

**PREDIKSI ANGKA KEMISKINAN DI INDONESIA  
MENGGUNAKAN MODEL ARTIFICIAL  
NEURAL NETWORK**

**TESIS**

**OLEH :  
LIA ALFIA  
NIM. 210605220009**



**PROGRAM STUDI MAGISTER INFORMATIKA  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM  
MALANG  
2025**

**PREDIKSI ANGKA KEMISKINAN DI INDONESIA  
MENGGUNAKAN MODEL ARTIFICIAL  
NEURAL NETWORK**

**TESIS**

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

**OLEH :**  
**LIA ALFIA**  
**NIM. 210605220009**

**PROGRAM STUDI MAGISTER INFORMATIKA  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM  
MALANG  
2025**

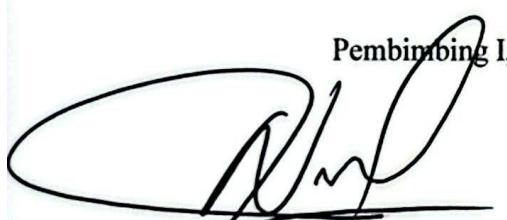
**PREDIKSI ANGKA KEMISKINAN DI INDONESIA  
MENGGUNAKAN MODEL ARTIFICIAL  
NEURAL NETWORK**

**TESIS**

**OLEH :**  
**LIA ALFIA**  
**NIM. 210605220009**

Telah diperiksa dan disetujui untuk diuji  
Tanggal: 6 November 2025

Pembimbing I,



Dr. Ir. Presy Nugroho, ST., MT, IPM., ASEAN Eng  
NIP 19710722 201101 1 001

Pembimbing II,



Dr. Ir. Yunifa Miftachul Arif, S.ST.,  
NIP. 19830616 201101 1 004

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



Prof. Dr. Ir. Muhammad Faisal, S.Kom., M.T.  
NIP. 19740510 200501 1 007

**PREDIKSI ANGKA KEMISKINAN DI INDONESIA MENGGUNAKAN  
MODEL ARTIFICIAL NEURAL NETWORK**

**THESIS**

Oleh:  
**LIA ALFIA**  
**NIM. 210605220009**

Telah Dipertahankan di Depan Dewan Pengaji Thesis  
Dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan  
Untuk Memperoleh Gelar Magister Komputer (M.Kom)  
Tanggal: 6 November 2025

**Susunan Dewan Pengaji**

**Tanda Tangan**

Pengaji I : Dr. Usman Pagalay, M.Si  
NIP. 19650414 200312 1 001

()

Pengaji II : Prof. Dr. M. Faisal, M.T  
NIP 19740510 200501 1 007

()

Pembimbing I : Dr. Ir. Fresy Nugroho, ST., MT, IPM., ASEAN Eng  
NIP 19710722 201101 1 001

()

Pembimbing II : Dr. Yunifa Miftachul Arif, M. T  
NIP. 19830616 201101 1 004

Mengetahui dan Mengesahkan  
Ketua Progra Studi Magister Informatika  
FAKULTAS SAINS DAN TEKNOLOGI  
Universitas Islam Negeri Maulana Malik Ibrahim Malang



Prof. Dr. M. Faisal, M.T  
NIP 19740510 200501 1 007

## **PERNYATAAN KEASLIAN TULISAN**

Saya yang bertanda tangan dibawah ini:

Nama : Lia Alfia  
NIM : 210605220009  
Program Studi : Magister Informatika  
Fakultas : Sains dan Teknologi

Menyatakan dengan sebenarnya bahwa Thesis yang saya tulis ini benar-benar merupakan hasil karya saya 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 di kemudian hari terbukti atau dapat dibuktikan Thesis ini hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 01 November 2025  
(Yang membuat pernyataan,  
  
Lia Alfia  
NIM. 210605220009

## **MOTTO**

*“Hidup yang tidak dipertaruhkan, tidak akan pernah dimenangkan”*

*~ Sutan Sjahrir~*

## HALAMAN PERSEMBAHAN

الْحَمْدُ لِلّٰهِ رَبِّ الْعَالَمِينَ

Puji syukur atas kehadirat Allah SWT  
Shalawat serta salam kepada Rasulullah SAW

Dengan segenap hati, penulis mempersembahkan sebuah karya ini untuk orang-orang yang paling kusayangi, kubanggakan dan selalu memberikan energi semangat, yaitu :

1. Untuk diriku sendiri, terima kasih atas ketekunan, kesabaran, dan keberanian untuk menuntaskan perjalanan ini. Semoga setiap usaha dan pengorbanan menjadi bagian dari diriku yang semakin kuat, bijaksana, dan penuh harapan, serta mampu menghadirkan kebahagiaan bagi orang-orang yang kucintai.
2. Kepada cahaya yang tak pernah padam di hidupku, yang menanamkan akar keteguhan dan bunga kebijaksanaan. Setiap doa, pelukan, dan pengorbanan kalian adalah benih yang menumbuhkan aku hingga bisa berdiri dan melangkah dengan percaya diri. Mama Masusi dan Bapak Imam As'adi semoga sehat, bahagia selalu serta dalam lindungan Allah, hidup lebih lama ya untuk Lia dan adik.
3. Untuk Firnas Maulanal Abshar, adikku tercinta, sahabat kecil yang selalu menghadirkan tawa dan semangat, terima kasih telah menjadi teman perjalanan yang tak tergantikan. Semoga kita selalu saling menguatkan, menapaki setiap hari dengan rasa syukur dan keceriaan.
4. Untuk kamu yang kelak akan berjalan bersamaku, semoga kamu merasa beruntung menemukan aku, bukan hanya pada hatiku, tetapi juga pada perjalanan dan setiap proses yang membentukku. Semoga kita bisa saling melengkapi, menumbuhkan satu sama lain, dan menapaki hari-hari dengan hangatnya kebersamaan.

Penulis ucapkan “*jazakumullah khairan katsiiraa*”. Semoga Allah SWT selalu melindungi dan menjaga mereka. Aamiin Ya Rabbal ‘Alamiin.

## KATA PENGANTAR

*Assalamu'alaikum Warahmatullahi Wabarakatuh*

Alhamdulillah, penulis memanjatkan puji dan syukur ke hadirat Allah SWT atas limpahan rahmat dan hidayah-Nya, sehingga penulis dapat menyelesaikan studi pada Program Studi Magister Informatika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang, serta menyelesaikan penyusunan tesis ini dengan baik. Selanjutnya, penulis menyampaikan ucapan terima kasih yang disertai doa dan harapan *Jazakumullah Ahsanal Jaza'* kepada seluruh pihak yang telah memberikan bantuan, dukungan, dan kontribusi dalam penyelesaian tesis ini. Penulis mengucapkan terima kasih kepada:

1. Dr. Ir Fresy Nugroho, ST., MT. IPM & Dr. Yunifa Miftachul Arif, MT selaku pembimbing tesis yang telah memberikan arahan, bimbingan, serta pengalaman berharga selama proses penyusunan tesis ini.
2. Segenap civitas akademika Program Studi Magister Informatika, khususnya seluruh dosen, atas ilmu pengetahuan, arahan, dan bimbingan yang telah diberikan kepada penulis selama masa studi.
3. Keluarga tercinta yang telah banyak memberikan doa dan dukungan moril hingga thesis ini dapat terselesaikan.
4. Teman-teman Magister Informatika angkatan 5 dan sahabat-sahabat saya yang selalu memberikan semangat dan juga motivasi.

Wassalamu'alaikum Wr. Wb .

Malang, 01 November 2025

Penulis

## DAFTAR ISI

|   |      |
|---|------|
| COVER .....   | i    |
| JUDUL .....   | ii   |
| HALAMAN PERSETUJUAN .....                                 | iii  |
| HALAMAN PENGESAHAN .....                                  | iv   |
| PERNYATAAN KEASLIAN TULISAN .....                         | v    |
| MOTTO .....   | vi   |
| HALAMAN PERSEMBAHAN .....                                 | vii  |
| KATA PENGANTAR .....                                      | viii |
| DAFTAR ISI .....  | ix   |
| DAFTAR TABEL .....  | xi   |
| DAFTAR GAMBAR .....                                       | xii  |
| ABSTRAK .....   | xiv  |
| ABSTRACT .....  | xv   |
| الملخص .....  | xvi  |
| BAB I .....   | 1    |
| PENDAHULUAN .....   | 1    |
| 1.1 Latar Belakang .....                                  | 1    |
| 1.2 Pernyataan Masalah .....                              | 6    |
| 1.3 Tujuan Penelitian .....                               | 7    |
| 1.4 Batasan Masalah .....                                 | 7    |
| 1.5 Manfaat Penelitian .....                              | 7    |
| BAB II .....  | 8    |
| STUDI PUSTAKA .....                                       | 8    |
| 2.1 Prediksi Angka Kemiskinan menggunakan model ANN ..... | 8    |
| 2.2 Kerangka Teoritis .....                               | 11   |
| BAB III .....   | 16   |
| METODOLOGI PENELITIAN .....                               | 16   |
| 3.1 Desain Penelitian .....                               | 16   |
| 3.2 Prosedur Penelitian .....                             | 17   |
| 3.2.1 Pengumpulan Data .....                              | 17   |
| 3.2.2 Rekayasa Data .....                                 | 20   |
| 3.2.3 Design Sistem .....                                 | 21   |
| 3.3 Instrumen Penelitian .....                            | 39   |
| BAB IV .....  | 41   |
| MODEL ANN SATU HIDDEN LAYER .....                         | 41   |
| 4.1 Design .....  | 41   |
| 4.2 Implementasi .....                                    | 45   |

|  |     |
|--|-----|
| 4.3 Uji Coba .....   | 46  |
| 4.3.1 Pelatihan BPN-1A.....  | 46  |
| 4.3.2 Pelatihan BPN-1B .....                                       | 48  |
| 4.3.3 Pelatihan BPN-1C .....                                       | 50  |
| 4.3.4 Pelatihan BPN-1D .....                                       | 52  |
| 4.3.5 Pelatihan BPN-1E .....                                       | 54  |
| 4.3.6 Pelatihan BPN-1F.....  | 56  |
| 4.3.7 Pelatihan BPN-1G.....  | 58  |
| 4.3.8 Pelatihan BPN-1H.....  | 60  |
| 4.3.9 Pelatihan BPN-1I .....                                       | 62  |
| 4.3.10 Pelatihan BPN-1J .....                                      | 64  |
| 4.4 Hasil Prediksi .....   | 66  |
| 4.5 Kesimpulan.....  | 67  |
| BAB V.....   | 69  |
| MODEL ANN DUA HIDDEN LAYER.....                                    | 69  |
| 5.1 Design.....  | 69  |
| 5.2 Implementasi .....   | 74  |
| 5.3 Uji Coba .....   | 74  |
| 5.3.1 Pelatihan BPN-2A.....  | 74  |
| 5.3.2 Pelatihan BPN-2B .....                                       | 76  |
| 5.3.3 Pelatihan BPN-2C .....                                       | 78  |
| 5.3.4 Pelatihan BPN-2D .....                                       | 80  |
| 5.3.5 Pelatihan BPN-2E .....                                       | 82  |
| 5.3.6 Pelatihan BPN-2F.....  | 84  |
| 5.3.7 Pelatihan BPN-2G.....  | 86  |
| 5.3.8 Pelatihan BPN-2H.....  | 88  |
| 5.3.9 Pelatihan BPN-2I .....                                       | 90  |
| 5.3.10 Pelatihan BPN-2J .....                                      | 92  |
| 5.4 Hasil Prediksi .....   | 94  |
| 5.5 Kesimpulan.....  | 95  |
| BAB VI .....   | 98  |
| PEMBAHASAN .....   | 98  |
| 5.1 Analisis Perbandingan Arsitektur .....                         | 98  |
| 5.2 Pandangan Islam terhadap Upaya Penanggulangan Kemiskinan ..... | 100 |
| BAB VII.....   | 103 |
| KESIMPULAN .....   | 103 |
| 7.1 Kesimpulan.....  | 103 |
| 7.2 Saran.....   | 104 |
| DAFTAR PUSTAKA .....   | 105 |

## DAFTAR TABEL

|  |    |
|--|----|
| Tabel 2. 1 Analisis Kinerja Model ANN dalam Memprediksi Angka Kemiskinan ..... | 13 |
| <br>   |    |
| Tabel 3. 1 Atribut Data Penelitian & Kodifikasi.....                           | 18 |
| Tabel 3. 2 Dataset Tingkat kemiskinan di Indonesia .....                       | 20 |
| Tabel 3. 3 Hasil Normalisasi Data .....  | 21 |
| Tabel 3. 4 Desain Sistem.....  | 22 |
| Tabel 3. 5 Detail Data Kemiskinan .....  | 22 |
| Tabel 3. 6 Identifikasi Standar Indikator Kemiskinan .....                     | 24 |
| Tabel 3. 7 Konfigurasi Arsitektur ANN (Hidden Layer Node) .....                | 27 |
| Tabel 3. 8 Ekstrasi Fitus .....  | 36 |
| Tabel 3. 9 Atribut Data Variable Target .....                                  | 38 |
| Tabel 3. 10 Banyaknya Node Pada Masing-Masing Hidden Layer.....                | 38 |
| Tabel 3. 11 Variasi Input pada Model yang dilatih.....                         | 39 |
| Tabel 3. 12 Instrumen pada Penelitian .....                                    | 39 |
| <br>   |    |
| Tabel 4. 1 Indikator/Parameter/Atribut Model.....                              | 45 |
| Tabel 4. 2 Tahap Pelatihan BPN-1A.....   | 46 |
| Tabel 4. 3 Tahap Pelatihan BPN-1B .....  | 48 |
| Tabel 4. 4 Proses Pelatihan BPN-1C.....  | 50 |
| Tabel 4. 5 Proses Pelatihan BPN-1D .....                                       | 52 |
| Tabel 4. 6 Proses Pelatihan BPN-1E.....  | 54 |
| Tabel 4. 7 Proses Pelatihan BPN-1F .....                                       | 56 |
| Tabel 4. 8 Proses Pelatihan BPN-1G .....                                       | 58 |
| Tabel 4. 9 Proses Pelatihan BPN-1H .....                                       | 60 |
| Tabel 4. 10 Proses Pelatihan BPN-1I .....                                      | 62 |
| Tabel 4. 11 Proses Pelatihan BPN-1J .....                                      | 64 |
| Tabel 4. 12 Hasil Prediksi Model BPN 1 Hidden Layer.....                       | 66 |
| <br>   |    |
| Tabel 5. 1 Parameter Model .....   | 74 |
| Tabel 5. 2 Proses Uji BPN-2A .....   | 74 |
| Tabel 5. 3 Proses Uji BPN-2B .....   | 76 |
| Tabel 5. 4 Proses Uji BPN-2C .....   | 79 |
| Tabel 5. 5 Proses Uji BPN-2D .....   | 81 |
| Tabel 5. 6 Proses Uji BPN-2E.....  | 83 |
| Tabel 5. 7 Proses Uji BPN-2F .....   | 85 |
| Tabel 5. 8 Proses Uji BPN-2G .....   | 87 |
| Tabel 5. 9 Proses Uji BPN-2H .....   | 89 |
| Tabel 5. 10 Proses Uji BPN-2I.....   | 91 |
| Tabel 5. 11 Proses Uji BPN-2J .....  | 93 |
| Tabel 5. 12 Hasil Prediksi Model BPN 2 Hidden Layer.....                       | 95 |

## DAFTAR GAMBAR

|  |    |
|--|----|
| Gambar 1. 1 Perkembangan Jumlah dan Persentase Penduduk Miskin di Indonesia, Maret 2022–Maret 2025 ..... | 2  |
| Gambar 2. 1 Kerangka Teoritis Prediksi Angka Kemiskinan.....   | 12 |
| Gambar 3. 1 Tahapan Penelitian .....   | 17 |
| Gambar 3. 2 Grafik Perbandingan MAPE pada Berbagai Arsitektur ANN .....                                  | 27 |
| Gambar 3. 3 Arsitektur Jaringan Pertama Model ANN .....  | 28 |
| Gambar 3. 4 Arsitektur Jaringan Kedua Model ANN.....   | 29 |
| Gambar 4. 1 Flowchart Proses Pelatihan .....   | 42 |
| Gambar 4. 2 Pelatihan BPN-1A .....   | 47 |
| Gambar 4. 3 Perbandingan Data Aktual dan Prediksi BPN-1A.....  | 47 |
| Gambar 4. 4 Skor Loss Pelatihan BPN-1B .....   | 49 |
| Gambar 4. 5 Perbandingan Data Aktual dan Prediksi BPN-1B.....  | 49 |
| Gambar 4. 6 Nilai Loss Pelatihan BPN-1C.....   | 51 |
| Gambar 4. 7 Perbandingan Data Aktual dan Prediksi BPN-1C.....  | 51 |
| Gambar 4. 8 Nilai Loss Pelatihan BPN-1D .....  | 53 |
| Gambar 4. 9 Perbandingan Data Aktual dan Prediksi BPN-1D.....  | 53 |
| Gambar 4. 10 Nilai Loss Pelatihan BPN-1E.....  | 55 |
| Gambar 4. 11 Perbandingan Data Aktual dan Prediksi BPN-1E .....  | 55 |
| Gambar 4. 12 Nilai Loss Pelatihan BPN-1F .....   | 57 |
| Gambar 4. 13 Perbandingan Data Aktual dan Prediksi BPN-1F .....  | 57 |
| Gambar 4. 14 Nilai Loss Pelatihan BPN-1G .....   | 59 |
| Gambar 4. 15 Perbandingan Data Aktual dan Prediksi BPN-1G.....   | 59 |
| Gambar 4. 16 Nilai Loss Pelatihan BPN-1H .....   | 61 |
| Gambar 4. 17 Perbandingan Data Aktual dan Prediksi BPN-1H.....   | 61 |
| Gambar 4. 18 Nilai Loss Pelatihan BPN-1I.....  | 63 |
| Gambar 4. 19 Perbandingan Data Aktual dan Prediksi BPN-1I .....  | 63 |
| Gambar 4. 20 Nilai Loss Pelatihan BPN-1J.....  | 65 |
| Gambar 4. 21 Perbandingan Data Aktual dan Prediksi BPN-1A.....   | 65 |
| Gambar 4. 22 Perbandingan performa model arsitektur 1 hidden layer terbaik ...                           | 67 |
| Gambar 5. 1 Rancangan Arsitektur Model ANN Dengan Lapisan Tersembunyi Pertama.....                       | 41 |
| Gambar 5. 2 Rancangan Arsitektur Model ANN Dengan Lapisan Tersembunyi Kedua .....                        | 69 |
| Gambar 5. 3 Nilai Loss Pelatihan BPN-2A .....  | 75 |
| Gambar 5. 4 Perbandingan Data Aktual dan Prediksi BPN-2A.....  | 76 |
| Gambar 5. 5 Nilai Loss Pelatihan BPN-2B.....   | 77 |
| Gambar 5. 6 Perbandingan Data Aktual dan Prediksi BPN-2B.....  | 78 |
| Gambar 5. 7 Nilai Loss Pelatihan BPN-2C.....   | 79 |
| Gambar 5. 8 Perbandingan Data Aktual dan Prediksi BPN-2C.....  | 80 |
| Gambar 5. 9 Nilai Loss Pelatihan BPN-2D .....  | 81 |
| Gambar 5. 10 Perbandingan Data Aktual dan Prediksi BPN-2D.....   | 82 |
| Gambar 5. 11 Nilai Loss Pelatihan BPN-2E .....   | 83 |
| Gambar 5. 12 Perbandingan Data Aktual dan Prediksi BPN-2E .....  | 84 |
| Gambar 5. 13 Nilai Loss Pelatihan BPN-2F .....   | 85 |

|  |    |
|--|----|
| Gambar 5. 14 Perbandingan Data Aktual dan Prediksi BPN-2F .....                | 86 |
| Gambar 5. 15 Nilai Loss Pelatihan BPN-2G .....                                 | 87 |
| Gambar 5. 16 Perbandingan Data Aktual dan Prediksi BPN-2G.....                 | 88 |
| Gambar 5. 17 Nilai Loss Pelatihan BPN-2H .....                                 | 89 |
| Gambar 5. 18 Perbandingan Data Aktual dan Prediksi BPN-2H.....                 | 90 |
| Gambar 5. 19 Nilai Loss Pelatihan BPN-2I .....                                 | 91 |
| Gambar 5. 20 Perbandingan Data Aktual dan Prediksi BPN-2I .....                | 92 |
| Gambar 5. 21 Nilai Loss Pelatihan BPN-2J.....                                  | 93 |
| Gambar 5. 22 Perbandingan Data Aktual dan Prediksi BPN-2J.....                 | 94 |
| Gambar 5. 23 Perbandingan performa model arsitektur 2 hidden layer terbaik ... | 96 |

## ABSTRAK

Alfia, Lia. 2025. Prediksi Angka Kemiskinan Menggunakan Model Artificial Neural Network. Program Studi Magister Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Dr. Ir Fresy Nugroho, ST., MT. IPM. Pembimbing (II): Dr. Yunifa Miftachul Arif , MT

**Kata kunci:** Kemiskinan, ANN, Propagasi mundur, Perkiraan.

Kemiskinan merupakan permasalahan sosial ekonomi yang kompleks dan menjadi tantangan utama pembangunan di Indonesia. Penelitian ini bertujuan untuk menganalisis faktor-faktor sosial ekonomi yang memengaruhi angka kemiskinan serta mengevaluasi kemampuan model *Artificial Neural Network* (ANN) dalam memprediksi tingkat kemiskinan di Indonesia. Data yang digunakan merupakan data sekunder yang diperoleh dari situs Kaggle dengan tiga belas indikator sosial ekonomi sebagai variabel input. Model yang digunakan adalah *Backpropagation Neural Network* (BPN) dengan variasi arsitektur 1 dan 2 *hidden layer*. Hasil pengujian menunjukkan bahwa arsitektur 13-6-1 (BPN-1D) memiliki performa terbaik dengan nilai Mean Squared Error (MSE) sebesar 4,4 dan Mean Absolute Percentage Error (MAPE) sebesar 2,95%. Hal ini membuktikan bahwa ANN mampu memberikan hasil prediksi yang akurat terhadap angka kemiskinan di Indonesia. Dengan demikian, penerapan model ini dapat menjadi alternatif alat bantu bagi pemerintah dalam perencanaan kebijakan pengentasan kemiskinan secara lebih tepat sasaran.

## ***ABSTRACT***

*Alfia, Lia. 2025. Prediction of Poverty Rate Using Artificial Neural Network Model. Master's Program in Informatics, Faculty of Science and Technology, State Islamic University of Maulana Malik Ibrahim Malang. Supervisor (I): Dr. Ir. Fresy Nugroho, ST., MT., IPM. Supervisor (II): Dr. Yunifa Miftachul Arif, MT.*

**Keywords:** *Poverty, ANN, Backpropagationn, Prediction.*

*Poverty is a complex socio-economic problem and remains a major challenge for development in Indonesia. This study aims to analyze the socio-economic factors affecting poverty rates and evaluate the performance of the Artificial Neural Network (ANN) model in predicting poverty levels in Indonesia. The data used are secondary data obtained from Kaggle, consisting of thirteen socio-economic indicators as input variables. The Backpropagation Neural Network (BPN) model was applied with architectures containing one and two hidden layers. The results show that the 13-6-1 architecture (BPN-1D) achieved the best performance with a Mean Squared Error (MSE) of 4.4 and a Mean Absolute Percentage Error (MAPE) of 2.95%. This finding indicates that ANN can provide accurate predictions of poverty rates in Indonesia. Therefore, the implementation of this model can serve as a useful tool for policymakers to design more targeted poverty reduction strategies.*

## الملخص

أفيا، لـيا. 2025. التنبؤ بمعدل الفقر باستخدام نموذج الشبكة العصبية الاصطناعية. برنامج الماجستير في المعلوماتية، كلية العلوم والتكنولوجيا، جامعة مولانا مالك إبراهيم الإسلامية الحكومية مالانغ. المشرف MT ،المشرف (الثاني): د. يونيفا مفتاح العارف. ST ، MT ، IPM . (الأول): د. إير. فريسي نوغر وهو

**الكلمات المفتاحية:** الفقر، الشبكة العصبية الاصطناعية، الانتشار العكسي، التنبؤ

يُعد الفقر مشكلة اجتماعية-اقتصادية معقدة ولا يزال يشكل تحدياً رئيسياً للتنمية في إندونيسيا. تهدف هذه الدراسة إلى تحليل العوامل الاجتماعية والاقتصادية المؤثرة في معدلات الفقر، وتقدير أداء نموذج الشبكة في التنبؤ بمستويات الفقر في إندونيسيا. تعتمد البيانات المستخدمة على بيانات (ANN) العصبية الاصطناعية وتتكون من ثلاثة عشر مؤشراً اجتماعياً واقتصادياً بوصفها Kaggle، ثانوية تم الحصول عليها من منصة – Backpropagation Neural Network (BPN) متغيرات إدخال. تم تطبيق نموذج شبكة الانتشار العكسي باستخدام معماريات تحتوي على طبقة خفية واحدة وطبقتين خفيتين. أظهرت النتائج أن المعمارية (BPN) بلغت 4.4، ومتوسط نسبة (MSE) حققت أفضل أداء بقيمة متوسط مربع الخطأ (BPN-1D) 13-6-1 بنسبة 2.95%. وتشير هذه النتائج إلى أن الشبكات العصبية الاصطناعية قادرة (MAPE) الخطأ المطلق على تقديم تنبؤات دقيقة لمعدلات الفقر في إندونيسيا. وبناءً على ذلك، يمكن أن يُسهم تطبيق هذا النموذج كأدلة مفيدة لصناعة السياسات في تصميم استراتيجيات أكثر استهدافاً للحد من الفقر.

## **BAB I**

### **PENDAHULUAN**

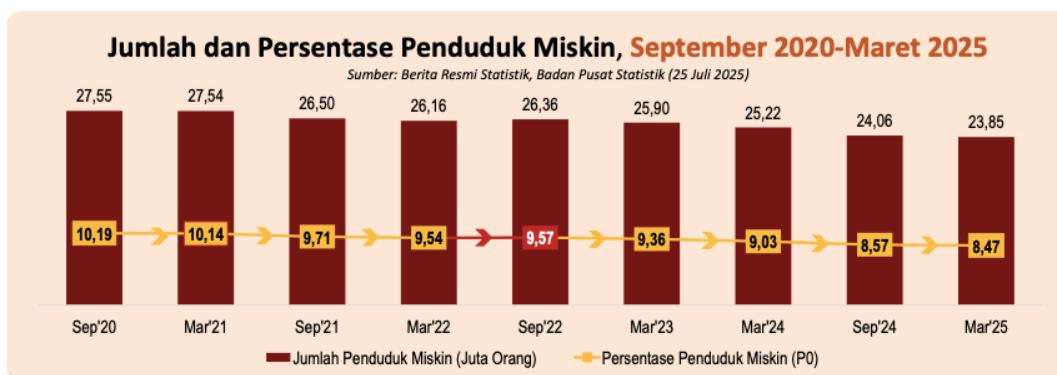
#### **1.1 Latar Belakang**

Isu-isu utama yang menjadi tantangan bagi negara-negara berkembang mencakup berbagai aspek, salah satunya adalah kemiskinan. Kemiskinan merupakan bentuk dari fenomena sosial yang bersifat kompleks serta multidimensional, hal ini tidak hanya menggambarkan adanya keterbatasan ekonomi, akan tetapi juga tentang adanya ketidakadilan dan kesenjangan sosial yang mana bisa dilihat dari minimnya akses terhadap kesehatan, pendidikan, serta peluang untuk memiliki kehidupan yang layak (Kurniawan & Apriani, 2025). Permasalahan ini menjadi isu global yang terus mendapat perhatian karena memberikan dampak signifikan terhadap kesejahteraan masyarakat dan keberlanjutan pembangunan suatu negara. Lembaga Internasional dan pemerintah di seluruh dunia, termasuk Indonesia terus berupaya menekan angka kemiskinan melalui kebijakan ekonomi, program sosial, maupun pemberdayaan masyarakat.

Dalam hal ini, Indonesia sebagai negara berkembang terus menghadapi tantangan untuk mencapai pemerataan pembangunan dan kesejahteraan masyarakat. Menurut (Purwanti, 2024) kemiskinan merupakan tantangan dan permasalahan yang menggambarkan keterbatasan seseorang dalam memenuhi keperluan dasar hidupnya sehari-hari. Hal ini adalah tantangan serius yang telah menjadi fokus perhatian negara selama bertahun-tahun. Meskipun pertumbuhan ekonomi nasional menunjukkan trend positif dalam beberapa dekade terakhir, namun tidak semua lapisan masyarakat merasakan dampak yang merata. Kemiskinan yang merupakan bagian dalam kesenjangan sosial dan ekonomi tetap menjadi perhatian utama dalam agenda pembangunan nasional, masalah ini menjadi krusial karena berkaitan langsung dengan kualitas hidup masyarakat dan stabilitas sosial di berbagai wilayah Indonesia. (Vita Ferezagia, 2018)

Permasalahan kemiskinan yang terjadi di Indonesia masih menjadi isu yang menuntut perhatian serius, sebagaimana tertuang dalam UUD 1945 Pasal 34 dan diperkuat melalui Undang-Undang Nomor 13 Tahun 2011 tentang Penanganan

Fakir Miskin, yang menekankan pentingnya penanganan terstruktur, sistematis, dan menyeluruh terhadap masyarakat miskin (Oktariani, 2024). Berdasarkan data Badan Pusat Statistik (BPS), jumlah penduduk miskin di Indonesia pada Maret 2023 tercatat sebesar 25,90 juta orang, menurun dari 26,36 juta orang pada Maret 2022. Meskipun demikian, persentase penduduk miskin masih tergolong tinggi, yaitu sebesar 9,36 persen (Purwanti, 2024). Data terbaru dari Badan Pusat Statistik (BPS) menunjukkan bahwa pada Maret 2025 jumlah penduduk miskin di Indonesia tercatat sebesar 23,85 juta orang, turun 0,21 juta orang dari September 2024, sementara persentase pada Maret 2025 sebesar 8,47 persen, yang artinya mengalami penurunan 0,10 persen poin dari September 2024. Perbandingan tren dari tahun ke tahun dapat dilihat secara lebih jelas pada Gambar 1.1.



**Gambar 1. 1 Perkembangan Jumlah dan Persentase Penduduk Miskin di Indonesia, Maret 2022–Maret 2025**  
(Sumber: Badan Pusat Statistik)

Berdasarkan Gambar 1.1 tersebut dapat diketahui bahwa penurunan angka kemiskinan di Indonesia memang menunjukkan adanya kemajuan, akan tetapi jumlah penduduk miskin yang masih mencapai puluhan juta jiwa menandakan bahwa persoalan ini masih tergolong tinggi dan belum terselesaikan. Oleh karena itu, persoalan kemiskinan tetap memerlukan perhatian serius, mengingat dampaknya yang luas terhadap kualitas hidup masyarakat dan keberlanjutan pembangunan. Kemiskinan juga dipengaruhi oleh berbagai faktor sosial ekonomi. Hal ini menjadi penting untuk mengetahui faktor-faktor yang dapat berpengaruh

terhadap naik-turunnya angka kemiskinan. Misalnya, seperti yang disebutkan oleh (Priseptian & Primandhana, 2022) diantara beberapa faktornya yaitu upah minimum provinsi, Indeks pembangunan manusia (IPM), Pertumbuhan ekonomi, dan tingkat pengangguran. Faktor-faktor tersebut menunjukkan bahwa kemiskinan sangat dipengaruhi oleh berbagai aspek sosial-ekonomi.

Selanjutnya, selain ditinjau dari aspek sosial ekonomi, kemiskinan juga mendapat perhatian dalam perspektif agama, khususnya islam. Hal ini sejalan dengan pandangan Islam yang juga menekankan fokus terhadap kesejahteraan kelompok masyarakat miskin. Pandangan tersebut tercermin dalam ketentuan zakat yang diatur dalam Al-Qur'an. Dijelaskan bahwa orang miskin termasuk golongan yang berhak menerima zakat, yang menunjukkan bahwa Islam memberikan perhatian terhadap kesejahteraan dan keadilan sosial. Hal ini tercantum dalam Al-Qur'an surah At-Taubah ayat 60 sebagai berikut:

وَفِي الْغَرِيمِينَ الرَّقَابِ وَفِي قُوْبَهُمْ وَالْمُؤَنَّفَةِ عَلَيْهَا وَالْعَمِيلِينَ وَالْمَسِكِينِ لِلْفُقَرَاءِ الصَّدَقَةُ إِنَّمَا  
حِكْمَةُ عَلِيهِمْ وَاللهُ أَعْلَمُ مَنْ فَرِيَضَهُ السَّبِيلُ وَابْنُ اللهِ سَبِيلٌ

Artinya:

*“Sesungguhnya zakat itu hanyalah untuk orang-orang fakir, orang-orang miskin, para amil zakat, orang-orang yang dilunakkan hatinya (mualaf), untuk (memerdekaan) para hamba sahaya, untuk (membebaskan) orang-orang yang berutang, untuk jalan Allah dan untuk orang-orang yang sedang dalam perjalanan (yang memerlukan pertolongan), sebagai kewajiban dari Allah. Allah Maha Mengetahui lagi Mahabijaksana.”*

Ayat tersebut menegaskan bahwa kemiskinan bukan hanya urusan pribadi, tetapi juga tanggung jawab bersama dalam masyarakat. Dengan demikian, kemiskinan bukan hanya permasalahan ekonomi semata, melainkan tantangan pembangunan yang harus ditanggulangi secara serius demi tercapainya kesejahteraan masyarakat.

Penanggulangan kemiskinan menjadi bagian penting dari cita-cita bangsa menuju masyarakat yang adil dan sejahtera. Apabila tidak segera diatasi, kondisi kemiskinan yang terus berlangsung dapat menghambat laju pembangunan nasional (Sinurat, 2023). Salah satu hal yang sangat penting dalam upaya penanggulangan

kemiskinan adalah ketersediaan data yang akurat (Wulandari et al., 2022). Data tersebut dapat menjadi dasar bagi pemerintah dalam mengambil keputusan yang tepat, sekaligus menjadi bahan evaluasi dengan membandingkan kondisi pada tahun-tahun sebelumnya. Data ini tidak hanya membantu pemerintah dalam menentukan langkah strategis, tetapi juga memberikan gambaran perkembangan dari waktu ke waktu sebagai bahan evaluasi.

Upaya penanggulangan kemiskinan dapat dilakukan dengan memprediksi angka kemiskinan di Indonesia pada tahun-tahun berikutnya. Hasil prediksi tersebut dapat menjadi bahan pertimbangan bagi pemerintah daerah dalam merumuskan kebijakan yang lebih tepat. Melalui prediksi, diharapkan dampak kemiskinan yang mungkin terjadi di masa depan dapat diminimalkan. Prediksi merupakan proses memperkirakan kejadian yang akan datang berdasarkan informasi dari kondisi masa lalu dan masa sekarang. (Sinaga & Sipayung, 2024)

Ada beberapa metode dan model yang bisa dipakai dalam melakukan suatu prediksi pada angka kemiskinan, salah satunya adalah *Artificial Neural Network* (ANN). Metode ANN merupakan sistem pengolahan informasi yang meniru cara kerja jaringan saraf pada otak manusia. Sistem ini belajar dengan menyesuaikan bobot-bobotnya, sehingga ANN mampu mengenali pola berdasarkan data yang telah ada sebelumnya (Kumalasari, 2025). Keunggulan ini menjadikan ANN sebagai metode yang potensial dalam memprediksi angka kemiskinan, terutama karena indikator sosial ekonomi yang memengaruhi kemiskinan seringkali memiliki hubungan yang tidak linear. ANN bekerja dengan membentuk jaringan lapisan neuron yang saling terhubung dan belajar dari data historis untuk menyesuaikan bobot antar variabel, sehingga mampu mengenali pola hubungan yang tidak tampak secara langsung dalam data, kemampuan ANN untuk melakukan generalisasi dari data latih membuatnya efektif dalam menangani variabel yang saling memengaruhi secara tidak sederhana (Putri et al., 2024). Dengan demikian, ANN menjadi pendekatan yang relevan untuk mendukung analisis prediktif tentang kemiskinan yang dipengaruhi oleh banyak faktor.

Beberapa penelitian terdahulu telah membuktikan bahwa ANN merupakan metode yang efektif dalam memprediksi angka kemiskinan. Penelitian oleh (Putri

et al., 2024) yang menggunakan model ANN untuk memperkirakan jumlah penduduk miskin di Indonesia dengan struktur jaringan 7-7-2 dan *Mean Absolute Percentage Error* (MAPE) sebesar 0,076%, penelitian tersebut berhasil menunjukkan bahwa ANN cukup unggul digunakan dalam konteks prediksi kemiskinan. Penelitian lain oleh (Fahrial, 2024) menerapkan algoritma *Backpropagation Conjugate Gradient Polak Ribiere* pada data kabupaten/kota di Provinsi Sumatera Barat dan memperoleh akurasi sebesar 90% dengan nilai MSE 0,0003s. Berdasarkan penelitian tersebut, dapat diketahui bahwa ANN merupakan metode prediktif yang kuat dalam konteks sosial ekonomi, khususnya dalam menganalisis dan memproyeksikan angka kemiskinan.

Pada Penelitian oleh (Syaharuddin et al., 2020) diperlihatkan bahwa model ANN mampu memperkirakan tingkat kemiskinan di Indonesia dengan tingkat akurasi yang sangat tinggi, yakni mencapai 99,66% dengan memanfaatkan data time series untuk membangun model prediksi. Namun, masih memiliki keterbatasan dari sisi validasi dataset, sehingga hasil yang diperoleh belum sepenuhnya dapat digeneralisasikan untuk kondisi yang lebih luas. Berdasarkan hal tersebut, penggunaan dataset *cross-section* dengan indikator yang lebih beragam menjadi pembeda utama pada penelitian ini, sehingga memberikan potensi hasil prediksi yang lebih optimal. Dataset yang lebih kaya ini kemudian diolah menggunakan Artificial Neural Network (ANN), dengan tujuan meningkatkan akurasi prediksi serta menghasilkan temuan yang lebih relevan dan dapat digeneralisasikan.

Pemilihan model ANN didasarkan pada kemampuannya dalam menangani permasalahan non-linear yang kompleks serta kemampuannya dalam mempelajari pola tersembunyi dari data yang beragam (Jamil & Pulukadang, 2025). ANN memiliki arsitektur yang menyerupai cara kerja jaringan saraf manusia, sehingga mampu melakukan generalization terhadap data baru dan menghasilkan prediksi yang lebih presisi dibandingkan metode tradisional. Selain itu, ANN fleksibel untuk diadaptasikan dengan berbagai jenis atribut sosial-ekonomi, baik yang bersifat kuantitatif maupun kualitatif, sehingga sangat relevan untuk digunakan dalam memodelkan fenomena kompleks seperti kemiskinan. Dengan karakteristik

tersebut, penggunaan ANN dalam penelitian ini dipandang tepat untuk menjawab kebutuhan prediksi yang akurat, adaptif, dan mampu mencerminkan dinamika nyata di lapangan.

Penelitian ini dilakukan untuk mengembangkan model prediksi kemiskinan menggunakan ANN, menggunakan data seluruh provinsi di Indonesia dari Badan Pusat Statistik (BPS) untuk periode 2015–2023, dengan melibatkan 13 indikator sosial-ekonomi, yaitu: 1) Provinsi, 2) Kode, 3) Tahun, 4) Persentase Penduduk Miskin (P0) Menurut Kabupaten/Kota, 5) Rata-rata Lama Sekolah Penduduk, 6) Pengeluaran per Kapita, 7) Indeks Pembangunan Manusia, 8) Umur Harapan Hidup, 9) Persentase rumah tangga yang memiliki akses terhadap sanitasi layak, 10) Persentase rumah tangga yang memiliki akses terhadap air minum layak, 11) Tingkat Pengangguran Terbuka, 12) Tingkat Partisipasi Angkatan Kerja, dan 13) PDRB atas Dasar Harga Konstan menurut Pengeluaran.

