BAB III
METODOLOGI PENELITIAN

3.1. Jenis Penelitian

Berdasarkan pada tujuannya penelitian yang dilakukan ini tergolong ke dalam penelitian terapan, di mana penelitian terapan bertujuan untuk menerapkan, menguji dan mengevaluasi masalah praktis sehigga dapat bermanfaat untuk kehidupan manusia. Hal tersebut sesuai dengan penelitian yang terfokus pada penerapan model peramalan runtun waktu. Selain itu penelitian ini juga tergolong pada penelitian kuantitatif karena bersifat numerik, terdapat uji hipotesis dan juga hasil dari penelitian dijelaskan secara deskriptif.

3.2. Jenis dan Sumber Data

Pada penelitian ini jenis data yang digunakan adalah data sekunder. Data yang digunakan terdiri dari data inflasi yang terjadi di Indonesia, data jumlah uang beredar di Indonesia dan data nilai tukar rupiah terhadap dolar Amerika diperoleh dari *website* resmi Bank Indonesia (BI) dan Kementrian Perdagangan.

3.3. Variabel Penelitian

Terdapat tiga variabel penelitian yang digunakan disini yaitu, inflasi sebagai variabel output, dan dua variabel input yang terdiri dari persentase perubahan jumlah uang beredar serta persentase perubahan kurs dolar Amerika.

3.3.1. Inflasi di Indonesia (y_t)

Pada penelitian ini, inflasi yang terjadi di Indonesia dijadikan sebagai variabel output . Inflasi merupakan kenaikan harga barang dan jasa yang meningkat secara umum dan terus-menerus dalam jangka waktu tertentu. Pada penelitian ini data inflasi yang digunakan itu dimulai dari bulan Januari 2010 hingga Januari 2022.

3.3.2. Jumlah Uang Beredar di Indonesia (x_{1t})

Pada penelitian ini, jumlah uang beredar (JUB) di Indonesia yang digunakan adalah persentase perubahannya serta datanya dijadikan sebagai variabel input pertama. Jumlah uang beredar merupakan nilai keseluruhan dari uang yang beredar di masyarakat. Pada penelitian ini data jumlah uang yang beredar yang digunakan itu dimulai dari bulan Januari 2010 hingga Januari 2022.

3.3.3. Kurs Dolar Amerika (x_{2t})

Pada penelitian ini, kurs dolar Amerika yang digunakan adalah persentase perubahannya serta datanya dijadikan sebagai variabel input yang kedua. Kurs dolar Amerika merupakan harga satu unit mata uang rupiah terhadap mata uang dolar Amerika. Pada penelitian ini data nilai tukar rupiah terhadap dolar Amerika yang digunakan itu dimulai dari bulan Januari 2010 hingga Januari 2022.

3.4. Pembentukkan Model SARIMAX dengan Fungsi Transfer

3.4.1. Identifikasi Bentuk Model *Single-input*

a) Persiapan runtun input dan runtun output

Pada tahap ini dilakukan identifikasi karakteristik data, seperti kestasioneran data dan orde SARIMA. Apabila data belum stasioner dalam rata-rata, maka perlu dilakukan *differencing*. Sedangkan jika data belum stasioner dalam varian maka perlu dilakukan transformasi terhadap data.

b) *Prewhitening*

Pada tahap ini dilakukan *prewhitening* terhadap runtun input dengan menggunakan persamaan (2.30). Setelah itu dilakukan juga *prewhitening* terhadap runtun output dengan menggunakan persamaan (2.31).

c) Perhitungan CCF runtun input dan runtun output yang telah dilakukan *prewhitening*

Fungsi korelasi silang merupakan ukuran kekuatan hubungan antara dua variabel. Korelasi silang antara *X* dan *Y* menentukan ukuran kekuatan hubungan antar nilai *X* pada waktu ke-*t* dengan nilai dari *Y* pada waktu ke-*t*-*1*. CCF dari runtun input dengan runtun output untuk lag ke-*k* didefinisikan pada persamaan (2.34) yang nantinya ditaksir oleh persamaan (2.35).

Perhitungan korelasi silang ini digunakan unntuk penentuan nilai dari *r*,*s*, dan *b* yang diidentifikasi melalui plot dari korelasi silang. Setelah diperoleh nilai dari *r*,*s*,dan *b* pada masing-masing input maka perlu dillakukan korelasi silang serentak antar runtun output dengan seluruh runtun input.

d) Identifikasi model awal fungsi transfer *single-input*

Model fungsi transfer memiiki tiga tingkat yang utama yaitu *r*, *s*, dan *b* yang di mana *r* menunjukkan tingkat dari fungsi $\delta(B)$, *s* menunjukkan tingkat dari fungsi $\omega(B)$ serta *b* menunjukkan keterlambatan sebesar *b* periode sebelum *x* mempengaruhi *y* yang ditulis x_{t-b} pada persamaan (2.26).

