

## PELATIHAN *FEED FORWARD NEURAL NETWORK* MENGGUNAKAN ALGORITMA GENETIKA DENGAN METODE SELEKSI TURNAMEN UNTUK DATA *TIME SERIES*

David Yuliandar<sup>1</sup>, Budi Warsito<sup>2</sup>, Hasbi Yasin<sup>3</sup>

<sup>1</sup>Mahasiswa Jurusan Statistika FSM UNDIP

<sup>2,3</sup>Staf Pengajar Jurusan Statistika FSM UNDIP

### ABSTRAK

Pemodelan *time series* seringkali dikaitkan dengan proses peramalan suatu nilai karakteristik tertentu pada periode mendatang. Salah satu metode peramalan yang berkembang saat ini adalah menggunakan *artificial neural network* atau yang lebih dikenal dengan *neural network*. Penggunaan *neural network* dalam peramalan *time series* dapat menjadi solusi yang baik, namun yang menjadi masalah adalah arsitektur jaringan dan pemilihan metode pelatihan yang tepat. Salah satu pilihan yang mungkin adalah menggunakan algoritma genetika. Algoritma genetika adalah suatu algoritma pencarian stokastik berdasarkan cara kerja melalui mekanisme seleksi alam dan genetik yang bertujuan untuk mendapatkan solusi dari suatu masalah. Algoritma ini dapat digunakan sebagai metode pembelajaran dalam melatih model *feed forward neural network*. Penerapan algoritma genetika dan *neural network* untuk peramalan *time series* bertujuan untuk mendapatkan bobot-bobot yang optimum dengan meminimumkan *error*. Dari hasil pelatihan dan pengujian pada data kurs Dolar Australia terhadap Rupiah didapatkan nilai RMSE sebesar 117.3599 dan 82.4917. Model ini baik untuk digunakan karena memberikan hasil prediksi yang cukup akurat yang ditunjukkan oleh kedekatan target dengan output.

**Kata kunci:** peramalan *time series*, *feed forward neural network*, algoritma genetika, bobot, kurs

### 1. PENDAHULUAN

Pemodelan *time series* seringkali dikaitkan dengan proses peramalan (*forecasting*) suatu nilai karakteristik tertentu pada periode mendatang. Peramalan adalah menduga atau memperkirakan suatu keadaan di masa yang akan datang berdasarkan keadaan masa lalu dan sekarang yang diperlukan untuk menetapkan kapan suatu peristiwa akan terjadi, sehingga tindakan yang tepat dapat dilakukan (Makridakis, dkk., 1999). Pada umumnya peramalan *time series* dilakukan dengan menggunakan metode *Autoregressive Integrated Moving Average* (ARIMA), *exponential smoothing*, dekomposisi atau regresi. Meskipun pendekatan semacam ini efisien untuk peramalan *time series*, namun masih menunjukkan kekurangan ketika terjadi gangguan *noise* atau hubungan nonlinier lainnya yang umum terjadi pada data finansial. Salah satu metode peramalan yang berkembang saat ini adalah dengan menggunakan *Artificial Neural Network* (ANN) atau yang lebih dikenal dengan *Neural Network* (NN). NN memiliki kemampuan untuk mempelajari dan mengadaptasi situasi baru dengan mengingat pola data masa lalu bahkan ketika terjadi *noise*. Penerapan NN untuk peramalan *time series* bertujuan untuk mengatasi keterbatasan pada metode peramalan konvensional terutama bila digunakan pada data finansial.

Penggunaan NN dalam peramalan *time series* dapat menjadi solusi yang baik, namun yang menjadi masalah adalah arsitektur jaringan dan pemilihan metode pelatihan yang tepat. Salah satu pilihan yang mungkin adalah dengan menggunakan Algoritma Genetika (AG). AG

sangat cocok untuk menyelesaikan masalah kombinatorial yang membutuhkan waktu komputasi lama. Oleh karena itu, integrasi antara NN dan AG untuk peramalan *time series* memberikan keuntungan dari kedua metode tersebut. Studi kasus lainnya menunjukkan bahwa NN dan AG memberikan hasil yang baik dan lebih akurat untuk peramalan *time series* dibandingkan dengan metode konvensional (Neves and Cortez, 1998). Dalam tulisan ini akan dijelaskan mengenai penggunaan AG dalam melatih model *Feed Forward Neural Network* (FFNN) serta penerapannya pada data harian kurs Dolar Australia terhadap Rupiah untuk menentukan bobot atau parameter yang optimum sehingga menghasilkan nilai prediksi yang sesuai dengan nilai aktualnya.

