

**PREDIKSI *PEAK GROUND ACCELERATION (PGA)* DI PULAU
JAWA MENGGUNAKAN METODE *ARTIFICIAL NEURAL
NETWORK***

TESIS

**Oleh:
SOFYAN HADI RAHMAWAN
NIM. 210605220014**



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

**PREDIKSI *PEAK GROUND ACCELERATION* (PGA) DI PULAU JAWA
MENGUNAKAN METODE *ARTIFICIAL NEURAL NETWORK***

TESIS

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

**Oleh:
SOFYAN HADI RAHMAWAN
NIM. 210605220014**

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

**PREDIKSI PEAK GROUND ACCELERATION (PGA) DI PULAU JAWA
MENGUNAKAN METODE ARTIFICIAL NEURAL NETWORK**

TESIS


**Oleh:
SOFYAN HADI RAHMAWAN
NIM. 210605220014**

**Telah Diperiksa dan Disetujui untuk Diuji
Tanggal : 26 November 2025**

Pembimbing I,


**Dr. Eddy Crysdiyan, M.Cs
NIP. 19740424 200901 1 008**

Pembimbing II,


**Prof. Dr. Sri Harini, M.Si
NIP. 19731014 200112 2 002**

**Mengetahui,
Kecamatan Program Studi Magister Informatika
Fakultas Sains dan Teknologi
Universitas Tulungagung Maulana Malik Ibrahim Malang**



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

**PREDIKSI *PEAK GROUND ACCELERATION* (PGA) DI PULAU JAWA
MENGUNAKAN METODE *ARTIFICIAL NEURAL NETWORK***

TESIS

**Oleh:
SOFYAN HADI RAHMAWAN
NIM. 210605220014**

Telah Dipertahankan di Depan Dewan Penguji
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
untuk Memperoleh Gelar Magister Komputer (M. Kom)
Tanggal : 26 November 2025

Susunan Dewan Penguji


Penguji I : Dr. M. Amin Hariyadi, M.T
NIP. 19670118 200501 1 001


Penguji II : Dr. Usman Pagalay, M.Si
NIP. 19650414 200312 1 001

Pembimbing I : Dr. Cahyo Crysdian, M.Cs
NIP. 19740424 200901 1 008

Pembimbing II : Prof. Dr. Sri Harini, M.Si
NIP. 19731014 200112 2 002

Tanda Tangan

()

()

()

()

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



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

PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan dibawah ini:

Nama : Sofyan Hadi Rahmawan
NIM : 210605220014
Program Studi : Magister Informatika (S-2)
Fakultas : Sains dan Teknologi
Judul Tesis : "Prediksi *Peak Ground Acceleration* (PGA) di Pulau Jawa
Menggunakan Metode *Artificial Neural Network*"

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

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

Malang, 26 November 2025
Yang membuat pernyataan,



Sofyan Hadi Rahmawan
NIM. 210605220014

MOTTO

“Jangan bersedih, sesungguhnya Allah bersama kita.”

“Jaga sholatmu, maka Allah akan memudahkan segala urusanmu.”

PERSEMBAHAN

Puji syukur atas kehadiran Allah SWT. Shalawat serta salam kepada Rasulullah SAW. Dengan segenap hati dan penuh hormat, penulis mempersembahkan Tesis ini kepada:

1. Orang tua, istri, dan anak-anak yang senantiasa mendoakan dan memberi semangat dalam menyelesaikan penulisan tesis ini.
2. Segenap civitas akademika UIN Maulana Malik Ibrahim Malang, khususnya Fakultas Sains dan Teknologi serta Program Studi Magister Informatika atas ilmu, bimbingan dan pengalaman berharga yang telah diberikan.
3. Pimpinan dan rekan-rekan BMKG Stasiun Geofisika Nganjuk yang selalu memberikan bantuan, motivasi, serta dukungan dalam penyelesaian tesis ini.
4. Seluruh rekan-rekan mahasiswa Magister Informatika UIN Maulana Malik Ibrahim, khususnya angkatan 5.

KATA PENGANTAR

Assalamualaikum Wr. Wr.

Syukur alhamdulillah penulis haturkan kehadiran Allah SWT yang telah melimpahkan Rahmat dan Hidayah-Nya, sehingga penulis dapat menyelesaikan studi di Program Studi Magister Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang sekaligus menyelesaikan Thesis ini dengan baik.

Selanjutnya penulis haturkan ucapan terimakasih seiring do'a dan harapan jazakumullah ahsanal jaza' kepada semua pihak yang telah membantu terselesaikannya Tesis ini. Ucapan terima kasih ini penulis sampaikan kepada:

1. Bapak Dr. Cahyo Crysdian, M.Cs dan Ibu Prof. Dr. Sri Harini, M.si selaku dosen pembimbing Tesis, yang telah banyak memberikan pengarahan dan pengalaman yang berharga.
2. Bapak Dr. M. Amin Hariyadi, M.T dan Bapak Dr. Usman Pagalay, M.Si selaku dosen penguji yang telah banyak memberikan pengarahan dan pengalaman yang berharga.
3. Segenap civitas akademika program Studi Magister Informatika, terutama seluruh Bapak/ibu dosen, terimakasih atas segenap ilmu dan bimbingannya.
4. Keluarga tercinta yang senantiasa memberikan doa dan semangat kepada penulis dalam menuntut ilmu.
5. Semua pihak yang ikut membantu dalam menyelesaikan Tesis ini baik berupa materiil maupun moril.

Penulis menyadari bahwa dalam penyusunan Tesis ini masih terdapat kekurangan dan penulis berharap semoga Tesis ini bisa memberikan manfaat kepada para pembaca khususnya bagi penulis secara pribadi. *Aamiin Ya Rabbal Alamin.*

Wassalamu'alaikum Wr. Wb.

Malang, 26 November 2025

Penulis

DAFTAR ISI

HALAMAN PENGAJUAN	ii
HALAMAN PERSETUJUAN	iii
HALAMAN PENGESAHAN.....	iv
HALAMAN PERNYATAAN.....	v
MOTTO.....	vi
PERSEMBAHAN.....	vii
KATA PENGANTAR.....	viii
DAFTAR ISI	ix
DAFTAR GAMBAR.....	xii
DAFTAR TABEL.....	xiv
ABSTRAK.....	xv
ABSTRAC.....	xvi
ستخلص ال	xvii
BAB 1	1
PENDAHULUAN	1
1.1 Latar Belakang	1
1.2. Pernyataan Masalah	5
1.3 Tujuan Penelitian.....	5
1.4 Manfaat Penelitian.....	5
1.5 Ruang Lingkup Penelitian.....	6
BAB 2.....	7
STUDI LITERATUR.....	7
2.1 Prediksi Peak Ground Acceleration (PGA).....	7
2.2 Kerangka Teori.....	12
BAB 3.....	16
STRATEGI PENELITIAN.....	16

3.1 Prosedur Penelitian.....	16
3.1.1 Studi Literatur.....	16
3.1.2 Pengumpulan Data.....	16
3.1.3 Desain Sistem.....	17
3.1.4 Uji Coba.....	18
3.1.5 Evaluasi Model.....	21
3.2 Instrumen Penelitian.....	22
BAB 4.....	23
METODE ANN SATU <i>HIDDEN LAYER</i>	23
4.1 Desain Model.....	23
4.2. Implementasi.....	29
4.3 Uji Coba.....	30
4.3.1 Komposisi Data 70 : 30.....	30
4.3.2 Komposisi Data 80 : 20.....	36
BAB 5.....	41
METODE ANN DUA <i>HIDDEN LAYER</i>	41
5.1 Desain Model.....	41
5.2. Implementasi.....	47
5.3 Uji Coba.....	48
5.3.1 Komposisi Data 70 : 30.....	48
5.3.2 Komposisi Data 80 : 20.....	54
5.4 Arsitektur ANN Mulai Dari Satu Hidden Layer.....	60
BAB 6.....	61
PEMBAHASAN.....	61
6.1 Perbandingan Iterasi ANN Satu Hidden Layer dan Dua Hidden Layer...	62
6.2 Perbandingan Waktu Konvergensi ANN Satu Hidden Layer dan Dua Hidden Layer.....	64
6.3 Perbandingan MSE ANN Satu Hidden Layer dan Dua Hidden Layer....	66

6.4 Perbandingan MSE ANN Mulai Dari Satu Hidden Layer	68
6.5 Analisis Model Terbaik Prediksi Peak Ground Acceleration (PGA).....	68
BAB 7.....	74
KESIMPULAN DAN SARAN.....	74
5.1 Kesimpulan.....	74
5.2 Saran.....	75
DAFTAR PUSTAKA.....	76
LAMPIRAN	79
1. Grafik Loss Function Proses Training Komposisi Data 70:30	80
2. Grafik Loss Function Proses Training Komposisi Data 80:20	95

DAFTAR GAMBAR

Gambar 1. 1 Faktor yang mempengaruhi nilai PGA.....	2
Gambar 2. 1 Kerangka Teori.....	13
Gambar 3. 1 Prosedur Penelitian.....	16
Gambar 3.2 Desain Sistem.....	17
Gambar 3.3 Instrumen Penelitian.....	22
Gambar 4.1 Desain ANN Satu Hidden Layer.....	24
Gambar 4.2 Flowchart Proses Pelatihan.....	25
Gambar 4.3 Perbandingan Iterasi ReLU dan tanh (Rasio 70:30).....	31
Gambar 4.4 Perbandingan Waktu Konvergensi ReLU dan tanh (Rasio 70:30).....	32
Gambar 4.5 Perbandingan MSE Testing ReLU dan tanh (Rasio 70:30).....	33
Gambar 4.6 Grafik loss function arsitektur HL1-A11	34
Gambar 4.7 Grafik loss function arsitektur HL1-A2	35
Gambar 4.8 Perbandingan Iterasi ReLU dan tanh (Rasio 80:20).....	36
Gambar 4.9 Perbandingan Waktu Konvergensi ReLU dan tanh (Rasio 80:20).....	38
Gambar 4.10 Perbandingan MSE Testing ReLU dan tanh (Rasio 80:20).....	38
Gambar 4.11 Grafik loss function arsitektur HL1-B11	39
Gambar 4.12 Grafik loss function arsitektur HL1-B5	40
Gambar 5.1 Desain ANN Dua Hidden Layer.....	42
Gambar 5.2 Perbandingan Iterasi ReLU dan tanh (Rasio 70:30).....	49
Gambar 5.3 Perbandingan Waktu Konvergensi ReLU dan tanh (Rasio 70:30).....	50
Gambar 5.4 Perbandingan MSE Testing ReLU dan tanh (Rasio 70:30).....	51
Gambar 5.5 Grafik loss function arsitektur HL2-A9	52
Gambar 5.6 Grafik loss function arsitektur HL2-A4	53
Gambar 5.7 Perbandingan Iterasi ReLU dan tanh (Rasio 80:20).....	55
Gambar 5.8 Perbandingan Waktu Konvergensi ReLU dan tanh (Rasio 80:20).....	56
Gambar 5.9 Perbandingan MSE Testing ReLU dan tanh (Rasio 80:20).....	57
Gambar 5.10 Grafik loss function arsitektur HL2-B10	58
Gambar 5.11 Grafik loss function arsitektur HL2-B1	59

Gambar 6.1 Perbandingan Iterasi ANN Satu Hidden Layer dan dua.Hidden Layer	62
Gambar 6.2 Perbandingan Waktu Konvergensi ANN Satu Hidden Layer dan Dua Hidden Layer.....	64
Gambar 6.3 Perbandingan MSE satu Hidden Layer dan dua Hidden Layer	66
Gambar 6.4 Grafik MSE Mulai Dari Satu Hidden Layer	68
Gambar 6.5. Perbandingan PGA aktul dan PGA prediksi model HL2-A9.....	70
Gambar 6.6. Distribusi Residual PGA aktul dan PGA prediksi model HL2-A9	71

DAFTAR TABEL

Tabel 2. 1 Studi Literatur	14
Tabel 2. 2 Tabel 2.2. Peringkat Jumlah Penerapan Metode Machine Learning.....	15
Tabel 3. 1 Data Magnitudo, Jarak Hiposenter, Kecepatan dan PGA	17
Tabel 3.2 Nama Pelatihan ANN Satu Hidden Layer	20
Tabel 3.3 Nama Pelatihan ANN Dua Hidden Layer	21
Tabel 4.1 Parameter Model ANN Satu Hidden Layer.....	29
Tabel 4.2 Hasil pelatihan dan pengujian komposisi data 70:30	30
Tabel 4.3 Hasil pelatihan dan pengujian komposisi data 80:20.....	36
Tabel 5.1 Parameter Model ANN Dua Hidden Layer	47
Tabel 5.2 Hasil pelatihan dan pengujian komposisi data 70:30	48
Tabel 5.3 Hasil pelatihan dan pengujian komposisi data 80:20	54
Tabel 5.4 Hasil Uji Coba Mulai Dari Satu Hidden Layer	60
Tabel 6.1 Hasil performa terbaik ANN 1 hidden layer dan 2 hidden layer.	61

ABSTRAK

Rahmawan, S. H. 2025. **Prediksi *Peak Ground Acceleration* (PGA) DI Pulau Jawa Menggunakan Metode *Artificial Neural Network***. Tesis. Program Studi Magister Informatika, Fakultas Sain dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim, Malang. Pembimbing: (I). Dr. Cahyo Crysdian, (II). Prof. Dr. Sri Harini, M.Si.

Pulau Jawa memiliki tingkat seismisitas yang tinggi akibat pengaruh zona subduksi Lempeng Samudra Hindia–Eurasia serta keberadaan berbagai sesar aktif di daratan. Data BMKG periode 2009–2024 menunjukkan tren peningkatan aktivitas gempa bumi, yang berdampak pada meningkatnya potensi risiko bencana. Dalam konteks mitigasi, *Peak Ground Acceleration* (PGA) merupakan parameter penting dalam perencanaan bangunan tahan gempa, sehingga diperlukan metode prediksi yang akurat. Penelitian ini bertujuan untuk mengevaluasi strategi terbaik dalam memprediksi nilai *Peak Ground Acceleration* (PGA) berdasarkan metode *Artificial Neural Network* (ANN), serta menganalisis pengaruh variasi arsitektur jaringan, jumlah neuron pada *hidden layer* dan pemilihan fungsi aktivasi terhadap performa model. Parameter input yang digunakan adalah magnitudo, jarak hiposenter dan V_{s30} . Hasil menunjukkan bahwa ANN dua *hidden layer* memberikan performa terbaik dengan nama pelatihan HL2-A9. Arsitektur ini memiliki konfigurasi neuron (75–75) dan fungsi aktivasi ReLU dengan nilai MSE 0.2260 serta koefisien korelasi 0,699 yang menunjukkan hubungan linear positif cukup kuat antara PGA aktual dan prediksi.

Kata kunci: *Artificial Neural Network* (ANN), *hidden layer*, *prediksi*, *gempa bumi*, *Peak Ground Acceleration* (PGA).

ABSTRACT

Rahmawan, S. H. 2025. **Prediction of Peak Ground Acceleration (PGA) on Java Island Using Artificial Neural Network Methods**. Thesis Master's Program in Informatics, Faculty of Science and Technology, Islamic State University Maulana Malik Ibrahim Malang. Advisors: (I) Dr. Cahyo Crysdiyan, (II) Prof. Dr. Sri Harini, M.Si.

Java Island has a high level of seismicity due to the influence of the Indo-Australian–Eurasian oceanic plate subduction zone and the presence of various active faults on land. BMKG data from 2009–2024 indicate an increasing trend in earthquake activity, which in turn elevates potential disaster risk. In the context of mitigation, Peak Ground Acceleration (PGA) is an important parameter in earthquake-resistant building design, making accurate prediction methods essential. This study aims to evaluate the optimal strategy for predicting Peak Ground Acceleration (PGA) using the Artificial Neural Network (ANN) method, as well as to analyze the effects of varying network architectures, numbers of neurons in hidden layers, and activation function selection on model performance. The input parameters used are magnitude, hypocenter distance, and V_{s30} . The results show that the ANN with two hidden layers provides the best performance, specifically the HL2-A9 training model. This architecture employs a (75–75) neuron configuration and the ReLU activation function, producing an MSE value of 0.2260 and a correlation coefficient of 0.699, indicating a moderately strong positive linear relationship between actual and predicted PGA values.

Keywords: Artificial Neural Network (ANN), hidden layer, prediction, earthquake, Peak Ground Acceleration (PGA).

مستخلص البحث

رحماوان، س. هـ. 2025. التنبؤ بذروة التسارع الأرضي (PGA) في جزيرة جاوى باستخدام طريقة الشبكة العصبية الاصطناعية. رسالة الماجستير. قسم المعلومات، كلية العلوم والتكنولوجيا بجامعة مولانا مالك إبراهيم الإسلامية الحكومية مالانج. المشرف الأول: د. جهيو كريسديان، الماجستير. المشرف الثاني: أ. د. سري هارينى، الماجستير.

تمتلك جزيرة جاوى مستوى عالٍ من النشاط الزلزالي نتيجة لتأثير منطقة اندساس صفيحة المحيط الهندي-أوراسيا وكذلك وجود العديد من الصدوع النشطة على اليابسة. تظهر بيانات وكالة الأرصاد الجوية والمناخ والجيوفيزياء (BMKG) للفترة 2009-2024 اتجاهًا لزيادة نشاط الزلازل، مما يؤثر على ارتفاع احتمالية خطر الكوارث. في سياق التخفيف، تُعتبر ذروة تسارع الأرضي (PGA) من المعايير المهمة في تخطيط المباني المقاومة للزلازل، وبالتالي هناك حاجة إلى طرق تنبؤية دقيقة. تهدف هذه الرسالة إلى تقييم أفضل الاستراتيجيات في التنبؤ بقيمة ذروة التسارع الأرضي بناءً على طريقة الشبكة العصبية الاصطناعية (ANN)، وكذلك تحليل تأثير اختلافات بنية الشبكة وعدد الخلايا العصبية في الطبقات المخفية واختيار وظيفة التنشيط على أداء النموذج. تشمل معلمات الإدخال المستخدمة: المقدار وعمق البؤرة وVs30. أظهرت النتائج أن الشبكة العصبية ذات طبقتين مخفيتين تقدم أفضل أداء باسم تدريب HL2-A9. تمتلك هذه الهندسة العصبية تكوينًا للخلية العصبية (75-75) ووظيفة تنشيط ReLU بقيمة MSE تبلغ 0.2260 ومعامل ارتباط 0.699 مما يشير إلى وجود علاقة خطية موجبة قوية نسبيًا بين ذروة التسارع الأرضي الفعلي والتنبؤي.

الكلمات الرئيسية: شبكة عصبية اصطناعية، طبقة مخفية، تنبؤ، زلزال، ذروة تسارع أرضي.

BAB 1

PENDAHULUAN

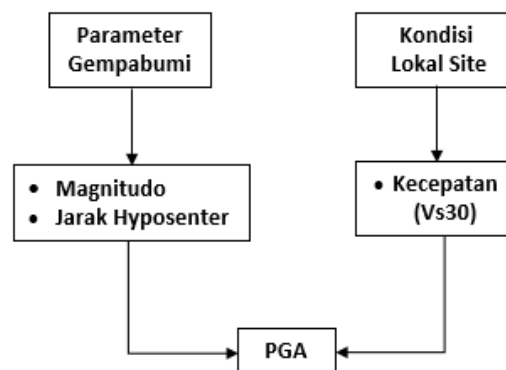
1.1 Latar Belakang

Pulau Jawa merupakan salah satu pulau di Indonesia yang sering mengalami gempa bumi. Seismotektonik Pulau Jawa dapat dibagi menjadi dua lajur, yakni lajur seismotektonik subduksi selatan Jawa dan lajur seismotektonik sesar aktif daratan Jawa (Soehaimi, 2008). Dampak dari subduksi pada Pulau Jawa menyebabkan terjadinya sesar-sesar lokal di daratan Pulau Jawa diantaranya: Sesar Cimandiri, Sesar Lembang, Sesar Baribis, Sesar Citanduy dan Sesar Opak. Beberapa ahli juga berpendapat adanya sesar Kendeng yang berarah barat-timur di kawasan pantai utara Jawa Tengah hingga Jawa Timur (Smyth dkk., 2008). Hasil monitoring aktivitas gempa bumi yang dilakukan oleh BMKG selama periode 2009–2024 menunjukkan adanya tren peningkatan aktivitas gempa bumi secara signifikan sejak tahun 2013, khususnya di pulau Jawa.

Berdasarkan kondisi tektonik yang kompleks serta adanya tren peningkatan aktivitas seismik yang terjadi di pulau Jawa, maka hal ini menunjukkan pentingnya dilakukan upaya mitigasi bencana gempa bumi. Salah satu langkah strategis dalam mitigasi tersebut adalah penyusunan peta bahaya gempa (*seismic hazard map*) yang berfungsi sebagai dasar perencanaan tata ruang wilayah dan pemetaan lokasi risiko bahaya gempa bumi. Dalam proses penyusunan peta bahaya gempa, *Peak Ground Acceleration* (PGA) merupakan salah satu parameter fundamental yang digunakan untuk menggambarkan tingkat percepatan maksimum gerakan tanah akibat gempa bumi. PGA sangat diperlukan dalam perhitungan perencanaan struktur bangunan, evaluasi risiko seismik, serta penentuan standar ketahanan bangunan tahan gempa. Sehingga akurasi

prediksi nilai PGA memegang peranan krusial dalam meningkatkan ketepatan analisis bahaya gempa di suatu wilayah.

Hasil rekaman seismograf dari stasiun terdekat dapat dijadikan dasar menghitung nilai PGA, namun metode estimasi yang handal juga berguna untuk memberikan informasi yang lebih rinci mengenai karakteristik dan gerakan gempa tersebut (Kreh T., Chaw D. 2002). Parameter yang dipertimbangkan dalam menentukan prediksi PGA adalah parameter gempa bumi, seperti magnitudo dan jarak hiposenter. Selain itu, nilai PGA juga dipengaruhi oleh kondisi lokal *site*. “Tipe gempabumi dan kondisi lokal *site* yang berbeda akan menghasilkan gerakan tanah yang berbeda meskipun memiliki magnitudo dan jarak yang sama” (Zhao *et al.*, 2006). Lebih jelasnya bisa dilihat di Gambar 1.1.



Gambar 1.1. Faktor yang mempengaruhi nilai PGA

Dengan kemajuan teknologi informasi dan kecerdasan buatan, khususnya *Artificial Neural Networks* (ANN), upaya peningkatan akurasi *prediksi Peak Ground Acceleration* (PGA) semakin berkembang. ANN mampu mempelajari pola nonlinier dan kompleks dalam data seismik, sehingga sangat sesuai untuk memodelkan variabilitas gerakan tanah yang dipengaruhi oleh magnitudo, jarak hiposenter, dan kondisi lokal *site* (Vs30). Berbagai penelitian, seperti Khosravikia *et al.* (2018), menunjukkan keunggulan

metode ANN dalam memprediksi gerakan tanah (PGA) dibandingkan pendekatan empiris tradisional.

Sementara itu, metode konvensional seperti Ground Motion Prediction Equation (GMPE) masih mengandalkan formulasi matematis yang relatif linier dan kurang fleksibel untuk menangkap kerumitan hubungan antarparameter seismik, terutama pada wilayah dengan kondisi tektonik kompleks seperti Pulau Jawa. Oleh karena itu, ANN menjadi alternatif yang lebih adaptif karena tidak memerlukan asumsi bentuk hubungan tertentu dan mampu mengekstraksi pola nonlinier secara efektif. Berdasarkan kebutuhan tersebut, penelitian ini difokuskan pada pengembangan dan evaluasi berbagai arsitektur ANN untuk prediksi PGA di Pulau Jawa. Variasi arsitektur jaringan, jumlah neuron, komposisi data training–testing, dan jenis fungsi aktivasi dianalisis untuk menentukan konfigurasi model yang paling optimal.

Penelitian ini menggunakan data gempa dirasakan di pulau Jawa selama periode Agustus 2020 hingga Januari 2024 yang bersumber dari katalog gempa bumi BMKG. Jumlah gempa dirasakan sebanyak 137 kejadian yang sebagian besar terjadi di laut selatan pulau Jawa. Gempa tersebut bermagnitudo antara 2,1 hingga 6,7 dengan skala intensitas dirasakan antara II MMI (Getaran dirasakan oleh beberapa orang, benda-benda ringan yang digantung bergoyang) hingga V MMI (Getaran dirasakan oleh hampir semua penduduk, orang banyak terbangun, gerabah pecah, barang-barang terpelanting dan tiang-tiang besar tampak bergoyang).

Gempa bumi merupakan bencana alam yang berpotensi menyebabkan berbagai bentuk kerusakan dan korban jiwa. Ada beberapa ayat Al-Qur'an yang membahas tentang gempa bumi yang menggambarkan besarnya kekuasaan Allah SWT. Dalam surah Al-Hajj ayat 1, Allah SWT berfirman :

يَا أَيُّهَا النَّاسُ اتَّقُوا رَبَّكُمُ إِنَّ زَلْزَلَةَ السَّاعَةِ شَيْءٌ عَظِيمٌ

Artinya : "Wahai manusia, bertakwalah kepada Tuhanmu. Sesungguhnya guncangan hari kiamat itu adalah sesuatu yang sangat besar". (QS. Al-Hajj: 1).

Menurut *Tafsir Al-Madinah Al-Munawwarah*, Syaikh Prof. Dr. Imad Zuhair Hafidz, QS. Al-Hajj ayat 1, Allah memerintahkan manusia agar bertakwa kepada Allah dengan mentaati segala perintah-Nya dan menjauhi segala larangan-Nya. Dan Dia menyampaikan pada hari kiamat mereka akan menghadapi goncangan bumi yang dahsyat, dan kejadiannya sangat mengerikan. Ayat tersebut menghimbau umat manusia agar mawas diri serta menjaga dirinya dari azab Allah dengan mengikuti perintah-Nya dan menjauhi larangan-Nya. Perintah tersebut berlaku sejak ayat ini diturunkan sampai datangnya hari kiamat kelak dengan ditandai oleh terjadinya gempa bumi yang amat dahsyat menghancurkan seluruh isi jagat raya.

Dalam surah Az-Zalzalah ayat 1 Allah SWT juga berfirman :

إِذَا زُلْزِلَتِ الْأَرْضُ زِلْزَالَهَا

Artinya: "Apabila bumi diguncangkan dengan guncangan yang dahsyat," (QS. Az-Zalzalah: 1).

Menurut *Tafsir Al-Madinah Al-Munawwarah* karya Syaikh Prof. Dr. Imad Zuhair Hafidz, QS. Az-Zalzalah ayat 1 menggambarkan peristiwa goncangan bumi yang sangat dahsyat pada hari kiamat sebagai bagian dari kehancuran alam semesta dan awal kebangkitan manusia untuk menghadapi hisab di hadapan Allah SWT. Ayat ini menunjukkan bahwa peristiwa gempa bumi pada hari kiamat merupakan fenomena kosmik yang luar biasa kuat dan bersifat final bagi kehidupan dunia. Fenomena meningkatnya aktivitas kegempaan khususnya di pulau Jawa ini, merupakan salah satu tanda Allah SWT

memberikan peringatan kepada umat manusia bahwa hari kiamat pasti akan terjadi yang ditandai dengan terjadinya gempa bumi yang sangat dahsyat.

1.2 Pernyataan Masalah

1. Strategi apakah yang paling optimal dalam memprediksi nilai *Peak Ground Acceleration* (PGA) berdasarkan metode *Artificial Neural Network* (ANN) ?
2. Bagaimana pengaruh variasi arsitektur jaringan, jumlah *neuron* pada *hidden layer* dan jenis fungsi aktivasi terhadap performa metode ANN dalam memprediksi nilai PGA di pulau Jawa?

1.3 Tujuan Penelitian

1. Mengevaluasi strategi terbaik dalam memprediksi nilai *Peak Ground Acceleration* (PGA) berdasarkan metode *Artificial Neural Network* (ANN)
2. Menganalisis pengaruh variasi arsitektur jaringan, jumlah *neuron* pada *hidden layer* dan pemilihan fungsi aktivasi terhadap performa metode ANN dalam memprediksi nilai PGA di pulau Jawa.

1.4 Manfaat Penelitian

1. Dapat dimanfaatkan oleh *stakeholder* terkait dalam memperkirakan potensi tingkat guncangan dan kerusakan akibat gempa bumi, sehingga mendukung pengambilan keputusan dalam perencanaan mitigasi
2. Dapat dijadikan referensi oleh para ahli teknik sipil dalam perencanaan dan perancangan bangunan tahan gempa.
3. Dapat menambah referensi ilmiah dalam pengembangan model prediksi *Peak Ground Acceleration* (PGA) berbasis *Artificial Neural Network* (ANN)

1.5 Ruang Lingkup Penelitian

1. Penelitian hanya menggunakan data gempa dirasakan yang pernah terjadi di pulau Jawa selama periode Agustus 2020 hingga Januari 2024.
2. Data *Peak Ground Acceleration* (PGA) yang digunakan berasal dari sensor *accelerometer* BMKG yang ada di pulau Jawa
3. Metode prediksi hanya menggunakan metode *Artificial Neural Network* (ANN) dengan jaringan satu *hidden layer* dan dua *hidden layer*.
4. Data input terdiri dari data magnitudo, jarak hiposenter dan kecepatan gelombang geser pada kedalaman 30 km (V_{s30}). Sedangkan data target adalah data *Peak Ground Acceleration* (PGA).

BAB 2

STUDI LITERATUR

2.1 Prediksi Peak Ground Acceleration (PGA)

Hong Hanping et al. (2012) memprediksi percepatan pseudospektral (PSA) dan *Peak Ground Acceleration* (PGA) menggunakan model ANN dengan metode backpropagation feed-forward. Data yang digunakan sekitar 600 rekaman dari 39 gempa bumi di California. Statistik residual atau kesalahan pemodelan untuk model berbasis ANN yang dilatih hampir sama dengan statistik untuk persamaan prediksi gerakan tanah parametrik, yang diperoleh melalui analisis regresi. Kesalahan pemodelan untuk model berbasis ANN dapat dimodelkan sebagai variabel normal, dan terdapat korelasi (linier) yang dapat diabaikan antara kesalahan pemodelan dan parameter input .

Dhanya dan Raghukanth (2017) mengembangkan persamaan prediksi gerakan tanah berdasarkan algoritma ANN untuk gempa bumi kerak dangkal. Teknik hibrida yang menggabungkan algoritma genetika dan teknik Levenberg–Marquardt digunakan untuk melatih model. Model saat ini dikembangkan untuk memprediksi *Peak Ground Velocity* (PGV), *Peak Ground Acceleration* (PGA), dan Spektral Acceleration (SA). Parameter input untuk prediksi adalah besaran momen (M_w), jarak terdekat ke bidang patahan (R_{rup}), kecepatan gelombang geser di wilayah tersebut (V_{s30}) dan mekanisme fokus (F). Hasilnya menunjukkan bahwa algoritma ANN memiliki akurasi 94,9 % dapat digunakan sebagai algoritma prediktif PGA. Dalam penelitian ini ditemukan sebanding dengan hubungan yang ada dalam basis data global. Model ANN yang dikembangkan mampu menangkap karakteristik yang menonjol dengan GMPE dan dapat memperkirakan spektrum respons spesifik lokasi untuk kota Shimla yang terletak di wilayah Himalaya.

Peak Ground Acceleration (PGA) dihitung untuk memprediksi potensi kerusakan akibat gempa bumi di lokasi rekonstruksi. Zadeh Mostafa A, et.al. (2018) dalam penelitiannya menggunakan algoritma pelatihan berdasarkan penurunan gradien dengan menggunakan rekaman gerakan tanah yang kuat. Algoritma ANN menunjukkan bahwa kecocokan antara gerakan tanah kuat yang diprediksi oleh jaringan dan nilai PGA yang diamati mampu menghasilkan koefisien korelasi tinggi. Metode didefinisikan dengan menggunakan sistem inferensi fuzzy berdasarkan jaringan adaptif, jaringan saraf umpan maju (FFBP) dengan empat parameter dasar sebagai variabel input yang memengaruhi gempa bumi di wilayah yang diteliti. Indeks gempa bumi yang terpengaruh meliputi besaran momen magnitudo, jarak sumber gempa, mekanisme patahan, dan kondisi lokasi (V_s30). Hasilnya model ini dapat sesuai memprediksi *Peak Ground Acceleration* (PGA) untuk gempa bumi di masa mendatang dengan akurasi sebesar 99,23 %.

Khosravikia et.al. (2018) menggunakan ANN untuk mengembangkan model prediksi gerakan tanah (PGA dan PGV) berdasarkan magnitudo, jarak sumber gempa ke lokasi dan kecepatan gelombang geser di wilayah Amerika Utara, Tengah dan Timur. Akurasi yang didapat dari pembelajaran model ini adalah 90 % untuk prediksi PGA dan 92 % untuk prediksi PGV. Hasil dari penelitian ini menunjukkan bahwa model prediksi gerakan tanah yang dikembangkan menggunakan model ANN memberikan prediksi yang jauh lebih akurat dari ukuran intensitas gerakan tanah pada semua jarak dan besaran.