Berdasarkan uraian tersebut, penelitian ini dilakukan untuk mengembangkan model prediksi angka kemiskinan di Indonesia dengan menggunakan *Artificial Neural Network* (ANN). Oleh karena itu, peneliti mengangkat judul penelitian tentang “**Prediksi Angka Kemiskinan di Indonesia Menggunakan Model *Artficial Neural Network***”. Penelitian ini diharapkan dapat memberikan kontribusi ilmiah sekaligus menjadi masukan bagi kebijakan penanggulangan kemiskinan di masa mendatang.

## 1.2 Pernyataan Masalah

Adapun pernyataan masalah dalam penelitian ini, diantaranya:

1. Bagaimana pengaruh atribut data terhadap prediksi angka kemiskinan di Indonesia menggunakan model *Artificial Neural Network*?
2. Bagaimana tingkat akurasi model *Artificial Neural Network* dalam memprediksi angka kemiskinan di Indonesia?

### **1.3 Tujuan Penelitian**

Tujuan dari penelitian ini adalah:

1. Menganalisis pengaruh atribut data terhadap prediksi angka kemiskinan di Indonesia berdasarkan model *Artificial Neural Network*.
2. Mengevaluasi tingkat akurasi model *Artificial Neural Network* dalam memprediksi angka kemiskinan di Indonesia.

### **1.4 Batasan Masalah**

Batasan masalah dalam penelitian ini, diantaranya:

1. Penelitian ini hanya berfokus pada prediksi angka kemiskinan di Indonesia.
2. Dataset penelitian yang digunakan merupakan data sekunder yang sumber datanya di unduh melalui website resmi [bps.go.id](http://bps.go.id) dengan cakupan data seluruh provinsi di Indonesia.
3. Atribut data atau variable bebas yang digunakan terbatas pada tiga belas (13) indikator sosial ekonomi.
4. Model prediksi yang dikembangkan dibatasi pada penggunaan algoritma *backpropagation* dalam ANN.

### **1.5 Manfaat Penelitian**

Penelitian ini diharapkan dapat membantu pemerintah dalam memprediksi angka kemiskinan sebagai dasar pengambilan kebijakan penanggulangan kemiskinan. Selain itu, penelitian ini juga memberikan kontribusi dalam penerapan ANN sebagai metode prediksi pada permasalahan sosial ekonomi di Indonesia.

## **BAB II**

### **STUDI PUSTAKA**

#### **2.1 Prediksi Angka Kemiskinan menggunakan model ANN**

Metode ANN telah banyak dipakai dalam melakukan prediksi permasalahan sosial ekonomi, termasuk kemiskinan. Dalam penelitian Purbowati et al (2023) memakai model *Backpropagation Neural Network* (BPNN) dalam memprediksi jumlah penduduk miskin di Sumatera Utara. Data yang digunakan merupakan data sekunder dari BPS Provinsi Sumatera Utara periode 2017–2020, dengan total 25 kabupaten/kota, data tersebut terdiri atas 20 data yang digunakan sebagai data latih dan 5 data sebagai data uji. Temuan dari penelitian ini diketahui bahwasanya model Backpropagation bisa memprediksi jumlah penduduk miskin di Sumatera Utara pada tahun 2022 sebesar 77.927 jiwa. Penelitian ini membuktikan bahwa jaringan syaraf tiruan dapat diimplementasikan secara efektif dalam peramalan angka kemiskinan di tingkat provinsi.

Penelitian serupa juga dilakukan oleh (Syaharuddin et al., 2020) untuk melakukan prediksi jumlah penduduk miskin di Indonesia menggunakan metode *Backpropagation Neural Network* (BPNN). Data yang digunakan adalah jumlah penduduk miskin periode 2012–2019 pada 33 provinsi dengan pengujian menggunakan parameter akurasi MSE dan MAPE. Hasil penelitian menunjukkan arsitektur jaringan terbaik terdiri dari 264 neuron input, dua hidden layer masing-masing 20 dan 5 neuron, serta satu neuron output dengan fungsi aktivasi logsig. Model tersebut menghasilkan prediksi jumlah penduduk miskin tahun 2020 sebesar 332.005 jiwa, dengan nilai MSE 0,119, MAPE 2,298, dan akurasi mencapai 99,66%. Temuan ini memperlihatkan bahwa ANN *Backpropagation* mampu memberikan hasil prediksi yang sangat tepat dalam konteks kemiskinan di Indonesia.

Hal ini juga selaras dengan penelitian dari (Descania, 2022) tentang pemanfaatan Artificial Neural Network (ANN) untuk memprediksi tingkat pertumbuhan penduduk di Kecamatan Baregbeg, Kabupaten Ciamis. Penelitian ini menggunakan data pertumbuhan penduduk yang diolah dengan perangkat Matlab

untuk menghasilkan model prediksi. Hasil pemodelan menunjukkan bahwa ANN mampu memberikan prediksi dengan baik, ditunjukkan oleh nilai regresi sebesar 0,978 dan korelasi sebesar 0,993. Hal ini menandakan bahwa model ANN memiliki kemampuan yang tinggi dalam mengenali pola data pertumbuhan penduduk. Penulis juga menyarankan agar penelitian selanjutnya dapat membandingkan ANN dengan algoritma lain untuk memperoleh hasil prediksi yang lebih komprehensif.

Penelitian selanjutnya oleh (Oloyede et al., 2024) pemodelan kemiskinan di Nigeria dengan menggunakan *Artificial Neural Network regression modelling* (*ANN Regression*). Fokus penelitian ini adalah menganalisis kontribusi bobot setiap variabel ekonomi terhadap pembentukan *Multidimensional Poverty Index* (MPI). Hasil penelitian menunjukkan bahwa ketahanan pangan memiliki pengaruh relatif tertinggi dengan magnitude 0,085, sedangkan sanitasi memiliki pengaruh terendah dengan magnitude 0,045 terhadap MPI. Penelitian ini juga mengisi kesenjangan yang ada, karena sebagian besar literatur sebelumnya hanya mendeskripsikan arsitektur ANN tanpa menafsirkan bobot dan bias dalam model. Dengan demikian, studi ini memberikan pemahaman yang lebih mendalam mengenai bagaimana variabel-variabel ekonomi berkontribusi terhadap kemiskinan multidimensioal melalui pendekatan ANN.

Dalam penelitian (Vasquez et al., 2024) terkait kemiskinan petani jagung skala kecil di Meksiko dengan menggunakan metode *Feed-Forward Backpropagation Artificial Neural Network* atau BPNN. Model ini digunakan untuk memperkirakan variabel yang paling berpengaruh terhadap efisiensi ekonomi, efisiensi energi, dan poverty coverage line. Hasil penelitian menunjukkan bahwa variabel yang paling berpengaruh adalah luas lahan yang digarap, energi yang dikonsumsi per hektar, serta tingkat penanaman, dengan nilai koefisien korelasi  $R = 0,86$ . ANN berhasil memprediksi bahwa faktor utama penyebab garis kemiskinan di sektor pertanian adalah besarnya lahan yang ditanami dan energi yang diinvestasikan petani. Dari penelitian ini diketahui bahwa ANN dapat dijadikan alat bantu pengambilan keputusan dalam pengelolaan sumber daya pertanian, meskipun interpretasi hasilnya masih perlu disempurnakan dan dikembangkan dengan variabel tambahan seperti iklim dan indikator sosial ekonomi lainnya.

Kemudian (Putri et al., 2024) juga melakukan penelitian tentang peramalan tingkat kemiskinan di Indonesia dengan menggunakan metode *Backpropagation Neural Network* (BPNN). Data yang digunakan adalah data tahunan kemiskinan Indonesia periode 2017–2021 yang dipisahkan menjadi data pelatihan dan data pengujian. Temuan penelitian yaitu arsitektur jaringan terbaik adalah BP (7,7,2) dengan 7 neuron pada input layer, 7 neuron pada hidden layer, dan 2 neuron pada output layer. Model ini menghasilkan tingkat akurasi sebesar 89,65% dengan nilai MAPE 10,35, yang menunjukkan ANN mampu memberikan prediksi dengan error relatif rendah. Penelitian ini membuktikan bahwa ANN dapat digunakan sebagai acuan dalam memproyeksikan jumlah penduduk miskin di masa depan serta mendukung perencanaan kebijakan pemerintah.

Penelitian dari (Poerwanto & Fajriani, 2020) yang dilakukan untuk memprediksi tingkat kemiskinan di Sulawesi Selatan dengan menggunakan metode *Resilient Backpropagation Neural Network* (RBNN). Data yang digunakan mencakup periode 2010–2014 dengan variabel prediktor Indeks Pembangunan Manusia (IPM), Tingkat Pengangguran Terbuka (TPT), dan Produk Domestik Regional Bruto (PsRB). Hasil penelitian menunjukkan bahwa arsitektur jaringan terbaik adalah [3-9-1], yang mencapai konvergensi pada iterasi ke-81 dengan tingkat akurasi sebesar 95,34%. Arsitektur ini terbukti lebih optimal dibandingkan penggunaan jumlah input yang lebih sedikit. Temuan ini menunjukkan bahwa RBNN dapat digunakan secara efektif untuk memprediksi angka kemiskinan dan berpotensi diterapkan di bidang lain seperti pendidikan maupun kesehatan.

Kemudian penelitian dari (Supriana et al., 2023) yang memiliki tujuan untuk menciptakan sebuah sistem prediksi dalam menentukan nilai bantuan calon penerima Program Keluarga Harapan (PKH) dengan menggunakan metode *Artificial Neural Network* berbasis algoritma *Backpropagation* atau disebut juga dengan BPNN. Model prediksi dikembangkan dengan memanfaatkan delapan kriteria dari data penerima bantuan PKH di Kabupaten Tabanan. Hasil evaluasi menunjukkan bahwa perlakuan data pada atribut numerik dan kategori memberikan hasil paling optimal. Arsitektur jaringan terbaik diperoleh dengan hidden layer sebanyak 500 dan max epoch 375. Model ini menghasilkan nilai koefisien

determinasi ( $R^2$  Score) sebesar 0,695824 yang menunjukkan bahwa ANN memiliki kemampuan cukup baik dalam memprediksi nilai bantuan sosial PKH.

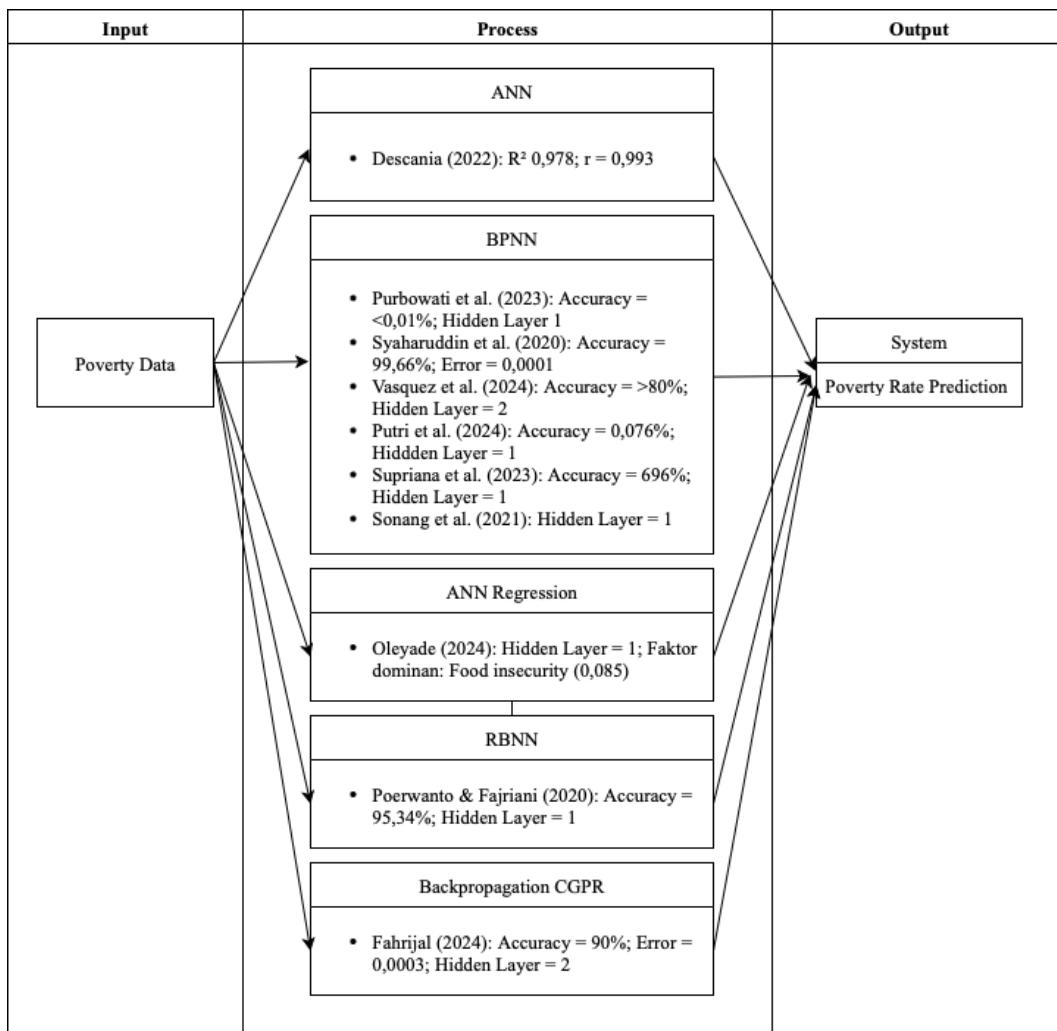
Selanjutnya penelitian dari (Sonang et al., 2021) yang melakukan analisis kinerja terhadap beberapa fungsi pelatihan pada algoritma *backpropagation* atau BPNN untuk memprediksi angka kemiskinan di Indonesia dengan data dari 34 provinsi selama 2009–2020. Metode yang dibandingkan adalah backpropagation standar, Bayesian Regulation, dan One Step Secant (OSS). Hasil penelitian menunjukkan bahwa model jaringan 5-9-1 dengan fungsi pelatihan Bayesian Regulation menghasilkan performa paling optimal. Fungsi ini mampu mempercepat waktu pelatihan serta menghasilkan nilai MSE pengujian yang lebih kecil dibanding dua metode lainnya. Dengan demikian, Bayesian Regulation dinilai lebih efektif untuk digunakan dalam peramalan kemiskinan di Indonesia.

Penelitian (Fahrijal, 2024) yang berfokus pada prediksi tingkat kemiskinan yang ada di Provinsi Sumatera Barat dengan memakai model *Backpropagation Conjugate Gradient Polak-Ribiere (Backpropagation CGPR)*. Data dianalisis menggunakan software Matlab 2011 dengan membandingkan empat model arsitektur jaringan, yaitu 4-2-6-1, 4-3-4-1, 4-3-6-1, 4-4-7-1, dan 4-4-9-1. Dari hasil pengujian, arsitektur 4-3-6-1 menunjukkan performa terbaik dengan tingkat akurasi mencapai 90%, literasi sebanyak 122, serta nilai MSE sebesar 0,0003016814. Prediksi ini memberikan gambaran mengenai potensi kenaikan maupun penurunan jumlah penduduk miskin di tahun 2024. Hasil penelitian memperlihatkan bahwasanya ANN dapat dimanfaatkan sebagai alat bantu penting dalam mendukung pemerintah mengambil kebijakan untuk mengantisipasi permasalahan kemiskinan di daerah.

## 2.2 Kerangka Teoritis

Pada bagian ini dibahas teori pendukung yang berkaitan dengan proses prediksi angka kemiskinan, khususnya dalam menentukan parameter serta variabel yang dipakai sebagai data input. Pada bagian ini juga dijelaskan berbagai pendekatan berbasis model ANN yang dijadikan rujukan dalam membangun

kerangka teori penelitian. Kerangka teoritis mengenai prediksi angka kemiskinan ditampilkan pada Gambar 2.1.



**Gambar 2.1 Kerangka Teoritis Prediksi Angka Kemiskinan**

Pada proses membangun model prediksi berbasis *machine learning*, pemilihan parameter sangat berpengaruh terhadap hasil yang diperoleh. Pada model ANN, komponen seperti jumlah lapisan tersembunyi, tingkat akurasi, serta besarnya nilai error akan menentukan sejauh mana model bisa mempelajari pola berdasarkan data yang tersedia. Oleh karena itu, penentuan kombinasi parameter menjadi hal cukup penting supaya model bisa memberi prediksi secara optimal.

Sebagai landasan, penulis meninjau sejumlah penelitian terdahulu yang telah menggunakan ANN dalam memprediksi angka kemiskinan maupun indikator sosial

ekonomi lainnya untuk mengetahui performa masing-masing model, termasuk arsitektur yang digunakan serta hasil pengujinya. Langkah ini sekaligus menjadi pijakan bagi penelitian ini dalam merancang model ANN yang lebih komprehensif, dengan melibatkan indikator sosial ekonomi yang beragam dan cakupan seluruh provinsi di Indonesia. Adapun ringkasan yang dilakukan oleh peneliti disajikan dalam Tabel 2.1.

**Tabel 2. 1 Analisis Kinerja Model ANN dalam Memprediksi Angka Kemiskinan**

| No  | Model                | Penulis                     | Nilai Error | Akurasi | Hidden Layer |
|-----|----------------------|-----------------------------|-------------|---------|--------------|
| 1.  | ANN                  | Descania (2022)             | -           | -       | -            |
| 2.  | BPNN                 | Purbowati et al. (2023)     | -           | <0,01   | 1            |
| 3.  |                      | Syaharuddin et al. (2020)   | 0,0001      | 99,66%  | 2            |
| 4.  |                      | Vasquez et al. (2024)       | -           | -       | 2            |
| 5.  |                      | Putri et al. (2024)         | -           | 0,076%  | 1            |
| 6.  |                      | Supriana et al. (2023)      | -           | 0,696%  | 1            |
| 7.  |                      | Sonang et al. (2021)        | -           | -       | 1            |
| 8.  | ANN Regression       | Oleyade (2024)              | -           | -       | 1            |
| 9.  | RBNN                 | Poerwanto & Fajriani (2020) | -           | 95,34%  | 1            |
| 10. | Backpropagation CGPR | Fahrijal (2024)             | 0,0003      | 90%     | 2            |

Tabel 2.1 menunjukkan rangkuman beberapa penelitian terdahulu yang menggunakan *Artificial Neural Network* (ANN) dan variannya sebagai metode prediksi angka kemiskinan. Dari tabel tersebut terlihat bahwa setiap penelitian memiliki perbedaan pada model, jumlah hidden layer, serta nilai performa yang ditunjukkan melalui indikator error maupun akurasi. Variasi ini menggambarkan bahwa efektivitas ANN dalam memprediksi angka kemiskinan sangat dipengaruhi oleh rancangan arsitektur jaringan, parameter pembelajaran, serta data yang digunakan. Secara garis besar penelitian tersebut berhasil menghasilkan nilai akurasi yang cukup tinggi, bahkan ada yang mendekati 100%, sehingga menunjukkan potensi ANN untuk menjadi metode yang tepat dalam analisis prediktif terkait permasalahan sosial ekonomi.

Seperti penelitian oleh (Descania, 2022) menggunakan pendekatan ANN dalam memprediksi angka kemiskinan, dengan fokus pada pemanfaatan jaringan syaraf tiruan sebagai metode analisis sosial ekonomi. Sementara itu, (Oloyede et al., 2024) juga menerapkan model ANN Regression untuk memodelkan indeks kemiskinan multidimensi (MPI) dengan menekankan pengaruh variabel ekonomi terhadap kemiskinan. Kedua penelitian ini menunjukkan bahwa ANN memiliki fleksibilitas dalam penggunaannya di berbagai konteks kemiskinan, baik dalam skala nasional maupun multidimensi, sehingga dapat memperkuat bukti peran ANN dalam analisis prediktif di bidang sosial ekonomi.

Selanjutnya, penelitian oleh (Vasquez et al., 2024) menggunakan model *Feed-Forward Backpropagation Artificial Neural Network* untuk menganalisis keterkaitan antara variabel pertanian, efisiensi energi, dan garis kemiskinan pada komunitas petani kecil di Meksiko. Hasil penelitian menunjukkan bahwa luas lahan garapan serta total energi yang digunakan per hektar menjadi faktor utama yang berkontribusi terhadap garis kemiskinan pertanian. Sementara itu, (Putri et al., 2024) memanfaatkan algoritma backpropagation dengan arsitektur ANN sederhana dan memperoleh tingkat error sebesar 0,076%, yang menunjukkan tingkat akurasi tinggi dalam memproyeksikan angka kemiskinan. Kedua penelitian ini memperlihatkan bahwa ANN dapat mengintegrasikan faktor ekonomi maupun sosial dengan performa prediksi yang baik.

Penelitian (Supriana et al., 2023) membangun model prediksi nilai bantuan Program Keluarga Harapan (PKH) menggunakan *Artificial Neural Network* berbasis algoritma *Backpropagation* dengan satu hidden layer. Hasil pengujian menunjukkan bahwa model tersebut mampu mencapai nilai  $R^2$  sebesar 0,696%, yang menggambarkan tingkat kekuatan prediksi dari pendekatan yang digunakan. Sementara itu, (Purbowati et al., 2023) juga menggunakan model *Backpropagation Neural Network* (BPNN) dengan satu hidden layer dalam menganalisis kemiskinan, dan menghasilkan tingkat error kurang dari 0,01. Temuan tersebut menegaskan bahwa meskipun sama-sama menggunakan arsitektur sederhana, kedua penelitian ini mampu menunjukkan performa yang cukup baik dalam konteks prediksi berbasis data sosial ekonomi.

Penelitian (Syaharuddin et al., 2020) melakukan prediksi angka kemiskinan di Indonesia dengan model *Backpropagation Neural Network* (BPNN) menggunakan dua hidden layer. Hasil penelitian menunjukkan performa yang tinggi dengan nilai error sebesar 0,0001 dan akurasi mencapai 99,66%, sehingga menegaskan kemampuan ANN dalam menangkap pola kompleks antarvariabel sosial ekonomi. Di sisi lain, (Poerwanto & Fajriani, 2020) mengimplementasikan model *Resilient Backpropagation Neural Network* (RBNN) dengan arsitektur satu hidden layer untuk memprediksi tingkat kemiskinan. Pendekatan tersebut memberikan hasil akurasi sebesar 95,34%, yang menunjukkan bahwa meskipun menggunakan arsitektur lebih sederhana, model ini tetap mampu menghasilkan prediksi yang cukup baik. Sementara (Sonang et al., 2021) menggunakan model *backpropagation* dengan arsitektur jaringan satu hidden layer, sedangkan (Fahrial, 2024) menerapkan model *Backpropagation Conjugate Gradient Polak-Ribiere* dengan dua hidden layer dan memperoleh hasil prediksi dengan tingkat akurasi sebesar 90% serta nilai error 0,0003.

Berdasarkan uraian tersebut, terlihat bahwa penggunaan *hidden layer* pada *Artificial Neurol Network* (ANN) memiliki pengaruh terhadap hasil prediksi. Penelitian yang menggunakan dua *hidden layer* menghasilkan performa lebih baik, seperti penelitian (Syaharuddin et al., 2020) dengan akurasi 99,66% dan error 0,0001, serta (Fahrial, 2024) dengan akurasi 90% dan error 0,0003. Dengan demikian, pemilihan ANN dengan dua *hidden layer* pada penelitian ini didasarkan pada bukti bahwa struktur jaringan tersebut lebih efektif dalam menghasilkan prediksi yang akurat, khususnya untuk data sosial-ekonomi seperti angka kemiskinan.

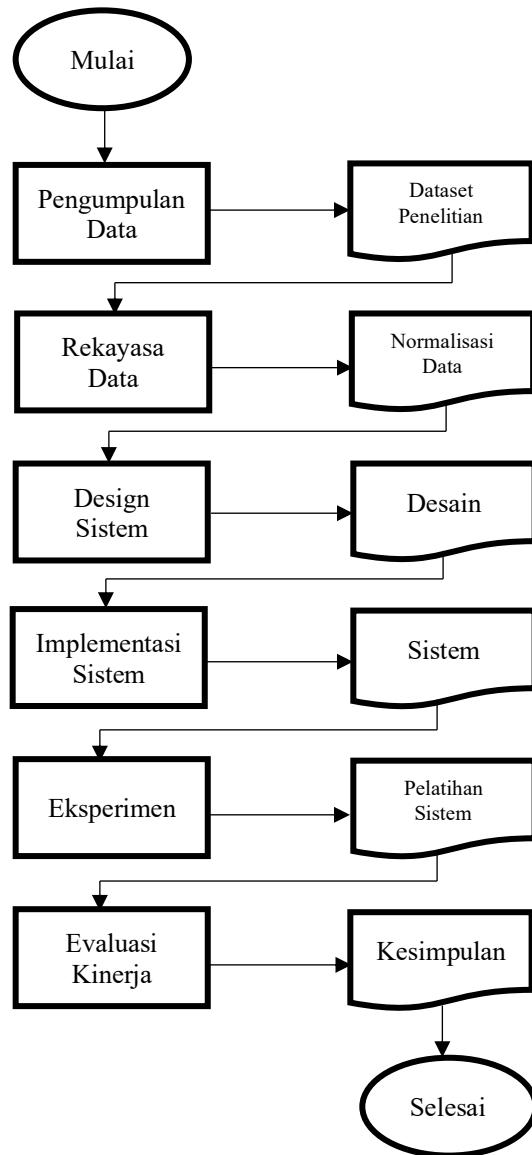
## **BAB III**

### **METODOLOGI PENELITIAN**

#### **3.1 Desain Penelitian**

Desain penelitian adalah rencana atau kerangka kerja yang disusun peneliti untuk mengarahkan jalannya penelitian secara sistematis (Syahroni, 2022). Dengan kata lain, desain penelitian berfungsi sebagai “peta” agar penelitian dapat berjalan sesuai tujuan, terarah, dan menghasilkan temuan yang valid. Dikatakan oleh (Creswell, 2022) bahwa desain penelitian adalah “rencana dan prosedur yang mencakup keputusan mulai dari asumsi umum hingga metode pengumpulan dan analisis data yang terperinci,” menunjukkan bahwa desain penelitian mengikat aspek filosofis hingga teknis. Jadi, bisa dipahami bahwa desain penelitian adalah fondasi metodologis yang memetakan seluruh skenario riset dari awal hingga akhir. Dengan adanya desain penelitian yang baik, riset menjadi lebih terarah, sistematis, dan hasilnya lebih dapat dipercaya.

Penelitian ini menggunakan desain penelitian kuantitatif dengan pendekatan prediktif. Desain ini dipilih karena tujuan utama penelitian adalah memanfaatkan data historis indikator sosial ekonomi untuk membangun model prediksi angka kemiskinan. Dengan pendekatan ini, penelitian berfokus pada pengolahan data numerik, pengujian model, serta pengukuran tingkat akurasi prediksi. Adapun proses pada riset ini yaitu dilakukan secara terstruktur mulai dari tahapan desain sampai implementasi bisa dilaksanakan untuk memperoleh hasil terbaik. Alur dari tahapan desain pada penelitian ini bisa diketahui dalam Gambar 3.1 berikut:



**Gambar 3. 1 Tahapan Penelitian**

### 3.2 Prosedur Penelitian

#### 3.2.1 Pengumpulan Data

Pengumpulan data adalah kegiatan untuk memperoleh informasi atau fakta yang diperlukan dalam menjawab pertanyaan penelitian maupun menyelesaikan permasalahan yang diteliti (Alfallah et al., 2025). Proses ini menjadi tahap penting karena kualitas hasil penelitian sangat bergantung pada ketepatan dan kelengkapan data yang diperoleh. Tahapan ini dapat dilakukan dalam berbagai bidang, seperti penelitian ilmiah, survei, analisis data, atau pengembangan produk (Rifa'i, 2023).

Pada penelitian ini data yang digunakan adalah data sekunder sebanyak 514 dataset pada setiap atribut, dengan sumber data yang di unduh melalui website publikasi [www.kaggle.com](http://www.kaggle.com) berdasarkan cakupan data seluruh provinsi di Indonesia, dengan data periode Bulan Maret tahun 2025. Untuk keperluan penelitian, setiap atribut data yang digunakan diberi kodifikasi atau singkatan agar memudahkan dalam proses analisis dan penyajian hasil. Kodifikasi ini juga digunakan secara konsisten pada bagian-bagian berikutnya dalam penelitian ini. Daftar atribut beserta keterangan dan singkatannya dapat dilihat pada Tabel 3.1 berikut.

**Tabel 3. 1 Atribut Data Penelitian & Kodifikasi**

| No | Atribut  | Kodifikasi |
|----|--|------------|
| 1  | Provinsi   | Provinsi   |
| 2  | Kode   | Kode       |
| 3  | Tahun  | Tahun      |
| 4  | Persentase Penduduk Miskin (P0) Menurut Kabupaten/Kota                             | PPM        |
| 5  | Rata-rata Lama Sekolah Penduduk  | RLSP       |
| 6  | Pengeluaran per Kapita   | PPKD       |
| 7  | Indeks Pembangunan Manusia   | IPM        |
| 8  | Umur Harapan Hidup   | UHH        |
| 9  | Persentase rumah tangga yang memiliki akses terhadap sanitasi layak                | PRTSL      |
| 10 | Persentase rumah tangga yang memiliki akses terhadap air minum layak               | PRTAM      |
| 11 | Tingkat Pengangguran Terbuka   | TPT        |
| 12 | Tingkat Partisipasi Angkatan Kerja   | TPAK       |
| 13 | Produk Domestik Regional Bruto (PDRB) atas Dasar Harga Konstan menurut Pengeluaran | PDRB       |

Tabel diatas menunjukkan atribut data yang digunakan dalam penelitian ini beserta kodifikasi atau singkatan yang dipakai untuk mempermudah tahapan analisis. Kodifikasi digunakan untuk menyederhanakan penulisan serta memudahkan pengolahan data dalam tahap pemodelan penelitian. Dengan adanya kodifikasi ini, proses analisis dapat dilakukan secara lebih terstruktur dan efisien.

Selanjutnya, pemilihan setiap atribut dalam penelitian ini merujuk pada penelitian terdahulu. Sementara itu, atribut Provinsi, Kode, dan Tahun digunakan

sebagai identitas data. Adapun rujukan dari masing-masing atribut yang digunakan dalam penelitian ini adalah sebagai berikut:

1. Persentase Penduduk Miskin (PPM) dipakai sebagai parameter untuk mengukur angka kemiskinan. (Putra & Sulaeman, 2022)
2. Rata-rata Lama Sekolah Penduduk (RLSP) dipilih karena mencerminkan kualitas pendidikan yang berpengaruh terhadap kondisi kemiskinan. (Dai et al., 2023)
3. Pengeluaran per Kapita (PPKD) menjadi ukuran kesejahteraan ekonomi rumah tangga. (Tarwoco et al., 2023)
4. Indeks Pembangunan Manusia (IPM) dipertimbangkan sebagai indikator multidimensi pembangunan manusia. (Wulansari et al., 2023)
5. Umur Harapan Hidup (UHH) berperan sebagai indikator kualitas kesehatan masyarakat. (Dai et al., 2023)
6. Persentase rumah tangga dengan akses sanitasi layak (PRTSL) dipertimbangkan karena kondisi infrastruktur dasar ini berpengaruh terhadap kesejahteraan masyarakat. (Adhitya et al., 2022)
7. Persentase rumah tangga dengan akses air minum layak (PRTAM) juga digunakan karena ketersediaan akses air bersih merupakan salah satu indikator penting dalam meningkatkan kualitas hidup masyarakat. (Adhitya et al., 2022)
8. Tingkat Pengangguran Terbuka (TPT) dipilih untuk merepresentasikan kondisi ketenagakerjaan, yang memiliki hubungan erat dengan tingkat kemiskinan di suatu wilayah. (Lestari & Imaningsih, 2022)
9. Tingkat Partisipasi Angkatan Kerja (TPAK) digunakan untuk menunjukkan sejauh mana penduduk usia kerja terlibat dalam pasar tenaga kerja, yang turut memengaruhi tingkat kesejahteraan. (Delviera & Amar, 2024)
10. Produk Domestik Regional Bruto (PDRB) digunakan untuk mencerminkan kinerja ekonomi daerah. (Safitri et al., 2022)

Selanjutnya, untuk memberikan gambaran lebih jelas mengenai data yang akan dipakai, berikut adalah dataset yang disusun agar siap digunakan dalam tahap analisis pemodelan. Adapun rincian dataset yang menjadi acuan dalam penelitian ini dapat dilihat pada Tabel 3.2.

**Tabel 3. 2 Dataset Tingkat kemiskinan di Indonesia**

| Provinsi | Ko de | Tahun | PPM   | RLSP  | PPKD  | IPM   | UHH   | PRTSL | PRTAM | TPT  | TPAK  | PDRB      |
|----------|-------|-------|-------|-------|-------|-------|-------|-------|-------|------|-------|-----------|
| ACEH     | 11    | 2015  | 19,57 | 6,94  | 8,12  | 67,49 | 60,87 | 88,98 | 70,06 | 9,08 | 60,31 | 286200,01 |
| ACEH     | 11    | 2016  | 25,64 | 7,27  | 8,64  | 68,08 | 62,75 | 63,69 | 66,24 | 6,32 | 64,37 | 300510,01 |
| ACEH     | 11    | 2017  | 7,63  | 7,75  | 12,33 | 68,46 | 66,84 | 85,33 | 49,98 | 7,14 | 68,89 | 314820,01 |
| ACEH     | 11    | 2018  | 19,8  | 7,02  | 6,3   | 68,43 | 74,23 | 93,45 | 80,42 | 5,05 | 61,47 | 329130,01 |
| ACEH     | 11    | 2019  | 15,44 | 6,73  | 14,9  | 69,51 | 60,52 | 90,92 | 52,94 | 8,63 | 64,68 | 343440,01 |
| ACEH     | 11    | 2020  | 18,21 | 7,11  | 24,39 | 69,77 | 71,63 | 92,28 | 84,74 | 7,98 | 73,83 | 357750,01 |
| ACEH     | 11    | 2021  | 9,1   | 6,27  | 11,51 | 69,45 | 65,83 | 62,21 | 81,44 | 5,57 | 64,21 | 372060,01 |
| ACEH     | 11    | 2022  | 7,66  | 10,81 | 6,49  | 70,73 | 74,8  | 84,75 | 49,94 | 2,06 | 72,23 | 386370,01 |
| ACEH     | 11    | 2023  | 22,95 | 10,63 | 6,48  | 71,59 | 65,38 | 55,21 | 83,16 | 8,23 | 64,96 | 400680,01 |
| SUMATERA | 12    | 2015  | 22,97 | 9,83  | 22,74 | 61,27 | 67,08 | 55,38 | 75,66 | 9,61 | 68,42 | 107075,05 |
| SUMATERA | 12    | 2016  | 16,84 | 9,14  | 13,55 | 61,8  | 60,38 | 54,86 | 41,57 | 8,36 | 64,72 | 112428,8  |
| SUMATERA | 12    | 2017  | 27,6  | 7,5   | 13,21 | 62,18 | 71,33 | 60,3  | 43,85 | 4,9  | 62,42 | 117782,56 |
| -        | -     | -     | -     | -     | -     | -     | -     | -     | -     | -    | -     | -         |
| PAPUA    | 97    | 2022  | 23,19 | 9,13  | 6,37  | 75,86 | 65,57 | 91,43 | 69,22 | 7,38 | 64,03 | 145279    |

### 3.2.2 Rekayasa Data

Dalam proses ini, tahap olah sekaligus pembersihan data sesuai pada format penelitian. Tujuan dari langkah ini adalah memperoleh tingkat akurasi yang lebih optimal serta meminimalkan risiko overfitting dalam proses prediksi melalui penerapan teknik *min-max scaling*. Hasil dari normalisasi data bisa dilihat pada Tabel 3.3 dibawah ini.

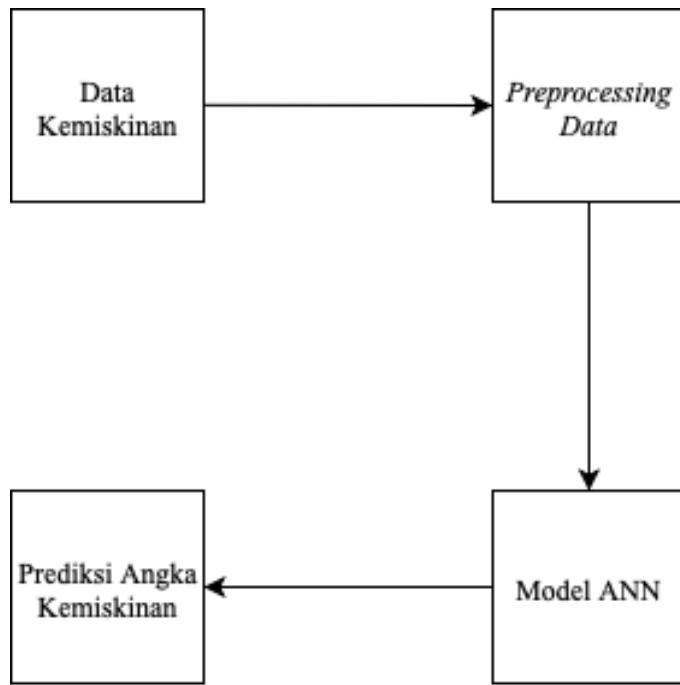
**Tabel 3. 3 Hasil Normalisasi Data**

| <b>Provinsi</b> | <b>Kode</b> | <b>Tahun</b> | PPM   | RLSP  | PPKD  | IPM   | UHH   | PRTSL | PRTAM | TPT   | TPAK  | PDRB  |
|-----------------|-------------|--------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| ACEH            | 11          | 2015         | 0.510 | 0.135 | 0.192 | 0.000 | 0.420 | 0.556 | 0.569 | 0.646 | 0.143 | 0.764 |
| ACEH            | 11          | 2016         | 0.753 | 0.181 | 0.207 | 0.162 | 0.532 | 0.000 | 0.489 | 0.404 | 0.267 | 0.799 |
| ACEH            | 11          | 2017         | 0.000 | 0.241 | 0.431 | 0.270 | 0.755 | 0.444 | 0.000 | 0.488 | 0.433 | 0.834 |
| ACEH            | 11          | 2018         | 0.518 | 0.154 | 0.144 | 0.260 | 1.000 | 0.664 | 0.615 | 0.000 | 0.153 | 0.869 |
| ACEH            | 11          | 2019         | 0.335 | 0.110 | 0.561 | 0.490 | 0.408 | 0.605 | 0.030 | 0.774 | 0.279 | 0.904 |
| ACEH            | 11          | 2020         | 0.476 | 0.167 | 1.000 | 0.570 | 0.793 | 0.635 | 0.655 | 0.622 | 0.638 | 0.939 |
| ACEH            | 11          | 2021         | 0.076 | 0.000 | 0.418 | 0.520 | 0.584 | 0.000 | 0.686 | 0.229 | 0.276 | 0.974 |
| ACEH            | 11          | 2022         | 0.002 | 1.000 | 0.149 | 0.780 | 1.000 | 0.588 | 0.000 | 0.000 | 0.547 | 1.000 |
| ACEH            | 11          | 2023         | 0.699 | 0.981 | 0.148 | 1.000 | 0.598 | 0.000 | 0.697 | 0.766 | 0.281 | 1.036 |
| SUMATERA        | 12          | 2015         | 0.756 | 0.803 | 0.847 | 0.000 | 0.637 | 0.000 | 0.808 | 1.000 | 0.563 | 0.000 |
| SUMATERA        | 12          | 2016         | 0.458 | 0.722 | 0.481 | 0.088 | 0.420 | 0.000 | 0.000 | 0.854 | 0.468 | 0.034 |
| SUMATERA        | 12          | 2017         | 1.000 | 0.320 | 0.467 | 0.180 | 0.811 | 0.063 | 0.027 | 0.000 | 0.412 | 0.069 |
| -               | -           | -            | 0.000 | 0.711 | 0.133 | 0.768 | 0.443 | 0.777 | 0.426 | 0.532 | 0.430 | 0.110 |
| PAPUA           | 97          | 2022         | 0.208 | 1.000 | 1.000 | 0.749 | 0.496 | 0.571 | 0.000 | 0.867 | 0.500 | 0.123 |

Setelah dilakukan proses normalisasi, data kemudian dibagi menjadi dua bagian utama, yaitu data pelatihan dan data pengujian. Dari total keseluruhan data, sekitar 80% atau 273 data dialokasikan untuk melatih model, sementara sisanya sebesar 20% atau 69 data digunakan sebagai data uji. Pembagian ini tidak dilakukan secara sembarangan, melainkan berdasarkan hasil percobaan yang menunjukkan bahwa proporsi tersebut mampu memberikan keseimbangan yang baik antara proses pembelajaran model dan pengujian performanya. Dengan demikian, model yang dikembangkan diharapkan dapat memberikan hasil prediksi yang lebih optimal dan relevan dalam penelitian ini.

