

**ANALISIS STRUKTUR TERBAIK *NEURAL NETWORK* DENGAN  
ALGORITMA *BACKPROPAGATION* DALAM MEMPREDIKSI INDEKS  
KANDUNGAN SULFIDA (SO<sub>2</sub>) DI IBU KOTA JAKARTA**

**(Skripsi)**

**Oleh**

**M NAUFAL AMMAR RAFDIONO  
NPM 1857031008**



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

## **ABSTRACT**

### **THE BEST STRUCTURE ANALYSIS OF NEURAL NETWORK WITH BACKPROPAGATION ALGORITHM SULFIDE CONTENT INDEX (SO<sub>2</sub>) IN THE CITY OF JAKARTA**

**By**

**M NAUFAL AMMAR RAFDIONO**

Air is one of the sources of human life. Polluted air causes various major problems, such as: health. This study aims to compare the performance of the most suitable activation function and obtain the best network structure in predicting the sulfide content index (SO<sub>2</sub>) in the capital city of Jakarta. The performance of each activation function is compared using Mean Square Error (MSE), Mean Absolute Percentage Error (MAPE) and model accuracy. So that the best network structure is obtained in the form of 2 input nodes, 3 hidden layers (3 nodes in the hidden layer, 2 nodes in the hidden layer, and 1 node in the using third hidden layer) the 90% training and 10% testing and the TanH activation function. . With an MSE of 0.009990629 and a MAPE 15.87412% or an accuracy of 84.1258%.

**Keywords: Air Pollution Index, Backpropagation, Neural Network, Time Series, Predict**

## ABSTRAK

### ANALISIS STRUKTUR TERBAIK *NEURAL NETWORK* DENGAN ALGORITMA *BACKPROPAGATION* DALAM MEMPREDIKSI INDEKS KANDUNGAN SULFIDA (SO<sub>2</sub>) DI IBU KOTA JAKARTA

Oleh

M NAUFAL AMMAR RAFDIONO

Udara merupakan salah satu sumber kehidupan manusia. Udara yang tercemar mengakibatkan berbagai permasalahan besar, seperti: kesehatan. Penelitian ini bertujuan untuk membandingkan kinerja fungsi aktivasi yang paling sesuai dan memperoleh struktur jaringan terbaik dalam memprediksi indeks kandungan sulfida (SO<sub>2</sub>) di ibu kota Jakarta. Kinerja masing-masing fungsi aktivasi dibandingkan dengan menggunakan *Mean Square Error (MSE)*, *Mean Absolute Percentage Error (MAPE)* dan akurasi model. Sehingga didapatkan struktur jaringan terbaik berupa 2 nodes input, 3 *hidden layer* (3 nodes pada *hidden layer* pertama, 2 nodes pada *hidden layer* kedua, dan 1 nodes pada *hidden layer* ketiga) dengan menggunakan skema 90% *training* dan 10% *testing* serta fungsi aktivasi TanH. Dengan nilai *MSE* sebesar 0.009990629 dan nilai *MAPE* 15.87412% atau akurasi sebesar 84.12588%.

Kata Kunci: **Indeks Standar Pencemaran Udara (ISPU), Backpropagation, Neural Network, Deret Waktu, Prediksi**

**ANALISIS STRUKTUR TERBAIK *NEURAL NETWORK* DENGAN  
ALGORITMA *BACKPROPAGATION* DALAM MEMPREDIKSI INDEKS  
KANDUNGAN SULFIDA (SO<sub>2</sub>) DI IBU KOTA JAKARTA**

**Oleh**

**M NAUFAL AMMAR RAFDIONO**

**Skripsi**

**Sebagai Salah Satu Syarat untuk Mencapai Gelar  
SARJANA MATEMATIKA**

**Pada**

**Jurusan Matematika  
Fakultas Matematika dan Ilmu Pengetahuan Alam  
Universitas Lampung**



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

Judul Skripsi : **ANALISIS STRUKTUR TERBAIK NEURAL NETWORK DENGAN ALGORITMA BACKPROPAGATION DALAM MEMPREDIKSI INDEKS KANDUNGAN SULFIDA (SO<sub>2</sub>) DI IBU KOTA JAKARTA**

Nama Mahasiswa : **M Naufal Ammar Rafdiono**

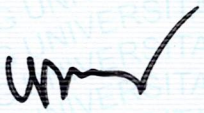
Nomor Induk Mahasiswa : **1857031008**

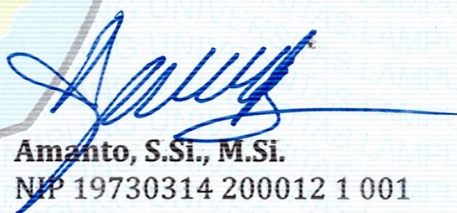
Program Studi : **Matematika**

Fakultas : **Matematika dan Ilmu Pengetahuan Alam**

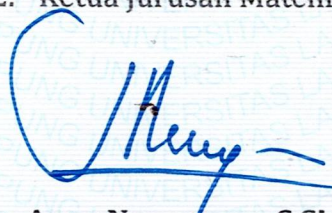


1. Komisi Pembimbing

  
**Ir. Warsono, M.S., Ph.D.**  
NIP 19630216 198703 1 001

  
**Amanto, S.Si., M.Si.**  
NIP 19730314 200012 1 001

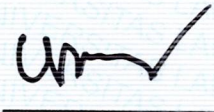
2. Ketua Jurusan Matematika

  
**Dr. Aang Nuryaman, S.Si., M.Si.**  
NIP 19740316 200501 1 001

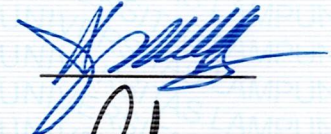
**MENGESAHKAN**

**1. Tim Penguji**

**Ketua : Ir. Warsono, M.S., Ph.D.**



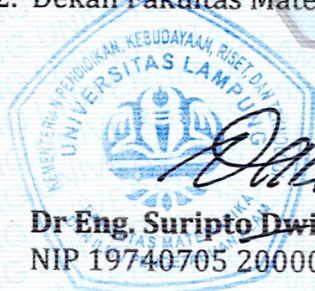
**Sekretaris : Amanto, S.Si., M.Si.**



**Penguji  
Bukan Pembimbing : Dian Kurniasari, S.Si., M.Sc.**



**2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam**



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



**Tanggal Lulus Ujian Skripsi : 22 Juli 2022**

## PERNYATAAN SKRIPSI MAHASISWA

Yang bertanda tangan di bawah ini :

Nama Mahasiswa : M Naufal Ammar Rafdiono

Nomor Pokok Mahasiswa : 1857031008

Jurusan : Matematika

Judul Skripsi : **ANALISIS STRUKTUR TERBAIK  
NEURAL NETWORK DENGAN  
ALGORITMA BACKPROPAGATION  
DALAM MEMPREDIKSI INDEKS  
KANDUNGAN SULFIDA (SO<sub>2</sub>) DI IBU  
KOTA JAKARTA**

Dengan ini menyatakan bahwa penelitian ini adalah hasil pekerjaan saya sendiri dan apabila kemudian hari terbukti bahwa skripsi ini merupakan salinan atau dibuat oleh orang lain, maka saya bersedia menerima sanksi sesuai dengan ketentuan akademik yang berlaku.

Bandar Lampung, 22 Juli 2022

Penulis,



M Naufal Ammar Rafdiono

## **RIWAYAT HIDUP**

Penulis bernama lengkap M Naufal Ammar Rafdiono, dilahirkan pada tanggal 9 Oktober 2000 di Bandar Lampung. Penulis merupakan putra sulung dari Bapak Tunardiyono dan Ibu Rahmi Fadhillah Chan.

Penulis mengawali pendidikan di Taman Kanak-Kanak (TK) Amalia pada tahun 2005-2006. Kemudian menempuh pendidikan Sekolah dasar (SD) di SD Al-Azhar 2 Bandar Lampung pada tahun 2006 – 2012. Kemudian melanjutkan ke Sekolah Menengah Pertama di SMP Negeri 8 Bandar Lampung pada tahun 2012-2015. Selanjutnya belajar pada jenjang Sekolah Menengah Kejuruan di SMK SMTI Bandar Lampung pada tahun 2015-2018.

Pada tahun 2018 penulis terdaftar sebagai mahasiswa Program Studi S1 Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung melalui jalur SMMPTN. Pada Tahun 2020 penulis melaksanakan Kuliah Kerja Nyata (KKN) di Pematang Wangi, Kecamatan Tanjung Senang, Kota Bandar Lampung, sebagai bentuk pengabdian kepada masyarakat. Pada tahun yang sama penulis juga melaksanakan Kuliah Kerja Praktik (KP) di Telkom Witel Lampung, sebagai bentuk aplikasi bidang ilmu di dunia kerja.



## **KATA INSPIRASI**

“Dan bersabarlah kamu. Sesungguhnya janji Allah adalah benar.”

**(Q.S Ar-Rum : 60)**

“Dan hanya kepada Tuhanmulah hendaknya kamu berharap.”

**(Q.S Al-Insyirah : 8)**

Sungguh Atas Kehendak Allah Semua Ini Terwujud, Tiada Kekuatan Kecuali  
Dengan Pertolongan Allah.

**(Q.S Al-Kahfi : 39)**

Seseorang Yang Bersabar Tidak Akan Pernah Kehilangan Kesuksesan Meskipun  
Membutuhkan Waktu Yang Lama Untuk Mencapainya.

**(Ali Bin Abi Thalib)**

Jangan kamu melakukan hal yang kamu sendiri tidak suka jika diperlakukan  
orang lain seperti itu.

**(Ayah)**

*I dream of painting and then I paint my dream*

**(Vincent Van Gogh)**

## **PERSEMBAHAN**

Dengan mengucapkan rasa syukur atas segala puji dan kehadiran Allah SWT. yang telah melimpahkan nikmat serta hidayah-Nya sehingga skripsi ini dapat diselesaikan. Serta tak lupa juga sholawat serta salam selalu tercurahkan kepada junjungan kita Nabi Muhammad SAW. Dengan penuh ketulusan, penulis mempersembahkan karya kecil ini untuk:

### **Ayah dan Bunda**

Tidak ada kata yang dapat aku sampaikan untuk kalian kecuali ucapan terimakasih atas semua yang telah kalian berikan untukku. Cinta, kasih sayang, motivasi, waktu, pengorbanan yang belum bisa aku balas, serta doa dan sujud yang selalu menantikan keberhasilanku dengan sabar dan penuh pengertian. Terimakasih karena selalu mendoakan dan mendukung setiap langkah yang aku pilih. Karena atas doa dan ridho kalian, Allah memudahkan setiap perjalanan hidup ini. Terimalah bukti kecil ini sebagai kado keseriusanku untuk membalas semua pengorbanan, keikhlasan, dan jerih payah yang selama ini kalian lakukan.

### **Dosen Pembimbing dan Pembahas**

Terimakasih kepada dosen pembimbing dan pembahas yang sudah sangat membantu, memberikan motivasi, memberikan arahan serta ilmu yang berharga.

