

**PENENTUAN TINGKAT KERUSAKAN SEKTOR PASCA BENCANA ALAM  
MENGUNAKAN METODE TOPSIS BERBASIS *MACHINE LEARNING***

**SKRIPSI**

**Oleh:**

**AMILA FADHILA RAHMANIATI**

**NIM. 18650127**



**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM  
MALANG  
2022**

**PENENTUAN TINGKAT KERUSAKAN SEKTOR PASCA BENCANA  
ALAM MENGGUNAKAN METODE TOPSIS BERBASIS *MACHINE  
LEARNING***

**SKRIPSI**

**Diajukan kepada:  
Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang  
Untuk Memenuhi Salah Satu Persyaratan Dalam  
Memperoleh Gelar Sarjana Komputer (S.Kom)**

**Oleh:  
AMILA FADHILA RAHMANIATI  
NIM. 18650127**

**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM  
MALANG  
2022**

**HALAMAN PERSETUJUAN**

**PENENTUAN TINGKAT KERUSAKAN SEKTOR PASCA BENCANA  
ALAM MENGGUNAKAN METODE TOPSIS BERBASIS *MACHINE  
LEARNING***

Oleh:  
**AMILA FADHILA RAHMANIATI**  
NIM. 18650127

Telah Diperiksa dan Disetujui Untuk Diuji  
Tanggal : 30 November 2022

Pembimbing I



Agung Teguh Wibowo Almais, M.T.  
NIDT. 19860103 20180201 1 235

Pembimbing II



Dr. M. Imamudin Lc, M.A.  
NIP. 19740602 200901 1 010

Mengetahui,  
Ketua Program Studi Teknik Informatika  
Fakultas Sains dan Teknologi  
Universitas Islam Negeri Maulana Malik Ibrahim Malang



  
Dr. Fachrul Kurniawan, M.MT.  
NIP. 19771020 200912 1 001

## HALAMAN PENGESAHAN

### PENENTUAN TINGKAT KERUSAKAN SEKTOR PASCA BENCANA ALAM MENGGUNAKAN METODE TOPSIS BERBASIS *MACHINE* *LEARNING*

#### SKRIPSI

Oleh:  
**AMILA FADHILA RAHMANIATI**  
NIM. 18650127

Telah Dipertahankan di Depan Dewan Penguji Skripsi  
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan  
Untuk Memperoleh Gelar Sarjana Komputer (S.Kom)  
Tanggal : 14 Desember 2022

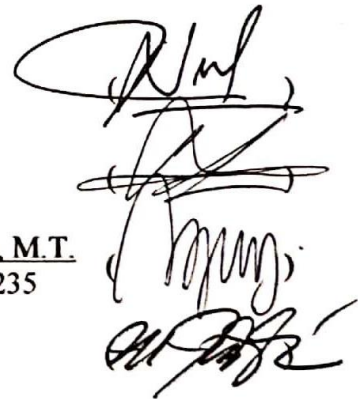
#### Susunan Dewan Penguji

Ketua Penguji : Dr. Fresy Nugroho, M.T.  
NIP. 19710722 201101 1 001

Anggota Penguji I : Supriyono, M.Kom.  
NIP. 19841010 201903 1 012


Anggota Penguji II : Agung Teguh Wibowo Almais, M.T.  
NIDT. 19860103 20180201 1 235

Anggota Penguji III : Dr. M. Imamudin Lc., M.A.  
NIP. 19740602 200901 1 010



Mengetahui,  
Ketua Program Studi Teknik Informatika  
Fakultas Sains dan Teknologi  
Universitas Islam Negeri Maulana Malik Ibrahim Malang



  
Dr. Fachrul Kurniawan, M.MT.  
NIP. 19771020 200912 1 001

## PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan dibawah ini:

Nama : Amila Fadhila Rahmaniati  
NIM : 18650127  
Fakultas/Program Studi : Sains dan Teknologi/Teknik Informatika  
Judul Skripsi : PENENTUAN TINGKAT KERUSAKAN  
SEKTOR PASCA BENCANA ALAM  
MENGUNAKAN METODE TOPSIS  
BERBASIS *MACHINE LEARNING*

Menyatakan dengan sebenarnya bahwa Skripsi yang saya tulis ini benar-benar merupakan hasil karya sendiri, bukan merupakan pengambilalihan data, tulisan atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka.

Apabila dikemudian hari terbukti atau dapat dibuktikan Skripsi ini hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 30 November 2022  
Yang membuat pernyataan,



Amila Fadhila Rahmaniati  
NIM. 18650127

**HALAMAN MOTTO**

*“Stay Positive and Always Be Grateful”*

## **HALAMAN PERSEMBAHAN**

Segala puji syukur ke hadirat Allah SWT. yang telah melimpahkan rahmat dan hidayah-Nya, sehingga penulis dapat menyelesaikan tugas skripsi ini dengan baik dan lancar. Shalawat serta salam juga dihaturkan kepada Nabi Muhammad SAW. yang telah membawa kita dari zaman kebodohan menuju zaman kebenaran yakni Islam. Penulis mempersembahkan tugas skripsi ini untuk seluruh pihak yang telah berjasa dalam pengerjaan penelitian ini.

Kepada kedua orang tua penulis, Bapak Agus Maimun dan Ibu Novi Khila Firani, serta adik penulis, Azry Ahsan Syauqi, yang sangat penulis cintai, yang tidak pernah berhenti dalam memberikan motivasi, dukungan, semangat, maupun doa kepada penulis hingga penulis dapat menyelesaikan tugas skripsi ini dengan baik dan lancar. Tidak lupa juga untuk Yuk Paini, yang selalu mengingatkan untuk jangan lupa makan, dan seluruh keluarga besar penulis yang senantiasa mendukung dan mendoakan penulis.

Kepada Bapak Agung Teguh Wibowo Almais, M.T. selaku Dosen Pembimbing I dan Bapak Dr. M. Imamudin Lc., M.A. selaku Dosen Pembimbing II yang senantiasa membimbing, memberi semangat, memberikan arahan dan masukan, serta membantu penulis dalam mengerjakan dan menyelesaikan tugas skripsi. Kepada seluruh Dosen dan Jajaran Staff Program Studi Teknik Informatika yang senantiasa membantu penulis secara langsung maupun tidak langsung dalam penulisan dan pengerjaan tugas skripsi yang dilakukan penulis.

Kepada seluruh teman angkatan seperjuangan penulis, UFO 2018 dan Alien 2019, serta teman-teman penghuni Lab. Database sesama pejuang skripsi, yang turut memberikan saran, semangat, motivasi, dan informasi kepada penulis selama masa studi. Seluruh keluarga, teman, sahabat, dan kerabat penulis yang tidak dapat penulis sebutkan satu per satu, yang turut memberikan bantuan, semangat, dukungan, dan doa untuk penulis.



## KATA PENGANTAR

*Assalamu'alaikum Warahmatullahi Wabarakatuh*

*Alhamdulillah* rabbilalamin, segala puji syukur ke hadirat Allah SWT. yang telah melimpahkan rahmat dan hidayah-Nya, sehingga penulis dapat menyelesaikan tugas skripsi yang berjudul “Penentuan Tingkat Kerusakan Sektor Pasca Bencana Alam Menggunakan Metode Topsis Berbasis *Machine Learning*” dengan baik dan lancar. Shalawat serta salam dihaturkan dan dilimpahkan kepada Nabi Muhammad SAW. yang telah membawa kita dari zaman kebodohan menuju zaman kebenaran yakni Islam dan zaman yang penuh dengan ilmu pengetahuan sebagaimana yang kita rasakan pada saat ini.

Penulis melaksanakan penelitian pada tugas skripsi ini untuk memenuhi salah satu syarat lulus sebagai sarjana komputer pada Program Studi Teknik Informatika, Fakultas Sains dan Teknologi, UIN Maulana Malik Ibrahim Malang. Dalam pengerjaan dan penyusunan tugas skripsi ini, banyak sekali pihak yang membantu penulis baik secara langsung maupun tidak langsung. Penulis mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Prof. Dr. M. Zainuddin, M.A. selaku Rektor Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang.
2. Dr. Sri Harini, M.Si. selaku Dekan Fakultas Sains dan Teknologi UIN Maulana Malik Ibrahim Malang.
3. Dr. Fachrul Kurniawan M.MT. selaku Ketua Program Studi UIN Maulana Malik Ibrahim Malang.

4. Agung Teguh Wibowo Almais, M.T. selaku Dosen Pembimbing I yang senantiasa membimbing, memberi semangat, memberikan arahan dan masukan, serta membantu penulis dalam mengerjakan dan menyelesaikan tugas skripsi.
5. Dr. M. Imamudin Lc., M.A. selaku Dosen Pembimbing II yang senantiasa memberikan bimbingan dan masukan untuk penulis dalam mengerjakan dan menyelesaikan tugas skripsi.
6. Dr. Fresy Nugroho, M.T. selaku Dosen Penguji I dan Supriyono, M.Kom. selaku Dosen Penguji II yang telah memberikan ilmu, kritik, saran, dan masukan untuk penulis agar penulis dapat menyelesaikan dan mengerjakan tugas skripsi dengan baik.
7. Dr. Cahyo Crysdiان selaku dosen wali yang senantiasa membantu serta turut memberikan ilmu dan saran untuk penulis selama menjalani masa studi pada Program Studi Teknik Informatika UIN Maulana Malik Ibrahim Malang.
8. Bapak Agus Maimum dan Ibu Novi Khila Firani selaku kedua orang tua penulis, serta Azry Ahsan Syauqi selaku adik penulis, yang senantiasa mendampingi, memotivasi, dan mendukung penulis selama mengerjakan tugas skripsi, serta tidak pernah lepas mendoakan penulis hingga dapat menyelesaikan tugas skripsi.
9. Seluruh Dosen dan Jajaran Staff Program Studi Teknik Informatika yang senantiasa membantu penulis secara langsung maupun tidak langsung dalam penulisan dan pengerjaan tugas skripsi yang dilakukan penulis.

10. Seluruh teman angkatan UFO 2018 dan Alien 2019, terutama teman-teman penghuni Lab. Database sesama pejuang skripsi, yang turut memberikan motivasi dan semangat kepada penulis selama masa studi hingga pengerjaan skripsi selesai.
11. Seluruh keluarga, teman, sahabat, dan kerabat penulis yang tidak dapat penulis sebutkan satu per satu, yang turut memberikan bantuan, semangat, dukungan, dan doa untuk penulis.
12. Diri sendiri yang telah berusaha dan pantang menyerah dalam menghadapi segala cobaan dan drama yang terjadi selama masa studi hingga selesainya tugas skripsi.

Penulis menyadari bahwa penelitian yang dilakukan pada tugas skripsi ini belum sempurna dan masih memiliki banyak sekali kekurangan. Oleh sebab itu, penulis sangat terbuka untuk kritik dan saran yang membangun dari para pembaca sebagai masukan untuk lebih baik lagi. Penelitian yang dilakukan pada tugas skripsi ini juga dapat dikembangkan lagi pada penelitian selanjutnya, sebagai pelengkap untuk kekurangan yang ada. Penulis juga mengharapkan bahwa penelitian yang dilakukan dapat memberikan berbagai manfaat tak hanya bagi pembaca, tetapi juga bagi masyarakat luas.

*Wassalamu'alaikum Warahmatullahi Wabarakatuh*

Malang, 30 November 2022

Penulis

## DAFTAR ISI

<b>HALAMAN JUDUL .....</b>	<b>ii</b>
<b>HALAMAN PERSETUJUAN .....</b>	<b>iii</b>
<b>HALAMAN PENGESAHAN.....</b>	<b>iv</b>
<b>PERNYATAAN KEASLIAN TULISAN .....</b>	<b>v</b>
<b>HALAMAN MOTTO .....</b>	<b>vi</b>
<b>HALAMAN PERSEMBAHAN .....</b>	<b>vii</b>
<b>KATA PENGANTAR.....</b>	<b>ix</b>
<b>DAFTAR ISI.....</b>	<b>xii</b>
<b>DAFTAR GAMBAR.....</b>	<b>xiv</b>
<b>DAFTAR TABEL .....</b>	<b>xvi</b>
<b>ABSTRAK .....</b>	<b>xvii</b>
<b>ABSTRACT.....</b>	<b>xviii</b>
<b>المخلص.....</b>	<b>xix</b>
<b>BAB I PENDAHULUAN.....</b>	<b>1</b>
1.1 Latar Belakang.....	1
1.2 Pernyataan Masalah.....	6
1.3 Tujuan Penelitian.....	6
1.4 Batasan Masalah.....	7
1.5 Manfaat Penelitian.....	7
<b>BAB II STUDI PUSTAKA .....</b>	<b>8</b>
2.1 Penelitian Terdahulu.....	8
2.2 Bencana Alam .....	12
2.3 Sistem Pendukung Keputusan .....	12
2.4 Metode ROC.....	13
2.5 Metode TOPSIS .....	14
2.6 <i>Machine Learning</i> dan <i>Forward Propagation in Neural Network</i> .....	16
2.7 MSE, MAPE, dan Akurasi .....	20
<b>BAB III DESAIN DAN IMPLEMENTASI .....</b>	<b>22</b>
3.1 <i>Data Collection</i> .....	22
3.2 <i>Preprocessing</i> .....	30
3.3 Desain Sistem .....	32
3.4 <i>Research Instrument</i> .....	33
3.5 <i>Conceptual Framework</i> .....	35
3.6 Perhitungan Manual .....	36
3.7 Hipotesis .....	44
<b>BAB IV UJI COBA DAN PEMBAHASAN.....</b>	<b>45</b>

4.1 Implementasi Sistem .....	45
4.1.1 Implementasi Metode TOPSIS .....	45
4.1.1.1 <i>Database</i> Sistem .....	45
4.1.1.2 Tampilan Sistem.....	55
4.1.2 Implementasi <i>Forward Propagation in Neural Network</i> .....	66
4.2 Uji Coba .....	69
4.2.1 Pengolahan Data dengan Metode TOPSIS .....	69
4.2.2 Pengujian Data dengan <i>Forward Propagation in Neural Network</i> .....	74
4.2.3 Perhitungan MSE, MAPE, dan Akurasi .....	220
4.3 Pembahasan .....	273
4.4 Integrasi Islam .....	278
4.4.1 <i>Muamalah Ma'a Allah</i> .....	280
4.4.2 <i>Muamalah Ma'a an-Nas</i> .....	281
4.4.3 <i>Muamalah Ma'a al-Alam</i> .....	282
<b>BAB V KESIMPULAN DAN SARAN .....</b>	<b>284</b>
5.1 Kesimpulan.....	284
5.2 Saran .....	285
<b>DAFTAR PUSTAKA</b>	
<b>LAMPIRAN</b>	

## DAFTAR GAMBAR

Gambar 2. 1 Arsitektur <i>Neural Networks</i> .....	18
Gambar 3. 1 Desain Sistem.....	32
Gambar 3. 2 <i>Conceptual Framework</i> .....	35
Gambar 4. 1 <i>Database</i> Sistem.....	46
Gambar 4. 2 Tabel Alternatif .....	46
Gambar 4. 3 Tabel Kriteria .....	47
Gambar 4. 4 Tabel Bobot .....	47
Gambar 4. 5 Tabel <i>Cases</i> .....	48
Gambar 4. 6 Tabel Skala.....	48
Gambar 4. 7 Tabel Matrix Keputusan.....	49
Gambar 4. 8 Tabel <i>User</i> .....	50
Gambar 4. 9 <i>Pseudocode View</i> Matrix Keputusan.....	51
Gambar 4. 10 <i>Pseudocode View</i> Topsis Pembagi.....	51
Gambar 4. 11 <i>Pseudocode View</i> Topsis Normalisasi .....	52
Gambar 4. 12 <i>Pseudocode View</i> Topsis Terbobot .....	53
Gambar 4. 13 <i>Pseudocode View</i> Topsis Max Min.....	54
Gambar 4. 14 <i>Pseudocode View</i> Topsis SIP SIN.....	54
Gambar 4. 15 <i>Pseudocode View</i> Topsis Nilai V .....	55
Gambar 4. 16 <i>Pseudocode View Result</i> .....	55
Gambar 4. 17 Halaman <i>Login</i> .....	56
Gambar 4. 18 Halaman <i>Dashboard</i> .....	56
Gambar 4. 19 Halaman Data <i>User</i> .....	57
Gambar 4. 20 Halaman Data Kriteria .....	57
Gambar 4. 21 Halaman Data Alternatif .....	58
Gambar 4. 22 Halaman Data Bobot .....	59
Gambar 4. 23 Halaman Skala Penilaian.....	59
Gambar 4. 24 Halaman Data Kerusakan Bangunan.....	60
Gambar 4. 25 Halaman Input Data (1).....	60
Gambar 4. 26 Halaman Input Data (2).....	61
Gambar 4. 27 Halaman Perhitungan TOPSIS ( <i>Step 1</i> ) .....	62
Gambar 4. 28 Halaman Perhitungan TOPSIS ( <i>Step 2</i> ) .....	62
Gambar 4. 29 Halaman Perhitungan TOPSIS ( <i>Step 3</i> ) .....	63
Gambar 4. 30 Halaman Perhitungan TOPSIS ( <i>Step 4</i> ) .....	64
Gambar 4. 31 Halaman Perhitungan TOPSIS ( <i>Step 5</i> ) .....	64
Gambar 4. 32 Halaman Perhitungan TOPSIS ( <i>Step 6</i> ) .....	65
Gambar 4. 33 Halaman Hasil .....	65
Gambar 4. 34 Halaman Profil .....	66
Gambar 4. 35 <i>Pseudocode</i> Impor Data .....	67
Gambar 4. 36 <i>Pseudocode Forward Propagation in Neural Network</i> .....	68
Gambar 4. 37 Grafik Hasil Model A1.....	83
Gambar 4. 38 Grafik Hasil Model A2.....	89
Gambar 4. 39 Grafik Hasil Model A3.....	94
Gambar 4. 40 Grafik Hasil Model A4.....	100

Gambar 4. 41 Grafik Hasil Model A5.....	107
Gambar 4. 42 Grafik Hasil Model B1 .....	112
Gambar 4. 43 Grafik Hasil Model B2.....	117
Gambar 4. 44 Grafik Hasil Model B3.....	123
Gambar 4. 45 Grafik Hasil Model B4.....	129
Gambar 4. 46 Grafik Hasil Model B5.....	135
Gambar 4. 47 Grafik Hasil Model C1 .....	140
Gambar 4. 48 Grafik Hasil Model C2.....	145
Gambar 4. 49 Grafik Hasil Model C3.....	151
Gambar 4. 50 Grafik Hasil Model C4.....	157
Gambar 4. 51 Grafik Hasil Model C5.....	163
Gambar 4. 52 Grafik Hasil Model D1.....	168
Gambar 4. 53 Grafik Hasil Model D2.....	173
Gambar 4. 54 Grafik Hasil Model D3.....	179
Gambar 4. 55 Grafik Hasil Model D4.....	185
Gambar 4. 56 Grafik Hasil Model D5.....	191
Gambar 4. 57 Grafik Hasil Model E1 .....	196
Gambar 4. 58 Grafik Hasil Model E2 .....	201
Gambar 4. 59 Grafik Hasil Model E3 .....	207
Gambar 4. 60 Grafik Hasil Model E4 .....	213
Gambar 4. 61 Grafik Hasil Model E5 .....	220
Gambar 4. 62 Grafik MSE, MAPE, dan Akurasi Model A .....	275
Gambar 4. 63 Grafik MSE, MAPE, dan Akurasi Model B .....	275
Gambar 4. 64 Grafik MSE, MAPE, dan Akurasi Model C .....	276
Gambar 4. 65 Grafik MSE, MAPE, dan Akurasi Model D .....	276
Gambar 4. 66 Grafik MSE, MAPE, dan Akurasi Model E.....	277

## DAFTAR TABEL

Tabel 3. 1 Data Kejadian Bencana Alam.....	22
Tabel 3. 2 Alternatif (Safitri et al., 2022).....	28
Tabel 3. 3 Kriteria (Safitri et al., 2022).....	29
Tabel 3. 4 Ketentuan Nilai Kriteria (Bachriwindi et al., 2019).....	30
Tabel 3. 5 Penentuan Kriteria.....	37
Tabel 3. 6 Perhitungan Metode ROC.....	37
Tabel 3. 7 Preferensi Alternatif dan Kriteria.....	38
Tabel 3. 8 Solusi Ideal Positif dan Negatif .....	40
Tabel 3. 9 Nilai Pemisahan .....	41
Tabel 3. 10 Nilai Preferensi .....	42
Tabel 3. 11 Hasil Metode TOPSIS.....	43
Tabel 4. 1 Data Hasil <i>Preprocessing</i> .....	69
Tabel 4. 2 Data Hasil Metode TOPSIS .....	72
Tabel 4. 3 Data <i>Training</i> .....	74
Tabel 4. 4 Data <i>Testing</i> .....	75
Tabel 4. 5 Model FPNN .....	77
Tabel 4. 6 Hasil Perhitungan MSE, MAPE, dan Akurasi Model A1.....	220
Tabel 4. 7 Hasil Perhitungan MSE, MAPE, dan Akurasi Model A2.....	223
Tabel 4. 8 Hasil Perhitungan MSE, MAPE, dan Akurasi Model A3.....	225
Tabel 4. 9 Hasil Perhitungan MSE, MAPE, dan Akurasi Model A4.....	227
Tabel 4. 10 Hasil Perhitungan MSE, MAPE, dan Akurasi Model A5.....	229
Tabel 4. 11 Hasil Perhitungan MSE, MAPE, dan Akurasi Model B1 .....	231
Tabel 4. 12 Hasil Perhitungan MSE, MAPE, dan Akurasi Model B2 .....	233
Tabel 4. 13 Hasil Perhitungan MSE, MAPE, dan Akurasi Model B3 .....	235
Tabel 4. 14 Hasil Perhitungan MSE, MAPE, dan Akurasi Model B4.....	237
Tabel 4. 15 Hasil Perhitungan MSE, MAPE, dan Akurasi Model B5.....	239
Tabel 4. 16 Hasil Perhitungan MSE, MAPE, dan Akurasi Model C1 .....	241
Tabel 4. 17 Hasil Perhitungan MSE, MAPE, dan Akurasi Model C2.....	244
Tabel 4. 18 Hasil Perhitungan MSE, MAPE, dan Akurasi Model C3 .....	246
Tabel 4. 19 Hasil Perhitungan MSE, MAPE, dan Akurasi Model C4.....	248
Tabel 4. 20 Hasil Perhitungan MSE, MAPE, dan Akurasi Model C5 .....	250
Tabel 4. 21 Hasil Perhitungan MSE, MAPE, dan Akurasi Model D1.....	252
Tabel 4. 22 Hasil Perhitungan MSE, MAPE, dan Akurasi Model D2.....	254
Tabel 4. 23 Hasil Perhitungan MSE, MAPE, dan Akurasi Model D3.....	256
Tabel 4. 24 Hasil Perhitungan MSE, MAPE, dan Akurasi Model D4.....	258
Tabel 4. 25 Hasil Perhitungan MSE, MAPE, dan Akurasi Model D5.....	260
Tabel 4. 26 Hasil Perhitungan MSE, MAPE, dan Akurasi Model E1 .....	262
Tabel 4. 27 Hasil Perhitungan MSE, MAPE, dan Akurasi Model E2 .....	265
Tabel 4. 28 Hasil Perhitungan MSE, MAPE, dan Akurasi Model E3 .....	267
Tabel 4. 29 Hasil Perhitungan MSE, MAPE, dan Akurasi Model E4 .....	269
Tabel 4. 30 Hasil Perhitungan MSE, MAPE, dan Akurasi Model E5 .....	271
Tabel 4. 31 Hasil MSE, MAPE, dan Akurasi Setiap Model Data .....	273



## ABSTRAK

Rahmaniati, Amila Fadhila. 2022. *Penentuan Tingkat Kerusakan Sektor Pasca Bencana Alam Menggunakan Metode Topsis Berbasis Machine Learning*. Skripsi. Program Studi Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing : (I) Agung Teguh Wibowo Almais, M.T (II) Dr. M. Imamudin Lc., M.A.

---

Kata Kunci : Surveyor, Pasca Bencana Alam, TOPSIS, *Machine Learning*, *Forward Propagation*, *Neural Network*

Salah satu kerugian yang disebabkan oleh bencana alam adalah kerusakan dari sektor bangunan. Pada umumnya, penentuan tingkat kerusakan bangunan setelah terjadinya bencana dilakukan oleh tim surveyor. Akan tetapi, seringkali terjadi ketidakakuratan data dengan yang terjadi di lapangan, akibat penilaian subjektif yang menyebabkan pengkategorian data menjadi berbeda. Oleh karena itu, pada penelitian ini dilakukan implementasi untuk menentukan tingkat kerusakan sektor pasca bencana alam. Penelitian ini dilakukan dengan tujuan untuk mengefisiensi waktu dan membantu tim surveyor dalam menentukan tingkat kerusakan sektor setelah terjadinya bencana alam. Data yang digunakan pada penelitian ini merupakan data kejadian bencana alam dari Badan Penanggulangan Bencana Daerah (BPBD) Kota Malang. Ada 2 metode yang digunakan, antara lain metode TOPSIS, yang digunakan untuk mengolah *data training* untuk model *machine learning*, serta *forward propagation in neural network*, yang merupakan metode *machine learning* dan digunakan dalam pengujian data. Hasil dari penelitian ini terdiri dari 25 model yang dibedakan berdasarkan jumlah *epoch* dan *hidden layer*. Hasil akurasi terbesar didapatkan oleh model E5 dengan *epoch* 125 dan 5 *hidden layer*, yang menghasilkan nilai MSE 0,06, MAPE 3%, dan akurasi 97%. Sehingga, dapat disimpulkan bahwa pada *forward propagation in neural network*, jumlah *hidden layer* dan *epoch* berpengaruh pada besar akurasi perhitungan metode yang dihasilkan. Semakin banyak jumlah *epoch* dan *hidden layer*, maka tingkat akurasi sistem juga akan semakin besar.

## ABSTRACT

Rahmaniati, Amila Fadhila. 2022. *Determination of The Sector Damage Level of The Post Natural Disaster Using The TOPSIS Method Based on Machine Learning*. Theses. Department of Informatics Engineering Faculty of Science and Technology Maulana Malik Ibrahim State Islamic University of Malang. Advisor : (I) Agung Teguh Wibowo Almais, M.T (II) Dr. M. Imamudin Lc., M.A.

---

Keywords : Surveyor, Post Natural Disaster, TOPSIS, Machine Learning, Forward Propagation, Neural Network

One of the losses that can be caused by natural disasters is damage to the building sector. Usually, the determination of the damage level of the building sector after a disaster is carried out by a surveyor team. However, there are many data inaccuracies compared to what happens in the field, due to subjective assessments which can cause data to be categorized differently. Therefore, in this study, an implementation was carried out to determine the sector damage level after a natural disaster. This research was conducted with the aim of saving time and assisting the surveyor team in determining the sector damage level after a natural disaster. The data used in this study is natural disaster events data from the Regional Disaster Management Agency of Malang City. There are 2 methods used, including the TOPSIS method, which is used to process training data for machine learning models, and forward propagation in neural networks, which is a machine learning method and is used in data testing. The results of this study consist of 25 models that are distinguished by the number of epochs and hidden layers. The highest accuracy results were obtained by the E5 model with epoch 125 and 5 hidden layers, which produced an MSE value of 0.06, MAPE 3%, and an accuracy of 97%. Thus, it can be concluded that in forward propagation in a neural network, the number of hidden layers and epochs has an effect on the accuracy of the method calculations results. The greater the number of epochs and hidden layers, the greater the level of system accuracy.

## الملخص

رحمانية, عاملا فاضلا .2022. تحديد مستوى الضرر القطاعي لما بعد الكارثة الطبيعية باستخدام طريقة TOPSIS القائمة على التعلم الآلي. برنامج دراسة هندسة المعلوماتية، كلية العلوم والتكنولوجيا، جامعة مولانا مالك إبراهيم الإسلامية الحكومية مالانج. مشرف: (١) اجونج تيجوه ويوو الميس الما جستير (٢) الدكتور محمد إمام الدين الماجستير.

الكلمات الرئيسية: مساح، ما بعد الكوارث الطبيعية، TOPSIS، التعلم الآلي، الانتشار المتقدم، الشبكة العصبية

من الحسائر التي يمكن أن تسببها الكوارث الطبيعية الأضرار التي لحقت بقطاع البناء. عادة، يتم تحديد مستوى الضرر في قطاع البناء بعد وقوع الكارثة بواسطة فريق مساح. ومع ذلك، هناك العديد من عدم دقة البيانات مقارنة بما يحدث في الميدان، بسبب التقييمات الذاتية التي يمكن أن تؤدي إلى تصنيف البيانات بشكل مختلف. لذلك، في هذه الدراسة، تم تنفيذ لتحديد مستوى الضرر القطاعي بعد وقوع كارثة طبيعية. تم إجراء هذا البحث بهدف توفير الوقت ومساعدة فريق المساح في تحديد مستوى الضرر في القطاع بعد وقوع كارثة طبيعية. البيانات المستخدمة في هذه الدراسة هي بيانات أحداث الكوارث الطبيعية من وكالة إدارة الكوارث الإقليمية في مدينة مالانج. هناك طريقتان مستخدمتان، بما في ذلك طريقة TOPSIS، والتي تُستخدم لمعالجة بيانات التدريب لنماذج التعلم الآلي، والانتشار الأمامي في الشبكات العصبية، وهي طريقة تعلم آلي تُستخدم في اختبار البيانات. تتكون نتائج هذه الدراسة من 25 نماذج تتميز بعدد الحقب والطبقات المخفية. تم الحصول على أعلى نتائج دقة بواسطة نموذج E5 مع طبقات 125 و 5 طبقات مخفية، والتي أنتجت قيمة MSE تبلغ 0.06 و 3% MAPE ودقة تبلغ 97%. وبالتالي، يمكن استنتاج في الانتشار الأمامي في الشبكة العصبية، فإن عدد الطبقات والعهود المخفية له تأثير على دقة نتائج حسابات الطريقة الناتجة. كلما زاد عدد الحقب والطبقات المخفية، زاد مستوى دقة النظام.

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Indonesia dinobatkan sebagai salah satu negara yang memiliki potensi besar untuk terdampak bencana alam. Berdasarkan data yang dirilis oleh Badan Nasional Penanggulangan Bencana (BNPB), terdapat 3.058 kejadian bencana alam yang melanda berbagai wilayah di Indonesia selama tahun 2021 (*Data Bencana Indonesia, 2022*). Kondisi Indonesia yang sangat rawan bencana alam tersebut dipengaruhi beberapa faktor, salah satunya yakni faktor iklim dan geografis. Indonesia berada pada wilayah yang beriklim tropis, yang menjadikan Indonesia hanya terdiri dari dua musim yakni hujan dan kemarau. Hal tersebut selain menjadikan tanah Indonesia menjadi subur, juga menyebabkan Indonesia rawan untuk terdampak bencana hidrometeorologi seperti kekeringan, banjir, dan kebakaran hutan (BNPB, 2016). Indonesia secara geografis juga dilewati oleh tiga lempeng tektonik dunia. Selain itu, Indonesia juga dilewati oleh sabuk vulkanik yang membentang di wilayah timur dan selatan Indonesia melewati Pulau Sumatera, Jawa, Nusa Tenggara, hingga Sulawesi. Kondisi geografis Indonesia tersebut menyebabkan Indonesia juga rawan terdampak berbagai bencana alam seperti contohnya banjir, gempa bumi, gunung berapi, tanah longsor, hingga tsunami.

Bencana alam sendiri memiliki pengertian yakni peristiwa atau serangkaian peristiwa yang dapat menimbulkan kerugian dalam kehidupan masyarakat yang

ditimbulkan karena faktor alam (Undang-Undang Republik Indonesia Nomor 24 Tahun 2007 Tentang Penanggulangan Bencana, 2007). Ada banyak kerugian yang dapat disebabkan oleh bencana alam, mulai dari kerugian secara fisik hingga sosial (Afida et al., 2020). Allah berfirman pada QS. Al-Baqarah ayat 155 – 157 sebagai berikut.

وَلَنَبْلُوَنَّكُمْ بِشَيْءٍ مِّنَ الْخَوْفِ وَالْجُوعِ وَنَقْصٍ مِّنَ الْأَمْوَالِ وَالْأَنْفُسِ وَالثَّمَرَاتِ ۗ وَبَشِّرِ الصَّابِرِينَ ۝ الَّذِينَ إِذَا أَصَابَتْهُمُ مُصِيبَةٌ قَالُوا إِنَّا لِلَّهِ وَإِنَّا إِلَيْهِ رَاجِعُونَ ۝ أُولَٰئِكَ عَلَيْهِمْ صَلَوَاتٌ مِّن رَّبِّهِمْ وَرَحْمَةٌ وَأُولَٰئِكَ هُمُ الْمُتَّقُونَ

*“Dan sungguh akan Kami berikan cobaan kepadamu, dengan sedikit ketakutan, kelaparan, kekurangan harta, jiwa dan buah-buahan. Dan berikanlah berita gembira kepada orang-orang yang sabar. (yaitu) orang-orang yang apabila ditimpa musibah, mereka mengucapkan: "Inna lillaahi wa innaa ilaihi raaji'uun (sesungguhnya kami milik Allah dan sesungguhnya kami sedang menuju kemabali kepada-Nya). Mereka itulah yang mendapat keberkatan yang sempurna dan rahmat dari Tuhan mereka dan mereka itulah orang-orang yang mendapat petunjuk.” (QS. Al-Baqarah : 155 – 157).*

Berdasarkan buku “Tafsir Al-Mishbah”, Surah Al-Baqarah ayat 155 – 157 menjelaskan sesungguhnya hakikat kehidupan dunia ditandai dengan salah satunya yakni terdapat berbagai macam cobaan atau ujian (Shihab, 2009). Cobaan atau ujian yang diberikan Allah tersebut pada hakikatnya sedikit, karena cobaan masih bisa terjadi dalam bentuk yang lebih besar, juga apabila dibandingkan dengan ganjaran dan imbalan yang nantinya akan diterima. Berbagai bentuk cobaan atau ujian tersebut sesungguhnya merupakan nikmat yang besar, karena kita dapat mempersiapkan diri dan naik ke tingkat yang lebih baik. Kita diperintahkan untuk bersabar, karena seluruh apa yang terdapat di dunia ini hanyalah milik Allah semata dan segala cobaan atau ujian tersebut pastilah ada hikmah dibaliknya. Orang-orang

yang selalu bersabar ketika menghadapi cobaan atau ujian yang diberikan akan mendapatkan rahmat dan petunjuk, tidak hanya untuk mengatasi kesulitan, tetapi juga untuk kebahagiaan duniawi dan ukhrawi.

Bencana sebagai cobaan maupun ujian sesungguhnya mengajarkan manusia agar dapat mengambil hikmah dari segala peristiwa yang terjadi. Hikmah yang dapat diambil antara lain terhadap Allah, manusia, dan alam. Hal tersebut akan dapat meningkatkan derajat manusia di mata Allah juga meningkatkan kualitas hidup dengan ber-tasamuh (berbuat baik) pada sesama. Tindakan penanganan setelah terjadinya bencana alam sendiri harus diupayakan tidak hanya untuk hal-hal fisik seperti bangunan maupun kondisi alam dan lingkungan, tetapi juga terhadap hal-hal nonfisik seperti kondisi mental dan keimanan (Hakim, 2013).

Salah satu kerugian paling fatal yang dapat disebabkan oleh bencana alam adalah kerusakan dari sektor bangunan. Pada umumnya, penentuan tingkat kerusakan bangunan setelah terjadinya bencana dilakukan oleh tim surveyor yang terdiri dari tim SAR, tim BPBD, Tim P3B, atau relawan yang membantu melakukan *assessment* penentuan tingkat kerusakan sektor pasca bencana alam (Bachriwindi et al., 2019). Akan tetapi, seringkali terjadi ketidakakuratan data dengan yang terjadi di lapangan, akibat penilaian subjektif yang menyebabkan pengkategorian data menjadi berbeda (Safitri et al., 2022). Hal tersebut dapat menyebabkan data yang dilaporkan dan dimasukkan ke Badan Penanggulangan Bencana Daerah (BPBD) merupakan data yang tidak sesuai dan akan menghambat proses rehabilitasi dan rekonstruksi pasca terjadinya bencana yang dilakukan oleh pemerintah.

Oleh sebab itu, diperlukan sebuah sistem pendukung keputusan atau *decision support system* yang bisa digunakan dalam membantu tim surveyor untuk menentukan tingkat kerusakan bangunan pasca terjadinya bencana alam. Penilaian terhadap tingkat kerusakan sektor sendiri berguna dalam membantu pemerintah untuk melakukan rehabilitasi dan rekonstruksi pasca terjadinya bencana alam (Putra et al., 2020). *Decision Support System* (DSS) atau yang dikenal juga sebagai Sistem Pendukung Keputusan (SPK) sendiri merupakan suatu sistem informasi komputer interaktif yang berguna bagi para pengguna tertentu sebagai penentuan keputusan, yang mengaplikasikan berbagai model keputusan, *database*, serta pemikiran manajer sendiri (Surahman & Nursadi, 2019). Pada penelitian ini, DSS yang dibangun mengimplementasikan metode TOPSIS, dimana metode TOPSIS termasuk salah satu dari metode dasar *Multiple Criteria Decision Making* (MCDM) yang banyak diimplementasikan karena kesederhanaan dan kemudahan penerapannya bahkan untuk kasus yang memiliki banyak kriteria dan alternatif sekalipun, serta konsep yang mendasarinya, dimana solusi terbaik merupakan solusi yang berposisi paling dekat dari solusi ideal positif juga berposisi paling jauh dari solusi ideal negatif (Çelikkilek & Tüysüz, 2020).

Sistem yang dibangun pada penelitian ini berbasis *machine learning*, dimana *machine learning* sendiri merupakan teknologi yang sangat berguna dalam sistem pendukung keputusan (Merkert et al., 2015). Metode *machine learning* yang digunakan yakni *Artificial Neural Networks* (ANN) atau biasa disebut sebagai Jaringan Saraf Tiruan (JST). Sesuai namanya, *neural networks* merupakan sebuah metode yang terinspirasi dari cara kerja bagian tubuh manusia, yakni otak dan

jaringan saraf, dan merupakan sekelompok saraf tiruan yang saling berhubungan untuk komputasi pemrosesan informasi menggunakan model matematika dan/atau komputasi (Dhanaseely et al., 2012). *Neural networks* digunakan pada penelitian ini, karena metode ini termasuk salah satu metode matematika-komputasi modern, yang dipakai dalam pemecahan masalah yang bersifat dinamis seiring berjalannya waktu (Khaze et al., 2013). Perhitungan yang digunakan dalam *machine learning* yang dibangun adalah *forward propagation*. *Forward propagation* sendiri termasuk salah satu algoritma yang bisa digunakan untuk perhitungan *neural networks*. Seperti namanya, *forward propagation* atau propagasi maju mengumpan data *input* ke arah depan melalui jaringan. Berdasarkan penelitian yang dilakukan, disebutkan bahwa ada kalanya waktu komputasi *forward propagation* lebih pendek dibandingkan dengan *backward propagation* (Hirasawa et al., 1996).

Beberapa penelitian terdahulu yang sudah pernah dilakukan sebelumnya, salah satunya merupakan penelitian (A. T. Almais et al., 2020), yang pada penelitiannya menyebutkan bahwa persiapan tindakan rehabilitasi dan rekonstruksi setelah terjadinya bencana dilakukan guna mengetahui besar tingkat kerusakan juga kerugian yang perlu untuk ditindaklanjuti oleh pemerintah. Pada penelitian ini, diimplementasikan *Decision Support System Dynamic* (DSSD) menggunakan metode *Fuzzy-Weighted Product* (FWP) yang menghasilkan tiga tipe data yang berbeda, dan disimpulkan bahwa metode FWP dapat diterapkan dalam DSSD guna membantu tim surveyor pada kegiatan mempersiapkan tindakan rehabilitasi juga rekonstruksi pascabencana. Selain itu, pada penelitian yang dilakukan (Safitri et al., 2022), diusulkan solusi berupa pengujian optimasi *query* dalam metode TOPSIS



yang diterapkan pada *decision support system* dalam menentukan tingkat kerusakan sektor pasca bencana alam. Pada penelitian ini dilakukan tiga kali percobaan dengan jumlah data yang berbeda dan didapatkan hasil bahwa tahapan perankingan dengan optimasi *query* memiliki *response time* yang lebih cepat.

## 1.2 Pernyataan Masalah

Latar belakang yang telah dijabarkan sebelumnya merupakan dasar dari pernyataan masalah yang ditarik sebagai berikut.

- a. Bagaimana dalam menentukan tingkat kerusakan sektor setelah terjadinya bencana alam menggunakan metode TOPSIS berbasis *machine learning*?
- b. Berapa besar akurasi dalam penentuan tingkat kerusakan sektor setelah terjadinya bencana alam menggunakan metode TOPSIS berbasis *machine learning*?

## 1.3 Tujuan Penelitian

Beberapa tujuan yang ingin dicapai pada penelitian yang dilakukan ini antara lain sebagai berikut.

- a. Menentukan tingkat kerusakan sektor setelah terjadinya bencana alam menggunakan metode TOPSIS yang berbasis *machine learning*.
- b. Mengetahui dan mengukur besar akurasi dalam menentukan tingkat kerusakan sektor setelah terjadinya bencana alam dengan metode TOPSIS berbasis *machine learning*.

#### **1.4 Batasan Masalah**

Dalam menjadikan penelitian yang dilakukan tetap terfokus dan terarah, terdapat batasan-batasan masalah yang diterapkan sebagai berikut.

- a. Penentuan keputusan yang dilakukan hanya untuk mengukur tingkat kerusakan sektor bangunan setelah terjadi bencana alam.
- b. Data yang digunakan merupakan data kejadian bencana alam BPBD Kota Malang.

#### **1.5 Manfaat Penelitian**

Harapan dari penelitian yang dilakukan yakni dapat memberikan manfaat dan maslahat di kemudian hari, antara lain sebagai berikut.

- a. Membantu mengefisiensi waktu dalam menentukan tingkat kerusakan sektor bangunan setelah terjadi bencana alam.
- b. Membantu mempercepat tindakan proses rehabilitasi dan rekonstruksi setelah terjadinya bencana alam.
- c. Membantu tim surveyor dalam menentukan tingkat kerusakan sektor setelah terjadinya bencana alam.

## **BAB II**

### **STUDI PUSTAKA**

#### **2.1 Penelitian Terdahulu**

Penelitian terdahulu menyebutkan bahwa persiapan tindakan rehabilitasi dan rekonstruksi setelah terjadinya bencana dilakukan dengan tujuan yakni mengetahui besar tingkat kerusakan dan kerugian yang perlu untuk ditindaklanjuti oleh pemerintah (A. T. W. Almais et al., 2020). Pada penelitian ini, dilakukan pengimplementasian *Decision Support System Dynamic* (DSSD) yang berfungsi untuk mencegah terjadinya kesalahan data mengenai tingkat kerusakan dan kerugian pascabencana yang dialami. DSSD yang merupakan pengembangan model *decision support system* yang baru, pada penelitian ini diimplementasikan dengan metode Fuzzy-Weighted Product (FWP). Hasil yang didapatkan berupa tiga tipe data berbeda, yakni 73% data uji yang sama dengan data pola, 22% data uji yang tidak sama dengan data pola, serta 5% data uji yang tidak bisa digunakan dalam pengujian. Berdasarkan hasil yang didapatkan tersebut, kesimpulan yang diambil yakni bahwa metode FWP dapat diterapkan dalam DSSD yang berfungsi dalam membantu tim surveyor untuk mempersiapkan tindakan rehabilitasi dan rekonstruksi pascabencana.

Berdasarkan penelitian yang pernah dilakukan (Safitri et al., 2022), diusulkan solusi berupa pengujian optimasi *query* dalam metode TOPSIS yang diterapkan pada sistem pendukung keputusan dalam menentukan tingkat kerusakan sektor setelah terjadinya bencana alam. Hal tersebut dilatarbelakangi oleh jumlah

data yang diperoleh tim surveyor Perencanaan dan Pengendalian Penanganan Bencana (P3B) berjumlah sangat besar, yang menyebabkan dibutuhkan sumber daya sistem yang besar serta waktu pemrosesan yang lama. Pada penelitian ini dilakukan tiga kali percobaan dengan jumlah data yang berbeda, yang berdasarkan percobaan tersebut didapatkan hasil bahwa tahapan perankingan dengan optimasi *query* membutuhkan waktu atau *response time* yang lebih cepat dibandingkan dengan non-optimasi *query*.

Pada penelitiannya (Chamid, 2016), mengimplementasikan metode TOPSIS pada sistem pendukung keputusan dalam penentuan prioritas kondisi rumah tidak sehat. Hasil perhitungan metode TOPSIS yang diterapkan digunakan sebagai masukan untuk dinas kesehatan terkait dalam melakukan tindakan terhadap prioritas rumah tidak sehat. Penelitian ini menggunakan tiga kriteria dalam penentuan kondisi rumah tidak sehat, yakni komponen rumah, sarana rumah, dan perilaku. Berdasarkan hasil perhitungan yang dilakukan, diketahui dari sepuluh sampel yang digunakan, didapatkan delapan rumah dengan kondisi tidak sehat.

Penelitian lainnya menggunakan metode TOPSIS yang diterapkan dalam sistem pendukung keputusan pada penentuan kenaikan gaji karyawan berdasarkan penilaian kinerja (Surahman & Nursadi, 2019). Kriteria penentuan keputusan yang digunakan pada penelitian ini antara lain penilaian tanggung jawab, kerjasama, masa kerja, loyalitas, dan presensi. Pada penelitian ini, digunakan metode TOPSIS yang terdapat bobot untuk setiap kriteria, dalam menilai kinerja karyawan yang selama ini belum optimal. Metode TOPSIS yang digunakan pada penelitian ini dilakukan dengan menghitung nilai kedekatan relatif pada masing-masing

alternatif. Nilai kedekatan relatif yang didapatkan kemudian digunakan sebagai pertimbangan dalam menentukan kenaikan gaji maupun jabatan untuk para karyawan. Bobot yang terdapat pada setiap kriteria yang digunakan diubah nilainya untuk mengetahui kriteria yang paling berpengaruh sebagai pengujian hasil.

Pada penelitian yang dilakukan (Wang et al., 2020), diusulkan model jaringan saraf tiruan dengan berbasis algoritma *forward propagation* untuk meningkatkan kinerja proses deteksi kapal. Sebelumnya, banyak metode deteksi yang telah diusulkan untuk pendeteksian kapal, tetapi umumnya metode-metode tersebut hanya berfokus pada peningkatan akurasi dan jarang memperhatikan kecepatan deteksi. Model yang diusulkan pada penelitian ini sendiri merupakan *hybrid learning model*, dimana model jaringan saraf tiruan yang dibuat dapat secara efektif mengurangi jumlah parameter dan meningkatkan kecepatan deteksi tanpa menurunkan tingkat akurasi. Algoritma *forward propagation* yang diusulkan dapat menghapus kernel konvolusi yang berlebihan, sehingga dapat meningkatkan akurasi deteksi dan mempercepat proses deteksi secara signifikan.

Penelitian terkait lainnya mengusulkan pendekatan baru untuk pemantauan multi-urutan pada *streaming time-series* yang terinspirasi dari algoritma *forward propagation* pada jaringan saraf tiruan (Gong et al., 2018). Algoritma *forward propagation* diimplementasikan untuk memotong perhitungan yang tidak diperlukan. Pendekatan yang diusulkan digunakan untuk menghitung sebagian kecil dari matriks, dengan hanya menda data perhitungan yang dibutuhkan menggunakan algoritma *forward propagation*. Hal tersebut menyebabkan pendekatan yang dilakukan dapat mengurangi waktu eksekusi sistem secara efektif.

Dalam penelitiannya (Mathew et al., 2020), diusulkan algoritma *forward propagation*, yang digunakan ketika memperbarui parameter *Recurrent Neural Networks* (RNN) dengan mengoptimalkan fungsi risiko. Keuntungan menggunakan algoritma *forward propagation* dalam menangkap momen statistik *Quantities of Interest* (QoI) di seluruh domain diamati dengan menggunakan masalah elastis linier yang dipilih. Keuntungan kedua yakni perluasan dari *Uncertainty Quantification* (UQ) yang diusulkan untuk memperhitungkan *multiple independent random fields*, diikuti oleh analisis sensitivitas berbasis *Polynomial Chaos Expansion* (PCE). Pada penelitian ini, *forward propagation* dari *multiple independent random fields* diusulkan menggunakan kecocokan permukaan polinomial.

Berdasarkan penelitian yang dilakukan (Kag & Saligrama, 2021), diusulkan algoritma *forward propagation* dalam memperbarui parameter *Recurrent Neural Networks* (RNN) dengan mengoptimalkan faktor risiko. Risiko yang diusulkan pada penelitian ini berupa regulasi pada waktu yang berkembang secara dinamis dan memungkinkan pembaruan parameter RNN untuk konvergen ke solusi stasioner dari tujuan RNN. Pada penelitian ini, disimpulkan bahwa secara empiris *forward propagation* mengungguli *back propagation* pada banyak tugas *benchmark*, sehingga memungkinkan untuk memecahkan masalah dependensi jarak jauh. *Back propagation* sendiri merupakan algoritma yang banyak digunakan dalam pelatihan RNN, dimana *back propagation* memperbarui parameter RNN yang menyebabkan kemampuan latih menjadi buruk.

## 2.2 Bencana Alam

Bencana alam merupakan peristiwa atau serangkaian peristiwa yang dapat menimbulkan kerugian dalam kehidupan masyarakat yang ditimbulkan karena faktor alam (Undang-Undang Republik Indonesia Nomor 24 Tahun 2007 Tentang Penanggulangan Bencana, 2007). Adapun kerugian yang bisa disebabkan oleh bencana alam antara lain seperti adanya korban luka maupun jiwa, kerugian secara psikologis, kerugian secara materil, hingga kerusakan yang terjadi di berbagai sektor. Kerugian akibat bencana alam sendiri dibagi menjadi dua, yakni dampak secara langsung yang berupa kerusakan fisik, serta dampak secara tidak langsung yang merupakan kerugian akibat kerusakan fisik yang menimbulkan kerugian berkelanjutan yang dapat berupa kerusakan lingkungan dan kerugian secara materil (Putra et al., 2020). Faktor penyebab yang dapat menyebabkan terjadinya bencana alam antara lain faktor alam dan faktor manusia, kerentanan yang tinggi, serta kapasitas yang rendah dari komponen-komponen yang ada di masyarakat (Sushanti et al., 2020).

## 2.3 Sistem Pendukung Keputusan

*Decision Support System* (DSS) atau yang juga biasa disebut sebagai Sistem Pendukung Keputusan (SPK) merupakan sebuah sistem informasi komputer interaktif yang berguna bagi para pengguna tertentu sebagai penentuan keputusan, yang mengaplikasikan berbagai model keputusan, *database*, serta pemikiran manajer sendiri (Surahman & Nursadi, 2019). Sistem Pendukung Keputusan dibuat dengan menggabungkan informasi dan pengetahuan yang dimiliki oleh seseorang

dengan kemampuan komputer yang berfungsi untuk menghasilkan penentuan keputusan yang lebih berkualitas. Sistem Pendukung Keputusan muncul berdasarkan gagasan penggunaan komputer sebagai pendukung bagi para pembuat keputusan (Merkert et al., 2015). Hasil keputusan yang diambil pada umumnya ditentukan berdasarkan beberapa kriteria seperti alternatif dan tujuan yang ingin dicapai (Chamid, 2016).

#### 2.4 Metode ROC

Metode ROC, yang merupakan singkatan dari *Rank Order Centroid*, merupakan salah satu metode yang dapat digunakan untuk menentukan nilai bobot untuk setiap kriteria yang digunakan (Mesran et al., 2019). Metode ROC menitikberatkan kriteria yang menjadi prioritas utama dalam penentuan bobot, dimana kriteria 1 memiliki prioritas lebih tinggi daripada kriteria 2 dan kriteria 2 memiliki prioritas lebih tinggi daripada kriteria 3, dan seterusnya. Hal tersebut ditunjukkan oleh persamaan (2.1) dan (2.2), dimana kriteria dinotasikan sebagai  $K$  dan bobot dinotasikan sebagai  $W$ .

$$K_1 \geq K_2 \geq K_3 \geq \dots \geq K_n \dots (2.1)$$

sehingga,

$$W_1 \geq W_2 \geq W_3 \geq \dots \geq W_n \dots (2.2)$$

Rumus perhitungan untuk menentukan bobot menggunakan metode ROC sendiri ditunjukkan oleh persamaan (2.3) (Damanik & Utomo, 2020).

$$W_i = \frac{1}{i} \sum_{j=1}^i \left(\frac{1}{j}\right) \dots (2.3)$$



dimana,

$W_i$  = Normalisasi rasio perkiraan skala bobot tujuan

$j$  = Total jumlah tujuan

$i$  = Ranking dari  $j$  tujuan

## 2.5 Metode TOPSIS

Metode *Technique for Order Preference by Similarity to Ideal Solution*, atau yang biasa disebut sebagai metode TOPSIS, adalah salah satu dari sekian metode yang bisa diterapkan pada sebuah sistem pendukung keputusan. Metode TOPSIS termasuk salah satu metode dasar dari *Multiple Criteria Decision Making*, atau yang biasa disingkat menjadi MCDM, yang banyak diimplementasikan karena kesederhanaan dan kemudahan penerapannya bahkan untuk kasus yang memiliki banyak kriteria dan alternatif sekalipun, serta konsep yang mendasarinya, dimana solusi terbaik merupakan solusi yang berposisi paling dekat dari solusi ideal positif juga berposisi paling jauh dari solusi ideal negatif (Çelikkbilek & Tüysüz, 2020). Metode TOPSIS telah banyak digunakan dalam berbagai bidang sebagai metode penentuan keputusan, seperti dalam perbandingan kinerja perusahaan, penilaian kualitas layanan, sistem manufaktur, dan sebagainya (Ren et al., 2007). Bahkan menurut penelitian yang dilakukan, metode TOPSIS merupakan metode MCDM kedua yang paling banyak digunakan, setelah metode *Analytic Hierarchy Process* atau metode AHP (Zyoud & Fuchs-Hanusch, 2017).

Metode TOPSIS sendiri dapat dijabarkan menjadi beberapa tahapan perhitungan sebagai berikut (Chakraborty, 2022).

- a. Menghitung matriks keputusan yang ternormalisasi.

*Rating* kinerja atau yang juga dikenal sebagai matriks keputusan merupakan preferensi untuk setiap alternatif terhadap kriteria. Normalisasi dilakukan pada matriks keputusan awal (matriks  $X$ ) untuk menjadikan perbandingan antar kriteria menjadi lebih mudah. Matriks yang ternormalisasi (matriks  $Y$ ) dapat dihitung menggunakan persamaan (2.4).

$$y_{ij} = \frac{x_{ij}}{\sqrt{\sum_{i=1}^I x_{ij}^2}} \quad \dots (2.4)$$

- b. Mengintegrasikan bobot dengan *rating*.

Matriks keputusan terbobot dan ternormalisasi  $v_{ij}$  dihitung menggunakan persamaan (2.5) yang kemudian membentuk matriks  $V$ , dimana  $W$  menunjukkan atribut bobot.

$$v_{ij} = W_j * y_{ij}; (i = 1, 2, \dots, I; j = 1, 2, \dots, J) \quad \dots (2.5)$$

- c. Menemukan solusi ideal positif dan solusi ideal negatif.

Solusi ideal positif dinotasikan dalam  $A^+$  dan solusi ideal negatif dinotasikan dalam  $A^-$  yang bisa didapatkan melalui persamaan (2.6) dan (2.7).

$$A^+ = [v_1^+, v_2^+, \dots, v_j^+] \quad \dots (2.6)$$

$$A^- = [v_1^-, v_2^-, \dots, v_j^-] \quad \dots (2.7)$$

dimana,

$$v_j^+ = \begin{cases} \max v_{ij}, & \text{jika } j \text{ merupakan atribut benefit} \\ \min v_{ij}, & \text{jika } j \text{ merupakan atribut cost} \end{cases}$$

$$v_j^- = \begin{cases} \min v_{ij}, & \text{jika } j \text{ merupakan atribut benefit} \\ \max v_{ij}, & \text{jika } j \text{ merupakan atribut cost} \end{cases}$$

- d. Memperoleh nilai pemisahan.

Nilai pemisahan merupakan jarak setiap *rating* alternatif dari solusi ideal positif dan negatif, yang bisa didapatkan melalui persamaan (2.8) untuk nilai pemisahan positif dan persamaan (2.9) untuk nilai pemisahan negatif.

$$S_i^+ = \sqrt{\sum_{j=1}^J (v_{ij} - v_j^+)^2} \dots (2.8)$$

$$S_i^- = \sqrt{\sum_{j=1}^J (v_{ij} - v_j^-)^2} \dots (2.9)$$

- e. Menghitung keseluruhan nilai preferensi.