Khosravikia et.al. (2021) membandingkan algoritma machine learning dan regresi linear konvensional dalam memprediksi gerakan tanah (PGA) berdasarkan karakteristik sumber, jarak sumber ke lokasi dan kondisi lokasi setempat. Hasilnya menunjukkan bahwa ketika data yang tersedia cukup semua algoritma machine learning memberikan estimasi yang lebih akurat dibandingkan regresi linear konvensional. Untuk sesama

algoritma machine learning, Random Forest lebih unggul dibandingkan ANN dan SVM dengan akurasi sebesar 95,8 %.

Junjie (2021) membandingkan kinerja persamaan atenuasi regresi linear sederhana dengan *machine learning* (*Random Forest* dan *Neural Network*) dalam memprediksi *Peak Ground Velocity* (PGV). Hasilnya kinerja prediksi persamaan atenuasi regresi linear tidak sebaik prediktor *machine learning*. Hal itu dikarenakan *machine learning* sepenuhnya merupakan model prediktif yang digerakkan oleh dataset. Untuk algoritma *machine learning* *Neural Network* lebih baik dari pada *Random Forest* dengan akurasi sebesar 60 %.

Model gerakan tanah telah memperoleh perhatian utama selama beberapa tahun terakhir karena mampu memprediksi tingkat intensitas gerakan tanah untuk skenario seismik di masa mendatang. Model ini sangat bermanfaat untuk memperkirakan bahaya seismik dan selalu menuntut penyempurnaan untuk meningkatkan kualitas peta bahaya seismik. Prezioso et al. (2022) mengusulkan model prediksi gerakan tanah yang disebabkan oleh gempa bumi yang tercatat di wilayah panas bumi *The Geysers* dengan menggunakan model ANN. Kinerja dari model ini menghasilkan akurasi sebesar 82,1 %. Hasil yang diperoleh dalam penelitian ini menunjukkan peningkatan moderat dalam prediksi gerakan tanah dan fitur pemodelan yang tidak diperhitungkan oleh model empiris.

Mandal (2022) menggunakan algoritma XGBoost, yang merupakan algoritma *Machine Learning* ansambel berbasis *decission tree* dan menggunakan kerangka peningkatan gradien untuk memprediksi PGA dengan parameter input Magnitudo, kedalaman fokus, jarak episenter, dan Vs30) di India dan Nepal, menunjukkan akurasi yang sangat baik, yaitu 99,4 %. Pemodelan machine learning supervise XGBoost terhadap data PGA yang diamati dari gempa bumi India dan Nepal mengungkapkan

bahwa pemodelan ML dapat meningkatkan prediksi nilai PGA secara signifikan dan dapat memberikan GMPE yang jauh lebih baik untuk wilayah mana pun. Melalui pemodelan *machine Learning* akan menjadi langkah awal dalam membangun bangunan yang lebih tahan gempa, mengurangi jumlah korban tewas akibat terjadinya gempa bumi dan membangun masyarakat yang tangguh terhadap gempa di India dan sekitarnya.

Model gerakan tanah konvensional banyak ditetapkan di seluruh dunia berdasarkan analisis regresi konvensional. Sebagai alternatif, (Mohammadi et al., 2023) menggunakan algoritma machine learning ANN dan peningkatan gradien ekstrem (XGBoost) dalam memprediksi *Peak Ground Acceleration* (PGA), kecepatan tanah puncak (PGV) dan percepatan pseudo-spektral (PSA) di negara Turki. Hasilnya menunjukkan bahwa pendekatan XGBoost memberikan model yang tidak bias dengan korelasi yang lebih tinggi dan residual yang lebih rendah dari pada ANN dengan akurasi sebesar 92 %.

Pendekatan *machine learning* dalam mengembangkan *Ground Motion Model* (GMM) semakin meningkat, khususnya untuk wilayah yang aktif secara tektonik seperti California, di mana banyak data percepatan tanah yang tersedia. Teknik *machine learning* ini sepenuhnya berbasis data dapat mempelajari tren linier dan/atau nonlinier yang kompleks dalam data berdimensi tinggi. *machine learning* adalah bagian dari kecerdasan buatan yang memungkinkan komputer untuk belajar hubungan dan pola dalam data tanpa memerlukan bentuk fungsional yang telah ditentukan sebelumnya. Sedaghati dan Pezeshk (2023) menggunakan empat algoritma pembelajaran mesin (ML) terawasi nonparametrik yang berbeda termasuk ANN, *Kernel-Ridge Regressor* (KRR), *Random Forest Regressor* (RFR), dan *Support Vector Regressor* (SVR) untuk membangun empat model gerakan tanah (PGA, PGV, PGD dan PSA). Data yang digunakan berasal dari *Pacific Earthquake Engineering Research Center* (PEER) NGA2 USA. Parameter input model adalah magnitudo, jarak rupture, kecepatan Vs30 dan kedalaman rupture. Hasil yang diperoleh

dengan membandingkan empat metode machine learning tersebut, *Random Forest* menghasilkan akurasi yang lebih baik dengan akurasi 90,26 %.

Prediksi *Peak Ground Acceleration* (PGA) sangat penting dalam desain teknik seismik struktur. Pembelajaran mesin adalah metode baru untuk memprediksi PGA dan memang memiliki beberapa kelebihan. Percepatan batuan dasar puncak, frekuensi dominan, kedalaman tanah saat kecepatan gelombang geser mencapai 800 m/s (D800), dan kecepatan gelombang geser batuan dasar (Bedrock Vs) digunakan sebagai masukan untuk memprediksi PGA. Sun Rui et al. (2023) membangun model *XGBoost* (*eXtreme Gradient Boosting*), *random forest*, dan *decision tree* untuk memprediksi PGA. Hasil prediksi dibandingkan dengan hasil simulasi numerik. Pengaruh antara fitur masukan dan hasil prediksi model dianalisis dengan nilai SHAP (*SHapley Additive exPlanations*). Hasilnya menunjukkan bahwa akurasi dari dataset pelatihan dan dataset pengujian masing-masing mencapai 94,5 % dan 91,5 %. Hasil prediksi model XGBoost lebih baik daripada *Random Forest* dan *Decision Tree*.

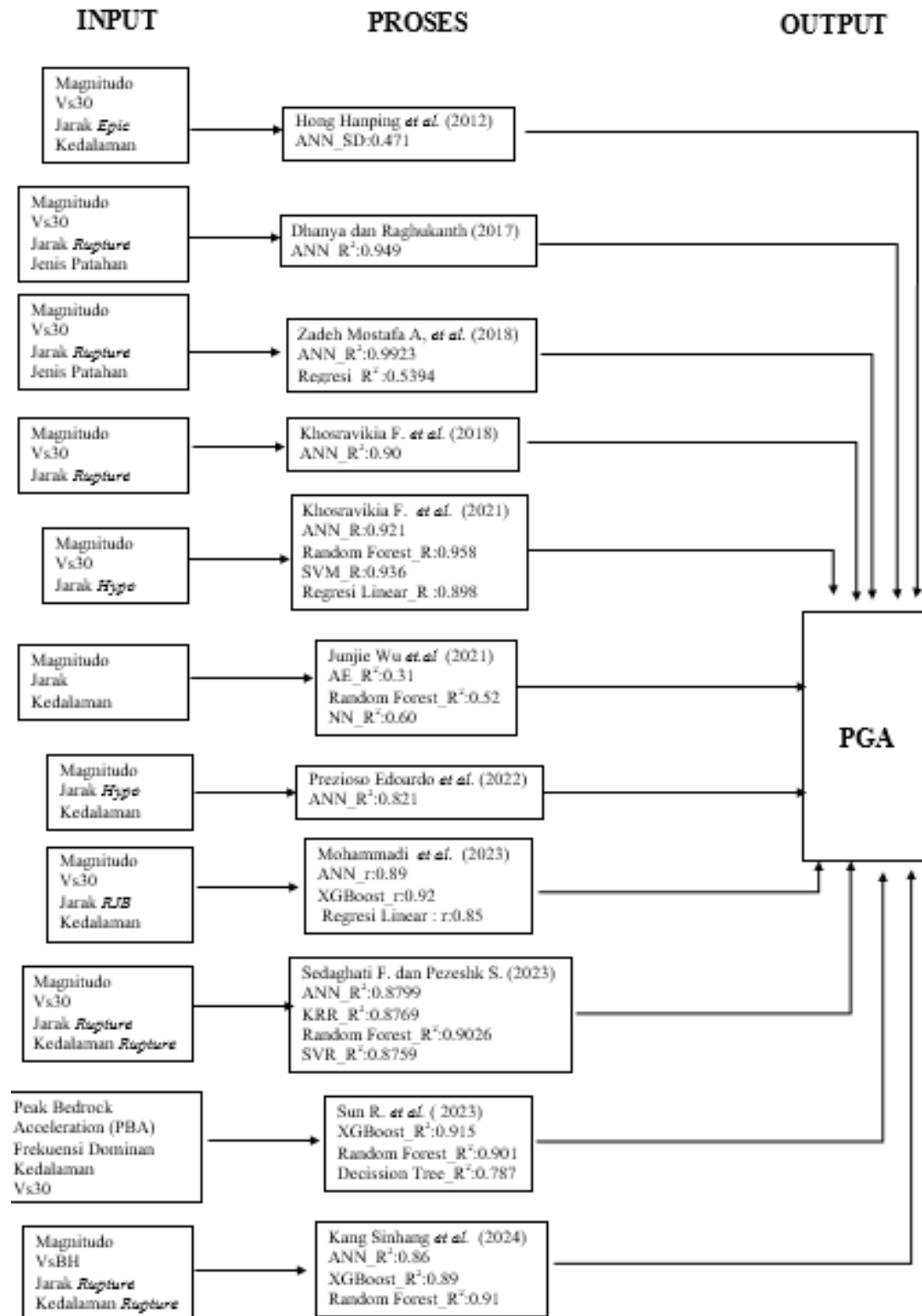
Luzi et al. (2023) meneliti kelebihan dan kekurangan penggunaan bentuk fungsional versus algoritma machine learning dalam model gerakan tanah. Variabel yang digunakan dalam penelitian ini adalah magnitudo, jarak, kedalaman hiposenter, jenis patahan, dan kecepatan VS30. Algoritme ML dengan kemampuan prediksi terbaik adalah Regresi Proses Gaussian yang secara signifikan mengurangi standar deviasi prediksi. Model berbasis regresi tradisional menunjukkan kinerja yang lebih baik daripada algoritma machine learning ketika jumlah pengamatan terbatas karena model tersebut mengompensasi kekurangan data dengan persamaan linier atau nonlinier yang telah ditentukan sebelumnya. Sedangkan pendekatan Machine Learning menunjukkan kinerja yang lebih baik ketika data tersedia dalam jumlah besar. Penggunaan variabel yang tidak terkait langsung dengan fenomena berbasis fisika (seperti kedalaman kejadian, koordinat

stasiun seismik, backazimuth stasiun, atau lainnya) bisa sangat membantu dalam meningkatkan kinerja prediksi *machine learning*.

Kang Sinhang et al. (2024) mengembangkan *Ground Motion Model* (GMM) untuk beberapa tipe tipe gempa dengan memperhatikan perbedaan karakteristik antara gempa utama dan gempa susulan. Data yang digunakan adalah gerakan tanah Jepang yang kuat, yang tercatat pada batuan di dalam lubang bor. Metode yang diterapkan metode *machine learning ansambel* dan ANN untuk meningkatkan akurasi GMM. Ketika data yang sama diterapkan, hasilnya kinerja prediksi GMM yang diusulkan metode *ansambel* (*Random Forest dan Gradien Boosting*) relatif tidak bias, dan varian residunya kecil untuk semua jenis gempa dibandingkan dengan gempa berbasis GMM pada ANN. Kinerja metode Random Forest lebih baik daripada Gradient Boosting dan ANN dengan akurasi sebesar 0,91 %. Namun untuk GMM berdasarkan tiga metode *machine learning* (*Random Forest dan Gradien Boosting dan ANN*) memiliki kinerja yang lebih baik jika dibandingkan metode berbasis analisis regresi.

2.2 Kerangka Teori

Gambar 2.2 disajikan kerangka teori berdasarkan beberapa penelitian sebelumnya dalam kurun waktu 2012 hingga 2024 yaitu tentang prediksi *Peak Ground Acceleration* (PGA), dengan mengkategorikan parameter atau variabel sumber gempa dan lokal site sebagai data input, penggunaan metode dan output yang diperoleh dari hasil kerangka teoritis.



Gambar 2.1. Kerangka Teori Penelitian

Dalam penelitian ini kinerja setiap metode dievaluasi berdasarkan tingkat akurasi yang dinyatakan dalam persentase. Semakin mendekati nilai 100%, semakin baik tingkat ketepatan hasil yang diperoleh. Zadeh Mostafa A. et al. (2018) dalam penelitiannya menunjukkan bahwa metode *Artificial Neural Network (ANN)* mampu mencapai tingkat akurasi tertinggi, yaitu sebesar 99,23%. Rincian urutan kriteria kinerja tersebut disajikan pada Tabel 2.1.

Tabel 2.1. Peringkat Akurasi Kinerja

Rangking	Penulis	Metode	Akurasi
1	Zadeh Mostafa A, et.al. (2018)	ANN	99,23 %
2	Khosravikia F. et.al (2021)	Random Forest	95,8 %
3	Dhanya dan Raghukanth (2017)	ANN	94,9 %
4	Mohammadi et al. (2023)	XGBoost	92.0 %
5	Sun Rui et al. (2023)	XGBoost	91,5 %
6	Kang Sinhang (2024)	Random Forest	91,0 %
7	Khosravikia et.al. (2018)	ANN	90 %
8	Sedaghati F. dan Pezeshk S. (2023)	Random Forest	90 %
9	Prezioso et al. (2022)	ANN	82,1 %
10	Junjie Wu et al. (2021)	ANN	60 %
11	Hanping et.al (2012)	ANN	-

Selain itu, berdasarkan kerangka teori juga diperoleh informasi mengenai metode *machine learning* yang paling sering digunakan oleh peneliti dalam memprediksi nilai PGA. ANN merupakan metode yang paling sering digunakan dalam memprediksi PGA. Urutan metode yang paling sering digunakan diberikan pada Tabel 2.2.

Tabel 2.2. Peringkat Jumlah Penerapan Metode *Machine Learning*

No	Metode	Jumlah
1	ANN	6
2	Random Forest	3
3	XGBoost	2

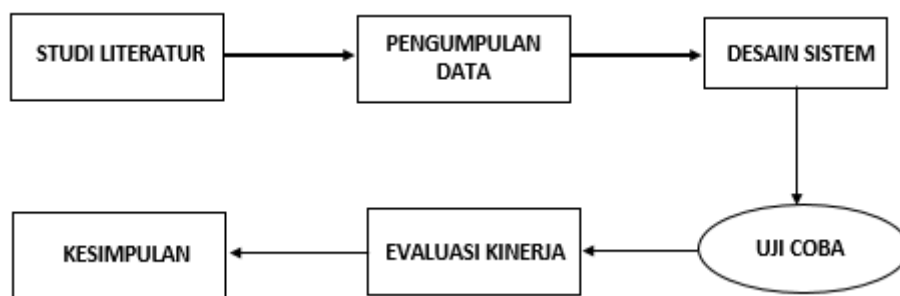
Berdasarkan peringkat kinerja dan metode yang paling sering digunakan dalam penelitian prediksi PGA, maka dalam penelitian ini metode ANN yang nantinya akan digunakan dalam menentukan prediksi nilai PGA.

BAB 3

STRATEGI PENELITIAN

3.1 Prosedur Penelitian

Prosedur penelitian dalam memprediksi *Peak Ground Acceleration (PGA)* disajikan pada Gambar 3.1, menjelaskan tentang tahapan-tahapan penelitian yang ditempuh untuk memperoleh data serta menjawab rumusan masalah. Penelitian ini mencakup kegiatan studi literatur, proses pengumpulan data, desain sistem, uji coba, serta evaluasi terhadap kinerja model yang dihasilkan.



Gambar 3.1 Prosedur Penelitian

3.1.1 Studi Literatur

Studi literatur menjelaskan mengenai penelitian sebelumnya terkait prediksi PGA. Dalam hal ini penulis menggunakan referensi penelitian terkait prediksi PGA dalam kurun waktu 2012 hingga 2024.

3.1.2 Pengumpulan Data

Dalam penelitian ini menggunakan data sekunder yang diperoleh dari Badan Meteorologi Klimatologi dan Geofisika (BMKG) dan United States Geological Survey

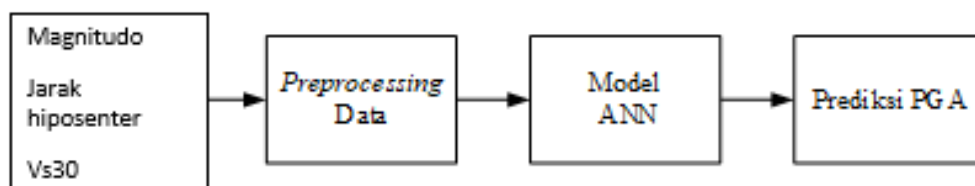
(USGS). Data yang digunakan dalam penelitian ini adalah data parameter gempa bumi, kondisi lokal *site* dan PGA. Data parameter gempabumi terdiri dari magnitudo dan jarak hiposenter. Untuk data lokal *site* menggunakan data kecepatan gelombang geser (V_{s30}). Data magnitudo, jarak hiposenter dan PGA yang digunakan bersumber dari BMKG sebanyak 1828 data yang berasal dari 137 even gempa bumi dirasakan di pulau Jawa selama periode Agustus 2020 hingga Januari 2024. Sedangkan data kecepatan gelombang geser pada kedalaman 30 km atau V_{s30} bersumber dari USGS. Data tersebut disajikan pada tabel 3.1.

Tabel 3. 1 Data Magnitudo, Jarak Hiposenter, Kecepatan dan PGA

No.	Mag (SR)	Jarak Hipo (km)	Kecepatan (m/s)	PGA (gal)
1	4.6	109.1	675	1.2
2	4.6	121.6	418	1.7
3	4.6	124.7	493	0.2
4	4.6	97.3	383	3.3
...
1827	3.5	583.0	24.8	0.1
1828	3.5	378.0	32.0	0.4

3.1.3 Desain Sistem

Desain sistem menggambarkan alur proses bagaimana sistem memprediksi PGA berdasarkan metode ANN seperti yang disajikan pada gambar 3.2.



Gambar 3.2 Desain Sistem

Pada tahap input, dataset magnitudo, jarak hyposenter dan Vs30 dilakukan *preprocessing* terlebih dahulu, karena dataset masing – masing atribut memiliki satuan yang berbeda-beda dan terdapat selisih nilai yang cukup signifikan. Oleh karena itu, diperlukan proses normalisasi data untuk menyamakan skala antarvariabel sehingga seluruh data berada pada rentang nilai yang seragam dan tidak terdapat nilai yang terlalu besar maupun terlalu kecil. Dalam penelitian ini, teknik normalisasi yang diterapkan adalah *min–max scaler*, yang memetakan nilai data ke dalam interval 0 hingga 1. Adapun persamaan normalisasi tersebut ditunjukkan sebagai berikut:

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} \quad \dots\dots\dots 3.1)$$

Dimana:

- x' : data hasil normalisasi
- x : data asli
- x_{min} : nilai minimum dari data
- x_{max} : nilai maximum dari data

Data yang telah dinormalisasi kemudian masuk ke tahap proses, yaitu menggunakan model ANN dengan algoritma *Backpropagation*. Detail model ANN akan dijelaskan di bab tersendiri. Tahap akhir dari desain sistem ini adalah output, yaitu nilai prediksi PGA (*Peak Ground Acceleration*).

3.1.4 Uji Coba

Secara konseptual, metode *Artificial Neural Network (ANN)* tidak menetapkan ketentuan yang bersifat pasti dalam menentukan arsitektur jaringan yang paling optimal. Oleh karena itu, perancangan arsitektur biasanya didasarkan pada penelitian sebelumnya

serta melalui proses *trial and error* untuk memperoleh kinerja terbaik. Variasi jumlah node pada *hidden layer* dimulai dari nilai 3 karena jaringan memiliki tiga fitur input utama, yaitu Magnitudo, Jarak hiposenter dan kecepatan Vs30. Memulai jumlah node dari jumlah input merupakan pendekatan yang umum digunakan dalam perancangan ANN, karena pada tahap awal jaringan hanya memerlukan kapasitas minimal yang setara dengan jumlah fitur untuk menangkap representasi dasar dari data. Selanjutnya, jumlah node ditingkatkan secara bertahap menjadi 6, 9, 12, 15, 30, 45, 60, 75, 90, dan 105 untuk mengevaluasi peningkatan kapasitas jaringan dalam mempelajari hubungan non-linear yang lebih kompleks antara fitur input dan PGA.

Penelitian ini menggunakan dua fungsi aktivasi, yaitu *Rectified Linear Unit (ReLU)* dan *hyperbolic tangent (tanh)*. Penggunaan fungsi aktivasi *ReLU* dan *tanh* dalam penelitian ini didasarkan pada karakteristik data *Peak Ground Acceleration (PGA)* yang memiliki sifat nonlinearitas yang tinggi. Fungsi *ReLU* dipilih karena kemampuannya dalam mengatasi permasalahan *vanishing gradient* serta efisiensinya dalam mempercepat proses pelatihan ANN. Sementara itu, fungsi *tanh* dipilih karena memiliki sifat simetris terhadap nol sehingga mampu memberikan stabilitas numerik yang lebih baik dalam memodelkan hubungan kontinu pada data seismik. Pengujian kedua fungsi aktivasi ini bertujuan untuk memperoleh konfigurasi arsitektur ANN yang paling optimal dalam memprediksi PGA di wilayah Pulau Jawa. Fungsi aktivasi *ReLU* didefinisikan sebagai $f(x) = \max(0, x)$. Jika input nilainya positif, maka *ReLU* mengeluarkan input apa adanya. Tetapi jika input nilainya negatif, maka outputnya adalah nol. Sedangkan fungsi aktivasi *tanh* didefinisikan sebagai $\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$, dimana nilai e adalah konstanta logaritma natural, nilainya 2,71828. Fungsi ini menghasilkan output antara -1 dan 1.

Total data yang digunakan dalam penelitian ini adalah 1828 data, yang akan dilakukan pembagian data latih dan data uji dengan komposisi data latih dan data uji 70

% : 30 % dan 80 % : 20 %. Berdasarkan jumlah *hidden layer*, variasi jumlah node dan komposisi data latih dan uji, maka dihasilkan beberapa kombinasi yang digunakan dalam tahap pelatihan serta pengujian model dalam memprediksi nilai PGA. Nama-nama pelatihan metode ANN satu *hidden layer* selengkapnya disajikan pada tabel 3.2. Sedangkan nama-nama pelatihan metode ANN dua *hidden layer* selengkapnya disajikan pada tabel 3.3.

Tabel 3.2 Nama Pelatihan ANN Satu *Hidden Layer*

Metode ANN	Komposisi Data	Arsitektur Jaringan	Nama Pelatihan
Satu <i>Hidden Layer</i>	A 70 % : 30 %	3-3-1	HL1-A1
		3-6-1	HL1-A2
		3-9-1	HL1-A3
		3-12-1	HL1-A4
		3-15-1	HL1-A5
		3-30-1	HL1-A6
		3-45-1	HL1-A7
		3-60-1	HL1-A8
		3-75-1	HL1-A9
		3-90-1	HL1-A10
		3-105-1	HL1-A11
	B 80 % : 20 %	3-3-1	HL1-B1
		3-6-1	HL1-B2
		3-9-1	HL1-B3
		3-12-1	HL1-B4
		3-15-1	HL1-B5
		3-30-1	HL1-B6
		3-45-1	HL1-B7
		3-60-1	HL1-B8
		3-75-1	HL1-B9
		3-90-1	HL1-B10
		3-105-1	HL1-B11

Tabel 3.3 Nama Pelatihan ANN Dua *Hidden Layer*

Metode ANN	Komposisi Data	Arsitektur Jaringan	Nama Pelatihan
Dua <i>Hidden Layer</i>	A 70 % : 30 %	3-3-3-1	HL2-A1
		3-6-6-1	HL2-A2
		3-9-9-1	HL2-A3
		3-12-12-1	HL2-A4
		3-15-15-1	HL2-A5
		3-30-30-1	HL2-A6
		3-45-45-1	HL2-A7
		3-60-60-1	HL2-A8
		3-75-75-1	HL2-A9
		3-90-90-1	HL2-A10
		3-105-105-1	HL2-A11
	B 80 % : 20 %	3-3-3-1	HL2-B1
		3-6-6-1	HL2-B2
		3-9-9-1	HL2-B3
		3-12-12-1	HL2-B4
		3-15-15-1	HL2-B5
		3-30-30-1	HL2-B6
		3-45-45-1	HL2-B7
		3-60-60-1	HL2-B8
		3-75-75-1	HL2-B9
		3-90-90-1	HL2-B10
		3-105-105-1	HL2-B11

3.1.5 Evaluasi Model

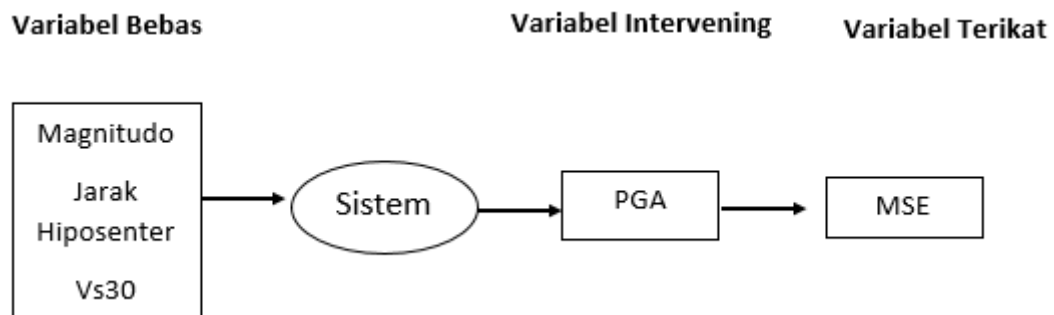
Evaluasi dilakukan dengan menguji hasil pelatihan dari setiap arsitektur model pada dua strategi pelatihan yang telah diterapkan, yaitu strategi pertama dan strategi kedua. Kinerja masing-masing model diukur menggunakan *Mean Squared Error (MSE)* sebagai metrik utama. Nilai MSE yang lebih rendah menunjukkan bahwa selisih antara nilai prediksi dan nilai aktual semakin kecil, sehingga mengindikasikan tingkat kesalahan prediksi yang lebih rendah dan performa model yang lebih baik.

$$mse = \frac{1}{n} \sum_k^n (t_k - y_k)^2 \quad \dots\dots\dots 3.2)$$

Dimana t_k merupakan nilai PGA aktual, y_k merupakan nilai PGA prediksi, dengan n adalah jumlah data.

3.2 Instrumen Penelitian

Variabel yang digunakan dalam penelitian ini yaitu, variabel bebas, variabel intervening, dan variabel terikat. Variabel bebas terdiri dari magnitudo, jarak hiposenter, dan Vs30 yang berfungsi sebagai input utama dalam sistem. Variabel intervening berupa *Peak Ground Acceleration* (PGA), yang merupakan output dari sistem dan menjadi perantara hubungan variabel bebas dengan variabel terikat. Sedangkan variabel terikat dalam penelitian ini adalah *Mean Squared Error* (MSE), yang digunakan sebagai ukuran kinerja model dengan mengukur tingkat kesalahan antara nilai PGA prediksi dan nilai PGA aktual.. Instrumen penelitian secara lengkap disajikan pada Gambar 3.3



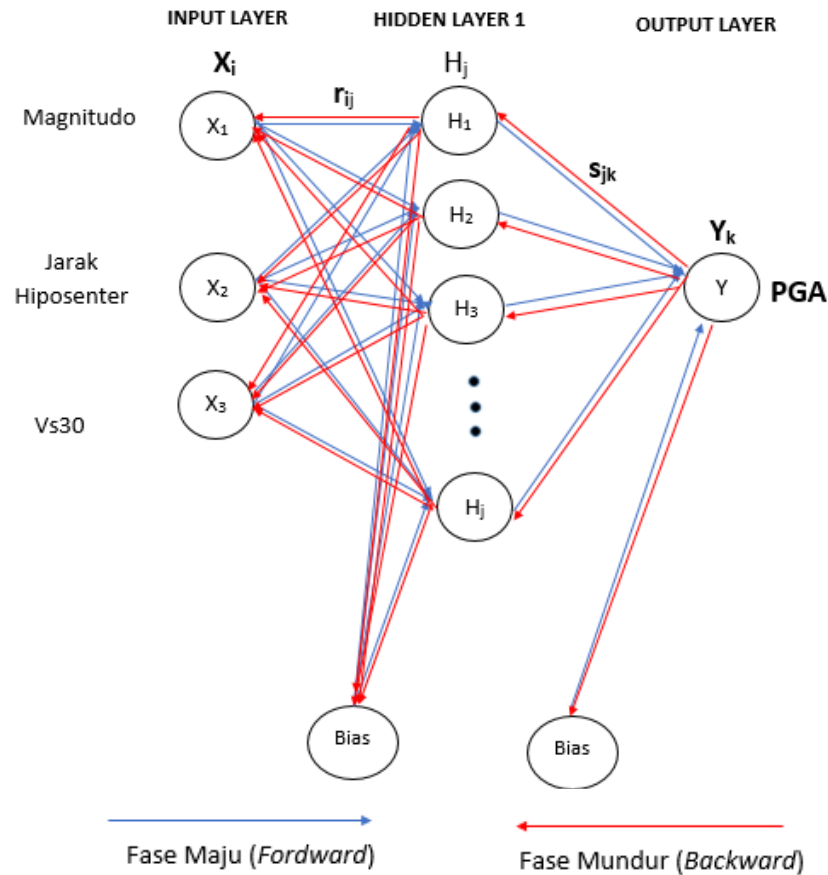
Gambar 3.3 Instrumen Penelitian

BAB 4

METODE ANN SATU HIDDEN LAYER

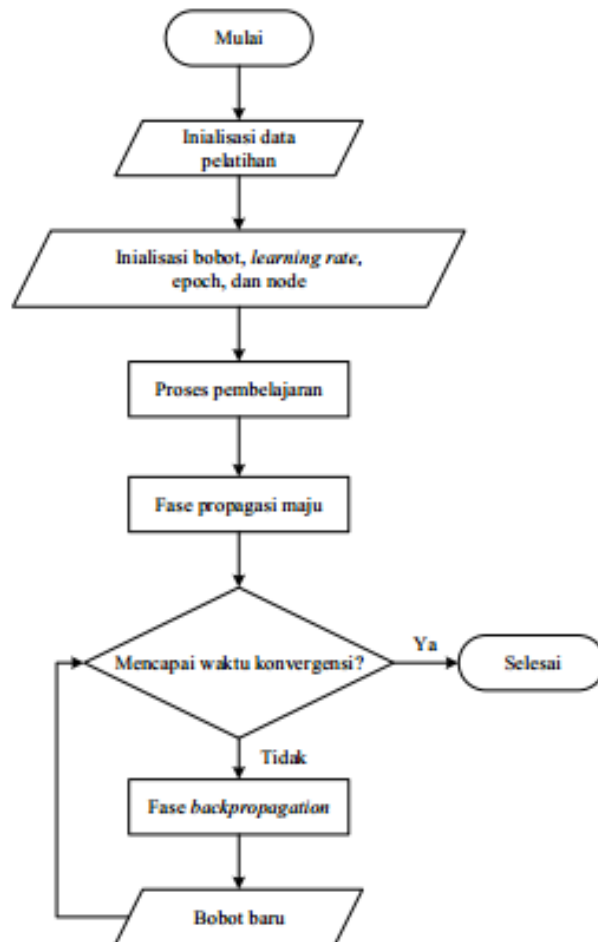
4.1 Desain Model

Penelitian ini menggunakan model Artificial Neural Network (ANN) algoritma Backpropagation dengan satu *hidden layer* untuk memprediksi *Peak Ground Acceleration* (PGA) di Pulau Jawa. Pendekatan ANN dengan satu *hidden layer* juga telah diterapkan oleh beberapa penelitian sebelumnya terkait prediksi PGA, seperti Hanping (2012), Dhanya (2017), dan Khosravikia (2018). Penggunaan satu *hidden layer* memungkinkan model mengekstraksi fitur awal, menangkap pola dasar, serta memodelkan hubungan non-linear tingkat awal dari data input. Dengan struktur ini, jaringan dapat mempelajari keterkaitan sederhana antar variabel dan memperoleh pemahaman dasar terhadap karakteristik data sebelum melanjutkan ke proses pembelajaran yang lebih mendalam. Arsitektur jaringan ANN dengan satu *hidden layer* disajikan pada Gambar 4.1.



