

## PERAMALAN INFLASI DENGAN METODE *PARTICLE SWARM OPTIMIZATION* – *EXTREME LEARNING MACHINE*

**Raudhatul Ilmiyah\*, Dewi Rachmatin, Rini Marwati**

Program Studi Matematika, FPMIPA, Universitas Pendidikan Indonesia  
e-mail: iroudholotul@gmail.com

Diterima: 31 Januari 2022, disetujui: 1 Maret 2022, dipublikasi: 22 April 2022

**Abstract:** *The Extreme Learning Machine (ELM) method is a learning method on an artificial neural network that is used to solve non-linear problems. In this article, the Particle Swarm Optimization (PSO) method is combined with the Extreme Learning Machine (ELM) method to determine the number of neurons in the hidden layer, so that the results obtained are quite accurate and more efficient in terms of time. Furthermore, the PSO-ELM method was applied to inflation data in Indonesia, and the results showed that the Roots Mean Square Error (RMSE) value in the training process was 0.005487935, and the RMSE value in the testing process was 0.4124935. These results indicate that the PSO-ELM method is suitable for forecasting inflation in Indonesia.*

**Keywords:** *Particle Swarm Optimization (PSO), Extreme Learning Machine (ELM)*

**Abstrak:** *Metode Extreme Learning Machine (ELM) merupakan metode pembelajaran pada jaringan syaraf tiruan yang digunakan untuk mengatasi masalah non-linier. Pada artikel ini metode Particle Swarm Optimization (PSO) digabungkan dengan metode Extreme Learning Machine (ELM) untuk menentukan jumlah neuron di lapisan tersembunyi, sehingga hasil yang diperoleh cukup akurat dan lebih efisien dari segi waktu. Selanjutnya metode PSO-ELM ini diterapkan pada data inflasi di Indonesia, dan diperoleh hasil bahwa nilai Roots Mean Square Error (RMSE) pada proses training sebesar 0,005487935, dan nilai RMSE pada proses testing sebesar 0,4124935. Hasil ini menunjukkan bahwa metode PSO-ELM cocok digunakan untuk peramalan inflasi di Indonesia.*

**Kata Kunci:** *Particle Swarm Optimization (PSO), Extreme Learning Machine (ELM)*

## PENDAHULUAN

Isu perekonomian menjadi perhatian penting dari pemerintah negara-negara di dunia khususnya negara berkembang yaitu Indonesia adalah inflasi. Inflasi merupakan salah satu indikator stabilitas perekonomian [8]. Secara sederhana, Inflasi diartikan sebagai kenaikan harga barang dan jasa secara umum dan terus menerus dalam jangka waktu tertentu. Kenaikan harga dari satu atau dua barang saja tidak dapat disebut inflasi kecuali bila kenaikan itu meluas (atau mengakibatkan kenaikan harga) pada barang lainnya. Kenaikan harga akan mempengaruhi masyarakat terutama mereka yang berpenghasilan rendah dan yang berpenghasilan tetap karena nilai upah yang sama, tetapi biaya pengeluaran meningkat [15]. Dalam hal ini, pemerintah melalui bank sentral berkepentingan untuk mengendalikan laju perubahannya. Pentingnya pengendalian inflasi didasarkan bahwa inflasi yang tinggi dan tidak stabil memberikan dampak negatif kepada kondisi sosial ekonomi masyarakat [16].

Beberapa penelitian yang berkaitan dengan peramalan inflasi antara lain dilakukan oleh [14] dengan menggunakan metode ARMA (*Autoregressive Moving Average*) dan metode ANFIS (*Adaptive Neuro-Fuzzy Inference System*). Hasil penelitian menunjukkan bahwa metode ANFIS lebih baik dibanding dengan menggunakan metode konvensional. Penelitian yang dilakukan oleh [4] menggunakan metode ARIMA (*Autoregressive Integrated Moving Average*) menyatakan bahwa hasil peramalan laju inflasi mampu mengikuti pergerakan data aktual dari laju inflasi. Penelitian lain yang dilakukan oleh [12] menggunakan analisis regresi linier dengan metode ANN (*Artificial Neural Network*) menyatakan bahwa hasil peramalan menggunakan metode ANN lebih baik dengan analisis regresi linier. Selain itu, penelitian yang dilakukan oleh [10] menggunakan metode ARIMA (*Autoregressive Integrated Moving Average*) dan metode ANN (*Artificial Neural Network*) menunjukkan bahwa metode ANN menghasilkan nilai RMSE (*Root Mean Squared Error*) lebih kecil daripada model ARIMA. Sehingga dapat dikatakan bahwa masalah inflasi merupakan masalah yang kompleks dan non-linier yang tidak dapat disimulasikan menggunakan metode langsung konvensional.