### **Sahabat-sahabatku**

**Almamater Tercinta Universitas Lampung**

## SANWACANA

Puji syukur kehadiran Allah SWT. atas rahmat dan hidayah-Nya, shalawat serta salam selalu tercurahkan kepada baginda besar Nabi Muhammad SAW. sehingga penulis dapat menyelesaikan skripsi yang berjudul “Analisis Struktur Terbaik Neural Network Dengan Algoritma *Backpropagation* Dalam Memprediksi Indeks Kandungan Sulfida (SO<sub>2</sub>) Di Ibu Kota Jakarta”. Dalam penulisan skripsi ini tidak dapat terselesaikan tanpa adanya bimbingan, bantuan dan dukungan dari berbagai pihak. Sehingga, dalam kesempatan ini penulis mengucapkan terimakasih kepada:

1. Bapak Ir. Warsono, M.S., Ph.D, selaku dosen pembimbing I yang senantiasa membimbing, memberi masukan serta saran serta mendukung penulis dalam menyelesaikan skripsi ini.
2. Bapak Amanto, S.Si., M.Si. selaku dosen pembimbing II memberikan bimbingan, pengarahan, serta saran sehingga penulis dapat menyelesaikan skripsi ini.
3. Ibu Dian Kurniasari, S.Si., M.Sc. selaku dosen penguji yang telah memberikan kritik dan saran yang membangun sehingga skripsi ini dapat terselesaikan.
4. Ibu Dra. Dorrah Aziz, M.Si., selaku dosen pembimbing akademik yang telah memberikan bimbingan dan arahan selama masa perkuliahan.
5. Bapak Dr.Aang Nuryaman, S.Si., M.Si., selaku Ketua Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.
6. Bapak Dr. Eng. Suropto Dwi Yuwono, S.Si., M.T. selaku Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.
7. Seluruh dosen, staff, karyawan Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung..
8. Teruntuk kedua orang tuaku tercinta, Ayah Tunardiyono dan Bunda Rahmi Fadhillah Chan terimakasih atas doa, dukungan, pengorbanan, cinta kasih, perhatian, demi kesuksesan penulis semoga dikemudian hari dapat membahagiakan dan menjadi kebanggan kalian

9. Adik-adikku tercinta, M. Almer Arya Putra Rafdiono, dan M. Azka Sauqi Ramadhan Rafdiono, terimakasih telah memberikan dukungan, doa yang telah diberikan demi kesuksesanku.
10. Teruntuk sahabatku sejak awal masuk kuliah Nadia Vivo dan Regita Elza, terimakasih atas doa, semangat, motivasi, canda tawa, susah senang, serta bantuan dalam menyelesaikan skripsi ini. Semoga kita sukses selalu dan persahabatan kita tetap terjaga sampai kapanpun, jangan pernah lupa dengan kenangan dan perjuangan yang telah kita lalui bersama.
11. Teman-teman skripsianku Farrel, Ferzy, Zaenal, Amel, Caca, Rapid, Amelie, Juliana, Pia, Intan, Fatur, dll. terimakasih atas doa, motivasi, dukungan, semangat, yang selalu di berikan kepada penulis. Semoga kalian menjadi orang yang sukses dan bahagia dimanapun kalian berada. *See you on top.*
12. Semua teman sejurusan matematika 2018 dan teman kelas C yang telah membantu serta memberikan semangat kepada penulis yang mana tidak bisa disebutkan satu persatu.
13. Teruntuk sahabatku sejak kecil, Hadi, Rahmat, Dinar, terimakasih atas doa, dukungan, saran, dan motivasi, serta bantuan dalam menyelesaikan skripsi ini.
14. Teruntuk sahabatku sejak SMK, Arsyi, Niko, Imam, Aldila, Rakha, terimakasih atas doa, dukungan, saran, dan motivasi, serta bantuan dalam menyelesaikan skripsi ini.
15. Orang-orang baik yang namanya tidak bisa saya sebutkan satu persatu yang telah menjadi teman terbaik penulis yang selalu memberikan semangat dan menemani penulis dalam keadaan apapun serta telah memberikan pengalaman dan banyak cerita selama masa perkuliahan.
16. Semua pihak yang tidak dapat disebutkan satu persatu yang telah membantu penulis dalam menyelesaikan skripsi ini.
17. *Last but not least, I wanna thank me, for believing in me, for doing all this hard work, for having no days off, for never quitting, for just being me at all time.*

Penulis menyadari bahwa masih banyak kekurangan dalam penulisan skripsi ini. Oleh karena itu, penulis mengharapkan masukan serta saran untuk dijadikan pelajaran kedepannya.

Bandar Lampung, 22 Juli 2022  
Penulis,

M Naufal Ammar Rafdiono

## DAFTAR ISI

|  | Halaman  |
|--|----------|
| <b>I. PENDAHULUAN .....</b>                                      | <b>1</b> |
| 1.1. Latar Belakang dan Masalah .....                            | 1        |
| 1.2. Tujuan Penelitian .....                                     | 3        |
| 1.3. Manfaat Penelitian .....                                    | 3        |
| <br>   |          |
| <b>II. TINJAUAN PUSTAKA .....</b>                                | <b>4</b> |
| 2.1. Prediksi .....  | 4        |
| 2.2. <i>Time Series</i> .....                                    | 4        |
| 2.3. <i>Knowledge Discovery in Database (KDD)</i> .....          | 5        |
| 2.4. <i>Statistical Learning</i> .....                           | 7        |
| 2.4.1. <i>Supervised Learning</i> .....                          | 7        |
| 2.4.2. <i>Unsupervised Learning</i> .....                        | 7        |
| 2.4.3. <i>Semi-supervised Learning</i> .....                     | 8        |
| 2.5. Fungsi Aktivasi .....                                       | 8        |
| 2.5.1. Fungsi Aktivasi Sigmoid .....                             | 9        |
| 2.5.2. Fungsi Aktivasi TanH .....                                | 10       |
| 2.5.3. Fungsi Aktivasi <i>Rectified Linear Unit (ReLU)</i> ..... | 11       |
| 2.6. <i>Artificial Neural Network</i> .....                      | 12       |
| 2.6.1. Konsep Dasar <i>Artificial Neural Network</i> .....       | 13       |
| 2.6.2. Struktur <i>Artificial Neural Network</i> .....           | 14       |
| 2.6.3. <i>Perceptron</i> .....                                   | 15       |
| 2.7. Algoritma <i>Backpropagation</i> .....                      | 16       |
| 2.7.1. <i>Hidden Layer</i> (Lapisan Tersembunyi) .....           | 17       |
| 2.7.2. <i>Epoch, Batch</i> dan Iterasi .....                     | 17       |
| 2.7.3. Propagasi Maju, Mundur dan <i>Update Bobot</i> .....      | 18       |

|             |   |           |
|-------------|---|-----------|
| 2.8.        | Evaluasi Kinerja Algoritma .....  | 22        |
| 2.8.1.      | Validasi Model.....   | 22        |
| 2.9.        | Indeks Standar Pencemaran Udara (ISPU).....   | 23        |
| <b>III.</b> | <b>METODOLOGI PENELITIAN .....</b>  | <b>25</b> |
| 3.1.        | Tempat dan Waktu Penelitian .....   | 25        |
| 3.2.        | Data Penelitian.....  | 25        |
| 3.3.        | Metode Penelitian .....   | 26        |
| <b>IV.</b>  | <b>HASIL DAN PEMBAHASAN .....</b>   | <b>29</b> |
| 4.1.        | Seleksi Data .....  | 29        |
| 4.2.        | <i>Preprocessing</i> Data .....   | 30        |
| 4.2.1.      | Uji Korelasi.....   | 30        |
| 4.2.2.      | Mengubah Data Menjadi Format Deret Waktu .....  | 30        |
| 4.2.3.      | <i>Missing</i> Data .....   | 32        |
| 4.3.        | <i>Scaling</i> Data.....  | 32        |
| 4.4.        | Pembagian Data <i>Training</i> dan <i>Testing</i> .....                                     | 33        |
| 4.5.        | Membangun Model <i>Neural Network</i> Menggunakan Algoritma<br><i>Backpropagation</i> ..... | 34        |
| 4.5.1       | Menentukan <i>Hidden Layer</i> .....  | 34        |
| 4.5.2       | Mengaktifkan Neuron Dengan Fungsi Aktivasi .....  | 35        |
| 4.6         | Pengujian Model.....  | 36        |
| 4.6.1       | Skema 60% Data <i>Training</i> dan 40% Data <i>Testing</i> .....                            | 36        |
| 4.6.2       | Skema 70% Data <i>Training</i> dan 30% Data <i>Testing</i> .....                            | 38        |
| 4.6.3       | Skema 80% Data <i>Training</i> dan 20% Data <i>Testing</i> .....                            | 40        |
| 4.6.4       | Skema 90% Data <i>Training</i> dan 10% Data <i>Testing</i> .....                            | 42        |
| 4.7         | Prediksi Data dan Evaluasi Model .....  | 44        |
| <b>V.</b>   | <b>KESIMPULAN .....</b>   | <b>51</b> |
|             | <b>DAFTAR PUSTAKA .....</b>   | <b>52</b> |
|             | <b>LAMPIRAN</b>   |           |

## DAFTAR TABEL

| Tabel  | Halaman |
|--|---------|
| 1. Data Penelitian .....                                   | 25      |
| 2. Data Awal .....   | 29      |
| 3. Pembagian Data <i>Training</i> dan <i>Testing</i> ..... | 33      |
| 4. Nilai <i>MSE</i> Pada Fungsi Aktivasi .....             | 35      |
| 5. Nilai Akurasi Data .....                                | 44      |
| 6. Data Aktual dan Hasil Prediksi $SO_2$ .....             | 49      |



## DAFTAR GAMBAR

| Gambar  | Halaman |
|---|---------|
| 1. Pola Data <i>Time Series</i> .....   | 5       |
| 2. Fungsi Aktivasi Sigmoid.....   | 10      |
| 3. Fungsi Aktivasi TanH .....   | 11      |
| 4. Fungsi Aktivasi <i>ReLU</i> .....  | 12      |
| 5. Alur <i>Neural Network</i> .....   | 14      |
| 6. <i>Single-Layer Perceptron</i> .....   | 15      |
| 7. <i>Multi-Layer Perceptron</i> .....  | 16      |
| 8. Arsitektur Jaringan <i>Backpropagation</i> .....   | 18      |
| 9. Proses Propagasi Maju ( <i>Forward Pass</i> ) .....  | 19      |
| 10. Proses <i>Backward</i> dan <i>Update Bobot</i> .....  | 20      |
| 11. Nilai Koefisien Korelasi Antar Variabel.....  | 30      |
| 12. <i>Code</i> Mengubah Data Menjadi Format Deret Waktu .....  | 31      |
| 13. Plot Data ISPU .....  | 31      |
| 14. <i>Missing Value</i> Pada Data .....  | 32      |
| 15. <i>Summary Min-Max Normalization</i> .....  | 32      |
| 16. <i>Min-Max Normalization</i> .....  | 33      |
| 17. Grafik Nilai <i>Loss</i> dan Nilai <i>Validation Loss</i> menggunakan Fungsi Aktivasi Sigmoid .....     | 36      |
| 18. Grafik Nilai <i>Loss</i> dan Nilai <i>Validation Loss</i> menggunakan Fungsi Aktivasi TanH .....        | 37      |
| 19. Grafik Nilai <i>Loss</i> dan Nilai <i>Validation Loss</i> menggunakan Fungsi Aktivasi <i>ReLU</i> ..... | 37      |
| 20. Grafik Nilai <i>Loss</i> dan Nilai <i>Validation Loss</i> menggunakan Fungsi Aktivasi Sigmoid .....     | 38      |
| 21. Grafik Nilai <i>Loss</i> dan Nilai <i>Validation Loss</i> menggunakan Fungsi Aktivasi TanH .....        | 39      |