Keseluruhan nilai preferensi yang dinotasikan dalam  $V_i$  untuk setiap alternatif dihitung menggunakan persamaan (2.10), dimana alternatif akan di-*rating* berdasarkan nilai  $V_i$  yang lebih tinggi.

$$V_i = \frac{S_i^-}{S_i^- + S_i^+} \dots (2.10)$$

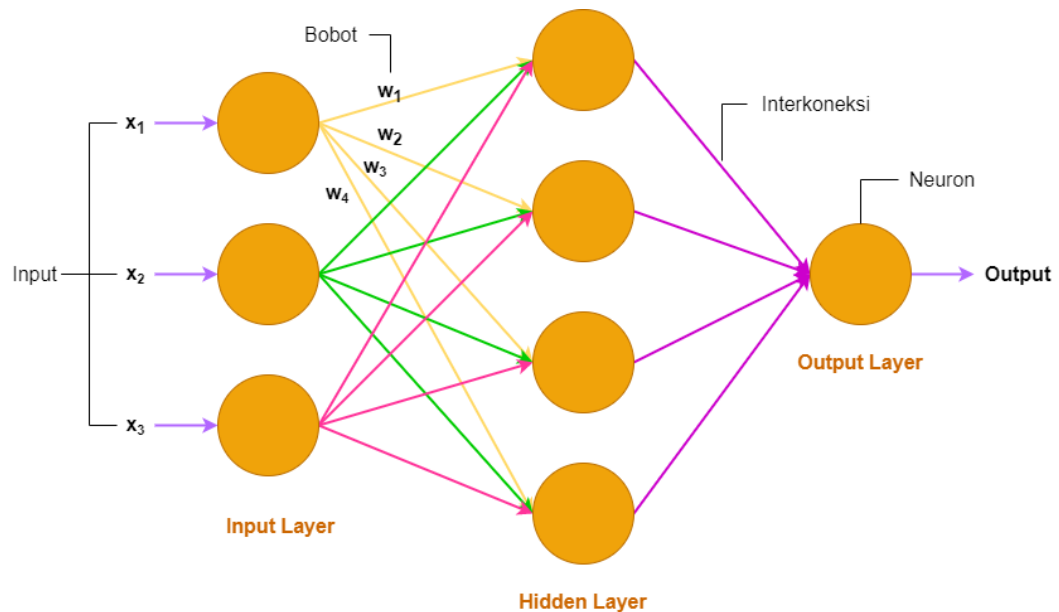
## 2.6 Machine Learning dan Forward Propagation in Neural Network

*Machine Learning* (ML) adalah salah satu bagian atau cabang dari kecerdasan buatan atau *Artificial Intelligence* (AI) yang berguna untuk melakukan prediksi dan inferensi dengan menghasilkan model matematika menggunakan pembelajaran data (Permadi, 2020). *Machine learning* merupakan teknologi yang sangat berguna dalam sistem pendukung keputusan (Merkert et al., 2015). Teknologi *machine learning* sendiri terdiri dari mengenali dan merumuskan suatu masalah atau situasi yang membutuhkan suatu keputusan. Hal tersebut menjadi faktor bahwa teknologi *machine learning* dapat mendukung berbagai tahapan yang

ada pada sistem pendukung keputusan. Model *machine learning* dapat membantu mengurangi pekerjaan yang dilakukan dan meningkatkan efisiensi sistem pada sistem pendukung keputusan (Kumar, 2020).

*Artificial Neural Networks* (ANN) atau yang biasa disebut sebagai Jaringan Saraf Tiruan (JST), sesuai namanya, merupakan sebuah metode yang terinspirasi dari cara kerja bagian tubuh manusia, yakni otak dan jaringan saraf (Sazli, 2006). ANN atau yang biasa disebut sebagai *Neural Network* saja, juga merupakan sekelompok saraf tiruan yang saling berhubungan untuk komputasi pemrosesan informasi menggunakan model matematika dan/atau komputasi (Dhanaseely et al., 2012). *Neural networks* sendiri dapat diimplementasikan ke berbagai bidang meliputi identifikasi dan kontrol sistem, pengambilan keputusan, pola pengenalan, diagnosis medis, aplikasi keuangan, penambangan data, penyaringan spam email, dan sebagainya.

Berdasarkan penelitian yang pernah dilakukan (Khaze et al., 2013), ditunjukkan bahwa *neural networks* merupakan salah satu metode yang dipandang sebagai metode matematika-komputasi modern, yang diterapkan dalam pemecahan masalah dinamis seiring berjalannya waktu. *Neural networks* juga mampu untuk menyelesaikan berbagai masalah yang tidak bisa diselesaikan oleh metode komputasi serta statistik, dengan cara yakni mengenali pola dari data lain. Model sistem dari *neural networks* sendiri dapat dibangun menggunakan data-data yang disediakan (Wee et al., 2010).



Gambar 2. 1 Arsitektur *Neural Networks*

Gambar 2.1 menunjukkan arsitektur dari metode *neural networks*. Pada gambar tersebut dapat dilihat bahwa *neural networks* berupa kumpulan unit atau *node* yang disebut sebagai *artificial neuron*. Neuron yang ada pada *neural networks* tersusun dalam layer-layer secara berurutan. Dalam *neural networks* sendiri terdapat 3 jenis *layer*, yakni *input layer*, *hidden layer*, dan *output layer*. Setiap neuron yang berada di *layer* yang sama menggunakan fungsi aktivasi yang sama dan tidak mengirimkan informasi satu sama lain. Semua informasi masuk melalui *input layer* dan keluar melalui *output layer*. Setiap interkoneksi yang ada pada *neural networks* memiliki bobot, sedangkan neuron memiliki bias.

*Forward propagation* sendiri termasuk salah satu algoritma yang bisa digunakan untuk perhitungan *neural networks*. Seperti namanya, *forward propagation* atau propagasi maju mengumpan data *input* ke arah depan melalui jaringan. Berdasarkan penelitian yang dilakukan, disebutkan bahwa ada kalanya

waktu komputasi *forward propagation* lebih pendek dibandingkan dengan *backward propagation* (Hirasawa et al., 1996). Pada *forward propagation*, dalam menghasilkan *output*, data *input* harus diumpangkan ke arah depan saja. Data tidak boleh mengalir dalam arah terbalik selama pemrosesan *output*, karena hal tersebut membuat siklus tidak akan terbentuk dan *output* tidak akan pernah bisa dihasilkan.

Berdasarkan penelitian yang dilakukan (Wang et al., 2020), terdapat dua langkah dalam algoritma *forward propagation*. Langkah pertama yakni menganalisis distribusi bobot parameter pada lapisan konvolusi, yang akan membantu menemukan parameter robot redundan. Parameter bobot yang berlebihan nantinya akan dihapus pada langkah kedua, menggunakan parameter pemangkasan. Algoritma *forward propagation* dapat menghapus kernel konvolusi yang berlebihan, sehingga dapat meningkatkan akurasi deteksi dan mempercepat proses pendeteksian secara signifikan.

Berikut merupakan rumus perhitungan dari metode *forward propagation in neural networks* (Deore, 2021).

a. Langkah 1

Langkah pertama dalam perhitungan *forward propagation in neural networks* adalah menjumlahkan perkalian bobot dengan *input*, kemudian menjumlahkannya dengan nilai bias. Perhitungan langkah 1 dapat dilihat pada rumus (2.11).

$$y = w_1x_1 + w_2x_2 + \dots + w_ix_i + b$$

$$y = \sum_{i=1}^n \{(w_i * x_i) + b\} \dots (2.11)$$

b. Langkah 2

Langkah kedua yang dilakukan adalah mengaktivasi nilai  $y$  menggunakan fungsi aktivasi untuk menghasilkan nilai *output* yang dinotasikan sebagai  $z$ . Rumus perhitungan langkah 2 ditunjukkan oleh rumus (2.12).

$$z = y_{pred} = \text{Activation}(y)$$

$$z = \frac{1}{1 + e^{-y}} \dots (2.12)$$

dimana,

$$y = w_1x_1 + w_2x_2 + \dots + w_ix_i + b$$

## 2.7 MSE, MAPE, dan Akurasi

Proses *training* dan *testing* yang dilakukan pada prediksi akan menghasilkan nilai MSE atau yang biasa disebut *Mean Squared Error* (A. T. W. Almais et al., 2022). MSE sendiri merupakan jumlah *squared error* dibagi dengan banyaknya data. Rumus dari MSE ditunjukkan pada rumus (2.13).

$$MSE = \frac{\sum_{i=1}^n (x_i - y_i)^2}{n} \dots (2.13)$$

Selain MSE, nilai yang bisa didapatkan setelah melakukan pengujian pada prediksi adalah nilai MAPE atau *Mean Absolute Percentage Error*. MAPE merupakan error mutlak rata-rata dibagi dengan jumlah banyaknya data prediksi. Rumus perhitungan MAPE ditunjukkan pada rumus (2.14).

$$MAPE = \frac{\sum_{i=1}^n \frac{|x_i - y_i|}{x_i}}{n} \times 100\% \dots (2.14)$$

Perhitungan atau uji akurasi dilakukan dengan tujuan untuk menghitung tingkat keberhasilan hasil prediksi dengan kejadian yang sebenarnya. Akurasi merupakan cara untuk melihat tingkat kesesuaian dalam perhitungan yang dilakukan dalam melakukan prediksi (Sari et al., 2018). Untuk menghitung nilai akurasi, nilai MAPE dapat digunakan dimana 100% (nilai akurasi sempurna) dikurangi dengan nilai MAPE, yang dapat dilihat pada rumus (2.15) sebagai berikut.

$$Akurasi = 100\% - MAPE \dots (2.15)$$



**BAB III**  
**DESAIN DAN IMPLEMENTASI**

Pembahasan pada bab ini dibagi menjadi 5 subbab yaitu *data collection*, desain sistem, *research instrument*, *conceptual framework*, dan hipotesis.

**3.1 Data Collection**

Data yang digunakan dalam penentuan kerusakan sektor pasca terjadinya bencana alam adalah data sekunder, yang merupakan data yang diperoleh dan berasal dari lembaga atau peneliti lain. Penelitian ini menggunakan data kejadian bencana alam yang dirilis Badan Penanggulangan Bencana Daerah (BPBD) Kota Malang. Data tersebut sebagai contohnya ditunjukkan pada tabel 3.1.

Tabel 3. 1 Data Kejadian Bencana Alam

No	Jenis Bencana	Lokasi Kejadian	Kerusakan/ Kehilangan		Keterangan
			Kondisi Kerusakan	Jenis Sektor	
1	Tanah Longsor	Desa Patutkumpul Kecamatan Turi Kabupaten Bojonegoro	Rumah tergenang	Perumahan	Tetap berdiri, Dibeberapa bagian ada kerusakan ringan, rusak <30%, Tidak berbahaya, dibeberapa

No	Jenis Bencana	Lokasi Kejadian	Kerusakan/ Kehilangan		Keterangan
			Kondisi Kerusakan	Jenis Sektor	
					bagian mengalami kerusakan ringan
2	Tanah Longsor	Desa Pomahanjangan Kecamatan Turi Kabupaten Bojonegoro	Rumah tergenang	Perumahan	Tetap berdiri, Dibeberapa bagian ada kerusakan ringan, rusak <30%, Tidak berbahaya, dibeberapa bagian mengalami kerusakan ringan
3	Tanah Longsor	Desa Somosari Kecamatan Kali Tengah	Rumah ukuran 5x10x2.5 m ambles 60 cm	Perumahan	Condong, Rusak sebagian, rusak 30-50

No	Jenis Bencana	Lokasi Kejadian	Kerusakan/ Kehilangan		Keterangan
			Kondisi Kerusakan	Jenis Sektor	
		Kabupaten Bojonegoro			%, Relatif berbahaya, rusak sebagian
4	Tanah Longsor	Desa Pomahanjangan Kecamatan Turi Kabupaten Lamongan	TK tergenang	Sosial	Tetap berdiri, Dibeberapa bagian ada kerusakan ringan, rusak <30%, Tidak berbahaya, dibeberapa bagian mengalami kerusakan ringan
5	Angin Puuting Beliung	Desa pojok Kecamatan Campurdarat	rumah mengalami kerusakan pd bagian atap	Perumahan	Tetap berdiri, Dibeberapa bagian ada kerusakan

No	Jenis Bencana	Lokasi Kejadian	Kerusakan/ Kehilangan		Keterangan
			Kondisi Kerusakan	Jenis Sektor	
		Kabupaten Tulungagung			ringan, rusak <30%, Tidak berbahaya, dibebepa bagian mengalami kerusakan ringan
6	Badai	Desa Karang Kecamatan Bareng Kabupaten Jombang	Rumah roboh	Perumahan	Benar-benar runtuh, rusak total, rusak >50 %,Sangat berbahaya, benar-benar rusak
7	Tanah Longsor	Desa Baosankidul Kecamatan Ngrayun	rumah rusak bagian tembok retak	Perumahan	Condong, Rusak sebagian, rusak 30-50

No	Jenis Bencana	Lokasi Kejadian	Kerusakan/ Kehilangan		Keterangan
			Kondisi Kerusakan	Jenis Sektor	
		Kabupaten Ponorogo			%, Relatif berbahaya, rusak sebagian
8	Tanah Longsor	Desa Pronojiwo Kecamatan Pronojiwo Kabupaten Ponorogo	Plengsengan penahan tebing T= 5 m longsor dan menimpa dinding belakang rumah warga	Infrastruktur	Benar-benar runtuh, rusak total, rusak >50 %,Sangat berbahaya, benar-benar rusak
9	Banjir Bandang	Desa Kasiman Kecamatan Kasiman Kabupaten Bojonegoro	Jembatan Jalan PU putus	Infrastruktur	Benar-benar runtuh, rusak total, rusak >50 %,Sangat berbahaya,

No	Jenis Bencana	Lokasi Kejadian	Kerusakan/ Kehilangan		Keterangan
			Kondisi Kerusakan	Jenis Sektor	
					benar-benar rusak
10	Banjir	Dusun Perak Kecamatan Perak Kabupaten Jombang	SDN tergenang	Sosial	Tetap berdiri, Dibeberapa bagian ada kerusakan ringan, rusak <30%, Tidak berbahaya, dibeberapa bagian mengalami kerusakan ringan

Data tersebut merupakan data hasil penentuan tim surveyor pada penentuan tingkat kerusakan sektor setelah terjadinya bencana alam. Ada beberapa kriteria yang digunakan sebagai acuan penilaian dan penentuan pada data tersebut. Kriteria-kriteria tersebut akan digunakan sebagai pendukung pada sistem yang dirancang, yang bertujuan untuk membantu tim surveyor dalam penentuan tingkat kerusakan

sektor. Alternatif dan kriteria yang digunakan dijabarkan pada tabel 3.2 serta tabel 3.3 sebagai berikut.

Tabel 3. 2 Alternatif (Safitri et al., 2022)

Kode Alternatif	Nama Alternatif
A1	Rusak Ringan
A2	Rusak Sedang
A3	Rusak Berat

Tabel 3.2 menunjukkan alternatif yang digunakan dalam penelitian ini. Ada tiga alternatif yang digunakan, yakni tiga tingkat kerusakan yang antara lain rusak ringan sebagai A1, rusak sedang sebagai A2, serta rusak berat sebagai A3. Penentuan alternatif yang berupa tiga tingkat jenis kerusakan, mengacu pada jurnal yang berjudul “Pengujian *Optimization* dan *Non-Optimization Query* Metode Topsis untuk Menentukan Tingkat Kerusakan Sektor Bencana Alam” yang menggunakan alternatif berupa tingkat kerusakan yakni rusak ringan, rusak sedang, serta rusak berat dalam metode TOPSIS yang digunakan. Selain itu, pada jurnal “*Implementation of Web-Based Weighted Product Use Decision Support System to Determine the Post-Disaster Damage and Loss*”, yang ditulis oleh Bachriwindi dkk. (2019), juga menggunakan ketiga alternatif tersebut dalam penelitian yang dilakukannya. Penggunaan ketiga alternatif tersebut juga sesuai dengan data yang dirilis oleh BPBD Kota Malang.

Tabel 3. 3 Kriteria (Safitri et al., 2022)

Kode Kriteria	Nama Kriteria
C1	Kondisi Bangunan
C2	Kondisi Struktur Bangunan
C3	Kondisi Fisik Bangunan
C4	Fungsi Bangunan
C5	Kondisi Penunjang Lainnya

Kriteria yang digunakan pada penelitian ini ditunjukkan oleh tabel 3.3. Ada lima kriteria yang digunakan yang terdiri dari kondisi bangunan sebagai C1, kondisi struktur bangunan sebagai C2, kondisi fisik bangunan sebagai C3, fungsi bangunan sebagai C4, dan kondisi penunjang lainnya sebagai C5. Penentuan kriteria yang digunakan tersebut juga mengacu pada jurnal yang berjudul “Pengujian *Optimization* dan *Non-Optimization Query* Metode Topsis untuk Menentukan Tingkat Kerusakan Sektor Bencana Alam” yang menggunakan lima kriteria tersebut dalam penilaian. Selain itu, pada jurnal “*Implementation of Web-Based Weighted Product Use Decision Support System to Determine the Post-Disaster Damage and Loss*”, yang ditulis oleh Bachriwindi dkk. (2019), juga menggunakan kelima kriteria tersebut dalam menentukan tingkat kerusakan sektor setelah terjadinya bencana. Kriteria-kriteria yang digunakan tersebut juga sesuai dengan data kejadian bencana alam yang dirilis oleh BPBD Kota Malang.



### 3.2 Preprocessing

Data mentah yang akan digunakan pada penelitian ini berjumlah 328 data, yang merupakan data kejadian bencana dari Badan Penanggulangan Bencana Daerah (BPBD) Kota Malang tahun 2020 dari bulan Januari hingga Desember. Data kejadian bencana yang didapatkan tersebut akan diolah dan dipilah untuk mendapatkan data bersih yang nantinya digunakan dalam sistem. Data bersih yang sebelumnya telah diolah kemudian digunakan sebagai data *training* dan data *testing* dalam penentuan kerusakan sektor setelah terjadinya bencana alam menggunakan metode TOPSIS berbasis *machine learning*. Komposisi *data training* dan *data testing* yang nantinya digunakan pada penelitian ini adalah sebesar 70% untuk *data training* dan 30% untuk *data testing*.

Pengolahan dan pemilahan data mentah yang digunakan pada penelitian ini dilakukan dengan cara mengkonversi nilai yang ada pada data mentah sesuai dengan penilaian kriteria yang telah ditetapkan, seperti yang ditunjukkan pada tabel 3.4 sebagai berikut.

Tabel 3. 4 Ketentuan Nilai Kriteria (Bachriwindi et al., 2019)

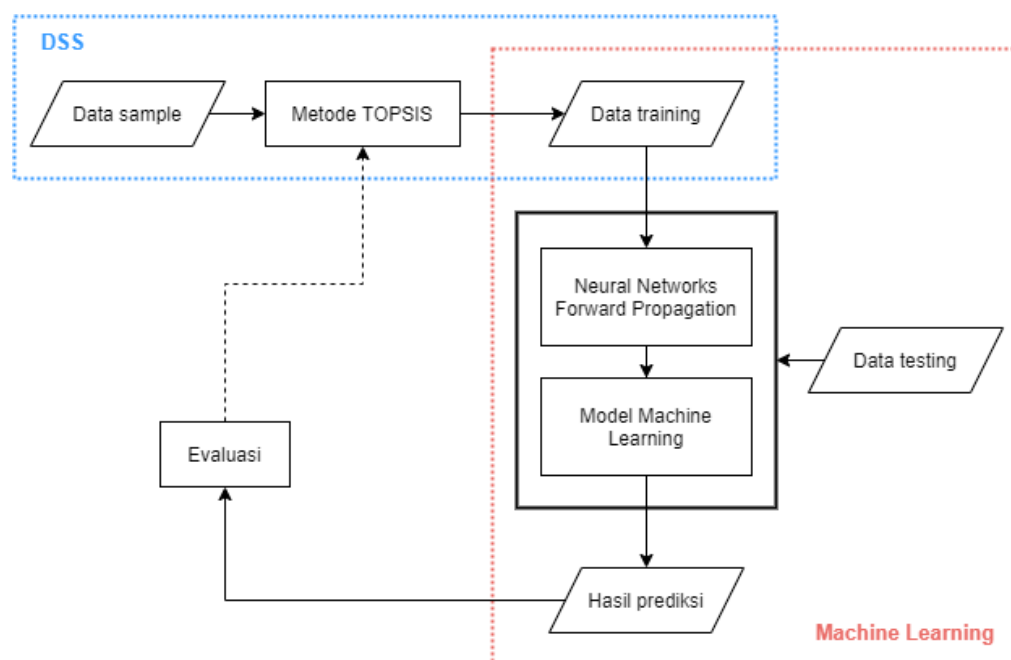
Kriteria	Nilai	Keterangan
Kondisi Bangunan	1	Berdiri
	2	Miring
	3	Hancur
Kondisi Struktur Bangunan	1	Sebagian kecil bangunan rusak
	2	Sebagian bangunan rusak

Kriteria	Nilai	Keterangan
	3	Sebagian besar bangunan rusak
Kondisi Fisik Bangunan	1	< 30 %
	2	30 – 50 %
	3	> 50 %
Fungsi Bangunan	1	Tidak berbahaya untuk digunakan
	2	Relatif bahaya untuk digunakan
	3	Berbahaya untuk digunakan
Kondisi Penunjang Lainnya	1	Sebagian kecil rusak
	2	Sebagian besar rusak
	3	Rusak total

Tabel 3.4 menunjukkan ketentuan penilaian setiap kriteria penentuan tingkat kerusakan sektor bangunan setelah terjadinya bencana alam. Penilaian setiap kriteria tersebut mengacu pada jurnal yang berjudul “*Implementation of Web-Based Weighted Product Use Decision Support System to Determine the Post-Disaster Damage and Loss*” yang mengimplementasikan sistem pendukung keputusan dengan metode *Weighted Product* (WP) untuk menentukan jenis kerusakan sektor (Bachriwindi et al., 2019). Penilaian untuk setiap kriteria yang dilakukan dikonversi dengan nilai pada interval 1 hingga 3, sesuai dengan kondisi tingkat kerusakan bangunan.

### 3.3 Desain Sistem

Pemrograman sistem yang akan dilakukan pada penelitian ini dibangun menggunakan 2 bahasa pemrograman, yakni PHP dan Python. Bahasa pemrograman PHP digunakan untuk membangun metode TOPSIS, yang nantinya akan digunakan dalam mengolah data sampel yang diperoleh untuk kemudian *output* yang dihasilkan akan menjadi data *training* pada pembuatan model *machine learning* menggunakan metode *forward propagation in neural network*. Sedangkan bahasa pemrograman Python digunakan untuk membangun model *machine learning* yang berupa *forward propagation in neural network*. Data *training* yang telah terbentuk tersebut diinputkan ke dalam *forward propagation in neural networks* yang dibuat menggunakan bahasa Python untuk membentuk model *machine learning*. Setelah model *machine learning* tersebut maka akan dimasukkan *data testing* untuk kemudian didapatkan hasil prediksi.



Gambar 3. 1 Desain Sistem

Gambar 3.1 menunjukkan desain sistem yang akan dibangun dan diimplementasikan pada penelitian ini. Berdasarkan gambar tersebut, dapat dilihat langkah-langkah sistem yang dibuat pada penelitian ini. Data sample yang digunakan akan diproses atau diolah menggunakan *Decision Support System* dengan metode TOPSIS. Data hasil dari pemrosesan DSS tersebut kemudian akan dijadikan sebagai data *training* dalam sistem *machine learning* yang dirancang. Data *training* tersebut akan diolah menggunakan metode *neural networks* dengan algoritma *forward propagation*, sehingga terbentuk model *machine learning* yang akan digunakan sebagai perhitungan prediksi. Data *testing* yang digunakan pada penelitian merupakan data baru yang diinputkan oleh tim surveyor, yang berupa data kriteria kerusakan sektor pasca terjadinya bencana. Data *testing* tersebut kemudian akan diolah menggunakan model *machine learning* yang telah terbentuk untuk mendapatkan hasil prediksi. Hasil prediksi tersebut merupakan hasil akhir yang akan diperoleh tim surveyor pada kegiatan penentuan tingkat kerusakan sektor pasca bencana alam. Data hasil prediksi yang juga adalah hasil pemodelan *machine learning* tersebut kemudian akan dievaluasi berdasarkan hasil perhitungan metode TOPSIS untuk mengetahui tingkat akurasi setiap model data.

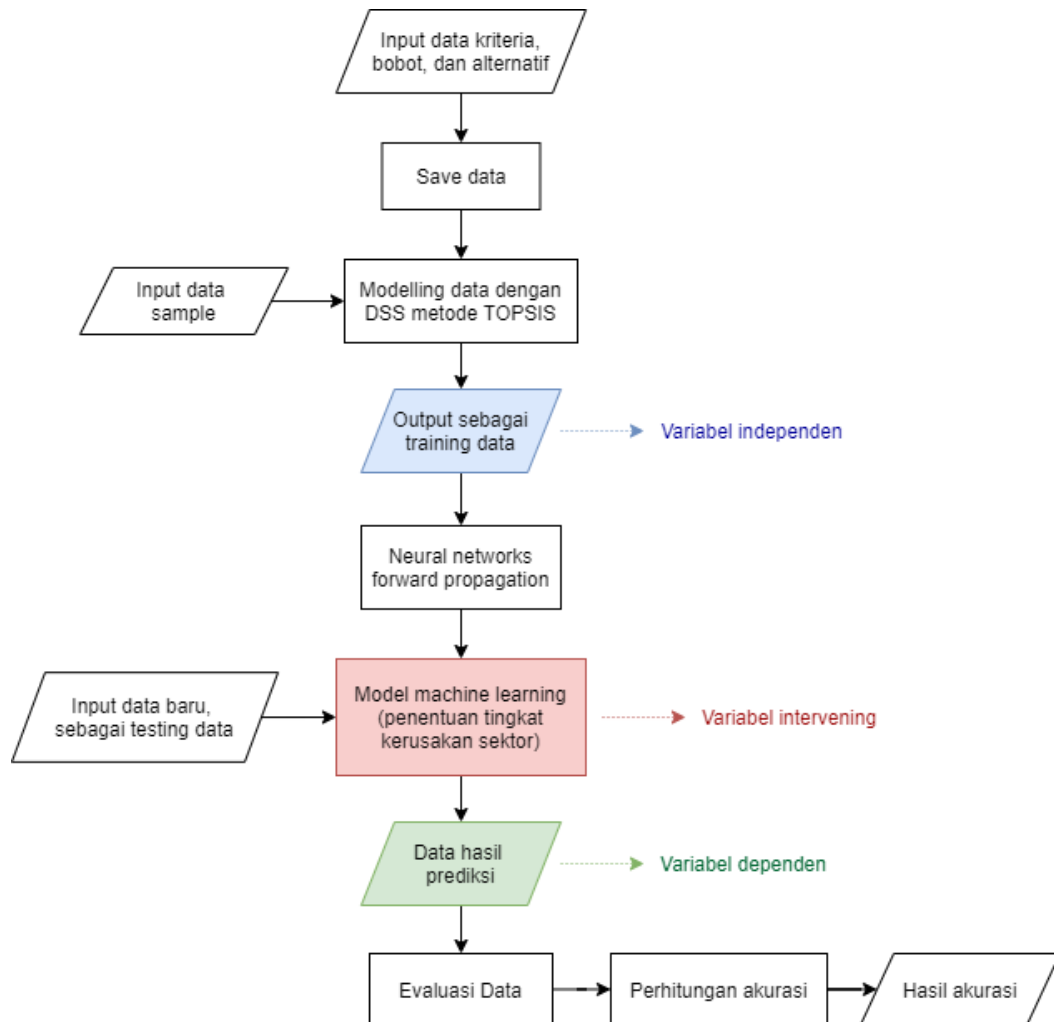
### **3.4 Research Instrument**

Ada tiga jenis variabel yang dipakai dalam penelitian ini, antara lain variabel independen, variabel dependen, serta variabel intervening. Ketiga jenis variabel tersebut masing-masing berupa data yang berbeda dan memiliki fungsi yang juga berbeda. Variabel independen memiliki arti yakni variabel yang mempengaruhi atau menyebabkan munculnya variabel dependen. Variabel

intervening merupakan variabel yang mempengaruhi relasi antara variabel independen dengan variabel dependen secara tidak langsung.

Variabel independen yang dipakai pada penelitian ini merupakan data hasil perhitungan metode TOPSIS, yang berfungsi sebagai data *training* dalam pembuatan model *machine learning* penentuan tingkat kerusakan sektor setelah terjadinya bencana alam. Perhitungan metode TOPSIS yang dilakukan tersebut menggunakan data kriteria, bobot, alternatif, dan matriks keputusan. Sedangkan data *testing* yang digunakan merupakan data baru dari surveyor yang diinputkan, yang kemudian diolah dan di-*training* dengan data *training* pada model *machine learning* yang dirancang. Hasil dari proses tersebut berupa data prediksi tingkat kerusakan sektor pasca bencana alam yang kemudian dievaluasi dengan membandingkannya dengan data *training* yang digunakan. Data hasil prediksi tersebut sendiri merupakan variabel dependen yang ada pada penelitian ini. Sedangkan variabel intervening pada penelitian ini merupakan penentuan atau penilaian tingkat kerusakan sektor setelah terjadinya bencana alam, karena menjelaskan relasi antara tingkat kerusakan sektor setelah terjadinya bencana alam dengan metode TOPSIS berbasis *machine learning*.

### 3.5 Conceptual Framework



Gambar 3. 2 *Conceptual Framework*

Gambar 3.2 menunjukkan *conceptual framework* yang dirancang pada penelitian ini. *Conceptual framework* atau kerangka kerja konseptual yang dijabarkan menunjukkan alur yang dilakukan dalam perancangan sistem pada penelitian ini secara spesifik. *Conceptual framework* tersebut juga menunjukkan hubungan antar variabel yang digunakan, yakni variabel independen (data *training* yang berupa hasil perhitungan metode TOPSIS), variabel dependen (data hasil

prediksi tingkat kerusakan sektor), dan data intervening (penilaian atau penentuan tingkat kerusakan sektor).

Langkah awal yang dilakukan yakni menginput data kriteria, bobot, dan alternatif yang akan dijadikan acuan dalam perhitungan atau pemodelan data menggunakan sistem pendukung keputusan dengan metode TOPSIS. Data sample kemudian diinputkan untuk diolah dan dihitung dengan metode TOPSIS. Hasil perhitungan metode TOPSIS tersebut berperan sebagai data *training* dalam pembuatan model *machine learning*, yang dibuat menggunakan metode *neural networks* dengan algoritma *forward propagation*.

Data baru yang diinputkan oleh tim surveyor berperan sebagai data *testing* pada penentuan tingkat kerusakan sektor setelah terjadinya bencana alam. Data tersebut kemudian diolah dan di-*training* pada model *machine learning* yang telah dirancang. Hasil dari proses tersebut berupa data prediksi tingkat kerusakan sektor pasca bencana alam, yang kemudian akan dievaluasi untuk mengetahui tingkat akurasi sistem.

### **3.6 Perhitungan Manual**

Perhitungan manual yang dilakukan dibagi menjadi 3 tahapan, yang terdiri dari penentuan bobot, perhitungan metode TOPSIS, dan perhitungan metode *forward propagation in neural networks*.

#### **3.6.1 Penentuan Bobot**

Penentuan bobot kriteria yang digunakan pada penelitian ini menggunakan metode ROC. Sebelum melakukan pembobotan untuk masing-masing kriteria,

kelima kriteria yang digunakan dalam penelitian ini diurutkan berdasarkan prioritas yang lebih utama dan ditentukan apakah kriteria tersebut merupakan *benefit* atau *cost*. Penentuan kriteria tersebut dapat dilihat pada tabel 3.5 sebagai berikut.

Tabel 3. 5 Penentuan Kriteria

<b>Kode Kriteria</b>	<b>Nama Kriteria</b>	<b><i>Benefit / Cost</i></b>
C1	Kondisi Bangunan	<i>Benefit</i>
C2	Kondisi Struktur Bangunan	<i>Benefit</i>
C3	Kondisi Fisik Bangunan	<i>Benefit</i>
C4	Fungsi Bangunan	<i>Benefit</i>
C5	Kondisi Penunjang Lainnya	<i>Benefit</i>

Berdasarkan penentuan kriteria yang ditunjukkan oleh tabel 3.5, dilakukan perhitungan bobot menggunakan rumus metode ROC, yang sebelumnya telah dijabarkan pada bab 2, yang ditunjukkan oleh tabel 3.6 sebagai berikut.

Tabel 3. 6 Perhitungan Metode ROC

<b>Kode Kriteria</b>	<b>Metode ROC</b>	<b>Nilai Bobot</b>
C1	$W_1 = (1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \frac{1}{5})/5$	0.4567
C2	$W_2 = (0 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \frac{1}{5})/5$	0.2567
C3	$W_3 = (0 + 0 + \frac{1}{3} + \frac{1}{4} + \frac{1}{5})/5$	0.1567
C4	$W_4 = (0 + 0 + 0 + \frac{1}{4} + \frac{1}{5})/5$	0.09



Kode Kriteria	Metode ROC	Nilai Bobot
C5	$W_5 = (0 + 0 + 0 + 0 + \frac{1}{5})/5$	0.04

### 3.6.2 Perhitungan Metode TOPSIS

Perhitungan dari metode TOPSIS dibagi ke dalam beberapa tahapan sesuai dengan yang telah dijabarkan pada bab 2. Data yang digunakan dalam perhitungan manual metode TOPSIS yang dilakukan merupakan data *dummy*. Perhitungan manual metode TOPSIS dalam penelitian ini dijabarkan sebagai berikut.

- a. Menghitung matriks keputusan yang ternormalisasi.

Sebelum menghitung matriks keputusan ternormalisasi, harus dibentuk matriks keputusan awal (matriks  $X$ ) yang merupakan preferensi dari setiap alternatif terhadap kriteria. Preferensi alternatif terhadap kriteria ditunjukkan oleh tabel 3.7.

Tabel 3. 7 Preferensi Alternatif dan Kriteria

Kode Alternatif	Kode Kriteria				
	C1	C2	C3	C4	C5
A1	1	1	1	1	1
A2	2	2	2	2	2
A3	3	3	3	3	3

Berdasarkan tabel 3.7, dapat dibentuk matriks keputusan awal sebagai berikut.

$$X_{ij} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 2 & 2 \\ 3 & 3 & 3 & 3 & 3 \end{bmatrix}$$

Langkah selanjutnya yakni menghitung matriks keputusan ternormalisasi (matriks  $Y$ ) berdasarkan matriks keputusan awal sebagai berikut.

$$y_{11} = \frac{x_{11}}{\sqrt{\sum_{i=1}^I x_{ij}^2}} = \frac{1}{\sqrt{1^2 + 2^2 + 3^2}} = \frac{1}{3,74} = 0,267$$

$$y_{12} = \frac{x_{12}}{\sqrt{\sum_{i=1}^I x_{ij}^2}} = \frac{1}{\sqrt{1^2 + 2^2 + 3^2}} = \frac{1}{3,74} = 0,267$$

$$y_{13} = \frac{x_{13}}{\sqrt{\sum_{i=1}^I x_{ij}^2}} = \frac{1}{\sqrt{1^2 + 2^2 + 3^2}} = \frac{1}{3,74} = 0,267$$

$$y_{14} = \frac{x_{14}}{\sqrt{\sum_{i=1}^I x_{ij}^2}} = \frac{1}{\sqrt{1^2 + 2^2 + 3^2}} = \frac{1}{3,74} = 0,267$$

$$y_{15} = \frac{x_{15}}{\sqrt{\sum_{i=1}^I x_{ij}^2}} = \frac{1}{\sqrt{1^2 + 2^2 + 3^2}} = \frac{1}{3,74} = 0,267$$

$$y_{21} = \frac{x_{21}}{\sqrt{\sum_{i=1}^I x_{ij}^2}} = \frac{2}{\sqrt{1^2 + 2^2 + 3^2}} = \frac{2}{3,74} = 0,534$$

$$y_{22} = \frac{x_{22}}{\sqrt{\sum_{i=1}^I x_{ij}^2}} = \frac{2}{\sqrt{1^2 + 2^2 + 3^2}} = \frac{2}{3,74} = 0,534$$

Langkah tersebut terus dilakukan hingga  $y_{35}$  dan akan menghasilkan matriks keputusan yang ternormalisasi sebagai berikut.

$$Y_{ij} = \begin{bmatrix} 0,267 & 0,267 & 0,267 & 0,267 & 0,267 \\ 0,534 & 0,534 & 0,534 & 0,534 & 0,534 \\ 0,801 & 0,801 & 0,801 & 0,801 & 0,801 \end{bmatrix}$$

b. Menghitung matriks keputusan ternormalisasi terbobot.

Matriks keputusan ternormalisasi terbobot (matriks  $V$ ) dihitung dengan mengkalikan data pada matriks keputusan ternormalisasi dengan bobot masing-masing kriteria.

$$v_{11} = W_1 * y_{11} = 0,267 * 0,456 = 0,122$$

$$v_{12} = W_2 * y_{12} = 0,267 * 0,256 = 0,068$$

$$v_{13} = W_3 * y_{13} = 0,267 * 0,156 = 0,041$$

$$v_{14} = W_4 * y_{14} = 0,267 * 0,09 = 0,024$$

$$v_{15} = W_1 * y_{15} = 0,267 * 0,04 = 0,01$$

Perhitungan tersebut terus dilakukan hingga  $v_{35}$ , sehingga akan menghasilkan matriks keputusan ternormalisasi terbobot sebagai berikut.

$$V_{ij} = \begin{bmatrix} 0,122 & 0,068 & 0,041 & 0,024 & 0,01 \\ 0,244 & 0,137 & 0,083 & 0,048 & 0,021 \\ 0,366 & 0,205 & 0,125 & 0,072 & 0,032 \end{bmatrix}$$

c. Menemukan solusi ideal positif dan solusi ideal negatif.

Langkah selanjutnya yang dilakukan adalah menghitung solusi ideal positif ( $A^+$ ) dan solusi ideal negatif ( $A^-$ ) yang ditunjukkan pada tabel 3.8.

Tabel 3. 8 Solusi Ideal Positif dan Negatif

Solusi Ideal	Kode Kriteria				
	C1	C2	C3	C4	C5
$A^+$	0,366	0,205	0,125	0,72	0,032
$A^-$	0,122	0,068	0,041	0,24	0,01

d. Memperoleh nilai pemisahan.

Nilai pemisahan merupakan jarak solusi ideal positif dan negatif dengan setiap alternatif pada matriks normalisasi terbobot yang dinotasikan sebagai  $S^+$  untuk nilai pemisahan positif dan  $S^-$  untuk nilai pemisahan negatif.

$$S_1^+ = \sqrt{\sum_{j=1}^J (v_{1j} - v_j^+)^2}$$

$$\begin{aligned} \sum_{j=1}^J (v_{1j} - v_j^+)^2 &= (0,122 - 0,366)^2 + (0,068 - 0,205)^2 + \\ &\quad (0,041 - 0,125)^2 + (0,024 - 0,072)^2 + \\ &\quad (0,01 - 0,032)^2 = 0,088 \end{aligned}$$

$$S_1^+ = \sqrt{0,088} = 0,296$$

$$S_1^- = \sqrt{\sum_{j=1}^J (v_{1j} - v_j^-)^2}$$

$$\begin{aligned} \sum_{j=1}^J (v_{1j} - v_j^-)^2 &= (0,122 - 0,122)^2 + (0,068 - 0,068)^2 + \\ &\quad (0,041 - 0,041)^2 + (0,024 - 0,024)^2 + \\ &\quad (0,01 - 0,01)^2 = 0 \end{aligned}$$

$$S_1^- = \sqrt{0} = 0$$

Perhitungan tersebut dilakukan hingga alternatif ke-3 ( $S_3^+$  dan  $S_3^-$ ).

Hasil perhitungan nilai pemisahan ditunjukkan oleh tabel 3.9.

Tabel 3. 9 Nilai Pemisahan

Kode Alternatif	Nilai Pemisahan	
	$S^+$	$S^-$
A1	0,296	0
A2	0,148	0,148

Kode Alternatif	Nilai Pemisahan	
	$S^+$	$S^-$
A3	0	0,296

e. Menghitung keseluruhan nilai preferensi.

Langkah terakhir dalam perhitungan metode TOPSIS adalah menghitung nilai preferensi ( $V_i$ ) sebagai berikut.

$$V_1 = \frac{S_1^-}{S_1^- + S_1^+} = \frac{0}{0+0,296} = 0$$

$$V_2 = \frac{S_2^-}{S_2^- + S_2^+} = \frac{0,148}{0,148 + 0,148} = 0,5$$

$$V_3 = \frac{S_3^-}{S_3^- + S_3^+} = \frac{0,296}{0+0,296} = 1$$

Hasil perhitungan tersebut juga ditunjukkan oleh tabel 3.10 sebagai berikut.

Tabel 3. 10 Nilai Preferensi

Kode Alternatif	Nilai V	Ranking
A1	0	3
A2	0,5	2
A3	1	1

### 3.6.3 Perhitungan Metode *Forward Propagation in Neural Networks*

Hasil perhitungan metode TOPSIS yang telah dilakukan sebelumnya dalam *forward propagation in neural networks* digunakan sebagai *data training* yang ditunjukkan dalam tabel 3.11 sebagai berikut.

Tabel 3. 11 Hasil Metode TOPSIS

<b>Kriteria</b> <b>Alternatif</b>	<b>C1</b>	<b>C2</b>	<b>C3</b>	<b>C4</b>	<b>C5</b>	<b>Nilai V</b>
<b>A1</b>	1	1	1	1	1	0
<b>A2</b>	2	2	2	2	2	0,5
<b>A3</b>	3	3	3	3	3	1

Nilai bobot awal yang didapatkan, yang berupa angka random dengan interval 0 sampai 1, nilai bias, serta target diasumsikan adalah sebagai berikut.

$$\begin{bmatrix} W_1 = 0,1 \\ W_2 = 0,3 \\ W_3 = 0,1 \\ W_4 = 0,1 \\ W_5 = 0,2 \end{bmatrix} [b = 1] [target = 1]$$

Setelah mendapatkan nilai bobot dan bias, kemudian menentukan *data testing* yang digunakan. Diasumsikan *data testing* yang digunakan adalah sebagai berikut.

$$X = [3 \quad 2 \quad 3 \quad 1 \quad 1]$$

Berikut merupakan perhitungan manual metode *forward propagation in neural network* berdasarkan *data input*, bobot, dan bias yang telah ditentukan.

$$y = 0,1 * 3 + 0,3 * 2 + 0,1 * 3 + 0,1 * 1 + 0,2 * 1 + 1$$

$$y = 2,5$$

Langkah kedua yang dilakukan adalah menghitung nilai prediksi dengan mengaktivasi nilai *y* sebagai berikut.

$$z = \frac{1}{1 + e^{-2,5}} = 0,924$$

Berdasarkan perhitungan yang telah dilakukan, didapatkan nilai prediksi sebesar 0,924. Nilai target dan prediksi yang didapatkan kemudian dapat digunakan untuk menghitung nilai *squared error*, yang nantinya digunakan untuk mengetahui besar nilai MSE, MAPE, dan akurasi. Perhitungan *squared error* yang didapatkan dari *data input* adalah sebagai berikut.

$$E = (\text{target} - \text{prediksi})^2 = (1 - 0,924)^2 = 0,005776$$

### 3.7 Hipotesis

Pada penelitian ini dilakukan penentuan kerusakan sektor setelah terjadinya bencana alam dengan menggunakan metode TOPSIS berbasis *machine learning*. Hasil perhitungan pada *decision support system* dengan metode TOPSIS digunakan sebagai data *training* dan data patokan dalam sistem yang dibangun. Implementasi *machine learning* dengan menggunakan metode *neural networks* dan algoritma *forward propagation* yang diterapkan dalam sistem dapat digunakan untuk memprediksi tingkat kerusakan sektor. Sistem yang dibuat pada penelitian ini juga dapat digunakan dalam membantu tim surveyor pada penentuan tingkat kerusakan sektor setelah terjadinya bencana alam.

## **BAB IV**

### **UJI COBA DAN PEMBAHASAN**

#### **4.1 Implementasi Sistem**

Implementasi sistem yang diterapkan pada penelitian ini sesuai dengan metode yang digunakan, yakni metode TOPSIS dan *forward propagation in neural network*.

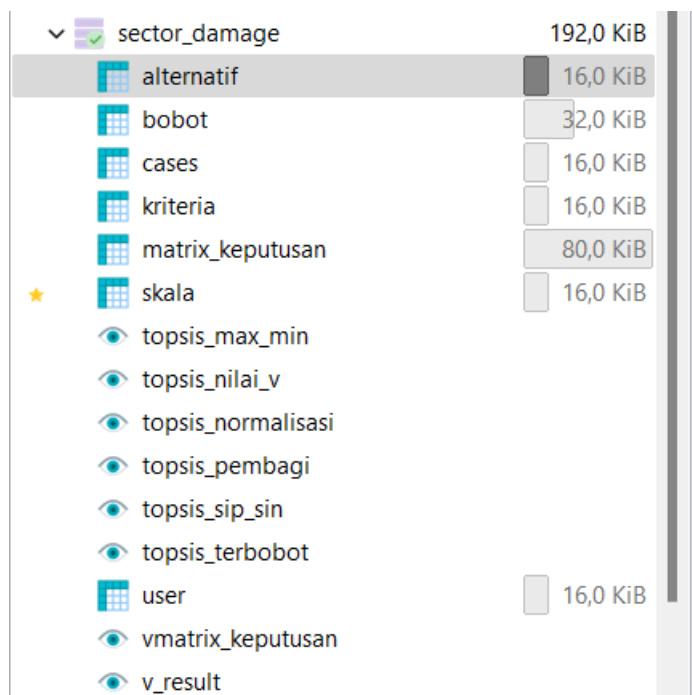
##### **4.1.1 Implementasi Metode TOPSIS**

Pada tahapan ini, dibuat sistem menggunakan bahasa pemrograman PHP serta *database mysql*. Perhitungan metode TOPSIS yang dilakukan menggunakan *view* yang ada pada *database*. Berikut merupakan detail *database* serta tampilan sistem yang dibuat.

###### **4.1.1.1 Database Sistem**

Data-data yang digunakan pada penelitian ini disimpan ke dalam *database* “sector-damage”, yang berisi 7 tabel dan 8 *view*. Tabel yang dibuat digunakan untuk menyimpan data alternatif, bobot, *cases*, kriteria, matriks keputusan, skala penilaian, dan *user*. Sedangkan *view* dibuat untuk melakukan perhitungan setiap langkah dari metode TOPSIS. Struktur *database* yang dibuat pada penelitian ini dapat dilihat pada gambar 4.1 sebagai berikut.





Gambar 4. 1 Database Sistem

Tabel alternatif berfungsi untuk menyimpan data alternatif yang digunakan dalam penelitian ini. Pada tabel ini terdapat 2 kolom, yakni kolom “id\_alternatif”, sebagai *primary key*, dan kolom “nama\_alternatif”. Gambar 4.2 merupakan tabel alternatif yang ada pada *database*.

id_alternatif	nama_alternatif
1	Rusak Ringan
2	Rusak Sedang
3	Rusak Berat

Gambar 4. 2 Tabel Alternatif

Data kriteria yang digunakan dalam penelitian ini disimpan dalam tabel kriteria. Pada tabel ini terdapat 2 kolom, yakni kolom “id\_kriteria” dan “nama\_kriteria”. Gambar 4.3 menunjukkan tabel kriteria yang ada pada *database*.

id_kriteria	nama_kriteria
1	Kondisi Bangunan
2	Kondisi Struktur Bangunan
3	Kondisi Fisik Bangunan
4	Fungsi Bangunan
5	Kondisi Penunjang Lainnya

Gambar 4. 3 Tabel Kriteria

Tabel bobot difungsikan sebagai tempat menyimpan data bobot setiap kriteria yang digunakan pada penelitian ini. Pada tabel ini terdapat 3 kolom, yakni kolom “id\_bobot” sebagai *primary key*, kolom “id\_kriteria” yang merupakan *foreign key* dari tabel kriteria, dan kolom “value” yang berisi *value* bobot masing-masing kriteria. Tabel bobot ditunjukkan oleh gambar 4.4 sebagai berikut.

id_bobot	id_kriteria	value
1	1	0.456
2	2	0.256
3	3	0.156
4	4	0.09
5	5	0.04

Gambar 4. 4 Tabel Bobot

Tabel cases digunakan untuk menyimpan data kejadian kerusakan akibat bencana alam, yang didapat dari BPBD Kota Malang. Ada 6 kolom yang terdapat pada tabel ini, antara lain kolom “id\_case” sebagai *primary key*, “jenis\_bencana”, “tgl\_kejadian”, “lokasi” yang berupa lokasi terjadinya bencana, “sektor” yang berupa sektor bangunan yang mengalami kerusakan, serta “tgl\_input”. Gambar 4.5 menunjukkan tabel cases yang terdapat pada *database*.

id_case	jenis_bencana	tgl_kejadian	lokasi	sektor	tgl_input
16	Tanah Longsor	2020-01-03	Jl. Embong Brantas II, Kidul Dalem, Klojen	Plengsengan & pos bermain	2022-10-11
17	Pohon Tumbang	2020-01-04	Jl. Raya Walet, Sukun, Sukun	warung	2022-10-11
18	Tanah Longsor	2020-01-07	Jl. Kanjuruhan IV no. 3, Tlogomas, Lowokwaru	Plengsengan	2022-10-11
19	Tanah Longsor	2020-01-16	Jl. Kanjuruhan IV no. 3, Tlogomas, Lowokwaru	Plengsengan	2022-10-11
20	Tanah Longsor	2020-01-21	Jl. Embong Brantas II, Kidul Dalem, Klojen	Plengsengan & pos bermain	2022-10-11
21	Pohon Tumbang	2020-01-22	Jl. Ir. H. Juanda Gg. IX, Jodipan, Blimbing	rumah	2022-10-11
22	Pohon Tumbang	2020-01-24	Jl. Embong Brantas, Jodipan, Blimbing	rumah	2022-10-11
23	Pohon Tumbang	2020-01-25	Jl. Bromo Gg. II no. 8, Oro-oro dowo, Klojen	rumah	2022-10-11
24	Tanah Longsor	2020-01-26	Jl. MT. Haryono Gg.21, Dinoyo, Lowokwaru	Plataran rumah	2022-10-11
25	Tanah Longsor	2020-01-26	Jl. MT. Haryono XI, Dinoyo, Lowokwaru	Plengsengan	2022-10-11
26	Tanah Longsor	2020-01-30	Jl. Kolonel Sugiono V, Mergosono, Kedungkand...	rumah	2022-10-11
27	banjir	2020-02-01	Jl. Bareng Raya IIA / 394, Bareng, Klojen	rumah	2022-10-11
28	Tanah Longsor	2020-02-01	Jl. S. Supriadi, Sukun, Sukun	rumah	2022-10-11
29	Tanah Longsor	2020-02-07	Pos Kamling, Sawojajar, Kedungkandang	Pos Kamling	2022-10-11
30	Tanah Longsor	2020-02-07	Jl. Raya Balearjosari, Balearjosari, Blimbing	Plengsengan	2022-10-11
31	Tanah Longsor	2020-02-10	Jl. Jaksa Agung Suprpto I, Samaan, Klojen	Plengsengan	2022-10-11
32	Tanah Longsor	2020-02-15	Jl. Kyai Sofyan Yusuf, Kedungkandang, Kedung...	rumah	2022-10-11
33	banjir	2020-02-16	Jl. Ir. Rais Gg.IX	rumah	2022-10-11
34	banjir	2020-02-16	Sentra Industri Sanitair, Karangbesuki, Sukun	rumah	2022-10-11
35	banjir	2020-02-16	Jl. Bareng Raya, Bareng, Klojen	rumah	2022-10-11
36	banjir	2020-10-16	Jl. Soekarno Hatta Indah no. 1A, Mojolangu, Lo...	rumah	2022-10-11
37	banjir	2020-02-16	Jl. Bandungrejosari, Bandungrejosari, Sukun	rumah	2022-10-11
38	Tanah Longsor	2020-02-16	Jl. Klayatan III no.14, Bandungrejosari, Sukun	rumah	2022-10-11
39	Cuaca Ekstrem	2020-02-17	Jl. Kalimantan, Kasin, Klojen	Pujasera	2022-10-11

Gambar 4. 5 Tabel Cases

Skala penilaian yang digunakan pada penelitian ini disimpan dalam tabel skala yang ada pada *database*. Tabel ini sendiri terdiri dari 3 kolom, yakni kolom “id\_skala” sebagai *primary key*, “value”, dan “keterangan”. Tabel skala yang dibuat dapat dilihat pada gambar 4.6.

id_skala	value	keterangan
1	1	Ringan
2	2	Sedang
3	3	Berat

Gambar 4. 6 Tabel Skala

Tabel matrix keputusan berguna untuk membangun matriks keputusan yang nantinya digunakan dalam perhitungan metode TOPSIS. Pada tabel ini terdapat 5 kolom, yakni kolom “id\_matrix” sebagai *primary key*, “id\_case” yang merupakan *foreign key* dari tabel cases, “id\_alternatif” yang merupakan *foreign key* dari tabel

alternatif, “id\_bobot” yang merupakan *foreign key* dari tabel bobot, serta “id\_skala” yang berupa *foreign key* dari tabel skala. Gambar 4.7 menunjukkan tabel matrix keputusan yang dibuat pada penelitian ini.

id_matrix	id_case	id_alternatif	id_bobot	id_skala
1	1	1	1	1
2	1	1	2	1
3	1	1	3	1
4	1	1	4	1
5	1	1	5	1
6	1	2	1	2
7	1	2	2	2
8	1	2	3	2
9	1	2	4	2
10	1	2	5	2
11	1	3	1	3
12	1	3	2	3
13	1	3	3	3
14	1	3	4	3
15	1	3	5	3
46	2	1	1	1
47	2	2	1	1
48	2	3	1	3
49	2	1	2	1
50	2	2	2	1
51	2	3	2	1
52	2	1	3	1
53	2	2	3	1
54	2	3	3	1

Gambar 4. 7 Tabel Matrix Keputusan

Data *user* sistem yang dibuat disimpan dalam tabel user yang ada pada *database*. Tabel ini sendiri terdiri dari 4 kolom, yakni kolom “username”, “nama”, “email”, dan “password”. Tabel user yang dibuat dapat dilihat pada gambar 4.8 sebagai berikut.

username	nama	email	password
amilafadh	amila fadhila	amila@gmail.com	123456

Gambar 4. 8 Tabel *User*

*View* pada *database* digunakan sebagai implementasi langkah-langkah pada perhitungan metode TOPSIS. Langkah-langkah tersebut dimulai dari membuat matriks keputusan, menentukan nilai pembagi, membuat matriks ternormalisasi dan ternormalisasi terbobot, menentukan nilai maksimum (solusi ideal positif) dan nilai minimum (solusi ideal negatif), menghitung nilai pemisahan positif dan negatif, menghitung nilai  $V$  (preferensi), hingga yang terakhir menentukan hasil tingkat kerusakan bangunan. Berikut merupakan *view* yang ada pada *database*, yang digunakan dalam perhitungan metode TOPSIS.

