

Hasil_Cek_03

by victor.siregar11 victor.siregar11

Submission date: 22-Oct-2021 06:14PM (UTC+0900)

Submission ID: 1656356957

File name: Paper-UAS-Sahat-Muklis-03.docx (1.05M)

Word count: 7938

Character count: 53676



Analisis Optimasi Fungsi Pelatihan Machine Learning dalam Peramalan Kemiskinan

Sahat Sonang S^{*1}, Sarjon Defit^{#2}, Mukhlis Ramadhan^{#3}

²Universitas Putra Indonesia YPTK Padang, Indonesia

³STMIK TRIGUNA DHARMA, Medan, Indonesia

²sarjond@yahoo.co.uk

³mukhlis.ramadhan99@gmail.com

^{*1}Politeknik Bisnis Indonesia, Pematangsiantar, Indonesia

¹sahatsonangstg@gmail.com

Abstrak— Banyak metode fungsi pelatihan dalam Machine Learning Neural Network yang digunakan dalam menyelesaikan masalah komputasi yang berkaitan dengan prediksi. Fungsi pelatihan yang digunakan pada Machine Learning metoda algoritma backpropagation dapat menghasilkan prediksi yang berbeda, yang dipengaruhi oleh parameter dan data yang digunakan. Tujuan dari penelitian dilakukan untuk menganalisa performance dan keakuratan algoritma backpropagation standard serta mengoptimalkan fungsi pelatihan dengan algoritma Bayesian Regulation, dan One Step Secant. Dalam proses analisis, penelitian ini menggunakan Dataset jumlah kemiskinan di Indonesia dalam jangka waktu 12 tahun (tahun 2009 - 2020) yang terdiri dari 34 provinsi. Data diperoleh dari website Badan Pusat Statistik (BPS) Indonesia <https://www.bps.go.id/>. Berdasarkan pelatihan, pengujian, dan analisa yang dilakukan diperoleh hasil dari penelitian, bahwa model jaringan 5-9-1 menggunakan fungsi pelatihan Bayesian Regulation mampu melakukan optimasi yang lebih baik dengan percepatan waktu pelatihan, MSE Pengujian, Performance lebih rendah dibandingkan dengan 2 metode yang lain, dengan demikian disimpulkan bahwa model jaringan 5-9-1 menggunakan algoritma Bayesian Regulation dapat digunakan untuk prediksi kemiskinan di Indonesia.

Kata kunci— Optimasi, Machine Learning, Backpropagation, Bayesian Regulation, One Step Secant, Kemiskinan

Abstract— Many methods of training functions in Machine Learning Neural Networks are used in solving computational problems related to prediction. The training function used in Machine Learning backpropagation algorithm method can produce different predictions, which is influenced by the parameters and data used. The purpose of this research is to analyze the performance and accuracy of the standard backpropagation algorithm and optimize the training function with the Bayesian Regulation algorithm, and One Step Secant. In the analysis process, This study uses a dataset on the number of poverty in Indonesia for a period of 12 years (2009 - 2020) consisting of 34 provinces. Data obtained from the website Indonesian Statistical Center (BPS) <https://www.bps.go.id/>. Based on the training, testing, and

analysis carried out, the results obtained from the research, that the 5-9-1 network model using the Bayesian Regulation training function is able to perform better optimization by accelerating the training time, MSE Testing, Performance is lower than the other 2 methods, thus it is concluded that the 5-9-1 network model using the Bayesian Regulation algorithm can be used to predict poverty in Indonesia.

Keywords— Optimization, Machine Learning, Backpropagation, Bayesian Regulation, One Step Secant, Poverty

I. PENDAHULUAN

Saat ini Machine Learning sangat banyak diperbincangkan dalam perkembangan teknologi informasi secara khusus dalam bidang penelitian yang diterapkan di berbagai bidang seperti teknologi, perbankan, bisnis, kesehatan, dan lainnya [1]. Machine Learning mampu menyelesaikan masalah yang kompleks [2]–[4], seperti menganalisa gambar, menganalisa data times-series, dan melacak objek [5]. Teknik Machine Learning bertujuan agar komputer dapat belajar secara otomatis tanpa adanya campur tangan manusia [6], serta menyesuaikan tindakan yang tepat [7]. Fokus penelitian Machine Learning tentang bagaimana meningkatkan kinerja sistem pembelajaran otomatis melalui pelatihan-pelatihan atau pengalaman [8]. permasalahan Machine Learning digunakan sebagai pilihan dalam mengoptimalkan cara kerja komputer sesuai dengan data lalu [9]. Penelitian ini membahas teknik Machine Learning dengan metode jaringan syaraf Backpropagation. Metode backpropagation pada Machine Learning banyak dipergunakan dalam pelatihan jaringan syaraf feedforward [10]–[12].

Algoritma Backpropagation memiliki beberapa fungsi, yaitu: fungsi transfer dan fungsi pelatihan dimana setiap fungsi memiliki beberapa tehnik dan cara untuk menyelesaikan permasalahan yang kompleks. fungsi transfer banyak digunakan pada Machine Learning neural network secara khusus algoritma propagasi balik seperti fungsi sigmoid-tangen-hiperbolik (tansig) [13], log-

sigmoid (logsig) [14], linear-purelin [15]. Metode Backpropagation standard umumnya menggunakan fungsi pelatihan gradient descent yaitu: traingd, trainngdx, traingda, dan traingdm [16]. selain itu masih banyak fungsi pelatihan yang digunakan untuk optimasi yang dapat mempengaruhi hasil komputasi, seperti halnya gradien konjugasi berskala (traincsg) [17], pelatihan urutan acak dengan fungsi pembelajaran (trainr) [18], regulasi bayesian (trainbr) [19], pelatihan bobot atau bias perintah acak tidak terawasi (trainru) [20], Levenberg-Marquardt (trainlm) [21], [22], OSS (trainoss) [23], pelatihan inkremental berurutan fungsi pembelajaran (trains) [24], pelatihan batch bobot atau bias tanpa pengawasan (trainbu) [25], pelatihan batch menggunakan aturan pembelajaran bobot dan bias (trainb) [26], pelatihan konjugasi gradien (traingcgp, traingcb, traingcf) [27], resilient (trainrp) [28], BFGS quasi-newton (trainbfg) [29], urutan siklus bobot atau bias (trainc) [30], dan BFGS quasi-newton referensi adaftif kontrol (trainbfgc) [31]. Pemanfaatan fungsi transfer maupun fungsi pelatihan menghasilkan peramalan yang tingkat keakuratan berbeda-beda, yang dipengaruhi oleh paramete atau metode yang diberikan serta data yang akan di uji [32]–[36]. Sesuai dengan hal tersebut, penelitian ini fokus membahas pada penggunaan fungsi pelatihan dalam mengoptimalkan kemampuan algoritma backpropagation standard.

Dalam penyelesaian masalah yang banyak dan kompleks dilakukan pelatihan-pelatihan untuk meningkatkan kinerja yang lebih optimal. S Leloho, et al (2019) dalam penelitian perbandingan kinerja empat fungsi pelatihan: Resilient Backpropagation, Scale Conjugate Gradieant, Quasi-Newton, dan Levenberg-Marquardt diterapkan pada pelatihan Jaringan Syaraf Tiruan dalam prediksi rata-rata radiasi matahari [20]. proses perbandingan dilaksanakan dengan mengamati nilai Root Mean Square Error (RMSE), model Koefisien Korelasi (R), dan Mean Absolute Percentage Error (MAPE). Hasil dari penelitian ini diperoleh TRAIN-LM bekerja lebih baik dibandingkan model lainnya dan nilai koefisien korelasi 99,742% MAPE 2,394%. Pengujian, pelatihan, dan validasi yang dilakukan dengan menggunakan data aktual dari database NASA dengan enam lokasi yang berbeda di kota-kota Barat Laut. Hasil dari penelitian ini bermanfaat untuk optimalisasi kinerja jaringan mikro [37]. D. Uwanuakwa and P Akpinar (2020) dalam penelitiannya membahas mengoptimalkan peramalan dalamnya karbon tembok menggunakan algoritma OSS. Pelatihan jaringan yang dibangun dalam makalah ini menggabungkan 10 hidden layer yang tidak sama dengan 11 rasio distribusi data. Diman nilai (R) 0,99 dan persentasi hasil variasi 30-50% yang berarti nilai R meningkat signifikan dari yang diamati 60-80% [38]. Penelitian selanjutnya dilakukan untuk meramalkan radiasi gelombang panjang teluk benggala menggunakan lima fungsi pelatihan yang berbeda, yaitu: kuasi-Newton Broyden Fletcher Goldfarb Shanno (BFGS), garis potong satu langkah, gradien konjugasi dengan Beale-Powell restart, Levenberg-Marquardt, dan penurunan gradien. dalam penelitian menggunakan analisa performance (Root

mean square error, korelasi koefisien, akurasi prediksi) dan analisa parameter untuk mengetahui performansi fungsi pelatihan. berdasarkan hasil penelitian ditemukan bahwa kelima fungsi pelatihan yang dilakukan Levenberg-Marquardt menghasilkan nilai akurasi 99,81% lebih tinggi dari fungsi pelatihan lainnya [39].

Berdasarkan dari penelitian-penelitian yang dipap arkan, maka penelitian yang diusulkan dalam penelitian ini adalah analisa pemanfaatan fungsi pelatihan Machine Learning neural network bacpropagation standard dalam peramalan kemiskinan di Indonesia. Peneltian ini membahas teknik dan metode fungsi pelatihan yang digunakan dalam menyelesaikan masalah, dataset jumlah kemiskinan yang digunakan hanya membantu pembuktian dan pengujian. Dalam penerapannya Machine Learning dengan metode backpropagation standard pada proses pelatihan sering menghasilkan kecepatan konvergensi yang tidak baik, untuk itu perlu dilakukan kombinasi fungsi pelatihan dalam mempercepat konvergensi pelatihan jaringan, dengan menggunakan metode Bayesian Regulation, dan *One Step Secant* (OSS). Kinerja *Backpropagation standard* dilakukan analisis dan dibandingkan dengan metode *Bayesian Regulation*, dan *One Step Secant*. Hasil dari penelitian ini diharapkan dapat digunakan ataupun rujukan dalam menyelesaikan kasus peramalan kemiskinan di Indonesia, serta dapat membantu pihak akademisi dalam pengembangan studi selanjutnya.

II. METODE

A. Dataset Penelitian

Dalam proses analisis, penelitian ini menggunakan Dataset jumlah kemiskinan di Indonesia dalam jangka waktu 12 tahun [3] tahun 2009 - 2020) yang terdiri dari 34 provinsi, yaitu: Propinsi Aceh, Propinsi Sumatera Utara, Propinsi Sumatera Barat, Propinsi Riau, Propinsi Jambi, Propinsi Sumatera Selatan, Propinsi Bengkulu, Propinsi Lampung, Propinsi Kepulauan Bangka Belitung, Propinsi Kepulauan Riau, Propinsi DKI Jakarta, Propinsi Jawa Barat, Propinsi Jawa Tengah, Propinsi Di Yogyakarta, Propinsi Jawa Timur, Propinsi Banten, Propinsi Bali, Propinsi Nusa Tenggara Barat, Propinsi Nusa Tenggara Timur, Propinsi Kalimantan Barat, Propinsi Kalimantan Tengah, Propinsi Kalimantan Selatan, Propinsi Kalimantan Timur, Propinsi Kalimantan Utara, Propinsi Sulawesi Utara, Propinsi Sulawesi Tengah, Propinsi Sulawesi Tenggara, Propinsi Gorontalo, Propinsi Sulawesi Barat, Propinsi Maluku, Propinsi Maluku Utara, Propinsi Papua Barat, Propinsi Papua. Data diperoleh dari website Badan Pusat Statistik (BPS) Indonesia <https://www.bps.go.id/>.