|   |    |
|---|----|
| 22. Grafik Nilai <i>Loss</i> dan Nilai <i>Validation Loss</i> menggunakan Fungsi Aktivasi <i>ReLU</i> .....                               | 39 |
| 23. Grafik Nilai <i>Loss</i> dan Nilai <i>Validation Loss</i> menggunakan Fungsi Aktivasi <i>Sigmoid</i> .....                            | 40 |
| 24. Grafik Nilai <i>Loss</i> dan Nilai <i>Validation Loss</i> menggunakan Fungsi Aktivasi <i>TanH</i> .....                               | 41 |
| 25. Grafik Nilai <i>Loss</i> dan Nilai <i>Validation Loss</i> menggunakan Fungsi Aktivasi <i>ReLU</i> .....                               | 41 |
| 26. Grafik Nilai <i>Loss</i> dan Nilai <i>Validation Loss</i> menggunakan Fungsi Aktivasi <i>Sigmoid</i> .....                            | 42 |
| 27. Grafik Nilai <i>Loss</i> dan Nilai <i>Validation Loss</i> menggunakan Fungsi Aktivasi <i>TanH</i> .....                               | 43 |
| 28. Grafik Nilai <i>Loss</i> dan Nilai <i>Validation Loss</i> menggunakan Fungsi Aktivasi <i>ReLU</i> .....                               | 43 |
| 29. Plot <i>Actual VS Prediction</i> Terbaik (Skema 60% <i>training</i> dan 40% <i>testing</i> dengan Fungsi Aktivasi <i>TanH</i> ) ..... | 45 |
| 30. Plot <i>Actual VS Prediction</i> Terbaik (Skema 70% <i>training</i> dan 30% <i>testing</i> dengan Fungsi Aktivasi <i>TanH</i> ) ..... | 46 |
| 31. Plot <i>Actual VS Prediction</i> Terbaik (Skema 80% <i>training</i> dan 20% <i>testing</i> dengan Fungsi Aktivasi <i>TanH</i> ) ..... | 47 |
| 32. Plot <i>Actual VS Prediction</i> Terbaik (Skema 90% <i>training</i> dan 10% <i>testing</i> dengan Fungsi Aktivasi <i>TanH</i> ) ..... | 48 |
| 33. Keluaran Pada <i>Output Layer</i> .....   | 49 |
| 34. <i>Code</i> Denormalisasi .....   | 49 |
| 35. Plot <i>Actual VS Prediction</i> Terbaik .....  | 50 |

## I. PENDAHULUAN

### 1.1 Latar Belakang dan Masalah

Kehidupan kota besar yang penuh dengan kendaraan memiliki permasalahan yang cukup kompleks. Karena udara yang menyelimuti bumi mempunyai kapasitas yang sangat vital bagi kehidupan di dunia. Udara yang dihirup manusia mengandung komponen oksigen ( $O_2$ ) untuk bernafas, karbon dioksida ( $CO_2$ ) untuk fotosintesis oleh klorofil di daun dan ozon ( $O_3$ ) untuk menahan sinar ultra violet (Wardhana, 2001).

Pencemaran udara menyebabkan perubahan komposisi udara dari keadaan semula. Kehadiran bahan atau zat asing di udara dalam jumlah tertentu dan dalam waktu lama dapat mengganggu kehidupan manusia. Ketika situasi seperti itu terjadi, dapat dikatakan bahwa udara telah tercemar (Wardhana, 2001).

Sulfur dioksida ( $SO_2$ ) merupakan salah satu pencemar udara dengan jumlah tertinggi hingga 18% di udara (Sastrawijaya, 2009). Muziansyah, dkk. (2015) menyatakan bahwa efek negatif dari polutan tersebut pada manusia adalah iritasi saluran pernapasan dan penurunan fungsi paru-paru dengan gejala batuk, sesak napas, dan peningkatan gejala asma.

Prediksi merupakan hal penting yang digunakan untuk menemukan peristiwa masa depan dengan mengenali pola peristiwa masa lalu. Penerapan prediksi di kehidupan sehari-hari orang lebih siap untuk apa pun, baik dalam hal kehidupan manusia maupun kekayaan. Prediksi adalah salah satu pemanfaatan teknologi *data mining*. Menurut Daryl Pregibon dalam Prasetyo (2014), *data mining* merupakan gabungan

dari penelitian statistik, kecerdasan buatan dan database yang masih dalam pengembangan.

Seiring dengan perkembangan teknologi, berbagai model prediksi juga berkembang cukup pesat. *Artificial neural network* dapat digunakan untuk melakukan prediksi karena memiliki kemampuan untuk menemukan dan mengenali pola data historis. Penerapan jaringan saraf tiruan dalam peramalan dan prediksi meningkat di hampir setiap penelitian ilmiah dalam satu tahun terakhir. Hal ini dikarenakan bahwa *artificial neural network* memiliki akurasi yang lebih baik dibandingkan model statistik dan matematika klasik. Selain itu *artificial neural network* fleksibel, baik dalam desain maupun dalam penggunaannya (Sharma dan Nijhawan, 2015).

Berdasarkan penelitian Rahman, dkk. (2013) menyatakan bahwa peramalan indeks polusi udara menggunakan metode *autoregressive integrated moving average (ARIMA)*, *fuzzy time series (FTS)* dan *artificial neural network (ANN)* yaitu membandingkan nilai *root mean square error (RMSE)* diperoleh hasil bahwa *ANN* memberikan *RMSE* terkecil untuk indeks polusi udara dibandingkan dengan *FTS* dan *ARIMA*. Penelitian mengenai *artificial neural network* untuk meramalkan indeks kualitas udara juga dilakukan oleh Maleki, dkk. (2019) dan diperoleh nilai *RMSE* sebesar 59.9. Penelitian lain yang dilakukan oleh Ganesh, dkk. (2018) terhadap indeks kualitas udara dengan metode *artificial neural network* memperoleh nilai *MAE* sebesar 7.33, nilai *MAPE* sebesar 4.05% dan nilai *RMSE* sebesar 9.69.

Oleh karena itu, penelitian ini akan menerapkan metode *neural network* dengan algoritma *backpropagation* untuk memprediksi komponen zat indeks standar pencemaran udara (ISPU) pada indeks kandungan sulfida ( $\text{SO}_2$ ). Metode ini menggunakan 3 fungsi aktivasi didalam menentukan nilai *MSE*. Hasil analisis struktur terbaik menggunakan data data indeks standar pencemaran udara per hari. Metode *neural network* diterapkan karena memiliki tingkat ketelitian yang tinggi dan kelebihan pada prediksi hubungan non linier (Capparuccia, dkk., 2007).

## **1.2. Tujuan Penelitian**

Adapun tujuan dari penelitian ini diantaranya:

1. Membandingkan fungsi aktivasi yang paling sesuai untuk indeks kandungan sulfida ( $\text{SO}_2$ ) pada data indeks standar pencemaran udara.
2. Menemukan struktur jaringan terbaik guna memprediksi indeks kandungan sulfida ( $\text{SO}_2$ ) pada data indeks standar pencemaran udara.
3. Melakukan prediksi indeks kandungan sulfida ( $\text{SO}_2$ ) pada indeks standar pencemaran udara.

## **1.3. Manfaat Penelitian**

Adapun manfaat penelitian ini yaitu:

1. Sebagai rujukan pengembangan ilmu matematika dalam memprediksi indeks kandungan sulfida ( $\text{SO}_2$ ) pada indeks standar pencemaran udara serta dapat menjadi bahan pertimbangan dan informasi tambahan bagi peneliti yang akan melakukan penelitian tentang indeks standar pencemaran udara terhadap komponen-komponen zatnya.
2. Sebagai bahan pertimbangan bagi pihak-pihak lain yang memiliki kepentingan terhadap indeks standar pencemaran udara.

## II. TINJAUAN PUSTAKA

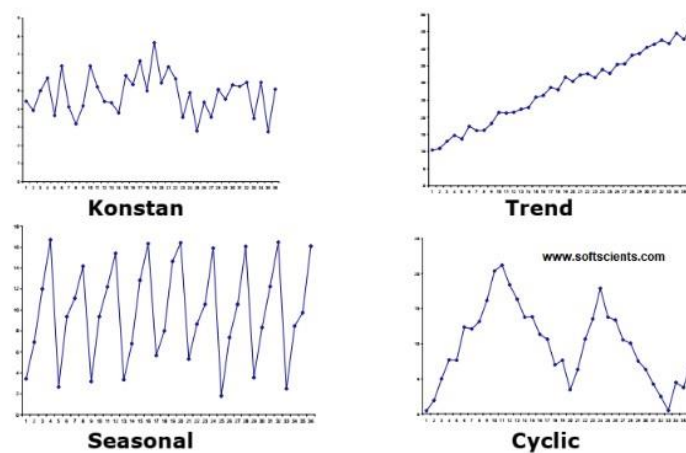
### 2.1 Prediksi

Prediksi ialah mencari nilai suatu variabel berdasarkan nilai variabel tersebut (Makridakis, dkk., 1999). Prediksi dapat digunakan untuk memperkirakan satu atau lebih peristiwa di waktu mendatang berdasarkan pola data yang dianalisis secara ilmiah di masa lalu. Model prediksi berkaitan dengan membangun model yang dapat melakukan pemetaan dari setiap himpunan variabel ke setiap targetnya, kemudian menggunakan model tersebut untuk memberikan nilai target pada himpunan baru yang didapat. Setiap model peramalan pasti menghasilkan kesalahan. Semakin kecil tingkat kesalahan dalam hasil, maka hasil peramalan akan semakin mendekati tepat. Setelah semua tahap dilakukan dan diperoleh model, maka model ini selanjutnya dapat digunakan untuk melakukan peramalan untuk data periode selanjutnya. Ada dua model prediksi yaitu klasifikasi dan regresi. Klasifikasi digunakan untuk variabel target diskrit dan regresi digunakan untuk variabel target kontinu (Prasetyo, 2014).

### 2.2 *Time Series*

*Time series* atau runtun waktu adalah sekumpulan pengamatan yang diurutkan dalam waktu (Hanke dan Wichern, 2005). Metode *time series* merupakan metode peramalan yang menggunakan analisa pola hubungan antara variabel yang akan diperkirakan dengan variabel waktu. Saat meramalkan data runtun waktu harus memperhatikan tipe atau pola datanya. Secara umum ada empat jenis pola data *time series*, yaitu horizontal, trend, musiman, dan siklis (Hanke dan Wichren, 2005).

Pola horizontal adalah kejadian tak terduga dan bersifat *random*, tetapi keberadaannya dapat memengaruhi fluktuasi data *time series*. Pola *trend* adalah kecenderungan arah data yang jangka panjang, dapat berupa penurunan maupun kenaikan. Pola musiman adalah fluktuasi dari data yang terjadi dalam kurun waktu satu tahun, seperti triwulan, kuartalan, bulanan, mingguan, ataupun harian. Sedangkan pola siklis adalah fluktuasi dari data untuk waktu satu tahun lebih.



Gambar 1. Pola Data *Time Series*

(Sumber: <https://softscients.com/2021/11/28/uji-stasioneritas-data/>)

### 2.3 *Knowledge Discovery in Database (KDD)*

*Knowledge Discovery in Database* dan *data mining* sering kali digunakan secara bergantian untuk menjelaskan proses penggalian atau penarikan informasi dalam suatu data tertentu, padahal kedua istilah ini merupakan dua hal yang berbeda secara konsep. Proses *KDD* secara garis besar dapat dijelaskan sebagai berikut (Fayyad, 1996):

#### 1. Seleksi Data

Dari sekumpulan data operasional perlu dilakukan pemilihan (seleksi) sebelum dimulainya tahap penggalian informasi dalam *KDD*.

