

Aplikasi Model *Backpropagation Neural Network* Untuk Perkiraan Produksi Tebu Pada PT. Perkebunan Nusantara IX

Oleh: Intan Widya Kusuma
Program Studi Matematika, FMIPA Universitas Negeri Yogyakarta
Email: lucifer_elf@rocketmail.com
Agus Maman Abadi
Jurusan Pendidikan Matematika, FMIPA Universitas Negeri Yogyakarta
Email: agusmaman@uny.ac.id

Abstrak

Kebutuhan akan ketersediaan gula yang berbahan dasar tebu cukup tinggi mengingat gula adalah salah satu kebutuhan pokok sehingga Industri gula perlu diperhatikan untuk meningkatkan kesejahteraan masyarakat. Oleh karena itu, perlu adanya cara memprediksi produksi tebu agar produsen tebu dapat memperkirakan jumlah produksi tebu tahun depan apakah sudah dapat memproduksi gula sesuai dengan kebutuhan konsumsi masyarakat. Tujuan penulisan ini adalah untuk memprediksi produksi tebu di PT. Perkebunan Nusantara IX menggunakan model *backpropagation neural network*.

Perkiraan produksi tebu ini menggunakan variabel input produksi tebu dan curah hujan masa lalu. Prosedur peramalan/ perkiraan diawali dengan pembagian data menjadi data pelatihan dan pengujian. Selanjutnya dilakukan pemilihan variabel input yang memberikan korelasi cukup signifikan terhadap variabel output. Kemudian dilakukan perancangan struktur jaringan yang optimum serta pemilihan learning rate dan momentum. Proses validasi dilakukan terhadap struktur jaringan yang optimum untuk mengetahui tingkat keakuratan perkiraan produksi tebu.

Model *backpropagation neural network* terpilih adalah model dengan 4 input, 1 lapisan tersembunyi (dengan 8 neuron), dan 1 output yang menggunakan fungsi aktivasi sigmoid biner pada pelatihan dan fungsi aktivasi linear pada output. Hasilnya adalah perkiraan produksi tebu menggunakan *backpropagation neural network* dengan tingkat keakuratan pada proses pelatihan mencapai MSE sebesar 14,2486 dan MAPE sebesar 0,0217%, sedangkan pada proses pengujian mencapai MSE sebesar 36.612 dan MAPE sebesar 2,6547%.

Kata kunci: produksi tebu, *backpropagation neural network*, pelatihan, pengujian

1. PENDAHULUAN

Pada tahun 2007-2009, industri gula berbasis tebu merupakan salah satu pendapatan bagi sekitar 900.000 petani tebu dan tenaga kerja yang terlibat mencapai 1.300.000 orang, dengan luas areal perkebunan tebu sekitar 400.000 ha. (Mulyadi, dalam jurnal P3GI 2009). Kondisi industri gula merupakan salah satu aspek yang perlu menjadi perhatian untuk meningkatkan kesejahteraan masyarakat. Oleh karena itu, perlu adanya cara memprediksi produksi tebu agar produsen dapat memperkirakan jumlah produksi tebu tahun depan apakah sudah dapat memproduksi gula sesuai dengan kebutuhan konsumsi masyarakat.

Salah satu metode perkiraan yang dapat diterapkan untuk perkiraan produksi tebu adalah *Backpropagation* atau *Feedforward Neural Network* (FFNN). Kelebihan metode ini mampu memformulasikan pengalaman dan pengetahuan peramal, serta sangat

fleksibel dalam perubahan aturan perkiraan. (Mataram, 2008). Selain itu, penelitian tentang perkiraan produksi tebu dengan menggunakan metode *Backpropagation Neural Network* belum pernah dilakukan. Pemodelan perkiraan produksi tebu yang pernah dilakukan adalah menggunakan model ARIMA (Mandal, 2006) dan *Artificial Neural Network* (Obe, 2010).

Backpropagation merupakan model *neural network* dengan banyak lapisan yang sering digunakan pada perkiraan *time series*. Algoritma pembelajaran *backpropagation* mengaktifkan *neuron-neuron* pada perambatan maju (*forward propagation*) menggunakan fungsi aktivasi yang dapat dideferensialkan untuk mendapatkan *error output*. Kemudian *error output* ini digunakan untuk mengubah nilai bobot-bobotnya kearah mundur (*backward*). Modifikasi atau perubahan bobot dilakukan untuk menurunkan kesalahan yang terjadi. (Kusumadewi, 2003). Pada tulisan ini akan dibahas bagaimana menentukan perkiraan produksi tebu pada PT. Perkebunan Nusantara IX dengan model *backpropagation neural network*.

