BAB II

TINJAUAN PUSTAKA

2.1 Pendahuluan

Berdasarkan latar belakang yang sudah diuraikan, yaitu ingin mengetahui bagaimana meramalkan curah hujan di Kota Pontianak yang dipengaruhi oleh beberapa faktor diantaranya suhu udara, kecepatan angin dan kelembaban udara dengan menggunakan metode fungsi transfer *input* ganda. Maka pada bab ini akan dijelaskan beberapa teori yang digunakan dalam penelitian.

2.2 Pengertian Curah Hujan

Curah hujan adalah banyaknya air yang jatuh ke permukaan bumi. Derajat curah hujan dinyatakan dengan jumlah curah hujan dalam suatu satuan waktu. Biasanya satuan yang digunakan adalah mm/jam. Dalam meteorologi, butiran hujan dengan diameter lebih dari 0.5 mm disebut hujan dan diameter antara 0.5 – 0.1 mm disebut gerimis. Semakin besar ukuran butiran hujan maka semakin besar pula kecepatan jatuhnya. Ketelitian alat ukur curah hujan adalah 1/10 mm. Curah hujan dapat diartikan sebagai ketinggian air yang tekumpul dalam tempat yang datar, dengan asumsi tidak meresap, tidak mengalir dan tidak menguap ke atmosfer (Tjasyono 2004). Alat yang dipakai untuk mengukur curah hujan adalah tabung gelas ukur (*raingauge*) atau perekam (*Automatic Rain Recorder* atau Pluviometer). Pengukuran curah hujan dilakukan melalui alat yang disebut penakar curah hujan dan diukur setiap jam 7 pagi waktu setempat. Dalam skala waktu keragaman curah hujan dibagi atas tipe harian, bulanan dan tahunan. Menurut Wilson(1993:7), curah hujan disebabkan oleh beberapa faktor yaitu kelembaban udara, tekanan udara, temperatur atau suhu udara dan kecepatan angin yang dapat dicari korelasinya untuk meramalkan curah hujan.

Intensitas hujan adalah tinggi atau kedalaman air hujan persatuan waktu. Sifat umum hujan adalah makin singkat hujan berlangsung intensitasnya cenderung makin tinggi dan makin besar periode ulangnya makin tinggi pula intensitasnya.

Seperti yang kita ketahui, Kota Pontianak adalah kota yang dilewati garis khatulistiwa dimana suhu udaranya bisa dikatakan lebih panas dibandingkan dengan kota-kota lain yang ada di Indonesia. Namun BMKG Kalimantan Barat mengatakan, tidak menutup kemungkinan kota khatulistiwa ini memiliki cuaca ekstrim yang perlu diwaspadai karena dapat menyebabkan banjir. Ketika intensitas hujan tinggi dan berlangsung selama beberapa hari, maka air dari sungai Kapuas akan meluap, hal ini merupakan salah satu penyebab banjir. Intensitas curah hujan tertinggi di Kota Pontianak terjadi diantara bulan November sampai bulan Februari setiap tahunnya. Curah hujan menurut BMKG dibagi menjadi empat kelompok yaitu :

1. Curah hujan rendah : 0-20mm, 21-50mm, 51-100mm.
2. Curah hujan menengah : 101-150mm, 151-200mm, 201-300mm.
3. Curah hujan tinggi : 301-400mm.
4. Curah hujan sangat tinggi : 401-500mm.

2.3 Faktor-Faktor yang Mempengaruhi Curah Hujan

a. Suhu Udara

Suhu udara adalah keadaan panas atau dinginnya udara. Suhu juga disebut temperatur yang diukur dengan alat termometer. Beberapa faktor yang mempengaruhi suhu udara diantaranya: tinggi tempat, daratan/lautan, radiasi matahari, indeks datang matahari dan angin. Pengukuran suhu biasa dinyatakan dalam skala Celsius (C), Reamur (R), dan Fahrenheit (F). Suhu udara tertinggi di permukaan bumi adalah di daerah tropis (sekitar ekuator) dan makin ke kutub makin dingin (Soewarno,2000:12). Kota Pontianak meliliki suhu rata-rata 27.8 °C.

b. Kecepatan Angin

Kecepatan angin adalah jarak tempuh angin atau pergerakan udara per satuan waktu dan dinyatakan dalam satuan meter per detik (m/d), kilometer per jam (km/j), dan mil per jam (mi/j). Satuan kecepatan angin umumnya yang digunakan adalah 'knot' (kn) dimana 1 kn = 1,85 km/j = 1,151mi/j = 0,514 m/d atau 1 m/d = 2,237 mi/j = 1,944 kn. Kecepatan angin dapat diukur dengan alat yang disebut anemometer. Angin adalah udara yang bergerak akibat adanya perbedaan tekanan udara dengan arah aliran angin dari tempat yang memiliki tekanan tinggi ke tempat yang bertekanan rendah atau dari daerah yang memiliki suhu/temperatur rendah ke wilayah bersuhu tinggi. Angin memiliki hubungan yang erat dengan sinar matahari karena daerah yang terkena banyak paparan sinar mentari akan memiliki suhu yang lebih tinggi serta tekanan udara yang lebih rendah dari daerah lain di sekitarnya sehingga menyebabkan terjadinya aliran udara. Angin juga dapat disebabkan oleh pergerakan benda sehingga mendorong udara di sekitarnya untuk bergerak ke tempat lain (Soewarno,2000:15).

c. Kelembaban Udara

Kelembaban adalah perbandingan antara massa uap dalam suatu satuan volum dengan massa uap yang jenuh dalam satuan volum itu pada suhu yang sama. Secara umum kelembaban menyatakan banyaknya kadar air yang ada di udara. Banyaknya uap yang bergerak di dalam atmosfer berpengaruh terhadap besarnya hujan, lamanya hujan, dan intensitas curah hujan. Kelembaban tertinggi umumnya terjadi pada musim penghujan dan paling rendah pada musim kemarau. Satuan untuk kelembaban yang umum digunakan adalah RH yaitu *Relative Humadity* atau kelembaban relatif.

Tingkat kelembaban bervariasi menurut suhu. Semakin hangat suhu udara, semakin banyak uap air yang dapat ditampung. Semakin rendah suhu udara, semakin sedikit jumlah uap air yang dapat ditampung. Jadi pada siang hari yang panas dapat menjadi lebih lembab dibandingkan dengan hari yang dingin. Umumnya semakin tinggi suatu daerah dari permukaan laut maka kelembaban udaranya semakin tinggi. Makin tinggi kelembaban udara akan dapat menyebabkan bertambah banyak uap air yang dapat diserap awan.

2.4 Peramalan

Ramalan merupakan *input* bagi proses perencanaan dan pengambilan keputusan. Peramalan adalah memperkirakan sesuatu pada masa yang akan datang berdasarkan data masa lampau yang dianalisis secara ilmiah. Sehingga perencanaan menggunakan ramalan digunakan unutk membantu para pengambil keputusan dalam memilih alternatif terbaik untuk menemukan suatu kebijakan (Yanti, 2009).

Ramalan dapat diperoleh dengan bermacam-macam cara yang dikenal dengan metode peramalan. Secara ilmiah metode peramalan dapat diklasifikasikan menjadi dua kelompok yaitu metode kuantitatif dan metode kualitatif. Metode peramalan kuantitafif diterapkan apabila terdapat tiga kondisi diantaranya tersedianya informasi tentang masa lalu dan informasi tersebut dapat berupa bentuk data numerik serta dapat diasumsikan data masa lalu tersebut akan terus berlanjut di masa mendatang. Sedangkan metode kualitatif tidak memerlukan data yang serupa seperti metode peramalan kuantitatif karena data yang dibutuhkan tergantung pada metode tertentu dan biasanya merupakan hasil dari pemikiran intuitif, perkiraan, dan pengetahuan yang telah didapat oleh si pengguna. Peramalan dengan metode kualitatif ini akan sangat berguna jika data kuantitatif yang akurat sulit diperoleh (Yanti, 2009). Dalam hal ini ramalan dikatakan baik/tidak bergantung dari banyak hal antara lain pengalaman, perkiraan dan pengetahuan yang didapat.

2.5 Analisis Deret Waktu (*Time Series*)

Data berkala (*time series)* adalah data yang disusun berdasarkan urutan waktu atau data yang dikumpulkan dari waktu ke waktu. Data berkala berhubungan dengan data statistik yang dicatat dan diselidiki dalam batas-batas (interval) waktu tertentu. Data berkala dapat dijadikan sebagai dasar untuk membuat keputusan saat ini, peramalaan pada masa yang akan datang dan perencanaan kegiatan untuk masa depan. Analisis data berkala adalah analisis yang menerangkan dan mengukur berbagai perubahan atau perkembangan data selama satu periode.

2.6 *Autoregressive Integrated Moving Average* (ARIMA)

Metode ARIMA pertama kali diperkenalkan oleh Box dan Jenkins pada tahun 1970. Oleh karena itu, pemodelan ARIMA juga dikenal dengan metode Box dan Jenkins. Model ARIMA merupakan model umum regresi *time series.* Penetapan karakteristik data *time series* seperti stasioner, musiman dan sebagainya memerlukan suatu pendekatan yang sistematis dan ini akan membantu mendapatkan gambaran yang jelas mengenai model-model dasar yang akan ditangani. Secara umum, model ARIMA ditulis dengan ARIMA (p, d, q) yang artinya model ARIMA dengan derajat AR (p), derajat pembedaan (d), dan derajat MA (q). sebelum membahas lebih rinci mengenai ARIMA, terlebih dahulu mengembangkan dua ide kunci dalam analisis deret waktu yaitu kestasioneran data dan fungsi autokorelasi (Pankratz, 1991).

