

**PREDIKSI CURAH HUJAN BULANAN BERBASIS INDEKS IKLIM
GLOBAL DI PACET MENGGUNAKAN *ENSEMBLE
MULTI-LAYER PERCEPTRON***

TESIS

**Oleh:
DWI DEFINTA OKTAVIA SISWOYO
NIM. 220605220012**



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

**PREDIKSI CURAH HUJAN BULANAN BERBASIS INDEKS IKLIM
GLOBAL DI PACET MENGGUNAKAN *ENSEMBLE*
*MULTI-LAYER PERCEPTRON***

TESIS

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

**Oleh:
DWI DEFINTA OKTAVIA SISWOYO
NIM. 220605220012**

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

**PREDIKSI CURAH HUJAN BULANAN BERBASIS INDEKS IKLIM
GLOBAL DI PACET MENGGUNAKAN *ENSEMBLE*
*MULTI-LAYER PERCEPTRON***

TESIS

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

**Oleh:
DWI DEFINTA OKTAVIA SISWOYO
NIM. 220605220012**

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

**PREDIKSI CURAH HUJAN BULANAN BERBASIS INDEKS IKLIM
GLOBAL DI PACET MENGGUNAKAN *ENSEMBLE
MULTI-LAYER PERCEPTRON***

TESIS

**Oleh:
DWI DEFINTA OKTAVIA SISWOYO
NIM. 220605220012**

Telah diperiksa dan disetujui untuk di uji:

Tanggal: 19 November 2025

Pembimbing I,

Dr. Irwanto Budi Santoso, M.Kom.
NIP 19770103 201101 1 004

Pembimbing II

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

Mengetahui,

Ketua Program Studi Magister Informatika

Fakultas Sains dan Teknologi

Universitas Islam Alauddin Malik Ibrahim Malang



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

**PREDIKSI CURAH HUJAN BULANAN BERBASIS INDEKS IKLIM
GLOBAL DI PACET MENGGUNAKAN ENSEMBLE
MULTI-LAYER PERCEPTRON**

TESIS

**Oleh:
DWI DEFINTA OKTAVIA SISWOYO
NIM. 22060522012**

Telah Dipertahankan Di Depan Dewan Penguji Tesis
Dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
Untuk Memperoleh Gelar Magister Komputer (M.Kom)
Tanggal: 19 November 2025

Susunan Dewan Penguji

Penguji I	:	<u>Dr. Cahyo Crysdiar, M.Cs</u> NIP 19740424 200901 1 008
Penguji II	:	<u>Dr. M. Amin Hariyadi, M.T</u> NIP. 19670018 200501 1 001
Pembimbing I	:	<u>Dr. Irwan Budi Santoso, M.Kom</u> NIP 19770103 201101 1 004
Pembimbing II	:	<u>Prof. Dr. Ir. M. Faisal, S.Kom, M.T</u> NIP 19740510 200501 1 007

Tanda Tangan

()
()
()
()

Mengetahui,

Ketua Program Studi Magister Informatika

Fakultas Sains dan Teknologi

Universitas Islam Negeri Maulana Malik Ibrahim Malang



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

PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini:

Nama : Dwi Definta Oktavia Siswoyo

NIM : 220605220012

Program Studi : Magister Informatika

Fakultas : Sains dan Teknologi

Menyatakan dengan sebenar-benarnya 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 tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar Pustaka.

Apabila di kemudian hari terbukti atau dapat dibuktikan Tesis ini hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, November 2025

Yang menyatakan,

A handwritten signature in black ink is written over a yellow 10000 Rupiah postage stamp. The stamp features the Garuda Pancasila emblem and the text '10000', 'METERA', and 'TEMPER'. The serial number '5Q182/ANX130705617' is visible at the bottom of the stamp.

Dwi Definta Oktavia Siswoyo

NIM. 220605220012

MOTTO

“Setiap proses belajar memperluas cara berpikir dan memperdalam pemahaman terhadap ilmu, sebagai fondasi dalam membangun pengetahuan yang utuh dan bermakna.”

PERSEMBAHAN

Dengan segenap rasa syukur kepada Allah SWT, tesis ini saya persembahkan kepada:

1. Seluruh keluarga saya (suami, keempat orangtua, dan 2 anak perempuan) yang selalu memberikan doa, dukungan, dan semangat tanpa henti.
2. Dosen pembimbing dan seluruh pengajar, yang telah sabar membimbing, memberi ilmu, dan berbagi pengalaman yang begitu berharga, sehingga saya bisa belajar lebih banyak dalam bidang ilmu ini.
3. Rekan-rekan angkatan 7 Magister Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang yang telah mendukung, berbagi pengetahuan, ide, dan dukungan moral selama ini.
4. Bapak, Ibu, rekan-rekan dan semua pihak yang tidak dapat saya sebutkan satu per satu yang telah membantu saya selama masa studi ini.

Semoga karya ini dapat bermanfaat dan menjadi langkah awal bentuk kontribusi nyata bagi lingkungan dan ilmu pengetahuan.

KATA PENGANTAR

Assalamu 'alaikum warahmatullah wabarakatuh

Puji syukur, alhamdulillah, kehadiran Allah SWT atas limpahan rahmat dan kasih sayangNya, sehingga penulis dapat menyelesaikan studi di Program Studi Magister Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang sekaligus menyelesaikan Tesis ini dengan baik. Penulis mengucapkan terima kasih sebesar-besarnya kepada semua pihak yang telah membantu terselesaikannya Tesis ini. Ucapan terima kasih ini penulis sampaikan kepada:

1. Bapak Dr. Irwan Budi S., M.Kom. dan Bapak Prof. Dr. Ir. M. Faisal, S.Kom. M.T selaku dosen pembimbing Tesis atas arahan dan motivasi yang telah diberikan,
2. Segenap civitas akademika Program Studi Magister Informatika atas ilmu dan bimbingannya,
3. Keluarga tercinta atas dukungan dan doa yang selalu dilangitkan, dan
4. Semua rekan-rekan Magister Informatika, baik angkatan 7 maupun senior-junior yang berkenan berbagi pengalaman dan memberi masukan.

Penulis menyadari bahwa penyusunan Tesis ini masih mengandung kekurangan. Meski demikian, penulis berharap semoga Tesis ini dapat memberi manfaat kepada para pembaca, khususnya pihak-pihak yang menjadi tujuan penyelenggaraan penelitian ini, dan penulis secara pribadi.

Wassalamu 'alaikum warahmatullah wabarakatuh.

Malang, November 2025

Penulis

DAFTAR ISI

HALAMAN JUDUL	 i
HALAMAN PENGAJUAN	 ii
HALAMAN PERSETUJUAN.....	 iv
HALAMAN PENGESAHAN	 v
HALAMAN PERNYATAAN.....	Error! Bookmark not defined.
MOTTO	 vii
PERSEMBAHAN.....	 viii
KATA PENGANTAR.....	 ix
DAFTAR ISI.....	 x
DAFTAR GAMBAR.....	 xiii
DAFTAR TABEL.....	 xvi
ABSTRAK	 xviii
ABSTRACT	 xix
مستخلص البحث.....	 xx
BAB I.....	 1
PENDAHULUAN.....	 1
1.1 Latar Belakang.....	 1
1.2 Pernyataan Masalah.....	 5
1.3 Tujuan Penelitian.....	 5
1.4 Ruang Lingkup Penelitian	 6
1.5 Manfaat Penelitian.....	 6
BAB II	 8
STUDI PUSTAKA	 8
2.1 Model Prediksi Curah Hujan.....	 8
2.2 Kerangka Teoritis	 19
BAB III.....	 25
MODEL KONSEPTUAL	 25
3.1 Kerangka Konseptual	 25
3.2 Desain Penelitian.....	 27
3.2.1 Pengumpulan Data.....	 27

3.2.2 Desain Sistem dan Implementasi.....	34
3.2.3.1 Prediksi Menggunakan MLP (<i>Multi-Layer Perceptron</i>)	34
3.2.3.2 Prediksi Menggunakan <i>Ensemble</i>	53
3.2.3.2 Implementasi.....	57
3.2.3 Eksperimen	60
3.3 Instrumen Penelitian.....	62
BAB IV	64
MLP UNTUK PREDIKSI CURAH HUJAN.....	64
4.1 Prediksi Curah Hujan Menggunakan MLP 1 <i>Hidden Layer</i>	64
4.1.1 Verifikasi Parameter Kontigensi BMKG Skenario MLP 1 <i>Hidden Layer</i>	67
4.1.2 Analisis Perbandingan Performa <i>Training</i> dan <i>Testing</i> Skenario MLP 1 <i>Hidden Layer</i>	68
4.2 Prediksi Curah Hujan Menggunakan Arsitektur MLP 2 <i>Hidden Layer</i>	70
4.2.1 Verifikasi Parameter Kontigensi BMKG dengan Skenario MLP 2 <i>Hidden Layer</i>	74
4.2.2 Analisis Perbandingan Performa <i>Training</i> dan <i>Testing</i> Skenario MLP 2 <i>Hidden Layer</i>	75
4.3 Prediksi Curah Hujan Menggunakan MLP 3 <i>Hidden Layer</i>	77
4.3.1 Verifikasi Parameter Kontigensi BMKG Pada Skenario MLP 3 <i>Hidden Layer</i>	80
4.3.2 Analisis Perbandingan Performa <i>Training</i> dan <i>Testing</i> Skenario MLP 3 <i>Hidden Layer</i>	81
4.4 Seleksi Model	83
BAB V.....	86
ENSEMBLE MLP UNTUK PREDIKSI CURAH HUJAN.....	86
5.1 Prediksi Curah Hujan Menggunakan <i>Ensemble</i> MLP.....	86
BAB VI.....	94
PERBANDINGAN MODEL MLP DENGAN ENSEMBLE MLP.....	94
6.1 Verifikasi Model MLP	94
6.2 Verifikasi Model <i>Ensemble Bagging</i>	98
BAB VII.....	103
PENUTUP.....	103
7.1 Kesimpulan.....	103
7.2 Saran	104

DAFTAR PUSTAKA	106
-----------------------------	------------

DAFTAR GAMBAR

Gambar 2.1 Kerangka Teoritis	20
Gambar 3.1 Kerangka Konseptual Penelitian	25
Gambar 3.2 Desain Penelitian	27
Gambar 3.3 Peta wilayah Indeks ENSO	29
Gambar 3.4 Anomali Suhu Muka Laut	30
Gambar 3.5 Wilayah Monsunal Indonesia	31
Gambar 3.6 Skema Proses <i>Training</i> dan <i>Testing</i>	33
Gambar 3.7 Desain Sistem Penelitian Yang Dilakukan.	34
Gambar 3.8 Arsitektur <i>MLP 1</i>	36
Gambar 3.9 <i>Flowchart MLP 1 Hidden Layer</i>	36
Gambar 3.10 Arsitektur <i>MLP 2</i>	41
Gambar 3.11 <i>Flowchart MLP 2 Hidden Layer</i>	42
Gambar 3.12 Arsitektur <i>MLP 3</i>	47
Gambar 3.13 <i>Flowchart MLP 3 Hidden Layer</i>	47
Gambar 3.14 <i>Flowchart Ensemble Bagging MLP</i>	54
Gambar 4.1 Grafik Perbandingan Hasil Prediksi Arsitektur <i>MLP 1 Hidden Layer</i> (1 <i>neuron</i>) dengan Data CHIRPS	64
Gambar 4.2 Grafik Perbandingan Hasil Prediksi Arsitektur <i>MLP 1 Hidden Layer</i> (2 <i>neuron</i>) dengan Data CHIRPS	65
Gambar 4.3 Grafik Perbandingan Hasil Prediksi Arsitektur <i>MLP 1 Hidden Layer</i> (1 <i>neuron</i>) dengan Data CHIRPS	65

Gambar 4.4 Grafik Perbandingan Hasil Prediksi Arsitektur MLP 1 <i>Hidden Layer</i> (1 <i>neuron</i>) dengan Data CHIRPS	66
Gambar 4.5 Grafik Perbandingan Hasil Prediksi Arsitektur MLP 1 <i>Hidden Layer</i> (1 <i>neuron</i>) dengan Data CHIRPS	66
Gambar 4.6 Grafik Perbandingan Hasil Prediksi Arsitektur MLP 2 <i>Hidden Layer</i> (<i>neuron</i> 1,1) dengan Data CHIRPS.....	71
Gambar 4.7 Grafik Perbandingan Hasil Prediksi Arsitektur MLP 2 <i>Hidden Layer</i> (<i>neuron</i> 2,2) dengan Data CHIRPS.....	71
Gambar 4.8 Grafik Perbandingan Hasil Prediksi Arsitektur MLP 2 <i>Hidden Layer</i> (<i>neuron</i> 3,3) dengan Data CHIRPS.....	72
Gambar 4.9 Grafik Perbandingan Hasil Prediksi Arsitektur MLP 2 <i>Hidden Layer</i> (<i>neuron</i> 4,4) dengan Data CHIRPS.....	72
Gambar 4.10 Grafik Perbandingan Hasil Prediksi Arsitektur MLP 2 <i>Hidden Layer</i> (<i>neuron</i> 5,5) dengan Data CHIRPS.....	73
Gambar 4.11 Grafik Perbandingan Hasil Prediksi Arsitektur MLP 3 <i>Hidden Layer</i> (<i>neuron</i> 1,1,1) dengan Data CHIRPS.....	77
Gambar 4.12 Grafik Perbandingan Hasil Prediksi Arsitektur MLP 3 <i>Hidden Layer</i> (<i>neuron</i> 2,2,2) dengan Data CHIRPS.....	78
Gambar 4.13 Grafik Perbandingan Hasil Prediksi Arsitektur MLP 3 <i>Hidden Layer</i> (<i>neuron</i> 3,3,3) dengan Data CHIRPS.....	78
Gambar 4.14 Grafik Perbandingan Hasil Prediksi Arsitektur MLP 3 <i>Hidden Layer</i> (<i>neuron</i> 4,4,4) dengan Data CHIRPS.....	79

Gambar 4.15 Grafik Perbandingan Hasil Prediksi Arsitektur MLP 3 <i>Hidden Layer</i> (<i>neuron</i> 5,5,5) dengan Data CHIRPS.....	79
Gambar 4.16 Penentuan Rekomendasi Model.....	83
Gambar 6.1 Tren Parameter Verivikasi Model MLP terhadap Jumlah <i>Hidden Layer</i>	96
Gambar 6.2 Tren Parameter Verivikasi Pada Model <i>Ensemble</i>	100

DAFTAR TABEL

Tabel 2.1 Daftar Penelitian Sebelumnya Yang Dijadikan Acuan.....	21
Tabel 3.1 Pembagian Data <i>Training</i> dan Data <i>Testing</i>	33
Tabel 3.2 Sampel <i>Output</i> Arsitektur MLP 1	60
Tabel 3.3 Sampel <i>Output</i> Arsitektur MLP 2.....	60
Tabel 3.4 Sampel <i>Output</i> Arsitektur MLP 3.....	61
Tabel 3.5 Rekomendasi Model Terbaik Prediksi Curah Hujan	61
Tabel 4.1 Hasil Verifikasi Skenario MLP 1 Menggunakan Kontigensi BMKG	67
Tabel 4.2 Performa Model MLP 1 <i>Hidden Layer</i> pada Data <i>Training</i> dan <i>Testing</i> ...	69
Tabel 4.3 Hasil Verifikasi Skenario MLP 2 Menggunakan Kontigensi BMKG	74
Tabel 4.4 Performa Model MLP 2 <i>Hidden Layer</i> pada Data <i>Training</i> dan <i>Testing</i> ...	75
Tabel 4.5 Hasil Verifikasi Skenario MLP 3 Menggunakan Kontigensi BMKG	80
Tabel 4.6 Performa Model MLP 3 <i>Hidden Layer</i> pada Data <i>Training</i> dan <i>Testing</i> ...	82
Tabel 4.7 Rekomendasi Arsitektur Tiap Skenario.....	84
Tabel 5.1 Data Sampel Prediksi Curah Hujan Hasil <i>Resample with Replacement</i> Pada <i>Ensemble Bagging</i>	87
Tabel 5.2 Data Sampel Konversi Biner Prediksi Probabilitas Curah Hujan Setelah Diresample	88
Tabel 5.3 <i>Sample</i> Hasil Verifikasi <i>Ensemble</i> MLP Menggunakan Parameter Kontigensi BMKG	90
Tabel 5.4 Hasil Verifikasi Model <i>Ensemble Bagging</i> Menggunakan Parameter Kontigensi	91

Tabel 6.1 Hasil Verifikasi Menggunakan Kontigensi BMKG Tiap Skenario Model	95
Tabel 6.2. Hasil Verifikasi <i>Ensemble Bagging</i> dan Model Terbaik MLP	99

ABSTRAK

Siswoyo, Dwi Definta Oktavia. 2025. Prediksi Curah Hujan Bulanan Berbasis Indeks Iklim Di Pacet Menggunakan Ensemble MLP. Thesis. Program Studi Magister Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Dr. Irwan Budi S., M.Kom. (II) Prof. Dr. M. Faisal, M.T.

Kata kunci: curah hujan, anomali, longsor, MLP, ensemble bagging,

Kecamatan Pacet, Kabupaten Mojokerto, merupakan wilayah dataran tinggi dengan kemiringan lereng curam dan struktur tanah yang gembur, sehingga memiliki tingkat kerentanan tinggi terhadap bencana tanah longsor. Oleh karena itu, prediksi curah hujan bulanan menjadi langkah penting dalam upaya mitigasi bencana. Penelitian ini bertujuan untuk memprediksi curah hujan bulanan di Pacet menggunakan pendekatan Artificial Neural Network (ANN) berbasis Multilayer Perceptron (MLP) dengan skema ensemble bagging dan variabel input berupa indeks iklim global dan variabel target merupakan data curah hujan CHIRPS yang telah terbukti akurat di wilayah Jawa Timur. Model dasar (base learner) yang digunakan dalam ensemble merupakan arsitektur MLP terbaik dengan konfigurasi dual hidden layer (1–1) yang sebelumnya diperoleh dari hasil evaluasi berbagai skenario arsitektur. Skema ensemble bagging dilakukan dengan teknik bootstrap resampling sebanyak 100 kali (resampling with replacement), di mana setiap hasil prediksi model dirata-ratakan untuk memperoleh keluaran akhir. Hasil prediksi tersebut kemudian diverifikasi menggunakan tabel kontingensi BMKG (akurasi, false alarm, dan miss classification) terhadap ambang curah hujan ekstrem >500 mm per bulan berdasarkan ambang probabilitas 50%, 60%, 70%, 80%, dan 90%,. Hasil penelitian menunjukkan bahwa model MLP tunggal mencapai akurasi 88,4%, dengan false alarm 22,8% dan miss classification 29,3%. Selanjutnya, MLP ensemble bagging menunjukkan peningkatan kinerja, mencapai akurasi tertinggi 89,0%, false alarm 22,2% dan miss classification terendah 26,3%. pada ambang batas Pa50, sementara false alarm terendah 20,0% diperoleh pada ambang batas Pa70. Temuan ini menunjukkan bahwa MLP ensemble bagging meningkatkan stabilitas dan keandalan prediktif dibandingkan dengan model MLP tunggal. Pendekatan ini mampu meningkatkan keandalan prediksi anomali curah hujan bulanan, sehingga dapat digunakan sebagai alat bantu dalam sistem peringatan dini bencana longsor di wilayah Pacet.

ABSTRACT

Siswoyo, Dwi Definta Oktavia. 2025. **Monthly Rainfall Prediction Based on Climate Index in Pacet Using MLP Ensemble**. Theses. Program Studi Magister Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Promotor: (I) Dr. Irwan Budi S., M.Kom. (II) Prof. Dr. M. Faisal, M.T.

Key words: Rainfall prediction, anomaly, landslide, MLP, *ensemble bagging*,

Pacet District, located in Mojokerto Regency, is a highland area characterized by steep slopes and loose soil structure, resulting in a high level of vulnerability to landslides. Therefore, predicting monthly rainfall becomes an essential step in disaster mitigation efforts. This study aims to predict monthly rainfall in Pacet using an Artificial Neural Network (ANN) based on a Multilayer Perceptron (MLP) with an ensemble bagging scheme. The input variables consist of global climate indices, while the target variable uses CHIRPS rainfall data, which has been proven accurate for the East Java region. The base learner used in the ensemble is the best-performing MLP architecture with a dual hidden layer configuration (1–1), selected based on the evaluation of various architectural scenarios. The ensemble bagging scheme is implemented through 100 bootstrap resampling iterations (resampling with replacement), and the prediction outputs of all models are averaged to obtain the final result. The predictions are then verified using BMKG's contingency table metrics (accuracy, false alarm, and miss classification) against an extreme monthly rainfall threshold of >500 mm, evaluated at probability thresholds of 50%, 60%, 70%, 80%, and 90%. The results show that the single MLP model achieves an accuracy of 88.4%, with a false alarm rate of 22.8% and a miss classification rate of 29.3%. Meanwhile, the MLP ensemble bagging approach demonstrates improved performance, achieving the highest accuracy of 89.0%, a false alarm rate of 22.2%, and the lowest miss classification rate of 26.3% at the 50% probability threshold (Pa50). The lowest false alarm rate of 20.0% is observed at the 70% probability threshold (Pa70). These findings indicate that the MLP ensemble bagging approach enhances the stability and predictive reliability compared to the single MLP model. This approach effectively improves the reliability of monthly rainfall anomaly prediction and can serve as a supporting tool for landslide early warning systems in the Pacet region.

مستخلص البحث

سيسويو، دوي دفيتا أوكتافيا. 2025. التنبؤ بمعدل هطول الأمطار الشهري بناءً على مؤشرات المناخ في باجيت باستخدام نموذج تجميع الشبكات العصبونية متعددة الطبقات (MLP). رسالة الماجستير. قسم المعلومات، كلية العلوم والتكنولوجيا بجامعة مولانا مالك إبراهيم الإسلامية الحكومية مالانج. المشرف الأول: د. إروان بودي س.، الماجستير؛ المشرف الثاني: أ. د. محمد فيصل، الماجستير.

الكلمات الرئيسية: هطول أمطار، شذوذ، انهيارات أرضية، MLP، تجميع بحقائب.

منطقة باجيت، مقاطعة موجوكيرتو، هي منطقة مرتفعة ذات منحدرات شديدة الانحدار وتربة منفوشة، ما يجعلها عرضة بشكل كبير لانزلاقات الأراضي. لذلك، يُعد التنبؤ بالأمطار الشهرية خطوة مهمة في جهود التخفيف من الكوارث. تهدف هذه الرسالة إلى التنبؤ بالأمطار الشهرية في باجيت باستخدام مدخل الشبكات العصبية الاصطناعية (ANN) المستند إلى متعدد الطبقات (MLP) مع مخطط التجميع بالحقائب ومتغيرات الإدخال هي مؤشر المناخ العالمي، ومتغير الهدف هو بيانات هطول الأمطار CHIRPS التي أثبتت دقتها في منطقة جاوى الشرقية. النموذج الأساسي المستخدم في التجميع هو أفضل هيكل MLP مع تكوين مزدوج للطبقة المخفية (1-1) الذي تم الحصول عليه مسبقاً من تقييم سيناريوهات هيكلية متعددة. تم تنفيذ مخطط التجميع بالحقائب باستخدام تقنية إعادة أخذ عينات بالتمهيد 100 مرة (إعادة أخذ العينات مع الاستبدال)، حيث يتم حساب متوسط كل نتائج تنبؤ النموذج للحصول على المخرجات النهائية. بعد ذلك، تم التحقق من صحة نتائج التنبؤ باستخدام جدول التوافق من وكالة الأرصاد الجوية والمناخ والجيوفيزياء BMKG (الدقة، الإنذار الكاذب، والتصنيف الخاطئ) مقابل عتبة هطول الأمطار القصوى $500 <$ ملم شهرياً بناءً على احتمالية 50%، 60%، 70%، 80%، و90%. أظهرت نتائج البحث أن نموذج MLP الفردي يصل إلى دقة 88.4%، مع إنذار كاذب بنسبة 22.8% وتصنيف خاطئ بنسبة 29.3%. بعد ذلك، أظهر نموذج MLP التجميع بالحقائب تحسناً في الأداء، حيث وصل إلى أعلى دقة 89.0%، وإنذار كاذب 22.2% وتصنيف خاطئ منخفض بنسبة 26.3% عند عتبة Pa50، بينما كانت أقل نسبة إنذار كاذب 20.0% عند عتبة Pa70. أشارت هذه النتائج إلى أن MLP التجميع بالحقائب يعزز الاستقرار والموثوقية التنبؤية مقارنةً بالنموذج MLP الفردي. يمكن لهذا المدخل تعزيز موثوقية التنبؤ بالشذوذ في هطول الأمطار الشهري، مما يجعله أداة مساعدة في نظام الإنذار المبكر بالانهيارات الأرضية في منطقة باجيت.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Kecamatan Pacet yang terletak di Kabupaten Mojokerto, Jawa Timur, merupakan wilayah dengan karakteristik geografis berupa kawasan pegunungan dengan topografi yang curam serta jenis tanah yang tergolong gembur. Kondisi tersebut menjadikan Pacet sebagai salah satu daerah yang memiliki tingkat kerentanan tinggi terhadap bencana tanah longsor, terutama pada saat musim hujan. Kombinasi antara lereng yang terjal dan struktur tanah yang mampu menyerap air dalam jumlah besar, namun mudah kehilangan stabilitas saat jenuh, menjadikan curah hujan berintensitas tinggi yang terjadi secara terus-menerus sebagai faktor dominan penyebab tanah longsor. Oleh karena itu, akumulasi curah hujan dalam periode waktu tertentu, khususnya skala bulanan, menjadi indikator penting dalam mengidentifikasi risiko bencana tanah longsor secara lebih komprehensif dan prediktif.

Tingginya potensi bencana longsor di Pacet yang dipicu oleh akumulasi curah hujan ekstrem bukan hanya menjadi isu lingkungan, tetapi juga persoalan kemanusiaan yang menuntut perhatian lintas disiplin. Dalam perspektif Islam, fenomena hujan dan dinamika atmosfer adalah bagian dari *sunnatullah*, hukum alam yang diciptakan Allah dan berjalan secara konsisten sehingga memungkinkan manusia untuk menelitinya secara ilmiah. Al-Qur'an menggambarkan proses meteorologi, pembentukan awan, hingga keluarnya hujan sebagai tanda kebesaran Allah yang dapat dipelajari dan

dimanfaatkan untuk kemaslahatan manusia. Allah *subhānahu wa ta'ala* berfirman dalam QS Ar-Rum ayat 48:

اللَّهُ الَّذِي يُرْسِلُ الرِّيحَ فَتُثِيرُ سَحَابًا فَيَبْسُطُهُ فِي السَّمَاءِ كَيْفَ يَشَاءُ وَيَجْعَلُهُ كِسْفًا فَتَرَى الْوَدْقَ يَخْرُجُ مِنْ خِلَالِهِ فَإِذَا أَصَابَ بِهِ مَنْ يَشَاءُ مِنْ عِبَادِهِ إِذَا هُمْ يَسْتَبْشِرُونَ ﴿٤٨﴾

“Allah-lah yang mengirimkan angin, lalu angin itu menggerakkan awan; kemudian Allah membentangkannya di langit menurut kehendak-Nya dan menjadikannya bergumpal-gumpal, lalu engkau melihat hujan keluar dari celah-celahnya. Maka apabila hujan itu diturunkan kepada siapa yang Dia kehendaki dari hamba-hamba-Nya, seketika itu mereka bergembira.”

Ayat ini menggambarkan hubungan sebab-akibat atmosfer yang dapat diteliti secara ilmiah, termasuk pemodelan curah hujan. Selain itu, dinamika awan berlapis, hujan deras, hingga intensitas hujan yang berat juga dijelaskan dalam Al-Qur'an melalui QS. An-Nur ayat 43, yang sangat relevan dengan fenomena curah hujan ekstrem di Pacet sebagai berikut:

أَلَمْ تَرَ أَنَّ اللَّهَ يُرْجِي سَحَابًا ثُمَّ يُؤَلِّفُ بَيْنَهُ ثُمَّ يَجْعَلُهُ رُكَامًا فَتَرَى الْوَدْقَ يَخْرُجُ مِنْ خِلَالِهِ وَيُنْزِلُ مِنَ السَّمَاءِ مِنْ جِبَالٍ فِيهَا مِنْ بَرَدٍ فَيُصِيبُ بِهِ مَنْ يَشَاءُ وَيَصْرِفُهُ عَنْ مَنْ يَشَاءُ يَكَادُ سَنَا بَرْقِهِ يَذْهَبُ بِالْأَبْصَارِ ﴿٤٣﴾

“Tidakkah engkau melihat bahwa Allah mengarak awan, kemudian mengumpulkannya, lalu menjadikannya bertumpuk-tumpuk; maka engkau melihat hujan keluar dari celah-celahnya. Dan Dia menurunkan dari langit, dari gunung-gunung (awan), butiran es; lalu Dia menimpakannya kepada siapa yang Dia kehendaki dan memalingkannya dari siapa yang Dia kehendaki. Kilauan kilatnya hampir-hampir menghilangkan penglihatan.”

Ayat ini selaras dengan konsep ilmiah tentang hujan lebat, pembentukan awan konvektif, dan intensitas curah hujan ekstrem yang menjadi penyebab longsor. Selain

menggambarkan proses meteorologi, Al-Qur'an juga menegaskan bahwa fenomena alam adalah *tanda-tanda* (ayat-ayat kauniyah) yang dapat diteliti. Hal ini ditegaskan dalam QS. Surat Fussilat ayat 53:

سَنُرِيهِمْ آيَاتِنَا فِي الْأَفَاقِ وَفِي أَنْفُسِهِمْ حَتَّىٰ يَتَبَيَّنَ لَهُمْ أَنَّهُ الْحَقُّ أَوَلَمْ يَكْفِ
بِرَبِّكَ أَنَّهُ عَلَىٰ كُلِّ شَيْءٍ شَهِيدٌ ﴿٥٣﴾

“Akan Kami perlihatkan kepada mereka tanda-tanda (kekuasaan) Kami di segenap penjuru dan pada diri mereka sendiri, sehingga jelaslah bagi mereka bahwa Al-Qur'an itu benar. Dan apakah Tuhanmu tidak cukup bahwa Dia menjadi saksi atas segala sesuatu?”

Ayat ini menjadi dasar teologis bahwa penelitian terhadap gejala alam, termasuk curah hujan, bukan hanya dibolehkan tetapi dianjurkan sebagai bagian dari memahami tanda-tanda kebesaran Allah untuk menjaga kemaslahatan manusia.

Dalam pengembangan sistem prediksi curah hujan bulanan, pendekatan berbasis teknologi informasi, khususnya bidang *Data Science* dan *Machine Learning*, menjadi sangat relevan untuk menangani kompleksitas data iklim yang bersifat non-linear dan dinamis. Sittihock (2024) menunjukkan bahwa pemodelan prediksi curah hujan berbasis indeks iklim global secara signifikan meningkatkan akurasi dibandingkan pendekatan lokal. You et al. (2024) membuktikan kemampuan arsitektur ANN (*Artificial Neural Network*) kompleks dalam memodelkan intensitas hujan di wilayah tropis. Helali et al. (2022) mengverifikasi performa berbagai model *Machine Learning* untuk prediksi musiman di Iran dan menjelaskan bahwa ANN memberikan hasil terbaik.

Namun, beberapa penelitian tersebut masih memiliki beberapa keterbatasan yang menjadi latar belakang utama dari penelitian ini. Bouach (2024) dalam penelitiannya menguji satu arsitektur ANN tanpa eksplorasi kombinasi jumlah *neuron* atau variasi *hidden layer*, sehingga potensi optimalisasi model belum tercapai. Helali et al. (2022) menggunakan pendekatan musiman yang kurang sesuai untuk analisis longsor, karena tidak menggambarkan akumulasi kelembapan tanah secara bulanan yang lebih relevan. Mohammadpouri (2023) menerapkan GRNN (*Generalized Regression Neural Network*) untuk meningkatkan akurasi produk presipitasi global berbasis zona iklim untuk prediksi spasial global, namun belum menyesuaikan model terhadap karakteristik lokal wilayah rawan bencana. Sementara itu, Sittihock (2024) melakukan satu kali pelatihan ANN tanpa proses perataan (*averaging*), yang dapat menghasilkan prediksi tidak stabil akibat inisialisasi bobot awal yang acak. Selain itu, sebagian besar studi sebelumnya hanya mengverifikasi model berdasarkan statistik kesalahan seperti MSE (*Mean Squared Error*) atau RMSE (*Root Mean Squared Error*).

Berdasarkan pertimbangan tersebut, penelitian ini mengusulkan pengembangan sistem prediksi curah hujan bulanan berbasis MLP (*Multi-Layer Perceptron*) dengan pendekatan *Ensemble Bagging*. Dengan cara ini, *Ensemble Bagging* tidak hanya meningkatkan kestabilan hasil prediksi dibandingkan model tunggal, tetapi juga memberikan informasi probabilistik terhadap potensi kejadian ekstrem. Hal ini penting dalam konteks sistem peringatan dini, di mana probabilitas di atas ambang tertentu dapat dijadikan indikator risiko tinggi bagi bencana longsor.

