

BAB II

KAJIAN PUSTAKA

A. Peramalan

Peramalan adalah suatu teknik untuk meramalkan keadaan di masa yang akan datang melalui pengujian keradaan di masa lalu. Pada dasarnya, meramalkan sama halnya dengan memprediksi atau memperkirakan suatu hal, kejadian atau peristiwa masa datang yang berdasarkan pada masa lalu hingga saat ini.

Berdasarkan periode waktunya, peramalan diklasifikasikan menjadi 3 bentuk (Montgomery, *et al*, 2008:1-2):

1. Jangka Pendek (*Short Term*)

Peramalan jangka pendek adalah peramalan yang hanya mencakup kurang dari tiga bulan. Sebagai contoh peramalan jangka pendek adalah peramalan jumlah produksi atau penjualan suatu barang. Pada peramalan jangka pendek, data sebelumnya masih berhubungan untuk meramalkan di masa yang akan datang.

2. Jangka Menengah (*Medium Term*)

Perkiraan jangka menengah umumnya mencakup hitungan bulan hingga tiga tahun. Kegiatan peramalan dalam jangka menengah masih menggunakan metode kuantitatif dan kualitatif karena data historis masa lalu dianggap masih cukup relevan untuk meramalkan masa datang. Contoh peramalan dalam jangka menengah adalah meramalkan anggaran penjualan atau produksi.

3. Jangka Panjang (*Long Term*)

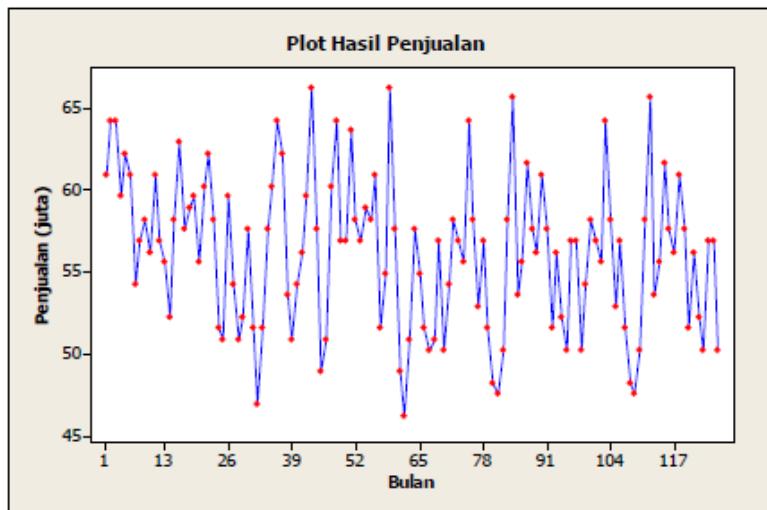
Peramalan jangka panjang adalah peramalan yang meliputi kurun waktu lebih dari tiga tahun. Peramalan jangka panjang digunakan untuk merencanakan produk baru, lokasi, serta penelitian dan pengembangan.

B. Analisis Deret Berkala (*Time Series*)

Deret berkala (*time series*) adalah serangkaian pengamatan terhadap suatu peristiwa, kejadian, gejala atau peubah yang diambil dari waktu ke waktu, dicatat secara teliti menurut urutan waktu terjadinya dan kemudian disusun sebagai data statistik (Hanke & Wichern, 2005: 58). *Time series* ini digunakan untuk memperoleh gambaran dari kegiatan suatu keadaan atau sifat variabel di waktu yang lalu untuk peramalan dari nilai variabel itu pada periode yang akan datang. Berikut merupakan komponen-komponen *time series* yang harus diperhatikan (Hanke & Wichern, 2005:58-59).

1. Gerakan horizontal

Gerakan horizontal adalah suatu pergerakan data yang berfluktuasi di sekitar nilai konstan atau rata-rata yang membentuk garis horizontal. Data ini juga disebut data stasioner. Contoh gerakan horizontal dapat dilihat pada Gambar 2.1. yang merupakan plot data suatu penjualan.

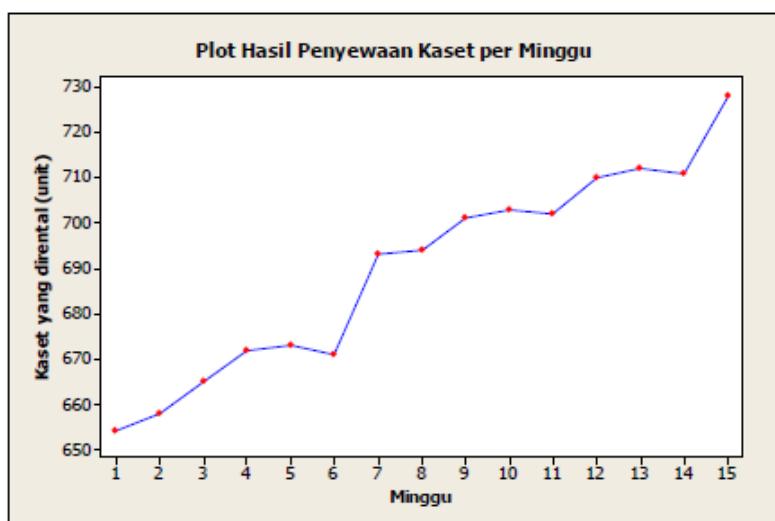


(Sumber: Hanke & Wichern, 2005: 434)

Gambar 2.1. Plot Contoh Pola Horizontal pada Data Penjualan

2. Gerakan Trend

Gerakan *Trend* adalah jika suatu data bergerak pada jangka waktu tertentu dan cenderung menuju ke satu arah baik naik atau turun. Contoh dari pola gerakan trend dapat dilihat pada gambar 2.2. yang merupakan plot dari data penyewaan kaset film sebuah perusahaan di Denver, Colorado.

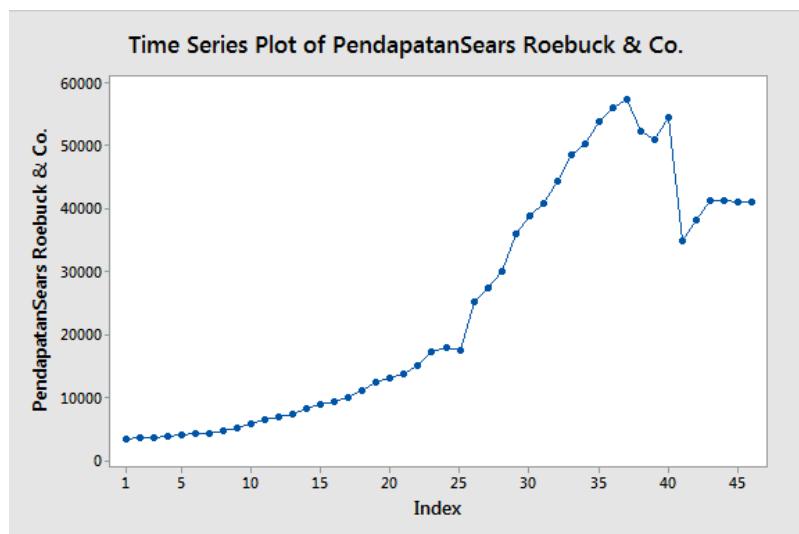


(Sumber: Hanke & Wichern, 2005: 111)

Gambar 2.2. Plot Contoh Pola *Trend* pada Data Penyewaan Kaset

3. Gerakan Siklik

Gerakan Siklik adalah gerakan naik atau turun secara siklik di sekitar tren atau kondisi normal. Data yang sering mengalami siklik antara lain data perdagangan, industri, dan keuangan. Pada gambar 2.3. menunjukkan pola data dengan pergerakan siklik pada data pajak perusahaan Sears Roebuck &Co.