## 2. *Preprocessing Data*

*Preprocessing data* merupakan proses membersihkan data dengan beberapa teknik, seperti membuang duplikasi data, memeriksa data yang inkonsisten, dan memperbaiki kesalahan pada data, seperti kesalahan cetak (tipografi).

## 3. Transformasi Data

Transformasi data digunakan untuk mengubah suatu skala pengukuran dari bentuk asli ke dalam bentuk lain sehingga data tersebut dapat digunakan untuk analisis dan asumsi-asumsi tertentu. Ada beberapa cara untuk mengubah *dataset* yaitu dengan *min-max normalization*, *z-score normalization* dan *decimal scaling*. Proses transformasi *min-max normalization* akan menghasilkan nilai dengan rentang (0,1).

$$x^* = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (2.1)$$

dengan:

$x^*$  : nilai x baru

$x_{min}$  : nilai minimum dari x

$x_{max}$  : nilai maksimum dari x

$$Z = \frac{x - \mu}{\sigma} \quad (2.2)$$

dengan:

$x$  : nilai yang diamati

$\mu$  : rata-rata

$\sigma$  : standar deviasi

$$v^* = \frac{v}{10^j} \quad (2.3)$$

Dengan  $j$  merupakan bilangan bulat terkecil sehingga nilai maksimum  $|v^*| < 1$ .

## 4. *Data mining*

*Data mining* adalah proses mencari pola atau informasi yang menarik pada data terpilih dengan menggunakan teknik atau metode tertentu. Dalam *data mining* terdapat teknik, metode, atau algoritma yang sangat bervariasi. Penentuan metode atau algoritma yang sesuai sangat bergantung pada tujuan dan proses *KDD* secara keseluruhan.

## 5. Interpretasi/Evaluasi

Dari proses yang dihasilkan, pola informasi perlu ditampilkan dalam bentuk yang mudah dimengerti oleh pihak yang berkepentingan.



## 2.4 *Statistical Learning*

*Statistical learning* sebagai alat pembelajaran diklasifikasikan menjadi 3 bagian yaitu *supervised learning*, *semi supervised learning* dan *unsupervised learning* (James, dkk., 2017).

### 2.4.1 *Supervised Learning*

Metode ini setiap pola yang diberikan ke dalam *artificial neural network* memiliki *output* yang telah diketahui. Dalam pembelajaran terawasi, terdapat sejumlah data yang digunakan untuk melatih jaringan hingga tercapai nilai bobot yang diinginkan. Data ini berfungsi sebagai guru atau sebagai representasi dari sekumpulan sampel data yang digunakan untuk melatih jaringan hingga tercapai nilai yang diinginkan. Pada setiap waktu pelatihan, suatu unit input diberikan ke jaringan dan kemudian akan diproses oleh jaringan sehingga didapatkan sebuah *output*. Nilai selisih antara pola *output* aktual (*output* yang dihasilkan) dengan pola *output* yang diharapkan (*output target*) yang disebut *error* kemudian digunakan untuk mengoreksi bobot *artificial neural network*, sehingga *artificial neural network* mampu menghasilkan *output* sedekat mungkin dengan pola target yang telah diketahui oleh *artificial neural network*, contoh algoritma *neural network* yang menggunakan metode ini adalah: *Hebbian*, *Perceptron*, *ADALINE*, *Backpropagation* (Fausett, 1994).

### 2.4.2 *Unsupervised Learning*

Metode ini tidak diperlukan target *output*. Pada metode ini tidak memungkinkan untuk ditentukan hasil seperti apakah yang diharapkan selama proses pembelajaran. Nilai bobot disusun dalam suatu *range* tertentu tergantung pada nilai *input* yang diberikan selama proses pembelajaran. Tujuan pembelajaran ini ialah untuk

mengelompokkan unit-unit yang hampir sama pada suatu wilayah tertentu. Klasifikasi pola biasanya sangat cocok pada pembelajaran ini. Contoh algoritma *artificial neural network* yang menggunakan metode ini adalah: *Competitive, Kohonen* dan *LVQ (Learning Vector Quantization)* (Fausett, 1994).

### **2.4.3 Semi-supervised Learning**

*Semi-supervised learning* dapat dikatakan sebagai kombinasi antara *supervised* dan *unsupervised learning*, dimana pada *semi-supervised learning* dapat menggunakan data berlabel maupun tanpa label. *Semi-supervised learning* akan sangat berguna digunakan ketika sulit untuk mendapatkan fitur yang relevan dari data dan ketika data yang dimiliki berjumlah besar. *Semi-supervised learning* sangat cocok untuk gambar kesehatan, di mana sejumlah kecil data pelatihan dapat menyebabkan peningkatan akurasi yang signifikan. Misalnya, ahli radiologi dapat melabeli subset kecil *CT scan* untuk tumor atau penyakit sehingga mesin dapat memprediksi pasien mana yang mungkin membutuhkan lebih banyak perhatian medis dengan lebih akurat. Sedangkan, proses evaluasi dalam *machine learning* dibagi menjadi 2 bagian: proses *training* dan *testing*. Dalam proses *training* algoritma *machine learning* akan merubah parameter pada dirinya agar sesuai dengan data yang diberikan sehingga *machine learning* dapat menghasilkan suatu informasi dari data yang telah diberikan. Kemudian menguji performa algoritma menggunakan *testing set* (data *testing*) untuk mengevaluasi performa dari data tersebut (James, dkk., 2017).

## **2.5 Fungsi Aktivasi**

Fungsi aktivasi adalah fungsi matematis yang digunakan untuk memperoleh nilai *output* berdasarkan nilai *input*, sesuai dengan namanya fungsi ini dapat mengaktifkan neuron. Faktor penentu di sini ialah bahwa informasi diproses

melalui fungsi aktivasi. Setiap fungsi aktivasi meniru neuron pada otak karena bergantung pada kekuatan sinyal *input*. Hasil dari proses ini kemudian ditimbang dan didistribusikan ke lapisan neuron berikutnya. Pada dasarnya, neuron saling mengaktifkan melalui jumlah tertimbang. Ini memastikan bahwa kekuatan koneksi antara dua neuron berukuran sesuai dengan berat informasi yang diproses (Lewis, 2017).

Fungsi aktivasi yang digunakan dalam algoritma backpropagation harus memiliki beberapa sifat penting: kontinu, terdifferensial dan monoton (Fausett, 1994). Jaringan saraf tiruan memiliki beberapa fungsi aktivasi seperti fungsi sigmoid, fungsi TanH, fungsi linear, fungsi biner, fungsi *hard limit*, fungsi *ReLU*, fungsi *leaky ReLU*, dan sebagainya. Fungsi aktivasi yang akan digunakan pada kasus ini yaitu fungsi aktivasi sigmoid, TanH dan *ReLU*.

### 2.5.1 Fungsi Aktivasi Sigmoid

Fungsi sigmoid atau fungsi logistik adalah pilihan yang sering digunakan. Fungsi sigmoid mengambil angka dalam rentang 0 hingga 1. Secara khusus, bilangan negatif besar menjadi 0 dan bilangan positif besar menjadi 1 (Lewis, 2017).

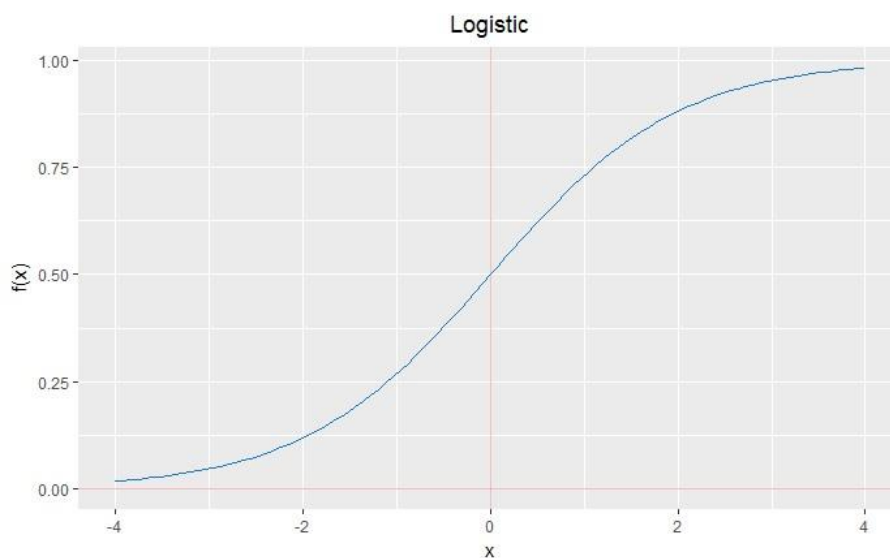
$$\text{Sigmoid}(x) = \frac{1}{1+e^{-x}} \quad (2.4)$$

dengan menggunakan aturan turunan parsial

$$\begin{aligned} f'(x) &= \frac{u(x)}{v(x)} = \frac{u'(x)v(x) - u(x)v'(x)}{[v(x)]^2} \\ \frac{d}{dx} S(x) &= \frac{d}{dx} \frac{1}{1+e^{-x}} \\ &= \frac{e^{-x}}{(1+e^{-x})^2} \\ &= \frac{1-1+e^{-x}}{(1+e^{-x})^2} \\ &= \frac{1+e^{-x}}{(1+e^{-x})^2} - \frac{1}{(1+e^{-x})^2} \end{aligned}$$

$$\begin{aligned}
&= \frac{1}{(1 + e^{-x})} - \frac{1}{(1 + e^{-x})^2} \\
&= \frac{1}{(1 + e^{-x})} - \left(1 - \frac{1}{(1 + e^{-x})}\right) \\
&= S(x)(1 - S(x)) \tag{2.5}
\end{aligned}$$

Fungsi sigmoid akan membentuk huruf “S” jika digambarkan menjadi sebuah grafik.



