

**PERAMALAN DATA DERET WAKTU MENGGUNAKAN ALGORITMA
*BACKPROPAGATION NEURAL NETWORK (BNN)***

(Skripsi)

Oleh

Novita Veronika
NPM 1817031064



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2022**

ABSTRAK

PERAMALAN DATA DERET WAKTU MENGGUNAKAN ALGORITMA BACKPROPAGATION NEURAL NETWORK (BNN)

Oleh

NOVITA VERONIKA

Jaringan Syaraf Tiruan (JST) merupakan pengolahan informasi yang terinspirasi oleh cara kerja sistem syaraf biologis dalam memproses informasinya. ANN sering digunakan dalam peramalan data runtun waktu yang salah satunya yaitu algoritma *Backpropagation Neural Network* (BNN). BNN adalah metode penurunan gradien untuk meminimalkan *error* yang memiliki dua tahap perhitungan. Dengan menggunakan algoritma BNN dapat dilatih untuk menganalisa pola data masa lalu dengan *output* yang diinginkan pada saat ini. Dari hasil analisis yang telah dilakukan menggunakan algoritma BNN menghasilkan model terbaik yaitu MSE 0.0002452 dan MAPE 5.1165471 pada fungsi aktivasi *tanh* di *hidden layer* ke-2 dengan pembagian data pelatihan 70% (1534 data) dan data pengujian 30% (658 data). Peramalan dengan analisis ini menghasilkan harga emas terbesar pada 23 Juni 2022 sebesar Rp873.639,00 dan harga emas terkecil pada 26 Juni 2022 sebesar Rp841.742,00.

Kata kunci: *jaringan syaraf tiruan, backpropagation, peramalan, harga emas*

ABSTRACT

TIME SERIES DATA FORECASTING USING BACKPROPAGATION NEURAL NETWORK (BNN) ALGORITHMS

By

NOVITA VERONIKA

Artificial Neural Network (ANN) is information processing that is inspired by how the biological nervous system works in processing information. ANN is often used in time-continuous data forecasting, one of which is the algorithm *Backpropagation Neural Network* (BNN). BNN is a gradient drop method to minimize errors that have two calculation stages. Using the BNN algorithm can be trained to analyze past data patterns with the desired output at the present moment. From the results of the analysis that has been carried out using the BNN algorithm, it produces the best model, that is MSE 0.0002452 and MAPE 5.1165471 on the activation function *tanh* in *hidden layer* 2 with training data 70% (1534 data) and testing data 30% (658 data). Forecasting with this analysis resulted in the largest gold price on June 23, 2022 of IDR 873.639 and the smallest gold price on June 26, 2022 was IDR 841,742.

Keywords: *artificial neural network, backpropagation, forecasting, gold price*

**PERAMALAN DATA DERET WAKTU MENGGUNAKAN ALGORITMA
BACKPROPAGATION NEURAL NETWORK (BNN)**

Oleh

Novita Veronika

Skripsi

Sebagai Salah Satu Syarat untuk Mencapai Gelar
SARJANA MATEMATIKA

Pada

Jurusan Matematika
Fakultas Matematika dan Ilmu Pengetahuan Alam
Universitas Lampung



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2022**

Judul Skripsi

: **PERAMALAN DATA DERET WAKTU
MENGUNAKAN ALGORITMA
BACKPROPAGATION NEURAL NETWORK
(BNN)**

Nama Mahasiswa

: **Novita Veronika**

Nomor Pokok Mahasiswa : **1817031064**

Jurusan

: **Matematika**

Fakultas

: **Matematika dan Ilmu Pengetahuan Alam**



1. **Komisi Pembimbing**

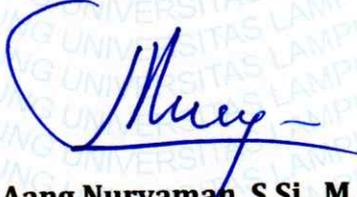

Drs. Nusyirwan, M.Si.

NIP 19661010 199205 1 001


Agus Sutrisno, S.Si., M.Si.

NIP 19700831 199903 1 002

2. **Ketua Jurusan Matematika**


Dr. Aang Nuryaman, S.Si., M.Si.

NIP 19740316 200501 1 001

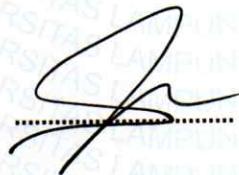
MENGESAHKAN

1. Tim Penguji

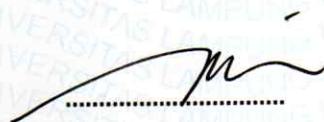
Ketua : Drs. Nusyirwan, M.Si.



Sekretaris : Agus Sutrisno, S.Si., M.Si.



**Penguji
Bukan Pembimbing : Ir. Netti Herawati, M.Sc., Ph.D.**



2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam



Dr. Eng. Surtpto Dwi Yuwono, S.Si., M.T.
NIP 19740705 200003 1 001



Tanggal Lulus Ujian Skripsi : 22 Juni 2022

PERNYATAAN SKRIPSI MAHASISWA

Saya bertanda tangan di bawah ini:

Nama : **Novita Veronika**
Nomor Pokok Mahasiswa : **1817031064**
Jurusan : **Matematika**
Judul : **Peramalan Data Deret Waktu Menggunakan
Algoritma *Backpropagation Neural Network*
(BNN)**

Dengan ini menyatakan bahwa skripsi ini adalah hasil pekerjaan saya sendiri dan semua tulisan yang tertuang dalam skripsi ini telah mengikuti kaidah karya penulisan ilmiah Universitas Lampung.

Bandar Lampung, 22 Juni 2022

Penulis,



Novita Veronika

RIWAYAT HIDUP

Penulis bernama lengkap Novita Veronika, anak kedua dari empat bersaudara yang dilahirkan di Jakarta pada tanggal 22 November 1999 oleh pasangan Bapak Drs. Pidel Lumbantoruan, M.M. dan Ibu Lisda Sianturi.

Penulis menempuh pendidikan di TK Santo Fransiskus yang diselesaikan pada tahun 2006, kemudian melanjutkan sekolah di SD Santo Fransiskus III yang diselesaikan pada tahun 2012, kemudian melanjutkan sekolah di SMP Negeri 193 Jakarta, dan kemudian melanjutkan sekolah di SMA Negeri 102 Jakarta pada tahun 2018.

Pada tahun 2018 penulis terdaftar sebagai Mahasiswa Program Studi S1 Matematika Fakultas Matematika dan Ilmu pengetahuan Alam Universitas Lampung melalui jalur SBMPTN. Selama menjadi mahasiswa penulis pernah bergabung menjadi Staf Hubungan Luar BEM FMIPA Unila 2019, penulis juga pernah aktif menjadi Anggota Bidang Minat dan Bakat Himpunan Matematika (HIMATIKA) 2020, dan penulis juga pernah aktif menjadi Anggota Divisi Pelayanan dan Doa (Divisi 3) UKM Kristen Unila 2021.

Pada awal tahun 2021, penulis melaksanakan Kuliah Kerja Nyata (KKN) selama 40 hari di Kelurahan Kebon Pala, Kecamatan Makasar, Jakarta Timur, DKI Jakarta. Pada pertengahan tahun 2021, penulis melaksanakan Kerja Praktik (KP) di Badan Pusat Statistika (BPS) Pusat.

KATA INSPIRASI

*“Janganlah hendaknya kamu kuatir tentang apa pun juga,
tetapi nyatakanlah dalam segala hal keinginanmu kepada Allah
dalam doa dan permohonan dengan ucapan syukur”*

(Filipi 4 : 6)

*“Apapun juga yang kamu perbuat,
perbuatlah dengan segenap hatimu
seperti untuk Tuhan dan bukan untuk manusia”*

(Kolose 3 : 23)

*“Bersyukurlah kepada Tuhan, sebab Ia baik!
Bahwasanya untuk selama-lamanya kasih setia-Nya”*

(Mazmur 136 : 1)

PERSEMBAHAN

Puji Tuhan atas segala berkat-Nya dan anugerah-Nya yang telah menjadi dasar pengharapan sehingga penulis dapat menyelesaikan skripsi ini.

Kupersembahkan skripsi ini kepada Papa dan Mama yang sudah berkorban banyak hal sehingga aku memiliki kesempatan untuk bahagia bersama kalian. Hanya karena doa dan didikan kalian yang membawaku bertahan dan kuat sampai sejauh ini.

Juga kepada abangku dan adik-adikku, terima kasih telah mendoakan dan selalu memberikan dukungan kepada aku.

Serta teman-temanku yang memberikan begitu banyak dukungan, doa, canda dan tawa dalam perkuliahaan. Semua hal yang sudah kita lewati akan kusimpan selalu.