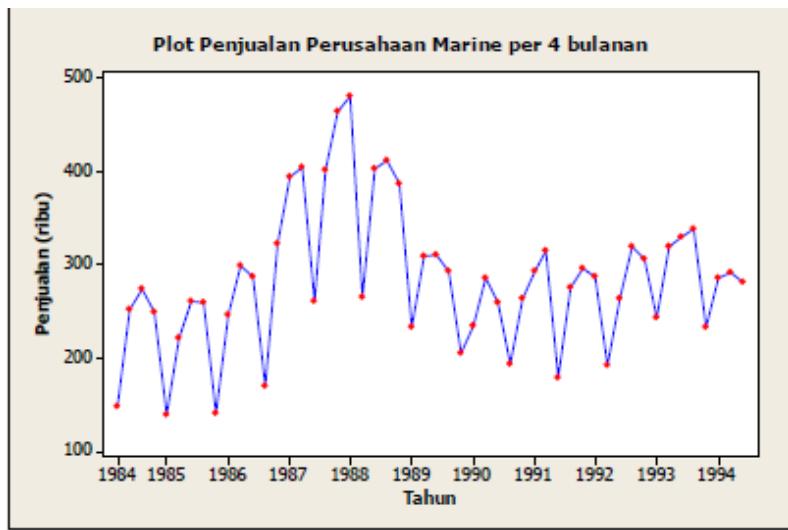


(Sumber: Hanke & Wichern, 2005: 70)

Gambar 2.3. Plot Contoh Pola Siklik pada Data Pajak Perusahaan Sear

4. Gerakan Musiman

Gerakan Musiman merupakan gerakan yang berulang-ulang secara teratur selama kurang lebih satu tahun. Sebagai contoh pola yang berulang setiap minggu, bulan atau kuartalan. Pada kuartalan perulangan terjadi setiap empat bulan. Contoh pola data musiman adalah data penjualan Marine yang ditunjukkan pada gambar 2.4.



(Sumber: Hanke & Wichern, 2005: 73)

Gambar 2.4. Plot Contoh Pola Musiman pada Data Penjualan

5. Stasioner

Stasioner adalah keadaan dimana tidak ada perubahan rata-rata (*mean*) dan varians dari waktu ke waktu atau keduanya selalu kontinu (tidak terjadi pertumbuhan atau penurunan) setiap waktu (Popavic, 2005:38).

Para peneliti sering mengamati pola pada plot data untuk memutuskan data yang diperoleh stasioner atau tidak stasioner. Jika plot data *time series* cenderung konstan atau tidak terdapat pertumbuhan atau penurunan maka data sudah stasioner.

6. Autokorelasi

a. Autocorrelation Function (ACF)

Autokorelasi merupakan hubungan (korelasi) variabel dari sebuah data *time series* untuk selang waktu (*lag*) yang berlainan. Autokorelasi dapat digunakan untuk menentukan ada tidaknya faktor musiman (*seasonality*)

beserta panjang musim dalam deret tersebut (Makridakis et al, 1999: 512).

Selain itu, autokorelasi dapat digunakan untuk menentukan kestasioneran suatu data.

Dalam suatu proses stasioner Y_t , rata-rata $E(Y_t) = \mu$ dan $\text{var}(Y_t) = E(Y_t - \mu)^2 = \sigma^2$ adalah konstan, dan kovarians $\text{cov}(Y_t, Y_{t+k})$ antara Y_t dan nilainya pada periode waktu lain Y_{t+k} disebut autokovarian pada *lag* k , didefinisikan sebagai (Wei, 2006: 10):

$$\gamma_k = \text{cov}(Z_t, Z_{t+k}) = E(Z_t - \mu)(Z_{t+k} - \mu) \quad (2.1)$$

Nilai-nilai γ_k pada saat $k = 1, 2, \dots$ disebut fungsi autokovarian. Koefisien autokorelasi pada *lag* k (ρ_k) antara pengamatan Y_t dan Y_{t+k} pada populasi dinyatakan sebagai berikut (Montmogomery, Jennings & Kulahci, 2008:30):

$$\rho_k = \frac{[(y_t - \mu)(y_{t+k} - \mu)]}{\sqrt{E[(y_t - \mu)^2]E[(y_{t+k} - \mu)^2]}} = \frac{\text{cov}(y_t, y_{t+k})}{\text{var}(y_t)} = \frac{\gamma_k}{\gamma_0} \quad (2.2)$$

dengan

γ_k : autokovarians pada *lag* k

ρ_k : autokorelasi pada *lag* k

t : waktu pengamatan, $t = 1, 2, 3, \dots$

Y_t : pengamatan pada saat t

Y_{t+k} : pengamatan pada saat $t + k$

μ : rata-rata seluruh pengamatan

$$\text{var}(Y_{t+k}) = \text{var}(Y_t) = \gamma_0$$

Nilai-nilai ρ_k (atau r_k) pada saat $k = 1, 2, 3, \dots$ disebut fungsi autokorelasi (ACF). Fungsi autokorelasi dapat diperkirakan dari fungsi autokorelasi sampel yang didefinisikan dengan (Montgomery *et al* 2008:30)

$$r_k = \frac{c_k}{c_0} \quad (2.3)$$

dengan c_k adalah perkiraan fungsi autokovarian sampel yang didefinisikan sebagai

$$c_k = \frac{1}{n} \sum_{t=1}^{n-k} (y_t - \bar{y})(y_{t+k} - \bar{y}) \quad (2.4)$$

dengan

r_k : autokorelasi pada lag k

y_t : pengamatan pada saat t

y_{t+k} : pengamatan pada saat $t + k$

\bar{y} : nilai rata-rata dari pengamatan

Nilai autokorelasi berada pada interval -1 hingga 1 . Ketika nilai autokorelasi tepat atau mendekati ± 1 , maka dapat disimpulkan bahwa terdapat hubungan yang erat antara data *time series*. Jika nilai autokorelasi bernilai 0 maka tidak terdapat hubungan antara data *time series*. Untuk mengetahui suatu autokorelasi signifikan atau tidaknya dapat menggunakan suatu pengujian dengan hipotesis sebagai berikut:

$H_0: \rho_k = 0$ (koefisien autokorelasi *lag k* tidak berbeda signifikan dari nol)

$H_1: \rho_k \neq 0$ (koefisien autokorelasi *lag k* berbeda signifikan dari nol)

Statistik uji yang digunakan adalah

$$t_{hitung} = \frac{r_k}{SE(r_k)} \quad (2.5)$$

dengan SE adalah standart *error* yang didefinisikan (Hanke & Wichern, 2005: 64)

$$SE(r_k) = \sqrt{\frac{1 + 2 \sum_{i=1}^{k-1} r_i^2}{n}} \quad (2.6)$$

dengan

$SE(r_k)$: standar *error* koefisien autokorelasi pada *lag* k

r_k : autokorelasi pada *lag* k

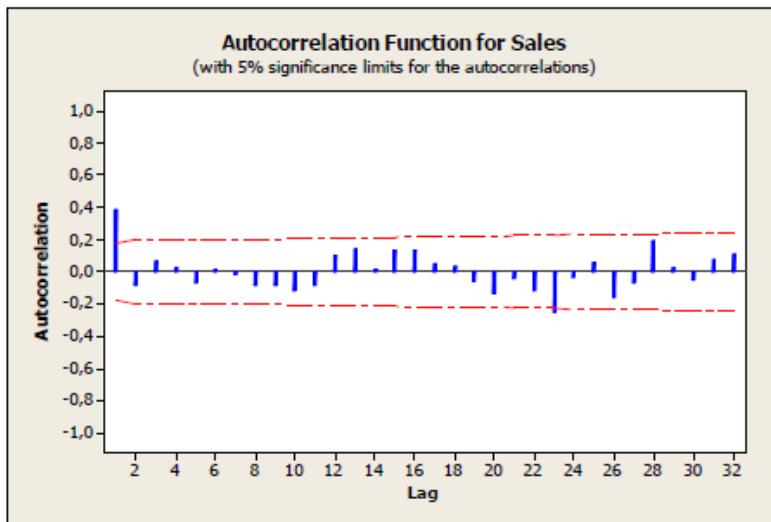
n : banyak pengamatan.

