

BAB II

TINJAUAN PUSTAKA

2.1 Tinjauan Non Statistik

2.1.1 Peramalan

Peramalan adalah prakiraan tentang sesuatu yang akan terjadi dimasa yang akan datang. Sebuah produksi adalah salah satu masalah yang tidak stabil atau berubah sewaktu – waktu dan tidak dapat di tentukan. Dari masalah tersebut maka peramalan merupakan salah satu metode yang tepat untuk mengetahui hasil dengan kesalahan yang kecil di waktu mendatang. Metode peramalan dapat memberikan pengerjaan yang mudah dipahami, teratur, dan terarah. Dengan teknik peramalan diharapkan dapat memberikan tingkat keakuratan yang lebih besar.

Menurut Barry (2009:167), metode peramalan atau teknik peramalan dibagi menjadi dua yaitu :

1. Peramalan Kuantitatif

Peramalan kuantitatif yaitu peramalan yang diambil pada informasi di masa lalu dengan asumsi bahwa pola masa lalu akan terus berlanjut dimasa mendatang. Dari data masa lalu tersebut, maka dapat diolah menjadi nilai yang sering di sebut dengan data *time series* atau data runtun waktu.

2. Peramalan Kualitatif

Peramalan Kualitatif adalah peramalan yang di ambil pada pendapat para ahli dan data yang digunakan tidak dapat diolah menjadi angka. Pendapat para ahli tersebut hanya menjadi sebuah pertimbangan dalam mengambil keputusan untuk peramalan.

2.1.2 Harga Gabah

Gabah dibagi menjadi beberapa jenis menurut kualitas dan harga produsennya. Salah satunya yaitu GKP. GKP yaitu gabah yang mengandung kadar air lebih besar dari 18% dan lebih kecil dari atau sama dengan 25%, hampa/kotoran lebih dari 6% dan lebih kecil dari atau sama dengan 10%. Kualitas gabah di suatu daerah biasanya ditentukan oleh musim panen. Pada musim panen paceklik lebih baik hasilnya dibandingkan dengan musim panen raya. Maka dari itu, harga gabah akan tinggi pada periode paceklik dan mengikuti perkembangan kualitas gabah dan tingkat produksi gabah. Berbeda halnya saat terjadi pandemi Covid-19, dimana para petani mengalami penurunan meskipun musim paceklik dan harga gabah mengalami penurunan.

2.2 Tinjauan Statistik

2.2.1 Analisis Time Series

Time series atau deret waktu yaitu nilai pengamatan berdasarkan waktu dan interval yang sama dan diukur selama kurun waktu tertentu. Waktu pada *time series* yaitu berupa mingguan, bulanan, semester, tahunan, dan lain – lain. Menurut Wei (2006) peramalan menggunakan deret waktu yaitu untuk mengetahui kejadian di masa depan dengan harapan dapat membuat keputusan

yang tepat dan terbaik. Tujuan dari peramalan deret waktu yaitu untuk menemukan pola deret data yang ada di masa lalu dan mengolah data di masa lalu menjadi data masa depan. Menurut Taylor (2013) metode *time series* mengansumsikan bahwa apa yang terjadi pada data masa lalu akan terus berlanjut dan terjadi di masa yang akan mendatang.

2.2.2 Stasioneritas Data

Kestasioneran data *time series* yaitu keadaan dimana data berada pada titik seimbang di sekitar nilai rata-rata yang konstan dan variasi di sekitar rata-rata tersebut konstan dalam jangka waktu tertentu. Menurut Markridakis (1999) stasioneritas memiliki makna bahwa tidak terdapat perubahan yang drastis pada sebuah data. Di dalam stasioneritas Data ada dua perilaku stasioner yaitu :

a. Stasioner dalam *varians*

Menurut Wei (2006) stasioner dalam *varians* yaitu apabila data berfluktuasi dengan varian yang tetap dari waktu ke waktu dan memenuhi persamaan berikut ini :

$$\text{Var}(Y_t) = \text{Var}(Y_{t+k}) = \sigma^2 \quad (2.1)$$

Data dikatakan tidak memenuhi stasioner maka dilakukan transformasi *Box-Cox*. *Box-Cox Transformation* yaitu transformasi pangkat pada variabel tak bebas, suatu deret waktu yang tidak stasioner dalam hal *varians* yang harus diubah menjadi data stasioner dengan melakukan transformasi. Transformasi digunakan untuk mengetahui apakah data sudah stasioner dalam *varians* yang dapat dilihat dari nilai

lamda. Apabila nilai lamda=1 maka data dikatakan stasioner dalam varians (Rosyidah, Rahmawati, dan Prahutama, 2017). Menurut Wei (2006) rumus matematis transformasi dan tabel beberapa lamda yaitu sebagai berikut:

Tabel 2. 1 Rumus Transformasi

Nilai λ	Transformasi
-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

b. Stasioner dalam *mean*

Data dikatakan Stasioner pada nilai tengah (*mean*) adalah apabila data berfluktuasi disekitar nilai tengah yang tetap selama mandiri. Jika plot data berfluktuasi disekitar garis sejajar dengan sumbu waktu ke-t atau di sekitar nilai mean yang konstan.

