

## **APLIKASI PREDIKSI KETERSEDIAAN PANGAN DI KABUPATEN KETAPANG MENGGUNAKAN METODE JARINGAN SYARAF TIRUAN *RESILIENT-BACKPROPAGATION* BERBASIS *WEB***

<sup>[1]</sup>Muhammad Rijian Junanda, <sup>[2]</sup>Dwi Marisa Midyanti

<sup>[1],[2]</sup>Jurusan Rekayasa Sistem Komputer, Fakultas MIPA Universitas Tanjungpura

Jalan Prof Dr. H. Hadari Nawawi Pontianak

Telp./Fax.: (0561) 577963

e-mail: <sup>[1]</sup>mrjunanda@siskom.untan.ac.id, <sup>[2]</sup>dwi.marisa@siskom.untan.ac.id

### **Abstrak**

*Ketersediaan pangan tergantung pada berbagai faktor seperti iklim, jenis tanah, curah hujan, irigasi, komponen produksi pertanian yang digunakan, dan bahkan insentif bagi para petani untuk menghasilkan tanaman pangan. Terjadi penurunan jumlah produksi tanaman padi, ubi kayu, dan ubi jalar pada tahun 2015 dikarenakan besarnya permintaan konsumsi sedangkan produksi ketersediaan pangan tidak mampu mengimbangi besarnya permintaan tersebut, sehingga dikhawatirkan jumlah ketersediaan pangan untuk jenis tanaman tersebut mengalami penurunan yang signifikan untuk beberapa tahun kedepan. Pertumbuhan jumlah penduduk dan jumlah lahan pertanian yang semakin sempit dapat mengakibatkan terjadinya krisis kekurangan pangan, oleh karena itu dibutuhkan sistem prediksi untuk memperkirakan jumlah ketersediaan pangan untuk satu tahun kedepan. Penelitian ini bertujuan untuk membuat aplikasi prediksi ketersediaan pangan dan mencari kinerja mean square error (MSE) terbaik berdasarkan parameter resilient-backpropagation (RPROP) yang digunakan. MSE digunakan untuk mengukur kinerja keluaran yang dihasilkan oleh parameter resilient-backpropagation (RPROP). Data yang digunakan adalah data luas tanam, data luas panen, data produktivitas, dan data produksi untuk masing-masing jenis sereal. Berdasarkan hasil penelitian diperoleh MSE pelatihan sebesar 0.00371664 untuk padi, 0.00935488 untuk jagung, 0.012203 untuk ubi kayu dan 0.0240659 untuk ubi jalar. MSE pengujian sebesar 0,001663 untuk padi, 0,001064 untuk jagung, 0,00091 untuk ubi kayu dan 0,012694 untuk ubi jalar.*

**Kata Kunci :** *Sereal, Jaringan Syaraf Tiruan, MSE, Resilient, parameter.*

### **1. PENDAHULUAN**

Ketersediaan pangan menurut Peta Ketahanan dan Kerentanan Pangan adalah kondisi tersedianya pangan dari hasil produksi dalam negeri, cadangan pangan, serta pemasukan pangan, termasuk didalamnya bantuan pangan yang tersedia untuk di konsumsi. Ketersediaan pangan merupakan salah satu komponen utama atau subsistem dari ketahanan pangan. Produksi pangan tergantung pada berbagai faktor seperti iklim, jenis tanah, curah hujan, irigasi, komponen produksi pertanian yang digunakan, dan bahkan insentif bagi para petani untuk menghasilkan tanaman pangan.

Menurut data dari BPS (Badan Pusat Statistik) Kabupaten Ketapang, Kabupaten Ketapang merupakan kabupaten terluas di

Provinsi Kalimantan Barat yang terdiri dari 20 Kecamatan yang memiliki produksi pertanian mayoritas yaitu padi, jagung, ubi kayu, dan ubi jalar. Menurut data dari Dinas Ketahanan Pangan dan Perikanan Kabupaten Ketapang, mayoritas jumlah produksi tanaman sereal pada tahun 2015 mengalami penurunan dibandingkan dengan tahun 2014. Penyebab penurunan produksi dikarenakan besarnya permintaan konsumsi, sedangkan produksi ketersediaan pangan tidak mampu mengimbangi besarnya permintaan tersebut. Jumlah penduduk yang terus bertambah dan jumlah lahan yang semakin sempit juga menjadi faktor terjadinya krisis kekurangan pangan. Apabila jumlah produksi pangan menurun dan tidak ada peningkatan produksi pangan dari petani lokal dikhawatirkan

ketergantungan terhadap permintaan pangan secara impor, oleh karena itu diperlukan sistem untuk memprediksi jumlah ketersediaan pangan untuk satu tahun kedepan.

Penelitian yang menggunakan metode RPROP pernah dilakukan oleh Apriliyah untuk memprediksi penjualan beban listrik dengan hasil MSE (Mean Square Error) sebesar 0.00002832 untuk unit Blimbing, 0.000197 untuk unit Dinoyo, 0.00001836 untuk unit Kota dan 0.00000875 untuk unit Kebon Agung [1].

Penelitian lain yang menggunakan RPROP pernah dilakukan oleh Susanti (2014) untuk memprediksi harga ayam yang menghasilkan nilai MSE sebesar 0,0113 [2].

Penelitian yang berkaitan dengan ketersediaan pangan pernah dilakukan oleh Fuad (2011) untuk memprediksi ketersediaan beras menggunakan logika fuzzy dengan jaringan syaraf tiruan yang menghasilkan rata-rata kesalahan (error) 0.3873 [3].

Dari permasalahan yang telah diuraikan pada alinea kedua, maka perlu dibangun sebuah sistem untuk memprediksi jumlah ketersediaan pangan untuk satu tahun kedepan dengan metode jaringan syaraf tiruan, *backpropagation* merupakan salah satu metode jaringan syaraf tiruan yang sering digunakan untuk mengatasi kasus prediksi karena mampu menghasilkan hasil prediksi dengan baik. Kekurangan dari metode ini adalah dibutuhkan pelatihan yang cukup lama dan harus memilih nilai laju pembelajaran terbaik untuk mendapatkan hasil pelatihan yang optimal. Untuk mengatasi hal tersebut dibutuhkan metode *resilient* sehingga pelatihan *backpropagation* memiliki kinerja yang lebih baik. Penggabungan metode *backpropagation* dengan metode optimasi *resilient* disebut dengan *Resilient-Backpropagation*.