2. METODE PENELITIAN

Pada penelitian tentang perkiraan produksi tebu ini, diperlukan data yang merupakan faktor-faktor yang mempengaruhi produksi tebu. Data yang digunakan adalah data curah hujan dan data produksi tebu masa lalu yang diambil dari PT. Perkebunan Nusantara IX (PG. Mojo) pada tahun 1979-2010.

Variabel yang digunakan pada perkiraan produksi tebu pada penelitian ini terdiri dari 4 variabel *input* dan 1 variabel target *output*, yaitu:

X_1 = variabel *input* curah hujan setahun yang lalu

X_2 = variabel *input* curah hujan dua tahun yang lalu

X_3 = variabel *input* produksi tebu setahun yang lalu

X_4 = variabel *input* produksi tebu dua tahun yang lalu

Y_1 = variabel target *output* produksi tebu yang diperkirakan

Data curah hujan dan produksi tebu masa lalu yang diperoleh sebanyak 30 pasang data. Dari data tersebut 20 pasang data digunakan sebagai data pelatihan dan 10 pasang data digunakan sebagai data pengujian. Analisis data menggunakan program MATLAB 7.0. Keakuratan model diukur menggunakan *Mean Square Error* (MSE) dan *Mean Absolute Percentage Error* (MAPE).

Menurut Kusumadewi (2010: 105-110), algoritma *backpropagation* adalah sebagai berikut:

- a. Inisiasi bobot dengan mengambil bobot awal menggunakan nilai random yang terkecil.
- b. Menetapkan:
 1. Maksimum Epoch
Maksimum epoch adalah jumlah maksimum iterasi yang ditetapkan.
 2. Target *Error*
Target *error* adalah batas toleransi *error* yang diijinkan.
 3. *Learning Rate* (α)
Learning rate adalah laju pembelajaran, semakin besar *learning rate* akan berimplikasi pada semakin besarnya langkah pembelajaran.
- c. Inisiasi: Epoch=0, MSE=1.
- d. Mengerjakan langkah-langkah berikut selama kondisi penghentian belum terpenuhi:
 1. Epoch = Epoch + 1
 2. Untuk tiap-tiap pasangan elemen yang akan dilakukan pembelajaran, kerjakan:

Feedforward:

 - i. Tiap-tiap unit *input* (x_i , $i=1,2,3,\dots,n$) menerima sinyal x_i dan meneruskan sinyal tersebut ke semua unit yang ada di atasnya, yaitu *hidden layer*.
 - ii. Tiap-tiap unit pada *hidden layer* (z_j , $j=1,2,3,\dots,p$) menjumlahkan sinyal-sinyal *input* terbobot sebagai berikut:

$$z_{mj} = v_{j0} + \sum_{i=1}^n x_i v_{ji}$$

v_{j0} = bias pada unit tersembunyi j

untuk menghitung sinyal *output*nya digunakan fungsi aktivasi sebagai berikut:

$$z_j = f(z_{mj})$$

Kemudian sinyal tersebut dikirim ke semua unit di lapisan atasnya yaitu unit-unit *output*. Langkah kedua ini dilakukan sejumlah banyaknya *hidden layer*.

- iii. Tiap-tiap unit *output* (y_k , $k=1,2,3,\dots,m$) menjumlahkan sinyal-signal *input* terbobot sebagai berikut:

$$y_{in_k} = w_{k0} + \sum_{j=1}^p x_j w_{kj}$$

w_{k0} = bias pada unit *output* k

untuk menghitung sinyal *output*nya digunakan fungsi aktivasi sebagai berikut:

$$y_k = f(y_{in_k})$$

Kemudian sinyal tersebut dikirim ke semua unit di lapisan atasnya yaitu unit-unit *output*.

Backpropagation

- i. Tiap-tiap unit *output* (y_k , $k=1,2,3,\dots,m$) menerima target pola yang berhubungan dengan pola *input* pembelajaran, hitung informasi *error*nya:

$$\delta_k = (t_k - y_k) f'(y_{in_k})$$

kemudian hitung koreksi bobot (yang nantinya akan digunakan untuk memperbaiki nilai w_{kj}) sebagai berikut:

$$\Delta w_{kj} = \alpha \delta_k x_j$$

selain itu, hitung juga koreksi bias (yang nantinya akan digunakan untuk memperbaiki nilai w_{k0}), dan kirimkan δ_k ke unit-unit pada lapisan di bawahnya.

