

**ANALISIS PREDIKSI JUMLAH PENERIMA BANTUAN
SOSIAL MENGGUNAKAN JARINGAN SYARAF TIRUAN
DENGAN METODE *BACKPROPAGATION* DAN *EXTREME
LEARNING MACHINE***

SKRIPSI

Diajukan untuk Memenuhi Sebagian Syarat Guna Memperoleh
Gelar Sarjana Matematika
dalam Ilmu Matematika



Oleh : **KHOIRUL ANWAR**
NIM : 2008046034

PROGRAM STUDI MATEMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI WALISONGO
SEMARANG
2024

PERNYATAAN KEASLIAN

Yang bertanda tangan di bawah ini :

Nama : Khoirul Anwar
NIM : 2008046034
Jurusan/Program Studi : Matematika/ Matematika

menyatakan bahwa skripsi yang berjudul :

**ANALISIS PREDIKSI JUMLAH PENERIMA BANTUAN SOSIAL
MENGUNAKAN JARINGAN SYARAF TIRUAN DENGAN METODE
BACKPROPAGATION DAN EXTREME LEARNING MACHINE**

secara keseluruhan adalah hasil penelitian/karya saya sendiri,
kecuali bagian tertentu yang dirujuk sumbernya.

Semarang, 20 Desember 2024
Pembuat pernyataan,



Khoirul Anwar
NIM : 2008046034



KEMENTERIAN AGAMA R.I.
UNIVERSITAS ISLAM NEGERI WALISONGO
FAKULTAS SAINS DAN TEKNOLOGI
Jl. Prof. Dr. Hamka (Kampus II) Ngaliyan Semarang
Telp. 024-7601295 Fax. 7615387

PENGESAHAN

Naskah skripsi berikut ini :

Judul : **ANALISIS PREDIKSI JUMLAH PENERIMA
BANTUAN SOSIAL MENGGUNAKAN JARINGAN
SYARAF TIRUAN DENGAN METODE
BACKPROPAGATION DAN EXTREME LEARNING
MACHINE**

Penulis : Khoirul Anwar

NIM : 2008046034

Jurusan : Matematika

Telah diujikan dalam sidang *tugas akhir* oleh Dewan Penguji Fakultas Sains dan Teknologi UIN Walisongo dan dapat diterima sebagai salah satu syarat memperoleh gelar sarjana dalam Ilmu Matematika.

Semarang, 27 Desember 2024

DEWAN PENGUJI

Penguji I,

Agus Wayan Yulianto, M.Sc.

NIP : 198907162019031007

Penguji II,

Herma Diah Miasary, M.Sc.

NIP : 198709212019032010

Penguji III,

Siti Mashhah, M.Si

NIP : 197706112011012004

Penguji IV,

Aini Fitriyah, M.Sc.

NIP : 198909292019032021

Pembimbing,

Agus Wayan Yulianto, M.Sc.

NIP : 198907162019031007

NOTA DINAS

Semarang, 20 Desember 2024

Yth. Ketua Program Studi Matematika
Fakultas Sains dan Teknologi
UIN Walisongo Semarang

Assalamu'alaikum warahmatullahi wabarakatuh

Dengan ini diberitahukan bahwa saya telah melakukan bimbingan, arahan dan koreksi naskah skripsi dengan:

Judul : ANALISIS PREDIKSI JUMLAH PENERIMA BANTUAN
SOSIAL MENGGUNAKAN JARINGAN SYARAF
TIRUAN DENGAN METODE BACKPROPAGATION
DAN EXTREME LEARNING MACHINE

Nama : Khoirul Anwar

NIM : 2008046034

Jurusan : Matematika

Saya memandang bahwa naskah skripsi tersebut sudah dapat diajukan kepada Fakultas Sains dan Teknologi UIN Walisongo untuk diujikan dalam Sidang Munaqasyah.

Wassalamu'alaikum warahmatullahi wabarakatuh

Pembimbing,



Agus Wayan Yulianto, M.Sc.

NIP : 198907162019031007

ABSTRAK

Kemiskinan merupakan suatu keadaan dimana seseorang tidak mampu mencukupi kebutuhan dasar seperti makanan, pakaian dan tempat tinggal. Sebagai salah satu kota metropolitan, Kota Semarang masuk kedalam 6 kota di Jawa Tengah yang memiliki jumlah penduduk miskin tertinggi sebesar 80.530 jiwa pada tahun 2023. Dalam upaya untuk mengatasi kemiskinan yang mengalami fluktuasi setiap tahunnya, pemerintah melalui Kementerian Sosial memberikan berbagai jenis bantuan sosial. Pemberian bantuan sosial perlu dipersiapkan pemerintah guna menentukan anggaran dana bantuan sosial yang harus dikeluarkan setiap tahunnya. Salah satu caranya yaitu dengan melakukan prediksi pada jumlah penerima bantuan sosial. Penelitian ini menggunakan metode *Backpropagation* dan *Extreme Learning Machine*. Data yang digunakan adalah data jumlah penerima BPNT Kota Semarang Bulan Januari 2022 - Desember 2023 yang bersumber dari Dinas Sosial Kota Semarang. Berdasarkan penelitian yang telah dilakukan, perbandingan metode *Backpropagation* dan *Extreme Learning Machine* jika dibandingkan dengan nilai MSE pada metode *Backpropagation* diperoleh nilai 0,00000094,, sedangkan pada metode ELM diperoleh nilai 0,00005277. Dimana nilai ini menunjukkan bahwa metode *Backpropagation* tingkat kesalahannya lebih kecil dibandingkan metode ELM. Nilai RMSE dan MAPE pada metode *Backpropagation* juga memiliki nilai yang lebih kecil dibandingkan metode ELM. Jadi metode terbaik untuk penelitian ini adalah dengan metode *Backpropagation*.

Kata kunci : *Kemiskinan, Bantuan Pangan Non Tunai, Prediksi, Jaringan Syaraf Tiruan, Backpropagation, Extreme Learning Machine*

KATA PENGANTAR

Assalamu'alaikum Wr.Wb.

Puji syukur atas segala rahmat, karunia dan kemudahan yang diberikan Allah SWT. sehingga penulis dapat menyelesaikan penyusunan skripsi ini dengan baik dan lancar. Shalawat serta salam semoga senantiasa tercurah kepada Rasulullah SAW. yang selalu dinantikan syafaatnya kelak di akhirat.

Penyusunan skripsi ini bertujuan guna memenuhi tugas dan persyaratan untuk memperoleh gelar sarjana dalam Ilmu Matematika Fakultas Sains dan Teknologi UIN Walisongo Semarang. Penyusunan skripsi ini tidak lepas dari bantuan, dukungan, motivasi, dan doa dari berbagai pihak. Pada kesempatan ini penulis mengucapkan terima kasih kepada:

1. Bapak Prof. Dr. Nizar, M.Ag. selaku Rektor Universitas Islam Negeri Walisongo Semarang.
2. Bapak Prof Dr. H. Musahadi, M. Ag. Dekan Fakultas Sains dan Teknologi UIN Walisongo Semarang.
3. Ibu Any Muanalifah, M. Si, Ph. D selaku ketua Program Studi Matematika Fakultas Sains dan Teknologi UIN Walisongo Semarang.
4. Bapak Agus Wayan Yulianto, M. Sc., selaku Dosen Pembimbing yang telah memberikan bimbingan, arahan, selama proses penulisan hingga terselesaikannya skripsi ini dengan baik.
5. Ibu Zulaikha, M. Si., selaku Dosen Wali yang telah memberikan arahan, bimbingan, dan semangat selama

proses perkuliahan.

6. Segenap Ibu dan Bapak dosen Fakultas Sains dan Teknologi UIN Walisongo Semarang yang telah memberikan bekal pengetahuan selama proses perkuliahan.
7. Ibu Siti Khotimah dan Bapak Sukeri yang telah memberikan doa, semangat, motivasi, dukungan, kesabaran, dan kebaikan lainnya sehingga penulis dapat menyelesaikan penyusunan skripsi.
8. Segenap teman-teman Matematika angkatan 2020 yang telah berjuang, belajar, saling support dan berbagi rasa yang sama selama di bangku perkuliahan.
9. Kepada semua pihak yang telah memberikan dukungan yang tidak dapat penulis sebutkan satu per satu.

Penulis tidak dapat memberikan balasan apapun selain ucapan terima kasih dan doa, semoga Allah senantiasa membalas semua kebaikan mereka dengan sebaik-baiknya balasan. Penulis menyadari bahwa skripsi ini masih jauh dari kata sempurna. Oleh karena itu, saran perbaikan yang membangun sangat di harapkan demi kesempurnaan skripsi ini dan semoga dapat bermanfaat bagi semua pihak.

DAFTAR ISI

HALAMAN JUDUL	i
PERNYATAAN KEASLIAN	ii
PENGESAHAN	iv
NOTA PEMBIMBING I	iv
KATA PENGANTAR	vii
DAFTAR ISI	viii
DAFTAR TABEL	x
DAFTAR GAMBAR	xiii
DAFTAR LAMPIRAN	xv
BAB I PENDAHULUAN	1
A. Latar Belakang Masalah	1
B. Rumusan Masalah	5
C. Tujuan Penelitian	5
D. Manfaat Penelitian	6
E. Batasan Penelitian	6
BAB II LANDASAN PUSTAKA	7
A. Kajian Teori	7
1. Bantuan Sosial	7
2. Bantuan Pangan Non Tunai (BPNT)	7
3. Peramalan (<i>Forecasting</i>)	8
4. Jaringan Syaraf Tiruan	10
5. <i>Backpropagation</i>	15
6. <i>Extreme Learning Machine</i> (ELM)	21
B. Kajian Pustaka	28
BAB III Metode Penelitian	31
A. Jenis Penelitian	31
B. Variabel Penelitian	31
C. Sumber Data	31
D. Metode Pengumpulan Data	32
E. Metode Analisis Data	32
BAB IV HASIL DAN PEMBAHASAN	37

A.	Analisis Deskriptif	37
B.	Metode <i>Backpropagation</i>	39
C.	Metode <i>Extreme Learning Machine</i>	61
D.	Perbandingan Metode <i>Backpropagation</i> dan Metode <i>Extreme Learning Machine</i>	80
BAB V PENUTUP		82
A.	Kesimpulan	82
B.	Saran	83
DAFTAR PUSTAKA		84
Lampiran-lampiran		91

DAFTAR TABEL

Tabel	Judul	Halaman
Tabel 4.1	Data jumlah penerima BPNT Kota Semarang	37
Tabel 4.2	Data Normalisasi jumlah penerima BPNT Kota Semarang	40
Tabel 4.3	Pola Input dan Output	41
Tabel 4.4	Pembagian Data	42
Tabel 4.5	Perbandingan nilai data metode <i>Backpropagation</i>	46
Tabel 4.6	Hasil Prediksi metode <i>Backpropagation</i> pola 12-2-1	46
Tabel 4.7	Nilai <i>error</i> metode <i>Backpropagation</i> 12-2-1	48
Tabel 4.8	Perbandingan nilai data metode <i>Backpropagation</i>	50
Tabel 4.9	Hasil Prediksi metode <i>Backpropagation</i> pola 12-9-1	50
Tabel 4.10	Nilai <i>error</i> metode <i>Backpropagation</i> 12-9-1	52
Tabel 4.11	Perbandingan nilai data metode <i>Backpropagation</i>	54
Tabel 4.12	Hasil Prediksi metode <i>Backpropagation</i> pola 12-16-1	54
Tabel 4.13	Nilai <i>error</i> metode <i>Backpropagation</i> 12-16-1	56
Tabel 4.14	Perbandingan nilai data metode <i>Backpropagation</i>	58
Tabel 4.15	Hasil Prediksi metode <i>Backpropagation</i> pola 12-23-1	58

Tabel 4.16	Nilai <i>error</i> metode <i>Backpropagation</i> 12-23-1	60
Tabel 4.17	Model Arsitektur Jaringan	60
Tabel 4.18	Data Normalisasi jumlah penerima BPNT Kota Semarang	62
Tabel 4.19	Pembagian Data	63
Tabel 4.20	Perbandingan nilai data metode <i>Extreme Learning Machine</i>	65
Tabel 4.21	Hasil Prediksi metode <i>Extreme Learning Machine</i> pola 12-2-1	65
Tabel 4.22	Nilai <i>error</i> metode <i>Extreme Learning Machine</i> 12-2-1	67
Tabel 4.23	Perbandingan nilai data metode <i>Extreme Learning Machine</i>	69
Tabel 4.24	Hasil Prediksi metode <i>Extreme Learning Machine</i> pola 12-9-1	69
Tabel 4.25	Nilai <i>error</i> metode <i>Extreme Learning Machine</i> 12-9-1	71
Tabel 4.26	Perbandingan nilai data metode <i>Extreme Learning Machine</i>	73
Tabel 4.27	Hasil Prediksi metode <i>Extreme Learning Machine</i> pola 12-16-1	73
Tabel 4.28	Nilai <i>error</i> metode <i>Extreme Learning Machine</i> 12-16-1	75
Tabel 4.29	Perbandingan nilai data metode <i>Extreme Learning Machine</i>	77
Tabel 4.30	Hasil Prediksi metode <i>Extreme Learning Machine</i> pola 12-23-1	77

Tabel 4.31	Nilai <i>error</i> metode <i>Extreme Learning Machine</i> 12-23-1	79
Tabel 4.32	Model Arsitektur Jaringan	79
Tabel 4.33	Perbandingan Nilai Akurasi Metode <i>Backpropagation</i> dan Metode ELM	81

DAFTAR GAMBAR

Gambar	Judul	Halaman
Gambar 1.1	Grafik Jumlah Penduduk Miskin di Kota Semarang (BPS, 2023)	2
Gambar 2.1	Fungsi <i>sigmoid biner</i>	14
Gambar 2.2	Fungsi <i>sigmoid bipolar</i>	15
Gambar 3.1	<i>flowchart</i> metode <i>Bacpropagation</i>	34
Gambar 3.2	<i>Flowchart</i> metode ELM	36
Gambar 4.1	Grafik Data Penerima Bantuan BPNT	38
Gambar 4.2	<i>Backpropagation</i> 12-2-1	44
Gambar 4.3	Perbandingan data asli dan data prediksi	45
Gambar 4.4	Hasil Prediksi jumlah penerima BPNT metode <i>Backpropagation</i>	47
Gambar 4.5	<i>Backpropagation</i> 12-9-1	48
Gambar 4.6	Perbandingan data asli dan data prediksi	49
Gambar 4.7	Hasil Prediksi jumlah penerima BPNT metode <i>Backpropagation</i>	51
Gambar 4.8	<i>Backpropagation</i> 12-16-1	52
Gambar 4.9	Perbandingan data asli dan data prediksi	53
Gambar 4.10	Hasil Prediksi jumlah penerima BPNT metode <i>Backpropagation</i>	55
Gambar 4.11	<i>Backpropagation</i> 12-23-1	56
Gambar 4.12	Perbandingan data asli dan data prediksi	57
Gambar 4.13	Hasil Prediksi jumlah penerima BPNT metode <i>Backpropagation</i>	59

Gambar 4.14	<i>Extreme Learning Machine</i> 12-2-1	63
Gambar 4.15	Perbandingan data asli dan data prediksi	64
Gambar 4.16	Hasil Prediksi jumlah penerima BPNT metode <i>Extreme Learning Machine</i>	66
Gambar 4.17	<i>Extreme Learning Machine</i> 12-9-1	67
Gambar 4.18	Perbandingan data asli dan data prediksi	68
Gambar 4.19	Hasil Prediksi jumlah penerima BPNT metode <i>Extreme Learning Machine</i>	70
Gambar 4.20	<i>Extreme Learning Machine</i> 12-16-1	71
Gambar 4.21	Perbandingan data asli dan data prediksi	72
Gambar 4.22	Hasil Prediksi jumlah penerima BPNT metode <i>Extreme Learning Machine</i>	74
Gambar 4.23	<i>Extreme Learning Machine</i> 12-23-1	75
Gambar 4.24	Perbandingan data asli dan data prediksi	76
Gambar 4.25	Hasil Prediksi jumlah penerima BPNT metode <i>Extreme Learning Machine</i>	78
Gambar 4.26	Perbandingan Nilai Data Aktual, Nilai <i>Backpropagation</i> , Nilai ELM	80

DAFTAR LAMPIRAN

	Halaman	
Lampiran 1	Normalisasi Data	91
Lampiran 2	Langkah-langkah Backpropagation di R Studio	93
Lampiran 3	Langkah-langkah Extreme Learning Machine di R Studio	98
Lampiran 4	Perhitungan Manual <i>Backpropagation</i>	103
Lampiran 5	Simulasi Metode <i>Extreme Learning Machine</i>	115
Lampiran 6	Daftar Riwayat Hidup	119

BAB I

PENDAHULUAN

A. Latar Belakang Masalah

Kemiskinan ialah keadaan di mana individu tidak memiliki kecukupan untuk memberi pemenuhan pada kebutuhan primer seperti konsumsi pangan, sandang, serta papan (Dasangga & Cahyono, 2020). Kemiskinan masih menjadi masalah yang cukup serius yang dihadapi masyarakat Indonesia setiap tahunnya. Kemiskinan juga dapat merujuk pada minimnya akses terhadap pendidikan serta pekerjaan yang berkualitas (Purba et al., 2020). Kenaikan jumlah penduduk miskin dapat menurunkan kualitas hidup masyarakat, yang pada gilirannya menghambat pertumbuhan ekonomi.

Dalam upaya mengatasi persoalan kemiskinan, pemerintah telah mengimplementasikan berbagai kebijakan dan program. Salah satunya, melalui Kementerian Sosial, pemerintah melaksanakan berbagai jenis bantuan sosial sebagai langkah mitigasi untuk menurunkan tingkat kemiskinan di Indonesia. Salah satu jenis bantuan sosial yang telah diberikan pemerintah adalah Bantuan Pangan Non Tunai (BPNT).

Bantuan Pangan Non Tunai ialah satu dari bentuk bantuan sosial yang diberikan oleh pemerintah kepada Keluarga Penerima Manfaat (KPM). Bantuan ini disalurkan melalui dana elektronik yang bisa dimanfaatkan oleh KPM untuk membeli kebutuhan pangan di e-warong yang sudah ditentukan (Permensos 5/2021). Penerapan BPNT diharapkan dapat mengurangi tekanan finansial yang dihadapi oleh Keluarga Penerima Manfaat dengan membantu

mencukupi sebagian kebutuhan pangan mereka, serta mendukung perbaikan status gizi bagi KPM. Selain itu, program ini juga berpotensi mendorong pertumbuhan ekonomi lokal yang bergerak dalam sektor penjualan bahan pangan seperti telur dan biji-bijian.

Berdasarkan data jumlah penduduk miskin Provinsi Jawa Tengah yang mencakup 29 kabupaten dan 6 kota, Kota Semarang masuk kedalam 6 kota yang jumlah penduduk miskinnya tertinggi di Jawa Tengah. Pada tahun 2022 Kota Semarang mempunyai jumlah penduduk miskin sebesar 79.870 jiwa dan pada tahun 2023 meningkat dengan besaran 660 jiwa dengan jumlah penduduk miskin sejumlah 80.530 jiwa (BPS, 2023).



Gambar 1.1. Grafik Jumlah Penduduk Miskin di Kota Semarang (BPS, 2023)

Jumlah penduduk miskin di Kota Semarang digambarkan dalam grafik pada Gambar 1.1. Tahun 2011 memiliki jumlah penduduk miskin terbanyak, yaitu 88.500 jiwa, sedangkan tahun 2019 memiliki jumlah penduduk miskin paling sedikit, yakni

71.970 jiwa. Grafik tersebut memperlihatkan bahwasanya jumlah penduduk miskin di Kota Semarang mengalami fluktuasi. Hal tersebut dapat berpengaruh terhadap jumlah bantuan sosial yang akan diberikan. Semakin banyak jumlah penduduk miskin maka semakin banyak juga jumlah bantuan sosial yang akan diberikan. Untuk mengantisipasi jumlah bantuan sosial yang berbeda setiap bulannya, perlu dilakukan suatu tindakan memperkirakan jumlah bantuan sosial guna membantu pemerintah dalam menentukan anggaran dana bantuan sosial dan menganalisis efektif atau tidaknya bantuan sosial dalam menurunkan kemiskinan di Kota Semarang.

Prediksi merujuk pada upaya analitis untuk mengestimasi kemungkinan hasil yang akan terjadi di waktu mendatang dengan mengandalkan data historis serta informasi terkini, dengan tujuan meminimalkan kesalahan atau selisih antara kenyataan yang terjadi dan hasil yang telah diperkirakan (Adiguno et al., 2022). Jaringan Syaraf Tiruan ialah satu diantara teknik yang seringkali dimanfaatkan pada peramalan dan prediksi. Jaringan syaraf tiruan bertindak sebagai pengganti saraf dan otak manusia, yang memungkinkannya untuk belajar dengan cepat dan mudah saat mengidentifikasi pola. Jaringan saraf ini dioperasikan oleh perangkat lunak yang memiliki kemampuan untuk melaksanakan berbagai rangkaian prosedur perhitungan kompleks (Supriyanto et al., 2022).

Salah satu algoritma pada jaringan syaraf tiruan ialah *Backpropagation*. Metode *Backpropagation* adalah teknik matematika yang sering dimanfaatkan untuk peramalan dan prediksi. *Backpropagation* melatih jaringan untuk memanfaatkan penggunaan berbagai pola, sehingga menghasilkan output yang

maksimal. Algoritma ini melibatkan penggunaan pola penyesuaian bobot guna memenuhi capaian tingkat *error* yang rendah antara *output* yang dihasilkan dan *output* aktual (Junaidi et al., 2022).

Kemudian algoritma lainnya dalam jaringan syaraf tiruan yakni *Extreme Learning Machine* (ELM). Metode ELM ialah jenis jaringan saraf tiruan *Single Hidden Layer Feedforward Neural Network* (SLFN) karena hanya mempunyai satu lapisan tersembunyi (Hemeida et al., 2020). Jika dibanding metode jaringan syaraf tiruan lain seperti *Backpropagation*, metode ini memiliki keunggulan dalam hal *learning speed*. *Learning speed* mengacu pada kecepatan pelatihan jaringan syaraf tiruan. (Rumandan et al., 2022). Hal ini karena setiap parameter, termasuk bobot *input* dan bias tersembunyi, dipilih secara acak, sehingga menghasilkan laju pembelajaran yang signifikan lebih cepat dan kemampuan generalisasi yang lebih baik (Fikriya et al., 2017).

Penelitian sebelumnya berjudul Perbandingan Algoritma ELM Dan Backpropagation Terhadap Prestasi Akademik Mahasiswa yang diteliti oleh Heny Pratiwi & Kusno Harianto. Penelitian ini bertujuan untuk mengetahui algoritma mana yang paling sesuai dalam mengetahui prestasi akademik mahasiswa. Hasil penelitian ini menyatakan bahwa algoritma Extreme Learning Machine memiliki 14,84% tingkat kesalahan lebih rendah dibandingkan dengan Backpropagasi 28,20%.

Sesuai uraian tersebut, peneliti sangat memiliki ketertarikan untuk membahas bagaimana metode *Backpropagation* dan *Extreme Learning Machine* bisa dipergunakan untuk meramalkan jumlah penerima bantuan sosial BPNT. Untuk itu, peneliti mengambil judul "**Analisis Prediksi Jumlah Penerima Bantuan Sosial Menggunakan Jaringan Syaraf Tiruan Dengan Metode**

Backpropagation dan Extreme Learning Machine". Prediksi ini dipergunakan untuk memperkirakan jumlah penerima BPNT di Kota Semarang. Prediksi ini dimaksudkan untuk membantu pemerintah dalam menentukan anggaran bantuan sosial dan menganalisis efektif atau tidaknya bantuan sosial dalam menurunkan angka kemiskinan di Kota Semarang.

B. Rumusan Masalah

Dari latar belakang yang telah dijelaskan, dapat dirumuskan beberapa permasalahan, yaitu:

1. Bagaimana hasil prediksi jumlah penerima BPNT dengan metode *Backpropagation* dan *Extreme Learning Machine*?
2. Bagaimana perbandingan hasil prediksi jumlah penerima BPNT dengan metode *Backpropagation* dan *Extreme Learning Machine*?

C. Tujuan Penelitian

Tujuan dari penelitian ini sebagai berikut:

1. Mengetahui hasil prediksi jumlah penerima BPNT dari penerapan metode *Backpropagation* dan metode *Extreme Learning Machine*.
2. Mengetahui hasil perbandingan metode *Backpropagation* dan metode *Extreme Learning Machine* .