$$E(Y_t) = E(Y_{t+k}) = \mu \quad (2.2)$$

Menurut Nabilah (2017) kondisi stasioner tidak terpenuhi dalam *mean* maka perlu dilakukan proses pembedaan (*differencing*) terhadap data

asli (Y_t). Secara umum proses pembedaan pada orde ke-d dapat dinyatakan sebagai berikut

$$W_t = (1 - B)^d Y_t \quad (2.3)$$

Menurut Rosadi (2012) untuk menguji stasioneritas dari data deret waktu dapat dilakukan dengan berbagai cara yaitu sebagai berikut:

1. Untuk mengetahui data yang tidak stasioner dalam *mean* maka digunakan plot runtun waktu yaitu plot ACF dan PACF. Jika data ada komponen *trend*, maka plot ACF dan PACF akan meluruh dengan perlahan dan data stasioner dalam *mean*.
2. Untuk mengetahui data yang tidak stasioner maka dapat diperiksa dengan melihat apakah data runtun waktu mengandung akar unit, terdapat berbagai metode untuk melakukan uji akar unit contohnya yaitu uji *Augmented Dickey Fuller*

2.2.3 Fungsi Autokorelasi (ACF)

Autokorelasi atau sering disebut dengan ACF adalah fungsi yang memperlihatkan besarnya korelasi antara pengamatan pada waktu k pada masa sekarang dengan penelitian pada waktu di masa lalu. ($k-1, k-2, \dots, k-m$). dengan mengambil sampel dari populasi maka ACF dapat dihitung dengan persamaan matematis sebagai berikut (Cryer & Chan, 2008):

$$Z_t = \hat{\rho}_k = \frac{\sum_{t=k+1}^{n-k} (y_t - \bar{y})(y_{t+k} - \bar{y})}{\sum_{i=1}^n (y_t - \bar{y})^2} \quad (2.4)$$

Keterangan:

ρ_k : Koefisien autokorelasi pada *lag*- k

- k : selisih waktu
 n : Jumlah observasi
 \bar{y} : Rata-rata dari pengamatan
 y_t : Pengamatan pada waktu ke- t
 y_{t-k} : Pengamatan pada waktu ke- $t-k$

2.2.4 Fungsi Autokorelasi Parsial (PACF)

Fungsi PACF adalah fungsi yang menunjukkan hubungan atau korelasi parsial antara pengamatan dari waktu ke waktu pada pengamatan waktu sebelumnya ($t-1, t-2, \dots, t-k$). Menurut Cryer & Chan (2008) Persamaan dari PACF adalah sebagai berikut:

$$\vartheta_{k+1,k+1} = \frac{\hat{\rho}_k - \sum_{j=1}^k \hat{\vartheta}_{kj} \hat{\rho}_{k+1-j}}{1 - \sum_{j=1}^k \hat{\vartheta}_{kj} \hat{\rho}_j} \quad (2.5)$$

2.2.5 Identifikasi Model

Identifikasi Model dapat dilihat dari plot ACF dan PACF pada analisis runtun waktu. Menurut Wei (2006) secara teoritis bentuk-bentuk plot ACF dan PACF pada model ARIMA dapat dilihat pada tabel di bawah ini:

Tabel 2. 2 Identifikasi Plot ACF dan PACF

Model	ACF	PACF
AR (p)	<i>Dies Down</i>	<i>Cut off</i> setelah lag- p
MA (q)	<i>Cut off</i> setelah lag- q	<i>Dies Down</i>
ARMA (p, q)	<i>Dies Down</i>	<i>Dies Down</i>
AR (p) atau MA (q)	<i>Cut off</i> setelah lag- q	<i>Cut off</i> setelah lag- p

2.2.6 Autoregressive Integrated Moving Average (ARIMA)

Model ARIMA yaitu model atau metode yang digunakan untuk meramalkan masa depan dengan data masa lalu. Data yang di ramalkan pada model ARIMA yaitu data yang sudah stasioner. ARIMA terdiri dari AR(p), pembedaan atau *differencing*(d) dan MA(q) yaitu sebagai berikut :

a. Autoregressive AR(p)

Menurut Sugiarto dan Harijono (2000). *Autoregressive* adalah model dimana variabel dependen dipengaruhi oleh variabel independen pada periode waktu sebelumnya. Model Model *Autoregressive* dapat ditulis sebagai berikut:

$$Y_t = \theta_0 + \theta_1 Y_{t-1} + \theta_2 Y_{t-2} + \dots + \theta_p Y_{t-p} - e_t \quad (2.6)$$

Keterangan :

Y_t : Deret Waktu Stasioner

θ_0 : Konstanta

Y_{t-1}, Y_{t-p} : Nilai masa lalu yang berhubungan

$\theta_1, \dots, \theta_p$: Koefisien atau parameter dari model *autoregressive*

e_t : residual pada waktu ke-t

b. Moving Average MA(q)