## 2. TINJAUAN PUSTAKA

Algoritma jaringan syaraf tiruan *Resilient-Backpropagation* (RPROP) merupakan hasil pengembangan algoritma *backpropagation*. Pada *backpropagation* perubahan bobot dipengaruhi oleh learning rate (laju pembelajaran) dan tergantung dari kesalahan pada bobot-bobot pembelajaran ( $\frac{\partial E}{\partial w_{ij}}$ ). Semakin kecil learning rate (laju

pembelajaran), proses pembelajaran semakin lama. Sedangkan semakin besar learning rate, nilai bobot akan jauh dari bobot minimum. Untuk mengatasi hal tersebut, dikembangkan algoritma yang disebut RPROP. Algoritma ini menggunakan tanda (positif atau negatif) dari gradient untuk menunjukkan arah penyesuaian bobot. Sedangkan ukuran perubahan bobot ditentukan dengan nilai penyesuaian ( $\Delta_0$ ).

Nilai penyesuaian mempunyai batas bawah ( $\Delta_{min}$ ) dan batas atas ( $\Delta_{max}$ ). Parameter menggunakan  $\Delta_{min} = 10^{-6}$  dan  $\Delta_{max} = 50$ . Parameter RPROP yang lain adalah faktor penurun ( $\eta^-$ ) dan faktor penaik ( $\eta^+$ ) atau dapat dikatakan sebagai laju pembelajaran seperti pada *backpropagation*. Nilai laju pembelajaran yang digunakan adalah 1.2 untuk nilai  $\eta^+$  dan 0.5 untuk nilai  $\eta^-$ .

Menentukan jumlah nilai neuron pada lapisan tersembunyi perlu dilakukan agar didapat hasil pelatihan optimal. Jika jumlah neuron terlalu banyak akan mengakibatkan perhitungan memakan waktu lama, jika jumlah neuron terlalu sedikit, hasil keluaran akan menjadi besar [4].

Biasanya jumlah neuron pada lapisan tersembunyi ditentukan dengan kondisi:

1. Jumlah neuron pada lapisan tersembunyi berjumlah  $2/3$  dari jumlah neuron pada lapisan masukan.
2. Jumlah neuron pada lapisan tersembunyi harus kurang dari 2 kali jumlah neuron pada lapisan masukan.
3. Jumlah neuron pada lapisan tersembunyi berkisar antara jumlah neuron pada lapisan masukan dan lapisan keluaran.

Penelitian ini menggunakan fungsi aktivasi sigmoid biner untuk menghitung keluaran jaringan pada *backpropagation*. Fungsi aktivasi sigmoid biner memiliki nilai pada interval 0 sampai 1. Oleh karena itu, fungsi ini sering digunakan untuk jaringan syaraf yang membutuhkan nilai keluaran yang terletak pada interval 0 sampai 1. Fungsi aktivasi sigmoid biner ditunjukkan dengan persamaan 2.1.

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

Algoritma Backpropagation:

1. Fase *Feed Forward*:

- a. Tiap unit masukan  $x_i$  ( $i = 1, 2, 3, \dots, n$ ) menerima sinyal dan meneruskannya ke semua unit lapisan tersembunyi.
- b. Setiap unit di lapisan tersembunyi  $z_j$  ( $j = 1, 2, 3, \dots, p$ ) menjumlahkan sinyal-sinyal *input* dengan bobotnya masing-masing:

$$z_{netj} = v_{j0} + \sum_{i=1}^n x_i v_{ji} \quad (2)$$

Dimana :

$v_{j0}$  = bobot garis yang menghubungkan bias di unit masukan ke unit di lapisan tersembunyi  $z_j$

$v_{ji}$  = bobot garis dari unit masukan ke unit di lapisan tersembunyi:  $z_j$

Sinyal *Output* dihitung dengan fungsi aktivasi pada persamaan 2.1.

- c. Unit di lapisan keluaran  $y_k$  ( $k = 1, 2, 3, \dots, m$ ) menjumlahkan sinyal-sinyal dari unit tersembunyi dengan bobotnya masing-masing:

$$y_{netk} = w_{k0} + \sum_{j=1}^n z_j w_{kj} \quad (3)$$

Dimana:

$w_{k0}$  = bobot garis yang menghubungkan bias di unit lapisan tersembunyi ke unit lapisan keluaran  $y_k$

$w_{kj}$  = bobot garis dari unit lapisan tersembunyi  $z_j$  ke unit di lapisan keluaran  $y_k$ .

Sinyal *output* dihitung dengan fungsi aktivasi pada persamaan 2.1.

2. Fase *Backpropagation*

- d. Unit di lapisan keluaran  $y_k$  ( $k = 1, 2, 3, \dots, m$ ) menerima target ( $t$ ) lalu dibandingkan dengan masukan yang diterimanya. Faktor kesalahan (*error*) dihitung dengan menggunakan persamaan:

$$\delta_k = (t_k - y_k) f'(y_{netk}) = (t_k - y_k) y_k (1 - y_k) \quad (4)$$

Dimana:

$\delta_k$  = Kesalahan yang akan dipakai dalam penyesuaian bobot lapisan di bawahnya.

Hitung penyesuaian bobot  $\Delta w_{kj}$  :

$$\Delta w_{kj} = \delta_k z_j \quad (5)$$

- e. Hitung faktor  $\delta$  unit tersembunyi berdasarkan kesalahan disetiap unit tersembunyi  $z_j$  :

$$\delta_{netj} = \sum_{k=1}^m \delta_k w_{kj} \quad (6)$$

Nilai ini dikalikan dengan turunan fungsi aktivasi untuk mendapatkan informasi kesalahan (*error*).

$$\delta_j = \delta_{netj} f'(z_{netj}) = \delta_{netj} z_j (1 - z_j) \quad (7)$$

Kemudian dihitung penyesuaian bobot  $\Delta v_{ji}$ :

$$\Delta v_{ji} = \delta_j x \quad (8)$$

Aturan nilai penyesuaian:

$$\Delta v_{ji}^{(t)} = \begin{cases} \min(\Delta v_{ji}^{(t-1)} * \eta^+, \Delta_{max}), & \text{if } \frac{\partial E^{(t)}}{\partial w_{ji}} \frac{\partial E^{(t-1)}}{\partial w_{ji}} > 0 \\ \max(\Delta v_{ji}^{(t-1)} * \eta^-, \Delta_{min}), & \text{if } \frac{\partial E^{(t)}}{\partial w_{ji}} \frac{\partial E^{(t-1)}}{\partial w_{ji}} < 0 \\ \Delta v_{ji}^{(t-1)}, & \text{else} \end{cases} \quad (9)$$

Dimana:

$\eta^-$  = *learning rate* penurunan (faktor penurunan)

$\eta^+$  = *learning rate* kenaikan (faktor kenaikan)

$\frac{\partial E^{(t)}}{\partial w_{ij}}$  = nilai kesalahan bobot pelatihan pada iterasi saat ini.

