

# **PERBANDINGAN ARIMA DAN *ARTIFICIAL NEURAL NETWORKS* DALAM PERAMALAN JUMLAH POSITIF COVID-19 DI DKI JAKARTA**

**TRI WAHYUNI**



**DEPARTEMEN STATISTIKA  
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
INSTITUT PERTANIAN BOGOR  
BOGOR  
2021**

- Hak Cipta Dilindungi Undang-undang
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
    - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah
    - b. Pengutipan tidak merugikan kepentingan yang wajar IPB University.
  2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IPB University.



@Hak cipta milik IPB University

IPB University



IPB University  
— Bogor Indonesia —

Hak Cipta Dilindungi Undang-undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah
  - b. Pengutipan tidak merugikan kepentingan yang wajar IPB University.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IPB University.

Perpustakaan IPB University

## PERNYATAAN MENGENAI SKRIPSI DAN SUMBER INFORMASI SERTA PELIMPAHAN HAK CIPTA

Dengan ini saya menyatakan bahwa skripsi dengan judul “Perbandingan ARIMA dan *Artificial Neural Networks* dalam Peramalan Jumlah Positif Covid-19 di DKI Jakarta” adalah karya saya dengan arahan dari dosen pembimbing dan belum diajukan dalam bentuk apa pun kepada perguruan tinggi mana pun. Sumber informasi yang berasal atau dikutip dari karya yang diterbitkan maupun tidak diterbitkan dari penulis lain telah disebutkan dalam teks dan dicantumkan dalam Daftar Pustaka di bagian akhir skripsi ini.

Dengan ini saya melimpahkan hak cipta dari karya tulis saya kepada Institut Pertanian Bogor.

Bogor, Juli 2021

Tri Wahyuni  
G14170006



## ABSTRAK

TRI WAHYUNI. Perbandingan ARIMA dan *Artificial Neural Networks* dalam Peramalan Jumlah Positif Covid-19 di DKI Jakarta. Dibimbing oleh INDAHAWATI dan KUSMAN SADIK.

DKI Jakarta menjadi pusat penyebaran Covid-19. Hal ini ditunjukkan dari jumlah kumulatif positif Covid-19 di DKI Jakarta lebih tinggi dibandingkan dengan provinsi lain. Tingginya kasus di DKI Jakarta menjadi perhatian bagi semua kalangan, sehingga perlu dilakukan peramalan untuk meramalkan jumlah positif Covid-19 periode selanjutnya. Peramalan yang akurat dibutuhkan untuk mendapatkan hasil yang lebih baik. Penelitian ini membandingkan metode *Autoregressive Integrated Moving Average* (ARIMA) dan *Artificial Neural Networks* (ANN) dalam peramalan jumlah positif Covid-19 di DKI Jakarta. Akurasi peramalan dihitung menggunakan nilai *Root Mean Square Error* (RMSE), *Mean Absolute Percentage Error* (MAPE), dan korelasi. Hasil penelitian menunjukkan bahwa model terbaik untuk peramalan jumlah positif Covid-19 di DKI Jakarta adalah ARIMA(0,1,1) *with drift* dengan nilai MAPE sebesar 15,748, RMSE sebesar 268,808, dan korelasi antara nilai ramalan dengan nilai aktual sebesar 0,845. Peramalan menggunakan model ARIMA(0,1,1) *with drift* dan BP(3,10,1) menghasilkan ramalan terbaik pada panjang periode peramalan enam minggu ke depan.

Kata kunci: ANN, ARIMA, Covid-19, peramalan

## ABSTRACT

TRI WAHYUNI. Comparison of ARIMA and Artificial Neural Networks in Forecasting the Number of Covid-19 Positive in DKI Jakarta. Supervised by INDAHAWATI and KUSMAN SADIK.

DKI Jakarta is the center of the spread of Covid-19. This is indicated by the higher cumulative number of Covid-19 positive in DKI Jakarta compared to other provinces. The high number of cases in DKI Jakarta is a concern for all groups, so it is necessary to do forecasting to predict the number of Covid-19 positive in the next period. Accurate forecasting is needed to get better results. This study compares the *Autoregressive Integrated Moving Average* (ARIMA) and *Artificial Neural Networks* (ANN) methods in predicting the number of Covid-19 positive in DKI Jakarta. Forecasting accuracy is calculated using the value of *Root Mean Square Error* (RMSE), *Mean Absolute Percentage Error* (MAPE), and correlation. The results show that the best model for forecasting the number of Covid-19 positive in DKI Jakarta is ARIMA(0,1,1) *with drift*, with a MAPE value of 15.748, an RMSE of 268.808, and the correlation between the forecast value and the actual value of 0.845. Forecasting using ARIMA(0,1,1) *with drift* and BP(3,10,1) models produces the best forecast for the long forecasting period of the next six weeks.

Keywords: ANN, ARIMA, Covid-19, forecast



@Hak cipta milik IPB University

IPB University



IPB University  
— Bogor Indonesia —

Hak Cipta Dilindungi Undang-undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah
  - b. Pengutipan tidak merugikan kepentingan yang wajar IPB University.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IPB University.

© Hak Cipta milik IPB, tahun 2021<sup>1</sup>  
Hak Cipta dilindungi Undang-Undang

*Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan atau menyebutkan sumbernya. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik, atau tinjauan suatu masalah, dan pengutipan tersebut tidak merugikan kepentingan IPB.*

*Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apa pun tanpa izin IPB.*

# **PERBANDINGAN ARIMA DAN *ARTIFICIAL NEURAL NETWORKS* DALAM PERAMALAN JUMLAH POSITIF COVID-19 DI DKI JAKARTA**

**TRI WAHYUNI**

Skripsi  
sebagai salah satu syarat untuk memperoleh gelar  
Sarjana pada  
Program Studi Statistika dan Sains Data

**DEPARTEMEN STATISTIKA  
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
INSTITUT PERTANIAN BOGOR  
BOGOR  
2021**



@Hak cipta milik IPB University

IPB University



IPB University  
— Bogor Indonesia —

Hak Cipta Dilindungi Undang-undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah
  - b. Pengutipan tidak merugikan kepentingan yang wajar IPB University.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IPB University.

Penguji pada Ujian Skripsi:  
Dr. Farit Mochamad Afendi, S.Si., M.Si.

Perpustakaan IPB University



Judul Skripsi : Perbandingan ARIMA dan *Artificial Neural Networks* dalam  
Peramalan Jumlah Positif Covid-19 di DKI Jakarta

Nama : Tri Wahyuni  
NIM : G14170006

@Hak cipta milik IPB University

Disetujui oleh

Pembimbing 1:  
Dr. Ir. Indahwati, M.Si.

---

Pembimbing 2:  
Dr. Kusman Sadik, S.Si., M.Si.

---

Diketahui oleh

Ketua Departemen Statistika:  
Dr. Anang Kurnia, S.Si., M.Si.  
NIP. 197308241997021001

---

Tanggal Ujian:  
9 Juni 2021

Tanggal Lulus:



## PRAKATA

Puji dan syukur penulis panjatkan kepada Allah subhanaahu wa ta'ala atas segala karunia-Nya, sehingga karya ilmiah yang berjudul Perbandingan ARIMA dan *Artificial Neural Networks* dalam Peramalan Jumlah Positif Covid-19 di DKI Jakarta ini berhasil diselesaikan. Shalawat dan salam kepada Nabi Muhammad SAW beserta keluarga, sahabat, dan umatnya. Selesaiannya penyusunan karya ilmiah ini tidak lepas dari dukungan, saran, dan kritik dari berbagai pihak. Oleh karena itu, penulis mengucapkan terima kasih kepada:

1. Ayah, Umak, Kakak, Adik, semua keluarga besar atas dukungan dan doa yang selalu diberikan kepada penulis.
2. Ibu Dr. Ir. Indahwati, M.Si. dan Bapak Dr. Kusman Sadik, S.Si., M.Si. selaku pembimbing yang telah banyak memberikan saran dan nasihat kepada penulis, sehingga karya ilmiah ini bisa terselesaikan.
3. Seluruh dosen Departemen Statistika IPB atas ilmu yang sangat bermanfaat bagi penulis.
4. Staf tata usaha Departemen Statistika IPB atas bantuan dalam kelancaran administrasi.
5. Mahasiswa Departemen Statistika IPB angkatan 54 terkhusus Khusnul, Syahnita, Amalia, Anistia, Elda yang selalu memberikan kebahagiaan dan membantu penulis dalam menyelesaikan karya ilmiah ini.
6. Mahasiswa IKPB Cabang Bogor terkhusus Lilik, Inee, Gio yang telah memberikan dukungan kepada penulis dan arti persahabatan yang tidak terhitung nilainya.
7. Asrama Mahasiswa Belitung yang telah memberikan fasilitas tempat tinggal kepada penulis.

Semoga karya ilmiah ini bermanfaat bagi pihak yang membutuhkan dan bagi kemajuan ilmu pengetahuan..

Bogor, Juli 2021

Tri Wahyuni



## DAFTAR ISI

DAFTAR TABEL	viii
DAFTAR GAMBAR	viii
DAFTAR LAMPIRAN	viii
PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan	2
1.4 Manfaat	2
TINJAUAN PUSTAKA	3
2.1 Eksplorasi Data Deret Waktu	3
2.2 Kestasioneran Data Deret Waktu	3
2.3 Model ARIMA	4
2.4 Proses Peramalan ARIMA	7
2.5 Kriteria Pemilihan Model	8
2.6 <i>Artificial Neural Networks</i> (ANN)	8
2.7 Arsitektur jaringan ANN	9
2.8 Metode <i>Backpropagation</i>	10
2.9 Validasi Model	13
III METODE	15
3.1 Data	15
3.2 Prosedur Analisis Data	15
3.3 Diagram Analisis Data	16
IV HASIL DAN PEMBAHASAN	18
4.1 Eksplorasi Data	18
4.2 Kestasioneran Data	19
4.3 Identifikasi Model dan Pendugaan Parameter	20
4.4 Diagnostik Model	21
4.5 <i>Artificial Neural Networks</i>	25
4.6 Perbandingan ARIMA dan ANN	28
V SIMPULAN DAN SARAN	32
5.1 Simpulan	32
5.2 Saran	32
DAFTAR PUSTAKA	33
LAMPIRAN	35
RIWAYAT HIDUP	39

## DAFTAR TABEL

1	Transformasi Box-Cox	4
2	Selang nilai MAPE	14
3	Nilai dugaan parameter model ARIMA untuk jumlah positif Covid-19 di DKI Jakarta	21
4	Hasil uji Ljung-Box sisaan pada model ARIMA	21
5	Pemeriksaan kenormalan sisaan	22
6	Nilai AIC dan BIC dari model tentatif awal	23
7	Nilai dugaan parameter model ARIMA untuk jumlah positif Covid-19 di DKI Jakarta hasil <i>overfitting</i>	23
8	Hasil uji Ljung-Box sisaan model ARIMA hasil <i>overfitting</i>	24
9	Perbandingan nilai AIC dan BIC terkecil pada model ARIMA	24
10	Jaringan hasil <i>trial and error</i> metode ANN	26
11	Hasil validasi model-model terbaik dengan data latih	29
12	Hasil validasi model-model terbaik dengan data uji	30
13	Nilai akurasi pada berbagai panjang periode peramalan	30

## DAFTAR GAMBAR

1	Tipe-tipe arsitektur jaringan ANN	9
2	Jaringan syaraf manusia (A) dan skema dari ANN untuk satu <i>output</i> (B)	10
3	Diagram alir metode ANN	16
4	Diagram alir metode ARIMA	17
5	Plot data jumlah harian positif Covid-19 di DKI Jakarta	18
6	Plot Box-Cox jumlah positif Covid-19 di DKI Jakarta yang belum stasioner (A) dan sudah stasioner (B) dalam ragam	19
7	Plot ACF (A) dan PACF (B) data jumlah positif Covid-19 di DKI Jakarta yang belum stasioner dalam rata-rata	19
8	Plot ACF (A) dan PACF (B) data jumlah positif Covid-19 di DKI Jakarta yang sudah stasioner dalam rata-rata	20
9	Arsitektur BP(3,10,1)	27
10	Plot data aktual dan hasil peramalan dengan data latih pada ARIMA (0,1,1) <i>with drift</i> (A) dan BP (3,10,1) (B)	28
11	Hasil peramalan model ARIMA dan ANN dengan data uji	29

## DAFTAR LAMPIRAN

1	Plot jumlah positif Covid-19 periode bulanan di DKI Jakarta	36
2	Plot jumlah positif Covid-19 periode mingguan di DKI Jakarta	36
3	Plot RACF dan RPACF model ARIMA(0,1,1)	36
4	Plot RACF dan RPACF model ARIMA(3,1,0)	37



5	Plot sebaran normal sisaan model ARIMA(0,1,1)	37
6	Plot sebaran normal sisaan model ARIMA(3,1,0)	37
7	Plot RACF dan RPACF model ARIMA(0,1,1) <i>with drift</i>	37
8	Plot RACF dan RPACF model ARIMA(1,1,2)	38
9	Plot sebaran normal sisaan model ARIMA(0,1,1) <i>with drift</i>	38
10	Plot sebaran normal sisaan model ARIMA(1,1,2)	38
11	Plot PACF jumlah positif Covid-19 di DKI Jakarta dengan data latih yang belum stasioner dalam ragam dan rataaan	38

## I PENDAHULUAN

### 1.1 Latar Belakang

*Severe Acute Respiratory Syndrome Coronavirus 2* (SARS-Cov-2) merupakan virus jenis baru yang menyebabkan penyakit *Coronavirus Disease* 2019 (Covid-19). *Coronavirus disease* jenis baru ditemukan pada manusia sejak kejadian luar biasa di Wuhan, China pada 31 Desember 2019. Angka kematian Covid-19 kurang dari 5% lebih kecil dibandingkan SARS sebesar 9,6%, tetapi penyebaran Covid-19 lebih luas dan cepat ke beberapa negara dibanding dengan SARS (Kemenkes 2020). Virus Covid-19 saat ini sangat mudah menular dan menyebabkan kematian akibat susah bernafas. Terlepas dari peningkatan fasilitas pemerintah, pengujian, dan pemberian vaksinasi, jumlah kasus positif terus mengalami penambahan setiap harinya.

Indonesia adalah negara dengan kasus jumlah positif Covid-19 terbanyak ke-20 pada 18 Desember 2020. Indonesia pertama kali mengonfirmasi kasus positif Covid-19 pada 2 Maret 2020 sebanyak dua orang. DKI Jakarta menjadi episentrum penyebaran Covid-19 di Indonesia. Menurut Satuan Tugas Penanganan Covid-19, pada 18 Desember 2020 DKI Jakarta menyumbang 24,6% kasus positif Covid-19 dari jumlah kasus terkonfirmasi nasional. Jumlah kumulatif kasus terkonfirmasi positif Covid-19 di DKI Jakarta terbanyak dibandingkan provinsi lain. Tingginya penambahan kasus Covid-19 di DKI Jakarta menjadi perhatian bagi semua kalangan, sehingga perlu dilakukan peramalan untuk mengetahui gambaran penambahan kasus pada periode selanjutnya.

Hasil peramalan yang baik dihasilkan jika metode peramalan yang digunakan sesuai dengan kondisi data. Data jumlah positif Covid-19 di DKI Jakarta memiliki *trend* naik dan tidak stasioner. Salah satu metode yang dapat digunakan untuk peramalan data yang tidak stasioner adalah metode *Autoregressive Integrated Moving Average* (ARIMA). Model ARIMA disebut juga dengan model Box-Jenkins yang mengasumsikan fungsi linear dari beberapa pengamatan dimasa lalu. Asumsi kestasioneran menjadi hal yang harus dipenuhi pada model ARIMA. Namun, ketika model linear menghasilkan tingkat akurasi peramalan yang kecil dan kesalahan peramalan yang besar, kemungkinan model taklinear (*nonlinear*) mampu menjelaskan dan meramalkan dengan lebih baik dibandingkan dengan model linear (Zhang 2003). Selain itu, dalam dunia nyata banyak sekali ditemukan data yang taklinear, sehingga metode ARIMA mungkin kurang sesuai untuk menggambarkan data tersebut. *Artificial Neural Networks* (ANN) merupakan model yang mampu menjelaskan permasalahan yang kompleks dengan hubungan yang taklinear untuk peramalan jangka panjang (Adeyinka dan Muhajirine 2020). ANN adalah salah satu representasi buatan dari otak manusia yang menyimulasikan proses pembelajaran melalui komputasi. Adebiyi *et al.* (2014) menunjukkan bahwa model ARIMA dan ANN dapat mencapai ramalan yang baik dalam penerapannya pada masalah kehidupan nyata.

