08	8.500	8.650	8.500	8.600
322	12-05-08	8.600	8.700	8.600	8.650
323	13-05-08	8.700	8.700	8.600	8.650
324	14-05-08	8.650	8.800	8.600	8.700
325	15-05-08	8.750	8.750	8.500	8.500
326	16-05-08	8.550	8.600	8.500	8.500
327	19-05-08	8.550	8.650	8.500	8.600
328	21-05-08	8.500	8.550	8.500	8.550
329	22-05-08	8.450	8.700	8.450	8.650
330	23-05-08	8.650	8.700	8.450	8.600
331	26-05-08	8.350	8.450	8.250	8.300
332	27-05-08	8.350	8.350	8.000	8.050
333	28-05-08	8.100	8.100	7.800	7.900
334	29-05-08	8.000	8.000	7.900	7.950
335	30-05-08	8.000	8.200	7.950	8.100
336	02-06-08	8.100	8.100	7.950	7.950
337	03-06-08	7.850	7.950	7.750	7.950
338	04-06-08	7.900	7.950	7.850	7.850
339	05-06-08	7.950	8.150	7.900	8.150
340	06-06-08	8.250	8.300	7.900	7.950
341	10-06-08	7.950	8.050	7.900	7.950
342	11-06-08	7.900	7.900	7.700	7.750
343	12-06-08	7.600	7.650	7.500	7.550
344	13-06-08	7.750	7.600	7.450	7.550
345	16-06-08	7.650	7.750	7.450	7.500
346	17-06-08	7.500	7.500	7.250	7.350
347	18-06-08	7.450	7.750	7.450	7.500
348	19-06-08	7.450	7.500	7.400	7.550
349	20-06-08	7.600	7.750	7.550	7.750
350	23-06-08	7.600	7.800	7.550	7.750

## Lanjutan Lampiran 1

Data ke-	Tanggal	Input			Output
		Open	High	Low	Close
351	24-06-08	7.750	7.800	7.600	7.700
352	25-06-08	7.600	7.650	7.400	7.500
353	26-06-08	7.500	7.550	7.300	7.400
354	27-06-08	7.300	7.450	7.200	7.300
355	30-06-08	7.300	7.350	7.250	7.300
356	01-07-08	7.400	7.550	7.350	7.550
357	02-07-08	7.550	7.650	7.500	7.650
358	03-07-08	7.450	7.500	7.300	7.350
359	04-07-08	7.400	7.450	7.300	7.350
360	07-07-08	7.350	7.500	7.350	7.400
361	08-07-08	7.400	7.500	7.250	7.300
362	09-07-08	7.350	7.450	7.300	7.400
363	10-07-08	7.300	7.500	7.300	7.500
364	11-07-08	7.550	7.700	7.550	7.700
365	14-07-08	7.500	7.500	7.250	7.350
366	15-07-08	7.200	7.250	6.900	6.900
367	16-07-08	7.050	7.100	6.950	6.950
368	05-08-08	7.700	7.850	7.700	7.850
369	06-08-08	7.900	7.900	7.700	7.850
370	07-08-08	7.750	8.050	7.700	7.850
371	08-08-08	7.750	7.850	7.700	7.750
372	11-08-08	7.800	7.900	7.700	7.750
373	12-08-08	7.650	7.700	7.400	7.500
374	13-08-08	7.300	7.450	7.300	7.400
375	14-08-08	7.250	7.550	7.200	7.450
376	15-08-08	7.300	7.600	7.300	7.450
377	22-08-08	7.450	7.850	7.450	7.850
378	26-08-08	7.750	7.850	7.700	7.750
379	29-08-08	7.950	8.000	7.850	8.000
380	01-09-08	7.950	8.000	7.750	7.850
381	02-09-08	7.850	7.950	7.800	7.900
382	05-09-08	7.400	7.650	7.350	7.600



## Lanjutan Lampiran 1

Data ke-	Tanggal	Input			Output
		Open	High	Low	Close
383	09-09-08	7.600	7.600	7.450	7.500
384	12-09-08	7.200	7.200	6.800	6.850
385	19-09-08	6.850	7.050	6.750	6.900
386	26-09-08	6.950	7.100	6.900	7.100
387	29-09-08	7.100	7.300	7.100	7.150
388	06-10-08	6.900	7.350	6.700	7.100
389	07-10-08	7.000	7.400	6.950	7.150
390	08-10-08	7.050	7.100	6.400	6.450
391	14-10-08	6.950	7.250	6.950	7.250
392	17-10-08	6.700	6.950	6.100	6.200
393	20-10-08	6.200	6.800	6.200	6.800
394	21-10-08	7.300	7.400	6.950	7.100
395	24-10-08	6.300	6.350	5.900	5.900
396	27-10-08	5.650	5.650	5.350	5.350
397	28-10-08	4.900	5.300	4.850	5.000
398	29-10-08	5.500	5.500	4.900	5.100
399	31-10-08	5.450	5.700	5.350	5.400
400	03-11-08	5.500	5.650	5.250	5.300
401	04-11-08	5.350	5.450	5.150	5.400
402	05-11-08	5.650	6.000	5.600	5.650
403	06-11-08	5.400	5.750	5.200	5.700
404	07-11-08	5.500	6.100	5.500	6.100
405	10-11-08	6.200	6.450	6.000	6.150
406	11-11-08	6.000	6.250	5.950	6.100
407	14-11-08	5.800	6.000	5.750	5.800
408	17-11-08	5.950	6.150	5.800	5.900
409	18-11-08	5.850	5.850	5.400	5.550
410	19-11-08	5.600	5.700	5.450	5.600
411	21-11-08	5.550	5.700	5.500	5.500
412	24-11-08	5.600	5.700	5.500	5.600
413	25-11-08	5.850	5.850	5.400	5.450
414	28-11-08	5.600	5.950	5.600	5.850

## Lanjutan Lampiran 1

Data ke-	Tanggal	Input			Output
		Open	High	Low	Close
415	01-12-08	5.700	6.200	5.700	6.000
416	02-12-08	5.700	5.950	5.700	5.750
417	03-12-08	5.850	5.950	5.800	5.800
418	05-12-08	5.500	6.000	5.500	6.000
419	09-12-08	6.300	6.600	6.250	6.600
420	10-12-08	6.600	6.900	6.500	6.900
421	12-12-08	6.450	6.750	6.400	6.600
422	15-12-08	6.750	7.250	6.650	7.250
423	16-12-08	7.200	7.400	6.900	6.950
424	17-12-08	7.100	7.250	7.050	7.150
425	19-12-08	6.700	6.900	6.650	6.800
426	22-12-08	6.800	7.050	6.800	7.050
427	26-12-08	6.850	6.950	6.750	6.800
428	30-12-08	6.850	7.000	6.850	6.900
429	01-06-09	7.600	7.600	7.500	7.600
430	02-06-09	7.700	7.750	7.600	7.600
431	03-06-09	7.700	7.800	7.600	7.800
432	04-06-09	7.650	7.800	7.650	7.800
433	05-06-09	7.850	7.850	7.700	7.750
434	08-06-09	7.750	8.000	7.650	7.700
435	09-06-09	7.850	7.850	7.600	7.800
436	10-06-09	7.800	7.900	7.750	7.750
437	11-06-09	7.750	7.800	7.500	7.550
438	12-06-09	7.650	7.750	7.550	7.650
439	15-06-09	7.700	7.800	7.650	7.750
440	16-06-09	7.700	7.700	7.500	7.550
441	17-06-09	7.600	7.650	7.400	7.450
442	19-06-09	7.450	7.550	7.400	7.450
443	22-06-09	7.500	7.650	7.450	7.500
444	23-06-09	7.450	7.450	7.300	7.350
445	24-06-09	7.400	7.400	7.300	7.400
446	25-06-09	7.450	7.700	7.450	7.600