Kriteria keputusan dari pengujian ini adalah autokorelasi signifikan jika $t_{hitung} > t_{\alpha/2}$ dengan derajat bebas $n-1$.

Signifikansi autokorelasi juga dapat dilihat dengan selang kepercayaan r_k dengan pusat 0. Selang kepercayaan r_k dapat dihitung menggunakan rumus pada persamaan 2.7.

$$0 \pm t_{n-1}(\alpha/2) \times SE(r_k) \quad (2.7)$$

Selang kepercayaan r_k dapat direpresentasikan dalam sebuah plot autokorelasi dengan bantuan program Minitab 17. Contoh plot autokorelasi dapat dilihat dari Gambar 2.5. Selang kepercayaan direpresentasikan dengan garis putus-putus merah. Kriteria autokorelasi pada suatu *lag* dikatakan signifikan jika nilai autokorelasi melewati garis putus-putus merah. Pada Gambar 2.5 plot contoh autokorelasi *lag* yang berbeda signifikan dari nol adalah *lag* 1 dan *lag* 24.



Gambar 2.5. Plot Contoh Fungsi Autokorelasi pada Data Penjualan

b. Partial Autocorrelation Function (PACF)

Partial Autocorrelation Function (PACF) atau autokorelasi parsial digunakan untuk mengukur korelasi antara Y_t dengan Y_{t+k} setelah Y_{t+1}, Y_{t+2}, \dots dan Y_{t+k-1} dihilangkan. Autokorelasi parsial dinotasikan dengan ϕ_{kk} dengan rumus (Montgomery, *et al*, 2008:250):

$$\phi_{kk} = \frac{|\rho_k|}{|P_k|} \quad (2.8)$$

Rumus (2.8) didasarkan pada persamaan Yule-Walker

$$P_k \phi_{kk} = \rho_k \quad (2.9)$$

dengan

$$\rho_k = \begin{bmatrix} \rho_1 \\ \rho_2 \\ \vdots \\ \rho_k \end{bmatrix} \text{ dan } \phi_{kk} = \begin{bmatrix} \phi_{k1} \\ \phi_{k2} \\ \vdots \\ \phi_{kk} \end{bmatrix}$$

dan

$$P_k = \begin{bmatrix} 1 & \rho_1 & \rho_2 & \cdots & \rho_{k-1} \\ \rho_1 & 1 & \rho_1 & \cdots & \rho_{k-2} \\ \rho_2 & \rho_2 & 1 & \cdots & \rho_{k-3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \rho_{k-1} & \rho_{k-2} & \rho_{k-3} & \cdots & 1 \end{bmatrix}$$

Untuk $k = 1, 2, 3, \dots$ diperoleh

$$\phi_{11} = \frac{|\rho_1|}{|P_1|} = \frac{|\rho_1|}{1} = |\rho_1|$$

$$\phi_{22} = \frac{|\rho_2|}{|P_2|} = \frac{\begin{vmatrix} 1 & \rho_1 \\ \rho_1 & \rho_2 \end{vmatrix}}{\begin{vmatrix} 1 & \rho_1 \\ \rho_1 & 1 \end{vmatrix}} = \frac{\rho_2 - \rho_1^2}{1 - \rho_1^2}$$

⋮

$$\phi_{kk} = \frac{\begin{vmatrix} 1 & \rho_1 & \rho_2 & \cdots & \rho_{k-2} & \rho_1 \\ \rho_1 & 1 & \rho_1 & \cdots & \rho_{k-3} & \rho_2 \\ \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ \rho_{k-1} & \rho_{k-2} & \rho_{k-3} & \cdots & \rho_1 & \rho_2 \end{vmatrix}}{\begin{vmatrix} 1 & \rho_1 & \rho_2 & \cdots & \rho_{k-2} & \rho_{k-1} \\ \rho_1 & 1 & \rho_1 & \cdots & \rho_{k-3} & \rho_{k-2} \\ \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ \rho_{k-1} & \rho_{k-2} & \rho_{k-3} & \cdots & \rho_1 & 1 \end{vmatrix}}. \quad (2.10)$$

ϕ_{kk} adalah fungsi dari k yang disebut fungsi autokorelasi parsial. Pengujian signifikansi autokorelasi parsial menggunakan hipotesis :

$H_0: \phi_{kk} = 0$ (autokorelasi parsial pada lag k tidak berbeda signifikan dari nol)

$H_0: \phi_{kk} \neq 0$ (autokorelasi pasrial pada lag k berbeda signifikan dari nol)

Uji signifikan ini menggunakan statistik uji :

$$t = \frac{\widehat{\phi}_{kk}}{SE(\phi_{kk})} \text{ dengan } df = n - 1 \quad (2.11)$$

Standar error autokorelasi parsial menggunakan rumus (Wei, 2006:22) :

$$SE(\widehat{\phi}_{kk}) = \sqrt{\frac{1}{n}} \quad (2.12)$$

dengan

$SE(\hat{\phi}_{kk})$: standar error autokorelasi parsial pada *lag k*

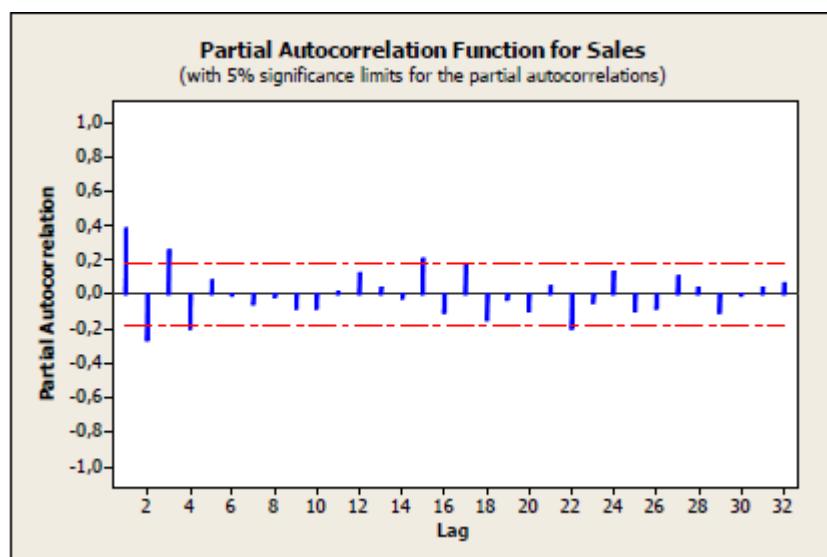
$\hat{\phi}_{kk}$: autokorelasi parsial pada *lag k*

n : banyak pengamatan

Autokorelasi parsial dikatakan berbeda signifikan dari nol jika H_0 ditolak

dengan kriteria $t_{hitung} > t_{n-1} \left(\frac{\alpha}{2} \right)$ atau $t_{hitung} < -t_{n-1} \left(\frac{\alpha}{2} \right)$.

Signifikansi autokorelasi parsial dapat juga diketahui dengan melihat *correlogram* untuk autokorelasi parsial. Berikut pada gambar 2.6 dapat dilihat bahwa data penjualan berbeda signifikan dari nol pada *lag 1, lag 2, lag 3, lag 4, lag 15, dan lag 22*.



