

MODEL PERAMALAN DEBIT ALIRAN SUNGAI MENGGUNAKAN METODE GABUNGAN SELF ORGANIZING MAPS – ARTIFICIAL NEURAL NETWORK (Studi Kasus: Sungai Tapung Kiri)

Muhammad Idraq Ibnuts Tsauri¹, Imam Suprayogi², Manyuk Fauzi²

¹Mahasiswa Jurusan Teknik Sipil, Fakultas Teknik, Universitas Riau

²Dosen Jurusan Teknik Sipil, Fakultas Teknik, Universitas Riau

Kampus Bina Widya Jl. HR Soebrantas KM 12,5 Pekanbaru, Kode Pos 28293

Email : muhammad.idraq@student.unri.ac.id

Abstract

Siak river basin now in critical condition. This condition is due to land use changes in the upper body and cause silting in the river. The silting decreasing the capacity of the river so when heavy rain occurs, the rivers could not accommodate it and it will cause flooding. The flood disaster which is very difficult to predict and lack of water resources at dry season made the citizens lack of preparation to deal with. A solution to anticipate that is to provide an early warning system. In order for the system to work well, it needs a prediction method that could provide good quality of data, the method called SOM-ANN. SOM-ANN method consisted by two different types of algorithms, there are Self Organizing Maps (SOM) and Backpropagation. The purpose of this study was to test the reliability of SOM-ANN method in discharge predicting in Tapung Kanan River. The weighting result from SOM's learning applied to Backpropagation's learning, so that the pattern recognizing becomes faster and gaining more accuracy. By a comparison with ANN method, SOM-ANN method can improve a better performance and accuracy of predicting results with 1099% increase in performance and 44,74% increase in accuracy with an error value MSE = 0,001725, so that the discharge prediction modelling can be used to predict the discharge in the future.

Keywords : artificial neural network, self organizing maps, discharge, prediction modelling

A. PENDAHULUAN

Hingga saat ini, DAS (Daerah Aliran Sungai) Siak masih dikategorikan sebagai DAS kritis yang merupakan kawasan rawan bencana banjir (Setyawati, 2009). Banjir dan genangan sering terjadi di daerah yang dilalui Sungai Siak, salah satunya ialah Kota Pekanbaru. Luas daerah rawan banjir di Kota Pekanbaru mencapai 8.775 hektar (BWS Sumatera III, 2011).

Curah hujan yang tinggi saat musim penghujan dan daya infiltrasi tanah yang kurang baik akibat perubahan tata guna lahan mengakibatkan limpasan permukaan saat hujan menjadi besar. Limpasan tersebut mengalir ke sungai sehingga menimbulkan sedimentasi dan memicu terjadinya pendangkalan yang mengakibatkan daya tampung pada badan sungai berkurang sehingga air sungai akan

meluap dan memicu terjadinya bencana banjir.

Ketersediaan sumber daya air pada musim kemarau turut menjadi perhatian. Pembukaan hutan sekunder untuk keperluan lahan pertanian dan kebun penduduk telah menyebabkan terbentuknya lahan-lahan kritis. Lahan kritis yang ada di wilayah Sungai Siak sebesar $\pm 5.500 \text{ km}^2$, meningkat 50% dari Tahun 2005 yang sebesar $\pm 3.600 \text{ km}^2$ (Dewan SDA Nasional, 2013). Hilangnya fungsi hutan sebagai penyimpan cadangan air menyebabkan ketersediaan sumber daya air pada musim kemarau menjadi berkurang.

Salah satu tindakan preventif dalam menghadapi permasalahan-permasalahan tersebut ialah dengan mengembangkan

sistem peringatan dini (*early warning system*). Sebagai acuan pada sistem tersebut, data debit prediksi dapat diperoleh dengan salah satu metode yaitu Metode *Artificial Neural Network* (ANN) atau Jaringan Saraf Tiruan.

Algoritma *Backpropagation* merupakan algoritma yang umum digunakan dalam Metode ANN untuk menyelesaikan masalah yang berkaitan dengan peramalan atau prediksi. Namun kekurangan dari algoritma ini ialah penentuan bobot awal yang acak sehingga pelatihan jaringan menjadi kurang optimal. Berdasarkan hal tersebut, maka dalam penelitian ini akan dikembangkan *Self Organizing Maps* (SOM) yang dikombinasikan dengan ANN untuk meningkatkan performansi dan akurasi dalam memprediksi nilai debit.

Berdasarkan uraian di atas, maka penelitian dalam rangka tugas akhir ini akan difokuskan pada kombinasi algoritma SOM dan ANN untuk menghasilkan prediksi yang lebih akurat dan dengan proses yang lebih cepat.

Berdasar latar belakang penelitian tugas akhir di atas, maka rumusan masalah dari penelitian ini ialah menguji kemampuan algoritma SOM dalam menghasilkan pembobotan pada pola data untuk membantu kinerja algoritma *Backpropagation* sehingga dapat diketahui tingkat keandalan Metode SOM-ANN dalam memprediksi debit untuk masa yang akan datang.

B. TINJAUAN PUSTAKA

B.1 Pemodelan Hidrologi

Model merupakan representasi atau gambaran tentang sistem, obyek atau benda dan kejadian. Representasi tersebut dinyatakan dalam bentuk sederhana yang dapat dipergunakan untuk berbagai macam tujuan penelitian. Penyederhanaan dilakukan secara representatif terhadap perilaku proses yang relevan dari keadaan sebenarnya. Penggunaan model sebagai usaha untuk memahami suatu sistem yang rumit merupakan teknik pengkajian yang

lebih sederhana dibandingkan jika melalui keadaan yang sebenarnya.

Pemodelan terbagi menjadi 2 jenis, yaitu secara konseptual dan *black box*. Prakiraan debit secara konseptual adalah dengan memperhitungkan semua aspek siklus hidrologi yang ada dalam satu daerah aliran sungai. Metode ini mempunyai banyak kendala di antaranya sulitnya mendapatkan data di lapangan, sedangkan sistem prediksi banjir model *black-box* hanya menggunakan data debit yang didapat dari AWLR. Dalam penelitian ini, digunakan model *black box* untuk membangun pemodelan hidrologi debit-debit dengan *input* data debit yang didapat dari AWLR dan *output* data debit pada masa akan datang melalui pendekatan *soft computing* berbasis kecerdasan buatan dengan metode jaringan saraf tiruan.

B.2 Debit

Menurut Asdak (1995) debit adalah laju aliran air yang melewati suatu penampang melintang sungai/aliran air per satuan waktu. Satuan debit yang lazim digunakan ialah m³/detik. Dalam hal ini, debit dapat juga dikatakan sebagai satuan besaran air yang keluar dari Daerah Aliran Sungai (DAS).

Debit air dapat dihitung dengan menggunakan rumus sebagai berikut:

$$Q = A \times V \quad (1)$$

Dengan Q adalah debit aliran (m³/s), A adalah luas penampang (m²) dan V adalah kecepatan aliran (m/s).

B.3 AWLR (*Automatic Water Level Recorder*)

Automatic Water Level Recorder (AWLR) adalah alat untuk mengukur tinggi muka air pada sungai, danau, maupun aliran irigasi. AWLR merupakan alat pengganti sistem pengukuran tinggi air konvensional dengan sistem perekaman data masih dilakukan secara manual sehingga sistem pengukuran dan penyimpanan data menjadi kurang tepat dan akurat.