## Deklarasi

```

id_matrix      : INTEGER
id_case        : INTEGER
id_alternatif  : INTEGER
nama_alternatif : VARCHAR
id_kriteria    : INTEGER
nama_kriteria  : VARCHAR
id_bobot       : INTEGER
value          : VARCHAR
nilai          : INTEGER
keterangan     : VARCHAR

```

## Deskripsi

```

READ (id_matrix, id_case, id_alternatif,
      nama_alternatif, id_kriteria, nama_kriteria,

```

```

        id_bobot, value, nilai, keterangan)
IF id_alternatif = id_alternatif AND
    id_bobot = id_bobot AND
    id_skala = id_skala AND
    id_kriteria = id_kriteria THEN
WRITE (id_matrix, id_case, id_alternatif,
        nama_alternatif, id_kriteria, nama_kriteria,
        id_bobot, value, nilai, keterangan)
End

```

Gambar 4. 9 Pseudocode View Matrix Keputusan

```

Deklarasi
    id_case      : INTEGER
    id_kriteria  : INTEGER
    nama_kriteria : VARCHAR
    nilai        : INTEGER
    bagi         : DOUBLE
Deskripsi
    READ (id_case, id_kriteria, nama_kriteria, nilai)
    bagi <- SQRT(SUM(POW(nilai, 2)))
    WRITE (id_case, id_kriteria, nama_kriteria, bagi)
        GROUP BY (id_kriteria, id_case)
End

```

Gambar 4. 10 Pseudocode View Topsis Pembagi

```

Deklarasi
    id_matrix      : INTEGER
    id_case        : INTEGER
    id_alternatif  : INTEGER
    nama_alternatif : VARCHAR
    id_kriteria    : INTEGER
    nama_kriteria  : VARCHAR

```

```

id_bobot      : INTEGER
value         : INTEGER
nilai         : INTEGER
keterangan    : VARCHAR
bagi          : DOUBLE
normalisasi   : DOUBLE

```

#### Deskripsi

```

READ (id_matrix, id_case, id_alternatif,
      nama_alternatif, id_kriteria, nama_kriteria,
      id_bobot, value, nilai, keterangan, bagi)
IF id_kriteria = id_kriteria AND
   id_case = id_case THEN
normalisasi <- nilai/bagi
WRITE (id_matrix, id_case, id_alternatif,
       nama_alternatif, id_kriteria, nama_kriteria,
       id_bobot, value, nilai, keterangan,
       normalisasi)
GROUP BY (id_matrix, id_case)

```

End

Gambar 4. 11 Pseudocode View Topsis Normalisasi

#### Deklarasi

```

id_matrix      : INTEGER
id_case       : INTEGER
id_alternatif  : INTEGER
nama_alternatif : VARCHAR
id_kriteria    : INTEGER
nama_kriteria  : VARCHAR
id_bobot      : INTEGER
value         : INTEGER
nilai         : INTEGER

```

```

keterangan      : VARCHAR
normalisasi    : DOUBLE
terbobot       : DOUBLE
Deskripsi
  READ (id_matrix, id_case, id_alternatif,
        nama_alternatif, id_kriteria, nama_kriteria,
        id_bobot, value, nilai, keterangan,
        normalisasi)
  IF id_kriteria = id_kriteria THEN
  terbobot <- value * normalisasi
  WRITE (id_matrix, id_case, id_alternatif,
         nama_alternatif, id_kriteria, nama_kriteria,
         id_bobot, value, nilai, keterangan,
         normalisasi, terbobot)
  GROUP BY (id_matrix, id_case)
End

```

Gambar 4. 12 Pseudocode View Topsis Terbobot

```

Deklarasi
  id_matrix      : INTEGER
  id_case       : INTEGER
  id_kriteria   : INTEGER
  nama_kriteria : VARCHAR
  terbobot      : DOUBLE
  maximum       : DOUBLE
  minimum       : DOUBLE
Deskripsi
  READ (id_mtarix, id_case, id_kriteria,
        nama_kriteria, terbobot)
  maximum <- MAX(terbobot)
  minimum <- MIN(terbobot)

```



```

WRITE (id_mtarix, id_case, id_kriteria,
      nama_kriteria, maximum, minimum)
GROUP BY (id_kriteria, id_case)
End

```

Gambar 4. 13 *Pseudocode View Topsis Max Min*

```

Deklarasi
id_case      : INTEGER
id_alternatif : INTEGER
maximum      : DOUBLE
minimum      : DOUBLE
terbobot     : DOUBLE
dplus        : DOUBLE
dmin         : DOUBLE

Deskripsi
READ (id_case, id_alternatif, maximum, minimum,
     terbobot)
IF id_kriteria = id_kriteria AND
   id_case = id_case THEN
dplus <- SQRT(SUM(POW((maximum - terbobot),2)))
dmin  <- SQRT(SUM(POW((minimum - terbobot),2)))
WRITE (id_case, id_alternatif, dplus, dmin)
GROUP BY (id_alternatif, id_case)
End

```

Gambar 4. 14 *Pseudocode View Topsis SIP SIN*

```

Deklarasi
id_case      : INTEGER
id_alternatif : INTEGER
dplus        : DOUBLE
dmin         : DOUBLE
nilai_v      : DOUBLE

```

```

Deskripsi
  READ (id_case, id_alternatif, dplus, dmin)
  nilai_v <- dmin / (dplus + dmin)
  WRITE (id_case, id_alternatif, dplus, dmin,
        nilai_v)
  GROUP BY (id_alternatif, id_case)
End

```

Gambar 4. 15 *Pseudocode View Topsis Nilai V*

```

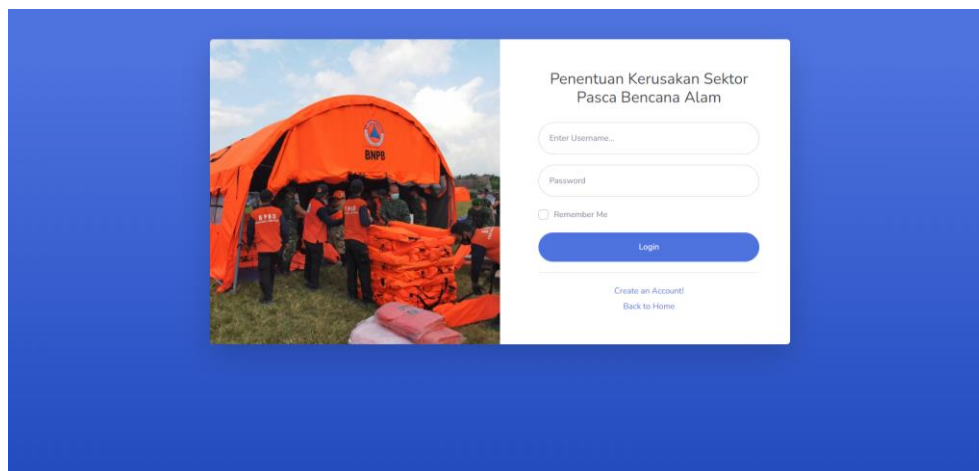
Deklarasi
  id_case      : INTEGER
  nilai_v      : DOUBLE
  id_alternatif : INTEGER
  result       : INTEGER
Deskripsi
  READ (id_case, nilai_v, id_alternatif)
  IF nilai_v IN MAX(nilai_v) THEN
  result <- id_alternatif
  WRITE (id_case, nilai_v, result)
End

```

Gambar 4. 16 *Pseudocode View Result*

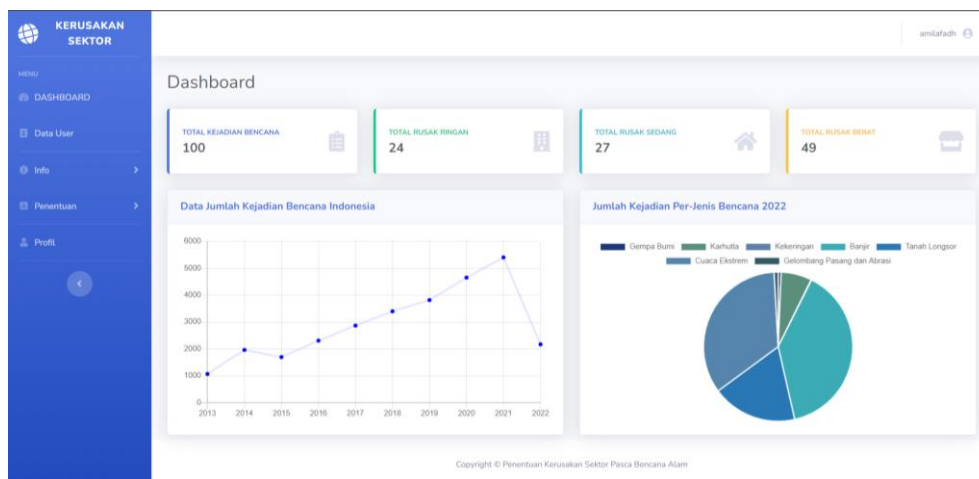
#### 4.1.1.2 Tampilan Sistem

Sistem untuk metode TOPSIS yang dibuat pada penelitian ini menerapkan *user interface* untuk mempermudah *user* dalam memasukkan data ke dalam *database*. Sistem yang dibuat memiliki beberapa halaman tampilan sesuai dengan penelitian yang dilakukan. Berikut merupakan *screenshot* tampilan setiap halaman pada sistem yang dibuat.

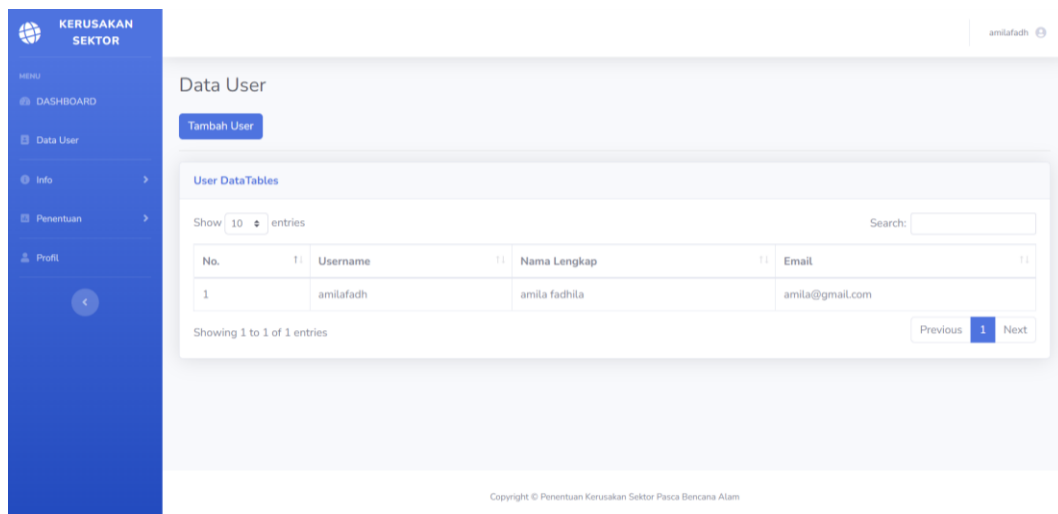


Gambar 4. 17 Halaman Login

Gambar 4.17 merupakan halaman *login*. Pada halaman ini, *user* harus meng-*input*-kan *username* dan *password* dengan benar untuk dapat masuk ke dalam sistem. Apabila *username* atau *password* yang di-*input*-kan salah, maka *user* tidak akan dapat masuk ke dalam sistem.

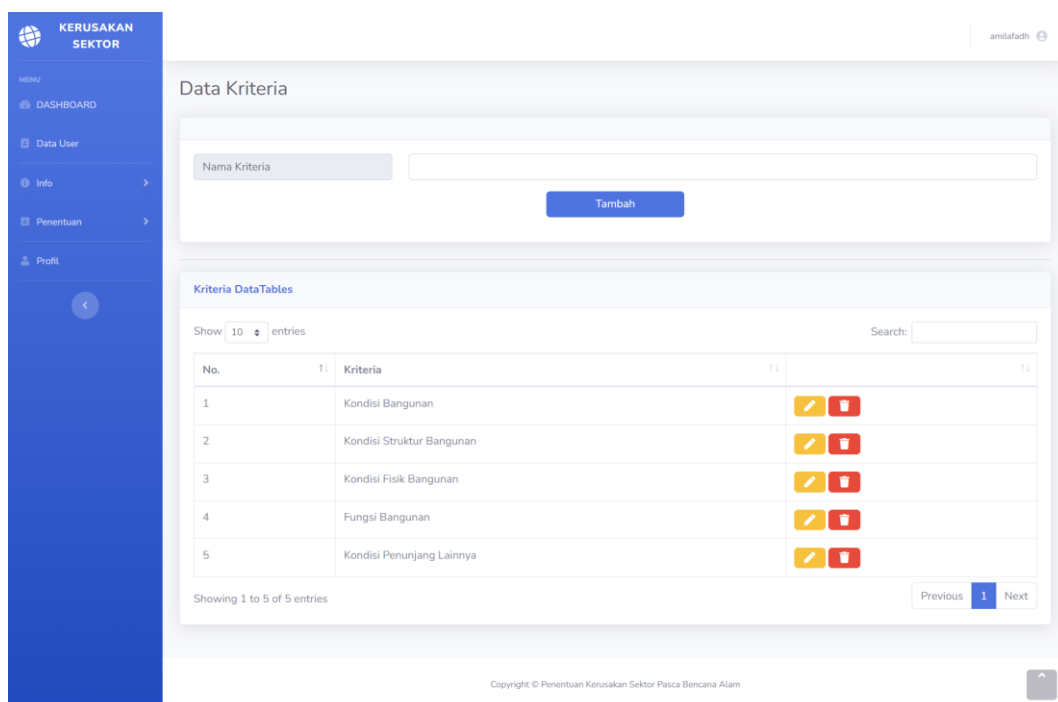
Gambar 4. 18 Halaman *Dashboard*

Gambar 4.18 merupakan halaman *dashboard* yang berisi mengenai informasi tentang jumlah data kerusakan yang telah di-*input* serta jumlah kasus rusak ringan, rusak sedang, dan rusak berat. Pada halaman ini juga terdapat grafik mengenai jumlah kejadian bencana di Indonesia.



Gambar 4. 19 Halaman Data *User*

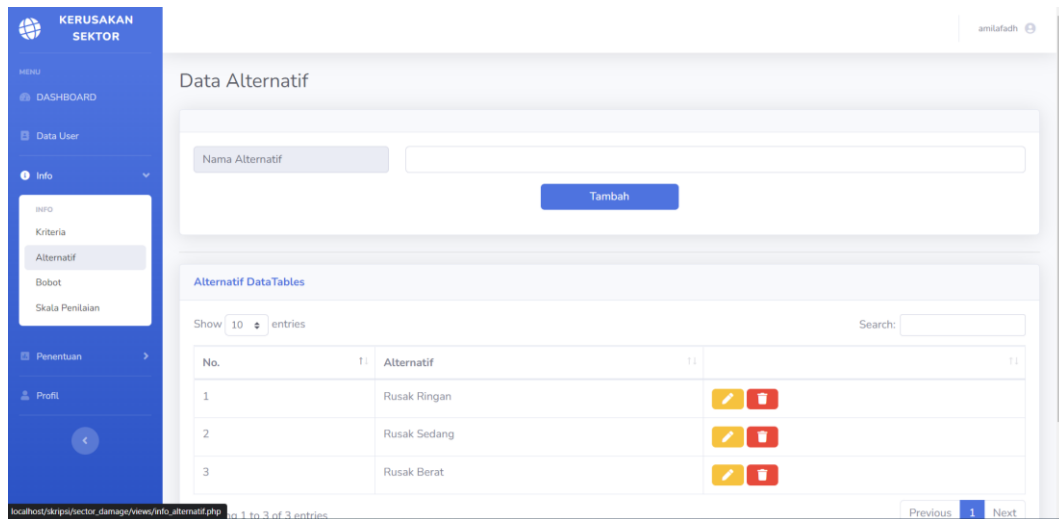
Gambar 4.19 merupakan halaman data *user* yang terdaftar pada sistem. Pada halaman ini, terdapat tabel informasi mengenai *user* yang terdiri dari kolom *username*, nama lengkap, dan *email*.



Gambar 4. 20 Halaman Data Kriteria

Gambar 4.20 merupakan halaman untuk menampilkan dan menambahkan data kriteria yang digunakan pada penelitian ini. Halaman ini berisi tabel informasi

data kriteria yang terdiri dari kolom nomor dan nama kriteria, serta *form* untuk *input* data kriteria.



Gambar 4. 21 Halaman Data Alternatif

Gambar 4.21 merupakan halaman untuk menampilkan dan menambahkan data alternatif. Halaman ini berisi tabel informasi data alternatif yang terdiri dari kolom nomor dan nama alternatif, serta *form input* untuk menambahkan data alternatif.

**Data Bobot**

Kriteria:  Value:

**Tambah**

**Bobot DataTables**

Show 10 entries Search:

No.	Kriteria	Value	
1	Kondisi Bangunan	0.456	
2	Kondisi Struktur Bangunan	0.256	
3	Kondisi Fisik Bangunan	0.156	
4	Fungsi Bangunan	0.09	
5	Kondisi Penunjang Lainnya	0.04	

Showing 1 to 5 of 5 entries Previous 1 Next

Copyright © Penentuan Kerusakan Sektor Pasca Bencana Alam

Gambar 4. 22 Halaman Data Bobot

Gambar 4.22 merupakan halaman untuk data bobot. Pada halaman ini terdapat tabel informasi data bobot yang terdiri dari kolom nomor, nama kriteria, dan *value* bobot masing-masing kriteria, serta *form* untuk *input* data bobot.

**Skala Penilaian**

Value:  Keterangan:

**Tambah**

**Skala Penilaian DataTables**

Show 10 entries Search:

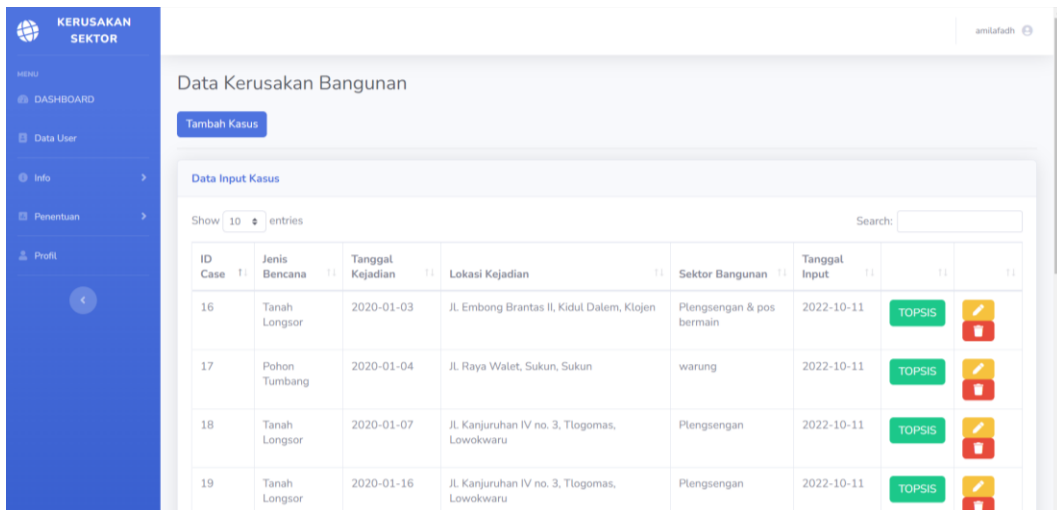
Value	Keterangan	
1	Ringan	
2	Sedang	
3	Berat	

Showing 1 to 3 of 3 entries Previous 1 Next

Copyright © Penentuan Kerusakan Sektor Pasca Bencana Alam

Gambar 4. 23 Halaman Skala Penilaian

Gambar 4.23 merupakan halaman skala penilaian. Halaman ini berisi tabel skala penilaian yang digunakan, yang terdiri dari kolom *value* dan keterangan, serta *form input* untuk menambahkan data skala penilaian baru.

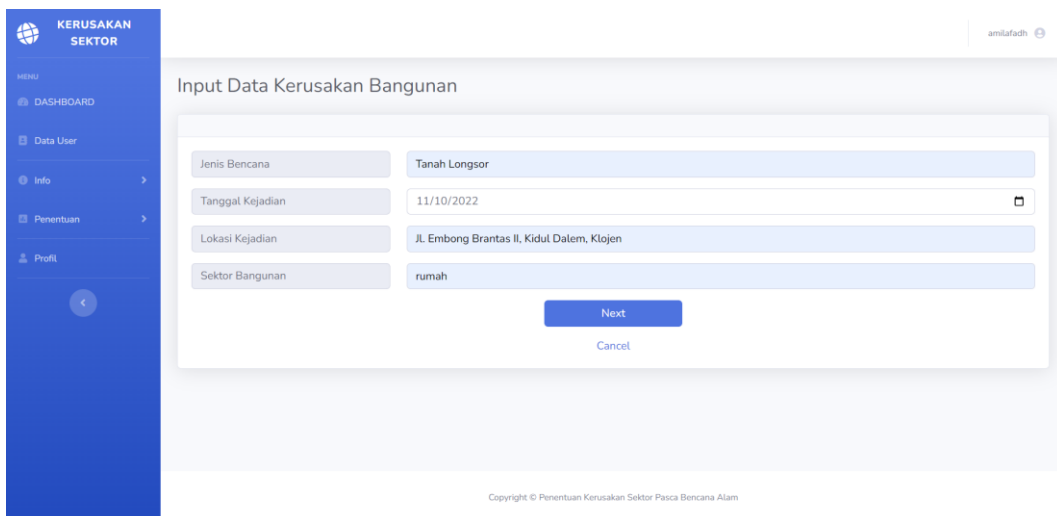


The screenshot shows a web application interface for 'KERUSAKAN SEKTOR'. The main content area is titled 'Data Kerusakan Bangunan' and features a 'Tambah Kasus' button. Below this is a 'Data Input Kasus' section with a search bar and a table of cases. The table has columns for ID Case, Jenis Bencana, Tanggal Kejadian, Lokasi Kejadian, Sektor Bangunan, and Tanggal Input. Each row includes a 'TOPSIS' button and a trash icon.

ID Case	Jenis Bencana	Tanggal Kejadian	Lokasi Kejadian	Sektor Bangunan	Tanggal Input
16	Tanah Longsor	2020-01-03	Jl. Embong Brantas II, Kidul Dalem, Klojen	Pengsengan & pos bermain	2022-10-11
17	Pohon Tumbang	2020-01-04	Jl. Raya Walet, Sukun, Sukun	warung	2022-10-11
18	Tanah Longsor	2020-01-07	Jl. Kanjuruhan IV no. 3, Tlogomas, Lowokwaru	Pengsengan	2022-10-11
19	Tanah Longsor	2020-01-16	Jl. Kanjuruhan IV no. 3, Tlogomas, Lowokwaru	Pengsengan	2022-10-11

Gambar 4. 24 Halaman Data Kerusakan Bangunan

Gambar 4.24 merupakan halaman untuk data kasus kerusakan bangunan. Halaman ini berisi tabel informasi data kasus kerusakan yang terdiri dari kolom *id case*, jenis bencana, tanggal kejadian, lokasi kejadian, sektor bangunan, dan tanggal *input* data. Pada halaman ini juga terdapat tombol ke halaman tambah kasus.



The screenshot shows the 'Input Data Kerusakan Bangunan' form. It contains several input fields: 'Jenis Bencana' (Tanah Longsor), 'Tanggal Kejadian' (11/10/2022), 'Lokasi Kejadian' (Jl. Embong Brantas II, Kidul Dalem, Klojen), and 'Sektor Bangunan' (rumah). There are 'Next' and 'Cancel' buttons at the bottom of the form.

Gambar 4. 25 Halaman Input Data (1)

KERUSAKAN SEKTOR

amalfadh

### Input Penilaian Kerusakan

ID Case: 128

Jenis Bencana: Tanah Longsor

Tanggal Kejadian: 2022-11-10

Lokasi Kejadian: Jl. Embong Brantas II, Kidul Dalem, Klojen

Sektor Bangunan: rumah

Kriteria	Rusak Ringan	Rusak Sedang	Rusak Berat
Kondisi Bangunan	Ringan	Ringan	Ringan
Kondisi Struktur Bangunan	Ringan	Ringan	Ringan
Kondisi Fisik Bangunan	Ringan	Ringan	Ringan
Fungsi Bangunan	Ringan	Ringan	Ringan
Kondisi Penunjang Lainnya	Ringan	Ringan	Ringan

Submit

Copyright © Penentuan Kerusakan Sektor Pasca Bencana Alam

Gambar 4. 26 Halaman Input Data (2)

Gambar 4.25 dan 4.26 merupakan halaman *input* data. Halaman tersebut digunakan untuk menambahkan data kasus kerusakan bangunan yang terjadi akibat terjadinya bencana alam. Data tersebut meliputi jenis bencana, tanggal kejadian, lokasi kejadian, sektor bangunan, serta matriks keputusan berdasarkan kriteria dan alternatif yang ada.



**Perhitungan TOPSIS**

← Kembali

ID Case: 16

Jenis Bencana: Tanah Longsor

Tanggal Kejadian: 2020-01-03

Lokasi Kejadian: Jl. Embong Brantas II, Kidul Dalem, Klojen

Sektor Bangunan: Plengsengan & pos bermain

1 - Matriks Keputusan | 2 - Pembagi | 3 - Normalisasi Terbobot | 4 - Maximum & Minimum | 5 - Pemisahan | 6 - Preferensi

**Matrix Keputusan**

Show 10 entries Search:

Kriteria	1	Alternatif 1 - Rusak Ringan	1	Alternatif 2 - Rusak Sedang	1	Alternatif 3 - Rusak Berat	1
K1 - Kondisi Bangunan		1		1		3	
K2 - Kondisi Struktur Bangunan		1		1		3	
K3 - Kondisi Fisik Bangunan		1		1		3	
K4 - Fungsi Bangunan		1		1		3	
K5 - Kondisi Penunjang Lainnya		1		1		3	

Showing 1 to 5 of 5 entries Previous 1 Next

Copyright © Penentuan Kerusakan Sektor Pasca Bencana Alam

Gambar 4. 27 Halaman Perhitungan TOPSIS (Step 1)

**Perhitungan TOPSIS**

← Kembali

ID Case: 16

Jenis Bencana: Tanah Longsor

Tanggal Kejadian: 2020-01-03

Lokasi Kejadian: Jl. Embong Brantas II, Kidul Dalem, Klojen

Sektor Bangunan: Plengsengan & pos bermain

1 - Matriks Keputusan | 2 - Pembagi | 3 - Normalisasi Terbobot | 4 - Maximum & Minimum | 5 - Pemisahan | 6 - Preferensi

**Nilai Pembagi**

ID Kriteria	Nama Kriteria	Nilai Pembagi
1	Kondisi Bangunan	3.3166247903554
2	Kondisi Struktur Bangunan	3.3166247903554
3	Kondisi Fisik Bangunan	3.3166247903554
4	Fungsi Bangunan	3.3166247903554
5	Kondisi Penunjang Lainnya	3.3166247903554

Copyright © Penentuan Kerusakan Sektor Pasca Bencana Alam

Gambar 4. 28 Halaman Perhitungan TOPSIS (Step 2)

**KERUSAKAN SEKTOR**

HOME /

- DASHBOARD
- Data User
- Info >
- Penentuan >
- Profil

amilafadh

### Perhitungan TOPSIS

← Kembali

ID Case

16

Jenis Bencana

Tanah Longsor

Tanggal Kejadian

2020-01-03

Lokasi Kejadian

Jl. Embong Brantas II, Kidul Dalam, Klojen

Sektor Bangunan

Plengsengan & pos bermain

1 - Matriks Keputusan
2 - Pembagi
3 - Normalisasi Terbobot
4 - Maximum & Minimum
5 - Pemisahan
6 - Preferensi

Matriks Normalisasi Terbobot

ID Alternatif	Nama Alternatif	ID Kriteria	Nama Kriteria	ID Bobot	Value	Nilai	Keterangan	Normalisasi	Terbobot
1	Rusak Ringan	1	Kondisi Bangunan	1	0.456	1	Ringan	0.30151134457776363	0.13748917312746023
2	Rusak Sedang	1	Kondisi Bangunan	1	0.456	1	Ringan	0.30151134457776363	0.13748917312746023
3	Rusak Berat	1	Kondisi Bangunan	1	0.456	3	Berat	0.9045340337332909	0.41246751938238063
1	Rusak Ringan	2	Kondisi Struktur Bangunan	2	0.256	1	Ringan	0.30151134457776363	0.07718690421190749
2	Rusak Sedang	2	Kondisi Struktur Bangunan	2	0.256	1	Ringan	0.30151134457776363	0.07718690421190749
3	Rusak Berat	2	Kondisi Struktur Bangunan	2	0.256	3	Berat	0.9045340337332909	0.23156071263572248
1	Rusak Ringan	3	Kondisi Fisik Bangunan	3	0.156	1	Ringan	0.30151134457776363	0.047035769754131124
2	Rusak Sedang	3	Kondisi Fisik Bangunan	3	0.156	1	Ringan	0.30151134457776363	0.047035769754131124
3	Rusak Berat	3	Kondisi Fisik Bangunan	3	0.156	3	Berat	0.9045340337332909	0.14110730926239337
1	Rusak Ringan	4	Fungsi Bangunan	4	0.09	1	Ringan	0.30151134457776363	0.027136021011998725
2	Rusak Sedang	4	Fungsi Bangunan	4	0.09	1	Ringan	0.30151134457776363	0.027136021011998725
3	Rusak Berat	4	Fungsi Bangunan	4	0.09	3	Berat	0.9045340337332909	0.08140806303599618
1	Rusak Ringan	5	Kondisi Penunjang Lainnya	5	0.04	1	Ringan	0.30151134457776363	0.012060453783110546
2	Rusak Sedang	5	Kondisi Penunjang Lainnya	5	0.04	1	Ringan	0.30151134457776363	0.012060453783110546
3	Rusak Berat	5	Kondisi Penunjang Lainnya	5	0.04	3	Berat	0.9045340337332909	0.036181361349331635

Copyright © Penentuan Kerusakan Sektor Pasca Bencana Alam

Gambar 4. 29 Halaman Perhitungan TOPSIS (Step 3)

Perhitungan TOPSIS

← Kembali

ID Case: 16

Jenis Bencana: Tanah Longsor

Tanggal Kejadian: 2020-01-03

Lokasi Kejadian: Jl. Embong Brantas II, Kidul Dalam, Klojen

Sektor Bangunan: Plengsengan & pos bermain

1 - Matriks Keputusan   2 - Pembagi   3 - Normalisasi Terbobot   4 - Maximum & Minimum   5 - Pemisahan   6 - Preferensi

Nilai Maximum & Minimum

ID Kriteria	Nama Kriteria	Maximum (SIP)	Minimum (SIN)
1	Kondisi Bangunan	0.41246751938238063	0.13748917312746023
2	Kondisi Struktur Bangunan	0.23156071263572248	0.07718690421190749
3	Kondisi Fisik Bangunan	0.14110730926239337	0.047035769754131124
4	Fungsi Bangunan	0.08140806303599618	0.027136021011998725
5	Kondisi Penunjang Lainnya	0.036181361349331635	0.012060453783110546

Copyright © Penentuan Kerusakan Sektor Pasca Bencana Alam

Gambar 4. 30 Halaman Perhitungan TOPSIS (Step 4)

Perhitungan TOPSIS

← Kembali

ID Case: 16

Jenis Bencana: Tanah Longsor

Tanggal Kejadian: 2020-01-03

Lokasi Kejadian: Jl. Embong Brantas II, Kidul Dalam, Klojen

Sektor Bangunan: Plengsengan & pos bermain

1 - Matriks Keputusan   2 - Pembagi   3 - Normalisasi Terbobot   4 - Maximum & Minimum   5 - Pemisahan   6 - Preferensi

Nilai Pemisahan

ID Alternatif	Positif (S+)	Negatif (S-)
1 - Rusak Ringan	0.3343966072033191	0
3 - Rusak Berat	0	0.3343966072033191
2 - Rusak Sedang	0.3343966072033191	0

Copyright © Penentuan Kerusakan Sektor Pasca Bencana Alam

Gambar 4. 31 Halaman Perhitungan TOPSIS (Step 5)

The screenshot shows the 'Perhitungan TOPSIS' page. On the left is a blue sidebar with a menu containing 'DASHBOARD', 'Data User', 'Info', 'Penentuan', and 'Profil'. The main content area has a header 'Perhitungan TOPSIS' and a 'Kembali' button. Below are input fields for: ID Case (16), Jenis Bencana (Tanah Longsor), Tanggal Kejadian (2020-01-03), Lokasi Kejadian (Jl. Embong Brantas II, Kidul Dalem, Klojen), and Sektor Bangunan (Plengsengan & pos bermain). A progress bar shows steps 1-6, with step 6 'Preferensi' selected. Below the progress bar is a table titled 'Nilai Preferensi (V)'. At the bottom, there is a copyright notice: 'Copyright © Penentuan Kerusakan Sektor Pasca Bencana Alam'.

Ranking	Nilai V	Nama Alternatif
1	1	Rusak Berat
2	0	Rusak Sedang
3	0	Rusak Ringan

Gambar 4. 32 Halaman Perhitungan TOPSIS (Step 6)

Gambar 4.27 hingga 4.32 merupakan halaman untuk menampilkan hasil tiap langkah dari perhitungan TOPSIS yang dilakukan. Langkah-langkah tersebut meliputi matriks keputusan awal, nilai pembagi, matriks ternormalisasi dan terbobot, nilai maksimum dan minimum, nilai pemisahan, serta nilai preferensi.

The screenshot shows the 'Hasil Input Data Kejadian' page. It features a table titled 'Hasil Datatables' with columns: ID Case, Jenis Bencana, Tanggal Kejadian, Lokasi Kejadian, Sektor Bangunan, Result, and Keterangan. The table contains 7 rows of incident data. At the top of the table area, there is a 'Show 10 entries' dropdown and a search box.

ID Case	Jenis Bencana	Tanggal Kejadian	Lokasi Kejadian	Sektor Bangunan	Result	Keterangan
16	Tanah Longsor	2020-01-03	Jl. Embong Brantas II, Kidul Dalem, Klojen	Plengsengan & pos bermain	3	Rusak Berat
17	Pohon Tumbang	2020-01-04	Jl. Raya Walet, Sukun, Sukun	warung	2	Rusak Sedang
18	Tanah Longsor	2020-01-07	Jl. Kanjuruhan IV no. 3, Tlogomas, Lowokwaru	Plengsengan	3	Rusak Berat
19	Tanah Longsor	2020-01-16	Jl. Kanjuruhan IV no. 3, Tlogomas, Lowokwaru	Plengsengan	3	Rusak Berat
20	Tanah Longsor	2020-01-21	Jl. Embong Brantas II, Kidul Dalem, Klojen	Plengsengan & pos bermain	3	Rusak Berat
21	Pohon Tumbang	2020-01-22	Jl. Ir. H. Juanda Gg. IX, Jodipan, Blimbing	rumah	1	Rusak Ringan

Gambar 4. 33 Halaman Hasil

Gambar 4.33 merupakan halaman untuk menampilkan hasil penentuan tingkat kerusakan sektor bangunan tiap kasus kerusakan, berdasarkan perhitungan TOPSIS yang dilakukan. Halaman ini berisi tabel informasi yang terdiri dari kolom *id case*, jenis bencana, tanggal kejadian, lokasi kejadian, sektor bangunan, *result*, dan keterangan.

The screenshot displays a web application interface for 'KERUSAKAN SEKTOR'. The main content area is titled 'Profil' and contains a form for updating user information. The form fields are as follows:

Field	Value
Username	amilafadh
Nama Lengkap	amila fadhila
Email	amila@gmail.com
Password	*****

An 'Update' button is located at the bottom right of the form. The left sidebar menu includes: DASHBOARD, Data User, Info, Penentuan, and Profil. The top right corner shows the user's name 'amilafadh'.

Gambar 4. 34 Halaman Profil

Gambar 4.22 merupakan halaman profil. Pada halaman ini terdapat *form* yang berupa informasi data akun *user* yang saat ini digunakan.

#### 4.1.2 Implementasi *Forward Propagation in Neural Network*

Pembuatan sistem untuk pemodelan *machine learning* dilakukan menggunakan metode *forward propagation in neural network*. Sistem yang dibuat, dibangun menggunakan bahasa pemrograman Python. Implementasi dimulai dengan memasukkan *data training* yang berupa hasil metode TOPSIS ke dalam sistem. *Data training* tersebut diambil secara langsung dari *database* sistem metode TOPSIS. Data yang diambil antara lain data kasus pada tabel *cases*, matriks keputusan yang dibuat, serta hasil akhir perhitungan metode TOPSIS.

```

Deklarasi
  host      : VARCHAR
  user      : VARCHAR
  password  : VARCHAR
  database  : VARCHAR
  records   : array of INTEGER
  records2  : array of INTEGER
  records3  : array of INTEGER
Deskripsi
  host      <- "localhost"
  user      <- "root"
  password  <- ""
  database  <- "sector_damage"
  IF connect = true THEN
  records2  <- execute("select id_case from cases")
  records   <- execute("select * from matrix_keputusan")
  records3  <- execute("select result from v_result")
  WRITE (records, records2, records3)
End

```

Gambar 4. 35 Pseudocode Impor Data

Data hasil metode TOPSIS yang telah terbentuk kemudian di-*training* menggunakan metode *forward propagation in neural network* (FPNN) untuk membangun model *machine learning*. Dalam membuat sistem metode FPNN, digunakan *package library* “keras” yang ada pada “tensorflow”, untuk mempermudah dalam membangun model *machine learning*. Pada *input layer*, terdapat 5 neuron *input* sesuai dengan jumlah kriteria yang digunakan pada penelitian ini. Sedangkan pada *output layer*, terdapat 4 neuron yang bernilai 0, 1, 2, dan 3. Penentuan jumlah neuron pada *output layer* tersebut sesuai dengan jumlah

alternatif yang digunakan, dimana hasil 0 menunjukkan bahwa data tersebut tidak terdefinisi, hasil 1 menunjukkan rusak ringan, hasil 2 menunjukkan rusak sedang, serta hasil 3 menunjukkan rusak berat.

```

Deklarasi model
  neuron      : INTEGER
  input_shape : INTEGER
  activation  : CHAR
  optimizer   : CHAR
  loss        : FUNCTION
  metrics     : CHAR
Deskripsi
  READ (neuron, input_shape, activation, optimizer,
        Loss, metrics, batch_size, epoch)
  #input layer & hidden layer
  neuron      <- 5
  input_shape <- 5
  activation  <- 'relu'
  #output layer
  neuron      <- 4
  activation  <- 'sigmoid'
  #loss function & akurasi
  optimizer   <- 'adam'
  loss        <- SparseCategoricalCrossentropy()
  metrics     <- 'accuracy'
  WRITE (predicted data)
End

```

Gambar 4. 36 Pseudocode Forward Propagation in Neural Network

## 4.2 Uji Coba

Tahap uji coba yang dilakukan pada penelitian ini meliputi pengolahan data dengan metode TOPSIS, pengujian data dengan *forward propagation in neural network*, serta perhitungan akurasi.

### 4.2.1 Pengolahan Data dengan Metode TOPSIS

Data yang digunakan pada pengolahan data dengan metode TOPSIS merupakan data yang telah melalui tahap *preprocessing*, yang telah dijabarkan pada bab 3 sebelumnya. Data mentah yang didapatkan dari BPBD Kota Malang dikonversi ke dalam satuan angka, sesuai dengan kriteria yang digunakan pada penelitian ini. Data hasil *preprocessing* dapat dilihat pada tabel 4.1, yang selengkapnya ada pada lampiran 1.

Tabel 4. 1 Data Hasil *Preprocessing*

No.	Jenis Bencana	Tanggal kejadian	Lokasi	Sektor	K1	K2	K3	K4	K5
1	Tanah Longsor	03/01/2020	Jl. Embong Brantas II, Kidul Dalem, Klojen	Plengsengan & pos bermain	3	3	3	3	3
2	Pohon Tumbang	04/01/2020	Jl. Raya Walet, Sukun, Sukun	warung	2	2	2	2	1
3	Tanah Longsor	07/01/2020	Jl. Kanjuruhan IV no. 3, Tlogomas, Lowokwaru	Plengsengan	3	3	3	2	2
4	Tanah Longsor	16/01/2020	Jl. Kanjuruhan IV no. 3,	plengsengan	3	3	3	3	2



No.	Jenis Bencana	Tanggal kejadian	Lokasi	Sektor	K1	K2	K3	K4	K5
			Tlogomas, Lowokwaru						
5	Tanah Longsor	21/01/2020	Jl. Embong Brantas II, Kidul Dalem, Klojen	Plengsengan & pos bermain	3	3	3	2	3
6	Pohon Tumbang	22/01/2020	Jl. Ir. H. Juanda Gg. IX, Jodipan, Blimbing	rumah	1	1	1	2	1
7	Pohon Tumbang	24/01/2020	Jl. Embong Brantas, Jodipan, Blimbing	rumah	1	1	2	1	1
8	Pohon Tumbang	25/01/2020	Jl. Bromo Gg. II no. 8, Oro-oro dowo, Klojen	rumah	2	2	2	2	1
9	Tanah Longsor	26/01/2020	Jl. MT. Haryono Gg.21, Dinoyo, Lowokwaru	plataran rumah	2	2	2	2	2
10	Tanah Longsor	26/01/2020	Jl. MT. Haryono XI, Dinoyo, Lowokwaru	plengsengan	3	3	3	3	2

Data yang telah melalui tahap *preprocessing* tersebut kemudian dimasukkan ke dalam sistem yang telah dibuat. Data hasil *preprocessing* kemudian diolah dan dihitung menggunakan metode TOPSIS, untuk menentukan tingkat kerusakan dari masing-masing kasus kejadian yang ada. Setiap kasus kejadian yang ada ditentukan

tingkat kerusakannya, apakah rusak ringan, rusak sedang, atau rusak berat. Hasil pengolahan data menggunakan metode TOPSIS dapat dilihat pada tabel 4.2, yang selengkapnya ada pada lampiran 2.

Tabel 4. 2 Data Hasil Metode TOPSIS

No.	Jenis Bencana	Tanggal kejadian	Lokasi	Sektor	K1	K2	K3	K4	K5	Result	Keterangan
1	Tanah Longsor	03/01/2020	Jl. Embong Brantas II, Kidul Dalem, Klojen	Plengsengan & pos bermain	3	3	3	3	3	3	Rusak Berat
2	Pohon Tumbang	04/01/2020	Jl. Raya Walet, Sukun, Sukun	warung	2	2	2	2	1	2	Rusak Sedang
3	Tanah Longsor	07/01/2020	Jl. Kanjuruhan IV no. 3, Tlogomas, Lowokwaru	Plengsengan	3	3	3	2	2	3	Rusak Berat
4	Tanah Longsor	16/01/2020	Jl. Kanjuruhan IV no. 3, Tlogomas, Lowokwaru	plengsengan	3	3	3	3	2	3	Rusak Berat
5	Tanah Longsor	21/01/2020	Jl. Embong Brantas II, Kidul Dalem, Klojen	Plengsengan & pos bermain	3	3	3	2	3	3	Rusak Berat
6	Pohon Tumbang	22/01/2020	Jl. Ir. H. Juanda Gg. IX, Jodipan, Blimbing	rumah	1	1	1	2	1	1	Rusak Ringan
7	Pohon Tumbang	24/01/2020	Jl. Embong Brantas, Jodipan, Blimbing	rumah	1	1	2	1	1	1	Rusak Ringan

No.	Jenis Bencana	Tanggal kejadian	Lokasi	Sektor	K1	K2	K3	K4	K5	Result	Keterangan
8	Pohon Tumbang	25/01/2020	Jl. Bromo Gg. II no. 8, Oro-oro dowo, Klojen	rumah	2	2	2	2	1	2	Rusak Sedang
9	Tanah Longsor	26/01/2020	Jl. MT. Haryono Gg.21, Dinoyo, Lowokwaru	plataran rumah	2	2	2	2	2	2	Rusak Sedang
10	Tanah Longsor	26/01/2020	Jl. MT. Haryono XI, Dinoyo, Lowokwaru	plengsengan	3	3	3	3	2	3	Rusak Berat

#### 4.2.2 Pengujian Data dengan *Forward Propagation in Neural Network*

Tahap selanjutnya, setelah melakukan pengolahan data dengan metode TOPSIS, yakni pengujian data menggunakan metode *forward propagation in neural network*. Data hasil metode TOPSIS yang telah dilakukan sebelumnya digunakan sebagai *data training* pada pengujian data. Tabel 4.3 merupakan *data training* yang berjumlah 100 data, yang dimasukkan ke dalam sistem *forward propagation in neural network*, yang selengkapnya terdapat pada lampiran 3.

Tabel 4. 3 Data *Training*

K1	K2	K3	K4	K5	Data Aktual
3	3	3	3	3	3
2	2	2	2	1	2
3	3	3	2	2	3
3	3	3	3	2	3
3	3	3	2	3	3
1	1	1	2	1	1
1	1	2	1	1	1
2	2	2	2	1	2
2	2	2	2	2	2
3	3	3	3	2	3

Sedangkan, *data testing* yang digunakan merupakan data baru yang nantinya di-*input*-kan langsung ke dalam sistem *forward propagation in neural*

*network* yang telah dibuat. Tabel 4.4 merupakan data *testing* yang digunakan, yang berjumlah 50 data, yang selengkapnya dapat dilihat pada lampiran 4.

Tabel 4. 4 Data *Testing*

No.	Jenis Bencana	Tanggal kejadian	Lokasi	Sektor	K1	K2	K3	K4	K5
1	Tanah longsor	20/03/2020	Jl. Aris Munandar Gg. IV RT.06/RW.03, Kelurahan Kiduldalem, Kecamatan Klojen	rumah	1	1	1	2	1
2	pohon tumbang	20/03/2020	Jl. Tenaga Utara RW.05, Kelurahan Blimbing, Kecamatan Blimbing	rumah	1	1	1	2	1
3	Tanah longsor	26/03/2020	Jl. Kalimosodo XI No.42 RT.06/RW.06, Kelurahan Polehan, Kecamatan Blimbing	rumah	2	2	3	3	2
4	Tanah longsor	26/03/2020	Jl. Raya Candi VI D2, Kelurahan Karangbesuki, Kecamatan Sukun	plengsengan	3	3	3	3	2

No.	Jenis Bencana	Tanggal kejadian	Lokasi	Sektor	K1	K2	K3	K4	K5
5	Tanah longsor	29/03/2020	Jl. Simpang sukun 54A, Kelurahan Sukun, Kecamatan Sukun	rumah	2	2	3	3	2
6	Tanah longsor	30/03/2020	Jl. Tapak Siring IV, Kel. Samaan, Kec. Klojen	rumah	3	3	3	3	3
7	Tanah longsor	29/03/2020	Jl. Gadang Gg. 17 <sup>B</sup> Rt.05/Rw.03, Kelurahan Gadang, Kecamatan Sukun	rumah	2	2	2	3	1
8	Tanah longsor	02/04/2020	Jl. Mergosono Gg. IX No. 11 RT.06/Rw.01, Kelurahan Mergosono, Kecamatan Kedungkandang	rumah	2	2	2	3	2
9	Tanah longsor	02/04/2020	Rumah Bapak Komar Jl. Kebalen Wetan Gg. Mawar No.31 Rt.09./Rw.04, Kelurahan	plengsengan	3	3	3	3	2

No.	Jenis Bencana	Tanggal kejadian	Lokasi	Sektor	K1	K2	K3	K4	K5
			Kotalama, Kecamatan Kedungkandang						
10	Tanah longsor	08/04/2020	Jl. Muharto Gg.VB RT.13/RW.08, Kelurahan Kotalama, Kecamatan Kedungkandang	rumah	3	3	3	3	2

Dalam pengujian data menggunakan metode *forward propagation in neural network*, dibentuk 5 model yang dibedakan berdasarkan jumlah *hidden layer* dan jumlah *epoch* atau iterasi berapa kali algoritma *neural network* bekerja melewati seluruh *data training* yang ada. Lima model tersebut antara lain model A, model B, model C, model D, dan model E seperti yang ditunjukkan oleh tabel 4.5.

Tabel 4. 5 Model FPNN

Model	<i>Epoch</i>	<i>Hidden Layer</i>
A1	25	1
A2		2
A3		3
A4		4
A5		5



Model	<i>Epoch</i>	<i>Hidden Layer</i>
B1	50	1
B2		2
B3		3
B4		4
B5		5
C1	75	1
C2		2
C3		3
C4		4
C5		5
D1	100	1
D2		2
D3		3
D4		4
D5		5
E1	125	1
E2		2
E3		3
E4		4
E5		5

a. Model A1

Model A1 merupakan model *forward propagation in neural network* (FPNN) yang memiliki *epoch* sebanyak 25 dan *hidden layer* sejumlah 1. Data *testing* yang telah ditentukan sebelumnya, akan diolah menggunakan model ini untuk mengetahui hasil prediksi metode FPNN. Hasil dari perhitungan menggunakan model A1, didapatkan bahwa akurasi di *epoch* ke-25 adalah sebesar 49% dengan *loss* sebesar 1,0332.

```
Epoch 25/25
10/10 [=====] - 0s 1ms/step -
loss: 1.0332 - accuracy: 0.4900
```

Pada model yang terbentuk, nilai bobot dan nilai bias awal akan ditentukan secara random, hingga didapatkan nilai bobot dan bias dengan *loss function* terendah untuk tiap *hidden layer* dan *output layer*. Nilai bobot dan nilai bias akhir yang terbentuk pada masing-masing *hidden layer* dan *output layer* pada model A1 antara lain sebagai berikut.

```
----- layer 1 -----
weights
[[-0.19608231 -0.541612      0.5657446  -0.38080713  -
0.39272624]
 [ 0.00333235  0.3195171  -0.29036513  0.44735342
0.41717628]
 [ 0.56890094  0.7443153   0.04125895  0.7045349
0.5070805 ]
 [-0.20732702  0.5490416  -0.40930015 -0.13133761  -
0.00288861]
```

```

[ 0.40569037 -0.25413644 -0.16713864 0.2067038
0.17956829]]
biases
[ 0.05092429 0.16317523 -0.04720425 -0.10214657
0.0236436 ]
----- layer 2 -----
weights
[[ 0.07466228 0.4400676 0.79408586 0.00954074]
[-0.6038612 -0.1083618 -0.18805638 0.14103708]
[ 0.25850806 -0.15111752 0.466314 -0.35773054]
[ 0.11486309 -0.897497 0.12966695 -0.4127169 ]
[-0.13321128 0.5062318 -0.64094913 0.6711021 ]]
biases
[-0.15762256 0.0181229 0.13784212 0.08996844]

```

Hasil perhitungan prediksi metode FPNN model A1 sendiri adalah sebagai berikut.

```

1/1 [=====] - 0s 70ms/step
1 X=[1 1 1 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
2 X=[1 1 1 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 35ms/step
3 X=[2 2 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 30ms/step
4 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
5 X=[2 2 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
6 X=[3 3 3 3 3], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
7 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
8 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step

```

```

9 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 25ms/step
10 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 31ms/step
11 X=[2 2 2 3 3], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 27ms/step
12 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
13 X=[1 1 1 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 30ms/step
14 X=[1 1 1 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
15 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
16 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
17 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 29ms/step
18 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
19 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
20 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 28ms/step
21 X=[1 1 1 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
22 X=[1 1 1 1 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
23 X=[1 1 1 1 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
24 X=[1 1 1 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 18ms/step
25 X=[1 1 1 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
26 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
27 X=[1 1 1 1 1], Predicted=(array([3], dtype=int64),)

```

```

1/1 [=====] - 0s 21ms/step
28 X=[1 1 1 1 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 27ms/step
29 X=[1 1 1 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
30 X=[1 1 1 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
31 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 26ms/step
32 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
33 X=[1 1 1 2 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
34 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 29ms/step
35 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 34ms/step
36 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
37 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
38 X=[1 1 1 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 18ms/step
39 X=[1 1 1 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 30ms/step
40 X=[1 1 1 2 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
41 X=[1 1 1 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
42 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
43 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 33ms/step
44 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 34ms/step
45 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 36ms/step

```

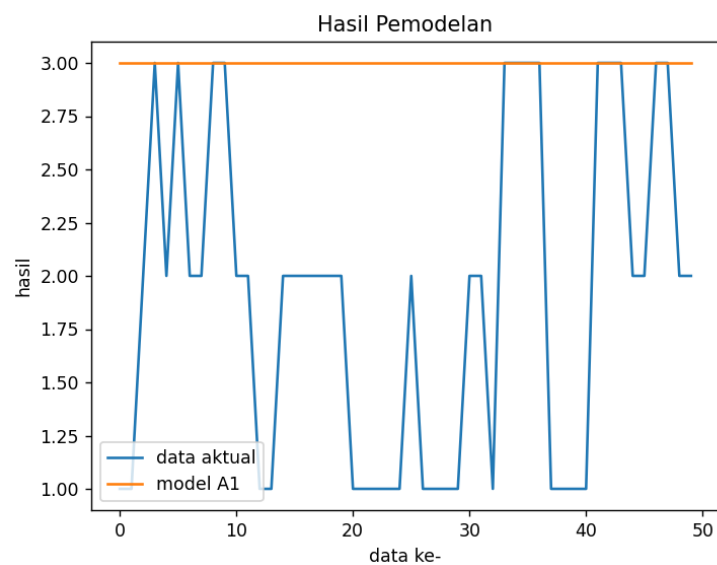
```

46 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
47 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 32ms/step
48 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 26ms/step
49 X=[2 2 1 2 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 18ms/step
50 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)

Process finished with exit code 0

```

Gambar 4.37 menunjukkan grafik perbandingan antara data aktual dengan data hasil prediksi dari model A1. Garis warna biru merupakan data aktual dan garis warna oranye merupakan data hasil prediksi. Ada 50 data *testing* yang digunakan, yang diolah menggunakan model A untuk mengetahui hasil prediksinya. *Output* yang dihasilkan, baik itu aktual maupun prediksi, berupa angka 1 (rusak ringan), 2 (rusak sedang), dan 3 (rusak berat).



Gambar 4. 37 Grafik Hasil Model A1

## b. Model A2

Model A2 merupakan model *forward propagation in neural network* (FPNN) dengan *epoch* 25 dan *hidden layer* sejumlah 2. Hasil dari perhitungan menggunakan model A2, didapatkan bahwa akurasi di *epoch* ke-25 adalah sebesar 50% dengan *loss* sebesar 1,1350.

```
Epoch 25/25
10/10 [=====] - 0s 1ms/step -
loss: 1.1350 - accuracy: 0.5000
```

Pada model A2, nilai bobot dan nilai bias awal akan ditentukan secara random, hingga didapatkan nilai bobot dan bias dengan *loss function* terendah untuk tiap *hidden layer* dan *output layer*. Berikut merupakan nilai bobot dan nilai bias akhir yang terbentuk pada model A2.

```
----- layer 1 -----
weights
[[-0.20601946 -0.36072567 -0.08661732 -0.32040697 -
0.25220504]
 [-0.38067192 -0.04694032 0.2725701 0.08465242 -
0.7229021 ]
 [ 0.3199886 -0.42465687 0.2685821 -0.27522284
0.3608157 ]
 [-0.68181765 0.34333792 0.5250421 0.15925944
0.33154356]
 [ 0.49735677 0.591992 -0.583385 0.5348147 -
0.06619985]]
biases
[ 0. 0.11716054 -0.10672643 -0.02609755 -
0.02099646]
----- layer 2 -----
weights
```

```

[[-0.5166278 -0.6033248 -0.06773168 0.19630873 -
0.23498923]
 [-0.35633484 -0.03585147 0.36122727 -0.41825578 -
0.568314 ]
 [-0.37545916 0.33227426 0.03705076 0.6123665
0.37407777]
 [-0.69170845 0.15601355 0.87673426 0.52151465
0.15216088]
 [ 0.5856583 -0.10996213 -0.20076041 -0.0115077
0.5650906 ]]
biases
[ 0.          0.21005696 -0.06241243 -0.16493833 -
0.00679066]
----- layer 3 -----
weights
[[-0.73456496 0.6725613 0.40780425 0.7908002 ]
 [-0.21444276 0.66072536 0.24954045 0.9588276 ]
 [ 0.08047212 0.37635806 0.4665551 -0.01138204]
 [ 0.4065486 -0.5586923 0.25746682 -0.4527011 ]
 [ 0.31134585 -0.86458766 0.52471036 0.5185578 ]]
biases
[-0.20042151 0.11589187 -0.08489127 0.17752376]

```

Berikut merupakan hasil perhitungan prediksi metode FPNN dari model A2.

```

1/1 [=====] - 0s 82ms/step
1 X=[1 1 1 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
2 X=[1 1 1 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 25ms/step
3 X=[2 2 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 25ms/step
4 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 34ms/step
5 X=[2 2 3 3 2], Predicted=(array([3], dtype=int64),)

```



```

1/1 [=====] - 0s 19ms/step
6 X=[3 3 3 3 3], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
7 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
8 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
9 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
10 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 28ms/step
11 X=[2 2 2 3 3], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 24ms/step
12 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
13 X=[1 1 1 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
14 X=[1 1 1 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
15 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
16 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 27ms/step
17 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 28ms/step
18 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
19 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
20 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
21 X=[1 1 1 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
22 X=[1 1 1 1 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
23 X=[1 1 1 1 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 33ms/step

```

```

24 X=[1 1 1 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 18ms/step
25 X=[1 1 1 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 26ms/step
26 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 27ms/step
27 X=[1 1 1 1 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
28 X=[1 1 1 1 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 33ms/step
29 X=[1 1 1 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 27ms/step
30 X=[1 1 1 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 29ms/step
31 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 26ms/step
32 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 27ms/step
33 X=[1 1 1 2 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 26ms/step
34 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
35 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
36 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
37 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
38 X=[1 1 1 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
39 X=[1 1 1 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 25ms/step
40 X=[1 1 1 2 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 21ms/step
41 X=[1 1 1 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
42 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)

```

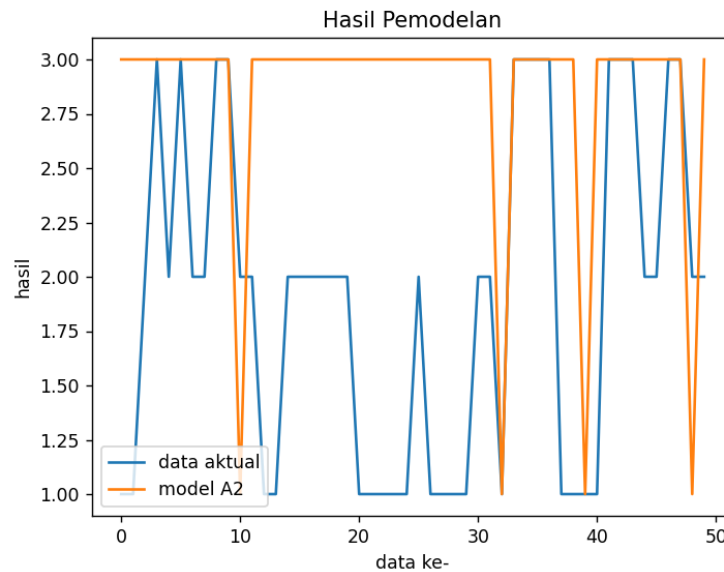
```

1/1 [=====] - 0s 37ms/step
43 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 29ms/step
44 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 29ms/step
45 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
46 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
47 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
48 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 29ms/step
49 X=[2 2 1 2 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 22ms/step
50 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)

Process finished with exit code 0

```

Gambar 4.38 menunjukkan grafik perbandingan antara data aktual dengan data hasil prediksi dari model A2. Garis warna biru merupakan data aktual dan garis warna oranye merupakan data hasil prediksi. Ada 50 data *testing* yang digunakan, yang diolah untuk mengetahui hasil prediksinya. *Output* yang dihasilkan berupa angka 1 (rusak ringan), 2 (rusak sedang), dan 3 (rusak berat).



