

**IMPLEMENTASI ALGORITMA *ARTIFICIAL NEURAL NETWORK*
BACKPROPAGATION UNTUK MELAKUKAN PERAMALAN
CURAH HUJAN DI KABUPATEN MALANG**

SKRIPSI

**Oleh:
DONNY KURNIAWAN
NIM. 17650086**



**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2021**

**IMPLEMENTASI ALGORITMA *ARTIFICIAL NEURAL NETWORK*
BACKPROPAGATION UNTUK MELAKUKAN PERAMALAN
CURAH HUJAN DI KABUPATEN MALANG**

SKRIPSI

**Diajukan Kepada:
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang
Untuk Memenuhi Salah Satu Persyaratan Dalam
Memperoleh Gelar Sarjana Komputer (S.Kom)**

**Oleh:
DONNY KURNIAWAN
NIM. 17650086**

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2021**

HALAMAN PERSETUJUAN

**IMPLEMENTASI ALGORITMA *ARTIFICIAL NEURAL NETWORK*
BACKPROPAGATION UNTUK MELAKUKAN PERAMALAN
CURAH HUJAN DI KABUPATEN MALANG**

SKRIPSI

Oleh:
DONNY KURNIAWAN
NIM. 17650086

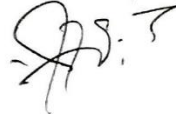
Telah Diperiksa dan Disetujui untuk Diuji:
Tanggal: 14 Desember 2021

Pembimbing I,



Fatchurrohman, M.Kom
NIP. 19700731 200501 1 002

Pembimbing II,



Khadijah Fahmi Hayati Holle, M.Kom
NIP. 19900626 20160801 2 077

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



Dr. Fatchro Kurniawan, M.MT
NIP. 1971020 200912 1 001

HALAMAN PENGESAHAN



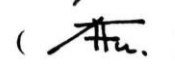

**IMPLEMENTASI ALGORITMA *ARTIFICIAL NEURAL NETWORK*
BACKPROPAGATION UNTUK MELAKUKAN PERAMALAN
CURAH HUJAN DI KABUPATEN MALANG**

SKRIPSI

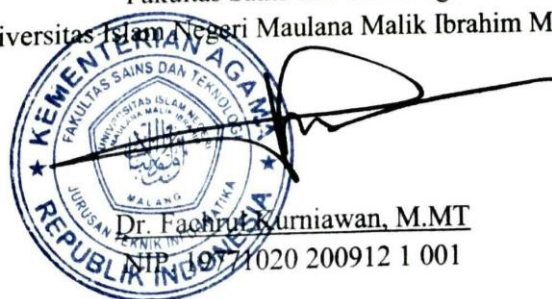
Oleh:
DONNY KURNIAWAN
NIM. 17650086

Telah Dipertahankan di Depan Dewan Penguji Skripsi
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
Untuk Memperoleh Gelar Sarjana Komputer (S.Kom)
Tanggal: 14 Desember 2021

Susunan Dewan Penguji

Penguji Utama	: <u>Agung Teguh Wibowo A, M.T</u> () NIP: 19860103 20180201 1 235
Ketua Penguji	: <u>Supriyono, M. Kom</u> () NIP: 19841010 201903 1 012
Sekretaris Penguji	: <u>Fatchurrohman, M.Kom</u> () NIP: 19700731 200501 1 002
Anggota Penguji	: <u>Khadijah Fahmi Hayati H, M.Kom</u> () NIP: 19900626 20160801 2 077

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



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

PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini:

Nama : Donny Kurniawan
NIM : 17650086
Jurusan : Teknik Informatika
Fakultas : Sains dan Teknologi

Menyatakan dengan sebenarnya bahwa Skripsi 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 Skripsi ini hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 14 Desember 2021
Yang membuat pernyataan,



Donny Kurniawan
NIM. 17650086

MOTTO

“Bakat itu tidak ada, yang ada adalah kerja keras”

HALAMAN PERSEMBAHAN

Puji syukur kehadiran Allah SWT, atas limpahan karunia-Nya penulis dapat menyelesaikan skripsi ini. Shalawat serta salam tak lupa saya ucapkan kepada Nabi Muhammad SAW. Saya mempersembahkan skripsi ini kepada:

Kedua orang tua saya Bapak Imam Fatah, dan Ibu Binti Sulaimah yang saya sangat sayangi. Terima kasih karena telah senantiasa mendukung dan mengiringi perjalanan saya dengan do'a dan motivasi yang tak terhitung. Dan terima kasih kepada kakak saya yang juga selalu memberikan dukungan. Semoga kebaikan, rahmat dan perlindungan Allah SWT senantiasa tercurah kepada mereka semua. Aamin.

Dosen pembimbing penulis, Bapak Fatchurrohman, M.Kom dan Ibu Khadijah Fahmi Hayati Holle, M.Kom yang telah dengan sabar membimbing penelitian skripsi ini dan selalu memberikan masukan dan semangat untuk menjalani setiap tahapan skripsi.

Seluruh Dosen Teknik Informatika Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang yang telah membimbing dan memberikan ilmu yang bermanfaat.

Tidak lupa seluruh Keluarga Teknik Informatika, terutama teman-teman Teknik Informatika angkatan 2017 yang telah memberikan semangat dan motivasi. Serta teman-teman penulis yang terus mendorong penulis untuk melewati dan menyelesaikan skripsi ini.

KATA PENGANTAR

Assalamu'alaikum Wr. Wb.

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

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

1. Prof. Dr. H. M. Zainuddin, M.A, selaku rektor UIN Maulana Malik Ibrahim Malang, yang telah banyak memberikan pengetahuan dan pengalaman yang berharga.
2. Dr. Sri Harini, M.Si selaku Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
3. Dr. Fachrul Kurniawan, M.MT, selaku Ketua Jurusan Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang
4. Fatchurrohman, M.Kom dan Khadijah Fahmi Hayati Holle, M.Kom selaku dosen pembimbing Skripsi, yang telah banyak memberikan pengarahan dan pengalaman yang berharga.
5. Segenap sivitas akademika Jurusan Teknik Informatika, terutama seluruh dosen, terima kasih atas segenap ilmu dan bimbingannya.
6. Ayahanda dan Ibunda tercinta yang senantiasa memberikan doa dan restunya kepada penulis dalam menuntut ilmu.
7. Kakak dan teman-teman penulis yang selalu memberikan semangat kepada penulis untuk menyelesaikan Skripsi ini.
8. Semua pihak yang ikut membantu dalam menyelesaikan Skripsi ini baik berupa materil maupun moril.

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

Wassalamu'alaikum Wr. Wb.

Malang, Desember 2021
Penulis

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PERSETUJUAN	ii
HALAMAN PENGESAHAN	iii
HALAMAN PERNYATAAN	iv
MOTTO	v
HALAMAN PERSEMBAHAN	vi
KATA PENGANTAR	vii
DAFTAR ISI	vii
DAFTAR GAMBAR	x
DAFTAR TABEL	xi
ABSTRAK	xii
ABSTRACT	xiii
الملخص	xiv
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Pernyataan Masalah	4
1.3 Batasan Masalah	4
1.4 Tujuan Penelitian	4
1.5 Manfaat Penelitian	5
BAB II STUDI PUSTAKA	6
2.1 Penelitian Terdahulu	6
2.2 Tinjauan Pustaka	9
2.2.1 Hujan	9
2.2.2 Curah Hujan	10
2.2.3 Peramalan/Forecasting	11
2.2.4 Peramalan berdasarkan waktu	12
2.2.5 <i>Mean Squared Error</i> (MSE)	13
2.2.6 <i>Weighted Mean Absolute Percentage Error</i> (WMAPE)	13
2.2.7 <i>Artificial Neural Network</i> (ANN)	14
2.2.8 <i>Arsitektur Artificial Neural Network</i>	17
2.2.9 Algoritma Backpropagation	18
2.2.10 Fungsi Aktivasi Sigmoid Biner	19
2.2.11 Bias Node	20
2.2.12 Normalisasi dan Denormalisasi Data	22
2.2.13 Proses Pelatihan	24
BAB III DESAIN DAN IMPLEMENTASI	27
3.1 Prosedur Penelitian	27
3.2 Data yang Digunakan	29
3.3 Desain Sistem	30

3.4	Perhitungan Manual	31
3.5	Implementasi Program	35
3.5.1	Implementasi Normalisasi Data	35
3.5.2	Implementasi Denormalisasi Data	36
3.5.3	Implementasi Pengubahan Skala Data	37
3.5.4	Implementasi MSE	38
3.5.5	Implementasi WMAPE	39
3.5.6	Implementasi Sigmoid Biner	40
3.5.7	Implementasi Input, Bias, Weight	40
3.5.8	Implementasi Forward Propagation dan Backpropagation	42
3.6	Skenario Uji Coba	44
BAB IV UJI COBA DAN PEMBAHASAN		46
4.1	Pengujian Model	46
4.1.1	Pengujian Model A	47
4.1.2	Pengujian Model B	48
4.1.3	Pengujian Model C	50
4.1.4	Pengujian Model D	51
4.1.5	Pengujian Model E	53
4.1.6	Pengujian Model F	54
4.1.7	Pengujian Model G	56
4.1.8	Pengujian Model H	57
4.1.9	Pengujian Model I	59
4.1.10	Pengujian Model J	60
4.2	Hasil Pengujian Model	62
BAB V KESIMPULAN DAN SARAN		66
5.1	Kesimpulan	66
5.2	Saran	67
DAFTAR PUSTAKA		
LAMPIRAN		

DAFTAR GAMBAR

Gambar 2.1	Arsitektur Artificial Neural Network	17
Gambar 2.2	Arsitektur fungsi aktivasi pada hidden layer ANN.	18
Gambar 2.3	Fungsi aktivasi sigmoid biner	20
Gambar 2.4	Arsitektur Neural Network dengan Bias node.	21
Gambar 3.1	Flowchart prosedur penelitian.	27
Gambar 3.2	Contoh arsitektur ANN dengan 3 hidden layer, dan 6 neuron	30
Gambar 3.3	Flowchart proses peramalan dengan ANN	31
Gambar 3.4	Kode program implementasi normalisasi data	35
Gambar 3.5	Kode program implementasi denormalisasi data	36
Gambar 3.6	Kode program implementasi pengubahan skala data	37
Gambar 3.7	Kode program proses perhitungan MSE.	39
Gambar 3.8	Kode program proses perhitungan WMAPE	39
Gambar 3.9	Kode program implementasi sigmoid biner	40
Gambar 3.10	Kode program implementasi input, bias, dan weight	41
Gambar 3.11	Kode program implementasi forward propagation dan backpropagation	43
Gambar 4.1	Plot data pengujian peramalan model A	48
Gambar 4.2	Plot data pengujian peramalan model B	49
Gambar 4.3	Plot data pengujian peramalan model C	51
Gambar 4.4	Plot data pengujian peramalan model D	52
Gambar 4.5	Plot data pengujian peramalan model E	54
Gambar 4.6	Plot data pengujian peramalan model F	55
Gambar 4.7	Plot data pengujian peramalan model G	57
Gambar 4.8	Plot data pengujian peramalan model H	58
Gambar 4.9	Plot data pengujian peramalan model I.	60
Gambar 4.10	Plot data pengujian peramalan model J.	61
Gambar 4.11	Grafik 10 model peramalan dari yang terkecil ke terbesar	63

DAFTAR TABEL

Tabel 2.1 Penelitian terdahulu.	6
Tabel 2.2 Persamaan dan perbedaan dengan penelitian terdahulu	9
Tabel 2.3. Tabel kesalahan/error peramalan.	14
Tabel 3.1 Contoh data curah hujan (mm) dari tahun 2010 hingga 2016 (Januari hingga Juli)	29
Tabel 4.1 Daftar pengujian model ANN.	46
Tabel 4.2 Hasil pengujian peramalan model A.	47
Tabel 4.3 Hasil pengujian peramalan model B.	48
Tabel 4.4 Hasil pengujian peramalan model C.	50
Tabel 4.5 Hasil pengujian peramalan model D.	51
Tabel 4.6 Hasil pengujian peramalan model E.	53
Tabel 4.7 Hasil pengujian peramalan model F	54
Tabel 4.8 Hasil pengujian peramalan model G.	56
Tabel 4.9 Hasil pengujian peramalan model H.	57
Tabel 4.10 Hasil pengujian peramalan model I.	59
Tabel 4.11 Hasil pengujian peramalan model J	60
Tabel 4.12 Hasil pengujian model.	62

ABSTRAK

Kurniawan, Donny. 2021. **Implementasi Algoritma *Artificial Neural Network Backpropagation* Untuk Melakukan Peramalan Curah Hujan di Kabupaten Malang**. Skripsi. Jurusan Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Fatchurrohman, M.Kom. (II) Khadijah Fahmi Hayati Holle, M.Kom.

Kata Kunci: Peramalan, Artificial Neural Network, Backpropagation, MSE, WMAPE, Sigmoid Biner, Gradient Descent, Time Series.

Kabupaten Malang merupakan wilayah yang padat akan kegiatan masyarakat di dalamnya, tak jarang bencana dari cuaca buruk dapat menimbulkan kerugian yang besar dan bahkan memakan korban jiwa. Untuk meminimalisir hal tersebut maka dilakukan penelitian yang dapat melakukan peramalan curah hujan. Artificial Neural Network (ANN) merupakan sebuah algoritma yang dapat melakukan prediksi dan pengenalan pola, dengan cara terus belajar memperkecil nilai kesalahan/*error*-nya, melakukan *epoch* secara berulang-ulang dan akan berhenti ketika nilai *error* sudah sesuai target atau sudah mencapai *epoch* maksimalnya. Dengan menggunakan data historis curah hujan dari tahun 2000 hingga 2020, fungsi aktivasi sigmoid biner, optimasi *gradient descent*, dan 10 model dengan jumlah *hidden layer*, neuron, *input*, *learning rate*, dan *max epoch* yang berbeda-beda, penelitian ini menghasilkan hasil terbaik dengan model yang memiliki fitur 1 *hidden layer*, 5 neuron, 3 *input*, *learning rate* 0,1, dan *epoch* sebanyak 100 kali, dengan MSE sebesar 0,17056, WMAPE 5,6036%, dan akurasi sebesar 94,3964%.

ABSTRACT

Kurniawan, Donny. 2021. **Implementation of Artificial Neural Network Backpropagation Algorithm to Rainfall Forecasting in Malang**. Undergraduate Thesis. Informatics Engineering Department, Faculty of Science and Technology, Islamic State University of Maulana Malik Ibrahim Malang. Advisor: (I) Fatchurrohman, M.Kom. (II) Khadijah Fahmi Hayati Holle, M.Kom.

Keywords: Forecasting, Artificial Neural Network, Backpropagation, MSE, WMAPE, Sigmoid Function, Gradient Descent, Time Series.

Malang is a city in East Java with a dense of human activities in it. Bad weather is one of the most common problems that can cause great damage and casualties. To minimize it, research on rainfall forecasting is carried out. Artificial Neural Network (ANN) is an algorithm that can predict and recognize patterns, by continuing to learn to minimize the value of the error, doing epochs repeatedly and will stop when the error value is on target or has reached its maximum epoch. By using historical rainfall data from 2000 to 2020, sigmoid activation function, gradient descent optimization, and 10 models with different numbers of hidden layers, neurons, inputs, learning rate, and max epoch. This research finds a model with the best results that have 1 hidden layer, 5 neurons, 3 input, 0.1 learning rate, and 100 max epoch of features, with 0.17056 of MSE, 5.6036% of WMAPE, and 94.3964% of accuracy.

الملخص

كرنياوان, دوبي, 2021. تطبيق الخوارزمية شبكة انتشار خلفي عصبونية اصطناعية للقيام بالتنبؤ بمطول الأمطار بتنسيق في منطقة مالانج. البحث الجامعي, قسم معلومات تقنية كلية العلوم والتكنولوجيا. جامعة مولانا مالك ابراهيم مالانج. المشرف: (1) فتح الرحمن الماجستير, (2) خادجة فهم هياتي هولي الماجستير.

الكلمات الرئيسية: التوقع، شبكة عصبونية اصطناعية، انتشار خلفي، MSE، WMAPE، دالة سينية، خوارزمية أصل التدرج، متسلسلة زمنية

المنطقة مالانج هو إحدى من المناطق في جاوة الشرقية المكتظة بالأنشطة المجتمعية فيها، وليس من النادر أن تتسبب الكوارث الناجمة عن سوء الأحوال الجوية في خسائر فادحة وحتى تودي بالأرواح. لتقليل ذلك، يتم إجراء بحث يمكنه التنبؤ بمطول الأمطار. شبكة عصبونية اصطناعية هي خوارزمية يمكنها التنبؤ والتعرف على الأنماط، من خلال التعلم المستمر لتقليل قيمة الخطأ إلى الحد الأدنى، وتكراره باستمرار وسيتوقف عندما تكون قيمة الخطأ على الهدف أو وصلت إلى الحد الأقصى من التكرار. باستخدام بيانات هطول الأمطار من 2000 إلى 2020، دالة سينية، تابع التفعيل، و 10 نماذج بأعداد مختلفة من الطبقات المخفية، والخلايا العصبية، والمدخلات، ومعدل التعلم، والعصر الأقصى. يجد هذا البحث نموذجًا بأفضل النتائج يحتوي على طبقة واحدة مخفية، و 5 خلايا عصبية، و 3 مدخلات، ومعدل تعليمي 0.1، و 100 حقبة كحد أقصى من الميزات، مع 0.17056 من MSE، و 5.6036٪ من WMAPE، و 94.3964٪ من الدقة.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Cuaca merupakan kejadian alam yang sudah melekat dalam kehidupan sehari-hari, karena banyak kegiatan manusia yang sangat bergantung dengan cuaca. Cuaca di Indonesia sendiri secara garis besar hanya terbagi dalam dua musim, yakni musim kemarau dan musim hujan. Hujan adalah suatu proses jatuhnya cairan dari langit (atmosfer bumi) dalam bentuk cair ke atas tanah. Hujan merupakan nikmat yang sangat besar pemberian dari Allah SWT, berdasarkan firman-Nya dalam surat Al-Baqarah ayat 22 yang berbunyi,

الَّذِي جَعَلَ لَكُمُ الْأَرْضَ فِرَاشًا وَالسَّمَاءَ بِنَاءً وَأَنْزَلَ مِنَ السَّمَاءِ مَاءً فَأَخْرَجَ بِهِ مِنَ الثَّمَرَاتِ رِزْقًا لَكُمْ ۗ فَلَا
تَجْعَلُوا لِلَّهِ أَنْدَادًا وَأَنْتُمْ تَعْلَمُونَ

“Dialah yang menjadikan bumi sebagai hamparan bagimu dan langit sebagai atap, dan Dia menurunkan air (hujan) dari langit, lalu Dia menghasilkan dengan hujan itu segala buah-buahan sebagai rezeki untukmu; karena itu janganlah kamu mengadakan sekutu-sekutu bagi Allah, padahal kamu mengetahui.” (Q.S Al-Baqarah ayat 22).

Ayat tersebut menjelaskan bahwa Allah telah menjadikan bumi sebagai hamparan bagi kita supaya kehidupan berjalan dengan mudah di atas permukaannya, dan langit sebagai atap yang kuat dan menurunkan hujan dari awan yang dengan itu Dia mengeluarkan beragam buah dan berbagai macam tumbuhan sebagai rizqi bagi kita. Maka janganlah kita sebagai manusia mengadakan tandingan-tandingan bagi Allah dalam beribadah sedangkan kita mengetahui keesaannya dalam menciptakan dan memberi rezeki serta hak tunggalnya untuk diibadahi (Tafsir Al-Muyassar).