1. Kestasioneran Data

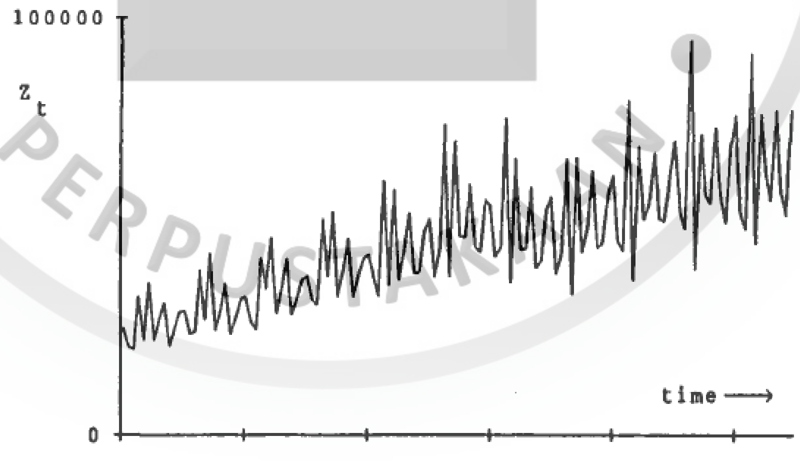
Data *time series* dikatakan stasioner jika memiliki rata-rata dalam varians yang konstan untuk semua t. Standar analisis ARIMA memiliki asumsi sederhana yaitu bahwa proses menghasilkan satu *time series* yang bersifat stasioner. Menurut (Wei, 2006), proses stasioner untuk satu (*Zt*), mempunyai rata-rata *E(Zt)* = μ, dan Var(Zt)= $E(Z_t - \mu)^2 = \sigma^2$, yang keduanya konstan. Kovarian dari *Zt* dan Z_{t+k} dapat ditulis pada persamaan (2.1).

$$\mathrm{Cov}(Z_t, Z_{t+k}) = E[((Zt) - \mu)(Z_{t+k} - \mu)] = \gamma_k \tag{2.1}$$

Jika data tidak stasioner dalam rata-rata maka dapat diatasi dengan melakukan *differencing*, sedangkan jika data tidak stasioner dalam varians maka dapat diatasi dengan melakukan transformasi. Untuk tahap menstasionerkan data, langkah pertama yang dilakukan adalah stasioner dalam varians untuk mencapai varians yang stasioner kemudian dilakukan stasioner dalam rata-rata untuk mencapai data yang stasioner (Pankratz, 1991).

a. Stasioner dalam Varians

Data dapat dikatakan tidak stasioner dalam varians ketika tinggi rendahnya kelompok data yang satu dengan yang lainnya berbeda jauh atau ketika plot data menunjukkan pola melebar atau pola terompet seperti pada Gambar 2.1



Gambar 2.1 Plot Data Tidak Stasioner dalam Varians

Untuk mengatasi ketidakstasioneran terhadap varians dilakukan transformasi dengan melihat plot Box-Cox. Jika *Lower* CL dan *Upper* CL mencakup 1 maka data sudah stasioner. Transformasi Box-Cox adalah transformasi pangkat pada respon yaitu λ yang dipangkatkan dengan Z_t sehingga transformasinya menjadi Z^{λ}, dengan λ merupakan parameter transformasi. Jika diperoleh nilai $\lambda = 1$ maka tidak ada transformasi yang artinya data telah stasioner dalam varians. Persamaan dari transformasi ini adalah : (Wei, 2006)

$$T(Y_t) = \frac{Z_t{}^{\lambda} - 1}{\lambda} \tag{2.2}$$

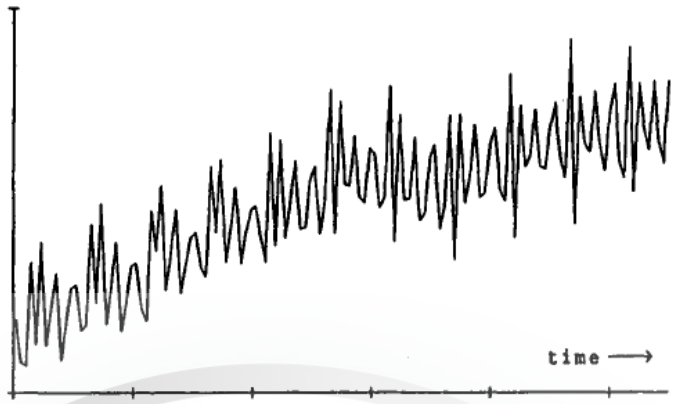
Transformasi varians ini dilakukan apabila nilai data postif, jika beberapa nilai data negatif dapat tambahkan sebuah konstanta positif untuk data dan harus dilakukan sebelum analisis seperti *differencing*. Jika transformasi kuasa ini dihubungkan dengan bentuk transformasi stabilitas varians yang lain, maka diperoleh tabel kesetaraan seperti pada Tabel 2.1.

Tabel 2.1 Hubungan Nilai λ dengan Transformasi Stabilitas Varians

λ	**Transformasi**
-1	$\frac{1}{Z_t}$
-0,5	$\frac{1}{\sqrt{Z_t}}$
0	$\mathrm{Ln}(Z_t)$
0,5	$\sqrt{Z_t}$
1	Z_t (tidak ada transformasi)

b. Stasioner dalam Rata-rata

Data dikatakan tidak stasioner dalam rata-rata ketika data bergerak terus naik ke atas yang membentuk *trend* atau ketika data tidak sejajar dengan sumbu waktu seperti pada Gambar 2.2.



Gambar 2.2 Plot Data Tidak Stasioner dalam Rata-rata

Untuk melihat apakah data stasioner dalam rata-rata dengan melihat plot ACF dan PACF. Ada beberapa cara untuk menguji stasioneritas yaitu dengan cara melihat grafik *time series* data dan korelogram, akan tetapi keduanya memiliki kelemahan dalam unsur subjektivitas. Sehingga dapat digunakan uji formal yaitu dengan *Augmented Dickey Fuller* (ADF) untuk mengetahui stasioneritas data, hipotesisnya:

$H_0 : \delta = 0$ (data tidak stasioner)

$H_1 : \delta < 0$ (data stasioner)

Statistik Uji :

$$\tau = \frac{\hat{\delta}}{SE(\hat{\delta})} \tag{2.3}$$

Kriteria uji: Tolak H_0 jika *p-value* < α, artinya data telah stasioner.

Untuk mengatasi ketidakstasioner terhadap rata-rata juga dapat dilakukan *differencing*. Proses *differencing* dilakukan dengan mengurangkan data dengan data sebelumnya dengan persamaan $Z_t' = Z_t - Z_{t-1}$. Dalam proses *differencing* terdapat notasi B yang merupakan operator *shift* mundur (*backward shift*). Penggunaan notasi mempunyai pengaruh menggeser data 1 periode sebelumnya, dengan prosesnya yaitu :

$$BZ_t = Z_{t-1} \quad (2.4)$$

Jika dua notasi B diterapkan maka $BZ_t = B^2 Z_t = Z_{t-2}$. Dengan demikian untuk *differencing* pertama, persamaannya :

$$Z_t' = Z_t - BZ_t = (1-B)Z_t \quad (2.5)$$

Dan untuk *differencing* kedua, persamaannya :

$$\begin{aligned} Z_t'' &= Z'_t - Z'_{t-1} \\ &= (Z_t - Z_{t-1}) - (Z_{t-1} - Z_{t-2}) \\ &= Z_t - 2BZ_t + B^2 Z_t \\ &= (1 - 2B + B^2)Z_t \\ &= (1-B)^2 Z_t \end{aligned} \quad (2.6)$$

Sehingga *differencing* untuk orde ke *d* adalah :

$$Z_t^d = (1-B)^d Z_t \quad (2.7)$$

2. Fungsi Autokorelasi dan Autokorelasi Parsial

Fungsi autokorelasi digunakan untuk menjelaskan berapa besar korelasi *time series* dengan *time series* itu sendiri. Fungsi autokorelasi parsial pada lag *k* digunakan untuk menghitung korelasi antara *Zt* dan Z_{t+1} pada variabel-variabel diantara $Z_{t+1}, Z_{t+2} + \cdots + Z_{t+1}$ (Wei, 2006).

a. Fungsi Autokorelasi (*Autocorelation Function* / ACF)

Fungsi autokorelasi merupakan korelasi atau hubungan antar data pengamatan suatu data deret waktu. Suatu proses stasioner baik dalam rata-rata maupun varians, (Wei, 2006) menyatakan bahwa kovarians pada *Zt* dan Z_{t+k} ada pada persamaan (2.1) dan autokorelasi dari Zt dan Z_{t+1} adalah :

$$\rho_k = \frac{Cov(Z_t, Z_{t+1})}{\sqrt{Var(Z_t)}\sqrt{Var(Z_{t+1})}} = \frac{\gamma_k}{\gamma_0} \quad (2.8)$$

dimana :

Var(Z_t) = Var(Z_{t+k}) = γ_0.

Dalam analisi *time series*, γ_k disebut fungsi autokovarians dan ρ_k menunjukkan kovarians dan autokorelasi (ACF) karena γ_k dan ρ_k menunjukkan kovarians dan autokorelasi antara *Zt* dan Z_{t+k}. Menurut (Wei, 2006) fungsi autokovarian γ_k dan fungsi autokorelasi ρ_k mempunyai sifat-sifat sebagai berikut :

$\gamma_0 = Var(Z_t)$ dan $\rho_0 = 1$

$|\gamma_0| \leq \gamma_0$ dan $|\rho_k| \leq 1$

$\gamma_k = \gamma_{-k}$ dan $\rho_k = \rho_{-k}$, untuk semua nilai *k*.