Tanpa adanya prediksi curah hujan bulanan berbasis indeks iklim global yang stabil dan terkalibrasi, wilayah Pacet akan tetap bergantung pada prediksi umum skala

nasional yang belum tentu mencerminkan kondisi iklim lokal. Ketidakcocokan ini dapat menghambat kecepatan peringatan dini, sehingga masyarakat tetap rentan terhadap kejadian longsor berulang yang berdampak pada keselamatan jiwa, infrastruktur, pertanian, dan ekonomi lokal. Dengan demikian, penelitian ini bukan hanya memberikan kontribusi ilmiah dalam pengembangan model prediksi curah hujan berbasis *Artificial Intelligence*, tetapi juga merupakan bentuk implementasi nilai keislaman dalam menjaga kelestarian lingkungan dan melindungi kehidupan manusia dari bencana.

1.2 Pernyataan Masalah

Penelitian ini akan membahas beberapa permasalahan utama berikut:

1. Bagaimana performa prediksi curah hujan di Pacet yang dihasilkan skenario arsitektur MLP?
2. Sejauh mana performa usulan *Ensemble Bagging* MLP dapat memberikan keunggulan dibandingkan dengan model tunggal MLP?

1.3 Tujuan Penelitian

Tujuan dari penelitian ini adalah:

1. Mengukur performa skenario model MLP yang digunakan dalam prediksi curah hujan di Pacet.
2. Membandingkan performa model tunggal dengan usulan *Ensemble Bagging* MLP, melalui verifikasi kontingensi BMKG.

1.4 Ruang Lingkup Penelitian

Penelitian ini memiliki beberapa ruang lingkup yang dibatasi agar tujuan dapat dicapai secara terfokus, yaitu:

1. Ruang Lingkup Wilayah:

Penelitian difokuskan pada wilayah Kecamatan Pacet, Kabupaten Mojokerto, Jawa Timur, sebagai lokasi studi kasus prediksi curah hujan bulanan.

2. Variabel *Input* dan *Output*:

Variabel *input* yang digunakan adalah indeks iklim global yang meliputi NINO, ONI, IOD, WNPPI (*Western North Pacific Monsoon Index*), dan AUSMI (*Australian Monsoon Index*). Sedangkan variabel *output* yang diprediksi adalah curah hujan bulanan dalam satuan mm (milimeter) dan data CHIRPS sebagai nilai acuannya.

3. Analisa Data:

Analisis hasil prediksi menggunakan parameter tabel kontingensi BMKG dengan variabel akurasi, *false alarm*, dan *miss classification*.

4. Periode Data:

Data yang digunakan merupakan data indeks global dan curah hujan bulanan dari periode Januari 1991 hingga Maret 2025.

1.5 Manfaat Penelitian

Penelitian ini diharapkan dapat memberikan kontribusi nyata dalam berbagai sektor, antara lain:

1. Bagi BMKG dan Pemerintah Daerah

Mendukung peningkatan akurasi prediksi curah hujan bulanan, memperkuat sistem peringatan dini bencana longsor berbasis teknologi informasi, serta membantu perumusan kebijakan mitigasi bencana secara lebih terukur.

2. Bagi Pengelola Sumber Daya Air

Menyediakan informasi prediksi yang akurat untuk mendukung perencanaan musim tanam, pengaturan irigasi, dan pengelolaan air secara adaptif terhadap perubahan iklim.

3. Bagi Masyarakat dan Lembaga Swadaya Masyarakat (LSM)

Menyediakan informasi iklim yang mudah dipahami dan dapat digunakan dalam aktivitas sehari-hari, termasuk dalam upaya peningkatan kesiapsiagaan terhadap potensi bencana hidrometeorologi.

BAB II

STUDI PUSTAKA

2.1 Model Prediksi Curah Hujan

Model prediksi curah hujan merupakan pendekatan ilmiah yang digunakan untuk memprediksi kondisi atmosfer pada masa mendatang, baik dalam skala harian, mingguan, maupun musiman. Dalam konteks peringatan dini bencana hidrometeorologi seperti tanah longsor, model prediksi curah hujan bulanan berperan penting sebagai alat mitigasi berbasis informasi iklim. Safril et al. (2020) menekankan bahwa curah hujan bulanan merupakan variabel yang sangat menentukan terhadap potensi longsor, terutama di daerah pegunungan yang memiliki karakteristik tanah kemiringan lereng yang curam. Studi ini memanfaatkan pendekatan *Statistical Downscaling* untuk menurunkan skala data iklim global menjadi prediksi curah hujan lokal dengan resolusi yang lebih tinggi. Pendekatan ini kemudian dikombinasikan dengan metode *ensemble* untuk meningkatkan kemampuan prediksi curah hujan bulanan sebagai dasar peringatan dini bencana longsor.

Agustina et al. (2020) menjelaskan bahwa setiap wilayah memiliki ambang batas curah hujan bulanan yang berbeda dalam memicu kejadian longsor. Dalam studi tersebut, dilakukan analisis terhadap data kejadian longsor dan curah hujan aktual untuk menentukan nilai ambang curah hujan spesifik yang dapat digunakan dalam sistem peringatan dini. Pendekatan ini sangat relevan karena menunjukkan bahwa tidak semua intensitas curah hujan bulanan berpotensi menyebabkan longsor, dan ambang

batas lokal menjadi acuan penting dalam pembuatan sistem deteksi risiko. Wilayah yang dikaji adalah Kecamatan Pejawaran, Kabupaten Banjarnegara, Provinsi Jawa Tengah dengan hasil penelitiannya adalah ambang curah hujan signifikan untuk wilayah tersebut adalah >300 mm/bulan menyebabkan longsor. Penelitian ini juga menerapkan pendekatan *ensemble* berbasis *Statistical Downscaling* untuk menggabungkan beberapa model prediksi curah hujan. Hasilnya menunjukkan bahwa metode *ensemble* mampu meningkatkan akurasi dan stabilitas prediksi dibandingkan model tunggal, sehingga lebih efektif digunakan dalam sistem peringatan dini longsor.

Dalam konteks wilayah Pacet, berdasarkan konfirmasi ilmiah kepada BMKG dan merujuk pada klasifikasi curah hujan ekstrem dalam sistem peringatan dini bencana yang ditetapkan pada peraturan BMKG no 09 tahun 2019 tentang Penyediaan dan Penyebaran Peringatan Dini Iklim Ekstrem, bahwa ambang batas curah hujan bulanan yang digunakan untuk indikasi peringatan dini longsor adalah >500 mm/bulan. Artinya, ketika model prediksi mengestimasi nilai curah hujan melampaui ambang yang telah ditentukan, maka potensi longsor dianggap signifikan dan perlu dilakukan tindakan mitigasi oleh pihak berwenang. Prediksi curah hujan merupakan salah satu aspek penting dalam perencanaan sumber daya air, pertanian, dan mitigasi bencana. Berbagai pendekatan telah dikembangkan, baik statistik maupun berbasis kecerdasan buatan. Salah satu tantangan utama dalam memprediksi curah hujan adalah karakteristiknya yang bersifat non-linear dan dipengaruhi oleh banyak faktor, termasuk indeks iklim global.

Sittichok (2024) membandingkan kinerja metode statistik Regresi dengan model ANN tipe MLP (*Multi-Layer Perceptron*) dalam menghasilkan prediksi curah

hujan musiman jangka panjang. Data historis musiman digunakan sebagai variabel *input*, kemudian model dikembangkan untuk mengidentifikasi pola musiman jangka panjang yang bersifat non-linear. Penelitian ini juga menerapkan pendekatan *ensemble* pada metode Regresi, di mana teknik *bagging* digunakan untuk meningkatkan kestabilan prediksi Regresi dan mengurangi varians model. Hasil menunjukkan bahwa MLP memiliki kinerja paling unggul dengan nilai $R^2 = 0.91$, sedangkan model Regresi *ensemble* memiliki nilai R^2 yang lebih rendah. Temuan ini menegaskan bahwa model MLP lebih mampu menangkap pola variabilitas musiman yang kompleks dibandingkan model regresi konvensional, bahkan ketika regresi tersebut telah diperkuat dengan *bagging*.

Aizansi et al. (2024) membahas upaya untuk memprediksi curah hujan bulanan di Republik Benin menggunakan metode ANN MLP karena adanya kebutuhana untuk menyediakan prediksi curah hujan yang lebih akurat di kawasan Benin, yang sangat bergantung pada pertanian berbasis hujan. Dalam penelitian ini, data historis curah hujan diamati dan digunakan sebagai input ke dalam model ANN, untuk menangkap pola musiman dan tren jangka panjang dari data tersebut. ANN dipilih karena kemampuannya dalam mengenali hubungan non-linear yang kompleks antara variabel iklim. Proses model pelatihan dilakukan dengan memperhatikan aspek normalisasi data untuk meningkatkan kecepatan dan akurasi pelatihan. Hasil penelitian menunjukkan bahwa model ANN mampu memberikan prediksi curah hujan bulanan dengan performa yang cukup baik. Verifikasi kinerja model menggunakan parameter $RMSE=0,1$ hingga $0,3$ dan $MAE=0,08$ hingga $0,25$ menunjukkan nilai kesalahan yang rendah, membuktikan efektivitas penggunaan ANN dalam konteks prediksi curah

hujan di wilayah dengan variabilitas iklim tinggi seperti Benin. Penelitian ini merekomendasikan penggunaan ANN sebagai alternatif untuk prediksi iklim di negara-negara berkembang.

Putra et al. (2024) melakukan estimasi curah hujan resolusi tinggi di Indonesia dengan mengintegrasikan data multisensor menggunakan metode MLP yang dikombinasikan dengan pendekatan *Ensemble Bagging*. Tujuan penelitian ini adalah meningkatkan representasi spasial curah hujan yang umumnya terbatas oleh kerapatan stasiun pengamatan. Model MLP dilatih pada dataset gabungan yang memuat informasi satelit, radar, dan observasi lapangan, kemudian dikombinasikan melalui skema *bagging* untuk mengurangi varians dan meningkatkan stabilitas prediksi. Hasil menunjukkan bahwa akurasi estimasi curah hujan meningkat secara konsisten, dengan nilai akurasi mencapai 92%, lebih tinggi dibandingkan model non *ensemble*. Hal ini menegaskan bahwa *ensemble* MLP efektif dalam memperhalus estimasi curah hujan spasial.

Surana et al. (2024) mengusulkan model *Fine Tuned* MLP berbasis *Ensemble Bagging* untuk memprediksi kejadian hujan harian berdasarkan data satelit. Fokus penelitian ini adalah pada kemampuan model dalam mendeteksi transisi kondisi hujan–tidak hujan dengan tingkat kesalahan minimal. Model MLP dilatih secara berulang menggunakan beberapa subset data, dan hasil setiap model digabungkan untuk menghasilkan prediksi akhir yang lebih stabil. Hasil pengujian menunjukkan akurasi mencapai 91.3%, menunjukkan bahwa pendekatan *bagging* berhasil mengurangi risiko *overfitting* yang sering muncul pada model ANN tunggal.

Hafyani et al. (2024) mengembangkan pendekatan *Multi View Stacking* untuk meningkatkan akurasi prediksi curah hujan bulanan. Dalam penelitian ini, sejumlah model dasar termasuk MLP diterapkan pada pandangan data (*views*) yang berbeda, seperti tren musiman, anomali iklim, dan komponen variabilitas jangka pendek. *Output* dari berbagai model dasar tersebut kemudian disatukan menggunakan teknik *Ensemble Stacking*, di mana meta-model mempelajari pola gabungan terbaik dari prediksi masing-masing model awal. Hasil menunjukkan bahwa pendekatan *stacking* mampu menurunkan nilai RMSE menjadi 17.5 mm, lebih baik daripada model tunggal tanpa *stacking*.

Bouach (2024) membahas pengembangan model prediksi curah hujan bulanan berdasarkan kombinasi teknik *Wavelet Transformation* dan *Ensemble Stacking*. Penelitian ini bertujuan untuk meningkatkan akurasi prediksi curah hujan dengan menggabungkan keunggulan berbagai model prediktif menggunakan pendekatan *stacking*, di mana *output* dari beberapa model dasar dikombinasikan untuk membentuk prediksi akhir. Metode yang digunakan adalah Dekomposisi *Wavelet* untuk memecah data curah hujan menjadi beberapa komponen frekuensi yang berbeda, sehingga pola musiman, tren jangka panjang, dan kebisingan dapat ditangani secara lebih efektif. Selanjutnya, beberapa model *Machine Learning* termasuk ANN MLP dan SVM, diterapkan pada masing-masing komponen hasil *wavelet* sebelum hasilnya digabungkan menggunakan teknik penumpukan. Hasil penelitian menunjukkan bahwa pendekatan *Wavelet Stacking* secara signifikan mampu mengurangi nilai *error* dibandingkan model individu. Verifikasi kinerja menunjukkan nilai MSE (*Mean*

Squared Error) sebesar 2.27, yang merupakan peningkatan dibandingkan dengan model konvensional tanpa *wavelet* atau tanpa *ensemble*. Penelitian ini menegaskan bahwa integrasi Dekomposisi *Wavelet* dengan *ensemble* memberikan keunggulan dalam menangkap dinamika data curah hujan yang kompleks, sehingga dapat digunakan untuk penerapan prediksi cuaca jangka menengah dan panjang.

Aprianto (2024) membahas perbandingan performa beberapa model *Deep Learning* dalam prediksi curah hujan musiman secara *realtime*. Penelitian ini fokus pada verifikasi efektivitas model LSTM (*Long ShortTerm Memory*) dibandingkan dengan model *Deep Learning* lainnya dalam memprediksi curah hujan jangka pendek berdasarkan data observasi harian. Model yang diuji melibatkan struktur RNN dan GRU (*Gated Recurrent Unit*). Penelitian ini fokus pada verifikasi efektivitas model LSTM (*Long Short-Term Memory*) dibandingkan dengan model *Deep Learning* lainnya dalam memprediksi curah hujan jangka pendek berdasarkan data observasi harian. Model yang diuji melibatkan struktur RNN, GRU, dan LSTM, dengan LSTM dipilih sebagai model utama karena kemampuannya dalam menangani data deret waktu yang memiliki ketergantungan jangka panjang. Metode yang digunakan adalah *Deep Learning* berbasis LSTM, di mana data curah hujan harian diproses melalui beberapa tahap, termasuk normalisasi dan pembentukan *Sliding Window* untuk *input* ke model. Penelitian ini menggunakan dataset historis dari beberapa stasiun hujan untuk melatih dan menguji model prediksi. Hasil penelitian menunjukkan bahwa model LSTM menghasilkan performa yang lebih baik dibandingkan model lainnya, dengan nilai RMSE sebesar 14.56 mm dan MAE sebesar 10.62 mm pada data pengujian. Hasil ini

menunjukkan bahwa model LSTM dapat secara efektif mempelajari pola curah hujan harian dan memberikan prediksi yang akurat secara *realtime*.

Baljon dan Sharma (2023) mengembangkan model prediksi curah hujan harian di Arab Saudi menggunakan algoritma *Machine Learning* seperti *Random Forest*, *SVM*, dan *Gradient Boosting*. *Input* yang digunakan adalah data meteorologis historis seperti suhu, kelembaban, dan tekanan. Penelitian ini menunjukkan bahwa pendekatan *Machine Learning* lebih mampu memprediksi curah hujan secara akurat dibandingkan model statistik sederhana, terutama pada wilayah gurun dengan pola cuaca ekstrem.

Mohammadpouri (2023) membahas penggunaan ANN dalam memprediksi curah harian berdasarkan data meteorologi historis. Fokus penelitian ini untuk mengetahui seberapa efektif ANN dalam menangkap pola kompleks hubungan antara variabel cuaca dan curah hujan di masa depan. Model yang dikembangkan menggunakan beberapa variabel *input* seperti suhu, kelembaban, tekanan udara, dan data. Model yang dikembangkan menggunakan beberapa variabel *input* seperti suhu, kelembaban, tekanan udara, dan data curah hujan masa lalu, dengan proses awal berupa normalisasi data untuk meningkatkan kinerja model pelatihan. *Input* data diproses dalam bentuk *time series*, dan dilakukan pembagian data menjadi data pelatihan dan pengujian. Jaringan saraf yang digunakan mengadopsi model MLP (*Feedforward Multi-Layer Perceptron*) dengan algoritma pelatihan *Backpropagation*. Hasil penelitian menunjukkan bahwa ANN mampu memberikan prediksi curah hujan dengan performa yang cukup baik. Koefisien korelasi antara data prediksi dan observasi mencapai 0,85 menunjukkan keakuratan metode yang digunakan. Selain itu, nilai MSE mencapai 0,001 membuktikan keakuratan model dalam memprediksi curah hujan.

Attaf et al. (2024) memfokuskan penelitian pada peningkatan kinerja ANN dalam estimasi curah hujan berbasis citra penginderaan jauh. Metode yang digunakan adalah MLP dengan pendekatan *Ensemble Bagging* untuk memperbaiki sensitivitas model terhadap variasi spasial dan temporal pada citra hujan. Dengan melakukan pelatihan berulang menggunakan subset data raster yang berbeda, model mampu mempelajari pola curah hujan secara lebih general. Penelitian ini menunjukkan bahwa metode *ensemble* secara signifikan meningkatkan ketepatan estimasi dengan tingkat akurasi mencapai 96.8%, menjadikannya sangat efektif untuk aplikasi monitoring berbasis satelit.

Khastagir (2022) meneliti penerapan ANN untuk prediksi curah hujan bulanan di wilayah tropis. Data meteorologi seperti suhu, kelembaban, dan tekanan udara digunakan sebagai input variabel. Normalisasi data dilakukan sebelum pelatihan. Model ANN yang digunakan adalah MLP *Backpropagation*. Penelitian ini menunjukkan bahwa ANN mampu memodelkan curah hujan dengan *error* prediksi yang kecil.

Martínez et al. (2022) membandingkan beberapa teknik *Machine Learning* untuk prediksi curah hujan harian di Australia, termasuk MLP dan *Random Forest* (RF). Hasil penelitian menunjukkan bahwa *ensemble* RF mendapatkan nilai $R^2 = 0.88$, sementara MLP memperoleh $R^2 = 0.82$. Meskipun performa RF sedikit lebih tinggi, penelitian ini menyoroti bahwa MLP dapat mencapai akurasi yang kompetitif, terutama ketika diberi parameterisasi yang optimal. Studi ini memberikan gambaran bahwa

pemilihan algoritma perlu mempertimbangkan karakteristik variabilitas curah hujan lokal dan struktur dataset.

Putra dan Rani (2020) memanfaatkan model MLP untuk memprediksi curah hujan harian di Stasiun Meteorologi Kemayoran, Jakarta. Penelitian ini menggunakan *input* data historis curah hujan dan parameter cuaca lainnya seperti suhu, kelembapan, dan tekanan. Data diperoleh dari BMKG dengan rentang waktu harian selama 15 tahun. Hasil validasi menunjukkan bahwa model MLP mampu mengantisipasi anomali curah hujan harian, dengan MAE sebesar 2,18 mm dan R^2 sebesar 0,88. Model ini direkomendasikan untuk sistem peringatan dini cuaca ekstrem di kawasan perkotaan.

Helali et al. (2022) melakukan verifikasi komparatif terhadap beberapa model machine learning, termasuk MLP, SVM, dan *Random Forest*, untuk simulasi presipitasi musiman di Iran. Meskipun penelitian ini tidak menggunakan bagging secara khusus, pendekatan yang digunakan termasuk multi-model *ensemble*, dimana hasil prediksi beberapa model dibandingkan dan dianalisis untuk memperoleh model yang paling optimal. Hasil menunjukkan bahwa model MLP mencapai nilai $R^2 = 0.93$, namun nilai R^2 model SVM dan RF tercatat sedikit lebih tinggi (>0.93). Studi ini menunjukkan bahwa meskipun ANN mampu memberikan performa prediksi yang baik, beberapa pendekatan *ensemble* berbasis pohon keputusan dan kernel juga dapat memberikan hasil yang kompetitif, terutama pada kasus variabilitas curah hujan musiman.

Papailiou et al. (2022) mengkaji kemampuan ANN dalam prediksi curah hujan bulanan di India menggunakan data historis selama 30 tahun. *Input* utama adalah data curah hujan masa lalu yang dinormalisasi. Model ANN yang digunakan adalah MLP

dengan optimasi jumlah *neuron* pada lapisan tersembunyi melalui percobaan *grid search*. Hasil penelitian menunjukkan ANN mampu menghasilkan prediksi dengan nilai korelasi 0,93 dan *error* yang relatif kecil (MSE rendah), hal ini menegaskan ANN sebagai metode yang efektif untuk prediksi jangka menengah.

Najib dan Mustika (2022) mengkaji penggunaan ANN untuk prediksi curah hujan harian di Kerala, India. Data yang digunakan termasuk suhu permukaan laut dan parameter atmosfer lainnya. Setelah data dinormalisasi, data digunakan untuk melatih model MLP. Penelitian menunjukkan ANN memberikan hasil prediksi yang cukup akurat, dengan koefisien korelasi mencapai 0,915 dan *error* prediksi yang relatif rendah yaitu 8,65mm.

Singh (2021) mengembangkan sistem prediksi curah hujan skala global dalam kerangka *Numerical Weather Prediction* (NWP) berbasis *Deep Learning*, khususnya arsitektur *CNN Multi-Layer*. Dalam penelitian ini, pendekatan *ensemble* diterapkan, beberapa model dilatih dengan kondisi awal yang berbeda, kemudian outputnya digabungkan (*Ensemble Averaging*) untuk menghasilkan prediksi akhir yang lebih stabil dan mengurangi sensitivitas terhadap noise dinamika atmosfer. Pendekatan ini terbukti meningkatkan kinerja model NWP secara signifikan, dengan peningkatan nilai korelasi prediksi sebesar 16% dibandingkan *baseline* model tunggal. Hal ini menunjukkan bahwa integrasi *deep learning* dengan strategi *ensemble* memiliki potensi besar dalam memperbaiki performa prediksi hujan berskala luas, terutama pada sistem prediksi numerik.

Chand dan Nand (2020) meneliti kemampuan ANN untuk prediksi curah hujan bulanan di wilayah Rajasthan. Data historis digunakan sebagai input ke dalam ANN,

dengan beberapa *layer* ANN untuk prediksi curah hujan bulanan di wilayah Rajasthan. Data historis digunakan sebagai *input* ke dalam ANN, dengan beberapa *preprocessing* seperti normalisasi *min-max*. Struktur ANN yang digunakan adalah MLP dengan *single hidden layer* dan pelatihan menggunakan *backpropagation*. Penelitian ini membuktikan bahwa ANN berhasil memprediksi curah hujan bulanan dengan koefisien korelasi sekitar 0,87 yang menunjukkan kemampuan model dalam situasi geografis yang kering dan curah hujan yang tidak merata.

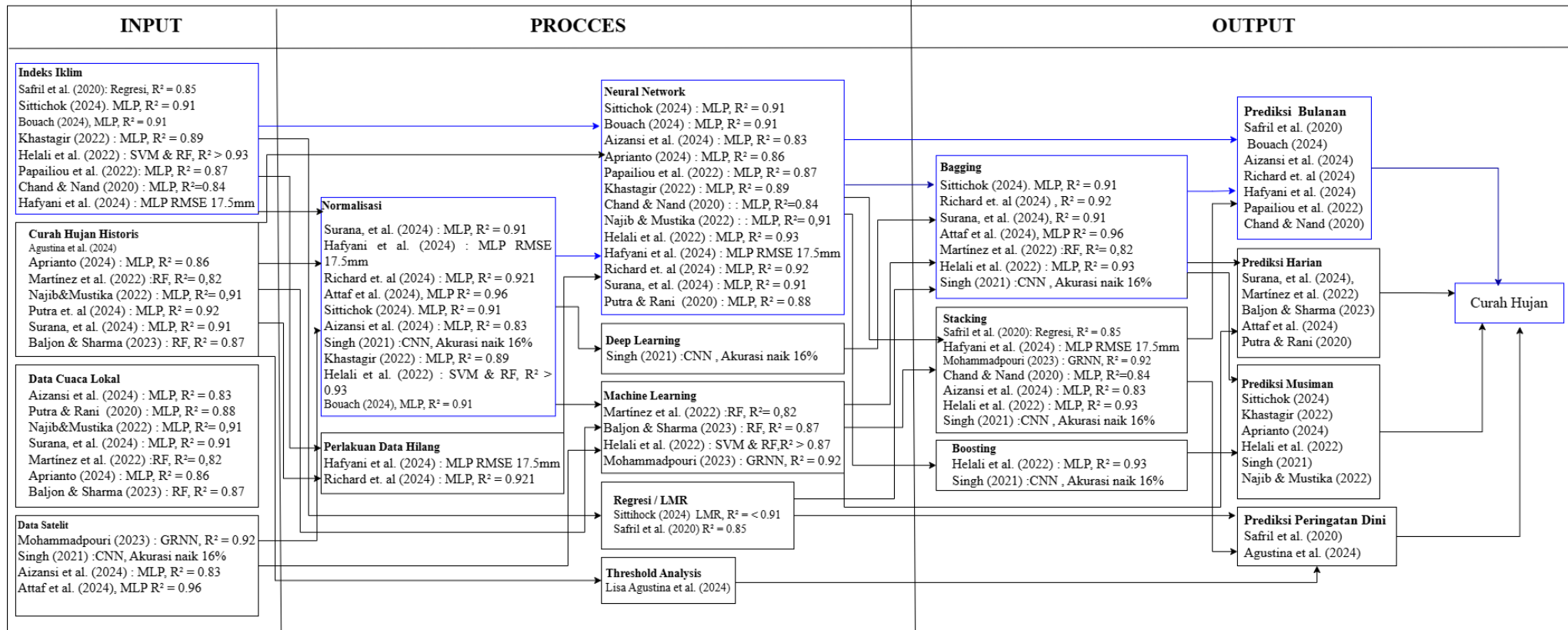
Berdasarkan hasil telaah berbagai penelitian sebelumnya, dapat disimpulkan bahwa pengembangan model prediksi curah hujan bulanan berbasis variabel iklim global dan parameter lokal memiliki peranan penting dalam sistem peringatan dini bencana, khususnya tanah longsor. Integrasi indeks iklim seperti ENSO, IOD, dan monsun, terbukti mampu meningkatkan kemampuan model dalam menangkap pola variabilitas curah hujan yang bersifat non-linear dan dipengaruhi dinamika atmosfer skala luas. Selain itu, pendekatan ANN tipe MLP (*Multi-Layer Perceptron*) menjadi model yang paling luas digunakan pada penelitian-penelitian terkini. Kinerja MLP terbukti konsisten lebih unggul dibandingkan metode statistik konvensional karena kemampuannya dalam mempelajari hubungan kompleks antara *input* dan *output*. Lebih jauh lagi, berbagai studi menunjukkan bahwa kinerja MLP meningkat secara signifikan ketika dikombinasikan dengan pendekatan *ensemble*, terutama metode *bagging*. Teknik *bagging* bekerja dengan menghasilkan beberapa model pelatihan kemudian menggabungkan hasil prediksinya untuk menurunkan varians model dan meningkatkan stabilitas prediksi, sehingga menghasilkan prediksi curah hujan yang lebih akurat dan lebih tahan terhadap fluktuasi data.

Dengan demikian, dapat ditegaskan bahwa MLP dengan pendekatan *ensemble bagging* merupakan pendekatan yang paling banyak digunakan dan paling konsisten menunjukkan hasil terbaik dalam prediksi curah hujan, baik pada skala harian, bulanan, maupun musiman. Keandalan model ini menjadi semakin bernilai ketika prediksi curah hujan tersebut dihubungkan dengan ambang batas curah hujan ekstrem yang telah divalidasi melalui analisis klimatologis dan data kejadian longsor aktual. Pendekatan ini memungkinkan hasil prediksi diterjemahkan secara langsung ke dalam klasifikasi tingkat risiko bencana, sehingga dapat digunakan secara operasional untuk mendukung sistem mitigasi dan pengambilan keputusan kebencanaan yang lebih akurat dan berbasis bukti.

2.2 Kerangka Teoritis

Dalam sub bagian ini disusun kerangka teori yang merujuk pada berbagai studi pustaka yang relevan sebagai acuan konseptual penelitian. Penyusunan kerangka teori dilakukan melalui tahapan pengelompokan parameter atau variabel yang digunakan sebagai *input* penelitian, dilanjutkan dengan uraian mendalam mengenai metode analisis yang diterapkan, serta deskripsi terperinci mengenai *output* atau hasil yang diharapkan.

Seluruh keterhubungan antar variabel tersebut divisualisasikan pada Gambar 2.1 sebagai berikut:



Gambar 2.1 Kerangka Teori

Berdasarkan Gambar 2.1 dapat diketahui bahwa terdapat beberapa opsi data masukan untuk membuat prediksi curah hujan, meliputi, data curah hujan historis, data cuaca observasi, indeks iklim. Seluruh data disediakan dalam format numerik sehingga dapat diproses secara matematis. Tahapan selanjutnya adalah menentukan metode terbaik yang akan digunakan dalam penelitian ini berdasarkan hasil studi pustaka. Beberapa penelitian yang dijadikan acuan untuk penelitian ini dituangkan pada Tabel 2.1 sebagai berikut.

Tabel 2.1 Daftar Penelitian Sebelumnya Yang Dipakai Acuan

No	Identitas Jurnal	Masalah Yang Diselesaikan	Metode Yang Digunakan	Hasil
1.	Safril et al. (2020). Prediksi Curah Hujan Bulanan untuk Peringatan Dini Longsor di Banjarnegara Bagian Selatan dengan <i>Statistical Downscaling</i> dan <i>Ensemble</i>	Prediksi curah hujan bulanan untuk peringatan dini longsor di Banjarnegara	Regresi	$R^2 = 0.85$
2.	Agustina et al. (2020). Analisis Ambang Batas Hujan untuk Pengembangan Sistem Peringatan Dini Tanah Longsor	Ambang batas hujan untuk peringatan longsor	<i>Threshold analysis</i>	>300 mm/bulan menyebabkan longsor
3.	Sittichok (2024). <i>Long-Term Seasonal Rainfall Forecasting Using Regression and ANN</i>	Prediksi musiman jangka panjang	MLP Regresi	MLP: $R^2 = 0.91$ Regresi: $R^2 = < 0.91$
4.	Aizansi et al. (2024). <i>Monthly rainfall prediction using artificial neural network</i>	Prediksi curah hujan bulanan di Benin	MLP	$R^2 = 0.83$ MAE = 9.3 mm
5.	Putra et al. (2024). <i>High-Resolution Rainfall Estimation Using Ensemble Learning & Multisensor Integration. Sensors</i>	Estimasi hujan resolusi-tinggi di Indonesia	MLP	$R^2 = 0.92$

6.	Surana et al. (2024). <i>Ensemble fine-tuned MLP for rainfall forecasting.</i>	Deteksi kejadian hujan harian	MLP	$R^2 = 0.91$
7.	Hafyani et al. (2024). <i>Improving monthly precipitation prediction accuracy using multi-view stacking. Frontiers in Water.</i>	Peningkatan akurasi prediksi curah hujan bulanan	MLP	RMSE = 17.5 mm
8.	Bouach (2024). <i>Artificial neural networks for monthly precipitation prediction in north-west Algeria</i>	Prediksi curah hujan bulanan di Aljazair Barat Laut	MLP	$R^2 = 0.91$; RMSE = 8.12 mm
9.	Aprianto (2024). Analisis Pola Musim Hujan dan Kemarau Berdasarkan Prediksi Curah Hujan Tahun 2024 Menggunakan <i>Artificial Neural Network</i> (ANN) di Kabupaten Sumbawa	Prediksi musim hujan dan kemarau di Sumbawa	MLP	$R^2 = 0.86$; RMSE = 7.65 mm
10.	Baljon & Sharma (2023). <i>Rainfall Prediction Rate in Saudi Arabia Using Improved Machine Learning Techniques</i>	Prediksi hujan harian Arab Saudi	RF SVM GBoost	RF: $R^2 = 0.87$; SVM&Gboos: $R^2 < 0.87$
11.	Mohammad Pouri (2023). <i>A Generalized Regression Neural Network Model for Accuracy Improvement of Global Precipitation Products</i>	Model GRNN untuk produk presipitasi global	GRNN	$R^2 = 0.92$
12.	Attaf et al (2024). <i>Enhancement of ANN performance for remote-sensing rainfall estimates using ensemble methods.</i>	Estimasi curah hujan harian dari citra	MLP	$R^2 = 0.96$
13.	Khastagir (2022). <i>Efficacy of LMR and ANN for Long Term Rainfall Forecasting in Western Australia.</i>	Prediksi jangka panjang curah hujan	LMR ANN	ANN: $R^2 = 0.89$ LMR: $R^2 = 0.72$
14.	Martínez et al. (2022). <i>Prediction of Rainfall in Australia Using Machine Learning.</i>	Prediksi hujan harian di berbagai kota Australia	MLP	RF: $R^2 = 0.88$; MLP: $R^2 = 0.82$

15.	Putra & Rani. (2020). Prediksi Curah Hujan Harian di Stasiun Meteorologi Kemayoran Menggunakan <i>Artificial Neural Network</i>	Prediksi hujan harian di Stasiun Meteorologi Kemayoran	MLP	$R^2 = 0,88$ MAE=2,18
16.	Helali et al. (2022). <i>Assessment of Machine Learning Model Performance for Seasonal Precipitation Simulation in Iran</i>	Simulasi presipitasi musiman di Iran	SVM RF MLP	MLP: $R^2 = 0.93$ SVM&RF: $R^2 > 0.93$
17.	Papailiou et al. (2022). <i>Artificial neural networks for monthly precipitation prediction in north-west Algeria: a case study in the Oranie-Chott-Chergui basin</i>	Prediksi curah hujan bulanan Aljazair	MLP	$R^2 = 0.87$ RMSE = 7.4
18.	Najib & Mustika (2022). <i>Weather Forecasting Using ANN for Rice Farming in Delanggu Village</i>	Prediksi cuaca untuk pertanian di Delanggu	MLP	$R^2 = 0.91$
19.	Singh (2021). <i>Deep Learning for Improved Global Precipitation in Numerical Weather Prediction Systems</i>	<i>Deep Learning</i> untuk NWP global	CNN	R^2 naik 16% dibanding <i>baseline</i>
20.	Chand & Nand (2020). <i>Rainfall Prediction Using Artificial Neural Network in the South Pacific Region</i>	Prediksi curah hujan di <i>South Pacific</i>	MLP	$R^2 = 0.84$

Berdasarkan tabel diatas menunjukkan MLP mampu memodelkan hubungan non-linear yang kompleks antara faktor-faktor iklim global maupun lokal dengan curah hujan, yang tidak dapat secara optimal ditangkap oleh metode Regresi Linear Sederhana. Sejumlah studi seperti yang dilakukan dalam beberapa penelitian jurnal menunjukkan bahwa MLP menghasilkan nilai akurasi prediksi yang lebih tinggi, serta nilai *error* prediksi seperti RMSE, MSE, dan MAE yang lebih rendah dibandingkan model lainnya, performa prediksi yang dihasilkan umumnya lebih rendah, dengan nilai

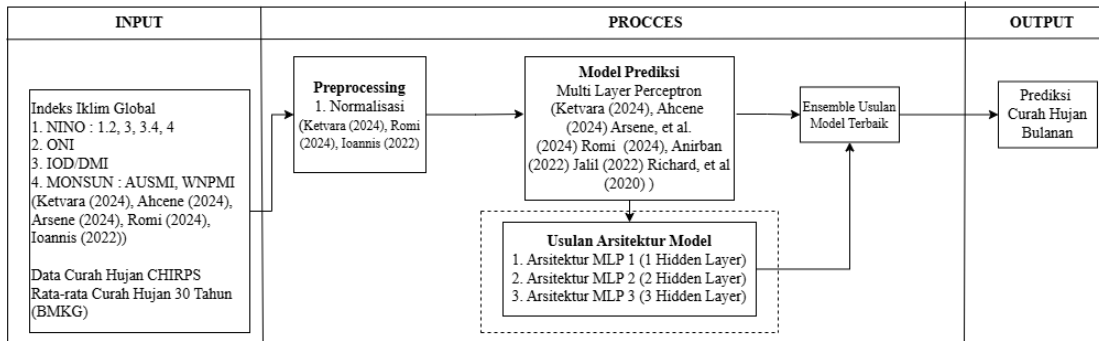
koefisien determinasi (R^2) yang juga lebih baik. Hal ini memberikan dasar bahwa penelitian dengan menggunakan MLP masih relevan, sekaligus membuka peluang untuk melakukan pengembangan lebih lanjut. Dengan demikian, penelitian ini diarahkan untuk memanfaatkan MLP dalam memprediksi curah hujan bulanan berbasis indeks iklim global di Pacet sebagai upaya mengoptimasi kualitas informasi prediksi musim di Pacet sesuai dengan kebutuhannya.