Metode jaringan syaraf tiruan (*artificial neural network*) dapat digunakan untuk mengatasi masalah non-linier, berikut ini beberapa metode pembelajaran pada jaringan syaraf tiruan di antaranya ELM (*Extreme Learning Machine*) dan *backpropagation*. Pada penelitian ini akan digunakan metode *neural network* dengan *backpropagation*. Hasil penelitian dapat disimpulkan bahwa perform metode *neural network* dengan *backpropagation* memberikan tingkat akurasi peramalan yang cukup baik [2]. Pada penelitian yang dilakukan oleh [1] dengan menggunakan metode ELM dan metode *backpropagation* menunjukkan bahwa metode ELM mendapatkan nilai kesalahan lebih kecil dibandingkan dengan metode *backpropagation*. Hal tersebut membuktikan bahwa metode ELM sangat cocok digunakan untuk peramalan. Selain itu, ELM juga mempunyai kelemahan pada proses penentuan jumlah neuron di lapisan tersembunyi (*hidden layer*) yang menggunakan proses *trial* dan *error*. Terdapat pula penelitian yang dilakukan oleh [10], yang menyatakan bahwa sistem neuro-fuzzy berbasis ELM lebih baik dibanding dengan jenis lainnya. Selain itu, sistem neuro-fuzzy berbasis populasi memiliki kemampuan dalam mendapatkan solusi optimal.

Dalam penelitian ini, digunakan metode ELM yang merupakan metode yang memiliki tingkat pembelajaran yang cepat dan akurasi yang tinggi. Dalam penelitian sebelumnya dijelaskan bahwa metode ELM mempunyai kelemahan dalam proses

penentuan jumlah neuron di lapisan tersembunyi (*hidden layer*) yang menggunakan proses *trial* dan *error*. Untuk mendapatkan hasil yang optimal, metode ELM dapat digabungkan dengan salah satu metode optimasi yaitu PSO (*Particle Swarm Optimization*), sehingga didapatkan hasil peramalan yang lebih akurat.

## METODOLOGI PENELITIAN

### 1. *Extreme learning machine* (ELM)

*Extreme learning machine* (ELM) pertama kali diperkenalkan oleh Huang tahun 2004. ELM merupakan metode pembelajaran baru dari jaringan syaraf tiruan umpan maju (*feedforward network*) yang memiliki satu lapisan tersembunyi atau dikenal dengan istilah *single hidden layer feedforward neural network* (SLNs). Metode ini diciptakan untuk mengatasi kelemahan jaringan syaraf tiruan umpan maju dalam proses kecepatan pembelajaran (*learning speed*)[5].

Langkah-langkah yang dilakukan dalam perhitungan dengan metode ELM menurut [3] adalah sebagai berikut:

Langkah 1: Normalisasi data

Normalisasi data merupakan suatu proses perubahan bentuk data menjadi nilai yang lebih spesifik dalam batas nilai 0 sampai 1. Perhitungan menggunakan persamaan (1)

$$v' = \frac{v}{max} \quad (1)$$

di mana,

$v'$  : data hasil normalisasi.

$v$  : data asli.

$max$ : nilai maksimum dari seluruh data pada setiap patameter.

Langkah 2: Proses *Training*

Proses *training* bertujuan untuk melakukan pelatihan dengan menggunakan data latih. Pada pelatihan ini akan didapatkan nilai bobot dan bias yang optimal. Langkah-langkah pada proses *training*, sebagai berikut:

- 1) Inisialisasi bobot *input* dan bias secara acak pada *range* [0,1]
- 2) Menghitung matriks *output hidden layer* menggunakan persamaan (2)

$$H = \frac{1}{1 + \exp(-(x_{train}w^T + b))} \quad (2)$$

di mana,

$H$  : matriks *output hidden layer*

$x_{train}$  : matriks *input* atau data latih

$w^T$  : matriks transpose dari bobot *input*

$b$  : matriks bias

- 3) Menghitung matriks *Moore-Penrose pseudo inverse* menggunakan persamaan (3)

$$H^+ = (H^T H)^{-1} H^T \quad (3)$$

di mana,

$H^+$  : matriks *Moore-Penrose pseudo inverse*

$H^T$  : matriks transpose *output hidden layer*.