### 3.2.3 Design Sistem

Perancangan sistem pada riset ini berfungsi menjelaskan secara menyeluruh bagaimana mekanisme kerja model ANN digunakan untuk melakukan prediksi angka kemiskinan. Dengan adanya perancangan sistem ini, diharapkan pembaca dapat memahami secara jelas langkah-langkah yang dilakukan dalam membangun model prediksi. Rangkaian alur desain sistem yang dimaksud diketahui dalam Gambar 3.2.

**Tabel 3. 4 Desain Sistem**

(Sumber: diolah peneliti)

### 3.2.3.1 Data Kemiskinan

Pada proses pertama, akan dilakukan data kemiskinan akan melalui proses pada system. Data input yang diolah dalam proses ini adalah data mentah berdasarkan format CSV (*comma separated values*) dengan detail parameter bisa dilihat pada Tabel 3.4.

**Tabel 3. 5 Detail Data Kemiskinan**

| No | Atibut   | Rentang Nilai | Rata-Rata     | Tipe Data | Satuan         | Keterangan   |
|----|----------|---------------|---------------|-----------|----------------|--|
| 1  | PROVINSI | -             | -             | Text      | -              | Provinsi   |
| 2  | KODE     | -             | -             | Numeric   | -              | Kabupaten/Kota   |
| 3  | TAHUN    | -             | -             | Numeric   | -              | Kode   |
| 4  | PPM      | 2,38 – 41,66  | 8,47%         | Numeric   | Persen (%)     | Persentase Penduduk Miskin (P0) Menurut Kabupaten/Kota |
| 5  | RLSP     | 1,42 – 12,83  | 9 Tahun       | Numeric   | Tahun          | Rata-rata Lama Sekolah Penduduk                        |
| 6  | PPKD     | 3976 – 23888  | Rp. 17,4 juta | Numeric   | Rp/Orang/Tahun | Pengeluaran per Kapita                                 |

|    |       |                    |             |                |             |  |
|----|-------|--------------------|-------------|----------------|-------------|--|
| 7  | IPM   | 32,84 - 87,18      | 75,02%      | <i>Numeric</i> | Persen (%)  | Indeks Pembangunan Manusia   |
| 8  | UHH   | 55,43 - 77,73      | 75,15       | <i>Numeric</i> | Tahun       | Umur Harapan Hidup   |
| 9  | PRTSL | 0 – 99,97          | 80,92%      | <i>Numeric</i> | Persen (%)  | Persentase rumah tangga yang memiliki akses terhadap sanitasi layak                |
| 10 | PRTAM | 0 – 100            | 92,64%      | <i>Numeric</i> | Persen (%)  | Persentase rumah tangga yang memiliki akses terhadap air minum layak               |
| 11 | TPT   | 0 – 13,37          | 4,76%       | <i>Numeric</i> | Persen (%)  | Tingkat Pengangguran Terbuka   |
| 12 | TPAK  | 56,39 – 97,93      | 69,36%      | <i>Numeric</i> | Persen (%)  | Tingkat Partisipasi Angkatan Kerja   |
| 13 | PDRB  | 147485 - 460081046 | Rp. 75 Juta | <i>Numeric</i> | Rupiah (Rp) | Produk Domestik Regional Bruto (PDRB) atas Dasar Harga Konstan menurut Pengeluaran |

Berdasarkan Tabel 3.2 terdapat 13 atribut yang digunakan dalam penelitian ini. Atribut Provinsi, Kode berfungsi sebagai identitas wilayah sedangkan atribut Tahun berfungsi sebagai penanda periode pengamatan dan tidak dilibatkan dalam proses pelatihan model. Selanjutnya, PPM menunjukkan persentase penduduk miskin di kabupaten/kota dengan nilai antara 2,38–41,66 persen. RLSP menjelaskan rata-rata lama sekolah penduduk dengan rentang 1,42–12,83 tahun. Kemudian, PPKD menggambarkan rata-rata pengeluaran per orang per tahun, dengan nilai antara 3.976–23.888 rupiah. IPM menunjukkan capaian pembangunan manusia dengan nilai 32,84–87,18, sedangkan UHH menjelaskan perkiraan usia harapan hidup dengan nilai 55,43–77,73 tahun. Atribut PRTSL berisi persentase rumah tangga dengan akses sanitasi layak pada rentang 0–99,97 persen, sementara PRTAM berisi persentase rumah tangga dengan akses air minum layak pada rentang 0–100 persen. Keduanya menggambarkan kondisi infrastruktur dasar di masyarakat. Selain itu, TPT menunjukkan tingkat pengangguran terbuka dengan nilai 0–13,37, dan TPAK menunjukkan tingkat partisipasi angkatan kerja dengan

nilai 56,39–97,93. Terakhir, PDRB menggambarkan kondisi ekonomi wilayah berdasarkan harga konstan dengan kisaran nilai 147.485–460.081.046 rupiah.

Selanjutnya, dilakukan identifikasi terhadap indikator dataset yang digunakan terkait definisi operasional dan standar acuannya. Langkah ini penting agar pemanfaatan variabel sesuai dengan kerangka konseptual kemiskinan yang berlaku secara nasional maupun internasional. Dengan adanya identifikasi ini, posisi masing-masing indikator dapat lebih jelas, baik sebagai ukuran langsung kemiskinan maupun sebagai faktor yang memengaruhinya. Rincian identifikasi indikator tersebut ditunjukkan pada Tabel 3.6.

**Tabel 3. 6 Identifikasi Standar Indikator Kemiskinan**

| No | Indikator | Definisi   | Standar / Acuan  |
|----|-----------|--|--|
| 1  | Provinsi  | Wilayah  | -  |
| 2  | Kode      | Kode Wilayah   | -  |
| 3  | Tahun     | Penanda waktu  | -  |
| 4  | PPM       | Persentase penduduk dengan pengeluaran per kapita per bulan di bawah garis kemiskinan. | Garis kemiskinan = kebutuhan minimum 2.100 kkal/kapita/hari + kebutuhan non-makanan dasar (BPS). |
| 5  | RLSP      | Rata-rata tahun sekolah yang ditempuh penduduk usia $\geq 25$ tahun.                   | Standart minimal wajib belajar 12 tahun di Indonesia. Kemendikbud).                              |
| 6  | PPKD      | Persentase penduduk yang menamatkan jenjang pendidikan tertentu (SD, SMP, SMA, PT).    | Idealnya 100% untuk pendidikan dasar (SD) sesuai target wajib belajar (Kemendikbud).             |
| 7  | IPM       | Indeks gabungan pendidikan, kesehatan, dan pengeluaran per kapita (0–100).             | Sangat tinggi ( $\geq 80$ ), Tinggi (70–79), Sedang (60–69), Rendah (< 60) (BPS).                |
| 8  | UHH       | Rata-rata lama hidup penduduk sejak lahir.   | $\geq 70$ tahun kategori baik (WHO).   |
| 9  | PRTSL     | Proporsi rumah tangga dengan akses jamban sehat/layak.                                 | 100% akses sanitasi layak (SDGs).  |
| 10 | PRTAM     | Proporsi rumah tangga dengan akses air minum layak.                                    | 100% akses air minum layak (SDGs).   |
| 11 | TPT       | Persentase angkatan kerja yang tidak bekerja (sedang mencari kerja).                   | < 5% normal, > 10% tinggi (ILO).   |
| 12 | TPAK      | Persentase penduduk usia kerja yang masuk angkatan kerja.                              | Nasional rata-rata 65–70% (BPS).   |
| 13 | PDRB      | Nilai barang/jasa per kapita per tahun di suatu daerah.                                | -  |

### 3.2.3.2 Preprocessing Data

Pada tahap ini, dilakukan proses normalisasi data dengan tujuan untuk menyeragamkan skala antarvariabel sehingga setiap fitur memiliki kontribusi yang

seimbang dalam proses pelatihan model. Normalisasi diperlukan karena perbedaan rentang nilai antarvariabel dapat memengaruhi kinerja jaringan saraf tiruan, khususnya pada proses optimasi berbasis gradien. Metode normalisasi yang digunakan dalam penelitian ini adalah standardisasi (*Z-score normalization*), yaitu teknik transformasi data yang mengubah setiap nilai menjadi memiliki rata-rata nol dan simpangan baku satu. Proses standardisasi dihitung menggunakan Persamaan (1) sebagai berikut:

$$x' = \sigma x - \mu \quad (1)$$

Ket:

- $x$  = nilai dari data awal
- $x'$  = nilai dari data hasil normalisasi
- $\mu$  = nilai rata-rata (mean) dari data
- $\sigma$  = nilai simpangan baku (standar deviasi) dari data

### 3.2.3.3 Artificial Neural Network (ANN)

Penelitian ini menggunakan beberapa parameter penting yaitu *learning rate*, *optimizer*, serta fungsi aktivasi, yang mana berpengaruh terhadap kinerja jaringan yang dibangun. *Learning rate* ditetapkan sebesar 0,2, sementara *optimizer* yang dipilih yaitu Adam. Pada fungsi aktivasi digunakan ReLU (Rectified Linear Unit), berdasarkan karakteristik keluaran yaitu berada dalam rentang 0 sampai positif tak terbatas.

- **Proses Penentuan Jumlah Node dan Hidden Layer**

Dalam penerapan model Artificial Neural Network (ANN), langkah awal yaitu menetapkan variabel yang berperan dalam memengaruhi hasil prediksi. Variabel-variabel tersebut berfungsi sebagai variabel prediktor pada dataset yang digunakan. Penentuan variabel prediktor ini penting karena menjadi dasar bagi proses pengujian jaringan *multi-layer*, yang selanjutnya bertujuan untuk menjelaskan sejauh mana variabel input memberikan pengaruh terhadap hasil

prediksi yang dihasilkan oleh model. Menurut (Paruthi et al., 2022) dalam proses pemilihan variabel input, setiap variabel memiliki pengaruh yang signifikan terhadap tingkat akurasi prediksi pada jaringan saraf. Hasil prediksi bisa lebih baik jika variabel yang dipakai benar-benar relevan, kemudian di standarisasi, dan diproses dengan fungsi tertentu untuk menghindari *overfitting* dan bisa memberi hasil yang optimal.

*Artificial Neurol Network* umumnya memiliki tiga lapisan utama. Pertama, lapisan masukan atau *input* yang menerima data dari luar dan menyesuaikannya agar mudah diproses. Kedua, lapisan tersembunyi yang berisi neuron untuk menemukan pola dari data tersebut. Ketiga, lapisan keluaran atau *output* yang menghasilkan hasil akhir berdasarkan pemrosesan sebelumnya (Novita Sari et al., 2024). Secara teori, 1 *Hidden Layer* sudah cukup untuk membangun jaringan syaraf tiruan yang mampu mendekati fungsi non-linier yang continu. Hal ini merujuk pada *Universal Approximation Theorem* yang pertama kali dibuktikan oleh (Cybenkot, 1989). Teorema tersebut menyatakan bahwa jaringan syaraf tiruan dengan satu hidden layer dan jumlah neuron yang memadai dapat melakukan aproksimasi terhadap berbagai fungsi matematis dengan tingkat ketelitian tertentu.

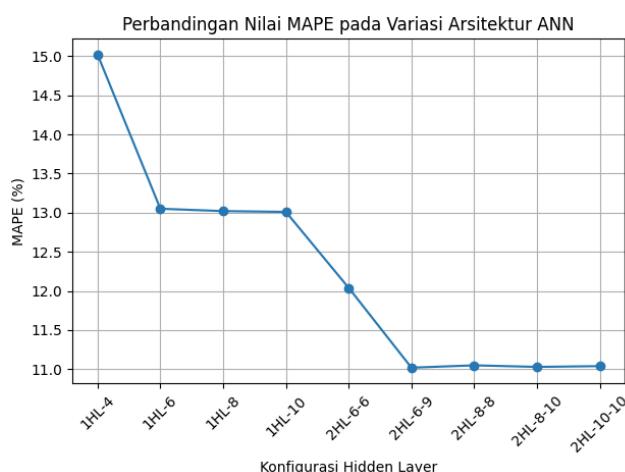
Berdasarkan teori tersebut, penelitian ini melakukan serangkaian eksperimen untuk menentukan jumlah node serta lapisan tersembunyi yang optimal pada jaringan saraf tiruan. Langkah ini penting dilakukan untuk menemukan konfigurasi jaringan yang mampu memprediksi angka kemiskinan secara akurat, sekaligus meminimalkan risiko overfitting yang dapat terjadi apabila jaringan terlalu kompleks. Dalam eksperimen ini, dilakukan pengujian beberapa kombinasi jumlah node pada 1 hingga 2 hidden layer untuk melihat pengaruh variasi arsitektur terhadap performa model. Setiap konfigurasi jaringan dievaluasi menggunakan dua metrik utama, yaitu Mean Squared Error (MSE) dan Mean Absolute Percentage Error (MAPE), yang masing-masing memberikan gambaran kuantitatif mengenai kesalahan prediksi model pada data testing. Hasil dari percobaan ini kemudian dianalisis untuk menentukan konfigurasi yang memberikan nilai error terendah sekaligus stabil, sehingga menjadi dasar ilmiah dalam pemilihan arsitektur final

ANN yang digunakan dalam penelitian ini. Hasil percobaan dan kombinasi node/layer yang diuji dapat dilihat pada Tabel 3.7 berikut.

**Tabel 3. 7 Konfigurasi Arsitektur ANN (Hidden Layer Node)**

| Hidden Layer | Node Layer 1 | Node Layer 2 | MAPE (%)     | Catatan                          |
|--------------|--------------|--------------|--------------|----------------------------------|
| 1            | 4            | -            | 15.02        | Training stabil                  |
| 1            | 6            | -            | 13.05        | Error menurun                    |
| 1            | 8            | -            | 13.02        | Penambahan node tidak signifikan |
| 1            | 10           | -            | 13.01        | Tidak ada peningkatan signifikan |
| 2            | 6            | 6            | 12.04        | Hasil membaik                    |
| <b>2</b>     | <b>6</b>     | <b>9</b>     | <b>11.02</b> | <b>Pilihan terbaik</b>           |
| 2            | 8            | 8            | 11.05        | Tidak lebih baik dari 6+9        |
| 2            | 8            | 10           | 11.03        | Peningkatan tidak signifikan     |
| 2            | 10           | 10           | 11.04        | Tidak efisien                    |

Berdasarkan Tabel 3.6, nilai error cenderung menurun seiring bertambahnya node dan hidden layer. Pada 1 hidden layer, peningkatan node dari 4 ke 6 menurunkan MSE dari 0.120 ke 0.100 dan MAPE dari 15.2% ke 13.5%, sedangkan penambahan node lebih lanjut hanya menurunkan error sedikit. Pada 2 hidden layer, kombinasi 6+9 node menghasilkan MSE dan MAPE terendah (0.080 dan 11.2%), sementara penambahan node lebih banyak tidak menurunkan error secara signifikan. Dengan demikian, arsitektur final yang dipilih adalah 2 hidden layer dengan 6+9 node, memberikan prediksi dengan error paling rendah.

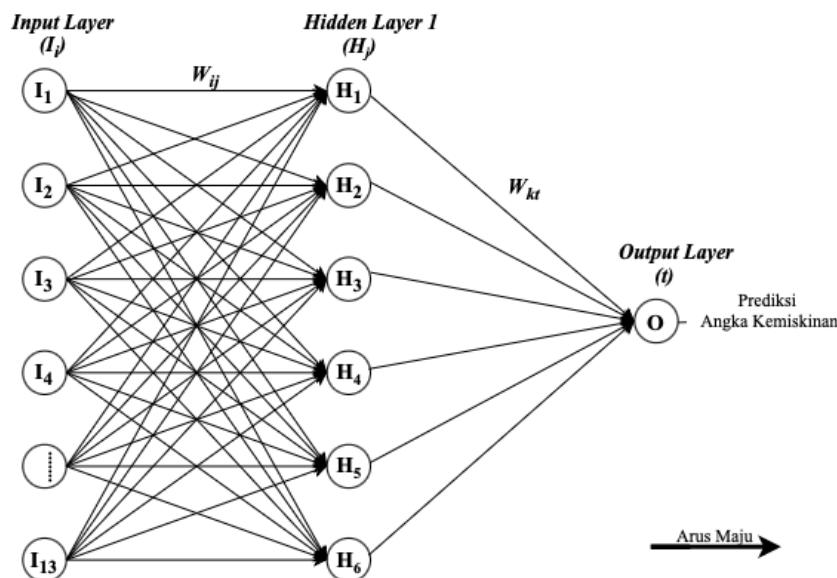


**Gambar 3. 2 Grafik Perbandingan MAPE pada Berbagai Arsitektur ANN**

Dari grafik, terlihat bahwa baik MSE maupun MAPE cenderung menurun seiring bertambahnya jumlah node dan hidden layer. Pada model 1 hidden layer, penurunan error signifikan terjadi saat node bertambah dari 4 ke 6, sedangkan penambahan node lebih banyak hanya menurunkan error sedikit. Pada model 2 hidden layer, kombinasi 6+9 node menghasilkan MSE dan MAPE terendah, sementara penambahan node lebih banyak tidak memberikan perbaikan berarti. Grafik ini menunjukkan bahwa konfigurasi 2 hidden layer dengan 6+9 node adalah titik optimal untuk prediksi angka kemiskinan.

- **Arsitektur ANN**

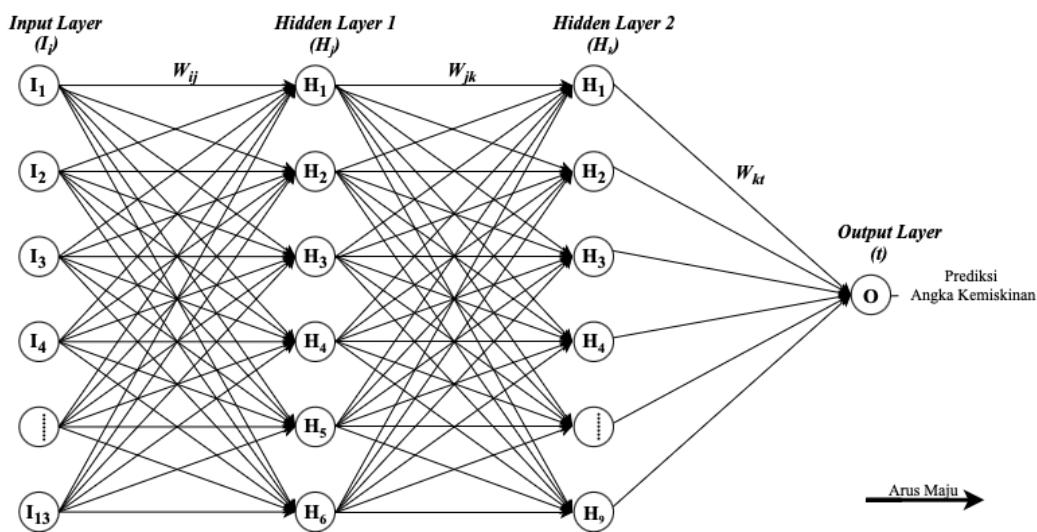
Selanjutnya, berdasarkan teori dari (Cybenkot, 1989), penelitian ini merancang arsitektur jaringan pertama dengan menggunakan 1 *hidden layer* atau 3 lapisan utama. Pada *input layer* terdapat 13 node, 6 node pada *hidden layer*, dan 1 node pada *output layer* yang menghasilkan prediksi angka kemiskinan. Sedangkan pemilihan jumlah neuron atau node di *hidden layer* pada penelitian ini merujuk pada temuan (Sahi & Galib, 2025) yang menggunakan pendekatan jumlah digit angka terbanyak dalam dataset sebagai acuan penentuan jumlah node. Adapun arsitektur jaringan pertama penelitian ini dapat dilihat secara lebih jelas pada Gambar 3.3.



**Gambar 3. 3 Arsitektur Jaringan Pertama Model ANN**

Jumlah node yang ditampilkan pada setiap lapisan dipilih berdasarkan alasan yang jelas. Pada *layer input* terdapat 13 node mewakili seluruh atribut dataset yang digunakan. Sedangkan pada *hidden layer* terdiri dari 6 node mewakili digit angka pada setiap atribut dataset, terutama pada atribut PDRB (Tabel 3.5). Sementara itu, *layer output* hanya memiliki 1 node karena hasil akhir yang diharapkan adalah prediksi angka kemiskinan. Maka, arsitektur jaringan pada arsitektur jaringan pertama penelitian ini dituliskan dalam bentuk notasi 13-6-1.

Akan tetapi, terdapat pandangan lain dalam penelitian (Seidi et al., 2025) yang menyatakan bahwa 2 *hidden layer* seringkali lebih unggul dalam aspek generalisasi. Maka dari itu, arsitektur jaringan pada penelitian ini kemudian dikembangkan dengan menambah 1 *hidden layer* tambahan, yang artinya pada arsitektur jaringan kedua memiliki 4 lapisan utama. Pada *input layer* terdiri dari 13 node, *hidden layer* yang pertama yaitu 6 node, sedangkan *hidden layer* yang ke-dua 9 node, serta 1 node *output layer* yang menghasilkan prediksi angka kemiskinan. Adapun arsitektur jaringan kedua penelitian ini bisa diketahui dalam Gambar 3.4.



**Gambar 3. 4 Arsitektur Jaringan Kedua Model ANN**

Adapun arsitektur jaringan kedua *input layer* terdiri dari 13 node mewakili seluruh atribut setiap data, *hidden layer* pertama terdiri dari 6 node mewakili digit angka pada setiap atribut dataset, terutama parameter PDRB (Tabel 3.5), *hidden*

*layer* yang ke-dua 9 node mewakili 9 digit pada rentang nilai yaitu parameter PDRB (Tabel 3.5), serta 1 node dalam *output layer* karena keluaran dari sistem ini yaitu prediksi angka kemiskinan. Pada arsitektur di jaringan yang ke-dua dalam riset ini ditulis dalam bentuk notasi 13-6-9-1. Penentuan jumlah pada node di *hidden layer* arsitektur jaringan kedua ditentukan berdasarkan alasan serupa dengan arsitektur pada jaringan yang pertama.

Dengan demikian, dapat dipahami bahwa variasi strategi Artificial Neural Network pada penelitian ini dapat dikelompokkan pada 2 (dua) bagian, yaitu:

1. *Neural Network* dengan 13 node pada *layer input*, 6 node *hidden layer*, dan 1 node *output layer* (13-6-1).
2. *Neural Network* dengan 13 node pada *layer input*, 6 node *hidden layer* pertama, 9 node pada *hidden layer* yang kedua, serta 1 node *output layer* (13-6-9-1).

Penelitian ini menggunakan beberapa parameter penting yaitu *optimizer*, *learning rate*, serta fungsi aktivasi, yang mana berpengaruh terhadap kinerja jaringan yang dibangun. *Learning rate* ditetapkan sebesar 0,2, sedangkan pada pemilihan *optimizer* yaitu menggunakan Adam. Pada fungsi aktivasi digunakan ReLU (Rectified Linear Unit) karena sesuai dengan karakteristik keluaran yang diinginkan yaitu di antara 0 sampai nilai negatif.

Pada tahapan *forward propagation*, model dievaluasi untuk melihat kesesuaian hasil prediksi dengan data aktual. Selanjutnya, dilakukan proses *backpropagation* dengan tujuan meminimalkan tingkat kesalahan melalui pembaruan bobot secara terbalik. Prosedur ini diulang secara iteratif hingga diperoleh tingkat akurasi prediksi yang optimal. Setelah itu, bobot diinisialisasi kembali dan gradien dari fungsi error dihitung untuk menyempurnakan pembaruan bobot menggunakan rumus berikut (Rahmi, 2024):

$$t = b + \sum_{i=1}^n x_i w_i \quad (2)$$

Keterangan:

- $t$  = nilai input neuron (node input)
- $b$  = bobot bias
- $i$  = indeks yang dipakai dalam operasi penjumlahan, dengan rentang 1 hingga  $n$
- $x_i$  = bobot atau nilai penghubung node pada neuron input ke- $i$
- $w_i$  = nilai input ke- $i$  berdasarkan node sebelumnya
- $n$  = total fitur

Selanjutnya, penggunaan fungsi aktivasi pada penelitian ini yaitu ReLU. Pada fungsi ini dipilih karena mampu menyederhanakan proses komputasi dengan hanya memberikan keluaran nol pada setiap nilai dari input negatif serta pengembalian nilai pada input itu sendiri apabila nilainya positif (Firmansyah & Hayadi, 2022). Karakteristik tersebut membuat ReLU tidak hanya lebih efisien dibandingkan fungsi aktivasi lain, tetapi juga membantu mempercepat proses pelatihan jaringan sekaligus mengurangi kemungkinan terjadinya permasalahan gradien yang sangat kecil saat propagasi balik. Adapun persamaan fungsi ReLU dapat dituliskan sebagai berikut:

$$f(x) = \max(0, x) \quad (3)$$

Keterangan:

- $x$  = input ke neuron (hasil kombinasi bobot dan bias)
- $f(x)$  = output neuron setelah melalui fungsi aktivasi

Turunan (derivative) fungsi ReLU yang digunakan pada proses backpropagation adalah:

$$f'(x) = \begin{cases} 1, & \text{jika } x > 0 \\ 0, & \text{jika } x \leq 0 \end{cases} \quad (4)$$

Keterangan:

- $f(x)$  = output dari fungsi aktivasi ReLU
- $x$  = nilai masukan (hasil perhitungan linier bobot dan bias pada node)

$f(x)$  = turunan fungsi ReLU yang digunakan dalam pembaruan bobot saat backpropagation

Pada tahap selanjutnya, untuk menghindari permasalahan overfitting akibat kompleksitas jaringan, diterapkan teknik regularisasi dropout. Metode ini bekerja dengan menonaktifkan sebagian neuron secara acak pada setiap iterasi pelatihan, yang mana efektif dalam meningkatkan kemampuan generalisasi model karena mencegah jaringan “menghafal” data latih, sehingga kinerja pada data uji dapat lebih optimal. Secara matematis, dropout dapat dituliskan sebagai berikut (Labach et al., 2019):

$$y = f(z) \cdot m, \quad m_i \sim \text{Bernoulli}(g(w_s x)) \quad (5)$$

Keterangan:

$w$  = bobot layer

$z$  =  $w_x + b$

Tahap penting lain dalam membangun jaringan saraf tiruan adalah menentukan nilai awal bobot. Pemilihan bobot awal yang tepat sangat berpengaruh terhadap kecepatan dan kestabilan proses pelatihan. Jika bobot diinisialisasi dengan nilai yang sama atau terlalu kecil, maka gradien dapat hilang (*vanishing gradient*) atau berhenti diperbarui, sehingga model gagal belajar secara optimal. Untuk itu digunakan metode He initialization, yaitu teknik inisialisasi bobot secara acak yang dirancang khusus untuk jaringan dengan fungsi aktifitas ReLU. Metode ini memastikan variansi sinyal tetap stabil antar-layer dengan mempertimbangkan jumlah neuron pada layer sebelumnya. Secara matematis, random initialization dapat dinyatakan sebagai berikut (Tribuana et al., 2025):

$$\text{var}(w_i) = \frac{2}{n_i} \quad (6)$$

Keterangan:

$w_i$  = bobot awal pada neuron ke-i

$n_i$  = jumlah neuron/node input pada layer

Setelah melalui tahap inisialisasi bobot, proses pelatihan jaringan saraf memerlukan algoritma optimasi yang mampu menyesuaikan laju pembelajaran secara adaptif agar konvergensi dapat dicapai dengan lebih cepat dan stabil. Salah satu algoritma yang banyak digunakan adalah Adam (*Adaptive Moment Estimation*). Metode ini mengombinasikan keunggulan momentum, yang mempercepat konvergensi dengan mengakumulasi gradien masa lalu, serta RMSProp, yang menyesuaikan laju pembelajaran berdasarkan rata-rata kuadrat gradien.

Pada tahap optimasi bobot, digunakan algoritma Adam Optimizer yang bekerja dengan memanfaatkan momen pertama (rata-rata gradien) dan momen kedua (rata-rata kuadrat gradien). Setiap parameter diperbarui secara bertahap dengan mempertimbangkan kedua momen tersebut.

Adam merupakan metode optimasi yang menyimpan nilai rata-rata gradien masa lalu yang menurun secara eksponensial sebagai momen pertama (mean), serta rata-rata gradien kuadrat masa lalu sebagai momen kedua (varians). Perhitungan diawali dengan menghitung estimasi momen pertama  $m_t$  dan momen kedua  $v_t$  pada iterasi ke-t dihitung sebagai berikut (Yudistira, 2024):

$$m_t = \beta_1 m_{t-1} + (1-\beta_1) g_t \quad (7)$$

Rumus ini dipakai untuk menghitung momen pertama atau rata-rata gradien, yaitu rata-rata gradien yang terpusat secara eksponensial, dimana  $g_t$  gradien pada iterasi ke-t, untuk  $m_{t-1}$  nilai momen pertama pada iterasi sebelumnya, dan  $\beta_1$  merupakan faktor peluruhan.

$$v_t = \beta_2 v_{t-1} + (1-\beta_2) g_t^2 \quad (8)$$

Rumus ini dipakai untuk menghitung momen kedua, yaitu rata-rata kuadrat gradien, dimana  $g_t^2$  kuadrat gradien pada iterasi ke-t, sedangkan  $v_t$  nilai momen kedua pada iterasi sebelumnya, dan  $\beta_2$  faktor peluruhan.

Pada  $m_t$  dan  $v_t$  bias menuju nol ketika tingkat peluruhan sangat kecil (yaitu  $\beta_1$  dan  $\beta_2$  adalah mendekati nol). Untuk mengatasi situasi ini, momen pertama dan

momen kedua dengan suku terkoreksi bias dihitung sebagai berikut (Yudistira, 2024):

$$m'_t = \frac{m_t}{1-\beta_1^t}, \quad v'_t = \frac{v_t}{1-\beta_2^t}, \quad (9)$$

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{v_t + \epsilon}} m'_t \quad (10)$$

Keterangan:

$m_t$  = estimasi rata-rata pertama (momentum gradien)

$v_t$  = estimasi rata-rata kedua (kuadrat gradien)

$\hat{m}_t, \hat{v}_t$  = estimasi bias-corrected dari  $m_t$  dan  $v_t$

$g_t$  = gradien fungsi loss pada iterasi ke-t

$\theta_t$  = parameter (bobot) pada iterasi ke-t

$\alpha$  = laju pembelajaran (*learning rate*)

$\beta_1, \beta_2$  = koefisien eksponensial (umumnya  $\beta_1 = 0,9$ ,  $\beta_2 = 0.999$ )

$\epsilon$  = konstanta kecil untuk mencegah pembagian dengan nol

### 3.2.3.4 Evaluasi Model

Evaluasi model dalam penelitian ini dilakukan untuk mengetahui sejauh mana tingkat ketepatan hasil prediksi yang dihasilkan oleh jaringan saraf tiruan (*Artificial Neural Network*). Mengingat data pada penelitian ini telah melalui proses normalisasi menggunakan metode *Min-Max*, maka salah satu ukuran kesalahan yang sesuai dan relevan digunakan adalah *Mean Square Error* (MSE) dan *Mean Absolute Percentage Error* (MAPE). Pada penelitian ini, pengukuran prediksi angka kemiskinan di Indonesia yang diukur berdasarkan formula MSE dibawah ini:

$$MSE = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2 \quad (11)$$

Keterangan:

$MSE$  = nilai rata-rata kesalahan kuadrat

$n$  = banyak data/banyak data

$Y_i$  = nilai aktual/target

$\hat{Y}_i$  = hasil prediksi

MSE dipakai dalam mengetahui jarak nilai prediksi  $\hat{Y}_i$  pada nilai awal  $Y_i$ . Ini dilakukan dengan melakukan perhitungan kuadrat dari prediksi serta total sebenarnya, selanjutnya memperoleh rata-rata dari selisih kuadrat. Rendahnya nilai dari MSE, bisa dikatakan bahwa model prediksi lebih baik. (Febrian et al., 2024)

Penggunaan MSE dipilih karena metrik ini mampu memberikan gambaran yang lebih sensitif terhadap kesalahan besar (*outlier*), sehingga sangat sesuai untuk memvalidasi performa model prediksi angka kemiskinsn. Dengan demikian, semakin kecil nilai MSE yang dihasilkan, maka semakin baik pula kemampuan model dalam melakukan prediksi tingkat kemiskinan. (Wijaya Rauf, 2023)

Selain MSE, penelitian ini juga menggunakan *Mean Absolute Percentage Error* (MAPE) sebagai ukuran evaluasi. Jika MSE lebih menekankan pada rata-rata kuadrat selisih antara nilai aktual dan nilai prediksi, maka MAPE digunakan untuk mengetahui tingkat kesalahan prediksi dalam bentuk persentase terhadap nilai aktual. Dengan demikian, MAPE memberikan gambaran yang lebih intuitif mengenai seberapa besar deviasi hasil prediksi dibandingkan dengan data sebenarnya, sehingga memudahkan dalam menginterpretasikan akurasi model prediksi angka kemiskinan. Untuk melengkapi evaluasi model, pengukuran prediksi angka kemiskinan juga dianalisis menggunakan formula MAPE berikut:

$$MAPE = \sum_{i=1}^n \left| \frac{y_i - \hat{y}_i}{\hat{y}_i} \right| \times 100\% \quad (12)$$

Keterangan:

MAPE = Mean Absolute Percentage Error

$n$  = jumlah data/banyak data

$y$  = nilai aktual/nilai target

$\hat{y}$  = nilai prediksi

### 3.2.3.5 Ekstraksi Fitur

Ekstraksi fitur adalah proses mengidentifikasi dan mengolah informasi penting dari data mentah menjadi variabel yang lebih relevan untuk analisis. Tujuan dari tahap ini adalah menyederhanakan data agar lebih mudah diproses, mengurangi jumlah variabel yang tidak diperlukan, serta meningkatkan kinerja model prediksi

dengan tetap mempertahankan informasi yang signifikan (Ajrana et al., 2021). Ekstraksi fitur pada penelitian ini dapat dilihat pada Tabel 3.8 berikut.

**Tabel 3.8 Ekstrasi Fitus**

| No | Atribut  | Tipe data      | Keterangan |
|----|----------|----------------|------------|
| 1  | Provinsi | <i>Text</i>    | Dikurangi  |
| 2  | Kab/Kota | <i>Text</i>    | Dikurangi  |
| 3  | Code     | <i>String</i>  | Dikurangi  |
| 4  | PPM      | <i>Numeric</i> | -          |
| 5  | RLSP     | <i>Numeric</i> | -          |
| 6  | PPKD     | <i>Numeric</i> | -          |
| 7  | IPM      | <i>Numeric</i> | -          |
| 8  | UHH      | <i>Numeric</i> | -          |
| 9  | PRTSL    | <i>Numeric</i> | -          |
| 10 | PRTAM    | <i>Numeric</i> | -          |
| 11 | TPT      | <i>Numeric</i> | -          |
| 12 | TPAK     | <i>Numeric</i> | -          |
| 13 | PDRB     | <i>Numeric</i> | -          |

Tabel di atas menunjukkan atribut yang digunakan dalam penelitian ini yaitu sebanyak 13 atribut. Dari semua atribut, terdapat 3 variabel berupa teks dan string, seperti Provinsi, Kabupaten/Kota, dan Code, yang kemudian dikurangi. Pengurangan atribut tersebut dilakukan untuk menyederhanakan model serta memfokuskan analisis hanya pada variabel numerik, sehingga hasil prediksi yang dihasilkan dapat lebih optimal. Setelah dilakukan pengurangan, jumlah atribut yang digunakan dalam penelitian ini menjadi 10 variabel numerik.

### 3.2.3.6 Proses Utama

Proses analisis dengan memakai metode ANN dilakukan melalui beberapa tahapan. Pertama, ditentukan data yang akan digunakan sebagai input, kemudian ditetapkan jumlah *hidden layer* yang sesuai. Setelah itu, dipilih fungsi aktivasi pada model *backpropagation* untuk membantu proses pembelajaran jaringan. Tahap berikutnya adalah melakukan perhitungan bobot pada fase *feedforward* hingga diperoleh hasil perhitungan. Selanjutnya, hasil tersebut dianalisis untuk

menghasilkan output akhir yang kemudian dijadikan dasar dalam penarikan kesimpulan.

### 3.2.3.7 Implementasi Sistem

Setiap model ANN pada dasarnya dibangun dari node buatan. Mekanismenya mengikuti tiga aturan utama, yaitu perkalian, penjumlahan, dan aktivasi. Pada tahap awal, setiap data masukan dikalikan dengan bobot masing-masing. Hasil perkalian tersebut kemudian dijumlahkan bersama bias. Selanjutnya, nilai yang diperoleh diproses melalui fungsi aktivasi sebelum akhirnya diteruskan menjadi keluaran dari jaringan ANN.

### 3.2.3.8 Eksperimen

Tahap eksperimen dalam penelitian ini dilakukan menggunakan *Google Colab* untuk membangun dan menguji model *Artificial Neural Network* (ANN) dalam memprediksi angka kemiskinan. Setelah diperoleh hasil prediksi, langkah selanjutnya adalah melakukan analisis dengan cara membandingkan performa dari kedua arsitektur jaringan yang dirancang. Model terbaik dipilih berdasarkan tingkat kesalahan terkecil yang diukur dengan parameter *Root Mean Square Error* (RMSE).

Secara konseptual, ANN tidak memiliki aturan baku dalam penentuan arsitektur jaringan yang optimal. Oleh karena itu, perancangan arsitektur pada penelitian ini dilakukan secara bertahap dengan memodifikasi jumlah node dan hidden layer sesuai kebutuhan model. Setiap rancangan kemudian dievaluasi menggunakan algoritma backpropagation dengan tolok ukur nilai RMSE.

Dari semua atribut numerik yang tersedia dijadikan sebagai variabel input, sedangkan tingkat kemiskinan digunakan sebagai variabel target. Data penelitian ini dibagi menjadi dua kelompok, yaitu 80% data latih dan 20% data uji, sehingga model dapat diuji sekaligus dievaluasi performanya. Rincian pembagian data bisa diketahui dalam Tabel 3.9.

**Tabel 3. 9 Atribut Data Variable Target**

| No | Model | Atribut | Tipe Data      | Data Latih | Data Uji |
|----|-------|---------|----------------|------------|----------|
| 1  | A     | PPM     | <i>Numeric</i> | 411        | 103      |
| 2  | B     | RLSP    | <i>Numeric</i> |            |          |
| 3  | C     | PPKD    | <i>Numeric</i> |            |          |
| 4  | D     | IPM     | <i>Numeric</i> |            |          |
| 5  | E     | UHH     | <i>Numeric</i> |            |          |
| 6  | F     | PRTSL   | <i>Numeric</i> |            |          |
| 7  | G     | PRTAM   | <i>Numeric</i> |            |          |
| 8  | H     | TPT     | <i>Numeric</i> |            |          |
| 9  | I     | TPAK    | <i>Numeric</i> |            |          |
| 10 | J     | PDRB    | <i>Numeric</i> |            |          |

Selain membagi 10 satribut yang merupakan variabel target, pada riset ini juga melakukan variasi total node di lapisan tersembunyi (Tabel 3.8). Dengan demikian, didapatkan beberapa kombinasi atau pola arsitektur yang kemudian digunakan dalam tahapan latih dan uji (Tabel 3.9). Pola tersebut selanjutnya disajikan pada tabel berikut untuk memperjelas rancangan arsitektur yang diuji pada riset ini, bisa diketahui dalam tabel 3.10.

**Tabel 3. 10 Banyaknya Node Pada Masing-Masing Hidden Layer**

| Model | Node     | Arsitektur                            |
|-------|----------|---------------------------------------|
| 1     | 13-6-1   | 13 input – 6 node – 1 output          |
| 2     | 13-6-9-1 | 13 input – 6 node – 9 node – 1 output |

Penelitian ini melakukan uji coba sebanyak 20 kali pada setiap strategi sesuai dengan nama pelatihan yang telah ditentukan. Pengujian dilakukan menggunakan *Google Colab* dalam membangun model ANN berbasis algoritma *backpropagation* dalam memprediksi angka kemiskinan. Hasil prediksi kemudian dianalisis dengan membandingkannya terhadap nilai actual, sehingga diperoleh struktur kenodean yang paling tepat bisa dilihat pada tabel 3.11.

