



Artikel Penelitian

## Penerapan Prediksi Produksi Padi Menggunakan *Artificial Neural Network* Algoritma Backpropagation

Hasdi Putra <sup>a,\*</sup>, Nabilah Ulfa Walmi <sup>b</sup>

<sup>a, b</sup> Program Studi Sistem Informasi Universitas Andalas, Kampus Unand Limau Manis, Padang 25163

### INFORMASI ARTIKEL

#### Sejarah Artikel:

Diterima Redaksi: 15 Juli 2020

Revisi Akhir: 02 September 2020

Diterbitkan Online: 05 September 2020

### KATA KUNCI

prediksi produksi padi  
artificial neural network  
backpropagation  
hidden layer  
akurasi

### KORESPONDENSI

E-mail: [hasdiputra@it.unand.ac.id](mailto:hasdiputra@it.unand.ac.id) \*

### A B S T R A C T

Prediksi produksi padi menjadi penting dilakukan untuk menunjang pembangunan nasional sektor pertanian pada suatu negara atau wilayah. Artificial Neural Network (ANN) termasuk metode yang terbaik dalam melakukan prediksi. Masalah utamanya adalah bagaimana menentukan jumlah neuron dan hidden layer yang optimal sehingga akurasi prediksinya tinggi. Artikel ini bertujuan untuk merancang arsitektu ANN untuk melakukan prediksi terhadap produksi padi menggunakan ANN dengan algoritma backpropagation. Tahapan penelitian yang dilakukan adalah mengumpulkan data produksi padi, melakukan pre-processing data, memproses prediksi, dan pengujian akurasi dan error serta implementasi. Dalam memproses prediksi dilakukan sesuai dengan rancangan model prediksi, yaitu parameter epoch, momentum, learning rate, hidden layer untuk menghasilkan keakuratan yang tinggi. Temuan yang diperoleh berupa rancangan optimal untuk melakukan prediksi yaitu dengan menggunakan multilayer. Hasil pengujian sistem prediksi produksi padi yang terdiri dari 75 kali pengujian pada di 19 daerah di Sumatera Barat, diperoleh tingkat akurasi mencapai 88,14% atau dengan tingkat error yang relatif rendah yaitu 11,86%.

## 1. PENDAHULUAN

Prediksi produksi padi merupakan suatu yang vital bagi Indonesia khususnya di sektor pertanian [1]. Dengan jumlah penduduk yang mencapai 267 juta orang serta tingkat konsumsi beras mencapai 124,89 kg/kapita/tahun, perencanaan pembangunan produksi padi menjadi semakin strategis [2]. Untuk efektifitas perencanaan pembangunan sektor pertanian, akurasi dari prediksi produksi padi menjadi semakin krusial [3]. Untuk memperoleh akurasi yang baik, beragam metode sudah dikembangkan. Metode dan teknik prediksi yang berkembang sudah direview pada penelitian sebelumnya [4], [5]. Support Vector Machines (SVM) memiliki kelebihan dalam menginterpretasi hasil, tingkat error yang rendah, dan juga dapat menghasilkan prediksi yang baik, tapi memiliki kelemahan pada saat *training* dengan kumpulan data besar sehingga *training* jadi lambat [6]. K-Nearest Neighbor (KNN) memiliki kelebihan efektif terhadap data yang berukuran

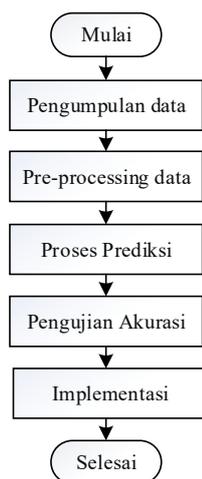
besar, dan handal terhadap data yang memiliki banyak noise, namun juga memiliki kekurangan yaitu perlunya menghitung satu persatu data *testing* terhadap semua data *training* [7], dan sensitif terhadap data *pre processing* [8].

Artificial Neural Network (ANN) atau Jaringan Syaraf Tiruan (JST) merupakan suatu sistem pemrosesan informasi dengan suatu karakteristik menyerupai sistem saraf pada manusia yang dapat memecahkan masalah SVM dan KNN dengan melakukan *training* data yang besar dan ANN memiliki kemampuan untuk mentoleransi kesalahan sehingga dapat menghasilkan prediksi yang baik [6], [9], [10]. Selain itu metode ini juga dapat digunakan untuk memodelkan hubungan yang kompleks antara masukan (*input*) dan keluaran (*output*) dalam menemukan pola-pola pada data [11]. Namun masalahnya adalah ANN memiliki kelemahan yaitu sulit untuk mengetahui berapa banyak neuron dan lapisan yang diperlukan, dan mengalami perlambatan saat *learning* [12].

Dalam metode ANN, terdapat beberapa algoritma yang telah digunakan salah satunya adalah *backpropagation* [13]. Algoritma *Backpropagation* merupakan salah satu prosedur yang paling populer, efektif, dan mudah dipelajari pada jaringan multilayer yang kompleks untuk mengoptimalkan pelatihan jaringan saraf tiruan [14]. *Backpropagation* melakukan pembelajaran terbimbing (*supervised learning*) yang digunakan pada jaringan *multi-layer* yang terdiri dari beberapa *hidden-layer* yang bertujuan untuk meminimalkan *error* terhadap jaringan yang menghasilkan keluaran (*output*) [15]. Menggunakan fungsi pelatihan (*training functions*) variabel laju pemahaman (*traingdx*) untuk mempercepat pelatihan *backpropagation*, yang merupakan kombinasi dari parameter laju pemahaman (*learning rate*) dan momentum [16] sehingga mendapatkan hasil yang relatif lebih akurat [17]. Makalah ini membahas bagaimana menentukan banyak neuron dan jumlah layer yang diperlukan untuk prediksi hasil produksi padi.