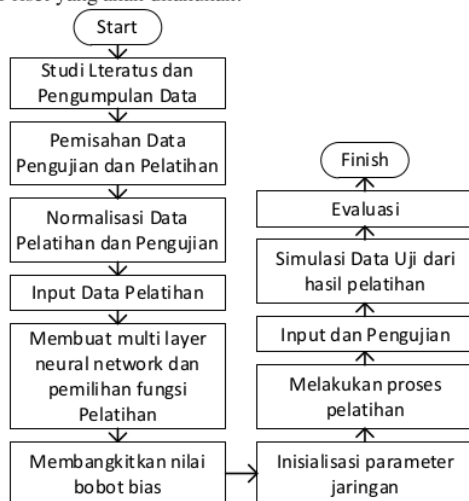
JUMLAH PENDUDUK MISKIN DI INDONESIA (Tahun 2009-2020)

No	Provinsi	Tahun			
		2009	2010		2020
1	Acch	892,86	861,85		833,91
2	Sumatera Utara	1499,68	1490,89		1356,72
3	Sumatera Barat	429,25	430,02		364,79
4	Riau	527,49	500,26		491,22

5	Jambi	249,69	241,61		288,1
6	Sumatera Selatan	1167,87	1125,73		1119,65
7	Bengkulu	324,13	324,93		306
8	Lampung	1558,28	1479,93		1091,14
9	Kep. Bangka Belitung	76,63	67,75		72,05
10	Kep. Riau	128,21	129,66		142,61
11	Dki Jakarta	323,17	312,18		496,84
12	Jawa Barat	4983,57	4773,72		4188,52
13	Jawa Tengah	5725,69	5369,16		4119,93
14	Di Yogyakarta	585,78	577,3		503,14
...					
34	Papua	760,35	761,62		912,23
	Jumlah	32571,31	31064,43		27549,7

B. Tahapan Riset

Dalam menyelesaikan riset pada gambar 1 ditampilkan alur riset yang akan dilakukan.



Gambar. 1 Tahapan Penelitian

Sesuai dengan Gambar 1 di atas bahwa langkah pertama yang dilakukan pada penelitian ini adalah melaksanakan studi literatur dari Jurnal, Proceeding, dan buku-buku yang selanjutnya mengumpulkan dataset penelitian (T₁₃ 11). tahapan berikutnya melakukan pemisahan dataset menjadi 2 bagian, yaitu: Data Pelatihan, dan Data Pengujian. Data pelatihan yang digunakan data jumlah kemiskinan tahun 2009-2013 (X1-X5) dengan target tahun 2014(Y1). Data pengujian tahun 2015-2019 (X6-X10) dengan target Y2 (tahun 2020). langkah berikutnya melaksanakan proses menormalkan dataset pelatihan dan data pengujian menggunakan rumus (1) [40].

$$x' = \frac{0.8(x-a)}{b-a} + 0.1 \quad (1)$$

Dimana : x' merupakan hasil data yang sudah dinormalisasi, 0.8 dan 0.1 nilai ketetapan. x = data yang akan dinormalkan, a = nilai terkecil dari dataset, b = nilai terbesar dari dataset. Data pelatihan yang sudah dinormalkan di input kedalam software Matlab_R2021a untuk diolah dengan membangun jaringan syaraf multi

layer. Arsitektur yang akan digunakan dalam pelatihan dan pengujian pada riset ini 5-8-1 (input 5, hidden layer 8, output 1), 5-9-1 (input 5, hidden layer 9, output 1), dan 5-11-1 (input 5, hidden layer 11, output 1). Dalam membangun arsitektur jaringan syaraf, hidden layer menggunakan fungsi transfer logsig (sigmoid biner), output layer menggunakan fungsi linear (purelin). Pada tahap ini akan dipilih fungsi pelatihan dari data yang dilatih bergantian menggunakan fungsi pelatihan traingd pada propogasi balik standard selanjutnya dilakukan optimasi dengan OSS (trainoss) dan Sysian Regulations (trainbr). Berikutnya membangkitkan nilai bobot (IW, LW) dan bias (b). Tahap berikutnya melakukan inisialisasi nilai parameter jaringan sesuai dengan fungsi pelatihan yang dipakai. Selanjutnya memasukkan script dalam melaksanakan proses pelatihan serta melihat nilai performances. Setelah pelatihan selesai dilaksanakan, selanjutnya memasukkan data pengujian yang dinormalisasi untuk melakukan proses simulasi data uji sesuai dengan hasil pelatihan. Jika semua tahapan sudah dilaksanakan, maka dilanjutkan dengan proses evaluasi untuk melihat kinerja dan kemampuan metode fungsi pelatihan yang terbaik.

C. Parameter Fungsi Pelatihan

Parameter fungsi pelatihan yang digunakan dalam penelitian ini akan diolah dan diproses dengan menggunakan aplikasi Matlab R2021a.

Parameter Backpropagation Standard

```

clc; %bersihkan Layar
clear; %bersihkan Layar
close all; %bersihkan Layar
warning off all; %bersihkan Layar

```

```

% Read Ms.Excel
all_sample_asli = xlsread('Pengolahan Data (1).xlsx','C4:N37');
data_latih = xlsread('Pengolahan Data (1).xlsx','C4:G37');
targetlatih = xlsread('Pengolahan Data (1).xlsx','H48:H81');

```

```

% melakukan transpose
data_latih = data_latih';
targetlatih = targetlatih';

```

```

% mengubah matriks menjadi vektor
all_data_asli_vektor = all_sample_asli(:);
latih_data_vektor = data_latih(:);

```

```

% mencari nilai min dan max dari data asli
mindata = min(all_data_asli_vektor);
maxdata = max(all_data_asli_vektor);

```

```

% proses dinormalkan
[m1,n1] = size(latih_data_vektor);
data_norm1 = zeros(m1,n1);
for xi = 1:m1
    for y1 = 1:n1
        data_norm1(xi,y1) = (0.8*(latih_data_vektor(xi,y1)-mindata)/(maxdata-mindata))+0.1;
    end
end

```

```

end

% menyiapkan sample normal
var = 34;
jlh_data = 5;
latih_data_norm = zeros(var,jlh_data);
% menyusun sampel latih normal
for m1 = 1:var
    for n1 = 1:jlh_data
        latih_data_norm(m1,n1) =
data_norm1((jlh_data*m1)+(m1+n1)-(jlh_data+m1));
    end
end

% melakukan transpose terhadap data latih ternormalisasi
latih_data_norm = latih_data_norm';

% menetapkan parameter JST
jumlah_neuron1 = 11;
fungsiaktivasi1 = 'logsig';
fungsiaktivasi2 = 'purelin';
fungsipelatihan1 = 'traingd';

% Merancang Model JST BG
rng('default') % nilai acak
jaringan = newff(minmax(latih_data_norm),[jumlah_neuron1
1],...
    {fungsiaktivasi1,fungsiaktivasi2},fungsipelatihan1);

% Nilai parameter backpropagation
jaringan.trainParam.epochs = 10000;
jaringan.trainParam.goal = 0.001;
jaringan.trainParam.lr = 0.01;
jaringan.trainParam.show = 1000;

% melakukan pelatihan jaringan
jaringan = train(jaringan,latih_data_norm,targetlatih);

% membaca hasil simulasi
latih_hasil_norm = sim(jaringan,latih_data_norm);

% melakukan denormalisasi terhadap hasil latih normalisasi
hasil_latih_asli = (((latih_hasil_norm-0.1)*(maxdata-
mindata))/0.8)+mindata;
% menghitung nilai MSE
nilai_error_latih = latih_hasil_norm - targetlatih;
SSE_latih = nilai_error_latih.^2;
Total_SSE_latih = sum(SSE_latih);
error_MSE_latih = (1/m)*sum(nilai_error_latih.^2);

% membaca data asli dari file excel
data_uji = xlsread('Pengolahan Data (1).xlsx',1,'I4:M37');
target_uji = xlsread('Pengolahan Data (1).xlsx',1,'N48:N81');

% melakukan transpose
data_uji = data_uji';
target_uji = target_uji';

% mengubah matriks menjadi vektor
data_uji_vektor = data_uji(:);

% proses normalisasi
[m1,n1] = size(data_uji_vektor);

```

```

data_norm1 = zeros(m1,n1);
for x1 = 1:m1
    for y1 = 1:n1
        data_norm1(x1,y1) = (0.8*(data_uji_vektor(x1,y1)-
mindata)/(maxdata-mindata))+0.1;
    end
end

% menyiapkan saple uji normal
var = 34;
jlh_data = 5;
data_uji_norm = zeros(var,jlh_data);
% menyusun sampleuji normal
for m1 = 1:var
    for n1 = 1:jlh_data
        data_uji_norm(m1,n1) =
data_norm1((jlh_data*m1)+(m1+n1)-(jlh_data+m1));
    end
end

% melakukan transpose terhadap data uji ternormalisasi
data_uji_norm = data_uji_norm';

% membaca pengolahan sample
uji_norm_hasil1 = sim(jaringan,data_uji_norm);

% melakukan denormalisasi terhadap hasil uji normalisasi
hasil_uji_asli = (((uji_norm_hasil1-0.1)*(maxdata-
mindata))/0.8)+mindata;

% menghitung nilai error MSE
error_uji1 = uji_norm_hasil1 - target_uji;
SSE_uji = error_uji1.^2;
Total_SSE_uji = sum(error_uji1.^2);
error_MSE_uji = (1/m)*sum(error_uji1.^2);

```

Parameter One Step Secant (OSS)

```

clc; %bersihkan Layar
clear; %bersihkan Layar
close all; %bersihkan Layar
warning off all; %bersihkan Layar

% Read Ms.Excel
all_sample_asli = xlsread('Pengolahan Data (1).xlsx',1,'C4:N37');
data_latih = xlsread('Pengolahan Data (1).xlsx',1,'C4:G37');
targetlatih = xlsread('Pengolahan Data (1).xlsx',1,'H48:H81');

% melakukan transpose
data_latih = data_latih';
targetlatih = targetlatih';

% mengubah matriks menjadi vektor
all_data_asli_vektor = all_sample_asli(:);
latih_data_vektor = data_latih(:);

% mencari nilai min dan max dari data asli
mindata = min(all_data_asli_vektor);
maxdata = max(all_data_asli_vektor);

% proses dinormalkan
[m1,n1] = size(latih_data_vektor);
data_norm1 = zeros(m1,n1);
for xi = 1:m1

```

```

    for y1 = 1:n1
        data_norm1(xi,y) = (0.8*(latih_data_vektor(xi,y1)-
mindata)/(maxdata-mindata))+0.1;
    end
end

% menyiapkan sample normal
var = 34;
jlh_data = 5;
latih_data_norm = zeros(var,jlh_data);
% menyusun sampel latih normal
for m1 = 1:var
    for n1 = 1:jlh_data
        latih_data_norm(m1,n1) =
data_norm1((jlh_data*m1)+(m1+n1)-(jlh_data+m1));
    end
end

% melakukan transpose terhadap data latih ternormalisasi
latih_data_norm = latih_data_norm';

% menetapkan parameter JST
jumlah_neuron1 = 11;
fungsiaktivasi1 = 'logsig';
fungsiaktivasi2 = 'purelin';
fungsipelatihan1 = 'trainoss';

% Merancang Model JST BG
rng('default') % nilai acak
jaringan = newff(minmax(latih_data_norm),[jumlah_neuron1
1],...
    {fungsiaktivasi1,fungsiaktivasi2},fungsipelatihan1);

% Nilai parameter backpropagation
jaringan.trainParam.epochs = 10000;
jaringan.trainParam.goal = 0.001;
jaringan.trainParam.lr = 0.01;
jaringan.trainParam.show = 1000;

% melakukan pelatihan jaringan
jaringan = train(jaringan,latih_data_norm,targetlatih);

% membaca hasil simulasi
latih_hasil_norm = sim(jaringan,latih_data_norm);

% melakukan denormalisasi terhadap hasil latih normalisasi
hasil_latih_asli = (((latih_hasil_norm-0.1)*(maxdata-
mindata))/(0.8)+mindata);
% menghitung nilai MSE
nilai_error_latih = latih_hasil_norm - targetlatih;
SSE_latih = nilai_error_latih.^2;
Total_SSE_latih = sum(SSE_latih);
error_MSE_latih = (1/m)*sum(nilai_error_latih.^2);

% membaca data asli dari file excel
data_uji = xlsread('Pengolahan Data (1).xlsx',1,'I4:M37');
target_uji = xlsread('Pengolahan Data (1).xlsx',1,'N48:N81');

% melakukan transpose
data_uji = data_uji';
target_uji = target_uji';

% mengubah matriks menjadi vektor

