

Perbandingan Metode *Seasonal Autoregressive Integrated Moving Average* (SARIMA) dan *Extreme Learning Machine* (ELM) pada Peramalan Peredaran Uang Kartal di Indonesia

Fita Ulya Pratiwi, Etik Zukhronah*, Isnandar Slamet
Program Studi Statistika, Universitas Sebelas Maret, Surakarta, Indonesia

*Corresponding author: etikzukhronah@staff.uns.ac.id

Submitted: 09-Jan-2024

Revised: 22-Oct-2024

Accepted: 20-Feb-2025

Abstract. Money is generally accepted as legal tender in fulfilling an obligation. Money circulation is very important to be considered and controlled, to have a positive impact on the economy. Control of money circulation is usually emphasized on the type of cash, which is in the form of paper and metals. One of the ways that can help in controlling cash is by forecasting. This study aims to compare the accuracy of forecasting results on cash circulation data using the SARIMA and ELM methods. The data used is the circulation cash from January 2011 to April 2022. The SARIMA method is a method for forecasting time series data containing seasonality, while the ELM method is a method on artificial neural networks that can do forecasting. The best SARIMA model obtained is SARIMA (1,1,0)(0,1,0)₁₂. The best ELM architecture obtains 12 input layer neurons, 45 hidden layer neurons, and 1 output layer. The measure of forecasting error to determine the best model is using MAPE. The results show that the SARIMA method has a training data MAPE of 2,3270% and testing data of 2,2772%, while the ELM method has a training data MAPE of 4,2548% and testing data of 3,8615%. Therefore, the SARIMA method is better than the ELM method at forecasting the circulation of cash in Indonesia.

Keywords: Cash; extreme learning machine; seasonal autoregressive integrated moving average.

1. PENDAHULUAN

Uang merupakan sesuatu yang diterima secara umum sebagai alat dalam pembayaran resmi untuk memenuhi suatu kewajiban. Salah satu fungsi uang yaitu alat pendorong ekonomi [1]. Oleh karena itu, untuk merealisasikan fungsinya diperlukan adanya peredaran uang agar nilai mata uang tetap stabil. Menurut Ranita [2], jumlah uang beredar mempengaruhi daya beli dan juga komoditi kebutuhan masyarakat. Uang beredar yang berada di tangan masyarakat harus berkembang secara normal, sehingga memberikan dampak positif terhadap perekonomian. Apabila perkembangannya meningkat terlalu tajam, maka hal tersebut dapat memicu adanya inflasi dan memberikan dampak negatif terhadap pertumbuhan ekonomi negara. Dengan demikian, jumlah peredaran uang tentu harus dikendalikan sesuai kapasitas perekonomian negara. Pengendalian peredaran uang biasanya ditekankan pada jenis uang kartal, yaitu yang berbentuk uang kertas dan logam. Uang kartal ini sering dijumpai dan beredar di tangan masyarakat. Salah satu cara yang dapat dilakukan dalam membantu pengendalian uang kartal yaitu melalui peramalan, sehingga didapatkan perkiraan peredaran uang kartal di masa depan yang dapat dijadikan sebagai bahan kajian pihak-pihak terkait.

Salah satu metode peramalan yang dapat digunakan untuk meramalkan peredaran uang kartal adalah metode *seasonal autoregressive integrated moving average* (SARIMA). Metode ini merupakan pengembangan dari metode ARIMA berpola musiman. Selain metode SARIMA, terdapat metode pengembangan dari jaringan syaraf tiruan yang dapat digunakan untuk peramalan yaitu metode *extreme learning machine* (ELM). Metode ini memiliki aturan pembelajaran yang cepat sehingga menghasilkan proses peramalan yang bagus [3]. Metode SARIMA merupakan metode yang baik dalam peramalan dengan parametrik sedangkan metode ELM merupakan metode yang baik dalam peramalan dengan non parametrik, sehingga dapat dibandingkan menggunakan data peredaran uang kartal untuk mendapatkan metode mana yang lebih baik.

Terdapat penelitian yang telah dilakukan menggunakan metode SARIMA maupun metode ELM. Prianda dan Widodo [4] melakukan perbandingan metode SARIMA dan ELM pada peramalan jumlah wisatawan mancanegara ke Bali. Metode optimal yang didapatkan yaitu metode SARIMA dengan *mean absolute percentage error* (MAPE) sebesar 4,97%. Rizki dan Taqiyyuddin [5], Utomo dan Fanani [6], serta Yahya [7] melakukan peramalan menggunakan metode SARIMA. Hasil ketiga penelitian mendapatkan nilai MAPE di bawah 10%, artinya akurasi peramalan tergolong sangat baik. Penelitian lainnya dilakukan oleh Hendayanti dan Nurhidayati [8] dan Lubis, dkk. [9] yaitu membandingkan metode SARIMA dengan metode *support vector regression* (SVR) dan *singular spectrum analysis* (SSA). Hasil penelitian menunjukkan bahwa metode SARIMA lebih unggul daripada metode pembandingnya. Selanjutnya Izati dkk. [10], Karmila dan Nirmala [11], Ardian, dkk. [3], Libianto, dkk. [12] melakukan peramalan menggunakan metode ELM. Nilai MAPE yang dihasilkan dari penelitian yaitu di bawah 10%, sehingga hasil peramalan tergolong sangat baik. Alfiyatin, dkk. [13] melakukan peramalan dengan membandingkan metode ELM dan Backpropagation. Hasil penelitian menunjukkan bahwa metode ELM lebih unggul daripada Backpropagation. Berdasarkan penelitian-penelitian tersebut, diketahui bahwa metode SARIMA dan ELM merupakan metode yang unggul karena menghasilkan nilai MAPE di bawah 10%, sehingga dilakukan penelitian dengan tujuan mengetahui hasil pemodelan dan membandingkan akurasi hasil peramalan data peredaran uang kartal di Indonesia menggunakan metode SARIMA dan ELM.

2. LANDASAN TEORI

2.1. Metode SARIMA

Metode SARIMA merupakan metode statistika yang digunakan dalam peramalan data runtun waktu mengandung musiman. Kelebihan metode SARIMA adalah dapat menerima semua jenis pola data runtun waktu, meskipun dalam prosesnya harus distasionerkan terlebih dahulu [14]. Penentuan orde metode SARIMA berdasarkan plot *autocorrelation function* dan *partial autocorrelation function* yang telah stasioner. Oleh karena itu, perlu dilakukan pengujian stasioneritas data. Suatu data dikatakan stasioner dalam variansi yaitu apabila data berfluktuasi dengan variansi yang tetap dari waktu ke waktu. Jika stasioneritas dalam variansi tidak terpenuhi, maka dilakukan transformasi Box-Cox. Selanjutnya suatu data dikatakan stasioner dalam rata-rata yaitu apabila plot data berfluktuasi disekitar nilai mean yang konstan. Uji yang digunakan untuk melihat kestasioneran data dalam rata-rata adalah uji akar unit augmented Dickey-Fuller (ADF). Jika kondisi tersebut tidak terpenuhi, maka dilakukan *differencing*. Model SARIMA yang optimal yaitu model yang telah memenuhi uji signifikansi parameter dan uji diagnostik residu, sehingga dapat dilanjutkan untuk peramalan. Uji diagnostik residu terdiri dari uji normalitas, uji