## 2. TINJAUAN PUSTAKA

### Algoritma Genetika

Algoritma genetika adalah suatu algoritma pencarian (*searching*) stokastik berdasarkan cara kerja melalui mekanisme seleksi alam dan genetik. Tujuannya untuk menentukan struktur-struktur yang disebut individu yang berkualitas tinggi di dalam suatu domain yang disebut populasi untuk mendapatkan solusi suatu persoalan (Pandjaitan, 2007). Pada algoritma ini, teknik pencarian dilakukan sekaligus atas sejumlah solusi yang mungkin yang dikenal dengan istilah *populasi*. Masing-masing individu di dalam populasi disebut *kromosom*, yang merepresentasikan suatu penyelesaian terhadap masalah yang sedang ditangani. Sebuah kromosom berisi sejumlah *gen*, yang mengkodekan informasi yang disimpan di dalam kromosom. Sebuah kromosom berkembang biak melalui berbagai iterasi yang berulang-ulang yang disebut *generasi*. Pada setiap generasi, kromosom-kromosom yang dihasilkan akan dievaluasi menggunakan suatu pengukuran, *fitness*. Untuk menghasilkan suatu generasi baru, dilakukan penyeleksian berdasarkan nilai *fitness* untuk menentukan kromosom orang tua (*parent*) yang akan menghasilkan kromosom anak (*offspring*) yang dibentuk dengan menggabungkan dua buah kromosom orang tua yang terpilih menggunakan operator kawin silang (*crossover*) dan memodifikasi sebuah kromosom menggunakan operator mutasi. Setelah melalui beberapa generasi maka algoritma ini akan konvergen ke kromosom terbaik (Fadlisyah, dkk., 2009).

Pada dasarnya ada beberapa komponen utama yang ada dalam AG tetapi banyak variasi metode yang diusulkan pada masing-masing komponen tersebut. Masing-masing metode tersebut mempunyai kelebihan dan kekurangan tersendiri. Prosedur Inisialisasi adalah membangkitkan sejumlah individu atau kromosom secara acak sebagai populasi awal. Ukuran populasi tergantung pada masalah yang akan dipecahkan dan jenis operator genetika yang akan diimplementasikan. Setelah ukuran populasi ditentukan, kemudian harus dilakukan inisialisasi terhadap kromosom yang terdapat pada populasi tersebut. Inisialisasi kromosom dilakukan secara acak, namun demikian harus tetap memperhatikan domain solusi dan kendala permasalahan yang ada (Kusumadewi, 2003).

Dalam AG, gen-gen yang diinisialisasi sebagai estimasi nilai mula-mula mengandung informasi dalam bentuk kode-kode tertentu. Satu gen mewakili satu parameter yang akan diestimasi nilainya agar suatu fungsi optimal. Tujuan dari pendekodean kromosom ini adalah mempermudah perhitungan untuk mencari nilai *fitness* dari setiap kromosom. Dengan menggunakan suatu interval tertentu, jika batas bawah  $r_b$  dan batas atas  $r_a$ , kode suatu gen  $g$  dan kode bit-bit yang menyusun suatu gen  $g_1, g_2, \dots, g_n$  maka pendekodean dapat dilakukan dengan cara sebagai berikut

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

(Suyanto, 2005)

Suatu individu dievaluasi berdasarkan fungsi tertentu sebagai ukuran performansinya atau yang disebut nilai *fitness*. Penentuan nilai *fitness* sangat berpengaruh pada performansi AG secara keseluruhan. Dikarenakan seleksi dilakukan secara random, maka tidak ada jaminan bahwa suatu individu bernilai *fitness* tertinggi akan selalu dipilih. Untuk menjaga agar individu bernilai *fitness* tertinggi tersebut tidak hilang selama evolusi, maka perlu dibuat satu atau beberapa salinan. Prosedur ini dikenal dengan istilah elitisme. Untuk menghindari kecenderungan konvergen pada solusi optimal lokal dengan diperoleh nilai *fitness* baru yang memiliki variansi yang lebih besar maka perlu dilakukan penskalaan nilai *fitness* dengan menggunakan persamaan

$$f(i) = f_{max} - (f_{max} - f_{min}) \left( \frac{R(i) - 1}{N - 1} \right)$$

(Suyanto, 2005)

