

## MODEL NEURAL NETWORK BERBASIS PSO DALAM PREDIKSI NILAI TUKAR RUPIAH TERHADAP EURO

**Adani Dharmawati**

Fakultas Teknologi Informasi

Universitas Islam Kalimantan Muhammad Arsyad Al Banjari Banjarmasin

Email : adanidharmawati@uniska-  
bjm.ac.id

### ABSTRAK

Bagi sebuah negara, nilai tukar mata uang merupakan indikator yang sangat penting bagi perekonomian mereka. Tujuan dalam prediksi kurs mata uang yaitu untuk mengetahui nilai tukar mata uang di masa yang akan datang. Setelah hasil prediksi didapatkan, selanjutnya data tersebut akan digunakan dalam menentukan langkah-langkah strategis.

Penelitian ini menghasilkan sebuah model optimasi Neural Network berbasis Particle Swarm Optimization (PSO) yang memiliki kinerja yang paling akurat dalam prediksi nilai tukar Rupiah terhadap Euro dengan nilai RMSE sebesar 93.219 +/- 19.567.

**Kata Kunci:** *prediksi, kurs, Neural Network, Particle Swarm Optimization*

### PENDAHULUAN

Faktanya, tidak ada satu negara pun di dunia ini yang mampu memenuhi semua kebutuhan konsumsi negara dari hasil produksi sendiri, walaupun ada beberapa komoditi yang berlebih dan menjadi komoditi ekspor. Sebab itu, bagi negara yang memiliki komoditi ekspor pasti membutuhkan mata uang asing pada setiap transaksi internasionalnya. Disinilah timbul permasalahan yang cukup pelik yaitu dalam menentukan besaran nilai tukar mata uang dari sebuah negara terhadap negara lain.

Tujuan dalam prediksi kurs mata uang yaitu untuk mengetahui nilai tukar mata uang di masa yang akan datang. Setelah hasil prediksi didapatkan, selanjutnya data tersebut akan digunakan oleh pihak-pihak yang membutuhkannya dalam menentukan langkah-langkah strategis yang dianggap perlu dilakukan dalam meminimalisir kerugian. Contohnya dalam perusahaan multinasional, nilai kurs dapat mempengaruhi keputusan pembiayaan, keputusan investasi, penganggaran modal, dan keputusan lainnya

yang sangat dipengaruhi dengan adanya fluktuasi nilai tukar mata uang.

Didalam dunia pendidikan, sangat banyak ditemukan penelitian tentang prediksi nilai tukar mata uang tersebut. Begitupun dengan metode-metode yang telah digunakan beserta algoritma optimasi yang mampu memprediksi nilai tukar mata uang. Salah satu metode yang sering digunakan dan terbukti memiliki tingkat akurasi tinggi dalam bidang prediksi *time series* yaitu Metode *Neural Network* [1].

*Particle Swarm Optimization* (PSO) merupakan algoritma optimasi yang efektif yang dapat memecahkan masalah yang ada pada algoritma *Neural Network* [2]. *Particle Swarm Optimization* (PSO) memiliki kelebihan dalam memilih fitur dan juga memiliki kinerja yang lebih unggul dalam banyak permasalahan optimasi yang bisa diselesaikan dengan lebih cepat dan juga tingkat konvergensi yang cenderung lebih stabil [3]. Dalam penelitian oleh Kresna Ramanda mencoba melakukan prediksi terhadap angka kelahiran premature di RSUPN Cipto Mangunkusumo. Penelitian

ini bertujuan untuk mencari model dari metode *Neural Network* yang memiliki tingkat akurasi tertinggi. Penelitian ini juga mencoba memasukkan algoritma *Particle Swarm Optimization* (PSO) kedalam *Neural Network*. Dengan menggunakan metode *Neural Network* didapatkan tingkat akurasi yaitu sebesar 94,60%, dan ketika algoritma *Particle Swarm Optimization* (PSO) dimasukkan, maka akurasi mengalami kenaikan sebesar 0,60% menjadi 95,20% [4].