Alat ini banyak digunakan pada pengukuran parameter dalam kegiatan hidrologi pada daerah aliran sungai, sehingga alat tersebut dapat digunakan untuk melakukan berbagai aplikasi di bidang hidrologi, salah satunya ialah untuk mencatat tinggi muka air secara otomatis yang kemudian akan dikonversi dengan liku kalibrasi untuk mengetahui debit yang mengalir pada sungai tersebut.

B.4 Kecerdasan Buatan (*Artificial Intelligence*)

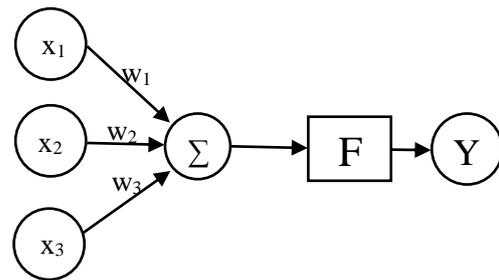
Menurut Russel dan Norvig (2010), kecerdasan buatan atau *Artificial Intelligence* (AI) merupakan bagian ilmu pengetahuan yang digunakan untuk menyelesaikan masalah manusia dengan cara memahami, memprediksi dan memanipulasi. Kecerdasan tersebut dibuat oleh sistem menggunakan algoritma tertentu sehingga seolah-olah dapat berpikir seperti manusia. AI biasanya dihubungkan dengan Ilmu Komputer, akan tetapi AI juga dapat dihubungkan dengan bidang-bidang lainnya seperti untuk keperluan pengenalan pola, peramalan, pengelompokan dan lain-lain. Kemampuan untuk mengkombinasikan pengetahuan dari semua bidang ini pada akhirnya akan bermanfaat bagi kemajuan dalam upaya menciptakan suatu kecerdasan buatan.

B.5 Jaringan Saraf Tiruan (*Artificial Neural Network*)

Jaringan Saraf Tiruan atau *Artificial Neural Network* (ANN) merupakan bagian dari sistem kecerdasan buatan yang digunakan untuk memproses informasi yang didesain dengan menirukan cara kerja otak manusia dalam menyelesaikan masalah dengan melakukan proses belajar melalui perubahan bobot sinapsisnya (Hermawan, 2006).

Pada jaringan saraf biologis, setiap sel saraf (*neuron*) akan memiliki satu inti sel yang bertugas untuk melakukan pemrosesan informasi. Informasi yang datang akan diterima oleh *dendrit*. Selain menerima informasi, *dendrit* juga menyertai *axon* sebagai keluaran dari suatu

pemrosesan informasi. Informasi hasil olahan ini akan menjadi masukan bagi *neuron* lainnya.



Gambar 1 Arsitektur Sederhana ANN

Gambar 1 merupakan contoh arsitektur sederhana ANN yang menunjukkan bahwa Y menerima input dari *neuron* x_1 , x_2 , dan x_3 dengan bobot hubungan masing-masing adalah w_1 , w_2 , dan w_3 .

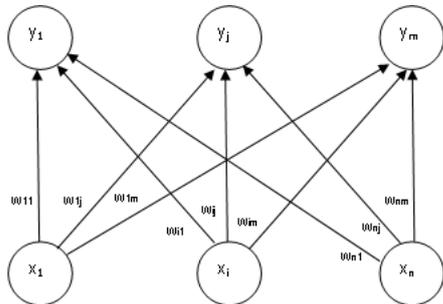
B.6 *Self Organizing Maps*

Self Organizing Maps (SOM) merupakan salah satu jaringan dengan pelatihan tak terbimbing (*unsupervised training*) yang banyak dipakai, antara lain untuk mengeksplorasi data *mining* dan melakukan *clustering*, yaitu membagi *input* ke dalam beberapa kelompok. SOM pertama kali diperkenalkan oleh Tuevo Kohonen dari University of Helsinki pada tahun 1981 (Kristanto, 2004).

Dalam SOM, masukan berupa vektor yang terdiri dari n komponen yang akan dikelompokkan dalam maksimum m buah kelompok. Keluaran jaringan adalah kelompok yang paling dekat atau mirip dengan masukan yang diberikan. Ukuran yang dipakai adalah jarak *Euclidean* yang paling minimum (Siang, 2009).

Vektor bobot untuk sebuah unit *cluster* menyediakan sebuah contoh dari pola input yang dikumpulkan dalam *cluster*. Selama proses *self organizing*, unit *cluster* yang mempunyai bobot dicocokkan dengan pola input yang terdekat dan dipilih sebagai pemenang. Unit pemenang dan unit tetangganya memperbaiki bobot mereka masing-masing (Kristanto, 2004). SOM disusun oleh sebuah lapisan unit

input yang dihubungkan seluruhnya ke lapisan unit output, yang kemudian unit-unit diatur dalam sebuah struktur jaringan. Secara umum arsitektur jaringan SOM dapat dilihat pada Gambar 2.



Gambar 2 Arsitektur SOM
(Sumber: Kristanto, 2004)

Gambar 2 memperlihatkan arsitektur SOM yang terdiri dari dua lapisan (*layer*), yaitu lapisan *input* dan lapisan *output*. Setiap *neuron* dalam lapisan *input* terhubung dengan setiap *neuron* pada lapisan *output*. Setiap *neuron* dalam lapisan *output* merepresentasikan kelas dari *input* yang diberikan. Dalam beberapa kasus, terkadang ditambahkan *hidden layer* di antara *layer input* dan *layer output* untuk meningkatkan kualitas pembelajaran dan mengontrol jumlah *cluster* yang diinginkan.

Algoritma pelatihan Jaringan SOM adalah sebagai berikut (Salim & Jauhari, 2016):

1. Data observasi didefinisikan sebagai *input* untuk kemudian diolah menggunakan SOM. Neuron pada lapisan *input* (*neuron input*) sebanyak n dinotasikan sebagai x_1, x_2, \dots, x_n dan neuron pada lapisan *output* (*neuron output*) sebanyak m dinotasikan sebagai y_1, y_2, \dots, y_m . Bobot koneksi antara *neuron input* dan *output* dinotasikan sebagai w_{ij} dan ditentukan secara acak antara 0 dan 1.
2. Selama kondisi penghentian bernilai salah, lakukan langkah 3 – 8.
3. Untuk setiap masukan (x_1, x_2, \dots, x_n) lakukan langkah 4 – 6.

