



Prediksi nilai tukar petani di Jawa Tengah dengan metode *Hybrid Autoregressive Integrated Moving Average* dan *Neural Network (Hybrid ARIMA-NN)*

Purwaningsih*, Universitas Negeri Yogyakarta

Dhoriva Urwatul Wutsqa, Universitas Negeri Yogyakarta

*e-mail: purwaningsih.2019@student.uny.ac.id

Abstrak. Nilai Tukar Petani (NTP) merupakan salah satu indikator untuk mengukur tingkat kesejahteraan petani. Penelitian ini bertujuan untuk memperoleh model *hybrid* ARIMA-NN dalam memprediksi NTP di Jawa Tengah serta mengetahui hasil prediksi NTP pada periode berikutnya. Data yang digunakan merupakan data bulanan NTP Jawa Tengah periode Januari 2013 sampai Desember 2021 yang diperoleh dari publikasi Badan Pusat Statistik Provinsi Jawa Tengah. Analisis dilakukan melalui pemodelan ARIMA untuk menangkap pola linear, kemudian *residual* model ARIMA dimodelkan menggunakan Neural Network dengan algoritma *backpropagation* untuk menangkap pola nonlinear. Hasil penelitian menunjukkan bahwa model *hybrid* ARIMA(0,1,2)-NN(2,5,1) layak digunakan karena menghasilkan MAPE *training* sebesar 0,5006% dan MAPE *testing* sebesar 0,5928%, serta *residual* model telah memenuhi asumsi *white noise*. Prediksi NTP tahun 2022 menghasilkan MAPE sebesar 1,36%, dengan nilai prediksi tertinggi pada Desember 2022 sebesar 105,29 dan terendah pada Januari 2022 sebesar 103,69.

Kata Kunci: NTP, hybrid, ARIMA, NN

Abstract. Farmers' Terms of Trade (NTP) is one of the indicators used to measure farmers' welfare. This study aims to obtain a hybrid ARIMA-NN model for predicting NTP in Central Java and to determine the forecasted NTP values for the following period. The data used in this study are monthly NTP data for Central Java from January 2013 to December 2021 obtained from publications of the Central Java Statistics Agency. The analysis was conducted by applying the ARIMA model to capture linear patterns, while the residuals from the ARIMA model were then modeled using a Neural Network with the backpropagation algorithm to capture nonlinear patterns. The results show that the hybrid ARIMA(0,1,2)-NN(2,5,1) model is feasible because it produces a training MAPE of 0.5006% and a testing MAPE of 0.5928%, and the residuals satisfy the white noise assumption. The 2022 NTP prediction produces a MAPE of 1.36%, with the highest predicted value in December 2022 at 105.29 and the lowest in January 2022 at 103.69.

Keywords: NTP, hybrid, ARIMA, NN

PENDAHULUAN

Pertanian merupakan salah satu sektor penting bagi negara agraris seperti Indonesia. Sektor pertanian tidak hanya berperan sebagai penyedia kebutuhan pangan, tetapi juga

memberikan kontribusi terhadap perekonomian nasional. Oleh karena itu, perkembangan sektor pertanian perlu diimbangi dengan perhatian terhadap kesejahteraan petani sebagai pelaku utama dalam kegiatan produksi pertanian.

Kesejahteraan petani dapat dilihat melalui beberapa indikator, salah satunya adalah Nilai Tukar Petani (NTP). Menurut Badan Pusat Statistik, NTP merupakan rasio antara indeks harga yang diterima petani (It) dan indeks harga yang dibayarkan petani (Ib). Secara umum, NTP menggambarkan kemampuan daya beli petani dari hasil produksi pertanian terhadap kebutuhan produksi dan konsumsi rumah tangganya. Apabila NTP mengalami penurunan, maka hal tersebut dapat menjadi sinyal awal adanya tekanan terhadap kesejahteraan petani.

Jawa Tengah merupakan salah satu provinsi dengan aktivitas pertanian yang cukup besar. Luas lahan pertanian dan produksi beras yang tinggi menunjukkan bahwa sektor pertanian memiliki peran penting bagi perekonomian daerah. Namun, nilai NTP di Jawa Tengah cenderung mengalami fluktuasi dari waktu ke waktu. Kondisi ini menunjukkan bahwa diperlukan informasi prediktif mengenai perkembangan NTP agar pemerintah dan pihak terkait dapat mengantisipasi potensi penurunan daya beli petani melalui kebijakan yang lebih tepat.

Perkembangan metode peramalan memungkinkan penggunaan lebih dari satu pendekatan untuk memperoleh hasil prediksi yang lebih baik. Metode ARIMA banyak digunakan pada data runtun waktu karena mampu menangkap pola linear berdasarkan hubungan data masa lalu. Namun, data ekonomi dan sosial, termasuk NTP, tidak selalu hanya memiliki pola linear. Oleh karena itu, Neural Network dapat digunakan sebagai pelengkap karena memiliki kemampuan dalam menangkap pola nonlinear. Kombinasi ARIMA dan Neural Network dalam bentuk *hybrid* diharapkan dapat memanfaatkan keunggulan keduanya, yaitu ARIMA untuk komponen linear dan Neural Network untuk komponen nonlinear.

Metode *hybrid* ARIMA-NN telah diterapkan pada berbagai bidang. Karo (2021) menggunakan *hybrid* ARIMA-NN untuk memprediksi penyebaran demam berdarah dan memperoleh nilai error yang rendah. Purbasari (2020) menggunakan metode *hybrid* ARIMA-NN untuk memprediksi indeks harga konsumen. Sari dan Maulidany (2020) menerapkan *hybrid* ARIMA-NN untuk meramalkan rata-rata kecepatan angin di Lombok Timur. Selain itu, Naveena et al. (2017) menunjukkan bahwa *hybrid* ARIMA-NN dapat memberikan hasil yang baik dalam peramalan harga kopi robusta karena menggabungkan struktur linear dan nonlinear pada data.

Penelitian yang menggunakan metode *hybrid* ARIMA-NN untuk memprediksi Nilai Tukar Petani masih relatif terbatas. Berdasarkan hal tersebut, penelitian ini dilakukan untuk memprediksi NTP di Jawa Tengah menggunakan metode *hybrid* ARIMA-NN. Penelitian ini diharapkan dapat memberikan informasi mengenai pola perkembangan NTP serta menjadi salah satu pertimbangan bagi pihak terkait dalam memantau kesejahteraan petani.

METODE

Data

Data yang digunakan dalam penelitian ini adalah data sekunder berupa data bulanan Nilai Tukar Petani (NTP) di Jawa Tengah yang diperoleh dari publikasi resmi Badan Pusat Statistik Provinsi Jawa Tengah melalui laman <https://jateng.bps.go.id>. Periode data yang digunakan adalah Januari 2013 sampai dengan Desember 2021, sehingga jumlah observasi sebanyak 108 data. Data tersebut dibagi menjadi data *training* dan data *testing* dengan perbandingan 80% dan 20%. Karena data yang digunakan merupakan data runtun waktu,

pembagian data dilakukan secara kronologis, yaitu data awal digunakan sebagai data *training* dan data periode berikutnya digunakan sebagai data *testing*. Analisis dilakukan dengan metode *hybrid* ARIMA-NN menggunakan bantuan perangkat lunak RStudio untuk pemodelan ARIMA dan MATLAB untuk pemodelan Neural Network.

