

## PEMODELAN VOLATILITAS UNTUK PENGHITUNGAN *VALUE AT RISK* (VaR) MENGUNAKAN *FEED FORWARD NEURAL NETWORK* DAN ALGORITMA GENETIKA

Hasbi Yasin<sup>1</sup>, Suparti<sup>2</sup>

<sup>1</sup>Dosen Jurusan Statistika FSM UNDIP, [hasbiyasin@undip.ac.id](mailto:hasbiyasin@undip.ac.id)

<sup>2</sup>Dosen Jurusan Statistika FSM UNDIP, [suparti@undip.ac.id](mailto:suparti@undip.ac.id)

### Abstract

High fluctuations in stock returns is one problem that is considered by the investors. Therefore we need a model that is able to predict accurately the volatility of stock returns. One model that can be used is a model Generalized Autoregressive Conditional Heteroskedasticity (GARCH). This model can serve as a model input in the model Feed Forward Neural Network (FFNN) with Genetic Algorithms as a training algorithm, known as GA-Neuro-GARCH. This modeling is one of the alternatives in modeling the volatility of stock returns. This method is able to show a good performance in modeling the volatility of stock returns. The purpose of this study was to determine the stock return volatility models using a model GA-Neuro-GARCH on stock price data of PT. Indofood Sukses Makmur Tbk. The result shows that the determination of the input variables based on the ARIMA (1,0,1) -GARCH (1,1), so that the model used FFNN consists of 2 units of neurons in the input layer, 5 units of neurons in the hidden layer neuron layer and 1 unit in the output layer. then using a genetic algorithm with crossover probability value of 0.4, was obtained that the Mean Absolute Percentage Error (MAPE) of 0,0039%.

**Keywords:** FFNN, Genetic Algorithm, GARCH, Volatility

### 1. Pendahuluan

Secara statistik volatilitas merupakan standar deviasi dari pengembalian harga (return) saham. Besaran ini merepresentasikan risiko pengembalian harga saham tersebut<sup>[2],[14]</sup>. Semakin tinggi volatilitas, semakin besar resiko mengalami keuntungan atau kerugian<sup>[6],[13]</sup>. Ketidakpastian nilai volatilitas dalam pasar keuangan menyebabkan dibutuhkannya suatu alat untuk meramalkannya. Sedangkan *Value at Risk* (VaR) merupakan sebuah konsep yang digunakan untuk pengukuran risiko dalam manajemen risiko<sup>[11]</sup>. Secara sederhana VaR dapat diartikan sebagai seberapa besar investor dapat merugi selama periode investasi tertentu. Dalam penghitungan VaR, masalah utama yang harus dipecahkan adalah menentukan prediksi volatilitas return saham secara akurat yang akan dijadikan dasar penghitungan VaR.

Menurut Jorion<sup>[7]</sup>, data return saham biasanya memiliki variansi yang tidak konstan di setiap titik waktunya. Kondisi data yang seperti ini disebut sebagai heteroskedastisitas bersyarat (*Conditional Heteroskedasticity*). Salah satu model *time series* finansial yang dapat mengakomodasi heteroskedastisitas adalah model *Autoregressive Conditional Heteroskedasticity* (ARCH) yang diperkenalkan oleh Engle<sup>[3]</sup>. Sedangkan model yang lebih fleksibel untuk pemodelan variansi yang tidak konstan adalah model *Generalized Autoregressive Conditional Heteroskedasticity* (GARCH) yang diusulkan oleh Bollerslev<sup>[1]</sup>. Struktur GARCH terdiri dari dua persamaan, yaitu persamaan mean bersyarat yang merupakan model standar ARCH dan persamaan variansi bersyarat yang memungkinkan variansi berubah setiap waktu<sup>[15]</sup>. Model ini akan kurang optimal bila digunakan untuk prediksi volatilitas return saham. Salah satu metode peramalan yang

berkembang saat ini adalah dengan menggunakan *Artificial Neural Network* (ANN). ANN memiliki kemampuan untuk mempelajari dan mengadaptasi situasi baru dengan mengingat pola data masalah bahkan ketika terjadi gangguan (*noise*). Penerapan model ANN khususnya model *Feed Forward Neural Network* (FFNN) untuk peramalan *time series* bertujuan untuk mengatasi keterbatasan pada metode peramalan konvensional terutama bila digunakan pada data finansial dalam hal ini adalah data harga saham.