Model MA diberi notasi q ditentukan oleh jumlah periode variabel independen yang masuk dalam model. Model MA mempunyai bentuk sebagai berikut (Wei,2006) :

$$Z_t = \delta + \alpha_1 - \theta_1 \alpha_{t-1} - \theta_2 \alpha_{t-2} - \dots - \theta_q \alpha_{t-q} \quad (2.7)$$

Dimana :

Y_t : Deret waktu stasioner

θ_q : koefisien model *moving average* ke q

α_t : Nilai galat pada t

c. Autoregressive Moving Average ARMA(p,q)

Model ARMA yaitu model yang menggabungkan antara proses AR dan MA. Model ini dapat dibentuk sebagai berikut (Wei,2006):

$$Z_t = \phi_1 Z_{t-1} + \phi_2 Z_{t-2} + \alpha_t - \theta_1 \alpha_{t-1} - \theta_2 \alpha_{t-2} \quad (2.8)$$

Dimana :

- saat kejadian $q = 0$ maka model *autoregressive* dengan orde p,AR(p).
- Saat kejadian $p = 0$ maka model *moving average* dengan orde q, MA(q).

d. Autoregressive Integrated Moving Average atau ARIMA (p,d,q)

Model ARIMA dibagi menjadi dua yaitu model ARIMA musiman dan non musiman. Model ARIMA non musiman yaitu model deret waktu yang bersifat stasioner pada rata – rata dan membutuhkan differencing sebanyak d agar data tersebut stasioner. Model ARIMA musiman yaitu biasanya terdapat pada model SARIMA (*Seasonal Autoregressive Integrated Moving Average*) dimana pada model ini mempunyai sifat yang berulang setelah beberapa waktu tertentu. Model ARIMA (p,d,q) adalah model analisis deret waktu yang tidak stasioner dengan rata – rata dan membutuhkan proses *differencing* untuk membuat data tersebut menjadi stasioner.

2.2.7 Autoregressive Integrated Moving Average With Exogenous Variable (ARIMAX)

ARIMAX merupakan model ARIMA dengan adanya penambahan variabel eksogen. Persamaan ARIMAX secara umum ditulis sebagai berikut (Cryer & Chan, 2008) :

$$Z_t = \beta_0 + \beta_1 X_{1,t} + \dots + \beta_g X_{g,t} + \frac{\theta_q(B)\theta_Q(B^S)}{\phi_p(B)\phi_P(B^S)(1-B)^d(1-B^S)^D} \alpha_t \quad (2.9)$$

Dimana :

β_g : Koefisien variabel eksogen

$X_{g,t}$: Variabel Eksogen

$\theta_q(B)$: $1 - \theta_1 B - \theta_2 B^2 - \dots - \theta_q B^q$

$\theta_p(B)$: $1 - \theta_1 B - \theta_2 B^2 - \dots - \theta_p B^p$

$(1 - B)^d$: Operator *differencing* dengan orde d

$(1 - B^S)^D$: Operator *differencing seasonal* dengan orde D

ϕ_p : Parameter *Autoregressive* dengan orde p

ϕ_q : Parameter *Autoregressive* dengan orde q

B : *Backshift* operator

S : Orde musiman

α_t : Sisaan dari model

2.2.8 Estimasi Parameter

Setelah melakukan identifikasi model ARIMA sementara maka kemudian dilanjutkan dengan mengestimasi parameter-parameternya. Estimasi parameter

dapat dilakukan yaitu dengan *Likelihood*, metode yang sering digunakan untuk melakukan estimasi parameter. Parameter yang diestimasi kemudian diuji untuk mengetahui signifikansi pada model yang telah diperoleh. Menurut Bowerman and O'Connell (2003) pengujian hipotesis estimasi parameter yaitu sebagai berikut:

$H_0 = \theta = 0$ (Parameter tidak signifikan)

$H_1 = \theta \neq 0$ (Parameter signifikan)

Statistik Uji:

$$t = \frac{\hat{\theta}}{Se\hat{\theta}} \quad (2.10)$$

Kriteria Penolakan:

Tolak H_0 apabila $|t_{hitung}| > t_{\frac{\alpha}{2}, n-np}$ atau $p\text{-value} < \alpha$, artinya parameter telah signifikan.

2.2.9 Diagnostik Checking

Untuk melakukan diagnosis pada residual maka perlu dilakukan uji *white noise* dan uji normalitas.

a. Uji asumsi *White Noise*

Dikatakan *white noise* apabila ada sebuah garis dari variabel bebas yang tidak berhubungan atau berkorelasi dengan *mean*, *varians* dan *kovarians*. Pengujian *white noise* dapat dilihat pada plot ACF dan PACF. Statistik uji yang digunakan pada pengujian asumsi residual *white noise* yaitu statistik uji *Ljung-Box*. Dibawah ini merupakan bentuk pengujian *white noise* dengan menggunakan statistik uji *Ljung-Box* (Wei, 2006):

Hipotesis Uji *white noise* :

$H_0: \rho_1 = \rho_2 = \dots \rho_3$: parameter sama dengan 0 atau tidak signifikan
(residual sudah memenuhi syarat white noise)

H_1 : minimal ada 1 $\rho_1 \neq 0$ parameter tidak sama dengan 0 atau signifikan
(residual tidak memenuhi syarat white noise)

$$Q = n'(n' + 2) \sum_{k=1}^m \frac{r_k^2}{(n' + 2)} \quad (2.11)$$

Keterangan:

n' : banyaknya observasi dalam deret berkala

m : banyaknya lag yang diuji

r_k : koefisien autokorelasi pada periode ke-k

r_k : lag waktu

Taraf signifikansi : $\alpha = 5\%$

Kriteria pengambilan keputusan:

Membandingkan nilai Q hasil perhitungan dengan nilai $\chi^2_{(\alpha, ab), db = k - p - q}$.