Penelitian sebelumnya, Sahai *et al.* (2020) melakukan peramalan dengan pemodelan ARIMA di lima negara yang mengalami dampak terbesar Covid-19 dan menghasilkan MAPE kurang dari 5%. Wiecek *et al.* (2020) menunjukkan metode ANN memiliki nilai akurasi yang lebih tinggi dan mencapai akurasi



maksimum dalam iterasi yang lebih sedikit dibandingkan RNN dalam peramalan penyebaran Covid-19. Oleh karena itu, penelitian ini dilakukan dengan pengujian beberapa parameter untuk mengidentifikasi nilai parameter terbaik dari model linear ARIMA dan model taklinear ANN dalam peramalan jumlah positif Covid-19 di DKI Jakarta.

## 1.2 Rumusan Masalah

Berdasarkan uraian pada latar belakang didapatkan rumusan masalah dari penelitian ini adalah sebagai berikut:

1. Bagaimana membangun model terbaik ARIMA dan ANN pada kasus jumlah positif Covid-19 di DKI Jakarta?
2. Manakah model terbaik antara metode ARIMA dan ANN untuk peramalan jumlah positif Covid-19 di DKI Jakarta?

## 1.3 Tujuan

Penelitian ini bertujuan:

1. Membangun model terbaik ARIMA dan ANN pada kasus jumlah positif Covid-19 di DKI Jakarta.
2. Membandingkan model terbaik ARIMA dan ANN pada peramalan jumlah positif Covid-19 di DKI Jakarta.

## 1.4 Manfaat

Penelitian ini diharapkan dapat menambah dinamika keilmuan terutama di bidang peramalan menggunakan data yang cenderung taklinear dengan metode *Autoregressive Integrated Moving Average* (ARIMA) dan *Artificial Neural Networks* (ANN). Selain itu, bagi pihak yang ingin melakukan kajian mendalam mengenai keilmuan tentang peramalan, khususnya peramalan jumlah kasus positif Covid-19, diharapkan penelitian ini dapat dijadikan sebagai referensi bagi penelitian selanjutnya.

## II TINJAUAN PUSTAKA

### 2.1 Ekplorasi Data Deret Waktu

Data deret waktu adalah data yang dikumpulkan berdasarkan urutan waktu. Data dicatat berdasarkan periode waktu mingguan, bulanan, tahunan, atau periode tertentu lainnya dalam selang waktu yang sama. Data deret waktu dapat dimodelkan untuk meramalkan masa yang akan datang (Cryer dan Chan 2008). Analisis data deret waktu diawali dengan pengeplotan data. Plot data bertujuan untuk menemukan pola yang akan digunakan untuk menduga metode yang sesuai dengan data deret waktu. Pola seperti *trend* dan musiman mudah dilihat pada plot data deret waktu dibandingkan menggunakan histogram dan diagram dahan daun (Montgomery *et al.* 2015). Pola data yang dihasilkan dapat dibedakan menjadi pola konstan, *trend*, musiman, dan siklis. Pola *trend* terjadi jika ada kecenderungan nilai data untuk naik atau turun pada jangka waktu yang panjang. Jumlah peubah yang digunakan pada analisis data deret waktu dibedakan menjadi dua, yaitu data deret waktu peubah tunggal dan peubah ganda. Metode yang digunakan pada analisis data deret waktu peubah tunggal, yaitu *single moving average*, *double moving average*, *single exponential smoothing*, *double exponential smoothing*, ARIMA, dan SARIMA.

### 2.2 Kestasioneran Data Deret Waktu

Data deret waktu dikatakan stasioner apabila perilaku data tersebut berfluktuasi di sekitar nilai tengah dan ragam yang relatif konstan untuk seluruh periode waktu. Data yang stasioner menunjukkan bahwa data tersebut stabil. Namun, pada kenyataannya banyak data deret waktu yang tidak memenuhi asumsi kestasioneran. Kestasioneran dibagi menjadi dua, yaitu stasioner dalam ragam dan rata-rata. Transformasi data dilakukan jika data deret waktu tidak stasioner dalam ragam, sedangkan pembedaan (*differencing*) dilakukan untuk menghilangkan pola yang sistematis pada data. Secara eksplorasi kestasioneran data dalam rata-rata dapat dilihat menggunakan plot *Autocorrelation Function* (ACF). Plot ACF yang menurun secara perlahan menunjukkan data tidak stasioner dalam rata-rata, jika plot ACF menurun secara cepat mendekati nol setelah *lag* satu atau dua maka data deret waktu stasioner dalam rata-rata (Montgomery *et al.* 2015). Kestasioneran dalam ragam diperiksa menggunakan plot Box-Cox. Data yang tidak stasioner dalam ragam akan dilakukan transformasi Box-Cox sesuai dengan *rounded value* ( $\lambda$ ) yang dihasilkan. Wei (2006), satu keuntungan besar menggunakan transformasi adalah dapat memperlakukan  $\lambda$  sebagai parameter transformasi dan memperkirakan nilainya dari data. Selain itu, dapat memasukkan  $\lambda$  sebagai parameter dalam model dan memilih nilai  $\lambda$  yang memberikan kesalahan kuadrat rata-rata sisaan yang minimum. Secara umum dapat digunakan transformasi sebagai berikut:

$$T(Y_t) = \frac{Y_t^\lambda - 1}{\lambda} \quad (1)$$

Transformasi ini hanya dilakukan jika data deret waktu bernilai positif. Tabel 1 menunjukkan kasus dengan  $\lambda$  yang umum digunakan dan transformasi terkait  $\lambda$  yang dihasilkan (Wei 2006):

Tabel 1 Transformasi Box-Cox

<i>Rounded value</i> ( $\lambda$ )	Transformasi ( $Y^*$ )
-1,0	$\frac{1}{Y_t}$
-0,5	$\frac{1}{\sqrt{Y_t}}$
0,0	$\ln Y_t$
0,5	$\sqrt{Y_t}$
1,0	$Y_t$

Kestasioneran rata-rata diketahui dengan menggunakan uji *Augmented Dickey Fuller* (ADF). Persamaan model ADF sebagai proses  $AR(p)$ ,  $p \geq 1$  (Rusdi 2011):

$$\Delta y_t = a_0 + \gamma y_{t-1} + \beta_i \sum_{i=2}^p \Delta y_{t-i+1} + \varepsilon_t \quad (2)$$

Keterangan:

$\Delta y_t$  :  $y_t - y_{t-1}$

$\gamma$  :  $\sum_{i=1}^p a_i - 1$

$\beta_i$  :  $\sum_{j=1}^p a_j$

$y_{t-1}$  : pengamatan pada waktu ke- ( $t - 1$ )

$a_0$  : konstanta

$p$  : panjang ordo AR

$\gamma$  : koefisien dari  $y_{t-1}$

$\varepsilon_t$  : sisaan pada waktu ke- $t$  dengan asumsi  $\varepsilon_t \sim N(0, \sigma^2)$

Berdasarkan persamaan (2), berikut hipotesis yang digunakan pada uji ADF:

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

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

Statistik uji yang digunakan pada uji ADF adalah:

$$\tau = \frac{\hat{\gamma} - \gamma}{SE(\hat{\gamma})} \quad (3)$$

Keterangan:

$SE(\hat{\gamma})$  : galat baku dari  $\hat{\gamma}$

$\hat{\gamma}$  : penduga kuadrat terkecil dari  $\gamma$

Kriteria keputusan uji ADF adalah tolak  $H_0$  apabila nilai kritis statistik- $t$  kurang dari nilai kritis ADF atau nilai peluang dari statistik uji ADF kurang dari  $\alpha$  yang artinya data stasioner. Data yang tidak stasioner dalam rata-rata akan dilakukan pembedaan (*differencing*).

### 2.3 Model Autoregressive Integrated Moving Average (ARIMA)

Metode ARIMA adalah metode yang mengabaikan peubah independen (bebas). Metode ini menggunakan nilai sekarang dan nilai-nilai lampau dari peubah dependen (tidak bebas) untuk menghasilkan peramalan. Model ARIMA atau disebut model Box-Jenkins merupakan gabungan dari model *Autoregressive*



(AR) dengan ordo  $p$  dan model *Moving Average* (MA) dengan ordo  $q$  serta proses pembedaan (*differencing*) dengan ordo  $d$ . Pembedaan dilakukan untuk mengondisikan data deret waktu dalam keadaan yang stasioner. Menurut Adeyinka dan Muhajarine (2020), jika ingin menggunakan model ARIMA untuk peramalan setidaknya diperlukan 50 pengamatan yang tidak hilang untuk memberikan perkiraan yang baik dalam mengidentifikasi model. Bentuk umum model ARIMA dari data deret waktu adalah (Montgomery *et al.* 2015):

$$\phi_p(B)(1-B)^d y_t = \delta + \theta_q(B)\varepsilon_t \quad (4)$$

Keterangan:

$\phi_p(B)$	: $(1 - \phi_1 B - \dots - \phi_p B^p)$ , AR(p)
$\theta_q(B)$	: $(1 - \theta_1 B - \dots - \theta_q B^q)$ , MA(q)
$\phi_p$	: komponen <i>autoregressive</i> ordo $p$
$\theta_q$	: komponen <i>moving average</i> ordo $q$
$B$	: operator pergeseran mundur ( <i>backshift</i> )
$(1-B)^d$	: operator proses pembedaan ordo $d$
$y_t$	: pengamatan pada waktu ke- $t$
$\delta$	: konstanta
$\varepsilon_t$	: sisaan pada waktu ke- $t$ dengan asumsi $\varepsilon_t \sim N(0, \sigma^2)$

Prosedur pembentukan model ARIMA dilakukan dengan beberapa tahapan (Montgomery *et al.* 2015):

#### 1. Identifikasi Model

Identifikasi model dilakukan setelah data deret waktu dalam kondisi yang stasioner dalam ragam dan ratahan. Model tentatif ditentukan berdasarkan pola korelasi diri (ACF) dan korelasi diri parsial (PACF). Ciri proses MA( $q$ ) adalah *cuts off* pada plot ACF setelah *lag* ke- $q$  dan koefisien korelasi diri parsial menurun secara eksponensial atau berbentuk sinus dengan bertambahnya waktu ketertinggalan. Ciri dari proses AR( $p$ ) adalah *cuts off* pada plot PACF setelah *lag* ke- $p$  dan koefisien korelasi diri menurun secara eksponensial atau berbentuk sinus dengan bertambahnya waktu ketertinggalan. Jika plot ACF dan PACF menunjukkan *tails off* maka hal tersebut ciri proses ARMA( $p, q$ ) (Cryer dan Chan 2008).

#### 2. Pendugaan Parameter

Pendugaan parameter dilakukan dengan melihat signifikan parameter dari model tentatif yang sudah ditentukan pada proses identifikasi model. Jumlah parameter yang diduga bergantung pada banyaknya koefisien model awal yang terbentuk. Nilai parameter diuji dengan statistik uji untuk mengetahui apakah berpengaruh atau tidaknya parameter tersebut terhadap model yang terbentuk. Parameter dikatakan signifikan atau berpengaruh nyata terhadap model, jika  $p$ -value yang diperoleh lebih kecil dari nilai taraf nyata yang ditentukan. Metode yang biasa digunakan untuk menduga parameter, yaitu metode momen, metode kuadrat terkecil, dan metode kemungkinan maksimum. Penelitian ini menggunakan metode kemungkinan maksimum dari *default* software R.

### 3. Diagnostik Model

Proses ini dilakukan untuk memeriksa kelayakan model yang sudah diidentifikasi. Pemeriksaan ini dilakukan melalui pengujian asumsi model, sehingga diperoleh model yang memadai, jika model yang diidentifikasi tidak layak atau tidak memenuhi asumsi kebebasan dan kenormalan sisaan maka tahap identifikasi model kembali dilakukan untuk mendapatkan model yang lebih baik. Pemeriksaan diagnostik model juga dapat dilakukan dengan mengamati nilai autokorelasi diri sisaan (RACF) dan nilai autokorelasi diri parsial sisaan (RPACF). Diagnostik model didasarkan pada analisis sisaan, yaitu mengecek kebebasan sisaan menggunakan uji Ljung–Box dan kenormalan sisaan menggunakan uji Kolmogorov–Smirnov.

#### a. Uji Ljung–Box

Uji ini digunakan untuk memeriksa apakah  $k$  korelasi diri sisaan yang pertama dapat menunjukkan kebebasan sisaan.

Hipotesis yang digunakan adalah:

$H_0$  : tidak terdapat autokorelasi sisaan

$H_1$  : terdapat autokorelasi sisaan

Statistik uji (Montgomery *et al.* 2015):

$$Q_{LB} = T(T + 2) \sum_{k=1}^K \left( \frac{1}{T-k} \right) r_k^2 \quad (5)$$

Keterangan:

$T$  : banyaknya data

$k$  : lag ke- $k$

$K$  : lag maksimum

$r_k^2$  : koefisien autokorelasi antar sisaan pada lag ke- $k$

Kriteria keputusan adalah tak tolak  $H_0$  jika nilai statistik uji  $Q_{LB}$  lebih kecil dari nilai  $\chi^2_{\alpha, K-p-q}$  atau nilai peluang lebih besar dari nilai  $\alpha$  yang artinya tidak terdapat autokorelasi sisaan

#### b. Uji Kolmogorov–Smirnov

Uji ini digunakan untuk memeriksa kenormalan sisaan secara formal. Namun, secara eksplorasi kenormalan dapat diperiksa dengan membuat plot quantil-quantil (plot Q-Q) atau histogram untuk mengetahui sebaran sisaannya.

Hipotesis yang digunakan adalah:

$H_0$  : sisaan menyebar normal

$H_1$  : sisaan tidak menyebar normal

Statistik uji:

$$D = \sup_x |F_n(x) - F(x)| \quad (6)$$

Keterangan:

$F_n(x)$  : sebaran frekuensi kumulatif contoh

$F(x)$  : sebaran frekuensi kumulatif empirik

Kriteria keputusan adalah tak tolak  $H_0$  jika statistik uji  $D$  lebih kecil dari nilai kritis pada tabel Kolmogorov–Smirnov atau nilai peluang yang dihasilkan lebih besar dari  $\alpha$  yang artinya sisaan menyebar normal.

#### 4. Overfitting Model

*Overfitting* digunakan untuk mendapatkan kandidat-kandidat dari model ARIMA. *Overfitting* dilakukan dengan penambahan atau pengurangan *lag* ordo dari model tentatif pada model awal. Model yang didapat dari *overfitting* akan dilakukan pendugaan parameter dan pemeriksaan asumsi, jika hasil pendugaan signifikan dan asumsi terpenuhi maka model tersebut dipilih sebagai kandidat model terbaik (Cryer dan Chan 2008).