```

```

data_uji_vektor = data_uji(:);

% proses normalisasi
[m1,n1] = size(data_uji_vektor);
data_norm1 = zeros(m1,n1);
for x1 = 1:m1
    for y1 = 1:n1
        data_norm1(x1,y1) = (0.8*(data_uji_vektor(x1,y1)-
mindata)/(maxdata-mindata))+0.1;
    end
end

% menyiapkan saple uji normal
var = 34;
jlh_data = 5;
data_uji_norm = zeros(var,jlh_data);
% menyusun sampleuji normal
for m1 = 1:var
    for n1 = 1:jlh_data
        data_uji_norm(m1,n1) =
data_norm1((jlh_data*m1)+(m1+n1)-(jlh_data+m1));
    end
end

% melakukan transpose terhadap data uji ternormalisasi
data_uji_norm = data_uji_norm';

% membaca pengolahan sample
uji_norm_hasil1 = sim(jaringan,data_uji_norm);

% melakukan denormalisasi terhadap hasil uji normalisasi
hasil_uji_asli = (((uji_norm_hasil1-0.1)*(maxdata-
mindata))/(0.8)+mindata);

% menghitung nilai error MSE
error_uji1 = uji_norm_hasil1 - target_uji;
SSE_uji = error_uji1.^2;
Total_SSE_uji = sum(error_uji1.^2);
error_MSE_uji = (1/m)*sum(error_uji1.^2);

```

Parameter Bayesian Regulation

```

clc; %bersihkan Layar
clear; %bersihkan Layar
close all; %bersihkan Layar
warning off all; %bersihkan Layar

% Read Ms.Excel
all_sample_asli = xlsread('Pengolahan Data (1).xlsx',1,'C4:N37');
data_latih = xlsread('Pengolahan Data (1).xlsx',1,'C4:G37');
targetlatih = xlsread('Pengolahan Data (1).xlsx',1,'H48:H81');

% melakukan transpose
data_latih = data_latih';
targetlatih = targetlatih';

% mengubah matriks menjadi vektor
all_data_asli_vektor = all_sample_asli(:);
latih_data_vektor = data_latih(:);

% mencari nilai min dan max dari data asli
mindata = min(all_data_asli_vektor);
maxdata = max(all_data_asli_vektor);

```

```

% proses dinormalkan
[m1,n1] = size(latih_data_vektor);
data_norm1 = zeros(m1,n1);
for xi = 1:m1
    for y1 = 1:n1
        data_norm1(xi,y1) = (0.8*(latih_data_vektor(xi,y1)-
mindata)/(maxdata-mindata))+0.1;
    end
end

% menyiapkan sample normal
var = 34;
jlh_data = 5;
latih_data_norm = zeros(var,jlh_data);
% menyusun sampel latih normal
for m1 = 1:var
    for n1 = 1:jlh_data
        latih_data_norm(m1,n1) =
data_norm1((jlh_data*m1)+(m1+n1)-(jlh_data+m1));
    end
end

% melakukan transpose terhadap data latih ternormalisasi
latih_data_norm = latih_data_norm';

% menetapkan parameter JST
jumlah_neuron1 = 11;
fungsiaktivasi1 = 'logsig';
fungsiaktivasi2 = 'purelin';
fungsiipelatihan1 = 'trainbr';

% Merancang Model JST BG
rng('default') % nilai acak
jaringan = newff(minmax(latih_data_norm),[jumlah_neuron1
1],...
    {fungsiaktivasi1,fungsiaktivasi2},fungsiipelatihan1);

% Nilai parameter backpropagation
jaringan.trainParam.epochs = 10000;
jaringan.trainParam.goal = 0.001;
jaringan.trainParam.lr = 0.01;
jaringan.trainParam.show = 1000;

% melakukan pelatihan jaringan
jaringan = train(jaringan,latih_data_norm,targetlatih);

% membaca hasil simulasi
latih_hasil_norm = sim(jaringan,latih_data_norm);

% melakukan denormalisasi terhadap hasil latih normalisasi
hasil_latih_asli = (((latih_hasil_norm-0.1)*(maxdata-
mindata))/0.8)+mindata;
% menghitung nilai MSE
nilai_error_latih = latih_hasil_norm - targetlatih;
SSE_latih = nilai_error_latih.^2;
Total_SSE_latih = sum(SSE_latih);
error_MSE_latih = (1/m)*sum(nilai_error_latih.^2);

% membaca data asli dari file excel
data_uji = xlsread('Pengolahan Data (1).xlsx',1,'I4:M37');
target_uji = xlsread('Pengolahan Data (1).xlsx',1,'N48:N81');

% melakukan transpose

```

```

data_uji = data_uji';
target_uji = target_uji';

% mengubah matriks menjadi vektor
data_uji_vektor = data_uji(:);

% proses normalisasi
[m1,n1] = size(data_uji_vektor);
data_norm1 = zeros(m1,n1);
for x1 = 1:m1
    for y1 = 1:n1
        data_norm1(x1,y1) = (0.8*(data_uji_vektor(x1,y1)-
mindata)/(maxdata-mindata))+0.1;
    end
end

% menyiapkan saple uji normal
var = 34;
jlh_data = 5;
data_uji_norm = zeros(var,jlh_data);
% menyusun sampleuji normal
for m1 = 1:var
    for n1 = 1:jlh_data
        data_uji_norm(m1,n1) =
data_norm1((jlh_data*m1)+(m1+n1)-(jlh_data+m1));
    end
end

% melakukan transpose terhadap data uji ternormalisasi
data_uji_norm = data_uji_norm';

% membaca pengolahan sample
uji_norm_hasil1 = sim(jaringan,data_uji_norm);

% melakukan denormalisasi terhadap hasil uji normalisasi
hasil_uji_asli = (((uji_norm_hasil1-0.1)*(maxdata-
mindata))/0.8)+mindata;

% menghitung nilai error MSE
error_uji1 = uji_norm_hasil1 - target_uji;
SSE_uji = error_uji1.^2;
Total_SSE_uji = sum(error_uji1.^2);
error_MSE_uji = (1/m)*sum(error_uji1.^2);

```

III. HASIL DAN PEMBAHASAN

A. Normalisasi Data

Data jumlah kemiskinan di negara Indonesia (Tabel 1), dinormalkan menggunakan rumus (1) dan selanjutnya dibuat 2 kelompok, yakni Dataset Pelatihan, dan Dataset Pengujian. Dataset pelatihan diambil dari dataset jumlah kemiskinan di Indonesia dari tahun 2009-2013 (X1-X5) dan target tahun 2014 (Y1)

TA 6 IL 2

NORMALISASI DATA PELATIHAN

No	X1	X2	X3	X4	X5	Y1
1	0.1072	0.1070	0.1072	0.1071	0.1069	0.1068

2	0,1124	0,1123	0,1122	0,1113	0,1114	0,1112
3	0,1033	0,1033	0,1034	0,1030	0,1029	0,1027
4	0,1041	0,1039	0,1037	0,1037	0,1041	0,1039
5	0,1018	0,1017	0,1020	0,1019	0,1020	0,1020
6	0,1096	0,1092	0,1088	0,1085	0,1090	0,1089
7	0,1024	0,1024	0,1022	0,1023	0,1024	0,1023
8	0,1129	0,1122	0,1107	0,1100	0,1093	0,1093
9	0,1003	0,1002	0,1003	0,1003	0,1003	0,1002
10	0,1007	0,1008	0,1008	0,1008	0,1007	0,1007
11	0,1024	0,1023	0,1027	0,1028	0,1028	0,1032
12	0,1419	0,1401	0,1390	0,1371	0,1368	0,1356
13	0,1482	0,1451	0,1429	0,1409	0,1395	0,1383
14	0,1046	0,1045	0,1044	0,1044	0,1042	0,1042
15	0,1507	0,1465	0,1450	0,1417	0,1409	0,1399
16	0,1063	0,1061	0,1055	0,1051	0,1054	0,1052
17	0,1012	0,1011	0,1011	0,1010	0,1012	0,1013
18	0,1086	0,1082	0,1072	0,1067	0,1065	0,1066
19	0,1082	0,1082	0,1082	0,1081	0,1082	0,1081
20	0,1033	0,1033	0,1029	0,1027	0,1030	0,1029
...	...	...	...	...	...	...
34	0,1061	0,1061	0,9000	0,1079	0,1086	0,1070

Data pengujian diambil dari dataset jumlah kemiskinan di Indonesia dari tahun 2015-2019 (X6-X10) dan target tahun 2020 (Y2).

TABEL 3

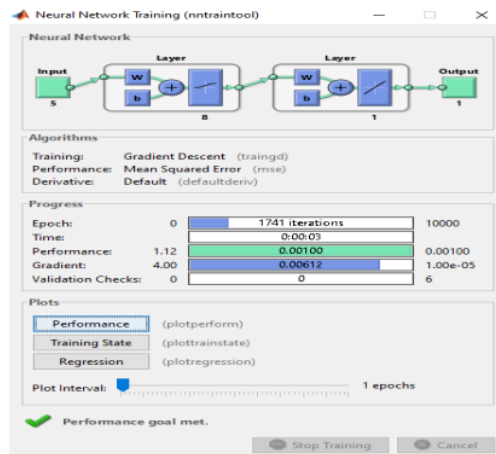
12 NORMALISASI DATA PENGUJIAN						
No	X6	X7	X8	X9	X10	Y2
1	0,1069	0,1068	0,1067	0,1067	0,1065	0,1067
2	0,1124	0,1120	0,1109	0,1106	0,1103	0,1112
3	0,1026	0,1028	0,1027	0,1026	0,1026	0,1027
4	0,1044	0,1039	0,1039	0,1038	0,1038	0,1038
5	0,1023	0,1021	0,1020	0,1020	0,1020	0,1021
6	0,1091	0,1089	0,1089	0,1088	0,1087	0,1091
7	0,1024	0,1024	0,1022	0,1022	0,1022	0,1022
8	0,1090	0,1093	0,1088	0,1089	0,1085	0,1089
9	0,1002	0,1003	0,1003	0,1002	0,1002	0,1003
10	0,1006	0,1007	0,1007	0,1007	0,1007	0,1009
11	0,1028	0,1029	0,1030	0,1028	0,1027	0,1039
12	0,1377	0,1350	0,1316	0,1296	0,1283	0,1351
13	0,1378	0,1377	0,1352	0,1324	0,1308	0,1346
14	0,1038	0,1038	0,1036	0,1035	0,1034	0,1039
15	0,1401	0,1390	0,1370	0,1360	0,1340	0,1385
16	0,1055	0,1052	0,1056	0,1053	0,1051	0,1069
17	0,1015	0,1011	0,1012	0,1011	0,1010	0,1013
18	0,1065	0,1063	0,1060	0,1059	0,1056	0,1060
19	0,1095	0,1094	0,1093	0,1093	0,1092	0,1096
20	0,1031	0,1030	0,1030	0,1028	0,1028	0,1028
...	...	...	...	...	...	...
34	0,1073	0,1074	0,1074	0,1074	0,1073	0,1074

B. Hasil Pelatihan dan Pengujian

Simulasi dan testing dilaksanakan dengan Aplikasi Matlab R2021a dengan 3 model jaringan, yaitu: 5-8-1 (input 5, hidden layer 8, output 1), 5-9-1 (input 5, hidden layer 9, output 1), dan 5-11-1 (input 5, hidden layer 11, output 1).

Pelatihan dan pengujian dengan Metode Backpropagation Standard.