Kriteria Pengujian :

Jika *p-value* < $\alpha = 5\%$ maka H_0 ditolak sedangkan jika *p-value* > $\alpha = 5\%$ maka H_0 diterima.

b. Uji normalitas

Uji normalitas bertujuan untuk mengetahui data yang diuji apakah memenuhi asumsi normalitas atau tidak. Cara untuk mengetahui apakah data

tersebut normal atau tidak yaitu dengan uji *Kolmogorov-Smirnov*. Berikut ini merupakan bentuk pengujian uji normalitas dengan *Komogorov-Smirnov*:

H_0 = Residual berdistribusi normal

H_1 = Residual tidak berdistribusi normal

Statistik Uji:

$$D = \max |S(x) - F_0(x)| \quad (2.12)$$

Taraf signifikansi : $\alpha = 5 \%$

Kriteria pengujian : Tolak H_0 jika $D \geq d_{1-\alpha}$ atau nilai probabilitas p-value $\leq \alpha$ dengan $d_{1-\alpha}$ merupakan nilai kritis yang diperoleh dari tabel *Kolmogorov-Smirnov*.

2.2.10 Neural Network

Neural Network adalah sistem informasi yang mempunyai ciri-ciri mirip dengan jaringan syaraf pada manusia. Menurut Kusumadewi (2003) proses informasi otak manusia terdiri dari berbagai *neuron* yang mempunyai tugas sederhana dan saling terhubung antara satu dengan yang lainnya. Menurut Haykin (1994) jaringan sistem syaraf yaitu mesin yang dibuat untuk memodelkan sistem kerja pada otak dalam menyelesaikan beberapa tugas dan fungsi tertentu. Beberapa pengertian *Neural Network* menurut Pandjaitan (2007) adalah sebagai berikut:

1. *Neural Network* adalah model yang mengandung elemen yang cukup besar yang diorganisasikan ke dalam *layer*

2. NN yaitu suatu teknik berbasis komputer yang digunakan untuk memodelkan sistem syaraf biologis
3. NN yaitu sistem komputer yang terdapat pada model syaraf biologis dan digunakan untuk memodelkan syaraf pada masukan *input*

Berbagai macam pengertian di atas, maka dapat disimpulkan bahwa *Neural Network* adalah teknik komputasi informasi dengan meniru cara kerja otak manusia yang disajikan dalam bentuk model matematika dan digunakan untuk menyelesaikan berbagai macam permasalahan.

2.2.11 Artificial Neural Network

Artificial Neural Network yaitu konsep rekayasa pengetahuan dengan mengadopsi sistem syaraf dalam bidang kecerdasan buatan. Artificial Neural Network disebut juga salah satu kesatuan dalam proses informasi yang tersebar dan bekerja dengan bersamaan Menurut Hecth-Nielsen (1989). ANN adalah suatu metode yang fleksibel yang digunakan untuk mengatasi berbagai masalah komputasi masalah nonlinier. Menurut Gupta (2000) ANN dengan pendekatan nonlinier dapat memberikan nilai akurasi yang tinggi pada saat memodelkan pola data yang kompleks. Menurut Haykin, S (1994) *Artificial Neural Network* adalah prosesor yang tersebar dan mampu menyimpan pengalaman dan pengetahuan, sehingga mirip kinerja otak asli. Dalam jaringan syaraf tiruan ada beberapa istilah yang sering ditemui yaitu :

1. *Neuron* atau Unit : yaitu sel syaraf tiruan yang berasal dari pengolahan jaringan syaraf tiruan.

2. Jaringan : kumpulan *neuron* yang saling terhubung dan membentuk lapisan.
3. Lapisan tersembunyi : lapisan yang tidak secara langsung berinteraksi dengan dunia luar.
4. *Input* : Nilai *input* dari penelitian yang akan diproses menjadi *output*
5. *Output* : Sebuah solusi dari hasil nilai *input*
6. Bobot : Nilai matematis dari sebuah koneksi antar *neuron*
7. Fungsi Aktivasi : fungsi yang digunakan untuk memperbarui nilai – nilai bobot per-iterasi dari semua nilai *input*.
8. Fungsi aktivasi sederhana : mengkalikan *input* dengan bobotnya dan kemudian menjumlahkannya berbentuk linier atau tidak linier dan sigmoid.
9. Paradigma pembelajaran : bentuk pembelajaran, *supervised learning*, atau *unsupervised learning*.