independensi, dan uji homoskedastisitas. Uji normalitas dilakukan untuk mengetahui apakah residu mengikuti distribusi normal atau tidak. Uji ini dapat dilakukan dengan uji Kolmogorov-Smirnov. Uji Independensi Residu didefinisikan dengan tidak adanya hubungan antar residu satu dengan lainnya. Uji ini menggunakan uji Ljung-Box. Selanjutnya uji homoskedastisitas digunakan untuk melihat apakah dalam model statistik terjadi ketidaksamaan variansi antar residu. Uji ini menggunakan uji white. Apabila residu berdistribusi normal, independen, dan homogen, maka sebuah model SARIMA dapat dinyatakan sebagai model yang tepat. Menurut Wei [15], secara umum model SARIMA yang dapat diestimasi dengan metode estimasi maksimum likelihood, ditulis dengan persamaan:

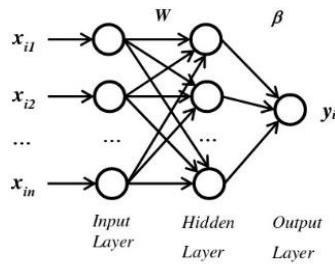
$$\phi_p(B)\Phi_P(B^S)(1-B)^d(1-B^S)^D X_t = \theta_q(B)\Theta_Q(B^S)\alpha_t,$$

dengan :

p, d, q	: orde AR, <i>differencing</i> , dan MA non musiman atau reguler,
P, D, Q	: orde AR, <i>differencing</i> , dan MA musiman,
$(1-B)^d$	: orde <i>differencing</i> non musiman,
$(1-B^S)^D$	: orde <i>differencing</i> musiman,
$\phi_p(B)$	: faktor AR non musiman,
$\phi_p(B)$	$= 1 - \phi_1(B) - \phi_2(B^2) - \dots - \phi_p(B^p),$
$\theta_q(B)$	: faktor MA non musiman,
$\theta_q(B)$	$= 1 - \theta_1(B) - \theta_2(B^2) - \dots - \theta_q(B^q),$
$\Phi_P(B^S)$	: faktor AR musiman,
$\Phi_P(B^S)$	$= 1 - \Phi_1(B^S) - \Phi_2(B^{2S}) - \dots - \Phi_P(B^{PS}),$
$\Theta_Q(B^S)$	: faktor MA musiman,
$\Theta_Q(B^S)$	$= 1 - \Theta_1(B^S) - \Theta_2(B^{2S}) - \dots - \Theta_Q(B^{QS}),$
X_t	: nilai pengamatan pada periode t,
α_t	: nilai residu pada waktu t.

2.2. Jaringan Syaraf Tiruan (JST)

JST merupakan bagian dari *machine learning* berupa sistem pemroses informasi yang karakteristiknya menyerupai jaringan syaraf biologi. Jaringan ini memiliki 3 lapisan penyusun yaitu *input layer*, *hidden layer*, dan *output layer*. Salah satu metode pada JST yang mampu menganalisis pola data dan melakukan peramalan adalah metode ELM. Huang *et al.* [16] menyatakan bahwa metode ELM merupakan pengembangan baru dari metode *feedforward* (sistem jaringan satu arah) dengan satu *hidden layer*. Metode ELM dibuat untuk mengatasi kelemahan dari metode JST lainnya terutama dalam hal *learning speed* [3]. Parameter-parameter yang digunakan, seperti *input weight* dan bias, ditentukan secara random dalam suatu rentang tertentu. Menurut Khotimah, dkk. [17], hal tersebut menyebabkan ELM memiliki *learning speed* yang cepat dan mampu menghasilkan *good generalized performance*. Struktur jaringan ELM dapat dilihat Gambar 1.



Gambar 1. Struktur Jaringan ELM

Metode ELM menggunakan tahapan *training* dan tahapan *testing*. Tujuan dari tahapan *training* ini adalah untuk mendapatkan nilai *output weight*. Menurut Huang *et al.* [16], langkah dalam tahapan *training* yaitu:

1. Menginisialisasi secara acak pada nilai *input weight* dan bias.
2. Menerima sinyal pada unit masukan x_i dan meneruskannya ke semua unit pada *hidden layer*.
3. Menghitung faktor keluaran *hidden layer* (H_{init}) yang berukuran $N \times \tilde{N}$ dengan persamaan:

$$H_{init} = (x_i w_j) + b_j,$$

dengan :

- H_{init} : Faktor keluaran pada *hidden layer*,
- i : $(1, 2, \dots, N)$, N adalah banyaknya data,
- j : $(1, 2, \dots, \tilde{N})$, $\tilde{N}$ adalah banyaknya neuron *hidden layer*,
- x_i : Vektor pola *input* i ,
- w_j : Vektor *input weight* yang telah di inisialisasi,
- b_j : Nilai bias yang telah di inisialisasi.

4. Menghitung nilai keluaran *hidden layer* (H) menggunakan fungsi aktivasi *sigmoid biner*, dengan persamaan:

$$H = \frac{1}{1 + \exp(-H_{init})},$$

dengan :

- H : Nilai dari fungsi aktivasi keluaran *hidden layer*,
- H_{init} : Faktor keluaran pada *hidden layer*.

5. Menghitung matriks *Moore-Penrose generalized invers* dari matriks hasil keluaran *hidden layer* (H). Langkah yang dilakukan yaitu mentranspose matriks hasil keluaran *hidden layer* (H). Selanjutnya matriks transpose tersebut dikalikan dengan matriks hasil keluaran *hidden layer*. Kemudian menghitung nilai *invers* dari matriks H tersebut dan dikalikan dengan matriks tranpose keluaran *hidden layer*. Rumus mencari matriksnya yaitu :

$$H^{\dagger} = ((H)^T H)^{-1} H^T,$$

dengan :

- $H^{\dagger}$: Matriks *Moore-Penrose generalized invers* dari matriks H ,
- H^T : Matriks H transpose,
- H : Nilai dari fungsi aktivasi keluaran *hidden layer*.

6. Langkah terakhir yaitu mendapatkan *output weight* (bobot antara *hidden layer* dan *output layer*) dengan rumus yaitu:

$$\beta = H^{\dagger} T,$$

dengan:

β : Matriks *output weight*,

$H^\dagger$: Matriks *Moore-Penrose generalized invers* dari matriks H ,

T : Matriks target.