## Lanjutan Lampiran 1

Data ke-	Tanggal	Input			Output
		Open	High	Low	Close
447	26-06-09	7.700	7.700	7.550	7.600
448	29-06-09	7.650	7.650	7.350	7.400
449	30-06-09	7.550	7.600	7.500	7.500
450	01-07-09	7.550	7.900	7.550	7.900
451	02-07-09	8.000	8.200	8.000	8.200
452	03-07-09	8.150	8.700	8.050	8.600
453	07-07-09	8.450	8.600	8.300	8.550
454	09-07-09	8.600	8.650	8.050	8.100
455	10-07-09	8.200	8.200	7.900	7.950
456	13-07-09	7.900	8.000	7.800	7.900
457	14-07-09	8.100	8.150	7.950	8.000
458	15-07-09	8.100	8.100	7.900	8.050
459	16-07-09	8.150	8.250	7.800	7.850
460	17-07-09	7.850	8.000	7.750	8.000
461	21-07-09	8.150	8.400	8.100	8.200
462	22-07-09	8.200	8.250	7.950	8.100
463	23-07-09	8.200	8.300	8.150	8.250
464	24-07-09	8.350	8.600	8.350	8.550
465	27-07-09	8.600	8.950	8.600	8.800
466	28-07-09	8.750	9.000	8.750	8.950
467	29-07-09	8.850	8.900	8.650	8.850
468	30-07-09	8.850	9.450	8.800	9.400
469	31-07-09	9.300	9.400	8.900	8.950
470	03-08-09	8.950	9.100	8.850	8.900
471	04-08-09	9.000	9.050	8.800	8.800
472	05-08-09	8.800	8.800	8.500	8.600
473	06-08-09	8.650	8.750	8.550	8.750
474	07-08-09	8.650	8.850	8.650	8.800
475	10-08-09	8.800	9.100	8.800	9.000
476	11-08-09	9.050	9.050	8.850	8.900
477	12-08-09	8.750	8.750	8.500	8.550
478	13-08-09	8.650	8.800	8.650	8.750

## Lanjutan Lampiran 1

Data ke-	Tanggal	Input			Output
		Open	High	Low	Close
479	14-08-09	8.750	8.800	8.550	8.700
480	18-08-09	8.500	8.650	8.450	8.500
481	19-08-09	8.550	8.550	8.300	8.350
482	20-08-09	8.500	8.550	8.400	8.450
483	21-08-09	8.400	8.500	8.400	8.450
484	24-08-09	8.500	8.550	8.350	8.450
485	25-08-09	8.450	8.650	8.350	8.650
486	26-08-09	8.600	8.650	8.550	8.550
487	27-08-09	8.500	8.700	8.500	8.600
488	28-08-09	8.600	8.750	8.600	8.650
489	31-08-09	8.650	8.650	8.350	8.400
490	01-09-09	8.400	8.550	8.350	8.350
491	02-09-09	8.300	8.400	8.250	8.250
492	03-09-09	8.350	8.550	8.300	8.400
493	04-09-09	8.400	8.550	8.400	8.400
494	07-09-09	8.400	8.550	8.400	8.550
495	08-09-09	8.500	8.550	8.450	8.450
496	09-09-09	8.400	8.400	8.300	8.350
497	10-09-09	8.450	8.500	8.350	8.450
498	11-09-09	8.500	8.550	8.400	8.450
499	14-09-09	8.400	8.400	8.300	8.300
500	15-09-09	8.350	8.400	8.300	8.400
501	16-09-09	8.400	8.400	8.300	8.350
502	17-09-09	8.400	8.450	8.300	8.400
503	24-09-09	8.500	8.800	8.500	8.750
504	25-09-09	8.650	8.700	8.550	8.650
505	28-09-09	8.600	8.600	8.450	8.500
506	29-09-09	8.550	8.600	8.500	8.550
507	30-09-09	8.550	8.650	8.550	8.650
508	01-10-09	8.600	8.750	8.600	8.700
509	02-10-09	8.550	8.650	8.450	8.550
510	05-10-09	8.400	8.550	8.400	8.500

## Lanjutan Lampiran 1

Data ke-	Tanggal	Input			Output
		Open	High	Low	Close
511	06-10-09	8.450	8.650	8.450	8.600
512	07-10-09	8.700	8.700	8.550	8.650
513	08-10-09	8.700	8.700	8.500	8.650
514	09-10-09	8.650	8.650	8.550	8.550
515	12-10-09	8.700	8.700	8.650	8.700
516	13-10-09	8.700	8.950	8.700	8.800
517	14-10-09	8.900	8.900	8.600	8.700
518	15-10-09	8.700	8.700	8.550	8.650
519	16-10-09	8.650	8.700	8.600	8.700
520	19-10-09	8.700	8.800	8.650	8.700
521	20-10-09	8.700	8.800	8.700	8.700
522	21-10-09	8.650	8.700	8.600	8.650
523	22-10-09	8.650	8.650	8.550	8.600
524	23-10-09	8.650	8.700	8.550	8.650
525	26-10-09	8.550	8.600	8.500	8.500
526	27-10-09	8.500	8.600	8.450	8.450
527	28-10-09	8.550	8.550	8.300	8.350
528	29-10-09	8.200	8.250	7.850	8.250
529	30-10-09	8.400	8.450	8.250	8.400
530	02-11-09	8.150	8.300	8.100	8.250
531	03-11-09	8.300	8.450	8.300	8.350
532	04-11-09	8.400	8.600	8.350	8.600
533	05-11-09	8.550	8.750	8.550	8.700
534	06-11-09	8.800	8.800	8.650	8.700
535	09-11-09	8.700	8.700	8.600	8.650
536	10-11-09	8.750	8.850	8.700	8.750
537	11-11-09	8.750	8.900	8.750	8.750
538	12-11-09	8.800	8.900	8.800	8.800
539	13-11-09	8.800	8.800	8.750	8.750
540	16-11-09	8.750	8.900	8.750	8.850
541	17-11-09	8.850	9.000	8.850	8.900
542	18-11-09	8.900	9.000	8.850	8.900



## Lanjutan Lampiran 1

Data ke-	Tanggal	Input			Output
		Open	High	Low	Close
543	19-11-09	9.000	9.100	9.000	9.000
544	20-11-09	9.000	9.100	8.950	9.000
545	23-11-09	9.050	9.050	8.850	8.950
546	24-11-09	9.000	9.050	8.900	9.000
547	25-11-09	9.050	9.150	9.050	9.050
548	26-11-09	9.100	9.150	8.750	8.750
549	30-11-09	8.750	9.050	8.700	9.000
550	01-12-09	9.000	9.050	8.950	9.000
551	02-12-09	9.100	9.100	8.950	9.000
552	03-12-09	9.050	9.200	9.000	9.200
553	04-12-09	9.150	9.300	9.150	9.250
554	07-12-09	9.300	9.450	9.200	9.300
555	08-12-09	9.300	9.500	9.300	9.400
556	09-12-09	9.300	9.350	9.200	9.200
557	10-12-09	9.250	9.500	9.250	9.450
558	11-12-09	9.550	9.800	9.550	9.800
559	14-12-09	9.700	9.750	9.450	9.500
560	15-12-09	9.550	9.700	9.550	9.600
561	16-12-09	9.650	10.000	9.600	9.850
562	17-12-09	10.000	10.350	9.950	10.100
563	21-12-09	9.850	9.850	9.500	9.500
564	22-12-09	9.600	9.750	9.600	9.700
565	23-12-09	9.700	9.700	9.450	9.600
566	28-12-09	9.600	9.650	9.450	9.600
567	29-12-09	9.600	9.650	9.500	9.500
568	30-12-09	9.500	9.650	9.400	9.450
569	04-01-10	9.450	9.600	9.450	9.550
570	05-01-10	9.650	9.650	9.550	9.600
571	06-01-10	9.600	9.700	9.450	9.500
572	07-01-10	9.500	9.500	9.250	9.250
573	08-01-10	9.250	9.450	9.200	9.350
574	11-01-10	9.350	9.550	9.350	9.450

