

# **APLIKASI ARTIFICIAL NEURAL NETWORK UNTUK PERAMALAN ALIRAN SUNGAI BLEGA**

*Manyuk Fauzi, Minarni Nur Trilita*

Mahasiswa S3 MRSA, Jurusan Teknik Sipil-ITS dan Pengajar Jurusan Teknik Sipil Univ. Riau

Mahasiswa S3 MRSA, Jurusan Teknik Sipil-ITS dan Pengajar Jurusan Teknik Sipil UPN-Veteran

## **ABSTRACT**

*A multivariate non-linear model for synthetic generation (forecasting) of daily flow (discharge) series is presented in this study. The time series is built upon a multilayer feedforward neural network with an added multivariate random component normally distributed. The usual error backpropagation algorithm is used to train the network, with a sequential training scheme using shuffled patterns for a stochastic search in the weight space. Using neural network model, study case of Blega River-Madura island. The results by calculating show that the solution based on backpropagation algorithms are consistent with those based multivariate regression model. They also indicate that backpropagation model in this paper is reasonable and feasible.*

**Keyword** : forecasting flow, neural network

## **ABSTRAK**

Suatu model multivariate non-linier untuk membangkitkan seri data debit harian diperkenalkan dalam kajian ini. Suatu seri data dibangun di atas *multilayer feedforward neural network* dengan menambahkan suatu komponen multivariate acak dalam distribusi normal. Kesalahan yang umum *backpropagation algoritim* digunakan untuk melakukan pembelajaran jaringan, dengan suatu urutan rancangan pembelajaran menggunakan pengenalan pola untuk pencarian suatu bobot secara stokastik. Pemanfaatan model *neural network*, untuk kasus sungai Blega, pulau Madura. Hasil analisa menunjukkan bahwa penyelesaian berdasarkan *backpropagation algorithm* konsisten dengan analisa yang didasarkan pada model regresi *multivariate*. Hasil kajian juga menunjukkan bahwa *backpropagation* dalam kajian ini dapat diandalkan.

**Kata Kunci** : Peramalan aliran, *neural network*

## PENDAHULUAN

Terdapat banyak kegiatan operasi dan perencanaan yang berhubungan dengan sistem sumber daya air, membutuhkan peramalan kejadian yang akan datang. Sebagai contoh di dalam bidang hidrologi, di sana dibutuhkan peramalan aliran sungai dengan maksud untuk optimalisasi atau perencanaan sistem sungai tersebut. Di beberapa negara, banyak terdapat sistem sungai dengan luas area yang sangat besar akan tetapi memiliki koleksi data pengamatan yang kurang memadai. Kurang memadai dalam hal ini berarti jenis data yang terukur sedikit, kerapatan jaringan setasiun pengamatan tidak ideal serta periode pengamatan pendek. Sebagai contoh untuk jenis data yang kurang memadai adalah misalnya suatu sistem sungai hanya memiliki pengukuran debit, hujan dan klimatologi, akan tetapi tidak memiliki data kelembaban tanah, infiltrasi, kondisi aliran air tanah dan lain-lain.

Metode peramalan yang sering digunakan selama ini adalah metode *autoregressive moving average* atau ARMA yang dipelajari secara mendalam oleh George Box dan Gwilyn Jenkins (1976) [lihat Makridakis S. *et. al.*, 1991]. Pemanfaatan metode tersebut didasarkan pada pengembangan Model ARMA yang meliputi tiga arah yaitu identifikasi efisien serta prosedur penaksiran (untuk proses

AR, MA, dan ARMA campuran), perluasan dari hasil tersebut untuk mencakup deret berkala musiman (*seasonal time series*) dan pengembangan sederhana yang mencakup proses-proses non stasioner (*non-stationary processes*). Beberapa ahli hidrologi yang menggunakan metode ARMA untuk peramalan aliran antara lain Haltiner J.P. *et. al.* (1988), Gottardi *et. e l.* (1994) [lihat Lauzon N. *et. al.*, 2000].

