

# Pemanfaatan Backpropagation untuk Memprediksi Produksi Buah Kelapa Sawit pada PT. Tunas Baru Lampung Tbk

THE USE OF BACKPROPAGATION METHOD FOR PREDICTION OF PALM OIL FRUIT AT PT. TUNAS BARU LAMPUNG TBK

Fadhila Tangguh Admojo<sup>\*1</sup>, Andra Fahmi<sup>2</sup>, Eki Ariawan<sup>3</sup>, Hendy Apriza<sup>4</sup>  
<sup>1,2,3,4</sup> STMIK PalComTech: Jl. Basuki Rahmat No. 05, Palembang 30129, Indonesia  
<sup>1,2,3,4</sup> Jurusan Teknik Informatika STMIK PalComTech Palembang  
e-mail: <sup>\*1</sup>fadhila.tangguh@palcomtech.ac.id

## Abstrak

PT. Tunas Baru Lampung Tbk merupakan salah satu perusahaan di Indonesia yang bergerak dibidang industri tanaman kelapa sawit, dalam industri kelapa sawit hasil produksi merupakan faktor yang paling penting guna menghadapi persaingan global insudtri kelapa sawit. Untuk dapat menghadapi permasalahan dalam bidang produksi diperlukan suatu solusi sehingga perusahaan dapat mengembangkan strategi yang tepat. Prediksi hasil produksi kelapa sawit diperlukan untuk dijadikan acuan dalam melakukan pengawasan dan mengambil kebijakan agar hasil produksi tetap stabil atau bahkan dapat ditingkatkan sesuai target perusahaan. Metode backpropagation menggunakan data latih untuk memprediksi hasil produksi kelapa sawit pada PT. Tunas Baru Lampung Tbk yang dimplementasi menggunakan bahasa pemrograman Java. Berdasarkan hasil pengujian terhadap 6 data latih dan 3 data uji menggunakan 9 hidden layer dengan nilai learning rate sebesar 0.3, batas error sebesar 0.01 dan iterasi maksimal sebesar 60000 maka tingkat akurasi hasil prediksi rata – rata sebesar 79%.

**Kata kunci** — Backpropagation, jaringan saraf tiruan, kelapa sawit, prediksi,

## Abstract

PT. Tunas Baru Lampung Tbk is one of the companies in Indonesia which is engaged in the oil palm plantation industry. In this industry, production become the most important thing in facing global competition. The company needs to find a suitable solution to develop the right strategy in facing the competition in the palm oil industry. In conducting supervision and policymaking, prediction of palm oil fruit production is necessary to increase the production of palm oil fruit. This research used the backpropagation method to predict the production of oil palm fruit at PT. Tunas Baru Lampung Tbk. The yield of palm oil production was predicted using training data and implemented in the Java programming language. Based on the test results on six training data and three test data using nine hidden layers with a learning rate value of 0.3, an error limit of 0.01, and a maximum iteration of 60000, the average prediction accuracy rate is 79%.

**Keywords** — Artificial neural network, backpropagation, palm oil, prediction

## 1. PENDAHULUAN

**K**elapa sawit (*Elaeis Guineensis Jacq*) merupakan tanaman perkebunan yang berasal dari Afrika dan merupakan bahan utama pembuatan minyak nabati dan bahan agro industri.

PT. Tunas Baru Lampung Tbk. merupakan salah satu perusahaan di Indonesia yang bergerak di bidang industri tanaman kelapa sawit sebagai bahan pembuatan minyak goreng [1]. Salah satu masalah yang muncul yaitu pada bidang produksi karena hasil produksi merupakan faktor yang paling penting dalam menentukan kebijakan dan strategi perusahaan untuk menghadapi persaingan industri kelapa sawit. Prediksi diperlukan untuk mengetahui dan melihat perkembangan hasil produksi di masa yang akan datang sehingga tercapainya target produksi yang telah ditetapkan oleh perusahaan.