BAB III

MODEL KONSEPTUAL

3.1 Kerangka Konseptual

Kerangka konseptual penelitian ini disusun untuk menggambarkan alur sistematis dari tahapan yang dilakukan dalam membangun model prediksi curah hujan bulanan berbasis indeks iklim global sebagai panduan utama yang menunjukkan hubungan antar variabel *input*, proses pengolahan data, dan *output* yang dihasilkan seperti pada gambar 3.1. Pemilihan indeks iklim global didasarkan pada pengaruhnya yang signifikan terhadap variabilitas curah hujan di wilayah tropis, khususnya Pacet. Selanjutnya, model MLP (*Multi-Layer Perceptron*) digunakan sebagai pendekatan utama dalam pemodelan karena telah terbukti efektif pada berbagai penelitian sebelumnya dengan tingkat akurasi yang tinggi.



Gambar 3.1 Alur Kerangka Konseptual Penelitian

Gambar 3.1 menjelaskan kerangka konseptual penelitian prediksi curah hujan berbasis indeks iklim global menggunakan model MLP. Penelitian ini menggunakan data masukan berupa indeks iklim global, data curah hujan historis serta nilai rata-rata

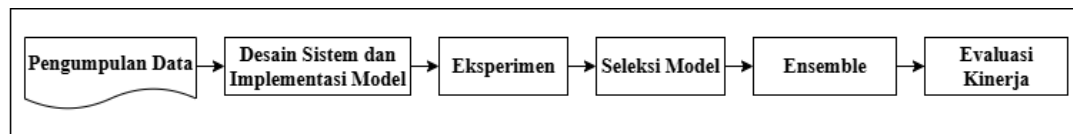
curah hujan di Pacet selama 30 tahun. Indeks iklim yang dipakai mencakup NINO1.2, NINO3, NINO3.4, dan NINO4), ONI (*Oceanic Nino Indicator*), IOD (*Indian Oceanic Dipole*), serta indeks monsun AUSMI (*Australian Monsoon Index*) dan WNPMP (*Western North Pacific Monsoon Index*) yang dalam berbagai penelitian sebelumnya terbukti berpengaruh terhadap variabilitas curah hujan di Indonesia. Selain itu, digunakan pula data curah hujan historis dari CHIRPS dan data rata-rata curah hujan 30 tahun dari BMKG yang berfungsi sebagai acuan pembandingan dalam proses validasi model. Sebelum dilakukan pemodelan, data *input* diproses melalui tahap normalisasi agar berada pada rentang nilai tertentu, sehingga seluruh variabel memiliki skala yang sebanding. Normalisasi bertujuan untuk meningkatkan kestabilan perhitungan dan mempercepat proses konvergensi pada model MLP.

Tahapan utama penelitian adalah pembangunan model prediksi berbasis MLP yang mampu menangkap pola nonlinier dari hubungan antara indeks iklim global dengan curah hujan bulanan. Tiga variasi arsitektur MLP diusulkan, yaitu dengan tiga *hidden layer* berbeda. Variasi ini untuk menguji pengaruh kompleksitas jaringan terhadap performa prediksi. Dari hasil pelatihan, dilakukan proses seleksi untuk menentukan model terbaik dengan mempertimbangkan nilai *error* dan indikator kinerja lainnya. Model dengan performa paling baik kemudian dijadikan *base learning* untuk proses *ensemble*. Tahap *ensemble* dilakukan dengan menggabungkan hasil dari beberapa arsitektur terbaik guna menghasilkan model yang lebih stabil dan akurat, sekaligus mengurangi potensi kesalahan yang muncul jika hanya mengandalkan satu model tunggal.

Hasil akhir dari keseluruhan proses ini adalah prediksi curah hujan bulanan di Pacet. Prediksi yang dihasilkan kemudian diverifikasi menggunakan data historis dan dapat dipakai untuk mengidentifikasi kategori hujan, termasuk potensi hujan ekstrem >500 mm per bulan yang berkaitan dengan potensi longsor di wilayah Pacet.

3.2 Desain Penelitian

Penelitian ini didesain berdasarkan hasil studi pustaka dan pendeskripsiannya dituangkan dalam kerangka teori. Tahapan yang akan dilalui meliputi pengumpulan data, desain sistem dan implementasi model, eksperimen, pembahasan, dan verifikasi kinerja metode. Tahapan tersebut ditunjukkan dalam Gambar 3.2 sebagai berikut:



Gambar 3.2 Desain Penelitian

3.2.1 Pengumpulan Data

Data *input* yang digunakan dalam penelitian ini bertujuan untuk membangun model prediksi curah hujan berbasis indeks iklim global. Data yang digunakan dalam penelitian ini terbagi menjadi dua kategori utama yaitu, data indeks iklim global meliputi NINO, ONI, IOD, WNPPI, AUSMI yang berfungsi sebagai variabel prediktor untuk model bisa melatih dan menghasilkan nilai prediksi curah hujan. Data input selanjutnya adalah data curah hujan bulanan aktual yang diperoleh dari CHIRPS dan data nilai rata-rata curah hujan klimatologis selama 30 tahun (*baseline*) sebagai variabel target atau pembanding dengan nilai prediksi yang telah dihasilkan oleh

model. Seluruh data *input* ini disajikan dalam bentuk numerik selama periode Januari 1991- Maret 2025 (30 tahun).

A. Data Indeks Iklim Global

Data indeks iklim global digunakan sebagai variabel *input* dalam model prediksi. Indeks iklim global berperan sebagai faktor telekoneksi yang mempengaruhi dinamika curah hujan regional, termasuk di wilayah Indonesia. Indeks yang digunakan dalam penelitian ini meliputi:

1. NINO

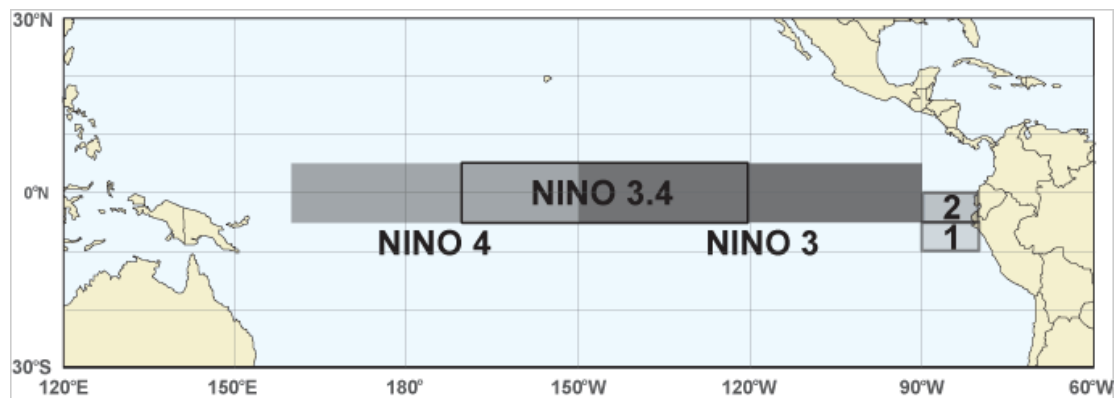
NINO adalah serangkaian indeks yang menggambarkan anomali suhu permukaan laut di wilayah ekuator Pasifik dan dikaitkan dengan fenomena El Nino dan La Nina dalam sistem ENSO (*El Nino Southern Oscillation*). Dalam penelitian ini, digunakan beberapa variasi indeks NINO, yaitu:

- NINO1.2: Mencakup wilayah timur Pasifik dekat pesisir Amerika Selatan (0° – 10° S, 90° W– 80° W). Indeks ini sangat sensitif terhadap fluktuasi suhu yang mendadak, dan sering kali menjadi indikator awal perubahan fase ENSO.
- NINO3: Terletak di wilayah ekuator Pasifik timur-tengah (5° N– 5° S, 150° W– 90° W). Indeks ini penting untuk mengamati perkembangan awal El Nino yang berdampak pada pola curah hujan global.
- NINO3.4: Wilayah antara NINO3 dan NINO4 (5° N– 5° S, 170° W– 120° W). NINO3.4 adalah indeks utama dalam penentuan ONI (*Oceanic Nino Index*), dan paling sering digunakan untuk menilai kekuatan serta durasi El Nino atau

La Nina. Nilai $ONI \geq +0.5$ selama tiga bulan berturut-turut mengindikasikan El Nino, sedangkan nilai ≤ -0.5 menunjukkan La Nina.

- NINO4: Terletak di wilayah Pasifik barat (5°N – 5°S , 160°E – 150°W), sangat dekat dengan wilayah Indonesia. Karena lokasinya, NINO4 memiliki pengaruh langsung terhadap pola curah hujan di wilayah maritim termasuk Indonesia.

Indeks-indeks ini digunakan bersama-sama untuk memberikan pemahaman komprehensif terhadap pengaruh suhu permukaan laut terhadap cuaca dan iklim di Indonesia.



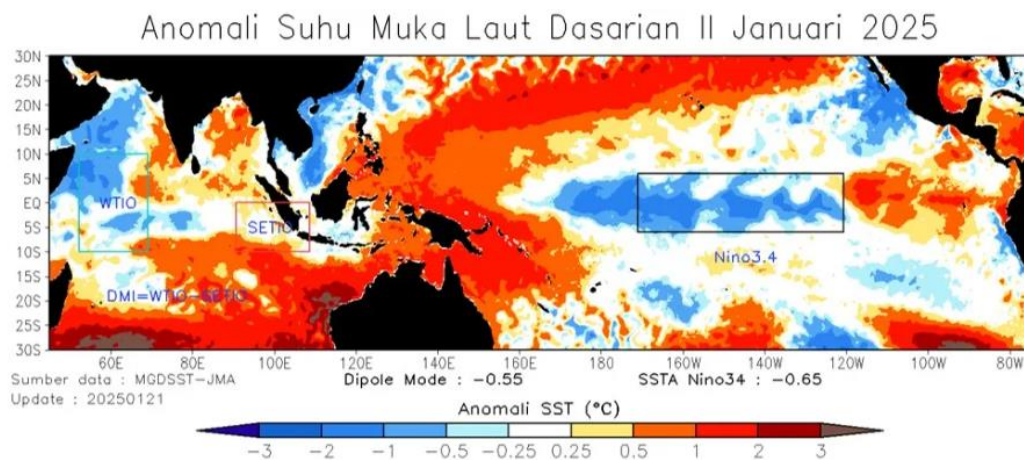
Gambar 3.3 Peta wilayah Indeks ENSO (Sumber : www.noaa.gov)

2. ONI (*Oceanic Nino Index*)

ONI merupakan rata-rata anomali suhu permukaan laut yang berada di wilayah NINO3.4, ONI merupakan nilai dari NINO3.4 yang dihitung selama periode 3 bulan berjalan, dengan *baseline* klimatologis 30 tahun. ONI digunakan sebagai indikator operasional resmi oleh lembaga seperti NOAA untuk menyatakan apakah terjadi El Nino atau La Nina.

3. IOD (*Indeks Indian Ocean Dipole*)

Indeks IOD mengukur perbedaan suhu permukaan laut antara wilayah barat Samudra Hindia (sekitar pantai Afrika Timur) dan bagian timur dekat Indonesia. IOD positif umumnya dikaitkan dengan curah hujan lebih rendah di sebagian besar wilayah Indonesia. Sebaliknya, IOD negatif berkorelasi dengan curah hujan yang meningkat.

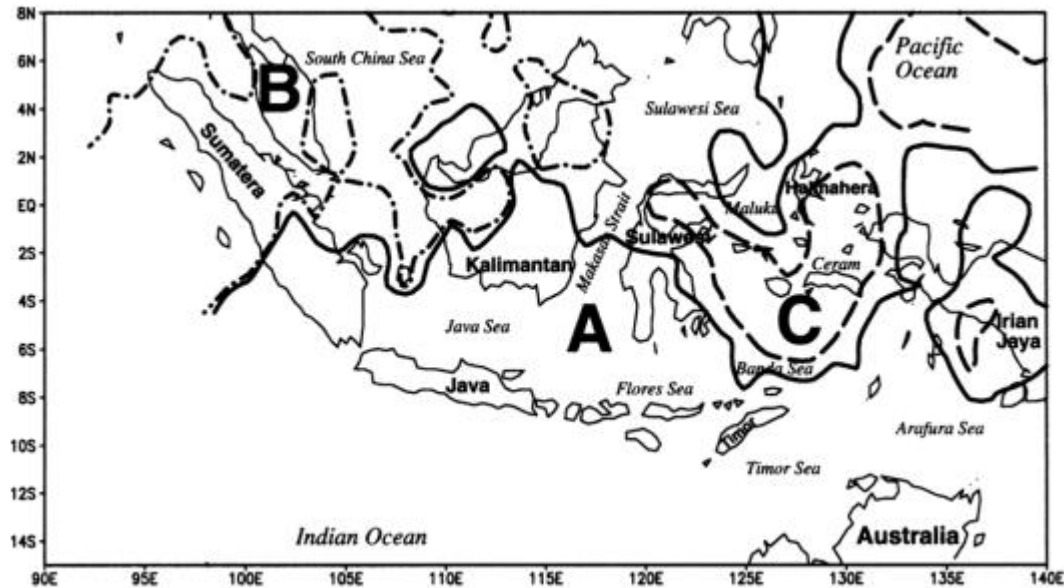


Gambar 3.4 Anomali Suhu Muka Laut (Sumber: www.bmkg.go.id)

4. WNPMI (*Western North Pacific Monsoon Index*)

Indeks WNPMI menggambarkan kekuatan monsun musim panas di wilayah Pasifik barat (sekitar Asia Tenggara). Indeks ini dihitung berdasarkan perbedaan tekanan geopotensial antara dua wilayah tertentu dan berpengaruh besar terhadap intensitas hujan musiman di wilayah Indonesia bagian barat dan tengah. Aldrian dan Susanto (2003) menekankan bahwa Jawa termasuk dalam wilayah yang memiliki curah hujan prediositas 12 bulanan dan berada pada monsun A. Artinya curah hujan di Jawa, salah satunya dipengaruhi dengan WNPMI (*Western North Pacific Monsoon Index*), Dimana monsun A adalah

hujan di wilayah tersebut memiliki pola tahunan yang stabil dan berulang setiap tahun, dan bisa dikategorikan lebih konsisten dan homogen dibandingkan wilayah lainnya (B dan C), menjadikannya lebih prediktif untuk model statistik atau *Machine Learning*.



Gambar 3.5 Wilayah Monsunal Indonesia (Sumber : Edvin dan Susanto (2003)).

5. AUSMI (*Australian Monsoon Index*)

Indeks AUSMI merupakan kekuatan monsun musim panas Australia yang biasanya aktif selama periode Desember sampai Februari. Indeks ini berpengaruh terhadap distribusi hujan di Indonesia bagian selatan termasuk wilayah Jawa Timur. AUSMI diukur dari kecepatan angin dan tekanan udara pada lapisan troposfer bawah.

Semua data indeks iklim tersebut diperoleh dari sumber lembaga iklim internasional terpercaya yaitu NOAA (*National Oceanic and Atmospheric Administration*), dalam format CSV. Untuk indeks WNPPI dan AUSMI, data angin

zonal (arah Timur-Barat) dari NOAA akan digunakan untuk perhitungannya. Apabila terdapat nilai yang hilang (*miss value*), maka akan dilakukan interpolasi linear sesuai dengan rekomendasi Lakshminarayana, et al. (2020). Validasi data dilakukan dengan membandingkan antara sumber data untuk menjamin konsistensi temporal dan keakuratan.

B. Data Curah Hujan Dan Rata-Rata Klimatologis (*Baseline*)

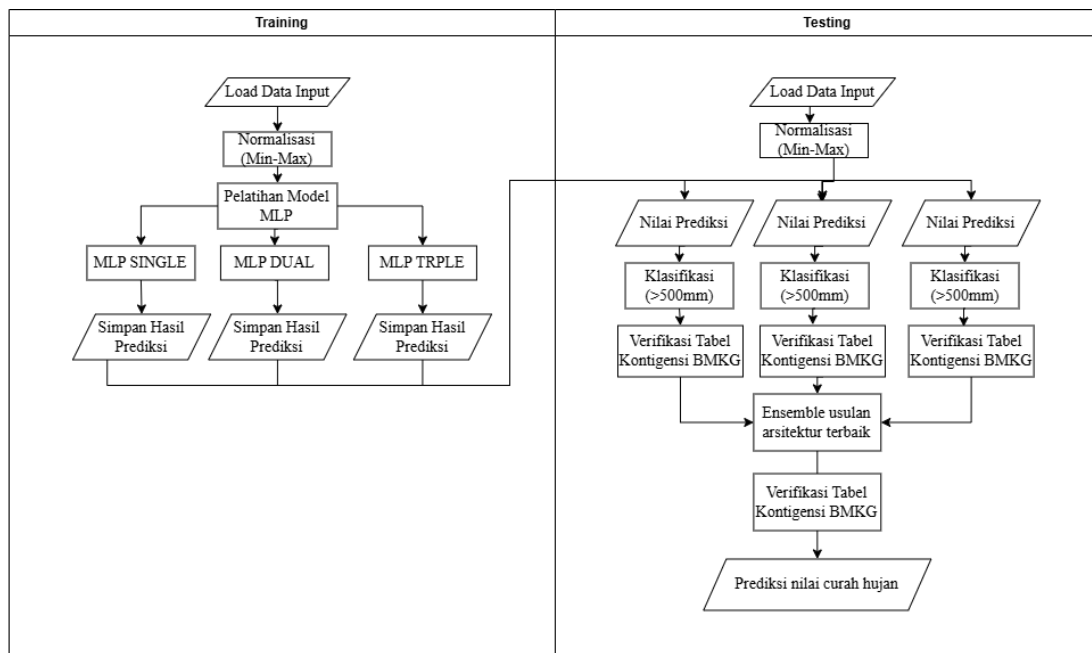
Data curah hujan aktual dari data CHIRPS (*Climate Hazards Group InfraRed Precipitation with Station*) digunakan sebagai variabel target dalam model prediksi. Maharani (2019) menjelaskan bahwa data CHIRPS telah terbukti relevan untuk bisa digunakan di area wilayah Jawa Timur. Data disajikan dalam satuan milimeter (mm), artinya 1 mm setara dengan 1 liter air per meter persegi. Data ini bersifat *time-series* dan telah melalui *preprocessing* seperti deteksi nilai hilang dan normalisasi sebelum digunakan dalam pelatihan model. Data curah hujan aktual dari CHIRPS digunakan sebagai variabel target dalam model prediksi. Dalam konteks prediksi anomali curah hujan, digunakan pula nilai rata-rata klimatologis bulanan (*baseline*) sebagai acuan. Rata-rata klimatologis dihitung berdasarkan rerata curah hujan dalam rentang waktu panjang (30 tahun), yang mewakili kondisi normal tiap bulan. Nilai *baseline* ini tidak digunakan sebagai *input* model, tetapi sebagai referensi untuk mengonversi *output* model (prediksi curah hujan) menjadi nilai curah hujan aktual.

Proporsi pembagian data yang digunakan adalah 80% data sebagai *Training* dan 20% data sebagai *Testing*. Data *training* digunakan untuk membangun dan menyesuaikan bobot model MLP selama proses pelatihan, sedangkan data *testing*

digunakan untuk mengverifikasi performa model setelah pelatihan selesai. Pembagian dilakukan secara berurutan karena data bersifat *time-series* bulanan yang merepresentasikan urutan waktu sehingga menjaga kontinuitas dan representasi kronologis data sangat penting agar tidak terjadi data *leakage*. Pembagian data ditunjukkan pada tabel 3.1 dan skema proses *training* dan *testing* ditunjukkan pada gambar 3.6 sebagai berikut :

Tabel 3.1 Pembagian Data *Training* dan Data *Testing*

Baris Data <i>Training</i>	Data <i>Training</i>	Baris Data <i>Testing</i>	Data <i>Testing</i>
1-328	Januari 1991-Agustus 2018	329-410	September 2018-Maret 2025

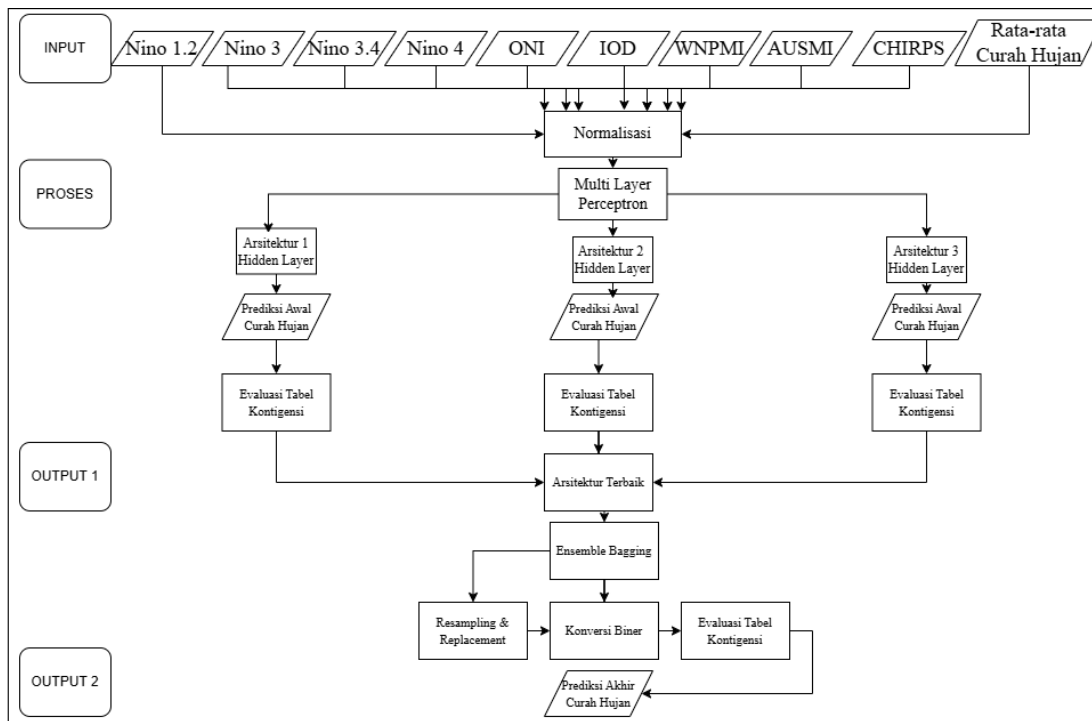


Gambar 3.6 Skema Proses *Training* dan *Testing*.

Tahapan *preprocessing* ini berperan penting karena data yang belum dinormalisasi dan tidak konsisten dapat menyebabkan bias, memperlambat konvergensi, atau bahkan membuat model MLP gagal belajar.

3.2.2 Desain Sistem dan Implementasi

Desain sistem ini bertujuan untuk membangun dan menguji model ANN *Multi-Layer Perceptron* dalam memprediksi curah hujan bulanan. Desain Sistem dalam penelitian ini menggambarkan tahapan dalam melakukan analisis seperti yang ditunjukkan pada gambar 3.6 sebagai berikut :



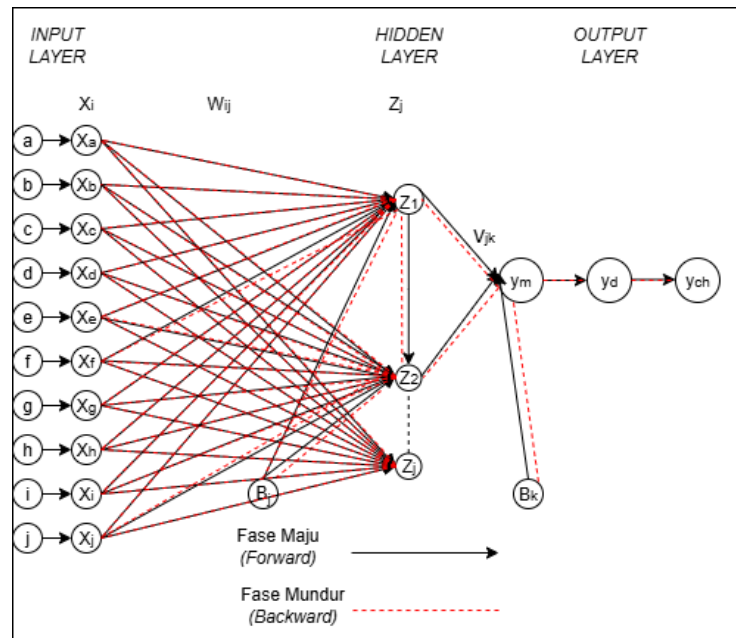
Gambar 3.7 Desain Sistem Penelitian Yang Dilakukan

3.2.3.1 Prediksi Menggunakan MLP (*Multi-Layer Perceptron*)

Model prediksi yang digunakan dalam penelitian ini adalah MLP karena kemampuannya dalam menangkap hubungan nonlinier yang kompleks antara variabel iklim global (NINO, IOD, ONI, dan indeks monsun) dengan curah hujan bulanan. Dalam penelitian ini, dilakukan tiga skenario perancangan arsitektur MLP dengan jumlah *hidden layer* yang berbeda dan diuraikan sebagai berikut:

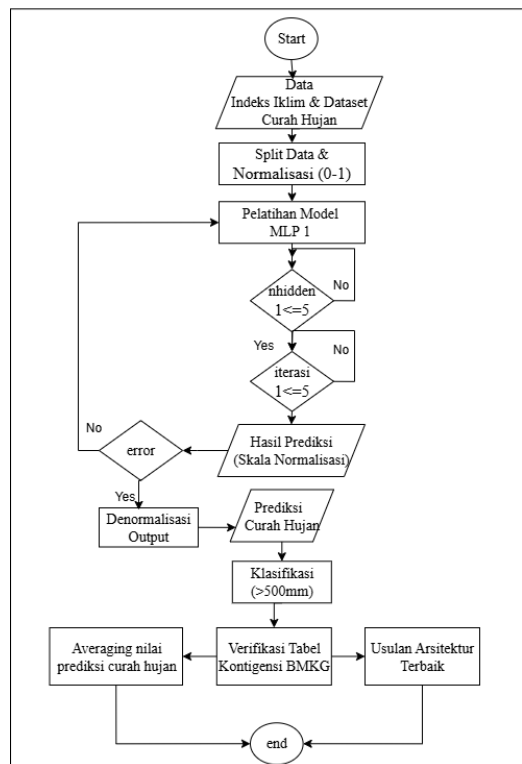
A. Skenario MLP 1 *Hidden Layer*

Pada skenario pertama, arsitektur yang digunakan adalah MLP dengan satu *hidden layer*. Arsitektur ini merupakan bentuk paling sederhana dari MLP, di mana *hidden layer* berfungsi sebagai pengolah utama untuk menangkap pola nonlinier antara variabel *input* dan target *output*. Variabel *input* yang digunakan meliputi indeks iklim global dan indeks monsun, sedangkan variabel *output* yang dihasilkan berupa curah hujan bulanan di Pacet dalam satuan milimeter (mm). Dalam skenario ini, jumlah *neuron* pada *hidden layer* divariasikan menjadi lima arsitektur berbeda untuk memverifikasi pengaruh kapasitas jaringan terhadap performa prediksi curah hujan. Variasi jumlah *neuron* penting karena bisa mengakibatkan *underfitting* jika terlalu sedikit, dan bisa mengakibatkan resiko *overfitting* jika terlalu banyak. Dengan menguji lima variasi *neuron* dari 1 sampai 5, penelitian ini bertujuan mencari arsitektur yang paling optimal dalam menangkap pola data *input* untuk menghasilkan prediksi curah hujan yang lebih akurat. Skenario ini bertujuan untuk melihat kemampuan arsitektur MLP dari yang paling sederhana dalam menghasilkan prediksi curah hujan. Hasil dari skenario ini bertujuan sebagai dasar pembandingan dengan skenario arsitektur MLP yang lebih kompleks (2 dan 3 *Hidden Layer*). Arsitektur pada skenario MLP 1 *Hidden Layer* ditampilkan pada gambar 3.8 sebagai berikut:



Gambar 3.8 Arsitektur MLP 1 *Hidden Layer*

Setiap tahapan pada skenario pengujian ditunjukkan pada gambar 3.8 sebagai berikut:



Gambar 3.9 Flowchart MLP 1 *Hidden Layer*

1. Perhitungan aktivasi pada *hidden layer*

Setiap input dikalikan dengan bobot dan dijumlahkan. Hasilnya, disebut sebagai *input total* ke *neuron hidden layer*, proses ini dihitung menggunakan rumus 3.1 sebagai berikut:

$$z_{net,j} = b_j + \sum_{i=1}^n x_i w_{ij} \quad (3.1)$$

Keterangan:

$z_{net,j}$ = *input total* ke *neuron j* pada *hidden layer*

b_j = bobot bias dari *neuron* ke *hidden layer*

x_i = jumlah *input* ke *i* (hasil normalisasi input $a - j$)

w_{ij} = bobot penghubung *input* dengan *neuron hidden*

n = jumlah *neuron input*

i = indeks untuk *neuron* pada *input layer*

Aktivasi dari setiap *input layer* pada *hidden layer* dihitung menggunakan fungsi aktivasi melalui persamaan rumus 3.2 sebagai berikut,

$$z_j = f(z_{net,j}) + \frac{1}{1+e^{-z_{net,j}}} \quad (3.2)$$

Keterangan:

z_j = *output* aktivasi dari *neuron ke- j* setelah fungsi sigmoid diterapkan

$z_{net,j}$ = *input total* ke *neuron ke- j* pada *hidden layer*

$f(.)$ = fungsi aktivasi sigmoid

e = konstanta matematika

2. Perhitungan *output layer*

Nilai-nilai aktivasi ini kemudian menjadi *input* bagi *neuron output layer*. Proses komputasi dari *hidden* ke *output layer* juga dilakukan dengan menjumlahkan sinyal yang masuk dan mengalikannya dengan bobot yang bersesuaian. Jika opsi `linear.output = FALSE`, maka fungsi aktivasi sigmoid kembali digunakan untuk menghitung *output* dari jaringan, melalui rumus 3.3 dan 3.4 sebagai berikut:

$$y_{net,k} = b_k + \sum_{j=1}^n z_j v_{jk} \quad (3.3)$$

$$y_m = f(y_{net,k}) + \frac{1}{1 + e^{-(b_k + \sum_{j=1}^m y_{net,k})}} \quad (3.4)$$

Keterangan :

y_m = nilai prediksi model MLP (skala normalisasi 0-1)

$f(.)$ = fungsi aktivasi sigmoid

$y_{net,k}$ = total *input* bersih yang diterima oleh *neuron output*

e = konstanta matematika

b_k = bobot bias di *layer output*

m = jumlah *neuron* pada *hidden layer*

z_j = *output* dari *neuron ke j* pada *hidden layer*

v_{jk} = bobot koneksi dari *neuron ke j* pada *hidden layer*

3. Denormalisasi *Output*

Untuk memperoleh nilai prediksi dalam satuan curah hujan aktual (mm), dilakukan proses denormalisasi. Proses mengembalikan hasil prediksi dari

skala normalisasi (hasil prediksi model) ke skala curah hujan aktual (dalam mm) melalui rumus 3.5 sebagai berikut:

$$y_d = y_m (y_x - y_n) + y_n \quad (3.5)$$

Keterangan:

y_d = Hasil prediksi curah hujan setelah denormalisasi.

y_m = Nilai prediksi hasil dari model MLP dalam bentuk skala normalisasi

y_x = Nilai maksimum curah hujan dari data uji

y_n = Nilai minimum curah hujan dari data uji

4. Prediksi Curah Hujan

Dihitung berdasarkan selisih antara prediksi dengan rata-rata klimatologi melalui rumus 3.6 sebagai berikut:

$$y_{ch} = y_d - x_{10} \quad (3.6)$$

Keterangan:

y_{ch} = nilai prediksi curah hujan

y_d = hasil prediksi curah hujan setelah denormalisasi.

x_{10} = nilai rata-rata curah hujan klimatologi (*baseline*).