$\frac{\partial E^{(t-1)}}{\partial w_{ij}}$  = nilai kesalahan bobot pelatihan pada iterasi sebelumnya.

Fase penyesuaian bobot:

Penyesuaian bobot pada RPROP berbeda dengan backpropagation. Pada RPROP, bobot pada data berikutnya dapat bertambah atau berkurang tergantung dengan perubahan gradien error.

$$w_{ji}^{(t)} = \begin{cases} -\Delta_{ji}^{(t)}, & \text{if } \frac{\partial E^{(t)}}{\partial w_{ji}} > 0 \\ +\Delta_{ji}^{(t)}, & \text{if } \frac{\partial E^{(t)}}{\partial w_{ji}} < 0 \\ 0, & \text{else} \end{cases} \quad (10)$$

Algoritma RPROP:

- a. Inisialisasi penyesuaian awal:

$$\Delta_{ji}(t-1) = \Delta_0 = 0,01 \quad (11)$$

$$\frac{\partial E}{\partial w_{ij}}(t-1) = 0 \quad (12)$$

b. Lakukan langkah-langkah berikut sampai bobot konvergen:

1. Hitung gradient  $\frac{\partial E}{\partial W}(t)$
2. Untuk semua bobot, hitung nilai penyesuaian dengan persamaan 9.

a. Jika  $\frac{\partial E}{\partial W}(t) * \frac{\partial E}{\partial W_{ji}}(t-1) > 0$  maka:

$$\Delta_{ij}(t) = \min(\Delta_{ij}(t-1) * \eta^+, \Delta_{max}) \quad (13)$$

b. Jika  $\frac{\partial E}{\partial W}(t) * \frac{\partial E}{\partial W_{ij}}(t-1) < 0$  maka:

$$\Delta_{ij}(t) = \max(\Delta_{ij}(t-1) * \eta^-, \Delta_{min}) \quad (14)$$

c. Jika  $\frac{\partial E}{\partial W}(t) * \frac{\partial E}{\partial W_{ij}}(t-1) = 0$  maka:

$$\Delta_{ij}(t) = \Delta_{ij}(t-1) \quad (15)$$

3. Hitung perubahan bobot dengan persamaan 2.10.

a. Jika  $\frac{\partial E}{\partial W}(t) > 0$  maka:

$$w(t+1) = w(t) - \Delta_{ij}(t) \quad (16)$$

b. Jika  $\frac{\partial E}{\partial W}(t) < 0$  maka:

$$w(t+1) = w(t) + \Delta_{ij}(t) \quad (17)$$

c. Selesai

### 3. METODE PENELITIAN

Metode penelitian dimulai dengan tahap studi pustaka yang bertujuan untuk mencari literatur baik dari buku, jurnal, atau melalui situs yang berhubungan dengan penelitian terkait. Pada penelitian Prediksi Ketersediaan Pangan Di Kabupaten Ketapang, referensi yang dibutuhkan antara lain tentang pengertian jaringan syaraf tiruan, pengertian Resilient-Backpropagation (RPROP), pengertian ketersediaan pangan, pengertian Mean Square Error (MSE), pengertian diagram alir (flowchart) dan data flow diagram (DFD) dan pengertian bahasa pemrograman PHP dan SQL. Selanjutnya dilakukan tahap observasi dan Pengumpulan data yang dilakukan dengan mengunjungi kantor Dinas Ketahanan Pangan dan

Perikanan Kabupaten Ketapang untuk melakukan pengumpulan data-data yang terkait dengan penelitian.

Pada penelitian prediksi ketersediaan pangan di Kabupaten Ketapang data yang dibutuhkan adalah:

1. Data luas tanam
2. Data luas panen
3. Data produktivitas panen
4. Data produksi panen

Selanjutnya dilakukan tahap yang analisis dibagi menjadi analisis kebutuhan perangkat lunak dan analisis kebutuhan perangkat keras. Tahap perancangan juga dibagi menjadi dua yaitu perancangan basis data dan perancangan antarmuka. Perancangan basis data dilakukan untuk merancang penyimpanan data-data pada penelitian prediksi ketersediaan pangan. Perancangan antarmuka dilakukan dengan merancang tampilan beserta fitur-fitur yang dibutuhkan pengguna saat mengakses aplikasi.

## 4. PERANCANGAN

### 4.1. Perancangan Sistem

Perancangan model jaringan syaraf tiruan prediksi ketersediaan pangan di Kabupaten Ketapang menggunakan metode RPROP sebagai sistem prediksi. Data-data yang telah dikumpulkan akan dibagi menjadi data pelatihan dan data pengujian, setiap masing-masing data memiliki nilai masukan dan nilai keluaran.

a. Penetapan Masukan

Pada sistem prediksi ketersediaan pangan di Kabupaten Ketapang menggunakan metode RPROP, data yang digunakan sebagai masukan sistem adalah data luas tanam sereal, data luas panen sereal, data produktivitas sereal dan data produksi pangan sereal di Kabupaten Ketapang dari tahun 2010 sampai tahun 2016. Data diambil dari Dinas Ketahanan Pangan dan Perikanan Kabupaten Ketapang.

b. Penetapan Keluaran

Metode jaringan syaraf tiruan dengan pembelajaran terawasi membutuhkan target keluaran sebagai pembanding untuk menentukan keberhasilan pembelajaran. Data keluaran dibutuhkan pada proses pembelajaran agar sistem prediksi yang diharapkan memberikan hasil yang optimal.

Pada penelitian prediksi ketersediaan pangan di Kabupaten Ketapang menggunakan metode RPROP, data keluaran sistem yang digunakan adalah data ketersediaan pangan jenis sereal di Kabupaten Ketapang dari tahun 2011 sampai tahun 2016.

c. Arsitektur Jaringan RPROP

Arsitektur jaringan syaraf tiruan RPROP merupakan metode yang memiliki satu lapisan tersembunyi. Pada proses pembelajaran terawasi jaringan syaraf tiruan, parameter pelatihan mempengaruhi keberhasilan pembelajaran [5].

