

# Penentuan Arsitektur Terbaik Model NAR-NN untuk Peramalan Kasus Covid-19

Qonita Ilmi Awal<sup>1</sup>, Dian Anggraeni<sup>2</sup>, Alfian Futuhul Hadi<sup>3\*</sup>

<sup>123</sup>Prodi Matematika, Fakultas Matematika dan Ilmu Pengetahuan Alam,  
Universitas Jember, Jember, Indonesia

\* Corresponding author, email: [afhadi.fmipa@unej.ac.id](mailto:afhadi.fmipa@unej.ac.id)

## Abstract

The NAR-NN model will be applied in time series forecasting, namely data on confirmed cases of Covid-19 in East Kalimantan Province. The use of time series data as the basis for forecasting so that it can recognize patterns that occur which can then be used as a reference to predict the number of cases that will occur. This research data is 300 daily data for the time period from October 23, 2020 to August 18, 2021, which follows a nonlinear pattern and experiences an upward trend. In this study, the best architecture was determined for the NAR-NN model using the sigmoid activation function and the Levenberg-Marquadt Backpropagation training algorithm. The NAR-NN architecture consists of three layers, namely the input layer, the hidden layer, and the output layer. The evaluation model used is the Mean Absolute Percentage Error (MAPE). The results of this study by experimenting with the number of hidden neurons showed that the model with the best architecture at the time of delay was 4 and the number of hidden neurons was 8 with the MAPE value forecast with actual data of 7.5083%.

**Keywords:** NAR-NN, forecasting, time series, nonlinear, MAPE.

## Abstrak

Model NAR-NN akan diterapkan dalam peramalan deret waktu yaitu data kasus terkonfirmasi Covid-19 di Provinsi Kalimantan Timur. Penggunaan data deret waktu sebagai dasar peramalan sehingga bisa mengenali pola yang terjadi yang kemudian pada waktunya dapat dijadikan acuan untuk meramalkan jumlah kasus yang akan terjadi. Data penelitian ini merupakan data harian sebanyak 300 data pada periode waktu 23 Oktober 2020 hingga 18 Agustus 2021 yang mengikuti pola nonlinear dan mengalami tren naik. Dalam penelitian ini dilakukan penentuan arsitektur terbaik model NAR-NN yang menggunakan fungsi aktivasi sigmoid dan algoritma pelatihan Levenberg-Marquadt Backpropagation. Arsitektur NAR-NN terdiri atas tiga lapisan, yaitu lapisan input, lapisan tersembunyi, dan lapisan output. Evaluasi model yang digunakan adalah Mean Absolute Percentage Error (MAPE). Hasil penelitian ini dengan percobaan banyaknya hidden neuron diperoleh bahwa model dengan arsitektur terbaik pada saat delay 4 dan jumlah hidden neuron sebanyak 8 buah dengan nilai MAPE hasil ramalan dengan data aktual sebesar 7.5083%.

**Kata Kunci:** NAR-NN, peramalan, deret waktu, nonlinear, MAPE.

## 1. Pendahuluan

Ilmu statistika dapat diterapkan dalam suatu peramalan kasus terkonfirmasi Covid-19. Peramalan merupakan pengolahan data masa lalu dengan metode matematika untuk memperkirakan nilai atau kejadian di masa mendatang. Permasalahan peramalan seringkali melibatkan penggunaan data deret waktu, yaitu data yang diamati dan dicatat

dalam periode waktu tertentu. Pada analisis deret waktu, terdapat suatu proses dimana nilai data yang berpengaruh adalah nilai data masa lalunya, proses yang terjadi disebut autoregressive (AR). Dalam suatu model deret waktu, terdapat parameter yang didalamnya memiliki sebuah nilai dimana nilai itu yang akan menentukan persamaan atau arsitektur dari model digunakan untuk peramalan nantinya.

Banyak data deret waktu cenderung mengandung komponen nonlinear sehingga kurang cocok jika yang digunakan adalah metode linear. Oleh karena itu, dibutuhkan model baru yang cocok untuk digunakan pada data yang nonlinear tersebut. Terdapat beberapa model yang nonlinear, seperti STAR (Smooth Transition Autoregressive), ARCH (Autoregressive Conditional Heteroskedasticity), dan NAR (Nonlinear Autoregressive).