### 2.4 Proses Peramalan ARIMA

Proses peramalan dilakukan setelah model data deret waktu didapatkan. Model tersebut akan digunakan untuk memperkirakan pengamatan pada masa yang akan datang. Kriteria yang digunakan untuk mendapatkan peramalan yang baik adalah kesalahan kuadrat rata-rata sisaan dimana nilai harapan dari kuadrat kesalahan dugaan diminimalkan. Ramalan terbaik berdasarkan pada kesalahan kuadrat rata-rata sisaan adalah (Montgomery *et al.* 2015):

$$\hat{y}_{T+\tau}(T) = E[y_{T+\tau}|y_T, y_{T-1}, \dots] = \mu + \sum_{i=\tau}^{\infty} \psi_i \varepsilon_{T+\tau-i} \quad (7)$$

$$E[\varepsilon_{T+\tau-i}|y_T, y_{T-1}, \dots] = \begin{cases} 0 & \text{jika } i < \tau \\ \varepsilon_{T+\tau-i} & \text{jika } i \geq \tau \end{cases} \quad (8)$$

$T$  adalah waktu sekarang dan  $\tau$  adalah ramalan periode kedepan maka kesalahan peramalan (*forecast error*) dihitung dengan:

$$e_T(\tau) = y_{T+\tau} - \hat{y}_{T+\tau}(T) = \sum_{i=0}^{\tau-1} \psi_i \varepsilon_{T+\tau-i} \quad (9)$$

Nilai harapan dan ragam dapat dituliskan sebagai berikut:

$$E[e_T(\tau)] = 0 \quad (10)$$

$$\begin{aligned} Var[e_T(\tau)] &= Var[\sum_{i=0}^{\tau-1} \psi_i \varepsilon_{T+\tau-i}] = \sum_{i=0}^{\tau-1} \psi_i^2 Var(\varepsilon_{T+\tau-i}) \\ &= \sigma^2 \sum_{i=0}^{\tau-1} \psi_i^2 = \sigma^2(\tau), \quad \tau = 1, 2, 3, \dots \end{aligned} \quad (11)$$

Ragam kesalahan peramalan akan semakin besar dengan meningkatkan waktu peramalan ( $\tau$ ). Berdasarkan persamaan (10) dan (11), kesalahan peramalan diasumsikan menyebar  $N(0, \sigma^2)$ . Kesalahan untuk model ARIMA dapat dihitung:

$$\hat{\varepsilon}_t = y_t - [\delta + \sum_{i=1}^{p+d} \phi_i y_{t-i} - \sum_{i=1}^q \theta_i \hat{\varepsilon}_{t-i}] \quad (12)$$

Nilai awal  $\varepsilon_t$  akan diatur menjadi nol untuk  $t < p + d + 1$ . Hasil yang lebih akurat dapat diperoleh dengan metode yang mana nilai awal untuk  $t \leq 0$  dan  $y_t$  awal, yang terjadi sesaat sebelum dimulainya periode data dapat ditentukan dengan teknik *back-forecasting*. Teknik ini pada dasarnya menentukan nilai harapan bersyarat dari  $\varepsilon_t$  dan  $y_t$  awal sebelum periode data berdasarkan pada data yang tersedia untuk periode data tersebut, hingga yang dimulai pada waktu  $t = 1$ . Nilai  $y_{-t}$  memiliki hubungan peluang yang sama dengan barisan nilai  $y_1, y_2, \dots, y_n$ , seperti halnya dengan nilai  $y_{n+t+1}$  ke barisan nilai  $y_n, y_{n-1}, y_{n-2}, \dots, y_1$ . Jadi, untuk meramalkan nilai  $t + 1$  sebelum pengamatan dimulai, dapat mempertimbangkan apa yang akan menjadi perkiraan optimal  $t + 1$  periode setelah deret berakhir, lalu terapkan prosedur ini pada rangkaian terbalik, dengan kata lain *back-forecasting* meramalkan seri terbalik (Box *et al.* 2008).

## 2.5 Kriteria Pemilihan Model

Model terbaik adalah model yang memiliki nilai AIC dan BIC terkecil. Model dengan jumlah parameter sekecil mungkin dipilih untuk mewakili sebaran data (Adeyinka dan Muhajarine 2020). AIC merupakan penduga yang bias dan cenderung memilih model dengan jumlah parameter yang lebih banyak dibandingkan dengan BIC, sehingga untuk data yang besar BIC lebih baik digunakan karena akan lebih konsisten (Cryer dan Chan 2008). Menurut Montgomery *et al.* 2015, persamaan AIC dan BIC adalah:

$$AIC = \ln \left( \frac{\sum_{t=1}^T e_t^2}{T} \right) + \frac{2p}{T} \quad (13)$$

$$BIC = \ln \left( \frac{\sum_{t=1}^T e_t^2}{T} \right) + \frac{p \ln(T)}{T} \quad (14)$$

Keterangan:

$\sum_{t=1}^T e_t^2$  : kuadrat sisaan  
 $p$  : banyaknya parameter dalam model  
 $T$  : banyaknya data

## 2.6 Artificial Neural Networks (ANN)

Menurut Adeyinka dan Muhajarine (2020), ANN merupakan pemodelan taklinear yang mengikuti struktur dan fungsional otak manusia dan neuronnya untuk memecahkan masalah kompleks lebih cepat dengan tingkat kesalahan yang kecil. ANN akan cenderung menyimpan pengetahuan pada proses pelatihan yang kemudian akan diterapkan untuk peramalan. ANN memiliki tiga tipe lapisan, yaitu *input layer* (lapisan masukan), *hidden layer* (lapisan tersembunyi), dan *output layer* (lapisan keluaran) yang dijadikan sebagai dasar dalam penyusunan jaringan ANN. Neuron pada *input layer* terhubung pada neuron *hidden layer* dan neuron pada *hidden layer* terhubung dengan neuron *output layer*. ANN tidak memerlukan asumsi-asumsi seperti pada peramalan dengan metode statistika klasik untuk pemrosesan datanya. Pembobot menjadi penghubung dari tiap lapisan yang nilainya diatur dengan menggunakan fungsi aktivasi (Irsyadinna 2009). Model ANN dengan menggunakan algoritma *backpropagation* secara sistematis adalah (Adebiyi *et al.* 2014):

$$y_t = w_0 + \sum_{j=1}^q w_j \cdot g(v_{0j} + \sum_{i=1}^p v_{ij} \cdot y_{t-i}) + \varepsilon_t \quad (15)$$

Keterangan:

$w_j$  : bobot neuron *hidden layer* ke-j pada neuron *output*  
 $w_0$  : bobot bias neuron *hidden layer* pada neuron *output*  
 $v_{0j}$  : bias neuron *input layer* pada neuron *hidden layer* ke-j ( $j = 1, 2, 3, \dots, q$ )  
 $v_{ij}$  : bobot antara neuron *input layer* ke-i dengan neuron *hidden layer* ke-j ( $i = 1, 2, 3, \dots, p ; j = 1, 2, 3, \dots, q$ )  
 $g(.)$  : fungsi aktivasi dari *hidden layer*  
 $y_{t-i}$  : pengamatan ke- $t - i$  ( $i = 1, 2, 3, \dots, n$ )  
 $p$  : jumlah neuron *input layer*  
 $q$  : jumlah neuron *hidden layer*  
 $\varepsilon_t$  : sisaan pada waktu ke- $t$



Pemrosesan informasi pada ANN terjadi pada neuron. Sinyal-sinyal informasi disampaikan melalui penghubung dari satu neuron ke neuron lainnya melalui fungsi aktivasi tertentu. Keluaran yang didapatkan sesuai dengan fungsi aktivasi yang digunakan. ANN dibangun atas tiga hal, yaitu:

1. Pola hubungan antar neuron (arsitektur jaringan)
2. Metode penentuan pembobot pada hubungan-hubungan (pelatihan, pembelajaran, dan algoritma).
3. Fungsi aktivasi

## 2.7 Arsitektur Jaringan ANN

*Artificial neural networks* memiliki beberapa pola hubungan neuron yang dibedakan menjadi tiga macam arsitektur adalah sebagai berikut (Fausett 1994):

1. Jaringan lapisan tunggal (*single layer networks*)

Jaringan lapisan tunggal tersusun dari lapisan masukan (*input layer*) dan lapisan keluaran (*output layer*) tanpa adanya suatu lapisan tersembunyi (*hidden layer*). Lapisan masukan setelah menerima semua masukan yang diterimanya akan langsung diproses untuk menghasilkan suatu keluaran melalui bobot-bobot penghubung antara lapisan masukan dan lapisan keluaran.

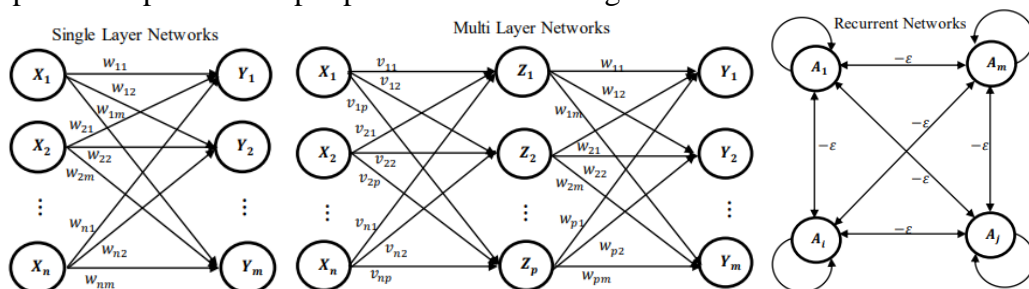
2. Jaringan lapisan ganda (*multi layer networks*)

Jaringan ini merupakan perluasan dari jaringan lapisan tunggal. Jaringan lapisan ganda tersusun atas tiga lapisan, yaitu *input layer*, *hidden layer*, dan *output layer*. Pemrosesan data dimulai dari *input layer* menuju *hidden layer* dan disampaikan ke *output layer*. Jaringan lapisan ganda digunakan untuk permasalahan yang kompleks dibandingkan dengan jaringan lapisan tunggal dan proses pelatihan terkadang membutuhkan waktu yang lebih lama dan rumit (Fausett 1994).

3. Jaringan lapisan kompetitif (*recurrent networks*)

Jaringan lapisan kompetitif memiliki lapisan yang berbeda dengan jaringan lapisan tunggal dan ganda. Jaringan ini antara neuron satu dengan yang lainnya saling berhubungan satu sama lain dan memiliki bobot yang telah ditetapkan. Nilai bobot untuk tiap neuron adalah satu dan bobot acak negative dengan bobot  $-\varepsilon$  untuk neuron lain. Lapisan ini mempunyai *feedback loop* atau neuron keluaran yang memberikan sinyal pada neuron masukan, dengan kata lain sinyal mengalir dua arah.

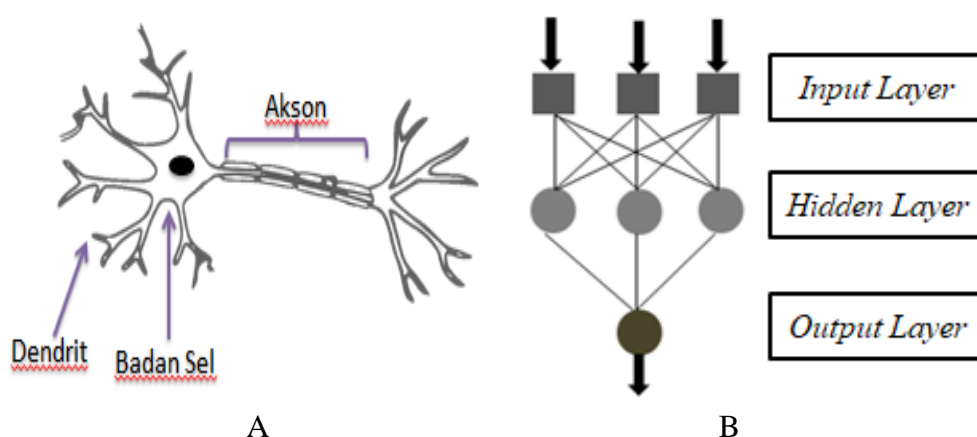
Arsitektur jaringan lapisan tunggal, jaringan lapisan ganda, dan jaringan lapisan kompetitif terdapat pada Gambar 1 sebagai berikut:



Gambar 1 Tipe-tipe arsitektur jaringan ANN

## 2.8 Metode *Backpropagation*

Metode *backpropagation* adalah metode yang dapat menurunkan gradien untuk meminimalkan jumlah kesalahan kuadrat dari keluaran jaringan. Selain itu, metode ini sangat baik dalam menangani masalah pengenalan pola-pola yang kompleks. Metode *backpropagation* termasuk kedalam teknik pembelajaran terawasi (*supervised learning*). Pembelajaran terawasi merupakan pembelajaran yang nilai keluarannya sudah diketahui. Perbedaan antara keluaran data aktual dengan data keluaran yang diinginkan digunakan untuk mengoreksi bobot pada jaringan ANN, agar dapat menghasilkan jaringan yang mempunyai nilai sedekat mungkin dengan data aktualnya yang telah diketahui oleh ANN (Wuryandari dan Afrianto 2012).



Gambar 2 Jaringan syaraf manusia (A) dan skema dari ANN untuk satu *output* (B)

Gambar 2 menunjukkan bahwa jaringan saraf manusia dan arsitektur ANN. ANN terdiri atas tiga lapisan yaitu :

- Input layer* berfungsi menerima sinyal *input* atau informasi data.
- Hidden layer* berfungsi mengumpulkan bobot dan mengakumulasikannya, sebagai penghubung antara *input layer* dan *output layer*, sebagai peningkat kemampuan jaringan dalam memecahkan permasalahan.
- Output layer* berfungsi menghasilkan keluaran dari proses ANN.

Salah satu model ANN yang paling sering digunakan adalah jaringan lapis ganda dengan metode pembelajaran propagasi balik (*backpropagation*). Parameter model akan disesuaikan secara berulang dengan proses meminimalkan kesalahan peramalan. Algoritma *backpropagation* terdiri dari beberapa tahapan proses belajar yang dilakukan secara iteratif, yaitu (Fausett 1994):

1. Inisialisasi bobot secara acak (*random*).

### ***Feedforward (umpan maju)***

2. Masing-masing neuron *input*  $y_i$  ( $i = 1, 2, 3, \dots, n$ ) menerima sinyal *input*  $y_i$  dan menyebarkan ke semua neuron pada lapisan di depannya. Dalam penelitian ini  $y_i$  merupakan jumlah positif Covid-19 di DKI Jakarta yang sudah dinormalisasi.



3. Pada masing-masing neuron *hidden layer*, jumlahkan dengan:

$$z\_in_j = \sum_{i=1}^n y_i v_{ij} + v_{oj} \quad (16)$$

Keterangan:

- $y_i$  : jumlah positif Covid-19 di DKI Jakarta yang sudah dinormalisasi ke- $i$  ( $i = 1, 2, 3, \dots, n$ )  
 $v_{ij}$  : bobot antara neuron *input* ke- $i$  dengan neuron *hidden* ke- $j$   
 $v_{oj}$  : bobot bias (parameter yang berperan sebagai *intercept*) neuron *input* pada neuron *hidden* ke- $j$   
 $n$  : jumlah neuron *input*  
 $z\_in_j$  : nilai neuron *hidden* ke- $i$  dari *input* ke- $j$

Hitung fungsi aktivasi  $z_j = f(z\_in_j) = \frac{1}{1+e^{-z\_in_j}}$  dan sebar ke semua neuron pada lapisan di depannya.