Gambar 2. Fungsi Aktivasi Sigmoid  
(Sumber: <https://rpubs.com/shailesh/activation-functions>)

### 2.5.2 Fungsi Aktivasi TanH

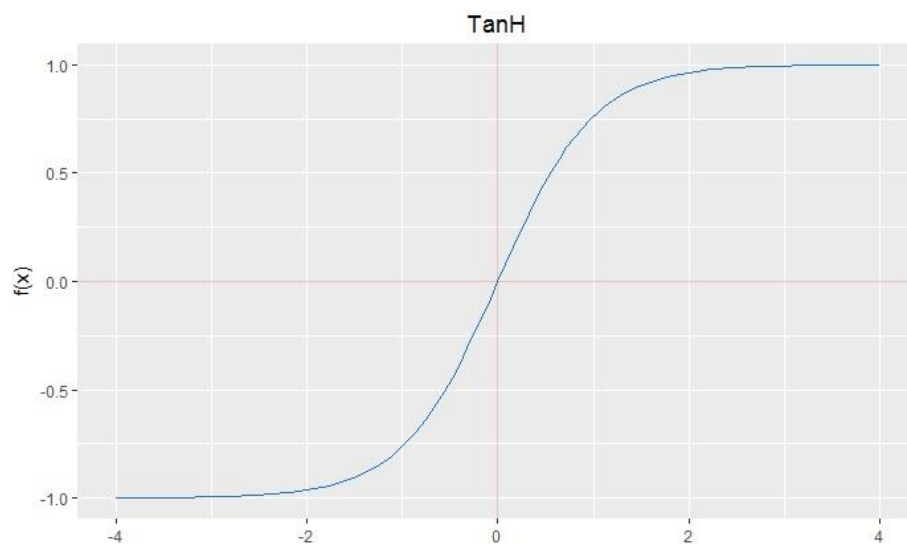
Fungsi aktivasi alternatif untuk sigmoid adalah fungsi aktivasi TanH. Seperti fungsi sigmoid, fungsi TanH berbentuk "S", tetapi fungsi ini menghasilkan nilai antara -1 dan 1. Oleh karena itu, fungsi TanH mirip dengan fungsi sigmoid, tetapi rentang nilai yang luas membuatnya efektif untuk pemodelan *nonlinier* yang kompleks (Lewis, 2017). Bentuk dari fungsi TanH adalah:

$$\text{TanH}(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \tag{2.6}$$

$$\text{TanH}(x) = \frac{\sinh(x)}{\cosh(x)} \quad (2.7)$$

dengan menggunakan fungsi turunan parsial

$$\begin{aligned} f'(x) &= \frac{u(x)}{v(x)} = \frac{u'(x)v(x) - u(x)v'(x)}{[v(x)]^2} \\ \frac{d}{dx} \text{TanH}(x) &= \frac{d \sinh(x)}{dx \cosh(x)} \\ &= \frac{\cosh^2(x) - \sinh^2(x)}{\cosh^2(x)} \\ &= \frac{\cosh^2(x)}{\cosh^2(x)} - \frac{\sinh^2(x)}{\cosh^2(x)} \\ &= 1 - \text{TanH}(x) \end{aligned} \quad (2.8)$$



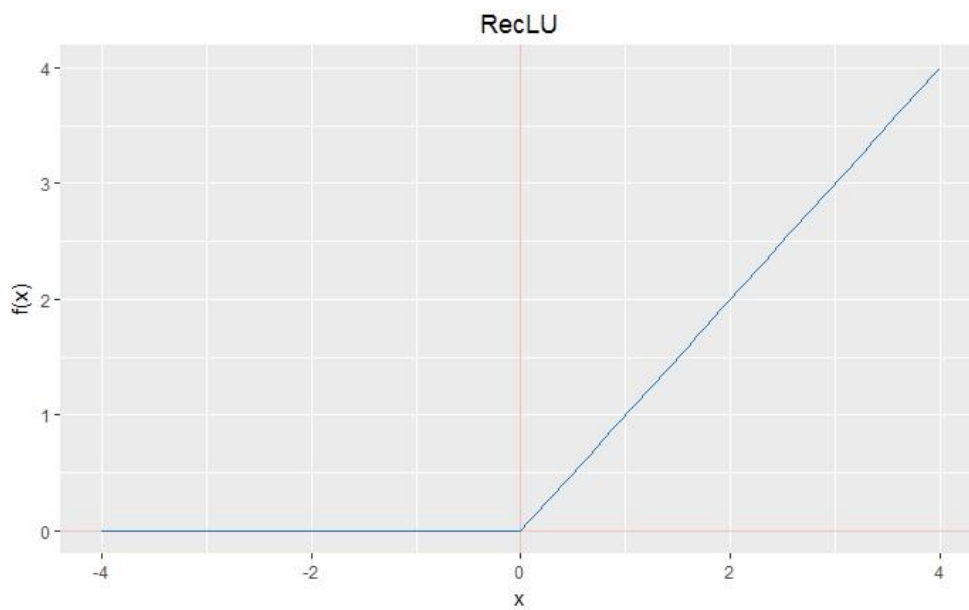
Gambar 3. Fungsi Aktivasi TanH  
(Sumber: <https://rpubs.com/shailesh/activation-functions>)

### 2.5.3 Fungsi Aktivasi *Rectified Linear Unit (ReLU)*

*ReLU (Rectified Linear Unit)* adalah fungsi aktivasi yang digunakan untuk menormalkan nilai yang dihasilkan oleh sebuah *layer*.

$$f(x) = \max(0, x) \quad (2.9)$$

Dari persamaan (2.9), aktivasi *ReLU* diketahui akan menghasilkan nilai  $f(x) = 0$  apabila  $x$  bernilai 0 atau kurang dari 0. Oleh karena itu, setiap entri negatif selalu dipetakan ke 0. Proses pelatihan konvergensi dengan fungsi *ReLU* hingga 6 kali lebih cepat dari fungsi TanH (Krizhevsky, dkk., 2012).



Gambar 4. Fungsi Aktivasi ReLU  
(Sumber: <https://rpubs.com/shailesh/activation-functions>)

## 2.6 *Artificial Neural Network*

Prinsip *artificial neural network* (atau jaringan saraf tiruan) adalah program komputer yang didasarkan pada kemampuan jaringan biologis otak manusia. Secara fungsional, *artificial neural network* dibuat untuk merancang sebuah komputer yang dapat melakukan proses pembelajaran pada contoh kejadian. Dalam hal struktur rancangan, jaringan saraf tiruan dirancang untuk perangkat komputasi yang tampaknya dapat melakukan sesuatu yang mirip dengan jaringan biologis otak manusia. Beberapa istilah yang umum digunakan dalam *artificial neural network* adalah sebagai berikut (Hermawan, 2006).

- a. Bobot adalah nilai matematis dari koneksi antar neuron.

- b. Fungsi aktivasi adalah fungsi untuk menentukan nilai keluaran.
- c. Fungsi aktivasi sederhana digunakan untuk mengalikan *input* dengan bobotnya dan kemudian menjumlahkannya (disebut penjumlahan sigma) berbentuk linier atau tidak linier dan sigmoid.
- d. *Input* adalah nilai *input* yang diolah menjadi nilai *output*.
- e. *Output* adalah solusi dari nilai *input*.
- f. Lapisan tersembunyi (*hidden layer*) adalah lapisan yang tidak berinteraksi langsung dengan dunia luar. Lapisan ini memperluas kemampuan jaringan saraf tiruan untuk mengatasi masalah yang kompleks.
- g. Neuron atau node atau unit adalah sel syaraf tiruan yang merupakan elemen pengolahan *artificial neural network*. Setiap neuron menerima *input*, memproses *input*, dan kemudian mengirimkan hasilnya dalam bentuk *output*.

### 2.6.1 Konsep Dasar *Artificial Neural Network*

*Artificial neural network* (atau jaringan syaraf tiruan) secara umum memiliki dua lapisan, yaitu lapisan *input* dan lapisan *output*. Namun dalam perkembangannya, beberapa *artificial neural network* memiliki lapisan lagi antara *input layer* dan *output layer*, lapisan ini disebut *hidden layer*. Komponen-komponen *artificial neural network* dijelaskan di bawah ini (Kusumadewi, 2003).

#### 1. *Input Layer*

*Input Layer* berisi node-node. Setiap node menyimpan nilai *input* yang tidak berubah selama fase pelatihan dan hanya dapat diubah jika nilai *input* baru dimasukkan. Node pada lapisan ini bergantung pada jumlah *input* dari satu lapisan.

#### 2. *Hidden Layer*

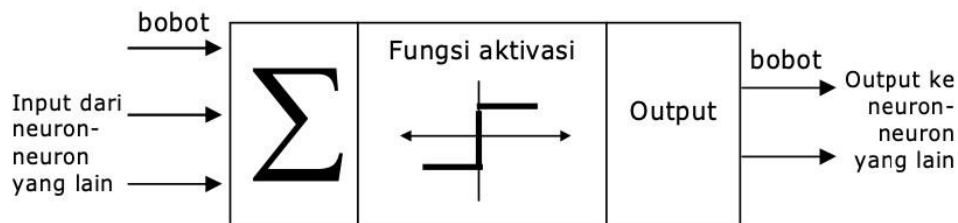
Lapisan ini disebut *hidden layer* karena tersembunyi. Namun, semua proses dalam fase pelatihan dan penerapan berjalan pada lapisan ini. Jumlah lapisan ini tergantung pada arsitektur yang dirancang, tetapi biasanya terdiri dari satu *hidden layer*.

### 3. *Output Layer*

*Output layer* digunakan untuk menampilkan hasil perhitungan sistem oleh fungsi aktivasi pada *hidden layer* berdasarkan *input* yang diterima.

#### 2.6.2 Struktur *Artificial Neural Network*

Kusumadewi (2003) menyatakan bahwa ada berbagai jenis *artificial neural network*, tetapi hampir semua jenis memiliki komponen yang sama. Seperti otak manusia, jaringan saraf tiruan memiliki neuron. Ketika otak manusia memiliki dendrit yang bertindak sebagai penerima rangsangan, terdapat lapisan input di jaringan saraf. Informasi yang diterima dari lapisan input melewati neuron dengan bobot tertentu.



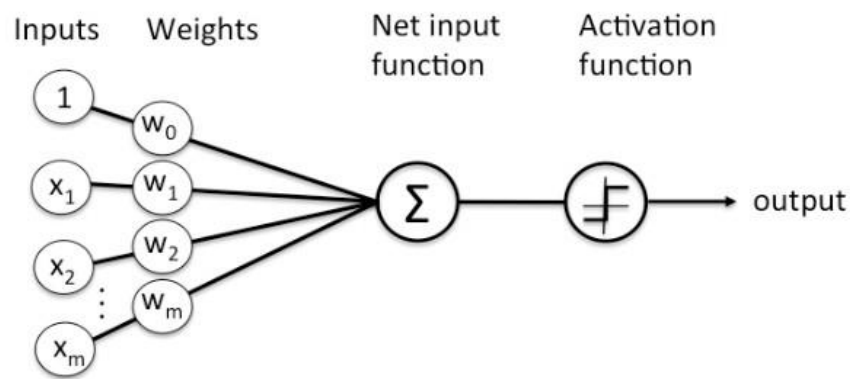
Gambar 5. Alur *Neural Network*  
(Sumber: Kusumadewi, 2003)

Gambar 5 menjelaskan dapat terlihat bahwa *input* ini sedang diproses oleh fungsi perambatan yang menjumlahkan nilai semua bobot *input* yang datang. Selanjutnya, hasil penjumlahan ini dibandingkan dengan nilai ambang batas (*threshold*) tertentu oleh fungsi aktivasi masing-masing neuron. Ketika *input* melebihi ambang batas tertentu, neuron menjadi aktif. Tetapi jika tidak, neuron tersebut tidak akan aktif. Ketika neuron diaktifkan, maka neuron tersebut mengirimkan *output* ke semua neuron yang dihubungkan dengannya melalui bobot *output*nya.



### 2.6.3 Perceptron

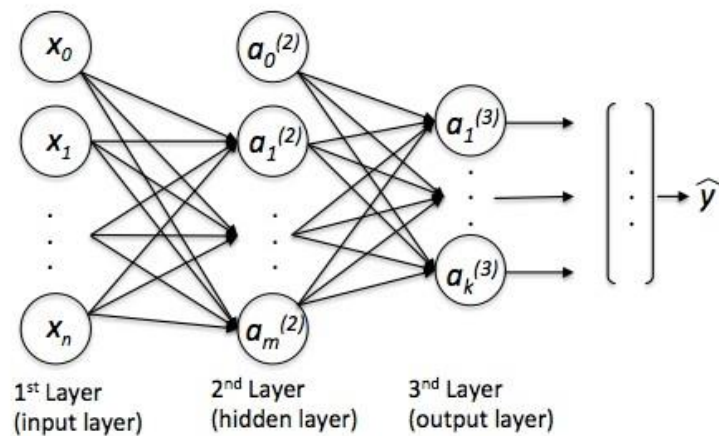
*Perceptron* adalah kombinasi linier yang memetakan *output* ke salah satu dari dua nilai diskrit (Shynk, 1990). Hubungan antara bobot dan ambang batas (*threshold*) dapat diperbaiki atau disesuaikan dengan menggunakan berbagai algoritma pembelajaran. Ada dua jenis *perceptrons* dalam *neural network*: *single-layer perceptrons* dan *multi-layer perceptrons*. *Single-layer perceptron* adalah *artificial neural network* yang hanya terdiri dari *input layer* dan *output layer*.



