

BAB II

TINJAUAN PUSTAKA

2.1 Tinjauan Statistik

2.1.1 Analisis Deskriptif

Analisis Deskriptif merupakan suatu teknik untuk mengumpulkan, mengolah, menyederhanakan, menyajikan serta menganalisis data kuantitatif secara deskriptif agar bertujuan memberikan gambaran yang teratur tentang suatu peristiwa ke dalam bentuk tabel, grafik atau diagram yang sesuai (Makridakis *et al*, 1999). Menurut Walpole (1995) Statistika deskriptif merupakan metode-metode yang berkaitan dengan pengumpulan dan penyajian suatu gugus data sehingga memberikan informasi yang berguna. Statistika deskriptif hanya memberikan informasi tentang data yang dimiliki, tetapi tidak bermaksud menguji dan membuat kesimpulan yang lebih luas (generalisasi) (Usman dan Purnomo, 2015).

2.1.2 Peramalan

Peramalan merupakan suatu dugaan atau perkiraan atas terjadinya kejadian diwaktu mendatang. Menurut jangka waktunya, peramalan dibagi menjadi 3 periode sesuai dengan materi yang diramalkannya. Periode peramalan dibagi menjadi 3, yaitu:

- 1) Peramalan Jangka Panjang (*Long-Term Forecasting*) Merupakan peramalan yang memperkirakan keadaan dalam waktu beberapa tahun ke depan.

- 2) Peramalan Jangka Menengah (*Mid-Term Forecasting*) Merupakan peramalan dalam jangka waktu bulanan atau mingguan.
- 3) Peramalan Jangka Pendek (*Short-Term Forecasting*) Merupakan peramalan dalam jangka waktu harian hingga setiap jam. (Nasution, 2005)

Menurut Golyandina (2001), Suatu ramalan yang layak dari deret waktu dapat dilakukan hanya jika kondisi berikut terpenuhi :

1. Deret memiliki struktur
2. Sebuah mekanisme (metode, algoritma) yang mengidentifikasi struktur ditemukan.
3. Sebuah metode kelanjutan deret waktu (*time series continuation*) berdasarkan pada struktur yang diidentifikasi tersedia.
4. Struktur dari deret waktu dipertahankan untuk periode waktu mendatang, yang mana kita akan meramalkan (melanjutkan) deret.

2.1.3 Deret Waktu (*Time Series*)

Menurut Makridakis *et al*, 1999 Model *Time Series* dikenalkan oleh George E. P. Box dan Gwilym M. Jenkins pada tahun 1970 melalui bukunya *Time Series Analysis: forecasting and control*. Analisis *Time Series* merupakan metode peramalan kuantitatif untuk menentukan pola data pada masa lampau yang dikumpulkan berdasarkan urutan waktu, yang disebut data *Time Series*. *Time series* atau runtun waktu adalah urutan pengamatan yang teratur (Wei.,2006) dimana pengumpulan data yang terekam atau teramati dari suatu peristiwa dari waktu ke waktu. Langkah yang perlu diperhatikan dalam melakukan analisis runtun waktu yaitu memahami bentuk pola dari sebuah data yang akan digunakan untuk model

peramalan masa akan datang. Pola data runtun waktu terbagi menjadi empat macam antara lain (Herawati, 2016):

1. Pola Horizontal (H) merupakan sebuah pola yang ketika data terjadi fluktuasi di sekitar nilai *mean* yang konstan.
2. Pola Musiman (S) merupakan sebuah pola dimana ketika data runtun waktu dipengaruhi faktor musiman.
3. Pola Trend (T) merupakan sebuah pola dimana data mengalami penurunan maupun kenaikan sekuler jangka panjang.
4. Pola Siklis (C) merupakan sebuah pola dimana data dipengaruhi fluktuasi perekonomian jangka panjang.

Tujuan analisis deret waktu antara lain:

- 1) Meramalkan kondisi dimasa yang akan datang.
- 2) Mengetahui hubungan antar peubah.
- 3) Kepentingan kontrol (untuk mengetahui apakah proses terkendali atau tidak).

2.1.4 *Exponential Smoothing Holt Winters*

Metode *Exponential Smoothing* merupakan prosedur perbaikan terus-menerus pada peramalan terhadap objek pengamatan terbaru (Makridakis *et al*, 1983). Metode peramalan ini fokus pada penurunan prioritas secara eksponensial pada objek pengamatan yang lebih lama. Metode *Exponential Smoothing* adalah kelas metode yang menghasilkan perkiraan dengan formula sederhana, dengan mempertimbangkan *trend* dan efek musiman dari data. Prosedur ini banyak digunakan sebagai teknik peramalan dalam manajemen persediaan dan peramalan penjualan (Tratar, 2014).

