

EFEKTIFITAS JARINGAN SYARAF TIRUAN METODE BACKPROPAGATION DALAM MEMPREDIKSI POTENSI BANJIR (Studi Kasus : Kecamatan Sungai Serut Bengkulu)

Agung Pradana Iskandar
Universitas Dehasen Bengkulu
Jl. Meranti No.32, Sawah Lebar, Kec. Ratu Agung, Kota Bengkulu, Bengkulu 3822
Email : agung30pradana.iskandar@gmail.com

Abstrak – Jaringan syaraf tiruan telah banyak digunakan untuk membantu menyelesaikan berbagai macam permasalahan, salah satu permasalahan tersebut adalah pengambilan keputusan berdasarkan pelatihan yang diberikan. Jaringan syaraf tiruan dapat diterapkan dalam berbagai bidang, salah satunya pada bidang kesehatan. Dalam penelitian ini Aplikasi jaringan syaraf tiruan digunakan untuk memprediksi potensi banjir berdasarkan faktor – faktor yang terjadi di alam. Aplikasi ini menggunakan pilihan arsitektur yang terbaik, yaitu dengan 2 buah masukan, 8 unit tersembunyi dan 1 buah keluaran. Metode yang digunakan adalah metode Backpropagation. Data yang digunakan pada penelitian ini sebanyak 120 data, data pada penelitian ini didapat dari BMKG Pulau Baai Klas 1 Bengkulu Penelitian ini menggunakan maksimum iterasi sebanyak 20.000, learning rate 0,1 dan target error sebesar 0,0001. Hasil pengujian terhadap 120 data didapat hasil keakuratan pengujian pada pola 2 - 8 - 1 dengan error -0.0118. Hal tersebut dapat terjadi karena pada jaringan syaraf tiruan jika terdapat data pelatihan yang hampir sama akan sulit mengenali pola.
Kata Kunci : *Jaringan Syaraf Tiruan, Backpropagation, Banjir*

Abstract - Artificial neural networks have been widely used to help solve various kinds of problems, one of these problems is decision making based on the training given. Artificial neural networks can be applied in various fields, one of which is in the health sector. In this study, the artificial neural network application is used to predict the potential for flooding based on the factors that occur in nature. This application uses the best choice of interpreters, namely with 2 inputs, 8 hidden units and 1 output. The method used is the Backpropagation method. The data used in this study were 120 data, the data in this study were obtained from the BMKG Pulau Baai Klas 1 Bengkulu. This study used a maximum iteration of 20,000, a learning rate of 0.1 and a target error of 0.0001. The test results on 120 data obtained the accuracy of the test results in the pattern 2 - 8 - 1 with an error of -0.0118. This can occur because in the artificial neural network, if there is almost the same training data, it will be difficult to recognize patterns.

Keywords: *Artificial Neural Networks, Backpropagation, Flood*

I. PENDAHULUAN

Jaringan Syaraf Tiruan dapat melatih jaringan untuk mendapatkan keseimbangan antara kemampuan jaringan untuk mengenali pola (data Historis) yang digunakan selama pelatihan serta kemampuan jaringan untuk merespon yang benar terhadap pola masukan yang serupa (tapi tidak sama) dengan pola yang dipakai selama pelatihan. Jaringan Syaraf Tiruan adalah algoritma yang paling sesuai dengan permasalahan yang ada karena algoritma Jaringan Syaraf Tiruan dapat mengenali pola yang ada untuk kemudian mengklasifikasi curah hujan dengan akurasi yang baik¹.

Penelitian tentang algoritma *backpropagation* sebelumnya digunakan diberbagai bidang yaitu, kedokteran, peramalan saham, peramalan cuaca. Metode *Backpropagation* merupakan salah satu metode yang sangat baik dalam menangani masalah pengenalan pola-pola kompleks. Di dalam jaringan *Backpropagation*, setiap unit yang berada di lapisan *input* berhubungan

dengan setiap unit yang ada di lapisan tersembunyi. Dalam penelitian ini, sistem yang digunakan adalah sistem Jaringan Syaraf Tiruan dengan pelatihan Perambatan-balik(*Backpropagation*).

Banjir merupakan fenomena alam yang sumbernya dari curah hujan dengan intensitas tinggi dan durasi lama pada daerah aliran sungai (DAS). Metode *Backpropagation* merupakan salah satu metode pembelajaran dari Jaringan Syaraf Tiruan. Jaringan Syaraf Tiruan mampu mengenali kegiatan dengan berbasis masa lalu. Data masa lalu akan dipelajari oleh Jaringan Syaraf Tiruan sehingga mempunyai kemampuan untuk memberi keputusan terhadap data yang belum pernah dipelajari dalam hal ini obyek penelitiannya adalah banjir.