Penentuan nilai *r, s* dan *b* berdasarkan pada *lag* dari perhitungan korelasi silang. Nilai dari *r, s* dan *b* tidak terbatas pada 0, 1, 2 tetapi dapat juga lebih dari itu. Namun pada praktiknya, jarang sekali nilainya melebihi itu. Oleh sebab itu, terdapat model fungsi transfer yang sering digunakan yaitu dengan nilai $b = 2$, nilai *r* dan *s* antara 0, 1, atau 2. Berikut tabel yang menunjukkan model fungsi trasfer yang sering digunakan: (Montgomery et al., 2015)

Tabel 3.1. Model fungsi transfer yang sering digunakan

b r s	**Model Fungsi Transfer**
2 0 0	$y_t = \omega_0 x_{t-2}$
2 0 1	$y_t = (\omega_0 - \omega_1 B) x_{t-2}$
2 0 2	$y_t = (\omega_0 - \omega_1 B - \omega_2 B^2) x_{t-2}$
2 1 0	$y_t = \frac{\omega_0}{(1 - \delta_1 B)} x_{t-2}$
2 1 1	$y_t = \frac{(\omega_0 - \omega_1 B)}{(1 - \delta_1 B)} x_{t-2}$
2 1 2	$y_t = \frac{(\omega_0 - \omega_1 B - \omega_2 B^2)}{(1 - \delta_1 B)} x_{t-2}$
2 2 0	$y_t = \frac{\omega_0}{(1 - \delta_1 B - \delta_2 B^2)} x_{t-2}$
2 2 1	$y_t = \frac{(\omega_0 - \omega_1 B)}{(1 - \delta_1 B - \delta_2 B^2)} x_{t-2}$
2 2 2	$y_t = \frac{(\omega_0 - \omega_1 B - \omega_2 B^2)}{(1 - \delta_1 B - \delta_2 B^2)} x_{t-2}$

e) Penaksiran awal runtun *noise* (n_{it})

Penaksiran awal runtun gangguan dapat diperoleh berasarkan pada persamaan berikut ini:

$$n_t = y_t - \frac{\omega(B)}{\delta(B)} x_{t-b} \tag{3.7}$$

f) Identifikasi model SARIMA $(\boldsymbol{p_n}, \boldsymbol{d}, \boldsymbol{q_n})$ $(\boldsymbol{P_n}, \boldsymbol{D}, \boldsymbol{Q_n})^{\boldsymbol{m}}$ dari runtun *noise*

Penentuan runtun gangguan (p_m, q_n) dilakukan dengan menganalisa nillai-nilai n_t dengan menggunakan ARIMA, untuk menentukan model ARIMA yang tepat. Sehingga fungsi $\phi_m(B)$ dan $\theta_n(B)$ untuk runtun gangguan n_t dapat diperoleh dari persamaa berikut:

$$\phi_m(B)n_t = \theta_n(B)\alpha_t \tag{3.8}$$

3.4.2. Estimasi Parameter

Setelah dilakukannya identifikasi model input tuggal,langkah selanjutnya adalah mengestimasi parameter-parameter pada model fungsi transfer *single-input*. Model fungsi transfer sementara yang diperoleh pada persamaan (2.26), terdapat parameter-parameter yang harus di estimasi, diantaranya adalah $\omega, \delta, \phi, \theta$ dan σ^2. Persamaan (2.26) dapat dibentuk persamaan baru dengan mengkalikan $\phi(B)\delta(B)$, sehingga diperoleh persamaan berikut:

$$\phi(B)\delta(B)y_t = \phi(B)\omega(B)x_{t-b} + \delta(B)\theta(B)\varepsilon_t \tag{3.9}$$

Atau dapat juga ditulis sebagai:

$$c(B)y_t = d(B)x_{t-b} + e(B)\varepsilon_t \tag{3.10}$$

di mana:

$$\begin{aligned} c(B) &= \phi(B)\delta(B) \\ &= \left(1 - \phi_1 B - \phi_2 B^2 - \cdots - \phi_p B^p\right)(1 - \delta_1 B - \delta_2 B^2 - \cdots - \delta_r B^r) \\ &= 1 - c_1 B - c_2 B^2 - \cdots - c_{p+r} B^{p+r} \end{aligned}$$

$$\begin{aligned} d(B) &= \phi(B)\omega(B) \\ &= \left(1 - \phi_1 B - \phi_2 B^2 - \cdots - \phi_p B^p\right)(\omega_0 - \omega_1 B - \omega_2 B^2 - \cdots - \omega_s B^s) \\ &= d_0 - d_1 B - d_2 B^2 - \cdots - d_{p+s} B^{p+s} \end{aligned}$$

$$\begin{aligned} e(B) &= \delta(B)\theta(B) \\ &= (1 - \delta_1 B - \delta_2 B^2 - \cdots - \delta_r B^r)(1 - \theta_1 B - \theta_2 B^2 - \cdots - \theta_q B^q) \\ &= 1 - e_1 B - e_2 B^2 - \cdots - e_{r+q} B^{r+q} \end{aligned}$$

