PRISMA 8 (2025): 359-369
PRISMA, Prosiding Seminar Nasional Matematika
https://journal.unnes.ac.id/sju/index.php/prisma/
ISSN 2613-9189

Penggunaan Backpropagtion Neural Network Untuk Memprediksi Ketersediaan Pangan Di Kota Semarang

Adiwidyanto Nugroho[a,*], Walid[b]

[a] *Mahasiswa, Unnes, Semarang 50229, Indonesia*
[b] *Dosen, Unnes, Semarang 50229, Indonesia*

** adi.widyanton@students.unnnes.ac.id*

Abstrak

Artikel ini bertujuan untuk merancang dan menguji sistem jaringan saraf tiruan menggunakan algoritma backpropagation dalam memprediksi ketersediaan pangan di Kota Semarang. Metode yang digunakan melibatkan pengumpulan data dari Badan Pusat Statistik, studi pustaka, dan implementasi algoritma neural network backpropagation dengan parameter yang dioptimalkan melalui Matlab. Arsitektur terbaik yang ditemukan adalah 12-17-1 dengan fungsi aktivasi logsig dan purelin. Hasil penelitian menunjukkan kinerja prediksi yang baik dengan nilai MSE sebesar 0,015243 dan MAE sebesar 0,106222, menandakan akurasi prediksi yang tinggi. Kesimpulannya, model ini efektif dalam memprediksi ketersediaan pangan dengan tingkat kesalahan yang rendah.

Kata kunci:

Ketersediaan Pangan, Jaringan Saraf Tiruan, Backpropagation, Prediksi, Neural Network, Kota Semarang, Matlab, MSE, MAE, Arsitektur Jaringan.

© 2025 Dipublikasikan oleh Jurusan Matematika, Universitas Negeri Semarang

1. Pendahuluan

Pangan atau makanan adalah bahan mentah maupun olahan yang dikonsumsi oleh manusia atau hewan untuk memenuhi kebutuhan pertumbuhan, kesehatan, serta kepuasan. Secara kimiawi, makanan sebagian besar terdiri dari air, lemak, protein, dan karbohidrat, serta mengandung mineral dan senyawa organik dalam jumlah kecil. Mineral tersebut mencakup garam, sementara senyawa organik meliputi vitamin, asam, perasa, pengemulsi, dan lainnya.

Ketersediaan pangan yang cukup dan merata menjadi salah satu elemen utama dalam mewujudkan ketahanan pangan yang stabil dan berkelanjutan. Hal ini sesuai dengan Undang-Undang No. 7 Tahun 1996 tentang pangan serta penjelasan dalam PP No. 68 Tahun 2002. Data mengenai ketahanan pangan sangat penting sebagai indikator dalam menilai keberhasilan pembangunan di sektor ini serta sebagai alat prediksi untuk mendeteksi potensi permasalahan dan upaya perbaikannya. Definisi ini sejalan dengan pandangan FAO dan WHO pada tahun 1992, yang menyatakan bahwa ketahanan pangan berarti setiap individu atau rumah tangga memiliki akses terhadap pangan yang cukup setiap saat guna menjalani hidup sehat.

Artificial Neural Network (ANN) atau Jaringan Syaraf Tiruan (JST) adalah sistem pemrosesan informasi yang menyerupai cara kerja sistem saraf manusia. ANN mampu menyelesaikan permasalahan yang dihadapi oleh metode seperti SVM dan KNN dengan melakukan pelatihan menggunakan data dalam jumlah besar. Keunggulan ANN terletak pada kemampuannya menoleransi kesalahan, sehingga dapat menghasilkan prediksi yang lebih akurat. Selain itu, metode ini efektif dalam memodelkan hubungan kompleks antara input dan output untuk mengidentifikasi pola dalam data. Namun, kelemahan ANN adalah sulitnya menentukan jumlah neuron serta lapisan yang optimal, serta cenderung mengalami perlambatan dalam proses pembelajaran.

To cite this article:
Nugroho, A., Walid., (2025). Penggunaan Backpropagation Neural Network Untuk Memprediksi Ketersediaan Pangan Di Kota Semarang. *PRISMA, Prosiding Seminar Nasional Matematika 8*, 359-369

Dalam metode ANN, terdapat berbagai algoritma yang telah diterapkan, salah satunya adalah backpropagation. Algoritma ini merupakan salah satu prosedur yang paling populer, efektif, dan mudah dipahami dalam jaringan multilayer yang kompleks untuk mengoptimalkan pelatihan jaringan saraf tiruan.