4. Pada masing-masing neuron *output layer*, jumlahkan dengan:

$$y\_in_k = \sum_{j=1}^p z_j w_{jk} + w_{ok} \quad (17)$$

Keterangan:

- $w_{jk}$  : bobot neuron *hidden* ke- $j$  ( $j = 1, 2, \dots, p$ ) pada neuron *output* ke- $k$  ( $k = 1, 2, \dots, m$ )  
 $w_{ok}$  : bobot bias neuron *hidden* pada neuron *output* ke- $k$   
 $z_j$  : nilai neuron *hidden* ke- $j$  ( $j = 1, 2, \dots, p$ )  
 $y\_in_k$  : nilai *output* ke- $k$  pada *output layer* dari neuron *hidden layer*

Hitung fungsi aktifasi:  $Y_k = f(y\_in_k) = \frac{1}{1+e^{-y\_in_k}}$

### Backpropagation of error

5. Hitung galat ( $\delta$ ) pada masing-masing neuron *output layer*:

$$\delta_k = (t_k - Y_k) f'(y\_in_k), \quad (18)$$

$$f'(y\_in_k) = f(y\_in_k) [1 - f(y\_in_k)] \quad (19)$$

Keterangan:

- $\delta_k$  : kesalahan  
 $y\_in_k$  : nilai *output* ke- $k$  pada *output layer* dari neuron *hidden layer*  
 $t_k$  : nilai data aktual pada waktu ke- $t$   
 $Y_k$  : nilai *output* jaringan pada waktu ke- $t$

Hitung kenaikan nilai bobot:

$$\Delta w_{jk} = \alpha \delta_k z_j \quad (20)$$

Keterangan:

- $\Delta w_{jk}$  : perubahan bobot antara neuron *hidden layer* ke- $j$  dengan neuron *output layer*  
 $\alpha$  : koefisien pembelajaran (*learning rate*)  
 $\delta_k$  : kesalahan  
 $z_j$  : nilai neuron *hidden layer* ke- $j$  ( $j = 1, 2, \dots, p$ )

Hitung kenaikan nilai bobot neuron bias:

$$\Delta w_{ok} = \alpha \delta_k \quad (21)$$

Keterangan:

$\Delta w_{ok}$  : perubahan bobot bias

$\alpha$  : koefisien pembelajaran (*learning rate*)

$\delta_k$  : kesalahan

6. Hitung galat ( $\delta$ ) pada masing-masing neuron *hidden layer*:

$$\delta_j = \delta_{in_j} f'(z_{in_j}) = \sum_{k=1}^m \delta_k w_{jk} f'(z_{in_j}) \quad (22)$$

$$f'(z_{in_j}) = f(z_{in_j})[1 - f(z_{in_j})] \quad (23)$$

Keterangan:

$\delta_j$  : kesalahan pada neuron *hidden* ke-j

$z_{in_j}$  : nilai neuron *hidden* ke-i dari *input* ke-j

$\delta_{in_j}$  : penjumlahan *input* delta

$w_{jk}$  : bobot neuron *hidden* ke-j ( $j = 1, 2, \dots, p$ ) pada neuron *output* ke-k ( $k = 1, 2, \dots, m$ )

Hitung kenaikan nilai bobot:

$$\Delta v_{ij} = \alpha \delta_j y_i \quad (24)$$

Keterangan:

$\Delta v_{ij}$  : perubahan nilai bobot penghubung antara neuron *input layer* ke-i dengan neuron *hidden layer* ke-j

$\alpha$  : koefisien pembelajaran (*learning rate*)

$\delta_j$  : kesalahan pada neuron *hidden* ke-j

$y_i$  : jumlah positif Covid-19 di DKI Jakarta yang sudah di normalisasi ke-i ( $i = 1, 2, \dots, n$ )

Hitung kenaikan nilai bobot neuron bias:

$$\Delta v_{oj} = \alpha \delta_j \quad (25)$$

dengan  $\alpha$  adalah koefisien pembelajaran (*learning rate*)

Keterangan:

$\Delta v_{oj}$  : perubahan bobot bias

$\alpha$  : koefisien pembelajaran (*learning rate*)

$\delta_j$  : kesalahan pada neuron *hidden* ke-j

### Perubahan bobot

7. Lakukan updating semua bobot antara *hidden layer* dan *output layer* :

$$w_{jk}(\text{new}) = w_{jk}(\text{old}) + \Delta w_{jk} \quad (26)$$

$$w_{ok}(\text{new}) = w_{ok}(\text{old}) + \Delta w_{ok} \quad (27)$$

Lakukan updating semua bobot antara *input layer* dan *hidden layer* :

$$v_{ij}(\text{new}) = v_{ij}(\text{old}) + \Delta v_{ij} \quad (28)$$

$$v_{oj}(\text{new}) = v_{oj}(\text{old}) + \Delta v_{oj} \quad (29)$$



Keterangan:

$w_{jk}(new)$  : nilai bobot baru yang menghubungkan neuron *hidden layer* ke- $j$  dengan neuron pada *output layer*

$w_{jk}(old)$  : nilai bobot lama yang menghubungkan neuron *hidden layer* ke- $j$  dengan neuron *output layer*

$\Delta w_{jk}$  : perubahan bobot antara neuron *hidden layer* ke- $j$  dengan neuron *output layer*

$w_{ok}(new)$  : nilai bobot bias baru pada *output layer*

$w_{ok}(old)$  : nilai bobot bias lama pada *output layer*

$\Delta w_{ok}$  : perubahan bobot bias

$v_{ij}(new)$  : nilai bobot baru yang menghubungkan neuron *input layer* ke- $i$  dengan neuron pada *hidden layer* ke- $j$

$v_{ij}(old)$  : nilai bobot lama yang menghubungkan neuron *input layer* ke- $i$  dengan neuron pada *hidden layer* ke- $j$

$\Delta v_{ij}$  : perubahan nilai bobot penghubung antara neuron *input layer* ke- $i$  dengan neuron *hidden layer* ke- $j$

$v_{oj}(new)$  : nilai bobot bias baru pada *hidden layer*

$v_{oj}(old)$  : nilai bobot bias lama pada *hidden layer*

$\Delta v_{oj}$  : perubahan bobot bias

8. Kondisi berhenti jika sesuai dengan jumlah siklus perubahan bobot (*epoch*) saat pelatihan sudah mencapai maksimum atau kesalahan yang dihasilkan lebih kecil dari pada kesalahan yang ditentukan.

## 2.9 Validasi Model

Validasi dilakukan untuk mengetahui model yang digunakan untuk peramalan sudah baik atau tidak. Semakin kecil nilai *Root Mean Square Error* (RMSE) dan *Mean Absolute Percentage Error* (MAPE) menunjukkan bahwa model yang digunakan semakin baik untuk peramalan. Nilai MAPE dan RMSE dapat dihitung dengan rumus sebagai berikut (Montgomery *et al.* 2015):

$$RMSE = \sqrt{\frac{\sum_{t=1}^n (y_t - \hat{y}_t)^2}{n}} \quad (30)$$

$$MAPE = \frac{\sum_{t=1}^n |(y_t - \hat{y}_t)/y_t|}{n} \quad (31)$$

Keterangan:

$y_t$  : nilai pengamatan pada waktu ke- $t$

$\hat{y}_t$  : nilai dugaan pada waktu ke- $t$

$t$  : 1,2,3,4, ... ...,  $n$ .

$n$  : banyaknya pengamatan

Chang *et al.* (2007) menyatakan untuk nilai MAPE terdapat selang nilai yang dapat dijadikan bahan pengukuran mengenai kebaikan model dalam melakukan peramalan. Tabel 2 memperlihatkan selang nilai MAPE adalah sebagai berikut:

Tabel 2 Selang nilai MAPE

MAPE	Signifikan
<10%	Kemampuan peramalan yang sangat baik
10-20%	Kemampuan peramalan yang baik
20-50%	Kemampuan peramalan yang layak
>50%	Kemampuan peramalan yang buruk

- Hak Cipta Dilindungi Undang-undang
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
    - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah
    - b. Pengutipan tidak merugikan kepentingan yang wajar IPB University.
  2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IPB University.

### III METODE

#### 3.1 Data

Data yang digunakan pada penelitian ini adalah data sekunder. Data tersebut merupakan data harian jumlah positif Covid-19 di DKI Jakarta periode 2 Maret 2020 hingga 31 Desember 2020 sebanyak 305 data pengamatan. Data dibagi menjadi data latih dan data uji. Data latih dari 2 Maret 2020 hingga 31 Oktober 2020, sedangkan data uji dari 1 November 2020 hingga 31 Desember 2020.

#### 3.2 Prosedur Analisis Data

Tahapan analisis yang dilakukan pada penelitian ini adalah sebagai berikut:

1. Melakukan eksplorasi data untuk mengetahui gambaran dan mengidentifikasi pola data dengan cara melihat plot.
2. Membagi data menjadi dua bagian, yaitu data latih dan data uji. Adayinka dan Muhajirine (2020), membagi data latih sebanyak 80% dan data uji sebanyak 20% berdasarkan pada prinsip Pareto.
3. Metode ARIMA
  - a. Pemeriksaan kestasioneran data dalam ragam dan rata-rata.
  - b. Mengidentifikasi model ARIMA melalui plot ACF dan PACF.
  - c. Melakukan pendugaan parameter dan pemeriksaan signifikan parameter dari model tentatif yang diperoleh pada tahap identifikasi model.
  - d. Melakukan diagnostik model untuk mengetahui kelayakan model, melalui pemeriksaan kebebasan sisaan dan kenormalan sisaan.
  - e. Melakukan *overfitting* untuk dibandingkan dengan model tentatif awal. Model yang didapat pada *overfitting* akan dilakukan uji signifikan parameter dan periksa asumsi seperti pada tahap diagnostik model.
  - f. Memilih model terbaik dari model-model ARIMA yang sudah terbentuk dengan cara memilih model yang memiliki nilai AIC dan BIC terkecil.
  - g. Melakukan peramalan sebanyak data uji untuk menghitung nilai RMSE dan MAPE.
4. Metode *Artificial Neural Networks*
  - a. Menentukan *input* dan *output layer*. *Input layer* ditentukan berdasarkan *trail and error*, yaitu jumlah positif Covid-19 di DKI Jakarta beberapa hari sebelumnya, sedangkan *output layer* adalah jumlah Covid-19 di DKI Jakarta periode ke-*t*. Koefisien pembelajaran (*learning rate*) sebesar 0,01.
  - b. Melakukan normalisasi data, sehingga data *input* dan *output* berada pada selang nilai dari 0 sampai 1. Normalisasi data dilakukan dengan rumus sebagai berikut:

$$y'_t = \frac{y_t - y_{\min}}{y_{\max} - y_{\min}} \quad (32)$$

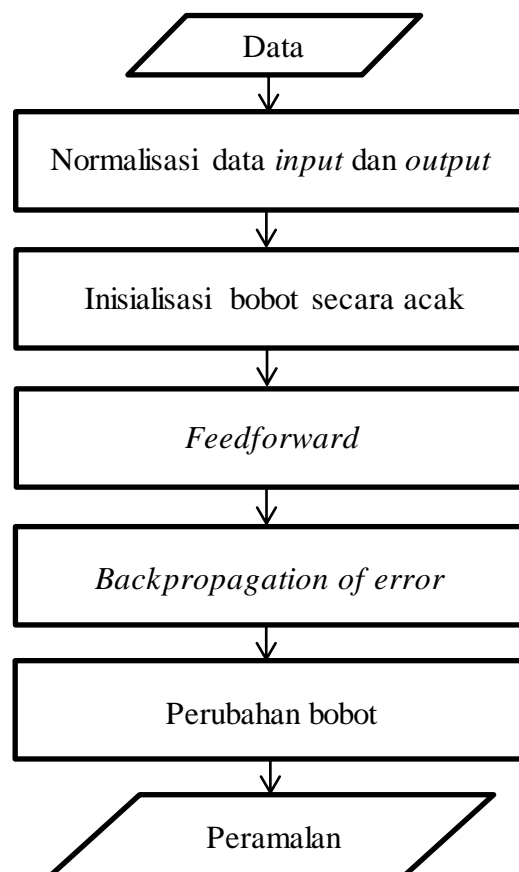
Keterangan:

$y'_t$  : jumlah Covid-19 pada waktu ke- $t$  yang telah dinormalisasi  
 $y_t$  : jumlah Covid-19 pada waktu ke- $t$   
 $y_{maks}$  : jumlah Covid-19 tertinggi  
 $y_{min}$  : jumlah Covid-19 terendah

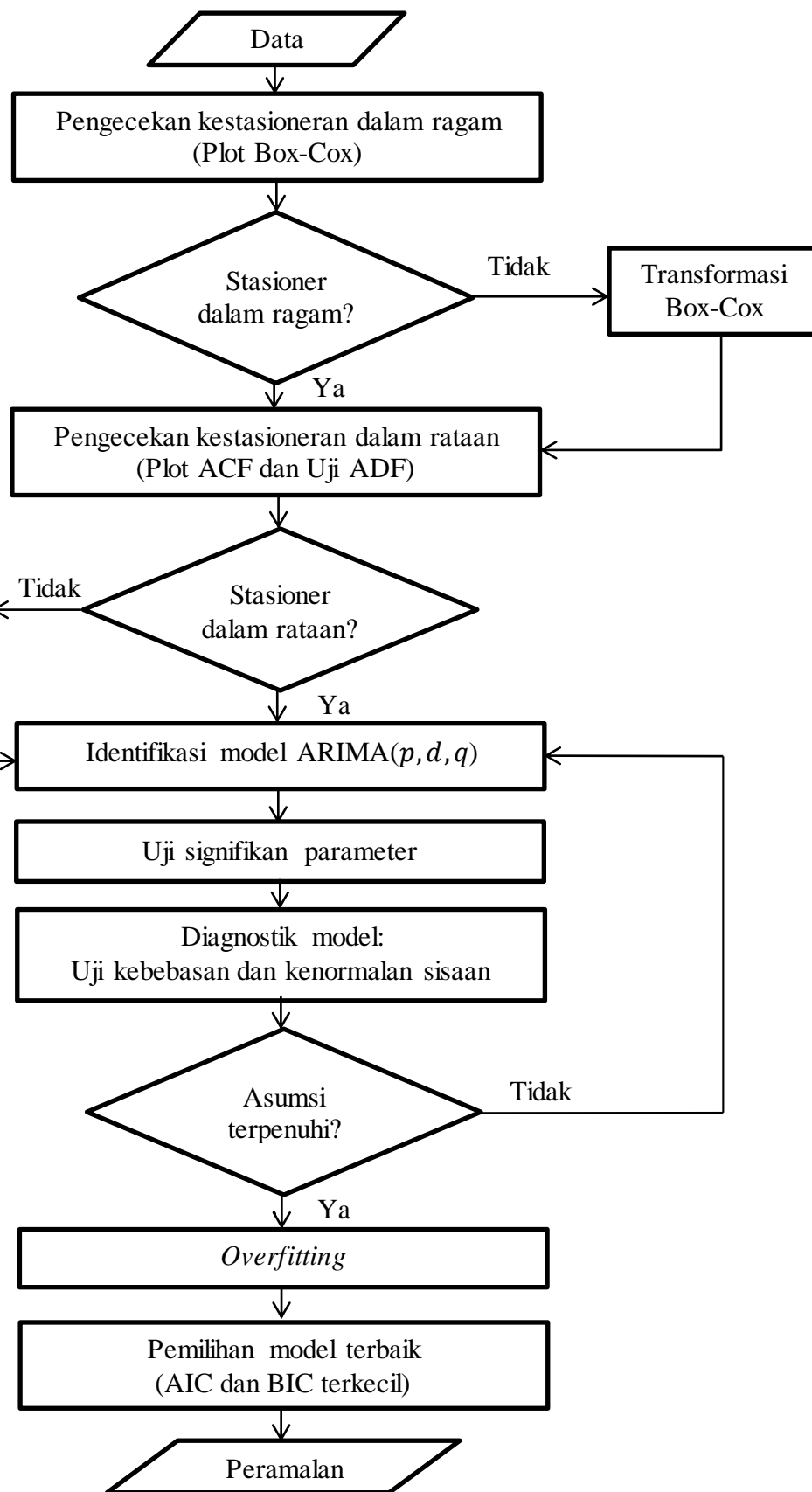
- c. Membangun arsitektur ANN dari neuron *input* dan *output*, dengan terlebih dahulu menentukan banyaknya *hidden* dan jumlah neuron pada *hidden layer*. Neuron yang sudah ditentukan pada masing-masing lapisan akan diterapkan algoritma *backpropagation* secara berulang sampai didapatkan model yang diinginkan.
- d. Melakukan proses validasi dengan menggunakan data uji setelah terbentuk model akhir untuk mendapatkan nilai RMSE dan MAPE.
5. Membandingkan hasil peramalan metode ARIMA dan ANN dengan melihat nilai RMSE dan MAPE terkecil serta hasil peramalan dari jumlah positif Covid-19 di DKI Jakarta.

### 3.3 Diagram Analisis Data

Diagram alir pada Gambar 3 dan 4 merupakan alur dari pelaksanaan penelitian ini dengan menggunakan metode *Artificial Neural Networks* (ANN) dan *Autoregressive Integrated Moving Average* (ARIMA).