Sehingga diperoleh:

$$\begin{aligned} \varepsilon_t = {} & y_t - c_1 y_{t-1} - \cdots - c_{p+r} y_{t-p-r} - d_0 x_{t-b} + d_1 x_{t-b-1} \\ & + d_2 x_{t-b-2} + \cdots + d_{p+s} x_{t-b-p-s} + e_1 \varepsilon_{t-1} + e_2 \varepsilon_{t-2} \\ & + \cdots + e_{r+q} \varepsilon_{t-r-q} \end{aligned} \tag{3.11}$$

di mana $c_i d_j$ dan e_k merupakan fungsi transfer dari $\delta_i, \omega_j, \phi_k$ dan θ_k. Dengan diasumsikan bahwa ε_t sebagai runtun *white noise* yang berdistribusi normal $N(0, \sigma_\alpha^2)$, berikut ini fungsi *likelihood*: (Wei, 2005)

$$L(\delta, \omega, \phi, \theta, \sigma_\alpha^2 \mid b, x, y, x_0, y_0, \varepsilon_0) = (2\pi\sigma_\varepsilon^2)^{-\frac{n}{2}} \exp\left(-\frac{1}{2\sigma_\varepsilon^2}\sum_{t=1}^{n} \alpha_t^2\right) \quad (3.12)$$

di mana x_0, y_0, ε_0 merupakan nilai-nilai awal untuk menghitung α_t.

3.4.3. Uji Diagnostik

Setelah dilakukan identifikasi model dan estimasi parameter, langkah selanjutnya perlu dilakukan pengujian kecocokanmodel sebelum digunakan untuk peramalan. Pada model fungsi transfer diasumsikan bahwa ε_t *white noise* dan *independent* terhadap runtun input x_t, serta *independent* juga terhadap runtun input yang telah diilakukan *prewhitening* (α_t). Penngujian diagnostik darimodel fungsi transfer dilakukan dengan memeriksa residual $\hat{\varepsilon}_t$ dari model *noise,* dan residual α_t dari model input yag telah dilakukan *prewhitening.*

1) Pengujian *cross correlation*

Menurut Wei (2005) sebuah model fungsi transfer dikatakan layak apabila CCF dari runtun *noise* dengan runtun input tidak menunjukkan suatu pola tertentu dan berada antara dua kesalahan yang standar $2(n-k)^{-\frac{1}{2}}$. Pengujian CCF antara runtun *noise* dengan runtun input menggunakan statistik uji Q berikut:

- Perumusan hipotesis

  H_0 : CCF antara runtun *noise* dengan runtun input tidak sginifikan

  H_1 : CCF antara runtun *noise* dengan runtun input sginifikan

- Statistika uji

$$Q = m(m+2)\sum_{i=0}^{k} \frac{r_{\varepsilon_t \alpha_t}^2(k)}{m-i} \quad (3.13)$$

di mana:

$m = n - t_0 + 1$

$t_0 = \max\{p + r + 1, b + p + s + 1\}$

n : Banyaknya pengamatan

K : *lag* maksimum

$r_{a_t \alpha_t}(k)$: Koefisien korelasi silang antara ε_t dengan α_t pada *lag k*

- Kriteria pengujian

  Statistik Q menyebar mengikuti sebaran *chi-square* dengan derajat kebebasan (*K+1-r-s*). Terima H_0 jika $Q < \chi^2{}_{(K+1-r-s)}$ hal tersebut berarti bahwa model fungsi transfer telah layak.

2) Pengujian Autokorelasi

Menurut Wei (2005) sebuah model dikatakan layak apabila FAK dan FAKP dari residual seharusnnya tidak menunjukkan suatu pola tertentu. Pengujian autokorelasi ini menggunakan statistik uji Q sebagai berikut:

- Perumusan hipotesis

  H_0 : Autokorelasi pada residual tidak siginifikan

  H_1 : Autokorelasi pada residual siginifikan

- Statistika uji

$$Q = m(m+2)\sum_{i=1}^{k}\frac{r_{\varepsilon_t}^2(k)}{m-i} \tag{3.14}$$

  di mana:

  $m = n - t_0 + 1$

  $t_0 = \max\{p + r + 1, b + p + s + 1\}$

  n : Banyaknya pengamatan

  K : *lag* maksimum

  $r_{\varepsilon_t}(k)$: Koefisien autokorelasi ε_t pada *lag k*

- Kriteria pengujian

  Statistik Q menyebar mengikuti sebaran *chi-square* dengan derajat kebebasan (*K-p-q*). Terima H_0 jika $Q < \chi^2{}_{(K-p-q)}$ hal tersebut berarti bahwa model fungsi transfer telah layak.