Seleksi dilakukan dengan memilih dua kromosom atau individu untuk dijadikan pasangan orang tua. Metode seleksi yang digunakan adalah *tournament selection*. Dalam bentuk paling sederhana, metode ini mengambil dua kromosom secara random dan kemudian menyeleksi salah satu yang bernilai *fitness* tertinggi untuk menjadi orang tua pertama. Cara yang sama dilakukan lagi untuk mendapatkan orang tua yang kedua. Metode *tournament selection* yang lebih rumit adalah dengan mengambil  $k$  kromosom secara random. Kemudian kromosom bernilai *fitness* tertinggi dipilih sebagai orang tua pertama jika bilangan random yang dibangkitkan kurang dari suatu nilai batas yang ditentukan  $p_k$  dalam interval  $[0,1]$ . Pemilihan orang tua akan dilakukan secara random dari  $k-1$  kromosom yang ada jika bilangan random yang dibangkitkan lebih dari atau sama dengan  $p_k$ . Biasanya  $k$  diset sebagai suatu nilai yang sangat kecil, misal 4 atau 5. Sedangkan  $p_k$  biasanya diset sekitar 0.75 (Suyanto, 2005).

Seleksi dilakukan untuk mendapatkan pasangan orang tua yang nantinya akan digunakan untuk kawin silang (*crossover*) dan mutasi. Proses ini dapat terjadi jika suatu bilangan random yang muncul kurang dari probabilitas kawin silang ( $p_c$ ) dan probabilitas mutasi ( $p_m$ ) yang ditentukan. Generasi baru didapatkan dengan cara mengganti anggota populasi awal hasil inisialisasi dengan populasi baru yang terdiri dari kromosom-kromosom hasil elitisme, kawin silang dan mutasi.

### **Neural Network**

Jaringan syaraf tiruan (*Artificial Neural Network*) atau yang dikenal dengan istilah *Neural Network* (NN) merupakan sistem pemroses informasi yang memiliki karakteristik mirip dengan jaringan syaraf biologi, dimana dalam memproses informasi, otak manusia terdiri dari sejumlah neuron yang melakukan tugas sederhana. Karena adanya keterhubungan antar neuron, maka otak dapat melakukan fungsi pemrosesan yang cukup kompleks. Pemrosesan informasi pada manusia bersifat adaptif, yang artinya hubungan antar neuron terjadi secara dinamis dan selalu memiliki kemampuan untuk mempelajari informasi-informasi yang belum diketahui sebelumnya (Fausett, 1994).

Secara garis besar pada NN memiliki dua tahap pemrosesan informasi, yaitu tahap pelatihan dan tahap pengujian. Tahap pelatihan dimulai dengan memasukkan pola-pola belajar (data latih) ke dalam jaringan. Dengan menggunakan pola-pola ini, jaringan akan mengubah-ubah bobot yang menjadi penghubung antara node. Pada setiap iterasi dilakukan evaluasi terhadap output jaringan. Tahap ini berlangsung pada beberapa iterasi dan berhenti setelah jaringan menemukan bobot yang sesuai dimana nilai *error* yang diinginkan telah tercapai atau jumlah iterasi telah mencapai nilai maksimal yang ditetapkan. Selanjutnya bobot ini menjadi

*knowledge base* (dasar pengetahuan) pada tahap pengenalan. Sedangkan pada tahap pengujian dilakukan pengujian terhadap suatu pola masukan yang belum pernah dilatihkan sebelumnya (data uji) dengan menggunakan bobot hasil tahap pelatihan. Diharapkan bobot-bobot hasil pelatihan yang sudah menghasilkan *error* minimal juga akan memberikan *error* yang kecil pada tahap pengujian (Warsito, 2009).