## 2. METODE

Dalam pelaksanaan penelitian ini, ada beberapa tahapan yang dilaksanakan yaitu mengumpulkan data, melakukan pemrosesan awal, memprediksi produksi padi, melakukan pengujian akurasi dan implementasi seperti pada gambar 1.



Gambar 1. Tahapan Penelitian

Data penelitian ini berasal dari Dinas Tanaman Pangan, Hortikultura, dan Perkebunan Provinsi Sumatera Barat sebagai institusi yang bertanggungjawab dalam mewujudkan kemandirian pangan dan meningkatkan kualitas hasil pertanian.

### 2.1. Pengumpulan Data

Data yang dikumpulkan merupakan data produksi padi selama 5 (lima) tahun, yang berasal dari 19 Kota/Kabupaten di Sumatera Barat. Data yang dikumpulkan untuk masing-masing kota/kabupaten adalah; (1) data target luas tanam padi (Ha), (2) data target luas panen padi (Ha), (3) data target produktifitas padi (ton/Ha), (4) data target produksi padi (ton).

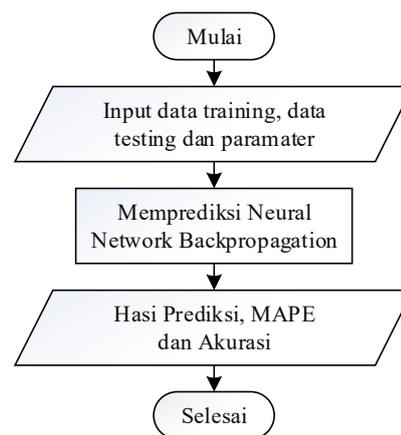
### 2.2. Pre-Processing Data

Pada tahap ini, target luas tanam padi, target luas panen padi dan target produktifitas padi dari tahun 2013-2017 dan target produksi padi dari tahun 2013-2018 tersebut dikelompokkan menjadi 2

jenis data, yaitu data latih (*training*) dan data uji (*testing*) sebagai data input. Tahap ini bertujuan juga untuk mempersiapkan data agar dapat digunakan dalam prediksi. Selanjutnya melakukan normalisasi atau transformasi data, karena lapisan tersembunyi (*hidden layer*) menggunakan fungsi aktivasi *sigmoid biner* atau *logsig* untuk lapisan tersembunyi dan *purelin* untuk lapisan keluaran. Proses normalisasi atau transformasi memiliki tujuan untuk mempermudah perhitungan dan mendapatkan hasil prediksi yang lebih akurat.

### 2.3. Proses Prediksi Produksi Padi

Proses prediksi produksi padi menggunakan metode Artificial Neural Network (ANN) *Backpropagation* dapat ditunjukkan seperti pada gambar 2 berikut;



Gambar 2. Flowchart Prediksi Produksi Padi dengan Jaringan Saraf Tiruan *Backpropagation*

Pada rancangan prediksi produksi padi menggunakan ANN *Backpropagation* ini, dimulai dengan melakukan penginputan data uji dan data latih. Data uji dan data latih dapat dikatakan sebagai variabel input dan data target. variabel input terdiri dari data target luas tanam padi, target luas panen padi dan target produktivitas padi tahun 2013-2017. Dan variabel target yaitu data target produksi padi tahun 2013-2017. Output merupakan hasil prediksi produksi padi tahun 2018.

Untuk merancang arsitektur metode ANN *Backpropagation* yang menghasilkan prediksi yang optimal, maka dilakukan penentuan atau penginputan jumlah nilai parameter-parameter yang digunakan, seperti jumlah *hidden layer*, laju pemahaman (*learningrate*), maksimum iterasi (*epoch*), dan momentum dengan menggunakan proses *trial and error*. Setelah menentukan jenis data dan parameter, prediksi dilakukan dengan metode ANN *Backpropagation* berdasarkan langkah-langkah arahan Laurene Fausett [18]. Selanjutnya, hasil prediksi yang diperoleh dievaluasi nilai *error*nya dengan menggunakan MAPE untuk memperoleh tingkat akurasi pada prediksi.

### 2.4. Pengujian akurasi dan error

Pada tahapan ini dilakukan pengujian hasil prediksi yang sudah diperoleh dengan cara melihat tingkat keakuratan dan *error* pada sistem. Pengujian dilakukan kepada masing-masing data *training* dan data *testing*. Pengujian bertujuan untuk mengetahui sistem kerja *input*, proses dan *output* apakah sesuai dengan tujuan yang diharapkan. Cara menghitung nilai *error* digunakan MAPE (*Mean*

*Average Percentage Error*). Nilai MAPE menunjukkan seberapa banyak kesalahan dalam hasil prediksi dibandingkan dengan nilai aktual. *Mean Absolute Percentage Error* (MAPE) dihitung menggunakan kesalahan mutlak dalam setiap periode dibagi dengan nilai-nilai yang diamati yang jelas untuk periode tersebut. Semakin kecil nilai MAPE maka semakin baik kinerja prediksi [27]. Untuk menghitung akurasi terhadap prediksi yang dilakukan, maka sebuah akurasi 100% dikurangi dengan nilai MAPE [19].

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

$$Akurasi = 100\% - MAPE \dots\dots\dots (2)$$

Dimana  $x_t$  adalah nilai target data asli pada periode  $t$ , sedangkan  $y_t$  merupakan nilai prediksi pada periode  $t$  dan  $n$  adalah banyaknya data prediksi pada periode  $t$ . MAPE digunakan dalam penelitian ini untuk mengevaluasi kinerja berbagai jenis model prediksi. Semakin kecil MAPE maka semakin baik model prediksinya [20] seperti pada tabel 1.