4. Hitung jarak vector *input* terhadap bobot koneksi d_j untuk masing-masing *neuron output* dengan menggunakan rumus:

$$d_j = \sum_{i=1}^n (w_{ij} - x_i)^2 \quad (2)$$

5. Cari indeks j di mana d_j minimum.
 6. Perbaharui bobot koneksi untuk setiap w_{ij} .
 7. Uji kondisi penghentian.
- Syntax* yang digunakan untuk membangun jaringan SOM pada *Software MATLAB* ialah:

$$net = newc(PR,S) \quad (3)$$

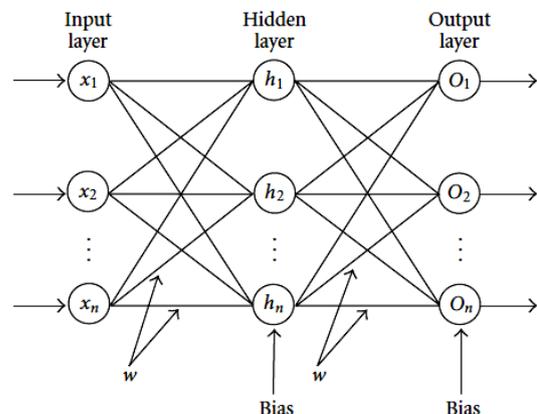
Keterangan:

PR = matriks dengan ordo $R \times 2$ yang berisi nilai minimum dan maksimum

S = jumlah neuron target

B.7 Backpropagation

Backpropagation adalah algoritma pembelajaran atau pelatihan terbimbing dan biasanya digunakan banyak lapisan untuk mengubah bobot-bobot yang terhubung dengan *neuron-neuron* yang ada pada bagian tersembunyi (Kusumadewi, 2003). Pada *backpropagation*, setiap unit yang berada di *input layer* terhubung dengan setiap unit yang ada di *hidden layer*. Hal serupa berlaku pula pada *hidden layer*, setiap unit pada *hidden layer* terhubung dengan setiap unit yang ada pada *output layer*, seperti yang dapat dilihat pada Gambar 3.



Gambar 3 Jaringan *Backpropagation*
(Sumber: Goel, 2011)

Gambar 3 memperlihatkan arsitektur *Backpropagation* yang terdiri dari:

- Input layer* (1 buah). *Input layer* terdiri dari neuron-neuron atau unit-unit *input*.
- Hidden Layer* (minimal 1 buah). *Hidden layer* terdiri dari unit-unit tersembunyi
- Output layer* (1 buah). *Output layer* terdiri dari unit-unit keluaran.

Algoritma *backpropagation* menggunakan *error output* untuk mengubah nilai-nilai bobot-bobotnya dalam arah mundur (*backward*). Untuk mendapatkan *error* ini, tahap perambatan (*forward propagation*) harus dikerjakan terlebih dahulu. Pelatihan dengan Algoritma *Backpropagation* meliputi 3 fase. Berikut merupakan algoritma dalam pelatihan *Backpropagation* (Kusumadewi, 2004):

- Inisialisasi bobot (ambil bobot awal dengan nilai random yang cukup kecil). Tetapkan: Maksimum *Epoch*, Target *error*, *Learning rate*, Momentum. Inisialisasi: $Epoch = 0$, $MSE = 1$
- Kerjakan langkah-langkah berikut selama ($Epoch < \text{Maksimum Epoch}$) dan ($MSE > \text{Target error}$)
- $Epoch = Epoch + 1$
- Untuk tiap-tiap pasangan elemen yang akan dilakukan pembelajaran, kerjakan langkah-langkah berikutnya.

Fase I : *Feed Forward*

- Tiap-tiap unit *input* (X_i , $i=1,2,\dots,n$) menerima sinyal X_i dan meneruskan sinyal tersebut ke semua unit pada lapisan yang ada di atasnya (lapisan tersembunyi).
- Tiap-tiap unit pada suatu lapisan tersembunyi (Z_j , $j=1,2,\dots,p$) menjumlahkan sinyal-sinyal *input* terbobot:

$$Z_{in_j} = b1_j + \sum_{i=1}^n X_i V_{ij} \quad (4)$$

Gunakan fungsi aktivasi untuk menghitung sinyal *output*:

$$Z_j = f(Z_{in_j}) \quad (5)$$

- Tiap-tiap unit *output* Y_k ($k=1,2,\dots,m$) menjumlahkan sinyal-sinyal *input* terbobot:

$$Y_{-in_k} = b2_k + \sum_{i=1}^p Z_i W_{jk} \quad (6)$$

Gunakan fungsi aktivasi untuk menghitung sinyal *output*:

$$Y_k = f(Y_{-in_k}) \quad (7)$$

Lalu kirimkan sinyal tersebut ke semua unit di lapisan atasnya (unit-unit *output*).

Fase II : *Backpropagation*

- Tiap-tiap unit *output* Y_k ($k=1,2,\dots,m$) menerima target pola yang berhubungan dengan pola *input* pelatihan, hitung informasi *error*nya:

$$\delta = (t_k - Y_k) f'(Y_{-in_k}) \quad (8)$$

$$\varphi 2_{jk} = \delta_k Z_j \quad (9)$$

$$\beta 2_k = \delta_k \quad (10)$$

Kemudian hitung koreksi bobot

$$\Delta W_{jk} = \alpha \varphi 2_{jk} \quad (11)$$

Hitung juga koreksi bias

$$\Delta b 2_k = \alpha \beta 2_k \quad (12)$$

- Tiap-tiap unit tersembunyi (Z_j , $j=1,2,\dots,p$) menjumlahkan delta *input*nya

$$\delta_{in_j} = b 2_k + \sum_{k=1}^m \delta 2_k W_{jk} \quad (13)$$

Kalikan nilai ini dengan turunan dari fungsi aktivasinya untuk menghitung informasi *error*:

$$\delta 1_j = \delta_{in_j} f'(Z_{in_j}) \quad (14)$$

$$\varphi 1_{ij} = \delta 1_j X_j \quad (15)$$

$$\beta 1_j = \delta 1_j \quad (16)$$

Kemudian hitung koreksi bobot yang nantinya akan digunakan untuk memperbaiki nilai *output*

$$\Delta V_{ij} = \alpha \varphi 1_{ij} \quad (17)$$

Hitung juga koreksi bias yang nantinya akan digunakan untuk memperbaiki nilai *output*

$$\Delta b 1_j = \alpha \beta 1_j \quad (18)$$

Fase III : Modifikasi bobot

- Tiap-tiap unit *output* Y_k ($k=1,2,\dots,m$) memperbaiki bias dan bobotnya ($j=0,1,2,\dots,o$):

$$W_{jk}(\text{baru}) = W_{jk}(\text{lama}) + \Delta W_{jk} \quad (19)$$

$$b 2_k(\text{baru}) = b 2_k(\text{lama}) + \Delta b 2_k \quad (20)$$

Tiap-tiap unit tersembunyi (Z_j , $j=1,2,\dots,p$) memperbaiki bias dan bobotnya ($i=0,1,2,\dots,n$):

$$V_{ij}(\text{baru}) = V_{ij}(\text{lama}) + \Delta V_{ij} \quad (21)$$

$$b_{1j}(\text{baru}) = b_{1j}(\text{lama}) + \Delta b_{1j} \quad (22)$$

b) Hitung MSE

Ketiga fase tersebut diulang-ulang terus hingga kondisi penghentian dipenuhi. Umumnya kondisi penghentian yang sering dipakai adalah jumlah iterasi atau kesalahan. Iterasi akan dihentikan jika jumlah iterasi yang dilakukan sudah melebihi jumlah maksimum iterasi yang ditetapkan atau jika kesalahan yang terjadi sudah lebih kecil dari batas toleransi yang diizinkan. *Syntax* yang digunakan untuk membangun jaringan *Backpropagation* menggunakan *Software* MATLAB ialah:

$$\text{net} = \text{newff}(\text{PR}, [\text{S1 S2 ... SN}], \{\text{TF1 TF2 ... TFN}\}, \text{BTF}, \text{BLF}, \text{PF} \quad (23)$$