SANWACANA

Puji syukur kepada Tuhan Yang Maha Esa yang telah memberikan karunia serta kasihNya sehingga penulis dapat menyelesaikan skripsi yang berjudul “**Peramalan Data Deret Waktu Menggunakan Algoritma *Backpropagation Neural Network* (BNN)**”.

Dalam proses penyusunan skripsi ini, banyak pihak yang telah membantu memberikan bimbingan, dukungan, motivasi, serta saran sehingga skripsi ini dapat terselesaikan. Untuk itu penulis ingin mengucapkan terima kasih yang tak terhingga kepada:

1. Bapak Drs. Nusyirwan, M.Si., selaku Pembimbing I yang selalu bersedia memberikan kesediaan waktu, arahan, bimbingan, dan dukungan yang membangun dalam proses penyusunan skripsi ini.
2. Bapak Agus Sutrisno, S.Si., M.Si., selaku Pembimbing II atas kesediaan waktu, saran, dan bimbingan selama proses penyusunan skripsi ini.
3. Ibu Ir. Netti Herawati, M. Sc. Ph.D., selaku Penguji yang telah bersedia memberikan kritik dan saran serta evaluasi kepada penulis sehingga dapat lebih baik lagi.
4. Bapak Prof. Dr. La Zakaria, S.Si., M.Sc., selaku dosen Pembimbing Akademik yang selalu bersedia memberikan bimbingan, saran serta dukungan kepada penulis pada hal yang berkaitan dengan akademik.
5. Bapak Dr. Aang Nuryaman, S.Si., M.Si., selaku Kepala Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung

6. Bapak Dr. Eng. Suripto Dwi Yuwono, S.Si., M.T., selaku Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.
7. Seluruh Dosen, Staf Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.
8. Papa dan Mama atas cinta, kasih sayang, dukungan dan selalu mendoakan kepada penulis dalam menyelesaikan perkuliahan.
9. Sandro Alfredo, Michael Agustinus, dan Jerico Immanuel yang selalu memberikan dukungan dan doa kepada penulis.
10. Sahabat-sahabatku yang selalu menemani selalu menemani disaat suka maupun duka di masa perkuliahan, Laurensia Klarisa, Skolastika Ria, Ramona Rahmawati, Anisa Dwi Putri.
11. Sahabat yang selalu menemani dan memahami keluh kesah penulis dalam perkuliahan, Rully Marlan Eliezer Sitanggang, Dea Carolin, Valerie Ixion, Sarah Laura, Evelyn, Yemima, Herlambang Sihombing, Rexi Soaloon.
12. *Last but not least, I wanna thank me, I wanna thank me for believing in me, I wanna thank me for doing all this hard work, I wanna thank me for having no days off, I wanna thank me for never quitting, just being me at all times.*

Kiranya semoga skripsi ini dapat memberikan sumbangan yang berarti bagi semua pihak yang berkepentingan. Penulis menyadari dalam penyusunan skripsi masih memiliki kekurangan, karena itu kritik dan saran yang membangun diharapkan untuk penyempurnaan skripsi.

Bandar Lampung, 22 Juni 2022

Penulis,

Novita Veronika

DAFTAR ISI

	Halaman
DAFTAR TABEL.....	xv
DAFTAR GAMBAR.....	xvi
I. PENDAHULUAN.....	1
1.1 Latar Belakang dan Masalah.....	1
1.2 Tujuan Penelitian.....	3
1.3 Manfaat Penelitian.....	3
II. TINJAUAN PUSTAKA.....	4
2.1 Data Deret Waktu.....	4
2.1.1 Pola Data Deret Waktu.....	4
2.1.2 Autokorelasi.....	6
2.2 Peramalan.....	8
2.3 Pembagian Data.....	9
2.3.1 <i>Data Training</i>	9
2.3.2 <i>Data Testing</i>	10
2.4 Normalisasi Data.....	10
2.4.1 <i>Min Max Normalization</i>	11
2.4.2 <i>Z-Score Normalization</i>	11
2.4.3 <i>Decimal Scaling</i>	12
2.5 Proses Pembelajaran.....	12
2.5.1 <i>Supervised Learning</i> (Pembelajaran Terawasi).....	13
2.5.2 <i>Unsupervised Learning</i> (Pembelajaran Tidak Terawasi).....	13
2.6 <i>Artifical Neural Network</i> (ANN).....	14
2.7 <i>Backpropagation Neural Network</i> (BNN).....	15
2.8 Fungsi Aktivasi.....	21
2.8.1 Fungsi Aktivasi <i>Sigmoid Biner</i>	22
2.8.2 Fungsi Aktivasi <i>Rectified Linear Unit</i> (ReLU).....	23
2.8.3 Fungsi Aktivasi <i>Tanh</i>	23

2.9	Denormalisasi.....	24
2.10	Ketepatan Model.....	25
2.11	Emas.....	26
III.	METODOLOGI PENELITIAN.....	27
3.1	Waktu dan Tempat Penelitian.....	27
3.2	Data Penelitian.....	27
3.3	Metode Penelitian.....	28
IV.	HASIL DAN PEMBAHASAN.....	29
4.1	Statistika Deskriptif.....	29
4.2	Menentukan Input Jaringan Berdasarkan <i>Lag</i> yang Signifikan.....	30
4.3	Melakukan Normalisasi Data.....	32
4.4	Pembagian Data.....	33
4.5	Fungsi Aktivasi yang Digunakan.....	34
4.6	Proses Pelatihan dan Pengujian.....	34
	4.6.1 Proses Pelatihan.....	35
	4.6.2 Proses Pengujian.....	35
4.7	Denormalisasi Data.....	42
4.8	Peramalan.....	43
V.	KESIMPULAN.....	45
	DAFTAR PUSTAKA.....	46
	LAMPIRAN.....	49

DAFTAR TABEL

Tabel	Halaman
1. Simbol-Simbol Algoritma <i>Backpropagation Neural Network</i>	20
2. Karakteristik Data Harga Harian Emas Dunia	30
3. Data Input Jaringan	32
4. Normalisasi Data	33
5. Pembagian Data <i>Training & Testing</i>	34
6. Hasil Evaluasi Fungsi Aktivasi <i>Sigmoid Biner</i>	36
7. Hasil Evaluasi Fungsi Aktivasi ReLU	38
8. Hasil Evaluasi Fungsi Aktivasi <i>Tanh</i>	40
9. Hasil Peramalan Data Harga Harian Emas	43

DAFTAR GAMBAR

Gambar	Halaman
1. Berbagai Plot Data Deret Waktu	5
2. Plot <i>Autocorrelation Function</i> (ACF)	7
3. Plot <i>Partial Autocorrelation Function</i> (PACF)	8
4. Arsitektur Jaringan Syaraf Tiruan (JST)	14
5. Arsitektur <i>Backpropagation Neural Network</i>	16
6. Fungsi Aktivasi <i>Sigmoid Biner</i>	22
7. Fungsi Aktivasi ReLU	23
8. Fungsi Aktivasi <i>Tanh</i>	24
9. Plot Data Harga Harian Emas Dunia	29
10. Plot Data Emas <i>Autocorrelation Function</i> (ACF)	31
11. Plot Data Emas <i>Partial Autocorrelation Function</i> (PACF)	31
12. Plot Perbandingan Data Prediksi dan Aktual dari Denormalisasi	42

I. PENDAHULUAN

1.1 Latar Belakang dan Masalah

Artificial Neural Network (ANN) atau Jaringan Syaraf Tiruan (JST) merupakan sebuah pendekatan pengolahan informasi yang terinspirasi oleh cara kerja sistem syaraf biologis dalam memproses informasi. ANN bermanfaat untuk pengenalan pola, *signal processing*, pengklasifikasian dan peramalan (Siang, 2009). ANN terdiri atas kumpulan neuron yang tersusun dalam beberapa lapisan yaitu lapisan input (*input layer*), lapisan tersembunyi (*hidden layer*) dan lapisan *output* (*output layer*). Dalam penerapannya, analisis ANN sering digunakan dalam prediksi atau peramalan suatu data runtun waktu. Salah satunya dalam melakukan analisis prediksi atau peramalan sering digunakan algoritma *Backpropagation Neural Network*.

