

# Optimasi *Fuzzy Artificial Neural Network* dengan Algoritma Genetika untuk Prediksi Harga *Crude Palm Oil*

<http://dx.doi.org/10.28932/jutisi.v6i2.2617>

Anwar Rifa'i ✉ #1

# Jurusan Teknik Informatika, Universitas Budiluhur,  
Jl. Ciledug Raya, Petungkang Utara, Pesanggrahan, Jakarta Selatan  
1anwar.rifai@budiluhur.ac.id

**Abstract**—Crude Palm Oil (CPO) is one of Indonesia's best export commodities. CPO production competition causes price fluctuations so that it can trigger losses. The solution that can be taken to avoid losses is to predict the price of CPO. Time series data in the previous months, starting from January 2009 until January 2020, are used as a reference to predict the next CPO price. In this research, CPO price prediction is carried out with a combination of artificial intelligence concepts, namely Radial Basis Function Neural Network (RBFNN), and fuzzy logic. The combination of these methods, namely Fuzzy Radial Basis Function Neural Network (FRBFNN), is then optimized using genetic algorithms. The prediction results show that the error based on the MAPE value for FRBFNN prediction on training data is 11.7% and the MAPE value for testing data is 9.4%. In the FRBFNN prediction that was optimized using a genetic algorithm, the MAPE value was 10.2% for training data and 8.3% for testing data.

**Keywords**—Algorithm; CPO; Fuzzy; Genetic; Prediction; RBFNN

## I. PENDAHULUAN

Indonesia merupakan produsen minyak kelapa sawit atau *crude palm oil* (CPO) terbesar di dunia [1]. Minyak kelapa sawit banyak digunakan untuk berbagai keperluan seperti untuk pelumas, memasak, bahan pembuatan sabun, dan berbagai material barang lainnya. Di Indonesia CPO menjadi kontributor yang memicu perkembangan investasi, teknologi, dan manajemen pengetahuan [2]. Produksi CPO merupakan aspek yang penting bagi kegiatan perekonomian di Indonesia. Bidang ini menjadi salah satu pencipta lapangan pekerjaan untuk banyak orang. Dalam waktu dekat CPO dipertimbangkan sebagai orientasi perekonomian di dunia karena minyak kelapa sawit merupakan minyak yang paling banyak digunakan di dunia [3].

Persaingan produksi CPO kini semakin ketat. Banyak negara yang mulai memproduksi CPO dalam jumlah yang

besar. Akibatnya, indeks harga kelapa sawit menjadi tidak stabil karena persaingan ini. Fluktuasi harga CPO menimbulkan risiko yang besar bagi para petani kelapa sawit dan investor. Keadaan ini juga akan berdampak pada perekonomian nasional. Manajemen risiko perlu dilakukan untuk menghindari kerugian dalam jumlah besar. Salah satu pendukung dalam manajemen risiko adalah melakukan prediksi terhadap harga CPO. Prediksi dapat dilakukan dengan memanfaatkan data harga minyak kelapa sawit pada waktu lampau.

Beberapa penelitian telah dilakukan untuk memprediksi harga CPO. Penelitian [4] yang memprediksi harga CPO menggunakan SARIMA menunjukkan bahwa metode ini mampu memberikan prediksi yang baik. Di sisi lain, penelitian [5] yang mengimplementasikan metode *Artificial Intelligence Neural Network* menunjukkan bahwa metode *artificial Intelligent* mampu memprediksi secara lebih baik dibandingkan dengan metode statistika biasa termasuk SARIMA.

*Neural Network* merupakan metode yang memiliki karakteristik seperti dengan jaringan syaraf manusia. Metode ini memiliki keunggulan dibandingkan dengan metode-metode prediksi lainnya [6]. Salah satu metode *Neural Network* yang memiliki kemampuan untuk memprediksi adalah *Radial Basis Function Neural Network* (RBFNN) [7]. Model ini terdiri atas lapisan *input*, lapisan tersembunyi, dan lapisan *output*. Ciri utama RBFNN adalah hanya memiliki lapisan tersembunyi tunggal, menggabungkan tipe pembelajaran terawasi (*supervised learning*) dan pembelajaran tak terawasi (*unsupervised learning*), serta hanya mempunyai bobot lapisan *output*.

Logika *fuzzy* merupakan salah satu konsep yang mampu menangani kejadian tidak pasti di dunia. *Fuzzy* akan memetakan nilai kepada derajat keanggotaan yang berada pada rentang 0 sampai 1 [8]. Konsep ini sering digunakan untuk melakukan pengambilan keputusan. Logika *Fuzzy*

dapat diintegrasikan dengan RBFNN menjadi *Fuzzy Radial Basis Function Neural Network* (FRBFNN). Kemampuan RBFNN dalam memberikan prediksi akan semakin akurat dengan bantuan konsep *fuzzy* [9].

Prediksi yang baik adalah prediksi dengan tingkat kesalahan (*error*) kecil. Tingkat kesalahan pada FRBFNN dapat diminimalisir dengan mengoptimalkan bobot pada sistem. Algoritma yang memiliki potensi untuk melakukan hal ini adalah algoritma genetika. Algoritma ini memiliki kemampuan untuk menentukan solusi terbaik dengan mengoptimasi bobot yang dihasilkan pada proses FRBFNN, sehingga *error* yang dihasilkan menjadi lebih kecil [10]. Pada beberapa kasus seperti peramalan mata uang, algoritma genetika telah terbukti mampu meminimalisir tingkat kesalahan sistem peramalan [11]. Oleh karena itu, pada penelitian ini akan dilakukan peramalan harga CPO menggunakan FRBFNN kemudian dioptimasi menggunakan algoritma genetika.

## II. TINJAUAN PUSTAKA

### A. Crude Palm Oil (CPO)

Kelapa sawit diperkenalkan di Asia Tenggara pada abad ke-20. Kelapa sawit atau *Elaeis guinensis* merupakan tanaman yang dapat menghasilkan minyak kelapa sawit atau Crude Palm Oil (CPO). Minyak ini memiliki banyak kegunaan seperti untuk pelumas, memasak, bahan baku pembuatan sabun, dan lain sebagainya. Saat ini CPO menjadi salah satu komoditas ekspor yang paling besar di Indonesia. Produksi CPO akan mempengaruhi kegiatan perekonomian suatu negara [12].