Beberapa metode yang dapat digunakan untuk menghitung prediksi produksi kelapa sawit [2][3][4][5][6]. Penelitian ini menggunakan suatu model Jaringan Syaraf Tiruan yaitu *backpropagation* untuk memprediksi jumlah hasil panen/produksi kelapa sawit. *Backpropagation*

merupakan salah satu metode prediksi berbasis *machine learning* yang dapat digunakan untuk memprediksi, meskipun tingkat keberhasilan *neural network* sangat bergantung pada ketersediaan data dan validnya data untuk digunakan pada proses belajar (*training*) namun kelebihan metode ini fleksibel dalam perubahan aturan perkiraan sehingga mampu memformulasikan pengalaman dan pengetahuan untuk memprediksi dengan nilai error yang kecil [7].

2. METODE PENELITIAN

2.1. Pengumpulan Data

Prediksi dilakukan terhadap 1 blok area dengan luas lahan 22.5 Ha dengan jumlah pohon 141 pohon/Ha, maka total pohon dalam 1 blok sebanyak 3179 dengan rata rata umur 14 tahun. Prediksi dilakukan untuk jumlah produksi per periode (4 bulan). Data latih dan data uji yang digunakan berdasarkan laporan blok 1A hasil panen pada PT. Tunas Baru Lampung berisi data-data terkait dengan parameter yang mempengaruhi panen/produksi kelapa sawit meliputi data periode, data penggunaan pupuk, data curah hujan, dan data jumlah panen/produksi. Data yang digunakan dalam rentang tahun 2017 sampai 2019. Data disajikan pada Tabel 1.

Tabel 1. Data Latih blok 1A (2017-2018) dan Data Uji blok 1A (2019)

Data Latih 2017 - 2018 blok 1A				Data Uji 2019 blok 1A			
Periode/Bulan	Produksi (Kg)	Pemakaian Pupuk (Kg)	Curah Hujan (Mm)	Periode/Bulan	Produksi (Kg)	Pemakaian Pupuk (Kg)	Curah Hujan (Mm)
1/Jan-Apr 2017	10050	9537	1098.5	1/Jan-Apr 2019	11519	9537	1195.6
2/Mei-Agu 2017	13009	7947.5	512.8	2/Mei-Agu 2019	12338	7947.5	336.3
3/Sept-Des 2017	11852	7947.5	797.9	3/Sept-Des 2019	15395	7947.5	714.8
1/Jan-Apr 2018	9189	9537	976.1				
2/Mei-Agu 2018	17195	7947.5	411.3				
3/Sept-Des 2018	19062	7947.5	926.3				

2.2 Pengolahan Data

Pada penelitian ini data diolah melalui beberapa hapan menggunakan metode *Backpropagation* [8] dengan fungsi aktivasi sigmoid biner. Tahap pertama yaitu normalisasi data menggunakan Persamaan (1) dengan hasil normalisasi data disajikan pada Tabel 2.

$$x_i \text{ normalisasi} = (x_i - \min(x)) / (\max(x) - \min(x)) \dots\dots\dots(1)$$

Dimana :

$x_i$  normalisasi : Hasil normalisasi terhadap data  $x$  ke- $i$

$x_i$  : Data  $x$  ke- $i$

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

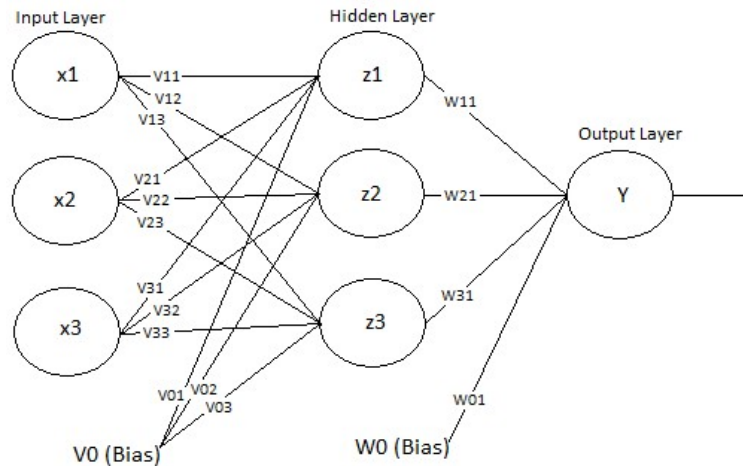
$\max(x)$  : Data maksimum yang terdapat pada data  $x$

