

BAB III

METODE PENELITIAN

3.1. Jenis Penelitian

Penelitian yang akan dilakukan termasuk kedalam golongan penelitian terapan karena merupakan kegiatan ilmiah yang mengungkapkan gejala alam maupun kehidupan yang perlu diperbaiki karena adanya kekurangan dengan metode sistematis, teratur, objektif, tertib dan dapat dipertanggung jawabkan (Nawawi & Martini, 2005). Dalam penelitian ini, data yang digunakan adalah data harga beras di Provinsi Jawa Barat dimana data yang akan dihasilkan berupa peramalan.

3.2. Sumber Data

Data yang digunakan dalam penelitian ini adalah data sekunder berupa harga beras di Jawa Barat dan data curah hujan yang diperoleh dari *website* resmi Badan Pusat Statistik (BPS) dan Badan Meteorologi, Klimatologi, dan Geofisika (BMKG).

3.3. Variabel Penelitian

Model deret waktu yang digunakan adalah model deret waktu multivariat, karena terdapat dua variabel yang digunakan untuk penelitian ini, yaitu jumlah harga beras dan curah hujan di Jawa Barat. Data tersebut merupakan data dalam waktu bulanan (Lewis, 1982)

3.4. Prosedur Pembentukan Model ARIMAX dengan Fungsi Transfer

Adapun langkah-langkah untuk melakukan prosedur pembentukan model ARIMAX dengan fungsi transfer.

3.4.1. Identifikasi Bentuk Model Fungsi Transfer

1) Mempersiapkan deret *input* dan *output*

Pada tahap ini akan dilakukan identifikasi model seperti, kestasioneran deret *input* dan *output* dimana jika data belum stasioner dalam rata-rata, maka perlu dilakukan pembedaan (*differencing*) atau transformasi deret *input* dan *output*.

2) *Prewhitening* deret *input*

Prewhitening deret *input* pada model bertujuan menghilangkan seluruh pola yang diketahui sehingga dapat membentuk model *white noise*. Pembentukan

prewhiteng mengikuti model ARIMA deret *input* (x_t) sehingga dapat didefinisikan sebagai berikut: (Wei, 2005)

$$\phi(B)x_t = \theta(B)\alpha_t \quad (3.1)$$

Dengan α_t adalah *white noise*

$$\phi(B) = 1 - \phi_1(B) - \phi_2(B)^2 - \dots - \phi_p(B)^p \quad (3.2)$$

dan

$$\theta(B) = 1 - \theta_1 B - \theta_2 B^2 - \dots - \theta_q B^q \quad (3.3)$$

Hasil dari *prewhitening* deret *input* adalah dengan menjadikan deret *input* menjadi lebih mudah diatur dengan menghilangkan pola yang serupa agar hanya tersisa deret waktu *white noise*. Berikut deret *input* hasil dari *prewhitening* :

$$\alpha_t = \frac{\phi(B)}{\theta(B)} x_t \quad (3.4)$$

Dimana:

α_t : deret *input* dari hasil *prewhitening*

$\phi(B)$: operator *autoregressive*

$\theta(B)$: operator *moving average*

x_t : deret *input*

3) *Prewhitening* deret *output*

Setelah melakukan *prewhitening* deret *input* (x_t) maka dilakukan penerapan untuk transformasi *prewhitening* deret *output* (y_t). Transformasi pada deret *output* y_t tidak harus mejadi *white noise* (Makridakis dkk., 1999) sehingga arti dari *prewhitening* deret *input* akan berbeda dengan *prewhitening* deret *output*. Berikut adalah deret y_t yang telah diputihkan disebut dengan β_t :

$$\beta_t = \frac{\phi(B)}{\theta(B)} y_t \quad (3.5)$$

Diberikan $\varepsilon_t = \frac{\phi(B)}{\theta(B)} n_t$, model fungsi transfer menjadi:

$$\beta_t = v(B)\alpha_t + \varepsilon_t \quad (3.6)$$

Bobot respons impuls untuk model fungsi transfer dapat diperoleh dari:

$$v_k = \frac{\sigma_\beta}{\sigma_\alpha} \rho_{\alpha\beta}(k) \quad (3.7)$$

Dimana:

β_t : deret *output* dari hasil *prewhitening*

$\phi(B)$: operator *autoregressive*

$\theta(B)$: operator *moving average*

y_t : deret *output*

4) Korelasi silang atau *Cross Correlation Function (CCF)* dan autokorelasi deret *input* dan deret *output* yang telah *diprewhitening*

Cross Correlation Function (CCF) atau korelasi silang adalah keakuratan hubungan antar dua variabel, yang dapat menentukan tingkat hubungan antar nilai antara α_t dan β_t . Nilai α_t pada waktu t dengan nilai β_t pada waktu $t + k$. Perhitungan korelasi silang untuk mengetahui tiga parameter kunci dalam fungsi transfer yaitu (b,r,s) yang diidentifikasi dari plot korelasi silang.

5) Identifikasi model awal fungsi transfer deret *input*

Setelah diperoleh nilai korelasi silang, maka selanjutnya dilakukan identifikasi model awal fungsi transfer deret *input*. Terdapat tiga orde utama dalam model fungsi transfer yaitu r, s , dan b yang dimana r menunjukkan orde fungsi $\delta(B)$, s menunjukkan orde fungsi $\omega(B)$ dan b menunjukkan keterlambatan sebesar b periode sebelum x mempengaruhi y yang ditulis x_{t-b}

Orde r, s , dan b ditentukan berdasarkan *lag-lag* dari perhitungan korelasi silang. Pada dasarnya nilai dari r, s , dan b tidak terbatas pada 0,1,2 saja tetapi bisa lebih dari itu. Namun jarang sekali proses perhitungan yang telah dilakukan sebelumnya menyentuh angka angka besar, terutama orde r dimana yang umum ditemukan hanya 0,1,2. Oleh karena itu, terdapat fungsi transfer yang sering digunakan dengan nilai $b = 2$, nilai r dan s antara 0,1 atau 2. Berikut tabel model fungsi transfer yang umum digunakan: (Montgomery dkk., 2015)

Tabel 3. 1 Model Fungsi Transfer yang Umum Digunakan

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

6) Identifikasi Awal Deret Gangguan (Noise)

Setelah didapat model fungsi transfer maka langkah selanjutnya adalah penaksiran awal deret *noise* dari model fungsi transfer dengan menggunakan persamaan berikut:

$$y_t = \frac{\omega(B)}{\delta(B)} x_{t-b} + n_t \quad (3.8)$$

$$n_t = y_t - \frac{\omega(B)}{\delta(B)} x_{t-b} \quad (3.9)$$

7) Identifikasi Model ARIMA (p_n, d, q_n) dari Deret Gangguan

(Makridakis dkk.,1999) menjelaskan bahwa penentuan deret *noise* (p, q) dilakukan dengan menganalisa nilai-nilai n_t dengan menggunakan ARIMA, untuk menentukan model ARIMA yang tepat. Fungsi $\phi_m(B)$ dan $\theta_n(B)$ untuk deret *noise* (gangguan) n_t dapat diperoleh persamaan sebagai berikut:

$$\phi_m(B)n_t = \theta_n(B)\alpha_t \quad (3.10)$$

8) Estimasi Parameter Model Fungsi Transfer

Setelah melakukan identifikasi model maka langkah selanjutnya adalah estimasi parameter-parameter model fungsi transfer input. Model fungsi transfer sementara diperoleh pada persamaan (2.32), terdapat parameter-parameter yang perlu dilakukan estimasi yaitu $\delta = (\delta_1, \dots, \delta_r)'$, $\omega = (\omega_1, \dots, \omega_s)'$, $\phi = (\phi_1, \dots, \phi_p)'$, $\theta = (\theta_1, \dots, \theta_q)'$, dan σ_α^2 . Pada persamaan (2.32) dapat dibentuk persamaan baru dengan mengalikan dengan $\delta(B)\phi(B)$, sehingga diperoleh persamaan sebagai berikut:

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

Atau dapat juga ditulis sebagai berikut:

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

Dimana:

$$\begin{aligned} c(B) &= \delta(B)\phi(B) = (1 - \delta_1 B - \dots - \delta_r B^r)(1 - \phi_1 B - \dots - \phi_p B^p) \\ &= (1 - c_1 B - c_2 B^2 - \dots - c_{r+p} B^{r+p}) \\ d(B) &= \phi(B)\omega(B) = (1 - \phi_1 B - \dots - \phi_p B^p)(\omega_0 - \omega_1 B - \dots - \omega_s B^s) \\ &= (d_0 - d_1 B - d_2 B^2 - \dots - d_{p+s} B^{p+s}) \\ e(B) &= \delta(B)\theta(B) = (1 - \delta_1 B - \dots - \delta_r B^r)(1 - \theta_1 B - \dots - \theta_q B^q) \\ &= (1 - e_1 B - e_2 B^2 - \dots - e_{r+q} B^{r+q}) \end{aligned}$$

Sehingga diperoleh

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

Dimana c_i, d_j , dan e_k merupakan fungsi transfer dari $\delta_i, \omega_j, \phi_k$ dan θ_l .

Lalu dengan mengasumsikan bahwa ε_t adalah deret *white noise* berdistribusi normal $N(0, \sigma_\alpha^2)$, berikut merupakan fungsi *likelihood* : (Wei, 2005)

$$L = (\delta, \omega, \phi, \theta, \sigma_\alpha^2 | b, x, y, x_0, y_0, \varepsilon_0) = (2\pi\sigma_\varepsilon^2)^{-\frac{n}{2}} \exp\left(-\frac{1}{2\sigma_\varepsilon^2} \sum_{t=1}^n \alpha_t^2\right) \quad (3.11)$$

Dimana x_0, y_0, ε_0 merupakan nilai awal untuk menghitung α_t .

9) Uji Diagnostik Model Fungsi Transfer

Setelah melakukan estimasi parameter maka langkah selanjutnya adalah melakukan uji diagnostik. Uji diagnostik bertujuan untuk melihat kecocokan model sebelum melakukan peramalan, selain itu juga dapat untuk menguji apakah asumsi bahwa ε_t merupakan *white noise* dan bebas terhadap deret *input* yang telah di *prewhitening* (α_t). Jika asumsi terpenuhi maka model fungsi transfer yang telah diuji merupakan model yang layak digunakan untuk peramalan. (Wei W., 1994)

Pengujian diagnostik dari model fungsi transfer dilakukan dengan memberikan residual dari model dan dari residual model *input* yang telah dilakukan *prewhitening*. Model fungsi transfer yang telah dipilih dapat menjadi model yang tidak layak, karena pemodelan deret *noise*, pemodelan fungsi transfer

atau keduanya salah sehingga terdapat 2 macam untuk melakukan pemeriksaan dan pengujian da;am uji diagnostik yaitu: (Abraham & Ledorter, 1983)

a. Pengujian Korelasi Silang atau *Cross Correlation Funtion* (CCF)

Korelasi silang anantara deret *input* dan deret *output* yang telah dilakukan *prewhitening* dapat memberikan gambaran mengenai tingkatan hubungan antar deret. Korelasi silang dapat diubah ke dalam estimasi bobot-bobot *respon impuls* ,dimana susunan dalam bobot *respon impuls* akan menghasilkan bentuk model fungsi transfer sementara. Dalam proses perkiraan bobot fungsi transfer terdapat asumsi bahwa deret *input* yang telah *diprewhitening* (α_t) adalah bebas dari komponen *noise* (α_t). (Makridakis dkk., 1999)

Sebuah model dapat dikatakan fungsi transfer yang layak apabila koefisien korelasi silang dari deret *noise* dengan deret *input* tidak menunjukkan suatu pola tertentu dan berada pada dua kesalahan yang standar $2(n - k)^{-\frac{1}{2}}$ (Wei , 1994). Berikut merupakan korelasi silang antara deret *noise* dengan deret *input* menggunakan uji Q:

- Hipotesis

H_0 : Korelasi silang antara deret *noise* dengan deret *input* tidak signifikansi.