Arsitektur jaringan syaraf tiruan digunakan untuk menjelaskan arah datangnya sinyal atau yang berada di dalam jaringan. *Artificial Neural Network* (ANN) terdiri dari sekumpulan *neuron–neuron* yang saling berikatan satu dengan yang lainnya. ANN tergantung pada *arsitektur, training, testing*, dan algoritma pada jaringan syaraf tiruan.

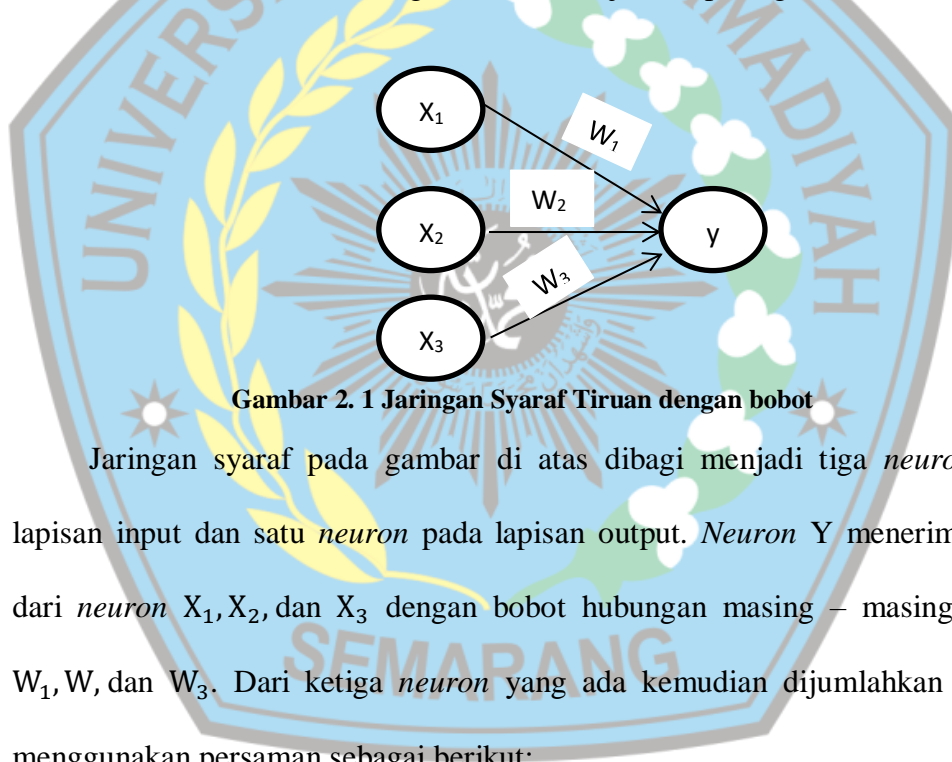
Arsitektur Jaringan

Arsitektur jaringan dalam hal ini *neuron* dalam JST sering diganti namanya menjadi simpul. Sinyal yang disampaikan dikirimkan melalui penghubung yang sering disebut dengan bobot. Setiap perubahan bobot dapat

dilakukan dengan berbagai cara tergantung dengan jenis algoritma pelatihan yang digunakan. Menurut Fausett (2004) ada tiga komponen yang berperan penting yaitu :

1. Pola hubungan antar neuron yang disebut arsitektur
2. Metode penentuan bobot penghubung yang disebut pelatihan
3. Fungsi aktivasi yang dijalankan masing-masing neuron pada input jaringan untuk menentukan input.

Arsitektur ANN sederhana dengan bobot ditunjukkan pada gambar dibawah ini:



Gambar 2. 1 Jaringan Syaraf Tiruan dengan bobot

Jaringan syaraf pada gambar di atas dibagi menjadi tiga *neuron* pada lapisan input dan satu *neuron* pada lapisan output. *Neuron Y* menerima input dari *neuron* X_1 , X_2 , dan X_3 dengan bobot hubungan masing – masing adalah W_1 , W_2 , dan W_3 . Dari ketiga *neuron* yang ada kemudian dijumlahkan dengan menggunakan persamaan sebagai berikut:

$$y_{in} = \sum_{i=1}^n x_i w_i \quad (2.13)$$

2.2.12 Arsitektur Artificial Neural Network

Neural Network memiliki *neuron-neuron* yang tersusun dalam lapisan (*layer*) dan mempunyai pola hubungan yang cukup baik dalam lapisan dan diluar lapisan. Menurut Rofik (2017) *Neural Network* terdiri dari beberapa lapisan yaitu

1. *Input Layer*

Input layer yaitu lapisan yang menerima masukan/sinyal dari luar jaringan.

2. *Hidden Layer*

Lapisan yang menerima dan mengirim sinyal ke jaringan syaraf dan terletak di dalam satu atau beberapa lapisan dan tidak berhubungan langsung dengan keadaan di luar jaringan.