Tujuan dari tahapan *testing* ini adalah mengevaluasi metode ELM dari hasil tahapan *training*. Tahapan *testing* ini menggunakan *input weight*, bias, dan *output weight*. Langkah-langkah tahapan *testing* yaitu [16]:

1. Menginisialisasi *input weight* dan bias yang telah didapatkan pada tahapan *training*.
2. Menghitung nilai keluaran *hidden layer* (H) menggunakan fungsi aktivasi *sigmoid biner*.
3. Menghitung keluaran *output layer* yang merupakan hasil peramalan, berdasarkan nilai *output weight* yang telah didapatkan dari tahapan *training*. Rumus menghitung *output layer* yaitu:

$$y = H\beta,$$

dengan:

y : Vektor nilai *output* (peramalan),

H : Matriks nilai dari fungsi aktivasi keluaran *hidden layer*,

β : Matriks *output weight* (didapat dari hasil tahapan *training*).

4. Menghitung nilai *error* untuk semua *output layer*. Nilai *error* ini menunjukkan kesalahan peramalan dari hasil yang telah didapatkan.

Alat ukur yang digunakan untuk menentukan besarnya kesalahan peramalan yaitu *mean absolute percentage error* (MAPE). Nilai MAPE merupakan rata-rata dari keseluruhan persentase kesalahan (selisih) antara data aktual dengan data hasil peramalan. Ukuran akurasi dicocokkan dengan data runtun waktu dalam bentuk persentase. Berikut merupakan persamaannya [18]:

$$MAPE = \frac{1}{n} \sum_{t=1}^n \left(\frac{X_t - \hat{X}_t}{X_t} \right) \times 100\%,$$

dengan:

X_t : Data ke- t , $t = 1, 2, \dots, n$,

$\hat{X}_t$: Ramalan ke- t , $t = 1, 2, \dots, n$,

n : Banyak data.

3. METODE PENELITIAN

Data yang digunakan pada penelitian ini merupakan data sekunder peredaran uang kartal (milyar rupiah) di Indonesia pada Bulan Januari 2011 sampai April 2022. Data diambil dari website Badan Pusat Statistik dengan total sebanyak 136 data. Data ini dibagi menjadi dua bagian yaitu data *training* dan data *testing* dengan proporsi pembagian 80% dan 20%. Data *training* sebanyak 109 data (Januari 2011 – Januari 2020) dan data *testing* sebanyak 27 data (Februari 2020 – April 2022). Penelitian ini menggunakan 2 metode yaitu metode SARIMA dan metode ELM. *Software* yang digunakan adalah R-studio, Minitab, serta Microsoft Excel.

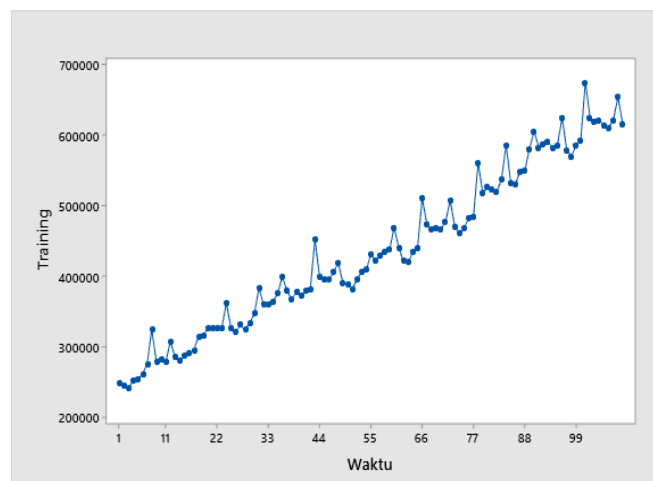
Metode penelitian yang digunakan untuk melakukan tahapan metode SARIMA diuraikan sebagai berikut: (1) menguji stasioneritas data *training* berdasarkan plot runtun waktu dan plot ACF. Apabila data tidak stasioner dalam variansi, maka dilakukan transformasi dengan metode Box-Cox. Sedangkan apabila data tidak stasioner dalam rata-rata, maka dilakukan *differencing*, (2) mengidentifikasi plot ACF dan PACF pada data *training* peredaran uang kartal yang telah stasioner untuk mendapatkan dugaan model dan melihat orde SARIMA yang memungkinkan dalam pemodelan, (3) menguji kelayakan model dengan uji signifikansi parameter dan uji diagnostik residu. Uji diagnostik residu meliputi uji normalitas, uji independensi, dan uji

homoskedastisitas. Model yang memenuhi semua uji digunakan dalam tahap selanjutnya, (4) meramalkan data *testing* berdasarkan model yang telah didapatkan, dan (5) menentukan model SARIMA terbaik berdasarkan perhitungan nilai MAPE terkecil.

Selanjutnya metode penelitian yang digunakan untuk melakukan tahapan metode ELM diuraikan sebagai berikut: (1) menormalisasikan data dengan metode Min-Max, sehingga data untuk *input* ELM berada pada interval yang sama, (2) melakukan tahapan *training* untuk mendapatkan nilai *input weight*, bias, dan *output weight* yang optimal, (3) melakukan tahapan *testing* untuk mengevaluasi hasil peramalan dari metode ELM, (4) melakukan percobaan kombinasi neuron pada *hidden layer* agar mendapatkan arsitektur jaringan yang paling optimal, (5) melakukan denormalisasi data untuk mengembalikan ke data asli, dan (6) menentukan model ELM terbaik berdasarkan perhitungan nilai MAPE terkecil. Langkah terakhir yaitu membandingkan hasil peramalan metode SARIMA dan metode ELM berdasarkan nilai MAPE.

4. HASIL DAN PEMBAHASAN

Data peredaran uang kartal yang telah dibagi menjadi data *training* dan data *testing*, selanjutnya dilakukan pembuatan plot data *training* untuk melihat pola dari data tersebut. Plot runtun waktu data *training* dapat dilihat pada Gambar 2.

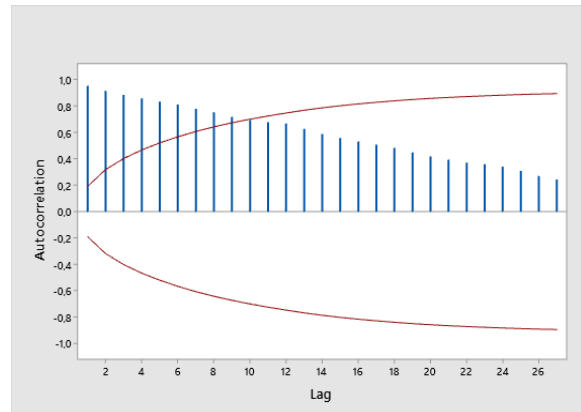


Gambar 2. Plot runtun waktu data *training*

Gambar 2 menunjukkan pola tren naik dan terlihat mengalami fluktuasi yang hampir sama, yaitu memiliki puncak tertinggi setiap tahunnya yang berjangka sekitar 12 periode bulan. Oleh karena itu, diindikasikan bahwa data *training* peredaran uang kartal di Indonesia dari Bulan Januari 2011 sampai Januari 2020 menunjukkan pola tren naik yang mengandung musiman.