$H$  : matriks *output hidden layer*

4) Menghitung bobot *output* menggunakan persamaan (4)

$$\beta = H^+T \quad (4)$$

di mana,

- $\beta$  : matriks bobot *output*
- $H^+$  : matriks *Moore-Penrose pseudo inverse*
- $T$  : matriks data *output* atau target latih

5) Menghitung hasil peramalan dari perkalian matriks *output hidden layer* dengan bobot *output* menggunakan persamaan (5)

$$Y = H\beta \quad (5)$$

di mana,

- $Y$  : hasil peramalan
- $H$  : matriks *output hidden layer*
- $\beta$  : matriks bobot *output*

### Langkah 3: Proses *Testing*

Setelah melakukan proses *training*, dilakukan proses *testing* menggunakan data uji. Proses *testing* bertujuan untuk melakukan uji coba hasil *training*, sehingga dapat diketahui akurasi dari proses. Langkah-langkah pada proses *testing* sebagai berikut:

- 1) Inisialisasi bobot *input* dan bias yang didapat dari proses *training*
- 2) Menghitung matriks *output hidden layer* menggunakan persamaan (2)
- 3) Menghitung hasil peramalan dari perkalian matriks *output hidden layer* dengan bobot *output* dari proses *training* menggunakan persamaan (5)

### Langkah 4: Hitung Evaluasi

Evaluasi digunakan sebagai proses pengukuran tingkat akurasi hasil peramalan dari suatu hal. Perhitungan evaluasi menggunakan persamaan (6). Nilai RMSE (*Root Mean Square Error*) yang kecil menunjukkan bahwa nilai yang dihasilkan oleh suatu model peramalan mendekati nilai observasinya. Semakin kecil nilai RMSE, maka semakin dekat nilai yang diramalkan.

$$RMSE = \sqrt{\sum_{i=1}^N \frac{((data\ aktual)_i - (data\ peramalan)_i)^2}{N}} \quad (6)$$

di mana,

- $N$  : jumlah seluruh data
- $i$  : data ke- $i$
- data aktual: data sebenarnya
- data peramalan : data hasil peramalan

### Langkah 5: Denormalisasi Data

Denormalisasi bertujuan untuk mengembalikan nilai yang telah ternormalisasi ke nilai aslinya supaya memperoleh nilai *output* pada *range* yang sebenarnya. Perhitungan menggunakan persamaan (7)

$$v = \frac{(v' - new\_min)}{(new\_max)(max - min)} + min \quad (7)$$

di mana,

- $v'$  : data hasil normalisasi
- $v$  : data hasil denormalisasi
- $max$  : nilai maksimum dari seluruh data pada setiap parameter
- $min$  : nilai minimum dari seluruh data pada setiap parameter
- $new\_max$ : nilai batas maksimum dalam pemetaan

$new\_min$  : nilai batas minimum dalam pemetaan

## 2. Particle swarm optimization (PSO)

Particle swarm optimization (PSO) pertama kali diperkenalkan oleh James Kennedy dan Russell Eberhart pada tahun 1995 sebagai salah satu metode optimasi yang digunakan untuk menentukan parameter-parameter proses yang menghasilkan nilai respon yang optimum. PSO terinspirasi pada kejadian fenomena alam yang berawal dari pengamatan perilaku sosial sekawanan burung dan ikan pada organisme. Pengertian dari perilaku sosial yang dimaksud adalah tindakan individu dan pengaruh dari individu-individu lain dalam suatu kelompok [13]. Setiap individu berperilaku dengan menggunakan kecerdasannya (*intelligence*) sendiri dan juga dipengaruhi perilaku kelompok kolektifnya.