Keterangan:

PR = matriks ordo $R \times 2$ yang berisi nilai minimum dan maksimum R buah elemen masukannya

S = jumlah unit pada layer ke- i sampai n

TF = fungsi aktivasi yang dipakai pada layer ke- i hingga ke- n

BTF = fungsi pelatihan jaringan, dengan nilai *default*-nya adalah TRAINGD

BLF = fungsi perubahan bobot/bias, nilai *default*-nya adalah LEARNGDM

PF = fungsi perhitungan *error*, dengan isian *default* = MSE

B.8 Metode Evaluasi Kinerja ANN

1. Koefisien Korelasi

Koefisien korelasi (R) adalah nilai yang menunjukkan kuat atau tidaknya hubungan linier antar dua variabel yang memperlihatkan perbandingan nilai antara hasil prediksi dengan nilai yang sebenarnya (target). Hal tersebut diinterpretasikan dengan kriteria sebagai berikut (Sarwono, 2008):

- $R = 0$, tidak ada korelasi antara dua variabel,
- $0 < R \leq 0,25$, korelasi sangat lemah,
- $0,25 < R \leq 0,50$, korelasi cukup,
- $0,50 < R \leq 0,75$, korelasi kuat,
- $0,75 < R \leq 0,99$, korelasi sangat kuat, dan

f) $R = 1,00$, korelasi sempurna.

2. MSE (Mean Square Error)

MSE adalah nilai dari kuadrat error. Error yang ada menunjukkan seberapa besar perbedaan hasil prediksi dengan nilai observasi. MSE dirumuskan sebagai berikut:

$$MSE = \frac{\sum(x-y)^2}{N} \quad (24)$$

Keterangan:

MSE = Mean Squared Error

N = Jumlah Data

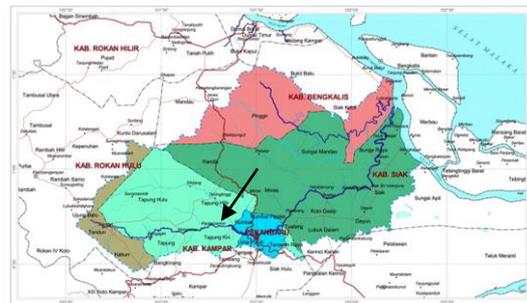
X = Nilai Observasi

Y = Nilai Prediksi

C. METODOLOGI PENELITIAN

C.1 Lokasi Penelitian

Lokasi penelitian adalah Sungai Tapung Kanan, dengan lokasi Stasiun AWLR di Pantai Cermin, Kecamatan Tapung, Kabupaten Kampar, Provinsi Riau.

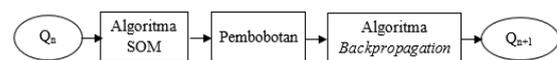


Gambar 4 Lokasi Penelitian

(Sumber: PSDA Wilayah Sungai Siak, 2013)

C.2 Pengembangan Model

Skema penelitian dapat dilihat sebagai berikut:



Gambar 5 Skema Sistem Prediksi Debit Menggunakan Algoritma SOM-ANN

Berdasarkan Gambar 5, Q_n yang merupakan debit yang mengalir pada Sungai Siak pada hari ke- n berperan sebagai data *input*, kemudian data tersebut akan melalui proses *clustering* atau diklasifikasikan menurut polanya dengan menggunakan algoritma SOM. Pembobotan pada masing-masing *neuron*

hasil dari proses *clustering* tersebut menjadi faktor pembimbing pada pembelajaran dengan algoritma *Backpropagation*. Jaringan yang telah dilatih, diuji, dan divalidasi kemudian dijadikan sebagai model prediksi. Tahapan pembangunan model pada *Backpropagation* yaitu pelatihan (*training*) dan pengujian (*testing*).

C.3 Pengumpulan dan Pengelompokan Data

Data yang digunakan dalam penelitian ini ialah data dari AWLR yang telah dikonversi menjadi data debit dari tahun 2002 sampai dengan tahun 2006 dengan persamaan liku kalibrasi $Q = 14,78 \times (H + 0,384)^{1,580}$. Sumber data diambil dari Balai Wilayah Sungai (BWS) Sumatera III Provinsi Riau, Jalan Cut Nyak Dien 01, Pekanbaru. Data tersebut didistribusikan sebagai berikut:

1. Data debit tahun 2002 sampai dengan tahun 2005 digunakan sebagai data yang akan diperoleh pembobotannya dengan menggunakan algoritma SOM.
2. 70% dari total data digunakan sebagai data pelatihan (*training*) pada algoritma *Backpropagation*.
3. 30% dari total data digunakan sebagai data pengujian (*testing*) pada algoritma *Backpropagation*.
4. Seluruh data digunakan sebagai data *input* data validasi (*validasi*) pada algoritma *Backpropagation*.

C.4 Fungsi Aktivasi dan Normalisasi Data

Pemilihan fungsi aktivasi dilihat berdasarkan *trend* dan nilai data. Berdasarkan data yang akan diolah, pola debit tidak bersifat linear dan nilai minimum dari data debit ialah 0, dengan kata lain tidak ada yang bernilai negatif. Maka dari itu, fungsi yang memenuhi kriteria tersebut adalah Fungsi *Sigmoid Biner* yang memiliki range 0 hingga 1. Agar jaringan tidak mengalami kegagalan dalam proses pelatihan dan pengujian, maka range data debit yang akan diolah harus memiliki *range* yang sama dengan

fungsi aktivasi yang digunakan. Dikarenakan Fungsi *Sigmoid* memiliki nilai maksimum = 1, maka setiap data debit untuk *input* dan *target* harus terlebih dahulu dinormalisasi sehingga semua nilainya memiliki *range* yang sama dengan Fungsi *Sigmoid*. Rumus normalisasi adalah sebagai berikut (Siang, 2009):

$$x' = \frac{0,8(x-x.min)}{x.max-x.min} + 0,1 \quad (25)$$

Keterangan:

x = data asli

$x.max$ = data maksimum asli

$x.min$ = data minimum asli

Data yang telah melewati proses pelatihan, pengujian maupun validasi kemudian didenormalisasikan untuk memperoleh nilai asli. Rumus denormalisasi adalah sebagai berikut (Siang, 2009):

$$x = \frac{(x'-0,1)(x.max-x.min)}{0,8} + x.min \quad (26)$$

Keterangan:

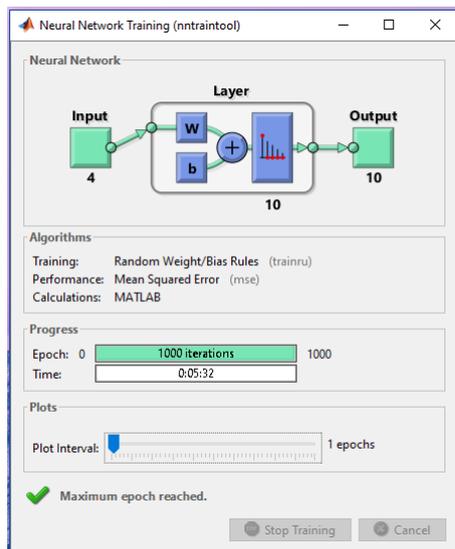
x' = data normalisasi

$x.max$ = data maksimum asli

$x.min$ = data minimum asli

C.5 Clustering dengan Algoritma SOM

Struktur Jaringan SOM yang digunakan pada penelitian ini ialah 4-10-10, yang berarti 4 *input* - 10 *neuron* - 10 *output*. Jumlah *cluster* ditentukan dengan jumlah *neuron*, yang berarti hasil pemrosesan *input* akan di-*cluster* dan ditransfer ke layer *ouput* menjadi 10 *cluster*. Jumlah *epoch* yang diset ialah 1000 *epoch* (nilai *default*).