D. Manfaat Penelitian

Manfaat penelitian ini adalah sebagai berikut:

1. Bagi penulis dan pembaca yaitu untuk meningkatkan pemahaman terkait prediksi dengan memanfaatkan penggunaan metode *Backpropagation* dan *Extreme Learning Machine*.
2. Bagi pemerintah dapat menggunakannya sebagai panduan dan faktor untuk memperkirakan berapa banyak orang yang akan menerima BPNT di masa depan.

E. Batasan Penelitian

Batasan masalah pada penelitian ini adalah sebagai berikut:

1. Ruang lingkup pada penelitian ini yaitu jumlah penerima bantuan pangan non tunai di Kota Semarang.
2. Data yang digunakan dalam penelitian ini dimulai pada bulan Januari 2022 sampai Desember 2023.
3. Digunakan bantuan *software R Studio* dalam penelitian ini.

BAB II

LANDASAN PUSTAKA

A. Kajian Teori

1. Bantuan Sosial

Bantuan Sosial adalah bantuan berupa uang, barang, atau jasa kepada seseorang, keluarga, kelompok atau masyarakat miskin, tidak mampu dan/atau rentan terhadap resiko sosial (Permensos, 2019). Bantuan sosial ini diberikan untuk mengurangi beban masyarakat miskin dan sebagai sarana untuk mengatasi masalah kemiskinan di Indonesia. Bantuan sosial hanya diberikan secara sementara dengan tujuan agar mereka dapat melakukan perbaikan kehidupan yang wajar. Bantuan sosial merupakan realisasi pemerintah atau pemerintah daerah untuk memenuhi kewajibannya dalam memperhatikan masyarakat miskin tingkat paling bawah (Iswandy, 2015).

2. Bantuan Pangan Non Tunai (BPNT)

Bantuan Pangan Non Tunai ialah satu diantara bentuk bantuan sosial yang diberikan oleh pemerintah kepada Keluarga Penerima Manfaat melalui penggunaan mata uang elektronik. Dana tersebut dapat dimanfaatkan guna membeli bahan pokok yang sudah ditetapkan di e-warong, sebagaimana diatur dalam Permensos 5/2021. Sasaran dari program BPNT ialah keluarga-keluarga yang kemudian ditetapkan sebagai Keluarga Penerima Manfaat. Pada tahun 2017, KPM BPNT mengacu pada masyarakat yang tergolong dalam 25% lapisan sosial ekonomi terendah di wilayah

pelaksanaan, serta nama-nama mereka tercatat dalam Data Terpadu Kesejahteraan Sosial yang disahkan oleh pemerintah daerah.

Setelah adanya *covid-19* tahun 2019, KPM bisa mendapatkan bantuan BPNT sebesar Rp.200.000/ bulan (Utami et al., 2023). Bantuan disalurkan bukan dalam wujud uang tunai, melainkan dapat dipertukarkan dengan bahan pangan pokok melalui sistem e-warung. Apabila dana bantuan tersebut tidak dimanfaatkan pada bulan yang bersangkutan, saldo yang belum terpakai akan dipertahankan dan diakumulasikan dalam rekening penerima.

Tujuan dan manfaat dari bantuan ini antara lain untuk mengurangi tekanan ekonomi yang dihadapi keluarga kurang mampu dengan memenuhi kebutuhan pangan tertentu, memperbaiki keseimbangan gizi bagi KPM, mendorong pertumbuhan ekonomi di tingkat lokal, memperkuat ketahanan pangan pada level rumah tangga, serta berperan sebagai instrumen dalam upaya pengentasan kemiskinan (Kemensos).

3. Peramalan (*Forecasting*)

Peramalan merupakan proses untuk memperkirakan atau meramalkan kejadian yang akan terjadi di waktu mendatang berlandaskan data historis atau informasi yang telah ada sebelumnya (Rahayu & Bernadus, 2021). Proses ini menggabungkan unsur seni dan ilmu dalam merumuskan prediksi tentang peristiwa yang akan datang. Dalam peramalan, penggunaan model matematika menjadi penting untuk mengolah dan mengkalkulasi data historis, seperti angka penjualan di masa lalu, guna menghasilkan proyeksi yang relevan untuk masa depan. Peramalan bertujuan untuk mengurangi risiko perbedaan dan

kesalahan. Dengan hasil peramalan, diyakini bahwa kegiatan atau keputusan perusahaan atau organisasi akan berdampak lebih besar di waktu mendatang (Sitompul & Puspasari, 2023).

Dalam peramalan terdapat banyak metode yang dapat digunakan, namun tidak semua metode dapat sesuai dengan kasus yang ada. Penelitian ini menggunakan *Mean Square Error*, *Root Mean Square Error*, dan *Mean Absolute Percentage Error* untuk melihat seberapa besar tingkat kesalahan dalam peramalan (Maricar, 2019).

a. *Mean Square Error* (MSE)

Merupakan perhitungan yang digunakan untuk menghitung rata-rata kesalahan berpangkat (Maricar, 2019). Rumus berikut ini digunakan untuk menentukan MSE.

$$MSE = \frac{\sum_{t=1}^n (X_t - F_t)^2}{n} \quad (2.1)$$

Keterangan:

n : Jumlah data

X_t : Data sebenarnya

F_t : Data prediksi

b. *Root Mean Square Error* (RMSE)

RMSE adalah akar kuadrat dari MSE (Maricar, 2019). Rumus berikut ini digunakan untuk menentukan RMSE.

$$RMSE = \sqrt{\frac{\sum_{t=1}^n |X_t - F_t|^2}{n}} \quad (2.2)$$

Keterangan:

X_t : Data sebenarnya

F_t : Data prediksi

n : Jumlah data

c. *Mean Absolute Percentage Error* (MAPE)

Merupakan perhitungan yang digunakan untuk menghitung rata-rata persentase kesalahan mutlak (Maricar, 2019).

Rumus berikut ini digunakan untuk menentukan MAPE.

$$MAPE = \left(\frac{100\%}{n}\right) \sum_{t=1}^n \frac{|X_t - F_t|}{X_t} \quad (2.3)$$

Keterangan:

n : Jumlah data

X_t : Data sebenarnya

F_t : Data prediksi

4. Jaringan Syaraf Tiruan

a. Model JST

Jaringan syaraf tiruan ialah sebuah model buatan yang terinspirasi dari struktur dan fungsi otak manusia, dirancang untuk meniru cara otak belajar. Istilah "tiruan" digunakan karena JST dibangun dengan menggunakan perangkat lunak komputer yang mampu menjalankan serangkaian kalkulasi dalam proses pembelajaran (Kusumadewi, 2004).

Jaringan syaraf tiruan dibuat dengan penggunaan model matematika yang mengasumsikan:

- 1) Informasi diproses oleh elemen sederhana yang dikenal dengan *neuron*.
- 2) *Neuron* yang saling terhubung saling bertukar sinyal.
- 3) *Neuron* yang saling terhubung memiliki bobot yang menentukan seberapa kuat atau lemahnya sinyal yang akan dikirimkan.
- 4) Fungsi aktivasi yang digunakan setiap *neuron* untuk menentukan *output*nya sebanding dengan jumlah *input* yang diterimanya. Setelah itu, *output* akan dijalankan perbandingan dengan batas ambang yang telah ditentukan. (Muis, 2021).

b. Aplikasi Jaringan Syaraf Tiruan

Menurut Siang (2009), aplikasi jaringan syaraf tiruan dibedakan menjadi tiga yaitu :

1) Pengenalan Pola

Jaringan syaraf tiruan dapat digunakan untuk mengenali pola, seperti huruf, angka, suara, atau tanda tangan, meskipun telah mengalami sedikit perubahan. Kemampuan ini menyerupai otak manusia yang dapat mengenali seseorang meskipun terdapat perubahan pada wajah atau bentuk tubuh setelah beberapa waktu tidak bertemu.

2) Signal Processing

Jaringan saraf tiruan (model ADALINE) dapat dipakai untuk menekan noise dalam saluran telepon.

3) Peramalan

Jaringan syaraf tiruan juga dapat digunakan untuk memprediksi kejadian di masa depan berdasarkan pola yang telah terjadi di masa lalu. Hal ini dimungkinkan karena kemampuannya untuk menyimpan informasi dan membuat generalisasi dari data yang telah dipelajari sebelumnya.

c. Lapisan Penyusun JST

Lapisan dalam JST dibagi tiga lapisan, yaitu:

1) Lapisan *Input*

Dalam lapisan *input* terdapat unit *input*. Unit *input* ini mendapatkan pola *input* data dari luar yang menjelaskan suatu masalah.

2) Lapisan Tersembunyi

Unit tersembunyi adalah unit yang terdapat pada lapisan tersembunyi. pada lapisan ini, *output* belum dihasilkan.

3) Lapisan *Output*

Unit *output* adalah unit yang terdapat pada lapisan *output*. Solusi JST untuk suatu masalah adalah *output* dari lapisan ini (Nugroho, 2021).

d. Arsitektur JST

Terdapat 2 jenis model arsitektur JST yaitu:

- 1) Jaringan lapisan tunggal jaringan di mana *input* diproses langsung ke *output* tanpa melalui lapisan tersembunyi (Hasan et al., 2019).

2) Jaringan banyak lapisan

Dibandingkan dengan jaringan lapisan tunggal, jaringan banyak lapisan dapat mengatasi permasalahan kompleks karena memiliki satu atau lebih lapisan tersembunyi. Akan tetapi, jaringan *multi layer* dalam pelatihannya jauh lebih rumit (Hasan et al., 2019).

e. Fungsi Aktivasi

Fungsi aktivasi bertujuan dalam penentuan *output* dari suatu *neuron*. Berikut berbagai fungsi aktivasi yang kerap dipergunakan pada jaringan syaraf tiruan:

1) Fungsi *Sigmoid Biner*

Fungsi sigmoid biner mempunyai nilai *output* antara 0 sampai 1 (Kusumadewi, 2021). Fungsi *sigmoid biner* dirumuskan sebagai berikut:

$$y = f(x) = \frac{1}{1 + e^{-x}} \quad (2.4)$$

Turunan fungsi *sigmoid biner*

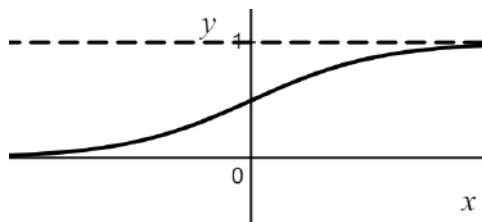
$$f'(x) = f(x)(1 - f(x)) \quad (2.5)$$

Keterangan:

$f(x)$: fungsi *sigmoid biner*

e^{-x} : nilai eksponensial

x : data



Gambar 2.1. Fungsi *sigmoid biner*

2) Fungsi *Sigmoid Bipolar*

Mirip dengan fungsi sigmoid biner, fungsi sigmoid bipolar juga mengadopsi bentuk kurva S, namun dengan perbedaan pada rentang hasil keluaran yang berada di antara -1 sampai 1 (Kusumadewi, 2021).

Berikut rumus dari fungsi *sigmoid bipolar*:

$$y = f(x) = \frac{1 - e^{-x}}{1 + e^{-x}} \quad (2.6)$$

Turunan fungsi *sigmoid bipolar*

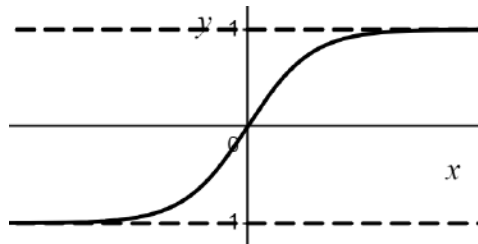
$$f'(x) = \frac{(1 + f(x))(1 - f(x))}{2} \quad (2.7)$$

Keterangan:

$f(x)$: fungsi *sigmoid bipolar*

e^{-x} : nilai eksponensial

x : data



Gambar 2.2. Fungsi *sigmoid bipolar*

5. *Backpropagation*

Backpropagation merupakan metode pembelajaran terawasi yang bertujuan untuk meminimalkan kesalahan dengan mengejar target tertentu. Ciri khas dari algoritma ini adalah proses penyesuaian bobot yang dilakukan untuk mengurangi selisih antara *output* yang diciptakan oleh jaringan dan target yang diinginkan. Pada umumnya, propagasi mundur diterapkan pada jaringan yang mempunyai beberapa lapisan *multilayer*, yang mencakup tiga komponen utama: lapisan *input*, lapisan tersembunyi, serta lapisan *output* (Anugerah, 2007). Dibandingkan dengan metode lain yang hanya memiliki satu lapisan tersembunyi, metode ini memanfaatkan keberadaan unit-unit khusus yang terorganisir di satu atau lebih lapisan tersembunyi, yang berperan dalam menangkap pola-pola kompleks dalam data (Siang, 2005).

Metode *Backpropagation* melibatkan penyesuaian bobot sesuai nilai *error* yang didapat. *Backpropagation* melatih jaringan untuk memanfaatkan penggunaan berbagai pola, sehingga menghasilkan *output* yang maksimal. Dibandingkan dengan metode lain yang sebatas memiliki satu lapisan tersembunyi, metode ini mempunyai keuntungan dengan mempunyai beberapa unit yang berada pada

satu atau lebih lapisan tersembunyi. (Siang, 2005).

Fungsi sigmoid biner, dengan rentang 0 hingga 1, adalah fungsi yang kerap dipergunakan pada metode ini (Kusumadewi, 2021). Data harus ditransformasikan atau dinormalisasi ke interval 0 hingga 1 jika fungsi aktivasi yang hendak dipergunakan yaitu fungsi sigmoid biner. Normalisasi data bertujuan untuk memperkecil jarak antar nilai data *input* dan mempermudah penghitungan bobot pada JST (Azka et al., 2018). Metode normalisasi min-max dapat digunakan untuk menormalisasi data dengan memanfaatkan penggunaan rumus berikut (Siang, 2005):

$$X_i^{norm} = \frac{0,8(x_i - \min(x))}{\max(x) - \min(x)} + 0,1 \quad (2.8)$$

Keterangan:

X_i^{norm} : Hasil normalisasi data x pada indeks ke-i

X_i : Data x pada indeks ke-i

$\min(x)$: Data minimum pada x

$\max(x)$: Data maksimum pada x

Sedangkan rumus berikut ini digunakan untuk mengembalikan data ke bentuk aslinya, yang juga dikenal sebagai denormalisasi (Siang, 2005).

$$d = \left[\frac{(d' - 0,1)(\max(x) - \min(x))}{0,8} \right] + \min(x) \quad (2.9)$$

Keterangan:

d : Nilai prediksi setelah denormalisasi

d' : Nilai prediksi sebelum denormalisasi

$\min(x)$: Data minimum pada x

$max(x)$: Data maksimum pada x

Ada tiga fase dalam prosedur metode *Backpropagation* (Siang, 2005), yakni:

a. Fase *Feedforward* (Propagasi maju)

Melalui penggunaan fungsi aktivasi yang sudah ditetapkan, perhitungan akan dilakukan dalam fase propagasi maju mulai dari *input layer*, bergerak ke *hidden layer*, dan pada akhirnya berakhir pada *output layer*.

b. Fase *Backpropagation* (Propagasi mundur)

Fase propagasi mundur adalah proses penghitungan *error* yang berasal dari penghitungan propagasi maju. Setelah membandingkan nilai *output* yang dihasilkan melalui propagasi maju dengan nilai target yang telah ditetapkan, kesalahan pada setiap elemen yang terkait dengan lapisan keluaran akan dianalisis dan dievaluasi.

c. Fase Perubahan Bobot

Pada fase perubahan bobot, bobot akan disesuaikan dengan nilai *error* yang dihasilkan dari propagasi mundur.

Secara detail, langkah-langkah algoritma *Backpropagation* yakni mencakup (Kusumadewi 2021):

Algoritma dari jaringan *Backpropagation* adalah seperti berikut:

- Inisialisasi bobot, umumnya bobot awal diinisiasi secara acak dengan nilai antara -0,5 hingga 0,5 atau interval lainnya.
- Tetapkan Maksimum *Epo*, Target *Error*, dan *Learning Rate* (α).

- Inisialisasi $Epoch = 0$
- Kerjakan langkah-langkah berikut selama ($Epoch < \text{Maksimum Epoch}$) dan ($MSE < \text{Target Error}$).

1. $Epoch = Epoch + 1$

2. Untuk setiap pasangan elemen yang akan dilakukan pembelajaran, kerjakan:

Tahap 1 Propagasi Maju

- Setiap unit *input* ($X_i, i = 1, 2, 3, \dots, n$) menerima serta mengirimkan sinyal *input* ke setiap unit di lapisan tersembunyi.
- Setiap unit di lapisan tersembunyi ($Z_j, j = 1, 2, 3, \dots, p$) menjumlahkan berbagai sinyal *input* terbobot.

$$Z_{in_j} = v_{0j} + \sum_{i=1}^n x_i v_{ij} \quad (2.10)$$

Menghitung fungsi aktivasi dari sinyal *output*.

$$Z_j = f(Z_{in_j}) = \frac{1}{1 + e^{-Z_{in_j}}} \quad (2.11)$$

Dan mengirim sinyal ke setiap unit lapisan *output*.

Keterangan:

Z_{in_j} : Unit pada lapisan tersembunyi

Z_j : Fungsi aktivasi

v_{0j} : Nilai bias

x_i : Nilai *input*

v_{ij} : Bobot

- c. Setiap unit di lapisan *output* ($Y_k, k = 1, 2, 3, \dots, m$) menjumlahkan berbagai sinyal tersembunyi terbobot.

$$Y_{in_k} = w_{0k} + \sum_{j=1}^m Z_j w_{jk} \quad (2.12)$$

Menghitung sinyal *output* dengan fungsi aktivasi.

$$Y_k = f(Y_{in_k}) = \frac{1}{1 + e^{-Y_{in_k}}} \quad (2.13)$$

Keterangan:

Y_{in_k} : Unit pada lapisan *output*

Y_k : Fungsi aktivasi

w_{0k} : Bias pada lapisan *output*

w_{jk} : Bobot antara lapisan tersembunyi dan lapisan *output*

Tahap 2 Propagasi Mundur

- d. Setiap unit *output* ($Y_k, k = 1, 2, 3, \dots, m$) mendapatkan pola target yang sesuai dengan pola *input* pelatihan dan menghitung informasi *error*nya:

$$\delta_k = (t_k - y_k) f'(y_{in_k}) = (t_k - Y_k) Y_k (1 - Y_k) \quad (2.14)$$

Kemudian hitung koreksi bobot untuk memperbaiki nilai w_{jk} :

$$\Delta w_{jk} = \alpha \delta_k z_j \quad (2.15)$$

Hitung juga koreksi bias untuk memperbaiki nilai w_{0k}

$$\Delta w_{0k} = \alpha \delta_k \quad (2.16)$$

Keterangan:

δ_k : Nilai *error* lapisan *output*

t_k : Target

Δw_{jk} : Nilai perubahan bobot

Δw_{0k} : Nilai perubahan bias

α : *Learning Rate*

- e. Setiap unit tersembunyi ($Z_j, j = 1, 2, 3, \dots, p$) menjumlahkan delta *input*nya.

$$\delta_{in_j} = \sum_{k=1}^m \delta_k w_{jk} \quad (2.17)$$

Untuk menghitung informasi *error*, kalikan nilai dengan fungsi aktivasinya.

$$\delta_j = \delta_{in_j} f'(z_{in_j}) \quad (2.18)$$

Menghitung koreksi bobot untuk memperbaiki nilai v_{ij}

$$\Delta v_{ij} = \alpha \delta_j x_i \quad (2.19)$$

Hitung juga koreksi bias untuk memperbaiki nilai v_{0j}

$$\Delta v_{0j} = \alpha \delta_j \quad (2.20)$$

Keterangan:

δ_{inj} : jumlah *error* pada lapisan tersembunyi

δ_j : nilai *error* pada lapisan tersembunyi

Δv_{ij} : nilai perubahan bobot antara lapisan tersembunyi dan lapisan *input*

Δv_{0j} : nilai perubahan bias lapisan tersembunyi

Tahap 3 Perubahan Bobot

- f. Tiap-tiap unit *output* ($Y_k, k = 1, \dots, m$) memperbaiki bobot dan biasnya.

$$w_{jk}(\text{baru}) = w_{jk}(\text{lama}) + \Delta w_{jk} \quad (2.21)$$

Setiap unit tersembunyi ($Z_j, j = 1, \dots, p$) memperbaiki bobot dan biasnya.

$$v_{ij}(\text{baru}) = v_{ij}(\text{lama}) + \Delta v_{ij} \quad (2.22)$$

3. Hitung MSE

6. *Extreme Learning Machine* (ELM)

Metode *Extreme Learning Machine* ialah Jaringan Syaraf Tiruan baru dengan satu lapisan tersembunyi, yang juga disebut sebagai Jaringan Syaraf *Single Hidden Layer Feedforward Neural Network* (SLFNs) (Huang et al., 2006). Metode ini dikembangkan guna meningkatkan kecepatan pembelajaran jaringan saraf tiruan feedforward. Metode *Extreme Learning Machine* tidak menjalankan pelatihan terhadap bobot atau bias pada lapisan masukan; sebaliknya, fokus utamanya adalah pada perolehan bobot pada lapisan keluaran dengan memanfaatkan invers

Moore-Penrose untuk menyelesaikan sistem linier secara umum (Humaini, 2015). ELM sering disebut sebagai metode pembelajaran mesin yang cepat, karena menggunakan bobot dan bias masukan yang ditetapkan secara acak, yang memungkinkan proses pembelajaran berlangsung lebih cepat dan menghasilkan kinerja generalisasi yang efisien (Huang et al., 2006).

Proses prediksi metode ELM terdiri dari beberapa langkah. Langkah-langkah ini melibatkan normalisasi, pelatihan, pengujian, dan denormalisasi data, serta penghitungan nilai kesalahan.

a. Normalisasi Data

Normalisasi data bertujuan untuk memperkecil jarak antar nilai data input kedalam skala tertentu. *Input* akan diproses ke nilai *output* yang kecil, oleh karena itu data yang dipergunakan perlu diubah untuk menghasilkan nilai normalisasi yang kecil (Pratiwi et al., 2019). Rumus normalisasi data sebagai berikut:

$$X_{inorm} = \frac{0,8(x_i - \min(x))}{\max(x) - \min(x)} + 0,1 \quad (2.23)$$

Keterangan:

X_{inorm} : Hasil normalisasi data

0,8 : Nilai default

X_i : Data

$\min(x)$: Data minimum

$\max(x)$: Data maksimum

0,1 : Nilai default (Windarto et al., 2020)

b. Proses *Training*

Sebelum melanjutkan ke proses prediksi, proses *training* perlu dikerjakan sebelumnya. Proses *training* ditujukan guna mendapat nilai *output weight*. Proses *training* terdiri dari langkah-langkah berikut (Huang et al., 2006):

- 1) Inisialisasi secara acak nilai bobot *input* dan bias dengan nilai mulai dari -0,5 hingga 0,5 (Kusumadewi, 2021).
- 2) Hitung keluaran *hidden layer* (H_{init}), persamaan berikut untuk menghitung *hidden layer*:

$$H_{inittrain} = X_{train} \cdot W^T \quad (2.24)$$

Keterangan:

$H_{inittrain}$: Matriks keluaran *hidden layer*

X_{train} : *Input data*

W^T : *Transpose bobot input*

Sesudah memperoleh dan menghitung H_{init} melalui penggunaan fungsi aktivasi sigmoid biner, fungsi aktivasi ini dipilih karena kemampuannya yang optimal dalam menangani masalah yang kompleks dan non-linier. Rumus fungsi aktivasi sigmoid biner sebagai berikut:

Setelah H_{init} diperoleh dan dihitung menggunakan fungsi aktivasi *sigmoid biner*. Fungsi aktivasi ini sangat ideal untuk menyelesaikan situasi yang kompleks dan non-linear.

$$H = \frac{1}{1 + e^{-H_{inittrain}}} \quad (2.25)$$

Keterangan:

H : Fungsi *sigmoid biner*

e : Eksponensial

$H_{inittrain}$: Matriks hasil keluaran *hidden layer*

- 3) Menghitung *output weight*. Matriks keluaran dari lapisan tersembunyi perlu mengalami perubahan urutan (transposisi) sebelum diterapkan fungsi aktivasi. Setelah itu, matriks yang telah ditransposisikan akan dikalikan dengan matriks keluaran lapisan tersembunyi menggunakan fungsi aktivasi. Tahapan selanjutnya ialah menghitung invers dari matriks yang sudah dimodifikasi sebelumnya. Kemudian, dilakukan perhitungan invers umum Moore-Penrose dari matriks keluaran lapisan tersembunyi dengan memanfaatkan fungsi aktivasi tersebut. Persamaan berikut untuk menghitung matriks *Moore-Penrose Generalized Invers*:

$$H^+ = (H^T H)^{-1} H^T \quad (2.26)$$

Keterangan:

H^+ : Matriks *Moore-Penrose Generalized Invers*

H : Matriks keluaran *hidden layer*

H^T : Matriks H transpose

Persamaan berikut untuk menghitung nilai *output weight*:

$$\hat{\mathbf{B}} = H^+ \cdot Y \quad (2.27)$$

Keterangan:

$\hat{\mathbf{B}}$: Matriks *output weight*

H^+ : Matriks *Moore-Penrose Generalized Invers*

Y : Matriks Target

c. Proses *Testing*

Tujuan dari proses ini adalah guna mengevaluasi pendekatan ELM dengan menggunakan hasil dari perhitungan sebelumnya. Implementasi *input weight*, bias, dan *output weight* dari proses *training* digunakan dalam proses *testing*. Berikut ini adalah berbagai langkah yang terlibat pada proses *testing*:

- 1) Menginisialisasi *input weight* dan bias yang telah didapatkan dari proses *training*.
- 2) Menghitung keluaran *hidden layer* (H_{init}), persamaan berikut untuk menghitung *hidden layer*:

$$H_{inittest} = X_{test} \cdot W^T \quad (2.28)$$

Keterangan:

$H_{inittest}$: Matriks hasil keluaran *hidden layer*

X_{test} : *Input data menggunakan data testing*

W^T : *Transpose input weight*

- 3) Keluaran di *hidden layer* dihitung menggunakan fungsi aktivasi.

$$H = \frac{1}{1 + e^{-H_{inittest}}} \quad (2.29)$$

Keterangan:

H : Fungsi aktivasi *sigmoid biner*

e : Eksponensial

$H_{inittest}$: Matriks hasil keluaran *hidden layer*

- 4) Nilai bobot keluaran yang dihasilkan selama proses *training* dipergunakan dalam perhitungan keluaran *output layer*, yang merepresentasikan hasil prediksi. Persamaan berikut digunakan dalam perhitungan *output layer*:

$$\hat{Y} = H \cdot \hat{B} \quad (2.30)$$

Keterangan:

\hat{Y} : *Output layer* yang merupakan hasil prediksi

\hat{B} : Nilai *output weight* dari proses *training*

H : Keluaran di *hidden layer* dihitung dengan fungsi aktivasi

d. Denormalisasi Data

Proses mengembalikan data yang masih dinormalisasi ke keadaan semula dikenal sebagai denormalisasi data. Berikut

ini adalah rumus denormalisasi data (Alfiyatin et al., 2018):

$$d = \left[\frac{(d' - 0, 1)(\max(x) - \min(x))}{0, 8} \right] + \min(x) \quad (2.31)$$

Keterangan:

d' : Nilai denormalisasi

d : Nilai setelah denormalisasi

$\min(x)$: Data minimum

$\max(x)$: Data maksimum

e. Mean Square Error (MSE)

Langkah terakhir adalah menentukan nilai *error* dari setiap *output layer* yang telah didenormalisasi, yang menampilkan nilai kesalahan dari hasil prediksi yang dihasilkan.