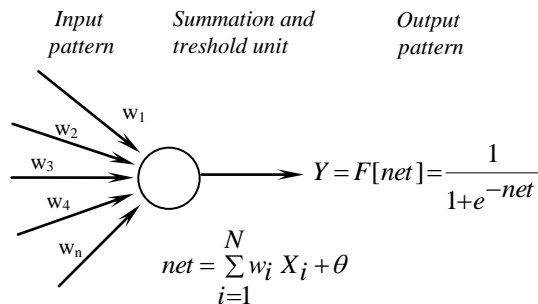
Namun sejak 10 tahun terakhir, beberapa ahli hidrologi melakukan kajian peramalan aliran dengan menggunakan metode baru yang disebut *artificial neural network* (ANN). Pada tahun 1999, Abrahart R. J. dan See L. dalam penelitiannya yang berjudul '*Neural Network vs ARMA Modelling : Constructing Benchmark Case Study of River Flow Prediction*', menulis tentang keuntungan ANN dibandingkan ARMA untuk pemodelan peramalan aliran sungai. Beberapa keuntungan ANN dibandingkan ARMA antara lain (i) bisa bekerja pada sistem yang tidak memiliki banyak jenis data (ii) distribusi dan stasioner data tidak harus diketahui, jika dibandingkan dengan model ARMA, stasioner data dan distribusi data normal menjadi suatu tuntutan, (iii) *neural network* lebih baik mengatasi masalah data yang *noise, distorted* dan *often incomplete*.

Berdasarkan uraian di atas, pada studi ini akan dikaji aplikasi metode ANN untuk

APLIKASI *ARTIFICIAL NEURAL NETWORK* UNTUK PERAMALAN ALIRAN SUNGAI BLEGA  
(Manyuk Fauzi & Minarni Nur Trilita)

peramalan aliran sungai Blega. Tujuan kajian ini adalah untuk memperoleh hasil terbaik dari pemodelan dengan ANN, berdasarkan variabel-variabel hidrologi

didefinisikan sebagai sistem pemrosesan informasi yang memiliki karakteristik performansi yang didasarkan atas pemodelan sistem syaraf biologis melalui pendekatan dari sifat-sifat komputasi biologis. ANN terdiri dari beberapa neuron atau node atau elemen pemrosesan (PE), dan terdapat hubungan antara neuron-neuron tersebut. Neuron akan mentransformasikan informasi (*signals*) yang diterima melalui jalur keluarannya menuju neuron yang lain, hubungan ini dikenal dengan istilah bobot (*weight, w*). Gambar 1 adalah struktur sederhana ANN dengan fungsi aktivasinya.



Gambar 1. Struktur Sederhana ANN

***Backpropagation Neural Network***

yang tersedia di lokasi yaitu hujan dan debit.

***ARTIFICIAL NEURAL NETWORK***

*Artificial neural network* (ANN)

*Backpropagation neural network* (BPNN) merupakan metode sistematis untuk *training* (kalibrasi) pada *multilayer* jaringan syaraf. Lapisan (*layer*) pertama terdiri dari satu set *input* dan lapisan akhir merupakan *output* (target). Diantara lapisan *input* dan lapisan *output* terdapat lapisan tersembunyi (*hidden layer*). *Input layer* merepresentasikan variabel input, *hidden layer* merepresentasikan ketidak linieran (*non-linearity*) dari sistem jaringan sedangkan *output layer* berisi variabel *output* (Danh Ng, *et. al.*, 1999). Pada masing-masing lapisan bisa terdapat beberapa node.

Secara sederhana BPNN dijelaskan sebagai berikut, suatu pola *input* dimasukkan ke dalam sistem jaringan untuk menghasilkan *output*, yang kemudian dibandingkan dengan pola *output* aktual. Jika tidak terdapat perbedaan antara keluaran dari sistem jaringan dan aktualnya, maka pembelajaran tidak diperlukan. Dengan kata lain, suatu bobot yang menunjukkan kontribusi *input node* ke *hidden node*, serta dari *hidden node* ke *output*, dimana jika terjadi selisih (*error*)

antara nilai *output* dari sistem jaringan dengan aktualnya, maka perbaikan bobot dilakukan secara mundur, yaitu dari *output* melewati *hidden node* dan kembali *input node*. Secara matematis dapat dijelaskan dalam algoritma *backpropagation* di bawah ini.