Tabel 1. Kriteria Skala MAPE

Skala MAPE	Kriteria
<10%	Sangat Baik
10%-20%	Baik
20%-50%	Cukup
>50%	Buruk

### 2.5. Implementasi

Pada tahapan ini, hasil penelitian prediksi jaringan saraf tiruan *backpropagation* diterapkan pada GUI MATLAB agar mempermudah *user* dalam pengoperasian model yang telah dibangun.

## 3. HASIL

### 3.1. Pengumpulan Data

Hasil dari pengumpulan data adalah berupa data target luas tanam padi, target luas panen padi, dan target produktifitas padi dari tahun 2013-2017 serta target produksi padi dari tahun 2013-2018. Pada tabel 2 berikut ini akan menunjukkan data yang digunakan.

Tabel 2. Data Target Produksi Tahun 2013

Kabupaten/Kota	X1	X2	X3	Y
A= Kota Padang	13.511	13.323	5,64	75.168
B= Kota Bukittinggi	836	808	5,84	4.715
C= Kota Payakumbuh	7.315	7.258	5,55	40.282
D= Kota Padang Panjang	1.675	1.782	5,18	9.231
E= Kota Solok	2.296	2.196	5,90	12.956
F= Kota Sawahlunto	2.841	2.785	5,01	13.953
G= Kota Pariaman	6.063	6.179	5,10	31.513
H= Kabupaten Pasaman	50.887	46.218	5,11	236.266
I= Kabupaten Pasaman Barat	26.631	27.126	4,67	126.678
J= Kabupaten Lima Puluh Kota	47.123	47.530	4,85	230.521
K= Kabupaten Agam	55.553	55.287	5,44	300.706
L= Kabupaten Tanah Datar	43.525	43.742	5,75	251.517
M. Kab. Padang Pariaman	50.806	52.045	5,08	264.389
N= Kabupaten Solok	61.360	60.655	5,56	337.242
O= Kabupaten Solok Selatan	29.521	26.788	5,00	133.941
P= Kabupaten Sijunjung	18.339	18.689	4,52	84.530
Q= Kabupaten Dharmasraya	15.670	15.230	4,60	70.058

R= Kabupaten Pesisir Selatan	57.483	57.494	5,00	287.470
S= Kabupaten Mentawai	119	296	3,50	1.036

X1= target luas tanam padi (Ha); X2 = target luas panen padi (Ha); X3 = target produktivitas padi (Ton/Ha); Y = target produksi padi (Ton).

### 3.2. Pre-Processing

Data produksi padi dan variabel lainnya dibagi atas dua kelompok yaitu sebagai data latih (training) dan data uji (testing). Maka proporsi pembagian data training dan data testing yang digunakan dalam penelitian ini yaitu 70% data training berjumlah 66 data setiap variabel dan 30% data testing berjumlah 29 data setiap variabel. Setelah data terbagi menjadi dua bagian, selanjutnya dilakukan pemeriksaan agar data tersebut *valid* dan siap digunakan untuk melakukan prediksi produksi padi. Setiap data *input* dilakukan normalisasi. Setelah data terbagi menjadi dua bagian, selanjutnya akan dilakukan pemeriksaan agar tidak ada yang *missing* hingga data tersebut *valid* dan dapat digunakan dalam melakukan prediksi produksi padi. Karena pada penelitian ini menggunakan fungsi aktivasi *sigmoid biner* yang mempunyai rentang nilai 0 hingga 1, maka setiap data *input* melakukan normalisasi atau transformasi. Normalisasi data dilakukan dengan menggunakan persamaan (3) berikut :

$$X' = \frac{0,8(X - X_{min})}{X_{max} - X_{min}} + 0,1 \dots\dots\dots (3)$$

Data primer yang digunakan merupakan data dari Dinas Tanaman Pangan, Hortikultura, dan Perkebunan Provinsi Sumatera Barat diproses terlebih dahulu sebelum diolah pada MATLAB. Data tersebut dimasukkan ke dalam *Microsoft Excel*, data di kelompokkan berdasarkan data *input* dan data target. Lalu dilakukan proses normalisasi. Setelah itu data siap untuk di *input* pada MATLAB. Pada tabel 3 berikut ini akan menunjukkan hasil data setelah melakukan proses normalisasi pada data tahun 2013.

Tabel 3. Data Normalisasi

Kabupaten/ Kota	X1	X2	X3	Y
A	0.2325	0.2306	0.1003	0.2446
B	0.1079	0.1077	0.1003	0.1072
C	0.1716	0.1710	0.1003	0.1765
D	0.1162	0.1172	0.1002	0.1160
E	0.1223	0.1213	0.1003	0.1232
F	0.1276	0.1271	0.1002	0.1252
G	0.1593	0.1604	0.1002	0.1594
H	0.5998	0.5539	0.1002	0.5587
I	0.3614	0.3663	0.1002	0.3450
J	0.5628	0.5668	0.1002	0.5475
K	0.6456	0.6430	0.1003	0.6844
L	0.5274	0.5296	0.1003	0.5885
M	0.5990	0.6112	0.1002	0.6136
N	0.7027	0.6958	0.1003	0.7556
O	0.3898	0.3630	0.1002	0.3592
P	0.2799	0.2834	0.1002	0.2628
Q	0.2537	0.2494	0.1002	0.2346
R	0.6646	0.6647	0.1002	0.6586
S	0.1009	0.1026	0.1001	0.1000

X1: target luas tanam padi yang telah dinormalisasi, X2: target luas panen padi yang telah dinormalisasi. X3: target produktivitas padi yang telah dinormalisasi dan Y: target produksi padi yang telah dinormalisasi