Mean Square Error (MSE) digunakan untuk mengevaluasi hasil prediksi. Persamaan berikut ini digunakan untuk menghitung nilai *error* pada hasil prediksi:

$$MSE = \frac{\sum_{t=1}^n (X_t - F_t)^2}{n} \quad (2.32)$$

Keterangan:

n : Total keseluruhan data

X_t : Data sebenarnya

F_t : Data prediksi

B. Kajian Pustaka

Terkait dengan judul penelitian, peneliti akan mengkaji beberapa penelitian terdahulu. Hal ini dimaksudkan agar dapat digunakan sebagai panduan dan pembanding bagi penelitian selanjutnya untuk mengumpulkan data-data yang berkaitan dengan judul tersebut. Di antara penelitian-penelitian tersebut adalah:

1. Penelitian yang dijalankan oleh Okky Putra Barus & Ronaldo dengan judul "Perbandingan Metode *Extreme Learning Machine* dan *Backpropagation* untuk Mengklasifikasi *Phishing Websites*". Penelitian ini dijalankan guna mendeteksi apakah suatu *website* termasuk kategori *phishing* atau tidak. Penelitian ini menyajikan metodologi pendeteksian *phishing websites* berbasis *machine learning* dengan menggunakan dan membandingkan metode *Extreme Learning Machine* (ELM) dan *Backpropagation*. Pada penelitian menunjukkan metode *backpropagation* dalam mengklasifikasi *phishing websites* memberikan ketepatan klasifikasi sebesar 91.85% lebih besar dibanding dengan metode ELM dengan ketepatan klasifikasi 84.07%. Penelitian yang dijalankan oleh Okky Putra Barus & Ronaldo memiliki kesamaan dengan penelitian ini dalam penggunaan metode *Extreme Learning Machine* dan *Backpropagation*.
2. Penelitian yang dijalankan oleh Heny Pratiwi & Kusno Harianto (2019) dengan judul "Perbandingan Algoritma ELM Dan *Backpropagation* Terhadap Prestasi Akademik Mahasiswa" pada Jurnal Sains Komputer & Informatika. Penelitian ini bertujuan untuk mengetahui algoritma mana

yang paling sesuai dalam mengetahui prestasi akademik mahasiswa. Variabel yang digunakan adalah nilai ujian masuk mahasiswa baru, nilai IP Semester 1, Jenis Kelamin, dan Status Bekerja, sedangkan variabel keluaran adalah nilai mutu sebagai klasifikasi kinerja akademik. Hasil penelitian ini menyatakan bahwa algoritma Extreme Learning Machine memiliki 14,84% tingkat kesalahan lebih rendah dibandingkan dengan Backpropagasi 28,20%. Dari tahapan pengujian model didapatkan hasil yang paling akurat adalah algoritma Extreme Learning Machine karena mempunyai tingkat akurasi paling tinggi dan tingkat kesalahan paling rendah. Penelitian yang dijalankan oleh Heny Pratiwi & Kusno Harianto memiliki kesamaan dengan penelitian ini dalam penggunaan algoritma ELM Dan *Backpropagation*.

3. Penelitian yang dijalankan oleh Adyan Nur Alfiyatin, dkk (2019) dengan judul “Penerapan *Extreme Learning Machine* (ELM) Untuk peramalan Laju Inflasi Di Indonesia”. Penelitian ini bermaksud guna memprediksi laju inflasi di Indonesia. Untuk itu, penelitian ini menggunakan metode kecerdasan buatan yakni *Extreme Learning Machine* (ELM). Kelebihan ELM yaitu cepat dalam proses pembelajaran. Berdasarkan pengujian yang dilakukan metode ELM mendapatkan nilai kesalahan sebesar 0,0202008, lebih kecil dibandingkan dengan metode *Backpropagation* sebesar 1,16035821. Hal tersebut membuktikan bahwa metode ELM sangat cocok digunakan untuk peramalan. Penelitian yang dijalankan oleh Adyan Nur Alfiyatin, dkk memiliki kesamaan dengan penelitian ini dalam penggunaan perbandingan metode

Extreme Learning Machine dan Backpropagation.

4. Penelitian yang dijalankan oleh Malau S Solikhun dengan judul Implementasi Jaringan Syaraf Tiruan *Backpropagation* Untuk Prediksi Jumlah Penerima Bantuan Sosial Pangan Di Sumatera Utara. Penelitian ini dijalankan guna memprediksi jumlah penerima bantuan sosial pangan dari pemerintah . Penelitian ini menggunakan data jumlah penerima bantuan sosial pangan dari tahun 2018-2022. Pada penelitian ini menggunakan model arsitektur 10-5-1 dan memperoleh nilai MSE sebesar 0,0000185136 dan akurasi 76%. Dari model arsitektur tersebut dapat digunakan untuk memprediksi jumlah penerima bantuan sosial di Sumatera Utara yang mengalami peningkatan dibandingkan dengan tahun sebelumnya. Penelitian yang dijalankan oleh Malau S Solikhun memiliki kesamaan dengan penelitian ini dalam penggunaan Jaringan Syaraf Tiruan *Backpropagation*.

BAB III

Metode Penelitian

A. Jenis Penelitian

Jenis penelitian yang dipergunakan yaitu penelitian kuantitatif. Penelitian kuantitatif ialah proses pengumpulan data untuk penelitian yang melibatkan angka-angka (Martono, 2011). Sedangkan sesuai perspektif Wahidmurni (2017) penelitian kuantitatif ialah cara untuk mendapat jawaban atas pertanyaan penelitian dengan menggunakan data serta program statistik.

B. Variabel Penelitian

Bersumber dari Silaen (2018) variabel dalam penelitian mengacu kepada konsep yang mempunyai rentang nilai yang beragam atau fluktuatif, yang menggambarkan karakteristik, atribut, atau fenomena yang dapat diobservasi atau dievaluasi, dengan nilai yang dapat bervariasi atau berubah-ubah. Dalam penelitian ini melibatkan variabel penelitian jumlah bantuan sosial BPNT di Kota Semarang tahun 2022-2023.

C. Sumber Data

Sumber data untuk penelitian ini yaitu data sekunder dalam bentuk data time series. Sesuai perspektif Kuncoro (2009) data sekunder ialah data yang sudah dihimpun pihak lain. Pada penelitian ini data sekunder didapat melalui Dinas Sosial Kota Semarang.

D. Metode Pengumpulan Data

Dokumentasi adalah pendekatan pengumpulan data yang dimanfaatkan pada penelitian ini. Dokumentasi ialah cara memperoleh data dari dokumen atau catatan tertulis yang ada. (Nasser, 2021). Dokumen atau data yang dipergunakan pada penelitian ini asalnya dari Dinas Sosial di Kota Semarang.

E. Metode Analisis Data

Metode *Backpropagation* dan *Extreme Learning Machine* dimanfaatkan guna menganalisis data pada penelitian ini, yang dijalankan melalui pemanfaatan perangkat lunak R Studio. Berbagai langkah pada analisis data melalui kedua pendekatan ini yakni mencakup:

1. *Backpropagation*

a. Pengumpulan Data

Data yang akan dipergunakan yaitu data jumlah penerima bantuan sosial BPNT Kota Semarang.

b. Pembagian Data

Penelitian ini menggunakan data *training* dan data *testing*.

c. Normalisasi Data

Proses merubah skala nilai data aktual ke skala yang lebih kecil tanpa mengubah makna nilai dikenal sebagai normalisasi data.

d. Menentukan Struktur dan Parameter Jaringan

Untuk mendapatkan arsitektur terbaik, struktur dan parameter jaringan akan ditetapkan pada titik ini.

e. Melakukan Proses *Training* dan *Testing*

Proses *training* ditujukan guna menurunkan nilai *error* model, sehingga dapat digunakan untuk memprediksi. Pada proses *testing* menggunakan parameter yang diperoleh dari proses *training*.

f. Menghitung nilai MSE dan MAPE.

MSE dan MAPE digunakan untuk mendapatkan model arsitektur jaringan terbaik.

g. Denormalisasi Data

mengembalikan skala nilai data yang masih dinormalisasi ke skala nilai sesungguhnya.

h. Hasil prediksi jumlah penerima bantuan.

Pada tahap ini didapat hasil prediksi jumlah penerima bantuan dengan arsitektur jaringan terbaik yang akan digunakan dalam prediksi.



Gambar 3.1. *flowchart* metode *Bacpropagation*

2. *Extreme Learning Machine*

- a. Menyiapkan data yang akan dimanfaatkan.

Data yang akan dimanfaatkan yaitu jumlah penerima bantuan BPNT Kota Semarang Bulan Januari 2022 sampai Desember 2023.

- b. Membagi data

Penelitian ini menggunakan data *training* dan data

testing.

c. Melakukan normalisasi data.

Normalisasi data dijalankan guna mentransformasi rentang nilai asli menjadi skala yang lebih terkompresi, tanpa mengubah esensi atau informasi yang terkandung dalam nilai tersebut.

d. Melakukan inisialisasi *input weight*.

e. Melakukan proses *training* dan *testing*.

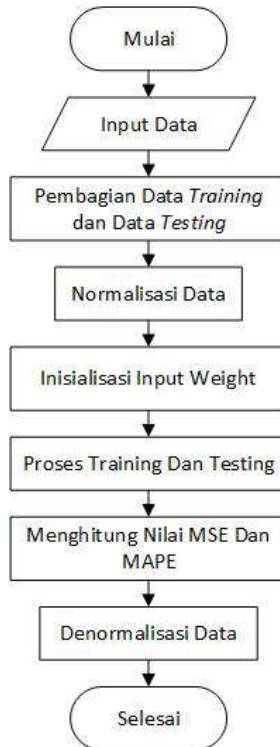
Proses *training* bermaksud guna menurunkan nilai *error* model, sehingga dapat digunakan untuk memprediksi. Pada proses *testing* memanfaatkan penggunaan parameter yang didapati dari proses *training*.

f. Melakukan perhitungan nilai MSE dan MAPE.

g. Melakukan denormalisasi data.

mengembalikan skala nilai data yang masih dinormalisasi ke skala nilai sesungguhnya.

h. Hasil prediksi jumlah bantuan.



Gambar 3.2. *Flowchart* metode ELM

BAB IV

HASIL DAN PEMBAHASAN

A. Analisis Deskriptif

Jumlah penerima BPNT yang dipergunakan dalam Skripsi ini ialah data yang terjadi selama 2 tahun. Data yang dimanfaatkan untuk simulasi prediksi ialah jumlah penerima BPNT di Kota Semarang. Data jumlah penerima BPNT diperoleh dari Dinas Sosial Kota Semarang pada bulan Januari 2022 hingga Desember 2023. Data jumlah penerima BPNT terlihat pada Tabel 4.1.

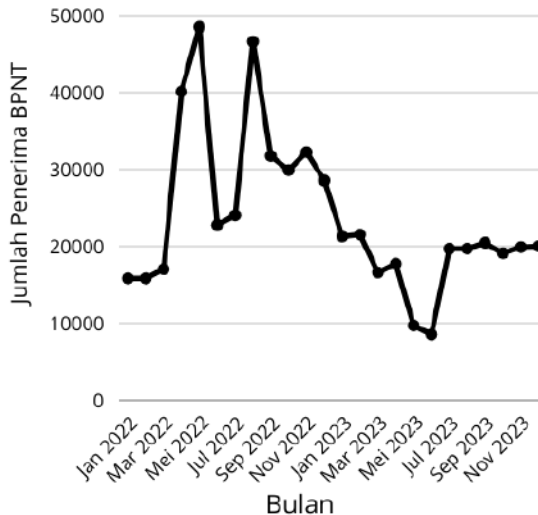
Tabel 4.1. Data jumlah penerima BPNT Kota Semarang

No.	Bulan	Penerima BPNT (Orang)
1	Januari 2022	15832
2	Februari 2022	16673
3	Maret 2022	17016
4	April 2022	40131
5	Mei 2022	48501
6	Juni 2022	22765
7	Juni 2022	24028
8	Agustus 2022	46607
9	September 2022	31734
10	Oktober 2022	29877
11	November 2022	32234
12	Desember 2022	28538
13	Januari 2023	21275
14	Februari 2023	21533
15	Maret 2023	16583
16	April 2023	17745
17	Mei 2023	9705
18	Juni 2023	8535
19	Juli 2023	19672

No.	Bulan	Penerima BPNT (Orang)
20	Agustus 2023	19679
21	September 2023	20452
22	Oktober 2023	19088
23	November 2023	19889
24	Desember 2023	20004

Tabel 4.1 di atas menunjukkan data penerima bantuan sosial BPNT Kota Semarang periode Januari 2022 sampai Desember 2023.

Penelitian ini mengkomparasikan dua metode pendekatan yakni metode *Backpropagation* dan *Extreme Learning Machine*. Dalam penelitian ini, analisis deskriptif dimanfaatkan guna menyajikan representasi umum terkait jumlah penerima BPNT di Kota Semarang pada Januari 2022 sampai Desember 2023.



Gambar 4.1. Grafik Data Penerima Bantuan BPNT

Gambar 4.1 di atas merupakan grafik jumlah penerima bantuan sosial BPNT Kota Semarang tahun 2022 sampai 2023. Jumlah penerima bantuan sosial BPNT terendah adalah 8535 yang terjadi pada bulan Juni 2023 dan jumlah penerima bantuan sosial BPNT tertinggi adalah 48501 yang terjadi pada bulan Mei 2022.

B. Metode *Backpropagation*

1. Normalisasi Data

Normalisasi data bertujuan untuk memperkecil jarak nilai antar data ke dalam skala tertentu. Dalam metode ini menggunakan fungsi aktivasi *sigmoid biner*, maka diperlukan normalisasi. Persamaan 2.8 digunakan untuk melakukan normalisasi.

$$X_{inorm} = \frac{0,8(x_i - \min(x))}{\max(x) - \min(x)} + 0,1$$

Keterangan:

$\min(x)$: Nilai terendah penerima BPNT

$\max(x)$: Nilai tertinggi penerima BPNT

$$x_1 = \frac{0,8(15832 - 8535)}{48501 - 8535} + 0,1 = 0,2460642$$

$$x_2 = \frac{0,8(16673 - 8535)}{48501 - 8535} + 0,1 = 0,2628985$$

$$x_3 = \frac{0,8(17016 - 8535)}{48501 - 8535} + 0,1 = 0,2697643$$

$$x_4 = \frac{0,8(40131 - 8535)}{48501 - 8535} + 0,1 = 0,7324576$$

$$x_5 = \frac{0,8(48501 - 8535)}{48501 - 8535} + 0,1 = 0,9000000$$

Perhitungan lengkap normalisasi terlihat pada lampiran 1

Tabel 4.2 di bawah merupakan hasil normalisasi data jumlah penerima BPNT.

Tabel 4.2. Data Normalisasi jumlah penerima BPNT Kota Semarang

No.	Data Normalisasi
1	0,2460642
2	0,2628985
3	0,2697643
4	0,7324576
5	0,9000000
6	0,3848421
7	0,4101236
8	0,8620878
9	0,5643747
10	0,5272031
11	0,5743832
12	0,5004003
13	0,3550168
14	0,3601812
15	0,2610969
16	0,2843567
17	0,1234199
18	0,1000000
19	0,3229295
20	0,3230696
21	0,3385428
22	0,3112396
23	0,3272732
24	0,3295751

2. Pola *Input* dan *Output*

Berikut pola data yang dipergunakan pada penelitian ini:

Tabel 4.3. Pola *Input* dan *Output*

No	<i>Input</i>					<i>Output</i>
1	X1	X2	X3	...	X12	X13
2	X2	X3	X4	...	X13	X14
3	X3	X4	X5	...	X14	X15
4	X4	X5	X6	...	X15	X16
5	X5	X6	X7	...	X16	X17
6	X6	X7	X8	...	X17	X18
7	X7	X8	X9	...	X18	X19
8	X8	X9	X10	...	X19	X20
9	X9	X10	X11	...	X20	X21
10	X10	X11	X12	...	X21	X22
11	X11	X12	X13	...	X22	X23
12	X12	X13	X14	...	X23	X24

Pada Tabel 4.3 menggunakan *neuron input* sebanyak 12 *neuron*, yang artinya data 1 tahun atau 12 bulan sebelumnya adalah data sebagai acuan untuk memprediksi *output*. *Neuron output* sebanyak 1 *neuron* yang merupakan data yang akan diprediksi. X1 didefinisikan sebagai bulan Januari 2022, X2 didefinisikan sebagai bulan Februari 2022, dan seterusnya sampai X24 didefinisikan sebagai bulan Desember 2023.

3. Pembagian Data *Training* dan Data *Testing*

Data *training* dan data *testing* adalah dua kategori di mana data dipisahkan selama proses ini. Tergantung pada peneliti, persentase pembagian data dapat berubah-ubah. Persentase data *training* lebih tinggi dari data *testing*. Komposisi data *testing* yang sering digunakan adalah 10% sampai 30% dari

data terakhir (Aisyah, 2021).

Pembagian data harus diperhatikan agar jaringan memiliki data *training* secukupnya dan data *testing* dapat melakukan proses pengujian berdasarkan hasil *training* yang dilakukan. Jaringan mungkin tidak dapat mempelajari data dengan benar jika tidak ada cukup data yang tersedia untuk fase *training*. Masalah lain dengan data yang terlalu banyak adalah bahwa hal itu akan memperlambat pemusatan proses pelatihan (konvergensi). Alih-alih menggeneralisasi data masukan, pelatihan yang berlebihan justru membuat jaringan menghafalnya (Rufiyanti, 2015).

Proporsi pembagian data *training* dan data *testing* yang digunakan dalam penelitian ini ialah 90% data *training* dan 10% data *testing*, 80% data *training* dan 20% data *testing*, dan 70% data *training* dan 30% data *testing*. Dari proporsi-proporsi tersebut, akan dipilih pembagian data terbaik. Berikut adalah hasil evaluasi untuk masing-masing proporsi pembagian data *training* dan data *testing*.

Tabel 4.4. Pembagian Data

Pembagian Data	MAPE <i>Training</i>	MAPE <i>Testing</i>
70%-30%	1,026038	0,200165
80%-20%	0,627123	0,109657
90%-10%	0,781099	0,217838

Pada Tabel 4.4 diperoleh pembagian data training 80% dan data testing 20% mendapatkan nilai MAPE testing terendah. Penelitian ini akan menggunakan pembagian data 80% data *training* dan 20% data *testing*.

4. Arsitektur Jaringan *Backpropagation*

Lapisan *input*, lapisan tersembunyi, serta lapisan *output* ialah tiga lapisan yang membentuk arsitektur jaringan saraf tiruan. Membuat arsitektur jaringan ialah tahap pertama dalam metode *Backpropagation* sebelum memulai tahap *training* dan *testing*. Menemukan model optimal yang akan menciptakan nilai *error* paling kecil serta paling akurat adalah tujuan dari pembentukan arsitektur jaringan.

Agar JST dapat menghasilkan nilai *error* yang rendah, arsitektur jaringan harus dirancang dengan baik. Namun dalam menentukan jumlah *neuron* pada *hidden layer* dilakukan dengan model empiris bukan secara teoritis (Kesuma, 2023). Hal tersebut disebabkan karena tidak ada aturan yang pasti untuk menentukan parameter jaringan yang ideal, seperti jumlah *neuron* pada *hidden layer*, sehingga dijalankan melalui *trial and error* (Kardan et al., 2013). Heaton (2008) mengemukakan beberapa aturan praktis untuk menentukan jumlah neuron yang tepat untuk digunakan dalam lapisan tersembunyi.

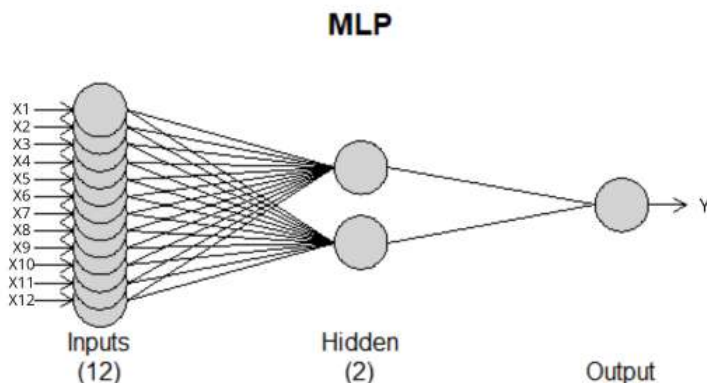
- a) Jumlah *hidden neuron* harus berada diantara ukuran *input layer* dan *output layer*.
- b) Jumlah *hidden neuron* harus $2/3$ ukuran *input layer*, ditambah ukuran *output layer*.
- c) Jumlah *hidden neuron* harus kurang dari dua kali ukuran *input layer*.

Pada kenyataannya, *overfitting* dapat terjadi karena menggunakan terlalu banyak *neuron* pada *hidden layer*.

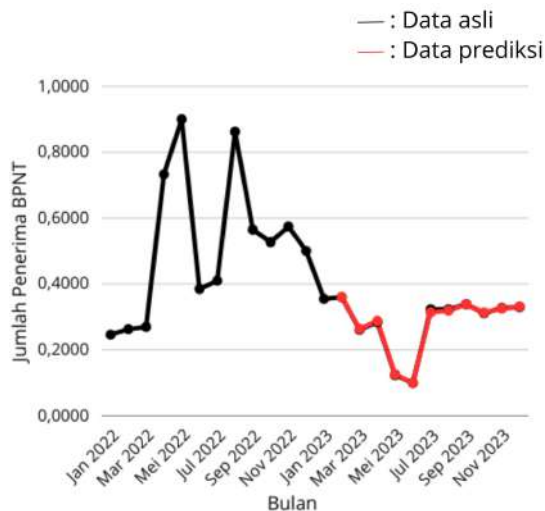
Sementara itu, *underfitting* dapat terjadi jika terlalu sedikit *neuron* yang digunakan (Pothuganti, 2018). *Overfitting* terjadi ketika model *Machine Learning* memiliki *error* yang rendah pada data *training* tetapi *error* yang jauh lebih tinggi pada data *testing* (Magdalena, 2021).