Backpropagation Neural Network (BNN) adalah metode penurunan gradient untuk meminimalkan *error* yang diterapkan pada *Artificial Neural Network* (ANN) dengan pasangan input (x) dan *output* (y) sebagai *training data*. Algoritma *Backpropagation Neural Network* bekerja dengan dua tahap perhitungan yaitu perhitungan maju pada saat menghitung nilai kesalahan (*error*) antara hasil nilai *output* yang dikeluarkan dengan nilai yang seharusnya dan perhitungan mundur untuk memperbaiki bobot berdasarkan nilai *error* tersebut. Dengan menggunakan algoritma *Backpropagation Neural Network* dapat dilatih untuk mempelajari dan menganalisa pola data masa lalu dan berusaha mencari suatu formula atau fungsi yang menghubungkan pola data masa

lalu dengan *output* yang diinginkan pada saat ini (Salman & Prasetio, 2010). Algoritma *Backpropagation Neural Network* menggunakan *error* pada nilai *output* untuk mengganti nilai bobot-bobotnya pada arah mundur. Pada saat perambatan maju, neuron-neuron diaktifkan dengan menggunakan fungsi aktivasi terbaik berdasarkan tingkat akurasi model. Fungsi aktivasi digunakan untuk menentukan keluaran suatu neuron (Siang, 2009).

Peramalan adalah suatu teknik untuk meramalkan atau memprediksi keadaan di masa yang datang melalui pengujian keadaan di masa lalu. Dasarnya meramalkan samahalnya dengan memprediksi atau memperkirakan suatu hal, kejadian atau peristiwa masa datang yang berdasarkan data pada masa lalu (runtun waktu) hingga saat ini. Dalam melakukan peramalan analisis deret waktu dibutuhkan model statistik yang tepat untuk menjelaskan suatu data deret waktu.

Model yang sudah dilakukan dengan menggunakan algoritma *Backpropagation Neural Network* digunakan dalam melakukan peramalan data deret waktu. Metode peramalan sangat banyak dan seringkali memerlukan beberapa asumsi-asumsi yang harus dipenuhi, namun terdapat juga model yang tidak memerlukan asumsi-asumsi yang salah satunya adalah *Neural Network* (Rinjani, Hoyyi, & Suparti, 2019). Oleh karena itu, penulis ingin melakukan analisis menggunakan algoritma *Backpropagation Neural Network* (BNN) pada data deret waktu untuk melakukan peramalan harga harian emas dunia di Indonesia.

1.2 Tujuan Penelitian

Tujuan dari penulisan penelitian ini adalah:

1. Menentukan model *Backpropagation Neural Network* (BNN) berdasarkan tingkat akurasi terbaik pada fungsi aktivasinya.
2. Meramalkan harga harian emas dengan menggunakan hasil dari fungsi aktivasi terbaik.

1.3 Manfaat Penelitian

Manfaat yang didapatkan dalam penelitian ini adalah:

1. Mengembangkan wawasan keilmuan dan pengetahuan tentang analisis *Backpropagation Neural Network* (BNN).
2. Dapat menjadi referensi bagi pembaca apabila ingin melakukan penelitian mengenai peramalan dengan menggunakan analisis *Backpropagation Neural Network* (BNN).

II. TINJAUAN PUSTAKA

2.1 Data Deret Waktu

Data deret waktu diartikan sebagai rangkaian pengamatan yang berurutan terhadap suatu variabel berdasarkan waktu dengan interval waktu yang tetap (Wei, 2006). Data deret waktu merupakan analisis peramalan atau prediksi menggunakan hubungan antar variabel yang diperkirakan dengan deret waktu. Runtun waktu adalah pengamatan pada suatu variabel dari waktu lampau dan dicatat secara berurutan menurut urutan waktu dengan periode yang tetap (Hanke & Wichern, 2004).

2.1.1 Pola Data Deret Waktu

Langkah penting dalam memilih metode pada peramalan yang tepat yaitu mempertimbangkan jenis pola data, sehingga metode peramalan yang paling sesuai dengan pola tersebut dapat diterapkan. Pola data dibagi menjadi 4 yaitu horizontal, musiman, sklis dan tren dengan masing-masing pola data mempunyai kriteria masing-masing (Aden, 2020):

a. Horizontal

Pola data horizontal adalah pola data yang menetap pada rata-rata yang konstan. Contohnya seperti data penjualan barang dan jasa yang tidak mengalami peningkatan dan penurunan dalam jangka waktu yang ditentukan.

b. Musiman

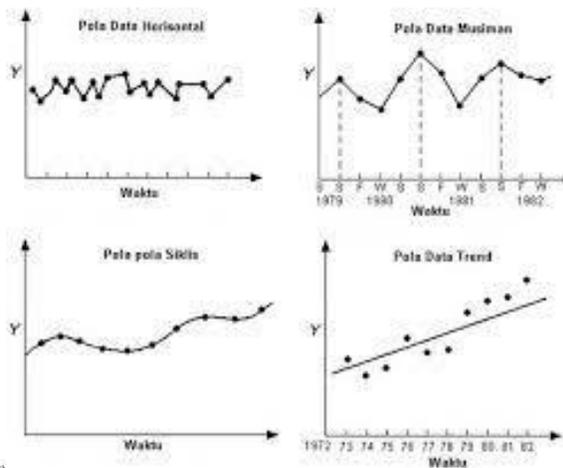
Seperti namanya, pola data musiman adalah data yang meningkat atau menurun pada waktu tertentu dikarenakan suatu faktor. Contohnya seperti produk yang laku keras menjelang hari perayaan tertentu.

c. Siklis

Pola data siklis dipengaruhi oleh fluktuasi dalam jangka panjang. Contohnya seperti data penjualan produk utama selama beberapa tahun, produk utama ini seperti rumah dan mobil.

d. Tren

Pola data tren adalah data yang naik atau turun dalam jangka waktu yang panjang. Contohnya seperti data jumlah pemilik industri kecil yang terus naik dari tahun ke tahun.



Gambar 1. Berbagai Pola Data Deret Waktu.

2.1.2 Autokorelasi

Definisi autokorelasi yaitu terjadinya korelasi antar data di dalam data deret waktu yang dipisahkan dalam *lag* (selang waktu). Artinya, jika $lag = 2$, maka nilai t akan dikorelasikan dengan $t + 2$ (data pertama akan dikorelasikan dengan data ketiga), jika menggunakan $lag = 4$, maka nilai t akan dikorelasikan dengan $t + 4$ (data pertama akan dikorelasikan dengan data kelima), dan begitu seterusnya. Hal ini biasanya ditulis dengan rumus korelasi (X_t, X_{t+k}) dengan k adalah jumlah *lag* dan t adalah nilainya. Untuk mendefinisikan autokorelasi diperlukan definisi autokovarians. Autokovarians dan autokorelasi antara X_t dan X_{t+k} berturut-turut dapat didefinisikan sebagai berikut (Wei, 2006):

$$\gamma_k = Cov(X_t, X_{t+k}) = E[(X_t - \mu)(X_{t+k} - \mu)] \quad (2.1)$$

dan

$$\rho_k = r_k \frac{Cov(X_t, X_{t+k})}{\sqrt{Var(X_t)}\sqrt{Var(X_{t+k})}} = \frac{\gamma_k}{\gamma_0} \quad (2.2)$$

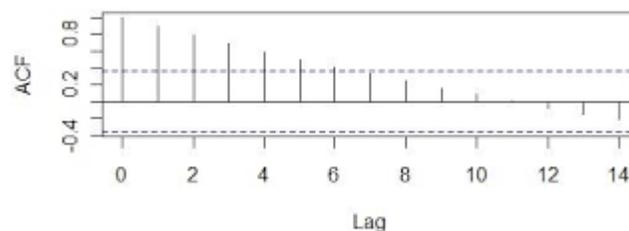
dengan:

- ρ_k = autokorelasi pada *lag-k*
- γ_k = autokovarians pada *lag- k*
- t = waktu pengamatan, $t = 1,2,3, \dots$
- X_t = pengamatan pada waktu ke- t
- X_{t+k} = pengamatan pada waktu ke- $t + k$.

Dimana $Var(X_t) = Var(X_{t+k}) = \gamma_0$ dan $\rho_0 = 1$. Sebagai fungsi dari k , γ_k disebut fungsi autokovarians dan ρ_k disebut sebagai fungsi autokorelasi, yang mewakili kovarians dan korelasi antara X_t dan X_{t+k} dari proses yang sama, hanya dipisahkan oleh *lag-k*. Jika X_t dan X_{t+k} independen maka $\gamma_k = Cov(X_t, X_{t+k}) = 0$ tetapi tidak berlaku sebaliknya. X_t dan X_{t+k} dikatakan tidak berkorelasi jika $\rho_k = 0$. Hubungan negatif sempurna dengan dua variabel memiliki koefisien korelasi sebesar -1 dan

hubungan positif sempurna dengan dua variabel memiliki koefisien korelasi sebesar +1. Dengan demikian, koefisien korelasi bervariasi antara -1 dan +1 (Hanke & Winchern, 2004).

Fungsi autokorelasi atau *Autocorrelation Function* (ACF) merupakan suatu fungsi yang menunjukkan besarnya korelasi atau hubungan linear antara pengamatan pada waktu t saat sekarang dengan pengamatan pada waktu-waktu sebelumnya ($t - 1, t - 2, \dots, t - k$).