### 3.3. Proses Prediksi

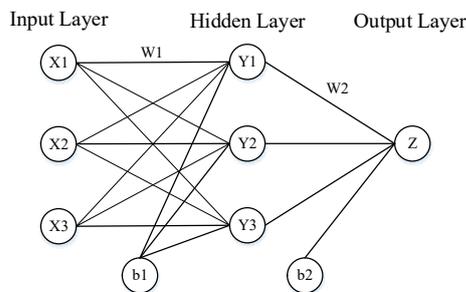
#### 3.3.1. Perancangan Arsitektur Model ANN

Rincian rancangan model prediksi produksi padi seperti yang ditunjukkan pada Tabel 4

Tabel 4. Rancangan Model ANN Multilayer Prediksi

Parameter	Jumlah	Keterangan
<i>Input Layer</i>	3 neuron	Target Luas Tanam Padi; Target Luas Panen Padi; Target Produktifitas Padi
<i>Hidden Layer</i>	trial and error	3 neuron
<i>Output Layer</i>	1 neuron	Produksi padi
Bobot Awal	trial and error	BB : 0.05 BA : 0.08 Bilangan acak antara 0.05-0.08
<i>Learning Rate</i>	Trial and error	0.1-0.5
<i>Momentum</i>	Trial and error	0.5-0.9
<i>Epoch</i>	Trial and error	100-200 (dengan interval 50)
Fungsi Aktivasi	2	Logsig dan Purenlin

Dari tabel 4 dapat digambarkan arsitekturnya seperti gambar 3 berikut;

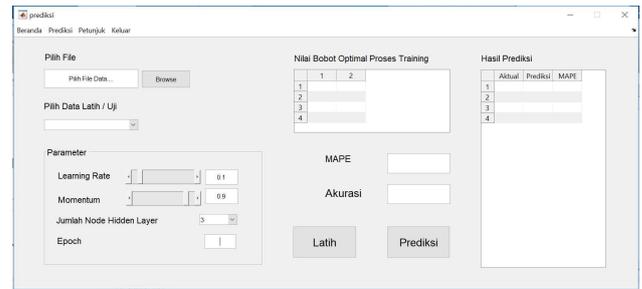


Gambar 3. Arsitektur ANN Prediksi Produksi Padi

Arsitektur ANN yang dirancang menggunakan tiga neuron pada input layer yaitu target luas tanam padi, target luas panen padi dan target produktifitas padi. Hidden layer terdiri dari tiga (3) neuron diperoleh dari proses trial dan error. Bobot yang digunakan adalah rentang 0.05 sampai dengan 0.08. Output layer berupa prediksi produksi padi dengan 1 neuron.

#### 3.3.2. Aplikasi Pengujian Prediksi

Pada halaman prediksi terdapat *pop up menu* untuk memilih untuk memprediksi data latih atau data uji yang telah di *input*-kan sebelumnya pada Microsoft Excel. Setelah data uji atau data input dipilih, lalu *input* nilai-nilai pada setiap parameter yaitu, *Epoch*, *Learning Rate*, *Momentum*, dan *Hidden Layer*. Saat semua nilai telah terisi, maka pengguna menekan *button* prediksi untuk melakukan proses prediksi dengan metode ANN *Backpropagation*. Lalu sistem akan menampilkan hasil prediksi pada tabel hasil prediksi padi, yang dibedakan atas dua kolom, yaitu kolom aktual (target produksi) yang berasal dari variabel (data) target produksi padi, dan kolom prediksi padi yang merupakan hasil prediksi sistem. Setelah hasil prediksi ditampilkan, maka dilakukan perhitungan *error* dan akurasi. Perhitungan *error* dilakukan dengan rumus MAPE. Sistem akan menghasilkan akurasi 100% jika nilai *error* bernilai 0, atau dapat dikatakan hasil prediksi sistem mirip atau sama dengan nilai aktual. Antarmuka aplikasi pengujian prediksi produksi padi seperti ditunjukkan pada gambar 4.



Gambar 4. Antarmuka Halaman Prediksi

#### 3.3.3. Pengujian Aplikasi

Setelah sistem selesai dibangun, maka dilakukan pengujian untuk menghitung akurasi dan nilai *error* pada prediksi produksi padi menggunakan jaringan saraf tiruan *backpropagation*. Ada dua jenis pengujian, yaitu pengujian fungsionalitas sistem dan pengujian aplikasi. Pengujian sistem ini dilakukan dengan fungsionalitas dari sudut pandang *user* (pengguna). Format dan pengujian yang dilakukan terhadap sistem ditunjukkan pada tabel 5 berikut ini.

Tabel 5. Pengujian Fungsionalis Sistem

No.	Komponen Pengujian	Input	Output	Status
1.	Halaman Beranda	User memilih menu beranda	Menampilkan halaman beranda	OK
2.	Halaman Prediksi	User memilih menu prediksi	Menampilkan halaman prediksi	OK
3.	Halaman Petunjuk	User memilih menu petunjuk	Menampilkan halaman petunjuk	OK

Dari tabel 5 berdasarkan pengujian *input* dan *output* pada setiap komponen pengujian yaitu halaman beranda, halaman prediksi, dan halaman petunjuk dapat dikatakan bahwa fungsionalitas sistem berhasil. Hasil pengujian parameter prediksi produksi padi menggunakan jaringan saraf tiruan *backpropagation* seperti yang ditunjukkan pada tabel 6.