Sejak adanya artikel penelitian oleh [1] dan penelitian oleh [2] model NAR menjadi pemodelan nonlinear yang dapat dipilih untuk digunakan dalam terapan di bidang kesehatan, salah satunya untuk peramalan kasus Covid-19. Model NAR merupakan pengembangan dari model AR, dimana model AR merupakan model regresi yang menggabungkan nilai yang diamati dengan nilai sebelumnya pada interval waktu tertentu. Kemudian untuk keperluan evaluasi dikembangkan model baru yaitu model NAR dengan neural network, yang dinamakan Nonlinear Autoregressive-Neural Network (NAR-NN).

Berdasarkan peninjauan penelitian terdahulu dan pengujian nonlinearitas pada data kasus terkonfirmasi Covid-19 yang menunjukkan adanya pola data nonlinear, NAR-NN dipilih sebagai model untuk mendasari peramalan pada penelitian ini. Penentuan arsitektur terbaik yang digunakan untuk peramalan dalam model ini menggunakan evaluasi model Mean Squared Error (MSE). Sedangkan, peramalan yang dilakukan dalam penelitian ini menggunakan evaluasi model Mean Absolute Percentage Error (MAPE). Adapun permasalahan yang dirumuskan dalam artikel ini adalah bagaimana penentuan arsitektur yang dapat digunakan untuk peramalan kasus Covid-19 di Provinsi Kalimantan Timur.

## **2. Material dan Metode**

### **2.1 Model Nonlinear Autoregressive Neural Network**

Menurut [3], suatu proses dikatakan proses autoregressive dengan orde  $n$ , disingkat  $AR(n)$  yang direpresentasikan oleh persamaan (1). Jika  $F(X_{t-1})$  merupakan suatu fungsi linear, prosesnya disebut AR linear. Sedangkan, jika  $F(X_{t-1})$  merupakan fungsi nonlinear, maka merupakan AR nonlinear.

$$x_t = F(X_{t-1}) + \varepsilon_t \quad (1)$$

NAR-NN adalah jaringan feed-forward yang merupakan pendekatan fungsi nonlinear  $h(\cdot)$  yang didefinisikan sebagai:

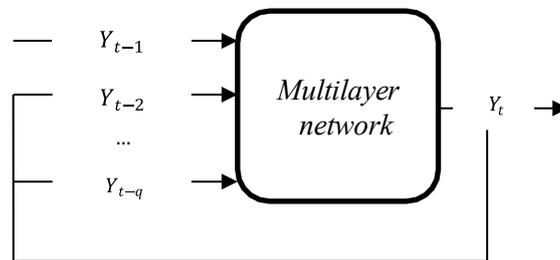
$$\hat{y}_t = \hat{h}(y_{t-1}, \hat{y}_{t-2}, \dots, y_{t-p}) \quad (2)$$

Saat menggunakan jaringan NAR, jaringan *close loop* digunakan untuk melakukan prediksi *multistep-ahead*. *Output* dari jaringan *close loop* NAR adalah:

$$\hat{y}_{t+p} = f(y_{t-1} + \hat{y}_{t-2} + \dots + y_{t-p}) \quad (3)$$

dimana  $p$  merepresentasikan tahapan peramalan di masa depan.

Model NAR-NN menggunakan model NAR yang fungsi autoregressive-nya diperoleh menggunakan pendekatan NN. Menurut [4], NAR-NN memiliki kemampuan prediksi yang sangat baik untuk sinyal periodik dengan model data nonlinear dan nonstasioner. Menurut [5], topologi NAR divisualisasikan pada Gambar 1.



**Gambar 1.** Topologi NAR

Adapun nilai  $q$  merupakan parameter waktu tunda pada nilai masa lalu yaitu  $Y_{t-1}, Y_{t-2}, \dots, Y_{t-q}$  yang disebut juga sebagai feedback delays. Jumlah layer dan hidden neuron tiap layer sangat fleksibel dan dioptimalkan melalui prosedur trial and error untuk mendapatkan topologi yang menghasilkan kinerja terbaik. Optimalisasi arsitektur ditujukan untuk mengurangi sebanyak mungkin jumlah bobot dan neuron agar nantinya akan mengurangi kompleksitas jaringan, meningkatkan waktu komputasi, dan mempertahankan generalisasi kemampuan.