Berdasarkan *rules of thumb* untuk memperkirakan Jumlah *hidden neuron* (Heaton, 2008), maka pada penelitian ini menggunakan *hidden neuron* 2,9,16, dan 23 untuk menguji keempatan model yang memberikan nilai *error* paling sedikit. Proses perhitungan untuk model arsitektur 12-2-1, 12-9-1, 4-16-1, dan 12-23-1 adalah sebagai berikut:

a) *Backpropagation* 12-2-1



Gambar 4.2. *Backpropagation* 12-2-1
Arsitektur jaringan metode *Backpropagation* dengan pola 12-2-1 ditunjukkan pada gambar 4.2.



Gambar 4.3. Perbandingan data asli dan data prediksi Terlihat pada Gambar 4.3 yang merujuk pada hasil plot dari perbandingan data asli dan data prediksi pada data penerima BPNT di Kota Semarang. Sumbu y menampilkan jumlah data penerima BPNT yang belum didenormalisasi, sedangkan sumbu x menggambarkan waktu data penerima BPNT bulanan. Kurva berwarna hitam menunjukkan data asli penerima BPNT, sedangkan kurva berwarna merah menunjukkan data prediksi yang dipelajari oleh metode *Backpropagation*. Oleh karena itu, data tersebut harus didenormalisasi terlebih dahulu untuk dapat melihat data aslinya.

Berikut ini pada tabel 4.5 merupakan hasil prediksi penerima BPNT Kota Semarang menggunakan metode *Backpropagation* dengan pola arstiketur 12-2-1. Tabel 4.5 menjelaskan tentang perbandingan data asli dan data prediksi yang ditunjukkan dengan skala angka.

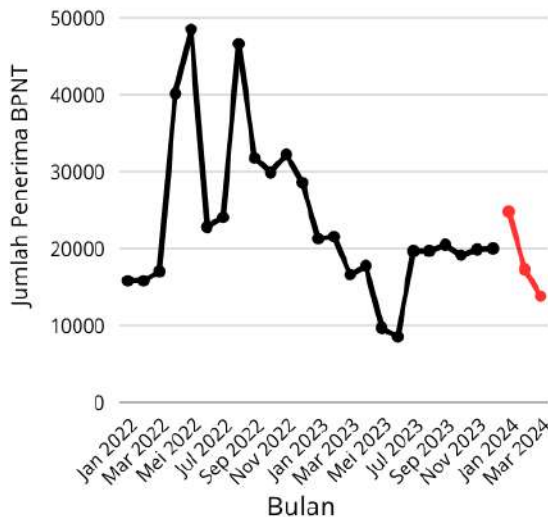
Tabel 4.5. Perbandingan nilai data metode *Backpropagation*

Bulan	Normalisasi		Denormalisasi	
	Data Asli	Data Prediksi	Data Asli	Data Prediksi
Feb 2023	0,3601812	0,36024087	21533	21536
Mar 2023	0,2610969	0,26379937	16583	16718
Apr 2023	0,2843567	0,28789209	17745	17922
Mei 2023	0,1234199	0,12577660	9705	9823
Jun 2023	0,1000000	0,09993418	8535	8532
Jul 2023	0,3229295	0,31431892	19672	19242
Agu 2023	0,3230696	0,31908312	19679	19480
Sep 2023	0,3385428	0,33800865	20452	20425
Okt 2023	0,3112396	0,31334658	19088	19193
Nov 2023	0,3272732	0,32596971	19889	19824
Des 2023	0,3295751	0,33131710	20004	20091

Hasil prediksi jumlah penerima BPNT di Kota Semarang antara bulan Januari hingga Maret 2024 dapat dilihat pada Tabel 4.6

Tabel 4.6. Hasil Prediksi metode *Backpropagation* pola 12-2-1

Bulan	Hasil Prediksi	Denormalisasi
Januari 2024	0,4257841	24810
Februari 2024	0,2756242	17309
Maret 2024	0,2055279	13807



Gambar 4.4. Hasil Prediksi jumlah penerima BPNT metode *Backpropagation*

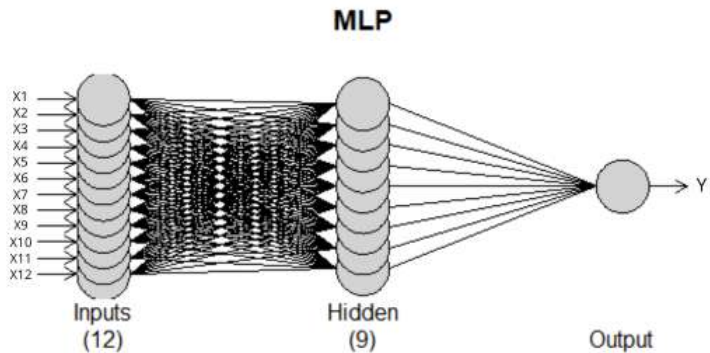
Hasil prediksi jumlah penerima BPNT ditampilkan pada Gambar 4.4. Data asli mengenai jumlah penerima BPNT dari bulan Januari 2022 hingga Desember 2023 ditampilkan oleh kurva hitam, sedangkan kurva berwarna merah menunjukkan data prediksi jumlah penerima BPNT bulan Januari 2024 hingga Maret 2024. Pada metode ini dilakukan analisis untuk menghitung nilai akurasi hasil prediksi metode *Backpropagation* dengan arsitektur jaringan 12-2-1. Analisis dilakukan dengan tujuan untuk mengetahui kelayakan dari metode *Backpropagation*. Program yang digunakan adalah *software R Studio* didapatkan nilai akurasi pada Tabel 4.7

Tabel 4.7. Nilai *error* metode *Backpropagation* 12-2-1

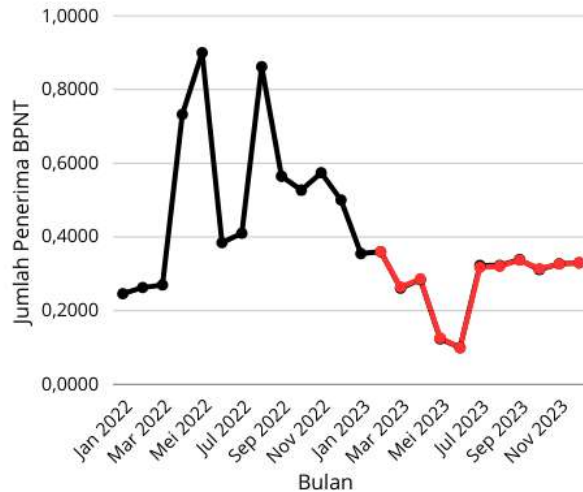
Parameter	Nilai
MSE	0,00001135
RMSE	0,00336906
MAPE	0,626098%

Berdasarkan Tabel 4.7, metode *Backpropagation* dengan model arsitektur 12-2-1 memiliki MAPE <10% dan nilai MSE mendekati nol, yang mengindikasikan bahwa metode ini memiliki kemampuan sangat baik dalam sistem prediksi.

b) *Backpropagation* 12-9-1



Gambar 4.5. *Backpropagation* 12-9-1
Arsitektur jaringan metode *Backpropagation* dengan pola 12-9-1 ditunjukkan pada gambar 4.5.



Gambar 4.6. Perbandingan data asli dan data prediksi Terlihat pada Gambar 4.6 yang merujuk pada hasil plot dari perbandingan data asli dan data prediksi pada data penerima BPNT di Kota Semarang. Sumbu y menampilkan jumlah data penerima BPNT yang belum didenormalisasi, sedangkan sumbu x menggambarkan waktu data penerima BPNT bulanan. Kurva berwarna hitam menunjukkan data asli penerima BPNT, sedangkan kurva berwarna merah menunjukkan data prediksi yang dipelajari oleh metode *Backpropagation*. Oleh karena itu, data tersebut harus didenormalisasi terlebih dahulu untuk dapat melihat data aslinya.

Berikut ini pada Tabel 4.8 merupakan hasil prediksi penerima BPNT Kota Semarang menggunakan metode *Backpropagation* dengan pola arstiketur 12-9-1. Tabel 4.8 menjelaskan tentang perbandingan data asli dan data prediksi yang ditunjukkan dengan skala angka.

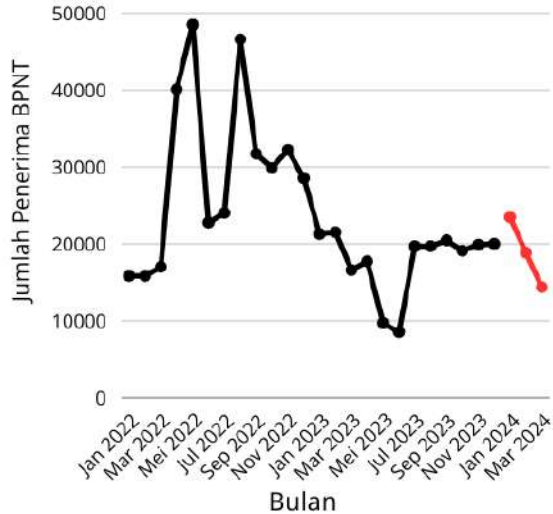
Tabel 4.8. Perbandingan nilai data metode *Backpropagation*

Bulan	Normalisasi		Denormalisasi	
	Data Asli	Data Prediksi	Data Asli	Data Prediksi
Feb 2023	0,3601812	0,35974201	21533	21511
Mar 2023	0,2610969	0,26454754	16583	16755
Apr 2023	0,2843567	0,28616078	17745	17835
Mei 2023	0,1234199	0,12634715	9705	9851
Jun 2023	0,1000000	0,09956011	8535	8513
Jul 2023	0,3229295	0,31728569	19672	19390
Agu 2023	0,3230696	0,32069604	19679	19560
Sep 2023	0,3385428	0,33722808	20452	20386
Okt 2023	0,3112396	0,31406626	19088	19229
Nov 2023	0,3272732	0,32652401	19889	19852
Des 2023	0,3295751	0,32982165	20004	20016

Hasil prediksi jumlah penerima BPNT di Kota Semarang antara bulan Januari hingga Maret 2024 dapat dilihat pada Tabel 4.9

Tabel 4.9. Hasil Prediksi metode *Backpropagation* pola 12-9-1

Bulan	Hasil Prediksi	Denormalisasi
Januari 2024	0,3988484	23465
Februari 2024	0,3064366	18848
Maret 2024	0,2167685	14368



Gambar 4.7. Hasil Prediksi jumlah penerima BPNT metode *Backpropagation*

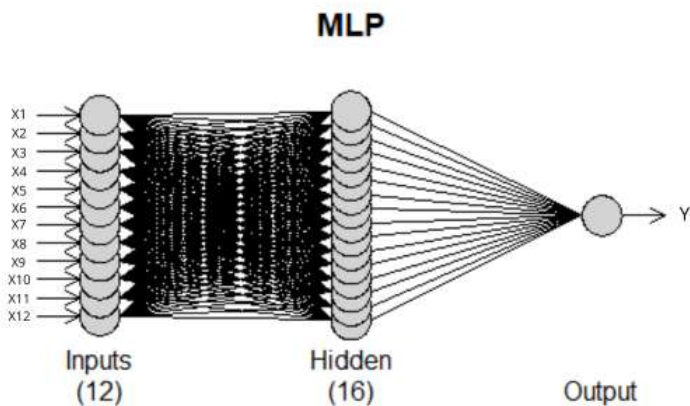
Hasil prediksi jumlah penerima BPNT ditampilkan pada Gambar 4.7. Data asli mengenai jumlah penerima BPNT dari bulan Januari 2022 hingga Desember 2023 ditampilkan oleh kurva hitam, sedangkan kurva berwarna merah menunjukkan data prediksi jumlah penerima BPNT bulan Januari 2024 hingga Maret 2024. Pada metode ini dilakukan analisis untuk menghitung nilai akurasi hasil prediksi metode *Backpropagation* dengan arsitektur jaringan 12-9-1. Analisis dilakukan dengan tujuan untuk mengetahui kelayakan dari metode *Backpropagation*. Program yang digunakan adalah *software R Studio* didapatkan nilai akurasi pada Tabel 4.10

Tabel 4.10. Nilai *error* metode *Backpropagation* 12-9-1

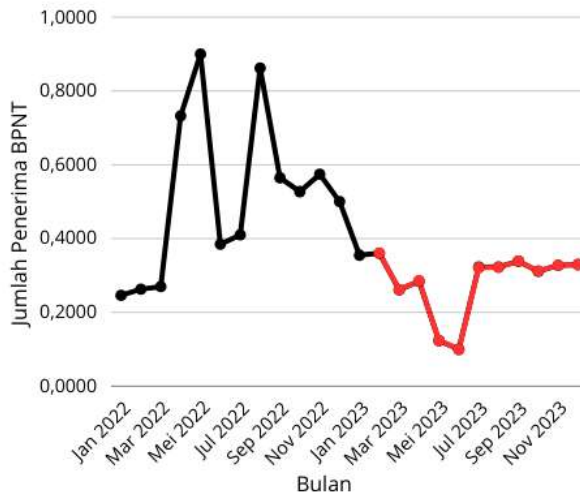
Parameter	Nilai
MSE	0,00000654
RMSE	0,00255740
MAPE	0,499265%

Berdasarkan Tabel 4.10, metode *Backpropagation* dengan model arsitektur 12-9-1 memiliki MAPE <10% dan nilai MSE mendekati nol, yang mengindikasikan bahwa metode ini memiliki kemampuan sangat baik dalam sistem prediksi..

c) *Backpropagation* 12-16-1



Gambar 4.8. *Backpropagation* 12-16-1
Arsitektur jaringan metode *Backpropagation* dengan pola 12-16-1 ditunjukkan pada gambar 4.8.



Gambar 4.9. Perbandingan data asli dan data prediksi Terlihat pada Gambar 4.9 yang merujuk pada hasil plot dari perbandingan data asli dan data prediksi pada data penerima BPNT di Kota Semarang. Sumbu y menampilkan jumlah data penerima BPNT yang belum didenormalisasi, sedangkan sumbu x menggambarkan waktu data penerima BPNT bulanan. Kurva berwarna hitam menunjukkan data asli penerima BPNT, sedangkan kurva berwarna merah menunjukkan data prediksi yang dipelajari oleh metode *Backpropagation*. Oleh karena itu, data tersebut harus didenormalisasi terlebih dahulu untuk dapat melihat data aslinya.

Berikut ini pada tabel 4.11 merupakan hasil prediksi penerima BPNT Kota Semarang menggunakan metode *Backpropagation* dengan pola arstiketur 12-16-1. Tabel 4.11 menjelaskan tentang perbandingan data asli dan data prediksi yang ditunjukkan dengan skala angka.

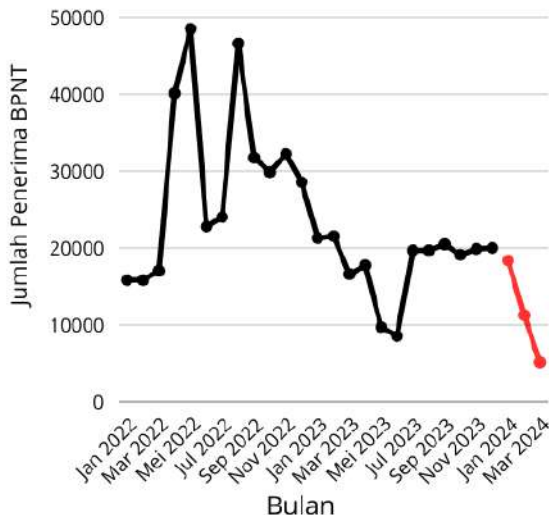
Tabel 4.11. Perbandingan nilai data metode *Backpropagation*

Bulan	Normalisasi		Denormalisasi	
	Data Asli	Data Prediksi	Data Asli	Data Prediksi
Feb 2023	0,3601812	0,36102644	21533	21575
Mar 2023	0,2610969	0,26184687	16583	16620
Apr 2023	0,2843567	0,28552328	17745	17803
Mei 2023	0,1234199	0,12429483	9705	9749
Jun 2023	0,1000000	0,09971365	8535	8521
Jul 2023	0,3229295	0,32066724	19672	19559
Agu 2023	0,3230696	0,32223937	19679	19638
Sep 2023	0,3385428	0,33872411	20452	20461
Okt 2023	0,3112396	0,31210410	19088	19131
Nov 2023	0,3272732	0,32739738	19889	19895
Des 2023	0,3295751	0,32904431	20004	19977

Hasil prediksi jumlah penerima BPNT di Kota Semarang antara bulan Januari hingga Maret 2024 dapat dilihat pada Tabel 4.12

Tabel 4.12. Hasil Prediksi metode *Backpropagation* pola 12-16-1

Bulan	Hasil Prediksi	Denormalisasi
Januari 2024	0,29654166	18354
Februari 2024	0,15422525	11244
Maret 2024	0,03160093	5118



Gambar 4.10. Hasil Prediksi jumlah penerima BPNT metode *Backpropagation*

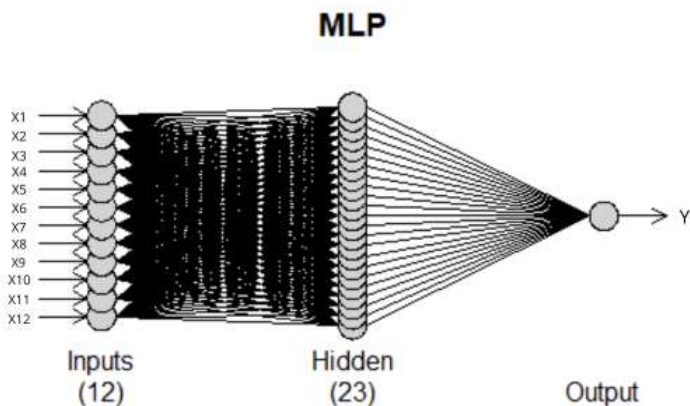
Hasil prediksi jumlah penerima BPNT ditampilkan pada Gambar 4.10. Data asli mengenai jumlah penerima BPNT dari bulan Januari 2022 hingga Desember 2023 ditampilkan oleh kurva hitam, sedangkan kurva berwarna merah menunjukkan data prediksi jumlah penerima BPNT bulan Januari 2024 hingga Maret 2024. Pada metode ini dilakukan analisis untuk menghitung nilai akurasi hasil prediksi metode *Backpropagation* dengan arsitektur jaringan 12-16-1. Analisis dilakukan dengan tujuan untuk mengetahui kelayakan dari metode *Backpropagation*. Program yang digunakan adalah *software R Studio* didapatkan nilai akurasi pada Tabel 4.13

Tabel 4.13. Nilai *error* metode *Backpropagation* 12-16-1

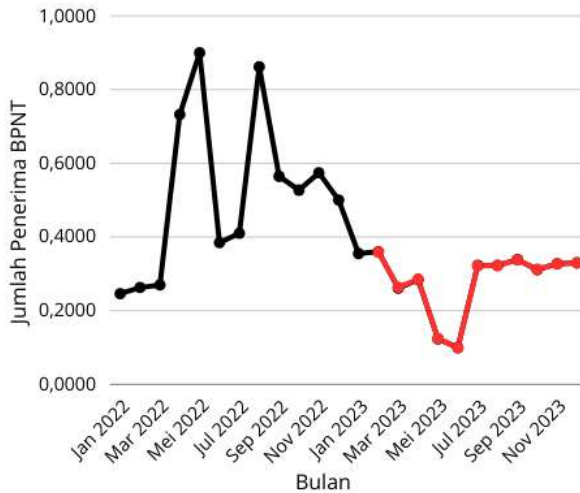
Parameter	Nilai
MSE	0,00000094
RMSE	0,00097093
MAPE	0,208451%

Berdasarkan Tabel 4.13, metode *Backpropagation* dengan model arsitektur 12-16-1 memiliki MAPE <10% dan nilai MSE mendekati nol, yang mengindikasikan bahwa metode ini memiliki kemampuan sangat baik dalam sistem prediksi.

d) *Backpropagation* 12-23-1



Gambar 4.11. *Backpropagation* 12-23-1
Arsitektur jaringan metode *Backpropagation* dengan pola 12-23-1 ditunjukkan pada gambar 4.11.



Gambar 4.12. Perbandingan data asli dan data prediksi Terlihat pada Gambar 4.12 yang merujuk pada hasil plot dari perbandingan data asli dan data prediksi pada data penerima BPNT di Kota Semarang. Sumbu y menampilkan jumlah data penerima BPNT yang belum didenormalisasi, sedangkan sumbu x menggambarkan waktu data penerima BPNT bulanan. Kurva berwarna hitam menunjukkan data asli penerima BPNT, sedangkan kurva berwarna merah menunjukkan data prediksi yang dipelajari oleh metode *Backpropagation*. Oleh karena itu, data tersebut harus didenormalisasi terlebih dahulu untuk dapat melihat data aslinya.

Berikut ini pada tabel 4.14 merupakan hasil prediksi penerima BPNT Kota Semarang menggunakan metode *Backpropagation* dengan pola arsitektur 12-23-1. Tabel 4.14 menjelaskan tentang perbandingan data aktual dan data prediksi yang ditunjukkan dengan skala angka.

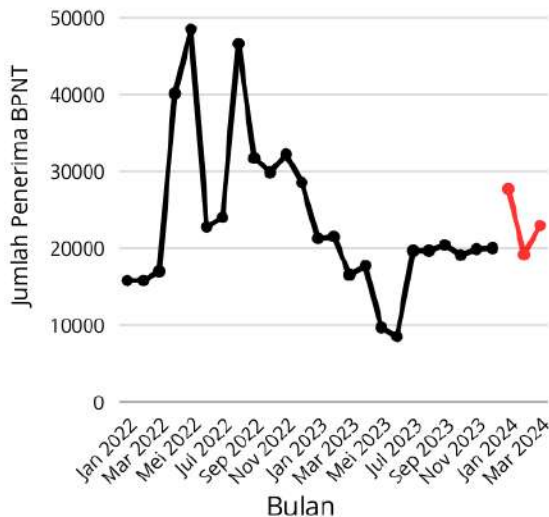
Tabel 4.14. Perbandingan nilai data metode *Backpropagation*

Bulan	Normalisasi		Denormalisasi	
	Data Asli	Data Prediksi	Data Asli	Data Prediksi
Feb 2023	0,3601812	0,35881160	21533	21465
Mar 2023	0,2610969	0,26361648	16583	16709
Apr 2023	0,2843567	0,28530897	17745	17793
Mei 2023	0,1234199	0,12425074	9705	9747
Jun 2023	0,1000000	0,09977164	8535	8524
Jul 2023	0,3229295	0,32122597	19672	19587
Agu 2023	0,3230696	0,32238323	19679	19645
Sep 2023	0,3385428	0,33816330	20452	20433
Okt 2023	0,3112396	0,31179502	19088	19116
Nov 2023	0,3272732	0,32699801	19889	19875
Des 2023	0,3295751	0,32965191	20004	20008

Hasil prediksi jumlah penerima BPNT di Kota Semarang antara bulan Januari hingga Maret 2024 dapat dilihat pada Tabel 4.15

Tabel 4.15. Hasil Prediksi metode *Backpropagation* pola 12-23-1

Bulan	Hasil Prediksi	Denormalisasi
Januari 2024	0,4837302	27705
Februari 2024	0,3138424	19218
Maret 2024	0,3888703	22966



Gambar 4.13. Hasil Prediksi jumlah penerima BPNT metode *Backpropagation*

Hasil prediksi jumlah penerima BPNT ditampilkan pada Gambar 4.13. Data asli jumlah penerima BPNT dari bulan Januari 2022 hingga Desember 2023 ditampilkan oleh kurva hitam, sedangkan kurva berwarna merah menunjukkan data prediksi jumlah penerima BPNT bulan Januari 2024 hingga Maret 2024.

Pada metode ini dilakukan analisis untuk menghitung nilai akurasi hasil prediksi metode *Backpropagation* dengan arsitektur jaringan 12-23-1. Analisis dilakukan dengan tujuan untuk mengetahui kelayakan dari metode *Backpropagation*. Program yang digunakan adalah *software R Studio* didapatkan nilai akurasi pada Tabel 4.16

Tabel 4.16. Nilai *error* metode *Backpropagation* 12-23-1

Parameter	Nilai
MSE	0,00000125
RMSE	0,00111926
MAPE	0,256797%

Berdasarkan Tabel 4.16, metode *Backpropagation* dengan model arsitektur 12-23-1 memiliki MAPE <10% dan nilai MSE mendekati nol, yang mengindikasikan bahwa metode ini memiliki kemampuan sangat baik dalam sistem prediksi.