II. LANDASAN TEORI

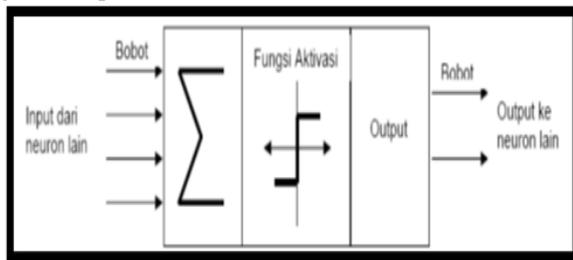
A Jaringan Syaraf Tiruan

Jaringan Syaraf Tiruan adalah sistem pemroses informasi yang memiliki karakteristik mirip dengan Jaringan Syaraf Tiruan².

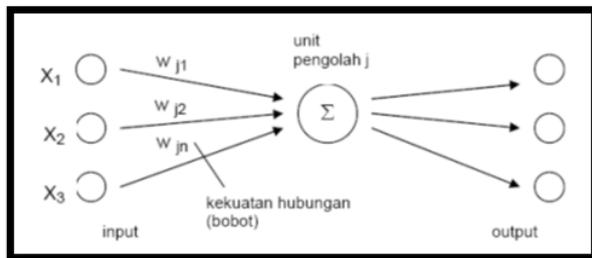
Jaringan Syaraf Tiruan (JST) merupakan suatu sistem pemrosesan informasi yang mempunyai karakteristik menyerupai jaringan syaraf biologi (JSB). JST tercipta sebagai suatu generalisasi model matematis dari pemahaman manusia (*human cognition*) yang didasarkan atas asumsi sebagai berikut :

1. Pemrosesan informasi terjadi pada elemen sederhana yang disebut neuron;
2. Sinyal mengalir diantara sel saraf / *neuron* melalui suatu sambungan penghubung ;
3. Setiap sambungan penghubung memiliki bobot yang bersesuaian. Bobot ini akan digunakan untuk menggandakan / mengalikan sinyal yang dikirim melaluinya ;
4. Setiap sel syaraf akan menerapkan fungsi aktivasi terhadap sinyal hasil penjumlahan berbobot yang masuk kepadanya untuk menentukan sinyal keluarannya.

Model struktur *neuron* jaringan syaraf tiruan dijelaskan pada



Gambar 1 Model Struktur Jaringan Syaraf Tiruan



Gambar 2 Model Struktur Jaringan Syaraf Tiruan

Jaringan syaraf tiruan dapat belajar dari pengalaman, melakukan generalisasi atas contoh-contoh yang diperolehnya dan mengabstraksi karakteristik esensial masukan bahkan untuk data yang tidak relevan. Algoritma untuk JST beroperasi secara langsung dengan angka sehingga data yang tidak numerik harus diubah menjadi data numerik. JST tidak diprogram untuk menghasilkan keluaran tertentu. Semua keluaran atau kesimpulan yang ditarik oleh jaringan didasarkan pada pengalamannya selama mengikuti proses pembelajaran. Pada proses pembelajaran, ke dalam JST dimasukkan

pola-pola masukan (dan keluaran) lalu jaringan akan diajari untuk memberikan jawaban yang bisa diterima. Pada dasarnya karakteristik JST ditentukan oleh ³.

- A. Pola hubungan antar *neuron* (disebut arsitektur jaringan);
- B. Metode penentuan bobot-bobot sambungan (disebut dengan pelatihan atau proses belajar jaringan);
- C. Fungsi aktivasi.

B. Backpropagation

Jaringan Syaraf Tiruan propagasi balik adalah Jaringan Syaraf Tiruan dengan topologi multi-lapis (*multilayer*) dengan satu lapis masukan (lapis X), satu atau lebih lapis *hidden* atau tersembunyi (lapis Z) dan satu lapis keluaran (lapis Y). Setiap lapis memiliki *neuron-neuron* (unit-unit) yang dimodelkan dengan lingkaran. Di antara *neuron* pada satu lapis dengan *neuron* pada lapis berikutnya dihubungkan dengan model koneksi yang memiliki bobot-bobot (*weights*), *w* dan *v*. Lapis tersembunyi dapat memiliki *bias*, yang memiliki bobot sama dengan satu⁴.

Backpropagation merupakan algoritma pembelajaran yang terawasi dan biasanya digunakan oleh perceptron dengan banyak lapisan untuk mengubah bobot – bobot yang terhubung dengan *neuron – neuron* yang ada pada lapisan tersembunyi.