Rumus yang dapat digunakan untuk menghitung ACF pada suatu data adalah (Yanti, 2009) :

$$\hat{\rho}_k = \frac{\hat{\gamma}_k}{\hat{\gamma}_0} = \frac{\frac{\sum(Z_t - \bar{Z})}{T}}{\frac{\sum(Z_t - \bar{Z})^2}{T}} \quad (2.9)$$

dimana :

γ_k = nilai kovarians sampel dengan lag *k*

$\hat{\gamma}_k$ = nilai kovarians smapel dengan *k* = 0

T = banyaknya pengamatan

b. Fungsi Autokorelasi Parsial (PACF)

Koefisien autokorelasi parsial digunakan untuk mengukur derajat hubungan antara nilai sekarang dengan nilai sebelumnya dengan pengaruh variabel *time lag* yang lain dianggap konstan. Sedangkan *Partial Autocorrelation* digunakan untuk mengukur tingkat keeratan hubungan antar pengamatan suatu *time series* yaitu Z_t dan Z_{t+1} dengan mengabaikan ketidak-bebasan $Z_{t+1}, \ldots, Z_{t+k-1}$, sehingga Z_t dianggap konstanta. Korelasi antara Z_t dan Z_{t+1} dapat digambarkan dalam persamaan (2.10) (Wei, 2006) :

$$Corr(Z_t, Z_{t+k} \mid Z_{t+1}, \dots, Z_{t+k-1}) \tag{2.10}$$

Sama halnya dengan autokorelasi antara Z_t dan Z_{t+k}, autokorelasi parsial antara $(Z_t - \hat{Z}_t)$ dan $\left(Z_{t+k} - \hat{Z}_{t+k}\right)$, jika ditulis dalam persamaan menjadi :

$$\hat{\rho}_{kk} = \frac{Cov\left[\left(Z_t - \hat{Z}_t\right), \left(Z_{t+k} - \hat{Z}_{t+k}\right)\right]}{\sqrt{Var((Z_t - \hat{Z}_t)}\sqrt{Var}(Z_{t+k} - \hat{Z}_{t+k})} \tag{2.11}$$

maka autokorelasi parsial populasi dapat dihitung dengan rumus :

$$\hat{\rho}_{kk} = \frac{\sum_{t=1}^{n-k}\left(Z_t - \hat{Z}_t\right)\left(Z_{t+k} - \hat{Z}_{t+k}\right)}{\sqrt{\sum_{t-1}^{n}\left(Z_t - \hat{Z}_t\right)\left(Z_{t+k} - \hat{Z}_{t+k}\right)}} \tag{2.12}$$

2.6.1 Model ARIMA

Secara umum model ARIMA ditulis dengan ARIMA (p, d, q) yang artinya model ARIMA dengan derajat AR (p), derajat pembedaan d, dan derajat MA (q).

1. Model *Autoregressive* (AR)

Model Autoregressive (AR) merupakan suatu model persamaan regresi yang menghubungkan nilai-nilai sebelumnya dari suatu variabel tak bebas (*dependent*) dengan variabel itu sendiri. Model *Autoregressive* (AR) dengan orde *p* dinotasikan dengan AR*(p)*. Bentuk umum model *AR(p)* adalah :

$$Z_t = \mu + \phi_1 Z_{t-1} + \phi_2 Z_{t-2} + \cdots + \phi_p Z_{t-p} + a_t \tag{2.13}$$

dengan :

Z_t : variabel *dependent* pada waktu ke-*t*

$Z_{t-1}, Z_{t-2}, \dots, Z_{t-p}$: variabel *independent* yang merupakan lag dari Z_t

$\phi_1, \phi_2, \dots, \phi_p$: parameter model *Autoregressive* (*AR)*

a_t : nilai residual (kesalahan) pada waktu ke-*t*

p : orde AR

a. Model *Autoregressive* orde 1 atau AR(1)

Model *Autoregressive* orde 1 atau *AR*(1) secara matematis didefinisikan sebagai :

$$Z_t = \mu + \phi_1 Z_{t-1} + a_t \tag{2.14}$$

dengan *random error* $a_t \sim N(0, \sigma^2{}_a)$ dan model memenuhi asumsi stasioner.

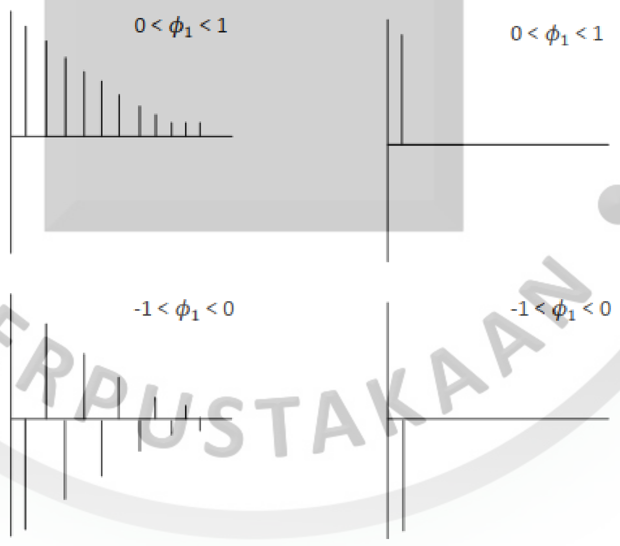
Persamaan (2.14) dapat ditulis dengan operator *backshift* (B) dari persamaan (2.4) menjadi :

$$Z_t = \mu + \phi Z_{t-1} + a_t$$

$$Z_t - \phi Z_{t-1} = \mu + a_t$$

$$(1 - \phi B) Z_t = \mu + a_t \tag{2.15}$$

Contoh gambar pola grafik ACF dan PACF yang menggambarkan model *Autoregressive* orde 1 atau AR(1) ditunjukkan pada Gambar (2.3) (Wei,1990: 35).



(a) (b)

Gambar 2.3 Grafik ACF (a) dan PACF (b) pada model AR (1)

Gambar 2.3 menunjukkan pola ACF dan PACF model AR (1). Terlihat pada gambar bahwa ACF menurun mendekati nol dan plot PACF signifikan pada lag pertama.

Maka persamaan umum untuk Autogresif (AR) orde p atau ARIMA (p,0,0) :

$$Z_t = \phi_1 Z_{t-1} + \phi_2 Z_{t-2} + \cdots + a_t \quad (2.16)$$

2. Model *Moving Average* (MA)

Model *Moving Average* (MA) merupakan suatu model persamaan regresi yang menghubungkan nilai residual sebelumnya dari suatu variabel *dependent* (tak bebas) dengan residual. Model *Moving Average* (MA) dengan orde *q* dinotasikan dengan MA (*q*). bentuk umum model MA(*q*) adalah :

$$Z_t = \mu + a_t - \theta_1\, a_{t-1} - \theta_2\, Z_{t-2} - \cdots - \theta_q\, Z_{t-q} \quad (2.17)$$

a. Model *Moving Average* orde 1 atau MA (1)

Model *Moving Average* orde 1 atau *MA*(1) secara matematis didefinisikan seperti persamaan (2.18).

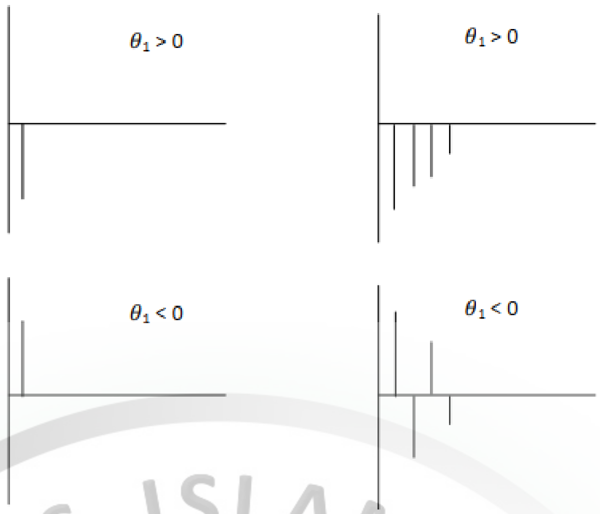
$$Z_t = \mu + a_t - \theta_1\, a_{t-1} \quad (2.18)$$

Persamaan (2.18) dapat ditulis dengan operator *backshift* (B), dari persamaan (2.4) menjadi :

$$Z_t = \mu + a_t - \theta_1\, a_{t-1}$$

$$Z_t = \mu + (1 - \theta_1 B) a_t \quad (2.19)$$

Grafik ACF dan PACF yang menggambarkan model *Moving Average* orde 1 atau MA(1) (Wei, 2006: 49) adalah



(a) (b)

Gambar 2.4 Grafik ACF (a) dan PACF (b) Model MA(1)

Gambar 2.4 menunjukkan bahwa pola plot ACF dan PACF pada model MA(1), yaitu plot ACF signifikan pada di *lag* pertama dan PACF menurun mendekati nol. Dengan persamaan umum AR dan MA, maka persamaan umum model ARMA (p, 0, q) adalah :

$$Z_t = \mu + \phi_1 Z_{t-1} + \mu + \phi_2 Z_{t-2} + \cdots + a_t - \theta_1 a_{t-1} - \theta_2 a_{t-2} - \theta_q a_{t-q} \qquad (2.20)$$

jika d adalah bilangan bulat non negatif, maka (Z_t) dikatakan proses ARIMA dengan $(1-B)^d Z_t$ merupakan akibat dari proses ARMA. Definisi berikut digunakan dalam membangun model ARIMA :

$\phi_p\,(B) = (1 - \phi_1 B - \phi_2 B^2 - \cdots - \phi_p B^p)$ (operator orde p)

$\theta_p\,(B) = (1 - \theta_1 B - \theta_2 B^2 - \cdots - \theta_p B^p)$ (operator orde q)

Berdasarkan model sebelumnya, maka model ARIMA (p, d, q) adalah :

$$\phi(B)(1-B)^d Z_t = \mu + \theta(B) a_t$$

$$Z_t = \frac{\mu + \theta(B)}{\phi(B)(1-B)^d} a_t \qquad (2.21)$$

dengan :

p,d,q : orde AR, *differencing* dan MA (non-musiman),

X_t : deret waktu stasioner,

a_t : residual.