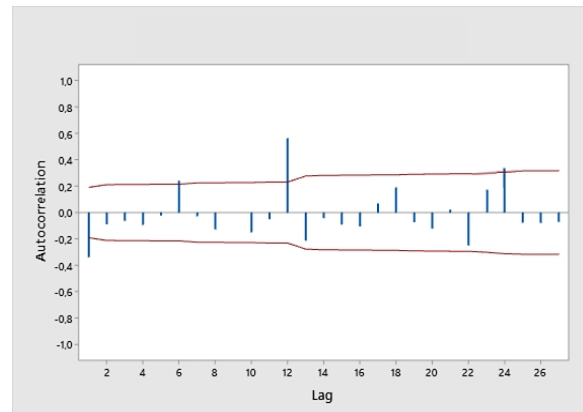
4.1. Pemodelan SARIMA

Pemodelan SARIMA membutuhkan suatu data yang telah stasioner untuk menemukan model yang memungkinkan, sehingga perlu dilakukan pengecekan plot ACF untuk memastikan pola pada data tersebut. Plot ACF untuk data *training* ditunjukkan pada Gambar 3.



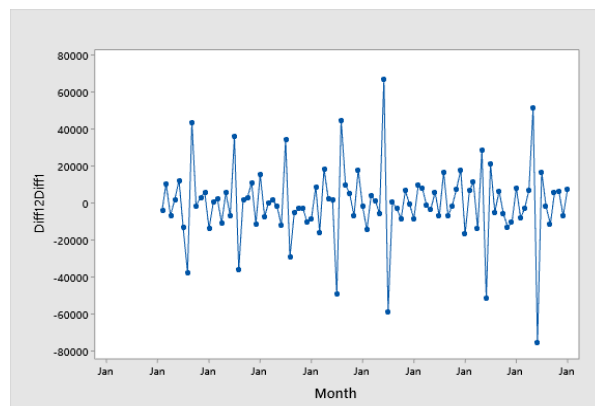
Gambar 3. Plot ACF data *training*

Gambar 3 menunjukkan lag menurun secara perlahan seiring dengan bertambahnya lag yang berarti bahwa data memiliki pola tren. Hal tersebut menandakan bahwa data peredaran uang kartal belum stasioner. Data yang belum stasioner tersebut dilakukan *differencing* dengan lag 1 karena plot ACF memiliki pola tren. Plot ACF setelah dilakukan *differencing* pertama dapat dilihat pada Gambar 4.

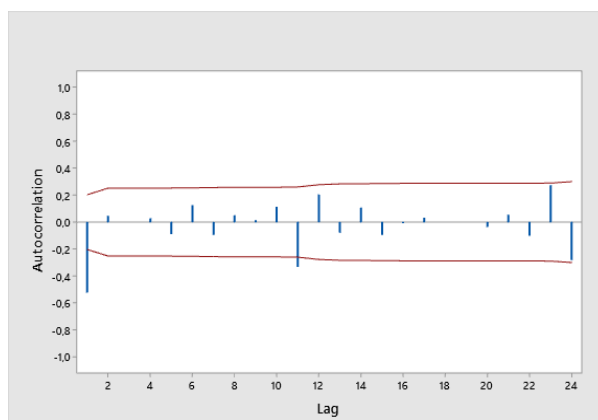


Gambar 4. Plot ACF *differencing 1*

Gambar 4 menunjukkan bahwa data hasil *differencing* pertama masih belum stasioner karena terdapat lag musiman yang menurun secara perlahan. Oleh karena itu, dilakukan *differencing* kedua dengan lag 12 karena lag 12 dan 24 keluar dari batas pita konfidensi. Plot data runtun waktu dan plot ACF setelah dilakukan *differencing* kedua dapat dilihat pada Gambar 5 dan Gambar 6.



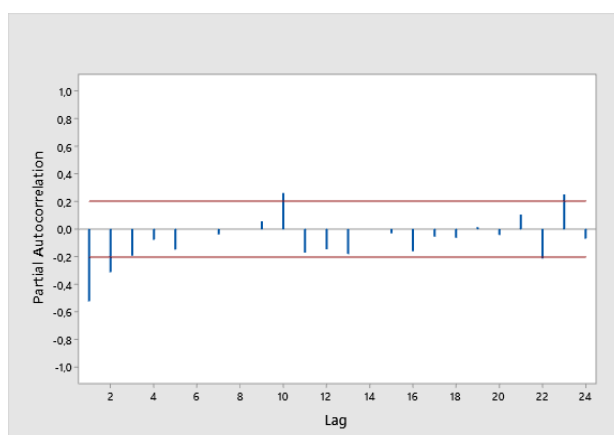
Gambar 5. Plot data runtun waktu *differencing 1 differencing 12*



Gambar 6. Plot ACF *differencing 1 differencing 12*

Hasil *differencing* kedua menunjukkan bahwa plot runtun waktu pada Gambar 5 semakin berfluktuasi di sekitar nilai rata-rata yang konstan dan plot ACF pada Gambar 6 yang terputus (*cut off*) setelah lag 1, sehingga data peredaran uang kartal sudah stasioner.

Selanjutnya untuk lebih memastikan kestasioneran data dalam rata-rata, maka dilakukan uji stasioneritas Augmented Dickey-Fuller (ADF). Hasil uji stasioner menunjukkan *p-value* kurang dari nilai α (0,05), sehingga data peredaran uang kartal di Indonesia sudah stasioner terhadap rata-rata. Selanjutnya dilakukan uji Box-Cox untuk melihat kestasioneran data dalam variansi. Hasil pengujian menunjukkan nilai λ (lambda) yaitu 1,030756, sehingga data peredaran uang kartal di Indonesia dari Bulan Januari 2011 sampai April 2022 sudah stasioner terhadap variansi.



Gambar 7. Plot PACF *differencing 1 differencing 12*

Langkah berikutnya yaitu penentuan identifikasi model SARIMA. Berdasarkan Gambar 6 dan Gambar 7, kombinasi orde yang memungkinkan untuk pemodelan SARIMA yaitu SARIMA(2,1,1)(0,1,0)₁₂ ; SARIMA(2,1,0)(0,1,0)₁₂ ; SARIMA(1,1,1)(0,1,0)₁₂ ; SARIMA(1,1,0)(0,1,0)₁₂; dan SARIMA(0,1,1)(0,1,0)₁₂. Semua model yang diperoleh dilanjutkan untuk dilihat signifikansi parameternya. Hasil estimasi parameter dan signifikansinya disajikan pada Tabel 1.