Model Autoregressive Integrated Moving Average (ARIMA)

Metode *Autoregresif Integrated Moving Average* (ARIMA) dikembangkan oleh George Box dan Gwilym Jenkins sehingga sering juga disebut sebagai metode runtun waktu *Box-Jenkins*. Model *Autoregresif Integrated Moving Average* (ARIMA) adalah model yang secara penuh mengabaikan variabel independen dalam membuat peramalan. ARIMA menggunakan nilai masa lalu dan sekarang dari variabel dependen untuk menghasilkan peramalan jangka pendek yang akurat. Model ARIMA memiliki ordo p, d, dan q yang dinotasikan sebagai ARIMA(p,d,q). persamaan yang terbentuk menggunakan operator *Backward shift* oleh B dinyatakan sebagai berikut (Wei, 2006):

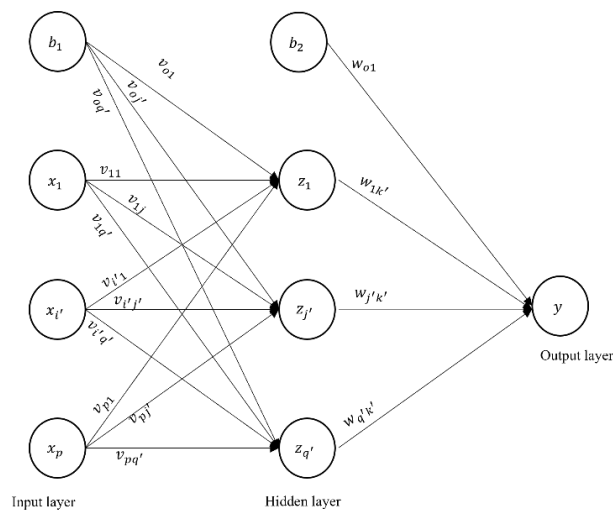
$$\phi(B)x_t = \phi^p(B)\nabla^d x_t = \theta_0 + \theta^q(B)a_t \tag{1}$$

$$(1 - \phi_1 B - \phi_2 B^2 - \dots - \phi_p B^p)(1 - B)^d x_t = \theta_0 + (1 - \theta_1 B - \theta_2 B^2 - \dots - \theta_q B^q)a_t \tag{2}$$

Neural Network

Artificial neural network (ANN) atau yang sering disebut juga jaringan saraf tiruan (JST) merupakan salah satu metode *artificial intelligence* yang memiliki konsep meniru sistem jaringan saraf biologis pada manusia. Dikatakan meniru sistem saraf biologis manusia dikarenakan jaringan ini terbentuk dari sebagian besar neuron yang memiliki hubungan erat dari satu neuron dengan neuron lainnya (Siang, 2009). *Backpropagation* merupakan salah satu metode *neural network* atau jaringan saraf tiruan yang merupakan algoritma pembelajaran terawasi (*supervised learning*) yang mampu menyesuaikan bobot sesuai target dan *output* guna memperkecil tingkat suatu eror. Dalam metode *backpropagation* algoritma yang dikerjakan adalah inialisasi bobot, komputasi *forward* dan *backpropagation* dan inialisasi kondisi stopping berdasarkan nilai batas error atau jumlah batas epoch. Menurut Fausett (1994) terdapat tiga jenis model arsitektur jaringan yakni sebagai berikut:

1. Arsitektur *Backpropagation Neural Network*



Gambar 1. Arsitektur jaringan *backpropagation*

Gambar 1 merupakan arsitektur jaringan *backpropagation neural network* dengan p merupakan banyak neuron input dengan $1 < i' < p$ ditambah 1 bias, dan q' merupakan neuron tersembunyi dengan $1 < j' < q'$ ditambah 1 bias dan juga 1 *output* layer dengan, $x_{i'}$ adalah nilai neuron input ke- i' ($i' = 1, 2, \dots, p$), dengan p merupakan banyak neuron input, $z_{j'}$ merupakan nilai neuron tersembunyi ke- j' ($j' = 1, 2, 3, \dots, q'$), q' merupakan banyak neuron tersembunyi, y merupakan nilai variabel *output*, $v_{i'j'}$ merupakan bobot dari neuron input ke- i' menuju neuron tersembunyi ke- j' , $v_{0j'}$ merupakan bobot bias pada lapisan tersembunyi, $w_{j'1}$ merupakan bobot dari neuron tersembunyi ke- j' menuju neuron *output*, dan w_{01} merupakan bobot bias pada lapisan *output*

2. Fungsi Aktivasi

Fungsi aktivasi yang digunakan dalam penelitian ini adalah fungsi *sigmoid* biner (*logsig*). Fungsi *sigmoid* biner disebut juga sebagai fungsi *sigmoid* logistik. Fungsi *sigmoid* biner digunakan karena memiliki nilai fungsi antara 0 sampai 1 dan dapat diturunkan dengan mudah (Kusumadewi, 2003). Fungsi *sigmoid* biner memiliki rumus sebagai berikut (Fausset, 1994):

$$y = f(x) = \frac{1}{1 + e^{-\sigma x}}, -\infty < x < \infty \quad (3)$$

$$f'(x) = f(x) - (1 - f(x)) \quad (4)$$

3. Algoritma Pembelajaran *Backpropagation* Neural Network

Algoritma *backpropagation* merupakan salah satu jaringan yang termasuk dalam *supervised learning*. Penggunaan metode *backpropagation* memiliki 2 tahap yaitu tahap pelatihan atau tahap pembelajaran dimana pada tahap ini digunakan untuk melatih dengan berbagai data yang diberikan. Selanjutnya tahap pengujian atau penggunaan pada tahap ini dilakukan untuk menguji pada tahap pelatihan. Langkah-langkah dalam algoritma *backpropagation* adalah sebagai berikut (Kusumadewi, 2003):

Step 0 : Inisialisasi bobot dan bias

Step 1 : Jika *stopping condition* masih belum terpenuhi, jalankan step 2-9.

Step 2 : Untuk setiap data *training*, lakukan step 3-8.

Fase I : Umpan maju (*feedforward*)

Step 3 : Setiap neuron input ($x_{i'}; i' = 1, 2, \dots, p$) menerima sinyal input $x_{i'}$ dan menyebarkan sinyal tersebut pada seluruh neuron pada lapisan tersembunyi.