Metode *Holt-Winters* merupakan salah satu variasi dari *Exponential Smoothing*. Metode ini berawal dari *Holt* (1957) dan *Winters* (1960) yang memperluas metode *Holt* untuk menangkap pola musiman dalam data. Metode ini memungkinkan data dimodelkan dengan *mean* lokal, *trend* lokal, dan faktor musiman lokal yang semuanya diperbarui dengan perataan eksponensial. Dalam metode *Holt-Winters* digunakan persamaan peramalan (F_t) dan tiga persamaan pemulusan yang terdiri dari persamaan untuk level (L_t), trend (b_t), dan komponen *seasonal* (S_t) dengan parameter pemulusan berupa α , β , dan γ (Rob J Hyndman, 2014).

Terdapat dua jenis *Holt-Winters* berdasarkan tipe musimannya, yaitu multiplikatif dan aditif. Hal ini dapat digambarkan seperti operasi dasar penambahan dan perkalian dalam matematika. Untuk penerapannya sendiri metode aditif digunakan apabila komponen musiman bersifat konstan sementara metode multiplikatif digunakan apabila ukuran komponen musiman sebanding dengan tingkat trend (Chatfield, 1978).

Persamaan dasar *Holt-Winters* dengan tipe musiman multiplikatif (Makridakis *et al*, 1983) ditunjukkan pada persamaan berikut:

$$L_t = \alpha \frac{y_t}{S_{t-s}} + (1 - \alpha) (L_{t-1} + b_{t-1}) \quad (2.1)$$

$$b_t = \beta (L_t - L_{t-1}) + (1 - \beta) b_{t-1} \quad (2.2)$$

$$S_t = \gamma \frac{y_t}{L_t} + (1 - \gamma) S_{t-1} \quad (2.3)$$

$$F_{t+m} = (L_t + b_{tm}) S_{t-s+m} \quad (2.4)$$

Keterangan:

L_t = level pada periode t,

b_t = trend pada periode t,

S_t = komponen seasonal periode t,

F_{t+m} = peramalan pada periode t+m.

α, β, γ = parameter pemulusan harus bernilai pada interval 0 hingga 1

m = jumlah ramalan ke depan

s = panjang musiman misalnya jumlah bulan atau kuartal dalam setahun

Y_t = mewakili data yang diamati pada titik waktu t.

Untuk menginisiasi *level*, *trend*, dan komponen *seasonal* dapat digunakan rumus pada persamaan berikut:

$$L_s = \frac{1}{s} \times (Y_1 + Y_2 + \dots + Y_s) \quad (2.5)$$

$$b_s = \frac{1}{s} \times \left(\frac{Y_{s+1} - Y_1}{s} + \frac{Y_{s+2} - Y_2}{s} + \dots + \frac{Y_{s+s} - Y_s}{s} \right) \quad (2.6)$$

$$S_p = \frac{Y_p}{L_s}; \text{ dengan } p = 1, 2, 3, \dots, s \quad (2.7)$$

Sementara untuk persamaan dasar *Holt-Winters* dengan tipe musiman aditif (Makridakis *et al*, 1983) ditunjukkan pada persamaan berikut:

$$L_t = \alpha (Y_t - S_{t-s}) + (1 - \alpha) (L_{t-1} + b_{t-1}) \quad (2.8)$$

$$b_t = \beta (L_t - L_{t-1}) + (1 - \beta) b_{t-1} \quad (2.9)$$

$$S_t = \gamma (Y_t - L_t) + (1 - \gamma) S_{t-1} \quad (2.10)$$

$$F_{t+m} = L_t + b_{tm} + S_{t-s+m} \quad (2.11)$$

Inisiasi nilai *level* dan *trend* pada aditif dilakukan dengan menggunakan rumus yang sama dengan metode multiplikatif. Sementara untuk menginisiasi indeks musiman digunakan rumus pada persamaan berikut:

$$S_p = Y_p - L_s; \text{ dengan } p = 1, 2, 3, \dots, s \quad (2.12)$$

Persamaan (2.4) dan (2.11) digunakan untuk menghitung peramalan m periode ke depan. Persamaan ini hanya berlaku untuk $m = 1, 2, \dots, s$. Sementara untuk meramalkan $m = s + 1, s + 2, \dots, 2s$, komponen *seasonal* pada persamaan tersebut seharusnya S_{t-2s+m} (Gardner dan McKenzie, 1989).

2.1.5 *Artificial Neural Network*

Artificial Neural Network atau Jaringan Syaraf Tiruan merupakan kerangka yang fleksibel untuk mengatasi pemodelan komputasi masalah non-linear. Untuk pemodelan sistem, Jaringan Syaraf Tiruan mempunyai kemampuan untuk memetakan hubungan yang non-linear, sehingga sangat menjanjikan untuk mengatasi permasalahan kontrol non-linear (Haykin, 2005). Jaringan syaraf dengan pendekatan non-linear dapat memberikan akurasi yang tinggi ketika memodelkan pola data yang kompleks (Smith dan Gupta, 2000).