Orde (p,d,q) didapatkan dari *lag* yang signifikan pada plot *Autocorelation Function* (ACF) dan *Partial Autocorelation Function* (PACF) secara grafis mengikuti semua ketentuan seperti pada Tabel 2.2 (Yanti, 2009) :

Tabel 2.2 Ketentuan AR dan MA

Tipe Model	Pola Umum dari ACF	Pola Umum dari PACF
AR (*p*)	Eksopensial menurun atau seperti pola gelombang sinus atau membentuk keduanya	Menurun drastis pada *lag* p
MA (*q*)	Menurun drastis pada *lag* p	Menurun secara eksponensial atau seperti pola gelombang sinus atau dapat membentuk keduanya
ARMA (*p*, *q*)	Menurun secara eksponensial	Menurun secara eksponensial

2.6.2 *Seasonal Autoregressive Intregrated Moving Average* (SARIMA)

Deret berkala musiman yaitu deret berkala yang mempunyai sifat berulang setelah beberapa periode waktu tertentu, misalnya satu tahun, satu bulan, triwulanan dan seterusnya. Notasi untuk musiman dalam ARIMA atau SARIMA adalah (Chatfield, 2004:66) :

$$(p,d,q)\ (P,D,Q)^S \qquad (2.22)$$

dimana :

(p, d, q) : model ARIMA yang tidak mengandung unsur musiman,

(P, D, Q) : model ARIMA yang mengandung unsur musiman, S : jumlah periode musiman

μ : konstanta

Maka, model untuk SARIMA adalah :

$$\Phi_P(B^s)\phi_p(B)(1-B^D)(1-B)^d Z_t = \mu + \Theta_Q(B^s)\theta_q(B)a_t$$

$$Z_t = \frac{\mu + \Theta_Q(B^s)\theta_q(B)a_t}{\Phi_P(B^s)\phi_p(B)(1-B^S)(1-B)^d} \quad (2.23)$$

2.6.3 Langkah-Langkah Pemodelan ARIMA

1. Identifikasi Model

Identifikasi model berkaitan dengan penentuan orde pada ARIMA. Identifikasi model bisa dilakukan dengan membuat plot data *time series* untuk mengetahui adanya pola suatu data, kemudian menganalisis koefisien autokorelasi dan koefisien autokorelasi parsial. Koefisien autokorelasi (ACF) merupakan korelasi deret berkala terhadap deret berkala itu sendiri dengan selisih waktu (*lag*) 0, 1, 2 atau lebih. Persamaan koefisien autokorelasi bisa dilihat pada persamaan (2.9).

Sedangkan autokorelasi parsial (PACF) merupakan suatu fungsi yang menunjukkan besarnya hubungan antara pengamatan pada waktu ke-t dengan pengamatan pada waktu-waktu sebelumnya (t_{-1}, t_{-2}, …, t_{-k}). Persamaan autokorelasi parsial dapat dilihat pada persamaan (2.12).

2. Estimasi dan Pengujian Parameter Model

Setelah berhasil mendapatkan identifikasi model sementara, selanjutnya parameter-parameter AR dan MA baik untuk musiman maupun tidak musiman harus ditetapkan dengan cara yang terbaik. Ada 2 cara yang mendasar untuk mendapatkan parameter-parameter tersebut, yaitu (Makridakis, 1999:407) :

1. Dengan cara mencoba-coba, yaitu menguji beberapa nilai yang berbeda dan memilih satu nilai tersebut (atau sekumpulan nilai, apabila terdapat lebih dari

satu parameter yang akan ditaksir) yang meminimumkan jumlah kuadrat nilai sisa.

2. Perbaikan secara iteratif (berulang) dengan memilih taksiran awal dan kemudian membiarkan program komputer memperhalus penaksiran tersebut secara interaktif.

Setelah dilakukan estimasi parameter maka parameter tersebut perlu diuji signifikansinya untuk mengetahui apakah parameter tersebut dapat dimasukkan dalam model dengan uji hipotesis sebagai berikut :

- AR(*Autoregressive*)

$H_0: \phi_i = 0$, dimana i=1,2,..,p (AR tidak signifikan dalam model)

$H_0: \phi_i \neq 0$, dimana i=1,2,..,p (AR signifikan dalam model)

- MA(*Moving Average*)

$H_0: \theta_i = 0$, dimana i=1,2,..,q (MA tidak signifikan dalam model)

$H_0: \theta_i \neq 0$, dimana i=1,2,..,q (MA signifikan dalam model)

Statistik uji :

- Proses AR - Proses AR (musiman)

$$AR = \frac{\hat{\phi}_i}{SE(\hat{\phi}_i)} \qquad AR = \frac{\hat{\Phi}_i}{SE(\hat{\Phi}_i)} \tag{2.24}$$

- Proses MA - Proses MA (musiman)

$$MA = \frac{\hat{\theta}_i}{SE(\hat{\theta}_i)} \qquad MA = \frac{\hat{\Theta}_i}{SE(\hat{\Theta}_i)} \tag{2.25}$$

dengan $\hat{\phi}_i$ adalah estimator dari ϕ_i, $\hat{\theta}_i$ estimator dari θ_i, $\hat{\Phi}$ merupakan estimator musiman dari Φ, dan $\hat{\Theta}$ merupakan estimator musiman dari Θ, sedangkan $SE(\hat{\phi}_i)$ adalah *standard error* yang diestimasi dari ϕ_i.

Kriteria keputusan yang digunakan untuk menolak H_0 adalah jika $|t| > t_{\frac{\alpha}{n},df}$, df = n-p dengan p banyaknya parameter dan n banyaknya pengamatan atau H_0 ditolak jika *p-value* < α.

3. Diagnosis Model

Setelah menentukan nilai – nilai parameter dari model ARIMA sementara, selanjutnya perlu dilakukan pemeriksaan diagnostik pada model ARIMA sementara untuk membuktikan bahwa model sementara yang telah ditetapkan sudah cocok.

Pemeriksaan diagnosis dilakukan dengan analisis residual. Analisis residual yaitu melakukan pemeriksaan terhadap nilai residual (a_t) yang dihasilkan dari tahap estimasi parameter, jika (a_t) adalah suatu proses *white noise* (gerakan random) maka model sudah cocok. Pada proses *white noise*, *ACF* dan *PACF* menunjuk ke nol. Untuk mendeteksi bahwa suatu proses (a_t) *white noise*, pada analisis residual dilakukan uji independensi residual yaitu autokorelasi residual.

Analisis yang dapat digunakan untuk analisis autokorelasi residu ini adalah Ljung-Box *Statistic* untuk menguji *white noise* residual. Yang dimaksud *white noise* residual yaitu tidak ada pola apapun dalam deret residu. Dengan menentukan apakah gugus autokorelasi secara signifikan berbeda dengan nol, dengan hipotesisnya adalah:

H_0 : Model sudah cocok

H_1 : Model tidak cocok

Statistik uji :

$$Q = T(T+2)\sum_{k=1}^{k}\frac{{r_k}^2}{T-k} \qquad (2.26)$$

dimana :

T : banyaknya pengamatan asli,

r_k : autokorelasi untuk lag ke k.

Kriteria pengujian nya tolak H_0 jika $Q > \lambda^2{}_{(\alpha;k-p-q)}$, dimana nilai p dan q adalah order dari ARMA(p,q). Atau dapat dilihat dari nilai *p-value* > 0.05, yang artinya residual tidak berautokorelasi atau *white noise*.

4. Kriteria Pemilihan

Jika terdapat dua atau lebih model deret waktu dilakukan pemilihan kriteria model, tujuannya untuk memilih model yang layak digunakan dalam peramalan. Salah satu pendekatan yang dapat digunakan adalah AIC (*Akaike's Information Criterion*). Pada pemilihan model terbaik menggunakan AIC adalah model yang memiliki nilai AIC yang minimum. Rumus untuk memperoleh nilai AIC ditulis pada persamaan (2.27) (Wei,2006) :

$$AIC = ln(\hat{\sigma}^2) + \frac{2}{n}r \quad (2.27)$$

dimana

n : banyaknya pengamatan residual,

$\hat{\sigma}^2$: jumlah kuadrat residual dibagi banyak pengamatan,

r : banyaknya parameter pada model ARIMA.

2.7 Fungsi Transfer

Fungsi transfer merupakan gabungan dari karakteristik analisis regresi berganda dan karakteristik deret bekala ARIMA (*Autoregressive Intergrated Moving Average*). Fungsi transfer menggambarkan suatu deret berkala *output*, disebut Y_t yang dipengaruhi oleh deret berkala *input*, disebut X_t dan *input-input* lain yang digabungkan dalam satu kelompok yang disebut "gangguan" *noise*, disebut N_t. Tujuan pemodelan fungsi transfer adalah untuk membuat model sederhana yang menghubungkan

Y_t dengan X_t dan N_t (Makridakis, 1999). Berdasarkan jumlah variabel *input*, terdapat dua jenis model fungsi transfer, yaitu fungsi transfer *single input* dan *input* ganda.

b. *Single Input*

Bentuk umum dari model fungsi transfer *single input* yaitu (Makridakis, 1999) :

$$Y_t = v(B)X_t + N_t \tag{2.28}$$

dengan :

Y_t = deret *output*,

Xt = deret *input*,

N_t = pengaruh kombinasi dari seluruh faktor yang mempengaruhi Y_t atau disebut juga deret '*noise*'.

$v(B) = (v_0 + v_1 B + v_2 B^2 + \cdots + v_k B^k)$, dimana k merupakan orde fungsi transfer.

Untuk mendapatkan model fungsi transfer deret *input* dan *output* perlu ditransformasikan dan didiferensiasi untuk menghilangkan ketidakstasioneran serta proses musiman mungkin perlu dihilangkan terlebih dahulu agar bentuk fungsi transfer lebih sederhana. Jika ketiga hal tersebut sudah dilakukan, maka parameter dari persamaan fungsi transfer (2.28) akan berubah menjadi menjadi huruf kecil seperti persamaan berikut :

$$y_t = v(B)x_t + n_t \tag{2.29}$$

y_t, x_t dan n_t adalah bentuk Y_t, X_t dan N_t yang telah stasioner. Orde dari fungsi transfer adalah *k*, biasanya nilai *k* bisa sangat besar dan tak terbatas yang dapat membuat model menjadi kurang akurat. Oleh karena itu, model fungsi transfer pada persamaan (2.29) bisa diwakili dengan parameter yang lebih sedikit. Fungsi $v(B)$ dapat ditulis seperti persamaan (2.30) (Makridakis, 1999) :

$$v(B) = \frac{\omega(B)}{\delta(B)} \tag{2.30}$$

Dari persamaan (2.30) dapat dilihat $v(B)$ diwakilkan dengan 2 parameter yaitu $\omega(B)$ dan $\delta(B)$ maka persamaan (2.28) dapat ditulis :

$$y_t = \frac{\omega(B)}{\delta(B)} x_t + n_t \quad (2.31)$$

dimana :

$\omega(B) = \omega_0 - \omega_1 B - \omega_2 B^2 - \cdots - \omega_S B^s$,

$\delta(B) = 1 - \delta_1 B - \delta_2 B^2 - \cdots - \delta_r B^r$,

$\omega_s(B)$: operator *Moving Average* (MA) orde s,

$\delta_r(B)$: operator *autoregressive* (AR) orde r.