Namun, hujan juga dapat menimbulkan bencana badai dan petir, hal ini pun juga telah difirmankan oleh Allah SWT dalam surat Ar-Ra'd ayat 12 yang berbunyi,

هُوَ الَّذِي يُرِيكُمُ الْبَرْقَ خَوْفًا وَطَمَعًا وَيُنزِلُ السَّحَابَ الثِّقَالَ

“Dialah Tuhan yang memperlihatkan kan kilat kepadamu untuk menimbulkan ketakutan dan harapan, dan Dia mengadakan awan mendung” (Q.S Ar-Ra'd ayat 12)

Dialah Dzat yang memperlihatkan kepada kalian sebagian dari tanda kekuasaanNya, berupa kilat yaitu cahaya amat terang yang tampak di sela awan-awan mendung, lalu kalian takut halilintar-halilintar yang membakar menyambar kalian darinya, dan kalian amat berharap hujan turun bersamaan dengannya. Dan dengan kuasaNya, Allah menciptakan awan yang penuh dengan air yang banyak untuk kepentingan-kepentingan kalian (Tafsir Al-Muyassar).

Badai atau cuaca buruk merupakan salah satu permasalahan yang terjadi ketika musim hujan tiba. Banyak sektor yang mendapat hambatan, seperti transportasi, pertanian, industri, hingga pariwisata. Dalam bidang transportasi misalnya, cuaca buruk akan berpengaruh pada kelancaran dalam berkendara di darat, laut, maupun udara. Dalam pertanian, petani akan mengaitkan dengan cuaca ketika masa awal tanam dimulai, dan memastikan bahwa tanaman yang akan mereka tanam cocok pada musim tersebut. Begitu pun dengan bidang industri dan pariwisata, mereka juga memiliki masalahnya masing-masing ketika terjadi cuaca buruk. (Navianti & Widjajati, 2012)

Data dari Badan Pusat Statistik, kabupaten Malang merupakan kabupaten di Jawa Timur dengan luas wilayah 2.977,05 km² dengan populasi sebanyak 2.619.975 jiwa, menjadikan Malang merupakan kabupaten terluas nomor dua di Jawa timur setelah Banyuwangi. Dikutip dari okezone.com, hujan deras pada Senin (18/1/2021) petang, banjir terjadi di beberapa lokasi di Malang. Hujan deras juga mengakibatkan

sungai meluap hingga mengakibatkan longsor di halaman rumah warga dan menghanyutkan seorang penghuni rumah. Hujan deras juga mengakibatkan sebagian rumah warga ikut longsor. Kejadian ini menimbulkan korban yakni pemilik rumah yang bernama Roland Sumarna yang hanyut terbawa longsor masuk ke aliran Sungai. (*Banjir Dan Longsor Terjang Kota Malang, Seorang Warga Hilang Diduga Tertimbun : Okezone News*, n.d.). Tidak hanya sekali, banjir juga terjadi hampir di setiap tahun. Hal inilah yang mendorong penulis untuk membuat prediksi curah hujan yang bermanfaat bagi masyarakat Malang khususnya di daerah yang rawan banjir atau longsor agar dapat melakukan antisipasi lebih dini terhadap curah hujan yang tinggi, sehingga dapat meminimalisir semua kemungkinan buruk yang dapat terjadi.

Prediksi cuaca telah banyak dilakukan untuk memperkirakan cuaca yang kira-kira yang akan terjadi di masa depan dengan mempertimbangkan banyak faktor alam maupun faktor historis yang pernah terjadi di masa lalu. Dengan perhitungan manual, tentunya akan susah untuk melakukan hal tersebut, cara yang mudah yakni menggunakan sebuah algoritma, salah satunya adalah *Artificial Neural Network*. *Artificial Neural Network* atau ANN merupakan kumpulan elemen pemrosesan sederhana yang saling terhubung yang disebut dengan neuron. Menggunakan skema mirip sistem saraf manusia, neuron saling berhubungan dengan neuron lain, tiap koneksi memiliki bobot, dan akhirnya dilatih, sehingga memperoleh output yang diinginkan. Pada penjelasan diatas, penulis akan melakukan peramalan dengan menggunakan algoritma *Artificial Neural Network* dalam studi kasus curah hujan di kabupaten Malang.

1.2 Pernyataan Masalah

1. Bagaimana pengaruh jumlah *hidden layer*, jumlah neuron, jumlah *input*, *learning rate*, dan jumlah *epoch* pada hasil algoritma *Artificial Neural Network Backpropagation* dalam melakukan peramalan curah hujan di kabupaten Malang?
2. Seberapa besar kesalahan/*error* dan akurasi algoritma *Artificial Neural Network Backpropagation* dalam melakukan peramalan curah hujan di kabupaten Malang?

1.3 Batasan Masalah

1. Pengukuran curah hujan dilakukan di kabupaten Malang.
2. Data historis curah hujan merupakan data rata-rata per-bulan antara tanggal 1 Januari 2000 hingga 31 Desember 2020, yang diperoleh dari stasiun BMKG di Genengan, Girimoyo, Kec. Karangploso, Kab. Malang.
3. Jarak waktu prediksi adalah 1 bulan.
4. Pengukuran kesalahan/*error* menggunakan MSE dan WMAPE

1.4 Tujuan Penelitian

1. Mengetahui pengaruh jumlah *hidden layer*, jumlah neuron, jumlah *input*, jumlah data latih, *learning rate*, dan jumlah *epoch* pada hasil algoritma *Artificial Neural Network* dalam memprediksi curah hujan di kabupaten Malang.
2. Mengetahui besar kesalahan/*error* algoritma *Artificial Neural Network* dalam memprediksi curah hujan di kabupaten Malang.

1.5 Manfaat Penelitian

1. Manfaat penelitian di bidang akademis adalah sebagai sumber referensi penelitian bagi pelajar, dan juga sebagai acuan untuk penelitian yang lebih baik di masa depan
2. Manfaat penelitian di bidang non-akademis adalah dapat digunakan sebagai prediksi curah hujan bagi masyarakat khususnya di daerah kabupaten Malang untuk mengetahui intensitas curah hujan yang akan terjadi dan meminimalisir kerugian yang akan terjadi pada curah hujan yang tinggi.

BAB II

STUDI PUSTAKA

2.1 Penelitian Terdahulu

Penelitian terdahulu bertujuan untuk mendapatkan perbandingan dan acuan dari penelitian lain, selain itu untuk menghindari anggapan kesamaan dengan penelitian yang telah ada. Maka dari itu dalam kajian pustaka ini peneliti mencantumkan dan menjelaskan hasil-hasil penelitian terdahulu pada tabel 2.1.

Tabel 2.1 Penelitian terdahulu

NO	PENELITI	JUDUL	METODE	VARIABEL	HASIL
1.	Oktaviani & Afdal (2013)	Prediksi Curah Hujan Bulanan Menggunakan Jaringan Syaraf Tiruan Dengan Beberapa Fungsi Pelatihan <i>Backpropagation</i> (Studi Kasus: Stasiun Meteorologi Tabing Padang, Tahun 2001-2012)	Metode <i>Backpropagation</i> , fungsi pelatihan <i>traingd</i> , <i>traingdm</i> , <i>traingdx</i>	Curah hujan kota Padang tahun 2001-2012	Semakin banyak jumlah neuron dan hidden layer, menghasilkan prediksi yang semakin bagus. Metode learning dengan <i>traingdx</i> dengan arsitektur (12,20,20,20,1) memiliki keberhasilan dalam pengenalan pola hingga 99.0%
2.	Handayani & Adri (2015)	Penerapan JST (<i>Backpropagation</i>) Untuk Prediksi Curah Hujan (Studi Kasus: Kota Pekanbaru)	Metode <i>Backpropagation</i> , fungsi aktivasi <i>logsig</i> , <i>tansig</i>	Kelembaban udara, temperatur, kecepatan angin, tahun 2012-2013 Pekanbaru	Mendapatkan hasil akurasi sebesar 96% menggunakan 100 hidden layer dengan fungsi aktivasi <i>logsig</i> dan jumlah <i>epoch</i> sebanyak 1000
3.	Andrian & Ningsih (2017)	Prediksi Curah Hujan di Kota Medan Menggunakan Metode <i>Backpropagation Neural Network</i>	Metode <i>Backpropagation</i>	Curah hujan kota Medan tahun 1997-2012	Menghasilkan hasil terbaik pada 5 hidden layer dengan akurasi 43,37% pada target error 0,0072 dan total iterasi sebanyak 3000. Hasil terendahnya terdapat pada 7 hidden layer dengan akurasi

					42,88% dan jumlah iterasi sebanyak 4500 kali.
4.	Abhishek <i>et al.</i> , (2012)	<i>Weather Forecasting Model Using Artificial Neural Network</i>	Metode <i>Backpropagation</i> , fungsi aktivasi <i>tansig</i> , <i>purelin</i>	Curah hujan selama 10 tahun	Menambahkan jumlah neuron/layer dapat memperkecil MSE (menambah akurasi). Menambah data latih dapat menambah akurasi. MSE yang kecil tidak selalu menghasilkan hasil peramalan yang bagus, ada kemungkinan terjadinya <i>overfitting</i> .
5.	Sutawinaya <i>et al.</i> , (2017)	Perbandingan Metode Jaringan Syaraf Tiruan Pada Peramalan Curah Hujan	Metode <i>Backpropagation</i> , <i>adaline</i> , fungsi pelatihan <i>traingdx</i>	Curah hujan kota Denpasar tahun 2006-2016	Semakin banyak iterasi maka nilai MSE dan RMSE semakin mengecil. Backpropagation mengalami penurunan nilai MSE dan RMSE yang lebih signifikan daripada ADALINE. Metode <i>Backpropagation</i> lebih unggul dengan nilai MSE 0.0019 dan RMSE 0.0435, sedangkan ADALINE memiliki nilai MSE 0.0142 dan RMSE 0.1192, masing-masing 1000 iterasi.

Berdasarkan penelitian di atas, penulis menemukan beberapa hasil yang relevan sebagai pendukung penelitian ini, seperti penelitian yang telah dilakukan oleh (Oktaviani & Afdal, 2013). Pada penelitian ini melakukan prediksi curah hujan bulanan menggunakan jaringan syaraf tiruan. Penelitian ini menghasilkan akurasi prediksi yang tinggi, jika menambah jumlah neuron dan lapisan tersembunyi, dan metode *traingdx* memiliki keberhasilan dalam pengenalan pola hingga 99.0%

Penelitian kedua dilakukan oleh (Handayani & Adri, 2015) yang melakukan penelitian tentang penerapan jaringan syaraf tiruan untuk prediksi curah hujan. Dalam penelitian ini peramalan dilakukan menggunakan beberapa faktor cuaca seperti kelembaban udara, temperatur, dan kecepatan angin. Mendapatkan akurasi sebesar 96% menggunakan fungsi aktivasi *logsig* dengan jumlah 100 hidden layer dan 1000 *epoch*.

Penelitian ketiga dilakukan oleh (Andrian & Ningsih, 2017) yang melakukan penelitian tentang prediksi curah hujan menggunakan metode *backpropagation neural network*. Penelitian ini mencoba pengaruh jumlah *hidden layer* dan total iterasi, dengan hasil terbaik pada 5 *hidden layer* dan 3000 iterasi yang memiliki akurasi sebesar 43,37% pada target error 0,0072.

Penelitian keempat dilakukan oleh (Abhishek et al., 2012) yang melakukan penelitian tentang peramalan cuaca menggunakan *artificial neural network*. Penelitian ini menggunakan fungsi aktivasi *purelin* dan *tansig* dengan menambah jumlah data latih dapat memperbesar akurasi. Dalam penelitian ini juga menyinggung tentang *overfitting*.

Penelitian kelima dilakukan oleh (Sutawinaya et al., 2017) yang melakukan penelitian tentang peramalan curah hujan menggunakan jaringan syaraf tiruan. Penelitian ini menggunakan dua metode arsitektur yakni *Backpropagation* dan ADALINE. Dengan fungsi pembelajaran TRAINGD, memperoleh hasil bahwa *Backpropagation* lebih unggul daripada ADALINE.

Dalam penelitian ini terdapat beberapa persamaan dan perbedaan dengan penelitian terdahulu yang telah disajikan dalam tabel 2.1. Adapun persamaan dan perbedaannya disajikan dalam tabel 2.2 berikut.

Tabel 2.2 Persamaan dan perbedaan dengan penelitian terdahulu

NO	PENELITI	JUDUL	PERSAMAAN	PERBEDAAN
1.	Oktaviani & Afdal (2013)	Prediksi Curah Hujan Bulanan Menggunakan Jaringan Syaraf Tiruan Dengan Beberapa Fungsi Pelatihan <i>Backpropagation</i> (Studi Kasus: Stasiun Meteorologi Tabing Padang, Tahun 2001-2012)	<ul style="list-style-type: none"> • Variabel curah hujan • Metode <i>Backpropagation</i> 	<ul style="list-style-type: none"> • Lokasi dan tahun variabel • Fungsi pelatihan
2.	Handayani & Adri (2015)	Penerapan JST (<i>Backpropagation</i>) Untuk Prediksi Curah Hujan (Studi Kasus: Kota Pekanbaru)	<ul style="list-style-type: none"> • Metode <i>Backpropagation</i> • Fungsi aktivasi <i>logsig</i> 	<ul style="list-style-type: none"> • Variabel • Fungsi aktivasi <i>tansig</i>
3.	Andrian & Ningsih (2017)	Prediksi Curah Hujan di Kota Medan Menggunakan Metode <i>Backpropagation Neural Network</i>	<ul style="list-style-type: none"> • Metode <i>Backpropagation</i> • Variabel curah hujan 	<ul style="list-style-type: none"> • Lokasi dan tahun variabel
4.	Abhishek et al., (2012)	<i>Weather Forecasting Model Using Artificial Neural Network</i>	<ul style="list-style-type: none"> • Metode <i>Backpropagation</i> • Variabel curah hujan 	<ul style="list-style-type: none"> • Lokasi dan tahun variabel • Fungsi aktivasi <i>tansig, purelin</i>
5.	Sutawinaya et al., (2017)	Perbandingan Metode Jaringan Syaraf Tiruan Pada Peramalan Curah Hujan	<ul style="list-style-type: none"> • Metode <i>Backpropagation</i> • Variabel curah hujan 	<ul style="list-style-type: none"> • Lokasi dan tahun variabel • Metode <i>adaline</i> • Fungsi pelatihan

2.2 Tinjauan Pustaka

2.2.1 Hujan

Hujan adalah jatuhnya cairan yang berupa cair maupun beku dari atmosfer ke permukaan bumi (proses presipitasi). Hujan memerlukan lapisan atmosfer yang tebal agar menemukan suhu di atas titik beku di atas permukaan bumi. Lalu terjadilah proses kondensasi (berubahnya wujud benda ke wujud baru yang lebih padat) uap air pada atmosfer bumi menjadi butiran-butiran air yang akhirnya jatuh ke bumi. (Prawaka et al., 2016).

Butiran air yang jatuh sebagai presipitasi akan beragam. Butiran air dengan ukuran lebih dari 0,5 mm dikenal sebagai hujan, ukuran antara 0,2 mm sampai 0,5 mm dikenal sebagai gerimis, sedangkan untuk ukuran butiran air yang kurang dari 0,2 mm tidak akan sampai ke permukaan bumi, karena akan menguap dalam perjalanannya menuju permukaan bumi. (Handayani & Adri, 2015).

Dilihat dari bentuk dan sifatnya, hujan dibagi menjadi dua, yakni hujan *shower* atau tiba-tiba, dan hujan kontinyu atau berkelanjutan. Hujan *shower* biasanya ditandai dengan permulaan dan akhir yang mendadak dengan intensitas cepat, biasanya datang dari awan *Cumulus* (Cu) ataupun *Cumulonimbus* (Cb) dengan tetesan air yang lebih besar daripada hujan biasa. Sedangkan untuk hujan kontinyu, permulaan dan akhirnya tidak secara mendadak dan awan tidak tampak mengalami perubahan sejak permulaan hingga akhir. Biasanya jatuh dari awan *Stratus* (St), *Altostratus* (As), *Nimbostratus* (Ns), yakni awan-awan yang umumnya berbentuk merata. (Fadholi, 2013).

2.2.2 Curah Hujan

Curah hujan merupakan salah satu unsur iklim yang penting bagi bumi. Pengukuran curah hujan memakai satuan inci atau milimeter, artinya apabila menunjukkan curah hujan 1 mm, air memiliki ketinggian 1 mm setiap meter persegi (m²) dengan catatan air itu tidak meresap, menguap, ataupun mengalir. Hujan berperan penting dalam siklus hidrologi yang secara langsung mempengaruhi keseimbangan sumber daya air di bumi. Hal tersebut akhirnya dapat dimanfaatkan oleh semua makhluk hidup khususnya manusia untuk

memenuhi kebutuhan dan keberlangsungan kesejahteraan dengan mengolahnya menjadi banyak hal misalnya perikanan, irigasi pertanian, cadangan air, serta sebagai kebutuhan energi. (Syaifullah, 2014).

Berdasarkan penggolongan iklim dari data curah hujan, perbandingan antara jumlah rata-rata bulan kering dengan jumlah rata-rata bulan basah menyimpulkan bahwa apabila curah hujan kurang dari 60 mm/bulan, maka masuk sebagai bulan kering. Sedangkan apabila curah hujan lebih dari 100 mm/bulan, maka bisa dikatakan sebagai bulan basah. Di antara kedua bulan tersebut, terdapat bulan lembab, yaitu apabila curah hujan antara 60-100 mm/bulan. (Warsito & Sumiyati, 2007).

2.2.3 Peramalan/Forecasting

Peramalan adalah suatu teknik analisa perhitungan yang dilakukan dengan pendekatan kualitatif maupun kuantitatif untuk memperkirakan kejadian dimasa depan dengan menggunakan referensi data-data di masa lalu. Peramalan memiliki tujuan untuk memperkirakan banyak hal di masa depan seperti dalam kegiatan ekonomi dalam kenaikan harga, tren, dan pengaruh lingkungan lainnya.

(Heizer & Render, 2009) mengatakan, peramalan (*forecasting*) merupakan ilmu dan seni untuk memperkirakan suatu kejadian di masa yang akan datang. Hal ini dapat dilakukan dengan melibatkan pengambilan data historis dan memproyeksikannya ke masa mendatang dengan suatu bentuk model matematis. Selain itu, prediksi bisa juga dari intuisi yang bersifat subjektif, atau dapat melakukan kombinasi model matematis yang telah disesuaikan oleh seorang manajer dengan pertimbangan yang baik.

Fungsi peramalan atau *forecasting* akan terlihat ketika keputusan diambil. Keputusan yang baik adalah keputusan yang sesuai dengan apa yang telah diperkirakan dan bahkan dalam memperkirakan keputusan selanjutnya. Apabila ramalan yang disusun kurang tepat, maka kita akan menghadapi banyak masalah seperti model peramalan tersebut (Ginting, 2007).