Gambar 4.1 Desain ANN Satu *Hidden Layer*

Desain ANN metode pertama ini terdiri dari tiga *layer*, yaitu satu *layer input*, satu *hidden layer* dan satu *layer output*. Data masukan atau input (X_i) berupa data magnitudo, jarak hiposenter dan kecepatan Vs30. Bobot koneksi yang menghubungkan lapisan input dengan *hidden layer* dinyatakan sebagai (r_{ij}), nilai pembobotan antara *hidden layer* ke output (s_{jk}), sedangkan keluaran dari *hidden layer* dengan jumlah neuron yang telah ditetapkan dilambangkan sebagai (H_j), dan nilai keluaran akhir dari jaringan merupakan nilai *Peak Ground Acceleration (PGA)* yang direpresentasikan sebagai (Y_k). Proses pelatihan model menggunakan *Mean Square Error (MSE)* sebagai kriteria penghentian sekaligus dasar penentuan bobot akhir yang optimal. Dengan demikian, pelatihan akan dihentikan ketika kondisi konvergensi telah tercapai. Alur atau diagram tahapan pelatihan pada strategi pertama untuk prediksi PGA disajikan pada Gambar 4.2.



Gambar 4.2 Flowchart Proses Pelatihan

Secara rinci alur proses pelatihan strategi pertama dapat dijelaskan sebagai berikut :

1. Proses dimulai
2. Data latih yang telah melalui tahap normalisasi disiapkan sebagai input.
3. Bobot awal ditentukan dengan memberikan nilai acak pada setiap koneksi dalam rentang 0 hingga 1, selanjutnya ditetapkan nilai α (learning rate) serta jumlah neuron yang digunakan.
4. Proses pembelajaran, terdapat dua fase, yaitu fase propagasi maju dan propagasi mundur.
5. Tahap Propagasi Maju (*Forward Propagation*)

- a. Seluruh sinyal yang menuju ke hidden layer (H_j) dihitung. Pada tahap ini, informasi awal diberikan pada variabel input (X_i) yang selanjutnya meneruskan sinyal tersebut ke seluruh neuron pada *hidden layer*.

$$H_{net\ j} = r_{0j} + \sum_{i=1}^3 X_i r_{ij} \quad (4.1)$$

$$H_{net\ j} = r_{0j} + (M \cdot r_{1j} + R \cdot r_{2j} + V \cdot r_{3j}) \quad (4.2)$$

Dengan i merupakan indeks input ke- i ($i = 1, 2$ dan 3) pada lapisan input dan j adalah node ke- j ($j = 1, 2, \dots, p$) pada *hidden layer*. X_i merupakan nilai input di dalam node masukan ke- i dan r_{ij} adalah bobot pada node input ke- i yang menuju ke node *hidden* ke- j . Sedangkan r_{0j} adalah bobot pada node bias yang menuju ke node *hidden layer*. M adalah nilai magnitudo, R adalah jarak hiposenter, dan V adalah kecepatan V_s 30.

- b. Seluruh nilai keluaran pada *hidden layer* dihitung dengan menerapkan fungsi aktivasi *Rectified Linear Unit (ReLU)*. Nilai hasil aktivasi tersebut kemudian dimanfaatkan oleh setiap neuron pada *hidden layer* sebagai masukan pada tahap pemrosesan berikutnya.

$$H_j = f(H_{net\ j}) = \max(0, H_{net\ j}) \quad (4.3)$$

Nilai max menunjukkan nilai terbesar antara 0 dan $H_{net\ j}$. Hasil dari fungsi $f(H_{net\ j})$ merepresentasikan nilai aktivasi pada neuron tersembunyi ke- j , yang selanjutnya diteruskan sebagai masukan ke seluruh neuron pada lapisan output.

- c. Setelah didapatkan nilai tiap node pada *hidden layer* selanjutnya dijumlahkan semua sinyal yang masuk ke output ($H_j \dots H_n$) dengan bobotnya.

$$y_{net\ k} = s_{0k} + \sum_{j=1}^k H_j s_{jk} \quad (4.4)$$

s_{0k} adalah nilai bias pada *hidden layer* dan H_j adalah hasil fungsi aktivasi yang keluar dari *hidden layer* dan s_{jk} adalah bobot pada node hidden ke- j menuju ke node output k ($k = 1, 2, \dots, m$).

- d. Dengan menggunakan fungsi aktivasi *Rectified Linear Units* (ReLU) untuk menghitung sinyal output, maka akan didapatkan nilai keluarannya yaitu:

$$y_k = f(y_{\text{net } k}) = \max(0, y_{\text{net } k}) \quad (4.5)$$

- e. Setelah diperoleh nilai output (y_k) kemudian dihitung nilai *loss function* atau selisih antara nilai *PGA* aktual (*ground truth*) dengan nilai *PGA* prediksi menggunakan rumus :

$$Loss(L) = (t_k - y_k)^2 \quad (4.6)$$

$$\begin{aligned} mse &= \frac{1}{n} \sum_k^n L \\ &= \frac{1}{n} \sum_k^n (t_k - y_k)^2 \end{aligned} \quad (4.7)$$

L adalah fungsi *loss function*, t_k adalah nilai *PGA* aktual (*ground truth*) dan (y_k) adalah nilai *PGA* output. Sedangkan k adalah indeks data ke- k (1,2,3....n). Proses pelatihan akan berhenti jika waktu konvergensi telah tercapai. Akan tetapi jika waktu konvergensi belum tercapai, maka dilakukan proses *backpropagation* dengan memperbarui bobot.

6. Tahap Propagasi Mundur (*Backward Propagation*)

- a. Hitung nilai kesalahan (δ_k) pada lapisan output (y_k) dengan membandingkan output terhadap pola target (t_k) Nilai kesalahan ini kemudian diteruskan ke lapisan sebelumnya untuk digunakan dalam perhitungan koreksi bobot dan bias antara lapisan input dan *hidden layer*.

$$\delta_k = (t_k - y_k) f'(y_{net_k}) \quad (4.8)$$

Dimana:

$$f'(y_{net_k}) = (1 - f(y_{net_k})) f(y_{net_k}) \quad (4.9)$$

- b. Setelah nilai kesalahan pada *hidden layer* dan *output layer* diperoleh, tahap berikutnya adalah menghitung besarnya kesalahan yang terjadi pada lapisan input ke *hidden layer*.

$$\delta_j = \delta_{net_j} f'(H_{net_j}) \quad (4.10)$$

Dimana:

$$\delta_{net_j} = \sum_{k=1}^m \delta_k s_{jk} \quad (4.11)$$

$$f'(H_{net_j}) = (1 - f(H_{net_j})) f(H_{net_j}) \quad (4.12)$$

- c. Setelah nilai kesalahan diperoleh, besarnya penyesuaian atau koreksi yang akan diterapkan pada bobot lama dapat dihitung.

$$\Delta r_{ij} = \alpha \delta_j X_i \quad (4.13)$$

$$\Delta s_{jk} = \alpha \delta_k H_j \quad (4.14)$$

Δr_{ij} dan Δs_{jk} menyatakan besarnya penyesuaian yang akan ditambahkan pada bobot sebelumnya dalam proses pembaruan bobot. Dengan α laju pembelajaran (*learning rate*), δ_j nilai error yang dibawa antara *input layer* dan *hidden layer* dan δ_k nilai error antara *hidden layer* dan *output layer*. X_i merupakan nilai input yang berasal dari node i ke node j sedangkan H_j adalah keluaran hasil fungsi aktivasi pada *hidden layer*.

7. Tahap Perubahan Bobot

Setelah Δr_{ij} dan Δs_{jk} diperoleh selanjutnya adalah memperbaharui bobot lama menjadi bobot baru.

$$r_{ij}(\text{baru}) = r_{ij}(\text{lama}) + \Delta r_{ij} \quad (4.15)$$

$$s_{jk}(\text{baru}) = s_{jk}(\text{lama}) + \Delta s_{jk} \quad (4.16)$$

$r_{ij}(\text{baru})$ merupakan bobot baru antara input layer dan *hidden layer* yang akan dicari, $r_{ij}(\text{lama})$ merupakan bobot lama yang akan diperbaharui. $s_{jk}(\text{baru})$ adalah bobot baru antara *hidden layer* dan output layer yang akan dicari sedangkan $s_{jk}(\text{lama})$ adalah bobot lama yang akan diperbaharui.

4.2. Implementasi

Proses implementasi model ANN metode pertama dilakukan menggunakan *software* python pada *Google Colab*. Parameter pelatihan yang digunakan dalam penelitian ini selengkapnya disajikan pada Tabel 4.1.

Tabel 4.1 Parameter Model ANN Satu *Hidden Layer*

Tipe Jaringan	BP-ANN
Fungsi Aktivasi	<i>Rectified Linear Units</i> (ReLU) <i>hyperbolic tangent</i> (tanh)
Fungsi Optimasi	<i>Adaptive Moment Estimation</i> (Adam)
Learning Rate	0.001
Jumlah Node input layer	3
Jumlah Node Hidden Layer 1	3, 6, 9, 12, 15, 30, 45, 60, 75, 90, 105
Jumlah Node Output Layer	1
Komposisi Data	70:30 dan 80:20
Kinerja	MSE (<i>Mean Square Error</i>)

4.3 Uji Coba

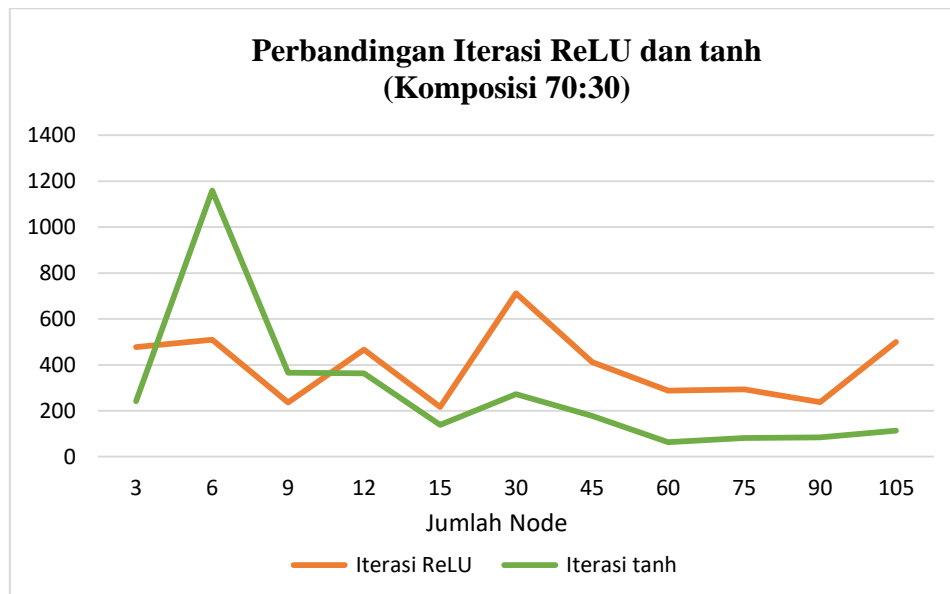
Pada penelitian ini dilakukan uji coba model ANN satu *hidden layer* dengan dua komposisi data *training* dan data *testing*, yaitu Komposisi data 70 : 30 dan 80 : 20. Masing-masing komposisi data tersebut juga dilakukan uji coba performa dari dua fungsi aktivasi, yaitu *Rectified Linear Units* (ReLU) dan *hyperbolic tangent* (*tanh*).

4.3.1 Komposisi Data 70 : 30

Komposisi data 70 : 30 artinya data yang digunakan untuk pelatihan adalah 70 % dari 1828 total data , yaitu 1280 data. Sedangkan data yang digunakan untuk pengujian adalah 30 % dari 1828 total data, yaitu 548 data. Hasil pelatihan dan pengujian selengkapnya disajikan pada tabel 4.2.

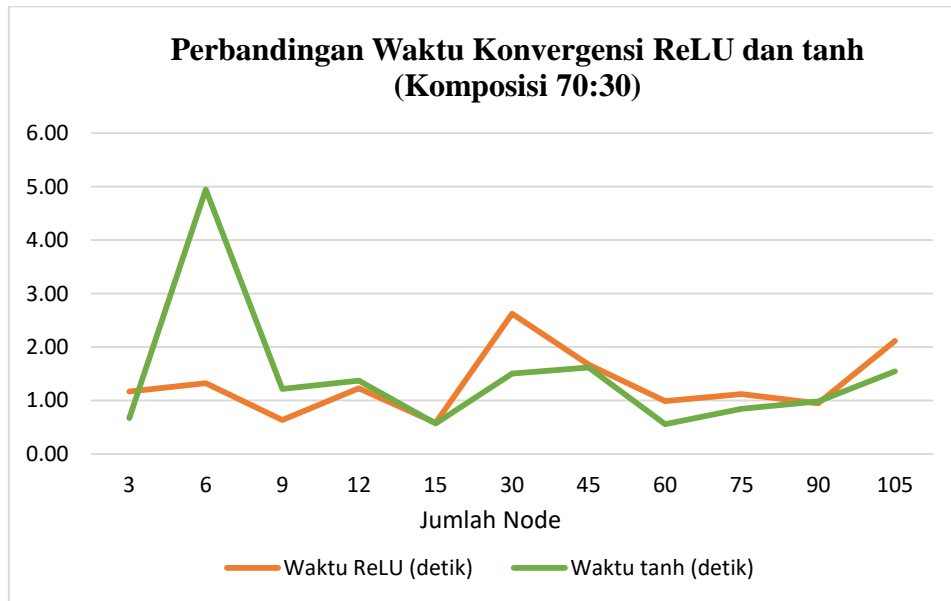
Tabel 4.2 Hasil pelatihan dan pengujian komposisi data 70:30

Nama Arsitektur	Node	Iterasi ReLU	Iterasi tanh	Waktu ReLU (detik)	Waktu tanh (detik)	MSE Training ReLU	MSE Training tanh	MSE Testing ReLU	MSE Testing tanh
HL1-A1	3	477	242	1.17	0.67	0.2724	0.3685	0.2676	0.3741
HL1-A2	6	510	1159	1.33	4.95	0.3700	0.2729	0.3805	0.2617
HL1-A3	9	236	366	0.63	1.22	0.3684	0.3667	0.3798	0.3790
HL1-A4	12	467	363	1.23	1.37	0.3682	0.3676	0.3765	0.3769
HL1-A5	15	217	139	0.58	0.57	0.3842	0.3762	0.3928	0.3838
HL1-A6	30	712	272	2.62	1.51	0.2691	0.3680	0.2539	0.3784
HL1-A7	45	412	178	1.67	1.62	0.2576	0.3675	0.2470	0.3789
HL1-A8	60	288	63	0.99	0.56	0.2459	0.3684	0.2326	0.3788
HL1-A9	75	294	81	1.12	0.84	0.2482	0.3684	0.2350	0.3797
HL1-A10	90	238	84	0.95	0.99	0.2597	0.3685	0.2505	0.3796
HL1-A11	105	500	114	2.11	1.55	0.2396	0.3681	0.2282	0.3799



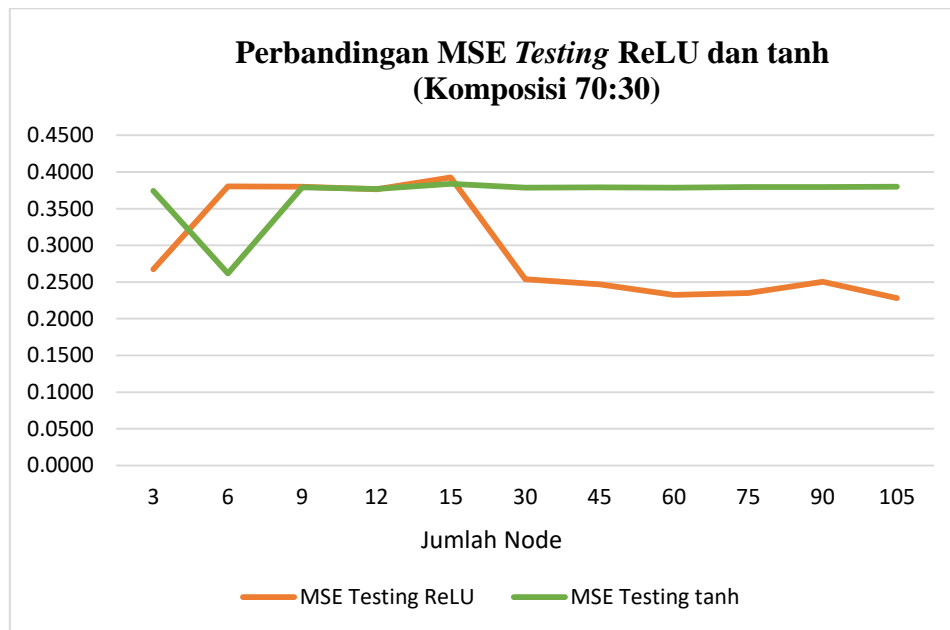
Gambar 4.3 Perbandingan Iterasi ReLU dan tanh (Komposisi 70:30)

Gambar 4.3 merupakan perbandingan jumlah iterasi pelatihan antara fungsi aktivasi ReLU dan tanh pada arsitektur ANN satu *hidden layer* dengan Komposisi data 70:30. Pola konvergensi antara fungsi aktivasi ReLU dan tanh cenderung fluktuatif, namun menunjukkan kecenderungan tertentu pada jumlah node yang lebih besar. Pada jumlah node kecil (3 -15), fungsi aktivasi tanh cenderung membutuhkan iterasi yang lebih banyak dibandingkan ReLU. Namun, ketika jumlah node mulai ditingkatkan (30–105), tanh cenderung mencapai konvergensi dengan iterasi lebih sedikit dari pada ReLU.



Gambar 4.4 Perbandingan Waktu Konvergensi ReLU dan tanh (Komposisi 70:30)

Gambar 4.4 merupakan perbandingan waktu konvergensi pelatihan antara fungsi aktivasi ReLU dan tanh pada arsitektur ANN satu *hidden layer* dengan Komposisi data 70:30. Waktu konvergensi menunjukkan pola yang relatif konsisten dengan grafik jumlah iterasi. Pada node kecil (3-15) waktu konvergensi tanh cenderung lebih lama dari pada ReLU. Namun, ketika jumlah node semakin besar waktu konvergensi tanh cenderung lebih cepat dari pada ReLU.

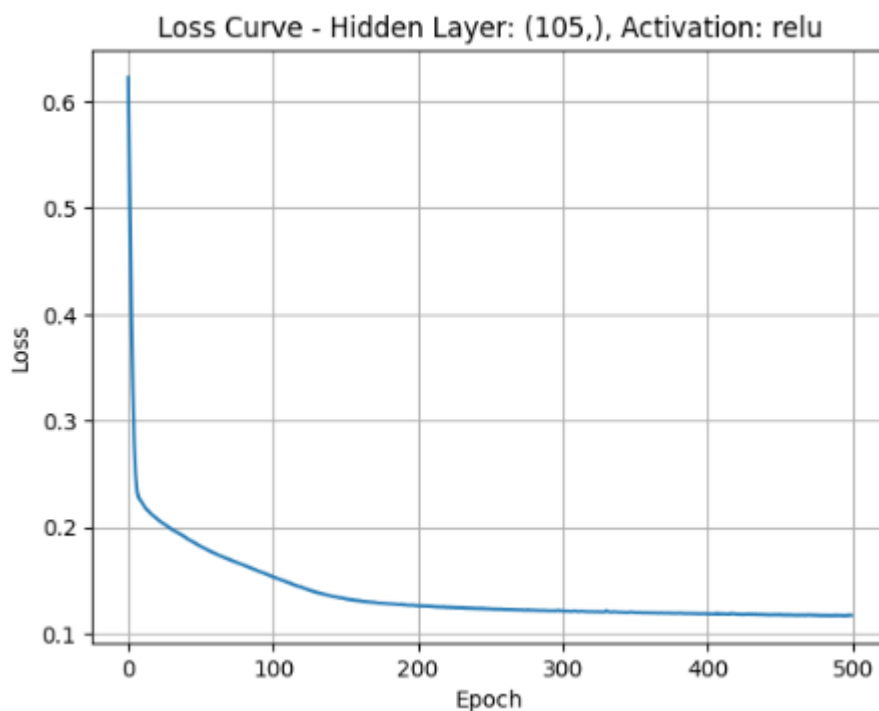


Gambar 4.5 Perbandingan MSE Testing ReLU dan tanh (Komposisi 70:30)

Gambar 4.5 merupakan perbandingan MSE *testing* antara fungsi aktivasi ReLU dan tanh pada arsitektur ANN satu *hidden layer* dengan Komposisi data 70:30. Nilai MSE *testing* menunjukkan perbedaan performa yang paling signifikan antara ReLU dan tanh. Pada jumlah node kecil (3–15), performa kedua fungsi aktivasi masih relatif sebanding. Namun, ketika jumlah node ditingkatkan, tren yang sangat jelas mulai terlihat, di mana fungsi aktivasi ReLU secara konsisten menghasilkan nilai MSE testing yang lebih rendah dibandingkan tanh.

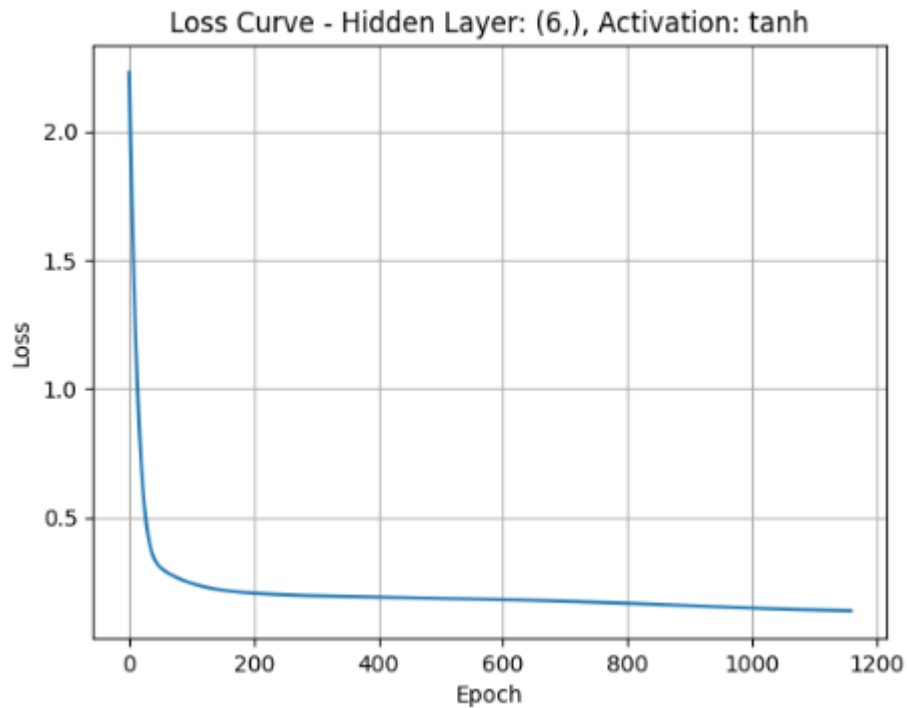
Berdasarkan tabel 4.2 hasil pengujian arsitektur ANN satu *hidden layer* dengan Komposisi data 70:30 menggunakan fungsi aktivasi ReLU, diperoleh performa terbaik pada arsitektur HL1-A11 dengan jumlah node 105, nilai MSE *testing* sebesar 0.2282, jumlah iterasi 500 dan waktu konvergensi 2.11 detik. Sedangkan pada penggunaan fungsi aktivasi tanh, performa terbaik diperoleh pada arsitektur HL1-A2 dengan jumlah node 6, nilai MSE testing sebesar 0.2617, jumlah iterasi 1159 dan waktu konvergensi 4.95 detik. Untuk melihat pola konvergensi selama proses pelatihan dari arsitektur HL1-A11 dan

HL1-A2 ditampilkan grafik *loss function* sebagaimana ditunjukkan pada Gambar 4.6 dan 4.7.



Gambar 4.6 Grafik *loss function* arsitektur HL1-A11

Gambar 4.6 merupakan grafik *loss function* selama proses pelatihan model ANN dengan arsitektur HL1-A11 (satu *hidden layer* dengan 105 *node*) menggunakan fungsi aktivasi *ReLU* dan komposisi data 70:30. Berdasarkan grafik tersebut terlihat bahwa penurunan nilai *loss* yang sangat tajam pada awal iterasi (*epoch* 1 – 20), dari kisaran sekitar 0,62 menjadi sekitar 0,23. Hal ini mengindikasikan bahwa model mampu dengan cepat mempelajari pola dasar hubungan antara parameter input (magnitudo, jarak, dan Vs30) terhadap nilai output *Peak Ground Acceleration* (PGA). Pada fase selanjutnya (sekitar *epoch* 20 – 200), penurunan *loss* berlangsung lebih gradual hingga mencapai nilai sekitar 0,13. Setelah melewati *epoch* 200 hingga akhir pelatihan (*epoch* 500), kurva *loss* tampak relatif stabil dan mendatar di kisaran 0,11–0,12 yang menandakan bahwa model telah mencapai kondisi konvergen.



Gambar 4.7 Grafik *loss function* arsitektur HL1-A2

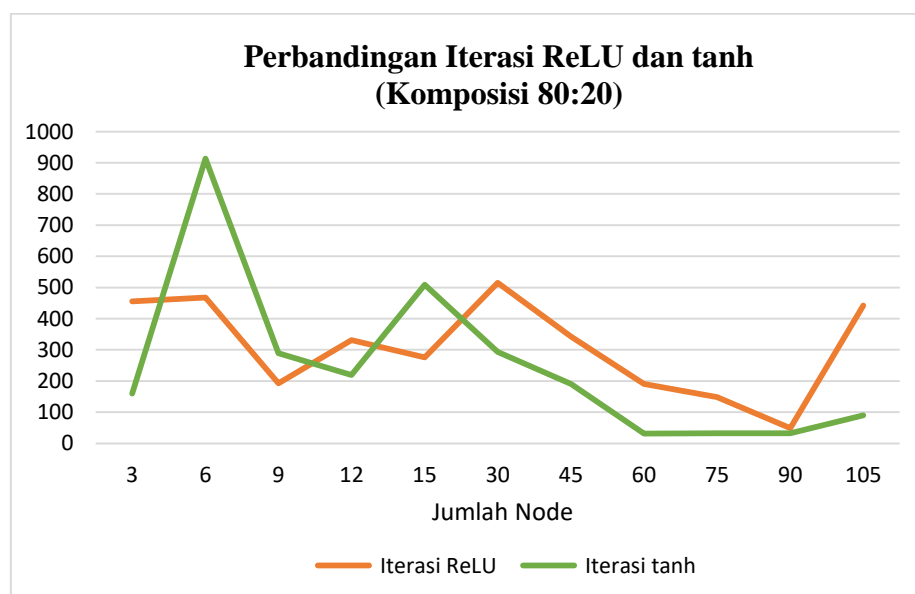
Gambar 4.7 merupakan grafik *loss function* selama proses pelatihan model ANN dengan arsitektur HL1-A2 (satu *hidden layer* dengan 6 *node*) menggunakan fungsi aktivasi tanh dan komposisi data 70:30. Berdasarkan grafik tersebut terlihat bahwa penurunan nilai *loss* yang sangat signifikan pada fase awal pelatihan (epoch 1 – 50), dari kisaran sekitar 2,2 menjadi sekitar 0,35. Memasuki fase menengah pelatihan (sekitar epoch 50 – 300), kurva *loss* terus menurun secara perlahan hingga mencapai kisaran 0,20 yang menandakan proses pembelajaran mulai melambat. Pada tahap akhir pelatihan hingga epoch 1159, kurva *loss* terlihat semakin landai di kisaran 0,15 – 0,12 yang menandakan kondisi konvergensi tercapai.

4.3.2 Komposisi Data 80 : 20

Komposisi data 80 : 20 artinya data yang digunakan untuk pelatihan adalah 80 % dari 1828 total data , yaitu 1462 data. Sedangkan data yang digunakan untuk pengujian adalah 20 % dari 1828 total data, yaitu 366 data. Hasil pelatihan dan pengujian selengkapnya disajikan pada tabel 4.3.

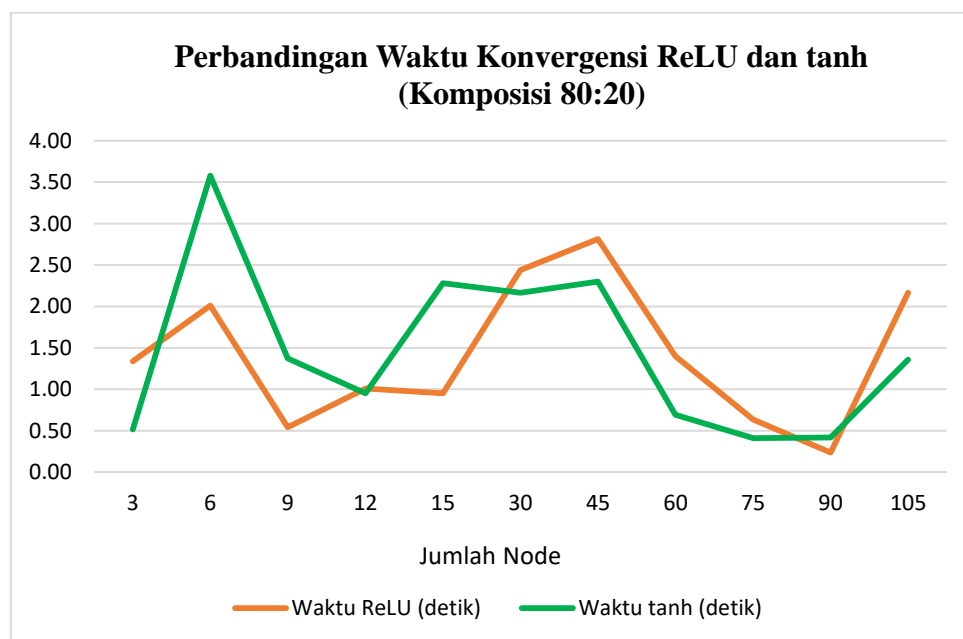
Tabel 4.3 Hasil pelatihan dan pengujian komposisi data 80:20

Nama Arsitektur	Node	Iterasi ReLU	Iterasi tanh	Waktu ReLU (detik)	Waktu tanh (detik)	MSE Training ReLU	MSE Training tanh	MSE Testing ReLU	MSE Testing tanh
HL1-B1	3	455	160	1.34	0.51	0.2647	0.3738	0.2620	0.3595
HL1-B2	6	468	913	2.01	3.58	0.3738	0.2805	0.3646	0.2650
HL1-B3	9	193	289	0.54	1.37	0.3739	0.3729	0.3649	0.3653
HL1-B4	12	331	219	1.01	0.95	0.3736	0.3747	0.3617	0.3654
HL1-B5	15	276	509	0.95	2.28	0.3741	0.2762	0.3650	0.2570
HL1-B6	30	515	293	2.44	2.17	0.2657	0.3710	0.2509	0.3623
HL1-B7	45	342	190	2.82	2.30	0.2532	0.3712	0.2458	0.3635
HL1-B8	60	190	31	1.40	0.69	0.2473	0.3777	0.2389	0.3651
HL1-B9	75	148	32	0.63	0.41	0.2578	0.3789	0.2467	0.3657
HL1-B10	90	49	32	0.24	0.42	0.3789	0.3954	0.3633	0.3760
HL1-B11	105	442	90	2.16	1.36	0.2363	0.3735	0.2300	0.3654



Gambar 4.8 Perbandingan Iterasi ReLU dan tanh (Komposisi 80:20)

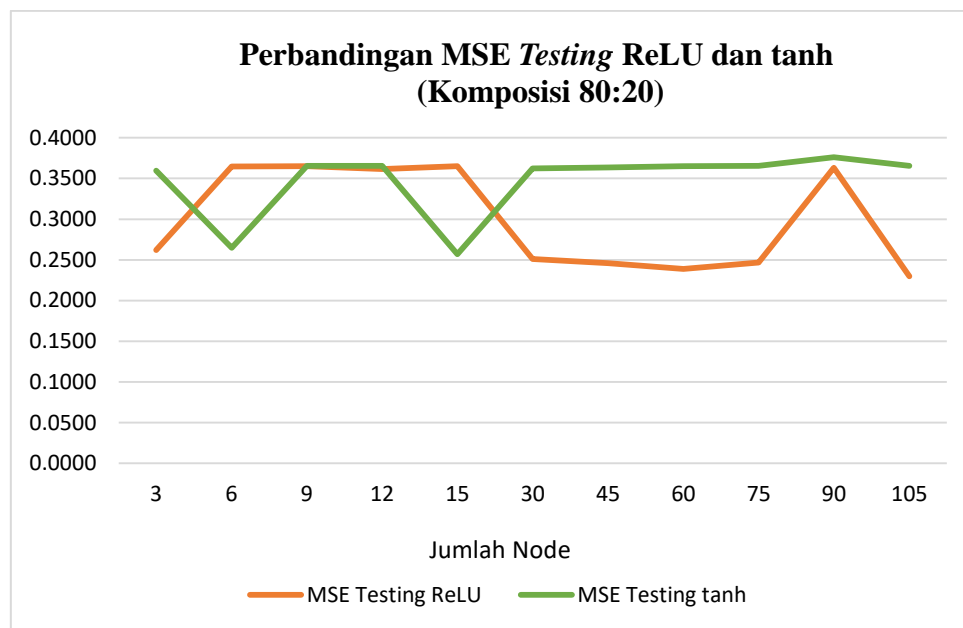
Gambar 4.8 merupakan perbandingan jumlah iterasi pelatihan antara fungsi aktivasi ReLU dan tanh pada arsitektur ANN satu *hidden layer* dengan Komposisi data 80:20. Pola konvergensi antara fungsi aktivasi ReLU dan tanh pada komposisi data 80 : 20 cenderung ada kesamaan dengan komposisi data 70 : 30, yaitu pada konfigurasi jumlah node kecil fungsi aktivasi tanh cenderung membutuhkan jumlah iterasi yang lebih banyak dibandingkan dengan ReLU. Namun dengan penambahan jumlah node fungsi aktivasi tanh cenderung membutuhkan jumlah iterasi yang lebih sedikit dari pada ReLU. Hal ini menunjukkan pola kecenderungan jumlah iterasi lebih dipengaruhi oleh fungsi aktivasi dan jumlah node dari pada komposisi data.