Gambar 6. *Single-Layer Preceptron*

(Sumber: [https://sebastianraschka.com/Articles/2015\\_singlelayer\\_neurons.html](https://sebastianraschka.com/Articles/2015_singlelayer_neurons.html))

*Single-layer preceptron*, sinyal *input*  $x_m$  diskalakan oleh satu set bobot yang disesuaikan  $w_m$  untuk menghasilkan *output*  $y$ . *Multi-layer preceptron* adalah varian lain dari *perceptron* dengan satu atau lebih *hidden layer* antara *input layer* dan *output layer* (Ramchoun, dkk., 2016).



Gambar 7. *Multi-Layer Perceptron*

(Sumber: [https://sebastianraschka.com/Articles/2015\\_singlelayer\\_neurons.html](https://sebastianraschka.com/Articles/2015_singlelayer_neurons.html))

## 2.7 Algoritma *Backpropagation*

Algoritma *backpropagation* adalah salah satu algoritma *multilayer perceptron*. Pelatihan *backpropagation* terdiri dari 3 fase: fase maju, fase mundur, dan fase koreksi bobot. Algoritma ini merupakan salah satu metode *supervised learning*.

*Backpropagation* hanyalah salah satu implementasi dari metode *gradient descent* (Kulkarni, 2011). Algoritma ini dapat digunakan jika diketahui turunan dari *error* yang berkaitan dengan bobot. Jika dilambangkan  $E_m$  adalah *error* pada *training*  $m$ , *gradient descent* menunjukkan bahwa bobot  $W_{ij}$  diupdate dengan jumlah berikut (dimana  $\eta$  adalah konstanta).

$$\Delta w_{ij} = -\eta \frac{\partial E_m}{\partial w_{ij}} \quad (2.10)$$

dengan:

$\Delta w_{ij}$  : perubahan bobot sambungan dari unit  $i$  ke unit  $j$

$\eta$  : *learning rate*

$\partial E_m$  : turunan dari *error* pada *training*  $m$

$\partial w_{ij}$  : turunan dari bobot

### 2.7.1 *Hidden Layer (Lapisan Tersembunyi)*

Heaton (2017), jarang terjadi dalam *deep learning*, kasus yang menggunakan lebih dari 2 *hidden layer*. 2 atau kurang dari 2 sudah cukup untuk kumpulan data sederhana. Selain menentukan jumlah *hidden layer*, juga perlu menentukan jumlah *nodes* atau neuron yang akan digunakan pada *hidden layer* untuk menentukan arsitektur jaringan secara keseluruhan. Jika terlalu sedikit neuron yang digunakan pada *hidden layer*, itu mengarah pada apa yang disebut *underfitting*. *Underfitting* terjadi ketika ada terlalu sedikit neuron di *hidden layer* untuk mengenali sinyal dengan benar dalam kumpulan data yang kompleks.

Penggunaan sejumlah besar neuron juga menyebabkan masalah seperti *overfitting*. *Overfitting* terjadi ketika kapasitas pemrosesan informasi jaringan saraf sangat tinggi dan jumlah terbatas informasi yang terkandung dalam set *training* tidak cukup untuk melatih semua neuron pada *hidden layer*. Berdasarkan aturan Jeff Heaton *rule* (Heaton, 2017):

1. Jumlah neuron pada *hidden layer* harus berada di antara ukuran *input layer* dan ukuran *output layer*.
2. Jumlah neuron *hidden layer* harus bernilai  $\frac{2}{3}$  dari jumlah *input layer* ditambah *output layer*  $\left(\frac{2}{3}(\text{input} + \text{output})\right)$ .
3. Jumlah neuron *hidden layer* harus kurang dari dua kali jumlah *input layer*.

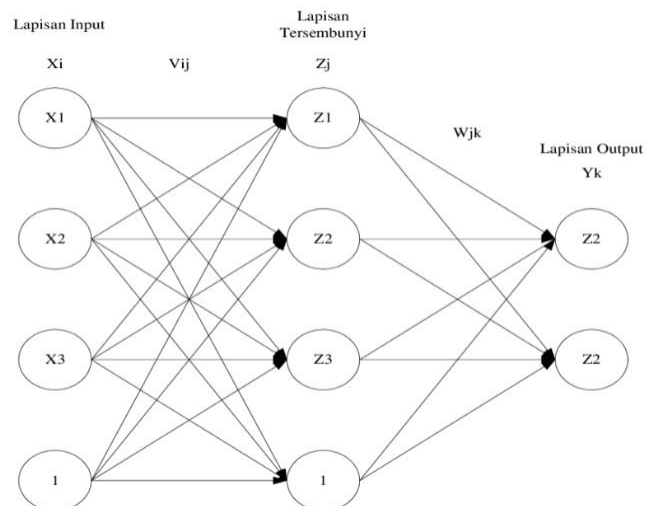
### 2.7.2 *Epoch, Batch dan Iterasi*

*Machine learning* memiliki istilah *epoch*, *batch* dan iterasi, lalu apa perbedaan dari ketiga istilah tersebut. Satu *epoch* adalah dikatakan setelah seluruh *dataset* telah melalui proses *forward* dan *backward* dalam *neural network* sebanyak satu kali. Namun, karena satu *epoch* masih terlalu besar, sehingga dibagi lagi menjadi unit yang lebih kecil yaitu *batch*.

Proses *batching* adalah salah satu pendekatan paling umum untuk mempercepat komputasi pada *neural network* (Lewis, 2017). Proses ini melibatkan penghitungan gradien pada beberapa sampel pelatihan (*batch*). Oleh karena itu, *batch* itu sendiri dapat diartikan sebagai jumlah sampel data yang didistribusikan dalam jaringan saraf. *Batch* berisi beberapa contoh *training* dalam satu kali *forward pass* / *backward pass*. Untuk mengetahui efisiensi komputasi proses *batch*, diasumsikan terdapat ukuran *batch* sebesar 500 dengan 1000 contoh *training*. Maka untuk menyelesaikan satu *epoch* hanya diperlukan 2 iterasi.

### 2.7.3 Propagasi Maju, Mundur dan *Update* Bobot

Fausett (1994) menyatakan bahwa algoritma *backpropagation* adalah algoritma dengan *multi layer perceptron* yang berarti terdapat *input layer*, *hidden layer* dan *output layer* pada algoritma ini. Arsitektur jaringan algoritma *backpropagation* secara umum dapat dilihat pada gambar.



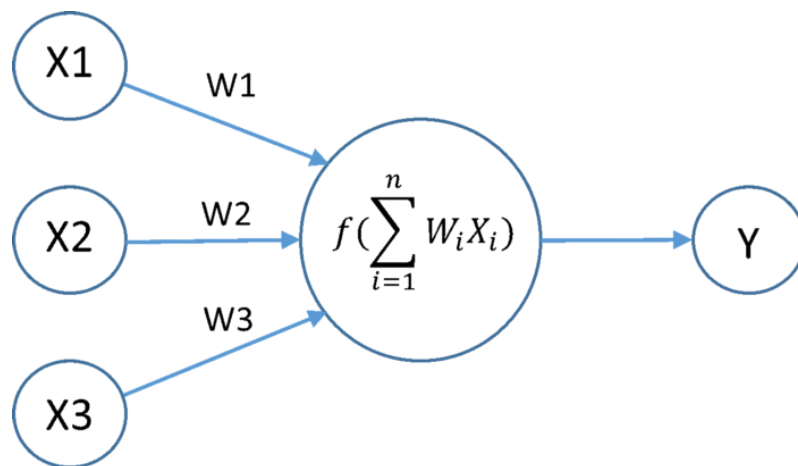
Gambar 8. Arsitektur Jaringan *Backpropagation*  
(Sumber: <https://www.researchgate.net/figure>)

Algoritma *backpropagation* terdapat 3 fase yaitu fase propagasi maju (*forward pass*), propagasi mundur (*backward pass*) dan *update* parameter bobot serta bias (Fausett, 1994).

### 1. Propagasi maju (*forward pass*)

Selama propagasi maju, sinyal *input* ( $x_i$ ) dipropagasikan ke *hidden layer* menggunakan fungsi aktivasi yang ditentukan. *Output* dari setiap unit *hidden layer* ( $z_j$ ) kemudian dipropagasikan kembali ke *hidden layer* di atasnya menggunakan fungsi aktivasi yang ditentukan. Dan terus berlanjut hingga menghasilkan *output* jaringan ( $y_k$ ).

Selanjutnya, bandingkan *output* ( $y_k$ ) dengan tujuan yang harus dicapai ( $t_k$ ). Selisih antara  $t_k$  dengan  $y_k$  adalah kesalahan yang terjadi (*error*). Jika kesalahan ini kurang dari toleransi yang ditentukan, maka iterasi dihentikan. Namun, jika kesalahan masih lebih besar dari batas toleransinya, maka bobot setiap baris pada jaringan akan *update* untuk mengurangi kesalahan yang terjadi (Fausett, 1994).



Gambar 9. Proses Propagasi Maju (*Forward Pass*)  
(Sumber: <https://medium.com/analytics-vidhya/>)

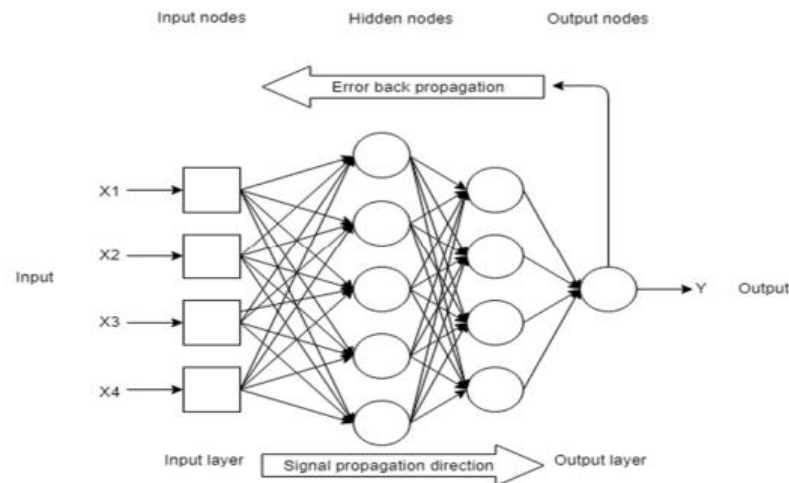
### 2. Propagasi mundur (*backward pass*) dan *update* bobot