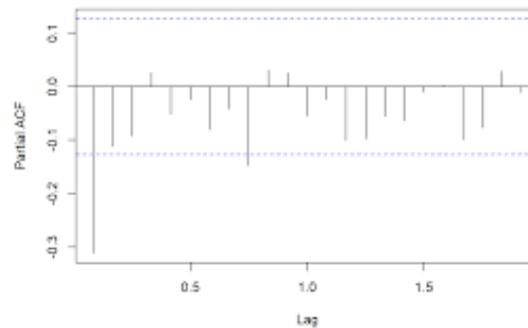
Gambar 2. Plot *Autocorrelation Function* (ACF).

Pada gambar diatas menunjukkan plot fungsi autokorelasi pada data deret waktu dimana *lag* yang keluar dari batas (daerah interval) menunjukkan autokorelasi signifikan. Pola dari koefisien autokorelasi sering digunakan untuk menetapkan ada tidaknya faktor musiman di dalam data deret waktu dan menentukan order model data deret waktu yang tepat pada situasi tertentu, serta menentukan kestasioneran data.

Rumus perhitungan fungsi autokorelasi yaitu (Machmudin & Ulama, 2012):

$$\hat{\rho}_k = \frac{\hat{\gamma}_k}{\hat{\gamma}_0} = \frac{\sum_{t=1}^{n-k} (Z_t - \bar{Z})(Z_{t+k} - \bar{Z})}{\sum_{t=1}^n (Z_t - \bar{Z})^2}, k = 0, 1, 2, \dots \quad (2.3)$$

dengan nilai rata-rata dari deret waktu ($\bar{Z} = \sum_{t=1}^n \frac{Z_t}{n}$) dapat didefinisikan \bar{Z} .



Gambar 3. Plot *Partial Autocorrelation Function* (PACF).

Pada gambar diatas menunjukkan plot fungsi autokorelasi parsial atau *Partial Autocorrelation Function* (PACF). Fungsi autokorelasi parsial digunakan untuk mengukur tingkat keeratan antara X_t dan X_{t+k} , apabila pengaruh dari *lag* 1, 2, 3, . . . , dan seterusnya sampai $k - 1$ dianggap terpisah. Pada PACF, pengawalan perhitungan nilai dimulai dengan $\widehat{\phi}_{kk} = \widehat{\rho}_1$, dimana $\widehat{\rho}_1$ nilai dari autokorelasi *lag* pertama. Berikut ini adalah persamaan untuk menghitung nilai fungsi autokorelasi parsial *lag-k* (Machmudin & Ulama, 2012):

$$\widehat{\phi}_{kk} = \frac{\widehat{\rho}_1 \sum_{j=1}^{k-1} \phi_{k-1j} \rho_{k-j}}{1 - \sum_{j=1}^{k-1} \phi_{k-1j} \rho_{k-j}} \quad (2.4)$$

dengan:

$\widehat{\phi}_{kk}$ = nilai PACF pada *lag* ke-k

ρ_k = nilai ACF pada *lag* ke-k.

2.2 Peramalan

Peramalan adalah proses memperkirakan nilai di masa mendatang dengan menggunakan data yang ada di masa lampau (Hyndman & Athanasopoulos, 2014).

Data di masa lampau secara sistematis dikombinasikan dan diolah untuk memperkirakan suatu nilai di masa mendatang. Peramalan dalam deret waktu merupakan prediksi untuk memperkirakan kejadian-kejadian yang terjadi di masa yang datang berdasarkan data-data sebelumnya (Wei, 2006). Prediksi sebagai langkah untuk meramal kejadian sangat dibutuhkan di berbagai kegiatan sehari-hari khususnya di bidang ekonomi yang biasanya digunakan untuk mengetahui tingkat penjualan, harga, maupun hal lain pada perusahaan dengan tujuan meminimalisir kemungkinan buruk yang terjadi di masa mendatang. Dalam Jaringan Syaraf Tiruan (JST) terdapat teknik peramalan yang sering digunakan yaitu *Backpropagation*.

2.3 Pembagian Data

Dalam pembuatan model *Backpropagation Neural Network* (BNN), tentunya dibutuhkan data. Sekumpulan data yang digunakan dalam jaringan syaraf tiruan dapat disebut data set, yang kemudian dibagi menjadi dua yaitu *training* dataset dan *testing* dataset. Beberapa proporsi pembagian data yang pada pembagian datanya berfungsi untuk mencari pembagian data terbaik yaitu menggunakan 50:50, 60:40, 70:30, dan 80:20 (Utami & Ulama, 2015).

2.3.1 Data Training

Data *training* adalah data pelatihan yang nantinya dapat digunakan untuk melatih algoritma dan juga mencari model yang cocok. Tahapan ini dimulai dengan memasukan data latih ke dalam jaringan (Warsito, 2009). Dengan menggunakan data latihan, jaringan mengubah bobot yang menjadi penghubung antar neuron-neuron. Pada setiap evaluasi dilakukan evaluasi terhadap *output* jaringan. Tahapan ini berlangsung pada beberapa iterasi dan berhenti setelah menemukan bobot yang sesuai

dimana nilai *error* yang diinginkan telah tercapai atau jumlah iterasi telah mencapai nilai maksimal yang ditetapkan, selanjutnya bobot ini menjadi dasar pengetahuan pada tahapan pengenalan.

2.3.2 Data Testing

Data *testing* atau data pengujian adalah bagian dataset yang kita uji untuk melihat keakuratannya atau performanya. Data pengujian ini hanya mencakup data masukan, bukan data keluaran yang diharapkan. Data pengujian digunakan untuk menilai seberapa baik algoritma yang dilatih dan memperkirakan model untuk prediksi data deret waktu. Pada proses *testing*, algoritma diuji dengan menggunakan data *testing* dan data *training* yang merupakan dua data berbeda. Pengujian dilakukan dengan memasukan suatu pola yang belum pernah dilatih sebelumnya (data uji) dengan menggunakan bobot hasil tahap pelatihan. Diharapkan bobot-bobot hasil pelatihan yang telah menghasilkan *error* minimal juga menghasilkan *error* minimal pada tahap pengujian (Warsito, 2009). Data *training* digunakan dalam proses pembentukan arsitektur terbaik dalam proses pelatihan, sedangkan data *testing* digunakan dalam pengujian keakuratan dari arsitektur yang telah terbentuk.

2.4 Normalisasi Data

Menurut Hidayat, dkk. (2012), data yang dilakukan normalisasi dengan membagi nilai data tersebut dengan nilai *range* data (nilai data maksimum – nilai data minimum). Normalisasi data adalah mentransformasi data ke dalam *range* 0 dan 1. Tujuan dari normalisasi yaitu menghilangkan kerangkapan data, merubah nilai menjadi satu satuan dan mempermudah pemodifikasian data.

2.4.1 *Min-Max Normalization*

Menurut Hanifah, dkk. (2017), *min-max normalization* merupakan metode normalisasi dengan melakukan transformasi linear terhadap data asli sehingga menghasilkan keseimbangan nilai perbandingan antar data saat sebelum dan sesudah proses. Metode ini dapat menggunakan rumus sebagai berikut:

$$x'_t = \frac{x_t - \min(x)}{\max(x) - \min(x)} \quad (2.5)$$

dimana:

x'_t = data hasil normalisasi ke- t

x_t = data sebelum normalisasi ke- t

$\min(x)$ = nilai minimum dari data sebelum dinormalisasi

$\max(x)$ = nilai maksimum dari data sebelum dinormalisasi.

2.4.2 *Z-Score Normalization*

Z-score normalization merupakan metode normalisasi berdasarkan nilai rata-rata dan standar deviasi dari data. Metode ini sangat berguna jika tidak diketahui nilai minimum dan maksimum dari data. Rumus yang digunakan yaitu:

$$x'_t = \frac{x_t - \mu}{\sigma} \quad (2.6)$$

dimana:

μ = rata-rata data data sebelum dinormalisasi

σ = standar deviasi data sebelum dinormalisasi.

2.4.3 *Decimal Scaling*

Decimal scaling merupakan metode normalisasi dengan menggerakkan nilai desimal dari data ke arah yang diinginkan. Normalisasi data dengan teknik ini dilakukan dengan membagi setiap nilai pada data dengan nilai absolut maksimum pada data. Nilai x_t pada data dinormalisasi menjadi x'_t dengan menggunakan rumus yaitu:

$$x'_t = \frac{x_t}{10^j} \quad (2.7)$$

dengan:

j = nilai integer terkecil dengan nilai maksimum kurang dari 1 $Max(|x'_t|) < 1$.

