

## **BAB II**

### **TINJAUAN PUSTAKA DAN DASAR TEORI**

#### **2.1 Tinjauan Pustaka**

Metode jaringan syaraf tiruan dapat digunakan untuk melakukan peramalan harga seperti pada penelitian tentang peramalan harga pasar dari produk agrikultur menggunakan metode BP *neural network* dan *time series* yang mendapatkan hasil bahwa model *hybrid* dengan BP *neural network* dan *time series* lebih akurat untuk meramalkan harga pasar dari produk agrikultur (Shouhua Yu, 2009). Penelitian lainya juga tentang peramalan harga juga pernah dilakukan untuk meramalkan harga dari produk agrikultur agar berguna untuk petani, pembuat kebijakan dan industri agrikultur dengan hasil bahwa *neural network* lebih unggul daripada model linier lainya (Girish, 2013).

Algoritma backpropagation dapat digunakan untuk penelitian tentang penggunaan jaringan syaraf tiruan backpropagation untuk deteksi dini penyakit TBC yang menggunakan variabel batuk, sesak nafas, nyeri dada, demam sore/malam hari, penurunan nafsu makan, badan lemah yang menghasilkan tingkat akurasi 100% (Purnamasari, 2013). Selain itu algoritma backpropagation pernah digunakan dalam peramalan penyakit demam berdarah yang menggunakan variabel sakit kepala berat, sakit sendi dan otot, ruam yang mendapatkan hasil tingkat ketepatan dalam proses klasifikasi DBD adalah 74% dan proses diagnosa DBD mempunyai ketepatan 99% (Widodo, dkk., 2014). Selanjutnya penelitian tentang prediksi kanker serviks mempunyai tingkat akurasi 95,14%(Susanto, 2012).

*Neural network backpropagation* dapat digunakan untuk membangun sistem informasi prediksi harga ikan budidaya dengan variabel-variabel yang telah ditentukan. Sistem ini juga dapat dimanfaatkan sebagai acuan pelaku bisnis dalam pengambilan keputusan.

Penelitian pada bidang perikanan pernah dilakukan untuk membuat perencanaan dan pengambilan keputusan untuk evaluasi kebijakan manajemen perikanan guna mendapatkan hasil yang optimal (Kaplan dan Jerry, 2012). Hasil

perikanan memiliki kontribusi yang penting dalam kegiatan produksi lain maupun dalam bidang ketahanan pangan serta dapat membuka lapangan pekerjaan, dengan demikian diperlukan kebijakan untuk memajemen keberlanjutan sosial-ekonomi dari sektor perikanan (Belhabib dkk., 2018).

## **2.2 Dasar Teori**

Pada bab ini akan dibahas beberapa teori yang digunakan dalam penelitian yang pertama konsep dasar *time series*, kedua peramalan, ketiga jaringan syaraf tiruan, keempat macam-macam arsitektur jaringan, kelima macam-macam fungsi aktivasi, keenam *backpropagation*, ketujuh fungsi aktivasi *backpropagation*, kedelapan alogaritma pelatihan *backpropagation*, kesembilan alogaritma nguyen widrow, kesepuluh perhitungan *error* dan yang kesebelas korelasi.

### **2.2.1 Konsep dasar *time series***

*Time Series* merupakan kejadian-kejadian yang terjadi berdasarkan waktu-waktu tertentu secara berurutan. *Forecasting* dalam *time series* merupakan prediksi untuk memperkirakan kejadian-kejadian yang akan terjadi di masa yang akan datang berdasarkan data-data sebelumnya (Wei, 2006).

Analisa data *time series* adalah analisa yang menerangkan dan mengukur berbagai perubahan atau perkembangan data selama satu periode (Hasan, 2002). Analisa *time series* dilakukan untuk memperoleh pola data *time series* dengan menggunakan data masa lalu yang akan digunakan untuk meramalkan suatu nilai pada masa yang akan datang. Dalam *time series* terdapat empat macam tipe pola data, yaitu *horizontal*, musiman (*Seasonal*), *trend*, *cyclical*.