Tabel 6. Hasil Prediksi dengan Pengujian Parameter

Epoch	Momen tum	Learning Rate	MAPE (%)	Akurasi (%)
200	0,5	0,1	13,271129	86,728871
		0,2	12,005893	87,994107
		0,3	11,886828	88,113172
		0,4	11,865150	88,134850
		0,5	11,861308	88,138692
	0,6	0,1	13,274849	86,725151
		0,2	12,006093	87,993907
		0,3	11,886851	88,113149
		0,4	11,865101	88,134899
		0,5	11,861311	88,138689
	0,7	0,1	13,273208	86,726792
		0,2	12,005597	87,994403
0,3		11,886781	88,113219	
0,4		11,902695	88,097305	
0,5		11,861312	88,138688	
0,8	0,1	13,275736	86,724264	
	0,2	12,006159	87,993841	
	0,3	11,886636	88,113364	
	0,4	11,865135	88,134865	
	0,5	11,861316	88,138684	
0,9	0,1	13,276099	86,723901	

0,2	12,005949	87,994051
0,3	11,886803	88,113197
0,4	11,865106	88,134894
0,5	11,861320	88,138680

Berdasarkan hasil pengujian sistem terhadap rancangan model parameter-parameter yang telah ditentukan, maka diperoleh parameter yang menghasilkan nilai MAPE terkecil dan akurasi terbesar, yaitu dengan *epoch* sebesar 200, *momentum* sebesar 0,5, dan *learning rate* sebesar 0,5. Nilai MAPE yang didapatkan yaitu 11,86 % dan nilai akurasi atau keakuratan prediksi sebesar 88,14%.

3.3.4. Post-Processing Data

Setelah proses prediksi, maka akan didapatkan nilai *output* atau hasil prediksi, dimana data awal sebagai data *input* telah dilakukan proses normalisasi atau transformasi. Jadi untuk data prediksi yang baru dihasilkan perlu dilakukan proses denormalisasi. Proses ini disebut dengan *post-processing*, yaitu proses dimana nilai *output* yang menghasilkan nilai dengan rata-rata sama dengan nol, maka dalam proses ini akan mengembalikan nilai tersebut sesuai dengan data awal atau data yang asli. Nilai *output* yang dihasilkan pada proses simulasi adalah nilai dari data yang telah dinormalisasi pada langkah awal. Denormalisasi data dilakukan dengan menggunakan persamaan (4) berikut:

$$X = \frac{(X' - 0,1) \times X_{max} - X_{min}}{0,8} + X_{min} \dots\dots\dots (4)$$

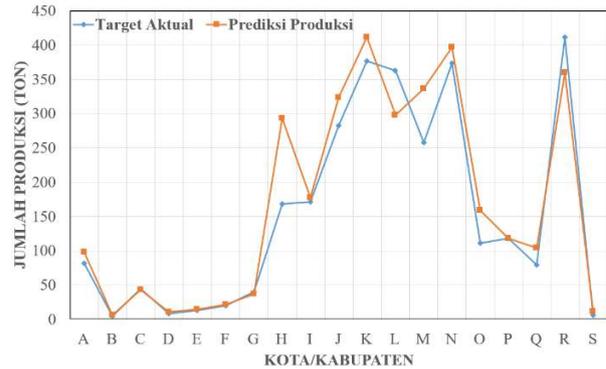
Setelah seluruh proses prediksi dilakukan pada MATLAB, data hasil prediksi disalin, dipindahkan, dan diolah pada Microsoft Excel untuk melakukan proses denormalisasi. Prediksi produksi padi yang dilakukan dengan menggunakan metode jaringan saraf tiruan *backpropagation* sehingga diperoleh model yang menghasilkan akurasi dan tingkat kesalahan (*error*) yang optimal. Hasil denormalisasi sekaligus hasil prediksi produksi padi pada kabupaten/kota di provinsi Sumatera Barat untuk tahun 2018 ditunjukkan pada tabel 7 berikut.

Tabel 7. Hasil Denormalisasi Prediksi Produksi

Kabupaten/Kota	Sebelum Denormalisasi	Setelah Denormalisasi
A	0,288718	97.808
B	0,110297	6.316
C	0,182405	43.292
D	0,119414	10.991
E	0,126920	14.84
F	0,139737	21.413
G	0,170091	36.977
H	0,670671	293.667
I	0,443951	177.408
J	0,728241	323.188
K	0,900000	411.263
L	0,679379	298.132
M	0,754264	336.532
N	0,872649	397.238
O	0,408359	159.157
P	0,328559	118.237
Q	0,301161	104.188

R	0,799787	359.876
S	0,121186	11.9

Untuk memprediksi target produksi padi pada tahun tertentu, sudah berhasil di rancang suatu aplikasi menggunakan metode Artificial Neural Network algoritma Backpropagation. Aplikasi tersebut sudah dilakukan pengujian untuk memprediksi target produksi tahun berikutnya. Dari pengolahan data, diperoleh grafik perbandingan jumlah produksi padi aktual yang dicapai pada tahun 2018 dengan prediksi target produksi padi pada Kota/Kabupaten di Sumatera Barat, Indonesia seperti yang ditunjukkan oleh gambar 5.



Gambar 5. Perbandingan Aktual Produksi dengan Prediksi

Dari perbandingan target aktual produksi dengan prediksi target produksi, diperoleh tingkat akurasi hasil prediksi tersebut. Persentase keakuratan prediksi target produksi padi yang dihasilkan aplikasi dengan produksi padi aktual dapat ditunjukkan pada Tabel 8.