2.5 Proses Pembelajaran

Pada proses pembelajaran, informasi yang dilewatkan dari satu neuron ke neuron yang lainnya melalui dendrit. Jika rangsangan tersebut diterima oleh suatu neuron, maka neuron tersebut membangkitkan *output* ke semua neuron yang berhubungan dengannya sampai informasi tersebut sampai ke tujuannya yaitu terjadinya suatu reaksi. Jika rangsangan yang diterima terlalu halus, maka *output* yang dibangkitkan oleh neuron tersebut tidak direspons. Perubahan yang terjadi selama proses pembelajaran adalah perubahan nilai bobot. Nilai bobot bertambah jika informasi yang diberikan oleh neuron tersampaikan, sebaliknya jika informasi tidak disampaikan oleh suatu neuron ke neuron yang lain, maka nilai bobot yang menghubungkan keduanya dikurangi. Pada saat pembelajaran dilakukan pada input yang berbeda, maka nilai bobot diubah secara dinamis hingga mencapai suatu nilai yang cukup seimbang. Apabila nilai ini telah tercapai maka tiap-tiap input telah berhubungan dengan *output* yang diharapkan.

2.5.1 *Supervised Learning* (Pembelajaran Terawasi)

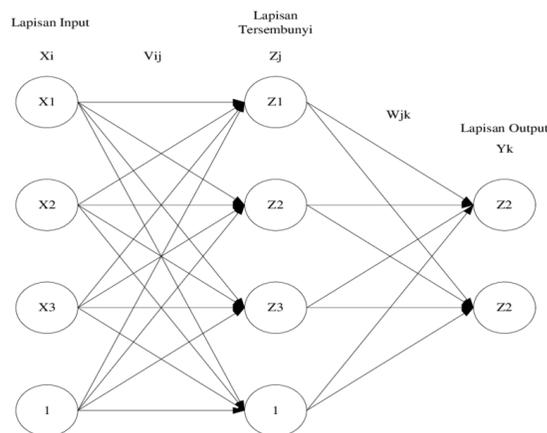
Supervised learning atau biasa disebut pembelajaran terawasi merupakan teknik yang membuat suatu fungsi dari data yang dipelajari di masa lalu untuk memprediksi peristiwa di masa depan. Pada pembelajaran terawasi menggunakan metode dengan adanya latihan dan pelatih yang memiliki target. Pada proses pembelajaran ini, satu pola input diberikan ke satu neuron pada lapisan input. Pola ini dirambatkan di sepanjang jaringan syaraf hingga sampai ke neuron pada lapisan *output*. Lapisan *output* ini membangkitkan pola *output* yang nantinya dicocokkan dengan pola *output* targetnya. Apabila terjadi perbedaan antara pola *output* hasil pembelajaran dengan pola target, maka disini akan muncul *error* dan jika nilai *error* ini masih cukup besar, maka masih perlu dilakukan lebih banyak pembelajaran lagi hingga mendapatkan nilai *error* terkecil. Contoh algoritma pembelajaran terawasi yaitu KNN, *Naive Bayes*, *Decision Trees*, *Regresi linier*, *Support Vector Machine*, *Backpropagation Neural Network*.

2.5.2 *Unsupervised Learning* (Pembelajaran Tidak Terawasi)

Unsupervised Learning atau biasa dikenal pembelajaran tidak terawasi merupakan algoritma yang tidak membutuhkan data berlabel dalam memprediksi target melainkan menggunakan kesamaan dari atribut yang dimiliki. Jika atribut dan sifat-sifat dari data yang dianalisis memiliki kemiripan, maka dikelompokkan (*clustering*). Sehingga hal ini menimbulkan kelompok-kelompok (klaster). Dari kelompok-kelompok itu model dilabelkan dan jika dilakukan prediksi, maka data dicocokkan dengan kelompok yang mirip. Pembelajaran ini biasanya sangat cocok untuk pengelompokan (klasifikasi) pola.

2.6 Artificial Neural Network (ANN)

Artificial Neural Network (ANN) atau Jaringan Syaraf Tiruan (JST) adalah sistem pemroses informasi yang mempunyai karakteristik mirip dengan jaringan syaraf biologi. Ide mendasar dari jaringan syaraf tiruan ini adalah mengadopsi mekanisme berpikir sebuah sistem atau aplikasi yang menyerupai otak manusia, baik untuk pemrosesan berbagai sinyal elemen yang diterima dan toleransi terhadap kesalahan. JST dibentuk sebagai generalisasi model matematika dari jaringan syaraf biologi dengan asumsi bahwa pada pemrosesan informasi terjadi pada banyak elemen sederhana (neuron). Sinyal dikirimkan di antara neuron-neuron melalui penghubung-penghubung di mana penghubung antar neuron memiliki bobot yang memperkuat atau memperlemah sinyal dan untuk menentukan *output*. Setiap neuron menggunakan fungsi aktivasi yang dikenakan pada jumlahan input yang diterima. Besarnya *output* ini selanjutnya dibandingkan dengan suatu batas ambang.



Gambar 4. Arsitektur Jaringan Syaraf Tiruan (JST).

Jaringan syaraf tiruan yang ditentukan oleh pola hubungan antar neuron (disebut arsitektur jaringan). Jaringan syaraf tiruan menggunakan metode untuk menentukan

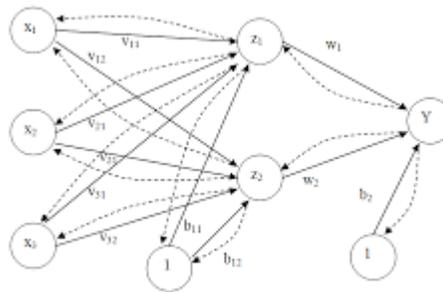
bobot penghubung (disebut metode *training*) dan fungsi aktivasi (Siang, 2009). Jaringan Saraf Tiruan (JST) dengan layer tunggal memiliki keterbatasan dalam pengenalan pola. Kelemahan ini bisa ditanggulangi dengan menambahkan satu atau beberapa lapisan tersembunyi di antara lapisan masukan dan keluaran. Meskipun penggunaan lebih dari satu lapisan tersembunyi memiliki kelebihan manfaat untuk beberapa kasus, tapi pelatihannya memerlukan waktu yang lama. Seperti halnya model jaringan syaraf tiruan lainnya, algoritma *Backpropagation* melatih jaringan untuk mendapatkan keseimbangan antara kemampuan jaringan dengan tujuan mengenali pola yang digunakan selama pelatihan serta memberikan respons yang benar terhadap pola masukan yang serupa dengan pola yang dipakai selama pelatihan.

2.7 Backpropagation Neural Network

Backpropagation atau propagasi balik merupakan algoritma yang berfungsi dalam pengenalan pola, klasifikasi, pengolahan citra, dan pengambilan keputusan. *Backpropagation* juga salah satu dari jaringan syaraf tiruan yang merupakan metode pelatihan yang terawasi (*Supervised Learning*) dengan jaringan *multilayer* dan memiliki ciri khusus meminimalkan *error* pada *output* yang dihasilkan oleh jaringan. Dikatakan sebagai algoritma pelatihan *multilayer* karena memiliki tiga lapisan dalam proses pelatihannya, yaitu lapisan *input*, lapisan tersembunyi dan lapisan *output*, dimana algoritma ini merupakan perluasan dari jaringan layer tunggal yang memiliki dua lapisan, yaitu lapisan *input* dan lapisan *output*. *Backpropagation Neural Network* memiliki lapisan tersembunyi yang dapat menyebabkan besarnya tingkat *error* pada *Backpropagation* lebih kecil dibanding tingkat *error* pada jaringan layer tunggal. Hal tersebut dikarenakan lapisan tersembunyi pada *Backpropagation* berfungsi sebagai tempat untuk menyesuaikan bobot, sehingga didapatkan nilai bobot yang baru yang bisa diarahkan mendekati dengan target *output* yang diinginkan.

Lapisan-lapisan penyusun jaringan syaraf tiruan *Backpropagation* dibagi menjadi tiga bagian (Widiastuti, Kaswidjanti, & Rustamaji, 2014):

- a. Lapisan input (*input layer*) yaitu neuron di dalam lapisan input yang disebut unit-unit input yang mulai dari lapisan input pertama hingga lapisan input n. Unit-unit input tersebut menerima pola inputan data dari luar yang menggambarkan suatu permasalahan. Semua neuron pada lapisan input ini dapat terhubung ke neuron pada lapisan tersembunyi (*hidden layer*) atau dapat langsung ke lapisan luaran (*output layer*) jika neuron tidak menggunakan lapisan tersembunyi.
- b. Lapisan tersembunyi (*hidden layer*) adalah neuron yang berada di dalam lapisan tersembunyi atau bias disebut unit-unit tersembunyi dimana *outputnya* tidak dapat secara langsung diamati.
- c. Lapisan *output* (*output layer*) merupakan neuron dalam lapisan *output* disebut unit-unit output. Output dari lapisan ini merupakan solusi jaringan syaraf tiruan terhadap suatu permasalahan.



