

Analisis Keakuratan Curah Hujan dengan Menggunakan Metode Jaringan Syaraf Tiruan

Darsono Nababan¹, Sony Winarto²

¹Fakultas Ilmu Komputer, Universitas Pelita
Harapan E-mail: darsono.nababan@uph.edu¹⁾

²Fakultas Ilmu Komputer, Universitas Pelita
Harapan E-mail: sony37105@gmail.com²⁾

Abstract – Rainfall that falls in each country is different, there are countries that have high rainfall and there are countries that have low rainfall. We have to estimate this climatology so that our country is ready to face the flood season, one of the ways is by using the backpropagation method of artificial neural networks. In order for predictive results to have a low error rate, an optimal backpropagation network architecture is needed, how to optimize the use of training data and test data taken from sample data and formed into tables of data in excel and calculated using excel, then the data is used in MATLAB in the form of input, hidden, and output, with the use of neurons in the input layer totaling 12, the hidden layer is 10, and the output layer is 1. Based on the results of the training process on training data using MATLAB, the results will be Regression = 0.98559 and MSE = 0.00099844. Calculation of training data will also get the output weight (best weight) used in the test data which will produce MSE in the form of = 0.10574.

Keywords: Rainfall, forecasting, neural network, backpropagation, MSE

Abstrak – Curah hujan yang jatuh di setiap negara itu berbeda beda, ada negara yang memiliki curah hujan tinggi dan ada negara yang memiliki curah hujan rendah. Perkiraan klimatologi ini harus bisa kita laksanakan agar negara kita siap untuk menghadapi musim banjir, salah satunya cara adalah dengan menggunakan jaringan syaraf tiruan metode backpropagation. Agar hasil prediksi memiliki tingkat error yang rendah, diperlukan arsitektur jaringan backpropagation yang optimal, caranya dengan mengoptimalkan penggunaan data latih dan data uji yang diambil dari data sampel dan dibentuk menjadi tabel data di excel dan dihitung menggunakan excel, kemudian data tersebut digunakan pada MATLAB berupa input, hidden, dan output, dengan penggunaan neuron pada input layer berjumlah 12, hidden layer berjumlah 10, dan output layer berjumlah 1. Berdasarkan hasil proses pelatihan pada data latih menggunakan MATLAB akan mendapatkan hasil berupa Regression = 0.98559 dan MSE = 0.00099844. Perhitungan data latih juga akan mendapatkan bobot keluaran (bobot terbaik) yang digunakan pada data uji yang akan menghasilkan MSE berupa = 0.10574

Kata Kunci: curah hujan, prediksi, jaringan syaraf tiruan, backpropagation, MSE

PENDAHULUAN

Curah hujan dapat didefinisikan sebagai jumlah air yang jatuh di permukaan tanah datar selama periode tertentu yang diukur dengan satuan tinggi (mm) di atas *horizontal* bila tidak terjadi evaporasi, *runoff* dan infiltrasi. Satuan Curah Hujan adalah mm dimana Curah hujan dengan 1 milimeter artinya dalam luasan satu meter persegi menampung air satu millimeter atau tertampung air sebanyak satu liter.

Mengutip pernyataan dari jurnal milik Defanto Hanif Yoranda, Muhammad Tanzil Furqon dan Mahendra Data pada jurnal *Prediksi Intensitas Curah Hujan Menggunakan Metode Jaringan Saraf Tiruan Backpropagation* (10 Oktober 2018) , Intensitas curah hujan adalah besarnya hujan yang turun yang dinyatakan dalam satuan volume (mm/waktu) , besarnya intensitas hujan berbeda-beda tergantung dari seberapa lama hujan dan frekuensi turunnya hujan di tempat kejadian. Intensitas curah hujan yang tinggi bisa menyebabkan bencana yang dimana data menimbulkan banjir dan longsor sehingga perlu dilakukan peramalan untuk memperkirakan seberapa besar curah hujan yang akan datang.