### Algoritma BPNN

Algoritma BPNN sebagaimana yang ditulis Fausett (1994) adalah sebagai berikut :

Langkah 0 : Inisialisasi bobot untuk *input* dan *hidden layer* secara acak dengan nilai antara 0 hingga 1.

Langkah 1 : Ketika berhenti, jika kondisi *false*, lakukan langkah 2 hingga 9.

Langkah 2 : Untuk setiap proses *training* lakukan langkah 3 hingga 8.

#### Feedforward

Langkah 3 : Setiap *input* ( $x_i$  dimana  $i = 1, 2, 3, \dots, n$ ) diberikan nilai antara 0 hingga 1, apabila memiliki nilai yang lebih besar 1, maka dilakukan normalisasi (melakukan transfer nilai antara 0 hingga 1)

Langkah 4 : Setiap *hidden node* ( $z_j$  dimana  $j = 1, 2, 3, \dots, p$ ) menjumlahkan sinyal input yang telah dikalikan nilai

bobot dan ditambah dengan bias dari *hidden layer*.

$$z\_in_j = v_{0j} + \sum_{i=1}^n x_i \cdot v_{ij}$$

dimana :

$z\_in_j$  = nilai fungsi aktivasi untuk menghitung sinyal *output* di *hidden node j*.

$x_i$  = nilai di *input node i*.

$v_{ij}$  = nilai bobot yang menghubungkan *input node i* dengan *hidden note j*.

$v_{0j}$  = nilai bias yang menghubungkan bias *node 1* dengan *hidden node j*.

$n$  = jumlah *input node* pada *input layer*.

Sinyal *output* dari *hidden node j* diberikan fungsi aktivasi sigmoid sebagai berikut :

$$z_j = f(z\_in_j) = \frac{1}{1 + e^{-z\_in_j}}$$

dimana  $z_j$  adalah sinyal output dari *hidden node j*.

APLIKASI ARTIFICIAL NEURAL NETWORK UNTUK PERAMALAN ALIRAN SUNGAI BLEGA  
(Manyuk Fauzi & Minarni Nur Trilita)

Langkah 5 : Setiap *output* ( $y_k$  dimana  $k = 1, 2, 3, \dots, m$ ) menjumlahkan sinyal input (*output* dari *hidden layer*) kemudian dikalikan dengan nilai bobot dan ditambah dengan bias dari *output layer*.

$$y\_in = w_{0k} + \sum_{i=1}^p z_i \cdot w_{jk}$$

dimana :

$y\_in$  = nilai fungsi aktivasi untuk menghitung sinyal *output*.

$z_i$  = nilai dari hasil *output hidden layer*, yang digunakan sebagai input untuk *output node k*.

$w_{jk}$  = nilai bobot yang menghubungkan *input node j* dengan *output node k*.

$w_{0k}$  = nilai bobot yang menghubungkan bias *node j* dengan *output node k*.

$p$  = jumlah *hidden node* pada *hidden layer*

Sinyal output dari *output node k* dengan memberikan fungsi aktivasi sigmoid biner sebagai berikut:

$$y = f(y\_in) = \frac{1}{1 + e^{-y\_in}}$$

Backpropagation

Langkah 6 : Menghitung nilai *error* dari *output node* pada *output layer* ( $y_k$  dimana  $k = 1, 2, \dots, m$ )

$$\delta_k = (t_k - y_k) f'(y\_in_k)$$

$$\delta_k = (t_k - y_k) y_k (1 - y_k)$$

dimana :

$\delta_k$  = nilai *error output node k*

$t_k$  = *output* yang diharapkan

$y_k$  = nilai hasil fungsi aktivasi pada *output layer node k*

Menghitung koreksi bobot yang nanti akan digunakan untuk memperbaiki nilai  $w_{jk}$  :

$$\Delta w_{jk} = \alpha \delta_k z_j$$

Menghitung koreksi bias yang nanti akan digunakan untuk memperbaiki nilai  $w_{0k}$ .