Parameter yang akan digunakan pada metode RPROP ditunjukkan pada tabel 4.1

Tabel 4.1 Parameter Pelatihan RPROP.

Parameter	Nilai/Fungsi
Jumlah Neuron Lapisan Masukan	4
Jumlah Neuron Lapisan Keluaran	1
Jumlah Neuron Lapisan Tersembunyi	3
Target Minimum Error (MSE)	$10^{-5}$
Fungsi Aktivasi	Sigmoid Biner
Maksimum Iterasi	100000
$\eta^+$	1 - 2
$\eta^-$	0,1 - 1
$\Delta_{max}$	10 - 100
$\Delta_{min}$	$10^{-10} - 1$

Pada pelatihan metode RPROP, jumlah neuron pada lapisan masukan terdiri dari 4 neuron yaitu data luas tanam (X1), data luas panen (X2), data produktivitas (X3), dan data produksi (X4) bahan pokok sereal. Data yang digunakan sebagai lapisan keluaran berjumlah 1 neuron yaitu data ketersediaan pangan sereal di Kabupaten Ketapang untuk 1 tahun kedepan (T). Lapisan tersembunyi berada diantara lapisan masukan dan lapisan keluaran, jumlah neuron yang digunakan pada lapisan tersembunyi berjumlah 3. Target minimum error yaitu nilai Mean Square Error merupakan besar selisih antara target dan keluaran perhitungan, nilai MSE di set sekecil mungkin ( $10^{-5}$ ) agar hasil keluaran mendekati target. Fungsi aktivasi yang digunakan adalah fungsi sigmoid biner dikarenakan data penelitian akan dinormalisasi dalam rentang 0 sampai 1, dan keluaran dari fungsi sigmoid biner berkisar antara 0 sampai 1. Maksimal iterasi yang

digunakan sebesar 100000. Parameter RPROP yang digunakan yaitu  $\eta^+$  bernilai dari 1 sampai 2,  $\eta^-$  bernilai dari 0,1 sampai 1,  $\Delta_{max}$  bernilai dari 10 sampai 100, dan  $\Delta_{min}$  bernilai dari  $10^{-10}$  sampai 1, hal ini bertujuan untuk mencari nilai MSE terkecil dari iterasi-iterasi yang digunakan.

d. Perancangan Prosedural

Perancangan prosedural merupakan gambaran alur tahap pelatihan dan pengujian. Pada tahap pelatihan, data latih beserta bobot awal pelatihan akan dilatih menggunakan metode RPROP untuk mendapatkan bobot akhir pelatihan. Bobot-bobot yang menghasilkan nilai MSE terendah dari hasil pelatihan akan digunakan pada tahap pengujian. Blok diagram pelatihan ditunjukkan pada gambar 4.1.



Gambar 4.1 Blok diagram pelatihan RPROP

Pada tahap pengujian data uji beserta bobot yang menghasilkan nilai MSE terkecil akan diuji menggunakan metode RPROP untuk mendapatkan keluaran pengujian. Gambar 4.2 menunjukkan blok diagram tahap pengujian.



Gambar 4.2 Blok diagram pengujian RPROP

e. Diagram Konteks Sistem

Diagram konteks merupakan gambaran alur secara umum interaksi antara external entity terhadap sistem. Perancangan diagram konteks harus jelas sehingga proses-proses yang melibatkan external entity terhadap sistem dapat dipahami. Diagram konteks sistem ditunjukkan pada gambar 4.3.



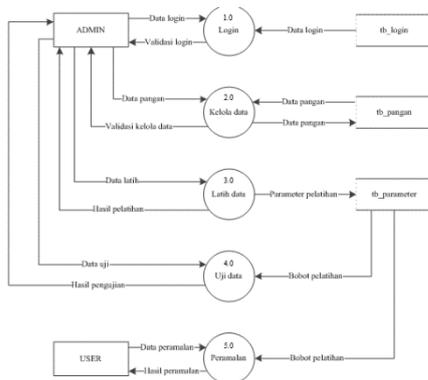
Gambar 4.3 Diagram Konteks

Pada gambar 4.5, lingkungan sistem adalah aplikasi prediksi ketersediaan pangan

di Kabupaten Ketapang dan entitas luar (external entity) adalah admin. Alur panah dari admin dan user menuju sistem menunjukkan proses masukan. Sedangkan alur panah dari sistem menuju admin dan user merupakan keluaran sistem.

1. *Data Flow Diagram* (DFD) level 1

DFD level 1 merupakan detail proses-proses yang melibatkan admin dan user terhadap sistem. Gambar 4.4 menunjukkan DFD level 1



Gambar 4.4 DFD Level 1

1. Pada proses login, admin memasukkan data login yaitu username dan password untuk diproses. Data login yang telah dimasukkan akan dicocokkan dengan data login pada tb\_admin, dan sistem akan memvalidasi proses login tersebut.
2. Pada proses kelola data, admin memasukkan data pangan yang akan dikelola. Data yang dikelola akan dikirim menuju tbl\_pangan dan sistem mengirim validasi data yang telah dikelola menuju admin.
3. Pada proses latihan data, admin melatih data latihan. Parameter pelatihan yang digunakan pada pelatihan disimpan ke dalam tbl\_bobot.
4. Pada proses uji data, admin melakukan pengujian data uji. Bobot-bobot parameter terbaik yang telah disimpan digunakan untuk menguji data.
5. Proses prediksi digunakan oleh user dengan memasukkan data prediksi. Sistem akan melakukan prediksi menggunakan bobot-bobot parameter terbaik dan mengirim hasil prediksi kembali menuju user.

5. IMPLEMENTASI, PENGUJIAN DAN PEMBAHASAN

5.1. Implementasi Perangkat Lunak

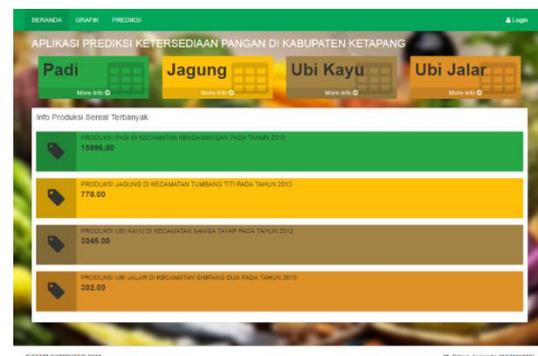
Tahap implementasi dilakukan untuk menerapkan hasil perancangan yang telah dibuat. Pada penelitian prediksi ketersediaan pangan di Kabupaten Ketapang, tahap awal implementasi dimulai dengan menentukan data masukan, data keluaran sistem, dan parameter penelitian. Tahap selanjutnya adalah implementasi tampilan aplikasi dan kode program.