Arsitektur jaringan dalam NN merupakan susunan dari neuron-neuron dalam lapisan dan pola keterhubungannya dalam dan antar lapisan. Neuron-neuron yang berada dalam satu lapisan tertentu akan mempunyai pola keterhubungan yang sama (Fausett, 1994). Sebuah NN biasanya terdiri atas lapisan-lapisan antara lain lapisan input (*input layer*) yaitu lapisan yang menerima masukan atau input dari jaringan luar, lapisan tersembunyi (*hidden layer*) yaitu lapisan yang terletak dalam satu atau beberapa lapisan dan tidak berhubungan langsung dengan keadaan diluar jaringan serta lapisan output (*output layer*) yaitu lapisan yang menghasilkan output dari jaringan. Berdasarkan jumlah *layer* yang dimiliki, NN dibedakan menjadi jaringan lapisan tunggal (*single layer*) dan jaringan multilapis (*multilayer*). Pada *single layer*, jaringan tersebut tidak mempunyai lapisan tersembunyi dari neuron atau hanya mempunyai satu lapisan bobot koneksi serta sekumpulan input neuron dihubungkan langsung dengan sekumpulan output. Sedangkan pada *multilayer* terdiri dari lapisan input, lapisan tersembunyi dan lapisan output. Lapisan tersembunyi terletak diantara lapisan input dan lapisan output. Output dari sebuah lapisan tersembunyi akan menjadi input bagi lapisan berikutnya. Jaringan ini paling tidak mempunyai satu lapisan tersembunyi (Siang, 2005).

### 3. METODOLOGI PENELITIAN

Data yang digunakan adalah data sekunder, yaitu data harian nilai tukar mata uang Dolar Australia (AUD) terhadap Rupiah periode 3 Januari 2011 sampai dengan 29 Agustus 2012, dimana terdapat 348 data kurs jual yang tercatat. Prosedur AG sebagai metode pelatihan dalam FFNN dilakukan dengan

1. Meginisialisasi populasi sebagai ruang solusi yang berisi kromosom-kromosom
2. Setiap kromosom merepresentasikan beberapa gen dimana ketika didekodekan akan menghasilkan bobot atau parameter jaringan
3. Kromosom-kromosom tersebut akan dievaluasi dengan menggunakan fungsi objektif tertentu untuk mendapatkan nilai *fitness*
4. Generasi baru diperoleh dengan menyeleksi kromosom menggunakan metode seleksi tertentu serta menggunakan operator genetika
5. Setelah melalui beberapa generasi yang ditentukan, maka AG akan konvergen ke kromosom terbaik dan diperoleh solusi berupa nilai bobot atau parameter yang optimum

### 4. HASIL DAN PEMBAHASAN

Penggunaan Algoritma Genetika (AG) sebagai metode pembelajaran pada *Neural Network* (NN) merupakan suatu prosedur pelatihan jaringan yang terintegrasi untuk mendapatkan bobot atau parameter yang optimum sehingga menghasilkan output sesuai target yang ditentukan dengan meminimumkan *error*. Pada pemodelan *Feed Forward Neural Network* (FFNN) untuk data *time series*, input model adalah data masa lalu ( $X_{t-1}, X_{t-2}, \dots, X_{t-p}$ ) dan targetnya adalah data masa sekarang ( $X_t$ ). Bentuk umum model FFNN untuk data *time series* dituliskan dalam persamaan berikut

$$X_t = \psi_0 \left\{ v_{bo} + \sum_{n=1}^H v_{no} \psi_n \left( w_{bn} + \sum_{i=1}^p w_{in} X_{t-i} \right) \right\}$$

dengan  $(w_{bn}, w_{in}, v_{bo}, v_{no})$  adalah nilai bobot-bobot atau parameter-parameter pada FFNN sedangkan  $(\psi_n, \psi_0)$  adalah fungsi aktivasi yang digunakan pada setiap unit pemroses.

Arsitektur jaringan yang akan digunakan dalam model FFNN adalah jaringan multilapis (*multilayer*) yang terdiri dari lapisan input (*input layer*), lapisan tersembunyi (*hidden layer*) dan lapisan output (*output layer*). Dalam penentuan banyaknya unit pada lapisan input tidak terdapat ketentuan yang baku. Begitu juga dengan banyaknya unit pada lapisan tersembunyi dan lapisan output. Oleh karena dalam penulisan tugas akhir ini masalah dibatasi pada jumlah unit *hidden layer* sama dengan jumlah unit pada *input layer*, maka arsitektur jaringan yang terbentuk terdiri dari 4 unit input variabel yang dianggap berpengaruh, 1 *hidden layer* yang terdiri dari 4 neuron, 1 neuron pada *output layer* dan bias. Sedangkan fungsi aktivasi yang digunakan pada *hidden layer* menuju ke *output layer* adalah sigmoid biner (sigmoid logistik) dan fungsi aktivasi yang digunakan untuk sinyal output adalah fungsi idensitas (purelin). Berdasarkan arsitektur jaringan FFNN yang telah terbentuk maka jumlah bobot atau parameter yang akan diestimasi dengan menggunakan AG sebanyak 25 unit yang terdiri dari 16 bobot neuron untuk memberikan sinyal input pada *hidden layer* ( $w_{in}$ ), 4 bobot bias untuk *hidden layer* ( $w_{bn}$ ), 4 bobot neuron untuk menghasilkan *output layer* ( $v_{no}$ ) dan 1 bobot bias untuk *output layer* ( $v_{bo}$ ).