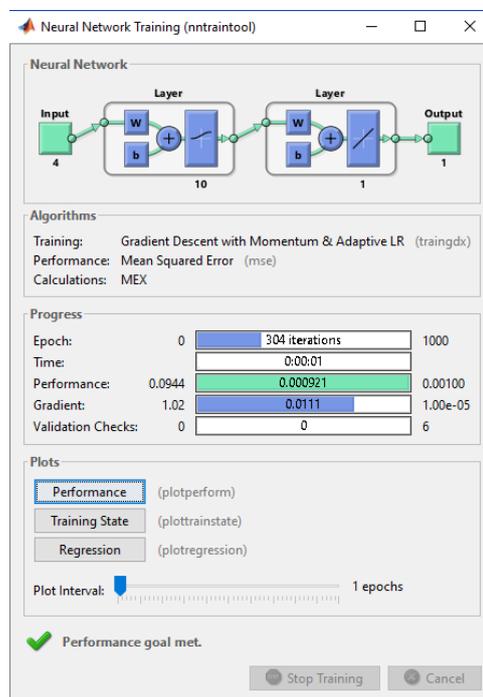
Gambar 6 Proses *Clustering* Menggunakan *Software* MATLAB

Gambar 6 memperlihatkan proses *clustering* dengan algoritma SOM pada *Software* MATLAB yang memiliki struktur 4-10-10 dengan 1000 *epoch* yang selesai dalam waktu 5 menit 32 detik. Output dari pelatihan ini berupa data yang telah memiliki *cluster* masing-masing, namun yang akan dihubungkan dengan algoritma *Backpropagation* bukanlah data hasil *cluster* tersebut, melainkan bobot dari hasil pembelajaran SOM. Bobot yang diambil di antara layer *input* dan node kemudian bobot tersebut disimpan ke dalam bentuk MAT-File untuk kemudian digunakan sebagai faktor pembimbing pada pembelajaran dengan supervisi *Backpropagation*.

C.6 Membuat dan Melatih Jaringan dengan Algoritma *Backpropagation*

Algoritma pelatihan dengan algoritma *Backpropagation* dibangun dengan struktur 4-10-10-1, yang berarti 4 *input* – 10 *neuron* pada layer 1 – 10 *neuron* pada layer 2 – dan 1 *output*. Fungsi aktivasi pada kedua layer tersebut menggunakan fungsi aktivasi yang lazim digunakan pada persoalan prediksi yaitu layer 1 menggunakan fungsi aktivasi *Sigmoid Biner* (Logsig) dan layer 2 menggunakan

fungsi aktivasi *Identitas* (Purelin). Algoritma pelatihan yang digunakan ialah *Gradient Descent with Momentum and Adaptive Learning Rate*. Proses training algoritma *Backpropagation* menggunakan *software* MATLAB dapat dilihat pada Gambar 7.



Gambar 7 Proses *Training* Menggunakan *Software* MATLAB

Gambar 7 memperlihatkan proses training dengan menggunakan *Software* MATLAB dengan struktur jaringan 4-10-1-1.

C.7 Pengujian Jaringan dan Prediksi

Setelah diperoleh korelasi yang terbaik dengan *error* yang minimal, dilakukan pengujian jaringan dengan data uji. Dari hasil pengujian akan diketahui korelasi antara data target uji dengan *output* yang dihasilkan oleh algoritma SOM-ANN menggunakan model yang telah dibentuk pada proses pelatihan. Jika korelasi masih tergolong cukup baik, selanjutnya dilakukan prediksi dengan memasukkan data debit Q_n untuk memperoleh data Q_{n+1} dengan *syntax*:

$$\text{Prediksi} = \text{sim}(a,b) \quad (27)$$

Keterangan:

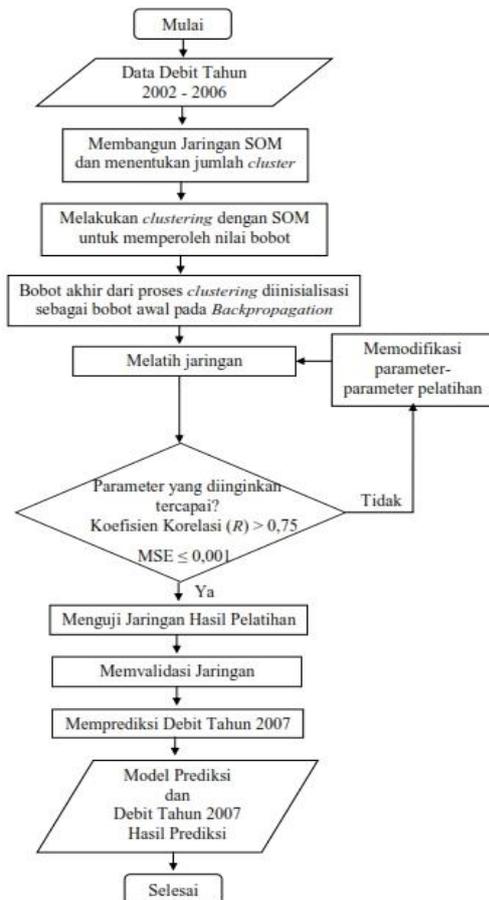
Sim = perintah untuk mensimulasikan jaringan yang telah terbentuk

a = Model jaringan yang telah dibentuk

b = Data debit hasil observasi

C.8 Bagan Alir Penelitian

Tahapan penelitian yang akan dilakukan dapat dilihat pada bagan alir penelitian seperti pada Gambar 8.