3. *Output Layer*

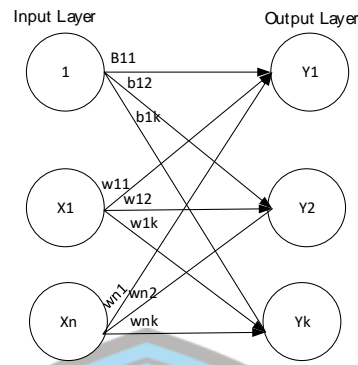
Lapisan yang menghasilkan output dari jaringan.

NN dibedakan menjadi beberapa jaringan berdasarkan jumlah layer yaitu sebagai berikut:

1. Jaringan Lapis Tunggal (*Single Layer*)

Menurut Fausett (1994) jaringan disebut dengan lapis tunggal karena jaringan tersebut tidak memiliki lapisan yang tersembunyi dari *neuron* atau hanya mempunyai satu lapisan bobot koneksi. Ciri dari *single layer* adalah hanya memiliki satu *layer input* dan satu *layer output*.

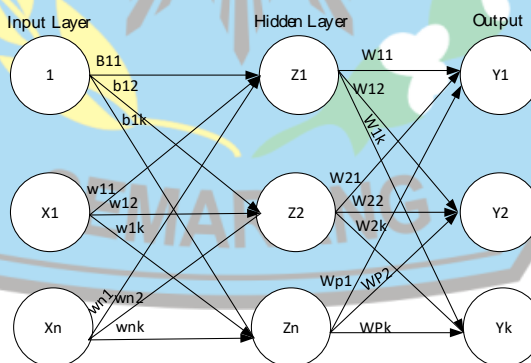
Arsitektur dari jaringan single layer digambarkan sebagai berikut:



Gambar 2. 2 Jaringan Single Layer

2. Jaringan Lapis Jamak (*Multiplayer Net*)

Jaringan lapis jamak adalah pengembangan dari jaringan single layer (Siang,2005). Jaringan lapis jamak ini terdiri dari lapisan input, lapisan tersembunyi, dan lapisan output. Lapisan tersembunyi atau hidden layer terletak antara lapisan input dan lapisan output. Pada jaringan ini paling sedikit memiliki satu lapisan tersembunyi. Arsitektur dari jaringan *multi layer* digambarkan sebagai berikut:



Gambar 2. 3 Jaringan Multi Layer

Pada metode Neural Network mempunyai dua tahapan pemrosesan informasi yaitu tahap pelatihan dan pengujian sebagai berikut:

a. Metode *Training*

Metode *training* atau pengaturan nilai bobot adalah karakteristik yang sangat penting dalam jaringan neural network (Fausset, 1994:15).

Ada dua jenis metode *training* pada *neural network* yaitu:

1. Pelatihan Terawasi

Pelatihan ini dilaksanakan dengan urutan vektor pelatihan terkait dengan vektor output. Nilai *output* kemudian dicocokkan pada nilai target, apabila terjadi perbedaan maka akan muncul *error*. Tujuan dari pelatihan terawasi ini untuk memperoleh hasil pemetaan vektor *input* ke nilai *output* sehingga dihasilkan *output* yang sesuai keinginan. Untuk mencapai tujuan maka perubahan nilai bobot ini dilakukan.

2. Pelatihan Tak Terawasi

Pada pelatihan tak terawasi ini jaringan syaraf mengatur semua kinerja pada dirinya sendiri. Pada proses ini dilakukan pengujian terhadap pola masukan yang belum diuji atau pada data *testing*. Dengan proses ini maka diharapkan dapat memperoleh hasil *error* minimum pada bobot-bobot hasil *training* atau pelatihan.

b. Metode *Testing*

Pada metode testing ini diterapkan pada masalah yang dimana output pada sebelumnya belum diketahui. Metode ini mengubah nilai bobot yang dilakukan tanpa menggunakan output yang diinginkan. Menurut Warsito (2009) tujuan dari pembelajaran ini yaitu untuk mengelompokkan input yang sama.

2.2.13 Feed Forward Neural Network (FFNN)

Metode Feed Forward Neural Network (FFNN) yaitu metode pemodelan Neural Network yang paling sering digunakan dalam pemodelan data deret waktu. FFNN terdiri dari satu input layer, satu atau lebih hidden layer, dan satu output layer. Persamaan umum FFNN sebagai berikut (Kusumadewi,2014) :

$$Y_t = f^0 \left[\sum_{j=1}^q \{ w_j^o f_j^h \left[\sum_{i=1}^p w_{ij}^h y_{t-i} + b_j^h \right] + b^0 \} \right] \quad (2.14)$$

Dimana :