Gambar 4.9 Perbandingan Waktu Konvergensi ReLU dan tanh (Komposisi 80:20)

Gambar 4.9 merupakan perbandingan waktu konvergensi pelatihan antara fungsi aktivasi ReLU dan tanh pada arsitektur ANN satu *hidden layer* dengan Komposisi data 80:20. Waktu konvergensi juga menunjukkan pola yang relatif sama dengan komposisi data 70:30, yaitu pada node kecil (3-15) waktu konvergensi tanh cenderung lebih lama

dari pada ReLU. Namun, ketika jumlah node semakin besar waktu komvergensi tanh cenderung lebih cepat dari pada ReLU. Hal ini juga menunjukkan pola waktu konvergensi lebih dipengaruhi oleh fungsi aktivasi dan jumlah node dari pada komposisi data.

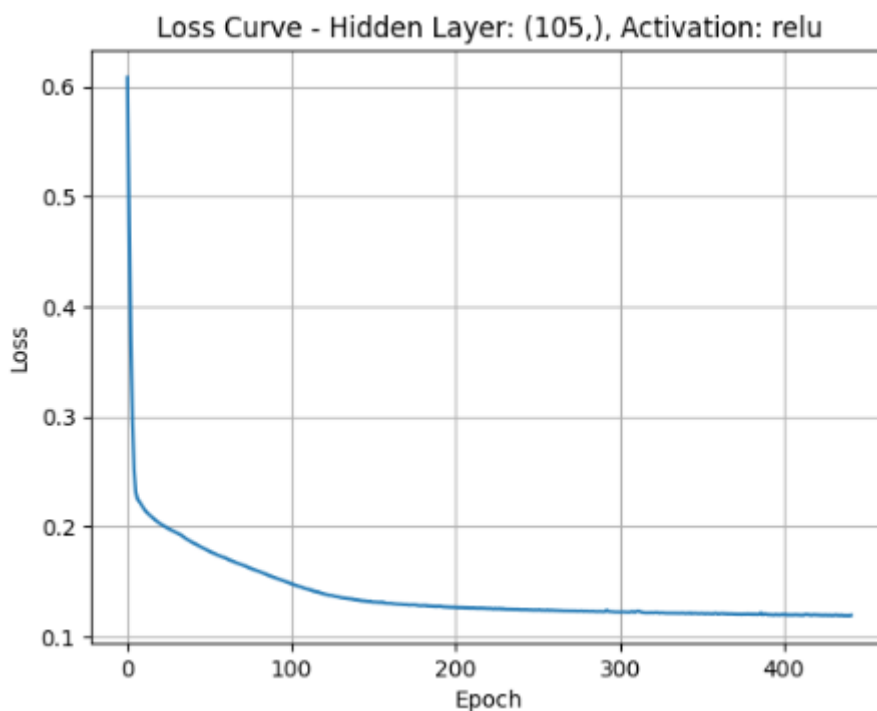


Gambar 4.10 Perbandingan MSE *Testing* ReLU dan tanh (Komposisi 80:20)

Gambar 4.10 merupakan perbandingan MSE *testing* antara fungsi aktivasi ReLU dan tanh pada arsitektur ANN satu *hidden layer* dengan Komposisi data 80:20. Pada jumlah node kecil (3–15), performa kedua fungsi aktivasi masih relatif sebanding. Namun, ketika jumlah node ditingkatkan, mulai terlihat tren yang semakin jelas di mana fungsi aktivasi ReLU secara umum menghasilkan nilai MSE *testing* yang lebih rendah dibandingkan tanh, meskipun ReLU masih ada fluktuatif di node yang lebih besar sementara tanh cenderung stabil. Hal ini dikarenakan ReLU lebih sensitif terhadap perubahan struktur jaringan.

Berdasarkan tabel 4.3 hasil pengujian arsitektur ANN satu *hidden layer* dengan Komposisi data 80:20 menggunakan fungsi aktivasi ReLU, diperoleh performa terbaik

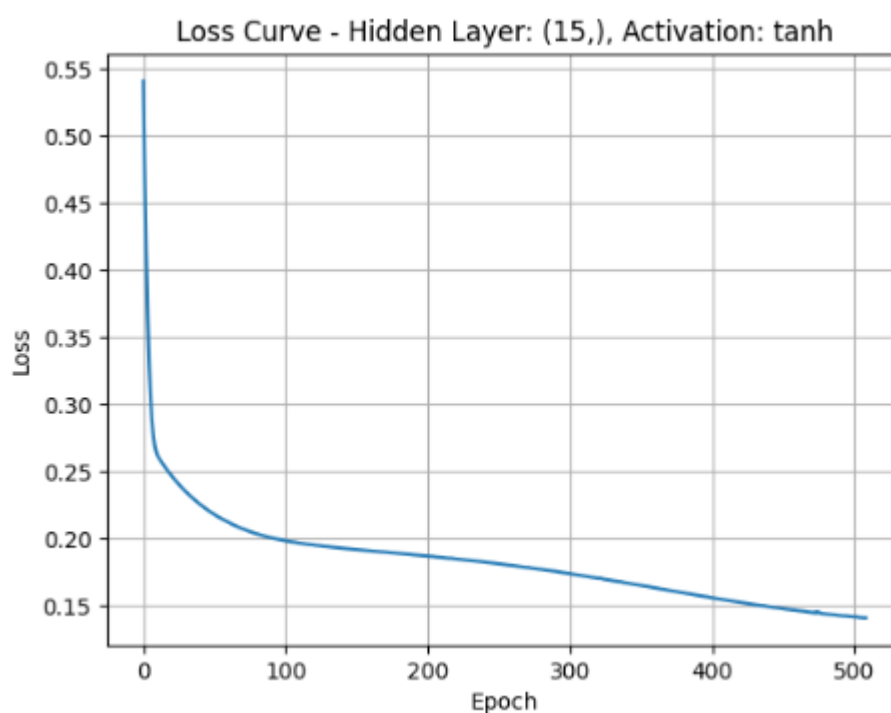
pada arsitektur HL1-B11 dengan jumlah node 105, nilai MSE *testing* sebesar 0.2300, jumlah iterasi 442 dan waktu konvergensi 2.16 detik. Sedangkan pada penggunaan fungsi aktivasi tanh, performa terbaik diperoleh pada arsitektur HL1-B5 dengan jumlah node 15, nilai MSE *testing* sebesar 0.2570, jumlah iterasi 509 dan waktu konvergensi 2.28 detik. Untuk melihat pola konvergensi selama proses pelatihan dari arsitektur HL1-B11 dan HL1-B5 ditampilkan grafik *loss function* sebagaimana ditunjukkan pada Gambar 4.11 dan 4.12.



Gambar 4.11 Grafik *loss function* arsitektur HL1-B11

Gambar 4.11 merupakan grafik *loss function* selama proses pelatihan model ANN dengan arsitektur HL1-B11 (satu *hidden layer* dengan 105 *node*) menggunakan fungsi aktivasi *ReLU* dan komposisi data 80:20. Berdasarkan grafik tersebut terlihat bahwa penurunan nilai *loss* yang sangat tajam pada awal iterasi (*epoch* 1 – 10), dari kisaran sekitar 0,62 menjadi sekitar 0,23. Hal ini mengindikasikan bahwa model mampu dengan cepat mempelajari pola dasar hubungan antara parameter input (magnitudo, jarak,

dan Vs30) terhadap nilai output *Peak Ground Acceleration* (PGA). Pada fase selanjutnya (sekitar *epoch* 10 – 150), penurunan *loss* berlangsung lebih gradual hingga mencapai nilai sekitar 0,13 yang menunjukkan proses pembelajaran mulai memasuki fase penyempurnaan. Setelah melewati *epoch* 150 hingga akhir pelatihan (*epoch* 442), kurva *loss* tampak relatif stabil dan mendatar di kisaran 0,12 – 0,11 yang menandakan bahwa model telah mencapai kondisi konvergen.



Gambar 4.12 Grafik *loss function* arsitektur HL1-B5

Gambar 4.12 merupakan grafik *loss function* selama proses pelatihan model ANN dengan arsitektur HL1-B5 (satu *hidden layer* dengan 15 *node*) menggunakan fungsi aktivasi tanh dan komposisi data 80:20. Berdasarkan grafik tersebut terlihat bahwa penurunan nilai *loss* yang sangat signifikan pada fase awal pelatihan (*epoch* 1 – 30), dari kisaran sekitar 0,54 menjadi sekitar 0,27. Hal ini mengindikasikan bahwa model mampu dengan cepat menyesuaikan bobot awal dan mulai mempelajari pola dasar hubungan

antara parameter input dan output. Memasuki fase menengah pelatihan (sekitar epoch 30 – 200), kurva loss terus menurun secara bertahap hingga mencapai kisaran 0,20. Pada tahap akhir pelatihan hingga epoch 509, kurva loss terlihat semakin landai dan stabil di kisaran 0,15 – 0,14 yang menandakan konvergensi tercapai.

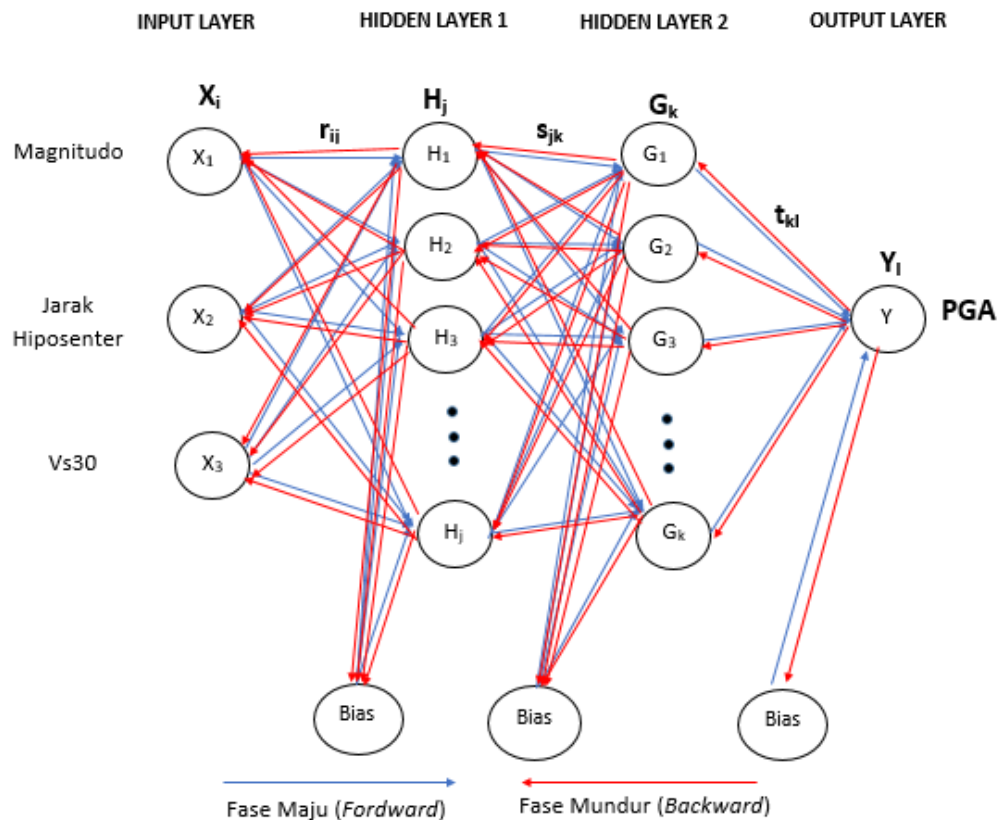
BAB 5

METODE ANN DUA HIDDEN LAYER

5.1 Desain Model

Metode kedua yang digunakan dalam penelitian ini adalah ANN dengan dua *hidden layer*, mengikuti pendekatan yang juga digunakan oleh Preziaso (2022) dalam prediksi PGA. Penambahan *hidden layer* kedua bertujuan untuk mempelajari pola yang lebih kompleks berdasarkan representasi fitur yang telah diekstraksi oleh *hidden layer* pertama. Dengan struktur ini, jaringan menggabungkan dan mengolah fitur tingkat rendah menjadi representasi fitur tingkat tinggi, sehingga model dapat menangkap interaksi antar variabel secara lebih menyeluruh dan memetakan hubungan non-linear yang kompleks pada data PGA. Pendekatan ini diharapkan meningkatkan akurasi prediksi dengan tetap mempertahankan kemampuan generalisasi model terhadap data baru.

Desain model ANN metode kedua terdiri dari empat *layer*, yaitu satu *layer input*, dua *hidden layer* dan satu *layer output*. Data masukan atau input (X_i) berupa data magnitudo, jarak hiposenter dan kecepatan Vs30. (r_{ij}) merupakan nilai pembobotan antara lapisan *input* ke *hidden layer* pertama, nilai keluaran pada *hidden layer* pertama yang jumlah nodenya sudah ditentukan (H_j), nilai pembobotan antara *hidden layer* pertama ke *hidden layer* kedua (s_{jk}), nilai keluaran pada *hidden layer* kedua yang jumlah nodenya sudah ditentukan (G_k), nilai pembobotan antara *hidden layer* kedua ke output (t_{kl}) dan nilai outputnya berupa nilai PGA (Y_l). Desain jaringan ANN dengan dua *hidden layer* dapat dilihat pada Gambar 5.1.



Gambar 5.1 Desain ANN Dua *Hidden Layer*

Proses perhitungan model ANN dua *hidden layer* adalah sebagai berikut :

1. Proses dimulai
2. Data latih yang telah melalui tahap normalisasi disiapkan sebagai input.
3. Bobot awal ditentukan dengan memberikan nilai acak pada setiap koneksi dalam rentang 0 hingga 1, selanjutnya ditetapkan nilai α (learning rate) serta jumlah neuron yang digunakan.
4. Proses pembelajaran, terdapat dua fase, yaitu fase propagasi maju dan propagasi mundur.
5. Tahap Propagasi Maju (Forward Propagation)

- a. Seluruh sinyal yang menuju ke hidden layer pertama (H_j) dihitung. Pada tahap ini, informasi awal diberikan pada variabel input (X_i) yang selanjutnya meneruskan sinyal tersebut ke seluruh neuron pada hidden layer.

$$H_{net\ j} = r_{0j} + \sum_{i=1}^3 X_i r_{ij} \quad (5.1)$$

$$H_{net\ j} = r_{0j} + (M \cdot r_{1j} + R \cdot r_{2j} + V \cdot r_{3j}) \quad (5.2)$$

Dengan i adalah indeks input ke- i ($i = 1, 2$ dan 3) pada lapisan input dan j adalah node ke- j ($j = 1, 2, \dots, p$) pada *hidden layer*. X_i adalah nilai input di dalam node masukan ke- i dan r_{ij} adalah bobot pada node input ke- i yang menuju ke node *hidden* ke- j . Sedangkan r_{0j} adalah bobot pada node bias yang menuju ke node *hidden layer*. M adalah nilai magnitudo, R adalah jarak hiposenter, dan V adalah kecepatan Vs30.

- b. Seluruh nilai keluaran pada *hidden layer* dihitung dengan menerapkan fungsi aktivasi *Rectified Linear Unit (ReLU)*. Nilai hasil aktivasi tersebut kemudian dimanfaatkan oleh setiap neuron pada hidden layer pertama sebagai masukan pada tahap pemrosesan berikutnya (G_k)

$$H_j = f(H_{net\ j}) = \max(0, H_{net\ j}) \quad (5.3)$$

Nilai \max menunjukkan nilai terbesar antara 0 dan $H_{net\ j}$. Hasil dari fungsi $f(H_{net\ j})$ merepresentasikan nilai aktivasi pada neuron tersembunyi ke- j , yang selanjutnya diteruskan sebagai masukan ke seluruh neuron pada lapisan output.

- c. . Hitung sinyal yang menuju ke *hidden layer* kedua (G_k). Pada tahap ini informasi diberikan pada input (H_j) yang selanjutnya meneruskan sinyal tersebut ke seluruh neuron pada *hidden layer* kedua.

$$G_{net\ k} = s_{0k} + \sum_{j=1}^n H_j s_{jk} \quad (5.4)$$

Dengan j merupakan node ke- j ($j = 1, 2, \dots, n$) pada lapisan *hidden layer* pertama dan k adalah node ke- k ($k = 1, 2, \dots, p$) pada *hidden layer* kedua. H_j adalah nilai input pada *hidden layer* pertama di dalam node masukan ke- j . Sedangkan s_{0j} adalah bias pada *hidden layer* pertama dan s_{jk} adalah bobot pada node input ke- i yang menuju ke node *hidden layer* kedua ke- k .

- d. Seluruh nilai keluaran pada *hidden layer* dihitung dengan menerapkan fungsi aktivasi *Rectified Linear Units* (ReLU). Hasilnya nanti akan digunakan oleh setiap node pada *hidden layer* kedua (G_k) dan selanjutnya masuk ke *output*.

$$G_k = f(G_{net\ k}) = \max(0, G_{net\ k}) \quad (5.5)$$

Nilai max menunjukkan nilai terbesar antara 0 dan $G_{net\ k}$. Hasil dari perhitungan $f(G_{net\ k})$ merupakan nilai aktivasi pada node hidden ke- k .

- e. Setelah didapatkan nilai tiap node pada *hidden layer* kedua selanjutnya menghitung nilai output (y_k) dengan cara menjumlahkan semua sinyal yang masuk ke output.

$$y_{net\ l} = s_{0l} + \sum_{k=1}^l G_k t_{kl} \quad (5.6)$$

s_{0l} adalah nilai bias pada *hidden layer* kedua, G_k adalah hasil fungsi aktivasi yang keluar dari *hidden layer* kedua, dan t_{kl} adalah bobot pada node *hidden layer* kedua ke- k menuju ke node output (Y_l), k adalah indeks node ke- k (1,2,3,...,m) pada *hidden layer* kedua, dan l adalah indeks data ke l (1,2,3,...,n)

- f. . Setelah diperoleh nilai output (Y_l) kemudian dihitung nilai *loss function* atau selisih antara nilai *PGA* aktual (*ground truth*) dengan nilai *PGA* prediksi menggunakan rumus :

$$Loss\ (L) = (T_l - Y_l)^2 \quad (5.7)$$

$$mse = \frac{1}{n} \sum_k^n L$$

$$= \frac{1}{n} \sum_k^n (T_l - Y_l)^2 \quad (5.8)$$

L adalah fungsi *loss function*, t_l adalah nilai PGA aktual (*ground truth*) dan (y_l) adalah nilai PGA output. Sedangkan l adalah indeks data ke- l (1,2,3....n). Jika waktu konvergensi telah tercapai, maka proses akan berhenti. Sebaliknya jika waktu konvergensi belum tercapai, maka dilakukan proses *backpropagation* dengan memperbarui bobot.

6. Tahap Propagasi Mundur (*Backward Propagation*)

- a. Hitung nilai kesalahan (δ_l) pada lapisan *output* (Y_l) dengan membandingkan output terhadap pola target (t_l). Nilai kesalahan ini kemudian diteruskan ke lapisan sebelumnya untuk digunakan menghitung koreksi bobot dan bias antara *output layer* (Y_l) dan *hidden layer* kedua (G_k)

$$\delta_l = (t_l - y_l) f'(y_{net_l}) \quad (5.9)$$

Dimana:

$$f'(y_{net_l}) = (1 - f(y_{net_l})) f(y_{net_l}) \quad (5.10)$$

- b. Setelah nilai kesalahan pada lapisan *output* (Y_l) dan *hidden layer* kedua (G_k) diperoleh, selanjutnya menghitung nilai kesalahan yang ada pada *hidden layer* kedua (G_k) dan *hidden layer* pertama (H_j)

$$\delta_k = \delta_{net_k} f'(G_{net_k}) \quad (5.11)$$

Dimana:

$$\delta_{net_k} = \sum_{l=1}^m \delta_l s_{kl} \quad (5.12)$$

$$f'(G_{net_k}) = (1 - f(G_{net_k})) f(G_{net_k}) \quad (5.13)$$

- c. Setelah nilai kesalahan pada *hidden layer* kedua dan *hidden layer* pertama diperoleh, maka selanjutnya menghitung nilai kesalahan yang ada pada *hidden layer* pertama (H_j) dan *input layer* (X_i)

$$\delta_j = \delta_{netj} f'(H_{netj}) \quad (5.14)$$

Dimana:

$$\delta_{netj} = \sum_{k=1}^m \delta_k s_{jk} \quad (5.15)$$

$$f'(H_{netj}) = (1 - f(H_{netj})) f(H_{netj}) \quad (5.16)$$

- d. Setelah nilai kesalahan masing-masing *layer* diketahui maka besarnya perubahan atau koreksi yang akan ditambahkan pada bobot lama dapat dihitung.

$$\Delta r_{ij} = \alpha \delta_j X_i \quad (5.17)$$

$$\Delta s_{jk} = \alpha \delta_k H_j \quad (5.18)$$

$$\Delta t_{kl} = \alpha \delta_l G_k \quad (5.19)$$

Δr_{ij} , Δs_{jk} dan Δt_{kl} merupakan perubahan yang akan ditambahkan pada bobot lama yang akan diperbaharui. Dengan α laju pembelajaran (*learning rate*), δ_j nilai kesalahan yang dibawa antara *input layer* dan *hidden layer* pertama, δ_k nilai kesalahan antara *hidden layer* pertama dan *hidden layer* kedua dan δ_l nilai error antara *hidden layer* kedua dan *output layer*. X_i adalah input yang berasal dari node i ke node j sedangkan H_j adalah hasil fungsi aktivasi dari *hidden layer* pertama, dan G_k merupakan hasil fungsi aktivasi dari *hidden layer* kedua.

7. Tahap Perubahan Bobot

Pemperbaikan bobot lama menjadi bobot baru setelah Δr_{ij} dan Δs_{jk} diperoleh.

$$r_{ij}(\text{baru}) = r_{ij}(\text{lama}) + \Delta r_{ij} \quad (5.20)$$

$$s_{jk}(\text{baru}) = s_{jk}(\text{lama}) + \Delta s_{jk} \quad (5.21)$$

$$t_{kl}(\text{baru}) = t_{kl}(\text{lama}) + \Delta t_{kl} \quad (5.22)$$

$r_{ij}(\text{baru})$ merupakan bobot baru antara lapisan *input* dan *hidden layer* pertama yang akan dicari, sedangkan $r_{ij}(\text{lama})$ merupakan bobot lama yang akan

diperbaharui. $s_{jk}(\text{baru})$ adalah bobot baru antara *hidden layer* pertama dan *hidden layer* kedua. $t_{kl}(\text{baru})$ adalah bobot baru antara *hidden layer* kedua dan *output layer* yang akan dicari, sedangkan $t_{kl}(\text{lama})$ adalah bobot lama yang akan diperbaharui.

5.2. Implementasi

Proses implementasi model ANN metode kedua juga dilakukan menggunakan laptop dengan spesifikasi *processor* Intel Core i3 berkecepatan 2.10 GHz dan RAM 4.0 GB. Seluruh pemrograman dilakukan menggunakan bahasa *Python* melalui platform *Google Colab*. Model ANN dibangun menggunakan *package MLPRegressor* dari pustaka *scikit-learn* untuk melakukan pelatihan dan pengujian model.. Parameter pelatihan yang digunakan dalam penelitian ini selengkapnya disajikan pada Tabel 5.1.

Tabel 5.1 Parameter Model ANN Dua *Hidden Layer*

Tipe Jaringan	BP-ANN
Fungsi Aktivasi	<i>Rectified Linear Units</i> (ReLU) <i>hyperbolic tangent</i> (tanh)
Fungsi Optimasi	<i>Adaptive Moment Estimation</i> (Adam)
Learning Rate	0.001
Jumlah Node input layer	3
Jumlah Node Hidden Layer 1 dan 2	3, 6, 9, 12, 15, 30, 45, 60, 75, 90, 105
Jumlah Node Output Layer	1
Komposisi Data	70:30 dan 80:20
Kinerja	MSE (<i>Mean Square Error</i>)

5.3 Uji Coba

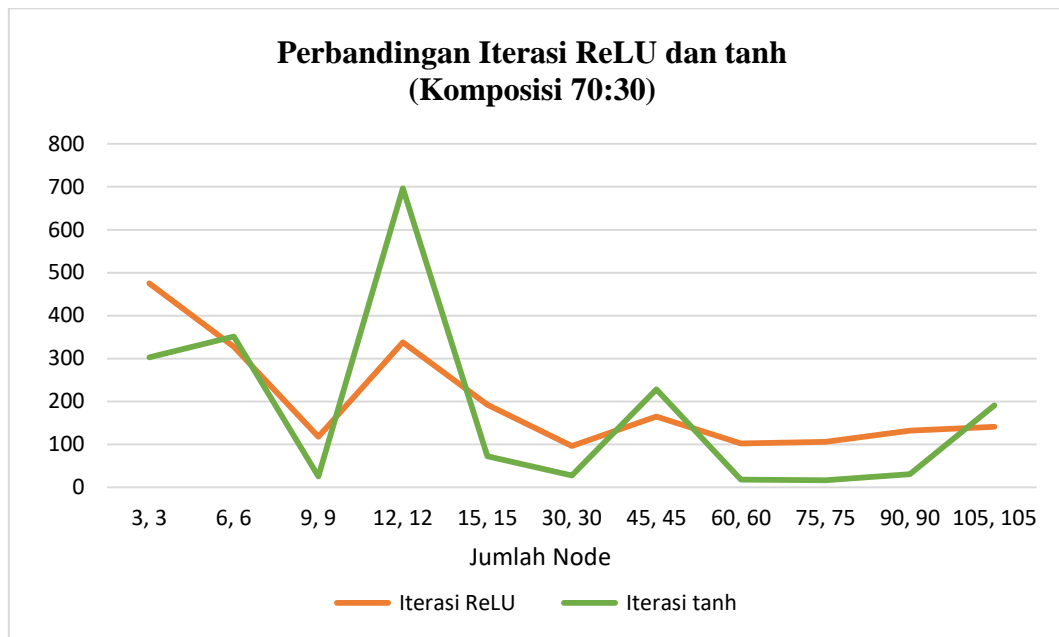
Pada penelitian ini dilakukan uji coba model ANN dua *hidden layer* dengan dua komposisi data *training* dan data *testing*, yaitu 70 : 30 dan 80 : 20. Masing-masing komposisi data tersebut juga dilakukan uji coba performa dari dua fungsi aktivasi, yaitu *Rectified Linear Units* (ReLU) dan *hyperbolic tangent* (*tanh*).

5.3.1 Komposisi Data 70 : 30

Komposisi data 70 : 30 artinya data yang digunakan untuk pelatihan adalah 70 % dari 1828 total data, yaitu 1280 data. Sedangkan data yang digunakan untuk pengujian adalah 30 % dari 1828 total data, yaitu 548 data. Hasil pelatihan dan pengujian selengkapnya disajikan pada tabel 5.2.

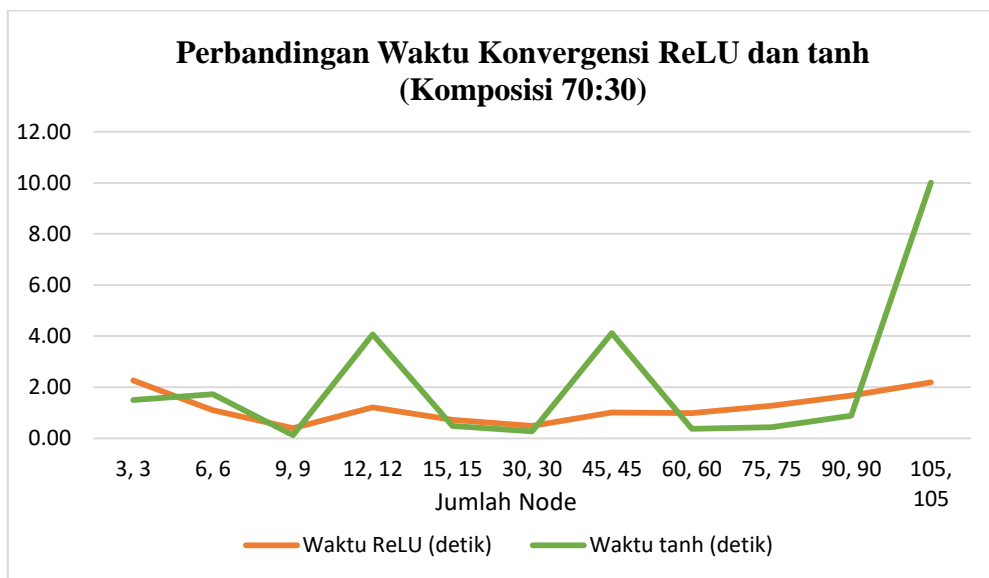
Tabel 5.2 Hasil pelatihan dan pengujian komposisi data 70:30

Nama Arsitektur	Node	Iterasi ReLU	Iterasi tanh	Waktu ReLU (detik)	Waktu tanh (detik)	MSE Training ReLU	MSE Training tanh	MSE Testin g ReLU	MSE Testin g tanh
HL2-A1	3, 3	475	303	2.26	1.50	0.2756	0.3882	0.2643	0.3921
HL2-A2	6, 6	327	351	1.10	1.73	0.2616	0.2717	0.2501	0.2547
HL2-A3	9, 9	118	25	0.39	0.12	0.3735	0.4692	0.3863	0.4682
HL2-A4	12, 12	338	697	1.21	4.07	0.2695	0.2580	0.2539	0.2412
HL2-A5	15, 15	192	72	0.73	0.48	0.2720	0.3775	0.2646	0.3853
HL2-A6	30, 30	96	27	0.48	0.27	0.2459	0.3739	0.2379	0.3828
HL2-A7	45, 45	165	228	1.00	4.11	0.2468	0.2670	0.2392	0.2512
HL2-A8	60, 60	102	18	0.99	0.37	0.2372	0.3711	0.2282	0.3818
HL2-A9	75, 75	106	16	1.27	0.43	0.2392	0.3819	0.2260	0.3880
HL2-A10	90, 90	132	30	1.67	0.89	0.2393	0.3685	0.2318	0.3803
HL2-A11	105, 105	141	191	2.19	10.01	0.2332	0.2662	0.2282	0.2473



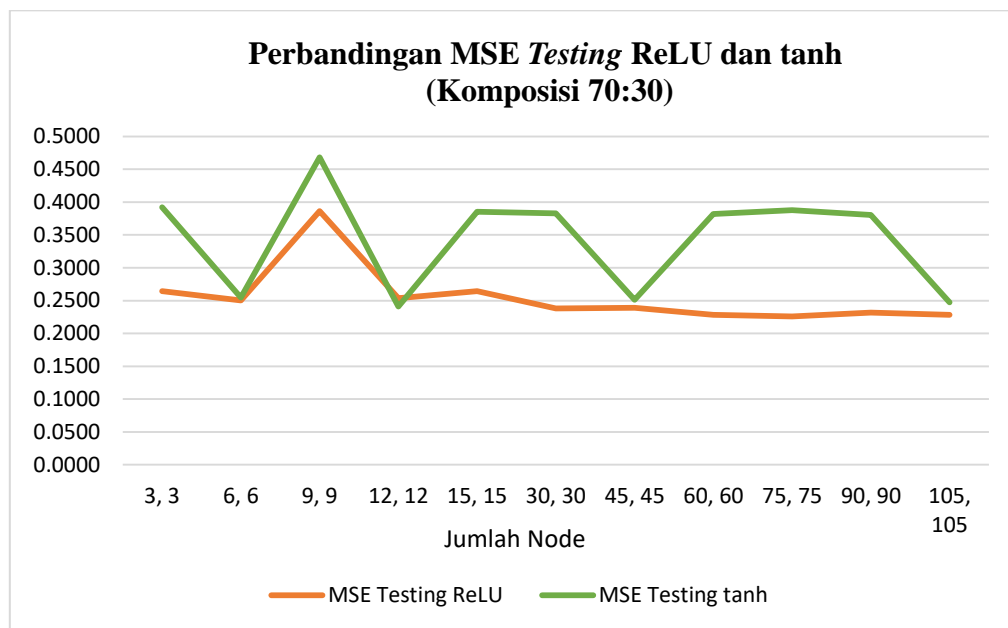
Gambar 5.2 Perbandingan Iterasi ReLU dan tanh (Komposisi 70:30)

Gambar 5.2 merupakan perbandingan jumlah iterasi pelatihan antara fungsi aktivasi ReLU dan tanh pada arsitektur ANN dua *hidden layer* dengan Komposisi data 70:30. Iterasi terbanyak terjadi pada fungsi aktivasi tanh, yaitu pada jumlah node menengah yaitu (12,12). Pada node (3,3) ReLU membutuhkan iterasi paling banyak, yaitu 475 iterasi, sedangkan untuk tanh membutuhkan 303 iterasi. Hal ini menunjukkan bahwa pada jaringan yang sangat kecil, proses pembelajaran belum berjalan optimal sehingga memerlukan lebih banyak epoch untuk mencapai konvergensi. Ketika jumlah node ditingkatkan menjadi sedang (45,45 hingga 105,105) jumlah iterasi yang dibutuhkan oleh ReLU menurun secara signifikan dan relatif stabil, berada pada kisaran 96–132 iterasi. Hal ini mengindikasikan bahwa ReLU mampu melakukan konvergensi lebih cepat pada jaringan yang lebih kompleks. Sedangkan tanh dari node kecil hingga besar konvergensinya cenderung fluktuatif.