Backpropagation menggunakan pendekatan pembelajaran terbimbing (supervised learning) pada jaringan multi-layer yang memiliki beberapa hidden layer, dengan tujuan utama meminimalkan kesalahan dalam menghasilkan keluaran (output). Proses pelatihan ini memanfaatkan fungsi pelatihan (training functions) dengan variabel laju pemahaman (traingdx) untuk mempercepat proses pembelajaran. Traingdx merupakan kombinasi antara parameter laju pemahaman (learning rate) dan momentum, yang membantu dalam memperoleh hasil yang lebih akurat. (Putra & Ulfa Walmi, 2020).

Permasalahan yang dibahas dalam penelitian ini adalah merancang suatu sistem jaringan saraf tiruan menggunakan algoritma backpropagation untuk memprediksi ketersedian pangan dan menguji sistem tersebut. Adapun point rumusan masalah yang akan diselesaikan adalah, Bagaimana arsitektur terbaik neural network backpropagation untuk memprediksi ketersediaan pangan di Kota Semarang, dan Bagaimana hasil prediksi dengan model neural network yang telah ditentukan untuk memprediksi ketersediaan pangan di Kota Semarang. Tujuan penulisan artikel ini adalah untuk merancang dan menguji sistem jaringan saraf tiruan menggunakan algoritma backpropagation untuk memprediksi ketersediaan pangan

2. Metode

2.1. Neural Network Backpropagation

Neural Network adalah metode kecerdasan buatan yang meniru operasi dan kinerja komputasi otak manusia, dan memiliki kemampuan untuk mencerminkan hubungan linier atau non-linier yang mendasari antara masukan dan data target. Memperoleh pengetahuan dari sumber luar, Neural Network menyimpan informasi di unit pemrosesan dalam dan mengirimkannya melalui interaksi fungsi transfer dan parameter koneksi antara lapisan yang berdekatan.

Backpropagation, atau propagation of error, adalah metode umum untuk mengajarkan jaringan saraf tiruan bagaimana melakukan tugas tertentu. BPNN mengacu pada jaringan saraf tiruan feed-forward multilayer dengan proses pembelajaran propagasi balik, yang dianggap sebagai arsitektur jaringan yang paling banyak digunakan. Secara garis besar, training jaringan dengan metode backpropagation meliputi tiga tahap :

1. Tahap Maju

   Tahap feedforward yang dimaksud adalah proses pengolahan input dari pola input training pada input layer sampai respon yang dihasilkan mencapai output layer. Nilai pada tiap unit masukan diteruskan ke unit tersembunyi dan dihitung dengan

   $$Z_{net_k} = V_{j0} + \sum_{i=1}^{n} X_i V_{ji} \quad (1)$$

   $$Z_j = f\left(Z_{net_k}\right) \quad (2)$$

   dimana Z_j adalah nilai pada unit tersembunyi. Z_j yang didapat dari penghitungan fungsi aktivasi yang ditentukan. Bias V_{j0} merupakan bobot bias di unit masukan ke unit tersembunyi Z_j, masukan X_i adalah masukan pada tiap X_i dan V_{ji} adalah bobot yang menghubungkan X_i dan Z_j. Setelah unit pada layer tersembunyi memiliki nilai, maka akan diteruskan ke unit tersembunyi selanjutnya, jika ada, atau ke unit keluaran. Nilai pada unit keluaran dihitung dengan,

   $$Y_{net_k} = W_{k0} + \sum_{j=1}^{p} Z_j W_{kj} \quad (3)$$

   $$Y_k = f\left(Y_{net_k}\right) \quad (4)$$

dimana W_{k0} merupakan bobot bias di unit tersembunyi ke unit keluaran $W_k\ dan\ W_{kj}$ adalah bobot yang menghubungkan $Z_j\ dan\ Y_k$.