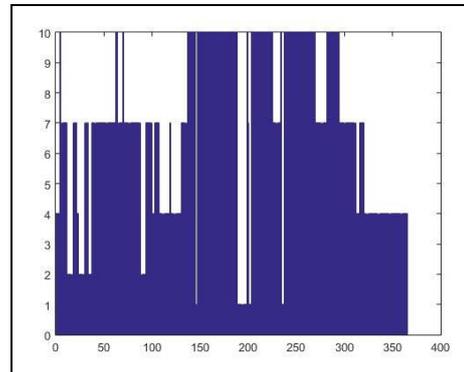
Gambar 8 Diagram Alir Penelitian

D. HASIL DAN PEMBAHASAN

D.1 Pembobotan Menggunakan SOM

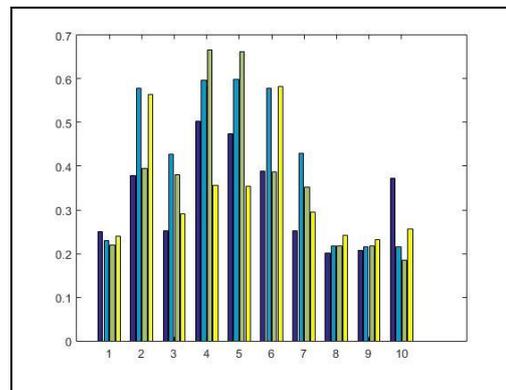
Klasifikasi pola dilakukan dengan tujuan untuk memperoleh kelompok pola (*cluster*) dan bobot pada setiap kelompok *input*. Setiap pola akan dimasukkan ke sebuah kelompok yang paling dekat atau mirip dengan *input* yang diberikan. Pada penelitian ini, data yang akan dijadikan *input* dan dicari pembobotannya ialah data debit Tahun 2002 sampai dengan Tahun 2005. Penentuan data yang akan di-*cluster*

ialah berdasarkan data yang juga akan dijadikan *input* pada algoritma *Backpropagation*. Hasil *cluster* menggunakan algoritma SOM dengan jumlah 10 *cluster* dapat dilihat pada Gambar 9.



Gambar 9 Hasil *Cluster* Menggunakan Algoritma SOM

Gambar 9 memperlihatkan pembagian data *input* menjadi 10 *cluster*. Pembagian bobot untuk masing-masing *cluster* dapat dilihat pada Gambar 10.



Gambar 10 Bobot Hasil Pembelajaran SOM

Bobot tersebut selanjutnya akan menjadi faktor pembimbing pada pelatihan jaringan dengan algoritma *Backpropagation*. *Syntax* yang digunakan untuk menginisialisasikan bobot akhir SOM menjadi bobot awal pada pembelajaran *Backpropagation* ialah:

$$net.IW \{1,1\} = bobot_akhir \quad (4.1)$$

Keterangan:

$Net.IW \{1,1\}$ = *Initial Weight* atau bobot inisial di bagian hubungan *input* menuju *layer 1 Backpropagation*

bobot_akhir = nilai bobot akhir hasil pembelajaran SOM

D.2 Kombinasi Konstanta Parameter

Untuk mencapai hasil yang optimal, maka dibutuhkan kombinasi konstanta parameter pelatihan yaitu *learning rate* dan momentum.

Tabel 1 Kombinasi Parameter Terbaik

Parameter	Kombinasi
Koefisien Korelasi	Mc = 0,9 ; Lr = 0,5
MSE	Mc = 0,9 ; Lr = 0,5
Epoch	Mc = 0,9 ; Lr = 0,5

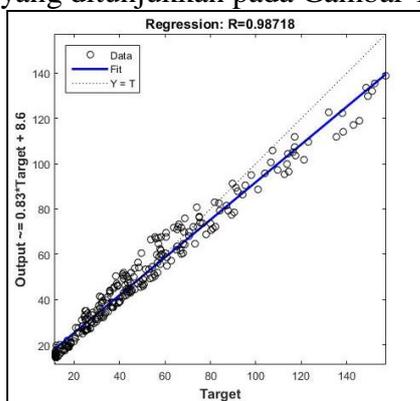
Berdasarkan nilai terbaik yang diperoleh maka digunakan kombinasi Mc = 0,9 dan Lr = 0,5 untuk pelatihan jaringan menggunakan Metode SOM-ANN.

D.3 Pelatihan Jaringan

Pelatihan jaringan melibatkan 70% data dengan produk yang akan dihasilkan ialah jaringan yang sudah terlatih dengan koefisien korelasi dan performa hasil pelatihan.

1. Koefisien Korelasi (*R*)

Nilai *R* hasil pelatihan dengan menggunakan metode SOM-ANN yang menghasilkan nilai *R* sebesar 0,98718 yang termasuk dalam kriteria korelasi sangat kuat yang ditunjukkan pada Gambar 11.

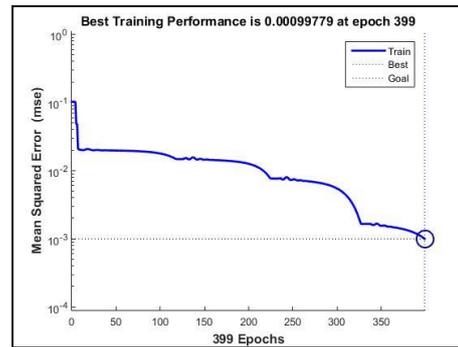


Gambar 11 Hubungan Pemodelan dan Observasi Hasil Pelatihan Metode SOM-ANN

2. Epoch

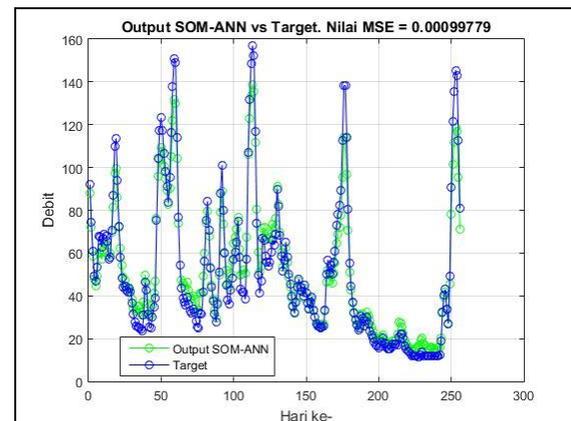
Untuk mengukur performa suatu pelatihan jaringan, dapat dilihat berdasarkan jumlah

Epoch yang dibutuhkan. Performa pelatihan menggunakan metode SOM-ANN dapat dilihat pada Gambar 12.



Gambar 12 Grafik Performa Hasil Pelatihan dengan Metode SOM-ANN

Gambar 12 merupakan grafik pelatihan dengan menggunakan Metode SOM-ANN yang berhasil mencapai *goal* pada *epoch* ke 399 dari 5000 *epoch* yang diberikan dengan nilai MSE = 0,00099779. Hal ini menunjukkan bahwa pelatihan dengan menggunakan Metode SOM-ANN jauh lebih cepat menuju status konvergen. Perbandingan antara target dan *output* dapat dilihat pada Gambar 13.

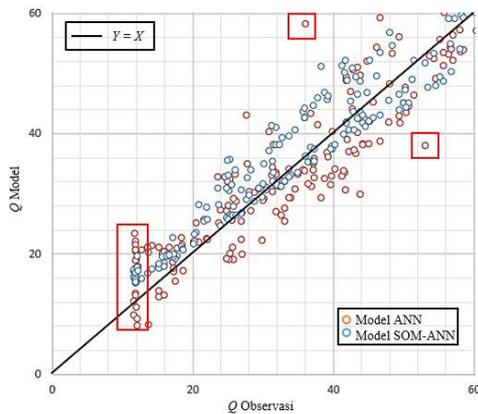


Gambar 13 Perbandingan *Output* dan Target pada Proses Pelatihan Metode SOM-ANN

3. Perbandingan Hasil Pelatihan Metode SOM-ANN dan ANN

Untuk mengetahui perbandingan hasil pelatihan antara Metode SOM-ANN dan

Metode ANN, hasil pemodelan kedua metode diplot ke dalam satu grafik sehingga dapat dilihat perbedaan nilai yang dihasilkan. Perbedaan tersebut dapat dilihat pada Gambar 14.