**Tabel 3. 11 Variasi Input pada Model yang dilatih**

| <b>Algoritma</b> | <b>Arsitektur</b> | <b>Variable Target</b> | <b>Nama Pelatihan</b> |
|------------------|-------------------|------------------------|-----------------------|
| BPN              | 1                 | A                      | BNP-1A                |
|                  |                   | B                      | BNP-1B                |
|                  |                   | C                      | BNP-1C                |
|                  |                   | D                      | BNP-1D                |
|                  |                   | E                      | BNP-1E                |
|                  |                   | F                      | BNP-1F                |
|                  |                   | G                      | BNP-1G                |
|                  |                   | H                      | BNP-1H                |
|                  |                   | I                      | BNP-1I                |
|                  |                   | J                      | BNP-1J                |
|                  | 2                 | A                      | BNP-2A                |
|                  |                   | B                      | BNP-2B                |
|                  |                   | C                      | BNP-2C                |
|                  |                   | D                      | BNP-2D                |
|                  |                   | E                      | BNP-2E                |
|                  |                   | F                      | BNP-2F                |
|                  |                   | G                      | BNP-2G                |
|                  |                   | H                      | BNP-2H                |
|                  |                   | I                      | BNP-2I                |
|                  |                   | J                      | BNP-2J                |

### 3.3 Instrumen Penelitian

Pada bagian ini, instrument penelitian berisi deskripsi mengenai parameter-parameter yang dijadikan acuan dalam proses analisis. Parameter tersebut mencakup variable independent, variable intervening sebagai penghubung, serta variable dependen. Adapun rincian secara lebih jelas terkait parameter yang digunakan bisa diketahui dalam Tabel 3.12.

**Tabel 3. 12 Instrumen pada Penelitian**

| <b>Variable Independen</b>  | <b>Main Process</b> | <b>Variable Intervening</b> | <b>Variable Dependen</b>          |
|---|---------------------|-----------------------------|-----------------------------------|
| PPM, RLSP,<br>PPKD, IPM,<br>UHH , PRTSL ,<br>PRTAM , TPT,<br>TPAK, PDRB | ANN                 | Prediksi Angka Kemiskinan   | Nilai Akurasi<br><i>MSE-based</i> |

Berdasarkan tabel diatas dapat dilihat bahwa Variabel independen dalam penelitian ini meliputi PPM, RLSP, PPKD, IPM, UHH , PRTSL , PRTAM , TPT, TPAK, PDRB. Selanjutnya variabel dependen yang merupakan hasil yang dipengaruhi oleh variabel independen tersebut. Menurut (Rahmawati, 2021) variabel dependen sering disebut juga variabel keluaran atau konsekuensi. Dalam penelitian ini, variabel dependen yang dianalisis adalah nilai akurasi berbasis MSE atau *MSE-based* untuk menilai kinerja ANN dalam memprediksi angka kemiskinan.

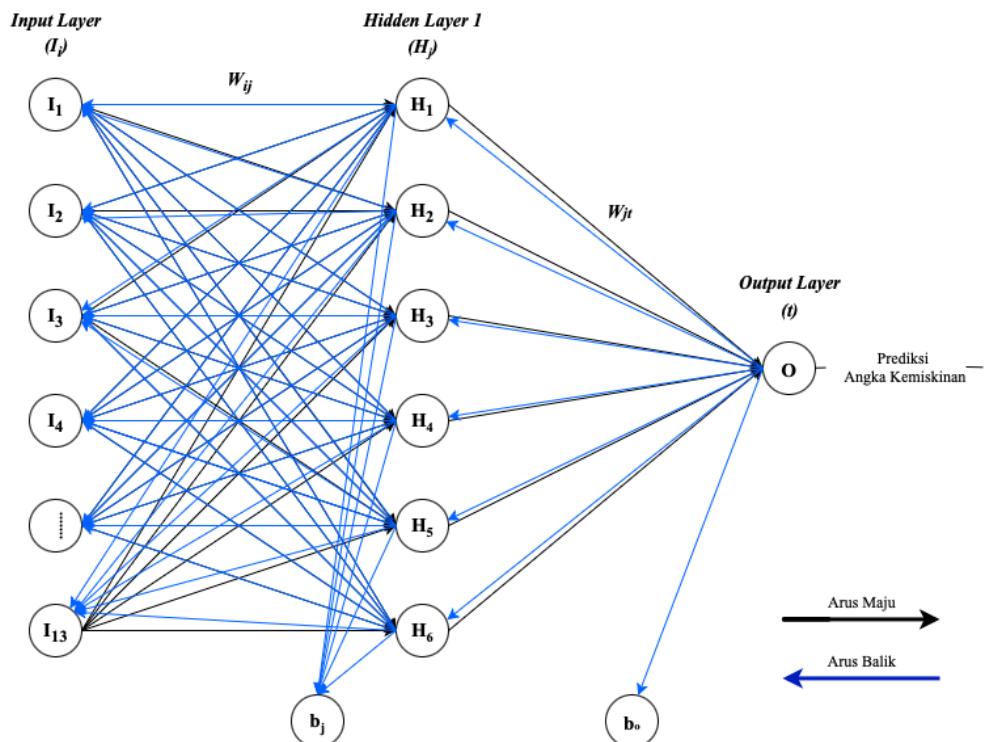
Variabel intervening atau variabel perantara adalah variabel yang secara teori berperan dalam menjembatani hubungan antara variabel independen dan variabel dependen (Widiastuti et al., 2022). Hubungan yang terbentuk menjadi tidak langsung karena pengaruh variabel independen disalurkan melalui variabel intervening ini. Dengan kata lain, variabel independen tidak memengaruhi variabel dependen secara langsung, melainkan melalui variabel perantara. Adapun variabel penghubung atau intervening pada penelitian ini yaitu prediksi angka kemiskinan.

## BAB IV

### MODEL ANN SATU HIDDEN LAYER

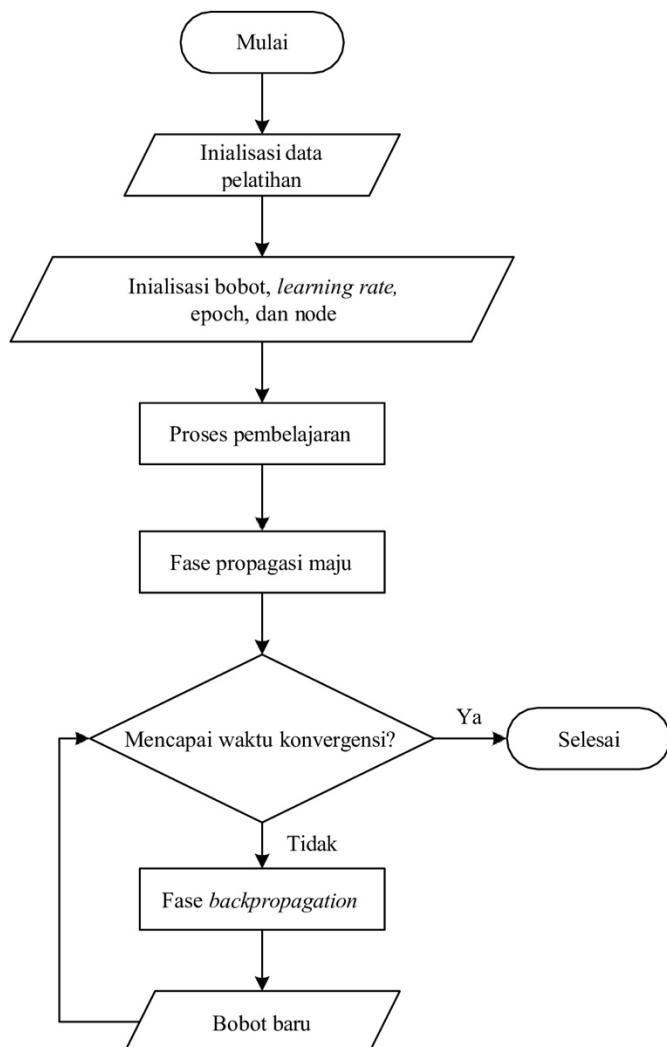
#### 4.1 Design

*Design* penelitian adalah proses yang dirancang untuk menjawab permasalahan yang diteliti. Melalui tahapan ini, objek penelitian dapat dipahami dengan lebih jelas dan menjadi dasar dalam menarik kesimpulan. Dalam penelitian ini, peneliti menggunakan Artificial Neural Network (ANN) dengan algoritma Backpropagation untuk memprediksi angka kemiskinan di Indonesia. Strategi awal dilakukan dengan membangun jaringan ANN yang memiliki satu hidden layer. Jaringan tersebut terdiri atas input berupa data kemiskinan ( $I_i$ ), bobot penghubung antara lapisan input dan lapisan tersembunyi ( $W_{ij}$ ), keluaran dalam lapisan tersembunyi dengan jumlah node yang sudah ditetapkan ( $H_j$ ), pada bobot penghubung menuju output ( $W_{jt}$ ), serta output berupa prediksi angka kemiskinan ( $O_t$ ). Lebih jelas bisa diketahui dalam Gambar 4.1.



**Gambar 5. 1 Rancangan Arsitektur Model ANN Dengan Lapisan Tersembunyi Pertama**

Strategi pertama dalam memprediksi tingkat kemiskinan dengan metode ANN dengan algoritma arus balik dirancang dengan satu hidden layer berkonfigurasi 13–6–1. Model ini bekerja dalam kerangka supervised learning, di mana proses pelatihan dipandu oleh dua metrik evaluasi, yaitu Mean Square Error (MSE) dan Mean Absolute Percentage Error (MAPE), yang berfungsi sebagai penentu penghentian proses sekaligus dasar penyesuaian bobot hingga mencapai nilai optimal. Dengan demikian, pelatihan akan berhenti secara otomatis saat konvergensi tercapai. Adapun alur pelatihan pada strategi pertama ini bisa diketahui dalam Gambar .2.



**Gambar 4. 1 Flowchart Proses Pelatihan**

1. Penyiapan data masukan
2. Penetapan parameter model
3. Tahap Forward Propagation
  - a. Menghitung sinyal pada setiap node di *hidden layer* ( $H_j$ ). Pada tahap ini, data dari variabel masukan ( $I_i$ ) diteruskan sebagai sinyal ke seluruh node pada hidden layer pertama.

$$H_{net\ j} = T_{0j} + \sum_{i=1}^n I_i T_{ij} \quad (1)$$

Dimana  $i$  node ke- $i$  ( $i = 1, 2, 3, \dots, n$ ) di dalam *input layer* dan  $j$  node ke- $j$  ( $j = 1, 2, 3, \dots, p$ ) di dalam *hidden layer* pertama.  $I_i$  merupakan nilai *input* di dalam node masukan  $i$  ke node tersembunyi  $j$ .  $T_{0j}$  merupakan bias di dalam *input layer* dan  $T_{ij}$  merupakan bobot di dalam node masukan  $i$  yang mengarah ke node  $j$ .

- b. Seluruh keluaran pada hidden layer pertama dihitung dengan fungsi ReLU. Nilai hasil perhitungan tersebut kemudian digunakan pada setiap node di hidden layer pertama untuk tahap selanjutnya.

$$H_j = f(H_{net\ j}) = \max(0, H_{net\ j}) \quad (2)$$

Dengan  $\max$  sebagai nilai maksimum antara 0 dan  $H_{net\ j}$ . Hasil perhitungan  $f(H_{net\ j})$  adalah nilai aktivasi di dalam node tersembunyi  $j$  yang kemudian didistribusikan ke semua node keluaran.

- c. Sesudah nilai didapatkan pada setiap lapisan tersembunyi, dihitung jumlah seluruh sinyal yang masuk ke keluaran ( $H_1, \dots, H_n$ ) dengan bobotnya.  $U_{0k}$  adalah nilai bias pada *hidden layer*,  $H_j$  adalah hasil yang keluar dari *hidden layer*, dan  $U_{jk}$  adalah bobot pada node tersembunyi  $j$  yang mengarah ke node keluaran  $k$  ( $k = 1, 2, 3, \dots, m$ ).

$$O_{net\ k} = U_{0k} + \sum_{j=1}^n H_j U_{jk} \quad (3)$$

- d. Berdasarkan pada fungsi aktivasi yaitu ReLU dalam melakukan penghitungan sinyal keluaran, adapun nilai yang dapatkan yaitu:

$$O_k = f(O_{net\ k}) = \max(0, O_{net\ k}) \quad (4)$$

$O_{net\ k}$  adalah hasil perolehan fungsi aktivasi, ialah total nilai dari node tersembunyi serta node *output*.

4. Sesudah didapatkan nilai output, selisih target dan nilai outputnya dihitung dengan rumus MSE dan MAPE.

$$MSE = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2 \quad (5)$$

$$MAPE = \sum_{T=1}^n \left| \frac{y_i - \hat{y}_i}{\hat{y}_i} \right| \times 100\% \quad (6)$$

Jika waktu konvergensi telah tercapai, maka proses akan terhenti. Sebaliknya jika waktu konvergensi belum tercapai, maka dilakukan proses backpropagation dengan memperbarui bobot

5. Tahap Backpropagation

- a. Jumlah nilai error ( $\delta_k$ ) di lapisan output ( $O_k$ ) yang menjadi tempat diterimanya pola yang ditargetkan ( $t_k$ ) lalu dikirim pada lapisan selanjutnya yang dipakai untuk melakukan perhitungan bobot & koreksi bias antara *input layer* serta *hidden layer*.

$$\delta_k = (T_k - O_k) f'(O_{net\ k}) \quad (7)$$

yaitu:

$$f'(O_{net\ k}) = (1 - f(O_{net\ k}))f(O_{net\ k}) \quad (8)$$

- b. Sesudah nilai error di *hidden layer* pertama serta lapisan keluaran didapatkan, dilanjutkan dengan melakukan perhitungan kesalahan terhadap lapisan masukan dan lapisan tersembunyi pertama.

$$\delta_j = (\delta_{net\ j} f'(O_{net\ j})) \quad (9)$$

yaitu:

$$\delta_{net\ j} = \sum_{k=1}^m \delta_k U_{jk} \quad (10)$$

$$f'(H_{net\ j}) = (1 - f(H_{net\ j}))f(H_{net\ j}) \quad (12)$$

- c. Sesudah skor *error* pada setiap layer telah diketahui, maka total koreksi atau perubahan yang bisa ditambah pada setiap bobot lama bisa dilakukan perhitungan.

$$\Delta T_{ij} = \alpha \delta_j I_i \quad (13)$$

$$\Delta U_{jk} = \alpha \delta_k H_j \quad (14)$$

$\Delta T_{ij}$  &  $\Delta U_{jk}$  merupakan perubahan yang bisa ditambah terhadap bobot yang lama bisa dilakukan perbaharuan berdasarkan  $\alpha$  (learning rate),  $\delta_j$

merupakan kesalahan dari lapisan masukan dan lapisan tersembunyi, serta  $\delta_k$  merupakan kesalahan antara *hidden layer* dan lapisan keluaran.  $I_i$  adalah input yang berasal dari node  $i$  ke node  $j$ , sedangkan  $H_j$  merupakan hasil aktivasi fungsi dari *hidden layer*.

#### 6. Tahap Perubahan Bobot

Sesudah  $\Delta T_{ij}$  &  $\Delta U_{jk}$  didapatkan, tahapan berikutnya yaitu melakukan perbaikan nilai atau bobot yang lama untuk menjadi bobot yang baru.

$$T_{ij}(\text{new}) = T_{ij}(\text{old}) + \Delta T_{ij} \quad (15)$$

$$U_{jk}(\text{new}) = U_{jk}(\text{old}) + \Delta U_{jk} \quad (16)$$

$T_{ij}(\text{new})$  adalah bobot baru antara *input layer* dan *hidden layer* pertama yang akan dicari, sedangkan  $T_{ij}(\text{old})$  adalah bobot lama yang akan diperbarui.  $U_{jk}(\text{new})$  adalah bobot baru antara *hidden layer* pertama dan lapisan keluaran yang akan dicari, sedangkan  $U_{jk}(\text{old})$  adalah bobot lama yang akan diperbarui.

#### 4.2 Implementasi

Program dibangun menggunakan platform Kaggle dengan arsitektur satu *hidden layer* berisi 13 node. Pada tahap evaluasi, proses optimasi dilakukan dengan mengacu pada nilai MSE terkecil selama pengujian. Setiap atribut digunakan secara bergantian untuk menjadi variabel yang di targetkan, sekaligus ditentukan total node dalam *hidden layer* yang merujuk pada pendekatan (Sahi & Galib, 2025) dengan menggunakan jumlah digit terbanyak dalam dataset sebagai acuan. Rincian parameter uji bisa diketahui dalam Tabel 4.1.

**Tabel 4. 1 Indikator/Parameter/Atribut Model**

|                          |                       |
|--------------------------|-----------------------|
| Tipe Jaringan            | BPN-ANN               |
| Fungsi Aktivasi          | ReLU                  |
| Optimizer                | Adam                  |
| Persamaan                | MSE, MAPE             |
| <i>Input Layer</i>       | 13                    |
| <i>Hidden Layer</i>      | 1 <i>Hidden Layer</i> |
| Node <i>Hidden Layer</i> | 6                     |
| <i>Output Layer</i>      | 1                     |
| Learning Rate            | 0,2                   |

### 4.3 Uji Coba

Pelatihan model dilakukan dengan memakai data yang tersedia, dengan proporsi 80% data latih serta 20% data uji. Data yang tersedia digunakan dalam proses evaluasi model. Pada tahap pelatihan, setiap parameter secara bergantian menjadi variabel target, selain parameter Provinsi, Kab/Kota, dan *Code*.

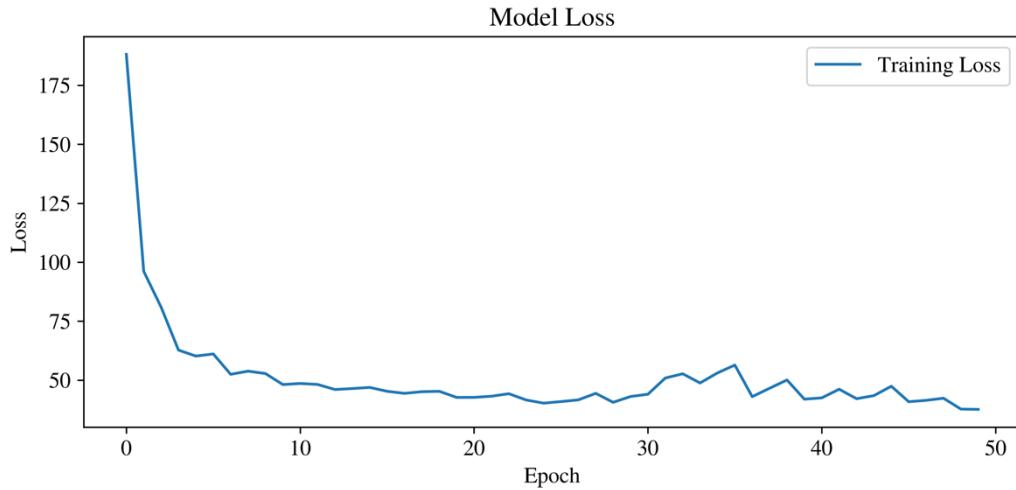
#### 4.3.1 Pelatihan BPN-1A

Pelatihan tersebut memperoleh hasil dari BPN-1A memakai parameter target atribut PPM untuk variable target disajikan dalam Tabel 4.2.

**Tabel 4. 2 Tahap Pelatihan BPN-1A**

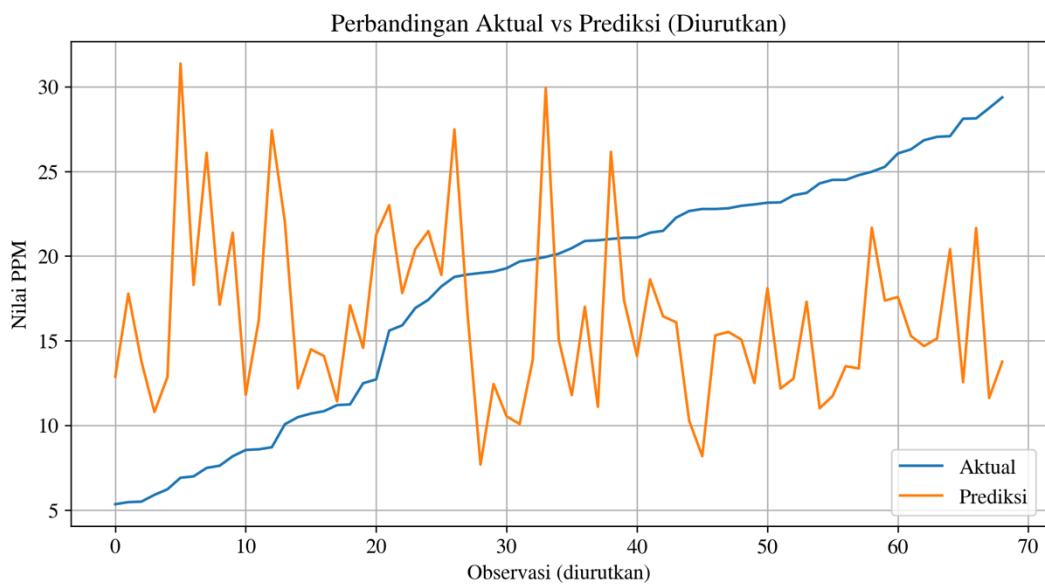
| <b><i>Epoch</i></b> | <b><i>Time</i></b> | <b><i>Training Loss</i></b> | <b><i>Validation Loss</i></b> |
|---------------------|--------------------|-----------------------------|-------------------------------|
| 1                   | 2s 201ms           | 7.152.238                   | 4.768.136                     |
| 2                   | 0s 12ms            | 3.083.944                   | 4.141.344                     |
| 3                   | 0s 13ms            | 2.638.846                   | 4.644.347                     |
| 4                   | 0s 12ms            | 2.425.899                   | 4.515.760                     |
| 5                   | 0s 11ms            | 2.324.124                   | 4.318.277                     |
| ...                 | ...                | ...                         | ...                           |
| 46                  | 0s 12ms            | 1.626.743                   | 3.553.773                     |
| 47                  | 0s 12ms            | 1.571.779                   | 3.500.410                     |
| 48                  | 0s 12ms            | 1.623.385                   | 3.593.774                     |
| 49                  | 0s 13ms            | 1.758.834                   | 3.631.966                     |
| 50                  | 0s 13ms            | 1.629.513                   | 4.028.017                     |

Berdasarkan Tabel 4.2, diketahui bahwa pelatihan model BPN-1A berlangsung dengan waktu eksekusi sekitar 13 ms dengan MSE senilai 29,19 dan MAPE senilai 12,87%. Proses pelatihan tersebut menghasilkan nilai *Training Loss* sebesar 1.629.513 dan nilai *Validation Loss* sebesar 4.028.017. Hasil dari pelatihan divisualisasikan berupa grafik, bisa diketahui dalam Gambar 4.1.



**Gambar 4. 2 Pelatihan BPN-1A**

Pada gambar 4.2 memperlihatkan grafik *training loss* pada model prediksi angka kemiskinan. Grafik menunjukkan penurunan loss yang tajam pada epoch awal, menandakan model mulai mempelajari pola dasar. Namun, pada epoch berikutnya loss berfluktuasi dan belum stabil, menunjukkan pembelajaran belum optimal dan model belum menangkap hubungan antar variabel secara kuat, kemungkinan karena keterbatasan arsitektur atau kompleksitas data.



**Gambar 4. 3 Perbandingan Data Aktual dan Prediksi BPN-1A**

Pada gambar 4.3 diatas, menampilkan perbedaan dari nilai aktual dan hasil prediksi dari model di 50 data pertama. Terlihat bahwa garis prediksi (warna oranye) bergerak searah dengan data aktual (warna biru), yang menunjukkan adanya kemampuan model dalam memahami pola pergerakan data. Namun, masih terdapat sejumlah deviasi yang menyebabkan hasil prediksi belum sepenuhnya identik dengan nilai aktual. Dengan demikian, grafik ini memperlihtakan bahwasanya model bisa mengikuti arah pergerakan data aktual secara umum, meskipun terdapat selisih nilai pada beberapa titik pengamatan.

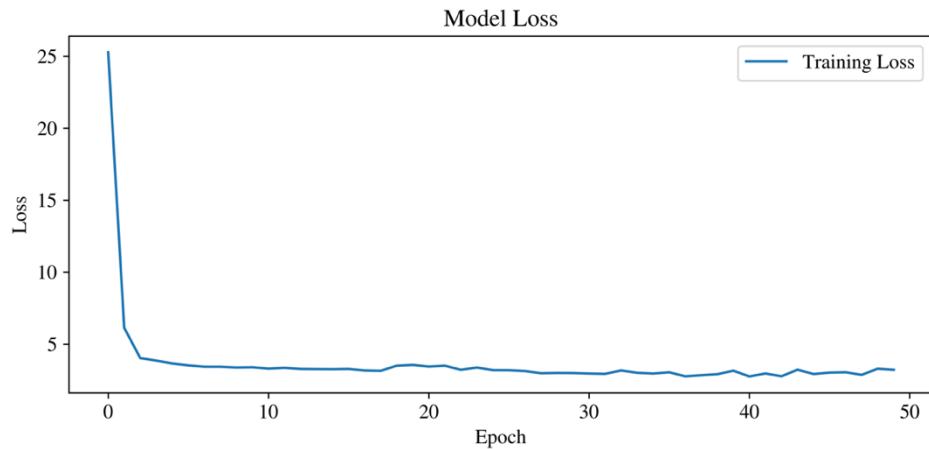
#### 4.3.2 Pelatihan BPN-1B

Hasil pada pelatihan BPN-1B memakai parameter RLSP menjadi variable target disajikan dalam Tabel 4.3

**Tabel 4. 3 Tahap Pelatihan BPN-1B**

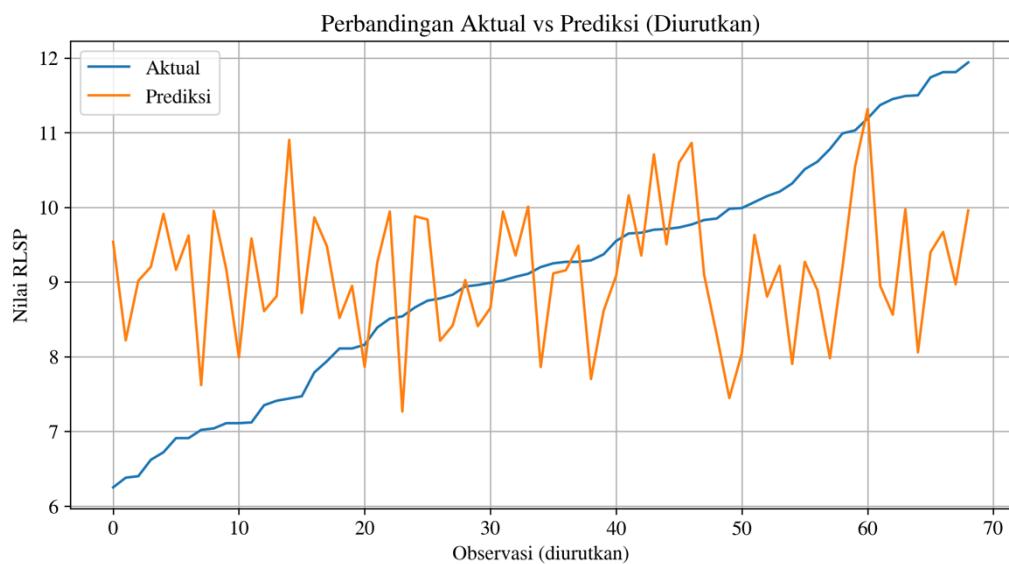
| <b><i>Epoch</i></b> | <b><i>Time</i></b> | <b><i>Training Loss</i></b> | <b><i>Validation Loss</i></b> |
|---------------------|--------------------|-----------------------------|-------------------------------|
| 1                   | 2s 195ms           | 2.527.408                   | 3.912.703                     |
| 2                   | 0s 12ms            | 6.130.209                   | 4.457.743                     |
| 3                   | 0s 12ms            | 4.026.125                   | 3.698.637                     |
| 4                   | 0s 13ms            | 3.852.706                   | 3.329.912                     |
| 5                   | 0s 13ms            | 3.653.929                   | 3.092.618                     |
| ...                 | ...                | ...                         | ...                           |
| 46                  | 0s 12ms            | 3.018.663                   | 4.330.347                     |
| 47                  | 0s 12ms            | 3.045.115                   | 4.709.894                     |
| 48                  | 0s 12ms            | 2.862.271                   | 4.916.405                     |
| 49                  | 0s 11ms            | 3.288.305                   | 4.255.150                     |
| 50                  | 0s 12ms            | 3.213.449                   | 4.186.967                     |

Berdasarkan Tabel 4.3, diketahui bahwa pelatihan model BPN-1B berlangsung dengan waktu eksekusi sekitar 12 ms dengan MSE senilai 4,9 dan MAPE senilai 3,22%. Proses pelatihan tersebut menghasilkan nilai *Training Loss* sebesar 3.213.449 dan nilai *Validation Loss* sebesar 4.186.967. Hasil pelatihan divisualisasikan dalam bentuk grafik, bisa diketahui dalam Gambar 4.3



**Gambar 4. 4 Skor Loss Pelatihan BPN-1B**

Pada Gambar 4.3 memperlihatkan grafik nilai *Training Loss* pada total epoch. Di awal pelatihan, skor *loss* cukup tinggi, namun mengalami penurunan yang sangat tajam pada epoch-epoch awal. Setelah itu, grafik *loss* terlihat menurun secara bertahap dan kemudian berada pada kondisi relatif stabil di kisaran nilai rendah hingga akhir epoch. Pola ini memperlihatkan tahapan pelatihan yang berjalan dengan baik, karena model mampu meminimalkan *loss* seiring bertambahnya epoch, serta tidak menunjukkan indikasi peningkatan *loss* yang signifikan di akhir pelatihan.



**Gambar 4. 5 Perbandingan Data Aktual dan Prediksi BPN-1B**

Dalam gambar 4.2 tersebut, menampilkan komparasi antara nilai faktual dengan hasil prediksi suatu model pada 50 data pertama. Terlihat bahwa garis prediksi cenderung mengikuti arah perubahan data aktual, baik ketika nilai mengalami kenaikan maupun penurunan. Hal ini menunjukkan bahwa model telah mampu menangkap pola umum dari data yang dianalisis. Walaupun demikian, masih terlihat beberapa perbedaan nilai antara prediksi dan data aktual, terutama pada titik-titik dengan perubahan yang cukup tajam. Dengan kata lain, model sudah bekerja dengan baik dalam menggambarkan tren secara umum, meskipun masih ada ketidakakuratan pada sebagian kasus individu.

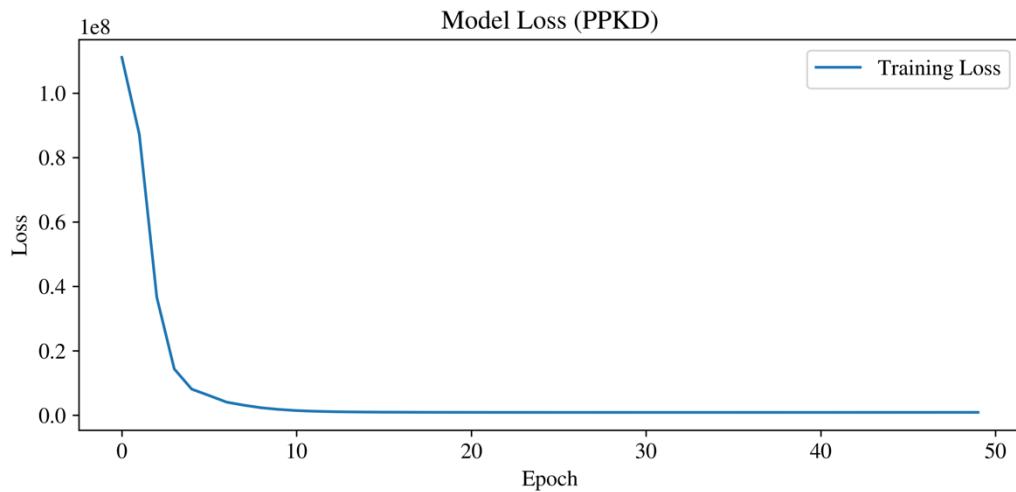
#### 4.3.3 Pelatihan BPN-1C

Hasil pelatihan BPN-1C memakai atribut PPKD untuk variabel target disajikan dalam Tabel 4.4.

**Tabel 4. 4 Proses Pelatihan BPN-1C**

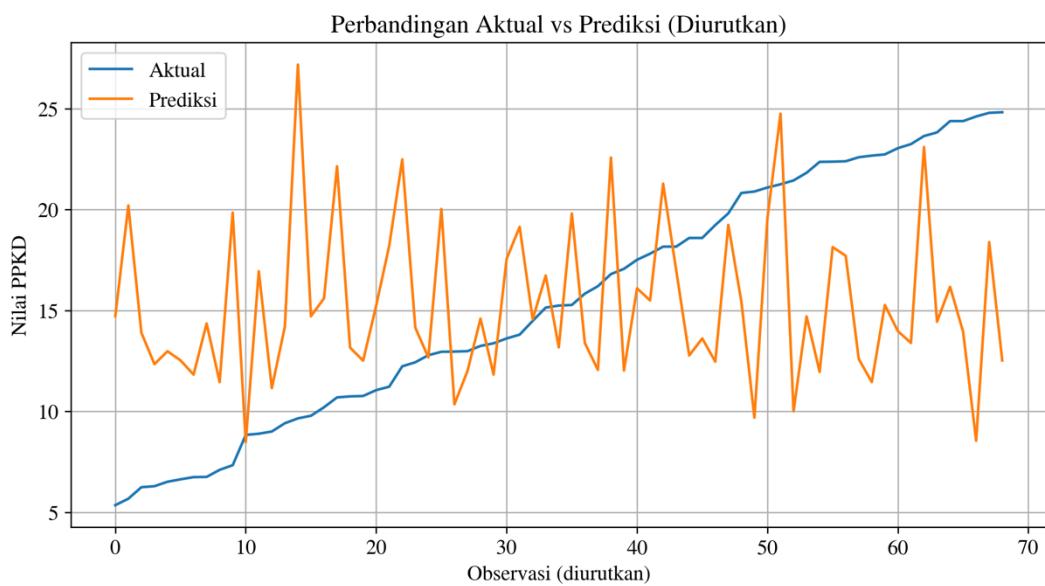
| <b><i>Epoch</i></b> | <b><i>Time</i></b> | <b><i>Training Loss</i></b> | <b><i>Validation Loss</i></b> |
|---------------------|--------------------|-----------------------------|-------------------------------|
| 1                   | 2s 208ms           | 1.111.049                   | 1.106.680                     |
| 2                   | 0s 12ms            | 8.258.076                   | 4.772.937                     |
| 3                   | 0s 13ms            | 2.733.583                   | 2.113.861                     |
| 4                   | 0s 12ms            | 1.259.817                   | 1.453.860                     |
| 5                   | 0s 13ms            | 8.030.601                   | 7.224.213                     |
| ...                 | ...                | ...                         | ...                           |
| 46                  | 0s 13ms            | 9.257.245                   | 1.108.343                     |
| 47                  | 0s 13ms            | 9.274.566                   | 1.107.318                     |
| 48                  | 0s 12ms            | 9.288.993                   | 1.103.736                     |
| 49                  | 0s 13ms            | 9.305.541                   | 1.103.041                     |
| 50                  | 0s 13ms            | 9.332.059                   | 1.103.341                     |

Berdasarkan Tabel 4.4, diketahui bahwa pelatihan model BPN-1C berlangsung dengan waktu eksekusi sekitar 13 ms dengan MSE senilai 1085070,25 dan MAPE senilai 999,98%. Proses pelatihan tersebut menghasilkan nilai *Training Loss* sebesar 9.332.059 dan nilai *Validation Loss* sebesar 1.103.341. Hasil pelatihan divisualisasikan dalam bentuk grafik, bisa diketahui dalam Gambar 4.5.



**Gambar 4. 6 Nilai Loss Pelatihan BPN-1C**

Pada Gambar 4.5 terlihat grafik Training Loss yang pada awal pelatihan bernilai sangat tinggi kemudian menurun tajam pada epoch-epoch awal. Setelah itu, penurunan berlangsung lebih lambat hingga akhirnya stabil pada nilai yang rendah mulai sekitar epoch ke-15. Pola ini menunjukkan bahwa model berhasil melakukan proses pelatihan dengan baik karena loss dapat diminimalkan dan tetap stabil hingga akhir epoch.



**Gambar 4. 7 Perbandingan Data Aktual dan Prediksi BPN-1C**

Pada Gambar 4.6 terlihat bahwa pola prediksi cukup mengikuti pola data aktual meskipun terdapat perbedaan nilai pada beberapa titik. Tren yang dihasilkan model sudah mampu merepresentasikan data dengan baik. Hal tersebut memperlihatkan performa model bisa dikatakan cukup baik meski masih terdapat sedikit penyimpangan.

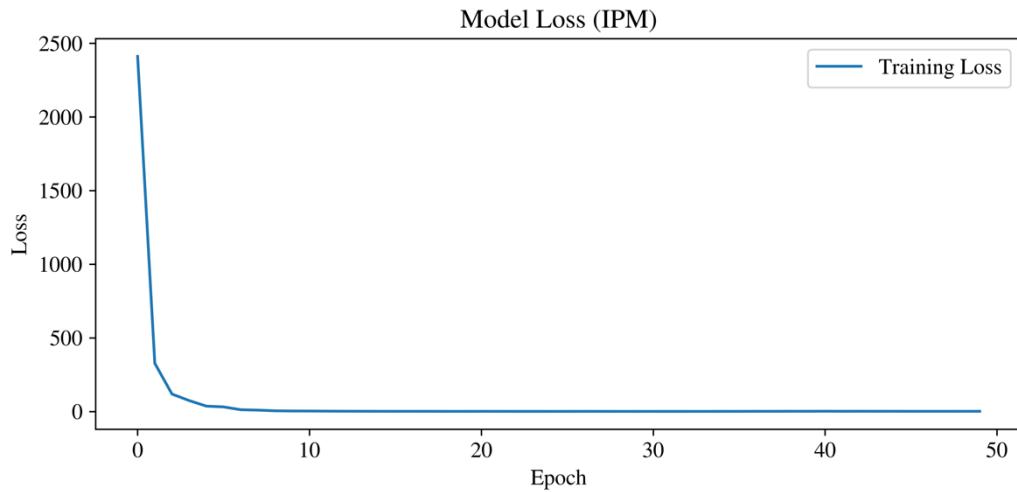
#### 4.3.4 Pelatihan BPN-1D

Hasil yang diperoleh dari proses pelatihan BPN-1D memakai atribut IPM dijabarkan dalam pada Tabel 4.5.