Menurut [6], model NAR secara umum ditulis sebagai persamaan berikut ini:

$$Y_t = F(Y_{t-1}, Y_{t-2}, \dots, Y_{t-q}) + \varepsilon_t \quad (4)$$

dimana formula tersebut merepresentasikan penggunaan NAR untuk memprediksi nilai dari deret waktu  $Y$  pada waktu  $t$  ( $Y_t$ ) menggunakan nilai-nilai  $q$  dari deret data sebelumnya. Dengan  $Y_{t-i}$  adalah data yang diamati ke- $(t - i)$  untuk  $i = 1, 2, \dots, q$ ,  $F$  adalah fungsi *autoregressive nonlinear* yang tidak diketahui sebelumnya sehingga

memerlukan pelatihan dari NN untuk mendekati fungsi tersebut dengan menggunakan bobot dan bias *neuron*,  $q$  adalah orde model NAR, dan  $\epsilon_t$  adalah pendekatan kesalahan dari deret  $Y$  pada waktu ke- $t$ .

Algoritma pelatihan yang digunakan pada penelitian ini adalah *Levenberg-Marquardt Backpropagation* (LMBP) yang merupakan salah satu jenis metode pelatihan *backpropagation* yang bersifat cepat secara komputasinya dan memiliki konvergensi yang stabil. Metode LMBP dirancang untuk mendekati orde kedua derivatif tanpa menghitung matriks Hessian, sehingga meningkatkan kecepatan proses pelatihan. Ketika fungsi kinerja memiliki bentuk jumlah kuadrat pada jaringan pelatihan *feedforward*, maka matriks Hessian dapat didekati seperti yang ditunjukkan dalam persamaan (5) dan gradient dapat dihitung seperti yang dijelaskan dalam persamaan (6) berikut:

$$H = J^T J \quad (5)$$

$$g = J^T e \quad (6)$$

Dalam persamaan (5) dan (6),  $J$  merupakan matriks Jacobian yang berisi turunan pertama dari jaringan kesalahan yang berhubungan dengan bobot dan bias, sementara  $e$  merupakan vektor kesalahan jaringan di semua sampel pelatihan. Metode ini memanfaatkan matriks Jacobian untuk perhitungannya, dengan asumsi bahwa kinerja rata-rata berdasarkan dari jumlah kesalahan kuadrat. Oleh karena itu, jaringan menggunakan *Mean Squared Error* (MSE) yang dinyatakan dalam persamaan (7), dimana  $y_i$  singkatan dari sampel data ke- $i$ ,  $\hat{y}_i$  adalah data yang diprediksikan diperoleh oleh jaringan untuk nilai  $y_i$  dan  $n$  adalah jumlah sampel data untuk pelatihan jaringan.

$$MSE = \frac{\sum_{i=1}^n (\hat{y}_i - y_i)^2}{n} \quad (7)$$

Persamaan NN yang menggunakan metode pelatihan *backpropagation* dengan satu *hidden layer* ditulis sebagaimana bentuk persamaan (8) berikut:

$$\hat{Y}_t = f^0 \left( b_2 + \sum_{j=1}^r w_j f_j^h \left( b_{1,j} + \sum_{i=1}^q w_{i,j} Y_{t,i} \right) \right) \quad (8)$$

dengan  $\hat{Y}_t$  adalah *output* pada waktu ke- $t$ , fungsi aktivasi *neuron* pada *output layer* dinotasikan dengan  $f^0$ ,  $b_2$  merupakan notasi untuk bobot bias antara *hidden layer* ke *output layer*,  $w_j$  adalah bobot penghubung *neuron* pada *hidden layer* dengan *neuron* pada *output layer*,  $f^h$  adalah notasi untuk fungsi aktivasi *neuron* ke- $j$  pada *hidden*