2.1.5.1 *Arsitektur Jaringan Syaraf Tiruan*

Suatu jaringan syaraf biasanya terdiri atas lapisan-lapisan (Rofiq, 2017):

1. *Input Layer*

Input layer menerima sinyal/masukan dari jaringan luar. Aktivasi dari *neuron* ini menunjukkan informasi dasar yang kemudian digunakan dalam Jaringan Syaraf Tiruan.

2. *Hidden Layer*

Hidden layer akan menerima dan mengirim sinyal/masukan ke jaringan syaraf. Aktivasi setiap *neuron hidden layer* ditentukan oleh aktivasi dari *neuron-neuron* masukan dan bobot dari koneksi antar *neuron* masukan dan

neuron-neuron pada *hidden layer*. *Hidden layer* terletak diantara *input* dan *output layer*, jumlahnya bisa satu atau beberapa lapisan.

3. *Output Layer*

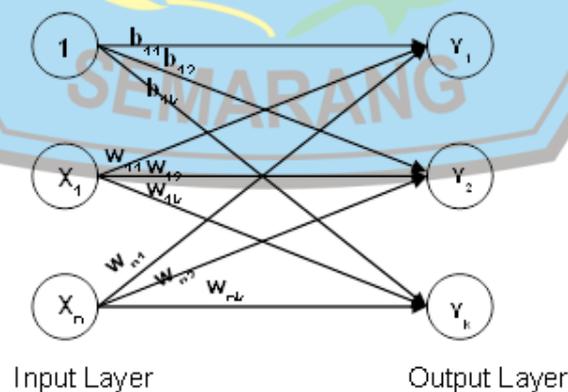
Output layer mengirim data ke jaringan syaraf dan menghasilkan nilai *output*.

Karakteristik dari *neuron-neuron* pada lapisan ini tergantung dari aktivasi *neuron-neuron* pada *hidden layer* dan bobot antara *neuron-neuron* pada *hidden layer* dan *neuron-neuron output*.

Berdasarkan jumlah lapisan yang dimiliki, arsitektur jaringan syaraf tiruan terdiri dari:

1. Jaringan Lapis Tunggal (*Single Layer*)

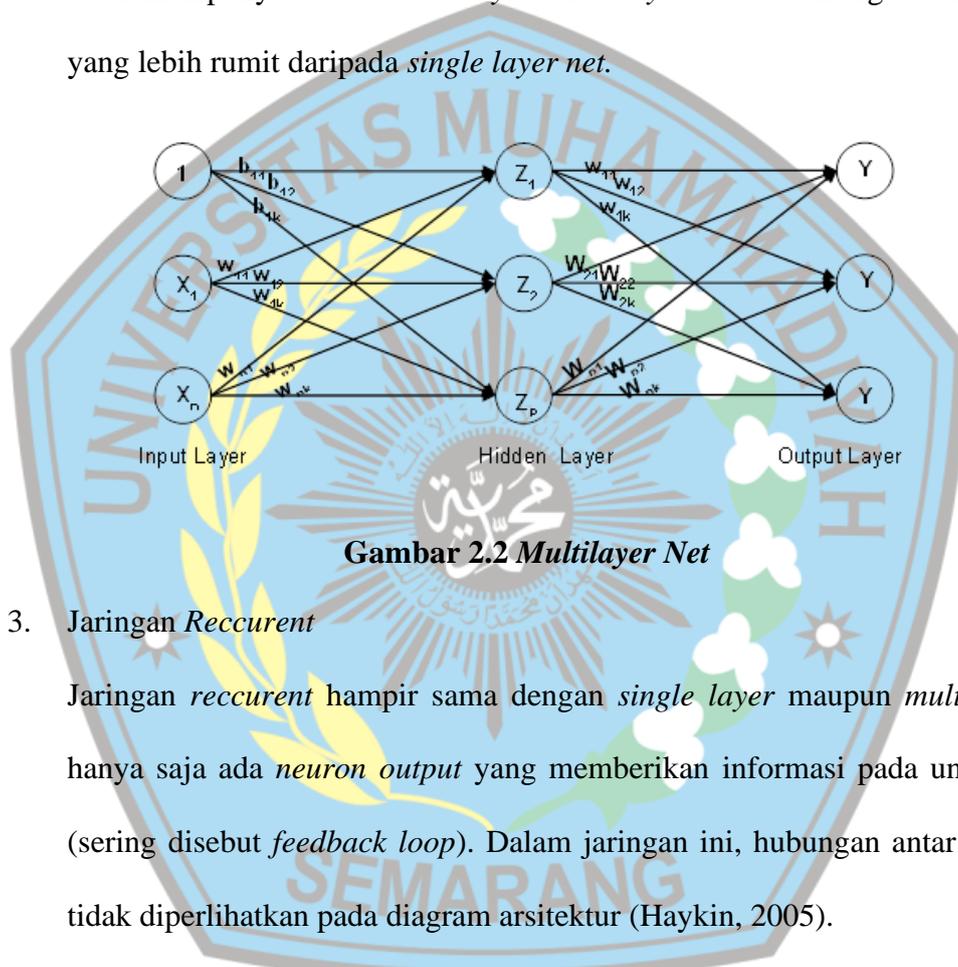
Single layer net merupakan jaringan yang hanya mempunyai satu *layer* untuk menghubungkan nilai bobotnya. *Neuron input* langsung berhubungan dengan *neuron output*. Jaringan ini hanya menerima informasi dan langsung mengolahnya menjadi *output* tanpa melalui *hidden layer*. Ciri yang dimiliki *single layer net* yaitu hanya mempunyai satu *layer input* dan satu *layer output*.