**Tabel 4. 5 Proses Pelatihan BPN-1D**

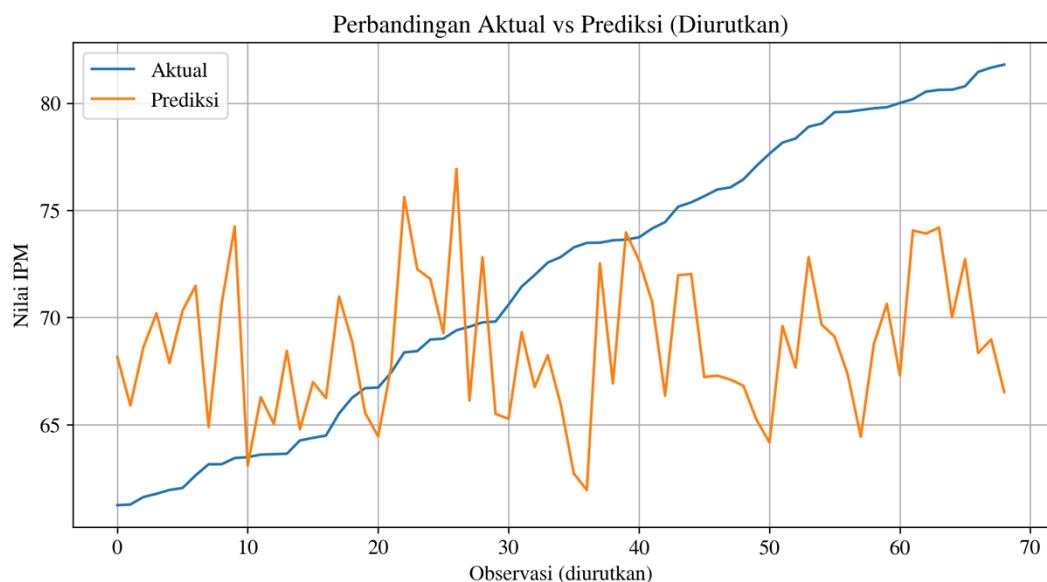
| Epoch | Time     | Training Loss | Validation Loss |
|-------|----------|---------------|-----------------|
| 1     | 2s 235ms | 2.411.532     | 5.182.910       |
| 2     | 0s 13ms  | 3.274.241     | 1.380.742       |
| 3     | 0s 15ms  | 1.186.895     | 1.555.335       |
| 4     | 0s 15ms  | 7.445.098     | 6.072.338       |
| 5     | 0s 13ms  | 3.670.745     | 5.140.309       |
| ...   | ...      | ...           | ...             |
| 46    | 0s 14ms  | 1.358.816     | 3.374.431       |
| 47    | 0s 13ms  | 1.442.306     | 3.460.105       |
| 48    | 0s 15ms  | 1.459.621     | 4.818.604       |
| 49    | 0s 15ms  | 1.511.617     | 5.617.855       |
| 50    | 0s 16ms  | 1.624.245     | 8.015.356       |

Berdasarkan Tabel 4.5, diketahui bahwa pelatihan model BPN-1D berlangsung dengan waktu eksekusi sekitar 16 ms dengan MSE senilai 4,4 dan MAPE senilai 2,95%. Proses pelatihan tersebut menghasilkan nilai *Training Loss* sebesar 1.624.245 dan nilai *Validation Loss* sebesar 8.015.356. Hasil pelatihan divisualisasikan dalam bentuk grafik, dapat diketahui pada Gambar 4.7.



**Gambar 4. 8 Nilai Loss Pelatihan BPN-1D**

Pada Gambar 4.7 diperlihatkan grafik Training Loss terhadap jumlah epoch. Pada awal pelatihan, nilai loss berada pada tingkat yang sangat tinggi, kemudian mengalami penurunan tajam pada epoch-epoch awal. Selanjutnya, nilai loss menurun secara bertahap hingga mencapai kondisi stabil pada nilai yang rendah, menunjukkan bahwa proses pelatihan berjalan dengan baik, di mana model mampu meminimalkan kesalahan prediksi secara konsisten.



**Gambar 4. 9 Perbandingan Data Aktual dan Prediksi BPN-1D**

Pada Gambar 4.8 diperlihatkan perbandingan antara data aktual dengan hasil prediksi. Secara umum, pola prediksi mampu mengikuti pola data aktual dengan cukup baik, meskipun terdapat beberapa perbedaan nilai pada titik tertentu. Hal ini menunjukkan bahwa model telah mampu merepresentasikan tren data secara konsisten, meski masih terdapat sedikit penyimpangan.

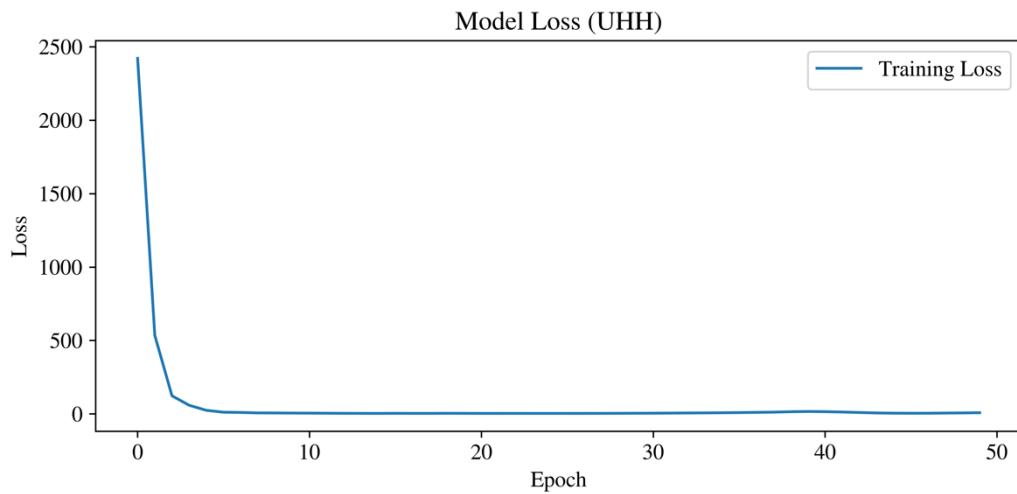
#### **4.3.5 Pelatihan BPN-1E**

Hasil yang diperoleh dari proses pelatihan BPN-1E memakai atribut UHH dijabarkan dalam pada Tabel 4.6.

**Tabel 4. 6 Proses Pelatihan BPN-1E**

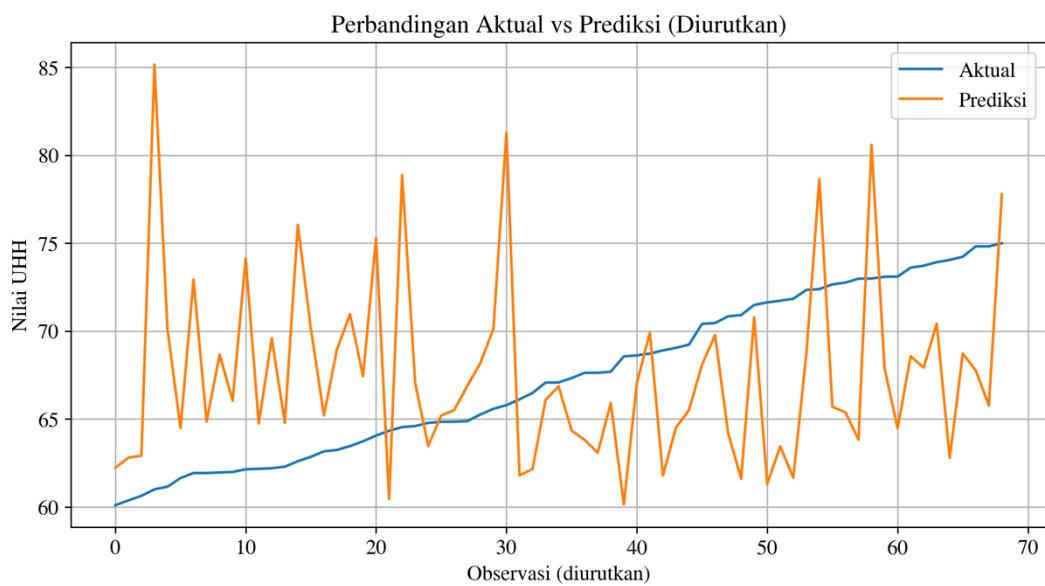
| Epoch | Time     | Training Loss | Validation Loss |
|-------|----------|---------------|-----------------|
| 1     | 2s 204ms | 2.420.913     | 8.506.942       |
| 2     | 0s 12ms  | 5.342.900     | 2.650.615       |
| 3     | 0s 12ms  | 1.229.381     | 1.203.678       |
| 4     | 0s 12ms  | 5.841.912     | 7.991.656       |
| 5     | 0s 12ms  | 2.399.945     | 2.745.620       |
| ...   | ...      | ...           | ...             |
| 46    | 0s 12ms  | 3.993.222     | 6.350.063       |
| 47    | 0s 12ms  | 4.190.309     | 9.899.504       |
| 48    | 0s 12ms  | 5.362.925     | 1.261.981       |
| 49    | 0s 12ms  | 6.379.387     | 1.111.968       |
| 50    | 0s 12ms  | 7.648.265     | 1.382.509       |

Berdasarkan Tabel 4.6, diketahui bahwa pelatihan model BPN-1D berlangsung dengan waktu eksekusi sekitar 12 ms dengan MSE senilai 10,79 dan MAPE senilai 5,41%. Proses pelatihan tersebut menghasilkan nilai *Training Loss* sebesar 7.648.265 dan nilai *Validation Loss* sebesar 1.382.509. Hasil pelatihan divisualisasikan dalam bentuk grafik, bisa diketahui dalam Gambar 4.9.



**Gambar 4. 10 Nilai Loss Pelatihan BPN-1E**

Pada Gambar 4.9 diperlihatkan grafik *Training Loss* terhadap jumlah epoch. Pada awal pelatihan, nilai loss berada pada tingkat yang sangat tinggi, kemudian mengalami penurunan tajam pada *epoch-epoch* awal. Selanjutnya, nilai loss menurun secara bertahap hingga mencapai kondisi stabil pada nilai yang rendah, yang menunjukkan bahwa proses pelatihan berjalan dengan baik dan model mampu meminimalkan kesalahan prediksi secara konsisten.



**Gambar 4. 11 Perbandingan Data Aktual dan Prediksi BPN-1E**

Pada Gambar 4.10 diperlihatkan perbandingan antara data aktual dengan hasil prediksi. Secara umum, pola prediksi mengikuti tren data aktual meskipun terdapat selisih pada beberapa titik pengamatan. Hal ini menunjukkan bahwa model telah mampu menangkap pola utama data, namun masih terdapat sedikit penyimpangan pada hasil prediksi.

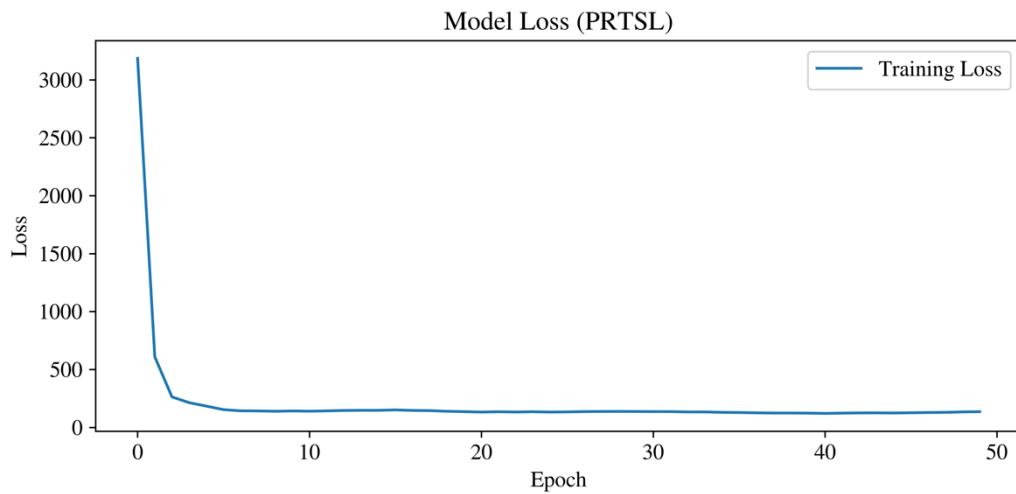
#### **4.3.6 Pelatihan BPN-1F**

Hasil yang diperoleh dari proses pelatihan BPN-1F memakai atribut PRTSL dijabarkan dalam pada Tabel 4.7.

**Tabel 4. 7 Proses Pelatihan BPN-1F**

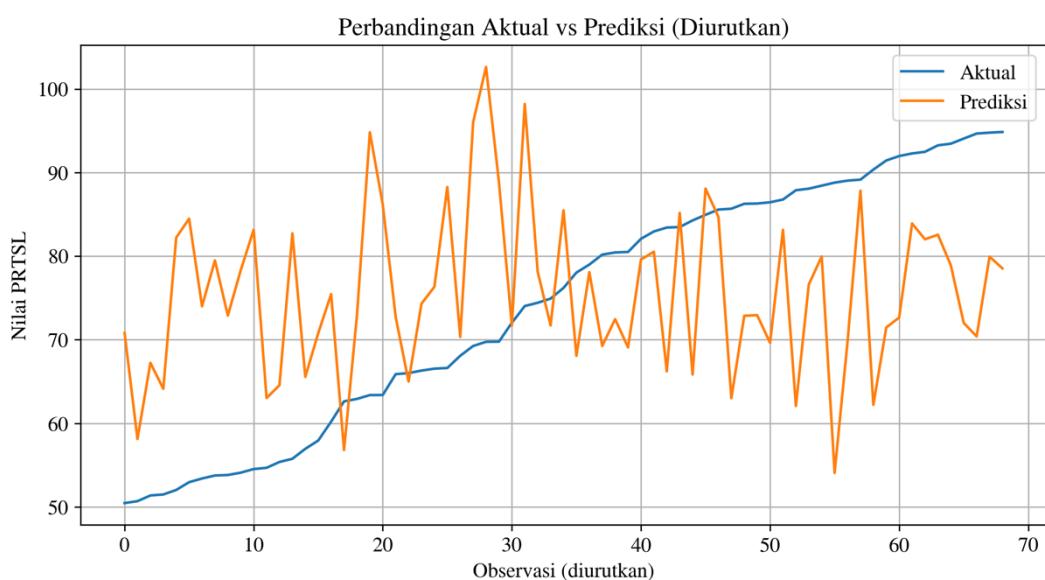
| <b>Epoch</b> | <b>Time</b> | <b>Training Loss</b> | <b>Validation Loss</b> |
|--------------|-------------|----------------------|------------------------|
| 1            | 2s 238ms    | 3.184.806            | 8.549.263              |
| 2            | 0s 12ms     | 6.083.233            | 4.664.186              |
| 3            | 0s 12ms     | 2.627.611            | 2.121.618              |
| 4            | 0s 13ms     | 2.132.459            | 2.776.656              |
| 5            | 0s 13ms     | 1.833.854            | 2.267.288              |
| ...          | ...         | ...                  | ...                    |
| 46           | 0s 12ms     | 1.262.052            | 3.297.103              |
| 47           | 0s 13ms     | 1.280.651            | 3.337.054              |
| 48           | 0s 13ms     | 1.292.961            | 3.409.234              |
| 49           | 0s 13ms     | 1.333.945            | 3.552.430              |
| 50           | 0s 14ms     | 1.353.116            | 3.426.347              |

Berdasarkan Tabel 4.7, diketahui bahwa pelatihan model BPN-1D berlangsung dengan waktu eksekusi sekitar 14 ms dengan MSE senilai 195,18 dan MAPE senilai 22,65%. Proses pelatihan tersebut menghasilkan nilai *Training Loss* sebesar 1.353.116 dan nilai *Validation Loss* sebesar 3.426.347. Hasil pelatihan divisualisasikan dalam bentuk grafik, bisa diketahui dalam Gambar 4.11.



**Gambar 4. 12 Nilai Loss Pelatihan BPN-1F**

Dalam Gambar 4.11 diperlihatkan grafik yaitu *Training Loss* pada total epoch. Di pelatihan pertama, jumlah loss berada pada tingkat yang sangat tinggi, kemudian mengalami penurunan tajam pada *epoch-epoch* awal. Selanjutnya, nilai loss menurun secara bertahap dan mencapai kondisi stabil pada nilai yang rendah, yang menunjukkan bahwa proses pelatihan berjalan dengan baik dan model mampu meminimalkan kesalahan prediksi secara konsisten.



**Gambar 4. 13 Perbandingan Data Aktual dan Prediksi BPN-1F**

Pada Gambar 4.12 diperlihatkan perbandingan antara data aktual dengan hasil prediksi. Secara keseluruhan, pola prediksi mengikuti tren data aktual, meskipun pada beberapa titik terdapat perbedaan nilai yang cukup terlihat. Hal ini menunjukkan bahwa model mampu merepresentasikan pola utama data, meski masih terdapat sedikit penyimpangan pada hasil prediksi.

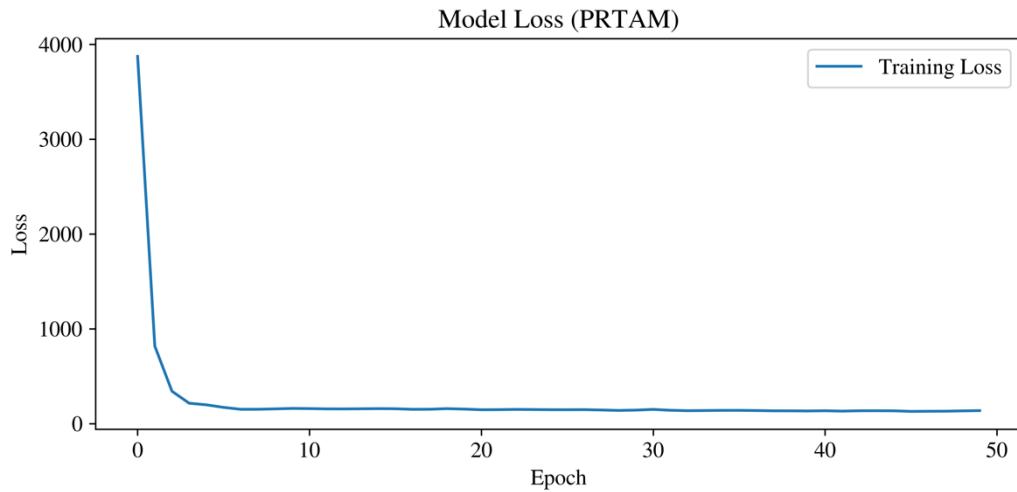
#### **4.3.7 Pelatihan BPN-1G**

Hasil yang diperoleh dari proses pelatihan BPN-1G memakai atribut PRTAM dijabarkan dalam pada Tabel 4.8.

**Tabel 4. 8 Proses Pelatihan BPN-1G**

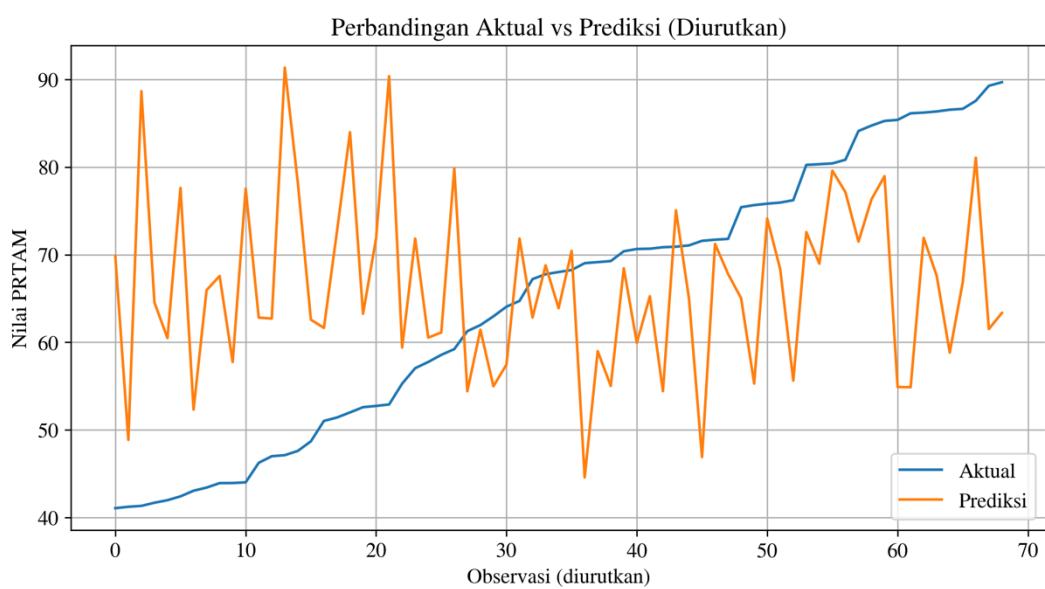
| Epoch | Time     | Training Loss | Validation Loss |
|-------|----------|---------------|-----------------|
| 1     | 2s 197ms | 3.873.781     | 1.078.547       |
| 2     | 0s 12ms  | 8.181.962     | 6.279.843       |
| 3     | 0s 13ms  | 3.418.394     | 3.065.466       |
| 4     | 0s 12ms  | 2.155.616     | 4.326.460       |
| 5     | 0s 12ms  | 1.987.065     | 3.210.522       |
| ...   | ...      | ...           | ...             |
| 46    | 0s 13ms  | 1.292.039     | 3.598.759       |
| 47    | 0s 12ms  | 1.306.114     | 3.526.720       |
| 48    | 0s 12ms  | 1.314.916     | 3.601.425       |
| 49    | 0s 12ms  | 1.348.820     | 3.660.886       |
| 50    | 0s 12ms  | 1.376.049     | 3.446.727       |

Berdasarkan Tabel 4.8, diketahui bahwa pelatihan model BPN-1D berlangsung dengan waktu eksekusi sekitar 12 ms dengan MSE senilai 279,65 dan MAPE senilai 25,37%. Proses pelatihan tersebut menghasilkan nilai *Training Loss* sebesar 1.376.049 dan nilai *Validation Loss* sebesar 3.446.727. Hasil pelatihan divisualisasikan dalam bentuk grafik, bisa diketahui dalam Gambar 4.13.



**Gambar 4. 14 Nilai Loss Pelatihan BPN-1G**

Pada Gambar 4.13 diperlihatkan grafik *Training Loss* terhadap jumlah epoch. Pada awal pelatihan, nilai loss berada pada tingkat yang sangat tinggi, kemudian mengalami penurunan tajam pada epoch-epoch awal. Selanjutnya, nilai loss menurun secara bertahap hingga mencapai kondisi stabil pada nilai yang rendah, menunjukkan bahwa proses pelatihan berjalan dengan baik dan model mampu meminimalkan kesalahan prediksi secara konsisten.



**Gambar 4. 15 Perbandingan Data Aktual dan Prediksi BPN-1G**

Dalam Gambar 4.14 diperlihatkan komparasi dari data faktual dengan hasil perkiraan. Secara umum, pola prediksi mengikuti tren data aktual dengan cukup baik, meskipun terdapat beberapa perbedaan nilai pada titik tertentu. Hal ini menunjukkan bahwa model mampu menangkap pola utama data, meski masih terdapat sedikit penyimpangan pada hasil prediksi.

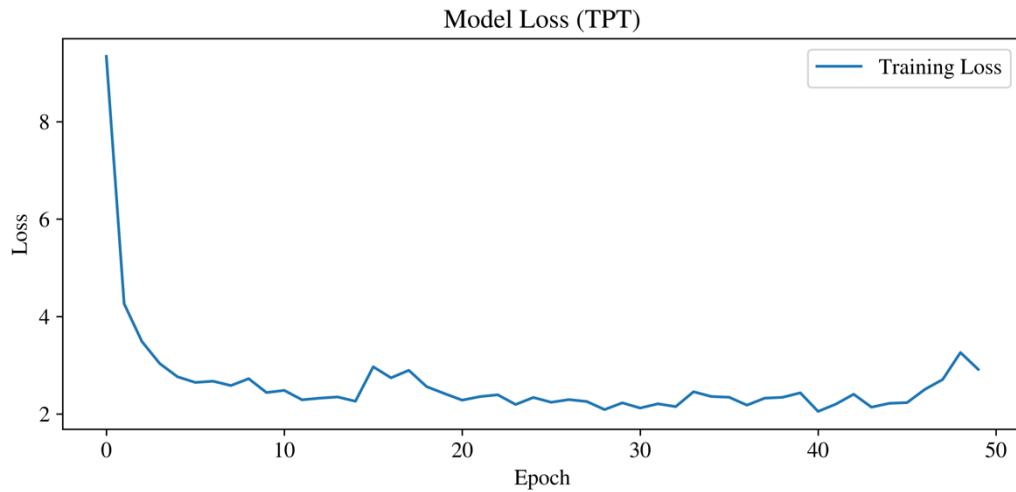
#### **4.3.8 Pelatihan BPN-1H**

Hasil yang diperoleh dari proses pelatihan BPN-1H memakai atribut TPT dijabarkan dalam pada Tabel 4.9.

**Tabel 4. 9 Proses Pelatihan BPN-1H**

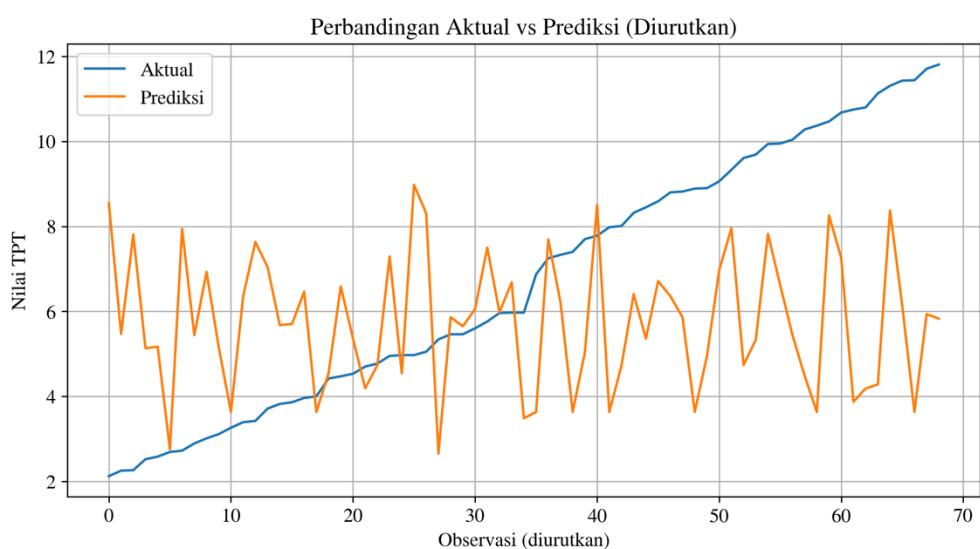
| <b>Epoch</b> | <b>Time</b> | <b>Training Loss</b> | <b>Validation Loss</b> |
|--------------|-------------|----------------------|------------------------|
| 1            | 2s 198ms    | 9.340.634            | 9.926.241              |
| 2            | 0s 13ms     | 4.262.479            | 3.964.404              |
| 3            | 0s 12ms     | 3.485.963            | 6.657.369              |
| 4            | 0s 13ms     | 3.034.373            | 4.307.104              |
| 5            | 0s 12ms     | 2.762.319            | 4.315.029              |
| ...          | ...         | ...                  | ...                    |
| 46           | 0s 14ms     | 2.231.799            | 1.005.121              |
| 47           | 0s 13ms     | 2.503.397            | 5.193.683              |
| 48           | 0s 15ms     | 2.706.036            | 1.054.214              |
| 49           | 0s 13ms     | 3.259.799            | 4.772.987              |
| 50           | 0s 14ms     | 2.914.044            | 9.114.474              |

Berdasarkan Tabel 4.9, diketahui bahwa pelatihan model BPN-1D berlangsung dengan waktu eksekusi sekitar 14 ms dengan MSE senilai 4,57 dan MAPE senilai 2,73%. Proses pelatihan tersebut menghasilkan nilai *Training Loss* sebesar 2.914.044 dan nilai *Validation Loss* sebesar 9.114.474. Hasil pelatihan divisualisasikan dalam bentuk grafik, bisa dilihat pada Gambar 4.15.



**Gambar 4. 16 Nilai Loss Pelatihan BPN-1H**

Pada Gambar 4.15 diperlihatkan grafik *Training Loss* terhadap jumlah epoch untuk variabel TPT. Pada awal pelatihan, nilai loss berada pada tingkat yang sangat tinggi, kemudian mengalami penurunan tajam pada *epoch-epoch* awal. Setelah itu, nilai loss terus menurun secara bertahap hingga mencapai kondisi relatif stabil pada kisaran nilai rendah, meskipun masih terdapat sedikit fluktuasi di beberapa titik *epoch*. Pola ini menunjukkan bahwa proses pelatihan berjalan dengan baik dan model mampu menurunkan kesalahan prediksi secara konsisten, meskipun terdapat sedikit variasi pada tahap akhir pelatihan.



**Gambar 4. 17 Perbandingan Data Aktual dan Prediksi BPN-1H**

Pada Gambar 4.16 diperlihatkan perbandingan antara data aktual dengan hasil prediksi. Hasil prediksi terlihat lebih halus dan mampu meredam fluktuasi tajam pada data aktual, terutama pada nilai puncak yang tinggi. Hal ini menunjukkan bahwa model dapat menangkap pola utama data dengan cukup baik meskipun masih terdapat deviasi pada beberapa titik tertentu.

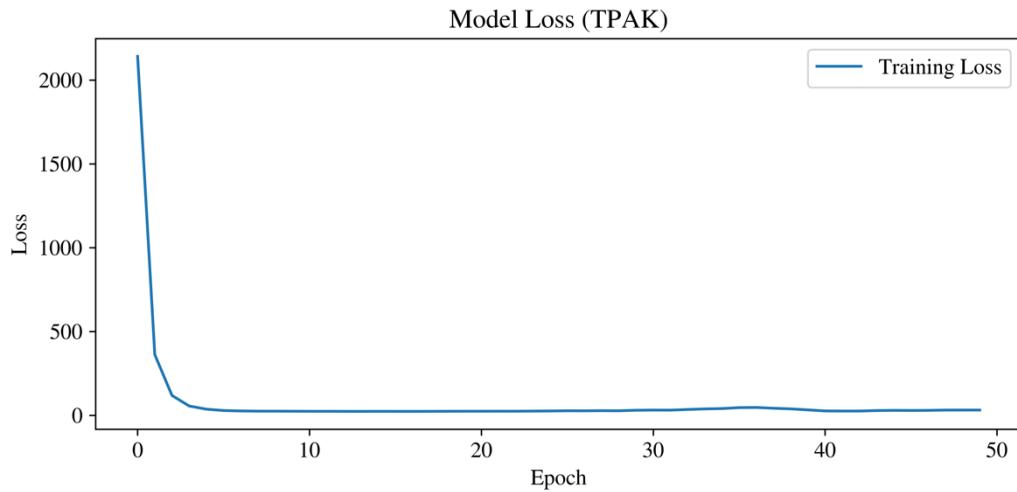
#### **4.3.9 Pelatihan BPN-1I**

Hasil yang diperoleh dari proses pelatihan BPN-11 memakai atribut TPAK dijabarkan dalam pada Tabel 4.10.

**Tabel 4. 10 Proses Pelatihan BPN-1I**

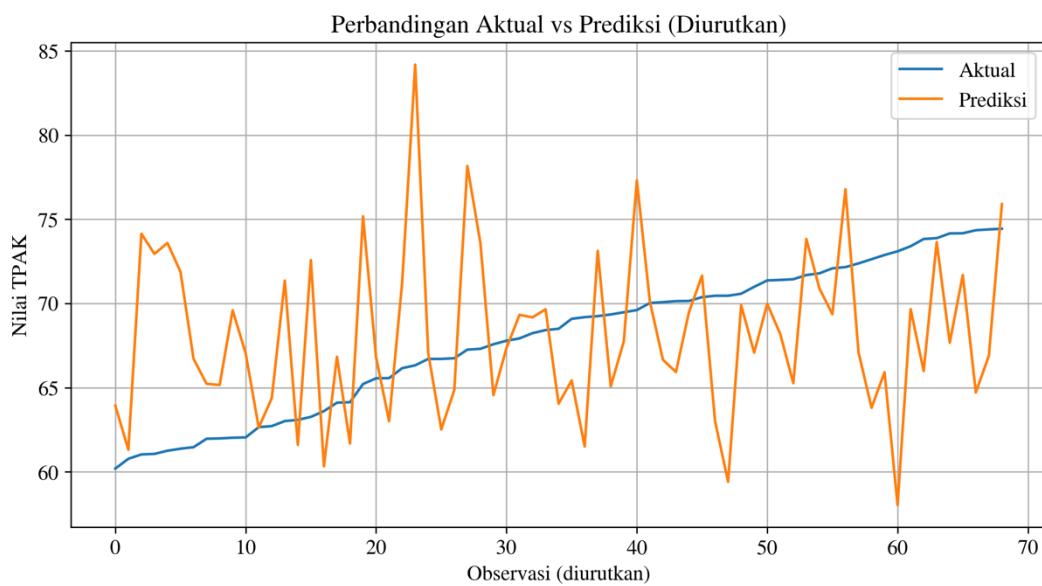
| <b>Epoch</b> | <b>Time</b> | <b>Training Loss</b> | <b>Validation Loss</b> |
|--------------|-------------|----------------------|------------------------|
| 1            | 3s 279ms    | 2.141.577            | 5.142.533              |
| 2            | 0s 13ms     | 3.617.323            | 1.772.052              |
| 3            | 0s 13ms     | 1.180.924            | 6.877.121              |
| 4            | 0s 14ms     | 5.515.708            | 4.496.387              |
| 5            | 0s 14ms     | 3.627.336            | 2.801.403              |
| ...          | ...         | ...                  | ...                    |
| 46           | 0s 13ms     | 2.836.284            | 3.357.354              |
| 47           | 0s 14ms     | 2.899.827            | 3.298.966              |
| 48           | 0s 14ms     | 3.089.568            | 3.002.415              |
| 49           | 0s 13ms     | 3.103.721            | 2.744.328              |
| 50           | 0s 13ms     | 3.103.498            | 2.400.037              |

Berdasarkan Tabel 4.10, diketahui bahwa pelatihan model BPN-1D berlangsung dengan waktu eksekusi sekitar 13 ms dengan MSE senilai 33,61 dan MAPE senilai 14,12%. Proses pelatihan tersebut menghasilkan nilai *Training Loss* sebesar 3.103.498 dan nilai *Validation Loss* sebesar 2.400.037. Hasil pelatihan divisualisasikan dalam bentuk grafik, bisa dilihat pada Gambar 4.17.



**Gambar 4. 18 Nilai Loss Pelatihan BPN-1I**

Pada Gambar 4.17 diperlihatkan grafik *Training Loss* terhadap jumlah *epoch*. Pada awal pelatihan, nilai loss berada pada tingkat yang sangat tinggi, kemudian mengalami penurunan tajam pada *epoch-epoch* awal. Selanjutnya, nilai loss mencapai kondisi stabil pada nilai yang sangat rendah dan relatif konstan hingga akhir pelatihan, yang menunjukkan bahwa model mampu meminimalkan kesalahan prediksi secara konsisten.



**Gambar 4. 19 Perbandingan Data Aktual dan Prediksi BPN-1I**

Pada Gambar 4.18 diperlihatkan perbandingan antara data aktual dengan hasil prediksi. Secara umum, pola prediksi mampu mengikuti tren data aktual dengan cukup baik, meskipun terdapat beberapa perbedaan pada titik tertentu. Hasil prediksi terlihat cenderung lebih halus, namun tetap menunjukkan kemampuan model dalam menangkap pola utama data meskipun masih terdapat deviasi pada beberapa kasus.

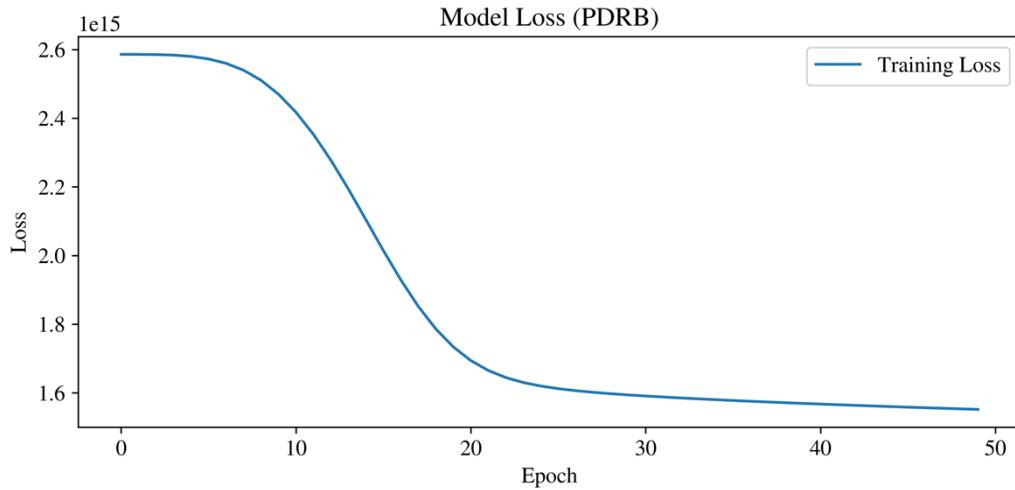
#### **4.3.10 Pelatihan BPN-1J**

Hasil yang diperoleh dari proses pelatihan BPN-1J memakai atribut PDRB dijabarkan dalam pada Tabel 4.11.

**Tabel 4. 11 Proses Pelatihan BPN-1J**

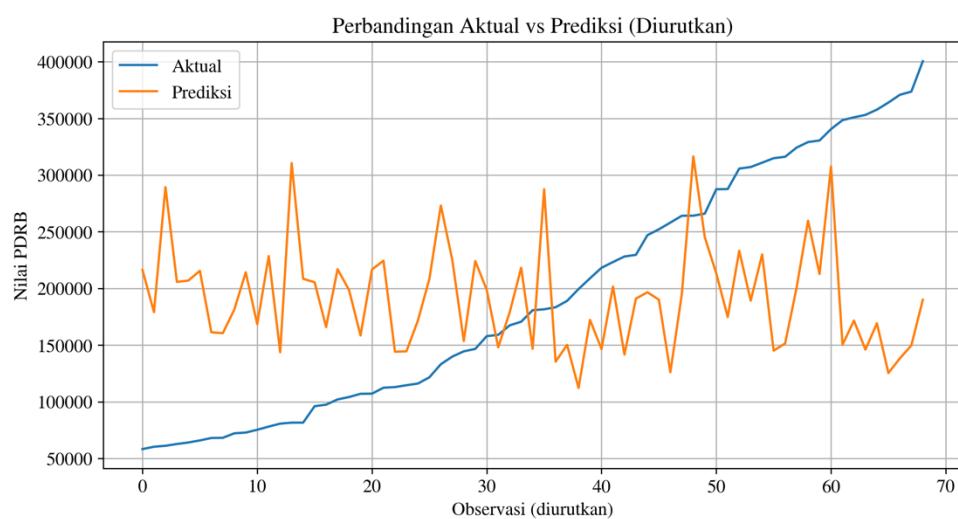
| Epoch | Time     | Training Loss | Validation Loss |
|-------|----------|---------------|-----------------|
| 1     | 2s 204ms | 2.586.144     | 6.496.978       |
| 2     | 0s 12ms  | 2.586.037     | 6.496.468       |
| 3     | 0s 12ms  | 2.585.519     | 6.494.635       |
| 4     | 0s 12ms  | 2.583.961     | 6.489.849       |
| 5     | 0s 12ms  | 2.580.214     | 6.479.657       |
| ...   | ...      | ...           | ...             |
| 46    | 0s 12ms  | 1.558.330     | 3.929.749       |
| 47    | 0s 13ms  | 1.556.664     | 3.921.692       |
| 48    | 0s 12ms  | 1.555.022     | 3.913.717       |
| 49    | 0s 12ms  | 1.553.395     | 3.905.741       |
| 50    | 0s 12ms  | 1.551.788     | 3.897.808       |

Berdasarkan Tabel 4.11, diketahui bahwa pelatihan model BPN-1D berlangsung dengan waktu eksekusi sekitar 13 ms dengan MSE senilai 1535238744309760,00 dan MAPE senilai 999999999.00%. Proses pelatihan tersebut menghasilkan nilai *Training Loss* sebesar 1.551.788 dan nilai *Validation Loss* sebesar 3.897.808. Hasil pelatihan divisualisasikan dalam bentuk grafik, bisa dilihat pada Gambar 4.19.



**Gambar 4. 20 Nilai Loss Pelatihan BPN-1J**

Pada Gambar 4.19 diperlihatkan grafik Training Loss terhadap jumlah epoch. Pada awal pelatihan, nilai loss berada pada tingkat yang sangat tinggi dan cenderung konstan hingga beberapa epoch pertama. Setelah itu, nilai loss mulai mengalami penurunan yang cukup signifikan hingga sekitar pertengahan epoch, kemudian terus menurun secara bertahap hingga mencapai kondisi stabil pada nilai yang lebih rendah. Pola ini menunjukkan bahwa proses pelatihan berlangsung baik, dan model mampu mengurangi kesalahan prediksi secara konsisten seiring bertambahnya jumlah epoch.