Proses pelatihan dan pengujian pada data harian kurs Dolar Australia (AUD) terhadap Rupiah dilakukan menggunakan AG dengan metode seleksi turnamen untuk  $k$  (ukuran turnamen) yang berbeda-beda. Pemilihan operator genetika dalam hal ini probabilitas kawin silang ( $p_c$ ) sangat penting dalam menentukan performansi AG sehingga nantinya bobot atau parameter yang dihasilkan telah mencapai optimum global. Proses pelatihan dihentikan jika nilai *fitness* telah mencapai nilai yang maksimum atau generasi maksimum yang telah ditentukan. Berikut adalah hasil pelatihan dan pengujian dengan AG untuk beberapa ukuran turnamen " $k$ " dan " $p_c$ ". Ukuran populasi yang ditetapkan sebesar 50 kromosom disertai kawin silang menggunakan penyilangan satu titik potong (*single point crossover*) dan mutasi dengan *default options* pada Matlab yaitu dengan probabilitas mutasi ( $p_m$ ) 0.01. Setelah mencapai 25000 generasi diperoleh hasil sebagai berikut:

**Tabel 1 Metode Seleksi Turnamen dengan  $k = 4$**

No	$p_c$	Pelatihan		Pengujian	
		MSE	RMSE	MSE	RMSE
1	0.9	1.5361e+004	123.9389	7.1756e+003	84.7086
2	0.75	1.5046e+004	122.6634	1.0031e+004	100.1554
<b>3</b>	<b>0.5</b>	<b>1.4283e+004</b>	<b>119.5103</b>	<b>7.1586e+003</b>	<b>84.1203</b>
4	0.35	1.4351e+004	119.7973	1.1828e+004	108.7585

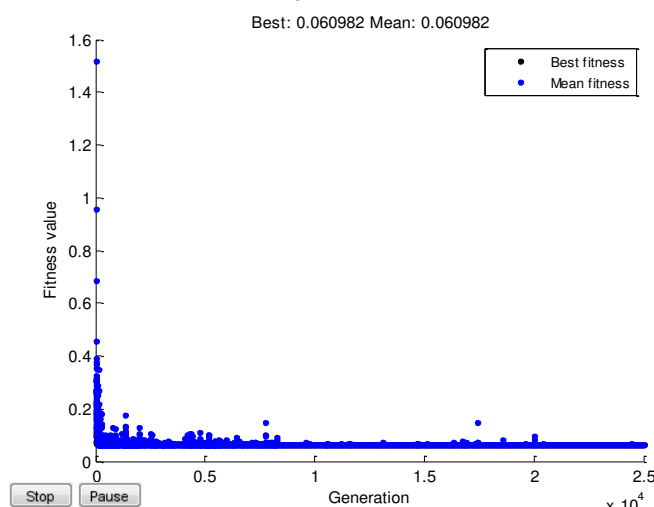
**Tabel 2 Metode Seleksi Turnamen dengan  $k = 5$**

No	$p_c$	Pelatihan		Pengujian	
		MSE	RMSE	MSE	RMSE
1	0.9	1.5299e+004	123.6904	9.8058e+003	99.0244
<b>2</b>	<b>0.75</b>	<b>1.4771e+004</b>	<b>121.5369</b>	<b>6.3692e+003</b>	<b>79.8072</b>
3	0.5	1.5096e+004	122.8672	7.6379e+003	87.3952
4	0.35	1.3853e+004	117.6985	2.5309e+005	503.0768

**Tabel 3 Metode Seleksi Turnamen dengan  $k = 6$**

No	$p_c$	Pelatihan		Pengujian	
		MSE	RMSE	MSE	RMSE
1	0.9	1.4775e+004	121.5519	6.8708e+003	82.8902
2	0.75	1.4184e+004	119.0973	9.3526e+003	96.7090
<b>3</b>	<b>0.5</b>	<b>1.3773e+004</b>	<b>117.3599</b>	<b>6.8308e+003</b>	<b>82.4917</b>
4	0.35	1.4005e+004	118.3417	7.6535e+003	87.4845

Untuk memilih model yang terbaik adalah dengan melihat nilai RMSE terkecil. Berdasarkan perbandingan nilai RMSE hasil pelatihan dan pengujian dengan menggunakan metode seleksi turnamen untuk  $k$  yang berbeda-beda maka model terbaik yang digunakan adalah metode seleksi turnamen dengan  $k=6$  dan  $p_c = 0.5$ .



