

**IMPLEMENTASI ALGORITMA *BACKPROPAGATION*  
UNTUK MEMPREDIKSI TINGKAT INFLASI  
DI KOTA MALANG**

**SKRIPSI**

**OLEH  
AISYAH SUKMAINDAH TRISTIANTI  
NIM. 200601110081**



**PROGRAM STUDI MATEMATIKA  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM  
MALANG  
2024**

**IMPLEMENTASI ALGORITMA *BACKPROPAGATION*  
UNTUK MEMPREDIKSI TINGKAT INFLASI  
DI KOTA MALANG**

**SKRIPSI**

**Diajukan Kepada  
Fakultas Sains dan Teknologi  
Universitas Islam Negeri Maulana Malik Ibrahim Malang  
untuk Memenuhi Salah Satu Persyaratan dalam  
Memperoleh Gelar Sarjana Matematika (S.Mat)**

**Oleh  
Aisyah Sukmaindah Trisianti  
NIM. 200601110081**

**PROGRAM STUDI MATEMATIKA  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM  
MALANG  
2024**

**IMPLEMENTASI ALGORITMA *BACKPROPAGATION*  
UNTUK MEMPREDIKSI TINGKAT INFLASI  
DI KOTA MALANG**

**SKRIPSI**

**Oleh  
Aisyah Sukmaindah Trisianti  
NIM. 200601110081**

Telah Disetujui Untuk Diuji

Malang, 07 Juni 2024

Dosen Pembimbing I



Hisyani Fahmi, M.Kom.  
NIP. 19890727 201903 1 018

Dosen Pembimbing II



Erna Herawati, M.Pd.  
NIPPPK. 19760723 202321 2 006



Mengetahui,  
Ketua Program Studi Matematika



Dr. Elly Susanti, M.Sc.  
NIP. 19741129 100012 2 005

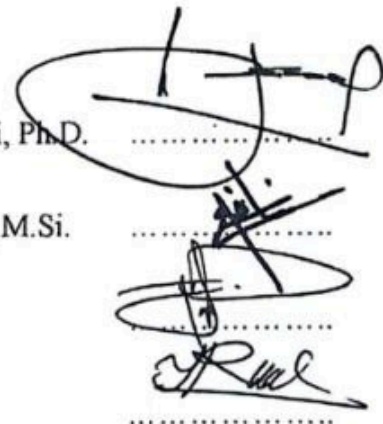
# IMPLEMENTASI ALGORITMA *BACKPROPAGATION* UNTUK MEMPREDIKSI TINGKAT INFLASI DI KOTA MALANG

## SKRIPSI

Oleh  
**Aisyah Sukmaindah Trisianti**  
NIM. 200601110081


Telah Dipertahankan di Depan Penguji Skripsi  
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan  
untuk Memperoleh Gelar Sarjana Matematika (S.Mat)  
Tanggal 14 Juni 2024

Ketua Penguji : Prof. Dr. H. Turmudi, M.Si, Ph.D. ....  
Anggota Penguji 1 : Mohammad Nafie Jauhari, M.Si. ....  
Anggota Penguji 2 : Hisyam Fahmi, M.Kom. ....  
Anggota Penguji 3 : Erna Herawati, M.Pd. ....



Mengetahui,  
Ketua Program Studi Matematika



  
Dr. Elly Susanti, M.Sc.  
NIP. 19741129 200012 2 005

## PERNYATAAN KEASLIAN TULISAN

Saya bertanda tangan dibawah ini

Nama : Aisyah Sukmaindah Trisianti

NIM : 200601110081

Jurusan : Matematika

Fakultas : Sains dan Teknologi

Judul Skripsi : Implementasi Algoritma Backpropagation untuk Memprediksi  
Tingkat Inflasi di Kota Malang

Menyatakan dengan sebenarnya bahwa skripsi yang saya tulis ini merupakan hasil karya sendiri, bukan pengambilan tulisan atau pemikiran orang lain yang saya akui sebagai pemikiran saya, kecuali dengan mencantumkan sumber cuplikan pada daftar rujukan di halaman terakhir. Apabila dikemudian hari terbukti skripsi ini adalah hasil jiplakan atau tiruan, maka saya bersedia menerima sanksi yang berlaku atas perbuatan tersebut.

Malang, 14 Juni 2024



Aisyah Sukmaindah Trisianti  
NIM. 200601110081

## **MOTO**

*“Innallaha ma’ashobirin”*

(Q.S. Al-Baqarah/2:153)

*“Man jadda wa jadda”*

*“Nothing is impossible if you want to try.”*

(Kim Jong In)

## **PERSEMBAHAN**

Bismillahirrahmaanirrahiim

Alhamdulillah robbil aalamiin

Dengan segenap hati skripsi ini dipersembahkan untuk :

Untuk diriku yang terus berusaha tidak menyerah dan mengakui bahwa rencana Allah selalu lebih indah, dan tak ada daya kekuatan tanpa kasih sayang Allah yang begitu luas memberikan penulis kesempatan untuk selalu memperbaiki diri setiap harinya melalui proses menyelesaikan tugas akhir.

## KATA PENGANTAR

*Assalamu'alaikum Warahmatullahi Wabarakatuh.*

Segala puji bagi Allah SWT yang selalu melimpahkan rahmat, taufik dan hidayah-Nya, sehingga penulis dapat menyelesaikan skripsi yang berjudul “Implementasi Algoritma *Backpropagation* untuk Memprediksi Tingkat Inflasi di Kota Malang” sebagai salah satu syarat untuk memperoleh gelar sarjana dalam bidang Matematika di Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang. Shalawat serta salam senantiasa tercurahkan kepada Nabi Muhammad SAW yang telah menuntun manusia dari jalan kegelapan menuju ke jalan yang terang benderang yaitu Islam.

Dalam penyusunan skripsi ini tidak lepas dari petunjuk dan bimbingan serta masukan dari berbagai pihak. Untuk itu, penulis mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Prof. Dr. H. M. Zainuddin, M.A., selaku rektor Universitas Islam Negeri Maulana Malik Ibrahim Malang.
2. Prof. Dr. Sri Harini, M.Si., selaku dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
3. Dr. Elly Susanti, S.Pd., M.Sc., selaku ketua Program Studi Matematika, Universitas Islam Negeri Maulana Malik Ibrahim Malang.
4. Hisyam Fahmi, M.Kom., selaku dosen pembimbing I yang telah memberikan berbagai pengetahuan, pengalaman, arahan, nasihat, serta motivasi kepada penulis.
5. Erna Herawati, M.Pd., selaku dosen pembimbing II yang telah memberikan berbagai pengetahuan, pengalaman, arahan, nasihat, serta motivasi kepada penulis.
6. Prof. Dr. H. Turmudi, M.Si, Ph.D., selaku Ketua Penguji dalam Ujian Skripsi.
7. Mohammad Nafie Jauhari, M.Si., selaku Anggota Penguji 1 dalam Ujian Skripsi.
8. Seluruh dosen Program Studi Matematika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang.



9. Kedua orang tua dan seluruh keluarga yang senantiasa memberikan doa, semangat, dukungan, nasihat, dan kasih sayang kepada penulis.
10. Teman-teman Mahatma'20 yang berjuang bersama dalam mengerjakan skripsi yang senantiasa memberikan dukungan dan semangat kepada penulis.
11. Teman-teman penulis khususnya Ajeng, Shofy, Pingka, dan Mila, yang telah menemani, membantu, dan memberi semangat kepada penulis dalam mengerjakan skripsi dari awal sampai akhir.

Semoga Allah SWT selalu memberikan balasan atas segala bantuan dan kebaikan yang telah diberikan kepada penulis. Penulis berharap agar skripsi ini dapat bermanfaat bagi penulis serta pembaca untuk menambah wawasan keilmuan yang selalu berkembang.

*Wassalamu'alaikum Warahmatullahi Wabarakatuh.*

Malang, 14 Juni 2024

Penulis

## DAFTAR ISI

HALAMAN JUDUL .....	i
HALAMAN PENGAJUAN .....	ii
HALAMAN PERSETUJUAN .....	iii
HALAMAN PENGESAHAN .....	iv
PERNYATAAN KEASLIAN TULISAN .....	v
MOTO .....	vi
PERSEMBAHAN.....	vii
KATA PENGANTAR.....	viii
DAFTAR ISI.....	x
DAFTAR GAMBAR.....	xii
DAFTAR TABEL .....	xiii
DAFTAR LAMPIRAN .....	xiv
ABSTRAK .....	xv
ABSTRACT .....	xvi
مستخلص البحث.....	xvii
<b>BAB I PENDAHULUAN.....</b>	<b>1</b>
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	6
1.3 Tujuan Penelitian.....	6
1.4 Manfaat Penelitian.....	7
1.5 Batasan Masalah .....	8
1.6 Istilah-istilah .....	8
<b>BAB II KAJIAN TEORI .....</b>	<b>10</b>
2.1 <i>Machine learning</i> (ML).....	10
2.2 Jaringan Saraf Tiruan (JST).....	12
2.3 <i>Backpropagation</i> .....	14
2.3.1 Pengertian <i>Backpropagation</i> .....	14
2.3.2 Komponen <i>Backpropagation</i> .....	15
2.3.3 Algoritma <i>Backpropagation</i> .....	18
2.3.4 Tahapan <i>Backpropagation</i> .....	20
2.3.5 Normalisasi Data.....	26
2.4 Inflasi.....	27
2.4.1 Pengertian Inflasi .....	27
2.4.2 Penyebab Inflasi.....	28
2.4.3 Tingkat Inflasi .....	30
2.4.4 Pengukuran Tingkat Inflasi.....	30
2.5 Akurasi.....	31
2.5.1 Pengertian Akurasi.....	31
2.5.2 Perhitungan Akurasi.....	31
2.6 Penelitian Terdahulu.....	33
2.7 Kajian Integrasi Islam.....	34
<b>BAB III METODE PENELITIAN .....</b>	<b>41</b>
3.1 Jenis Penelitian .....	41
3.2 Data dan Sumber Data.....	41
3.3 Teknik Analisis Data .....	42
<b>BAB IV HASIL DAN PEMBAHASAN .....</b>	<b>46</b>

4.1 Mengumpulkan Data .....	46
4.2 Melakukan <i>Preprocessing</i> Data.....	48
4.3 Menentukan Model Arsitektur Jaringan .....	50
4.4 Melakukan <i>Training</i> Data.....	52
4.5 Melakukan <i>Testing</i> Data dan Evaluasi .....	60
4.6 Implementasi Algoritma <i>Backpropagation</i> dalam Pandangan Islam .....	69
<b>BAB V PENUTUP.....</b>	<b>73</b>
5.1 Kesimpulan.....	73
5.2 Saran .....	74
<b>DAFTAR PUSTAKA .....</b>	<b>75</b>
<b>LAMPIRAN.....</b>	<b>78</b>
<b>RIWAYAT HIDUP .....</b>	<b>89</b>

## DAFTAR GAMBAR

Gambar 2.1	Arsitektur Jaringan <i>Backpropagation</i> 4-3-1, yaitu Empat Neuron di <i>Input Layer</i> Tiga Neuron di <i>Hidden Layer</i> dan Satu Neuron di <i>Output Layer</i> .....	15
Gambar 3.1	Diagram Alir Teknik Analisis Data.....	45
Gambar 4.1	Grafik Data Inflasi Kota Malang Tahun 2005-2019 Perbulan dan Perkelompok .....	47
Gambar 4.2	Grafik Data Inflasi Hasil Normalisasi .....	49
Gambar 4.3	Model Arsitektur Jaringan <i>Backpropagation</i> 7-4-1 yaitu Tujuh Neuron di <i>Input Layer</i> 4 Neuron di <i>Hidden Layer</i> dan Satu Neuron di <i>Output Layer</i> .....	51
Gambar 4.4	Grafik Perbandingan Hasil <i>Output</i> dalam Bentuk Denormalisasi dengan Data Target Asli Tingkat Inflasi di Kota Malang Tahun 2019 dengan Menggunakan Model Arsitektur 7-6-1 dengan Parameter Fungsi Aktivasi <i>Tanh</i> , Ukuran <i>Batch</i> 8, dan Jumlah <i>Epoch</i> 1000.....	68

## DAFTAR TABEL

Tabel 4.1	Data Inflasi Kota Malang Tahun 2005-2019 Perbulan dan Perkelompok .....	46
Tabel 4.2	Statistik Deskriptif dari Data Inflasi Setiap Kelompok .....	47
Tabel 4.3	Contoh 5 Data Inflasi Hasil Normalisasi .....	49
Tabel 4.4	Data <i>Training</i> .....	52
Tabel 4.5	Data <i>Testing</i> .....	52
Tabel 4.6	Rangkuman Hasil Proses <i>Training</i> Model Arsitektur 7-14-1 .....	59
Tabel 4.7	Hasil Bobot dan Bias $V$ antara <i>Input Layer</i> dengan <i>Hidden Layer</i> dari Proses <i>Training</i> .....	59
Tabel 4.8	Hasil Bobot $W$ dan Bias $W$ antara <i>Hidden Layer</i> dengan <i>Output Layer</i> dari Proses <i>Training</i> .....	60
Tabel 4.9	Hasil Akurasi Model Arsitektur dengan Parameter Fungsi Aktivasi Relu dan Ukuran <i>Batch</i> 8 .....	61
Tabel 4.10	Hasil Akurasi Model Arsitektur dengan Parameter Fungsi Aktivasi Relu dan Ukuran <i>Batch</i> 16 .....	62
Tabel 4.11	Hasil Akurasi Model Arsitektur dengan Parameter Fungsi Aktivasi <i>Tanh</i> dan Ukuran <i>Batch</i> 8 .....	63
Tabel 4.12	Hasil Akurasi Model Arsitektur dengan Parameter Fungsi Aktivasi <i>Tanh</i> dan Ukuran <i>Batch</i> 16 .....	64
Tabel 4.13	Tabel Perbandingan Hasil <i>Output</i> dan Nilai Target (dalam Bentuk Denormalisasi) dari Model Arsitektur 7-9-1 dengan Fungsi Aktivasi ReLu, Ukuran <i>Batch</i> 8 dan Jumlah <i>Epoch</i> 1000 .....	65
Tabel 4.14	Hasil Akurasi Hasil <i>Output</i> Bentuk Normalisasi dan Denormalisasi dari Model Terbaik dari Setiap Parameter .....	67
Tabel 4.15	Hasil Denormalisasi dari Model Arsitektur 7-14-1 dengan Parameter Fungsi Aktivasi <i>Tanh</i> , Ukuran <i>Batch</i> 16, dan Jumlah <i>Epoch</i> 1000 .....	67

## DAFTAR LAMPIRAN

Lampiran 1	Data Inflasi Bulanan Kota Malang Tiap Kelompok dari 2005-2019.....	78
Lampiran 2	Data Inflasi Hasil Normalisasi.....	82
Lampiran 3	Sriipt Python .....	87

## ABSTRAK

Trisianti, Aisyah Sukmaindah. 2024. **Implementasi Algoritma Backpropagation untuk Memprediksi Tingkat Inflasi di Kota Malang**. Skripsi. Program Studi Matematika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Hisyam Fahmi, M.Kom. (II) Erna Herawati, M.Pd.

**Kata Kunci:** Backpropagation, Tingkat Inflasi, Prediksi, Jaringan Syaraf Tiruan.

Backpropagation merupakan salah satu model dari Jaringan Saraf Tiruan (JST) yang menggunakan pembelajaran terawasi untuk memecahkan masalah yang kompleks, seperti memprediksi tingkat inflasi dengan cara dilatih menggunakan metode pembelajaran maju dan koreksi kesalahan mundur. Tingkat inflasi merupakan persentase perubahan harga-harga barang dan jasa di dalam perekonomian selama periode tertentu. Di sepanjang bulan Desember 2022 pihak Badan Pusat Statistik (BPS) mencatat angka inflasi di Kota Malang sebesar 0,58%, salah satu penyebabnya yaitu meningkatnya harga beras dan komoditas lain seperti telur, ayam, dan cabai rawit. Dari hal tersebut, maka perlu dilakukan prediksi untuk membantu mempersiapkan program yang harus dilakukan untuk mengatasi inflasi. Penelitian ini bertujuan untuk mengetahui tingkat akurasi dari algoritma backpropagation dengan menguji beberapa parameter serta mengetahui hasil prediksi inflasi di Kota Malang pada tahun 2019. Temuan dari penelitian ini adalah hasil prediksi inflasi terbaik didapatkan dengan model arsitektur 7-14-1 dengan parameter fungsi aktivasi tanh, ukuran *batch* 16 dan jumlah *epoch* 1000, dengan hasil akurasi sebesar 87,49%. Kemudian dari hasil prediksi tersebut, inflasi tertinggi di Kota Malang diprediksi terjadi di bulan Januari 2019 dan inflasi terendah diprediksi terjadi di bulan Februari 2019.

## ABSTRACT

Trisianti, Aisyah Sukmaindah. 2024. **Implementation of Backpropagation Algorithm to Predict Inflation Rate in Malang City**. Thesis. Mathematics Study Program, Faculty of Science and Technology, Universitas Islam Negeri Maulana Malik Ibrahim Malang. Advisor: (I) Hisyam Fahmi, M.Kom (II) Erna Herawati, M.Pd.

**Keywords:** Backpropagation, Inflation Rate, Prediction, Artificial Neural Network.

Backpropagation is a model of Artificial Neural Networks (ANN) that uses supervised learning to solve complex problems, such as predicting the inflation rate by being trained using forward learning methods and backward error correction. The inflation rate is the percentage change in prices of goods and services in the economy during a certain period. Throughout December 2022, the Central Bureau of Statistics recorded the inflation rate in Malang City at 0.58%. One of the causes is the increase in prices of rice and other commodities such as eggs, chicken, and cayenne pepper. From this, predictions need to be made to help prepare programs that must be carried out to overcome inflation. This research aims to determine the level of accuracy of the backpropagation algorithm by testing several parameters and finding out the results of inflation predictions in Malang City in 2019. The findings of this research are the best inflation prediction results were obtained with the 7-14-1 architectural model with the tanh activation function parameters, batch size 16, and number of epochs 1000, with accuracy results of 87.49%. From the results of these predictions, the highest inflation in Malang City is predicted to occur in January 2019 and the lowest inflation is predicted to occur in February 2019.



## مستخلص البحث

تريستيانتي، عائشة سوكماليندة. ٢٠٢٤. تنفيذ خوارزمية الانتشار العكسي للتنبؤ بمعدل التضخم في مدينة مالانج. أطروحة. قسم الرياضيات، كلية العلوم والتكنولوجيا، جامعة مولانا مالك إبراهيم الإسلامية الحكومية مالانج. المشرف: (١) هشام فهمي، الماجستير. (٢) إيرناهيرواوتي، الماجستير.

**الكلمات المفتاحية:** الانتشار العكسي، معدل التضخم، التنبؤ، الشبكة العصبية.

الانتشار العكسي هو أحد نماذج الشبكات العصبية الاصطناعية (*JST*) التي تستخدم التعلم الخاضع للإشراف لحل المشكلات المعقدة، مثل التنبؤ بمعدل التضخم عن طريق التدريب باستخدام طريقة التعلم الأمامي وتصحيح الأخطاء العكسي. معدل التضخم هو نسبة التغير في أسعار السلع والخدمات في الاقتصاد خلال فترة معينة. طوال شهر ديسمبر ٢٠٢٢، سجلت هيئة الإحصاء المركزية (*BPS*) معدل التضخم في مدينة مالانج بنسبة ٠,٥٨٪، وكان أحد الأسباب هو ارتفاع أسعار الأرز والسلع الأخرى مثل البيض والدجاج والفلفل الحار. من هذا المنطلق، يصبح من الضروري إجراء التنبؤات للمساعدة في إعداد البرامج التي يجب تنفيذها لمكافحة التضخم. تهدف هذه الدراسة إلى معرفة مستوى دقة خوارزمية الانتشار العكسي من خلال اختبار عدة معايير ومعرفة نتائج التنبؤ بمعدل التضخم في مدينة مالانج لعام ٢٠١٩. ووجدت هذه الدراسة أن أفضل نتائج التنبؤ بمعدل التضخم تم الحصول عليها باستخدام النموذج المعماري ٧-١٤-١ مع معايير دالة التفعيل تانغ، وحجم الدفعة ١٦، وعدد العصور ١٠٠٠، مع نسبة دقة بلغت ٨٧,٤٩٪. ثم من نتائج هذه التنبؤات، من المتوقع أن يكون أعلى معدل تضخم في مدينة مالانج في شهر يناير ٢٠١٩ وأدنى معدل تضخم في شهر فبراير ٢٠١٩.

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Kecerdasan buatan atau biasa dikenal dengan AI adalah bidang ilmu komputer yang melibatkan pengembangan sistem komputer yang dapat melakukan tugas-tugas yang berhubungan dengan kecerdasan manusia. AI memiliki beberapa cabang, salah satunya adalah *Artificial Neural Netowrk* atau Jaringan Saraf Tiruan (JST). JST terdiri dari kumpulan neuron tiruan yang saling berhubungan untuk memproses dan mengirimkan informasi dengan menggunakan model matematika dan komputasi, serta pendekatan komputasi antar neuron (Hartati, 2021).

JST merupakan replika otak manusia yang meniru model jaringan saraf pada tubuh manusia yang diaplikasikan dengan program komputer yang dapat melakukan proses perhitungan dan pembelajaran seperti halnya yang terjadi di otak manusia (Ayu Trimulya, 2015). Kebanyakan jaringan ini digunakan untuk menyelesaikan masalah yang akan jadi rumit jika diselesaikan secara manual oleh manusia, seperti pengolahan data, pengenalan pola dan urutan, analisis regresi, proses klasifikasi, prediksi deret waktu, serta pengolahan data dan untuk bidang robotik.

Salah satu algoritma pada JST yaitu *backpropagation* atau propagasi balik. Algoritma *backpropagation* dapat melakukan proses prediksi yang hanya memiliki satu *hidden layer* atau lebih dan melakukan proses propagasi balik untuk memperbaiki nilai *error*. Konsep dari algoritma *backpropagation* ini yaitu dengan

melibatkan perhitungan gradien fungsi kerugian terhadap bobot jaringan, dan kemudian menggunakan aturan rantai untuk mengupdate bobot-bobot tersebut agar mengurangi nilai fungsi kerugian. Proses ini melibatkan penggunaan turunan parsial dan aturan rantai dalam kalkulus. Algoritma ini biasanya digunakan sebagai cara untuk memecahkan permasalahan yang rumit dengan menggunakan model pelatihan terawasi (*supervised learning*), kemudian akan dilatih dengan metode pembelajaran (Matondang, 2013). Algoritma *backpropagation* ini banyak digunakan karena algoritma ini tergolong mudah dan dapat dijelaskan dengan aturan belajar (Hutabarat, Julham, & Wanto, 2018).

Algoritma *backpropagation* merupakan salah satu algoritma pembelajaran mesin yang dapat digunakan untuk memprediksi hasil di masa depan. Algoritma ini merupakan salah satu ilmu yang meminimalkan gradien bobot melalui proses pelatihan berulang pada JST menggunakan arsitektur jaringan *multi-layer* (Haykin, 1999).

Algoritma *backpropagation* pertama kali dibangun pada tahun 1986 oleh Rumelhart, Hinton dan Williams. Selain algoritmanya sederhana dan mudah dipahami, algoritma ini juga biasa dipakai untuk melakukan prediksi data yang memiliki berbagai fitur dengan jumlah data dengan banyak serta tipe data yang kontinu. Algoritma ini juga efektif dalam mengoptimalkan bobot, serta dapat digunakan secara luas dalam *machine learning*.

Inflasi adalah kondisi di mana harga barang dan jasa secara umum dan terus menerus mengalami kenaikan. Laju inflasi Indeks Harga Konsumen (IHK) Indonesia yang dicatat Badan Pusat statistik (BPS) pada bulan Desember 2022 sebesar 0,66% (bulanan), sehingga laju inflasi IHK tahun 2022 sebesar 5,51%

(*year-on-year*), meningkat dibandingkan tahun sebelumnya yakni 1,87% (*year-on-year*). Kenaikan ini merupakan akibat dari penyesuaian harga Bahan Bakar Minyak (BBM) bersubsidi di bulan September tahun 2022. Selain itu, inflasi tingkat Indeks Harga Konsumen pada bulan Desember tahun 2022 juga disebabkan oleh pola musiman di akhir tahun. Tingkat inflasi ini diperkirakan akan kembali ke sasaran  $3,0 \pm 1\%$  pada tahun 2023.

Mufti Muhammad Taqi Usmani pada tahun 1998 dalam bukunya yang berjudul “*An Introduction to Islamic Finance*” menyebutkan faktor penyebab terjadinya inflasi. Mufti Muhammad Usmani berpendapat bahwa dalam praktik yang tidak alami seperti perjudian, transaksi spekulatif, dan kontrak masa depan, cenderung memusatkan kekayaan hanya pada beberapa orang saja. Kemudian timbul keinginan untuk melakukan bisnis monopoli, bukan lagi keuntungan berbasis pasar, dikarenakan hasrat untuk mencari keuntungan yang sudah tidak terkendali, dan hal ini digunakan sebagai strategi untuk menghasilkan dan membelanjakan uang. Kerusakan dalam bisnis inilah yang kemudian menyebabkan inflasi dalam perekonomian.

Selain itu, salah satu penyebab inflasi yaitu meningkatnya tingkat konsumsi masyarakat. Dari sifat masyarakat yang konsumtif tersebut, maka akan timbul keinginan untuk melakukan penimbunan demi untuk kepentingan individu. Tindakan penimbunan barang ini tidak dibenarkan dalam Islam. Allah SWT berfirman dalam Al-Qur’an Surah At-Taubah ayat 35:

يَوْمَ يُحْمَىٰ عَلَيْهَا فِي نَارِ جَهَنَّمَ فُتْكُوىٰ بِمَا جِبَاهُهُمْ وَجُنُوبُهُمْ وظُهُورُهُمْ هَذَا مَا كَنَزْتُمْ لِأَنفُسِكُمْ فَذُوقُوا  
مَا كُنْتُمْ تَكْتَبُونَ ﴿٣٥﴾ (سورة التوبة: ٣٥)

Artinya: “(Ingatlah) pada hari ketika emas dan perak dipanaskan dalam neraka Jahanam, lalu dengan itu disetrika dahi, lambung dan punggung mereka (seraya

dikatakan) kepada mereka, "Inilah harta bendamu yang kamu simpan untuk dirimu sendiri, maka rasakanlah (akibat dari) apa yang kamu simpan itu" (Q.S. At-Taubah: 35).