Peramalan atau Prediksi adalah suatu proses untuk mengukur besarnya kebutuhan di masa yang akan datang, meliputi kuantitas, kualitas, waktu dan lokasi dengan metode statistika dalam rangka memenuhi permintaan barang ataupun jasa. Penggunaan data real time sering ditemukan pada data-data cuaca, riset biologis, astronomi dan financial. Pada bidang cuaca, data real time yang sering dipakai diantaranya digunakan untuk menunjukkan pergerakan nilai-nilai temperatur udara, kelembaban udara, kecepatan angin, dan curah hujan. Ada beberapa metode yang digunakan dalam melakukan peramalan dimana saya akan menyinggung tentang penggunaan metode Jaringan Syaraf Tiruan (JST)

JST (Jaringan Syaraf Tiruan / Artificial Neural Network) adalah sekelompok

jaringan unit kecil yang dimodelkan berdasarkan sistem saraf manusia yang mengubah struktur-struktur kecilnya untuk memecahkan masalah berdasarkan informasi-informasi yang dimasukkan ke jaringan tersebut.

Dari penjelasan di atas, maka diperlukan perhitungan yang akurat diantara metode-metode JST manakah yang dapat menghasilkan hasil yang paling akurat dan tepat. Informasi iklim berupa curah hujan yang diberikan secara berkala dan selalu di update secara real time.

METODE PENELITIAN

Prosedur Penelitian

1. Menentukan Data Latih Dan Data Uji
Proses pengambilan data latih dan data uji diambil dari range tanggal 1 januari 2011 sampai 31 desember 2013 untuk data latih, sedangkan data uji diambil dari tanggal 1 januari 2014 sampai tanggal 31 desember 2015. Dimana data latih itu tersendiri terdiri dari 2 yaitu, Data_latih dan Target_latih , dimana Data_latih terdiri dari bulan pertama hingga bulan ke-12 dan Target_latih merupakan bulan ke-13.Sama halnya dengan data uji yang terdiri dari Data_uji dan Target_uji, dimana Data_uji terdiri dari bulan pertama hingga bulan ke-12 dan Target_uji merupakan bulan ke-13.
2. Normalisasi Data
Normalisasi data bertujuan untuk memudahkan proses perhitungan dengan cara mengecilkan angka di dalam data menggunakan rumus

$$\frac{x - \min}{\max - \min} \times 0,8 + 0,1$$

Keterangan:
 x = data sesudah normalisasi
 \min = data sebelum normalisasi
 \max = angka terbesar dalam range data
 \min = angka terkecil dalam range data
3. Menentukan Neuron Pada Masing-Masing Lapisan

Menentukan banyaknya neuron yang ada pada lapisan input. Jumlah neuron yang ada di lapisan input berjumlah 12 neuron karena data jumlah data yang ada di lapisan input berjumlah 12 data curah hujan. Menentukan Banyaknya neuron yang ada pada lapisan hidden. Jumlah neuron yang ada di lapisan hidden berjumlah 10. Menentukan jumlah neuron pada lapisan output. Hasil yang dihasilkan dari lapisan output adalah data tunggal yang memberikan hasil prediksi data bulan selanjutnya, yang dimana hanya dibutuhkan 1 buah neuron di lapisan output karena tidak ada penggunaan kombinasi data.

4. Inisialisasi Bobot Pada Algoritma *Backpropagation*
Bobot yang digunakan dalam algoritma *Backpropagation sigmoid biner* adalah sebagai berikut :
net.trainParam.goal = 0,001
net.trainParam.show = 20
net.trainParam.epochs = 4000
net.trainParam.mc = 0,95
net.trainParam.lr = 0,1
5. Melakukan Pelatihan Data Latih
Pelatihan data latih menggunakan Data_latih yang terdiri dari 12 bulan dengan Target_latih bulan ke-13 menggunakan bobot parameter algoritma *Backpropagation*, yang dihitung terus menerus hingga mencapai angka terdekat dengan target. Tujuan dari melakukan pelatihan data latih ini adalah untuk mencari bobot keluaran terbaik yang nanti akan dipakai sebagai bobot perhitungan data uji.
6. Melakukan Denormalisasi Hasil Data Latih
Tujuan dari melakukan denormalisasi hasil data latih adalah untuk memastikan bahwa hasil yang dihasilkan dari data latih sesuai dengan data asli sebelum kita normalisasi yang akhirnya akan menghasilkan data hasil uji yang tepat. Rumus Denormalisasi. Setelah dilakukan denormalisasi, maka kita akan mendapatkan hasil data curah