Model Arsitektur Jaringan 5-8-1



Gambar. 2 Pelatihan Model 5-8-1 (trainingd)

Hasil dari pelatihan dengan fungsi traingd model 5-8-1, epoch= 1741 iterasi, waktu latih = 3 detik

TABEL 4
HASIL PELATIHAN MODEL 5-8-1

Target (Y2)	Epoch 1741			
	Actual	Error	SSE	Performance
0,1067	0,097795	-0,00893	7,97E-05	0,002287423
0,1112	0,088621	-0,02253	0,000508	
0,1027	0,100149	-0,0026	6,76E-06	
0,1038	0,099622	-0,0042	1,76E-05	
0,1021	0,100342	-0,00176	3,08E-06	
0,1091	0,09511	-0,01403	0,000197	
0,1022	0,100266	-0,00198	3,93E-06	
0,1089	0,094975	-0,01393	0,000194	
0,1003	0,100818	0,00055	3,02E-07	
0,1009	0,100726	-0,00014	1,95E-08	
0,1039	0,100111	-0,00376	1,41E-05	
0,1351	-0,07752	-0,21265	0,045222	
0,1346	-0,01687	-0,15143	2,29E-02	
0,1039	0,099752	-0,00417	1,74E-05	
0,1385	0,048973	-0,08953	8,02E-03	
0,1069	0,098972	-0,00795	6,32E-05	
0,1013	0,100601	-0,00072	5,25E-07	
0,106	0,098261	-0,00772	5,95E-05	
0,1096	0,094316	-0,01528	0,000234	
0,1028	0,100051	-0,00275	7,54E-06	
...	...	...	...	
0,1074	0,097204	-0,01018	0,000104	
Total SSE			0,077772	
MSE			0,002287	

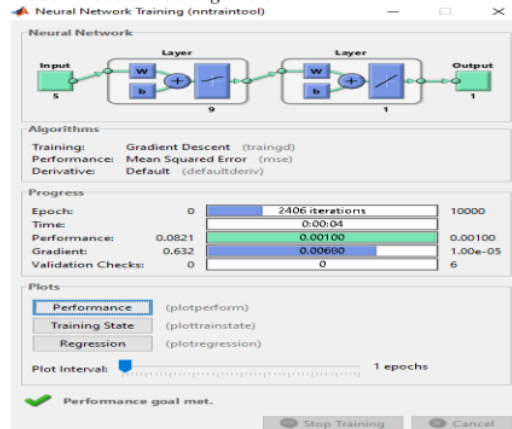
TABEL 5
HASIL PENGUJIAN MODEL 5-8-1

Target (Y2)	Epoch 1			
	Actual	Error	SSE	Performance
0,1067	0,124409	0,017687	0,000313	0,001574989
0,1112	0,125456	0,014306	0,000205	
0,1027	0,126966	0,024218	0,000587	
0,1038	0,126576	0,022757	0,000518	
0,1021	0,126952	0,024854	0,000618	
0,1091	0,126793	0,01765	0,000312	
0,1022	0,123906	0,021656	0,000469	
0,1089	0,12666	0,017759	0,000315	
0,1003	0,070645	-0,02962	0,000877	

0,1009	-0,06862	-0,16948	0,028724
0,1039	0,018405	-0,08546	0,007304
0,1351	0,097492	-0,03765	0,001417
0,1346	0,126965	-0,00759	5,77E+09
0,1039	0,126623	0,022704	0,000515
0,1385	0,129349	-0,00916	8,39E+09
0,1069	0,126539	0,019616	0,000385
0,1013	0,126568	0,025242	0,000637
0,106	0,126602	0,020624	0,000425
0,1096	0,126361	0,016762	0,000281
0,1028	0,126621	0,023823	0,000568
....			
0,1074	0,127993	0,020608	0,000425
		Total SSE	0,05355
		MSE	0,001575

Gambar 2, Tabel 4, dan Tabel 5 merupakan hasil pelatihan dan pengujian parameter backpropagation standar yang dilakukan dengan Matlab, selanjutnya membandingkan hasil pelatihan dan pengujian dengan menggunakan Ms. Excel. Nilai Actual dihasilkan dari output hasil pelatihan dan pengujian. Value Errors dihasilkan berdasarkan kalkulasi target (Y) Actual. value SSE dihasilkan dari perkiraan Errors kuadrat. Value total SSE dihasilkan berdasarkan pertambahan value SSE. Value MSE/Performances (Pref) dihasilkan dari jumlah SSE/34 dimana 34 merupakan jumlah baris data. Berdasarkan dari hasil pelatihan dan pengujian Matlab dan Ms. Excel hasilnya sesuai (valid).

Model Arsitektur Jaringan 5-9-1



Gambar. 3 Pelatihan Model 5-9-1 (traind)

Hasil dari pelatihan dengan fungsi traind model 5-9-1, epoch = 2406 iterasi, waktu latih = 4 detik

TABEL 6
HASIL PELATIHAN MODEL 5-9-1

Target (Y1)	Actual	Error	SSE	Performance
0,1068	0,06095	-0,0458	2,10E-03	0,000999688
0,1112	0,037115	-0,07407	0,005486	
0,1027	0,1058	0,003138	9,84E-06	
0,1039	0,104913	0,001035	1,07E-06	
0,102	0,112256	0,010212	1,04E-04	
0,1089	0,061878	-0,04698	0,002207	

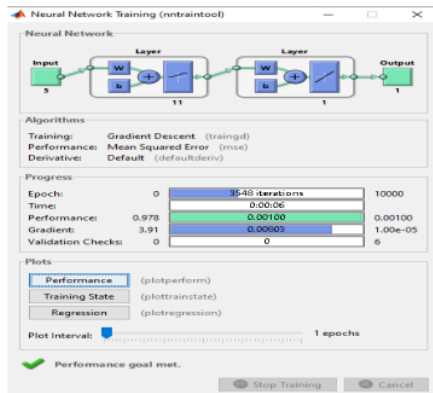
0,1023	0,113065	0,010726	1,15E-04
0,1093	0,058526	-0,05082	0,002583
0,1002	0,122885	0,022658	5,13E-04
0,1007	0,119506	1,88E-02	3,53E-04
0,1032	0,10376	0,000606	3,67E-07
0,1356	0,18019	0,044624	0,001991
0,1383	0,177354	0,039053	1,53E-03
0,1042	0,090616	-0,01355	1,84E-04
0,1399	0,183445	0,043563	1,90E-03
0,1052	0,097172	-0,00798	6,38E-05
0,1013	0,123248	0,021931	4,81E-04
0,1066	0,081615	-0,02496	0,000623
0,1081	0,054485	-0,05358	0,00287
0,1029	0,118188	0,015295	2,34E-04
....			
0,107	0,02815	-0,07883	0,006214
		Total SSE	0,033989
		MSE	0,001

TABEL 7
HASIL PENGUJIAN MODEL 5-9-1

Target (Y2)	Actual	Error	SSE	Performance
0,1067	0,066788	-0,03993	1,59E-03	0,000922011
0,1112	0,045302	-0,06585	0,004336	
0,1027	0,108341	0,005593	3,13E-05	
0,1038	0,098064	-0,00575	3,31E-05	
0,1021	0,112937	0,010839	1,17E-04	
0,1091	0,04702	-0,06212	0,003859	
0,1022	0,112449	0,010199	1,04E-04	
0,1089	0,044187	-0,06471	0,004188	
0,1003	0,122517	0,022249	4,95E-04	
0,1009	0,120228	0,019362	3,75E-04	
0,1039	0,106477	0,00261	6,81E-06	
0,1351	0,168881	0,033742	0,001139	
0,1346	0,172644	0,038086	1,45E-03	
0,1039	0,103181	-0,00074	5,45E-07	
0,1385	0,162449	0,023943	5,73E-04	
0,1069	0,079613	-0,02731	7,46E-04	
0,1013	0,119067	0,017742	3,15E-04	
0,106	0,078015	-0,02796	7,82E-04	
0,1096	0,042301	-0,0673	0,004529	
0,1028	0,108999	0,006201	3,85E-05	
....				
0,1074	0,058753	-0,04863	0,002365	
		Total SSE	0,031348	
		MSE	0,000922	

Gambar 3, Tabel 6, dan Tabel 7 merupakan hasil pelatihan dan pengujian parameter backpropagation standar yang dilakukan dengan Matlab, selanjutnya membandingkan hasil pelatihan dan pengujian dengan menggunakan Ms. Excel. Nilai Actual dihasilkan dari output hasil pelatihan dan pengujian. Value Errors dihasilkan berdasarkan kalkulasi target (Y) Actual. value SSE dihasilkan dari perkiraan Errors kuadrat. Value total SSE dihasilkan berdasarkan pertambahan value SSE. Value MSE/Performances (Pref) dihasilkan dari jumlah SSE/34 dimana 34 merupakan jumlah baris data. Berdasarkan dari hasil pelatihan dan pengujian Matlab dan Ms. Excel hasilnya sesuai (valid).

Model Arsitektur Jaringan 5-11-1



Gambar. 4 Pelatihan Model 5-11-1 (traingd)

Hasil dari pelatihan dengan fungsi traingd model 5-11-1, epoch= 3548 iterasi, waktu latih = 6 detik

TABEL 8
HASIL PELATIHAN MODEL 5-11-1

Target (Y1)	Epoch 3548			
	Actual	Error	SSE	Performance
0,1068	0,0896	-0,01715	2,94E-04	0,00099982
0,1112	0,068914	-0,04227	0,001787	
0,1027	0,103691	0,001029	1,06E-06	
0,1039	0,099318	-0,00456	2,08E-05	
0,102	0,126141	0,024096	5,81E-04	
0,1089	0,061608	-0,04725	0,002232	
0,1023	0,113312	0,010973	1,20E-04	
0,1093	0,029773	-0,07957	0,006332	
0,1002	0,134005	0,033778	1,14E-03	
0,1007	0,12999	2,93E-02	8,57E-04	
0,1032	0,126505	0,02335	5,45E-04	
0,1356	0,161609	0,026043	0,000678	
0,1383	0,158274	0,019973	3,99E-04	
0,1042	0,097855	-0,00631	3,99E-05	
0,1399	0,169402	0,02952	8,71E-04	
0,1052	0,066549	-0,03861	1,49E-03	
0,1013	0,122984	0,021666	4,69E-04	
0,1066	0,041618	-0,06496	0,004219	
0,1081	0,080882	-0,02718	0,000739	
0,1029	0,094652	-0,00824	6,79E-05	
.....				
0,107	0,051102	-0,05588	0,003122	
Total SSE			0,033994	0,001
MSE			0,001	

TABEL 9
HASIL PENGUJIAN MODEL 5-11-1

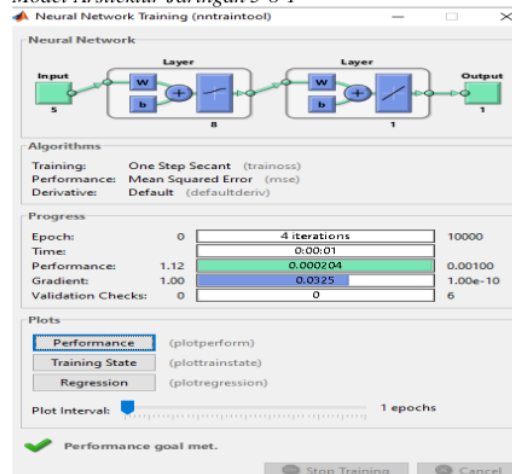
Target (Y2)	Epoch 1			
	Actual	Error	SSE	Performance
0,1067	0,085814	-0,02091	4,37E-04	0,000902276
0,1112	0,052959	-0,05819	0,003386	
0,1027	0,110221	0,007473	5,58E-05	
0,1038	0,100267	-0,00355	1,26E-05	
0,1021	0,115632	0,013534	1,83E-04	
0,1091	0,077406	-0,03174	0,001007	
0,1022	0,11228	0,01003	1,01E-04	
0,1089	0,073899	-0,035	0,001225	
0,1003	0,133944	0,033676	1,13E-03	
0,1009	0,131498	0,030633	9,38E-04	
0,1039	0,110872	0,007006	4,91E-05	
0,1351	0,220151	0,085012	0,007227	

0,1346	0,197125	0,062567	3,91E-03	0,000902
0,1039	0,098803	-0,00512	2,62E-05	
0,1385	0,170951	0,032445	1,05E-03	
0,1069	0,097484	-0,00944	8,91E-05	
0,1013	0,123074	0,021748	4,73E-04	
0,106	0,079358	-0,02662	7,09E-04	
0,1096	0,0785	-0,0311	0,000967	
0,1028	0,107156	0,004358	1,90E-05	
.....				
0,1074	0,087692	-0,01969	0,000388	
Total SSE			0,030677	
MSE			0,000902	

Gambar 4, Tabel 8, dan Tabel 9 merupakan hasil pelatihan dan pengujian parameter backpropagation standar yang dilakukan dengan Matlab, selanjutnya membandingkan hasil pelatihan dan pengujian dengan menggunakan Ms. Excel. Nilai Actual dihasilkan dari output hasil pelatihan dan pengujian. Value Errors dihasilkan berdasarkan kalkulasi target (Y) Actual. value SSE dihasilkan dari perkiraan Errors kuadrat. Value total SSE dihasilkan berdasarkan pertambahan value SSE. Value MSE/Performances (Pref) dihasilkan dari jumlah SSE/34 dimana 34 merupakan jumlah baris data. Berdasarkan dari hasil pelatihan dan pengujian Matlab dan Ms. Excel hasilnya sesuai (valid).