Tabel 2. Hasil Normalisasi data

Data ke-	Periode/ Bulan	Pemakaian Pupuk/ Kg	Curah Hujan/ Mm	Produksi/ Kg
1	0	1	0.887	0.087
2	0.5	0	0.205	0.386
3	1	0	0.537	0.269
4	0	1	0.744	0
5	0.5	0	0.087	0.810
6	1	0	0.686	1
7	0	1	1	0.236
8	0.5	0	0	0.318

9	1	0	0.440	0.628
---	---	---	-------	-------

Tahap kedua yaitu inisialisasi awal parameter-parameter *backpropagation* yaitu dengan jumlah input layer 3, jumlah *hidden layer* 3, jumlah *output layer* 1, *learning rate* 0.9, jumlah iterasi 15000, toleransi error 0.01. Berdasarkan inisialisasi arsitektur *Backpropagation* digambarkan pada Gambar 1.



Gambar 1. Arsitektur *backpropagation*

Tahap ketiga yaitu inisialisasi awal nilai V dan nilai W yang dilakukan dengan memberikan rentang nilai 0 sampai 1 secara acak. Hasil inisialisasi nilai V dan W dapat dilihat pada Gambar 2.

bobot awal V			
J \ I	1	2	3
0	0.024	0.679	0.972
1	0.921	0.642	0.275
2	0.975	0.379	0.207
3	0.364	0.295	0.461

(a)

bobot awal W	
J \ K	1
0	0.638
1	0.097
2	0.931
3	0.410

(b)

Gambar 2. Bobot Nilai Awal (a) Nilai Awal V, (b) Nilai Awal W

Tahap keempat adalah fase *feed forward* yaitu untuk menghitung sinyal yang dikirim dari *input layer* ke *hidden layer* ( $Z$ ) hingga menghasilkan sinyal keluaran dari *hidden layer*. Perhitungan nilai  $Z_j$  ( $Z_1, Z_2, Z_3$ ) dilakukan dengan menggunakan persamaan (2) dan (3). Sehingga didapatkan nilai  $Z_1 = 0.789$ ,  $Z_2 = 0.789$ ,  $Z_3 = 0.830$ .

$$z_{inj} = v_{0j} + \sum_{i=1}^n x_i v_{ij} \dots\dots\dots(2)$$

$$z_j = f(z_{inj}) = \frac{1}{1 + e^{-z_{inj}}} \dots\dots\dots(3)$$

Dimana :

- $V_{0j}$  : Bias pada *layer hidden layer* ke- $j$
- $x_i$  : *Layer input* ke- $i$
- $V_{ij}$  : Bobot yang menghubungkan *layer input* ke- $i$  dan *layer hidden* ke- $j$
- $Z_{inj}$  : Sinyal dari *input layer* ke *layer hidden* ke- $j$

$Z_j$  : Layer hidden layer ke-j  
 $f(Z_{inj})$  : Fungsi aktivasi terhadap nilai  $Z_{inj}$

Langkah kelima adalah menghitung nilai *output layer* ( $Y$ ), perhitungan nilai  $Y_i(Y_1, Y_2, \dots, Y_n)$  dilakukan dengan persamaan (4) dan (5). Sehingga didapatkan nilai  $Y = 0.857$

$$Y_{ink} = W_{0k} + \sum_{n/j=0} x_j W_{jk} \dots\dots\dots(4)$$

$$Y_k = fY_{ink} = \frac{1}{1 + e^{Y_{ink}}} \dots\dots\dots(5)$$

Dimana :