**Gambar 1 Nilai *fitness* yang telah mencapai optimum global**

Berdasarkan gambar 1 terlihat bahwa proses AG dihentikan setelah mencapai generasi ke 25000. Selain itu nilai *fitness* yang dihasilkan telah konvergen dan mencapai optimum global yaitu dengan nilai *fitness* terbaik sebesar 0.060982 dan rata-rata nilai *fitness* sebesar 0.060982. Sedangkan nilai bobot atau parameter optimum yang diperoleh sebagai berikut:

**Tabel 4 Bobot atau parameter optimum hasil pelatihan**

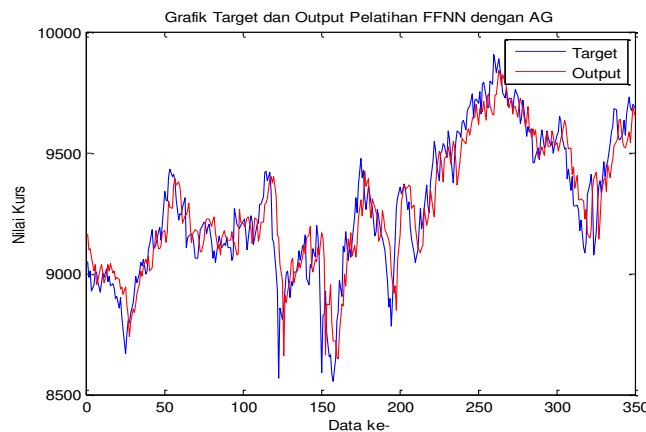
$w_{bn}$	$w_{in}$				$v_{bo}$	$v_{no}$
1.0000	1.0000	-0.0250	-0.0553	9.3558	0.7864	-1.9276
-0.9999	1.0000	0.2780	0.2869	0.7155		1.4292
-0.9999	0.1388	1.0000	1.0000	-3.3384		0.5311
-0.9997	-0.9998	1.0000	1.0000	-5.3640		0.6129

dimana  $w_{bn}$  adalah bobot bias dari *input layer* menuju neuron ke- $n$  pada *hidden layer*,  $w_{in}$  adalah bobot dari neuron ke- $i$  pada *input layer* menuju neuron ke- $n$  pada *hidden layer*,

$v_{bo}$  adalah bobot bias dari *hidden layer* menuju output, dan  $v_{no}$  adalah bobot dari neuron ke-n pada *hidden layer* menuju output. Dari bobot atau parameter hasil pelatihan yang telah optimum maka model FFNN untuk data *time series* dapat dituliskan dalam bentuk persamaan sebagai berikut:

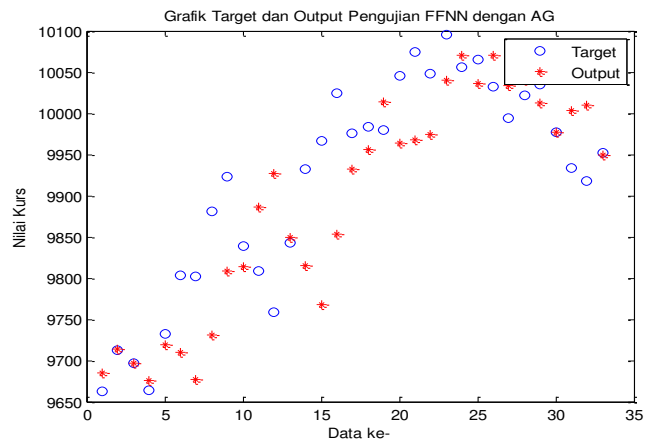
$$\hat{X}_t = 0.7864 + \frac{-1.9276}{1 + \exp\left(-\frac{1.0000 + 1.0000X_{t-3} + 1.0000X_{t-12} + 0.1388X_{t-14} - 0.9998X_{t-27}}{1.4292}\right)} + \frac{1.4292}{1 + \exp\left(-\frac{-0.9999 - 0.0250X_{t-3} + 0.2780X_{t-12} + 1.0000X_{t-14} + 1.0000X_{t-27}}{0.5311}\right)} + \frac{0.5311}{1 + \exp\left(-\frac{-0.9999 - 0.0553X_{t-3} + 0.2869X_{t-12} + 1.0000X_{t-14} + 1.0000X_{t-27}}{0.6129}\right)} + \frac{0.6129}{1 + \exp\left(-\frac{-0.9997 + 9.3558X_{t-3} + 0.7155X_{t-12} - 3.3384X_{t-14} - 5.3640X_{t-27}}{1.4292}\right)}$$