Tabel 8. Persentase Akurasi Prediksi Produksi Padi

Kota/ Kabupaten	Produksi Aktual (Ton)	Prediksi Produksi (Ton)	Akurasi Prediksi (%)
A	168.712	293.667	42,931
B	171.438	177.408	97,307
C	282.574	323.188	87,797
D	376.965	411.263	91,971
E	362.553	298.132	84,394
F	257.734	336.532	74,414
G	374.063	397.238	94,538
H	111.424	159.157	70,475
I	118.473	118.237	99,860
J	79.690	104.188	81,146
K	411.263	359.876	88,865
L	45.081	43.292	98,123
M	4.520	6.316	96,720
N	8.765	10.991	96,227
O	82.054	97.808	88,092
P	12.883	14.840	96,900
Q	19.987	21.413	97,971
R	6.505	11.900	90,493
S	40.224	36.977	96,411

## 4. PEMBAHASAN

### 4.1. Perancangan Arsitektur ANN

Teknik perancangan arsitektur sudah banyak dibahas [21], [22], [23]. Rancangan menggunakan *multilayer* maka layer terbagi atas 3 layer yaitu [24]; (1) lapisan masukan (input layer) yang jumlahnya dapat disesuaikan dengan variabel input. (2) Lapisan tersembunyi (hidden layer) yang jumlahnya didapatkan dengan proses trial and error yaitu berjumlah 3 disesuaikan dengan jumlah variabel input. Pada lapisan ini digunakan fungsi aktivasi sigmoid biner logsig yang berguna dalam mengirimkan suatu informasi dari input layer melalui bobot ke neuron pada *hidden layer*. (3) Lapisan keluaran (output layer) yang jumlahnya disesuaikan dengan variabel output. Pada lapisan ini digunakan fungsi aktivasi purelin yang berguna untuk menghasilkan output yang memiliki range yang sama dengan *input*. Desain yang diusulkan adalah multilayer dengan input 3 neuron, hidden layer trail and error dan output 1 neuron. Desain multilayer cenderung mempunyai akurasi yang baik [25]. Hal yang sama juga dilakukan dan lebih efektif [26].

Variabel penelitian dibagi menjadi 2 yaitu data input dan data target. Untuk variabel input data yang digunakan adalah target luas tanam padi, target luas panen padi, dan target produktivitas padi dari tahun 2013-2017. Variabel target adalah target produksi padi pada setiap kabupaten/kota di Sumatera Barat dari tahun 2013-2018. Karena menggunakan proses trial and error, jadi diperlukan penentuan terhadap parameter-parameter yang digunakan. Arsitektur layer yang digunakan yaitu multilayer yang terdiri dari lapisan input, hidden, dan output, pada input layer telah ditentukan atas 3 variabel yaitu produksi, luas panen, dan produktivitas padi, pada hidden layer jumlah neuron tersembunyi harus kurang dari dua kali ukuran input layer [27], karena jumlah neuron yang optimal ditentukan dengan proses trial and error sehingga jumlah hidden layer berjumlah kurang dari 6 neuron yaitu 3 hidden layer yang juga disamakan dengan jumlah variabel input.

Selanjutnya untuk fungsi aktivasi pada lapisan tersembunyi digunakan pada proses pelatihan adalah logsig, dan fungsi aktivasi pada lapisan keluaran adalah purelin. Bobot diambil dari nilai random yang cukup kecil, biasanya bobot awal diinisialisasi dengan nilai antara -0,5 sampai 0,5 (atau -1 sampai 1) [18]. Setelah melakukan trial and error, maka nilai yang tepat untuk dijadikan bobot awal yaitu 0.05 dan 0.08. Parameter Momentum berguna untuk menurunkan gradient dengan momentum dengan memberi nilai 0.9 sesuai default. Dengan melakukan perubahan momentum akan berdampak terhadap waktu proses, jadi semakin besar nilai momentum maka semakin cepat waktu proses yang terjadi [28]. Karena menggunakan proses trial and error maka momentum dilakukan dengan rentang nilai dari 0.5 - 0.9.

Parameter epoch (iterasi) sebagai parameter perulangan untuk pemberhentian pada proses pelatihan. Apabila jumlah iterasi telah mencapai jumlah maksimum maka pelatihan akan berhenti. Pada penelitian ini menentukan maksimum 1000 *epochs* untuk mendapatkan hasil yang baik, sesuai dengan default fungsi pelatihan variabel laju pemahaman *traingdx*. Jika maksimum

epoch tinggi, maka memerlukan waktu yang lama dalam melakukan pelatihan dan jika epoch rendah, maka tidak dihasilkan akurasi yang baik. Dengan menggunakan proses *trial and error* maka diberi rentang 100 hingga 150 *epochs*, karena dengan dengan sitem/aplikasi ini dapat memungkinkan iterasi yang tidak terlalu besar namun bisa mencapai hasil yang baik. Parameter laju pembelajaran (learning rate) berguna untuk mempercepat laju iterasi (epoch). Bila nilai learning rate-nya semakin kecil, maka ketelitian/ keakuratan akan semakin besar, namun terdapat resiko yaitu proses iterasi akan memakan waktu yang semakin lama. Karena jumlah learning rate ditentukan dengan proses trial and error maka rentang nilai untuk learning rate adalah 0,1 hingga 0.5.

### 4.2. Implementasi Aplikasi Pengujian

Implementasi aplikasi pengujian dirancang menggunakan tampilan seperti pada gambar 3. Dengan adanya GUI (graphic user interface) memudahkan user dalam menggunakan aplikasi yang sudah dibangun [29], Dari hasil pengujian dapat ditunjukkan bahwa pada halaman prediksi terdapat *pop up menu* untuk memilih untuk memprediksi data latih atau data uji yang telah di *input*-kan sebelumnya pada Microsoft Excel. Setelah data uji atau data input dipilih, lalu *input* nilai-nilai pada setiap parameter yaitu, *Epoch*, *Learning Rate*, *Momentum*, dan *Hidden Layer*.