2. Tahap Perhitungan Error

Respons yang dihasilkan pada lapisan keluaran (output layer) akan dibandingkan dengan target yang diharapkan, lalu dihitung selisih kesalahannya (error). Jika kriteria penghentian (stopping condition) belum tercapai, proses akan dilanjutkan ke tahap penyesuaian bobot dan bias (adjustment of weights and biases). Namun, jika kriteria tersebut sudah terpenuhi, seluruh proses pelatihan akan dihentikan. Dilakukan penghitungan selisih yang ada antara keluaran jaringan, Y_k dengan target T_k yang diinginkan dimulai dari unit-unit di layer keluaran sebagai berikut,

$$\delta_k = (T_k - Y_k) f'\left(Y_{net_k}\right) \tag{5}$$

faktor δ_k merupakan unit kesalahan di setiap unit keluaran Y_k. Setelah itu, hitung perubahan bobot pada W_{kj} menggunakan,

$$\Delta W_{kj} = \alpha \delta_k Z_j \tag{6}$$

dengan α adalah laju pembelajaran (learning rate). Nilai α berada antara 0 dan 1. Semakin besar nilai laju pembelajaran, maka semakin cepat proses pelatihannya. Sedangkan untuk menghitung perubahan bobot bias menggunakan,

$$\Delta W_{k0} = \alpha \delta_k \tag{7}$$

nilai δ_k selanjutnya dikirim ke layer di bawahnya, yaitu layer tersembunyi, dan digunakan untuk menghitung unit kesalahan dengan perhitungan,

$$\delta_{net_j} = \sum_{k=1}^{m} \delta_k W_{kj} \tag{8}$$

$$\delta_j = \delta_{net_j} f'\left(Z_{net_j}\right) \tag{9}$$

faktor δ_j merupakan unit kesalahan di setiap unit tersembunyi Z_j. Setelah nilai unit kesalahan didapat, maka akan diteruskan ke perhitungan faktor δ dari unit kesalahan layer tersembunyi di bawahnya, jika ada, atau unit masukan. Pada unit masukan, faktor δ digunakan untuk menghitung perubahan bobot unit masukan.

$$\Delta V_{ji} = \alpha \delta_j X_i \tag{10}$$

perubahan bobot bias dihitung dengan menggunakan,

$$\Delta V_{j0} = \alpha \delta_j \tag{11}$$

3. Tahap Pembaharuan Bobot dan Bias

Kondisi ini muncul ketika keluaran yang diharapkan tidak sesuai dengan target. Jaringan kemudian akan melakukan propagasi mundur (backward propagation) dari lapisan keluaran (output layer) menuju lapisan masukan (input layer), memperbarui nilai bobot dan bias, serta mengulangi proses propagasi maju (forward pass). Penyesuaian bobot bertujuan mengurangi kesalahan (error) yang muncul selama pelatihan. Perubahan bobot menuju unit keluaran dihitung dengan,

$$W_{kj}(baru) = W_{kj}(lama) + \Delta W_{kj} \tag{12}$$

untuk

$k = 1,2, \dots, m$ dan $j = 0,1,2, \dots, p$

dengan cara yang sama, perubahan bobot yang menuju unit tersembunyi didefinisikan

$$V_{ji}(baru) = V_{ji}(lama) + \Delta V_{ji} \quad (13)$$

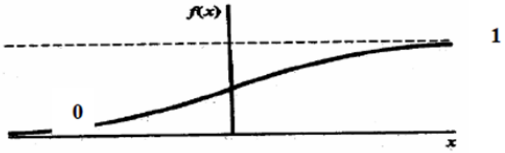
untuk

$j = 1,2, \dots, p$ dan $i = 0,1,2, \dots, n$

Setelah proses pelatihan selesai, jaringan dapat digunakan untuk proses pengujian. Pada proses pengujian, dilakukan perhitungan propagasi maju dengan bobot akhir yang dihasilkan oleh jaringan.

4. Fungsi Aktivasi

Fungsi aktivasi yang digunakan perlu memenuhi beberapa kriteria, yakni bersifat kontinu, mudah diturunkan (diferensiasi), dan memiliki sifat non-decreasing (tidak menurun). Penelitian ini mengadopsi fungsi aktivasi sigmoid biner. Seperti terlihat pada Gambar 2.1, fungsi sigmoid biner menghasilkan nilai keluaran dalam rentang 0 hingga 1,



Gambar 1. Grafik Sigmoid Biner

Karena itu, fungsi ini sering digunakan untuk jaringan syaraf yang membutuhkan nilai keluaran yang terletak pada interval 0 sampai 1. Fungsi *sigmoid biner* dirumuskan sebagai berikut :

$$y = f(x) = \frac{1}{1+e^{-\sigma x}} \quad (14)$$

$$y = f(x) = \sigma f(x)[1 - f(x)] \quad (15)$$

Untuk menormalisasikan/transformasi data dan nilai agar dapat bernilai 0 hingga 1, dapat menggunakan rumus sebagai berikut :