### **2.2.2 Peramalan**

Peramalan adalah proses memperkirakan nilai di masa mendatang dengan menggunakan data yang ada di masa lampau (Hyndman, 2014). Data di masa lampau secara sistematis dikombinasikan dan diolah untuk memperkirakan suatu nilai di masa mendatang. Terdapat dua pendekatan untuk melakukan peramalan, yaitu dengan pendekatan kualitatif dan pendekatan kuantitatif. Metode peramalan kualitatif menggabungkan faktor-faktor seperti intuisi pengambilan keputusan,

emosi, pengalaman pribadi sedangkan metode peramalan kuantitatif yang menggunakan satu atau lebih model matematis dengan data masa lalu dan variabel sebab akibat untuk meramalkan permintaan. Pada dasarnya metode peramalan kuantitatif dibagi menjadi dua, yaitu model deret waktu (*time series*), dan model kausal.

### 2.2.3 Jaringan syaraf tiruan

Jaringan syaraf tiruan adalah sistem pemroses informasi yang memiliki karakteristik mirip dengan jaringan syaraf biologi (Siang, 2005). Jaringan syaraf tiruan dibentuk sebagai generalisasi model matematika dari jaringan syaraf biologi, dengan asumsi bahwa:

- a. Pemrosesan informasi terjadi pada banyak elemen sederhana (*neuron*)
- b. Sinyal dikirimkan diantara neuron-neuron melalui penghubung-penghubung
- c. Penghubung antar neuron memiliki bobot yang akan memperkuat atau memperlemah sinyal
- d. Untuk menentukan *output*, setiap neuron menggunakan fungsi aktivasi yang dikenakan pada jumlah *input* yang diterima. Besarnya *output* ini selanjutnya dibandingkan dengan suatu batas ambang.

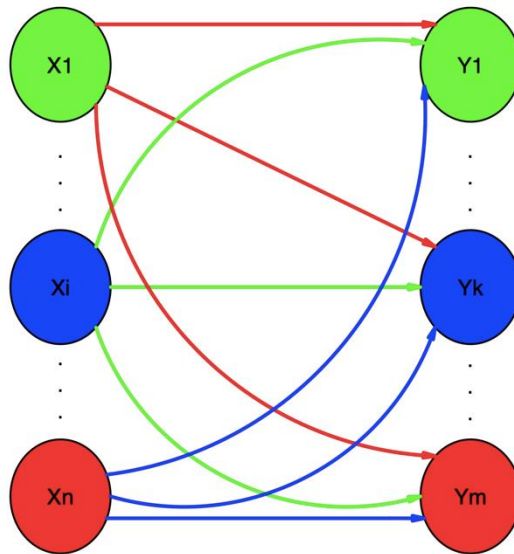
Jaringan syaraf tiruan ditentukan oleh 3 hal yaitu arsitektur jaringan, metode untuk menentukan bobot penghubung, serta fungsi aktivasi.

### 2.2.4 Macam-macam arsitektur jaringan

Dalam jaringan syaraf tiruan terdapat beberapa model arsitektur jaringan. Berikut ini beberapa model dari arsitektur jaringan syaraf tiruan :

- a. Jaringan *layer* tunggal (*single layer network*)

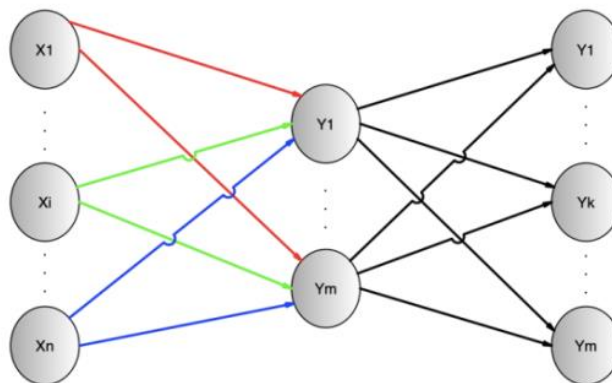
Pada jaringan *layer* tunggal, sekumpulan input neuron dihubungkan langsung dengan sekumpulan outputnya. Gambar 2.1 adalah contoh arsitektur jaringan *layer* tunggal dengan  $n$  unit *input* ( $x_1, x_2, \dots, x_i, \dots, x_n$ ) dan  $m$  buah unit *output* ( $y_1, y_2, \dots, y_k, \dots, y_m$ ).



