



## Prediksi Volume Penggunaan Air PDAM Menggunakan Metode Jaringan Syaraf Tiruan Backpropagation

Budy Satria

Teknik Komputer - AMIK Mitra Gama, budysatriadeveloper@gmail.com

### Abstract

*As the population growth rate in Duri increases, the need for clean water also increases as needed. In Indonesia, PDAM is an institution that regulates and manages the provision of clean water for the community. So the amount of water produced and distributed should be adjusted to the demand for water. However, the problem arises in the form of waste of water at PT. PDAM Duri. Purpose of this study is to predict the amount of water consumption at PT. PDAM Duri by implementing Backpropagation Artificial Neural Network method. Variables of data taken from customer data were social, general social, household 1, household 2, household 3, commerce 1, commerce 2 and commerce 3. Data used in the prediction process was training data in 2016 and data testing in 2017. Actual amount of data at PT. PDAM Duri City 2016 until 2017 was 2.840.165 when the prediction result using artificial neural network back propagation method was 2.843.388. The number of training epochs was 4595 and the achievement of MSE (Mean Squared Error) on the test was 0,001 and the result of accuracy was 99,99900000%. Final result of this research was artificial neural network using back propagation method could predict the using of water consumption at PT. PDAM Duri for next year.*

*Keywords: Backpropagation, Artificial Intelligence, Prediction, PT. PDAM Duri, Neural Network Method.*

### Abstrak

Seiring dengan meningkatnya laju pertumbuhan penduduk di kota Duri, kebutuhan akan air bersih pun meningkat sesuai kebutuhan. Di Indonesia, PDAM merupakan lembaga yang mengatur dan mengelola pengadaan air bersih bagi masyarakat. Jadi jumlah air yang diproduksi dan disalurkan sebaiknya disesuaikan dengan permintaan air. Namun, timbul persoalan berupa pemborosan air di PT. PDAM Kota Duri. Tujuan penelitian ini adalah melakukan prediksi jumlah penggunaan konsumsi air di PT. PDAM Kota Duri dengan menggunakan metode Jaringan Syaraf Tiruan *Backpropagation*. Variabel data diambil dari data pelanggan yaitu sosial umum, sosial khusus, rumah tangga 1, rumah tangga 2, rumah tangga 3, niaga 1, niaga 2 dan niaga 3. Data yang digunakan dalam proses prediksi adalah data pelatihan tahun 2016 dan data pengujian tahun 2017. Total data aktual pada PT. PDAM Kota Duri tahun 2016 s/d 2017 adalah 2.840.165 sedangkan hasil perhitungan prediksi menggunakan Jaringan Syaraf Tiruan Metode *Backpropagation* adalah 2,843,388. Jumlah *epoch* pelatihan adalah 4595 dan pencapaian *MSE (Mean Squared Error)* pada pengujian adalah 0,001 serta hasil akurasi yang didapatkan adalah 99,99900000%. Hasil akhir dari penelitian ini adalah jaringan syaraf tiruan menggunakan metode *backpropagation* dapat memprediksi penggunaan konsumsi air di PT. PDAM Kota Duri untuk tahun selanjutnya.

*Kata kunci: Propagasi Balik, Kecerdasan Buatan, Prediksi, PT. PDAM Duri, Metode Jaringan Syaraf Tiruan.*

© 2018 Jurnal RESTI

### 1. Pendahuluan

Air sebagai salah satu kebutuhan utama untuk menunjang kehidupan manusia dan merupakan salah satu sumber daya alam yang memiliki fungsi sangat penting serta merupakan komponen lingkungan hidup yang bermanfaat bagi kelangsungan hidup manusia dan makhluk hidup lainnya. Seiring dengan meningkatnya laju pertumbuhan penduduk di kota Duri, kebutuhan akan air bersih pun meningkat sesuai kebutuhan. Di Indonesia, PDAM merupakan lembaga yang mengatur dan mengelola pengadaan air bersih bagi masyarakat. Jadi jumlah air yang diproduksi dan disalurkan

sebaiknya disesuaikan dengan permintaan air. Namun, timbul persoalan berupa pemborosan air di PT. PDAM Kota Duri. Ada permasalahan dalam menghadapi kebutuhan air, salah satunya apabila jumlah air yang di produksi dan disalurkan lebih besar daripada permintaan akan air, maka timbul persoalan pemborosan volume air pada PDAM. Dalam hal ini diperlukan sebuah prediksi untuk mengetahui kebutuhan manajemen produksi jumlah air PDAM.

Pada penelitian ini prediksi menggunakan teknik kecerdasan buatan dengan metode Jaringan Syaraf Tiruan (JST). Konsep dari JST untuk memprediksi

jumlah produksi air PDAM dengan cara mengolah data jumlah produksi air PDAM periode masa lalu yang dimasukkan kedalam sistem dilakukan proses pelatihan menggunakan JST dengan metode *Backpropagation*. Setelah dilakukan proses pelatihan, sistem akan menghasilkan bobot-bobot yang akan digunakan untuk memprediksi jumlah produksi air PDAM pada periode tahun-tahun selanjutnya. Kecerdasan buatan (*Artificial Intelligence*) merupakan cabang ilmu komputer yang konsern dengan pengautomatisan tingkah laku [1].