Algoritma *Backpropagation* terdiri dari 3 tahapan yaitu⁵:

1. Tahap umpan maju (*Feedforward*)
2. Tahap umpan mundur (*Backward*)
3. Tahap update bobot dan bias.

Algoritma pelatihan untuk jaringan dengan satu *layer* tersembunyi menggunakan fungsi aktivasi *sigmoid biner* adalah sebagai berikut:

- Langkah 0 : Inialisasi semua bobot dengan bilangan acak kecil.
- Langkah 1 : Jika kondisi penghentian belum terpenuhi, lakukan langkah 2- 9
- Langkah 2 : Untuk setiap pasang data pelatihan lakukan langkah 3-8.

Fase I : Propagasi maju

- Langkah 3 : Tiap unit masukan menerima sinyal dan meneruskannya ke unit tersembunyi di atasnya.
- Langkah 4 : Hitung semua keluaran di unit tersembunyi z_j ($j = 1, 2, \dots, p$).

$$z_net_j = v_{j0} + \sum_{i=1}^n x_i v_{ji} \dots\dots\dots(7)$$

$$z_j = f(z_net_j) = \frac{1}{1 + e^{-z_net_j}} \dots\dots\dots(8)$$

- Langkah 5 : Hitung semua keluaran jaringan di unit y_k ($k = 1, 2, \dots, m$)

$$y_net_k = w_{k0} + \sum_{j=1}^p z_j w_{kj} \dots\dots\dots(9)$$

$$z_k = f(z_{net_k}) = \frac{1}{1 + e^{-y_{net_k}}} \dots\dots\dots (10)$$

Fase II : Propagasi mundur
 Langkah 6 : Hitung faktor δ unit keluaran berdasarkan kesalahan di setiap unit keluaran y_k ($k = 1, 2, \dots, m$)

$$\delta_k = (t_k - y_k) f'(y_{net_k}) = (t_k - y_k) y_k (1 - y_k) \dots\dots\dots (11)$$

δ_k merupakan unit kesalahan yang akan dipakai dalam perubahan bobot *layer* di bawahnya (langkah 7). Hitung suku perubahan bobot w_{kj} dengan laju percepatan α .

$$\Delta w_{kj} = \alpha \delta_k z_j \quad ; k = 1, 2, \dots, m \quad ; j = 0, 1, \dots, p$$

Langkah 7 : Hitung faktor δ unit tersembunyi berdasarkan kesalahan di setiap unit tersembunyi z_j ($j = 1, 2, \dots, p$)

$$\delta_{net_j} = \sum_{k=1}^m \delta_k w_{kj} \dots\dots\dots (12)$$

Faktor δ unit tersembunyi:

$$\delta_j = \delta_{net_j} f'(z_{net_j}) = \delta_{net_j} z_j (1 - z_j) \dots\dots\dots (13)$$

Hitung suku perubahan bobot v_{ji} (yang akan dipakai untuk merubah bobot v_{ji})

$$\Delta v_{ji} = \alpha \delta_j x_i \quad ; j = 1, 2, \dots, p \quad ; i = 0, 1, \dots, n$$

Fase III : Perubahan bobot

Langkah 8 : Hitung semua perubahan bobot n Perubahan bobot garis yang menuju ke unit keluaran:

$$w_{kj}(\text{baru}) = w_{kj}(\text{lama}) + \Delta w_{kj} \dots\dots\dots (14)$$

$$(k = 1, 2, \dots, m ; j = 0, 1, \dots, p)$$

Perubahan bobot garis yang menuju ke unit tersembunyi:

$$v_{ji}(\text{baru}) = v_{ji}(\text{lama}) + \Delta v_{ji} \dots\dots\dots (15)$$

$$(j = 1, 2, \dots, p ; i = 0, 1, \dots, n)$$

Langkah 9 : Menguji apakah kondisi berhenti sudah terpenuhi.

Kondisi berhenti ini terpenuhi jika nilai kesalahan yang dihasilkan lebih kecil dari nilai kesalahan referensi.

C. Banjir

Banjir adalah suatu peristiwa terjadinya genangan pada suatu daerah yang biasanya kering dan bukan daerah rawa, atau terjadinya limpasan air dari alur sungai yang disebabkan debit sungai melebihi kapasitas pengairannya². Terdapat dua pengertian mengenai banjir pertama Aliran air sungai yang tingginya melebihi muka air normal sehingga melimpas dari palung sungai menyebabkan adanya genangan pada lahan rendah disisi sungai. Kedua Gelombang banjir berjalan kearah hilir sistem sungai yang berinteraksi dengan kenaikan muka air di muara akibat badai.

III. ANALISA DAN PERANCANGAN

A. Analisa Permasalahan

Didalam analisa dan perancangan prediksi potensi banjir. Terdapat beberapa data sampel yang digunakan dalam memprediksi potensi banjir, sampel tersebut adalah sampel curah hujan, sampel *rain day*. Data yang digunakan pada penelitian ini adalah data dari tahun 2003 sampai dengan 2012. Data tersebut disajikan dalam bentuk tabel di mana data tersebut dalam satuan *milimeter* yang dinormalisasikan.