Y_t : Nilai dugaan variabel *output layer*

f^0 : fungsi aktivasi *neuron* pada *output layer* (linier)

w_j^o : Bobot *neuron* ke j pada *hidden layer* yang menuju ke *output layer*

f_j^h : fungsi aktivasi *neuron* pada *hidden layer*

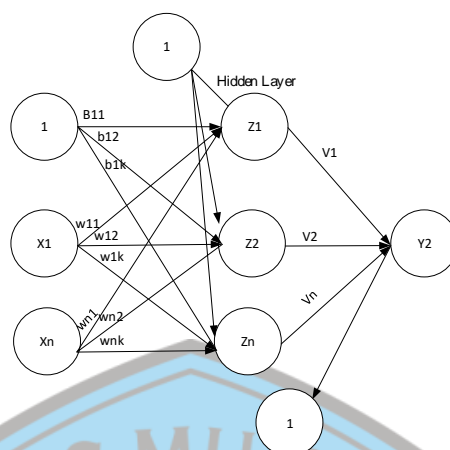
w_{ij}^h : Bobot *neuron* i pada *input layer* yang menuju ke *neuron* j pada *hidden*

y_{t-i} : Variabel *input*

b_j^h : Bias pada *hidden layer*

b^0 : Bias pada *ouput layer*

Arsitektur jaringan Feed Forward Neural Network (FFNN) pada dasarnya tersusun dari tiga layer yaitu input layer, hidden layer, dan output layer (Siang, 2005). Berikut merupakan gambar arsitektur jaringan Feed Forward Neural Network (FFNN):



Gambar 2. 4Ilustrasi Feed Forward Neural Network

1. Fungsi Aktivasi

Fungsi aktivasi digunakan untuk menentukan apakah sinyal dari input neuron akan diteruskan atau tidak (Siang,2005). Dalam jaringan syaraf tiruan, fungsi aktivasi digunakan untuk menentukan keluaran suatu neuron. Argumen fungsi aktivasi adalah net masukan (kombinasi linier masukan dan bobotnya). Jika $net = \sum x_i w_i$, maka fungsi aktivasinya yaitu $f(net) = f(\sum x_i w_i)$. Beberapa fungsi aktivasi yang sering dipakai yaitu sebagai berikut (Siang,2005):

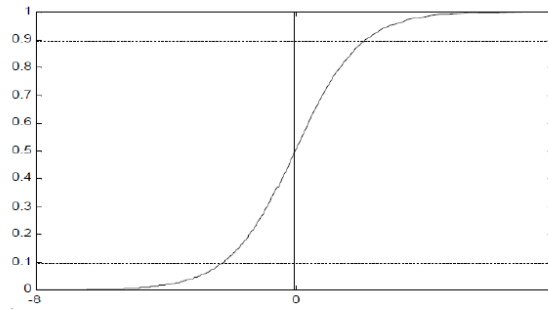
a Fungsi Sigmoid Biner

Fungsi sigmoid yang sering digunakan karena nilai fungsinya yang terletak diantara 0 dan 1.

$$f(x) = \frac{1}{(1 + e^{-x})} \quad (2.15)$$

Dengan turunannya yaitu sebagai berikut:

$$f'(x) = f(x)(1 - f(x)) \quad (2.16)$$



(Sumber : Fausett,L.,1994)

Gambar 2. 5 Grafik Fungsi Aktivasi Sigmoid Biner

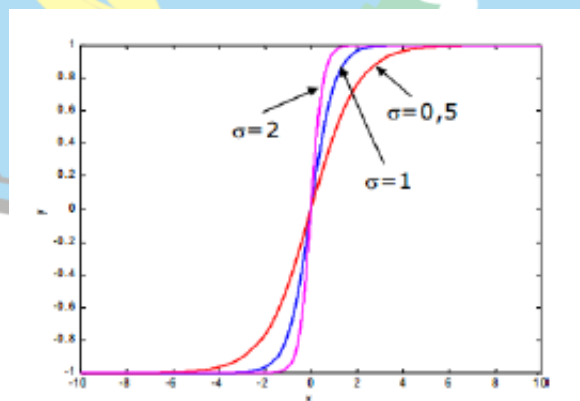
b Fungsi Sigmoid Bipolar

Fungsi lain yang dibentuk selain sigmoid biner yaitu sigmoid bipolar yang memiliki range $(-1,1)$ dengan rumus sebagai berikut:

$$f(x) = \frac{2}{(1 + e^{-x})} - 1 \quad (2.17)$$

Dengan fungsi turunannya sebagai berikut:

$$f'(x) = \frac{1 + f(x))(1 - f(x))}{2} \quad (2.18)$$



(Sumber : Fausett,L.,1994)

Gambar 2. 6 Grafik Fungsi Sigmoid Bipolar

2. Bias dan *Treshold*

Pada Neural Network ditambahkan unit masukan (input) yang memiliki nilai yang selalu sama dengan 1. Unit yang demikian itu disebut dengan bias. Bias memiliki fungsi yang digunakan untuk mengubah nilai *threshold* menjadi=0 (bukan=alpha). Jika melibatkan bias, maka keluaran unit penjumlahannya yaitu (Siang,2005):

$$net = b + \sum_{i=1}^n x_i w_i \quad (2.19)$$

Fungsi aktivasi *threshold* menjadi:

$$f(net) = 1 \text{ jika } x \geq 0 ; -1 \text{ jika } x < 0 \quad (2.20)$$

3. Normalisasi Data

Normalisasi data digunakan untuk standarisasi semua data yang akan digunakan pada penelitian agar berada pada rentang range tertentu. Normalisasi dilakukan karena nilai input yang tidak sama dengan data input lainnya maka dilakukan normalisasi data untuk memperoleh nilai normalisasi yang kecil pada rentang tertentu. Fungsi aktivasi yang digunakan disini yaitu fungsi sigmoid biner karena memiliki nilai range antara 0 sampai dengan 1. Dibawah ini merupakan perhitungan normalisasi data (Siang,2005):

$$Y't = \frac{0.8(Y_t - \min(x))}{(\max(x) - \min(x))} + 0.1 \quad (2.21)$$

Dimana :