hujan hasil prediksi yang kemudian akan dibandingkan dengan curah hujan data asli dari <http://climateknowledgeportal.worldbank.org/>

$$= \frac{(x - 0,1) \times (x_{max} - x_{min})}{x_{max} - x_{min}}$$

- Keterangan:
- = data sebelum normalisasi
 - = data sesudah normalisasi
 - = angka terbesar dalam range data
 - = angka terkecil dalam range data

7. Melakukan Pengujian Data Uji
Pengujian data uji mengambil Data_uji dan Target_uji yang dimana jumlah data tersebut sama dengan Data_latih dan Target_uji hanya saja berbeda angka. Pengujian biasanya menggunakan hasil bobot keluaran dari Hasil_latih Data latih yang diambil hanya keluaran bobot.
8. Melakukan Denormalisasi Data Uji
Tujuan Denormalisasi data uji adalah untuk mendapatkan hasil uji yang sebenarnya, dimana sama dengan Denormalisasi data latih, dimana denormalisasi data uji akan mendapatkan hasil akhir dari penelitian yang kemudian akan dibandingkan dengan data curah hujan yang asli dengan menggunakan validasi persentasi *figure* yang membandingkan hasil akhir prediksi dengan data asli.

HASIL DAN PEMBAHASAN

Transformasi Data

Tahun	Jan	Feb	Mar	Apr	Mai	Juni	Juli	Agst	Sep	Oktr	Nov	Des
2011	287,79	264,79	297,69	289,2	281,59	249,64	151,32	171,29	157,57	279,13	286,12	359,27
2012	376,93	391,99	451,71	389,42	385,18	175,33	188,61	158,18	163,59	335,63	372,19	399,74
2013	258,24	499,17	278,89	379,56	399,92	188,18	286,61	177,85	161,79	395,38	324,15	349,09
2014	269,88	147,08	199,89	246,79	224,18	214,46	152,32	209,12	156,71	191,12	218,99	217,74
2015	219,60	377,78	259,32	277,41	389,92	298,27	111,68	111,68	81,655	121,80	219,99	297,86

Tabel 1. Tabel Data Sebelum Normalisasi.

Tahun	Jan	Feb	Mars	Apr	Mei	Juni	Juli	Agst	Sep	Ok	Nov	Des
2011	0.702	0.839	0.728	0.730	0.689	0.3654	0.3003	0.2996	0.3723	0.6795	0.7035	0.8950
2012	0.875	0.718	0.9	0.743	0.730	0.3722	0.3850	0.3143	0.3340	0.5505	0.8115	0.7143
2013	0.844	0.862	0.875	0.684	0.711	0.4100	0.8653	0.7592	0.4269	0.4016	0.8174	0.8921
2014	0.435	0.587	0.417	0.795	0.518	0.4888	0.3657	0.4549	0.3158	0.3786	0.8371	0.8881
2015	0.713	0.678	0.586	0.671	0.834	0.4709	0.1809	0.1881	0.1	0.2205	0.3683	0.7390

Tabel 2. Tabel Data Sesudah Normalisasi.

Hasil perhitungan sampel dengan Matlab

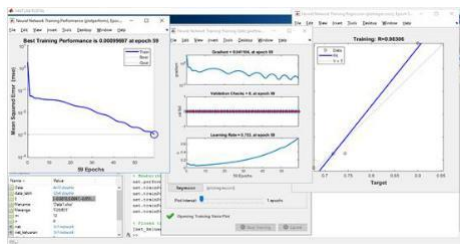
Parameter yang digunakan ini dikhususkan untuk mencari angka kesalahan terkecil pada angka 0.001, *epochs* (perulangan) sebanyak 4000, momentum *constant gradient* 0.95, dengan learning rate sebesar 0.1 dan Goal MSE berupa 0.001.