$W_{0k}$  : Bias pada *layer output layer* ke -k  
 $Z_j$  : Layer hidden layer ke -j  
 $W_{jk}$  : Bobot yang menghubungkan *hidden layer* ke-j dan *layer output layer* ke-k  
 $Y_{ink}$  : Sinyal dari *hidden layer* ke *layer output layer* ke-j  
 $Y_k$  : Layer output layer ke-k  
 $f_{Y\_ink}$  : Fungsi aktivasi terhadap nilai  $Y\_ink$

Langkah keenam adalah fase backpropagasi yaitu menghitung dari *output layer* ke *hidden layer* bertujuan untuk memperbaharui bobot yang menghubungkan *output layer* dan *hidden layer* ( $W$ ). Perhitungan menggunakan persamaan (6), (7) dan (8).

$$\square_k = (t_k - Y_k) * \square'(Y_{ink}) \dots\dots\dots(6)$$

$$\Delta W_{jk} = \square_k * \square_k * Z_j \dots\dots\dots(7)$$

$$\Delta W_{0k} = \square_k * \square_k \dots\dots\dots(8)$$

Dimana :

$\square_k$  : Faktor koreksi dari *layer output* ke-k  
 $t_k$  : Target output pada *layer input* ke-k  
 $Y_k$  : Layer output ke-k  
 $\square$  : Laju pembelajaran  
 $\Delta W_{jk}$  : Koreksi bobot yang menghubungkan *layer output* ke-k dan *layer hidden* ke-j  
 $\Delta W_{0k}$  : Koreksi bias pada *layer input* ke-k

Berdasarkan perhitungan maka didapatkan hasil berupa :

$$\begin{aligned} \square_k &= (0.087 - 0.857) * 0.857 * (1 - 0.857) = -0.094 \\ \Delta W_0 &= 0.9 * (-0.094) = -0.066 \\ \Delta W_1 &= 0.9 * (-0.094) * 0.789 = -0.052 \\ \Delta W_2 &= 0.9 * (-0.094) * 0.789 = -0.052 \\ \Delta W_3 &= 0.9 * (-0.094) * 0.830 = -0.054 \end{aligned}$$

Tahap ketujuh yaitu menghitung perubahan bobot  $W$  menggunakan persamaan (9). Hasil perhitungan dapat dilihat pada tabel 3.

$$W_{jk} \text{ (baru)} = W_{jk} \text{ (lama)} + \Delta W_{jk} \dots\dots\dots(9)$$

**Tabel 3.** Tabel Bobot W Baru

	W
0	0.572
1	0.045
2	0.879
3	0.355

Langkah kedelapan yaitu perhitungan dari *hidden layer* ke *input layer*, bertujuan untuk memperbaharui bobot yang menghubungkan *hidden layer* dan *input layer* (bobot V). perhitungan dilakukan menggunakan persamaan (10), (11), (12) dan (13)

$$\delta_{in_j} = \sum_{nk=1} \delta_k W_{jk} \dots\dots\dots(10)$$

$$\delta_j = \delta_{in_j} f'(Z_{in_j}) = \delta_{in_j} Z_j (1 - Z_j) \dots\dots\dots(11)$$

$$\Delta V_{0j} = a \square_j \dots\dots\dots(12)$$

$$\Delta V_{ij} = a \square_j x_i \dots\dots\dots(13)$$

Dimana :

- $\square_{in_j}$  : Sinyal faktor koreksi dari *layer output* ke *hidden layer* ke-j
- $\square_j$  : Faktor koreksi dari *layer hidden layer* ke-j
- $\Delta V_{ij}$  : Koreksi bobot yang menghubungkan *layer hidden* ke-j dan *layer input* ke-i
- $\Delta V_{0j}$  : koreksi bias pada *layer hidden layer* ke-j

Berdasarkan perhitungan maka didapatkan hasil berupa :