5 Penentuan Model Arsitektur *Backpropagation*

Tabel 4.17. Model Arsitektur Jaringan

Model	Nilai MSE	Nilai MAPE %
12-2-1	0,00001135	0,626098
12-9-1	0,00000654	0,499265
12-16-1	0,00000094	0,208451
12-23-1	0,00000125	0,256797

Berdasarkan Tabel 4.17 menunjukkan bahwa keempat model tersebut, yaitu 12-2-1, 12-9-1, 12-16-1, dan 12-23-1 mempunyai nilai *error* MSE mendekati nol dan Nilai MAPE <10%, yang berarti kemampuan keempat model tersebut sangat baik (Hutasuhut et al., 2014). Akan tetapi, dapat diketahui bahwa model arsitektur 12-16-1 menjadi model terbaik karena menghasilkan nilai *error* MSE terkecil yaitu 0,00000094 dan nilai MAPE terkecil yaitu 0,208451%.

C. Metode *Extreme Learning Machine*

1. Normalisasi Data

Data jumlah penerima BPNT terlebih dahulu menjalani proses normalisasi. Proses normalisasi ini dilaksanakan dengan menerapkan rumus persamaan 2.8.

$$X_{inorm} = \frac{0,8(x_i - \min(x))}{\max(x) - \min(x)} + 0,1$$

Keterangan:

$\min(x)$: Nilai minimal penerima BPNT

$\max(x)$: Nilai maksimal penerima BPNT

$$x_1 = \frac{0,8(15832 - 8535)}{48501 - 8535} + 0,1 = 0,2460642$$

$$x_2 = \frac{0,8(16673 - 8535)}{48501 - 8535} + 0,1 = 0,2628985$$

$$x_3 = \frac{0,8(17016 - 8535)}{48501 - 8535} + 0,1 = 0,2697643$$

$$x_4 = \frac{0,8(40131 - 8535)}{48501 - 8535} + 0,1 = 0,7324576$$

$$x_5 = \frac{0,8(48501 - 8535)}{48501 - 8535} + 0,1 = 0,9000000$$

Perhitungan lengkap normalisasi dapat dilihat pada lampiran 1.

Tabel 4.18 di bawah merupakan hasil normalisasi data jumlah penerima BPNT.

Tabel 4.18. Data Normalisasi jumlah penerima BPNT Kota Semarang

No.	Data Normalisasi
1	0,2460642
2	0,2628985
3	0,2697643
4	0,7324576
5	0,9000000
6	0,3848421
7	0,4101236
8	0,8620878
9	0,5643747
10	0,5272031
11	0,5743832
12	0,5004003
13	0,3550168
14	0,3601812
15	0,2610969
16	0,2843567
17	0,1234199
18	0,1000000
19	0,3229295
20	0,3230696
21	0,3385428
22	0,3112396
23	0,3272732
24	0,3295751

2. Pembagian Data *Training* dan *Testing*

Proporsi pembagian data *training* dan data *testing* yang digunakan dalam penelitian ini ialah 90% data *training* dan 10% data *testing*, 80% data *training* dan 20% data *testing*, dan 70% data *training* dan 30% data *testing*. Dari proporsi-proporsi tersebut, akan dipilih pembagian data terbaik. Berikut adalah hasil evaluasi untuk masing-masing

proporsi pembagian data *training* dan data *testing*.

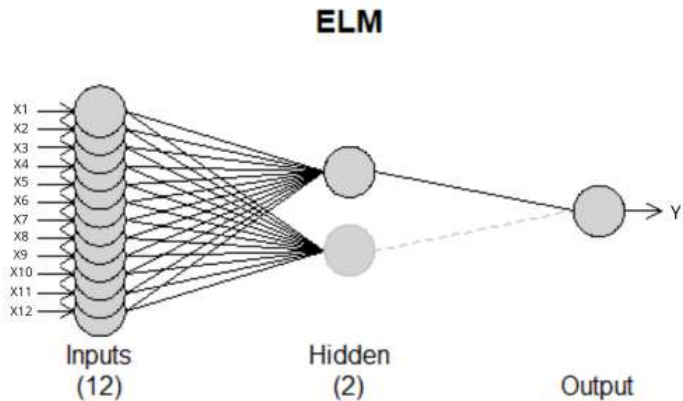
Tabel 4.19. Pembagian Data

Pembagian Data	MAPE <i>Training</i>	MAPE <i>Testing</i>
70%-30%	1,026038	0,200165
80%-20%	0,627123	0,109657
90%-10%	0,781099	0,217838

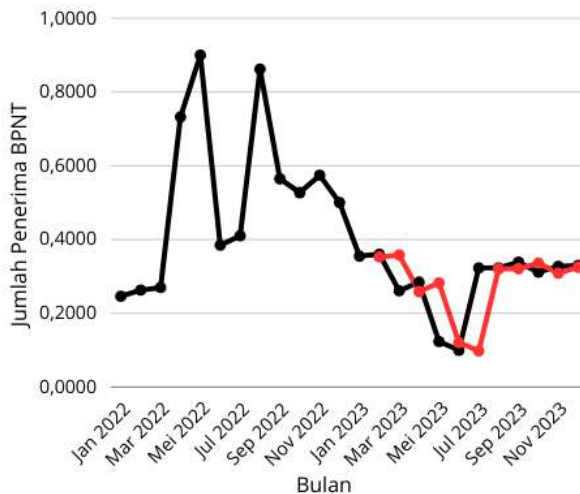
Pada Tabel 4.19 diperoleh pembagian data training 80% dan data testing 20% mendapatkan nilai MAPE testing terendah. Penelitian ini akan menggunakan pembagian data 80% data *training* dan 20% data *testing*.

3. Arsitektur Jaringan *Extreme Learning Machine*

a) *Extreme Learning Machine* 12-2-1



Gambar 4.14. *Extreme Learning Machine* 12-2-1
Arsitektur jaringan metode *Extreme Learning Machine* dengan pola 12-2-1 ditunjukkan pada gambar 4.14.



Gambar 4.15. Perbandingan data asli dan data prediksi Terlihat pada Gambar 4.15 yang merujuk pada hasil plot dari perbandingan data asli dan data prediksi pada data penerima BPNT di Kota Semarang. Sumbu y menampilkan jumlah data penerima BPNT yang belum dinormalisasi, sedangkan sumbu x menggambarkan waktu data penerima BPNT bulanan. Kurva berwarna hitam menunjukkan data asli penerima BPNT, sedangkan kurva berwarna merah menunjukkan data prediksi yang dipelajari oleh metode ELM. Oleh karena itu, data tersebut harus dinormalisasi terlebih dahulu untuk dapat melihat data aslinya.

Berikut ini pada tabel 4.20 merupakan hasil prediksi penerima BPNT Kota Semarang menggunakan metode ELM dengan pola arstiketur 12-2-1. Tabel 4.20 menjelaskan tentang perbandingan data asli dan data prediksi yang ditunjukkan dengan skala angka.

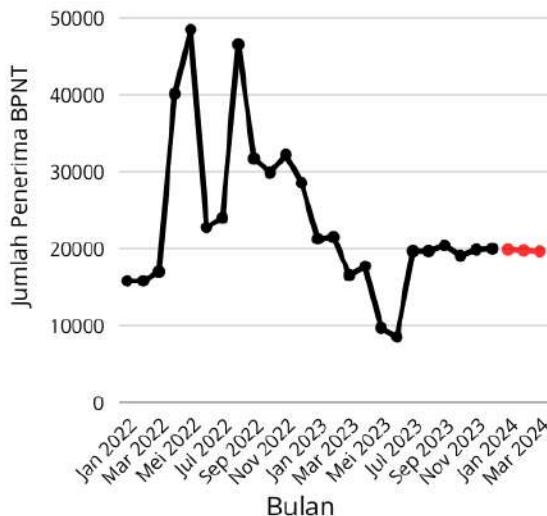
Tabel 4.20. Perbandingan nilai data metode *Extreme Learning Machine*

Bulan	Normalisasi		Denormalisasi	
	Data Asli	Data Prediksi	Data Asli	Data Prediksi
Feb 2023	0,3601812	0,35270389	21533	21159
Mar 2023	0,2610969	0,35786828	16583	21417
Apr 2023	0,2843567	0,25878406	17745	16467
Mei 2023	0,1234199	0,28204383	9705	17629
Jun 2023	0,1000000	0,12110703	8535	9589
Jul 2023	0,3229295	0,09768712	19672	8419
Agu 2023	0,3230696	0,32061662	19679	19556
Sep 2023	0,27046416	0,32075673	20452	19563
Okt 2023	0,3112396	0,33622989	19088	20336
Nov 2023	0,3272732	0,30892668	19889	18972
Des 2023	0,3295751	0,32496031	20004	19773

Hasil prediksi jumlah penerima BPNT di Kota Semarang antara bulan Januari hingga Maret 2024 dapat dilihat pada Tabel 4.21

Tabel 4.21. Hasil Prediksi metode *Extreme Learning Machine* pola 12-2-1

Bulan	Hasil Prediksi	Denormalisasi
Januari 2024	0,3272623	19888
Februari 2024	0,3249494	19773
Maret 2024	0,3226365	19657



Gambar 4.16. Hasil Prediksi jumlah penerima BPNT metode *Extreme Learning Machine*

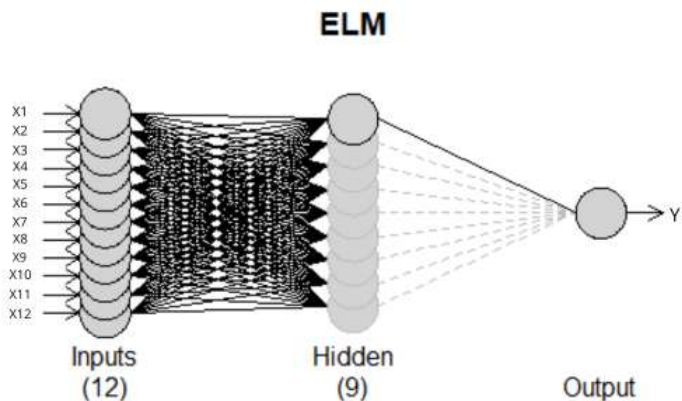
Hasil prediksi jumlah penerima BPNT ditampilkan pada Gambar 4.16. Data asli mengenai jumlah penerima BPNT dari bulan Januari 2022 hingga Desember 2023 ditampilkan oleh kurva hitam, sedangkan kurva berwarna merah menunjukkan data prediksi jumlah penerima BPNT bulan Januari 2024 hingga Maret 2024. Pada metode ini dilakukan analisis untuk menghitung nilai akurasi hasil prediksi metode *Extreme Learning Machine*. Analisis dilakukan dengan tujuan untuk mengetahui kelayakan dari metode *Extreme Learning Machine*. Program yang digunakan adalah *software R Studio* didapatkan nilai akurasi pada Tabel 4.22

Tabel 4.22. Nilai *error* metode *Extreme Learning Machine* 12-2-1

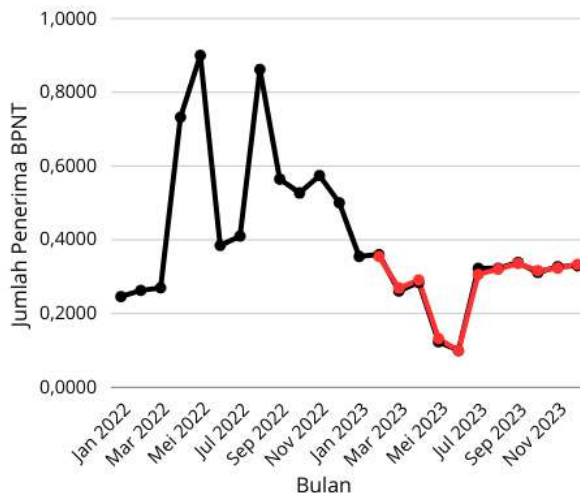
Parameter	Nilai
MSE	0,00797459
RMSE	0,08930057
MAPE	13,02096

Berdasarkan Tabel 4.22, metode *Extreme Learning Machine* dengan model arsitektur 12-2-1 memiliki MAPE 10-20% dan nilai MSE mendekati nol, yang mengindikasikan bahwa metode ini memiliki kemampuan baik dalam sistem prediksi.

b) *Extreme Learning Machine* 12-9-1



Gambar 4.17. *Extreme Learning Machine* 12-9-1
Arsitektur jaringan metode *Extreme Learning Machine* dengan pola 12-9-1 ditunjukkan pada gambar 4.17.



Gambar 4.18. Perbandingan data asli dan data prediksi Terlihat pada Gambar 4.18 yang merujuk pada hasil plot dari perbandingan data asli dan data prediksi pada data penerima BPNT di Kota Semarang. Sumbu y menampilkan jumlah data penerima BPNT yang belum didenormalisasi, sedangkan sumbu x menggambarkan waktu data penerima BPNT bulanan. Kurva berwarna hitam menunjukkan data asli penerima BPNT, sedangkan kurva berwarna merah menunjukkan data prediksi yang dipelajari oleh metode ELM. Oleh karena itu, data tersebut harus didenormalisasi terlebih dahulu untuk dapat melihat data aslinya.

Berikut ini pada tabel 4.23 merupakan hasil prediksi penerima BPNT Kota Semarang menggunakan metode ELM dengan pola arstiketur 12-9-1. Tabel 4.23 menjelaskan tentang perbandingan data asli dan data prediksi yang ditunjukkan dengan skala angka.

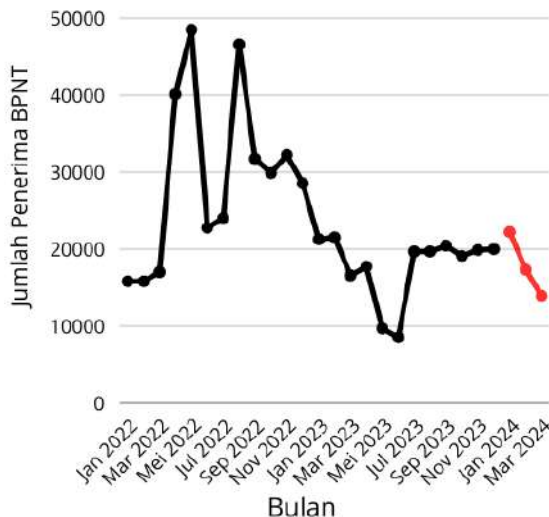
Tabel 4.23. Perbandingan nilai data metode *Extreme Learning Machine*

Bulan	Normalisasi		Denormalisasi	
	Data Asli	Data Prediksi	Data Asli	Data Prediksi
Feb 2023	0,3601812	0,3544998	21533	21249
Mar 2023	0,2610969	0,2694123	16583	16998
Apr 2023	0,2843567	0,2909636	17745	18075
Mei 2023	0,1234199	0,1323906	9705	10153
Jun 2023	0,1000000	0,0994102	8535	8506
Jul 2023	0,3229295	0,3062235	19672	18837
Agu 2023	0,3230696	0,3205781	19679	19555
Sep 2023	0,27046416	0,3358367	20452	20317
Okt 2023	0,3112396	0,3165977	19088	19356
Nov 2023	0,3272732	0,3232998	19889	19691
Des 2023	0,3295751	0,3337559	20004	20213

Hasil prediksi jumlah penerima BPNT di Kota Semarang antara bulan Januari hingga Maret 2024 dapat dilihat pada Tabel 4.24

Tabel 4.24. Hasil Prediksi metode *Extreme Learning Machine* pola 12-9-1

Bulan	Hasil Prediksi	Denormalisasi
Januari 2024	0,3742835	22238
Februari 2024	0,2763407	17345
Maret 2024	0,2084028	13951



Gambar 4.19. Hasil Prediksi jumlah penerima BPNT metode *Extreme Learning Machine*

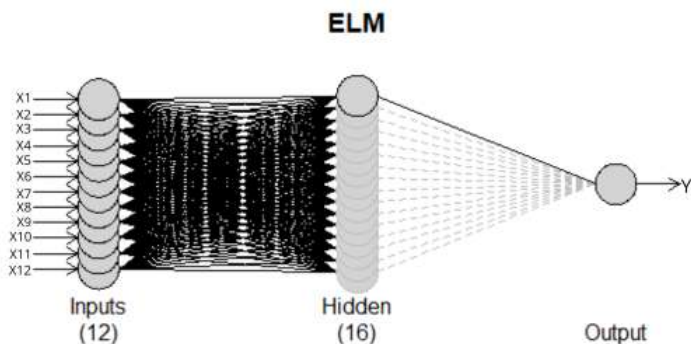
Hasil prediksi jumlah penerima BPNT ditampilkan pada Gambar 4.19. Data asli mengenai jumlah penerima BPNT dari bulan Januari 2022 hingga Desember 2023 ditampilkan oleh kurva hitam, sedangkan kurva berwarna merah menunjukkan data prediksi jumlah penerima BPNT bulan Januari 2024 hingga Maret 2024. Pada metode ini dilakukan analisis untuk menghitung nilai akurasi hasil prediksi metode *Extreme Learning Machine*. Analisis dilakukan dengan tujuan untuk mengetahui kelayakan dari metode *Extreme Learning Machine*. Program yang digunakan adalah *software R Studio* didapatkan nilai akurasi pada Tabel 4.25

Tabel 4.25. Nilai *error* metode *Extreme Learning Machine* 12-9-1

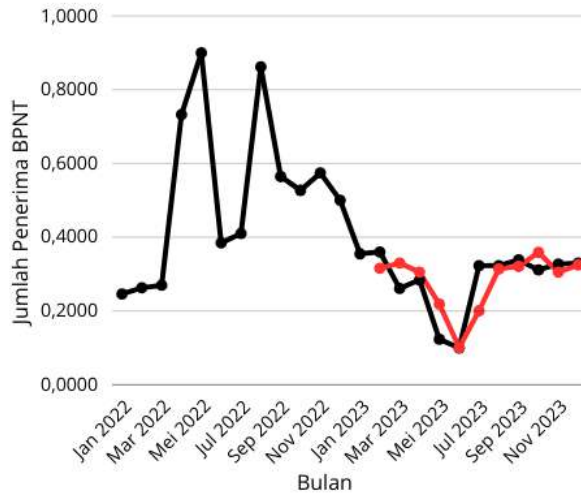
Parameter	Nilai
MSE	0,000052771
RMSE	0,007264428
MAPE	1,536778

Berdasarkan Tabel 4.25, metode *Extreme Learning Machine* dengan model arsitektur 12-9-1 memiliki MAPE <10% dan nilai MSE mendekati nol, yang mengindikasikan bahwa metode ini memiliki kemampuan sangat baik dalam sistem prediksi.

c) *Extreme Learning Machine* 12-16-1



Gambar 4.20. *Extreme Learning Machine* 12-16-1
Arsitektur jaringan metode *Extreme Learning Machine* dengan pola 12-16-1 ditunjukkan pada gambar 4.20.



Gambar 4.21. Perbandingan data asli dan data prediksi Terlihat pada Gambar 4.21 yang merujuk pada hasil plot dari perbandingan data asli dan data prediksi pada data penerima BPNT di Kota Semarang. Sumbu y menampilkan jumlah data penerima BPNT yang belum dinormalisasi, sedangkan sumbu x menggambarkan waktu data penerima BPNT bulanan. Kurva berwarna hitam menunjukkan data asli penerima BPNT, sedangkan kurva berwarna merah menunjukkan data prediksi yang dipelajari oleh metode ELM. Oleh karena itu, data tersebut harus dinormalisasi terlebih dahulu untuk dapat melihat data aslinya.

Berikut ini pada tabel 4.26 merupakan hasil prediksi penerima BPNT Kota Semarang menggunakan metode ELM dengan pola arstiketur 12-16-1. Tabel 4.26 menjelaskan tentang perbandingan data asli dan data prediksi yang ditunjukkan dengan skala angka.

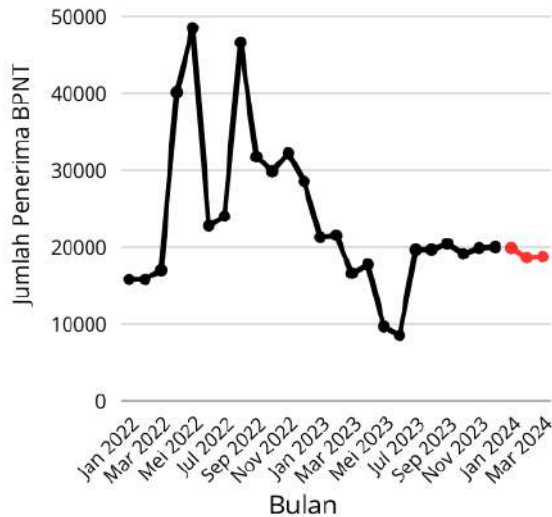
Tabel 4.26. Perbandingan nilai data metode *Extreme Learning Machine*

Bulan	Normalisasi		Denormalisasi	
	Data Asli	Data Prediksi	Data Asli	Data Prediksi
Feb 2023	0,3601812	0,3158616	21533	19319
Mar 2023	0,2610969	0,3299287	16583	20022
Apr 2023	0,2843567	0,3054496	17745	18799
Mei 2023	0,1234199	0,2185054	9705	14455
Jun 2023	0,1000000	0,1008046	8535	8575
Jul 2023	0,3229295	0,2006353	19672	13562
Agu 2023	0,3230696	0,3140930	19679	19231
Sep 2023	0,27046416	0,3207567	20452	19563
Okt 2023	0,3112396	0,3587742	19088	21463
Nov 2023	0,3272732	0,3053455	19889	18794
Des 2023	0,3295751	0,3249603	20004	19773

Hasil prediksi jumlah penerima BPNT di Kota Semarang antara bulan Januari hingga Maret 2024 dapat dilihat pada Tabel 4.27

Tabel 4.27. Hasil Prediksi metode *Extreme Learning Machine* pola 12-16-1

Bulan	Hasil Prediksi	Denormalisasi
Januari 2024	0,3272623	19888
Februari 2024	0,3022163	18637
Maret 2024	0,3044757	18750



Gambar 4.22. Hasil Prediksi jumlah penerima BPNT metode *Extreme Learning Machine*

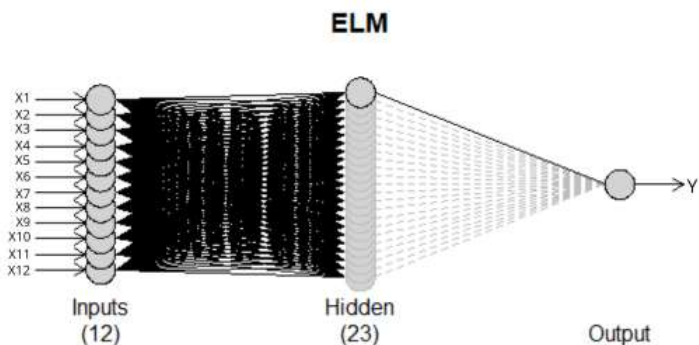
Hasil prediksi jumlah penerima BPNT ditampilkan pada Gambar 4.22. Data asli mengenai jumlah penerima BPNT dari bulan Januari 2022 hingga Desember 2023 ditampilkan oleh kurva hitam, sedangkan kurva berwarna merah menunjukkan data prediksi jumlah penerima BPNT bulan Januari 2024 hingga Maret 2024. Pada metode ini dilakukan analisis untuk menghitung nilai akurasi hasil prediksi metode *Extreme Learning Machine*. Analisis dilakukan dengan tujuan untuk mengetahui kelayakan dari metode *Extreme Learning Machine*. Program yang digunakan adalah *software R Studio* didapatkan nilai akurasi pada Tabel 4.28

Tabel 4.28. Nilai *error* metode *Extreme Learning Machine* 12-16-1

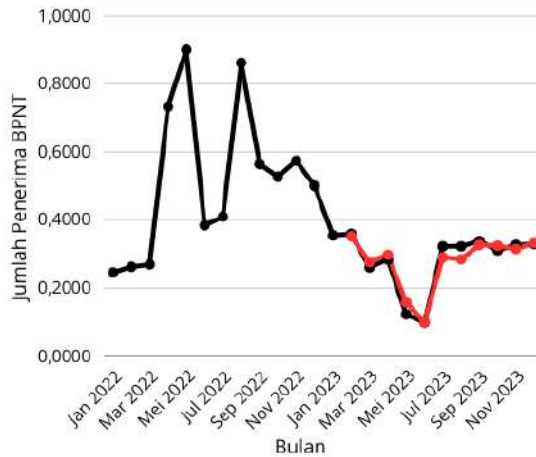
Parameter	Nilai
MSE	0,00311848
RMSE	0,05584335
MAPE	10,40893

Berdasarkan Tabel 4.28, metode *Extreme Learning Machine* dengan model arsitektur 12-16-1 memiliki MAPE 10-20% dan nilai MSE mendekati nol, yang mengindikasikan bahwa metode ini memiliki kemampuan baik dalam sistem prediksi.

d) *Extreme Learning Machine* 12-23-1



Gambar 4.23. *Extreme Learning Machine* 12-23-1
Arsitektur jaringan metode *Extreme Learning Machine* dengan pola 12-23-1 ditunjukkan pada gambar 4.23.