Gambar 2.1 Jaringan layer tunggal (Siang, 2015)

b. Jaringan layer jamak (*multi layer network*)

Jaringan *layer* jamak merupakan perluasan dari *layer* tunggal. Dalam jaringan *layer* jamak terdapat unit-unit selain unit *input* dan *output* yang disebut *layer* tersembunyi. *Layer* tersembunyi terletak diantara *input* dan *output*. *Layer* tersembunyi dapat berjumlah lebih dari satu pada suatu arsitektur jaringan. Gambar 2.2 adalah contoh jaringan *layer* jamak dengan n buah *unit input* ( $x_1, x_2, \dots, x_i, \dots, x_n$ ), sebuah *layer* tersembunyi yang terdiri dari p buah *unit* ( $z_1, \dots, z_j, \dots, z_p$ ) dan m buah *unit output* ( $y_1, y_2, \dots, y_k, \dots, y_m$ ).



Gambar 2.2 Jaringan layer jamak (Siang, 2005)

c. Jaringan *reccurent*

Pada jaringan *reccurent* terdapat neuron output yang memberikan sinyal pada unit input. Jaringan *reccurent* sering juga disebut *feedback loop*.

### 2.2.5 Macam-macam fungsi aktivasi

Fungsi aktivasi adalah fungsi yang digunakan untuk menentukan keluaran suatu neuron pada jaringan syaraf tiruan. Argumen fungsi aktivasi adalah net masukan yang didapat dari kombinasi linier masukan dan bobotnya. Beberapa fungsi aktivasi yang sering dipakai antara lain:

a. Fungsi *threshold*

$$f(x) = \begin{cases} 1 & \text{jika } x \geq a \\ -1 & \text{jika } x < a \end{cases} \dots\dots\dots (2.1)$$

b. Fungsi *sigmoid*

$$f(x) = \frac{1}{1 + e^{-x}} \dots\dots\dots (2.2)$$

c. Fungsi identitas

$$f(x) = x \dots\dots\dots (2.3)$$

### 2.2.6 Backpropagation

*Backpropagation* merupakan salah satu model jaringan syaraf tiruan. *Backpropagation* melatih jaringan untuk mendapatkan keseimbangan antara kemampuan jaringan untuk mengenali pola yang digunakan selama pelatihan serta kemampuan jaringan untuk memberikan respon yang benar terhadap pola masukan yang serupa tapi tidak sama dengan pola yang dipakai selama pelatihan (Siang, 2005).

### 2.2.7 Fungsi aktivasi *backpropagation*

Fungsi aktivasi yang digunakan dalam *backpropagation* harus memenuhi beberapa syarat yaitu kontinu, terdiferensial dengan mudah dan merupakan fungsi yang tidak turun. Terdapat 2 fungsi aktivasi yang dapat digunakan dalam *backpropagation* yaitu fungsi *sigmoid biner* dan fungsi *sigmoid bipolar*. Fungsi

*sigmoid biner* memiliki *range* (0,1) dengan rumus:

$$f(x) = \frac{1}{1 + e^{-x}} \dots\dots\dots (2.4)$$

dengan turunan

$$f'(x) = f(x)(1 - f(x)) \dots\dots\dots (2.5)$$

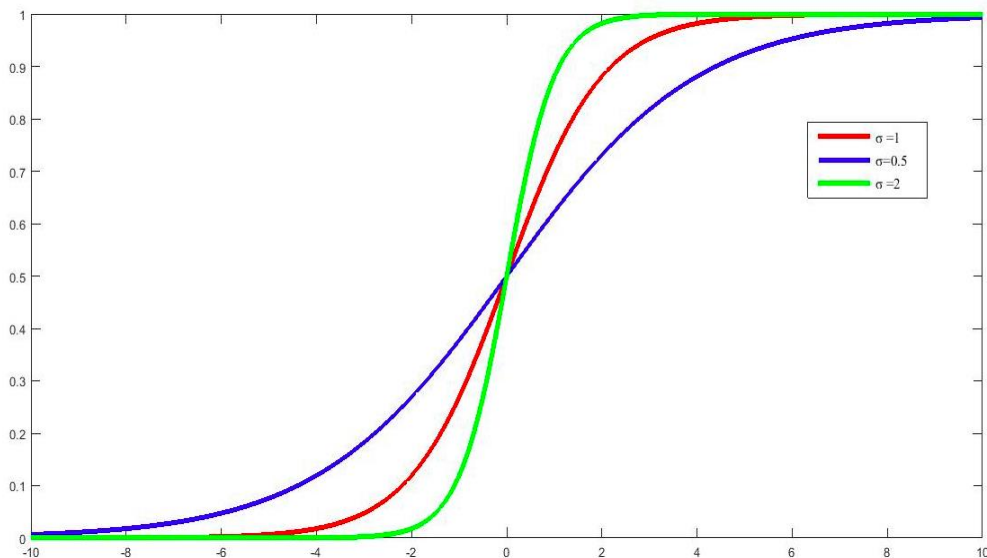
Fungsi *sigmoid bipolar* memiliki *range* (-1,1) dengan rumus:

$$f(x) = \frac{2}{1 + e^{-x}} - 1 \dots\dots\dots (2.6)$$

dengan turunan

$$f'(x) = \frac{(1 + f(x))(1 - f(x))}{2} \dots\dots\dots (2.7)$$

Fungsi *sigmoid* memiliki nilai maksimum 1, maka untuk pola yang targetnya lebih besar dari 1 pola masukan dan keluaran harus terlebih dahulu ditransformasi sehingga semua polanya memiliki *range* yang sama seperti fungsi *sigmoid* yang dipakai. Fungsi *sigmoid biner* ditunjukkan pada Gambar 2.3.



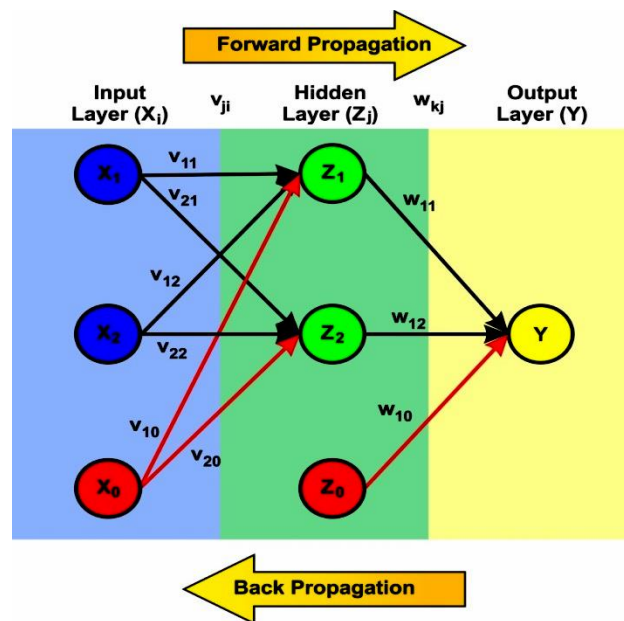
Gambar 2.3 Fungsi aktivasi *sigmoid biner*

### 2.2.8 Algoritma pelatihan *backpropagation*

Pelatihan *backpropagation* meliputi 3 fase. Fase pertama adalah fase maju. Pola masukan dihitung maju mulai dari layer masukan hingga *layer* keluaran

menggunakan fungsi aktivasi yang ditentukan. Fase kedua adalah fase mundur. Selisih antara keluaran jaringan dengan target yang diinginkan merupakan kesalahan yang terjadi. Kesalahan tersebut dipropagasikan mundur, dimulai dari garis yang berhubungan langsung dengan unit-unit di *layer* keluaran. Fase ketiga adalah modifikasi bobot untuk menurunkan kesalahan yang terjadi.

Ketiga fase tersebut diulang-ulang terus hingga kondisi penghentian dipenuhi. Umumnya kondisi penghentian yang sering dipakai adalah jumlah iterasi atau kesalahan. Iterasi akan dihentikan jika jumlah iterasi yang dilakukan sudah melebihi jumlah maksimum iterasi yang ditetapkan, atau jika kesalahan yang terjadi sudah lebih kecil dari batas toleransi yang diijinkan. Algoritma perambatan balik (*Backpropagation*) dengan satu *layer* tersembunyi dengan aktivasi *sigmoid biner* seperti yang ditunjukkan pada Gambar 2.4.