layer,  $b_{1,j}$  adalah bobot bias antara *input layer* dengan *hidden neuron* ke- $j$ ,  $w_{i,j}$  adalah bobot yang menghubungkan *neuro* ke- $i$  pada *input layer* dengan *hidden neuron* ke- $j$ ,  $Y_{t,i}$  adalah *input* ke- $t$ ,  $r$  adalah banyaknya *hidden neuron*, dan  $q$  adalah banyaknya *input*. Persamaan (4) dapat ditulis sebagai:

$$Y_t = \hat{Y}_t + \varepsilon_t \quad (9)$$

Misalkan  $Y_{t-1}, Y_{t-2}, \dots, Y_{t-q}$  adalah *input* pada NN sehingga  $Y_{t,i}$  menjadi  $Y_{t-i}$  untuk  $i = 1, 2, \dots, q$ , selanjutnya  $q$  disebut sebagai *delay* pada NN sehingga persamaan (8) disubstitusikan ke persamaan (9):

$$Y_t = f^0 \left( b_2 + \sum_{j=1}^r w_j f_j^h \left( b_{1,j} + \sum_{i=1}^q w_{i,j} Y_{t-i} \right) \right) + \varepsilon_t \quad (10)$$

Menurut Julpan dkk (2018), fungsi aktivasi yang digunakan pada *hidden layer* untuk NN yang dilatih metode *backpropagation* yaitu fungsi aktivasi sigmoid biner atau sigmoid bipolar, dalam penelitian ini akan menggunakan fungsi aktivasi sigmoid bipolar yang persamaannya ditulis sebagai:

$$f_j^h(x) = \frac{2}{1 + \exp(-1)} \quad (11)$$

Fungsi aktivasi yang digunakan pada lapisan *output* adalah fungsi *linear*

$$f^0(x) = x \quad (12)$$

Persamaan (11) dan (12) disubstitusikan pada persamaan (10), sehingga diperoleh model NAR-NN yang dirumuskan sebagai:

$$Y_t = b_2 + \sum_{j=1}^r w_j \left( \frac{2}{1 + \exp(-b_{1,j} - \sum_{i=1}^q w_{i,j} Y_{t-i})} - 1 \right) + \varepsilon_t \quad (13)$$

## 2.2 Evaluasi Model

Adapun evaluasi kebaikan model yang digunakan untuk peramalan adalah MAPE. *Mean Absolute Percentage Error* (MAPE) memiliki kemampuan yang lebih baik dibandingkan dengan MSE dalam menghitung *error*. Nilai MAPE ditulis sebagai formula berikut:

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{Y_t - \hat{Y}_t}{Y_t} \right| \times 100\% \quad (14)$$

dengan  $\hat{Y}_t$  adalah data pendekatan ke- $t$  dan  $n$  merupakan banyaknya data. Kriteria MAPE ditunjukkan pada Tabel 1.

**Tabel 1.** Kriteria Nilai MAPE

Nilai MAPE	Kriteria
<b>MAPE &lt; 10%</b>	Sangat Baik
<b>10% &lt; MAPE &lt; 20%</b>	Baik
<b>20% &lt; MAPE &lt; 50%</b>	Cukup
<b>MAPE &gt; 50%</b>	Buruk

### 2.3 Metode Penelitian

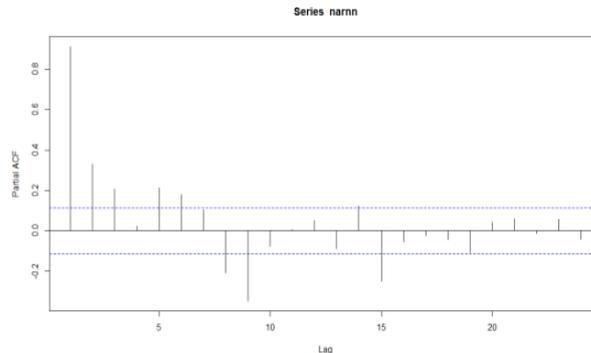
Data yang digunakan adalah data kasus terkonfirmasi Covid-19 di Provinsi Kalimantan Timur sejak 23 Oktober 2020 hingga 18 Agustus 2021 sebanyak 300 hari. Data tersebut merupakan data sekunder yang diperoleh dari website kaggle.com dimana data utama terdapat pada website covid19.go.id. Program komputer yang digunakan untuk membantu proses analisis data adalah Matlab R2015b. Tahapan analisis data yang dilakukan dalam penelitian ini yaitu:

1. Pembagian data menjadi dua bagian yaitu data *training* sebanyak 210 data dan data *testing* sebanyak 90 data.
2. Perancangan arsitektur model NAR-NN dengan menentukan parameter tetap dan parameter yang diubah-ubah.
3. Proses pelatihan untuk memperoleh jaringan terbaik melalui prosedur *trial and error* dengan menggunakan MSE.
4. Proses pengujian untuk melakukan peramalan selama 15 hari ke depan yaitu tanggal 19 Agustus 2021 hingga 2 September 2021.
5. Evaluasi model untuk memastikan kebaikannya dalam peramalan dengan MAPE.

### 3. Hasil dan Diskusi

NAR-NN yang digunakan dalam penelitian ini memiliki parameter utama yaitu satu *hidden layer* dengan *neuron* sebanyak 1 hingga 10, *feedback delays* dengan nilai tetap, fungsi sigmoid untuk *hidden layer* dan fungsi *linear* untuk *output layer*, dan algoritma pelatihan *Levenberg-Marquadt*. Jumlah *feedback delays* ditentukan dengan memeriksa fungsi autokorelasi parsial dari data yaitu dengan memilih nilai lag yang

sesuai dengan maksimum lag korelasi. *Feedback delays* yang digunakan bernilai 4 sebagaimana digambarkan oleh plot PACF pada Gambar 2, yang berarti model NAR-NN menggunakan empat data periode sebelumnya sebagai *input*.



**Gambar 2.** Plot PACF Data

Tahapan setelah menentukan parameter yang digunakan adalah proses pelatihan dengan jumlah data sebanyak 210 data. Data pelatihan tersebut dinormalisasi menggunakan metode *min-max* ke dalam interval  $[0.1, 0.9]$ . Pengembangan arsitektur yang optimal untuk model NAR-NN memerlukan penentuan *feedback delays*, jumlah *hidden neuron*, dan algoritma pelatihan yang efisien. Jumlah *hidden neuron* yang optimal diperoleh melalui prosedur *trial and error*. Selanjutnya, algoritma LM digunakan untuk pelatihan model NAR-NN dan kinerjanya dievaluasi di bawah struktur NN yang optimal. Dalam penelitian ini, kinerja model dievaluasi dengan *Mean Squared Error* (MSE) antara nilai aktual dan nilai prediksi. Tabel 2 menunjukkan hasil MSE dari pelatihan model NAR-NN yang dilatih dengan 10 arsitektur yang berbeda. Analisis error menunjukkan bahwa model NAR-NN dengan 8 hidden neuron memberikan kinerja terbaik, yaitu dengan nilai MSE 0.00148658.

**Tabel 2.** Training Arsitektur NAR-NN

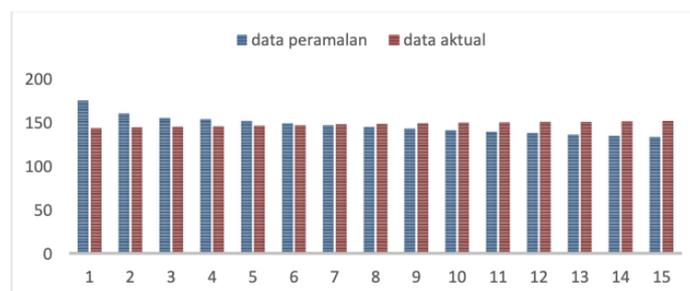
Hidden Neuron	MSE
1	0.00356725
2	0.00418190
3	0.00305810
4	0.00344864
5	0.00231013
6	0.00196741
7	0.00225837
<b>8</b>	<b>0.00148658</b>
9	0.00274220
10	0.00253634

Proses pelatihan di dalam jaringan *open loop* telah selesai, selanjutnya jaringan ditransformasikan menjadi jaringan *close loop*. Model terbaik diperoleh saat delay 4 dan jumlah hidden neuron sebanyak 8 akan diuji menggunakan data baru. Data ini merupakan data yang belum dikenali oleh model tersebut yaitu data *testing*. Hasil yang diperoleh dari pengujian ini yaitu nilai MAPE sebesar 15.76% yang berarti “baik” sesuai kriteria pada Tabel 1. Tahap terakhir yaitu melakukan peramalan dengan menggunakan model NAR-NN yang optimal. Hasil peramalan yang dilakukan yaitu untuk 15 data berikutnya disajikan pada Tabel 3.