Penggunaan FFNN 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 FFNN dan AG untuk peramalan *time series* memberikan keuntungan dari kedua metode tersebut. Studi kasus lainnya menunjukkan bahwa ANN dan AG memberikan hasil yang baik dan lebih akurat untuk peramalan *time series* dibandingkan dengan metode konvensional<sup>[9]</sup>. Oleh karena itu, dalam penelitian ini akan dikembangkan model alternatif yang menggabungkan model GARCH dengan model FFNN dengan algoritma pelatihan yang digunakan adalah Algoritma Genetika (GA-Neuro-GARCH). Model ini akan digunakan dalam memodelkan nilai volatilitas saham PT. Indofood Sukses Makmur Tbk, yang selanjutnya digunakan untuk menghitung *Value at Risk* (VaR).

## 2. Tinjauan Pustaka

### 2.1 Model *Generalized Autoregressive Conditional Heteroskedasticity* (GARCH)

Secara umum model ARCH berorde  $q$  digunakan untuk membentuk model varians bersyarat ( $\sigma_t^2$ ) pada waktu ke- $t$  berdasarkan nilai kuadrat error pada waktu ke  $t-1$  hingga  $t-q$ . Misalkan model rata-rata adalah:

$$Z_t = \mu_t + e_t$$

Menurut Tsay<sup>[14]</sup> bahwa  $\mu_t$  merupakan nilai harapan  $Z_t$  bersyarat  $F_{t-1}$ , dengan  $F_{t-1} = \{Z_{t-1}, Z_{t-2}, Z_{t-3}, \dots, Z_2, Z_1\}$ . Sehingga model ARMA( $r, m$ ) dari  $Z_t$  adalah:

$$\begin{aligned} \mu_t &= E(Z_t | F_{t-1}) \\ &= \theta_0 + \sum_{i=1}^r \phi_i Z_{t-i} + \sum_{j=1}^m \theta_j e_{t-j} \end{aligned}$$

dengan:

$X_t$  = return pada waktu ke- $t$

$F_{t-1}$  = seluruh himpunan informasi pada waktu ke-1 sampai ke-  $t-1$

$\mu_t$  = nilai harapan  $X_t$  bersyarat  $F_{t-1}$

$e_t$  = residual ARMA pada waktu ke- $t$

Tsay<sup>[14]</sup> menyatakan bahwa model ARCH adalah sisaan  $e_t$  dari model ARIMA yang pada orde tinggi akan saling berkorelasi,  $e_t$  dapat diuraikan sebagai berikut:

$$\begin{aligned} e_t &= \varepsilon_t \sigma_t & e_t | F_{t-1} &\sim iidN(0, \sigma_t^2) \\ & & \varepsilon_t &\sim iidN(0, 1) \end{aligned}$$

Diperoleh varians bersyarat bagi  $e_t$  :

$$\begin{aligned} \text{Var}(e_t | F_{t-1}) &= E(e_t^2 | F_{t-1}) \\ &= E(\varepsilon_t^2 \sigma_t^2 | F_{t-1}) \\ &= \sigma_t^2 E(\varepsilon_t^2 | F_{t-1}) \\ &= \sigma_t^2 \end{aligned}$$

sehingga ragam bersyarat yang mendefinisikan model ARCH dengan orde  $q$  yaitu :

$$\sigma_t^2 = \alpha_0 + \sum_{i=1}^q \alpha_i e_{t-i}^2$$

dengan  $q > 0$ ,  $\alpha_0 > 0$ , dan  $\alpha_i \geq 0$  untuk  $i = 1, 2, 3, \dots, q$ .

Model ARCH yang digeneralisasi dikenal dengan *Generalized Autoregressive Conditional Heteroskedasticity* (GARCH). Model GARCH digunakan untuk mengatasi orde yang terlalu besar pada model ARCH. Bentuk umum model GARCH( $p, q$ ) menurut Tsay<sup>[14]</sup>:

$$\sigma_t^2 = \omega + \sum_{j=1}^p \beta_j \sigma_{t-p}^2 + \sum_{i=1}^q \alpha_i e_{t-i}^2$$

Dalam model GARCH( $p, q$ ), proses  $e_t$  dapat didefinisikan dengan  $e_t = \varepsilon_t \sigma_t$

$\sigma_t$  adalah akar dari  $\sigma_t^2$  dan  $\varepsilon_t$  adalah proses *i.i.d* (*independent and identically distributed*), seringkali diasumsikan berdistribusi normal standar  $N(0,1)$ .