Gambar 5.3 Perbandingan Waktu Konvergensi ReLU dan tanh (Komposisi 70:30)

Gambar 5.3 merupakan perbandingan waktu konvergensi pelatihan antara fungsi aktivasi ReLU dan tanh pada arsitektur ANN dua *hidden layer* dengan Komposisi data 70:30. Grafik waktu konvergensi menunjukkan bahwa secara umum waktu pelatihan menggunakan ReLU lebih efisien dibandingkan tanh pada sebagian besar variasi jumlah node. Pada node terkecil (3,3), waktu konvergensi masih relatif tinggi untuk ReLU, namun masih lebih cepat dibandingkan tanh pada beberapa arsitektur lain yang memiliki node lebih besar. Ketika jumlah node meningkat hingga (30,30), waktu konvergensi ReLU menurun, menunjukkan bahwa ReLU sangat efisien secara komputasi pada jaringan dengan dua *hidden layer*. Sebaliknya, fungsi aktivasi tanh menunjukkan lonjakan waktu konvergensi yang signifikan, terutama pada arsitektur node (12,12), (45,45), dan (105,105). Pada node (105,105), tanh membutuhkan waktu hingga 10,01 detik, hampir 5 kali lebih lama dibandingkan ReLU pada arsitektur yang sama (2,19 detik). Hal ini disebabkan oleh sifat fungsi tanh yang lebih kompleks secara komputasi serta proses gradien yang lebih lambat dibandingkan ReLU.

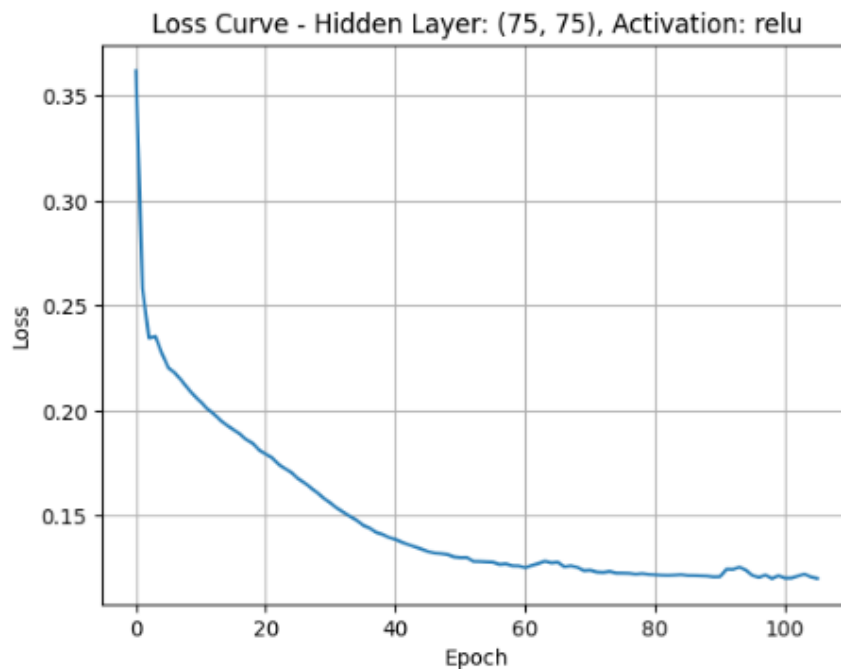


Gambar 5.4 Perbandingan MSE Testing ReLU dan tanh (Komposisi 70:30)

Gambar 5.4 merupakan perbandingan MSE *testing* antara fungsi aktivasi ReLU dan tanh pada arsitektur ANN dua *hidden layer* dengan Komposisi data 70:30. Pada jumlah node kecil (3,3 - 9,9) nilai MSE *testing* ReLU dan tanh relatif tinggi dan tidak stabil. Hal ini menunjukkan bahwa jumlah node terlalu sedikit belum mampu memodelkan kompleksitas hubungan antar variabel yang mempengaruhi nilai *Peak Ground Acceleration* (PGA). Namun, ketika jumlah node ditingkatkan, tren yang sangat jelas mulai terlihat, di mana fungsi aktivasi ReLU secara konsisten menghasilkan nilai MSE *testing* yang lebih rendah dibandingkan tanh.

Berdasarkan tabel 5.3 hasil pengujian arsitektur ANN dua *hidden layer* dengan Komposisi data 70:30 menggunakan fungsi aktivasi ReLU, diperoleh performa terbaik pada arsitektur HL2-A9 dengan jumlah node (75,75), nilai MSE *testing* sebesar 0.2260, jumlah iterasi 106 dan waktu konvergensi 1.27 detik. Sedangkan pada penggunaan fungsi aktivasi tanh, performa terbaik diperoleh pada arsitektur HL2-A4 dengan jumlah node (12,12), nilai MSE *testing* sebesar 0.2412, jumlah iterasi 697 dan waktu konvergensi 4.07

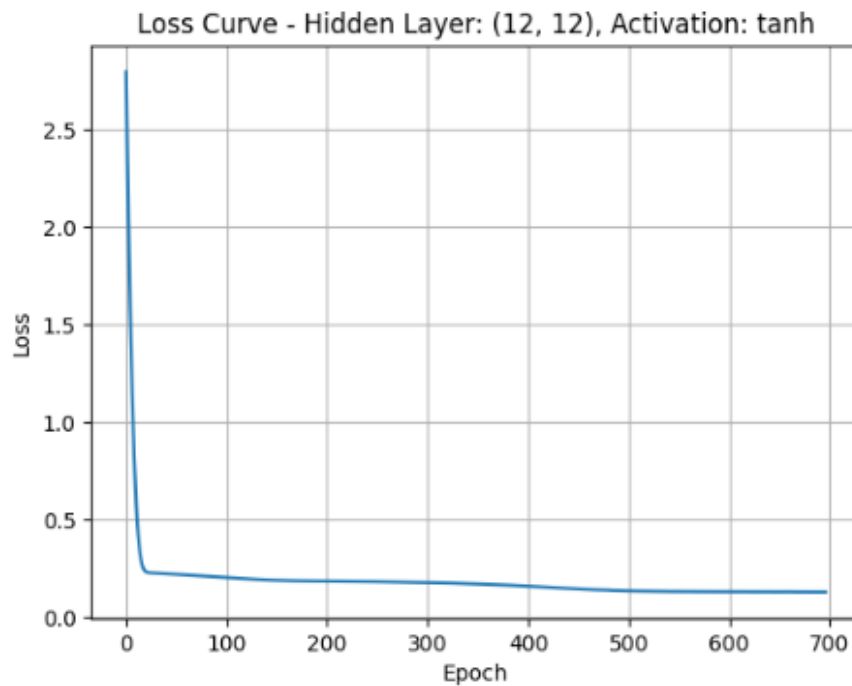
detik. Untuk melihat pola konvergensi selama proses pelatihan dari arsitektur HL2-A9 dan HL2-A4 ditampilkan grafik *loss function* sebagaimana ditunjukkan pada Gambar 5.5 dan 5.6



Gambar 5.5 Grafik *loss function* arsitektur HL2-A9

Gambar 5.5 merupakan grafik *loss function* selama proses pelatihan model ANN dengan arsitektur HL2-A9 (dua *hidden layer* dengan masing-masing 75 *node*) menggunakan fungsi aktivasi *ReLU* dan komposisi data 70:30. Berdasarkan grafik tersebut terlihat bahwa penurunan nilai *loss* yang sangat tajam pada awal iterasi (*epoch* 1 – 3), dari kisaran sekitar 0,37 menjadi sekitar 0,23. Hal ini mengindikasikan bahwa model mampu dengan cepat mempelajari pola dasar hubungan antara parameter input (magnitudo, jarak, dan Vs30) terhadap nilai output *Peak Ground Acceleration* (PGA). Pada fase selanjutnya (sekitar *epoch* 3 – 60), penurunan *loss* berlangsung lebih gradual hingga mencapai nilai sekitar 0,13 yang menunjukkan proses pembelajaran mulai

memasuki fase penyempurnaan. Setelah melewati *epoch* 60 hingga akhir pelatihan (*epoch* 106), kurva *loss* tampak relatif stabil di kisaran 0,12 – 0,11.



Gambar 5.6 Grafik *loss function* arsitektur HL2-A4

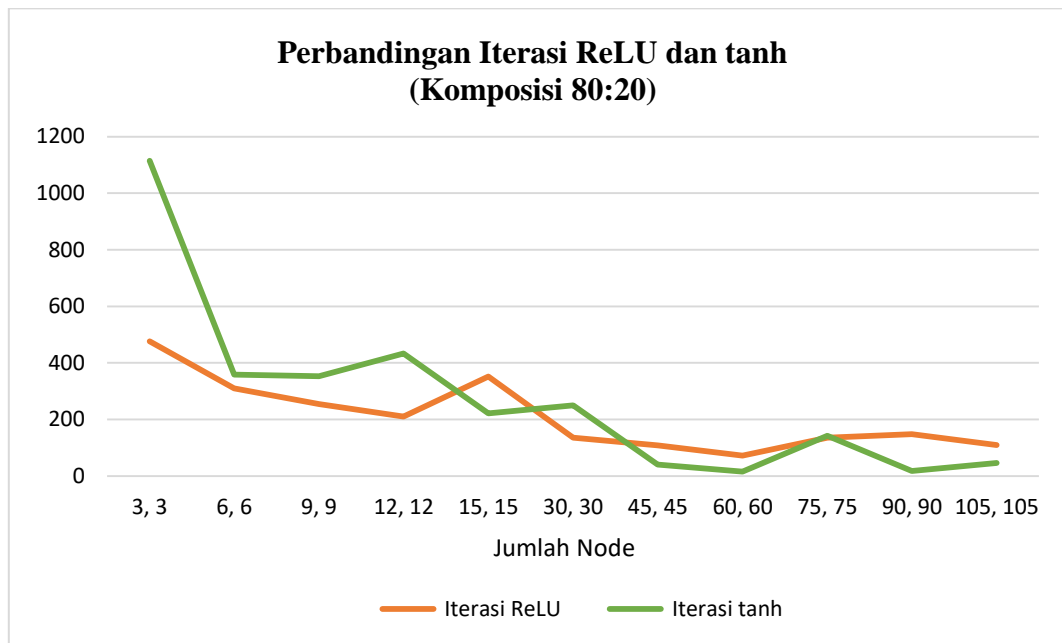
Gambar 5.6 merupakan grafik *loss function* selama proses pelatihan model ANN dengan arsitektur HL2-A4 (dua *hidden layer* dengan masing-masing 12 *node*) menggunakan fungsi aktivasi tanh dan komposisi data 70:30. Berdasarkan grafik tersebut terlihat bahwa penurunan nilai *loss* yang sangat signifikan pada fase awal pelatihan (*epoch* 1 – 30), dari kisaran sekitar 2,8 menjadi sekitar 0,3. Hal ini mengindikasikan bahwa model mampu dengan cepat menyesuaikan bobot awal dan mulai mengenali pola dasar hubungan antara parameter input dan nilai output *Peak Ground Acceleration* (PGA). Kemudian pada sekitar *epoch* 30 hingga akhir iterasi (*epoch* 697) kurva *loss* terlihat semakin landai dan stabil di kisaran 0,2 – 0,1 tanpa adanya fluktuasi yang tajam.

5.3.2 Komposisi Data 80 : 20

Komposisi data 80 : 20 artinya data yang digunakan untuk pelatihan adalah 80 % dari 1828 total data , yaitu 1462 data. Sedangkan data yang digunakan untuk pengujian adalah 20 % dari 1828 total data, yaitu 366 data. Hasil pelatihan dan pengujian selengkapnya disajikan pada tabel 5.3.

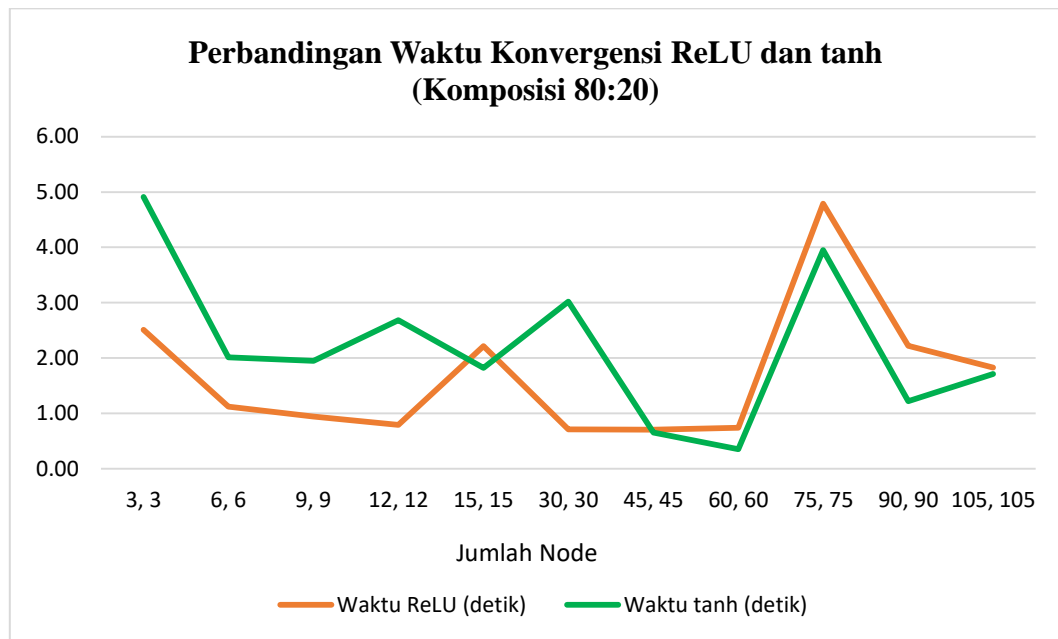
Tabel 5.3 Hasil pelatihan dan pengujian komposisi data 80:20

Nama Arsitektur	Node	Iterasi ReLU	Iterasi tanh	Waktu ReLU (detik)	Waktu tanh (detik)	MSE Training ReLU	MSE Training tanh	MSE Testin g ReLU	MSE Testin g tanh
HL2-B1	3, 3	476	1115	2.51	4.91	0.2697	0.2646	0.2532	0.2394
HL2-B2	6, 6	310	358	1.12	2.01	0.2545	0.2677	0.2427	0.2481
HL2-B3	9, 9	254	353	0.94	1.95	0.2647	0.2742	0.2526	0.2626
HL2-B4	12, 12	210	433	0.79	2.68	0.2771	0.2691	0.2584	0.2510
HL2-B5	15, 15	352	222	2.21	1.82	0.2480	0.2648	0.2416	0.2460
HL2-B6	30, 30	136	250	0.71	3.02	0.2361	0.2515	0.2405	0.2431
HL2-B7	45, 45	108	40	0.70	0.65	0.2475	0.3779	0.2455	0.3662
HL2-B8	60, 60	72	16	0.74	0.35	0.2384	0.3783	0.2366	0.3658
HL2-B9	75, 75	136	142	4.79	3.95	0.2295	0.2642	0.2351	0.2422
HL2-B10	90, 90	148	18	2.22	1.22	0.2346	0.3773	0.2344	0.3649
HL2-B11	105, 105	109	46	1.82	1.71	0.2317	0.3735	0.2406	0.3674



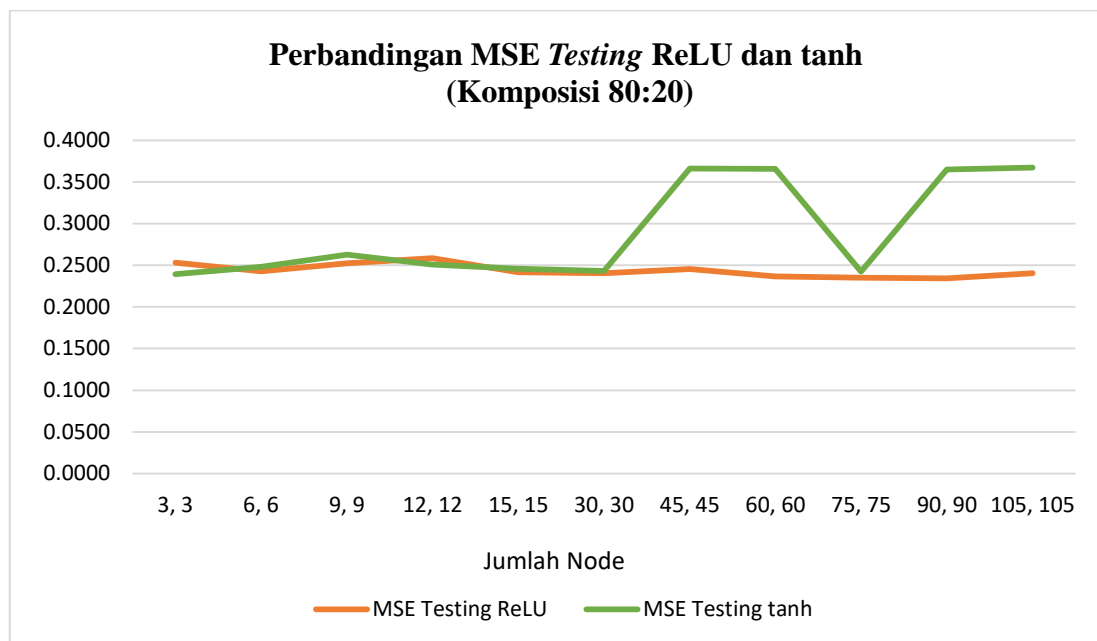
Gambar 5.7 Perbandingan Iterasi ReLU dan tanh (Komposisi 80:20)

Gambar 5.7 merupakan perbandingan jumlah iterasi pelatihan antara fungsi aktivasi ReLU dan tanh pada arsitektur ANN dua *hidden layer* dengan Komposisi data 80:20. Pada jumlah node kecil, khususnya pada arsitektur node (3,3), fungsi aktivasi tanh membutuhkan iterasi yang sangat besar, yaitu 1115 iterasi, sedangkan ReLU hanya membutuhkan 476 iterasi. Hal ini menunjukkan bahwa pada jaringan yang masih kecil, fungsi tanh mengalami kesulitan dalam mencapai kondisi konvergen, sehingga membutuhkan pembaruan bobot yang lebih banyak dibandingkan ReLU. Dengan bertambahnya jumlah node, jumlah iterasi ReLU menunjukkan kecenderungan menurun dan lebih stabil, terutama pada arsitektur node (30,30 - 60,60). Hal ini mengindikasikan bahwa penambahan jumlah neuron mampu meningkatkan kapasitas model sehingga proses pembelajaran menjadi lebih efisien. Sementara itu, jumlah iterasi pada fungsi tanh cenderung lebih fluktuatif. Secara umum fungsi aktivasi ReLU lebih konsisten dalam mencapai konvergensi yang stabil pada jumlah node menengah hingga besar dibandingkan fungsi tanh.



Gambar 5.8 Perbandingan Waktu Konvergensi ReLU dan tanh (Komposisi 80:20)

Gambar 5.8 merupakan perbandingan waktu konvergensi pelatihan antara fungsi aktivasi ReLU dan tanh pada arsitektur ANN dua *hidden layer* dengan Komposisi data 80:20. Pada jumlah node kecil (3,3), waktu konvergensi tanh mencapai 4,91 detik, hampir dua kali lipat lebih lama dibandingkan ReLU yang membutuhkan 2,51 detik. Ketika jumlah node ditingkatkan ke ukuran menengah (45,45 – 60,60), waktu konvergensi ReLU semakin kecil, bahkan berada di bawah 1 detik pada beberapa arsitektur. Hal ini membuktikan bahwa ReLU memiliki efisiensi yang tinggi pada jaringan yang lebih kompleks. Sebaliknya, fungsi tanh meskipun pada (45,45 – 60,60) menunjukkan waktu yang relatif singkat, namun pada arsitektur lain justru mengalami peningkatan waktu secara signifikan. Kondisi ini menggambarkan bahwa performa komputasi tanh kurang stabil ketika diterapkan pada jaringan dengan jumlah neuron yang bervariasi.

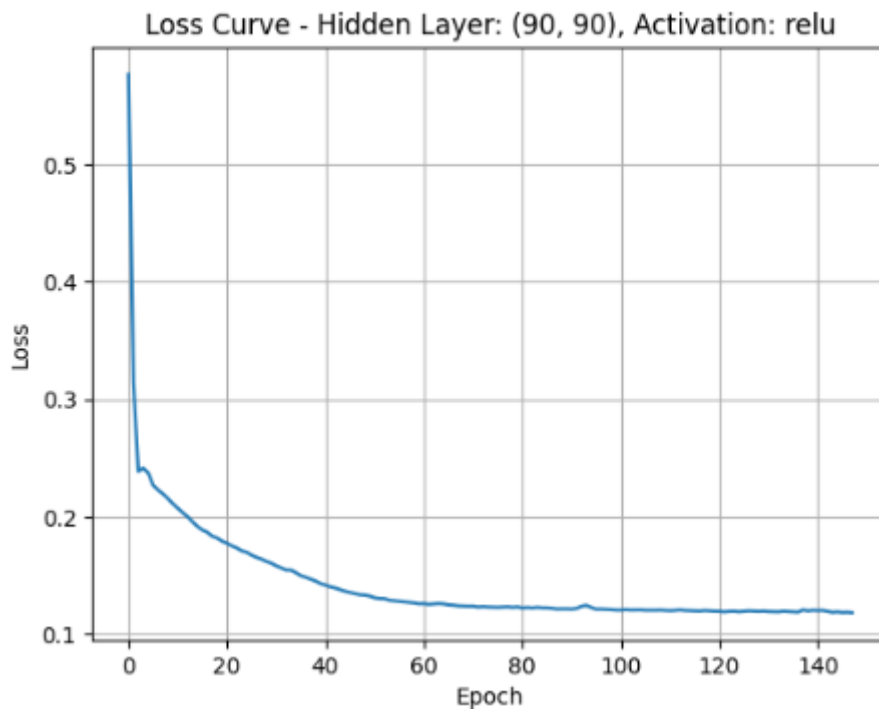


Gambar 5.9 Perbandingan MSE *Testing* ReLU dan tanh (Komposisi 80:20)

Gambar 5.9 merupakan perbandingan MSE *testing* antara fungsi aktivasi ReLU dan tanh pada arsitektur ANN dua *hidden layer* dengan Komposisi data 80:20. Pada jumlah node kecil (3–15), performa kedua fungsi aktivasi masih relatif sebanding. Namun, ketika jumlah node ditingkatkan mulai (45,45), terlihat tren yang semakin jelas di mana fungsi aktivasi ReLU secara konsisten menghasilkan nilai MSE *testing* yang lebih rendah. Sedangkan tanh justru menunjukkan peningkatan nilai MSE testing yang cukup signifikan pada node yang lebih besar, khususnya pada node (45,45 – 105,105). Fenomena ini mengindikasikan adanya kesulitan tanh dalam melakukan generalisasi pada struktur jaringan yang lebih besar, dan mengarah pada kemungkinan terjadinya overfitting terhadap data *training*.

Berdasarkan tabel 5.3, hasil pengujian arsitektur ANN dua *hidden layer* dengan Komposisi data 80:20 menggunakan fungsi aktivasi ReLU, diperoleh performa terbaik pada arsitektur HL2-B10 dengan jumlah node (90,90), nilai MSE *testing* sebesar 0.2344, jumlah iterasi 1482 dan waktu konvergensi 2.22 detik. Sedangkan pada penggunaan

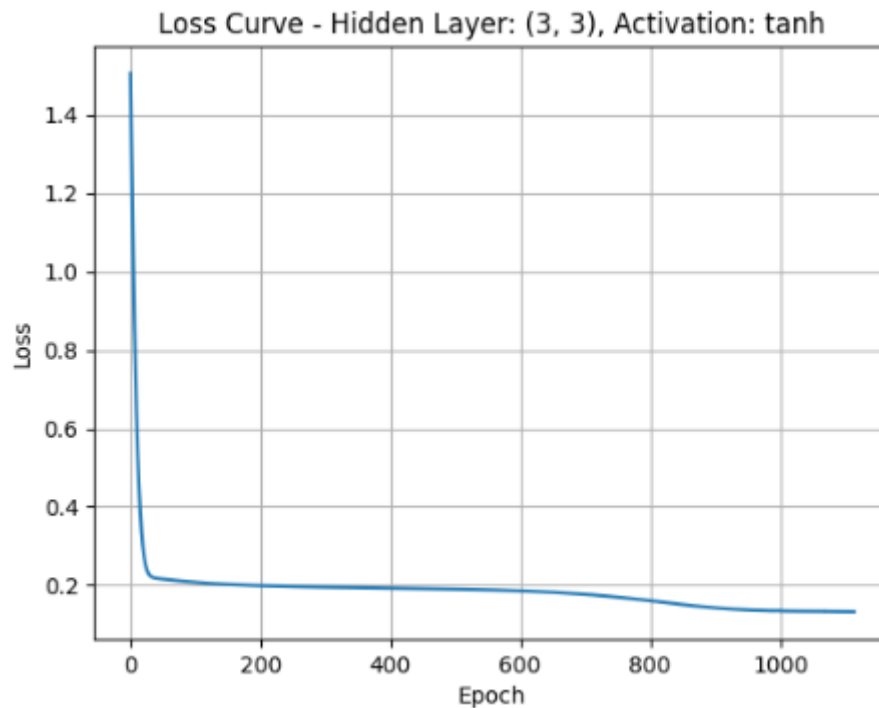
fungsi aktivasi tanh, performa terbaik diperoleh pada arsitektur HL2-B1 dengan jumlah node (3,3), nilai MSE *testing* sebesar 0.2394, jumlah iterasi 1115 dan waktu konvergensi 4.91 detik. Untuk melihat pola konvergensi selama proses pelatihan dari arsitektur HL2-B10 dan HL2-B1 ditampilkan grafik *loss function* sebagaimana ditunjukkan pada Gambar 5.10 dan 5.11.



Gambar 5.10 Grafik *loss function* arsitektur HL2-B10

Gambar 5.10 merupakan grafik *loss function* selama proses pelatihan model ANN dengan arsitektur HL2-B10 (dua *hidden layer* dengan masing-masing 90 *node*) menggunakan fungsi aktivasi *ReLU* dan komposisi data 80:20. Berdasarkan grafik tersebut terlihat bahwa penurunan nilai *loss* yang sangat tajam pada awal iterasi (*epoch* 1 – 5), dari kisaran sekitar 0,58 menjadi sekitar 0,23. Hal ini mengindikasikan bahwa model mampu dengan cepat mempelajari pola dasar hubungan antara parameter input (magnitudo, jarak, dan Vs30) terhadap nilai output *Peak Ground Acceleration* (PGA). Pada fase selanjutnya (sekitar *epoch* 5 – 60), penurunan *loss* berlangsung lebih gradual

hingga mencapai nilai sekitar 0,13. Setelah melewati *epoch* 60 hingga akhir pelatihan (*epoch* 148), kurva *loss* tampak relatif stabil di kisaran 0,12 – 0,11 yang menandakan bahwa model telah mencapai kondisi konvergen.



Gambar 5.11 Grafik *loss function* arsitektur HL2-B1

Gambar 5.11 merupakan grafik *loss function* selama proses pelatihan model ANN dengan arsitektur HL2-B1 (dua *hidden layer* dengan masing-masing 3 *node*) menggunakan fungsi aktivasi tanh dan komposisi data 80:20. Berdasarkan grafik tersebut terlihat bahwa penurunan nilai *loss* yang sangat signifikan pada fase awal pelatihan (*epoch* 1 – 40), dari kisaran sekitar 1,5 menjadi sekitar 0,25. Hal ini menunjukkan model mampu dengan cepat menyesuaikan bobot awal dan mulai mengenali pola dasar hubungan antara parameter input dan nilai output. Kemudian pada sekitar *epoch* 40 hingga akhir iterasi (*epoch* 1115) kurva *loss* terlihat semakin landai dan stabil di kisaran 0.1 tanpa adanya fluktuasi yang tajam.

5.4 Arsitektur ANN Mulai Dari Satu *Hidden Layer*

Uji coba dalam memprediksi nilai *Peak Ground Acceleration* (PGA) juga dilakukan mulai dari satu hidden layer hingga beberapa *hidden layer* untuk mendapatkan performa yang terbaik. Dalam uji coba ini masing-masing *hidden layer* menggunakan kombinasi terbaik dari uji coba sebelumnya, yaitu jumlah node 75, komposisi data 70:30 dan fungsi aktivasi ReLU. Hasil uji coba mulai satu *hidden layer* hingga beberapa *hidden layer* disajikan pada tabel 5.4.

Tabel 5.4 Hasil Uji Coba Mulai Dari Satu *Hidden Layer*

<i>Hidden Layer</i>	Iterasi	Waktu Konvergensi	MSE Testing	Korelasi
(75,)	294	1.17	0.2349	0.6780
(75, 75)	106	4.39	0.2260	0.6989
(75, 75, 75)	62	1.00	0.2344	0.6942

Berdasarkan Tabel 5.4 arsitektur satu *hidden layer* nilai MSE nya adalah 0.2349 dan nilai korelasi 0.6780. Arsitektur dua *hidden layer* memiliki nilai MSE 0.2260 dan nilai korelasi 0.6989. Sedangkan arsitektur tiga *hidden layer* memiliki nilai MSE 2344

BAB 6

PEMBAHASAN

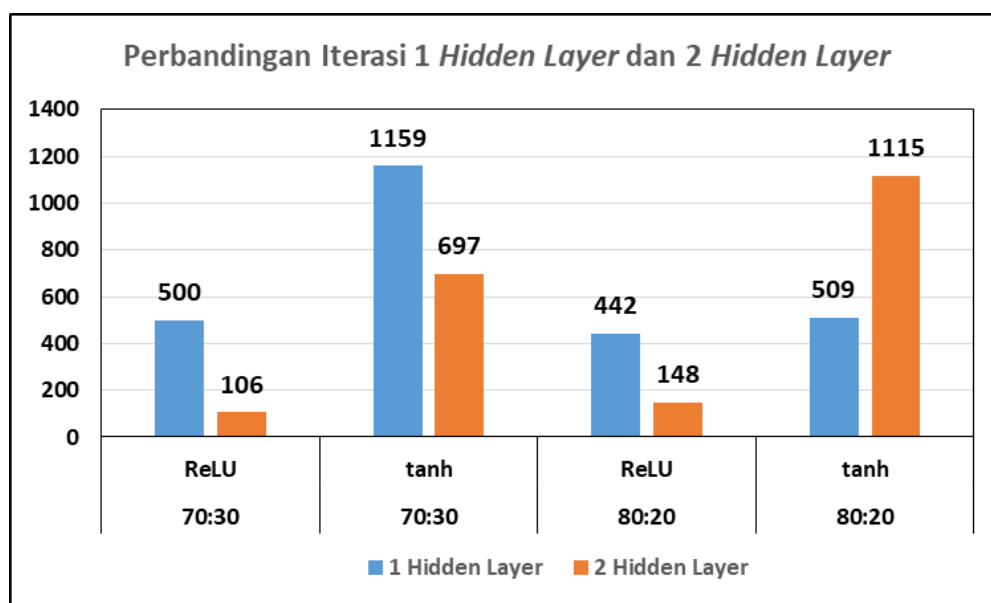
Penelitian ini menggunakan dua skenario yaitu metode ANN satu *hidden layer* dan metode ANN dua *hidden layer*. Kemudian masing-masing metode tersebut dilakukan uji coba untuk mendapatkan performa terbaik berdasarkan nilai MSE terkecil dalam memprediksi *Peak Ground Acceleration* (PGA) di pulau Jawa. Uji coba dilakukan dengan memvariasikan node pada *hidden layer* (3, 6, 9, 12, 15, 30, 45, 60, 75, 90, 105), komposisi data latih dan data uji (70:30 dan 80 : 20.), dan fungsi aktivasi (ReLU dan tanh). Hasil performa terbaik dari masing-masing uji coba disajikan pada tabel 6.1.