Banyak tenaga kerja yang bergantung pada produksi kelapa sawit. Di Indonesia CPO menjadi kontributor yang memicu perkembangan investasi, teknologi, dan manajemen pengetahuan [2]. Persaingan antar negara produsen CPO membuat harga CPO menjadi fluktuatif. Apabila perubahan harga CPO tidak dapat dimanajemen dengan baik oleh produsen dalam negeri hal ini akan menimbulkan kerugian yang cukup besar sehingga berpengaruh terhadap perekonomian di Indonesia.

### B. Logika Fuzzy

Logika *fuzzy* muncul dari ketidakmampuan logika *crisp* dalam mewakili pemikiran manusia. Perbedaan antara logika *fuzzy* dan logika *crisp* terdapat pada keanggotaan elemen pada suatu himpunan. Konsep dasar dari logika *fuzzy* terdiri atas himpunan *fuzzy*, fungsi keanggotaan, operator *fuzzy*, fuzzifikasi, dan defuzzifikasi.

Himpunan *crisp* memiliki keanggotaan 0 dan 1 saja, sedangkan pada himpunan *fuzzy* keanggotaan himpunan disajikan dalam bentuk bilangan real dalam selang  $[0, 1]$  [7]. Suatu himpunan *fuzzy*  $A$  dalam  $S$  dapat dinotasikan sebagai suatu pasangan berurut dari nilai  $x$  dengan derajat keanggotaan  $\mu_A(x)$ . Domain yang diakses dalam himpunan *fuzzy* merupakan seluruh nilai dalam semesta pembicaraan

dan dapat dioperasikan secara *fuzzy*. Notasi himpunan *fuzzy* disajikan dalam formula 1.

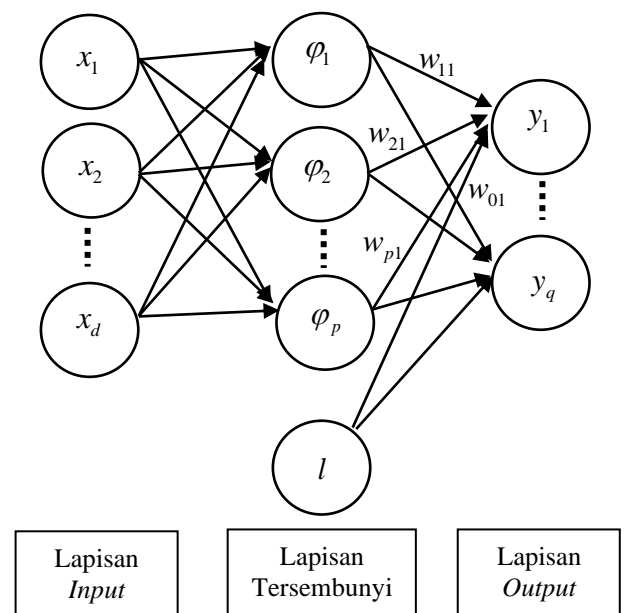
$$A = \{(x, \mu_A(x)) \mid x \in S\} \quad (1)$$

Fungsi keanggotaan yang digunakan pada suatu logika *fuzzy* terdiri atas beberapa bentuk yaitu representasi *linear*, kurva segitiga, *gaussian*, dan lain sebagainya [13]. Operator *fuzzy* terdiri atas beberapa operator yang difungsikan untuk melakukan perubahan atau kombinasi terhadap himpunan *fuzzy*. Tipe operator dasar *fuzzy* adalah AND, OR, dan NOT. Selanjutnya fuzzifikasi adalah proses untuk mentransformasikan nilai *input* menjadi derajat keanggotaan *fuzzy*. Hasil keanggotaan *fuzzy* ini akan diproses kemudian ditransformasikan kembali dalam bentuk *crisp* melalui proses defuzzifikasi.

### C. Radial Basis Function Neural Network (RBFNN)

*Radial Basis Function Neural Network* (RBFNN) merupakan model yang terdiri atas tiga lapisan yaitu *input*, *hidden*, dan *output*. Pada suatu proses, lapisan *input* akan menerima vektor untuk kemudian disalurkan pada lapisan tersembunyi. Pada lapisan ini vektor akan diolah secara *nonlinear* menggunakan suatu fungsi aktivasi. Hasil pemrosesan ini selanjutnya diteruskan dan diolah pada lapisan *output* secara *linear*.

Proses sistem RBFNN bergantung pada pemilihan empat parameter yaitu jarak maksimum, *cluster*, bobot pembelajaran, dan fungsi aktivasi. Arsitektur RBFNN disajikan pada gambar 1.



Gambar 1. Arsitektur RBFNN

Pada gambar 1 terdapat d vektor *input*  $x$ , fungsi aktivasi tersembunyi sebanyak  $p$  buah, dan 1 *neuron output*. Hasil

output merupakan kombinasi *linear* dari bobot  $w_{js}$  dengan fungsi aktivasi  $\varphi_j(x_t)$  dan bobot bias  $w_{0s}$ . Setelah melalui proses RBFNN *output* ke- $s$  data ke- $t$  ( $y_{ts}$ ) dihasilkan melalui rumus 2.

$$y_{ts} = \sum_{j=1}^p w_{js} \varphi_j(x_t) + w_{0s} \quad (2)$$

Fungsi aktivasi yang digunakan pada RBFNN adalah fungsi *gaussian* yang disajikan pada rumus 3. Fungsi *Gaussian* memiliki dua parameter utama yaitu  $c_{ji}$  yang menyatakan nilai pusat *cluster* ke- $j$  dari variabel input ke- $i$ , dan  $r_j$  yang menyatakan jarak maksimum data pusat *cluster* ke- $j$ .

$$\varphi_j(x_t) = \exp\left(-\sum_{i=1}^d \frac{(x_{ti} - c_{ji})^2}{r_j^2}\right) \quad (3)$$

Langkah pertama pada pembentukan sistem RBFNN adalah melakukan proses terhadap sinyal input menuju lapisan tersembunyi. Dalam proses ini ditentukan nilai pusat dan jarak yang digunakan pada lapisan tersembunyi.