Koefisien-koefisien dari model GARCH( $p, q$ ) bersifat:

- (1)  $\omega > 0$
- (2)  $\alpha_i \geq 0$  untuk  $i = 1, 2, \dots, q$
- (3)  $\beta_j \geq 0$  untuk  $j = 1, 2, \dots, p$
- (4)  $\sum_{i=1}^q \sum_{j=1}^p (\alpha_i + \beta_j) < 1$

Kondisi (4) diperlukan agar model bersifat stasioner, sedangkan kondisi (1), (2), dan (3) yang diperlukan agar  $\sigma_t^2 > 0$ <sup>[12]</sup>.

## 2.2 Algoritma Genetika

Fadlisyah<sup>[4]</sup> menerangkan bahwa Algoritma Genetika (AG) bekerja dari populasi yang merupakan himpunan solusi yang dihasilkan secara acak. Setiap anggota himpunan yang merepresentasikan satu solusi masalah dinamakan individu atau kromosom. 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 yang disebut *fitness*. Pada AG, *fitness* biasanya dapat berupa fungsi objektif dari masalah yang akan dioptimalisasi.

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<sup>[8]</sup>.

Terdapat dua operator genetika yaitu *crossover* dan mutasi. *Crossover* bertujuan menambah keanekaragaman string dalam satu populasi dengan penyilangan antar string yang diperoleh dari reproduksi sebelumnya. Proses *crossover* dilakukan pada setiap individu dengan probabilitas *crossover* ( $p_c$ ) yang ditentukan secara random dalam rentang [0, 1]. Artinya, kawin silang bisa dilakukan hanya jika suatu bilangan random [0, 1] yang dibangkitkan kurang dari  $p_c$  yang ditentukan. Sedangkan, mutasi merupakan proses untuk mengubah nilai dari satu atau beberapa gen dalam suatu kromosom. Proses mutasi terjadi jika bilangan random yang dibangkitkan kurang dari probabilitas mutasi  $p_m$  yang ditentukan maka ubah gen tersebut menjadi nilai kebalikannya (dalam *binary encoding*, 0 diubah 1 dan 1 diubah 0).

### 2.3 Aplikasi Algoritma Genetika untuk Pelatihan FFNN

Penggunaan Algoritma Genetika (AG) sebagai metode pembelajaran pada model FFNN 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 yang digunakan 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:

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

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<sup>[9]</sup>.

### 2.4 Model GA-Neuro-GARCH

GA-Neuro-GARCH merupakan teknik kombinasi antara FFNN dengan GARCH, di mana model GARCH digunakan sebagai model awal untuk penentuan variabel input pada model FFNN dan algoritma pelatihan yang digunakan adalah metode algoritma genetika (AG). Dalam memodelkan GA-Neuro-GARCH sejumlah data *return*  $Y_t$  pada waktu  $t_1, t_2, t_3, \dots, t_n$  selanjutnya dipakai untuk memperkirakan nilai *return* pada waktu  $t_{n+1}$ . Salah satu hal penting dalam model GA-Neuro-GARCH adalah menentukan variabel inputnya. Sebagai contoh untuk menentukan input dan target dari model GARCH (1,1). Misalkan model GARCH (1,1)  $\sigma_t^2 = \omega + \alpha_1 e_{t-1}^2 + \beta_1 \sigma_{t-1}^2$ , maka input yang dipakai adalah  $e_{t-1}^2$  dan  $\sigma_{t-1}^2$  dengan target adalah  $\sigma_t^2$ . Sehingga model dapat ditulis  $\sigma_t^2 = f(\sigma_{t-1}^2, e_{t-1}^2)$ .