**Gambar 4. 21 Perbandingan Data Aktual dan Prediksi BPN-1A**

Pada Gambar 4.20 diperlihatkan perbandingan antara data aktual dengan hasil prediksi. Secara umum, pola prediksi mengikuti tren data aktual dengan cukup baik, meskipun terdapat perbedaan pada beberapa titik tertentu. Terlihat bahwa hasil prediksi cenderung lebih stabil dan meredam fluktuasi tajam pada data aktual, khususnya pada nilai ekstrem, sehingga menunjukkan bahwa model mampu menangkap pola utama data meskipun masih terdapat perbedaan pada kasus tertentu.

#### 4.4 Hasil Prediksi

Sesudah tahapan pelatihan dilaksanakan di setiap model. Kemudian melakukan proses uji. Tahapan ini dilakukan dengan meksud ingin tau performa paling optimal berdasarkan arsitektur model yang sudah dirancang dan mengetahui variable mana yang paling berpengaruh pada angka kemiskinan menggunakan metode ANN. Adapun hasil dari tahap pengujian bisa dilihat di Tabel 4.13

**Tabel 4. 12 Hasil Prediksi Model BPN 1 Hidden Layer**

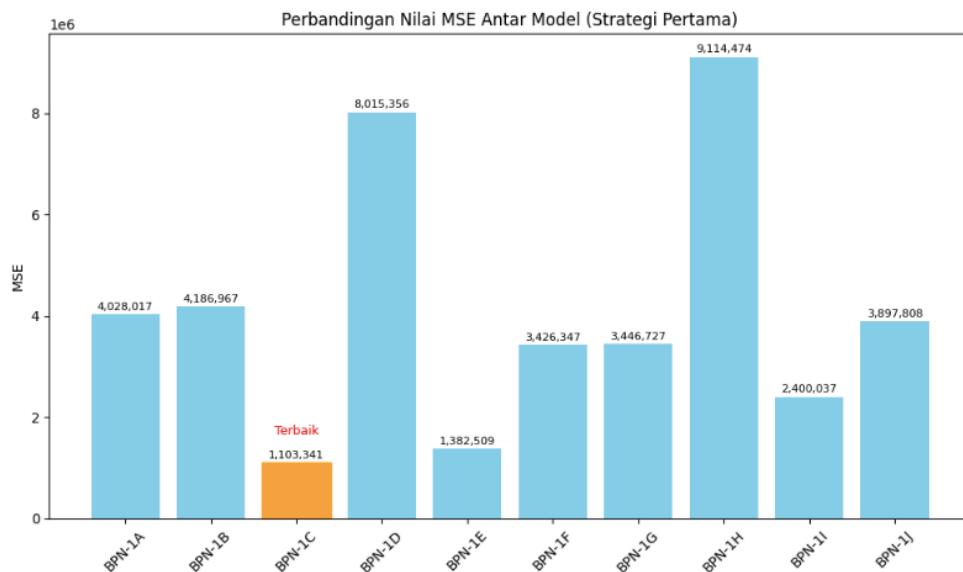
| Pelatihan | Arsitektur | Time (ms) | MSE                 | MAPE (%)     |
|-----------|------------|-----------|---------------------|--------------|
| BPN-1A    | 13-6-1     | 13        | 29,19               | 12,87        |
| BPN-1B    |            | 12        | 4,94                | 3,22         |
| BPN-1C    |            | 13        | 1085070.25          | 999,98       |
| BPN-1D    |            | 16        | 4,4                 | 2,95         |
| BPN-1E    |            | 12        | 10,79               | 5,41         |
| BPN-1F    |            | 14        | 195,18              | 22,65        |
| BPN-1G    |            | 12        | 279,65              | 25,37        |
| BPN-1H    |            | 14        | 4,57                | 2,73         |
| BPN-1I    |            | 13        | 33,61               | 14,12        |
| BPN-1J    |            | 12        | 1535238744309760.00 | 999999999.00 |

Berdasarkan hasil pelatihan yang ditampilkan pada Tabel 4.12, semua arsitektur yang digunakan memiliki struktur yang sama yaitu 13-6-1, namun menghasilkan nilai MSE dan MAPE yang berbeda. Menurut (Wijaya Rauf, 2023) mengatakan bahwa semakin kecil nilai MSE yang dihasilkan, maka semakin baik pula kemampuan model dalam melakukan prediksi. Hal serupa juga berlaku pada nilai MAPE, di mana semakin kecil nilainya menunjukkan tingkat kesalahan prediksi yang semakin rendah dalam bentuk persentase.

Dengan mengacu pada kedua indikator tersebut, pelatihan BPN-1D yang memiliki nilai MSE yaitu 4,4 dan MAPE sebesar 2,95% dapat dinyatakan sebagai model terbaik dalam memprediksi angka kemiskinan. Hal ini memperlihatkan penggunaan 1 hidden layer pada metode ANN sudah cukup untuk memenuhi target output. Temuan ini sejalan dengan (Cybenkot, 1989) mengatakan bahwa 1 hidden layer itu cukup dalam mendekati fungsi non-linier secara kontinu.

#### 4.5 Kesimpulan

Hasil pelatihan di arsitektur pertama menunjukkan bahwa penggunaan 1 lapisan tersembunyi sudah mampu menghasilkan prediksi angka kemiskinan dengan cukup baik. Model ANN menggunakan algoritme arus balik diketahui efektif untuk melakukan prediksi angka kemiskinan, seperti yang ditunjukkan oleh variasi nilai *Mean Squared Error* (MSE) dan *Mean Absolute Percentage Error* (MAPE) yang berbeda pada tiap model, di mana perbandingan nilai tersebut menjadi dasar dalam menentukan performa terbaik khususnya pada pelatihan BPN-1D. Adapun grafik hasil pelatihan dari strategi pertama yaitu 1 *hidden layer* bisa dilihat pada Gambar 4.21.



**Gambar 4. 22 Perbandingan performa model arsitektur  
1 hidden layer terbaik**

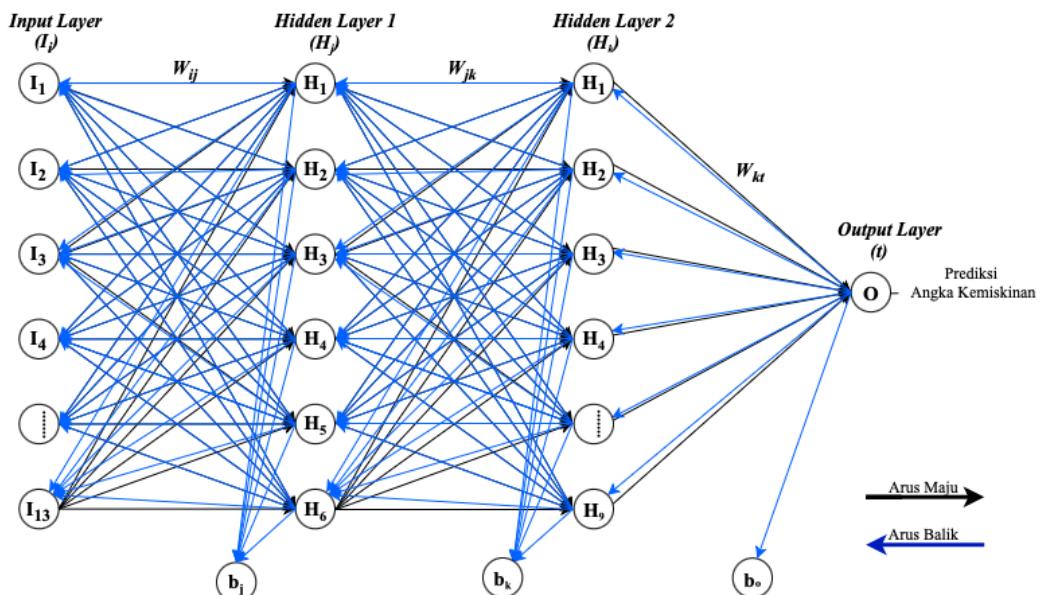
Gambar 4.21 di atas memperlihatkan perbandingan nilai MSE dari semua arsitektur pelatihan dengan konfigurasi 13-6-1. Dari hasil tersebut, dapat dilihat bahwa sebagian besar model menghasilkan MSE dan MAPE yang relatif tinggi, namun terdapat satu model yang menonjol, yaitu BPN-1D dengan MSE sebesar 4,4, dan MAPE sebesar 2,95% yang merupakan nilai terkecil dibandingkan model lainnya. Sesuai dengan yang dikatakan oleh (Wijaya Rauf, 2023) bahwa semakin kecil nilai MSE yang dihasilkan, maka semakin baik pula kemampuan model dalam melakukan prediksi, Hal serupa juga berlaku pada nilai MAPE, di mana semakin kecil nilainya menunjukkan tingkat kesalahan prediksi yang semakin rendah dalam bentuk persentase, sehingga BPN-1D dapat dinyatakan sebagai model terbaik pada strategi pertama ini.

## BAB V

### MODEL ANN DUA HIDDEN LAYER

#### 5.1 Design

*Design* penelitian merupakan rangkaian tahapan yang dirancang untuk menjawab permasalahan yang diteliti. Melalui tahapan ini, objek penelitian dapat dipahami dengan lebih jelas dan menjadi dasar untuk menarik kesimpulan. Dalam riset ini, peneliti menggunakan Artificial Neural Network (ANN) dengan algoritma Backpropagation untuk memprediksi angka kemiskinan di Indonesia. Strategi kedua dilakukan dengan membangun jaringan ANN yang memiliki dua hidden layer. Jaringan tersebut terdiri atas input berupa data kemiskinan ( $I_i$ ), bobot penghubung antara lapisan input dan lapisan tersembunyi pertama ( $W_{ij}$ ), keluaran di lapisan tersembunyi dengan jumlah node yang sudah ditetapkan ( $H_j$ ), bobot penghubung dari lapisan tersembunyi pertama dan lapisan tersembunyi kedua bobot penghubung menuju output ( $W_{jk}$ ), serta output berupa prediksi angka kemiskinan ( $O_t$ ). Lebih jelasnya dapat dilihat pada Gambar 4.1.



**Gambar 5. 2 Rancangan Arsitektur Model ANN Dengan Lapisan Tersembunyi Kedua**

Strategi pertama dalam memprediksi tingkat kemiskinan memakai model ANN dengan algoritma arus balik dirancang dengan satu hidden layer berkonfigurasi 13–6–1. Adapun tahapan perhitungannya di arsitektur kedua yaitu:

1. Penetapan data masukan
2. Penentuan parameter awal model
3. Proses Forward-Propagation
  - a. Menghitung Menghitung sinyal pada setiap neuron di lapisan tersembunyi ( $H_j$ ). Pada tahap ini, informasi disalurkan ke variabel input ( $I_i$ ) yang kemudian meneruskan sinyal tersebut ke seluruh neuron pada lapisan tersembunyi pertama.

$$H_{net\ j} = T_{0j} + \sum_{i=1}^n I_i T_{ij} \quad (1)$$

Dimana  $i$  menunjukkan node ke- $i$  ( $i = 1, 2, 3\dots, n$ ) pada lapisan input, dan  $j$  menunjukkan node ke- $j$  ( $j = 1, 2, 3\dots, p$ ) pada lapisan tersembunyi pertama.  $I_i$  merepresentasikan nilai input yang mengalir dari node input  $i$  menuju node tersembunyi  $j$ .  $T_{0j}$  berfungsi sebagai bias pada lapisan input, sedangkan  $T_{ij}$  adalah bobot yang menghubungkan node input  $i$  dengan node  $j$ . Semua output pada hidden layer pertama diproses menggunakan fungsi aktivasi ReLU, dan hasilnya kemudian diterapkan pada setiap node di lapisan tersembunyi pertama untuk tahap pemrosesan berikutnya.

$$H_{1j} = f(H_{1netj}) = \max(0, H_{1netj}) \quad (2)$$

Dengan fungsi max, nilai yang diambil adalah maksimum antara 0 dan  $H_{1netj}$ . Nilai  $f(H_{1netj})$  ini berperan sebagai aktivasi pada node tersembunyi  $j$ , yang kemudian diteruskan ke semua node di hidden layer kedua. Dengan demikian, output dari hidden layer pertama menjadi input bagi hidden layer kedua. Selanjutnya, sinyal dihitung untuk masing-masing node pada lapisan tersembunyi kedua ( $H_{2q}$ ).

$$H_{2net\ q} = W_{0q} + \sum_{j=1}^p H_{1j} W_{jq} \quad (3)$$

- b. Dimana  $j$  node ke- $j$  ( $j = 1, 2, 3, \dots, p$ ) pada hidden layer pertama serta  $q$  neuron ke- $q$  ( $q = 1, 2, 3, \dots, r$ ) pada lapisan tersembunyi kedua.  $H_{1j}$  adalah hasil keluaran dari node tersembunyi ke- $j$  pada hidden layer pertama yang diteruskan ke node  $q$  pada hidden layer kedua.  $W_{0q}$  adalah bias pada lapisan tersembunyi kedua serta  $W_{jq}$  adalah bobot pada node tersembunyi pertama  $j$  tertuju pada node tersembunyi kedua  $q$ . Setiap output dari lapisan tersembunyi kedua diproses melalui fungsi aktivasi ReLU. Nilai yang dihasilkan kemudian diteruskan ke seluruh neuron di lapisan tersembunyi kedua sebagai masukan untuk tahap pemrosesan selanjutnya.

$$H_{2q} = f(H_{2netq}) = \max(0, H_{2netq}) \quad (4)$$

- c. Dengan menggunakan fungsi max, nilai yang dipilih adalah yang lebih besar antara 0 dan  $H_{2netq}$ . Hasil perhitungan  $f(H_{2netq})$  merepresentasikan tingkat aktivasi pada node tersembunyi ke- $q$ , yang kemudian disebarluaskan ke seluruh node pada lapisan keluaran. Setelah semua nilai dari hidden layer diperoleh, selanjutnya dilakukan penjumlahan seluruh sinyal yang masuk ke masing-masing node keluaran ( $H_{2q} \dots H_{2r}$ ) dengan memperhitungkan bobot masing-masing. Di sini,  $U_{0k}$  berperan sebagai bias pada hidden layer kedua,  $H_{2q}$  merupakan output dari lapisan tersembunyi kedua, dan  $U_{qk}$  adalah bobot yang menghubungkan node tersembunyi ke- $q$  dengan node keluaran  $k$  ( $k = 1, 2, 3, \dots, m$ ).

$$O_{net\ k} = U_{0k} + \sum_{q=1}^r H_{2q} U_{qk} \quad (5)$$

Memakai fungsi aktivasi yaitu ReLU dalam melakukan perhitungan sinyal keluaran, nilai yang didapatkan adalah:

$$O_k = f(O_{net\ k}) = \max(0, O_{net\ k}) \quad (6)$$

- d.  $O_{net\ k}$  adalah output yang diperoleh dari penerapan fungsi aktivasi, yaitu hasil penjumlahan bobot yang menghubungkan node tersembunyi dengan node keluaran. Setelah nilai keluaran dihasilkan, perbedaan antara output tersebut dengan nilai target dihitung menggunakan rumus MSE dan MAPE.

$$MSE = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2 \quad (7)$$

$$MAPE = \sum_{T=1}^n \left| \frac{y_i - \hat{y}_i}{\hat{y}_i} \right| \times 100\% \quad (8)$$

Apabila kondisi konvergensi sudah terpenuhi, maka algoritme akan berhenti secara otomatis. Namun, jika konvergensi belum tercapai, sistem akan melanjutkan ke tahap backpropagation, di mana bobot-bobot jaringan diperbarui untuk mengurangi kesalahan dan meningkatkan akurasi hasil.

#### 4. Fase Propagasi Balik

- a. Tentukan besarnya error ( $\delta_k$ ) pada lapisan keluaran ( $O_k$ ), yang menerima nilai target ( $t_k$ ). Nilai error ini kemudian diteruskan ke tahap berikutnya untuk digunakan dalam perhitungan penyesuaian bobot dan koreksi bias antara hidden layer kedua dan lapisan keluaran.

$$\delta_k = (T_k - O_k) f'(O_{net\ k}) \quad (9)$$

dimana:

$$f'(O_{net\ k}) = (1 - f(O_{net\ k}))f(O_{net\ k}) \quad (10)$$

Sesudah diperoleh besaran kesalahan pada lapisan tersembunyi kedua, langkah berikutnya adalah menentukan nilai error pada lapisan tersembunyi pertama.

$$\delta_{2q} = f'(H_{2net\ q}) \sum_{k=1}^m \delta_k U_{qk} \quad (11)$$

$$\delta_{ij} = f'(H_{1net\ j}) \sum_{q=1}^r \delta_{2q} W_{jq} \quad (12)$$

Dimana:

$$f'(H_{2net\ q}) = (1 - f(H_{2net\ q}))f(H_{2net\ q}) \quad (13)$$

$$f'(H_{1net\ j}) = (1 - f(H_{1net\ j}))f(H_{1net\ j}) \quad (14)$$

## 5. Fase Perubahan Bobot

Begitu besaran kesalahan pada masing-masing lapisan teridentifikasi, selanjutnya dapat ditentukan total penyesuaian yang akan diaplikasikan pada bobot sebelumnya.

$$\Delta T_{ij} = \alpha \delta_{1j} I_i \quad (15)$$

$$\Delta W_{jq} = \alpha \delta_{2q} H_{ij} \quad (16)$$

$$\Delta U_{qk} = \alpha \delta_k H_{2q} \quad (17)$$

$\Delta T_{ij}$ ,  $\Delta W_{jq}$ , dan  $\Delta U_{qk}$  adalah Penyesuaian yang akan diterapkan pada bobot sebelumnya dilakukan dengan mempertimbangkan  $\alpha$  sebagai tingkat pembelajaran (learning rate). Di sini,  $\delta_{1j}$  menunjukkan kesalahan antara lapisan input dan hidden layer pertama,  $\delta_{2q}$  merepresentasikan deviasi antara hidden layer pertama dan kedua, sedangkan  $\delta_k$  menggambarkan selisih antara hidden layer kedua dengan lapisan keluaran.  $I_i$  merupakan masukan yang mengalir dari node  $i$  menuju node  $j$ ,  $H_{1j}$  adalah hasil aktivasi pada hidden layer pertama, dan  $H_{2q}$  merupakan output aktivasi dari hidden layer kedua.

Setelah nilai  $\Delta T_{ij}$ ,  $\Delta W_{jq}$ , dan  $\Delta U_{qk}$  dihitung, tahap berikutnya adalah memperbarui bobot lama sehingga menjadi bobot yang baru dan siap digunakan pada iterasi selanjutnya.

$$T_{ij}(\text{new}) = T_{ij}(\text{old}) + \Delta T_{ij} \quad (18)$$

$$W_{jq}(\text{new}) = W_{jq}(\text{old}) + \Delta W_{jq} \quad (19)$$

$$U_{qk}(\text{new}) = U_{qk}(\text{old}) + \Delta U_{qk} \quad (20)$$

$T_{ij}(new)$  adalah bobot baru antara input layer dan hidden layer pertama yang akan dicari, sedangkan  $T_{ij}(old)$  adalah bobot lama yang akan diperbarui.  $W_{jq}(new)$  adalah bobot baru antara hidden layer pertama dan hidden layer kedua yang akan dicari, sedangkan  $W_{jq}(old)$  adalah bobot lama yang akan diperbarui.  $U_{qk}(new)$  adalah bobot baru antara hidden layer kedua dan lapisan keluaran yang akan dicari, sedangkan  $U_{qk}(old)$  adalah bobot lama yang akan diperbarui.

## 5.2 Implementasi

Jaringan dibangun menggunakan platform Kaggle dengan dua *hidden layer*. Adapun lapisan tersembunyi pertama berisi 13 node, dan lapisan tersembunyi kedua berisi 9 node. Pada tahap evaluasi, proses optimasi dilakukan dengan mengacu pada nilai MSE terkecil selama pengujian. Setiap atribut digunakan secara bergantian sebagai atribut target, sekaligus ditentukan jumlah node pada hidden layer yang merujuk pada pendekatan (Sahi & Galib, 2025) dengan menggunakan jumlah digit terbanyak dalam dataset sebagai acuan. Rincian parameter uji bisa diketahui dalam Tabel 4.1.

**Tabel 5. 1 Parameter Model**

| Tipe Jaringan            | BPN-ANN               |
|--------------------------|-----------------------|
| Fungsi Aktivasi          | ReLU                  |
| Optimizer                | Adam                  |
| Persamaan                | MSE                   |
| <i>Input Layer</i>       | 13                    |
| <i>Hidden Layer</i>      | <i>2 Hidden Layer</i> |
| <i>Node Hidden Layer</i> | (6, 9)                |
| <i>Output Layer</i>      | 1                     |
| <i>Learning Rate</i>     | 0,2                   |

## 5.3 Uji Coba

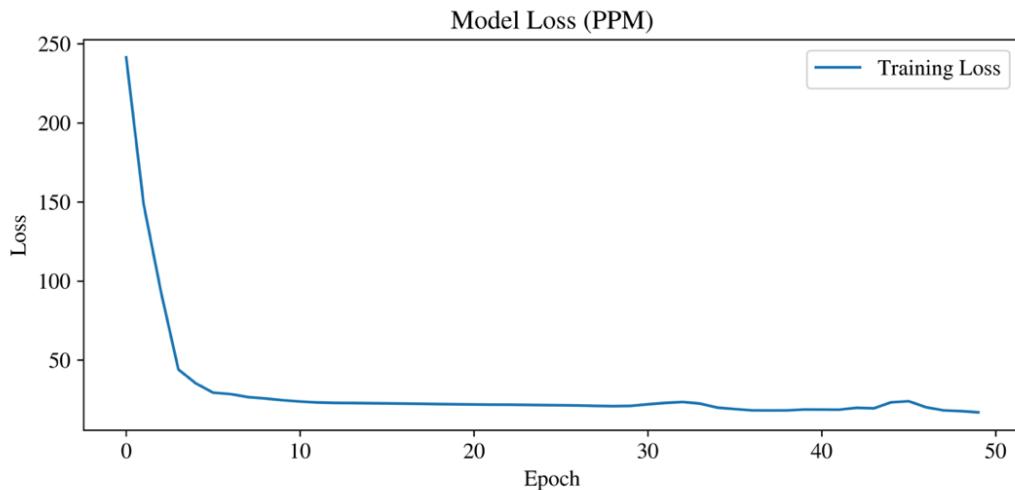
### 5.3.1 Pelatihan BPN-2A

Hasil percobaan BPN-2A menggunakan variable PPM yang menjadi target disajikan dalam Tabel 5.2.

**Tabel 5. 2 Proses Uji BPN-2A**

| Epoch | Time     | Training Loss | Validation Loss |
|-------|----------|---------------|-----------------|
| 1     | 3s 337ms | 2.414.477     | 1.923.037       |
| 2     | 0s 13ms  | 1.487.115     | 1.459.135       |
| 3     | 0s 12ms  | 9.319.689     | 7.617.441       |
| 4     | 0s 13ms  | 4.409.810     | 5.539.956       |
| 5     | 0s 12ms  | 3.534.023     | 4.673.075       |
| 46    | 0s 12ms  | 2.402.065     | 3.775.126       |
| 47    | 0s 12ms  | 2.022.783     | 3.795.759       |
| 48    | 0s 14ms  | 1.825.185     | 4.087.496       |
| 49    | 0s 13ms  | 1.777.405     | 4.191.708       |
| 50    | 0s 13ms  | 1.696.456     | 4.408.681       |

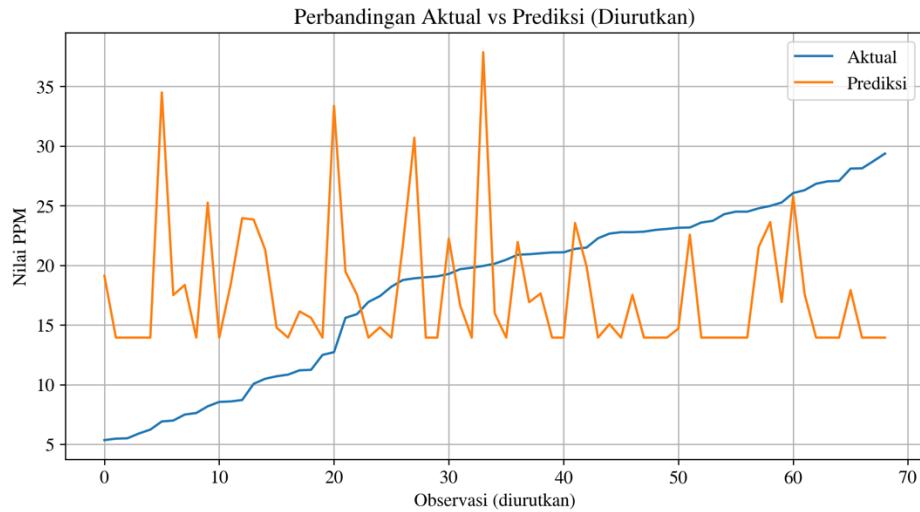
Berdasarkan Tabel 5.2, diketahui bahwa pelatihan model BPN-2A berlangsung dengan waktu eksekusi sekitar 13 ms dengan MSE senilai 32,12 dan MAPE senilai 13,51%. Proses pelatihan tersebut menghasilkan nilai *Training Loss* sebesar 1.696.456 dan nilai *Validation Loss* sebesar 4.408.681. Hasil pelatihan divisualisasikan dalam bentuk grafik, bisa diketahui dalam Gambar 5.1.



**Gambar 5.3 Nilai Loss Pelatihan BPN-2A**

Pada Gambar 5.1 ditampilkan grafik training loss pada model prediksi angka kemiskinan. Terlihat bahwa nilai loss awalnya sangat tinggi, namun mengalami penurunan tajam pada beberapa epoch pertama. Setelah melewati sekitar epoch ke-10, pola loss menunjukkan kecenderungan stabil dengan nilai yang relatif rendah, meskipun masih terdapat sedikit fluktuasi pada beberapa titik. Hal ini

mengindikasikan bahwa model mampu belajar dengan cepat di tahap awal dan kemudian beradaptasi secara bertahap untuk mencapai kondisi yang lebih optimal, dengan tingkat kesalahan yang semakin kecil seiring bertambahnya epoch.



**Gambar 5. 4 Perbandingan Data Aktual dan Prediksi BPN-2A**

Pada Gambar 5.6 ditampilkan perbandingan antara nilai aktual BPN-2A. Terlihat bahwa pola prediksi model (garis oranye) mengikuti tren nilai aktual (garis biru) dengan cukup baik, meskipun terdapat beberapa titik di mana perbedaan nilai masih terlihat cukup besar. Secara umum, model mampu menangkap pola fluktuasi data dengan arah perubahan yang serupa, menunjukkan bahwa model memiliki kemampuan prediktif yang baik dalam merepresentasikan variasi data aktual, meskipun masih terdapat sedikit deviasi pada beberapa observasi tertentu.

### 5.3.2 Pelatihan BPN-2B

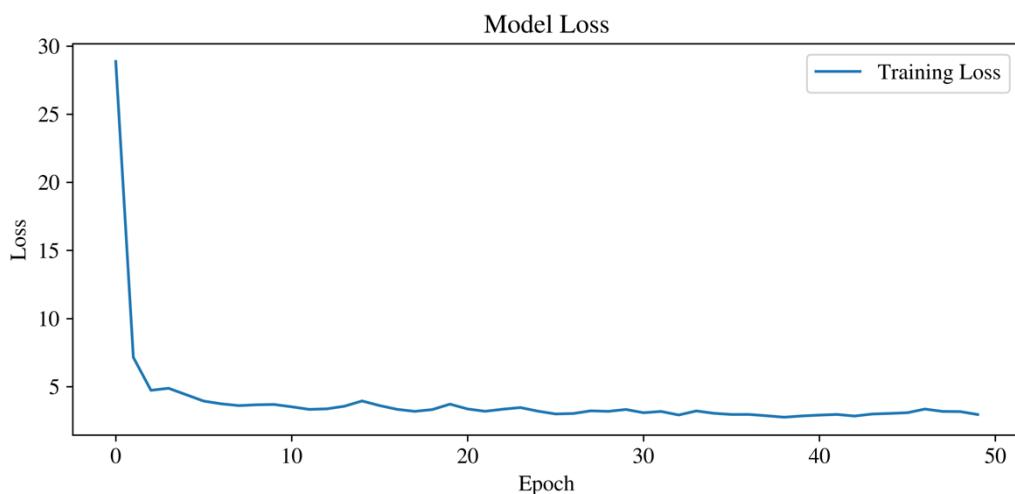
Hasil percobaan BPN-2B menggunakan variable RLSP yang menjadi target disajikan dalam Tabel 5.3.

**Tabel 5. 3 Proses Uji BPN-2B**

| Epoch | Time     | Training Loss | Validation Loss |
|-------|----------|---------------|-----------------|
| 1     | 3s 312ms | 2.886.642     | 1.326.619       |
| 2     | 0s 12ms  | 7.145.667     | 3.532.985       |
| 3     | 0s 12ms  | 4.728.229     | 3.572.728       |
| 4     | 0s 13ms  | 4.877.651     | 3.182.301       |
| 5     | 0s 12ms  | 4.407.230     | 2.711.187       |

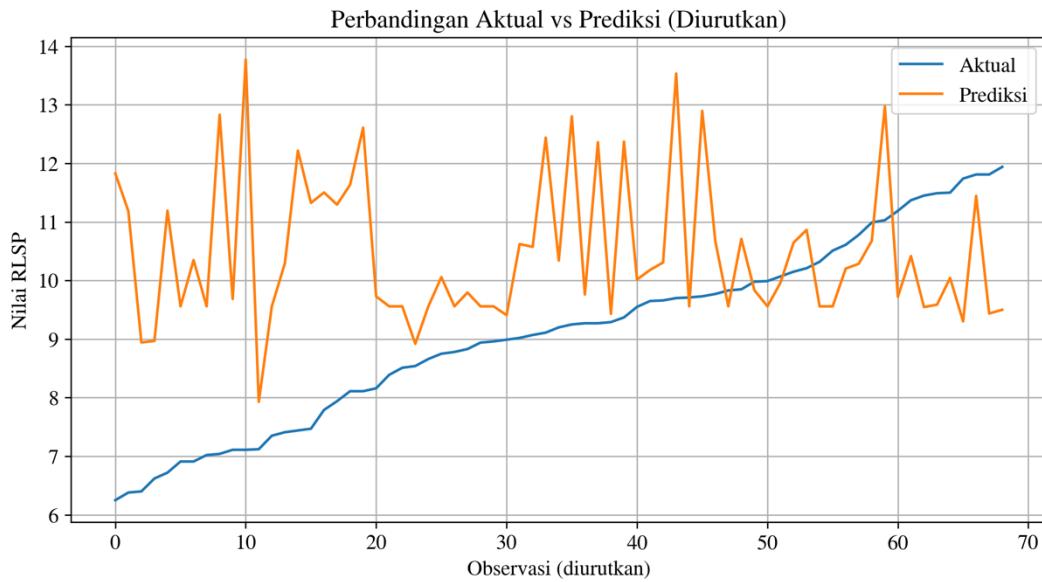
|    |         |           |           |
|----|---------|-----------|-----------|
| 46 | 0s 12ms | 3.083.466 | 3.259.235 |
| 47 | 0s 12ms | 3.351.166 | 2.902.959 |
| 48 | 0s 12ms | 3.177.677 | 4.463.291 |
| 49 | 0s 12ms | 3.165.982 | 4.192.760 |
| 50 | 0s 12ms | 2.950.175 | 3.129.801 |

Berdasarkan Tabel 5.3, diketahui bahwa pelatihan model BPN-2B berlangsung dengan waktu eksekusi sekitar 12 ms dengan MSE senilai 4,71 dan MAPE senilai 3,09%. Proses pelatihan tersebut menghasilkan nilai *Training Loss* sebesar 2.950.175 dan nilai *Validation Loss* sebesar 3.129.801. Hasil pelatihan divisualisasikan dalam bentuk grafik, bisa dilihat pada Gambar 5.3.



**Gambar 5.5 Nilai Loss Pelatihan BPN-2B**

Pada Gambar 5.3 ditampilkan grafik training loss pada model prediksi angka kemiskinan. Terlihat bahwa nilai loss awalnya sangat tinggi, namun mengalami penurunan tajam pada epoch awal. Terlihat bahwa mulai dari epoch-1 dan seterusnya mengalami penyesuaian bobot ditandai dengan grafik yang belum sepenuhnya stabil. Hal ini mengindikasikan bahwa model masih menyesuaikan bobot secara bertahap untuk mencapai kondisi yang lebih optimal, dengan tingkat kesalahan yang semakin kecil seiring bertambahnya epoch.



**Gambar 5.6 Perbandingan Data Aktual dan Prediksi BPN-2B**

ada Gambar 5.6 ditampilkan perbandingan antara nilai aktual dan hasil prediksi untuk variabel BPN-2B. Terlihat bahwa garis prediksi mengikuti pola nilai aktual dengan cukup baik, meskipun terdapat beberapa titik di mana perbedaan nilai masih tampak. Secara umum, model mampu merepresentasikan arah dan fluktuasi data aktual dengan pola yang serupa, menunjukkan bahwa model memiliki kemampuan prediksi yang cukup baik, meskipun masih terdapat sedikit selisih pada beberapa observasi.

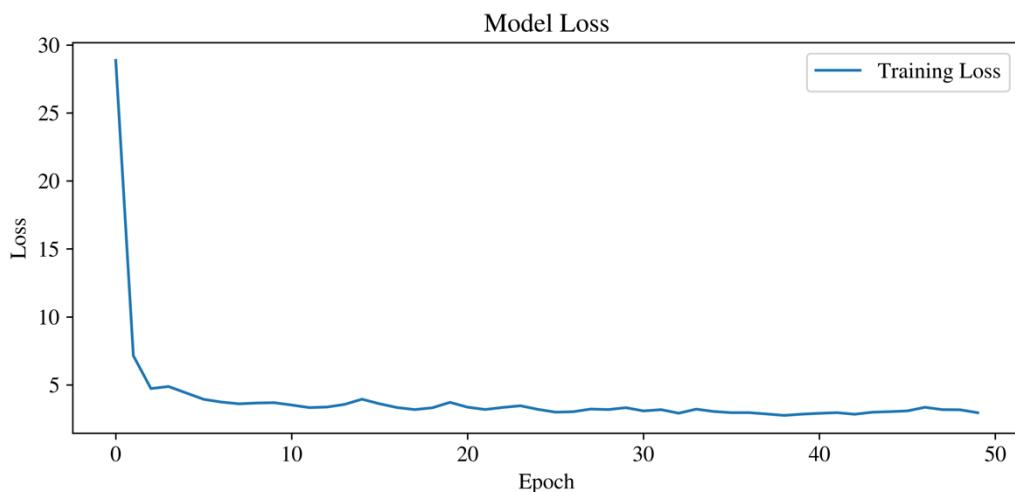
### 5.3.3 Pelatihan BPN-2C

Hasil percobaan BPN-2C menggunakan variable PPKD yang menjadi target disajikan dalam Tabel 5.4.

**Tabel 5. 4 Proses Uji BPN-2C**

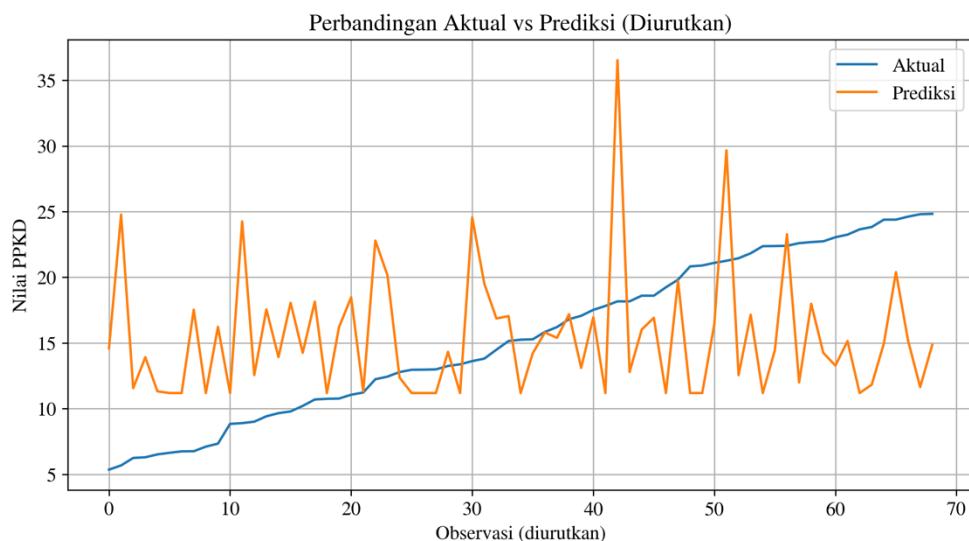
| Epoch | Time     | Training Loss | Validation Loss |
|-------|----------|---------------|-----------------|
| 1     | 2s 216ms | 9.483.124     | 2.160.814       |
| 2     | 0s 11ms  | 1.592.406     | 1.123.274       |
| 3     | 0s 11ms  | 7.019.338     | 6.652.294       |
| 4     | 0s 11ms  | 3.182.885     | 3.876.057       |
| 5     | 0s 10ms  | 2.269.817     | 2.202.492       |
| 46    | 0s 11ms  | 9.793.575     | 1.102.256       |
| 47    | 0s 10ms  | 9.669.066     | 1.085.160       |
| 48    | 0s 10ms  | 9.460.427     | 1.028.064       |
| 49    | 0s 10ms  | 8.982.458     | 1.036.644       |
| 50    | 0s 10ms  | 8.378.876     | 9.918.116       |

Berdasarkan Tabel 5.4, diketahui bahwa pelatihan model BPN-2C berlangsung dengan waktu eksekusi sekitar 10 ms dengan MSE senilai 1327162.12 dan MAPE senilai 999,99%. Proses pelatihan tersebut menghasilkan nilai *Training Loss* sebesar 8.378.876 dan nilai *Validation Loss* sebesar 9.918.116. Hasil pelatihan divisualisasikan dalam bentuk grafik, bisa diketahui dalam Gambar 5.5.

**Gambar 5. 7 Nilai Loss Pelatihan BPN-2C**

Pada Gambar 5.5 ditampilkan grafik training loss pada model prediksi angka kemiskinan. Terlihat bahwa nilai loss awalnya sangat tinggi, namun mengalami penurunan signifikan di epoch pertama. Setelah melewati sekitar epoch

ke-2 atau ke-3, pola loss menunjukkan kecenderungan stabil. Hal ini mengindikasikan bahwa model sudah cukup baik dan mampu belajar dengan cepat di tahap awal, kemudian beradaptasi secara bertahap untuk mencapai kondisi yang lebih optimal, dengan tingkat kesalahan yang semakin kecil seiring bertambahnya epoch.



**Gambar 5.8 Perbandingan Data Aktual dan Prediksi BPN-2C**

Pada Gambar 5.6 ditampilkan perbandingan antara nilai aktual dan hasil prediksi untuk variabel BPN-2C. Terlihat bahwa garis prediksi mengikuti pola nilai aktual dengan cukup baik, meskipun terdapat beberapa titik di mana perbedaan nilai masih tampak. Secara umum, model mampu merepresentasikan arah dan fluktuasi data aktual dengan pola yang serupa, menunjukkan bahwa model memiliki kemampuan prediksi yang cukup baik, meskipun masih terdapat sedikit selisih pada beberapa observasi.

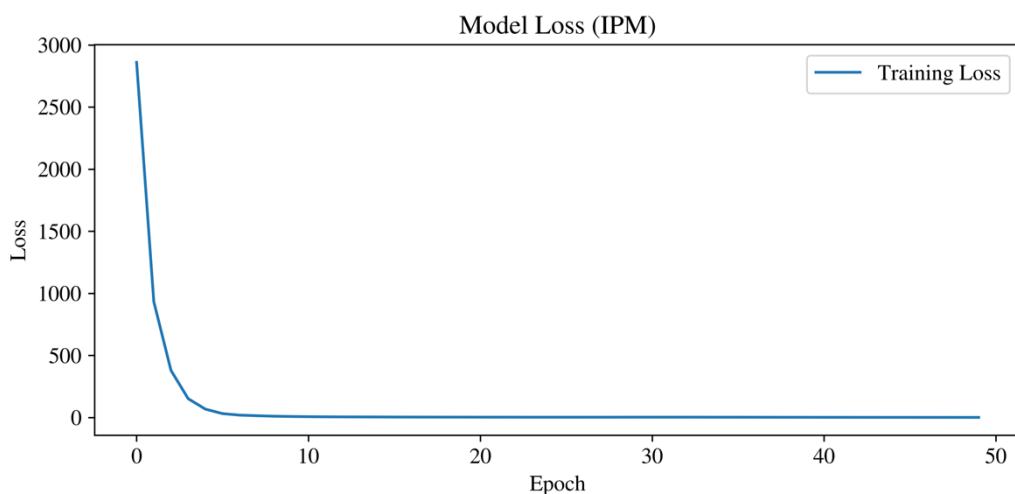
### 5.3.4 Pelatihan BPN-2D

Hasil percobaan BPN-2D menggunakan variable IPM yang menjadi target disajikan dalam Tabel 5.5.