Berdasarkan persamaan FFNN tersebut maka diperoleh hasil pelatihan jaringan sebagai berikut:



**Gambar 2 Perbandingan target dan output pelatihan FFNN dengan AG**

Berdasarkan gambar perbandingan target dan output pelatihan FFNN dengan AG terlihat bahwa pelatihan jaringan telah memberikan hasil prediksi yang cukup akurat yang ditunjukkan oleh kedekatan target (garis berwarna biru) dengan output (garis berwarna merah). Namun, meskipun demikian masih diperlukan evaluasi kinerja jaringan secara umum dengan melihat hasil pengujian dari jaringan FFNN dengan AG. Sedangkan untuk hasil pengujian jaringan diperoleh sebagai berikut:



**Gambar 3 Perbandingan target dan output pelatihan FFNN dengan AG**

Pengujian dilakukan dengan menggunakan bobot atau parameter yang sudah optimum hasil dari tahap pelatihan. Berdasarkan gambar perbandingan target dan output pelatihan FFNN dengan AG terlihat bahwa pengujian jaringan telah memberikan hasil prediksi yang cukup akurat yang ditunjukkan oleh kedekatan target (titik berwarna biru) dengan output (titik berwarna merah). Model tersebut dapat digunakan dengan baik untuk memprediksi laju pergerakan kurs harian AUD terhadap Rupiah.

## 5. KESIMPULAN

Algoritma Genetika (AG) merupakan salah satu alternatif metode pembelajaran yang dapat digunakan dengan baik untuk melatih jaringan *Feed Forward Neural Network* pada data harian nilai kurs Dolar Australia (AUD) terhadap Rupiah. Hal ini dibuktikan dengan *error* yang dihasilkan pada hasil pelatihan dan pengujian memiliki nilai yang minimum yaitu dengan nilai RMSE sebesar 117.3599 dan 82.4917. Selain itu, analisis visual menunjukkan bahwa AG memberikan hasil prediksi yang cukup akurat yang ditunjukkan oleh kedekatan target dengan output. Sedangkan bobot atau parameter yang dihasilkan telah mencapai optimum global pada generasi ke-25000 dengan nilai *fitness* sebesar 0.060982.

## 6. DAFTAR PUSTAKA

- Fadlisyah, dkk. 2009. *Algoritma Genetik*. Graha Ilmu. Yogyakarta.
- Fausett, L. 1994. *Fundamental of Neural Network : Architecture, Algorithm and Application*. New Jersey. Prentice-Hall.
- Kusumadewi, S. 2003. *Artificial Intelligence (Teknik dan Aplikasinya)*. Graha Ilmu. Yogyakarta.
- Makridakis, dkk. 1999. *Metodedan Aplikasi Peramalan Edisi Kedua*. (diterjemahkan oleh : Suminto, Hari). Binarupa Aksara. Jakarta. Terjemahan dari : *Forecasting Methods and Applications, Second Edition*.
- Neves, J, and Cortez, P. 1998. *Combining Genetic Algorithms, Neural Networks and Data Filtering for Time Series Forecasting*. Departamento de Informatica Universidade do Minho. Portugal.
- Pandjaitan, L.W. 2002. *Dasar-Dasar Komputasi Cerdas*. Andi Offset. Yogyakarta.
- Siang, J.J. 2005. *Jaringan Syaraf Tiruan dan Pemrogramannya Menggunakan MATLAB*. Andi Offset. Yogyakarta.
- Suyanto. 2005. *Algoritma Genetika dalam MATLAB*. Andi Offset. Yogyakarta.
- Warsito, B. 2009. *Kapita Selektta Statistika Neural Network*. BP Universitas Diponegoro. Semarang.