Pengumpulan data dilakukan dengan mengambil data historis curah hujan, *Rain day*, data tinggi muka air pada sungai Serut Tanjung Agung dari bulan Januari sampai dengan Desember tahun 2003 sampai dengan 2012, serta melakukan pengamatan terhadap faktor – faktor yang mempengaruhi banjir.

B. Faktor – Faktor Yang Mempengaruhi Dalam Banjir

Bencana banjir dipengaruhi oleh beberapa faktor yaitu, data curah hujan dan *rain day* pada pintu air sungai Serut Bengkulu. Dengan adanya data data *historis* tersebut dapat memberikan gambaran atau pola sehingga dapat digunakan untuk memprediksi potensi banjir. (BMKG Bengkulu Stasiun Klas 1 Pulau Baii).

C. Perancangan Jaringan Syaraf Tiruan untuk Memprediksi Potensi Banjir

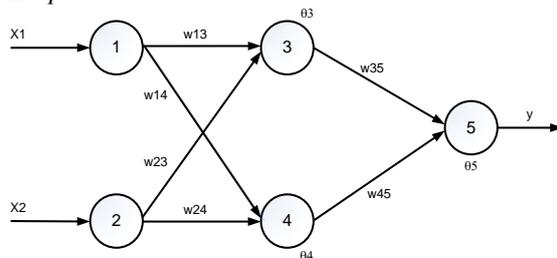
1. Variabel Masukan
 Agar data dapat dikenali oleh Jaringan Syaraf Tiruan dan sekaligus dapat diproses menggunakan perangkat lunak (*software*) maka data harus diubah ke dalam bentuk numerik. Di dalam penelitian ini variabel masukan dalam memprediksi potensi banjir terdiri dari : curah hujan dan *rain day*.
2. Data *Input* dan Target
 Terdapat beberapa hal yang harus didefinisi dan dalam menerapkan Jaringan Syaraf tiruan pada prediksi potensi banjir, yaitu : Nilai input akan dimasukkan pada sistem dalam bentuk angka - angka yang telah diinisialisasikan sebelum dimasukkan kedalam sistem. Sistem akan melakukan pengolahan apakah *output* akan sama dengan data yang diharapkan atau tidak. Sebagai *input* dalam memprediksi potensi banjir adalah data curah hujan, dan *rain day*.
- c. Arsitektur Jaringan
 Umumnya, *neuron – neuron* yang terletak pada lapisan yang sama akan memiliki keadaan yang sama. Faktor terpenting dalam menentukan kelakuan suatu *neuron* adalah fungsi *aktivasi* dan pola bobotnya.

Arsitektur jaringan yaitu sebuah arsitektur yang menentukan pola antar neuron. *Neuron – neuron* tersebut terkumpul dalam lapisan yang disebut *neuron layer*, lapisan penyusun *Jaringan Syaraf Tiruan* dibagi menjadi tiga. Dalam kasus ini arsitektur *Jaringan Syaraf*

Tiruan yang digunakan adalah *Jaringan Backpropagation*, yang terdiri dari 3 lapisan yaitu :

1. Lapisan Input (*Input Layer*), unit – unit dalam lapisan input disebut unit – unit input yang bertugas menerima pola inputan dari luar yang menggambarkan suatu permasalahan dengan 2 simpul.
2. Lapisan tersembunyi (*Hidden Layer*), unit – unit dalam lapisan tersembunyi disebut unit – unit tersembunyi yang mana nilai nilai *output* nya tidak dapat diamati secara langsung. Jumlah simpul yang ditentukan oleh pengguna.
3. Lapisan *Output* (*Output Layer*), unit – unit dalam lapisan *Output* disebut unit – unit *output* , yang merupakan solusi Jaringan Syaraf Tiruan terhadap suatu permasalahan. Terdiri dari 1 simpul yaitu Tinggi Muka Air (TMA), sebagai nilai yang diprediksi.

Pada gambar dibawah ini disajikan arsitektur Jaringan Syaraf Tiruan untuk prediksi potensi banjir dengan 2 buah *input*.



Gambar 3

Arsitektur Jaringan Syaraf Tiruan Untuk Prediksi Potensi Banjir

Gambar diatas memperlihatkan model Jaringan Syaraf Tiruan terdiri atas 2 *input*, 2 *hidden layer* dan 1 *output* (2-2-1) di mana :

$X1$ = Curah Hujan

$X2$ = *Rain Day*

w = Nilai pemberat/ Bobot

θ = *Thresold*

y = Keluaran/hasil.