Tabel 1. Hasil estimasi dan signifikansi parameter

Model	Parameter	Koefisien	<i>p-value</i>	Kesimpulan
SARIMA (2,1,1)(0,1,0) ₁₂	ϕ_1	-0,176	0,367	Tidak Signifikan
	ϕ_2	-0,048	0,753	Tidak Signifikan
	θ_1	0,594	0,00056	Signifikan

Model	Parameter	Koefisien	<i>p-value</i>	Kesimpulan
SARIMA (2,1,0)(0,1,0) ₁₂	ϕ_1	-0,681	$1,8 \cdot 10^{-12}$	Signifikan
	ϕ_2	-0,307	0,0014	Signifikan
SARIMA (1,1,1)(0,1,0) ₁₂	ϕ_1	-0,133	0,338	Tidak Signifikan
	θ_1	0,634	$3,1 \cdot 10^{-9}$	Signifikan
SARIMA (1,1,0)(0,1,0) ₁₂	ϕ_1	-0,519	$1,8 \cdot 10^{-9}$	Signifikan
SARIMA (0,1,1)(0,1,0) ₁₂	θ_1	0,702	$2,2 \cdot 10^{-16}$	Signifikan

Berdasarkan Tabel 1 menunjukkan model yang memenuhi uji signifikansi parameter adalah SARIMA (2,1,0)(0,1,0)₁₂ ; SARIMA (1,1,0)(0,1,0)₁₂ ; dan SARIMA(0,1,1)(0,1,0)₁₂. Ketiga model selanjutnya dilakukan uji diagnostik residu. Uji ini meliputi uji normalitas, uji independensi, dan uji homoskedastisitas. Hasil uji diagnostik residu ditampilkan pada Tabel 2.

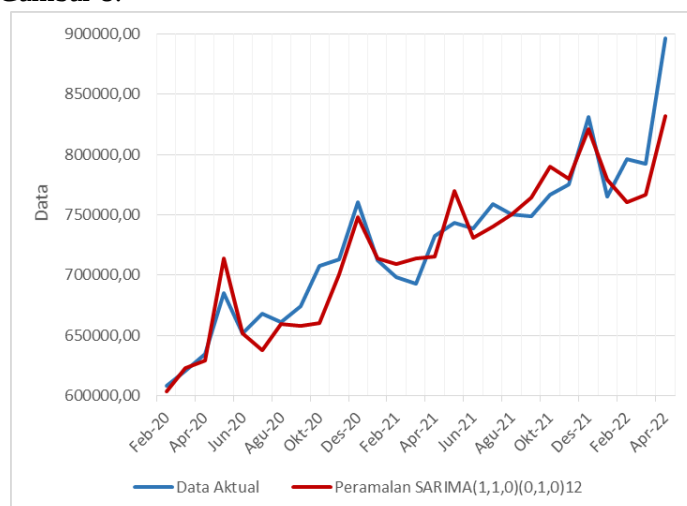
Tabel 2. Hasil uji diagnostik residu

Model	Uji Normalitas	Uji Independensi	Uji homoskedastisitas
SARIMA(2,1,0)(0,1,0) ₁₂	Tidak memenuhi	Memenuhi	Tidak memenuhi
SARIMA(1,1,0)(0,1,0) ₁₂	Memenuhi	Memenuhi	Memenuhi
SARIMA(0,1,1)(0,1,0) ₁₂	Memenuhi	Memenuhi	Tidak memenuhi

Dengan demikian, model SARIMA (1,1,0)(0,1,0)₁₂ telah memenuhi asumsi normalitas, independensi, dan homoskedastisitas residu. Berikut persamaan model SARIMA(1,1,0)(0,1,0)₁₂ :

$$\hat{X}_t = 0,481X_{t-1} + 0,519X_{t-2} + X_{t-12} - 0,481X_{t-13} - 0,519X_{t-14}$$

Langkah berikutnya adalah meramalkan data *testing* menggunakan kedua model SARIMA yang telah diperoleh. Visualisasi plot perbandingan data *testing* dengan hasil peramalan data *testing* ditunjukkan pada Gambar 8.



Gambar 8. Plot data aktual dan data peramalan

Gambar 8 menunjukkan bahwa peramalan data *testing* model SARIMA yang ditandai dengan warna merah memiliki nilai yang lumayan sama dengan data aktualnya yang berwarna biru. Model SARIMA(1,1,0)(0,1,0)₁₂ memiliki hasil peramalan yang tidak jauh berbeda dengan data aktual *testing*.

Ukuran kesalahan peramalan model SARIMA(1,1,0)(0,1,0)₁₂ dapat dihitung dengan menggunakan nilai MAPE pada data *training* dan data *testing* seperti pada Tabel 3.

Tabel 3. Ringkasan nilai MAPE

Data	MAPE
<i>Training</i>	2,3270%
<i>Testing</i>	2,2772%

Tabel 3 menunjukkan bahwa nilai MAPE baik pada data *training* maupun data *testing* dari model SARIMA(1,1,0)(0,1,0)₁₂ tergolong kecil karena kurang dari 10%, sehingga model tersebut termasuk memiliki kinerja peramalan yang sangat baik dalam meramalkan peredaran uang kartal di Indonesia.

4.2. Pemodelan ELM

Data runtun waktu peredaran uang kartal yang digunakan pada pemodelan ELM ini adalah data hasil *differencing* non musiman dengan lag 1. Langkah tersebut telah dilakukan pada subbab 4.1. *Differencing* lag 1 dalam pemodelan ELM bertujuan untuk menghilangkan pola tren pada data peredaran uang kartal, sehingga menghasilkan peramalan yang lebih optimal. Selanjutnya data dinormalisasikan dengan rentang jangkauan antara 0 sampai 1 menggunakan metode *min-max normalization*. Normalisasi data ini bertujuan untuk menyesuaikan nilai rentang data dengan fungsi aktivasi pada JST. Dengan demikian, *output* yang dihasilkan juga berada pada rentang 0 sampai 1, sehingga nantinya dilakukan denormalisasi data untuk mengetahui data aktualnya. Berikut salah satu contoh cara perhitungan normalisasi data pada Bulan Februari 2011.

$$X'_i = X_{\min b} + \frac{(X_t - X_{\min})(X_{\max b} - X_{\min b})}{(X_{\max} - X_{\min})}$$

$$X'_2 = 0 + \frac{(-2154,32 - (-66218,60))(1 - 0)}{(103799,37 - (-66218,60))}$$

$$X'_2 = \frac{64064,28}{170017,97}$$

$$X'_2 = 0,3768$$

Setelah tahap normalisasi data selesai, selanjutnya yaitu tahapan *training*. *Input* yang digunakan pada tahapan *training* ini sebanyak 12 *input* yaitu $X_1, X_2, X_3, \dots, X_{12}$ karena data peredaran uang kartal merupakan data bulanan, sehingga diasumsikan sebagai perwakilan dari 12 periode yang terjadi dalam 1 tahun. Sedangkan Y merupakan nilai target pada tahapan *training*. Pembentukan data *input* pola ke-1 berdasarkan dari hasil *differencing* dan normalisasi data ke-1 sampai data ke-12 yang dijadikan sebagai X_1 sampai X_{12} , sedangkan data ke-13 menjadi Y . Selanjutnya pola ke-2 berdasarkan data ke-2 sampai data ke-13 yang dijadikan sebagai X_1 sampai X_{12} , sedangkan data ke-14 menjadi Y . Langkah tersebut diulangi sampai mendapatkan pola input ke-96. Data *input* ELM menggunakan data *training* tersebut disajikan pada Tabel 4.