Gambar 4.24. Perbandingan data asli dan data prediksi

Terlihat pada Gambar 4.24 yang merujuk pada hasil plot dari perbandingan data asli dan data prediksi pada data penerima BPNT di Kota Semarang. Sumbu y menampilkan jumlah data penerima BPNT yang belum didenormalisasi, sedangkan sumbu x menggambarkan waktu data penerima BPNT bulanan. Kurva berwarna hitam menunjukkan data asli penerima BPNT, sedangkan kurva berwarna merah menunjukkan data prediksi yang dipelajari oleh metode ELM. Oleh karena itu, data tersebut harus didenormalisasi terlebih dahulu untuk dapat melihat data aslinya.

Berikut ini pada tabel 4.26 merupakan hasil prediksi penerima BPNT Kota Semarang menggunakan metode ELM dengan pola arstiketur 12-23-1. Tabel 4.29 menjelaskan tentang perbandingan data asli dan data prediksi yang ditunjukkan dengan skala angka.

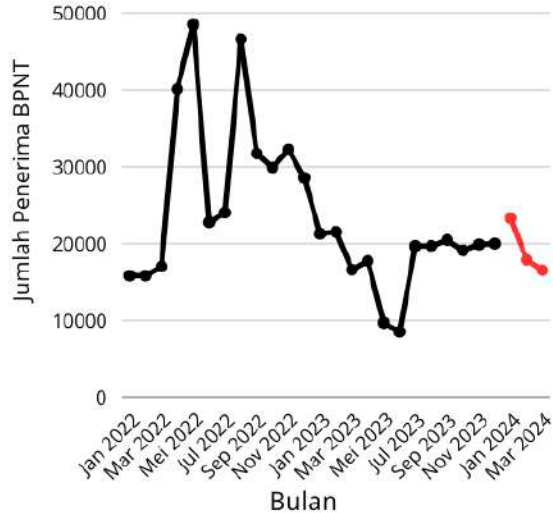
Tabel 4.29. Perbandingan nilai data metode *Extreme Learning Machine*

Bulan	Normalisasi		Denormalisasi	
	Data Asli	Data Prediksi	Data Asli	Data Prediksi
Feb 2023	0,3601812	0,35235551	21533	21142
Mar 2023	0,2610969	0,27678979	16583	17367
Apr 2023	0,2843567	0,29813886	17745	18434
Mei 2023	0,1234199	0,15865731	9705	11465
Jun 2023	0,1000000	0,09975119	8535	8523
Jul 2023	0,3229295	0,29067595	19672	18061
Agu 2023	0,3230696	0,28507504	19679	17781
Sep 2023	0,27046416	0,32734727	20452	19893
Okt 2023	0,3112396	0,32446917	19088	19749
Nov 2023	0,3272732	0,31518942	19889	19285
Des 2023	0,3295751	0,33312973	20004	20182

Hasil prediksi jumlah penerima BPNT di Kota Semarang antara bulan Januari hingga Maret 2024 dapat dilihat pada Tabel 4.30

Tabel 4.30. Hasil Prediksi metode *Extreme Learning Machine* pola 12-23-1

Bulan	Hasil Prediksi	Denormalisasi
Januari 2024	0,3958815	23317
Februari 2024	0,2871045	17882
Maret 2024	0,2601221	16534



Gambar 4.25. Hasil Prediksi jumlah penerima BPNT metode *Extreme Learning Machine*

Hasil prediksi jumlah penerima BPNT ditampilkan pada Gambar 4.25. Data asli mengenai jumlah penerima BPNT dari bulan Januari 2022 hingga Desember 2023 ditampilkan oleh kurva hitam, sedangkan kurva berwarna merah menunjukkan data prediksi jumlah penerima BPNT bulan Januari 2024 hingga Maret 2024. Pada metode ini dilakukan analisis untuk menghitung nilai akurasi hasil prediksi metode *Extreme Learning Machine*. Analisis dilakukan dengan tujuan untuk mengetahui kelayakan dari metode *Extreme Learning Machine*. Program yang digunakan adalah *software R Studio* didapatkan nilai akurasi pada Tabel 4.31

Tabel 4.31. Nilai *error* metode *Extreme Learning Machine* 12-23-1

Parameter	Nilai
MSE	0,00042564
RMSE	0,02063114
MAPE	3,93598

Berdasarkan Tabel 4.31, metode *Extreme Learning Machine* dengan model arsitektur 12-23-1 memiliki MAPE <10% dan nilai MSE mendekati nol, yang mengindikasikan bahwa metode ini memiliki kemampuan sangat baik dalam sistem prediksi.

4. Penentuan Model Arsitektur *Extreme Learning Machine*

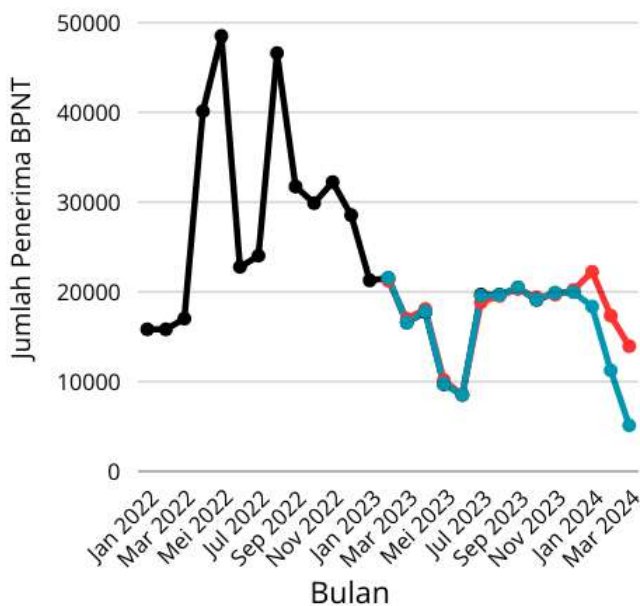
Tabel 4.32. Model Arsitektur Jaringan

Model	Nilai MSE	Nilai MAPE %
12-2-1	0,00797459	13,0209
12-9-1	0,00005277	1,53677
12-16-1	0,00311848	10,4089
12-23-1	0,00042564	3,93598

Berlandaskan Tabel 4.32 memperlihatkan bahwasanya keempat model tersebut, yakni 12-2-1, 12-9-1, 12-16-1, dan 12-23-1 mempunyai nilai *error* MSE dan MAPE mendekati nol. Akan tetapi, dapat diketahui bahwa model arsitektur 12-9-1 menjadi model terbaik karena menghasilkan nilai *error* MSE terkecil yaitu 0,00005277 dan nilai MAPE terkecil yaitu 1,53677%.

D. Perbandingan Metode *Backpropagation* dan Metode *Extreme Learning Machine*

Dalam proses ini, nilai akurasi dari setiap metode dianalisis untuk membandingkan hasil dari metode *Backpropagation* dan *Extreme Learning Machine*. Penelitian ini membandingkan nilai akurasi prediksi menggunakan MSE, RMSE, dan MAPE untuk menentukan metode terbaik untuk menghasilkan tingkat *error* terendah.



Gambar 4.26. Perbandingan Nilai Data Aktual, Nilai *Backpropagation*, Nilai ELM

Grafik perbandingan data asli dan data prediksi memanfaatkan penggunaan metode *Backpropagation* dan *Extreme Learning Machine* ditampilkan pada Gambar 4.26. Data aktual jumlah penerima BPNT ditampilkan pada kurva berwarna hitam, data

prediksi menggunakan metode *Backpropagation* ditampilkan pada kurva berwarna biru, dan data prediksi menggunakan pendekatan *Extreme Learning Machine* ditampilkan pada kurva berwarna merah.

Tabel 4.33. Perbandingan Nilai Akurasi Metode *Backpropagation* dan Metode ELM

Model	MSE	RMSE	MAPE (%)
<i>Backpropagation</i>	0,00000094	0,00097093	0,208451
ELM	0,00005277	0,00726442	1,536778

Tabel 4.33 memperlihatkan bahwasanya nilai *Mean Square Error* (MSE) metode *Backpropagation* menyentuh angka 0,00000094, sementara nilai MSE metode *Extreme Learning Machine* sebesar 0,00005277. Hasil tersebut memperlihatkan bahwasanya tingkat kesalahan metode *Backpropagation* lebih rendah dibandingkan dengan metode *Extreme Learning Machine*. Hasil penelitian memperlihatkan bahwasanya metode *Backpropagation* memiliki tingkat kesalahan yang lebih rendah dibandingkan dengan pendekatan *Extreme Learning Machine*, dengan nilai RMSE pada metode *Backpropagation* sebesar 0,00097093 dan metode *Extreme Learning Machine* sebesar 0,00726442. Hasil penelitian memperlihatkan bahwasanya metode *Backpropagation* memiliki tingkat kesalahan yang lebih rendah daripada pendekatan *Extreme Learning Machine*, dengan nilai MAPE pada metode *Backpropagation* sebesar 0,208451 dan pada metode *Extreme Learning Machine* sebesar 1,536778.

BAB V

PENUTUP

A. Kesimpulan

Kesimpulan yang dapat diambil dari hasil penerapan metode *Backpropagation* dan *Extreme Learning Machine* dalam meramalkan jumlah penerima BPNT di Kota Semarang adalah sebagai berikut:

1. Hasil Prediksi jumlah Penerima BPNT untuk metode *Backpropagation* pada bulan Januari 2024 sebesar 18354, bulan Februari 2024 sebesar 11244, dan bulan Maret 2024 sebesar 5118. Hasil Prediksi jumlah Penerima BPNT untuk metode *Extreme Learning Machine* pada bulan Januari 2024 sebesar 22238, bulan Februari 2024 sebesar 17345, dan bulan Maret 2024 sebesar 13951.
2. Hasil Perbandingan menunjukkan bahwa nilai Mean Square Error (MSE) metode *Backpropagation* sebesar 0,00000094, sementara nilai MSE metode *Extreme Learning Machine* sebesar 0,00005277. Hasil ini memperlihatkan bahwasanya tingkat kesalahan metode *Backpropagation* lebih rendah dibandingkan dengan metode *Extreme Learning Machine*. Hasil penelitian memperlihatkan bahwasanya metode *Backpropagation* memiliki tingkat kesalahan yang lebih rendah dibandingkan dengan pendekatan *Extreme Learning Machine*, dengan nilai RMSE pada metode *Backpropagation* sebesar 0,00097093 dan metode *Extreme Learning Machine* sebesar 0,00726442. Hasil penelitian memperlihatkan

bahwasanya metode *Backpropagation* memiliki tingkat kesalahan yang lebih rendah daripada pendekatan *Extreme Learning Machine*, dengan nilai MAPE pada metode *Backpropagation* sebesar 0,208451 dan pada metode *Extreme Learning Machine* sebesar 1,536778. Jadi, berdasarkan perbandingan nilai akurasi tersebut metode *Backpropagation* lebih baik daripada metode ELM.

B. Saran

Saran bagi penelitian ini meliputi:

- 1) Hanya data bulanan dari satu jenis bantuan Kota Semarang yang dilibatkan pada penelitian ini. Oleh karena itu, data yang lebih lengkap dari berbagai bentuk bantuan Kota Semarang dapat digunakan dalam penelitian selanjutnya.
- 2) Penelitian selanjutnya diharapkan dapat meneliti berbagai metode jaringan saraf tiruan tambahan yang mampu menghasilkan nilai yang lebih tepat.

DAFTAR PUSTAKA

- Adiguno, S., Syahra, Y., & Yetri, M. 2022. Prediksi Peningkatan Omset Penjualan Menggunakan Metode Regresi Linier Berganda. *Jurnal Sistem Informasi TGD*, 1(4), 275-281.
- Aisyah, S., Wahyuningsih, S., & Amijaya, F. D. S. 2021. Peramalan Jumlah Titik Panas Provinsi Kalimantan Timur Menggunakan Metode Radial Basis Function Neural Network. *Jambura Journal Of Probability And Statistics*, 2(2).
- Alfiyatin, A. N., Mahmudy, W. F., Ananda, C. F., & Anggodo, Y. P. 2018. Penerapan Extreme Learning Machine (ELM) Untuk Peramalan Laju Inflasi Di Indonesia Implementation Extreme Learning Machine for Inflation Forecasting in Indonesia. *Jurnal Teknologi Informasi Dan Ilmu Komputer (JTIK)*, 6(2), 179-186.
- Anugerah. 2007. *Perbandingan Jaringan Syaraf Tiruan Backpropogation dan Metode Deret Berkala Box-Jenkins (ARIMA) sebagai Metode Peramalan Curah Hujan*. Skripsi. Semarang: Universitas Negeri Semarang.
- Azka, T. H., Sofwan, A., & Sumardi. 2018. Perancangan Jaringan Sensor Nirkabel Untuk Pendeteksi Dini Terjadinya Tanah Longsor Dengan Jaringan Syaraf Tiruan Multilayer Perceptron. *TRANSIENT: Jurnal Ilmiah Teknik Elektro*, 7(3).
- Barus, O. P., & Ronaldo. 2019. Perbangingan Metode Extreme Learning Machine dan Backpropagation untuk

- Mengklasifikasi Phising Websites. *Informatics Engineering Research And Technology*, 1(1).
- Dasangga, D. G. R., & Cahyono, E. F. 2020. Analisis Peran Zakat Terhadap Pengentasan Kemiskinan Dengan Model Cibest (Studi Kasus Rumah Gemilang Indonesia Kampus Surabaya). *Jurnal Ekonomi Syariah Teori dan Terapan*, 7(6).
- Fikriya, Z. A., Irawan, M. I., & Soetrisno, S. 2017. Implementasi Extreme Learning Machine untuk Pengenalan Objek Citra Digital. *J. Sains dan Seni ITS*, 6(1).
- Hasan, N. F., Kusriani, & Fatta, H. A. 2019. Peramalan Jumlah Penjualan Menggunakan Jaringan Syaraf Tiruan Backpropagation Pada Perusahaan Air Minum Dalam Kemasan. *Jurnal Teknik Informatika dan Sistem Informasi*, 5(2).
- Heaton, J., 2008. *Introduction to Neural Networks for C, Second Edition*, Heaton Research, St Louis
- Hemeida, A. M. et al. 2020. Nature-inspired algorithms for feed-forward neural network classifiers: A survey of one decade of research. *Ain Shams Engineering Journal*, 11(3), 659–675.
- Huang, G., Bin, Zhu, Q.Y., & Siew, C.K. 2006. Extreme learning machine: Theory and applications. *Neurocomputing*, 70(1–3), 489–501.
- Humaini, Q. 2015. *Jaringan syarافتiruan extreme learning machine (ELM) untuk memprediksi kondisi cuaca di wilayah malang*. Skripsi. Malang: Uin Maulana Malik Ibrahim.

- Hutasuhut, A. H., Anggraeni, W., & Tyasnurita, R. 2014. Pembuatan Aplikasi Pendukung Keputusan untuk Peramalan Persediaan Bahan Baku Produksi Plastik Blowing dan Inject Menggunakan Metode ARIMA (Autoregressive Integrated Moving Average) di CV. Asia. *Jurnal Teknik ITS*, 3(2), A169-A174.
- Hutabarat, M. A. P., Handrizal, & Jalaluddin. 2020. Penerapan Algoritma Backpropagation Dalam Memprediksi Jumlah Penduduk di Kecamatan Pematang Bandar Berdasarkan Nagori/Kelurahan. *Journal of Information System Research (JOSH)*, 1(2), 63 - 69.
- Indonesia. Peraturan Menteri Sosial Nomor 5 Tahun 2021 Tentang Pelaksanaan Program Sembako.
- Indonesia. Sekretariat Direktorat Jenderal Penanganan Fakir Miskin. Kenali Lebih Dekat Program Bantuan Pangan Non Tunai. Diunduh di <https://kemensos.go.id/uploads/topics/15664651387355.pdf> tanggal 13 Desember 2024.
- Iswandy, E. 2015. Sistem Penunjang Keputusan Untuk Menentukan Penerimaan Dana Santunan Sosial Anak Nagari Dan Penyalurannya Bagi Mahasiswa Dan Pelajar Kurang Mampu Di Kenagarian Barung-Barung Balantai Timur. *Jurnal Teknoif*, 3(2).
- Junaidi, Mandasari, S., Franciska, Y., Fahmi, A., & Rosnelly, R. 2022. implementasi Jaringan Syaraf Tiruan Menggunakan Algoritma Backpropagation Dalam Meramalkan Kebutuhan

- Handsanitizer Di Pemerintah Kota Medan. *Journal of Science and Social Research*, 5(3), 671–676.
- Kardan, A. A., Sadeghi, H., Ghidary, S. S., & Sani, M. R. F. 2013. Prediction of Student Course Selection in Online Higher Education Institutes Using Neural Network. *Computers and Education*, 65, 1–11.
- Kesuma, I. M. S. A., Nugroho, A. S. B., Aminullah, A. 2023. Pengaruh Variasi Hidden Layer Terhadap Nilai MAPE Pada Pengembangan Model Estimasi Biaya Menggunakan Artificial Neural Network (Studi Kasus: Biaya Peningkatan Jalan Aspal di D.I.Yogyakarta). *SIKLUS: Jurnal Teknik Sipil*, 9(2), 152-163.
- Kusumadewi, S. 2021. *Membangun Jaringan Syaraf Tiruan Menggunakan Matlab & Excel Link*. Yogyakarta: Graha Ilmu
- Kuncoro, M. 2009. *Metode Riset untuk Bisnis & Ekonomi: Bagaimana Meneliti & Menulis Tesis?*. Jakarta: Erlangga.
- Magdalena, R., Saidah, S., Pratiwi, N. K. C., & Putra, A. T. 2021. Klasifikasi Tutupan Lahan Melalui Citra Satelit SPOT-6 dengan Metode Convolutional Neural Network (CNN). *JEPIN (Jurnal Edukasi dan Penelitian Informatika)*, 7(3).
- Maricar, M. A. 2019. Analisa Perbandingan Nilai Akurasi Moving Average dan Exponential Smoothing untuk Sistem Peramalan Pendapatan pada Perusahaan XYZ. *Jurnal Sistem Dan Informatika*.
- Martono, N. 2011. *Metode Penelitian Kuantitatif*. Jakarta: Raja Grafindo Persada.

- Muis, S. 2021. *Teori Jaringan Syaraf Tiruan: Sistem Kecerdasan Tiruan Dengan Kemampuan Belajar Dan Adaptasi Edisi 2*. Yogyakarta: Teknosain.
- Nasser, A. A. 2021. Sistem Penerimaan Siswa Baru Berbasis Web Dalam Meningkatkan Mutu Siswa Di Era Pandemi. *Biormatika: Jurnal Ilmiah Fakultas Keguruan Dan Ilmu Pendidikan*, 7(1), 100–109.
- Nugroho, P. 2021. Prediksi Nasabah Bank Menggunakan Algoritma Backpropagation. *Jurnal Mahasiswa Aplikasi Teknologi Komputer dan Informasi*, 3(3), 89-94.
- Nurhalizah, A. A., Cahyana, Y., & Rahmat. 2024. Model Prediksi Kekuatan Gempa Dengan Menggunakan Algoritma Linear Regression Dan Support Vector Regression (Studi Kasus BMKG). *Scientific Student Journal for Information, Technology and Science*, 5(2).
- Pothuganti, S. 2018. Review on Overfitting and Under-fitting Problems in Machine Learning and Solutions. *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, 7(9), 3692–3695.
- Pratiwi, P. G., Darma, I. P. K. G., & Putri, D. P. S. 2019. Peramalan Jumlah Tersangka Penyalahgunaan Narkoba Menggunakan Metode Multilayer Perceptron. *Jurnal Ilmiah Merpati (Menara Penelitian Akademika Teknologi Informasi)*, 7(2).
- Pratiwi, H., & Harianto, K. 2019. Perbandingan Algoritma ELM Dan Backpropagation Terhadap Prestasi Akademik Mahasiswa. *Jurnal Sains Komputer & Informatika*, 3(2), 282-294.

- Purba, R. H., Zarlis, M., & Gunawan, I. 2020. Algoritma Backpropagation dalam Memprediksi Jumlah Angka Kemiskinan di Provinsi Sumatera Utara. *Terapan Informatika Nusantara*, 1(1), 55–63.
- Rahayu, P. W., & Bernadus, I. N. 2021. Penerapan Metode Single Exponential Smoothing Pada Peramalan Penerimaan Siswa Baru (Studi Kasus SMK Wira Harapan). *Jurnal ilmu Komputer dan Bisnis (JIKB)*, 12(2a), 122-127.
- Rufiyanti, D. E. 2015. *Implementasi Jaringan Syaraf Tiruan Backpropagation dengan Input Model ARIMA untuk Peramalan Harga Saham*. Skripsi. Semarang: Universitas Negeri Semarang.
- Rumandan, R. J., Nuraini, R., Sadikin, N., & Rahmanto, Y. 2022. Klasifikasi Citra Jenis Daun Berkhasiat Obat Menggunakan Algoritma Jaringan Syaraf Tiruan Extreme Learning Machine. *Journal of Computer System and Informatics (JoSYC)*, 4(1), 145-154.
- Siang, J. J. 2005. *Jaringan Syaraf Tiruan & Pemrogramannya Menggunakan Matlab*. Yogyakarta: Andi Yogyakarta.
- Siang, J. J. 2009. *Jaringan Syaraf Tiruan & Pemrogramannya Menggunakan Matlab*. Yogyakarta: Andi Yogyakarta.
- Silaen, S. 2018. *Metodologi Penelitian Sosial untuk Penulisan Skripsi dan Tesis*. Bogor: In Media
- Sitompul, D., & Puspasari, R. 2023. Perancangan Aplikasi Peramalan Penjualan Minyak Solar PT. Willy Dwi Perkasa Menggunakan Metode Single Exponential Smoothing. Design

- of Solar Oil Sales Forecasting Application PT. Willy Dwi Perkasa uses the Single Exponential Smoothing Method. *Jurnal JUREKSI (Jurnal Rekayasa Sistem)*, 1(1), 191-204.
- Supriyanto, Sunardi, & Riadi, I. 2022. Pengaruh Nilai Hidden Layer Dan Learning Rate Terhadap Kecepatan Pelatihan Jaringan Syaraf Tiruan Backpropagation. *JIKO (Jurnal Informatika dan Komputer)*, 6(1), 27-33.
- Syahputra, M. D., Nurhayati, & Saragih, R. 2022. Jaringan Syaraf Tiruan Jumlah Penerima Bantuan Sembako Menggunakan Metode Backpropagation. *Jurnal Teknik Informatika Kaputama*, 6(2).
- Utami, S. G., Afriyanti, & Kalma, F. 2023. Implementasi Program Bantuan Pangan Non Tunai Dalam Meningkatkan Ekonomi Keluarga Di Desa Padangjantunng Kecamatan Siulak Kabupaten Kerinci. *Jurnal Administrasi Nusantara Maha (JAN Maha)*, 5(7).
- Wahidmurni. 2017. *Pemaparan Metode Penelitian Kuantitatif*. Malang: UIN Maulana Malik Ibrahim.
- Windarto, A. P., Nasution, D., Wanto, A., Tambunan, F., Hasibuan, M. S., Siregar, M. N. H., Lubis, M. R., Solikhun, Fadhillah, Y., & Nofriansyah, D. (2020). *Jaringan Syaraf Tiruan: Algoritma Prediksi dan Implementasi*. Medan: Yayasan Kita Menulis.