5.1.1. Tampilan Aplikasi

Tampilan aplikasi bertujuan untuk mengkomunikasikan fitur-fitur sistem yang tersedia agar pengguna dapat mengerti dan dapat menggunakan sistem. Berikut tampilan sistem pada aplikasi prediksi ketersediaan pangan.

a. Halaman Awal

Halaman awal merupakan halaman pertama yang diakses saat aplikasi dibuka. Gambar 5.1 menunjukkan tampilan halaman awal aplikasi.



Gambar 5.1 Halaman Awal Aplikasi

b. Halaman Manajemen Data

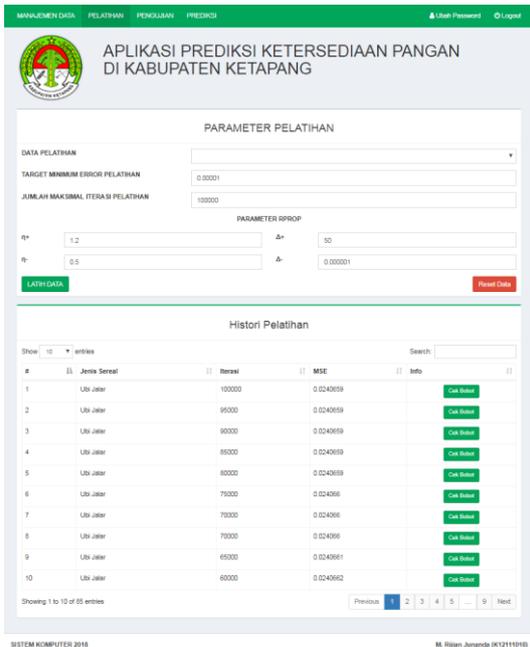
Halaman manajemen data digunakan admin untuk mengelola data-data prediksi ketersediaan pangan yaitu data luas tanam, data luas panen, data produktivitas, data produksi dan data ketersediaan pangan untuk masing-masing kecamatan di Kabupaten Ketapang. Tampilan awal halaman manajemen data ditunjukkan pada gambar 5.2.



Gambar 5.2 Halaman Manajemen Data

c. Halaman Pelatihan

Halaman pelatihan digunakan admin untuk melatih data-data yang telah dikelola di dalam form parameter pelatihan. Gambar 5.3 menunjukkan tampilan halaman pelatihan beserta *history* pelatihan.



Gambar 5.3 Halaman Pelatihan

d. Halaman Prediksi

Halaman prediksi digunakan oleh admin dan pengguna untuk memprediksi jumlah ketersediaan pangan untuk jenis sereal padi, jagung, ubi jalar dan ubi kayu pada masing-masing kecamatan di Kabupaten Ketapang. Halaman awal prediksi ditunjukkan pada gambar 5.4.



Gambar 5.4 Halaman Prediksi

5.1.2. Hitungan Manual

Proses prediksi ketersediaan pangan di Kabupaten Ketapang menggunakan metode RPROP dimulai dengan melakukan pelatihan data. Tujuannya adalah untuk mencari bobot-bobot yang menghasilkan nilai Mean Square Error (MSE) terkecil sehingga dapat digunakan pada proses prediksi. Berikut contoh pelatihan untuk satu baris data pada jenis sereal padi:

Tentukan nilai masukan dan target pelatihan:

$$x_1 = 3740$$

$$x_2 = 3764$$

$$x_3 = 35,62$$

$$x_4 = 13323$$

$$t = 11777$$

Dimana:

$$x_1 = \text{Data luas tanam}$$

$$x_2 = \text{Data luas panen}$$

$$x_3 = \text{Data produktivitas}$$

$$x_4 = \text{Data produksi}$$

$$t = \text{Data ketersediaan pangan}$$

Lakukan normalisasi data masukan dan target. Tentukan nilai maksimal dan minimal dari masing-masing data. Nilai tersebut diambil dari keseluruhan data yang tersedia, nilai diambil dari data terbesar dan terkecil.

$$x_{1max} = 5322$$

$$x_{1min} = 116$$

$$x_{2max} = 4906$$

$$x_{2min} = 26$$

$$x_{3max} = 53,8$$

$$x_{3min} = 19,92$$

$$x_{4max} = 15996$$

$$x_{4min} = 76$$

$$t_{max} = 15996$$

$$t_{min} = 76$$

Sehingga proses normalisasi untuk masukan  $x_1$  adalah sebagai berikut:

$$\begin{aligned} x_1' &= \frac{(x_1 - x_{1min})}{(x_{1max} - x_{1min})} \\ &= \frac{(3740 - 116)}{(5322 - 116)} \\ &= 0,6961 \end{aligned}$$

Lakukan normalisasi terhadap data-data yang lain, sehingga hasil normalisasinya sebagai berikut:

$$x_1 = 0,6961$$

$$x_2 = 0,7659$$

$$x_3 = 0,4634$$

$$x_4 = 0,8321$$

$$t = 0,7349$$

Inisialisasi bobot-bobot dan bias awal pelatihan untuk masing-masing lapisan:

$$w = \begin{bmatrix} 0,8696 & 0,6685 & 0,4387 \\ 0,7257 & 0,5052 & 0,5534 \\ 0,8247 & 0,4739 & 0,2698 \\ 0,2362 & 0,6884 & 0,6251 \end{bmatrix}$$

$$b_1 = [0,5144 \quad 0,6071 \quad 0,2501]$$

$$v = \begin{bmatrix} 0,7451 \\ 0,2639 \\ 0,7277 \end{bmatrix}$$

$$b_2 = 0,2402$$

Dimana :

$w$  = Bobot pada lapisan masukan menuju lapisan tersembunyi

$b_1$  = Bias dari lapisan masukan menuju lapisan tersembunyi

$v$  = Bobot pada lapisan tersembunyi menuju lapisan keluaran

$b_2$  = Bias dari lapisan tersembunyi menuju lapisan keluaran

Tentukan parameter *resilient* pelatihan, yaitu:

$$\alpha^+ = 1,2 \quad \alpha^- = 0,5$$

$$\Delta_{max} = 50 \quad \Delta_{min} = 10^{-6}$$

Tentukan gradien awal kesalahan pada masing-masing bobot dan bias jaringan.

$$\begin{aligned} \frac{\delta E}{\delta w}(t-1) &= \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \\ &= \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \\ &= 0 \end{aligned}$$

Dimana:

$\frac{\delta E}{\delta w}$  = Gradien kesalahan pada bobot lapisan masukan menuju lapisan tersembunyi

$\frac{\delta E}{\delta b_1}$  = Gradien kesalahan pada bias lapisan masukan menuju lapisan tersembunyi

$\frac{\delta E}{\delta v}$  = Gradien kesalahan pada bobot lapisan tersembunyi menuju lapisan keluaran

$\frac{\delta E}{\delta b_2}$  = Gradien kesalahan pada bias lapisan tersembunyi menuju lapisan keluaran.