2.2.4 Peramalan Berdasarkan Waktu

(Herjanto, 2007) mengatakan bahwa peramalan berdasarkan panjang waktu dapat dibagi menjadi tiga jenis, yaitu:

1. Peramalan jangka panjang, yaitu peramalan yang mencakup waktu lebih dari 18 bulan. Misal, peramalan yang berkaitan dengan penanaman modal, perencanaan fasilitas, dan perencanaan untuk kegiatan litbang.
2. Peramalan jangka menengah, yaitu peramalan yang mencakup waktu antara 3 sampai 18 bulan. Misal, peramalan pada perencanaan penjualan, produksi, dan perencanaan tenaga kerja tidak tetap.
3. Peramalan jangka pendek, yaitu peramalan yang mencakup jangka waktu kurang dari 3 bulan. Misalnya, peramalan yang berhubungan dengan perencanaan pembelian material, penjadwalan kerja, dan penugasan tenaga kerja.

2.2.5 Mean Squared Error (MSE)

Mean Squared Error (MSE), atau *Mean Squared Deviation* (MSD) adalah metode untuk mengevaluasi peramalan, yaitu rata-rata selisih kuadrat antara nilai taksiran dan nilai sebenarnya. Masing-masing kesalahan atau sisa

dikuadratkan, kemudian dijumlahkan dan ditambahkan dengan jumlah observasi. Pendekatan ini dapat menghasilkan kesalahan peramalan yang besar karena kesalahan-kesalahan tersebut dikuadratkan. *Mean Squared Error* selalu bernilai positif (dan bukan nol). Semakin kecil nilai MSE maka semakin baik model peramalan tersebut (Ferdiansyah, n.d.). Metode ini ditentukan dalam rumus 2.1 berikut.

$$MSE = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2 \quad (2.1)$$

Keterangan:

- Y_i = nilai ke-i sesungguhnya (*actual value*)
- \hat{Y}_t = hasil peramalan ke-t (*forecast value*)
- N = banyaknya data peramalan

2.2.6 Weighted Mean Absolute Percentage Error (WMAPE)

Weighted Mean Absolute Percentage Error (WMAPE) adalah perhitungan persentase kesalahan/error peramalan yang terjadi antara bobot dengan nilai aktual. WMAPE merupakan varian dari *Mean Absolute Percentage Error* (MAPE). MAPE sendiri adalah perhitungan kesalahan peramalan antara hasil peramalan dengan nilai aktual. Penggunaan WMAPE dengan MAPE memiliki fungsi yang sama, yakni untuk menghitung persentase kesalahan/error peramalan. Namun MAPE memiliki kekurangan dalam beberapa kasus, diantaranya:

- Tidak bisa dilakukan apabila data aktual adalah nol.
- Akan memungkinkan hasil angka yang sangat tinggi apabila data aktual mendekati nol.

Menurut (Lewis, 1982), *error* dari peramalan dapat dikelompokkan menjadi 4 kategori, yang disajikan dalam tabel 2.3:

Tabel 2.3 Tabel kesalahan/*error* peramalan

Hasil	Keterangan
< 10%	Hasil peramalan tinggi
10% - 20%	Hasil peramalan baik
20% - 50%	Hasil peramalan sedang
> 50%	Hasil peramalan rendah/buruk

Perhitungan dari WMAPE sendiri ditentukan dalam rumus 2.2 sebagai berikut.

$$WMAPE = \frac{\sum_{i=1}^n |Y_i - \hat{Y}_i|}{\sum_{i=1}^n |Y_i|} \times 100\% \quad (2.2)$$

Keterangan:

- Y_i = nilai ke-i sesungguhnya (*actual value*)
- \hat{Y}_t = hasil peramalan ke-t (*forecast value*)
- N = banyaknya data peramalan

2.2.7 Artificial Neural Network (ANN)

Artificial Neural Network (ANN) atau Jaringan Syaraf Tiruan (JST) adalah sistem komputasi di mana arsitektur dan operasinya diilhami dari pengetahuan tentang sel saraf di dalam otak dan menggunakan cara kerja biologisnya sebagai model dari algoritma. Di dalam Al-Qur'an, otak manusia secara tersirat disebutkan sebagai "ubun-ubun" pada ayat Al-Alaq ayat 16 yang berbunyi,

نَاصِيَةٍ كَاذِبَةٍ خَاطِئَةٍ

“Yaitu ubun-ubun orang yang mendustakan dan durhaka.” (Q.S Al-Alaq ayat 16)

Manusia merupakan makhluk ciptaan Allah SWT yang paling sempurna dan sebaik-baiknya ciptaan dibanding makhluk lain, salah satu alasannya yakni manusia memiliki otak untuk berfikir dan berakal. Allah SWT berfirman dalam Al Qur’an surat Yusuf ayat 2,

إِنَّا أَنْزَلْنَاهُ قُرْآنًا عَرَبِيًّا لَعَلَّكُمْ تَعْقِلُونَ

“Sesungguhnya Kami menurunkannya berupa Al-Qur’an dengan berbahasa Arab, agar kamu berfikir (memahaminya)” (Q.S Yusuf ayat 2).

Allah SWT menyebutkan kata *Aqal* dalam Al-Qur’an 48 kali. Hal ini menyatakan fakta bahwa otak manusia merupakan sebuah anugrah yang tak ternilai yang diberikan kepada manusia sehingga bisa berfikir dan sebagai sumber kecerdasan. Dalam Islam, kecerdasan manusia bukanlah semata-mata kemampuan kognitif, namun orang cerdas adalah orang yang melakukan Muhasabah diri di dunia, dan beramal untuk hidup setelah mati. Seperti yang telah diriwayatkan dari Syaddad bin Aus Radhiyallahu Anhu, dari Nabi Shallallahu Alaihi wa Sallam, beliau bersabda,

الْكَيْسُ مَنْ دَانَ نَفْسَهُ وَعَمِلَ لِمَا بَعْدَ الْمَوْتِ، وَالْعَاجِزُ مَنْ أَتْبَعَ نَفْسَهُ هَوَاهَا وَتَمَنَّى عَلَى اللَّهِ.

Artinya: “Orang yang akalnya cerdas adalah yang melakukan Muhasabah diri (di dunia), dan berbuat untuk hidup setelah mati. Sedangkan orang yang akalnya lemah adalah yang mengikuti dorongan hawa nafsunya dan berangan-angan kosong terhadap Allah.” (HR. Ahmad & At-Tirmidzi).

Umar bin Khaththab Radhiyallahu Anhu juga berkata, “Lakukanlah Muhasabah diri sebelum kalian dihisab. Takarlah amal perbuatan kalian sebelum amal perbuatan kalian ditimbang, dan persiapkanlah diri kalian untuk menghadapi hari pemeriksaan terbesar (Hari Kiamat).”

ANN didasarkan pada kumpulan unit atau node yang terhubung yang disebut neuron, seperti memodelkan neuron di otak biologis. Setiap koneksi, seperti sinapsis di otak biologis, dapat mengirimkan sinyal ke neuron lain. Neuron buatan yang menerima sinyal kemudian memprosesnya dan dapat memberi sinyal pada neuron yang terhubung dengannya. "Sinyal" pada suatu koneksi adalah bilangan *real*, dan *output* dari setiap neuron dihitung oleh beberapa fungsi non-linier dari jumlah inputnya. Koneksi tersebut disebut *edge*.

Neuron dan *edge* biasanya memiliki bobot yang menyesuaikan saat pembelajaran berlangsung, nilai bobotnya bisa bertambah, ataupun berkurang. Biasanya, neuron dikumpulkan menjadi *layer*/lapisan. *Layer-layer* ini biasanya dibagi menjadi tiga posisi, yakni *layer input*, *hidden layer*, dan *layer output*. *Layer* yang berbeda dapat melakukan transformasi yang berbeda pada masukannya. Sinyal bergerak dari *layer* pertama (*layer input*), ke lapisan terakhir (*layer output*). (Ashari & Muniar, 2015).

ANN merupakan salah satu contoh model non-linier yang memiliki bentuk fungsional fleksibel dan mengandung beberapa parameter yang tidak dapat diinterpretasikan seperti pada model parametrik namun hal inilah yang membuat ANN mampu menyelesaikan persoalan yang tidak terstruktur dan sulit didefinisikan (Susanto, 2016).

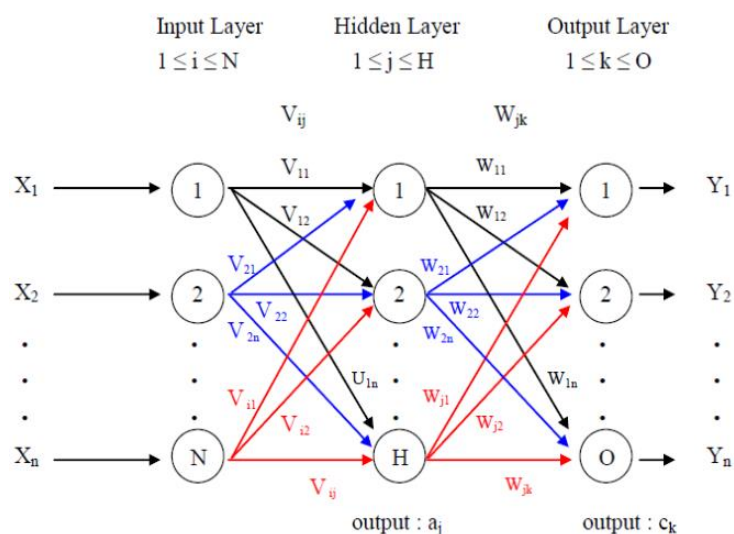
Fungsi dari *Artificial Neural Network* diantaranya adalah:

- Pengklasifikasian pola,
- Memetakan pola yang ada pada *input* ke dalam pola baru pada *output*,
- Penyimpan pola yang akan dipanggil kembali,

- Memetakan pola-pola yang sejenis,
- Pengoptimasi permasalahan,
- Prediksi.

2.2.8 Arsitektur *Artificial Neural Network*

Arsitektur atau struktur ANN adalah gambaran susunan komponen *layer* dan neuron pada *input*, *hidden* dan *output* yang terhubung dengan bobot, fungsi aktivasi dan fungsi pembelajaran (*learning function*). Arsitektur *Artificial Neural Network* digambarkan pada gambar 2.1.



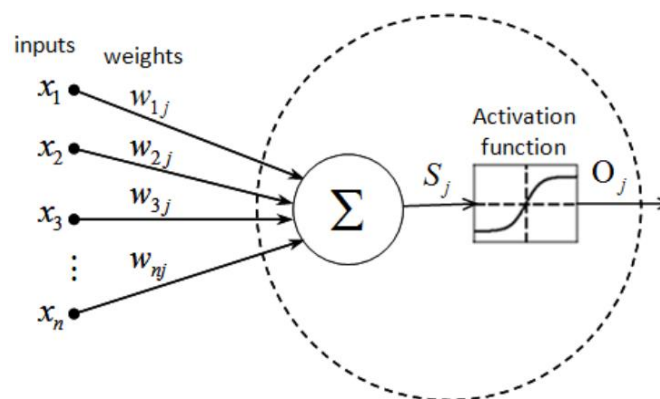
Gambar 2.1 Arsitektur Artificial Neural Network (Navianti & Widjajati, 2012)

Keterangan:

- $X_1 \dots X_n$: *input* pada jaringan
- i, j, k : variabel indeks pada *input*
- V_{ij} : bobot koneksi sel ke- i lapisan *input layer* dengan sel ke- j pada *hidden layer*
- W_{jk} : bobot koneksi sel ke- j pada lapisan *hidden layer* dengan sel ke- k pada *output layer*
- N : jumlah sel pada *input layer*

- H : jumlah sel pada *hidden layer*
- O : jumlah sel pada *output layer*
- a_j : keluaran pada *hidden layer*
- c_k : keluaran pada *output layer*

Dengan fungsi aktivasi yang terdapat pada tiap hidden layer seperti pada gambar 2.2,



Gambar 2.2 Arsitektur fungsi aktivasi pada hidden layer ANN (Zacharis, 2016)

2.2.9 Algoritma Backpropagation

Algoritma *Backpropagation* adalah metode yang digunakan untuk menyesuaikan bobot koneksi untuk memperkecil setiap kesalahan yang ditemukan selama pembelajaran. Algoritma *backpropagation* menggunakan *error output* untuk mengubah nilai bobotnya ke arah mundur (*backward*). Untuk mendapatkan error ini, tahap maju (*forward propagation*) harus dikerjakan terlebih dahulu. (Kiki & Kusumadewi, 2004).

Tiap-tiap hubungan tersebut mempunyai bobot koneksi (*weight*) yang dilatih untuk mencapai respon yang diinginkan. Output diperoleh dari pelatihan terhadap data berdasarkan bobot-bobot yang koneksi. Metode yang digunakan untuk menentukan bobot koneksi tersebut dinamakan algoritma pelatihan

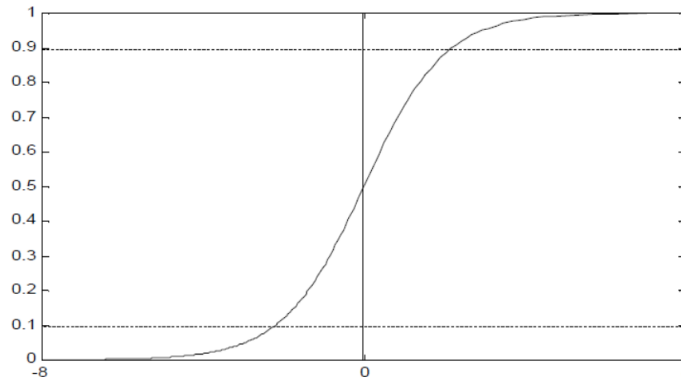
(*training algorithm*). Tiap-tiap hubungan antar neuron mempunyai bobot koneksi yang dilatih untuk mencapai respon yang diinginkan dengan melakukan suatu proses pelatihan (*training*). Selama proses pelatihan, terjadi perubahan yang cukup berarti pada bobot-bobot yang menghubungkan antar neuron.

Pada saat pelatihan dilakukan pada *input* yang berbeda, maka nilai bobot akan diubah secara dinamis hingga mencapai suatu nilai yang cukup seimbang. Apabila nilai ini telah tercapai mengindikasikan bahwa tiap-tiap *input* telah berhubungan dengan output yang diharapkan (Warsito & Sumiyati, 2007).

2.2.10 Fungsi Aktivasi Sigmoid Biner

Fungsi aktivasi merupakan fungsi yang digunakan pada jaringan saraf untuk mengaktifkan atau tidak mengaktifkan neuron. Karakteristik yang harus dimiliki oleh fungsi aktivasi jaringan perambatan balik antara lain harus kontinyu, terdiferensialkan, dan tidak menurun secara monotonis (*monotonically non-decreasing*). Lalu, untuk efisiensi komputasi, turunan fungsi tersebut mudah didapatkan dan nilai turunannya dapat dinyatakan dengan fungsi aktivasi itu sendiri. Fungsi aktivasi yang akan digunakan adalah sigmoid biner. Fungsi ini memiliki rentang nilai antara 0 sampai 1 (Julpan et al., 2018), dan ditunjukkan dalam rumus 2.3, dengan hasil kurva pada gambar 2.3.

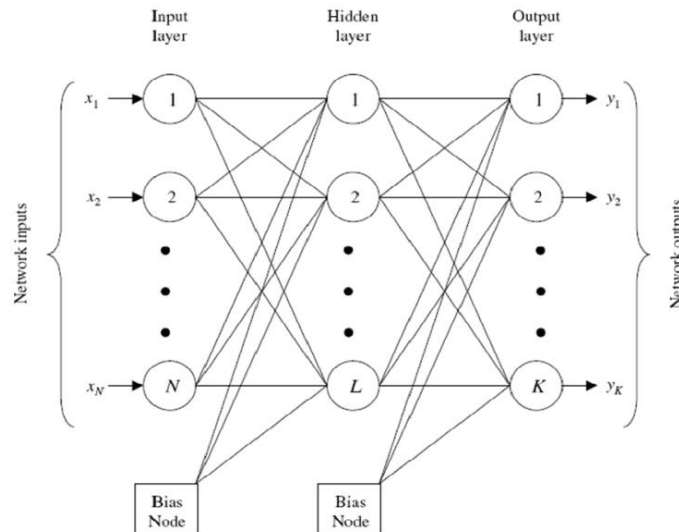
$$y = f(x) = \frac{1}{(1+e^{-x})} \quad (2.3)$$



Gambar 2.3 Fungsi aktivasi sigmoid biner (Fausett, 2006)

2.2.11 Bias Node

Artificial Neural Network yang memiliki *node* Bias adalah salah satu dari beberapa macam arsitektur ANN. Bias adalah nilai yang ditambahkan ke setiap neuron (kecuali pada *input layer*) selama fase *Forward Propagation*. Arsitektur dengan Bias *node* ditunjukkan pada gambar 2.4.



Gambar 2.4 Arsitektur Neural Network dengan Bias node (Lahiri & Ghanta, 2009)

Bias memiliki fungsi untuk menggeser kurva fungsi aktivasi secara horizontal (kiri/kanan) sambil membiarkan bentuk/kelengkungan yang tidak berubah. Ini akan memungkinkan jaringan untuk menghasilkan *output* yang

berbeda dari model biasa, sehingga dapat menyesuaikan/menggeser pemetaan kurva sesuai dengan kebutuhan.

2.2.12 Normalisasi dan Denormalisasi Data

Normalisasi data adalah penyesuaian data/nilai yang telah diukur pada skala yang berbeda ke skala yang tertentu. Normalisasi akan mengubah skala data menjadi antara nol hingga satu. Hal ini dilakukan karena data akan melewati fungsi aktivasi sigmoid yang juga akan menampilkan hasil dengan skala data antara nol hingga satu. Rumus dari normalisasi data ditunjukkan pada rumus 2.4.

$$X' = \frac{(X - X_{min})}{(X_{max} - X_{min})} \quad (2.4)$$

Keterangan:

- X' = Data hasil dari normalisasi
- X = Data asli
- X_{min} = Data asli yang terkecil
- X_{max} = Data asli yang terbesar

Setelah data dinormalisasi dan dihitung dengan fungsi aktivasi, maka hasil perlu dilakukannya denormalisasi. Denormalisasi adalah kebalikan dari normalisasi, yakni mengembalikan nilai dari skala tertentu menjadi skala awal. Proses ini dilakukan menggunakan rumus 2.5 sebagai berikut.

$$X = X'(X_{max} - X_{min}) + X_{min} \quad (2.5)$$

Keterangan

- X = Data asli
- X' = Data hasil dari normalisasi

- X_{min} = Data asli yang terkecil
- X_{max} = Data asli yang terbesar

Selain menggunakan kedua rumus tersebut, normalisasi/denormalisasi juga bisa menggunakan rumus konversi rentang angka, yakni mengonversi rentang angka lama ke rentang angka yang baru. Konversi ini lebih fleksibel karena bisa menentukan berapa nilai *MIN* dan *MAX* untuk rentang yang baru. Rumus 2.6 menunjukkan rumus untuk merubah skala deret angka dengan *MIN* dan *MAX* yang dapat disesuaikan.

$$X' = ((X - X_{min}) / (X_{max} - X_{min})) * (Y_{max} - Y_{min}) + Y_{min} \quad (2.6)$$

Keterangan

- X' = Data hasil normalisasi/konversi
- X = Data asli
- X_{min} = Data asli terkecil
- X_{max} = Data asli terbesar
- Y_{min} = Data baru terkecil
- Y_{max} = Data baru terbesar

2.2.13 Proses Pelatihan

Pelatihan/*training* adalah melatih *Artificial Neural Network* dengan memproses data latih yang berisi masukan dan keluaran sebagai contoh, dan memperbaharui pembobotan jaringan sehingga dapat menerima *input* baru dan menghasilkan hasil pendekatan yang maksimal. *Artificial Neural Network* akan belajar dengan terus memperkecil nilai kesalahan/*error*-nya dengan dilakukannya *epoch* secara berulang-ulang dan akan berhenti ketika nilai *error*

sudah sesuai target yang ditentukan. Proses pembelajaran ini biasa dikenal dengan pembelajaran yang diawasi (*supervised*).