Tabel 6.1 Hasil performa terbaik ANN 1 *hidden layer* dan 2 *hidden layer*

Arsitektur ANN	Nama Arsitektur	Node	Komposisi data <i>train</i> dan <i>test</i>	Fungsi Aktivasi	MSE <i>Testing</i>	Iterasi	Waktu Konvergensi (detik)
1 Hidden Layer	HL1-A11	105	70 : 30	ReLU	0.2282	500	2.11
1 Hidden Layer	HL1-A2	6	70 : 30	tanh	0.2617	1159	4.95
1 Hidden Layer	HL1-B11	105	80 : 20	ReLU	0.2300	442	2.16
1 Hidden Layer	HL1-B5	15	80 : 20	tanh	0.2570	509	2.28
2 Hidden Layer	HL2-A9	75, 75	70 : 30	ReLU	0.2260	106	1.27
2 Hidden Layer	HL2-A4	12, 12	70 : 30	tanh	0.2412	697	4.07
2 Hidden Layer	HL2-B10	90, 90	80 : 20	ReLU	0.2344	148	2.22
2 Hidden Layer	HL2-B1	3, 3	80 : 20	tanh	0.2394	1115	4.91

6.1 Perbandingan Iterasi ANN Satu Hidden Layer dan Dua Hidden Layer

Jumlah iterasi merupakan salah satu indikator yang digunakan dalam mengevaluasi kinerja proses pelatihan *Artificial Neural Network* (ANN). Semakin sedikit jumlah iterasi yang dibutuhkan untuk mencapai konvergensi, maka semakin efisien model tersebut dalam melakukan pembelajaran, baik dari sisi waktu maupun kompleksitas komputasi. Perbandingan iterasi satu *hidden layer* dan dua *hidden layer* disajikan pada gambar 6.1.



Gambar 6.1 Perbandingan Iterasi ANN Satu *Hidden Layer* dan dua *Hidden Layer*

Berdasarkan gambar 6.1 terlihat adanya perbedaan yang cukup signifikan antara jumlah iterasi pada arsitektur ANN dengan 1 *hidden layer* dan 2 *hidden layer*, baik pada variasi Komposisi data maupun fungsi aktivasi yang digunakan. Pada Komposisi data 70:30 dengan fungsi aktivasi ReLU, model dengan satu hidden layer membutuhkan 500 iterasi, sedangkan model dengan 2 hidden layer hanya membutuhkan 106 iterasi untuk mencapai kondisi konvergen. Hal ini menunjukkan bahwa penambahan 1 hidden layer

mampu mempercepat proses pembelajaran secara signifikan. Dengan struktur jaringan yang lebih dalam, model mampu membentuk representasi fitur yang lebih efektif sehingga proses optimasi berjalan lebih cepat dan stabil.

Perbedaan yang serupa juga terlihat pada Komposisi 80:20 dengan aktivasi ReLU, di mana jumlah iterasi pada satu hidden layer sebesar 442 iterasi, sementara pada 2 hidden layer menurun drastis menjadi 148 iterasi. Temuan ini kembali menguatkan bahwa penambahan *hidden layer* pada jaringan yang menggunakan fungsi aktivasi ReLU dapat meningkatkan efisiensi proses pembelajaran, meskipun jumlah data latih lebih besar. Berbeda dengan ReLU, pada fungsi aktivasi tanh, jumlah iterasi cenderung jauh lebih besar. Pada Komposisi 70:30, model dengan 1 hidden layer memerlukan 1159 iterasi, sedangkan model dengan 2 hidden layer membutuhkan 697 iterasi. Meskipun terjadi penurunan jumlah iterasi, nilai yang dihasilkan masih relatif tinggi dibandingkan dengan penggunaan fungsi aktivasi ReLU. Hal ini mengindikasikan bahwa fungsi aktivasi tanh memiliki laju konvergensi yang lebih lambat dalam mempelajari pola data PGA yang bersifat kompleks dan nonlinier.

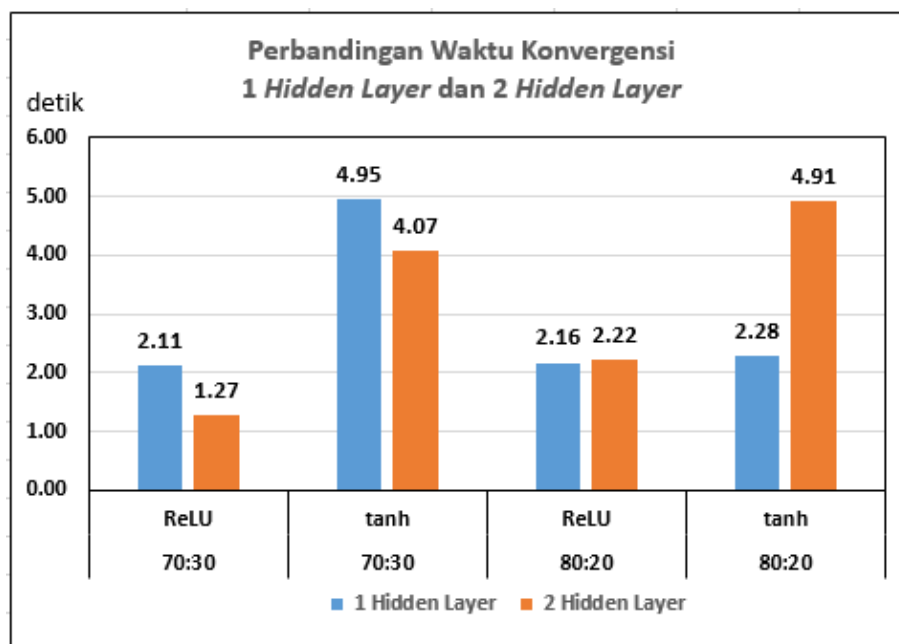
Kondisi paling kontras terlihat pada Komposisi data 80:20 dengan fungsi aktivasi tanh, di mana model dengan 2 *hidden layer* justru membutuhkan 1115 iterasi, lebih tinggi dibandingkan dengan 1 hidden layer yang hanya memerlukan 509 iterasi. Hasil ini mengindikasikan bahwa penambahan hidden layer pada fungsi aktivasi tanh, khususnya pada data latih yang lebih besar, tidak selalu berdampak positif terhadap kecepatan konvergensi.

Secara umum, dari sisi jumlah iterasi dapat disimpulkan bahwa model dengan 2 hidden layer dan fungsi aktivasi ReLU menunjukkan jumlah iterasi paling sedikit, baik pada Komposisi 70:30 maupun 80:20. Ini menunjukkan bahwa arsitektur tersebut memiliki proses pembelajaran yang paling efisien. Fungsi aktivasi tanh cenderung

membutuhkan jumlah iterasi yang jauh lebih besar dibandingkan ReLU, yang mengindikasikan proses pelatihan yang lebih lambat. Penambahan hidden layer tidak selalu menjamin jumlah iterasi menjadi lebih kecil, terutama ketika digunakan oleh fungsi aktivasi tanh dengan Komposisi data latih yang lebih besar. Kombinasi fungsi aktivasi ReLU , 2 hidden layer dan Komposisi data 70:30 merupakan konfigurasi yang paling optimal dari sisi efisiensi iterasi, dengan hanya 106 iterasi. Kombinasi ini terdapat pada naam pelatihan HL2-A9.

6.2 Perbandingan Waktu Konvergensi ANN Satu Hidden Layer dan Dua Hidden Layer

Waktu konvergensi merupakan durasi yang diperlukan oleh model *Artificial Neural Network* (ANN) selama proses pelatihan hingga fungsi *error* mencapai kondisi stabil atau minimum, yang menandakan bahwa model telah mempelajari pola data secara optimal dan proses pembaruan bobot telah berhenti. Perbandingan waktu konvergensi satu *hidden layer* dan dua *hidden layer* disajikan pada gambar 6.2.



Gambar 6.2 Perbandingan Waktu Konvergensi ANN Satu *Hidden Layer* dan dua *Hidden Layer*

Berdasarkan gambar 6.2 terlihat bahwa komposisi data 70:30, arsitektur dengan 2 hidden layer menggunakan fungsi aktivasi ReLU menunjukkan waktu konvergensi paling cepat, yaitu 1,27 detik, dibandingkan dengan 1 hidden layer ReLU yang memerlukan waktu 2,11 detik. Hal ini mengindikasikan bahwa pada Komposisi data ini, penambahan satu hidden layer justru membuat proses pembelajaran menjadi lebih efisien ketika menggunakan ReLU. Pola yang sama juga terlihat pada penggunaan fungsi aktivasi tanh, di mana 2 hidden layer tanh memiliki waktu konvergensi 4,07 detik, lebih cepat dibandingkan 1 hidden layer tanh yang mencapai 4,95 detik. Ini menunjukkan bahwa pada pembagian data 70:30, arsitektur dua hidden layer secara umum lebih cepat mencapai kondisi konvergen, baik pada ReLU maupun tanh.

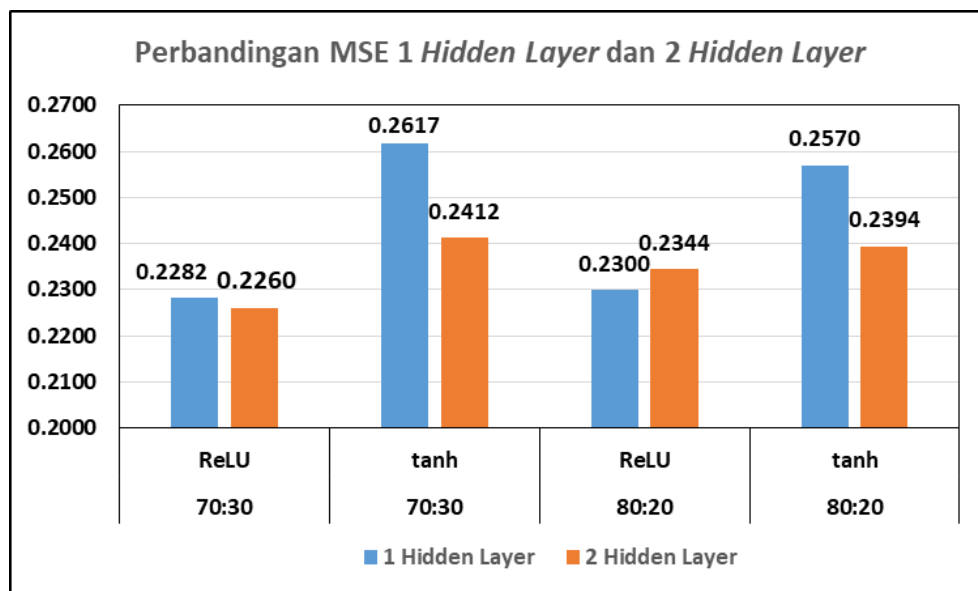
Sebaliknya, pada komposisi data 80:20, pola yang sedikit berbeda mulai muncul. Untuk fungsi aktivasi ReLU, waktu konvergensi antara 1 hidden layer (2,16 detik) dan 2 hidden layer (2,22 detik) menunjukkan nilai yang hampir sama, dengan 1 hidden layer sedikit lebih cepat. Hal ini mengindikasikan bahwa pada Komposisi data pelatihan yang lebih besar, peningkatan jumlah hidden layer tidak selalu berbanding lurus dengan peningkatan kecepatan konvergensi. Sementara itu, pada fungsi aktivasi tanh, perbedaan waktu konvergensi justru cukup signifikan. 1 hidden layer tanh memiliki waktu 2,28 detik, sedangkan 2 hidden layer tanh mencapai 4,91 detik. Ini menunjukkan bahwa pada komposisi data 80:20, penggunaan tanh pada arsitektur yang lebih dalam cenderung memperlambat proses konvergensi.

Secara umum hasil perbandingan waktu konvergensi menunjukkan bahwa ReLU cenderung lebih stabil dan cepat konvergen pada berbagai variasi jumlah hidden layer dan Komposisi data. Penambahan hidden layer tidak selalu mempercepat konvergensi, efektivitasnya sangat dipengaruhi oleh fungsi aktivasi dan komposisi data. Fungsi aktivasi tanh lebih sensitif terhadap peningkatan kompleksitas jaringan, terutama pada

Komposisi data 80:20, sehingga waktu konvergensi menjadi lebih lama pada arsitektur dua hidden layer. Dengan demikian, dari sisi efisiensi waktu konvergensi, kombinasi terbaik ditunjukkan oleh model ANN dua *hidden layer* dengan fungsi aktivasi ReLU pada Komposisi data 70:30, karena memiliki waktu konvergensi paling singkat dibandingkan konfigurasi lainnya, yaitu 1,27 detik. Kombinasi ini terdapat pada nama pelatihan HL2-A9.

6.3 Perbandingan MSE ANN Satu Hidden Layer dan Dua Hidden Layer

Mean Squared Error (MSE) merupakan ukuran besar kesalahan (*error*) rata-rata antara nilai prediksi model dengan nilai sebenarnya, yang dihitung dalam bentuk kuadrat. *MSE* menunjukkan seberapa jauh hasil prediksi model menyimpang dari data sebenarnya atau *ground truth*. Perbandingan MSE ANN satu hidden layer dan dua hidden layer disajikan pada gambar 6.3.



Gambar 6.3 Perbandingan MSE Satu Hidden Layer dan dua Hidden Layer

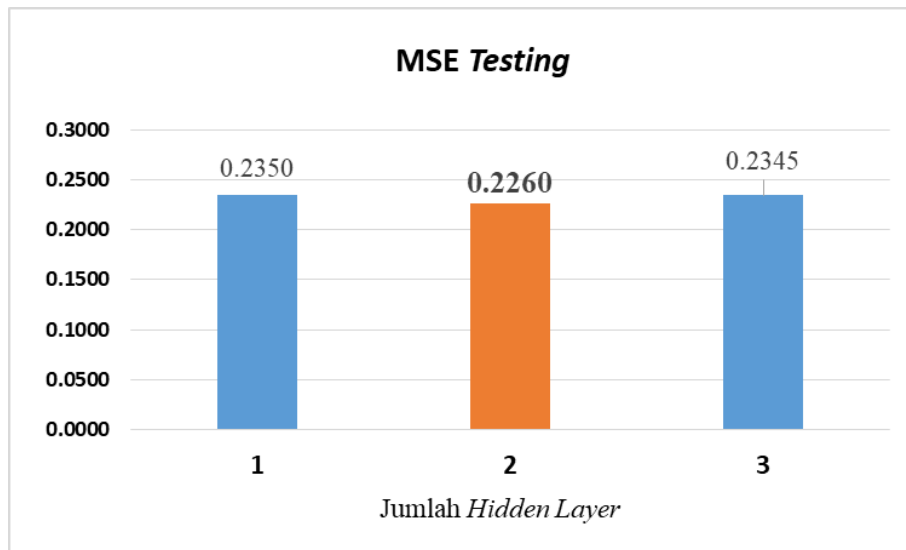
Berdasarkan gambar 6.3 dapat dilihat bahwa Komposisi data 70:30 dengan fungsi aktivasi ReLU, nilai MSE untuk 1 hidden layer sebesar 0,2282, sedangkan pada 2 hidden

layer menurun menjadi 0,2260. Penurunan nilai MSE ini menunjukkan bahwa penambahan satu hidden layer memberikan peningkatan akurasi prediksi meskipun selisihnya relatif kecil. Artinya, model dengan 2 hidden layer mampu melakukan generalisasi sedikit lebih baik terhadap data uji dibandingkan model dengan 1 hidden layer. Sementara itu, pada Komposisi yang sama (70:30) dengan fungsi aktivasi tanh, terlihat perbedaan yang lebih jelas. Nilai MSE pada 1 hidden layer sebesar 0,2617, sedangkan pada 2 hidden layer turun menjadi 0,2412. Penurunan ini cukup signifikan dibandingkan ReLU, yang mengindikasikan bahwa pada fungsi aktivasi tanh, penambahan *hidden layer* memberikan kontribusi yang lebih besar terhadap peningkatan performa model. Hal ini menunjukkan bahwa arsitektur yang lebih dalam membantu tanh dalam mengekstraksi pola nonlinier yang lebih kompleks dari data.

Sementara itu, pada Komposisi data 80:20 dengan fungsi aktivasi ReLU, nilai MSE pada 1 hidden layer adalah 0,2300, dan justru meningkat menjadi 0,2344 pada 2 hidden layer. Hasil ini menunjukkan bahwa penambahan hidden layer pada konfigurasi ini tidak meningkatkan performa model, melainkan sedikit menurunkannya. Sementara itu, pada fungsi aktivasi tanh dengan Komposisi 80:20, nilai MSE pada 1 hidden layer adalah 0,2570, kemudian menurun menjadi 0,2394 pada 2 hidden layer. Pola ini kembali memperlihatkan bahwa penambahan hidden layer pada tanh mampu memperbaiki akurasi prediksi, walaupun masih belum lebih baik dibandingkan kombinasi ReLU.

Dengan demikian, dari sisi tingkat kesalahan prediksi (*MSE*), kombinasi terbaik adalah ANN dua *hidden layer* dengan fungsi aktivasi ReLU pada Komposisi data 70:30, karena memiliki nilai *MSE* paling kecil, yaitu 0,2260. Nama pelatihan yang terbaik ini adalah HL2-A9.

6.4 Perbandingan MSE ANN Mulai Dari Satu *Hidden Layer*



Gambar 6.4 Grafik MSE Mulai Dari Satu *Hidden Layer*

Gambar 6.4 merupakan grafik perbandingan nilai MSE mulai dari satu *hidden layer*. Grafik tersebut menunjukkan bahwa nilai MSE cenderung mengalami penurunan pada dua *hidden layer*, dan mengalami peningkatan pada tiga *hidden layer*. Penambahan *hidden layer* kedua pada arsitektur membuat model lebih mampu menangkap kompleksitas hubungan non-linear antara input (magnitudo, jarak, Vs30,) dan output (PGA). Sebaliknya, penambahan *hidden layer* ketiga justru menyebabkan penurunan akurasi. Hal ini kemungkinan disebabkan oleh terjadinya *overfitting*. Sehingga proses uji coba dengan penambahan jumlah *hidden layer* dihentikan hanya sampai pada *hidden layer* ketiga. Dengan demikian performa terbaik dari hasil uji coba penambahan *hidden layer* terjadi pada arsitektur dua *hidden layer*.

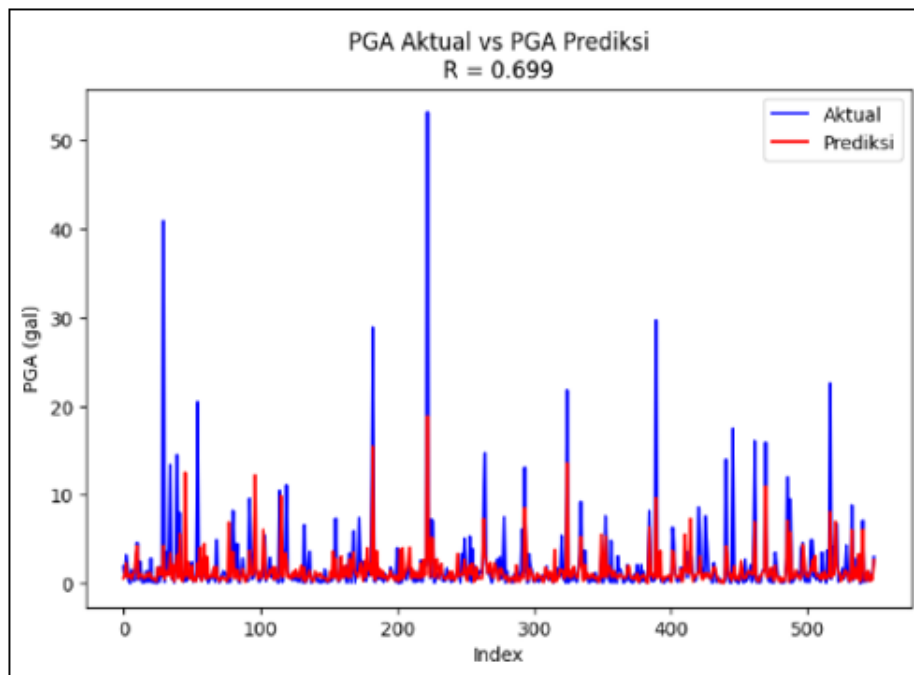
6.5 Analisis Model Terbaik Prediksi Peak Ground Acceleration (PGA)

Berdasarkan hasil perbandingan model ANN dengan beberapa *hidden layer* dan melakukan serangkaian pengujian terhadap berbagai kombinasi arsitektur *Artificial*

Neural Network (ANN), yang meliputi variasi jumlah node, fungsi aktivasi, serta Komposisi pembagian data latih dan data uji, diperoleh satu konfigurasi dengan performa paling optimal secara keseluruhan, yaitu model HL2-A9.

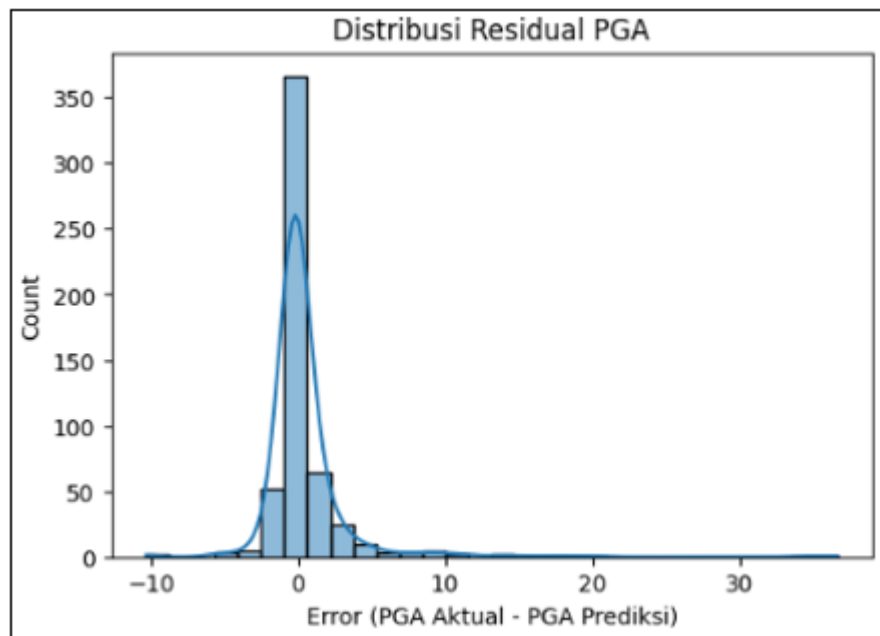
Model HL2-A9 merupakan model ANN dengan dua *hidden layer*, masing-masing terdiri atas 75 node, menggunakan fungsi aktivasi ReLU (*Rectified Linear Unit*), serta menerapkan Komposisi pembagian data 70% untuk pelatihan dan 30% untuk pengujian. Model ini menggunakan tiga parameter input utama, yaitu magnitudo gempa, jarak hiposenter, dan kecepatan gelombang geser rata-rata lapisan tanah hingga kedalaman 30 meter (V_{s30}), yang secara ilmu geofisika merupakan parameter dominan dalam mempengaruhi nilai *Peak Ground Acceleration* (PGA).

Dari sisi efisiensi iterasi, model HL2-A9 menunjukkan kinerja yang baik dengan memerlukan 106 iterasi untuk mencapai konvergensi. Nilai ini merupakan yang paling rendah dibandingkan seluruh konfigurasi lainnya, yang menandakan bahwa model ini mampu mempelajari pola hubungan antara parameter input dan output (PGA) secara lebih cepat dan efektif.. Dari sisi waktu konvergensi, model HL2-A9 juga menunjukkan performa terbaik dengan waktu pelatihan 1,27 detik. Waktu ini merupakan yang paling cepat dibandingkan model lainnya. Hal ini menunjukkan bahwa kombinasi dua *hidden layer* dan fungsi aktivasi ReLU mampu mempercepat proses pembaruan bobot serta meningkatkan efisiensi komputasi jaringan. Sementara itu, dari sisi akurasi prediksi, model HL2-A9 menghasilkan nilai MSE *testing* sebesar 0,2260, yang merupakan nilai MSE terkecil di antara seluruh skenario uji coba. Nilai MSE yang rendah mengindikasikan bahwa selisih antara nilai *ground truth* (data aktual) dengan nilai PGA prediksi relatif kecil.



Gambar 6.5. Perbandingan PGA aktul dan PGA prediksi model HL2-A9

Gambar 6.5 menunjukkan bahwa model HL2-A9 mampu menangkap pola *Peak Ground Accreration* (PGA) dengan cukup baik, khususnya untuk nilai PGA di bawah 15 gal. Dalam rentang ini, kurva PGA prediksi dan PGA aktual mengikuti tren yang hampir sama. Hal ini menunjukkan bahwa prediksi model cukup akurat. Namun, untuk nilai PGA di atas 15 gal, terdapat perbedaan yang signifikan antara PGA prediksi dan PGA aktual. Model cenderung kurang mampu memprediksi data dengan nilai PGA yang cukup tinggi. Nilai koefisien korelasi sebesar 0,699 mengindikasikan bahwa terdapat hubungan linear positif yang cukup kuat antara nilai *Peak Ground Acceleration* (PGA) aktual dan PGA hasil prediksi. Meskipun bukan korelasi yang sangat kuat, nilai ini masih menunjukkan bahwa model memiliki kemampuan memprediksi tren umum, khususnya untuk rentang nilai PGA yang lebih rendah.



Gambar 6.6. Distribusi Residual PGA aktul dan PGA prediksi model HL2-A9

. Gambar 6.6 merupakan distribusi residual PGA aktual dan PGA prediksi yang menunjukkan bahwa sebagian besar nilai residual berada di sekitar nol. Hal ini mengindikasikan bahwa model memiliki tingkat kesalahan yang relatif kecil dan tidak menunjukkan adanya bias yang signifikan terhadap *overestimate* maupun *underestimate* secara umum. Selain itu terdapat residual kecil ke arah positif dengan nilai PGA sekitar 38 gal dan ke arah negatif sekitar -10 gal. Hal ini mengindikasikan bahwa model masih memiliki keterbatasan dalam memprediksi nilai PGA yang cukup tinggi. Namun, karena jumlah residual besar tersebut relatif sedikit, secara keseluruhan model HL2-A9 tetap menunjukkan performa yang baik dan stabil dalam memprediksi PGA.

Penelitian mengenai prediksi PGA perlu dilakukan sebagai upaya mitigasi resiko gempa bumi. Fenomena gempa dan upaya mitigasi tersirat dalam Al Quran Surah Al-Ankabut (29:37) ,

فَكَذَّبُوهُ فَأَخَذَتْهُمُ الرَّجْفَةُ فَأَصْبَحُوا فِي دَارِهِمْ جُثَمِينَ

Artinya : Mereka mendustakannya (Syuaib), maka mereka ditimpa gempa yang dahsyat, lalu jadilah mereka mayat-mayat yang bergelimpangan di tempat-tempat tinggal mereka.

Menurut *tafsir Ibnu Katsir*, *Fathul Karim Mukhtashar Tafsir al-Qur'an al-'Adzhim*, karya Syaikh Prof. Dr. Hikmat bin Basyir bin Yasin, QS Al Ankabut ayat 37 menyatakan bahwa Allah melarang mereka berbuat onar dan kerusakan di bumi, yaitu berbuat sewenang-wenang terhadap penghuninya. Demikian itu karena mereka biasa mengurangi takaran dan timbangan serta menyamun. Bersamaan dengan semua tindakan itu, mereka juga kafir kepada Allah dan rasulNya, maka Allah membinasakan mereka dengan gempa dahsyat yang menghancurkan negeri mereka, dan pekikan mengguntur yang dapat mengeluarkan jantung mereka. Azab itu terjadi pada hari yang dinaungi awan yang gelap; sehingga arwah mereka meregang dari tempatnya, sesungguhnya itu merupakan azab yang besar

Prediksi dalam sains adalah mengamati pola dari data yang ada. Al-Qur'an mendorong manusia untuk belajar dari alam, menggunakan akal, melakukan perhitungan, dan menyiapkan diri. Semua ini selaras dengan metodologi penelitian: mengumpulkan input (magnitudo, Vs30, jarak hiposenter), mempelajari pola melalui ANN, dan memprediksi output (PGA). Hal ini tersirat dalam Al Quran surah Ali imran ayat 191.

الَّذِينَ يَذْكُرُونَ اللَّهَ قِيَامًا وَقُعُودًا وَعَلَىٰ جُنُوبِهِمْ وَيَتَفَكَّرُونَ فِي خَلْقِ
السَّمُوتِ وَالْأَرْضِ رَبَّنَا مَا خَلَقْتَ هَذَا بَاطِلًا سُبْحَانَكَ فَقِنَا عَذَابَ النَّارِ

Artinya :

(yaitu) orang-orang yang mengingat Allah sambil berdiri, duduk, atau dalam keadaan berbaring, dan memikirkan tentang penciptaan langit dan bumi (seraya berkata), “Ya

Tuhan kami, tidaklah Engkau menciptakan semua ini sia-sia. Mahasuci Engkau. Lindungilah kami dari azab neraka.

Menurut *tafsir Al-Mukhtashar* Syaikh Dr. Shalih bin Abdullah bin Humaid, QS. Ali Imran ayat 191 menyatakan bahwa Mereka adalah orang-orang yang senantiasa mengingat Allah dalam kondisi apapun. Baik dalam kondisi berdiri, duduk maupun berbaring. Dan mereka juga senantiasa menggunakan akal pikiran mereka untuk memikirkan penciptaan langit dan bumi. Mereka pun berkata, “Wahai Rabb, Engkau tidak menciptakan makhluk yang sangat besar ini untuk bersenda gurau. Mahasuci Engkau dari senda gurau. Maka jauhkanlah kami dari azab Neraka, dengan cara Engkau bimbing kami kepada perbuatan-perbuatan yang baik dan Engkau lindungi kami dari perbuatan-perbuatan yang buruk. Meskipun ayat tersebut tidak langsung menjelaskan tentang prediksi PGA, tetapi ayat tersebut memerintahkan kita untuk berfikir yang selaras dengan proses penelitian terhadap fenomena-fenomena yang terjadi di bumi ini. Seperti ketika terjadi gempa banyak hal yang bisa dipelajari, seperti dampak guncangan gempa yang bisa diasosiasikan dengan PGA. Kita juga bisa berfikir apa saja yang mempengaruhi PGA tersebut. Sehingga akhirnya bisa dilakukan penelitian mengenai prediksi PGA dimasa yang akan datang.

BAB 7

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil pelatihan dan pengujian metode ANN maka diperoleh kesimpulan bahwa arsitektur ANN dua *hidden layer* memiliki performa yang terbaik dalam memprediksi nilai PGA di pulau Jawa. Hal ini ditunjukkan oleh model dengan nama pelatihan HL2-A9 yang memiliki arsitektur dua *hidden layer* dengan jumlah node (75-75), fungsi aktivasi ReLU, dan Komposisi data 70:30. Model HL2-A9 memiliki nilai MSE 0,2260 dan nilai koefisien korelasi antara PGA aktual dan PGA prediksi sebesar 0,699 yang menunjukkan adanya hubungan linear positif yang cukup kuat.

Variasi jumlah node pada *hidden layer* memberikan pengaruh signifikan terhadap performa model. Namun penambahan jumlah *node* tidak selalu menghasilkan performa yang lebih baik. Fungsi aktivasi *ReLU* secara konsisten menunjukkan performa yang lebih baik dibandingkan fungsi tanh, baik itu dari nilai MSE *testing*, jumlah iterasi, maupun waktu konvergensi. Begitu juga dengan penambahan *hidden layer* tidak selalu meningkatkan performa model. Hal itu disebabkan karena terlalu banyak penggunaan *hidden layer* juga bisa menyebabkan terjadinya *overfitting*. Dengan demikian, pemilihan arsitektur yang optimal memerlukan keseimbangan antara kompleksitas jaringan dan kemampuan generalisasi model.

5.2 Saran

Secara keseluruhan, model HL2-A9 yang memiliki konfigurasi dua *hidden layer* dengan jumlah node (75-75), fungsi aktivasi ReLU, dan komposisi data 70:30 efektif dalam memprediksi PGA untuk rentang nilai rendah hingga sedang dengan tingkat kesalahan yang rendah. Namun, perlu adanya pengembangan lebih lanjut agar prediksi nilai PGA tinggi menjadi lebih akurat, misalnya melalui penambahan data yang lebih besar dan lebih beragam, baik dari segi lokasi maupun karakteristik gempa, mencoba arsitektur jaringan yang lebih kompleks, atau melakukan optimasi fungsi loss.