Dari ayat di atas, dijelaskan bahwa orang-orang yang mengumpulkan harta dan menyimpannya tanpa sebagian diinfakkan di jalan Allah (dibayarkan zakat), mereka akan dimasukkan ke dalam neraka. Hal ini berkaitan dengan penyebab inflasi yaitu melakukan penimbunan barang disaat persediaan barang semakin menipis karena tingkat konsumsi masyarakat yang tinggi. Apabila mereka melakukan penimbunan hanya untuk kepentingan dan keuntungan pribadi bahkan hingga menyebabkan perekonomian tidak merata maka mereka merupakan orang-orang yang akan mendapatkan hukuman di akhirat nanti.

Dalam kitab Shahih Muslim, pada hadis nomor 3012, telah dijelaskan mengenai keharaman menimbun bahan makanan.

حَدَّثَنَا عَبْدُ اللَّهِ بْنُ مَسْلَمَةَ بْنِ قَعْنَبٍ حَدَّثَنَا سُلَيْمَانُ بْنُ يَحْيَى ابْنُ بِلَالٍ عَنْ يَحْيَى وَهُوَ ابْنُ سَعِيدٍ قَالَ  
كَانَ سَعِيدُ بْنُ الْمُسَيَّبِ يُحَدِّثُ أَنَّ مَعْمَرًا قَالَ قَالَ رَسُولُ اللَّهِ صَلَّى اللَّهُ عَلَيْهِ وَسَلَّمَ مَنْ احْتَكَرَ فَهُوَ  
خَاطِئٌ - رواه مسلم ( ٣٠١٢ )

Artinya: "Telah menceritakan kepada kami Abdullah bin Maslamah bin Qa'nab telah menceritakan kepada kami Sulaiman -yaitu Ibnu Bilal- dari Yahya -yaitu Ibnu Sa'id- dia berkata, " Sa'id bin Musayyab menceritakan bahwa Ma'mar berkata, "Rasulullah shallallahu 'alaihi wasallam bersabda: "Barangsiapa menimbun barang, maka dia berdosa" (HR. Shahih Muslim, 3012).

Hadis di atas menerangkan bahwa Rasulullah melarang untuk melakukan penimbunan barang karena perbuatan tersebut merupakan dosa. Dengan demikian, melakukan penimbunan barang/jasa karena sifat konsumtif yang berlebihan hingga mengakibatkan meningkatnya inflasi juga merupakan perbuatan dosa.

Dari sumber berita VIVA Malang, di sepanjang bulan Desember 2022, pihak BPS Kota Malang kembali mencatat angka inflasi sebesar 0,58%. Salah satu pemicu naiknya angka inflasi ini yaitu dengan meningkatnya harga beras, yang memberi andil sebesar 0,063% terhadap inflasi. Selain itu, komoditas lain seperti telur ayam, cabai rawit, rokok, tempe, bawang merah, dan minyak goreng juga memberi dampak tersendiri bagi kenaikan inflasi di Kota Malang. Dengan kenaikan inflasi di Kota Malang, juga mengakibatkan meningkatnya tingkat inflasi tahunan di Indonesia.

Pada salah satu artikel berita di media CNBC Indonesia, dikatakan bahwa pengendalian tingkat inflasi di Indonesia masih menjadi tantangan bagi bank sentral dan otoritas. Hal ini disampaikan pada saat rapat Koordinasi Nasional Kepala Daerah dan FORKOPIMDA Tahun 2023 oleh Gubernur BI Perry Warjiyo. Dalam rapat tersebut, Perry juga menyampaikan bahwa pengendalian tingkat inflasi di daerah merupakan pondasi. Oleh sebab itu, pihak kementerian, lembaga, dan pemerintah daerah diharapkan dapat mengoptimalkan anggarannya.

Dari penelitian sebelumnya, oleh Kelvin Wong, dkk. (2019), proses prediksi inflasi di Kota Samarinda menggunakan algoritma *backpropagation* dengan pengukuran akurasi menggunakan metode *Mean Square Error* (MSE) dan arsitektur 5-5-5-1, diperoleh hasil *error* sebesar 0,00000424. Kemudian pada tahun 2021, penelitian oleh B Hauriza, dkk. (2021), tentang prediksi inflasi di Indonesia dengan menggunakan jaringan saraf tiruan, mendapatkan hasil *error* MSE sebesar 0,026 dengan arsitektur 7-15-1. Pada penelitian Desty Rakhmawati, dkk. (2023), menggunakan model *Holt-winters exponential smoothing additive*

dan *multiplicative* untuk memprediksi tingkat inflasi Indonesia tahun 2023, diperoleh hasil MAPE sebesar 11,584%.

Dari hasil penelitian sebelumnya yang mendapatkan tingkat keakuratan tinggi, maka dalam penelitian ini, akan digunakan juga algoritma *backpropagation* untuk memprediksi tingkat inflasi di Kota Malang dengan data yaitu data inflasi Kota Malang dari tahun 2005 sampai tahun 2019. Kemudian hasil dari penelitian ini diharapkan bisa membantu pemerintah Kota Malang untuk mengambil langkah yang tepat apabila inflasi terus meningkat, sehingga pemerintah bisa mempersiapkan program yang akan dilakukan untuk mengatasi inflasi yang terjadi dan juga perkembangan yang dilakukan agar tingkat inflasi kembali ke angka sasaran.

## **1.2 Rumusan Masalah**

Berdasarkan penjelasan di atas, maka dapat diambil rumusan masalah pada penelitian ini yaitu sebagai berikut:

1. Bagaimana tingkat akurasi dari algoritma *backpropagation* dalam memprediksi tingkat inflasi di Kota Malang?
2. Bagaimana hasil prediksi inflasi di Kota Malang pada tahun 2019 dengan menggunakan algoritma *backpropagation*?

## **1.3 Tujuan Penelitian**

Berdasarkan uraian rumusan masalah di atas, maka tujuan dari penelitian ini yaitu sebagai berikut:

1. Untuk mengetahui tingkat akurasi dari algoritma *backpropagation* dalam memprediksi tingkat inflasi di Kota Malang.
2. Untuk mengetahui hasil prediksi inflasi di Kota Malang pada tahun 2019 dengan menggunakan algoritma *backpropagation*.

#### **1.4 Manfaat Penelitian**

Pada penelitian ini terdapat beberapa manfaat yang bisa diambil yaitu sebagai berikut:

1. Manfaat Teoritis
  - a. Mampu merancang model arsitektur algoritma *backpropagation* untuk memprediksi inflasi di Kota Malang.
  - b. Mampu mengetahui proses dan kinerja algoritma *backpropagation* untuk memprediksi inflasi di Kota Malang.
  - c. Manfaat teoritis dari penelitian yaitu bagi pembaca dan peneliti yaitu untuk menambah wawasan pembaca mengenai inflasi, tingkat inflasi dan prediksinya, serta sebagai tugas akhir peneliti untuk syarat kelulusan.
2. Manfaat Praktis
  - a. Bagi pemerintah Kota Malang, algoritma ini bisa digunakan untuk mengetahui prediksi inflasi di Kota Malang pada bulan berikutnya, sehingga bisa membantu mempersiapkan rencana yang akan dilakukan jika hasil prediksi tingkat inflasi mengalami kenaikan.



- b. Bagi Program Studi, algoritma ini dapat digunakan sebagai wawasan ilmu pengetahuan dan bahan pembelajaran bagi mahasiswa *backpropagation*.
- c. Bagi pembaca, algoritma ini dapat memberikan informasi dan referensi terkait penerapan algoritma *backpropagation* serta penerapannya dalam memprediksi inflasi.

### **1.5 Batasan Masalah**

Batasan masalah yang digunakan pada penelitian ini yaitu sebagai berikut:

1. Fitur yang digunakan dalam penelitian ini adalah tujuh kelompok inflasi, dan satu kelompok inflasi umum di Kota Malang.
2. Data yang dikumpulkan dan digunakan berupa data runtun waktu (*time series*) yang kontinyu.
3. Data yang digunakan hanya data dari tahun 2005 sampai 2019. Karena untuk data pada tahun 2020 sampai saat ini terdapat pembaruan kelompok, sehingga fitur atau kelompok inflasi yang digunakan akan berbeda dan tidak bisa dilakukan prediksi bersama data tahun sebelumnya.

### **1.6 Istilah-istilah**

Terdapat beberapa istilah yang digunakan dalam penelitian ini yaitu:

1. Implementasi

Implementasi adalah suatu proses yang diterapkan tanpa ada batasan di berbagai bidang, mulai dari pendidikan, kemasyarakatan, politik, teknologi, kesehatan, informasi dan lain-lain.

## 2. Algoritma *Backpropagation*

Algoritma *Backpropagation* adalah algoritma yang digunakan untuk mengoptimalkan bobot dalam jaringan saraf tiruan. Algoritma ini bekerja dengan cara menghitung kesalahan *output* dan menyebar kesalahan tersebut ke *layer* sebelumnya hingga ke *input layer*.

## 3. Prediksi

Prediksi adalah suatu proses memperkirakan secara sistematis tentang sesuatu yang paling mungkin terjadi di masa depan berdasarkan informasi masa lalu dan sekarang yang dimiliki, agar kesalahannya (selisih antara sesuatu yang terjadi dengan hasil perkiraan) dapat diperkecil.

## 4. Inflasi

Inflasi adalah kenaikan harga barang dan jasa secara umum dan terus menerus dalam jangka waktu tertentu.

## 5. Akurasi

Akurasi adalah ketepatan atau kecermatan. Dalam konteks umum, akurasi merujuk pada sejauh mana hasil pengukuran atau pengamatan mendekati nilai sebenarnya atau nilai yang diharapkan.

## BAB II

### KAJIAN TEORI

#### 2.1 *Machine learning* (ML)

Kecerdasan buatan atau *Artificial Intelligence* (AI) merupakan salah satu bagian ilmu komputer yang memungkinkan mesin (komputer) melakukan tugas yang sama dan sebaik manusia. Pada awal diciptakannya, komputer hanya digunakan untuk menghitung, tetapi seiring berjalannya waktu, peran komputer semakin penting bagi kehidupan manusia. Komputer tidak lagi hanya digunakan sebagai alat hitung, mereka diharapkan dapat melakukan apa pun yang dapat dilakukan oleh manusia. Dengan kata lain AI merupakan sistem komputer yang bisa melakukan pekerjaan-pekerjaan yang umumnya memerlukan tenaga manusia atau kecerdasan manusia untuk menyelesaikan pekerjaan tersebut.

AI sendiri merupakan teknologi yang memerlukan data untuk dijadikan pengetahuan, sama seperti manusia. AI membutuhkan pengalaman dan data supaya kecerdasannya bisa lebih baik lagi. Poin penting dalam proses AI adalah *learning*, *reasoning* dan *self correction*. AI perlu belajar untuk memperkaya pengetahuannya. Proses belajar AI pun tidak selalu disuruh oleh manusia, melainkan AI akan belajar dengan sendirinya berdasarkan pengalaman AI saat digunakan oleh manusia.

AI memiliki ruang lingkup yang sangat luas, sehingga AI terbagi menjadi tujuh cabang untuk mempersempit ruang lingkup saat pengembangan atau belajar AI. Ketujuh cabang tersebut yaitu *machine learning* (ML), *natural language processing* (NLP), *expert system*, *vision*, *speech*, *planning* dan *robotics*.

Salah satu cabang dari AI, yaitu *machine learning* (ML), adalah mesin yang dikembangkan untuk bisa belajar dengan sendirinya tanpa arahan dari penggunanya. Pembelajaran mesin dikembangkan berdasarkan disiplin ilmu lainnya seperti statistika, matematika dan data mining sehingga mesin dapat belajar dengan menganalisa data tanpa perlu di program ulang atau diperintah.

Istilah *machine learning* pertama kali dikemukakan oleh beberapa ilmuwan matematika seperti Adrien Marie Legendre, Thomas Bayes dan Andrey Markov pada tahun 1920-an dengan mengemukakan dasar-dasar *machine learning* dan konsepnya. Sejak saat itu ML banyak yang mengembangkan. Salah satu contoh dari penerapan ML yang cukup terkenal adalah *Deep Blue* yang dibuat oleh *International Business Machines Corporation* (IBM) pada tahun 1996.

*Machine learning* adalah tentang mengekstraksi pengetahuan dari data. Penerapan ML dalam beberapa tahun terakhir menjadi di mana-mana dalam kehidupan sehari-hari. Dari rekomendasi film apa yang harus ditonton, makanan apa yang dipesan atau yang mana produk untuk dibeli, dan mengenali teman di foto. Banyak situs web dan perangkat modern memiliki algoritma pembelajaran ini seperti *Facebook*, *Amazon*, atau *Netflix*, itu adalah kemungkinan besar setiap bagian situs berisi beberapa model pembelajaran ML. (Guido, 2017)

ML memiliki setidaknya dua fungsi utama: klasifikasi dan prediksi. Salah satu karakteristik ML adalah adanya proses pelatihan, pembelajaran, atau pelatihan. Akibatnya, ML membutuhkan data untuk dipelajari, yang dikenal sebagai data pelatihan. Klasifikasi adalah metode dalam ML yang digunakan oleh mesin untuk memilah atau mengklasifikasikan obyek berdasarkan ciri tertentu sebagaimana manusia mencoba membedakan benda satu dengan yang lain.

Sedangkan prediksi atau regresi digunakan oleh mesin untuk menerka keluaran dari suatu data masukan berdasarkan data yang sudah dipelajari dalam training. Metode ML yang paling populer yaitu Sistem Pengambil Keputusan, *Support Vector Machine* (SVM) dan *Neural Network*.

Matematika menjadi komponen penting yang mendukung pengembangan AI untuk mengatasi tantangan dan memanfaatkan peluang di era modern. Matematika adalah bahasa universal yang memungkinkan kita menganalisis, memahami, dan memodelkan dunia digital yang semakin kompleks. Matematika juga membantu AI mengubah data mentah menjadi wawasan yang berharga dalam pemrosesan data. Teori probabilitas, statistik, dan analisis matematis digunakan untuk mendapatkan makna dari data yang sangat besar. Ini memberikan dasar untuk pengambilan keputusan yang cerdas yang bergantung pada bukti.

## **2.2 Jaringan Saraf Tiruan (JST)**

*Artificial Neural Network* atau Jaringan Saraf Tiruan (JST) adalah teknik dalam ML yang menirukan saraf manusia yang merupakan bagian fundamental dari otak. JST adalah jaringan sekelompok unit pemroses kecil yang dimodelkan berdasarkan sistem adaptif yang dapat mengubah strukturnya dalam memecahkan masalah.

Menurut Simon Haykin (1999), sebuah jaringan saraf adalah sebuah prosesor didistribusikan massal paralel yang terdiri dari pengolahan sederhana unit, yang mana juga memiliki kecenderungan alami untuk menyimpan pengalaman pengetahuan dan membuatnya tersedia untuk digunakan.

JST bekerja meniru kerja otak manusia dari sudut: 1) Pengetahuan yang diperoleh *network* dari lingkungan, melalui suatu proses pembelajaran. 2) Kekuatan koneksi antar unit yang disebut *synaptic weight*, berfungsi untuk menyimpan pengetahuan yang telah diperoleh dari jaringan tersebut. Karakteristik lain yang penting dari JST, yang sama dengan sistem saraf biologis adalah toleransi kesalahan.

JST biasanya tersusun dari elemen-elemen pada lapisan-lapisan yang terhubung dan diberi bobot. Jaringan ini memodifikasi bobot tersebut berdasarkan serangkaian *input* yang diberikan dari luar sistem tersebut, sehingga menghasilkan *output* yang konsisten dan serupa dengan *input* yang diberikan. Biasanya, setiap elemen akan memproses berdasarkan operasi matematika yang telah diberikan pada setiap elemen.

JST terdiri atas lapisan masukan (*input layer*) dan lapisan keluaran (*output layer*). Setiap lapisan terdiri atas satu atau beberapa unit neuron yang mempunyai sebuah fungsi aktivasi yang menentukan keluaran dari unit tersebut. Lapisan tersembunyi (*hidden layer*) dapat ditambahkan untuk menambah kemampuan dari JST tersebut. JST bisa dilatih dengan menggunakan data *training*.

Pada tahun 1943, Warren McCulloch dan Walter Pitts mengemukakan tiga hal: pengetahuan fisiologi dasar dan fungsi sel saraf dalam otak, analisis formal tentang logika proporsi (*propositional logic*), dan teori komputasi turing. Mereka juga berhasil membuat suatu model saraf tiruan (*artificial neuron*) di mana setiap neuron digambarkan sebagai *on* dan *off*. Hal ini menunjukkan bahwa setiap fungsi dapat dihitung dengan suatu jaringan sel saraf dan bahwa semua hubungan logis dapat diimplementasikan dengan struktur jaringan yang sederhana.

Setelah buku "*Perceptrons*" Minsky dan Papert diterbitkan, bidang ilmu komputer menolak jaringan saraf tiruan. Namun, para ilmuwan masih mempelajari bidang ilmu tersebut dari sudut pandang yang lain. Para ahli, seperti Hopfield (1982), memeriksa karakteristik penyimpanan dan optimasi jaringan saraf dengan menggunakan metode mekanika statistika. David Rumelhart dan Geoff Hinton, melanjutkan penelitian mengenai model saraf pada memori. Serta terdapat setidaknya empat tim penelitian menemukan kembali algoritma belajar propagasi balik (*Backpropagation Learning*) pada tahun 1985-an.

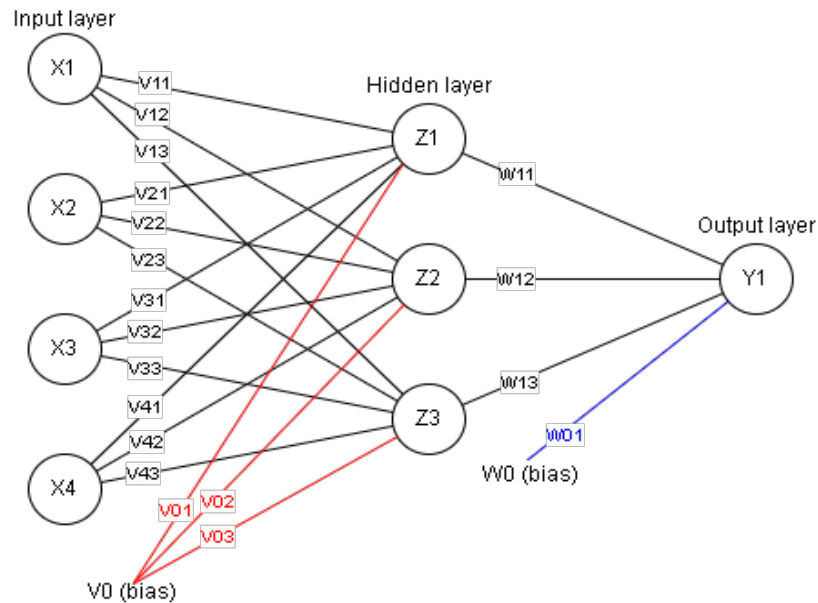
## **2.3 *Backpropagation***

### **2.3.1 Pengertian *Backpropagation***

*Backpropagation* adalah suatu model dari Jaringan Saraf Tiruan (JST). Model *backpropagation* ini menggunakan pembelajaran terawasi untuk memecahkan masalah yang kompleks dan dilatih menggunakan metode pembelajaran (Matondang, 2013). Metode tersebut terdiri dari proses pembelajaran maju dan koreksi kesalahan mundur. Model jaringan *backpropagation* banyak digunakan dalam proses prediksi, identifikasi dan peramalan (Candra Dewi, 2013).

*Backpropagation* pertama kali diterbitkan oleh Paul Werbos pada tahun 1974 memperkenalkan untuk melatih perceptron dengan banyak lapisan. Kemudian dirumuskan ulang oleh David Parker pada tahun 1982, dan dipopulerkan oleh Rumelhar dan McClland pada tahun 1986.

### 2.3.2 Komponen *Backpropagation*



**Gambar 2.1** Arsitektur Jaringan *Backpropagation* 4-3-1, yaitu Empat Neuron di *Input Layer* Tiga Neuron di *Hidden Layer* dan Satu Neuron di *Output Layer*

Berdasarkan Gambar 2.1, dapat dilihat bahwa arsitektur algoritma *backpropagation* memiliki beberapa komponen yaitu:

1. Lapisan Masukan (*Input Layer*)

*Input layer* yaitu *layer* atau lapisan yang berisi neuron-neuron masukan bagi *JST backpropagation*. Banyaknya neuron yang digunakan disesuaikan dengan banyaknya fitur data yang akan diproses. Neuron *input layer* pada arsitektur *JST backpropagation* disimbolkan dengan  $X_i$  dengan  $i = 1, 2, \dots, n$  dimana  $n$  yaitu banyak neuron di *input layer*.

Banyaknya neuron di *input layer* sesuai dengan banyaknya fitur atau kelompok pada data yang akan digunakan, jika terdapat tujuh fitur atau kelompok pada data masukan, maka jumlah neuron di *input layer* yaitu sebanyak tujuh neuron.



## 2. Lapisan Tersembunyi (*Hidden Layer*)

*Hidden layer* merupakan *layer* atau lapisan yang menjadi penghubung antara *input layer* dengan *output layer*. Lapisan ini berisi neuron-neuron *hidden layer* yang disimbolkan dengan  $Z_j$  dengan  $j = 1, 2, \dots, m$  dimana  $m$  adalah banyaknya neuron di *hidden layer*.

Menurut Heaton (2008), terdapat beberapa metode untuk menentukan banyak neuron di *hidden layer* yaitu sebagai berikut:

- a. Banyak neuron tersembunyi harus berada di antara ukuran *input layer* dan ukuran *output layer*.
- b. Banyak neuron tersembunyi harus  $\frac{2}{3}$  dari ukuran *input layer*, ditambah *output layer*.
- c. Banyak neuron tersembunyi harus kurang dari dua kali ukuran *input layer*

Jika banyak neuron yang dipilih lebih sedikit, hal ini akan menyebabkan *underfitting* dan bias statistik yang tinggi. Sedangkan jika kita memilih terlalu banyak neuron, hal ini dapat menyebabkan *overfitting*, varians tinggi, dan meningkatkan waktu yang diperlukan untuk melatih jaringan.

## 3. Lapisan Keluaran (*Output Layer*)

*Output layer* merupakan *layer* atau lapisan yang berisi neuron-neuron untuk keluaran JST *Backpropagation*. Neuron-neuron ini disimbolkan dengan  $Y_k$  yaitu dengan  $k = 1, 2, \dots, p$  dimana  $p$  adalah banyaknya neuron di *output layer*.

Banyaknya neuron di *output layer* tergantung pada jenis masalah yang ingin dipecahkan. Misalnya, jika masalah yang ingin dipecahkan adalah

klasifikasi gambar menjadi 10 kategori, maka *output layer* akan terdiri dari 10 neuron.

#### 4. Bobot V

Untuk setiap neuron pada *input layer* dan *hidden layer* akan dihubungkan dengan bobot dan bias. Bobot V yaitu bobot yang menjadi penghubung antara neuron-neuron pada *input layer* dan *hidden layer*. Bias merupakan komponen yang berfungsi untuk mempercepat laju pembelajaran, bobot boleh diabaikan atau ditiadakan. Bias V biasanya disimbolkan dengan  $V_0$ . Banyaknya bobot V tergantung pada banyak neuron di *input layer* dan *hidden layer* yaitu.

$$\text{bobot } V = n_{input} \times n_{hidden}$$

Apabila menggunakan bias, maka banyaknya bobot V yaitu

$$\text{bobot } V = (n_{input} + 1) \times n_{hidden}$$

Keterangan:

$n_{input}$  : Banyak neuron pada *input layer*

$n_{hidden}$  : Banyak neuron pada *hidden layer*

#### 5. Bobot W

Bobot W yaitu bobot yang menjadi penghubung antara neuron-neuron yang terdapat pada *hidden layer* dan *output layer*. Bias W biasanya disimbolkan dengan  $W_0$ .

Banyaknya bobot W tergantung pada banyak neuron di *input layer* dan *hidden layer* yaitu.

$$\text{bobot } W = n_{hidden} \times n_{output}$$

Apabila menggunakan bias, maka banyaknya bobot  $V$  yaitu

$$\text{bobot } W = (n_{\text{hidden}} + 1) \times n_{\text{output}}$$

Keterangan:

$n_{\text{hidden}}$  : Banyak neuron pada *hidden layer*

$n_{\text{output}}$  : Banyak neuron pada *output layer*

### 2.3.3 Algoritma *Backpropagation*

Algoritma *backpropagation* adalah algoritma yang digunakan untuk melatih jaringan saraf tiruan, khususnya dalam menghitung gradien dari fungsi kerugian terhadap bobot jaringan, yang kemudian digunakan untuk memperbarui bobot tersebut. Algoritma yang digunakan yaitu dengan menyesuaikan bobot berdasarkan selisih keluaran dan target yang diinginkan untuk mengurangi nilai *error*. Kesalahan atau perbedaan antara nilai target dengan data sebenarnya akan ditanggung oleh neuron masukan. Bobot tersebut kemudian dihitung kembali, neuron mengarahkan sinyal ke *hidden layer* neuron keluaran, dan proses penghitungan *error* antara target dan data aktual diulangi lagi hingga *error* mencapai batas yang ditentukan di awal.

Penentuan bobot dilakukan dengan memilih angka *random* dengan rentang dari 0 dan 1, atau -1 dan 1. Angka tersebut dipilih dari rentang yang kecil agar nilai bobot awal tidak terlalu besar, sehingga membantu dalam memastikan bahwa jaringan neural mulai pelatihan dari kondisi yang lebih seimbang. Ini bisa mempercepat konvergensi algoritma pembelajaran. Ketika bobot awal terlalu besar, jaringan bisa terjebak dalam pola pembelajaran yang tidak efisien.