Gambar 5. Arsitektur *Backpropagation Neural Network*.

Lapisan input dilambangkan dengan X , lapisan tersembunyi dilambangkan dengan Z , dan lapisan *output* dilambangkan dengan Y . Bobot antara X dan Z dilambangkan dengan “ v ” sedangkan bobot antara Z dan Y dilambangkan dengan “ w ”. Algoritma

Backpropagation Neural Network terdiri dari tiga lapisan, yaitu lapisan input, lapisan tersembunyi dan lapisan *output*. Pada pengambilan input, terlebih dahulu dilakukan inisialisasi bobot, kemudian masuk ke dalam algoritma *Backpropagation* yang terdiri dari komputasi maju yang bertujuan untuk menelusuri besarnya *error* dan komputasi balik untuk menyesuaikan bobot. Pelatihan dengan algoritma *Backpropagation* terdiri dua tahap yaitu perambatan maju dan perambatan mundur. Selama perambatan maju, tiap unit masukan (X_i) menerima sebuah masukan sinyal ini dan sinyal ini diteruskan ke tiap-tiap lapisan tersembunyi $Z_1, Z_2, Z_3, \dots, Z_{1p}$. Tiap unit tersembunyi ini kemudian menghitung aktivasinya dan mengirimkan sinyalnya (Z_j) ke tiap lapisan *output*. Unit-unit lapisan keluaran memberikan tanggapan yang disebut sebagai keluaran jaringan dan saat keluaran jaringan berbeda dengan keluaran yang diharapkan maka keluaran menyebar mundur (*backward*) pada lapisan tersembunyi diteruskan ke unit pada lapisan masukan (Rinjani, Hoyyi, & Suparti, 2019). Oleh karena itu mekanisme pelatihan tersebut dinamakan *Backpropagation*.

Dalam proses pelatihan pada algoritma *Backpropagation* untuk menghitung nilai koreksi bobot diperlukan *learning rate*. Nilai dari *learning rate* ini berada pada *range* 0 sampai dengan 1. Semakin besar nilai *learning rate*, maka ketelitian jaringan semakin berkurang dan semakin kecil nilai *learning rate*, maka ketelitian jaringan semakin besar atau waktu yang diperlukan untuk proses pelatihan semakin lama.

Algoritma *Backpropagation* sering digunakan dalam menyelesaikan masalah peramalan. Hal ini dimungkinkan karena algoritma *Backpropagation* merupakan salah satu jenis pelatihan JST dengan metode pembelajaran terawasi. Pada jaringan ini diberikan sepasang pola yang terdiri atas pola masukan dan pola yang diinginkan. Ketika suatu pola diberikan kepada jaringan, bobot-bobot diubah untuk memperkecil perbedaan pola keluaran dan pola yang diinginkan yang disebut *error*. Pada algoritma *Backpropagation* dilakukan latihan berulang-ulang sehingga mendapatkan

error terkecil dan mendapatkan pola yang diinginkan (Nurmila, Sugiharto, & Sarwoko, 2010).

Backpropagation Neural Network memiliki beberapa langkah yaitu (Siang, 2009):

- a. Inisialisasi bobot (ambil bobot awal dengan nilai random yang cukup kecil).
- b. Tetapkan: maksimum *epoch*, target *error*, dan *learning rate* (α).
- c. Inisialisasi: *Epoch* = 0, *mean square error* = 1.
- d. Langkah-langkah dalam algoritma pelatihan untuk jaringan *Backpropagation* adalah:

Langkah 1 : Menetapkan nilai input.

Langkah 2 : Membuat inisialisasi semua bobot dengan bilangan acak kecil.

Fase I : Propogasi Maju (*Forward Propagation*)

Langkah 3 : Tiap unit masukkan $X_i, i = 1, 2, 3, \dots, p$ menerima sinyal X_i dan meneruskan sinyal tersebut ke semua unit pada lapisan yang ada di atasnya.

Langkah 4 : Tiap-tiap unit lapisan tersembunyi $Y_j, j = 1, 2, 3, \dots, q$ menjumlahkan sinyal-sinyal *input* terbobot :

$$Y_{in_j} = V_{oj} + \sum_{i=1}^n X_i V_{ij} \quad (2.8)$$

Kemudian dihitung nilai keluaran dengan menggunakan fungsi aktivasi. Kemudian kirimkan sinyal tersebut ke semua unit di lapisan atasnya.

Langkah 5 : Hasil fungsi tersebut dikirim ke semua unit pada lapisan di atasnya. Tiap-tiap unit keluaran $Z_k, k = 1, 2, 3, \dots, r$ menjumlahkan sinyal-sinyal *input* terbobot.

$$Z_{in_k} = W_{ok} + \sum_{i=1}^p Z_j W_{jk} \quad (2.9)$$

Kemudian dihitung nilai keluaran dengan menggunakan

fungsi aktivasi. Kemudian kirimkan sinyal tersebut ke semua unit di lapisan atasnya.

Fase II : Propagasi Mundur (*Backpropagation*)

Langkah 6 : Tiap-tiap unit keluaran $Z_k, k = 1, 2, 3, \dots, r$ menerima target pola yang berhubungan dengan pola input pembelajaran, hitung informasi *error*.

$$\delta_k = (t_k - Z_k)f'(Z_in_k) \quad (2.10)$$

$$= (t_k - Z_k)y_k(1 - Y_k) \quad (2.11)$$

Kemudian hitung koneksi nilai bobot yang kemudian digunakan untuk memperbaharui nilai W_{jk} :

$$\Delta W_{jk} = \alpha \delta_k Y_j \quad (2.12)$$

Hitung koreksi nilai bias yang kemudian digunakan untuk memperbaharui nilai W_{ok} :

$$\Delta W_{ok} = \alpha \delta_k \quad (2.13)$$

Kirimkan δ_k ke unit-unit yang ada dilapisan bawahnya.

Langkah 7 : Tiap-tiap unit *hidden* $Y_j (j = 1, 2, 3, \dots, q)$ menjumlahkan delta input (dari unit-unit yang berada pada lapisan di atasnya):

$$\delta_j = \sum_{k=1}^r \delta_k W_{jk} f'(Y_in_j) \quad (2.14)$$

Kemudian nilai tersebut dikalikan dengan nilai turunan dari fungsi aktivasi untuk menghitung informasi kesalahan.

Hitung koreksi nilai bobot yang kemudian digunakan untuk memperbaharui V_{ij} :

$$\Delta V_{ij} = \alpha \delta_j X_i \quad (2.15)$$

Hitung juga koreksi bias (yang nantinya digunakan untuk memperbaiki nilai V_{oj}):

$$\Delta V_{oj} = \alpha \delta_j \quad (2.16)$$

Fase III : Perubahan Bobot

Langkah 8 : Tiap-tiap unit *output* Z_k ($k = 1, 2, 3, \dots, r$) memperbaiki bias dan bobotnya ($j = 1, 2, 3, \dots, q$):

$$W_{jk}(\text{baru}) = W_{jk}(\text{lama}) + \Delta W_{jk} \quad (2.17)$$

$$W_{ok}(\text{baru}) = W_{ok}(\text{lama}) + \Delta W_{ok} \quad (2.18)$$

Tiap-tiap unit *hidden* Y_j ($j = 1, 2, 3, \dots, q$) memperbaiki bias dan bobotnya ($i = 1, 2, 3, \dots, p$):

$$V_{ij}(\text{baru}) = V_{ij}(\text{lama}) + \Delta V_{ij} \quad (2.19)$$

$$V_{oj}(\text{baru}) = V_{oj}(\text{lama}) + \Delta V_{oj} \quad (2.20)$$

Langkah 9 : Menghitung *error*, jika nilai *error* belum lebih kecil dari pada target *error*, maka langkah 2-8 terus dilakukan.

Langkah 10 : Jika telah terpenuhi, maka pelatihan jaringan dapat dihentikan.