Step 4 : Setiap neuron tersembunyi ($z_{j'}; j' = 1, 2, \dots, q'$) akan menjumlahkan sinyal-sinyal input yang sudah berbobot, termasuk biasnya

$$z_{net,j'} = v_{0j'} + \sum_{i'=1}^p x_{i'} v_{i'j'} \quad (5)$$

dan memakai fungsi aktivasi yang telah ditentukan untuk menghitung sinyal *output* dari neuron tersembunyi yang bersangkutan,

$$z_{j'} = f(z_{net,j'}) \quad (6)$$

lalu mengirim sinyal *output* ini ke seluruh neuron pada neuron *output*

Step 5 : Setiap neuron *output* y akan menjumlahkan sinyal-sinyal input yang sudah berbobot, termasuk biasnya,

$$y_{net1} = w_{01} + \sum_{j'=1}^{q'} z_{j'} w_{j'1} \quad (7)$$

dan memakai fungsi aktivasi yang telah ditentukan untuk menghitung sinyal *output* dari neuron *output* yang bersangkutan:

$$y = f(y_{net1}) = \frac{1}{1 + e^{-y_{net1}}} \quad (8)$$

Fase II : Propagasi error (*backpropagation of error*)

Step 6 : Setiap neuron *output* y menerima suatu target *pattern* (*desired output*) yang sesuai dengan input *training pattern* untuk menghitung kesalahan (*error*) antara target dengan *output* yang dihasilkan jaringan:

$$\delta_1 = (t' - y) f'(y_{net1}) = (t' - y) y(1 - y) \quad (9)$$

faktor δ_1 ini digunakan untuk menghitung koreksi error ($\Delta w_{j'1}$) yang nantinya akan dipakai untuk memperbaharui $w_{j'1}$, di mana:

$$\Delta w_{j'1} = \alpha \times \delta_1 \times z_{j'}; j' = 1, 2, \dots, q' \quad (10)$$

Selain itu juga dihitung koreksi bias Δw_{01} yang nantinya akan dipakai untuk memperbaharui w_{01} , di mana:

$$\Delta w_{01} = \alpha \times \delta_1 \quad (11)$$

Faktor δ_1 ini kemudian dikirimkan ke lapisan yang berada pada step 7.

Step 7 : Setiap neuron tersembunyi ($z_{j'}, j' = 1, 2, \dots, q'$) menjumlah input delta (yang dikirim dari lapisan step 6) yang sudah berbobot.

$$\delta_{net,j'} = \sum_{j'=1}^{q'} \delta_1 w_{j'1} \quad (12)$$

Faktor δ neuron tersembunyi:

$$\delta_{j'} = \delta_{net,j'} f'(\delta_{net,j'}) = \delta_{net,j'} z_{j'}(1 - z_{j'}) \quad (13)$$

menghitung koreksi error ($\Delta v_{i'j'}$) yang nantinya akan dipakai untuk memperbaharui $v_{i'j'}$, di mana:

$$\Delta v_{i'j'} = \alpha \times \delta_{j'} \times x_{i'} \quad (14)$$

Selain itu juga dihitung koreksi bias Δv_{0j} yang nantinya akan dipakai untuk memperbaharui $v_{0j'}$ di mana

$$\Delta v_{0j'} = \alpha \times \delta_{j'} \quad (15)$$

Fase III: Pembaharuan bobot dan bias

Step 8 : Setiap neuron *output* akan memperbaharui bias dan bobotnya dari setiap neuron tersembunyi.

$$w_{j'1}(\text{baru}) = w_{j'1}(\text{lama}) + \Delta w_{j'1} \quad (16)$$

$$w_{01}(\text{baru}) = w_{01}(\text{lama}) + \Delta w_{01} \quad (17)$$

Demikian pula untuk setiap neuron tersembunyi akan memperbaharui bias dan bobotnya dari setiap neuron input.

$$v_{i'j'}(\text{baru}) = v_{i'j'}(\text{lama}) + \Delta v_{i'j'} \quad (18)$$

$$v_{0j'}(\text{baru}) = v_{0j'}(\text{lama}) + \Delta v_{0j'} \quad (19)$$

Hybrid ARIMA-NN

Metode *hybrid* adalah salah satu metode numerik yang dilakukan dengan menggabungkan dua atau lebih metode dalam fungsi suatu sistem. Salah satu metode *hybrid* yang dilakukan adalah penggabungan metode ARIMA dan *neural network*. Menurut Zhang (2001) menjelaskan alasan pengembangan metode *hybrid* yaitu meningkatkan efisiensi dalam tahapan analisis peramalan runtun waktu Secara umum, kombinasi dari model time series yang memiliki struktur autokorelasi linier dan nonlinier dapat dituliskan:

$$g_t = x_t + y_t \quad (20)$$

dengan,

x_t : komponen linier ke t; y_t : komponen nonlinier ke t; t : indeks waktu

Dua komponen tersebut dapat digunakan untuk meramalkan suatu data. Pertama, hasil ramalan model ARIMA digunakan sebagai komponen linier. Kemudian, *residual* dari model linier sebagai komponen nonlinier. Misal a_t sebagai *residual* pada saat t pada model linier, maka

$$a_t = x_t - \hat{x}_t \quad (21)$$

dimana, x_t adalah nilai aktual ke-t; \hat{x}_t adalah nilai prediksi ARIMA pada waktu t.; a_t adalah nilai error ke-t dari model ARIMA; t adalah indeks waktu.

Struktur korelasi linier pada *residual* akan membuat model linier tidak terpenuhi. Diagnostik hubungan non linier dapat diatasi dengan menggunakan NN. Pemodelan *residual* menggunakan NN dengan unit input sebanyak n dapat dituliskan sebagai berikut

$$a_t = f(a_{(t-1)}, a_{(t-2)}, \dots, a_{(t-n)}) + \varepsilon_t \quad (22)$$

dengan, ε_t adalah nilai *error* ke- t dari model NN; f adalah fungsi non linier dari NN; t adalah indeks waktu. Fungsi f merupakan fungsi nonlinier yang dijelaskan oleh NN dan ε_t merupakan eror yang acak ke t . Sehingga, peramalan dengan metode *hybrid* diperoleh

$$\hat{z}_t = \hat{x}_t + \hat{y}_t \quad (23)$$

Kriteria Pemilihan Model

Kriteria pemilihan yang digunakan untuk memilih model terbaik dapat menggunakan kriteria pemilihan yaitu *Akaike's information criterion* (AIC) yakni dengan nilai AIC terkecil. *Akaike's information criterion* (AIC) memiliki rumus sebagai berikut:

$$AIC = \ln(\partial_\alpha^2) + r \frac{2}{n} \quad (24)$$

dengan $r = p + q + 1$; ∂_α^2 = estimasi *maximum likelihood* dari σ_α^2 ; n = banyaknya pengamatan; p dan q = ordo dari model ARIMA

Selain dilihat dari nilai *Akaike's information criterion* (AIC) terkecil, pemilihan model terbaik juga dapat dilakukan dengan membandingkan nilai *Mean Absolute Percentage Error* (MAPE) dengan persamaan berikut (Hanke & Wichern, 2014):

$$MAPE = \frac{1}{n} \sum_{t=1}^n \left| \frac{x_t - \hat{x}_t}{x_t} \right| \times 100\% \quad (25)$$

dengan,

n = banyaknya pengamatan

x_t = data aktual pengamatan pada periode waktu ke- t

\hat{x}_t = hasil prediski periode waktu ke- t

Prosedur Analisis Data Model *Hybrid* ARIMA-NN

Langkah-langkah dalam melakukan penelitian tentang prediksi nilai tukar petani di Jawa Tengah dengan menggunakan metode *hybrid* ARIMA-NN adalah sebagai berikut:

1. Persiapan data

Data yang dikumpulkan berupa data sekunder yang diperoleh dari BPS Provinsi Jawa Tengah berupa data bulanan Nilai Tukar Petani. Data dibagi menjadi data *training* dan data *testing* dengan perbandingan 80% dan 20%. Pembagian dilakukan secara berurutan berdasarkan waktu, sehingga data periode awal digunakan untuk pelatihan model dan data periode akhir digunakan untuk pengujian model. Pembagian secara kronologis ini dilakukan agar proses evaluasi tetap sesuai dengan karakteristik data runtun waktu.