### 2.5 Value at Risk (VaR)

Nilai Value at Risk (VaR) untuk *return* asset tunggal PT. Indofood Sukses Makmur, Tbk dengan tingkat kepercayaan  $(1-\alpha)$  dan *holding period* ( $hp$ ), dapat dihitung menggunakan rumus:

$$VaR(1 - \alpha, hp) = -Z_{1-\alpha} * S_0 * \sqrt{\sigma_t^2 * hp}$$

dengan:

$S_0$  = nilai investasi awal

$\sigma_t^2$  = Volatilitas *return* saham PT. Indofood Sukses Makmur, Tbk pada waktu  $t$

### 3. Hasil dan Pembahasan

Pemodelan return saham PT. Indofood Sukses Makmur, Tbk. dilakukan dengan menggunakan model GARCH. Berdasarkan hasil pengolahan data menggunakan program GUI MATLAB tersebut diperoleh identifikasi model awal adalah model ARMA(3,3) GARCH(3,3). Tetapi untuk memperoleh model GARCH terbaik perlu dilakukan proses overfit dan underfit terhadap parameter model yang digunakan, dan diperoleh hasil seperti pada Tabel 1.

**Tabel 1.** Penentuan Model GARCH Terbaik untuk Return Saham PT. Indofood Sukses Makmur, Tbk.

NO	MODEL	AIC
1	ARMA(3,3) GARCH(3,3)	-2565,6574
2	ARMA(3,3) GARCH(1,3)	-2561,4930
3	ARMA(3,3) GARCH(1,1)	-2565,4930
4	ARMA(0,0) GARCH(1,1)	-2564,5232
5	ARMA(0,0) GARCH(1,2)	-2562,5230
<b>6</b>	<b>ARMA(1,1) GARCH(1,1)</b>	<b>-2567,8614</b>
7	ARMA(2,2) GARCH(1,1)	-2566,8801
8	ARMA(2,0) GARCH(1,1)	-2562,4037
9	ARMA(0,2) GARCH(1,1)	-2562,8395

Model terbaik untuk pemodelan return saham PT. Indofood Sukses Makmur, Tbk adalah model ARMA(1,1) GARCH(1,1) yang secara matematis dapat ditulis sebagai berikut:

$$Z_t = 2,6906 \times 10^{-5} + 0,77956Z_{t-1} - 0,860579e_{t-1} + e_t$$

$$\sigma_t^2 = 4,0618 \times 10^{-6} + 0,91152\sigma_{t-1}^2 + 0,079279e_{t-1}^2$$

#### 3.1 Pemodelan FFNN-AG Berdasarkan Model GARCH Terbaik

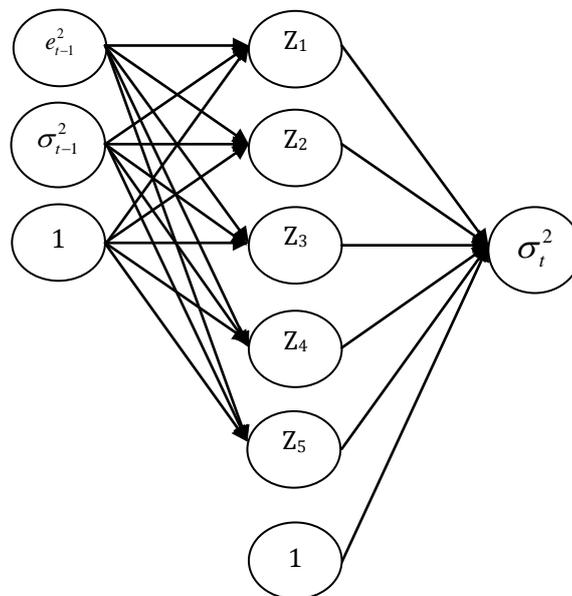
Model JST yang digunakan dalam penelitian ini adalah model FFNN dengan penentuan jumlah neuron pada lapisan *hidden layer* sesuai dengan ketentuan empiris menurut Fahmi<sup>[5]</sup>. Berdasarkan model GARCH yang terbentuk maka model untuk return saham PT. Indofood Sukses Makmur, Tbk adalah FFNN dengan 2 unit neuron pada lapisan input, 5 unit neuron pada lapisan *hidden layer* dan 1 unit neuron pada lapisan output. Sedangkan fungsi aktivasi yang digunakan pada *hidden layer* menuju ke *output layer* adalah sigmoid biner (*sigmoid logistic*) 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 21 unit yang terdiri dari 10 bobot neuron untuk memberikan sinyal input pada *hidden layer* ( $w_{in}$ ), 5 bobot bias untuk *hidden layer* ( $w_{bn}$ ), 5 bobot neuron untuk menghasilkan *output layer* ( $v_{no}$ ) dan 1 bobot bias untuk *output layer* ( $v_{bo}$ ). Model arsitektur jaringan FFNN seperti pada Gambar 1.