Tabel 1. Simbol-Symbol Algoritma *Backpropagation Neural Network*

No.	Simbol	Keterangan
1	i	Jumlah neuron pada <i>input layer</i>
2	j	Jumlah neuron pada <i>hidden layer</i>
3	k	Jumlah neuron pada <i>output layer</i>
4	X_i	Nilai masukan ke- i pada <i>input layer</i> ($i = 1, 2, 3, \dots, n$)
5	Y_j	Nilai aktivasi neuron ke- j tahap umpan maju pada <i>hidden layer</i>
6	Z_k	Nilai aktivasi neuron ke- k tahap umpan maju pada <i>output layer</i>
7	Y_{in_j}	Hasil penjumlahan sinyal masukan neuron ke- j pada <i>hidden layer</i>

Tabel 1 (lanjutan)

No.	Simbol	Keterangan
8	Z_{in_k}	Hasil penjumlahan sinyal masukan neuron ke- k pada <i>output layer</i>
9	V_{oj}	Nilai bobot antara neuron ke- i pada <i>input layer</i> dan neuron ke- j pada <i>hidden layer</i>
10	W_{ok}	Nilai bobot antara neuron ke- j pada <i>hidden layer</i> dan neuron ke- k pada <i>output layer</i>
11	V_{ij}	Nilai bias neuron ke- j pada <i>hidden layer</i>
12	W_{jk}	Nilai bias neuron ke- k pada <i>output layer</i>
13	t_k	Nilai target neuron ke- k pada <i>output layer</i>
14	δ_j	Nilai <i>gradient descent</i> neuron ke- j pada <i>hidden layer</i>
15	δ_k	Nilai <i>gradient descent</i> neuron ke- k pada <i>output layer</i>
16	α	Nilai <i>learning rate</i>

2.8 Fungsi Aktivasi

Fungsi aktivasi merupakan fungsi matematis yang digunakan untuk mendapatkan nilai *output* berdasarkan nilai input dan mengaktifkan atau tidak mengaktifkan neuron. Informasi yang diproses yaitu melalui fungsi aktivasi dan hasil dari proses ini didistribusikan ke neuron di lapisan berikutnya. Syarat fungsi aktivasi dalam *Backpropagation* bersifat kontinu, terdiferensial dengan mudah dan merupakan fungsi yang tidak turun (Nurmila, Sugiharto, & Sarwoko, 2010).

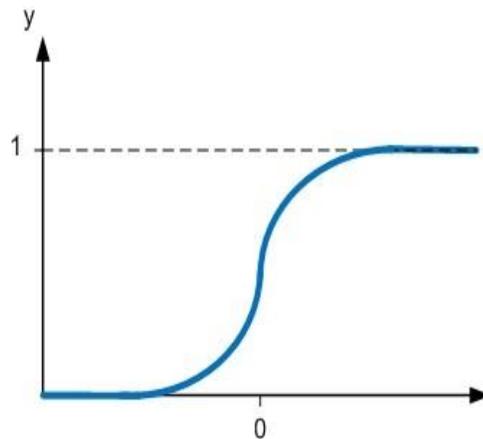
2.8.1 Fungsi Aktivasi *Sigmoid Biner*

Fungsi aktivasi *sigmoid biner* digunakan untuk jaringan syaraf tiruan yang menghasilkan nilai *output* yang terletak pada interval 0 sampai 1. Oleh karena itu, fungsi ini sering digunakan untuk jaringan syaraf tiruan yang membutuhkan nilai *output* yang terletak pada interval 0 sampai 1. Fungsi ini dirumuskan sebagai berikut (Gema & Kartika, 2018):

$$y = f(x) = \frac{1}{1 + e^{-x}} \quad (2.21)$$

dengan:

$$f'(x) = f(x)[1 - f(x)] \quad (2.22)$$

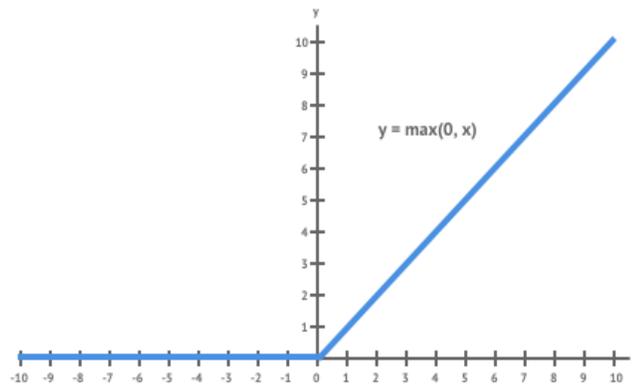


Gambar 6. Fungsi Aktivasi *Sigmoid Biner*.

2.8.2 Fungsi Aktivasi *Rectified Linear Unit (ReLU)*

Fungsi aktivasi *Rectified Linear Unit (ReLU)* merupakan fungsi aktivasi yang digunakan dalam pemodelan jaringan syaraf tiruan. Fungsi kembali ke angka 0 jika *output* menerima input yang nilai negatif, tetapi untuk nilai positif fungsi kembali ke nilai aktivasi itu sendiri. Konvergensi proses pelatihan jika menggunakan fungsi ReLU lebih cepat dibandingkan dengan fungsi *tanh*. Fungsi aktivasi ReLU menggunakan fungsi $\max(0, x)$. Fungsi aktivasi ReLU dapat dilihat dari persamaan (Sitepu & Sigiro, 2021):

$$f(x) = \begin{cases} x, & x \geq 0 \\ 0, & x < 0 \end{cases} \quad (2.23)$$



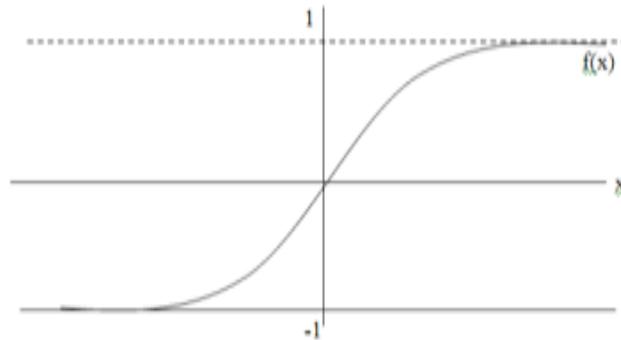
Gambar 7. Fungsi Aktivasi Relu.

2.8.3 Fungsi Aktivasi *Tanh*

Fungsi aktivasi *tanh* merupakan fungsi yang biasa disebut dengan fungsi aktivasi *sigmoid bipolar* dan memiliki nilai *output* antara -1 sampai 1. Rentang nilai dari

fungsi aktivasi *tanh* lebih luas dari fungsi aktivasi *sigmoid biner*. Fungsi aktivasi *tanh* dirumuskan sebagai berikut (Gema & Kartika, 2018):

$$y = f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (2.24)$$



Gambar 8. Fungsi Aktivasi *Tanh*.

2.9 Denormalisasi

Menurut Hidayat, dkk. (2012), denormalisasi dapat memberikan atau mengembalikan data ke bentuk data semula, sehingga didapatkan hasil peramalan dari data *training*. Adapun rumus denormalisasi dalam *range* $[0, 1]$ (Cynthia & Ismanto, 2017):

$$X_i = y_n(X_{\max} - X_{\min}) + X_{\min} \quad (2.25)$$

dengan:

X_i = nilai data normal

y_n = hasil output jaringan

X_{\max} = data dengan nilai maksimum

X_{\min} = data dengan nilai minimum.

2.10 Ketepatan Model

Ketepatan model adalah hal yang penting dalam melakukan peramalan dikarenakan ketepatan model berguna untuk mengevaluasi hasil dari peramalan yang telah dilakukan. Hasil peramalan terkadang muncul beberapa penyimpangan yang bisa disebabkan oleh faktor yang tidak diduga dimana tidak ada metode peramalan yang mampu menghasilkan peramalan yang akurat sehingga dibutuhkan ketepatan model. Banyak cara untuk menghitung ketepatan model, contohnya adalah *Mean Absolute Percentage Error* (MAPE), *Mean Square Error* (MSE) dan *Mean Absolute Error* (MAE). Pada penelitian ini menggunakan MAPE untuk menentukan model terbaik. Sama seperti metode-metode lain, MAPE juga berfungsi untuk mencari nilai kesalahan pada suatu peramalan dan semakin kecil nilai galat pada suatu model maka semakin bagus pula model yang dihasilkan. Berikut untuk rumus MSE yaitu (Fatchurin, Fanani, & Hafiyusholeh, 2010):

$$MSE = \frac{\sum_{t=1}^n (Y_t - \widehat{Y}_t)^2}{n} \quad (2.26)$$

Sedangkan untuk rumus MAPE yaitu (Fatchurin, Fanani, & Hafiyusholeh, 2020):

$$MAPE = \frac{\sum_{t=1}^n \frac{|Y_t - \widehat{Y}_t|}{Y_t}}{n} \times 100\% \quad (2.27)$$

dengan:

Y_t = data aktual periode t

\widehat{Y}_t = data hasil peramalan periode t

n = banyaknya data.

Nilai MAPE yang dihasilkan mempunyai interpretasi sebagai berikut (Fatchurin, Fanani, & Hafiyusholeh, 2020):

1. MAPE < 10% : peramalan sangat baik
2. MAPE 10% – 20% : beramalan baik
3. MAPE 20% – 50% : peramalan cukup
4. MAPE > 50% : peramalan tidak baik.