Metode yang digunakan adalah *K-Means Clustering*. Metode ini merupakan suatu metode yang paling sederhana tanpa pengawasan algoritma pembelajaran [14]. *Euclidean distance* dihitung untuk menentukan jarak data ke pusat. Data akan digolongkan ke beberapa kelompok dengan jarak minimum. Persamaan *euclidean distance* di formulasikan pada rumus 4.

$$d_{eucl}(x_t, c_{ji}) = \sqrt{\sum_{i=1}^d (x_{ti} - c_{ji})^2} \quad (4)$$

Pada rumus 4,  $x_{ti}$  melambangkan nilai data ke- $t$  pada variabel *input* ke- $i$ , sedangkan  $c_{ji}$  menunjukkan nilai pusat *cluster* ke- $j$  dari variabel *input* ke- $i$ .

Langkah selanjutnya yang dilakukan adalah mengestimasi bobot. Estimasi dilakukan menggunakan metode *Global Ridge-Regression* yang menambahkan parameter regulasi dengan nilai positif pada *sum square error* (SSE). Formula SSE disajikan pada rumus 4.

$$SSE = \sum_{s=1}^q \sum_{t=1}^n (y_{ts} - \hat{y}_{ts})^2 \quad (5)$$

Untuk metode *global ridge-regression* diformulasikan pada rumus 5.

$$C = \sum_{s=1}^q \sum_{t=1}^n (y_{ts} - \hat{y}_{ts})^2 + \lambda \sum_{s=1}^q \sum_{j=0}^p w_{js}^2 \quad (6)$$

Pada rumus 5  $y_{ts}$  menyatakan nilai variabel *output* ke- $s$  pada data ke- $t$ ,  $\hat{y}_{ts}$  menyatakan peramalan variabel *output* ke- $s$  pada data ke- $t$ ,  $w_{js}$  merupakan bobot *neuron* lapisan tersembunyi, dan  $\lambda$  adalah parameter regulasi.

#### D. Algoritma Genetika

Algoritma genetika merupakan metode optimasi yang dibuat berdasarkan konsep evolusi [15]. Metode ini menggunakan teknik di mana populasi awal dijadikan

sebagai calon solusi dari masalah. Algoritma Genetika memiliki dua proses utama dalam pemecahan masalah yaitu mutasi dan seleksi. Suatu generasi akan melalui sebuah proses mutasi dan sebuah proses seleksi. Pada penelitian ini digunakan *Non Dominated Sorting Genetic Algorithm* (NSGA II). Algoritma ini digunakan untuk meningkatkan kemampuan adaptasi populasi kandidat yang menjadi solusi untuk suatu *front pareto* yang dibatasi aturan objektif [16]. Algoritma ini memakai proses evolusi pengganti dengan operator evolusi termasuk seleksi, *crossover* genetik, dan mutasi genetik.

Menurut Xiao [17] secara singkat algoritma genetika dapat dijalankan dengan langkah berikut.

- Langkah 1. Parameter himpunan  $P$  dimutasi menjadi  $n$  vektor sebagai populasi pertama
- Langkah 2. Evaluasi terhadap semua fungsi *fitness*  $F(P)$
- Langkah 3. Dilakukan perbandingan antar  $F(P)$  untuk memperoleh  $F(P_m)$  maksimal.
- Langkah 4. Jika kriteria pengentian terpenuhi maka proses dihentikan, jika tidak maka akan diulangimulai dari langkah 1.

Evaluasi nilai *fitness* dapat dilakukan menggunakan rumus fungsi *fitness* yang disajikan pada rumus 7.

$$F(P) = e^{\frac{1}{2} X^2(P)} \quad (7)$$

Dimana  $X^2(P)$  merupakan perbedaan antara data eksperimen dan perhitungan teoritis dari parameter himpunan  $P$ . Nilai  $X^2(P)$  dapat ditentukan dengan rumus 8.

$$X^2(P) = \sum_{k=1}^{N_D} \left( \frac{y_k^{\text{exp}} - y_k^{\text{th}}(P)}{\sigma_k} \right)^2 \quad (8)$$

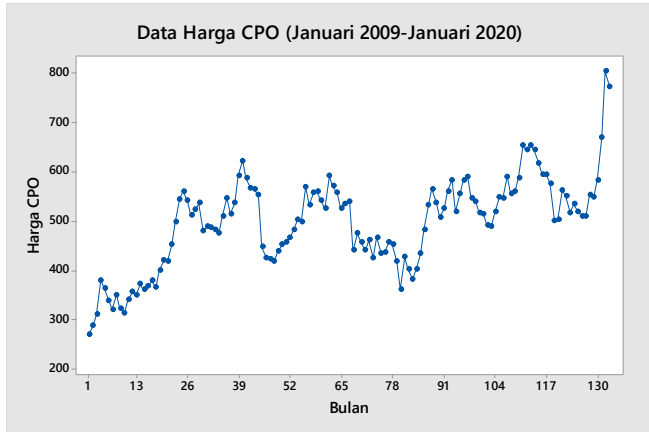
Pada persamaan 8,  $k$  merupakan iterasi terhadap  $N_D$  data, sedangkan  $y_k^{\text{exp}}$ ,  $y_k^{\text{th}}(P)$ , dan  $\sigma_k$  merupakan nilai eksperimental.

### III. METODE

Data yang digunakan berupa daftar harga CPO dunia tiap bulan yang terus mengalami fluktuasi. Data *time series* harga CPO yang digunakan adalah pada rentang waktu bulan Januari 2009 sampai dengan bulan Januari 2020. Data diakses melalui laman <https://id.investing.com/commodities/crude-palm-oil-historical-data>. Data ini disajikan melalui grafik pada gambar 2.

Pada pembentukan sistem *Fuzzy Radial Basis Fuction Neural Network* yang dioptimasi menggunakan Algoritma Genetika terdapat beberapa langkah yang harus ditempuh. Langkah pertama yang dilakukan adalah menentukan *input* yang akan digunakan pada sistem. *Input* ini ditentukan menggunakan grafik *autocorrelation function* (ACF). Banyak *input* yang digunakan ditentukan oleh banyaknya *lag* signifikan pada *plot* ACF.