2. Analisis model ARIMA

Dalam menganalisis model ARIMA digunakan beberapa tahapan yakni:

a. Identifikasi model

Identifikasi model dilakukan dengan pemeriksaan kestasioneran data. Data harus stasioner baik dalam variansi maupun rata-rata. Pengujian kestasioner data runtun waktu dapat dilakukan dengan cara uji Augmented Dickey-Fuller, yang dijabarkan sebagai berikut:

Hipotesis

$H_0 : \gamma = 0$ (data runtun waktu tidak stasioner)

$H_1 : \gamma < 0$ (data runtun waktu stasioner)

Taraf Signifikansi $\alpha = 5\%$

Statistik Uji yang digunakan dalam uji Augmented Dickey-Fuller dinyatakan sebagai berikut (Wei, 2006):

$$\Delta x_t = \alpha + \beta_t + \gamma x_{t-1} + \sum_{k=1}^K \delta_k \Delta x_{t-k} + e_t \quad (26)$$

dimana, $\Delta x_t = x_t - x_{t-1}$; $\Delta x_{t-k} = x_t - x_{t-k}$; t : indeks waktu atau periode ($t=1,2,3,\dots,n$); k : banyaknya lag ($k = 1,2,3,\dots, K$); x_{t-1} : pengamatan pada periode $t-1$; α : koefisien intersep model; β : koefisien variabel waktu; γ : koefisien variabel x_{t-1} atau unit roots; δ_k : koefisien variabel Δx_{t-k} ; e_t : residual model

Statistik uji:

$$t = \frac{\hat{\gamma}}{S_{e(\hat{\gamma})}} \quad (27)$$

$$\hat{\gamma} = \frac{\sum_{t=1}^n x_{t-1} x_t}{\sum_{t=1}^n x_{t-1}^2} \quad (28)$$

$$S_{e(\hat{\gamma})} = \sqrt{\frac{\hat{\sigma}_e^2}{\sum_{t=1}^n x_{t-1}^2}} \quad (29)$$

$$\hat{\sigma}_e^2 = \frac{\sum_{t=1}^n (x_t - \hat{\gamma} * x_{t-1})^2}{n - 1} \quad (30)$$

dimana,

$\hat{\gamma}$: unit root

$S_{e(\hat{\gamma})}$: standar error dari unit root

Kriteria keputusan: H_0 ditolak apabila nilai $t > t_{n-K-1}$ dengan n menyatakan jumlah data dan K menyatakan jumlah lag atau H_0 ditolak apabila $p\text{-value} < \text{taraf signifikansi } (\alpha)$. Apabila H_0 ditolak maka disimpulkan bahwa data tersebut stasioner.

Apabila data belum stasioner secara rata-rata maka dilakukan proses *differencing*. *Differencing* adalah menghitung perubahan atau selisih nilai observasi. Nilai selisih yang

diperoleh kemudian dihitung kembali apakah stasioner atau tidak, jika tidak maka dilakukan *differencing* kembali. Proses *differencing ke-d* dapat dituliskan sebagai berikut :

$$\nabla^d x_t = (1 - B)^d x_t \tag{31}$$

dengan, $d = differencing$; $x_t =$ pengamatan pada waktu ke- t ; $B =$ operator *backshift*

Sedangkan, apabila stasioner dalam variansi ini tidak terpenuhi maka perlu dilakukan proses Transformasi Box Cox. Transformasi Box Cox merupakan transformasi dengan memangkatkan variabel dependen (y) dengan parameter λ . Nilai parameter λ dan transformasinya dapat dilihat pada Tabel 1 berikut (Hillmer & Wei, 1991):

Tabel 1. Nilai *lambda* dan transformasi

| Nilai <i>Lambda</i> (λ) | Transformasi |
|-----------------------------------|------------------------|
| -2 | $\frac{1}{(x_t)^2}$ |
| -1 | $\frac{1}{x_t}$ |
| -0,5 | $\frac{1}{\sqrt{x_t}}$ |
| 0 | $\ln(x_t)$ |
| 0,5 | $\sqrt{x_t}$ |
| 1 | x_t |
| 2 | x_t^2 |

b. Pemodelan ARIMA

Setelah diperoleh data yang telah stasioner baik dalam rata-rata maupun dalam variansi selanjutnya adalah menentukan model ARIMA yang digunakan dengan menentukan ordo p dan q . Penentuan ordo p dan q dapat dilakukan dengan melihat nilai dan plot ACF dan PACF yang signifikan. Berikut karakteristik dari plot ACF dan PACF disajikan pada Tabel 2 (Wei, 2006):

Tabel 2. Karakteristik ACF & PACF

| Proses | Fungsi ACF | Fungsi PACF |
|---------------|--|--|
| AR(p) | Meluruh menuju 0 secara eskponensial | Terputus seketika (<i>cut off</i>) menuju nol sampai lag p |
| MA(q) | Terputus seketika (<i>cut off</i>) menuju nol sampai lag q | Meluruh menuju 0 secara eskponensial |
| ARMA(p,q) | Meluruh menuju 0 secara eskponensial | Meluruh menuju 0 secara eskponensial |

i. Estimasi parameter

Estimasi parameter dilakukan pada data *training* yaitu dilakukan dengan menggunakan model *maximum likelihood*.

ii. Uji Signifikansi Parameter

Setelah parameter diestimasi, langkah selanjutnya adalah melakukan uji untuk mengetahui apakah parameter tersebut signifikan atau tidak. Uji signifikansi parameter dapat dilakukan dengan tahap sebagai berikut:

Hipotesis

Untuk parameter AR:

$$H_0: \phi_j = 0, j = 1, 2, \dots, p \text{ (parameter model AR tidak signifikan)}$$

$$H_1: \phi_j \neq 0, j = 1, 2, \dots, p \text{ (parameter model AR signifikan)}$$

Untuk parameter MA:

$$H_0: \theta_l = 0, l = 1, 2, \dots, q \text{ (parameter model MA tidak signifikan)}$$

$$H_1: \theta_l \neq 0, l = 1, 2, \dots, q \text{ (parameter model MA signifikan)}$$

Taraf signifikansi: $\alpha = 5\%$

Statistik uji:

Untuk parameter AR:

$$t = \frac{\hat{\phi}_j}{SE(\hat{\phi}_j)} \quad (31)$$

Untuk parameter MA:

$$t = \frac{\hat{\theta}_l}{SE(\hat{\theta}_l)} \quad (32)$$

Kriteria keputusan: H_0 ditolak apabila nilai $t > t_{(\frac{\alpha}{2}; n-n_p)}$ dengan n menyatakan jumlah data dan n_p menyatakan jumlah parameter, atau H_0 ditolak apabila $p - value < \alpha$. Sehingga dapat disimpulkan bahwa parameter model ARIMA signifikan.

c. Pemilihan model terbaik

Model yang telah memenuhi uji signifikansi kemudian dipilih salah satu sebagai model terbaik. Pemilihan model terbaik berdasarkan nilai AIC dan MAPE yang terkecil.

d. Prediksi ARIMA

Model ARIMA terbaik yang diperoleh digunakan untuk memprediksi nilai tukar petani di Jawa Tengah baik data *training* maupun data *testing*. Hasil prediksi disimpan dan dilakukan perhitungan data *residual*. Data *residual* data *training* dari proses ARIMA kemudian digunakan sebagai input untuk proses NN selanjutnya.