Persamaan (2.31) menunjukkan bentuk yang lebih sederhana dibandingkan dengan bentuk (2.28).

Sebagai contoh misalkan $\omega(B) = (1{,}2 - 0{,}5B)$ dan $\delta(B) = (1 - 0{,}8B)$. Maka sisi sebelah kanan dijabarkan menjadi :

$$\frac{\omega(B)}{\delta(B)} = \frac{(1{,}2 - 0{,}5B)}{(1, -0{,}8B)}$$

$$= (1{,}2 - 0{,}5B)(1 - 0{,}8B)^{-1}$$

$$= (1{,}2 - 0{,}5B)(1 + 0{,}8B + 0{,}8^2 B^2 + 0{,}8^3 B^3 + \ldots)$$

$$= 1{,}2 + 0{,}46B + 0{,}368B^2 + 0{,}249B^3 + 0{,}236B^4 + \ldots$$

Parameter n_t atau disebut juga *white noise* (n_t) yang telah distasionerkan juga dapat ditulis menjadi bentuk yang lebih sederhana :

$$n_t = \frac{\theta(B)}{\phi(B)} a_t \quad (2.32)$$

dengan :

a_t adalah gangguan *random*.

$\phi(B) = 1 - \phi_1 B - \phi_2 B^2 - \cdots - \phi_p B^p$,

$\theta(B) = 1 - \theta_1 B - \theta_2 B^2 - \cdots - \theta_q B^q$,

$\phi(B)$: operator *autoregressive* orde p,

$\theta(B)$: operator *moving average* orde q.

Persamaan (2.30) jika dijabarkan bisa sangat panjang dan tak terbatas, maka model fungsi transfer dapat disederhanakan menjadi :

$$y_t = \frac{\omega(B)}{\delta(B)} x_{t-b} + \frac{\theta(B)}{\phi(B)} a_t \tag{2.33}$$

Orde-orde penting dalam fungsi transfer ini adalah (b,s,r) dan (*p* dan *q*). Orde (b,s,r) menghubungkan antara nilai y_t dan x_t, sedangkan (*p*,*q*) menunjukkan pembentukan orde dari *white noise* (n_t). Orde *r* menunjukkan banyaknya *y* sebelumnya yang mempengaruhi y_t, orde *s* menunjukkan banyaknya *x* sebelumnya yang mempengaruhi y_t, dan orde *b* menunjukkan banyaknya *lag* pada *y* yang mempengaruhi y_t. Orde *p* dan *q* adalah orde dari parameter model ARIMA.

Sebagai contoh jika model sudah diidentifikasi dan seluruh parameter sudah ditaksir (estimasi), maka persamaan (2.33) dapat ditentukan dengan cara mengalikan masing-masing ruas dengan $\frac{1-\phi(B)}{1-\phi(B)}$ untuk ruas kiri dan $\frac{1-\delta(B)}{1-\delta(B)}$ untuk ruas kanan. Misalkan untuk *r* =1, *s* = 1, *p* = 1 dan *q* = 1 kita dapatkan :

$$y_t = \left(\frac{\omega_0 - \omega_1 B}{1-\delta_1 B}\right)\left(\frac{1-\phi_1 B}{1-\phi_1 B}\right) x_{t-b} + \left(\frac{1-\theta_1 B}{1-\phi_1 B}\right)\left(\frac{1-\delta_1 B}{1-\delta_1 B}\right) a_t$$

$$y_t = \frac{(\omega_0 - \omega_1 B)(1-\phi_1 B) x_{t-b} + \ (1-\theta_1 B)(1-\delta_1 B) a_t}{(1-\delta_1 B)(1-\phi_1 B)}$$

$$(1-\delta_1 B)(1-\phi_1 B) y_t = (\omega_0 - \omega_1 B)(1-\phi_1 B) x_{t-b} + \ (1-\theta_1 B)(1-\delta_1 B) a_t$$

$$(1 - \ \delta_1 B - \ \phi_1 B + \ \delta_1 \ \phi_1 B^2) y_t = (\omega_0 - \omega_1 B - \omega_0 \phi_1 B + \omega_1 \phi_1 B^2) x_{t-b}$$

$$+ (1 - \theta_1 B - \delta_1 + \theta_1 \delta_1 B^2) a_t$$

$$(1 - (\delta_1 + \phi_1) B + \ \delta_1 \ \phi_1 B^2) y_t = (\omega_0 - (\omega_1 + \omega_0 \phi_1) B + \omega_1 \phi_1 B^2) x_{t-b}$$

$$+ (1 - (\theta_1 + \delta_1) B + \theta_1 \delta_1 B^2) a_t$$

$$y_t - (\delta_1 + \phi_1) y_{t-1} + (\delta_1 \ \phi_1) y_{t-2} = (\omega_0 x_{t-b} - (\omega_1 + \omega_0 \phi_1) x_{t-b-1} + (\omega_1 \phi_1) x_{t-b-2})$$

$$+ a_t - (\theta_1 + \delta_1) a_{t-1} + (\theta_1 \delta_1) a_{t-2})$$

$$y_t = (\delta_1 + \phi_1)y_{t-1} - (\delta_1\,\phi_1)y_{t-2} + \omega_0 x_{t-b} - (\omega_1 + \omega_0\phi_1)x_{t-b-1} + (\omega_1\phi_1)x_{t-b-2}$$
$$+a_t - (\theta_1 + \delta_1)a_{t-1} + (\theta_1\delta_1)a_{t-2} \quad (2.34)$$

Dengan mengetahui nilai parameter dan nilai sebelumnya dari $x, y,$ dan a maka persamaan diatas dapat digunakan untuk menetapkan nilai y untuk periode-periode yang akan datang.

c. *Input* Ganda

Prosedur pembentukan model fungsi transfer *input* ganda melalui dua tahap, yaitu pembentukan model fungsi transfer tunggal dari masing-masing *input*, kemudian dilakukan pembentukan model fungsi transfer dari semua variabel. Fungsi transfer *input* ganda merupakan hasil penjumlahan fungsi transfer *single input* dari masing-masing *input*. Bentuk umum fungsi transfer dengan *input* ganda ditulis pada persamaan (2.35), (2.36), dan (2.37) (Wei, 2006) :

$$y_t = \sum_{j=1}^{k} v_j(B)x_{jt} + n_t \quad (2.35)$$

$$y_t = \sum_{j=1}^{k} \frac{\omega_j(B)}{\delta_j(B)} x_{j,t} + \frac{\theta(B)}{\phi(B)} a_t \quad (2.36)$$

$$y_t = \sum_{j=1}^{k} \left[\delta_j(B)\right]^{-1} \omega_j(B) x_{jt-b_j} + [\phi(B)]^{-1}\theta(B)a_t \quad (2.37)$$

Misalkan persamaan (2.29) memiliki 2 variabel *input* ($x_{1,t}$ dan $x_{2,t}$) dan p = 1, q = 1. Maka akan diperoleh persamaan seperti berikut :

$$y_t = v_1(B)x_{1,t} + v_1(B)x_{2,t} + n_t$$

$$y_t = \frac{\omega_1(B)}{\delta_1(B)} x_{1,t-b_1} + \frac{\omega_2(B)}{\delta_2(B)} x_{2,t-b_2} + \frac{\theta_1(B)}{\phi_1(B)} a_t$$

$$y_t = \left(\frac{\omega_0 - \omega_1(B)}{1-\delta_1(B)}\right)\left(\frac{1-\phi_1(B)}{1-\phi_1(B)}\right)x_{1t-b} + \left(\frac{\omega_0 - \omega_1 B - \omega_2 B^2}{1-\delta_2(B)}\right)\left(\frac{1-\phi_1(B)}{1-\phi_1(B)}\right)x_{2t-b}$$
$$+\left(\frac{1-\theta_1(B)}{1-\phi_1(B)}\right)\left(\frac{1-\delta_1(B)}{1-\delta_1(B)}\right)\left(\frac{1-\delta_2(B)}{1-\delta_2(B)}\right)a_t$$

$$y_t = (\omega_0 - \omega_1 B)(\,1-\phi_1 B)x_{1t-b} + (\omega_0 - \omega_1 B - \omega_2 B^2)(\,1-\phi_1 B)x_{2t-b}$$
$$+\,(\,1-\theta_1 B)(1-\delta_1 B)(1-\delta_2 B)\,a_t$$

$$y_t = \frac{(\omega_0 - \omega_1 B)(1 - \phi_1 B)x_{1t-b} + (\omega_0 - \omega_1 B - \omega_2 B^2)(1 - \phi_1 B)x_{2t-b} + (1 - \theta_1 B)(1 - \delta_1 B)(1 - \delta_2 B)\, a_t}{(1 - \delta_1 B)(1 - \delta_2 B)(1 - \phi_1 B)}$$

$$(1 - \delta_1 B)(1 - \delta_2 B)(1 - \phi_1 B)y_t = (\omega_0 - \omega_1 B)(1 - \phi_1 B)x_{1t-b} + (\omega_0 - \omega_1 B - \omega_2 B^2)$$
$$(1 - \phi_1 B)x_{2t-b} + (1 - \theta_1 B)(1 - \delta_1 B)(1 - \delta_2 B)\, a_t$$

$$(1 - \delta_1 B - \delta_2 B + \delta_1 \delta_2 B^2)(1 - \phi_1 B)y_t = (\omega_0 - \omega_1 B - \omega_0 \phi_1 B + \omega_1 \phi_1 B^2)x_{1t-b}$$
$$+(\omega_0 - \omega_1 B - \omega_2 B^2 - \omega_0 \phi_1 B + \omega_1 \phi_1 B^2 + \omega_2 \phi_1 B^3)\, x_{2t-b}$$
$$+(1 - \theta_1 B)(1 - \delta_1 B - \delta_2 B + \delta_1 \delta_2 B^2)a_t$$