- ii. Tiap-tiap unit tersembunyi (x_j , $j=1,2,3,\dots,p$) menjumlahkan hasil perubahan *input*nya dari unit-unit di lapisan atasnya sebagai berikut:

$$\Delta_{in_j} = \sum_{k=1}^m \delta_k w_{kj}$$

untuk menghitung informasi *error*, kalikan nilai Δ_{in_j} dengan turunan dari fungsi aktifasinya:

$$\delta_j = \Delta_{in_j} f'(x_{in_j})$$

kemudian hitung koreksi bobot (yang nantinya akan digunakan untuk memperbaiki nilai w_{uj}),

- iii. Tiap-tiap unit *output* (y_k , $k=1,2,3,\dots,m$) memperbaiki bias dan bobotnya sebagai berikut:

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

Tiap-tiap unit *hidden layer* (z_j , $k=1,2,3,\dots,p$) memperbaiki bias dan bobotnya sebagai berikut:

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

iv. Tes kondisi berhenti

Langkah-langkah yang harus dilakukan untuk pemodelan perkiraan adalah sebagai berikut:

a. Preprocessing/ Normalisasi

Pada proses perkiraan menggunakan *Backpropagation Neural Network*, sebelum dilakukan pelatihan, data *input* dan target *output* harus dinormalisasi terlebih dahulu. Normalisasi adalah penskalaan terhadap nilai-nilai masuk ke dalam suatu range tertentu. Hal ini dilakukan agar nilai *input* dan target *output* sesuai dengan range dari fungsi aktivasi yang digunakan dalam jaringan.

Data *input* dan target *output* dinormalisasi dengan cara membawa data ke bentuk normal yang memiliki mean = 0 dan standar deviasi =1, berdasarkan rumus:

$$\text{nilai baru} = \frac{|\text{nilai lama} (\text{rata} - \text{rata})|}{\text{standar deviasi}}$$

b. Pemilihan Variabel Input

Misalkan calon variabel *input* x_i , $i = 1, 2, \dots, n$ dan variabel output y , akan ditetapkan variabel-variabel *input* yang relevan, x_i , $i = 1, 2, \dots, n$, yang berhubungan dengan y_j , $j = 1, 2, \dots, m$ menggunakan metode *backpropagation*. Hal ini dilakukan dengan menggunakan eliminasi *backward* dan fungsi biaya *mean square error* (MSE) serta *mean absolute percentage error* (MAPE). Pemilihan variabel dilakukan dengan mengeliminasi variabel yang tidak berguna dan mempertahankan variabel-variabel yang memberikan nilai *korelasi* yang cukup signifikan terhadap variabel output y_i .

c. Perancangan Struktur Jaringan yang Optimum

Langkah selanjutnya adalah menentukan jumlah lapisan *input*, lapisan tersembunyi (*hidden layer*), dan lapisan *output*. Jumlah lapisan input berdasarkan pada banyaknya data yang mempengaruhi perkiraan. Sedangkan banyaknya lapisan output adalah banyaknya hasil output perkiraan yang dicari.

d. Pemilihan Koefisien Pemahaman (*Learning Rate*) dan *Momentum*

Koefisien pemahaman pada *neural network* adalah learning rate atau laju pembelajaran. Besarnya learning rate akan berimplikasi pada besarnya langkah pembelajaran. *Momentum* dalam *neural network* adalah perubahan bobot yang didasarkan pada arah gradient pola terakhir dan pola sebelumnya. Pada pembangunan jaringan *Backpropagation* yang akan digunakan dalam perkiraan, hasil keputusan yang kurang memuaskan dapat diperbaiki dengan menggunakan learning rate dan *momentum* secara trial and error untuk mendapatkan nilai bobot yang optimum agar MSE dan MAPE jaringan dapat diperbaiki.

e. Pemilihan Struktur Jaringan yang Optimum dan Penggunaannya untuk Peramalan/ Perkiraan

Langkah-langkah pemilihan jaringan yang optimum dijelaskan oleh Samsodin dkk (2010: 4), sebagai berikut:

- i. Proses pelatihan dilakukan terhadap data pelatihan dengan struktur jaringan yang memiliki bagian simpul tersembunyi berbeda akan diperoleh nilai *output* jaringan. Nilai MSE dan MAPE dihitung. Jaringan yang memiliki nilai MSE dan MAPE terendah dipilih sebagai jaringan yang optimum dan digunakan untuk perkiraan.
- ii. Setelah proses pelatihan dilakukan proses pengujian dengan struktur jaringan yang memiliki bilangan simpul tersembunyi berbeda yang telah dilatih akan diperoleh nilai *output* jaringan. Nilai MSE dan MAPE dari masing-masing struktur jaringan dihitung. Proses pengujian digunakan untuk menguji prestasi pelatihan dan sebagai pendukung bahwa jaringan terpilih sebagai jaringan yang tepat untuk model peramalan.
- iii. Proses validasi dilakukan dengan menggunakan jaringan terpilih terhadap data validasi untuk melihat prestasi ramalannya.

f. Postprocessing/Denormalisasi

Setelah proses pelatihan dan pengujian selesai, untuk mengembalikan nilai ternormalisasi *output* jaringan ke nilai yang sebenarnya, dilakukan proses denormalisasi atau postprocessing.

3. HASIL PENELITIAN DAN PEMBAHASAN

Sebelum dilakukan perkiraan, data harus dinormalisasikan terlebih dahulu. Tahap pertama pada perkiraan produksi tebu adalah pemilihan variabel input. Pemilihan variabel input dilakukan dengan proses pembelajaran terhadap data pelatihan, data pengujian, serta data pelatihan dan pengujian untuk mendapatkan nilai MSE dan MAPE yang paling kecil. Pembelajaran dilakukan dengan menggunakan: (X_1, X_2, X_3, X_4) ; (X_2, X_3, X_4) ; (X_1, X_3, X_4) ; (X_1, X_2, X_4) ; (X_1, X_2, X_3) ; (X_1, X_2) ; (X_1, X_3) ; (X_1, X_4) ; (X_2, X_3) ; (X_2, X_4) ; (X_3, X_4) .

Proses pembelajaran dilakukan dengan menggunakan *backpropagation gradient descent* dengan *adaptive learning rate* (*traingda*) dengan *n input* ($n=2,3,4$), 1 lapisan tersembunyi (dengan 5 *neuron*), dan 1 output. Fungsi aktivasi sigmoid bipolar pada pelatihan dan fungsi aktivasi linear pada *output*. Parameter-parameter: maksimum epoh = 5000; laju pembelajaran = 0,1; toleransi *error* = 10^{-3} ; maksimum kenaikan kinerja = 1,06; rasio kenaikan *learning rate* = 1,2; dan rasio penurunan *learning rate* = 0,6.

MSE dan MAPE terkecil terhadap data pelatihan serta data pelatihan dan pengujian terjadi ketika semua variabel digunakan. Sedangkan MSE dan MAPE terkecil terhadap data pengujian terjadi ketika variabel curah hujan setahun yang lalu dieliminasi. Oleh karena MSE dan MAPE terkecil tidak terjadi pada variabel input yang sama, maka dilakukan proses pelatihan terhadap data pelatihan dan proses pengujian terhadap data pengujian dengan menggunakan semua variabel input dan mengeliminasi variabel curah hujan setahun yang lalu. Hasilnya ditampilkan oleh grafik di bawah ini.

Tabel 1. SSE, MSE, dan MAPE pada proses pelatihan dan pengujian untuk mencari variabel *input*.

Variabel Input		Semua variabel digunakan	Variabel curah hujan setahun yang lalu (c-1) dieliminasi
Proses Pelatihan	MSE	14,1214	1,5469e+003
	MAPE	0,0165	0,1155
Proses Pengujian	MSE	8,3941e+003	1,3844e+004
	MAPE	1,1532	1,5925

Table 1. menunjukkan bahwa MSE dan MAPE terkecil pada proses pelatihan maupun proses pengujian terjadi pada saat semua variabel input digunakan, sehingga variabel input yang akan berpartisipasi dalam perkiraan produksi tebu adalah semua variabel.