Untuk membentuk Jaringan Syaraf Tiruan, terlebih dahulu dilakukan inisialisasi bobot awal. Bobot awal yang menghubungkan simpul-simpul pada lapisan input dan lapisan tersembunyi untuk arsitektur di atas adalah $w = (w_{11}, w_{12}, w_{21}, w_{22},)$ dan bobot *bias* dipilih secara acak. Demikian pula bobot awal yang menghubungkan simpul-simpul pada lapisan tersembunyi dan lapisan *output* (w_1, w_2) juga dipilih secara acak.

D. Perancangan Algoritma *Backpogagation*

Adapun langkah – langkah penggunaan algoritma *Backpropagation* menggunakan fungsi aktivasi *sigmoid biner* adalah sebagai berikut :

1 Tahap *Initialization*

Merupakan tahapan untuk mendefinisikan nilai awal untuk variabel – variabel yang diperlukan seperti ; nilai input, nilai output yang

diharapkan, *learning rate* (α), θ dan sebagainya.

2 Tahap *Activation*

Pada tahap *Activation* ini dilakukan 2 kegiatan yaitu ; menghitung *actual output* pada *hidden layer* dan menghitung *actual output* pada *output layer*.

3 Tahap *Weight Training*

Pada tahap ini dilakukan 2 kegiatan yaitu ; menghitung *error gradient* pada *output layer* dan menghitung *error gradient* pada *hidden layer*.

4 Tahap *Iteration*

Tahapan terakhir ini adalah tahapan untuk pengujian di mana jika *error* yang diharapkan belum ditemukan maka akan kembali lagi pada tahap 2.

Perhitungan algoritma *backpropagation* dengan Jaringan Syaraf Tiruan pada prediksi potensi banjir menggunakan 2 variabel *input vector input* x, yaitu ;

$X1$: Data Curah Hujan

$X2$: *Data Rain Day* **Matlab**

E. Perancangan Parameter

Untuk melakukan pelatihan terhadap Jaringan Syaraf Tiruan dengan menggunakan *Matlab* ditentukan parameter berikut ;

1. Parameter *Epoch* adalah 20000
2. Parameter *Show Epoch* adalah 100
3. Parameter *goal* adalah $1e-2$
4. Parameter *ir* adalah 0.5
5. Parameter *mc* adalah 0.8
6. Fungsi Aktivasi dengan menggunakan *Sigmoid Biner* range 0 dan 1.

F. Rancangan Pelatihan dan Pengujian

Pelatihan *Backpropagation* meliputi tiga fase yaitu fase maju di mana masukan dihitung maju mulai dari *layer* masukan hingga *layer* keluaran dengan menggunakan fungsi *aktivasi* yang ditemukan. Fase kedua adalah fase mundur, di mana selisih antara keluaran dengan target. Fase ketiga adalah modifikasi bobot untuk menurunkan kesalahan yang terjadi.

Pada proses perhitungan algoritma *backpropagation* dalam Jaringan Syaraf Tiruan, terlebih dahulu dilakukan bobot awal (diambil random yang cukup kecil). Bobot awal yang menghubungkan simpul – simpul pada lapisan *input* dan lapisan tersembunyi untuk arsitektur di atas adalah $w = (w_{15}, w_{16}, w_{17}, w_{18}, w_{25}, w_{26}, \dots, w_{89})$ dan bobot bias dipilih secara acak. Demikian pula bobot – bobot awal yang menghubungkan simpul – simpul pada lapisan tersembunyi dan lapisan *output* (w_1, w_2) dipilih secara acak.

1. Tahap *Initialitation*

Merupakan tahapan untuk mendefinisikan nilai awal untuk variabel – variabel yang diperlukan seperti ; *input*, *weight*, *output* yang diharapkan, *learning rate* (α), θ dan sebagainya.

Bobot awal merupakan gabungan – gabungan pada lapisan input tersembunyi dan lapisan output untuk menentukan target. Mula – mula bobot diberi nilai acak kecil *range* (0,1) sebagai berikut ;

Tahap 1. Initialitation

Diketahui:

$$x_1 = 0,2408 \quad x_2 = 1$$

$$j = 3 \text{ s/d } 4 \quad k = 5$$

$$w_{13} = 0,5 \quad w_{23} = 0,4 \quad w_{35} = -1,2$$

$$w_{14} = 0,9 \quad w_{24} = 1,0 \quad w_{45} = 1,1$$

$$\theta_3 = 0,8 \quad \theta_4 = -0,1 \quad \theta_5 = 0,3$$

$$\text{Learning Rate } (\alpha) = 0,1 \quad \text{Disered output } (Yd) = 0$$

Tahap 2. Activation (iterasi P= 1)