2.11 Emas

Sejak dahulu emas telah dikenal sebagai alat investasi aman dan alat yang baik untuk melindungi aset kekayaan ketika masa-masa sulit. Hal ini karena emas dapat memperoleh keuntungan dalam jangka panjang dengan harga yang naik dari tahun ke tahun dan nilainya sangat stabil. Investasi emas dapat dilakukan oleh siapa saja, baik dari golongan berpendidikan atau bukan. Terdapat beberapa alasan yang menjadikan emas sebagai alat investasi yang banyak diminati masyarakat, yaitu harga emas yang cenderung stabil, emas merupakan investasi yang mudah dijual kembali tanpa perlu menunggu pembeli dan cocok jika disimpan dalam jangka menengah maupun jangka panjang. Adapun faktor-faktor yang mempengaruhi harga emas yaitu inflasi, krisis finansial, naiknya permintaan emas di pasaran, kurs dolar, harga minyak mengalami kenaikan dan situs politik dunia (Apriyanti, 2012).

III. METODOLOGI PENELITIAN

3.1 Waktu dan Tempat Penelitian

Penelitian ini dilakukan pada semester genap tahun akademik 2021/2022 bertempat di Jurusan Matematika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Lampung.

3.2 Data Penelitian

Data yang digunakan dalam penelitian ini adalah data sekunder yaitu data historis harga harian emas dunia periode 1 Februari 2016 sampai dengan 1 Februari 2022 yang diperoleh dari <https://id.investing.com/> dengan jumlah data sebanyak 2192 data. Satuan data harga harian emas yang digunakan yaitu satu *troy ounce* (31.1 gram) dalam satuan US Dollar. Pada penelitian ini, data dibagi menjadi dua bagian yaitu data *training* untuk pembentukan model dan data *testing* untuk pemilihan model terbaik. Untuk proses pembagian datanya dibagi menjadi 4 bagian. Pertama, 50% data *training* (1096 data) dan 50% data *testing* (1096 data). Kedua, 60% data *training* (1315 data) dan 40 % data *testing* (877 data). Ketiga, 70% data *training* (1534 data) dan 30% data *testing* (658 data). Keempat, 80% data *training* (1754 data) dan 20% data *testing* (438 data).

3.3 Metode Penelitian

Dalam penelitian ini dilakukan analisis menggunakan algoritma *Backpropagation Neural Network*. Langkah-langkah yang dilakukan yaitu:

1. Melakukan analisis deskriptif.
2. Menentukan input jaringan untuk pemodelan *Backpropagation* berdasarkan *lag-lag* yang signifikan pada plot PACF.
3. Melakukan normalisasi data.
4. Melakukan pembagian data yaitu data *training* dan data *testing*.
5. Menentukan fungsi aktivasi.
6. Pembuatan model *Backpropagation Neural Network*
 - a. Melakukan proses *training*
 - b. Melakukan proses *testing*.
7. Melakukan denormalisasi data.
8. Melakukan peramalan data emas pada bulan Juni 2022.

V. KESIMPULAN

Berdasarkan pembahasan mengenai pembentukan model menggunakan algoritma *Backpropagation Neural Network* yang diterapkan untuk meramalkan harga harian emas, maka dapat disimpulkan sebagai berikut:

1. Model dengan menggunakan algoritma *Backpropagation Neural Network* yang diterapkan pada data harga harian emas dunia mulai tanggal 1 Februari 2016 sampai dengan 1 Februari 2022 memiliki variabel input pada data *lag* 1 yang berjumlah 2192 data. Model ini menghasilkan model terbaik dengan hasil MSE (*Mean Squared Error*) terkecil 0.0002452 dan MAPE (*Mean Absolute Percentage Error*) 5.1165471 pada fungsi aktivasi *tanh* di *hidden layer* ke-2 menggunakan pembagian data pelatihan 70% (1534 data) dan data pengujian 30% (658 data).
2. Berdasarkan nilai MAPE pada model yang digunakan termasuk sangat baik karena nilai MAPE yang dihasilkan pada model ini dibawah 10%.
3. Peramalan harga harian emas dunia menggunakan model terbaik menghasilkan harga emas terbesar pada 23 Juni 2022 sebesar Rp873.639,00 dan harga emas terkecil pada 26 Juni 2022 sebesar Rp841.742,00.

DAFTAR PUSTAKA

- Aden. 2020. *Forecasting The Eksponential Smoothing Methods*. Unpam Press, Tangerang Selatan.
- Apriyanti, M. 2012. *Anti Rugi dengan Berinvestasi Emas*. Pustaka Baru Press, Yogyakarta.
- Cynthia, E. P. & Ismanto, E. 2017. Jaringan Syaraf Tiruan Algoritma Backpropagation dalam Memprediksi Ketersediaan Komoditi Pangan Provinsi Riau. *Rabit : Jurnal Teknologi dan Sistem Informasi Univrab*. **2(2)**: 83-98.
- Fatchurin, E., Fanani, A., & Hafiyusholeh, M. 2020. Peramalan Penggunaan Bahan Bakar pada Pembangkit Listrik Tenaga Gas Uap menggunakan Metode Backpropagation Neural Network. *Jurnal Riset dan Aplikasi Matematika*. **4(2)**: 82-92.
- Gema, R. L. & Kartika, D. 2018. Algoritma Propagasi Balik dalam Pencarian Pola Training Terbaik untuk Menentukan Prediksi Produksi Usaha Songket Silungkang dengan menggunakan Matlab. *Komik*. **2(1)**: 58-64.
- Hanifa, T.T., Adiwijaya & Al-Faraby, S. 2017. Analisis Churn Prediction pada Data Pelanggan PT. Telekomunikasi dengan Logistic Regression dan Underbagging. *Jurnal Teknik Informatika Universitas Telkom*. **4(2)**: 3210–3225.
- Hanke, J.E. & Winchern, D.W. 2004. *Business Forecasting*. 8th Edition. Pearson Education Inc., United States of Amerika.

- Hidayat, R. & Suprpto. 2012. Meminimalisasi Nilai Error Peramalan dengan Algoritma Extreme Learning Machine. *Jurnal Optimasi Sistem Industri UNAND*. **11**(1): 187-192.
- Hyndman, R.J. & Athanasopoulos, G. 2021. *Forecasting: Principles and Practice*. 3rd Edition. OTexts, Melbourne.
- Kristanto, A. 2004. *Jaringan Syaraf Tiruan (Konsep Dasar, Algoritma dan Aplikasi)*. Penerbit Gava Media. Yogyakarta.
- Machmudin, A. & Ulama, B. S. S. 2012. Peramalan Temperatur Udara di Kota Surabaya dengan Menggunakan ARIMA dan Artificial Neural Network. *Jurnal Sains dan Seni*. **1**(1): 118-123.
- Nurmila, N., Sugiharto, A. & Sarwoko, E. A. 2010. Algoritma Back Propagation Neural Network untuk Pengenalan Pola Karakter Huruf Jawa. *Jurnal Masyarakat Informatika*. **1**(1): 2086-4930.
- Rinjani, S.N., Hoyyi, A. & Suparti. 2019. Pemodelan Fungsi Transfer dan Backpropagation Neural Network untuk Peramalan Harga Emas. *Jurnal Gaussian*. **8**(4): 474-485.
- Salman, A.G. & Prasetyo, Y.L. 2010. Implementasi Jaringan Syaraf Tiruan Recurrent dengan Metode Pembelajaran Gradient Adaptive Learning Rate untuk Pendugaan Curah Hujan Berdasarkan Peubah ENSO. *ComTech*. **1**(2): 418-429.
- Siang, J.J. 2009. *Jaringan Syaraf Tiruan & Pemrogramannya Menggunakan Matlab*. Yogyakarta.
- Sitepu, A. C., Sigiro, M. 2021. Analisis Fungsi Aktivasi Relu dan Sigmoid Menggunakan Optimizer SGD dengan Representasi MSE pada Model Backpropagation. *Jutisal*. **1**(1): 12-25.

- Utami, A. T. W. & Ulama, B. S. S. 2015. Penerapan Backpropagation untuk Meningkatkan Efektivitas Waktu dan Akurasi pada Data Wall-Following Robot Navigation. *Jurnal Sains dan Seni ITS*. **4**(2): 279-284.
- Warsito, B. 2009. *Kapita Slekta Statistika Neural Network*. Badan Penerbit UNDIP, Semarang.
- Wei, W.W. 2006. *Time Series Analysis Univariate and Multivariate Methods*. 2nd Edition. Pearson Education, Canada.
- Widiastuti, F., Kaswidjanti, W. & Rustamaji, H. C. 2014. Jaringan Saraf Tiruan Backpropagation untuk Aplikasi Pengenalan Tanda Tangan. *Telematika*. **11**(1): 69-76.