Gambar 3 Diagram alir metode ANN

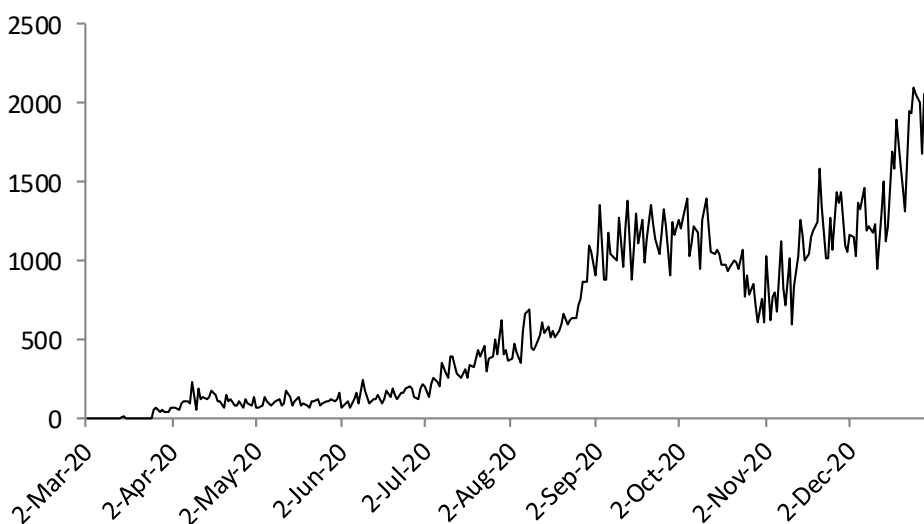


Gambar 4 Diagram alir metode ARIMA

## IV HASIL DAN PEMBAHASAN

### 4.1 Eksplorasi Data

Eksplorasi data dilakukan untuk melihat gambaran dan mengidentifikasi data jumlah positif Covid-19 di DKI Jakarta yang disajikan dalam periode harian sebanyak 305 data dari 2 Maret 2020–31 Desember 2020. Selain itu, eksplorasi data dilakukan dengan membuat grafik data deret waktu jumlah positif Covid-19 di DKI Jakarta untuk melihat fluktuasi data. Sumbu x menunjukkan periode data dan sumbu y menunjukkan jumlah harian positif Covid-19 di DKI Jakarta. Gambar 5 memperlihatkan plot data deret waktu jumlah harian positif Covid-19 yang berfluktuasi dan cenderung memiliki *trend* naik. Data terlihat tidak stasioner baik dalam ragam maupun rata-ran. Simpangan baku jumlah positif Covid-19 di DKI Jakarta sebesar 541,16. Hal tersebut menunjukkan bahwa data jumlah positif Covid-19 di DKI Jakarta cukup beragam dari waktu ke waktu. Total jumlah positif Covid-19 periode 2 Maret 2020–31 Desember 2020 adalah 183.246 kasus. Jumlah positif Covid-19 tertinggi terjadi pada 25 Desember 2020 sebanyak 2096 kasus. Hal tersebut terjadi bertepatan dengan adanya perayaan hari Natal, besarnya kenaikan kasus sebanyak 163 kasus dibandingkan tanggal 24 Desember 2020. Jumlah kasus terendah terjadi pada awal Maret 2020 sebanyak nol kasus. Rata-rata harian jumlah positif Covid-19 di DKI Jakarta periode 2 Maret 2020–31 Desember 2020 adalah 601 kasus. Rata-rata mingguan dan bulanan jumlah positif Covid-19 di DKI Jakarta periode tersebut masing-masing, yaitu 4385 dan 18.325 kasus.



Gambar 5 Plot data jumlah harian positif Covid-19 di DKI Jakarta

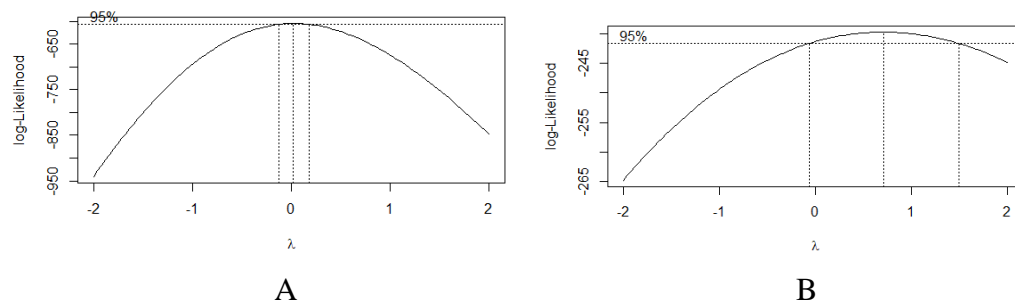
Lampiran 1 menunjukkan jumlah positif Covid-19 di DKI Jakarta mengalami lonjakan tertinggi antara bulan November sampai Desember 2020, sedangkan Lampiran 2 menunjukkan lonjakan tertinggi untuk penambahan kasus perminggunya terjadi pada minggu ke-2 sampai minggu ke-4 di bulan Desember. Hal ini dapat disebabkan liburan natal dan tahun baru, beberapa masyarakat masih memilih pergi liburan di tengah pandemi dengan melonggarkan protokol



kesehatan, sehingga lonjakan jumlah kasus positif Covid-19 semakin meningkat. Jumlah positif Covid-19 di DKI Jakarta setiap harinya bertambah dan belum menunjukkan adanya penurunan grafik yang signifikan.

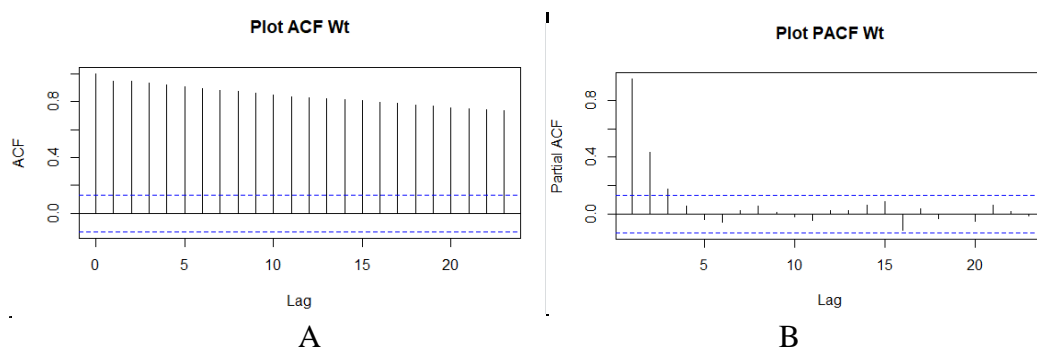
#### 4.2 Kestasioneran Data

Data jumlah positif Covid-19 di DKI Jakarta perlu dilakukan pemeriksaan kestasioneran ragam dan rataaan. Kestasioneran ragam menggunakan plot Box-Cox dan kestasioneran rataaan menggunakan plot ACF dan Uji *Augmented Dickey Fuller* (ADF).



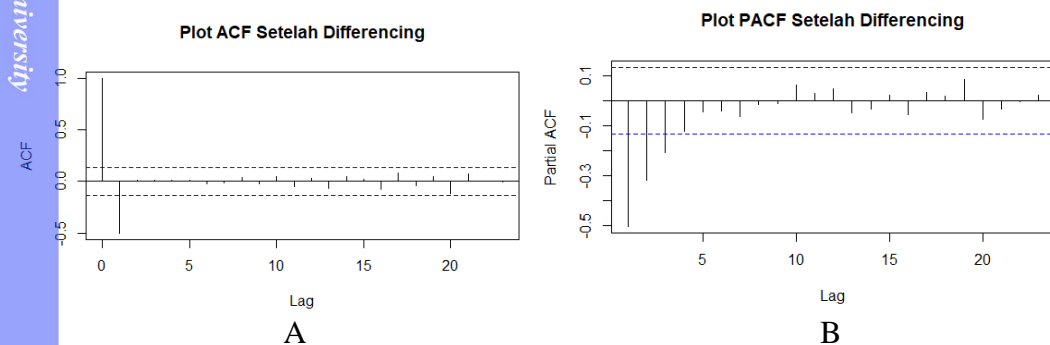
Gambar 6 Plot Box-Cox jumlah positif Covid-19 di DKI Jakarta yang belum stasioner (A) dan sudah stasioner (B) dalam ragam

Gambar 6 menunjukkan hasil pemeriksaan kestasioneran dalam ragam. Gambar 6(A) menghasilkan *rounded value* ( $\lambda$ ) sebesar 0,0223, yang menunjukkan data tersebut belum stasioner dalam ragam karena nilai  $\lambda$  yang dihasilkan tidak mendekati satu atau selang nilai  $\lambda$  tidak mencakup nilai satu, maka dilakukan transformasi Box-Cox sesuai dengan nilai  $\lambda$  yang diperoleh. Setelah dilakukan transformasi Box-Cox seperti pada Gambar 6(B),  $\lambda$  yang dihasilkan mendekati satu atau selang nilai  $\lambda$  mencakup nilai satu, artinya data sudah stasioner dalam ragam. Data yang sudah stasioner dalam ragam akan dilakukan uji kestasioneran rataaan. Uji kestasioneran rataaan paling mudah dilihat dari grafik ACF dan PACF.  $W_t$  merupakan data hasil transformasi. Gambar 7 menunjukkan ketidakstasioneran data dalam rataaan terlihat dari plot ACF yang memiliki pola menurun secara perlahan.



Gambar 7 Plot ACF (A) dan PACF (B) data jumlah positif Covid-19 di DKI Jakarta yang belum stasioner dalam rataaan

Pemeriksaan terhadap kestasioneran data dalam rataa juga dapat dilakukan dengan uji *Augmented Dickey Fuller* (ADF). Uji ADF diperoleh *p-value* sebesar 0,9232. Hasil tersebut menunjukkan bahwa data belum stasioner dalam rataa karena dihasilkan *p-value* yang lebih besar dari nilai taraf nyata 5%. Data deret waktu yang tidak stasioner dalam rataa dapat diatasi dengan melakukan pembedaan. Proses pembedaan dilakukan sampai data deret waktu menjadi stasioner. Pembedaan dilakukan satu kali pada data hasil transformasi, kemudian dilakukan pemeriksaan kestasioneran kembali dengan melihat plot ACF dan PACF. Gambar 8 menunjukkan bahwa data deret waktu jumlah positif Covid-19 di DKI Jakarta telah stasioner dalam rataa ditunjukkan dari plot ACF yang menurun secara cepat mendekati nol setelah *lag* satu. Plot ACF dan PACF tidak menunjukkan pola tertentu dan tidak terdapat pola musiman pada plot tersebut.



Gambar 8 Plot ACF (A) dan PACF (B) data jumlah positif Covid-19 di DKI Jakarta yang sudah stasioner dalam rataa

Kestasioneran data yang telah dilakukan pembedaan juga dilihat dari uji ADF. Hasil uji ADF diperoleh *p-value* sebesar 0,01 lebih kecil dari nilai taraf nyata 5%, sehingga dapat disimpulkan bahwa data jumlah positif Covid-19 di DKI Jakarta sudah dalam keadaan stasioner dalam rataa. Data yang sudah stasioner dalam ragam dan rataa akan dilakukan identifikasi dan pendugaan parameter model tentatif ARIMA.

### 4.3 Identifikasi Model dan Pendugaan Parameter

Identifikasi model dapat dilakukan dengan melihat plot ACF dan PACF. Gambar 8 memperlihatkan plot ACF memiliki pola *cuts off* setelah *lag* satu dan PACF memiliki pola *tails off*. Hal ini menunjukkan adanya kemungkinan bahwa jumlah positif Covid-19 di DKI Jakarta dipengaruhi oleh data satu hari sebelumnya. Model tentatif awal yang mungkin berdasarkan plot ACF dan PACF adalah ARIMA(0,1,1), ARIMA(1,1,0), ARIMA(2,1,0), dan ARIMA(3,1,0). Hasil pendugaan parameter dari kandidat model yang didapat pada tahap identifikasi model disajikan di Tabel 3.



Tabel 3 Nilai dugaan parameter model ARIMA untuk jumlah positif Covid-19 di DKI Jakarta

Model	Parameter	Koefisien parameter	<i>P-value</i>	AIC	BIC
ARIMA(0,1,1)	MA(1) <sup>*</sup>	-0,6721	<0,0001	66,84	73,610
ARIMA(1,1,0)	AR(1) <sup>*</sup>	-0,4994	<0,0001	97,67	104,45
ARIMA(2,1,0)	AR(1) <sup>*</sup>	-0,6550	<0,0001	77,29	87,460
	AR(2) <sup>*</sup>	-0,3124	<0,0001		
ARIMA(3,1,0)	AR(1) <sup>*</sup>	-0,7158	<0,0001	70,84	84,390
	AR(2) <sup>*</sup>	-0,4406	<0,0001		
	AR(3) <sup>*</sup>	-0,1950	0,0033		

\*Parameter yang signifikan pada taraf nyata 5%

Berdasarkan Tabel 3 semua kandidat model memiliki nilai penduga parameter yang signifikan pada taraf nyata 5%, yaitu ARIMA(0,1,1), ARIMA(1,1,0), ARIMA(2,1,0), dan ARIMA(3,1,0) karena dihasilkan nilai peluang yang lebih kecil dari taraf nyata 5%. Model ARIMA(0,1,1) memiliki nilai AIC dan BIC terkecil, sedangkan ARIMA(1,1,0) memiliki nilai AIC dan BIC terbesar dibandingkan model tentatif lainnya. Empat model yang signifikan akan dilakukan diagnostik model untuk memeriksa kelayakan model.

#### 4.4 Diagnostik Model

Model yang signifikan pada tahap identifikasi model, akan dilakukan diagnostik model menggunakan uji Ljung–Box dan Kolmogorov–Smirnov untuk memeriksa kelayakan model. Uji Ljung–Box digunakan untuk menguji apakah terdapat autokorelasi sisaan. Kenormalan sisaan dilakukan dengan plot Q-Q, histogram, dan uji Kolmogorov–Smirnov. Hasil uji Ljung–Box untuk *lag* 5, 10, 15, 20, 25, dan 30 dari keempat model yang signifikan pada taraf nyata 5% disajikan di Tabel 4.

Tabel 4 Hasil uji Ljung-Box sisaan pada model ARIMA

Model	<i>P-value</i>	<i>Lag</i>
ARIMA (0,1,1)	0,8318	5
	0,9709	10
	0,9928	15
	0,9779	20
	0,9961	25
	0,9953	30
ARIMA (1,1,0)	0,00005 <sup>*</sup>	5
	0,0016 <sup>*</sup>	10
	0,0119 <sup>*</sup>	15
	0,0207 <sup>*</sup>	20
	0,0557	25
	0,0505	30

ARIMA (2,1,0)	0,0435*	5
	0,2047	10
	0,4261	15
	0,4152	20
	0,5908	25
	0,4574	30
ARIMA (3,1,0)	0,5705	5
	0,8479	10
	0,9298	15
	0,9106	20
	0,9765	25
	0,9544	30

\*Parameter yang signifikan pada taraf nyata 5%

Tabel 4 menunjukkan bahwa model ARIMA(1,1,0) dan ARIMA(2,1,0) mempunyai autokorelasi pada sisaan karena beberapa *lag* memiliki *p-value* yang lebih kecil dari pada taraf nyata 5%. Sedangkan, model ARIMA(0,1,1) dan ARIMA(3,1,0) memiliki *p-value* yang lebih besar dari taraf nyata 5%, sehingga model tersebut tidak memiliki autokorelasi pada sisaannya. Selain itu, pemeriksaan sisaan dapat dilihat dari plot *Residuals Autocorrelation Function* (RACF) dan *Residuals Partial Autocorrelation Function* (RPACF). Grafik RACF dan RPACF pada Lampiran 3 dan 4 untuk model ARIMA(0,1,1) dan ARIMA(3,1,0) menunjukkan tidak ada nilai korelasi diri dan nilai korelasi diri parsial sisaan yang melebihi garis batas signifikan, sehingga bisa dikatakan bahwa tidak terdapat autokorelasi pada sisaan yang berarti model ARIMA(0,1,1) dan ARIMA(3,1,0) memenuhi asumsi kebebasan sisaan. Setelah dilakukan pengujian kebebasan sisaan, selanjutnya dilakukan pemeriksaan kenormalan sisaan yang disajikan di Tabel 5.

Tabel 5 Pemeriksaan kenormalan sisaan

Model	<i>P-value</i>	Keterangan
ARIMA(0,1,1)	0,1280	Sisaan menyebar normal
ARIMA(3,1,0)	0,2496	Sisaan menyebar normal

Tabel 5 menunjukkan tak tolak hipotesis nol karena *p-value* yang dihasilkan lebih besar dari taraf nyata 5% yang berarti bahwa kedua model memenuhi asumsi kenormalan sisaan. Selain itu, kenormalan sisaan dapat dilakukan dengan melihat plot sebaran normal sisaan. Lampiran 5 dan 6 menunjukkan bahwa plot sebaran normal sisaan model ARIMA(0,1,1) dan ARIMA(3,1,0) mendekati normal. Model tentatif terbaik merupakan model yang memiliki nilai AIC dan BIC terkecil. Berdasarkan Tabel 6, diperoleh model ARIMA(0,1,1) merupakan model yang memiliki nilai AIC dan BIC terkecil dibandingkan model ARIMA(3,1,0) dengan selisih nilai AIC dan BIC berturut-turut sebesar 4,00 dan 10,78.