Gambar 2.1 *Single Layer Net*

2. Jaringan Lapis Jamak (*Multilayer Net*)

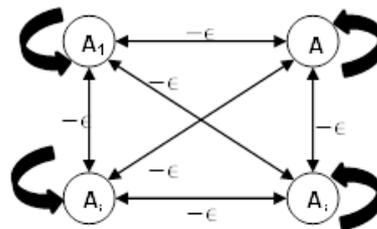
Multilayer net adalah jaringan yang terdiri dari *input layer*, *output layer*, dan *hidden layer*. *Hidden layer* terletak diantara *input* dan *output layer*. *Output* dari *hidden layer* akan menjadi *input* bagi *layer* berikutnya. Jaringan ini paling tidak mempunyai satu *hidden layer*. *Multilayer net* bisa mengatasi masalah yang lebih rumit daripada *single layer net*.



Gambar 2.2 Multilayer Net

3. Jaringan *Reccurent*

Jaringan *reccurent* hampir sama dengan *single layer* maupun *multi layer*, hanya saja ada *neuron output* yang memberikan informasi pada unit *input* (sering disebut *feedback loop*). Dalam jaringan ini, hubungan antar *neuron* tidak diperlihatkan pada diagram arsitektur (Haykin, 2005).



Gambar 2.3 Jaringan *Reccurent*

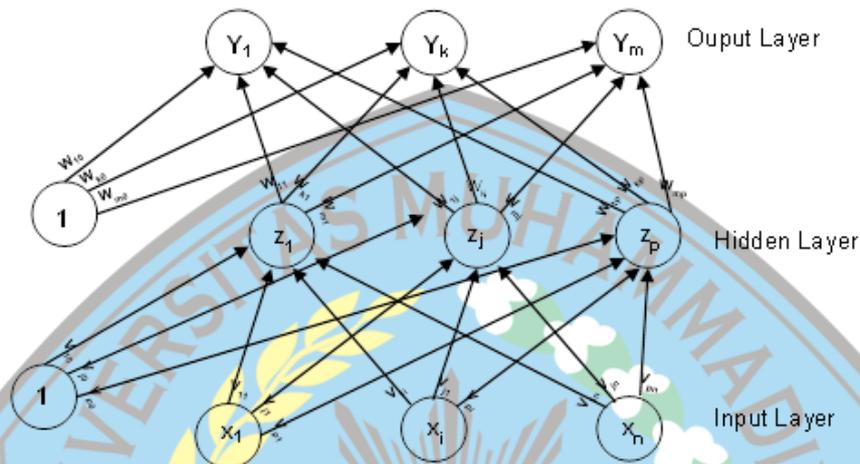
2.1.5.2 *Backpropagation Neural Network*

Backpropagation merupakan metode pelatihan Jaringan Syaraf Tiruan *multilayer* yang menggunakan prosedur pembelajaran terawasi (Saduf & Wani, 2013), dimana dilakukan penyesuaian bobot secara berulang untuk mendapatkan nilai *error* yang rendah. Dalam pemecahan masalah pada sistem, *Backpropagation* memiliki kelebihan yaitu bersifat adaptif (dapat menyesuaikan terhadap dataset) dan *fault tolerance* (kesalahan *error* kecil) (Azhar & Riksakomara, 2017). *Backpropagation* merupakan metode yang banyak digunakan untuk pengenalan pola-pola kompleks (Riedmiller & Braun, 1993). Hal ini terlihat dari kemampuannya dalam mengadaptasikan kondisi jaringan dengan data yang diberikan pada proses pembelajaran (Avan, 2017).

Backpropagation merupakan salah satu algoritma penentuan bobot. Algoritma *Backpropagation* biasa digunakan oleh perceptron dengan banyak lapisan untuk mengubah bobot-bobot yang terhubung dengan *neuron* yang ada pada *hidden layer*. Dasar perbaikan bobot pada algoritma *Backpropagation* dilakukan dengan cara mempertimbangkan kuadrat galat dari model. Perbaikan nilai bobot dalam *Backpropagation* dilakukan dalam arah mundur dengan memanfaatkan turunan dari fungsi aktivasi yang dipakai pada jaringan syaraf tiruan. Pembelajaran dilakukan dengan menggunakan *gradient descent*, dimana galat pada keluaran didorong kembali melalui propagasi balik untuk memperkirakan kesalahan pada *hidden layer* (Cowan, 2013 dalam (Avan, 2017)).