## Lanjutan Lampiran 1

Data ke-	Tanggal	Input			Output
		Open	High	Low	Close
575	12-01-10	9.400	9.550	9.400	9.500
576	13-01-10	9.450	9.500	9.350	9.400
577	14-01-10	9.450	9.500	9.350	9.350
578	15-01-10	9.350	9.500	9.350	9.500
579	18-01-10	9.400	9.500	9.400	9.400
580	19-01-10	9.400	9.450	9.350	9.450
581	20-01-10	9.600	9.600	9.400	9.500
582	21-01-10	9.350	9.500	9.350	9.450
583	22-01-10	9.300	9.350	9.250	9.300
584	25-01-10	9.200	9.400	9.200	9.250
585	26-01-10	9.300	9.400	9.300	9.300
586	27-01-10	9.300	9.350	9.200	9.250
587	28-01-10	9.300	9.400	9.300	9.300
588	29-01-10	9.250	9.450	9.250	9.350
589	01-02-10	9.350	9.450	9.300	9.300
590	02-02-10	9.350	9.450	9.250	9.300
591	03-02-10	9.400	9.400	9.300	9.350
592	04-02-10	9.300	9.350	9.200	9.250
593	05-02-10	9.000	9.100	8.850	8.950
594	08-02-10	8.950	8.950	8.600	8.700
595	09-02-10	8.750	8.750	8.400	8.550
596	10-02-10	8.650	8.700	8.450	8.550
597	11-02-10	8.600	8.750	8.500	8.750
598	12-02-10	8.750	8.750	8.600	8.700
599	15-02-10	8.700	8.700	8.600	8.700
600	16-02-10	8.700	8.800	8.650	8.800
601	17-02-10	8.850	8.900	8.700	8.750
602	18-02-10	8.750	8.750	8.550	8.700
603	19-02-10	8.600	8.650	8.500	8.600
604	22-02-10	8.650	8.650	8.350	8.500
605	23-02-10	8.400	8.500	8.200	8.500
606	24-02-10	8.400	8.500	8.300	8.450

## Lanjutan Lampiran 1

Data ke-	Tanggal	Input			Output
		Open	High	Low	Close
607	25-02-10	8.450	8.500	8.200	8.300
608	01-03-10	8.350	8.400	8.150	8.150
609	02-03-10	8.300	8.450	8.250	8.450
610	03-03-10	8.400	8.450	8.250	8.350
611	04-03-10	8.300	8.350	8.200	8.250
612	05-03-10	8.250	8.350	8.250	8.300
613	08-03-10	8.400	8.600	8.350	8.600
614	09-03-10	8.600	8.800	8.550	8.750
615	10-03-10	8.700	8.750	8.550	8.600
616	11-03-10	8.700	8.700	8.550	8.600
617	12-03-10	8.600	8.600	8.350	8.400
618	15-03-10	8.400	8.400	8.300	8.350
619	17-03-10	8.350	8.600	8.350	8.600
620	18-03-10	8.600	8.600	8.300	8.350
621	19-03-10	8.250	8.300	8.100	8.250
622	22-03-10	8.100	8.200	8.050	8.050
623	23-03-10	8.050	8.150	8.000	8.000
624	24-03-10	8.050	8.200	8.000	8.200
625	25-03-10	8.100	8.300	8.100	8.300
626	26-03-10	8.250	8.250	8.100	8.200
627	29-03-10	8.200	8.200	8.050	8.100
628	30-03-10	8.100	8.150	8.050	8.100
629	31-03-10	8.150	8.150	7.950	8.050
630	01-04-10	8.050	8.200	8.050	8.150
631	05-04-10	8.200	8.350	8.200	8.350
632	06-04-10	8.350	8.350	8.150	8.200
633	07-04-10	8.200	8.200	8.050	8.050
634	08-04-10	8.050	8.200	8.050	8.100
635	09-04-10	8.150	8.250	8.050	8.100
636	12-04-10	8.100	8.150	8.000	8.050
637	13-04-10	8.050	8.050	7.950	8.050
638	14-04-10	8.000	8.100	7.950	7.950

## Lanjutan Lampiran 1

Data ke-	Tanggal	Input			Output
		Open	High	Low	Close
639	15-04-10	8.000	8.100	7.950	8.000
640	16-04-10	8.000	8.100	7.900	8.000
641	19-04-10	7.950	8.000	7.850	7.900
642	20-04-10	7.950	8.000	7.750	7.950
643	21-04-10	8.000	8.100	8.000	8.100
644	22-04-10	8.100	8.100	7.950	8.100
645	23-04-10	8.100	8.100	7.950	8.050
646	26-04-10	8.100	8.100	7.850	7.900
647	27-04-10	7.850	7.850	7.750	7.800
648	28-04-10	7.700	7.750	7.650	7.650
649	29-04-10	7.650	7.750	7.500	7.700
650	30-04-10	7.750	7.900	7.750	7.850
651	03-05-10	7.850	8.050	7.800	8.000
652	04-05-10	8.000	8.050	7.850	7.850
653	05-05-10	7.750	7.800	7.600	7.650
654	06-05-10	7.600	7.800	7.550	7.750
655	07-05-10	7.600	7.650	7.500	7.600
656	10-05-10	7.650	7.800	7.650	7.800
657	11-05-10	7.800	7.800	7.600	7.650
658	12-05-10	7.650	7.750	7.550	7.600
659	14-05-10	7.600	7.700	7.600	7.650
660	17-05-10	7.550	7.700	7.500	7.650
661	18-05-10	7.700	7.750	7.600	7.700
662	19-05-10	7.700	7.700	7.500	7.550
663	20-05-10	7.600	7.650	7.450	7.600
664	21-05-10	7.400	7.550	7.300	7.450
665	24-05-10	7.500	7.550	7.100	7.100
666	25-05-10	7.000	7.300	6.950	7.200
667	26-05-10	7.250	7.650	7.250	7.500
668	27-05-10	7.450	7.650	7.350	7.600
669	31-05-10	7.700	7.900	7.600	7.750
670	01-06-10	7.650	7.700	7.450	7.450