Lampiran 1. Normalisasi Data

$$x_1 = \frac{0,8(15832 - 8535)}{48501 - 8535} + 0,1 = 0,2460642$$

$$x_2 = \frac{0,8(16673 - 8535)}{48501 - 8535} + 0,1 = 0,2628985$$

$$x_3 = \frac{0,8(17016 - 8535)}{48501 - 8535} + 0,1 = 0,2697643$$

$$x_4 = \frac{0,8(40131 - 8535)}{48501 - 8535} + 0,1 = 0,7324576$$

$$x_5 = \frac{0,8(48501 - 8535)}{48501 - 8535} + 0,1 = 0,9000000$$

$$x_6 = \frac{0,8(22765 - 8535)}{48501 - 8535} + 0,1 = 0,3848421$$

$$x_7 = \frac{0,8(24028 - 8535)}{48501 - 8535} + 0,1 = 0,4101236$$

$$x_8 = \frac{0,8(46607 - 8535)}{48501 - 8535} + 0,1 = 0,8620878$$

$$x_9 = \frac{0,8(31734 - 8535)}{48501 - 8535} + 0,1 = 0,5643747$$

$$x_{10} = \frac{0,8(29877 - 8535)}{48501 - 8535} + 0,1 = 0,5272031$$

$$x_{11} = \frac{0,8(32234 - 8535)}{48501 - 8535} + 0,1 = 0,5743832$$

$$x_{12} = \frac{0,8(28538 - 8535)}{48501 - 8535} + 0,1 = 0,5004003$$

$$x_{13} = \frac{0,8(21275 - 8535)}{48501 - 8535} + 0,1 = 0,3550168$$

$$x_{14} = \frac{0,8(21533 - 8535)}{48501 - 8535} + 0,1 = 0,3601812$$

$$x_{15} = \frac{0,8(16583 - 8535)}{48501 - 8535} + 0,1 = 0,2610969$$

$$x_{16} = \frac{0,8(17745 - 8535)}{48501 - 8535} + 0,1 = 0,2843567$$

$$x_{17} = \frac{0,8(9705 - 8535)}{48501 - 8535} + 0,1 = 0,1234199$$

$$x_{18} = \frac{0,8(8535 - 8535)}{48501 - 8535} + 0,1 = 0,1000000$$

$$x_{19} = \frac{0,8(19672 - 8535)}{48501 - 8535} + 0,1 = 0,3229295$$

$$x_{20} = \frac{0,8(19679 - 8535)}{48501 - 8535} + 0,1 = 0,3230696$$

$$x_{21} = \frac{0,8(20452 - 8535)}{48501 - 8535} + 0,1 = 0,3385428$$

$$x_{22} = \frac{0,8(19088 - 8535)}{48501 - 8535} + 0,1 = 0,3112396$$

$$x_{23} = \frac{0,8(19889 - 8535)}{48501 - 8535} + 0,1 = 0,3272732$$

$$x_{24} = \frac{0,8(20004 - 8535)}{48501 - 8535} + 0,1 = 0,3295751$$

Lampiran 2. Langkah-langkah Backpropagation di R Studio

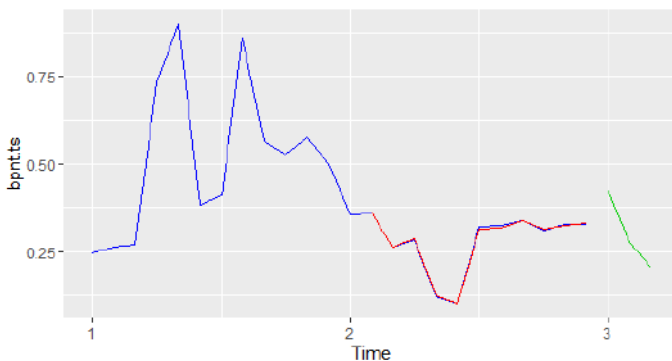
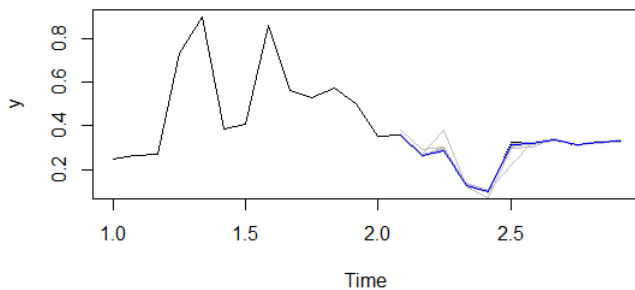
```
# Package
library(nnfor)
library(forecast)
bpnt <- read_excel("jurnal skripsi khoirul/bpnt.xlsx")
View(bpnt)
bpnt.ts=ts(bpnt$bpnt,frequency = 12)
bpnt.ts
plot(bpnt.ts)
# DATA TRAINING90% #
bpnt.train=ts(bpnt.ts[1:21],frequency = 12)
bpnt.train
#DATA TESTING 10% #
bpnt.test=ts(bpnt.ts[20:24],start=c(2,8), freq=12)
bpnt.test
#Backpropagation untuk Peramalan Data
bpnt.mlp<-mlp(bpnt.train,hd.auto.type="valid",hd=9,reprs=20,
lags=NULL,difforder=NULL,sel.lag=c(FALSE),outplot=c(TRUE))
bpnt.mlp
plot(bpnt.mlp)
bpnt.mlp$lags
bpnt.mlp$MSE
bpnt.mlp$fitted
bpnt.fcst<- forecast(bpnt.mlp, 3)
bpnt.fcst
accuracy(bpnt.fcst)
```

```
autoplot(bpnt.ts,color='blue')+autolayer(bpnt.mlp$fitted,color='red')
+autolayer(bpnt.fcst,color='green')
```

Output Pengujian

1. Backpropagation 12-2-1

```
> bpnt.mlp$MSE
[1] 1.135058e-05
> bpnt.fcst<- forecast(bpnt.mlp, 3)
> bpnt.fcst
      Jan      Feb      Mar
3 0.4257841 0.2756242 0.2055279
> accuracy(bpnt.fcst)
      ME      RMSE      MAE      MPE      MAPE  MASE      ACF1
Training set 0.0001815653 0.003369062 0.002454881 0.01498513 0.6260987 NaN 0.3859553
> bpnt.mlp$fitted
      Feb      Mar      Apr      May      Jun      Jul      Aug
2 0.36024087 0.26379937 0.28789209 0.12577660 0.09993418 0.31431892 0.31908312
      Sep      Oct      Nov      Dec
2 0.33800865 0.31334658 0.32596971 0.33131710
```

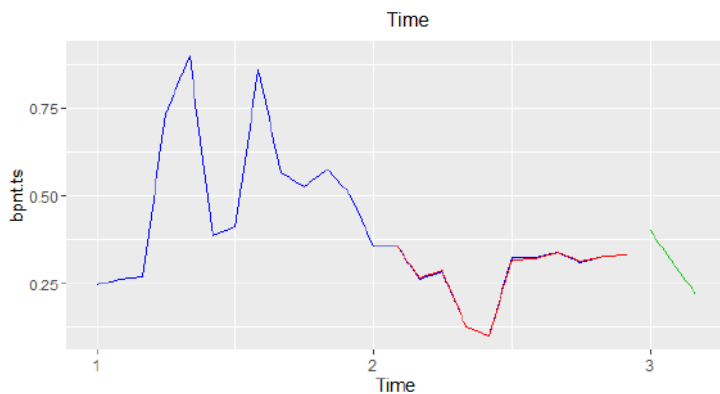
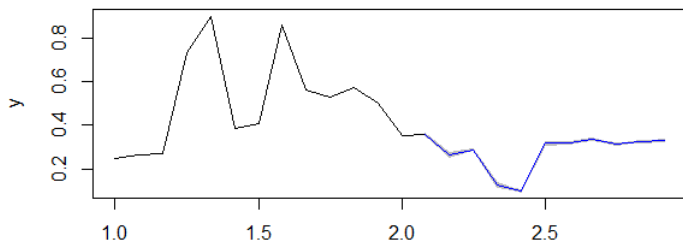


2. Backpropagation 12-9-1

```

> bpnt.mlp$MSE
[1] 6.540317e-06
> bpnt.mlp$fitted
      Feb      Mar      Apr      May      Jun      Jul      Aug
2 0.35974201 0.26454754 0.28616078 0.12634715 0.09956011 0.31728569 0.32069604
  Sep      Oct      Nov      Dec
2 0.33722808 0.31406626 0.32652401 0.32982165
> bpnt.fcst<- forecast(bpnt.mlp, 3)
> bpnt.fcst
      Jan      Feb      Mar
3 0.3988484 0.3064366 0.2167685
> accuracy(bpnt.fcst)
      ME      RMSE      MAE      MPE      MAPE      MASE
Training set -2.680722e-05 0.002557404 0.002019582 -0.0324989 0.4992654 NaN

```

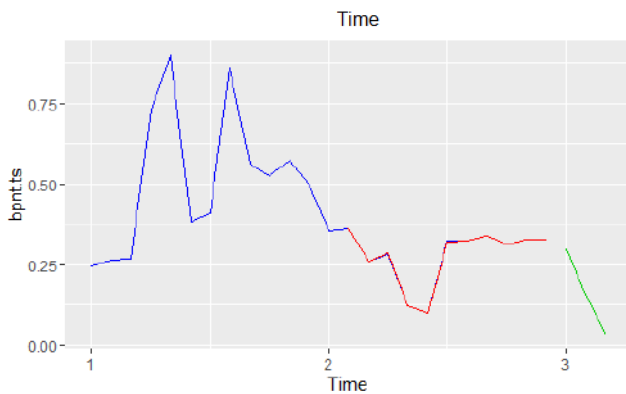
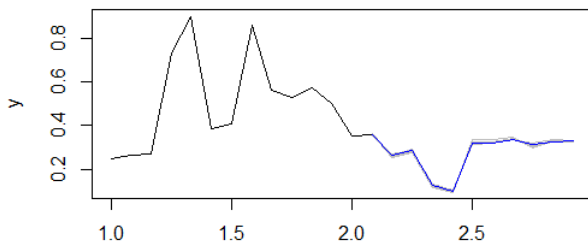


3. Backpropagation 12-16-1

```

> bpnt.mlp$MSE
[1] 9.427178e-07
> bpnt.mlp$fitted
      Feb      Mar      Apr      May      Jun      Jul      Aug      Sep
2 0.36102644 0.26184687 0.28552328 0.12429483 0.09971365 0.32066724 0.32223937 0.33872411
      Oct      Nov      Dec
2 0.31210410 0.32739738 0.32904431
> bpnt.fcst<- forecast(bpnt.mlp, 3)
> bpnt.fcst
      Jan      Feb      Mar
3 0.29654166 0.15422525 0.03160093
> accuracy(bpnt.fcst)
      ME      RMSE      MAE      MPE      MAPE      MASE      ACF1
Training set -8.156043e-05 0.0009709366 0.000792408 -0.04217728 0.2084513  NaN  0.4660129

```

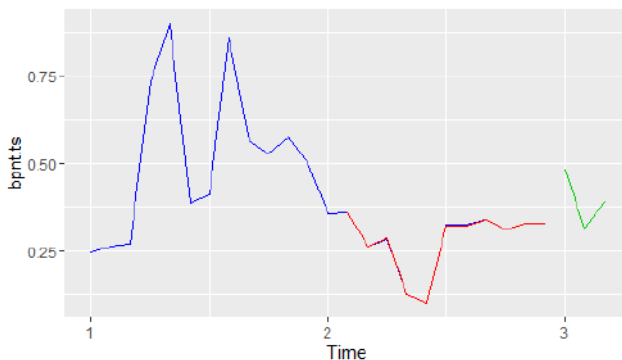
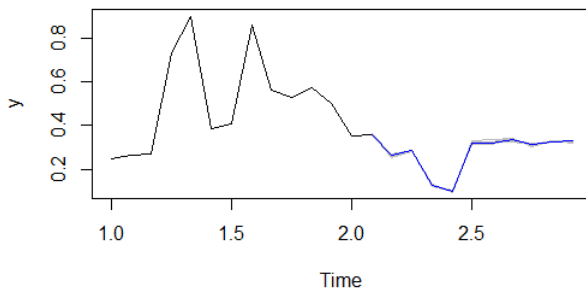


4. Backpropagation 12-23-1

```

> bpnt.mlp$MSE
[1] 1.252749e-06
> bpnt.mlp$fitted
      Feb      Mar      Apr      May      Jun      Jul      Aug      Sep
2 0.35881160 0.26361648 0.28530897 0.12425074 0.09977164 0.32122597 0.32238323 0.33816330
      Oct      Nov      Dec
2 0.31179502 0.32699801 0.32965191
> bpnt.fcst<- forecast(bpnt.mlp, 3)
> bpnt.fcst
      Jan      Feb      Mar
3 0.4837302 0.3138424 0.3888703
> accuracy(bpnt.fcst)
Training set -2.658667e-05  0.001119263  0.0008706646 -0.02258275  0.256797  NaN  0.06833841

```



Lampiran 3. Langkah-langkah Extreme Learning Machine di R Studio

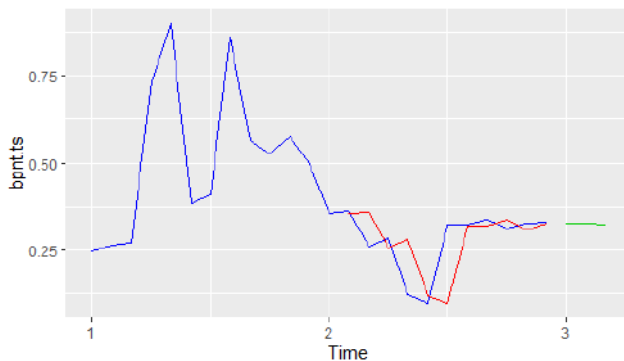
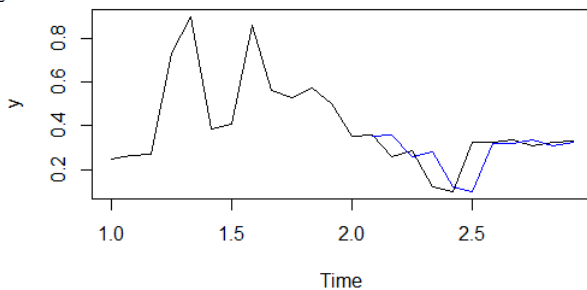
```
# Package
library(nnfor)
library(forecast)
bpnt <- read_excel("jurnal skripsi khoirul/bpnt.xlsx")
View(bpnt)
bpnt.ts=ts(bpnt$bpnt,frequency = 12)
bpnt.ts
plot(bpnt.ts)
# DATA TRAINING90% #
bpnt.train=ts(bpnt.ts[1:19],frequency = 12)
bpnt.train
#DATA TESTING 10% #
bpnt.test=ts(bpnt.ts[20:24],start=c(2,8), freq=12)
bpnt.test
#ELM untuk Peramalan Data
bpnt.elm<-elm (bpnt.train,hd=9, reps=20, lags=NULL ,
difforder=NULL ,outplot=c(TRUE),sel.lag=c(FALSE))
bpnt.elm
plot(bpnt.elm)
bpnt.elm$MSE
bpnt.elm$fitted
frc<-forecast(bpnt.elm,3)
frc
accuracy(frc)
```

```
autoplot(bpnt.ts,color='blue')+autolayer(bpnt.mlp$fitted,color='red')+
autolayer(frc,color='green')
```

Output Pengujian

1. Extreme Learning Machine 12-2-1

```
> bpnt.e1m$MSE
[1] 0.007974592
> bpnt.e1m$fitted
      Feb      Mar      Apr      May      Jun      Jul      Aug      Sep
2 0.35270389 0.35786828 0.25878406 0.28204383 0.12110703 0.09768712 0.32061662 0.32075673
      Oct      Nov      Dec
2 0.33622989 0.30892668 0.32496031
> frc<-forecast(bpnt.e1m,3)
> frc
      Jan      Feb      Mar
3 0.3272623 0.3249494 0.3226365
> accuracy(frc)
      ME      RMSE      MAE      MPE      MAPE      MASE      ACF1
Training set 5.046622e-18 0.08930057 0.05481684 1.159323 13.02096 NaN -0.1012646
```

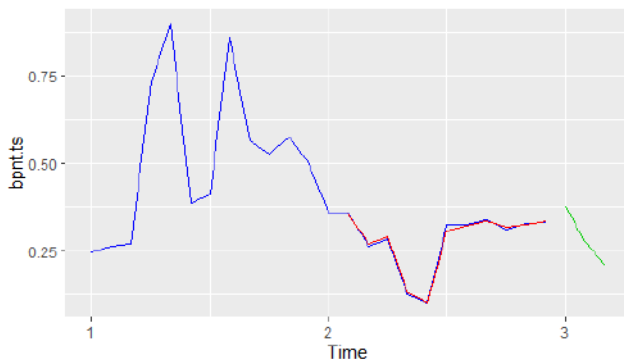
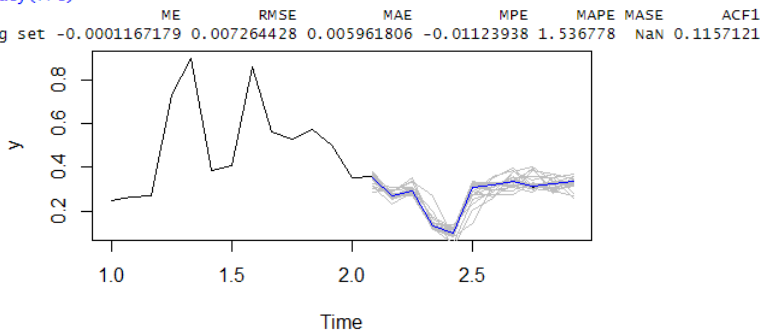


2. Extreme Learning Machine 12-9-1

```

> bpnt.e1m$MSE
[1] 5.277192e-05
> bpnt.e1m$fitted
      Feb      Mar      Apr      May      Jun      Jul      Aug      Sep      Oct
2 0.3544998 0.2694123 0.2909636 0.1323906 0.0994102 0.3062235 0.3205781 0.3358367 0.3165977
      Nov      Dec
2 0.3232998 0.3337559
> frc<-forecast(bpnt.e1m,3)
> frc
      Jan      Feb      Mar
3 0.3742835 0.2763407 0.2084028
> accuracy(frc)
Training set -0.0001167179  0.007264428  0.005961806  -0.01123938  1.536778  NaN  0.1157121

```

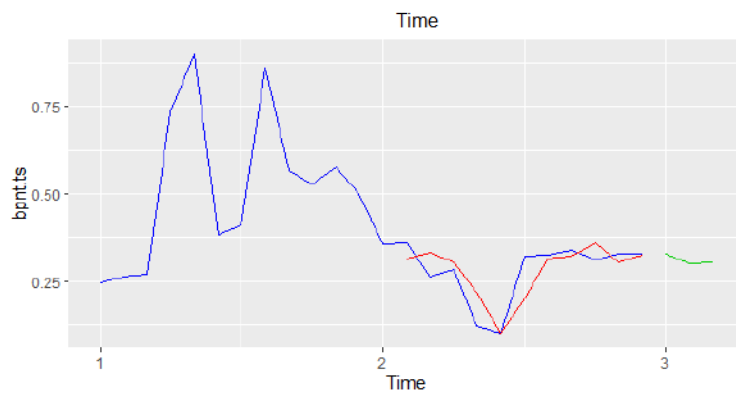
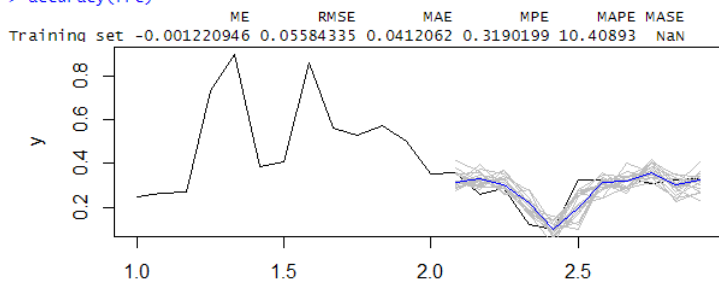


3. Extreme Learning Machine 12-16-1

```

> bpnt.e1m$MSE
[1] 0.00311848
> bpnt.e1m$fitted
  2 0.3158616 0.3299287 0.3054496 0.2185054 0.1008046 0.2006353 0.3140930 0.3207567
  2 0.3587742 0.3053455 0.3249603
> frc<-forecast(bpnt.e1m,3)
> frc
  Jan      Feb      Mar
3 0.3272623 0.3022163 0.3044757
> accuracy(frc)

```

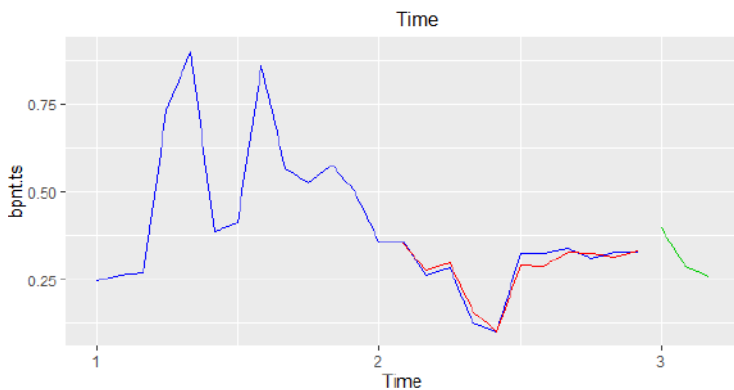
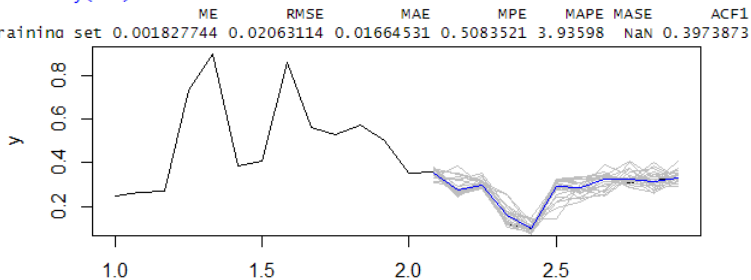


4. Extreme Learning Machine 12-23-1

```

> bpnt.elm$MSE
[1] 0.0004256439
> bpnt.elm$fitted
      Feb      Mar      Apr      May      Jun      Jul      Aug
2 0.35235551 0.27678979 0.29813886 0.15865731 0.09975119 0.29067595 0.28507504
      Sep      Oct      Nov      Dec
2 0.32734727 0.32446917 0.31518942 0.33312973
> frc<-forecast(bpnt.elm,3)
> frc
      Jan      Feb      Mar
3 0.3958815 0.2871045 0.2601221
> accuracy(frc)
      ME      RMSE      MAE      MPE      MAPE      MASE      ACF1
Training set 0.001827744 0.02063114 0.01664531 0.5083521 3.93598 NaN 0.3973873

```



Lampiran 4. Perhitungan Manual *Backpropagation*

Berikut merupakan perhitungan manual metode backpropagation untuk model arsitektur 4-9-1:

a. Inisialisasi bobot

- 1) Menentukan nilai bobot pada setiap unit di lapisan input ke setiap unit di lapisan tersembunyi (v_{ij}) dan nilai bias pada setiap unit di lapisan input ke setiap unit di lapisan tersembunyi (v_{0j}).

$$v_{11} = 0, 1$$

$$v_{21} = 0, 2$$

$$v_{31} = 0, 3$$

$$v_{41} = 0, 4$$

$$v_{12} = 0, 5$$

$$v_{22} = -0, 1$$

$$v_{32} = -0, 2$$

$$v_{42} = -0, 3$$

$$v_{13} = -0, 4$$

$$v_{23} = -0, 5$$

$$v_{33} = 0, 1$$

$$v_{43} = 0, 2$$

$$v_{14} = 0, 3$$

$$v_{24} = 0, 4$$

$$v_{34} = 0, 5$$

$$v_{44} = -0, 1$$

$$v_{15} = -0, 2$$

$$v_{25} = -0, 3$$

$$v_{35} = -0, 4$$

$$v_{45} = -0, 5$$

$$v_{16} = 0, 1$$

$$v_{26} = 0, 2$$

$$v_{36} = 0, 3$$

$$v_{46} = 0, 4$$

$$v_{17} = 0, 5$$

$$v_{27} = -0, 1$$

$$v_{37} = -0, 2$$

$$v_{47} = -0, 3$$

$$v_{18} = -0, 4$$

$$v_{28} = -0, 5$$

$$v_{38} = 0, 1$$

$$v_{48} = 0, 2$$

$$v_{19} = 0, 3$$

$$v_{29} = 0, 4$$

$$v_{39} = 0, 5$$

$$v_{49} = -0, 1$$

$$v_{01} = 0, 2$$

$$v_{02} = 0, 3$$

$$v_{03} = 0, 4$$

$$v_{04} = 0, 5$$

$$v_{05} = 0, 1$$

$$v_{06} = 0, 2$$

$$v_{07} = 0, 3$$

$$v_{08} = 0, 4$$

$$v_{09} = 0,5$$

- 2) Menentukan nilai bobot pada setiap unit di lapisan tersembunyi ke setiap unit di lapisan output (w_{jk}) dan nilai bias pada setiap unit di lapisan tersembunyi ke setiap unit di lapisan output (w_{0k}).

$$w_{11} = 0,1$$

$$w_{21} = 0,2$$

$$w_{31} = 0,3$$

$$w_{41} = 0,4$$

$$w_{51} = 0,5$$

$$w_{61} = -0,1$$

$$w_{71} = -0,2$$

$$w_{81} = -0,3$$

$$w_{91} = -0,4$$

$$w_{01} = 0,5$$

- b. ikuti langkah c sampai j selama kondisi berhenti belum terpenuhi.
- c. Ikuti langkah d sampai i untuk setiap pasangan data training.