Pelatihan dan pengujian dengan Metode One Step Secant (OSS).

Model Arsitektur Jaringan 5-8-1



Gambar. 5 Pelatihan Model 5-8-1 (trainoss)

Hasil dari pelatihan dengan fungsi trainoss model 5-8-1, epoch= 4 iterasi, waktu latih = 1 detik

TABEL 10
HASIL PELATIHAN MODEL 5-8-1

Target (Y1)	Epoch 4			
	Actual	Error	SSE	Performance
0,1068	0,110138	0,003387	1,15E-05	0,000204221
0,1112	0,100029	-0,01115	1,24E-04	
0,1027	0,112413	0,009751	9,51E-05	
0,1039	0,112073	0,008194	6,71E-05	

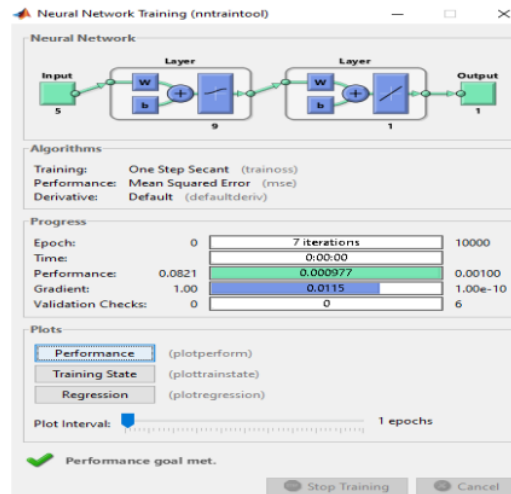
0,102	0,112816	0,010772	1,16E-04
0,1089	0,10723	-0,00163	2,64E-06
0,1023	0,112656	0,010318	1,06E-04
0,1093	0,102127	-0,00722	5,21E-05
0,1002	0,113138	0,012911	1,67E-04
0,1007	0,113052	1,23E-02	1,52E-04
0,1032	0,11264	0,009486	9,00E-05
0,1356	0,10321	-0,03236	1,05E-03
0,1383	0,154814	0,016513	2,73E-04
0,1042	0,111885	7,72E-03	5,95E-05
0,1399	0,158936	0,019054	3,63E-04
0,1052	0,110968	0,005811	3,38E-05
0,1013	0,112956	0,011638	1,35E-04
0,1066	0,109319	0,002744	7,53E-06
0,1081	0,108752	0,000692	4,79E-07
0,1029	0,112379	0,009486	9,00E-05
....			
0,107	0,155583	0,048606	2,36E-03
Total SSE			0,006944
MSE			2,04E-04

TABEL 11
HASIL PENGUJIAN MODEL 5-8-1

T _{target} (Y ₂)	Epoch 1			
	Actual	Error	SSE	Performance
0,1067	0,110375	0,003653	1,33E-05	0,002448018
0,1112	0,101697	-0,00945	8,94E-05	
0,1027	0,112569	0,009821	9,65E-05	
0,1038	0,112091	0,008272	6,84E-05	
0,1021	0,112734	0,010636	1,13E-04	
0,1091	0,107795	-0,00135	1,82E-06	
0,1022	0,112674	0,010425	1,09E-04	
0,1089	0,107665	-0,00124	1,53E-05	
0,1003	0,113142	0,012874	1,66E-04	
0,1009	0,113064	0,012198	1,49E-04	
0,1039	0,112531	0,008664	7,51E-05	
0,1351	-0,08646	-0,22159	4,91E-02	
0,1346	-0,01861	-0,15317	2,35E-02	
0,1039	0,112225	0,008305	6,90E-05	
0,1385	0,050025	-0,08848	7,83E-03	
0,1069	0,111461	0,004538	2,06E-05	
0,1013	0,112957	0,011631	1,35E-04	
0,106	0,110853	0,004876	2,38E-05	
0,1096	0,107013	-0,00259	6,69E-06	
0,1028	0,112486	0,009689	9,39E-05	
....				
0,1074	0,109805	0,00242	5,86E-06	
Total SSE			0,083233	
MSE			2,45E-03	

Gambar 5, Tabel 10, dan Tabel 11 merupakan hasil pelatihan dan pengujian parameter One Step Secant (OSS) yang dilakukan dengan Matlab, selanjutnya membandingkan hasil pelatihan dan pengujian dengan menggunakan Ms. Excel. Nilai Actual dihasilkan dari output hasil pelatihan dan pengujian. Value Errors dihasilkan berdasarkan kalkulasi target (Y) Actual. value SSE dihasilkan dari perkiraan Errors kuadrat. Value total SSE dihasilkan berdasarkan pertambahan value SSE. Value MSE/Performances (Pref) dihasilkan dari jumlah SSE/34 dimana 34 merupakan jumlah baris data. Berdasarkan dari hasil pelatihan dan pengujian Matlab dan Ms. Excel hasilnya sesuai (valid).

Model Arsitektur Jaringan 5-9-1



Gambar. 6 Pelatihan Model 5-9-1 (trainoss)

Hasil dari pelatihan dengan fungsi trainoss model 5-9-1, epoch = 7 iterasi, waktu latih = 0 detik

TABEL 12
HASIL PELATIHAN MODEL 5-9-1

T _{target} (Y ₁)	Epoch 7			
	Actual	Error	SSE	Performance
0,1068	0,051472	-0,05528	3,06E-03	0,000976727
0,1112	0,022561	-0,08862	7,85E-03	
0,1027	0,102974	0,000312	9,71E-08	
0,1039	0,100616	-0,00326	1,06E-05	
0,102	0,111411	0,009367	8,77E-05	
0,1089	0,050297	-0,05856	3,43E-03	
0,1023	0,111539	0,0092	8,46E-05	
0,1093	0,045026	-0,06432	4,14E-03	
0,1002	0,125211	0,024984	6,24E-04	
0,1007	0,120907	2,02E-02	4,08E-04	
0,1032	0,101446	-0,00171	2,92E-06	
0,1356	0,158409	0,022843	5,22E-04	
0,1383	0,157013	0,018712	3,50E-04	
0,1042	0,08532	-1,88E-02	3,55E-04	
0,1399	0,163368	0,023486	5,52E-04	
0,1052	0,090379	-0,01478	2,18E-04	
0,1013	0,124039	0,022721	5,16E-04	
0,1066	0,072422	-0,03415	1,17E-03	
0,1081	0,043611	-0,06445	4,15E-03	
0,1029	0,115794	0,012901	1,66E-04	
....				
0,107	0,087591	-0,01939	3,76E-04	
Total SSE			0,033209	
MSE			9,77E-04	

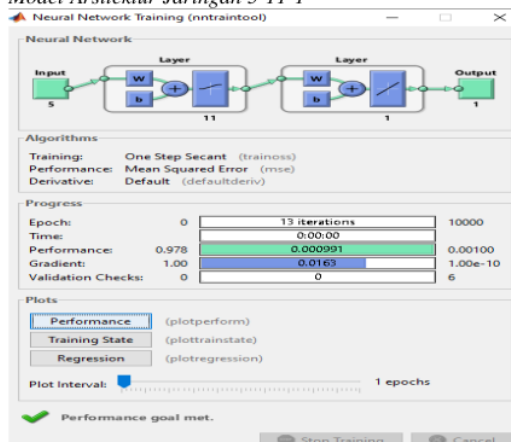
TABEL 13
HASIL PENGUJIAN MODEL 5-9-1

T _{target} (Y ₂)	Epoch 1			
	Actual	Error	SSE	Performance
0,1067	0,057849	-0,04887	2,39E-03	0,001164361
0,1112	0,031274	-0,07988	6,38E-03	
0,1027	0,106306	0,003558	1,27E-05	
0,1038	0,093518	-0,0103	1,06E-04	
0,1021	0,111798	0,0097	9,41E-05	
0,1091	0,035286	-0,07386	5,45E-03	
0,1022	0,111056	0,008806	7,76E-05	

0,1089	0,032442	-0,07646	5,85E-03
0,1003	0,124928	0,02466	6,08E-04
0,1009	0,121737	0,020872	4,36E-04
0,1039	0,10413	0,000264	6,95E-08
0,1351	0,146588	0,011449	1,31E-04
0,1346	0,150828	0,01627	2,65E-04
0,1039	0,099525	-0,00439	1,93E-05
0,1385	0,141017	0,002511	6,30E-06
0,1069	0,072803	-0,03412	1,16E-03
0,1013	0,119622	0,018297	3,35E-04
0,106	0,070256	-0,03572	1,28E-03
0,1096	0,029994	-0,0796	6,34E-03
0,1028	0,106501	0,003703	1,37E-05
....			
0,1074	0,04893	-0,05846	3,42E-03
Total SSE			0,039588
MSE			1,16E-03

Gambar 6, Tabel 12, dan Tabel 13 merupakan hasil pelatihan dan pengujian parameter One Step Secant (OSS) yang dilakukan dengan Matlab, selanjutnya membandingkan hasil pelatihan dan pengujian dengan menggunakan Ms. Excel. Nilai Actual dihasilkan dari output hasil pelatihan dan pengujian. Value Errors dihasilkan berdasarkan kalkulasi target (Y) Actual. value SSE dihasilkan dari perkiraan Errors kuadrat. Value total SSE dihasilkan berdasarkan pertambahan value SSE. Value MSE/Performances (Pref) dihasilkan dari jumlah SSE/34 dimana 34 merupakan jumlah baris data. Berdasarkan dari hasil pelatihan dan pengujian Matlab dan Ms. Excel hasilnya sesuai (valid).