2.1.5.3 Arsitektur Jaringan *Backpropagation*

Arsitektur jaringan algoritma *Backpropagation* pada dasarnya tersusun dari tiga layer, yaitu *input layer*, *hidden layer*, dan *output layer* (Siang, 2005).



Sumber: (Siang, 2005)

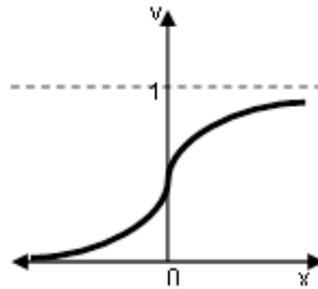
Gambar 2.4 Arsitektur Jaringan Syaraf *Backpropagation*

2.1.5.4 Fungsi Aktivasi

Fungsi aktivasi dalam *Backpropagation* harus mempunyai beberapa karakteristik penting, yaitu: kontinu, terdiferensial dengan mudah, dan merupakan fungsi yang tidak turun. Salah satu fungsi yang memenuhi ketiga syarat tersebut sehingga sering dipakai adalah fungsi *sigmoid biner* yang memiliki *range* [0,1] (Siang, 2005).

$$f(x) = \frac{1}{1+e^{-x}} \text{ dengan turunan } f'(x) = f(x)(1 - f(x)) \quad (2.13)$$

Grafik fungsinya sebagai berikut:



Gambar 2.4 Fungsi Sigmoid Biner

Sumber: (Siang, 2005)

2.1.5.5 Parameter

Parameter-parameter yang digunakan dalam *Backpropagation* adalah (Azhar & Riksakomara, 2017):

1. *Learning Rate*

Learning rate digunakan untuk mendefinisikan pembelajaran model untuk setiap iterasi. Skala penentuan parameter *learnig rate* antara 0 hingga 1.

2. *Epoch*

Epoch adalah banyaknya jumlah iterasi yang digunakan.

3. Momentum

Momentum adalah kemiringan batas kesalahan maksimum yang boleh dihasilkan oleh model. Skala momentum yang digunakan antara 0 sampai 1.

2.1.5.6 Algoritma Pelatihan *Backpropagation*

Backpropagation melatih jaringan untuk mendapatkan keseimbangan antara kemampuan jaringan untuk mengenali pola yang digunakan selama pelatihan serta memberikan respon yang benar terhadap pola masukan yang serupa (tapi tidak

sama) dengan pola yang dipakai selama pelatihan. Algoritma *Backpropagation* terdiri dari tiga fase pelatihan, yaitu sebagai berikut:

a) Fase I: Propagasi Maju (*Feedforward*)

Pada tahap *feed forward*, tiap unit *input* diberi masukan dari luar jaringan syaraf, kemudian *input* tersebut dipropagasikan ke *hidden layer*. Selanjutnya *hidden layer* akan menghitung nilai aktivasinya dan meneruskan sinyal *input* ke *layer output* sehingga menghasilkan *output* dari jaringan tersebut.

b) Fase II: Propagasi Mundur

Nilai aktivasi yang dihasilkan oleh tiap unit *output* akan dibandingkan dengan target sehingga akan menghasilkan *error* (selisih *output* dan target). *Error* tersebut akan dipropagasikan mundur, dimulai dari garis yang berhubungan langsung dengan untui-unit *output* (Fausett, 2004).

c) Fase III: Perubahan bobot

Pada tahap ini, bobot akan dimodifikasi untuk menurunkan nilai *error* (Siang, 2005). Proses ini akan berlangsung berulang kali hingga *error* yang dihasilkan oleh jaringan mendekati nol (Giantara, Hidayanto, & Christiyono, 2013).

Ketiga fase tersebut diulang-ulang terus hingga kondisi penghentian terpenuhi. Pada umumnya kondisi berhenti yang sering dipakai adalah jumlah iterasi. Iterasi akan dihentikan jika sudah melebihi batas maksimum iterasi yang ditetapkan (Siang, 2005).

Adapun langkah-langkah secara lengkap algoritma pembelajaran *backpropagation* adalah sebagai berikut:

- Langkah 0. Inisialisasi Bobot.
- Langkah 1. Jika kondisi berhenti bernilai False lakukan langkah 2-9.
- Langkah 2. Untuk setiap pasang data pelatihan (*Training*), lakukan langkah 3-8.

Feed Forward

Langkah 3. Tiap unit *input* ($x_i, i = 1, 2, 3, \dots, n$) menerima sinyal *input* x dan sinyal tersebut disebarkan ke semua unit pada *hidden layer*.

Langkah 4. Tiap *hidden layer* ($z_j, j = 1, 2, 3, \dots, p$) menjumlahkan bobot sinyal *input*,

$$z_in_j = v_{0j} + \sum_{i=1}^n x_i v_{ij} \quad (0.14)$$

menggunakan fungsi aktivasi untuk menghitung sinyal *output*

$$z_j = f(z_in_j) \quad (0.15)$$

dan mengirimkan sinyal ini ke setiap unit *output*.