Langkah kedua adalah melakukan pembagian terhadap data *input*. Data dibagi menjadi dua kelompok yaitu data *training* dan data *testing*. Merujuk pada penelitian terdahulu yang dilakukan oleh [18] proporsi yang digunakan untuk pembagian data pada penelitian ini adalah 75% untuk data *training* dan 25% untuk data *testing*.



Gambar 2. Data Time Series Harga CPO

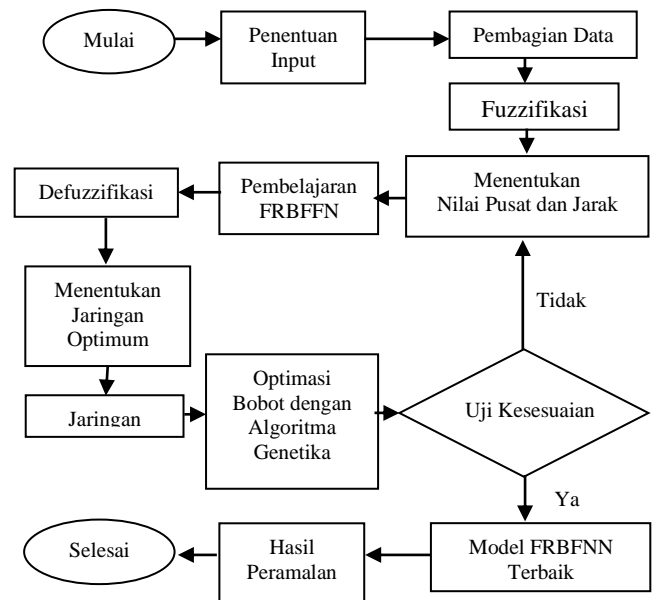
Setelah pembagian data dilakukan maka data di fuzzifikasi untuk mengubah nilai-nilai *input* yang semula berupa himpunan *crisp* menjadi himpunan *fuzzy*. Proses fuzzifikasi dilakukan menggunakan suatu fungsi keanggotaan. Setelah melalui beberapa percobaan, tipe fungsi keanggotaan yang paling sesuai untuk mengolah data pada penelitian ini adalah fungsi keanggotaan segitiga.

Langkah keempat yang dikerjakan adalah mengelompokkan data *input* yang telah defuzzifikasi menggunakan metode *K-Means* untuk memperoleh pusat *cluster* dan jarak maksimum data pada pusat *cluster*. Penentuan banyaknya *cluster* dilakukan secara *trial* dan *error* dari 2 hingga 10 *cluster*. Banyak *cluster* optimal yang dihasilkan menunjukkan jumlah *neuron* pada lapisan tersembunyi. Langkah selanjutnya adalah menentukan jaringan optimum. Proses ini dilakukan menggunakan program *rbfDesign* dan *globalRidge* dengan bantuan MATLAB. Penentuan ini dibantu menggunakan nilai MAPE.

Setelah banyak *neuron* pada lapisan *input* dan lapisan tersembunyi diperoleh pengembangan sistem dilanjutkan dengan penentuan *output* jaringan. Jumlah *neuron output* ditentukan berdasarkan banyak himpunan *fuzzy* yang digunakan. Nilai-nilai *output* yang masih berupa himpunan *fuzzy* perlu diubah kembali menjadi himpunan *crisp* untuk mempermudah representasi. Proses yang dilakukan untuk perubahan ini adalah defuzzifikasi. Pada penelitian ini digunakan tiga metode defuzzifikasi yaitu *mean of maximum*, *smallest of maximum*, dan *largest of maximum*. Langkah terakhir yang dilakukan adalah dengan mengevaluasi sistem yang telah dibangun menggunakan MAPE. Perhitungan MAPE disajikan pada rumus 9.

$$MAPE = \frac{1}{n} \sum_{t=1}^n \frac{|Y_t - \bar{Y}_t|}{Y_t} \times 100\% \quad (9)$$

Pada rumus 9  $Y_t$  menunjukkan nilai pengamatan ke- $t$  dan  $\bar{Y}_t$  menunjukkan nilai peramalan pada waktu ke- $t$ . Secara sederhana prosedur pembentukan FRBFNN dengan Algoritma Genetika disajikan melalui gambar 3.



Gambar 3. Langkah Pembentukan FRBFNN dioptimasi Algoritma Genetika

#### IV. HASIL DAN PEMBAHASAN

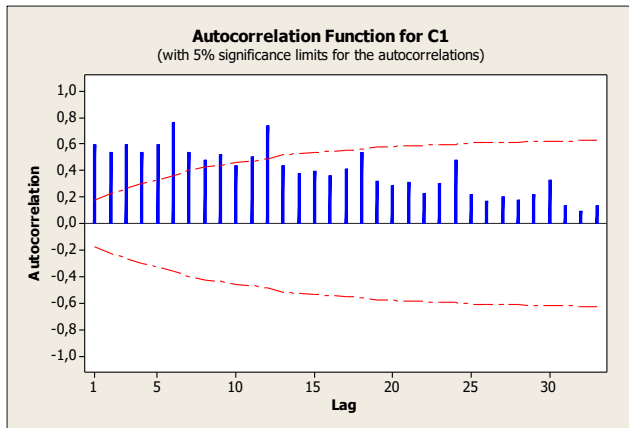
Proses penelitian dilakukan dengan terlebih dahulu mengembangkan sistem *Fuzzy Radial Basis Function Neural Network* (FRBFNN). Selanjutnya, sistem yang telah dikembangkan dioptimasi dengan Algoritma Genetika. Kedua sistem digunakan untuk memprediksi harga CPO. Hasil prediksi FRBFNN dan optimasi FRBFNN menggunakan Algoritma Genetika akan dibandingkan untuk mengetahui kualitas sistem yang dihasilkan berdasarkan nilai MAPE yang dihasilkan.

##### A. Fuzzy Radial Basis Function Neural Network (FRBFNN)

Pembentukan FRBFNN diawali dengan penentuan *input*. Penentuan data *input* menggunakan *plot autocorrelation function* (ACF) menggunakan bantuan MINITAB. Plot ACF yang dihasilkan berdasarkan data harga CPO ditampilkan pada gambar 4.