Model Arsitektur Jaringan 5-11-1



Gambar. 7 Pelatihan Model 5-11-1 (trainoss)
Hasil dari pelatihan dengan fungsi trainoss model 5-11-1, epoch= 13 iterasi, waktu latih = 0 detik

TABEL 14
HASIL PELATIHAN MODEL 5-11-1

T _{target} (Y1)	Epoch 13			
	Actual	Error	SSE	Performance
0,1068	0,099995	-0,00676	4,56E-05	0,000990799
0,1112	0,083937	-0,02725	7,42E-04	
0,1027	0,107748	0,005085	2,59E-05	
0,1039	0,104827	0,000948	8,99E-07	

0,102	0,127967	0,025923	6,72E-04
0,1089	0,074417	-0,03444	1,19E-03
0,1023	0,116065	0,013727	1,88E-04
0,1093	0,04345	-0,0659	4,34E-03
0,1002	0,132824	0,032597	1,06E-03
0,1007	0,12975	2,90E-02	8,43E-04
0,1032	0,12964	0,026485	7,01E-04
0,1356	0,163991	0,028425	8,08E-04
0,1383	0,159951	0,02165	4,69E-04
0,1042	0,104187	1,79E+09	3,21E-10
0,1399	0,171153	0,031271	9,78E-04
0,1052	0,074833	-0,03032	9,20E-04
0,1013	0,123478	0,02216	4,91E-04
0,1066	0,052131	-0,05444	2,96E-03
0,1081	0,092868	-0,01519	2,31E-04
0,1029	0,098613	-0,00428	1,83E-05
....			
0,107	0,007344	-0,09963	9,93E-03
Total SSE			0,033687
MSE			9,91E-04

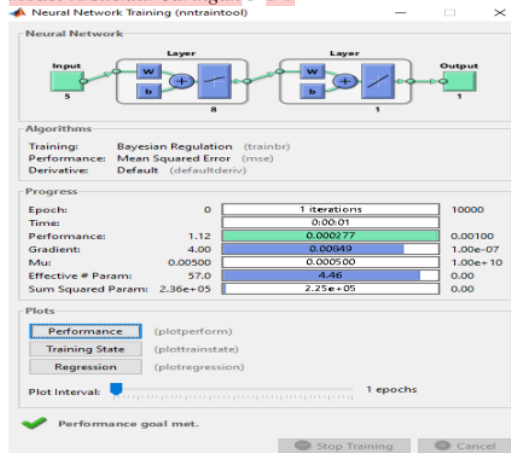
TABEL 15
HASIL PENGUJIAN MODEL 5-11-1

T _{target} (Y2)	Epoch 1			
	Actual	Error	SSE	Performance
0,1067	0,095744	-0,01098	1,21E-04	0,000825011
0,1112	0,067356	-0,04379	1,92E-03	
0,1027	0,113552	0,010804	1,17E-04	
0,1038	0,105735	0,001916	3,67E-06	
0,1021	0,117861	0,015763	2,48E-04	
0,1091	0,090146	-0,019	3,61E-04	
0,1022	0,114912	0,012662	1,60E-04	
0,1089	0,086659	-0,02224	4,95E-04	
0,1003	0,132752	0,032484	1,06E-03	
0,1009	0,131154	0,030288	9,17E-04	
0,1039	0,114433	0,010567	1,12E-04	
0,1351	0,22641	0,091271	8,33E-03	
0,1346	0,201794	0,067236	4,52E-03	
0,1039	0,103716	-0,0002	4,17E-08	
0,1385	0,17403	0,035524	1,26E-03	
0,1069	0,105179	-0,00174	3,04E-06	
0,1013	0,123532	0,022206	4,93E-04	
0,106	0,088219	-0,01776	3,15E-04	
0,1096	0,091796	-0,0178	3,17E-04	
0,1028	0,110827	0,008029	6,45E-05	
....				
0,1074	0,098572	-0,00881	7,77E-05	
Total SSE			0,02805	
MSE			8,25E-04	

Gambar 7, Tabel 14, dan Tabel 15 merupakan hasil pelatihan dan pengujian parameter One Step Secant (OSS) yang dilakukan dengan Matlab, selanjutnya membandingkan hasil pelatihan dan pengujian dengan menggunakan Ms. Excel. Nilai Actual dihasilkan dari output hasil pelatihan dan pengujian. Value Errors dihasilkan berdasarkan kalkulasi target (Y) Actual. value SSE dihasilkan dari perkiraan Errors kuadrat. Value total SSE dihasilkan berdasarkan pertambahan value SSE. Value MSE/Performances (Pref) dihasilkan dari jumlah SSE/34 dimana 34 merupakan jumlah baris data. Berdasarkan dari hasil pelatihan dan pengujian Matlab dan Ms. Excel hasilnya sesuai (valid).

Pelatihan dan pengujian dengan Metode Bayesian Regulation.

Model Arsitektur Jaringan 5-8-1



Gambar. 8 Pelatihan Model 5-8-1 (trainbr)

Hasil dari pelatihan dengan fungsi trainbr model 5-8-1, epoch= 1 iterasi, waktu latih = 1 detik

TABEL 16
HASIL PELATIHAN MODEL 5-8-1

Target (Y1)	Epoch 1			
	Actual	Error	SSE	Performance
0,1068	0,102458	-0,00429	1,84E-05	0,000276637
0,1112	0,092624	-0,01856	0,000344	
0,1027	0,104695	0,002033	4,13E-06	
0,1039	0,104348	0,00047	2,21E-07	
0,102	0,105112	0,003067	9,41E-06	
0,1089	0,09955	-0,00931	8,66E-05	
0,1023	0,104942	0,002603	6,78E-06	
0,1093	0,094547	-0,0148	0,000219	
0,1002	0,10544	0,005213	2,72E-05	
0,1007	0,105352	4,64E-03	2,15E-05	
0,1032	0,104937	0,001783	3,18E-06	
0,1356	0,099744	-0,03582	0,001283	
0,1383	0,139504	0,001203	1,45E-06	
0,1042	0,104168	-8,54E-07	7,30E-13	
0,1399	0,14135	0,001468	2,16E-06	
0,1052	0,103218	-0,00194	3,76E-06	
0,1013	0,105249	0,003931	1,55E-05	
0,1066	0,101568	-0,00501	2,51E-05	
0,1081	0,101081	-0,00698	4,87E-05	
0,1029	0,104648	0,001755	3,08E-06	
0,107	0,191142	0,084164	0,007084	
Total SSE			0,009406	0,000277
MSE			0,000277	

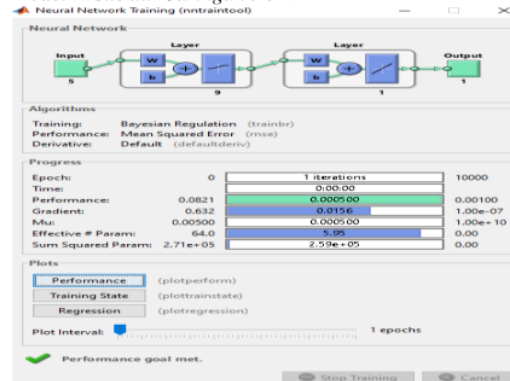
TABEL 17
HASIL PENGUJIAN MODEL 5-8-1

Target (Y2)	Epoch 1			
	Actual	Error	SSE	Performance
0,1067	0,102677	-0,00405	1,64E-05	0,002088815
0,1112	0,0942	-0,01695	0,000287	
0,1027	0,104853	0,002105	4,43E-06	
0,1038	0,104377	0,000558	3,11E-07	
0,1021	0,105025	0,002927	8,57E-06	
0,1091	0,100158	-0,00898	8,07E-05	
0,1022	0,10496	0,002711	7,35E-06	
0,1089	0,100013	-0,00889	7,90E-05	

0,1003	0,105444	0,005176	2,68E-05	0,000499547
0,1009	0,105364	0,004498	2,02E-05	
0,1039	0,104816	0,00095	9,02E-07	
0,1351	-0,069	-0,20414	0,041674	
0,1346	-0,00994	-0,1445	2,09E-02	
0,1039	0,104502	0,000583	3,39E-07	
0,1385	0,051492	-0,08701	7,57E-03	
0,1069	0,103757	-0,00317	1,00E-05	
0,1013	0,105254	0,003928	1,54E-05	
0,106	0,103134	-0,00284	8,09E-06	
0,1096	0,099402	-0,0102	0,00104	
0,1028	0,10477	0,001972	3,89E-06	
0,1074	0,102117	-0,00527	2,78E-05	
Total SSE			0,07102	
MSE			0,002089	

Gambar 8, Tabel 16, dan Tabel 17 merupakan hasil pelatihan dan pengujian parameter Bayesian Regulation yang dilakukan dengan Matlab, selanjutnya membandingkan hasil pelatihan dan pengujian dengan menggunakan Ms. Excel. Nilai Actual dihasilkan dari output hasil pelatihan dan pengujian. Value Errors dihasilkan berdasarkan kalkulasi target (Y) Actual, value SSE dihasilkan dari perkiraan Errors kuadrat. Value total SSE dihasilkan berdasarkan pertambahan value SSE. Value MSE/Performances (Pref) dihasilkan dari jumlah SSE/34 dimana 34 merupakan jumlah baris data. Berdasarkan dari hasil pelatihan dan pengujian Matlab dan Ms. Excel hasilnya sesuai (valid).

Model Arsitektur Jaringan 5-9-1



Gambar. 9 Pelatihan Model 5-9-1 (trainbr)

Hasil dari pelatihan dengan fungsi trainbr model 5-9-1, epoch = 1 iterasi, waktu latih = 0 detik

TABEL 18
HASIL PELATIHAN MODEL 5-9-1

Target (Y1)	Epoch 1			
	Actual	Error	SSE	Performance
0,1068	0,067443	-0,03931	1,55E-03	0,000499547
0,1112	0,050655	-0,06053	0,003664	
0,1027	0,102469	-0,00019	3,75E-08	
0,1039	0,103122	-0,00076	5,72E-07	
0,102	0,106676	0,004632	2,15E-05	
0,1089	0,071575	-0,03728	1,39E-03	
0,1023	0,108111	0,005773	3,33E-05	

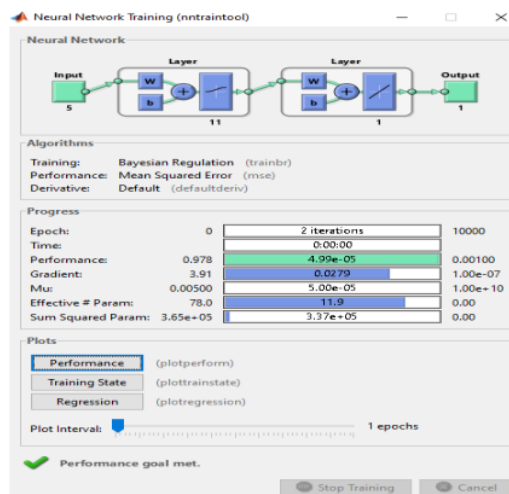
0,1093	0,067879	-0,04147	0,00172
0,1002	0,114449	0,014222	2,02E-04
0,1007	0,111861	1,12E-02	1,24E-04
0,1032	0,099822	-0,00333	1,11E-05
0,1356	0,151987	0,016421	0,00027
0,1383	0,139302	0,001001	1,00E-06
0,1042	0,090511	-1,37E-02	1,87E-04
0,1399	0,142003	0,002121	4,50E-06
0,1052	0,098917	-0,00624	3,89E-05
0,1013	0,11593	0,014613	2,14E-04
0,1066	0,08689	-0,01969	3,88E-04
0,1081	0,063343	-0,04472	2,00E-03
0,1029	0,113981	0,011088	1,23E-04
....			
0,107	0,050993	-0,05598	0,003134
Total SSE			0,016985
MSE			0,0005

TABEL 19
HASIL PENGUJIAN MODEL 5-9-1

Target (Y2)	Epoch 1			
	Actual	Error	SSE	Performance
0,1067	0,072309	-0,03441	1,18E-03	0,000549791
0,1112	0,057067	-0,05408	0,002925	
0,1027	0,104122	0,001374	1,89E-06	
0,1038	0,096709	-0,00711	5,05E-05	
0,1021	0,107576	0,005478	3,00E-05	
0,1091	0,057249	-0,05189	2,69E-03	
0,1022	0,107404	0,005154	2,66E-05	
0,1089	0,054355	-0,05455	2,98E-03	
0,1003	0,114067	0,013799	1,90E-04	
0,1009	0,112493	0,011628	1,35E-04	
0,1039	0,102622	-0,00124	1,55E-06	
0,1351	0,146018	0,010879	0,000118	
0,1346	0,147244	0,012686	1,61E-04	
0,1039	0,100814	-0,00311	9,65E-06	
0,1385	0,134892	-0,00361	1,31E-05	
0,1069	0,08178	-0,02514	6,32E-04	
0,1013	0,11199	0,010665	1,14E-04	
0,106	0,081555	-0,02442	5,96E-04	
0,1096	0,053503	-0,0561	0,003147	
0,1028	0,105145	0,002347	5,51E-06	
....				
0,1074	0,065878	-0,04151	1,72E-03	
Total SSE			0,018693	
MSE			0,00055	

Gambar 9, Tabel 18, dan Tabel 19 merupakan hasil pelatihan dan pengujian parameter Bayesian Regulation yang dilakukan dengan Matlab, selanjutnya membandingkan hasil pelatihan dan pengujian dengan menggunakan Ms. Excel. Nilai Actual dihasilkan dari output hasil pelatihan dan pengujian. Value Errors dihasilkan berdasarkan kalkulasi target (Y) Actual. value SSE dihasilkan dari perkiraan Errors kuadrat. Value total SSE dihasilkan berdasarkan pertambahan value SSE. Value MSE/Performances (Pref) dihasilkan dari jumlah SSE/34 dimana 34 merupakan jumlah baris data. Berdasarkan dari hasil pelatihan dan pengujian Matlab dan Ms. Excel hasilnya sesuai (valid).