$$\Delta w_{0k} = \alpha \delta_k$$

Kirim nilai  $\delta_k$  ke unit yang ada di lapisan bawahnya

Langkah 7 : Tiap-tiap unit tersembunyi ( $Z_j$  dimana  $j = 1,2,3,\dots,p$ ) menjumlahkan delta inputnya (dari unit-unit yang berada pada lapisan atasnya) :

$$\delta_{in_j} = \sum_{k=1}^m \delta_k w_{jk}$$

kalikan nilai ini dengan turunan dari fungsi aktivasi untuk menghitung informasi error :

$$\delta_j = \delta_{in_j} f'(z_{in_j})$$

kemudian hitung koreksi bobot (yang nantinya akan digunakan untuk memperbaiki nilai  $v_{ij}$ ) :

$$\Delta v_{jk} = \alpha \delta_j x_k$$

hitung juga koreksi bias (yang nantinya akan digunakan untuk memperbaiki nilai  $v_{0j}$ ) :

$$\Delta v_{0j} = \alpha \delta_j$$

#### Updating bobot dan bias

Langkah 8 : Tiap-tiap unit output ( $Y_k$  dimana  $k = 1,2,3,\dots,m$ ) memperbaiki bias dan bobotnya ( $j = 0,1,2,\dots,p$ ) :

$$w_{jk}(\text{baru}) = w_{jk}(\text{lama}) + \Delta w_{jk}$$

Tiap-tiap unit tersembunyi ( $Z_j$  dimana  $k = 1,2,3,\dots,p$ ) akan memperbaiki bias dan bobot ( $i = 0,1,2,\dots,n$ ) :

$$v_{ij}(\text{baru}) = v_{ij}(\text{lama}) + \Delta v_{ij}$$

Langkah 9 : *Test stopping condition*

### STUDI KASUS

Kajian dilakukan terhadap sistem sungai Blega yang berada di pulau Madura propinsi Jawa Timur. Daerah Aliran Sungai (DAS) Blega memiliki luas 122 km<sup>2</sup>, curah hujan tahunan sebesar 1.300 mm dan suhu rata-rata 28°C. Data yang akan digunakan dalam kajian meliputi data hujan dan data debit. Tahun data debit/aliran yang digunakan untuk DAS Blega adalah 1989 – 1990, dimana data tahun 1989 digunakan dalam tahap kalibrasi atau pembelajaran (*training*) dan data tahun 1990 dalam tahap verifikasi (*testing*).

Di dalam kajian akan dibahas beberapa bentuk pemodelan peramalan aliran terhadap sistem sungai Blega. Tujuan membuat beberapa bentuk pemodelan adalah untuk mendapatkan hasil terbaik dari peramalan aliran. Bentuk pemodelan tersebut, yaitu :

$$[1]. Q_t = f([Q_{t-1}], [R_t], [R_{t-1}])$$

$$[2]. Q_t = f([Q_{t-1}], [Q_{t-2}], [R_t], [R_{t-1}])$$

- [3].  $Q_t = f([Q_{t-1}], [Q_{t-2}], [Q_{t-3}], [R_t], [R_{t-1}])$   
 [4].  $Q_t = f([Q_{t-1}], [Q_{t-2}], [Q_{t-3}], [R_t], [R_{t-1}], [R_{t-2}])$

**TRANFORMASI DATA**

Luketina, *et. al.* (2001) menulis metode transformasi data jika menggunakan fungsi sigmoid untuk aktivasi. Semua nilai masukan dan keluaran ditransformasikan ke dalam interval [0.05, 0.95]. Penyelesaian untuk setiap (masukan/keluaran) variabel  $X'$  adalah :

$$X' = 0.05 + 0.90 \left[ \frac{(X - X_{\min})}{X_{\max} - X_{\min}} \right]$$

dimana  $X'$  adalah nilai hasil transformasi,  $X_{\min}$  dan  $X_{\max}$  adalah nilai minimum dan maksimum dari data pengamatan.