Gambar 4.38 Grafik Hasil Model A2

## c. Model A3

Model A3 adalah model FPNN dengan *epoch* 25 dan *hidden layer* sebanyak 3. Hasil dari perhitungan menggunakan model A3 adalah akurasi di *epoch* ke-25 sebesar 49% dengan *loss* sebesar 0,8110.

```
Epoch 25/25
10/10 [=====] - 0s 1ms/step -
loss: 0.8110 - accuracy: 0.4900
```

Nilai bobot dan nilai bias awal pada model A3 akan ditentukan secara random, hingga didapatkan nilai bobot dan bias dengan *loss function* terendah untuk tiap *hidden layer* dan *output layer*. Nilai bobot dan nilai bias akhir yang terbentuk pada model A3 adalah sebagai berikut.

```
----- layer 1 -----
weights
[[-0.20568742  0.26631406  0.36621198  0.10151122
 0.5293712 ]
```

```

[-0.3312516  0.02070669 -0.37795022  0.5989431
0.5519315 ]
[ 0.08866153  0.21401043 -0.17729953 -0.1952943
0.46572098]
[ 0.579269    0.01662633  0.5013343  -0.44730565
0.7244946 ]
[-0.7281595  -0.03982064  0.25298348 -0.80628043 -
0.33209652]]
biases
[-0.06790727  0.10324968  0.17922889 -0.03424028 -
0.10022192]
----- layer 2 -----
weights
[[-0.16557306 -0.04206212  0.2536831  -0.02065873 -
0.48411554]
[ 0.06429032  0.20178096  0.08164495 -0.56622416
0.5686084 ]
[ 0.7980323  0.71852654  0.23008025  0.20338303 -
0.0811426 ]
[ 0.09195018  0.00585723  0.24166012 -0.40487757
0.36058354]
[-0.21101506  0.8807016  -0.27357662  0.49606875 -
0.4687553 ]]
biases
[ 0.17942722  0.02594013  0.          -0.08744048  0.
]
----- layer 3 -----
weights
[[ 0.5415895  0.8239607  -0.35832444  0.4143893
0.11363733]
[-0.19056576  0.32720035  0.37110448  0.87958753 -
0.52044106]
[ 0.5828526  -0.38355398 -0.30640152 -0.30731604 -
0.36504537]
[-0.6818069  0.2540555  0.7310148  0.25704804
0.35648477]

```

```

[-0.7729429 -0.23272878 -0.26123732 -0.1347813 -
0.46956363]]
biases
[ 0.          0.09641018 -0.0923522   0.09811173  0.
]
----- layer 4 -----
weights
[[-0.42918873 -0.32517162 -0.5228396  -0.7670344 ]
 [-0.77218544  0.71113807  0.3141724  -0.46096727]
 [ 0.33188084 -0.40582335 -0.1850499   0.8648287 ]
 [-0.657989   -0.37073237 -0.19424087 -0.24315533]
 [-0.69671136 -0.5623504  -0.5861931   0.63046396]]
biases
[-0.12902151  0.03663122  0.08179057 -0.05735677]

```

Hasil perhitungan prediksi model A3 adalah sebagai berikut.

```

1/1 [=====] - 0s 79ms/step
1 X=[1 1 1 2 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 19ms/step
2 X=[1 1 1 2 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 20ms/step
3 X=[2 2 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
4 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
5 X=[2 2 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
6 X=[3 3 3 3 3], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
7 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
8 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
9 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 27ms/step
10 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)

```

```

1/1 [=====] - 0s 25ms/step
11 X=[2 2 2 3 3], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 30ms/step
12 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 27ms/step
13 X=[1 1 1 2 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 30ms/step
14 X=[1 1 1 2 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 26ms/step
15 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 28ms/step
16 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 30ms/step
17 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 29ms/step
18 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 27ms/step
19 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 30ms/step
20 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 30ms/step
21 X=[1 1 1 2 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 28ms/step
22 X=[1 1 1 1 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 27ms/step
23 X=[1 1 1 1 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 25ms/step
24 X=[1 1 1 2 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 19ms/step
25 X=[1 1 1 2 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 18ms/step
26 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
27 X=[1 1 1 1 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 27ms/step
28 X=[1 1 1 1 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 29ms/step

```

```
29 X=[1 1 1 2 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 26ms/step
30 X=[1 1 1 2 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 25ms/step
31 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 18ms/step
32 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
33 X=[1 1 1 2 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 19ms/step
34 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
35 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
36 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 18ms/step
37 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 17ms/step
38 X=[1 1 1 2 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 20ms/step
39 X=[1 1 1 2 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 19ms/step
40 X=[1 1 1 2 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 19ms/step
41 X=[1 1 1 2 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 19ms/step
42 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
43 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
44 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
45 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 25ms/step
46 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 25ms/step
47 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
```



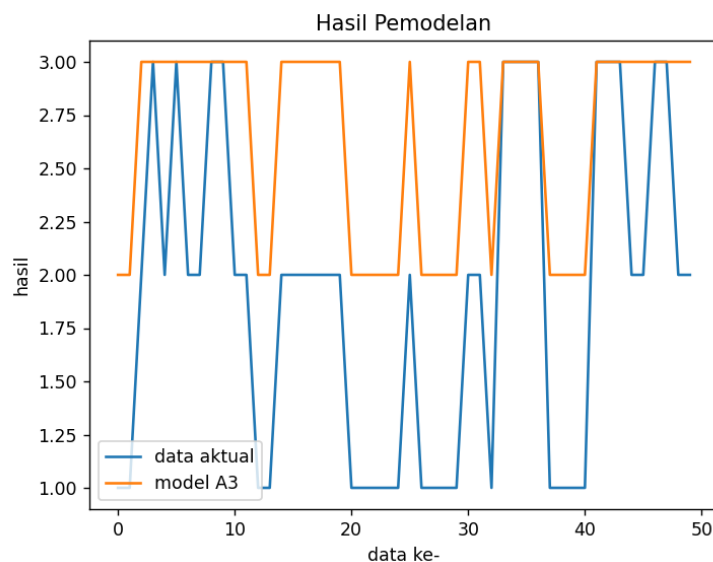
```

1/1 [=====] - 0s 24ms/step
48 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
49 X=[2 2 1 2 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
50 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)

Process finished with exit code 0

```

Gambar 4.39 menunjukkan grafik perbandingan antara data aktual dengan data hasil prediksi dari model A3. Garis warna biru merupakan data aktual dan garis warna oranye merupakan data hasil prediksi. 50 data *testing* diolah untuk mengetahui hasil perhitungan prediksi yang dilakukan. *Output* yang dihasilkan berupa angka 1 (rusak ringan), 2 (rusak sedang), dan 3 (rusak berat).



Gambar 4. 39 Grafik Hasil Model A3

## d. Model A4

Model A4 sendiri merupakan model FPNN dengan *epoch* 25 dan *hidden layer* sebanyak 4. Hasil dari perhitungan menggunakan model A4 adalah akurasi di *epoch* ke-25 sebesar 50% dengan *loss* sebesar 0,8578.

```
Epoch 25/25
10/10 [=====] - 0s 1ms/step -
loss: 0.8578 - accuracy: 0.5000
```

Nilai bobot dan nilai bias awal akan ditentukan secara random, hingga didapatkan nilai bobot dan bias dengan *loss function* terendah. Berikut merupakan nilai bobot dan nilai bias akhir pada masing-masing *hidden layer* dan *output layer* dari model A4.

```
----- layer 1 -----
weights
[[ 0.5112891 -0.18933749 -0.09393317  0.7482599
 0.00434792]
 [-0.71085507 -0.38784185  0.87854636 -0.09997217 -
 0.46009904]
 [-0.29331076 -0.44696584 -0.0783076  -0.34278777
 0.0854681 ]
 [ 0.39210755 -0.5411961  0.6491586  0.34354547 -
 0.3667674 ]
 [ 0.55981606 -0.34381014  0.7340924  -0.51305753
 0.70834816]]
biases
[ 0.08565938  0.          -0.07460982 -0.09286026 -
 0.10506616]
----- layer 2 -----
weights
[[-0.5376672  -0.74986315  0.16271281 -0.22102721 -
 0.49334568]
```

```

[-0.01334244 -0.15333188 -0.66034603  0.24559712  0.56636
]
[ 0.47337902  0.10806663 -0.76910305  0.42480138
0.46667004]
[ 0.80715257 -0.44302967 -0.3166314  -0.03288043
0.24693345]
[-0.14598206  0.49569592 -0.22657675  0.7026027
0.7182593 ]]
biases
[-0.07268302 -0.00142431  0.          0.08589169 -
0.07458637]
----- layer 3 -----
weights
[[-0.49464995  0.6261759  0.6687888  0.14581162
0.47933012]
[ 0.11731666 -0.7476245  -0.29076865 -0.6326698
0.37767828]
[-0.43961087  0.7615423  -0.0102343  0.33655596
0.763294  ]
[-0.15090483 -0.37009543  0.52933234  0.08816707 -
0.1016119  ]
[-0.6182623  0.5299983  -0.16371293 -0.58661366 -
0.34571317]]
biases
[ 0.          -0.09050573  0.06162438  0.          -
0.07427773]
----- layer 4 -----
weights
[[-0.21054077  0.6359415  0.5876236  -0.43495587 -
0.6149979  ]
[ 0.38173395 -0.07405768 -0.56012905  0.85601765
0.45584822]
[-0.06531716  0.8833617  0.85080796  0.0907572  -
0.5139755  ]
[-0.00585377 -0.36274093 -0.09528399  0.5262246
0.05566847]

```

```

[ 0.75740385  0.75549936 -0.21019112  0.73731554 -
0.31443244]]
biases
[-0.10532901  0.08113329  0.04119924 -0.07020289 -
0.00885247]
----- layer 5 -----
weights
[[ 0.21284746 -0.71560264 -0.86036795  0.78811884]
 [-0.22074421  0.23711106  0.6654929  0.33009994]
 [-0.6521793  -0.19388169  0.5335134  -0.15370198]
 [-0.29218805 -0.61395127 -0.7158498  0.3127196 ]
 [ 0.7273332  -0.32609707  0.42520168 -0.7341612 ]]
biases
[-0.16483077  0.1645805  -0.01304846  0.03982877]

```

Berikut merupakan hasil perhitungan prediksi model A4.

```

1/1 [=====] - 0s 104ms/step
1 X=[1 1 1 2 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 20ms/step
2 X=[1 1 1 2 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 21ms/step
3 X=[2 2 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
4 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
5 X=[2 2 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
6 X=[3 3 3 3 3], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
7 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
8 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
9 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
10 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)

```

```
1/1 [=====] - 0s 22ms/step
11 X=[2 2 2 3 3], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 24ms/step
12 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
13 X=[1 1 1 2 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 19ms/step
14 X=[1 1 1 2 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 20ms/step
15 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
16 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 25ms/step
17 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 26ms/step
18 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
19 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
20 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
21 X=[1 1 1 2 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 25ms/step
22 X=[1 1 1 1 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 28ms/step
23 X=[1 1 1 1 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 26ms/step
24 X=[1 1 1 2 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 24ms/step
25 X=[1 1 1 2 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 26ms/step
26 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 26ms/step
27 X=[1 1 1 1 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 26ms/step
28 X=[1 1 1 1 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 27ms/step
```

```
29 X=[1 1 1 2 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 26ms/step
30 X=[1 1 1 2 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 28ms/step
31 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
32 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 27ms/step
33 X=[1 1 1 2 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 24ms/step
34 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
35 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 26ms/step
36 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 26ms/step
37 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 28ms/step
38 X=[1 1 1 2 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 24ms/step
39 X=[1 1 1 2 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 27ms/step
40 X=[1 1 1 2 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 25ms/step
41 X=[1 1 1 2 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 25ms/step
42 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 26ms/step
43 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
44 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
45 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
46 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 17ms/step
47 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
```

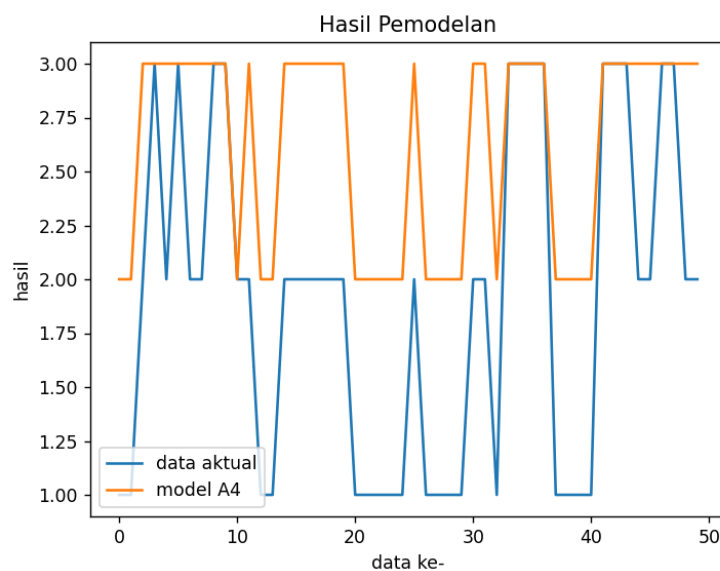
```

1/1 [=====] - 0s 20ms/step
48 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 18ms/step
49 X=[2 2 1 2 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
50 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)

Process finished with exit code 0

```

Garis warna biru merupakan data aktual dan garis warna oranye merupakan data hasil prediksi. 50 data *testing* diolah untuk mengetahui hasil perhitungan prediksi yang dilakukan. *Output* yang dihasilkan berupa angka 1 (rusak ringan), 2 (rusak sedang), dan 3 (rusak berat). Gambar 4.40 menunjukkan grafik perbandingan antara data aktual dengan data hasil prediksi dari model A4.



Gambar 4. 40 Grafik Hasil Model A4

## e. Model A5

Model A5 merupakan model FPNN yang memiliki *epoch* 25 dan *hidden layer* sebanyak 5. Hasil dari perhitungan menggunakan model A5 sendiri yakni akurasi di *epoch* ke-25 sebesar 66% dengan *loss* sebesar 1,1207.

```
Epoch 25/25
10/10 [=====] - 0s 1ms/step -
loss: 1.1207 - accuracy: 0.6600
```

Pada model ini, nilai bobot dan nilai bias awal juga akan ditentukan secara random, hingga didapatkan nilai bobot dan bias dengan *loss function* terendah. Nilai bobot dan nilai bias akhir pada masing-masing *hidden layer* dan *output layer* dari model A5 adalah sebagai berikut.

```
----- layer 1 -----
weights
[[ 0.64038146 -0.6966543  0.59040016  0.4422898 -
0.07588655]
 [ 0.6371937  -0.37770098  0.38786072  0.43037063 -
0.00966581]
 [-0.6525737  0.08166951  0.59183854 -0.08973332
0.35380703]
 [-0.6542419  -0.3888838  -0.60451114  0.82653993 -
0.12635203]
 [-0.5852543  -0.39922506 -0.45383048  0.4358748
0.6023221  ]]
biases
[-0.03782653  0.          0.07346807  0.11165783 -
0.08561338]
----- layer 2 -----
weights
```



```

[[-0.06366616 -0.7468768 -0.67575675 -0.40624604 -
0.5152412 ]
 [ 0.24576283 -0.43574 -0.6371534 -0.21088076 -
0.4856687 ]
 [ 0.3330083 0.6072628 0.42390138 -0.61403364
0.62051445]
 [-0.36613682 0.3859094 0.4135123 0.4277271 -
0.4434273 ]
 [-0.263691 -0.2663004 0.5668789 0.24844073
0.39345053]]
biases
[ 0. 0.08068044 -0.101207 0.11768877 -
0.02772431]
----- layer 3 -----
weights
[[ 0.07459062 0.59406507 0.66221833 0.7171544
0.00922078]
 [-0.34824136 0.2096882 0.27623358 -0.6120013 -
0.08999038]
 [ 0.38927972 0.62041277 -0.31542978 0.46960765 -
0.7531234 ]
 [-0.6071293 0.04947339 0.4184351 -0.2685559
0.11753088]
 [ 0.28306282 -0.7510909 0.02693106 -0.5676115
0.49648774]]
biases
[-0.05610684 -0.12401467 0.13073163 -0.02611374 0.
]
----- layer 4 -----
weights
[[-0.43516687 0.68943715 -0.08946897 0.42045966 -
0.25930554]
 [ 0.5876239 -0.17542315 0.43562138 0.73193806
0.03258479]
 [ 0.3114798 -0.0011363 -0.6021486 -0.80214 -
0.18405995]

```

```

[-0.57862127  0.53670454  0.68121266  0.291216
0.08106525]
 [ 0.16071695 -0.30721852 -0.01364809 -0.29727194 -
0.6635338  ]]
biases
 [ 0.13221774  0.00515863 -0.08027139 -0.14997485 -
0.02522219]
----- layer 5 -----
weights
[[-0.07793623  0.37948462 -0.6126488  0.6415747
0.15012749]
 [-0.41868514 -0.6688224  -0.19722538  0.48468435
0.05517578]
 [-0.16608131  0.68650234  0.67077786 -0.4085847  -
0.30675805]
 [-0.56932884 -0.5270101  0.23220089 -0.62134355 -
0.37056682]
 [ 0.5526655  0.732363  0.47745287 -0.39023417 -
0.43811366]]
biases
 [ 0.          0.16391322 -0.03115107  0.12189876 -
0.10198975]
----- layer 6 -----
weights
[[ 0.4722879  -0.6927386  0.78808284 -0.1880399 ]
 [-0.20527236  0.51480573  0.36300004  0.5008432 ]
 [ 0.43403727 -0.48481563 -0.4256858  -0.592675  ]
 [-0.7381878  0.88900036  0.51683176 -0.8237804 ]
 [ 0.54989135 -0.0095283  0.13961641  0.3468879  ]]
biases
[-0.21570551 -0.04936435  0.01110793  0.20018929]

```

Berikut merupakan hasil prediksi model A5.

```

1/1 [=====] - 0s 95ms/step
1 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 30ms/step

```

```
2 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 20ms/step
3 X=[2 2 3 3 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 23ms/step
4 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 25ms/step
5 X=[2 2 3 3 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 22ms/step
6 X=[3 3 3 3 3], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 26ms/step
7 X=[2 2 2 3 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 19ms/step
8 X=[2 2 2 3 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 27ms/step
9 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 30ms/step
10 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
11 X=[2 2 2 3 3], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 25ms/step
12 X=[2 2 2 3 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 30ms/step
13 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 23ms/step
14 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 23ms/step
15 X=[2 2 2 3 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 27ms/step
16 X=[2 2 2 3 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 25ms/step
17 X=[2 2 2 3 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 23ms/step
18 X=[2 2 2 3 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 30ms/step
19 X=[2 2 2 3 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 22ms/step
20 X=[2 2 2 3 1], Predicted=(array([1], dtype=int64),)
```

```

1/1 [=====] - 0s 20ms/step
21 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 26ms/step
22 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 21ms/step
23 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 29ms/step
24 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 24ms/step
25 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 20ms/step
26 X=[2 2 2 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 31ms/step
27 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 23ms/step
28 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 18ms/step
29 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 25ms/step
30 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 27ms/step
31 X=[2 2 2 3 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 19ms/step
32 X=[2 2 2 3 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 32ms/step
33 X=[1 1 1 2 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 32ms/step
34 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 29ms/step
35 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 37ms/step
36 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 27ms/step
37 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
38 X=[1 1 1 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 27ms/step

```

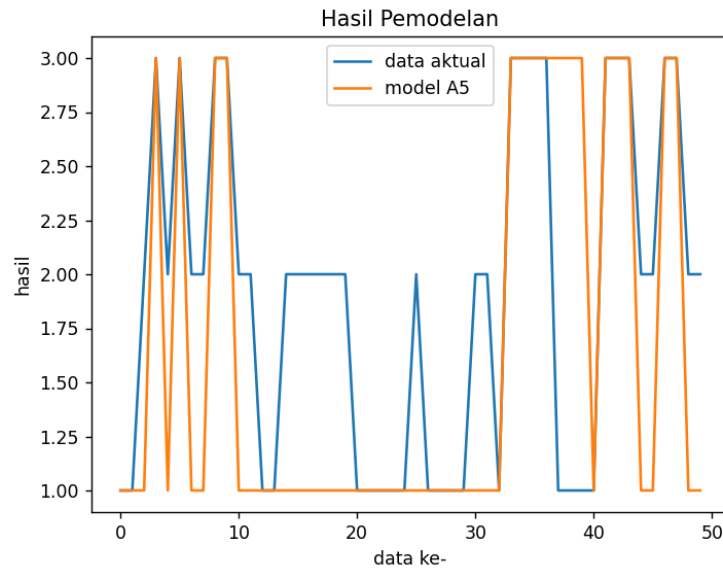
```

39 X=[1 1 1 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
40 X=[1 1 1 2 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 25ms/step
41 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 24ms/step
42 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 31ms/step
43 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 35ms/step
44 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
45 X=[2 2 2 3 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 24ms/step
46 X=[2 2 2 3 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 22ms/step
47 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
48 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
49 X=[2 2 1 2 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 24ms/step
50 X=[2 2 2 2 1], Predicted=(array([1], dtype=int64),)

Process finished with exit code 0

```

Garis warna biru merupakan data aktual dan garis warna oranye merupakan data hasil prediksi. 50 data *testing* diolah untuk mengetahui hasil perhitungan prediksi yang dilakukan. *Output* yang dihasilkan berupa angka 1 (rusak ringan), 2 (rusak sedang), dan 3 (rusak berat). Grafik perbandingan antara data aktual dengan data hasil prediksi dari model A5 ditunjukkan oleh gambar 4.41 sebagai berikut.



Gambar 4. 41 Grafik Hasil Model A5

f. Model B1

Model B1 merupakan model *forward propagation in neural network* (FPNN) dengan *epoch* sebanyak 50 dan *hidden layer* sejumlah 1. Data *testing* akan diolah menggunakan model ini untuk mengetahui hasil prediksinya. Hasil dari perhitungan menggunakan model B1 sendiri didapatkan bahwa akurasi di *epoch* ke-50 adalah sebesar 49% dengan *loss* sebesar 0,7967.

```
Epoch 50/50
10/10 [=====] - 0s 1ms/step -
loss: 0.7967 - accuracy: 0.4900
```

Nilai bobot dan nilai bias awal ditentukan secara random pada model ini, kemudian akan didapatkan nilai bobot dan bias dengan *loss function* terendah untuk masing-masing *hidden layer* dan *output layer*. Nilai bobot

dan nilai bias akhir yang terbentuk pada model B1 antara lain sebagai berikut.

```

----- layer 1 -----
weights
[[-0.23159328 -0.68946064 -0.21789838  0.21971938
 0.4660902 ]
 [-0.14876418  0.08813956  0.44307986  0.30462727 -
 0.3748235 ]
 [ 0.3969714  -0.27987966  0.51056886  0.10187647 -
 0.26211733]
 [ 0.12819995  0.81925637  0.29566282  0.6285195  -
 0.4236744 ]
 [-0.44852638 -0.02531764  0.1658646  0.3291071  -
 0.66761816]]
biases
[ 0.023491  0.38016558  0.06167474 -0.05547858  0.
 ]
----- layer 2 -----
weights
[[ 0.4009823  0.5458741 -0.10866511 -0.0389044 ]
 [-0.42557994  0.0800815 -0.19366479 -1.2158475 ]
 [-1.0783495  0.4876736  0.29029438  0.8960261 ]
 [ 0.29431322  0.49391496  0.74547625  0.6030121 ]
 [ 0.6777313  0.5811869 -0.7871751 -0.15225893]]
biases
[-0.22458841 -0.10883401  0.07663513  0.08988098]

```

Berikut merupakan hasil perhitungan prediksi metode FPNN pada model B1.

```

1/1 [=====] - 0s 77ms/step
1 X=[1 1 1 2 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 21ms/step
2 X=[1 1 1 2 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 27ms/step

```

```

3 X=[2 2 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 32ms/step
4 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
5 X=[2 2 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 25ms/step
6 X=[3 3 3 3 3], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 25ms/step
7 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
8 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 36ms/step
9 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 38ms/step
10 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 25ms/step
11 X=[2 2 2 3 3], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 27ms/step
12 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 18ms/step
13 X=[1 1 1 2 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 21ms/step
14 X=[1 1 1 2 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 21ms/step
15 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
16 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 25ms/step
17 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 32ms/step
18 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 33ms/step
19 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
20 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 29ms/step
21 X=[1 1 1 2 1], Predicted=(array([2], dtype=int64),)

```



```

1/1 [=====] - 0s 21ms/step
22 X=[1 1 1 1 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
23 X=[1 1 1 1 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
24 X=[1 1 1 2 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 30ms/step
25 X=[1 1 1 2 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 23ms/step
26 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
27 X=[1 1 1 1 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
28 X=[1 1 1 1 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 36ms/step
29 X=[1 1 1 2 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 29ms/step
30 X=[1 1 1 2 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 41ms/step
31 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 28ms/step
32 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 32ms/step
33 X=[1 1 1 2 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 25ms/step
34 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
35 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
36 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 27ms/step
37 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
38 X=[1 1 1 2 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 22ms/step
39 X=[1 1 1 2 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 20ms/step

```

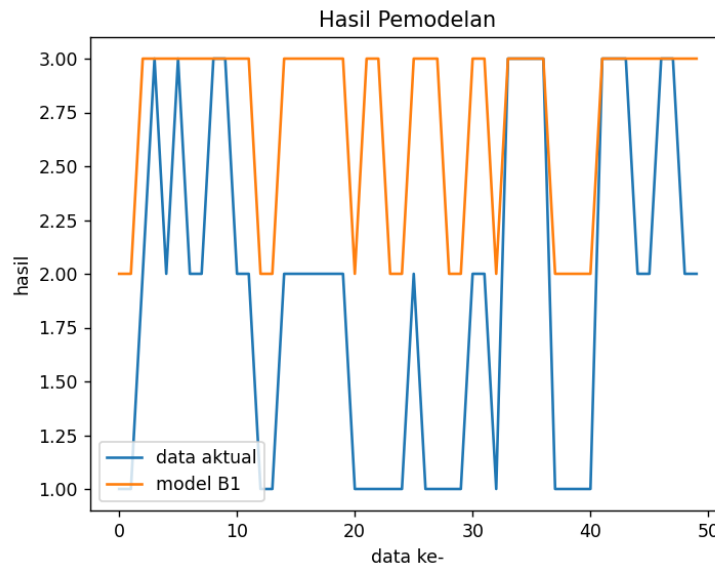
```

40 X=[1 1 1 2 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 20ms/step
41 X=[1 1 1 2 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 21ms/step
42 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
43 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 98ms/step
44 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
45 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
46 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 29ms/step
47 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 17ms/step
48 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
49 X=[2 2 1 2 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
50 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)

Process finished with exit code 0

```

Gambar 4.42 merupakan grafik perbandingan antara data aktual dengan data hasil prediksi yang dihasilkan oleh model B1. Garis warna biru menunjukkan data aktual dan garis warna oranye menunjukkan data hasil prediksi. Ada 50 data *testing* yang digunakan, yang diolah untuk mengetahui hasil prediksinya. *Output* yang dihasilkan, baik itu pada data aktual maupun prediksi, berupa angka 1 (rusak ringan), 2 (rusak sedang), atau 3 (rusak berat).



Gambar 4. 42 Grafik Hasil Model B1

## g. Model B2

Model B2 merupakan model *forward propagation in neural network* (FPNN) dengan *epoch* sebanyak 50 dan *hidden layer* sejumlah 2. Data *testing* akan diolah menggunakan model ini untuk mengetahui hasil prediksinya. Hasil dari perhitungan menggunakan model B2 sendiri didapatkan bahwa akurasi di *epoch* ke-50 adalah sebesar 62% dengan *loss* sebesar 0,6991.

```
Epoch 50/50
10/10 [=====] - 0s 1ms/step -
loss: 0.6991 - accuracy: 0.6200
```

Pada model B2, nilai bobot dan nilai bias awal akan ditentukan secara random, hingga didapatkan nilai bobot dan bias dengan *loss function* terendah untuk masing-masing *hidden layer* dan *output layer*. Nilai bobot

dan nilai bias akhir yang terbentuk pada model B2 antara lain sebagai berikut.

```

----- layer 1 -----
weights
[[-0.27655742  0.09892064  0.21778636  0.03487835
0.30719498]
 [ 0.56524456 -0.22104704 -0.8265249   0.70921886
0.39361298]
 [-0.28243765 -0.30939975  0.51223207  0.67270553
0.5310044 ]
 [ 0.89165246  0.35281488  0.48565376  0.5091759  -
0.3842609 ]
 [ 0.4175723   0.1582381  -0.4231398   0.38732603
0.65201664]]
biases
[ 0.08875675  0.23264508  0.19216809 -0.03169461 -
0.13816701]
----- layer 2 -----
weights
[[ 0.15310135  0.43222225  0.63244283 -0.42280963
0.7754582 ]
 [-0.7946243  0.32899436  0.68462706 -0.43915188
0.15943602]
 [-0.7131177  0.5377855   0.02754557 -0.04134889
0.7157761 ]
 [ 0.70835483  0.11180808  0.5153479   0.5456793  -
0.15397319]
 [-0.55354273 -0.7058373  -0.35782203 -0.54767483
0.3131644 ]]
biases
[-0.0933331  0.30996087  0.19297178 -0.05140684 -
0.12525102]
----- layer 3 -----
weights
[[-0.84960824 -0.4615587  -0.61003196  0.29640293]
 [-0.5171465  0.6579128  -0.40335917 -1.0777018 ]

```

```

[-0.62982714  0.08348093  0.10894619 -0.05259319]
[-0.8777929   0.26695988 -0.6428604  -0.69142103]
[-0.58031946 -0.44720218  0.11565133  0.22984195]]
biases
[-0.25784865  0.10268845  0.05917816 -0.10766051]

```

Berikut merupakan hasil prediksi metode FPNN model B2.

```

1/1 [=====] - 0s 86ms/step
1 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 20ms/step
2 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 22ms/step
3 X=[2 2 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
4 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 38ms/step
5 X=[2 2 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 26ms/step
6 X=[3 3 3 3 3], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 28ms/step
7 X=[2 2 2 3 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 35ms/step
8 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
9 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
10 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 47ms/step
11 X=[2 2 2 3 3], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
12 X=[2 2 2 3 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 30ms/step
13 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 21ms/step
14 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 20ms/step

```

```
15 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 34ms/step
16 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 33ms/step
17 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 28ms/step
18 X=[2 2 2 3 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 19ms/step
19 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
20 X=[2 2 2 3 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 22ms/step
21 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 24ms/step
22 X=[1 1 1 1 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 20ms/step
23 X=[1 1 1 1 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 21ms/step
24 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 24ms/step
25 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 22ms/step
26 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
27 X=[1 1 1 1 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 21ms/step
28 X=[1 1 1 1 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 20ms/step
29 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 21ms/step
30 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 27ms/step
31 X=[2 2 2 3 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 29ms/step
32 X=[2 2 2 3 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 30ms/step
33 X=[1 1 1 2 2], Predicted=(array([2], dtype=int64),)
```

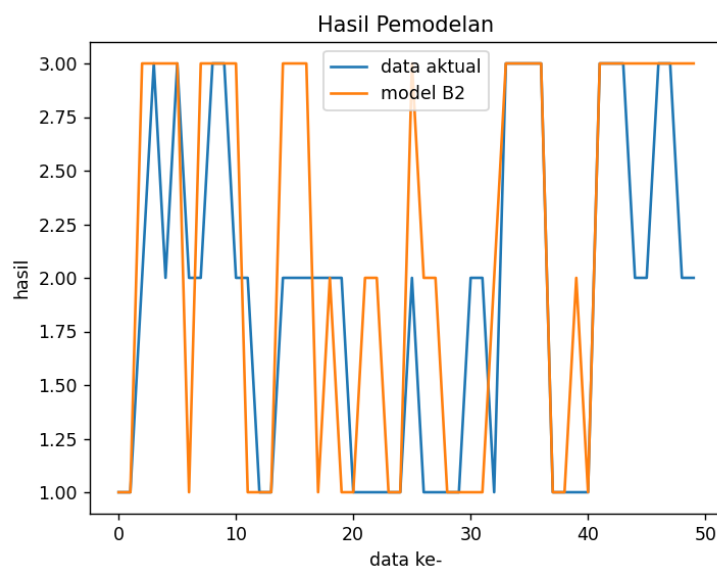
```

1/1 [=====] - 0s 24ms/step
34 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 33ms/step
35 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 34ms/step
36 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 30ms/step
37 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
38 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 19ms/step
39 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 24ms/step
40 X=[1 1 1 2 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 29ms/step
41 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 40ms/step
42 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 25ms/step
43 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 33ms/step
44 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 30ms/step
45 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 30ms/step
46 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
47 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 29ms/step
48 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
49 X=[2 2 1 2 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 26ms/step
50 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)

```

```
Process finished with exit code 0
```

Gambar 4.43 merupakan grafik perbandingan antara data aktual dengan data hasil prediksi yang dihasilkan oleh model B2. Garis warna biru menunjukkan data aktual dan garis warna oranye menunjukkan data hasil prediksi. Ada 50 data *testing* yang digunakan, yang diolah untuk mengetahui hasil prediksinya. *Output* yang dihasilkan, baik itu pada data aktual maupun prediksi, berupa angka 1 (rusak ringan), 2 (rusak sedang), atau 3 (rusak berat).



Gambar 4. 43 Grafik Hasil Model B2

#### h. Model B3

Model B3 merupakan model *forward propagation in neural network* (FPNN) dengan *epoch* sebanyak 50 dan *hidden layer* sejumlah 3. Data *testing* akan diolah menggunakan model ini untuk mengetahui hasil prediksinya. Hasil dari perhitungan menggunakan model B3 sendiri didapatkan bahwa akurasi di *epoch* ke-50 adalah sebesar 63% dengan *loss* sebesar 0,7483.



```
Epoch 50/50
10/10 [=====] - 0s 997us/step -
loss: 0.7483 - accuracy: 0.6300
```

Nilai bobot dan nilai bias awal pada model ini akan ditentukan secara random, hingga didapatkan nilai bobot dan bias dengan *loss function* terendah untuk masing-masing *hidden layer* dan *output layer*. Nilai bobot dan nilai bias akhir pada model B3 adalah sebagai berikut.

```
----- layer 1 -----
weights
[[ 0.586683  -0.4093929  -0.34196278  -0.49204212  -
0.01617081]
 [ 0.6817895  -0.00286895  -0.419827   -0.36616138
0.33700487]
 [-0.1198042   0.35917687  -0.63692296   0.35503364
0.33352673]
 [-0.40920338  -0.5695939  -0.56001407  -0.14114201  -
0.5637879  ]
 [ 0.58702683  -0.22496158  -0.6930284   -0.23709291
0.589845  ]]
biases
[-0.262555   0.         0.         0.         -
0.20602413]
----- layer 2 -----
weights
[[-0.7390101   0.4398615   0.07109665  -0.21474683
0.72229534]
 [ 0.44677234  -0.5286483   0.03978819   0.4131683  -
0.6983678  ]
 [ 0.31776702  -0.2820262  -0.63315713  -0.23398417  -
0.13973808]
 [-0.2782898   0.4411459  -0.01965916   0.27557075
0.07650703]
```

```

[-0.17748523  0.5236791  0.08773194  0.56523967
0.42566243]]
biases
[ 0.          -0.2814657  -0.19049504 -0.0313689  -
0.21334852]
----- layer 3 -----
weights
[[[ 0.42170644 -0.27762383 -0.3771813   0.7278513
0.6522273 ]
[-0.15201533 -0.49551815  0.341049   0.04563073
0.16118686]
[-0.5728679   0.6240208  -0.29909417 -0.11984122
0.7820891 ]
[ 0.01360303  0.31814718  0.32626435 -0.7917538  -
0.26208276]
[-0.38463178 -0.5926214   0.5932255   0.9271434
0.14316304]]]
biases
[ 0.          0.          -0.25986436 -0.03826926  -
0.25660267]
----- layer 4 -----
weights
[[[ 0.17242032 -0.6490177  -0.44093364  0.69625974]
[ 0.6845865   0.7172985   0.6512089  -0.79617405]
[ 0.0858346  -0.4905407   0.37470123  0.4919227 ]
[-0.46699187  0.15678605  0.41120237  0.881593 ]
[ 0.36673504 -0.29713082 -0.32369664  0.7478048 ]] ]
biases
[-0.29750592  0.37230146  0.18492916 -0.23271537]

```

Hasil perhitungan prediksi model B3 antara lain sebagai berikut.

```

1/1 [=====] - 0s 80ms/step
1 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 17ms/step
2 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 31ms/step

```

```

3 X=[2 2 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 26ms/step
4 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 18ms/step
5 X=[2 2 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
6 X=[3 3 3 3 3], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
7 X=[2 2 2 3 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 18ms/step
8 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
9 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 17ms/step
10 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
11 X=[2 2 2 3 3], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 25ms/step
12 X=[2 2 2 3 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 23ms/step
13 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 26ms/step
14 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 23ms/step
15 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
16 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
17 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 28ms/step
18 X=[2 2 2 3 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 19ms/step
19 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
20 X=[2 2 2 3 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 22ms/step
21 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)

```

```
1/1 [=====] - 0s 31ms/step
22 X=[1 1 1 1 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 34ms/step
23 X=[1 1 1 1 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 19ms/step
24 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 34ms/step
25 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 19ms/step
26 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
27 X=[1 1 1 1 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 27ms/step
28 X=[1 1 1 1 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 31ms/step
29 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 20ms/step
30 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 19ms/step
31 X=[2 2 2 3 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 19ms/step
32 X=[2 2 2 3 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 32ms/step
33 X=[1 1 1 2 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
34 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 33ms/step
35 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
36 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
37 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 29ms/step
38 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 20ms/step
39 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 24ms/step
```

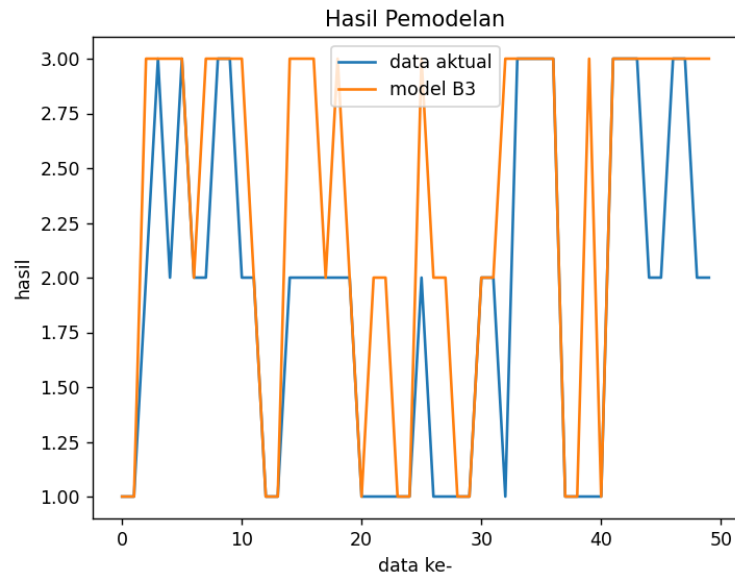
```

40 X=[1 1 1 2 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
41 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 27ms/step
42 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 26ms/step
43 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 25ms/step
44 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 26ms/step
45 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
46 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 18ms/step
47 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 27ms/step
48 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 16ms/step
49 X=[2 2 1 2 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 25ms/step
50 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)

Process finished with exit code 0

```

Gambar 4.44 merupakan grafik perbandingan antara data aktual dengan data hasil prediksi model B3. Garis warna biru merupakan data aktual dan garis warna oranye merupakan data hasil prediksi. *Output* yang dihasilkan, baik itu pada data aktual maupun prediksi, berupa angka 1 (rusak ringan), 2 (rusak sedang), atau 3 (rusak berat).



Gambar 4.44 Grafik Hasil Model B3

## i. Model B4

Model B4 merupakan model FPNN yang memiliki *epoch* 50 dan *hidden layer* sejumlah 4. Data *testing* akan diolah menggunakan model ini untuk mengetahui hasil prediksinya. Hasil dari perhitungan menggunakan model B4 sendiri didapatkan bahwa akurasi di *epoch* ke-50 adalah sebesar 70% dengan *loss* sebesar 0,6258.

```
Epoch 50/50
10/10 [=====] - 0s 1ms/step -
loss: 0.6258 - accuracy: 0.7000
```

Nilai bobot dan nilai bias awal akan ditentukan secara random, hingga didapatkan nilai bobot dan bias dengan *loss function* terendah. Nilai bobot dan nilai bias akhir pada model B4 adalah sebagai berikut.

```
----- layer 1 -----
weights
```

```

[[-0.6251094  0.37660086 -0.6547582  0.4464569
0.15945435]
 [ 0.2526246  0.19359249  0.01191968  0.2951663  -
0.303862  ]
 [ 0.38967526  0.23938328 -0.18021011  0.4836456  -
0.08327103]
 [ 0.2866482  0.3414261  -0.6763257  0.27602595  -
0.38840938]
[-0.13152231  0.18973371 -0.03125346  0.11415177  -
0.7388882  ]]
biases
[ 0.31154695 -0.11191623  0.          -0.2673243  0.
]
----- layer 2 -----
weights
[[-0.4834751  0.69887054  0.16824043 -0.60025215
0.59789383]
 [ 0.0902755  -0.12203942  0.4034819  -0.02489877  -
0.5815191  ]
 [-0.5807597  0.26677847 -0.198277  -0.13807207  -
0.21267694]
 [ 0.33921733 -0.23908198  0.7724451  0.6672311  -
0.25386703]
 [ 0.5045825  0.52976155 -0.37757283  0.52108574
0.58948183]]
biases
[ 0.16628492 -0.00222366 -0.2375828  -0.25057206  0.
]
----- layer 3 -----
weights
[[ 0.15272017  0.70135367 -0.42396712 -0.24507156
0.67681587]
 [ 0.10385765 -0.5591854  0.25249338  0.43093556  -
0.23493159]
 [ 0.42422456 -0.6832648  0.02288514  0.00916898  -
0.28399748]

```

```

[ 0.9057229 -0.7369033 -0.6409364 0.5885536 -
0.24397898]
[ 0.19994706 -0.03618586 -0.60679877 0.4688642
0.18984109]]
biases
[-0.21406718 0.          0.          -0.20718902 0.
]
----- layer 4 -----
weights
[[ 0.12483836 0.29881334 0.41962802 0.4094685 -
0.51354456]
[ 0.23570406 -0.07097721 0.3024001 0.18586463
0.50517094]
[ 0.4627756 0.6329025 -0.07211649 -0.34182298 -
0.24648082]
[ 0.75894177 0.59791875 -0.7465075 -0.01396916 -
0.6173775 ]
[-0.5279755 0.4686061 0.17948544 0.33912504
0.3550439 ]]
biases
[-0.12680432 -0.2834545 0.3425847 -0.24730438 0.
]
----- layer 5 -----
weights
[[-0.11038945 -0.42535365 -0.0071041 0.7866064 ]
[-0.4684261 -0.88864017 -0.34505972 0.8274752 ]
[-1.1050982 0.27188665 0.4173567 -0.40073732]
[-0.8864225 -0.7303649 0.06346291 0.8136272 ]
[ 0.7004471 -0.6821232 -0.6021623 0.36330533]]
biases
[-0.2567286 0.3333696 0.03474735 -0.21100058]

```

Hasil perhitungan prediksi model B4 adalah sebagai berikut.

```

1/1 [=====] - 0s 101ms/step
1 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 18ms/step

```



```
2 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 20ms/step
3 X=[2 2 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
4 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 28ms/step
5 X=[2 2 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
6 X=[3 3 3 3 3], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 31ms/step
7 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 29ms/step
8 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 31ms/step
9 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 35ms/step
10 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 26ms/step
11 X=[2 2 2 3 3], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 25ms/step
12 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 27ms/step
13 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 34ms/step
14 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 36ms/step
15 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 31ms/step
16 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
17 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 26ms/step
18 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
19 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 18ms/step
20 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
```

```

1/1 [=====] - 0s 36ms/step
21 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 23ms/step
22 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 35ms/step
23 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 27ms/step
24 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 36ms/step
25 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 24ms/step
26 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 26ms/step
27 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 27ms/step
28 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 24ms/step
29 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 30ms/step
30 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 26ms/step
31 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 39ms/step
32 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 29ms/step
33 X=[1 1 1 2 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 21ms/step
34 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 35ms/step
35 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 30ms/step
36 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 26ms/step
37 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
38 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 39ms/step

```

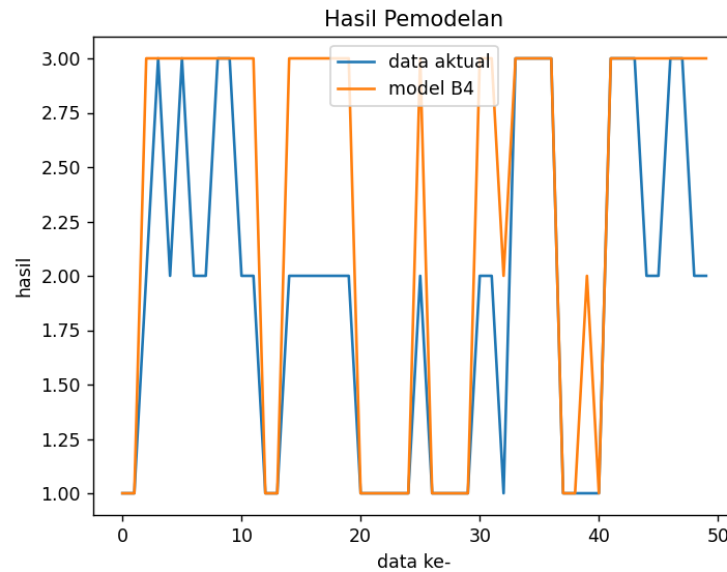
```

39 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 27ms/step
40 X=[1 1 1 2 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 27ms/step
41 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 32ms/step
42 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
43 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
44 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
45 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
46 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 26ms/step
47 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 31ms/step
48 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
49 X=[2 2 1 2 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
50 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)

Process finished with exit code 0

```

Gambar 4.45 menunjukkan grafik perbandingan antara data aktual dengan data hasil prediksi pada model B4. Garis warna biru menggambarkan data aktual dan garis warna oranye untuk data hasil prediksi. *Output* yang dihasilkan berupa angka 1 (rusak ringan), 2 (rusak sedang), atau 3 (rusak berat).



Gambar 4. 45 Grafik Hasil Model B4

## j. Model B5

Model B5 sendiri merupakan model FPNN dengan *epoch* 50 dan *hidden layer* sejumlah 5. Hasil dari perhitungan menggunakan model B5 sendiri didapatkan bahwa akurasi di *epoch* ke-50 adalah sebesar 72% dengan *loss* sebesar 0,6688.

```
Epoch 50/50
10/10 [=====] - 0s 2ms/step -
loss: 0.6688 - accuracy: 0.7200
```

Nilai bobot dan nilai bias awal pada model B5 akan ditentukan secara random, untuk kemudian diolah hingga didapatkan nilai bobot dan bias dengan *loss function* terendah untuk masing-masing *hidden layer* dan *output layer*. Berikut merupakan nilai bobot dan nilai bias akhir pada model B5.

```
----- layer 1 -----
```

```

weights
[[-0.5324094  0.8735001 -0.18654478  0.42729044
0.8786942 ]
 [-0.5257605 -0.2052081  0.32853484  0.02701932  0.90155
]
 [ 0.06890696  0.7295287 -0.4306185 -0.55942196
0.1668022 ]
 [ 0.3291862  0.5309823 -0.5662979 -0.6562525 -
0.3822975 ]
 [-0.47292978 -0.7875898  0.03475606 -0.5736673
0.07628789]]
biases
[ 0.          -0.3335546  0.          0.          -
0.37338984]
----- layer 2 -----
weights
[[ 0.05194557 -0.7567837  0.66255975  0.38112235 -
0.6232525 ]
 [ 0.7803185  0.32256937 -0.6807017  0.37337244
0.14384276]
 [ 0.73057544  0.28569245  0.02180499  0.1177935
0.29225612]
 [-0.20830762 -0.0606631 -0.49022192  0.03459632 -
0.2136755 ]
 [ 0.8775289  0.75797683 -0.36825415  0.29944164
0.45861742]]
biases
[-0.3133882 -0.18427595  0.          0.33584607 -
0.43690565]
----- layer 3 -----
weights
[[ 0.765918  0.11492778  0.82897055  0.15467632
0.9796137 ]
 [ 0.33054832  0.24788845  0.2927655 -0.29342473
0.39422098]
 [ 0.15686125 -0.15610927  0.3559934  0.37052655 -
0.56070846]

```

```

[ 0.0386375 -0.08730554 -0.5001582 -0.07725185
0.6751855 ]
[-0.6676892 -0.84667224 0.5729022 -0.29618734 -
0.41618103]]
biases
[ 0.34059387 0.29963824 -0.35268936 0.          -
0.2559791 ]
----- layer 4 -----
weights
[[ 0.5871041 0.44715932 0.31786644 -0.58625036 -
0.25634986]
[-0.7668118 0.6975431 -0.30681828 -0.70494264 -
0.4895582 ]
[-0.5560732 -0.6681717 0.67171514 0.04281893
0.01167649]
[ 0.00170791 0.4839815 -0.4644126 0.7609558 -
0.5581992 ]
[-0.18060184 0.14442475 0.97095263 0.30193242 -
0.37587988]]
biases
[ 0.          0.34712368 -0.28920367 -0.01747656 0.
]
----- layer 5 -----
weights
[[ -0.16543108 -0.60489583 -0.00370502 0.57371664
0.6381544 ]
[-1.2984031 -0.702837 -0.4188619 0.2575611 -
0.28728455]
[ 0.61835754 -0.10506546 -0.00653446 -0.31309795 -
0.3928276 ]
[-0.6615674 -0.1320396 0.25933647 -0.62978107
0.35143232]
[-0.04614961 0.00633281 0.7555976 -0.69238496 -
0.23277271]]
biases
[-0.26532227 0.          0.          0.          0.
]

```

```

----- layer 6 -----
weights
[[-0.23450935 -0.34314027  0.16037755  0.38914582]
 [ 0.00473726  0.80786645  0.58182144 -0.3284687 ]
 [ 0.7390853  -0.04468817 -0.04965085  0.0415262 ]
 [ 0.6494584   0.49268532 -0.60939014  0.12859452]
 [ 0.49189925 -0.05895501 -0.08968782  0.78006256]]
biases
[-0.30840507  0.4013827   0.09342942 -0.13009891]

```

Berikut merupakan hasil perhitungan prediksi metode FPNN pada model B5.

```

1/1 [=====] - 0s 105ms/step
1 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 20ms/step
2 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 22ms/step
3 X=[2 2 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
4 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 39ms/step
5 X=[2 2 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 52ms/step
6 X=[3 3 3 3 3], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 42ms/step
7 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 34ms/step
8 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 54ms/step
9 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 49ms/step
10 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 41ms/step
11 X=[2 2 2 3 3], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 47ms/step

```

```
12 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 42ms/step
13 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 44ms/step
14 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 35ms/step
15 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 38ms/step
16 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 36ms/step
17 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 32ms/step
18 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 44ms/step
19 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 40ms/step
20 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 56ms/step
21 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 37ms/step
22 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 42ms/step
23 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 52ms/step
24 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 37ms/step
25 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 50ms/step
26 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 51ms/step
27 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 39ms/step
28 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 48ms/step
29 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 44ms/step
30 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
```



```
1/1 [=====] - 0s 33ms/step
31 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 54ms/step
32 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 46ms/step
33 X=[1 1 1 2 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 41ms/step
34 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 28ms/step
35 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 39ms/step
36 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 47ms/step
37 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 38ms/step
38 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 41ms/step
39 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 46ms/step
40 X=[1 1 1 2 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 34ms/step
41 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 54ms/step
42 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 44ms/step
43 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 37ms/step
44 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 48ms/step
45 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 44ms/step
46 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 41ms/step
47 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 36ms/step
48 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 40ms/step
```

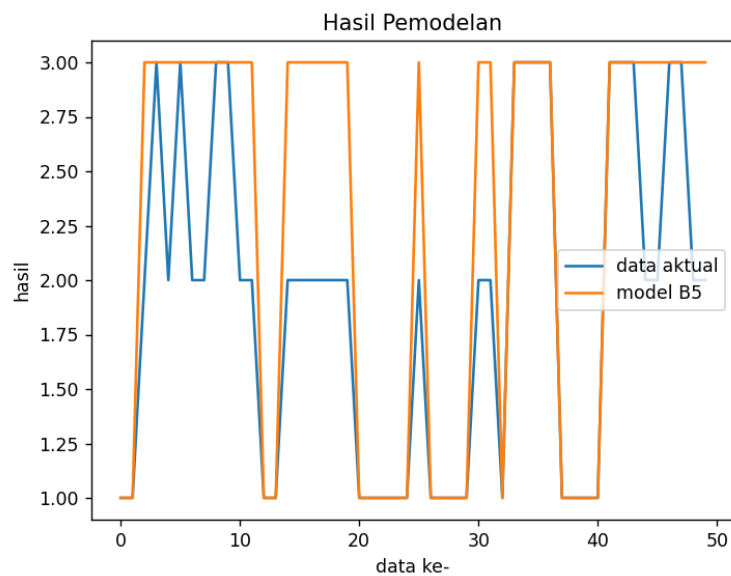
```

49 X=[2 2 1 2 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 41ms/step
50 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)

Process finished with exit code 0

```

Gambar 4.46 merupakan grafik perbandingan data aktual dan data hasil prediksi pada model B5. Garis warna biru menggambarkan data aktual dan garis warna oranye untuk data hasil prediksi.



Gambar 4. 46 Grafik Hasil Model B5

#### k. Model C1

Model C1 merupakan model FPNN yang memiliki *epoch* 75 dan *hidden layer* sebanyak 1. Hasil dari perhitungan menggunakan model C1 adalah akurasi di *epoch* ke-75 sebesar 62% dengan *loss* sebesar 0,7817.

```

Epoch 75/75
10/10 [=====] - 0s 790us/step -
loss: 0.7817 - accuracy: 0.6200

```

Nilai bobot dan nilai bias awal pada model C1 akan ditentukan secara random, untuk kemudian diolah hingga didapatkan nilai bobot dan bias dengan *loss function* terendah untuk masing-masing *hidden layer* dan *output layer*. Berikut merupakan nilai bobot dan nilai bias akhir pada model C1.

```

----- layer 1 -----
weights
[[-0.08068407  0.22618909  0.4408086  0.11806589
 0.77155155]
 [ 0.49903095  0.38197097  0.00295526 -0.56423616 -
 0.28171626]
 [-0.7030466  -0.17097668  0.4625808  0.08596253
 0.5740152 ]
 [-0.5757245  -0.25524825 -0.3850294  0.07978195 -
 0.55665433]
 [-0.23658389  0.6201258  0.36256194 -0.18647921
 0.41901684]]
biases
[ 0.          -0.11050148  0.11799137  0.          -
 0.23393676]
----- layer 2 -----
weights
[[ 0.79212713 -0.6954795  -0.2967131  -0.6679219 ]
 [-1.0031543  -0.6852671  0.50442195 -0.08304162]
 [-0.5292822  0.43355817 -0.8511688  0.2437862 ]
 [ 0.19790816  0.60471165  0.59333026 -0.12879235]
 [-1.0146925  -0.8942093  0.41235033  0.35855648]]
biases
[-0.35478774  0.41771865  0.07418545 -0.25512493]

```

Hasil perhitungan prediksi metode FPNN model C1 adalah sebagai berikut.

```

1/1 [=====] - 0s 66ms/step
1 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 19ms/step
2 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 19ms/step
3 X=[2 2 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 18ms/step
4 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 17ms/step
5 X=[2 2 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
6 X=[3 3 3 3 3], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
7 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 16ms/step
8 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 15ms/step
9 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 18ms/step
10 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 18ms/step
11 X=[2 2 2 3 3], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 18ms/step
12 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
13 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 18ms/step
14 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 17ms/step
15 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 17ms/step
16 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
17 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
18 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step

```

```
19 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
20 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
21 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 21ms/step
22 X=[1 1 1 1 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 17ms/step
23 X=[1 1 1 1 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
24 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 18ms/step
25 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 18ms/step
26 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
27 X=[1 1 1 1 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 18ms/step
28 X=[1 1 1 1 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 17ms/step
29 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 17ms/step
30 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 20ms/step
31 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
32 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
33 X=[1 1 1 2 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
34 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
35 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
36 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
37 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
```

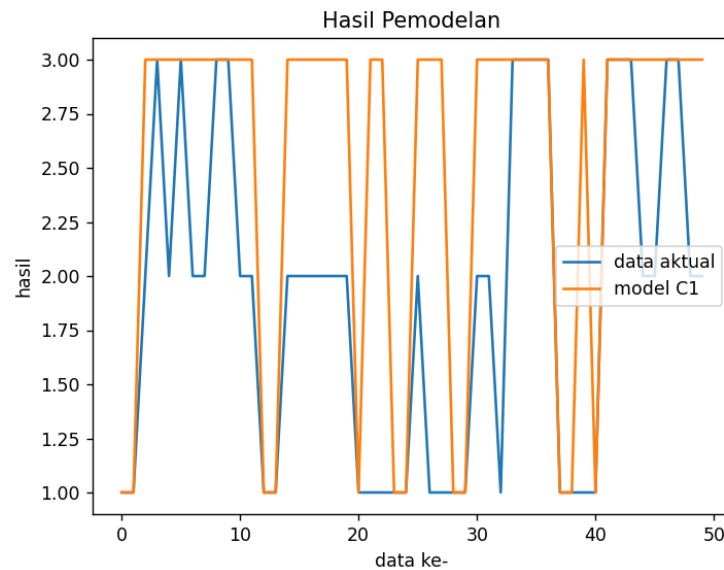
```

1/1 [=====] - 0s 18ms/step
38 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 18ms/step
39 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 18ms/step
40 X=[1 1 1 2 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 18ms/step
41 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 17ms/step
42 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 18ms/step
43 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 18ms/step
44 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 17ms/step
45 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
46 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 18ms/step
47 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
48 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 16ms/step
49 X=[2 2 1 2 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 17ms/step
50 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)

Process finished with exit code 0

```

Gambar 4.47 merupakan grafik perbandingan data aktual dan data hasil prediksi pada model C1. Garis warna biru menggambarkan data aktual dan garis warna oranye untuk data hasil prediksi.