$$(1 - \delta_1 B - \delta_2 B + \delta_1 \delta_2 B^2 - \phi_1 B + \delta_1 \phi_1 B^2 + \delta_2 \phi_1 B^2 + \delta_1 \delta_2 \phi_1 B^3)y_t = (\omega_0 - \omega_1 B - \omega_0 \phi_1 B + \omega_1 \phi_1 B^2)x_{1t-b} + (\omega_0 - \omega_1 B - \omega_2 B^2 - \omega_0 \phi_1 B + \omega_1 \phi_1 B^2 + \omega_2 \phi_1 B^3)\, x_{2t-b} + (1 - \delta_1 B - \delta_2 B + \delta_1 \delta_2 B - \theta_1 B - \delta_1 \theta_1 B^2 + \delta_2 \theta_1 B^2 + \delta_1 \delta_2 \theta_1 B^3)\, a_t$$

$$y_t - (\delta_1)y_{t-1} - (\delta_2)y_{t-1} + (\delta_1 \delta_2)y_{t-2} - \phi_1 y_{t-1} + \delta_1 \phi_1 y_{t-2} + \delta_2 \phi_1 y_{t-2} + \delta_1 \delta_2 \phi_1 y_{t-3})$$
$$= (\omega_0)x_{1t-b} - (\omega_1)x_{1t-b-1} - (\omega_0 \phi_1)x_{1t-b-1} + (\omega_1 \phi_1)x_{1t-b-2}$$
$$+ (\omega_0)x_{2t-b} - (\omega_1)x_{2t-b-1} - (\omega_2)x_{2t-b-2} - (\omega_0 \phi_1)x_{2t-b-1} + (\omega_1 \phi_1)x_{2t-b-2}$$
$$+ (\omega_2 \phi_1)x_{2t-b-3} + a_t - \delta_1 a_{t-1} - \delta_2 a_{t-1} + (\delta_1 \delta_2)a_{t-1} - \theta_1 a_{t-1} - (\delta_1 \theta_1)a_{t-2}$$
$$+ (\delta_2 \theta_1)a_{t-2} + (\delta_1 \delta_2 \theta_1)a_{t-3}$$

$$y_t = (\delta_1)y_{t-1} + (\delta_2)y_{t-1} - (\delta_1 \delta_2)y_{t-2} + \phi_1 y_{t-1} - (\delta_1 \phi_1)y_{t-2} - (\delta_2 \phi_1)y_{t-2} - (\delta_1 \delta_2 \phi_1)y_{t-3}) + (\omega_0)x_{1t-b} - (\omega_1)x_{1t-b-1} - (\omega_0 \phi_1)x_{1t-b-1} + (\omega_1 \phi_1)x_{1t-b-2} + (\omega_0)x_{2t-b} - (\omega_1)x_{2t-b-1} - (\omega_2)x_{2t-b-2} - (\omega_0 \phi_1)x_{2t-b-1} + (\omega_1 \phi_1)x_{2t-b-2} + (\omega_2 \phi_1)x_{2t-b-3} + a_t - \delta_1 a_{t-1} - \delta_2 a_{t-1} + (\delta_1 \delta_2)a_{t-1} - \theta_1 a_{t-1} - (\delta_1 \theta_1)a_{t-2} + (\delta_2 \theta_1)a_{t-2} + (\delta_1 \delta_2 \theta_1)a_{t-3} \qquad (2.38)$$

Bentuk Persamaan (2.37) sama dengan bentuk persamaan (2.35). Untuk sembarang *k*, dapat diperoleh dengan cara yang sama.

2.8 Tahap Pembentukan Model Fungsi Transfer

Berikut pemaparan mengenai langkah-langkah atau prosedur dalam analisis data dengan menggunakan fungsi transfer *input* ganda.

2.8.1 Tahap Pertama : Identifikasi Bentuk Model

a. Mempersiapkan Deret *Input* dan *Output*

Langkah pertama yaitu mempersiapkan deret *input* (x_t) dan *output* (y_t). Pada tahap ini yang perlu dilakukan adalah mengidentifikasi kestasioneran deret *input* dan *output.* Untuk menghilangkan ketidakstasioneran data maka perlu mentransformasi atau melakukan *differencing* deret-deret *input* dan *output* yang ada pada subbab (2.6). Setelah data telah stasioner, maka dapat dilakukan pemodelan ARIMA yang ada subbab (2.6.3).

b. Pemutihan Deret *Input* (x_t)

Mengubah deret *input* (x_t) menjadi deret *output* (y_t) dan meyederhanakannya akan membantu mempermudah memahami sistem dari fungsi transfer. Untuk melakukan pemutihan terhadap deret *input* dapat dilakukan dengan menggunakan model ARIMA. Oleh karena itu, sebelum proses pemutihan, dibangun terlebih dahulu model ARIMA bagi x_t. Untuk mendapatkan model ARIMA dapat dilihat pada subbab (2.6.3). Misalkan jika deret *input* (x_t) dimodelkan sebagai proses ARIMA *(*$p_x, 0, q_x$*)*, maka deret ini memliki model :

$$\phi_x(B)x_t = \theta_x(B)\alpha_t \tag{2.39}$$

Dengan $\phi_x(B)$ adalah operator autoregresif, $\theta_x(B)$ adalah operator rata-rata bergerak (*moving average operator*) (MA), dan α_t adalah kesalahan acak, yaitu *white noise* (dalam hal ini tidak memerlukan pembedaan (d_x) dalam model ARIMA, karena hal ini telah dilakukan pada saat mempersiapkan deret *input* dan *output*). Langkah-langkah untuk mendapatkan model ARIMA dapat dilihat pada subbab (2.6.3). Dengan demikian, deret *input* yang telah mengalami pemutihan (α_t) memiliki persamaan :

$$\alpha_t = \frac{\phi_x(B)}{\theta_x(B)}x_t \tag{2.40}$$

Persamaan (2.39) dapat ditulis dalam bentuk lain, misalkan untuk *p*=1 dan *q*=1 maka:

$\phi_x(B) = (1 - \phi_{1x}B)$

$\theta_x(B) = (1 - \theta_{1x}B)$

sehingga persamaan (2.40) menjadi :

$$\alpha_t = \frac{(1 - \phi_{1x}B)}{(1 - \theta_{1x}B)} x_t$$

$$(1 - \theta_{1x}B)\alpha_t = (1 - \phi_{1x}B)x_t$$

$$(\alpha_t - \theta_{1x}\alpha_{t-1}) = (x_t - \phi_{1x}x_{t-b})$$

$$\alpha_t = x_t - \phi_{1x}x_{t-b} + \theta_{1x}\alpha_{t-1}$$

dengan :

α_t : deret *input* yang diputihkan,

x_t : deret *input* yang telah stasioner.

b. Pemutihan Deret *Output* (y_t)

Apabila transformasi pada *prewhitening* diterapkan pada deret *input* (x_t) sebagaimana persamaan diatas, maka transformasi yang sama juga harus dilakukan untuk deret *output* (y_t). Dilakukan pemutihan deret output tujuannya agar mencari hubungan antara deret *output* dengan deret *input*.

Deret yang telah diputihkan diberi simbol β_t, dengan formulasi deret β_t memiliki persamaan :

$$\beta_t = \frac{\phi_x(B)}{\theta_x(B)} y_t \qquad (2.41)$$

Persamaan diatas ϕ_x dan θ_x diperoleh dari bentuk (2.39) dengan mengganti α_t menjadi β_t. Sama halnya seperti pemutihan deret *input*, jika memisalkan *p*=1 dan *q*=1, sehingga bentuk (2.41) menjadi :

$$\frac{(1-\phi_{1x}B)}{(1-\theta_{1x}B)} y_t = \beta_t$$

$$(1 - \phi_{1x}B)y_t = (1 - \theta_{1x}B)\,\beta_t$$

$$y_t - \phi_{1x}y_{t-1} = \beta_t - \theta_{1x}\beta_{t-1}$$

$\beta_t = y_t - \phi_{1x} y_{t-1} + \theta_{1x}\beta_{t-1}$

a. Perhitungan Korelasi Silang (CCF) dan Autokorelasi Deret *Input* dan *Output* yang Telah Diputihkan

Fungsi korelasi silang adalah ukuran kekuatan hubungan antar dua variabel. Korelasi silang antara X dan Y menentukan tingkat hubungan antar nilai X pada waktu t dengan nilai y pada waktu $t+k$ (Makridakis,1999:456). Di dalam memodelkan ARIMA univariat koefisien korelasi merupakan statistik kunci untuk menetapkan bentuk model. Korelasi silang berbeda dengan korelasi biasa karena berhubungan dengan dua deret yaitu deret *x* dan *y* yang telah diputihkan (α_t dan β_t), dapat dilihat pada persamaan (2.40) dan (2.41). Koefisien korelasi silang dari *input* (x_t) dan *output* (y_t) untuk *lag* ke-*k* didefinisikan pada persamaan (2.42) :

$$\rho_{\alpha\beta}(k) \widehat{=} r_{\alpha\beta} = \frac{S_{\alpha\beta}}{S_\alpha S_\beta} \tag{2.42}$$

dimana :

$$S_{\alpha\beta} = \frac{1}{n}\sum_{t=1}^{n-k}(X_t - \bar{X})(Y_{t+k} - \bar{Y})$$

$$S_\alpha^2 = \frac{1}{n}\sum_{t=1}^{n}(X_t - \bar{X})^2 \Rightarrow S_\alpha = \sqrt{S_\alpha^2}$$

$$S_\beta^2 = \frac{1}{n}\sum_{t=1}^{n}(Y_t - \bar{Y})^2 \Rightarrow S_\beta = \sqrt{S_\beta^2}$$

dengan $r_{\alpha\beta}(k)$ adalah korelasi silang antara α dan β pada *lag* ke-*k*.