Selanjutnya menentukan delta perubahan bobot dan bias pada masing-masing jaringan.

$$\Delta w(t-1) = \begin{bmatrix} 0,01 & 0,01 & 0,01 \\ 0,01 & 0,01 & 0,01 \\ 0,01 & 0,01 & 0,01 \\ 0,01 & 0,01 & 0,01 \end{bmatrix}$$

$$\Delta b_1(t-1) = [0,01 \quad 0,01 \quad 0,01]$$

$$\Delta v(t-1) = \begin{bmatrix} 0,01 \\ 0,01 \\ 0,01 \end{bmatrix}$$

$$\Delta b_2(t-1) = 0,01$$

Dimana:

$\Delta w$  = Delta perubahan bobot lapisan masukan menuju lapisan tersembunyi.

$\Delta b_1$  = Delta perubahan bias lapisan masukan menuju lapisan tersembunyi

$\Delta v$  = Delta perubahan bobot lapisan tersembunyi menuju lapisan keluaran

$\Delta b_2$  = Delta perubahan bias lapisan tersembunyi menuju lapisan keluaran.

*Fase Feed Forward:*

Hitung *output* pada lapisan tersembunyi menggunakan persamaan 2.

$$\begin{aligned} z_{net_1} &= b_1 + (x_1 * v_{11}) + (x_2 * v_{12}) \\ &\quad + (x_3 * v_{13}) + (x_4 * v_{14}) \\ &= 0,5144 + (0,6961 * 0,8696) + \\ &\quad (0,7659 * 0,7257) + (0,8321 * \\ &\quad 0,8247) + (0,7349 * 0,2362) \\ &= 2,2544 \end{aligned}$$

$$\begin{aligned} z_{net_2} &= b_2 + (x_1 * v_{21}) + (x_2 * v_{22}) \\ &\quad + (x_3 * v_{23}) + (x_4 * v_{24}) \\ &= 0,6071 + (0,6961 * 0,6685) + \\ &\quad (0,7659 * 0,5052) + (0,8321 * \\ &\quad 0,4739) + (0,7349 * 0,6884) \\ &= 2,2519 \end{aligned}$$

$$\begin{aligned} z_{net_3} &= b_3 + (x_1 * v_{31}) + (x_2 * v_{32}) \\ &\quad + (x_3 * v_{33}) + (x_4 * v_{34}) \\ &= 0,2501 + (0,6961 * 0,4387) + \\ &\quad (0,7659 * 0,5534) + (0,8321 * \\ &\quad 0,2698) + (0,7349 * 0,6251) \\ &= 1,6245 \end{aligned}$$

Aktivasi *output* pada lapisan tersembunyi menggunakan fungsi aktivasi sigmoid biner seperti pada persamaan 1.

$$\begin{aligned} z_1 &= \frac{1}{(1 + e^{-z_{net1}})} \\ &= \frac{1}{(1 + e^{-2,2544})} \\ &= 0,9050 \\ z_2 &= \frac{1}{(1 + e^{-z_{net2}})} \\ &= \frac{1}{(1 + e^{-2,2519})} \\ &= 0,9048 \\ z_3 &= \frac{1}{(1 + e^{-z_{net3}})} \\ &= \frac{1}{(1 + e^{-1,6245})} \\ &= 0,8354 \end{aligned}$$

Hitung nilai *output* pada lapisan keluaran menggunakan persamaan 3.

$$\begin{aligned} y_{net} &= b_2 + (z_1 * w_{11}) + (z_2 * w_{12}) + (z_3 * w_{13}) \\ &= 0,2402 + (0,9050 * 0,7451) + (0,9048 * 0,2639) \\ &\quad + (0,8354 * 0,7277) \\ &= 1,7613 \end{aligned}$$

Aktivasi *output* pada lapisan keluaran menggunakan fungsi aktivasi sigmoid biner.

$$\begin{aligned} y &= \frac{1}{(1 + e^{-y_{net}})} \\ &= \frac{1}{(1 + e^{-1,7613})} \\ &= 0,8534 \end{aligned}$$

*Fase Backward:*

Hitung kesalahan (*error*) pada target dan *output* jaringan.

$$\begin{aligned} error &= -(t - y) \\ &= -(0,7349 - 0,8534) \\ &= 0,1184 \end{aligned}$$

Hitung kesalahan jaringan pada lapisan keluaran menggunakan persamaan 4.

$$\begin{aligned} \delta_k &= error * y * (1 - y) \\ &= (-0,1184) * 0,8534 * (1 - 0,8534) \\ &= 0,0148 \end{aligned}$$

Hitung gradien kesalahan pada bobot lapisan tersembunyi menuju lapisan keluaran menggunakan persamaan 5, sehingga matriks kesalahan pada bobot lapisan tersembunyi menuju lapisan keluaran pada iterasi saat ini menjadi:

$$\frac{\delta E}{\delta v}(t) = \begin{bmatrix} 0,0134 \\ 0,0133 \\ 0,0124 \end{bmatrix}$$

Hitung gradien kesalahan pada bias lapisan tersembunyi menuju lapisan keluaran.

$$\begin{aligned} \frac{\delta E}{\delta b_2}(t) &= \delta_k * 1 \\ &= 0,0148 * 1 \\ &= 0,0148 \end{aligned}$$

Hitung kesalahan pada lapisan tersembunyi menggunakan persamaan 6 dan 7.