$$x = \frac{0,8(x-a)}{b-a} + 0,1 \quad (16)$$

5. Pengujian Eror

Setelah dilakukan proses training dan testing pola-pola yang dilatih, maka akan diperoleh hasil bahwa pengujian terhadap pola-pola tersebut apakah telah benar/akurat atau sebaliknya. Untuk menghitung rata-rata error jaringan, dapat dilakukan dengan dua cara yaitu RMSE (*Root Mean Squared Error*) dan MAPE (*Mean Avarage Percentage Error*). RMSE adalah fungsi kinerja yang umumnya digunakan untuk *backpropagation* dimana fungsi ini akan mengambil rata-rata kuadrat *error* yang terjadi antara *output* jaringan dan target, untuk perhitungan rata-rata error jaringan menggunakan RMSE dapat dilakukan dengan persamaan berikut,

$$RMSE = \sum_{i=1}^{n} \frac{(y_i - y_n)^2}{N} \quad (17)$$

dimana,

y_i = Nilai aktual data (*target*)

y_n = Nilai hasil prediksi (*actual output*)

N = jumlah data yang diujikan

Sedangkan untuk proses denormalisasi atau mengembalikan kembali nilai hasil prediksi jaringan ke bentuk data semula (sebelum normalisasi) dapat menggunakan persamaan berikut,

$$x_i = y_n(x_{max} - x_{min}) + x_{min} \quad (18)$$

dimana,

x_i = Nilai x yang akan dilakukan denormalisasi

y_n = nilai hasil prediksi (*actual output*) yang sesuai dengan x_i

x_{max} = nilai maksimum pada barisan x

x_{min} = nilai minimum pada barisan x

Sedangkan MAPE (Mean Absolute Percentage Error) dihitung dengan membagi kesalahan mutlak pada setiap periode dengan nilai aktual yang diamati dalam periode tersebut. Semakin kecil nilai MAPE, semakin baik kinerja prediksi yang dihasilkan. Untuk menentukan tingkat akurasi dari prediksi yang dilakukan, akurasi dihitung dengan mengurangkan nilai MAPE dari 100%,

$$MAPE = \frac{\sum_{t-1}^{n} \frac{|x_t - y_t|}{x_t}}{n} \times 100\% \quad (19)$$

$$Akurasi = 100\% - MAPE \quad (20)$$

dimana,

x_t = nilai target data asli pada periode t

y_t = nilai prediksi pada periode t

n = banyaknya data prediksi pada periode t

2.2. MATLAB

MATLAB (Matrix Laboratory) adalah sebuah bahasa pemrograman tingkat tinggi yang secara khusus digunakan untuk komputasi numerik, pemrograman, dan visualisasi. Perusahaan yang bertanggung-jawab atas produk hasil pengembangan dari MATLAB ialah MathWorks. Fungsi utama dari MATLAB ialah untuk melakukan analisis data, mengembangkan algoritme, serta membuat model dan aplikasi.

MATLAB ditemukan oleh ahli matematika dan pemrogram komputer Cleve Moler. Ide MATLAB didasarkan pada tesis PhD-nya pada tahun 1960an. Pada awalnya (sebelum versi 1.0) MATLAB "bukanlah bahasa pemrograman, dia adalah kalkulator matriks interaktif sederhana. Tidak ada program, tidak ada kotak alat, tidak ada grafik". Versi awal pertama MATLAB selesai pada akhir tahun 1970-an. Perangkat lunaknya diungkapkan kepada publik untuk pertama kalinya pada bulan Februari 1979 di Sekolah Pascasarjana Angkatan Laut di California.

3. Hasil dan Pembahasan

Dari penelitian ini metode yang digunakan adalah Jaringan Syaraf Tiruan dengan Algoritma model Backpropagation, menggunakan arsitektur 12-17-1, dengan jumlah epoch 250. Fungsi pelatihan yang digunakan adalah traingd atau Gradient Desecents, fungsi pelatihan ini berbasis gradien yang mencari minimum fungsi kesalahan dengan mengikuti gradien atau arah yang berlawanan dari gradien. Arsitektur ryang dilatih dengan algoritma gradien dapat memberikan hasil yang baik terutama pada masalah yang cukup sederhana dan dataset yang relatif kecil.

3.1. Arsitektur

Sebelum dilakukan prediksi data, dilakukan terlebih dahulu perancangan model arsitektur algoritma backpropagation yang akan digunakan. Dalam pengembangan jaringan saraf tiruan, pemilihan arsitektur

yang tepat adalah langkah penting yang memengaruhi kinerja dan kemampuan jaringan dalam memodelkan hubungan antara input dan output.