## Lanjutan Lampiran 1

Data ke-	Tanggal	Input			Output
		Open	High	Low	Close
671	02-06-10	7.550	7.850	7.550	7.850
672	03-06-10	7.950	8.050	7.950	8.000
673	04-06-10	8.000	8.000	7.900	7.900
674	07-06-10	7.700	7.800	7.650	7.750
675	08-06-10	7.700	7.950	7.700	7.950
676	09-06-10	7.950	7.950	7.750	7.850
677	10-06-10	7.850	7.900	7.650	7.750
678	11-06-10	7.850	7.900	7.800	7.850
679	14-06-10	7.950	8.000	7.900	8.000
680	15-06-10	8.000	8.000	7.900	7.950
681	16-06-10	8.000	8.000	7.900	7.950
682	17-06-10	7.950	8.000	7.900	8.000
683	18-06-10	8.000	8.050	7.950	8.050
684	21-06-10	8.100	8.100	7.800	7.900
685	22-06-10	7.900	7.900	7.850	7.900
686	23-06-10	7.850	8.000	7.850	8.000
687	24-06-10	8.000	8.000	7.900	8.000
688	25-06-10	8.000	8.000	7.950	7.950
689	28-06-10	8.000	8.000	7.900	7.900
690	29-06-10	7.950	8.000	7.650	7.700
691	30-06-10	7.650	7.800	7.650	7.700
692	01-07-10	7.700	7.750	7.600	7.700
693	02-07-10	7.700	7.700	7.600	7.650
694	05-07-10	7.650	7.750	7.650	7.750
695	06-07-10	7.700	7.850	7.700	7.850
696	07-07-10	7.850	7.900	7.800	7.900
697	08-07-10	7.800	7.800	7.650	7.750
698	09-07-10	7.750	7.800	7.700	7.800
699	12-07-10	7.800	7.850	7.750	7.850
700	13-07-10	7.850	7.850	7.750	7.800
701	14-07-10	7.850	7.950	7.850	7.900
702	15-07-10	7.900	8.000	7.850	8.000



## Lanjutan Lampiran 1

Data ke-	Tanggal	Input			Output
		Open	High	Low	Close
703	16-07-10	8.000	8.100	7.950	8.100
704	19-07-10	8.000	8.200	8.000	8.100
705	21-07-10	8.150	8.150	8.100	8.150
706	22-07-10	8.150	8.150	8.050	8.100
707	23-07-10	8.200	8.250	8.150	8.150
708	26-07-10	8.200	8.250	8.000	8.050
709	27-07-10	8.100	8.200	8.100	8.200
710	28-07-10	8.250	8.250	8.150	8.250
711	29-07-10	8.200	8.250	8.150	8.250
712	30-07-10	8.250	8.450	8.250	8.450
713	02-08-10	8.450	8.450	8.300	8.400
714	03-08-10	8.450	8.500	8.000	8.000
715	04-08-10	8.100	8.200	8.050	8.100
716	05-08-10	8.150	8.250	8.150	8.200
717	06-08-10	8.250	8.400	8.200	8.400
718	09-08-10	8.400	8.500	8.400	8.500
719	10-08-10	8.500	8.500	8.400	8.400
720	11-08-10	8.400	8.450	8.350	8.450
721	12-08-10	8.400	8.500	8.350	8.400
722	13-08-10	8.450	8.500	8.450	8.500
723	16-08-10	8.500	8.550	8.400	8.550
724	18-08-10	8.550	8.700	8.550	8.600
725	19-08-10	8.700	9.000	8.650	9.000
726	20-08-10	8.900	8.950	8.700	8.800
727	23-08-10	8.800	8.800	8.650	8.700
728	24-08-10	8.700	8.800	8.650	8.700
729	25-08-10	8.700	8.950	8.700	8.900
730	26-08-10	8.900	9.000	8.850	9.000
731	27-08-10	9.000	9.000	8.600	8.750
732	30-08-10	8.800	8.850	8.700	8.700
733	31-08-10	8.700	8.700	8.500	8.650
734	01-09-10	8.750	9.000	8.750	9.000

## Lanjutan Lampiran 1

Data ke-	Tanggal	Input			Output
		Open	High	Low	Close
735	02-09-10	9.000	9.100	8.850	8.950
736	03-09-10	8.950	9.050	8.950	9.050
737	06-09-10	9.050	9.100	9.000	9.000
738	07-09-10	9.100	9.150	8.950	9.000
739	15-09-10	9.200	9.450	9.150	9.300
740	16-09-10	9.300	9.350	9.150	9.200
741	17-09-10	9.200	9.300	9.100	9.250
742	20-09-10	9.250	9.300	9.100	9.250
743	21-09-10	9.300	9.350	8.900	9.000
744	22-09-10	9.000	9.200	8.900	8.950
745	23-09-10	8.950	8.950	8.600	8.800
746	24-09-10	8.800	9.000	8.800	8.900
747	27-09-10	9.050	9.150	9.000	9.000
748	28-09-10	9.050	9.200	9.000	9.050
749	29-09-10	9.150	9.300	9.150	9.300
750	30-09-10	9.300	9.350	9.200	9.200
751	01-10-10	9.400	9.450	9.300	9.400
752	04-10-10	9.500	9.600	9.450	9.550
753	05-10-10	9.550	9.800	9.400	9.800
754	06-10-10	9.800	9.800	9.400	9.500
755	07-10-10	9.400	9.400	9.100	9.200
756	08-10-10	9.200	9.200	9.050	9.100
757	11-10-10	9.150	9.250	9.100	9.150
758	12-10-10	9.150	9.200	9.050	9.150
759	13-10-10	9.150	9.150	9.000	9.050
760	14-10-10	9.150	9.200	8.900	9.050
761	15-10-10	9.000	9.100	8.950	9.050
762	18-10-10	9.100	9.150	9.000	9.050
763	19-10-10	9.050	9.200	9.050	9.200
764	20-10-10	9.150	9.150	9.050	9.150
765	21-10-10	9.200	9.250	9.000	9.000
766	22-10-10	9.100	9.100	8.900	9.000

## Lanjutan Lampiran 1

Data ke-	Tanggal	Input			Output
		Open	High	Low	Close
767	25-10-10	9.000	9.050	8.850	8.900
768	26-10-10	8.950	9.100	8.950	9.050
769	27-10-10	9.050	9.200	9.050	9.100
770	28-10-10	9.050	9.250	9.050	9.250
771	29-10-10	9.200	9.250	9.050	9.100
772	01-11-10	9.150	9.150	8.450	8.650
773	02-11-10	8.600	8.600	8.250	8.300
774	03-11-10	8.300	8.300	8.000	8.050
775	04-11-10	8.050	8.200	7.900	8.100
776	05-11-10	8.200	8.200	7.750	7.850
777	08-11-10	7.900	8.150	7.850	8.150
778	09-11-10	8.200	8.200	8.100	8.150
779	10-11-10	8.100	8.500	8.100	8.450
780	11-11-10	8.450	8.650	8.400	8.500
781	12-11-10	8.500	8.500	8.200	8.250
782	15-11-10	8.250	8.300	8.050	8.200
783	16-11-10	8.200	8.350	8.100	8.150
784	18-11-10	8.200	8.400	8.200	8.400
785	19-11-10	8.450	8.450	8.300	8.300
786	22-11-10	8.350	8.500	8.350	8.400
787	23-11-10	8.350	8.450	8.200	8.300
788	24-11-10	8.250	8.350	8.150	8.150
789	25-11-10	8.200	8.250	8.100	8.150
790	26-11-10	8.150	8.200	7.950	8.000
791	29-11-10	8.050	8.150	8.000	8.050
792	30-11-10	8.050	8.200	7.950	7.950
793	01-12-10	8.050	8.150	7.950	8.150
794	02-12-10	8.300	8.300	8.200	8.250
795	03-12-10	8.250	8.250	8.100	8.100
796	06-12-10	8.100	8.150	8.050	8.050
797	08-12-10	8.150	8.200	7.950	8.000
798	09-12-10	8.000	8.000	7.900	8.000