Tabel 4. Data *input* ELM

Pola	X1	X2	X3	X4	X5	...	X12	Y
1	0,377	0,368	0,451	0,402	0,433	...	0,263	0,353
2	0,368	0,451	0,402	0,433	0,471	...	0,353	0,430
3	0,451	0,402	0,433	0,471	0,679	...	0,430	0,412
4	0,402	0,433	0,471	0,679	0,122	...	0,412	0,412
5	0,433	0,471	0,679	0,122	0,402	...	0,412	0,507
6	0,471	0,679	0,122	0,402	0,376	...	0,507	0,394

Pola	X1	X2	X3	X4	X5	...	X12	Y
7	0,679	0,122	0,402	0,376	0,558	...	0,394	0,458
8	0,122	0,402	0,376	0,558	0,263	...	0,458	0,381
9	0,402	0,376	0,558	0,263	0,353	...	0,381	0,393
10	0,376	0,558	0,263	0,353	0,430	...	0,393	0,395
...	...	...	...	...	...	...	...	...
94	0,417	0,620	0,118	0,337	0,479	...	0,371	0,456
95	0,620	0,118	0,337	0,479	0,433	...	0,456	0,579
96	0,118	0,337	0,479	0,433	0,876	...	0,579	0,163

Arsitektur jaringan ELM terdiri dari jumlah unit *input*, jumlah unit *hidden layer*, dan jumlah unit *output*. Jumlah neuron pada *hidden layer* ditentukan berdasarkan hasil *trial and error* antara 5 – 100 neuron dengan kelipatan 5. Setelah pembentukan input ELM, langkah selanjutnya adalah menginisialisasi *input weight* dan bias secara acak dengan bantuan *software* Rstudio. Matriks *input weight* dan bias dibentuk berdasarkan jumlah *input* neuron dan *hidden* neuron paling optimal, sehingga pada subbab ini matriks *input weight* dan bias sudah melalui *trial and error* dari jumlah *hidden* neuron. Berdasarkan 12 *input* neuron dan 45 *hidden* neuron, diperoleh matriks w_j berukuran 12×45 dan vektor b_j berukuran 1×45 sebagai berikut:

$$w_j = \begin{bmatrix} 0,069 & 0,376 & \dots & 0,465 \\ 0,818 & 0,381 & \dots & 0,441 \\ \vdots & \vdots & \ddots & \vdots \\ 0,127 & 0,351 & \dots & 0,551 \end{bmatrix}$$

$$b_j = [0,288 \quad 0,788 \quad \dots \quad 0,152]$$

Langkah berikutnya yaitu menghitung matriks H_{init} yang berukuran 96×45 dengan rumus:

$$H_{init} = (x_i w_j) + b_j$$

$$H_{init} = \left(\begin{bmatrix} 0,377 & \dots & 0,263 \\ \vdots & \ddots & \vdots \\ 0,118 & \dots & 0,579 \end{bmatrix} \cdot \begin{bmatrix} 0,069 & \dots & 0,465 \\ \vdots & \ddots & \vdots \\ 0,127 & \dots & 0,551 \end{bmatrix} \right) + [0,288 \quad 0,788 \quad \dots \quad 0,152]$$

$$H_{init} = \begin{bmatrix} 1,882 & \dots & 2,871 \\ \vdots & \ddots & \vdots \\ 2,035 & \dots & 2,544 \end{bmatrix}$$

Setelah didapatkan matriks H_{init} , kemudian mencari matriks H menggunakan rumus hasil substitusi dari fungsi aktivasi *sigmoid biner*. Matriks H berukuran 96×45 dengan rumus:

$$H = \frac{1}{1 + \exp(-H_{init})}$$

$$H = 1 / 1 + \exp \left(- \begin{bmatrix} 1,882 & \dots & 2,871 \\ \vdots & \ddots & \vdots \\ 2,035 & \dots & 2,544 \end{bmatrix} \right)$$

$$H = \begin{bmatrix} 0,868 & \dots & 0,946 \\ \vdots & \ddots & \vdots \\ 0,884 & \dots & 0,927 \end{bmatrix}$$

Langkah selanjutnya yaitu menghitung matriks $H^\dagger$ yang berukuran 45×96 dengan rumus perhitungan sebagai berikut:

$$H^\dagger = ((H)^T H)^{-1} H^T$$

$$H^{\dagger} = \left(\begin{bmatrix} 0,868 & \cdots & 0,946 \\ \vdots & \ddots & \vdots \\ 0,884 & \cdots & 0,927 \end{bmatrix}^T \begin{bmatrix} 0,868 & \cdots & 0,946 \\ \vdots & \ddots & \vdots \\ 0,884 & \cdots & 0,927 \end{bmatrix} \right)^{-1} \cdot \begin{bmatrix} 0,868 & \cdots & 0,946 \\ \vdots & \ddots & \vdots \\ 0,884 & \cdots & 0,927 \end{bmatrix}^T$$

$$H^{\dagger} = \begin{bmatrix} 777,14 & \cdots & 294,52 \\ \vdots & \ddots & \vdots \\ 294,52 & \cdots & 7013,28 \end{bmatrix} \cdot \begin{bmatrix} 0,868 & \cdots & 0,884 \\ \vdots & \ddots & \vdots \\ 0,946 & \cdots & 0,927 \end{bmatrix}$$

$$H^{\dagger} = \begin{bmatrix} -6,709 & \cdots & 5,151 \\ \vdots & \ddots & \vdots \\ -6,247 & \cdots & 9,793 \end{bmatrix}$$

Langkah terakhir dari tahapan *training* yaitu mencari hasil berupa *output weight* berukuran 45×1 dengan rumus perhitungan:

$$\beta = H^{\dagger} T$$

$$\beta = \begin{bmatrix} -6,709 & \cdots & 5,151 \\ \vdots & \ddots & \vdots \\ -6,247 & \cdots & 9,793 \end{bmatrix} \cdot \begin{bmatrix} 0,353 \\ \vdots \\ 0,163 \end{bmatrix}$$

$$\beta = \begin{bmatrix} 3,565 \\ \vdots \\ -9,290 \end{bmatrix}$$

Nilai *input weight*, bias, dan *output weight* yang telah didapatkan selanjutnya digunakan dalam tahapan *testing* untuk melakukan peramalan data dengan rumus perhitungan:

$$y = H\beta$$

$$y = \begin{bmatrix} 0,868 & \cdots & 0,946 \\ \vdots & \ddots & \vdots \\ 0,884 & \cdots & 0,927 \end{bmatrix} \cdot \begin{bmatrix} 3,565 \\ \vdots \\ -9,290 \end{bmatrix}$$

$$y = \begin{bmatrix} 0,406 \\ \vdots \\ 0,198 \end{bmatrix}$$

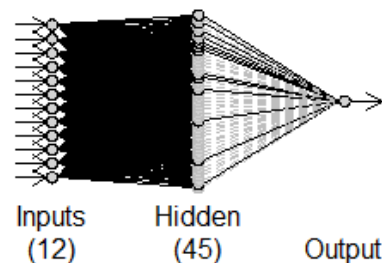
Selanjutnya, data hasil peramalan didenormalisasikan terlebih dahulu, kemudian membandingkan nilai MAPE untuk mendapatkan jumlah neuron pada *hidden layer* paling optimal. Tabel 5 menunjukkan perbandingan jumlah neuron pada *hidden layer*.