Pelatihan *backpropagation* akan melewati tiga tahap. Tahap pertama adalah tahap maju (*forward propagation*), yakni data berjalan maju dari *input* menuju *output*. Tahap yang kedua yakni tahap mundur (*backpropagation*) yakni data berjalan dari *output* menuju *input*, proses ini bertujuan untuk mengetahui nilai *error* antara hasil perhitungan dengan target. Setelah mengetahui nilai *error*, tahap ketiga yakni memperbaharui nilai bobot, agar perhitungan selanjutnya bisa menjadi lebih baik dengan nilai *error* yang lebih kecil. Berikut adalah proses perhitungannya:

Langkah 0. Inisialisasi semua bobot dan bias dengan angka acak skala kecil (-1 s/d 1).

(Fase forward propagation)

Langkah 1. Tiap *input* (x_i) menerima sinyal dan meneruskan ke *hidden layer*

Langkah 2. Hitung unit $Z_{net\ ij}$ ($ij = 1, 2, 3, \dots, n$)

$$Z_{net\ ij} = \left(\sum_{i,j=1}^n w_{ij} * x_{ij} \right) + b_{ij} \quad (2.7)$$

Hitung keluaran dari *hidden layer* Z_{ij} ($ij = 1, 2, 3, \dots, n$). Hitung fungsi dengan aktivasi

$$Z_{ij} = f(z_{net\ ij}) = \frac{1}{1+e^{-z_{net\ ij}}} \quad (2.8)$$

Langkah 3. Selanjutnya hitung unit O_{jk} ($jk = 1, 2, 3, \dots, n$)

$$O_{net\ jk} = \left(\sum_{j,k=1}^n w_{jk} * Z_{jk} \right) + b_{jk} \quad (2.9)$$

Hitung keluaran fungsi dengan aktivasi.

$$O_{jk} = f(O_{net\ jk}) = \frac{1}{1+e^{-O_{net\ jk}}} \quad (2.10)$$

(Fase backpropagation)

Langkah 4. Hitung faktor δ dari setiap unit keluaran O_{jk} ($jk = 1, 2, 3, \dots, n$)

$$\delta_{jk} = (O'_{jk} - O_{jk})f'(O_{net\ jk}) = (O'_{jk} - O_{jk})O_{jk}(1 - O_{jk}) \quad (2.11)$$

δ_{jk} merupakan nilai *error* yang akan digunakan untuk memperbaharui nilai bobot (w_{jk}) dengan laju percepatan α . O' merupakan *output* sesungguhnya, dan O adalah *output* hasil prediksi.

Langkah 5. Hitung faktor δ dari setiap unit keluaran Z_{ij} ($ij = 1, 2, 3, \dots, n$)

$$\delta_{ij} = f'(Z_{net\ ij})(\delta_{jk}w_{jk}) = Z_{ij}(1 - Z_{ij})(\delta_{jk}w_{jk}) \quad (2.12)$$

Langkah 6. Hitung *weight* dan bias yang baru

$$\Delta w_{jk} = \alpha \delta_{jk} Z_{ij}; \quad jk = 1, 2, \dots, n; \quad (2.13)$$

$$\Delta w_{ij} = \alpha \delta_{ij} x_i; \quad ij = 1, 2, \dots, n; \quad i = 1, 2, \dots, n \quad (2.14)$$

Untuk bias (b)

$$\Delta b_{jk} = \alpha \delta_{jk}; \quad jk = 1, 2, \dots, n; \quad (2.15)$$

$$\Delta b_{ij} = \alpha \delta_{ij}; \quad ij = 1, 2, \dots, n; \quad (2.16)$$

Langkah 7. Lakukan langkah 1 sampai 6 dengan menggunakan semua *input* & *output* yang terdapat dalam data latih.

(Memperbaharui bobot)

Langkah 8. Jumlahkan total semua Δw_{ij} , Δw_{jk} , Δb_{ij} , Δb_{jk} hasil perulangan/*looping* pada langkah sebelumnya.

$$\Delta w'_{ij} = \text{sum}(\Delta w_{ij}) \quad (2.17)$$

$$\Delta w'_{jk} = \text{sum}(\Delta w_{jk}) \quad (2.18)$$

$$\Delta b'_{ij} = \text{sum}(\Delta b_{ij}) \quad (2.19)$$

$$\Delta b'_{jk} = \text{sum}(\Delta b_{jk}) \quad (2.20)$$

Langkah 9. Perbaharui nilai *weight* dan bias

$$w_{ij} = w_{ij} + \Delta w'_{ij} \quad (2.21)$$

$$w_{jk} = w_{jk} + \Delta w'_{jk} \quad (2.22)$$

$$b_{ij} = b_{ij} + \Delta b'_{ij} \quad (2.23)$$

$$b_{jk} = b_{jk} + \Delta b'_{jk} \quad (2.24)$$

Langkah 10. Jika kondisi iterasi belum selesai ($epoch \neq max\ epoch$ atau $error\ target < MSE$), maka lakukan langkah 1-9

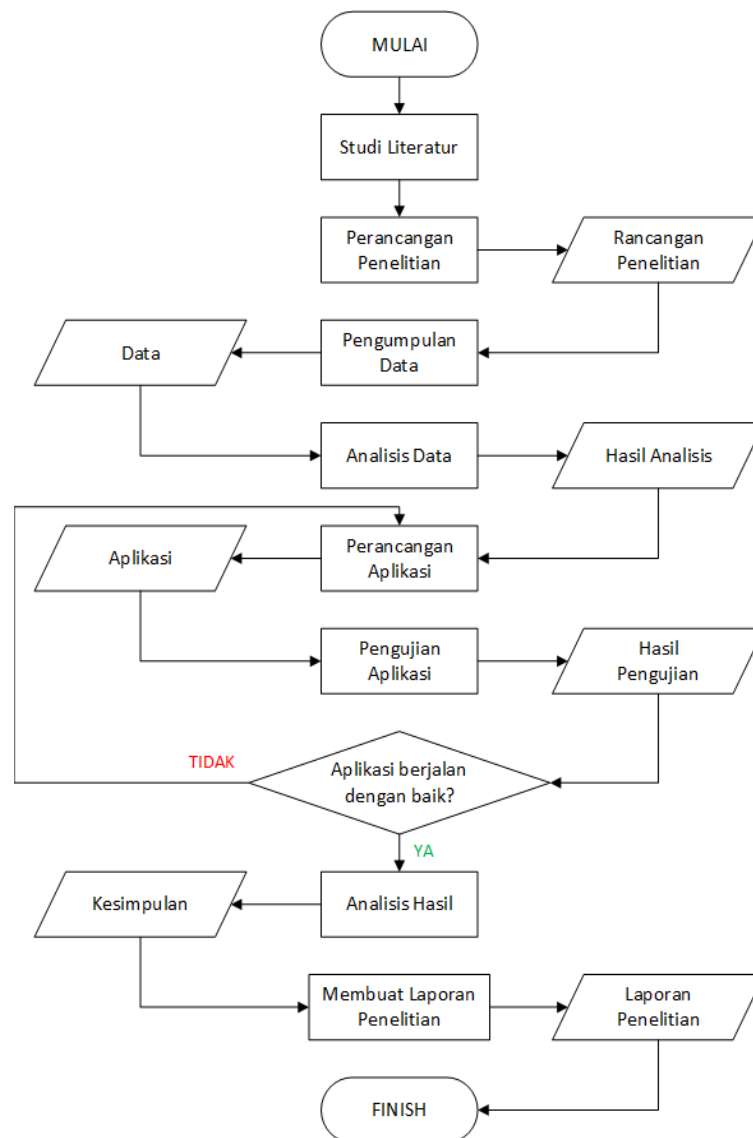
Setelah selesai, maka jaringan sudah dapat digunakan untuk pengenalan pola. Dalam hal ini, hanya *forward propagation* (langkah 4, 5, dan 6) yang dipakai untuk menentukan keluaran jaringan.

BAB III

DESAIN DAN IMPLEMENTASI

3.1 Prosedur Penelitian

Prosedur penelitian bertujuan agar penulis dapat melakukan penelitian dengan runtut dan memudahkan dalam mengambil langkah selanjutnya selama pengerjaan penelitian. Prosedur penelitian digambarkan berupa *flowchart* pada gambar 3.1 sebagai berikut.



Gambar 3.1 *Flowchart* prosedur penelitian

Prosedur penelitian ini menggunakan model pengembangan *Waterfall model*. *Waterfall model* adalah model yang menggunakan fase-fase berurutan dan sistematis dalam pengambilan langkah (Pressman & Lunak, 2012). Dalam penelitian ini, tahap pertama yang dilakukan adalah studi literatur, yakni mencari referensi dari materi-materi yang terkait dalam penelitian seperti curah hujan, pembagian musim di Indonesia, cara perhitungan curah hujan, hingga metode-metode peramalan yang terkait dengan curah hujan. Pencarian literatur ini penulis cari dari jurnal, buku, internet, dan video. yang sesuai dengan topik penelitian.

Proses selanjutnya adalah perancangan penelitian. Perancangan dilakukan dengan membuat segala persiapan apa saja yang hendak dilakukan sebelum memulai penelitian mulai dari sumber data, hingga instrumen yang terkait seperti PC dan program yang akan digunakan dalam penelitian. Pengumpulan data dilakukan dengan mengambil data curah hujan dari website resmi BMKG (*DATA ONLINE - PUSAT DATABASE - BMKG*, n.d.) yang ada di wilayah kabupaten Malang. Lalu dilakukanlah analisis dengan cara memilah dan mengelompokkan data agar dapat dimasukkan dalam program yang akan dirancang.

Selanjutnya perancangan aplikasi dikerjakan pada laptop ACER Aspire E1-470 dengan spesifikasi: Intel Core i3 (i3 - 3217U, 1.80 GHz, 3 MB Cache), Intel HD 4000 *Graphic Video Card*, 8 GB DDR3 SDRAM. Menggunakan *software* MATLAB 2019a untuk pembangunan aplikasi. Setelah aplikasi sudah selesai, selanjutnya melakukan pengujian aplikasi secara bertahap dengan memasukkan data hasil analisis ke dalam aplikasi, dan apakah aplikasi berjalan dengan benar atau tidak. Jika belum maka perlu dilakukannya perbaikan atau perancangan ulang, dan jika sudah berjalan lancar bisa langsung ke tahap selanjutnya.

Setelah data selesai diolah menggunakan aplikasi, maka hasil akan muncul. Selanjutnya mengubah beberapa parameter untuk membandingkan hasil yang terbaik, hingga bisa ditarik sebuah kesimpulan. Selanjutnya hasil dari kesimpulan ditulis dalam laporan beserta keterangannya.

3.2 Data yang Digunakan

Data yang digunakan dalam penelitian ini adalah data aktivitas curah hujan yang ada di kabupaten Malang. Pengumpulan data dilakukan dengan mengambil data curah hujan dari website resmi BMKG (<https://bmkg.go.id/>), dengan pengambilan data yang berada di lokasi Kabupaten Malang. Data dihimpun oleh Stasiun Klimatologi Malang yang beralamatkan di Genengan, Girimoyo, Kec. Karangploso, Kab. Malang. Rentang historis data yang diambil adalah dari tanggal 1 Januari 2000 hingga 31 Desember 2020.

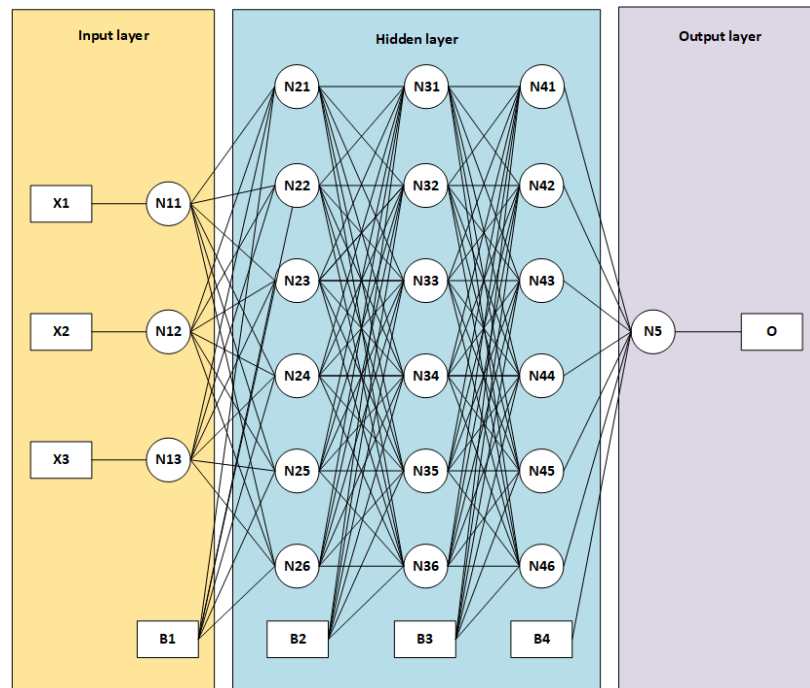
Setelah itu, data akan dilakukan perhitungan rata-rata per-bulan agar dapat mencakup rentang waktu yang relatif panjang dengan jumlah data yang tidak begitu besar. Contoh data curah hujan dapat dilihat pada tabel 3.1.

Tabel 3.1 Contoh data curah hujan (mm) dari tahun 2010 hingga 2016 (Januari hingga Juli)

	2010	2011	2012	2013	2014	2015	2016
JAN	11,322	4,484	10,214	11,8	12,332	10,336	7,425
FEB	7,786	6,5	15,071	7,625	6,4	15,652	24,944
MAR	6,742	10,935	6,903	9,258	5,874	8,562	7,512
APR	17,633	5,333	2,68	7,27	9,797	10,245	3,216
MEI	11	7,452	0,828	0,781	1,297	3,848	6,663
JUN	1	0,1667	0,592	0,592	1,47	1,473	5,636
JUL	2,968	0,064	0	0	0,290	0	3,872

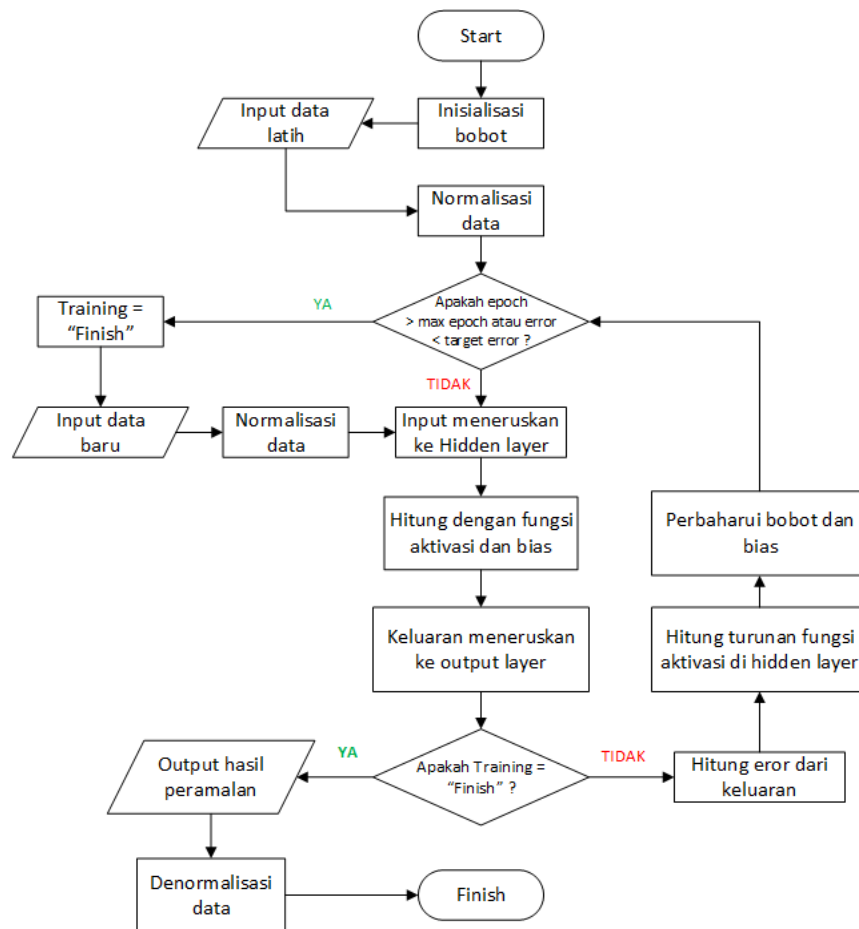
3.3 Desain Sistem

Desain sistem akan menyajikan proses dari pengolahan data, training, percobaan model, hingga perhitungan kesalahan/*error*. Dalam percobaan ini akan menggunakan 10 model ANN yang berbeda dengan beberapa parameter yang diubah, seperti jumlah *hidden layer*, jumlah neuron, jumlah input, *learning rate*, dan jumlah *epoch*. Contoh arsitektur ANN yang akan dilakukan dalam penelitian ini disajikan pada gambar 3.2



Gambar 3.2 Contoh arsitektur ANN dengan 3 *hidden layer*, dan 6 neuron

Flowchart jalannya peramalan dalam *Artificial Neural Network* adalah seperti dalam gambar 3.3 sebagai berikut.



Gambar 3.3 Flowchart proses peramalan dengan ANN

3.4 Perhitungan Manual

Disini akan dijelaskan proses perhitungan manual yang dilakukan oleh program hingga dapat menghasilkan sebuah peramalan.

Langkah 0. Inisialisasi.