Tabel 3 merupakan hasil pelatihan dengan AG untuk beberapa ukuran " $p_c$ ". 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 5000 generasi diperoleh hasil seperti pada Tabel 3.

**Tabel 2.** Rumus Empiris Penentuan Banyaknya Neuron pada Hidden Layer

$N_o$ (Neuron output)	Rumus Empiris menghitung hidden layer
1	$2 * N_i + 1$
2	$3 * N_i$
3	$\frac{2 + N_i * N_o + 0.5 N_i * (N_o^2 + N_i) - 3}{N_i + N_o}$
4	$\frac{(2 * N_i)}{3}$
5	$\sqrt{N_o * N_i}$
6	$2 * N_i$

$N_i$  (Neuron Input)



**Gambar 1.** Model FFNN untuk Volatilitas Return Saham PT. Indofood Sukses Makmur, Tbk.

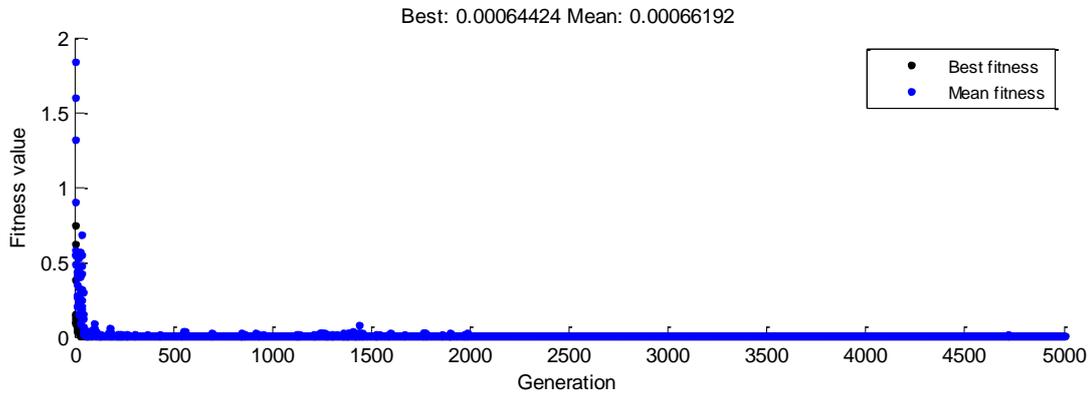
**Tabel 3.** Metode Seleksi Turnamen dengan  $k = 8$

No	$p_c$	Ukuran Kebaikan	
		RMSE	MAPE
1	0,2	$3,9627 \times 10^{-6}$	0,0093
2*)	0,4	$1,2811 \times 10^{-6}$	0,0039
3	0,6	$2,8443 \times 10^{-6}$	0,0049
4	0,8	$3,5953 \times 10^{-6}$	0,0095

\*) Model terbaik

Gambar 2 menunjukkan performansi pelatihan AG yang telah mencapai optimum global menggunakan metode seleksi turnamen dengan  $k = 8$  dan  $p_c = 0,4$ . Berdasarkan Gambar 2 terlihat bahwa proses AG dihentikan setelah mencapai generasi ke 5000. Selain

itu nilai *fitness* yang dihasilkan telah konvergen dan mencapai optimum global yaitu dengan nilai *fitness* terbaik sebesar 0,00064424 dan rata-rata nilai *fitness* sebesar 0,00066192. Pada pelatihan ini diperoleh parameter optimal global pada Tabel 4.