Gambar 2.6. Plot Contoh Fungsi Autokorelasi Parsial untuk data Penjualan. signifikansi autokorelasi dapat ditentukan dengan melihat *correlogram* dengan fungsi autokorelasi parsial. *Correlogram* adalah plot antara *lag k* dengan $\hat{\phi}_{kk}$. Selang kepercayaan yang berpusat di $\hat{\phi}_{kk} = 0$ dapat ditentukan dengan menggunakan rumus :

$$0 \pm t_{n-1} \left(\frac{\alpha}{2} \right) \times SE(\hat{\phi}_{kk}) \quad (2.13)$$

7. White Noise

Suatu proses (Y_t) disebut *white noise* jika suatu variabel acak tidak berkorelasi dan berdistribusi tertentu dengan rata-rata tetap $E(Y_t) = 0$, variansi konstan $var(a_t) = \sigma_a^2$, dan $cov(a_t, a_{t+k}) = 0$ untuk $k \neq 0$. Proses *white noise* adalah stasioner dengan fungsi autokovarian sebagai berikut:

$$\gamma_k = \begin{cases} \sigma^2, & k = 0 \\ 0, & k \neq 0 \end{cases} \quad (2.14)$$

fungsi autokorelasi,

$$\rho_1 = \begin{cases} 1, & k = 0 \\ 0, & k \neq 0 \end{cases} \quad (2.15)$$

dan fungsi autokorelasi parsial,

$$\varphi_{kk} = \begin{cases} 1, & k = 0 \\ 0, & k \neq 0 \end{cases} \quad (2.16)$$

Suatu proses *white noise* dapat diperoleh dengan melihat plot ACF dan PACF dengan nilai autokorelasi tidak melebihi garis signifikansi. Pada proses *white noise*, koefisien autokorelasi dan autokorelasi parsial tidak berbeda signifikan dari nol.

8. Kriteria Model Terbaik

Hasil peramalan atau hasil prediksi tidak dapat dipisahkan dengan ketidakpastian karena bukan hasil yang sebenarnya sehingga pasti ada kesalahan peramalan. Kesalahan peramalan dapat diukur dengan beberapa kriteria (Hanke & Wichern, 2005:79-80). Metode yang sering digunakan guna

perhitungan kesalahan dalam peramalan adalah *Mean Absolute Percent Error* (MAPE) dan *Mean Squared Error* (MSE).

a. Means Absolute Percent Error (MAPE)

Penentuan nilai MAPE menggunakan rumus :

$$MAPE = \frac{1}{n} \sum_{t=1}^n \frac{|Y_t - \hat{Y}_t|}{Y_t} \times 100\% \quad (2.17)$$

dengan

Y_t : nilai pengamatan ke- t

\hat{Y}_t : nilai peramalan pada waktu ke- t

n : banyak pengamatan

b. Mean Squared Error (MSE)

Nilai MSE digunakan untuk mengukur ketepatan nilai dugaan model yang dinyatakan dalam rata-rata kuadrat dari kesalahan. Berikut ini rumus untuk menghitung nilai MSE :

$$MSE = \frac{1}{n} \sum_{t=1}^n (Y_t - \hat{Y}_t)^2 \quad (2.18)$$

C. Model ARIMA (*Autoregressive Integrated Moving Average*)

ARIMA (*Autoregressive Integrated Moving Average*) terdiri penggabungan dua metode menjadi satu, yaitu AR (*Autoregressive*) dan MA (*Moving Average*). Model ARIMA tidak cukup baik untuk peramalan jangka panjang dikarenakan peramalan yang dihasilkan cenderung mendatar/konstan. Model ARIMA umumnya dituliskan dengan notasi ARIMA(p,d,q). P adalah derajat proses AR, d adalah orde pembedaan dan q adalah derajat proses MA.

Langkah-langkah pembentukan model ARIMA sebagai berikut (Hanke & Wichern,2005:389):

1. Identifikasi model

Langkah pertama dalam identifikasi model adalah melihat apakah data sudah stasioner. Stasioneritas data dilihat berdasarkan plot data. Data tidak stasioner pada *time series* dilihat dari plot data yang mengalami kenaikan maupun penurunan jika data tidak stasioner maka dilakukan proses *differencing*. Tujuan dari *differencing* ini adalah untuk mengubah data yang mulanya tidak stasioner menjadi stasioner. Banyaknya *differencing* (pembeda) dinotasikan dengan d . Jika data telah stasioner setelah proses *differencing* pertama, maka nilai $d=1$ dan seterusnya. Tetapi jika data telah stasioner tanpa dilakukan *differencing* maka nilai $d=0$.

Setelah data stasioner, maka dilakukan proses pemilihan model yang tepat. Proses pemilihan model yang tepat dilakukan dengan mengidentifikasi orde AR dan MA pada plot grafik ACF dan PACF. Tabel 2.1 menunjukkan pola plot ACF dan PACF.

Tabel 2.1 Pola Plot ACF dan PACF

No.	Model	ACF	PACF
1.	AR (p)	Menurun secara eksponensial	Terputus setelah $lag p$
2.	MA (q)	Terputus setelah $lag k$	Menurun secara eksponensial
3.	ARMA (p,q)	Menurun secara eksponensial setelah $lag (k-p)$	Menurun secara eksponensial setelah $lag (p-k)$

2. Melakukan proses estimasi

Proses estimasi merupakan proses pendugaan parameter untuk model ARIMA. Untuk mempermudah, proses eliminasi biasanya dilakukan dengan bantuan program, salah satunya dengan program MINITAB.

3. Melakukan proses diagnosik

Proses diagnosik yaitu mengevaluasi model apakah telah memenuhi syarat untuk digunakan. Evaluasi dilakukan dengan melihat apakah pada model terlihat autokorelasi dan residu sudah *white noise* atau bersifat acak. Untuk mengetahui apakah residu bersifat acak atau tidaknya, dapat dilakukan uji korelasi residu dengan melihat plot ACF dan PACF residu. Jika pada grafik ACF dan PACF tidak ada *lag* yang melebihi garis signifikansi, maka residu bersifat acak.

4. Menggunakan model terpilih untuk peramalan.

D. Himpunan Fuzzy

Himpunan klasik secara umum didefinisikan sebagai kumpulan elemen atau objek $x \in X$ yang dapat berupa *finite*, *countable* atau *overcountable*. Misalkan X adalah himpunan semesta dan $x \in X$.

Pada himpunan tegas (*crisp*), nilai keanggotaan suatu item x dalam suatu himpunan A yang biasa ditulis $\mu_A(x)$ memiliki 2 kemungkinan, yaitu:

- a. Satu (1), yang memiliki arti bahwa suatu item menjadi anggota dalam suatu himpunan, atau

- b. Nol (0), yang memiliki arti bahwa suatu item yang tidak menjadi bagian dari suatu himpunan.

Menurut Zimmermann (1991:11-12) jika X adalah kumpulan dari objek yang dinotasikan x , maka himpunan *fuzzy* A dalam X merupakan himpunan pasangan berurutan:

$$A = \{(x, \mu_A(x)) \mid x \in X\} \quad (2.19)$$

Dengan $\mu_A(x)$ adalah fungsi keanggotaan atau derajat keanggotaan dari x pada A yang memetakan X ke ruang anggota M yang terdapat pada interval $[0,1]$.

Contoh 2.1, Jika diketahui $S = [10, 20, 30, 40, 50, 60]$ dalam satuan ribu kwh adalah semesta pembicara, dan himpunan $A = [10, 20, 30]$ dan himpunan $B = [30, 40, 50]$, maka dapat dinyatakan menjadi 2 nilai keanggotaan pada himpunan A , $\mu_A(20) = 1$ karena $20 \in A$ dan nilai keanggotaan 20 pada himpunan B , $\mu_B(20) = 0$ karena $20 \notin A$.

Himpunan *fuzzy* memiliki 2 atribut, yaitu :

- a. Linguistik, yaitu penamaan sesuatu grup yang mewakili suatu keadaan atau kondisi tertentu dengan menggunakan bahasa alami, seperti: muda, tua, tinggi, rendah dll.
- b. Numeris, yaitu suatu nilai (angka) yang menunjukkan ukuran dari suatu variabel seperti, 10, 20, 30.

Selain memiliki 2 atribut, himpunan *fuzzy* juga memiliki beberapa pengertian pada sistem *fuzzy*, yaitu:

- a. Variabel *fuzzy*, adalah variabel yang akan digunakan dan dibahas dalam suatu sistem *fuzzy*, seperti umur, temperatur, dll.
- b. Himpunan *fuzzy*, adalah grup yang mewakili kondisi atau keadaan tertentu dalam suatu variabel *fuzzy*.
- c. Semesta pembicara, adalah keseluruhan nilai yang diperbolehkan untuk digunakan dalam suatu variabel *fuzzy*.
- d. Domain, adalah daerah nilai yang diizinkan dan dapat dioperasikan dalam suatu himpunan *fuzzy*.