3. Pemodelan *Hybrid* (ARIMA-NN)

Setelah diperoleh komponen linier berupa hasil prediksi model ARIMA, selanjutnya adalah membentuk komponen non linier yang didapatkan dan dimodelkan *residual* data *training* model ARIMA dengan menggunakan metode *backpropagation*.

e. Pembentukan input

Pembentukan input ditentukan berdasarkan plot ACF dan PACF *residual* data *training* dari model ARIMA. *Lag residual* yang signifikan digunakan sebagai dasar penentuan banyaknya neuron *input* pada jaringan Neural Network.

f. Normalisasi data

Normalisasi data dilakukan agar keluaran jaringan sesuai dengan fungsi aktivasi yang digunakan atau berada pada *range* [0,1]. Normalisasi dilakukan dengan menggunakan persamaan berikut:

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)} \quad (33)$$

dengan: x = data asli; x' = data hasil normalisasi

g. Penentuan jaringan *neural network*

Penentuan jaringan dilakukan terhadap data *training* yang telah dinormalisasi untuk menentukan banyaknya neuron input, lapisan tersembunyi, neuron tersembunyi, dan neuron *output*. Pemilihan arsitektur dilakukan melalui beberapa percobaan jumlah neuron tersembunyi, kemudian model dengan nilai MAPE *testing* terkecil dipilih sebagai model NN yang digunakan dalam *hybrid*.

h. Pelatihan *neural network*

Proses pelatihan data merupakan proses pengenalan pola data untuk mendapatkan nilai bobot yang mampu memetakan antara data *input* dan data target, dimana bobot dapat diganti dan dilakukan berulang hingga batas *learning rate*.

i. Pengujian *neural network*

Pada tahap pelatihan telah diperoleh nilai bobot dan bias yang terbaik yang dari sistem jaringan. Bobot dan bias tersebut kemudian dilakukan pengujian terhadap data *testing* guna melihat apakah jaringan dapat bekerja dengan baik untuk memprediksi nilai tukar petani.

j. Prediksi dengan *neural network*

Setelah ditentukan model jaringan yang baik dengan melihat MAPE *training* dan *testing*, kemudian model dapat digunakan dalam melakukan prediksi dimasa mendatang menggunakan model jaringan terbaik yang dimiliki.

k. Pemodelan *Hybrid* ARIMA-NN

Setelah diperoleh komponen linear dari model ARIMA dan komponen *nonlinear* dari model NN, kedua hasil prediksi digabungkan menjadi model *hybrid* ARIMA-NN. Selanjutnya dilakukan pemeriksaan *residual*, termasuk uji *white noise*, untuk melihat apakah *residual* model sudah bersifat acak dan tidak menunjukkan pola tertentu.

1. Prediksi *Hybrid* ARIMA-NN

Penentuan hasil prediksi *hybrid* ARIMA-NN dilakukan dengan menjumlahkan kedua hasil prediksi dari masing-masing model dengan persamaan berikut:

$$\hat{g}_t = \hat{x}_t + \hat{y}_t \quad (34)$$

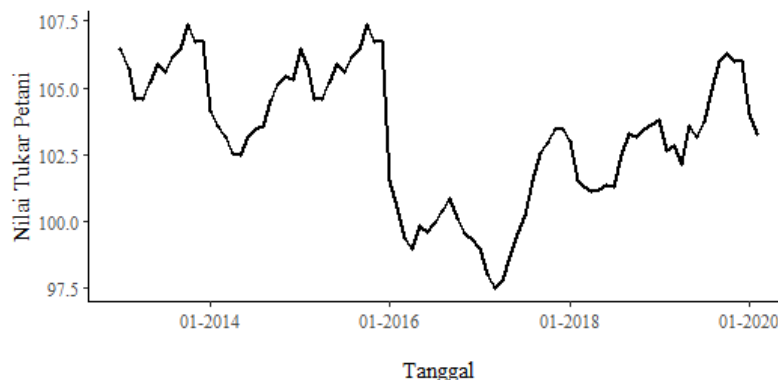
m. Analisis hasil prediksi

Setelah diperoleh hasil prediksi dengan model *hybrid* ARIMA-NN, nilai MAPE dihitung untuk mengetahui tingkat kesalahan prediksi. Nilai MAPE digunakan sebagai ukuran evaluasi karena dapat menunjukkan besarnya kesalahan prediksi dalam bentuk persentase sehingga lebih mudah diinterpretasikan.

HASIL DAN PEMBAHASAN

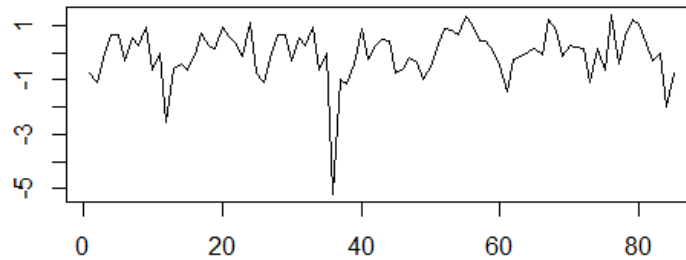
Hasil

Langkah pertama yang dilakukan adalah membagi data menjadi data *training* dan data *testing*, dengan perbandingan 80% untuk data *training* yakni 86 data dan 20% untuk data *testing* yakni 22 data. Selanjutnya data *training* digunakan dalam uji kestasioneran data baik dilihat dari plot maupun uji *Augmented Dickey Fuller* (ADF).



Gambar 2. Plot nilai tukar petani data *training*

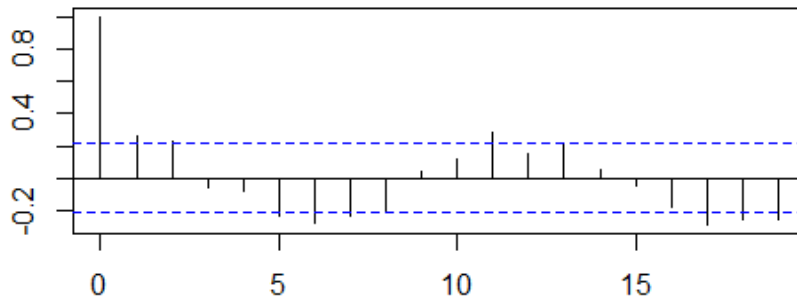
Gambar 2 menunjukkan bahwa data belum stasioner karena plot memiliki tren. Selain itu, dirujuk dari hasil uji *augmentend dickey fuller* diperoleh nilai *p-value* sebesar 0,37, dimana nilai *p-value* yang diperoleh lebih besar daripada nilai taraf signifikansi yang digunakan (5%), sehingga dapat disimpulkan bahwa data nilai tukar petani merupakan data yang belum stasioner. Data yang belum stasioner kemudian distasionerkan baik dalam variansi maupun rata-rata. Hal pertama yang dilakukan adalah menstasionerkan terhadap rata-rata yakni proses *differencing* dengan menggunakan fungsi *diff* pada Rstudio diperoleh hasil plot seperti pada Gambar 3 berikut:



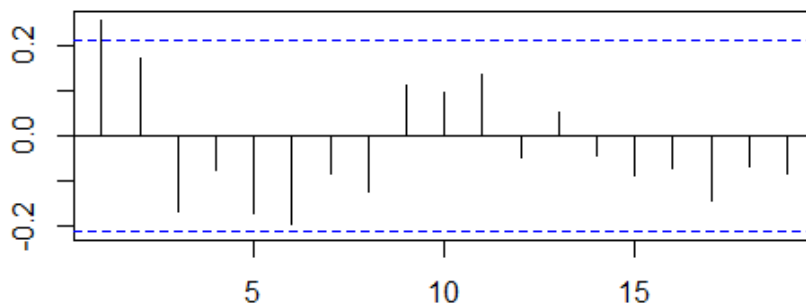
Gambar 3. *Differencing data training*

Selanjutnya mengecek kembali kestasioneran data dengan uji ADF dan diperoleh hasil p -value 0,01 sehingga data telah stasioner dalam rata-rata. Setelah diperoleh hasil stasioner dalam rata-rata, kemudian dicek stasioneritas dalam variansi menggunakan transformasi BoxCox diperoleh hasil lamda sama dengan 1, hal ini berarti data telah stasioner dalam variansi. Sehingga disimpulkan bahwa data telah stasioner baik dalam rata-rata maupun dalam variansi.

Data yang telah memenuhi syarat kestasioneran kemudian dilakukan proses ARIMA diawali dengan penentuan nilai AR dan MA. Penentuan AR dan MA dapat ditentukan dengan memperhatikan hasil plot ACF dan PACF. Plot ACF menjelaskan nilai ordo q , sedangkan plot PACF menjelaskan nilai ordo p dan d merupakan banyaknya proses *differencing*, karena data mengalami proses *differencing* sebanyak satu kali maka nilai d sama dengan 1. Plot ACF dan PACF data nilai tukar petani menggunakan Rstudio sebagai berikut:



Gambar 4. Plot ACF



Gambar 5. Plot PACF

Dilihat dari Gambar 4 terlihat bahwa plot ACF data nilai tukar petani memiliki *lag* yang signifikan atau yang memotong garis biru pada *lag* ke 0, 1, dan 2, dan nilai PACF pada Gambar 5 pada data nilai tukar petani juga memiliki *lag* yang signifikan pada *lag* 1. Hal ini

dapat dibentuk 4 model dugaan yakni ARIMA (1,1,1), ARIMA (0,1,1), ARIMA (1,1,2), dan ARIMA (0,1,2). Kemudian model-model dugaan dilakukan pengujian parameter untuk mengetahui apakah model layak digunakan atau tidak. Hasil perhitungan nilai estimasi dan uji signifikansi parameter disajikan pada Tabel 3 sebagai berikut:

Tabel 3. Uji estimasi dan signifikansi parameter

| ARIMA | Parameter | Nilai Estimasi | t | p-value | keterangan |
|---------------|------------|----------------|---------|---------|------------------|
| ARIMA (1,1,1) | ϕ_1 | 0,498 | 2,45 | 0,014 | Signifikan |
| | θ_1 | -0,243 | -1,16 | 0,244 | Tidak Signifikan |
| ARIMA (0,1,1) | θ_1 | 0,1792 | 2,0882 | 0,036 | Signifikan |
| | ϕ_1 | -0,0744 | -0,0846 | 0,932 | Tidak signifikan |
| ARIMA (1,1,2) | θ_1 | 0,3123 | 0,3593 | 0,71 | Tidak signifikan |
| | θ_1 | 0,2750 | 1,7399 | 0,08 | Tidak signifikan |
| ARIMA (0,1,2) | θ_1 | 0,239 | 2,25 | 0,0244 | Signifikan |
| | θ_1 | 0,264 | 2,6813 | 0,0073 | Signifikan |

Dari Tabel 3 diperoleh model ARIMA yang dapat digunakan dalam proses selanjutnya yakni model ARIMA (1,1,1), ARIMA (0,1,1), dan ARIMA (0,1,2). Dari beberapa model dugaan tersebut kemudian dipilih satu model terbaik. Dalam penelitian ini, pemilihan model terbaik dilakukan dengan membandingkan nilai *Akaike's Information Criterion* (AIC) dan nilai MAPE model ARIMA. Hasil nilai AIC dan MAPE dari masing-masing model data nilai tukar petani disajikan pada Tabel 4.

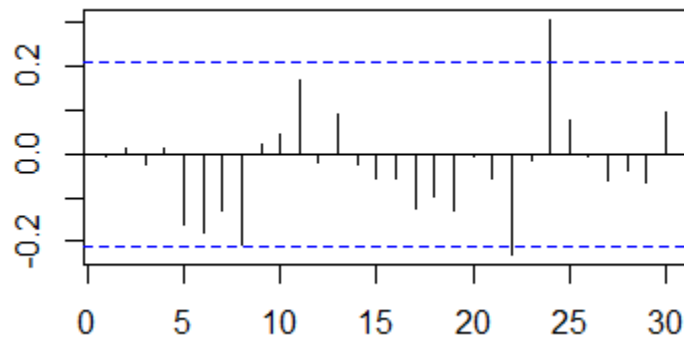
Tabel 4. Hasil AIC dan MAPE model ARIMA

| MODEL | AIC | MAPE |
|----------------------|---------------|--------------|
| ARIMA (1,1,1) | 228,03 | 0,565 |
| ARIMA (0,1,1) | 228,94 | 0,585 |
| ARIMA (1,1,2) | 226,7 | 0,566 |
| ARIMA (0,1,2) | 224,71 | 0,567 |

Berdasarkan Tabel 4, model ARIMA (0,1,2) memiliki nilai AIC terkecil, yaitu 224,71, dan nilai MAPE yang relatif kecil, yaitu 0,567. Oleh karena itu, model ARIMA (0,1,2) dipilih sebagai model ARIMA terbaik untuk tahap pemodelan berikutnya. Setelah diperoleh model terbaik untuk memprediksi Nilai Tukar Petani di Jawa Tengah, persamaan model ARIMA (0,1,2) adalah sebagai berikut:

$$x_t = a_t - 0,239a_{t-1} - 0,264a_{t-2} + x_{t-1}$$

Selanjutnya model digunakan untuk memprediksi nilai *training* dan *testing* guna mencari nilai *residual* dari data *training* dan *testing* untuk digunakan sebagai komponen linier atau untuk proses NN selanjutnya. Setelah melakukan pemodelan linier yang menggunakan metode ARIMA. Langkah selanjutnya adalah melakukan pemodelan non linier dengan menggunakan metode *backpropagation* dengan data yang digunakan adalah data *residual* atau data selisih antara data aktual dengan hasil prediksi. Penentuan variabel input dilakukan dengan melihat plot autokorelasi parsial atau PACF dari data *residual* ARIMA dengan memperhatikan *lag-lag* yang signifikan.