Pada gambar 4 terlihat bahwa terdapat sebelas *lag* signifikan yaitu *lag* 1, 2, 3, 4, 5, 6, 7, 8, 9, 11, dan 12. Berdasarkan *lag* signifikan ini maka diperoleh 11 *input* yaitu harga CPO *lag* ke-1 yang merupakan data harga CPO ke  $t-1$ , *lag* ke-2 yang merupakan data harga CPO ke  $t-2$ , dan seterusnya. Setelah jumlah *neuron input* ditentukan, selanjutnya data dipartisi menjadi dua bagian yaitu data

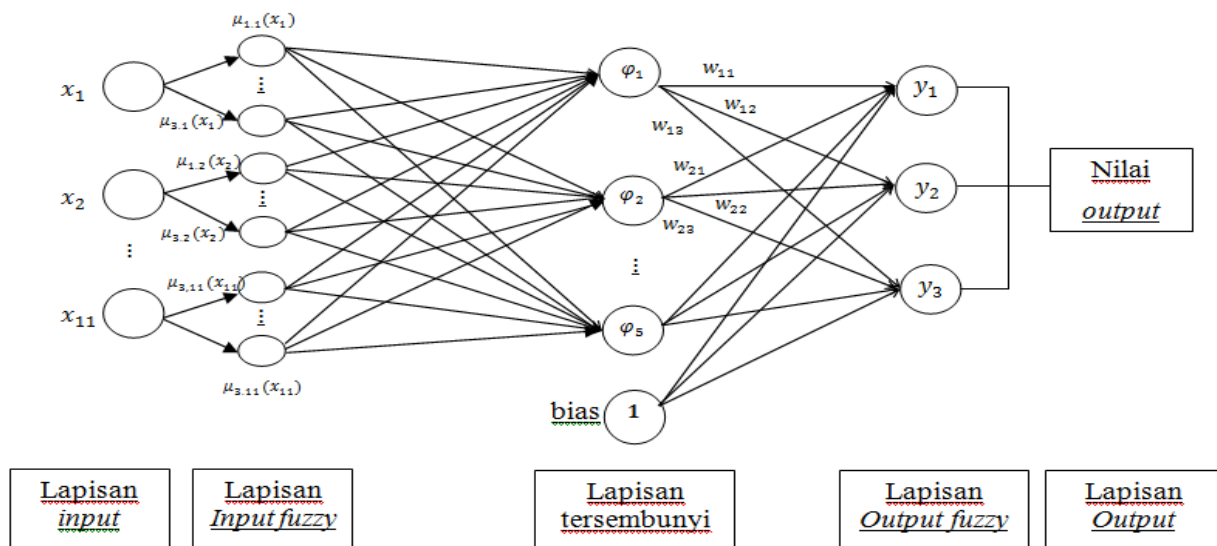
training dan data testing. Berdasarkan perolehan MAPE terkecil data dibagi menggunakan proporsi 75% untuk data training dan 25% untuk data testing.



Gambar 4. Plot Autocorrelation Function Harga CPO

Nilai-nilai input yang masih berupa himpunan *crisp* ditransformasikan menjadi himpunan *fuzzy* agar dapat diolah oleh sistem yang akan dikembangkan. Proses perubahan ini melalui beberapa tahap. Fuzzifikasi diawali dengan menentukan himpunan universal pada input dan output. Berdasarkan data harga CPO ditentukan bahwa himpunan universal input dan output adalah [269 803].

Menggunakan himpunan universal yang telah ditentukan, selanjutnya dibentuk himpunan *fuzzy* pada input dan output. Himpunan *fuzzy* pada keduanya dibagi menjadi tiga himpunan. Fungsi keanggotaan yang digunakan untuk pembentukan ini adalah fungsi keanggotaan segitiga. Himpunan yang terbentuk yaitu:  $A_1 = [55.4 \ 482.6]$ ;  $A_2 = [322.4 \ 749.6]$ ;  $A_3 = [589.4 \ 1017]$ . Proses fuzzifikasi dilakukan menggunakan MATLAB.



Gambar 5. Arsitektur FRBFNN dengan 11 input, 5 hidden, dan 3 output

Langkah selanjutnya data yang telah difuzzifikasi dikelompokkan menggunakan *K-Means*. Metode ini akan menghasilkan pusat *cluster* dan jarak maksimum data menuju pusat *cluster*. Guna memperoleh jumlah *cluster* terbaik dilakukan proses *trial error* dan menghasilkan bahwa *cluster* terbaik dengan nilai MAPE minimum adalah pada penggunaan 5 *cluster*. Hal ini menunjukkan bahwa jumlah *neuron* pada lapisan tersembunyi FRBFNN berjumlah 5 buah.

TABEL I  
NILAI MAPE MODEL FRBFNN

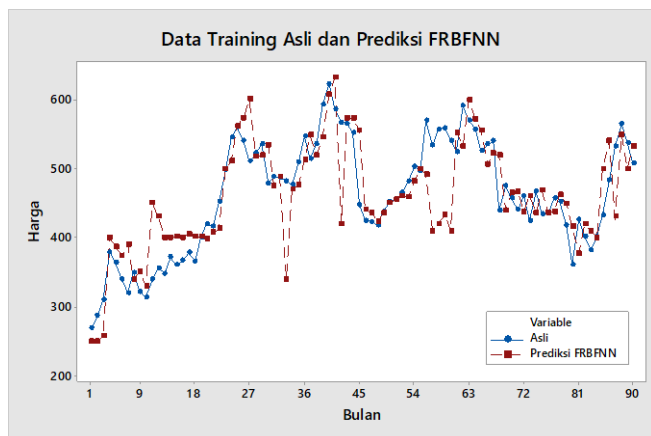
Cluster	Nilai MAPE					
	Training			Testing		
	Min	Max	Mean	Min	Max	Mean
2	22,8	13,5	18,5	12,4	10,3	12,3
3	22,5	12,6	18,5	10,9	10,4	12,3
4	23,4	13,1	18,4	15,0	10,5	11,5
5	23,5	<b>11,7</b>	18,4	14,9	<b>9,4</b>	11,4
6	21,7	11,8	18,7	11,7	11,5	12,3
7	21,8	13,0	18,3	13,7	10,4	12,4
8	22,9	14,7	18,9	14,5	9,9	12,4
9	21,2	14,8	19,2	12,5	11,4	12,4
10	22,2	12,5	17,1	12,6	10,3	12,4

Data yang telah dikelompokkan kemudian dikenai suatu pembelajaran menggunakan program **rbfDesign** dan **globalRidge** dengan bantuan MATLAB. Pada penelitian ini digunakan tiga metode defuzzifikasi yaitu *min*, *max*, dan *mean*. Hasilnya akan dievaluasi berdasarkan nilai MAPE yang dihasilkan. Nilai MAPE disajikan pada Tabel I. Pada Tabel I terlihat bahwa arsitektur terbaik terbentuk dengan 5 *cluster* yang menunjukkan jumlah *neuron* pada lapisan tersembunyi.