5. Klasifikasi Curah Hujan

Selanjutnya, setelah hasil prediksi diubah dalam satuan mm maka akan diklasifikasikan berdasarkan ambang batas curah hujan ekstrem menurut BMKG melalui rumus 3.7 sebagai berikut:

$$Klasifikasi = \begin{cases} PD & \text{jika } y_{ch} > 500 \\ T & \text{jika } y_{ch} < 500 \end{cases} \quad (3.7)$$

Keterangan:

y_{ch} = Nilai prediksi curah hujan

500 = Ambang batas ekstrem menurut BMKG untuk wilayah Pacet.

PD = Peringatan Dini, potensi longsor tinggi karena hujan ekstrem.

T = Tidak perlu peringatan, karena curah hujan di bawah ambang batas.

6. Verifikasi Hasil Prediksi

Data aktual dan hasil prediksi dibandingkan dalam bentuk tabel yang dibuat dalam format *confusion matrix*. Digunakan untuk klasifikasi “PD” atau “T” berdasarkan ambang (>500 mm). Parameter dihitung dari tabel melalui rumus 3.8 sampai 3.10 sebagai berikut:

$$Akurasi = \frac{TP+TN}{TP+FP+FN+TN} 100\% \quad (3.8)$$

$$False Alarm = \frac{FP}{FP+TN} 100\% \quad (3.9)$$

$$Miss Rate = \frac{FN}{TP+FN} 100\% \quad (3.10)$$

Keterangan:

TP (*True Positive*): Curah Hujan aktual menunjukkan “PD” dan hasil diprediksi menunjukkan “PD”

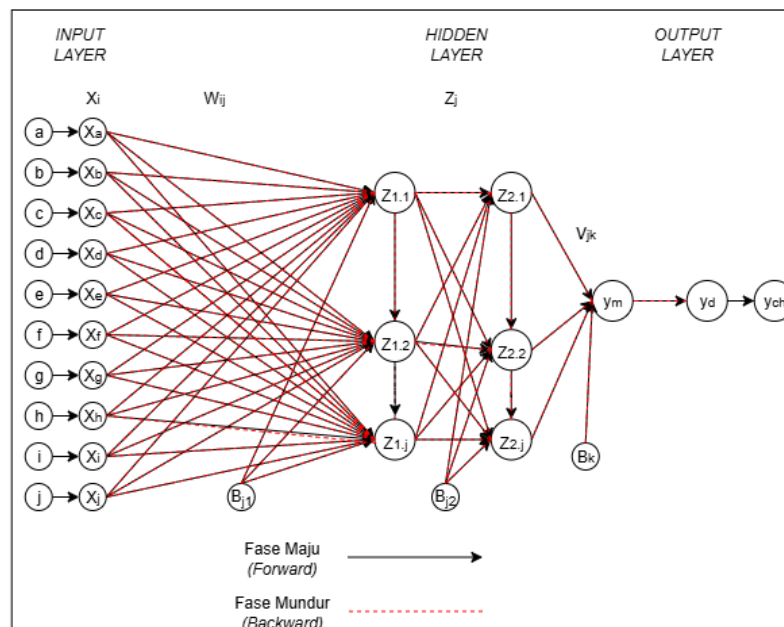
FP (*False Positive*) : Curah Hujan aktual menunjukkan “TIDAK” dan hasil diprediksi menunjukkan “PD” (*False Alarm*)

FN (*False Negative*) : Curah Hujan aktual menunjukkan “PD” dan hasil diprediksi menunjukkan “TIDAK” (*Miss Classification*)

TN (*True Negative*) : Curah Hujan actual menunjukkan “TIDAK” dan hasil diprediksi menunjukkan “TIDAK”.

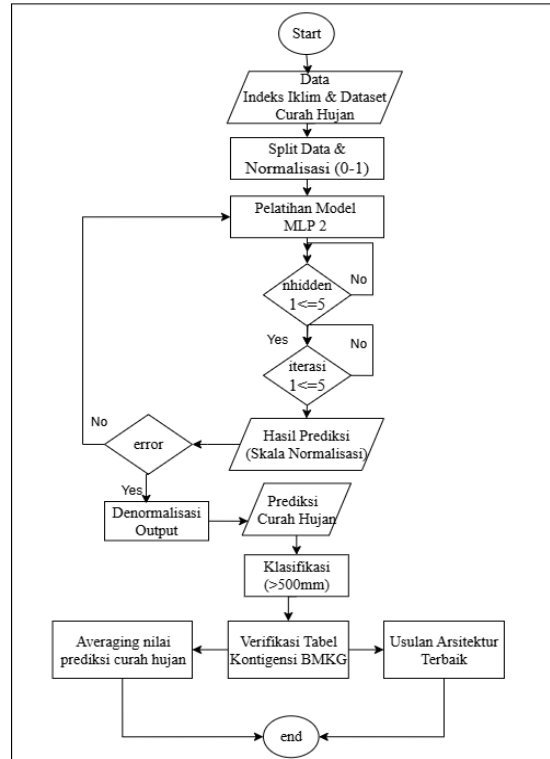
B. Skenario MLP 2 *Hidden Layer*

Skenario kedua menggunakan MLP dengan dua *hidden layer*. Dalam skenario ini, jumlah *neuron* divariasikan secara bertahap menggunakan lima arsitektur berbeda, yaitu (1–1), (2–2), (3–3), (4–4), (5–5). Variasi ini bertujuan mengevaluasi kombinasi kapasitas jaringan yang paling optimal dalam menghasilkan prediksi curah hujan. Melalui perbandingan dengan skenario pertama (1 *hidden layer*), dapat dianalisis apakah penambahan satu *hidden layer* memberikan dampak signifikan terhadap peningkatan akurasi sekaligus efisiensi model. Dengan membandingkan hasil pada skenario, dapat dianalisis apakah penambahan satu *hidden layer* memberikan dampak signifikan terhadap peningkatan akurasi prediksi curah hujan. Arsitektur pada skenario ditunjukkan pada gambar 3.10 sebagai berikut:



Gambar 3.10 Arsitektur MLP 2 *Hidden Layer*

Setiap tahapan pada skenario pengujian ditunjukkan pada gambar 3.11 sebaga berikut :



Gambar 3.11 Flowchart MLP 2 Hidden Layer

1. Perhitungan aktivasi pada *hidden layer*

Setiap input dikalikan dengan bobot dan dijumlahkan. Hasilnya, disebut sebagai input total ke *neuron hidden layer*, proses ini dihitung menggunakan rumus 3.11 sebagai berikut:

$$z_{net,1,j} = b_{j1} + \sum_{i=1}^n x_i w_{ij} \quad (3.11)$$

Keterangan :

$z_{net,1,j}$ = input total ke *neuron j* pada *hidden layer 1*

b_{j1} = bobot bias dari *neuron* ke *j* (*hidden layer 1*)

x_i = jumlah *input* ke *i* (hasil normalisasi input $a - j$)

w_{ij} = bobot penghubung *input* dengan *neuron hidden hidden layer 1*

n = jumlah *neuron input*

i = indeks untuk *neuron* pada *input layer*

Aktivasi dari setiap *input layer* pada *hidden layer* dihitung menggunakan fungsi aktivasi melalui persamaan rumus 3.12 sebagai berikut,

$$z_{j1} = f(z_{net,1,j}) + \frac{1}{1+e^{-z_{net,1,j}}} \quad (3.12)$$

Keterangan:

z_{j1} = *output* aktivasi dari *neuron* ke- j setelah fungsi sigmoid diterapkan

$z_{net,1,j}$ = *input* total *neuron* ke- j pada *hidden layer 1*

$f(.)$ = fungsi aktivasi sigmoid

e = konstanta matematika

2. Perhitungan aktivasi pada *hidden layer 2*

Output dari *hidden layer 1*, menjadi *input* bagi *hidden layer 2*. Proses ini dihitung menggunakan rumus 3.13 sebagai berikut :

$$z_{net,2,j} = b_{j2} + \sum_{j=1}^5 z_{j1} w_{ij2} \quad (3.13)$$

Keterangan:

$z_{net,2,j}$ = *input* total ke *neuron* j pada *hidden layer 2*

b_{j2} = bobot bias dari *neuron* ke j (*hidden layer 2*)

z_{j1} = jumlah *input hidden layer 1* ke *hidden layer 2*

w_{ij2} = bobot penghubung *hidden layer 1* ke *hidden layer 2*

n = jumlah *neuron input*

j = indeks untuk *neuron* pada *hidden layer*

Aktivasi dari setiap *output* pada *hidden layer* sebelumnya menuju *hidden layer* berikutnya dihitung menggunakan fungsi aktivasi melalui persamaan rumus 3.14 sebagai berikut,

$$z_{j2} = f(z_{net,2,j}) + \frac{1}{1+e^{-z_{net,2,j}}} \quad (3.14)$$

Keterangan :

z_{j2} = *output* aktivasi dari *neuron* ke- j setelah fungsi sigmoid diterapkan

$z_{net,2,j}$ = *input* total dari *hidden layer* 1 ke *hidden layer* 2

$f(.)$ = fungsi aktivasi sigmoid

e = konstanta matematika

3. Perhitungan *output layer*

Nilai-nilai aktivasi ini kemudian menjadi input bagi *neuron output layer*. Proses komputasi dari *hidden* ke *output layer* juga dilakukan dengan menjumlahkan sinyal yang masuk dan mengalikannya dengan bobot yang bersesuaian. Jika opsi `linear.output = FALSE`, maka fungsi aktivasi sigmoid kembali digunakan untuk menghitung *output* dari jaringan, melalui rumus 3.15 dan 3.16 sebagai berikut:

$$y_{net,k} = b_k + \sum_{j=1}^5 z_{2,j} v_{jk} \quad (3.15)$$

$$y_m = f(y_{net,k}) + \frac{1}{1+e^{-(b_k + \sum_{j=1}^m y_{net,k})}} \quad (3.16)$$

Keterangan:

y_m = nilai prediksi model MLP (skala normalisasi 0-1)

$f(.)$ = fungsi aktivasi sigmoid

$y_{net,k}$ = total input bersih yang diterima oleh *neuron output*

e = konstanta matematika

b_k = bobot bias di *layer output*

m = jumlah *neuron* pada *hidden layer*

$z_{2,j}$ = *output* dari *neuron ke j* pada *hidden layer 2*

v_{jk} = bobot koneksi menuju *layer output*

4. Denormalisasi *Output*

Untuk memperoleh nilai prediksi dalam satuan curah hujan aktual (mm), dilakukan proses denormalisasi. Proses mengembalikan hasil prediksi dari skala normalisasi (hasil prediksi model) ke skala curah hujan aktual (dalam mm) menggunakan persamaan rumus 3.5 yang sama dengan skenario sebelumnya.

5. Prediksi Curah Hujan

Dihitung berdasarkan selisih antara prediksi dengan rata-rata klimatologi melalui persamaan rumus 3.6 seperti skenario sebelumnya.

(*baseline*).

6. Klasifikasi Curah Hujan

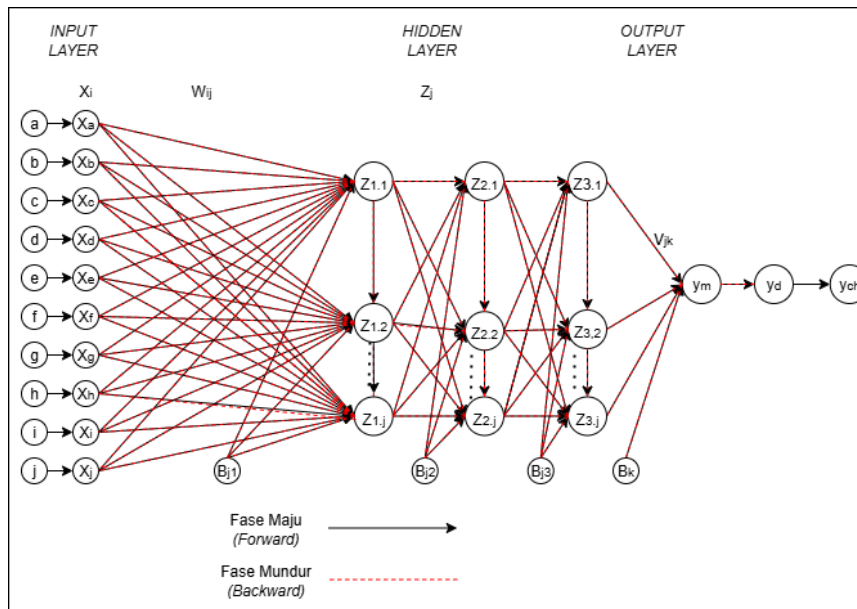
Selanjutnya, setelah hasil prediksi diubah dalam satuan mm maka akan diklasifikasikan berdasarkan ambang batas curah hujan ekstrem menurut BMKG melalui persamaan rumus 3.7 seperti skenario sebelumnya.

7. Verifikasi Hasil Prediksi

Data aktual dan hasil prediksi dibandingkan dalam bentuk tabel yang dibuat dalam format *confusion matrix*. Digunakan untuk klasifikasi “PD” atau “T” berdasarkan ambang (>500 mm). Parameter dihitung dari tabel menggunakan persamaan rumus 3.8 sampai 3.10 seperti skenario sebelumnya.

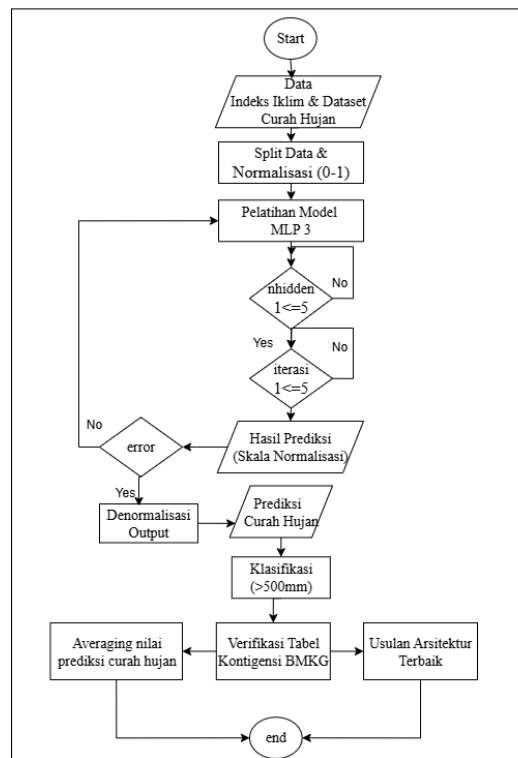
C. Skenario MLP 3 *Hidden Layer*

Pendekatan ini merepresentasikan jaringan yang lebih dalam untuk mengverifikasi sejauh mana penambahan *hidden layer* mampu meningkatkan kinerja prediksi curah hujan. Jumlah *neuron* pada ketiga *hidden layer* divariasikan dalam lima arsitektur berbeda, yaitu (1–1–1), (2–2–2), (3–3–3), (4–4–4), dan (5–5–5). Variasi ini bertujuan menilai dampak kedalaman dan kapasitas jaringan dalam membentuk representasi fitur yang lebih abstrak dari data *input*, sehingga diharapkan dapat menangkap pola iklim yang bersifat nonlinier dan dinamis. Namun, semakin kompleks arsitektur yang digunakan, semakin tinggi pula risiko *overfitting*. Oleh karena itu, skenario ini tidak hanya menjadi uji peningkatan performa, tetapi juga berfungsi untuk menentukan batas optimal antara kedalaman jaringan, akurasi prediksi, serta efisiensi komputasi. Penambahan *hidden layer* secara teoritis dapat memperluas kemampuan jaringan untuk mempelajari interaksi variabel iklim yang saling memengaruhi. Hasil dari skenario ini diharapkan mampu menunjukkan titik keseimbangan antara model yang sederhana namun kurang akurat dengan model yang kompleks namun rentan *overfitting*. Dengan demikian, skenario ini berperan penting dalam merumuskan rekomendasi arsitektur MLP yang tepat guna untuk prediksi curah hujan di Pacet.



Gambar 3.12 Arsitektur MLP 3 Hidden Layer

Langkah-langkah tersebut ditunjukkan pada gambar 3.13 sebagai berikut:



Gambar 3.13 Flowchart MLP 3 Hidden Layer

1. Perhitungan aktivasi pada *hidden layer*

Setiap input dikalikan dengan bobot dan dijumlahkan. Hasilnya, disebut sebagai input total ke *neuron hidden layer*, proses ini dihitung menggunakan rumus 3.17 sebagai berikut:

$$z_{net,1,j} = b_{j1} + \sum_{i=1}^n x_i w_{ij} \quad (3.17)$$

Keterangan:

$z_{net,1,j}$ = input total ke *neuron j* pada *hidden layer 1*

b_{j1} = bobot bias dari *neuron ke j* (*hidden layer 1*)

x_i = jumlah *input ke i* (hasil normalisasi input $a - j$)

w_{ij} = bobot penghubung *input* dengan *neuron hidden hidden layer 1*

n = jumlah *neuron input*

i = indeks untuk *neuron pada input layer*

Aktivasi dari setiap *input layer* pada *hidden layer* dihitung menggunakan fungsi aktivasi melalui persamaan rumus 3.18 sebagai berikut,

$$z_{j1} = f(z_{net,j1}) + \frac{1}{1+e^{-z_{net,1,j}}} \quad (3.18)$$

Keterangan:

z_{j1} = output aktivasi dari *neuron ke- j* setelah fungsi sigmoid diterapkan

$z_{net,1,j}$ = input total *neuron ke- j* pada *hidden layer 1*

$f(.)$ = fungsi aktivasi sigmoid

e = konstanta matematika

2. Perhitungan aktivasi pada *hidden layer 2*

Output dari *hidden layer* 1, menjadi *input* bagi *hidden layer* 2. Proses ini dihitung menggunakan rumus 3.19 sebagai berikut:

$$z_{net,2,j} = b_{j2} + \sum_{j=1}^5 z_{j1} w_{ij2} \quad (3.19)$$

Keterangan:

$z_{net,2,j}$ = *input* total ke *neuron j* pada *hidden layer* 2

b_{j2} = bobot bias dari *neuron* ke j (*hidden layer* 2)

z_{j1} = jumlah *input hidden layer* 1 ke *hidden layer* 2

w_{ij2} = bobot penghubung *hidden layer* 1 ke *hidden layer* 2

n = jumlah *neuron input*

j = indeks untuk *neuron* pada *hidden layer*

Aktivasi dari setiap output pada *hidden layer* sebelumnya menuju *hidden layer* berikutnya dihitung menggunakan fungsi aktivasi melalui persamaan rumus 3.20 sebagai berikut,

$$z_{j2} = f(z_{net,2,j}) + \frac{1}{1+e^{-z_{net,2,j}}} \quad (3.20)$$

Keterangan:

z_{j2} = *output* aktivasi dari *neuron* ke- j setelah fungsi sigmoid diterapkan

$z_{net,2,j}$ = *input* total dari *hidden layer* 1 ke *hidden layer* 2

$f(.)$ = fungsi aktivasi sigmoid

e = konstanta matematika

3. Perhitungan aktivasi pada *hidden layer* 3

Output dari *hidden layer* 2, menjadi input bagi *hidden layer* 3. Proses ini dihitung menggunakan rumus 3.21 sebagai berikut:

$$z_{net,3,j} = b_{j3} + \sum_{j=1}^5 z_{j2} w_{ij3} \quad (3.21)$$

Keterangan:

$z_{net,3,j}$ = input total ke *neuron j* pada *hidden layer* 3

b_{j3} = bobot bias dari *neuron* ke j (*hidden layer* 3)

z_{j2} = jumlah input *hidden layer* 2 ke *hidden layer* 3

w_{ij} = bobot penghubung *hidden layer* 2 ke *hidden layer* 3

n = jumlah *neuron* input

j = indeks untuk *neuron* pada *hidden layer*

Aktivasi dari setiap output pada *hidden layer* sebelumnya menuju *hidden layer* berikutnya dihitung menggunakan fungsi aktivasi melalui persamaan rumus 3.22 sebagai berikut:

$$z_{j3} = f(z_{net,3,j}) + \frac{1}{1+e^{-z_{net,3,j}}} \quad (3.22)$$

Keterangan:

z_{j3} = output aktivasi dari *neuron* ke- j setelah fungsi sigmoid diterapkan

$z_{net,3,j}$ = input total ke *neuron* ke- j pada *hidden layer*

$f(.)$ = fungsi aktivasi sigmoid

e = konstanta matematika

4. Perhitungan *output layer*

Nilai-nilai aktivasi ini kemudian menjadi input bagi *neuron output layer*. Proses komputasi dari *hidden* ke *output layer* juga dilakukan dengan menjumlahkan sinyal yang masuk dan mengalikannya dengan bobot yang bersesuaian. Jika opsi `linear.output = FALSE`, maka fungsi aktivasi sigmoid kembali digunakan untuk menghitung *output* dari jaringan, melalui rumus 3.23 dan 3.24 sebagai berikut:

$$y_{net,k} = b_k + \sum_{j=1}^5 z_{3,j} v_{jk} \quad (3.23)$$

$$y_m = f(y_{net,k}) + \frac{1}{1+e^{-(b_k + \sum_{j=1}^m y_{net,k})}} \quad (3.24)$$

Keterangan:

y_m = nilai prediksi model MLP (skala normalisasi 0-1)

$f(.)$ = fungsi aktivasi sigmoid

$y_{net,k}$ = total input bersih yang diterima oleh *neuron output*

e = konstanta matematika

b_k = bobot bias di *layer output*

m = jumlah *neuron* pada *hidden layer*

$z_{3,j}$ = *output* dari *neuron ke j* pada *hidden layer 3*

v_{jk} = bobot koneksi menuju *layer output*

5. Denormalisasi *Output*

Untuk memperoleh nilai prediksi dalam satuan curah hujan aktual (mm), dilakukan proses denormalisasi. Proses mengembalikan hasil prediksi dari

skala normalisasi (hasil prediksi model) ke skala curah hujan aktual (dalam mm) menggunakan persamaan rumus 3.5 seperti skenario MLP sebelumnya.

6. Prediksi Curah Hujan

Dihitung berdasarkan selisih antara prediksi dengan rata-rata klimatologi menggunakan rumus 3.6 seperti skenario MLP sebelumnya.

7. Klasifikasi Curah Hujan

Selanjutnya, setelah hasil prediksi diubah dalam satuan mm maka akan diklasifikasikan berdasarkan ambang batas curah hujan ekstrem menurut BMKG menggunakan persamaan rumus 3.7 seperti skenario MLP sebelumnya.

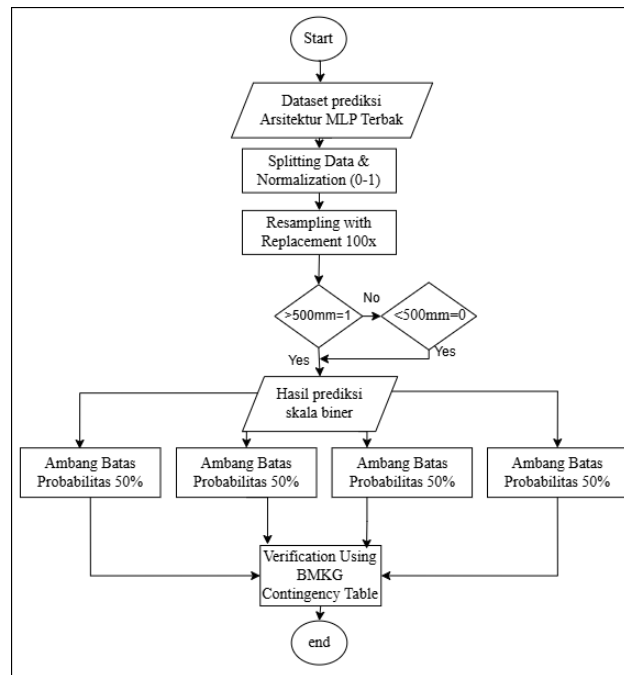
8. Verifikasi Hasil Prediksi

Data aktual dan hasil prediksi dibandingkan dalam bentuk tabel yang dibuat dalam format *confusion matrix*. Digunakan untuk klasifikasi “PD” atau “T” berdasarkan ambang (>500 mm). Parameter dihitung dari tabel melalui rumus 3.8 sampai 3.10 seperti skenario MLP sebelumnya

Setelah setiap skenario arsitektur MLP (*Multilayer Perceptron*) dilatih dan diverifikasi, model terbaik dari masing-masing arsitektur *hidden layer* ditetapkan sebagai *base learner* untuk membentuk model *Ensemble Bagging*. Tujuan utama dari pendekatan ini adalah meningkatkan stabilitas dan reliabilitas prediksi terhadap fluktuasi data iklim yang kompleks, dengan cara mengurangi variansi hasil model tunggal.

3.2.3.2 Prediksi Menggunakan *Ensemble*

Pada tahap ini, metode *Ensemble Bagging* diterapkan sebagai pendekatan lanjutan dengan tujuan untuk meningkatkan stabilitas dan akurasi prediksi curah hujan bulanan di Pacet. Pendekatan *ensemble* digunakan untuk mengetahui skenario *ensemble* apakah dapat menghasilkan prediksi yang lebih baik dibandingkan dengan model MLP tunggal, karena proses pelatihan ANN (*Artificial Neural Network*) sangat sensitif terhadap inisialisasi bobot, pembagian data, serta variasi pola indeks iklim global yang bersifat dinamis. Oleh karena itu, *ensemble* berperan sebagai mekanisme yang mampu mengurangi variansi dan memberikan prediksi yang lebih konsisten. Dalam metode *bagging*, model arsitektur MLP terbaik digunakan sebagai *base learning* untuk dilatih menggunakan proses *bootstrap resampling* pada data *testing* dengan melakukan *resampling* sebanyak 100 kali untuk memastikan bahwa hasil verifikasi tidak dipengaruhi oleh satu kondisi data tertentu. Setelah seluruh model selesai diresampling, hasil prediksi setiap model dihitung menggunakan ambang batas probabilitas (50%, 60%, 70%, 80%, dan 90%), selanjutnya, dilakukan verifikasi menggunakan tabel kontigensi BMKG. Setiap tahapan pada skenario *ensemble* ditampilkan pada gambar 3.14 sebaga berikut:



Gambar 3.14 Flowchart Ensemble Bagging MLP

Langkah-langkah proses *ensemble bagging* sebagai berikut:

1. *Resampling with replacement*

Dari dataset pengujian (nilai prediksi dari arsitektur MLP terbaik) dilakukan pengambilan sampel secara acak dengan pengembalian (*resampling with replacement*) sebanyak 100 kali untuk membentuk himpunan data bootstrap melalui persamaan 3.25 berikut:

$$y_{ens} = \{y_{ch1}, y_{ch2}, \dots, y_{ch100}\} \quad (3.25)$$

Keterangan:

y_{ch} = Nilai prediksi curah hujan

y_{ens} = Nilai prediksi curah hujan yang telah diensemble

Setiap himpunan y_{ens} berukuran sama dengan data asli, namun memiliki komposisi sampel yang bervariasi karena proses pengambilan acak dengan pengembalian.

2. Binerisasi Hasil Prediksi

Nilai prediksi dari proses *resampling* dikonversi ke bentuk biner (0 dan 1). Proses ini dilakukan untuk mempermudah klasifikasi kejadian hujan tinggi (anomali curah hujan ekstrem) melalui persamaan 3.26 sebagai berikut:

$$y_{bin} = \begin{cases} 1 & \text{jika } y_{ensm} > 500 \\ 0 & \text{jika } y_{ensm} < 500 \end{cases} \quad (3.26)$$

Keterangan:

y_{bin} = Nilai prediksi curah hujan yang dikonversi ke biner

y_{ensm} = nilai prediksi *ensemble*

1 = Peringatan Dini, potensi longsor tinggi karena hujan ekstrem.

0 = Tidak perlu peringatan, karena curah hujan di bawah ambang batas.

3. Threshold Probability

Selanjutnya, nilai prediksi yang telah dikonversi ke biner (0 dan 1) dihitung menggunakan ambang batas probabilitas (50%, 60%, 70%, 80%, dan 90%) berdasarkan 100 sampel hasil *resampling* untuk menentukan apakah keadaan akhir diklasifikasikan sebagai hujan ekstrem atau tidak. Proses ini dilakukan untuk menilai sensitivitas model dalam mendeteksi kejadian hujan tinggi (anomali curah hujan ekstrem) melalui persamaan 3.27 dan 3.28 sebagai berikut:

$$y_p = \frac{1}{n} \sum_{i=1}^n y_{bin} \quad (3.27)$$

Keterangan:

y_{bin} = Nilai prediksi curah hujan yang dikonversi ke biner

y_p = nilai prediksi terhadap jumlah total kejadian ekstrem

n = Jumlah resampling

$$y_{pt} = \begin{cases} 1 & \text{jika } y_p > t \\ 0 & \text{jika } y_p < t \end{cases} \quad (3.28)$$

Keterangan:

y_{pt} = Nilai probabilitas kejadian hujan ekstrem yang diprediksi *ensemble*

y_p = nilai prediksi terhadap jumlah total kejadian ekstrem

t = nilai probabilitas (50%, 60%, 70%,80%,90%)

a. *Threshold 50%*

Ambang 50% merupakan prediksi akhir ditetapkan sebagai ekstrem jika setengah atau lebih dari nilai y_{pt} yang menghasilkan nilai 1 itu lebih dari 50”.

b. *Threshold 60%*

Ambang 60% merupakan prediksi akhir ditetapkan sebagai ekstrem jika setengah atau lebih dari nilai y_{pt} yang menghasilkan nilai 1 itu lebih dari 60”.

c. *Threshold 70%*

Ambang 70% merupakan prediksi akhir ditetapkan sebagai ekstrem jika setengah atau lebih dari nilai y_{pt} yang menghasilkan nilai 1 itu lebih dari 70”.

d. *Threshold 80%*

Ambang 80% merupakan prediksi akhir ditetapkan sebagai ekstrem jika setengah atau lebih dari nilai y_{pt} yang menghasilkan nilai 1 itu lebih dari 80”.

e. *Threshold* 90%

Ambang 90% merupakan prediksi akhir ditetapkan sebagai ekstrem jika setengah atau lebih dari nilai y_{pt} yang menghasilkan nilai 1 itu lebih dari 90”.

4. Verifikasi Menggunakan Parameter Kontigensi BMKG

Hasil biner dari *ensemble* kemudian diverifikasi menggunakan tabel kontingensi BMKG, dengan parameter akurasi, *false alarm*, *miss classification* (menggunakan persamaan rumus 3.8 sampai 3.10 seperti pada skenario MLP) Verifikasi ini dilakukan untuk setiap nilai ambang batas (50%–90%), sehingga diperoleh tren stabilitas dan kepekaan model terhadap variasi *threshold*.

5. Perbandingan dengan arsitektur MLP tunggal terbaik

Nilai rata-rata hasil dari model *ensemble* kemudian dibandingkan dengan performa model MLP tunggal terbaik. Perbandingan ini bertujuan untuk mengetahui apakah pendekatan *bagging* mampu memberikan peningkatan performa, khususnya dalam aspek stabilitas prediksi dan penurunan tingkat kesalahan deteksi hujan ekstrem.