Model Arsitektur Jaringan 5-11-1



Gambar. 10 Pelatihan Model 5-11-1 (trainbr)

Hasil dari pelatihan dengan fungsi trainbr model 5-11-1, epoch=2 iterasi, waktu latih = 0 detik

TABEL 20
HASIL PELATIHAN MODEL 5-11-1

Target (Y1)	Epoch 2			
	Actual	Error	SSE	Performance
0,1068	0,098905	-0,00785	6,16E-05	4,99135E-05
0,1112	0,105022	-0,00616	3,80E-05	
0,1027	0,095152	-0,00751	5,64E-05	
0,1039	0,096105	-0,00777	6,04E-05	
0,102	0,094841	-0,0072	5,19E-05	
0,1089	0,100866	-0,00799	6,38E-05	
0,1023	0,094853	-0,00749	5,60E-05	
0,1093	0,101892	-0,00746	5,56E-05	
0,1002	0,093413	-0,00681	4,64E-05	
0,1007	0,093738	-6,97E-03	4,86E-05	
0,1032	0,095569	-0,00759	5,75E-05	
0,1356	0,134683	-0,00088	7,79E-07	
0,1383	0,138787	0,000486	2,36E-07	
0,1042	0,096269	-7,90E-03	6,24E-05	
0,1399	0,14101	0,001128	1,27E-06	
0,1052	0,096823	-0,00833	6,95E-05	
0,1013	0,094027	-0,00729	5,32E-05	
0,1066	0,097681	-0,00889	7,91E-05	
0,1081	0,100278	-0,00778	6,06E-05	
0,1029	0,094993	-0,0079	6,24E-05	
....				
0,107	0,105305	-0,00167	2,80E-06	
Total SSE			0,001697	
MSE			4,99E-05	

TABEL 21
HASIL PENGUJIAN MODEL 5-11-1 (trainbr)

Target (Y2)	Epoch 1			
	Actual	Error	SSE	Performance
0,1067	0,098433	-0,00829	6,87E-05	5,68557E-05
0,1112	0,103327	-0,00782	6,12E-05	
0,1027	0,095003	-0,00774	6,00E-05	
0,1038	0,09581	-0,00801	6,41E-05	

0,1021	0,094534	-0,00756	5,72E-05
0,1091	0,100892	-0,00825	6,81E-05
0,1022	0,094667	-0,00758	5,75E-05
0,1089	0,100839	-0,00806	6,50E-05
0,1003	0,093388	-0,00688	4,73E-05
0,1009	0,093792	-0,00707	5,00E-05
0,1039	0,095165	-0,0087	7,57E-05
0,1351	0,127927	-0,00721	5,20E-05
0,1346	0,133252	-0,00131	1,71E-05
0,1039	0,095518	-0,0084	7,06E-05
0,1385	0,134923	-0,00358	1,28E-05
0,1069	0,097133	-0,00979	9,58E-05
0,1013	0,0938	-0,00753	5,66E-05
0,106	0,097401	-0,00858	7,36E-05
0,1096	0,101582	-0,00802	6,43E-05
0,1028	0,095108	-0,00769	5,91E-05
.....			
0,1074	0,09936	-0,00802	6,44E-05
Total SSE			0,001933
MSE			5,69E-05

Gambar 10, Tabel 20, dan Tabel 21 merupakan hasil pelatihan dan pengujian parameter Bayesian Regulation yang dilakukan dengan Matlab, selanjutnya membandingkan hasil pelatihan dan pengujian dengan menggunakan Ms. Excel. Nilai Actual dihasilkan dari output hasil pelatihan dan pengujian. Value Errors dihasilkan berdasarkan kalkulasi target (Y) Actual. value SSE dihasilkan dari perkiraan Errors kuadrat. Value total SSE dihasilkan berdasarkan pertambahan value SSE. Value MSE/Performances (Pref) dihasilkan dari jumlah SSE/34 dimana 34 merupakan jumlah baris data. Berdasarkan dari hasil pelatihan dan pengujian Matlab dan Ms. Excel hasilnya sesuai (valid).

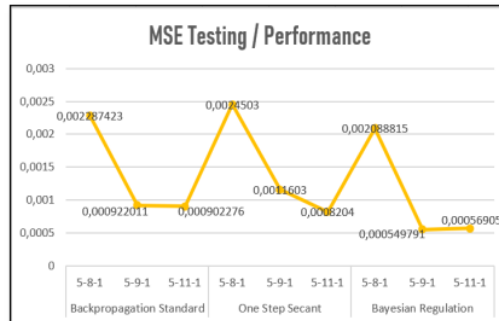
C. Analisis Hasil dan Evaluasi

Berdasarkan hasil pelatihan dan pengujian yang dilakukan dengan menggunakan aplikasi Matlab R2021a dan Ms. Excel, dilanjutkan dengan analisis dan evaluasi hasil dalam menghasilkan model arsitektur jaringan serta fungsi pelatihan yang baik.

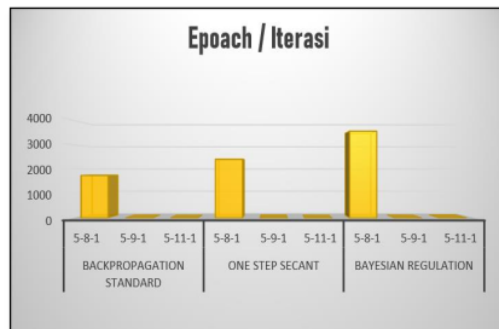
TABEL 21
HASIL ANALISA FUNGSI PELATIHAN

Algoritma Machine Learning Neural Network						
Model	Algoritma	Fungsi Aktivas i	Fungsi Trainin g	Epoch (Iterasi)	MSE Training	MSE Testing / Performanc e
5-8-1	Backpropagation Standard	logsig', 'purelin'	traingd	1741	0,0009998	0,002287423
	One Step Secant (OSS)	logsig', 'purelin'	traingos	4	0,00020404	0,0024503
	Bayesian Regulation	logsig', 'purelin'	traingbr	1	0,0002766	0,002088815
5-9-1	Backpropagation Standard	logsig', 'purelin'	traingd	2406	0,0009997	0,000922011
	One Step Secant (OSS)	logsig', 'purelin'	traingos	7	0,0009,7704	0,0011603
	Bayesian Regulation	logsig', 'purelin'	traingbr	1	0,0004995	0,000549791
5-11-1	Backpropagation Standard	logsig', 'purelin'	traingd	3548	0,0009998	0,000902276
	One Step Secant (OSS)	logsig', 'purelin'	traingos	2	0,00099104	0,0008204
	Bayesian Regulation	logsig', 'purelin'	traingbr	13	0,0004990	0,00056905

Dari Tabel 21 dapat dilihat perbandingan setiap metode fungsi pelatihan yang digunakan. Dari hasil analisa yang dilakukan ada 2 value MSI Pengujian yang paling rendah. Arsitektus 5-9-1 algoritma bayesian regulations dengan jumlah iterasi 1, value MSE Testing/Performances 0,000549791, dan Model 5-11-1 algoritma bayesian Regulations dimana jumlah iterasi 13, value MSE Testing/Performance 0,00056905. Dengan demikian metode Bayesian Regulation terpilih menjadi fungsi pelatihan terbaik karena jumlah epoach dan nilai MSE Testing/Performance lebih rendah dari fungsi pelatihan lainnya.



Gambar. 11 Perbandingan MSE Testing / Performance



Gambar. 12 Perbandingan Epoach/Iterasi

Gambar 11 dan 12 merupakan perbandingan setiap algoritma dan model arsitektur yang digunakan. Sesuai dengan gambar, fungsi pelatihan Bayesian Regulation merupakan hasil yang terbaik dengan model arsitektur jaringan 5-9-1.

IV. KESIMPULAN

Proses pelatihan dan pengujian dalam penelitian ditemukan bahwa metode dan fungsi pelatihan Bayesian Regulation merupakan fungsi pelatihan terbaik dari Backpropagation Standard dan One Step Secant (OSS), dengan demikian fungsi pelatihan Bayesian Regulation dapat digunakan dalam penyelesaian masalah peramalan

kemiskinan di Indonesia. Fungsi pelatihan Bayesian Regulation (*trainbr*) dapat melatih jaringan lebih optimal, dan menghasilkan epoach/performance pelatihan lebih cepat, serta MSE pengujian lebih kecil dari fungsi pelatihan yang lain.