- $\square_{in1} = -0.094 \times 0.097 = -0.009$
- $\square_{in2} = -0.094 \times 0.931 = -0.087$
- $\square_{in3} = -0.094 \times 0.410 = -0.038$
- $\square_1 = -0.009 \times 0.789 \times (1 - 0.789) = -0.001$
- $\square_2 = -0.097 \times 0.789 \times (1 - 0.789) = -0.014$
- $\square_3 = -0.038 \times 0.830 \times (1 - 0.830) = -0.005$
- $\Delta V_{01} = 0.9 \times (-0.001) = -0.001$
- $\Delta V_{02} = 0.9 \times (-0.014) = -0.010$
- $\Delta V_{03} = 0.9 \times (-0.005) = -0.003$
- $\Delta V_{11} = 0.9 \times (-0.001) \times 0 = 0$
- $\Delta V_{12} = 0.9 \times (-0.014) \times 0 = 0$
- $\Delta V_{13} = 0.9 \times (-0.005) \times 0 = 0$
- $\Delta V_{21} = 0.9 \times (-0.001) \times 1 = -0.001$
- $\Delta V_{22} = 0.9 \times (-0.014) \times 1 = -0.010$
- $\Delta V_{23} = 0.9 \times (-0.005) \times 1 = -0.003$
- $\Delta V_{31} = 0.9 \times (-0.001) \times 0.887 = -0.001$
- $\Delta V_{32} = 0.9 \times (-0.014) \times 0.887 = -0.009$
- $\Delta V_{33} = 0.9 \times (-0.005) \times 0.887 = -0.003$

Langkah kesembilan adalah menghitung perubahan bobot V menggunakan persamaan (14), hasil perhitungan disajikan pada tabel 4

$$V_{ij} \text{ (baru)} = V_{ij} \text{ (lama)} + \Delta V_{ij} \dots\dots\dots(14)$$

**Tabel 4.** Tabel Bobot V Baru

i\j	1	2	3
0	0.023	0.669	0.968
1	0.921	0.642	0.275
2	0.973	0.369	0.203
3	0.363	0.286	0.457

Setelah pembaharuan terhadap bobot  $W$  dan  $V$  selesai dilakukan, maka proses pelatihan untuk iterasi ke-1 telah selesai dilakukan, selanjutnya melakukan proses perhitungan *Mean Square Error* (MSE) untuk hasil iterasi ke-1. Hasil perhitungan MSE disajikan pada table 5.

**Tabel 5.** Tabel Perhitungan MSE untuk iterasi ke-1

Data Ke -	Target Output	JST Output	$(X_i - F_i)$	$(X_i - F_i)^2$	$\sum_{i=1}^n (X_i - F_i)^2$	$\sum_{i=1}^n (X_i - F_i)^2 / n$
	$X_i$	$F_i$				
1	0.087	0.857	-0.769	0.592	1.765	0.294
2	0.386	0.821	-0.434	0.188		
3	0.269	0.815	-0.546	0.298		
4	0.000	0.784	-0.784	0.615		
5	0.810	0.730	0.080	0.006		
6	1.000	0.747	0.252	0.063		

Nilai MSE hasil iterasi ke-1 sebesar 0.294 lebih besar dari toleransi error yang telah ditetapkan yaitu sebesar 0.01 sehingga proses pelatihan dilanjutkan ke iterasi ke-2 dan jika hasil iterasi ke-2 masih menghasilkan MSE yang lebih besar dari toleransi error maka iterasi akan terus dilakukan hingga batas iterasi yang telah ditentukan atau nilai error sudah lebih kecil dari toleransi error. Bobot akhir nilai  $V$  dan  $W$  setelah pelatihan dapat dilihat pada tabel 6.