Gambar 4.47 Grafik Hasil Model C1

### 1. Model C2

Model C2 merupakan model *forward propagation in neural network* (FPNN) dengan *epoch* sebanyak 75 dan *hidden layer* sejumlah 2. Data *testing* akan diolah menggunakan model ini untuk mengetahui hasil prediksinya. Hasil dari perhitungan menggunakan model C2 sendiri didapatkan akurasi di *epoch* ke-75 sebesar 65% dengan *loss* sebesar 0,6500.

```
Epoch 75/75
10/10 [=====] - 0s 2ms/step -
loss: 0.6362 - accuracy: 0.6500
```

Pada model C2, nilai bobot dan nilai bias awal akan ditentukan secara random, hingga didapatkan nilai bobot dan bias dengan *loss function* terendah untuk tiap *hidden layer* dan *output layer*. Berikut merupakan nilai bobot dan nilai bias akhir yang terbentuk pada model C2.

```
----- layer 1 -----
```

```

weights
[[ 0.8125291 -0.40021443  0.6336018 -0.57146597
0.14875484]
 [ 0.8603959  0.31617746  0.67721593 -0.3039983 -
0.7422283 ]
 [ 0.75048953 -0.31379372  0.3034459  0.5439451 -
0.76198614]
 [ 0.1368083  0.77252924 -0.41257983 -0.74489635 -
0.25414568]
 [ 0.61986023 -0.5599561 -0.5604878 -0.34851757
0.35419095]]
biases
[ 0.00460715  0.12254894 -0.22679795  0.          0.
]
----- layer 2 -----
weights
[[-7.0676160e-01  5.7751703e-01 -6.6566128e-01
8.8529021e-02
 4.0804375e-02]
 [ 6.8056571e-01 -1.2389301e-01  7.4460828e-01
5.0492054e-01
 1.7235693e-01]
 [-7.0717436e-01  5.7505164e-02 -4.9220297e-01 -
5.9110790e-01
 7.9054195e-01]
 [ 6.4050519e-01 -3.4201470e-01 -1.5103817e-04
6.5184867e-01
 4.3746269e-01]
 [-3.9245695e-01  2.9518044e-01 -4.9642086e-02
5.0792515e-01
 4.9656641e-01]]
biases
[ 0.          -0.0077847  0.          0.4575922 -0.2502498]
----- layer 3 -----
weights
[[-0.42519453  0.32654536 -0.16642398  0.64299047]
 [-0.7619545  0.27752104  0.44567886  0.7065374 ]

```



```

[-0.70423377  0.15137345 -0.02179521  0.47949564]
[-1.1594958   0.9791085  -0.18360335 -0.93089104]
 [ 0.40933084 -0.5930153  -0.5325469  -0.10775367]]
biases
[-0.21073493 -0.02787346  0.32317635 -0.14905953]

```

Berikut merupakan hasil perhitungan prediksi model C2.

```

1/1 [=====] - 0s 95ms/step
1 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 24ms/step
2 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 27ms/step
3 X=[2 2 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
4 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 36ms/step
5 X=[2 2 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 35ms/step
6 X=[3 3 3 3 3], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 37ms/step
7 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
8 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 27ms/step
9 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
10 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
11 X=[2 2 2 3 3], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 18ms/step
12 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 34ms/step
13 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 25ms/step
14 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 25ms/step

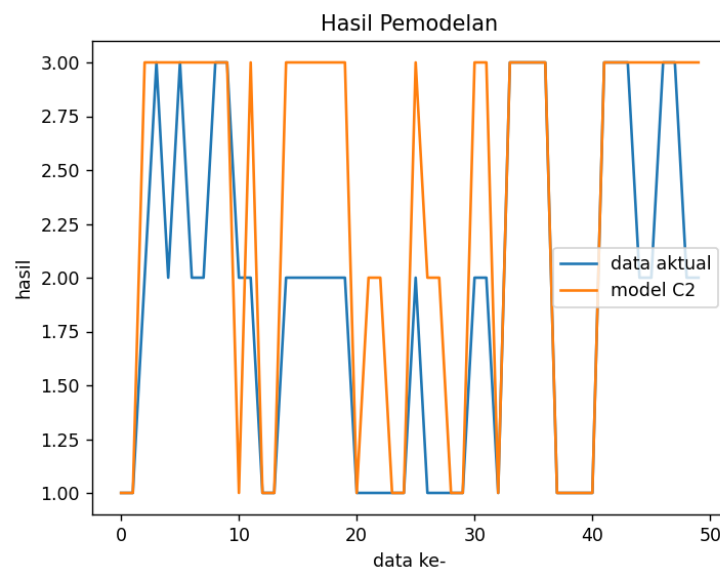
```

```
15 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 35ms/step
16 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
17 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
18 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
19 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 29ms/step
20 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
21 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 22ms/step
22 X=[1 1 1 1 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 25ms/step
23 X=[1 1 1 1 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 29ms/step
24 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 31ms/step
25 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 34ms/step
26 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
27 X=[1 1 1 1 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 19ms/step
28 X=[1 1 1 1 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 31ms/step
29 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 19ms/step
30 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 24ms/step
31 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 34ms/step
32 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 28ms/step
33 X=[1 1 1 2 2], Predicted=(array([1], dtype=int64),)
```

```
1/1 [=====] - 0s 29ms/step
34 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 30ms/step
35 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
36 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 25ms/step
37 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
38 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 21ms/step
39 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 36ms/step
40 X=[1 1 1 2 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 30ms/step
41 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 25ms/step
42 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
43 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
44 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
45 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
46 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 30ms/step
47 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
48 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 26ms/step
49 X=[2 2 1 2 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 18ms/step
50 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)
```

```
Process finished with exit code 0
```

Gambar 4.39 menunjukkan grafik perbandingan antara data aktual dengan data hasil prediksi dari model C2. Garis warna biru menunjukkan data aktual dan garis warna oranye menunjukkan data hasil prediksi. Ada 50 data *testing* yang digunakan, yang diolah untuk mengetahui hasil prediksinya. *Output* yang dihasilkan, baik itu aktual maupun prediksi, berupa angka 1 (rusak ringan), 2 (rusak sedang), dan 3 (rusak berat).



Gambar 4. 48 Grafik Hasil Model C2

#### m. Model C3

Model C3 merupakan model *forward propagation in neural network* (FPNN) yang memiliki *epoch* sebanyak 75 dan *hidden layer* sejumlah 3. Data *testing* kemudian juga akan diolah menggunakan model ini. Hasil dari perhitungan menggunakan model C3, didapatkan bahwa akurasi di *epoch* ke-75 adalah sebesar 65% dengan *loss* sebesar 0,6186.

```
Epoch 75/75
10/10 [=====] - 0s 1ms/step -
loss: 0.6186 - accuracy: 0.6500
```

Pada model C3 yang terbentuk, nilai bobot dan bias awal akan ditentukan secara random, hingga didapatkan nilai bobot dan bias dengan *loss function* terendah. Nilai bobot dan bias akhir yang terbentuk pada masing-masing *hidden layer* dan *output layer* pada model C3 antara lain sebagai berikut.

```

----- layer 1 -----
weights
[[ 0.20638548  0.06917398 -0.57221687 -0.0490145
 0.9421192 ]
 [ 0.49451333  0.31501588  0.4993831  0.2057397
 0.4674656 ]
 [-0.22435553  0.46290627 -0.61276484  0.68141365
 -0.10170441]
 [-0.67524886  0.6621373  -0.17789686  0.44627082
 -0.36722952]
 [ 0.6135747  0.30923367  0.07161063 -0.21230271
 -0.35054132]]
biases
[-0.15314679  0.25106195  0.          -0.2528874
 -0.24646837]
----- layer 2 -----
weights
[[ 0.00973326 -0.7396404  0.12807488  0.20055467
 0.5217037 ]
 [-0.4473631  -0.6908566  -0.22463438 -0.23465347
 0.11882226]
 [ 0.08179677 -0.3477484  -0.0091759  -0.19205797
 -0.2008385 ]
 [ 0.4772836  0.14603895  0.7109193  0.28924742
 -0.7282599 ]
 [-0.23938012  0.6901165  1.1298277  -0.27026922
 0.7096626 ]]
biases

```

```

[ 0.          0.          -0.26941454 -0.01046326
-0.09297357]
----- layer 3 -----
weights
[[-2.1817333e-01  3.6572921e-01 -7.2253186e-01
-6.9169372e-01
  4.2666793e-02]
 [-2.5355816e-04  7.3137879e-03 -7.3896229e-01
-6.7050135e-01
  -6.9038951e-01]
 [-1.9821857e-01  1.1819030e+00 -4.8162627e-01
6.1751914e-01
  6.6382617e-01]
 [-4.3736979e-01 -6.5939027e-01  7.9789340e-02
4.6679765e-01
  4.6063599e-01]
 [ 3.4214234e-01  7.5423867e-01 -1.6616553e-01
-4.3656531e-01
  3.2661313e-01]]
biases
[-0.05548073 -0.15758027  0.          -0.02656673
-0.12380288]
----- layer 4 -----
weights
[[-0.07935132  0.32113326 -0.14506067 -0.33790156]
 [-0.3853207  0.15171662  0.51180965  0.8629192 ]
 [-0.71181    -0.5326867  -0.713235   -0.31857705]
 [-0.6112084  -1.0685377   0.6803564  -0.01442981]
 [-1.2569027  0.03331246 -0.46928415  0.79220825]]
biases
[-0.35797194  0.42456597  0.2517382  -0.33658957]

```

Hasil perhitungan prediksi metode FPNN dari model C3 adalah sebagai berikut.

```
1/1 [=====] - 0s 92ms/step
```

```

1 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 22ms/step
2 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 30ms/step
3 X=[2 2 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 37ms/step
4 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 32ms/step
5 X=[2 2 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 26ms/step
6 X=[3 3 3 3 3], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
7 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 26ms/step
8 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
9 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
10 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 28ms/step
11 X=[2 2 2 3 3], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 28ms/step
12 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
13 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 31ms/step
14 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 22ms/step
15 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 25ms/step
16 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 37ms/step
17 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
18 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
19 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)

```

```
1/1 [=====] - 0s 24ms/step
20 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 31ms/step
21 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 32ms/step
22 X=[1 1 1 1 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 38ms/step
23 X=[1 1 1 1 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 23ms/step
24 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 27ms/step
25 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 20ms/step
26 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
27 X=[1 1 1 1 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 19ms/step
28 X=[1 1 1 1 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 21ms/step
29 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 23ms/step
30 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 30ms/step
31 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 31ms/step
32 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 38ms/step
33 X=[1 1 1 2 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 26ms/step
34 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
35 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 26ms/step
36 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 36ms/step
37 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
```



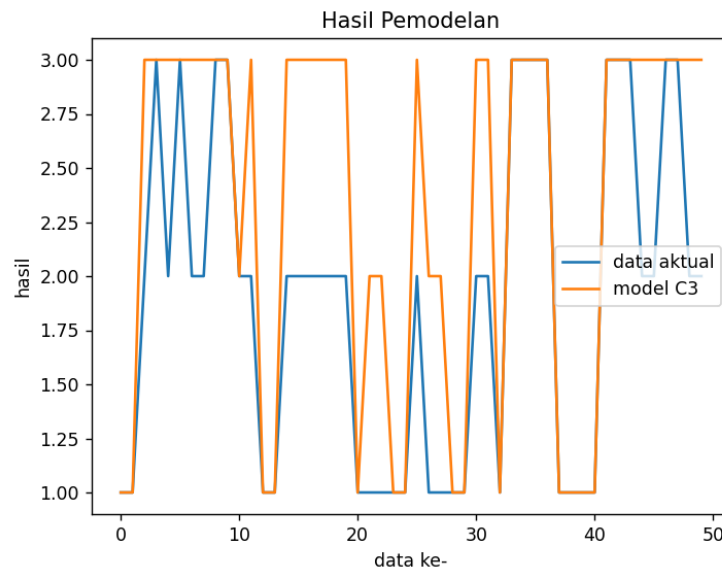
```

38 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 23ms/step
39 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 25ms/step
40 X=[1 1 1 2 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 25ms/step
41 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 22ms/step
42 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 29ms/step
43 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
44 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 31ms/step
45 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
46 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 31ms/step
47 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
48 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
49 X=[2 2 1 2 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
50 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)

Process finished with exit code 0

```

Gambar 4.49 menunjukkan grafik perbandingan antara data aktual dengan data hasil prediksi dari model C3. Garis warna biru menunjukkan data aktual dan garis warna oranye menunjukkan data hasil prediksi. Ada 50 data *testing* yang digunakan, yang diolah untuk mengetahui hasil prediksinya. *Output* yang dihasilkan, baik itu aktual maupun prediksi, berupa angka 1 (rusak ringan), 2 (rusak sedang), dan 3 (rusak berat).



Gambar 4. 49 Grafik Hasil Model C3

## n. Model C4

Model C4 merupakan model *forward propagation in neural network* (FPNN) yang memiliki *epoch* sebanyak 75 dan *hidden layer* sejumlah 4. Model ini juga akan digunakan untuk mengolah *data testing*, hingga diketahui hasil prediksinya. Hasil perhitungan model C4, didapatkan bahwa akurasi di *epoch ke-75* adalah sebesar 69% dengan loss sebesar 0,8365.

```
Epoch 75/75
10/10 [=====] - 0s 776us/step -
loss: 0.8365 - accuracy: 0.6900
```

Nilai bobot dan bias awal pada model C4 akan ditentukan secara random, hingga didapatkan nilai bobot dan bias dengan *loss function* terendah. Nilai bobot dan bias akhir yang terbentuk pada masing-masing *hidden layer* dan *output layer* pada model C4 antara lain sebagai berikut.

```

----- layer 1 -----
weights
[[-0.7762281  0.19823658  0.10816862  0.04122
0.17808041]
 [-0.07483214 -0.5270654  0.4115013  0.5431875
0.47915855]
 [-0.13099435 -0.37611052 -0.532767  -0.8200799
0.7351797 ]
 [ 0.65049994 -0.46739256 -0.01458476  0.2349459
0.41711238]
 [ 0.28400648  0.49333286  0.35380152  0.41200358 -
0.5577218 ]]
biases
[ 0.45516232  0.          -0.00772634  0.03160242 -
0.12342267]
----- layer 2 -----
weights
[[ 0.8302054  1.0905144 -1.3153937  0.21148354 -
0.2641318 ]
 [ 0.73871136  0.10065204  0.7172476  0.6744863 -
0.06802958]
 [ 0.4714899  0.24160853  0.11393297  0.28825915
0.55445284]
 [ 0.272575  0.15998763  0.11332744 -0.3717549 -
0.19943139]
 [ 0.343118  -0.07796378  0.41181144 -0.20191443
0.09241252]]
biases
[ 0.00613398  0.1194593  0.01838726  0.          -
0.08405763]
----- layer 3 -----
weights
[[ 0.2272535  0.35418472 -0.37935275 -0.02601469
0.15600723]
 [ 0.74938273  0.6312821  0.31172466 -0.5483581 -
0.5240507 ]

```

```

[-0.6031627 -0.50315636 0.47196698 -0.38797373 -
0.03738344]
[-0.7524576 0.31177914 -0.10737431 0.73619604
0.66637707]
[-0.41404694 -0.07173681 -0.7053708 -0.14694995 -
0.31456378]]
biases
[ 0.13313593 -0.105546 0. 0. -
0.09139478]
----- layer 4 -----
weights
[[ 0.42092103 -0.72139585 0.74818087 0.744122 -
0.66886544]
[ 0.05924163 -0.64072025 -0.17102338 0.39764923 -
0.31274197]
[ 0.35295784 0.28415334 0.5685754 -0.01590651 -
0.47343966]
[-0.74177235 -0.1473524 -0.13663733 -0.7009544 -
0.24822402]
[ 0.3173211 0.4878133 0.39165783 0.52535874
0.14885917]]
biases
[-0.00773621 -0.04516816 -0.00054071 -0.06371886 -
0.01447603]
----- layer 5 -----
weights
[[-0.86970854 0.17145115 0.49324512 -0.47480416]
[ 0.13299532 -0.7899667 0.39087144 -0.25398734]
[-0.4560966 0.49368623 0.4884161 -1.1432422 ]
[-0.983685 0.6204352 0.4191886 -0.77680033]
[-0.5324389 0.43889347 -0.7902468 -0.70337003]]
biases
[-0.67374057 -0.31913298 -0.2588495 0.5733692 ]

```

Berikut merupakan hasil prediksi dari model C4.

```
1/1 [=====] - 0s 92ms/step
```

```
1 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 22ms/step
2 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 24ms/step
3 X=[2 2 3 3 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 16ms/step
4 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 30ms/step
5 X=[2 2 3 3 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 21ms/step
6 X=[3 3 3 3 3], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
7 X=[2 2 2 3 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 21ms/step
8 X=[2 2 2 3 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 19ms/step
9 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
10 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 25ms/step
11 X=[2 2 2 3 3], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 23ms/step
12 X=[2 2 2 3 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 18ms/step
13 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 23ms/step
14 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 19ms/step
15 X=[2 2 2 3 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 21ms/step
16 X=[2 2 2 3 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 20ms/step
17 X=[2 2 2 3 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 25ms/step
18 X=[2 2 2 3 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 19ms/step
19 X=[2 2 2 3 2], Predicted=(array([1], dtype=int64),)
```

```
1/1 [=====] - 0s 19ms/step
20 X=[2 2 2 3 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 21ms/step
21 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 21ms/step
22 X=[1 1 1 1 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 32ms/step
23 X=[1 1 1 1 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 28ms/step
24 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 20ms/step
25 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 20ms/step
26 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 18ms/step
27 X=[1 1 1 1 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 25ms/step
28 X=[1 1 1 1 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 26ms/step
29 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 22ms/step
30 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 19ms/step
31 X=[2 2 2 3 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 20ms/step
32 X=[2 2 2 3 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 17ms/step
33 X=[1 1 1 2 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 21ms/step
34 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 17ms/step
35 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 30ms/step
36 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
37 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 16ms/step
```

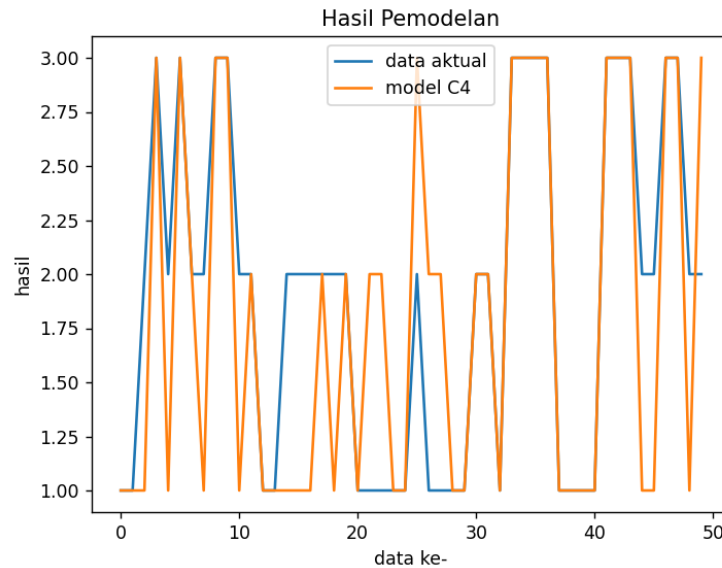
```

38 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 23ms/step
39 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 19ms/step
40 X=[1 1 1 2 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 28ms/step
41 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 25ms/step
42 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
43 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
44 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
45 X=[2 2 2 3 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 23ms/step
46 X=[2 2 2 3 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 23ms/step
47 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
48 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
49 X=[2 2 1 2 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 25ms/step
50 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)

Process finished with exit code 0

```

Gambar 4.50 menunjukkan grafik perbandingan antara data aktual dengan data hasil prediksi dari model C4. Garis warna biru menunjukkan data aktual dan garis warna oranye menunjukkan data hasil prediksi. Ada 50 data *testing* yang digunakan, yang diolah untuk mengetahui hasil prediksi yang berupa angka 1 (rusak ringan), 2 (rusak sedang), dan 3 (rusak berat).



Gambar 4. 50 Grafik Hasil Model C4

## o. Model C5

Model C5 adalah model FPNN dengan *epoch* sebanyak 75 dan *hidden layer* sebanyak 5. Hasil dari perhitungan menggunakan model C5 adalah akurasi di *epoch* ke-75 sebesar 80% dengan *loss* sebesar 0,4061.

```
Epoch 75/75
10/10 [=====] - 0s 1ms/step -
loss: 0.4061 - accuracy: 0.8000
```

Pada model C5, nilai bobot dan nilai bias awal pada model C1 akan ditentukan secara random, untuk kemudian diolah hingga didapatkan nilai bobot dan bias dengan *loss function* terendah untuk masing-masing *hidden layer* dan *output layer*. Nilai bobot dan nilai bias akhir pada model C1 antara lain sebagai berikut.

```
----- layer 1 -----
weights
```



```

[[ 0.7008986  0.94161224  0.5316979 -0.9264161 -
0.4590729 ]
 [ 0.1133272 -0.04141214  0.66095066  0.49628425 -
0.58323956]
 [ 0.36652732  0.13871565 -0.56207263  0.11621604
0.1726987 ]
 [-0.385978  0.87276775 -0.04022878  0.5007899
0.8646353 ]
 [ 0.6281096  0.22417578 -0.59241635  0.19595565 -
0.10521781]]
biases
[-0.2171302 -0.2892615 -0.00077003  0.24039984
0.41499734]
----- layer 2 -----
weights
[[ 0.7279862  0.68469924 -0.23744857 -0.3617246
0.81693465]
 [ 0.74566734  0.21361151 -0.7585762  0.59471756 -
0.07236859]
 [-0.30015817 -0.20169382  0.4174888  0.91814685
0.5907049 ]
 [-0.6589911 -0.43910688  0.09412384 -0.17156933 -
0.08392942]
 [-0.3469093 -1.0774701  0.6015519  0.37119618 -
0.59888494]]
biases
[-0.23573543 -0.1475771  0.          -0.03426367 -
0.22598945]
----- layer 3 -----
weights
[[ 0.18038113 -0.54042876  0.26072326  0.39235628
0.72181886]
 [-0.7979323 -0.20473337  0.5638266  0.51111186 -
0.19559728]
 [ 0.01343662 -0.26750308  0.6927637 -0.5423738 -
0.3950262 ]

```

```

[ 0.5962764 -0.22681278 0.5959302 -0.13357238 -
0.61076355]
[-0.3522975 -0.6892644 0.5996462 -0.41091472
0.78360236]]
biases
[ 0.2480226 0. -0.32776684 -0.28235367 -
0.03852906]
----- layer 4 -----
weights
[[ 0.2913635 0.953444 0.45398685 0.6245249 -
1.0298233 ]
[ 0.55340147 0.49171317 -0.09179932 -0.66905475
0.5250883 ]
[ 0.9745526 -0.277276 0.89898384 0.16553885
0.37489468]
[-0.57683593 -0.79075134 0.39077452 -0.5369
0.5456795 ]
[-0.6223436 -0.1415039 0.29696542 -0.7869577
0.19408402]]
biases
[-0.34376714 0.2092486 -0.23104656 0.28262675 -
0.21556886]
----- layer 5 -----
weights
[[ 0.29327458 -0.38211942 0.76956874 -0.09046799
0.82865685]
[ 0.59431446 0.21283948 -0.6380891 -0.05055863 -
0.61063844]
[ 0.25939748 -0.25303692 -0.16946845 -0.5932543
0.95880073]
[ 0.9266262 0.21439868 -0.61490893 -0.52377903 -
1.2263945 ]
[ 1.0817996 -0.07082981 0.66491896 -0.509441
0.42131773]]
biases
[-0.09372782 0. -0.24866071 0. -
0.20741874]

```

```

----- layer 6 -----
weights
[[-0.35625547  0.707156   -0.10775853  0.62058973]
 [ 0.4375862  -0.15788895 -0.4895757  -0.5894324 ]
 [-0.4803561  -1.1298519  -0.93598634 -0.00581208]
 [-0.40564403 -0.42131674  0.06275553 -0.25271308]
 [-0.6282842  -0.75069356  0.19843917 -0.19431809]]
biases
[-0.33736056  0.46992734  0.30303562 -0.30433118]

```

Hasil perhitungan dari model C5 adalah sebagai berikut.

```

1/1 [=====] - 0s 91ms/step
1 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 21ms/step
2 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 24ms/step
3 X=[2 2 3 3 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 21ms/step
4 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 18ms/step
5 X=[2 2 3 3 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 20ms/step
6 X=[3 3 3 3 3], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 26ms/step
7 X=[2 2 2 3 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 25ms/step
8 X=[2 2 2 3 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 24ms/step
9 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
10 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 26ms/step
11 X=[2 2 2 3 3], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 30ms/step
12 X=[2 2 2 3 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 30ms/step

```

```

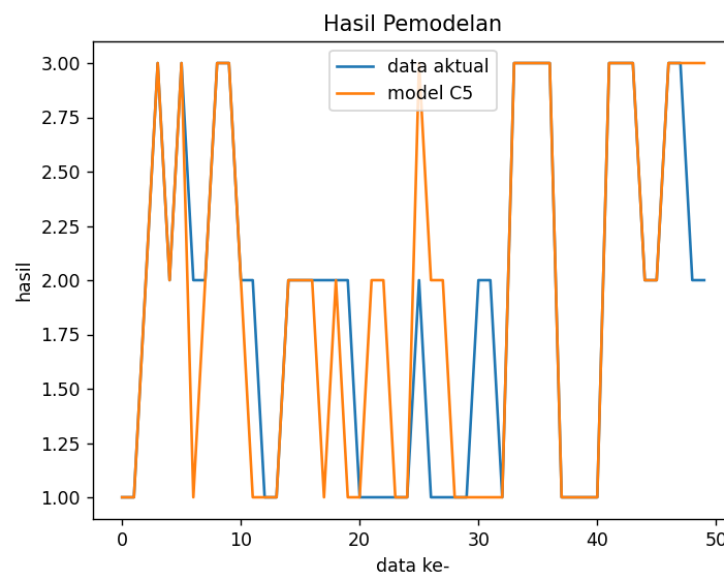
13 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 20ms/step
14 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 33ms/step
15 X=[2 2 2 3 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 19ms/step
16 X=[2 2 2 3 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 34ms/step
17 X=[2 2 2 3 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 33ms/step
18 X=[2 2 2 3 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 19ms/step
19 X=[2 2 2 3 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 21ms/step
20 X=[2 2 2 3 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 20ms/step
21 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 26ms/step
22 X=[1 1 1 1 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 31ms/step
23 X=[1 1 1 1 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 19ms/step
24 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 18ms/step
25 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 31ms/step
26 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
27 X=[1 1 1 1 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 17ms/step
28 X=[1 1 1 1 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 17ms/step
29 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 16ms/step
30 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 31ms/step
31 X=[2 2 2 3 1], Predicted=(array([1], dtype=int64),)

```

```
1/1 [=====] - 0s 24ms/step
32 X=[2 2 2 3 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 20ms/step
33 X=[1 1 1 2 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 18ms/step
34 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
35 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
36 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 32ms/step
37 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
38 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 22ms/step
39 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 22ms/step
40 X=[1 1 1 2 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 21ms/step
41 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 25ms/step
42 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 17ms/step
43 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 30ms/step
44 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 27ms/step
45 X=[2 2 2 3 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 19ms/step
46 X=[2 2 2 3 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 21ms/step
47 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 18ms/step
48 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
49 X=[2 2 1 2 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
```

```
50 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)
Process finished with exit code 0
```

Grafik perbandingan antara data aktual dengan data hasil prediksi dari model C5 dapat dilihat pada gambar 4.51. Garis warna biru menunjukkan data aktual sedangkan garis warna oranye menunjukkan data hasil prediksi. Ada 50 data *testing* yang digunakan, yang diolah untuk mengetahui hasil prediksi yang berupa angka 1 (rusak ringan), 2 (rusak sedang), dan 3 (rusak berat).



Gambar 4. 51 Grafik Hasil Model C5

p. Model D1

Model D1 merupakan model *forward propagation in neural network* (FPNN) dengan *epoch* sebanyak 100 dan *hidden layer* sejumlah 1. Data *testing* akan diolah menggunakan model ini untuk mengetahui hasil prediksinya. Hasil dari perhitungan menggunakan model D1 sendiri

didapatkan akurasi di *epoch* ke-100 sebesar 65% dengan *loss* sebesar 0,7869.

```
Epoch 100/100
10/10 [=====] - 0s 665us/step -
loss: 0.7869 - accuracy: 0.6500
```

Pada model ini, nilai bobot dan nilai bias awal akan ditentukan secara random, hingga didapatkan nilai bobot dan bias dengan *loss function* terendah untuk tiap *hidden layer* dan *output layer*. Nilai bobot dan nilai bias akhir yang terbentuk pada model D1 antara lain sebagai berikut.

```
----- layer 1 -----
weights
[[-0.51581985 -0.26781744  0.5341617   0.4098686  -
0.17824449]
 [ 0.4511064  -0.41520342 -0.45796198  0.6049675
0.53432995]
 [-0.40598714 -0.39601073 -0.58567286  0.57807827
0.36523432]
 [ 0.6700914  -0.53448486 -0.43879238  0.5617048
0.45401612]
 [-0.3326907  0.7330725   0.06932139 -0.09575544 -
0.6690757  ]]
biases
[ 0.21170108  0.          0.          0.08897647 -
0.3327138  ]
----- layer 2 -----
weights
[[ 0.1565504  1.212284  -0.1780935  -0.5822979  ]
 [ 0.14083332  0.34204853 -0.05704474 -0.1615094  ]
 [-0.57458174 -0.26118815  0.4243691   0.21062016]
 [-0.93164617 -0.5389401  -0.11856207 -0.16876422]
 [ 0.35796502 -0.38567895 -0.5618466   0.10546473]]
```

```
biases
```

```
[-0.35567474  0.5961664 -0.10635242  0.18804812]
```

Berikut merupakan hasil perhitungan metode FPNN yang didapatkan dari model D1.

```
1/1 [=====] - 0s 65ms/step
1 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 24ms/step
2 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 20ms/step
3 X=[2 2 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
4 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 18ms/step
5 X=[2 2 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
6 X=[3 3 3 3 3], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
7 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 27ms/step
8 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
9 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
10 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
11 X=[2 2 2 3 3], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
12 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 36ms/step
13 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 24ms/step
14 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 20ms/step
15 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
```

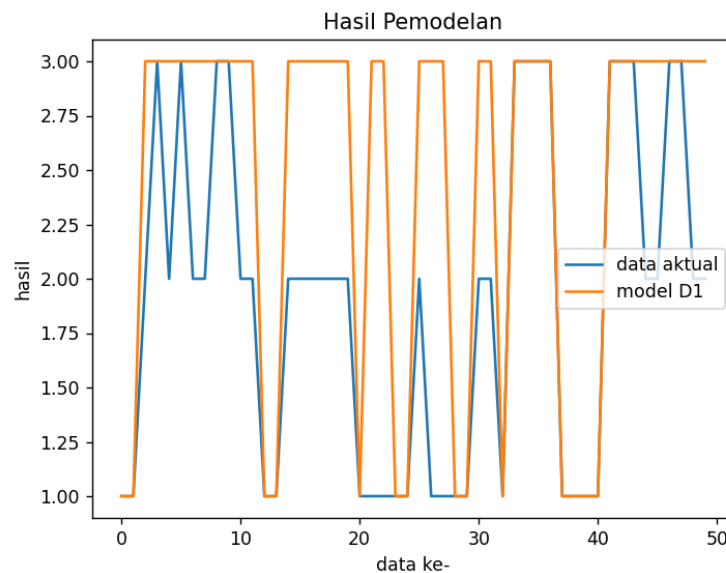


```
1/1 [=====] - 0s 33ms/step
16 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 28ms/step
17 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 33ms/step
18 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 32ms/step
19 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
20 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
21 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 19ms/step
22 X=[1 1 1 1 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 18ms/step
23 X=[1 1 1 1 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 31ms/step
24 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 18ms/step
25 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 17ms/step
26 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
27 X=[1 1 1 1 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
28 X=[1 1 1 1 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
29 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 26ms/step
30 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 21ms/step
31 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 27ms/step
32 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 18ms/step
33 X=[1 1 1 2 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 28ms/step
```

```
34 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
35 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 28ms/step
36 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
37 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 31ms/step
38 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 26ms/step
39 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 29ms/step
40 X=[1 1 1 2 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 19ms/step
41 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 18ms/step
42 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
43 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
44 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
45 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
46 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 18ms/step
47 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
48 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 25ms/step
49 X=[2 2 1 2 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 25ms/step
50 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)

Process finished with exit code 0
```

Gambar 4.52 menunjukkan perbandingan antara data aktual dengan data hasil prediksi dari model D1. Garis warna biru menunjukkan data aktual dan garis warna oranye menunjukkan data hasil prediksi.



Gambar 4. 52 Grafik Hasil Model D1

#### q. Model D2

Model D2 adalah model FPNN yang memiliki *epoch* sebanyak 100 dengan *hidden layer* sejumlah 2. Data *testing* juga diolah menggunakan model ini untuk menghitung hasil prediksi. Hasil dari perhitungan menggunakan model D2 antara lain akurasi di *epoch* ke-100 sebesar 73% dengan *loss* sebesar 0,5807.

```
Epoch 100/100
10/10 [=====] - 0s 1ms/step -
loss: 0.5807 - accuracy: 0.7300
```

Nilai bobot dan nilai bias awal pada model ini juga akan ditentukan secara random, hingga didapatkan nilai bobot dan bias dengan *loss function*

terendah untuk tiap *hidden layer* dan *output layer*. Berikut merupakan nilai bobot dan nilai bias akhir yang terbentuk pada model D2.

```

----- layer 1 -----
weights
[[-0.27739927  0.85602134  0.5728277  -0.5230965  -
0.4642846 ]
 [-0.11796063  0.6550841  -0.00628849 -0.11010057 -
0.72684705]
 [-0.3403452  -0.36157253  0.26170146 -0.36535192
0.099828  ]
 [-0.4780771  -0.27170214  0.40534642  0.5827687
0.76984286]
 [-0.10379857 -0.1505653  -0.15319663 -0.6998186  -
0.6652234  ]]
biases
[ 0.          -0.5915974 -0.3504077  0.          0.          ]
----- layer 2 -----
weights
[[-0.2115469  -0.25024492 -0.37466174  0.5583168  -
0.39110032]
 [ 0.15360206  0.16784978 -0.01073849  0.05635997
1.598298  ]
 [-0.2562704  -0.37058237 -0.13121128  0.43536592
0.33026904]
 [-0.3122968  -0.08790964  0.5277208  0.72201824  -
0.30929118]
 [-0.43964964 -0.48824292  0.57423913  0.660048  -
0.63094914]]
biases
[ 0.          0.          0.          -0.3131197  -
0.46934682]
----- layer 3 -----
weights
[[-0.04991561 -0.41008267  0.22423625  0.7616508 ]
 [-0.6108792  -0.5678227  0.7650472  -0.14906037]
 [ 0.7513039  -0.39159194 -0.39620382  0.2722603 ]

```

```

[-1.1818691 -0.5902299  0.5382491  0.15839997]
[-0.7638122 -0.6238526  0.36698124  0.9988029 ]]
biases
[-0.8075538  0.69417906 -0.11629573 -0.22348496]

```

Hasil perhitungan dari model D2 antara lain sebagai berikut.

```

1/1 [=====] - 0s 76ms/step
1 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 36ms/step
2 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 20ms/step
3 X=[2 2 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 32ms/step
4 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 33ms/step
5 X=[2 2 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
6 X=[3 3 3 3 3], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 33ms/step
7 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 29ms/step
8 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 26ms/step
9 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 25ms/step
10 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 25ms/step
11 X=[2 2 2 3 3], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
12 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 26ms/step
13 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 28ms/step
14 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 29ms/step
15 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)

```

```

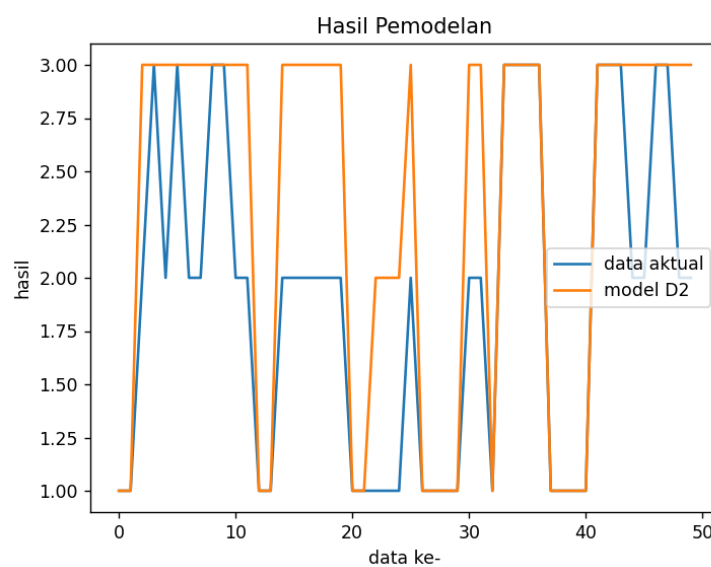
1/1 [=====] - 0s 25ms/step
16 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
17 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
18 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 28ms/step
19 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 33ms/step
20 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
21 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 21ms/step
22 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 31ms/step
23 X=[1 1 1 1 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 30ms/step
24 X=[1 1 1 2 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 19ms/step
25 X=[1 1 1 2 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 20ms/step
26 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 29ms/step
27 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 39ms/step
28 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 19ms/step
29 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 32ms/step
30 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 25ms/step
31 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
32 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
33 X=[1 1 1 2 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 26ms/step

```

```
34 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 28ms/step
35 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
36 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
37 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 26ms/step
38 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 27ms/step
39 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 31ms/step
40 X=[1 1 1 2 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 22ms/step
41 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 23ms/step
42 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 28ms/step
43 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 18ms/step
44 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 30ms/step
45 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 27ms/step
46 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 28ms/step
47 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 27ms/step
48 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 26ms/step
49 X=[2 2 1 2 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 25ms/step
50 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)

Process finished with exit code 0
```

Gambar 4.53 merupakan grafik perbandingan antara data aktual dengan data hasil prediksi dari model D2. Data aktual ditunjukkan oleh garis warna biru dan data hasil prediksi ditunjukkan oleh garis warna oranye. Ada 50 data *testing* yang digunakan, yang diolah menggunakan model D2 untuk mengetahui hasil prediksinya. *Output* yang dihasilkan berupa angka 1 (rusak ringan), 2 (rusak sedang), atau 3 (rusak berat).



Gambar 4. 53 Grafik Hasil Model D2

#### r. Model D3

Model D3 merupakan model *forward propagation in neural network* (FPNN) yang memiliki *epoch* sebanyak 100 dan *hidden layer* sejumlah 3. Model ini juga akan digunakan untuk mengolah data *testing*, untuk kemudian mengetahui hasil prediksinya. Hasil dari perhitungan menggunakan model D3, didapatkan bahwa akurasi di *epoch* ke-100 adalah sebesar 74% dengan *loss* sebesar 0,6209.



```
Epoch 100/100
10/10 [=====] - 0s 1ms/step -
loss: 0.6209 - accuracy: 0.7400
```

Pada model D3 sendiri, nilai bobot dan nilai bias awal akan ditentukan secara random, hingga didapatkan nilai bobot dan bias dengan *loss function* terendah untuk tiap *hidden layer* dan *output layer*. Berikut merupakan nilai bobot dan nilai bias akhir yang terbentuk pada model ini.

```
----- layer 1 -----
weights
[[-0.25471216  0.7992849 -0.12121184 -0.34091023 -
0.70222247]
 [-0.20041735  1.1785312 -0.6089584 -0.41252375 -
0.13894077]
 [ 0.9247389  0.18042348  0.13077962  0.42873928
0.47492617]
 [ 0.5345291  0.5728621 -0.42565536  0.3303196 -
0.2047232 ]
 [ 0.96003497  0.23129106  0.69668907 -0.4319567
0.13794987]]
biases
[ 0.14852561 -0.59642124  0.11443179 -0.008829 -
0.06334031]
----- layer 2 -----
weights
[[ 0.8704144 -0.6315966 -0.65149367  0.3840257
0.6186061 ]
 [-0.51082456 -0.03279185 -0.7259129 -0.27966207
0.57192385]
 [ 0.68681186 -0.18076807  0.2799436  0.5147027
0.3754189 ]
 [ 0.59802103 -0.70473033  0.7247932 -0.42700633 -
0.08553261]
```

```

[-0.61239785  0.66604686 -0.37259758  0.1722572
0.62060684]]
biases
[ 0.16325773  0.          0.          0.58930993 -
0.41966784]
----- layer 3 -----
weights
[[-0.44081908 -0.38461012 -0.17372018  0.7165208
0.6539985 ]
[ 0.3533913  -0.4717916  0.31323886 -0.2032615
0.1948322 ]
[ 0.07404637 -0.74052024  0.29636443  0.47636676 -
0.705744 ]
[ 0.64656657 -1.2826093  -0.55690056  0.3878219  -
0.6713477 ]
[-0.20338109  0.5829804  -0.18708795 -0.7541144  -
0.18124986]]
biases
[-0.06476155 -0.37631705  0.          0.          0.
]
----- layer 4 -----
weights
[[-0.09698178 -0.8333956  0.05184383  0.63425624]
[-0.6831262  -1.1326345  0.04178076  0.73975915]
[ 0.2513553  -0.19088107  0.7667315  -0.401753 ]
[-0.22990674  0.491552  -0.1094026  0.4450662 ]
[-0.37320146  0.29897714  0.30244184  0.47001922]]
biases
[-0.60651916  0.662529  0.27587518 -0.16904336]

```

Berikut adalah hasil perhitungan prediksi pada model ini.

```

1/1 [=====] - 0s 95ms/step
1 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 19ms/step
2 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 22ms/step

```

```
3 X=[2 2 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
4 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
5 X=[2 2 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
6 X=[3 3 3 3 3], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
7 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
8 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
9 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 35ms/step
10 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
11 X=[2 2 2 3 3], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
12 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 29ms/step
13 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 20ms/step
14 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 21ms/step
15 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
16 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
17 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
18 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
19 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
20 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
21 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
```

```
1/1 [=====] - 0s 18ms/step
22 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 20ms/step
23 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 21ms/step
24 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 28ms/step
25 X=[1 1 1 2 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 20ms/step
26 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
27 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 20ms/step
28 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 27ms/step
29 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 19ms/step
30 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 32ms/step
31 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
32 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 29ms/step
33 X=[1 1 1 2 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 29ms/step
34 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
35 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
36 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
37 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 18ms/step
38 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 20ms/step
39 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 20ms/step
```

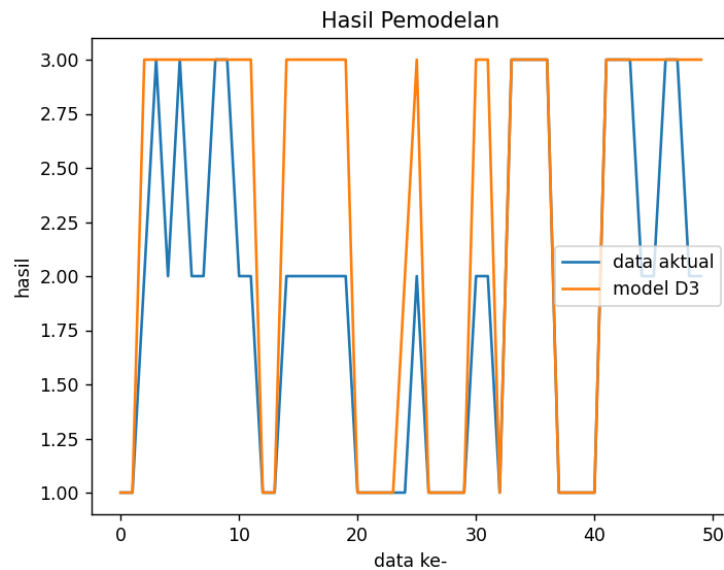
```

40 X=[1 1 1 2 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 35ms/step
41 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 19ms/step
42 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 25ms/step
43 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 35ms/step
44 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
45 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
46 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
47 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
48 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 34ms/step
49 X=[2 2 1 2 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 36ms/step
50 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)

Process finished with exit code 0

```

Gambar 4.54 menunjukkan grafik perbandingan antara data aktual dengan data hasil prediksi dari model D3. Data aktual ditunjukkan oleh garis warna biru dan data hasil prediksi ditunjukkan oleh garis warna oranye. Ada 50 data *testing* yang digunakan, yang diolah untuk mengetahui hasil prediksinya. *Output* yang dihasilkan berupa angka 1 (rusak ringan), 2 (rusak sedang), atau 3 (rusak berat).



Gambar 4. 54 Grafik Hasil Model D3

## s. Model D4

Model D4 merupakan model *forward propagation in neural network* (FPNN) yang memiliki *epoch* sebanyak 100 dan *hidden layer* sejumlah 4. Model ini juga akan digunakan untuk mengolah data *testing*, untuk kemudian mengetahui hasil prediksinya. Hasil dari perhitungan menggunakan model D4, didapatkan bahwa akurasi di *epoch* ke-100 adalah sebesar 73% dengan *loss* sebesar 0,5317.

```
Epoch 100/100
10/10 [=====] - 0s 2ms/step -
loss: 0.5317 - accuracy: 0.7300
```

Seperti pada model lainnya, nilai bobot dan nilai bias awal pada model ini akan ditentukan secara random, hingga didapatkan nilai bobot dan bias dengan *loss function* terendah untuk tiap *hidden layer* dan *output layer*.

Nilai bobot dan nilai bias akhir yang terbentuk pada model D4 antara lain sebagai berikut.

```

----- layer 1 -----
weights
[[ 0.41195464  0.00637337 -0.60956746  0.9655982  -
0.03848511]
 [ 0.91940886  0.75566804  0.32549906  0.21684627  -
0.52131176]
 [ 0.34424448  0.59575444  0.20647621 -0.18908074  -
0.21773565]
 [-0.43748832 -0.29410586  0.30633414  0.13556395  -
0.3270085 ]
 [-0.3548019  0.2161296  -0.6981292  0.39941978
0.3152007 ]]
biases
[-0.6029978 -0.63503486  0.          -0.6225183  0.
]
----- layer 2 -----
weights
[[-0.47328454  0.27516317  1.2073665  0.59993505
0.58444417]
 [-0.5947358  -0.10937476  0.9221739  -0.77025455
-0.762504 ]
 [-0.19136786 -0.7279625  0.2727604  0.10662913
0.4006015 ]
 [-0.43028164 -0.7738334  0.63588566 -0.3494648
-0.48504615]
 [ 0.6565064  -0.4667013  -0.5842813  0.1434477
0.717672  ]]
biases
[ 0.          0.          -0.6133993  0.          0.          ]
----- layer 3 -----
weights
[[ 0.53417623  0.09409338 -0.4290057  -0.5457715
0.37046993]
 [ 0.7423458  0.47362864  0.56392384  0.3502258

```

```

-0.1643058 ]
  [-0.0435425  0.62394077 -0.23189825  0.61710644
-0.5912398 ]
  [-0.49603042  0.4429748  -0.13212341 -0.5269456
0.21538526]
  [-0.1352936  0.20965117  0.43046665 -0.64712644
0.47731733]]
biases
[ 0.          -0.5373132  0.          -0.5390308  0.          ]
----- layer 4 -----
weights
[[ [ 0.23949552 -0.5500359  -0.25932628  0.7604592
-0.27767465]
  [ 0.5290617  0.6640864  -0.37402883 -0.586987
-0.2889171 ]
  [-0.11112386 -0.09113353  0.40299308 -0.2265917
0.01923239]
  [ 0.5196203  0.64043236  0.12422901 -0.25916934
-0.05440176]
  [ 0.5278994  0.3890425  0.11222214  0.76299405
0.2726128 ]]
biases
[-0.02419926 -0.36174935  0.          0.          0.          ]
]
----- layer 5 -----
weights
[[[-0.9588527  -0.5073572  0.82186425  0.30335382]
  [-0.5780118  -0.1914551  -0.43527758  0.55203736]
  [ 0.07540524  0.80843246  0.14601111  0.81000984]
  [ 0.44983363 -0.8101031  0.55260515 -0.5963213 ]
  [ 0.7185435  0.11955649 -0.71199435  0.6114639 ]]
biases
[-0.59849095  0.643486  0.15241337 -0.5734325 ]

```

Berikut merupakan hasil perhitungan prediksi metode FPNN dari model D4.



```
1/1 [=====] - 0s 124ms/step
1 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 31ms/step
2 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 32ms/step
3 X=[2 2 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 25ms/step
4 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
5 X=[2 2 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 37ms/step
6 X=[3 3 3 3 3], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 26ms/step
7 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 40ms/step
8 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
9 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
10 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
11 X=[2 2 2 3 3], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
12 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
13 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 21ms/step
14 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 26ms/step
15 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 38ms/step
16 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 18ms/step
17 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
18 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
```

```
19 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
20 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 26ms/step
21 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 18ms/step
22 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 26ms/step
23 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 24ms/step
24 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 20ms/step
25 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 22ms/step
26 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 32ms/step
27 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 28ms/step
28 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 29ms/step
29 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 24ms/step
30 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 25ms/step
31 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 34ms/step
32 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 28ms/step
33 X=[1 1 1 2 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 29ms/step
34 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 27ms/step
35 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 32ms/step
36 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 29ms/step
37 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
```

```

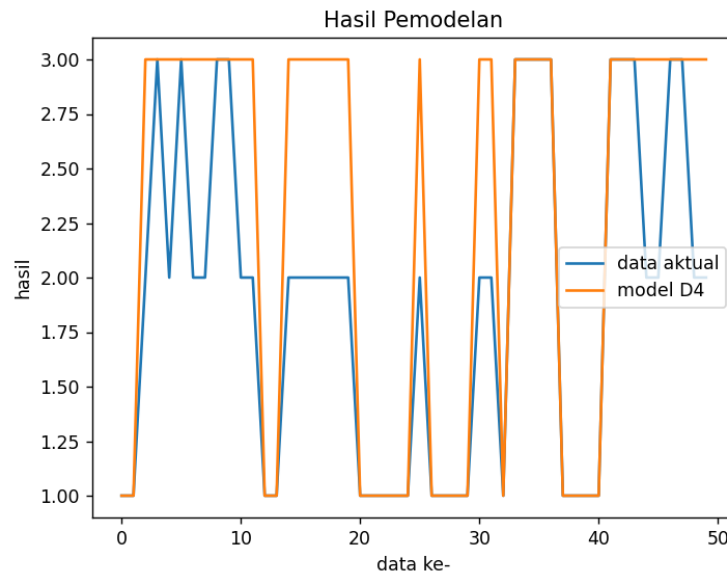
1/1 [=====] - 0s 21ms/step
38 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 23ms/step
39 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 26ms/step
40 X=[1 1 1 2 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 20ms/step
41 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 29ms/step
42 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 36ms/step
43 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 27ms/step
44 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 31ms/step
45 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
46 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 35ms/step
47 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
48 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 35ms/step
49 X=[2 2 1 2 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 40ms/step
50 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)

Process finished with exit code 0

```

Gambar 4.55 merupakan grafik perbandingan antara data aktual dengan data hasil prediksi dari model D4. Data aktual ditunjukkan oleh garis warna biru dan data hasil prediksi ditunjukkan oleh garis warna oranye. Ada 50 data *testing* yang digunakan, yang diolah menggunakan model D4 untuk

mengetahui hasil prediksinya. *Output* yang dihasilkan berupa angka 1 (rusak ringan), 2 (rusak sedang), atau 3 (rusak berat).



Gambar 4.55 Grafik Hasil Model D4

#### t. Model D5

Model D5 merupakan model *forward propagation in neural network* (FPNN) yang memiliki *epoch* sebanyak 100 dan *hidden layer* sejumlah 5. Model ini juga akan digunakan untuk mengolah data *testing*, untuk kemudian mengetahui hasil prediksinya. Hasil dari perhitungan menggunakan model D5, didapatkan bahwa akurasi di *epoch* ke-100 adalah sebesar 73% dengan *loss* sebesar 0,5757.

```
Epoch 100/100
10/10 [=====] - 0s 1ms/step -
loss: 0.5757 - accuracy: 0.7300
```

Pada model ini, nilai bobot dan nilai bias awal akan ditentukan secara random, hingga didapatkan nilai bobot dan bias dengan *loss function*

terendah untuk tiap *hidden layer* dan *output layer*. Berikut merupakan nilai bobot dan nilai bias akhir yang terbentuk pada model D5.

```

----- layer 1 -----
weights
[[-0.6199584   0.7050071   0.8582944  -0.39356264
0.6907331 ]
 [ -0.29468516 -0.31381297   0.90463364 -0.4982065  -
0.09353136]
 [ -0.20480007   0.68434554   0.648758    0.19588709
0.19724517]
 [ 0.07121819   0.04278979 -0.23747675 -0.6249957  -
0.2886344 ]
 [ 0.21773231   0.03849822 -0.26147842   0.5381179  -
0.21648642]]
biases
[ 0.          -0.34206393 -0.39210334  0.          -
0.2953926 ]
----- layer 2 -----
weights
[[-0.24007171   0.13723606 -0.01891863   0.3482492  -
0.5735233 ]
 [ 0.73959154   0.7420393  -0.6555332  -0.36771488  0.57901
]
 [ 0.17760088   0.4989625   0.18448366 -0.5702936  -
0.52430415]
 [-0.5526264   0.6867957   0.48661184   0.41719997
0.08743072]
 [ 1.3903594  -0.12898423   0.5189093   0.67383254  -
0.12453055]]
biases
[-0.56877524 -0.30398628 -0.04336359  0.          0.
]
----- layer 3 -----
weights
[[ 0.82355696 -0.6693774  -0.5720501   0.5833534
0.43482944]

```

```

[ 0.6445476 -0.66977423 0.05217475 0.6598515
0.8358274 ]
[-0.3394745 0.12220234 -0.6921148 -0.48943725 -
0.41557157]
[ 0.23341608 0.04116106 0.4333737 -0.04604381 -
0.46162468]
[ 0.01286948 -0.34836927 -0.09534067 -0.07107878 -
0.4122809 ]]
biases
[-0.34280488 0. 0. -0.08029202 -
0.10408083]
----- layer 4 -----
weights
[[ 0.08476365 0.09389494 0.54696983 -0.5273895
0.13544947]
[-0.7657015 -0.21270132 0.6354791 0.44893026
0.75869405]
[-0.6057018 0.21719342 -0.07625401 -0.68797356 -
0.29000005]
[-0.03581357 0.40434343 0.184516 -0.440513 -
0.7486832 ]
[-0.46378395 -0.42614603 0.2631167 0.5483551
0.5013759 ]]
biases
[ 0. 0.01073736 -0.23460814 0. 0.
]
----- layer 5 -----
weights
[[ -0.46700493 0.02791429 0.24408686 0.57508993
0.67453253]
[-0.7526235 0.41906726 -0.18936215 0.31011224 -
0.10688126]
[ 0.5536024 -0.6491828 0.93277144 -0.36676803 -
0.32720977]
[ 0.18077523 -0.09911054 0.45607913 0.46502686 -
0.6363656 ]

```

```

[-0.37069428 -0.26473033  0.6855668  0.38223815 -
0.11809766]]
biases
[ 0.2763928  0.          -0.4535846  0.          0.          ]
----- layer 6 -----
weights
[[-1.3026639  0.41524026  0.18095225  0.31358287]
 [-0.75819063  0.5121465  0.6833482  -0.42266735]
 [-0.07746622 -0.48052114  0.42366272  0.97327816]
 [ 0.7544553  0.750278  0.6823392  0.2245413 ]
 [-0.7632012 -0.6607747  0.03587401  0.02897751]]
biases
[-0.48326877  0.440059  0.43922848 -0.4885744 ]

```

Berikut merupakan hasil perhitungan prediksi dari model D5.

```

1/1 [=====] - 0s 126ms/step
1 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 27ms/step
2 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 21ms/step
3 X=[2 2 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 25ms/step
4 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
5 X=[2 2 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 26ms/step
6 X=[3 3 3 3 3], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
7 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 35ms/step
8 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 28ms/step
9 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 33ms/step
10 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 25ms/step

```

```
11 X=[2 2 2 3 3], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
12 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 26ms/step
13 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 30ms/step
14 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 28ms/step
15 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
16 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 32ms/step
17 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 36ms/step
18 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 30ms/step
19 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 25ms/step
20 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 26ms/step
21 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 21ms/step
22 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 24ms/step
23 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 33ms/step
24 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 27ms/step
25 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 28ms/step
26 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
27 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 18ms/step
28 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 22ms/step
29 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
```



```

1/1 [=====] - 0s 25ms/step
30 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 30ms/step
31 X=[2 2 2 3 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 40ms/step
32 X=[2 2 2 3 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 18ms/step
33 X=[1 1 1 2 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 18ms/step
34 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
35 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
36 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 26ms/step
37 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
38 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 25ms/step
39 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 20ms/step
40 X=[1 1 1 2 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 19ms/step
41 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 22ms/step
42 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 32ms/step
43 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
44 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 34ms/step
45 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
46 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
47 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 26ms/step

```

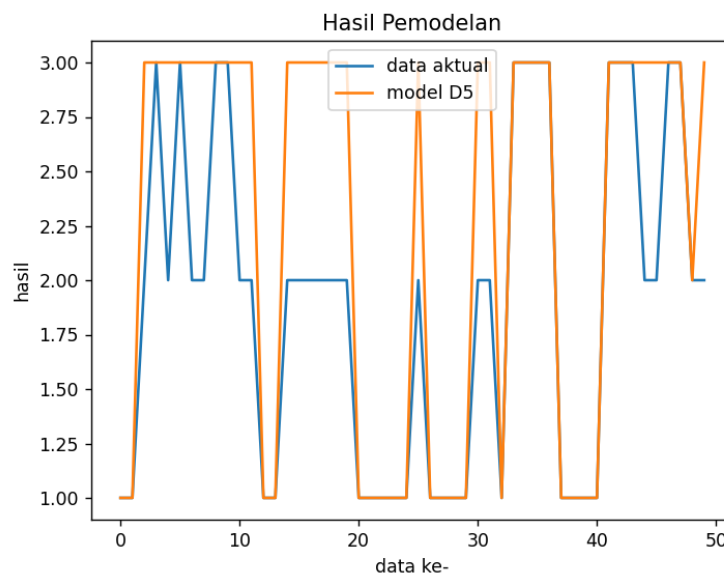
```

48 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
49 X=[2 2 1 2 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 25ms/step
50 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)

Process finished with exit code 0

```

Gambar 4.56 merupakan grafik perbandingan antara data aktual dengan data hasil prediksi dari model D5. Data aktual ditunjukkan oleh garis warna biru dan data hasil prediksi ditunjukkan oleh garis warna oranye. Ada 50 data *testing* yang digunakan, yang diolah menggunakan model D5 untuk mengetahui hasil prediksinya. *Output* yang dihasilkan berupa angka 1 (rusak ringan), 2 (rusak sedang), atau 3 (rusak berat).