*Particle Swarm Optimization*, disingkat sebagai PSO, didasarkan pada perilaku sebuah kawanan serangga, seperti semut, rayap, lebah atau burung. Algoritma PSO meniru perilaku sosial organisme ini. Perilaku sosial terdiri dari tindakan individu dan pengaruh dari individu-individu lain dalam suatu kelompok [5].

Dalam konteks optimasi multivariabel, kawanan diasumsikan mempunyai ukuran tertentu atau tetap dengan setiap partikel posisi awalnya terletak di suatu lokasi yang acak dalam ruang multidimensi. Setiap partikel diasumsikan memiliki dua karakteristik: posisi dan kecepatan [6].

Berdasarkan penuturan diatas, maka penelitian ini akan mencoba menggunakan *Neural Network* yang dikombinasikan dengan algoritma optimasi yaitu *Particle Swarm Optimization* (PSO) untuk optimasi bobot atribut.

## RUMUSAN MASALAH

Perlu adanya penelitian tentang model optimasi *Neural Network* yang memiliki kinerja yang paling akurat dalam prediksi nilai tukar Rupiah terhadap Euro.

## TUJUAN PENELITIAN

Penelitian ini bertujuan untuk membandingkan akurasi dari berbagai model metode *Neural Network* berbasis

*Particle Swarm Optimization* (PSO) dalam prediksi nilai tukar Rupiah terhadap Euro.

## TARGET LUARAN DAN MANFAAT PENELITIAN

Target luaran dalam pembuatan laporan penelitian ini adalah untuk memfasilitasi tercapainya optimasi *Neural Network* yang memiliki kinerja yang paling akurat dalam prediksi nilai tukar Rupiah terhadap Euro. Adapun manfaat dari penelitian ini adalah sebagai berikut:

### Manfaat Praktis

Membantu mempercepat dalam pembuatan angka prediksi nilai tukar Rupiah terhadap Euro di masa yang akan datang dengan menggunakan model metode *Neural Network* berbasis *Particle Swarm Optimization* (PSO) yang memiliki kinerja yang paling akurat.

### Manfaat Akademis

Hasil penelitian ini diharapkan dapat memberikan sumbangsih ilmu pengetahuan bagi pengembangan teori dan metode yang berkaitan dengan prediksi nilai tukar Rupiah terhadap Euro.

### Manfaat bagi Kebijakan Institusi

Setelah hasil prediksi didapatkan, selanjutnya data tersebut akan digunakan oleh pihak-pihak yang membutuhkannya dalam menentukan langkah-langkah strategis yang dianggap perlu dilakukan agar meminilisir kerugian. Contohnya dalam perusahaan multinasional, nilai kurs dapat mempengaruhi keputusan pembiayaan, keputusan investasi, penganggaran modal, dan keputusan lainnya yang sangat dipengaruhi dengan adanya fluktuasi nilai tukar mata uang.

## METODE PENELITIAN

### Metode Pengumpulan Data

Data yang digunakan adalah data nilai tukar mata uang Rupiah terhadap Euro yang diperoleh secara langsung ataupun tidak langsung, baik bersumber dari dokumentasi, literatur ataupun sumber lainnya. Data yang digunakan pada penelitian ini adalah data nilai tukar mata uang Rupiah terhadap Euro yang didapat dari data publikasi website Bank Indonesia (BI) dari bulan Januari 2011 sampai bulan Desember 2015, yaitu sebanyak 1.227 data.

### Metode Pengolahan Data

Data yang telah diperoleh tersebut akan diterapkan ke dalam beberapa model algoritma *Neural Network* berbasis PSO untuk memprediksi nilai tukar mata uang Rupiah terhadap Euro. Model-model tersebut akan diimplementasikan dengan menggunakan RapidMiner 5.3.015.