Tahap 1 Propagasi Maju

- d. Setiap unit *input* ($X_i, i = 1, 2, 3, \dots, n$) menerima dan mengirimkan sinyal *input* ke setiap unit di lapisan tersembunyi.

Diketahui: $x_1 = 0,2460642$; $x_2 = 0,2628985$; $x_3 = 0,2697643$; $x_4 = 0,7324576$

- e. Setiap unit di lapisan tersembunyi ($Z_j, j = 1, 2, 3, \dots, p$) menjumlahkan sinyal-sinyal *input* terbobot.

$$Z_{in j} = v_{0j} + \sum_{i=1}^n x_i v_{ij}$$

$$\begin{aligned} Z_{in1} &= v_{01} + \sum_{i=1}^4 x_i v_{i1} = \\ &0, 2 + (0, 2460642 * 0, 1) + (0, 2628985 * 0, 2) + (0, 2697643 * \\ &0, 3) + (0, 7324576 * 0, 4) = 0, 65109845 \end{aligned}$$

$$\begin{aligned} Z_{in2} &= v_{02} + \sum_{i=1}^4 x_i v_{i2} = \\ &0, 3 + (0, 2460642 * 0, 5) + (0, 2628985 * -0, 1) + \\ &(0, 2697643 * -0, 2) + (0, 7324576 * -0, 3) = 0, 12305211 \end{aligned}$$

$$\begin{aligned} Z_{in3} &= v_{03} + \sum_{i=1}^4 x_i v_{i3} = \\ &0, 4 + (0, 2460642 * -0, 0, 4) + (0, 2628985 * -0, 5) + \\ &(0, 2697643 * 0, 0, 1) + (0, 7324576 * 0, 2) = 0, 34359302 \end{aligned}$$

$$\begin{aligned} Z_{in4} &= v_{04} + \sum_{i=1}^4 x_i v_{i4} = \\ &0, 5 + (0, 2460642 * 0, 3) + (0, 2628985 * 0, 4) + (0, 2697643 * \\ &0, 5) + (0, 7324576 * -0, 0, 1) = 0, 74061505 \end{aligned}$$

$$\begin{aligned} Z_{in5} &= v_{05} + \sum_{i=1}^4 x_i v_{i5} = \\ &0, 1 + (0, 0, 2460642 * -0, 2) + (0, 2628985 * -0, 3) + \\ &(0, 2697643 * -0, 0, 4) + (0, 7324576 * -0, 5) = -0, 50221691 \end{aligned}$$

$$\begin{aligned} Z_{in6} &= v_{06} + \sum_{i=1}^4 x_i v_{i6} = \\ &0, 2 + (0, 2460642 * 0, 1) + (0, 0, 2628985 * 0, 2) + \\ &(0, 2697643 * 0, 3) + (0, 7324576 * 0, 4) = 0, 65109845 \end{aligned}$$

$$\begin{aligned} Z_{in7} &= v_{07} + \sum_{i=1}^4 x_i v_{i7} = \\ &0, 3 + (0, 2460642 * 0, 5) + (0, 2628985 * -0, 1) + \\ &(0, 2697643 * -0, 2) + (0, 7324576 * -0, 3) = 0, 12305211 \end{aligned}$$

$$Z_{in8} = v_{08} + \sum_{i=1}^4 x_i v_{i8} =$$

$$0,4 + (0,0,2460642 * -0,4) + (0,2628985 * -0,0,5) + (0,2697643 * 0,1) + (0,7324576 * 0,2) = 0,34359302$$

$$Z_{in9} = v_{09} + \sum_{i=1}^4 x_i v_{i9} =$$

$$0,5 + (0,2460642 * 0,3) + (0,2628985 * 0,4) + (0,2697643 * 0,0,0,5) + (0,7324576 * -0,0,1) = 0,74061505$$

Menghitung sinyal *output* dengan fungsi aktivasi.

$$Z_j = f(Z_{in_j}) = \frac{1}{1 + e^{-Z_{in_j}}}$$

$$Z_1 = \frac{1}{1+e^{-Z_{in1}}} = \frac{1}{1+e^{-0,65109845}} = 0,0,6572579532$$

$$Z_2 = \frac{1}{1+e^{-Z_{in2}}} = \frac{1}{1+e^{-0,12305211}} = 0,5307242688$$

$$Z_3 = \frac{1}{1+e^{-Z_{in3}}} = \frac{1}{1+e^{-0,34359302}} = 0,5850630456$$

$$Z_4 = \frac{1}{1+e^{-Z_{in4}}} = \frac{1}{1+e^{-0,74061505}} = 0,6771303361$$

$$Z_5 = \frac{1}{1+e^{-Z_{in5}}} = \frac{1}{1+e^{-0,50221691}} = 0,3770198283$$

$$Z_6 = \frac{1}{1+e^{-Z_{in6}}} = \frac{1}{1+e^{-0,65109845}} = 0,6572579532$$

$$Z_7 = \frac{1}{1+e^{-Z_{in7}}} = \frac{1}{1+e^{-0,12305211}} = 0,5307242688$$

$$Z_8 = \frac{1}{1+e^{-Z_{in8}}} = \frac{1}{1+e^{-0,34359302}} = 0,5850630456$$

$$Z_9 = \frac{1}{1+e^{-Z_{in9}}} = \frac{1}{1+e^{-0,74061505}} = 0,6771303361$$

- f. Setiap unit di lapisan *output* ($Y_k, k = 1, 2, 3, \dots, m$) menjumlahkan sinyal-sinyal tersembunyi terbobot.

$$Y_{ink} = w_{0k} + \sum_{j=1}^m Z_j w_{jk}$$

$$Y_{in1} = w_{01} + \sum_{j=1}^m Z_j w_{j1} = 0,5 + (0,0,6572579532 * 0,1) + (0,5307242688 * 0,2) + (0,5850630456 * 0,0,3) + (0,0,6771303361 * 0,4) + (0,3770198283 * 0,5) +$$

$$(0,6572579532 * -0,1) + (0,5307242688 * -0,2) + (0,5850630456 * -0,3) + (0,6771303361 * -0,4) = 0,6885099142$$

Menghitung sinyal *output* dengan fungsi aktivasi.

$$Y_k = f(Y_{ink}) = \frac{1}{1 + e^{-Y_{ink}}}$$

$$Y_k = \frac{1}{1 + e^{-Y_{ink}}} = \frac{1}{1 + e^{-0,6885099142}} = 0,66563536$$

Tahap 2 Propagasi Mundur

- g. Setiap unit *output* ($Y_k, k = 1, 2, 3, \dots, m$) mendapatkan pola target yang sesuai dengan pola *input* pelatihan dan menghitung informasi *error*-nya:

$$\delta_k = (t_k - y_k) f'(y_{ink}) = (t_k - Y_k) Y_k (1 - Y_k)$$

Diketahui $t_k = 0,9$

$$\delta_k = (t_k - Y_k) Y_k (1 - Y_k) = (0,9 - 0,66563536) * 0,66563536 * (1 - 0,66563536) = 0,052161349$$

Kemudian hitung koreksi bobot untuk memperbaiki nilai w_{jk} :

$$\Delta w_{jk} = \alpha \delta_k z_j$$

$$\Delta w_{11} = \alpha \delta_k z_1 = 0,01 * 0,052161349 * 0,6572579532 = 0,0003428346148$$

$$\Delta w_{21} = \alpha \delta_k z_2 = 0,01 * 0,052161349 * 0,5307242688 = 0,0002768329381$$

$$\Delta w_{31} = \alpha \delta_k z_3 = 0,01 * 0,052161349 * 0,5850630456 = 0,0003051767771$$

$$\Delta w_{41} = \alpha \delta_k z_4 = 0,01 * 0,052161349 * 0,6771303361 = 0,0003532003178$$

$$\Delta w_{51} = \alpha \delta_k z_5 = 0,01 * 0,052161349 * 0,3770198283 = 0,0001966586284$$

$$\Delta w_{61} = \alpha \delta_k z_6 = 0,01 * 0,052161349 * 0,6572579532 = 0,0003428346148$$

$$\Delta w_{71} = \alpha \delta_k z_7 = 0,01 * 0,052161349 * 0,5307242688 = 0,0002768329381$$

$$\Delta w_{81} = \alpha \delta_k z_8 = 0,01 * 0,052161349 * 0,5850630456 = 0,0003051767771$$

$$\Delta w_{91} = \alpha \delta_k z_9 = 0,01 * 0,052161349 * 0,6771303361 = 0,0003532003178$$

Hitung juga koreksi bias untuk memperbaiki nilai w_{0k}

$$\Delta w_{0k} = \alpha \delta_k$$

$$\Delta w_{01} = \alpha \delta_k = 0,01 * 0,052161349 = 0,00052161349$$

- h. Setiap unit tersembunyi ($Z_j, j = 1, 2, 3, \dots, p$) menjumlahkan delta *inputnya*.

$$\delta in_j = \sum_{k=1}^m \delta_k w_{jk}$$

$$\delta in_1 = \sum_{k=1}^m \delta_k w_{11} = 0,052161349 * 0,1 = 0,0052161349$$

$$\delta in_2 = \sum_{k=1}^m \delta_k w_{21} = 0,052161349 * 0,2 = 0,0104322698$$

$$\delta in_3 = \sum_{k=1}^m \delta_k w_{31} = 0,052161349 * 0,3 = 0,0156484047$$

$$\delta in_4 = \sum_{k=1}^m \delta_k w_{41} = 0,052161349 * 0,4 = 0,0208645396$$

$$\delta_{in_5} = \sum_{k=1}^m \delta_k w_{51} = 0,052161349 * 0,5 = 0,0260806745$$

$$\delta_{in_6} = \sum_{k=1}^m \delta_k w_{61} = 0,052161349 * -0,1 = -0,0052161349$$

$$\delta_{in_7} = \sum_{k=1}^m \delta_k w_{71} = 0,052161349 * -0,2 = -0,0104322698$$

$$\delta_{in_8} = \sum_{k=1}^m \delta_k w_{81} = 0,052161349 * -0,3 = -0,0156484047$$

$$\delta_{in_9} = \sum_{k=1}^m \delta_k w_{91} = 0,052161349 * -0,4 = -0,0208645396$$

Untuk menghitung informasi *error*, kalikan nilai dengan fungsi aktivasinya.

$$\delta_j = \delta_{in_j} f'(z_{in_j}) = \delta_{in_j} Z_j (1 - Z_j)$$

$$\delta_1 = \delta_{in_1} Z_1 (1 - Z_1) = 0,001175038376$$

$$\delta_2 = \delta_{in_2} Z_2 (1 - Z_2) = 0,002598219589$$

$$\delta_3 = \delta_{in_3} Z_3 (1 - Z_3) = 0,003798873673$$

$$\delta_4 = \delta_{in_4} Z_4 (1 - Z_4) = 0,004561506716$$

$$\delta_5 = \delta_{in_5} Z_5 (1 - Z_5) = 0,006125721306$$

$$\delta_6 = \delta_{in_6} Z_6 (1 - Z_6) = -0,001175038376$$

$$\delta_7 = \delta_{in_7} Z_7 (1 - Z_7) = -0,002598219589$$

$$\delta_8 = \delta_{in_8} Z_8 (1 - Z_8) = -0,003798873673$$

$$\delta_9 = \delta_{in_9} Z_9 (1 - Z_9) = -0,004561506716$$

Menghitung koreksi bobot untuk memperbaiki nilai v_{ij}

$$\Delta v_{ij} = \alpha \delta_j x_i$$

$$\begin{aligned}
\Delta v_{11} &= \alpha \delta_1 x_1 = 0, 00000289134878 \\
\Delta v_{12} &= \alpha \delta_2 x_1 = 0, 000006393288245 \\
\Delta v_{13} &= \alpha \delta_3 x_1 = 0, 000009347668113 \\
\Delta v_{14} &= \alpha \delta_4 x_1 = 0, 00001122423501 \\
\Delta v_{15} &= \alpha \delta_5 x_1 = 0, 00001507320713 \\
\Delta v_{16} &= \alpha \delta_6 x_1 = -0, 00000289134878 \\
\Delta v_{17} &= \alpha \delta_7 x_1 = -0, 000006393288245 \\
\Delta v_{18} &= \alpha \delta_8 x_1 = -0, 000009347668113 \\
\Delta v_{19} &= \alpha \delta_9 x_1 = -0, 00001122423501 \\
\Delta v_{21} &= \alpha \delta_1 x_2 = 0, 000003089158265 \\
\Delta v_{22} &= \alpha \delta_2 x_2 = 0, 000006830680325 \\
\Delta v_{23} &= \alpha \delta_3 x_2 = 0, 000009987181904 \\
\Delta v_{24} &= \alpha \delta_4 x_2 = 0, 00001199213273 \\
\Delta v_{25} &= \alpha \delta_5 x_2 = 0, 00001610442943 \\
\Delta v_{26} &= \alpha \delta_6 x_2 = -0, 000003089158265 \\
\Delta v_{27} &= \alpha \delta_7 x_2 = -0, 000006830680325 \\
\Delta v_{28} &= \alpha \delta_8 x_2 = -0, 000009987181904 \\
\Delta v_{29} &= \alpha \delta_9 x_2 = -0, 00001199213273 \\
\Delta v_{31} &= \alpha \delta_1 x_3 = 0, 00000316983405 \\
\Delta v_{32} &= \alpha \delta_2 x_3 = 0, 000007009068886 \\
\Delta v_{33} &= \alpha \delta_3 x_3 = 0, 00001024800497 \\
\Delta v_{34} &= \alpha \delta_4 x_3 = 0, 00001230531666 \\
\Delta v_{35} &= \alpha \delta_5 x_3 = 0, 0000165250092
\end{aligned}$$

$$\Delta v_{36} = \alpha \delta_6 x_3 = -0,00000316983405$$

$$\Delta v_{37} = \alpha \delta_7 x_3 = -0,000007009068886$$

$$\Delta v_{38} = \alpha \delta_8 x_3 = -0,00001024800497$$

$$\Delta v_{39} = \alpha \delta_9 x_3 = -0,00001230531666$$

$$\Delta v_{41} = \alpha \delta_1 x_4 = 0,000008606657888$$

$$\Delta v_{42} = \alpha \delta_2 x_4 = 0,00001903085684$$

$$\Delta v_{43} = \alpha \delta_3 x_4 = 0,00002782513893$$

$$\Delta v_{44} = \alpha \delta_4 x_4 = 0,00003341110262$$

$$\Delta v_{45} = \alpha \delta_5 x_4 = 0,00004486831126$$

$$\Delta v_{46} = \alpha \delta_6 x_4 = -0,000008606657888$$

$$\Delta v_{47} = \alpha \delta_7 x_4 = -0,00001903085684$$

$$\Delta v_{48} = \alpha \delta_8 x_4 = -0,00002782513893$$

$$\Delta v_{49} = \alpha \delta_9 x_4 = -0,00003341110262$$

Hitung juga koreksi bias untuk memperbaiki nilai v_{0j}

$$\Delta v_{0j} = \alpha \delta_j$$

$$\Delta v_{01} = \alpha \delta_1 = 0,00001175038376$$

$$\Delta v_{02} = \alpha \delta_2 = 0,00002598219589$$

$$\Delta v_{03} = \alpha \delta_3 = 0,00003798873673$$

$$\Delta v_{04} = \alpha \delta_4 = 0,00004561506716$$

$$\Delta v_{05} = \alpha \delta_5 = 0,00006125721306$$

$$\Delta v_{06} = \alpha \delta_6 = -0,00001175038376$$

$$\Delta v_{07} = \alpha \delta_7 = -0,00002598219589$$

$$\Delta v_{08} = \alpha \delta_8 = -0,00003798873673$$

$$\Delta v_{09} = \alpha \delta_9 = -0,00004561506716$$

Tahap 3 Perubahan Bobot

- i. Setiap unit *output* ($Y_k, k = 1, \dots, m$) memperbaiki bobot dan biasnya.

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

$$w_{11}(\text{baru}) = w_{11}(\text{lama}) + \Delta w_{11} = 0,1003428346$$

$$w_{21}(\text{baru}) = w_{21}(\text{lama}) + \Delta w_{21} = 0,2002768329$$

$$w_{31}(\text{baru}) = w_{31}(\text{lama}) + \Delta w_{31} = 0,0,3003051768$$

$$w_{41}(\text{baru}) = w_{41}(\text{lama}) + \Delta w_{41} = 0,4003532003$$

$$w_{51}(\text{baru}) = w_{51}(\text{lama}) + \Delta w_{51} = 0,5001966586$$

$$w_{61}(\text{baru}) = w_{61}(\text{lama}) + \Delta w_{61} = -0,09965716539$$

$$w_{71}(\text{baru}) = w_{71}(\text{lama}) + \Delta w_{71} = -0,1997231671$$

$$w_{81}(\text{baru}) = w_{81}(\text{lama}) + \Delta w_{81} = -0,2996948232$$

$$w_{91}(\text{baru}) = w_{91}(\text{lama}) + \Delta w_{91} = -0,3996467997$$

$$w_{01}(\text{baru}) = w_{01}(\text{lama}) + \Delta w_{01} = 0,5005216135$$

- Setiap unit tersembunyi ($Z_j, j = 1, \dots, p$) memperbaiki bobot dan biasnya.

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

Tahapan di atas dikatakan *epoch* 1 pada data jumlah penerima BPNT yang mana proses ini belum mencapai error yang diinginkan maka dari itu pelatihan jaringan harus dilakukan pengulangan dimulai dari langkah 2 sampai error yang dihasilkan sesuai dengan yang

diharapkan. Perulangan kedua menggunakan bobot baru yang dihasilkan pada *epoch* 1.

Lampiran 5. Simulasi Metode *Extreme Learning Machine*

Training

No	X1	X2	X3	X4	Y
1	0,2460642	0,2628985	0,2697643	0,7324576	0,9000000
2	0,2628985	0,2697643	0,7324576	0,9000000	0,3848421
3	0,2697643	0,7324576	0,9000000	0,3848421	0,4101236
4	0,7324576	0,9000000	0,3848421	0,4101236	0,8620878
5	0,9000000	0,3848421	0,4101236	0,8620878	0,5643747

Testing

6	0,3848421	0,0,4101236	0,8620878	0,5643747	0,5272031
7	0,4101236	0,8620878	0,0,5643747	0,5272031	0,5743832
8	0,8620878	0,5643747	0,5272031	0,5743832	0,5004003
9	0,5643747	0,5272031	0,5743832	0,5004003	0,3550168

Training

- 1) Siapkan matriks data latih dengan feature sejumlah d

$$X = \begin{bmatrix} 0,2460642 & 0,2628985 & 0,2697643 & 0,7324576 \\ 0,2628985 & 0,2697643 & 0,7324576 & 0,9000000 \\ 0,0,2697643 & 0,7324576 & 0,9000000 & 0,3848421 \\ 0,7324576 & 0,9000000 & 0,0,3848421 & 0,4101236 \\ 0,9000000 & 0,3848421 & 0,4101236 & 0,8620878 \end{bmatrix}$$

- 2) Siapkan target data latih

$$Y = \begin{bmatrix} 0,9000000 & 0,3848421 & 0,4101236 & 0,8620878 & 0,8620878 \end{bmatrix}$$

- 3) Tentukan jumlah *neuron h* pada *hidden layer*

$$h = 4$$

4) Inisialisasi nilai bobot W dengan ukuran hxd

$$W = \begin{bmatrix} 0,1 & 0,2 & 0,3 & 0,4 \\ -0,3 & -0,1 & 0,2 & 0,4 \\ 0,1 & 0,1 & -0,2 & 0,4 \\ 0,3 & 0,2 & -0,1 & 0,1 \end{bmatrix}$$

5) Menghitung keluaran *hidden layer* (H_{init}), persamaan berikut untuk menghitung *hidden layer*:

$$H_{init} = X_{train} \cdot W^T$$

$$H_{init} =$$

$$\begin{bmatrix} 0,19245058 & 0,0,19639094 & -0,00079966 & 0,38473656 \\ 0,28860632 & 0,27884903 & -0,10366911 & 0,59604816 \\ 0,01269178 & 0,14767552 & 0,0089366 & 0,79937297 \\ -0,03523295 & 0,17700045 & 0,2817565 & 0,84793224 \\ 0,27418607 & 0,0,35494571 & 0,17873492 & 0,76419506 \end{bmatrix}$$

6) Menghitung matriks *output* dari *hidden layer*

$$H = \frac{1}{1 + e^{-H_{init}train}}$$

$$H =$$

$$\begin{bmatrix} 0,5479646964 & 0,5489405354 & 0,499800085 & 0,5950149938 \\ 0,5716549028 & 0,5692640258 & 0,4741059093 & 0,644751668 \\ 0,5031729024 & 0,5479646964 & 0,5022341351 & 0,5479646964 \\ 0,4911926736 & 0,5441349467 & 0,5699768006 & 0,0,7001332028 \\ 0,568120289 & 0,5878163847 & 0,5445651526 & 0,682263833 \end{bmatrix}$$

7) Menghitung *output weight*

$$H^+ = (H^T H)^{-1} H^T$$

$$\begin{bmatrix} 31,736361 & -10,723700 & -17,851485 & -5,922893 & 2,8717783 \\ -40,103473 & 19,425154 & 27,754628 & -1,401059 & -4,2356938 \\ 20,961220 & -18,535281 & -4,645634 & 2,593436 & 0,3053167 \\ -8,272069 & 7,384822 & -5,283128 & 4,438982 & 1,3890779 \end{bmatrix}$$

$$\hat{\mathbf{B}} = H^+Y$$

$$\begin{bmatrix} 14,484150 \\ -22,094057 \\ 12,325636 \\ -1,745308 \end{bmatrix}$$

Testing

- 1) Menghitung keluaran *hidden layer* (H_{init}), persamaan berikut untuk menghitung *hidden layer*:

$$H_{init} = X_{test} \cdot W^T$$

$$\begin{bmatrix} 0,6048851 & 0,24170245 & 0,1328289 & 0,1677060 \\ 0,5936236 & 0,11451032 & 0,2252274 & 0,2917375 \\ 0,5869979 & 0,02013009 & 0,2669589 & 0,3762193 \\ 0,5343532 & 0,09300404 & 0,1944413 & 0,2673547 \end{bmatrix}$$

- 2) Menghitung matriks *output* dari *hidden layer*

$$H = \frac{1}{1 + e^{-H_{init} \text{train}}}$$

$$H = \begin{bmatrix} 0,6467731432 & 0,5601331486 & 0,5331584866 & 0,5418285093 \\ 0,6441961352 & 0,5285963391 & 0,5560700263 & 0,5724214484 \\ 0,6426760285 & 0,5050323526 & 0,5663461676 & 0,5929609214 \\ 0,6304978532 & 0,5232342649 & 0,5484577492 & 0,5664433728 \end{bmatrix}$$

3) Hitung nilai output

$$\hat{\mathbf{Y}} = H.\hat{\mathbf{B}}$$

$$\begin{bmatrix} 2,618205 \\ 3,506661 \\ 4,096080 \\ 3,343330 \end{bmatrix}$$

Nilai output di atas merupakan hasil prediksi jumlah penerima BPNT di Kota Semarang

Lampiran 10. Daftar Riwayat Hidup**DAFTAR RIWAYAT HIDUP****A. Identitas Diri**

Nama Lengkap : Khoirul Anwar
Tempat, Tanggal Lahir : Kab. Semarang, 12 Juli 2002
Alamat : Dsn. Kropoh RT 06/RW 05,
Ds. Gogodalem, Kec. Bringin,
Kab. Semarang
Nomor HP : 0812-2582-8170
Email : dungalfian@gmail.com

B. Riwayat Pendidikan

- | | |
|--------------------------|------------------|
| 1. SD Negeri Wiru 03 | Lulus tahun 2014 |
| 2. SMP Negeri 01 Bringin | Lulus tahun 2017 |
| 3. SMA Negeri 01 Bringin | Lulus tahun 2020 |

Semarang, 16 September 2024



Khoirul Anwar

NIM. 2008046034