Setelah didapatkan nilai kesalahan (*error*), optimasi menggunakan proses *gradient descent* dapat dilakukan (Adam, 2019). Bentuk utama untuk meng*update* suatu bobot yaitu:

$$w_{baru} = w_{lama} - \eta \frac{\partial E}{\partial w} \quad (2.11)$$

Persamaan juga berlaku untuk memperbaiki nilai bias

$$b_{baru} = b_{lama} - \eta \frac{\partial E}{\partial b} \quad (2.12)$$



Gambar 10. Proses *Backward* dan *Update* Bobot  
(Sumber: <https://www.researchgate.net/figure>)

Adam (2019) menyatakan bahwa konsep *chaining* dapat digunakan untuk menghitung  $\frac{\partial E}{\partial w}$ . Proses *backward chaining* digambarkan pada Gambar 10 dengan garis merah. Untuk memperoleh nilai E menggunakan rumus *Mean Square Error (MSE)*.

$$E = \frac{1}{2} (t_j - y_k)^2 \quad (2.13)$$

Tidak terdapat variabel  $w_{jk}$  pada persamaan, tetapi variabel ini bisa didapatkan dengan melakukan *backward* atau “jalan mundur” dimana

$$y_k = \sigma(z_j) \quad (2.14)$$

Variabel y dapat diperoleh dengan menerapkan fungsi aktivitas terhadap variabel z. Sedangkan variabel z dapat dihitung dengan

$$z_j = \sum_{i=1}^n w_{ij} x_i + b_j \quad (2.15)$$

Terlihat pada persamaan bahwa variabel  $w_{jk}$  terdapat pada perhitungan  $z_j$  yang secara tidak langsung berpengaruh terhadap nilai E. Hal ini yang disebut dengan

konsep *chaining* atau rantai. Akan dilakukan proses turunan dengan menggunakan konsep turunan setelah mengetahui hubungan antara  $E$  dan  $w_{jk}$ .

$$\begin{aligned}\frac{\partial f(x)}{\partial x} &= \frac{\partial f(x)}{\partial z} \cdot \frac{\partial z}{\partial x} \\ \frac{\partial E}{\partial w_{ij}} &= \left( \frac{\partial E}{\partial y_k} \right) \cdot \left( \frac{\partial y_k}{\partial w_{ij}} \right) \\ \frac{\partial E}{\partial y_k} &= \frac{\partial}{\partial y_k} \left( \frac{1}{2} (t_j - y_k)^2 \right) \\ &= 2 \times \frac{1}{2} (t_j - y_k)^{2-1} - 1 + 0 \\ &= -(t_j - y_k)\end{aligned}\quad (2.16)$$

Tidak terdapat variabel  $w_{ij}$  pada persamaan  $y_k$ , oleh karena itu untuk menghitung

$\frac{\partial y_k}{\partial w_{ij}}$  diperlukan persamaan (2.15)

Dengan menggunakan konsep turunan, maka didapatkan

$$\begin{aligned}\frac{\partial y_k}{\partial w_{ij}} &= \frac{\partial y_k}{\partial z_j} \cdot \frac{\partial z_j}{\partial w_{ij}} \\ \frac{\partial y_k}{\partial z_j} &= \sigma(z_j) (1 - \sigma(z_j)) = y_k(1 - y_k) \\ \frac{\partial z_j}{\partial w_{ij}} &= \frac{\partial}{\partial w_{ij}} \left( \sum_{i=1}^n w_{ij}x_i + b_j \right) \\ &= x_i\end{aligned}\quad (2.17)$$

Dari perhitungan-perhitungan diatas didapatkan

$$\frac{\partial E}{\partial w_{ij}} = -(t_j - y_k) \times y_k(1 - y_k) \times x_i \quad (2.18)$$

Sehingga untuk mengupdate bobot dapat dilakukan dengan

$$w_{baru} = w_{lama} - \eta \frac{\partial E}{\partial w_{ij}} \quad (2.19)$$

Selain mengupdate bobot, akan ditentukan pula nilai update bias. Proses update nilai bias serupa dengan cara mengupdate nilai bobot menggunakan konsep *chaining* atau rantai. Untuk mengupdate bias  $b_j$  dapat dilakukan dengan persamaan

$$\frac{\partial E}{\partial b_j} = \frac{\partial E}{\partial y_k} \cdot \frac{\partial y_k}{\partial b_j} \quad (2.20)$$

Dimana nilai  $\frac{\partial E}{\partial y_k}$  telah diperoleh pada persamaan (2.16). Konsep turunan digunakan untuk menemukan nilai  $\frac{\partial y_k}{\partial z_j}$ .

$$\frac{\partial y_k}{\partial b_j} = \frac{\partial y_k}{\partial z_j} \cdot \frac{\partial z_j}{\partial b_j} \quad (2.21)$$

Dimana nilai dari  $\frac{\partial y_k}{\partial z_j}$  telah diperoleh pada persamaan (2.17)

$$\frac{\partial z_j}{\partial b_j} = \frac{\partial(\sum_{i=1}^n w_{ij}x_i + b_j)}{\partial b_j} = 1 \quad (2.22)$$

Dari perhitungan-perhitungan tersebut, didapatkan

$$\begin{aligned} \frac{\partial E}{\partial b_j} &= \frac{\partial E}{\partial y_k} \cdot \frac{\partial y_k}{\partial z_j} \cdot \frac{\partial z_j}{\partial b_j} \\ &= -(t_j - y_k) \times y_k(1 - y_k) \times 1 \end{aligned} \quad (2.23)$$

Sehingga *update* bias  $b_j$  dapat dilakukan dengan

$$b_{\text{baru}} = b_{\text{lama}} - \eta \frac{\partial E}{\partial b_j} \quad (2.24)$$

## 2.8 Evaluasi Kinerja Algoritma

Setelah dilakukan proses *training* dan *testing* menggunakan algoritma *backpropagation*, akan dilihat seberapa baik algoritma yang digunakan dalam proses analisis. Proses evaluasi akan ditampilkan berdasarkan proses *training* dan akan dilihat *error* yang dihasilkan oleh proses *testing*.

### 2.8.1 Validasi Model

Teknik peramalan yang menggunakan data kuantitatif dengan data deret waktu tertentu, memiliki *error* / kesalahan yang ditimbulkan oleh teknik tersebut (Hanke dan Wichern, 2014). Oleh karena itu, yang diperlukan adalah metode untuk mengukur seberapa *error* yang dapat dihasilkan oleh metode-metode *forecasting* perlu ditinjau kembali sebelum mengambil keputusan. Metode yang dapat



digunakan untuk mengevaluasi *error* pada teknik *forecasting* adalah *Mean Square Error (MSE)*, *Root Mean Square Error (RMSE)*, *Mean Absolute Derivation (MAD)*, *Mean Absolute Percent Error (MAPE)*, dan *Mean Percent Error (MPE)*.

Metode yang paling umum digunakan adalah *MSE* dan *RMSE*. *Mean Square Error* mengukur rata-rata kuadrat dari *error* antara target yang diamati dan nilai yang diprediksi. Semakin kecil *MSE*, semakin baik hasil prediksinya (Subagyo, 1986).

$$MSE = \frac{1}{n} \sum_{t=1}^n (\hat{y}_t - y_t)^2 \quad (2.25)$$

dengan:

- $\hat{y}_t$  : nilai prediksi pada periode  $t$   
 $y_t$  : nilai aktual pada periode  $t$

*Root Mean Square Error* dapat diartikan sebagai jarak rata-rata karena *MSE* diukur dalam satuan kuadrat, diukur dalam satuan variabel target antara nilai yang diprediksi dan diamati (Lestari, 2012).

$$RMSE = \sqrt{MSE} \quad (2.26)$$

## 2.9 Indeks Standar Pencemaran Udara (ISPU)

Pencemaran udara adalah masuknya atau dimasukkannya zat, energi, atau komponen lain ke udara oleh aktivitas manusia yang melebihi baku mutu udara yang ditetapkan (Waluyo, 2011). Sumber pencemaran udara dapat dibagi menjadi 3 jenis yaitu:

1. Sumber kota dan industri ini berasal dari kemajuan teknologi yang telah melahirkan banyak pabrik industri, pembangkit listrik, dan mobil.
2. Sumber pedesaan/pertanian adalah penggunaan pestisida sebagai zat senyawa kimia (zat pengatur tumbuh dan perangsang tumbuh), virus dan zat-zat lain yang digunakan untuk melindungi tanaman atau bagian tanaman.

3. Sumber alami berasal dari alam, seperti abu yang keluar dari gunung berapi, gas-gas vulkanik, debu yang tertiuap angin, dan bau yang tidak sedap dari proses pembusukan sampah organik dan lainnya.

Indeks Standar Pencemaran Udara (ISPU) berdasarkan Peraturan Pemerintah Republik Indonesia Nomor 41 Tahun 1999 adalah suatu angka yang tidak memiliki satuan yang dapat menjelaskan keadaan kualitas udara di sekitar suatu tempat tertentu berdasarkan dampak pada kesehatan manusia, nilai estetika dan makhluk hidup lainnya (Abidin dan Hasibuan, 2019).

### III. METODOLOGI PENELITIAN

#### 3.1 Tempat dan Waktu Penelitian

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

#### 3.2 Data Penelitian

Data yang digunakan adalah data sekunder yang diperoleh dari <https://data.jakarta.go.id/dataset/> mengenai data histori indeks standar pencemaran udara di Provinsi DKI Jakarta selama 16 bulan yang terhitung sejak April 2020 sampai dengan Juli 2021. Tabel 1 menunjukkan bahwa data yang didapatkan adalah data harian yang memiliki 10 variabel namun hanya akan digunakan 5 variabel saja yaitu: partikulat (PM10), sulfida (SO<sub>2</sub>), nitrit (NO<sub>2</sub>), karbon monoksida (CO) dan ozon (O<sub>3</sub>).

Tabel 1. Data Penelitian

| Tanggal   | PM 10 | PM 25 | SO <sub>2</sub> | CO | O <sub>3</sub> | NO <sub>2</sub> | MA X | CRITICAL | CATEGORI    | LOKASI |
|-----------|-------|-------|-----------------|----|----------------|-----------------|------|----------|-------------|--------|
| 4/01/2020 | 51    | -     | 25              | 35 | 129            | 10              | 129  | O3       | TIDAK SEHAT | DKI5   |
| 4/02/2020 | 55    | -     | 25              | 22 | 57             | 7               | 57   | O3       | SEDANG      | DKI2   |
| 4/03/2020 | 55    | -     | 27              | 22 | 62             | 6               | 62   | O3       | SEDANG      | DKI2   |
| 4/04/2020 | 66    | -     | 28              | 25 | 83             | 7               | 83   | O3       | SEDANG      | DKI2   |

|               |     |     |     |     |         |     |     |      |                |      |
|---------------|-----|-----|-----|-----|---------|-----|-----|------|----------------|------|
| 4/05/20<br>20 | 66  | -   | 28  | 19  | 10<br>3 | 8   | 103 | O3   | TIDAK<br>SEHAT | DKI5 |
| 4/06/20<br>20 | 56  | -   | 26  | 29  | 73      | 9   | 73  | O3   | SEDANG         | DKI5 |
| ...           | ... | ... | ... | ... | ...     | ... | ... | ...  | ...            | ...  |
| 7/26/20<br>21 | 79  | 131 | 54  | 16  | 46      | 32  | 131 | PM25 | TIDAK<br>SEHAT | DKI4 |
| 7/27/20<br>21 | 82  | 140 | 56  | 13  | 41      | 35  | 140 | PM25 | TIDAK<br>SEHAT | DKI4 |
| 7/28/20<br>21 | 82  | 145 | 53  | 18  | 40      | 45  | 145 | PM25 | TIDAK<br>SEHAT | DKI4 |
| 7/29/20<br>21 | 78  | 140 | 52  | 18  | 53      | 39  | 140 | PM25 | TIDAK<br>SEHAT | DKI5 |
| 7/30/20<br>21 | 90  | 154 | 54  | 15  | 81      | 35  | 154 | PM25 | TIDAK<br>SEHAT | DKI4 |
| 7/31/20<br>21 | 63  | 100 | 50  | 18  | 65      | 29  | 100 | PM26 | SEDANG         | DKI4 |