Pengujian aplikasi dilakukan dengan menjalankan program dan melakukan proses prediksi sesuai dengan nilai-nilai *input* setiap parameter yang terdapat pada rancangan model prediksi tabel 4 yaitu sebanyak 75 kali untuk mendapatkan hasil akurasi yang tinggi dan nilai *error* yang rendah. Sehingga tingkat keakuratan sistem akan bergantung dari hasil yang diperoleh. Penentuan parameter model prediksi dilakukan terhadap arsitektur jaringan saraf tiruan terbaik pada hasil uji coba (*trial and error*) [30]. Tujuan uji coba parameter adalah untuk mendapatkan nilai prediksi yang baik dengan mengetahui nilai parameter paling optimal menggunakan *Mean Absolute Percentage Error* (MAPE) [31] dengan menggunakan model yang telah dibentuk sebelumnya. Nilai MAPE didapatkan dengan membandingkan hasil prediksi dengan data aktual, semakin kecil nilai MAPE maka semakin baik kinerja prediksi.

### 4.3. Prediksi Produksi Padi

Berdasarkan hasil pengujian sistem terhadap rancangan model parameter-parameter yang telah ditentukan, maka diperoleh parameter yang menghasilkan nilai MAPE terkecil dan akurasi terbesar, yaitu dengan *epoch* sebesar 200, *momentum* sebesar 0,5 dan *learning rate* sebesar 0,5. Nilai MAPE yang didapatkan yaitu 11,86 % dan nilai akurasi atau keakuratan prediksi sebesar 88,14% dengan akurasi tertinggi mencapai 98,89%. Hasil ini sangat baik dengan perolehan akurasi yang tinggi. Sebelumnya juga ada dengan akurasi yang bervariasi [4] untuk beragam metode [32]. Untuk satu algoritma menghasilkan output dengan tingkat akurasi yang tidak sama juga.

Hasil prediksi ini bisa juga dibandingkan dengan beberapa hasil prediksi produksi padi menggunakan Backpropagation Neural Network dengan akurasi mencapai 89,56 % pada [33] dan juga 94,90%. Sedangkan untuk desainnya dengan menggunakan jumlah neuron yang sama untuk inputnya yaitu 3 neuron, dan hidden layernya 4, *learning rate* 0,8.

## 5. KESIMPULAN

Rancangan arsitektur Artificial Neural Network (ANN) sudah berhasil dibuat dan diterapkan untuk prediksi produksi padi di Sumatera Barat, Indonesia. Rancangan ANN yang menggunakan multilayer pada aplikasi sudah berhasil diujikan dengan memperoleh hasil prediksi produksi padi untuk periode berikutnya. Dari implementasi sistem yang dirancang, diperoleh hasil prediksi dengan akurasi mencapai 88,14%. Patut dicatat bahwa proses trial dan error perlu mendapat perhatian untuk mendapatkan hasil yang maksimal dengan durasi waktu yang lebih baik. Aplikasi prediksi produksi padi yang dibuat telah mampu melakukan prediksi produksi padi untuk tahun 2018 dengan melakukan 75 kali pengujian terhadap parameter yang terdapat pada rancangan model prediksi sehingga didapatkan model dengan parameter yang optimal meliputi, *input layer* berjumlah 3, *hidden layer* berjumlah 3, dan *output layer* berjumlah 1 dengan *epoch* sebesar 200, *momentum* sebesar 0,5, dan *learning rate* sebesar 0,5.