a. Menghitung aktual output pada hidden layer

$$Y_j(P) = \text{Fungsi} \left[\sum_{i=1}^n x_i(P) \cdot w_{ij}(P) - \theta_j \right]$$

$$\begin{aligned} Y_3(1) &= \text{Sigmoid} [x_1(1) \cdot w_{13}(1) + x_2(1) \cdot w_{23}(1) - \theta_3] \\ &= \text{Sigmoid} [0,2408 \cdot 0,5 + 1,0 \cdot 0,4 - 0,8] \\ &= \text{Sigmoid} [0,1204 + -0,4] \end{aligned}$$

$$= \text{Sigmoid} [-0,2796] \rightarrow \frac{1}{1 + e^{(0,2796)}} = \frac{1}{1,7561} = 0,5694$$

$$\begin{aligned} Y_4(1) &= \text{Sigmoid} [x_1(1) \cdot w_{14}(1) + x_2(1) \cdot w_{24}(1) - \theta_4] \\ &= \text{Sigmoid} [0,2408 \cdot 0,9 + 1,1 - (-0,1)] \\ &= \text{Sigmoid} [0,21672 + 1,1] \end{aligned}$$

$$= \text{Sigmoid} [1,3] \rightarrow \frac{1}{1 + e^{-(1,3)}} = \frac{1}{4,7312} = 0,2114$$

b. Menghitung actual output pada output layer

$$Y_k(P) = \text{Fungsi} \left[\sum_{j=1}^m x_{jb}(p) \cdot x_{wjk}(p) - \theta_k \right]$$

$$\begin{aligned} Y_5(1) &= \text{Sigmoid} [Y_3 \cdot w_{35} + Y_4 \cdot w_{45} - \theta_5] \\ &= \text{Sigmoid} [0,4306 \cdot -1,2 + 0,2466 \cdot 1,1 - 0,3] \\ &= \text{Sigmoid} [-0,7508] \end{aligned}$$

$$= \text{Sigmoid} [-0,7508] \rightarrow \frac{1}{1 + e^{(0,7508)}} = 0,6794$$

Menghitung error:

$$(5) = (Yd * Y5) - Y5$$

$$(0 * 0,6794) - 0,6794$$

$$e(5) = -0,6794$$

Tahap 3: Weight training

a. Menghitung error gradient pada output layer

$$\delta_k(p) = y_k(p) \cdot x [1 - y_k(p)] \cdot x e_k(p)$$

$$\delta_5(1) = y_5(1) \cdot x [1 - y_5(1)] \cdot x e_5(1)$$

$$= 0,6794 [1 - (0,6794)] \cdot -0,6794 = -0,4615$$

Menghitung koreksi nilai weight:

$$W_{jk}(P+1) = W_{jk}(P) + \Delta w_{jk}(P)$$

$$\Delta w_{jk}(P) = \alpha * Y_j(P) * \delta_k(P)$$

$$j = 3 \text{ dan } k = 5$$

$$\begin{aligned} \Delta w_{35}(1) &= \alpha \cdot y_3(1) - \delta_5(1) \\ &= 0,1 * 0,5694 - (-0,4615) \\ &= 0,5185 \end{aligned}$$

$$j = 4 \text{ dan } k = 5$$

$$\begin{aligned} \Delta w_{45}(1) &= \alpha \cdot y_4(1) - \delta_5(1) \\ &= 0,1 * 0,2114 - (-0,4615) \\ &= 0,06027 \end{aligned}$$

$$\begin{aligned} \Delta \theta_5 &= \alpha \cdot (-1) \cdot \delta_5(1) \\ &= 0,1(-1) \cdot (-0,4615) \\ &= 0,0462 \end{aligned}$$

$$\begin{aligned} w_{35}(2) &= w_{35}(1) + \Delta w_{35}(1) \\ &= -1,2 + (0,5185) \\ &= -0,6815 \end{aligned}$$

$$\begin{aligned} w_{45}(2) &= w_{45}(1) + \Delta w_{45}(1) \\ &= 1,1 - 0,4973 \\ &= 0,6027 \end{aligned}$$

$$\theta_5(2) = \theta_5(1) + \Delta \theta_5 = 0,3 + 0,0462 = 0,3462$$

b. Menghitung error gradient pada hidden layer

$$\delta_j(P) = Y_j(p) \cdot [1 - Y_j(p)] \cdot \sum \delta_k(p) \cdot w_{jk}(p)$$

$$j = 3$$

$$\delta_3(1) = Y_3(1) \cdot [1 - Y_3(1)] \cdot \sum \delta_5(P) \cdot w_{35}(P)$$

$$\begin{aligned} \delta_3(1) &= 0,5694 \cdot [1 - 0,5694] \cdot -0,4615 \cdot (-1,2) \\ &= -0,0136 \end{aligned}$$

$$j = 4$$

$$\delta_4(1) = Y_4(1) \cdot [1 - Y_4(1)] \cdot \sum \delta_5(P) \cdot w_{45}(P)$$

$$\begin{aligned} \delta_4(1) &= 0,2114 \cdot [1 - 0,2114] \cdot -0,4615 \cdot 1,1 \\ &= 0,0085 \end{aligned}$$