Algoritma *backpropagation* dapat digunakan untuk menyelesaikan permasalahan yang rumit dan kompleks, yaitu dengan menggunakan pelatihan *multi-layer perceptron* (MLP) untuk proses pelatihan *input* dan *output* dengan tujuan menghitung bobot untuk mencapai hasil yang diinginkan. Algoritma ini menggunakan lebih sedikit memori dibandingkan dengan algoritma lainnya. Algoritma ini menerima suatu nilai masukan dan melakukan perhitungan berdasarkan bobot yang diperoleh secara acak yang diterima di awal. Apabila nilai keluaran masih belum sesuai dengan nilai target yang diinginkan, maka bobot yang ada pada saat itu disesuaikan. Proses ini berlanjut hingga nilai keluaran sesuai dengan nilai target yang diinginkan. Proses pembelajaran juga terhenti jika keluaran dan target mencapai nilai bobot atau tingkat kesalahan atau nilai toleransi lebih kecil.

Algoritma *backpropagation* menggunakan fungsi aktivasi pada *hidden layer* dengan tujuan untuk mengurangi nilai *error* dari *output* yang dihasilkan neuron. Fungsi sigmoid, *Tangen hyperbolicus* (*Tanh*), dan *Rectified Linear Unit* (ReLU) adalah beberapa fungsi aktivasi yang paling umum digunakan pada algoritma *backpropagation*. Dalam penelitian ini fungsi aktivasi yang akan digunakan yaitu ReLU dan *Tanh*, karena fungsi ReLU dan *Tanh* mampu memecahkan masalah non-linier, serta memerlukan sumber daya komputasi yang jauh lebih sedikit. Berikut yaitu rumus dari fungsi ReLU dan *Tanh*.

1. ReLU

$$f(x) = \max(0, x) \quad (2.1)$$

2. *Tanh*

$$f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (2.2)$$

Keterangan:

$f(x)$  : Hasil aktivasi

$x$  : Nilai *input*

Pada algoritma *backpropagation* juga digunakan fungsi optimisasi untuk memperbarui bobot secara iteratif yang didasarkan pada data *training*. Tujuan dari fungsi optimisasi yaitu untuk meminimalkan kerugian dengan proses *training*. Terdapat beberapa fungsi optimisasi yang biasa digunakan pada *neural network* yaitu *Stochastic Gradient Descent* (SGD), dan *Adapting Moment Estimation* (Adam). Pada penelitian ini akan menggunakan fungsi optimisasi Adam karena fungsi ini merupakan kombinasi antara *RMSprop* dan SGD, sehingga hasil dari fungsi ini lebih baik dibanding dengan kedua fungsi sebelumnya. Pada umumnya nilai Adam yang digunakan yaitu nilai default sebesar 0,0001

#### 2.3.4 Tahapan *Backpropagation*

Terdapat beberapa tahap untuk menggunakan algoritma *backpropagation* yaitu (R Sovia, 2018).

1. Tahap inisialisasi

Langkah ini merupakan langkah untuk menentukan nilai awal bagi variabel yang dibutuhkan.

2. Tahap aktivasi

Menghitung keluaran aktual pada *hidden layer* dan keluaran aktual pada *output layer* adalah dua tugas yang dilakukan pada langkah ini.

### 3. Tahap *weight training*

Pada langkah ini, dua tugas dilakukan yaitu menghitung *gradient error* pada *output layer* dan menghitung *gradient error* pada *hidden layer*.

### 4. Tahap iterasi

Pada langkah terakhir ini, pengujian dilakukan. Jika tidak ditemukan kesalahan, maka akan kembali ke tahap kedua yaitu aktivasi.

Hamdan Wadi (2021) menyatakan bahwa proses prediksi data dengan algoritma *backpropagation* memerlukan dua tahap: pelatihan dan pengujian.

#### 1. Proses pelatihan

Tujuan dari proses pelatihan adalah untuk mengenali pola pada data dengan melatih bobot agar dapat mengenalinya. Proses ini menghasilkan bias yang terlatih, bobot  $V$ , dan bobot  $W$ . Dalam proses ini terdapat dua tahap yaitu propagasi maju dan propagasi mundur. Proses pelatihan diartikan sebagai perulangan terhadap propagasi maju dan propagasi mundur dengan tujuan memperoleh bobot  $V$ , bobot  $W$ , dan bias terlatih.

#### 2. Proses pengujian

Pengujian adalah proses klasifikasi dan prediksi data uji atau data yang akan datang. Dalam proses ini, algoritma *backpropagation* hanya melakukan propagasi maju.

Tahapan *backpropagation*, menurut Andrian dan Wahyudi (2014), adalah sebagai berikut

#### **Langkah 1:**

Menentukan *epoch*, target *error*, tingkat pembelajaran, dan inisialisasi bobot *random*

**Langkah 2:**

Inisialisasi kondisi berhenti. Kondisi berhenti dimulai dengan iterasi (*epoch* < *epoch maksimum*) atau *minimum error*.

**Langkah 3:**

Melakukan langkah 4-10 pada setiap data *training*.

**Tahap I : Feed Forward****Langkah 4:**

Sinyal *input*  $X_i$  diterima oleh setiap neuron *input*  $X_i$  dengan  $i = 1, 2, 3, \dots, n$  dan didistribusikan ke neuron *hidden layer*.

**Langkah 5:**

Sinyal *input* terbobot dan biasanya dijumlahkan oleh setiap neuron tersembunyi  $Z_j$  dengan  $j = 1, 2, 3, \dots, m$  dengan menggunakan Persamaan (2.3).

$$Z_{input_j} = v_{0_j} + \sum_{i=1}^n X_i V_{ij} \quad (2.3)$$

Keterangan:

$Z_{input_j}$  : Sinyal informasi dari *input layer* ke neuron *hidden layer* ke- $j$

$V_{0_j}$  : Bias pada *hidden layer* ke- $j$

$X_i$  : Neuron *input layer* ke- $i$

$V_{ij}$  : Bobot yang menghubungkan neuron *input layer* ke- $i$  dan neuron *hidden layer* ke- $j$

Dengan menggunakan fungsi aktivasi, dihitung sinyal *output*-nya mengikuti rumus pada Persamaan (2.5) berikut.

$$Z_j = f(z_{input_j}) \quad (2.4)$$

Keterangan:

$Z_j$  : Neuron *hidden layer* ke- $j$

$f(Z_{input_j})$  : Fungsi aktivasi terhadap nilai  $Z_{input_j}$

Unit *output* kemudian menerima sinyal dari seluruh neuron *input*.

### Langkah 6:

Unit  $Y_k$ , yang memiliki nilai  $k = 1, 2, 3, \dots, p$ , menerima sinyal *output* dari *hidden layer*, dan kemudian menjumlahkan sinyal *output* yang terbobot dan biasanya dengan menggunakan Persamaan (2.5).

$$Y_{input_k} = W_{0_k} + \sum_{j=1}^n Z_j W_{jk} \quad (2.5)$$

Keterangan:

$Y_{input_k}$  : Sinyal informasi dari *hidden layer* ke neuron *output layer* ke- $k$

$Z_j$  : Neuron *hidden layer* ke- $j$

$W_{0_k}$  : Bias pada neuron *output layer* ke- $k$

$W_{jk}$  : Bobot yang menghubungkan neuron *hidden layer* ke- $j$  dan neuron *output layer* ke- $k$

Hitung sinyal *output* dengan menggunakan fungsi aktivasi yang telah ditentukan mengikuti rumus dari Persamaan (2.6) berikut.

$$Y_{output_k} = f(Y_{input_k}) \quad (2.6)$$

Keterangan:

$Y_{output_k}$  : Keluaran dari nilai  $Y_{input_k}$

$f(Y_{input_k})$  : Fungsi aktivasi terhadap nilai  $Y_{input_k}$

Setelah itu, sinyal yang dihasilkan ditransmisikan ke seluruh unit di *layer* atasnya.



## Tahap II: *Feed Backward*

### Langkah 7:

Untuk menghitung kesalahan *error* pada *output unit*, setiap unit  $Y_{output_k}$  dengan  $k = 1, 2, 3, \dots, p$  menerima *output* yang diinginkan, yang sesuai dengan *input* data pelatihan dengan menggunakan rumus pada Persamaan (2.7).

$$\delta_k = (t_k - Y_{output_k})Y_{output_k}(1 - Y_{output_k}) \quad (2.7)$$

Dengan:

$\delta_k$  : Faktor koreksi dari neuron *output layer* ke- $k$

$t_k$  : Target *output* pada neuron *output layer* ke- $k$

$Y_{output_k}$  : Keluaran pada *output unit*

Kemudian hitung faktor koreksi bobot (yang akan digunakan untuk memperbarui bobot baru). Faktor *error*  $\delta_k$  digunakan untuk mengoreksi nilai *error* pada bobot antara unit tersembunyi dan unit keluaran  $\Delta W_{jk}$  yang nantinya digunakan untuk memperbarui bobot  $W_{jk}$  dengan menggunakan Persamaan (2.8).

$$\Delta W_{jk} = \alpha \delta_k Z_j \quad (2.8)$$

Dengan:

$\Delta W_{jk}$  : Koreksi bobot yang menghubungkan antara neuron *output layer* ke- $k$  dan neuron *hidden layer* ke- $j$

$\alpha$  : Tingkat pembelajaran (*learning rate*)

### Langkah 8:

Setiap unit tersembunyi  $Z_j$  dengan  $j = 1, 2, 3, \dots, m$  menjumlahkan *input* delta yang sudah terbobot seperti pada Persamaan (2.9).

$$\delta_{input_j} = \sum_{k=1}^m \delta_k W_{jk} \quad (2.9)$$

Keterangan:

$\delta_{input_j}$  : Sinyal faktor koreksi dari *output layer* ke neuron *hidden layer* ke- $j$

Selanjutnya, hasil delta *input* dikalikan dengan turunan fungsi aktivasi yang telah ditentukan dan menghasilkan faktor koreksi *error*  $\delta_j$  seperti pada Persamaan (2.10) berikut.

$$\delta_j = \delta_{input_j} Z_j (1 - Z_j) \quad (2.10)$$

Keterangan:

$\delta_j$  : Faktor koreksi dari neuron *hidden layer* ke- $j$

Hitung koreksi *error*  $\Delta V_{ji}$  yang nantinya akan digunakan untuk memperbarui  $V_{ji}$  dengan menggunakan rumus pada Persamaan (2.11)

$$\Delta V_{ji} = \alpha \delta_j X_i \quad (2.11)$$

Keterangan:

$\Delta V_{ji}$  : Koreksi bobot yang menghubungkan antara neuron *hidden layer* ke- $j$  dan neuron *input layer* ke- $i$

### Tahap III: Perubahan Bobot dan Bias

#### Langkah 9:

Bobot dan bias diperbarui untuk setiap unit *output*  $Y_k$  yang memiliki  $k = 1, 2, 3, \dots, p$  dengan menggunakan rumus pada Persamaan (2.12)

$$W_{jk(\text{baru})} = W_{jk(\text{lama})} + \Delta W_{jk} \quad (2.12)$$

Bobot dan bias diperbarui untuk setiap unit tersembunyi  $Z_j$  ( $j = 1, 2, 3, \dots, m$ ) dengan menggunakan rumus pada Persamaan (2.13) berikut.

$$V_{ij(\text{baru})} = V_{ij(\text{lama})} + \Delta V_{ij} \quad (2.13)$$

**Langkah 10:**

Tes kondisi berhenti. Iterasi akan berhenti ketika hasil keluaran sudah memenuhi target atau nilai *error* dari hasil prediksi dan target sudah melebihi batas inisialisasi *error*.

**2.3.5 Normalisasi Data**

Normalisasi data adalah proses penyesuaian skala nilai data sehingga mereka berada dalam rentang tertentu atau memiliki distribusi tertentu. Ini bertujuan untuk meningkatkan kinerja model pembelajaran mesin dengan memastikan bahwa fitur-fitur memiliki skala yang sama, sehingga menghindari dominasi fitur dengan skala yang lebih besar.

Normalisasi data dilakukan selama tahap pra-pemrosesan sebelum pelatihan model untuk memastikan kinerja yang optimal dari algoritma pembelajaran mesin. Proses ini dilakukan ketika dataset memiliki fitur dengan skala yang sangat berbeda, misalnya umur dalam puluhan atau ratusan tahun dan pendapatan dalam ribuan atau jutaan dolar.

Salah satu cara normalisasi data adalah dengan metode *max-min scaling*. Berikut merupakan rumus normalisasi data dengan rentang dari 0,1 sampai dengan 1.

$$v'_n = 0,1 + \frac{(v_n - v_{min}) \times (1 - 0,1)}{v_{max} - v_{min}} \quad (2.14)$$

Keterangan:

$v_n$  : Nilai asli (belum dinormalisasi)

$v'_n$  : Nilai baru yang telah dinormalisasi

$v_{min}$  : Nilai minimal dari data yang belum dinormalisasi

$v_{max}$  : Nilai maksimal dari data yang belum dinormalisasi

Rentang yang digunakan yaitu dari 0,1 sampai 1 yaitu untuk menghindari adanya nilai nol. Karena apabila data mengandung nilai nol, perhitungan yang melibatkan logaritma (seperti dalam perhitungan error MAPE) bisa menyebabkan kesalahan atau hasil yang tidak terdefinisi.

Kemudian, untuk mengembalikan data hasil normalisasi menjadi data dengan rentang asli maka perlu dilakukan proses denormalisasi dengan rumus sebagai berikut.

$$v_n = \frac{v'_n - 0,1}{1 - 0,1} (v_{max} - v_{min}) + v_{min} \quad (2.15)$$

## 2.4 Inflasi

### 2.4.1 Pengertian Inflasi

Inflasi diartikan sebagai penurunan nilai mata uang dibandingkan dengan nilai barang dan jasa secara keseluruhan. Ini karena inflasi adalah kecenderungan harga barang dan jasa umumnya meningkat secara konsisten, dan ketika nilai inflasi meningkat, harga barang dan jasa dalam negeri juga meningkat (BPS Kota Malang, 2023).

Fenomena ekonomi yang mengganggu ekonomi dunia, termasuk Indonesia, adalah inflasi. Dibandingkan dengan berinvestasi dalam lembaga keuangan atau investasi, masyarakat lebih memilih karir alternatif sebagai akibat dari inflasi. Ini karena inflasi menyebabkan tiga hal: (1) penurunan produktivitas dan efisiensi produksi, (2) kenaikan biaya modal, dan (3) ketidakjelasan mengenai ongkos dan pendapatan di masa depan (Salam, 2020).

Alat keuangan yang dikenal sebagai indeks harga konsumen dapat memberikan informasi tentang bagaimana harga barang dan jasa meningkat di antara konsumen. Perhitungan indeks harga konsumen bertujuan untuk mengetahui bagaimana harga barang dan jasa tertentu telah berubah dari waktu ke waktu. Perubahan indeks harga konsumen menunjukkan laju kenaikan (inflasi) atau penurunan (deflasi) dari harga barang dan jasa yang diperlukan untuk memenuhi kebutuhan sehari-hari rumah tangga. Peningkatan atau penurunan harga barang dan jasa ini terkait erat dengan kemampuan uang masyarakat untuk membeli barang tersebut. Peningkatan inflasi berkorelasi langsung dengan penurunan nilai uang dan daya beli (BPS Kota Malang, 2020).

#### **2.4.2 Penyebab Inflasi**

Berikut merupakan beberapa alasan terjadinya inflasi menurut situs web resmi Bank Indonesia (Bank Indonesia, 2020).

1. Tekanan dari sisi penawaran (*Cost Push Inflation*): Inflasi yang disebabkan oleh peningkatan biaya produksi atau tekanan dari sisi penawaran terjadi. Beberapa penyebabnya adalah sebagai berikut:
  - a. Depresiasi nilai tukar: Jika mata uang suatu negara menurun terhadap mata uang asing, harga impor akan naik, yang meningkatkan biaya produksi dan akhirnya mendorong inflasi.
  - b. Dampak inflasi luar negeri: Inflasi di negara mitra dagang atau di pasar global dapat berdampak pada harga impor, yang pada gilirannya dapat meningkatkan biaya produksi di dalam negeri.

- c. Peningkatan harga komoditas yang diatur pemerintah: Kenaikan harga komoditas yang signifikan dapat menyebabkan biaya produksi yang lebih tinggi.
  - d. Bencana alam dan gangguan distribusi barang dan jasa dapat mengurangi penawaran, yang dapat menyebabkan kenaikan harga.
2. Tekanan dari sisi permintaan (*Demand Pull Inflation*): Terjadi ketika inflasi disebabkan oleh tekanan dari sisi permintaan atau meningkatnya permintaan barang dan jasa relatif terhadap ketersediaannya. Dalam konteks makro ekonomi, kondisi ini digambarkan oleh *output* riil yang melebihi *output* potensialnya atau permintaan total (*aggregate demand*) lebih besar dari pada kapasitas perekonomian hal tersebut dapat mendorong kenaikan harga.
  3. Ekspektasi Inflasi: Ekspektasi inflasi adalah faktor yang dipengaruhi oleh persepsi dan harapan masyarakat serta pelaku ekonomi terhadap tingkat inflasi di masa depan. Faktor ini dapat mempengaruhi keputusan konsumen, investor, dan pelaku ekonomi lainnya. Ada dua jenis ekspektasi inflasi:
    - a. Ekspektasi inflasi adaptif, bergantung pada sejarah atau data historis.
    - b. Ekspektasi inflasi *forward-looking*, memperkirakan faktor-faktor ekonomi dan kebijakan yang mempengaruhi inflasi di masa depan.
- (Bank Indonesia, 2020)

### 2.4.3 Tingkat Inflasi

Tingkat inflasi (inflation rate) merupakan persentase perubahan harga-harga barang dan jasa di dalam perekonomian selama periode tertentu. Jika inflasi menunjukkan tentang suatu keadaan di mana harga barang dan jasa meningkat atau menurun, maka tingkat inflasi menunjukkan berapa persen peningkatan atau penurunan tersebut.

Tingkat inflasi adalah ukuran kuantitatif yang menunjukkan seberapa cepat harga barang dan jasa meningkat dalam suatu periode waktu tertentu, biasanya dinyatakan sebagai persentase per tahun. Tingkat inflasi mengukur laju kenaikan harga dan memberikan angka spesifik yang dapat digunakan untuk analisis ekonomi.

### 2.4.4 Pengukuran Tingkat Inflasi

Menurut Badan Pusat Statistik (2023), cara perhitungan IHK dan inflasi adalah sebagai berikut:

1. Menghitung Indeks Kelompok

$$IHK_{nb} = \frac{NK_{nb}}{NK_{0b}} \times 100 \quad (2.16)$$

Keterangan:

$IHK_{nb}$  : Indeks harga konsumen kelompok  $b$ , periode ke- $n$

$NK_{nb}$  : Nilai konsumsi kelompok  $b$ , periode ke- $n$

$NK_{0b}$  : Nilai Konsumsi dasar kelompok  $b$

2. Inflasi bulanan (*month to month/mtm*)

$$\text{Inflasi} = \frac{IHK_n - IHK_{n-1}}{IHK_{n-1}} \times 100\% \quad (2.17)$$

Keterangan:

$IHK_n$  : Indeks harga konsumen bulan ke- $n$

$IHK_{n-1}$  : Indeks harga konsumen bulan ke- $(n - 1)$

## 2.5 Akurasi

### 2.5.1 Pengertian Akurasi

Akurasi adalah salah satu konsep penting dalam berbagai bidang, termasuk ilmu pengetahuan, matematika, teknologi, dan statistik. Secara umum, akurasi mengacu pada tingkat ketepatan atau kebenaran suatu informasi atau hasil pengukuran. Sedangkan akurasi dalam konteks prediksi dan ML merujuk pada seberapa dekat prediksi model dengan nilai sebenarnya. Ini adalah ukuran kinerja yang penting untuk menilai seberapa baik model dalam memprediksi hasil yang diharapkan.

Akurasi memiliki peranan penting dalam berbagai aspek kehidupan. Dalam bidang ilmu pengetahuan, akurasi adalah salah satu kunci untuk mendapatkan pengetahuan yang benar dan dapat diandalkan. Hasil penelitian yang akurat dapat membantu dalam pengambilan keputusan. Sedangkan dalam bidang teknologi, akurasi menjadi faktor kunci dalam pengembangan sistem dan aplikasi.

### 2.5.2 Perhitungan Akurasi

Untuk mengukur akurasi, sering digunakan istilah error atau kesalahan. Error merupakan perbedaan antara nilai yang diukur atau diestimasi dengan nilai



yang sebenarnya. Terdapat beberapa metode yang digunakan untuk mengukur akurasi salah satunya yaitu *Mean Absolute Percentage Error* (MAPE).

MAPE merupakan sebuah ukuran ketepatan relatif yang digunakan untuk menghitung persentase penyimpangan hasil prediksi. Metode ini memiliki rumus sebagai berikut:

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{x_i - y_i}{x_i} \right| \times 100\% \quad (2.18)$$

Keterangan:

- $x_i$  : Nilai aktual
- $y_i$  : Nilai prediksi
- $n$  : Jumlah data

Dengan menggunakan hasil MAPE maka didapatkan akurasi dengan rumus:

$$Akurasi = 100\% - MAPE \quad (2.19)$$

Kriteria nilai akurasi adalah sebagai berikut:

1.  $Akurasi \geq 90\%$  : sangat baik
2.  $80\% \leq Akurasi < 90\%$  : baik
3.  $70\% \leq Akurasi < 80\%$  : cukup
4.  $Akurasi < 70\%$  : buruk

Dari kriteria-kriteria di atas, maka semakin besar akurasi akan semakin baik dan semakin akurat pula hasil penelitian.

## 2.6 Penelitian Terdahulu

Terdapat beberapa penelitian terbaru yang menjadi acuan dan pendukung dalam penelitian ini yaitu penelitian oleh Kelvin Wong, Aji Prasetya Wibawa, Herman Santoso Pakpahan, Anton Prafanto, dan Hario Jati Setyadi pada tahun 2019 dengan judul “Prediksi Tingkat Inflasi Dengan Menggunakan Metode *Backpropagation Neural Network*”. Penelitian ini menggunakan metode *Backpropagation Neural Network* (BPNN) untuk memprediksi tingkat inflasi. Hasil terbaik dari penelitian ini adalah model arsitektur 5-5-5-1 yaitu lima neuron pada *input layer*, dua *hidden layer* dengan masing masing lapisan terdapat lima neuron, dan satu neuron pada *output layer*. Selain itu parameter pada penelitian tersebut yaitu digunakan fungsi pembelajaran *trainlm*, fungsi aktivasi *logsig* dan *purelin*, serta *learning rate* sebesar 0,1 menghasilkan tingkat kesalahan dalam prediksi yang baik dengan nilai MSE sebesar 0,00000424.

Kemudian pada tahun 2021, penelitian oleh B. Hauriza, Muladi, I M Wirawan dengan judul “Prediksi Tingkat Inflasi Bulanan Indonesia Menggunakan Metode Jaringan Saraf Tiruan” juga menjadi acuan dalam penelitian ini. Hasil dari penelitian tersebut yaitu model arsitektur 7-15-1, 7 neuron pada *input layer*, 15 neuron pada *hidden layer*, dan 1 neuron pada *output layer*. Model tersebut diproses dengan *learning rate* sebesar 0,01 menghasilkan *error* terkecil sebesar 0,026.

Pada penelitian “Prediksi Inflasi di Kota Mataram Menggunakan *Backpropagation Neural Network*” oleh Jihan Khairunnisa dan Syaharuddin tahun 2022, digunakan data inflasi dari tahun 2011 hingga 2018 sebagai data latih dan data dari tahun 2019 sampai 2021 sebagai data uji. Penelitian tersebut

menghasilkan nilai *error* MSE sebesar 0,000317243. Hal ini menyatakan bahwa pada penelitian tersebut akurasi dari model yang digunakan untuk prediksi sangat baik.

## 2.7 Kajian Integrasi Islam

Inflasi merupakan salah satu masalah masyarakat modern yang timbul karena berbagai sebab, termasuk keinginan masyarakat untuk melakukan konsumsi berlebihan. Seperti firman Allah SWT dalam Al-Qur'an Surah Ali Imran ayat 14 (Kemenag, 2019).

زَيْنَ لِلنَّاسِ حُبُّ الشَّهَوَاتِ مِنَ النِّسَاءِ وَالْبَنِينَ وَالْقَنَاطِيرِ الْمُقَنْطَرَةِ مِنَ الذَّهَبِ وَالْفِضَّةِ وَالْخَيْلِ  
الْمُسَوَّمَةِ وَالْأَنْعَامِ وَالْحَرْثِ ذَلِكَ مَتَاعُ الْحَيَاةِ الدُّنْيَا وَاللَّهُ عِنْدَهُ حُسْنُ الْمَبَاقِ ﴿١٤﴾ (سورة آل  
عمران: ١٤)

*Artinya: "Dijadikan indah pada (pandangan) manusia kecintaan kepada apa-apa yang diingini, yaitu: wanita-wanita, anak-anak, harta yang banyak dari jenis emas, perak, kuda pilihan, binatang-binatang ternak dan sawah ladang. Itulah kesenangan hidup di dunia, dan di sisi Allah-lah tempat kembali yang baik (surga)" (Q.S. Ali Imran: 14).*

Menurut tafsir Quraish Shihab pada buku *Tafsir Al-Mishbah* dijelaskan bahwa wajar jika manusia mencintai apa yang diidam-idamkannya, yaitu wanita, anak-anak, banyak emas dan perak, kuda yang bagus dan terlatih, hewan ternak seperti unta, sapi dan domba. Kecintaan ini juga diungkapkan di sawah ladang yang luas. Namun semua ini merupakan kenikmatan hidup di dunia yang fana. Hal ini tidak ada artinya jika dibandingkan dengan rahmat Allah SWT terhadap hamba-hamba-Nya yang berjuang di jalan-Nya ketika mereka kembali kepada-Nya di akhirat nanti (Shihab, 2002).