H_1 : Korelasi silang antara deret *noise* dengan deret *input* signifikan.

- Statistik Uji

$$Q = m(m + 2) \sum_{i=0}^k \frac{r_{atat}^2(k)}{m - i}$$

Dimana:

m : $n - t_0 + 1$

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

n : banyaknya pengamatan

k : lag maksimum

r_{atat} : koefisien korelasi silang antara deret *at* dengan *at* pada lag k

- Kriteria

Statistik Q menyebar mengikuti sebara *chi-square* dengan derajat kebebasan $(K + 1 - r - s)$. Lalu terima H_0 jika $Q < \chi^2_{(K+1-r-s)}$ maka dapat dikatakan bahwa model fungsi transfer layak.

b. Pengujian Autokorelasi

Pengujian kelayakan suatu model dilakukan untuk mengetahui keseesuaian model yaitu dengan memenuhi syarat *white noise*. Sebuah model dapat dikatakan layak jika koefisien Fak dan FAKP dari residualnya tidak menunjukkan pola tertentu (Wei, 2005). Berikut meruoakan pengujian autokorelasi untuk residualnya:

- Hipotesis

H_0 : Autokorelasi pada deret sisa a_t tidak signifikansi.

H_1 : Autokorelasi pada deret sisa a_t signifikan.

- Statistik Uji

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

Dimana:

m : $n - t_0 + 1$

m : jumlah parameter

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

n : banyaknya pengamatan

k : lag maksimum

r_{at} : koefisien autokorelasi a_t pada lag k

- Kriteria

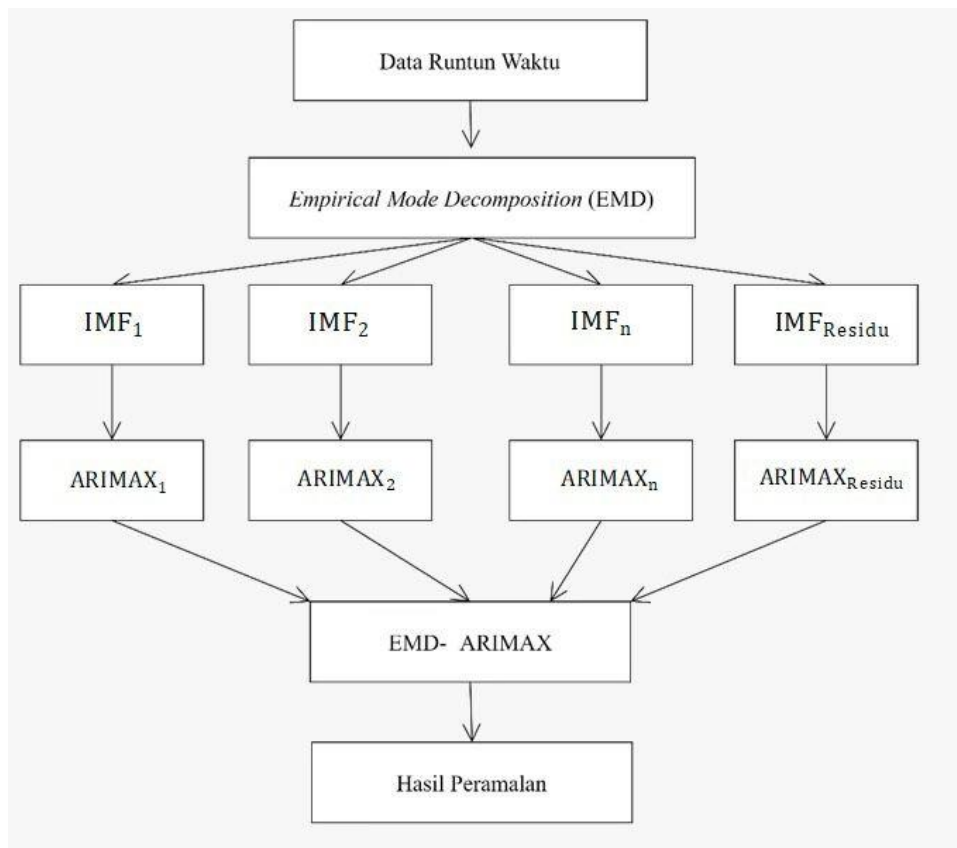
Statistik Q menyebar mengikuti sebara *chi-square* dengan derajat kebebasan $(K - p - q)$. Lalu terima H_0 jika $Q < \chi^2_{(K-p-q)}$ maka dapat dikatakan bahwa model layak.