**Tabel 5. 5 Proses Uji BPN-2D**

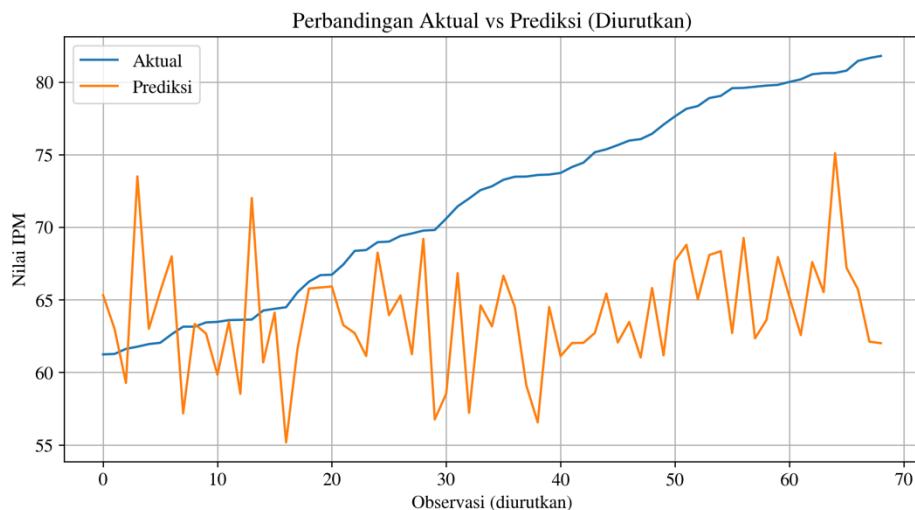
| Epoch | Time     | Training Loss | Validation Loss |
|-------|----------|---------------|-----------------|
| 1     | 2s 244ms | 2.859.909     | 1.649.017       |
| 2     | 0s 12ms  | 9.333.262     | 5.502.628       |
| 3     | 0s 12ms  | 3.799.130     | 2.101.733       |
| 4     | 0s 12ms  | 1.523.039     | 9.810.342       |
| 5     | 0s 12ms  | 6.806.007     | 4.391.829       |
| 46    | 0s 13ms  | 1.652.928     | 4.887.349       |
| 47    | 0s 13ms  | 1.598.930     | 4.621.906       |
| 48    | 0s 12ms  | 1.536.581     | 4.539.695       |
| 49    | 0s 12ms  | 1.473.426     | 4.283.538       |
| 50    | 0s 13ms  | 1.386.608     | 4.038.388       |

Berdasarkan Tabel 5.5, diketahui bahwa pelatihan model BPN-2D berlangsung dengan waktu eksekusi sekitar 13 ms dengan MSE senilai 2,74 dan MAPE senilai 2,18%. Proses pelatihan tersebut menghasilkan nilai *Training Loss* sebesar 1.386.608 dan nilai *Validation Loss* sebesar 4.038.388. Hasil pelatihan divisualisasikan dalam bentuk grafik, bisa dilihat pada Gambar 5.7.

**Gambar 5. 9 Nilai Loss Pelatihan BPN-2D**

Pada Gambar 5.7 ditampilkan grafik training loss pada model prediksi angka kemiskinan. Terlihat bahwa nilai loss awalnya sangat tinggi, namun mengalami penurunan signifikan di epoch pertama. Setelah melewati sekitar epoch

ke-5, pola loss menunjukkan kecenderungan stabil. Hal ini mengindikasikan bahwa model sudah cukup baik dan mampu belajar dengan cepat di tahap awal, kemudian beradaptasi secara bertahap untuk mencapai kondisi yang lebih optimal, dengan tingkat kesalahan yang semakin kecil seiring bertambahnya epoch.



**Gambar 5.10 Perbandingan Data Aktual dan Prediksi BPN-2D**

Pada Gambar 5.8 ditampilkan perbandingan antara nilai aktual dan hasil prediksi untuk variabel BPN-2C. Terlihat bahwa garis prediksi mengikuti tren nilai aktual dengan cukup baik, di mana keduanya menunjukkan pola kenaikan dan penurunan yang searah pada sebagian besar titik data. Meskipun terdapat beberapa perbedaan nilai pada beberapa observasi, secara umum model mampu menangkap variasi dan fluktuasi data aktual dengan cukup akurat, sehingga dapat dikatakan memiliki kemampuan prediksi yang baik terhadap pola data IPM.

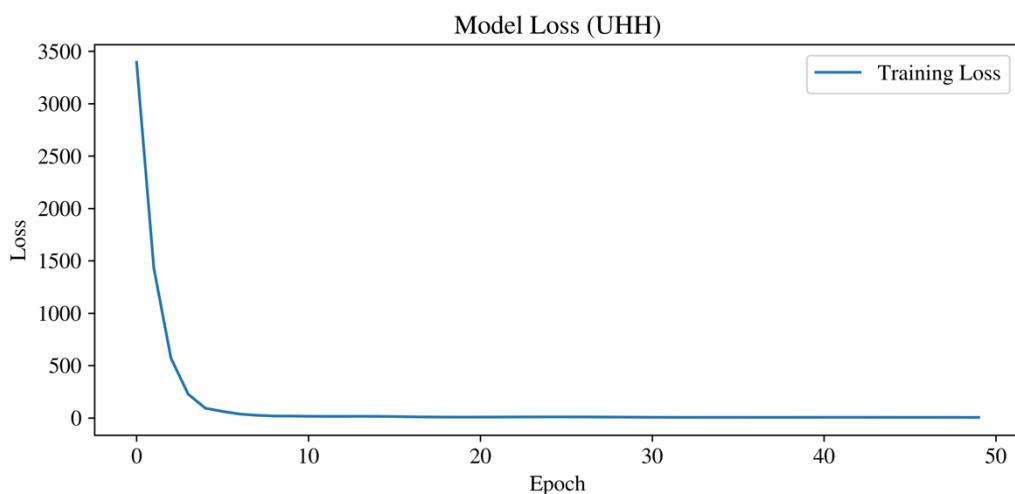
### 5.3.5 Pelatihan BPN-2E

Hasil percobaan BPN-2E menggunakan variable UHH yang menjadi target disajikan dalam Tabel 5.6.

**Tabel 5. 6 Proses Uji BPN-2E**

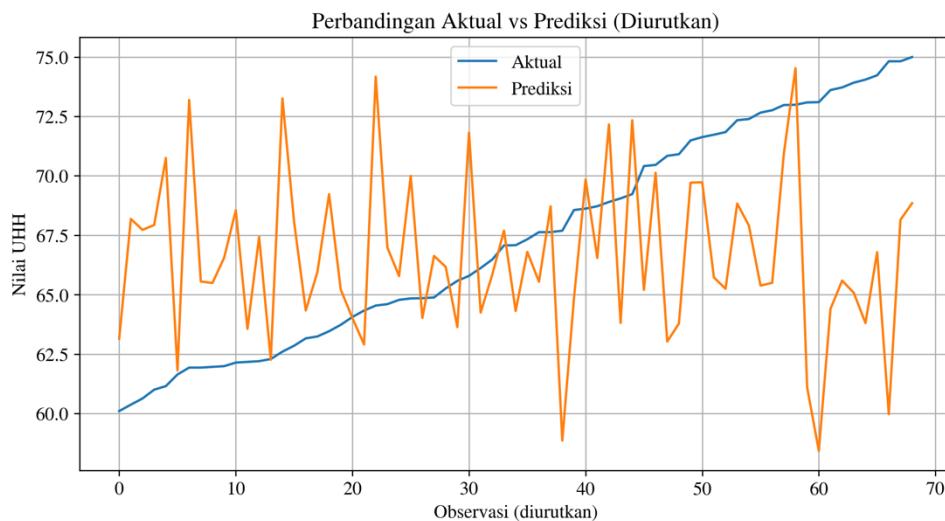
| Epoch | Time     | Training Loss | Validation Loss |
|-------|----------|---------------|-----------------|
| 1     | 3s 310ms | 3.394.626     | 2.415.813       |
| 2     | 0s 12ms  | 1.432.416     | 7.403.925       |
| 3     | 0s 12ms  | 5.701.343     | 2.637.668       |
| 4     | 0s 12ms  | 2.269.825     | 1.164.786       |
| 5     | 0s 12ms  | 9.273.307     | 9.335.894       |
| 46    | 0s 13ms  | 5.293.089     | 8.994.229       |
| 47    | 0s 12ms  | 5.168.646     | 8.424.203       |
| 48    | 0s 13ms  | 5.048.906     | 7.841.152       |
| 49    | 0s 13ms  | 4.910.801     | 7.471.987       |
| 50    | 0s 14ms  | 4.784.503     | 7.190.042       |

Berdasarkan Tabel 5.6, diketahui bahwa pelatihan model BPN-2E berlangsung dengan waktu eksekusi sekitar 14 ms dengan MSE senilai 5,03 dan MAPE senilai 4,82%. Proses pelatihan tersebut menghasilkan nilai *Training Loss* sebesar 4.784.503 dan nilai *Validation Loss* sebesar 7.190.042. Hasil pelatihan divisualisasikan dalam bentuk grafik, bisa diketahui dalam Gambar 5.9.

**Gambar 5. 11 Nilai Loss Pelatihan BPN-2E**

Pada Gambar 5.9 ditampilkan grafik training loss pada model prediksi angka kemiskinan. Terlihat bahwa nilai loss awalnya sangat tinggi, namun mengalami penurunan signifikan di epoch pertama. Setelah melewati sekitar epoch

ke-4, pola loss menunjukkan kecenderungan stabil. Hal ini mengindikasikan bahwa model sudah cukup baik dan mampu belajar dengan cepat di tahap awal, kemudian beradaptasi secara bertahap untuk mencapai kondisi yang lebih optimal, dengan tingkat kesalahan yang semakin kecil seiring bertambahnya epoch.



**Gambar 5.12 Perbandingan Data Aktual dan Prediksi BPN-2E**

Pada Gambar 5.6 ditampilkan perbandingan antara nilai aktual dan hasil prediksi untuk variabel BPN-2C. Terlihat bahwa garis prediksi (oranye) mengikuti tren nilai aktual (biru) dengan cukup baik, di mana keduanya menunjukkan pola kenaikan dan penurunan yang searah pada sebagian besar titik data. Meskipun terdapat beberapa perbedaan nilai pada beberapa observasi, secara umum model mampu menangkap variasi dan fluktuasi data aktual dengan cukup akurat, sehingga dapat dikatakan memiliki kemampuan prediksi yang baik terhadap pola data PDRB.

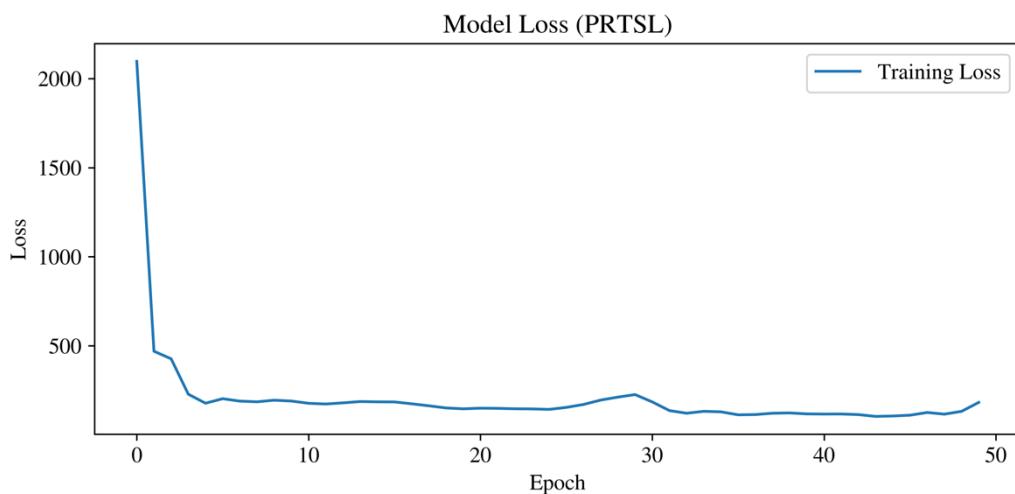
### 5.3.6 Pelatihan BPN-2F

Hasil percobaan BPN-2F menggunakan variable PRTSL yang menjadi target disajikan dalam Tabel 5.7.

**Tabel 5. 7 Proses Uji BPN-2F**

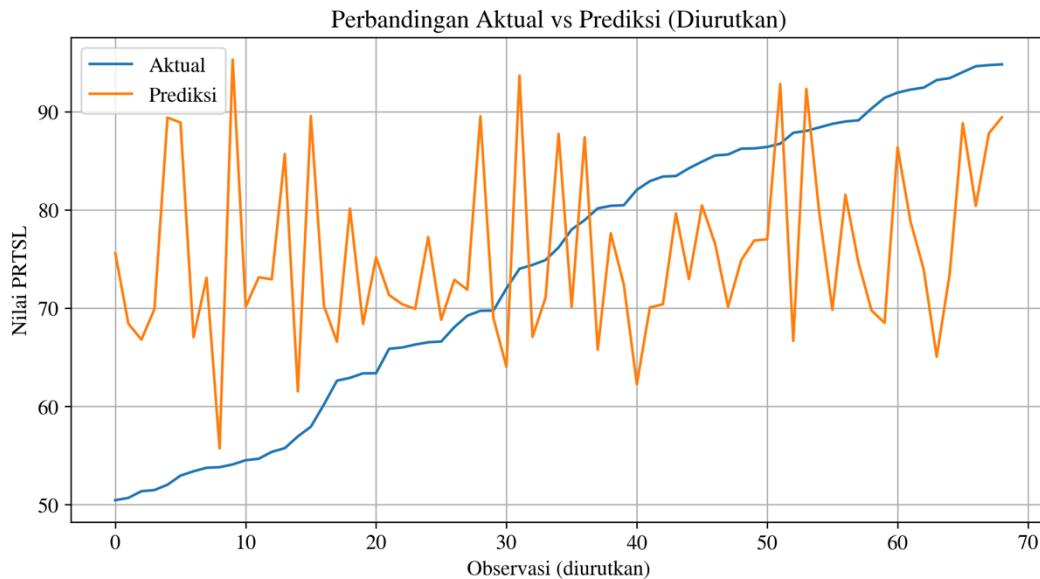
| Epoch | Time     | Training Loss | Validation Loss |
|-------|----------|---------------|-----------------|
| 1     | 3s 255ms | 2.097.037     | 4.538.287       |
| 2     | 0s 12ms  | 4.697.707     | 4.343.690       |
| 3     | 0s 12ms  | 4.271.173     | 2.758.001       |
| 4     | 0s 12ms  | 2.284.097     | 2.346.942       |
| 5     | 0s 13ms  | 1.778.098     | 2.345.423       |
| 46    | 0s 12ms  | 1.105.631     | 2.511.761       |
| 47    | 0s 12ms  | 1.257.606     | 2.410.489       |
| 48    | 0s 12ms  | 1.164.870     | 2.792.164       |
| 49    | 0s 14ms  | 1.321.365     | 2.799.578       |
| 50    | 0s 13ms  | 1.824.193     | 3.141.447       |

Berdasarkan Tabel 5.7, diketahui bahwa pelatihan model BPN-2F berlangsung dengan waktu eksekusi sekitar 13 ms dengan MSE senilai 158,58 dan MAPE senilai 18,44%. Proses pelatihan tersebut menghasilkan nilai *Training Loss* sebesar 1.824.193 dan nilai *Validation Loss* sebesar 3.141.447. Hasil pelatihan divisualisasikan dalam bentuk grafik, bisa dilihat pada Gambar 5.11.

**Gambar 5. 13 Nilai Loss Pelatihan BPN-2F**

Pada Gambar 5.11 ditampilkan grafik training loss pada model prediksi angka kemiskinan. Terlihat bahwa nilai loss awalnya sangat tinggi, namun mengalami penurunan tajam dan fluktuatif pada epoch awal, terlihat stabil pada

pertengahan epoch dan mulai mengalami fluktuatif di epoch ke-30. Hal ini mengindikasikan bahwa model masih mempelajari dan menyesuaikan bobot secara bertahap untuk mencapai kondisi yang lebih optimal.



**Gambar 5. 14 Perbandingan Data Aktual dan Prediksi BPN-2F**

Pada Gambar 5.12 ditampilkan perbandingan antara nilai aktual dan hasil prediksi untuk pelatihan BPN-2F. Terlihat bahwa garis prediksi secara umum mengikuti pola nilai aktual dengan arah perubahan yang serupa. Meskipun terdapat beberapa titik di mana nilai prediksi sedikit lebih tinggi atau lebih rendah dibandingkan nilai aktual, pola keseluruhan menunjukkan bahwa model mampu menangkap tren dan fluktuasi data dengan cukup baik. Hal tersebut menggambarkan model mempunyai kemampuan memperkirakan yang baik untuk merepresentasikan variasi nilai aktual, meskipun masih terdapat sedikit deviasi pada beberapa observasi tertentu.

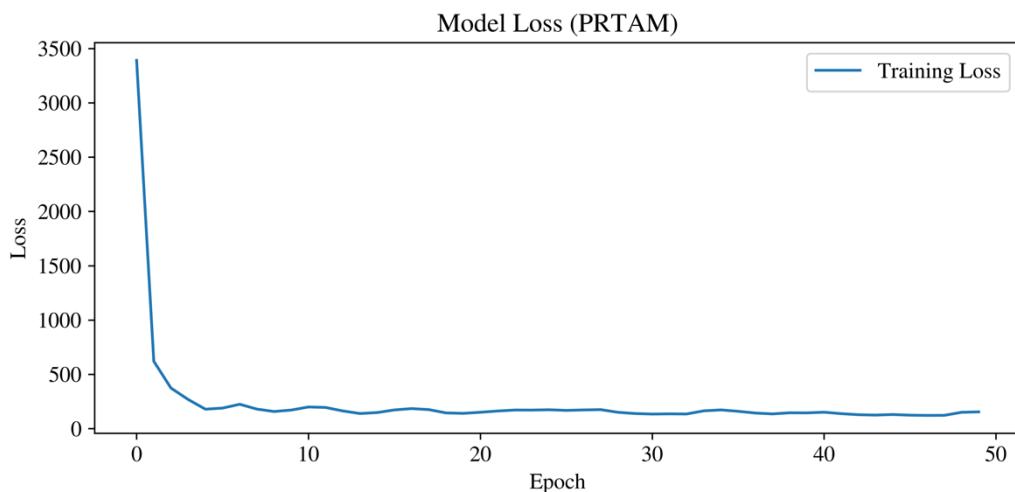
### 5.3.7 Pelatihan BPN-2G

Hasil percobaan BPN-2G menggunakan variable PRTAM yang menjadi target disajikan dalam Tabel 5.8.

**Tabel 5. 8 Proses Uji BPN-2G**

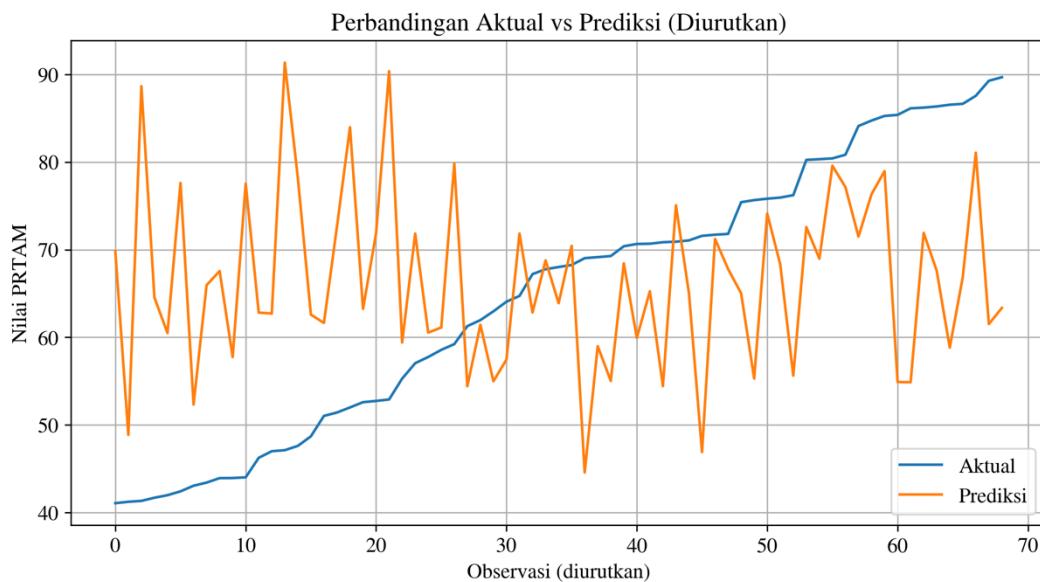
| Epoch | Time     | Training Loss | Validation Loss |
|-------|----------|---------------|-----------------|
| 1     | 3s 295ms | 3.390.404     | 1.268.595       |
| 2     | 0s 13ms  | 6.203.105     | 3.432.006       |
| 3     | 0s 14ms  | 3.736.602     | 3.467.810       |
| 4     | 0s 15ms  | 2.687.261     | 2.632.769       |
| 5     | 0s 13ms  | 1.796.190     | 2.895.244       |
| 46    | 0s 12ms  | 1.241.076     | 2.749.259       |
| 47    | 0s 12ms  | 1.220.912     | 2.941.206       |
| 48    | 0s 12ms  | 1.227.777     | 3.252.675       |
| 49    | 0s 12ms  | 1.506.447     | 4.385.191       |
| 50    | 0s 12ms  | 1.547.834     | 3.282.410       |

Berdasarkan Tabel 5.8, diketahui bahwa pelatihan model BPN-2G berlangsung dengan waktu eksekusi sekitar 12 ms dengan MSE senilai 229,60 dan MAPE senilai 21,73%. Proses pelatihan tersebut menghasilkan nilai *Training Loss* sebesar 1.547.834 dan nilai *Validation Loss* sebesar 3.282.410. Hasil pelatihan divisualisasikan dalam bentuk grafik, bisa diketahui dalam Gambar 5.13.

**Gambar 5. 15 Nilai Loss Pelatihan BPN-2G**

Pada Gambar 5.13 ditampilkan grafik training loss pada model prediksi angka kemiskinan. Terlihat bahwa nilai loss awalnya sangat tinggi, namun mengalami penurunan tajam pada epoch awal. Terlihat bahwa mulai dari epoch ke-

2 dan seterusnya mengalami penyesuaian bobot ditandai dengan grafik yang belum sepenuhnya stabil. Hal ini mengindikasikan bahwa model masih menyesuaikan bobot secara bertahap untuk mencapai kondisi yang lebih optimal.



**Gambar 5. 16 Perbandingan Data Aktual dan Prediksi BPN-2G**

Pada Gambar 5.14 ditampilkan perbandingan antara nilai aktual dan hasil prediksi untuk pelatihan BPN-2G. Terlihat bahwa garis prediksi secara umum mengikuti pola nilai aktual dengan arah perubahan yang relatif searah pada sebagian besar titik data. Namun, terdapat beberapa bagian di mana nilai prediksi cenderung sedikit lebih tinggi dibandingkan nilai aktual. Secara keseluruhan, pola grafik menunjukkan bahwa model mampu merepresentasikan tren dan fluktuasi data aktual dengan cukup baik, sehingga dapat dikatakan bahwa model memiliki kemampuan prediksi yang baik meskipun masih terdapat deviasi kecil pada beberapa observasi.

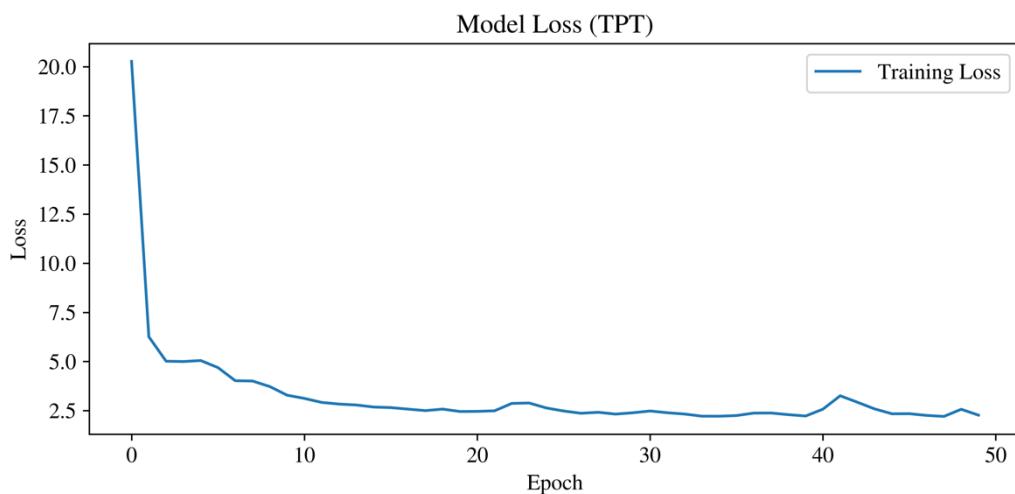
### 5.3.8 Pelatihan BPN-2H

Hasil percobaan BPN-2H menggunakan variable TPT yang menjadi target disajikan dalam Tabel 5.9.

**Tabel 5. 9 Proses Uji BPN-2H**

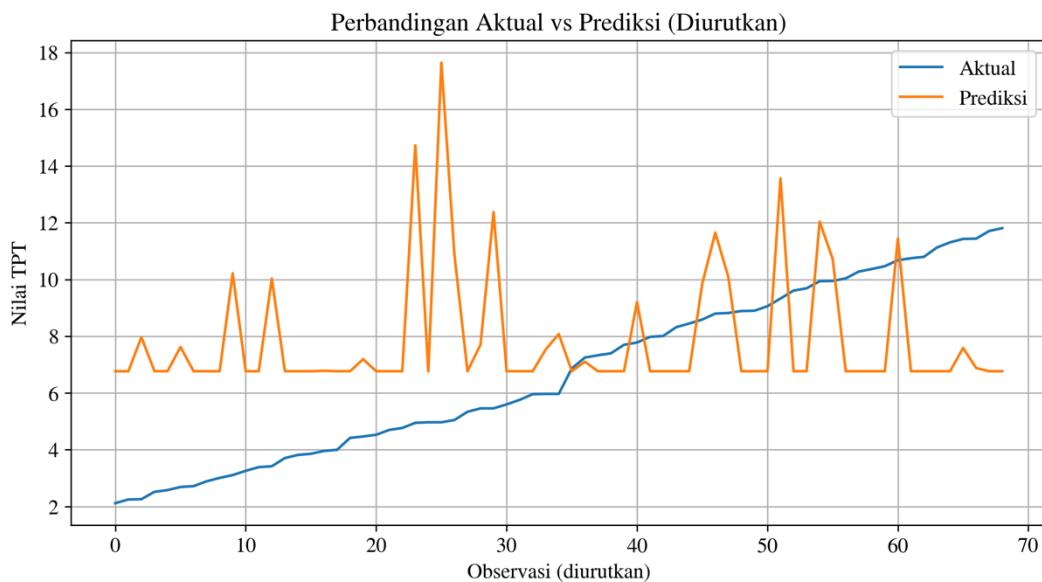
| Epoch | Time     | Training Loss | Validation Loss |
|-------|----------|---------------|-----------------|
| 1     | 2s 250ms | 2.027.163     | 7.524.944       |
| 2     | 0s 12ms  | 6.258.221     | 7.035.666       |
| 3     | 0s 13ms  | 5.014.710     | 8.567.695       |
| 4     | 0s 13ms  | 4.997.364     | 8.754.034       |
| 5     | 0s 13ms  | 5.049.349     | 7.981.068       |
| 46    | 0s 13ms  | 2.344.022     | 3.922.550       |
| 47    | 0s 13ms  | 2.255.961     | 4.262.139       |
| 48    | 0s 13ms  | 2.206.063     | 4.808.414       |
| 49    | 0s 13ms  | 2.568.895     | 3.652.538       |
| 50    | 0s 13ms  | 2.271.041     | 4.514.218       |

Berdasarkan Tabel 5.9, diketahui bahwa pelatihan model BPN-2H berlangsung dengan waktu eksekusi sekitar 13 ms dengan MSE senilai 3,60 dan MAPE senilai 2,66%. Proses pelatihan tersebut menghasilkan nilai *Training Loss* sebesar 2.271.041 dan nilai *Validation Loss* sebesar 4.514.218. Hasil pelatihan divisualisasikan dalam bentuk grafik, bisa diketahui dalam Gambar 5.15.

**Gambar 5. 17 Nilai Loss Pelatihan BPN-2H**

Pada Gambar 5.15 ditampilkan grafik training loss pada model prediksi angka kemiskinan. Terlihat bahwa nilai loss awalnya sangat tinggi, namun mengalami penurunan tajam pada epoch awal. Terlihat bahwa model mengalami

penyesuaian bobot ditandai dengan grafik yang belum sepenuhnya stabil dan fluktuatif. Hal ini mengindikasikan bahwa model masih menyesuaikan bobot secara bertahap untuk mencapai kondisi yang lebih optimal.



**Gambar 5. 18 Perbandingan Data Aktual dan Prediksi BPN-2H**

Pada Gambar 5.16 ditampilkan perbandingan antara nilai aktual dan hasil prediksi untuk pelatihan BPN-2H. Terlihat bahwa garis prediksi secara umum mengikuti pola nilai aktual dengan arah perubahan yang serupa di sebagian besar titik data. Meskipun terdapat beberapa perbedaan nilai, terutama pada beberapa puncak dan lembah data, pola keseluruhan menunjukkan bahwa model mampu menangkap tren dan fluktuasi nilai aktual dengan cukup baik. Hal ini mengindikasikan bahwa model memiliki kemampuan prediksi yang cukup baik, meskipun masih terdapat sedikit deviasi pada beberapa observasi tertentu.

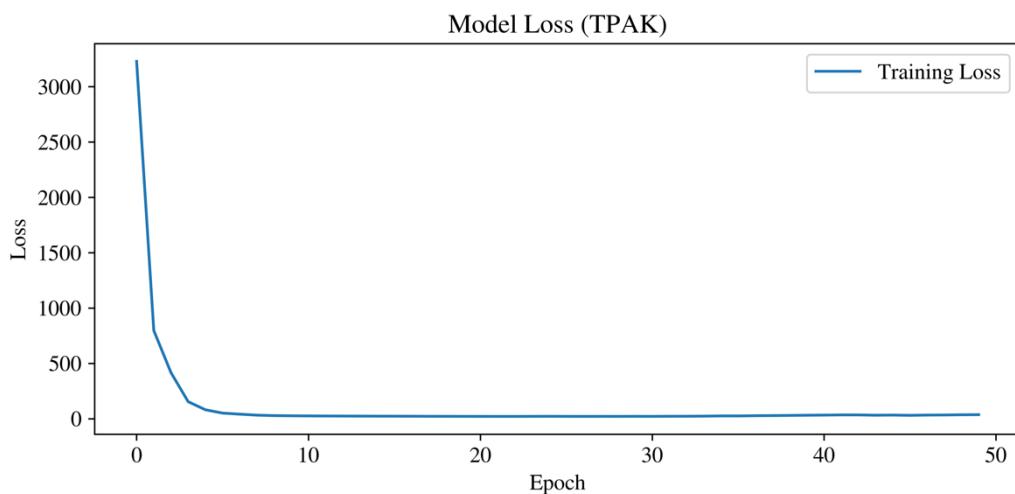
### 5.3.9 Pelatihan BPN-2I

Hasil percobaan BPN-2I menggunakan variable TPAK yang menjadi target disajikan dalam Tabel 5.10.

**Tabel 5. 10 Proses Uji BPN-2I**

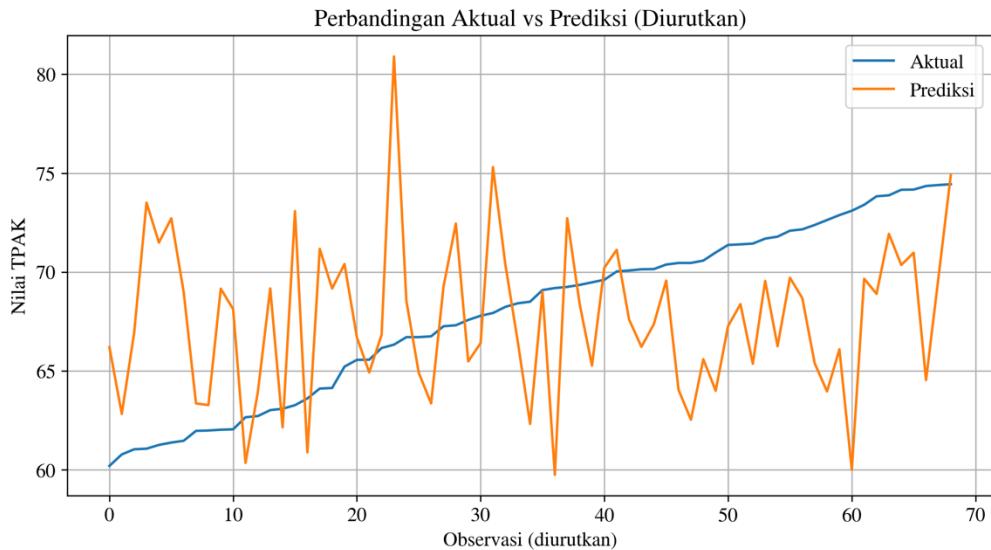
| Epoch | Time     | Training Loss | Validation Loss |
|-------|----------|---------------|-----------------|
| 1     | 4s 429ms | 3.228.864     | 5.675.247       |
| 2     | 0s 16ms  | 7.972.405     | 6.432.510       |
| 3     | 0s 17ms  | 4.170.216     | 1.881.488       |
| 4     | 0s 23ms  | 1.547.095     | 9.998.964       |
| 5     | 0s 20ms  | 8.178.572     | 6.177.878       |
| 46    | 0s 22ms  | 3.121.815     | 4.502.349       |
| 47    | 0s 15ms  | 3.364.881     | 2.951.655       |
| 48    | 0s 14ms  | 3.458.338     | 2.641.515       |
| 49    | 0s 14ms  | 3.644.922     | 2.069.797       |
| 50    | 0s 15ms  | 3.732.066     | 2.661.949       |

Berdasarkan Tabel 5.6, diketahui bahwa pelatihan model BPN-2I berlangsung dengan waktu eksekusi sekitar 15 ms dengan MSE senilai 32,51 dan MAPE senilai 14,05%. Proses pelatihan tersebut menghasilkan nilai *Training Loss* sebesar 3.732.066 dan nilai *Validation Loss* sebesar 2.661.949. Hasil pelatihan divisualisasikan dalam bentuk grafik, bisa diketahui dalam Gambar 5.17.

**Gambar 5. 19 Nilai Loss Pelatihan BPN-2I**

Pada Gambar 5.17 ditampilkan grafik training loss pada model prediksi angka kemiskinan. Terlihat bahwa nilai loss awalnya sangat tinggi, namun mengalami penurunan signifikan di epoch pertama. Setelah melewati sekitar epoch ke-1, pola loss menunjukkan kecenderungan stabil. Hal ini mengindikasikan bahwa

model sudah cukup baik dan mampu belajar dengan cepat di tahap awal, kemudian beradaptasi secara bertahap untuk mencapai kondisi yang lebih optimal, dengan tingkat kesalahan yang semakin kecil seiring bertambahnya epoch.



**Gambar 5. 20 Perbandingan Data Aktual dan Prediksi BPN-2I**

Pada Gambar 5.18 ditampilkan perbandingan antara nilai aktual dan hasil prediksi untuk pelatihan BPN-2I. Terlihat bahwa garis prediksi secara umum mengikuti pola nilai aktual dengan tren perubahan yang relatif searah pada sebagian besar titik data. Namun, terlihat bahwa nilai prediksi cenderung sedikit lebih tinggi dibandingkan nilai aktual di beberapa bagian grafik. Secara keseluruhan, pola ini menunjukkan bahwa model mampu menangkap tren dan fluktuasi data aktual dengan cukup baik, meskipun masih terdapat deviasi kecil pada beberapa titik ekstrem.

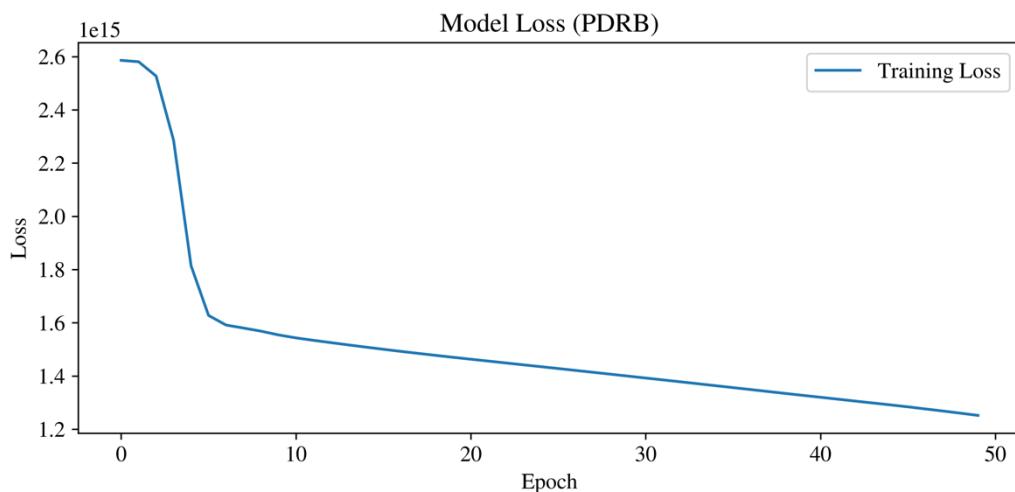
### 5.3.10 Pelatihan BPN-2J

Hasil percobaan BPN-2J menggunakan variable PDRB yang menjadi target disajikan dalam Tabel 5.11.

**Tabel 5. 11 Proses Uji BPN-2J**

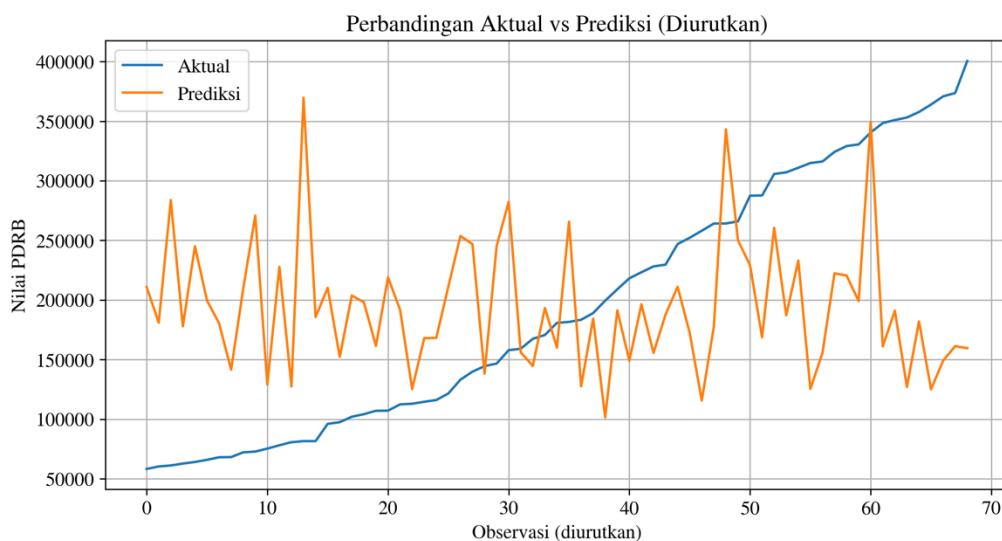
| Epoch | Time     | Training Loss | Validation Loss |
|-------|----------|---------------|-----------------|
| 1     | 2s 249ms | 2.586.069     | 6.495.199       |
| 2     | 0s 13ms  | 2.580.860     | 6.456.792       |
| 3     | 0s 12ms  | 2.527.075     | 6.203.696       |
| 4     | 0s 13ms  | 2.285.608     | 5.332.990       |
| 5     | 0s 13ms  | 1.814.129     | 4.150.413       |
| 46    | 0s 13ms  | 1.284.091     | 2.456.751       |
| 47    | 0s 12ms  | 1.276.160     | 2.430.230       |
| 48    | 0s 12ms  | 1.268.346     | 2.406.616       |
| 49    | 0s 12ms  | 1.260.287     | 2.382.013       |
| 50    | 0s 12ms  | 1.251.926     | 2.358.883       |

Berdasarkan Tabel 5.11, diketahui bahwa pelatihan model BPN-2J berlangsung dengan waktu eksekusi sekitar 12 ms dengan MSE senilai 1178873630818304.00 dan MAPE senilai 999999999.00%. Proses pelatihan tersebut menghasilkan nilai *Training Loss* sebesar 1.251.926 dan nilai *Validation Loss* sebesar 2.358.883. Hasil pelatihan divisualisasikan dalam bentuk grafik, bisa diketahui dalam Gambar 5.19.