DAFTAR PUSTAKA

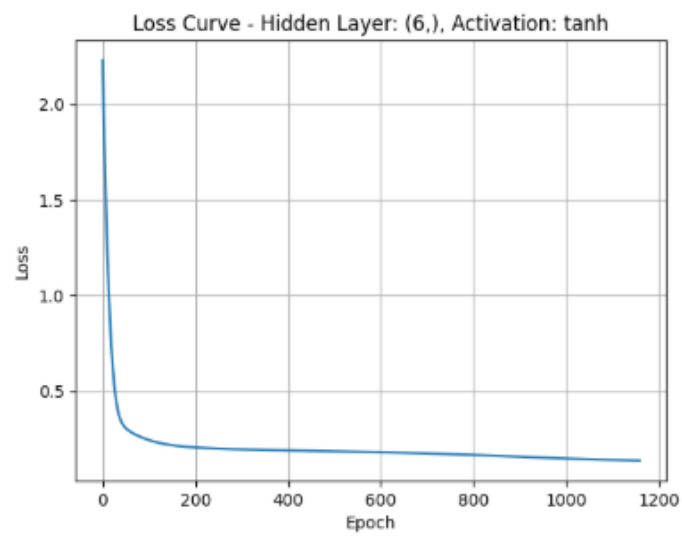
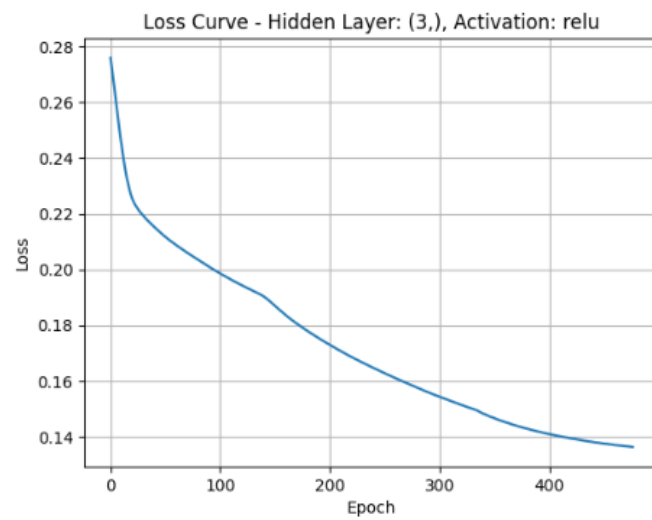
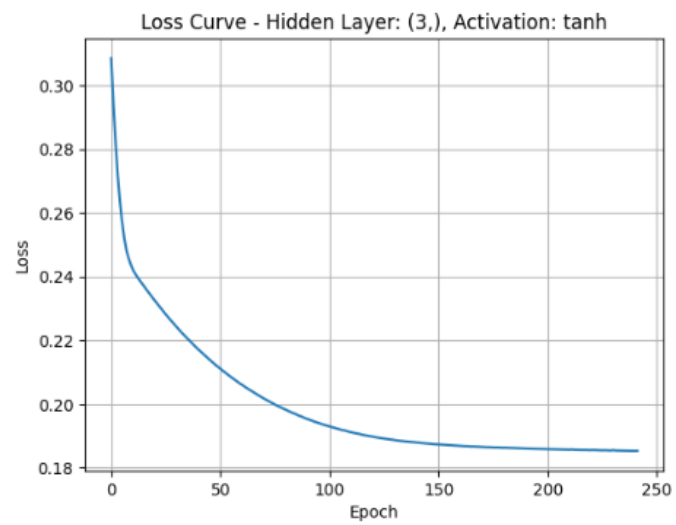
- Alimoradi, A., dan Beck, J.L., “Machine-learning methods for earthquake ground motion analysis and simulation”. *Journal of Engineering Mechanics*, Vol. 141, 2015. [https://doi.org/10.1061/\(asce\)em.1943-7889.0000869](https://doi.org/10.1061/(asce)em.1943-7889.0000869).
- Asenahabi, B. M., “Basics of research design: A guide to selecting appropriate research design. *International Journal of Contemporary Applied Researches*”, 6(5), 76–89. 2019.
- Bock, Y., Prawirodirdjo, L., Genrich, J. F., Stevens, C. W., McCaffrey, R., Subarya, C., Puntodewo, S. S. O., dan Calais, E., “Crustal motion in Indonesia from Global Positioning System measurements”, *Journal of Geophysical Research: Solid Earth*, Vol. 108(B8), 2003. <https://doi.org/10.1029/2001JB000324>.
- Dhanya, J., dan Raghukanth, S.T.G., “Ground Motion Prediction Model Using Artificial Neural Network”, *Springer Nature, Pure and Applied Geophysics*, 2017. <https://doi.org/10.1007/s00024-017-1751-3>.
- Fausett, L., “*Fundamentals of Neural Networks: Architectures, Applications and Algorithms*”. Prentice-Hall International, Inc. 1994.
- Hong, H., Liu, T., dan Lee, C.S., “Observations on the application of artificial neural network to predicting ground motion measures”, *Earthquake Science*, Vol. 25 No. 161-175, 2012. <https://doi.org/10.1007/s11589-012-0843-5>.
- Junjie, W., Maruyama, Y., dan Liu, W., “Construction of Attenuation Relationship of Peak Ground Velocity Using Machine Learning and Examination of Station Correction Factor”, 2021. www.researchgate.net/publication/355841966.
- Kang, S., Mun, E., Phuong, D.T.T., dan Kim, B., “Machine learning-based ground motion models for predicting PSAs of borehole motions in Japan”, *Journal of Seismology*, 2024. <https://doi.org/10.1007/s10950-024-10203-w>.
- Khosravikia, F., Zeinali, Y., Nagy, Z., Clayton, P., dan Rathje, E.M., “Neural Network-Based Equations for Predicting PGA and PGV in Texas, Oklahoma, and Kansas”, *Geotechnical Earthquake Engineering and Soil Dynamics*, Vol. GSP 291, 2018. <https://www.researchgate.net/publication/325557379>.
- Khosravikia, F., dan Clayton, P., “Machine Learning in Ground Motion Prediction”, *Computer and Geosciences*, Vol. 148, 2021.
- Kong, Q., Trugman, D.T., Ross, Z.E., Bianco, M.J., Meade, B.J., dan Gerstoft, P., “Machine learning in seismology: turning data into insights”, *Seismological Research Letters*, Vol. 90:3–14, 2019. <https://doi.org/10.1785/0220180259>.
- Kreh T, Chaw D., “Neural network approach and micro tremor measurements in estimating peak ground acceleration due to strong motion”. *Engineering Software* 33: 733-742, 2002.

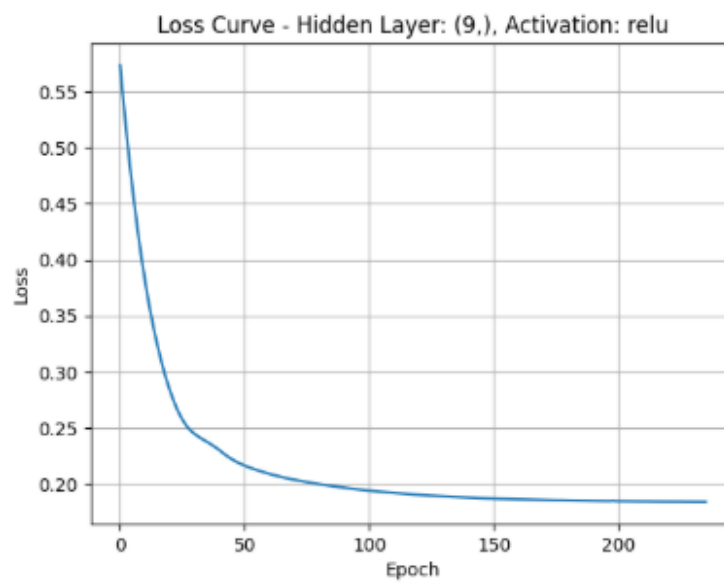
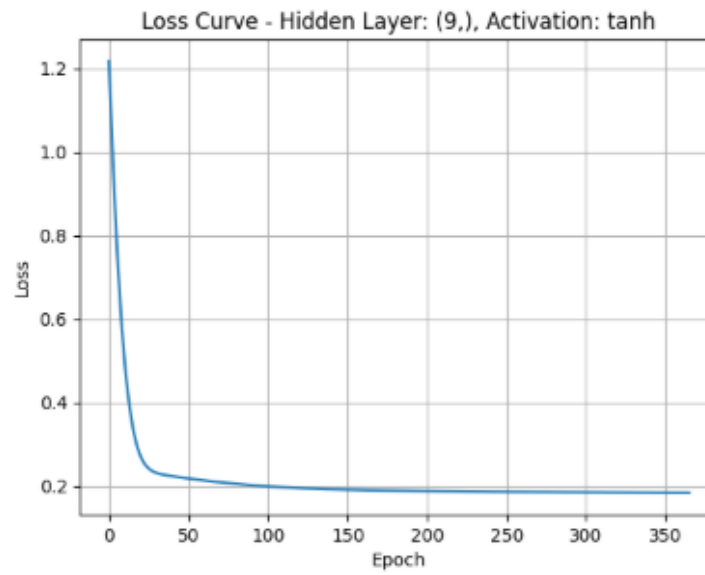
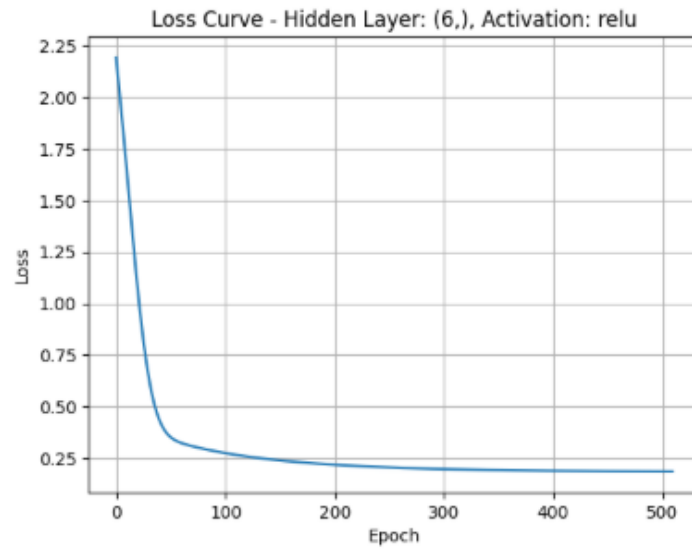
- Luzi, L., Felicetta, C., dan Lanzano, G., "Functional form or Machine-Learning-Based Ground-Motion Models? An application to the Italian dataset", *Research Square*, 2023. <https://doi.org/10.21203/rs.3.rs-3025369/v1>.
- Lim, W. T., Wang, L., Wang, Y., & Chang, Q., "Housing price prediction using neural networks". 22th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD), 518–522.2016.
- Mandal Prantik, dan Mandal, Priyank, "Peak Ground Acceleration Prediction using supervised Machine Learning algorithm for earthquakes of Mw5.6-7.9 occurring in India and Nepal", *Research Square*, 2022. <https://doi.org/10.21203/rs.3.rs-1806354/v1>.
- Mohammadi, A., Karimzadeh, S., dan Banimahd, S.A., "The potential of region-specific machine-learning-based ground motion models: Application to Turkey", *Soil Dynamics and Earthquake Engineering*, Vol. 172, 2023.
- Prezioso, E., Sharma, N., Piccialli, F., dan Convertito, V., "A data-driven artificial neural network model for the prediction of ground motion from induced seismicity: The case of The Geysers geothermal field", *Frontiers in Earth Science*, 2022. <https://doi.org/10.3389/feart.2022.917608>.
- Safari, A., & Ghavifekr, A. A., "International stock index prediction using artificial neural network (ANN) and Python programming". 2021 7th 96 International Conference on Control, Instrumentation and Automation (ICCIA), 1–7. 2021.
- Smyth, H. R., Hall, R., dan Nichols, G. J., "Cenozoic volcanic arc history of East Java, Indonesia: the stratigraphic record of eruptions on an active continental margin", *Geological Society of America Special Papers*, Vol. 436, 199–222, 2008. [https://doi.org/10.1130/2008.2436\(10\)](https://doi.org/10.1130/2008.2436(10)).
- Soehaimi, " Seismotektonik dan Potensi Kegempaan di Wilayah Jawa", *Jurnal Geologi Indonesia*, Vol. 3 No. 4, 2008.
- Sedaghati, F., dan Pezeshk, S., "Machine learning-based ground motion models for shallow crustal earthquakes in active tectonic regions", *Earthquake Spectra*, 2023. <https://doi.org/10.1177/87552930231191759>.
- Sun, R., Qi, W., Zheng, T., dan Qi, J., "Explainable Machine-Learning Predictions for Peak Ground Acceleration", *Applied Sciences*, Vol. 13, 2023. <https://doi.org/10.3390/app13074530>.
- Thomas, A. J., Petridis, M., Walters, S. D., Gheytaei, S. M., & Morgan, R. E., "Two hidden layers are usually better than one". Engineering Applications of Neural Networks: 18th International Conference, EANN 2017, Athens, Greece, August 25–27, 2017, Proceedings, 279–290.
- Ulusay R, Tuncay E, Sonmez H., "An attenuation relationship based on Turkish strong motion data an iso-acceleration map of Turkey". *EngGeol* 74: 265-291, 2004.

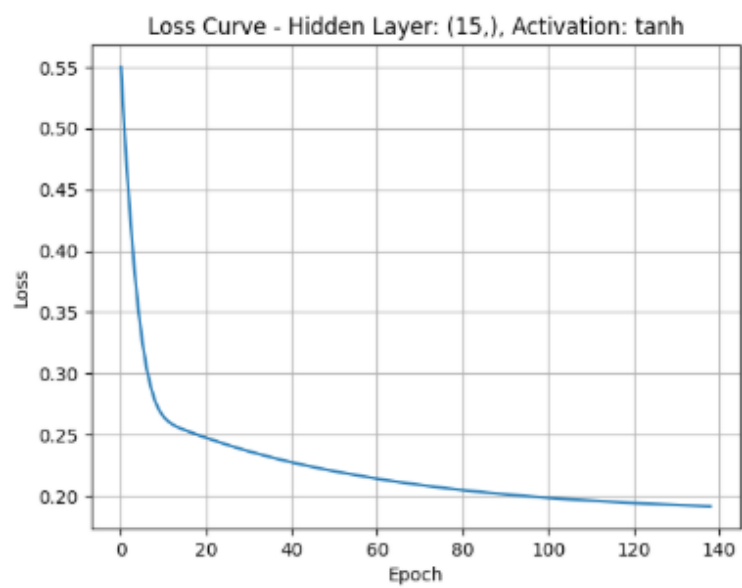
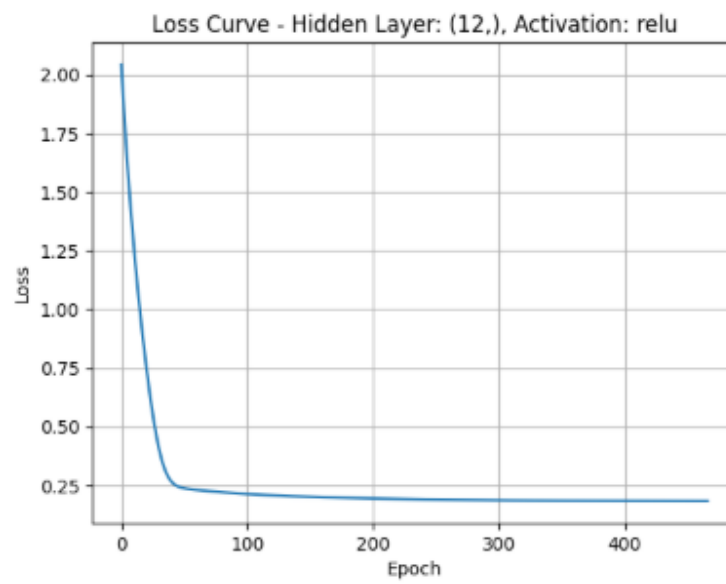
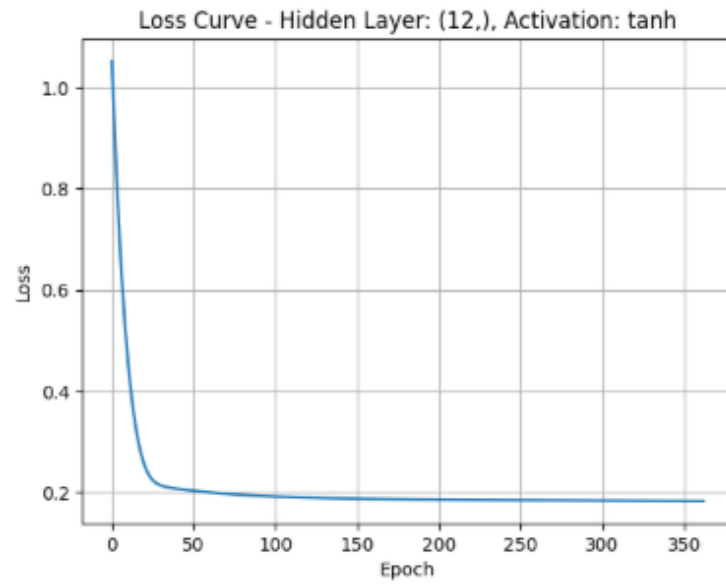
- Xiao, D., Li, B., & Mao, Y., "A multiple hidden layers extreme learning machine method and its application". *Mathematical Problems in Engineering*, 2017.
- Zadeh, M.A., Doloiee, G., dan Nasrollahnejad, A., "Prediction of Strong Ground Motion Using Fuzzy Inference Systems Based on Adaptive Networks", *Biostatistics and Biometrics Journal*, Vol. 6, 2018.
<https://doi.org/10.19080/BBOAJ.2018.06.555680>.
- Zhao, J.X., Takahashi, T., Asano, A., Okada, H., Saiki, T., Irikura, K., Zhang, J., Thio, Hong. K., Somerville, P.G., dan Fukushima, Y., "Attenuation Relations of Strong Motion in Japan Using Site Classification Based on Predominant Period", *Buletin o f Seismological Society ofAmerica*, Vol. 96, No. 3, 2006.

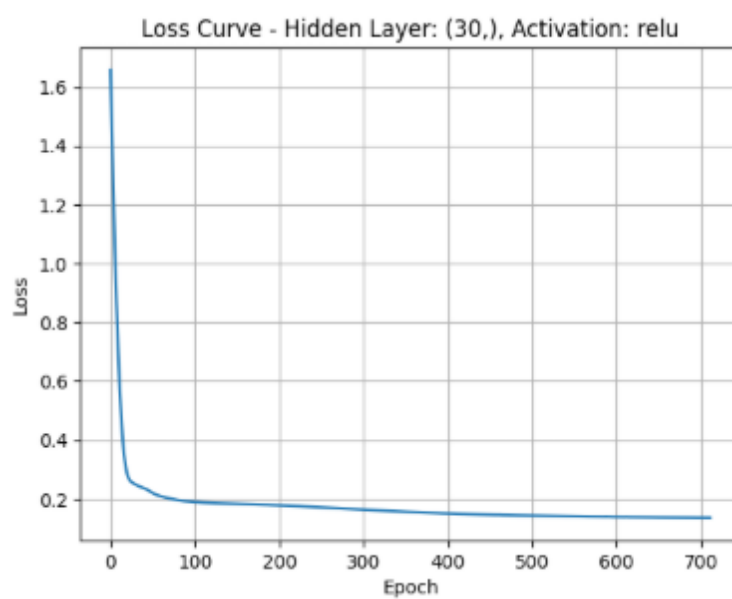
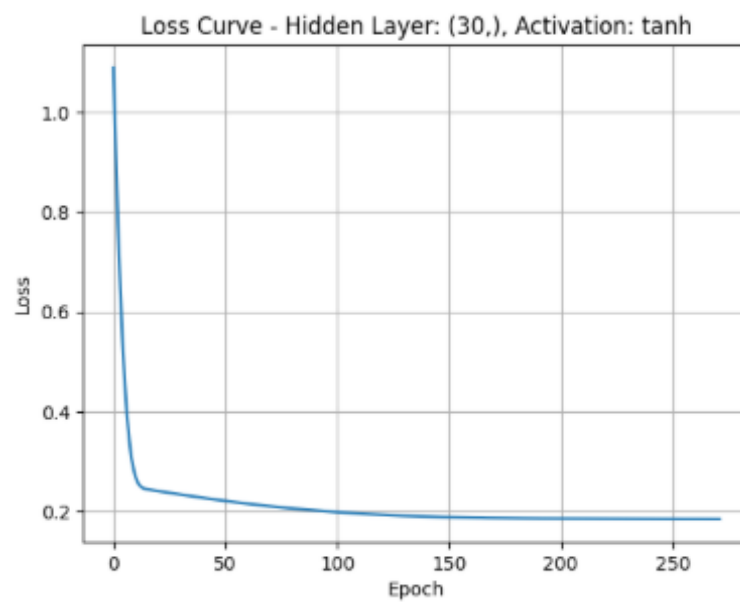
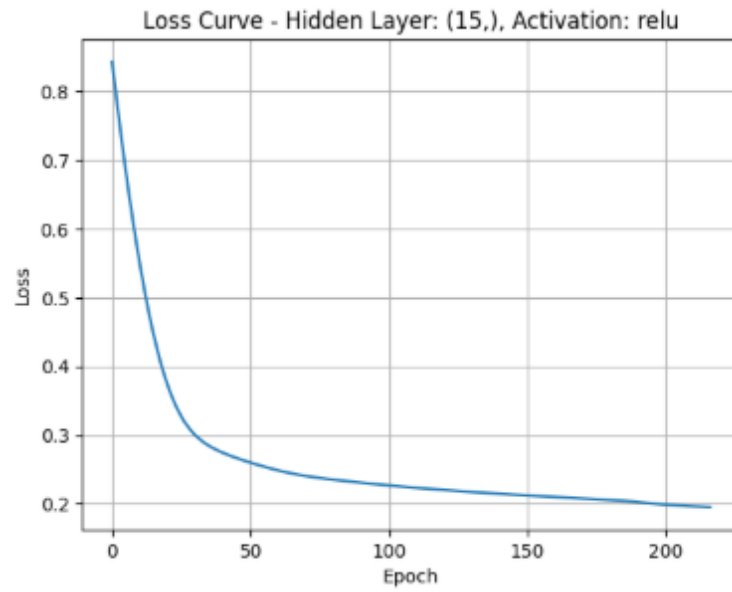
LAMPIRAN

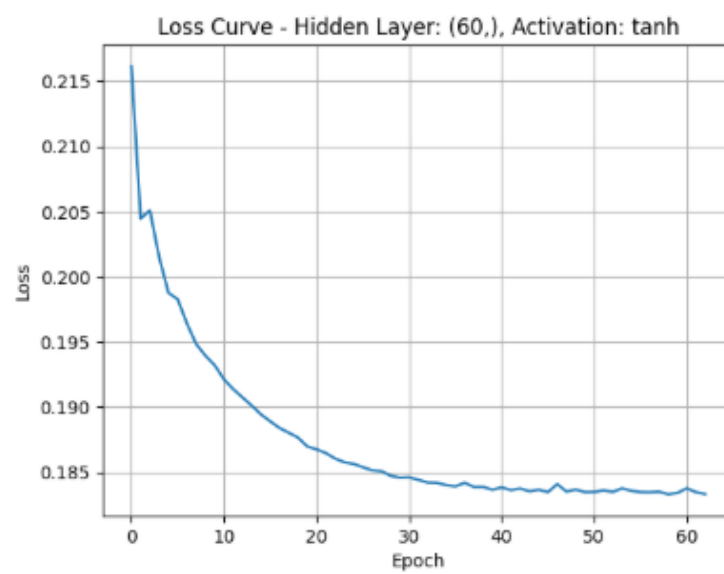
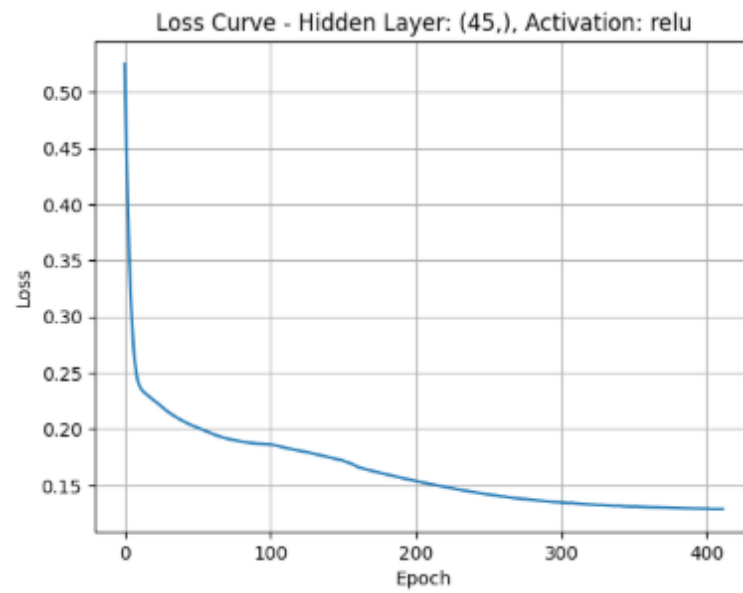
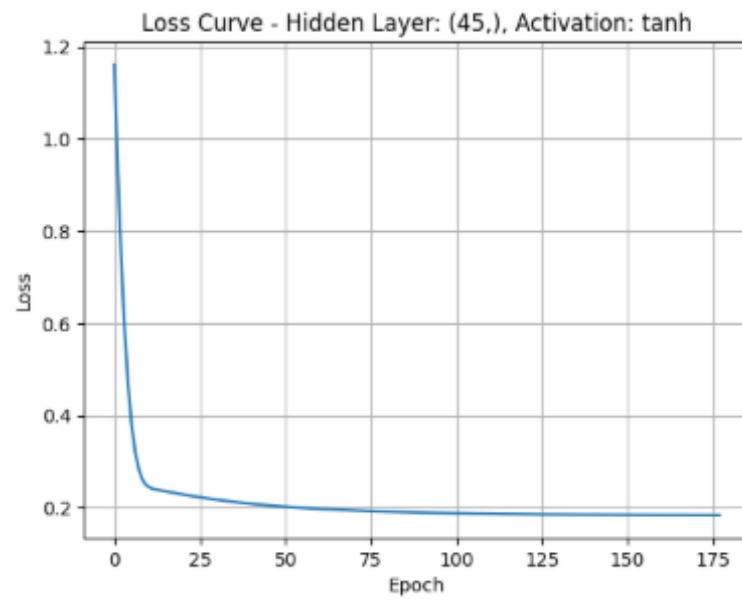
1. Grafik *Loss Function* Proses Training Komposisi Data 70:30

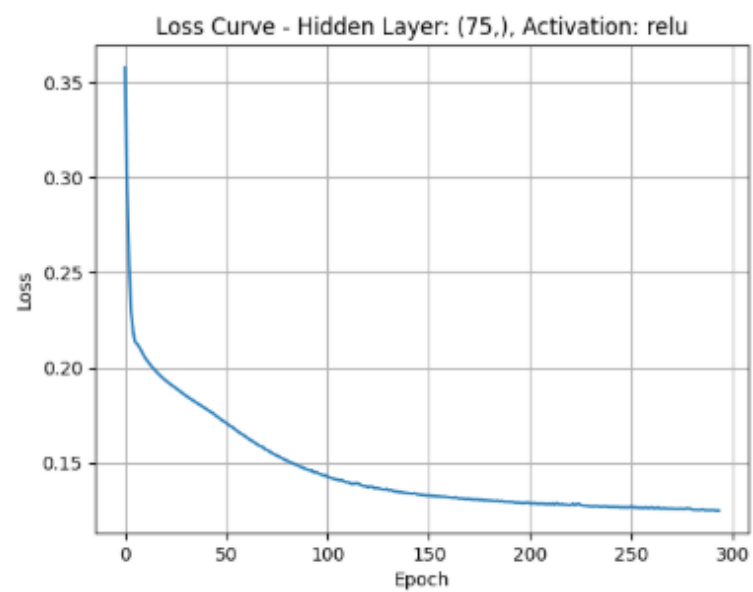
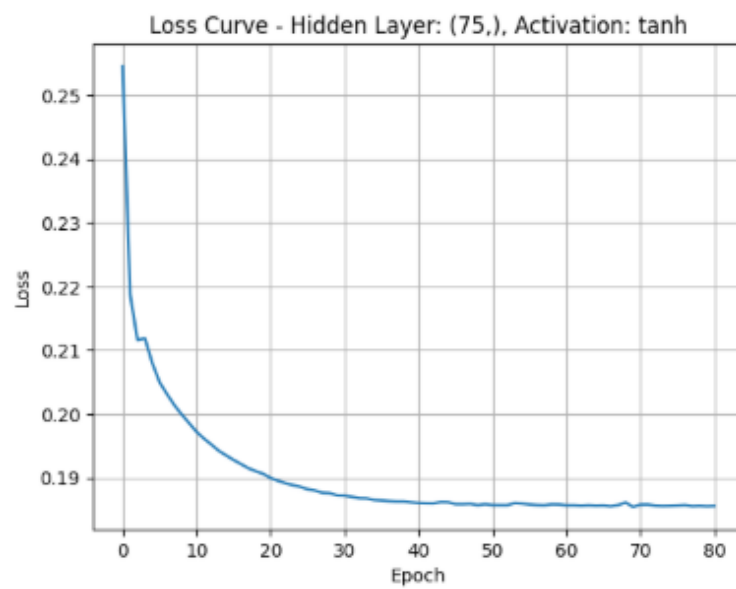
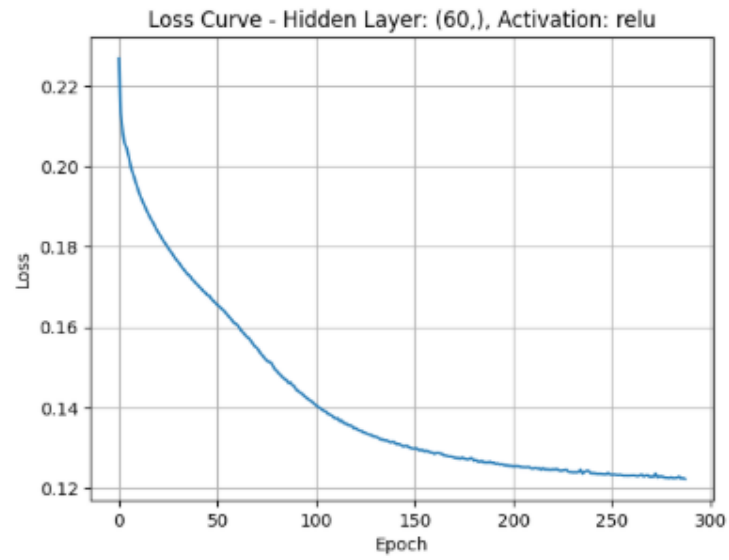


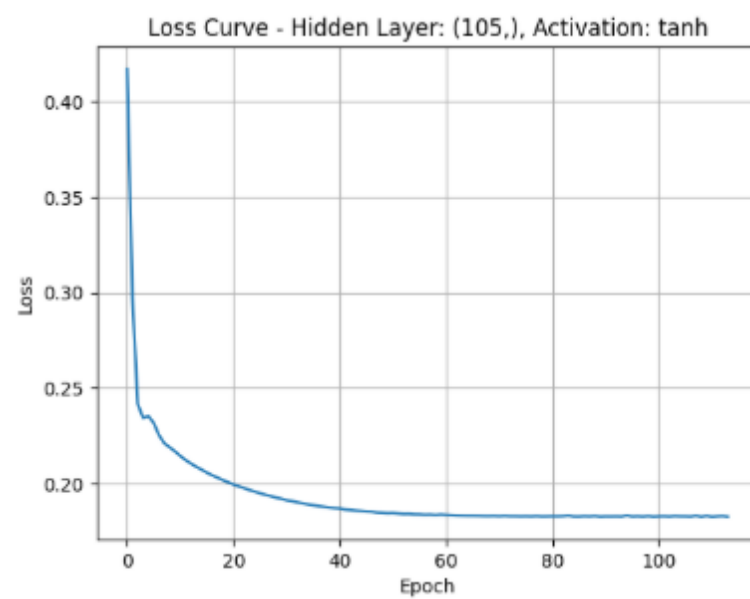
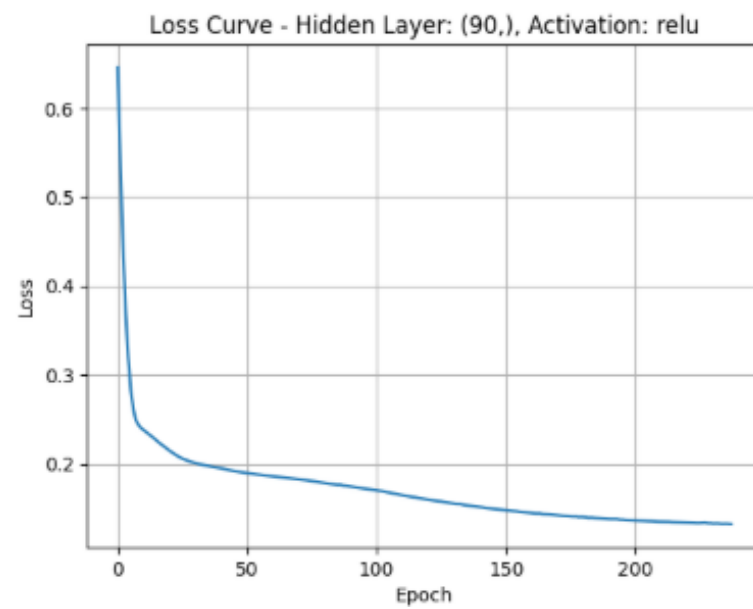
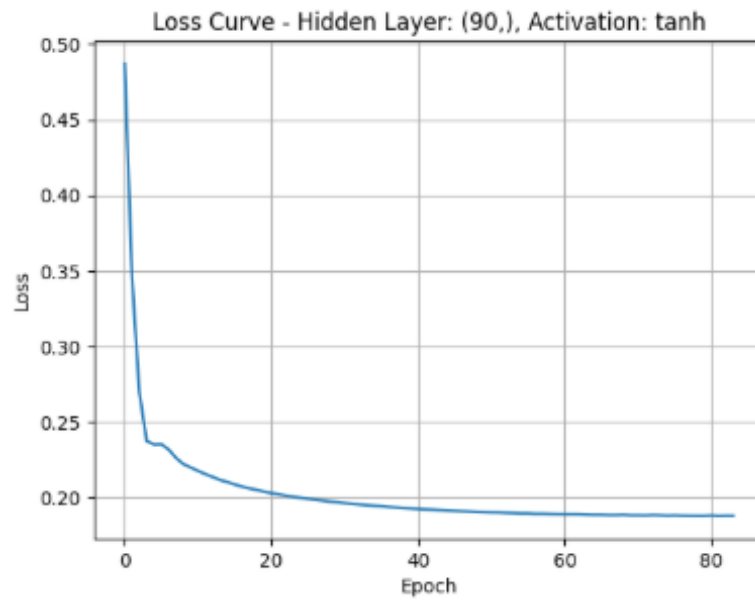