3.2.3.2 Implementasi

Setelah desain sistem, tahapan berikutnya adalah implementasi sistem yang akan dilakukan dengan merancang *pseudocode* dengan skenario yang sudah ditetapkan untuk variasi *hidden layer* dan *neuron* pada proses pelatihan model MLP Setiap tahapan sebagai berikut:

No	Pseudocode
1	<i>input = DATA_INTENET.csv</i>
2	<i>batasi data hingga baris ke-410</i>
3	<i>inisialisasi kolom kosong KAT_ANOM_CH</i>
4	<i>buat formula ANN berdasarkan input dan output (form3)</i>
5	<i>bagi data menjadi 80% training dan 20% testing</i>
6	<i>normalisasi data training dan testing (min-max normalization)</i>
7	<i>for arsitektur $\in \{single, dual, triple\}$ do</i>
8	<i>if arsitektur = single then hidden_config = c(n)</i>
9	<i>else if arsitektur = dual then hidden_config = c(n, n)</i>
10	<i>else if arsitektur = triple then hidden_config = c(n, n, n)</i>
11	<i>nhidden = 1</i>
12	<i>while nhidden ≤ 5 do</i>
13	<i>iterasi = 1</i>
14	<i>while iterasi ≤ 30 do</i>
15	<i>latih model MLP menggunakan fungsi neuralnet() dengan arsitektur hidden_config</i>
16	<i>hasil_pelatihan = model\$result.matrix</i>
17	<i>lakukan prediksi: predict(model, test_data_nrm)</i>
18	<i>transformasi output ke skala anomali dan curah hujan aktual</i>
19	<i>klasifikasikan curah hujan ≥ 500 mm/bulan</i>
20	<i>hitung metrik verifikasi: akurasi, false alarm, miss, error, dan step</i>
21	<i>simpan hasil ke file verifikasi sesuai arsitektur (VERIF_SINGLE, VERIF_DUAL, VERIF_TRIPLE)</i>
22	<i>iterasi = iterasi + 1</i>
23	<i>end while</i>
24	<i>nhidden = nhidden + 1</i>
25	<i>end while</i>
26	<i>end for</i>
27	<i>kumpulkan semua file hasil verifikasi dari folder OUTPUT/VERIF</i>
28	<i>gabungkan hasil ke satu file REKAP_VERIF_MLP.csv</i>

Pseudocode di atas menggambarkan implementasi untuk tiga arsitektur jaringan MLP, yaitu 1 *hidden layer*, 2 *hidden layer*, dan 3 *hidden layer*. Proses pelatihan dilakukan secara berulang dengan variasi jumlah *neuron* pada tiap *hidden layer* (1–5) dan pengulangan iterasi sebanyak 30 kali untuk memperoleh hasil yang konsisten. Setiap hasil pelatihan kemudian diverifikasi berdasarkan parameter verifikasi meliputi akurasi, *false alarm*, *miss classification*, lalu disimpan dalam file verifikasi masing-

masing arsitektur. Seluruh hasil akhir digabung dalam satu file sebagai dasar pemilihan arsitektur terbaik.

Model dengan kombinasi performa optimal selanjutnya ditetapkan sebagai arsitektur terbaik dan digunakan untuk tahap *ensemble bagging*, yang bertujuan meningkatkan reliabilitas dan stabilitas hasil prediksi curah hujan ekstrem di Pacet.

Setiap tahapan sebagai berikut:

No	Pseudocode
1	<i>input = DATA INTENET.csv</i>
2	<i>gunakan arsitektur terbaik hasil verifikasi (BEST MLP MODEL)</i>
3	<i>bagi data menjadi 80% training dan 20% testing</i>
4	<i>normalisasi data (min-max normalization)</i>
5	<i>set jumlah resampling = 100</i>
6	<i>for i = 1 to 100 do</i>
7	<i>lakukan resampling with replacement pada data training → boot data</i>
8	<i>latih model MLP dengan boot_data menggunakan arsitektur BEST MLP MODEL</i>
9	<i>lakukan prediksi predict(model i, test data nrm)</i>
10	<i>simpan hasil prediksi probabilitas kejadian ekstrem ke dalam PROB_ENSEMBLE[i]</i>
11	<i>end for</i>
12	<i>hitung rata-rata probabilitas dari seluruh model: P_agg = mean(PROB_ENSEMBLE)</i>
13	<i>for threshold ∈ {0.5, 0.6, 0.7, 0.8, 0.9} do</i>
14	<i>binerkan hasil prediksi: jika P_agg ≥ threshold maka prediksi = 1 (kejadian hujan ekstrem), else = 0</i>
15	<i>lakukan verifikasi hasil terhadap observasi aktual menggunakan matriks kontingensi BMKG</i>
16	<i>hitung parameter: akurasi, false alarm, miss classification</i>
17	<i>simpan hasil verifikasi untuk tiap threshold ke file VERIF_ENSEMBLE.csv</i>
18	<i>end for</i>
19	<i>bandingkan hasil verifikasi ensemble dengan hasil MLP tunggal</i>
20	<i>tentukan arsitektur threshold terbaik berdasarkan keseimbangan antara akurasi tinggi dan kesalahan rendah</i>

3.2.3 Eksperimen

Tahapan eksperimen dalam penelitian ini bertujuan untuk memastikan bahwa model prediksi curah hujan yang telah dirancang dapat berjalan secara optimal sesuai dengan tujuan penelitian. Eksperimen dilakukan dengan menggunakan data *input* indeks iklim global dan data curah hujan bulanan serta rata-rata curah hujan klimatologis yang telah melalui tahap *preprocessing*, normalisasi, dan pembagian data latih serta data uji. Hasil prediksi sampel dari masing-masing arsitektur MLP tertera pada Tabel 3.2 hingga Tabel 3.4 sebagai berikut :

Tabel 3.2 Sampel *Output* Arsitektur MLP 1

Prediksi MLP	Hasil Denormalisasi	Rerata Curah Hujan	Prediksi Curah Hujan	Observasi Curah Hujan	Keterangan
0.421	-24.4364	99.0962	74.7	83.6	SINGLE_1
0.388	-47.7455	25.36599	0	24.9	SINGLE_2
0.417	-27.2618	29.87074	2.6	14.1	SINGLE_3
0.447	-6.07171	29.87074	23.8	14.1	SINGLE_4
0.386	-49.1582	25.36599	0	24.9	SINGLE_5

Tabel 3.3 Sampel *Output* Arsitektur MLP 2

Prediksi MLP	Hasil Denormalisasi	Rerata Curah Hujan	Prediksi Curah Hujan	Observasi Curah Hujan	Keterangan
0.42	-25.1428	99.0962	74	83.6	DUAL_1.1
0.389	-47.0392	25.36599	0	24.9	DUAL_2.2
0.411	-31.4998	99.0962	67.6	83.6	DUAL_3.3
0.389	-47.0392	25.36599	0	24.9	DUAL_4.4
0.421	-24.4364	99.0962	74.7	83.6	DUAL_5.5

Tabel 3.4 Sampel *Output* Arsitektur MLP 3

Prediksi MLP	Hasil Denormalisasi	Rerata Curah Hujan	Prediksi Curah Hujan	Observasi Curah Hujan	Keterangan
0.419	-25.8491	99.0962	73.2	83.6	TRIPLE_1.1.1
0.456	0.285305	25.36599	25.7	24.9	TRIPLE_2.2.2
0.425	-21.6111	99.0962	77.5	83.6	TRIPLE_3.3.3
0.394	-43.5075	25.36599	0	24.9	TRIPLE_4.4.4
0.41	-32.2061	99.0962	66.9	83.6	TRIPLE_5.5.5

Selanjutnya setelah semua arsitektur model dilatih dan menghasilkan prediksi curah hujan, tahapan selanjutnya adalah membandingkan dengan data curah hujan hasil observasi dari CHIRPS untuk diverifikasi menggunakan parameter kontigensi BMKG dengan tiga variabel utama yaitu akurasi, *false alarm*, dan *miss clasisation* guna menentukan model terbaik yang akan digunakan pada tahap *ensemble* untuk meningkatkan akurasi dan reliabilitas prediksi. Berdasarkan beberapa arsitektur model yang memiliki performa paling baik dari setiap skenario yang dilakukan, maka direkomendasikan arsitektur yang tertera pada tabel 3.5 sebagai berikut:

Tabel 3.5 Rekomendasi Arsitektur Tiap Skenario

Variabel Verifikasi	Nilai Tertinggi Skenario MLP 1 (Jumlah <i>Neuron</i>)	Nilai Tertinggi Skenario MLP 2 (Jumlah <i>Neuron</i>)	Nilai Tertinggi Skenario MLP 3 (Jumlah <i>Neuron</i>)
Akurasi	87.5 (1 dan 3)	88.4 (1.1)	87.9 (3.3.3)
<i>False Alarm</i>	23.1 (3)	22.8 (1.1)	21.8 (1.1.1)
<i>Miss Classification</i>	33.3 (1 dan 3)	29.3 (1.1)	31.9 (3.3.3)

Dari hasil tersebut, kemudian diterapkan pendekatan *ensemble bagging*, yaitu dengan menghitung nilai rata-rata prediksi dari ketiga model terbaik untuk setiap bulan pengamatan.

3.3 Instrumen Penelitian

Penelitian ini melibatkan beberapa variabel yang terdiri dari variabel independen, antara (*intervening*), dan dependen yang digunakan dalam pembangunan model prediksi curah hujan bulanan berbasis ANN (*Artificial Neural Network*) model MLP (*Multi-Layer Perceptron*).

A. Variabel independen dalam penelitian ini adalah

1. Indeks iklim global yang terbukti secara ilmiah memengaruhi pola curah hujan di wilayah Indonesia, khususnya Pacet. Indeks yang digunakan antara lain:
 - NINO1.2, NINO3, NINO3.4, NINO4, yang merepresentasikan anomali suhu permukaan laut (SST) di kawasan Pasifik dan menjadi indikator utama dari fenomena El Nino–La Nina;
 - ONI (*Oceanic Niño Index*) sebagai pengukuran rata-rata SST NINO3.4 selama tiga bulan berturut-turut;
 - DMI (*Dipole Mode Index*) untuk menangkap pengaruh Indian Ocean Dipole;
 - WNPPI (*Western North Pacific Monsoon Index*) dan AUSMI (*Australian Monsoon Index*) yang merepresentasikan kekuatan sistem monsun yang

mempengaruhi curah hujan di Indonesia bagian selatan termasuk Jawa Timur.

2. Data curah hujan bulanan yang didapat dari CHIRPS sebagai variabel pembanding terhadap nilai prediksi curah hujan yang dihasilkan model
 3. Nilai rata-rata klimatologis curah hujan dalam rentang waktu 30 tahun (*baseline*) sebagai variabel untuk menentukan nilai anomali curah hujan
- B. Variabel antara (*intervening*) pada penelitian ini adalah hasil prediksi curah hujan (dalam satuan milimeter) dan anomali curah hujan yang dihasilkan dari pemrosesan data *input* oleh MLP. Nilai prediksi inilah yang kemudian dibandingkan dengan data aktual CHIRPS untuk dinilai tingkat akurasinya.
- C. Variabel dependen dalam penelitian ini adalah akurasi model prediksi curah hujan bulanan yang diukur berdasarkan selisih antara nilai prediksi dan nilai aktual. Verifikasi performa dilakukan menggunakan parameter tabel kontingensi yaitu akurasi, *false alarm*, dan *miss classification* berdasarkan ambang batas ekstrem >500mm perbulan.

Seluruh variabel dikumpulkan dan diproses menggunakan perangkat lunak R Studio dengan bantuan *library neuralnet*, *dplyr*, dan *base*, serta *preprocessing* melalui normalisasi (*min-max scaling*) agar nilai antar variabel berada pada skala yang sebanding dan dapat diproses secara optimal dalam arsitektur MLP.

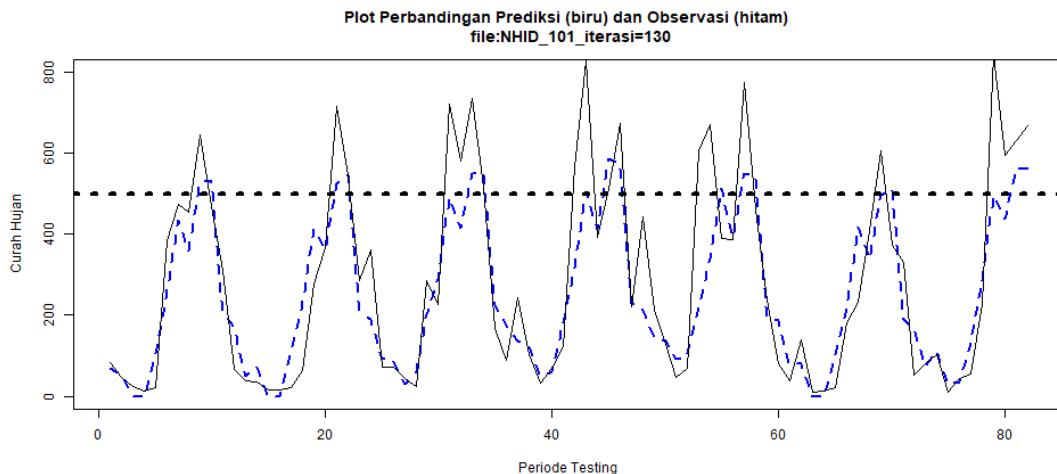
BAB IV

MLP UNTUK PREDIKSI CURAH HUJAN

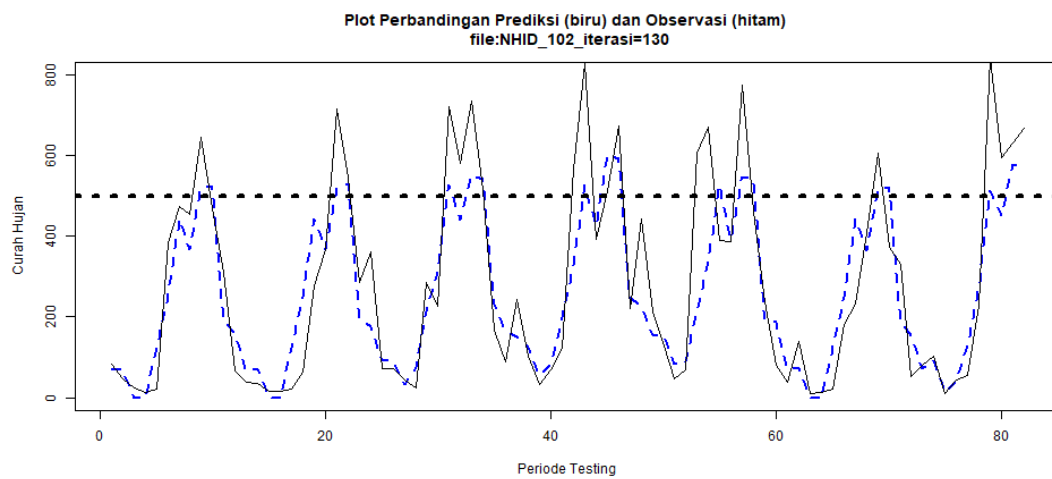
4.1 Prediksi Curah Hujan Menggunakan MLP 1 *Hidden Layer*

Pada skenario pertama, digunakan arsitektur MLP 1 *hidden layer*. Jumlah *neuron* pada *hidden layer* divariasikan 5 *neuron* (1 sampai 5) yang bertujuan untuk memverifikasi pengaruh kapasitas jaringan terhadap kinerja prediksi curah hujan bulanan. Proses pelatihan dilakukan menggunakan *library neuralnet* pada *platform* R Studio.

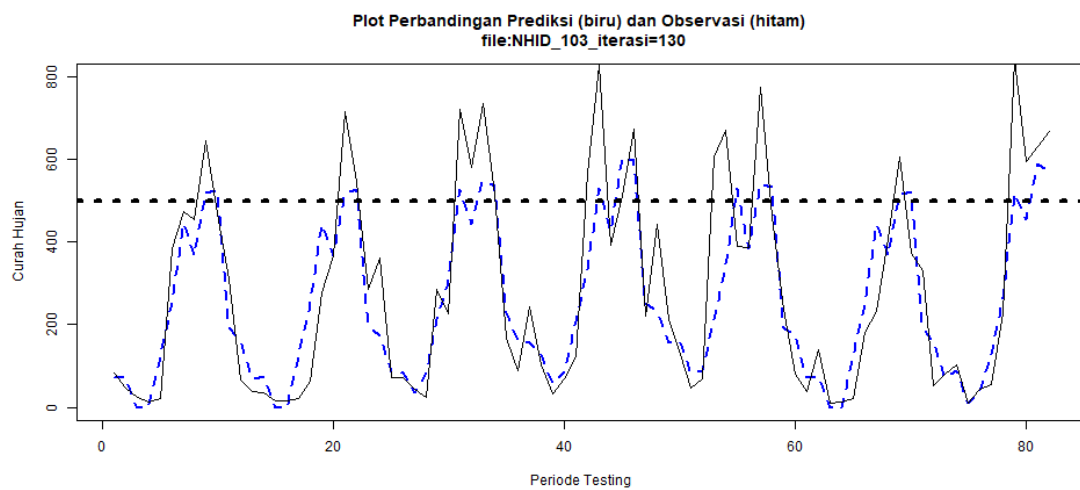
Hasil prediksi curah hujan dari MLP 1 *hidden layer* dibandingkan dengan data curah hujan CHIRPS pada setiap arsitektur. Grafik perbandingan hasil prediksi curah hujan di Pacet dengan data curah hujan CHIRPS menggunakan arsitektur model MLP 1 *hidden layer* ditampilkan pada Gambar 4.1 hingga 4.5.



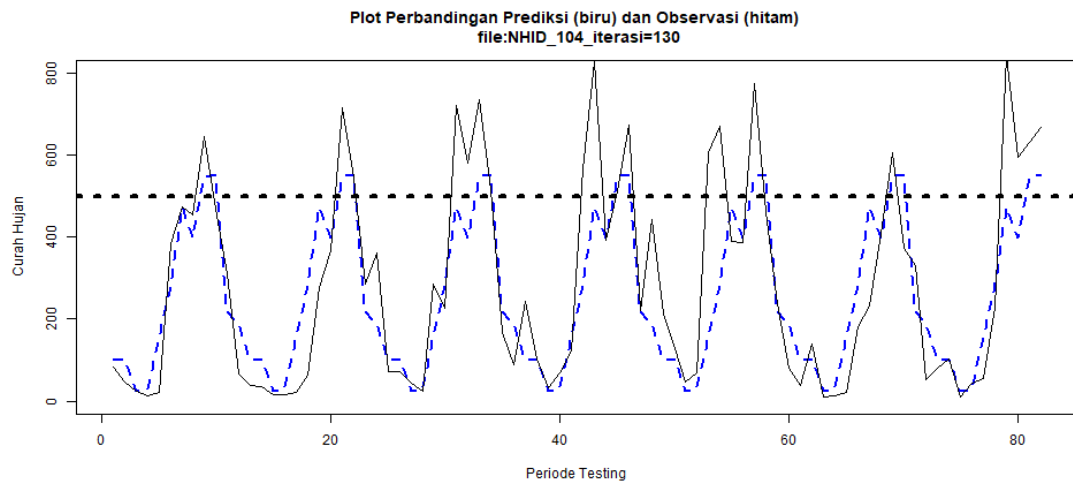
Gambar 4.1 Grafik Perbandingan Hasil Prediksi Arsitektur MLP 1 *Hidden Layer* (1 *neuron*) dengan Data CHIRPS



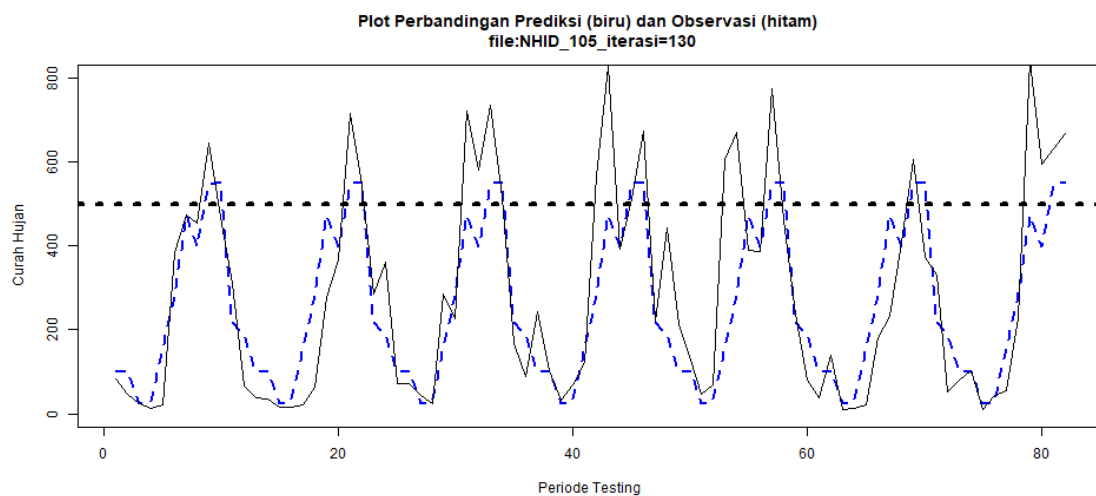
Gambar 4.2 Grafik Perbandingan Hasil Prediksi Arsitektur MLP 1 *Hidden Layer* (2 *neuron*) dengan Data CHIRPS



Gambar 4.3 Grafik Perbandingan Hasil Prediksi Arsitektur MLP 1 *Hidden Layer* (3 *neuron*) dengan Data CHIRPS



Gambar 4.4 Grafik Perbandingan Hasil Prediksi Arsitektur MLP 1 *Hidden Layer* (4 *neuron*) dengan Data CHIRPS



Gambar 4.5 Grafik Perbandingan Hasil Prediksi Arsitektur MLP 1 *Hidden Layer* (5 *neuron*) dengan Data CHIRPS

Grafik-grafik tersebut menunjukkan hasil prediksi mengikuti tren utama curah hujan dari CHIRPS. Secara umum, pola kenaikan dan penurunan curah hujan dapat ditangkap dengan baik oleh model, sehingga menjelaskan kemampuan MLP 1 *hidden layer* dalam menangani hubungan non-linier antara indeks iklim global dengan curah

hujan lokal. Hal ini mengindikasikan bahwa model memiliki akurasi yang memadai, meskipun masih terdapat beberapa deviasi di beberapa bulan dengan curah hujan yang sangat tinggi. Secara keseluruhan, penggunaan variasi 5 *neuron* pada *hidden layer* menghasilkan keseimbangan antara kompleksitas jaringan dan akurasi prediksi.

4.1.1 Verifikasi Parameter Kontigensi BMKG Skenario MLP 1 *Hidden Layer*

Verifikasi performa model diverifikasi menggunakan parameter kontigensi BMKG, menggunakan variabel akurasi, *false alarm*, dan *miss classification*. Putra dan Rani (2020) menjelaskan bahwa pentingnya melakukan verifikasi *output* menggunakan komponen variabel tabel kontingensi BMKG untuk menilai seberapa akurat model dalam memprediksi kejadian aktual. Verifikasi ini tidak hanya memungkinkan identifikasi kelemahan model dalam memberikan peringatan yang tepat, tetapi juga menjadi dasar untuk melakukan perbaikan dan optimasi sistem prediksi curah hujan ekstrem yang berhubungan dengan potensi bencana hidrometeorologi. Tabel 4.1 menyajikan hasil verifikasi model MLP dengan 1 *hidden layer* yang terdiri atas 30 kali kali percobaan menggunakan variasi 5 *neuron* sebagai berikut:

Tabel 4.1 Hasil Verifikasi Skenario MLP 1 Menggunakan Kontigensi BMKG

Keterangan	Error	Step	Akurasi	False Alarm	Miss
SINGLE 1	3.888397279	194	87.5	23.7	33.3
SINGLE 2	3.857234947	334	87.0	24.1	33.3
SINGLE 3	3.886538201	120.8	87.5	23.1	34.2
SINGLE 4	3.76252277	805	86.36	24.66	38.94
SINGLE 5	3.888840115	113.8	85.88	26.02	39.98

Dari tabel tersebut, dapat diketahui nilai dari hasil verifikasi berdasarkan tabel kontigensi bahwa akurasi yang dihasilkan model MLP 1 *hidden layer* berada pada kisaran 85.88%–87.5%. Percobaan pada arsitektur 1 *hidden layer* 1 *neuron* (*SINGLE_1*) menunjukkan akurasi tertinggi sebesar 87.5%. *False alarm* terendah didapat pada arsitektur 1 *hidden layer* 3 *neuron* (*SINGLE_3*) yaitu 23.1%, mengindikasikan bahwa model tidak terlalu sering memberikan peringatan hujan tinggi ketika kondisi sebenarnya tidak terjadi, sehingga masih dapat diterima dalam konteks prediksi curah hujan. *Miss classification* pada arsitektur arsitektur 1 *hidden layer* 1 *neuron* (*SINGLE_1*) dan 1 *hidden layer* 2 *neuron* (*SINGLE_2*) yaitu 33.3%. Hal ini mengindikasikan bahwa meskipun akurasi secara umum cukup baik, masih terdapat kelemahan model dalam mendeteksi kejadian curah hujan tinggi pada beberapa periode.

Secara keseluruhan, hasil verifikasi menunjukkan bahwa model MLP 1 *hidden layer* menggunakan variasi 5 *neuron* mampu memberikan akurasi yang konsisten di atas 85%, dengan tingkat *false alarm* yang relatif rendah. Namun, nilai *miss classification* yang masih cukup tinggi, hal ini menegaskan perlunya eksplorasi arsitektur yang lebih kompleks menggunakan 2 sampai 3 *hidden layer* meningkatkan keandalan deteksi hujan tinggi.

4.1.2 Analisis Perbandingan Performa *Training* dan *Testing* Skenario MLP 1 *Hidden Layer*

Untuk menilai kemampuan generalisasi model MLP 1 *hidden layer*, dilakukan perbandingan hasil pelatihan (*training*) dan pengujian (*testing*) menggunakan

parameter akurasi, *false alarm*, dan *miss classification* sebagaimana ditunjukkan pada Tabel 4.2.

Tabel 4.2 Performa Model MLP 1 *Hidden Layer* pada Data *Training* dan *Testing*

Keterangan	Akurasi Train	False Alarm Train	Miss Train	Akurasi Test	False Alarm Test	Miss Test
SINGLE 1	87.5	43.42	29.8	87.5	23.7	33.3
SINGLE 2	87.56	43.38	28.56	87.0	24.1	33.3
SINGLE 3	87.74	42.86	28.16	87.5	23.1	34.2
SINGLE 4	87.8	42.7	27.76	86.36	24.66	38.94
SINGLE 5	87.62	42.98	30.22	85.88	26.02	39.98

Berdasarkan hasil pada Tabel 4.2, akurasi *training* model MLP dengan 1 *hidden layer* berada pada kisaran 87.50%–87.80%, dengan rata-rata sekitar 87.64%. Nilai ini menunjukkan bahwa model berhasil mengenali pola hubungan antara indeks iklim global dan curah hujan bulanan dengan baik pada data *training*. Sementara itu, akurasi *testing* berada pada kisaran 85.88%–87.80%, dengan selisih terhadap data *training* <1%. Selisih yang kecil ini menunjukkan bahwa model memiliki kemampuan generalisasi yang baik dan tidak mengalami *overfitting*, karena performa pada data *testing* relatif stabil terhadap hasil pelatihan. Dari sisi *false alarm*, nilai pada data *training* berkisar 42.70%–43.42%, sedangkan pada data *testing* lebih rendah sekitar 23.10%–26.02%. Perbedaan ini menunjukkan bahwa model lebih konservatif dalam memberikan prediksi kejadian hujan tinggi pada data *testing*. Dengan kata lain, model cenderung menekan kesalahan tipe saat dihadapkan pada data baru.

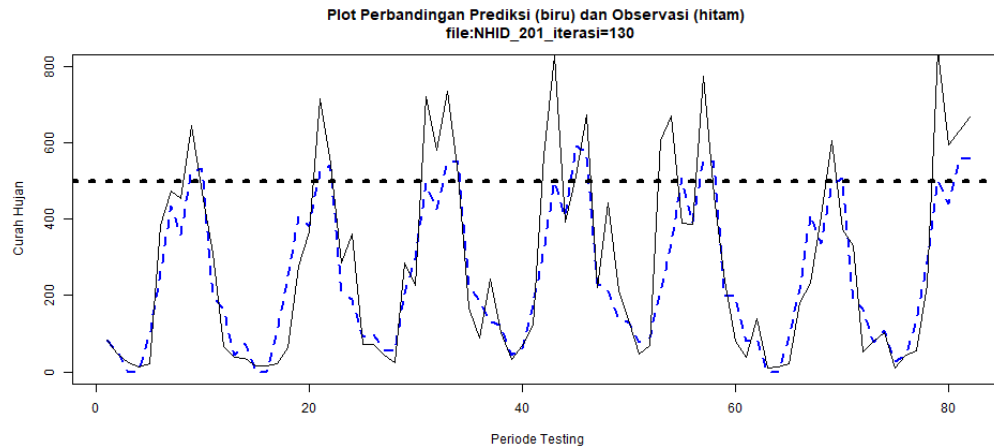
Namun, nilai *miss classification* pada data *testing* berada pada kisaran 33.30%–39.98%, sedikit lebih tinggi dibandingkan data *training* (27.76%–30.22%). Hal ini

mengindikasikan bahwa masih terdapat sebagian kecil kejadian hujan tinggi yang belum berhasil diidentifikasi dengan tepat oleh model, kemungkinan karena keterbatasan representasi variasi curah hujan ekstrem dalam data *training*. Secara keseluruhan, model MLP dengan 1 *hidden layer* menunjukkan performa yang stabil dan seimbang antara hasil *training* dan *testing*, dengan tidak adanya indikasi *overfitting* yang signifikan. Akurasi yang konsisten dan penurunan *training* pada data *testing* menandakan bahwa model mampu beradaptasi dengan baik terhadap data baru. Meskipun demikian, peningkatan *miss classification* menunjukkan bahwa arsitektur jaringan yang lebih kompleks, seperti MLP 2 *hidden layer*, dapat dieksplorasi untuk memperbaiki kemampuan model dalam mendeteksi curah hujan ekstrem secara lebih akurat.