Disajikan data seperti tabel 3.1, sehingga *input* 1 = Januari 2000, *input* 2 = Februari 2000, *input* 3 = Maret 2000, dan *output* = April 2000. *Input* disimbolkan (I), *output* (O), *weight* (W), *bias* (B), dan *learning rate* (α). Jumlah

hidden layer adalah 3x1 (3 neuron, 1 hidden layer), *input* dan *output* telah dinormalisasi.

$$I = [0.4513, 0.3577, 0.4182]$$

$$O = 0.1521$$

$$W_1 = [0.1, -0.3, 0.7 ; -0.9, 0.3, 0.2 ; -0.2, -0.5, 0.4]$$

$$W_2 = [-0.4, 0.2, -0.3]$$

$$B_1 = [0.1, -0.8, -0.6]$$

$$B_2 = -0.1$$

$$\alpha = 0.1$$

Langkah 1. Proses Forward Propagation

$$Z_{W_1} = f(Z_{net W_1}) = \frac{1}{1+e^{-Z_{net W_1}}} \text{ (rumus 2.8)}$$

$$Z_{W_{11}} = \frac{1}{1+\exp((-(0.1) * 0.4513)+(-(-0.3)*0.3577)+(-(0.7)*0.4132)-0.1)} = 0.5819$$

$$Z_{W_{12}} = \frac{1}{1+\exp((-(0.9) * 0.4513)+(-0.3) * 0.3577)+(-(0.2) * 0.4132)-(-0.8)} = 0.2660$$

$$Z_{W_{13}} = \frac{1}{1+\exp((-0.2 * 0.4513)+(-0.5 * 0.3577)+(0.4 * 0.4132)-(-0.6)} = 0.3314$$

$$Z_{W_1} = [0.5819, 0.2660, 0.3314]$$

$$O' = Y_{net W_2} = f(Y_{net W_2}) = \frac{1}{1+e^{-Y_{net W_2}}} \text{ (rumus 2.10)}$$

$$O' = \frac{1}{1+\exp((-(0.4) * 0.5819)+(-0.2) * 0.2660)+(-(-0.3) * 0.3314)-(-0.1))}$$

$$O' = 0.4064$$

Langkah 2. Proses Backpropagation

$$\delta_2 = (O - O') f'(y_{net W_2}) = (O - O') O' (1 - O') \text{ (rumus 2.11)}$$

$$\delta_2 = (0.1521 - 0.4064) * 0.4064 * (1 - 0.4064) = -0.0613$$

$$\delta_1 = f'(Z_{net W_1})(\delta_2 W_2) = Z_{W_1}(1 - Z_{W_1})(\delta_2 W_2) \text{ (rumus 2.12)}$$

$$\delta_{11} = 0.5819 * (1 - 0.5819) * (-0.0613 * -0.4) = 0.0060$$

$$\delta_{12} = 0.2660 * (1 - 0.2660) * (-0.0613 * 0.2) = -0.0024$$

$$\delta_{13} = 0.3314 * (1 - 0.3314) * (-0.0613 * -0.3) = 0.0041$$

$$\delta_1 = [0.0060, -0.0024, 0.0041]$$

$$W_{2\ new} = \alpha \delta_2 Z_{W_1} \text{ (rumus 2.13)}$$

$$W_{21\ new} = 0.1 * -0.0613 * 0.5819 = -0.0036$$

$$W_{22\ new} = 0.1 * -0.0613 * 0.2660 = -0.0016$$

$$W_{23\ new} = 0.1 * -0.0613 * 0.3314 = -0.0020$$

$$W_{2\ new} = [-0.0036, -0.0016, -0.0020]$$

$$B_{2\ new} = \alpha \delta_2 \text{ (rumus 2.15)}$$

$$B_{2\ new} = 0.1 * -0.0613 = -0.0061$$

$$W_{1\ new} = \alpha \delta_1 I \text{ (rumus 2.14)}$$

$$W_{111\ new} = 0.1 * 0.0060 * 0.4513 = 0.0002695$$

$$W_{112\ new} = 0.1 * 0.0060 * 0.3577 = 0.0002136$$

$$W_{113\ new} = 0.1 * 0.0060 * 0.4183 = 0.0002497$$

$$W_{121\ new} = 0.1 * -0.0024 * 0.4513 = -0.0001081$$

$$W_{122\ new} = 0.1 * -0.0024 * 0.3577 = -0.00008568$$

$$W_{123\ new} = 0.1 * -0.0024 * 0.4183 = -0.0001002$$

$$W_{131\ new} = 0.1 * 0.0041 * 0.4513 = 0.0001841$$

$$W_{132\ new} = 0.1 * 0.0041 * 0.3577 = 0.0001459$$

$$W_{133\ new} = 0.1 * 0.0041 * 0.4183 = 0.0001706$$

$$W_{1\ new} = [0.0002695, 0.0002136, 0.0002497;$$

$$-0.0001081, -0.00008568, -0.0001002;$$

$$0.0001841, 0.0001459, 0.0001706]$$

$$B_{l\ new} = \alpha \delta_l \text{ (rumus 2.16)}$$

$$B_{11\ new} = 0.1 * 0.0060 = 0.0006$$

$$B_{12\ new} = 0.1 * -0.0024 = -0.00024$$

$$B_{13\ new} = 0.1 * 0.0041 = 0.00041$$

$$B_{l\ new} = [0.0006, -0.00024, 0.00041]$$

Langkah 3. Lakukan/loop langkah 1 dan 2 sebanyak jumlah *input & output* data

Langkah 4. Simpan tiap hasil dari $W_{l\ new}$, $W_{2\ new}$, $B_{l\ new}$, $B_{2\ new}$ dari perulangan pertama hingga terakhir ke dalam array baru (array 3 dimensi).

Langkah 5. Jika *looping* selesai, jumlahkan total $W_{l\ new}$, $W_{2\ new}$, $B_{l\ new}$,

$B_{2\ new}$.

$$W_{l\ new} = \text{sum}(W_{l\ new}) \text{ (rumus 2.17)}$$

$$W_{2\ new} = \text{sum}(W_{2\ new}) \text{ (rumus 2.18)}$$

$$B_{l\ new} = \text{sum}(B_{l\ new}) \text{ (rumus 2.19)}$$

$$B_{2\ new} = \text{sum}(B_{2\ new}) \text{ (rumus 2.20)}$$

Langkah 6. Update *weight* dan bias

$$W_1 = W_1 + W_{l\ new} \text{ (rumus 2.21)}$$

$$W_2 = W_2 + W_{2\ new} \text{ (rumus 2.22)}$$

$$B_1 = B_1 + B_{l\ new} \text{ (rumus 2.23)}$$

$$B_2 = B_2 + B_{2\ new} \text{ (rumus 2.24)}$$

Langkah 7. Lakukan langkah 1 sampai 6 hingga MSE lebih kecil dari target, atau perulangan/*epoch* sudah mencapai jumlah maksimal yang sudah ditetapkan.

Setelah *weight* dan bias sudah terlatih/*supervised*, selanjutnya adalah melakukan *forward propagation* yang terdapat pada langkah 1 untuk melakukan peramalan menggunakan *input* baru.

3.5 Implementasi Program

Implementasi program merupakan tahap memasukkan rumus ke dalam program. Penulis menggunakan *software* MATLAB 2019a, yakni sebuah IDE (*Integrated Development Environment*) sekaligus bahasa pemrograman yang dikhususkan dalam komputasi numerik. MATLAB merupakan singkatan dari Laboratorium Matriks, karena memiliki kemampuan untuk menghitung dan memanipulasi matriks, pembuatan fungsi, implementasi algoritma, hingga pembuatan antar muka pengguna. Implementasi akan dipisah menjadi beberapa blok-blok program yang akan berjalan sesuai fungsinya masing-masing.

3.5.1 Implementasi Normalisasi Data

Pada proses normalisasi, data perlu disesuaikan menjadi skala tertentu. Untuk itu perlu mencari nilai MIN dan MAX dari semua data, yang selanjutnya akan digunakan untuk menormalisasi semua data tersebut menjadi skala 0 hingga 1. Berikut adalah kode program untuk proses normalisasi. Proses perhitungan normalisasi data ditunjukkan pada gambar 3.4.

```

1 function normalisasi=get_normalisasi(data)
2     max_val=max(data);
3     min_val=min(data);
4     [x,~]=size(data);
5     new_data=zeros(1,x);
6     for i=1:x
7         normalized=(data(i)-min_val)/(max_val-min_val);
8         new_data(i)=normalized;
9     end
10    normalisasi=new_data;
11 end

```

Gambar 3.4 Kode program implementasi normalisasi data

Penjelasan kode program pada gambar 3.4.

1. Baris 1 adalah deklarasi fungsi yang bernama 'get_normalisasi' dengan nilai yang didapat dari variabel normalisasi.

2. Baris 2 mencari nilai *max* dari variabel 'data'.
3. Baris 3 mencari nilai *min* dari variabel 'data'.
4. Baris 4 mencari ukuran data dengan menghitung banyaknya data asli.
5. baris 5 inisialisasi variabel 'new_data' untuk menyimpan hasil perhitungan normalisasi.
6. Baris 6 melakukan perulangan sebanyak jumlah data.
7. Baris 7 melakukan perhitungan normalisasi seperti rumus 2.4.
8. Baris 8 menyimpan hasil perhitungan kedalam variabel 'new_data'
9. Baris 10 mengembalikan nilai hasil ke dalam fungsi.

3.5.2 Implementasi Denormalisasi Data

Pada proses denormalisasi, data hasil normalisasi yang diubah ke skala 0 sampai 1 akan diubah kembali ke skala nilai aslinya. Perlu mencari nilai MIN dan MAX dari data asli sama seperti normalisasi. Proses perhitungan denormalisasi data ditunjukkan pada gambar 3.5.

```

1 function denormalisasi=get_denormalisasi(data,data_normal)
2     max_val=max(data);
3     min_val=min(data);
4     [x,~]=size(data);
5     new_data=zeros(1,x);
6     for i=1:x
7         denormal=(data_normal(i)*(max-min))+min;
8         new_data(i)=denormal;
9     end
10    denormalisasi=new_data;
11 end

```

Gambar 3.5 Kode program implementasi denormalisasi data

Penjelasan kode program pada gambar 3.5.

1. Baris 1 adalah deklarasi fungsi yang bernama 'get_denormalisasi' dengan membutuhkan nilai asli dan nilai hasil normalisasi.
2. Baris 2 mencari nilai max dari data asli.

3. Baris 3 mencari nilai min dari data asli.
4. Baris 4 mencari ukuran data dengan menghitung banyaknya data asli.
5. baris 5 inisialisasi variabel 'new_data' untuk menyimpan hasil perhitungan denormalisasi.
6. Baris 6 melakukan perulangan sebanyak jumlah data.
7. Baris 7 melakukan perhitungan denormalisasi seperti rumus 2.4
8. Baris 8 menyimpan hasil perhitungan kedalam variabel 'new_data'
9. Baris 10 mengembalikan nilai hasil ke dalam fungsi

3.5.3 Implementasi Perubahan Skala Data

Proses ini dilakukan ketika pada denormalisasi pertama pada peramalan menunjukkan hasil yang tidak maksimal, sehingga diperlukannya denormalisasi lagi. Proses ini akan merentangkan skala nilai lama menjadi nilai baru yang diinginkan. *Input* yang diperlukan meliputi data normal, data asli, data asli terkecil, data asli terbesar, target nilai terkecil, target nilai terbesar. Proses perhitungan perubahan skala data ditunjukkan pada gambar 3.6.

```

1 function hasil=get_ubah_skala(data_normal,data)
2     old_min=min(data_normal);
3     old_max=max(data_normal);
4     new_min=min(data);
5     new_max=max(data);
6     [x,~]=size(data_normal);
7     new_data=zeros(1,x);
8     for i=1:x
9         result=((data_normal(i)-old_min)/
10            (old_max-old_min))*(new_max-new_min)+new_min;
11         new_data(i)=result;
12     end
13     hasil=new_data;
14 end

```

Gambar 3.6 Kode program implementasi perubahan skala data

Penjelasan kode program pada gambar 3.6.

1. Baris 1 adalah deklarasi fungsi yang bernama 'get_ubah_skala' dengan membutuhkan nilai asli dan nilai hasil normalisasi.
2. Baris 2 mencari nilai min dari data normal.
3. Baris 3 mencari nilai max dari data normal.
4. Baris 4 mencari nilai min dari data asli.
5. Baris 5 mencari nilai max dari data asli.
6. Baris 6 mencari ukuran data dengan menghitung banyaknya data normal.
7. baris 7 inisialisasi variabel 'new_data' untuk menyimpan hasil perhitungan denormalisasi.
8. Baris 8 melakukan perulangan sebanyak jumlah data normal.
9. Baris 9 melakukan perhitungan pengubahan skala data seperti rumus 2.6
10. Baris 10 menyimpan hasil perhitungan kedalam variabel 'new_data'.
11. Baris 12 mengembalikan nilai hasil ke dalam fungsi

3.5.4 Implementasi MSE

Pada proses ini, akan dilakukan evaluasi peramalan dengan menghitung hasil MSE nya, yakni rata-rata selisih kuadrat antara nilai-nilai taksiran dengan nilai yang sebenarnya. MSE atau *Mean Squared Error* membutuhkan beberapa *input* seperti hasil *forecasting*, data target, dan total inputan data. Proses perhitungan MSE ditunjukkan pada gambar 3.7.

```

1 function mse = get_mse(target, forecasting, total_input)
2     mse = sum((target-forecasting).^2)/total_input;
3 end

```

Gambar 3.7 Kode program proses perhitungan MSE

Penjelasan kode program pada gambar 3.7.

1. Baris 1 adalah deklarasi fungsi yang bernama 'get_mse' dengan membutuhkan nilai target, hasil peramalan, dan total *input*.
2. Baris 3 melakukan perhitungan MSE seperti rumus 2.1

3.5.5 Implementasi WMAPE

Proses ini akan dihitung persentase *error* yang akan menghasilkan tingkat persentase kesalahan model untuk melakukan peramalan. WMAPE atau *Weighted Mean Absolute Percentage Error* membutuhkan beberapa *input* seperti hasil *forecasting*, dan data target. Proses perhitungan WMAPE ditunjukkan pada Kode Program 3.8.

```

1 function wmape = get_wmape(target, forecasting)
2     wmape = sum(abs(target-forecasting))/sum(abs(target))*100;
3 end

```

Gambar 3.8 Kode program proses perhitungan WMAPE

Penjelasan dari kode program pada gambar 3.8.

1. Baris 1 adalah deklarasi fungsi yang bernama 'get_wmape' dengan membutuhkan nilai target dan hasil peramalan.
2. Baris 3 melakukan perhitungan WMAPE seperti rumus 2.2

3.5.6 Implementasi Sigmoid Biner

Sigmoid biner atau fungsi aktivasi dilakukan ketika proses *Forward Propagation*. Fungsi aktivasi membutuhkan *input* dari *weight* dan nilai *net* dari

layer sebelumnya. Proses perhitungan sigmoid biner ditunjukkan pada gambar 3.9.

1	function sigmoid = get_sigmoid(weight, net)
2	sigmoid = 1 ./ (1 + exp(-weight * net));
3	end

Gambar 3.9 Kode program implementasi sigmoid biner

Penjelasan dari kode program pada gambar 3.9.

1. Baris 1 adalah deklarasi fungsi yang bernama 'get_sigmoid' dengan membutuhkan nilai *weight* dan *net* layer sebelumnya
2. Baris 2 adalah perhitungan Sigmoid Biner seperti rumus 2.3. Tanda './' adalah untuk pembagian matriks.

3.5.7 Implementasi Input, Bias, dan Weight

Proses ini mengimplementasikan inisialisasi nilai awal pada program. Dimulai dari memasukkan data kedalam sebuah matriks 252x1 dan memisahkannya menjadi masing-masing 3 *input* dan 1 *output*, menambahkan bias pada tiap neuron, weight untuk menghubungkan neuron di tiap layer, *max iteration*, dll. Implementasi berikut ditunjukkan pada gambar 3.10.

```

1 load('data_semua.mat');
2 data_normal=get_normalisasi(data);
3
4 for i=1:length(data)-3
5     input1(i)=data_normal(1,i);
6     input2(i)=data_normal(1,i+1);
7     input3(i)=data_normal(1,i+2);
8     output(i)=data_normal(1,i+3);
9 end
10
11 input_all=[input1' input2' input3'];
12 output=output';
13 [n_input_data, n_input] = size(input_all);
14 n_output = size(output,2);
15 input_all = [ones(n_input_data,1), input_all];
16
17 learning_rate = 0.1;
18 max_iteration = 100;
19 hidden_neuron = 5;
20
21 w1 = 2 * rand(hidden_neuron, n_input + 1) - 1;
22 w2 = 2 * rand(n_output, hidden_neuron + 1) - 1;
23
24 w1_new = zeros(hidden_neuron, n_input + 1, n_input_data);
25 w2_new = zeros(n_output, hidden_neuron + 1, n_input_data);

```

Gambar 3.10 Kode program implementasi input, bias, dan weight

Penjelasan dari kode program pada gambar 3.10.

1. Baris 1 memanggil file data yang sudah dalam bentuk format (.mat).
File ini berisi tabel 252x1
2. Baris 2 adalah melakukan normalisasi data dengan memasukkan ke dalam fungsi 'get_normalisasi' seperti pada kode program 3.1
3. Baris 4 sampai 9 adalah memecah data ke dalam variabel *input* dan *output* yang berjumlah 3 *input* dan 1 *output*.
4. Baris 11 menggabungkan semua variabel *input* ke dalam satu variabel bernama 'input_all' dalam posisi transpose.
5. Baris 12 adalah mentranspose variabel *output*.
6. Baris 13 adalah menghitung dimensi matriks 'input all' untuk menyimpan berapa jumlah data *input* yang tersedia ke dalam 'n_input_data', dan jumlah angka yang akan menjadi *input* ke dalam 'n_input'.

7. Baris 14 adalah untuk menghitung jumlah output, dan disimpan ke 'n_output'.
8. Baris 15 adalah proses menyisipkan bias ke dalam matriks *input* yang memiliki nilai 1
9. Baris 17 adalah mendeklarasikan nilai *learning rate* untuk *neural network* pada program.
10. Baris 18 adalah mendeklarasi nilai iterasi maksimum untuk menghitung jaringan paling optimal yang akan digunakan sebagai peramalan.
11. Baris 19 adalah mendeklarasi nilai jumlah neuron pada tiap *hidden layer*.
12. Baris 21 sampai 22 adalah mendeklarasi *weight* bernilai acak dengan rentang angka -1 sampai dengan 1.
13. Baris 24 sampai 25 adalah deklarasi matrik kosong untuk menampung *weight* 1 dan 2 setelah perhitungan.

3.5.8 Implementasi Forward Propagation dan Backpropagation

Forward Propagation atau propagasi maju berfungsi untuk mengubah *input* ke dalam *output* yang diinginkan melalui perhitungan antara *weight*, neuron, dan fungsi aktivasi. *Forward Propagation* akan dilakukan ke semua *input*, dan nilai *output* yang keluar akan dilanjutkan ke dalam proses *Backpropagation*. Dalam proses *Backpropagation*, nilai *output* sebelumnya akan dihitung turunannya, sehingga akan diketahui nilai error yang terjadi dalam peramalan. Nilai tersebut akan digunakan sebagai evaluasi melalui pembaharuan bobot secara berkala hingga mencapai iterasi maksimumnya.

Implementasi *Forward Propagation* dan *Backpropagation* ditunjukkan pada gambar 3.11.

```

1 for iteration = 1 : max_iteration
2   for i = 1:n_input_data
3     forward_prob_1(i,:) = get_sigmoid(w1,input_all(i,:));
4     forward_prob_out(i,:) = get_sigmoid(w2,[1;
5       forward_prob_1(i,:)]);
6
7     delta2(i,:) = (output(i,:) - forward_prob_out(i,:)) .*
8       forward_prob_out(i,:) .* (1 - forward_prob_out(i,:));
9     delta1(i,:) = forward_prob_1(i,:) .* (1 -
10      forward_prob_1(i,:)) .* (delta2(i,:) * w2(:,2:end));
11    w2_new(:, :, i) = learning_rate * delta2(i,:) * [1,
12      forward_prob_1(i,:)]';
13    w1_new(:, :, i) = learning_rate * delta1(i,:) *
14      input_all(i,:);
15  end
16
17  w2_new = sum(w2_new,3);
18  w1_new = sum(w1_new,3);
19
20  w2 = w2 + w2_new;
21  w1 = w1 + w1_new;
22 end

```

Gambar 3.11 Kode program implementasi forward propagation dan backpropagation

Penjelasan dari kode program pada gambar 3.11.