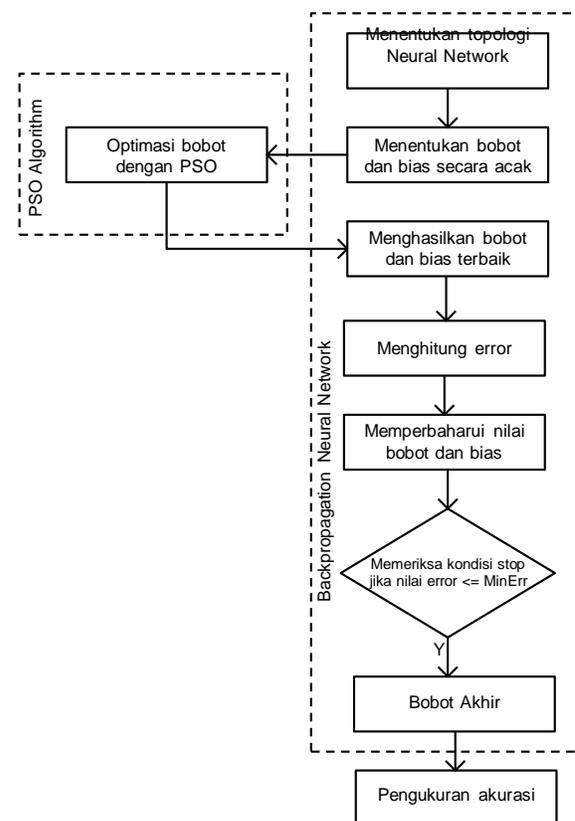
Selanjutnya dilakukan proses *windowing*, *windowing* biasanya digunakan untuk mengubah data *time series* menjadi contoh set yang berisi contoh dengan beberapa atribut yang sesuai dengan poin berurutan. Contoh set ini kemudian dapat digunakan untuk pembentukan model, klasifikasi, atau analisis prediktif. *Windowing* juga dapat digunakan untuk memvisualisasikan data [7].

## ANALISIS HASIL DAN PEMBAHASAN

Model yang diusulkan yaitu algoritma *Neural Network* dengan *Particle Swarm Optimization* (PSO) untuk optimasi bobot atribut. Selanjutnya model tersebut akan diimplementasikan dengan menggunakan RapidMiner 5.3.015. Eksperimen penelitian ini terletak pada pengujian menggunakan data yang identik namun berbeda dalam nilai beberapa

parameter. Nilai dari parameter – parameter ini ditentukan secara coba-coba, kemudian akan dihitung kinerja prediksinya.

Kinerja *Neural Network* tergantung tidak hanya pada variabel input tetapi juga pada ukuran jaringan. Ukuran jaringan yang tidak memadai dan kesalahan pengaturan parameter akan mempengaruhi kecepatan konvergensi dan kualitas prediksi. Pemilihan nilai yang tepat dalam pengaturan parameter, sangat diperlukan karena banyak parameter harus dipilih dan ditetapkan sebelum proses pengujian.



Gambar 1. Model Yang Diusulkan

Setiap algoritma akan diuji dengan beberapa perubahan parameter. Setiap kali hasil akan dicatat dan dihitung nilai rata-ratanya. Berdasarkan nilai rata-rata ini hasil terbaik dari tiap konfigurasi akan ditentukan.

Evaluasi dilakukan dengan menganalisa dan membandingkan hasil prediksi nilai tukar Rupiah terhadap Euro

dari berbagai percobaan dengan mengubah beberapa nilai parameter pada model yang diusulkan untuk mencari konfigurasi parameter tersebut yang menghasilkan tingkat akurasi yang tertinggi.

Pengukuran tingkat akurasi prediksi dilakukan dengan menghitung rata-rata error yang terjadi melalui besaran *Root Mean Square Error* (RMSE).

*Root Mean Square Error* adalah penjumlahan kuadrat error atau selisih antara nilai sebenarnya (aktual) dan nilai prediksi, kemudian membagi jumlah tersebut dengan banyaknya waktu data peramalan dan kemudian menarik akarnya. Jika nilai RMSE semakin kecil maka estimasi model atau variabel tersebut semakin valid. Nilai RMSE dapat dirumuskan sebagai berikut [8].

Semakin kecil nilai akurasi dari masing-masing parameter kinerja ini menyatakan semakin dekat nilai prediksi dengan nilai sebenarnya. Dengan demikian dapat diketahui algoritma yang lebih akurat.