1. Hidden Layer
   Hidden layer (lapisan tersembunyi) adalah salah satu komponen utama dalam arsitektur jaringan saraf tiruan (neural network). Lapisan ini terletak di antara lapisan input dan lapisan output, dan berperan dalam mengekstraksi fitur-fitur yang relevan dari data input. Penentuan jumlah hidden layer merupakan salah satu keputusan utama dalam merancang arsitektur jaringan saraf tiruan. Pada kasus penelitian ini jumlah hidden layer yang akan digunakan sebanyak 17 neuron, alasan penulis menggunakan 17 neuron hidden layer dikarenakan jumlah ini yang cukup baik untuk memprediksi data yang dipakai untuk penelitian ini.
2. Fungsi Aktivasi
   Setiap neuron dalam hidden layer dan output layer memerlukan fungsi aktivasi non-linear. Fungsi ini memungkinkan jaringan untuk mempelajari relasi yang kompleks antara input dan output. Pada penelitian ini penulis menggunakan fungsi aktivasi sigmoid logsig untuk input dan purelin untuk output. Fungsi aktivasi logsig merupakan versi dari fungsi sigmoid biner, yang menghasilkan output dalam rentang 0 hingga 1 dan sering digunakan sebagai fungsi aktivasi dalam jaringan saraf tiruan, terutama untuk klasifikasi biner. Fungsi ini penting karena memperkenalkan non-linearitas ke dalam jaringan, memungkinkan model untuk mempelajari pola yang lebih kompleks daripada model linier. Fungsi aktivasi purelin adalah fungsi identitas linear yang sederhana, yang beroperasi tanpa transformasi non-linear apa pun. Karena sifatnya yang linear, fungsi purelin umumnya digunakan pada lapisan output untuk masalah regresi, di mana output yang diinginkan dapat berupa nilai kontinu.
3. Fungsi Pelatihan
   Pemilihan algoritma pelatihan adalah salah satu bagian penting dari perancangan model arsitektur jaringan saraf tiruan yang menggunakan algoritma backpropagation. Pada penelitian ini penulis menggunakan fungsi pelatihan Gradient Decents untuk melakaukan prediksi. Prinsip kerja Gradient Decents pada umumnya setiap iterasi pelatihan, algoritma ini memperbarui bobot dan bias jaringan dengan mengikuti arah yang menurunkan nilai fungsi kesalahan (loss function). Gradien dari fungsi kesalahan terhadap setiap bobot dan bias dihitung, dan kemudian bobot dan bias diperbarui dengan mengurangi gradien tersebut, dikalikan dengan laju pembelajaran (learning rate).

3.2. Pre-Processing

Data yang digunakan dalam penelitian ini adalah data yang diambil dari Badan Pusat Statistik Kota Semarang dari tahun 2018-2022. Data tersebut di input ke dalam excel dengan format tahun produksi dalam kolom dan bulan produksi dalam baris.

Tabel 1. Tabel Produksi Beras

Tahun	2018 (Ton)	2019 (Ton)	2020 (Ton)	2021 (Ton)	2022 (Ton)
Januari	1.093	635	351	935	1.315
Februari	1.677	625	211	1.018	321
Maret	905	910	1.100	1.805	513
April	1.316	1.538	1.936	1.019	1.391
Mei	1.376	2.363	2.446	751	1.418
Juni	1.287	867	836	604	716
Juli	1.573	404	586	1.688	280
Agustus	951	1.665	1.183	556	249
September	1.449	1.627	2.018	1.054	505
Oktober	783	633	1.750	400	1.412
November	1.002	805	287	760	460
Desember	466	735	439	1.076	161

Fungsi normalisasi yang digunakan untuk mengolah data latih (training) dan uji (testing) diatas adalah fungsi sigmoid biner, maka data harus ditransformasikan terlebih dahulu karena batas (range) keluaran fungsi aktivasi sigmoid adalah fungsi asimtotik (tidak pernah mencapai 0 ataupun 1). Data ditransformasikan ke interval yang lebih kecil, misalnya pada interval [0,1;0,8],

$$x = \frac{0{,}8(x-a)}{b-a} + 0{,}1 \quad (21)$$

Tabel 2.

Data Training											
0.3814	0.6199	0.3046	0.4724	0.4969	0.4606	0.5774	0.3234	0.5267	0.2548	0.3442	0.1253
0.1943	0.1903	0.3066	0.5631	0.9000	0.2891	0.1000	0.6150	0.5994	0.1935	0.2638	0.2352

Tabel 3.