**Tabel 6.** Bobot Terlatih  $V$  dan  $W$

J \ I		Bobot $V$			J \ I		Bobot $W$
		1	2	3			1
0		-2.998	-5.566	1.933	0	-2.464	
1		-2.607	-1.161	-2.293	1	-5.322	
2		4.752	-2.915	0.163	2	7.075	
3		1.454	10.485	-5.461	3	5.819	

### 2.3. Pengujian Data

Pengujian dilakukan menggunakan data uji pada tabel 1 terhadap nilai  $V$  dan nilai  $W$  baru hasil pelatihan pada tabel 6. Proses pengujian hanya dilakukan menggunakan perhitungan pada fase propagasi maju saja yaitu pada tahap ke-empat dan ke-lima. Berdasarkan hasil pengujian untuk periode 1/Jan-Apr2019 didapatkan hasil 0.240 sedangkan jumlah produksi periode 1/Jan-Apr2019 sebesar 0.236, sehingga dapat dikatakan error antara hasil prediksi dan data sebenarnya adalah 0.004. Sedangkan untuk mengetahui berapa kg jumlah produksi hasil prediksi maka *output* hasil prediksi harus dinormalisasikan dengan menggunakan persamaan (15). Berdasarkan hasil perhitungan didapatkan  $Y(\text{denormalisasi}) = (0.240 * 19.062 - 0.240 * 9.189) + 9.189 = 11.560,315$

$$Y(\text{denormalisasi}) = (Y * \max(\text{jumlah panen}) - Y * \min(\text{jumlah panen})) + \min(\text{jumlah panen}) \quad \dots(15)$$

Berdasarkan hasil denormalisasi diketahui bahwa prediksi jumlah produksi kelapa sawit untuk periode 1/Jan-Apr2019 dengan parameter *learning rate* 0.9, toleransi error 0.01, max iterasi 15000, dan arsitektur 3 *input layer*, 3 *hidden layer*, dan 1 *output layer* adalah 11.560 kg, sedangkan data produksi kelapa sawit periode 1 Jan-Apr2019 yaitu 11.519 kg. Proses prediksi untuk data periode 2/Mei-Agu 2019 dan 3/Sept-Des 2019 dilakukan dengan cara yang sama.

### 2.4 Implementasi

Implementasi dilakukan menggunakan bahasa pemrograman java mengikuti prosedur pemrosesan data yang dijabarkan dalam *flowchart* pada.



Gambar 3. Flowchart

Aplikasi Prediksi Panen Buah Kelapa Sawit

HIDDEN LAYER: 9  
 LEARNING RATE: 0.3  
 ITERASI MAX: 60000  
 TOLERANSI ERROR: 0.01

Table Data Pelatihan

Data ke	Periode (X1)	Pemakaian Pup...	Curah Hujan (M...	Produksi (KG) (...)
1	0	9,537	1,098.5	10,050
2	1	7,947.5	512.8	13,009
3	2	7,947.5	797.9	11,852
4	0	9,537	976.1	9,189
5	1	7,947.5	411.3	17,195
6	2	7,947.5	926.1	19,062

Table Data Pengujian

Data ke	Periode...	Pemaka...	Curah H...	Produks...
1	0	9,537	1,195.6	11,519
2	1	7,947.5	336.3	12,338
3	2	7,947.5	714.8	13,309

Table Data Prediksi

Data ke	Periode (X1)	Pemakaian...	Curah Huja...

Table hasil data uji

Data ke	X1	X2	X3	Output JST	Output Target	Error	rata-rata err...	Output Den...	output deno...	error	rata-rata err...	persentase	rata-rata pe...	Tingkat aku...	Rata-rata fi...
1	0	1	1	0.25	0.236	0.024	0.345	11,757.978	11,519	238.978	3,402.916	2.032	20.537	97.968	79.463
2	0.5	0	0	0.887	0.319	0.568		17,947.386	12,338	5,609.386		31.255		68.745	
3	1	0	0.44	0.187	0.629	0.442		11,034.617	15,395	4,360.383		28.323		71.677	