Dalam model fungsi transfer *input* ganda perhitungan korelasi silang pada masing-masing *input* (x_t) terhadap *output* (y_t) digunakan untuk mengetahui nilai b, s, r yang diidentifikasikan dari plot korelasi silang. Untuk mengetahui nilai b, s, r dapat dilihat pada subbab bagian f. Setelah didapatkan nilai b, s, r pada masing-masing *input* kemudian dilakukan korelasi silang secara serentak antara nilai *output* (y) terhadap seluruh variabel *inputnya*.

b. Penaksiran Bobot Respon *Impuls*

Langkah berikutnya setelah perhitungan korelasi silang adalah penaksiran bobot respon *impuls* yang berguna untuk menghitung deret *noise* (n_t). Bentuk deret *noise* dapat dilihat pada persamaan (2.32). Untuk memperoleh penaksiran bobot respon *impuls* dapat dihitung seperti persamaan (2.43) :

$$v_k = \frac{r_{\alpha\beta}(k)s_\beta}{s_\alpha} \tag{2.43}$$

dengan :

v_k = bobot respon *impuls* ke-*k*

$r_{\alpha\beta}(k)$ = korelasi silang antara α dan β pada *lag* ke-k

s_β = simpangan baku (*standard deviation*) dari deret β

s_α = simpangan baku (*standard deviation*) dari deret α

c. Penetapan (b, s, r) Model Fungsi Transfer

Untuk menentukan b, s, r dapat dilihat berdasarkan pola fungsi korelasi silang. Berikut merupakan aturan yang digunakan unutk menduga b, s, r dari suatu fungsi transfer (Wei, 2006) :

a. Nilai *b* menyatakan banyaknya *lag* yang mempengaruhi y_t. Besarnya *b* dapat ditentukan dari *lag* yang pertama kali signifikan pada plot korelasi silang.

b. Nilai *s* menyatakan berapa lama deret *output* secara terus menerus dipengaruhi oleh $x_{t-b-1}, x_{t-b-2}, \dots, x_{t-b-s}$ sehingga dapat dikatakan bahwa nilai *s* adalah bilangan pada *lag* plot korelasi silang sebelum terjadinya pola menurun.

c. Nilai *r* menyatakan banyaknya y_t sebelumnya (y_{t-1}, y_{t-2}, …, y_{t-r}) yang dipengaruhi oleh X_t. Terdapat tiga kondisi pada nilai *r* yang mempunyai indikasi pemodelan berbeda, yaitu:

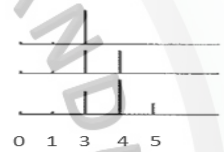
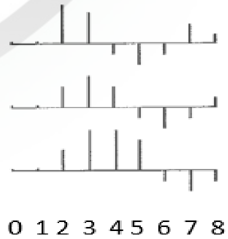
   r = 0, bila ada beberapa *lag* pada plot korelasi silang yang terpotong

r = 1, bila pada plot korelasi silang menunjukkan suatu pola eksponensial menurun.

r = 2, bila plot pada korelasi silang menunjukkan suatu pola eksponensial menurun dan mengikuti pola sinus.

Dalam praktiknya nilai s dan r tidak pernah melebihi 2, sehingga beberapa bentuk fungsi transfer yang umum digunakan dalam peramalan adalah sebagai berikut (Wei, 2006) :

Tabel 2.3 Pemodelan Fungsi Transfer

r, s, b	**Fungsi Transfer**	**Grafik**
(0, 0, 2)	$v(B)x_t = \omega_0 x_{t-2}$	
(0, 1, 2)	$v(B)x_t = (\omega_0 - \omega_1 B)x_{t-2}$	
(0, 2, 2)	$v(B)x_t = (\omega_0 - \omega_1 B - \omega_0 B^2)x_{t-2}$	
(1, 0, 2)	$v(B)x_t = \dfrac{\omega_0}{(1-\delta_1 B)} x_{t-2}$	
(1, 1, 2)	$v(B)x_t = \dfrac{(\omega_0 - \omega_1 B)}{(1-\delta_1 B)} x_{t-2}$	
(1, 2, 2)	$v(B)x_t = \dfrac{(\omega_0 - \omega_1 B - \omega_2 B^2)}{(1-\delta_1 B)} x_{t-2}$	
(2, 0, 2)	$v(B)x_t = \dfrac{\omega_0}{(1-\delta_1 B - \delta_2 B^2)} x_{t-2}$	
(2, 1, 2)	$v(B)x_t = \dfrac{(\omega_0 - \omega_1 B)}{(1-\delta_1 B - \delta_2 B^2)} x_{t-2}$	
(2, 2, 2)	$v(B)x_t = \dfrac{(\omega_0 - \omega_1 B - \omega_2 B^2)}{(1-\delta_1 B - \delta_2 B^2)} x_{t-2}$	

Pada tabel 2.3, dapat dilihat orde b dikatakan b=2 dikarenakan *lag* yang pertama dan kedua memiliki keterlambatan atau *lag* menurun. Kemudian orde r dikatakan r=1 karena plot nya membentuk pola eksponensial dan jika orde r=2 dikarenakan plot membentuk pola eksponensial menurun mengikutipola sinus. Orde *s* menunjukkan *s*=0 dikarenakan nilai pada *lag* sebelumnya lebih tinggi.

d. Penaksiran Deret *Noise* (n_t)

Setelah mendapatkan model fungsi transfer berdasarkan nilai b,s,r maka dilanjutkan dengan mengestimasi deret *noise* (n_t) dengan persamaan (2.44) (Wei, 2006) :

$y_t = \hat{v}(B)x_t + n_t$

$n_t = y_t - \hat{v}(B)x_t$

$$n_t = y_t - \frac{\widehat{\omega}(B)}{\hat{\delta}(B)} x_{t-b} \qquad (2.44)$$

dimana $\hat{v}(B) = \frac{\widehat{\omega}(B)}{\hat{\delta}(B)} x_{t-b}$, estimasi dari $\widehat{\omega}(B)$ dan $\hat{\delta}(B)$ tidak dapat dihitung secara analitik sehingga menggunakan bantuan *software*.

e. Penetapan (p_n, q_n) dari Model Deret *Noise*

Selanjutnya, penetapan orde p_n dan q_n dengan model ARIMA dari deret *noise* (n_t) dapat ditulis pada persamaan (2.45) :

$$\phi_n(B)n_t = \theta_n(B)a_t \qquad (2.45)$$

Sesudah menggunakan persamaan deret *noise* (n_t) nilai-nilai n_t dianalisis dengan cara ARIMA biasa yang dapat dilihat pada subbab (2.6.3) sampai diperoleh nilai (p_n dan q_n).

2.8.2 Tahap Kedua : Penaksiran Parameter Model Fungsi Transfer

Model fungsi transfer *single input* adalah (Wei, 2006) :

$$y_t = \frac{\omega(B)}{\delta(B)} x_{t-b} + \frac{\theta(B)}{\phi(B)} a_t \qquad (2.46)$$

Penaksiran parameter model fungsi transfer pada persamaan (2.46) meliputi estimasi parameter yaitu :

$\delta = (\delta_1, \dots, \delta_r)', \omega = (\omega_0, \omega_1, \dots, \omega_s)', \phi = (\phi_1, \dots, \phi_p)', \theta = (\theta_1, \dots, \theta_q)'$ dan σ_a^2. Untuk mencari taksiran awal terbaik pada persamaan (2.46) digunakan *Conditional Maximum Likelihood*. Diasumsikan bahwa $a_t \sim N(0, \sigma^2)$, sehingga fungsi *likelihood* menggunakan fungsi distribusi normal ditulis pada persamaan (2.46) :

$$L(\delta, \omega, \phi, \theta, \sigma_a{}^2) = \prod_{t=1}^{n} f(a_{t,}\sigma_a{}^2) = (2\pi\sigma_a{}^2)^{-\frac{n}{2}} exp\left[-\frac{1}{2\sigma_a{}^2}\sum_{t=1}^{n} a_t{}^2\right]$$

$$\ln L(\delta, \omega, \phi, \theta, \sigma_a{}^2) = -\frac{1}{2n}\ln(2\pi) - \frac{1}{2n}\ln(2\sigma^2) - \frac{1}{2\sigma_a{}^2}\sum_{t=1}^{n} a_t{}^2 \qquad (2.47)$$

dimana $a_t = \frac{\delta(B)\phi(B)y_t}{\phi(B\omega(B)x_{t-b} + (B)\phi(B)}$

Penaksiran parameter tersebut diperoleh harus dengan menggunakan metode iterasi karena tidak dapat diselesaikan secara analitik, sehingga perhitungannya diselesaikan dengan menggunakan bantuan *software*.

2.8.3 Tahap Ketiga : Pemeriksaan Diagnosis Model Fungsi Transfer

Menurut Wei (2006), diagnosis model bertujuan untuk menguji apakah asumsi bahwa a_t merupakan *white noise* dan bebas terhadap deret *input* yang telah diputihkan dan disesuaikan (α_t) telah terpenuhi. Jika asumsi telah terpenuhi maka model fungsi transfer yang telah diuji ini merupakan model fungsi transfer yang layak untuk digunakan dalam peramalan. Terdapat dua macam pengujian dalam diagnostik, yaitu:

a. Pemeriksaan Autokorelasi Nilai Sisa dari Model (b,s,r) yang Menghubungkan Deret *Input* dan *Output*

Model untuk *noise* dikatakan layak jika koefisien ACF dan PACF dari *noise* tidak menunjukkan suatu pola tertentu (Wei, 2006). Caranya adalah dengan memeriksa autokorelasi dan korelasi residualnya. Pemeriksaan autokorelasi untuk nilai sisa menggunakan uji χ^2 Box-Pierce dengan hipotesis (Makidakis:1999,481) :

H$_0$: Autokorelasi pada deret sisaan (a_t) tidak signifikan

H$_1$: Autokorelasi pada deret sisaan (a_t) signifikan

Statistik uji:

$$Q = (n - r - s - b)\sum_{k=1}^{m} \rho_a^2\,(k) \tag{2.48}$$

dengan :

n : banyak pengamatan

b, s, r : parameter model fungsi transfer

m : *lag* terbesar yang diperhatikan

(ρ_a) : autokorelasi residual untuk *lag k*

Selanjutnya membandingkan hasilnya dengan tabel distribusi χ^2 dengan taraf signifikansi α, derajat bebas $m - p_n - q_n$ (merupakan nilai *autoregressive* dan *moving average* dari deret *noise*) dan tolak H$_0$ jika $Q \geq \chi^2_{(\alpha,df)}$ atau p-value < α. Model fungsi untuk deret *noise* sudah layak jika H$_0$ diterima yaitu tidak terdapat autokorelasi nilai sisaan a_t.

b. Pemeriksaan Korelasi Silang

Pemeriksaan korelasi silang antara nilai sisa dengan deret *noise* yang telah diputihkan menggunakan statistik uji Q dengan hipotesis (Makridakis:1999,483) :

H$_0$: $\rho_{a\alpha}(k) = 0$ (Tidak terdapat korelasi silang antara deret (α_t) dan nilai sisaan (a_t)

H$_1$: Minimal ada satu $\rho_{a\alpha}(k) \neq 0$ (Terdapat korelasi silang antara deret deret (α_t) dan nilai sisaan a_t)

Statistik uji :

$$Q = (n_\alpha - (s + b + p_x))\sum_{k=1}^{m} \rho_a^2\,(k) \tag{2.49}$$

dengan :

n_{α} : banyak data pada deret (x yang telah *white noise*),

m : *lag* maksimum,

p_x : jumlah parameter AR pada model ARIMA dengan deret *input* (x_t), *s* dan *b* adalah parameter yang diperoleh dari hasil perhitungan.