Selain terlalu mencintai sesuatu hanya bersifat fana, memiliki rasa yang berlebih juga hanya akan memberikan dampak yang buruk, seperti sabda Rasulullah SAW yang mengingatkan kepada manusia bahwa kemewahan dunia itu bahaya yaitu *“Bergembiralah dan renungkanlah apa yang sesungguhnya menggembirakan kamu. Demi Allah! Aku tidak akan mengkhawatirkan kemelaratan yang menimpa kamu. Tetapi yang aku khawatirkan adalah bila kemewahan dunia menimpamu sebagaimana orang-orang yang sebelum kamu ditimpa kemewahan dunia. Lalu kamu berlomba-lomba dengan kemewahan dan kamu binasa oleh mereka.”* serta hadis lain dari riwayat yang sama. *“Sangatlah celaka orang yang diperhamba oleh harta, baik berupa emas, perak dan lainnya”*. (Hadis Riwayat Muslim, 2961). Sama halnya dengan masyarakat yang memiliki sifat konsumtif yang berlebihan ketika persediaan barang/jasa yang dibutuhkan masih atau sedang rendah, maka sifat tersebut bisa mengakibatkan naiknya harga barang/jasa.

Taqyuddin Ahmad Ibn Al-Maqrizi (1364-1441) menyatakan, seperti yang dikutip Euis Amalia dalam bukunya Sejarah Pemikiran Ekonomi Islam dari Masa Klasik hingga Kontemporer, bahwa Inflasi terjadi ketika harga secara umum mengalami kenaikan yang berlangsung secara terus-menerus. Pada saat itu, persediaan barang dan jasa mengalami kelangkaan, sementara konsumen harus mengeluarkan lebih banyak uang untuk sejumlah barang dan jasa yang sama (Amalia, 2005). Menurut Sukirno (2008) inflasi yaitu kenaikan dalam harga barang dan jasa yang terjadi karena permintaan bertambah lebih besar dibandingkan dengan penawaran barang di pasar. Dengan kata lain, terlalu banyak uang yang memburu barang yang terlalu sedikit.

Pada kisah Nabi Yusuf AS di mana saat ia menjadi budak yang kemudian diangkat menjadi anak dari seorang Menteri Keuangan di negara Mesir, dan dipanggil oleh Raja Mesir untuk menafsirkan mimpi raja. Allah berfirman dalam Q.S. Yusuf ayat 46 (Kemenag, 2019).

يُوسُفُ أَيُّهَا الصِّدِّيقُ أَفْتِنَا فِي سَبْعِ بَقَرَاتٍ سِمَانٍ يَأْكُلُهُنَّ سَبْعٌ عِجَافٌ وَسَبْعِ سُنبُلَاتٍ خُضْرٍ وَأُخَرَ  
يَابَسَتْ لَعَلِّي أَرْجِعُ إِلَى النَّاسِ لَعَلَّهُمْ يَعْلَمُونَ ﴿٤٦﴾ (سورة يوسف: ٤٦)

*Artinya:“(Setelah pelayan itu berjumpa dengan Yusuf dia berseru): "Yusuf, hai orang yang amat dipercaya, terangkanlah kepada kami tentang tujuh ekor sapi betina yang gemuk-gemuk yang dimakan oleh tujuh ekor sapi betina yang kurus-kurus dan tujuh bulir (gandum) yang hijau dan (tujuh) lainnya yang kering agar aku kembali kepada orang-orang itu, agar mereka mengetahuinya” (Q.S. Yusuf: 46).*

Ayat di atas merupakan awal mula kisah Nabi Yusuf dan Raja, di mana saat itu Raja Mesir bermimpi aneh yaitu melihat tujuh ekor sapi betina yang gemuk dimakan oleh tujuh ekor sapi kurus dan tujuh bulir (gandum) hijau dan tujuh bulir kering. Pada saat itu, semua ahli tafsir dan ahli sihir tidak ada yang bisa menafsirkan mimpi Raja tersebut, mereka hanya mengatakan jika mimpi-mimpi tersebut hanya mimpi kosong dan tidak bisa ditafsirkan. Kemudian salah satu dari pelayan Raja yang mimpinya pernah ditafsirkan oleh Nabi Yusuf menyarankan untuk mendatangkan Nabi Yusuf untuk menafsirkan mimpinya. Ketika Nabi Yusuf didatangkan ke hadapan Sang Raja, Raja menceritakan semua mimpinya kepada Nabi Yusuf. Kemudian Nabi Yusuf mengatakan bahwa arti tafsir mimpi tersebut adalah negeri Mesir akan mengalami masa subur selama tujuh tahun, tetapi di tahun berikutnya akan mengalami masa kemarau panjang selama tujuh tahun pula. Nabi Yusuf kemudian mengatakan kepada Raja bahwa dia harus menyimpan hasil gandum di musim subur sebagai bekal untuk bertahan

di musim kemarau yang panjang. Dalam Al-Qur'an Surah Yusuf, Ayat 47–49, Allah menggambarkan bagaimana Nabi Yusuf menanggapi Raja Mesir (Kemenag, 2019).

قَالَ تَزْرَعُونَ سَبْعَ سِنِينَ دَأْبًا فَمَا حَصَدْتُمْ فَذَرُوهُ فِي سُنْبُلِهِ إِلَّا قَلِيلًا مِمَّا تَأْكُلُونَ ﴿٤٧﴾ ثُمَّ يَأْتِي مِنْ بَعْدِ ذَلِكَ سَبْعٌ شِدَادٌ يَأْكُلْنَ مَا قَدَّمْتُمْ لَهُنَّ إِلَّا قَلِيلًا مِمَّا تُحْصِنُونَ ﴿٤٨﴾ ثُمَّ يَأْتِي مِنْ بَعْدِ ذَلِكَ عَامٌ فِيهِ يُغَاثُ النَّاسُ وَفِيهِ يَعْرِضُونَ ﴿٤٩﴾ (سورة يوسف: ٤٧-٤٩)

*Artinya: "Yusuf berkata: 'Supaya kamu bertanam tujuh tahun (lamanya) sebagaimana biasa; maka apa yang kamu tuai hendaklah kamu biarkan bulirnya kecuali sedikit untuk kamu makan. Kemudian sesudah itu akan datang tujuh tahun yang amat sulit, yang menghabiskan apa yang kamu simpan untuk menghadapinya (tahun sulit), kecuali sedikit dari (bibit gandum) yang kamu simpan. Kemudian setelah itu akan datang tahun yang padanya manusia diberi hujan (dengan cukup) dan di masa itu mereka memeras anggur'" (Q.S. Yusuf: 47-49).*

Dari kebijaksanaan dan pengetahuan nabi Yusuf itulah yang membuat Sang Raja memutuskan untuk memberi kedudukan tinggi kepadanya. Jabatan yang ditawarkan oleh Raja kepada Nabi Yusuf yaitu untuk mengambil alih pengelolaan selama tahun-tahun yang baik dan selama masa paceklik. Namun Nabi Yusuf menolak tawaran itu apabila kasusnya dengan istri Al-Aziz tidak terselesaikan. Maka saat Nabi Yusuf diadili, ia dinyatakan tidak bersalah dan dibebaskan. Kemudian, raja menjadikan Nabi Yusuf sebagai orang kepercayaan, dan Nabi Yusuf berkata kepada raja,

*"Jika memang engkau percaya kepadaku, jadikanlah aku bendaharawan negara. Sesungguhnya, aku mampu menjaga juga berpengetahuan dalam hal tersebut."* (Q.S. Yusuf 12:55 dalam Kemenag, 2019).

Setelah Nabi Yusuf menjadi perdana menteri Mesir, ia mengambil alih urusan ekonomi dan pangan. Di masa tahun-tahun baik, Nabi Yusuf mengatur

produksi dan penyimpanan hasil bumi. Ketika masa baik berlalu dan memasuki masa paceklik, Nabi Yusuf meningkatkan pendistribusian dengan tetap menjaga stabilitas pasokan pangan. Selama Mesir mengalami kekeringan, Nabi Yusuf memimpin operasi pendistribusian makanan kepada masyarakat. Karena langkah bijaksana dari Nabi Yusuf inilah ia populer karena kemampuannya untuk mengatur kebutuhan rakyat, mendistribusikan pasokan pangan, dan menyelamatkan masyarakat dari keterpurukan.

Dari kisah Nabi Yusuf dalam menghadapi masa-masa paceklik tersebut mengajarkan bahwa boleh menyimpan harta atau sejenisnya sesuai dengan kebutuhan yang wajar serta melakukan pendistribusian dengan adil, anjuran untuk selalu hidup hemat dan larangan untuk berbuat boros.

Selain dari kisah Nabi Yusuf tersebut, terdapat beberapa hadis yang melarang untuk menimbun barang berlebihan, salah satunya yaitu hadis yang diriwayatkan oleh Abu Daud dalam Kitab *Sunan Abi Dawud* pada bab *Kitab Al-Ijarah*, hadis ke 3447.

حَدَّثَنَا وَهْبُ بْنُ بَقِيَّةَ، أَخْبَرَنَا خَالِدٌ، عَنْ عَمْرِو بْنِ يَحْيَى، عَنْ مُحَمَّدِ بْنِ عَمْرٍو بْنِ عَطَاءٍ، عَنْ سَعِيدِ بْنِ الْمُسَيَّبِ، عَنْ مَعْمَرِ بْنِ أَبِي مَعْمَرٍ، أَحَدِ بَنِي عَدِيٍّ بْنِ كَعْبٍ قَالَ قَالَ رَسُولُ اللَّهِ صَلَّى اللَّهُ عَلَيْهِ وَسَلَّمَ " لَا يَحْتَكِرُ إِلَّا خَاطِئٌ " . فَقُلْتُ لِسَعِيدٍ فَإِنَّكَ تَحْتَكِرُ قَالَ وَمَعْمَرٌ كَانَ يَحْتَكِرُ . قَالَ أَبُو دَاوُدَ وَسَأَلْتُ أَحْمَدَ مَا الْحِكْمَةُ قَالَ مَا فِيهِ عَيْشُ النَّاسِ . قَالَ أَبُو دَاوُدَ قَالَ الْأَوْزَاعِيُّ الْمُحْتَكِرُ مَنْ يَعْزِضُ الشُّوقَ

*Artinya: "Telah menceritakan kepada kami (Wahb bin Baqiyyah), telah mengabarkan kepada kami (Khalid) dari ('Amr bin Yahya), dari (Muhammad bin 'Amr bin 'Atha') dari (Sa'id bin Al Musayyab) dari (Ma'mar bin Abu Ma'mar) salah satu Bani Adi bin Ka'b, ia berkata; Rasulullah shallallahu 'alaihi wasallam bersabda: "Tidaklah seseorang menimbun barang, kecuali telah berbuat salah." Kemudian aku katakan kepada Sa'id; sesungguhnya engkau menimbun. Ia berkata; dan Ma'mar pernah menimbun. Abu Daud berkata; dan aku bertanya kepada Ahmad; apakah hukrah itu? Ia berkata; sesuatu yang padanya terdapat kehidupan manusia. Abu Daud berkata; Al Auza'i berkata; muhtakir adalah orang*

yang datang ke pasar untuk membeli apa yang dibutuhkan orang-orang dan menyimpannya” (Hadis Riwayat Abu Daud, 3447).

Dari riwayat lain dalam kitab *Bulugh al-Maram* karya Al-Hafizh Ibnu Hajar Al-Asqalani (2007) disampaikan,

وَعَنْ مَعْمَرِ بْنِ عَبْدِ اللَّهِ - رَضِيَ اللَّهُ عَنْهُ - عَنِ رَسُولِ اللَّهِ - صَلَّى اللَّهُ عَلَيْهِ وَسَلَّمَ - قَالَ: { لَا يَحْتَكِرُ إِلَّا خَاطِئٌ } { رَوَاهُ مُسْلِمٌ .

Artinya: “Dan diriwayatkan Ma'mar bin 'Abdullah (RA): Rasulullah SAW bersabda: “Tidak ada seorang pun yang menahan suatu barang sampai harganya naik, melainkan orang yang berdosa” (Hadis Riwayat Muslim, 784).

Dari kedua hadis di atas, dijelaskan bahwa Rasulullah SAW telah melarang tindakan seseorang yang menahan atau menimbun suatu barang sampai harganya naik. Apabila ia melakukan hal tersebut maka termasuk orang yang berdosa. *Hukrah* (penimbunan) yang dimaksud pada hadis tersebut yaitu orang yang menahan persediaan barang di pasar. Kesimpulan dari hadis tersebut yaitu akan menjadi orang yang berdosa apabila orang tersebut melakukan penimbunan barang yang mengakibatkan kenaikan harga barang tersebut atau bahkan hingga menimbulkan inflasi.

Pada kitab *Fathul Mu'in* karya seorang alim ulama Syaikh Zainudin al-Malibari, beliau mendefinisikan *ihtikar* sebagai berikut:

وَحَرْمٌ أَحْتِكَاؤُ قُوْتٍ، كَتَمَرٍ وَرَيْبٍ وَكُلِّ مُجْزِيٍّ فِي الْفِطْرَةِ، وَهُوَ إِمْسَاكُ مَا أَلْشْتَرَاهُ فِي وَقْتِ الْعَلَاءِ لَا الرَّحْصِ لِيَبْعَهُ بِأَكْثَرٍ عِنْدَ أَشْتِدَادِ حَاجَةِ أَهْلِ مَحَلِّهِ أَوْ غَيْرِهِمْ إِلَيْهِ، وَإِنْ لَمْ يَشْتَرِهِ بِقَصْدِ ذَلِكَ، لَا لِيُمْسِكَهُ لِنَفْسِهِ أَوْ عِيَالِهِ، أَوْ لِيَبْعَهُ بِتَمَنٍ مِثْلِهِ، وَلَا إِمْسَاكُ غَلَّةٍ أَرْضِيهِ.

Artinya: “Dan haram menimbun bahan makanan pokok, misalnya kurma dan anggur serta segala bahan makanan yang mencukupi dalam zakat fitrah. *Ikhtikar* (menimbun) adalah menahan bahan makanan dari pembelian di waktu harga mahal –bukan sewaktu murah-, untuk dijual kembali dengan harga di atasnya ketika penduduk setempat atau orang-orang lain sangat membutuhkannya,



*sekalipun di waktu membeli bukan bertujuan menjual dengan harga yang lebih tinggi. Tidak termasuk ihtikar, jika menahan bahan makanan pokok itu untuk keperluan diri sendiri atau keluarganya, atau untuk dijual dengan harga yang sepadan dengan harga pembelian. Tidak termasuk pula, jika yang ditahan adalah hasil panen sendiri.”*

Dari hadis-hadis di atas dapat disimpulkan bahwa hukum dari menimbun barang secara besar-besaran dan menjualnya kembali dengan harga yang mahal disaat terutama disaat masa-masa dibutuhkannya barang tersebut adalah haram. Selain itu, hal yang seperti ini akan menjadikan nilai inflasi di daerah tersebut bahkan di negara tersebut akan naik karena ulah nakal oknum yang menimbun barang.

## **BAB III**

### **METODE PENELITIAN**

#### **3.1 Jenis Penelitian**

Jenis penelitian yang dilakukan yaitu penelitian kuantitatif evaluasi. Pada penelitian ini, pengolahan data yang dilakukan yaitu menggunakan data dalam bentuk angka. Hasil penelitian ditulis dalam bentuk pernyataan yang didukung oleh angka dan diuji berdasarkan pola hubungan antar variabel yang akan dibuktikan. Peneliti melakukan penelitian dengan langkah-langkah yang sistematis. Data inflasi akan diproses menggunakan metode jaringan saraf tiruan *backpropagation*.

#### **3.2 Data dan Sumber Data**

Data yang akan digunakan dalam penelitian ini adalah nilai inflasi bulanan untuk masing-masing kelompok dari tahun 2005 hingga 2019. Data ini diperoleh melalui pendekatan sekunder dari *website* Badan Pusat Statistik ([www.bps.go.id](http://www.bps.go.id)), dan jenis data yang digunakan adalah data berkala (*time series*) dari tahun 2005 hingga 2019. Data tersebut terdiri dari delapan kelompok inflasi yaitu sebagai berikut:

1. Bahan makanan;
2. Makanan jadi, minuman, rokok, dan tembakau;
3. Perumahan, air, listrik, gas, dan bahan bakar;
4. Sandang;
5. Kesehatan;

6. Pendidikan, rekreasi dan olahraga;
7. Transport, komunikasi, dan jasa keuangan
8. Umum

Dari kedelapan kelompok tersebut, tujuh kelompok pertama akan menjadi data masukan dan kelompok kedelapan akan menjadi data target keluaran.

### 3.3 Teknik Analisis Data

Terdapat beberapa proses pada penelitian ini yaitu sebagai berikut:

1. Mengumpulkan Data

Pada proses ini, data inflasi bulanan dari *website* BPS Kota Malang akan dikumpulkan mulai dari data pada bulan Januari tahun 2005 sampai dengan data bulan Desember tahun 2019. Data tersebut berjumlah 180 data dengan tujuh kelompok.

2. Melakukan *Preprocessing* Data

Pada tahap *preprocessing*, data yang telah dikumpulkan akan dinormalisasi dari bentuk awal menjadi data dengan skala 0,1 sampai 1 untuk meningkatkan kinerja model pembelajaran mesin dengan memastikan bahwa fitur-fitur memiliki skala yang sama serta untuk menghindari adanya nilai nol. Proses normalisasi dilakukan dengan menggunakan rumus pada Persamaan (2.14).

3. Menentukan Model Arsitektur Jaringan

Pada proses ini, akan ditentukan jumlah neuron pada *input layer* yaitu sebanyak tujuh neuron, jumlah lapisan dan jumlah neuron pada *hidden layer* yaitu sebanyak satu lapis dengan empat sampai 14 neuron, mengikuti

metode penentuan jumlah neuron di *hidden layer*, serta jumlah neuron pada *output layer* yaitu satu neuron. Sehingga pada penelitian ini, terdapat 11 model arsitektur yang akan dicoba yaitu arsitektur 7-4-1, 7-5-1, 7-6-1, 7-7-1, 7-8-1, 7-9-1, 7-10-1, 7-11-1, 7-12-1, 7-13-1 dan 7-14-1.

Pada proses ini akan ditentukan juga beberapa parameter yang akan dicoba pada penelitian yaitu fungsi aktivasi yaitu ReLu dan *tanh*, karena kedua fungsi ini mampu memecahkan masalah non-linier, serta memerlukan sumber daya komputasi yang jauh lebih sedikit. Kemudian ukuran *batch* yang dicoba yaitu 8 dan 16, karena dengan menggunakan ukuran *batch* kecil dapat menghasilkan konvergensi yang lebih stabil dan mungkin membantu dalam mendapatkan model yang lebih baik. Sedangkan apabila menggunakan ukuran *batch* besar akan menyebabkan error yang lebih besar dan kurang stabil. Selain itu, parameter jumlah *epoch* yang digunakan yaitu 1000 dan 1500, Karena pelatihan model dengan jumlah *epoch* yang banyak dapat membantu memastikan bahwa model telah mencapai konvergensi atau mendekati nilai loss minimum. Sehingga model tidak hanya berlatih dengan cepat, namun juga mendapatkan hasil yang optimal.

#### 4. Melakukan *Training Data*

Pada proses ini, data yang telah dikumpulkan akan dibagi menjadi dua, yaitu 93% data sebagai data pelatihan yang terdiri dari 168 data pada tahun 2005-2018 dan 7% data sebagai data pengujian yang terdiri dari 12 data pada tahun 2019. Pemilihan persentase pembagian data tersebut dikarenakan pada penelitian ini yang akan diprediksi yaitu tingkat inflasi

di tahun 2019, sehingga data target untuk proses testing yang harus digunakan yaitu data dari bulan Januari 2019 sampai dengan bulan Desember 2019.

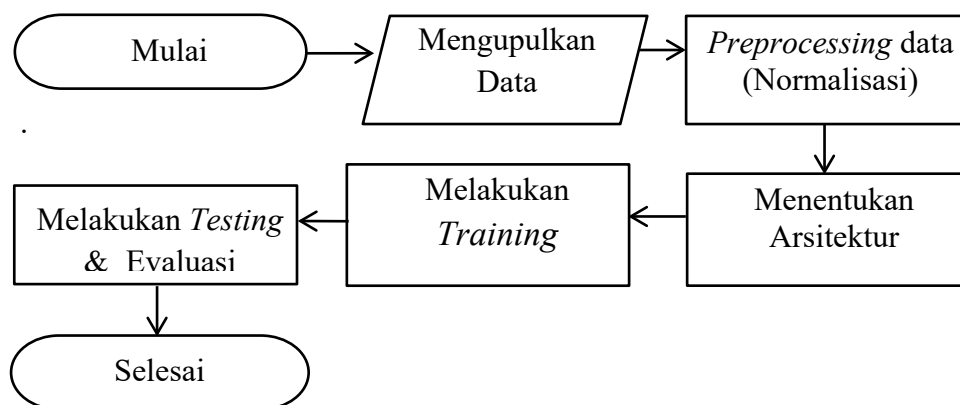
Kemudian dari data pelatihan akan diolah dengan nilai bobot yang telah ditentukan dan diaktivasi sehingga mendapatkan suatu hasil *error*, dari *error* tersebut apabila nilai *error* belum mencapai target *error* atau nilai hasil belum mendekati target *output* maka bobot akan diperbarui dan dilakukan aktivasi ulang sampai hasil *error* mencapai target atau sampai jumlah perulangan mencapai target *epoch*. Proses tersebut akan dilakukan untuk semua model arsitektur.

#### 5. Melakukan *Testing* dan Evaluasi

Pada proses *testing*, data pengujian akan diolah seperti pada proses *training* dengan menggunakan bobot dan bias yang dihasilkan dan disimpan dari hasil proses *training*, namun proses ini hanya dilakukan sekali saja tanpa perulangan dan pembaruan bobot, sehingga dihasilkan suatu hasil *output*. Kemudian hasil *output* tersebut akan dibandingkan dengan nilai target inflasi dalam bentuk normalisasi dari sehingga didapatkan nilai akurasi dari model testing dari tiap parameter. Kemudian dari akurasi tersebut, akan diambil model dengan akurasi tertinggi dari tiap parameter yang dicoba, selanjutnya hasil output dari model tersebut akan dikembalikan menjadi data dengan rentang seperti data asli dengan dinormalisasi menggunakan Persamaan (2.15). Setelah itu, hasil dinormalisasi dari tiap model terbaik tiap parameter tersebut akan dievaluasi, yaitu dengan membandingkan data hasil dinormalisasi dengan

data aktual inflasi di Kota Malang pada tahun 2019 sehingga didapat nilai MAPE dan akurasi dari hasil denormalisasi. Perhitungan MAPE dan akurasi dilakukan dengan menggunakan Persamaan (2.18) dan Persamaan (2.19). Dari nilai akurasi hasil denormalisasi tersebut, akan dicari nilai akurasi yang tertinggi, sehingga model dengan akurasi tertinggi tersebut akan menjadi model arsitektur akhir untuk melakukan prediksi tingkat inflasi di Kota Malang.

Berikut merupakan diagram alir teknik analisis data pada penelitian ini.



Gambar 3.1 Diagram Alir Teknik Analisis Data

## BAB IV

### HASIL DAN PEMBAHASAN

#### 4.1 Mengumpulkan Data

Data penelitian didapatkan sebanyak 180 data yang diambil dari data inflasi bulanan Kota Malang mulai bulan Januari tahun 2005 sampai dengan bulan Desember tahun 2019. Data yang diambil secara langsung dari *website* BPS Kota Malang tersebut memiliki delapan kolom dengan tujuh kolom inflasi bulanan dari setiap kelompok dan 1 kolom inflasi bulanan secara keseluruhan.

**Tabel 4.1** Data Inflasi Kota Malang Tahun 2005-2019 Perbulan dan Perkelompok

<b>Kelompok Inflasi</b>	<b>Januari 2005</b>	<b>Februari 2005</b>	<b>...</b>	<b>November 2019</b>	<b>Desember 2019</b>
$x_1$	1,86	-3,39	...	0,19	1,01
$x_2$	0,35	1,49	...	0,08	1,03
$x_3$	1,71	0,18	...	0,19	0,34
$x_4$	-0,28	-0,26	...	-0,22	-0,36
$x_5$	0,92	-0,04	...	0,09	0,23
$x_6$	0	0	...	0,16	0,05
$x_7$	0,15	0	...	-0,43	0,26
$y$	0,95	-0,43	...	0,01	0,5

Keterangan:

$x_1$  : Bahan makanan

$x_2$  : Makanan Jadi, Minuman, Rokok, dan Tembakau

$x_3$  : Perumahan, Air, Listrik, Gas, dan Bahan Bakar

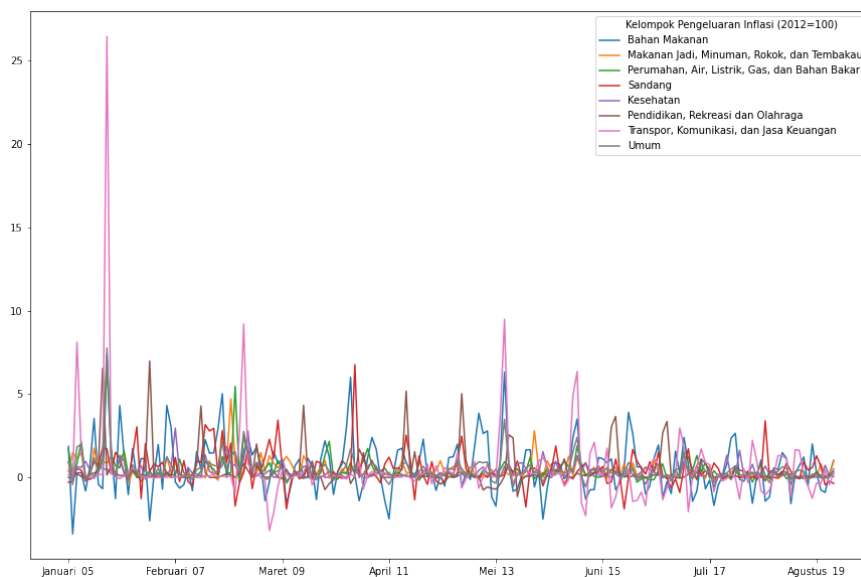
$x_4$  : Sandang

$x_5$  : Kesehatan

$x_6$  : Pendidikan, Rekreasi dan Olahraga

$x_7$  : Transpor, Komunikasi, dan Jasa Keuangan

$y$  : Umum



**Gambar 4.1** Grafik Data Inflasi Kota Malang Tahun 2005-2019 Perbulan dan Perkelompok

Dari Tabel 4.1 di atas, data dari kelompok  $x_1$  sampai dengan  $x_7$  merupakan variabel independen pada penelitian ini. Kemudian, data pada baris  $y$  merupakan variabel dependen. Data inflasi tersebut yang lebih lengkap terdapat pada Lampiran 1. Berikut merupakan deskriptif statistik sederhana dari data di atas.