Gambar 6. Plot PACF Residual Model ARIMA (0,1,2)

Berdasarkan Gambar 6, plot PACF plot PACF *residual* model ARIMA (0,1,2) signifikan pada *lag* 22 dan *lag* 24 sehingga input yang digunakan dalam jaringan NN sebanyak dua neuron input. Selanjutnya, data *residual* dinormalisasi menggunakan normalisasi min-max agar sesuai dengan rentang fungsi aktivasi yang digunakan. Penentuan parameter jaringan dilakukan dengan mencoba jumlah neuron tersembunyi 2 sampai 10, maksimum epoch 1000 sampai 3000, dan learning rate 0,1.

Selanjutnya dinilai akurasi melalui rumus *Mean Absolute Percentage* (MAPE) dimana nilai MAPE terkecil merupakan model yang paling baik. Hasil dari akurasi data *training* dan *testing* diperoleh dengan bantuan *software Matlab* yang disajikan dalam Tabel 5 berikut:

Tabel 5. Uji model NN

| Banyak neuron tersembunyi | MAPE <i>Training</i> (%) | MAPE <i>Testing</i> (%) |
|---------------------------|--------------------------|-------------------------|
| 2 | 285,11 | 369,34 |
| 3 | 518,05 | 345,24 |
| 4 | 583,29 | 565,10 |
| 5 | 613,0 | 129,14 |
| 6 | 259,39 | 341,82 |
| 7 | 346,31 | 1324,5 |
| 8 | 201,40 | 1248,3 |
| 9 | 402,96 | 1496,1 |
| 10 | 383,03 | 625,05 |

Berdasarkan Tabel 5, nilai MAPE *testing* terkecil diperoleh pada model NN (2,5,1). Model ini terdiri atas 2 neuron input, 5 neuron pada lapisan tersembunyi, dan 1 neuron *output*. Meskipun nilai MAPE pada tahap pemodelan *residual* masih cukup besar, model ini dipilih karena memiliki MAPE *testing* paling kecil dibandingkan alternatif arsitektur NN lainnya. Oleh karena itu, model NN (2,5,1) digunakan sebagai komponen nonlinear dalam model *hybrid* ARIMA-NN.

Tabel 6. Hasil uji coba parameter

| <i>Epoch</i> | MAPE <i>Training</i> | MAPE <i>Testing</i> |
|--------------|----------------------|---------------------|
| 1000 | 613,0 | 129,14 |
| 1500 | 617,96 | 145,08 |
| 2000 | 614,74 | 157,69 |
| 2500 | 608,32 | 166,42 |
| 3000 | 593,37 | 175,10 |

Berdasarkan Tabel 6, parameter yang menghasilkan nilai MAPE *testing* terkecil adalah *epoch* 1000 dengan *learning rate* 0,1. Dengan demikian, model NN yang digunakan dalam pemodelan *hybrid* adalah NN (2,5,1) dengan *epoch* 1000 dan *learning rate* 0,1.

$$\hat{y} = \sum_{j'=1}^5 w_{j'1} \times \frac{1}{1 + e^{-[v_{0j'} + \sum_{i=1}^1 x_i v_{ij'}]}} + w_{01}$$

Setelah melalui proses ARIMA dan NN diperoleh model *hybrid* ARIMA-NN adalah ARIMA (0,1,2) dan dari NN diperoleh molde NN (2,5,1) dengan *epoch* 1000 dan *learning rate* 0,1, diperoleh hasil prediksi nilai tukar petani tahun 2022 dengan model *hybrid* ARIMA-NN disajikan pada Tabel 7.

Tabel 7. Hasil prediksi *hybrid* ARIMA-NN

| Data Training | | | | | |
|---------------|-------------|----------------|-------|---------|----------|
| Date | Data Aktual | Hasil Prediksi | error | MAPE | |
| Jan-13 | 106,45 | 106,34445 | | 0,10645 | 0,001163 |
| Feb-13 | 105,70 | 106,40726 | - | 0,70786 | 0,007787 |
| ⋮ | ⋮ | ⋮ | | ⋮ | ⋮ |
| Jan-20 | 104,03 | 104,63880 | - | 0,60880 | 0,006805 |
| Feb-20 | 103,29 | 103,04366 | | 0,24634 | 0,002773 |
| | | | | MAPE | 0,5006 |
| Data Testing | | | | | |
| Date | Data Aktual | Hasil Prediksi | error | MAPE | |
| Mar-20 | 102,12 | 102,05712 | | 0,06288 | 0,00293 |
| Apr-20 | 100,51 | 98,84636 | | 1,66364 | 0,07882 |
| ⋮ | ⋮ | ⋮ | | ⋮ | ⋮ |
| Nov-21 | 101,34 | 100,89135 | | 0,44865 | 0,02108 |
| Dec-21 | 103,18 | 105,01470 | - | 1,83470 | 0,08467 |
| | | | | MAPE | 0,5928 |

Dari Tabel 7 terlihat bahwa hasil prediksi Nilai Tukar Petani di Jawa Tengah menggunakan model *hybrid* ARIMA-NN menghasilkan MAPE *training* sebesar 0,5006% dan MAPE *testing* sebesar 0,5928%. Nilai tersebut menunjukkan bahwa tingkat kesalahan prediksi model relatif kecil. Selanjutnya, model *hybrid* ARIMA-NN dibandingkan dengan model ARIMA (0,1,2) dan model *hybrid* lain berdasarkan input ACF untuk melihat perbedaan kinerja model. Perbandingan dilakukan dengan melihat nilai MAPE *training* dan *testing* sebagaimana disajikan pada Tabel 8.

Tabel 8. Perbandingan model

| Model | MAPE Training | MAPE Testing |
|---------------------------------|---------------|--------------|
| ARIMA (0,1,2) | 0,567 | 0,588 |
| Hybrid ARIMA (0,1,2)-NN (2,5,1) | 0,5006 | 0,5928 |
| Hybrid ARIMA(0,1,2)-NN (1,10,1) | 0,449 | 1,046 |

Berdasarkan Tabel 8, model *hybrid* ARIMA (0,1,2)-NN (2,5,1) memiliki MAPE *training* yang lebih kecil dibandingkan model ARIMA (0,1,2), tetapi MAPE *testing*-nya sedikit lebih besar dibandingkan model ARIMA. Hal ini menunjukkan bahwa model *hybrid* belum memberikan peningkatan performa *testing* yang signifikan dibandingkan ARIMA

tunggal. Namun, model *hybrid* ARIMA (0,1,2)-NN (2,5,1) tetap dapat dipertimbangkan karena nilai MAPE *testing* masih sangat kecil dan *residual* model telah memenuhi asumsi *white noise*. Sebaliknya, model *hybrid* ARIMA (0,1,2)-NN (1,10,1) memiliki MAPE *testing* yang lebih besar dan tidak memenuhi asumsi *white noise*.