**Gambar 5. 21 Nilai Loss Pelatihan BPN-2J**

Pada Gambar 5.5 ditampilkan grafik training loss pada model BPN-2J. Terlihat bahwa nilai loss pada awal pelatihan sangat tinggi, namun mengalami

penurunan yang tajam pada beberapa epoch pertama. Setelah itu, penurunan loss berlangsung secara lebih lambat namun konsisten hingga akhir epoch. Pola ini menunjukkan bahwa model mampu mempelajari pola data dengan baik sejak awal proses pelatihan dan terus melakukan penyesuaian bobot secara bertahap untuk mengoptimalkan kinerjanya, dengan tingkat kesalahan yang semakin menurun seiring bertambahnya epoch.



**Gambar 5. 22 Perbandingan Data Aktual dan Prediksi BPN-2J**

Pada Gambar 5.18 ditampilkan perbandingan antara nilai aktual dan hasil prediksi untuk pelatihan BPN-2J. Terlihat bahwa garis prediksi secara umum mengikuti pola nilai aktual dengan tren perubahan yang relatif searah pada sebagian besar titik data. Meskipun terdapat perbedaan pada beberapa titik, terutama pada nilai ekstrem yang menunjukkan lonjakan cukup tinggi, pola keseluruhan memperlihatkan bahwa model mampu menangkap tren dan fluktuasi data aktual dengan cukup baik. Hal ini menunjukkan bahwa model memiliki kemampuan prediksi yang baik, meskipun masih terdapat deviasi pada beberapa observasi dengan nilai yang sangat besar.

#### 5.4 Hasil Prediksi

Sesudah melaksanakan tahap pelatihan di setiap model, kemudian melakukan tahapan uji. Tahapan ini dilakukan dengan maksud untuk tau model

terbaik pada arsitektur kedua ini untuk melakukan prediksi pada angka kemiskinan menggunakan metode ANN. Adapun hasil uji dijabarkan dalam Tabel 5.12 berikut.

**Tabel 5. 12 Hasil Prediksi Model BPN 2 Hidden Layer**

| Pelatihan | Arsitektur | Time (ms) | MSE                 | MAPE (%)     |
|-----------|------------|-----------|---------------------|--------------|
| BPN-2A    | 13-6-9-1   | 13        | 32,12               | 13,5         |
| BPN-2B    |            | 12        | 4,71                | 3,09         |
| BPN-2C    |            | 10        | 1327162.12          | 999,99       |
| BPN-2D    |            | 12        | 2,74                | 2,18         |
| BPN-2E    |            | 13        | 5,03                | 4,82         |
| BPN-2F    |            | 12        | 158,58              | 18,44        |
| BPN-2G    |            | 12        | 229,6               | 21,73        |
| BPN-2H    |            | 13        | 3,6                 | 2,66         |
| BPN-2I    |            | 15        | 32,51               | 14,5         |
| BPN-2J    |            | 12        | 1178873630818304.00 | 999999999.00 |

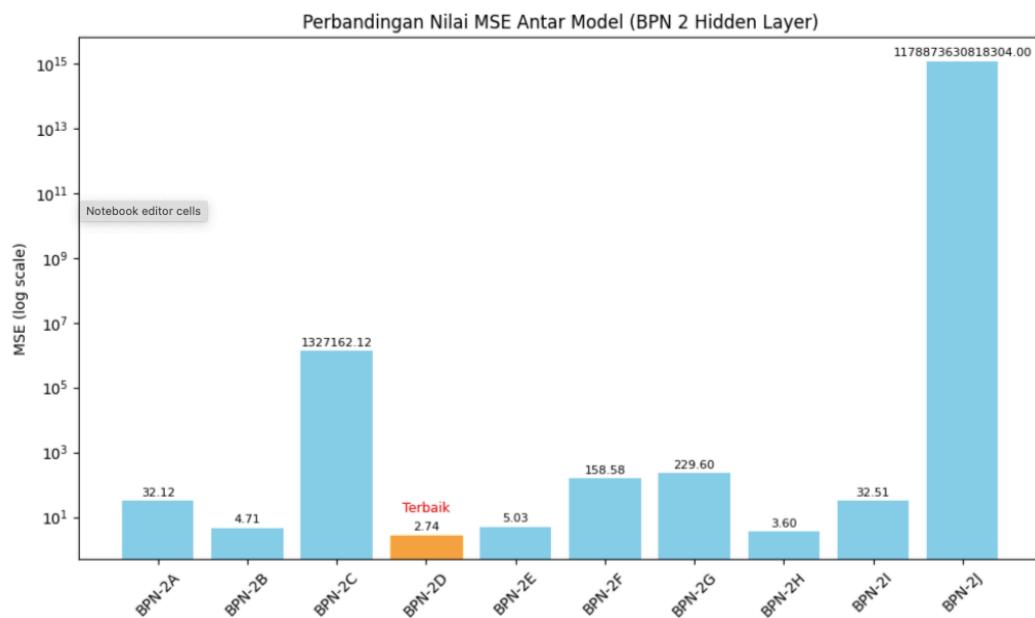
Berdasarkan hasil pelatihan yang ditampilkan pada Tabel 5.12, memperlihatkan hasil yang berbeda walaupun memiliki arsitektur atau struktur dasar yang sama yaitu 13-6-9-1. Mengacu pada pendapat (Wijaya Rauf, 2023) yang mengatakan bahwa semakin kecil nilai MSE yang dihasilkan, maka semakin baik pula kemampuan model dalam melakukan prediksi. Hal serupa juga berlaku pada nilai MAPE, di mana semakin kecil nilainya menunjukkan tingkat kesalahan prediksi yang semakin rendah dalam bentuk persentase.

Berdasarkan hal tersebut, pelatihan BPN-2D yang memiliki nilai MSE yaitu 2,47 dan MAPE 2,18% dapat dinyatakan sebagai model terbaik dalam memprediksi angka kemiskinan, karena memiliki tingkat kesalahan prediksi paling rendah. Hasil ini menunjukkan bahwa penggunaan dua hidden layer dalam jaringan saraf tiruan memberikan kemampuan pemetaan yang lebih baik terhadap hubungan nonlinier antar variabel input. Temuan ini sejalan dengan pendapat (Seidi et al., 2025) yang menyatakan bahwa arsitektur dengan 2 *hidden layer* sering kali memiliki kemampuan generalisasi yang lebih baik dibandingkan satu hidden layer

## 5.5 Kesimpulan

Hasil pelatihan di strategi kedua memperlihatkan pemakaian dua lapisan tersembunyi mampu meningkatkan kemampuan metode dalam memprediksi angka kemiskinan. Model ANN menggunakan algoritma propagasi balik diketahui efektif

untuk melakukan prediksi angka kemiskinan, seperti yang ditunjukkan oleh variasi nilai *Mean Squared Error* (MSE) dan *Mean Absolute Percentage Error* (MAPE) yang berbeda pada tiap model, di mana perbandingan nilai tersebut menjadi dasar dalam menentukan performa terbaik khususnya pada pelatihan BPN-2D. Adapun grafik hasil pelatihan dari strategi kedua yaitu 2 hidden layer bisa dilihat pada Gambar 5.21.



**Gambar 5. 23 Perbandingan performa model arsitektur 2 hidden layer terbaik**

Gambar 5.21 di atas memperlihatkan perbandingan nilai MSE dari semua arsitektur pelatihan dengan konfigurasi 13-6-9-1. Dari hasil tersebut terlihat bahwa sebagian besar model menghasilkan nilai MSE yang cukup bervariasi, dengan beberapa model menunjukkan kesalahan prediksi yang cukup tinggi. Namun, terdapat satu model yang menonjol, yaitu BPN-2D dengan nilai MSE sebesar 2,74 dan MAPE 2,18% yang merupakan nilai terkecil dibandingkan model lainnya. Sesuai dengan yang dikatakan oleh (Wijaya Rauf, 2023) bahwa semakin kecil nilai MSE yang dihasilkan, maka semakin baik pula kemampuan model dalam melakukan prediksi, Hal serupa juga berlaku pada nilai MAPE, di mana semakin kecil nilainya menunjukkan tingkat kesalahan prediksi yang semakin rendah dalam

bentuk persentase, sehingga BPN-2D dapat dinyatakan sebagai model terbaik pada strategi kedua ini.

## **BAB VI**

### **PEMBAHASAN**

#### **5.1 Analisis Perbandingan Arsitektur**

Proses pelatihan jaringan saraf tiruan pada penelitian ini dilakukan dengan dua strategi arsitektur, yaitu menggunakan 1 hidden layer dan 2 hidden layer. Setiap arsitektur memiliki jumlah input sebanyak 13 node yang menjelaskan jumlah parameter atau atribut data dari variabel indikator kemiskinan yang digunakan, sedangkan jumlah output hanya satu node, yaitu nilai prediksi angka kemiskinan. Perbedaan utama antara kedua strategi tersebut terletak pada jumlah hidden layer dan jumlah node yang digunakan di dalamnya yaitu strategi pertama menggunakan 1 hidden layer dan strategi kedua menggunakan 2 hidden layer. Setiap arsitektur diuji untuk melihat konsistensi hasil prediksi yang ditunjukkan oleh nilai *Mean Squared Error* (MSE) dan *Mean Absolute Percentage Error* (MAPE) serta waktu konvergensi selama proses pelatihan.

Kinerja model ANN dengan algoritma backpropagation tidak hanya ditentukan oleh banyaknya node dalam setiap *hidden layer*, tetapi juga oleh jumlah *hidden layer* yang digunakan dalam arsitektur jaringan. Semakin banyak hidden layer yang dimiliki model, proses pelatihannya akan melibatkan lebih banyak iterasi dan pembaruan bobot, sehingga pola hubungan antarvariabel dapat dipelajari dengan lebih mendalam. Kondisi ini dapat membantu model mencapai titik konvergensi lebih cepat, selama proses pembelajaran tetap stabil. Selain itu, peran atribut input sangat penting karena setiap indikator kemiskinan memiliki pengaruh berbeda terhadap nilai prediksi akhir. Dengan kata lain, kombinasi antara total lapisan tersembunyi serta pemilihan atribut input adalah tahapan untuk menentukan seberapa baik model mampu menghasilkan prediksi yang akurat. Adapun perbandingan performa kedua strategi arsitektur dalam penelitian ini dari segi waktu konvergensi, hasil dari MSE dan MAPE, bisa dilihat pada Tabel 6.1

**Tabel 6. 1 Waktu Konvergensi, MSE, dan MAPE**

| Variable Target | 1 Hidden Layer |                         |                   | 2 Hidden Layer |                         |                   |
|-----------------|----------------|-------------------------|-------------------|----------------|-------------------------|-------------------|
|                 | Time           | MSE                     | MAPE              | Time           | MSE                     | MAPE              |
| A               | 13ms           | 29,19                   | 12,87%            | 13ms           | 32,12                   | 13,5%             |
| B               | 12ms           | 4,94                    | 3,22%             | 12ms           | 4,71                    | 3,09%             |
| C               | 13ms           | 1085070.25              | 999,98%           | 10ms           | 1327162.12              | 999,99%           |
| D               | 16ms           | 4,4                     | 2,95%             | 12ms           | 2,74                    | 2,18%             |
| E               | 12ms           | 10,79                   | 5,41%             | 13ms           | 5,03                    | 4,82%             |
| F               | 14ms           | 195,18                  | 22,65%            | 12ms           | 158,58                  | 18,44%            |
| G               | 12ms           | 279,65                  | 25,37%            | 12ms           | 229,6                   | 21,73%            |
| H               | 14ms           | 4,57                    | 2,73%             | 13ms           | 3,6                     | 2,66%             |
| I               | 13ms           | 33,61                   | 14,12%            | 15ms           | 32,51                   | 14,5%             |
| J               | 12ms           | 1535238744<br>309760.00 | 9999999<br>99.00% | 12ms           | 117887363081<br>8304.00 | 999999999.0<br>0% |

Tabel 6.1 memperlihatkan variasi waktu pelatihan dan nilai MSE pada setiap arsitektur model yaitu 1 *hidden layer* dan 2 *hidden layer*. Diketahui bahwa hasil MSE antar model menunjukkan perbedaan, diantaranya pada model dengan 2 *hidden layer* cenderung memberikan hasil yang lebih baik, ditunjukkan oleh nilai MSE dan MAPE yang lebih kecil pada beberapa variabel target, terutama pada model BPN-2D dengan MSE sebesar 2,74 dan MAPE 2,18%. Dari sisi waktu pelatihan, kedua arsitektur tidak menunjukkan perbedaan signifikan karena rata-rata waktu konvergensi berada di kisaran 10–16 ms. Penambahan satu *hidden layer* tidak secara signifikan memperlambat proses pelatihan, namun justru mampu meningkatkan kemampuan model dalam mengenali pola data yang kompleks.

Pada beberapa pelatihan seperti BPN-1C dan BPN-1J menunjukkan nilai MSE dan MAPE yang sangat besar, mengindikasikan adanya ketidakstabilan proses pelatihan atau kesalahan pembelajaran (*overfitting* atau *underfitting*). Menurut (Pranatha et al., 2024), kondisi tersebut umumnya terjadi ketika model terlalu fokus pada data pelatihan sehingga kehilangan kemampuan untuk menggeneralisasi data baru. Sementara itu, (Nurdiawan et al., 2024) menambahkan bahwa waktu konvergensi yang terlalu lama dapat menjadi salah satu penyebab terjadinya overfitting.

## 5.2 Pandangan Islam terhadap Upaya Penanggulangan Kemiskinan

Dalam Islam, kemiskinan (*al-faqr*) bukan hanya persoalan ekonomi, tetapi juga merupakan masalah sosial yang memiliki dimensi moral dan spiritual. Islam memandang kemiskinan sebagai kondisi yang harus diatasi melalui tanggung jawab bersama, baik individu, masyarakat, maupun negara. Upaya mengentaskan kemiskinan dipandang sebagai bagian dari pelaksanaan nilai keadilan (*al-'adl*) dan kesejahteraan sosial (*al-maslahah al-'ammah*), yang merupakan tujuan utama dari syariat Islam (*maqāṣid al-syari‘ah*).

Allah Swt. menegaskan pentingnya memperhatikan dan membantu kaum miskin dalam firman-Nya:

وَفِي أَمْوَالِهِمْ حَقٌّ لِلْسَّائِلِ وَالْمَحْرُوفُمْ ⑯

Artinya:

“Dan pada harta-harta mereka ada hak untuk orang miskin yang meminta dan orang miskin yang tidak mendapat bagian.”(QS. *Adz-Dzāriyāt* [51]: 19)

Ayat ini menjelaskan bahwa setiap harta yang dimiliki seseorang memiliki hak pihak lain, terutama fakir dan miskin. Islam menekankan bahwa kekayaan tidak boleh dikumpulkan semata-mata untuk kepentingan pribadi. Keadilan sosial menuntut pembagian kekayaan agar setiap orang memperoleh haknya, sehingga kemiskinan dapat dikurangi secara sistematis.

Berdasarkan tafsir Quraisy Shihab, ayat ini menekankan bahwa distribusi kekayaan harus dilakukan dengan prinsip keadilan dan tanggung jawab sosial. Quraisy Shihab menjelaskan bahwa hak fakir dan miskin bukan sekadar kewajiban sosial, tetapi bagian dari struktur moral masyarakat yang menjamin kesejahteraan bersama. Dengan kata lain, ayat ini menekankan tema tematik: kesejahteraan sosial, pemerataan harta, dan pencegahan ketimpangan ekonomi.

Selain itu, Islam telah mengatur mekanisme distribusi kekayaan melalui kewajiban zakat, sebagaimana firman Allah Swt:

إِنَّمَا الصَّدَقَاتُ لِلْفُقَرَاءِ وَالْمَسْكِينِ وَالْعَمَلِينَ عَلَيْهَا وَالْمُؤْلَفَةِ قُلُوبُهُمْ وَفِي الرِّقَابِ وَالْغَرِيمِينَ ◻  
وَفِي سَبِيلِ اللَّهِ وَابْنِ السَّبِيلِ فَرِيضَةٌ مِّنَ اللَّهِ وَاللَّهُ عَلِيمٌ حَكِيمٌ ◻

Artinya:

“Sesungguhnya zakat-zakat itu hanyalah untuk orang-orang fakir, orang miskin, pengurus zakat, para muallaf yang dibujuk hatinya, untuk memerdekakan budak, orang-orang yang berhutang, untuk jalan Allah, dan untuk orang yang sedang dalam perjalanan, sebagai suatu ketetapan yang diwajibkan Allah; dan Allah Maha Mengetahui lagi Maha Bijaksana.” (QS. At-Taubah [9]: 60)

Ayat tersebut menjelaskan Islam memiliki sistem sosial yang sistematis dalam menanggulangi kemiskinan. Zakat bukan hanya ibadah ritual, tetapi juga instrumen ekonomi untuk mengurangi kesenjangan sosial dan menjaga keseimbangan distribusi harta di masyarakat. Islam juga menekankan pentingnya kerja keras, kemandirian, dan tanggung jawab sosial dalam mengatasi kemiskinan. Rasulullah SAW. bersabda: “Sebaik-baik harta adalah harta yang dimiliki oleh orang saleh.” (HR. Ahmad). Hadis ini mengandung makna bahwa kekayaan tidak dilarang dalam Islam, selama diperoleh dengan cara yang halal dan dimanfaatkan untuk kemaslahatan bersama. Oleh karena itu, Islam mendorong umatnya untuk berinovasi, bekerja keras, dan menggunakan ilmu pengetahuan sebagai sarana mencapai kesejahteraan.

Menurut Quraisy Shihab, ayat tentang zakat ini menekankan dimensi tematik sistem ekonomi Islam bertujuan mengurangi kemiskinan dan menegakkan keadilan sosial. Ia menyoroti peran institusi sosial (seperti zakat) dalam menyeimbangkan distribusi harta dan mencegah akumulasi kekayaan secara sewenang-wenang (Suharyat & Asiah, 2022). Tafsir ini relevan untuk konteks modern, termasuk penerapan teknologi seperti ANN dalam mengefektifkan distribusi sumber daya sosial.

Dalam konteks modern, nilai-nilai Islam tentang keadilan sosial dan pemerataan ekonomi dapat diimplementasikan melalui pemanfaatan ilmu pengetahuan dan teknologi yang berorientasi pada kemaslahatan umat (*maslahah mursalah*). Salah satu bentuknya adalah penerapan model *Artificial Neural Network (ANN)* untuk memprediksi angka kemiskinan. Teknologi ini dapat membantu pemerintah dan lembaga sosial dalam membuat kebijakan yang lebih akurat, efektif, dan tepat sasaran. Dengan demikian, penggunaan ANN dalam

penelitian ini sejalan dengan semangat Islam untuk menegakkan keadilan, efisiensi, dan akuntabilitas dalam pengelolaan sumber daya publik.

Pemanfaatan teknologi seperti ANN juga dapat dimaknai sebagai implementasi amanah dalam bekerja secara profesional dan bertanggung jawab. Hal ini sejalan dengan firman Allah Swt:

قالَتْ احْدِنُهُمَا يَأْبَتِ اسْتَأْجِرْهُ إِنَّ خَيْرَ مَنْ اسْتَأْجَرْتَ الْقَوِيُّ الْأَمِينُ (٢٨)

Artinya:

“Sesungguhnya orang yang paling baik yang kamu ambil untuk bekerja (mengelola sesuatu) ialah orang yang kuat lagi dapat dipercaya.” (QS. Al-Qashash [28]: 26)

Ayat tersebut menegaskan pentingnya kompetensi dan integritas dalam menjalankan tugas, termasuk dalam bidang penelitian dan pengelolaan data sosial. Dengan kemampuan ANN dalam mengidentifikasi pola kemiskinan secara lebih objektif, proses pengambilan kebijakan dapat dilakukan secara transparan dan berbasis bukti, sehingga membantu pemerintah dalam menyalurkan bantuan secara adil dan tepat sasaran.

Dari perspektif *maqāṣid al-syārī‘ah*, penelitian ini turut mendukung tujuan utama syariat, yaitu menjaga harta (*hifz al-māl*) dan menjaga jiwa (*hifz al-nafs*). Melalui sistem prediksi yang akurat, hak-hak masyarakat miskin dapat terlindungi, dan kebijakan yang diambil mampu mengurangi ketimpangan sosial. Dengan demikian, penerapan *Artificial Neural Network* dalam konteks penanggulangan kemiskinan bukan hanya inovasi teknologi, tetapi juga bentuk aktualisasi nilai Islam dalam mewujudkan kesejahteraan dan keadilan sosial bagi seluruh umat manusia.

## **BAB VII**

### **KESIMPULAN**

#### **7.1 Kesimpulan**

Berdasarkan hasil penelitian dan analisis yang telah dilakukan mengenai penerapan model *Artificial Neural Network (ANN)* dalam memprediksi angka kemiskinan di Indonesia, dapat disimpulkan hal-hal berikut:

1. Hasil pengujian menunjukkan bahwa setiap atribut data yang digunakan memiliki pengaruh yang berbeda terhadap hasil prediksi angka kemiskinan. Variabel-variabel indikator seperti pendidikan, pengangguran, kesehatan, dan pengeluaran rumah tangga berkontribusi terhadap variasi nilai *Mean Squared Error (MSE)* dan *Mean Absolute Percentage Error (MAPE)* yang dihasilkan. Misalnya, pada variabel D dan E, model dengan dua *hidden layer* menghasilkan nilai MSE dan MAPE yang jauh lebih kecil dibandingkan satu *hidden layer*, yaitu 2,74 dan 5,03 nilai MSE serta 2,18% dan 4,82% nilai MAPE pada dua *hidden layer*, dibandingkan dengan 4,40 dan 10,79 nilai MSE serta 2,95% dan 5,41% nilai MAPE pada satu *hidden layer*. Hal ini menunjukkan bahwa struktur jaringan yang lebih dalam mampu mengenali pola hubungan antaratribut dengan lebih efektif. Sementara itu, pada variabel A dan C, hasil MSE dan MAPE relatif lebih tinggi, masing-masing sebesar 29,19–32,12 dan 1.085.070,25–1.327.162,12 nilai MSE serta nilai MAPE yaitu 12,87% dan 13,5% dan 999,98%-999,99%, yang mengindikasikan bahwa atribut yang digunakan pada data tersebut memiliki kompleksitas atau tingkat variasi yang tinggi sehingga model memerlukan penyesuaian parameter lebih lanjut. Dengan demikian, dapat disimpulkan bahwa relevansi dan kualitas atribut input sangat berpengaruh terhadap kestabilan dan akurasi hasil prediksi model ANN.

Berdasarkan hasil pengujian terhadap dua arsitektur jaringan, yaitu 1 lapisan tersembunyi serta 2 lapisan tersembunyi, didapatkan bahwa model

dua *hidden layer* secara umum memberikan performa yang lebih baik. Hal ini terlihat dari nilai MSE yang lebih kecil pada sebagian besar variabel target, seperti D (2,74), E (5,03), F (158,58), G (229,60), dan H (3,60) dan MAPE yang juga bernilai sebanding yaitu D (2,18%), E (4,82%), F (18,44%), G (21,73%), dan H (2,66%). Nilai-nilai tersebut menunjukkan peningkatan akurasi dibandingkan dengan model satu *hidden layer* yang memiliki MSE masing-masing 4,40, 10,79, 195,18, 279,65, dan 4,57 dan MAPE masing-masing 2,95%, 5,4%, 22,65%, 25,37%, 2,73%. Dari sisi efisiensi, waktu konvergensi antar model relatif stabil, berkisar antara 10–16 milidetik (*ms*), sehingga penambahan satu *hidden layer* tidak menyebabkan penurunan kinerja secara signifikan. Dengan demikian, dapat disimpulkan bahwa arsitektur dengan dua *hidden layer* mampu memberikan keseimbangan antara akurasi prediksi dan kecepatan pelatihan, serta lebih efektif dalam mempelajari pola kompleks pada data kemiskinan.

## 7.2 Saran

Berdasarkan hasil penelitian yang telah dilakukan, maka dapat diberikan beberapa saran sebagai berikut:

1. Penelitian selanjutnya disarankan menggunakan data dengan cakupan yang lebih luas dan terbaru agar hasil prediksi kemiskinan menjadi lebih akurat.
2. Karena model dengan dua *hidden layer* menunjukkan hasil terbaik, penelitian berikutnya dapat mencoba pengaturan jumlah neuron atau fungsi aktivasi lain untuk meningkatkan kinerja model.

## DAFTAR PUSTAKA

- Adhitya, B., Prabawa, A., & Kencana, H. (2022). Analisis Pengaruh Pendidikan, Kesehatan, Sanitasi dan Rata-Rata Jumlah Anggota Keluarga Per Rumah Tangga terhadap Kemiskinan di Indonesia. *Ekonomis: Journal of Economics and Business*, 6(1), 288. <https://doi.org/10.33087/ekonomis.v6i1.501>
- Ajrana, A., Akbar, A., & Lawi, A. (2021). Implementasi Algoritma Deep Artificial Neural Network Menggunakan Mel Frequency Cepstrum Coefficient Untuk Klasifikasi Audio Emosi Manusia. *Proceeding KONIK (Konferensi Nasional Ilmu Komputer)*, 5, 66–73.
- Alfallah, F., Yuhandri, Y., & Sumijan, S. (2025). Penerapan Artificial Neural Network untuk Memprediksi Persediaan Obat Esensial. *Jurnal KomtekInfo*, 63–72. <https://doi.org/10.35134/komtekinfo.v12i1.630>
- Creswell, J. W. (2022). Desain Penelitian. *Indri Primayenti Nim. 11643201377*, 24.
- Cybenkot, G. (1989). Mathematics of Control, Signals, and Systems Approximation by Superpositions of a Sigmoidal Function\*. In *Math. Control Signals Systems* (Vol. 2).
- Delviera, A., & Amar, S. (2024). Pengaruh Tingkat Partisipasi Angkatan Kerja, Upah Minimum Dan Investasi Terhadap Kemiskinan Di Indonesia. *Media Riset Ekonomi Pembangunan (MedREP)*. <https://medrep.ppj.unp.ac.id/index.php/MedREP/login>
- Descania, D. Y. (2022). Prediksi Pertumbuhan Penduduk Di Kecamatan Baregbeg Kabupaten Ciamis Menggunakan Metode Artificial Neural Network. *Jurnal Informatika Dan Teknologi Pendidikan*, 2(2), 37–43. <https://doi.org/10.25008/jitp.v2i2.28>
- Edna Safitri, S., Triwahyuningtyas, N., & Sugianto, S. (2022). Faktor-Faktor yang Mempengaruhi Tingkat Kemiskinan Provinsi Banten. *SIBATIK JOURNAL: Jurnal Ilmiah Bidang Sosial, Ekonomi, Budaya, Teknologi, Dan Pendidikan*, 1(4), 259–274. <https://doi.org/10.54443/sibatik.v1i4.30>
- Fahrijal, R. (2024a). Penerapan Algoritma Backpropagation Conjugate Grandient Polak Ribiere Untuk Memprediksi Penduduk Miskin Pada Kabupaten/Kota di Provinsi Sumatera Barat. In *Jurnal JPILKOM* (Vol. 2, Issue 3). Online.
- Fahrijal, R. (2024b). Penerapan Algoritma Backpropagation Conjugate Grandient Polak Ribiere Untuk Memprediksi Penduduk Miskin Pada

- Kabupaten/Kota Di Provinsi Sumatera Barat. In *Jurnal JPILKOM* (Vol. 2, Issue 3). Online.
- Febiola Putri, F., Atus Amadi Putra, Yenni Kurniawati, & Zamahsary Martha. (2024a). Artificial Neural Network Model for Estimating the Poor Population in Indonesia as an Effort to Alleviate Poverty. *UNP Journal of Statistics and Data Science*, 2(2), 122–129. <https://doi.org/10.24036/ujsds/vol2-iss2/154>
- Febiola Putri, F., Atus Amadi Putra, Yenni Kurniawati, & Zamahsary Martha. (2024b). Artificial Neural Network Model for Estimating the Poor Population in Indonesia as an Effort to Alleviate Poverty. *UNP Journal of Statistics and Data Science*, 2(2), 122–129. <https://doi.org/10.24036/ujsds/vol2-iss2/154>
- Febrian, S. R., Sunarto, A. A., & Pambudi, A. (2024). Prediksi Penjualan Suku Cadang Motor Dengan Penerapan Random Forest Di Pt Terus Jaya Sentosa Motor. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 8(5), 10507–10513.
- Firmansyah, I., & Hayadi, B. H. (2022). Komparasi Fungsi Aktivasi Relu Dan Tanh Pada Multilayer Perceptron. *JIKO (Jurnal Informatika Dan Komputer)*, 6(2), 200–206.
- Jamil, J., & Pulukadang, S. (2025). Application of Deep Learning Method in Learning. *Formosa Journal of Sustainable Research*, 4(6), 1019–1038.
- Kumalasari, L. (2025). Jurnal J-MendiKKom (Jurnal Manajemen, Pendidikan dan Ilmu Komputer) Peningkatan Akurasi Pada Backpropagation Dengan Teknik Grid Search Terhadap Prediksi Kemiskinan Di Indonesia. In *Jurnal J-MENDIKKOM* (Vol. 2, Issue 1). Online.
- Kurniawan, D., & Apriani, S. (2025). Prediksi Angka Kemiskinan Di Kabupaten Ogan Komering Ulu Menggunakan Metode Data Mining Berbasis SVM, Neural Network & SGD. *Seminar Nasional Teknologi Informasi Dan Bisnis (SENATIB)*, 2025.
- Labach, A., Salehinejad, H., & Valaee, S. (2019). *Survey of Dropout Methods for Deep Neural Networks*. <http://arxiv.org/abs/1904.13310>
- Lestari, T. I., & Imaningsih, N. (2022). Analisis faktor-faktor yang mempengaruhi tingkat kemiskinan. In *Online) Jurnal Manajemen* (Vol. 14, Issue 4).
- Mateus Sinaga, L., & Pardingotan Sipayung, S. (2024). *Analisis dan Prediksi Persentase Angka Kemiskinan di Indonesia Menggunakan Metode Regresi Berganda Analisis dan Prediksi Persentase Angka Kemiskinan*

*di Indonesia menggunakan Metode Regresi Linier Berganda* (Vol. 06, Issue 02).

Muhammad Wicaksono Hasdyani Putra, & Sulaeman. (2022). *Determinant Of Poverty Level In East Java Province: Implications For Socio-Economic Policy After Covid-19 Pandemic*. 6(3).

Novita Sari, S., Gilang Pratama, B., & Prastowo, R. (2024). Pemodelan Artificial Neural Network (ANN) Untuk Identifikasi Bangunan Daerah Rawan Longsor. *Jurnal Device*, 14(1), 9–18.

Nurdiawan, O., Fathurrohman, F., & Faqih, A. (2024). Optimisasi Model Backpropagation untuk Meningkatkan Deteksi Kejang Epilepsi pada Sinyal Electroencephalogram. *Information System For Educators And Professionals: Journal of Information System*, 9(2), 151–160.

Oktariani, T. (2024). *Implementasi Kebijakan Rehabilitasi Anak Terlantar di Panti Sosial Bina Remaja Harapan Bengkulu*. <https://doi.org/10.62159/petahaha.xxxx.xxxx>

Oloyede, I., Abiodun, A. A., & Qaiser, A. (2024). Artificial Neural Network Regression Modelling of Poverty Index in Nigeria Artificial Neural Network Regression Modelling. *Journal of Statistical Modelling and Analytics*, 6(2), 1–10. <https://doi.org/10.22452/josma.vol6no2.4>

Paruthi, S., Husain, A., Alam, P., Khan, A. H., Hasan, M. A., & Magbool, H. M. (2022). A review on material mix proportion and strength influence parameters of geopolymer concrete: Application of ANN model for GPC strength prediction. *Construction and Building Materials*, 356, 129253.

Poerwanto, B., & Fajriani, F. (2020). Resilient Backpropagation Neural Network on Prediction of Poverty Levels in South Sulawesi. *MATRIX : Jurnal Manajemen, Teknik Informatika Dan Rekayasa Komputer*, 20(1), 11–18. <https://doi.org/10.30812/matrik.v20i1.726>

Pranatha, M. D. A., Setiawan, G. H., & Maricar, M. A. (2024). Utilization of ResNet Architecture and Transfer Learning Method in the Classification of Faces of Individuals with Down Syndrome. *Journal of Applied Informatics and Computing*, 8(2), 434–442.

Priseptian, L., & Priana Primandhana, W. (2022). Analisis faktor-faktor yang mempengaruhi kemiskinan. *Forum Ekonomi*, 24(1), 45–53. <http://journal.feb.unmul.ac.id/index.php/FORUMEKONOMI>

Purbowati, S., Dur, S., & Widyasari, R. (2023). Prediksi Jumlah Penduduk Miskin Di Sumatera Utara Dengan Metode Jaringan Syaraf Tiruan Backpropagation. *Justek: Jurnal Sains Dan Teknologi*, 6(4), 527–535.

- Purroy Vasquez, R., Aguilar Lasserre, A. A., Meza Palacios, R., & Fernández Lambert, G. (2024). Artificial neural network (ANN) in forecasting of poverty line and economic-energetic efficiencies into the maize-based agroecosystems. *Archives of Agronomy and Soil Science*, 70(1), 1–17. <https://doi.org/10.1080/03650340.2023.2287751>
- Purwanti, E. (2024). Analisis Deskriptif Profil Kemiskinan Indonesia Berdasarkan Data BPS Tahun 2023. *Jurnal Mahasiswa Humanis*, 4(1), 1–10.
- Rahmawati, I. (2021). Pengaruh dewan komisaris independen terhadap nilai perusahaan sub sektor perkebunan yang terdaftar di BEI. *Progress: Jurnal Pendidikan, Akuntansi Dan Keuangan*, 4(2), 96–106.
- Rahmi, A. (2024). Penerapan Jaringan Syaraf Tiruan Metode Backpropagation untuk Memprediksi Tingkat Pemahaman Siswa terhadap Mata Pelajaran Matematika Menggunakan Bahasa Pemrograman PHP dan Database MySQL. *Journal of Research and Investigation in Education*, 65–72. <https://doi.org/10.37034/residu.v2i2.174>
- Rifa'i, Y. (2023). Analisis Metodologi Penelitian Kulitatif dalam Pengumpulan Data di Penelitian Ilmiah pada Penyusunan Mini Riset. In *Jurnal Ilmu Sosial dan Humaniora* (Vol. 1, Issue 1). <https://glorespublication.org/index.php/ekodestinasi>
- S. Dai, S. I., Canon, S., & Bauty, D. O. (2023). Analisis Pengaruh RLS, Pengeluaran Perkapita, UHH, dan Tingkat Kemiskinan Terhadap Ketimpangan Distribusi Pendapatan di KBI & KTI. *Jesya*, 6(1), 535–544. <https://doi.org/10.36778/jesya.v6i1.950>
- Sahi, M., & Galib, G. R. H. (2025). Artificial Intelligence Application of Back-propagation Neural Network in Cryptocurrency Price Prediction. *IJEIE: International Journal of Electrical and Intelligent Engineering*, 1(1), 32–46.
- Sari Wulandari, Ahmad Prayendi Dasopang, Ginie Aulia Rawani, Idzni Hasfizetty, Muhammad Yunus Sofian, Reni Dwijaya, & Selfia Rachmalija. (2022). Kebijakan Anti Kemiskinan Program Pemerintah Dalam Penananggulangan Kemiskinan di Indonesia. *Jurnal Inovasi Penelitian*, 2.
- Seidi, E., Kaviari, F., & Miller, S. F. (2025). Hyperparameter Tuning of Artificial Neural Network-Based Machine Learning to Optimize Number of Hidden Layers and Neurons in Metal Forming. *Journal of Manufacturing and Materials Processing*, 9(8), 260. <https://doi.org/10.3390/jmmp9080260>

- Sinurat, R. P. P. (2023). Analisis Faktor-Faktor Penyebab Kemiskinan Sebagai Upaya Penanggulangan Kemiskinan di Indonesia. *Jurnal Registratie*, 5(2), 87–103. <https://doi.org/10.33701/jurnalregistratie.v5i2.3554>
- Sonang, S., Defit, S., & Ramadhan, M. (2021). *Analisis Optimasi Fungsi Pelatihan Machine Learning Neural Network dalam Peramalan Kemiskinan*. <https://www.bps.go.id/>.
- Suharyat, Y., & Asiah, S. (2022). Metodologi tafsir al-mishbah. *Jurnal Pendidikan Indonesia: Teori, Penelitian, Dan Inovasi*, 2(5).
- Supriana, W., Raharja, M. A., Made, I., & Bimantara, S. (2023). *Pengembangan Sistem Prediksi Bantuan Program Keluarga Harapan (PKH) Berbasis Machine Learning*. <https://doi.org/10.31598>
- Syaharuddin, Etika, P., Purnama Sari, I., Melia Mardika, V., & Putri, M. (2020). *Analisis Algoritma Back Propagation dalam Prediksi Angka Kemiskinan di Indonesia*. 3(1), 11–17. <https://doi.org/10.31764>
- Syahroni, M. I. (2022). Prosedur Penelitian Kuantitatif. *Jurnal Al-Musthafa STIT Al-Aziziyah Lombok Barat*, 43(3).
- Tarwoco, B. S., Faizin, N., Indriani, Y., & Widodo, E. (2023). *Faktor-Faktor yang Mempengaruhi Kemiskinan di Provinsi DI Yogyakarta Tahun 2019-2021 Menggunakan Regresi Data Panel*.
- Tribuana, D., Maramis, L., Resky, A. M., & Hidayat, R. (2025). *Deep Learning* (Yori Apridonal M, Ed.). Serasi Media Teknologi.
- Vita Ferezagia, D. (2018). Issue 1 Article 1 Recommended Citation Recommended Citation Ferezagia. In *Jurnal Sosial Humaniora Terapan* (Vol. 1, Issue 1).
- Widiastuti, N., Riauwanto, S., & Harwati, C. A. (2022). Variabel Intervening. *Jurnal Riset Akuntansi Dan Bisnis Indonesia STIE Widya Wiwaha*, 2(4), 1224–1242.
- Wijaya Rauf, B. (2023). Prediksi Penduduk Miskin di Daerah Tertinggal Indonesia dengan Algoritma Prophet. *Jurnal Ilmu Manajemen Sosial Humaniora (JIMSH)*, 5(2), 116–125. <https://doi.org/10.51454/jimsh.v5i2.1024>
- Wulansari, R. Y., Fadhilah, N., Huda, M., Abidin, A. Z., & Sujianto, A. E. (2023). Faktor Yang Mempengaruhi Kemiskinan di Indonesia. *Journal of Economic, Management, Accounting and Technology*, 6(1), 82–95. <https://doi.org/10.32500/jematech.v6i1.3928>

Yudistira, E. N. (2024). *Deep Learning: Teori, Contoh Perhitungan, dan Implementasi*. Deepublish.