Gambar 4. 56 Grafik Hasil Model D5

#### u. Model E1

Model E1 adalah model FPNN dengan *epoch* sebanyak 125 dan *hidden layer* sejumlah 1. Model ini juga digunakan untuk mengolah data

*testing*, untuk mengetahui hasil prediksinya. Hasil dari perhitungan menggunakan model E1, didapatkan bahwa akurasi di *epoch* ke-125 adalah sebesar 72% dengan *loss* sebesar 0,6905.

```
Epoch 125/125
10/10 [=====] - 0s 2ms/step -
loss: 0.6905 - accuracy: 0.7200
```

Nilai bobot dan nilai bias awal model E1 juga akan ditentukan secara random, hingga didapatkan nilai bobot dan bias dengan *loss function* terendah untuk tiap *hidden layer* dan *output layer*. Berikut merupakan nilai bobot dan nilai bias akhir yang terbentuk pada model E1.

```
----- layer 1 -----
weights
[[ 0.58449376  0.21908912  0.05287781  0.29270768 -
0.40266907]
 [-0.61901397 -0.2147591  0.9371679  0.23469669
0.8215907 ]
 [-0.7182451  -0.02513008  0.9632355  0.09664831
0.11329681]
 [-0.09579372  0.6980282  -0.39709684  0.41929457
0.5942926 ]
 [ 0.2878536  -0.5270742  0.49440527 -0.04005756 -
0.38818988]]
biases
[ 0.          -0.00960802 -0.70070326  0.08966997
0.29593825]
----- layer 2 -----
weights
[[ 0.45661294  0.00757766 -0.04158729 -0.00170201]
 [ 0.5635061  0.13555613  0.70772684 -0.02143221]
 [-0.58907765 -0.40252128  0.64250195  0.93916166]
 [ 0.25612378 -0.03176538  0.18633235 -0.19995856]
```

```

[-1.2816612  0.4860721 -0.805428  -0.10281581]]
biases
[-3.4487903e-01  6.8404168e-01  6.5149507e-04 -3.6444491e-
01]

```

Hasil dari perhitungan prediksi menggunakan model E1 adalah sebagai berikut.

```

1/1 [=====] - 0s 84ms/step
1 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 33ms/step
2 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 29ms/step
3 X=[2 2 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 29ms/step
4 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 34ms/step
5 X=[2 2 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
6 X=[3 3 3 3 3], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
7 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
8 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
9 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
10 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 26ms/step
11 X=[2 2 2 3 3], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 27ms/step
12 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 35ms/step
13 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 26ms/step
14 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)

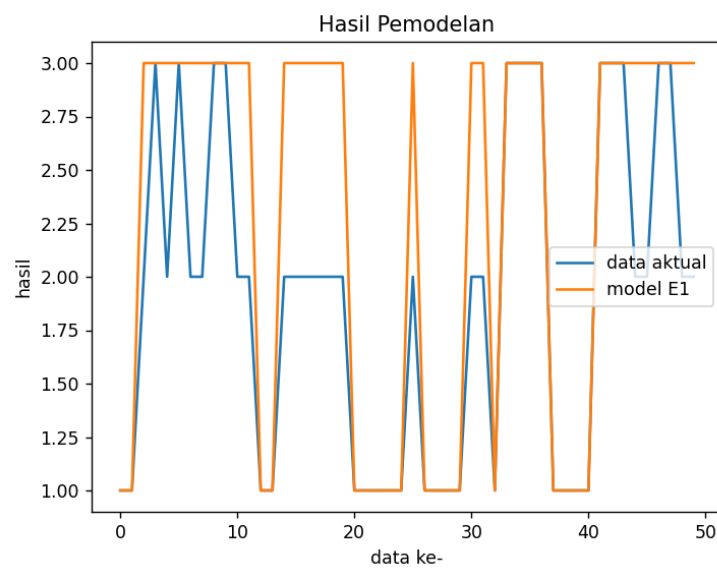
```

```
1/1 [=====] - 0s 19ms/step
15 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 30ms/step
16 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
17 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
18 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 33ms/step
19 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
20 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 28ms/step
21 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 31ms/step
22 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 19ms/step
23 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 24ms/step
24 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 29ms/step
25 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 26ms/step
26 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
27 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 20ms/step
28 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 23ms/step
29 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 20ms/step
30 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 30ms/step
31 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
32 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
```

```
33 X=[1 1 1 2 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 21ms/step
34 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 29ms/step
35 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
36 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 26ms/step
37 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 32ms/step
38 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 41ms/step
39 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 29ms/step
40 X=[1 1 1 2 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 22ms/step
41 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 26ms/step
42 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 41ms/step
43 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
44 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 35ms/step
45 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
46 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 37ms/step
47 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 30ms/step
48 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 33ms/step
49 X=[2 2 1 2 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
50 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)

Process finished with exit code 0
```

Gambar 4.57 merupakan grafik perbandingan antara data aktual dengan data hasil prediksi dari model E1. Data aktual ditunjukkan oleh garis warna biru dan data hasil prediksi ditunjukkan oleh garis warna oranye. Ada 50 data *testing* yang digunakan, yang diolah menggunakan model ini untuk mengetahui hasil prediksinya. *Output* yang dihasilkan berupa angka 1 (rusak ringan), 2 (rusak sedang), atau 3 (rusak berat).



Gambar 4. 57 Grafik Hasil Model E1

#### v. Model E2

Model E2 sendiri merupakan model FPNN dengan *epoch* sebanyak 125 dan *hidden layer* sejumlah 2. Model E2 juga digunakan untuk mengolah data *testing*, untuk mengetahui hasil perhitungan prediksi metode FPNN. Hasil dari perhitungan menggunakan model E2, didapatkan bahwa akurasi di *epoch* ke-125 adalah sebesar 84% dengan *loss* sebesar 0,4720.

```
Epoch 125/125
10/10 [=====] - 0s 1ms/step -
loss: 0.4720 - accuracy: 0.8400
```

Pada model E2, nilai bobot dan nilai bias awal akan ditentukan secara random, hingga didapatkan nilai bobot dan bias dengan *loss function* terendah untuk tiap *hidden layer* dan *output layer*. Berikut merupakan nilai bobot dan nilai bias akhir yang terbentuk pada model E2.

```

----- layer 1 -----
weights
[[ 0.0801271  0.5324078  1.0226283  0.5883699
 0.70957935]
 [-0.30475748 -0.14990199  0.12530342  0.16283707
 0.47131422]
 [ 0.3806919 -0.49480072  0.23657517  0.6190297
 0.48257563]
 [ 0.20782469 -0.51694953 -0.07623231 -0.05716458
 0.3278189 ]
 [-0.10131804  0.22154018 -0.56819904 -0.44524202
 0.46355373]]
biases
[-0.39565405 -0.00515922 -0.740901  -0.7305792  -
 0.5436438 ]
----- layer 2 -----
weights
[[ 0.71315944 -0.1172396  0.14250034  0.5565009
 0.36546052]
 [-0.0282678  0.18357015  0.68741107  0.3411367  -
 0.49143475]
 [-0.27883792  0.5620201  0.2026515  -0.44232672
 0.6589736 ]
 [ 0.19939804  0.91560835 -0.03685308 -0.1661892
 0.44081274]
 [-0.40235886  0.032876  -0.75239766 -0.27032667
 0.22804287]]
biases
[ 0.          -0.3586792  0.          0.          -0.5806989]

```



```

----- layer 3 -----
weights
[[ 0.5224966 -0.6305969  0.59567034 -0.11201954]
 [-1.6758825 -1.7657208  0.16535923  1.035059  ]
 [-0.4092304 -0.3291876 -0.44661814  0.22356641]
 [-0.06384271  0.1315667 -0.44262648 -0.38939705]
 [-0.9229396 -0.08352222  0.24364834  0.36792162]]
biases
[-1.0175332  0.8758171  0.22897904 -0.8186115 ]

```

Berikut merupakan hasil perhitungan prediksi model E2.

```

1/1 [=====] - 0s 110ms/step
1 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 21ms/step
2 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 19ms/step
3 X=[2 2 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
4 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
5 X=[2 2 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
6 X=[3 3 3 3 3], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
7 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
8 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 28ms/step
9 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
10 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
11 X=[2 2 2 3 3], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 26ms/step
12 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step

```

```
13 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 26ms/step
14 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 28ms/step
15 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 27ms/step
16 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
17 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 36ms/step
18 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
19 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
20 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
21 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 20ms/step
22 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 32ms/step
23 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 23ms/step
24 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 28ms/step
25 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 21ms/step
26 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
27 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 28ms/step
28 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 19ms/step
29 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 26ms/step
30 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 19ms/step
31 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
```

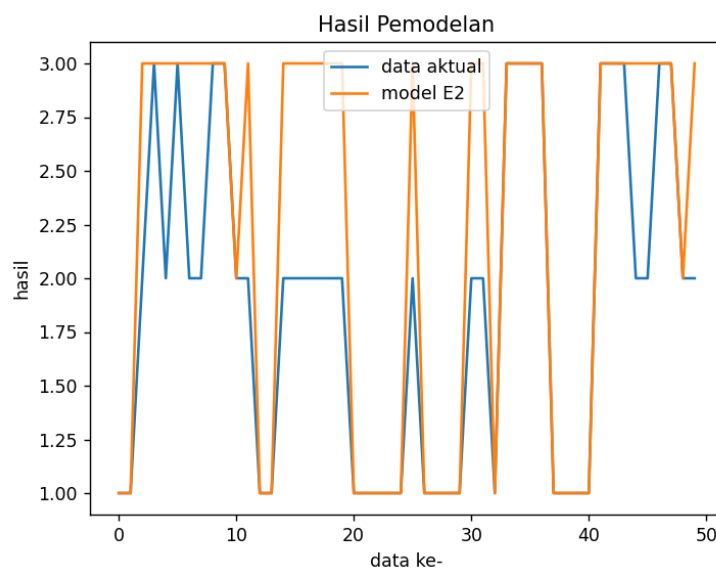
```

1/1 [=====] - 0s 21ms/step
32 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 26ms/step
33 X=[1 1 1 2 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 26ms/step
34 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
35 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
36 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
37 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
38 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 23ms/step
39 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 28ms/step
40 X=[1 1 1 2 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 27ms/step
41 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 29ms/step
42 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 30ms/step
43 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 39ms/step
44 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 35ms/step
45 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 29ms/step
46 X=[2 2 2 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
47 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 34ms/step
48 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 35ms/step
49 X=[2 2 1 2 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 31ms/step

```

```
50 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)
Process finished with exit code 0
```

Gambar 4.58 merupakan grafik perbandingan antara data aktual dengan data hasil prediksi dari model E2. Data aktual ditunjukkan oleh garis warna biru dan data hasil prediksi ditunjukkan oleh garis warna oranye. Ada 50 data *testing* yang digunakan, yang diolah menggunakan model ini untuk mengetahui hasil prediksinya. *Output* yang dihasilkan berupa angka 1 (rusak ringan), 2 (rusak sedang), atau 3 (rusak berat).



Gambar 4. 58 Grafik Hasil Model E2

#### w. Model E3

Model E3 merupakan model *forward propagation in neural network* (FPNN) yang memiliki *epoch* sebanyak 125 dan *hidden layer* sejumlah 3. Model ini juga akan digunakan untuk mengolah *data testing*, hingga diketahui hasil prediksinya. Berdasarkan hasil perhitungan yang dilakukan,

didapatkan bahwa akurasi di *epoch* ke-125 adalah sebesar 91% dengan loss sebesar 0,3118.

```
Epoch 125/125
10/10 [=====] - 0s 1ms/step -
loss: 0.3118 - accuracy: 0.9100
```

Pada model ini, nilai bobot dan nilai bias awal akan ditentukan secara random, hingga didapatkan nilai bobot dan bias dengan *loss function* terendah. Nilai bobot dan nilai bias akhir yang terbentuk pada masing-masing *hidden layer* dan *output layer* pada model E3 antara lain sebagai berikut.

```
----- layer 1 -----
weights
[[-0.6147351  0.8399657  0.43237746  0.6991087
0.267026  ]
 [-0.27339858  0.11580756  0.4882179  -0.5752413
0.623464  ]
 [-0.66745394  1.1220418  0.57433534  -0.72201437
0.98700523]
 [ 0.04391336  0.21056117 -0.1124926  -0.19735157 -
0.29222742]
 [ 0.11919445  0.54422015 -0.6085545  -0.5675912
0.8589748  ]]
biases
[ 0.          -0.06852141 -0.6645557  0.          -
0.4625077  ]
----- layer 2 -----
weights
[[ 0.16158605  0.24228406  0.7491534  0.3352852  -
0.09215558]
 [ 0.6880702  -0.58204377  0.01461404  0.32176447
0.770712  ]
```

```

[-0.21578129 -0.30587408  0.96905714  0.69183457
0.20646477]
[-0.5495753  0.6858555  0.67099345 -0.04818642
0.42645288]
[ 0.5944844 -0.6848178  0.5980057 -0.09493382
0.42611426]]
biases
[ 0.62694913  0.          -0.6936222  -0.76792425 -
0.38486135]
----- layer 3 -----
weights
[[ 0.7549653 -0.0770916  0.526613  0.13691738
0.4340095 ]
[ 0.696023  0.2167539  0.6155076  0.7582072
0.57935095]
[-0.9404008  0.5978851 -0.82125294 -0.8256095
0.35484442]
[-0.9361552 -0.43845916 -0.5840777  -0.2612914  -
0.16779867]
[ 0.03270468 -0.4671905  -0.5775009  -0.01990551
0.95717776]]
biases
[ 0.68290484 -0.01438088 -0.18341103 -0.1733955  -
0.4312408 ]
----- layer 4 -----
weights
[[[-0.6409315  1.6231202  0.8081365  -0.6548636 ]
[-0.5906116  0.4794904  -0.38430625 -0.23642734]
[ 0.31745237 -0.3071961  0.6526061  0.04298332]
[-0.17066132 -0.7074204  -0.6002199  -0.43578252]
[-0.6715058  -0.389862  0.04811149  0.33102003]]
biases
[-0.24319771  0.37569165 -0.02239688 -0.0573376 ]

```

Hasil perhitungan metode FPNN pada model E3 adalah sebagai berikut.

```
1/1 [=====] - 0s 92ms/step
1 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 42ms/step
2 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 27ms/step
3 X=[2 2 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
4 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
5 X=[2 2 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 25ms/step
6 X=[3 3 3 3 3], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 26ms/step
7 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 29ms/step
8 X=[2 2 2 3 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 23ms/step
9 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
10 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
11 X=[2 2 2 3 3], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 20ms/step
12 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
13 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 26ms/step
14 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 19ms/step
15 X=[2 2 2 3 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 19ms/step
16 X=[2 2 2 3 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 20ms/step
17 X=[2 2 2 3 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 20ms/step
18 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
```

```
19 X=[2 2 2 3 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 21ms/step
20 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 29ms/step
21 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 20ms/step
22 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 20ms/step
23 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 35ms/step
24 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 23ms/step
25 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 32ms/step
26 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
27 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 20ms/step
28 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 33ms/step
29 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 21ms/step
30 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 22ms/step
31 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
32 X=[2 2 2 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 25ms/step
33 X=[1 1 1 2 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 19ms/step
34 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
35 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
36 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
37 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
```



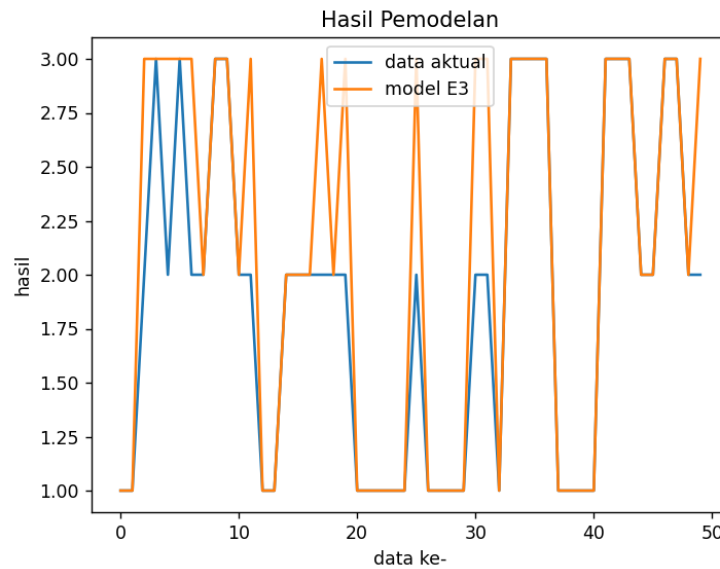
```

1/1 [=====] - 0s 37ms/step
38 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 26ms/step
39 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 33ms/step
40 X=[1 1 1 2 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 22ms/step
41 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 27ms/step
42 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 38ms/step
43 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
44 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
45 X=[2 2 2 3 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 27ms/step
46 X=[2 2 2 3 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 23ms/step
47 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
48 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
49 X=[2 2 1 2 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 20ms/step
50 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)

Process finished with exit code 0

```

Gambar 4.59 merupakan grafik perbandingan antara data aktual dengan hasil prediksi perhitungan model E3. Data aktual ditunjukkan oleh garis warna biru dan data hasil prediksi ditunjukkan oleh garis warna oranye. Ada 50 *data testing* yang digunakan, dan *output* yang dihasilkan berupa angka 1 (rusak ringan), 2 (rusak sedang), atau 3 (rusak berat).



Gambar 4. 59 Grafik Hasil Model E3

#### x. Model E4

Model E4 sendiri merupakan model FPNN dengan *epoch* sebanyak 125 dan *hidden layer* sejumlah 4. Model ini juga akan digunakan untuk mengolah *data testing*, untuk mengetahui hasil perhitungan prediksinya. Berdasarkan perhitungan dari model E4, didapatkan hasil berupa akurasi di *epoch* ke-125 adalah sebesar 93% dengan loss sebesar 0,3934.

```
Epoch 125/125
10/10 [=====] - 0s 1ms/step -
loss: 0.3934 - accuracy: 0.9300
```

Nilai bobot dan nilai bias awal pada model ini akan ditentukan secara random, hingga didapatkan nilai bobot dan bias dengan *loss function* terendah. Nilai bobot dan nilai bias akhir yang terbentuk pada masing-

masing *hidden layer* dan *output layer* pada model E4 antara lain sebagai berikut.

```

----- layer 1 -----
weights
[[ 0.10826236  0.10937732 -0.09071338  1.0218103
0.8866913 ]
 [ 0.43974203 -0.47219384 -0.15784818  0.97612786
0.64565104]
 [ 0.67781305 -0.257671  -0.7210991  0.44528654 -
0.161407 ]
 [-0.10411871 -0.24618185  0.57174015 -0.4522661 -
0.4717407 ]
 [-0.48496407 -0.02884692 -0.73987144  0.509637 -
0.04267162]]
biases
[-0.6239897  0. 0. -0.53572613 -
0.6545929 ]
----- layer 2 -----
weights
[[-0.31496358  0.8373847  1.057452  0.25498962 -
0.09368283]
 [ 0.27050614 -0.57617784  0.04596364  0.6391249 -
0.46405765]
 [ 0.7023672  0.602489  -0.7036252  0.1081574
0.2916187 ]
 [-0.11875325  0.20663925  0.5881295  -0.7397645 -
0.58252096]
 [-0.7515474  0.85057193  0.9805956  0.76062536 -
0.25756097]]
biases
[ 0. -0.51904774 -0.19199102  0. 0.
]
----- layer 3 -----
weights
[[ 0.52577484 -0.48850313  0.4943838  -0.42420536 -
0.38611954]

```

```

[ 0.826909    0.803316   -0.46613365   0.2967764   -
0.35593906]
[ 0.34284496   0.3511512    0.87477124  -0.2350049   -
0.6881758   ]
[ 0.03005397  -0.3838709    0.73469496  -0.17814356  -
0.674006   ]
[-0.30132195  -0.7129149    0.53589344   0.72691035
0.6590581  ]]
biases
[-0.45154023  -0.46621263   0.08091477   0.          0.
]
----- layer 4 -----
weights
[[ 1.2604860e+00 -9.5876493e-02 -1.1200625e-03
1.8841735e-01
   6.0306114e-01]
 [ 5.9945589e-01  5.3884745e-01 -2.9234639e-01
4.3840566e-01
   5.3341353e-01]
 [ 7.6753807e-01 -5.5500561e-01  1.0086092e-01 -
7.8144687e-01
  -6.7595339e-01]
 [-3.0162796e-01  1.3726598e-01  6.8789005e-01
1.3550025e-01
  -7.4229407e-01]
 [-3.8315991e-01  5.8594644e-01 -7.4867266e-01 -
5.4203677e-01
   7.0995474e-01]]
biases
[-0.23757917  -0.03356277  -0.11319721  -0.02300003  -
0.33204535]
----- layer 5 -----
weights
[[-1.1094354   0.1772347   0.7336301   0.6053772   ]
 [ 0.60677433  -0.19168249   0.52397424   0.14361541]
 [ 0.5657393   0.40851152   0.6952318   0.2149408   ]
 [ 0.26062554   0.19077072   0.7408588   -0.56670845]

```

```
[ 0.11314538 -1.3533137 -1.1774226  0.78900397]]
biases
[-0.5028338  0.71457434  0.07207792 -0.4772986 ]
```

Berikut merupakan hasil perhitungan prediksi dari model E4.

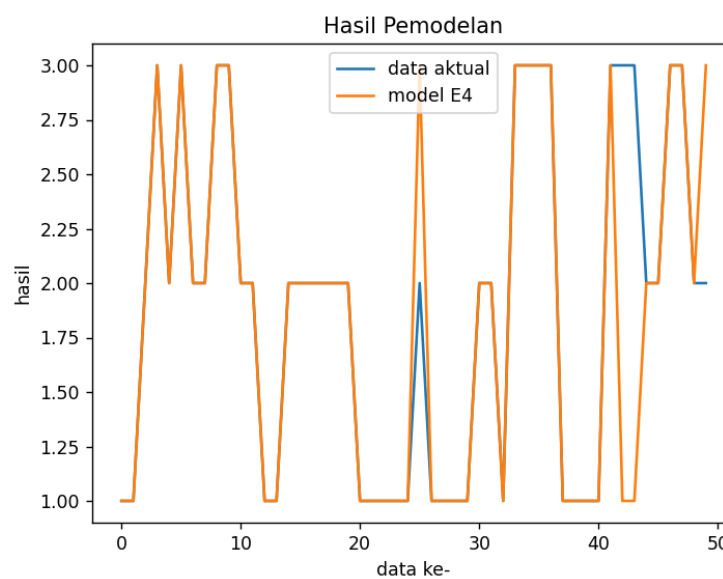
```
1/1 [=====] - 0s 99ms/step
1 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 20ms/step
2 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 20ms/step
3 X=[2 2 3 3 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 22ms/step
4 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
5 X=[2 2 3 3 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 27ms/step
6 X=[3 3 3 3 3], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 21ms/step
7 X=[2 2 2 3 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 32ms/step
8 X=[2 2 2 3 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 32ms/step
9 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
10 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 26ms/step
11 X=[2 2 2 3 3], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 39ms/step
12 X=[2 2 2 3 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 22ms/step
13 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 27ms/step
14 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 42ms/step
15 X=[2 2 2 3 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 23ms/step
```

```
16 X=[2 2 2 3 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 23ms/step
17 X=[2 2 2 3 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 34ms/step
18 X=[2 2 2 3 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 18ms/step
19 X=[2 2 2 3 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 26ms/step
20 X=[2 2 2 3 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 31ms/step
21 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 24ms/step
22 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 35ms/step
23 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 21ms/step
24 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 20ms/step
25 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 23ms/step
26 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 32ms/step
27 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 23ms/step
28 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 28ms/step
29 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 30ms/step
30 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 18ms/step
31 X=[2 2 2 3 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 29ms/step
32 X=[2 2 2 3 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 34ms/step
33 X=[1 1 1 2 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 29ms/step
34 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
```

```
1/1 [=====] - 0s 26ms/step
35 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
36 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 30ms/step
37 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 32ms/step
38 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 37ms/step
39 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 27ms/step
40 X=[1 1 1 2 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 20ms/step
41 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 18ms/step
42 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
43 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
44 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
45 X=[2 2 2 3 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 20ms/step
46 X=[2 2 2 3 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 20ms/step
47 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 23ms/step
48 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
49 X=[2 2 1 2 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 19ms/step
50 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)
```

```
Process finished with exit code 0
```

Gambar 4.60 merupakan grafik perbandingan antara data aktual dengan hasil prediksi perhitungan model E4. Data aktual ditunjukkan oleh garis warna biru dan data hasil prediksi ditunjukkan oleh garis warna oranye. Ada 50 *data testing* yang digunakan, dan *output* yang dihasilkan berupa angka 1 (rusak ringan), 2 (rusak sedang), atau 3 (rusak berat).



Gambar 4. 60 Grafik Hasil Model E4

#### y. Model E5

Model E5 merupakan model *forward propagation in neural network* (FPNN) terakhir yang dibentuk, dimana model ini memiliki *epoch* sebanyak 125 dan *hidden layer* sejumlah 5. Model ini juga akan digunakan untuk mengolah *data testing*, hingga diketahui hasil prediksinya. Hasil perhitungan model E5, didapatkan bahwa akurasi di *epoch* ke-125 adalah sebesar 93% dengan loss sebesar 0,1417.

Epoch 125/125



```
10/10 [=====] - 0s 1ms/step -
loss: 0.1417 - accuracy: 0.9300
```

Pada model E5, nilai bobot dan nilai bias awal akan ditentukan secara random, hingga didapatkan nilai bobot dan bias dengan *loss function* terendah. Nilai bobot dan nilai bias akhir yang terbentuk pada masing-masing *hidden layer* dan *output layer* pada model E5 antara lain sebagai berikut.

```
----- layer 1 -----
weights
[[-0.47540608  0.06513638  0.56316257 -0.8116619  0.59025
 ]
 [-0.64204484  0.5647334   0.36220965  0.2495572
 0.3851902 ]
 [ 0.10506664 -0.09949473 -0.77314115 -0.54019827
 0.5975158 ]
 [ 0.74880654  0.24543072 -0.48781386  0.62333316
 0.52882844]
 [ 0.83276165 -0.26759377  0.5100808   0.68475014
 0.8694118 ]]
biases
[ 0.4744167 -0.41179082 -0.3455665  0.26613122
-0.38706747]
----- layer 2 -----
weights
[[ 0.9845768  0.12145346 -0.56924653  0.83599025
 0.2536213 ]
 [-0.58087087 -0.20016575  0.41249096 -1.1113211
 0.066039 ]
 [-0.30474707  0.01606148  0.32900792 -0.6135081
 -0.17578703]
 [ 0.91695356 -0.57176757 -0.49744117 -0.17052089
 1.012304 ]
 [ 0.8627693 -0.7540058  0.96954376 -0.13663302
```

```

-0.07636704]]
biases
[ 0.13631208  0.          -0.3770318  0.4924409
0.09118003]
----- layer 3 -----
weights
[[ 0.3797087  0.19823055  0.29117152  0.6307198
-0.26250017]
 [-0.00733286  0.5524175  -0.04812253 -0.25969636
0.09069198]
 [-0.63938946 -0.37227556  0.95276016  0.08750331
-0.55619603]
 [ 0.9325407  0.8071845  -0.6648722  -0.71607757
0.0515725 ]
 [ 0.22155052  0.55047333  0.37924886  1.0595098
0.23440933]]
biases
[ 0.3266761  0.45676497 -0.29057312 -0.12982492  0.
]
----- layer 4 -----
weights
[[ 0.5460618  -0.34184885 -0.6953884  -0.8428678
0.7486266 ]
 [ 1.0093979  -0.32903236 -0.6459378  -0.93758845
0.79755425]
 [-0.3802527  -0.14021361 -0.35292676  0.82478243
-0.37699237]
 [ 0.12162377  0.13870275  0.23410773  1.0165317
0.5161022 ]
 [-0.6291292  0.31959903 -0.5518304  -0.4056103
-0.02663076]]
biases
[ 0.30738133  0.          0.          -0.21154138
0.0471732 ]
----- layer 5 -----
weights
[[-0.781571  -0.7577572  0.05163955  1.0376126

```

```

-0.7173915 ]
 [-0.03147197 -0.30027446  0.6837579   0.6259147
0.6824553 ]
 [ 0.21124864 -0.15461355 -0.6923748   0.04951483
0.06361073]
 [ 0.96665734 -0.4882357   0.46375984  0.7648609
-0.53786707]
 [ 0.45042196 -0.59657335 -0.95715964  1.6036899
-0.34787232]]
biases
[-0.29502216  0.           -0.36020914  0.47210908  0.
]
----- layer 6 -----
weights
[[ 0.45100996 -0.98011625  0.32835364  0.22893374]
 [ 0.19088614  0.05618256  0.8070396   0.31757116]
 [ 0.38253814 -0.39074248 -0.41129386  1.0432352 ]
 [-0.7261776   1.0855068   0.74227864  0.4352772 ]
 [-0.14850771  0.56027913  0.71484256 -0.1986832 ]]
biases
[-0.16578773  0.44666404  0.07558151 -0.45950568]

```

Berikut merupakan hasil perhitungan prediksi metode FPNN dari model E5.

```

1/1 [=====] - 0s 101ms/step
1 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 27ms/step
2 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 29ms/step
3 X=[2 2 3 3 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 29ms/step
4 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
5 X=[2 2 3 3 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 21ms/step

```

```

6 X=[3 3 3 3 3], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 24ms/step
7 X=[2 2 2 3 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 23ms/step
8 X=[2 2 2 3 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 27ms/step
9 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 19ms/step
10 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 18ms/step
11 X=[2 2 2 3 3], Predicted=(array([1, 2], dtype=int64),)
1/1 [=====] - 0s 21ms/step
12 X=[2 2 2 3 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 23ms/step
13 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 25ms/step
14 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 21ms/step
15 X=[2 2 2 3 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 30ms/step
16 X=[2 2 2 3 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 26ms/step
17 X=[2 2 2 3 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 26ms/step
18 X=[2 2 2 3 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 21ms/step
19 X=[2 2 2 3 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 23ms/step
20 X=[2 2 2 3 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 27ms/step
21 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 23ms/step
22 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 21ms/step
23 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 26ms/step
24 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)

```

```

1/1 [=====] - 0s 31ms/step
25 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 24ms/step
26 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
27 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 25ms/step
28 X=[1 1 1 1 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 33ms/step
29 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 20ms/step
30 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 20ms/step
31 X=[2 2 2 3 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 30ms/step
32 X=[2 2 2 3 1], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 29ms/step
33 X=[1 1 1 2 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 30ms/step
34 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 29ms/step
35 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
36 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 20ms/step
37 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 33ms/step
38 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 34ms/step
39 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 28ms/step
40 X=[1 1 1 2 2], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 23ms/step
41 X=[1 1 1 2 1], Predicted=(array([1], dtype=int64),)
1/1 [=====] - 0s 18ms/step
42 X=[3 3 3 3 2], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 30ms/step

```

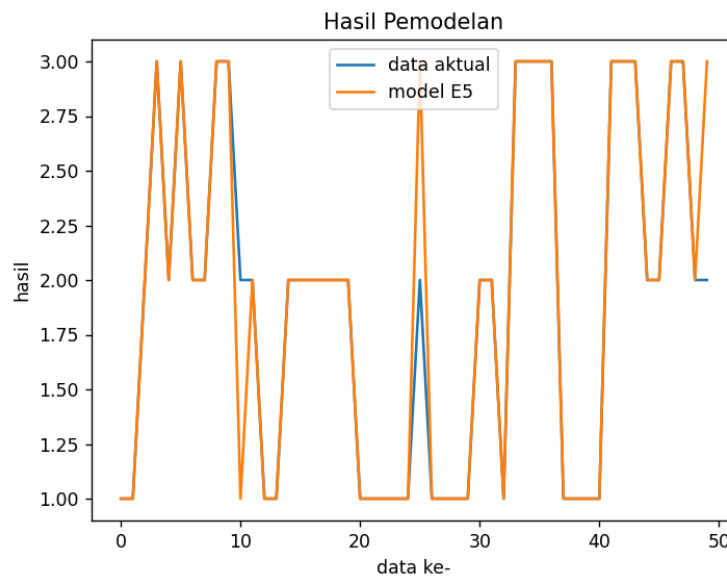
```

43 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 18ms/step
44 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 25ms/step
45 X=[2 2 2 3 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 22ms/step
46 X=[2 2 2 3 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 25ms/step
47 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 30ms/step
48 X=[3 3 3 3 1], Predicted=(array([3], dtype=int64),)
1/1 [=====] - 0s 22ms/step
49 X=[2 2 1 2 2], Predicted=(array([2], dtype=int64),)
1/1 [=====] - 0s 24ms/step
50 X=[2 2 2 2 1], Predicted=(array([3], dtype=int64),)

Process finished with exit code 0

```

Gambar 4.61 merupakan grafik perbandingan antara data aktual dengan hasil prediksi perhitungan model E5. Data aktual ditunjukkan oleh garis warna biru dan data hasil prediksi ditunjukkan oleh garis warna oranye. Ada 50 *data testing* yang digunakan, dan *output* yang dihasilkan berupa angka 1 (rusak ringan), 2 (rusak sedang), atau 3 (rusak berat).



Gambar 4. 61 Grafik Hasil Model E5

#### 4.2.3 Perhitungan MSE, MAPE, dan Akurasi

Berdasarkan hasil perhitungan metode FPNN yang telah dibuat sebelumnya, dapat diketahui besar MSE, MAPE, dan akurasi dari masing-masing model yang telah dibuat. Perhitungan MSE sendiri diawali dengan proses perhitungan nilai *error*, *error square*, dan persentase *error*. Setelah diketahui nilai MSE yang didapatkan, langkah berikutnya adalah menghitung nilai MAPE. Langkah terakhir yakni menghitung nilai akurasi, berdasarkan nilai MAPE yang telah diketahui sebelumnya. Berikut merupakan hasil perhitungan MSE, MAPE, dan akurasi dari masing-masing model.

Tabel 4. 6 Hasil Perhitungan MSE, MAPE, dan Akurasi Model A1

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
1	1	3	-2	2	4	200
2	1	3	-2	2	4	200
3	2	3	-1	1	1	50

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
4	3	3	0	0	0	0
5	2	3	-1	1	1	50
6	3	3	0	0	0	0
7	2	3	-1	1	1	50
8	2	3	-1	1	1	50
9	3	3	0	0	0	0
10	3	3	0	0	0	0
11	2	3	-1	1	1	50
12	2	3	-1	1	1	50
13	1	3	-2	2	4	200
14	1	3	-2	2	4	200
15	2	3	-1	1	1	50
16	2	3	-1	1	1	50
17	2	3	-1	1	1	50
18	2	3	-1	1	1	50
19	2	3	-1	1	1	50
20	2	3	-1	1	1	50
21	1	3	-2	2	4	200
22	1	3	-2	2	4	200
23	1	3	-2	2	4	200
24	1	3	-2	2	4	200
25	1	3	-2	2	4	200
26	2	3	-1	1	1	50
27	1	3	-2	2	4	200
28	1	3	-2	2	4	200
29	1	3	-2	2	4	200
30	1	3	-2	2	4	200
31	2	3	-1	1	1	50



No.	Data Aktual	Data Prediksi	Error	error	error square	%error
32	2	3	-1	1	1	50
33	1	3	-2	2	4	200
34	3	3	0	0	0	0
35	3	3	0	0	0	0
36	3	3	0	0	0	0
37	3	3	0	0	0	0
38	1	3	-2	2	4	200
39	1	3	-2	2	4	200
40	1	3	-2	2	4	200
41	1	3	-2	2	4	200
42	3	3	0	0	0	0
43	3	3	0	0	0	0
44	3	3	0	0	0	0
45	2	3	-1	1	1	50
46	2	3	-1	1	1	50
47	3	3	0	0	0	0
48	3	3	0	0	0	0
49	2	3	-1	1	1	50
50	2	3	-1	1	1	50
<b>Total</b>				<b>55</b>	<b>91</b>	<b>4550</b>
<b>MSE : 1.82</b> <b>MAPE : 91</b> <b>AKURASI : 9</b>						

Tabel 4. 7 Hasil Perhitungan MSE, MAPE, dan Akurasi Model A2

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
1	1	3	-2	2	4	200
2	1	3	-2	2	4	200
3	2	3	-1	1	1	50
4	3	3	0	0	0	0
5	2	3	-1	1	1	50
6	3	3	0	0	0	0
7	2	3	-1	1	1	50
8	2	3	-1	1	1	50
9	3	3	0	0	0	0
10	3	3	0	0	0	0
11	2	1	1	1	1	50
12	2	3	-1	1	1	50
13	1	3	-2	2	4	200
14	1	3	-2	2	4	200
15	2	3	-1	1	1	50
16	2	3	-1	1	1	50
17	2	3	-1	1	1	50
18	2	3	-1	1	1	50
19	2	3	-1	1	1	50
20	2	3	-1	1	1	50
21	1	3	-2	2	4	200
22	1	3	-2	2	4	200
23	1	3	-2	2	4	200
24	1	3	-2	2	4	200
25	1	3	-2	2	4	200
26	2	3	-1	1	1	50
27	1	3	-2	2	4	200

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
28	1	3	-2	2	4	200
29	1	3	-2	2	4	200
30	1	3	-2	2	4	200
31	2	3	-1	1	1	50
32	2	3	-1	1	1	50
33	1	1	0	0	0	0
34	3	3	0	0	0	0
35	3	3	0	0	0	0
36	3	3	0	0	0	0
37	3	3	0	0	0	0
38	1	3	-2	2	4	200
39	1	3	-2	2	4	200
40	1	1	0	0	0	0
41	1	3	-2	2	4	200
42	3	3	0	0	0	0
43	3	3	0	0	0	0
44	3	3	0	0	0	0
45	2	3	-1	1	1	50
46	2	3	-1	1	1	50
47	3	3	0	0	0	0
48	3	3	0	0	0	0
49	2	1	1	1	1	50
50	2	3	-1	1	1	50
<b>Total</b>				<b>51</b>	<b>83</b>	<b>4150</b>
				<b>MSE</b>	<b>: 1.66</b>	
				<b>MAPE</b>	<b>: 83</b>	
				<b>AKURASI</b>	<b>: 17</b>	

Tabel 4. 8 Hasil Perhitungan MSE, MAPE, dan Akurasi Model A3

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
1	1	2	-1	1	1	100
2	1	2	-1	1	1	100
3	2	3	-1	1	1	50
4	3	3	0	0	0	0
5	2	3	-1	1	1	50
6	3	3	0	0	0	0
7	2	3	-1	1	1	50
8	2	3	-1	1	1	50
9	3	3	0	0	0	0
10	3	3	0	0	0	0
11	2	3	-1	1	1	50
12	2	3	-1	1	1	50
13	1	2	-1	1	1	100
14	1	2	-1	1	1	100
15	2	3	-1	1	1	50
16	2	3	-1	1	1	50
17	2	3	-1	1	1	50
18	2	3	-1	1	1	50
19	2	3	-1	1	1	50
20	2	3	-1	1	1	50
21	1	2	-1	1	1	100
22	1	2	-1	1	1	100
23	1	2	-1	1	1	100
24	1	2	-1	1	1	100
25	1	2	-1	1	1	100
26	2	3	-1	1	1	50

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
27	1	2	-1	1	1	100
28	1	2	-1	1	1	100
29	1	2	-1	1	1	100
30	1	2	-1	1	1	100
31	2	3	-1	1	1	50
32	2	3	-1	1	1	50
33	1	2	-1	1	1	100
34	3	3	0	0	0	0
35	3	3	0	0	0	0
36	3	3	0	0	0	0
37	3	3	0	0	0	0
38	1	2	-1	1	1	100
39	1	2	-1	1	1	100
40	1	2	-1	1	1	100
41	1	2	-1	1	1	100
42	3	3	0	0	0	0
43	3	3	0	0	0	0
44	3	3	0	0	0	0
45	2	3	-1	1	1	50
46	2	3	-1	1	1	50
47	3	3	0	0	0	0
48	3	3	0	0	0	0
49	2	3	-1	1	1	50
50	2	3	-1	1	1	50
<b>Total</b>				<b>37</b>	<b>37</b>	<b>2750</b>
				<b>MSE</b>	<b>: 0.74</b>	
				<b>MAPE</b>	<b>: 55</b>	

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
<b>AKURASI : 45</b>						

Tabel 4. 9 Hasil Perhitungan MSE, MAPE, dan Akurasi Model A4

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
1	1	2	-1	1	1	100
2	1	2	-1	1	1	100
3	2	3	-1	1	1	50
4	3	3	0	0	0	0
5	2	3	-1	1	1	50
6	3	3	0	0	0	0
7	2	3	-1	1	1	50
8	2	3	-1	1	1	50
9	3	3	0	0	0	0
10	3	3	0	0	0	0
11	2	2	0	0	0	0
12	2	3	-1	1	1	50
13	1	2	-1	1	1	100
14	1	2	-1	1	1	100
15	2	3	-1	1	1	50
16	2	3	-1	1	1	50
17	2	3	-1	1	1	50
18	2	3	-1	1	1	50
19	2	3	-1	1	1	50
20	2	3	-1	1	1	50
21	1	2	-1	1	1	100
22	1	2	-1	1	1	100
23	1	2	-1	1	1	100

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
24	1	2	-1	1	1	100
25	1	2	-1	1	1	100
26	2	3	-1	1	1	50
27	1	2	-1	1	1	100
28	1	2	-1	1	1	100
29	1	2	-1	1	1	100
30	1	2	-1	1	1	100
31	2	3	-1	1	1	50
32	2	3	-1	1	1	50
33	1	2	-1	1	1	100
34	3	3	0	0	0	0
35	3	3	0	0	0	0
36	3	3	0	0	0	0
37	3	3	0	0	0	0
38	1	2	-1	1	1	100
39	1	2	-1	1	1	100
40	1	2	-1	1	1	100
41	1	2	-1	1	1	100
42	3	3	0	0	0	0
43	3	3	0	0	0	0
44	3	3	0	0	0	0
45	2	3	-1	1	1	50
46	2	3	-1	1	1	50
47	3	3	0	0	0	0
48	3	3	0	0	0	0
49	2	3	-1	1	1	50
50	2	3	-1	1	1	50
<b>Total</b>				<b>36</b>	<b>36</b>	<b>2700</b>

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
<b>MSE : 0.72</b> <b>MAPE : 54</b> <b>AKURASI : 46</b>						

Tabel 4. 10 Hasil Perhitungan MSE, MAPE, dan Akurasi Model A5

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
1	1	1	0	0	0	0
2	1	1	0	0	0	0
3	2	1	1	1	1	50
4	3	3	0	0	0	0
5	2	1	1	1	1	50
6	3	3	0	0	0	0
7	2	1	1	1	1	50
8	2	1	1	1	1	50
9	3	3	0	0	0	0
10	3	3	0	0	0	0
11	2	1	1	1	1	50
12	2	1	1	1	1	50
13	1	1	0	0	0	0
14	1	1	0	0	0	0
15	2	1	1	1	1	50
16	2	1	1	1	1	50
17	2	1	1	1	1	50
18	2	1	1	1	1	50
19	2	1	1	1	1	50
20	2	1	1	1	1	50



No.	Data Aktual	Data Prediksi	Error	error	error square	%error
21	1	1	0	0	0	0
22	1	1	0	0	0	0
23	1	1	0	0	0	0
24	1	1	0	0	0	0
25	1	1	0	0	0	0
26	2	1	1	1	1	50
27	1	1	0	0	0	0
28	1	1	0	0	0	0
29	1	1	0	0	0	0
30	1	1	0	0	0	0
31	2	1	1	1	1	50
32	2	1	1	1	1	50
33	1	1	0	0	0	0
34	3	3	0	0	0	0
35	3	3	0	0	0	0
36	3	3	0	0	0	0
37	3	3	0	0	0	0
38	1	3	-2	2	4	200
39	1	3	-2	2	4	200
40	1	3	-2	2	4	200
41	1	1	0	0	0	0
42	3	3	0	0	0	0
43	3	3	0	0	0	0
44	3	3	0	0	0	0
45	2	1	1	1	1	50
46	2	1	1	1	1	50
47	3	3	0	0	0	0
48	3	3	0	0	0	0

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
49	2	1	1	1	1	50
50	2	1	1	1	1	50
<b>Total</b>				<b>19</b>	<b>19</b>	<b>950</b>
<b>MSE : 0.62</b>						
<b>MAPE : 31</b>						
<b>AKURASI : 69</b>						

Tabel 4. 11 Hasil Perhitungan MSE, MAPE, dan Akurasi Model B1

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
1	1	2	-1	1	1	100
2	1	2	-1	1	1	100
3	2	3	-1	1	1	50
4	3	3	0	0	0	0
5	2	3	-1	1	1	50
6	3	3	0	0	0	0
7	2	3	-1	1	1	50
8	2	3	-1	1	1	50
9	3	3	0	0	0	0
10	3	3	0	0	0	0
11	2	3	-1	1	1	50
12	2	3	-1	1	1	50
13	1	2	-1	1	1	100
14	1	2	-1	1	1	100
15	2	3	-1	1	1	50
16	2	3	-1	1	1	50
17	2	3	-1	1	1	50

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
18	2	3	-1	1	1	50
19	2	3	-1	1	1	50
20	2	3	-1	1	1	50
21	1	2	-1	1	1	100
22	1	3	-2	2	4	200
23	1	3	-2	2	4	200
24	1	2	-1	1	1	100
25	1	2	-1	1	1	100
26	2	3	-1	1	1	50
27	1	3	-2	2	4	200
28	1	3	-2	2	4	200
29	1	2	-1	1	1	100
30	1	2	-1	1	1	100
31	2	3	-1	1	1	50
32	2	3	-1	1	1	50
33	1	2	-1	1	1	100
34	3	3	0	0	0	0
35	3	3	0	0	0	0
36	3	3	0	0	0	0
37	3	3	0	0	0	0
38	1	2	-1	1	1	100
39	1	2	-1	1	1	100
40	1	2	-1	1	1	100
41	1	2	-1	1	1	100
42	3	3	0	0	0	0
43	3	3	0	0	0	0
44	3	3	0	0	0	0
45	2	3	-1	1	1	50

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
46	2	3	-1	1	1	50
47	3	3	0	0	0	0
48	3	3	0	0	0	0
49	2	3	-1	1	1	50
50	2	3	-1	1	1	50
<b>Total</b>				<b>41</b>	<b>49</b>	<b>3150</b>
<b>MSE : 0.98</b> <b>MAPE : 63</b> <b>AKURASI : 37</b>						

Tabel 4. 12 Hasil Perhitungan MSE, MAPE, dan Akurasi Model B2

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
1	1	1	0	0	0	0
2	1	1	0	0	0	0
3	2	3	-1	1	1	50
4	3	3	0	0	0	0
5	2	3	-1	1	1	50
6	3	3	0	0	0	0
7	2	1	1	1	1	50
8	2	3	-1	1	1	50
9	3	3	0	0	0	0
10	3	3	0	0	0	0
11	2	3	-1	1	1	50
12	2	1	1	1	1	50
13	1	1	0	0	0	0
14	1	1	0	0	0	0

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
15	2	3	-1	1	1	50
16	2	3	-1	1	1	50
17	2	3	-1	1	1	50
18	2	1	1	1	1	50
19	2	2	0	0	0	0
20	2	1	1	1	1	50
21	1	1	0	0	0	0
22	1	2	-1	1	1	100
23	1	2	-1	1	1	100
24	1	1	0	0	0	0
25	1	1	0	0	0	0
26	2	3	-1	1	1	50
27	1	2	-1	1	1	100
28	1	2	-1	1	1	100
29	1	1	0	0	0	0
30	1	1	0	0	0	0
31	2	1	1	1	1	50
32	2	1	1	1	1	50
33	1	2	-1	1	1	100
34	3	3	0	0	0	0
35	3	3	0	0	0	0
36	3	3	0	0	0	0
37	3	3	0	0	0	0
38	1	1	0	0	0	0
39	1	1	0	0	0	0
40	1	2	-1	1	1	100
41	1	1	0	0	0	0
42	3	3	0	0	0	0

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
43	3	3	0	0	0	0
44	3	3	0	0	0	0
45	2	3	-1	1	1	50
46	2	3	-1	1	1	50
47	3	3	0	0	0	0
48	3	3	0	0	0	0
49	2	3	-1	1	1	50
50	2	3	-1	1	1	50
<b>Total</b>				<b>24</b>	<b>24</b>	<b>1500</b>
<b>MSE : 0.48</b> <b>MAPE : 30</b> <b>AKURASI : 70</b>						

Tabel 4. 13 Hasil Perhitungan MSE, MAPE, dan Akurasi Model B3

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
1	1	1	0	0	0	0
2	1	1	0	0	0	0
3	2	3	-1	1	1	50
4	3	3	0	0	0	0
5	2	3	-1	1	1	50
6	3	3	0	0	0	0
7	2	2	0	0	0	0
8	2	3	-1	1	1	50
9	3	3	0	0	0	0
10	3	3	0	0	0	0
11	2	3	-1	1	1	50

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
12	2	2	0	0	0	0
13	1	1	0	0	0	0
14	1	1	0	0	0	0
15	2	3	-1	1	1	50
16	2	3	-1	1	1	50
17	2	3	-1	1	1	50
18	2	2	0	0	0	0
19	2	3	-1	1	1	50
20	2	2	0	0	0	0
21	1	1	0	0	0	0
22	1	2	-1	1	1	100
23	1	2	-1	1	1	100
24	1	1	0	0	0	0
25	1	1	0	0	0	0
26	2	3	-1	1	1	50
27	1	2	-1	1	1	100
28	1	2	-1	1	1	100
29	1	1	0	0	0	0
30	1	1	0	0	0	0
31	2	2	0	0	0	0
32	2	2	0	0	0	0
33	1	3	-2	2	4	200
34	3	3	0	0	0	0
35	3	3	0	0	0	0
36	3	3	0	0	0	0
37	3	3	0	0	0	0
38	1	1	0	0	0	0
39	1	1	0	0	0	0

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
40	1	3	-2	2	4	200
41	1	1	0	0	0	0
42	3	3	0	0	0	0
43	3	3	0	0	0	0
44	3	3	0	0	0	0
45	2	3	-1	1	1	50
46	2	3	-1	1	1	50
47	3	3	0	0	0	0
48	3	3	0	0	0	0
49	2	3	-1	1	1	50
50	2	3	-1	1	1	50
<b>Total</b>				<b>21</b>	<b>25</b>	<b>1450</b>
				<b>MSE</b>	<b>: 0.5</b>	
				<b>MAPE</b>	<b>: 29</b>	
				<b>AKURASI</b>	<b>: 71</b>	

Tabel 4. 14 Hasil Perhitungan MSE, MAPE, dan Akurasi Model B4

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
1	1	1	0	0	0	0
2	1	1	0	0	0	0
3	2	3	-1	1	1	50
4	3	3	0	0	0	0
5	2	3	-1	1	1	50
6	3	3	0	0	0	0
7	2	3	-1	1	1	50
8	2	3	-1	1	1	50



No.	Data Aktual	Data Prediksi	Error	error	error square	%error
9	3	3	0	0	0	0
10	3	3	0	0	0	0
11	2	3	-1	1	1	50
12	2	3	-1	1	1	50
13	1	1	0	0	0	0
14	1	1	0	0	0	0
15	2	3	-1	1	1	50
16	2	3	-1	1	1	50
17	2	3	-1	1	1	50
18	2	3	-1	1	1	50
19	2	3	-1	1	1	50
20	2	3	-1	1	1	50
21	1	1	0	0	0	0
22	1	1	0	0	0	0
23	1	1	0	0	0	0
24	1	1	0	0	0	0
25	1	1	0	0	0	0
26	2	3	-1	1	1	50
27	1	1	0	0	0	0
28	1	1	0	0	0	0
29	1	1	0	0	0	0
30	1	1	0	0	0	0
31	2	3	-1	1	1	50
32	2	3	-1	1	1	50
33	1	2	-1	1	1	100
34	3	3	0	0	0	0
35	3	3	0	0	0	0
36	3	3	0	0	0	0

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
37	3	3	0	0	0	0
38	1	1	0	0	0	0
39	1	1	0	0	0	0
40	1	2	-1	1	1	100
41	1	1	0	0	0	0
42	3	3	0	0	0	0
43	3	3	0	0	0	0
44	3	3	0	0	0	0
45	2	3	-1	1	1	50
46	2	3	-1	1	1	50
47	3	3	0	0	0	0
48	3	3	0	0	0	0
49	2	3	-1	1	1	50
50	2	3	-1	1	1	50
<b>Total</b>				<b>21</b>	<b>21</b>	<b>1150</b>
				<b>MSE</b>	<b>: 0.42</b>	
				<b>MAPE</b>	<b>: 23</b>	
				<b>AKURASI</b>	<b>: 77</b>	

Tabel 4. 15 Hasil Perhitungan MSE, MAPE, dan Akurasi Model B5

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
1	1	1	0	0	0	0
2	1	1	0	0	0	0
3	2	3	-1	1	1	50
4	3	3	0	0	0	0
5	2	3	-1	1	1	50

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
6	3	3	0	0	0	0
7	2	3	-1	1	1	50
8	2	3	-1	1	1	50
9	3	3	0	0	0	0
10	3	3	0	0	0	0
11	2	3	-1	1	1	50
12	2	3	-1	1	1	50
13	1	1	0	0	0	0
14	1	1	0	0	0	0
15	2	3	-1	1	1	50
16	2	3	-1	1	1	50
17	2	3	-1	1	1	50
18	2	3	-1	1	1	50
19	2	3	-1	1	1	50
20	2	3	-1	1	1	50
21	1	1	0	0	0	0
22	1	1	0	0	0	0
23	1	1	0	0	0	0
24	1	1	0	0	0	0
25	1	1	0	0	0	0
26	2	3	-1	1	1	50
27	1	1	0	0	0	0
28	1	1	0	0	0	0
29	1	1	0	0	0	0
30	1	1	0	0	0	0
31	2	3	-1	1	1	50
32	2	3	-1	1	1	50
33	1	1	0	0	0	0

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
34	3	3	0	0	0	0
35	3	3	0	0	0	0
36	3	3	0	0	0	0
37	3	3	0	0	0	0
38	1	1	0	0	0	0
39	1	1	0	0	0	0
40	1	1	0	0	0	0
41	1	1	0	0	0	0
42	3	3	0	0	0	0
43	3	3	0	0	0	0
44	3	3	0	0	0	0
45	2	3	-1	1	1	50
46	2	3	-1	1	1	50
47	3	3	0	0	0	0
48	3	3	0	0	0	0
49	2	3	-1	1	1	50
50	2	3	-1	1	1	50
<b>Total</b>				<b>19</b>	<b>19</b>	<b>950</b>
				<b>MSE</b>	<b>: 0.38</b>	
				<b>MAPE</b>	<b>: 19</b>	
				<b>AKURASI</b>	<b>: 81</b>	

Tabel 4. 16 Hasil Perhitungan MSE, MAPE, dan Akurasi Model C1

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
1	1	1	0	0	0	0
2	1	1	0	0	0	0

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
3	2	3	-1	1	1	50
4	3	3	0	0	0	0
5	2	3	-1	1	1	50
6	3	3	0	0	0	0
7	2	3	-1	1	1	50
8	2	3	-1	1	1	50
9	3	3	0	0	0	0
10	3	3	0	0	0	0
11	2	3	-1	1	1	50
12	2	3	-1	1	1	50
13	1	1	0	0	0	0
14	1	1	0	0	0	0
15	2	3	-1	1	1	50
16	2	3	-1	1	1	50
17	2	3	-1	1	1	50
18	2	3	-1	1	1	50
19	2	3	-1	1	1	50
20	2	3	-1	1	1	50
21	1	1	0	0	0	0
22	1	3	-2	2	4	200
23	1	3	-2	2	4	200
24	1	1	0	0	0	0
25	1	1	0	0	0	0
26	2	3	-1	1	1	50
27	1	3	-2	2	4	200
28	1	3	-2	2	4	200
29	1	1	0	0	0	0
30	1	1	0	0	0	0

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
31	2	3	-1	1	1	50
32	2	3	-1	1	1	50
33	1	3	-2	2	4	200
34	3	3	0	0	0	0
35	3	3	0	0	0	0
36	3	3	0	0	0	0
37	3	3	0	0	0	0
38	1	1	0	0	0	0
39	1	1	0	0	0	0
40	1	3	-2	2	4	200
41	1	1	0	0	0	0
42	3	3	0	0	0	0
43	3	3	0	0	0	0
44	3	3	0	0	0	0
45	2	3	-1	1	1	50
46	2	3	-1	1	1	50
47	3	3	0	0	0	0
48	3	3	0	0	0	0
49	2	3	-1	1	1	50
50	2	3	-1	1	1	50
<b>Total</b>				<b>31</b>	<b>43</b>	<b>2150</b>
				<b>MSE</b>	<b>: 0.86</b>	
				<b>MAPE</b>	<b>: 43</b>	
				<b>AKURASI</b>	<b>: 57</b>	