Gambar 2.4 Arsitektur dan langkah JST *Backpropagation*

Berdasarkan Gambar 2.4 tersebut terdapat beberapa proses dalam *backpropagation*. Berikut proses-proses yang dilakukan dalam *backpropagation* :

- a. Proses Perhitungan Algoritma Pelatihan Propagasi Balik
  - 1) Langkah 0 : Inisialisasi semua bobot dengan bilangan acak kecil, epoch = 1, tentukan laju pembelajaran ( $\alpha$ ), tentukan jumlah unit pada *layer* tersembunyi

(p) serta tentukan kondisi penghentian. Kondisi penghentian berupa maksimum max epoch dan error target.

- 2) Langkah 1: Jika  $epoch \neq max\ epoch$  dan  $error\ target < MSE$ , lakukan langkah 2–9.
- 3) Langkah 2: Untuk setiap pasang data pelatihan (1 sampai dengan a dimana a adalah jumlah data pelatihan), lakukan langkah 3–8

**Fase I: Propagasi maju**

- 4) Langkah 3 : Tiap unit masukan menerima sinyal dan meneruskannya ke unit tersembunyi di atasnya.
- 5) Langkah 4 : Hitung semua keluaran di unit tersembunyi  $z_j$  ( $j = 1, 2, \dots, p$ )

$$z\_net_j = v_{j0} + \sum_{i=1}^n x_i v_{ji} \dots\dots\dots (2.8)$$

dengan :

$z\_net_j$  = nilai yang menuju unit tersembunyi (*hidden layer*)

$v_{j0}$  = nilai bobot bias, ( $j= 1,2, \dots,p$ )

$x$  = variabel ( $i = 1,2, \dots,p$ )

$v_j$  = nilai bobot unit tersembunyi (*hidden layer*)

$n$  = banyaknya unit yang menuju unit tersembunyi (*hidden layer*)

jika menggunakan fungsi *aktivasi sigmoid*, hitung *output* dengan

$$z_j = f(z\_net_j) = \frac{1}{1+e^{-z\_net_j}} \dots\dots\dots (2.9)$$

dengan:

$z_j$  = nilai unit tersembunyi (*hidden layer*) ,  $e$  = bilangan *euler* ( $e = 2,718$ )

- 6) Langkah 5: hitung semua keluaran jaringan di unit  $y_k$  ( $k = 1, 2 \dots m$ )

$$y\_net_k = w_{j0} + \sum_{j=1}^p z_j w_{kj} \dots\dots\dots (2.10)$$

dengan :

$y\_net_k$  = nilai yang menuju  $y$  (*output*)

$w_{j0}$  = nilai bobot bias = 1, ( $j= 1,2, \dots,p$ )

$z_j$  = nilai unit tersembunyi (*hidden layer*)

$p$  = banyaknya unit yang menuju ke  $y$  (*output*)



$w_{kj}$  = nilai bobot yang menuju unit  $y$  (*output*)

jika kita menggunakan fungsi *aktivasi sigmoid*, hitung output dengan:

$$y_k = f(y\_net_k) = \frac{1}{1+e^{-(y\_net_k)}} \dots\dots\dots (2.11)$$

$y_k$  = nilai keluaran (*output*)

**Fase II : Propagasi Mundur**

- 7) Langkah 6 : hitung faktor  $\delta$  unit keluaran berdasarkan kesalahan disetiap unit keluaran  $y_k$  ( $k=1,2 \dots , m$ )

$$\delta_k = (t_k - y_k)f'(y\_net_k) = (t_k - y_k)y_k(1 - y_k) \dots\dots\dots (2.12)$$

dengan :

$\delta_k$  (delta k) = faktor unit keluaran  $y$  (*output*) , ( $k = 1,2, \dots p$ )

$t_k$  = nilai target

$y_k$  = nilai keluaran (*output*)

hitung suku perubahan bobot  $W_{kj}$  dengan laju percepatan  $\alpha$

$$\Delta w_{kj} = \alpha \delta_k z_j \dots\dots\dots (2.13)$$

$\Delta w_{kj}$  (delta  $w_{kj}$ ) = nilai perubahan bobot yang menuju unit  $y$ (*output*)

$k = 1, 2, \dots, m ; j = 1, 2, \dots , p$

$\alpha$  = laju percepatan (nilai ditentukan acak)