## Lanjutan Lampiran 1

Data ke-	Tanggal	Input			Output
		Open	High	Low	Close
799	10-12-10	8.000	8.000	7.800	7.850
800	13-12-10	7.900	7.900	7.750	7.800
801	14-12-10	7.800	7.850	7.750	7.800
802	15-12-10	7.800	7.800	7.650	7.800
803	16-12-10	7.800	7.850	7.650	7.750
804	17-12-10	7.800	7.900	7.700	7.850
805	20-12-10	7.800	8.000	7.800	7.850
806	21-12-10	7.850	8.050	7.850	8.000
807	22-12-10	8.000	8.000	7.850	7.900
808	23-12-10	7.950	8.000	7.850	7.900
809	27-12-10	7.900	8.000	7.850	8.000
810	28-12-10	7.950	8.000	7.900	8.000
811	29-12-10	8.000	8.000	7.850	8.000
812	30-12-10	8.000	8.000	7.900	7.950
813	03-01-11	8.000	8.050	7.900	8.000
814	04-01-11	8.000	8.000	7.900	7.900
815	05-01-11	7.900	8.000	7.900	8.000
816	06-01-11	7.950	7.950	7.650	7.700
817	07-01-11	7.600	7.600	7.250	7.350
818	10-01-11	7.250	7.300	7.150	7.200
819	11-01-11	7.250	7.450	7.100	7.200
820	12-01-11	7.250	7.550	7.250	7.500
821	13-01-11	7.600	7.700	7.350	7.350
822	14-01-11	7.400	7.550	7.400	7.500
823	17-01-11	7.550	7.650	7.500	7.600
824	18-01-11	7.600	7.800	7.550	7.800
825	19-01-11	7.750	7.850	7.650	7.800
826	20-01-11	7.700	7.700	7.550	7.550
827	21-01-11	7.550	7.550	7.300	7.550
828	24-01-11	7.550	7.700	7.550	7.700
829	25-01-11	7.750	7.800	7.600	7.650
830	26-01-11	7.700	7.800	7.650	7.750

## Lanjutan Lampiran 1

Data ke-	Tanggal	Input			Output
		Open	High	Low	Close
831	27-01-11	7.800	7.800	7.600	7.650
832	28-01-11	7.650	7.750	7.550	7.750
833	31-01-11	7.650	7.650	7.500	7.550
834	01-02-11	7.550	7.600	7.450	7.550
835	02-02-11	7.600	7.750	7.600	7.650
836	04-02-11	7.700	7.800	7.650	7.750
837	07-02-11	7.800	7.850	7.650	7.700
838	08-02-11	7.700	7.800	7.650	7.700
839	09-02-11	7.700	7.850	7.650	7.800
840	10-02-11	7.700	7.750	7.550	7.650
841	11-02-11	7.600	7.600	7.450	7.600
842	14-02-11	7.600	7.600	7.450	7.450
843	16-02-11	7.450	7.500	7.350	7.450
844	17-02-11	7.450	7.450	7.350	7.450
845	18-02-11	7.400	7.500	7.400	7.450
846	21-02-11	7.450	7.600	7.450	7.600
847	22-02-11	7.550	7.550	7.450	7.450
848	23-02-11	7.550	7.550	7.450	7.450
849	24-02-11	7.400	7.500	7.350	7.350
850	25-02-11	7.400	7.600	7.400	7.450
851	28-02-11	7.550	7.550	7.450	7.450
852	01-03-11	7.550	7.550	7.450	7.500
853	02-03-11	7.450	7.500	7.350	7.350
854	03-03-11	7.400	7.450	7.350	7.350
855	04-03-11	7.400	7.450	7.250	7.300
856	07-03-11	7.300	7.400	7.300	7.350
857	08-03-11	7.350	7.400	7.300	7.300
858	09-03-11	7.350	7.500	7.350	7.450
859	10-03-11	7.400	7.400	7.350	7.400
860	11-03-11	7.350	7.350	7.250	7.300
861	14-03-11	7.300	7.350	7.250	7.350
862	15-03-11	7.300	7.300	7.150	7.150



## Lanjutan Lampiran 1

Data ke-	Tanggal	Input			Output
		Open	High	Low	Close
863	16-03-11	7.200	7.250	7.000	7.050
864	17-03-11	7.000	7.000	6.750	6.800
865	18-03-11	6.900	6.950	6.650	6.700
866	21-03-11	6.750	6.800	6.650	6.750
867	22-03-11	6.800	6.800	6.600	6.600
868	23-03-11	6.600	6.800	6.600	6.800
869	24-03-11	6.800	7.150	6.800	7.150
870	25-03-11	7.150	7.300	7.100	7.150
871	28-03-11	7.100	7.150	7.050	7.100
872	29-03-11	7.050	7.050	7.000	7.000
873	30-03-11	7.000	7.250	7.000	7.200
874	31-03-11	7.250	7.350	7.150	7.350

Sumber: [www.telkom.co.id](http://www.telkom.co.id)

## Lampiran 2 Pengkodean dengan *binary encoding*

No	Variabel	<i>Binary Encoding</i>
1	$A_1$	010011100110110
2	$B_1$	101100101010101
3	$C_1$	000111010011101
4	$D_1$	011101000110010
5	$E_1$	010011100110110
6	$F_1$	101100101010101
7	$A_2$	000111010011101
8	$B_2$	011101000110010
9	$C_2$	101110001110100
10	$D_2$	110011001101001
11	$E_2$	010011110001001
12	$F_2$	011011000110000
13	$A_3$	010011100110110
14	$B_3$	101100101010101
15	$C_3$	000111010011101
16	$D_3$	011101000110010
17	$E_3$	010011100110110
18	$F_3$	101100101010101
19	$A_4$	000111010011101
20	$B_4$	011101000110010
21	$C_4$	101110001110100
22	$D_4$	110011001101001
23	$E_4$	010011110001001
24	$F_4$	011011000110000
25	$G_4$	010011100110110
26	$H_4$	101100101010101
27	$I_4$	000111010011101
28	$J_4$	011101000110010

**Lampiran 3** Perhitungan dekode kromosom yang berisi *binary encoding* menjadi individu  $x$  yang bernilai riil

$$\begin{aligned}
 A_1 &= -10 + (10 - (-10)) \left( \begin{array}{c} g_1 \times 2^{-1} + g_2 \times 2^{-2} + \dots \\ + g_{15} \times 10^{-15} \end{array} \right) \\
 &= -10 + 20 \left( \begin{array}{c} 0 + 0,25 + 0 + 0 + 0,03125 + 0,015625 \\ 0,0078125 + 0 + 0 + 0,0009765625 \\ 0,00048828125 + 0 + 0,0001220703125 \\ 0,00006103515625 + 0 \end{array} \right) \\
 &= -3,873291016
 \end{aligned}$$

$$\begin{aligned}
 B_1 &= -10 + (10 - (-10)) \left( \begin{array}{c} g_1 \times 2^{-1} + g_2 \times 2^{-2} + \dots \\ + g_{15} \times 10^{-15} \end{array} \right) \\
 &= -10 + 20 \left( \begin{array}{c} 0,5 + 0 + 0,125 + 0,0625 + 0 + 0 \\ + 0,0078125 + 0 + 0,0001953125 + 0 \\ + 0,00048828125 + 0 + 0,0001220703125 \\ + 0,00003051757813 \end{array} \right) \\
 &= 3,958129883
 \end{aligned}$$

$$\begin{aligned}
 C_1 &= -10 + (10 - (-10)) \left( \begin{array}{c} g_1 \times 2^{-1} + g_2 \times 2^{-2} + \dots \\ + g_{15} \times 10^{-15} \end{array} \right) \\
 &= -10 + 20 \left( \begin{array}{c} 0 + 0 + 0 + 0,0625 + 0,03125 + 0,015625 \\ + 0 + 0,0039063 + 0 + 0 + + 0,00048828125 \\ + 0,000244140625 + 0,0001220703125 \\ + 0,00003051757813 \end{array} \right) \\
 &= -7,716674805
 \end{aligned}$$