**Gambar 2.** Nilai *Fitness* yang Telah Mencapai Optimum Global

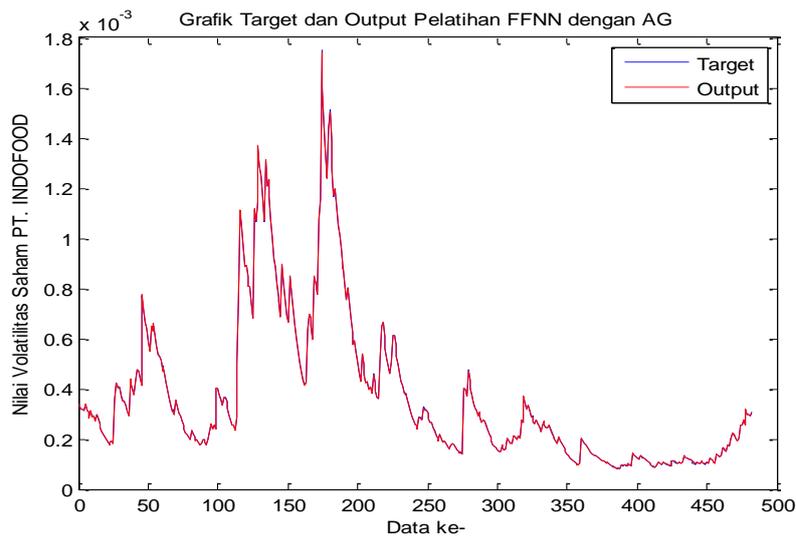
**Tabel 4.** Bobot atau Parameter Optimal Global Hasil Pelatihan

$w_{bn}$	$w_{1n}$	$w_{2n}$	$v_{bo}$	$v_{no}$
0,6808	0,2147	0,8900	0,0110	1,3823
-0,7221	0,4818	1,4876		1,5245
-0,7092	-0,2196	-0,1117		0,5632
0,0359	-0,0657	0,6539		-0,1498
1,0000	0,4778	-0,5409		-2,0966

Berdasarkan Tabel 4 maka model FFNN untuk volatilitas return saham harian PT. Indofood Sukses Makmur, Tbk dapat dituliskan sebagai berikut :

$$\hat{x}_t = 0,0110 + \frac{1,3823}{1 + \exp(-(0,6808 + 0,2147e_{t-1}^2 + 0,8900\sigma_{t-1}^2))} + \frac{1,5245}{1 + \exp(-(-0,7221 + 0,4818e_{t-1}^2 + 1,4876\sigma_{t-1}^2))} + \frac{0,5632}{1 + \exp(-(-0,7092 - 0,2196e_{t-1}^2 - 0,1117\sigma_{t-1}^2))} + \frac{-0,1498}{1 + \exp(-(0,0359 - 0,0657e_{t-1}^2 + 0,6539\sigma_{t-1}^2))} + \frac{-2,0966}{1 + \exp(-(1,0000 + 0,4778e_{t-1}^2 - 0,5409\sigma_{t-1}^2))}$$

Berdasarkan persamaan FFNN tersebut maka diperoleh hasil pelatihan jaringan pada Gambar 3.



**Gambar 3.** Perbandingan Target dan Output Pelatihan FFNN dengan AG

### 3.2 Penghitungan VaR menggunakan Model Terbaik

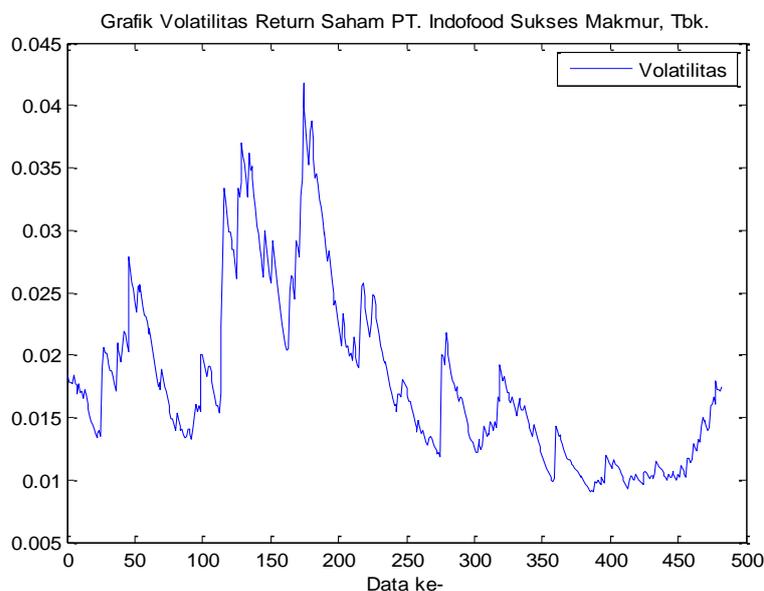
Nilai Value at Risk (VaR) untuk return asset tunggal PT. Indofood Sukses Makmur Tbk dengan tingkat kepercayaan  $(1-\alpha)$  dan *holding period* ( $hp$ ) dapat dihitung menggunakan rumus:

$$VaR(1 - \alpha, hp) = -Z_{1-\alpha} * S_0 * \sqrt{\sigma_{INDF}^2 * hp}$$

dengan:  $S_0$  = nilai investasi awal

$\sigma_{INDF}^2$  = Volatilitas return saham PT. Indofood Sukses Makmur, Tbk.