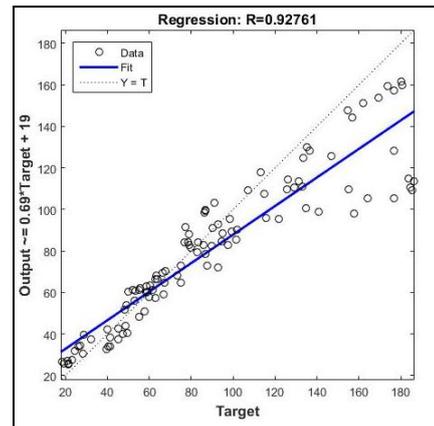


Gambar 14 Perbedaan Nilai Pemodelan

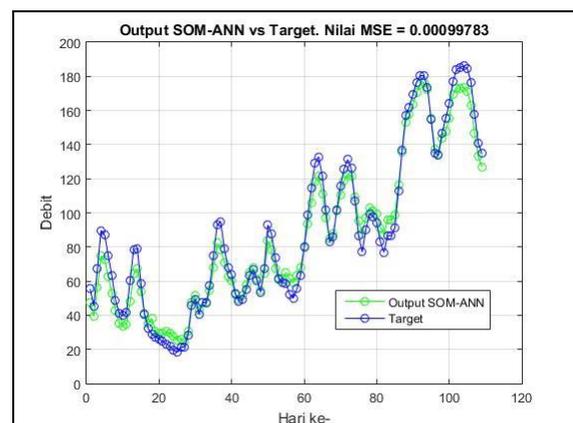
Debit hasil pemodelan SOM-ANN yang dihasilkan memperlihatkan pola hubungan yang lebih baik dengan debit observasi dibandingkan dengan debit hasil pemodelan ANN. Hal tersebut dapat dilihat dari nilai debit yang dihasilkan oleh SOM-ANN tersebar mengikuti garis linier, sedangkan debit hasil model ANN terutama pada *range* debit 0 – 20 memiliki nilai yang berbeda-beda untuk nilai debit observasi dengan nilai yang hampir sama. Pada *range* debit 20 – 60, jika dibandingkan dengan hasil pemodelan SOM-ANN, terdapat beberapa nilai debit pemodelan ANN yang memiliki perbedaan cukup jauh dari nilai debit observasi. Gambaran tersebut menunjukkan bahwa pemodelan ANN belum cukup handal dalam mendekati target yang diberikan.

D.4 Pengujian Jaringan

Jaringan yang telah terbentuk pada proses pelatihan kemudian diuji dengan data-data di luar data latih dengan porsi 30%. Nilai koefisien korelasi dan hasil pemodelan yang dihasilkan dengan Metode SOM-ANN berdasarkan hasil pengujian dapat dilihat pada Gambar 15 dan Gambar 16.



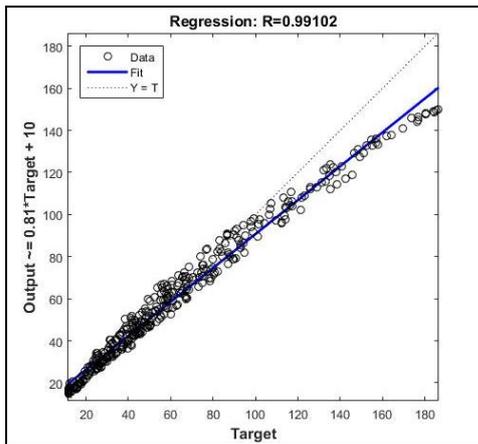
Gambar 15 Hubungan Pemodelan dan Observasi Hasil Pengujian Metode SOM-ANN



Gambar 16 Perbandingan *Output* dan Target pada Proses Pengujian Metode SOM-ANN

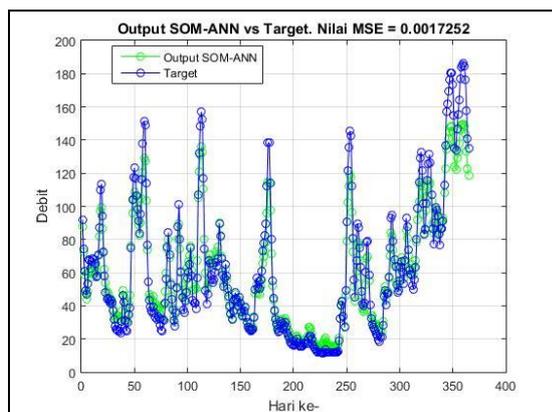
D.5 Validasi Jaringan

Jaringan yang telah terbentuk pada proses pelatihan dan kemudian diuji dengan data-data di luar data latih, berikutnya diuji dengan semua data latih dan uji dengan porsi 100%. Nilai koefisien korelasi yang dihasilkan dengan Metode SOM-ANN berdasarkan hasil validasi dapat dilihat pada Gambar 17.



Gambar 17 Hubungan Pemodelan dan Observasi Hasil Validasi dengan Metode SOM-ANN

Gambar 17 memperlihatkan nilai R hasil pelatihan dengan menggunakan Metode SOM-ANN yang menghasilkan nilai R sebesar 0,99102 yang masuk pada kategori korelasi sangat kuat, sedangkan hasil pemodelan proses validasi dapat dilihat pada Gambar 18.



Gambar 18 Perbandingan *Output* dan Target pada Proses Validasi Metode SOM-ANN

D.6 Rekapitulasi Perbandingan Performansi dan Akurasi

Perbandingan performansi dan akurasi dari Metode ANN dan Metode SOM-ANN secara keseluruhan, seperti yang dapat dilihat pada Tabel 2.

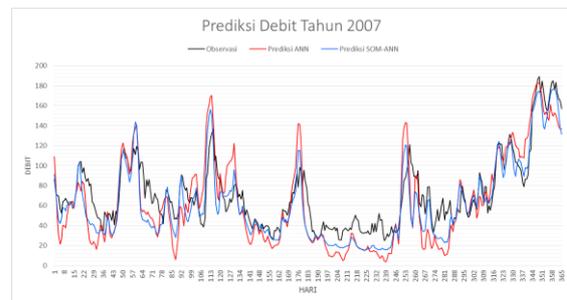
Tabel 2 Perbandingan Performansi dan Akurasi

Evaluasi	Pelatihan		Pengujian		Validasi	
	ANN	SOM-ANN	ANN	SOM-ANN	ANN	SOM-ANN
R	0,97667	0,98718	0,92761	0,99039	0,97555	0,99102
MSE	0,000999	0,000977	0,008693	0,000997	0,001777	0,001725
Epoch	4786	399	-	-	-	-

Berdasarkan nilai yang diperoleh dari berbagai evaluasi, dapat disimpulkan bahwa Metode SOM-ANN memiliki performansi dan akurasi yang lebih baik dibandingkan Metode ANN biasa sehingga jaringan yang dihasilkan oleh metode ini dapat digunakan untuk melakukan simulasi prediksi debit pada masa yang akan datang.