\hat{Y}_t : Data ke-t dari hasil normalisasi

Y_t : Data ke-t sebelum normalisasi

Min (x) : Nilai minimum data sebelum di normalisasi

4. Denormalisasi Data

Denormalisasi data yaitu kebalikan dari proses normalisasi data.

Denormalisasi data bertujuan untuk membangkitkan nilai yang telah dinormalisasi menjadi nilai asli (awal). Dibawah ini merupakan perhitungan proses denormalisasi data (Siang,2005):

$$\hat{Y}_t = \frac{(\hat{Y}_t - 0.1)(\max(x) - \min(x))}{0.8} + \min(x) \quad (2.22)$$

5. Parameter *Resilient Backpropagation* (Rprop)

Perubahan pada bobot diinisialisasikan dengan parameter Δ_0 . Besarnya parameter Δ_0 yang dipilih tidak berpengaruh penting terhadap waktu konvergensi yang diperlukan. Waktu Δ_0 konvergensi akan tetap cepat meskipun Δ_0 yang terlalu besar atau kecil. Δ_0 yang baik digunakan yaitu sebesar 0.1. Untuk mencegah bobot yang melebihi batas maksimum maka perubahan bobot akan ditentukan sama dengan maksimum perubahan bobot. Pemilihan faktor penurunan η^- dan Pemilihan faktor kenaikan η^+ yaitu dengan pertimbangan sebagai berikut: jika kenaikan sebuah nilai minimum maka nilai pembaruan sebelumnya akan terlalu besar.

2.2.14 Model Feedforward Backpropagation

Arsitektur jaringan algoritma *Backpropagation* pada umumnya memiliki susunan yang terdiri dari tiga layer yaitu input layer, hidden layer, dan output layer (Siang, 2005). *Feed Forward Backpropagation* yaitu model neural network yang sering digunakan. Algoritma *Feed-Forward Backpropagation* disebut sebagai propagasi balik karena ketika jaringan diberikan pola masukan sebagai pola yang memiliki pelatihan maka pola tersebut menuju ke unit-unit pada lapisan tersembunyi (*hidden layer*) yang diteruskan ke unit lapisan keluaran (*output*). Unit lapisan keluaran memberikan tanggapan yang disebut sebagai keluaran jaringan, dimana saat keluaran jaringan tidak sama dengan keluaran yang diharapkan maka keluaran tersebut akan mundur pada lapisan tersembunyi dan diteruskan ke unit pada lapisan masukan.

2.2.15 Pemodelan Hybrid ARIMAX-FFNN

ARIMAX merupakan model linier, sehingga model ini tidak dapat membaca pola data non-linier. Sebaliknya, FFNN merupakan model non linier yang baik akan tetapi hasilnya sulit untuk diinterpretasikan. Pemodelan *Hybrid* bertujuan untuk menambah keakuratan peramalan data dari model linier yang mudah diinterpretasikan dengan mengombinasikan model linier dengan model non-linier. Pemodelan Hybrid ditulis sebagai berikut (Zhang, 2003):

$$Z_t = L_t + R_t \quad (2.24)$$

Dimana :

L_t : Komponen linier

R_t : Komponen non-linier

2.2.16 Pemilihan Model Terbaik

Setelah mendapatkan hasil prediksi melalui proses yang dilakukan sebelumnya maka langkah selanjutnya yang dilakukakan yaitu melakukan evaluasi kerja atau pemilihan model terbaik peramalan. Pemilihan model terbaik pada penelitian ini yaitu menggunakan perhitungan MAPE (*Mean Absolute Percentage Error*).

Mean Absolute Percentage Error (MAPE) memiliki fungsi untuk menunjukkan tingkat akurasi peramalan. Perhitungan dengan nilai MAPE banyak digunakan karena menggunakan persentase dengan formula sebagai berikut :

$$MAPE = \sum_{t=1}^n \frac{(y_t - \hat{y}_t)}{y_t} \times 100\% \quad (2.25)$$

Dimana :

- n : jumlah periode peramalan
- y_t : Nilai sebenarnya pada waktu ke t
- \hat{y}_t : Nilai peramalan pada waktu ke t

Menurut Lewis (1982), nilai MAPE dapat dijelaskan dalam 4 kategori yaitu sebagai berikut:

1. <10% = Kemampuan peramalan “Sangat Baik”
2. 10-20% = Kemampuan Peramalan “Baik”
3. 20-50% = Kemampuan Peramalan “Cukup”
4. >50% = Kemampuan Peramalan “Buruk” atau “Gagal”