**Tabel 3.** Hasil Peramalan NAR-NN

<b>Data ke-</b>	<b>Data Normalisasi</b>	<b>Data Denormalisasi</b>	<b>Data Aktual</b>
301 (19 Agustus 2021)	0.155088668	174.891	143.534
302 (20 Agustus 2021)	0.150117618	160.283	144.455
303 (21 Agustus 2021)	0.148382825	155.185	145.112
304 (22 Agustus 2021)	0.147867502	153.670	145.711
305 (23 Agustus 2021)	0.147170918	151.623	146.198
306 (24 Agustus 2021)	0.146301629	149.068	146.989
307 (25 Agustus 2021)	0.145565560	146.905	147.854
308 (26 Agustus 2021)	0.144869755	144.860	148.491
309 (27 Agustus 2021)	0.144203722	142.903	149.476
310 (28 Agustus 2021)	0.143572588	141.048	149.807
311 (29 Agustus 2021)	0.142977762	139.300	150.150
312 (30 Agustus 2021)	0.142415959	137.649	150.698
313 (31 Agustus 2021)	0.141886110	136.092	150.698
314 (1 September 2021)	0.141387102	134.626	151.235
315 (2 September 2021)	0.140917713	133.246	151.738

Program yang dijalankan menghasilkan ramalan dengan bentuk data normalisasi, lalu dikembalikan ke *range* data aktualnya. Setelah dilakukan denormalisasi, data ramalan dengan data aktual dievaluasi dengan menggunakan MAPE. Evaluasi tersebut menghasilkan model yang sangat baik untuk peramalan yaitu sebesar 7.51%. Ilustrasi data hasil peramalan dengan model NAR-NN dibandingkan dengan data aktualnya disajikan pada Gambar 3.



**Gambar 3.** Visualisasi Data Peramalan dengan Data Aktual

#### 4. Kesimpulan

Berdasarkan analisis yang telah dilakukan dapat ditarik kesimpulan bahwa diperoleh model NAR-NN dengan arsitektur terbaik untuk data kasus harian terkonfirmasi Covid-19 di Provinsi Kalimantan Timur diperoleh pada saat *delay* 4 dan sebanyak 8 *hidden neuron*. Perhitungan error menggunakan MAPE untuk data ramalan dan data aktual didapatkan nilai sebesar 7.51%.

#### Daftar Pustaka

- [1] Swaraj, A., Verma, K., Kaur, A., Singh, G., Kumar, A., & Sales, L. M. Implementation of Stacking Based ARIMA Model for Prediction of Covid-19 Cases in India. *Journal of Biomedical Informatics*, 2020.
- [2] Khan, Farhan M., & Gupta, R. ARIMA and NAR Based Prediction Model for Time Series Analysis of Covid-19 in India. *Journal of Safety Science and Resilience*, 1, 12-18, 2020.
- [3] Dietz, S. *Autoregressive Neural Network Processes Univariate, Multivariate, and Cointegrated Models with Application to the German Automobile Industry*. PhD Thesis. Passau University, 2010.
- [4] Jiang, L., Li, Y., Tian, Y., & Zhou, Q. *A Novel Method Based on Nonlinear Auto-Regression Neural Network and Convolutional Neural Network for Imbalanced Fault Diagnosis of Rotating Machinery*. China: Wuhan University, 2020.
- [5] Ruiz, Luis, G. B., Cuellar, Manuel P., Flores, Miguel, D., & Jimenez, Carmen, P. An Application of Non-Linear Autoregressive Neural Network to Predict Energy Consumption in Public Buildings. *Energies*, 4, 2016.
- [6] Ismarani, Saputro, Dewi, R. S., & Setiyowati, R. Pemodelan Banyaknya Penumpang Kereta Api di Pulau Jawa dengan Nonlinear Autoregressive Neural Network. *Prosiding Seminar Nasional Matematika*, 4, 646, 2021.