Data Target Training											
0.4157	0.6247	0.3484	0.4955	0.5170	0.4851	0.5875	0.3649	0.5431	0.3047	0.3831	0.1913

Tabel 4.

Data Target Testing											
0.5535	0.8377	0.4620	0.6620	0.6912	0.6479	0.7871	0.4844	0.7268	0.4027	0.5092	0.2484
0.3307	0.3258	0.4645	0.7701	1.1715	0.4436	0.2182	0.8319	0.8134	0.3297	0.4134	0.3793

3.3. Prediksi Data

Proses awal yang dilakukan untuk memprediksi data dengan membuat tab editor baru di matlab, lalu memasukan kode perintah yang akan digunakan untuk melakukan prediksi data menggunakan algoritma *backpropagation*.

Baris ini yang membuat dan melatih jaringan syaraf tiruan menggunakan fungsi feedforwardnet, angka di dalam kurung merupakan hidden layer dari jaringan sayarf tiruan tersebut dalam kasus ini hidden layer yang digunakan adalah 17 hidden layer.

```
net = feedforwardnet(17);
```

Baris berikutnya di tentukan pemilihan algoritma fungsi pelatihan menggunakan trainFcn dengan algoritma traingd.

```
net.trainFcn = 'traingd';
```

Dua baris selanjutnya dilakukan penetuan iterasi pelatihan (epoch) dan laju pembelajaran (learning rate) mengunkan sub fungsi trainParam. Epoch yang dipilih pada kasus ini sebanyak 250 iterasi, Laju pembelajaran yang dipilih untuk kasus ini adalah 0.1, nilai laju pembelajaran yang dipilih akan memengaruhi seberapa besar langkah yang diambil oleh algoritma pelatihan saat menyesuaikan bobot jaringan

```
net.trainParam.epochs = 250;

net.trainParam.lr = 0.1;
```

Baris selanjutnya melatih jaringan saraf tiruan dengan menggunakan data training (data_norm dan dataT_norm). Fungsi train digunakan untuk melatih jaringan, dengan argumen berupa jaringan (net) yang sudah dibuat sebelumnya, data input (data_norm), dan target output (dataT_norm).

net = train(net, data_norm, dataT_norm);

Fungsi sim menerima dua argumen yaitu, jaringan yang sudah dilatih (net) dan data input untuk dimasukkan ke dalam jaringan (dataTe_norm).

outputTestData = sim(net, dataTe_norm);

3.4. Post-Processing Data

Setelah dilakukan prediksi data, maka akan didapatkan nilai output atau hasil prediksi, dimana data awal sebagai data input telah dilakukan proses normalisasi atau transformasi.

Tabel 5.

Hasil Prediksi Saraf Tiruan	
Bulan	Hasil
Januari	0,539817593
Februari	0,644483866
Maret	0,46745118
April	0,582985474
Mei	0,164017356
Juni	0,538676424
Juli	0,664026354
Agustus	0,303725242
September	0,622119759
Oktobet	0,368266968
November	0,515603148
Desember	0,026588711

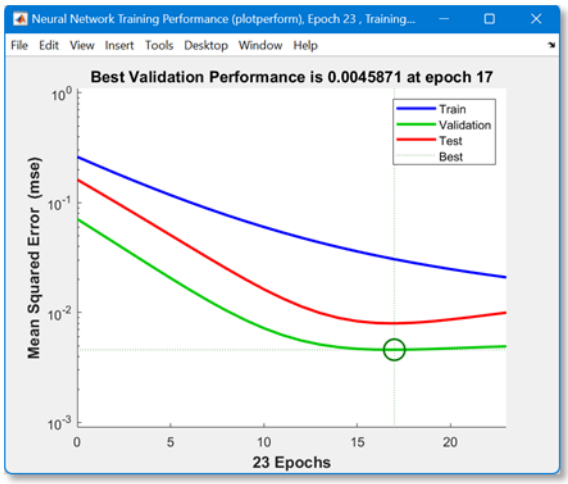
Hasil Prediksi Denormalisasi	
Bulan	Hasil (Ton)
Januari	518,6150396
Februari	781,9109902
Maret	303,6010607
April	319,4893917
Mei	984,1855025
Juni	199,4094857
Juli	904,4045982
Agustus	729,3032774
September	519,7318429
Oktobet	243,7721508
November	786,8504593
Desember	107,6694904