PSO merupakan algoritma berbasis populasi yang mengeksploitasi individu dalam pencarian. Dalam PSO, perilaku seekor burung dan ikan dalam kawanan disebut *swarm* dan seekor burung dan ikan dalam sekelompok kawanan tersebut disebut dengan partikel. Jika seekor burung menemukan jalan yang tepat (*optimal*) atau posisi terbaik yang pernah dilaluinya untuk menuju ke sumber makan (dapat direpresentasikan sebagai nilai fungsi tujuan), maka sisa anggota kelompok yang lain juga akan mengikuti jalan tersebut meskipun lokasi mereka jauh di kelompok tersebut. Meskipun pada kawanan burung relatif kurang canggih dengan kemampuan terbatas yang dimilikinya, namun kemampuan tersebut mampu mencapai tujuan akhir dari penyelesaian terkait pola perilaku dan interaksi melalui berbagai sumber informasi[9].

Langkah-langkah untuk menghitung dengan metode PSO menurut [6] adalah sebagai berikut :

Langkah 1: Inisialisasi parameter, posisi dan kecepatan awal partikel secara acak.

Langkah 2: Menghitung nilai *fitness* terbaik dari setiap partikel berdasarkan posisinya.

Langkah 3: Menentukan nilai  $P_{best}$ . Tetapkan  $P_{best}$  awal sama dengan nilai posisi awal partikel. Jika nilai *fitness* terbaik saat ini lebih baik dari nilai *fitness* posisi partikel terbaik ( $P_{best}$ ), maka nilai  $P_{best}$  yang baru diambil dari nilai posisi partikel saat ini.

Langkah 4: Menentukan nilai  $G_{best}$  dilakukan dengan cara melihat  $P_{best}$  yang memiliki nilai *fitness* posisi partikel terbaik (nilai *fitness* minimum).

Langkah 5: Hitung *update* kecepatan partikel menggunakan persamaan (8)

$$v_{l,k}^{t+1} = wv_{l,k}^t + C_1r_1(P_{best_{l,k}}^t - x_{l,k}^t) + C_2r_2(G_{best_{g,k}}^t - x_{l,k}^t) \quad (8)$$

dan hitung *update* posisi setiap partikel menggunakan persamaan (9)

$$x_{l,k}^{t+1} = x_{l,k}^t + v_{l,k}^{t+1} \quad (9)$$

di mana,

$w$  : bobot inersia.

$v_{l,k}^t$  : vektor kecepatan partikel ke- $l$  dalam dimensi  $k$  pada waktu  $t$ .

$x_{l,k}^t$  : vektor posisi partikel ke- $l$  dalam dimensi  $k$  pada waktu  $t$ .

$C_1, C_2$  : parameter *learning rates*.

$r_1, r_2$  : vector acak dari distribusi seragam dalam kisaran 0 sampai dengan 1.

$P_{best_{l,k}}^t$  : posisi terbaik dari individu partikel ke- $l$  dalam dimensi  $k$  yang ditemukan dari inisialisasi pada waktu  $t$ .

$G_{best_{g,k}}^t$  : posisi terbaik dari global partikel ke- $g$  dalam dimensi  $k$  yang ditemukan dari inisialisasi pada waktu  $t$ .

Langkah 6: Ulangi langkah 2-5 sampai memenuhi kondisi pemberhentian (*stopping criteria*). Ada beberapa kondisi berhenti yang dapat digunakan, yaitu:

- ketika iterasi sudah mencapai maksimum.
- ketika nilai  $G_{best}$  tidak ada perubahan yang signifikan.

Parameter bobot inersia (*inertia weight*) digunakan untuk mengontrol pengaruh dari kecepatan partikel sebelumnya. Jika nilai inersia terlalu besar, maka kecepatan akan terus meningkat sehingga partikel akan divergen. Jarak partikel terhadap nilai optimumnya akan terus meningkat tiap iterasi. Berikut ini rumus bobot inersia acak (*random inertia weight*):

$$w = 0,5 + \frac{r}{2}; r \sim U[0,1] \quad (10)$$

Parameter lain dalam algoritma PSO adalah *learning rates*. Parameter  $C_1$  disebut parameter kognitif, yaitu parameter yang digunakan untuk menilai suatu partikel dengan mengontrol pengaruh  $P_{best}$  terhadap posisi partikel. Sedangkan parameter  $C_2$  disebut parameter sosial, yaitu parameter yang digunakan untuk menilai suatu kemampuan sosial dari suatu *swarm* dengan mengontrol pengaruh  $G_{best}$  terhadap posisi partikel [7].