**Tabel 4.2** Statistik Deskriptif dari Data Inflasi Setiap Kelompok

<b>Kelompok</b>	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$x_7$	$y$
<b>Ukuran</b>	180	180	180	180	180	180	180	180
<b>Rata-rata</b>	0,52	0,16	0,18	0,27	0,07	0,30	0,37	0,18
<b>Standar Deviasi</b>	1,39	0,47	0,68	0,81	0,30	1,03	2,36	0,67



<b>Kelompok</b> <b>Ukuran</b>	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$x_7$	$y$
<b>Minimum</b>	-3,39	-0,31	-0,33	-1,89	-0,32	-1,15	-3,18	-0,57
<b>Maksimum</b>	7,38	4,72	6,6	6,77	2,96	6,99	26,47	7,77
<b>Jangkauan</b>	10,77	5,03	6,93	8,66	3,28	8,14	29,65	8,34

Dari tabel di atas ditunjukkan bahwa jumlah keseluruhan data yaitu 180 data, kemudian terdapat data dengan *range* yang cukup besar dan yang tidak sebanding dengan *range* dari kelompok lain yaitu pada kelompok  $x_7$  atau kelompok Transpor, Komunikasi, dan Jasa Keuangan. Oleh karena itu, data yang terdapat penyebarannya yang tidak merata tersebut maka akan dilakukan *pre-processing* data dengan melakukan normalisasi data.

#### 4.2 Melakukan *Preprocessing* Data

Setelah data terkumpul, maka dilakukan preprocessing dengan menormalisasi data menggunakan rumus pada Persamaan (2.14) sehingga didapat data dengan rentang 0,1 sampai 1. Adapun rumus perhitungan normalisasi seperti berikut.

$$v'_n = 0,1 + \frac{(v_n - v_{min}) \times (1 - 0,1)}{v_{max} - v_{min}}$$

Di mana  $v_n$  adalah operasi pada salah satu data inflasi ke- $n$  dan menentukan nilai terkecil dan terbesar dari  $v$ . Berikut contoh untuk mencari hasil normalisasi ke-1, yaitu data inflasi bulan Januari 2005 untuk kelompok bahan makanan.

$$v'_n = 0,1 + \frac{(v_n - v_{min}) \times (1 - 0,1)}{v_{max} - v_{min}}$$

$$v'_1 = 0,1 + \frac{(1,86 - (-3,39)) \times (1 - 0,1)}{7,38 - (-3,39)}$$

$$v'_1 = 0,1 + \frac{(1,86 + 3,39) \times 0,9}{(7,38 + 3,39)}$$

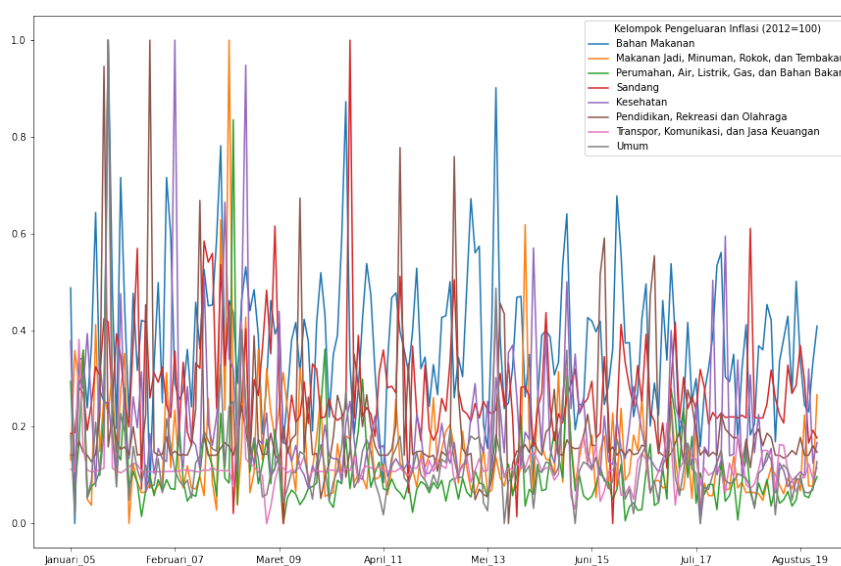
$$v'_1 = 0,1 + \frac{4,725}{10,77}$$

$$v'_1 = 0,538$$

Perhitungan tersebut dilakukan terus menerus hingga ke seluruh data sehingga didapatkan hasil normalisasi pada Tabel 4.3 (hasil lebih lengkap dapat dilihat pada lampiran 2).

**Tabel 4.3** Contoh 5 Data Inflasi Hasil Normalisasi

Kelompok Bulan	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$x_7$	$y$
<b>Januari 2005</b>	0,54	0,22	0,36	0,27	0,44	0,23	0,20	0,26
<b>Februari 2005</b>	0,10	0,42	0,17	0,27	0,18	0,23	0,20	0,12
<b>Maret 2005</b>	0,41	0,36	0,25	0,35	0,37	0,25	0,44	0,36
<b>April 2005</b>	0,41	0,42	0,42	0,34	0,35	0,23	0,24	0,38
<b>Mei 2005</b>	0,32	0,14	0,15	0,27	0,45	0,23	0,20	0,15



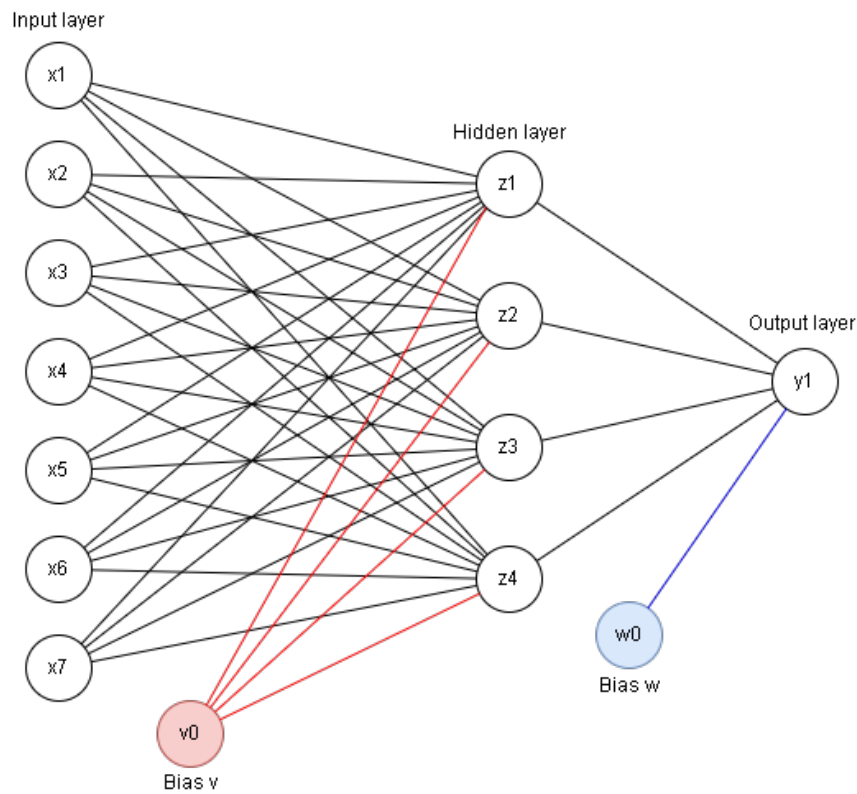
**Gambar 4.2** Grafik Data Inflasi Hasil Normalisasi

Dari Gambar 4.2 di atas, dapat disimpulkan bahwa proses normalisasi data bertujuan untuk meratakan penyebaran data. Data awal seperti pada Gambar 4.1 yang awalnya memiliki nilai dengan selisih yang jauh berbeda dengan data lainnya, setelah dinormalisasi maka selisih antar data menjadi lebih dekat dan merata.

### 4.3 Menentukan Model Arsitektur Jaringan

Proses menentukan model arsitektur jaringan pada penelitian ini yaitu dengan memilih jumlah *input layer*, *hidden layer*, dan *output layer* beserta jumlah neuron pada masing-masing *layer*. Dipilih satu *input layer* dengan tujuh neuron karena terdapat tujuh fitur  $x$  pada data *training* seperti pada Tabel 4.1, kemudian *hidden layer* sebanyak satu lapisan yang akan dicoba dengan jumlah neuron berbeda yaitu 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, dan 14 neuron, serta satu *output layer* dengan satu neuron karena *output* dari proses ini yaitu satu nilai inflasi bulanan. Jumlah neuron-neuron tersebut disesuaikan dengan metode penentuan neuron oleh Heaton (2008). Dengan demikian, uji coba yang dilakukan sebanyak sepuluh kali yaitu dengan model arsitektur 7-4-1, 7-5-1, 7-6-1, 7-7-1, 7-8-1, 7-9-1, 7-10-1, 7-11-1, 7-12-1, 7-13-1 dan 7-14-1. Dari 11 model arsitektur tersebut, model arsitektur yang menghasilkan nilai akurasi tertinggi yang akan dipilih sebagai model akhir untuk prediksi nilai inflasi di Kota Malang.

Berikut merupakan gambar contoh arsitektur jaringan dengan satu *input layer* dengan tujuh neuron, satu *hidden layer* dengan empat neuron, dan satu *output layer* dengan satu neuron.



**Gambar 4.3** Model Arsitektur Jaringan *Backpropagation* 7-4-1 yaitu 7 Neuron di *Input Layer* 4 Neuron di *Hidden Layer* dan 1 Neuron di *Output Layer*.

Selain itu, parameter besar nilai bobot dan bias untuk proses *training* dipilih secara random dengan interval antara -1 sampai dengan 1. Nilai tersebut dipilih dan diperbarui secara otomatis dari *software tensorflow*. Kemudian, bobot dan bias pada proses *testing* yaitu bobot dan bias yang digunakan saat proses *training* dimana dengan bobot dan bias tersebut, model arsitektur yang dicoba tersebut dapat menghasilkan nilai error terkecil.

Selanjutnya, untuk parameter lain yang dicoba pada penelitian ini yaitu fungsi aktivasi yaitu ReLu dan *Tanh*, jumlah *epoch* sebanyak 1000 dan 1500, serta jumlah *batch* sebanyak 8 dan 16.

#### 4.4 Melakukan *Training Data*

Data yang telah dinormalisasi kemudian dibagi menjadi dua yaitu data *training* dan data *testing*. Data *training* pada penelitian ini yaitu data inflasi dari bulan Januari tahun 2005 hingga bulan Desember 2018. Data ini yang kemudian dilatih dalam proses *training*. Kemudian data *testing* pada penelitian ini yaitu data inflasi bulan Januari tahun 2019 hingga bulan Desember 2019. Persentase dari masing-masing data *training* dan data *testing* tersebut yaitu 93% data *training* dan 7% data *testing*. pembagian Pemilihan persentase pembagian data tersebut dikarenakan pada penelitian ini yang akan diprediksi yaitu tingkat inflasi di tahun 2019, sehingga data target untuk proses testing yang harus digunakan yaitu data dari bulan Januari 2019 sampai dengan bulan Desember 2019.

**Tabel 4.4** Data *Training*

<b>Neuron</b> <b>Bulan</b>	<b>X<sub>1</sub></b>	<b>X<sub>2</sub></b>	<b>X<sub>3</sub></b>	<b>X<sub>4</sub></b>	<b>X<sub>5</sub></b>	<b>X<sub>6</sub></b>	<b>X<sub>7</sub></b>	<b>Y</b>
<b>Januari 2005</b>	0,54	0,22	0,36	0,27	0,44	0,23	0,20	0,26
<b>Februari 2005</b>	0,10	0,42	0,17	0,27	0,18	0,23	0,20	0,12
...	...	...	...	...	...	...	...	...
<b>November 2018</b>	0,42	0,14	0,15	0,30	0,20	0,24	0,24	0,20
<b>Desember 2018</b>	0,51	0,18	0,18	0,32	0,22	0,27	0,24	0,23

**Tabel 4.5** Data *Testing*

<b>Neuron</b> <b>Bulan</b>	<b>X<sub>1</sub></b>	<b>X<sub>2</sub></b>	<b>X<sub>3</sub></b>	<b>X<sub>4</sub></b>	<b>X<sub>5</sub></b>	<b>X<sub>6</sub></b>	<b>X<sub>7</sub></b>	<b>Y</b>
<b>Januari 2019</b>	0,48	0,20	0,13	0,39	0,24	0,26	0,23	0,22
<b>Februari 2019</b>	0,25	0,19	0,17	0,33	0,16	0,23	0,16	0,12
<b>Maret 2019</b>	0,4	0,17	0,14	0,3	0,21	0,23	0,25	0,20
<b>April 2019</b>	0,44	0,16	0,14	0,29	0,20	0,22	0,25	0,21
<b>Mei 2019</b>	0,49	0,18	0,16	0,40	0,17	0,23	0,20	0,20
<b>Juni 2019</b>	0,34	0,16	0,13	0,34	0,18	0,24	0,18	0,14
<b>Juli 2019</b>	0,55	0,19	0,14	0,36	0,19	0,23	0,16	0,18
<b>Agustus 2019</b>	0,41	0,16	0,19	0,43	0,20	0,26	0,19	0,18
<b>September 2019</b>	0,32	0,30	0,15	0,37	0,19	0,23	0,18	0,16

Neuron Bulan	$X_1$	$X_2$	$X_3$	$X_4$	$X_5$	$X_6$	$X_7$	$Y$
Oktober 2019	0,31	0,17	0,15	0,25	0,39	0,23	0,21	0,16
November 2019	0,40	0,17	0,17	0,27	0,21	0,24	0,18	0,16
Desember 2019	0,47	0,34	0,19	0,26	0,25	0,23	0,20	0,22

Setelah itu, dari setiap data *training* dan *testing*, akan dipisahkan antara fitur masukan dan target keluaran. Pada data *training* dan *testing* di atas, fitur masukan untuk proses *training* dan *testing* yaitu seluruh data pada kolom  $X_i$  dengan  $i = 1, 2, \dots, 7$ . Kemudian, data target keluaran untuk proses *training* dan *testing* yaitu data pada kolom  $y$ .

Dari data *training* di atas, maka langkah selanjutnya yaitu melakukan proses *training* sesuai dengan langkah-langkah *backpropagation*. Berikut merupakan contoh perhitungan manual dari proses *training*.

**Langkah 1:** Inisialisasi nilai awal. Bobot dan bias awal masing-masing neuron diinisialisasi secara *random*, tetapi pada contoh ini diinisialisasi dengan angka 1 dan 0.

$$\begin{aligned}
 X_1 &= 0,54 & X_2 &= 0,22 & X_3 &= 0,36 & X_4 &= 0,27 & X_5 &= 0,44 \\
 X_6 &= 0,32 & X_7 &= 0,20 & V_{11} &= 1 & V_{21} &= 1 & V_{31} &= 1 \\
 V_{41} &= 1 & V_{51} &= 1 & V_{61} &= 1 & V_{71} &= 1 & W_{11} &= 1 \\
 \alpha &= 0,01 & Y = t_k &= 0,26 & V_{01} &= 0
 \end{aligned}$$

**Langkah 2:** Inisialisasi kondisi berhenti. Iterasi akan berhenti ketika hasil *error* sudah mencapai batas *error* atau ketika jumlah iterasi sudah mencapai batas maksimal.

Pada contoh perhitungan ini, dimisalkan  $error = 0,001$

**Langkah 3:** Melakukan langkah 4-10 pada setiap data *training*.

**Langkah 4:** Sinyal *input*  $X_i$  diterima oleh setiap neuron *input*  $X_i$  dengan  $i = 1, 2, 3, \dots, 7$  dan didistribusikan ke neuron *hidden layer*.

**Langkah 5:** Menentukan nilai neuron pada *hidden layer*  $Z_{input_j}$  dengan  $j = 1, 2, \dots, m$  dimana  $m$  adalah jumlah neuron di *hidden layer*.

Nilai masukan pada neuron *hidden layer* dari *input layer* adalah sebagai berikut

$$Z_{input_j} = V_{0_j} + \sum_{i=1}^n X_i V_{ij}$$

$$Z_{input_1} = V_{0_1} + \sum_{i=1}^7 X_i V_{i1}$$

$$\begin{aligned} Z_{input_1} &= 0 + (0,54 \times 1) + (0,22 \times 1) + (0,36 \times 1) + (0,27 \times 1) + (0,44 \times 1) \\ &\quad + (0,23 \times 1) + (0,20 \times 1) \end{aligned}$$

$$Z_{input_1} = 2,52$$

Sehingga didapat nilai masukan untuk neuron pertama di *hidden layer* yaitu 1,73.

Kemudian dihitung nilai neuron pada *hidden layer* dengan fungsi aktivasi. Pada contoh perhitungan ini, fungsi aktivasi yang digunakan yaitu fungsi aktivasi ReLu

Jadi, nilai  $Z_j = 1, 2, \dots, m$  dimana  $m$  adalah jumlah neuron di *hidden layer* sebagai berikut.

$$Z_j = f(Z_{input_j})$$

$$Z_1 = \max(0; Z_{input_1})$$

$$Z_1 = \max(0; 2,52)$$

$$Z_1 = 2,52$$

Sehingga didapatkan nilai akhir pada neuron pertama dari *hidden layer* yaitu 2,52.

**Langkah 6:** Menghitung hasil keluaran pada *output layer*.

Dihitung nilai masukan pada neuron *output layer*, yaitu  $Y_{input_{jk}}$  dengan  $j = 1, 2, \dots, m$  dimana  $m$  adalah jumlah neuron di *hidden layer* dan  $k = 1, 2, \dots, p$  dimana  $p$  adalah jumlah neuron di *output layer*, adalah sebagai berikut.

$$Y_{input_{jk}} = Z_j W_{jk}$$

$$Y_{input_{11}} = Z_1 W_{11}$$

$$Y_{input_{11}} = 2,52 \times 1$$

$$Y_{input_{11}} = 2,52$$

Didapat hasil nilai masukan pada neuron *output* pertama dari neuron *input layer* pertama yaitu sebesar 2,52.

Kemudian dihitung nilai keluaran dengan menggunakan fungsi aktivasi, pada contoh perhitungan ini digunakan fungsi aktivasi ReLU.

$$Y_{output} = f(Y_{input_{jk}})$$

$$Y_{output} = \max(0; Y_{input_{11}})$$

$$Y_{output} = \max(0; 2,52)$$

$$Y_{output} = 2,52$$

Sehingga didapat hasil akhir untuk nilai pada neuron keluaran yaitu 2,52.

**Langkah 7 :** Menghitung faktor koreksi dari neuron *output*.

Pada tahap ini dilakukan secara mundur, yaitu dari *output layer*, menuju *hidden layer*. Nilai faktor koreksi dari neuron *output layer* yaitu  $\delta_k$  dengan  $k = 1, 2, \dots, p$



dimana  $p$  adalah jumlah neuron di *output layer*, yang akan digunakan untuk memperbarui bobot, yaitu sebagai berikut.

$$\delta_k = (t_k - Y_{output_k})Y_{output_k}(1 - Y_{output_k})$$

$$\delta_1 = (t_1 - Y_{output_1})Y_{output_1}(1 - Y_{output_1})$$

$$\delta_1 = (0,26 - 2,52)(2,52)(1 - 2,52)$$

$$\delta_1 = 8,656$$

Sehingga, dapat dihitung nilai koreksi bobot yang menghubungkan antara neuron *output layer* dan neuron *hidden layer* yaitu  $\Delta W_{jk}$ , dengan  $j = 1, 2, \dots, m$  dimana  $m$  adalah jumlah neuron di *hidden layer* dan  $k = 1, 2, \dots, p$  dimana  $p$  adalah jumlah neuron di *output layer*, yaitu sebagai berikut.

$$\Delta W_{jk} = \alpha \delta_k Z_j$$

$$\Delta w_{11} = \alpha \delta_1 z_1$$

$$\Delta w_{11} = 0,01 \times (8,656) \times 2,52$$

$$\Delta w_{11} = 0,218$$

Sehingga didapat nilai koreksi bobot yang menghubungkan antara neuron pertama di *output layer* dan neuron pertama di *hidden layer* yaitu sebesar 0,218.

**Langkah 8** : Menghitung faktor koreksi dari *hidden layer*.

Dihitung nilai sinyal faktor koreksi dari neuron *hidden layer* yaitu  $\delta_{in_j}$ , dengan  $j = 1, 2, \dots, m$  dimana  $m$  adalah jumlah neuron di *hidden layer*, yang akan digunakan untuk menghitung faktor koreksi, yaitu sebagai berikut.

$$\delta_{input_j} = \sum_{k=1}^m \delta_k W_{jk}$$

$$\delta_{input_1} = \sum_{k=1}^m \delta_k W_{1k}$$

$$\delta_{input_1} = 8,656 \times 1$$

$$\delta_{input_1} = 8,656$$

Dari hasil sinyal faktor koreksi tersebut, maka dapat dihitung nilai faktor koreksi dari neuron *hidden layer* yaitu

$$\delta_j = \delta_{input_j} Z_j (1 - Z_j)$$

$$\delta_1 = \delta_{input_1} Z_1 (1 - Z_1)$$

$$\delta_1 = (8,656)(2,52)(1 - 2,52)$$

$$\delta_1 = -33,15$$

Kemudian dapat dihitung pula nilai koreksi bobot yang menghubungkan antara neuron *hidden layer* dengan neuron *input* yaitu sebagai berikut.

$$\Delta V_{ji} = \alpha \delta_j X_i$$

$$\Delta V_{11} = \alpha \delta_1 X_1$$

$$\Delta V_{11} = 0,01 \times (-33,15) \times 0,54$$

$$\Delta V_{11} = -0,179$$

Sehingga nilai koreksi bobot yang menghubungkan antara neuron pertama di *hidden layer* dengan neuron pertama di *input layer* yaitu sebesar -0,179.

**Langkah 9:** Menghitung *update* bobot.

Setelah didapat nilai koreksi bobot dari *output layer* ke *hidden layer* dan *hidden layer* ke *input layer*, maka bobot baru dari keduanya bisa diperbarui.

Untuk bobot baru antara *hidden layer* dengan *output layer* atau bobot  $W_{jk}$  dengan  $j = 1, 2, \dots, m$  dimana  $m$  adalah jumlah neuron di *hidden layer* dan  $k = 1, 2, \dots, p$  dimana  $p$  adalah jumlah neuron di *output layer*, adalah sebagai berikut.

$$W_{jk(\text{baru})} = W_{(lama)} + \Delta W_{jk}$$

$$W_{11(\text{baru})} = W_{11(\text{lama})} + \Delta W_{11}$$

$$W_{11(\text{baru})} = 1 + 0,218$$

$$W_{11(\text{baru})} = 1,0218$$

Sedangkan bobot baru antara *input layer* dengan *hidden layer* yaitu  $V_{ij}$  dengan  $i = 1, 2, \dots, n$  dimana  $n$  adalah jumlah neuron di *input layer* dan  $j = 1, 2, \dots, m$  dimana  $m$  adalah jumlah neuron di *hidden layer*, adalah sebagai berikut.

$$V_{ij(\text{baru})} = V_{ij(\text{lama})} + \Delta V_{ij}$$

$$V_{11(\text{baru})} = V_{11(\text{lama})} + \Delta V_{11}$$

$$V_{11(\text{baru})} = 1 + (-0,179)$$

$$V_{11(\text{baru})} = 0,821$$

Bobot baru tersebut digunakan untuk menghitung kembali *data training* hingga didapat nilai keluaran  $Y_{out}$  yang mendekati target  $Y$  atau nilai faktor koreksi yang mendekati batas nilai *error*. Kemudian seluruh *data training* yang ada, dihitung dengan cara yang sama dengan cara seperti di atas dari langkah 1 hingga langkah 9 sehingga didapat nilai *error* dan akurasi yang tertinggi.

Selain itu, pada penelitian ini, proses *training* juga dilakukan dengan menggunakan *software tensorflow* pada *jupyter notebook* untuk menghemat waktu dan juga untuk memberikan hasil yang lebih akurat karena menggunakan bobot dengan nilai *random*. Hasil dari proses *training* tersebut adalah model untuk melakukan prediksi yaitu sebagai berikut.

**Tabel 4.6** Rangkuman Hasil Proses *Training* Model Arsitektur 7-14-1

Lapisan (tipe)	Bentuk Keluaran	Parameter
Dense_8 (Dense)	(Tidak ada, 14)	112
Dense_9 (Dense)	(Tidak ada, 1)	15
Total parameter: 127		
Parameter yang bisa dilatih: 127		
Parameter yang tidak bisa dilatih: 0		

Dari Tabel 4.6 di atas, dapat dilihat bahwa model untuk melakukan prediksi yaitu model dengan satu *input layer* dengan tujuh neuron, satu *hidden layer* dengan 14 neuron, dan satu *output layer* dengan satu neuron. Sehingga, pada model tersebut terdapat 127 bobot yang terbagi menjadi 98 bobot V dan 14 bobot bias antara *input layer* dengan *hidden layer*, dan 14 bobot W dan satu bobot bias antara *hidden layer* dengan *output layer*. Rincian bobot pada model hasil proses *training* yaitu terdapat pada tabel-tabel berikut.