- 8) Langkah 7 : hitung faktor  $\delta$  unit keluaran berdasarkan kesalahan disetiap unit tersembunyi  $z_j$  ( $j= 1,2, \dots, p$ )

$$\delta\_net_j = \sum_{k=1}^m \delta_k w_{kj} \dots\dots\dots (2.14)$$

$$\delta_j = \delta\_net_j f'(z\_net_j) = \delta\_net_j z_j(1 - z_j) \dots\dots\dots (2.15)$$

hitung suku perubahan bobot  $v_{ji}$  dengan laju percepatan  $\alpha$

$$\Delta v_{ji} = \alpha \delta_j x_i \dots\dots\dots (2.16)$$

**Fase III : Perubahan Bobot**

- 9) Langkah 8 : hitung semua perubahan bobot

Perubahan bobot garis yang menuju ke unit keluaran

$$w_{kj}(baru) = w_{kj}(lama) + \Delta w_{kj} \dots\dots\dots (2.17)$$

$k = 1, 2, \dots, m ; j = 1, 2, \dots , p$

perubahan bobot garis yang menuju ke unit tersembunyi:

$$v_{kj}(\text{baru}) = v_{kj}(\text{lama}) + \Delta v_{ji} \dots \dots \dots (2.18)$$

$$j = 1, 2, \dots, p ; i = 0, 1, \dots, n$$

10) Langkah 9: perbarui nilai epoch

$$\text{epoch} = \text{epoch} + 1 \dots \dots \dots (2.19)$$

kemudian, keluaran jaringan dibandingkan dengan target, hitung *error* E menggunakan:

$$E = t - y_k \dots \dots \dots (2.20)$$

untuk menghitung *mean squared error* (MSE), dengan rumus:

$$MSE = \frac{\sum_{k=1}^{\alpha} (t_k - y_k)^2}{\alpha} \dots \dots \dots (2.21)$$

### 2.2.9 Alogaritma Nguyen Widrow

Bobot awal akan mempengaruhi apakah jaringan mencapai titik minimum lokal atau global, dan seberapa cepat konvergensinya. Nguyen dan Widrow (2010) mengusulkan cara membuat inisialisasi bobot ke unit tersembunyi sehingga menghasilkan iterasi yang lebih cepat. Algoritma Nguyen Widrow adalah sebagai berikut (Siang, 2005) :

1. Inisialisasi semua bobot (  $v_{ji}(\text{lama})$  ) dengan bilangan acak dalam interval [-0.5, 0.5]

2. Hitung

$$\|v_{ji}\| = \sqrt{v_{j1}^2 + v_{j2}^2 + \dots + v_{jn}^2} \dots \dots \dots (2.22)$$

3. Bobot digunakan sebagai inisialisasi:

$$v_{ji} = \frac{\beta v_{ji}(\text{lama})}{\|v_{ji}\|} \dots \dots \dots (2.23)$$

Dimana  $\beta$  adalah faktor skala dengan rumus :

$$\beta = 0.7^n \sqrt{p} \dots \dots \dots (2.24)$$

4. Bobot bias yang dipakai sebagai inisialisasi (  $v_{j0}$  ) adalah bilangan acak dalam interval  $[-\beta, \beta]$

### 2.2.10 Perhitungan *Error*

Pada dasarnya, pengawasan peramalan dilakukan dengan membandingkan hasil peramalan dengan kenyataan yang terjadi. Penggunaan teknik peramalan yang menghasilkan penyimpangan/kesalahan terkecil adalah teknik peramalan yang paling baik untuk digunakan. Tsaur (2012) menggunakan metode MAPE (*Mean Absolute Percentage Error*) untuk mengetahui besarnya penyimpangan yang terjadi pada data hasil peramalan terhadap data aktual. MAPE merupakan rata-rata dari keseluruhan persentase kesalahan (selisih) antara data aktual dengan data hasil peramalan. Ukuran akurasi dicocokkan dengan data *time series*, dan ditunjukkan dalam persentase. MAPE yang digunakan untuk ukuran keakuratan adalah sebagai berikut:

$$MAPE = \frac{1}{n} \sum_{t=1}^n \frac{|Y(t) - F'(t)|}{Y(t)} * 100\% \dots\dots\dots (2.25)$$

Kriteria keakuratan MAPE menurut Chang, Wang dan Liu (2007) disajikan pada Tabel 2.1.