Parameter di atas dapat kita ubah dari segi apapun untuk mendapatkan hasil prediksi terbaik. Menggunakan Neuron pada input layer berjumlah 12 buah, penggunaan Neuron pada hidden layer berjumlah 10 dan penggunaan Neuron pada output layer berjumlah satu. Berikut adalah code yang digunakan untuk memberikan Neuron pada MatLab :

```
net=newff(minmax(data_latih),[101],{'log sig','purelin'},'traingdx');
```

Setelah menambahkan neuron pada masing masing layer, selanjutnya akan dilakukan proses training dengan menggunakan *code* :

```
[net_keluaran,tr,Y,E]=train(net,data_latih,t arget_latih);
```



Gambar 1. Screenshot hasil perhitungan matlab.

Berdasarkan Gambar 1., bisa kita simpulkan bahwa, performa pelatihan terbaik jatuh pada epochs (perulangan) ke-59 dengan angka 0.00099697 dengan *gradient* di bawah 0.95, learning rate yang cukup baik dengan meningkat terus hingga *epochs* ke 59, dan validasi kesalahan 0 yang artinya tidak ada kesalahan pada perhitungan. Gambar 4.3.1 yang menunjukkan fungsi linear Y menunjukkan

bahwa fungsi R menunjukkan angka = 0.98306 yang berarti adanya *relationship* / kecocokan antara *target* and *output*.

Hasil perhitungan data latih dengan Matlab

Perhitungan data latih di Matlab menggunakan parameter bobot pelatihan yang akan mempengaruhi hasil akhir dari pelatihan.

	2011	2012	2013
Bulan 1	0.702548	0.675972	0.614032
Bulan 2	0.639353	0.718137	0.862441
Bulan 3	0.738193	0.9	0.675837
Bulan 4	0.730713	0.743369	0.684728
Bulan 5	0.689967	0.730599	0.711928
Bulan 6	0.565482	0.372251	0.41008
Bulan 7	0.30033	0.385672	0.465345
Bulan 8	0.299617	0.314301	0.379207
Bulan 9	0.372356	0.334099	0.426954
Bulan 10	0.679599	0.550555	0.461604
Bulan 11	0.703519	0.811591	0.817459
Bulan 12	0.895698	0.714384	0.892173

Tabel 1. Data latih

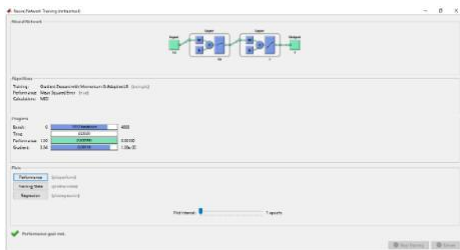
Parameter yang digunakan untuk melakukan pelatihan adalah:

1. net.performFcn='mse';Parameter ini mengubah parameter awal net.perform.Fcn='traingdm' menjadi property mse
2. net.trainParam.goal=0.001; Parameter goal disini adalah tujuan dari perhitungan dari MatLab, ketika perhitungan MSE sudah mencapai titik 0,001 maka perhitungan akan dihentikan.
3. net.trainParam.show = 20; menunjukkan 20 epochs pada progress window
4. net.trainParam.epochs = 4000; Epochs adalah jumlah perhitungan yang akandiulang terus menerus untuk mencapai parameter goal. Defaultrange epochs adalah 1000-4000.
5. net.trainParam.mc = 0.95; Menggunakan momentum *constant* pada angka 0.95.
6. net.trainParam.lr = 0.1;

7. Melakukan pelatihan dengan learning rate sebar 0.1

Tujuan dari pelatihan ini adalah untuk mencari bobot keluaran terbaik, yang dimana terdiri dari