Tabel 5. Perbandingan jumlah neuron *hidden layer*

Neuron	MAPE Training (%)	MAPE Testing (%)
5	5,753	4,905
10	5,211	7,664
15	4,666	4,139
20	4,765	9,486
25	5,642	5,518
30	4,302	14,507
35	4,631	4,612
40	4,453	12,960
45	4,255	3,862
50	4,393	12,904
55	5,002	5,349

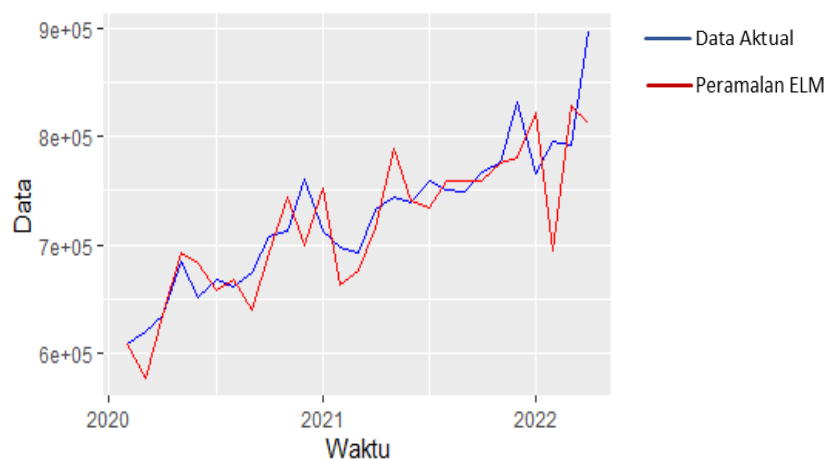
Neuron	MAPE <i>Training</i> (%)	MAPE <i>Testing</i> (%)
60	4,507	35,222
65	4,238	8,238
70	4,227	16,827
75	4,210	7,520
80	4,020	19,827
85	4,210	5,994
90	3,855	21,589
95	3,805	6,092
100	3,805	6,092

Model menghasilkan arsitektur jaringan terbaik berdasarkan MAPE *testing* terkecil yaitu dengan 12 neuron *input layer*, 45 neuron *hidden layer*, dan 1 *output layer*. Arsitektur jaringan ELM ditampilkan pada Gambar 9.



Gambar 9. Arsitektur jaringan ELM

Setelah didapatkan arsitektur terbaik, maka dilanjutkan peramalan data *testing* berdasarkan arsitektur jaringan ELM dengan 12 neuron *input layer*, 45 neuron *hidden layer*, dan 1 *output layer*. Visualisasi data *testing* dan peramalannya dapat dilihat pada Gambar 10.



Gambar 10. Plot data aktual dan data peramalan

Gambar 10 menunjukkan hasil peramalan data *testing* berwarna merah ternyata memiliki nilai yang polanya berbeda, namun masih lumayan dekat dengan data *testing* berwarna biru.

Langkah selanjutnya yaitu melihat ukuran kesalahan model ELM yang diperoleh dengan melakukan pengecekan kembali nilai MAPE pada data *training* dan data *testing* yang ditunjukkan pada Tabel 6.

Tabel 6. Ringkasan nilai MAPE

Data	MAPE (%)
<i>Training</i>	4,2548
<i>Testing</i>	3,8615

Tabel 6 menunjukkan bahwa MAPE pada data *training* maupun data *testing* bernilai kurang dari 10%, sehingga model ELM memiliki kinerja yang tergolong sangat baik.

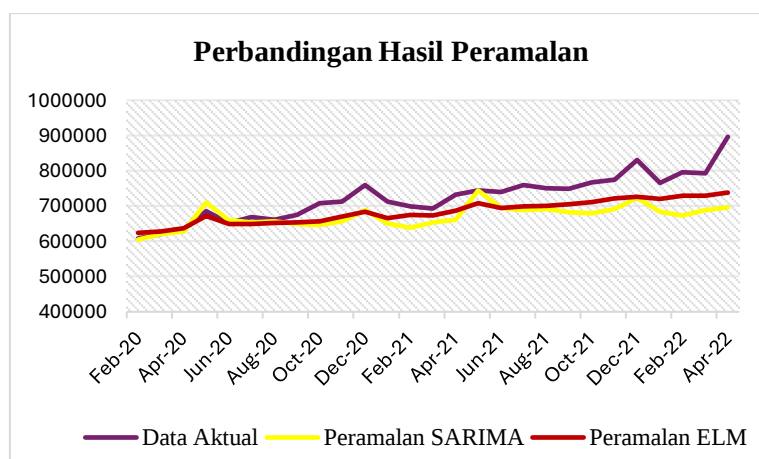
4.3. Perbandingan Hasil Peramalan Metode SARIMA dan ELM

Model terbaik dari masing-masing metode SARIMA dan ELM selanjutnya dibandingkan berdasarkan nilai MAPE. Perbandingan nilai MAPE kedua metode dapat dilihat pada Tabel 7. Hasil peramalan 1 periode kedepan di bulan Mei 2022 untuk metode SARIMA yaitu senilai 778225,5, sedangkan hasil peramalan metode ELM di bulan Mei 2022 yaitu senilai 740246,3.

Tabel 7. Perbandingan MAPE metode SARIMA dan ELM

Metode	MAPE <i>Training</i> (%)	MAPE <i>Testing</i> (%)
SARIMA	2,3270	2,2772
ELM	4,2548	3,8615

Tabel 7 menunjukkan bahwa nilai MAPE pada kedua metode memiliki nilai kurang dari 10% yang artinya hasil peramalan tergolong sangat baik. Metode SARIMA memiliki nilai MAPE yang lebih kecil daripada metode ELM. Dengan demikian, metode SARIMA lebih baik digunakan dalam meramalkan data peredaran uang kartal di Indonesia. Hasil plot runtun waktu perbandingan data aktual, peramalan SARIMA, dan peramalan ELM juga dapat dilihat pada Gambar 11.



Gambar 11. Plot perbandingan data aktual, peramalan SARIMA, dan peramalan ELM

5. KESIMPULAN

Model SARIMA terbaik untuk peramalan data peredaran uang kartal di Indonesia adalah SARIMA (1,1,0)(0,1,0)₁₂, sedangkan model ELM terbaik dengan arsitektur jaringan 12 neuron *input layer*, 45 neuron *hidden layer*, dan 1 *output layer*.