$$\begin{aligned}
 D_1 &= -10 + (10 - (-10)) \left( \begin{array}{c} g_1 \times 2^{-1} + g_2 \times 2^{-2} + \dots \\ + g_{15} \times 10^{-15} \end{array} \right) \\
 &= -10 + 20 \left( \begin{array}{c} 0 + 0,25 + 0,125 + 0,0625 + 0 + 0,015625 \\ + 0 + 0 + 0 + 0,0009765625 \\ + 0,00048828125 + 0 + 0 \\ + 0,00006103515625 + 0 \end{array} \right) \\
 &= -0,906982421
 \end{aligned}$$

$$E_1 = -10 + (10 - (-10)) \left( \begin{array}{c} g_1 \times 2^{-1} + g_2 \times 2^{-2} + \dots \\ + g_{15} \times 10^{-15} \end{array} \right)$$

$$= -10 + 20 \left( \begin{array}{c} 0 + 0,25 + 0 + 0 + 0,03125 + 0,015625 \\ + 0,0078125 + 0 + 0 + 0,0009765625 \\ + 0,00048828125 + 0 + 0,0001220703125 \\ + 0,00006103515625 + 0 \end{array} \right)$$

$$= -3,873291016$$

$$F_1 = -10 + (10 - (-10)) \left( \begin{array}{c} g_1 \times 2^{-1} + g_2 \times 2^{-2} + \dots \\ + g_{15} \times 10^{-15} \end{array} \right)$$

$$= -10 + 20 \left( \begin{array}{c} 0,5 + 0 + 0,125 + 0,0625 + 0 + 0 \\ + 0,0078125 + 0 + 0,0001953125 + 0 \\ + 0,00048828125 + 0 + 0,0001220703125 \\ + 0,00003051757813 \end{array} \right)$$

$$= 3,958129883$$

$$A_2 = -10 + (10 - (-10)) \left( \begin{array}{c} g_1 \times 2^{-1} + g_2 \times 2^{-2} + \dots \\ + g_{15} \times 10^{-15} \end{array} \right)$$

$$= -10 + 20 \left( \begin{array}{c} 0 + 0 + 0 + 0,0625 + 0,03125 + 0,015625 \\ + 0 + 0,0039063 + 0 + 0 + + 0,00048828125 \\ + 0,000244140625 + 0,0001220703125 \\ + 0,00003051757813 \end{array} \right)$$

$$= -7,716674805$$

$$B_2 = -10 + (10 - (-10)) \left( \begin{array}{c} g_1 \times 2^{-1} + g_2 \times 2^{-2} + \dots \\ + g_{15} \times 10^{-15} \end{array} \right)$$

$$= -10 + 20 \left( \begin{array}{c} 0 + 0,25 + 0,125 + 0,0625 + 0 + 0,015625 \\ + 0 + 0 + 0 + 0,0009765625 \\ + 0,00048828125 + 0 + 0 \\ + 0,00006103515625 + 0 \end{array} \right)$$

$$= -0,906982421$$

$$C_2 = -10 + (10 - (-10)) \left( \begin{array}{c} g_1 \times 2^{-1} + g_2 \times 2^{-2} + \dots \\ + g_{15} \times 10^{-15} \end{array} \right)$$

$$= -10 + 20 \left( \begin{array}{c} 0,5 + 0 + 0,125 + 0,0625 + 0,03125 \\ 0 + 0 + 0 + 0,0001953125 \\ 0,0009765625 + 0,00048828125 + 0 \\ + 0,0001220703125 + 0 + 0 \end{array} \right)$$

$$= 4,445800781$$

$$D_2 = -10 + (10 - (-10)) \left( \begin{array}{c} g_1 \times 2^{-1} + g_2 \times 2^{-2} + \dots \\ + g_{15} \times 10^{-15} \end{array} \right)$$

$$= -10 + 20 \left( \begin{array}{c} 0,5 + 0,25 + 0 + 0 + 0,03125 + 0,015625 \\ + 0 + 0 + 0,0001953125 + 0,0009765625 \\ + 0 + 0,000244140625 + 0 + 0 \\ + 0,00003051757813 \end{array} \right)$$

$$= 6,001586914$$

$$E_2 = -10 + (10 - (-10)) \left( \begin{array}{c} g_1 \times 2^{-1} + g_2 \times 2^{-2} + \dots \\ + g_{15} \times 10^{-15} \end{array} \right)$$

$$= -10 + 20 \left( \begin{array}{c} 0 + 0,25 + 0 + 0 + 0,03125 + 0,015625 \\ + 0,0078125 + 0,0039063 + 0 + 0 + 0 \\ + 0,000244140625 + 0 + 0 \\ + 0,00003051757813 \end{array} \right)$$

$$= -3,822631836$$

$$F_2 = -10 + (10 - (-10)) \left( \begin{array}{c} g_1 \times 2^{-1} + g_2 \times 2^{-2} + \dots \\ + g_{15} \times 10^{-15} \end{array} \right)$$

$$= -10 + 20 \left( \begin{array}{c} 0 + 0,25 + 0,125 + 0 + 0,03125 + 0,015625 \\ + 0 + 0 + 0 + 0,0009765625 \\ + 0,00048828125 + 0 + 0 + 0 + 0 \end{array} \right)$$



$$= -1,533203125$$

$$A_3 = -10 + (10 - (-10)) \left( \begin{array}{c} g_1 \times 2^{-1} + g_2 \times 2^{-2} + \dots \\ + g_{15} \times 10^{-15} \end{array} \right)$$

$$= -10 + 20 \left( \begin{array}{c} 0 + 0,25 + 0 + 0 + 0,03125 + 0,015625 \\ 0,0078125 + 0 + 0 + 0,0009765625 \\ 0,00048828125 + 0 + 0,0001220703125 \\ 0,00006103515625 + 0 \end{array} \right)$$

$$= -3,873291016$$

$$B_3 = -10 + (10 - (-10)) \left( \begin{array}{c} g_1 \times 2^{-1} + g_2 \times 2^{-2} + \dots \\ + g_{15} \times 10^{-15} \end{array} \right)$$

$$= -10 + 20 \left( \begin{array}{c} 0,5 + 0 + 0,125 + 0,0625 + 0 + 0 \\ + 0,0078125 + 0 + 0,0001953125 + 0 \\ + 0,00048828125 + 0 + 0,0001220703125 \\ + 0,00003051757813 \end{array} \right)$$

$$= 3,958129883$$

$$C_3 = -10 + (10 - (-10)) \left( \begin{array}{c} g_1 \times 2^{-1} + g_2 \times 2^{-2} + \dots \\ + g_{15} \times 10^{-15} \end{array} \right)$$

$$= -10 + 20 \left( \begin{array}{c} 0 + 0 + 0 + 0,0625 + 0,03125 + 0,015625 \\ + 0 + 0,0039063 + 0 + 0 + + 0,00048828125 \\ + 0,000244140625 + 0,0001220703125 \\ + 0,00003051757813 \end{array} \right)$$

$$= -7,716674805$$

$$D_3 = -10 + (10 - (-10)) \left( \begin{array}{c} g_1 \times 2^{-1} + g_2 \times 2^{-2} + \dots \\ + g_{15} \times 10^{-15} \end{array} \right)$$