E. Logika *Fuzzy*

Logika *fuzzy* pertama kali diperkenalkan oleh Zadeh pada tahun 1965. Zadeh merupakan Profesor di bidang ilmu komputer, Universitas California, Berkeley. Zadeh beranggapan bahwa logika benar salah tidak dapat mewakili setiap pemikiran manusia. Perbedaan mendasar dari logika *fuzzy* dan logika *crisp* adalah keanggotaan elemen dalam suatu himpunan, jika dalam himpunan *fuzzy*, keanggotaan elemen berada pada selang [0,1]. Dan pada logika *crisp* keanggotaan elemen mempunyai dua pilihan, yaitu bernilai 1 untuk yang terdapat pada suatu himpunan dan bernilai 0 untuk yang tidak terdapat pada suatu himpunan (Kusumadewi & Purnomo, 2010:158).

Logika *fuzzy* adalah suatu cara yang tepat untuk memetakan suatu ruang *input* ke dalam suatu ruang *output*. Dalam logika *fuzzy* terdapat beberapa proses yang ada seperti himpunan *fuzzy*, fungsi keanggotaan, operasi dasar himpunan *fuzzy* dan penalaran dalam himpunan *fuzzy*. Seiring dengan perkembangan jaman, logika *fuzzy* sangat diminati di berbagai bidang. Hal ini dikarenakan

logika *fuzzy* dapat mewakili setiap keadaan atau mewakili pemikiran manusia. Selain itu alasan lain digunakannya logika *fuzzy* adalah (Kusumadewi Purnomo, 2010:154):

1. Konsep logika *fuzzy* mudah dimengerti.
2. Logika *fuzzy* sangat fleksibel.
3. Logika *fuzzy* memiliki toleransi terhadap data-data yang tidak tepat.
4. Logika *fuzzy* mampu memodelkan fungsi-fungsi nonlinear yang sangat kompleks.
5. Logika *fuzzy* dapat membangun dan mengaplikasikan pengalaman-pengalaman para pakar secara langsung tanpa harus melalui proses pelatihan.
6. Logika *fuzzy* dapat bekerjasama dengan teknik-teknik endala secara konvesional.
7. Logika *fuzzy* didasarkan pada bahasa alami.

F. Fungsi keanggotaan

Fungsi keanggotaan adalah kurva yang menunjukkan pemetaan titik-titik *input* data ke dalam nilai keanggotaannya (derajat keanggotaan) yang memiliki interval antara 0 sampai dengan 1.

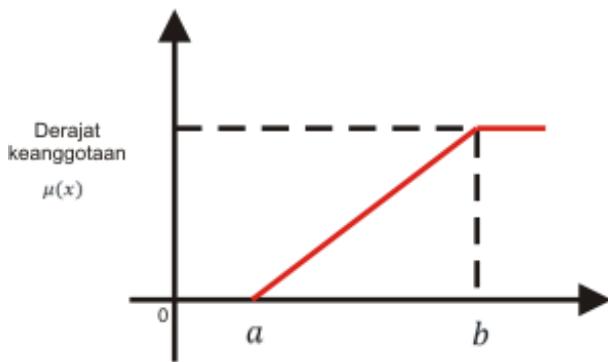
Pada fungsi keanggotaan, ada beberapa jenis fungsi yang digunakan, yaitu:

a. Representasi linear

Pada representasi linear terdapat 2 kemungkinan himpunan *fuzzy linear*, yaitu:

1. Representasi Linear Naik

Kenaikan himpunan dimulai pada nilai domain yang memiliki derajat keanggotaan nol [0] bergerak ke kanan menuju nilai domain yang memiliki derajat keanggotaan lebih tinggi. Grafik representasi kurva linear naik ditunjukkan pada Gambar 2.7.

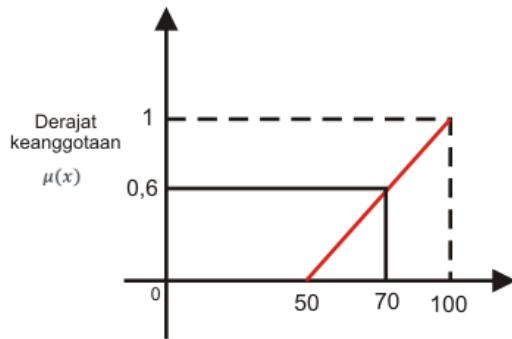


Gambar 2.7. Grafik Representasi Linear Naik

Fungsi keanggotaannya :

$$\mu[x] = \begin{cases} 0; & x \leq a \\ \frac{x-a}{b-a}, & a \leq x \leq b \\ 1; & x > b \end{cases} \quad (2.20)$$

Contoh 2.2 Terdapat variabel kebutuhan listrik dengan semesta pembicara 0 sampai 100 juta kwh. Terdapat suatu himpunan kebutuhan listrik TINGGI dengan domain antara 50 hingga 100 juta kwh. Maka representasi linear naik dapat dilihat pada Gambar 2.8.



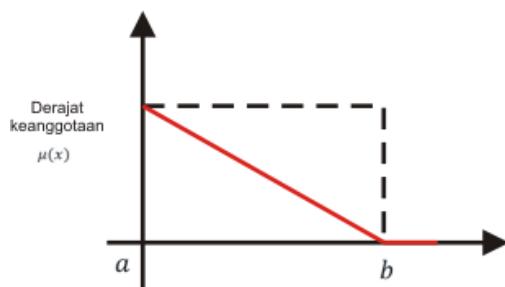
Gambar 2.8 Contoh Representasi Linear Naik

Kemudian akan ditentukan derajat keanggotaan untuk nilai $x = 70$, didapatkan

$$\mu_{TINGGI}[70] = \frac{70 - 50}{100 - 70} = \frac{20}{30} = 0,6$$

2. Representasi Linear Turun

Garis lurus dimulai dari nilai domain dengan derajat keanggotaan tertinggi pada sisi kiri, kemudian bergerak menurun ke nilai domain yang memiliki derajat keanggotaan lebih rendah. Grafik representasi linear turun ditunjukkan pada Gambar 2.9.

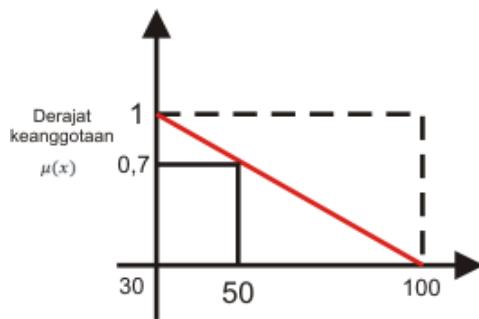


Gambar 2.9. Grafik Representasi Linear Turun

Fungsi keanggotaan:

$$\mu[x] = \begin{cases} \frac{(b-x)}{(b-a)}; & a < x \leq b \\ 0; & x > b \\ 1; & x \leq a \end{cases} \quad (2.21)$$

Contoh 2.3 Terdapat kebutuhan variabel listrik dengan semesta pembicara antara 30 juta hingga 100 juta kwh. Terdapat suatu himpunan kebutuhan listrik RENDAH dengan domain 50 juta hingga 100 juga kwh. Maka representasi linear turun dapat dilihat pada Gambar 2.10.