Untuk mendapatkan model *backpropagation neural network* yang baik, dapat dilakukan percobaan terhadap beberapa macam arsitektur jaringan agar menghasilkan nilai MSE dan MAPE yang terkecil dengan proses pembelajaran terhadap data pelatihan, data pengujian, serta data pelatihan. Proses pembelajaran dilakukan dengan menggunakan *backpropagation gradient descent* dengan *adaptive learning rate* (traingda) dengan 4 variabel *input*, 1 lapisan tersembunyi, dan 1 *output*. Parameter-parameter: maksimum epoch = 10000; laju pembelajaran = 0,1; toleransi *error* = 10^{-3} ; maksimum kenaikan kinerja = 1,06; rasio kenaikan *learning rate* = 1,2; dan rasio penurunan *learning rate* = 0,6. Jumlah *neuron* pada lapisan tersembunyi akan dicari dengan melakukan percobaan terhadap jumlah *neuron* pada lapisan tersembunyi ($N=2,3,4,\dots,10$). Fungsi aktivasi pada *hidden layer* juga akan dilakukan percobaan (sigmoid biner atau sigmoid bipolar), sedangkan fungsi aktivasi pada *output* adalah fungsi linear (purelin).

Proses pembelajaran terhadap data pelatihan dan pengujian menghasilkan MSE dan MAPE paling kecil pada saat jaringan menggunakan fungsi aktivasi sigmoid biner dan 9 *neuron* pada *hidden layer*. Proses pembelajaran terhadap data pelatihan menghasilkan MSE dan MAPE paling kecil pada saat jaringan menggunakan fungsi aktivasi sigmoid biner dan 3 *neuron* pada *hidden layer*. Proses pembelajaran terhadap data pelatihan dan pengujian menghasilkan MSE paling kecil pada saat jaringan menggunakan fungsi aktivasi sigmoid biner dan 8 *neuron* pada *hidden layer*, sedangkan MAPE terkecil yang terjadi adalah pada saat jaringan menggunakan fungsi aktivasi sigmoid bipolar dan 4 *neuron* pada *hidden layer*. Hal ini disebabkan arsitektur jaringan *backpropagation* yang optimum untuk masing-masing data pelatihan dan pengujian, pelatihan, serta pengujian berbeda. Oleh karena nilai MSE dan MAPE terkecil pada proses pembelajaran terhadap data pelatihan dan pengujian, data pelatihan, serta data pengujian tidak terjadi pada arsitektur jaringan yang sama, maka arsitektur jaringan optimum yang akan digunakan dalam perkiraan produksi tebu dipilih berdasarkan proses pelatihan dan pengujian *backpropagation* dengan MSE dan MAPE terkecil.

Tabel 2. MSE, dan MAPE pada proses pelatihan dan pengujian untuk menentukan arsitektur jaringan yang optimum.

Fungsi Aktifasi dan Jumlah <i>Neuron</i>	Proses Pelatihan		Proses Pengujian	
	MSE	MAPE	MSE	MAPE
Sigmoid biner dan 7 <i>neuron</i>	9,4365e+003	0,5286	1,3667e+004	1,6114
Sigmoid biner dan 3 <i>neuron</i>	7,4824e+003	0,4479	1,2372e+004	1,6107
Sigmoid biner dan 8 <i>neuron</i>	9,78552+003	0,5499	1,1699e+004	1,4518
Sigmoid bipolar dan 4 <i>neuron</i>	9,6370e+003	0,5608	1,5412e+004	1,7239

Table 2. menunjukkan bahwa MSE dan MAPE terkecil pada proses pelatihan terjadi pada saat jaringan menggunakan fungsi aktifasi sigmoid biner dan 3 *neuron* pada *hidden layer*, sedangkan MSE dan MAPE terkecil pada proses pengujian terjadi pada saat jaringan menggunakan fungsi aktifasi sigmoid biner dan 8 *neuron* pada *hidden layer*. Keakuratan perkiraan produksi tebu ditentukan oleh proses pengujian, oleh karena itu fungsi aktifasi yang akan digunakan untuk perkiraan adalah sigmoid biner dan 8 *neuron* pada *hidden layer*. Sehingga arsitektur jaringan yang optimum yang dihasilkan yaitu jaringan syaraf *backpropagation* menggunakan fungsi aktifasi sigmoid bipolar dengan 4 variabel *input*, 8 *neuron* pada *hidden layer*, dan 1 *output*

Setelah mendapatkan arsitektur jaringan yang optimum, akan dicari tingkat keakuratan model *backpropagation neural network* untuk mengetahui seberapa cocok model *backpropagation neural network* ini pada perkiraan produksi tebu.