Koreksi nilai bobot:

$$\Delta w_{13} = \alpha \cdot x_1(1) \cdot \delta_3(1) = 0,1 \cdot 0,2408 \cdot -0,0136 = -0,0003$$

$$\Delta w_{14} = \alpha \cdot x_1(1) \cdot \delta_4(1) = 0,1 \cdot 0,2408 \cdot -0,0085 = -0,0002$$

$$\Delta w_{23} = \alpha \cdot x_2(1) \cdot \delta_3(1) = 0,1 \cdot 1 \cdot -0,0136 = -0,0014$$

$$\Delta w_{24} = \alpha \cdot x_2(1) \cdot \delta_4(1) = 0,1 \cdot 1 \cdot -0,0085 = -0,0008$$

$$\Delta \theta_3 = \alpha \cdot (-1) \cdot \delta_3(1) = 0,1 \cdot (-1) \cdot -0,0136 = 0,0014$$

$$\Delta \theta_4 = \alpha \cdot (-1) \cdot \delta_4(1) = 0,1 \cdot (-1) \cdot -0,0085 = 0,0008$$

$$\Delta w_{13}(2) = w_{13}(1) + \Delta w_{13} = 0,5 + (-0,0003) = 0,4997$$

$$\Delta w_{14}(2) = w_{14}(1) + \Delta w_{14} = 0,9 + 0,0002 = 0,9002$$

$$\Delta w_{23}(2) = w_{23}(1) + \Delta w_{23} = 0,4 + (-0,0014) = 0,3986$$

$$\Delta w_{24}(2) = w_{24}(1) + \Delta w_{24} = 1 + (0,0008) = 1,0008$$

$$\theta_3(2) = \theta_3(1) + \Delta \theta_3 = 0,8 + (-0,0014) = 0,8014$$

$$\theta_4(2) = \theta_4(1) + \Delta \theta_4 = -0,1 + (-0,0008) = -0,1008$$

Untuk melakukan pengolahan data yang banyak sulit dilakukan secara manual karena proses *iterasi* yang panjang. Hasil pengolahan dan perancangan menggunakan *Jaringan Syaraf Tiruan* dengan metode *Backpropagation*, dapat dilihat hubungannya dengan jelas.

Dalam metode *Bacpropagation* pola terstruktur yang dipakai sangat mempengaruhi dalam penentuan hasil. Setiap hasil yang didapat dari perhitungan pola arsitekturnya, hasilnya memungkinkan mendekati nilai yang akan dicapai.

Sesuai dengan variable input yang terdiri dari curah hujan dan *rain day*, maka akan menghasilkan Tinggi Muka Air (TMA). Berdasarkan data sampel yang digunakan *output* yang dihasilkan akan dibandingkan dengan target yang harus dicapai, jadi selisih antara *output* dan target adalah *error*. Jika nilai *error* semakin kecil maka *iterasi* dihentikan dan dianggap *output* sudah sesuai dengan target yang ingin kita capai.

IV HASIL PENGUJIAN

A. Hasil Pengujian

Berdasarkan 5 kali percobaan yang dilakukan dengan pola arsitektur berbeda beda. Menggunakan fungsi pelatihan *TrainGDm*.

Tabel 1 Hasil Pengujian

Hasil Pengujian				
Arsitektur	Training Function	Perf	Error Terkecil	Akurasi
2 - 30 - 1	<i>TrainGDm</i>	0,009998	0.99982	99,93%
2 - 12 - 1	<i>TrainGDm</i>	0,0099988	0.0029	99.97%
2 - 20 - 1	<i>TrainGDm</i>	0,0099993	-0.106	99.98%
2 - 8 - 1	<i>TrainGDm</i>	0,0099991	0.006	99,75%
2 - 10 - 1	<i>TrainGDm</i>	0,0099991	0.0010	99.,97%

Berdasarkan data tabel di atas, bahwa tingkat pelatihan terbaik dimiliki oleh data dengan pola pengujian 2 - 20 - 1 di mana tingkat akurasi data mencapai 99,98 %. Data tingkat akurasi terendah dimiliki oleh arsitektur dengan pola 2 - 8 - 1, di mana tingkat akurasi data dengan pola tersebut sebesar 99,75 %. Berdasarkan pola arsitektur tersebut nilai *error* terkecil tidak bisa menentukan tingkat akurasi yang optimal.