## HASIL DAN PEMBAHASAN

Metode yang digunakan untuk peramalan inflasi di Indonesia adalah metode PSO-ELM. Metode PSO digunakan untuk menentukan nilai bobot dan bias yang optimal. Metode ELM digunakan untuk mengetahui akurasi dari proses. Pada penelitian ini dilakukan langkah-langkah sebagai berikut:

### 1. Pengumpulan Data

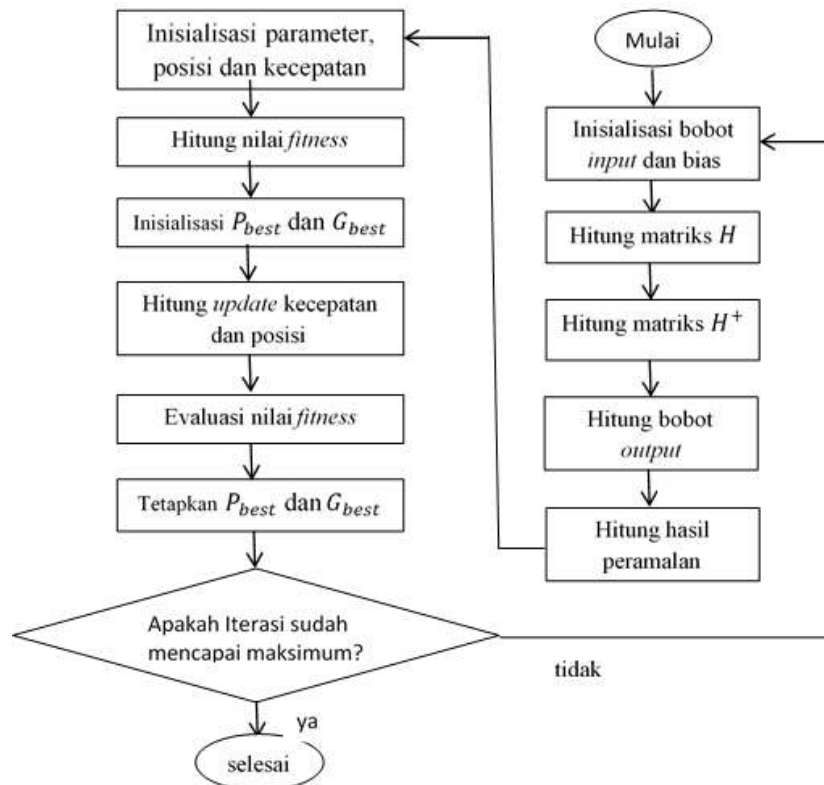
Data yang diperlukan dalam penelitian ini adalah data inflasi di Indonesia pada bulan April 2018 sampai Maret 2021. Data inflasi di Indonesia dapat diperoleh dari situs <https://www.bi.go.id/id/statistik/indikator/data-inflasi.aspx>.

### 2. Normalisasi Data

Tahap ini dilakukan untuk mencegah adanya anomali basis data. Normalisasi data digunakan untuk mengukur data sehingga nilainya berada dalam *range* 0 sampai 1, dengan cara membagi setiap periode data inflasi dengan nilai maksimum dari data inflasi. Perhitungan menggunakan persamaan (1).

### 3. Proses Training

Proses *training* dilakukan menggunakan data latih. *Flowchart* proses PSO-ELM dapat dilihat pada Gambar 1. Pada proses *training*, diperlukan beberapa parameter yang akan digunakan sebagai pembentuk sebuah jaringan. Langkah pertama yang dilakukan yaitu inisialisasi parameter bobot *input* dan bias diperoleh secara acak dengan interval 0 sampai. Pada penelitian ini digunakan jumlah neuron tersembunyi berjumlah 1 dan jumlah neuron *input* berjumlah 12. Setelah menginisialisasi bobot *input* dan bias, langkah selanjutnya dilakukan perhitungan hasil peramalan inflasi di Indonesia dan akan dihitung nilai evaluasinya dengan menggunakan persamaan RMSE.