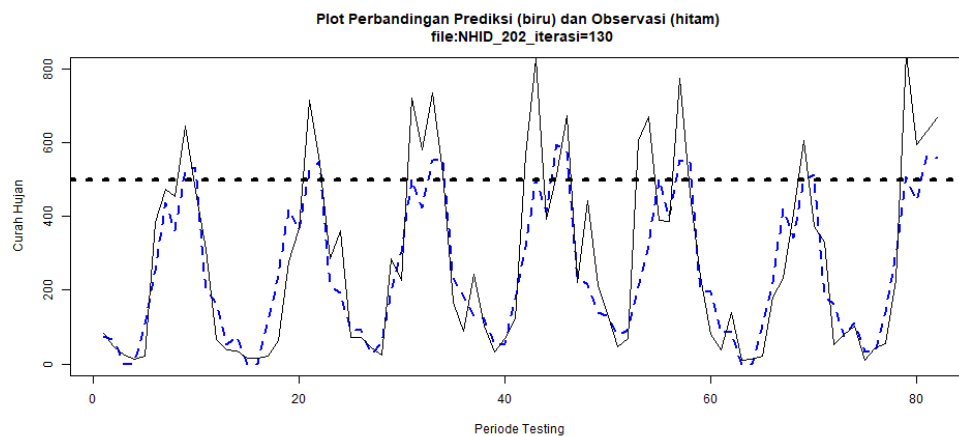
4.2 Prediksi Curah Hujan Menggunakan Arsitektur MLP 2 Hidden Layer

Pada skenario ini, pengujian prediksi curah hujan menggunakan arsitektur MLP 2 *hidden layer*. Perbedaan mendasar dengan skenario sebelumnya yaitu penambahan *hidden layer* diharapkan mampu meningkatkan kapasitas representasi jaringan dalam mempelajari pola kompleks antara variabel indeks iklim global dengan data curah hujan CHIRPS. Arsitektur MLP 2 *hidden layer* memungkinkan model membentuk representasi fitur yang lebih mendalam. *Hidden layer* pertama bertugas mengekstraksi pola nonlinier awal dari data *input*, sementara *hidden layer* kedua melakukan transformasi lanjutan sehingga menghasilkan prediksi yang diharapkan lebih baik dibandingkan arsitektur sebelumnya. Grafik perbandingan hasil prediksi curah hujan

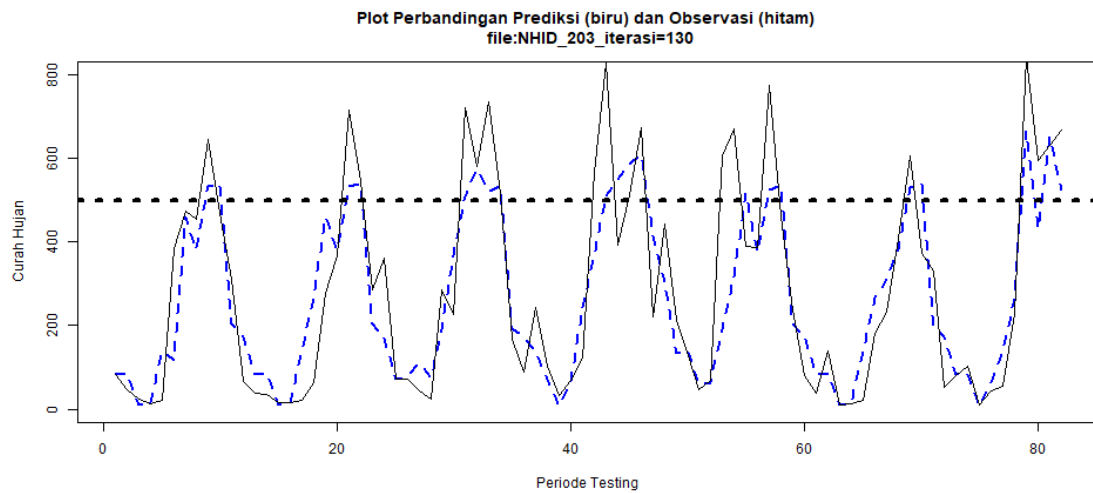
di Pacet dengan data curah hujan CHIRPS menggunakan arsitektur model MLP 1 *hidden layer* ditampilkan pada Gambar 4.6 sampai 4.10



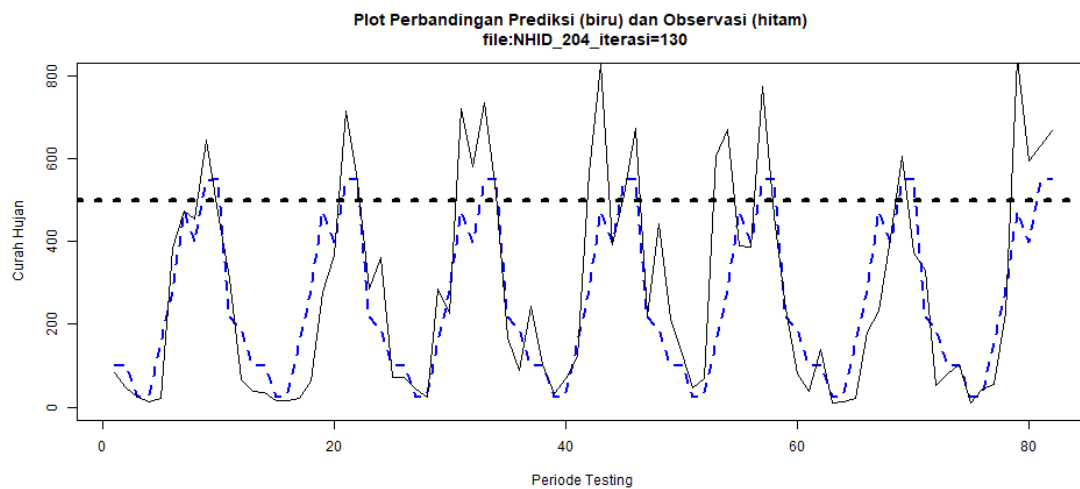
Gambar 4.6 Grafik Perbandingan Hasil Prediksi Arsitektur MLP 2 *Hidden Layer* (*neuron 1,1*) dengan Data CHIRPS



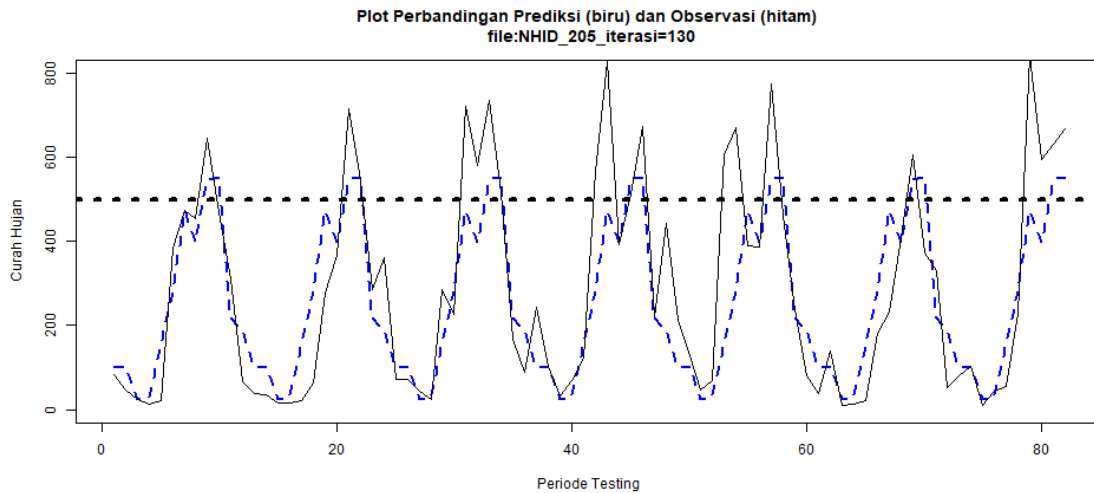
Gambar 4.7 Grafik Perbandingan Hasil Prediksi Arsitektur MLP 2 *Hidden Layer* (*neuron 2,2*) dengan Data CHIRPS



Gambar 4.8 Grafik Perbandingan Hasil Prediksi Arsitektur MLP 2 *Hidden Layer*
(*neuron 3,3*) dengan Data CHIRPS



Gambar 4.9 Grafik Perbandingan Hasil Prediksi Arsitektur MLP 2 *Hidden Layer*
(*neuron 4,4*) dengan Data CHIRPS



Gambar 4.10 Grafik Perbandingan Hasil Prediksi Arsitektur MLP 2 *Hidden Layer* (*neuron 5,5*) dengan Data CHIRPS

Secara umum, grafik-grafik diatas menunjukkan bahwa arsitektur MLP 2 *hidden layer* memberikan variasi kinerja yang lebih stabil dibandingkan model dengan satu *hidden layer*. Beberapa arsitektur *neuron* tertentu mampu menghasilkan nilai *error* yang lebih rendah dan akurasi yang lebih tinggi, meskipun arsitektur lain menurunkan performa akibat kompleksitas berlebih. Fenomena ini menunjukkan adanya titik optimal yang perlu dipertimbangkan dalam pemilihan arsitektur, agar diperoleh keseimbangan antara ketepatan prediksi dan efisiensi komputasi. Oleh karena itu, analisis hasil prediksi dengan arsitektur MLP 2 *hidden layer* ini menjadi penting sebagai pembanding terhadap model sebelumnya, sekaligus dasar dalam pengembangan skenario arsitektur yang lebih kompleks, termasuk uji coba pada 3 *hidden layer* maupun *ensemble* model.

4.2.1 Verifikasi Parameter Kontigensi BMKG dengan Skenario MLP 2 *Hidden Layer*

Pada penelitian ini, verifikasi kontigensi diterapkan pada skenario MLP 2 *hidden layer* menggunakan 5 variasi *neuron* yang berbeda. Setiap arsitektur diuji untuk melihat sejauh mana kompleksitas arsitektur jaringan memengaruhi performa model berdasarkan standar BMKG. Hasil verifikasi ditampilkan dalam Tabel 4.3 sebagai berikut:

Tabel 4.3 Hasil Verifikasi Skenario MLP 2 Menggunakan Kontigensi BMKG

Keterangan	Error	Step	Akurasi	False Alarm	Miss
DUAL 1.1	3.899129287	170.8	88.4	22.8	29.3
DUAL 2.2	3.937308086	174	87.4	24.1	33.3
DUAL 3.3	3.859002738	216	87.2	23.9	34.6
DUAL 4.4	3.893530152	82	86.36	25.38	37.88
DUAL 5.5	3.381936645	1930	85.86	25.38	41.04

Hasil verifikasi menggunakan parameter kontigensi BMKG pada arsitektur MLP 2 *hidden layer* memberikan hasil yang lebih jelas mengenai performa tiap variasi *neuron*. Akurasi yang dihasilkan model MLP 2 *hidden layer* berada pada kisaran 85.86%–88.4%. Percobaan pada arsitektur 2 *hidden layer neuron* 1.1 (DUAL_1.1) menunjukkan akurasi tertinggi sebesar 88.4%. Hal ini menegaskan bahwa penambahan jumlah *neuron* tidak selalu meningkatkan performa model. *False alarm* terendah juga didapat pada arsitektur 2 *hidden layer neuron* 1.1 (DUAL_1.1) yaitu 22.8%, mengindikasikan bahwa model tidak terlalu sering memberikan peringatan hujan tinggi ketika kondisi sebenarnya tidak terjadi, sehingga masih dapat diterima dalam konteks prediksi curah hujan. Sedangkan nilai tertinggi terjadi pada 2 *hidden layer neuron* 4.4 (DUAL_4.4) dan 2 *hidden layer neuron* 5.5 (DUAL_5.5) yaitu 25.38%. *Miss*

classification terendah ada pada arsitektur 2 *hidden layer neuron* 1.1 (DUAL_1.1) yaitu 29.3%. Sebaliknya, arsitektur DUAL 5.5 menghasilkan nilai *miss* tertinggi yaitu 41.04%, menunjukkan penurunan kemampuan model dalam menangkap kejadian hujan aktual ketika kompleksitas jaringan meningkat. Secara umum, hasil pengujian MLP 2 *hidden layer* memperlihatkan peningkatan kinerja dibandingkan arsitektur MLP 1 *hidden layer*. Temuan ini menegaskan bahwa arsitektur MLP 2 *hidden layer*, khususnya arsitektur 2 *hidden layer neuron* 1.1 (DUAL_1.1), mampu memberikan keseimbangan paling optimal antara akurasi, *false alarm*, dan *miss classification*, sehingga dapat dipertimbangkan sebagai arsitektur terbaik pada skenario ini.

4.2.2 Analisis Perbandingan Performa *Training* dan *Testing* Skenario MLP 2 *Hidden Layer*

Untuk menilai kemampuan generalisasi model MLP 2 *hidden layer*, dilakukan perbandingan hasil *training* dan *testing* terhadap parameter akurasi, *false alarm*, dan *miss classification*. Hasilnya ditunjukkan pada Tabel 4.4 berikut.

Tabel 4.4 Performa Model MLP 2 *Hidden Layer* pada Data *Training* dan *Testing*

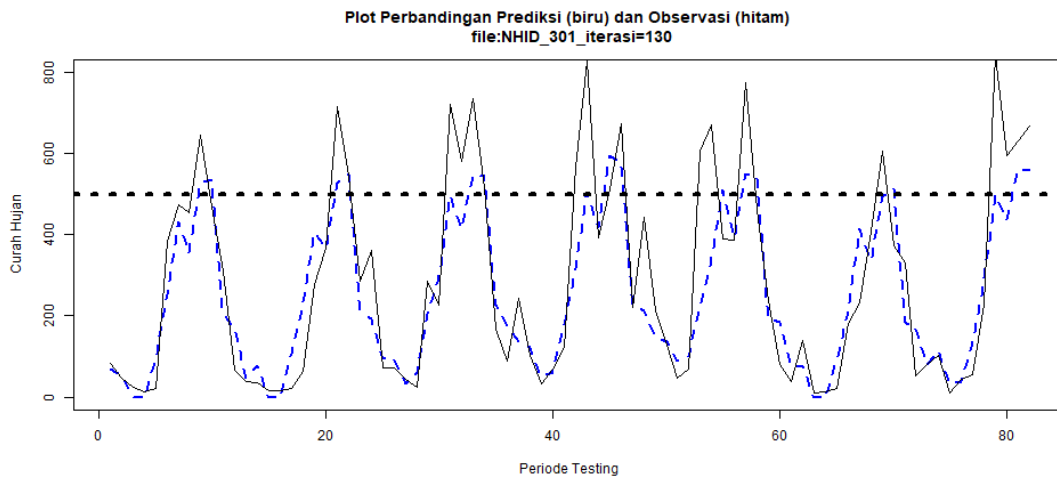
Keterangan	Akurasi Train	False Alarm Train	Miss Train	Akurasi Test	False Alarm Test	Miss Test
DUAL_1.1	88.22	41.66	26.12	88.4	22.8	29.3
DUAL_2.2	87.98	42.2	27.34	87.4	24.1	33.3
DUAL_3.3	87.86	42.64	26.92	87.2	23.9	34.6
DUAL_4.4	88.22	41.74	25.3	86.36	25.38	37.88
DUAL_5.5	87.68	43	28.58	85.86	25.38	41.04

Berdasarkan hasil pada Tabel 4.4, akurasi *training* model MLP dengan 2 *hidden layer* berada pada kisaran 87.68%–88.22%, dengan rata-rata sekitar 88.0%. Nilai ini menunjukkan bahwa model mampu mempelajari pola hubungan antara indeks iklim

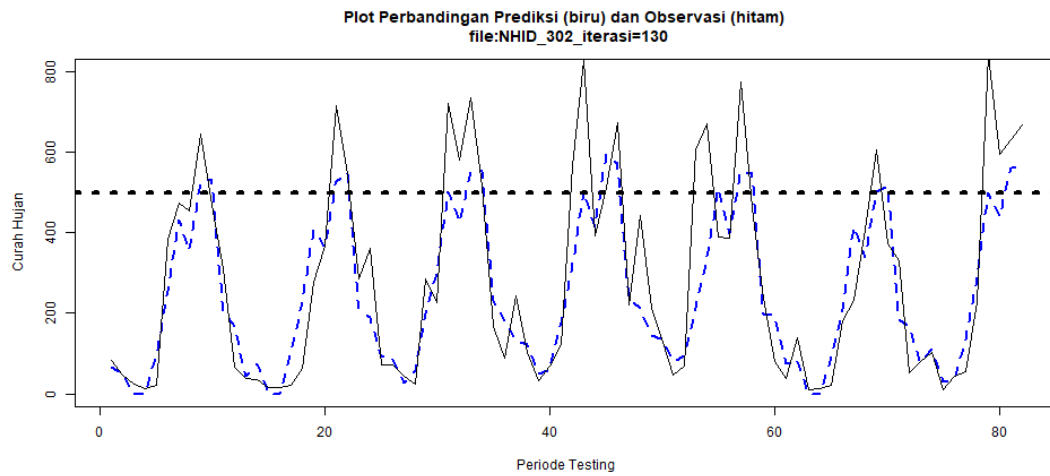
global dan curah hujan bulanan dengan baik pada data *training*. Sementara itu, akurasi *testing* berada pada kisaran 85.86%–88.40%, dengan selisih rata-rata sekitar 1.6% terhadap akurasi *training*. Selisih yang relatif kecil ini menunjukkan bahwa model memiliki kemampuan generalisasi yang baik tanpa indikasi *overfitting*, karena performa pada data baru tidak mengalami penurunan signifikan. Dari sisi *false alarm*, nilai pada data *training* berada di kisaran 41.66%–43.00%, sedangkan pada data *testing* menurun menjadi 22.80%–25.38%. Penurunan yang cukup besar ini menunjukkan bahwa model menjadi lebih selektif dalam memprediksi kejadian curah hujan tinggi pada data *testing*, yang berarti sensitivitas terhadap *noise* dan variasi ekstrem berkurang. Namun, dari sisi *miss classification*, nilai pada data *training* berkisar 25.30%–28.58%, sedangkan pada data *testing* meningkat menjadi 29.30%–41.04%. Hal ini menunjukkan bahwa beberapa kejadian hujan ekstrem pada data *testing* masih belum sepenuhnya dapat dikenali oleh model, yang dapat diakibatkan oleh perbedaan pola atau fluktuasi iklim yang tidak terwakili secara penuh pada data *training*. Secara keseluruhan, arsitektur MLP 2 *hidden layer* menunjukkan performa yang seimbang dan stabil antara kemampuan belajar (*training*) dan kemampuan generalisasi (*testing*). Model mampu menjaga tingkat akurasi tinggi dengan penurunan minimal pada data *testing*, serta memperlihatkan kemampuan adaptif terhadap variasi data baru. Dengan performa yang konsisten ini, MLP 2 *hidden layer* dapat dikategorikan sebagai arsitektur optimal dan handal, serta dapat dijadikan *baseline* kuat sebelum pengujian model yang lebih kompleks seperti MLP 3 *hidden layer* dan *ensemble* model pada subbab berikutnya.

4.3 Prediksi Curah Hujan Menggunakan MLP 3 *Hidden Layer*

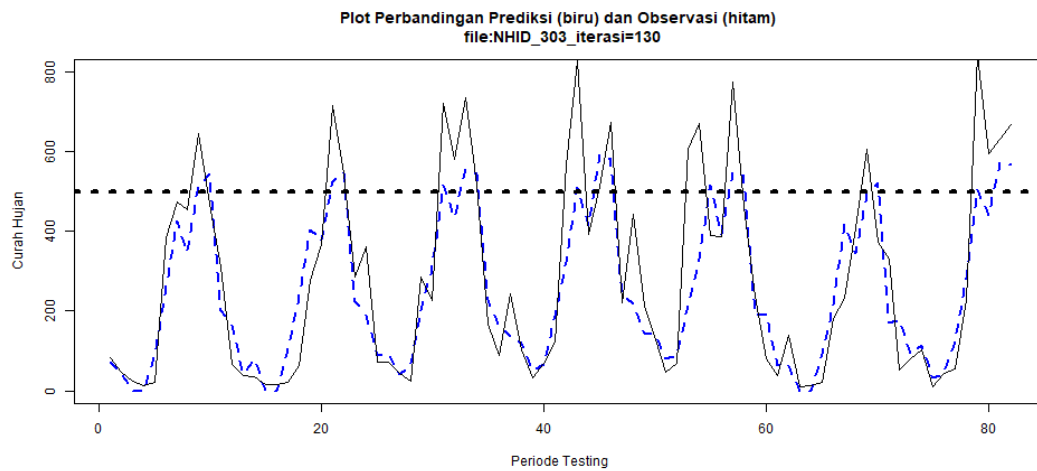
Arsitektur MLP 3 *hidden layer* digunakan untuk mengverifikasi apakah penambahan *hidden layer* mampu meningkatkan ketepatan model dalam memprediksi curah hujan bulanan di Pacet. Peningkatan jumlah *hidden layer* pada dasarnya ditujukan untuk menangkap hubungan nonlinier yang lebih kompleks antara indeks iklim dengan curah hujan CHIRPS. Dengan 3 *hidden layer*, model diharapkan dapat mengekstraksi pola yang lebih dalam dan memperkaya representasi fitur, sehingga mampu menghasilkan prediksi yang lebih mendekati data curah hujan dari CHIRPS. Pada tahap ini, dilakukan serangkaian pengujian dengan variasi 5 *neuron* untuk memperoleh nilai *error*, tingkat akurasi, serta pola prediksi yang lebih baik. Grafik perbandingan hasil prediksi curah hujan di Pacet dengan data curah hujan CHIRPS menggunakan arsitektur model MLP 3 *hidden layer* ditampilkan pada gambar 4.11 hingga 4.15.



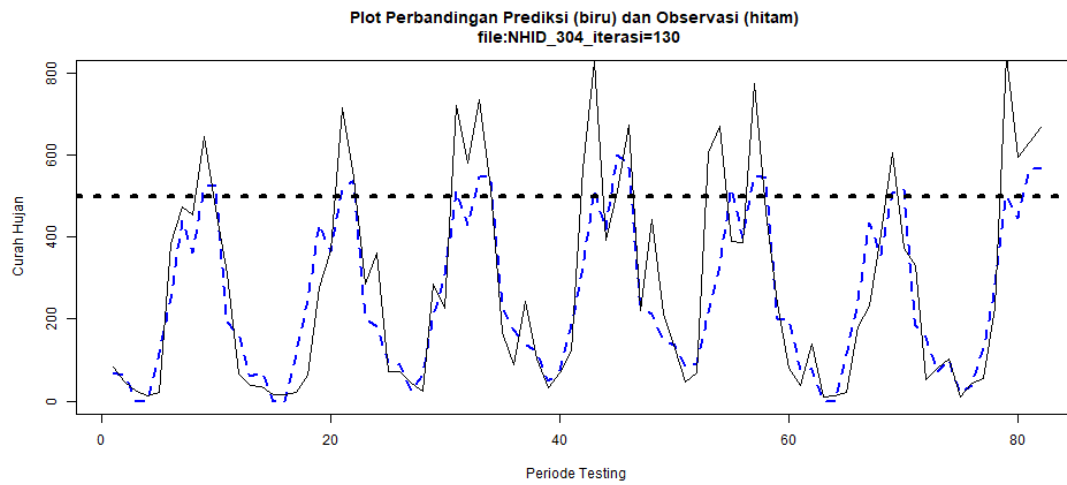
Gambar 4.11 Grafik Perbandingan Hasil Prediksi Arsitektur MLP 3 *Hidden Layer* (*neuron 1.1.1*) dengan Data CHIRPS



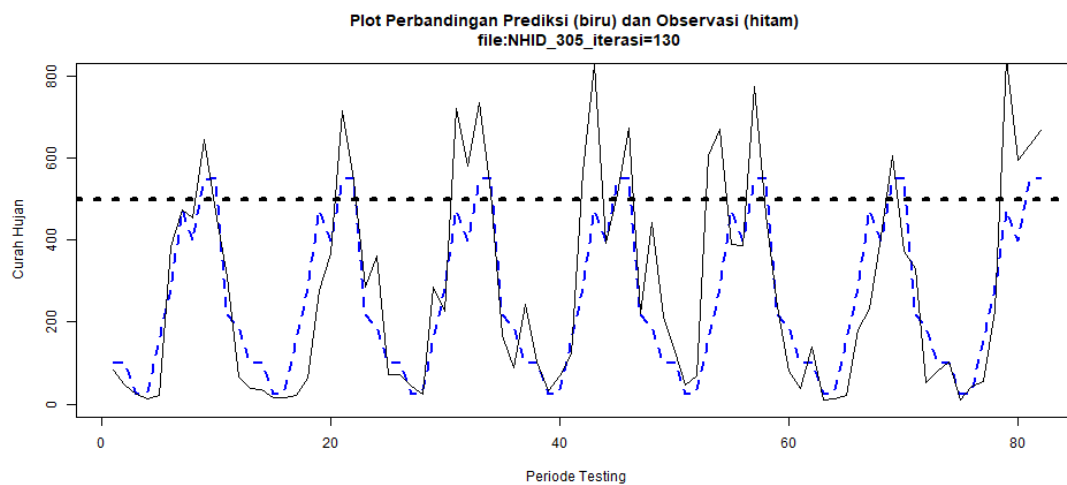
Gambar 4.12 Grafik Perbandingan Hasil Prediksi Arsitektur MLP 3 *Hidden Layer* (*neuron 2.2.2*) dengan Data CHIRPS



Gambar 4.13 Grafik Perbandingan Hasil Prediksi Arsitektur MLP 3 *Hidden Layer* (*neuron 3.3.3*) dengan Data CHIRPS



Gambar 4.14 Grafik Perbandingan Hasil Prediksi Arsitektur MLP 3 *Hidden Layer* (*neuron 4.4.4*) dengan Data CHIRPS



Gambar 4.15 Grafik Perbandingan Hasil Prediksi Arsitektur MLP 3 *Hidden Layer* (*neuron 5.5.5*) dengan Data CHIRPS

Dari gambar-gambar diatas, terlihat bahwa penambahan *hidden layer* ketiga tidak selalu menghasilkan peningkatan kinerja secara signifikan. Beberapa arsitektur *neuron* mampu memperlihatkan pola prediksi yang lebih halus dan mendekati data aktual, sementara arsitektur lain justru mengalami penurunan performa akibat

kompleksitas berlebih. Oleh karena itu, pemilihan jumlah *neuron* yang tepat pada masing-masing *hidden layer* menjadi faktor krusial dalam menjaga keseimbangan antara ketepatan prediksi dan efisiensi model.

4.3.1 Verifikasi Parameter Kontigensi BMKG Pada Skenario MLP 3 *Hidden Layer*

Verifikasi ini bertujuan untuk menilai sejauh mana model dengan berbagai arsitektur *neuron* mampu merepresentasikan kejadian curah hujan aktual dalam kategori yang sesuai. Hasil verifikasi ditampilkan dalam Tabel 4.5 sebagai berikut:

Tabel 4.5 Hasil Verifikasi Skenario MLP 2 Menggunakan Kontigensi BMKG

Keterangan	Error	Step	Akurasi	False Alarm	Miss
TRIPLE_1.1.1	4.139898993	103.6	87.6	21.8	35.6
TRIPLE_2.2.2	3.877182965	131.2	87.8	22.8	32.5
TRIPLE_3.3.3	3.876604432	133.2	87.9	22.4	31.9
TRIPLE_4.4.4	3.709861122	807.8	85.86	27.18	40
TRIPLE_5.5.5	3.568735746	3057.4	86.1	25.76	38.94

Hasil pengujian menunjukkan bahwa akurasi model berada pada kisaran 85.86%-87.9%. Arsitektur 3 *hidden layer neuron* 2.2.2 (TRIPLE_2.2.2) memberikan performa terbaik dengan akurasi tertinggi sebesar 87.8%. Sementara itu, akurasi terendah ditunjukkan oleh arsitektur 3 *hidden layer neuron* 4.4.4 (TRIPLE_4.4.4) dengan nilai 85.86%. Hal ini menandakan bahwa model dengan jumlah *neuron* 2,2 mampu memberikan hasil prediksi yang lebih stabil dan akurat dibandingkan arsitektur dengan *neuron* terlalu sedikit ataupun terlalu banyak. Dari sisi *false alarm*, nilai yang diperoleh berkisar antara 21.8%-27.18%. Arsitektur 3 *hidden layer neuron* 2.2.2

(TRIPLE_2.2.2) berhasil menekan *false alarm* hingga 22.8%, yang merupakan nilai terendah di antara seluruh arsitektur. Sebaliknya, arsitektur 3 *hidden layer neuron* 4.4.4 (TRIPLE_4.4.4) memperlihatkan nilai *false alarm* tertinggi yaitu 27.18%, menunjukkan adanya kecenderungan model untuk memberikan peringatan hujan tinggi pada kondisi yang sebenarnya tidak terjadi. Untuk parameter *miss classification*, kisarannya cukup lebar, yaitu antara 28.4%-40%. Arsitektur 3 *hidden layer neuron* 2.2.2 (TRIPLE_2.2.2) kembali menunjukkan kinerja terbaik dengan nilai *miss classification* terendah, yaitu 32.5%, yang mengindikasikan kemampuan model ini dalam menangkap kejadian hujan tinggi dengan lebih baik. Secara keseluruhan, hasil verifikasi ini memperlihatkan bahwa arsitektur 3 *hidden layer neuron* 2.2.2 (TRIPLE_2.2.2) merupakan pilihan yang paling optimal dalam skenario ini. Arsitektur ini mampu menyeimbangkan akurasi tinggi dengan tingkat *false alarm* dan *miss classification* yang relatif rendah. Sementara itu, penggunaan jumlah *neuron* yang terlalu besar menimbulkan kompleksitas berlebih yang mengurangi stabilitas dan keandalan prediksi.

4.3.2 Analisis Perbandingan Performa *Training* dan *Testing* Skenario MLP 3 *Hidden Layer*

Analisis perbandingan antara hasil training dan testing dilakukan untuk menilai stabilitas model dan mendeteksi kemungkinan adanya *overfitting* atau *underfitting*. Hasil *training* dan testing model mlp 3 *hidden layer* ditampilkan pada Tabel 4.6 berikut:

Tabel 4.6. Hasil *Training* dan *Testing* Model MLP 3 *Hidden Layer*

Keterangan	Jumlah Neuron	Akurasi Train	False Alarm Train	Miss Train	Akurasi Test	False Alarm Test	Miss Test
TRIPLE 1.1.1	1.1.1	88.28	41.48	26.54	87.6	21.8	35.6
TRIPLE 2.2.2	2.2.2	88.52	41.06	23.24	87.8	22.8	32.5
TRIPLE 3.3.3	3.3.3	88.16	41.82	26.52	87.9	22.4	31.9
TRIPLE 4.4.4	4.4.4	88.34	41.1	26.54	85.86	27.18	40
TRIPLE 5.5.5	5.5.5	88.66	40.4	24.48	86.1	25.76	38.94

Hasil pada Tabel 4.6 menunjukkan bahwa akurasi *training* model MLP dengan 3 *hidden layer* berada pada kisaran 88.16%–88.66%, dengan rata-rata sebesar 88.39%. Nilai ini menandakan bahwa model mampu mempelajari hubungan non-linier antara variabel *input* (indeks iklim global) dan *output* (anomali curah hujan) secara efektif. Sementara itu, akurasi *testing* berkisar antara 85.86%–87.90%. Selisih yang kecil ini menunjukkan bahwa model memiliki kemampuan generalisasi yang baik dan tidak mengalami *overfitting*, karena performa pada data baru (*testing*) tetap stabil. Dari sisi *false alarm*, nilai pada data *training* berkisar antara 40.40%–41.82%, sedikit lebih tinggi dibandingkan data *testing* yang berada pada kisaran 21.80%–27.18%. Hal ini menunjukkan bahwa pada tahap pelatihan, model lebih sering mengklasifikasikan curah hujan ekstrem secara positif (lebih sensitif terhadap kejadian hujan tinggi), sedangkan pada tahap *testing* model menjadi lebih konservatif dalam memberikan prediksi tersebut. Untuk *miss classification*, nilai pada data *training* berada di rentang 23.24%–26.54%, sementara pada data *testing* meningkat menjadi 31.90%–40.00%. Kenaikan nilai *miss* ini menunjukkan bahwa beberapa kejadian hujan tinggi pada data *testing* belum sepenuhnya terdeteksi oleh model, yang dapat disebabkan oleh variasi

data atau ketidaksesuaian pola antara data *training* dan *testing*. Secara keseluruhan, model MLP dengan 3 *hidden layer* menunjukkan performa yang stabil dan seimbang antara kemampuan belajar dan kemampuan generalisasi. Pola peningkatan jumlah *neuron* tidak memberikan perubahan signifikan terhadap akurasi, namun sedikit mempengaruhi tingkat *false alarm* dan *miss classification*. Dengan demikian, arsitektur 3 *hidden layer* mampu merepresentasikan kompleksitas hubungan antar indeks iklim global terhadap curah hujan bulanan di Pacet secara cukup baik tanpa menyebabkan *overfitting*.

4.4 Seleksi Model

Proses seleksi model pada penelitian ini bertujuan untuk menentukan arsitektur MLP yang paling optimal dalam memprediksi curah hujan bulanan di Pacet. Berikut seleksi model yang dilakukan untuk mengetahui skenario arsitektur terbaik.

```

1 library(dplyr)
2
3 FU="E:\\DEFINTA_THESIS\\TES_PRA_SEMPRO"
4 rekap <- read.csv(paste(FU,"\\OUTPUT\\REKAP_VERIF.csv",sep=""))
5
6 # Pisahkan skenario berdasarkan nama model
7 rekap$kategori <- ifelse(grepl("SINGLE", rekap$X), "MLP_1",
8                           ifelse(grepl("DUAL", rekap$X), "MLP_2",
9                                   ifelse(grepl("TRIPLE", rekap$X), "MLP_3", NA)))
10
11 # Buat fungsi perhitungan skor berdasarkan prioritas verifikasi
12 hitung_skor <- function(df){
13   df$skor <- (df$Akurasi_Test * 2) - df$FalseAlarm_Test - df$Miss_Test
14   return(df)
15 }
16
17 rekap <- rekap %>% group_by(kategori) %>% do(hitung_skor(.))
18
19 # Pilih model terbaik per kategori
20 best_mlp1 <- rekap %>% filter(kategori == "MLP_1") %>% arrange(desc(skor)) %>% slice(1)
21 best_mlp2 <- rekap %>% filter(kategori == "MLP_2") %>% arrange(desc(skor)) %>% slice(1)
22 best_mlp3 <- rekap %>% filter(kategori == "MLP_3") %>% arrange(desc(skor)) %>% slice(1)
23
24 # Gabungkan
25 hasil_best <- rbind(best_mlp1, best_mlp2, best_mlp3)
26
27 # Simpan hasil rekomendasi
28 write.csv(hasil_best, paste(FU,"\\OUTPUT\\HASIL_MODEL_TERBAIK.csv",sep=""), row.names=FALSE)
29
30 print("=== HASIL SELEKSI MODEL TERBAIK ===")
31 print(hasil_best)

```

Gambar 4.16 Penentuan Rekomendasi Model.

Evaluasi dilakukan pada 3 skenario arsitektur, yaitu MLP 1, 2, dan 3 *hidden layer*, masing-masing terdiri dari beberapa arsitektur *neuron* yang diuji menggunakan verifikasi kontingensi BMKG. Pendekatan ini memungkinkan penilaian menyeluruh terhadap pengaruh kompleksitas jaringan terhadap akurasi prediksi, *false alarm*, serta *miss classification*. Pada tahap awal pengujian, arsitektur MLP dengan 1 *hidden layer* digunakan sebagai skenario dasar. Arsitektur ini memiliki kompleksitas rendah dan stabil, namun menunjukkan keterbatasan dalam mengenali pola non-linier yang lebih dalam antara indeks iklim global dan curah hujan. Untuk meningkatkan kapasitas representasi model, arsitektur 2 *hidden layer* kemudian diuji dan menunjukkan peningkatan performa, terutama dalam mempertajam kemampuan model dalam mengurangi *miss classification*. Pada tahap berikutnya, arsitektur 3 *hidden layer* diuji sebagai konfigurasi paling kompleks, namun hasilnya tidak selalu lebih baik karena pada beberapa konfigurasi terjadi indikasi *overfitting* sehingga akurasi tidak meningkat secara signifikan. Seleksi ini menghasilkan rekomendasi model yang ditampilkan pada Tabel 4.7 yang dalam menentukan arsitektur optimal untuk prediksi curah hujan.

Tabel 4.7 Rekomendasi Arsitektur Tiap Skenario

Variabel Verifikasi	Nilai	MLP 1	Nilai	MLP 2	Nilai	MLP 3
Akurasi	87.5	SINGLE_1 & SINGLE_3	88.4	DUAL_1.1	87.9	TRIPLE_3.3.3
<i>False Alarm</i>	23.1	SINGLE_3	22.8	DUAL_1.1	21.8	TRIPLE_1.1.1
<i>Miss</i>	33.3	SINGLE_1 & SINGLE_3	29.3	DUAL_1.1	31.9	TRIPLE_3.3.3

Tabel 4.7 menampilkan rekapitulasi model terbaik pada masing-masing kelompok arsitektur. Berdasarkan hasil tersebut dapat dilihat bahwa ketiga kelompok arsitektur pada dasarnya menunjukkan performa yang kompetitif, dengan akurasi di atas 87%. Meskipun demikian, arsitektur MLP 2 *hidden layer* dengan *neuron* 1-1 (DUAL_1.1), terbukti sebagai model dengan performa paling unggul. Model ini mencapai akurasi 88,4% yang merupakan nilai tertinggi di antara seluruh skenario pengujian, sekaligus menghasilkan nilai *false alarm* sebesar 22,8% dan *miss classification* sebesar 29,3%, keduanya merupakan nilai terendah dibandingkan skenario lainnya. Konsistensi performa tersebut menunjukkan bahwa arsitektur MLP 2 *hidden layer* dengan *neuron* 1-1 (DUAL_1.1) memiliki keseimbangan terbaik antara kemampuan mengenali kejadian hujan ekstrem dan kemampuan menjaga kestabilan prediksi, sehingga layak dinyatakan sebagai model paling optimal dalam tahap seleksi ini. Meskipun arsitektur DUAL_1.1 menunjukkan performa paling baik, penggunaan satu model MLP tunggal tetap memiliki keterbatasan, terutama karena tingginya variabilitas curah hujan dan sifat nonlinier data iklim di Pacet. Ketergantungan pada satu arsitektur dapat menyebabkan hasil prediksi tidak stabil ketika kondisi data berubah atau ketika model menghadapi sampel yang berada di luar pola mayoritas. Untuk mengatasi keterbatasan tersebut sekaligus meningkatkan reliabilitas prediksi, arsitektur MLP 2 *hidden layer* dengan *neuron* 1-1 (DUAL_1.1) dipilih sebagai *base learner* pada model *Ensemble Bagging*. Dengan memanfaatkan model terbaik hasil seleksi sebagai dasar *ensemble*, sistem prediksi akhir diharapkan mampu menghasilkan keluaran yang lebih stabil, lebih konsisten, dan lebih adaptif terhadap variasi data dibandingkan penggunaan satu model MLP saja.