V KESIMPULAN DAN SARAN

A. Kesimpulan

Berdasarkan pembahasan dan hasil pengujian dengan menggunakan *algoritma Backpropogation*, maka dapat ditarik kesimpulan sebagai berikut :

1. Algoritma *Backpropagation* dapat melakukan proses prediksi, akan tetapi baik atau tidaknya nilai yang dihasilkan sangat dipengaruhi oleh

penentuan parameter seperti besarnya *learning rate* dan jumlah *neuron* pada *hidden layer*.

2. Semakin banyak jumlah unit dalam *hidden layer* maka hasil prediksi potensi banjir semakin mendekati target.
3. Terdapat faktor – faktor yang mempengaruhi tingkat kebenaran prediksi pada Jaringan Syarat Tiruan *Backpropagation* yaitu *learning rate*, *target error*, jumlah data pembelajaran dan nilai bobot yang diberikan secara *random* yang pada tiap – tiap *neuron*.
4. Dengan *learning rate* dan *target error* untuk data pembelajaran yang sama bekum pasti menghasilkan tingkat prediksi sama. Hal ini dikarenakan nilai bobot pada setiap *neuron* yang dihasilkan oleh setiap pembelajaran berbeda. Hal ini disebabkan karena pemberian nilai bobot awal dengan nilai *random* di mana dilai *random* setiap pembelajaran berbeda.

VI KESIMPULAN DAN SARAN

A. Kesimpulan

Berdasarkan pembahasan dan hasil pengujian dengan menggunakan *Software Matlab*, maka dapat ditarik kesimpulan sebagai berikut :

1. Algoritma *Backpropagation* dapat melakukan proses prediksi, akan tetapi baik atau tidaknya nilai yang dihasilkan sangat dipengaruhi oleh penentuan parameter seperti besarnya *learning rate* dan jumlah *neuron* pada *hidden layer*.
2. Semakin banyak jumlah unit dalam *hidden layer* maka hasil prediksi potensi banjir semakin mendekati target.
3. Terdapat faktor – faktor yang mempengaruhi tingkat kebenaran prediksi pada Jaringan Syarat Tiruan *Backpropagation* yaitu *learning rate*, *target error*, jumlah data pembelajaran dan nilai bobot yang diberikan secara *random* yang pada tiap – tiap *neuron*.
4. Untuk melakukan pengolahan data yang banyak sulit dilakukan secara manual karena proses *iterasi* yang panjang. Hasil pengolahan dan perancangan menggunakan *Jaringan Syaraf Tiruan* dengan metode *Backpropagation*, dapat dilihat hubungannya dengan jelas.
5. Dengan *learning rate* dan *target error* untuk data pembelajaran yang sama bekum pasti menghasilkan tingkat prediksi sama. Hal ini dikarenakan nilai bobot pada setiap *neuron* yang dihasilkan oleh setiap pembelajaran berbeda. Hal ini disebabkan karena pemberian nilai bobot awal dengan nilai *random* di mana nilai *random* setiap pembelajaran berbeda

B. Saran

Berdasarkan kesimpulan, maka disarankan agar diadakan penelitian lebih lanjut terhadap penentuan prediksi potensi banjir dengan metode yang lain misalnya menggunakan *Logika fuzzy*, *Perceptron*, dan

lain sebagainya. Mengembangkan penelitian lebih mendalam dan variasi algoritma pelatihan, supaya hasil yang dihasilkan lebih optimal dan singkat. Pengesetan parameter dilakukan agar mendapatkan hasil yang lebih akurat perlu menggunakan *Software* seperti *Matlab*,

DAFTAR PUSTAKA

- [1] HANESWARA, G. & MOERTINI, V. S. 2004. Jaringan Syaraf Tiruan Propagasi Balik untuk Klasifikasi Data. INTEGRAL, 9, 117-131.
- [2] SUHARI, Y. 2010. Jaringan Syaraf Tiruan: Aplikasi Pemilihan Merek. Dinamik-Jurnal Teknologi Informasi, 15.
- [3] Wuryandari, M. D., & Afrianto, I. (2012). Perbandingan Metode Jaringan Syaraf Tiruan Backpropagation Dan Learning Vector Quantization Pada Pengenalan Wajah. Jurnal komputer dan informatika (Komputa), 1(1), 1.
- [4] Siang,Jong Jek. 2009. Jaringan Syaraf Tiruan dan Pemrogramannya menggunakan MATLAB. Andi
- [5] Maria, A. (2013). Penggunaan jaringan syaraf tiruan backpropagation untuk seleksi penerimaan mahasiswa baru pada jurusan teknik komputer di Politeknik Negeri Sriwijaya (Doctoral dissertation, Diponegoro University).