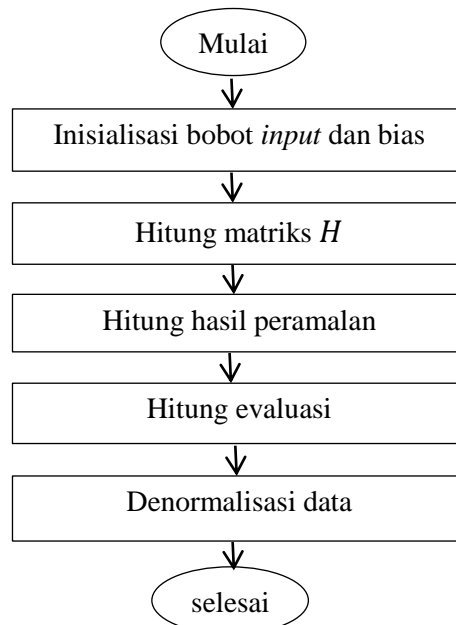
Gambar 1. Proses PSO-ELM

Setelah diperoleh nilai RMSE pada proses *training*, langkah selanjutnya akan dilakukan algoritma PSO untuk menentukan bobot dan bias yang optimal. Langkah pertama yaitu inisialisasi parameter dan inisialisasi populasi. Inisialisasi parameter dari algoritma PSO meliputi:  $w = 0,9$ ,  $C_1 = 1$ ,  $C_2 = 2$ ,  $r_1 = 0,8794171$ , dan  $r_2 = 0,03622444$ . Inisialisasi populasi dari algoritma PSO meliputi: inisialisasi posisi awal partikel. Tetapkan bobot *input* sebagai posisi awal partikel. Inisialisasi kecepatan awal dilakukan secara acak. Proses inisialisasi posisi dan kecepatan awal tersebut dilakukan sebanyak 12 jumlah partikel dan 1 jumlah variabel. Langkah selanjutnya yaitu menghitung nilai *fitness*. Pada langkah ini, tetapkan nilai RMSE sebagai nilai *fitness*. Selanjutnya, inisialisasi nilai  $P_{best}$  dan  $G_{best}$ . Setelah mendapatkan semua parameter algoritma PSO akan dihitung *update* kecepatan dan posisi partikel. Selanjutnya yaitu evaluasi nilai *fitness*. Tetapkan  $P_{best}$  dan  $G_{best}$  seperti pada langkah sebelumnya. Proses perhitungan akan berhenti ketika iterasi sudah mencapai maksimum atau nilai  $G_{best}$  tidak ada perubahan yang signifikan. Jika kondisi belum optimal, maka ulangi pada langkah menentukan bobot *input* dan bias secara acak. Hasil tersebut menunjukkan bahwa pada iterasi ke-2 proses perhitungan sudah mencapai kondisi optimal, yang artinya bobot dan bias pada iterasi ke-2 yang akan digunakan untuk proses selanjutnya (proses *testing*).

#### 4. Proses Testing

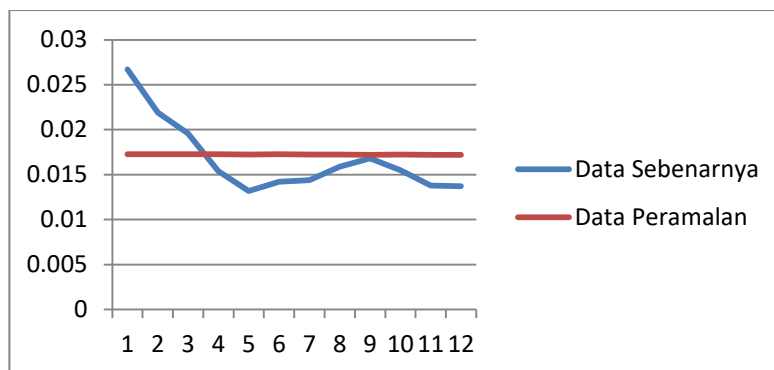
Proses *testing* dilakukan dengan menggunakan data uji. *Flowchart* pengujian PSO-ELM dapat dilihat pada Gambar 2. Langkah pertama yaitu menghitung matriks  $H$

dengan menggunakan bobot *input* dan bias yang sama pada proses *training*. Langkah selanjutnya yaitu menghitung hasil peramalan menggunakan bobot *output* yang diperoleh dari proses *training*. Dari hasil pengujian proses *testing* dengan menggunakan persamaan (6), diperoleh nilai RMSE pada proses *training* sebesar 0,4124935. Langkah terakhir yaitu denormalisasi data yang bertujuan untuk mengembalikan nilai yang telah ternormalisasi ke nilai aslinya supaya memperoleh nilai *output* pada *range* yang sebenarnya.