D.7 Hasil Prediksi

Hasil prediksi debit pada Tahun 2007 untuk masing-masing metode ditampilkan secara visual dalam bentuk grafik pada Gambar 19.



Gambar 19 Hasil Prediksi Debit Tahun 2007

Perbandingan hasil prediksi debit pada Metode ANN dan Metode SOM-ANN dapat dilihat dari perbedaan nilai *output* yang dihasilkan, yakni pada Metode SOM-ANN nilai *output* yang dihasilkan lebih dominan mendekati nilai observasi dibandingkan dengan nilai *output* pada Metode ANN. Metode SOM-ANN mampu meminimalisir kesalahan prediksi terutama pada titik-titik puncak debit sehingga mampu mendekati nilai debit observasi lebih baik daripada Metode ANN. Dengan demikian, dapat disimpulkan bahwa Metode SOM-ANN terbukti memiliki tingkat akurasi yang lebih baik dibanding Metode ANN biasa.

E. KESIMPULAN DAN SARAN

E.1 Kesimpulan

Berdasarkan hasil penelitian tugas akhir ini, dapat disimpulkan beberapa hal sebagai berikut.

1. Pemodelan menggunakan Metode ANN menghasilkan nilai $R=0,9755$, $MSE=0,001777$ dan membutuhkan 4786 *epoch* untuk menyelesaikan tahap pelatihan.
2. Pemodelan menggunakan Metode SOM-ANN menghasilkan nilai $R=0,99102$, $MSE=0,001725$ dan membutuhkan 399 *epoch* untuk menyelesaikan tahap pelatihan.
3. Metode SOM-ANN memiliki keunggulan dibanding Metode ANN biasa baik dari segi performansi maupun akurasi. Pada penelitian ini, akurasi yang dihasilkan dengan Metode SOM-ANN meningkat sebesar 44,74% dan performa pelatihan jaringan meningkat secara signifikan sebesar 1099%.
4. Model peramalan yang dibangun dengan Metode SOM-ANN mampu mereduksi kesalahan dalam peramalan sehingga lebih dapat dipercaya untuk dijadikan sebagai acuan pada *Early Warning System* mendekati nilai debit observasi.

E.2 Saran

Pada penelitian ini, penggunaan algoritma SOM hanya dibatasi pada penggunaan hasil pembobotan pada proses *clustering*, sehingga pada penelitian selanjutnya hendaknya dapat lebih mengeksplorasi kemampuan algoritma SOM dalam menyelesaikan permasalahan dalam bidang prediksi atau peramalan.. Selain itu, pemodelan prediksi sebaiknya dapat divisualisasikan dengan tampilan antar muka (*Graphical User Interface*) agar model prediksi yang telah dibentuk menjadi lebih mudah untuk digunakan.

DAFTAR PUSTAKA

- Ade Gafar Abdullah (2010). *Buku Ajar Pengantar Kecerdasan Buatan*. Universitas Pendidikan Indonesia. Bandung. Tersedia di [http://file.upi.edu/Direktori/fptk/jur._pend._teknik_elektro/197211131999031_ade_gafar_abdullah/file_mk_pengantar_kecerdasan_buatan_\(9files\)/bab_iv_kcb.pdf](http://file.upi.edu/Direktori/fptk/jur._pend._teknik_elektro/197211131999031_ade_gafar_abdullah/file_mk_pengantar_kecerdasan_buatan_(9files)/bab_iv_kcb.pdf), diakses pada 2 November 2016, Pukul 9:49 WIB.
- Asdak, C. (1995). *Hidrologi dan Pengolahan Daerah Aliran Sungai*. Yogyakarta: Gadjah Mada University Press.
- Dewan SDA Nasional (2013). *Pola Pengelolaan Sumber Daya Air Wilayah Sungai Siak Stranas 2013*. Jakarta.
- Farias, C. A. S. (2015). *Runoff-Erosion Modelling at Micro-Watershed Scale: a Comparison of Self-Organizing Maps Structures*. *Jurnal Geoenvironmental Disasters* 2:14
- Fausett, L. (1994). *Fundamentals of Neural Network; Architecture, Algorithms and Applications*. Prentice Hall.
- Goel, A. (2011). *ANN-Based Approach for Predicting Rating Curve of an Indian River*. *International Scholarly Research Network ISRN Civil Engineering*, Volume 2011, Article ID 291370, 4 pages doi:10.5402/2011/291370.
- Harto, S. (1993). *Analisis Hidrologi*. Jakarta: Nafiri.
- Hermawan, A. (2006). *Jaringan Syaraf Tiruan Teori dan Aplikasinya*. Yogyakarta: Andi Offset.
- Isnanto, R. R. (2014). *Penerapan Metode Self-Organizing Map (SOM) untuk Visualisasi Data Geospasial pada Informasi Sebaran Data Pemilih Tetap (DPT)*. *Jurnal Sistem Informasi Bisnis* 01.
- Jonathan Sarwono (2008). *Teori Analisis Korelasi*. <http://www.jonathansarwono.info/ko>

- relasi//korelasi.htm, diakses pada 28 Juli 2016, Pukul 05:30 WIB.
- Kristanto, A. (2004). *Jaringan Syaraf Tiruan (Konsep Dasar, Algoritma dan Aplikasi)*. Yogyakarta: Gava Media.
- Kusumadewi, S. (2003). *Artificial Intellegence (Teknik dan Aplikasinya)*. Yogyakarta: Graha Ilmu.
- Kusumadewi, S. (2004). *Membangun Jaringan Syaraf Tiruan Menggunakan MATLAB dan EXCEL LINK*. Yogyakarta: Graha Ilmu.
- Mahyudin (2013). *Model Prediki Liku Kalibrasi Menggunakan Pendekatan Jaringan Saraf Tiruan (JST)*. Tugas Akhir Jurusan Teknik Sipil. Universitas Riau. Pekanbaru.
- Nur Anita Setyawati (2009). *Das Sungai Siak Provinsi Riau*. Blog UI. Tersedia di: https://staff.blog.ui.ac.id/tarsoen.waryono/files/2009/12/das_siak_nuranita_setyawati_0706265705.pdf, diakses pada 6 Oktober 2016, Pukul 21.44 WIB.
- Puspaningrum, D. (2006). *Pengantar Jaringan Syaraf Tiruan*. Yogyakarta : Andi Offset.
- Russell, S. & Norvig, P. (2003). *Artificial Intellegence a Modern Approach*. New Jersey: Person Education, Inc.
- Salim R. R. M. & Jauhari A. S. (2016). *Perancangan Pengenalan Karakter Alfabet Menggunakan Metode SOM-ANN Jaringan Syaraf Tiruan*. Skripsi Sarjana, Fakultas Teknik, Jurusan Sistem Informasi, STMIK Mikroskil, Medan.
- Siang, J. J. (2009). *Jaringan Syaraf Tiruan dan Pemrogramannya Menggunakan MATLAB (Ed. II)*. Yogyakarta: Andi Offset.
- Sutojo, dkk. (2011). *Kecerdasan Buatan*. Yogyakarta: Andi Offset.