Tabel 6 Nilai AIC dan BIC dari model tentatif awal

Model	AIC	BIC
ARIMA(0,1,1)	66,84	73,61
ARIMA(3,1,0)	70,84	84,39

Model terbaik ARIMA(0,1,1) akan dilakukan *overfitting* dengan cara menambah ordo pada model tentatif terbaik. Hasil *overfitting* beberapa kandidat model ARIMA dapat dilihat pada Tabel 7.

Tabel 7 Nilai dugaan parameter model ARIMA untuk jumlah positif Covid-19 di DKI Jakarta hasil *overfitting*

Model	Parameter	Koefisien parameter	P-value	AIC	BIC
ARIMA(0,1,1) <i>with drift</i>	MA(1)*	-0,7069	<0,0001	63,30	73,47
	Konstanta*	0,0139	0,0118		
ARIMA(0,1,2)	MA(1)*	-0,7460	<0,0001	66,94	77,11
	MA(2)	0,0940	0,1672		
ARIMA(1,1,1)	AR(1)	-0,1149	0,2048	67,26	77,43
	MA(1)*	-0,6186	<0,0001		
ARIMA(1,1,2)	AR(1)*	0,8716	<0,0001	65,98	79,54
	MA(1)*	-1,6164	<0,0001		
	MA(2)*	0,6812	<0,0001		
ARIMA(1,1,3)	AR(1)*	0,9738	<0,0001	67,30	84,25
	MA(1)*	-1,7290	<0,0001		
	MA(2)*	0,7951	<0,0001		
	MA(3)	-0,0467	0,5124		

\*Parameter yang signifikan pada taraf nyata 5%

Tabel 7 memperlihatkan bahwa model dengan koefisien parameter yang signifikan pada taraf nyata 5%, yaitu ARIMA(0,1,1) *with drift* dan ARIMA(1,1,2), karena memiliki nilai peluang yang lebih kecil dari taraf nyata 5%. Model ARIMA(0,1,2), ARIMA(1,1,1), dan ARIMA(1,1,3) memiliki nilai peluang yang lebih besar dari taraf nyata 5% pada parameter AR(1), MA(2), dan MA(3), sehingga model tersebut tidak signifikan. Model ARIMA(0,1,1) *with drift* memiliki nilai AIC dan BIC terkecil dibandingkan model yang lainnya. Selanjutnya, akan dilakukan diagnostik model. Hasil uji Ljung-Box yang dihasilkan dari model ARIMA(0,1,1) *with drift* dan ARIMA(1,1,2) yang signifikan tersebut disajikan pada Tabel 8.

Tabel 8 Hasil uji Ljung-Box sisaan pada model ARIMA hasil *overfitting*

Model	P-value	Lag
ARIMA (0,1,1) <i>with drift</i>	0,9077	5
	0,9791	10
	0,9959	15
	0,9854	20
	0,9978	25
	0,9972	30
ARIMA (1,1,2)	0,9996	5
	0,9994	10
	0,9979	15
	0,9877	20
	0,9984	25
	0,9975	30

\*Parameter yang signifikan pada taraf nyata 5%

Tabel 8 menunjukkan bahwa model ARIMA(0,1,1) *with drift* dan ARIMA(1,1,2) memiliki nilai-nilai yang tidak signifikan pada taraf nyata 5%. Hal tersebut berarti kedua model tidak memiliki autokorelasi antar sisaan. Grafik RACF dan RPACF pada Lampiran 7 dan 8, menunjukkan tidak nyata pada semua *lag*, sehingga bisa disimpulkan bahwa model ARIMA(0,1,1) *with drift* dan ARIMA(1,1,2) tidak memiliki autokorelasi pada sisaan. Pengujian kenormalan sisaan menunjukkan *p-value* sebesar 0,2274 pada ARIMA(0,1,1) *with drift* dan 0,0946 pada ARIMA(1,1,2) yang nilai peluangnya lebih besar dari taraf nyata 5%, berarti sisaan menyebar normal. Selain itu, kenormalan sisaan dapat dilakukan dengan melihat plot sebaran normal sisaan. Berdasarkan Lampiran 9 dan 10, plot sebaran normal sisaan model ARIMA(0,1,1) *with drift* dan ARIMA(1,1,2) menunjukkan bahwa sebaran sisaan mendekati normal.

Berdasarkan kandidat model ARIMA yang terbentuk, ARIMA(0,1,1), ARIMA(0,1,1) *with drift*, dan ARIMA(1,1,2) merupakan model yang signifikan pada taraf nyata 5 %, memenuhi asumsi kebebasan sisaan, dan kenormalan sisaan. Model ARIMA yang terbentuk pada tahap identifikasi model akan dibandingkan dengan model hasil *overfitting* berdasarkan pada nilai AIC dan BIC terkecil diantara model-model tersebut. Model terbaik akan digunakan untuk peramalan jumlah positif Covid-19 di DKI Jakarta. Tabel 9 memperlihatkan AIC dan BIC yang dimiliki model ARIMA(0,1,1), ARIMA(0,1,1) *with drift*, dan ARIMA(1,1,2).

Tabel 9 Perbandingan nilai AIC dan BIC terkecil pada model ARIMA

Model	AIC	BIC
ARIMA(0,1,1)	66,84	73,61
ARIMA(0,1,1) <i>with drift</i>	63,30	73,47
ARIMA(1,1,2)	65,98	79,54



AIC dan BIC yang dihasilkan dari model ARIMA(0,1,1) *with drift* lebih kecil dibandingkan ARIMA(0,1,1) dan ARIMA(1,1,2). Model ARIMA(0,1,1) *with drift* dipilih sebagai model ARIMA terbaik untuk meramalkan jumlah positif Covid-19 di DKI Jakarta. Persamaan model akhir untuk peramalan jumlah positif Covid-19 di DKI Jakarta adalah ARIMA(0,1,1) *with drift*:

$$\begin{aligned}(1 - B)W_t &= 0,0139 + (1 + 0,7069B)\varepsilon_t \\ W_t - W_{t-1} &= 0,0139 + \varepsilon_t + 0,7069\varepsilon_{t-1} \\ W_t &= 0,0139 + W_{t-1} + \varepsilon_t + 0,7069\varepsilon_{t-1}\end{aligned}\quad (33)$$

dengan  $W_t$  adalah data hasil transformasi jumlah positif Covid-19 di DKI Jakarta dengan  $\lambda=0,0223$  pada periode ke- $t$ .

Model ARIMA pada persamaan (33) akan dikembalikan menjadi data awal ( $y_t$ ) menjadi persamaan berikut:

$$\begin{aligned}\frac{y_t^\lambda - 1}{\lambda} &= 0,0139 + \frac{y_{t-1}^\lambda - 1}{\lambda} + \varepsilon_t + 0,7069\varepsilon_{t-1} \\ y_t^\lambda - 1 &= 0,00031 + y_{t-1}^\lambda - 1 + 0,0223\varepsilon_t + 0,01576\varepsilon_{t-1} \\ y_t^\lambda &= 0,00031 + y_{t-1}^\lambda + 0,0223\varepsilon_t + 0,01576\varepsilon_{t-1} \\ y_t &= (0,00031 + y_{t-1}^{0,0223} + 0,0223\varepsilon_t + 0,01576\varepsilon_{t-1})^{1/0,0223}\end{aligned}\quad (34)$$

dengan  $y_t$  adalah jumlah positif Covid-19 di DKI Jakarta periode ke- $t$ .

Berdasarkan persamaan (34) model ARIMA(0,1,1) *with drift*, jumlah positif Covid-19 di DKI Jakarta dipengaruhi oleh jumlah kasus pada satu hari sebelumnya. Jumlah kasus satu hari sebelumnya memiliki hubungan yang positif terhadap  $y_t$  yang berarti jumlah positif Covid-19 di DKI Jakarta pada periode satu hari sebelumnya memiliki hubungan yang searah dengan jumlah positif Covid-19 di DKI Jakarta pada periode yang akan datang.

#### 4.5 Artificial Neural Networks

Jaringan *Artificial Neural Networks* (ANN) yang optimal dibangun melalui beberapa tahapan, yaitu penentuan banyaknya lapisan masukan (*input layer*), lapisan tersembunyi (*hidden layer*), dan lapisan keluaran (*output layer*). Penelitian ini menentukan *input layer* berdasarkan plot PACF yang signifikan. Lampiran 11 menunjukkan plot PACF yang signifikan pada lag 1, 2, dan 3, sehingga peramalan jumlah positif Covid-19 di DKI Jakarta menggunakan masukan berupa lag data deret waktu ( $y_{t-1}, y_{t-2}, y_{t-3}$ ), sementara keluaran merupakan data deret waktu ke- $t$  ( $y_t$ ). Jaringan ini nantinya akan membentuk suatu keterikatan antara data pada periode ke- $t$  ( $y_t$ ) dengan data pada periode-periode sebelumnya ( $y_{t-1}, y_{t-2}, y_{t-3}$ ).

Penentuan banyaknya *input layer* dan *hidden layer* terbaik dilakukan dengan *trial and error*. Proses *trial and error* dilakukan karena jumlah neuron pada *input layer* dan *hidden layer* sangat berpengaruh terhadap tingkat akurasi yang dihasilkan. Jumlah neuron yang digunakan pada *hidden layer* pada dasarnya dapat berjumlah tidak terbatas karena tidak ada aturan baku mengenai jumlah neuron pada *hidden layer*. Penelitian ini menggunakan neuron pada *hidden layer* sebanyak 1 hingga 10 neuron. Jaringan terbaik dipilih berdasarkan nilai RMSE terkecil yang diperoleh pada proses *trial and error*. Proses normalisasi dilakukan pada data jumlah Covid-19 di DKI Jakarta agar data mempunyai selang [0,1].

Jaringan dibentuk dengan mengkombinasikan antara tiga neuron pada *input layer*, 10 neuron pada *hidden layer*, dan satu neuron pada *output layer*. Tabel 10 menunjukkan 30 jaringan yang terbentuk dari metode ANN algoritma *backpropagation*.

Tabel 10 Jaringan hasil *trial and error* metode ANN

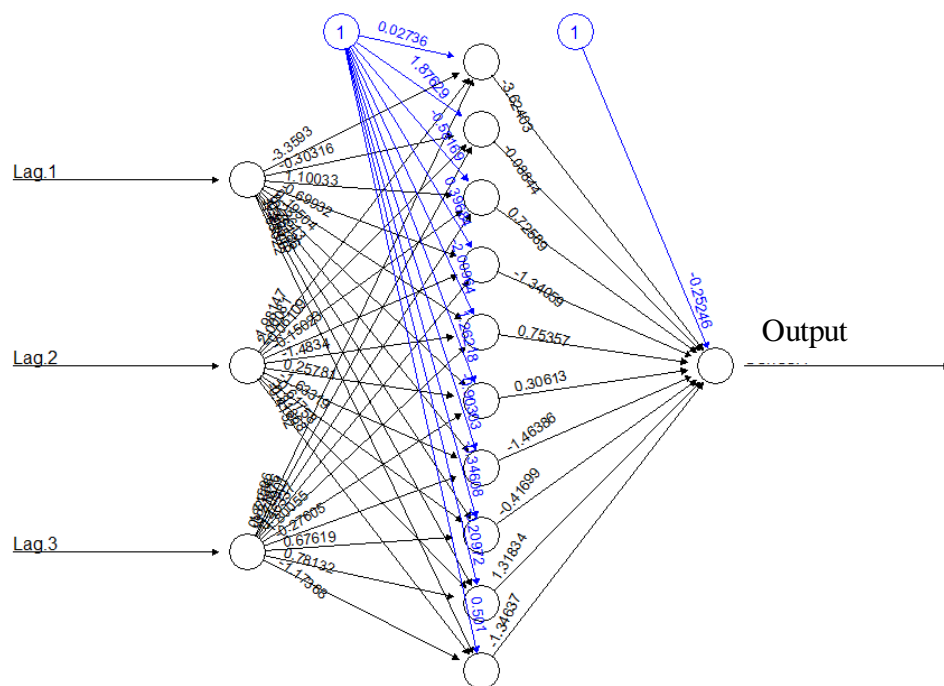
Jaringan	Lag input	Banyak parameter	RMSE
BP(1,1,1)	1	4	99,5199
BP(2,1,1)	1,2	5	97,1253
BP(3,1,1)	1,2,3	6	95,4314
BP(1,2,1)	1	7	99,8178
BP(2,2,1)	1,2	9	97,6610
BP(3,2,1)	1,2,3	11	95,2690
BP(1,3,1)	1	10	100,2443
BP(2,3,1)	1,2	13	97,3730
BP(3,3,1)	1,2,3	16	95,0067
BP(1,4,1)	1	13	100,3212
BP(2,4,1)	1,2	17	97,0065
BP(3,4,1)	1,2,3	21	94,6440
BP(1,5,1)	1	16	99,9412
BP(2,5,1)	1,2	21	97,5355
BP(3,5,1)	1,2,3	26	95,2173
BP(1,6,1)	1	19	100,2444
BP(2,6,1)	1,2	25	98,9347
BP(3,6,1)	1,2,3	31	95,3243
BP(1,7,1)	1	22	100,0494
BP(2,7,1)	1,2	29	98,1456
BP(3,7,1)	1,2,3	36	95,8256
BP(1,8,1)	1	25	100,6499
BP(2,8,1)	1,2	33	98,2619
BP(3,8,1)	1,2,3	41	96,8274
BP(1,9,1)	1	28	102,0901
BP(2,9,1)	1,2	37	97,6007
BP(3,9,1)	1,2,3	46	94,5901
BP(1,10,1)	1	31	100,4364
BP(2,10,1)	1,2	41	97,4893
BP(3,10,1)*	1,2,3	51	93,3417

\*model ANN terbaik dari proses *trial and error*

Tabel 10 menunjukkan kemungkinan jaringan yang optimal untuk digunakan pada peramalan jumlah positif Covid-19 di DKI Jakarta. Jaringan yang terpilih adalah jaringan yang memiliki nilai kriteria akurasi yang paling kecil. Kriteria akurasi yang digunakan adalah RMSE. Jaringan yang memiliki nilai RMSE terkecil merupakan jaringan yang paling optimal dan jaringan yang akan digunakan untuk proses peramalan jumlah positif Covid-19 di DKI Jakarta.

Berdasarkan 30 jaringan yang terbentuk BP(3,10,1) memiliki nilai RMSE terkecil, yaitu 93,3417. Jaringan BP(3,10,1) terpilih sebagai jaringan yang paling optimal dibandingkan dengan 29 jaringan lainnya. Jaringan BP(3,10,1) inilah yang digunakan untuk proses validasi data jumlah positif Covid-19 di DKI Jakarta.

Tabel 10 juga menunjukkan bahwa penambahan neuron pada *input layer* dan *hidden layer* tidak menjamin jaringan yang terbentuk menjadi semakin optimal. Selain itu, penambahan neuron pada *hidden layer* tidak membuat nilai RMSE semakin mengecil. Akan tetapi, berdasarkan beberapa percobaan pembangunan jaringan yang dilakukan, penambahan *input layer* dan *hidden layer* akan menghasilkan nilai AIC dan BIC yang semakin meningkat. Gambar 9 memperlihatkan arsitektur BP(3,10,1) yang memiliki *input layer* sebanyak tiga, yaitu  $y_{t-1}, y_{t-2}, y_{t-3}$ . *Input layer* dan *output layer* dihubungkan oleh 10 neuron pada *hidden layer*. Nilai yang menghubungkan satu neuron dengan neuron yang lain disebut dengan bobot. Gambar 9 menunjukkan bobot-bobot yang menghubungkan antara neuron pada *input layer* dengan neuron pada *hidden layer* serta neuron pada *hidden layer* dengan neuron pada *output layer*. Bobot-bobot yang menghubungkan antara satu neuron dengan neuron yang lainnya dihasilkan melalui proses pengacakan bilangan di software R. Model terbaik BP(3,10,1) dari data latih akan dilakukan proses validasi model dengan melakukan peramalan sebanyak data uji yang nilai peramalannya akan dibandingkan dengan nilai aktual.