Dengan demikian, model *hybrid* ARIMA (0,1,2)-NN (2,5,1) dipilih sebagai model *hybrid* yang layak digunakan dalam memprediksi Nilai Tukar Petani di Jawa Tengah. Pemilihan ini didasarkan pada nilai MAPE yang kecil, perbedaan MAPE *training* dan *testing* yang tidak terlalu jauh, serta *residual* model yang telah memenuhi asumsi *white noise*. Namun, hasil perbandingan juga menunjukkan bahwa keunggulan model *hybrid* terhadap ARIMA tunggal perlu ditafsirkan secara hati-hati karena nilai MAPE *testing* keduanya relatif berdekatan.

$$\hat{h}_t = a_t - 0,239a_{t-1} - 0,264a_{t-2} + x_{t-1} + \sum_{j'=1}^5 w_{j'1} \times \frac{1}{1 + e^{-[v_{0j'} + \sum_{i=1}^t x_i v_{i'j'}]}} + w_{01}$$

Tabel 9. Hasil prediksi

| Periode | Aktual | Prediksi <i>Hybrid</i> |
|---------|--------|------------------------|
| Jan-22 | 103,18 | 103,69305 |
| Feb-22 | 102,83 | 104,11326 |
| Mar-22 | 102,62 | 104,25929 |
| Apr-22 | 100,93 | 104,38622 |
| May-22 | 101,96 | 104,43513 |
| Jun-22 | 103,3 | 104,47372 |
| Jul-22 | 104,31 | 104,47704 |
| Aug-22 | 105,38 | 104,49651 |
| Sep-22 | 105,97 | 104,48911 |
| Oct-22 | 105,66 | 104,50871 |
| Nov-22 | 105,26 | 104,46732 |
| Dec-22 | 107,27 | 105,29848 |
| MAPE | | 1,36 |

Berdasarkan hasil prediksi pada Tabel 9, nilai prediksi Nilai Tukar Petani tertinggi terjadi pada bulan Desember 2022, sedangkan nilai prediksi terendah terjadi pada bulan Januari 2022. Hasil prediksi NTP tahun 2022 menghasilkan MAPE sebesar 1,36%. Berdasarkan kriteria MAPE, nilai tersebut menunjukkan bahwa model *hybrid* ARIMA-NN memiliki tingkat kesalahan prediksi yang rendah pada periode evaluasi tahun 2022.

Pembahasan

Hasil penelitian menunjukkan bahwa model *hybrid* ARIMA-NN dapat digunakan untuk memprediksi Nilai Tukar Petani di Jawa Tengah. Model ini dibangun dengan memanfaatkan ARIMA untuk menangkap pola linear dan Neural Network untuk memodelkan *residual* sebagai komponen nonlinear. Hasil MAPE *training* dan *testing* yang kecil menunjukkan bahwa model mampu menghasilkan prediksi dengan kesalahan yang rendah. Namun, perbandingan dengan model ARIMA tunggal menunjukkan bahwa peningkatan

performa model *hybrid* pada data *testing* belum terlalu besar. Oleh karena itu, model *hybrid* dalam penelitian ini lebih tepat dipahami sebagai alternatif model yang layak, bukan sebagai model yang secara mutlak lebih unggul pada seluruh indikator evaluasi.

Prediksi NTP tahun 2022 menunjukkan pola yang cenderung meningkat. Secara substantif, peningkatan NTP dapat mengindikasikan adanya perbaikan daya beli petani, karena indeks harga yang diterima petani relatif lebih tinggi dibandingkan indeks harga yang dibayarkan. Informasi prediksi seperti ini dapat dimanfaatkan oleh pemerintah daerah atau pihak terkait sebagai bahan pemantauan awal terhadap kondisi kesejahteraan petani. Meskipun demikian, penelitian ini masih terbatas pada data historis NTP sebagai variabel tunggal. Faktor lain seperti inflasi, harga pupuk, perubahan musim, produksi pertanian, dan kebijakan pemerintah belum dimasukkan ke dalam model, sehingga hasil prediksi perlu ditafsirkan sesuai dengan batasan data dan metode yang digunakan.

SIMPULAN

Model *hybrid* yang layak digunakan dalam memprediksi Nilai Tukar Petani di Jawa Tengah adalah *hybrid* ARIMA (0,1,2)-NN (2,5,1). Model ini menghasilkan MAPE *training* sebesar 0,5006% dan MAPE *testing* sebesar 0,5928%, serta *residual* yang telah memenuhi asumsi *white noise*. Hasil prediksi NTP tahun 2022 menghasilkan MAPE sebesar 1,36%, sehingga model memiliki tingkat kesalahan prediksi yang rendah pada periode evaluasi tersebut. Namun, hasil perbandingan menunjukkan bahwa MAPE *testing* model *hybrid* tidak jauh berbeda dari model ARIMA tunggal, sehingga keunggulan *hybrid* perlu ditafsirkan secara hati-hati. Penelitian selanjutnya dapat mengembangkan model dengan menambahkan variabel eksternal yang memengaruhi NTP atau membandingkan metode lain seperti ANFIS, regresi, maupun pendekatan machine learning lainnya.

UCAPAN TERIMA KASIH

Terimakasih kepada koordinator Program Studi Statistika dan seluruh Dosen Statistika yang telah memberikan ilmu hingga terselesainya artikel ini.

DAFTAR PUSTAKA

- Karo, I. M. K (2021). Prediksi penyebaran demam berdarah dengue dengan algoritma *Hybrid* autoregressive integrated moving average dan artificial neural network: studi kasus di kabupaten bandung. *Journal of Software Engineering, Information and Communication Technology (SEICT)*, 2(2), 130-135.
- Sari, V., & Maulidany, D. A. (2020). Prediksi kecepatan angin dalam mendeteksi gelombang air laut terhadap skala beaufort dengan metode *Hybrid* arima-ann (Studi Kasus: Kabupaten Lombok Barat 2019). *Jurnal Statistika Universitas Muhammadiyah Semarang*, 8(1).
- Naveena, K., Singh, S., Rathod, S., & Singh, A. (2017). *Hybrid ARIMA-ANN modelling for forecasting the price of robusta coffee in India*. *Int. J. Curr. Microbiol. Appl. Sci*, 6(7), 1721-1726.
- Wei, W. (2006). *Time series analysis: univariate and multivariate methods*. New York: Pearson Education, Inc.
- Siang, J. J. (2009). *Jaringan syaraf tiruan dan pemrogramannya menggunakan matlab*. Yogyakarta: Andi.

- Fauset, L. (1994). *Fundamental of neural network*. Upper Saddle River: Prentice Hall Inc.
- Hanke, J. E., & Wichern, D. W. (2014). *Business forecasting*. New Jersey: Pearson Education, Inc.
- Kusumadewi, S. (2003). *Artificial intelligence*. Graha Ilmu, Yogyakarta.
- Zhang, G. P. (2001). Time series forecasting using a *Hybrid ARIMA* and Neural Network Model. *Elsevier Neuro Computing*, pp. 159 - 175.
- Hillmer, S. C. (1991). Time series analysis: univariate and multivariate methods. *Journal of the American Statistical Association*, 86(413), 245-247.
- Wahyudin, S. (2019). Prediksi inflasi indonesia memakai model arima dan artificial neural network. *Jurnal Tata Kelola dan Kerangka Kerja Teknologi Informasi*, 5(2), 57-63.