Langkah 5. Tiap unit *output* ($y_k, k = 1, 2, 3, \dots, m$) menjumlahkan bobot sinyal masuk,

$$y_in_k = w_{0k} + \sum_{j=1}^j z_j w_{jk} \quad (0.16)$$

menggunakan fungsi aktivasi untuk menghitung sinyal *output*

$$y_k = f(y_in_k) \quad (0.17)$$

Error dari backpropagation

Langkah 6. Tiap unit *output* ($y_k, k = 1, 2, 3, \dots, m$) menerima pola yang sesuai dengan pola pelatihan *input* dan dihitung errornya,

$$\delta_k = (t_k - y_k)f'(y_{in_k}) \quad (0.18)$$

menghitung bobot terkoreksi (digunakan untuk memperbaiki w_{jk}),

$$\Delta w_{jk} = \alpha \delta_k z_j \quad (0.19)$$

menghitung bias terkoreksi (untuk memperbaiki w_{jk})

$$\Delta w_{0k} = \alpha \delta_k \quad (0.20)$$

dan mengirim δ_k ke unit pada *layer* dibawahnya.

Langkah 7. Tiap unit *hidden* ($Z_j, j = 1, 2, 3, \dots, p$) menjumlahkan delta *input* (dari unit *layer* diatas),

$$\delta_{in_j} = \sum_{k=1}^m \delta_k w_{jk} \quad (0.21)$$

dikalikan dengan turunan fungsi aktivasi untuk menghitung error,

$$\delta_j = \delta_{in_j} f'(z_{in_j}) \quad (0.22)$$

menghitung koreksi bobot (digunakan untuk memperbaiki w_{ij})

$$\Delta w_{ij} = \alpha \delta_j x_i \quad (0.23)$$

menghitung bias koreksi (untuk memperbaiki w_{ij})

$$\Delta w_{ij} = \alpha \delta_j \quad (0.24)$$

Perbaharui bobot dan bias

Langkah 8. Tiap unit *output* ($y_k, k = 1, 2, 3, \dots, m$) memperbaharui bobot dan bias ($j = 0, \dots, p$):

$$w_{jk}(\text{new}) = w_{jk}(\text{old}) + \Delta w_{jk} \quad (0.25)$$

tiap unit *hidden* ($z, j = 1, 2, 3, \dots, p$) memperbaharui bobot dan bias ($i = 0, \dots, n$):

$$w_{ij}(\text{new}) = w_{ij}(\text{old}) + \Delta w_{ij} \quad (0.26)$$

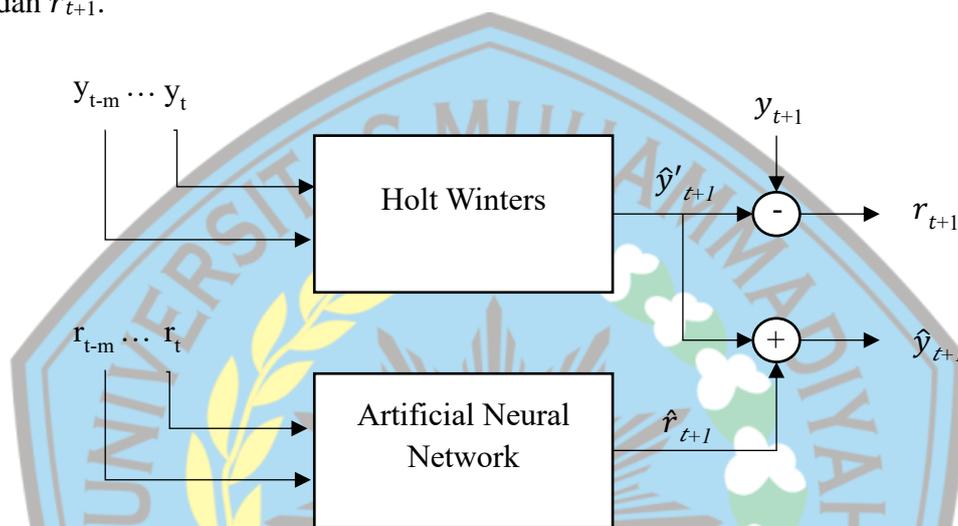
Langkah 9. Tes kondisi berhenti.

2.1.6 Hybrid Holt Winters-Artificial Neural Network

Hybrid Holt-Winters Artificial Neural Network merupakan penggabungan model peramalan *Holt-Winters* dan *Artificial Neural Network* (H. Rodriguez *et al*, 2016). Dimana *Holt-Winters* digunakan untuk peramalan data aktual sementara *Artificial Neural Network* digunakan untuk peramalan *error* atau residu yang diperoleh dari *Holt-Winters*. Penggunaan *Artificial Neural Network* untuk memodelkan residu yang dihasilkan model *Holt-Winters* karena kemampuannya untuk memodelkan rangkaian waktu nonlinear tanpa trend (Zhang *et al*, 1998).