$$\begin{aligned} d_1 &= \delta_k * w_{11} * z_1 * (1 - z_1) \\ &= 0,0148 * 0,7451 * 0,9050 * (1 - 0,9050) \\ &= 0,0009 \\ d_2 &= \delta_k * w_{12} * z_2 * (1 - z_2) \\ &= 0,0148 * 0,2639 * 0,9048 * (1 - 0,9048) \\ &= 0,0003 \\ d_3 &= \delta_k * w_{13} * z_3 * (1 - z_3) \\ &= 0,0148 * 0,7277 * 0,8354 * (1 - 0,8354) \\ &= 0,0014 \end{aligned}$$

Hitung gradien kesalahan pada bobot lapisan masukan yang menuju lapisan tersembunyi pertama ( $W_{j1}$ ), kedua ( $W_{j2}$ ) dan ketiga ( $W_{j3}$ ) menggunakan persamaan 8, sehingga matriks kesalahan bobot pada lapisan masukan yang menuju lapisan tersembunyi menjadi.

$$\frac{\delta E}{\delta w}(t) = \begin{bmatrix} 0,0006 & 0,0002 & 0,0010 \\ 0,0007 & 0,0003 & 0,0011 \\ 0,0004 & 0,0001 & 0,0007 \\ 0,0008 & 0,0003 & 0,0012 \end{bmatrix}$$

Hitung gradien kesalahan pada bias lapisan masukan yang menuju lapisan tersembunyi menggunakan persamaan 8, sehingga matriks gradien kesalahan pada bias lapisan masukan yang menuju lapisan tersembunyi menjadi:

$$\frac{\delta E}{\delta b_1}(t) = [0,0009 \quad 0,0003 \quad 0,0014]$$

*Fase Resilient:*

Setelah mencari gradien kesalahan pada bobot dan bias masing-masing jaringan, langkah selanjutnya mencari faktor naik atau turun dari gradien kesalahan yang telah didapat menggunakan *resilient*.

Pada bobot lapisan masukan menuju lapisan tersembunyi. Nilai faktor naik dan turun dicari dengan mengalikan antar elemen pada matriks gradien kesalahan iterasi sebelumnya dengan iterasi saat ini (bukan perkalian matriks standar).

$$\frac{\delta E}{\delta w}(t-1) * \frac{\delta E}{\delta w}(t) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} * \begin{bmatrix} 0,0006 & 0,0002 & 0,0010 \\ 0,0007 & 0,0003 & 0,0011 \\ 0,0004 & 0,0001 & 0,0007 \\ 0,0008 & 0,0003 & 0,0012 \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Berdasarkan hasil perkalian tidak terjadi perubahan gradien (naik atau turun), maka nilai delta perubahan bobot yang digunakan adalah nilai delta pada iterasi sebelumnya ( $\Delta w(t - 1)$ ). Nilai bobot lapisan masukan menuju lapisan tersembunyi terbaru didapat dengan nilai bobot sebelumnya dikurangi dengan delta perubahan.

$$w_{baru} = w_{lama} - \Delta w(t - 1)$$

$$w_{baru} = \begin{bmatrix} 0,8696 & 0,6685 & 0,4387 \\ 0,7257 & 0,5052 & 0,5534 \\ 0,8247 & 0,4739 & 0,2698 \\ 0,2362 & 0,6884 & 0,6251 \\ 0,8596 & 0,6585 & 0,4287 \\ 0,7157 & 0,4952 & 0,5434 \\ 0,8147 & 0,4639 & 0,2598 \\ 0,2262 & 0,6784 & 0,6151 \end{bmatrix} - \begin{bmatrix} 0,01 & 0,01 & 0,01 \\ 0,01 & 0,01 & 0,01 \\ 0,01 & 0,01 & 0,01 \\ 0,01 & 0,01 & 0,01 \end{bmatrix}$$

Selanjutnya mencari bias baru lapisan masukan menuju lapisan tersembunyi dengan mengalikan antar elemen pada matriks gradien kesalahan iterasi sebelumnya dengan iterasi saat ini (bukan perkalian matriks standar).

$$\frac{\delta E}{\delta b_1}(t - 1) * \frac{\delta E}{\delta b_1}(t) = [0 \ 0 \ 0] * [0,0009 \ 0,0003 \ 0,0014]$$

$$= [0 \ 0 \ 0]$$

Berdasarkan hasil perkalian tidak terjadi perubahan gradien (naik atau turun), maka nilai delta perubahan bias yang digunakan adalah nilai delta pada iterasi sebelumnya ( $\Delta b_1(t - 1)$ ). Nilai bias lapisan masukan menuju lapisan tersembunyi terbaru didapat dengan nilai bias sebelumnya dikurangi dengan delta perubahan.

$$b_{1baru} = b_{1lama} - \Delta b_1(t - 1)$$

$$= [0,5144 \ 0,6071 \ 0,2501] - [0,01 \ 0,01 \ 0,01]$$

$$= [0,5044 \ 0,5971 \ 0,2401]$$

Selanjutnya mencari bobot baru lapisan tersembunyi menuju lapisan keluaran dengan mengalikan antar elemen pada matriks gradien kesalahan iterasi sebelumnya dengan iterasi saat ini (bukan perkalian matriks standar).

$$\frac{\delta E}{\delta v}(t - 1) * \frac{\delta E}{\delta v}(t) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} * \begin{bmatrix} 0,0134 \\ 0,0133 \\ 0,0124 \end{bmatrix}$$

$$= \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Berdasarkan hasil perkalian tidak terjadi perubahan gradien (naik atau turun), maka nilai delta perubahan bobot yang digunakan adalah nilai delta pada iterasi sebelumnya ( $\Delta v(t - 1)$ ). Nilai bobot lapisan masukan menuju lapisan tersembunyi terbaru didapat dengan nilai bobot sebelumnya dikurangi dengan delta perubahan.

$$v_{baru} = v_{lama} - \Delta v(t - 1)$$

$$= \begin{bmatrix} 0,7451 \\ 0,2639 \\ 0,7277 \end{bmatrix} - \begin{bmatrix} 0,01 \\ 0,01 \\ 0,01 \end{bmatrix}$$

$$= \begin{bmatrix} 0,7351 \\ 0,2539 \\ 0,7177 \end{bmatrix}$$

Terakhir mencari faktor naik atau turun pada bias lapisan tersembunyi yang menuju lapisan keluaran.