Dalam tahapan ini, parameter yang akan digunakan dan uji yaitu:

**Tabel 1. Parameter Pengujian**

Tools	Parameter	Angka yang diujikan
Neural Network	Momentum	0,1-0,9
	Learning Rate	0,1-0,10
	Training Cycles	10-100
PSO	Hidden Layers Size	2-5
	Maximum number of generation	1-10
	Population Size	1-10
Windowing	Window Size	1-10
X-Validations	Number of Validations	2-10

### Momentum

Dengan menggunakan nilai parameter *Learning Rate* = 0,3, *Training Cycles* = 10, *Hidden Layers Size* = 2, *Maximum number of generation* = 5, *Population Size* = 5, *window size* = 4, *Number of Validations* = 10, berikut merupakan hasil pengujian yang dilakukan dengan melakukan perubahan pada nilai *Momentum*.

**Tabel 2. Hasil Pengujian Momentum**

Momentum	RMSE
0,1	102,105 +/- 23,671
0,2	98,146 +/- 22,541
0,3	97,143 +/- 23,008
0,4	97,280 +/- 22,440
0,5	99,021 +/- 20,766
0,6	98,711 +/- 20,891
0,7	98,194 +/- 25,779
0,8	102,511 +/- 24,074
0,9	128,736 +/- 26,832

Dari pengujian tersebut diatas, dengan menggunakan nilai parameter yang telah ditetapkan, maka diperoleh bahwa *Momentum* = 0,3 yang menghasilkan error paling kecil, Dan untuk pengujian selanjutnya maka nilai untuk parameter *Momentum* akan selalu bernilai 0,3,

### Learning Rate

Dengan menggunakan nilai parameter *Momentum* = 0,3, *Training Cycles* = 10, *Hidden Layers Size* = 2, *Maximum number of generation* = 5, *Population Size* = 5, *window size* = 4, *Number of Validations* = 10, berikut merupakan hasil pengujian yang dilakukan dengan melakukan perubahan pada nilai *Learning Rate*.

**Tabel 3. Hasil Pengujian Learning Rate**

Learning Rate	RMSE
0,1	110,148 +/- 21,045
0,2	100,939 +/- 21,937
0,3	97,143 +/- 23,008
0,4	96,363 +/- 23,579

0,5	99,533 +/- 24,095
0,6	99,601 +/- 24,994
0,7	105,611 +/- 20,818
0,8	100,455 +/- 25,407
0,9	107,698 +/- 27,010
0,10	112,060 +/- 28,515

Dari pengujian tersebut diatas, dengan menggunakan nilai parameter yang telah ditetapkan, maka diperoleh bahwa *Learning Rate* = 0,4 yang menghasilkan error paling kecil, Dan untuk pengujian selanjutnya maka nilai untuk parameter *Learning Rate* akan selalu bernilai 0,4.

**Training Cycles**

Dengan menggunakan nilai parameter *Momentum* = 0,3, *Learning Rate* = 0,4, *Hidden Layers Size* = 2, *Maximum number of generation* = 5, *Population Size* = 5, *window size* = 4, *Number of Validations* = 10, berikut merupakan hasil pengujian yang dilakukan dengan melakukan perbahan pada nilai *Training Cycles*.

**Tabel 4. Hasil Pengujian Training Cycles**

<i>Training Cycles</i>	RMSE
10	96.363 +/- 23.579
20	97.084 +/- 21.891
30	96.628 +/- 19.032
40	96.185 +/- 20.348
50	94.994 +/- 18.853
60	93.905 +/- 19.648
70	94.311 +/- 19.385
80	94.135 +/- 20.122
90	94.134 +/- 19.998
100	94.056 +/- 19.032

Dari pengujian tersebut diatas, dengan menggunakan nilai parameter yang telah ditetapkan, maka diperoleh bahwa *Training Cycles* = 60 yang menghasilkan error paling kecil, Dan untuk pengujian selanjutnya maka nilai untuk parameter *Training Cycles* akan selalu bernilai 60.