Tabel 4. 17 Hasil Perhitungan MSE, MAPE, dan Akurasi Model C2

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
1	1	1	0	0	0	0
2	1	1	0	0	0	0
3	2	3	-1	1	1	50
4	3	3	0	0	0	0
5	2	3	-1	1	1	50
6	3	3	0	0	0	0
7	2	3	-1	1	1	50
8	2	3	-1	1	1	50
9	3	3	0	0	0	0
10	3	3	0	0	0	0
11	2	1	1	1	1	50
12	2	3	-1	1	1	50
13	1	1	0	0	0	0
14	1	1	0	0	0	0
15	2	3	-1	1	1	50
16	2	3	-1	1	1	50
17	2	3	-1	1	1	50
18	2	3	-1	1	1	50
19	2	3	-1	1	1	50
20	2	3	-1	1	1	50
21	1	1	0	0	0	0
22	1	2	-1	1	1	100
23	1	2	-1	1	1	100
24	1	1	0	0	0	0
25	1	1	0	0	0	0
26	2	3	-1	1	1	50
27	1	2	-1	1	1	100

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
28	1	2	-1	1	1	100
29	1	1	0	0	0	0
30	1	1	0	0	0	0
31	2	3	-1	1	1	50
32	2	3	-1	1	1	50
33	1	1	0	0	0	0
34	3	3	0	0	0	0
35	3	3	0	0	0	0
36	3	3	0	0	0	0
37	3	3	0	0	0	0
38	1	1	0	0	0	0
39	1	1	0	0	0	0
40	1	1	0	0	0	0
41	1	1	0	0	0	0
42	3	3	0	0	0	0
43	3	3	0	0	0	0
44	3	3	0	0	0	0
45	2	3	-1	1	1	50
46	2	3	-1	1	1	50
47	3	3	0	0	0	0
48	3	3	0	0	0	0
49	2	3	-1	1	1	50
50	2	3	-1	1	1	50
<b>Total</b>				<b>23</b>	<b>23</b>	<b>1350</b>
<b>MSE : 0.46</b> <b>MAPE : 27</b> <b>AKURASI : 73</b>						



Tabel 4. 18 Hasil Perhitungan MSE, MAPE, dan Akurasi Model C3

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
1	1	1	0	0	0	0
2	1	1	0	0	0	0
3	2	3	-1	1	1	50
4	3	3	0	0	0	0
5	2	3	-1	1	1	50
6	3	3	0	0	0	0
7	2	3	-1	1	1	50
8	2	3	-1	1	1	50
9	3	3	0	0	0	0
10	3	3	0	0	0	0
11	2	2	0	0	0	0
12	2	3	-1	1	1	50
13	1	1	0	0	0	0
14	1	1	0	0	0	0
15	2	3	-1	1	1	50
16	2	3	-1	1	1	50
17	2	3	-1	1	1	50
18	2	3	-1	1	1	50
19	2	3	-1	1	1	50
20	2	3	-1	1	1	50
21	1	1	0	0	0	0
22	1	2	-1	1	1	100
23	1	2	-1	1	1	100
24	1	1	0	0	0	0
25	1	1	0	0	0	0
26	2	3	-1	1	1	50

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
27	1	2	-1	1	1	100
28	1	2	-1	1	1	100
29	1	1	0	0	0	0
30	1	1	0	0	0	0
31	2	3	-1	1	1	50
32	2	3	-1	1	1	50
33	1	1	0	0	0	0
34	3	3	0	0	0	0
35	3	3	0	0	0	0
36	3	3	0	0	0	0
37	3	3	0	0	0	0
38	1	1	0	0	0	0
39	1	1	0	0	0	0
40	1	1	0	0	0	0
41	1	1	0	0	0	0
42	3	3	0	0	0	0
43	3	3	0	0	0	0
44	3	3	0	0	0	0
45	2	3	-1	1	1	50
46	2	3	-1	1	1	50
47	3	3	0	0	0	0
48	3	3	0	0	0	0
49	2	3	-1	1	1	50
50	2	3	-1	1	1	50
<b>Total</b>				<b>22</b>	<b>22</b>	<b>1300</b>
				<b>MSE</b>	<b>: 0.44</b>	
				<b>MAPE</b>	<b>: 26</b>	

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
<b>AKURASI : 74</b>						

Tabel 4. 19 Hasil Perhitungan MSE, MAPE, dan Akurasi Model C4

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
1	1	1	0	0	0	0
2	1	1	0	0	0	0
3	2	1	1	1	1	50
4	3	3	0	0	0	0
5	2	1	1	1	1	50
6	3	3	0	0	0	0
7	2	2	0	0	0	0
8	2	1	1	1	1	50
9	3	3	0	0	0	0
10	3	3	0	0	0	0
11	2	1	1	1	1	50
12	2	2	0	0	0	0
13	1	1	0	0	0	0
14	1	1	0	0	0	0
15	2	1	1	1	1	50
16	2	1	1	1	1	50
17	2	1	1	1	1	50
18	2	2	0	0	0	0
19	2	1	1	1	1	50
20	2	2	0	0	0	0
21	1	1	0	0	0	0
22	1	2	-1	1	1	100
23	1	2	-1	1	1	100

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
24	1	1	0	0	0	0
25	1	1	0	0	0	0
26	2	3	-1	1	1	50
27	1	2	-1	1	1	100
28	1	2	-1	1	1	100
29	1	1	0	0	0	0
30	1	1	0	0	0	0
31	2	2	0	0	0	0
32	2	2	0	0	0	0
33	1	1	0	0	0	0
34	3	3	0	0	0	0
35	3	3	0	0	0	0
36	3	3	0	0	0	0
37	3	3	0	0	0	0
38	1	1	0	0	0	0
39	1	1	0	0	0	0
40	1	1	0	0	0	0
41	1	1	0	0	0	0
42	3	3	0	0	0	0
43	3	3	0	0	0	0
44	3	3	0	0	0	0
45	2	1	1	1	1	50
46	2	1	1	1	1	50
47	3	3	0	0	0	0
48	3	3	0	0	0	0
49	2	1	1	1	1	50
50	2	3	-1	1	1	50
<b>Total</b>				<b>17</b>	<b>17</b>	<b>1050</b>

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
				<b>MSE</b>	<b>: 0.34</b>	
				<b>MAPE</b>	<b>: 21</b>	
				<b>AKURASI</b>	<b>: 79</b>	

Tabel 4. 20 Hasil Perhitungan MSE, MAPE, dan Akurasi Model C5

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
1	1	1	0	0	0	0
2	1	1	0	0	0	0
3	2	2	0	0	0	0
4	3	3	0	0	0	0
5	2	2	0	0	0	0
6	3	3	0	0	0	0
7	2	1	1	1	1	50
8	2	2	0	0	0	0
9	3	3	0	0	0	0
10	3	3	0	0	0	0
11	2	2	0	0	0	0
12	2	1	1	1	1	50
13	1	1	0	0	0	0
14	1	1	0	0	0	0
15	2	2	0	0	0	0
16	2	2	0	0	0	0
17	2	2	0	0	0	0
18	2	1	1	1	1	50
19	2	2	0	0	0	0
20	2	1	1	1	1	50

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
21	1	1	0	0	0	0
22	1	2	-1	1	1	100
23	1	2	-1	1	1	100
24	1	1	0	0	0	0
25	1	1	0	0	0	0
26	2	3	-1	1	1	50
27	1	2	-1	1	1	100
28	1	2	-1	1	1	100
29	1	1	0	0	0	0
30	1	1	0	0	0	0
31	2	1	1	1	1	50
32	2	1	1	1	1	50
33	1	1	0	0	0	0
34	3	3	0	0	0	0
35	3	3	0	0	0	0
36	3	3	0	0	0	0
37	3	3	0	0	0	0
38	1	1	0	0	0	0
39	1	1	0	0	0	0
40	1	1	0	0	0	0
41	1	1	0	0	0	0
42	3	3	0	0	0	0
43	3	3	0	0	0	0
44	3	3	0	0	0	0
45	2	2	0	0	0	0
46	2	2	0	0	0	0
47	3	3	0	0	0	0
48	3	3	0	0	0	0

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
49	2	3	-1	1	1	50
50	2	3	-1	1	1	50
<b>Total</b>				<b>13</b>	<b>13</b>	<b>850</b>
<b>MSE</b>						<b>: 0.26</b>
<b>MAPE</b>						<b>: 17</b>
<b>AKURASI</b>						<b>: 83</b>

Tabel 4. 21 Hasil Perhitungan MSE, MAPE, dan Akurasi Model D1

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
1	1	1	0	0	0	0
2	1	1	0	0	0	0
3	2	3	-1	1	1	50
4	3	3	0	0	0	0
5	2	3	-1	1	1	50
6	3	3	0	0	0	0
7	2	3	-1	1	1	50
8	2	3	-1	1	1	50
9	3	3	0	0	0	0
10	3	3	0	0	0	0
11	2	3	-1	1	1	50
12	2	3	-1	1	1	50
13	1	1	0	0	0	0
14	1	1	0	0	0	0
15	2	3	-1	1	1	50
16	2	3	-1	1	1	50
17	2	3	-1	1	1	50

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
18	2	3	-1	1	1	50
19	2	3	-1	1	1	50
20	2	3	-1	1	1	50
21	1	1	0	0	0	0
22	1	3	-2	2	4	200
23	1	3	-2	2	4	200
24	1	1	0	0	0	0
25	1	1	0	0	0	0
26	2	3	-1	1	1	50
27	1	3	-2	2	4	200
28	1	3	-2	2	4	200
29	1	1	0	0	0	0
30	1	1	0	0	0	0
31	2	3	-1	1	1	50
32	2	3	-1	1	1	50
33	1	1	0	0	0	0
34	3	3	0	0	0	0
35	3	3	0	0	0	0
36	3	3	0	0	0	0
37	3	3	0	0	0	0
38	1	1	0	0	0	0
39	1	1	0	0	0	0
40	1	1	0	0	0	0
41	1	1	0	0	0	0
42	3	3	0	0	0	0
43	3	3	0	0	0	0
44	3	3	0	0	0	0
45	2	3	-1	1	1	50



No.	Data Aktual	Data Prediksi	Error	error	error square	%error
46	2	3	-1	1	1	50
47	3	3	0	0	0	0
48	3	3	0	0	0	0
49	2	3	-1	1	1	50
50	2	3	-1	1	1	50
<b>Total</b>				<b>27</b>	<b>35</b>	<b>1750</b>
				<b>MSE</b>	<b>: 0.7</b>	
				<b>MAPE</b>	<b>: 35</b>	
				<b>AKURASI</b>	<b>: 65</b>	

Tabel 4. 22 Hasil Perhitungan MSE, MAPE, dan Akurasi Model D2

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
1	1	1	0	0	0	0
2	1	1	0	0	0	0
3	2	3	-1	1	1	50
4	3	3	0	0	0	0
5	2	3	-1	1	1	50
6	3	3	0	0	0	0
7	2	3	-1	1	1	50
8	2	3	-1	1	1	50
9	3	3	0	0	0	0
10	3	3	0	0	0	0
11	2	3	-1	1	1	50
12	2	3	-1	1	1	50
13	1	1	0	0	0	0
14	1	1	0	0	0	0

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
15	2	3	-1	1	1	50
16	2	3	-1	1	1	50
17	2	3	-1	1	1	50
18	2	3	-1	1	1	50
19	2	3	-1	1	1	50
20	2	3	-1	1	1	50
21	1	1	0	0	0	0
22	1	1	0	0	0	0
23	1	2	-1	1	1	100
24	1	2	-1	1	1	100
25	1	2	-1	1	1	100
26	2	3	-1	1	1	50
27	1	1	0	0	0	0
28	1	1	0	0	0	0
29	1	1	0	0	0	0
30	1	1	0	0	0	0
31	2	3	-1	1	1	50
32	2	3	-1	1	1	50
33	1	1	0	0	0	0
34	3	3	0	0	0	0
35	3	3	0	0	0	0
36	3	3	0	0	0	0
37	3	3	0	0	0	0
38	1	1	0	0	0	0
39	1	1	0	0	0	0
40	1	1	0	0	0	0
41	1	1	0	0	0	0
42	3	3	0	0	0	0

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
43	3	3	0	0	0	0
44	3	3	0	0	0	0
45	2	3	-1	1	1	50
46	2	3	-1	1	1	50
47	3	3	0	0	0	0
48	3	3	0	0	0	0
49	2	3	-1	1	1	50
50	2	3	-1	1	1	50
<b>Total</b>				<b>22</b>	<b>22</b>	<b>1250</b>
				<b>MSE</b>	<b>: 0.44</b>	
				<b>MAPE</b>	<b>: 25</b>	
				<b>AKURASI</b>	<b>: 75</b>	

Tabel 4. 23 Hasil Perhitungan MSE, MAPE, dan Akurasi Model D3

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
1	1	1	0	0	0	0
2	1	1	0	0	0	0
3	2	3	-1	1	1	50
4	3	3	0	0	0	0
5	2	3	-1	1	1	50
6	3	3	0	0	0	0
7	2	3	-1	1	1	50
8	2	3	-1	1	1	50
9	3	3	0	0	0	0
10	3	3	0	0	0	0
11	2	3	-1	1	1	50

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
12	2	3	-1	1	1	50
13	1	1	0	0	0	0
14	1	1	0	0	0	0
15	2	3	-1	1	1	50
16	2	3	-1	1	1	50
17	2	3	-1	1	1	50
18	2	3	-1	1	1	50
19	2	3	-1	1	1	50
20	2	3	-1	1	1	50
21	1	1	0	0	0	0
22	1	1	0	0	0	0
23	1	1	0	0	0	0
24	1	1	0	0	0	0
25	1	2	-1	1	1	100
26	2	3	-1	1	1	50
27	1	1	0	0	0	0
28	1	1	0	0	0	0
29	1	1	0	0	0	0
30	1	1	0	0	0	0
31	2	3	-1	1	1	50
32	2	3	-1	1	1	50
33	1	1	0	0	0	0
34	3	3	0	0	0	0
35	3	3	0	0	0	0
36	3	3	0	0	0	0
37	3	3	0	0	0	0
38	1	1	0	0	0	0
39	1	1	0	0	0	0

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
40	1	1	0	0	0	0
41	1	1	0	0	0	0
42	3	3	0	0	0	0
43	3	3	0	0	0	0
44	3	3	0	0	0	0
45	2	3	-1	1	1	50
46	2	3	-1	1	1	50
47	3	3	0	0	0	0
48	3	3	0	0	0	0
49	2	3	-1	1	1	50
50	2	3	-1	1	1	50
<b>Total</b>				<b>20</b>	<b>20</b>	<b>1050</b>
				<b>MSE</b>	<b>: 0.4</b>	
				<b>MAPE</b>	<b>: 21</b>	
				<b>AKURASI</b>	<b>: 79</b>	

Tabel 4. 24 Hasil Perhitungan MSE, MAPE, dan Akurasi Model D4

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
1	1	1	0	0	0	0
2	1	1	0	0	0	0
3	2	3	-1	1	1	50
4	3	3	0	0	0	0
5	2	3	-1	1	1	50
6	3	3	0	0	0	0
7	2	3	-1	1	1	50
8	2	3	-1	1	1	50

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
9	3	3	0	0	0	0
10	3	3	0	0	0	0
11	2	3	-1	1	1	50
12	2	3	-1	1	1	50
13	1	1	0	0	0	0
14	1	1	0	0	0	0
15	2	3	-1	1	1	50
16	2	3	-1	1	1	50
17	2	3	-1	1	1	50
18	2	3	-1	1	1	50
19	2	3	-1	1	1	50
20	2	3	-1	1	1	50
21	1	1	0	0	0	0
22	1	1	0	0	0	0
23	1	1	0	0	0	0
24	1	1	0	0	0	0
25	1	1	0	0	0	0
26	2	3	-1	1	1	50
27	1	1	0	0	0	0
28	1	1	0	0	0	0
29	1	1	0	0	0	0
30	1	1	0	0	0	0
31	2	3	-1	1	1	50
32	2	3	-1	1	1	50
33	1	1	0	0	0	0
34	3	3	0	0	0	0
35	3	3	0	0	0	0
36	3	3	0	0	0	0

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
37	3	3	0	0	0	0
38	1	1	0	0	0	0
39	1	1	0	0	0	0
40	1	1	0	0	0	0
41	1	1	0	0	0	0
42	3	3	0	0	0	0
43	3	3	0	0	0	0
44	3	3	0	0	0	0
45	2	3	-1	1	1	50
46	2	3	-1	1	1	50
47	3	3	0	0	0	0
48	3	3	0	0	0	0
49	2	3	-1	1	1	50
50	2	3	-1	1	1	50
<b>Total</b>				<b>19</b>	<b>19</b>	<b>950</b>
				<b>MSE</b>	<b>: 0.38</b>	
				<b>MAPE</b>	<b>: 19</b>	
				<b>AKURASI</b>	<b>: 81</b>	

Tabel 4. 25 Hasil Perhitungan MSE, MAPE, dan Akurasi Model D5

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
1	1	1	0	0	0	0
2	1	1	0	0	0	0
3	2	3	-1	1	1	50
4	3	3	0	0	0	0
5	2	3	-1	1	1	50

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
6	3	3	0	0	0	0
7	2	3	-1	1	1	50
8	2	3	-1	1	1	50
9	3	3	0	0	0	0
10	3	3	0	0	0	0
11	2	3	-1	1	1	50
12	2	3	-1	1	1	50
13	1	1	0	0	0	0
14	1	1	0	0	0	0
15	2	3	-1	1	1	50
16	2	3	-1	1	1	50
17	2	3	-1	1	1	50
18	2	3	-1	1	1	50
19	2	3	-1	1	1	50
20	2	3	-1	1	1	50
21	1	1	0	0	0	0
22	1	1	0	0	0	0
23	1	1	0	0	0	0
24	1	1	0	0	0	0
25	1	1	0	0	0	0
26	2	3	-1	1	1	50
27	1	1	0	0	0	0
28	1	1	0	0	0	0
29	1	1	0	0	0	0
30	1	1	0	0	0	0
31	2	2	0	0	0	0
32	2	2	0	0	0	0
33	1	1	0	0	0	0



No.	Data Aktual	Data Prediksi	Error	error	error square	%error
34	3	3	0	0	0	0
35	3	3	0	0	0	0
36	3	3	0	0	0	0
37	3	3	0	0	0	0
38	1	1	0	0	0	0
39	1	1	0	0	0	0
40	1	1	0	0	0	0
41	1	1	0	0	0	0
42	3	3	0	0	0	0
43	3	3	0	0	0	0
44	3	3	0	0	0	0
45	2	3	-1	1	1	50
46	2	3	-1	1	1	50
47	3	3	0	0	0	0
48	3	3	0	0	0	0
49	2	2	0	0	0	0
50	2	3	-1	1	1	50
<b>Total</b>				<b>18</b>	<b>18</b>	<b>900</b>
				<b>MSE</b>	<b>: 0.32</b>	
				<b>MAPE</b>	<b>: 16</b>	
				<b>AKURASI</b>	<b>: 84</b>	

Tabel 4. 26 Hasil Perhitungan MSE, MAPE, dan Akurasi Model E1

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
1	1	1	0	0	0	0
2	1	1	0	0	0	0

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
3	2	3	-1	1	1	50
4	3	3	0	0	0	0
5	2	3	-1	1	1	50
6	3	3	0	0	0	0
7	2	3	-1	1	1	50
8	2	3	-1	1	1	50
9	3	3	0	0	0	0
10	3	3	0	0	0	0
11	2	3	-1	1	1	50
12	2	3	-1	1	1	50
13	1	1	0	0	0	0
14	1	1	0	0	0	0
15	2	3	-1	1	1	50
16	2	3	-1	1	1	50
17	2	3	-1	1	1	50
18	2	3	-1	1	1	50
19	2	3	-1	1	1	50
20	2	3	-1	1	1	50
21	1	1	0	0	0	0
22	1	1	0	0	0	0
23	1	1	0	0	0	0
24	1	1	0	0	0	0
25	1	1	0	0	0	0
26	2	3	-1	1	1	50
27	1	1	0	0	0	0
28	1	1	0	0	0	0
29	1	1	0	0	0	0
30	1	1	0	0	0	0

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
31	2	3	-1	1	1	50
32	2	3	-1	1	1	50
33	1	1	0	0	0	0
34	3	3	0	0	0	0
35	3	3	0	0	0	0
36	3	3	0	0	0	0
37	3	3	0	0	0	0
38	1	1	0	0	0	0
39	1	1	0	0	0	0
40	1	1	0	0	0	0
41	1	1	0	0	0	0
42	3	3	0	0	0	0
43	3	3	0	0	0	0
44	3	3	0	0	0	0
45	2	3	-1	1	1	50
46	2	3	-1	1	1	50
47	3	3	0	0	0	0
48	3	3	0	0	0	0
49	2	3	-1	1	1	50
50	2	3	-1	1	1	50
<b>Total</b>				<b>19</b>	<b>19</b>	<b>950</b>
				<b>MSE</b>	<b>: 0.38</b>	
				<b>MAPE</b>	<b>: 19</b>	
				<b>AKURASI</b>	<b>: 81</b>	

Tabel 4. 27 Hasil Perhitungan MSE, MAPE, dan Akurasi Model E2

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
1	1	1	0	0	0	0
2	1	1	0	0	0	0
3	2	3	-1	1	1	50
4	3	3	0	0	0	0
5	2	3	-1	1	1	50
6	3	3	0	0	0	0
7	2	3	-1	1	1	50
8	2	3	-1	1	1	50
9	3	3	0	0	0	0
10	3	3	0	0	0	0
11	2	2	0	0	0	0
12	2	3	-1	1	1	50
13	1	1	0	0	0	0
14	1	1	0	0	0	0
15	2	3	-1	1	1	50
16	2	3	-1	1	1	50
17	2	3	-1	1	1	50
18	2	3	-1	1	1	50
19	2	3	-1	1	1	50
20	2	3	-1	1	1	50
21	1	1	0	0	0	0
22	1	1	0	0	0	0
23	1	1	0	0	0	0
24	1	1	0	0	0	0
25	1	1	0	0	0	0
26	2	3	-1	1	1	50
27	1	1	0	0	0	0

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
28	1	1	0	0	0	0
29	1	1	0	0	0	0
30	1	1	0	0	0	0
31	2	3	-1	1	1	50
32	2	3	-1	1	1	50
33	1	1	0	0	0	0
34	3	3	0	0	0	0
35	3	3	0	0	0	0
36	3	3	0	0	0	0
37	3	3	0	0	0	0
38	1	1	0	0	0	0
39	1	1	0	0	0	0
40	1	1	0	0	0	0
41	1	1	0	0	0	0
42	3	3	0	0	0	0
43	3	3	0	0	0	0
44	3	3	0	0	0	0
45	2	3	-1	1	1	50
46	2	3	-1	1	1	50
47	3	3	0	0	0	0
48	3	3	0	0	0	0
49	2	2	0	0	0	0
50	2	3	-1	1	1	50
<b>Total</b>				<b>17</b>	<b>17</b>	<b>850</b>
				<b>MSE</b>	<b>: 0.34</b>	
				<b>MAPE</b>	<b>: 17</b>	
				<b>AKURASI</b>	<b>: 83</b>	

Tabel 4. 28 Hasil Perhitungan MSE, MAPE, dan Akurasi Model E3

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
1	1	1	0	0	0	0
2	1	1	0	0	0	0
3	2	3	-1	1	1	50
4	3	3	0	0	0	0
5	2	3	-1	1	1	50
6	3	3	0	0	0	0
7	2	3	-1	1	1	50
8	2	2	0	0	0	0
9	3	3	0	0	0	0
10	3	3	0	0	0	0
11	2	2	0	0	0	0
12	2	3	-1	1	1	50
13	1	1	0	0	0	0
14	1	1	0	0	0	0
15	2	2	0	0	0	0
16	2	2	0	0	0	0
17	2	2	0	0	0	0
18	2	3	-1	1	1	50
19	2	2	0	0	0	0
20	2	3	-1	1	1	50
21	1	1	0	0	0	0
22	1	1	0	0	0	0
23	1	1	0	0	0	0
24	1	1	0	0	0	0
25	1	1	0	0	0	0
26	2	3	-1	1	1	50

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
27	1	1	0	0	0	0
28	1	1	0	0	0	0
29	1	1	0	0	0	0
30	1	1	0	0	0	0
31	2	3	-1	1	1	50
32	2	3	-1	1	1	50
33	1	1	0	0	0	0
34	3	3	0	0	0	0
35	3	3	0	0	0	0
36	3	3	0	0	0	0
37	3	3	0	0	0	0
38	1	1	0	0	0	0
39	1	1	0	0	0	0
40	1	1	0	0	0	0
41	1	1	0	0	0	0
42	3	3	0	0	0	0
43	3	3	0	0	0	0
44	3	3	0	0	0	0
45	2	2	0	0	0	0
46	2	2	0	0	0	0
47	3	3	0	0	0	0
48	3	3	0	0	0	0
49	2	2	0	0	0	0
50	2	3	-1	1	1	50
<b>Total</b>				<b>10</b>	<b>10</b>	<b>500</b>
				<b>MSE</b>	<b>: 0.2</b>	
				<b>MAPE</b>	<b>: 10</b>	

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
<b>AKURASI : 90</b>						

Tabel 4. 29 Hasil Perhitungan MSE, MAPE, dan Akurasi Model E4

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
1	1	1	0	0	0	0
2	1	1	0	0	0	0
3	2	2	0	0	0	0
4	3	3	0	0	0	0
5	2	2	0	0	0	0
6	3	3	0	0	0	0
7	2	2	0	0	0	0
8	2	2	0	0	0	0
9	3	3	0	0	0	0
10	3	3	0	0	0	0
11	2	2	0	0	0	0
12	2	2	0	0	0	0
13	1	1	0	0	0	0
14	1	1	0	0	0	0
15	2	2	0	0	0	0
16	2	2	0	0	0	0
17	2	2	0	0	0	0
18	2	2	0	0	0	0
19	2	2	0	0	0	0
20	2	2	0	0	0	0
21	1	1	0	0	0	0
22	1	1	0	0	0	0
23	1	1	0	0	0	0



No.	Data Aktual	Data Prediksi	Error	error	error square	%error
24	1	1	0	0	0	0
25	1	1	0	0	0	0
26	2	3	-1	1	1	50
27	1	1	0	0	0	0
28	1	1	0	0	0	0
29	1	1	0	0	0	0
30	1	1	0	0	0	0
31	2	2	0	0	0	0
32	2	2	0	0	0	0
33	1	1	0	0	0	0
34	3	3	0	0	0	0
35	3	3	0	0	0	0
36	3	3	0	0	0	0
37	3	3	0	0	0	0
38	1	1	0	0	0	0
39	1	1	0	0	0	0
40	1	1	0	0	0	0
41	1	1	0	0	0	0
42	3	3	0	0	0	0
43	3	1	2	2	4	66.66667
44	3	1	2	2	4	66.66667
45	2	2	0	0	0	0
46	2	2	0	0	0	0
47	3	3	0	0	0	0
48	3	3	0	0	0	0
49	2	2	0	0	0	0
50	2	3	-1	1	1	50
<b>Total</b>				<b>6</b>	<b>10</b>	<b>233.3333</b>

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
<b>MSE : 0.2</b> <b>MAPE : 4.666667</b> <b>AKURASI : 95.33333</b>						

Tabel 4. 30 Hasil Perhitungan MSE, MAPE, dan Akurasi Model E5

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
1	1	1	0	0	0	0
2	1	1	0	0	0	0
3	2	2	0	0	0	0
4	3	3	0	0	0	0
5	2	2	0	0	0	0
6	3	3	0	0	0	0
7	2	2	0	0	0	0
8	2	2	0	0	0	0
9	3	3	0	0	0	0
10	3	3	0	0	0	0
11	2	1	1	1	1	50
12	2	2	0	0	0	0
13	1	1	0	0	0	0
14	1	1	0	0	0	0
15	2	2	0	0	0	0
16	2	2	0	0	0	0
17	2	2	0	0	0	0
18	2	2	0	0	0	0
19	2	2	0	0	0	0
20	2	2	0	0	0	0

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
21	1	1	0	0	0	0
22	1	1	0	0	0	0
23	1	1	0	0	0	0
24	1	1	0	0	0	0
25	1	1	0	0	0	0
26	2	3	-1	1	1	50
27	1	1	0	0	0	0
28	1	1	0	0	0	0
29	1	1	0	0	0	0
30	1	1	0	0	0	0
31	2	2	0	0	0	0
32	2	2	0	0	0	0
33	1	1	0	0	0	0
34	3	3	0	0	0	0
35	3	3	0	0	0	0
36	3	3	0	0	0	0
37	3	3	0	0	0	0
38	1	1	0	0	0	0
39	1	1	0	0	0	0
40	1	1	0	0	0	0
41	1	1	0	0	0	0
42	3	3	0	0	0	0
43	3	3	0	0	0	0
44	3	3	0	0	0	0
45	2	2	0	0	0	0
46	2	2	0	0	0	0
47	3	3	0	0	0	0
48	3	3	0	0	0	0

No.	Data Aktual	Data Prediksi	Error	error	error square	%error
49	2	2	0	0	0	0
50	2	3	-1	1	1	50
<b>Total</b>				<b>3</b>	<b>3</b>	<b>150</b>
				<b>MSE</b>	<b>: 0.06</b>	
				<b>MAPE</b>	<b>: 3</b>	
				<b>AKURASI</b>	<b>: 97</b>	

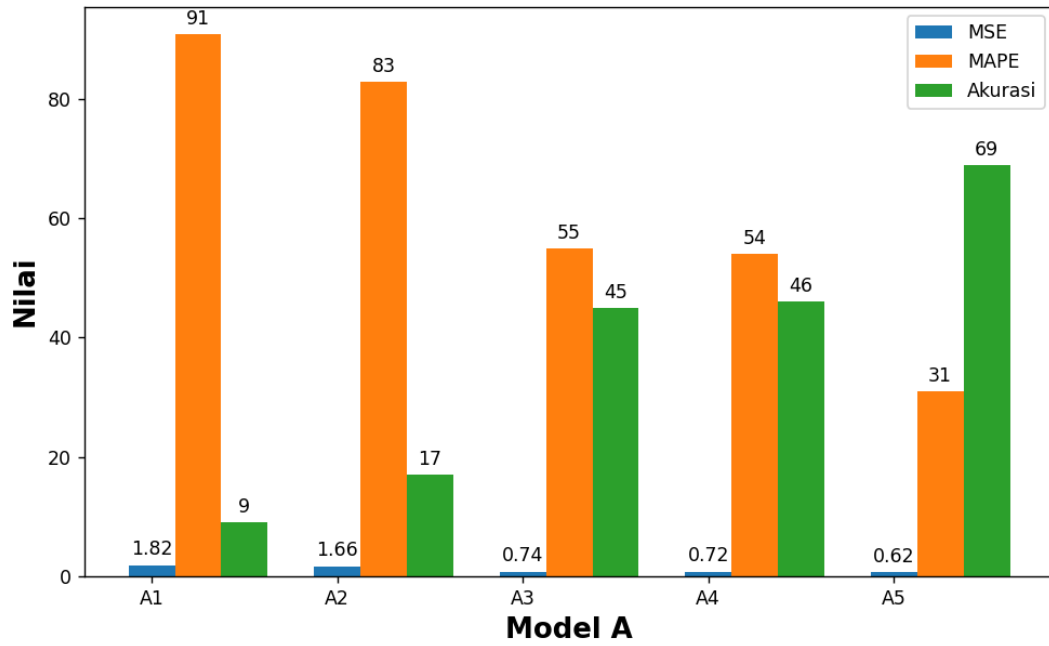
### 4.3 Pembahasan

Untuk mengukur tingkat kerusakan sektor setelah terjadinya bencana alam, dapat dilakukan menggunakan *forward propagation in neural network*. Hal tersebut dapat diketahui berdasarkan uji coba yang telah dilakukan pada 25 model yang dibuat, yang dibedakan berdasarkan jumlah *hidden layer* dan *epoch*. Berikut merupakan perbandingan MSE, MAPE, dan akurasi yang dilakukan pada seluruh model yang dibentuk.

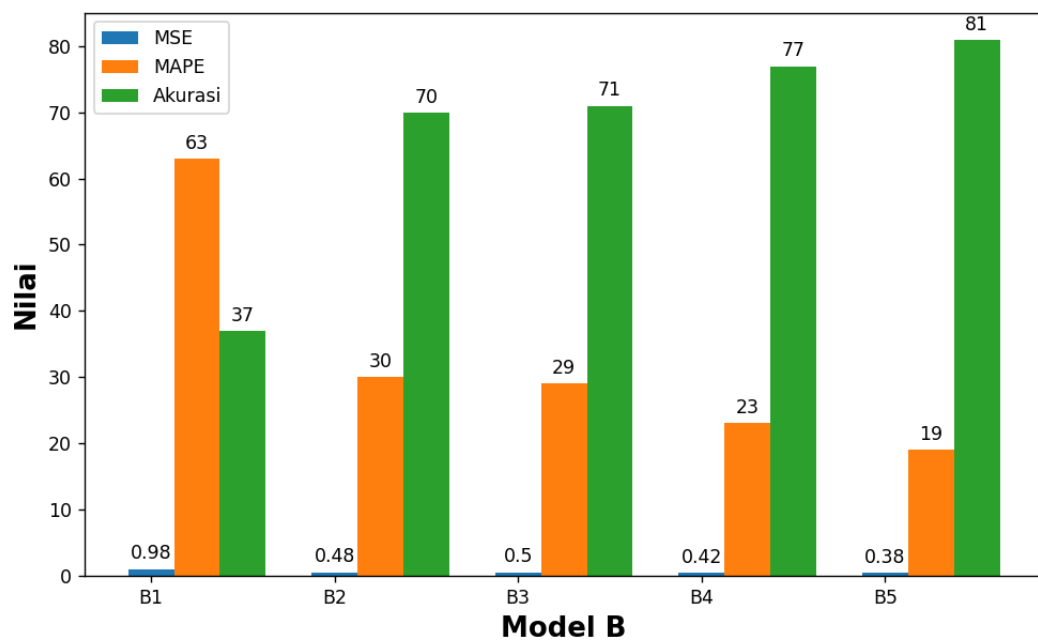
Tabel 4. 31 Hasil MSE, MAPE, dan Akurasi Setiap Model Data

Model	Epoch	Hidden Layer	MSE	MAPE (%)	Akurasi (%)
A1	25	1	1,82	91	9
A2		2	1,66	83	17
A3		3	0,74	55	45
A4		4	0,72	54	46
A5		5	0,62	31	69

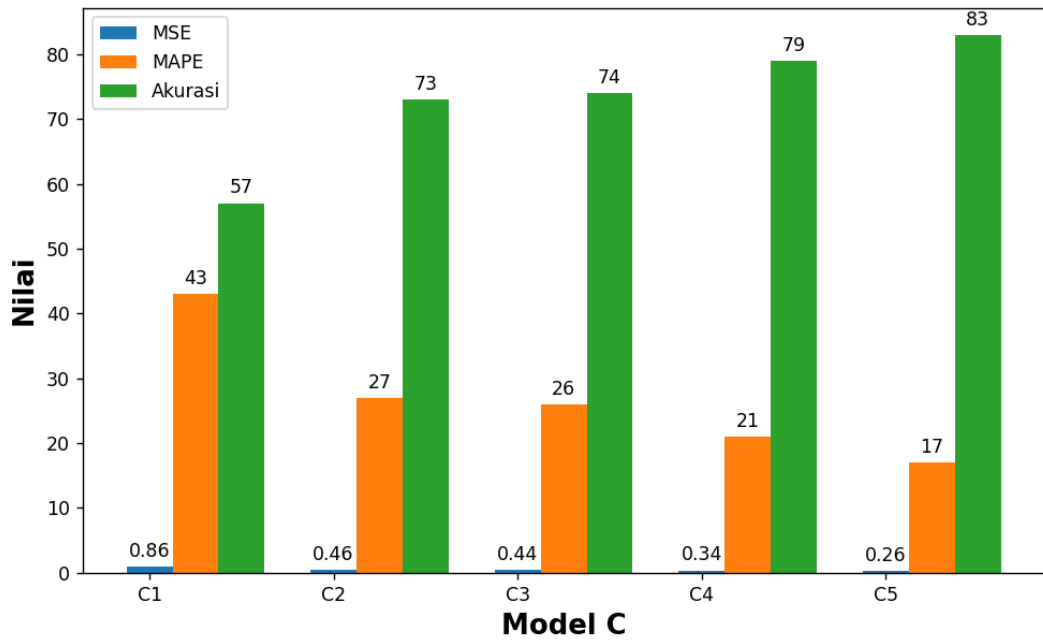
<b>Model</b>	<b>Epoch</b>	<b>Hidden Layer</b>	<b>MSE</b>	<b>MAPE (%)</b>	<b>Akurasi (%)</b>
B1	50	1	0,98	63	37
B2		2	0,48	30	70
B3		3	0,5	29	71
B4		4	0,42	23	77
B5		5	0,38	19	81
C1	75	1	0,86	43	57
C2		2	0,46	27	73
C3		3	0,44	26	74
C4		4	0,34	21	79
C5		5	0,26	17	83
D1	100	1	0,7	35	65
D2		2	0,44	25	75
D3		3	0,4	21	79
D4		4	0,38	19	81
D5		5	0,36	18	82
E1	125	1	0,38	19	81
E2		2	0,34	17	83
E3		3	0,2	10	90
E4		4	0,2	4,67	95,33
E5		5	0,06	3	97



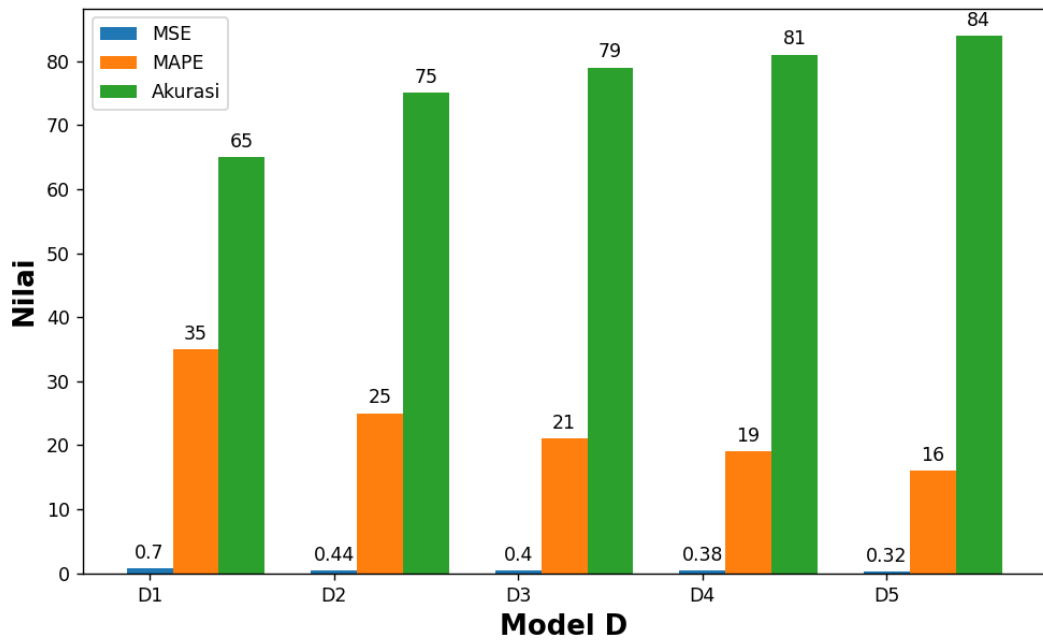
Gambar 4. 62 Grafik MSE, MAPE, dan Akurasi Model A



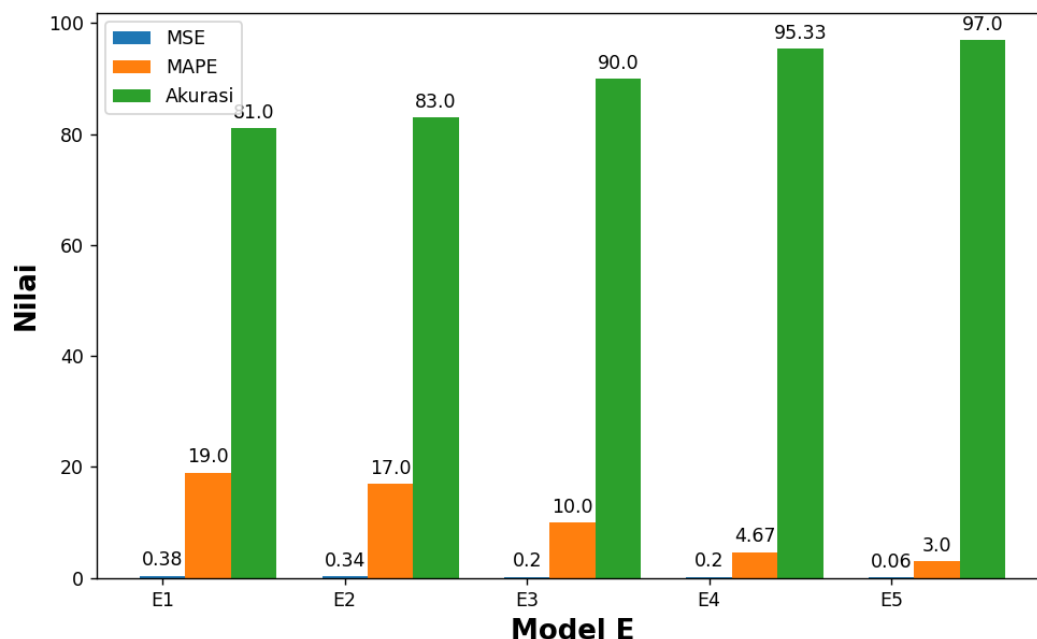
Gambar 4. 63 Grafik MSE, MAPE, dan Akurasi Model B



Gambar 4. 64 Grafik MSE, MAPE, dan Akurasi Model C



Gambar 4. 65 Grafik MSE, MAPE, dan Akurasi Model D



Gambar 4. 66 Grafik MSE, MAPE, dan Akurasi Model E

Berdasarkan tabel 4.31 serta gambar 4.62 hingga 4.66 tersebut, dapat diketahui bahwa akurasi terbesar ada pada model E5, yang menggunakan *hidden layer* sejumlah 5 dan *epoch* sebanyak 125. Model E5 sendiri memiliki nilai akurasi sebesar 97%, yang merupakan nilai akurasi terbesar dibandingkan dengan model-model yang lain. Hal tersebut dapat diketahui dari perhitungan MSE, MAPE, dan akurasi yang berdasarkan hasil uji coba yang didapatkan dari seluruh model yang ada.

Berdasarkan hasil uji coba yang telah dilakukan, dapat diketahui bahwa pada *forward propagation in neural network*, jumlah *hidden layer* dan *epoch* berpengaruh pada besar akurasi perhitungan metode yang dihasilkan. Semakin banyak jumlah *epoch* yang dilakukan, maka akurasi juga akan semakin besar. Begitu pula dengan jumlah *hidden layer*, semakin banyak jumlah *hidden layer* yang digunakan, maka tingkat akurasi sistem juga akan semakin besar.



#### 4.4 Integrasi Islam

Segala hal yang menimpa seseorang dan tidak disukai disebut dengan kata *musibah*, yang berasal dari bahasa Arab (Hidayatullah, 2013). Kata *musibah* sendiri kemudian diserap ke dalam bahasa Indonesia dan paling banyak digunakan untuk menggambarkan konsep bencana. Dalam al-Qur'an sendiri, kata *musibah* juga digunakan untuk menggambarkan konsep bencana, yang disebutkan dalam kurang lebih 50 ayat (Thaib, 2021).

Segala bencana yang menimpa manusia, sesungguhnya tidaklah terjadi terkecuali dengan izin Allah. Seperti pada firman Allah pada QS. At-Taghabun ayat 11 sebagai berikut.

مَا أَصَابَ مِنْ مُصِيبَةٍ إِلَّا بِإِذْنِ اللَّهِ ۗ وَمَنْ يُؤْمِنْ بِاللَّهِ يَهْدِ اللَّهُ قَلْبَهُ ۗ وَاللَّهُ بِكُلِّ شَيْءٍ عَلِيمٌ

*“Tidak ada sesuatu musibah yang menimpa (seseorang), kecuali dengan izin Allah; dan barangsiapa beriman kepada Allah, niscaya Allah akan memberi petunjuk kepada hatinya. Dan Allah Maha Mengetahui segala sesuatu.”* (QS. At-Taghabun : 11).

Ayat tersebut sesungguhnya merupakan dasar atau hakikat keimanan, yang menjelaskan bahwa segala sesuatu yang terjadi merupakan takdir Allah dan hanya terjadi atas izin Allah (Zaini, 2020). Hal tersebut merupakan pengingat bagi orang yang tertimpa bencana atau musibah bahwa segala hal yang terjadi merupakan atas kehendak Allah. Dengan begitu, orang yang beriman dapat akan tetap tenang hatinya walaupun sedang tertimpa bencana, sedangkan orang yang sempat lalai dapat menginstropeksi diri dan kembali teringat kepada Allah. Selain itu, bagi orang

yang ingkar, bencana atau musibah yang terjadi merupakan hukuman dari Allah atas kesalahan-kesalahan yang telah diperbuat.

Meskipun bencana sendiri menimbulkan banyak kerugian, sesungguhnya banyak hikmah yang dapat diambil dari kejadian bencana. Allah berfirman pada QS. Al-Baqarah ayat 155 – 157 sebagai berikut.

وَلَنَبْلُوَنَّكُمْ بِشَيْءٍ مِّنَ الْخَوْفِ وَالْجُوعِ وَنَقْصٍ مِّنَ الْأَمْوَالِ وَالْأَنْفُسِ وَالتَّمَرَّتِ ۗ وَيَبْشُرِ الصَّابِرِينَ ۝ الَّذِينَ إِذَا

أَصَابَتْهُمْ مُصِيبَةٌ قَالُوا إِنَّا لِلَّهِ وَإِنَّا إِلَيْهِ رَاجِعُونَ ۝ أُولَئِكَ عَلَيْهِمْ صَلَوَاتٌ مِّن رَّبِّهِمْ وَرَحْمَةٌ وَأُولَئِكَ هُمُ الْمُهْتَدُونَ

*“Dan sungguh akan Kami berikan cobaan kepadamu, dengan sedikit ketakutan, kelaparan, kekurangan harta, jiwa dan buah-buahan. Dan berikanlah berita gembira kepada orang-orang yang sabar. (yaitu) orang-orang yang apabila ditimpa musibah, mereka mengucapkan: "Inna lillaahi wa innaa ilaihi raaji'uun (sesungguhnya kami milik Allah dan sesungguhnya kami sedang menuju kemabali kepada-Nya). Mereka itulah yang mendapat keberkatan yang sempurna dan rahmat dari Tuhan mereka dan mereka itulah orang-orang yang mendapat petunjuk.” (QS. Al-Baqarah : 155 – 157).*

Berdasarkan buku “Tafsir Al-Mishbah”, Surah Al-Baqarah ayat 155 – 157 menjelaskan sesungguhnya hakikat kehidupan dunia ditandai dengan salah satunya yakni terdapat berbagai macam cobaan atau ujian (Shihab, 2009). Cobaan atau ujian yang diberikan Allah tersebut pada hakikatnya sedikit, karena cobaan masih bisa terjadi dalam bentuk yang lebih besar, juga apabila dibandingkan dengan ganjaran dan imbalan yang nantinya akan diterima. Berbagai bentuk cobaan atau ujian tersebut sesungguhnya merupakan nikmat yang besar, karena kita dapat mempersiapkan diri dan naik ke tingkat yang lebih baik. Kita diperintahkan untuk bersabar, karena seluruh apa yang terdapat di dunia ini hanyalah milik Allah semata dan segala cobaan atau ujian tersebut pastilah ada hikmah dibalikinya. Orang-orang

yang selalu bersabar ketika menghadapi cobaan atau ujian yang diberikan akan mendapatkan rahmat dan petunjuk, tidak hanya untuk mengatasi kesulitan, tetapi juga untuk kebahagiaan duniawi dan ukhrawi.

Bencana sebagai cobaan maupun ujian sesungguhnya mengajarkan manusia agar dapat mengambil hikmah dari segala peristiwa yang terjadi. Hikmah yang dapat diambil antara lain terhadap Allah, manusia, dan alam. Hal tersebut akan dapat meningkatkan derajat manusia di mata Allah juga meningkatkan kualitas hidup dengan ber-*tasamuh* (berbuat baik) pada sesama.

Pada penelitian ini sendiri, diterapkan 3 konsep muamalah. Muamalah merupakan hubungan manusia, baik itu dengan Allah, sesama manusia, maupun dengan alam. Tiga konsep muamalah yang ada pada penelitian ini antara lain *mualah ma'a Allah*, *muamalah ma'a an-nas*, dan *muamalah ma'a al-alam*.

#### **4.4.1 Muamalah Ma'a Allah**

Bencana alam atau musibah yang menimpa manusia sesungguhnya memiliki banyak hikmah yang dapat diambil. Dengan adanya musibah, hendaknya menjadi pengingat untuk mendekatkan diri kepada Allah atau *taqarrub ila Allah* (Hakim, 2013). Musibah juga merupakan salah satu bentuk Allah dalam menguji keimanan manusia. Bencana alam maupun musibah sendiri mengajarkan untuk meningkatkan keimanan, sebab segala yang ada di langit dan bumi hanyalah milik Allah, sedangkan manusia hanya bisa menerima takdir-Nya .

Dalam menerima semua musibah dan ketentuan dari Allah sendiri, kita harus menghadapinya dengan ikhlas dan penuh kesabaran (Abdullah, 2021). Pahala

yang diberikan kepada orang-orang yang bersabar ketika mendapatkan musibah sangatlah besar. Seperti pada hadis Rasulullah sebagai berikut.

عَنْ عَائِشَةَ أَنَّ رَسُولَ اللَّهِ صَلَّى اللَّهُ عَلَيْهِ وَسَلَّمَ قَالَ مَا مِنْ مُصِيبَةٍ يُصَابُ بِهَا الْمُسْلِمُ إِلَّا كُفِرَ بِهَا عَنْهُ حَتَّى الشُّوْكَةِ

يُشَاكُّهَا

“Aisyah RA berkata, “Rasulullah SAW bersabda, "Tidak ada satupun musibah (cobaan) yang menimpa seorang Muslim, melainkan dosanya dihapus Allah Ta'ala karenanya, sekalipun musibah itu hanya karena tertusuk duri.” (HR Muslim no. 4667).

#### 4.4.2 Muamalah Ma'a an-Nas

Penelitian yang dilakukan memiliki tujuan untuk menolong sesama manusia. Hal tersebut antara lain membantu tim surveyor dalam menentukan tingkat kerusakan bangunan pasca terjadinya bencana alam, agar waktu yang dibutuhkan lebih efisien. Sehingga, tindakan proses rehabilitasi dan rekonstruksi setelah terjadinya bencana alam oleh pemerintah dapat dilakukan dengan lebih cepat. Allah berfirman pada QS. Al-Maidah ayat 2 sebagai berikut.

وَتَعَاوَنُوا عَلَى الْبِرِّ وَالتَّقْوَىٰ وَلَا تَعَاوَنُوا عَلَى الْإِثْمِ وَالْعُدْوَانِ يَوْمَئِذٍ اللَّهُ شَدِيدُ الْعِقَابِ

“Saling menolonglah kamu dalam melakukan kebajikan dan taqwa. Dan jangan saling menolong pada perbuatan yang dosa dan permusuhan. Bertakwalah kepada Allah SWT. Sebenarnya siksaan Allah SWT sangatlah pedih.” (QS. Al-Maidah : 2).

Pada firman Allah tersebut, ditegaskan bahwa sikap saling tolong menolong dapat membangun kerukunan dalam bermasyarakat. Pada “Tafsir Al-Mishbah”, dijelaskan bahwa ayat tersebut menegaskan bahwa tolong-menolong merupakan

prinsip dasar dalam menjalin kerja sama dengan siapa pun, selama hal tersebut bertujuan dalam kebaikan dan ketakwaan, bukan dalam hal keburukan atau kemungkarannya (Shihab, 2009). Sikap tolong-menolong sendiri merupakan perilaku yang dapat memberikan manfaat bagi orang lain, dimana kita membantu meringankan beban yang dimiliki orang lain dengan melakukan suatu tindakan nyata.

#### **4.4.3 Muamalah Ma'a al-Alam**

Kejadian bencana alam sesungguhnya merupakan suatu peringatan bahwa kerusakan dan bencana yang terjadi merupakan ulah tangan manusia sendiri (Hakim, 2013). Hal tersebut terdapat pada firman Allah yakni QS. Ar-Rum ayat 41 sebagai berikut.

ظَهَرَ الْفَسَادُ فِي الْبَرِّ وَالْبَحْرِ بِمَا كَسَبَتْ أَيْدِي النَّاسِ لِيُذِيقَهُمْ بَعْضَ الَّذِي عَمِلُوا لَعَلَّهُمْ يَرْجِعُونَ

*“Telah nampak kerusakan di darat dan di laut disebabkan karena perbuatan tangan manusia, supaya Allah merasakan kepada mereka sebagian dari (akibat) perbuatan mereka, agar mereka kembali (ke jalan yang benar)”*. (QS Ar-Rum: 41)

Dalam ayat tersebut, Allah memperingatkan bahwa kerusakan yang terjadi di atas bumi merupakan akibat dari tidak bisa mengendalikan hawa nafsu. Berdasarkan buku “Tafsir Al-Mishbah”, ayat tersebut menjelaskan bahwa pelanggaran yang dilakukan manusia mengakibatkan gangguan keseimbangan yang terjadi di darat maupun di laut, akibat perbuatan manusia yang menyimpang. Semakin banyak kerusakan yang dibuat oleh manusia, maka juga akan semakin besar dampak buruknya terhadap manusia itu sendiri (Shihab, 2009). Bencana alam

yang terjadi adalah peringatan dari Allah agar para manusia merasakan akibat dari perbuatan buruk yang telah mereka lakukan, agar kembali ke jalan yang lurus. Bencana alam seperti tanah longsor dan banjir merupakan salah satu contoh bahwa bencana dapat terjadi akibat ulah tangan manusia, yang enggan dalam menjaga bumi. Sehingga, dengan adanya bencana alam, kita diharuskan untuk senantiasa menjaga dan merawat lingkungan, dimulai dari lingkungan sekitar kita.

## BAB V

### KESIMPULAN DAN SARAN

#### 5.1 Kesimpulan

Pada penelitian ini, diimplementasikan metode TOPSIS dengan bahasa pemrograman PHP untuk mengolah *data training* yang nantinya digunakan dalam membangun model *machine learning*. Pengujian data yang dilakukan pada penelitian ini berbasis *machine learning*, yakni menggunakan metode *Neural Networks* (NN) dengan algoritma *forward propagation*. *Data testing* yang digunakan dalam pengujian merupakan data baru yang di-*input*-kan langsung ke dalam sistem yang dibuat menggunakan bahasa pemrograman Python. Dalam pengujian data menggunakan metode *forward propagation in neural network*, dibentuk 25 model yang dibedakan berdasarkan jumlah *hidden layer* dan jumlah *epoch*. Dari hasil pengujian tersebut kemudian dapat diketahui nilai MSE, MAPE, dan akurasi dari tiap model yang dibentuk. Berdasarkan hasil perhitungan metode FPNN yang dilakukan, diketahui bahwa tingkat akurasi sistem tertinggi dimiliki oleh model E5 yang memiliki *epoch* sebanyak 125 dan *hidden layer* sejumlah 5, dengan nilai MSE sebesar 0,06, MAPE sebesar 3%, serta akurasi sebesar 97%. Hal tersebut menunjukkan bahwa pada *forward propagation in neural network*, jumlah *hidden layer* dan *epoch* berpengaruh pada besar akurasi perhitungan metode yang dihasilkan. Semakin banyak jumlah *epoch* dan *hidden layer*, maka tingkat akurasi sistem juga akan semakin besar.

## 5.2 Saran

Penelitian yang telah dilakukan tentu saja belum sempurna dan masih terdapat banyak kekurangan. Berikut merupakan beberapa saran untuk penelitian selanjutnya.

- a. Jumlah data *training* dan *testing* yang digunakan dalam pengujian dapat lebih banyak lagi, agar tingkat akurasi yang dihasilkan oleh sistem yang dibuat bisa lebih tinggi.
- b. Penelitian yang dilakukan dapat dikembangkan lebih lanjut dengan menggunakan metode lain, untuk mengetahui metode mana yang lebih cocok dalam penentuan tingkat kerusakan sektor setelah terjadinya bencana alam.
- c. Sistem yang dibuat pada penelitian ini dapat diimplementasikan langsung pada instansi bencana terkait, agar manfaat dari penelitian yang dilakukan dapat direalisasikan.
- d. Sistem yang dibuat pada penelitian ini dapat dikembangkan lebih lanjut dalam bentuk aplikasi *mobile* agar lebih praktis digunakan oleh *user*.