Gambar 2. Pengujian PSO-ELM

Selanjutnya akan dilihat perbandingan dari data inflasi yang sebenarnya dengan data peramalan bulan April 2020 sampai bulan Maret 2021. Perbandingan dari data inflasi yang sebenarnya dengan data peramalan dapat dilihat grafiknya Gambar 3.



Gambar 3. Grafik Perbandingan Data Peramalan dan Data Sebenarnya.

Sumbu  $x$  menunjukkan bulan ke- $i$  dan sumbu  $y$  menunjukkan skala data inflasi.

Dapat dilihat pada Gambar 3 menunjukkan bahwa hasil peramalan berbeda dengan dengan data sebenarnya (Data inflasi di Indonesia). Oleh karena itu, akan dihitung nilai galat relatif dari masing-masing periode inflasi di Indonesia. Galat mutlak adalah selisih antara data yang sebenarnya dengan data hasil peramalan. Sedangkan galat relatif adalah galat mutlak dibagi dengan nilai sebenarnya. Hasil perhitungan galat



relatif untuk inflasi bulan April 2020 sampai bulan September 2020 dapat ditunjukkan pada Tabel 1.

Tabel 1. Hasil Galat Relatif inflasi di Indonesia

No	Periode	Galat Mutlak	Galat Relatif
1	April 2020	0,00941705	0,35270
2	Mei 2020	0,00461691	0,21082
3	Juni 2020	0,0023145	0,11809
4	Juli 2020	0,00188461	0,12238
5	Agustus 2020	0,0040561	0,30728
6	September 2020	0,00306216	0,21565
7	Oktober 2020	0,00282999	0,19653
8	November 2020	0,00134241	0,08443
9	Desember 2020	0,00041988	0,02499
10	Januari 2021	0,00172518	0,11130
11	Februari 2021	0,00341448	0,24743
12	Maret 2021	0,00349411	0,25504

Dapat dilihat pada tabel 1 menunjukkan bahwa galat relatif hasil peramalan inflasi bulan April 2020 sampai bulan September 2020 menghasilkan nilai yang cukup kecil.

## KESIMPULAN

Hasil peramalan data inflasi di Indonesia untuk bulan April 2020 sampai bulan Maret 2021 menggunakan metode PSO-ELM dengan 1 jumlah neuron di lapisan tersembunyi dan 12 jumlah neuron *input* menyatakan bahwa galat relatif hasil peramalan dengan menggunakan metode PSO-ELM cukup baik, maka metode ini cocok digunakan peramalan data inflasi di Indonesia.

Akurasi data inflasi di Indonesia dengan menggunakan metode PSO-ELM menunjukkan bahwa pada proses *training* diperoleh nilai RMSE sebesar 0,00548793 dengan nilai bobot dan bias yang optimal pada iterasi ke-2, sedangkan pada proses *testing* diperoleh nilai RMSE sebesar 0,4124935. Oleh karena nilai RMSE yang diperoleh cukup kecil, maka metode PSO-ELM cocok digunakan untuk peramalan data inflasi di Indonesia.

## DAFTAR PUSTAKA

- [1] Alfiyati, A. N., Mahmudy, W. F., Ananda, C. F., & Anggodo, Y. P. (2019). Penerapan Extreme Learning Machine (ELM) untuk Peramalan Laju Inflasi di Indonesia. *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIK)*. **6(2)**: 179-186. <http://doi.org/10.25126/jtiik.201962900>