Table hasil prediksi

Data ke	X1	X2	X3	Output JST	Output Denormalisasi

Gambar 4. Hasil Implementasi dan pengujian

### 3. HASIL DAN PEMBAHASAN

Hasil implementasi menggunakan bahasa pemrograman Java dapat dilihat Gambar 3. Uji coba terhadap program dilakukan dengan memberikan *input* data latih dan data uji. Proses pengujian juga dilakukan dengan mengubah parameter *learning rate* dengan nilai antara 0.1 - 0.9 dengan iterasi maksimum 60000. Hasil pengujian berdasarkan perubahan nilai *learning rate* dapat dilihat pada Tabel 7. Sedangkan pengujian dengan perubahan parameter *hidden layer* dilakukan dengan mengubah banyak lapisan *hidden* dari 2 hingga 5 *layer*. Hasil pengujian berdasarkan parameter perubahan *hidden layer* dapat dilihat pada Tabel 8.

Berdasarkan Tabel 7 dapat dilihat bahwa semakin kecil *learning rate* maka jumlah iterasi akan semakin besar/banyak. Sebaliknya, semakin besar *learning rate* yang ditetapkan maka jumlah proses iterasi akan semakin kecil/sedikit. Namun, dapat dilihat juga bahwa semakin kecil *learning rate* yang ditetapkan maka tingkat akurasi data relatif lebih tinggi. Berdasarkan Tabel 8 dapat dikatakan bahwa semakin besar *hidden layer* yang ditetapkan maka tingkat rata - rata akurasi akan semakin tinggi. Sebaliknya, semakin kecil *hidden layer* maka rata-rata tingkat akurasi pun akan semakin rendah. Berdasarkan hasil pengujian didapatkan bahwa *learning rate* terbaik sebesar 0.3 dan *hidden layer* terbaik sebesar 9 *hidden layer*.

**Tabel 7.** Pengujian berdasarkan Learning Rate

Learning Rate	Percobaan Ke-	Jumlah Iterasi	max Iterasi	Target Error	Tingkat Akurasi	Rata-Rata Tingkat Akurasi
0.9	1	5520	60000	0.01	76.38	76.8171333
	2	8713			77.849	
	3	5292			76.385	
0.8	1	5976	60000	0.01	77.404	76.17033333
	2	8767			76.157	
	3	7515			74.95	
0.7	1	8506	60000	0.01	77.948	76.212
	2	12074			74.476	
	3	10454			76.424	
0.6	1	5777	60000	0.01	77.16	77.769
	2	8783			77.504	
	3	5775			79.988	
0.5	1	6967	60000	0.01	78.329	77.74166667
	2	10441			76.799	
	3	7689			78.097	
0.4	1	13330	60000	0.01	78.431	78.67133333
	2	14479			77.75	
	3	13184			79.833	
0.3	1	19300	60000	0.01	80.98	79.06633333
	2	13988			78.851	
	3	23784			77.368	
0.2	1	19.536	60000	0.01	78.334	78.02666667
	2	30066			78.642	
	3	14685			77.104	



Learning Rate	Percobaan Ke-	Jumlah Iterasi	max Iterasi	Target Error	Tingkat Akurasi	Rata-Rata Tingkat Akurasi
0.1	1	22522	60000	0.01	77.029	78.46166667
	2	42943			79.078	
	3	49046			79.278	

**Tabel 8.** Pengujian berdasarkan Jumlah Hidden Layer

Jumlah Hidden Layer	Hasil Uji (Periode Ke- )	Target Prediksi	Prediksi JST	Tingkat Akurasi %	Rata - Rata Tingkat Akurasi %
2	1	11519	9568.97	83.071	74.029
	2	12338	18050.38	68.353	
	3	15395	10878.535	70.663	
3	1	11519	10084.473	87.546	75.260
	2	12338	18187.591	67.837	
	3	15395	10837.673	70.397	
4	1	11519	10561.207	91.685	77.739
	2	12338	17157.671	71.910	
	3	15395	10718.23	69.622	
5	1	11519	10661.876	92.559	78.153
	2	12338	17489.483	70.545	
	3	15395	10985.015	71.354	
6	1	11519	10716.784	93.036	78.116
	2	12338	17768.436	69.438	
	3	15395	11065.243	71.876	
7	1	11519	11473.718	99.607	80.134
	2	12338	17882.256	68.996	
	3	15395	11053.703	71.801	
8	1	11519	11029.737	95.753	79.163
	2	12338	17645.686	69.921	
	3	15395	11056.192	71.817	
9	1	11519	11391.616	98.894	80.255
	2	12338	17574.865	70.203	
	3	15395	11033.441	71.669	