1. Baris 1 adalah melakukan iterasi sebanyak variabel 'max_iteration'
2. Baris 2 adalah iterasi kembali (*nested loop*) sebanyak data input.
3. Baris 3 adalah melakukan perhitungan *forward propagation* antara layer *input* ke hidden layer 1 dengan fungsi 'get_sigmoid' seperti pada Kode Program 3.5
4. Baris 4 adalah melakukan perhitungan *Forward Propagation* antara *hidden layer 1* ke layer *output*
5. Baris 6 adalah menghitung *Backpropagation* dari *output* ke *hidden layer 1* berdasarkan rumus 2.11
6. Baris 7 adalah menghitung *Backpropagation* dari *hidden layer 1* ke *input layer* berdasarkan rumus 2.12

7. Baris 8 adalah menghitung *weight* antara *hidden layer* ke *output* menggunakan rumus 2.13 dengan perkalian bias menggunakan rumus 2.15
8. Baris 9 adalah menghitung *weight* antara *input* ke *hidden layer* menggunakan rumus 2.14 dengan perkalian bias menggunakan rumus 2.16
9. Baris 12 sampai 13 adalah menghitung total *weight* yang ditampung dalam 3 dimensional matriks menjadi 2 dimensional menggunakan rumus 2.17, atau 2.18, juga layer bias dengan rumus 2.19, atau 2.20.
10. Baris 15 sampai 16 adalah memperbaharui nilai *weight* dengan menjumlahkan nilai *weight* lama dengan nilai *weight* baru menggunakan rumus 2.21, atau 2.22, juga pada bias menggunakan rumus 2.23, atau 2.24.

3.6 Skenario Uji Coba

Pada tahap ini dilakukan skenario uji coba pada model *Artificial Neural Network* untuk melakukan peramalan curah hujan di kabupaten Malang. Proses pengujian ini bertujuan untuk mendapatkan model ANN terbaik untuk melakukan peramalan. Perhitungan yang dilakukan akan sama seperti rumus perhitungan yang terdapat pada bab sebelumnya. Sebagai contoh, pelatihan pertama akan menggunakan data bulan Januari, Februari, Maret 2000, dan *output*-nya adalah bulan April 2000. Lalu untuk data latih selanjutnya akan bergeser satu bulan kedepan sehingga input yang digunakan adalah bulan Februari, Maret, April 2000, dan *output*-nya adalah Mei 2000. Begitu seterusnya hingga mencapai Desember 2020.

Selanjutnya, pengujian akan dilakukan pada 10 model ANN yang berbeda dengan beberapa parameter yang diubah seperti jumlah neuron, *hidden layer*, jumlah *input*, *learning rate*, dan jumlah perulangan/*epoch*. Tiap parameter akan diambil nilainya secara acak, dan akan menghasilkan nilai MSE, WMAPE, dan nilai akurasi peramalannya. Dari 10 model tersebut akan dapat disimpulkan model yang memiliki hasil peramalan yang bagus dan yang buruk dilihat dari nilai akurasi peramalannya.

BAB IV

UJI COBA DAN PEMBAHASAN

Pada bab ini akan menjelaskan dan membahas tentang hasil dari implementasi algoritma *Artificial Neural Network Backpropagation* untuk peramalan curah hujan di kabupaten Malang. Hasil pengujian berupa model *Neural Network* terbaik yang disajikan dalam bentuk tabel dan plot data, dengan *error* yang akan dihitung menggunakan *MSE*, *WMAPE*, dan besar akurasi peramalannya.

4.1 Pengujian Model

Pengujian ini bertujuan untuk menemukan model yang paling optimal untuk peramalan curah hujan. Proses pengujian ini akan menggunakan 10 model yang berbeda seperti yang ditampilkan dalam tabel 4.1 berikut:

Tabel 4.1 Daftar pengujian model ANN

No	Nama Model	Hidden Layer	Neuron	Input	Learning Rate	Epoch
1	Model A	1	3	4	0,5	50
2	Model B	1	5	3	0,1	100
3	Model C	2	5	5	0,05	300
4	Model D	2	8	6	0,2	100
5	Model E	3	1	3	0,08	200
6	Model F	3	2	7	0,04	500
7	Model G	4	4	5	0,3	400
8	Model H	4	6	7	0,4	300
9	Model I	5	4	3	0,1	200
10	Model J	5	5	4	0,06	50

Data latih yang akan digunakan adalah tahun 2000 hingga 2010, sedangkan untuk data ujinya menggunakan data curah hujan tahun 2011 hingga 2020. Percobaan akan dilakukan sesuai urutan yang akan dihitung nilai MSE dan nilai WMAPE-nya.

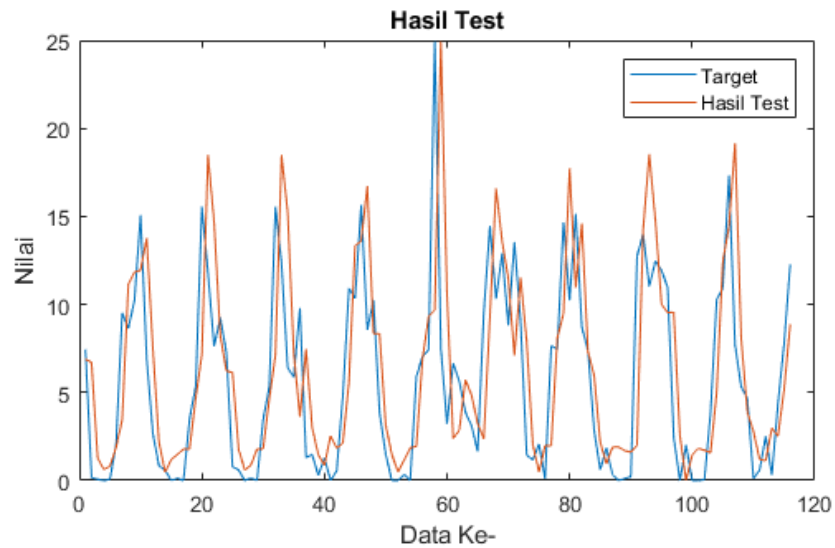
4.1.1 Pengujian Model A

Pengujian model A ini akan menggunakan parameter 1 *hidden layer*, 3 neuron, 4 *input*, 0,5 *learning rate*, dan 50 *epoch*. Hasil pengujian data ke-1 hingga ke-115 dapat dilihat pada tabel 4.2.

Tabel 4.2 Hasil pengujian peramalan model A

Data ke-	Data Asli	Hasil Peramalan	Error
1	7,4516	6,7045	0,7471
2	0,1667	1,2789	-1,1123
3	0,0645	0,6203	-0,5557
4	0	0,8126	-0,8126
5	0,0667	1,8861	-1,8194
6	2,0323	3,4335	-1,4012
7	9,5172	11,1475	-1,6302
8	8,6452	11,844	-3,1988
...
115	7,77	8,8781	-1,1081

Dapat dilihat bahwa *error* merupakan selisih antara data asli dengan hasil peramalan. Apabila hasil tersebut disajikan dalam bentuk plot data, maka dapat dilihat pada gambar 4.1, dengan grafik berwarna biru merupakan grafik target/data asli, dan grafik merah merupakan grafik hasil tes/percobaan.



Gambar 4.1 Plot data pengujian peramalan model A

Percobaan pada model ini menghasilkan nilai MSE sebesar 2.969 dan WMAPE sebesar 25.4792%, diperoleh dari perhitungan,

$$MSE = \frac{1}{115} (0,7471^2 + (-1,1123)^2 + \dots + (-1,1081)^2) = 2,969$$

$$WMAPE = \frac{|0,7471| + |-1,1123| + \dots + |-1,1081|}{|7,4516| + |0,1667| + \dots + |7,77|} \times 100 = 25.4792\%$$

Apabila ditinjau pada tabel 2.3, hasil peramalan tersebut masuk dalam kategori hasil peramalan sedang.

4.1.2 Pengujian Model B

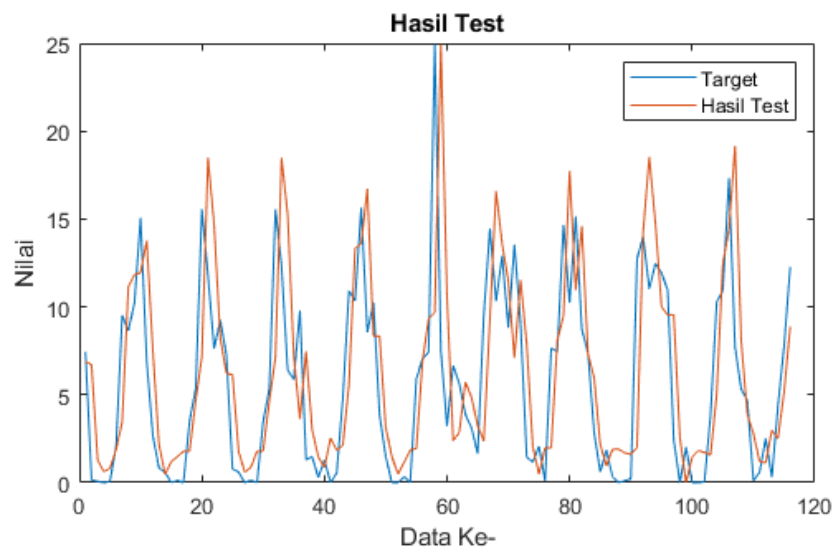
Pengujian model B ini menggunakan parameter 1 *hidden layer*, 5 neuron, 3 *input*, 0,1 *learning rate*, dan 100 *epoch*. Hasil pengujian data ke-1 hingga ke-116 dapat dilihat pada tabel 4.3.

Tabel 4.3 Hasil pengujian peramalan model B

Data ke-	Data Asli	Hasil Peramalan	Error
1	5,3333	4,9955	0,3378
2	7,4516	7,4682	-0,0166
3	0,1667	0,4606	-0,2939

4	0,0645	0,4981	-0,4336
5	0	0,0097	-0,0097
6	0,0667	0,054	0,0127
7	2,0323	1,6051	0,4271
8	9,5172	8,6593	0,8579
...
116	7,77	6,9022	0,8678

Error merupakan selisih antara data asli dengan hasil peramalan. Apabila hasil tersebut disajikan dalam bentuk plot data, maka dapat dilihat pada gambar 4.2, dengan grafik berwarna biru merupakan grafik target/data asli, dan grafik merah merupakan grafik hasil tes/percobaan.



Gambar 4.2 Plot data pengujian peramalan model B

Percobaan pada model ini menghasilkan nilai MSE sebesar 0.17056 dan WMAPE sebesar 5.6036%, diperoleh dari perhitungan,

$$MSE = \frac{1}{116} (0,3378^2 + (-0,0166)^2 + \dots + (-0,2939)^2) = 0.17056$$

$$WMAPE = \frac{|0,3378| + |-0,0166| + \dots + |-0,2939|}{|5,3333| + |7,4516| + \dots + |7,77|} \times 100 = 5.6036\%$$

Apabila ditinjau pada tabel 2.3, hasil peramalan tersebut masuk dalam kategori hasil peramalan sangat baik.

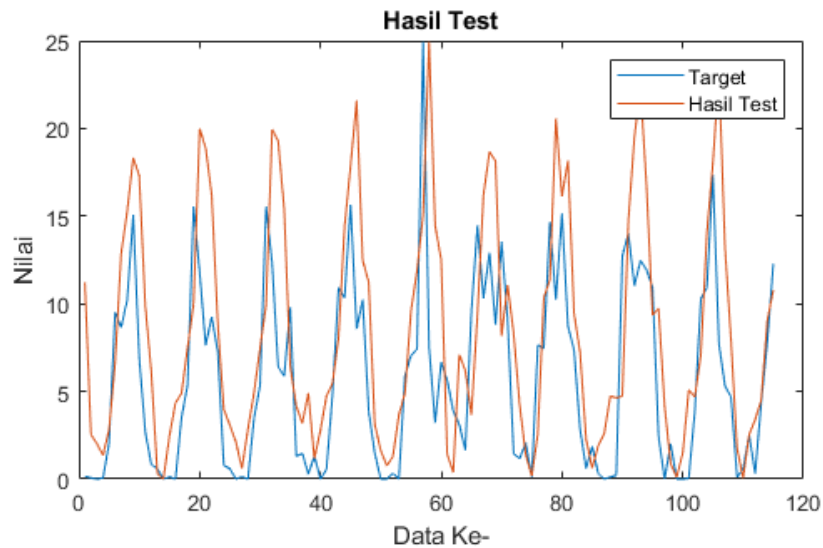
4.1.3 Pengujian Model C

Pengujian model C ini akan menggunakan parameter *2 hidden layer*, *5 neuron*, *5 input*, *0,05 learning rate*, dan *100 epoch*. Hasil pengujian data ke-1 hingga ke-114 dapat dilihat pada tabel 4.3.

Tabel 4.4 Hasil pengujian peramalan model C

Data ke-	Data Asli	Hasil Peramalan	Error
1	0,1667	2,5361	-2,3694
2	0,0645	2,013	-1,9485
3	0	1,347	-1,347
4	0,0667	2,8326	-2,7659
5	2,0323	6,6173	-4,5851
6	9,5172	12,8559	-3,3386
7	8,6452	15,2635	-6,6183
8	10,2143	18,3249	-8,1106
...
114	7,77	10,778	-3,008

Error merupakan selisih antara data asli dengan hasil peramalan. Apabila hasil tersebut disajikan dalam bentuk plot data, maka dapat dilihat pada gambar 4.3, dengan grafik berwarna biru merupakan grafik target/data asli, dan grafik merah merupakan grafik hasil tes/percobaan.



Gambar 4.3 Plot data pengujian peramalan model C

Percobaan pada model ini menghasilkan nilai MSE sebesar 17.3474 dan WMAPE sebesar 61.1503%, diperoleh dari perhitungan,

$$MSE = \frac{1}{114} ((-2,3694)^2 + (-1,9485)^2 + \dots + (-3,008)^2) = 17.3474$$

$$WMAPE = \frac{|-2,3694| + |-1,9485| + \dots + |-3,008|}{|0,1667| + |0,0645| + \dots + |7,77|} \times 100 = 61.1503\%$$

Apabila ditinjau pada tabel 2.3, hasil peramalan tersebut masuk dalam kategori hasil peramalan buruk.

4.1.4 Pengujian Model D

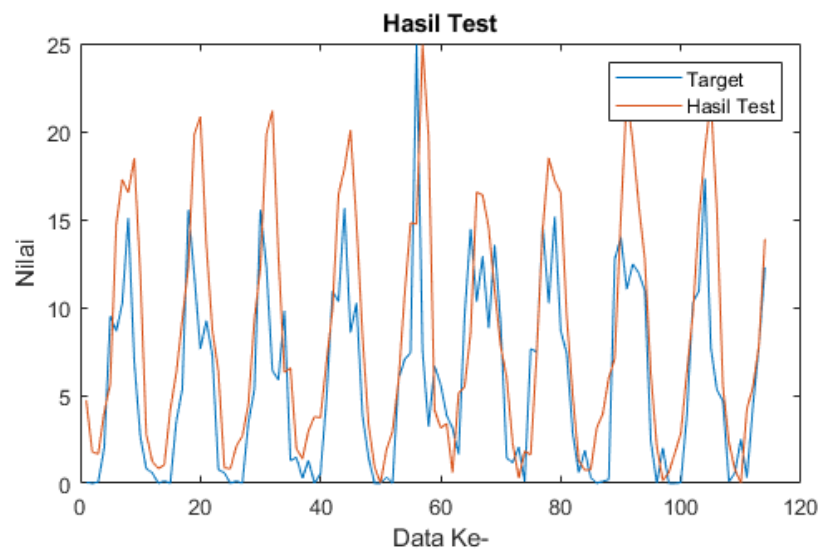
Pengujian model D ini menggunakan parameter 2 *hidden layer*, 8 neuron, 6 *input*, 0,2 *learning rate*, dan 400 *epoch*. Hasil pengujian data ke-1 hingga ke-113 dapat dilihat pada tabel 4.4.

Tabel 4.5 Hasil pengujian peramalan model D

Data ke-	Data Asli	Hasil Peramalan	Error
1	0,0645	1,7886	-1,7241
2	0	1,669	-1,669
3	0,0667	4,0517	-3,985

4	2,0323	5,6452	-3,613
5	9,5172	14,704	-5,1868
6	8,6452	17,2602	-8,615
7	10,2143	16,5244	-6,3101
8	15,0714	18,4716	-3,4002
...
113	7,77	13,8858	-6,1158

Error merupakan selisih antara data asli dengan hasil peramalan. Apabila hasil tersebut disajikan dalam bentuk plot data, maka dapat dilihat pada gambar 4.4, dengan grafik berwarna biru merupakan grafik target/data asli, dan grafik merah merupakan grafik hasil tes/percobaan.



Gambar 4.4 Plot data pengujian peramalan model D

Percobaan pada model ini menghasilkan nilai MSE sebesar 19.5839 dan WMAPE sebesar 62.3381%, diperoleh dari perhitungan,

$$MSE = \frac{1}{113} ((-1,7241)^2 + (-1,669)^2 + \dots + (-6,1158)^2) = 19.5839$$

$$WMAPE = \frac{|-1,7241| + |-1,669| + \dots + |-6,1158|}{|0,0645| + |0| + \dots + |7,77|} \times 100 = 62.3381\%$$

Apabila ditinjau pada tabel 2.3, hasil peramalan tersebut masuk dalam kategori hasil peramalan buruk.

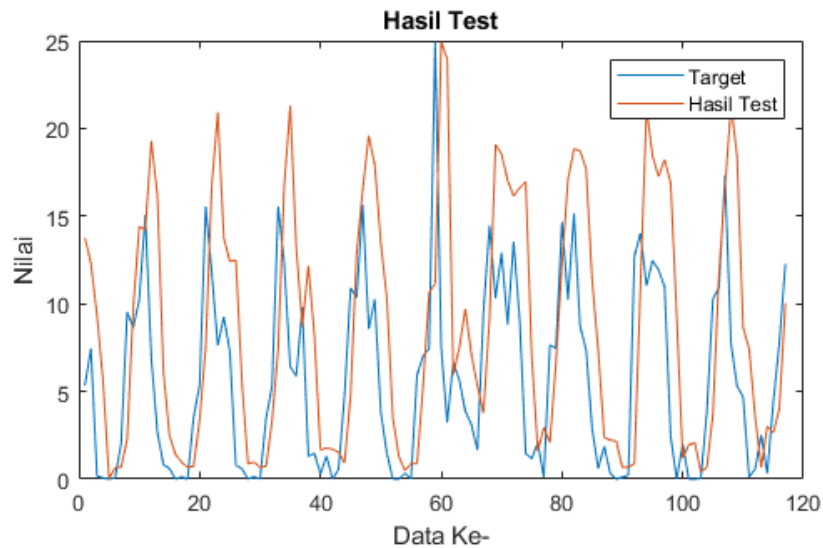
4.1.5 Pengujian Model E

Pengujian model E ini menggunakan parameter 3 *hidden layer*, 1 neuron, 3 *input*, 0,08 *learning rate*, dan 200 *epoch*. Hasil pengujian data ke-1 hingga ke-116 dapat dilihat pada tabel 4.5.

Tabel 4.6 Hasil pengujian peramalan model E

Data ke-	Data Asli	Hasil Peramalan	Error
1	5,3333	12,2937	-6,9604
2	7,4516	9,3257	-1,8741
3	0,1667	5,6237	-5,4571
4	0,0645	0	0,0645
5	0	0,6582	-0,6582
6	0,0667	0,6749	-0,6082
7	2,0323	2,273	-0,2407
8	9,5172	9,757	-0,2398
...
116	7,77	10,0498	-2,2798

Error merupakan selisih antara data asli dengan hasil peramalan. Apabila hasil tersebut disajikan dalam bentuk plot data, maka dapat dilihat pada gambar 4.5, dengan grafik berwarna biru merupakan grafik target/data asli, dan grafik merah merupakan grafik hasil tes/percobaan.