BAB V

***ENSEMBLE* MLP UNTUK PREDIKSI CURAH HUJAN**

5.1 Prediksi Curah Hujan Menggunakan *Ensemble* MLP

Metode *Ensemble Bagging (bootstrap aggregating)* merupakan salah satu pendekatan yang bertujuan meningkatkan stabilitas dan reliabilitas model. Prinsip dasarnya adalah menggabungkan hasil prediksi dari beberapa model atau dari beberapa variasi data melalui proses *resampling*, sehingga mampu mengurangi variansi prediksi yang sering muncul pada model tunggal. Meskipun arsitektur MLP terbaik, yaitu DUAL_1.1 menunjukkan performa paling optimal dibandingkan arsitektur lainnya, model tunggal tetap memiliki keterbatasan. Variasi data curah hujan yang sangat fluktuatif dan sifatnya nonlinier dapat menyebabkan model tunggal menghasilkan prediksi yang berbeda jika data uji sedikit berubah. Dengan *Ensemble Bagging*, variansi tersebut dapat ditekan, sehingga hasil prediksi menjadi lebih baik dan dapat dijadikan dasar pengambilan keputusan. Data yang digunakan dalam proses *ensemble* ini adalah hasil prediksi curah hujan dari arsitektur terbaik DUAL_1.1 yang diresampling dengan *replacement* terhadap data uji dari nilai prediksi yang dihasilkan arsitektur DUAL_1.1 sebanyak 100 kali. Pada tahap ini, dari himpunan data uji awal, sampel diambil kembali secara acak dengan kemungkinan data muncul lebih dari satu kali. Data sampel hasil *resample with replacement* ditampilkan pada Tabel 5.1 sebagai berikut:

Tabel 5.1. Data Sampel Prediksi Curah Hujan Hasil *Resample with Replacement* Pada*Ensemble Bagging*

V1	V2	V3	V4	V5	V6	V7	V8	V...	V100
89.5	89.5	86	93	71.8	72.5	54.2	75.4	90.2	95.1
53.9	92	80.7	53.9	29.2	58.8	56.7	77.9	92.7	61.6
0	14.3	0.9	0	0	0	0	1.6	16.5	0
0	17.4	6.1	0	0	1.2	0	6.1	21	0
103.5	136.7	121.9	102.1	71.7	102.8	109.9	131.1	145.9	112
259	273.1	263.2	271	255.4	254.7	235.7	256.9	271.7	257.6
437.4	459.3	444.4	440.2	414	435.9	428.9	450.1	464.9	435.9
359.6	385.7	371.6	359.6	333.5	359.6	351.8	373	387.9	358.2
525.9	533	521	529.4	504	521.7	502.6	523.8	538.6	514.6
541	536.8	529.7	552.3	540.3	526.2	507.1	528.3	543.1	536.1
198.9	206.7	198.2	214.4	208.1	199.6	174.2	195.4	210.2	195.4
171.2	170.5	160.6	186.1	173.3	164.2	141.6	162.7	177.6	167
49.9	81.7	66.9	49.2	21.7	49.9	54.2	75.4	90.2	57
82.1	87.1	78.6	90.6	75.8	72.2	56.7	77.9	92.7	78.6
5.2	13.6	7.3	5.2	0	0	0	1.6	16.5	0
0	14.6	3.3	0	0	0	0	6.1	21	0
110.6	137.4	124.7	106.4	76.7	107.1	109.9	131.1	145.9	113.4
242.7	265.3	254	243.4	219.4	232.8	235.7	256.9	271.7	246.3
416.9	455.7	439.5	414.8	378.7	414	428.9	450.1	464.9	429.6

Selanjutnya, distribusi nilai prediksi tersebut, dihitung persentase kejadian hujan ekstrem (lebih dari 500 mm/bulan). Persentase ini kemudian dikonversi menjadi nilai biner (0–1) dengan menggunakan ambang batas sebesar 50%, 60%, 70%, 80%, dan 90%. Jika persentase prediksi kejadian melebihi ambang batas, maka hasil luaran diberi label 1 (kejadian hujan ekstrem), sedangkan jika kurang dari ambang batas maka diberi label 0 (tidak kejadian).

Tabel 5.2 Data Sampel Konversi Biner Prediksi Probabilitas Curah Hujan Setelah
Diresample

Observasi	OBS	P(x) PRED	90%	80%	70%	60%	50%
83.6	0	0	0	0	0	0	0
48.2	0	0	0	0	0	0	0
24.9	0	0	0	0	0	0	0
14.1	0	0	0	0	0	0	0
20.4	0	0	0	0	0	0	0
383.4	0	0	0	0	0	0	0
475	0	0	0	0	0	0	0
453.7	0	0	0	0	0	0	0
644.7	1	97	1	1	1	1	1
468	0	100	1	1	1	1	1
309.8	0	0	0	0	0	0	0
67.1	0	0	0	0	0	0	0
39.6	0	0	0	0	0	0	0
35.4	0	0	0	0	0	0	0
14.9	0	0	0	0	0	0	0

Tabel di atas menunjukkan hasil konversi biner prediksi curah hujan ekstrem menggunakan metode *Ensemble Bagging* dengan arsitektur terbaik, MLP 2 *hidden layer* dengan *neuron* 1-1 (DUAL_1.1) sebagai *base learner*. Pada tabel, kolom Observasi merepresentasikan nilai curah hujan dari CHIRPS, lalu dikonversi ke biner

dan direpresentasikan pada kolom OBS. Kolom PRED menunjukkan nilai persentase prediksi kejadian hujan ekstrem yang diperoleh dari hasil 100 kali *resampling with replacement*. Sebagai contoh, jika dari 100 sampel prediksi terdapat 97 kali model memprediksi terjadinya hujan ekstrem, maka nilai prediksi adalah 97%. Selanjutnya, kolom 90%, 80%, 70%, 60%, dan 50% merupakan hasil proses binerisasi prediksi berdasarkan ambang batas persentase probabilitas. Sebagai contoh, pada baris dengan data observasi 644.7 mm, nilai observasi melebihi 500 mm sehingga dinyatakan 1 (kejadian). Dari hasil *ensemble*, prediksi = 97, artinya 97 dari 100 prediksi menyatakan bahwa hujan ekstrem terjadi. Karena 97% lebih besar daripada semua *threshold* (90%, 80%, 70%, 60%, dan 50%), maka semua kolom 90% hingga 50% bernilai 1. Pada baris dengan observasi 468 mm, nilai aktual kurang dari 500 mm sehingga dinyatakan 0 (tidak kejadian). Namun hasil prediksi menunjukkan 100%, yang berarti semua *resampling* memprediksi terjadinya hujan ekstrem. Akibatnya, nilai pada semua *threshold* 90%–50% adalah 1, yang menunjukkan adanya *false alarm* dari model. Sementara untuk observasi lain seperti 83.6 mm, 48.2 mm, 24.9 mm, hasil prediksi dari *ensemble* menunjukkan 0%, sehingga semua nilai pada *threshold* (90%–50%) adalah 0, sesuai dengan observasi. Dengan demikian, tabel ini menjadi dasar analisis performa *ensemble* pada berbagai ambang batas, sekaligus memperlihatkan sensitivitas model terhadap perubahan *threshold* dalam mendeteksi kejadian hujan ekstrem.

5.1.1 Verifikasi Parameter Kontigensi BMKG Pada Skenario *Ensemble* MLP

Untuk menilai performa prediksi hujan ekstrem menggunakan pendekatan *ensemble* MLP, penelitian ini tetap menggunakan tabel kontingensi yang merupakan

standar verifikasi di BMKG. Tabel kontingensi memungkinkan verifikasi model secara lebih komprehensif, tidak hanya berdasarkan akurasi, tetapi juga memperhitungkan kejadian salah prediksi berupa *false alarm* dan *miss classification*. Tabel 5.1 menunjukkan hasil verifikasi pada *ensemble* menggunakan parameter kontingensi BMKG sebagai berikut:

Tabel 5.3 Sampel Hasil Verifikasi *Ensemble* MLP Menggunakan Parameter Kontingensi BMKG

<i>Akurasi</i>					<i>False Alarm</i>					<i>Miss Classification</i>				
90	80	70	60	50	90	80	70	60	50	90	80	70	60	50
1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	1	1	1	1	0	0	0	0	0
1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
1	1	1	1	1	0	0	0	0	0	0	0	0	0	0

Pada tabel tersebut, kolom akurasi menunjukkan apakah prediksi pada threshold >500mm sesuai dengan observasi. Jika prediksi sama dengan observasi, nilainya 1 (benar), jika tidak sama nilainya 0 (salah). Kolom *false alarm* bernilai 1 jika model memprediksi kejadian hujan ekstrem tetapi observasi menunjukkan tidak ada kejadian. Kondisi ini menunjukkan kesalahan model yang terlalu sering memberi peringatan. Kolom *miss classification* bernilai 1 jika model memprediksi tidak ada kejadian tetapi observasi menunjukkan hujan ekstrem. Dari tabel ini dapat dilihat bahwa proses verifikasi menggunakan kontingensi BMKG tidak hanya menghitung berapa banyak prediksi yang tepat, tetapi juga mengukur seberapa sering model

memberi peringatan palsu (*false alarm*) maupun gagal mendeteksi kejadian (*miss classification*). Analisis ini penting untuk menilai reliabilitas model dalam konteks prediksi hujan ekstrem, khususnya untuk kebutuhan peringatan dini bencana. Hasil verifikasi akhir arsitektur *ensemble* menggunakan parameter kontigensi BMKG ditampilkan pada tabel 5.4

Tabel 5.4 Hasil Verifikasi Model *Ensemble Bagging* Menggunakan Parameter Kontigensi

Variabel Verifikasi	Threshold Probability				
	90%	80%	70%	60%	50%
<i>Akurasi</i>	85.4%	85.4%	87.8%	87.8%	89.0%
<i>False alarm</i>	23.1%	23.1%	20.0%	23.5%	22.2%
<i>Miss</i>	47.4%	47.4%	36.8%	31.6%	26.3%

Tabel 5.4 menampilkan hasil verifikasi performa prediksi curah hujan ekstrem dengan menggunakan metode *Ensemble Bagging* pada model MLP arsitektur 2 *hidden layer neuron* 1-1 (DUAL_1.1). Verifikasi dilakukan berdasarkan parameter verifikasi utama yaitu akurasi, *false alarm*, dan *miss classification* pada lima skenario ambang batas probabilitas (90%, 80%, 70%, 60%, dan 50%). Dari tabel terlihat bahwa nilai akurasi tertinggi sebesar 89,0% diperoleh pada *threshold* Pa50, diikuti oleh nilai 87,8% pada 70% dan 60. Hal ini menegaskan bahwa skenario *Ensemble Bagging* memberikan peningkatan akurasi signifikan dibandingkan model MLP tunggal, yang rata-rata akurasinya berada pada kisaran 82–88%. Peningkatan ini terjadi karena proses *bagging* mampu mengurangi variansi prediksi dan menstabilkan hasil dari beberapa model dasar yang dilatih secara independen. Nilai *false alarm* terendah dicapai pada 70% (20,0%), menunjukkan bahwa pada ambang probabilitas ini, model mampu menekan

kesalahan prediksi hujan ekstrem yang tidak benar-benar terjadi. Sementara pada 90%, 80%, 60%, dan 50%, nilai *false alarm* relatif sedikit lebih tinggi (sekitar 22–23%), namun masih berada dalam batas yang dapat diterima untuk sistem prediksi cuaca berbasis probabilistik. Nilai *miss classification* menunjukkan penurunan yang konsisten seiring menurunnya ambang probabilitas. Pada 90% dan 80%, nilai *miss classification* mencapai 47,4%, yang menunjukkan bahwa model terlalu ketat dalam mendeteksi kejadian hujan ekstrem sehingga banyak kejadian aktual yang terlewat. Sebaliknya, pada 50%, nilai *miss classification* menurun drastis menjadi 26,3%, yang berarti sensitivitas model terhadap kejadian ekstrem meningkat secara signifikan.

Secara keseluruhan, skenario arsitektur 2 *hidden layer* dengan *neuron* 1.1 (DUAL_1.1) pada pendekatan *Ensemble Bagging* menunjukkan performa paling unggul dibandingkan skenario lainnya. Hal ini terlihat dari capaian semua variabel verifikasi utama yang menunjukkan tren akurasi tertinggi, *false alarm* terendah pada Pa70, dan *miss classification* terendah pada 50%. Keseimbangan antara ketepatan prediksi dan kemampuan deteksi kejadian aktual menjadi keunggulan utama arsitektur 2 *hidden layer* dengan *neuron* 1.1 (DUAL_1.1) dibandingkan model MLP tunggal.

Scher dan Messori (2020) menerapkan *Ensemble Bagging Neural Network* untuk prediksi cuaca berbasis ANN dan melaporkan penurunan *error* hingga 30% dibandingkan model tunggal. Selain itu, Safril et al. (2020) juga menunjukkan efektivitas Pendekatan *Ensemble Statistical Downscaling* di wilayah Banjarnegara, yang berhasil meningkatkan akurasi prediksi dibandingkan model tunggal ANN maupun regresi. Hasil penelitian ini memperkuat bukti bahwa metode *Ensemble*

Bagging mampu meningkatkan reliabilitas dan stabilitas prediksi curah hujan, terutama pada wilayah dengan variabilitas iklim tinggi seperti Pacet.

Dengan demikian, hasil pada Tabel 5.4 menegaskan bahwa arsitektur arsitektur 2 *hidden layer* dengan *neuron* 1.1 (DUAL_1.1) dengan pendekatan *ensemble bagging* memberikan performa terbaik secara menyeluruh, baik dari sisi akurasi, kestabilan, maupun keseimbangan antara kesalahan *false alarm* dan *miss classification*. Strategi ini terbukti efektif dalam mengoptimalkan hasil prediksi curah hujan ekstrem dan memberikan dasar kuat untuk pengembangan sistem peringatan dini bencana hidrometeorologis di masa mendatang.

BAB VI

PERBANDINGAN MODEL MLP DENGAN ENSEMBLE MLP

6.1 Verifikasi Model MLP

Verifikasi model dilakukan untuk menilai kinerja MLP dengan berbagai arsitektur jumlah *hidden layer* dan jumlah *neuron* di setiap *hidden layer*. Tujuan utama dari verifikasi ini adalah untuk mengidentifikasi arsitektur terbaik yang mampu menghasilkan prediksi anomali curah hujan secara akurat, berdasarkan parameter verifikasi BMKG (akurasi, *false alarm*, *miss classification*). Hasil pengujian model MLP dengan 1 *hidden layer* hingga 3 *hidden layer* menunjukkan variasi performa yang signifikan. Secara umum, penambahan jumlah *hidden layer* cenderung meningkatkan kemampuan model dalam menangkap kompleksitas hubungan nonlinier antara indeks iklim global dan curah hujan bulanan di Pacet. Namun, peningkatan kompleksitas jaringan tidak selalu diikuti oleh peningkatan performa yang stabil, karena risiko *overfitting* juga meningkat.

Keteraturan alam yang dijelaskan dalam QS Ar-Rum ayat 48 sebelumnya, menuntut manusia untuk bekerja dengan cermat dan penuh tanggung jawab dalam memahami fenomena alam. Hal ini ditegaskan oleh Nabi Muhammad *ṣallallahu ‘alaihi wasalam* dalam HR. Al-Bayhaqi, *Syū‘ab al-Iman*, Jilid 4, No. 5313 sebagai berikut:

إِنَّ اللَّهَ يُحِبُّ إِذَا عَمِلَ أَحَدُكُمْ عَمَلًا أَنْ يُثَوِّبَهُ

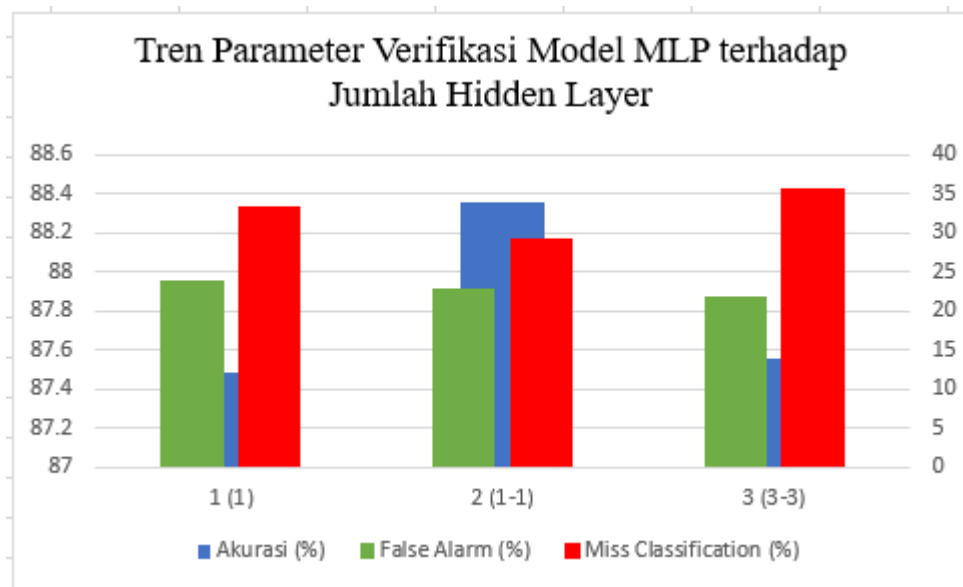
“Sesungguhnya Allah mencintai apabila seseorang melakukan suatu pekerjaan, maka ia melakukannya dengan *itqan* (tepat, teliti, dan optimal).” (HR. al-Bayhaqi, *Syū‘ab*

al-Iman, Jilid 4, No. 5313). Hadits ini menjadi representasi praktis dari QS Ar-Rum ayat 48 yaitu menekankan agar manusia melakukan prinsip ketelitian dalam pekerjaannya, sebagaimana hujan turun melalui mekanisme alam yang teratur dan presisi, penelitian ilmiah menuntut presisi dalam evaluasi model agar hasil yang dicapai sesuai dengan kaidah ilmiah dan bermanfaat bagi masyarakat. Hadits ini menjelaskan pentingnya keseimbangan dalam segala hal, termasuk dalam konteks ilmiah, model yang terlalu kompleks justru dapat merusak keseimbangan antara ketepatan dan generalisasi. Oleh karena itu, pemilihan arsitektur optimal perlu mempertimbangkan bukan hanya akurasi, tetapi juga stabilitas dan keandalan prediksi. Tabel 6.1 menampilkan perbandingan hasil pengujian dari setiap arsitektur model berdasarkan 30 kali pengujian.

Tabel 6.1 Hasil Verifikasi Menggunakan Kontigensi BMKG Tiap Skenario Model

Arsitektur		Rata-Rata dari 30 kali percobaan		
Hidden Layer	Neuron	Akurasi	False Alarm	Miss Classification
1	1	87.48	23.71	33.32666667
2	1.1	88.36	22.79333333	29.29333333
3	1.1.1	87.56	21.83333333	35.60666667
1	2	87.04	24.40333333	35.07333333
2	2.2	87.39666667	24.07333333	33.32666667
3	2.2.2	87.84	22.77	32.45666667
1	3	87.48	23.14666667	34.19666667
2	3.3	87.23666667	23.92333333	34.55333333
3	3.3.3	87.92	22.83666667	31.92
1	4	86.36	24.66	38.94
2	4.4	86.36	25.38	37.88
3	4.4.4	85.86	27.18	40
1	5	85.88	26.02	39.98
2	5.5	85.86	25.38	41.04
3	5.5.5	86.1	25.76	38.94

Hasil pada Tabel 6.1 menjelaskan bahwa peningkatan jumlah *hidden layer* dari 1 menjadi 2 *hidden layer* memberikan peningkatan akurasi yang cukup berarti, dengan rata-rata kenaikan sekitar 0,5–1% dibandingkan 1 *hidden layer*. Model 2 *hidden layer* dengan neuron 1.1 (DUAL_1.1) menunjukkan akurasi tertinggi sebesar 88.36%, dengan *false alarm* 22.79% dan *miss classification* 29.29%, sehingga dapat dikategorikan sebagai arsitektur paling baik antara ketepatan prediksi dan kesalahan klasifikasi. Sebaliknya, model dengan 3 *hidden layer* menunjukkan sedikit perbaikan pada variabel *miss classification*, peningkatan tersebut tidak signifikan dan diiringi dengan kenaikan *false alarm* pada beberapa arsitektur variasi *neuron*. Hal ini menandakan bahwa penambahan *hidden layer* yang berlebihan justru meningkatkan kompleksitas model tanpa menghasilkan peningkatan performa yang proporsional, bahkan cenderung mengarah pada *overfitting*.



Gambar 6.1 Tren Parameter Verifikasi Model MLP terhadap Jumlah *Hidden Layer*

Gambar 6.1 memperlihatkan perubahan nilai akurasi, *false alarm*, dan *miss classification* pada tiga arsitektur terbaik setiap skenario MLP. Terlihat bahwa akurasi meningkat signifikan dari arsitektur 1 *hidden layer* (87,48%) menuju 2 *hidden layer* (88,36%), sebelum kembali menurun pada 3 *hidden layer* (87,56%). Pola ini menunjukkan bahwa penambahan *hidden layer* hingga 2 *hidden layer* mampu meningkatkan kemampuan model dalam mengenali pola hubungan antara indeks iklim global dan curah hujan. Namun, ketika kompleksitas jaringan meningkat lebih jauh, performa model mulai jenuh dan cenderung menurun. Sebaliknya, parameter kesalahan (*false alarm* dan *miss classification*) menunjukkan tren yang relatif stabil, meskipun terjadi sedikit penurunan pada arsitektur 2 *hidden layer*. Hal ini mengindikasikan bahwa arsitektur 2 *hidden layer* memberikan keseimbangan terbaik antara akurasi tinggi dan tingkat kesalahan rendah. Meskipun demikian, hasil pengujian menunjukkan adanya variasi nilai akurasi yang cukup lebar, menandakan bahwa model masih sensitif terhadap kondisi awal bobot dan pembagian data pengujian.

Proses menguji, memvalidasi, dan mengulang prediksi ini selaras dengan prinsip QS. Fussilat ayat 53 tentang bagaimana kebenaran dapat menjadi jelas melalui pengamatan berulang terhadap fenomena alam yang menegaskan bahwa kebenaran dan keteraturan alam akan tampak jelas melalui proses pengujian yang berulang dan terukur, prinsip yang identik dengan evaluasi model. Dengan kata lain, stabilitas pengetahuan tidak lahir dari satu kali pengamatan, melainkan dari akumulasi bukti yang konsisten, sebagaimana model MLP memerlukan *ensemble* untuk mengurangi ketidakstabilan prediksi.

Rasulullah *ṣallallahu ‘alaihi wasalam* juga mengajarkan pentingnya konsistensi dan keberlanjutan dalam melakukan sesuatu agar hasilnya stabil dan dapat dipercaya. Hal ini ditegaskan melalui hadits HR. Bukhari No. 6463; Muslim No. 783 sebagai berikut:

«عَنْ عَائِشَةَ رَضِيَ اللَّهُ عَنْهَا قَالَتْ: قَالَ رَسُولُ اللَّهِ صَلَّى اللَّهُ عَلَيْهِ وَسَلَّمَ: «أَحَبُّ الْأَعْمَالِ إِلَى اللَّهِ أَدْوَمُهَا وَإِنْ قَلَّ»

“Amalan yang paling dicintai Allah adalah amalan yang dilakukan secara terus-menerus meskipun sedikit.”

Hadits ini selaras dengan QS Fussilat 53, yaitu menekankan bahwa nilai tertinggi bukan pada hasil besar, tetapi pada kestabilan, kontinuitas, dan konsistensi. Melalui pendekatan *bagging*, model memperoleh sifat “istiqamah” yang artinya kemampuan untuk memberikan hasil yang konsisten meskipun data dan bobot awal berubah.

Oleh karena itu, meskipun arsitektur 2 *hidden layer* menghasilkan nilai terbaik, kestabilan hasil prediksi masih perlu ditingkatkan melalui penerapan pendekatan *ensemble*, *bagging*, untuk mengurangi variansi model dan meningkatkan reliabilitas prediksi.

6.2 Verifikasi Model *Ensemble Bagging*

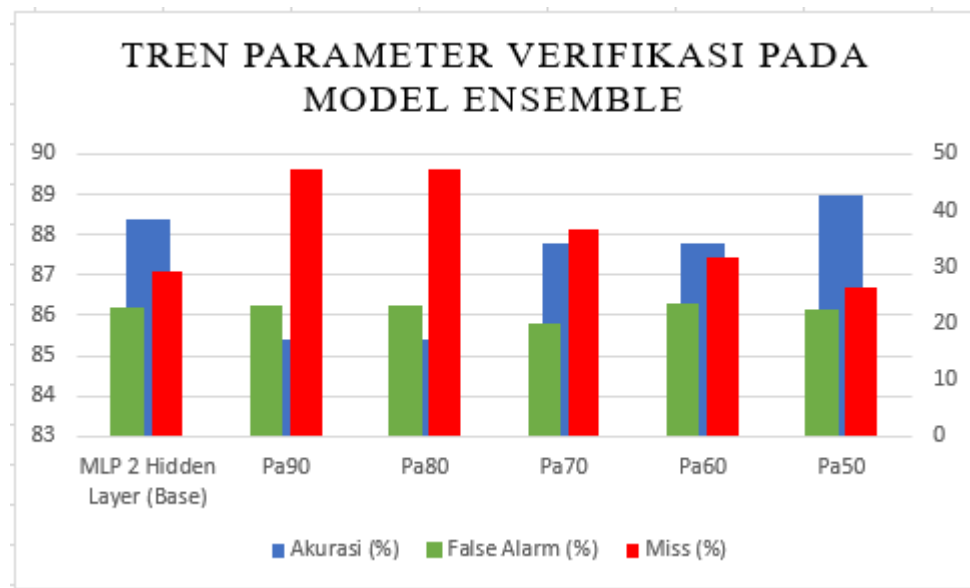
Pendekatan *ensemble bagging* diterapkan untuk meningkatkan reliabilitas hasil prediksi model MLP terhadap anomali curah hujan bulanan di Pacet. Pada penelitian ini, model dasar (*base learner*) yang digunakan dalam *ensemble* adalah arsitektur MLP dengan 2 *hidden layer* dan *neuron* 1.1 (DUAL_1.1), karena model tersebut terbukti

memberikan performa terbaik pada pengujian tunggal sebelumnya dengan akurasi 88,36%, *false alarm* 22,79%, dan *miss classification* 29,29%.

Tabel 6.2. Hasil Verifikasi *Ensemble Bagging* dan Model Terbaik MLP

Variabel Verivikasi	MLP 2	Threshold Probability				
		90%	80%	70%	60%	50%
Akurasi	88.4%	85.4%	85.4%	87.8%	87.8%	89.0%
<i>False Alarm</i>	22.8%	23.1%	23.1%	20.0%	23.5%	22.2%
<i>Miss Classification</i>	29.3%	47.4%	47.4%	36.8%	31.6%	26.3%

Tabel 6.2 menunjukkan bahwa *Ensemble Bagging* mampu meningkatkan akurasi sebesar 1,4% dibandingkan model MLP tunggal terbaik (dari 88,4% naik menjadi 89,0%). Selain itu, terjadi penurunan nilai *false alarm* dan *miss classification* secara konsisten di hampir seluruh skenario arsitektur *Ensemble Bagging*. Misalnya, pada skenario *threshold probability* 50%, nilai *false alarm* menurun menjadi 22,2% dan nilai *miss classification* turun menjadi 26,3%. Hal ini menunjukkan bahwa proses *ensemble* melalui *bagging* berhasil meningkatkan kestabilan model dan mengurangi kesalahan klasifikasi, terutama pada kondisi data pelatihan yang memiliki variasi tinggi. Dengan adanya skenario *Ensemble Bagging* melalui proses agregasi, kesalahan yang muncul pada satu model tunggal MLP dapat diimbangi oleh model yang dihasilkan dari *resampling* pada *Ensemble Bagging*, sehingga nilai akhir prediksi curah hujan akhir menjadi lebih seimbang dan bisa menjadi lebih baik dalam mengurangi variansi.



Gambar 6.2 Tren Parameter Verifikasi Pada Model *Ensemble*

Gambar 6.2 memperlihatkan tren perubahan nilai akurasi, *false alarm*, dan *miss classification* pada berbagai skema *threshold* (90%–50%) dibandingkan model MLP 2 *hidden layer* (*base learner*). Terlihat bahwa nilai akurasi mengalami peningkatan bertahap, dari 88,4% pada model MLP tunggal menjadi 89,0% pada skema 50%, yang merupakan nilai tertinggi dalam keseluruhan pengujian. Sebaliknya, nilai *false alarm* dan *miss classification* menunjukkan tren penurunan yang konsisten pada skema *ensemble* yang lebih rendah (70%–50%). Penurunan ini mengindikasikan bahwa pendekatan *bagging* mampu menekan kesalahan prediksi, baik berupa prediksi hujan yang salah (*false alarm*) maupun hujan yang terlewat (*miss classification*). Pola grafik juga memperlihatkan bahwa distribusi nilai akurasi antar skema *ensemble* cenderung lebih baik dibandingkan MLP tunggal, dengan deviasi antar titik yang kecil. Hal ini menunjukkan bahwa model *ensemble* berhasil mengurangi variansi hasil prediksi yang sebelumnya cukup tinggi pada model MLP individu. Dengan demikian, penerapan

bagging tidak hanya meningkatkan akurasi rata-rata, tetapi juga memperkuat konsistensi dan reliabilitas performa model dalam memprediksi anomali curah hujan bulanan di Pacet. Secara keseluruhan, tren pada Gambar 6.2 menegaskan bahwa kolaborasi antar model individu dalam *ensemble* menghasilkan sistem prediksi yang lebih seimbang dan akurasi meningkat, sementara tingkat kesalahan menurun. Hasil ini memperkuat temuan pada Tabel 6.2 bahwa metode *Ensemble Bagging* lebih unggul dan stabil dibandingkan model MLP tunggal.

Fenomena ini mengaitkan kajian ilmiah dengan nilai keislaman keteraturan proses meteorologis, sebagaimana telah dijelaskan sebelumnya pada QS. An-Nur ayat 43 yang menggambarkan mekanisme pembentukan awan, penggabungan massa udara, hingga keluarnya hujan yang merupakan fenomena dinamis yang menjadi dasar ilmiah dari model prediksi hidrometeorologi. Pemahaman tentang keteraturan fenomena hujan ini juga ditegaskan dalam hadits Rasulullah *ṣallallahu‘alaihi wasalam* yang menjelaskan bahwa hujan turun berdasarkan ketentuan Allah, bukan karena faktor-faktor mistik atau kepercayaan takhayul. Hadits tersebut diriwayatkan oleh Imam al-Bukhari dari Zaid bin Khalid al-Juhani no 846 sebagai berikut:

عَنْ زَيْدِ بْنِ خَالِدٍ الْجُهَنِيِّ قَالَ صَلَّى لَنَا رَسُولُ اللَّهِ ﷺ صَلَاةَ الصُّبْحِ بِالْحُدَيْبِيَّةِ عَلَى إِثْرِ سَمَاءٍ كَانَتْ مِنَ اللَّيْلِ، فَلَمَّا أَنْصَرَفَ أَقْبَلَ عَلَى النَّاسِ فَقَالَ: هَلْ تَذَرُونَ مَاذَا قَالَ رَبِّكُمْ؟ قَالُوا: اللَّهُ وَرَسُولُهُ أَعْلَمُ. قَالَ: أَصْبَحَ مِنْ عِبَادِي مُؤْمِنٌ بِي وَكَافِرٌ، فَأَمَّا مَنْ قَالَ مُطَرْنَا بِفَضْلِ اللَّهِ وَرَحْمَتِهِ فذلِكَ مُؤْمِنٌ بِي وَكَافِرٌ بِالْكُوفَةِ، وَأَمَّا مَنْ قَالَ مُطَرْنَا بِنُوءٍ كَذَا وَكَذَا فذلِكَ كَافِرٌ بِي وَمُؤْمِنٌ بِالْكُوفَةِ.