Gambar 2.6 merupakan bagan yang menjelaskan mekanisme *Holt-Winters Artificial Neural Network*. *Holt-Winters* digunakan pada tahap awal untuk meramalkan data deret waktu aktual y_t yang ada sehingga dapat diperoleh hasil peramalan data aktual \hat{y}'_{t+1} . Dari hasil peramalan yang diperoleh dengan menggunakan *Holt-Winters* dapat dihitung *error* atau residu r_t dengan mencari selisih antara data aktual dan hasil peramalan. Residu r_t yang diperoleh selanjutnya

digunakan untuk membuat model pada *Artificial Neural Network* sehingga dapat diperoleh hasil peramalan residu \hat{r}_{t+1} . Setelah diperoleh hasil peramalan dari *Holt-Winters* dan hasil peramalan dari *Artificial Neural Network*, maka mereka dapat digunakan untuk membuat peramalan ke depan yang dengan menjumlahkan \hat{y}'_{t+1} dan \hat{r}_{t+1} .



Gambar 2.6 Mekanisme Hybrid Holt Winters-Artificial Neural Network

Untuk menyempurnakan model *Holt-Winters* dan melatih residu *Artificial Neural Network*, informasi yang dibutuhkan adalah data deret waktu yang dijelaskan oleh $Y = \{y_1, y_2, y_3, \dots, y_n\}$ dimana $y_1, y_2, y_3, \dots, y_n$ merupakan elemen dari deret waktu. Sementara residu disimpan dalam vektor yang didefinisikan sebagai $r = \{r_1, r_2, r_3, \dots, r_n\}$. Residu ini diperoleh dengan menggunakan model *Holt-Winters* seperti yang ditunjukkan pada persamaan berikut.

$$r_{t+1} = y_{t+1} - \hat{y}'_{t+1} \quad (2.27)$$

Dimana prediksi \hat{y}'_{t+1} dihasilkan dari persamaan berikut.

$$\hat{y}'_{t+1} = f_{hw}(y_{t-m}, \dots, y_t) \quad (2.28)$$

Residu r dimodelkan dengan menggunakan *Artificial Neural Network* seperti yang ditunjukkan pada persamaan berikut.

$$\hat{r}_{t+1} = f_{ann}(r_{t-m}, \dots, r_t) \quad (2.29)$$

Ketika peramalan *Holt-Winters* dan *Artificial Neural Network* telah dimiliki, maka mereka dapat digunakan untuk membuat peramalan ke depan dengan menggunakan persamaan berikut (Rodrigues *et al*, 2016 dalam (Garini, 2018)).

$$\hat{y}_{t+1} = \hat{y}'_{t+1} + \hat{r}_{t+1} \quad (2.30)$$

2.1.7 Kebaikan Model

Dalam melakukan suatu peramalan yang merupakan kegiatan memprediksi masa depan dengan menggunakan data di masa lampau, hasil yang akan didapatkan tidaklah sama dengan data sesungguhnya (Ishcak, 2018). Maka dari itu usaha untuk membuat nilai error seminimal mungkin dibutuhkan pada proses peramalan. Berikut penjelasan untuk masing-masing ketepatan peramalan yang digunakan dalam penelitian ini:

2.1.7.1 Mean Absolute Deviation (MAD)

Mean Absolute Deviation merupakan penjumlahan absolut kesalahan peramalan tanpa menghiraukan tanda aljabarnya dibagi dengan banyaknya data yang diamati. Dalam MAD, kesalahan dengan tanda positif atau negatif akan diberlakukan sama, yang diukur hanya besar kesalahan secara absolut (Herjanto, 2009). Tujuan pengabaian ini adalah menghindari ukuran yang bernilai sama dengan nol yang secara statistik kurang bermanfaat. Karena nilai positif dan negatif

dapat saling menghapuskan satu sama lain (Mason, 1996 dalam (Garini, 2018)). MAD berguna untuk mengukur kesalahan ramalan dalam unit yang sama sebagai deret asli. Rumus untuk menghitung MAD dinyatakan seperti pada persamaan berikut (Herjanto, 2009).

$$MAD = \frac{1}{n} \sum_{t=1}^n |Y_t - \hat{Y}_t| \quad (2.31)$$

Dimana Y_t adalah nilai aktual hasil observasi, \hat{Y}_t adalah nilai hasil peramalan, dan t menunjukkan periode ke- t .

2.1.7.2 Mean Squared Error (MSE)

MSE merupakan penjumlahan kesalahan yang telah dikuadratkan dibagi dengan banyaknya data yang diamati. MSE memperkuat pengaruh dari angka-angka kesalahan peramalan yang besar, tetapi memperkecil pengaruh dari angka-angka kesalahan peramalan yang kecil (Herjanto, 2009). Hal ini terjadi karena kesalahan yang ada dikuadratkan. Rumus untuk menghitung MSE dinyatakan seperti pada persamaan berikut (Herjanto, 2009).

$$MSE = \frac{1}{n} \sum_{t=1}^n (Y_t - \hat{Y}_t)^2 \quad (2.32)$$

Dimana Y_t adalah nilai aktual hasil observasi, \hat{Y}_t adalah nilai hasil peramalan, dan t menunjukkan periode ke- t .