3.4.4. Pembentukan Model Multivariat

Setelah memperoleh model fungsi trasnfer *single-input* yang layak , lanngkah selanjutnya adalah mengestimasikan nilai *r,s* dan *b* pada masing-masing input secara serentak dengan menggunakan metode *least square estimation.* Lalu menentukan deret gangguannya (n_t) dengan menggunakan persamaan berikut:

$$n_t = y_t - \hat{y}_t \tag{3.15}$$

Setelah itu lakukan estimasi paramater kembali dengan mennggunakan model yang baru dan dilanjutkan denngan melakukan uji diagnostik untuk mengetahui kelayakan model, sehigga diperoleh model SARIMAX dengan fungsi trasnfer pada persamaan 2.33.

3.5. Pembentukkan Model ANFIS

Pembentukan model ANFIS dilakukan berdasarkan pada algoritma pelatihan *hybrid*. Terdapat dua langkah dalam pelatihan *hybrid* ini yaitu langkah *forward* dan langkah *backward*. Pada langkah *forward*, parameter premis tetap dengan input jaringan merambat maju hingga pada lapisan ke empat dan parameter konsekuennya akan diidentifikasi menggunakan *least-square estimation* (LSE). Sedangkan pada langkah mundur, *error* sinyal antara output yang diinginkan dengan output aktual,akan merambat mundur dan parameter premis akan diperbaiki dengan metode *gradient-descent*.

3.5.1. *Least-Square Estimation* (LSE) Rekursif

Asumsikan terdapat *m* elemen vektor output *y* dengan *y* berukuran $m \times 1$ dan *n* parameter θ dengan θ berukuran $n \times 1$, dengan baris ke-*i* pada matriks $[A \vdots y]$ yang dapat dinotasikan sebagai $[a_i^T \vdots y]$ maka estimasi *least-square* ditentukan dengan menggunakan persamaan berikut:

$$A^T A\hat{\theta} = A^T y \tag{3.16}$$

Apabila $A^T A$ merupakan *nonsingular* dan $\hat{\theta}$ bersifat unik, maka estimasi *least-square* diperoleh dari persamaan berikut:

$$\hat{\theta} = (A^T A)^{-1} A^T y \tag{3.17}$$

Atau dapat juga ditulis dalam bentuk lain dengan mengasumsikan jumlah baris berpasangan antara *A* dengan *y* adalah *k*, seperti berikut ini:

$$\theta_k = (A^T A)^{-1} A^T y \tag{3.18}$$

Pada LSE rekursif dapat ditambahkan suatu pasangan data $[a_i^T \vdots y]$ dengan a^T merupakan vektor baris pada matriks *A*, sehingga terdapat $m + 1$ pasangan data. Kemudian θ_{k+1} dihitung menggunakan LSE dengan bantuan θ_k. Karena terdapat *n* parameter, maka dapat diselesaikan matriks $n \times n$ menggunakan metode invers, seperti berikut ini:

$$P_n = \left(A_n{}^T A_n\right)^{-1} \tag{3.19}$$

$$\theta_n = P_n A_n{}^T y_n \tag{3.20}$$

Setelah itu iterasi dimulai dari data ke-$(n + 1)$ dengan nilai P_{k+1} dan θ_{k+1} dapat dihitung dengan persamaan berikut:

$$P_{k+1} = P_k - \frac{\left(P_k a_{k+1} a_{k+1}^T P_k\right)}{(1 + a_{k+1}^T P_k a_{k+1})} \tag{3.21}$$

$$\theta_{k+1} = \theta_k + P_{k+1} a_{k+1} (y_{k+1} - a_{k+1}^T \theta_k) \tag{3.22}$$

3.5.2. Model Propagasi *Error*

Propagasi *error* merupakan metode yang digunakan untuk menghitung simpangan dari suatu nilai yang berasal dari beberapa faktor. Model propagasi *error* digunakan untuk memperbaiki parameter premis. Pada model propagasi *error* ini konsep yang digunakan adalah *gradien descent.* Apabila terdapat jaringan adaptif *i* dan *error* pada neuron ke-*j* dengan lapisan ke-*i* dilambangkan dengan ε_{ij} maka perhitungan *error* untuk setiap neuron pada masing-masinglapisan dirumuskan sebagai berikut:

a) *Error* pada *layer* 5

Hanya terdapat satu buah neuron pada *layer* 5 ini, dengan propagasi *error* yang menuju *layer* ini dirumuskan dalam persamaan berikut:

$$\varepsilon_{51} = \frac{\partial E_y}{\partial_f} = -2(y - f) \tag{3.23}$$

di mana:

y : Output target

f : Output jaringan

E_y : Jumlah kuadrat *error* pada *layer* ini

b) *Error* pada *layer* 4

Layer 4 memiliki dua neuron dengan propagasi *error* yang menuju *layer* ini dirumuskan dalam persamaan berikut:

$$\varepsilon_{4j} = \left(\frac{\partial E_y}{\partial_f}\right)\left(\frac{\partial_f}{\partial f_{4j}}\right) \tag{3.24}$$

di mana:

ε_{4j} : *error* pada *layer* 4 dengan neuron ke-*j* $(j = 1{,}2)$

f_{4j} : output neuron ke-*j* pada *layer* 4

Akibat pada *layer* 4 output dari node diperoleh dengan persamaan $f = \sum \overline{w}_j f_j = \overline{w}_1 f_1 + \overline{w}_2 f_2$ maka dapat diperoleh:

$$\frac{\partial_f}{\partial f_{41}} = \frac{\partial_f}{\partial \overline{w}_1 f_1} = 1 \tag{3.25}$$

$$\frac{\partial_f}{\partial f_{42}} = \frac{\partial_f}{\partial \overline{w}_2 f_2} = 1 \tag{3.26}$$

sehingga

$$\varepsilon_{41} = \left(\frac{\partial E_y}{\partial_f}\right)\left(\frac{\partial_f}{\partial f_{41}}\right) = \varepsilon_{51}(1) = \varepsilon_{51} \quad (3.27)$$

$$\varepsilon_{42} = \left(\frac{\partial E_y}{\partial_f}\right)\left(\frac{\partial_f}{\partial f_{42}}\right) = \varepsilon_{51}(1) = \varepsilon_{51} \quad (3.28)$$

c) *Error* pada *layer* 3

Layer 3 memiliki dua neuron dengan propagasi *error* ynag menuju *layer* ini dapat dirumuskan dalam persamaan berikut:

$$\varepsilon_{4j} = \left(\frac{\partial E_y}{\partial_f}\right)\left(\frac{\partial_f}{\partial f_{4j}}\right)\left(\frac{\partial f_{4j}}{\partial f_{3j}}\right) \quad (3.29)$$

di mana:

ε_{3j} : *error* pada *layer* 3 dengan neuron ke-*j* ($j = 1{,}2$)

f_{3j} : output neuron ke-*j* pada *layer* 3

Karena pada *layer* ini $f_{31} = w_1$ dan $f_{32} = w_2$ maka diperoleh:

$$\frac{\partial f_{41}}{\partial f_{31}} = \left(\frac{\partial(\overline{w}_1 f_1)}{\partial(\overline{w}_1)}\right) = f_1 \quad (3.30)$$

$$\frac{\partial f_{42}}{\partial f_{32}} = \left(\frac{\partial(\overline{w}_2 f_2)}{\partial(\overline{w}_2)}\right) = f_2 \quad (3.31)$$

Sehingga

$$\varepsilon_{31} = \left(\frac{\partial E_y}{\partial_f}\right)\left(\frac{\partial_f}{\partial f_{41}}\right)\left(\frac{\partial f_{41}}{\partial f_{31}}\right) = \varepsilon_{51} f_1 \quad (3.32)$$

$$\varepsilon_{32} = \left(\frac{\partial E_y}{\partial_f}\right)\left(\frac{\partial_f}{\partial f_{42}}\right)\left(\frac{\partial f_{42}}{\partial f_{32}}\right) = \varepsilon_{51} f_2 \quad (3.33)$$

d) *Error* pada *layer* 2

Layer 2 memiliki dua buah neuron dengan propagasi *error* yang menuju *layer* ini dapat dirumuskan dalam persamaan berikut:

$$\begin{aligned}\varepsilon_{21} = &\left(\frac{\partial E_y}{\partial_f}\right)\left(\frac{\partial_f}{\partial f_{41}}\right)\left(\frac{\partial f_{41}}{\partial f_{31}}\right)\left(\frac{\partial f_{31}}{\partial f_{21}}\right) \\ &+ \left(\frac{\partial E_y}{\partial_f}\right)\left(\frac{\partial_f}{\partial f_{42}}\right)\left(\frac{\partial f_{42}}{\partial f_{32}}\right)\left(\frac{\partial f_{32}}{\partial f_{21}}\right)\end{aligned} \quad (3.34)$$

$$\varepsilon_{22} = \left(\frac{\partial E_y}{\partial_f}\right)\left(\frac{\partial_f}{\partial f_{41}}\right)\left(\frac{\partial f_{41}}{\partial f_{31}}\right)\left(\frac{\partial f_{31}}{\partial f_{22}}\right) + \left(\frac{\partial E_y}{\partial_f}\right)\left(\frac{\partial_f}{\partial f_{42}}\right)\left(\frac{\partial f_{42}}{\partial f_{32}}\right)\left(\frac{\partial f_{32}}{\partial f_{22}}\right) \tag{3.35}$$

di mana:

ε_{21} : *error* pada *layer* 2 dengan neuron ke-1

ε_{22} : *error* pada *layer* 2 dengan neuron ke-2

f_{31} : output neuron ke-1 pada *layer* 3

f_{31} : output neuron ke-1 pada *layer* 2

Karena pada *layer* ini $f_{21} = w_1$ dan $f_{22} = w_2$ maka diperoleh:

$$\frac{\partial f_{31}}{\partial f_{21}} = \left(\frac{\partial\left(\frac{w_1}{w_1 + w_2}\right)}{\partial w_1}\right) = \frac{w_2}{(w_1 + w_2)^2} \tag{3.36}$$

$$\frac{\partial f_{32}}{\partial f_{21}} = \left(\frac{\partial\left(\frac{w_2}{w_1 + w_2}\right)}{\partial w_1}\right) = -\frac{w_2}{(w_1 + w_2)^2} \tag{3.37}$$

$$\frac{\partial f_{31}}{\partial f_{22}} = \left(\frac{\partial\left(\frac{w_1}{w_1 + w_2}\right)}{\partial w_2}\right) = -\frac{w_1}{(w_1 + w_2)^2} \tag{3.38}$$

$$\frac{\partial f_{32}}{\partial f_{22}} = \left(\frac{\partial\left(\frac{w_2}{w_1 + w_2}\right)}{\partial w_2}\right) = \frac{w_1}{(w_1 + w_2)^2} \tag{3.39}$$

Sehingga

$$\varepsilon_{21} = \varepsilon_{31}\left(\frac{\partial f_{31}}{\partial f_{21}}\right) + \varepsilon_{32}\left(\frac{\partial f_{32}}{\partial f_{21}}\right) = (\varepsilon_{31} - \varepsilon_{32})\frac{w_2}{(w_1 + w_2)^2} \tag{3.40}$$

$$\varepsilon_{22} = \varepsilon_{31}\left(\frac{\partial f_{31}}{\partial f_{22}}\right) + \varepsilon_{32}\left(\frac{\partial f_{32}}{\partial f_{22}}\right) = (\varepsilon_{32} - \varepsilon_{31})\frac{w_1}{(w_1 + w_2)^2} \tag{3.41}$$

e) *Error* pada *layer* 1

Layer 1 memilikiempat buahneuron dengan propagasi *error* yang menuju *layer* ini dapat dirumuskan dalam persamaan berikut:

$$\varepsilon_{11} = \varepsilon_{21}\left(\frac{\partial f_{21}}{\partial f_{11}}\right) \tag{3.42}$$

$$\varepsilon_{12} = \varepsilon_{22}\left(\frac{\partial f_{22}}{\partial f_{12}}\right) \tag{3.43}$$

$$\varepsilon_{13} = \varepsilon_{21}\left(\frac{\partial f_{21}}{\partial f_{13}}\right) \tag{3.44}$$

$$\varepsilon_{14} = \varepsilon_{22}\left(\frac{\partial f_{21}}{\partial f_{14}}\right) \tag{3.45}$$

di mana

ε_{11} : *error* pada *layer* 1 dengan neuron ke-1

ε_{12} : *error* pada *layer* 1 dengan neuron ke-2

ε_{13} : *error* pada *layer* 1 dengan neuron ke-3

ε_{14} : *error* pada *layer* 1 dengan neuron ke-4

Karena pada *layer* ini $f_{21} = w_1 = \mu_{A_1}(x_1) \cdot \mu_{B_1}(x_2)$, $f_{22} = w_2 = \mu_{A_2}(x_1) \cdot \mu_{B_2}(x_2)$, $f_{11} = \mu_{A_1}(x_1)$, $f_{12} = \mu_{A_2}(x_1)$, $f_{13} = \mu_{B_1}(x_2)$ dan $f_{14} = \mu_{B_1}(x_2)$ maka diperoleh:

$$\varepsilon_{11} = \varepsilon_{21}\left(\frac{\partial\left(\mu_{A_1}(x_1) \cdot \mu_{B_1}(x_2)\right)}{\partial\left(\mu_{A_1}(x_1)\right)}\right) = \varepsilon_{21}\mu_{B_1}(x_2) \tag{3.46}$$

$$\varepsilon_{12} = \varepsilon_{22}\left(\frac{\partial\left(\mu_{A_2}(x_1) \cdot \mu_{B_2}(x_2)\right)}{\partial\left(\mu_{A_2}(x_1)\right)}\right) = \varepsilon_{22}\mu_{B_2}(x_2) \tag{3.47}$$

$$\varepsilon_{13} = \varepsilon_{21}\left(\frac{\partial\left(\mu_{A_1}(x_1) \cdot \mu_{B_1}(x_2)\right)}{\partial\left(\mu_{B_1}(x_2)\right)}\right) = \varepsilon_{21}\mu_{A_1}(x_1) \tag{3.48}$$

$$\varepsilon_{14} = \varepsilon_{22}\left(\frac{\partial\left(\mu_{A_2}(x_1) \cdot \mu_{B_2}(x_2)\right)}{\partial\left(\mu_{B_2}(x_2)\right)}\right) = \varepsilon_{22}\mu_{A_2}(x_1) \tag{3.49}$$

Error pada *layer* ini digunakan untuk memperoleh nilai *error* dari parameter premis

3.6. Model *Hybrid* SARIMAX-ANFIS

Model *hybrid* merupakan suatu metode kombinasi dari satu atau lebih model fungsi dalam satu sistem. Kombinasi model runtun waktu yang memliki struktur linier dan nonlinier secara umum dapat dituliskan pada persamaan berikut:

$$Z_t = L_t + N_t \tag{3.50}$$

di mana:

L_t : Komponen linier

N_t : Komponen nonlinier

Model SARIMAX akan digunakan untuk menyelesaikan kasus yang linier dengan residual yang masih mengandung hubungan dengan nonlinier. Secara matematis hubungan tersebut dapat dituliskan sebagai berikut:

$$e_t = Z_t - \hat{L}_t \tag{3.51}$$

di mana:

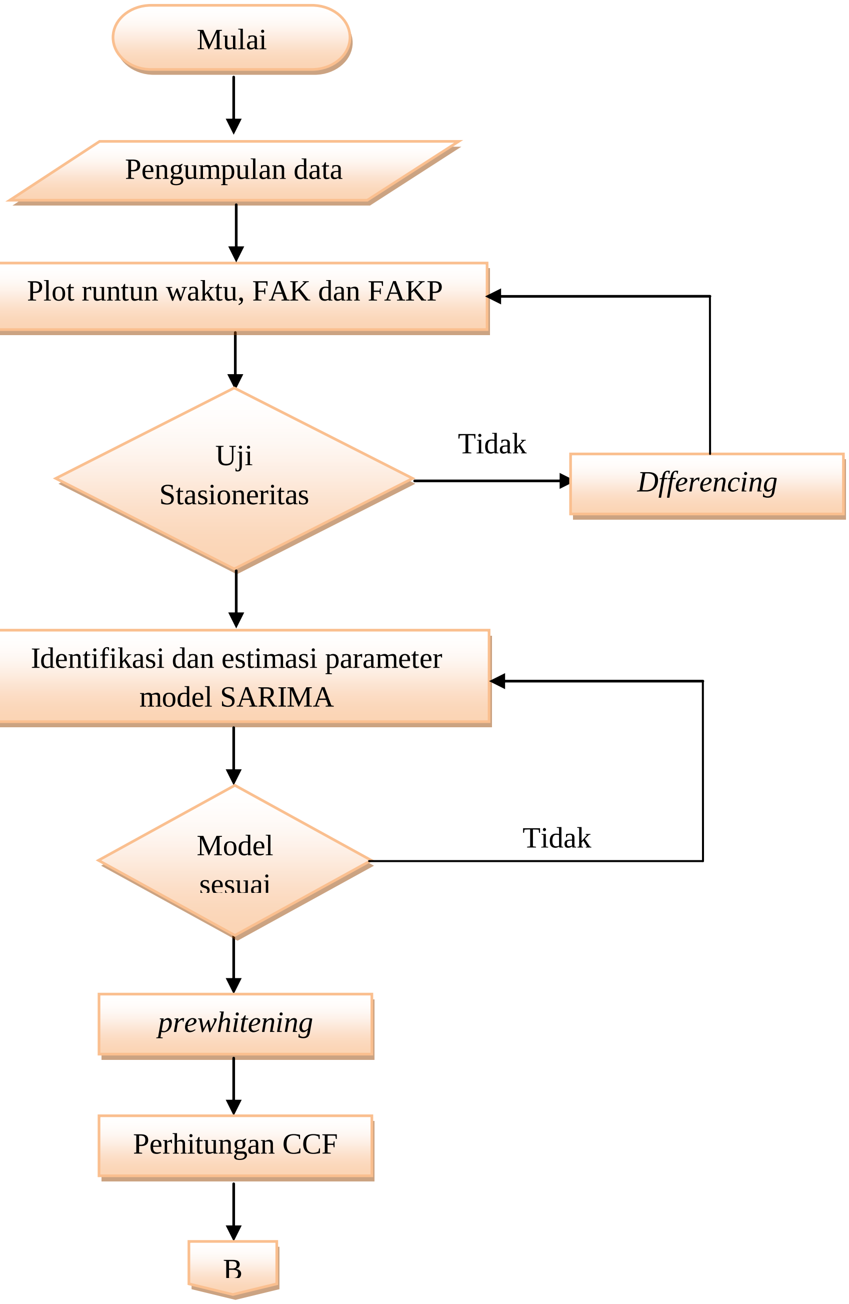
$\hat{L}_t$: nilai ramalan pada waktu ke- *t*