Nilai VaR return saham PT. Indofood Sukses Makmur, Tbk dengan tingkat kepercayaan 95% dan *holding period* 1 hari adalah  $VaR(95\%, 1) = -1,645 * S_0 \sqrt{\sigma_{INDF}^2}$   
 Hasil estimasi volatilitas untuk data *in sample* ditunjukkan Gambar 4.



**Gambar 4.** Estimasi Volatilitas Return Saham PT. Indofood Sukses Makmur, Tbk

#### 4. Kesimpulan

Estimasi model input yang digunakan untuk memprediksikan volatilitas return saham PT. Indofood Sukses Makmur, Tbk adalah model ARIMA (1,0,1)-GARCH(1,1). Berdasarkan model tersebut dapat disusun pemodelan FFNN dengan 2 unit neuron pada lapisan input, 5 unit neuron pada lapisan *hidden layer* dan 1 unit neuron pada lapisan output. Pengoptimalan parameter dapat dilakukan dengan Algoritma Genetika. Pemodelan GARCH dan FFNN dengan algoritma pelatihan Algoritma Genetika (AG) merupakan salah satu alternatif dalam pemodelan volatilitas return saham. Metode ini mampu menunjukkan performa yang baik dalam memodelkan volatilitas return saham dengan MAPE sebesar 0,0039%.

#### DAFTAR PUSTAKA

1. Bollerslev, T., Generalized Autoregressive Conditional Heteroscedasticity, *Journal of Econometrics*, 1986, **31**: 307-327
2. Engle, R.F., Autoregressive Conditional Heteroscedasticity with Estimates of Variance United Kingdom Inflation, *Econometrica*, 1982, Vol. 50, No.4: 987-1007.
3. Engle, R.F. and S. Manganelli, *Value at Risk Models in Finance*, Working Paper Series No. 75 August 2001 European Central Bank. Germany.
4. Fadlisyah, A. dan Faisal, *Algoritma Genetik*, Graha Ilmu, Yogyakarta, 2009.
5. Fahmi, Y., *Peramalan Nilai Harga Saham Menggunakan Jaringan Syaraf Tiruan dan Algoritma Genetik*, UII Yogyakarta, 2011.
6. Holton, G., *Value at Risk, Theory and Practice*, Academic Press, Boston, 2003.
7. Jorion, P., *Value at Risk: The New Benchmarking for Managing Financial Risk*. Mc Graw Hill, 2002.
8. Kusumadewi, S., *Artificial Intelligence (Teknik dan Aplikasinya)*, Graha Ilmu, Yogyakarta, 2003.
9. Kusumadewi, S., *Membangun Jaringan Syaraf Tiruan Menggunakan Matlab dan Excel Link*, Graha Ilmu, Yogyakarta, 2004.
10. Neves, J, and Cortez, P., *Combining Genetic Algorithms, Neural Networks and Data Filtering for Time Series Forecasting*, Departamento de Informatica Universidade do Minho, Portugal, 1998.
11. Morgan J.P., *Risk Metrics – Technical Document*, J. P. Morgan Global Research Foueth Edition, Reuters, 1996.
12. Rosadi, D., *Ekonometrika & Analisis Runtun Waktu Terapan dengan EViews*, Andi Offset, Yogyakarta, 2011.
13. Situngkir, H. dan Surya, Y., *Value at Risk yang Memperhatikan Sifat Statistika Distribusi Return*, Working Paper WDP 2006, Bandung Fe Institute.
14. Tsay, *Analysis of Financial Time Series, second Edition*, Wiley Interscience, A John Wiley and Sons. Inc. Publication, USA, 2005.
15. Wei, W.W.S., *Time Series Analysis: Univariate and Multivariate Methods*, Second Edition, Pearson Education Inc., Boston, 2006.