a. Proses Pelatihan

Proses pembelajaran dilakukan dengan *Backpropagation Neural Network* menggunakan algoritma *gradient descent* dengan *adaptive learning rate* (traingda), dengan 4 *input*, 1 lapisan tersembunyi (dengan 8 *neuron*), dan 1 *output*. Fungsi aktifasi sigmoid biner pada pelatihan dan fungsi aktifasi linear pada *output*. Parameter-parameter: maksimum epoh = 50000; laju pembelajaran = 0,1; toleransi *error* = 10^{-6} ; maksimum kenaikan kinerja = 1,06; rasio kenaikan *learning rate* = 1,2; dan rasio penurunan *learning rate* = 0,6. Hasil dari proses pelatihan yang berupa *output* jaringan dan selisih antara target *output* dengan *output* jaringan (*error*) dapat dilihat pada tabel di bawah ini.

Tabel 3. Nilai-nilai Target *Output*, *Output* Jaringan, dan *Error* Pelatihan

Data ke	Target <i>Output</i>	<i>Output</i> Jaringan	<i>Error</i>	Data ke	Target <i>Output</i>	<i>Output</i> Jaringan	<i>Error</i>
---------	----------------------	------------------------	--------------	---------	----------------------	------------------------	--------------

1	697,0000	696,7021	0,2979	11	609,0000	609,0377	-0,0377
2	734,0000	734,0184	-0,0184	12	753,0000	752,9743	0,0257
3	715,0000	715,3892	-0,3892	13	615,0000	615,0290	-0,0290
4	729,0000	728,8895	0,1105	14	794,0000	793,9881	0,0119
5	473,0000	473,0835	-0,0835	15	742,0000	742,0729	-0,0729
6	918,0000	917,9663	0,0337	16	616,0000	616,0143	-0,0143
7	883,0000	883,0033	-0,0033	17	531,0000	531,0759	-0,0759
8	849,0000	848,9931	0,0069	18	799,0000	799,0594	-0,0594
9	723,0000	723,0322	-0,0322	19	539,0000	539,0250	-0,0250
10	614,0000	614,0029	-0,0029	20	616,0000	615,9659	0,0341

Tabel 3. menunjukkan nilai *error* yang dihasilkan sangat kecil sehingga proses pelatihan sudah baik. Untuk melihat keakuratan hasil pengujian jaringan *backpropagation*, dapat dilihat melalui nilai MSE dan MAPE berturut-turut adalah 14,2486 dan 0,0217.

b. Proses Pengujian

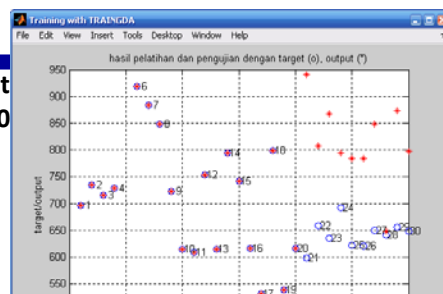
Selanjutnya dilakukan proses pengujian terhadap jaringan *backpropagation* untuk mengetahui tingkat keakuratan model. Hasil dari proses pengujian yang berupa *output* jaringan dan selisih antara target *output* dengan *output* jaringan (*error*) dapat dilihat pada tabel di bawah.

Tabel 4. Nilai-nilai Target *Output*, *Output* Jaringan, dan *Error* Pengujian

Data ke	Target <i>Output</i>	<i>Output</i> Jaringan	<i>Error</i>
1	598,0000	940,1905	-342,1905
2	659,0000	807,6067	-148,6067
3	635,0000	867,3063	-232,3063
4	692,0000	793,8293	-101,8293
5	622,0000	784,7211	-162,7211
6	621,0000	784,4022	-163,4022
7	650,0000	848,4313	-198,4313
8	641,0000	647,5006	-6,5006
9	655,0000	873,9701	-218,9701
10	648,0000	796,5412	-148,5412

Tabel 4 menunjukkan nilai-nilai dari target *output*, *output* jaringan, dan *error* hasil pengujian. Untuk melihat keakuratan hasil pengujian jaringan *backpropagation*, dapat dilihat melalui nilai MSE dan MAPE berturut-turut adalah 36.612 dan 2,6547.

Gambar 1. memperlihatkan hasil pelatihan dan pengujian jaringan *backpropagation*.