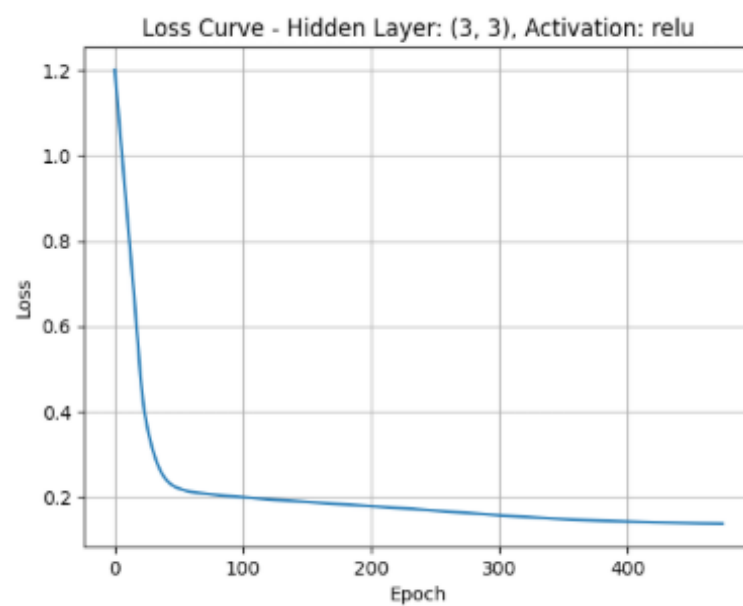
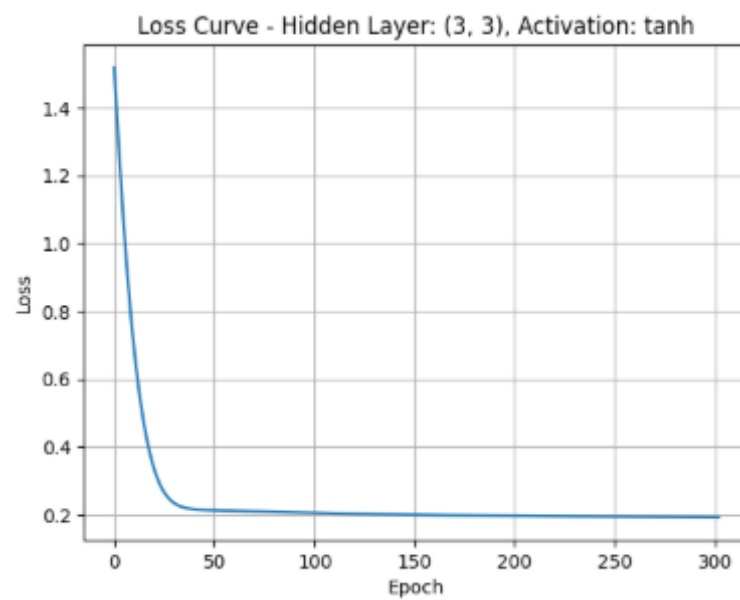
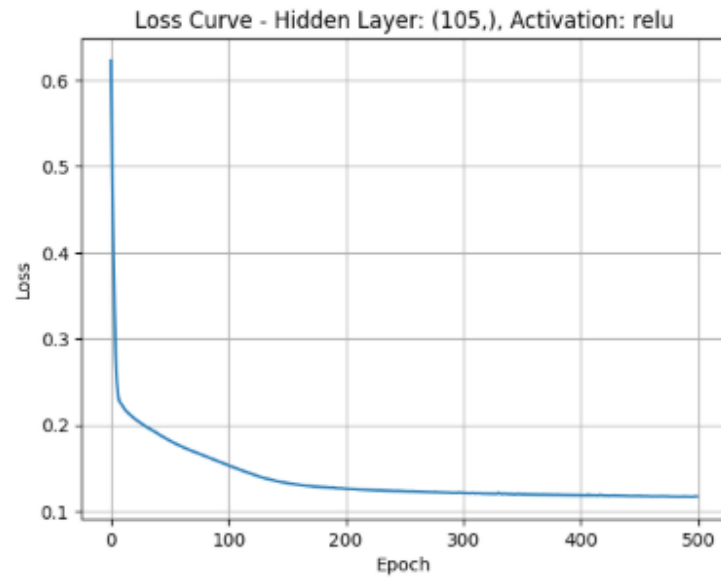


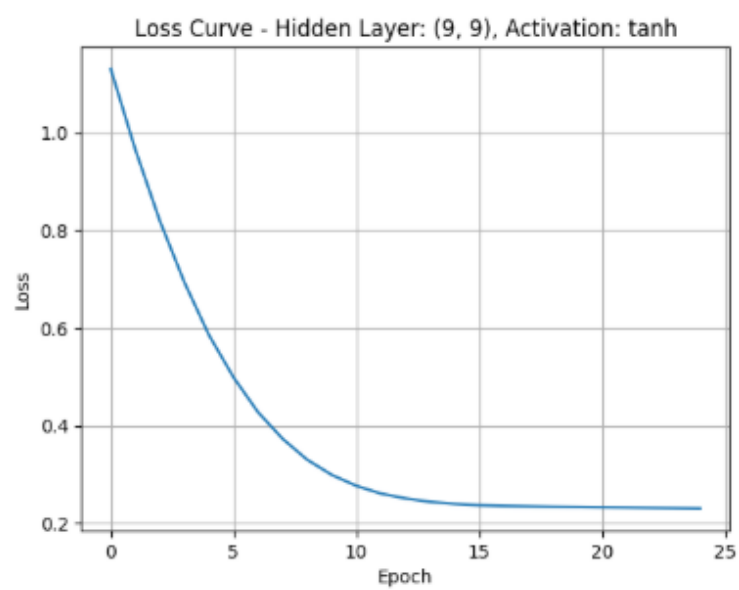
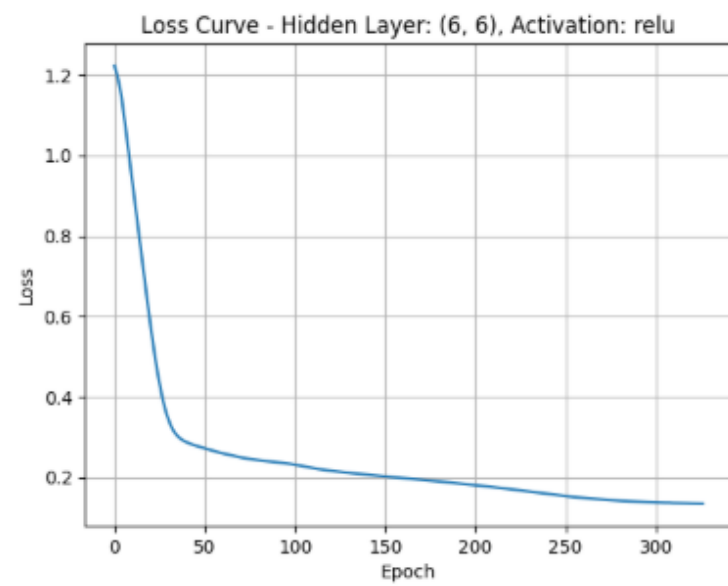
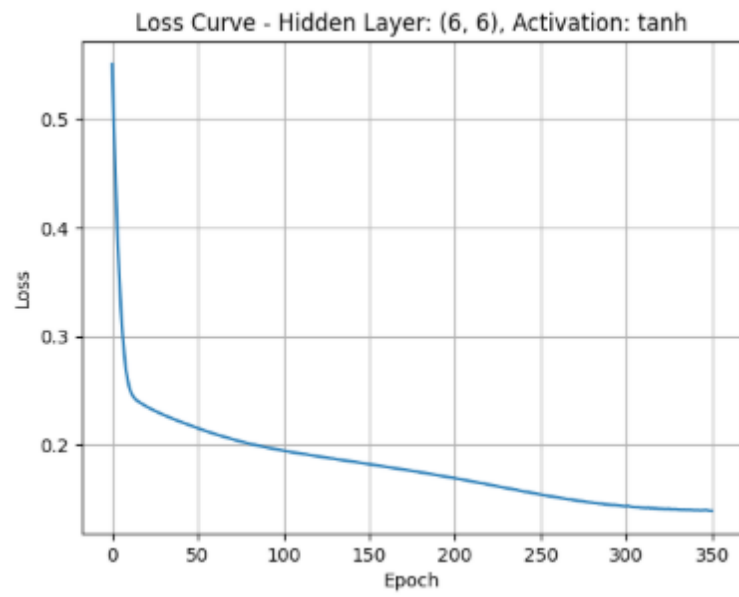


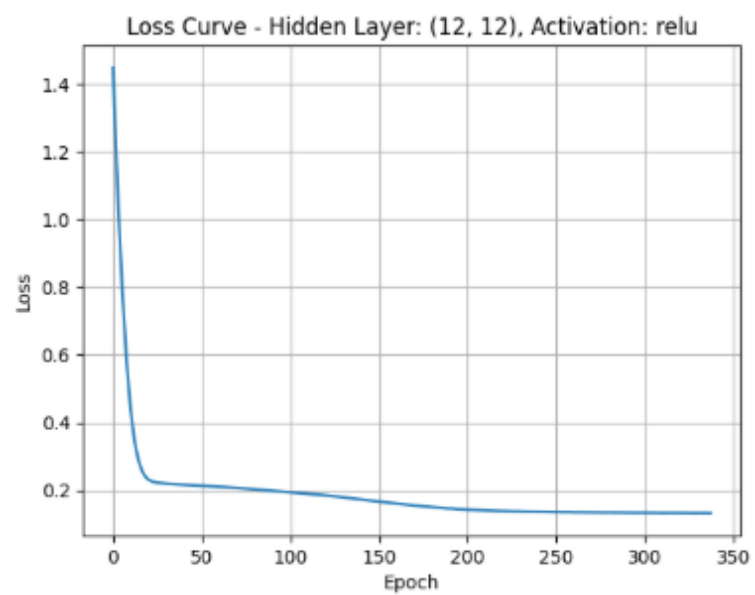
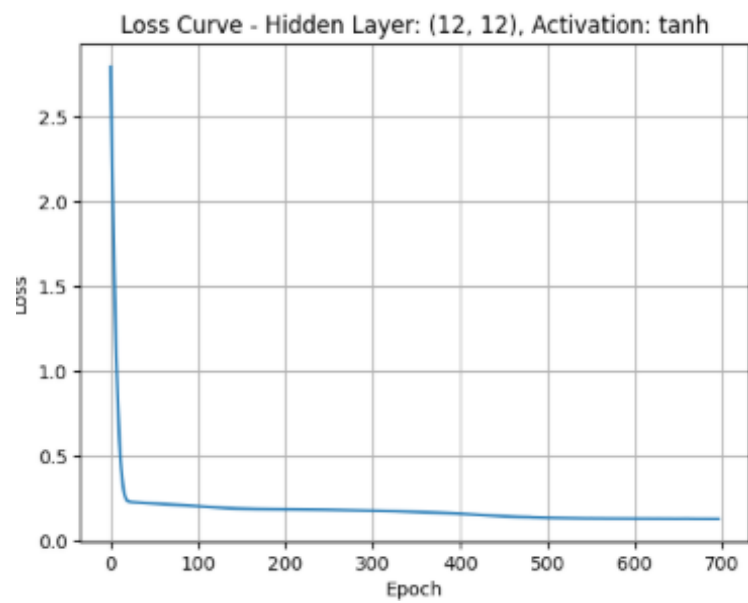
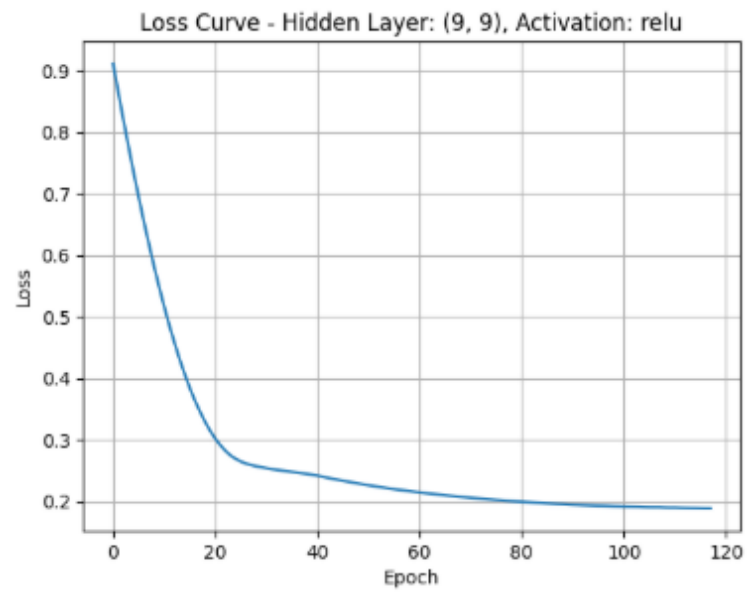


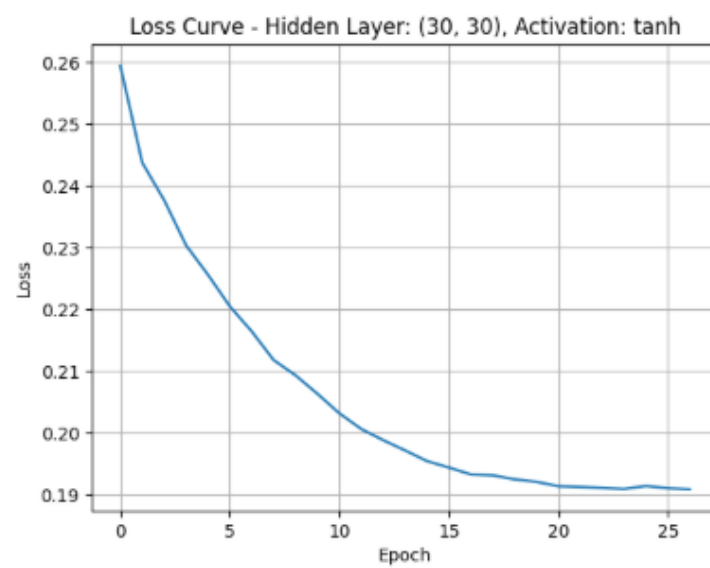
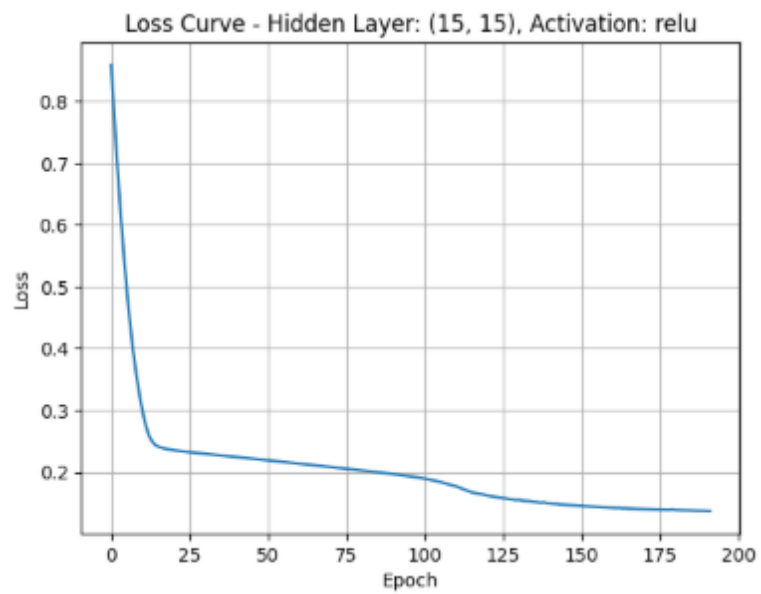
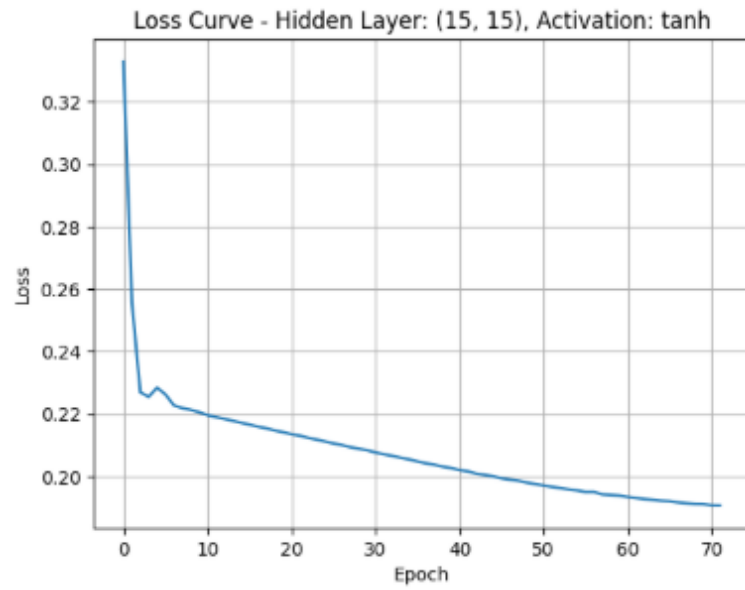


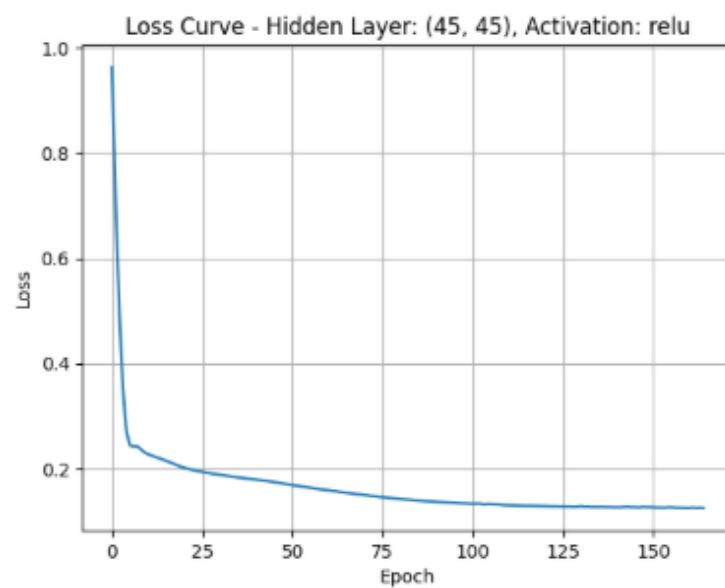
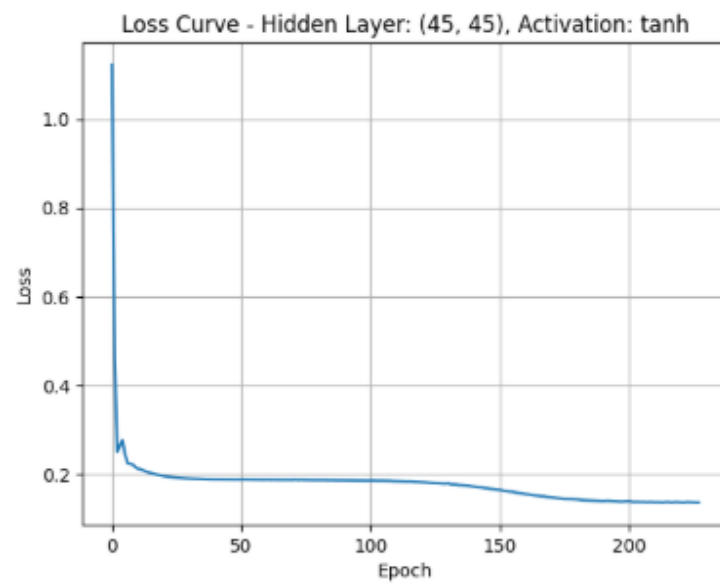
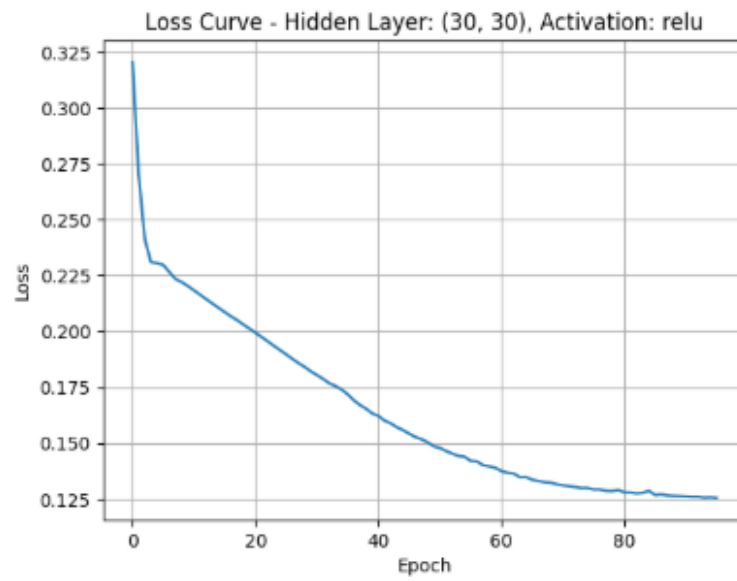


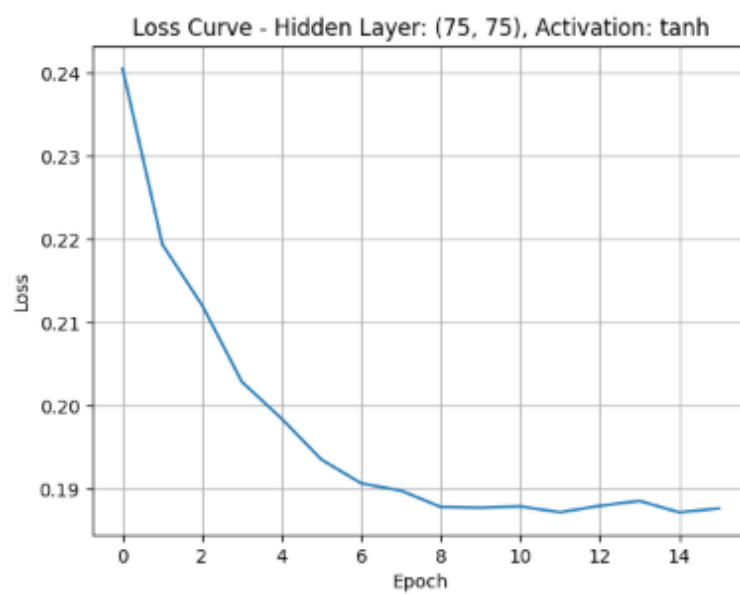
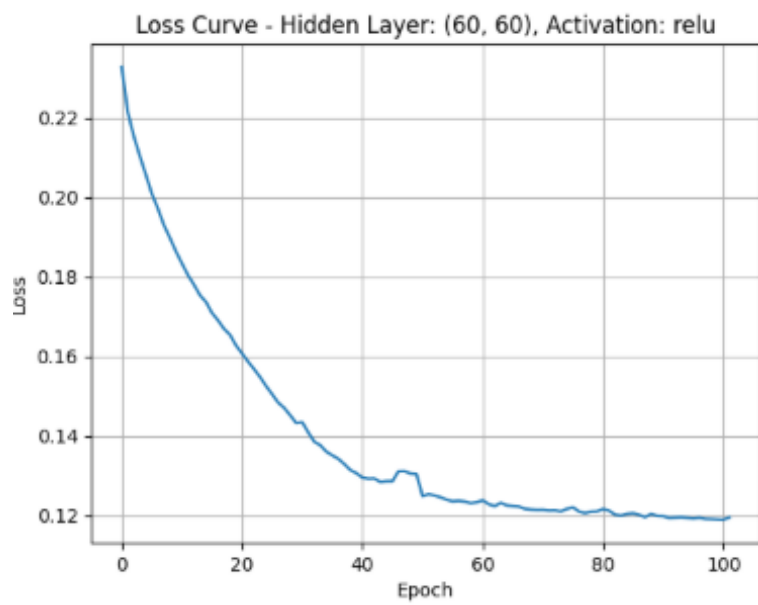
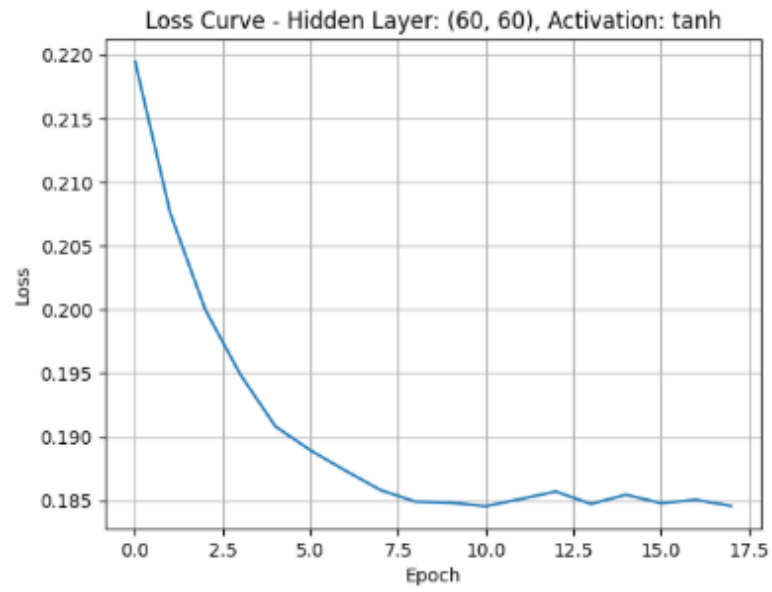


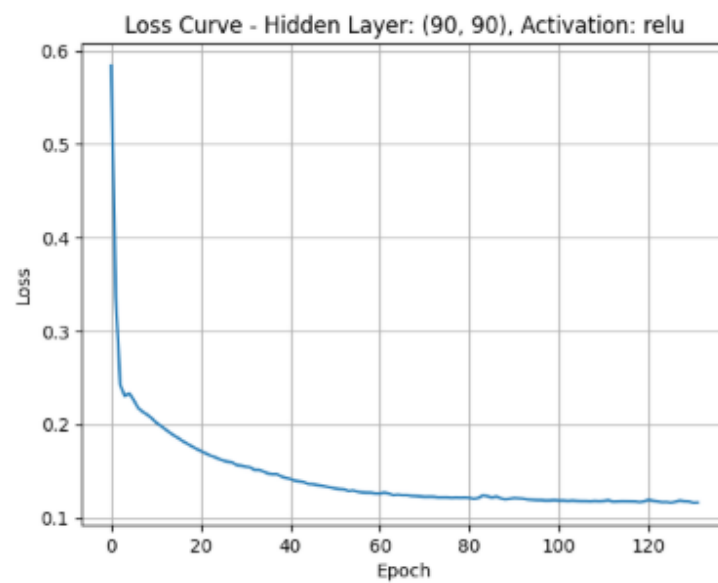
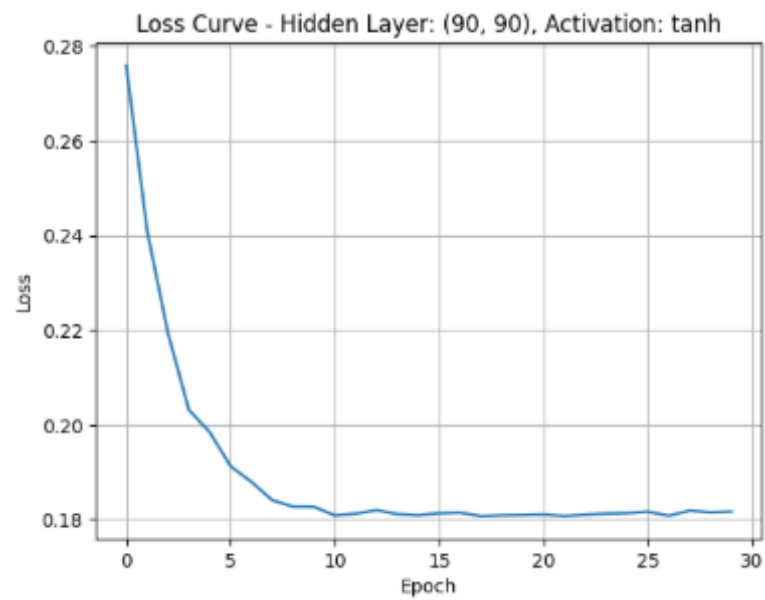
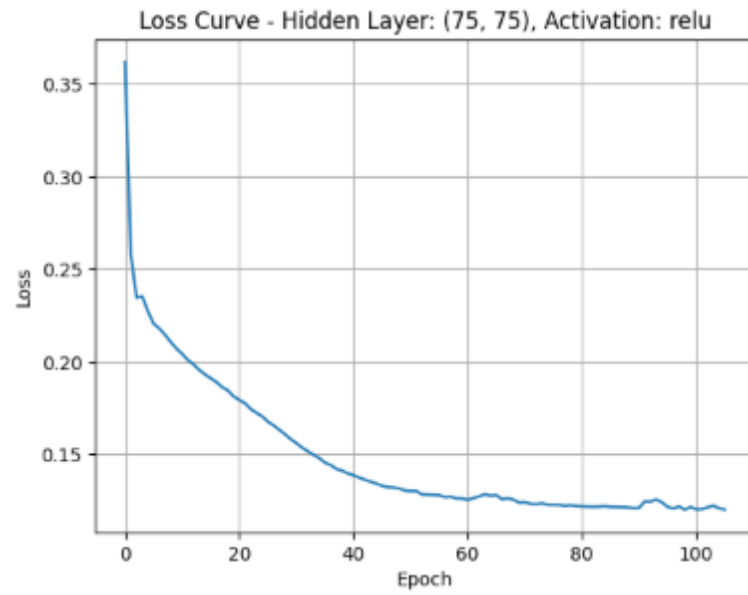


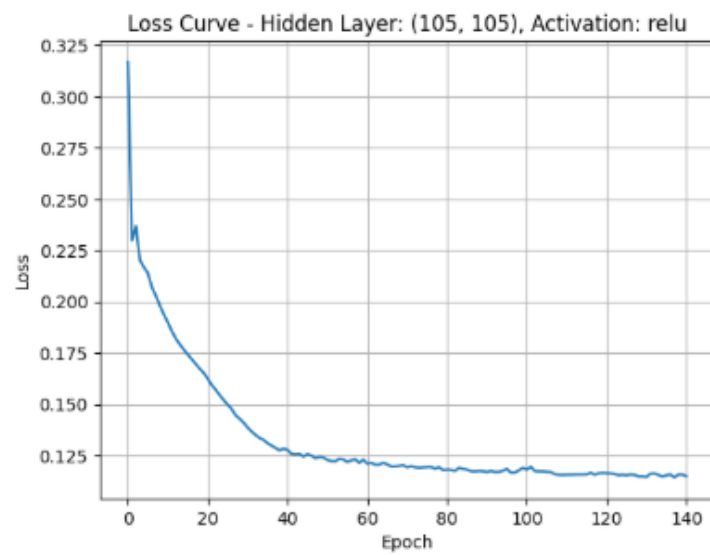
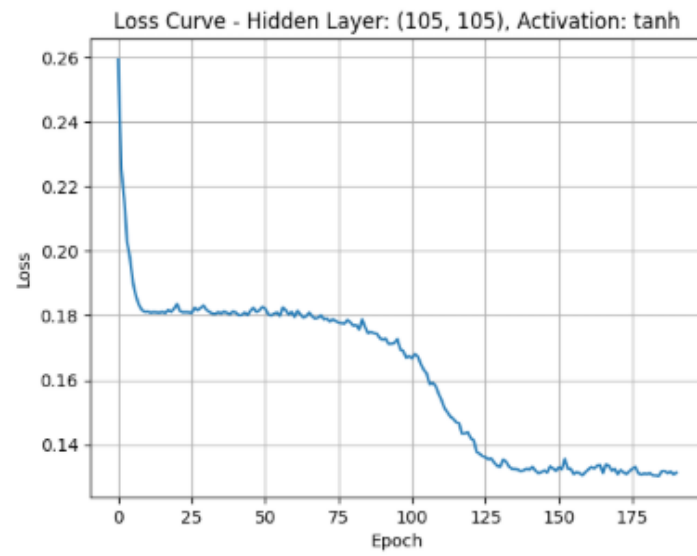












2. Grafik *Loss Function* Proses Training Komposisi Data 80:20