Perbandingan akurasi hasil peramalan menggunakan metode SARIMA dan ELM tergolong sangat baik. Hal tersebut berdasarkan ukuran kesalahan peramalan MAPE, baik pada data *training* maupun *testing* memiliki nilai kurang dari 10%. Metode SARIMA memiliki MAPE data *training*

sebesar 2,3270% dan pada data *testing* sebesar 2,2772%, sedangkan metode ELM memiliki MAPE data *training* sebesar 4,2548% dan pada data *testing* sebesar 3,8615%. Oleh karena itu, metode SARIMA lebih baik daripada metode ELM dalam meramalkan peredaran uang kartal di Indonesia karena nilai MAPE metode SARIMA lebih kecil daripada metode ELM.

Saran yang dapat diberikan dalam penelitian ini yaitu dengan mencoba pengembangan metode lain dalam meramalkan peredaran uang kartal, mengingat cakupan analisis peramalan yang sangat luas. Metode SARIMA masih dapat dikembangkan dengan metode lain, misalnya ARIMAX maupun SARIMAX. Kemudian untuk metode ELM dapat dibandingkan dengan metode JST lainnya, seperti Backpropagation.

DAFTAR PUSTAKA

- [1] Bank Indonesia, Daftar Istilah Sistem Pembayaran Indonesia. Jakarta : Bank Indonesia, 2014, <https://www.bi.go.id/id/fungsi-utama/sistem-pembayaran/default.aspx>.
- [2] S. V. Ranita, "Analisis volatilitas jumlah uang beredar di Indonesia," Medan: Program Studi Ilmu Ekonomi, Universitas Negeri Medan, 2010.
- [3] S. I. Y. Ardian, T. Rismawan, dan D. M. Midyanti, "Penerapan metode extreme learning machine untuk peramalan jumlah kepadatan penduduk di Kalimantan Barat," *Coding: Jurnal Komputer dan Aplikasi*, vol. 7, no. 3, pp. 34-43, 2019, doi: <http://dx.doi.org/10.26418/coding.v7i03.37147>.
- [4] B. G. Prianda dan E. Widodo, "Perbandingan metode seasonal ARIMA dan extreme learning machine pada peramalan jumlah wisatawan mancanegara ke Bali," *Barekeng : Jurnal Ilmu Matematika dan Terapan*, vol. 15, no. 4, pp. 639-650, 2021, doi: <https://doi.org/10.30598/barekengvol15iss4pp639-650>.
- [5] M. I. Rizki dan T. A. Taqiyyuddin, "Penerapan model SARIMA untuk memprediksi tingkat inflasi di Indonesia," *Jurnal Sains Matematika dan Statistika*, vol. 7, no. 2, pp. 62-72, 2021, doi: <https://dx.doi.org/10.24014/jsms.v7i2.13168>.
- [6] P. Utomo dan A. Fanani, "Peramalan jumlah penumpang kereta api di indonesia menggunakan metode seasonal autoregressive integrated moving average (SARIMA)," *Jurnal Mahasiswa Matematika ALGEBRA*, vol. 1, no. 1, pp. 169-178, 2020, <https://api.core.ac.uk/oai/oai:oai.jurnalsaintek.uinsby.ac.id:article/1030>.
- [7] A. Yahya, "Peramalan indeks harga konsumen indonesia menggunakan metode seasonal-arima (SARIMA)," *Jurnal Gaussian*, vol. 11, no. 2, pp. 313-322, 2022, doi: <https://doi.org/10.14710/j.gauss.v11i2.35528>.
- [8] N. P. Hendayanti dan M. Nurhidayati, "Perbandingan metode seasonal autoregressive integrated moving average (SARIMA) dengan support vector regression (SVR) dalam memprediksi jumlah kunjungan wisatawan mancanegara ke Bali," *Jurnal Varian*, vol. 3, no. 2, pp. 149-162, 2020, doi: <https://doi.org/10.30812/varian.v3i2.668>.
- [9] D. A. Lubis, M. B. Johra, dan G. Darmawan, "Peramalan indeks harga konsumen dengan metode singular spectral analysis (SSA) dan seasonal autoregressive integrated moving average (SARIMA)," *Jurnal Matematika (MANTIK)*, vol. 3, no. 2, pp. 74-82, 2017, doi: <https://doi.org/10.15642/mantik.2017.3.2.74-82>.
- [10] N. A. Izati, B. Warsito, dan T. Widiari, "Prediksi harga emas menggunakan feed forward neural network dengan metode extreme learning machine," *Jurnal Gaussian*, vol. 8, no. 2, pp. 171-183, 2019, doi: <https://doi.org/10.14710/j.gauss.8.2.171-183>.
- [11] M. Karmila dan I. Nirmala, "Prediksi jumlah produksi kebutuhan air pada perumda air minum tirta khatulistiwa Pontianak menggunakan metode extreme learning machine

- (ELM),” *Coding: Jurnal Komputer dan Aplikasi*, vol. 11, no. 1, pp. 137-146, 2023, doi: <http://dx.doi.org/10.26418/coding.v11i1.58052>.
- [12] M. S. Libianto, Tibyani, dan Y. A. Sari, “Prediksi jumlah kunjungan wisatawan mancanegara pada negara Singapura menggunakan algoritme extreme learning machine,” *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 3, no. 10, pp. 10324-10330, 2019, <https://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/6667>.
- [13] A. N. Alfiyatin, W. F. Mahmudy, C. F. Ananda, dan Y. P. Anggodo, “Penerapan extreme learning machine (ELM) untuk peramalan laju inflasi di Indonesia,” *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIK)*, vol. 6, no. 2, pp. 179-186, 2019, doi: <http://dx.doi.org/10.25126/jtiik.201962900>.
- [14] A. R. Dimashanti dan Sugiman, “Peramalan indeks harga konsumen kota semarang menggunakan SARIMA berbantuan Software Minitab,” *Prosiding Seminar Nasional Matematika (PRISMA)*, vol. 4, pp. 565-576, 2021, <https://journal.unnes.ac.id/sju/prisma/article/view/45030>.
- [15] W.W.S. Wei, *Time Series Analysis Univariate and Multivariate Methode*. Canada : Pearson Eddison Wesley, 2006.
- [16] G. B. Huang, Q. Y. Zhu, and C. K. Siew, “Extreme learning machine theory and applications,” *Neurocomputing*, vol. 70, pp. 489-501, 2006, doi: <http://dx.doi.org/10.1016/j.neucom.2005.12.126>.
- [17] B. K. Khotimah, E. M. Sari, dan H. Yulianarta, “Kinerja metode extreme learning machine pada sistem peramalan,” *Jurnal SimanteC*, vol. 1, no. 3, pp. 186-191, 2010, doi: <https://doi.org/10.21107/simantec.v1i3.13387>.
- [18] S. Makridakis, S.C. Wheelwright, and V.E. McGee, *Forecating: Methods and Applications*. New York : John Wiley & Sons, 1983. Terjemahan U.S. Andriyanto dan A. Basith, *Metode dan Aplikasi Peramalan*. Jakarta : Erlangga, 1999.