Nilai debit diperoleh dari transformasi kembali nilai keluaran yang dihasilkan model ANN dengan melakukan 'inverse' dari persamaan di atas :

$$X = X_{\min} + (X_{\max} - X_{\min}) * \frac{(X' - 0.05)}{0.90}$$

**PERFORMA MODEL**

Performa model digunakan untuk mengukur ketepatan dari model. Dalam kajian ini, performa model yang digunakan adalah nilai tengah kesalahan kuadrat (*mean square error*, MSE). MSE merupakan suatu ukuran ketepatan model

dengan mengkuadratkan kesalahan untuk masing-masing *point* data dalam sebuah susunan data dan kemudian memperoleh rata-rata atau nilai tengah jumlah kuadrat tersebut.

$$MSE = \frac{\sum_{i=1}^N (y_i - \hat{y}_i)^2}{N} = \frac{\sum_{i=1}^N e_i^2}{N}$$

dimana  $y_i$  = nilai aktual data,  $\hat{y}_i$  = nilai hasil peramalan,  $N$  = jumlah data serta  $e_i$  = kesalahan per-*point* data.

Kriteria ketepatan model yang lain adalah Nash, Nash memberi suatu indikasi yang baik untuk pencocokan 1:1 antara simulasi dan pengamatan.

$$Nash = 1 - \left[ \frac{(Q_{obsv} - Q_{sim})^2}{(Q_{obsv} - \bar{Q}_{obsv})^2} \right]$$

dimana  $Q_{obsv}$  = data pengamatan,  $\bar{Q}_{obsv}$  = rata-rata data pengamatan, dan  $Q_{sim}$  = nilai hasil simulasi.

Kemudian digunakan prosedur umum menghitung kesalahan per-*point* data, dimana untuk deret berkala 'rumus' yang diikuti adalah :

$$data = pola + kesalahan$$

untuk memudahkan, kesalahan (*error*) ditulis dengan  $e$ , data dengan  $X$  dan pola data  $\bar{X}$ . Sebagai tambahan, *subscript*  $i$  ( $i = 1, 2, 3, \dots, n$ ) dicantumkan untuk

menunjukkan *point* data ke- $i$ , sehingga ditulis  $e_i = X_i - \bar{X}$ . Jika hanya ingin diketahui besaran kesalahan tanpa memperhatikan arah maka disebut dengan *absolut error* atau  $e_i = |X_i - \bar{X}|$ .

### HASIL DAN PEMBAHASAN

Tahap kalibrasi (*training*) model dirancang berhenti pada epoch ke-3000 untuk masing-masing model. Kalibrasi merupakan suatu proses atau penyesuaian nilai-nilai parameter suatu model untuk mencapai sebuah pencocokan yang terbaik antara variabel pengamatan dan variabel prediksi. Kecocokan antara data aktual (observasi) dengan hasil peramalan berdasarkan hasil kalibrasi ditunjukkan dengan nilai koefisien korelasi.

Nilai-nilai bobot dari jaringan ANN yang dicapai dari hasil kalibrasi akan digunakan dalam tahap verifikasi (*testing*). Ketepatan hasil peramalan, direpresentasikan dengan kriteria performa model. Untuk mengetahui tingkat korespondensi antara data aktual dengan hasil peramalan digunakan tolok ukur koefisien korelasi, dengan rumus :

$$R = \frac{\sum xy}{\sqrt{\sum x \sum y}}$$

dimana  $x = X - \bar{X}$ ,  $X$  = debit aktual,  $\bar{X}$  = rata-rata nilai  $X$ ,  $y = Y - \bar{Y}$ ,  $Y$  = debit hasil peramalan,  $\bar{Y}$  = rata-rata nilai  $Y$ .

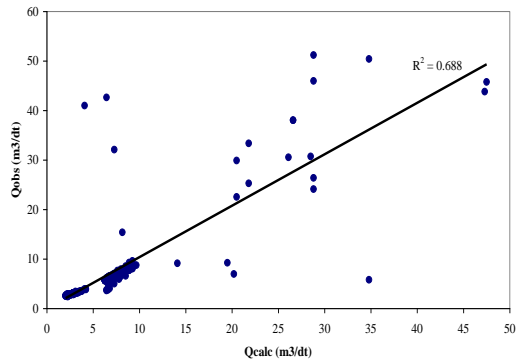
Tabel 1 menunjukkan performa peramalan dari tiap-tiap pemodelan, sedangkan Gambar 2a, 2b, 2c dan 2d menunjukkan *scatter plot* dari koresponding data aktual (*Qobs*) dengan hasil peramalan (*Qcalc*).