**Tabel 4.7** Hasil Bobot dan Bias V antara *Input Layer* dengan *Hidden Layer* dari Proses *Training*

Neuron Bobot	X1	X2	X3	X4	X5	X6	X7	V0 (bias)
V1	-0,381	-0,321	-0,285	-0,125	0,324	-0,216	0,247	0,030
V2	-0,400	-0,464	0,306	0,373	0,151	0,333	0,480	-0,021
V3	0,305	-0,310	-0,149	-0,387	0,270	-0,145	0,291	0,024
V4	-0,006	-0,255	0,329	-0,491	0,209	0,412	-0,490	0,022
V5	0,138	-0,076	0,016	0,195	-0,152	0,427	0,248	-0,052
V6	0,097	0,378	-0,327	-0,525	0,410	0,104	-0,005	-0,031
V7	-0,312	0,160	-0,460	0,519	0,331	-0,204	-0,016	0,045
V8	-0,069	-0,186	-0,602	0,249	-0,080	0,025	0,050	-0,051
V9	0,149	0,207	0,203	-0,455	-0,414	0,002	0,412	-0,063
V10	0,172	0,503	-0,233	0,322	0,134	-0,294	-0,136	0,015
V11	-0,231	-0,110	-0,213	0,128	-0,488	-0,048	0,219	0,049
V12	-0,309	-0,188	-0,650	-0,319	0,585	0,443	0,138	-0,053
V13	0,160	0,310	0,106	0,278	0,387	-0,285	0,369	-0,071
V14	0,426	0,027	0,071	-0,491	-0,322	-0,203	0,021	-0,042

**Tabel 4.8** Hasil Bobot W dan Bias W antara *Hidden Layer* dengan *Output Layer* dari Proses *Training*

Neuron Bobot	$Y_j$
<b>W0 (bias)</b>	-0,027
<b>W1</b>	-0,324
<b>W2</b>	0,212
<b>W3</b>	-0,019
<b>W4</b>	-0,628
<b>W5</b>	0,201
<b>W6</b>	-0,007
<b>W7</b>	-0,241
<b>W8</b>	0,086
<b>W9</b>	0,235
<b>W10</b>	-0,443
<b>W11</b>	-0,514
<b>W12</b>	0,223
<b>W13</b>	0,570
<b>W14</b>	0,143

Bobot-bobot dari model tersebut akan digunakan untuk perhitungan pada proses *testing*.

#### 4.5 Melakukan *Testing* Data dan Evaluasi

Setelah mendapatkan bobot dan model dari proses *training*, langkah selanjutnya yaitu melakukan *testing* data dilakukan pada data *testing* untuk mendapatkan hasil prediksi dari data tes. Langkah-langkah yang dilakukan sama seperti proses *training* namun hanya sampai *fase feed-forward*. Karena bobot yang digunakan pada proses *testing* merupakan bobot dari *training* yang menghasilkan *error* terkecil, sehingga hasilnya akan menjadi hasil prediksi.

Proses *testing* dilakukan ke semua model dengan mencoba menggunakan parameter berbeda kemudian dari hasil *output* atau hasil prediksi dari percobaan

tersebut, maka dihitung akurasi dari tiap model dengan tiap parameter sehingga didapat hasil seperti berikut.

1. Parameter fungsi aktivasi ReLu dan ukuran *batch* 8

**Tabel 4.9** Hasil Akurasi Model Arsitektur dengan Parameter Fungsi Aktivasi Relu dan Ukuran *Batch* 8

<b>Jumlah Epoch</b>	<b>Jumlah Neuron</b>	<b>Akurasi (%)</b>
1000	4	97,97
	5	98,28
	6	98,54
	7	97,81
	8	98,11
	9	98,80
	10	98,42
	11	97,70
	12	98,37
	13	98,47
	14	98,34
1500	4	98,00
	5	98,08
	6	98,02
	7	98,39
	8	98,33
	9	98,14
	10	98,37
	11	98,45
	12	97,32
	13	98,37
	14	98,12

Dari tabel di atas, didapatkan model terbaik untuk parameter fungsi aktivasi ReLu, ukuran *batch* 8 dan *epoch* 1000 yaitu model dengan 9 neuron pada *hidden layer* dengan hasil akurasi sebesar 98,80%. Sedangkan model terbaik untuk parameter fungsi aktivasi ReLu, ukuran *batch* 8 dan *epoch* 1500 yaitu model dengan 11 neuron pada *hidden layer* dengan hasil akurasi sebesar 98,45%.

2. Parameter fungsi aktivasi ReLu dan ukuran *batch* 16**Tabel 4.10** Hasil Akurasi Model Arsitektur dengan Parameter Fungsi Aktivasi Relu dan Ukuran *Batch* 16

<b>Jumlah Epoch</b>	<b>Jumlah Neuron</b>	<b>Akurasi (%)</b>
1000	4	98,48
	5	98,50
	6	98,51
	7	98,45
	8	97,66
	9	98,58
	10	96,98
	11	97,26
	12	97,17
	13	97,71
	14	98,09
1500	4	97,67
	5	97,79
	6	98,56
	7	98,30
	8	96,62
	9	98,09
	10	98,28
	11	98,50
	12	97,88
	13	97,58
	14	96,61

Dari tabel di atas, didapatkan model terbaik untuk parameter fungsi aktivasi ReLu, ukuran *batch* 16 dan *epoch* 1000 yaitu model dengan 9 neuron pada *hidden layer* dengan hasil akurasi sebesar 98,58%. Sedangkan model terbaik untuk parameter fungsi aktivasi ReLu, ukuran *batch* 16 dan *epoch* 1500 yaitu model dengan 6 neuron pada *hidden layer* dengan hasil akurasi sebesar 98,56%.

3. Parameter fungsi aktivasi *tanh* dan ukuran *batch* 8**Tabel 4. 11** Hasil Akurasi Model Arsitektur dengan Parameter Fungsi Aktivasi *Tanh* dan Ukuran *Batch* 8

Jumlah Epoch	Jumlah Neuron	Akurasi (%)
1000	4	97,8
	5	98,00
	6	96,21
	7	97,52
	8	97,12
	9	97,34
	10	97,83
	11	96,08
	12	95,29
	13	95,90
	14	98,25
1500	4	97,98
	5	98,11
	6	97,36
	7	95,16
	8	98,01
	9	97,69
	10	98,15
	11	96,76
	12	96,65
	13	95,94
	14	97,20

Dari tabel di atas, didapatkan model terbaik untuk parameter fungsi aktivasi *tanh*, ukuran *batch* 8 dan *epoch* 1000 yaitu model dengan 14 neuron pada *hidden layer* dengan hasil akurasi sebesar 98,25%. Sedangkan model terbaik untuk parameter fungsi aktivasi *tanh*, ukuran *batch* 8 dan *epoch* 1500 yaitu model dengan 10 neuron pada *hidden layer* dengan hasil akurasi sebesar 98,15%.



4. Parameter fungsi aktivasi *tanh* ukuran *batch* 16**Tabel 4.12** Hasil Akurasi Model Arsitektur dengan Parameter Fungsi Aktivasi *Tanh* dan Ukuran *Batch* 16

Jumlah Epoch	Jumlah Neuron	Akurasi (%)
1000	4	97,76
	5	97,48
	6	98,13
	7	97,66
	8	96,92
	9	97,91
	10	97,24
	11	98,20
	12	97,02
	13	98,00
	14	98,26
1500	4	98,01
	5	97,69
	6	98,14
	7	97,60
	8	97,91
	9	97,61
	10	98,06
	11	98,05
	12	97,64
	13	98,06
	14	96,94

Dari tabel di atas, didapatkan model terbaik untuk parameter fungsi aktivasi *tanh*, ukuran *batch* 16 dan *epoch* 1000 yaitu model dengan 14 neuron pada *hidden layer* dengan hasil akurasi sebesar 98,26%. Sedangkan model terbaik untuk parameter fungsi aktivasi *tanh*, ukuran *batch* 16 dan *epoch* 1500 yaitu model dengan 6 neuron pada *hidden layer* dengan hasil akurasi sebesar 98,14%.

Hasil akurasi tiap percobaan testing tersebut merupakan hasil akurasi dari nilai *output* dalam bentuk normalisasi, untuk mencari akurasi dari hasil prediksi dalam bentuk nilai asli maka dilakukan denormalisasi untuk mengembalikan data *output* menjadi data dengan rentang seperti data awal dengan rumus pada Persamaan (2.15). Berikut merupakan hasil denormalisasi serta perhitungan MAPE dan akurasi dari model arsitektur 7-9-1 dengan fungsi aktivasi ReLu, ukuran batch 8, dan jumlah *epoch* 1000.

**Tabel 4.13** Tabel Perbandingan Hasil *Output* dan Nilai Target (dalam Bentuk Denormalisasi) dari Model Arsitektur 7-9-1 dengan Fungsi Aktivasi ReLu, Ukuran *Batch* 8 dan Jumlah *Epoch* 1000

Bulan	Nilai Prediksi ( $Y_{output}$ )	Nilai Aktual ( $Y_i$ )	Error Multak ( $ Y_{output} - Y_i $ )	Error Relatif ( $(\left \frac{Y_{output} - Y_i}{Y_i}\right  \times 100\%)$ )
Januari 2019	0,550	0,53	0,020	3,76
Februari 2019	-0,416	-0,42	0,004	1,00
Maret 2019	0,361	0,36	0,001	0,27
April 2019	0,465	0,44	0,025	5,66
Mei 2019	0,369	0,35	0,019	5,41
Juni 2019	-0,168	-0,17	0,002	0,91
Juli 2019	0,288	0,20	0,088	43,79
Agustus 2019	0,199	0,19	0,009	4,90
September 2019	0,015	-0,03	0,045	151,49
Oktober 2019	-0,049	-0,04	0,009	23,35
November 2019	0,013	0,01	0,003	29,19
Desember 2019	0,517	0,50	0,017	3,35

Dari nilai prediksi hasil denormalisasi pada Tabel 4.13 di atas, selanjutnya adalah dihitung nilai MAPE dan akurasi dari model tersebut dengan menggunakan Persamaan (2.18) dan Persamaan (2.19).

$$MAPE = \frac{1}{n} \sum_{i=1}^{12} \left| \frac{x_i - y_i}{x_i} \right| \times 100\%$$

$$MAPE = \frac{1}{12} \sum_{i=1}^{12} \left| \frac{Y_{output_i} - Y_i}{Y_i} \right| \times 100\%$$

$$MAPE = \frac{1}{12} (3,76 + 1 + 0,27 + 5,66 + 5,41 + 0,91 + 43,79 + 4,90 + \\ 151,49 + 23,35 + 29,19 + 3,35) \times 100\%$$

$$MAPE = \frac{273,09}{12} \times 100\%$$

$$MAPE = 22,757 \%$$

Kemudian nilai akurasinya adalah sebagai berikut.

$$Akurasi = 100\% - MAPE$$

$$Akurasi = 100\% - 22,76 \%$$

$$Akurasi = 77,24 \%$$

Jadi, pada model arsitektur 7-9-1 dengan parameter fungsi aktivasi ReLu, ukuran batch 8, dan jumlah *epoch* 1000, menghasilkan nilai MAPE hasil denormalisasi sebesar 22,76% dan nilai akurasi hasil denormalisasi sebesar 77,24%.

Perhitungan nilai MAPE dan akurasi hasil denormalisasi tersebut dilakukan ke semua model terbaik dari tiap parameter sehingga didapatkan hasil perbandingan nilai akurasi dari hasil *output* dalam bentuk normalisasi dan hasil *output* dalam bentuk denormalisasi sebagai berikut.

**Tabel 4.14** Hasil Akurasi Hasil *Output* Bentuk Normalisasi dan Denormalisasi dari Model Terbaik dari Setiap Parameter

Fungsi Aktivasi	Ukuran <i>Batch</i>	Jumlah <i>Epoch</i>	Model Arsitektur Jaringan	Akurasi Hasil Normalisasi (%)	Akurasi Hasil Denormalisasi (%)
ReLU	8	1000	7-9-1	98,80	77,24
		1500	7-11-1	98,45	81,80
	16	1000	7-9-1	98,58	84,28
		1500	7-6-1	98,56	68,35
Tanh	8	1000	7-14-1	98,25	81,35
		1500	7-10-1	98,15	77,63
	16	1000	7-14-1	98,26	87,49
		1500	7-6-1	98,14	84,01

Dari Tabel 4.14 di atas, maka dapat dilihat bahwa model arsitektur yang memiliki tingkat akurasi dari hasil denormalisasi tertinggi adalah pada model arsitektur 7-14-1 dengan parameter fungsi aktivasi *tanh*, ukuran *batch* 16, dan jumlah *epoch* 1000 dengan akurasi sebesar 87,49%. Sehingga, model arsitektur ini yang akan menjadi model akhir untuk prediksi inflasi di kota Malang.

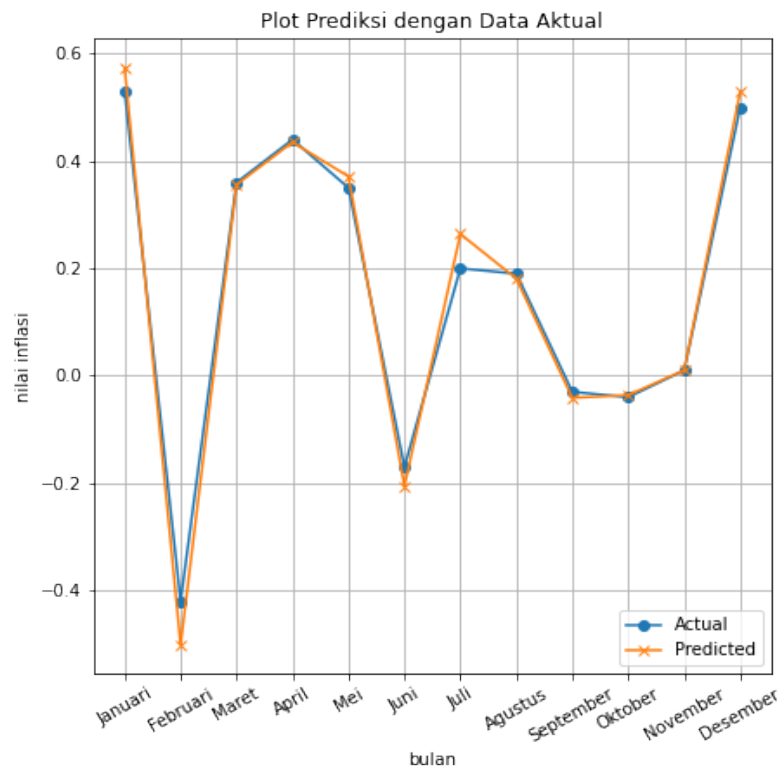
Berikut adalah rincian hasil denormalisasi dari model arsitektur 7-14-1 dengan parameter fungsi aktivasi *tanh*, ukuran *batch* 16, dan jumlah *epoch* 1000.

**Tabel 4.15** Hasil Denormalisasi dari Model Arsitektur 7-14-1 dengan Parameter Fungsi Aktivasi *Tanh*, Ukuran *Batch* 16, dan Jumlah *Epoch* 1000

Bulan	Nilai Prediksi ( $Y_{output}$ )	Nilai Aktual ( $Y_i$ )	Error Multak ( $ Y_{output} - Y_i $ )	Error Relatif ( $(\left \frac{Y_{output} - Y_i}{Y_i}\right  \times 100\%)$ )
Januari 2019	0,574	0,53	0,044	8,27
Februari 2019	-0,501	-0,42	0,081	19,32
Maret 2019	0,356	0,36	0,004	1,10
April 2019	0,435	0,44	0,005	1,12
Mei 2019	0,371	0,35	0,021	6,04
Juni 2019	-0,206	-0,17	0,036	20,93
Juli 2019	0,264	0,20	0,064	31,98
Agustus 2019	0,181	0,19	0,009	4,84
September 2019	-0,041	-0,03	0,011	37,33

Bulan	Nilai Prediksi ( $Y_{output}$ )	Nilai Aktual ( $Y_i$ )	Error Multak ( $ Y_{output} - Y_i $ )	Error Relatif ( $(\frac{ Y_{output} - Y_i }{Y_i} \times 100\%)$ )
Oktober 2019	-0,036	-0,04	0,004	9,39
November 2019	0,01	0,01	0,0004	3,99
Desember 2019	0,529	0,5	0,029	5,78

Dari Tabel 4.15 di atas, maka dapat disimpulkan prediksi tingkat inflasi tertinggi di Kota Malang pada tahun 2019 terjadi pada bulan Januari dengan hasil prediksi sebesar 0,57% dan tingkat inflasi terendah diprediksi terjadi pada bulan Februari dengan hasil prediksi sebesar -0,50%. Hal ini sesuai dengan nilai aktual tingkat inflasi di Kota Malang tahun 2019 dimana tingkat inflasi tertinggi terjadi pada bulan Januari dengan nilai inflasi sebesar 0,53% dan tingkat inflasi terendah terjadi pada bulan Februari dengan nilai inflasi sebesar -0,42%.



**Gambar 4.4** Grafik Perbandingan Hasil *Output* dalam Bentuk Denormalisasi dengan Data Target Asli Tingkat Inflasi di Kota Malang Tahun 2019 dengan Menggunakan Model Arsitektur 7-14-1 dengan Parameter Fungsi Aktivasi *Tanh*, Ukuran *Batch* 16, dan Jumlah *Epoch* 1000

Dari Gambar 4.4 di atas, dapat dilihat bahwa hasil prediksi sudah mendekati nilai target, namun ada beberapa data yang masih belum mencapai target, seperti pada data prediksi bulan Januari, Februari, Mei, dan Juli.

Dengan demikian, berdasarkan kriteria nilai akurasi, maka dengan nilai akurasi model arsitektur 7-14-1 yang menggunakan parameter fungsi aktivasi *tanh*, ukuran *batch* 16, dan jumlah *epoch* 1000 yaitu sebesar 87,49% seperti pada Tabel 4.17, model tersebut dapat dikatakan berjalan baik untuk memprediksi tingkat inflasi di Kota Malang.

#### **4.6 Implementasi Algoritma *Backpropagation* dalam Pandangan Islam**

Algoritma *Backpropagation* merupakan salah satu algoritma yang bisa digunakan untuk melakukan proses prediksi data di masa depan dengan menggunakan data dari masa lalu dan proses pelatihan yang berulang. Data-data yang telah dikumpulkan kemudian dilatih hingga didapatkan model yang terbaik untuk dilakukan training dan prediksi. Selanjutnya, hasil prediksi dievaluasi dengan dihitung nilai *error* atau selisihnya dengan nilai aktual serta dihitung pula nilai akurasi.

Sebagaimana uraian di atas, maka proses yang runtut dan sistematis juga dijelaskan pada salah satu ayat Al-Qur'an yaitu Q.S. Al-Mukminun ayat 12-14 yaitu ayat yang menjelaskan tentang proses penciptaan manusia oleh Allah SWT dari yang awalnya sebagai "saripati" atau esensi yang sangat sederhana hingga menjadi makhluk yang sempurna.

وَلَقَدْ خَلَقْنَا الْإِنْسَانَ مِنْ سُلَّةٍ مِنْ طِينٍ ﴿١٢﴾ ثُمَّ جَعَلْنَاهُ نُطْفَةً فِي قَرَارٍ مَكِينٍ ﴿١٣﴾ ثُمَّ خَلَقْنَا النَّطْفَةَ عَلَقَةً فَخَلَقْنَا الْعَلَقَةَ مُضْغَةً فَخَلَقْنَا الْمُضْغَةَ عِظْمًا فَكَسَوْنَا الْعِظْمَ لَحْمًا ثُمَّ أَنْشَأْنَاهُ خَلْقًا آخَرَ ۚ فَتَبَارَكَ اللَّهُ أَحْسَنُ الْخَالِقِينَ ﴿١٤﴾

*Artinya: “Dan sesungguhnya Kami telah menciptakan manusia dari suatu saripati (berasal) dari tanah. Kemudian Kami jadikan saripati itu air mani (yang disimpan) dalam tempat yang kokoh (rahim). Kemudian air mani itu Kami jadikan segumpal darah, lalu segumpal darah itu Kami jadikan segumpal daging, dan segumpal daging itu Kami jadikan tulang belulang, lalu tulang belulang itu Kami bungkus dengan daging. Kemudian Kami jadikan dia makhluk yang (berbentuk) lain. Maka Maha sucilah Allah, Pencipta Yang Paling Baik.”*

Seperti pada algoritma *backpropagation*, proses penciptaan manusia dalam ayat di atas mengikuti serangkaian langkah atau tahapan yang terstruktur, mulai dari saripati hingga menjadi makhluk yang sempurna. Hal ini mencerminkan proses yang terjadi dalam jaringan saraf tiruan, di mana informasi diproses melalui lapisan-lapisan neuron yang saling terhubung.

Kemudian, ayat di atas juga mencerminkan konsep transformasi dan pembentukan, di mana saripati manusia mengalami serangkaian perubahan dan transformasi menjadi bentuk yang lebih kompleks. Hal ini sejalan dengan konsep *backpropagation* dalam jaringan syaraf tiruan, di mana setiap iterasi dari proses pembelajaran menyebabkan penyesuaian bobot dan parameter yang akhirnya menghasilkan prediksi atau *output* yang lebih baik.

Ayat di atas juga menekankan bahwa penciptaan manusia oleh Allah SWT adalah penuh hikmah dan kebaikan. Demikian pula, penggunaan *backpropagation* atau teknik pembelajaran mesin dimaksudkan untuk menghasilkan model yang dapat mengerti dan memahami pola dalam data dengan lebih baik, sehingga dapat memberikan manfaat bagi manusia.

Selain itu, manusia sebagai makhluk ciptaan Allah SWT yang sebaik-baiknya, hendaklah menggunakan segala yang diberikan oleh Allah SWT dengan memelajari dan mengkaji berbagai aspek kehidupan yang dapat dipelajari dan diteliti dalam mengagumi akan kesempurnaan Allah. Hal ini tersirat dalam Al-Qur'an Surah Ali-Imran ayat 190-191.

إِنَّ فِي خَلْقِ السَّمَوَاتِ وَالْأَرْضِ وَأَخْتِلَافِ اللَّيْلِ وَالنَّهَارِ لَآيَاتٍ لِّأُولِي الْأَلْبَابِ ﴿١٩٠﴾ الَّذِينَ يَذْكُرُونَ  
 اللَّهَ قِيَامًا وَقُعُودًا وَعَلَىٰ جُنُوبِهِمْ وَيَتَفَكَّرُونَ فِي خَلْقِ السَّمَوَاتِ وَالْأَرْضِ رَبَّنَا مَا خَلَقْتَ هَذَا بَطْلًا  
 سُبْحَانَكَ فَقِنَا عَذَابَ النَّارِ ﴿١٩١﴾

*Artinya: "Sesungguhnya dalam penciptaan langit dan bumi, dan silih bergantinya malam dan siang terdapat tanda-tanda bagi orang-orang yang berakal, (yaitu) orang-orang yang mengingat Allah sambil berdiri atau duduk atau dalam keadan berbaring dan mereka memikirkan tentang penciptaan langit dan bumi (seraya berkata): "Ya Tuhan kami, tiadalah Engkau menciptakan ini dengan sia-sia, Maha Suci Engkau, maka peliharalah kami dari siksa neraka."*

Ayat-ayat di atas memuat pesan tentang refleksi atas kebesaran Allah dalam penciptaan alam semesta, serta ajakan untuk selalu mengingat dan memikirkan penciptaan tersebut dengan penuh kekaguman dan kepatuhan kepadanya.

Algoritma *backpropagation* adalah algoritma yang digunakan dalam pembelajaran mesin, khususnya dalam jaringan saraf tiruan (*neural networks*), untuk mengoptimalkan bobot dan bias agar hasil prediksi semakin mendekati nilai yang diharapkan. Ini dilakukan dengan meminimalkan nilai fungsi biaya (*cost function*) melalui penyesuaian bobot dan bias menggunakan turunan parsial.

Hubungan antara surat Ali Imran ayat 190-191 dengan algoritma *backpropagation* dapat dikaitkan melalui konsep refleksi, pemikiran, dan belajar dari penciptaan alam semesta yang luar biasa kompleks. Seperti halnya dalam



algoritma *backpropagation*, manusia diajak untuk merenungkan dan memikirkan penciptaan Allah, serta untuk belajar dan mengambil pelajaran dari kebesarannya.

Dalam konteks ini, mengingat penciptaan alam semesta dengan penuh kekaguman dan kepatuhan kepadanya dapat dianggap sebagai suatu bentuk refleksi dan pembelajaran yang kontemplatif, mirip dengan cara algoritma *backpropagation* memperbarui bobot dan bias berdasarkan evaluasi terhadap hasil prediksi dan target yang diharapkan.

## BAB V

### PENUTUP

#### 5.1 Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan, didapatkan kesimpulan bahwa algoritma *backpropagation* yang diimplementasikan untuk memprediksi tingkat inflasi di Kota Malang diperoleh hasil sebagai berikut:

1. Didapatkan hasil akurasi tertinggi dari penelitian ini yaitu sebesar 87,494% dari model arsitektur jaringan 7-14-1, yaitu model dengan tujuh neuron pada *input layer*, 14 neuron pada *hidden layer*, serta satu neuron di *output layer*, dengan fungsi aktivasi *tanh*, ukuran *batch* sebesar 16, dan jumlah *epoch* 1000. Model arsitektur tersebut memiliki akurasi hasil normalisasi sebesar 98,26%. Dengan demikian, berdasarkan kriteria nilai akurasi, maka model tersebut dapat dikatakan berjalan baik untuk memprediksi tingkat inflasi di Kota Malang.
2. Dengan menggunakan algoritma *backpropagation*, maka tingkat inflasi tertinggi di Kota Malang pada tahun 2019 diprediksi terjadi pada bulan Januari dengan hasil prediksi sebesar 0,57% dan tingkat inflasi terendah diprediksi terjadi pada bulan Februari dengan hasil prediksi sebesar -0,50%. Hal ini sesuai dengan nilai aktual tingkat inflasi di Kota Malang tahun 2019 dimana tingkat inflasi tertinggi terjadi pada bulan Januari dengan nilai inflasi sebesar 0,53% dan tingkat inflasi terendah terjadi pada bulan Februari dengan nilai inflasi sebesar -0,42%.