2.1.7.3 Mean Absolute Percentage Error

Mean Absolute Percentage Error (MAPE) yaitu rata-rata persentase kesalahan pertama dari beberapa periode. Tingkat keakuratan dapat dijelaskan dengan membandingkan nilai yang diproyeksikan dengan nilai aktual. Untuk

melakukan peramalan dan untuk mengetahui akuratnya sebuah model maka nilai akurasi harus semakin kecil.

Rumus MAPE secara umum sebagai berikut:

$$\text{MAPE} = \frac{1}{n} \sum_{t=1}^n \left| \frac{Y_t - \hat{Y}_t}{Y_t} \right| \times 100\% \quad (2.33)$$

Keterangan:

n = Jumlah Sampel

Y_t = Nilai Akurasi Indeks pada periode ke- t

\hat{Y}_t = Nilai Prediksi Indeks pada periode ke- t

Untuk ketentuan MAPE adalah sebagai berikut :

Tabel 2.1 Ketentuan MAPE

MAPE	Keterangan
<10%	Sangat Baik
10% - 20%	Baik
21% - 50%	Cukup
>50%	Tidak Akurat

Sumber : (Lewis, 1982).

2.2 Tinjauan Non Statistik

2.2.1 Pariwisata

Pariwisata berasal dari bahasa Sanskerta yang terdiri atas dua suku kata yaitu “pari” yang berarti banyak, berkali-kali, berputar-putar, lengkap dan kata “wisata” yang berarti perjalanan atau bepergian. Sehingga kata pariwisata dapat diartikan suatu perjalanan atau bepergian yang dilakukan secara berkali-kali atau berputar-putar dari suatu tempat ke tempat lain.

Pariwisata menyangkut perjalanan dari suatu tempat ke tempat lain atau disebut dengan istilah “*tour*”. Seperti kutipan dari batasan yang diberikan oleh WATA (*World Association of Travel Agent*) adalah merupakan perjalanan keliling yang memakan waktu lebih dari tiga hari yang diselenggarakan oleh Biro Perjalanan Wisata (BPW) dengan cara antara lain peninjauan di beberapa kota atau objek wisata dalam maupun di luar negeri.

Pariwisata internasional adalah salah satu indikator yang paling penting bagi negara-negara di dunia, dengan sektor pariwisata dapat menjadi pasar ekspor untuk produk-produk domestik (Song, Li, & Cao, 2018). Menurut Law *et al* (2019) pariwisata merupakan sumber penting dari pertumbuhan ekonomi serta pendapatan devisa serta terciptanya pekerjaan. Faktor yang menentukan perkembangan sektor pariwisata yaitu jumlah kunjungan wisatawan baik dalam negeri maupun mancanegara (Herawati, 2016).

2.2.2 Wisatawan Mancanegara

Wisatawan mancanegara adalah orang yang datang ke suatu negara yang bertujuan untuk berlibur dan sukarela berkunjung untuk mendapatkan sesuatu yang lain. Menurut *United Nation World Tourism Organization* atau UNWTO menyebutkan bahwa wisatawan mancanegara adalah seseorang yang berkunjung ke suatu negara di luar tempat tinggalnya tidak melebihi 12 bulan dengan tujuan untuk berlibur atau keperluan lainnya tanpa bermaksud untuk mencari penghasilan di tempat yang dikunjungi.

Menurut Badan Pusat Statistik, Wisatawan Mancanegara adalah setiap pengunjung yang mengunjungi suatu negara di luar tempat tinggalnya, didorong

oleh satu atau beberapa keperluan tanpa bermaksud memperoleh penghasilan di tempat yang dikunjungi dan lamanya kunjungan tersebut tidak lebih dari 12 (dua belas) bulan. Definisi ini mencakup dua kategori wisatawan mancanegara, yaitu: wisatawan dan pelancong.

- 1) Wisatawan (*Tourist*) adalah setiap pengunjung seperti definisi wisatawan mancanegara, yang tinggal paling sedikit 24 jam, akan tetapi tidak lebih dari 12 bulan di tempat yang dikunjungi, dengan antara lain berlibur/rekreasi, olahraga, bisnis, menghadiri pertemuan, studi, dan kunjungan dengan alasan kesehatan;
- 2) Pelancong (*Excursionist*) adalah setiap pengunjung seperti definisi wisatawan mancanegara, yang tinggal kurang dari 24 jam di tempat yang dikunjungi, termasuk *cruise passengers*. *Cruise Passengers* yaitu setiap pengunjung yang tiba di suatu negara dengan kapal atau kereta api, dimana mereka tidak menginap di akomodasi yang tersedia di negara tersebut) (Badan Pusat Statistik, 2021).