**Hidden Layers Size**

Dengan menggunakan nilai parameter *Momentum* = 0,3, *Learning Rate* = 0,4, *Training Cycles* = 60, *Maximum number of generation* = 5, *Population Size* = 5, *window size* = 4, *Number of Validations* = 10, berikut merupakan hasil pengujian yang dilakukan dengan melakukan perbahan pada nilai *Hidden Layers Size*.

**Tabel 5. Hasil Pengujian Hidden Layers Size**

<i>Hidden Layers Size</i>	RMSE
2	92,408 +/- 15,925
3	91,348 +/- 15,071
4	91,262 +/- 14,731
5	92,173 +/- 14,827

Dari pengujian tersebut diatas, dengan menggunakan nilai parameter yang telah ditetapkan, maka diperoleh bahwa *Hidden Layers Size* = 4 yang menghasilkan error paling kecil

**Maximum number of generation**

Dengan menggunakan nilai parameter *Momentum* = 0,3, *Learning Rate* = 0,4, *Training Cycles* = 60, *Hidden Layers Size* = 4, *Population Size* = 5, *window size* = 4, *Number of Validations* = 10, berikut merupakan hasil pengujian yang dilakukan dengan melakukan perbahan pada nilai *Maximum number of generation*.

**Tabel 6. Hasil Pengujian Maximum number of generation**

<i>Maximum number of generation</i>	RMSE
1	93.152 +/- 18.858
2	92.981 +/- 20.265
3	92.474 +/- 19.021
4	92.474 +/- 19.021
5	92.474 +/- 19.021
6	92.474 +/- 19.021
7	92.474 +/- 19.021
8	92.474 +/- 19.021
9	92.474 +/- 19.021

10	92.474 +/- 19.021
----	-------------------

Dari pengujian tersebut diatas, dengan menggunakan nilai parameter yang telah ditetapkan, maka diperoleh bahwa *Maximum number of generation* = 3 yang menghasilkan error paling kecil, Dan untuk pengujian selanjutnya maka nilai untuk parameter *Maximum number of generation* akan selalu bernilai 3.

**Population Size**

Dengan menggunakan nilai parameter *Momentum* = 0,3, *Learning Rate* = 0,4, *Training Cycles* = 60, *Hidden Layers Size* = 4, *Maximum number of generation* = 3, *window size* = 4, *Number of Validations* = 10, berikut merupakan hasil pengujian yang dilakukan dengan melakukan perbahan pada nilai *Population Size*.

**Tabel 7. Hasil Pengujian Population Size**

<b>Population Size</b>	<b>RMSE</b>
1	97.791 +/- 20.022
2	94.367 +/- 20.076
3	93.776 +/- 18.806
4	92.842 +/- 21.176
5	92.474 +/- 19.021
6	93.320 +/- 19.494
7	93.652 +/- 22.643
8	93.019 +/- 19.284
9	93.339 +/- 19.763
10	94.212 +/- 19.930

Dari pengujian tersebut diatas, dengan menggunakan nilai parameter yang telah ditetapkan, maka diperoleh bahwa *Population Size* = 5 yang menghasilkan error paling kecil, Dan untuk pengujian selanjutnya maka nilai untuk parameter *Population Size* akan selalu bernilai 5.

**Window Size**

Dengan menggunakan nilai parameter *Momentum* = 0,3, *Learning Rate*

= 0,4, *Training Cycles* = 60, *Hidden Layers Size* = 4, *Maximum number of generation* = 3, *Population Size* = 5, *Number of Validations* = 10, berikut merupakan hasil pengujian yang dilakukan dengan melakukan perbahan pada nilai *window size*.

**Tabel 8. Hasil Pengujian Window Size**

<b>Window Size</b>	<b>RMSE</b>
1	92.828 +/- 19.999
2	95.005 +/- 17.180
3	93.548 +/- 19.160
4	92.474 +/- 19.021
5	93.700 +/- 18.525
6	92.843 +/- 20.566
7	94.252 +/- 20.552
8	95.275 +/- 20.228
9	93.039 +/- 19.745
10	94.345 +/- 19.521

Dari pengujian tersebut diatas, dengan menggunakan nilai parameter yang telah ditetapkan, maka diperoleh bahwa *window size* = 4 yang menghasilkan error paling kecil.