## DAFTAR PUSTAKA

- Abdullah. (2021). *Bimbingan Perawatan Rohani Islam Bagi Orang Sakit*.
- Afida, B. A., Kamal, M., & Hadmoko, D. S. (2020). Identifikasi Kerusakan Bangunan Pasca Gempa Bumi Menggunakan Citra Satelit Worldview-2. *Jurnal Pengembangan Kota*, 8(1), 67–77. <https://doi.org/10.14710/jpk.8.1.67-77>
- Almais, A. T., Fatchurrohman, F., & Holle, K. F. H. (2020). Implementasi fuzzy weighted product penyusunan aksi rehabilitasi rekonstruksi pasca bencana berbasis decision support system dynamic. *Jurnal Eltek*, 18(1), 1. <https://doi.org/10.33795/eltek.v18i1.171>
- Almais, A. T. W., Crysdiyan, C., Holle, K. F. H., & Roihan, A. (2022). *Smart Assessment Menggunakan Backpropagation Neural Network Smart Assessment using Backpropagation Neural Network*. 21(3). <https://doi.org/10.30812/matrik.v21i3.1382>
- Almais, A. T. W., Fatchurrohman, ., Holle, K. F. H., Kinasih, K. S., Wiranti, D. A., & Yasin, S. Y. (2020). *Implementation Fuzzy Weighted Product Preparation Post Disaster Reconstruction and Rehabilitation Action based Dynamics Decision Support System*. 272–277. <https://doi.org/10.5220/0009909002720277>
- Bachriwindi, A., Putra, E. K., Munawaroh, U. M., & Almais, A. T. W. (2019). Implementation of Web-Based Weighted Product Use Decision Support System to Determine the Post-Disaster Damage and Loss. *Journal of Physics: Conference Series*, 1413(1). <https://doi.org/10.1088/1742-6596/1413/1/012019>
- BNPB. (2016). Risiko Bencana Indonesia (Disasters Risk of Indonesia). *International Journal of Disaster Risk Science*, 22. <https://doi.org/10.1007/s13753-018-0186-5>
- Çelikkbilek, Y., & Tüysüz, F. (2020). An in-depth review of theory of the TOPSIS method: An experimental analysis. *Journal of Management Analytics*, 7(2), 281–300. <https://doi.org/10.1080/23270012.2020.1748528>
- Chakraborty, S. (2022). TOPSIS and Modified TOPSIS: A comparative analysis. *Decision Analytics Journal*, 2(December 2021), 100021. <https://doi.org/10.1016/j.dajour.2021.100021>
- Chamid, A. A. (2016). Penerapan Metode Topsis Untuk Menentukan Prioritas Kondisi Rumah. *Simetris : Jurnal Teknik Mesin, Elektro Dan Ilmu Komputer*, 7(2), 537. <https://doi.org/10.24176/simet.v7i2.765>
- Damanik, S., & Utomo, D. P. (2020). Implementasi Metode ROC (Rank Order Centroid) Dan Waspas Dalam Sistem Pendukung Keputusan Pemilihan

Kerjasama Vendor. ... *Teknologi Informasi Dan ...*, 4, 242–248.  
<https://doi.org/10.30865/komik.v4i1.2690>

*Data Bencana Indonesia*. (2022). <https://Bnpb.Go.Id/>.

Deore, S. (2021). *What do you mean by Forward Propagation in ANN?*  
<https://medium.com/analytics-vidhya/what-do-you-mean-by-forward-propagation-in-ann-9a89c80dac1b>

Dhanaseely, A. J., Himavathi, S., & Srinivasan, E. (2012). Performance comparison of cascade and feed forward neural network for face recognition system. *IET Seminar Digest*, 2012(4). <https://doi.org/10.1049/ic.2012.0154>

Gong, X., Fong, S., & Si, Y. W. (2018). Fast multi-subsequence monitoring on streaming time-series based on Forward-propagation. *Information Sciences*, 450, 73–88. <https://doi.org/10.1016/j.ins.2018.03.023>

Hakim, A. (2013). MAKNA BENCANA MENURUT AL-QUR ' AN: Kajian Fenomena Terhadap Bencana di Indonesia A . Pendahuluan Agama Islam yang diturunkan Allah memiliki ajaran untuk kehidupan umat manusia secara menyeluruh di baik di dunia dan di akhirat . Ajaran Islam bersumberkan. *Hermeunetik*, 7(2), 279–296.

Hidayatullah, M. (2013). PersPektif al-Qur'an tentang Bencana alam. *Hermeunetik*, 7(2), 261–278.

Hirasawa, K., Ohbayashi, M., Koga, M., & Harada, M. (1996). Forward propagation universal learning network. *IEEE International Conference on Neural Networks - Conference Proceedings*, 1, 353–358. <https://doi.org/10.1109/icnn.1996.548917>

Kag, A., & Saligrama, V. (2021). Training Recurrent Neural Networks via Forward Propagation Through Time. *Proceedings of the 38th International Conference on Machine Learning*, 139, 5189–5200. <https://proceedings.mlr.press/v139/kag21a.html>

Khaze, S. R., Masdari, M., & Hojjatkah, S. (2013). Application of Artificial Neural Networks in Estimating Participation in Elections. *International Journal of Information Technology, Modeling and Computing*, 1(3), 23–31. <https://doi.org/10.5121/ijitmc.2013.1303>

Kumar, T. S. (2020). Data Mining Based Marketing Decision Support System Using Hybrid Machine Learning Algorithm. *Journal of Artificial Intelligence and Capsule Networks*, 2(3), 185–193. <https://doi.org/10.36548/jaicn.2020.3.006>

Mathew, T. V., Pramod, A. L. N., Ooi, E. T., & Natarajan, S. (2020). An efficient forward propagation of multiple random fields using a stochastic Galerkin scaled boundary finite element method. *Computer Methods in Applied Mechanics and Engineering*, 367, 112994. <https://doi.org/10.1016/j.cma.2020.112994>

- Merkert, J., Mueller, M., & Hubl, M. (2015). A survey of the application of machine learning in decision support systems. *23rd European Conference on Information Systems, ECIS 2015, 2015-May*, 1–15.
- Mesran, M., Diansyah, T. M., & Fadlina, F. (2019). Implemententasi Metode Rank Order Cendroid (ROC) dan Operational Competitiveness Rating Analysis (OCRA) dalam Penilaian Kinerja Dosen Komputer Menerapkan (Studi Kasus: STMIK Budi Darma). *Prosiding Seminar Nasional Riset Information Science (SENARIS), 1*(September), 822. <https://doi.org/10.30645/senaris.v1i0.89>
- Permadi, B. (2020). *Strategi Nasional Kecerdasan Artifisial Indonesia*. 194. <https://ai-innovation.id/server/static/ebook/stranas-ka.pdf>
- Putra, I. S. W., Hermawan, F., & Hatmoko, J. U. D. (2020). Penilaian Kerusakan Dan Kerugian Infrastruktur Publik Akibat Dampak Bencana Banjir Di Kota Semarang. *Wahana Teknik Sipil: Jurnal Pengembangan Teknik Sipil, 25*(2), 86. <https://doi.org/10.32497/wahanats.v25i2.2154>
- Ren, L., Zhang, Y., Wang, Y., & Sun, Z. (2007). Comparative analysis of a novel M-TOPSIS method and topsis. *Applied Mathematics Research EXpress, 2007*, 1–10. <https://doi.org/10.1093/amrx/abm005>
- Safitri, A. H., Almais, A. T. W., Syauqi, A., & Melani, R. I. (2022). Pengujian Optimization dan Non-Optimization Query Metode Topsis untuk Menentukan Tingkat Kerusakan Sektor Bencana Alam. *Jurnal ELTIKOM, 6*(1), 89–99. <https://doi.org/10.31961/eltikom.v6i1.532>
- Sari, D. R., Windarto, A. P., Hartama, D., & Solikhun, S. (2018). Decision Support System for Thesis Graduation Recommendation Using AHP-TOPSIS Method. *Jurnal Teknologi Dan Sistem Komputer, 6*(1), 1–6. <https://doi.org/10.14710/jtsiskom.6.1.2018.1-6>
- Sazli, M. H. (2006). A BRIEF REVIEW OF FEED-FORWARD NEURAL NETWORKS. *Commun. Fac. Sci. Univ. Ank. Series A2-A3, 50*(1), 11–17.
- Shihab, M. Q. (2009). *Tafsir Al-Mishbah : Pesan, Kesan, dan Keserasian al-Qur'an* (1st ed.). Lentera Hati.
- Surahman, A., & Nursadi, N. (2019). Sistem Pendukung Keputusan Kenaikan Gaji Karyawan Dengan Metode Topsis Berbasis Web. *JTKSI (Jurnal Teknologi Komputer Dan Sistem Informasi), 2*(3), 82–87.
- Sushanti, I. R., Ridha, R., Yuniarman, A., & Hamdi, A. I. (2020). Strategi Penanggulangan Kerusakan Rumah Tinggal Pasca Bencana Gempa Bumi Di Kawasan Permukiman. *Jurnal Planoearth PWK FT Universitas Muhammadiyah Mataram, 2*, 17–24.
- Thaib, Z. bin H. (2021). Bencana Dalam Perspektif Al-Qur'an. *Prosiding Mitigasi Bencana, Universitas Dharmawangsa, November 2021, November*, 15–26. <https://proceeding.dharmawangsa.ac.id/index.php/PROSUNDHAR/article/view/3>

- Undang-Undang Republik Indonesia Nomor 24 Tahun 2007 Tentang Penanggulangan Bencana, (2007).
- Wang, F., Liao, F., & Zhu, H. (2020). FPA-DNN: A Forward Propagation Acceleration based Deep Neural Network for Ship Detection. *Proceedings of the International Joint Conference on Neural Networks*. <https://doi.org/10.1109/IJCNN48605.2020.9207603>
- Wee, L. K., Min, T. Y., Arooj, A., & Supriyanto, E. (2010). Nuchal translucency marker detection based on artificial neural network and measurement via bidirectional iteration forward propagation. *WSEAS Transactions on Information Science and Applications*, 7(8), 1025–1036.
- Zaini, H. (2020). Bencana Menurut Perspektif Al-Qur'an. *El-Hekam*, 4(1), 1. <https://doi.org/10.31958/jeh.v4i1.1998>
- Zyoud, S. H., & Fuchs-Hanusch, D. (2017). A bibliometric-based survey on AHP and TOPSIS techniques. *Expert Systems with Applications*, 78, 158–181. <https://doi.org/10.1016/j.eswa.2017.02.016>

## LAMPIRAN

### Lampiran 1 Data Hasil *Preprocessing*

No.	Jenis Bencana	Tanggal kejadian	Lokasi	Sektor	K1	K2	K3	K4	K5
1	Tanah Longsor	03/01/2020	Jl. Embong Brantas II, Kidul Dalem, Klojen	Plengsengan & pos bermain	3	3	3	3	3
2	Pohon Tumbang	04/01/2020	Jl. Raya Walet, Sukun, Sukun	warung	2	2	2	2	1
3	Tanah Longsor	07/01/2020	Jl. Kanjuruhan IV no. 3, Tlogomas, Lowokwaru	Plengsengan	3	3	3	2	2
4	Tanah Longsor	16/01/2020	Jl. Kanjuruhan IV no. 3, Tlogomas, Lowokwaru	plengsengan	3	3	3	3	2
5	Tanah Longsor	21/01/2020	Jl. Embong Brantas II, Kidul Dalem, Klojen	Plengsengan & pos bermain	3	3	3	2	3
6	Pohon Tumbang	22/01/2020	Jl. Ir. H. Juanda Gg. IX, Jodipan, Blimbing	rumah	1	1	1	2	1
7	Pohon Tumbang	24/01/2020	Jl. Embong Brantas, Jodipan, Blimbing	rumah	1	1	2	1	1
8	Pohon Tumbang	25/01/2020	Jl. Bromo Gg. II no. 8, Oro-oro dowo, Klojen	rumah	2	2	2	2	1

9	Tanah Longsor	26/01/2020	Jl. MT. Haryono Gg.21, Dinoyo, Lowokwaru	plataran rumah	2	2	2	2	2
10	Tanah Longsor	26/01/2020	Jl. MT. Haryono XI, Dinoyo, Lowokwaru	plengsengan	3	3	3	3	2
11	Tanah Longsor	30/01/2020	Jl. Kolonel Sugiono V, Mergosono, Kedungkandang	rumah	3	3	2	2	3
12	Tanah longsor	03/01/2020	Jl. Embong brantas II, Kelurahan Kidul Dalem, Kecamatan Klojen	plengsengan	2	2	2	3	2
13	pohon tumbang	04/01/2020	Jl. Raya Walet, Kelurahan Sukun, Kecamatan Sukun	warung	1	1	1	2	2
14	Tanah longsor	07/01/2020	Jl. Kanjuruhan IV No.03 RT.04 RW.03, Kelurahan Tlogomas, Kecamatan Lowokwaru	plengsengan	3	3	3	3	2
15	Tanah longsor	16/01/2020	Jl. Kanjuruhan IV No.03 RT.04 RW.03, Kelurahan Tlogomas, Kecamatan Lowokwaru	plengsengan	3	3	3	2	1
16	Tanah longsor	21/01/2020	Jl. Embong Brantas II, Kelurahan Kiduldalem, Kecamatan Klojen	plengsengan & tempat bermain	3	3	3	3	2

17	pohon tumbang	22/01/2020	Jl. Ir. H. Juanda Gg. IX, Kampung Lampion, Kelurahan Jodipan, Kecamatan Blimbing	rumah	1	1	1	2	1
18	pohon tumbang	24/01/2020	Jl. Embong Brantas, Kelurahan Jodipan, Kecamatan Blimbing	rumah	1	1	1	2	1
19	angin kencang	25/01/2020	Jl. Bromo Gg. II No. 8, Kelurahan Oro-oro dowo, Kecamatan Klojen	rumah	2	2	2	3	2
20	tanah longsor	26/01/2020	Jl. MT. Haryono Gg.21 No.41, Kelurahan Dinoyo, Kecamatan Lowokwaru	plataran	3	3	2	3	1
21	Tanah longsor	26/01/2020	Jl. Mt. Haryono XI, RT.02/RW.03, Kelurahan : Dinoyo, Kecamatan : Lowokwaru	plengsengan	3	3	3	3	2
22	Tanah longsor	30/01/2020	Jl. Kol. Sugiono V, RT: 015/RW: 003, Kelurahan : Mergosono, Kecamatan : Kedungkandang	rumah	2	2	2	3	2
23	Banjir	01/02/2020	Jl. Barend Raya IIA / 394, Barend, Klojen	rumah	1	1	1	2	1
24	Tanah Longsor	01/02/2020	Jl. S. Supriadi, Sukun, Sukun	rumah	2	2	2	3	2
25	Tanah Longsor	07/02/2020	Pos Kamling, Sawojajar, Kedungkandang	pos kamling	3	3	3	2	2
26	Tanah Longsor	07/02/2020	Jl. Raya Balearjosari, Balearjosari, Blimbing	plengsengan	3	3	3	3	2

27	Tanah Longsor	10/02/2020	Jl. Jaksa Agung Suprpto I, Samaan, Klojen	plengsengan	2	2	2	3	2
28	Tanah Longsor	15/02/2020	Jl. Kyai Sofyan Yusuf, Kedungkandang, Kedungkandang	rumah	2	2	2	2	1
29	Banjir	16/02/2020	Jl. Ir. Rais Gg.IX	*rumah terendam	1	1	1	1	1
30	Banjir	16/02/2020	Sentra Industri Sanitair, Karangbesuki, Sukun	*rumah terendam	1	1	1	1	1
31	Banjir	16/02/2020	Jl. Bareng Raya, Bareng, Klojen	*rumah terendam	1	1	1	1	1
32	Banjir	16/02/2020	Jl. Soekarno Hatta Indah no. 1A, Mojolangu, Lowokwaru	*rumah terendam	1	1	1	1	1
33	Banjir	16/02/2020	Jl. Bandungrejosari, Bandungrejosari, Sukun	*rumah terendam	1	1	1	1	1
34	Tanah Longsor	16/02/2020	Jl. Klayatan III no.14, Bandungrejosari, Sukun	rumah	1	1	1	2	1
35	cuaca ekstrim	17/02/2020	Jl. Kalimantan, Kasin, Klojen	pujasera	2	2	2	1	1
36	Tanah Longsor	16/02/2020	Jl. Tembesi, Rampal Celaket, Klojen	plengsengan	2	2	2	2	3
37	Tanah Longsor	20/02/2020	Jl. Candi Mendut Selatan blok. VI, Tulusrejo, Lowokwaru	pembatas jalan	3	3	3	2	1
38	Tanah Longsor	20/02/2020	Jl. Bareng Taman Bunga 59, Bareng, Klojen	rumah	2	2	2	2	3



39	Tanah Longsor	21/02/2020	Jl. Madyopuro Gg.10, Madyopuro, Kedungkandang	plengsengan & rumah2	3	3	3	3	2
40	Tanah Longsor	21/02/2020	Jl. Brigjen Slamet Riadi Gg. XVII no.52, Oro-oro dowo, Klojen	rumah	2	2	2	3	2
41	Tanah Longsor	24/02/2020	Jl. Muharto Gg. Vc, Kotalama, Kedungkandang	pelengsengan & bangunan	3	3	3	3	2
42	cuaca ekstrim	24/02/2020	Jl. Gede, Oro-oro Dowo, Klojen	warung	1	1	1	2	1
43	Tanah Longsor	26/02/2020	Jl. Simpang Mega, Pisangcandi, Sukun	DAM	3	3	3	2	2
44	Tanah Longsor	26/02/2020	Jl. S. Supriadi, Sukun, Sukun	rumah	3	3	3	3	3
45	Tanah Longsor	26/02/2020	Jl. Muharto Gg VII, Kotalama, Kedungkandang	plengsengan	3	3	3	2	2
46	Tanah Longsor	27/02/2020	Jl. Kecipir, Bumiayu, Kedungkandang	rumah	3	3	2	3	2
47	Tanah Longsor	27/02/2020	Jl. Muharto Gg. Vc, Kotalama, Kedungkandang	plengsengan	3	3	3	3	2
48	Tanah Longsor	27/02/2020	Jl. Simpang Sukun, Sukun, Sukun	plengsengan	2	2	2	3	2
49	Tanah Longsor	28/02/2020	Jl. Janti Barat no.7, Bandungrejosari, Sukun	rumah (pohon menimpa)	2	2	2	2	1
50	Tanah Longsor	28/02/2020	Jl. Rawisari II no. 53, Mulyorejo, Sukun	rumah	1	1	1	1	1

51	Tanah longsor	01/02/2020	Jl. S. Supriadi VII/61 RT.05 RW.02, Kelurahan Sukun, Kecamatan Sukun	plengsengan	2	2	2	3	2
52	Tanah longsor	07/02/2020	Pos kamling RW 06 Kel. Sawojajar, Jl. Simp. Ranugrati, Kelurahan Sawojajar, Kecamatan Kedungkandang	plengsengan	3	3	3	3	1
53	Tanah longsor	07/02/2020	Perumahan Kalindra, Jl. Raya Balearjosari, Kelurahan Balearjosari Kecamatan Blimbing	plengsengan	3	3	3	3	1
54	banjir	10/02/2020	Perum Piranha Village	rumah	2	2	2	3	2
55	Tanah longsor	10/02/2020	Jl. Jaksa Agung Suprpto I, RT.11/RW.03, Kelurahan Samaan, Kecamatan Klojen	plengsengan	3	3	3	3	2
56	Tanah longsor	15/02/2020	Jl. Kyai Sofyan Yusuf, RT.04/RW.01, Kelurahan Kedungkandang, Kecamatan Kedungkandang	rumah	3	3	3	3	2
57	Tanah longsor	16/02/2020	Jl. Klayatan III No.14, RT.10/RW.02, Kelurahan Bandungrejosari, Kecamatan Sukun	rumah	1	1	1	1	1

58	pohon tumbang	17/02/2020	Jl. Kalimantan ( Depan Pujasera TNI AL), Kelurahan Kasin, Kecamatan Klojen	pujasera	1	1	1	2	1
59	Tanah longsor	20/02/2020	Jl. Candi mendut selatan Blok VI RT. 03/RW.11, Kelurahan Tulusrejo, Kecamatan Lowokwaru	pagar pembatas	3	3	2	2	1
60	Tanah longsor	20/02/2020	Rumah Bapak Slamet Supriadi, Jl. Bareng Taman Bunga 59 RT.05/RW.03, Kelurahan Bareng, Kecamatan Klojen	rumah	2	2	2	3	2
61	angin kencang	21/02/2020	Jl. Pelabuhan Tanjung Priok, Kel. Bakalankrajan, Kec. Sukun	rumah	1	1	1	2	2
62	Tanah longsor	21/02/2020	Jl. Madyopuro Gg. 10 RT.05/RW.02, Kelurahan Madyopuro, Kecamatan Kedungkandang	plengsengan & rumah	3	3	3	3	2
63	Tanah longsor	21/02/2020	Jl. Brigjen Slamet Riadi Gg. XVII No.52, RT.04/RW.06, Kelurahan Oro-Oro Dowo, Kecamatan Klojen	rumah	3	3	3	3	2
64	Tanah longsor	23/02/2020	Jl. Rawisari, RT.09/RW.05, Kelurahan Mulyorejo, Kecamatan Sukun	rumah	1	1	1	2	2

65	Tanah longsor	24/02/2020	Jl. Muharto Gg. VC RT.03/RW.09, Kelurahan Kotalama, Kecamatan Kedungkandang	plengsengan & bangunan	3	3	3	3	2
66	pohon tumbang	24/02/2020	Jl. Gede, Kelurahan Oro oro dowo, Kecamatan Klojen	warung	1	1	1	2	1
67	Tanah longsor	26/02/2020	Jl. Simpang Mega Mendung, Kelurahan Pisangcandi, Kecamatan Sukun	rumah	1	1	1	2	1
68	Tanah longsor	26/02/2020	Jl. S. Supriadi Gg. X RT.17/RW.06, Kelurahan Sukun, Kecamatan Sukun	rumah	2	2	2	3	2
69	Tanah longsor	26/02/2020	Jl. Muharto Gg VII RT.06 / RW.10, Kelurahan Kotalama, Kecamatan Kedungkandang	plengsengan & taman bermain	3	3	3	3	2
70	tanah ambles	27/02/2020	Jl. Kecipir RT.04/RW.03, Kelurahan Bumiayu, Kecamatan Kedungkandang	rumah	2	2	2	3	2
71	Tanah longsor	27/02/2020	Jl. Muharto Gg. VC RT.03/RW.09, Kelurahan Kotalama, Kecamatan Kedungkandang	plengsengan	3	3	3	3	2

72	Tanah longsor	27/02/2020	Jl. Simpang Sukun, Kelurahan Sukun, Kecamatan Sukun	rumah	1	1	1	2	1
73	Tanah longsor	28/02/2020	Rumah Bapak Totok Junaedi, Jl. Janti Barat No. 7 RT.01/RW.08, Kelurahan Bandungrejosari, Kecamatan Sukun	rumah	2	2	2	3	2
74	Tanah longsor	28/02/2020	Jl. Rawisari II No. 53 RT.01/RW.05, Kelurahan Mulyorejo, Kecamatan Sukun	rumah	1	1	1	2	1
75	Tanah Longsor	02/03/2020	Jl. Sawojajar Gg. IX, Sawojajar, Kedungkandang	Rumah	2	2	2	2	2
76	Tanah Longsor	02/03/2020	Jl. Simpang Kepuh No. 37, Bandungrejosari, Sukun	Rumah	3	2	2	2	2
77	Tanah Longsor	02/03/2020	Jl. S. Supriadi X no. 15, Sukun, Sukun	Rumah	2	1	2	2	1
78	Tanah Longsor	04/03/2020	Jl. Kalimosodo XI No.42, Polehan, blimbing	Rumah	3	2	2	2	2
79	Tanah Longsor	06/03/2020	Jl. Pulosari I Blok O, Purwodadi, Blimbing	tembok	3	3	3	3	2
80	Tanah Longsor	11/03/2020	Pos Kamling, Purwodadi, Blimbing	pos kamling	3	3	3	3	2
81	Tanah Longsor	12/03/2020	Pos Kamling Jl. Gadang GG. 17B, Gadang, Sukun	rumah	3	3	3	3	2

82	tanah longsor	02/03/2020	Jl. Sawojajar Gg. IX, Kelurahan Sawojajar, Kecamatan Kedungkandang	rumah	2	2	1	2	1
83	Tanah longsor	02/03/2020	Perumahan Dieng Residence, Kelurahan Pisangcandi, Kecamatan Sukun	plengsengan	3	3	3	3	1
84	Tanah longsor	02/03/2020	Jl. S. Supriadi X No. 15 RT.18/RW.06, Kelurahan Sukun, Kecamatan Sukun	rumah	2	2	2	3	2
85	Tanah longsor	06/03/2020	Jl. Pulosari I Blok O Rw. 07/ Rt. 07 Kel. Purwodadi Kec. Blimbing	plengsengan & tembok	3	3	2	3	2
86	Tanah longsor	12/03/2020	Pos Kamling RT.03/RW.11, Jl. Bandara Narita RT.03/RW.11, Kelurahan Cemorokandang, Kecamatan Kedungkandang	pos kamling	3	3	3	3	2
87	Tanah Longsor	04/02/2020	Jl. Mergosono Gg. IX No. 11, Mergosono, Kedungkandang	Rumah	3	2	2	2	2
88	Tanah Longsor	04/02/2020	Jl. Kebalen Wetan Gg. Mawar No.31, Kotalama, Kedungkandang	Plengsengan	3	3	2	3	2
89	Tanah Longsor	04/08/2020	Jl. Muharto Gg.VB , Kotalama, Kedungkandang	Rumah	3	2	2	2	3

90	Tanah Longsor	04/09/2020	Jl. Hamid Rusdi Timur Gg VI, Bunulrejo, Blimbing	Plengsengan	3	3	2	3	2
91	Tanah Longsor	04/10/2020	Jl. Gadang Gg VII, Gadang, Sukun	Plengsengan	1	1	1	2	1
92	Tanah Longsor	17/4/2020	Jl. Embong Brantas Gg. II, Kiduldalem, Klojen	Rumah	1	1	1	2	1
93	Tanah Longsor	18/5/2020	Jl. Tapaksiring 24C, Samaan, Klojen	Rumah	2	2	2	2	1
94	Tanah Longsor	23/5/2020	Jl. Simppang Sukun GG. Anggur No. 15, Sukun, Sukun	Rumah	2	2	2	2	2
95	Tanah Longsor	26/5/2020	Jl. Muharto Gg. VC No.26 , Kotalama, Kedungkandang	Plengsengan	3	2	3	3	2
96	Tanah Longsor	31/5/2020	Jl. Hamid Rusdi Timur No.21, Bunulrejo, Blimbing	Plengsengan	3	2	2	3	2
97	Cuaca Ekstrem	16/08/2020	Jl. Bulutangkis, Tasikmadu, Lowokwaru	Rumah	3	3	3	3	2
98	Cuaca Ekstrem (Angin Kencang)	02/10/2020	Jl. Batanghari, Rampel Celaket, Klojen	Rumah	3	3	3	3	2
99	Tanah Longsor	01/11/2020	Jl. S.Supriadi X No.10, Bandungrejosari, Sukun	Rumah	3	3	3	3	2

100	Tanah Longsor	22/11/2020	Jl. Mayjend Sungkono, Kedungkandang, Kedungkandang	Plengsengan	3	3	2	3	2
-----	---------------	------------	---	-------------	---	---	---	---	---

### Lampiran 2 Data Hasil Metode TOPSIS

No.	Jenis Bencana	Tanggal kejadian	Lokasi	Sektor	K1	K2	K3	K4	K5	Result	Keterangan
1	Tanah Longsor	03/01/2020	Jl. Embong Brantas II, Kidul Dalem, Klojen	Plengsengan & pos bermain	3	3	3	3	3	3	Rusak Berat
2	Pohon Tumbang	04/01/2020	Jl. Raya Walet, Sukun, Sukun	warung	2	2	2	2	1	2	Rusak Sedang
3	Tanah Longsor	07/01/2020	Jl. Kanjuruhan IV no. 3, Tlogomas, Lowokwaru	Plengsengan	3	3	3	2	2	3	Rusak Berat
4	Tanah Longsor	16/01/2020	Jl. Kanjuruhan IV no. 3, Tlogomas, Lowokwaru	plengsengan	3	3	3	3	2	3	Rusak Berat
5	Tanah Longsor	21/01/2020	Jl. Embong Brantas II, Kidul Dalem, Klojen	Plengsengan & pos bermain	3	3	3	2	3	3	Rusak Berat



6	Pohon Tumbang	22/01/2020	Jl. Ir. H. Juanda Gg. IX, Jodipan, Blimbing	rumah	1	1	1	2	1	1	Rusak Ringan
7	Pohon Tumbang	24/01/2020	Jl. Embong Brantas, Jodipan, Blimbing	rumah	1	1	2	1	1	1	Rusak Ringan
8	Pohon Tumbang	25/01/2020	Jl. Bromo Gg. II no. 8, Oro-oro dowo, Klojen	rumah	2	2	2	2	1	2	Rusak Sedang
9	Tanah Longsor	26/01/2020	Jl. MT. Haryono Gg.21, Dinoyo, Lowokwaru	plataran rumah	2	2	2	2	2	2	Rusak Sedang
10	Tanah Longsor	26/01/2020	Jl. MT. Haryono XI, Dinoyo, Lowokwaru	plengsengan	3	3	3	3	2	3	Rusak Berat
11	Tanah Longsor	30/01/2020	Jl. Kolonel Sugiono V, Mergosono, Kedungkandang	rumah	3	3	2	2	3	3	Rusak Berat
12	Tanah longsor	03/01/2020	Jl. Embong brantas II, Kelurahan Kidul Dalem, Kecamatan Klojen	plengsengan	2	2	2	3	2	2	Rusak Sedang
13	pohon tumbang	04/01/2020	Jl. Raya Walet, Kelurahan Sukun, Kecamatan Sukun	warung	1	1	1	2	2	1	Rusak Ringan

14	Tanah longsor	07/01/2020	Jl. Kanjuruhan IV No.03 RT.04 RW.03, Kelurahan Tlogomas, Kecamatan Lowokwaru	plengsengan	3	3	3	3	2	3	Rusak Berat
15	Tanah longsor	16/01/2020	Jl. Kanjuruhan IV No.03 RT.04 RW.03, Kelurahan Tlogomas, Kecamatan Lowokwaru	plengsengan	3	3	3	2	1	3	Rusak Berat
16	Tanah longsor	21/01/2020	Jl. Embong Brantas II, Kelurahan Kiduldalem, Kecamatan Klojen	plengsengan & tempat bermain	3	3	3	3	2	3	Rusak Berat
17	pohon tumbang	22/01/2020	Jl. Ir. H. Juanda Gg. IX, Kampung Lampion, Kelurahan Jodipan, Kecamatan Blimbing	rumah	1	1	1	2	1	1	Rusak Ringan
18	pohon tumbang	24/01/2020	Jl. Embong Brantas, Kelurahan Jodipan, Kecamatan Blimbing	rumah	1	1	1	2	1	1	Rusak Ringan

19	angin kencang	25/01/2020	Jl. Bromo Gg. II No. 8, Kelurahan Oro-oro dowo, Kecamatan Klojen	rumah	2	2	2	3	2	2	Rusak Sedang
20	tanah longsor	26/01/2020	Jl. MT. Haryono Gg.21 No.41, Kelurahan Dinoyo, Kecamatan Lowokwaru	plataran	3	3	2	3	1	3	Rusak Berat
21	Tanah longsor	26/01/2020	Jl. Mt. Haryono XI, RT.02/RW.03, Kelurahan : Dinoyo, Kecamatan : Lowokwaru	plengsengan	3	3	3	3	2	3	Rusak Berat
22	Tanah longsor	30/01/2020	Jl. Kol. Sugiono V, RT: 015/RW: 003, Kelurahan : Mergosono, Kecamatan : Kedungkandang	rumah	2	2	2	3	2	2	Rusak Sedang
23	Banjir	01/02/2020	Jl. Bareng Raya IIA / 394, Bareng, Klojen	rumah	1	1	1	2	1	1	Rusak Ringan
24	Tanah Longsor	01/02/2020	Jl. S. Supriadi, Sukun, Sukun	rumah	2	2	2	3	2	2	Rusak Sedang

25	Tanah Longsor	07/02/2020	Pos Kamling, Sawojajar, Kedungkandang	pos kamling	3	3	3	2	2	3	Rusak Berat
26	Tanah Longsor	07/02/2020	Jl. Raya Balarjosari, Balarjosari, Blimbing	plengsengan	3	3	3	3	2	3	Rusak Berat
27	Tanah Longsor	10/02/2020	Jl. Jaksa Agung Suprpto I, Samaan, Klojen	plengsengan	2	2	2	3	2	2	Rusak Sedang
28	Tanah Longsor	15/02/2020	Jl. Kyai Sofyan Yusuf, Kedungkandang, Kedungkandang	rumah	2	2	2	2	1	2	Rusak Sedang
29	Banjir	16/02/2020	Jl. Ir. Rais Gg.IX	*rumah terendam	1	1	1	1	1	1	Rusak Ringan
30	Banjir	16/02/2020	Sentra Industri Sanitair, Karangbesuki, Sukun	*rumah terendam	1	1	1	1	1	1	Rusak Ringan
31	Banjir	16/02/2020	Jl. Bareng Raya, Bareng, Klojen	*rumah terendam	1	1	1	1	1	1	Rusak Ringan
32	Banjir	16/02/2020	Jl. Soekarno Hatta Indah no. 1A, Mojolangu, Lowokwaru	*rumah terendam	1	1	1	1	1	1	Rusak Ringan

33	Banjir	16/02/2020	Jl. Bandungrejosari, Bandungrejosari, Sukun	*rumah terendam	1	1	1	1	1	1	Rusak Ringan
34	Tanah Longsor	16/02/2020	Jl. Klayatan III no.14, Bandungrejosari, Sukun	rumah	1	1	1	2	1	1	Rusak Ringan
35	cuaca ekstrim	17/02/2020	Jl. Kalimantan, Kasin, Klojen	pujasera	2	2	2	1	1	2	Rusak Sedang
36	Tanah Longsor	16/02/2020	Jl. Tembesi, Rampal Celaket, Klojen	plengsengan	2	2	2	2	3	2	Rusak Sedang
37	Tanah Longsor	20/02/2020	Jl. Candi Mendut Selatan blok. VI, Tulusrejo, Lowokwaru	pembatas jalan	3	3	3	2	1	3	Rusak Berat
38	Tanah Longsor	20/02/2020	Jl. Bareng Taman Bunga 59, Bareng, Klojen	rumah	2	2	2	2	3	2	Rusak Sedang
39	Tanah Longsor	21/02/2020	Jl. Madyopuro Gg.10, Madyopuro, Kedungkandang	plengsengan & rumah2	3	3	3	3	2	3	Rusak Berat
40	Tanah Longsor	21/02/2020	Jl. Brigjen Slamet Riadi Gg. XVII no.52, Oro-oro dowo, Klojen	rumah	2	2	2	3	2	2	Rusak Sedang

41	Tanah Longsor	24/02/2020	Jl. Muharto Gg. Vc, Kotalama, Kedungkandang	pelengsengan & bangunan	3	3	3	3	2	3	Rusak Berat
42	cuaca ekstrim	24/02/2020	Jl. Gede, Oro-oro Dowo, Klojen	warung	1	1	1	2	1	1	Rusak Ringan
43	Tanah Longsor	26/02/2020	Jl. Simpang Mega, Pisangcandi, Sukun	DAM	3	3	3	2	2	3	Rusak Berat
44	Tanah Longsor	26/02/2020	Jl. S. Supriadi, Sukun, Sukun	rumah	3	3	3	3	3	3	Rusak Berat
45	Tanah Longsor	26/02/2020	Jl. Muharto Gg VII, Kotalama, Kedungkandang	plengsengan	3	3	3	2	2	3	Rusak Berat
46	Tanah Longsor	27/02/2020	Jl. Kecipir, Bumiayu, Kedungkandang	rumah	3	3	2	3	2	3	Rusak Berat
47	Tanah Longsor	27/02/2020	Jl. Muharto Gg. Vc, Kotalama, Kedungkandang	plengsengan	3	3	3	3	2	3	Rusak Berat
48	Tanah Longsor	27/02/2020	Jl. Simpang Sukun, Sukun, Sukun	plengsengan	2	2	2	3	2	2	Rusak Sedang
49	Tanah Longsor	28/02/2020	Jl. Janti Barat no.7, Bandungrejosari, Sukun	rumah (pohon menimpa)	2	2	2	2	1	2	Rusak Sedang

50	Tanah Longsor	28/02/2020	Jl. Rawisari II no. 53, Mulyorejo, Sukun	rumah	1	1	1	1	1	1	Rusak Ringan
51	Tanah longsor	01/02/2020	Jl. S. Supriadi VII/61 RT.05 RW.02, Kelurahan Sukun, Kecamatan Sukun	plengsengan	2	2	2	3	2	2	Rusak Sedang
52	Tanah longsor	07/02/2020	Pos kamling RW 06 Kel. Sawojajar, Jl. Simp. Ranugrati, Kelurahan Sawojajar, Kecamatan Kedungkandang	plengsengan	3	3	3	3	1	3	Rusak Berat
53	Tanah longsor	07/02/2020	Perumahan Kalindra, Jl. Raya Balearjosari, Kelurahan Balearjosari Kecamatan Blimbing	plengsengan	3	3	3	3	1	3	Rusak Berat
54	banjir	10/02/2020	Perum Piranha Village	rumah	2	2	2	3	2	2	Rusak Sedang
55	Tanah longsor	10/02/2020	Jl. Jaksa Agung Suprpto I, RT.11/RW.03, Kelurahan Samaan, Kecamatan Klojen	plengsengan	3	3	3	3	2	3	Rusak Berat

56	Tanah longsor	15/02/2020	Jl. Kyai Sofyan Yusuf, RT.04/RW.01, Kelurahan Kedungkandang, Kecamatan Kedungkandang	rumah	3	3	3	3	2	3	Rusak Berat
57	Tanah longsor	16/02/2020	Jl. Klayatan III No.14, RT.10/RW.02, Kelurahan Bandungrejosari, Kecamatan Sukun	rumah	1	1	1	1	1	1	Rusak Ringan
58	pohon tumbang	17/02/2020	Jl. Kalimantan ( Depan Pujasera TNI AL), Kelurahan Kasin, Kecamatan Klojen	pujasera	1	1	1	2	1	1	Rusak Ringan
59	Tanah longsor	20/02/2020	Jl. Candi mendut selatan Blok VI RT. 03/RW.11, Kelurahan Tulusrejo, Kecamatan Lowokwaru	pagar pembatas	3	3	2	2	1	3	Rusak Berat
60	Tanah longsor	20/02/2020	Rumah Bapak Slamet Supriadi, Jl. Bareng Taman Bunga 59 RT.05/RW.03,	rumah	2	2	2	3	2	2	Rusak Sedang



			Kelurahan Bareng, Kecamatan Klojen								
61	angin kencang	21/02/2020	Jl. Pelabuhan Tanjung Priok, Kel. Bakalankrajan, Kec. Sukun	rumah	1	1	1	2	2	1	Rusak Ringan
62	Tanah longsor	21/02/2020	Jl. Madyopuro Gg. 10 RT.05/RW.02, Kelurahan Madyopuro, Kecamatan Kedungkandang	plengsengan & rumah	3	3	3	3	2	3	Rusak Berat
63	Tanah longsor	21/02/2020	Jl. Brigjen Slamet Riadi Gg. XVII No.52, RT.04/RW.06, Kelurahan Oro-Oro Dowo, Kecamatan Klojen	rumah	3	3	3	3	2	3	Rusak Berat
64	Tanah longsor	23/02/2020	Jl. Rawisari, RT.09/RW.05, Kelurahan Mulyorejo, Kecamatan Sukun	rumah	1	1	1	2	2	1	Rusak Ringan
65	Tanah longsor	24/02/2020	Jl. Muharto Gg. VC RT.03/RW.09, Kelurahan	plengsengan & bangunan	3	3	3	3	2	3	Rusak Berat

			Kotalama, Kecamatan Kedungkandang								
66	pohon tumbang	24/02/2020	Jl. Gede, Kelurahan Oro oro dowo, Kecamatan Klojen	warung	1	1	1	2	1	1	Rusak Ringan
67	Tanah longsor	26/02/2020	Jl. Simpang Mega Mendung, Kelurahan Pisangcandi, Kecamatan Sukun	rumah	1	1	1	2	1	1	Rusak Ringan
68	Tanah longsor	26/02/2020	Jl. S. Supriadi Gg. X RT.17/RW.06, Kelurahan Sukun, Kecamatan Sukun	rumah	2	2	2	3	2	2	Rusak Sedang
69	Tanah longsor	26/02/2020	Jl. Muharto Gg VII RT.06 / RW.10, Kelurahan Kotalama, Kecamatan Kedungkandang	plengsengan & taman bermain	3	3	3	3	2	3	Rusak Berat
70	tanah ambles	27/02/2020	Jl. Kecipir RT.04/RW.03, Kelurahan Bumiayu, Kecamatan Kedungkandang	rumah	2	2	2	3	2	2	Rusak Sedang
71	Tanah longsor	27/02/2020	Jl. Muharto Gg. VC RT.03/RW.09, Kelurahan	plengsengan	3	3	3	3	2	3	Rusak Berat

			Kotalama, Kecamatan Kedungkandang								
72	Tanah longsor	27/02/2020	Jl. Simpang Sukun, Kelurahan Sukun, Kecamatan Sukun	rumah	1	1	1	2	1	1	Rusak Ringan
73	Tanah longsor	28/02/2020	Rumah Bapak Totok Junaedi, Jl. Janti Barat No. 7 RT.01/RW.08, Kelurahan Bandungrejosari, Kecamatan Sukun	rumah	2	2	2	3	2	2	Rusak Sedang
74	Tanah longsor	28/02/2020	Jl. Rawisari II No. 53 RT.01/RW.05, Kelurahan Mulyorejo, Kecamatan Sukun	rumah	1	1	1	2	1	1	Rusak Ringan
75	Tanah Longsor	02/03/2020	Jl. Sawojajar Gg. IX, Sawojajar, Kedungkandang	Rumah	2	2	2	2	2	2	Rusak Sedang
76	Tanah Longsor	02/03/2020	Jl. Simpang Kepuh No. 37, Bandungrejosari, Sukun	Rumah	3	2	2	2	2	3	Rusak Berat

77	Tanah Longsor	02/03/2020	Jl. S. Supriadi X no. 15, Sukun, Sukun	Rumah	2	1	2	2	1	2	Rusak Sedang
78	Tanah Longsor	04/03/2020	Jl. Kalimosodo XI No.42, Polehan, blimbing	Rumah	3	2	2	2	2	3	Rusak Berat
79	Tanah Longsor	06/03/2020	Jl. Pulosari I Blok O, Purwodadi, Blimbing	tembok	3	3	3	3	2	3	Rusak Berat
80	Tanah Longsor	11/03/2020	Pos Kamling, Purwodadi, Blimbing	pos kamling	3	3	3	3	2	3	Rusak Berat
81	Tanah Longsor	12/03/2020	Pos Kamling Jl. Gadang GG. 17B, Gadang, Sukun	rumah	3	3	3	3	2	3	Rusak Berat
82	tanah longsor	02/03/2020	Jl. Sawojajar Gg. IX, Kelurahan Sawojajar, Kecamatan Kedungkandang	rumah	2	2	1	2	1	2	Rusak Sedang
83	Tanah longsor	02/03/2020	Perumahan Dieng Residence, Kelurahan Pisangcandi, Kecamatan Sukun	plengsengan	3	3	3	3	1	3	Rusak Berat

84	Tanah longsor	02/03/2020	Jl. S. Supriadi X No. 15 RT.18/RW.06, Kelurahan Sukun, Kecamatan Sukun	rumah	2	2	2	3	2	2	Rusak Sedang
85	Tanah longsor	06/03/2020	Jl. Pulosari I Blok O Rw. 07/ Rt. 07 Kel. Purwodadi Kec. Blimbing	plengsengan & tembok	3	3	2	3	2	3	Rusak Berat
86	Tanah longsor	12/03/2020	Pos Kamling RT.03/RW.11, Jl. Bandara Narita RT.03/RW.11, Kelurahan Cemorokandang, Kecamatan Kedungkandang	pos kamling	3	3	3	3	2	3	Rusak Berat
87	Tanah Longsor	04/02/2020	Jl. Mergosono Gg. IX No. 11, Mergosono, Kedungkandang	Rumah	3	2	2	2	2	3	Rusak Berat
88	Tanah Longsor	04/02/2020	Jl. Kebalen Wetan Gg. Mawar No.31, Kotalama, Kedungkandang	Plengsengan	3	3	2	3	2	3	Rusak Berat
89	Tanah Longsor	04/08/2020	Jl. Muharto Gg.VB , Kotalama, Kedungkandang	Rumah	3	2	2	2	3	3	Rusak Berat

90	Tanah Longsor	04/09/2020	Jl. Hamid Rusdi Timur Gg VI, Bunulrejo, Blimbing	Plengsengan	3	3	2	3	2	3	Rusak Berat
91	Tanah Longsor	04/10/2020	Jl. Gadang Gg VII, Gadang, Sukun	Plengsengan	1	1	1	2	1	1	Rusak Ringan
92	Tanah Longsor	17/4/2020	Jl. Embong Brantas Gg. II, Kiduldalem, Klojen	Rumah	1	1	1	2	1	1	Rusak Ringan
93	Tanah Longsor	18/5/2020	Jl. Tapaksiring 24C, Samaan, Klojen	Rumah	2	2	2	2	1	2	Rusak Sedang
94	Tanah Longsor	23/5/2020	Jl. Simppang Sukun GG. Anggur No. 15, Sukun, Sukun	Rumah	2	2	2	2	2	2	Rusak Sedang
95	Tanah Longsor	26/5/2020	Jl. Muharto Gg. VC No.26 , Kotalama, Kedungkandang	Plengsengan	3	2	3	3	2	3	Rusak Berat
96	Tanah Longsor	31/5/2020	Jl. Hamid Rusdi Timur No.21, Bunulrejo, Blimbing	Plengsengan	3	2	2	3	2	3	Rusak Berat
97	Cuaca Ekstrem	16/08/2020	Jl. Bulutangkis, Tasikmadu, Lowokwaru	Rumah	3	3	3	3	2	3	Rusak Berat

98	Cuaca Ekstrem (Angin Kencang)	02/10/2020	Jl. Batanghari, Rampel Celaket, Klojen	Rumah	3	3	3	3	2	3	Rusak Berat
99	Tanah Longsor	01/11/2020	Jl. S.Supriadi X No.10, Bandungrejosari, Sukun	Rumah	3	3	3	3	2	3	Rusak Berat
100	Tanah Longsor	22/11/2020	Jl. Mayjend Sungkono, Kedungkandang, Kedungkandang	Plengsengan	3	3	2	3	2	3	Rusak Berat

### Lampiran 3 Data *Training*

K1	K2	K3	K4	K5	Data Aktual
3	3	3	3	3	3
2	2	2	2	1	2
3	3	3	2	2	3
3	3	3	3	2	3
3	3	3	2	3	3
1	1	1	2	1	1
1	1	2	1	1	1
2	2	2	2	1	2
2	2	2	2	2	2
3	3	3	3	2	3
3	3	2	2	3	3
2	2	2	3	2	2
1	1	1	2	2	1
3	3	3	3	2	3
3	3	3	2	1	3
3	3	3	3	2	3
1	1	1	2	1	1
1	1	1	2	1	1
2	2	2	3	2	2
3	3	2	3	1	3
3	3	3	3	2	3
2	2	2	3	2	2
1	1	1	2	1	1
2	2	2	3	2	2
3	3	3	2	2	3



3	3	3	3	2	3
2	2	2	3	2	2
2	2	2	2	1	2
1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	2	1	1
2	2	2	1	1	2
2	2	2	2	3	2
3	3	3	2	1	3
2	2	2	2	3	2
3	3	3	3	2	3
2	2	2	3	2	2
3	3	3	3	2	3
1	1	1	2	1	1
3	3	3	2	2	3
3	3	3	3	3	3
3	3	3	2	2	3
3	3	2	3	2	3
3	3	3	3	2	3
2	2	2	3	2	2
2	2	2	2	1	2
1	1	1	1	1	1
2	2	2	3	2	2
3	3	3	3	1	3

3	3	3	3	1	3
2	2	2	3	2	2
3	3	3	3	2	3
3	3	3	3	2	3
1	1	1	1	1	1
1	1	1	2	1	1
3	3	2	2	1	3
2	2	2	3	2	2
1	1	1	2	2	1
3	3	3	3	2	3
3	3	3	3	2	3
1	1	1	2	2	1
3	3	3	3	2	3
1	1	1	2	1	1
1	1	1	2	1	1
2	2	2	3	2	2
3	3	3	3	2	3
2	2	2	3	2	2
3	3	3	3	2	3
1	1	1	2	1	1
2	2	2	3	2	2
1	1	1	2	1	1
2	2	2	2	2	2
3	2	2	2	2	3
2	1	2	2	1	2
3	2	2	2	2	3
3	3	3	3	2	3

3	3	3	3	2	3
3	3	3	3	2	3
2	2	1	2	1	2
3	3	3	3	1	3
2	2	2	3	2	2
3	3	2	3	2	3
3	3	3	3	2	3
3	2	2	2	2	3
3	3	2	3	2	3
3	2	2	2	3	3
3	3	2	3	2	3
1	1	1	2	1	1
1	1	1	2	1	1
2	2	2	2	1	2
2	2	2	2	2	2
3	2	3	3	2	3
3	2	2	3	2	3
3	3	3	3	2	3
3	3	3	3	2	3
3	3	3	3	2	3
3	3	2	3	2	3

**Lampiran 4 Data Testing**

No.	Jenis Bencana	Tanggal kejadian	Lokasi	Sektor	K1	K2	K3	K4	K5
1	Tanah longsor	20/03/2020	Jl. Aris Munandar Gg. IV RT.06/RW.03, Kelurahan Kiduldalem, Kecamatan Klojen	rumah	1	1	1	2	1
2	pohon tumbang	20/03/2020	Jl. Tenaga Utara RW.05, Kelurahan Blimbing, Kecamatan Blimbing	rumah	1	1	1	2	1
3	Tanah longsor	26/03/2020	Jl. Kalimosodo XI No.42 RT.06/RW.06, Kelurahan Polehan, Kecamatan Blimbing	rumah	2	2	3	3	2
4	Tanah longsor	26/03/2020	Jl. Raya Candi VI D2, Kelurahan Karangbesuki, Kecamatan Sukun	plengsengan	3	3	3	3	2
5	Tanah longsor	29/03/2020	Jl. Simpang sukun 54A, Kelurahan Sukun, Kecamatan Sukun	rumah	2	2	3	3	2
6	Tanah longsor	30/03/2020	Jl. Tapak Siring IV, Kel. Samaan, Kec. Klojen	rumah	3	3	3	3	3
7	Tanah longsor	29/03/2020	Jl. Gadang Gg. 17 <sup>B</sup> Rt.05/Rw.03, Kelurahan Gadang, Kecamatan Sukun	rumah	2	2	2	3	1

8	Tanah longsor	02/04/2020	Jl. Mergosono Gg. IX No. 11 RT.06/Rw.01, Kelurahan Mergosono, Kecamatan Kedungkandang	rumah	2	2	2	3	2
9	Tanah longsor	02/04/2020	Rumah Bapak Komar Jl. Kebalen Wetan Gg. Mawar No.31 Rt.09./Rw.04, Kelurahan Kotalama, Kecamatan Kedungkandang	plengsengan	3	3	3	3	2
10	Tanah longsor	08/04/2020	Jl. Muharto Gg.VB RT.13/RW.08, Kelurahan Kotalama, Kecamatan Kedungkandang	rumah	3	3	3	3	2
11	Tanah longsor	09/04/2020	Jl. Hamid Rusdi Timur Gg VI RT.02/RW.16, Kelurahan Bunulrejo, Kecamatan Blimbing	plengsengan & tembok rumah	2	2	2	3	3
12	Tanah longsor	10/04/2020	Jl. Gadang Gg VII RT.11/RW.01, Kelurahan Gadang, Kecamatan Sukun	plengsengan	2	2	2	3	1
13	Tanah longsor	17/04/2020	Jl. Embong Brantas Gg. II RT.04/RW.06, Kelurahan Kiduldalem, Kecamatan Klojen	rumah	1	1	1	2	1

14	Tanah longsor	18/05/2020	Jl. Tapaksiring 24C, Kelurahan Samaan, Kecamatan Klojen	rumah	1	1	1	2	1
15	tanah longsor	23/05/2020	Jl. Simpang Sukun Gg. Anggur No. 15 RT.02/RW.04, Kel. Sukun, Kec. Sukun	rumah	2	2	2	3	2
16	tanah longsor	26/05/2020	Jl. Muharto Gg. VC RT.03/RW.09 No. 26, Kelurahan Kotalama, Kecamatan Kedungkandang	plengsengan	2	2	2	3	2
17	tanah longsor	31/05/2020	Jl. Hamid Rusdi Timur No. 21 RT.01/RW.07, Kelurahan Bunulrejo, Kecamatan Blimbing	rumah	2	2	2	3	2
18	angin putting beliung	16/06/2020	Jl. Bulutangkis, Kelurahan:Tasikmadu, Kecamatan: Lowokwaru	rumah bpk julia	2	2	2	3	1
19				rumah bpk siswandi	2	2	2	3	2
20				rumah bpk syamsul anas	2	2	2	3	1
21				rumah bpk suwartono	1	1	1	2	1

22				rumah ibu sumarmi	1	1	1	1	1
23				rumah bpk usman	1	1	1	1	1
24				rumah bpk ahmad munijin	1	1	1	2	1
25				rumah bpk anjono	1	1	1	2	1
26				rumah bpk wiwin	2	2	2	2	1
27				rumah bpk mujiono	1	1	1	1	1
28	angin kencang	02/10/2020	Jl. Batanghari, Kelurahan Rampal Celaket, Kecamatan Klojen	bengkel	1	1	1	1	1
29	Tanah longsor	22/10/2020	Jl. Gunung Agung Utara II No.12 RT.03/RW,07, Kelurahan Pisangcandi, Kecamatan Sukun	rumah	1	1	1	2	1
30	pohon tumbang	31/10/2020	Perumahan Grand Mumtaz Residence, Jl. KH. Malik Dalam RT.04/RW.05, Kelurahan Buring, Kecamatan Kedungkandang	rumah	1	1	1	2	1

31	Tanah longsor	01/11/2020	Jl. S. Supriadi X No. 10 – 11 RT.18/RW.06, Kelurahan Bandungrejosari, Kecamatan Sukun	rumah	2	2	2	3	1
32	pohon tumbang	05/11/2020	Jl. Ir. H. Juanda No.06, Kelurahan Jodipan, Kecamatan Blimbing	warung	2	2	2	3	1
33	banjir	20/11/2020	Jl. Candi 3A RT.01/RW.03, Kelurahan Karangbesuki, Kecamatan Sukun	rumah	1	1	1	2	2
34	Tanah longsor	22/11/2020	Jembatan kedungkandang Jl. Mayjend Sungkono, Kelurahan Kedungkandang, Kecamatan Kedungkandang	plengsengan	3	3	3	3	2
35	Tanah longsor	24/11/2020	Jl. Titan Asri X RT.05 RW.12, Kelurahan Pandanwangi, Kecamatan Blimbing	plengsengan	3	3	3	3	1
36	Tanah longsor	29/11/2020	Jl. Danau Ranau, Kelurahan Sawojajar, Kecamatan Kedungkandang	plengsengan	3	3	3	3	2
37	Tanah longsor	29/11/2020	Jl. Muharto 5 RT.03 / RW.09, Kelurahan Kotalama, Kecamatan Kedungkandang	plengsengan	3	3	3	3	2



38	angin kencang	08/12/2020	§ Jl. Simpang Borobur Utara, Kelurahan Purwodadi, Kecamatan Blimbing	rumah	1	1	1	2	1
39			§ Jl. Simpang Sono Keling, Kelurahan Ciptomulyo, Kecamatan Sukun	warung	1	1	1	2	1
40			§ Jl. Sawojajar Gg VII RT.01/RW.02, Kelurahan Sawojajar, Kecamatan Kedungkandang	rumah	1	1	1	2	2
41			§Jl. Polowijen Gg.2 RT.01 RW.03, Kel. Polowijen Kec. Blimbing	mebel	1	1	1	2	1
42	Tanah longsor	08/12/2020	Jl. Panglima Sudirman No. 1, Kelurahan Kesatrian, Kecamatan Blimbing	kantor	3	3	3	3	2
43	Tanah longsor	12/12/2020	Jl. Terusan Kecubung Barat, Kelurahan Tlogomas, Kecamatan Lowokwaru	jembatan	3	3	3	3	1
44	Tanah longsor	12/12/2020	Jl. Muharto Gg.5C No.26 RT.03 RW.09, Kelurahan Kotalama, Kecamatan Kedungkandang	plengsengan	3	3	3	3	1

45	Tanah longsor	14/12/2020	Jl. Muharto Gg.5C No.26 RT.03 RW.09, Kelurahan Kotalama, Kecamatan Kedungkandang	rumah	2	2	2	3	2
46	Tanah longsor	13/12/2020	Jl.Tapak Siring IV RT.03/RW.08,Kelurahan Samaan, Kecamatan Klojen	rumah	2	2	2	3	2
47	Tanah longsor	13/12/2020	Jl. Panji Suroso No. 8 (SPBU Panji Suroso), Kelurahan Purwodadi, Kecamatan Blimbing	tembok	3	3	3	3	1
48	Tanah longsor	12/12/2020	Jl. Sisingamangaraja No. 267 RT.04 RW.13, Kelurahan Bunulrejo, Kecamatan Blimbing	plengsengan	3	3	3	3	1
49	banjir	25/12/2020	Jl. Bareng Gg. IIA RT.01/RW.07 Kelurahan Bareng, Klojen	rumah	2	2	1	2	2
50	Tanah longsor	25/12/2020	Jl. Pelabuhan Ketapang RT.10/RW.02, Kelurahan Bandungrejosari, Kecamatan Sukun	rumah	2	2	2	2	1