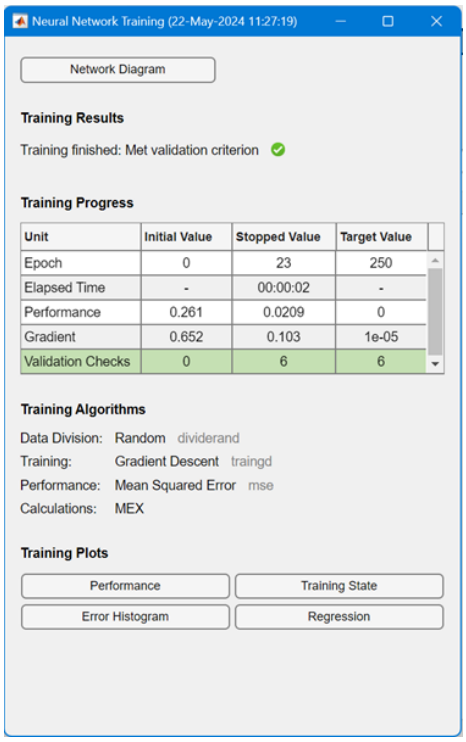
Jadi untuk data prediksi yang baru dihasilkan perlu dilakukan proses denormalisasi. menghitung denormalisasi data prediksi yang telah dilakukan sebelumnya menggunakan rumus

$$X = \frac{(X' - 0{,}1)X_{max} - X_{min}}{0{,}8} + X_{min} \quad (22)$$

3.5. Hasil Kinerja

Berdasarkan hasil pengujian sistem terhadap rancangan model dengan parameter-parameter yang telah ditentukan, diperoleh konfigurasi yang menghasilkan nilai error paling kecil serta akurasi tertinggi. Parameter optimal tersebut diperoleh dengan jumlah epoch sebesar 250, menggunakan arsitektur 12-17-1 dan *learning rate* sebesar 0,1. Model ini juga mencapai kinerja terbaik dengan nilai kinerja sebesar 0.0045871 pada *epoch* ke-17 selama proses pelatihan.





Dari hasil perhitungan error didapat hasil MSE sebesar 0.015243, MAE sebesar 0.106222, RMSE sebesar 0.123462, dan MAPE sebesar 24.626556%, yang dapat diartikan keakurasian prediksi dari jaringan saraf tiruan ini sebesar 75,373444%.

4. Simpulan

Berdasarkan hasil dan pembahasan penelitian yang telah dilakukan, tentang prediksi pangan di kota Semarang menggunakan algoritma neural network backpropagation dapat ditarik denga kesempilan sebagai berikut, (1) Arsitektur terbaik neural network backpropagation untuk memprediksi ketersedian pangan di Kota Semarang menggunakan arsitektur 12-17-1, yang dimana terdiri atas 12 input layer, 17 hidden layer, dan 1 output layer. Fungsi aktivasi yang digunakan dalam penelitian ini adalah logsig dan purelin. Fungsi pelatihan yang digunakan dalam penelitian ini adalah Gradient Decents dengan alasan cukup efektif dan andal, dan juga fleksibelitas kendali penuh atas parameter-parameter seperti laju pembelajaran (learning rate) dan jumlah iterasi dalam algoritma. 10 data yang diuji dinormalisasikan terlebih dahulu lalu di prediksi di dalam aplikasi matlab. Proses prediksi dilakukan dengan menentukan penentuan rasio data training dan testing sebesar 80% dan 20% masing-masing, penentuan hidden layer sebanyak 17, penentuan fungsi pelatihan yaitu Gradient Decents, besarnya epoch sebanyak 250 iterasi, dan laju pembelajaran sebanyak 0,01. (2) Hasil prediksi dengan model neural network yang telah ditentukan untuk memprediksi ketersediaan pangan di Kota Semarang ini dilakukan juga perhitungan metrik evaluasi kinerja model seperti MSE, MAE, RMSE, dan MAPE. Hasil dari penelitian ini didapatkan 10 data prediksi dari 10 variabel data yang diuji. Hasil kinerja prediksi yang didapat sebesar 1.5635e-05 atau 0.000015635 pada epoch 250 dan grafik regresi mendekati 1 menunjukkan hubungan linier yang kuat antara variabel prediktor dan variabel respons. Dari hasil perhitungan eror didapat hasil MSE 0.015243, MAE 0.106222, RMSE 0.123462, dan MAPE 24.626556% yang cukup rendah dan dapat diartikan keakurasian prediksi dari jaringan saraf tiruan ini cukup tinggi.