3.5. Langkah Analisis Data

Teknik peramalan yang digunakan adalah metode *Empirical Mode Decomposition* (EMD) – *Autoregressive Integrated Moving Average With Variable Exogeneous* (ARIMAX). Langkah-langkah sebagai berikut:

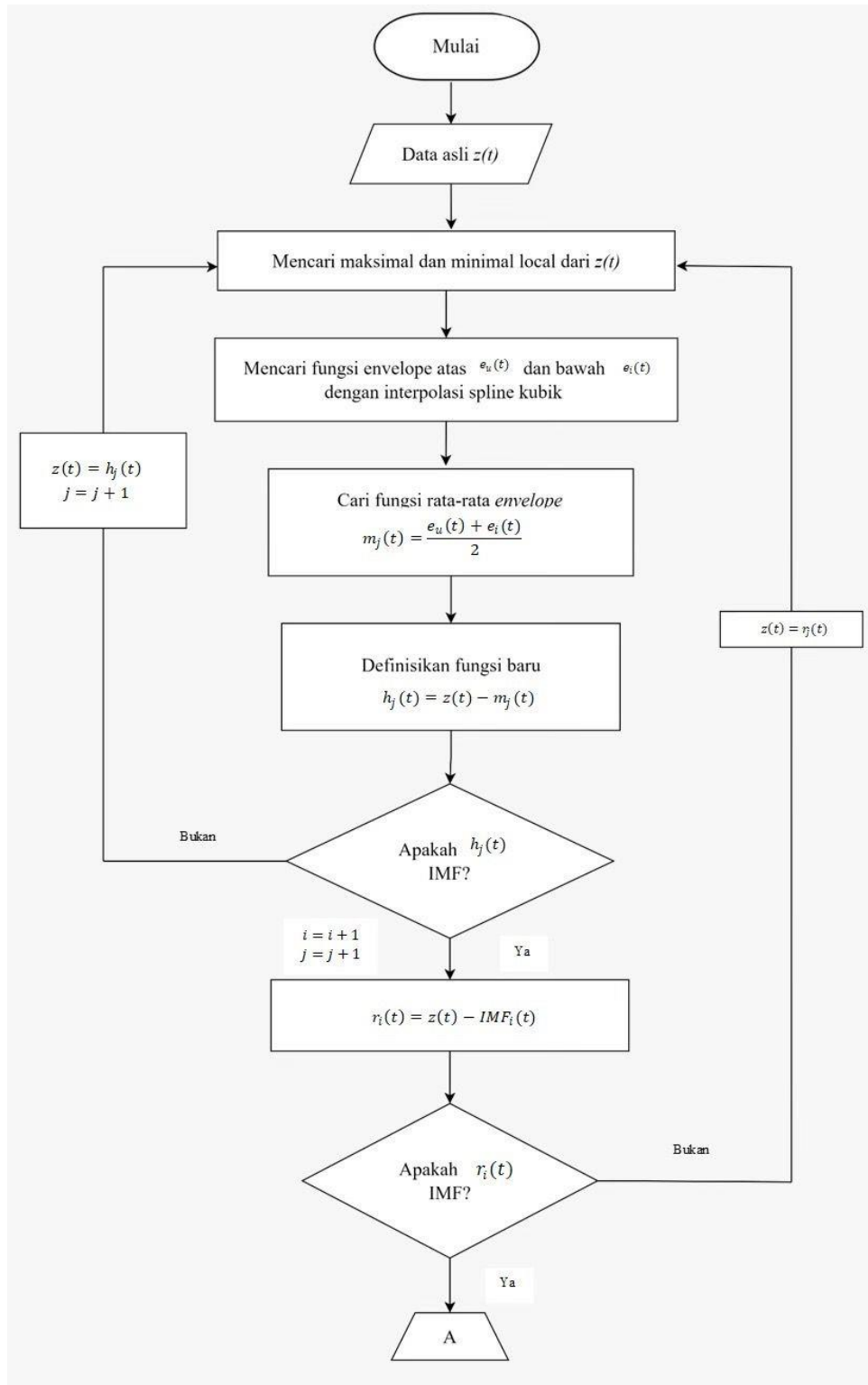
1. Mengidentifikasi nilai ekstrim lokal yaitu seluruh nilai maksimum dan minimum lokal dalam data deret waktu asli ($z(t)$) harga beras.
2. Menghubungkan seluruh nilai maksimum dengan *spline* kubik untuk mendapatkan *envelope* atas dan lakukan yang sama untuk nilai minimum untuk mendapatkan *envelope* bawah, kemudian menghitung rata-rata *envelope* ($m_j(t)$).
3. Mengekstrak rata-rata *envelope* dengan mengurangi sinyal masukkan awal $z(t)$ dengan rata-rata *envelope* sehingga akan menghasilkan data yang baru $h_j(t)$.
4. Data baru yang dihasilkan atau perbedaan nilai antara sinyal asli dengan rata-rata *envelope* merupakan IMF.
5. Mengidentifikasi nilai ekstrim hingga ekstrak rata-rata *envelope* dilakukan secara berulang hingga memenuhi kriteria henti sehingga akan didapatkan beberapa IMF dan residu.
6. Membuat plot deret waktu, fungsi autokorelasi dan fungsi autorekasi parsial dari data Curah Hujan (*input*), IMF dan residu (*output*) yang diperoleh berdasarkan hasil dari metode EMD.
7. Mengidentifikasi kestasioneran data, dan jika data dari data curah hujan, IMF dan residu belum stasioner maka harus dilakukan *differencing*.
8. Menentukan model ARIMA serta melakukan identifikasi model dan estimasi parameter dari model.
9. Melakukan *prewhitening* deret *input* dan deret *output*
10. Mengidentifikasi plot *Cross Correlation Function* (CCF) dari deret *input* dan *output* yang telah dilakukan *prewhitening*.
11. Mengidentifikasi model awal ARIMAX fungsi transfer *single-input*
12. Mengidentifikasi penaksiran awal deret *noise*.
13. Mengidentifikasi model ARIMA (p_n, d, q_n) dari deret *noise*
14. Melakukan peramalan untuk setiap data IMF dan residu lalu digabungkan dengan hasil peramalan tersebut sehingga menjadi peramalan dengan metode EMD-ARIMAX.