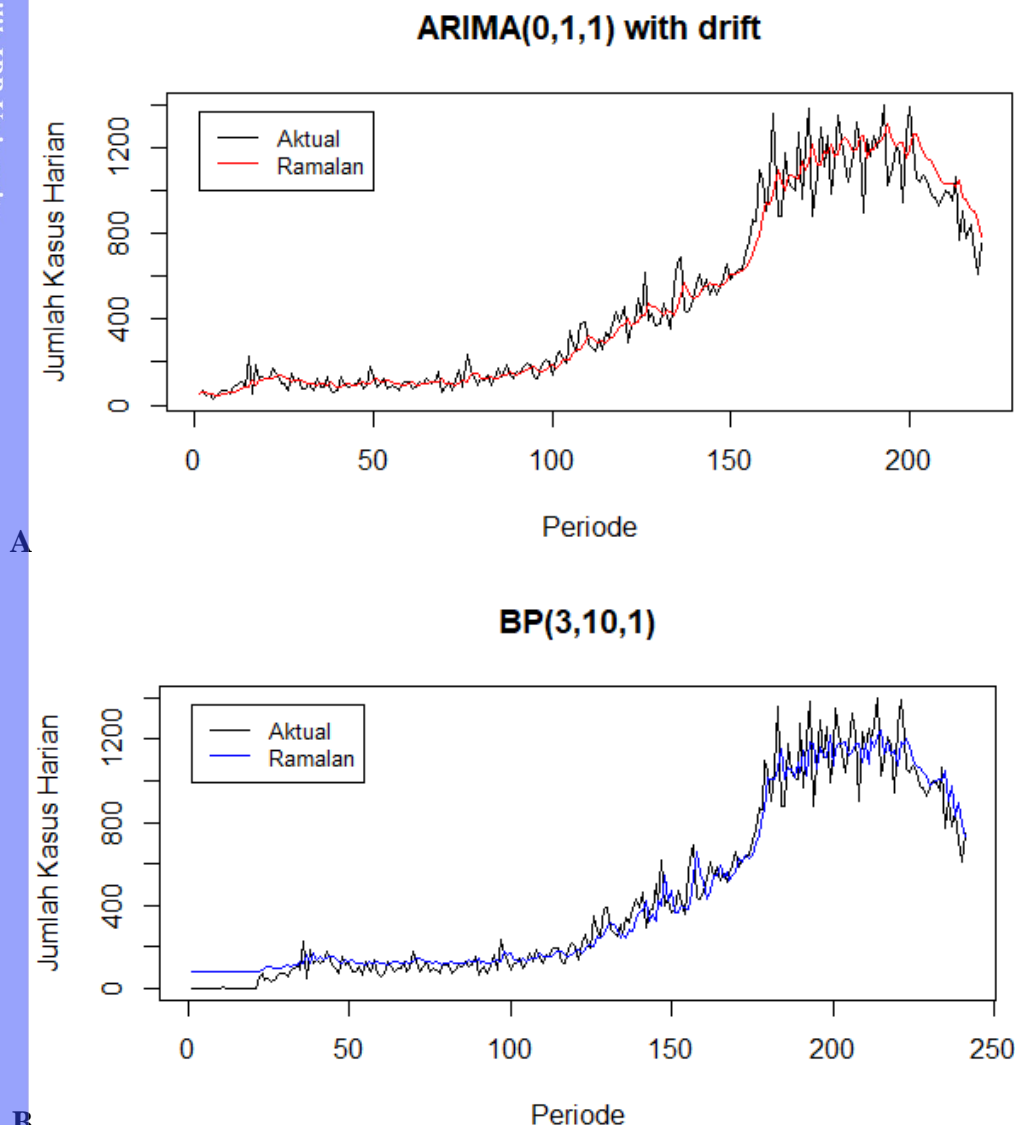


Gambar 9 Arsitektur BP(3,10,1)

#### 4.6 Perbandingan ARIMA DAN ANN

Validasi digunakan untuk membandingkan keakuratan peramalan jumlah positif Covid-19 di DKI Jakarta dengan metode ARIMA dan ANN. Perbandingan model terbaik dilakukan dengan membandingkan nilai RMSE, MAPE, dan hasil peramalan jumlah positif Covid-19 di DKI Jakarta. Proses validasi dilakukan terhadap masing-masing metode terbaik yang dihasilkan pada proses pelatihan, yaitu model ARIMA(0,1,1) *with drift* dan BP(3,10,1). Hasil peramalan dari kedua metode terbaik akan dilakukan perbandingan dengan data aktual.

Hak Cipta milik IPB University



Gambar 10 Plot data aktual dan hasil peramalan dengan data latih pada ARIMA(0,1,1) *with drift* (A) dan BP(3,10,1) (B)

Gambar 10 memperlihatkan plot hasil peramalan ARIMA(0,1,1) *with drift* dan BP(3,10,1) dengan menggunakan data latih. Gambar 10(A) dan (B) memperlihatkan bahwa hasil peramalan kedua model mengikuti pola dari data aktual. Namun, ARIMA(0,1,1) *with drift* dan BP(3,10,1) kurang bisa menangkap data-data yang memiliki fluktuasi yang tinggi. Selain menggunakan grafik,

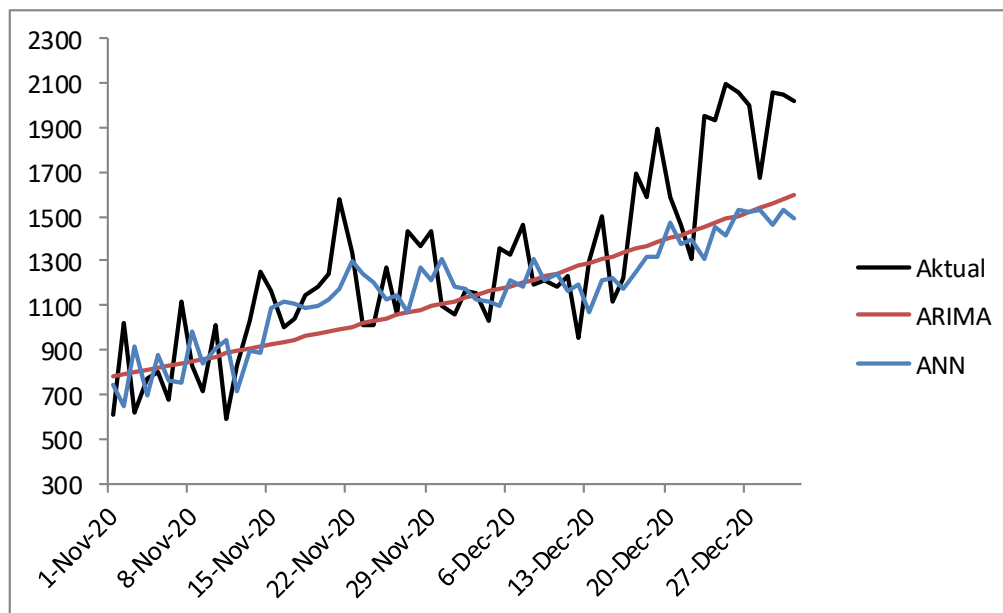


kebaikan peramalan menggunakan data latih dapat dilihat dari nilai RMSE dan korelasi yang diperoleh oleh kedua model. Hasil perhitungan akurasi peramalan metode ARIMA dan ANN dengan menggunakan data latih disajikan pada Tabel 11.

Tabel 11 Hasil validasi model-model terbaik dengan data latih

Model	Akurasi	
	RMSE	Korelasi
ARIMA	98,174	0,9738
ANN	93,342	0,9760

Tabel 11 menunjukkan bahwa nilai RMSE yang dihasilkan oleh metode ANN lebih kecil dibandingkan dengan metode ARIMA. Selain itu, nilai korelasi antara data aktual dengan hasil peramalan menunjukkan metode ANN memiliki nilai korelasi yang sedikit lebih besar dibandingkan dengan metode ARIMA. Selisih nilai RMSE dan korelasi kedua model berturut-turut, yaitu 4,832 dan 0,0022. Berdasarkan selisih tersebut dapat dikatakan bahwa metode ANN memiliki kemampuan yang lebih baik dalam melakukan peramalan dengan menggunakan data latih dibandingkan dengan metode ARIMA pada kasus jumlah positif Covid-19 di DKI Jakarta.



Gambar 11 Hasil peramalan model ARIMA dan ANN dengan data uji

Gambar 11 memperlihatkan hasil peramalan jumlah positif Covid-19 di DKI Jakarta periode 1 November 2020 sampai 31 Desember 2020. Peramalan dengan metode ANN cenderung mengikuti pola data aktual. ANN mungkin lebih cocok untuk menggambarkan data yang memiliki *trend* taklinear. Awal periode peramalan terlihat hasil ramalan dari model ANN cukup dekat dengan data aktual, sedangkan model ARIMA cenderung memberikan hasil peramalan yang terarah (linear) untuk periode peramalan jangka panjang. Selain menggunakan grafik,

kebaikan hasil peramalan dapat dilihat menggunakan kriteria akurasi, yaitu RMSE, MAPE, dan korelasi antara data aktual dengan data hasil ramalan. Hasil perhitungan akurasi peramalan metode ARIMA dan ANN ditampilkan pada Tabel 12.

Tabel 12 Hasil validasi model-model terbaik dengan data uji

Model	Akurasi		
	RMSE	MAPE(%)	Korelasi
ARIMA	268,808	15,748	0,845
ANN	283,365	16,872	0,798

Berdasarkan Tabel 12, terlihat bahwa metode ARIMA memiliki nilai RMSE, MAPE, dan korelasi yang lebih kecil dibandingkan dengan metode ANN. Nilai selisih RMSE, MAPE, dan korelasi dari kedua metode berturut-turut sebesar 14,557, 1,124%, dan 0,047. Berdasarkan selisih nilai RMSE, MAPE, dan korelasi terlihat bahwa metode ARIMA lebih unggul dibandingkan dengan metode ANN dalam kasus peramalan jumlah positif Covid-19 di DKI Jakarta. Namun, kedua model tersebut memiliki kemampuan yang baik untuk peramalan jumlah positif Covid-19 di DKI Jakarta karena memperoleh MAPE yang lebih kecil dari 20%. Menurut Chang *et al.* (2007) model dikatakan baik jika memiliki nilai MAPE 10–20%.

Tabel 13 Nilai akurasi pada berbagai panjang periode peramalan

Panjang periode peramalan	ARIMA(0,1,1) with drift		BP(3,10,1)	
	RMSE	MAPE(%)	RMSE	MAPE(%)
1 minggu	178,452	19,802	239,975	24,636
2 minggu	186,761	19,529	231,802	24,086
3 minggu	223,355	19,126	214,207	19,516
4 minggu	226,206	17,670	208,898	18,049
5 minggu	213,974	15,838	200,661	16,652
6 minggu	207,238	14,959	193,903	15,625
7 minggu	217,711	14,911	218,231	15,919
8 minggu	250,511	15,368	258,715	16,359
9 minggu	268,808	15,748	283,365	16,872

Perbandingan metode ARIMA dan ANN juga dilakukan dengan membandingkan pergerakan nilai RMSE dan MAPE ketika panjang periode peramalan diperbaharui. Tabel 13 menunjukkan pergerakan nilai RMSE dan MAPE dari model ARIMA(0,1,1) with drift dan BP(3,10,1). Tabel 13 memperlihatkan bahwa ARIMA(0,1,1) with drift memiliki nilai RMSE dan MAPE yang lebih kecil dibandingkan BP(3,10,1) pada panjang periode peramalan satu hingga sembilan minggu. Selain itu, ketika periode peramalan semakin panjang nilai RMSE dan MAPE yang dihasilkan oleh model ARIMA(0,1,1) with drift menjadi semakin kecil. Model BP(3,10,1) menghasilkan nilai RMSE dan

MAPE yang semakin kecil, seiring dengan bertambah panjangnya periode peramalan. Akan tetapi, ketika periode peramalan sebanyak tujuh minggu, delapan minggu, dan sembilan minggu menghasilkan nilai RMSE dan MAPE yang cenderung meningkat.

Pola peramalan yang cenderung *trend* yang terarah (linear) pada ARIMA(0,1,1) *with drift* inilah yang menyebabkan nilai RMSE dan MAPE yang dihasilkan menjadi semakin kecil dibandingkan dengan model ANN. Model ARIMA dan ANN menghasilkan peramalan dengan nilai MAPE yang cenderung stabil untuk periode peramalan jangka panjang dan data yang memiliki unsur *trend*. Peramalan dengan panjang periode enam minggu menggunakan model ARIMA(0,1,1) *with drift* dan BP(3,10,1) menghasilkan nilai RMSE dan MAPE yang paling kecil. Hal tersebut menunjukkan bahwa model ARIMA(0,1,1) *with drift* dan BP(3,10,1) dapat meramalkan dengan baik untuk panjang periode peramalan enam minggu ke depan pada kasus jumlah positif Covid-19 di DKI Jakarta.

@Hak cipta milik IPB University

IPB University



## V SIMPULAN DAN SARAN

### 5.1 Simpulan

Simpulan Berdasarkan hasil yang diperoleh, untuk data jumlah positif Covid-19 di DKI Jakarta model terbaik ARIMA didapatkan ARIMA(0,1,1) *with drift*, sedangkan yang didapatkan dari model ANN terbaik adalah BP(3,10,1). Permodelan ARIMA yang terbentuk menunjukkan bahwa jumlah positif Covid-19 di DKI Jakarta dipengaruhi oleh jumlah kasus satu hari sebelumnya, sedangkan model ANN yang terbentuk menunjukkan bahwa jumlah positif Covid-19 di DKI Jakarta dipengaruhi oleh jumlah kasus satu hari, dua hari, dan tiga hari sebelumnya. Berdasarkan nilai RMSE, MAPE, dan korelasi yang diperoleh dapat ditunjukkan bahwa metode ARIMA meramalkan jumlah positif Covid-19 di DKI Jakarta lebih baik dibandingkan dengan metode ANN. Peramalan yang dilakukan model ARIMA(0,1,1) *with drift* dan BP(3,10,1) menghasilkan ramalan terbaik pada panjang periode peramalan enam minggu.

### 5.2 Saran

Penelitian selanjutnya dapat mempertimbangkan peubah yang lainnya, karena penambahan jumlah positif Covid-19 di DKI Jakarta bukan hanya dipengaruhi oleh penambahan kasus pada periode sebelumnya tetapi bisa disebabkan oleh banyak faktor, seperti akses pengetahuan dan peralatan medis, populasi di wilayah tersebut, dan perilaku masyarakat di wilayah tersebut. Selain itu, pada penelitian ini belum adanya vaksinasi, sehingga untuk penelitian selanjutnya bisa ditambahkan variabel mengenai vaksinasi.