Gambar 4.5 Plot data pengujian peramalan model E

Percobaan pada model ini menghasilkan nilai MSE sebesar 18.6456 dan WMAPE sebesar 55.6322%, diperoleh dari perhitungan,

$$MSE = \frac{1}{116} ((-6,9604)^2 + (-1,8741)^2 + \dots + (-2,2798)^2) = 18.6456$$

$$WMAPE = \frac{|-6,9604| + |-1,8741| + \dots + |-2,2798|}{|5,3333| + |7,4516| + \dots + |7,77|} \times 100 = 55.6322\%$$

Apabila ditinjau pada tabel 2.3, hasil peramalan tersebut masuk dalam kategori hasil peramalan buruk.

4.1.6 Pengujian Model F

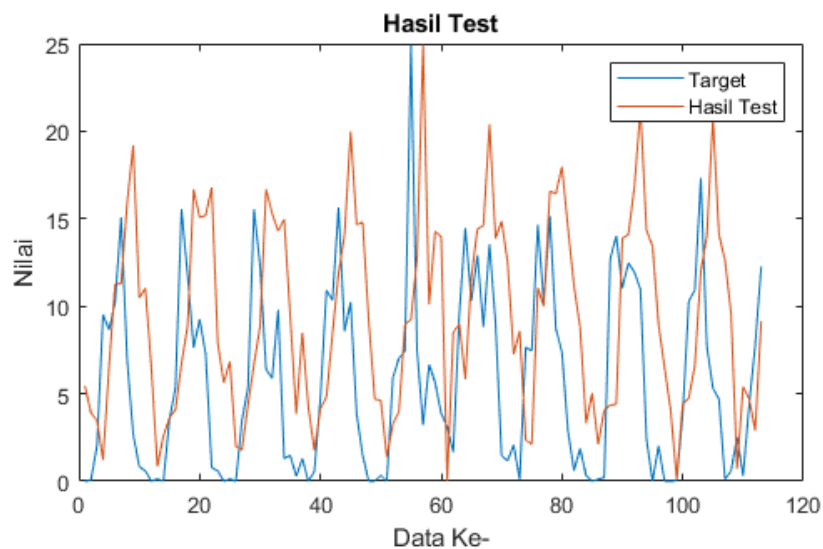
Pengujian model F ini menggunakan parameter 3 *hidden layer*, 2 neuron, 7 *input*, 0,04 *learning rate*, dan 500 *epoch*. Hasil pengujian data ke-1 hingga ke-112 dapat dilihat pada tabel 4.6.

Tabel 4.7 Hasil pengujian peramalan model F

Data ke-	Data Asli	Hasil Peramalan	Error
1	0	3,9003	-3,9003
2	0,0667	3,4695	-3,4028
3	2,0323	1,2161	0,8162

4	9,5172	6,5413	2,9759
5	8,6452	11,2596	-2,6145
6	10,2143	11,2851	-1,0708
7	15,0714	15,94	-0,8686
8	6,9032	19,1922	-12,2889
...
112	7,77	9,153	-1,383

Error merupakan selisih antara data asli dengan hasil peramalan. Apabila hasil tersebut disajikan dalam bentuk plot data, maka dapat dilihat pada gambar 4.6, dengan grafik berwarna biru merupakan grafik target/data asli, dan grafik merah merupakan grafik hasil tes/percobaan.



Gambar 4.6 Plot data pengujian peramalan model F

Percobaan pada model ini menghasilkan nilai MSE sebesar 36.6619 dan WMAPE sebesar 86.9375%, diperoleh dari perhitungan,

$$MSE = \frac{1}{112} ((-3,9003)^2 + (-3,4028)^2 + \dots + (-1,383)^2) = 36.6619$$

$$WMAPE = \frac{|-3,9003| + |-3,4028| + \dots + |-1,383|}{|0| + |0,0667| + \dots + |7,77|} \times 100 = 86.9375\%$$

Apabila ditinjau pada tabel 2.3, hasil peramalan tersebut masuk dalam kategori hasil peramalan buruk.

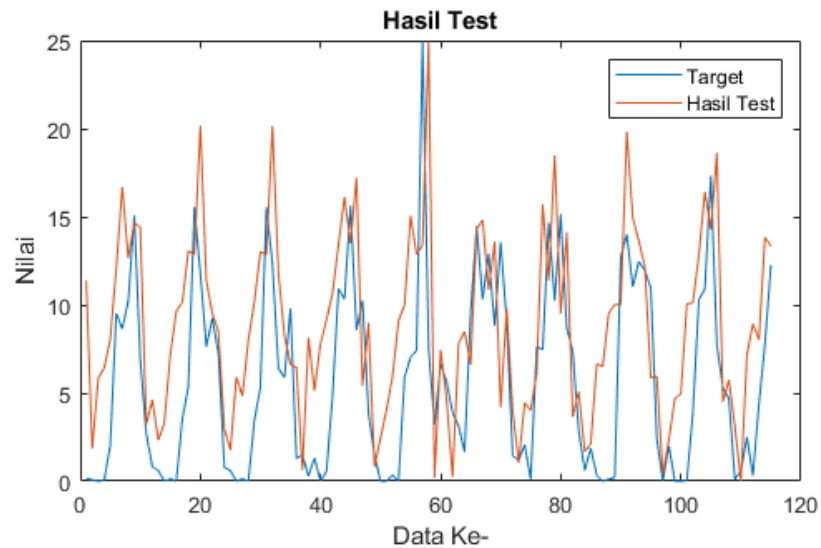
4.1.7 Pengujian Model G

Pengujian model G ini menggunakan parameter 4 *hidden layer*, 4 neuron, 5 *input*, 0,3 *learning rate*, dan 400 *epoch*. Hasil pengujian data ke-1 hingga ke-114 dapat dilihat pada tabel 4.7.

Tabel 4.8 Hasil pengujian peramalan model G

Data ke-	Data Asli	Hasil Peramalan	Error
1	0,1667	1,8473	-1,6807
2	0,0645	5,8313	-5,7668
3	0	6,4348	-6,4348
4	0,0667	8,0611	-7,9945
5	2,0323	12,0565	-10,0242
6	9,5172	16,6883	-7,171
7	8,6452	12,6613	-4,0161
8	10,2143	14,68	-4,4657
...
114	7,77	13,3333	-5,5633

Error merupakan selisih antara data asli dengan hasil peramalan. Apabila hasil tersebut disajikan dalam bentuk plot data, maka dapat dilihat pada gambar 4.7, dengan grafik berwarna biru merupakan grafik target/data asli, dan grafik merah merupakan grafik hasil tes/percobaan.



Gambar 4.7 Plot data pengujian peramalan model G

Percobaan pada model ini menghasilkan nilai MSE sebesar 31.3194 dan WMAPE sebesar 83.4231%, diperoleh dari perhitungan,

$$MSE = \frac{1}{114} ((-1,6807)^2 + (-5,7668)^2 + \dots + (-5,5633)^2) = 31.3194$$

$$WMAPE = \frac{|-1,6807| + |-5,7668| + \dots + |-5,5633|}{|0,1667| + |0,0645| + \dots + |7,77|} \times 100 = 83.4231\%$$

Apabila ditinjau pada tabel 2.3, hasil peramalan tersebut masuk dalam kategori hasil peramalan buruk.

4.1.8 Pengujian Model H

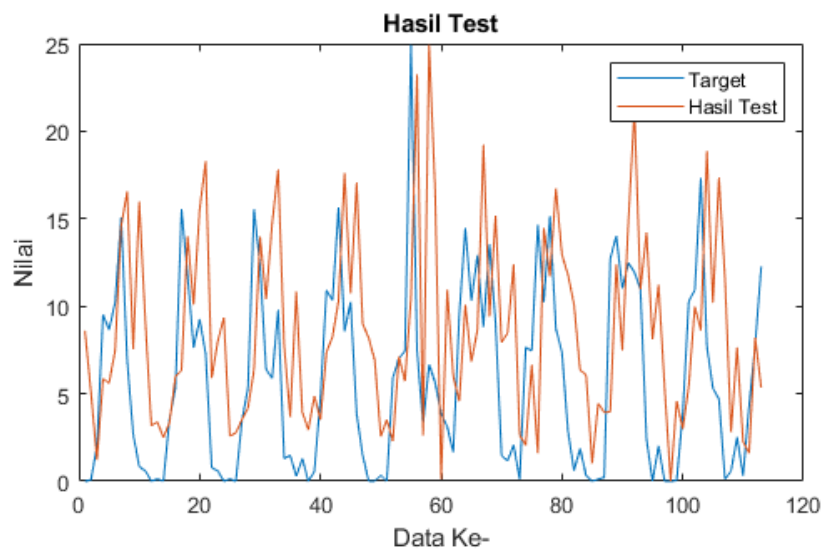
Pengujian model H ini menggunakan parameter 4 *hidden layer*, 6 neuron, 7 *input*, 0,4 *learning rate*, dan 300 *epoch*. Hasil pengujian data ke-1 hingga ke-112 dapat dilihat pada tabel 4.8.

Tabel 4.9 Hasil pengujian peramalan model H

Data ke-	Data Asli	Hasil Peramalan	Error
1	0	5,0925	-5,0925
2	0,0667	1,2194	-1,1527
3	2,0323	5,8589	-3,8267

4	9,5172	5,6009	3,9164
5	8,6452	7,4602	1,1849
6	10,2143	14,5735	-4,3592
7	15,0714	16,5495	-1,4781
8	6,9032	7,5552	-0,652
...
112	7,77	5,3403	2,4297

Error merupakan selisih antara data asli dengan hasil peramalan. Apabila hasil tersebut disajikan dalam bentuk plot data, maka dapat dilihat pada gambar 4.8, dengan grafik berwarna biru merupakan grafik target/data asli, dan grafik merah merupakan grafik hasil tes/percobaan.



Gambar 4.8 Plot data pengujian peramalan model H

Percobaan pada model ini menghasilkan nilai MSE sebesar 27.7248 dan WMAPE sebesar 70.0181%, diperoleh dari perhitungan,

$$MSE = \frac{1}{112} ((-5,0925)^2 + (-1,1527)^2 + \dots + 2,4297^2) = 27.7248$$

$$WMAPE = \frac{|-5,0925| + |-1,1527| + \dots + |2,4297|}{|0| + |0,0667| + \dots + |7,77|} \times 100 = 70.0181\%$$

Apabila ditinjau pada tabel 2.3, hasil peramalan tersebut masuk dalam kategori hasil peramalan buruk.

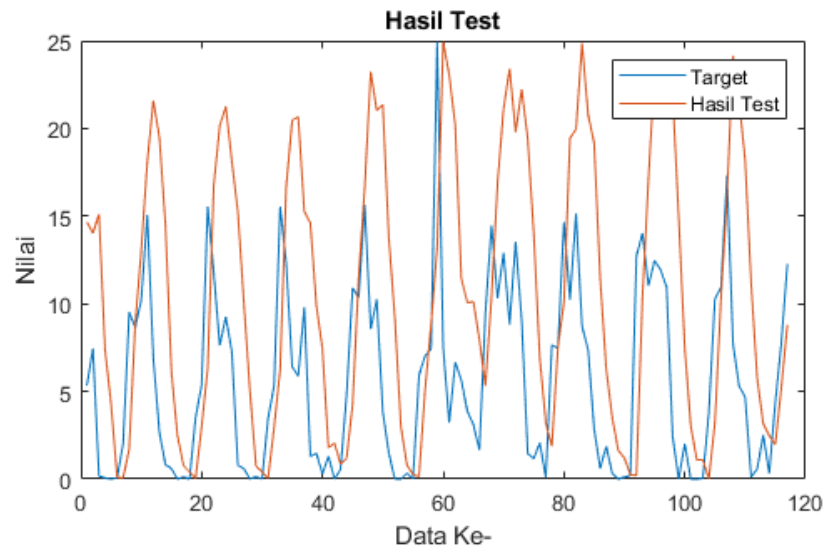
4.1.9 Pengujian Model I

Pengujian model I ini menggunakan parameter 5 *hidden layer*, 4 neuron, 3 *input*, 0,1 *learning rate*, dan 200 *epoch*. Hasil pengujian data ke-1 hingga ke-116 dapat dilihat pada tabel 4.9.

Tabel 4.10 Hasil pengujian peramalan model I

Data ke-	Data Asli	Hasil Peramalan	Error
1	5,3333	14,027	-8,6937
2	7,4516	15,1077	-7,6561
3	0,1667	7,3772	-7,2106
4	0,0645	4,3029	-4,2384
5	0	0,0954	-0,0954
6	0,0667	0,0578	0,0089
7	2,0323	1,7193	0,313
8	9,5172	8,8202	0,6971
...
116	7,77	8,8052	-1,0352

Error merupakan selisih antara data asli dengan hasil peramalan. Apabila hasil tersebut disajikan dalam bentuk plot data, maka dapat dilihat pada gambar 4.9, dengan grafik berwarna biru merupakan grafik target/data asli, dan grafik merah merupakan grafik hasil tes/percobaan.



Gambar 4.9 Plot data pengujian peramalan model I

Percobaan pada model ini menghasilkan nilai MSE sebesar 47.2243 dan WMAPE sebesar 91.2198%, diperoleh dari perhitungan,

$$MSE = \frac{1}{115} ((-8,6937)^2 + (-7,6561)^2 + \dots + (-1,0352)^2) = 47.2243$$

$$WMAPE = \frac{|-8,6937| + |-7,6561| + \dots + |-1,0352|}{|5,3333| + |7,4516| + \dots + |7,77|} \times 100 = 91.2198\%$$

Apabila ditinjau pada tabel 2.3, hasil peramalan tersebut masuk dalam kategori hasil peramalan buruk.

4.1.10 Pengujian Model J

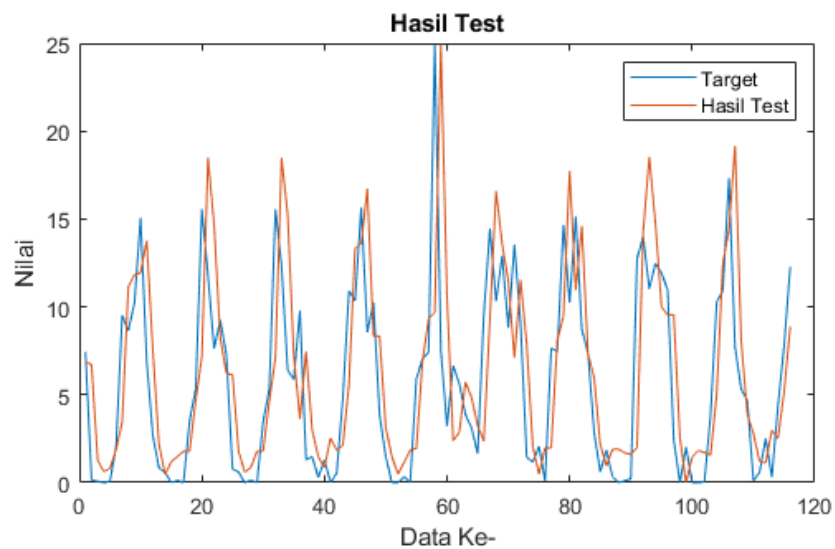
Pengujian model J ini menggunakan parameter 5 *hidden layer*, 5 neuron, 4 *input*, 0,06 *learning rate*, dan 50 *epoch*. Hasil pengujian data ke-1 hingga ke-115 dapat dilihat pada tabel 4.10.

Tabel 4.11 Hasil pengujian peramalan model J

Data ke-	Data Asli	Hasil Peramalan	Error
1	7,4516	5,4348	2,0168
2	0,1667	12,9142	-12,7476
3	0,0645	3,6874	-3,6229

4	0	5,7009	-5,7009
5	0,0667	3,0298	-2,9632
6	2,0323	2,4934	-0,4612
7	9,5172	1,9608	7,5565
8	8,6452	7,7039	0,9413
...
115	7,77	4,9548	2,8152

Error merupakan selisih antara data asli dengan hasil peramalan. Apabila hasil tersebut disajikan dalam bentuk plot data, maka dapat dilihat pada gambar 4.10, dengan grafik berwarna biru merupakan grafik target/data asli, dan grafik merah merupakan grafik hasil tes/percobaan.



Gambar 4.10 Plot data pengujian peramalan model J

Percobaan pada model ini menghasilkan nilai MSE sebesar 42.1293 dan WMAPE sebesar 93.6525%, diperoleh dari perhitungan,

$$MSE = \frac{1}{115} (2,0168^2 + (-12,7476)^2 + \dots + 2,8152^2) = 42.1293$$

$$WMAPE = \frac{|2,0168| + |-12,7476| + \dots + |2,8152|}{|7,4516| + |0,1667| + \dots + |7,77|} \times 100 = 25.4792\%$$

Apabila ditinjau pada tabel 2.3, hasil peramalan tersebut masuk dalam kategori hasil peramalan buruk.

4.2 Hasil Pengujian Model

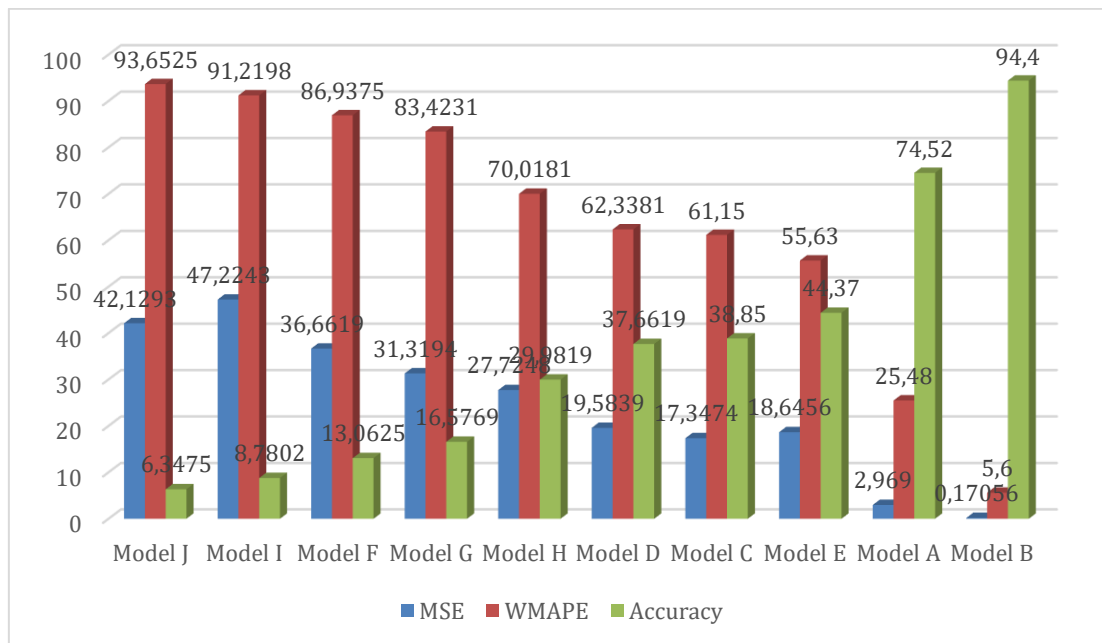
Pada bab 4.1 telah dilakukan pengujian model dari 10 model yang berbeda.

Hasil 10 model tersebut dihimpun dalam tabel 4.11 dibawah ini.