1. bobot_hidden = nilai bobot antara layer input dan layer hidden
2. bobot_keluaran = nilai bobot antara layer hidden dan layer output
3. bias_hidden = nilai bias antara layer input dan layer hidden
4. bias_keluaran = nilai bias antara layer hidden dan layer output
5. jumlah_iterasi = jumlah perulangan dalam melakukan perhitungan
6. nilai_keluaran = Y
7. nilai_error = E
8. $error_MSE = (1/n) * \sum (nilai_error.^2)$;



Gambar 2. Hasil latihan

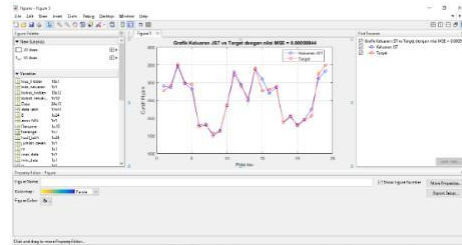
Berdasarkan Gambar 2., kita dapat bahwa terdapat 12 neuron pada input layer, 10 neuron pada hidden layer dan 1 neuron pada output layer, sesuai dengan parameter neuron yang kita inginkan, dengan epochs (perulangan) sebanyak 3410 kali, dalam waktu kurang dari 5 detik, performansi perhitungan mendekati parameter goal 0.001 dengan hasil 0.000998.

Setelah kita mendapatkan hasil pelatihan seperti gambar 2., maka selanjutnya data hasil latihan tersebut akan di denormalisasi kembali untuk di bandingkan dengan data target asli. Berikut adalah rumus untuk melakukan denormalisasi di MatLab :

1. hasil_latih = sim(net_keluaran, data_latih);
2. max_data = 351.712;
3. min_data = 84.6531;

$$hasil_latih = ((hasil_latih - 0.1) * (max_data - min_data) / 0.8) + min_data;$$

Setelah data di denormalisasi, selanjutnya adalah dengan membandingkan data target asli dengan hasil_latih untuk di bandingkan seperti gambar di bawah ini.



Gambar 3. Grafik perbandingan data keluaran dan target

Hasil perhitungan data latihan dengan Matlab

Setelah mendapatkan bobot terbaik pada pelatihan data latihan, maka selanjutnya adalah melakukan pengujian pada data uji dengan menggunakan bobot keluaran terbaik. Data uji yang digunakan untuk melakukan pengujian adalah sebagai berikut.

Tahun	Salah 1	Salah 2	Salah 3	Salah 4	Salah 5	Salah 6	Salah 7	Salah 8	Salah 9	Salah 10	Salah 11	Salah 12
2014	0.42762	0.24701	0.41797	0.70104	0.31800	0.40868	0.30292	0.43440	0.31388	0.33908	0.42218	0.38810
2015	0.71283	0.47834	0.39621	0.47133	0.43438	0.47633	0.38986	0.18817	0.1	0.23072	0.34934	0.73967

Tabel 2. Data uji

Perhitungan data uji dengan data latihan memiliki rumus yang berbeda, dimana data latihan menggunakan parameter yang bisa diubah, sedangkan pengujian pada data uji menggunakan bobot keluaran dari hasil data latihan yang tidak bisa kita ubah. Berikut adalah cara mengambil bobot keluaran pada data latihan untuk digunakan pada data uji

1. save net.mat net_keluaran
2. load net.mat

Berikut adalah rumus perhitungan data uji pada MatLab

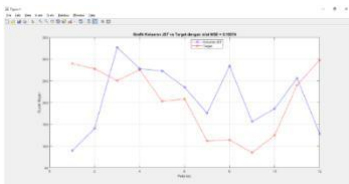
1. hasil_uji = sim(net_keluaran, data_uji)
2. nilai_error = hasil_uji - target_uji;

Setelah melakukan perhitungan hasil_uji dan nilai_error, maka akan melakukan proses denormalisasi untuk

membandingkan hasil_uji dan data asli. Rumus denormalisasi pada data_uji adalah sebagai berikut :

1. $\text{hasil_uji} = ((\text{hasil_uji} - 0.1) * (\text{max_data} - \text{min_data}) / 0.8) + \text{min_data}$;
2. $\text{error_MSE} = (1/n) * \sum (\text{nilai_error}^2)$ Menghitung nilai MSE dari nilai_error, untuk memastikan kebenaran hasil prediksi.