## DAFTAR PUSTAKA

- [1] H. Sutanta, A. R. Gunawan, and Y. Wibisono, "Calculation of rice field embankment coefficient using high-resolution satellite imagery," *IOP Conf. Ser. Earth Environ. Sci.*, vol. 500, p. 012049, Jul. 2020.
- [2] Badan Pusat Statistik, "Luas Panen dan Produksi Padi di Indonesia 2019 Hasil Survey Kerangka Sampel Area (KSA)," 2020. [Online]. Available: [https://www.bps.go.id/website/materi\\_ind/materiBrsInd-20200204112508.pdf](https://www.bps.go.id/website/materi_ind/materiBrsInd-20200204112508.pdf).
- [3] M. Al-Amin, D. Z. Karim, and T. A. Bushra, "Prediction of rice disease from leaves using deep convolution neural network towards a digital agricultural system," *2019 22nd International Conference on Computer and Information Technology, ICCIT 2019*. 2019.
- [4] D. Shah, H. Isah, and F. Zulkernine, "Stock market analysis: A review and taxonomy of prediction techniques," *Int. J. Financ. Stud.*, vol. 7, no. 2, 2019.
- [5] H. Putra, N. U. Walmi, and A. D. Kartika, "Data Mining Approach For Prediction Of Rice Production Using Backpropagation Artificial Neural Network Method," in *The International Conference on ASEAN 2019*, 2019, pp. 321–326.
- [6] N. R. Dzakiyullah, B. Hussin, C. Saleh, and A. M. Handani, "Comparison neural network and support vector machine for production quantity prediction," *Adv. Sci. Lett.*, vol. 20, no. 10–12, pp. 2129–2133, 2014.
- [7] S. García, J. Derrac, J. R. Cano, and F. Herrera, "Prototype selection for nearest neighbor classification: Taxonomy and empirical study," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 3, pp. 417–435, 2012.
- [8] G. F. Fan, Y. H. Guo, J. M. Zheng, and W. C. Hong, "Application of the weighted k-nearest neighbor algorithm for short-term load forecasting," *Energies*, vol. 12, no. 5, 2019.
- [9] N. A. Almansour *et al.*, "Neural network and support vector machine for the prediction of chronic kidney disease: A comparative study," *Comput. Biol. Med.*, vol. 109, no. October 2018, pp. 101–111, 2019.
- [10] S. M. Gorade and P. A. Deo, "A Study Some Data Mining Classification Techniques," *Int. J. Mod. Trends Eng. Res.*, vol. 4, no. 1, pp. 210–215, 2017.
- [11] J. Fei and C. Lu, *Adaptive Sliding Mode Control of Dynamic Systems Using Double Loop Recurrent Neural Network Structure*, vol. 29, no. 4. 2018.
- [12] B. Das, B. Nair, V. K. Reddy, and P. Venkatesh, "Evaluation of multiple linear, neural network and penalised regression models for prediction of rice yield based on weather parameters for west coast of India," *Int. J. Biometeorol.*, vol. 62, no. 10, pp. 1809–1822, 2018.
- [13] L. Wang, Y. Zeng, and T. Chen, "Back propagation neural network with adaptive differential evolution algorithm for time series forecasting," *Expert Syst. Appl.*, vol. 42, no. 2, pp. 855–863, 2015.
- [14] H. Elarabi, "Comparison of Different Methods of Application of Neural Network on Soil Profile of Khartoum State," *Int. J. Sci. Technol. Soc.*, vol. 2, no. 3, p. 59, 2014.
- [15] N. A. Hamid, N. M. Nawi, R. Ghazali, and M. N. M. Salleh, "Accelerating learning performance of back propagation algorithm by using adaptive gain together with adaptive momentum and adaptive learning rate on classification problems," *Int. J. Softw. Eng. its Appl.*, vol. 5, no. 4, pp. 31–44, 2011.
- [16] O. Krestinskaya, K. N. Salama, and A. P. James, "Learning in memristive neural network architectures using analog backpropagation circuits," *IEEE Trans. Circuits Syst. I Regul. Pap.*, vol. 66, no. 2, pp. 719–732, 2019.
- [17] G. Amaral *et al.*, *Modern Database Management*, vol. 369, no. 1. 2013.
- [18] Laurene V. Fausett, *Fundamentals Of Neural Networks: Architectures, Algorithms And Applications*, no. 888. 2007.
- [19] O. M. Rezapour, L. T. Shui, and A. A. Dehghani, "Review of genetic algorithm model for suspended sediment estimation," *Aust. J. Basic Appl. Sci.*, vol. 4, no. 8, pp. 3354–3359, 2010.
- [20] P. C. Chang, Y. W. Wang, and C. H. Liu, "The development of a weighted evolving fuzzy neural network for PCB sales forecasting," *Expert Syst. Appl.*, vol. 32, no. 1, pp. 86–96, 2007.
- [21] P. K. Sathy, N. K. Barpanda, A. K. Rath, and S. K. Behera, "Nitrogen Deficiency Prediction of Rice Crop Based on Convolutional Neural Network," *J. Ambient Intell. Humaniz. Comput.*, no. 0123456789, 2020.
- [22] A. Wanto, A. P. Windarto, D. Hartama, and I. Parlina, "Use of Binary Sigmoid Function And Linear Identity In Artificial Neural Networks For Forecasting Population Density," *IJISTECH (International J. Inf. Syst. Technol.)*, vol. 1, no. 1, p. 43, 2017.
- [23] I. N. da Silva, D. H. Spatti, R. A. Flauzino, L. H. B. Liboni, and S. F. dos Reis Alves, "Artificial neural networks: A practical course," *Artif. Neural Networks A Pract. Course*, pp. 1–307, 2016.
- [24] G. Deshpande, P. Wang, D. Rangaprakash, and B. Wilamowski, "Fully connected cascade artificial neural network architecture for attention deficit hyperactivity disorder classification from functional magnetic resonance imaging data," *IEEE Trans. Cybern.*, vol. 45, no. 12, pp. 2668–2679, 2015.
- [25] R. B. Santos, M. Rupp, S. J. Bonzi, and A. M. F. Fileti, "Comparison between multilayer feedforward neural networks and a radial basis function network to detect and locate leaks in pipelines transporting gas," *Chem. Eng. Trans.*, vol. 32, pp. 1375–1380, 2013.
- [26] A. A. Heidari, H. Faris, I. Aljarah, and S. Mirjalili, "An efficient hybrid multilayer perceptron neural network with grasshopper optimization," *Soft Comput.*, vol. 23, no. 17, pp. 7941–7958, 2019.
- [27] F. S. Panchal and M. Panchal, "Review on Methods of Selecting Number of Hidden Nodes in Artificial Neural

- Network,” *Int. J. Comput. Sci. Mob. Comput.*, vol. 311, no. 11, pp. 455–464, 2014.
- [28] V. T. Widyaningrum and A. S. Romadhon, “Pengaruh Pemberian Momentum Pada,” *Semin. Nas. Sains dan Teknol. 2014*, no. November, 2014.
- [29] V. Prema and K. Uma Rao, “Interactive Graphical User Interface (GUI) for Wind Speed Prediction Using Wavelet and Artificial Neural Network,” *J. Inst. Eng. Ser. B*, vol. 99, no. 5, pp. 467–477, 2018.
- [30] A. Rohani, M. H. Abbaspour-Fard, and S. Abdolahpour, “Prediction of tractor repair and maintenance costs using Artificial Neural Network,” *Expert Syst. Appl.*, vol. 38, no. 7, pp. 8999–9007, 2011.
- [31] E. Lee, Y. D. Seo, and Y. G. Kim, “Self-adaptive framework based on MAPE loop for internet of things,” *Sensors (Switzerland)*, vol. 19, no. 13, pp. 1–24, 2019.
- [32] Z. CÔMERT and A. KOCAMAZ, “A Study of Artificial Neural Network Training Algorithms for Classification of Cardiotocography Signals,” *Bitlis Eren Univ. J. Sci. Technol.*, vol. 7, no. 2, pp. 93–103, 2017.
- [33] G. Ramadhona, B. D. Setiawan, and F. A. Bachtiar, “Prediksi Produktivitas Padi Menggunakan Jaringan Syaraf Tiruan Backpropagation,” *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 2, no. 12, pp. 6048–6057, 2018.