$$\begin{aligned}
 &= -10 + 20 \begin{pmatrix} 0 + 0,25 + 0,125 + 0,0625 + 0 + 0,015625 \\ +0 + 0 + 0 + 0,0009765625 \\ +0,00048828125 + 0 + 0 \\ +0,00006103515625 + 0 \end{pmatrix} \\
 &= -0,906982421
 \end{aligned}$$

$$\begin{aligned}
 E_3 &= -10 + (10 - (-10)) \left( \begin{array}{l} g_1 \times 2^{-1} + g_2 \times 2^{-2} + \dots \\ + g_{15} \times 10^{-15} \end{array} \right) \\
 &= -10 + 20 \begin{pmatrix} 0 + 0,25 + 0 + 0 + 0,03125 + 0,015625 \\ +0,0078125 + 0 + 0 + 0,0009765625 \\ +0,00048828125 + 0 + 0,0001220703125 \\ +0,00006103515625 + 0 \end{pmatrix} \\
 &= -3,873291016
 \end{aligned}$$

$$\begin{aligned}
 F_3 &= -10 + (10 - (-10)) \left( \begin{array}{l} g_1 \times 2^{-1} + g_2 \times 2^{-2} + \dots \\ + g_{15} \times 10^{-15} \end{array} \right) \\
 &= -10 + 20 \begin{pmatrix} 0,5 + 0 + 0,125 + 0,0625 + 0 + 0 \\ +0,0078125 + 0 + 0,0001953125 + 0 \\ +0,00048828125 + 0 + 0,0001220703125 \\ +0,00003051757813 \end{pmatrix} \\
 &= 3,958129883
 \end{aligned}$$

$$\begin{aligned}
 A_4 &= -10 + (10 - (-10)) \left( \begin{array}{l} g_1 \times 2^{-1} + g_2 \times 2^{-2} + \dots \\ + g_{15} \times 10^{-15} \end{array} \right) \\
 &= -10 + 20 \begin{pmatrix} 0 + 0 + 0 + 0,0625 + 0,03125 + 0,015625 \\ +0 + 0,0039063 + 0 + 0 + +0,00048828125 \\ +0,000244140625 + 0,0001220703125 \\ +0,00003051757813 \end{pmatrix} \\
 &= -7,716674805
 \end{aligned}$$

$$\begin{aligned}
 B_4 &= -10 + (10 - (-10)) \left( \begin{array}{c} g_1 \times 2^{-1} + g_2 \times 2^{-2} + \dots \\ + g_{15} \times 10^{-15} \end{array} \right) \\
 &= -10 + 20 \left( \begin{array}{c} 0 + 0,25 + 0,125 + 0,0625 + 0 + 0,015625 \\ + 0 + 0 + 0 + 0,0009765625 \\ + 0,00048828125 + 0 + 0 \\ + 0,00006103515625 + 0 \end{array} \right) \\
 &= -0,906982421
 \end{aligned}$$

$$\begin{aligned}
 C_4 &= -10 + (10 - (-10)) \left( \begin{array}{c} g_1 \times 2^{-1} + g_2 \times 2^{-2} + \dots \\ + g_{15} \times 10^{-15} \end{array} \right) \\
 &= -10 + 20 \left( \begin{array}{c} 0,5 + 0 + 0,125 + 0,0625 + 0,03125 \\ 0 + 0 + 0 + 0,0001953125 \\ 0,0009765625 + 0,00048828125 + 0 \\ + 0,0001220703125 + 0 + 0 \end{array} \right) \\
 &= 4,445800781
 \end{aligned}$$

$$\begin{aligned}
 D_4 &= -10 + (10 - (-10)) \left( \begin{array}{c} g_1 \times 2^{-1} + g_2 \times 2^{-2} + \dots \\ + g_{15} \times 10^{-15} \end{array} \right) \\
 &= -10 + 20 \left( \begin{array}{c} 0,5 + 0,25 + 0 + 0 + 0,03125 + 0,015625 \\ + 0 + 0 + 0,0001953125 + 0,0009765625 \\ + 0 + 0,000244140625 + 0 + 0 \\ + 0,00003051757813 \end{array} \right) \\
 &= 6,001586914
 \end{aligned}$$

$$\begin{aligned}
 E_4 &= -10 + (10 - (-10)) \left( \begin{array}{c} g_1 \times 2^{-1} + g_2 \times 2^{-2} + \dots \\ + g_{15} \times 10^{-15} \end{array} \right) \\
 &= -10 + 20 \left( \begin{array}{c} 0 + 0,25 + 0 + 0 + 0,03125 + 0,015625 \\ + 0,0078125 + 0,0039063 + 0 + 0 + 0 \\ + 0,000244140625 + 0 + 0 \\ + 0,00003051757813 \end{array} \right)
 \end{aligned}$$

$$= -3,822631836$$

$$F_4 = -10 + (10 - (-10)) \left( \begin{array}{c} g_1 \times 2^{-1} + g_2 \times 2^{-2} + \dots \\ + g_{15} \times 10^{-15} \end{array} \right)$$

$$= -10 + 20 \left( \begin{array}{c} 0 + 0,25 + 0,125 + 0 + 0,03125 + 0,015625 \\ + 0 + 0 + 0 + 0,0009765625 \\ + 0,00048828125 + 0 + 0 + 0 + 0 \end{array} \right)$$

$$= -1,533203125$$

$$G_4 = -10 + (10 - (-10)) \left( \begin{array}{c} g_1 \times 2^{-1} + g_2 \times 2^{-2} + \dots \\ + g_{15} \times 10^{-15} \end{array} \right)$$

$$= -10 + 20 \left( \begin{array}{c} 0 + 0,25 + 0 + 0 + 0,03125 + 0,015625 \\ 0,0078125 + 0 + 0 + 0,0009765625 \\ 0,00048828125 + 0 + 0,0001220703125 \\ 0,00006103515625 + 0 \end{array} \right)$$

$$= -3,873291016$$

$$H_4 = -10 + (10 - (-10)) \left( \begin{array}{c} g_1 \times 2^{-1} + g_2 \times 2^{-2} + \dots \\ + g_{15} \times 10^{-15} \end{array} \right)$$

$$= -10 + 20 \left( \begin{array}{c} 0,5 + 0 + 0,125 + 0,0625 + 0 + 0 \\ + 0,0078125 + 0 + 0,0001953125 + 0 \\ + 0,00048828125 + 0 + 0,0001220703125 \\ + 0,00003051757813 \end{array} \right)$$

$$= 3,958129883$$

$$I_4 = -10 + (10 - (-10)) \left( \begin{array}{c} g_1 \times 2^{-1} + g_2 \times 2^{-2} + \dots \\ + g_{15} \times 10^{-15} \end{array} \right)$$

$$= -10 + 20 \left( \begin{array}{c} 0 + 0 + 0 + 0,0625 + 0,03125 + 0,015625 \\ + 0 + 0,0039063 + 0 + 0 + + 0,00048828125 \\ + 0,000244140625 + 0,0001220703125 \\ + 0,00003051757813 \end{array} \right)$$

$$= -7,716674805$$

$$J_4 = -10 + (10 - (-10)) \left( \begin{array}{l} g_1 \times 2^{-1} + g_2 \times 2^{-2} + \dots \\ + g_{15} \times 10^{-15} \end{array} \right)$$

$$= -10 + 20 \left( \begin{array}{l} 0 + 0,25 + 0,125 + 0,0625 + 0 + 0,015625 \\ + 0 + 0 + 0 + 0,0009765625 \\ + 0,00048828125 + 0 + 0 \\ + 0,00006103515625 + 0 \end{array} \right)$$

$$= -0,906982421$$

Hasil pengkodean kromosom

<i>FFNN struktur(i)</i>	Variabel	Nilai
1	$A_1$	-3,873291016
2	$B_1$	3,958129883
3	$C_1$	-7,716674805
4	$D_1$	-0,906982421
5	$E_1$	-3,873291016
6	$F_1$	3,958129883
7	$A_2$	-7,716674805
8	$B_2$	-0,906982421
9	$C_2$	4,445800781
10	$D_2$	6,001586914
11	$E_2$	-3,822631836
12	$F_2$	-1,533203125
13	$A_3$	-3,873291016
14	$B_3$	3,958129883
15	$C_3$	-7,716674805
16	$D_3$	-0,906982421
17	$E_3$	-3,873291016
18	$F_3$	3,958129883
19	$A_4$	-7,716674805
20	$B_4$	-0,906982421
21	$C_4$	4,445800781