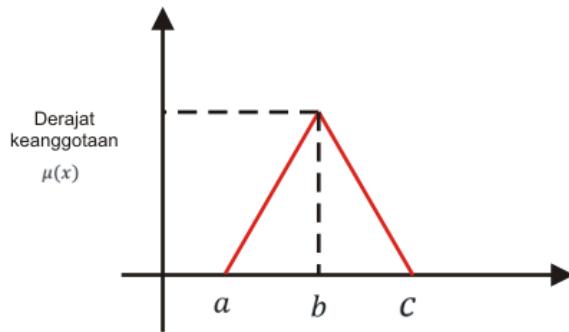
Gambar 2.10 Contoh Representasi Linear Turun

Kemudian akan ditentukan nilai derajat keanggotaan untuk $x = 50$, menggunakan persamaan 2.20 didapatkan

$$\mu_{RENDAH}[50] = \frac{100 - 50}{100 - 30} = \frac{50}{70} = 0,7$$

b. Representasi segitiga

Representasi segitiga merupakan gabungan dari dua garis linear, yaitu linear naik dan linear turun. Kurva segitiga hanya memiliki satu nilai x , dengan derajat keanggotaan tertinggi. Nilai x memiliki nilai tersebut ketika $x = b$. Nilai tersebut tersebar dipersekutaran b dan memiliki perubahan derajat keanggotaan menurun dengan menjauhi 1. Grafik reprsentasi kurva segitiga ditunjukkan pada Gambar 2.11.

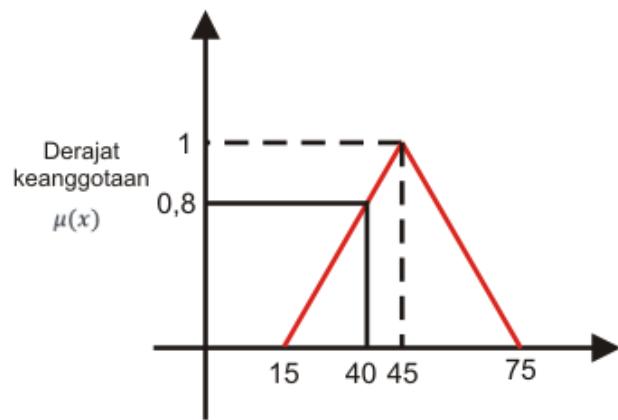


Gambar 2.11. Grafik Representasi Kurva Segitiga

Fungsi keanggotaan :

$$\mu[x] = \begin{cases} 0; & x \leq a \text{ dan } x > c \\ \frac{(x-a)}{(b-a)}; & a < x \leq b \\ \frac{(c-x)}{(c-b)}; & b < x \leq c \end{cases} \quad (2.22)$$

Contoh 2.4 Terdapat suatu himpunan kebutuhan listrik NORMAL dengan domain 15 juta hingga 75 juga kwh. Maka representasi linear turun dapat dilihat pada Gambar 2.12.



Gambar 2.12 Contoh Representasi Linear Segitiga

Kemudian akan ditentukan derajat keanggotaan untuk $x = 40$. Dengan menggunakan persamaan 2.21 didapatkan:

$$\mu_{NORMAL}[40] = \frac{40 - 15}{56 - 26} = \frac{25}{30} = 0,8$$

G. Operator-Operator pada Himpunan Fuzzy

Terdapat dua model operator *fuzzy* yang sering digunakan, yaitu operator yang dikemukakan oleh Zadeh dan operator alternatif yang merupakan perkembangan konsep transformasi tertentu.

Berikut merupakan beberapa operator dasar yang diciptakan oleh Zadeh (Kusumadewi & Hartati , 2010 : 175) yaitu:

1. Operator AND

Operator ini berhubungan dengan operator interseksi pada himpunan. α -predikat sebagai hasil operasi dengan operator AND diperoleh dengan mengambil nilai keanggotaan terkecil antar elemen pada himpunan-himpunan bersangkutan

$$\mu_{A \cap B} = \min(\mu_A[x], \mu_B[y]) \quad (2.23)$$

Contoh 2.5 Misalkan nilai keanggotaan 900 kwh pada himpunan RENDAH adalah 0,4 ($\mu_{RENDAH}[900] = 0,4$), dan nilai keanggotaan Rp. 2.000,00 pada himpunan harga listrik MURAH adalah 0,8 ($\mu_{MURAH}[2 \times 100] = 0,7$). Maka α -predikat untuk kebutuhan listrik RENDAH dan harga MURAH adalah:

$$\begin{aligned}\mu_{RENDAH} \cap \mu_{MURAH} &= \min(\mu_{RENDAH}[900], \mu_{MURAH}[2 \times 100]) \\ &= \min(0,4; 0,7) \\ &= 0,4\end{aligned}$$

2. Operator OR

Operator OR berhubungan dengan gabungan pada himpunan. α -predikat sebagai hasil operasi dengan operator OR diperoleh dengan

mengambil nilai keanggotaan terkecil antar elemen pada himpunan-himpunan yang bersangkutan

$$\mu_{A \cup B} = \max (\mu_A[x], \mu_B[y]) \quad (2.24)$$

Contoh 2.6 Pada Contoh 2.5 sebelumnya dapat dihitung nilai α -predikat untuk kebutuhan listrik RENDAH atau harga listrik MURAH adalah:

$$\begin{aligned} \mu_{RENDAH} \cup \mu_{MURAH} &= \max(\mu_{RENDAH}[900], \mu_{MURAH}[2 \times 100]) \\ &= \max(0,4; 0,7) \\ &= 0,7 \end{aligned}$$

3. Operator NOT

Operator ini berhubungan dengan operator komplemen himpunan. α -predikat sebagai hasil operasi dengan operator NOT diperoleh dengan mengurangkan nilai keanggotaan elemen pada himpunan yang bersangkutan dari 1. Operasi NOT didefinisikan sebagai berikut:

$$\mu_A = 1 - \mu_A[x] \quad (2.25)$$

Contoh 2.7 Pada Contoh 2.5 dapat ditentukan nilai α -predikat untuk permintaan listrik TINGGI adalah:

$$\begin{aligned} \mu_{TINGGI}[900] &= 1 - \mu_{RENDAH}[900] \\ &= 1 - 0,4 \\ &= 0,4 \end{aligned}$$

H. Defuzzifikasi (Penegasan)

Input dari defuzzifikasi adalah suatu himpunan yang diperoleh dari komposisi aturan-aturan *fuzzy*, sedangkan *output* yang dihasilkan merupakan suatu bilangan pada domain himpunan *fuzzy* tersebut.

Pada proses defuzzifikasi ini terdapat beberapa metode yang dapat digunakan, yaitu:

1. Metode **Largest of Maximum (LOM)**

Solusi tegas didapatkan dengan mengambil nilai terbesar dari domain yang memiliki nilai keanggotaan maksimum.

Contoh 2.5 Misal didapatkan hasil peramalan pada bulan Oktober 2013 yaitu $y_{92} = [0,3700559 \ 0,5109261847 \ 0,242147597]$, selanjutnya akan diambil nilai maksimum dari himpunan y_{92} . Nilai 0,5109261847 berada pada himpinan kedua. Fungsi keanggotaan A_2 adalah

$$\mu_{A_2}(x) = \begin{cases} 0 & 129489577,6 \leq x \\ \frac{x - 129489577,6}{37496463,43} & 129489577,6 < x \leq 166986041 \\ \frac{204482504,4 - x}{37496463,43} & 166986041 < x \leq 204482504,4 \\ 0 & x > 204482504,4 \end{cases}$$

Untuk pengubahan dari bilangan *fuzzy* ke bilangan crips dapat dilakukan dengan mensubtitusikan hasil terbesar yang didapat yaitu 0,5109261847 pada fungsi seperti yang diatas, didapatkan,

$$\mu_{A_2}(x) = \frac{x - 129489577,6}{37496463,43}$$

$$0,5109261847 = \frac{x - 129489577,6}{37496463,43}$$

$$x = 148647502,6$$

dan

$$\mu_{A_2}(x) = \frac{204482504,4 - x}{37496463,43}$$

$$0,5109261847 = \frac{204482504,4 - x}{37496463,43}$$

$$x = 185324579,4$$

Dari perhitungan diatas terdapat 2 nilai crips yang dihasilkan. Pada metode defuzzifikasi *Largest of Maximum Defuzzier* maka hasil yang digunakan adalah 185324579,4 Kwh.