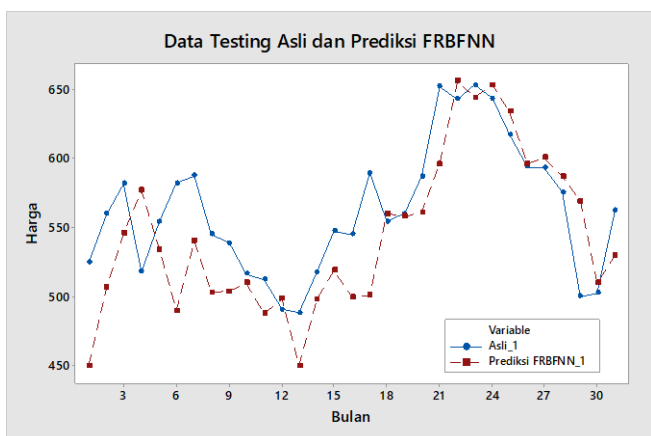
Selanjutnya *neuron* pada lapisan *output* ditentukan berdasarkan banyak himpunan *fuzzy* pada *output* yaitu tiga buah himpunan. Dengan demikian, diperoleh arsitektur FRBFNN terbaik adalah dengan 11 *neuron* pada *input*, 5 *neuron* pada lapisan tersembunyi, dan 3 *neuron* pada *output*. Ilustrasi model FRBFNN ini disajikan pada gambar 5.

Langkah terakhir proses pembentukan sistem FRBFNN adalah defuzzifikasi. Pada penelitian ini dilakukan tiga macam defuzzifikasi yaitu *mean of maximum*, *smallest of maximum*, dan *largest of maximum*. Dari ketiga macam defuzzifikasi ini, hasil terbaik diperoleh dengan metode *largest of maximum*.

Berdasarkan rumus 9 nilai MAPE yang dihasilkan untuk data *training* adalah sebesar 11,7% dan nilai MAPE untuk data *testing* adalah 9,4%. Grafik perbandingan data harga CPO asli dengan prediksi FRBFNN pada data *training* disajikan pada gambar 6 dan pada data *testing* disajikan pada gambar 7.



Gambar 6. Data Training Asli dan Prediksi FRBFNN



Gambar 7. Data Testing Asli dan Prediksi FRBFNN

Hasil ini sejalan dengan penelitian yang dilakukan oleh Xiong [19] bahwa *Radial Basis Fuction Nerual Network* memiliki potensi yang baik untuk melakukan prediksi. Teknik *K-means clustering* yang digunakan pada penelitian

ini ternyata juga memberikan hasil yang cukup baik sesuai dengan penelitian yang dilakukan oleh Pomares [20]. Selain itu, hasil prediksi FRBFNN pada penelitian ini juga mendukung penelitian [21] bahwa *artificial neural network* cocok untuk digunakan dalam prediksi di bidang ekonomi.

### B. Optimasi dengan Algoritma Genetika

Nilai bobot yang diperoleh pada FRBFNN dioptimasi kembali menggunakan algoritma genetika untuk mendapatkan *error* yang minimum. Tingkat *error* ini direpresentasikan dengan nilai MAPE. Langkah pertama dalam optimasi menggunakan algoritma genetika adalah pengkodean gen. Gen yang digunakan merupakan nilai bobot akhir dari model FRBFNN yang merupakan bobot *output* dan bobot bias.

Langkah selanjutnya dibangkitkan populasi awal yang merupakan gabungan dari beberapa individu, dimana tiap individu merupakan kumpulan gen. Individu pertama disusun berdasarkan bobot akhir FRBFNN, sedangkan individu selanjutnya dibentuk secara acak. Ukuran generasi pada penelitian ini disusun sebanyak seratus. Guna mendapatkan solusi yang terbaik dilakukan evaluasi nilai *fitness*.

Evaluasi ini berusaha untuk mencari nilai *error* terkecil. Nilai MAPE menjadi acuan utama pada proses evaluasi nilai *fitness*. Berdasarkan evaluasi yang dilakukan diketahui bahwa nilai *fitness* terbaik didapatkan saat populasi dengan ukuran delapan belas.

Individu-individu dengan nilai *fitness* terbaik disimpan melalui tahap *elitism* dengan tujuan agar tidak hilang saat dilakukan proses mutasi dan pindah silang. Penseleksian individu dengan nilai *fitness* terbaik dilakukan menggunakan metode seleksi *ranking*. Pada proses ini nilai *fitness* setiap individu ditentukan kemudian diurutkan dari nilai yang terkecil. Individu dengan nilai *fitness* paling kecil diberi nilai *fitness* baru sebesar satu, individu terkecil kedua diberi nilai *fitness* dua, dan seterusnya. Masing-masing nilai *fitness* dibagi untuk mendapatkan probabilitas dari masing-masing *fitness*. Kemudian dibangkitkan suatu nilai *random* pada interval [0 1]. Menggunakan bantuan MATLAB diperoleh nilai 0,0000511 dan 0,001733. Pada frekuensi kumulatif 0,0000511 terletak pada individu pertama, dan 0,001733 terletak pada individu ke-11. Oleh karena itu individu pertama dan ke-11 dipilih untuk menjadi induk pada proses pindah silang.

Pindah silang dilakukan untuk memperoleh kromosom baru dengan mempertukarkan gen. Proses ini dilakukan pada induk yang telah diseleksi. Nilai probabilitas yang digunakan adalah sebesar 0,8. Pindah silang diawali dengan membangkitkan nilai probabilitas induk. Menggunakan MATLAB diperoleh angka sebesar 0,089. Nilai ini mengindikasikan bahwa proses pindah silang dapat dilakukan. Penentuan posisi gen pindah silang dilakukan dengan membangkitkan dua buah nilai acak menggunakan MATLAB. Nilai yang diperoleh adalah 14 dan 18, sehingga gen ke-14 sampai gen ke-18 dikenai operasi pindah silang.