Dari Zaid bin Khalid al-Juhani, ia berkata bahwa Rasulullah *ṣallallahu‘alaihi wasalam* menunaikan shalat Subuh bersama kami di Hudaibiyah setelah turunnya hujan pada malam hari. Setelah selesai, beliau bersabda: “Tahukah kalian apa yang difirmankan Tuhan kalian?” Mereka menjawab: “Allah dan Rasul-Nya lebih mengetahui.” Beliau bersabda: “Pagi ini ada di antara hamba-hamba-Ku yang beriman kepada-Ku dan ada yang kafir. Adapun yang mengatakan ‘Kami diberi hujan karena karunia dan rahmat Allah,’ maka ia beriman kepada-Ku dan kufur terhadap

bintang-bintang. Sedangkan yang mengatakan ‘Kami diberi hujan karena bintang ini dan itu,’ maka ia kafir kepada-Ku dan beriman kepada bintang-bintang.”

QS. An-Nur ayat 43 serta hadits Shahih al-Bukhari tersebut memberikan dasar filosofis bahwa evaluasi model prediksi curah hujan bukan hanya proses matematis, tetapi juga merupakan manifestasi dari perintah Allah untuk mengamati, memahami, dan memanfaatkan pola-pola alam dalam rangka kemaslahatan manusia. Narasi ini memperkuat posisi penelitian sebagai integrasi antara sains atmosfer, teknologi kecerdasan buatan, dan nilai keislaman yang menekankan keteraturan ciptaan Allah.

Dengan demikian, hasil penelitian menunjukkan bahwa pendekatan *ensemble bagging* tidak hanya memperkuat performa sistem prediksi curah hujan, tetapi juga selaras dengan prinsip-prinsip Islam tentang keseimbangan, kebersamaan, dan ikhtiar kolektif dalam mencapai hasil terbaik.

BAB VII

PENUTUP

7.1 Kesimpulan

Berdasarkan hasil penelitian mengenai prediksi curah hujan bulanan di wilayah Pacet menggunakan metode *Artificial Neural Network* (ANN) dengan arsitektur *Multi-Layer Perceptron* (MLP) serta pendekatan *ensemble*, dapat disimpulkan hal-hal berikut:

1. Beberapa skenario arsitektur MLP menunjukkan variasi performa prediksi, di mana arsitektur 2 *hidden layer* dengan *neuron* 1,1 menghasilkan verifikasi terbaik dengan tingkat akurasi tertinggi serta *false alarm* dan *miss classification* paling rendah dibandingkan skenario arsitektur lain. Hasil pengujian menunjukkan bahwa arsitektur 2 *hidden layer* dengan *neuron* 1.1 mencapai akurasi 88,4%, dengan *false alarm* 22,8% dan *miss classification* 29,3%. Hal ini menegaskan bahwa pemilihan jumlah *hidden layer* dan *neuron* sangat memengaruhi hasil prediksi, sehingga pemilihan arsitektur yang tepat menjadi faktor krusial.
2. Pendekatan *Ensemble Bagging* yang diterapkan pada hasil arsitektur terbaik memberikan hasil prediksi dengan verifikasi kontigensi yang lebih baik dibanding arsitektur tunggal MLP. Dengan melakukan *resampling* data uji sebanyak 100 kali dan menerapkan ambang batas probabilitas (50%, 60%, 70%, 80%, 90%), *ensemble* mampu memperhalus variasi prediksi dan mengurangi resiko kesalahan tunggal dari satu model MLP. Hasil pengujian MLP *Ensemble*

Bagging menunjukkan peningkatan kinerja, mencapai akurasi tertinggi 89,0%, *false alarm* 22,2% dan *miss classification* terendah 26,3%. pada ambang batas 50%, sementara *false alarm* terendah 20,0% diperoleh pada ambang batas 70%. Temuan ini menunjukkan bahwa *Ensemble Bagging* berhasil mengurangi variansi model, mereduksi kesalahan klasifikasi, dan menghasilkan akurasi rata-rata yang lebih tinggi. Dengan performa yang lebih stabil dan *reliable*, *Ensemble*, dapat menjadi dasar sistem pendukung keputusan dalam pengelolaan sumber daya air serta mitigasi bencana longsor di Pacet.

7.2 Saran

Penelitian ini memberikan pandangan bagi penulis untuk menyampaikan beberapa saran sebagai berikut:

1. Direkomendasikan agar penelitian selanjutnya menggeser fokus dari penggunaan informasi curah hujan bulanan menuju prediksi dengan resolusi waktu yang lebih detail, misalnya skala dasarian (10 harian). Hal ini akan meningkatkan ketepatan dalam mendukung kebutuhan operasional lapangan, khususnya untuk pengelolaan sumber daya air dan mitigasi bencana di daerah rawan longsor.
2. Diperlukan penelitian lanjutan untuk menyusun pendekatan empiris yang lebih akurat terkait hubungan antara curah hujan ekstrem dan dampaknya terhadap kejadian hidrometeorologi, seperti aliran sungai maupun potensi longsor. Selain itu, integrasi dengan proyeksi iklim jangka panjang dapat memperkuat daya guna model dalam konteks adaptasi perubahan iklim.

3. Sehubungan dengan keterbatasan penelitian ini yang hanya berfokus pada wilayah Pacet, diharapkan penelitian serupa dapat diperluas pada daerah rawan longsor atau daerah tangkapan air lainnya. Dengan demikian, hasil penelitian dapat memberikan manfaat yang lebih umum dan mendukung kebijakan mitigasi bencana berbasis data iklim di berbagai wilayah.

DAFTAR PUSTAKA

- Agustina, L., et. al. (2020). Prediksi Curah Hujan Bulanan Untuk Peringatan Dini Longsor Di Banjarnegara Bagian Selatan Dengan *Statistical Downscaling* Dan *Ensemble*. *Jurnal Meteorologi Dan Geofisika*.
- Aldrian, E., & Susanto, R. D. (2003). *Identification of three dominant rainfall regions within Indonesia and their relationship to sea surface temperature*. *International Journal of Climatology*, 23(12), 1435–1452. doi: 10.1002/joc.950
- Attaf, Y., et. al. (2024). *Enhancement of ANN Performance for Remote-Sensing Rainfall Estimates Using Ensemble Methods*. *Journal of Earth System Science*, 133(92), Article 92. <https://doi.org/10.1007/s41748-024-0092>
- Aizansi, A. N., et al. (2024). *Monthly rainfall prediction using artificial neural network (case study: Republic of Benin)*. *Environmental Data Science*. doi: 10.1017/eds.2024.10
- Baljon, A., & Sharma, P. (2023). *Rainfall prediction rate in Saudi Arabia using improved machine learning techniques*. *International Journal of Advanced Computer Science and Applications (IJACSA)*, 14(1), 456–463.
- Badan Meteorologi Klimatologi dan Geofisika. (2019). Peraturan Badan Meteorologi Klimatologi dan Geofisika Nomor 9 Tahun 2019 Tentang Penyediaan Dan Penyebaran Peringatan Dini Iklim Ekstrem.

- Bouach, A. (2024). *Artificial Neural Networks For Monthly Precipitation Prediction In North-West Algeria: A Case Study In The Oranie-Chott-Chergui Basin. Journal Of Water And Climate Change*, 15(2), 582–598. Doi: 10.2166/Wcc.2024.494
- Chand, A., & Nand, R. (2020). *Rainfall Prediction Using Artificial Neural Network In The South Pacific Region*.
- Hafyani, et. al. (2024). *Improving Monthly Precipitation Prediction Accuracy Using Multi-View Stacking. Frontiers in Water*, 3, Article 1378598. <https://doi.org/10.3389/frwa.2024.1378598>
- Helali, J., et. al. (2022). *Assessment of machine learning model performance for seasonal precipitation simulation based on teleconnection indices in Iran. Arabian Journal of Geosciences*, 15(15), 1343. <https://doi.org/10.1007/s12517-022-10640-2>
- Khastagir, A. (2022). *Efficacy Of Lmr And Ann For Long Term Rainfall Forecasting In Western Australia. Meteorology And Atmospheric Physics*. Doi: 10.1007/S00703-022-00907-4
- Maharani, T. (2019). *Pemodelan Bahaya Kekeringan Meteorologis Di Provinsi Jawa Timur Dengan Menggunakan Data CHIRPS [Tesis]*. Universitas Gadjah Mada.
- Martínez, S., Tan, M. L., & Walker, G. (2022). *Prediction Of Rainfall In Australia Using Machine Learning. Sustainability*, 14(11), 6649. Doi: 10.3390/Su14116649

- Mohammadpouri, S. (2023). *A Generalized Regression Neural Network Model For Accuracy Improvement Of Global Precipitation Products*. *Sustainability*, 15(11), 8740. Doi: 10.3390/Su15118740
- Najib, F., & Mustika, I. W. (2022). *Weather Forecasting Using Ann For Rice Farming In Delanggu Village*. *Iop Conference Series: Earth And Environmental Science*, 1030(1), 012002. Doi: 10.1088/1755-1315/1030/1/012002
- Papailiou, I., et. al. (2022). *Artificial Neural Networks For Monthly Precipitation Prediction In North-West Algeria: A Case Study In The Oranie-Chott-Chergui Basin*. *Water*, 14(18), 2892. Doi: 10.3390/W14182892
- Putra, R. M. dkk. (2024). *High-Resolution Rainfall Estimation Using Ensemble Learning & Multisensor Integration*. *Sensors*, <https://doi.org/10.3390/s24155030>
- Putra, R. M., & Rani, N. A. (2020). *Prediksi Curah Hujan Harian Di Stasiun Meteorologi Kemayoran Menggunakan Artificial Neural Network (ANN)*. *Buletin GAW Bariri*, 1(2). doi: 10.31172/bgb.v1i2.35
- Safril, A., et al. (2020). *Prediksi Curah Hujan Bulanan Untuk Peringatan Dini Longsor Di Banjarnegara Bagian Selatan Dengan Statistical Downscaling Dan Ensemble*. *Jurnal Meteorologi dan Geofisika*.
- Singh, M. (2021). *Deep Learning For Improved Global Precipitation In Numerical Weather Prediction Systems*.
- Sittichok, K. (2024). *Long-Term Seasonal Rainfall Forecasting Using Regression And ANN*. *Journal of Scientific and Technological Reports*, 27(3). doi: 10.55164/ajstr.v27i3.253507

Surana, A., et. al. (2024). *Ensemble Fine-Tuned MLP for Rainfall Forecasting*.

ICTACT *Journal on Soft Computing*, 15(2), 3491–3496.

<https://doi.org/10.21917/ijsc.2024.03004>

You, H., et al. (2024). *Prediction Of Tropical Pacific Rain Rates With*

Overparameterized Neural Networks. Artificial Intelligence for the Earth

Systems. doi: 10.1175/AIES-D-23-0083.1

Lampiran 1. Skrip Prediksi Curah Hujan Menggunakan 1 *Hidden Layer*

```
library(neuralnet)
FU="E:\\DEFINTA_THESIS\\TES_PRA_SEMPRO"

DATA=read.csv(paste(FU,"\\DATA_INTENET.csv",sep=""))
DATA=DATA[1:410,]
DATA[, "KAT_ANOM_CH"]=rep(NA,nrow(DATA))
summary(DATA)
NAMA_KOL=colnames(DATA)

form1=paste0(as.character(NAMA_KOL[c(8,9,10,11,12,13,16,19)]),collapse=","sep="
+")
form2=substr(form1,1,(nchar(form1)-1))
form3=paste(as.character(NAMA_KOL[c(6)]), "~ ", form2)
form=as.formula(form3)

#NORMALISASI MAXMIN DATA
data_rows <- floor(0.80 * nrow(DATA))
train_indices <- 1:data_rows
train_data <- DATA[train_indices,]
MAX=as.numeric(apply(train_data,2,max))
MIN=as.numeric(apply(train_data,2,min))
RANGE=MAX-MIN
MAX_anomch=as.numeric(apply(train_data,2,max)["ANOM_CH"])
MIN_anomch=as.numeric(apply(train_data,2,min)["ANOM_CH"])
RANGE_anomch=MAX_anomch-MIN_anomch
train_data_nrm=(train_data-
matrix(MIN,nrow(train_data),ncol(train_data),byrow=T))/
matrix(RANGE,nrow(train_data),ncol(train_data),byrow=T)
summary(train_data_nrm)
test_data <- DATA[-train_indices,]
test_data_nrm=(test_data-matrix(MIN,nrow(test_data),ncol(test_data),byrow=T))/
matrix(RANGE,nrow(test_data),ncol(test_data),byrow=T)
summary(test_data_nrm)

nhidden=1
while(nhidden<=5){

iterasi=1
while(iterasi<=5){

model = neuralnet(form,
data=train_data_nrm,
hidden=c(nhidden),
```

```

linear.output = FALSE)

png(paste(FU,"\\OUTPUT\\PLOT\\SINGLE_",100+nhidden,"_iterasi=",100+iterasi,"
.png",sep=""))
plot(model,rep = "best")
dev.off()

saveRDS(model,paste(FU,"\\OUTPUT\\MODEL\\SINGLE_",100+nhidden,"_iterasi
=",100+iterasi,".rds",sep=""))

pred <- round(predict(model, test_data_nrm),3)
predanomch=as.numeric(pred*RANGE_anomch+MIN_anomch)
predch=round(as.numeric(test_data[, "NORMALCH_1991.2020"])+predanomch,1)
predch[predch<=0]=0
obsch=round(as.numeric(test_data[, "CHIRPS"]),1)
RKP=cbind(pred,predanomch,as.numeric(test_data[, "NORMALCH_1991.2020"]),pr
edch,obsch)
colnames(RKP)=c("Luaran_NN","Prediksi_Anom_CH","Rata-Rata 1991-
2020","Prediksi Curah Hujan","Observasi Curah Hujan")
write.csv(RKP,paste(FU,"\\OUTPUT\\HASIL_PREDIKSI\\SINGLE_",100+nhidden,
"_iterasi=",100+iterasi,".csv",sep=""))

VERIF=as.data.frame(cbind(predch,as.numeric((predch>=500)),as.numeric((obsch>=
500)),obsch))
colnames(VERIF)[1:4]=c("Pred_CH","Pred","Obs","Obs_CH")
TABEL=as.matrix(table(VERIF[,c("Pred","Obs")]))
TABEL
TABEL2=TABEL
#Akurasi
Akurasi=round((TABEL["1","1"]+TABEL["0","0"])/sum(TABEL)*100,1)
#False Alarm
FalseAlarm=round((TABEL["1","0"])/sum(TABEL["1",])*100,1)
#Miss
Miss=round((TABEL["0","1"])/sum(TABEL[, "1"])*100,1)
#Error
Error=model$result.matrix[1]
#Step
Step=model$result.matrix[3]

VERIF2=matrix(c(Error,Step,Akurasi,FalseAlarm,Miss,iterasi),1,)
rownames(VERIF2)=paste("SINGLE_",100+nhidden,sep="")
colnames(VERIF2)=c("Error","Step","Akurasi","FalseAlarm","Miss","Iterasi")

```

```

write.csv(VERIF2,paste(FU,"\\OUTPUT\\VERIF\\VERIF_SINGLE_",100+nhidden,
"_iterasi=",100+iterasi,".csv",sep=""))
print(paste("VERIF_SINGLE_",100+nhidden,"_iterasi=",100+iterasi,sep=""))
flush.console()

iterasi=iterasi+1
}
nhidden=nhidden+1
}
##### Mengumpulkan hasil verifikasi menjadi satu
setwd(paste(FU,"\\OUTPUT\\VERIF",sep=""))
nama=list.files()
batas=length(nama)
nk=1
RKP=c()
while(nk<=batas){
verif=read.csv(nama[nk])
RKP=rbind(RKP,verif)
nk=nk+1
}
write.csv(RKP,paste(FU,"\\OUTPUT\\REKAP_VERIF.csv",sep=""))

```

Lampiran 2. Skrip Prediksi Curah Hujan Menggunakan 2 *Hidden Layer*

```

library(neuralnet)
FU="E:\\DEFINTA_THESIS\\TES_PRA_SEMPRO"

DATA=read.csv(paste(FU,"\\DATA_INTENET.csv",sep=""))
DATA=DATA[1:410,]
DATA[,"KAT_ANOM_CH"]=rep(NA,nrow(DATA))
summary(DATA)
NAMAOL=colnames(DATA)

form1=paste0(as.character(NAMAOL[c(8,9,10,11,12,13,16,19)]),collapse=",sep="
+)
form2=substr(form1,1,(nchar(form1)-1))
form3=paste(as.character(NAMAOL[c(6)]), "~", form2)
form=as.formula(form3)

#NORMALISASI MAXMIN DATA
data_rows <- floor(0.80 * nrow(DATA))
train_indices <- 1:data_rows
train_data <- DATA[train_indices,]
MAX=as.numeric(apply(train_data,2,max))
MIN=as.numeric(apply(train_data,2,min))
RANGE=MAX-MIN
MAX_anomch=as.numeric(apply(train_data,2,max)["ANOM_CH"])
MIN_anomch=as.numeric(apply(train_data,2,min)["ANOM_CH"])
RANGE_anomch=MAX_anomch-MIN_anomch
train_data_nrm=(train_data-
matrix(MIN,nrow(train_data),ncol(train_data),byrow=T))/
matrix(RANGE,nrow(train_data),ncol(train_data),byrow=T)
summary(train_data_nrm)
test_data <- DATA[-train_indices,]
test_data_nrm=(test_data-matrix(MIN,nrow(test_data),ncol(test_data),byrow=T))/
matrix(RANGE,nrow(test_data),ncol(test_data),byrow=T)
summary(test_data_nrm)

nhidden=1
while(nhidden<=5){

iterasi=1
while(iterasi<=5){

model = neuralnet(form,
data=train_data_nrm,
hidden=c(nhidden,nhidden),

```

```

linear.output = FALSE)

png(paste(FU,"\\OUTPUT\\PLOT\\DUAL_",100+nhidden,"_iterasi=",100+iterasi,".png",sep=""))
plot(model,rep = "best")
dev.off()

saveRDS(model,paste(FU,"\\OUTPUT\\MODEL\\DUAL_",100+nhidden,"_iterasi=",100+iterasi,".rds",sep=""))

pred <- round(predict(model, test_data_nrm),3)
predanomch=as.numeric(pred*RANGE_anomch+MIN_anomch)
predch=round(as.numeric(test_data[, "NORMALCH_1991.2020"])+predanomch,1)
predch[predch<=0]=0
obsch=round(as.numeric(test_data[, "CHIRPS"]),1)

RKP=cbind(pred,predanomch,as.numeric(test_data[, "NORMALCH_1991.2020"]),predch,obsch)
colnames(RKP)=c("Luaran_NN", "Prediksi_Anom_CH", "Rata-Rata 1991-2020", "Prediksi Curah Hujan", "Observasi Curah Hujan")

write.csv(RKP,paste(FU,"\\OUTPUT\\HASIL_PREDIKSI\\DUAL_",100+nhidden,"_iterasi=",100+iterasi,".csv",sep=""))

VERIF=as.data.frame(cbind(predch,as.numeric((predch>=500)),as.numeric((obsch>=500)),obsch))
colnames(VERIF)[1:4]=c("Pred_CH", "Pred", "Obs", "Obs_CH")
TABEL=as.matrix(table(VERIF[,c("Pred", "Obs")]))
TABEL
TABEL2=TABEL
#Akurasi
Akurasi=round((TABEL["1","1"]+TABEL["0","0"])/sum(TABEL)*100,1)
#False Alarm
FalseAlarm=round((TABEL["1","0"])/sum(TABEL["1",])*100,1)
#Miss
Miss=round((TABEL["0","1"])/sum(TABEL[, "1"])*100,1)
#Error
Error=model$result.matrix[1]
#Step
Step=model$result.matrix[3]

VERIF2=matrix(c(Error,Step,Akurasi,FalseAlarm,Miss,iterasi),1,)
rownames(VERIF2)=paste("DUAL_",100+nhidden,sep="")
colnames(VERIF2)=c("Error", "Step", "Akurasi", "FalseAlarm", "Miss", "Iterasi")

```

```

write.csv(VERIF2,paste(FU,"\\OUTPUT\\VERIF\\VERIF_DUAL_",100+nhidden,"_
iterasi=",100+iterasi,".csv",sep=""))
print(paste("VERIF_DUAL_",100+nhidden,"_iterasi=",100+iterasi,sep=""))
flush.console()

iterasi=iterasi+1
}
nhidden=nhidden+1
}
##### Mengumpulkan hasil verifikasi menjadi satu
setwd(paste(FU,"\\OUTPUT\\VERIF",sep=""))
nama=list.files()
batas=length(nama)
nk=1
RKP=c()
while(nk<=batas){
verif=read.csv(nama[nk])
RKP=rbind(RKP,verif)
nk=nk+1
}
write.csv(RKP,paste(FU,"\\OUTPUT\\REKAP_VERIF.csv",sep=""))

#boxplot(matrix(RKP[, "Akurasi"],,8))
#boxplot(matrix(RKP[, "FalseAlarm"],,8))
#boxplot(matrix(RKP[, "Miss"],,8))

```



```

linear.output = FALSE)

par(mar=c(8,8,8,8))

png(paste(FU,"\\OUTPUT\\PLOT\\TRIPLE_",100+nhidden,"_iterasi=",100+iterasi,".
png",sep=""),width=2000,height=700)
plot(model,rep = "best",show.names=T)
dev.off()

saveRDS(model,paste(FU,"\\OUTPUT\\MODEL\\TRIPLE_",100+nhidden,"_iterasi=
",100+iterasi,".rds",sep=""))

pred <- round(predict(model, test_data_nrm),3)
predanomch=as.numeric(pred*RANGE_anomch+MIN_anomch)
predch=round(as.numeric(test_data[, "NORMALCH_1991.2020"])+predanomch,1)
predch[predch<=0]=0
obsch=round(as.numeric(test_data[, "CHIRPS"]),1)

RKP=cbind(pred,predanomch,as.numeric(test_data[, "NORMALCH_1991.2020"]),pr
edch,obsch)
colnames(RKP)=c("Luaran_NN", "Prediksi_Anom_CH", "Rata-Rata          1991-
2020", "Prediksi Curah Hujan", "Observasi Curah Hujan")

write.csv(RKP,paste(FU,"\\OUTPUT\\HASIL_PREDIKSI\\TRIPLE_",100+nhidden,
"_iterasi=",100+iterasi,".csv",sep=""))

VERIF=as.data.frame(cbind(predch,as.numeric((predch>=500)),as.numeric((obsch>=
500)),obsch))
colnames(VERIF)[1:4]=c("Pred_CH", "Pred", "Obs", "Obs_CH")
TABEL=as.matrix(table(VERIF[,c("Pred", "Obs")]))
TABEL
TABEL2=TABEL
#Akurasi
Akurasi=round((TABEL["1","1"]+TABEL["0","0"])/sum(TABEL)*100,1)
#False Alarm
FalseAlarm=round((TABEL["1","0"])/sum(TABEL["1",])*100,1)
#Miss
Miss=round((TABEL["0","1"])/sum(TABEL[, "1"])*100,1)
#Error
Error=model$result.matrix[1]
#Step
Step=model$result.matrix[3]

VERIF2=matrix(c(Error,Step,Akurasi,FalseAlarm,Miss,iterasi),1,)
rownames(VERIF2)=paste("TRIPLE_",100+nhidden,sep="")

```

```

colnames(VERIF2)=c("Error","Step","Akurasi","FalseAlarm","Miss","Iterasi")

write.csv(VERIF2,paste(FU,"\\OUTPUT\\VERIF\\VERIF_TRIPLE_",100+nhidden,"
_iterasi=",100+iterasi,".csv",sep=""))
print(paste("VERIF_TRIPLE_",100+nhidden,"_iterasi=",100+iterasi,sep=""))
flush.console()

iterasi=iterasi+1
}
nhidden=nhidden+1
}
##### Mengumpulkan hasil verifikasi menjadi satu
setwd(paste(FU,"\\OUTPUT\\VERIF",sep=""))
nama=list.files()
batas=length(nama)
nk=1
RKP=c()
while(nk<=batas){
  verif=read.csv(nama[nk])
  RKP=rbind(RKP,verif)
  nk=nk+1
}
write.csv(RKP,paste(FU,"\\OUTPUT\\REKAP_VERIF.csv",sep=""))

#boxplot(matrix(RKP[, "Akurasi"],,8))
#boxplot(matrix(RKP[, "FalseAlarm"],,8))
#boxplot(matrix(RKP[, "Miss"],,8))

```

Lampiran 4. Skrip Prediksi Curah Hujan Menggunakan Ensemble MLP

```

library(neuralnet)
library(ROCR)

FU="F:\\FOLDER_ANDANG_2025\\20251001_BMDEF3"
DATA=read.csv(paste(FU,"\\DATA_INTENET.csv",sep=""))
DATA=DATA[1:410,]
DATA[, "KAT_ANOM_CH"]=rep(NA,nrow(DATA))
summary(DATA)
NAMAOL=colnames(DATA)

form1=paste0(as.character(NAMAOL[c(8,9,10,11,12,13,16,19)]),collapse=","sep="
+")
form2=substr(form1,1,(nchar(form1)-1))
form3=paste(as.character(NAMAOL[c(6)]), "~", form2)
form=as.formula(form3)

#NORMALISASI MAXMIN DATA
data_rows <- floor(0.80 * nrow(DATA))
train_indices <- 1:data_rows
train_data <- DATA[train_indices,]
MAX=as.numeric(apply(train_data,2,max))
MIN=as.numeric(apply(train_data,2,min))
RANGE=MAX-MIN
MAX_anomch=as.numeric(apply(train_data,2,max)["ANOM_CH"])
MIN_anomch=as.numeric(apply(train_data,2,min)["ANOM_CH"])
RANGE_anomch=MAX_anomch-MIN_anomch
train_data_nrm=(train_data-
matrix(MIN,nrow(train_data),ncol(train_data),byrow=T))/
matrix(RANGE,nrow(train_data),ncol(train_data),byrow=T)
summary(train_data_nrm)
test_data <- DATA[-train_indices,]
test_data_nrm=(test_data-matrix(MIN,nrow(test_data),ncol(test_data),byrow=T))/
matrix(RANGE,nrow(test_data),ncol(test_data),byrow=T)
summary(test_data_nrm)

iterasi=1
while(iterasi<=100){

model = neuralnet(form,
data=train_data_nrm[sample(1:nrow(train_data_nrm), 1000,replace = T),],
hidden=c(1,1),
linear.output = FALSE)

```

```

png(paste(FU,"\\OUTPUT\\PLOT\\NHID_",102,"_bagg=",100+iterasi,".png",sep="
"))
plot(model,rep = "best")
dev.off()

saveRDS(model,paste(FU,"\\OUTPUT\\MODEL\\NHID_",102,"_bagg=",100+iterasi
, ".rds",sep=""))

pred <- round(predict(model, test_data_nrm),3)
predanomch=as.numeric(pred*RANGE_anomch+MIN_anomch)
predch=round(as.numeric(test_data[, "NORMALCH_1991.2020"])+predanomch,1)
predch[predch<=0]=0
obsch=round(as.numeric(test_data[, "CHIRPS"]),1)
RKP=cbind(pred,predanomch,as.numeric(test_data[, "NORMALCH_1991.2020"]),pr
edch,obsch)
colnames(RKP)=c("Luaran_NN","Prediksi_Anom_CH","Rata-Rata 1991-
2020","Prediksi_Curah_Hujan","Observasi_Curah_Hujan")
write.csv(RKP,paste(FU,"\\OUTPUT\\HASIL_PREDIKSI\\NHID_",102,"_bagg=",10
0+iterasi,".csv",sep=""))

VERIF=as.data.frame(cbind(predch,as.numeric((predch>=500)),as.numeric((obsch
>=500)),obsch))
colnames(VERIF)[1:4]=c("Pred_CH","Pred","Obs","Obs_CH")
TABEL=as.matrix(table(VERIF[,c("Pred","Obs")]))
TABEL
TABEL2=TABEL
#Akurasi
Akurasi=round((TABEL["1","1"]+TABEL["0","0"])/sum(TABEL)*100,1)
#False Alarm
FalseAlarm=round((TABEL["1","0"])/sum(TABEL["1",])*100,1)
#Miss
Miss=round((TABEL["0","1"])/sum(TABEL[, "1"])*100,1)
#Error
Error=model$result.matrix[1]
#Step
Step=model$result.matrix[3]

VERIF2=matrix(c(Error,Step,Akurasi,FalseAlarm,Miss,iterasi),1,)
rownames(VERIF2)=paste("NHID_",102,sep="")
colnames(VERIF2)=c("Error","Step","Akurasi","FalseAlarm","Miss","Iterasi")

write.csv(VERIF2,paste(FU,"\\OUTPUT\\VERIF\\VERIF_NHID_",102,"_bag=",100
+iterasi,".csv",sep=""))
print(paste("VERIF_NHID_",102,"_bag=",100+iterasi,sep=""))
flush.console()

```

```

iterasi=iterasi+1
}
##### Mengumpulkan hasil verifikasi menjadi satu
setwd(paste(FU,"\\OUTPUT\\VERIF",sep=""))
nama=list.files()
batas=length(nama)
nk=1
RKP=c()
while(nk<=batas){
verif=read.csv(nama[nk])
RKP=rbind(RKP,verif)
nk=nk+1
}
write.csv(RKP,paste(FU,"\\OUTPUT\\REKAP_VERIF.csv",sep=""))
##### Mengumpulkan hasil verifikasi menjadi satu
setwd(paste(FU,"\\OUTPUT\\HASIL_PREDIKSI",sep=""))
nama=list.files()
batas=length(nama)
nk=1
RKP=c()
while(nk<=batas){
verif=read.csv(nama[nk])
RKP=cbind(RKP,as.numeric(verif[, "Prediksi.Curah.Hujan"]))
nk=nk+1
}
write.csv(RKP,paste(FU,"\\OUTPUT\\REKAP_HASIL_PREDIKSI.csv",sep=""))
cekp=read.csv(paste(FU,"\\OUTPUT\\REKAP_HASIL_PREDIKSI.csv",sep=""))
RKP=cekp[,2:ncol(cekp)]
ceko=read.csv(paste(FU,"\\OUTPUT\\HASIL_PREDIKSI\\NHID_102_bagg=101.csv",sep=""))
obsch=as.numeric(ceko[, "Observasi.Curah.Hujan"])
pred=prediction(rowMeans(RKP>=500),as.numeric((obsch>=500)))
perf=performance(pred,"tpr","fpr")
plot(perf)

auc <- performance(pred, measure = "auc")
auc_value <- auc@y.values[[1]]
auc_value
#boxplot(matrix(RKP[, "Akurasi"],,8))
#boxplot(matrix(RKP[, "FalseAlarm"],,8))
#boxplot(matrix(RKP[, "Miss"],,8))

```

Lampiran 5. Percobaan Menggunakan 4 *Hidden Layer*.

Prediksi MLP	Prediksi Anomali Curah Hujan	Rerata Curah Hujan	Prediksi Curah Hujan	Observasi Curah Hujan	Keterangan
0.455	0.455	0.455	0.455	0.455	QUAD_1
0.455	-0.42103	25.36599	24.9	24.9	QUAD_1
0.456	0.285305	29.87074	30.2	14.1	QUAD_1
0.455	-0.42103	154.8094	154.4	20.4	QUAD_1
0.455	-0.42103	280.5882	280.2	383.4	QUAD_1
0.402	-37.8568	101.6208	63.8	48.2	QUAD_2
0.401	-38.5631	25.36599	0	24.9	QUAD_2
0.409	-32.9125	29.87074	0	14.1	QUAD_2
0.401	-38.5631	154.8094	116.2	20.4	QUAD_2
0.417	-27.2618	280.5882	253.3	383.4	QUAD_2
0.42	-25.1428	99.0962	74	83.6	QUAD_3
0.41	-32.2061	101.6208	69.4	48.2	QUAD_3
0.41	-32.2061	25.36599	0	24.9	QUAD_3
0.418	-26.5554	29.87074	3.3	14.1	QUAD_3
0.408	-33.6188	154.8094	121.2	20.4	QUAD_3
0.456	0.285305	101.6208	101.9	48.2	QUAD_4
0.456	0.285305	25.36599	25.7	24.9	QUAD_4
0.456	0.285305	29.87074	30.2	14.1	QUAD_4
0.455	-0.42103	154.8094	154.4	20.4	QUAD_4
0.456	0.285305	280.5882	280.9	383.4	QUAD_4
0.428	-19.4921	99.0962	79.6	83.6	QUAD_5
0.384	-50.5708	101.6208	51	48.2	QUAD_5
0.38	-53.3962	25.36599	0	24.9	QUAD_5
0.401	-38.5631	29.87074	0	14.1	QUAD_5
0.379	-54.1025	154.8094	100.7	20.4	QUAD_5

Keterangan	Error	Step	Akurasi	False Alarm	Miss
QUAD_1	4.206351761	22.8	86.6	21.4	42.1
QUAD_2	4.072522602	69.2	87.56	21.72	35.78
QUAD_3	4.079193639	50.8	87.32	21.98	36.84
QUAD_4	3.93998983	96.8	88.04	22.56	31.58
QUAD_5	3.522834157	7546	86.34	25.8	37.88

Variabel	Nilai	Skenario	Nilai	Skenario	Nilai	Skenario	Nilai	Skenario
Verifikasi	Tertinggi	MLP 1	Tertinggi	MLP 2	Tertinggi	MLP 3	Tertinggi	MLP 4
Akurasi	87.5	SINGLE_1 & SINGLE_3	88.4	DUAL_1.1	87.9	TRIPLE_3.3.3	88.04	QUAD_4.4.4
<i>False Alarm</i>	23.1	SINGLE_3	22.8	DUAL_1.1	21.8	TRIPLE_1.1.1	22.56	QUAD_4.4.4
<i>Miss</i>	33.3	SINGLE_1 & SINGLE_3	29.3	DUAL_1.1	31.9	TRIPLE_3.3.3	31.58	QUAD_4.4.4