Gambar 1. Perbandingan nilai target *output* dengan *output* jaringan pada data pelatihan dan pengujian

Tingkat keakuratan model *backpropagation neural network* telah diketahui, selanjutnya proses perkiraan produksi tebu dapat dilakukan menggunakan arsitektur jaringan terpilih. Data *input* untuk perkiraan produksi tebu pada tahun 2011 adalah 2897 (X_1), 1645 (X_2), 648 (X_3), dan 655 (X_4), kemudian data-data ini dinormalisasi menjadi 1,3694 (x_1), -1,8105 (x_2), -0,5967 (x_3), -0,4446 (x_4). Nilai *output* ternormalisasi yang dihasilkan jaringan adalah **-2,7598**. Kemudian, nilai tersebut didenormalisasi sehingga nilai *output* jaringan yang diperoleh adalah **367,1700**. Sehingga, perkiraan jumlah produksi tebu tahun depan sebanyak **367,1700** ton.

4. SIMPULAN DAN SARAN

Aplikasi model *neural network* pada perkiraan produksi tebu di PT. Perkebunan Nusantara IX menggunakan model *backpropagation neural network* dengan 4 *input* (yaitu: curah hujan setahun yang lalu, curah hujan dua tahun yang lalu, produksi tebu setahun yang lalu, dan produksi tebu dua tahun yang lalu), 1 *hidden layer* (dengan 3 *neuron*), dan 1 *output*. Proses pembelajaran menggunakan algoritma *gradient descent* dengan *adaptive learning rate* (traingda), Fungsi aktifasi sigmoid biner pada pelatihan dan fungsi aktifasi linear pada *output*. Parameter-parameter: maksimum epoch = 50000; laju pembelajaran = 0,1; toleransi *error* = 10^{-3} ; maksimum kenaikan kinerja = 1,06; rasio kenaikan *learning rate* = 1,2; dan rasio penurunan *learning rate* = 0,6. Nilai MAPE dan MSE yang diperoleh pada proses pelatihan adalah 14,2486 dan 0,0217%, sedangkan nilai MSE dan MAPE pada proses pengujian adalah 36.612 dan 2,6547%. Hasil perkiraan untuk satu tahun kedepan menunjukkan bahwa jumlah produksi tebu akan menurun drastis dibanding tahun-tahun sebelumnya.

Sedangkan bagi pembaca yang ingin menggunakan *neural network* secara umum untuk perkiraan dapat digunakan *radial basic function*. *Radial basic function*

Seminar Nasional Matematika dan Pendidikan Matematika
Yogyakarta, 3 Desember 2011

(RBF) merupakan bentuk *multilayer neural network* yang *unsupervised*. Arsitektur RBF menggunakan fungsi basis sebagai fungsi aktivasi pada *hidden layer* dan fungsi linier pada *output*.

DAFTAR PUSTAKA

- Kusumadewi, Sri. 2004. *Membangun Jaringan Syaraf Tiruan Menggunakan Matlab dan Excel Link*. Yogyakarta: Graha Ilmu.
- Kusumadewi, Sri & Hartati, Sri. 2010. *Neuro-Fuzzy Integrasi Sistem Fuzzy & Jaringan Syaraf Edisi 2*. Yogyakarta: Graha Ilmu.
- Mandal, B.N. 2006. “Forecasting Sugarcane Production in India with Arima Model”. PhD scholar, IASRI, New Delhi-12. Hlm. 1-13.
- Mataram, I Made. 2008. “Peramalan Baban Hari Libur Menggunakan *Artificial Neural Network*”. *Jurnal Teknik Elektro* (Vol.7 No.2 Juli-Desember 2008). Hlm. 53-56.
- Mulyadi, M., Toharisman, A., & PDN., Mirzawan. 2009. “Identifikasi Potensi Lahan untuk Mendukung Pengembangan Agribisnis Tebu di Wilayah Timur Indonesia”. P3GI. Hlm. 1-15 .
- Obe, O.O. & Shangodoyin, D.K. 2010. “*Artificial Neural Network* Based Model for Forecasting Sugar Cane Production”. *Journal of Computer Science* 6(4). Hlm. 439-445.
- Samsodin dkk. 2010. “Aplikasi Jaringan Syaraf Tiruan dalam Peramalan dan Klasifikasi”. <http://files.myopera.com/padangyulian/blog/kel-5.pdf>. Diakses tanggal 23 Agustus 2011.