22	$D_4$	6,001586914
23	$E_4$	-3,822631836
24	$F_4$	-1,533203125
25	$G_4$	-3,873291016
26	$H_4$	3,958129883
27	$I_4$	-7,716674805
28	$J_4$	-0,906982421

UNIVERSITAS BRAWIJAYA



#### Lampiran 4 Langkah pelatihan data

Data 1

$$x_1 = 0,5778$$

$$x_2 = 0,5944$$

$$x_3 = 0,5778$$

$$Y = 0,5944(\text{target})$$

$$\begin{aligned} Z_{1in} &= A_1 + B_1x_1 + C_1x_2 + D_1x_1^2 + E_1x_2^2 + F_1x_1x_2 \\ &= -3,873291016 + (3,958129883)(0,5778) \\ &\quad + (-7,716674805)(0,5944) + (-0,906982421) \\ &\quad (0,5778)^2 + (-3,873291016)(0,5944)^2 \\ &\quad + (3,958129883)(0,5778)(0,5944) \\ &= -6,484954221 \end{aligned}$$

$$Z_{1out} = \frac{1}{1 + \exp(6,484954221)} = 0,001523904804$$

$$\begin{aligned} Z_{2in} &= A_2 + B_2x_1 + C_2x_3 + D_2x_1^2 + E_2x_3^2 + F_2x_1x_3 \\ &= (-7,716674805) + (-0,906982421)(0,5778) \\ &\quad + (4,445800781)(0,5778) + (6,001586914)(0,5778)^2 \\ &\quad + (-3,822631836)(0,5778)^2 + (-1,533203125) \\ &\quad (0,5778)(0,5778) \\ &= -5,456359433 \end{aligned}$$

$$Z_{2out} = \frac{1}{1 + \exp(5,456359433)} = 0,004250921841$$

$$\begin{aligned} Z_{3in} &= A_3 + B_3x_2 + C_3x_3 + D_3x_2^2 + E_3x_3^2 + F_3x_2x_3 \\ &= (-3,873291016) + (3,958129883)(0,5944) \\ &\quad + (-7,716674805)(0,5778) + (-0,906982421) \\ &\quad (0,5944)^2 + (-3,873291016)(0,5778)^2 \\ &\quad + (3,958129883)(0,5944)(0,5778) \\ &= -6,233432488 \end{aligned}$$

$$Z_{3out} = \frac{1}{1 + \exp(6,233432488)} = 0,001958858707$$

$$\begin{aligned}
Y_{in} &= A_4 + B_4z_1 + C_4z_2 + D_4z_3 + E_4z_1^2 + F_4z_2^2 + G_4z_3^2 \\
&\quad + H_4z_1z_2 + I_4z_1z_3 + J_4z_2z_3 \\
&= (-7,716674805) + (-0,906982421)(0,001523904804) \\
&\quad + (4,445800781)(0,0042509218417) + (6,001586914) \\
&\quad (0,001958858707) + (-3,822631836) \\
&\quad (0,001523904804)^2 + (-1,533203125) \\
&\quad (0,0042509218417)^2 + (-3,873291016) \\
&\quad (0,001958858707)^2 + (3,958129883) \\
&\quad (0,001523904804)(0,0042509218417) \\
&\quad + (-7,716674805)(0,001523904804) \\
&\quad (0,001958858707) + (-0,906982421) \\
&\quad (0,0042509218417)(0,001958858707) \\
&= -7,699191783
\end{aligned}$$

$$Y_{out} = \frac{1}{1 + \exp(7,699191783)} = 0,0004529880222$$



## Lampiran 5 Program GPNN

```
[JPmasukan JumMasukan]=size(x);  
m=input('derajat polinomial= ');  
Nbit    = 4;  
Nvar    = variabel(m);  
JumGen  = Nbit*Nvar;  
Rb      = -10;  
Ra      = 10;  
MinDelta = 0.01;  
Fthreshold = 1/MinDelta;  
Bgraf   = Fthreshold;  
UkPop   = 50;  
Psilang = 0.95;  
Pmutasi = 0.01;  
MaxG    = 500;
```

```
Populasi = InisialisasiPopulasi(UkPop,JumGen);  
for generasi=1:MaxG,  
    FFNNstruk = DekodekanKromosom(Populasi(1,:),Nvar,Nbit,Ra,Rb);  
    Fitness(1) =  
    BinaryEvalInd(FFNNstruk,JumMasukan,JPmasukan,x,y,m);  
    MaxF = Fitness(1);  
    MinF = Fitness(1);  
    IndeksIndividuTerbaik = 1;  
    for ii=2:UkPop,  
        FFNNstruk =  
        DekodekanKromosom(Populasi(ii,:),Nvar,Nbit,Ra,Rb);  
        Fitness(ii) =  
        BinaryEvalInd(FFNNstruk,JumMasukan,JPmasukan,x,y,m);  
        if (Fitness(ii) > MaxF),  
            MaxF = Fitness(ii);  
            IndeksIndividuTerbaik = ii;  
            FFNNterbaik = FFNNstruk;  
        end  
        if (Fitness(ii) <= MinF),  
            MinF = Fitness(ii);  
        end
```

```

end
FitnessRataRata = mean(Fitness);
if MaxF > Fthreshold,
break;
end

TemPopulasi = Populasi;
if mod(UkPop,2)==0,
IterasiMulai = 3;
TemPopulasi(1,:) = Populasi(IndeksIndividuTerbaik,:);
TemPopulasi(2,:) = Populasi(IndeksIndividuTerbaik,:);
else
IterasiMulai = 2;
TemPopulasi(1,:) = Populasi(IndeksIndividuTerbaik,:);
end
LinearFitness = LinearFitnessRanking(UkPop,Fitness,MaxF,MinF);
for jj=IterasiMulai:2:UkPop,
IP1 = RouletteWheel(UkPop,LinearFitness);
IP2 = RouletteWheel(UkPop,LinearFitness);
if (rand < Psilang),
Anak =
PindahSilang(Populasi(IP1,:),Populasi(IP2,:),JumGen);
TemPopulasi(jj,:) = Anak(1,:);
TemPopulasi(jj+1,:) = Anak(2,:);
else
TemPopulasi(jj,:) = Populasi(IP1,:);
TemPopulasi(jj+1,:) = Populasi(IP2,:);
end
end
for kk=IterasiMulai:UkPop,
TemPopulasi(kk,:) =
Mutasi(TemPopulasi(kk,:),JumGen,Pmutasi);
end
Populasi = TemPopulasi;
end
FFNN_Stuktur_Terbaik=FFNNterbaik';
save ('FFNNterbaik.mat', 'FFNNterbaik', 'MaxF', 'm')

```