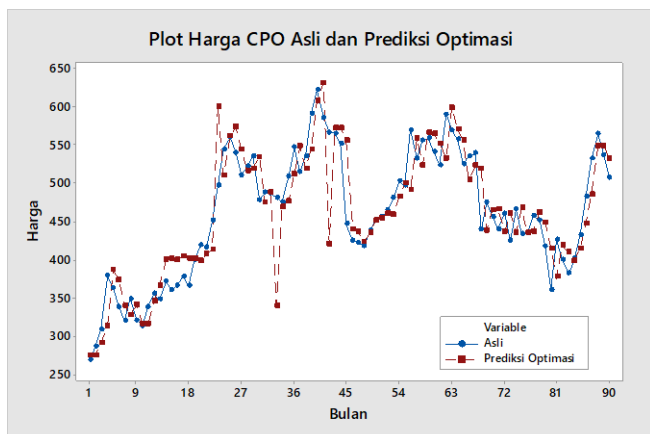
Langkah selanjutnya dalam algoritma genetika adalah proses mutasi. Pada penelitian ini dilakukan *random mutation* pada kromosom untuk menghasilkan kromosom baru. Nilai probabilitas yang digunakan sebesar 0,02. Pada anak satu diperoleh probabilitas individu 0,0023 dan pada anak 2 diperoleh probabilitas sebesar 0,009. Dalam interval [1 18] ditentukan nilai acak menggunakan MATLAB dan diperoleh nilai 9 untuk anak 1 dan 17 untuk anak dua. Akibatnya gen ke-9 dalam anak 1 dan gen ke-17 dalam anak dua dimutasi menggantikan nilai dengan nilai acak.

Langkah terakhir yang dilakukan adalah membentuk populasi baru. Populasi ini terbentuk berdasarkan evaluasi nilai fitness terbaik dan hasil *elitism*. Berdasarkan rumus 9, ditentukan nilai MAPE sistem FRBFNN yang telah dioptimasi pada data training sebesar 10,2% dan pada data testing sebesar 8,3%. Dengan demikian perbandingan nilai MAPE pada FRBFNN sebelum dan sesudah dioptimasi menggunakan algoritma genetika disajikan pada tabel II.

TABEL III  
NILAI MAPE FRBFNN DAN FRBFNN DENGAN ALGORITMA GENETIKA

Prediksi	Nilai MAPE	
	Training	Testing
FRBFNN	11,7%	9,4%
FRBFNN dengan Algoritma Genetika	10,2%	8,3%

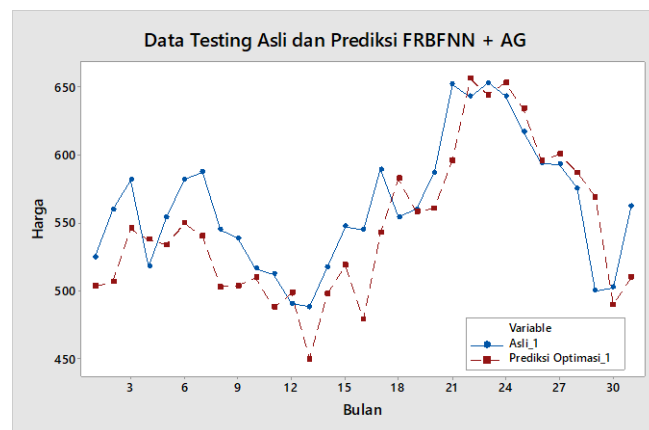
Perbandingan data asli dan data hasil prediksi FRBFNN dioptimasi menggunakan algoritma genetika untuk data training disajikan pada gambar 8 dan untuk data testing disajikan pada gambar 9.



Gambar 8. Data Training Asli dan Prediksi FRBFNN dengan Algoritma Genetika

Hasil penelitian ini mendukung penelitian yang dilakukan oleh Awad [22] bahwa *Radial Basis Fuction* yang dioptimasi menggunakan algoritma genetika memberikan hasil prediksi yang lebih baik dibandingkan dengan prediksi menggunakan *Radial Basis Fuction* saja. Penggunaan *Radial Basis Function* menyederhanakan perhitungan

sehingga proses yang dilakukan dapat mencapai *error* yang minimum [23].



Gambar 9. Data Testing Asli dan Prediksi FRBFNN dengan Algoritma Genetika

### C. Prediksi Harga CPO

Sistem prediksi FRBFNN dan FRBFNN yang telah dioptimasi, selanjutnya digunakan untuk melakukan prediksi harga CPO pada bulan Februari hingga Juni 2020. Hasil prediksi ini disajikan pada tabel III.

TABEL IIIII  
PREDIKSI CPO FEBRUARI – JUNI 2020

Bulan	Asli	FRBFNN	FRBFNN + AG
Februari	692.2	672.6	682
Maret	682	582	570
April	631.7	677.2	621.5
Mei	658.2	546	555
Juni	719.4	578	657.6

### V. KESIMPULAN

*Crude Palm Oil* (CPO) yang merupakan komoditi ekspor penting di Indonesia. Persaingan yang semakin ketat di antar negara penghasil minyak kelapa sawit mengakibatkan fluktuasi harga yang sering terjadi. Tanpa adanya manajemen yang baik, fluktuasi harga dapat menimbulkan kerugian yang besar untuk negara. Prediksi harga dapat menjadi salah satu alternatif solusi untuk manajemen penghasilan CPO sehingga dapat memberikan hasil optimal. Prediksi harga CPO menggunakan *Fuzzy Radial Basis Fuction Neural Network* (FRBFNN) pada penelitian ini terbukti cukup baik untuk memperkirakan harga CPO selanjutnya yaitu dengan tingkat *error* sebesar 11,7% pada data *training* dan nilai 9,4% pada data *testing*. Sistem FRBFNN memiliki arsitektur dengan 11 *neuron input*, 5 *neuron* lapisan tersembunyi, dan 3 *neuron* pada *output*. Pada hasil optimasi FRBFNN menggunakan algoritma genetika diketahui bahwa hasil prediksi meningkat dengan hanya menghasilkan *error* sebesar 10,2% pada data *training* dan 8,3% pada data *testing*. Pada penelitian selanjutnya diharapkan *error* yang dihasilkan dapat lebih diminimalisir dengan mencoba untuk menambahkan data

set pada proses *training*. Selain itu, juga dirasa juga perlu untuk mempertimbangkan aspek-aspek yang mempengaruhi harga CPO dalam proses prediksi yang dilakukan.