Tabel 2.1 Tingkat akurasi peramalan

Peramalan sangat akurat	MAPE < 10%
Peramalan akurat	MAPE 10%-20%
Peramalan cukup	MAPE 20%-50%
Peramalan tidak akurat	MAPE > 50%

Selain itu terdapat metode lain yaitu metode MSE (*Mean Square Error*). Berikut adalah rumus untuk MSE:

$$MSE = \frac{1}{n} \sum_{t=1}^n (Y(t) - F'(t))^2 \dots\dots\dots (2.26)$$

Kriteria keakuratan MSE dianggap baik jika memiliki nilai yang semakin kecil dari tingkat kesalahannya.

### 2.2.11 Korelasi

Korelasi merupakan teknik analisis yang digunakan untuk mengukur arah kedekatan hubungan antara dua variabel atau lebih. Arah kedekatan dinyatakan dalam bentuk hubungan positif atau negatif (Grech, 2018). Hubungan dua variabel atau lebih dikatakan hubungan positif, jika nilai satu variabel ditingkatkan, maka akan meningkatkan nilai variabel lain (nilai X naik maka nilai Y naik), dan sebaliknya jika nilai satu variabel diturunkan, maka nilai variabel yang lain akan turun (nilai X turun maka nilai Y turun). Hubungan dua variabel atau lebih dikatakan hubungan negatif, jika nilai satu variabel dinaikkan, maka akan menurunkan nilai variabel lain (nilai X naik maka nilai Y turun), dan sebaliknya jika nilai satu variabel diturunkan, maka nilai variabel lain akan naik ( nilai X turun maka nilai Y naik) (Sugiyono, 2007).

Kekuatan hubungan antar variabel dinyatakan dalam bentuk koefisien korelasi. Nilai koefisien korelasi positif terbesar = 1 dan nilai koefisien korelasi negatif terbesar = -1, sedangkan nilai yang terkecil = 0. Jika hubungan antara dua variabel atau lebih memiliki koefisien korelasi = 1 atau -1, maka hubungan tersebut dapat dikatakan sempurna (Grech, 2018). Koefisien korelasi > 0.5 menunjukkan korelasi yang kuat, sedangkan korelasi < 0.5 menunjukkan korelasi lemah sehingga dapat dijelaskan bahwa pada variabel yang satu dapat diprediksi dengan variabel lain tanpa adanya kesalahan (kesalahan prediksi kecil). Sebaliknya, jika nilai koefisien korelasi semakin kecil, maka kesalahan hasil prediksi semakin besar (Sugiyono, 2007).

Uji korelasi dapat dilakukan menggunakan uji korelasi pearson dengan persamaan berikut (Sugiyono, 2007) :

$$r_{xy} = \frac{n \sum x_i y_i - (\sum x_i)(\sum y_i)}{\sqrt{(n \sum x_i^2 - (\sum x_i)^2)(n \sum y_i^2 - (\sum y_i)^2)}} \dots\dots\dots (2.27)$$

dengan  $r_{xy}$  menyatakan koefisien korelasi,  $n$  menyatakan jumlah total dari data,  $\sum x$  menyatakan jumlah total  $x$ ,  $\sum x^2$  menyatakan jumlah dari kuadrat  $x$ ,  $\sum y$  menyatakan jumlah total  $y$ ,  $\sum y^2$  menyatakan jumlah dari kuadrat  $y$ , dan  $\sum xy$

menyatakan jumlah total perkalian  $x$  dan  $y$ . Selanjutnya nilai hubungan variabel tersebut dapat dikatakan signifikan apabila memenuhi ketentuan hasil koefisien korelasi  $r_{xy} >$  distribusi nilai  $r$  table 5%. Kemudian untuk memberikan skala penilaian koefisien korelasi tersebut besar atau kecil, maka dapat digunakan tabel interpretasi koefisien korelasi, seperti yang disajikan pada Tabel 2.2.

Tabel 2.2 Interpretasi koefisien korelasi (Sugiyono, 2007)

<b>Interval Koefisien</b>	<b>Tingkat Hubungan</b>
(+) (-) 0,00 – 0,199	Sangat rendah
(+) (-) 0,20 – 0,399	Rendah
(+) (-) 0,40 – 0,599	Sedang
(+) (-) 0,60 – 0,799	Kuat
(+) (-) 0,80 – 1,000	Sangat Kuat