**Tabel 1. Performa model**

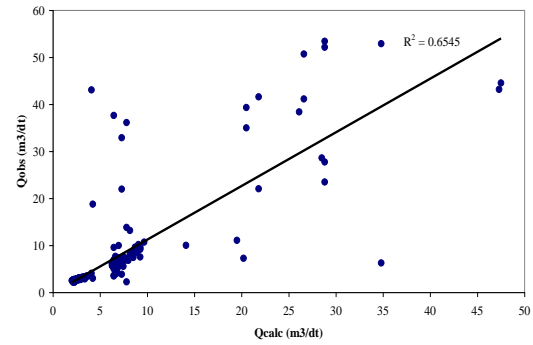
Model	MSE	Nash	Absolut Error		R
			Min	Max	
[1]	17,15	50,72	0,008	36,91	0,83
[2]	19,90	45,25	0,004	35,69	0,81
[3]	25,19	27,90	0,004	39,02	0,81
[4]	22,97	34,26	0,005	38,19	0,82



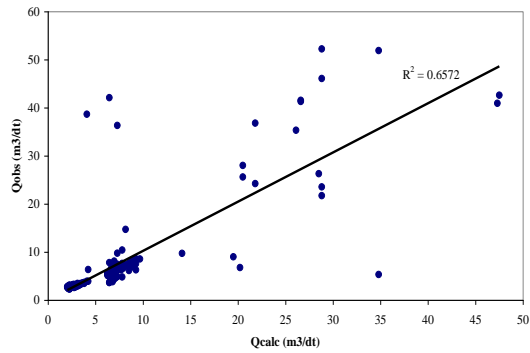
APLIKASI ARTIFICIAL NEURAL NETWORK UNTUK PERAMALAN ALIRAN SUNGAI BLEGA  
(Manyuk Fauzi & Minarni Nur Trilita)



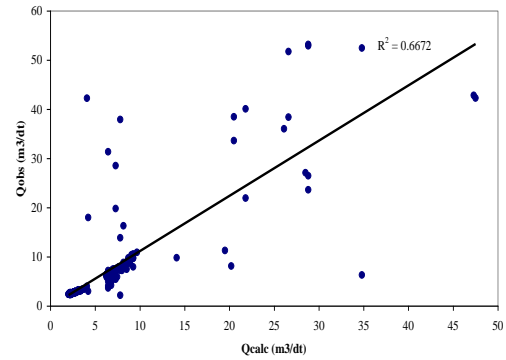
Gambar 2a. *Best linear fitting model [1]*



Gambar 2c. *Best linear fitting model [3]*



Gambar 2b. *Best linear fitting model [2]*



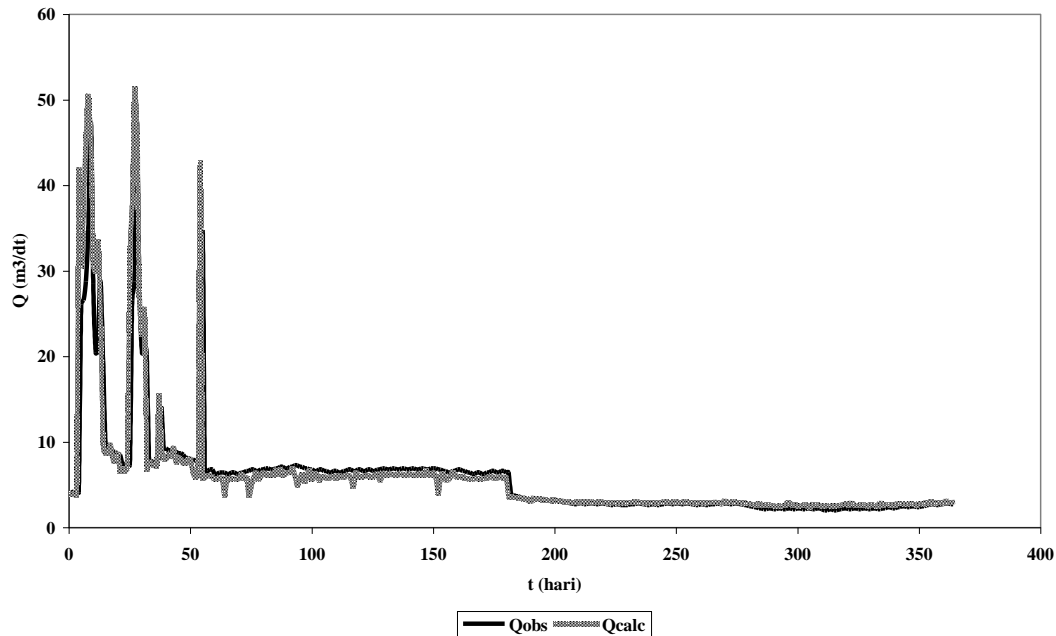
Gambar 2d. *Best linear fitting model [4]*