#### DAFTAR PUSTAKA

- [1] H. Aini and H. Haviluddin, "Crude Palm Oil Prediction Based on Backpropagation Neural Network Approach," *Knowl. Eng. Data Sci.*, vol. 2, pp. 1-9, Jun. 2019.
- [2] N. F. Rahim, M. Othman, R. Sokkalingam, and E. Abdul Kadir, "Forecasting Crude Palm Oil Prices Using Fuzzy Rule-Based Time Series Method," *IEEE Access*, vol. 6, no. 45, pp. 32216–32224, 2018.
- [3] A. L. Rucitra and M. A. Pradana, "Demand forecasting for crude palm oil (CPO) using the time series method," in *IOP Conference Series: Earth and Environmental Science*, 2020, vol. 475, no. 1, p. 475.
- [4] S. Ahmad and H. A. Latif, "Forecasting on the Crude Palm Oil and Kernel Palm Production: Seasonal ARIMA Approach," 2011 IEEE Colloquium on Humanities, Science and Engineering, Penang, 2011, pp. 939-944.
- [5] S. Ramakrishnan, S. Butt, M. A. Chohan, and H. Ahmad, "Forecasting Malaysian exchange rate using machine learning techniques based on commodities prices," *2017 International Conference on Research and Innovation in Information Systems (ICRIIS)*, 2017, Langnawi, p. 1.
- [6] J. Sanjaya, E. Renata, V. E. Budiman, F. Anderson, and M. Ayub, "Prediksi Kelalaian Pinjaman Bank Menggunakan Random Forest dan Adaptive Boosting," *J. Tek. Inform. dan Sist. Inf.*, vol. 6, no. 1, pp. 50–60, Apr. 2020.
- [7] S. K. & H. Purnomo, *Aplikasi Logika Fuzzy untuk Pendukung Keputusan*. Yogyakarta: Graha Ilmu, 2013.
- [8] U. of Walkato.(2017) Community development library. [Online]. Tersedia: <http://www.nzdl.org>.
- [9] S.-K. Oh, W.-D. Kim, W. Pedrycz, and K. Seo, "Fuzzy Radial Basis Function Neural Networks with information granulation and its parallel genetic optimization," *Fuzzy Sets Syst.*, vol. 237, pp. 96–117, Feb. 2014.
- [10] Suyanto, *Artificial Intelligence Searching, Reasoning, Planning, dan Learning Revisi Kedua*. Bandung: Informatika Bandung, 2014.
- [11] N. Widiangga, A. Dharma, and M. Sudarma, "Analisis Metode RBF-NN dengan Optimasi Algoritma Genetika pada Peramalan Mata Uang EUR/USD," *J. Teknol. Elektro*, vol. 15, no. 2, pp. 107-114, Des. 2016.
- [12] A. Norhidayu, M. Nur-Syazwani, R. Radzil, I. Amin, and N. Balu, "The production of crude palm oil in Malaysia," *Int. J. Econ. Manag.*, vol. 11, no. 3, pp. 591-606, Des. 2017.
- [13] S. Kusumadewi and H. Purnomo, *Aplikasi Logika Fuzzy untuk Pendukung Keputusan*. Yogyakarta: Graha Ilmu, 2010.
- [14] P. M. ; Shena and A. . Bapat, "Clustering Algorithms for Radial Basis Function Neural Network," *Semantic Scholar*, vol. 1, no. 1, pp. 113–116, Jan. 2013.
- [15] V. Sanchez-Tembleque, V. Vedia, L. M. Fraile, S. Ritt, and J. M. Udias, "Optimizing time-pickup algorithms in radiation detectors with a genetic algorithm," *Nucl. Instruments Methods Phys. Res. Sect. A Accel. Spectrometers, Detect. Assoc. Equip.*, vol. 927, pp. 54–62, May. 2019.
- [16] P. L. L. Belluano, "Optimasi Solusi Terbaik dengan Penerapan Non-Dominated Sorting II Algorithm," *J. Ilm. Ilk.*, vol. 8, no. 1, pp. 29–36, Apr. 2016.
- [17] X.-L. Luo, J. Feng, and H.-H. Zhang, "A genetic algorithm for astroparticle physics studies," *Comput. Phys. Commun.*, vol. 250, pp. 1–7, Jun. 2019.
- [18] J. T. Leonard and K. Roy, "On Selection of Training and Test Sets for the Development of Predictive QSAR models," *QSAR Comb. Sci.*, vol. 25, no. 3, pp. 235–251, Mar. 2006.
- [19] T. Xiong, Y. Bao, Z. Hu, and R. Chiong, "Forecasting interval time series using a fully complex-valued RBF neural network with DPSO and PSO algorithms," *Inf. Sci. (Ny.)*, vol. 305, pp. 77–92, jun. 2015.
- [20] M. Awad, H. Pomares, I. Rojas, O. Salameh, and M. Hamdon, "Prediction of time series using RBF neural networks: A new approach of clustering," *Int. Arab J. Inf. Technol.*, vol. 6, no. 2, pp. 138–143, Apr. 2009.
- [21] Y. chen Wu and J. wen Feng, "Development and Application of Artificial Neural Network," *Wirel. Pers. Commun.*, vol. 102, no. 2, pp. 1645–1656, Des 2018.
- [22] M. Awad, *Forecasting of Chaotic Time Series Using RBF Neural Networks Optimized By Genetic Algorithms*. Palestina: IAJIT First Online Publication, 2015.
- [23] L. Qu, Y. Chen, and Z. Liu, "Time series forecasting model with error correction by structure adaptive RBF neural network," in *Proceedings of the World Congress on Intelligent Control and Automation (WCICA)*, 2006, vol. 2, pp. 6831–6835.