**Number of Validations**

Dengan menggunakan nilai parameter *Momentum* = 0,3, *Learning Rate* = 0,4, *Training Cycles* = 60, *Hidden Layers Size* = 4, *Maximum number of generation* = 3, *Population Size* = 5, *window size* = 4 berikut merupakan hasil pengujian yang dilakukan dengan melakukan perbahan pada *Number of Validations* = 10.

**Tabel 9. Hasil Pengujian Number of Validations**

<b>Number of Validations</b>	<b>RMSE</b>
2	625.335 +/- 491.839
3	94.645 +/- 16.281
4	93.219 +/- 19.567
5	94.022 +/- 19.485
6	93.905 +/- 19.648
7	94.653 +/- 22.111

8	94.051 +/- 21.178
9	96.043 +/- 21.679
10	95.022 +/- 20.198

Dari pengujian tersebut diatas, dengan menggunakan nilai parameter yang telah ditetapkan, maka diperoleh bahwa *Number of Validations* = 4 yang menghasilkan error paling kecil, Dan untuk pengujian selanjutnya maka nilai untuk parameter *Number of Validations* akan selalu bernilai 4.

## PENUTUP

### Kesimpulan

Dengan menggunakan nilai parameter *Momentum* = 0,3, *Learning Rate* = 0,4, *Training Cycles* = 60, *Hidden Layers Size* = 4, *Maximum number of generation* = 3, *Population Size* = 5, *window size* = 4 dan *Number of Validations* = 4 merupakan model optimasi *Neural Network* berbasis *Particle Swarm Optimization* (PSO) yang memiliki kinerja yang paling akurat dalam prediksi nilai tukar Rupiah terhadap Euro dengan nilai RMSE sebesar 93.219 +/- 19.567.

### Saran

Untuk penelitian selanjutnya diharapkan memperbanyak pelatihan konfigurasi hingga mendapatkan model yang menghasilkan error sekecil-kecilnya. Selain itu juga dapat mengkombinasikan dengan algoritma optimasi lain seperti Ant Colony Optimization (ACO), Adaboost, atau algoritma optimasi lainnya.

## DAFTAR PUSTAKA

[1] R. Hadapiningradja Kusumodestoni and Suyatno, "Prediksi Forex Menggunakan Model *Neural Network*," *Jurnal SIMETRIS*, Vol 6 No 2 November 2015, vol. 6, no. 2, pp. 205-210, November 2015.

- [2] Tae Su Park, Ju Hong Lee, and Bumghi Choi, "Optimization For Artificial *Neural Network* With Adaptive Inertial Weight Of Particle Swarm Optimization," *Cognitive Informatics, IEEE*, pp. 481-485, 2009.
- [3] S. H. Ling, H. T. Nguyen, and K. Y. Chan, "A New Particle Swarm Optimization Algorithm for *Neural Network* Optimization," *Network and System Security, third International Conference*, pp. 516-521, 2009.
- [4] Setyoningsih Wibowo, "Neural Network Dengan Algoritma Genetika Sebagai Pemilihan Fitur Pada Prediksi Loyalitas Pelanggan," *Majalah Ilmiah Pawiyatan*, vol. XXI, pp. 78-91, Oktober 2014.
- [5] E. J. Alba, Jourdan L Garcia Nieto, and E. G. Talbi, "Gene selection in cancer classification using PSO/SVM and GA/SVM hybrid algorithms," *IEEE Congress on Evolutionary Computation*, pp. 284–290, September 2007.
- [6] B. Birge, "A Particle Swarm Optimization (PSO) Primer," 2007.
- [7] A. Chisholm, *Exploring Data with RapidMiner.*, 2013.
- [8] Pang M and Zhao X, "Traffic Flow Prediction of Chaos *Time series* by Using Subtractive Clustering for Fuzzy *Neural Network* Modeling," 2008 *Second International Symposium on Intelligent Information Technology Application*, pp. 23-27, Desember 2008.