Setelah mendapatkan hasil_uji dan mendapatkan data denormalisasi, langkah selanjutnya adalah untuk membandingkan data dalam bentuk *figure*, yang akan mengeluarkan gambar seperti ini :



Gambar 4. Perbandingan data asli dan data prediksi

KESIMPULAN

Berdasarkan penelitian , maka dapat disimpulkan bahwa data curah hujan untuk Negara Indonesia pada tahun 2011-2015 sebagai berikut :

1. Nilai MSE pada Data_Latih yaitu 0.002624 dengan 12 *neuron* di input layer 10 *neuron* di *hidden layer* dan 1 *neuron* di output layer dengan 3 tahun data latih dan 2 tahun data uji diperoleh 3410 epochs dan MSE Data_Uji berupa 0.10574 dengan parameter *learning rate* 0.1 dan *gradient constant* 0.95
2. Penambahan data pada Data_latih dapat membuat bobot keluaran dari Data_Latih semakin bagus yang dimana bobot keluaran ini nantinya pada saat digunakan di Data_Uji dapat menghasilkan data yang lebih tepat.
3. Perubahan bobot parameter pada awal sebelum dilakukan pelatihan di Data_Latih dapat membuat bobot keluaran Data_Latih semakin bagus.

DAFTAR PUSTAKA

- [1] MARIHOT TP. Manalu. (2016). Jaringan Syaraf Tiruan Untuk Memprediksi Curah Hujan Sumatera Utara Dengan Metode Backpropagation (Studi kasus : BMKG Medan)
- [2] I Putu Sutawinaya, I Nyoman Gede Arya Astawa, Ni Kadek Dessy Hariyanti (2017). Perbandingan Metode Jaringan Syaraf Tiruan Pada Peramalan Curah Hujan
- [3] I Made Dwi Udayana Putra, G. K. Gandhiadi, Luh Putu Ida Harini. (2016). Impelentasi *Backpropagation Neural Network* Dalam Prakiraan Cuaca Di Daerah Bali Selatan
- [4] Yudhi Andrian, Erlinda Ningsih. (2018). Prediksi Curah Hujan Di Kota Medan Menggunakan *Neural Backpropagation Neural Network*
- [5] Defanto Hanif Yoranda, Muhammad Tanzil Furqon, Mahendra Data. (2018). Prediksi Intensitas Curah Hujan Menggunakan Metode Jaringan Syaraf Tiruan *Backpropagation*
- [6] Samuel Agus Febru Haryanto, Ernawati, Diah Puspitaningrum (2015). Impelemntasi Jaringan Syaraf Tiruan *Backpropagation* Untuk Memprediksi Cuaca (Studi kasis : Kota Bengkulu)
- [7] Wellie Sulistijanti . Peramalan Curah Hujan Wilayah Semarang Barat Dengan Algoritma *Resilient Backpropagation*
- [8] Yuzy Alfahnie, Very Dermawan, Lily Montarcih Limnataru. Analisis Tinggi Curah Hujan Berdasarkan Kondisi Klimatologi Dengan Menggunakan Jaringan Syaraf Tiruan
- [9] Lestari Handayani , Muhammad Adri (2015). Penerapan JST (Jaringan Syaraf Tiruan) Untuk Prediksi Curah Hujan (Studi kasus : Kota Pekanbaru)

- [10] Siang, J.J. (2009) Jaringan Syaraf Tiruan dan Pemograman Menggunakan Matlab, Yogyakarta: ANDI
- [11] Badrul Anwar (2011) Algoritma Metode *Backpropagation*.