$$\frac{\delta E}{\delta b_2}(t - 1) * \frac{\delta E}{\delta b_2}(t) = 0 * 0,0148$$

$$= 0$$

Berdasarkan hasil perkalian tidak terjadi perubahan gradien (naik atau turun), maka nilai delta perubahan bias yang digunakan adalah nilai delta pada iterasi sebelumnya ( $\Delta b_2(t - 1)$ ). Nilai bias lapisan tersembunyi menuju lapisan keluaran terbaru didapat dengan nilai bias sebelumnya dikurangi dengan delta perubahan.

$$b_{2baru} = b_{2lama} - \Delta b_2(t - 1)$$

$$= 0,2402 - 0,01$$

$$= 0,2302$$

### Hasil Pelatihan

Hasil pelatihan terbaik ditunjukkan pada tabel 5.1

Tabel 5.5 Hasil terbaik dari perubahan Parameter RPROP

Jenis Sereal	Iterasi	$\eta^+$	$\eta^-$	$\Delta_{max}$	$\Delta_{min}$	MSE
Padi	100000	1.6	0.5	50	$10^{-6}$	0.00371891
		<b>1.2</b>	<b>0.8</b>	<b>50</b>	<b><math>10^{-6}</math></b>	<b>0.00371664</b>
		1.2	0.5	10	$10^{-6}$	0.00372371
		1.2	0.5	50	$10^{-5}$	0.00372081
Jagung	100000	1.4	0.5	50	$10^{-6}$	0.00935627
		<b>1.2</b>	<b>0.4</b>	<b>50</b>	<b><math>10^{-6}</math></b>	<b>0.00935488</b>
		1.2	0.5	10	$10^{-6}$	0.00936165
Ubi Kayu	100000	1.2	0.5	50	$10^{-9}$	0.0093604
		1.7	0.5	50	$10^{-6}$	0.0122044
		<b>1.2</b>	<b>0.8</b>	<b>50</b>	<b><math>10^{-6}</math></b>	<b>0.012203</b>
Ubi Jalar	100000	1.2	0.5	10	$10^{-6}$	0.0122069
		1.2	0.5	50	$10^{-10}$	0.0122056
		1.1	0.5	50	$10^{-6}$	0.0240659
Ubi Jalar	100000	1.2	0.4	50	$10^{-6}$	0.0240659
		1.2	0.5	10	$10^{-6}$	0.0240659
		<b>1.2</b>	<b>0.5</b>	<b>50</b>	<b><math>10^{-6}</math></b>	<b>0.0240659</b>

Hasil terbaik untuk sereal Padi hasil terbaik didapatkan dengan menggunakan parameter  $\eta^+$  1.2,  $\eta^-$  0.8,  $\Delta_{max}$  50 dan  $\Delta_{min}$   $10^{-6}$  dengan MSE sebesar 0.00371664.

Untuk sereal Jagung hasil terbaik didapatkan dengan menggunakan parameter  $\eta^+$  1.2,  $\eta^-$  0.4,  $\Delta_{max}$  50 dan  $\Delta_{min}$   $10^{-6}$  dengan MSE sebesar 0.00935488. Untuk sereal Ubi Kayu hasil terbaik didapatkan dengan menggunakan parameter  $\eta^+$  1.2,  $\eta^-$  0.8,  $\Delta_{max}$  50 dan  $\Delta_{min}$   $10^{-6}$  dengan MSE sebesar 0.012203. Untuk sereal Ubi Jalar hasil terbaik didapatkan dengan menggunakan parameter  $\eta^+$  1.1,  $\eta^-$  0.5,  $\Delta_{max}$  50 dan  $\Delta_{min}$   $10^{-6}$  dengan MSE sebesar 0.0240659.

## 6. KESIMPULAN

Berdasarkan hasil pelatihan dan pengujian terhadap sistem prediksi jumlah Ketersediaan Pangan di Kabupaten Ketapang menggunakan *Resilient-Backpropagation* (RPROP), maka kesimpulan yang dapat diperoleh dari penelitian ini adalah sebagai berikut:

1. Kinerja RPROP pada data jenis sereal padi menghasilkan *Mean Square Error* (MSE) pelatihan sebesar 0.00371664 dan MSE pengujian sebesar 0,001663. Kinerja RPROP pada data jenis sereal jagung menghasilkan *Mean Square Error* (MSE) pelatihan sebesar 0.00935488 dan MSE pengujian sebesar 0,001064. Kinerja RPROP pada data jenis sereal ubi kayu menghasilkan *Mean Square Error* (MSE) pelatihan sebesar 0.012203 dan MSE pengujian sebesar 0,00091. Kinerja RPROP pada data jenis sereal ubi jalar menghasilkan *Mean Square Error* (MSE) pelatihan sebesar 0.0240659 dan MSE pengujian sebesar 0,012694.
2. Berdasarkan hasil kinerja masing-masing data jenis sereal menunjukkan bahwa MSE pengujian lebih baik dari hasil MSE pelatihan, sehingga bobot yang menghasilkan MSE terbaik dapat digunakan untuk memprediksi ketersediaan pangan.

## 7. SARAN

Adapun hal yang menjadi saran untuk pengembangan sistem prediksi jumlah ketersediaan pangan di Kabupaten Ketapang agar menjadi lebih baik kedepannya adalah dengan menggunakan data perbulan agar hasil prediksi bisa didapatkan dengan lebih akurat.

## DAFTAR PUSTAKA

- [1]. Apriliyah. (2008). Perkiraan Penjualan Beban Listrik Menggunakan Jaringan Syaraf Tiruan Resilient-Backpropagation (RPROP). Malang: Cursor, Vol. 4, no. 2, pp. 41-47.
- [2]. Susanti, Nanik. (2014). Penerapan Model Neural Network Backpropagation Untuk Prediksi Harga Ayam. Kudus: Prosiding SNATIF Ke-1 Tahun 2014, ISBN 978-602-1180-04-4..
- [3]. Fuad FM, M. (2008). Prediksi Ketersediaan Beras di Masyarakat Dengan Menggunakan Logika Fuzzy dan Jaringan Syaraf Tiruan Dalam Upaya Meningkatkan Ketahanan Pangan. Madura: Agointek, Vol. 5, No. 1, Maret 2011.
- [4]. Karsoliya, Saurabh. (2012). Approximating Number of Hidden layer neurons in Multiple Hidden Layer BPNN Architecture. India: International Journal of Engineering Trends and Technology- Vol. 3 Issue 6-2012.
- [5]. Kusumadewi, S., & Hartati, S. (2009). Neuro-Fuzzy Integrasi Sistem Fuzzy dan Jaringan Syaraf Edisi 2. Yogyakarta: Graha Ilmu