Hasilnya dibandingkan dengan tabel χ^2 dengan derajat bebas m-r-s dengan kriteria uji, tolak H_0 jika $Q \geq \chi^2_{(\alpha,df)}$ atau *p-value* < α.

2.8.4 Tahap Keempat : Pemodelan Fungsi Transfer *Input* Ganda

Pemodelan fungsi transfer *input* ganda dilakukan dengan cara memodelkan secara bersamaan seluruh variabel, persamaan fungsi transfer yaitu

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

dimana : $i = 1, 2, \dots, p$

Secara teoritis, sebetulnya tahap-tahap pembentukan fungsi transfe *input* ganda sama dengan *single input* hanya saja tahapnya menjadi lebih banyak karena variabel input yang banyak.

Untuk setiap variabel *input* $x_t^{(i)}$ dan variabel *output* (y_t), tahap aling awal adalah mencari model ARIMA pada masing-masing variabel *input* $x_t^{(i)}$dimana $i = 1,2, \dots p$. Setelah didapat model ARIMA pada masing-masing *input*, maka langkah selanjutnya adalah proses pemutihan atau pemutihan deret pada *input* dan deret *output*. Pemutihan deret *input* dilambangkan dengan α_t. Untuk variabel *input* lebih dari satu atau multivariat, persamaan pemutihan deret *input* dapat ditulis dengan :

$$\alpha_{t(i)} = \frac{\phi_x^{(i)}(B)}{\theta_x^{(i)}(B)} x_t^{(i)}$$

Parameter $\phi_x^{(i)}(B)$ dan $\theta_x^{(i)}(B)$ didapat dari model ARIMA yang sudah dicari pada tahap sebelumnya. Setelah tahap pemutihan deret *input* selesai, selanjutnya pemutihan deret *output*. Sama halnya dengan pemutihan deret *input*, persamaan pemutihan deret *output* hanya mengganti koefisien α_t menjadi β_t dan untuk setiap $x_t^{(i)}$ pada pemutihan deret *input* berubah menjadi y_t. Persamaan nya sebagai berikut:

$$\beta_{t(i)} = \frac{\phi_x^{(i)}(B)}{\theta_x^{(i)}(B)} y_t$$

Kemudian tahap selanjutnya adalah perhitungan korelasi silang (CCF) dari deret *input* terhadap deret *output* yang telah diputihkan untuk setiap $\alpha_{t(i)}$ terhadap $\beta_{t(i)}$ yang berguna untuk menentukan bobot respon *impuls* yang digunakan untuk penetapan orde b, s, r dengan persamaan :

$$\rho_{\alpha\beta(k)}^{(i)} \triangleq r_{\alpha\beta}^{(i)} = \frac{S_{\alpha_{t(i)}\beta_{t(i)}}}{S_{\alpha_{t(i)}} S_{\beta_{t(i)}}}$$

selanjutnya, penaksiran bobot respon *impuls* untuk setiap ($v_k^{(i)}$) yang digunakan untuk menentukan orde b,s,r untuk setiap variabel x ke-*i* dengan persamaan :

$$v_k^{(i)} = \frac{r_{\alpha_{t(i)}\beta_{t(i)}}(k) s_{\beta_{t(i)}}}{s_{\alpha_{t(i)}}}$$

Jadi, ketika sudah mendapatkan nilai dari $v_k^{(i)}$ kita bisa menentukan orde dari b, s, r dengan cara melihat grafik dari $v_k^{(i)}$ yang sudah ditampilkan pada Tabel 2.3. Kemudian orde b, s, r untuk setiap $x_t^{(i)}$ dapat diduga sementara maka selanjutnya signifikansi parameter apakah signifikan atau tidak berdasarkan nilai AIC dan asumsi *white noise*. Jadi jika kita mempunyai variabel x_t sebanyak 3 maka nilai dari masing-masing parameter b, s, r dan parameter $\frac{\omega(B)^{(i)}}{\delta(B)^{(i)}} x_{t-b}^{(i)}$ akan berjumlah sebanyak 3. Kemudian, setelah mendapatkan orde b,s,r yang terbaik selanjutnya penaksiran deret *noise* gabungan untuk model fungsi transfer dengan persamaan :

$$n_{t(gab)} = y_t - \sum \frac{\omega(B)^{(i)}}{\delta(B)^{(i)}} x_{t-b}^{(i)}$$

Pada tahap ini dilakukan estimasi parameter pada model ARMA yang akan memperoleh orde (p_n dan q_n) untuk deret *noise* (n_t) gabungan. Tahap selanjutnya pemilihan kriteria model terbaik dari deret *noise* (n_t) berdasarkan nilai AIC yang terkecil dan memenuhi asumsi *white noise*. Kemudian deret *noise* (n_t) gabungan dimasukan ke dalam model fungsi transfer *input* ganda. Dari penjelasan diatas, model fungsi transfer *input* ganda jika digabungkan dengan deret *noise* gabungan ($n_{t(gab)}$) akan diperoleh model :

$$y_t = \frac{\omega(B)^{(1)}}{\delta(B)^{(1)}} x_{t-b}^{(1)} + \frac{\omega(B)^{(2)}}{\delta(B)^{(2)}} x_{t-b}^{(2)} + \cdots + \frac{\omega(B)^{(p)}}{\delta(B)^{(p)}} x_{t-b}^{(p)} + n_{t(gab)}$$

$$y_t = \frac{\omega(B)^{(1)}}{\delta(B)^{(1)}} x_{t-b}^{(1)} + \frac{\omega(B)^{(2)}}{\delta(B)^{(2)}} x_{t-b}^{(2)} + \cdots + \frac{\omega(B)^{(p)}}{\delta(B)^{(p)}} x_{t-b}^{(p)} + \frac{\theta_q(B)}{\phi_p(B)} a_t$$

dimana $i = 1,2,3, \ldots p$. Kemudian dilakukan penaksiran parameter model dengan metode *Maximum Likelihood* seperti persamaan (2.44) karena tidak dilakukan secara manual tetapi dilakukan menggunakan *software* SAS yang akan menghasilkan estimasi dari parameter $(\delta, \omega, \phi, \theta)$. Selanjutnya pemeriksaan diagnosis dari model *input* ganda untuk mengetahui kelayakan suatu model. Perlu dilakukan pemeriksaan terhadap kesesuaian deret *noise* dan ada tidaknya autokorelasi residual (a_t) pada model dengan menggunakan statistik uji Q pada persamaan (2.48) dengan membandingkan hasilnya pada tabel distribusi λ^2 atau dapat dilihat dari *p-value* dengan syarat *pvalue* > α. Kemudian pemeriksaan korelasi silang antara masing-masing deret *input* yang telah diputihkan ($\alpha_{t(i)}$) dengan nilai sisaan (a_t) menggunakan statistik uji Q yang dapat dilihat pada persamaan (2.49).

2.9 Pemilihan Model Terbaik pada Model Fungsi Transfer *Input* Ganda

Ketetapan pemilihan model peramalan jika terdapat dua atau lebih model deret waktu dilakukan pemilihan kriteria model, tujuannya untuk memilih model yang layak digunakan dalam peramalan. Salah satu pendekatan yang dapat digunakan adalah AIC (*Akaike's Information Criterion*). Nilai AIC semakin kecil maka model yang didapat semakin baik. Persamaan AIC adalah (Wei, 2006) :

$$AIC = \ln(\hat{\sigma}^2) + \frac{2}{n}r \tag{2.50}$$

dengan :

n : banyak pengamatan residual

$\hat{\sigma}^2$: jumlah kuadrat residual dibagi banyak pengamatan

r : banyak parameter model ARIMA

2.10 Multikolinieritas

Uji multikolinearitas adalah uji yang dilakukan untuk memastikan apakah di dalam sebuah model regresi ada korelasi atau kolinearitas antar variabel bebas. Korelasi yang dimaksud adalah hubungan yang linear atau hubungan yang kuat antara satu variabel bebas atau variabel prediktor dengan variabel prediktor lainnya di dalam sebuah model regresi. Korelasi itu dapat dilihat dengan nilai koefisien korelasi antara variabel bebas, nilai VIF dan *Tolerance*, nilai *Eigenvalue* dan *Condition Index*, serta nilai *standar error* koefisien beta atau koefisien regresi parsial. Multikolinearitas terjadi untuk setiap satu peubah prediktor berkorelasi tinggi dengan sejumlah peubah prediktor yang lainnya. Akibat jika terjadi masalah multikolineritas dalam data diantara peubah prediktor (x) diantaranya adalah kofisien regresi dan galat bakunya tak hingga kemudian jika variabel-variabel prediktor berkorelasi satu sama lain, maka variabel-variabel tersebut menjelaskan varians yang sama dalam mengestimasi variabel dependent (y), jadi penambahan variabel prediktor (x) tidak berpengaruh apa-

apa. Untuk memeriksa multikolinieritas digunakan *Variance Inflation Factors* (VIF) dengan rumus:

$$\text{VIF} = \frac{1}{1-R_i^2} \tag{2.51}$$

Dengan R_i^2 adalah koefisien determinasi ganda antara peubah prediktor x_i , dengan peubah prediktor lainnya. Apabila nilai VIF>10 maka peubah tersebut dikatakan mempunyai masalah multikolinieritas yang signifikan dengan peubah bebas lainnya (Achmad, 2010).