Tabel 4.12 Hasil pengujian model

Model ke-	MSE	WMAPE	Akurasi Peramalan
A	2,969	25,4792%	74,5208%
B	0,17056	5,6036%	94,3964%
C	17,3474	61,1503%	38,8497%
D	19,5839	62,3381%	37,6619%
E	18,6456	55,6322%	44,3678%
F	36,6619	86,9375%	13,0625%
G	31,3194	83,4231%	16,5769%
H	27,7248	70,0181%	29,9819%
I	47,2243	91,2198%	8,7802%
J	42,1293	93,6525%	6,3475%

Tabel tersebut menampilkan nilai MSE, WMAPE, dan akurasi dari 10 model yang berbeda. Nilai akurasi diambil dari 100% dikurangi hasil dari WMAPE (Widodo et al., 2017). Dari pegujian tersebut, hasil terbaik pertama yaitu pada model B dengan hasil akurasi 94,3964%, dan hasil terendah pada model J dengan akurasi 6,3475%. Hasil tersebut apabila disajikan dalam grafik yang diurutkan dari hasil terkecil ke terbesar dapat dilihat dalam gambar 4.11.



Gambar 4.11 Grafik 10 model peramalan dari yang terkecil ke terbesar

Melalui penelitian ini, diharapkan dapat membantu sesama umat manusia untuk saling mengingatkan dan menjaga satu sama lain agar terhindar dari mara bahaya cuaca buruk, yang dapat menyebabkan banjir dan tanah longsor, *naudzubillah min dzalik*. Seperti yang telah disebutkan dalam surat Adz-Dzariyat ayat 55 yang berbunyi,

وَذَكِّرْ فَإِنَّ الذِّكْرَى تَنْفَعُ الْمُؤْمِنِينَ

"Dan tetaplah memberi peringatan, karena sesungguhnya peringatan itu bermanfaat bagi orang-orang yang beriman." (Q.S Adz-Dzariyat ayat 55).

Selain surat Adz Dzariyat ayat 55 untuk menjadi pendorong semangat dalam mengingatkan dan menasehati, banyak hadits yang juga bisa menambah motivasi. Antara lain yang berasal dari Abu Hurairah RA, bahwa Rasulullah SAW bersabda yang artinya,

"Barang siapa mengajak kepada kebaikan, maka ia akan mendapat pahala sebanyak pahala yang diperoleh orang-orang yang mengikutinya tanpa mengurangi pahala mereka sedikitpun. Sebaliknya, barangsiapa mengajak kepada kesesatan, maka ia akan mendapat dosa sebanyak yang diperoleh orang-orang yang

mengikutinya tanpa mengurangi dosa mereka sedikitpun." (HR Muslim no. 4831 disahihkan oleh ijma' Ulama).

Selain itu, kita juga tidak boleh mengeluh sedikitpun akan cobaan yang telah diberikan oleh Allah SWT kepada kita. Cobaan seperti bencana alam merupakan cara Allah untuk memberikan pelajaran kepada seseorang agar bisa menjadi sosok yang lebih baik. Seperti yang telah difirmankan oleh Allah dalam surat Al-Baqarah ayat 155-157 yang berbunyi,

وَلَنَبْلُوَنَّكُمْ بِشَيْءٍ مِّنَ الْخَوْفِ وَالْجُوعِ وَنَقْصٍ مِّنَ الْأَمْوَالِ وَالْأَنْفُسِ وَالثَّمَرَاتِ وَبَشِّرِ الصَّابِرِينَ ﴿١٥٥﴾ الَّذِينَ إِذَا
 أَصَابَتْهُمُ مُصِيبَةٌ قَالُوا إِنَّا لِلَّهِ وَإِنَّا إِلَيْهِ رَاجِعُونَ ﴿١٥٦﴾ أُولَٰئِكَ عَلَيْهِمْ صَلَوَاتٌ مِّن رَّبِّهِمْ وَرَحْمَةٌ وَأُولَٰئِكَ هُمُ الْمُهْتَدُونَ

“Dan Kami pasti akan menguji kamu dengan sedikit ketakutan, kelaparan, kekurangan harta, jiwa, dan buah-buahan. Dan sampaikanlah kabar gembira kepada orang-orang yang sabar, (yaitu) orang-orang yang apabila ditimpa musibah, mereka berkata “Inna lillahi wa inna ilaihi raji’un” (sesungguhnya kami milik Allah dan kepada-Nya lah kami kembali). Mereka itulah yang memperoleh ampunan dan rahmat dari Tuhannya, dan mereka itulah orang-orang yang mendapat petunjuk.” (Q.S Al-Baqarah ayat 155-157).

Ayat tersebut menyiratkan bahwa tiap manusia akan dilanda dengan berbagai macam cobaan (ketakutan, kelaparan, kekurangan harta, jiwa, dan buah-buahan). Di antara orang-orang tersebut hanya orang-orang yang sabar dan tawakal lah yang bisa memperoleh pengampunan dan petunjuk.

Allah juga tidak akan menguji manusia dengan ujian yang di luar kemampuan umatnya, hal ini juga telah difirmankan oleh Allah dalam surat Al-Baqarah ayat 286 yang artinya, *“Allah tidak membebani seseorang melainkan sesuai dengan kesanggupannya.” (Q.S Al-Baqarah ayat 286).*

BAB V

KESIMPULAN DAN SARAN

Pada bab ini akan menjelaskan kesimpulan dari penelitian implementasi algoritma *Artificial Neural Network Backpropagation* untuk peramalan curah hujan di kabupaten Malang, yang telah dilakukan pada bab-bab sebelumnya. Juga saran yang bisa disampaikan apabila menggunakan penelitian ini sebagai referensi yang diharapkan dapat menghasilkan penelitian semakin baik untuk kedepannya.

5.1 Kesimpulan

Melalui penelitian ini, dapat disimpulkan beberapa hal sebagai berikut:

1. Perubahan fitur pada model *Artificial Neural Network* sangat berpengaruh pada peramalan yang dihasilkan. Jika ditinjau dari tabel 4.11, parameter yang paling mempengaruhi hasil peramalan adalah banyaknya *hidden layer*, karena hasil model terbaik terjadi pada model B yang memiliki 1 *hidden layer*, dan model terendah terjadi pada model J yang memiliki 5 *hidden layer*. Semakin banyak *hidden layer* maka hasil akurasinya semakin kecil.
2. Diambil dari model ANN terbaik yakni model B dari tabel 4.11, besar kesalahan/*error* dan akurasi penerapan *Artificial Neural Network Backpropagation* dalam melakukan peramalan curah hujan di kabupaten Malang adalah MSE sebesar 0.17056, WMAPE 5,6036%, dan akurasi peramalan sebesar 94,3964% yang termasuk dalam kategori peramalan yang sangat baik.

5.2 Saran

Saran yang dapat penulis sampaikan untuk penelitian selanjutnya adalah sebagai berikut:

1. Pada penelitian ini parameter yang digunakan hanya menggunakan data historis curah hujan saja, diharapkan pada penelitian selanjutnya bisa menggunakan parameter cuaca yang lain seperti temperatur udara, kelembaban udara, arah angin, kecepatan angin, dll.
2. Model dan arsitektur ANN pada penelitian ini menggunakan implementasi algoritma *gradient descent* sebagai optimasi, yakni optimasi yang umum dipakai pada model ANN. Sehingga pada penelitian selanjutnya diharapkan bisa menerapkan algoritma optimasi lain yang kemungkinan memiliki performa lebih baik seperti *Adaptive Gradient* (AdaGrad), *Adaptive Moment Estimation* (Adam), *Mini Batch Stochastic Gradient Descent* (MB-SGD), *Gradient Descent Momentum* (GDM), *Stochastic Gradient Descent* (SGD), dll. Atau mengkombinasikannya dengan algoritma lain seperti *Genetic Algorithm* (GA), *Particle Swarm Optimization* (PSO), *Fuzzy*, dll.

DAFTAR PUSTAKA

- Abhishek, K., Singh, M. P., Ghosh, S., & Anand, A. (2012). Weather forecasting model using artificial neural network. *Procedia Technology*, 4, 311–318.
- Andrian, Y., & Ningsih, E. (2017). Prediksi curah hujan di Kota Medan menggunakan metode backpropagation neural network. *Seminar Nasional Informatika (SNIf)*, 1(1), 184–189.
- Ashari, A., & Muniar, A. Y. (2015). Penerapan Jaringan Syaraf Tiruan dan Sistem Pakar untuk Mengidentifikasi Penyakit Pencernaan dengan Pengobatan Herbal. *Proceedings Konferensi Nasional Sistem Dan Informatika (KNS&I)*.
- Banjir dan Longsor Terjang Kota Malang, Seorang Warga Hilang Diduga Tertimbun: Okezone News.* (n.d.). Retrieved April 26, 2021, from <https://news.okezone.com/read/2021/01/18/519/2346662/banjir-dan-longsor-terjang-kota-malang-seorang-warga-hilang-diduga-tertimbun>
- DATA ONLINE - PUSAT DATABASE - BMKG.* (n.d.). Retrieved April 26, 2021, from <https://dataonline.bmkg.go.id/home>
- Fadholi, A. (2013). Pemanfaatan suhu udara dan kelembaban udara dalam persamaan regresi untuk simulasi prediksi total hujan bulanan di Pangkalpinang. *CAUCHY: Jurnal Matematika Murni Dan Aplikasi*, 3(1), 1–9.
- Fausett, L. V. (2006). *Fundamentals of neural networks: architectures, algorithms and applications*. Pearson Education India.
- Ferdiansyah, R. (n.d.). Analisis Model Perencanaan dan Pengendalian Persediaan Baby Product Studi Kasus di PT. Multi Indocitra, Tbk. *Operations Excellence*, 10(1), 26–40.
- Ginting, R. (2007). Sistem produksi. *Yogyakarta: Graha Ilmu*.
- Handayani, L., & Adri, M. (2015). Penerapan JST (Backpropagation) untuk Prediksi curah hujan (Studi kasus: Kota Pekanbaru). *Seminar Nasional Teknologi Informasi Komunikasi Dan Industri*.
- Heizer, J., & Render, B. (2009). Manajemen operasi buku 1 edisi 9. *Jakarta: Salemba Empat*.
- Herjanto, E. (2007). *Manajemen Operasi (Edisi 3)*. Grasindo.
- Julpan, J., Nababan, E. B., & Zarlis, M. (2018). Analisis Fungsi Aktivasi Sigmoid Biner dan Sigmoid Bipolar dalam Algoritma Backpropagation pada Prediksi Kemampuan Siswa. *Jurnal Teknovasi: Jurnal Teknik Dan Inovasi*, 2(1), 103–

- Kiki, K., & Kusumadewi, S. (2004). Jaringan Saraf Tiruan dengan Metode Backpropagation untuk Mendeteksi Gangguan Psikologi. *Media Informatika*, 2(2).
- Lahiri, S. K., & Ghanta, K. C. (2009). Artificial neural network model with the parameter tuning assisted by a differential evolution technique: The study of the hold up of the slurry flow in a pipeline. *Chemical Industry and Chemical Engineering Quarterly/CICEQ*, 15(2), 103–117.
- Lewis, C. D. (1982). *Industrial and business forecasting methods: A practical guide to exponential smoothing and curve fitting*. Butterworth-Heinemann.
- Navianti, D. R., & Widjajati, F. A. (2012). Penerapan fuzzy inference system pada prediksi curah hujan di Surabaya Utara. *Jurnal Sains Dan Seni ITS*, 1(1), A23–A28.
- Oktaviani, C., & Afdal, A. (2013). Prediksi Curah Hujan Bulanan menggunakan Jaringan Syaraf Tiruan dengan Beberapa Fungsi Pelatihan Backpropagation (Studi Kasus: Stasiun Meteorologi Tabing Padang, Tahun 2001-2012). *Jurnal Fisika Unand*, 2(4).
- Prawaka, F., Zakaria, A., & Tugiono, S. (2016). Analisis Data Curah Hujan yang Hilang Dengan Menggunakan Metode Normal Ratio, Inversed Square Distance, dan Cara Rata-Rata Aljabar (Studi Kasus Curah Hujan Beberapa Stasiun Hujan Daerah Bandar Lampung). *Jurnal Rekayasa Sipil Dan Desain*, 4(3), 397–406.
- Pressman, R. S., & Lunak, R. P. (2012). Buku Satu, Pendekatan Praktisi, Edisi ke7. Yogyakarta: Andi.
- Susanto, Y. (2016). *Peramalan Curah Hujan dengan Pendekatan Model ARIMA, Feed Forward Neural Network dan Hybrid (ARIMA-NN) di Banyuwangi*. Institut Teknologi Sepuluh Nopember.
- Sutawinaya, I. P., Astawa, I. N. G. A., & Hariyanti, N. K. D. (2017). Perbandingan Metode Jaringan Saraf Tiruan pada Peramalan Curah Hujan. *Logic: Jurnal Rancang Bangun Dan Teknologi*, 17(2), 92–97.
- Syaifullah, M. D. (2014). Validasi data TRMM terhadap data curah hujan aktual di tiga DAS di Indonesia. *Jurnal Meteorologi Dan Geofisika*, 15(2).
- Warsito, B., & Sumiyati, S. (2007). Prediksi Curah Hujan kota Semarang dengan Feedforward Neural Network Menggunakan Algoritma Quasi Newton BFGS dan Levenberg-Marquardt. *Jurnal Presipitasi*, 3(2), 46–52.
- Widodo, A. P., Suhartono, S., Sarwoko, E. A., & Firdaus, Z. (2017). Akurasi Model Prediksi Metode Backpropagation Menggunakan Kombinasi Hidden Neuron

Dengan Alpha. *Matematika*, 20(2), 79–84.

Zacharis, N. Z. (2016). Predicting student academic performance in blended learning using Artificial Neural Networks. *International Journal of Artificial Intelligence and Applications*, 7(5), 17–29.

**LAMPIRAN DATA RATA-RATA CURAH HUJAN BULANAN DAERAH
KABUPATEN MALANG TAHUN 2000 SAMPAI 2020**

Bulan	Rata-rata Curah Hujan	Tahun	Bulan	Rata-rata Curah Hujan	Tahun
Januari	11,25926	2000	Januari	4,483871	2011
Februari	8,923077		Februari	6,5	
Maret	10,43333		Maret	10,93548	
April	3,793103		April	5,333333	
Mei	4,827586		Mei	7,451613	
Juni	1,413793		Juni	0,166667	
Juli	0,258065		Juli	0,064516	
Agustus	0,096774		Agustus	0	
September	3,413793		September	0,066667	
Oktober	7,827586		Oktober	2,032258	
November	16,7931		November	9,517241	
Desember	2,870968		Desember	8,645161	
Januari	12,3871	2001	Januari	10,214285	2012
Februari	9,16129		Februari	15,071428	
Maret	9,967742		Maret	6,9032258	
April	3,833333		April	2,68	

Mei	3,709677		Mei	0,8275862	
Juni	4,166667		Juni	0,5925926	
Juli	1,344828		Juli	0	
Agustus	0,387097		Agustus	0,1290322	
September	0,571429		September	0	
Oktober	4,9		Oktober	3,5161290	
November	4,8		November	5,3571428	
Desember	7,833333		Desember	15,548387	
Januari	15,72414		Januari	11,8	
Februari	10,60714		Februari	7,625	
Maret	4,724138		Maret	9,258065	
April	6,3		April	7,27	
Mei	1,633333		Mei	0,780645	
Juni	0		Juni	0,592593	
Juli	0,064516	2002	Juli	0	2013
Agustus	0		Agustus	0,129032	
September	0,034483		September	0	
Oktober	0,433333		Oktober	3,458065	
November	3,517241		November	5,321429	
Desember	14,58621		Desember	15,53871	

Januari	7,741935	2003	Januari	12,33226	2014
Februari	15,21429		Februari	6,4	
Maret	9,064516		Maret	5,874194	
April	1,066667		April	9,796667	
Mei	3,322581		Mei	1,296774	
Juni	0,666667		Juni	1,47	
Juli	0,032258		Juli	0,290323	
Agustus	0		Agustus	1,296774	
September	0		September	0	
Oktober	1,483871		Oktober	0,548387	
November	7,4		November	4,776667	
Desember	20,90323		Desember	10,9129	
Januari	10,74194	2004	Januari	10,33636	2015
Februari	12,03448		Februari	15,652	
Maret	10,58065		Maret	8,562069	
April	1,466667		April	10,24483	
Mei	3,322581		Mei	3,847619	
Juni	0,266667		Juni	1,473077	
Juli	0,032258		Juli	0	
Agustus	0		Agustus	0	

September	1,1		September	0,337037	
Oktober	0,677419		Oktober	0	
November	11,4		November	5,914286	
Desember	6,935484		Desember	7,03	
Januari	5		Januari	7,425	
Februari	8,035714		Februari	24,94444	
Maret	9,322581		Maret	7,512	
April	6,833333		April	3,216	
Mei	0,096774		Mei	6,663333	
Juni	0,433333		Juni	5,635714	
Juli	2,225806	2005	Juli	3,872414	2016
Agustus	0,451613		Agustus	3,113793	
September	0,466667		September	1,648148	
Oktober	2,419355		Oktober	9,526923	
November	5,666667		November	14,46	
Desember	9,032258		Desember	10,31538	
Januari	10,06452		Januari	12,9069	
Februari	13,85714		Februari	8,818519	
Maret	9,032258	2006	Maret	13,54333	2017
April	5,6		April	9,078571	

Mei	5,354839		Mei	1,448276	
Juni	0,733333		Juni	1,166667	
Juli	0,032258		Juli	2,068421	
Agustus	0,258065		Agustus	0,1	
September	0,033333		September	7,65	
Oktober	0,483871		Oktober	7,453333	
November	0,866667		November	14,66296	
Desember	6,774194		Desember	10,23333	
Januari	3,774194		Januari	15,14828	
Februari	8,392857		Februari	8,696154	
Maret	5,645161		Maret	7,377778	
April	7,866667		April	2,853571	
Mei	0,16129		Mei	0,603448	
Juni	0,5		Juni	1,868966	
Juli	0,225806	2007	Juli	0,32069	2018
Agustus	0,032258		Agustus	0	
September	0,333333		September	0,11	
Oktober	1,645161		Oktober	0,213333	
November	9,066667		November	12,73077	
Desember	13,67742		Desember	14,01034	

Januari	6,677419	2008	Januari	11,02593	2019
Februari	10,89655		Februari	12,47037	
Maret	14,83871		Maret	11,94194	
April	2,2		April	10,95517	
Mei	2,032258		Mei	2,42	
Juni	0,066667		Juni	0	
Juli	0		Juli	2,010345	
Agustus	1,548387		Agustus	0	
September	0,3		September	0	
Oktober	2,967742		Oktober	0,041935	
November	5,866667		November	3,844828	
Desember	7,774194		Desember	10,28387	
Januari	8,322581	2009	Januari	10,91	2020
Februari	15,57143		Februari	17,32414	
Maret	2,645161		Maret	7,703226	
April	2,266667		April	5,317241	
Mei	3,258065		Mei	4,67931	
Juni	2,333333		Juni	0,106667	
Juli	1,258065		Juli	0,58	
Agustus	0		Agustus	2,513636	

September	0,133333		September	0,317647	
Oktober	1,096774		Oktober	4,36	
November	6,666667		November	7,77	
Desember	7,258065		Desember	12,27692	
Januari	11,32258	2010			
Februari	7,785714				
Maret	6,741935				
April	17,63333				
Mei	11				
Juni	1				
Juli	2,967742				
Agustus	4,354839				
September	6,266667				
Oktober	4,548387				
November	11,13333				
Desember	8,419355				