### 3.3 Metode Penelitian

Penelitian ini akan mencari model *neural network* untuk mendapatkan struktur model terbaik menggunakan algoritma *backpropagation* dengan bantuan *software R-studio*. Selain itu akan dievaluasi berdasarkan nilai kesalahan terkecil dari beberapa fungsi aktivasi yang digunakan. Adapun langkah-langkah penelitian ini adalah :

1. Memasukkan (meng-*input*) data indeks standar pencemaran udara yang didapatkan dari <https://data.jakarta.go.id/dataset/> ke dalam *software R-studio*.
2. Menentukan korelasi data dari lima variabel yaitu: partikulat (PM10), sulfida (SO<sub>2</sub>), nitrit (NO<sub>2</sub>), karbon monoksida (CO) dan ozon (O<sub>3</sub>), untuk menentukan variabel mana yang berpengaruh terhadap variabel SO<sub>2</sub>.
3. Mengubah data menjadi format deret waktu.
4. Membuat plot dari data untuk melihat pola penyebaran data.
5. Memeriksa adanya nilai yang hilang pada data.
6. Melakukan *scaling* data menggunakan *min-max normalization*.
7. Melakukan *training* dan *testing* data dengan empat skema, yaitu: skema pertama 60% sebagai data *training* dan 40% sebagai data *testing*, skema kedua 70% sebagai data *training* dan 30% sebagai data *testing*, skema ketiga

80% sebagai data *training* dan 20% sebagai data *testing*, dan skema keempat 90% sebagai data *training* dan 10% sebagai data *testing*.

8. Membangun Model *Neural Network* Menggunakan Algoritma *Backpropagation*.
9. Menentukan *hidden layer* untuk membangun model dengan menggunakan aturan Heaton.
10. Mengaktifkan *neuron* dengan 3 fungsi aktivasi, yaitu: Sigmoid, TanH dan *Rectified Linear Unit (ReLU)*.
11. Melakukan pengujian model yang sudah dibangun dengan mengecek nilai *loss* dan nilai *validation loss*.
12. Melakukan prediksi dengan menggunakan model yang telah diuji.
13. Membuat plot hasil prediksi yang didapat.
14. Mendenormalisasikan hasil prediksi ke bentuk data sebenarnya agar dapat dibandingkan dengan data aktual.
15. Mengevaluasi model dengan *MSE*, *MAPE* dan Akurasi.

Algoritma *backpropagation* dengan menggunakan perhitungan manual adalah sebagai berikut:

Langkah 1: Inisiasi bobot dengan nilai acak yang cukup kecil

- Fase Maju (*Feed Forward Pass*)

Langkah 2: *Input layer* menerima sinyal  $X_i, i = 1, 2, 3, \dots, n$  kemudian meneruskan sinyal tersebut ke lapisan berikutnya yaitu *hidden layer*.

Langkah 3: Tiap-tiap unit pada *hidden layer*  $Z_j, j = 1, 2, 3, \dots, p$  menjumlahkan sinyal-sinyal bobot,

$$Z_{-in_j} = b1_j + \sum_{i=1}^n X_i W_{ij} \quad (3.1)$$

digunakan fungsi aktivasi untuk menghitung sinyal *output*.

$$Z_j = f(Z_{-in_j}) \quad (3.2)$$

dan kirim sinyal tersebut ke semua unit di lapisan atasnya.

Langkah 4: Tiap unit *output*  $Y_k, k = 1, 2, 3, \dots, m$  menjumlahkan sinyal-sinyal *input* terbobot,

$$Y_{-in_k} = b2_k + \sum_{i=1}^p Z_j W_{jk} \quad (3.3)$$

digunakan fungsi aktivasi untuk menghitung sinyal *output*.

$$Y_k = f(Y_{-in_k}) \quad (3.4)$$

- Fase Mundur (*Backward Pass*)

Langkah 5: Tiap unit *output*  $Y_k, k = 1, 2, 3, \dots, m$  menerima target pola yang berhubungan dengan pola *input*.

Langkah 6: Tiap *hidden layer*  $Y_j, j = 1, 2, 3, \dots, p$  menghitung  $\delta_j$  kemudian digunakan untuk menghitung bobot terkoreksi dan bias antara *input* dan *hidden layer*.

Langkah 7: Masing-masing lapisan *output*  $Z_k, k = 1, 2, 3, \dots, m$  memperbaharui nilai bobot dan bias  $j = 1, 2, 3, \dots, p$  dan setiap *hidden layer* memperbaharui pembobot dan bias  $i = 1, 2, 3, \dots, n$  sehingga didapatkan bobot dan bias yang baru.

Langkah 8: Uji syarat (konvergen) terpenuhi, maka iterasi berakhir.

## V. KESIMPULAN

Hasil analisis dan pembahasan pada bab sebelumnya, dapat ditarik beberapa kesimpulan sebagai berikut:

1. Penerapan fungsi aktivasi diperoleh bahwa fungsi aktivasi TanH merupakan fungsi aktivasi terbaik yang digunakan dalam model dengan skema 90% data *training* dan 10% data *testing*.
2. Model *neural network* dengan menggunakan algoritma *backpropagation* untuk memprediksi sulfida (SO<sub>2</sub>) menggunakan variabel *input* yaitu karbon monoksida (CO) dan nitrit (NO<sub>2</sub>) dari data indeks standar pencemaran udara selama 16 bulan yang dihitung sejak April 2020 sampai dengan Juli 2021, diperoleh struktur jaringan terbaik berupa 2 input, 3 *hidden layer* (3 nodes pada *hidden layer* pertama, 2 nodes pada *hidden layer* kedua dan 1 nodes pada *hidden layer* ketiga) dan *output layer*.
3. Model *neural network* yang diperoleh cukup baik dalam memprediksi kandungan indeks Sulfida (SO<sub>2</sub>) dengan menghasilkan nilai MAPE sebesar 15.87412% atau akurasi sebesar 84.12588%.

## DAFTAR PUSTAKA

- Abidin, J., dan Hasibuan, F.E. 2019. Pengaruh Dampak Pencemaran Udara Terhadap Kesehatan Untuk Menambah Pemahaman Masyarakat Awam Tentang Bahaya Dari Polusi Udara. *Prosiding Seminar Nasional Fisika Universitas Riau IV (SNFUR-4)*. Universitas Riau, Pekanbaru.
- Adam, R. 2019. *Contoh Perhitungan Algoritma Backpropagation*. <https://structilmy.com/blog/2019/07/31/contoh-perhitungan-algoritma-backpropagation/>. Diakses pada 28 Oktober 2021.
- Capparuccia, R., Leone, R. D., dan Marchitto, E. 2007. Integrating support vector machines and neural networks. *Neural Networks*. 590-597.
- Fausett, L. 1994. *Fundamentals of Neural Network: Architectures, Algorithms and Applications*. New Jersey: Prencite Hall.
- Fayyad, U., Shapiro, G.P., dan Smyth, P. 1996. *Knowledge Discovery and Data Mining: Towards a Unifying Framework*. AAI Press.
- Ganesh, S. S., Arulmozhivarman, P., dan Tatavarti, V. S. N. R. 2018. Air Quality Index Forecasting Using Artificial Neural Networks. *Int. J. Environment and Waste Management*. 22: 4-22.
- Hanke, J. E., dan Wichern, D. W. 2005. *Business Forecasting, 8th Edition*. New Jersey: Prentice Hall.
- Heaton, J. 2017. *The Number of Hidden Layers*. Retrieved from Heaton Research: <https://www.heatonresearch.com/2017/06/01/hidden-layers.html>.
- Hermawan, A. 2006. *Jaringan Syaraf Tiruan, Teori, dan Aplikasi*. Andi, Yogyakarta.
- James, G., Witten, D., Hastie, T., dan Tibshirani, R. 2017. *An Introduction to Statistical Learning: Applications in R*. Berlin: Springer.
- Krizhevsky, A., Sutskever, I., dan Hinton, G.E. 2012. ImageNet Classification with Deep Convulational Neural Network. *Neural Information Processing System*. 25(2). Canada.



- Kulkarni, S., dan Harman, G. 2001. *An Elementary Introduction to Statistical Learning Theory*. Wiley and Sons, Canada.
- Kusumadewi, S. 2003. *Artificial Intellegent*. Graha Ilmu, Yogyakarta.
- Lestari, N., dan Wahyuningsih, N. 2012. Peramalan Kunjungan Wisata dengan Pendekatan Model SARIMA. *Jurnal Sains dan Seni ITS*. 1(1): 29-33.
- Lewis, N.D. 2017. *Neural Network for Time Series Forecasting with R*. CrateSpace Independent Publishing Platform, US.
- Makridakis, S., Wheelwright, S. C., dan Mc Gee, V. E. 1999. *Metode dan Aplikasi Peramalan* (2nd ed). Translated by Untung Sus Andriyanto dan Abdul Basith. Erlangga, Jakarta.
- Maleki, H., Sorooshian, A., Goudarzi, G., Baboli, Z., Birgani, Y. T., dan Rahmati, M. 2019. Air Pollution Prediction By Using An Artificial Neural Network Model. *Clean Technologies and Environmental Policy*, hlm. 1341–1352.
- Muziansyah, D. D. 2015 . Model Emisi Gas Buangan Kendaraan Bermotor Akibat Aktivitas Transportasi. *Studi Kasus, Terminal Pasar Bawah Ramayana Kota Bandar Lampung*, 57 - 70.
- Prasetyo, E. 2014. *Data Mining: Mengolah Data Menjadi Informasi Menggunakan Matlab*. Andi, Yogyakarta.
- Rahman, N. H. A., Lee, M. H., Latif, M. T., dan Suhartono. 2013. Forecasting of Air Pollution Index with Artificial Neural Network. *Jurnal Teknologi*. 63(2): 59-64.
- Ramchoun, H., Idrissi, M. A. J., Ghanou, Y., dan Ettaouil, M. 2016. Multilayer Perceptron: Architecture Optimization and Training. *International Journal of Interactive Multimedia and Artificial Intelligence*. 4(1).
- Sastrawijaya, A. T. 2009. *Pencemaran Lingkungan*. Rineka Cipta, Jakarta.
- Sharma, A. dan Nijhawan, G. 2015. Rainfall Prediction Using Neural Network. *International Journal of Computer Science Trends and Technology*. 3(3): 65-69.
- Shynk, J.J. 1990. Performance Surface of a Single-Layer Perceptron. *IEEE Transactions on Neural Networks*. 1(3).
- Subagyo, Pangestu. 1986. *Forecasting Konsep dan Aplikasi*. Yogyakarta: BPPE UGM.

Waluyo, E. C. 2011. Kajian Tingkat Pencemaran Sulfur Dioksida Dari Industri Di Beberapa Daerah Di Indonesia. *Berita Dirgantara* Vol. 12 No.4 Desember 2011: 132-137.

Wardhana, W.A. 2001. *Dampak pencemaran lingkungan*. Andi, Yogyakarta.