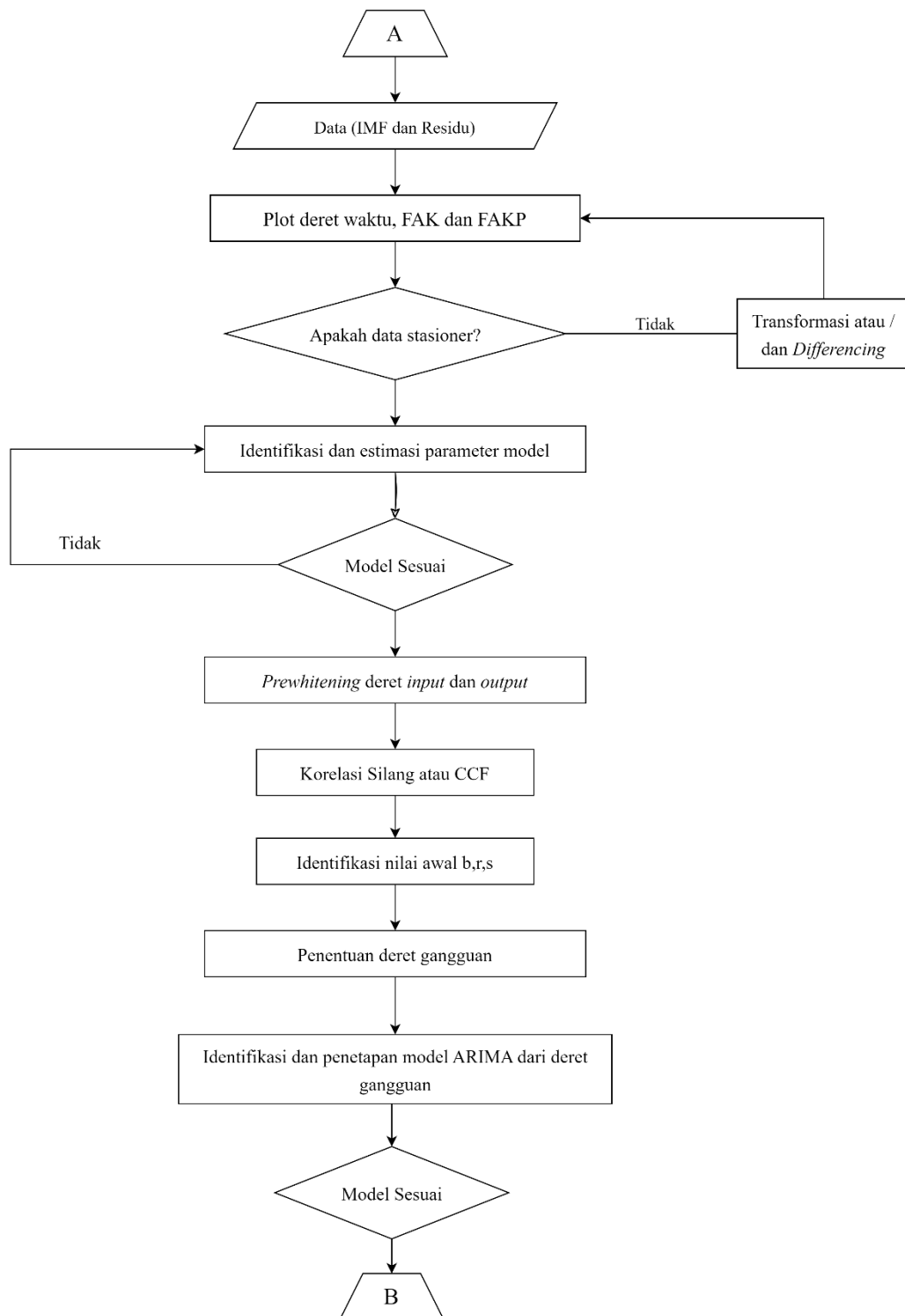
Di bawah ini merupakan bagan cara bekerja metode EMD-ARIMAX secara umum:

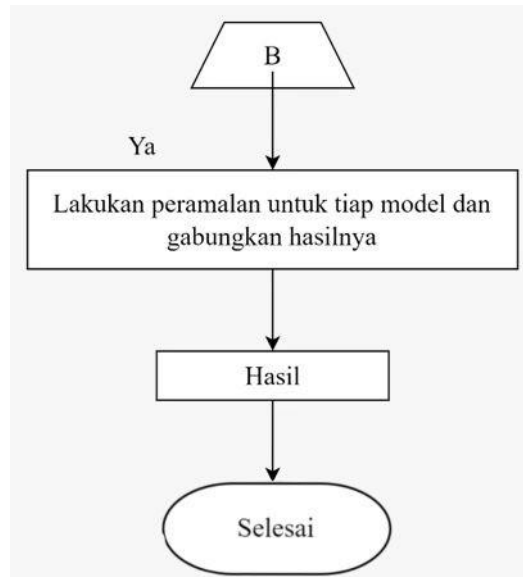


Gambar 3. 1 Kerangka berfikir

Setelah kerangka berfikir metode EMD-SARIMAX, berikut adalah *flowchart* metode EMD-SARIMAX:







Gambar 3. 2 *Flowchart EMD-SARIMAX*