Dari Tabel 1, terlihat bahwa, koefisien korelasi antara aktual dengan peramalan memiliki hubungan langsung positif kuat yang ditunjukkan dengan nilai  $R$  berkisar 0,80. Jika digunakan *best linear fitting* sebagaimana diperlihatkan pada Gambar 2, diperoleh koefisien determinasi  $R^2$  sekitar 0,70; atau dengan kata lain bahwa ketepatan model regresi linier antara observasi dengan peramalan sebesar 0,70.

Jika didasarkan pada kriteria Nash yang merupakan pencocokan 1:1, maka tingkat keberhasilan ANN untuk aplikasi peramalan aliran sungai Blega mencapai peramalan (model [1]) ditunjukkan pada Gambar 3.

50,72% (tertinggi) untuk model [1]. Nilai tersebut menunjukkan bahwa proporsi kecocokan model hanya sekitar 50%. Nilai tersebut kurang memuaskan, karena masih terjadi kesalahan peramalan hampir 50%. Terdapat beberapa alasan rasional dari hal tersebut yaitu (i) kualitas data dan (ii) sistem model ANN yang belum optimal. Beberapa hal yang perlu dilakukan untuk mengatasi hal tersebut antara lain melakukan kajian arsitektur ANN yang lebih optimal serta penggunaan metode-metode pencarian minimum global. Hasil terbaik model

APLIKASI *ARTIFICIAL NEURAL NETWORK* UNTUK PERAMALAN ALIRAN SUNGAI BLEGA  
(Manyuk Fauzi & Minarni Nur Trilita)



**Gambar 3. Hidrograf banjir hasil peramalan ANN**

### KESIMPULAN

1. Pemanfaatan ANN untuk aplikasi praktis peramalan aliran sungai secara umum dapat diandalkan.
2. Hasil terbaik dari model ANN tergantung dari kualitas data, termasuk dalam hal ini panjang data sehingga model ANN lebih dapat melakukan pengenalan pola hubungan *input* dan *output*.

### DAFTAR PUSTAKA

- Abrahart R. J. and See L., 1999, *Neural Network vs ARMA Modelling : Constructing Benchmark Case Study of River Flow Prediction*, <http://www.geog.leeds.ac.uk/research/ccg.htm>, School of Geography, University of Nottingham,

- Nottingham NG7 2RD, United Kingdom
- Fausett L. V., 1999, *Fundamental of Neural Networks : Architecture, Algorithms and Applications*, Prentice Hall, Englewood Cliffs, New Jersey
- Lauzon N., Rousselto J., Birikundavyi S., and Trung H.T., 2000, *Real Time Daily Forecasting Using Black Box Models, Diffusion Process and Neural Network*, Canadian Journal Civil Engineering 27 : p671-682, Canada
- Luketina D.A. & Wang F., 2001, *Hydrological Modelling*, Asian Institute Technology (AIT), Thailand
- Makridakis S., et. al., 1991, *Metode dan Aplikasi Peramalan*, Penerbit Erlangga, Jakarta
- Manyuk Fauzi, Nadjadji Anwar, Edijatno, M. Isa Irawan, 2004 *Pemodelan Transformasi Hujan-Debit Dengan Pendekatan Soft Computing*, Jurnal Teknik FT. Unibraw ISSN 0854-2139, Vol. XI No.3, pp 188-196