## DAFTAR PUSTAKA

- Adebiyi AA, Aderemi AO, Ayo CK. 2014. Comparison of ARIMA and artificial neural networks models for stock price prediction. *Journal of Applied Mathematics*. 2014:1–7.
- Adeyinka DA, Muhajarine N. 2020. Time series prediction of under-five mortality rates for Nigeria: comparative analysis of artificial neural networks, Holt-Winters exponential smoothing and autoregressive integrated moving average models. *BMC Medical Research Methodology*. 20(292):1–11.
- Box GEP, Jenkins GM, Reinsel GC. 2008. *Time Series Analysis: Forecasting and Control*. Ed ke-4. New Jersey(US): John Wiley & Sons.
- Chang PC, Wang YW, Liu CH. 2007. The development of weighted evolving fuzzy neural network for PCB sales forecasting. *Expert System with Applications*. 32(2007):86–96.
- Cryer JD, Chan KS. 2008. *Time Series Analysis with Application in R*. Ed ke-2. New York(US):Springer.
- Fausett L. 1994. *Fundamental of Neural Networks: Achitectures, Algorithms, and Applications*. New Jersey(US):Prentice Hall.
- Irsyadinna. 2009. Klasifikasi debitur untuk kredit pemilikan rumah bank X dengan regresi logistik biner dan *artificial neural networks* [skripsi]. Bogor:Institut Pertanian Bogor.
- Montgomery DC, Jennings CL, Kulahci M. 2015. *Introduction to Time Series Analysis and Forecasting*. New Jersey(US):John Wiley & Sons.
- Ristiana R. 2015. Perbandingan ARIMA dan jaringan syaraf tiruan propagasi balik dalam peramalan tingkat inflasi nasional [skripsi]. Bogor: Institut Pertanian Bogor.
- Rusdi. 2011. Uji akar-akar unit dalam model runtun waktu autoregresif. *Statistika*. 11(2):67–78.
- Sahai AK, Rath N, Sood V, Singh MP. 2020. ARIMA modeling & forecasting of Covid-19 in top five affected countries. *Diabetes & Metabolic Syndrome: Clinical Research & Reviews*. 14(2020):1419–1427.
- Satgas Penanganan Covid-19. [internet, diakses 2020 Desember 18]. Tersedia pada <https://covid19.go.id/peta-sebaran-covid19>.
- Suleiman S, Sani M. 2020. Application of ARIMA and artificial neural networks models for daily cumulative confirmed Covid-19 prediction in Nigeria. *Equity Journal of Science and Technology*. 7(2): 83–90.
- Wei WWS. 2006. *Time Series Analysis: Univariate and Multivariate Methods*. Ed ke-2. United States (US): Pearson Education.
- Wieczorek M, Silka J, Wozniak M. Neural network powered Covid-19 spread forecasting model. *Chaos, Solitons and Fractals*. 140(2020):1–15.
- Wikantara IPP. 2016. Perbandingan model ARIMA dengan model recurrent artificial neural network dalam memprediksi jumlah wisatawan asing yang datang ke provinsi Bali [skripsi]. Surabaya: Institut Teknologi Sepuluh Nopember.
- Wuryandari MD, Afrianto I. 2012. Perbandingan metode jaringan syaraf tiruan *backpropagation* dan *learning vector quantization* pada pengenalan wajah. *Komputer dan Informatika*. 1(1): 45–51.



- Yu L, Wang S, Lai KK. 2005. A novel nonlinear ensemble forecasting model incorporating GLAR and ANN for freight exchanges rates. *Computer & Operations Research*. 32(2005 ):2523–2541.
- Zhang GP. 2003. Time series forecasting using a hybrid ARIMA and neural network model. *Neurocomputing*. 50(2003):159–175.
- [WHO] World Health Organization. [internet, diakses 2020 Desember 18]. Tersedia pada [www.who.int](http://www.who.int).
- [Kemenkes RI] Kementerian Kesehatan Republik Indonesia. [internet, diakses 2020 Desember 18]. Tersedia pada <https://www.kemkes.go.id/folder/view/full-content/structure-faq.html>.

## LAMPIRAN

@Hak cipta milik IPB University

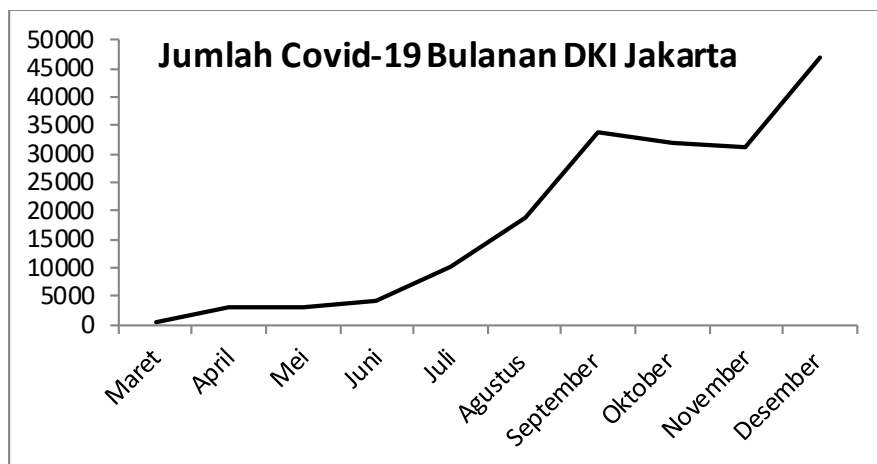
IPB University



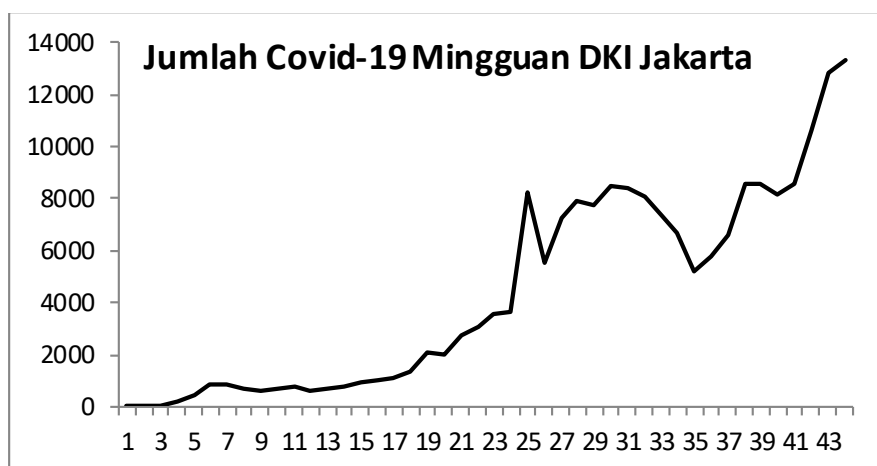
Hak Cipta Dilindungi Undang-undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah
  - b. Pengutipan tidak merugikan kepentingan yang wajar IPB University.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IPB University.

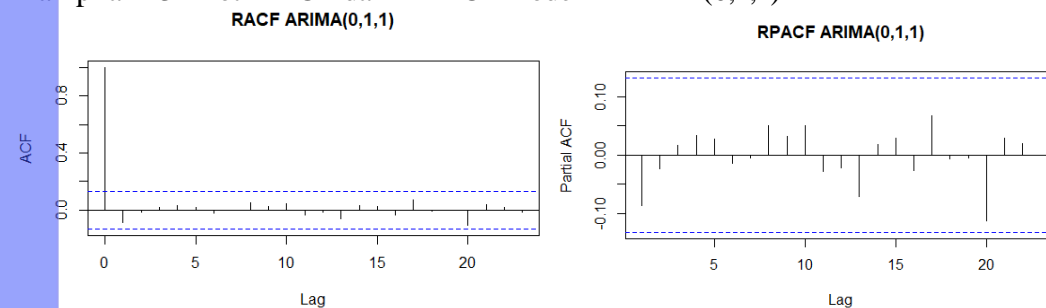
Lampiran 1 Plot jumlah positif Covid-19 periode bulanan di DKI Jakarta



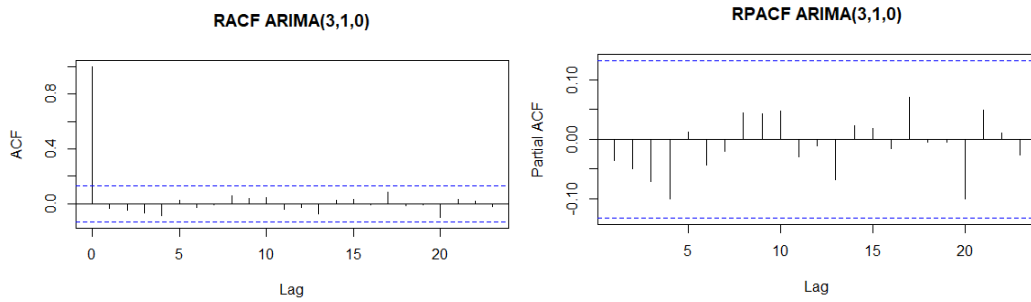
Lampiran 2 Plot jumlah positif Covid-19 periode mingguan di DKI Jakarta



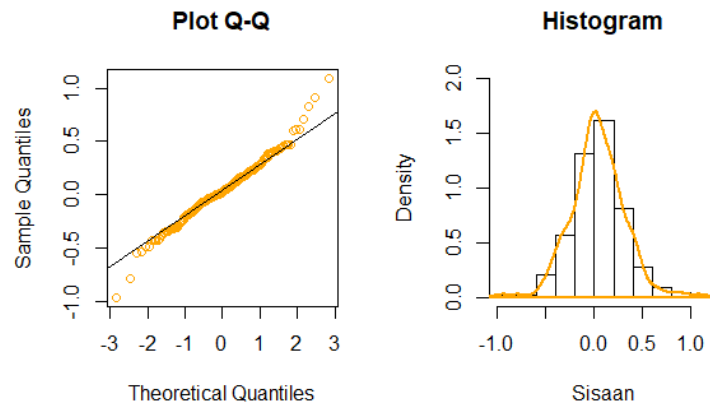
Lampiran 3 Plot RACF dan RPACF model ARIMA(0,1,1)



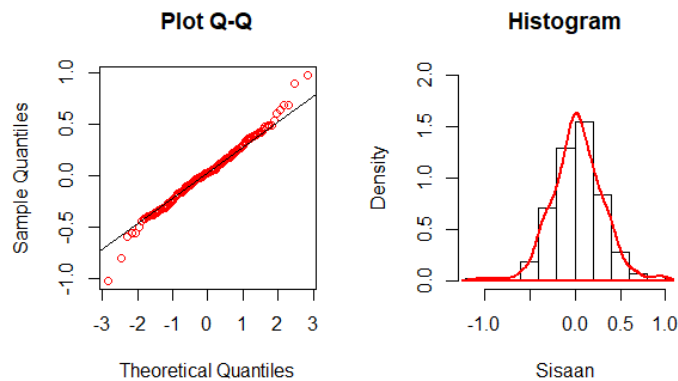
## Lampiran 4 Plot RACF dan RPACF model ARIMA(3,1,0)



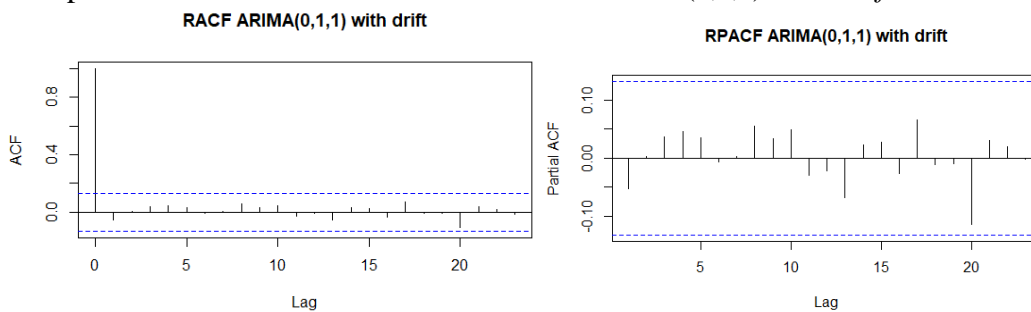
## Lampiran 5 Plot sebaran normal sisaan model ARIMA(0,1,1)



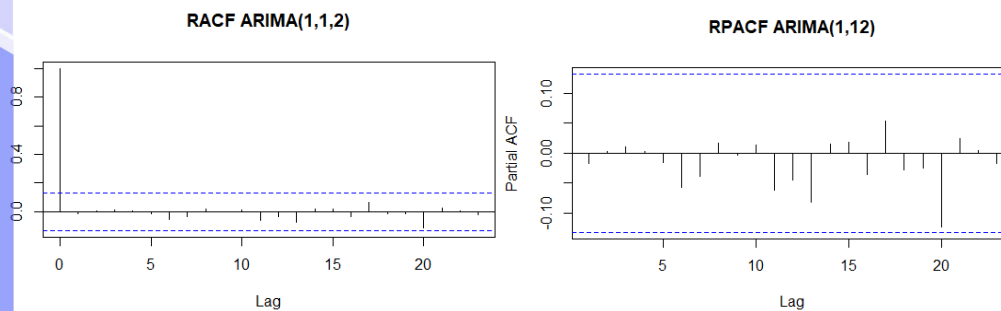
## Lampiran 6 Plot sebaran normal sisaan model ARIMA(3,1,0)



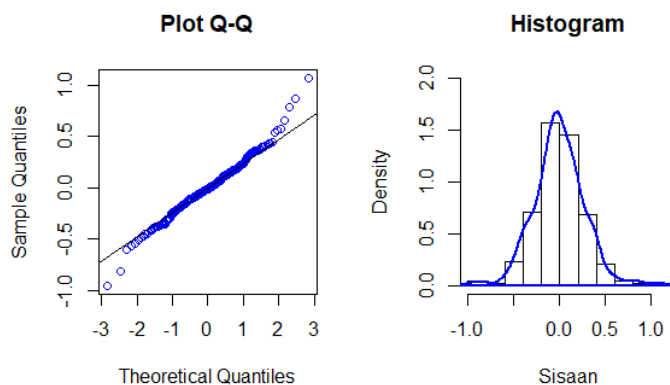
## Lampiran 7 Plot RACF dan RPACF model ARIMA(0,1,1) with drift



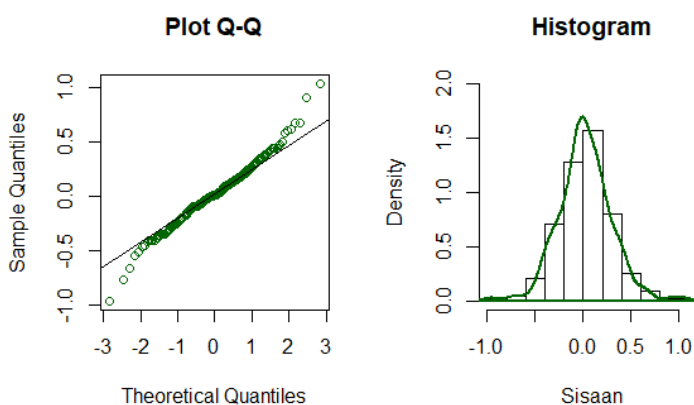
## Lampiran 8 Plot RACF dan RPACF model ARIMA(1,1,2)



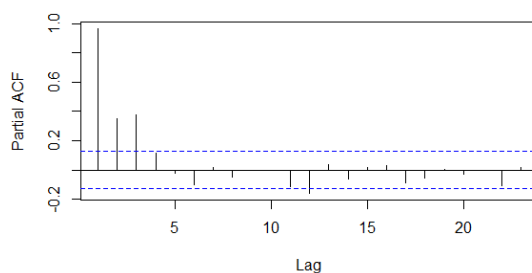
## Lampiran 9 Plot sebaran normal sisaan model ARIMA(0,1,1) with drift



## Lampiran 10 Plot sebaran normal sisaan model ARIMA(1,1,2)



## Lampiran 11 Plot PACF jumlah positif Covid-19 di DKI Jakarta dengan data latih yang belum stasioner dalam ragam dan rata-rata





## RIWAYAT HIDUP

Penulis dilahirkan di kota Tanjungpandan, Belitung pada 10 Juni 1999 sebagai anak ketiga dari pasangan Bapak Sufandi dan Ibu Ardiyanti. Pendidikan sekolah menengah atas (SMA) ditempuh di SMA Negeri 1 Tanjungpandan dan lulus pada tahun 2017. Penulis diterima sebagai mahasiswa program sarjana (S-1) di Program Studi Statistika dan Sains Data di IPB melalui jalur SNMPTN pada tahun 2017.

Selama mengikuti program S-1, penulis aktif menjadi pengurus di Departemen Database Center Himpunan Profesi Gamma Sigma Beta Statistika IPB periode 2018-2019, pengurus Koperasi Mahasiswa IPB, dan organisasi mahasiswa daerah Belitung. Penulis juga aktif dalam kegiatan kepanitiaan yang diadakan IPB sebagai anggota Liaison Officer Statistika Ria 2019, anggota divisi Humas Pesta Sains Nasional 2019, anggota divisi Humas Pekan Olahraga FMIPA, dan Anggota Kestari Kompetisi Statistika Junior 2020. Selain kegiatan non akademik, penulis juga aktif sebagai asisten praktikum mata kuliah Analisis Peubah Ganda. Juni hingga Agustus penulis melaksanakan praktik lapang di BPS Kabupaten Belitung dan ditempatkan di divisi Statistik Distribusi. Penulis juga pernah menjadi *surveyor* pada beberapa instansi *marketing riset*.

@Hak cipta milik IPB University

IPB University