Daftar Pustaka

(Daftar Pustaka menggunakan format seperti pada contoh. Seluruh penulis pustaka yang dikutip harus ditulis, baik itu pustaka yang ditulis oleh satu orang, dua orang, tiga orang, atau lebih. Pustaka diurutkan berdasarkan abjad pertama penulis).

(Gunakan style PRISMA-DaftarPustaka)

(Contoh penulisan referensi dari jurnal atau artikel ilmiah)

Bernardo, A. B. I. (2002). Language And Mathematical Problem Solving Among Bilinguals. *The Journal of Psychology, 136*(3), 283-297.

Campanario. J. M. (2009). The Parallelism Between Scientists' and Students' Resistance to New scientific ideas. *International Journal of Science Education, 24*(10), 1095-1110.

Asiala, M., Mathews, D., Morics, S., Oktac, A., & Dubinsky, E., (1997). Development of students understanding of cosets, normality and quotient groups, *Journal of Mathematical Behavior, 16*(4), 241-309.

Light, M. A., & Light, I. H. (2008). The geographic expansion of Mexican immigration in the United States and its implications for local law enforcement. *Law Enforcement Executive Forum Journal, 8*(1), 73-82.

(Contoh penulisan referensi dari skripsi / tesis / disertasi)

Luka, M.T. (2013). Misconceptions and Errors in Algebra at Grade 11 Level: The Case of Two Selected Secondary Schools In Petauke District. (*Doctoral Dissertation*). University of Zambia. Lusaka, Zambia.

Maryono. (2008). Eksplorasi Pemahaman Mahasiswa Mengenai Konsep Keterbagian Bilangan Bulat (Exploration of student understanding of the integers division). (*Master's Thesis*). Universitas Malang. Malang.

Tziritas, M. (2011). APOS Theory as a Framework to Study the Conceptual Stages of Related Rates Problems. (*Master's Thesis*). Concordia University Montreal. Montreal, Canada

(Contoh penulisan referensi artikel dalam suatu prosiding)

Liliasari. (2007). Scientific Concepts and Generic Science Skills Relationship In The 21st Century Science Education. In *Proceeding of The First International Seminar of Science Education*. Bandung.

Marpaung, Y. (2011). PMRI and Metacognitive Scaffolding. In *Proceeding International Seminar and The Fourth National Conference on Mathematics Education 2011 Yogyakarta State University*. Yogyakarta

Kurniasih, A. W. (2015). Budaya Mengembangkan Soal Cerita Kontekstual Open-Ended Mahasiswa Calon Guru Matematika untuk Meningkatkan Berpikir Kritis (Culture Developing Contextual Stories Open-Ended Student Candidates for Mathematics Teachers to Improve Critical Thinking). In Zaenuri (Chair). *Proceeding National Seminar of Mathematics IX Universitas Negeri Semarang*. Semarang.

Stewart, S., & Thomas, M. O. (2007). Embodied, symbolic and formal aspects of basic linear algebra concepts. In *Proceedings of the 31st Conference of the International Group for the Psychology of Mathematics Education* (Vol. 4, pp. 201-208). Seoul, Korea.

(Contoh penulisan referensi dari laporan penelitian)

Ivan, A. H. (2005). *Desain target optimal*. (Laporan Penelitian Hibah Bersaing). Jakarta: Dirjen Dikti Kemendikbud.

(Contoh penulisan referensi dari peraturan pemerintah dan produk hukum)

Undang-Undang Republik Indonesia Nomor 20 Tahun 2003 Tentang Sistem Pendidikan Nasional

Peraturan Menteri Pendidikan dan Kebudayaan Republik Indonesia Nomor 60 Tahun 2011 Tentang Larangan Pungutan Biaya Pendidikan pada Sekolah Dasar dan Sekolah Menengah Pertama

Peraturan Gubernur Jawa Tengah Nomor 17 Tahun 2015 Tentang Perubahan Kedua atas Peraturan Gubernur Jawa Tengah Nomor 39 Tahun 2008 Tentang Organisasi dan Tata Kerja Unit Pelaksana Teknis Pada Dinas Pendidikan Provinsi Jawa Tengah

(Contoh penulisan pustaka dari internet/online)

Haylock, D. 1997. Recognising Mathematical Creativity in Schoolchildren. (*Online*). (http://www.emis.de/journals/ZDM973a2.pdf, diakses 3 Maret 2010).

Chipperfield, B. 2004. Cognitive Load Theory and Instructional Design. (*Online*). (http://www.usask.ca/education/coursework/802papers/chipperfield/index.htm, diakses 14 September 2011).