- [2] Amrin. (2014). Peramalan Tingkat Inflasi Indonesia Menggunakan Neural Network Backpropagation Berbasis Metode Time Series. *Jurnal Techo Nusa Mandiri*. **11(2)**: 129-136. <https://doi.org/10.31227/osf.io/7hsp2>
- [3] Handika, P.S., Giriantari, I. A., & Dharma, A. (2016). Perbandingan Metode Extreme Learning Machine dan Particle Swarm Optimization Extreme Learning Machine untuk Peramalan Jumlah Penjualan Barang. *Majalah Ilmiah Teknologi Elektro*. **15(1)**: 84-90. <https://doi.org/10.24843/MITE.2016.v15i01p15>
- [4] Hartati. (2017). Penggunaan Metode ARIMA dalam Meramal Pergerakan Inflasi. *Jurnal Matematika, Sains, dan Teknologi*. **18(1)**: 1-10. <https://doi.org/10.33830/jmst.v18i1.163.2017>
- [5] Huang, G.-B., Zhu, Q.-Y., & Siew, C.-K. (2006). Extreme Learning Machine: A New Learning Scheme of Feedforward Neural Networks. *Neurocomputing*. **70**: 489-501. <https://doi.org/10.1016/j.neucom.2010.02.019>
- [6] Kaloop, M.R., Kumar, D., Samui, P., et all. (2019). Particle Swarm Optimization Algorithm-Extreme Learning Machine (PSO-ELM) Model for Predicting Resilient Modulus of Stabilized Aggregate Bases. *Appl. Sci*. **9(16)**: 1-13. <https://doi.org/10.3390/app9163221>
- [7] Nurmahaludin. (2013). Perancangan Algoritma Belajar Jaringan Syaraf Tiruan Menggunakan Particle Swarm Optimization (PSO). *Jurnal Poros Teknik*. **5(1)**: 18-23. <https://media.neliti.com/media/publications/126356-ID-perancangan-algoritma-belajar-jaringan-s.pdf>
- [8] Panjaitan, M. N., & Wardoyo. (2016). Faktor-Faktor yang Mempengaruhi Inflasi di Indonesia. *Jurnal Ekonomi Bisnis*. **21(3)**: 182-193. <https://media.neliti.com/media/publications/97274-ID-none.pdf>
- [9] Sateria, A., Saputra, I. D., & Dharta, Y. (2018). Penggunaan Metode Particle Swarm Optimization (PSO) pada Optimasi Multirespon Gaya Tekan dan Momen Torsi Penggurdian Material Komposit Glass Fiber Reinforce Polymer (GFRP) yang Ditumpuk dengan Material Stainless Steel (SS). *Jurnal Manutech*. **10(1)**: 1-7. <https://doi.org/10.33504/manutech.v10i01.52>
- [10] Tiruneh, G. G., Fakey, A. R., & Sumati, V. (2020). Neuro-fuzzy System in Construction Engineering and Management Research. *Automation in Construction*. 103348. <https://doi.org/10.1016/j.autcon.20.103348>
- [11] Wahyuddin, S. (2019). Prediksi Inflasi Indonesia Memakai Model ARIMA dan Artificial Neural Network. *Jurnal Tata Kelola dan Kerangka Kerja Teknologi Informasi*. <https://doi.org/10.34010/jtk3ti.v5i1.2297>
- [12] Wahyuningsih, D., Zuhroh, I., & Zainuri. (2008). Prediksi Inflasi Indonesia dengan Model Artificial Neural Network. *Jurnal of Indonesian Applied Economics*. **2(2)**: 113-127. <https://doi.org/10.21776/ub.jiae.2008.002.02.7>
- [13] Kennedy, J., & Eberhart, R. (1995). Particle Swarm Optimization. *IEEE Conference on Neural Networks*. (pp. 1942-1948).
- [14] Fitriah, A., & Abadi, A. M. (2011). Aplikasi Model Neuro Fuzzy Untuk Prediksi Tingkat Inflasi di Indonesia. *Prosiding*. <http://eprints.uny.ac.id/7301/1/t-2.pdf> [diakses 4 Desember 2021]
- [15] Kurniawati, F. (2019). Pengendalian Inflasi dalam Perspektif Ekonomi Islam (Studi Efektivitas Instrumen Moneter Syari'ah di Lampung). *Adzkiya*. <https://e->

journal.metrouniv.ac.id/index.php/adzkiya/article/download/1252/1145  
[diakses 4 Desember 2021]

- [16] Bank Indonesia. (2018). Moneter: Inflasi. <https://www.bi.go.id/id/fungsi-utama/moneter/inflasi/Default.aspx> [\[diakses 15 September 2021\]](#)
- [17] Ilmiyah, R. (2021). Metode Particle Swarm Optimization – Extreme Learning Machine (PSO-ELM) Untuk Peramalan Inflasi di Indonesia. Skripsi.