## 5.2 Saran

Berdasarkan hasil penelitian yang telah dilakukan, terdapat beberapa saran yang diberikan penulis untuk penelitian selanjutnya yaitu sebagai berikut:

1. Mencoba dengan menggunakan data terbaru dengan fitur yang terbaru agar mendapatkan hasil prediksi yang lebih terkini.
2. Menambah variasi pengujian dengan persentase pembagian data *training* dan data *testing* yang berbeda agar mendapatkan akurasi yang lebih baik.

## DAFTAR PUSTAKA

- Amalia, Euis. (2005). *Sejarah Pemikiran Ekonomi Islam dari Masa Klasik Hingga Kontemporer*. Jakarta: Pustaka Asatrus.
- Andrian, Y., & Wayahdi, M. R. (2014). *Analisis Algoritma Inisialisasi Nguyen Widrow Pada Proses Prediksi Curah Hujan Kota Medan Menggunakan Metode Backpropagation Neural Network*. Seminar Nasional Informatika.
- Al-Asqalani, A.-H. I. (2007). *Bulughul Maram*. (A. R. Siddiq, Trans.). Jakarta: Akbar Media.
- Al-Hajjaj, I. A. (2007). *Sahih Muslim*. (N. Al-Khattab, Trans.). Riyadh: Darussalam.
- Al-Malibari, A.-S. Z. (n.d.). *Fath-hul Mu'in*. (Ust. Abul Hiyadh, Trans.). Surabaya: Al-Hidayat.
- Ash'ath, I. H. (2008). *Sunan Abu Dawud*. (N. Al-Khattab, Trans.). Riyadh: Darussalam.
- Ayu Trimulya, S. F. (2015). Implementasi Jaringan Syaraf Tiruan Metode Backpropagation untuk Memprediksi Harga Saham. *Coding Jurnal Komputer dan Aplikasi*, 66-75.
- B Hauriza, M. M. (2021). Prediksi Tingkat Inflasi Bulanan Indonesia Menggunakan Metode Jaringan Saraf Tiruan. *Jurnal Teknologi dan Informasi*, 151-157.
- Bank Indonesia. (2020). *Inflasi*. Retrieved July 27, 2023, from Bank Indonesia Bank Sentral Republik Indonesia: <https://www.bi.go.id/id/fungsi-utama/moneter/inflasi/default.aspx>
- BPS Kota Malang. (2020). *Inflasi Kota Malang 2019*. Malang: BPS Kota Malang.
- BPS Kota Malang. (2023). *Malang Dalam Angka 2023*. Malang: BPS Kota Malang.
- Candra Dewi, M. M. (2013). Perbandingan Akurasi Backpropagation Neural Network dan ANFIS untuk Memprediksi Cuaca. *Journal of Scientific Modeling & Computation*.

- D Rakhmawati, M. N. (2023). Perbandingan Presiksi Inflasi di Indonesia Menggunakan Model Holt-Winters Exponential Smoothing Additive dan Multiplicative. *Jurnal Pendidikan Edutama*, 55-60.
- Fathullah, D. A. (n.d.). *Al-Quran Al-Hadi*. Jakarta: Pusat Kajian Hadis Jakarta.
- Guido, A. C. (2017). *Introduction to Machine Learning with Python*. United States of America.: O'Reilly Media, Inc
- Hadis riwayat Abu Dawud dalam kitab Al-Ijarah no.3447
- Hadis riwayat Muslim dalam kitab Bulughul Maram no.784
- Hadis riwayat Muslim no. 2961
- Hadis riwayat Shahih Muslim no. 3012
- Hartati, S. (2021). *Kecerdasan Buatan Berbasis Pengetahuan*. UGM Press.
- Haykin, Simon. (1999). *Neural Networks A Comprehensive Foundation*. Prentice-Hall Inc., USA.
- Heaton, Jeff. (2008). *Introduction to Neural Networks for Java, 2nd Edition (2nd. ed.)*. Heaton Research, Inc.
- K Wong, A. W. (2019). Prediksi tingkat Inflasi dengan Menggunakan Metode Backpropagation Neural Network. *Sains, Aplikasi, Komputasi dan Teknologi Informasi*, 8-13.
- Kemenag RI. (2019). *Qur'an Kemenag*. Kemenag. <https://quran.kemenag.go.id/>
- Khairunnisa, J., & Syaharuddin, S. (2022). Prediksi Inflasi di Kota Mataram Menggunakan Bacpropagation Neural Network. In *Seminar Nasional Lppm Ummat* (Vol. 1, pp. 151-157)
- Matondang, Z. A. (2013). *Jaringan Syaraf Tiruan dengan Algoritma Backpropagation untuk Penentuan Kelulusan Sidang Skripsi*. Pelita Informatika Budi Darma.
- Meizir, & Rikumahu, B. (2019). Prediction of Agriculture and Mining Stock Value Listed in Kompas100 Index Using Artificial Neural Network Backpropagation. *2019 7th International Conference on Information and Communication Technology (ICoICT)*. doi:10.1109/icoict.2019.8835284
- Hutabarat, M. A., Julham, M., & Wanto, A. (2018). Penerapan Backpropagation dalam Memprediksi Produksi Tanaman Padi Sawah menurut Kabupaten/Kota di Sumatera Utara. *semanTIK*, 77-86.

- R Sovia, M. Y. (2018). Bank Indonesia Interest Rate Prediction and Forecast With Backpropagation Neural Network. *International Conference on Information Technology Systems and Innovation (ICITSI)*, 429-435.
- Ramadhona, G. (2018). *Prediksi Produktivitas Padi Menggunakan Jaringan Syaraf Tiruan Backpropagation*. Malang: Universitas Brawijaya.
- Salam, W. R. (2020). Inflasi Ditengah Pandemi dalam Perspektif Islam. *Jurnal Syntax Transformation*, 187-192.
- Shihab, M. Q. (2002). *Tafsir Al-Mishbah*. Jakarta: Penerbit Lentera Hati.
- Sukirno, S. (2008). *Pengantar Teori Makro Ekonomi*. Jakarta: PT Raja Grafindo Persada.
- Usmani, M. M. (1998). *An Introductuion to Islamic Finance*. Arham Shamsi.
- Wadi, H. (2021). *Implementasi Jaringan Syaraf Tiruan Backpropagation menggunakan PYHTON GUI*. Jakarta: Turida Publisher.

## LAMPIRAN

**Lampiran 1** Data Inflasi Bulanan Kota Malang Tiap Kelompok dari 2005 – 2019

kelompok bulan	Bahan Makanan	Makanan Jadi, Minuman, Rokok, dan Tembakau	Perumahan, Air, Listrik, Gas, dan Bahan Bakar	Sandang	Kesehatan	Pendidikan, Rekreasi dan Olahraga	Transpor, Komunikasi, dan Jasa Keuangan	Umum
Januari 2005	1.86	0.35	1.71	-0.28	0.92	0	0.15	0.95
Februari 2005	-3.39	1.49	0.18	-0.26	-0.04	0	0	-0.43
Maret 2005	0.26	1.15	0.81	0.54	0.65	0.24	8.12	1.83
April 2005	0.3	1.49	2.16	0.44	0.6	0.05	1.49	1.99
Mei 2005	-0.81	-0.06	0.04	-0.22	0.97	0	0.08	-0.12
Juni 2005	0.6	-0.12	0.2	0.28	0.47	-0.11	0	0.18
Juli 2005	3.54	1.76	0.26	0.92	-0.07	0.02	0	1.18
Agustus 2005	-0.42	0.44	0.43	0.76	0.77	0.67	0.15	0.26
September 2005	-0.68	1.01	1.28	1.78	0.5	6.55	0.2	1.01
Oktober 2005	7.38	0.84	6.6	1.72	0.5	0.14	26.47	7.77
November 2005	1.84	1.26	1.37	0.27	0.25	0.68	0.21	1.06
Desember 2005	-1.27	0.16	0.7	1.51	0.06	0.25	0.05	0.06
Januari 2006	4.32	1.08	0.58	1.11	1.24	0.1	-0.07	1.33
Februari 2006	1.34	1.46	1.61	0.28	0.32	0.14	0.08	1
Maret 2006	-1.03	-0.31	0.18	-0.16	0.12	0	0.28	-0.17
April 2006	1.74	0.31	0.42	1.34	0.54	0.48	0.53	0.76
Mei 2006	0.02	0.12	0.25	3.04	0.33	0	0.42	0.36
Juni 2006	1.14	0.01	-0.23	-1.27	0.71	0.02	0.04	0.14
Juli 2006	1.1	0.02	0.14	2.03	-0.11	0.68	0	0.44
Agustus 2006	-2.6	0.1	0.37	0.36	0.29	6.99	0	0.07
September 2006	0.11	0.02	0.07	0.81	0.02	0.05	-0.03	0.09
Oktober 2006	1.98	0.09	0.3	0.64	0.19	0	0	0.55
November 2006	-0.7	0.13	0.13	0.91	0.12	0.02	-0.01	-0.04
Desember 2006	4.32	1.26	0.3	0.23	0.1	0.31	0.01	1.24
Januari 2007	3.04	0.27	0.17	0.01	0.58	0	0.02	0.77
Februari 2007	-0.27	0.87	0.16	1.2	2.96	0.07	0.02	0.35
Maret 2007	-0.64	0.23	1.1	-0.25	0.22	0	0.03	0.17
April 2007	-0.44	0.06	0.28	1	0.06	0.01	0.12	0.08
Mei 2007	0.49	0.29	-0.01	-0.25	0.61	0	0.07	0.18
Juni 2007	-0.79	0.06	0.09	-0.48	0.25	0.18	0.06	-0.13
Juli 2007	1.54	0.05	0.06	0.99	0.21	0.78	0	0.48
Agustus 2007	0.5	0.18	0.37	0.89	0.05	4.29	-0.01	0.59
September 2007	2.27	-0.02	0.95	3.17	1.36	0.3	0.12	1.03
Oktober 2007	1.46	0.99	0.68	2.79	0.36	0	0.06	0.86
November 2007	1.48	0.11	0.49	2.95	0.21	0	0.27	0.72

Desember 2007	2.94	-0.17	0.06	0.17	0.41	0	0.07	0.68
Januari 2008	5.03	2.85	1.25	2.75	0.72	0.12	0.07	2.17
Februari 2008	0	0.87	0.32	0.94	1.86	0.21	0.05	0.4
Maret 2008	1.58	4.72	0.25	2.06	0.8	0.15	0.07	1.44
April 2008	1.31	0.86	5.46	-1.71	0.73	0	-0.91	1.54
Mei 2008	-0.27	-0.12	-0.05	-0.34	0.26	0.32	2.58	0.33
Juni 2008	1.58	0.5	2.49	0.31	1.1	0.83	9.2	2.77
Juli 2008	2.33	1.84	1.3	1.6	2.79	0.41	0.8	1.54
Agustus 2008	1.35	0.01	0.41	-0.67	0.23	0	0.49	0.45
September 2008	1.82	0.22	0.86	0.69	0.08	2.01	0.3	0.91
Oktober 2008	0.81	1.5	0.24	0.4	0.25	0.25	0	0.56
November 2008	-1.4	0.35	0.52	1.25	0.22	0.01	-0.25	-0.11
Desember 2008	-0.28	1.3	0.88	2.29	0.43	0.02	-3.18	-0.07
Januari 2009	1.58	0.85	0.24	1.15	0	0.37	-2.14	0.28
Februari 2009	0.83	0.57	1.02	3.44	0.05	0	-0.55	0.39
Maret 2009	1.07	0.88	0.48	0.64	1.12	-0.08	0.44	0.61
April 2009	-1.6	1.26	-0.33	-1.89	0.24	0.49	0.22	-0.21
Mei 2009	-0.54	0.88	0.04	-0.32	0.2	0.03	-0.07	0.04
Juni 2009	0.7	0.33	0.15	0.41	0.1	0.27	0.12	0.32
Juli 2009	1.09	0.02	0.09	-0.08	0.21	0.35	0.06	0.31
Agustus 2009	-0.02	1.31	-0.06	0.04	0.06	4.33	0.3	0.67
September 2009	1.16	0.82	0.03	0.65	0.01	0.04	0.95	0.62
Oktober 2009	0.67	0.07	0.17	-0.1	-0.08	0.98	-0.45	0.21
November 2009	-1.33	0.26	0.2	0.97	0.07	0.01	0.03	-0.15
Desember 2009	1.09	0.59	0.25	0.87	0.01	0.03	0.09	0.48
Januari 2010	2.2	1.02	1.34	0	0.15	-0.73	0.01	0.79
Februari 2010	1.24	-0.03	2.17	0.02	0.35	-0.37	0.01	0.37
Maret 2010	-1.02	0	-0.01	0.34	0.14	-0.14	0.1	-0.17
April 2010	0.38	0.01	-0.1	0.03	0.12	0.31	0.09	0.14
Mei 2010	0.81	0.52	0.29	-0.04	0.12	1.01	0.04	0.35
Juni 2010	2.93	0.19	0.24	0.01	-0.04	0.73	-0.1	0.74
Juli 2010	6.01	0.6	0.9	0.18	0.36	-0.37	0.18	1.71
Agustus 2010	0.16	0.57	0.17	6.77	0.51	0.09	0.06	0.79
September 2010	-1.4	0.12	0.61	0	0.25	1.7	0.94	0.05
Oktober 2010	-0.37	0.2	1	1.48	-0.01	1.17	-0.69	0.19
November 2010	1.18	0.17	1.74	0.03	-0.02	0.48	0.11	0.68
Desember 2010	2.4	0.05	0.72	0.19	0.16	1.02	0.35	0.88
Januari 2011	1.7	0.61	0.19	0.07	0.25	0.47	0.24	0.67
Februari 2011	-0.01	0.15	0.4	-0.33	0.41	0.15	0.16	0.14
Maret 2011	-1.38	0.17	0.55	0.81	0.19	0	0.29	-0.09
April 2011	-2.48	0.31	0.17	1.22	0.21	0	0.04	-0.42
Mei 2011	0.2	-0.01	0.14	0.54	0	0	0.04	0.1
Juni 2011	1.64	0.27	0.31	0.57	0.07	0.1	0.02	0.56



Juli 2011	1.75	0.98	0.17	0.45	0.12	0.52	0.02	0.73
Agustus 2011	0.89	0.32	0.11	2.54	0.07	5.18	0.22	0.94
September 2011	0.48	0.24	0.02	1.02	0.05	0	-0.04	0.22
Oktober 2011	-0.01	0.16	0.25	-1.34	0.1	1.54	-0.22	0.12
November 2011	0.9	0.26	-0.17	1.29	0.03	0.01	0.23	0.34
Desember 2011	2.3	0.07	0.16	0.09	0.42	0.06	0.21	0.67
Januari 2012	0.06	0.42	0.27	-0.41	0.29	0.07	0.8	0.27
Februari 2012	0.32	0.28	0.23	0.95	0.02	0.01	-0.38	0.18
Maret 2012	-0.78	0.4	0.11	-0.21	0.02	-0.03	0.7	0.01
April 2012	0.15	1	0.26	-0.4	0.07	0	0.01	0.27
Mei 2012	-0.52	0.27	0.17	-0.24	0.15	-0.27	0.59	0.09
Juni 2012	1.2	0.46	0.29	0.35	0.36	0.06	0.37	0.54
Juli 2012	1.24	0.65	-0.01	0.12	0.17	0	0.2	0.48
Agustus 2012	2	0.72	0.28	1.03	0.09	0.07	1.46	1.04
September 2012	-0.59	0.48	0.32	2.48	0.23	5.03	-0.63	0.52
Oktober 2012	0.31	0.11	0.02	0.5	0.01	1.01	-0.07	0.22
November 2012	-0.12	0.65	0.32	0.14	0.08	0.11	0.29	0.23
Desember 2012	1.88	0.27	0.12	0.08	0.09	0	0.95	0.7
Januari 2013	3.85	0.04	0.05	-0.06	0.42	0.03	-0.65	0.94
Februari 2013	2.64	0	0.43	0.26	0.63	-0.75	0.09	0.88
Maret 2013	2.79	0.13	0.25	0	0.25	-0.57	0.8	0.93
April 2013	-1.17	0.48	0.23	0.3	0.16	-0.66	0.02	-0.21
Mei 2013	-1.72	0	0.05	0.14	0.51	-0.7	0.07	-0.35
Juni 2013	1.04	0.04	0.48	0.07	0.04	-0.39	3.61	0.91
Juli 2013	6.32	0.38	0.94	0.15	0.67	0.07	9.49	3.49
Agustus 2013	0.79	0.17	0.67	0.8	0.46	2.56	0.76	0.77
September 2013	-0.83	0.08	0.33	0.21	0.16	2.38	-0.33	-0.57
Oktober 2013	-0.7	0.29	0.52	0.99	0.84	-1.15	0.23	0.16
November 2013	0.09	0.02	0.07	0.01	0.89	-0.14	0.11	0.23
Desember 2013	1.65	0.07	0.36	-1.77	0.54	0.07	0.52	0.53
Januari 2014	1.67	0.31	1.22	0.55	0.05	0.08	0.25	0.76
Februari 2014	-0.57	2.8	-0.08	0.56	0.3	0.18	0.22	0.31
Maret 2014	0.38	0.54	0.16	0.39	0.25	0.05	0.98	0.43
April 2014	-2.5	0.01	0.09	-0.51	1.55	0.27	1.2	-0.31
Mei 2014	-0.64	1	0.52	0.17	0.59	0.09	0.71	0.37
Juni 2014	0.81	0.5	0.14	0.45	0.17	0	0.05	0.31
Juli 2014	0.9	0.36	0.16	1.89	0.97	0.39	0.23	0.49
Agustus 2014	0.77	0.35	0.62	0.22	0.18	0.47	0.24	0.47
September 2014	-0.08	0.2	0.98	-0.41	0.09	1.11	-0.52	0.26
Oktober 2014	0.2	1.27	0.64	0.08	0.27	0.02	-0.15	0.4
November 2014	2.37	0.12	0.32	-0.09	0.4	0	4.88	1.51
Desember 2014	3.51	1.2	1.89	0.36	1.32	0.26	6.36	2.42
Januari 2015	0.18	0.55	0.79	0.62	0.46	0.14	-1.53	0.07

Februari 2015	-1.3	0.09	0.14	0.88	0.83	0.11	-2.3	-0.57
Maret 2015	-0.74	0.37	0.36	0.08	0.43	0.12	1.42	0.34
April 2015	-0.73	0.61	0.07	0.25	0.53	0.32	2.11	0.49
Mei 2015	1.2	0.31	0.01	0.35	0.16	0.69	0.41	0.45
Juni 2015	1.12	0.61	0.06	0.66	0.06	0.17	0.08	0.38
Juli 2015	0.88	-0.04	0.22	-0.36	0.13	0.16	1.76	0.57
Agustus 2015	1.12	0.45	0.48	-0.27	0.01	3.04	-1.83	0.28
September 2015	-0.74	0.6	0.28	1.1	0.41	3.66	-1.19	0.21
2015	-0.31	0.06	0.26	0.01	0.1	0.05	0.01	0.03
November 2015	0.49	0.84	0.01	-1.89	0.08	0.14	0.11	0.16
Desember 2015	3.91	0.07	0.16	-0.01	0.38	0	0.52	0.89
Januari 2016	2.63	0.89	0.52	1.68	0.19	0.1	-1.44	0.58
Februari 2016	0.63	0.14	-0.29	0.99	0.12	0	-1.38	-0.15
Maret 2016	0.64	0.21	-0.09	0.56	0.34	0.1	-0.88	0.02
April 2016	-1.01	0.61	-0.03	0.21	0.61	0.01	-1.7	-0.4
Mei 2016	-0.48	0.45	-0.14	0.94	0.46	0.03	0.7	0.15
Juni 2016	1.17	0.99	-0.13	0.3	0.31	0.17	1.13	0.63
Juli 2016	1.95	0.33	0.41	1.5	0.16	0.14	0.74	0.78
Agustus 2016	-1.22	0.02	0.81	0.1	0.12	2.68	-1.34	-0.03
September 2016	-0.26	0.13	-0.07	-0.05	0.07	3.36	-0.63	0.17
Oktober 2016	-0.97	0.14	-0.03	-0.64	0.03	0.06	0.01	-0.2
November 2016	1.58	0.06	0.46	-0.09	0.12	0	0.1	0.45
Desember 2016	0.25	0.07	0.02	-0.9	0.17	-0.04	2.97	0.58
Januari 2017	2.4	0.96	1.6	0.65	0.99	0.03	1.79	1.45
Februari 2017	0.78	0.37	1.21	1.72	0.48	0.2	-2.05	0.24
Maret 2017	-1.43	0.03	0.43	-0.03	0.38	0.39	0.12	-0.09
April 2017	-0.82	0.05	1.32	0.73	0.33	-0.14	0.66	0.35
Mei 2017	1.09	0.46	0.33	0.17	0.08	1.1	1.72	0.82
Juni 2017	-0.68	-0.05	0.92	0.38	0.08	0.85	0.88	0.37
Juli 2017	-0.16	0.72	-0.04	0.15	0.5	0.4	0.77	0.3
Agustus 2017	-1.68	-0.15	0.4	0.87	-0.32	0	-0.86	-0.57
September 2017	-0.46	0.11	0.09	0.54	0.2	0.09	0.22	0.05
Oktober 2017	0.07	-0.02	0.23	-0.11	0.15	-0.02	-0.22	0.02
November 2017	0.79	-0.02	0.23	0.1	1.33	0.05	-0.06	0.27
Desember 2017	2.36	0.32	-0.13	-0.01	0.01	0	0.2	0.49
Januari 2018	2.65	0.45	0.56	0.09	0.11	0.7	-0.01	0.69
Februari 2018	-0.12	0.04	-0.01	0.01	1.63	0.44	-1.3	0.17
Maret 2018	-0.23	0.05	0.07	0.02	0.14	0.36	0.53	0.12
April 2018	0.38	0.39	0.32	0.02	0.15	0.29	-0.33	0.14
Mei 2018	-1.55	0.06	-0.28	0	0.79	0.3	2.23	0.29
Juni 2018	0.06	0.12	0.26	0.05	0.01	-0.16	1.02	0.25
Juli 2018	1.04	0.01	0.78	0	0.11	0.27	-0.81	0.21
Agustus 2018	-1.42	0.02	0.12	3.4	0.69	-0.21	-1.04	0.05

September 2018	-1.17	0.01	-0.09	0.01	0.16	0.34	-0.76	-0.31
Oktober 2018	0.57	0	0.25	0	0.42	0.44	0.12	0.3
November 2018	0.48	-0.07	0.08	0	0.06	0.16	1.31	0.37
Desember 2018	1.49	0.12	0.3	0.25	0.11	0.37	1.3	0.65
Januari 2019	1.14	0.27	-0.08	0.86	0.19	0.3	1.11	0.53
Februari 2019	-1.57	0.2	0.2	0.36	-0.11	0	-1.2	-0.42
Maret 2019	0.22	0.09	-0.04	0.06	0.09	0	1.66	0.36
April 2019	0.73	0	0.01	-0.09	0.04	-0.05	1.62	0.44
Mei 2019	1.23	0.16	0.13	0.95	-0.07	-0.01	0.05	0.35
Juni 2019	-0.49	0.05	-0.08	0.44	-0.03	0.08	-0.52	-0.17
Juli 2019	2.01	0.17	0.01	0.59	0	0.01	-1.25	0.2
Agustus 2019	0.26	0.03	0.36	1.3	0.03	0.3	-0.34	0.19
September 2019	-0.75	0.82	0.07	0.66	0	0	-0.38	-0.03
Oktober 2019	-0.9	0.08	0.04	-0.45	0.73	0	0.48	-0.04
November 2019	0.19	0.08	0.19	-0.22	0.09	0.16	-0.43	0.01
Desember 2019	1.01	1.03	0.34	-0.36	0.23	0.05	0.26	0.5

### Lampiran 2 Data Inflasi Hasil Normalisasi

kelompok bulan	Bahan Makanan	Makanan Jadi, Minuman, Rokok, dan Tembakau	Perumahan, Air, Listrik, Gas, dan Bahan Bakar	Sandang	Kesehatan	Pendidikan, Rekreasi dan Olahraga	Transpor, Komunikasi, dan Jasa Keuangan	Umum
Januari 2005	0.538719	0.218091	0.364935	0.267321	0.440244	0.22715	0.201079	0.264029
Februari 2005	0.1	0.422068	0.166234	0.2694	0.176829	0.22715	0.196526	0.115108
Maret 2005	0.405014	0.361233	0.248052	0.35254	0.366159	0.253686	0.443002	0.358993
April 2005	0.408357	0.422068	0.423377	0.342148	0.352439	0.232678	0.241754	0.376259
Mei 2005	0.315599	0.144732	0.148052	0.273557	0.453963	0.22715	0.198954	0.148561
Juni 2005	0.433426	0.133996	0.168831	0.32552	0.316768	0.214988	0.196526	0.180935
Juli 2005	0.679109	0.470378	0.176623	0.392032	0.168598	0.229361	0.196526	0.288849
Agustus 2005	0.348189	0.234195	0.198701	0.375404	0.399085	0.301229	0.201079	0.189568
September 2005	0.326462	0.336183	0.309091	0.481409	0.325	0.951351	0.202597	0.270504
Oktober 2005	1	0.305765	1	0.475173	0.325	0.242629	1	1
November 2005	0.537047	0.380915	0.320779	0.32448	0.256402	0.302334	0.202901	0.275899
Desember 2005	0.277159	0.184095	0.233766	0.453349	0.204268	0.254791	0.198044	0.167986
Januari 2006	0.74429	0.348708	0.218182	0.411778	0.528049	0.238206	0.194401	0.305036
Februari 2006	0.495265	0.4167	0.351948	0.32552	0.27561	0.242629	0.198954	0.269424
Maret 2006	0.297214	0.1	0.166234	0.279792	0.220732	0.22715	0.205025	0.143165
April 2006	0.528691	0.210934	0.197403	0.435681	0.335976	0.280221	0.212614	0.243525
Mei 2006	0.384958	0.176938	0.175325	0.612356	0.278354	0.22715	0.209275	0.20036
Juni 2006	0.478552	0.157256	0.112987	0.164434	0.382622	0.229361	0.19774	0.176619
Juli 2006	0.475209	0.159046	0.161039	0.50739	0.157622	0.302334	0.196526	0.208993