Z_t : nilai aktual pada waktu ke- *t*

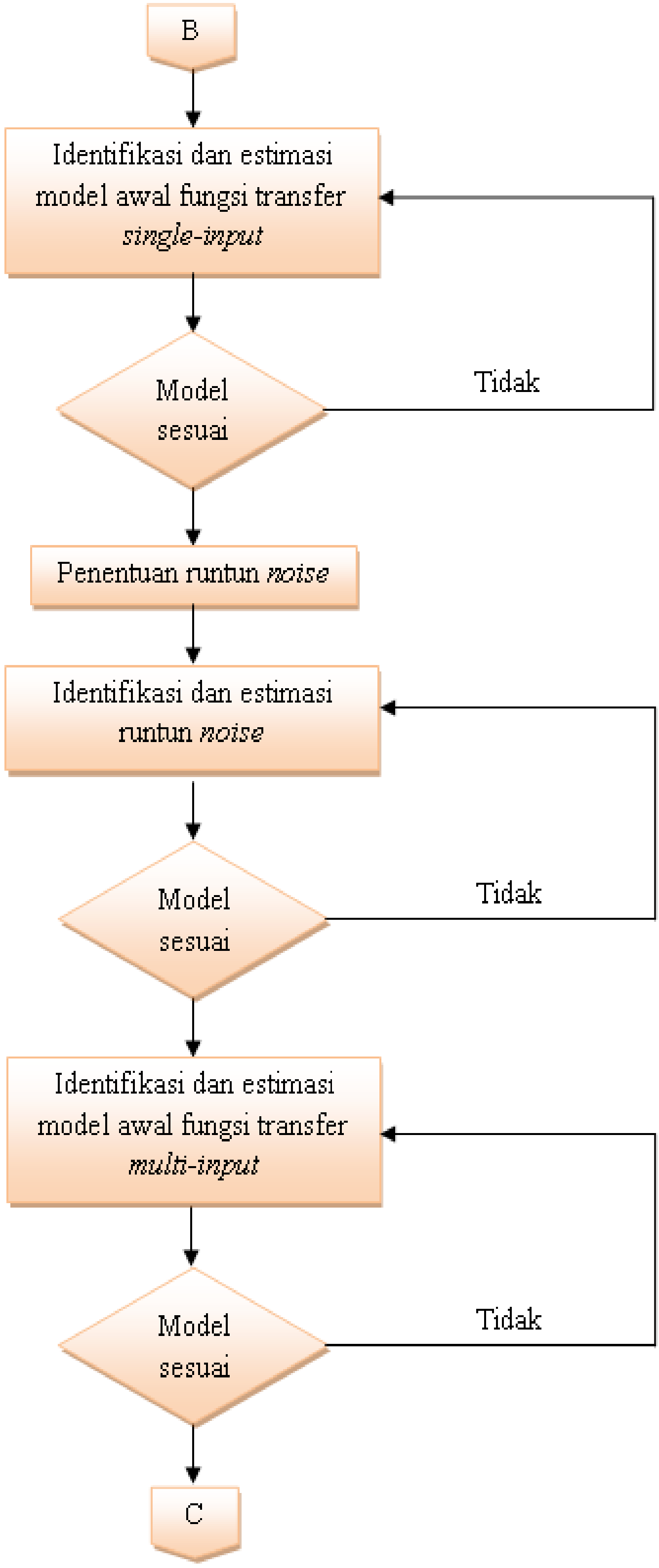
Setelah diperoleh model SARIMAX, selanjutnya memodelkan residual dari SARIMAX dengan menggunakan ANFIS. Hasil ramalan dari metode ANFIS dikombinasikan dengan hasil ramalan SARIMAX. Hasil ramalan keseluruhan secara matematis dapat dituliskan dalam persamaan berikut:

$$Z_t = \hat{L}_t + N_t \tag{3.52}$$

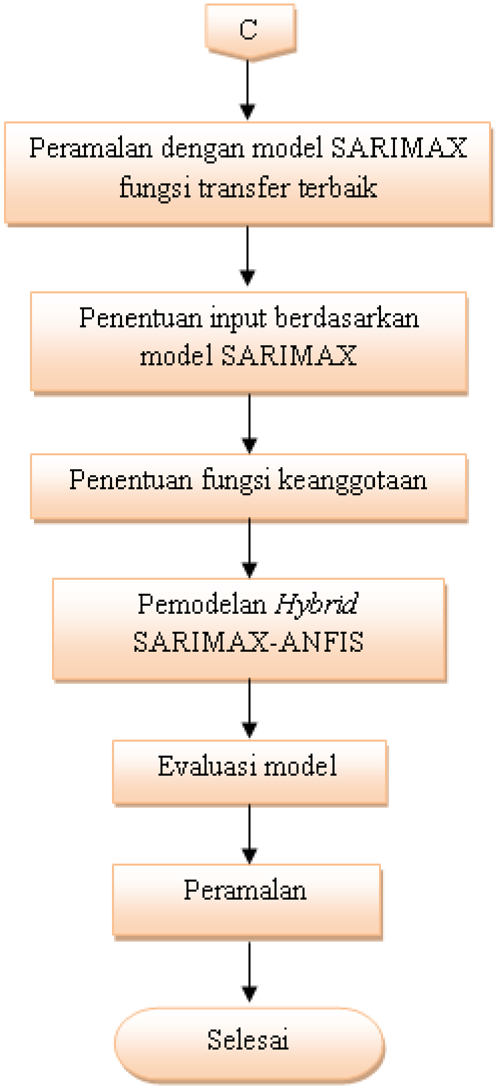
3.7. Alur Penelitian



Gambar 3.1. Flow chart alur penelitian



Gambar 3.2. Lanjutan flow chart alur penelitian



Gambar 3.3. Lanjutan flow chart alur penelitian