#### 4. KESIMPULAN

*Backpropagation* telah berhasil diimplementasikan dan diujikan untuk memprediksi jumlah panen/produksi buah kelapa sawit pada PT. Tunas Baru Lampung Tbk. menggunakan 6 data latih dan 3 data uji. Keakurasian tertinggi didapat pada prediksi periode Januari dan April sebesar 98%, dengan tingkat akurasi rata rata sebesar 79 %.

#### DAFTAR PUSTAKA

- [1] Sukanto. 2008. Kiat Meningkatkan Produktivitas dan Mutu Kelapa Sawit. Jakarta: Penebar Swadaya.  
V.V. Utari, A. Wanto, I. Gunawan, Sumarno, Z. M. Nasution, "PREDIKSI HASIL PRODUKSI KELAPA SAWIT PTPN IV BAHJAMBI MENGGUNAKAN ALGORITMA *BACKPROPAGATION*" *Journal of Computer System and Informatics (JoSYC)*, vol. 2, no. 3, pp. 271-279, 2021.
- [2] Y. Andriani, A. Wanto, Hendrizal, "JARINGAN SARAF TIRUAN DALAM MEMPREDIKSI PRODUKSI KELAPA SAWIT DI PT. KRE MENGGUNAKAN ALGORITMA LEVEMBERG MARQUARDT" Prosiding Seminar Nasional Riset Information Science (SENARIS), 2019.
- [3] J. Adhiva, Mustakim, S.A. Putri, S.G. Setyorini, "PREDIKSI HASIL PRODUKSI KELAPA SAWIT MENGGUNAKAN REGRESI PADA PT. PERKEBUNAN NUSANTARA V" Seminar Nasional Teknologi Informasi, Komunikasi dan Industri (SNTIKI) 12, Fakultas Sains dan Teknologi, UIN Sultan Syarif Kasim Riau, Pekanbaru, 2020.
- [4] H. Aini, Haviluddin, E. Budiman, M. Wati, N. Puspitasari, "PREDIKSI PRODUKSI MINYAK KELAPA SAWIT MENGGUNAKAN METODE *BACKPROPAGATION NEURAL NETWORK*," *Sains, Aplikasi, Komputasi dan Teknologi Informasi*, vol. 1, no. 1, pp. 24-33, 2019.
- [5] R.F.P. Sinaga, B.D. Setiawan, Marji, "PERAMALAN PRODUKSI KELAPA SAWIT MENGGUNAKAN JARINGAN SYARAF TIRUAN DENGAN METODE *BACKPROPAGATION*" *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 2, no. 11, pp. 4613-4620, 2018.
- [6] F. Irawan, Sumijan, Yuhandri, "PREDIKSI TINGKAT PRODUKSI BUAH KELAPA SAWIT DENGAN METODE SINGLE MOVING AVERAGE" *Jurnal Informasi dan Teknologi*, vol. 3, no. 4, pp. 251-256, 2021.
- [7] M.D. Wuryandari, I. Afrianto, "PERBANDINGAN METODE JARINGAN SARAF TIRUAN *BACKPROPAGATION* DAN *LEARNING VECTOR QUANTIZATION* PADA PENGENALAN WAJAH" *Jurnal Komputer dan Informatika (KOMPUTA)*. Edisi. I, vol. 1, pp. 45-51, 2012.
- [8] Livshin Igor, 2019. *Artificial Neural Networks With Java*. New York: Springer Science. Apress, Berkeley, CA. [https://doi.org/10.1007/978-1-4842-4421-0\\_1](https://doi.org/10.1007/978-1-4842-4421-0_1)