Agustus 2006	0.166017	0.17336	0.190909	0.333834	0.267378	1	0.196526	0.169065
September 2006	0.392479	0.159046	0.151948	0.3806	0.193293	0.232678	0.195616	0.171223
Oktober 2006	0.548747	0.171571	0.181818	0.362933	0.239939	0.22715	0.196526	0.220863
November 2006	0.324791	0.178728	0.15974	0.390993	0.220732	0.229361	0.196223	0.157194
Desember 2006	0.74429	0.380915	0.181818	0.320323	0.215244	0.261425	0.19683	0.295324
Januari 2007	0.637326	0.203777	0.164935	0.29746	0.346951	0.22715	0.197133	0.244604
Februari 2007	0.360724	0.311133	0.163636	0.421132	1	0.234889	0.197133	0.199281
Maret 2007	0.329805	0.19662	0.285714	0.270439	0.248171	0.22715	0.197437	0.179856
April 2007	0.346518	0.166203	0.179221	0.400346	0.204268	0.228256	0.200169	0.170144
Mei 2007	0.424234	0.207356	0.141558	0.270439	0.355183	0.22715	0.198651	0.180935
Juni 2007	0.31727	0.166203	0.154545	0.246536	0.256402	0.247052	0.198347	0.147482
Juli 2007	0.511978	0.164414	0.150649	0.399307	0.245427	0.313391	0.196526	0.213309
Agustus 2007	0.42507	0.187674	0.190909	0.388915	0.201524	0.701474	0.196223	0.22518
September 2007	0.572981	0.151889	0.266234	0.625866	0.560976	0.260319	0.200169	0.272662
Oktober 2007	0.505292	0.332604	0.231169	0.586374	0.286585	0.22715	0.198347	0.254317
November 2007	0.506964	0.175149	0.206494	0.603002	0.245427	0.22715	0.204722	0.239209
Desember 2007	0.628969	0.12505	0.150649	0.314088	0.300305	0.22715	0.198651	0.234892
Januari 2008	0.803621	0.665408	0.305195	0.582217	0.385366	0.240418	0.198651	0.395683
Februari 2008	0.383287	0.311133	0.184416	0.394111	0.698171	0.250369	0.198044	0.204676
Maret 2008	0.51532	1	0.175325	0.510508	0.407317	0.243735	0.198651	0.316906
April 2008	0.492758	0.309344	0.851948	0.118707	0.38811	0.22715	0.168904	0.327698
Mei 2008	0.360724	0.133996	0.136364	0.261085	0.259146	0.262531	0.27484	0.197122
Juni 2008	0.51532	0.24493	0.466234	0.328637	0.489634	0.318919	0.475784	0.460432
Juli 2008	0.577994	0.484692	0.311688	0.462702	0.953354	0.272482	0.220809	0.327698
Agustus 2008	0.4961	0.157256	0.196104	0.22679	0.250915	0.22715	0.2114	0.210072
September 2008	0.535376	0.194831	0.254545	0.368129	0.209756	0.449386	0.205632	0.259712
Oktober 2008	0.450975	0.423857	0.174026	0.337991	0.256402	0.254791	0.196526	0.221942
November 2008	0.266295	0.218091	0.21039	0.426328	0.248171	0.228256	0.188938	0.14964
Desember 2008	0.359889	0.388072	0.257143	0.534411	0.305793	0.229361	0.1	0.153957
Januari 2009	0.51532	0.307555	0.174026	0.415935	0.187805	0.268059	0.131568	0.191727
Februari 2009	0.452646	0.257455	0.275325	0.653926	0.201524	0.22715	0.179831	0.203597
Maret 2009	0.472702	0.312922	0.205195	0.362933	0.495122	0.218305	0.209882	0.227338
April 2009	0.249582	0.380915	0.1	0.1	0.253659	0.281327	0.203204	0.138849
Mei 2009	0.338162	0.312922	0.148052	0.263164	0.242683	0.230467	0.194401	0.165827
Juni 2009	0.441783	0.214513	0.162338	0.33903	0.215244	0.257002	0.200169	0.196043
Juli 2009	0.474373	0.159046	0.154545	0.288106	0.245427	0.265848	0.198347	0.194964
Agustus 2009	0.381616	0.389861	0.135065	0.300577	0.204268	0.705897	0.205632	0.233813
September 2009	0.480223	0.302187	0.146753	0.363972	0.190549	0.231572	0.225363	0.228417
Oktober 2009	0.439276	0.167992	0.164935	0.286028	0.165854	0.335504	0.182867	0.184173
November 2009	0.272145	0.201988	0.168831	0.397229	0.207012	0.228256	0.197437	0.145324
Desember 2009	0.474373	0.261034	0.175325	0.386836	0.190549	0.230467	0.199258	0.213309
Januari 2010	0.567131	0.337972	0.316883	0.29642	0.228963	0.146437	0.19683	0.246763
Februari 2010	0.486908	0.150099	0.424675	0.298499	0.283841	0.186241	0.19683	0.201439

Maret 2010	0.29805	0.155467	0.141558	0.331755	0.22622	0.211671	0.199562	0.143165
April 2010	0.415042	0.157256	0.12987	0.299538	0.220732	0.261425	0.199258	0.176619
Mei 2010	0.450975	0.248509	0.180519	0.292263	0.220732	0.338821	0.19774	0.199281
Juni 2010	0.628134	0.189463	0.174026	0.29746	0.176829	0.307862	0.193491	0.241367
Juli 2010	0.885515	0.262823	0.25974	0.315127	0.286585	0.186241	0.20199	0.346043
Agustus 2010	0.396657	0.257455	0.164935	1	0.327744	0.237101	0.198347	0.246763
September 2010	0.266295	0.176938	0.222078	0.29642	0.256402	0.415111	0.225059	0.166906
Oktober 2010	0.352368	0.191252	0.272727	0.450231	0.185061	0.356511	0.175582	0.182014
November 2010	0.481894	0.185885	0.368831	0.299538	0.182317	0.280221	0.199865	0.234892
Desember 2010	0.583844	0.164414	0.236364	0.316166	0.231707	0.339926	0.20715	0.256475
Januari 2011	0.525348	0.264612	0.167532	0.303695	0.256402	0.279115	0.203811	0.233813
Februari 2011	0.382451	0.182306	0.194805	0.262125	0.300305	0.243735	0.201383	0.176619
Maret 2011	0.267967	0.185885	0.214286	0.3806	0.239939	0.22715	0.205329	0.151799
April 2011	0.176045	0.210934	0.164935	0.42321	0.245427	0.22715	0.19774	0.116187
Mei 2011	0.4	0.153678	0.161039	0.35254	0.187805	0.22715	0.19774	0.172302
Juni 2011	0.520334	0.203777	0.183117	0.355658	0.207012	0.238206	0.197133	0.221942
Juli 2011	0.529526	0.330815	0.164935	0.343187	0.220732	0.284644	0.197133	0.240288
Agustus 2011	0.45766	0.212724	0.157143	0.560393	0.207012	0.799877	0.203204	0.26295
September 2011	0.423398	0.19841	0.145455	0.402425	0.201524	0.22715	0.195312	0.185252
Oktober 2011	0.382451	0.184095	0.175325	0.157159	0.215244	0.39742	0.189848	0.17446
November 2011	0.458496	0.201988	0.120779	0.430485	0.196037	0.228256	0.203508	0.198201
Desember 2011	0.575487	0.167992	0.163636	0.305774	0.303049	0.233784	0.202901	0.233813
Januari 2012	0.388301	0.230616	0.177922	0.253811	0.267378	0.234889	0.220809	0.190647
Februari 2012	0.410028	0.205567	0.172727	0.39515	0.193293	0.228256	0.184992	0.180935
Maret 2012	0.318106	0.227038	0.157143	0.274596	0.193293	0.223833	0.217774	0.16259
April 2012	0.395822	0.334394	0.176623	0.25485	0.207012	0.22715	0.19683	0.190647
Mei 2012	0.339833	0.203777	0.164935	0.271478	0.228963	0.197297	0.214435	0.171223
Juni 2012	0.483565	0.237773	0.180519	0.332794	0.286585	0.233784	0.207757	0.219784
Juli 2012	0.486908	0.271769	0.141558	0.308891	0.234451	0.22715	0.202597	0.213309
Agustus 2012	0.550418	0.284294	0.179221	0.403464	0.2125	0.234889	0.240843	0.273741
September 2012	0.333983	0.241352	0.184416	0.554157	0.250915	0.783292	0.177403	0.217626
Oktober 2012	0.409192	0.175149	0.145455	0.348383	0.190549	0.338821	0.194401	0.185252
November 2012	0.373259	0.271769	0.184416	0.31097	0.209756	0.239312	0.205329	0.186331
Desember 2012	0.54039	0.203777	0.158442	0.304734	0.2125	0.22715	0.225363	0.23705
Januari 2013	0.705014	0.162624	0.149351	0.290185	0.303049	0.230467	0.176796	0.26295
Februari 2013	0.6039	0.155467	0.198701	0.323441	0.360671	0.144226	0.199258	0.256475
Maret 2013	0.616435	0.178728	0.175325	0.29642	0.256402	0.164128	0.220809	0.261871
April 2013	0.285515	0.241352	0.172727	0.327598	0.231707	0.154177	0.197133	0.138849
Mei 2013	0.239554	0.155467	0.149351	0.31097	0.327744	0.149754	0.198651	0.123741
Juni 2013	0.470195	0.162624	0.205195	0.303695	0.19878	0.184029	0.306105	0.259712
Juli 2013	0.911421	0.223459	0.264935	0.312009	0.371646	0.234889	0.484587	0.538129
Agustus 2013	0.449304	0.185885	0.22987	0.379561	0.314024	0.510197	0.219595	0.244604
September 2013	0.313928	0.169781	0.185714	0.318245	0.231707	0.490295	0.186509	0.1

Oktober 2013	0.324791	0.207356	0.21039	0.399307	0.418293	0.1	0.203508	0.178777
November 2013	0.390808	0.159046	0.151948	0.29746	0.432012	0.211671	0.199865	0.186331
Desember 2013	0.52117	0.167992	0.18961	0.112471	0.335976	0.234889	0.21231	0.218705
Januari 2014	0.522841	0.210934	0.301299	0.35358	0.201524	0.235995	0.204115	0.243525
Februari 2014	0.335655	0.656461	0.132468	0.354619	0.270122	0.247052	0.203204	0.194964
Maret 2014	0.415042	0.252087	0.163636	0.336952	0.256402	0.232678	0.226273	0.207914
April 2014	0.174373	0.157256	0.154545	0.243418	0.61311	0.257002	0.232951	0.128058
Mei 2014	0.329805	0.334394	0.21039	0.314088	0.349695	0.237101	0.218078	0.201439
Juni 2014	0.450975	0.24493	0.161039	0.343187	0.234451	0.22715	0.198044	0.194964
Juli 2014	0.458496	0.219881	0.163636	0.492841	0.453963	0.27027	0.203508	0.214388
Agustus 2014	0.447632	0.218091	0.223377	0.319284	0.237195	0.279115	0.203811	0.21223
September 2014	0.376602	0.191252	0.27013	0.253811	0.2125	0.349877	0.180742	0.189568
Oktober 2014	0.4	0.382704	0.225974	0.304734	0.26189	0.229361	0.191973	0.204676
November 2014	0.581337	0.176938	0.184416	0.287067	0.297561	0.22715	0.344654	0.32446
Desember 2014	0.676602	0.370179	0.388312	0.333834	0.55	0.255897	0.389578	0.422662
Januari 2015	0.398329	0.253877	0.245455	0.360855	0.314024	0.242629	0.150084	0.169065
Februari 2015	0.274652	0.171571	0.161039	0.387875	0.415549	0.239312	0.126712	0.1
Maret 2015	0.321448	0.22167	0.18961	0.304734	0.305793	0.240418	0.239629	0.198201
April 2015	0.322284	0.264612	0.151948	0.322402	0.333232	0.262531	0.260573	0.214388
Mei 2015	0.483565	0.210934	0.144156	0.332794	0.231707	0.30344	0.208971	0.210072
Juni 2015	0.47688	0.264612	0.150649	0.365012	0.204268	0.245946	0.198954	0.202518
Juli 2015	0.456825	0.14831	0.171429	0.259007	0.223476	0.24484	0.249949	0.223022
Agustus 2015	0.47688	0.235984	0.205195	0.26836	0.190549	0.563268	0.140978	0.191727
September 2015	0.321448	0.262823	0.179221	0.410739	0.300305	0.631818	0.160405	0.184173
2015	0.357382	0.166203	0.176623	0.29746	0.215244	0.232678	0.19683	0.164748
November 2015	0.424234	0.305765	0.144156	0.1	0.209756	0.242629	0.199865	0.178777
Desember 2015	0.710028	0.167992	0.163636	0.295381	0.292073	0.22715	0.21231	0.257554
Januari 2016	0.603064	0.314712	0.21039	0.471016	0.239939	0.238206	0.152816	0.224101
Februari 2016	0.435933	0.180517	0.105195	0.399307	0.220732	0.22715	0.154637	0.145324
Maret 2016	0.436769	0.193042	0.131169	0.354619	0.281098	0.238206	0.169815	0.163669
April 2016	0.298886	0.264612	0.138961	0.318245	0.355183	0.228256	0.144924	0.118345
Mei 2016	0.343175	0.235984	0.124675	0.394111	0.314024	0.230467	0.217774	0.177698
Juni 2016	0.481058	0.332604	0.125974	0.327598	0.272866	0.245946	0.230826	0.229496
Juli 2016	0.54624	0.214513	0.196104	0.452309	0.231707	0.242629	0.218988	0.245683
Agustus 2016	0.281337	0.159046	0.248052	0.306813	0.220732	0.523464	0.155852	0.158273
September 2016	0.36156	0.178728	0.133766	0.291224	0.207012	0.598649	0.177403	0.179856
Oktober 2016	0.302228	0.180517	0.138961	0.229908	0.196037	0.233784	0.19683	0.139928
November 2016	0.51532	0.166203	0.202597	0.287067	0.220732	0.22715	0.199562	0.210072
Desember 2016	0.404178	0.167992	0.145455	0.202887	0.234451	0.222727	0.286678	0.224101
Januari 2017	0.583844	0.327237	0.350649	0.363972	0.459451	0.230467	0.25086	0.317986
Februari 2017	0.448468	0.22167	0.3	0.475173	0.319512	0.249263	0.1343	0.18741
Maret 2017	0.263788	0.160835	0.198701	0.293303	0.292073	0.27027	0.200169	0.151799
April 2017	0.314763	0.164414	0.314286	0.372286	0.278354	0.211671	0.21656	0.199281

Mei 2017	0.474373	0.237773	0.185714	0.314088	0.209756	0.348771	0.248735	0.25
Juni 2017	0.326462	0.146521	0.262338	0.335912	0.209756	0.32113	0.223238	0.201439
Juli 2017	0.369916	0.284294	0.137662	0.312009	0.325	0.271376	0.219899	0.193885
Agustus 2017	0.242897	0.128628	0.194805	0.386836	0.1	0.22715	0.170422	0.1
September 2017	0.344847	0.175149	0.154545	0.35254	0.242683	0.237101	0.203204	0.166906
Oktober 2017	0.389136	0.151889	0.172727	0.284988	0.228963	0.224939	0.189848	0.163669
November 2017	0.449304	0.151889	0.172727	0.306813	0.552744	0.232678	0.194705	0.190647
Desember 2017	0.580501	0.212724	0.125974	0.295381	0.190549	0.22715	0.202597	0.214388
Januari 2018	0.604735	0.235984	0.215584	0.305774	0.217988	0.304545	0.196223	0.235971
Februari 2018	0.373259	0.162624	0.141558	0.29746	0.635061	0.275799	0.157066	0.179856
Maret 2018	0.364067	0.164414	0.151948	0.298499	0.22622	0.266953	0.212614	0.17446
April 2018	0.415042	0.225249	0.184416	0.298499	0.228963	0.259214	0.186509	0.176619
Mei 2018	0.25376	0.166203	0.106494	0.29642	0.404573	0.260319	0.264216	0.192806
Juni 2018	0.388301	0.176938	0.176623	0.301617	0.190549	0.209459	0.227487	0.188489
Juli 2018	0.470195	0.157256	0.244156	0.29642	0.217988	0.257002	0.171939	0.184173
Agustus 2018	0.264624	0.159046	0.158442	0.649769	0.377134	0.203931	0.164958	0.166906
September 2018	0.285515	0.157256	0.131169	0.29746	0.231707	0.264742	0.173457	0.128058
Oktober 2018	0.430919	0.155467	0.175325	0.29642	0.303049	0.275799	0.200169	0.193885
November 2018	0.423398	0.142942	0.153247	0.29642	0.204268	0.24484	0.23629	0.201439
Desember 2018	0.507799	0.176938	0.181818	0.322402	0.217988	0.268059	0.235987	0.231655
Januari 2019	0.478552	0.203777	0.132468	0.385797	0.239939	0.260319	0.230219	0.218705
Februari 2019	0.252089	0.191252	0.168831	0.333834	0.157622	0.22715	0.160101	0.116187
Maret 2019	0.401671	0.171571	0.137662	0.302656	0.2125	0.22715	0.246914	0.20036
April 2019	0.44429	0.155467	0.144156	0.287067	0.19878	0.221622	0.2457	0.208993
Mei 2019	0.486072	0.184095	0.15974	0.39515	0.168598	0.226044	0.198044	0.199281
Juni 2019	0.34234	0.164414	0.132468	0.342148	0.179573	0.235995	0.180742	0.143165
Juli 2019	0.551253	0.185885	0.144156	0.357737	0.187805	0.228256	0.158583	0.183094
Agustus 2019	0.405014	0.160835	0.18961	0.431524	0.196037	0.260319	0.186206	0.182014
September 2019	0.320613	0.302187	0.151948	0.365012	0.187805	0.22715	0.184992	0.158273
Oktober 2019	0.308078	0.169781	0.148052	0.249654	0.38811	0.22715	0.211096	0.157194
November 2019	0.399164	0.169781	0.167532	0.273557	0.2125	0.24484	0.183474	0.16259
Desember 2019	0.467688	0.339761	0.187013	0.259007	0.250915	0.232678	0.204418	0.215468

### Lampiran 3 Script python

```
# Import library
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import tensorflow as tf
from keras.models import Sequential
from keras.layers import Dense
from sklearn.metrics import mean_absolute_percentage_error
```

```

# import data
data = pd.read_csv('DataInf.csv')
dat = data.dropna(how='all')
df = dat.T
df.columns = df.iloc[0]
df = df.drop('Kelompok Pengeluaran Inflasi (2012=100)')

# normalisasi data
normalized_df = 0.1 + ((df - df.min()) * 0.9 / (df.max() - df.min()))

# membagi data
normalized_df.loc['Januari_05':'Desember_18', 'Bahan
Makanan': 'Transpor, Komunikasi, dan Jasa Keuangan']
dttrain = normalized_df.loc['Januari_05':'Desember_18']
dttest = normalized_df.loc['Januari_19':'Desember_19']
dttrain.reset_index(drop=True, inplace=True)
dttest.reset_index(drop=True, inplace=True)
xtrain = dttrain.loc[:, 'Bahan Makanan': 'Transpor, Komunikasi, dan
Jasa Keuangan']
ytrain = dttrain['Umum']
xtest = dttest.loc[:, 'Bahan Makanan': 'Transpor, Komunikasi, dan Jasa
Keuangan']
ytest = dttest['Umum']
xtrains = xtrain.astype(float)
ytrains = ytrain.astype(float)
xtests = xtest.astype(float)
ytests = ytest.astype(float)

# Membuat model arsitektur
model1 = Sequential() #menambah layer input dan layer tersembunyi
model1.add(Dense(4, activation='relu',
input_shape=(xtrains.shape[1],))) #model dengan 4 neuron di hidden
layer
model1.add(Dense(1)) #menambahkan layer output

# Kompilasi model
model1.compile(optimizer='adam', loss='mse')

# Melatih model dengan data latih
model1.fit(xtrains, ytrains, epochs=1000, batch_size=8)

# Mendapatkan bobot dari model
model1.summary()
model1_weights = model1.get_weights()

# Menampilkan bobot dari setiap layer
for layer_num, layer_weights in enumerate(model1_weights):

```



```

    print(f"\nLayer {layer_num + 1} Weights:")
    print(layer_weights)

# Prediksi dengan data uji
y_pred1 = model1.predict(xtests)

# Evaluasi model menggunakan metrik MAPE
mape1 = mean_absolute_percentage_error(ytests, y_pred1)
akurasi1 = 100 - mape1 * 100
print('mape:', mape1)
print('akurasi:', akurasi1)

# Denormalisasi data
d = df['Umum']
d = d.astype(float)
dd = d.loc['Januari_19':'Desember_19']
denormalized1 = ((y_pred - 0.1) / 0.9 * (d.max() - d.min())) +
d.min()
denormalized1

# Akurasi dari hasil denormalisasi
error_relatif = np.abs(denormalized1 - dd) / np.abs(dd)*100
mapedenorm = mean_absolute_percentage_error(dd, denormalized1)
akurasidenorm = 100 - mapedenorm * 100

# Membuat grafik perbandingan hasil prediksi dengan data asli
x = ['Januari', 'Februari', 'Maret', 'April', 'Mei', 'Juni', 'Juli',
'Agustus', 'September', 'Oktober', 'November', 'Desember']
plt.figure(figsize=(7, 7))
plt.plot(x, ytests, label='Actual', marker='o')
plt.plot(x, y_pred1, label='Predicted', marker='x')
plt.title('Plot Prediksi dengan Data Aktual')
plt.xlabel('bulan')
plt.ylabel('nilai inflasi')
plt.legend()
plt.grid(True)
plt.show
plt.savefig('newnorm_hasilfix_741_relu_b8_e1k.png')

```

## RIWAYAT HIDUP



Aisyah Sukmaindah Trisianti lahir di Lamongan pada tanggal 29 Juli 2002. Perempuan yang biasa dipanggil Aisyah ini beralamat di Dusun Srampat RT 10/RW 02 Desa Klagensrampat, Kecamatan Maduran, Kabupaten Lamongan. Merupakan anak tunggal dari pasangan Bapak Subakri dan Ibu Sutri.

Penulis menempuh pendidikan dasar di SDN Klagensrampat dan lulus pada tahun 2014. Kemudian melanjutkan pendidikan menengah pertama di SMP Negeri 1 Maduran dan lulus pada tahun 2017. Setelah itu penulis melanjutkan pendidikan menengah atas di SMA Negeri 2 Lamongan dan lulus pada tahun 2020. Kemudian melanjutkan pendidikan perguruan tinggi Strata 1 di Universitas Islam Negeri Maulana Malik Ibrahim Malang mengambil jurusan matematika. Selama berkuliah penulis aktif dalam komunitas Matematika English Club.



**KEMENTERIAN AGAMA RI  
UNIVERSITAS ISLAM NEGERI  
MAULANA MALIK IBRAHIM MALANG  
FAKULTAS SAINS DAN TEKNOLOGI**

Jl. Gajayana No.50 Dinoyo Malang Telp. / Fax. (0341)558933

**BUKTI KONSULTASI SKRIPSI**

Nama : Aisyah Sukmaindah Trisianti  
NIM : 200601110081  
Fakultas / Jurusan : Sains dan Teknologi / Matematika  
Judul Skripsi : Implementasi Algoritma *Backpropagation* untuk  
Memprediksi Tingkat Inflasi di Kota Malang  
Pembimbing I : Hisyam Fahmi, M.Kom.  
Pembimbing II : Erna Herawati, M.Pd.

No	Tanggal	Hal	Tanda Tangan
1.	25 Juli 2023	Konsultasi Topik dan Data	1.
2.	6 November 2023	Konsultasi Bab I, II, dan III	2.
3.	10 November 2023	Konsultasi Bab I, II, dan III	3.
4.	20 November 2023	Konsultasi Bab I, II, dan III	4.
5.	28 November 2023	Konsultasi Bab I, II, dan III	5.
6.	30 November 2023	Konsultasi Kajian Agama Bab I dan II	6.
7.	5 Desember 2023	ACC Bab I, II, dan III	7.
8.	5 Desember 2023	ACC Kajian Agama Bab I dan II	8.
9.	5 Desember 2023	ACC Seminar Proposal	9.
10.	25 Februari 2024	Konsultasi Bab IV dan V	10.
11.	26 Februari 2024	Konsultasi Bab IV dan V	11.
12.	1 Maret 2024	Konsultasi Bab IV dan V	12.
13.	18 Maret 2024	Konsultasi Bab IV dan V	13.
14.	18 Maret 2024	Konsultasi Kajian Agama Bab IV	14.
15.	25 Maret 2024	ACC Bab IV dan V	15.



**KEMENTERIAN AGAMA RI  
UNIVERSITAS ISLAM NEGERI  
MAULANA MALIK IBRAHIM MALANG  
FAKULTAS SAINS DAN TEKNOLOGI**

Jl. Gajayana No.50 Dinoyo Malang Telp. / Fax. (0341)558933

16.	26 Maret 2024	ACC Kajian Agama Bab IV	16.
17.	27 Maret 2024	Konsultasi Revisi Seminar Proposal	17.
18.	25 April 2024	ACC Seminar Hasil	18.
19.	30 Mei 2024	Konsultasi Revisi Seminar Hasil	19.
20.	6 Juni 2024	ACC Sidang Skripsi	20.
21.	14 Juni 2024	ACC Akhir Keseluruhan	21.

Malang, 14 Juni 2024

Mengetahui,

Ketua Program Studi Matematika



**D. Susanti, M.Sc.**

NIP. 19741129 200012 2 005