2. Metode *Smallest of Maximum (SOM)*

Solusi tegas didapatkan dengan mengambil nilai terkecil dari domain yang memiliki nilai keanggotaan maksimum.

Contoh 2.6 Sama dengan soal pada **Contoh 2.5** hasil yang digunakan dari himpunan y_{92} adalah 0,242147597. Nilai 0,242147597 terletak pada himpunan A_3 , dengan fungsi keanggotaan:

$$\mu_{A_3}(x) = \begin{cases} 0 & 157611925,1 \leq x \\ \frac{x - 157611925,1}{37496463,43} & 157611925,1 < x \leq 195108388,6 \\ \frac{232604852 - x}{37496463,43} & 195108388,6 < x \leq 232604852 \\ 0 & x > 232604852 \end{cases}$$

Untuk pengubahan dari bilangan fuzzy ke bilangan crips dapat dilakukan dengan mensubtitusikan hasil terbesar yang didapat yaitu 0,242147597 pada fungsi seperti yang diatas, didapatkan,

$$\mu_{A_3}(x) = \frac{x - 157611925,1}{37496463,43}$$

$$0,242147597 = \frac{x - 157611925,1}{37496463,43}$$

$$x = 204188067$$

dan

$$\mu_{A_3}(x) = \frac{232604852 - x}{37496463,43}$$

$$0,242147597 = \frac{232604852 - x}{37496463,43}$$

$$x = 223525173$$

Dari perhitungan diatas terdapat 2 nilai crisp yang dihasilkan. Pada metode defuzzifikasi *Smallest of Maximum Defuzzier* maka hasil yang digunakan adalah 204188067 Kwh.

3. Metode *Mean of Maximum (MOM)*

Solusi tegas didapatkan dengan mengambil nilai rata-rata domain fuzzy yang memiliki nilai keanggotaan maksimum.

Contoh 2.7 Pada contoh 2.5 didapatkan 2 nilai crisp yaitu 148647502,6 dan 185324579,4. Pada metode *Mean of Maximum* maka nilai crisp yang digunakan adalah

$$x = \frac{148647502,6 + 185324579,4}{2} = 166986041 \text{ kwh}$$

Jadi nilai penyelesaian berupa nilai crisp adalah 166986041 kwh

I. Neural Network (NN)

Neural Network adalah proses sistem informasi yang memiliki beberapa karakteristik mirip dengan jaringan saraf biologi. NN dikembangkan sebagai generalisasi model matematika dari jaringan saraf biologi, dengan asumsi bahwa (Fausett, 1994: 3):

- a. Proses informasi terjadi pada banyak elemen sederhana (*neuron*).
- b. Sinyal dikirimkan diantara *neuron-neuron* melalui penghubung-penghubung.

- c. Penghubung antara *neuron* memiliki bobot yang akan memperkuat atau memperlemah sinyal, dan
- d. Untuk menentukan *output*, setiap *neuron* menggunakan fungsi aktivasi yang dikenakan pada jumlahan *input* yang diterima.

NN terdiri dari unsur-unsur sederhana yang beroperasi secara paralel.

Jaringan ini dapat ditentukan dengan hubungan antar unsur-unsurnya. Umumnya jaringan dapat dilatih untuk melakukan fungsi tertentu dengan menyesuaikan nilai-nilai bobot antar unsur-unsurnya untuk mencapai *output* atau target tertentu (Demuth & Beale, 1992: 18). NN dapat diaplikasikan di berbagai bidang terutama pada analisis yang lebih kompleks seperti masalah *nonlinear* atau suatu struktur paralel. NN digunakan untuk menyelesaikan masalah peramalan (Hu & Hwang, 2001:240).

Pada NN, *neuron-neuron* akan dikumpulkan dalam lapisan-lapisan (*layer*) yang disebut lapisan *neuron* (*neuron layer*). Pada umumnya, NN mempunyai tiga lapisan, yaitu (Yeung *et al*, 1998:3):

a. Lapisan *Input* (*Input Layer*)

Node-node di dalam lapisan *input* disebut *neuron-neuron input*. *Neuron-neuron input* menerima *input* berupa gambaran informasi atau permasalahan dari luar.

b. Lapisan Tersembunyi (*Hidden Layer*)

Node-node di dalam lapisan tersembunyi disebut *neuron-neuron* tersembunyi. *Neuron-neuron* pada lapisan tersembunyi ini berisi bobot yang

selanjutnya akan digunakan sebagai perhitungan untuk menentukan nilai *output*.

c. Lapisan Output (*Output Layer*)

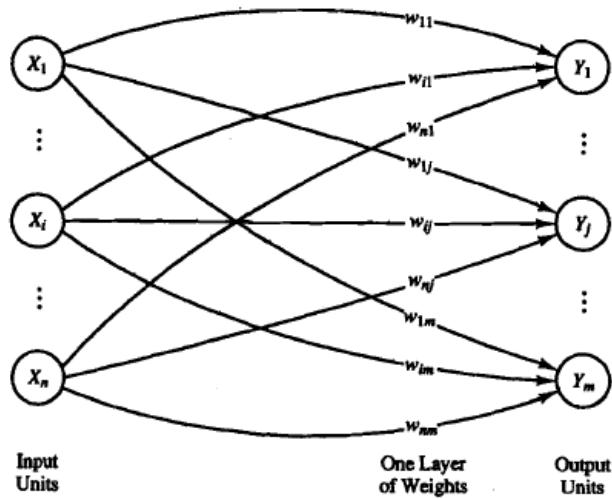
Node-node di dalam lapisan *output* disebut *neuron-neuron output*. Keluaran dari lapisan ini merupakan hasil dari NN terhadap suatu permasalahan.

1. Arsitektur Jaringan

Pengaturan *neuron* dalam setiap lapisan dan pola hubungan antar lapisan disebut arsitektur jaringan saraf. Arsitektur NN diklasifikasikan menjadi 3, yaitu (Fausett, 1994: 12-14):

a. Jaringan Lapisan Tunggal (*Single Layer Net*)

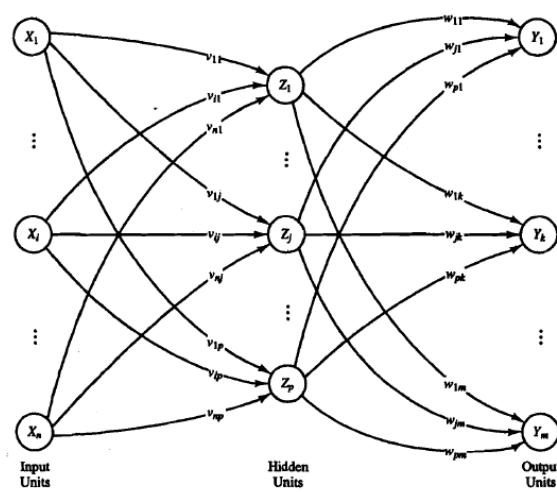
Jaringan dengan lapisan tunggal memiliki satu lapisan dengan bobot-bobot terhubung. Jaringan ini hanya menerima *input* kemudian secara langsung akan mengolahnya menjadi *output* tanpa harus melalui lapisan lain (lapisan tersembunyi). Dengan kata lain, ciri-ciri dari arsitektur jaringan saraf lapisan tunggal adalah hanya memiliki satu lapisan *input* dan satu lapisan *output*. Gambar 2.13 merupakan contoh dari jaringan lapisan tunggal.