REFERENCES

- [1] I. Waspada, A. Wibowo, and N. S. Meraz, "SUPERVISED MACHINE LEARNING MODEL FOR MICRORNA EXPRESSION DATA IN CANCER," *Jurnal Ilmu Komputer dan Informasi*, vol. 10, no. 2, p. 108, Jun. 2017, doi: 10.21609/jiki.v10i2.481.
- [2] I. Cong, S. Choi, and M. D. Lukin, "Quantum convolutional neural networks," *Nature Physics*, vol. 15, no. 12, pp. 1273–1278, Dec. 2019, doi: 10.1038/s41567-019-0648-8.
- [3] B. S. Rem *et al.*, "Identifying quantum phase transitions using artificial neural networks on experimental data," *Nature Physics*, vol. 15, no. 9, Nature Publishing Group, pp. 917–920, Sep. 01, 2019, doi: 10.1038/s41567-019-0554-0.
- [4] R. Novickis, D. J. Justs, K. Ozols, and M. Greitāns, "An approach of feed-forward neural network throughput-optimized implementation in FPGA," *Electronics (Switzerland)*, vol. 9, no. 12, pp. 1–16, Dec. 2020, doi: 10.3390/electronics9122193.
- [5] L. Yang and A. Shami, "On hyperparameter optimization of machine learning algorithms: Theory and practice," *Neurocomputing*, vol. 415, pp. 295–316, Nov. 2020, doi: 10.1016/j.neucom.2020.07.061.
- [6] C. T. Chen and G. X. Gu, "Generative Deep Neural Networks for Inverse Materials Design Using Backpropagation and Active Learning," *Advanced Science*, vol. 7, no. 5, Mar. 2020, doi: 10.1002/advs.201902607.
- [7] S. Alsammarraie and N. K. Hussein, "A New Hybrid Grasshopper Optimization-Backpropagation for Feedforward Neural Network Training," *Tikrit Journal of Pure Science*, vol. 25, no. 1, p. 2020, doi: 10.25130/tjps.25.2020.018.
- [8] I. T. Sui Kim, V. Sethu, S. K. Arumugasamy, and A. Selvarajoo, "Fenugreek seeds and okra for the treatment of palm oil mill effluent (POME) – Characterization studies and modeling with backpropagation feedforward neural network (BFNN)," *Journal of Water Process Engineering*, vol. 37, Oct. 2020, doi: 10.1016/j.jwpe.2020.101500.
- [9] F. Cichos, K. Gustavsson, B. Mehlig, and G. Volpe, "Machine learning for active matter," *Nature Machine Intelligence*, vol. 2, no. 2, pp. 94–103, Feb. 2020, doi: 10.1038/s42256-020-0146-9.
- [10] R. García-Ródenas, L. J. Linares, and J. A. López-Gómez, "Memetic algorithms for training feedforward neural networks: an approach based on gravitational search algorithm," *Neural Computing and Applications*, vol. 33, no. 7, pp. 2561–2588, Apr. 2021, doi: 10.1007/s00521-020-05131-y.
- [11] E. Yan, J. Song, C. Liu, J. Luan, and W. Hong, "Comparison of support vector machine, back propagation neural network and extreme learning machine for syndrome element differentiation," *Artificial Intelligence Review*, vol. 53, no. 4, pp. 2453–2481, Apr. 2020, doi: 10.1007/s10462-019-09738-z.
- [12] S. R. Salkuti, "A survey of big data and machine learning," *International Journal of Electrical and Computer Engineering*, vol. 10, no. 1, pp. 575–580, 2020, doi: 10.11591/ijece.v10i1.pp575-580.
- [13] I. C. Afolabi, S. I. Popoola, and O. S. Bello, "Modeling pseudo-second-order kinetics of orange peel-paracetamol adsorption process using artificial neural network," *Chemometrics and Intelligent Laboratory Systems*, vol. 203, Aug. 2020, doi: 10.1016/j.chemolab.2020.104053.
- [14] A. Munther, R. Razif, M. AbuAlhaj, M. Anbar, and S. Nizam, "A preliminary performance evaluation of K-means, KNN and em unsupervised machine learning methods for network flow classification," *International Journal of Electrical and Computer Engineering*, vol. 6, no. 2, pp. 778–784, Apr. 2016, doi: 10.11591/ijece.v6i1.8909.
- [15] A. Panyafong, N. Neamsorn, and C. Chaichana, "Heat load estimation using Artificial Neural Network," in *Energy Reports*, Feb. 2020, vol. 6, pp. 742–747, doi: 10.1016/j.egyr.2019.11.149.
- [16] E. W. Prasetyo, N. Hidetaka, D. A. Prasetya, W. Dirgantara, and H. F. Windi, "Spatial Based Deep Learning Autonomous Wheel Robot Using CNN," *Lontar Komputer: Jurnal Ilmiah Teknologi Informasi*, vol. 11, no. 3, p. 167, Dec. 2020, doi: 10.24843/lkji.2020.v11.i03.p05.
- [17] C. K. Arthur, V. A. Temeng, and Y. Y. Ziggah, "Performance Evaluation of Training Algorithms in Backpropagation Neural Network Approach to Blast-Induced Ground Vibration Prediction," *Ghana Mining Journal*, vol. 20, no. 1, pp. 20–33, Jul. 2020, doi: 10.4314/gm.v20i1.3.
- [18] "27 Application of Artificial Neural Networks for Airline Number of Passenger Estimation in time Series state".
- [19] R. Kountchev, S. Patnaik, J. Shi, and M. N. Favorskaya Editors, "Smart Innovation, Systems and Technologies 179." [Online]. Available: <http://www.springer.com/series/8767>
- [20] I. B. K. Sudiatmika, F. Rahman, Trisno, and Suyoto, "Image forgery detection using error level analysis and deep learning," *Telkomnika (Telecommunication Computing Electronics and Control)*, vol. 17, no. 2, pp. 653–659, Apr. 2019, doi: 10.12928/TELKOMNIKA.V17I2.8976.
- [21] M. Žic, V. Subotić, S. Pereverzyev, and I. Fajfar, "Solving CNLS problems using Levenberg-Marquardt algorithm: A new fitting strategy combining limits and a symbolic Jacobian matrix," *Journal of Electroanalytical Chemistry*, vol. 866, Jun. 2020, doi: 10.1016/j.jelechem.2020.114171.
- [22] J. Bilski, B. Kowalczyk, A. Marchlewska, and J. M. Zurada, "Local Levenberg-Marquardt Algorithm for Learning Feedforward Neural Networks," *Journal of Artificial Intelligence and Soft Computing Research*, vol. 10, no. 4, pp. 299–316, Oct. 2020, doi: 10.2478/jaiscr-2020-0020.
- [23] Q. H. Nguyen *et al.*, "A novel hybrid model based on a feedforward neural network and one step secant algorithm for prediction of load-bearing capacity of rectangular concrete-filled steel tube columns," *Molecules*, vol. 25, no. 15, Aug. 2020, doi: 10.3390/molecules25153486.
- [24] P. Panulian *et al.*, "Analysis of Sequential Order Incremental Methods in Predicting the Number of Victims Affected by Disasters," in *Journal of Physics: Conference Series*, Sep. 2019, vol. 1255, no. 1, doi: 10.1088/1742-6596/1255/1/012033.
- [25] M. Attamimi, R. Mardiyanto, and A. N. Irfansyah, "Inclined image recognition for aerial mapping using deep learning and tree based models," *Telkomnika (Telecommunication Computing Electronics and Control)*, vol. 16, no. 6, pp. 3034–3044, Dec. 2018, doi: 10.12928/TELKOMNIKA.v16i6.10157.
- [26] N. L. W. S. R. Ginantra, M. A. Hanafiah, A. Wanto, R. Winanjaya, and H. Okprana, "Utilization of the Batch Training Method for Predicting Natural Disasters and Their Impacts," *IOP Conference Series: Materials Science and Engineering*, vol. 1071, no. 1, p. 012022, Feb. 2021, doi: 10.1088/1757-899x/1071/1/012022.
- [27] IEEE Staff, *2019 Amity International Conference on Artificial Intelligence (AICAI)*. IEEE, 2019.
- [28] I. Fitriyaningsih and Y. Basani, "Flood Prediction with Ensemble Machine Learning using BP-NN and SVM," *Jurnal Teknologi dan Sistem Komputer*, vol. 7, no. 3, pp. 93–97, Jul. 2019, doi: 10.14710/jtsiskom.7.3.2019.93-97.
- [29] R. Jayaseelan, G. Pandulu, and G. Ashwini, "Neural networks for the prediction of fresh properties and compressive strength of flowable concrete," *Journal of Urban and Environmental Engineering*, vol. 13, no. 1, pp. 183–197, 2019, doi: 10.4090/juee.2019.v13n1.183197.
- [30] T. Afriliansyah and Z. Zulfahmi, "Prediction of Life Expectancy in Aceh Province by District City Using the Cyclical Order Algorithm," *International Journal of Information System & Technology Akreditasi*, vol. 3, no. 36, pp. 268–275, 2020.

- [31] S. Mutfrofin, M. Mughniy Machfud, D. H. Satyareni, R. Venantius, H. Ginardi, and C. Fatichah, "KOMPARASI KINERJA ALGORITMA C4.5, GRADIENT BOOSTING TREES, RANDOM FORESTS, DAN DEEP LEARNING PADA KASUS EDUCATIONAL DATA MINING", doi: 10.25126/jtiik.2020732665.
- [32] R. R. Pratama, "Analisis Model Machine Learning Terhadap Pengenalan Aktifitas Manusia," *MATRIK : Jurnal Manajemen, Teknik Informatika dan Rekayasa Komputer*, vol. 19, no. 2, pp. 302–311, May 2020, doi: 10.30812/matrik.v19i2.688.
- [33] M. Tyrtaiou, A. Papaleonidas, A. Elenas, and L. Iliadis, "Accomplished Reliability Level for Seismic Structural Damage Prediction Using Artificial Neural Networks," 2020, pp. 85–98. doi: 10.1007/978-3-030-48791-1_6.
- [34] F. A. Bachtiar, I. K. Syahputra, and S. A. Wicaksono, "PERBANDINGAN ALGORITME MACHINE LEARNING UNTUK MEMPREDIKSI PENGAMBIL MATA KULIAH," vol. 6, no. 5, pp. 543–548, 2019, doi: 10.25126/jtiik.2019611755.
- [35] I. Ferima Talia, I. Fitri Astuti, and P. Studi Ilmu Komputer Fakultas Ilmu Komputer dan Teknologi Informasi, "Peramalan Tingkat Kemiskinan Penduduk Provinsi Kalimantan Timur Menggunakan Metode Double Exponential Smoothing Zainal Arifin," *Prosiding Seminar Nasional Ilmu Komputer dan Teknologi Informasi*, vol. 4, no. 2, 2019.
- [36] N. Choiriyati, Y. Arkeman, and W. A. Kusuma, "Deep learning model for metagenome fragment classification using spaced k-mers feature extraction," *Jurnal Teknologi dan Sistem Komputer*, vol. 8, no. 3, pp. 234–238, Jul. 2020, doi: 10.14710/jtsiskom.2020.13407.
- [37] Sri Sivasubramaniya Nadar College of Engineering. Electronics and Communication Engineering Department and Institute of Electrical and Electronics Engineers, *WiSPNET 2019 : 2019 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET) : 21-23, March 2019*.
- [38] I. D. Uwanuakwa and P. Akpinar, "Investigations on the influence of variations in hidden neurons and training data percentage on the efficiency of concrete carbonation depth prediction with ann," in *Advances in Intelligent Systems and Computing*, 2020, vol. 1095 AISC, pp. 958–965. doi: 10.1007/978-3-030-35249-3_128.
- [39] K. v. Shende, M. R. Ramesh Kumar, and K. v. Kale, "Comparison of Neural Network Training Functions for Prediction of Outgoing Longwave Radiation over the Bay of Bengal," in *Advances in Intelligent Systems and Computing*, 2020, vol. 1025, pp. 411–419. doi: 10.1007/978-981-32-9515-5_39.
- [40] S. Setti and A. Wanto, "Analysis of Backpropagation Algorithm in Predicting the Most Number of Internet Users in the World," *Jurnal Online Informatika*, vol. 3, no. 2, pp. 110–115, 2018, doi: 10.15575/join.

ORIGINALITY REPORT

8%

SIMILARITY INDEX

7%

INTERNET SOURCES

4%

PUBLICATIONS

3%

STUDENT PAPERS

PRIMARY SOURCES

1

jurnal.iaii.or.id

Internet Source

3%

2

Submitted to Forum Perpustakaan Perguruan
Tinggi Indonesia Jawa Timur

Student Paper

2%

3

gis.depkes.go.id

Internet Source

1%

4

R. Meenal, A. Immanuel Selvakumar.
"Assessment of SVM, empirical and ANN
based solar radiation prediction models with
most influencing input parameters",
Renewable Energy, 2018

Publication

<1%

5

Submitted to Universitas Esa Unggul

Student Paper

<1%

6

jurnal.untan.ac.id

Internet Source

<1%

7

djournals.com

Internet Source

<1%

8	Submitted to Universitas Bina Darma Student Paper	<1 %
9	jurnal.unimed.ac.id Internet Source	<1 %
10	journal.untar.ac.id Internet Source	<1 %
11	Submitted to Universitas Diponegoro Student Paper	<1 %
12	repository.ipb.ac.id Internet Source	<1 %
13	www.scribd.com Internet Source	<1 %
14	www.coursehero.com Internet Source	<1 %
15	N L W S R Ginantra, Gita Widi Bhawika, GS Achmad Daengs, Pawan Darasa Panjaitan et al. "Performance One-step secant Training Method for Forecasting Cases", Journal of Physics: Conference Series, 2021 Publication	<1 %
16	digitalcommons.fiu.edu Internet Source	<1 %
17	Anjar Wanto. "PREDIKSI PRODUKTIVITAS JAGUNG DI INDONESIA SEBAGAI UPAYA ANTISIPASI IMPOR MENGGUNAKAN	<1 %

JARINGAN SARAF TIRUAN BACKPROPAGATION", SINTECH (Science and Information Technology) Journal, 2019

Publication

18

eprints.dinus.ac.id

Internet Source

<1 %

19

id.scribd.com

Internet Source

<1 %

20

repository.its.ac.id

Internet Source

<1 %

Exclude quotes On

Exclude matches Off

Exclude bibliography On

Hasil_Cek_03

PAGE 1

PAGE 2

PAGE 3

PAGE 4

PAGE 5

PAGE 6

PAGE 7

PAGE 8

PAGE 9

PAGE 10

PAGE 11

PAGE 12

PAGE 13

PAGE 14

PAGE 15

PAGE 16