Gambar 2.13. Contoh Jaringan Lapisan Tunggal
(Sumber: Fausett, 1994: 13)

b. Jaringan Banyak Lapisan (*Multilayer Net*)

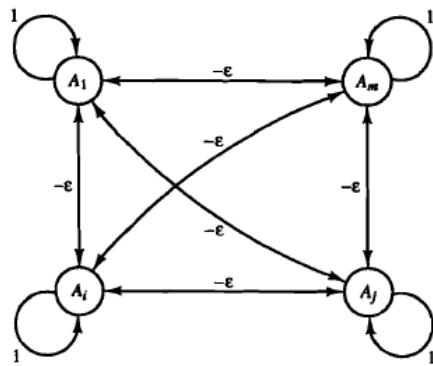
Jaringan dengan banyak lapisan memiliki satu atau lebih lapisan yang berada diantara lapisan *input* dan lapisan *output* (terdapat satu atau lebih lapisan tersembunyi). Jaringan dengan banyak lapisan ini dapat menyelesaikan permasalahan yang lebih sulit dibandingkan dengan lapisan tunggal. Gambar 2.14 menunjukkan contoh jaringan banyak lapisan.



Gambar 2.14. Contoh Jaringan Banyak Lapisan
(Sumber: Fausett, 1994: 13)

c. Jaringan Lapisan Kompetitif (*Competitive Layer Net*)

Pada jaringan ini, antar *neuron* dapat saling dihubungkan. Gambar 2.15. menunjukkan contoh arsitektur jaringan lapisan kompetitif.



Gambar 2.15. Contoh Jaringan Lapisan Kompetitif
(Sumber: Fausett, 1994: 14)

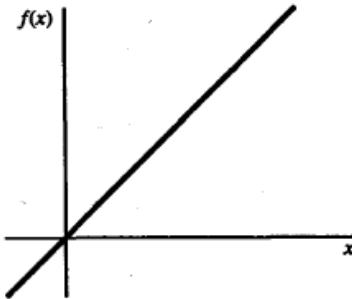
2. Fungsi Aktivasi

Fungsi aktivasi adalah fungsi yang digunakan untuk menentukan keluaran suatu *neuron*. Fungsi aktivasi berguna untuk mengaktifkan atau menonaktifkan *neuron* yang digunakan pada jaringan. Terdapat beberapa fungsi aktivasi yang sering digunakan pada NN sebagai berikut.

- a. Fungsi linear memiliki nilai *output* yang sama dengan nilai *inputnya*.

Fungsi linear ditunjukkan pada Gambar 2.16. Fungsi linear dirumuskan sebagai berikut (Fausett, 1994: 17):

$$y = x, \quad \text{untuk semua } x \quad (2.26)$$



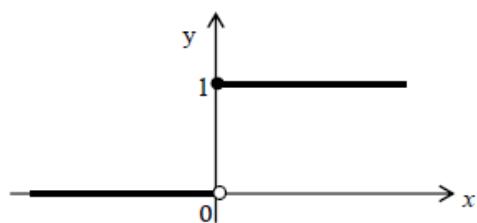
Gambar 2.16. Fungsi Aktivasi Linear

b. Fungsi Undak Biner (*Hard Limit*)

Jaringan dengan lapisan tunggal sering menggunakan fungsi undak (*step function*) untuk mengkonversikan *input* dari suatu variabel yang bernilai kontinu ke suatu *output* biner (0 atau 1). Fungsi undak biner (*hard limit*) dirumuskan sebagai berikut:

$$y = f(x) = \begin{cases} 0, & \text{jika } x < 0 \\ 1, & \text{jika } x \geq 0 \end{cases} \quad (2.27)$$

Grafik fungsi aktivasi undak biner terdapat pada Gambar 2.17.



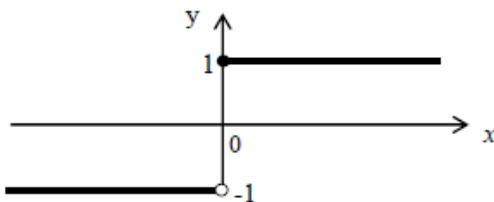
Gambar 2.17 Fungsi Aktivasi Undak Biner (*Hard Limit*)

c. Fungsi Bipolar (*Symmetric Hard Limit*)

Fungsi bipolar mirip dengan fungsi undak biner, perbedaannya terdapat pada nilai *output* yang dihasilkan. Nilai *output* bipolar berupa nilai 1 dan -1. Fungsi bipolar dirumuskan sebagai berikut:

$$y = f(x) = \begin{cases} -1, & \text{jika } z < 0 \\ 1, & \text{jika } x \geq 0 \end{cases} \quad (2.28)$$

Grafik fungsi bipolar terdapat pada Gambar 2.18 berikut.



Gambar 2.18 Fungsi Aktivasi Bipolar (*Symmetric Hard Limit*)

3. Metode Pembelajaran (*Learning Method*)

Salah satu bagian terpenting dari konsep NN adalah terjadinya proses pembelajaran. Tujuan utama dari proses pembelajaran adalah melakukan pengaturan terhadap bobot-bobot yang ada pada jaringan saraf, sehingga didapatkan bobot akhir yang tepat sesuai pola data yang dilatih. Pada dasarnya, metode pembelajaran dibagi menjadi 2, yaitu metode pembelajaran terawasi (*supervised learning*) dan metode pembelajaran tidak terawasi (*unsupervised learning*).

a. Pembelajaran Terawasi (*Supervised Learning*).

Pada proses pembelajaran ini, satu *input* yang telah diberikan pada satu *neuron* di lapisan *input* akan dijalankan sepanjang jaringan saraf sampai ke *neuron* pada lapisan *output*. Hasil yang diperoleh kemudian dicocokkan dengan target, jika terjadi perbedaan, maka akan muncul *error*. Jika *error* cukup besar, akan dilakukan pembelajaran yang lebih banyak lagi. Beberapa contoh *supervised learning* adalah *Hebbian*, *Perceptron*, *Adaline*, *Boltzman*, dan *Backpropagation*.

b. Pembelajaran Tidak Terawasi (*Unsupervised Learning*)

Pada proses pembelajaran ini, nilai bobot disusun dalam suatu interval atau *range* tertentu tergantung dari nilai *input* yang diberikan. Pembelajaran ini bertujuan mengelompokkan unit-unit yang hampir sama dalam suatu area tertentu. Beberapa contoh *unsupervised learning* adalah *Competitive*, *Kohonen* dan *LVQ* (*learning Vector Quantization*).

J. Listrik

Listrik adalah daya atau kekuatan yang ditimbulkan oleh adanya pergesekan atau melalui proses kimia, yang dapat digunakan untuk menghasilkan panas, cahaya, atau untuk menjalankan mesin. Pada awal ditemukannya, listrik hanyalah berasal dari batu ambar yang digosok-gosokkan kemudian dapat menarik bulu. Penemu ini ditemukan oleh Thales antara tahun 546-640 M, tetapi pada awal ditemukannya Thales belum mengetahui bahwa yang ditemukannya adalah listrik. Kemudian dari penemuan tersebut dikembangkan oleh Willian Gilbert (1733) yang mengatakan bahwa peristiwa Thales merupakan elektrik. Kata elektrik diambil dari bahasa Yunani yaitu elektron atau batu ambar. Selanjutnya listrik dikembangkan terus menerus hingga pada masa Michael Faraday.

Di Indonesia, PLN selaku penyalur utama listrik ke masyarakat secara tidak langsung telah menjadi tulang punggung bagi perkembangan perekonomian masyarakat. Kerangka perekonomian yang terdiri atas berbagai jenis lapisan masyarakat merupakan suatu variabel yang saling terkait dimana

satu bagiannya akan bergantung kepada bagian lainnya sehingga roda perekonomian tetap berjalan.

Kebutuhan listrik dari hari ke-hari semakin bertambah dikarenakan listrik tidak hanya digunakan sebagai penerangan melainkan sebagai penggerak alat-alat pengganti tenaga manusia pada industri besar maupun industri kecil. Masing-masing konsumen tersebut memiliki jumlah kebutuhan listrik yang berbeda-beda dikarenakan tingkat kebutuhannya. Semisal pada industri rumahan menengah keatas, kebutuhan listrik terbilang lumayan besar dikarenakan listrik digunakan untuk menggerakkan peralatan mesin yang digunakan untuk produksi.