Jaringan syaraf tiruan telah banyak digunakan dalam berbagai permasalahan diantaranya pada penelitian sebelumnya tentang platform jaringan syaraf tiruan *feedforward digital* yang menggunakan algoritma *backpropagation* selama pelatihan untuk memungkinkan putaran energi [2]. Jaringan syaraf tiruan juga bisa digunakan untuk peramalan beban listrik [3]. Jaringan syaraf tiruan digunakan untuk peramalan kunjungan rawat jalan menggunakan dekomposisi mode empiris ditambah dengan jaringan syaraf tiruan *backpropagation* [4]. Jaringan syaraf tiruan juga digunakan untuk prediksi deret waktu muncul di banyak masalah dunia nyata, misalnya, pasar keuangan, pemrosesan sinyal, peramalan cuaca antara lain [5] dan juga untuk peramalan yang mengintegrasikan transformasi wavelet diskrit dan jaringan saraf *backpropagation* untuk memprediksi deret waktu keuangan [6].

## 2. Tinjauan Pustaka

### 2.1 Kecerdasan Buatan

Kecerdasan buatan atau (*Artificial Intelligence*) merupakan bidang ilmu komputer yang membuat agar komputer dapat melakukan pekerjaan seperti dan sebaik yang dilakukan oleh manusia [7]. Teknologi kecerdasan buatan dipelajari dalam beberapa bidang, seperti : Robotika dan Sistem Sensor, Penglihatan Komputer (*Computer Vision*), Sistem Saraf Tiruan (*Artificial Neural System*), Agen Cerdas (*Intelligent Agent*), Pengenalan Suara (*Speech Recognition*), Permainan (*Game Playing*) dan Sistem Pakar (*Expert System*). [8]

### 2.2 Definisi Jaringan Syaraf Tiruan

Jaringan syaraf tiruan merupakan salah satu metode berbasis kecerdasan buatan, yang mampu mengidentifikasi pola, signal prosesing serta peramalan dari sistem dengan metode pembelajaran [9]. Dalam jaringan syaraf tiruan terdapat konsep yakni setiap pola-pola informasi input dan output yang diberikan ke dalam JST diproses dalam neuron. Neuron-neuron tersebut terkumpul di dalam lapisan-lapisan yang disebut neuron layers [10]. Lapisan-lapisan penyusun JST tersebut dapat dibagi menjadi 3, yaitu :

1. Lapisan Input, unit-unit di dalam lapisan input disebut unit-unit input. Unit-unit input tersebut menerima pola inputan data dari luar yang menggambarkan suatu permasalahan.
2. Lapisan tersembunyi, unit - unit didalam lapisan tersembunyi disebut unit –unit tersembunyi.
3. Lapisan *output*, unit – unit didalam lapisan *output* disebut unit - unit *output*.

### 2.3 Definisi Prediksi

Dalam penelitian ini dibutuhkan beberapa teori yang berhubungan dengan prediksi diantaranya adalah : Prediksi (*forecasting*) adalah suatu kegiatan yang memperkirakan apa yang terjadi pada masa akan datang. Masalah pengambilan keputusan merupakan masalah yang dihadapi maka peramalan juga merupakan masalah yang harus dihadapi, karena peramalan berkaitan erat dengan pengambilan suatu keputusan [11].

### 2.4 *Backpropagation*

*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 tersembunyinya[12]. Secara garis besar, training jaringan dengan dengan metode *backpropagation* meliputi tiga tahap [13]:

#### a. Tahap maju (*feedforward*)

Tahap *feedforward* yang dimaksud adalah proses pengolahan input dari pola input training pada *input layer* sampai respon yang dihasilkan mencapai *output layer*

#### b. Tahap perhitungan *error* propagasi balik (*backpropagation of error*)

Respon yang dihasilkan pada *output layer* akan dibandingkan dengan *output target*, kemudian dihitung errornya. Bila kriteria untuk kondisi berhenti (*stopping condition*) belum terpenuhi, maka dilanjutkan ke tahap ketiga (*adjustment of the weights and biases*). Namun jika kondisi berhenti sudah terpenuhi, maka proses perhitungan berhenti.

#### c. Tahap pembaharuan bobot dan bias (*adjustment of the weights and biases*)

Kondisi ini terjadi jika *output* yang diharapkan tidak sesuai, maka jaringan akan bergerak mundur (*backward*) dari *output layer* menuju ke *input layer* dan akan melakukan *update* bobot dan bisa serta mengulangi proses dari tahap 1. *Backpropagation* merupakan algoritma pembelajaran yang biasanya digunakan oleh *perceptron* dengan banyak lapisan untuk mengubah bobot-bobot yang terhubung dengan neuron-neuron yang ada pada lapisan tersembunyinya [14].

Algoritma pembelajaran untuk JST *Backpropagation* adalah sebagai berikut :

**Langkah 0** : Inialisasi bobot dengan nilai random atau acak yang cukup kecil.

**Langkah 1** : Selama kondisi berhenti belum terpenuhi, laksanakan langkah sampai 9.

**Langkah 2** : Untuk tiap pasang pelatihan, kerjakan langkah 3 sampai 8

**Langkah 3** : Untuk tiap *input* neuron ( $X_i, i=1,2,3,\dots,n$ ) menerima *input*  $x_i$  dan menyebarkan sinyal tersebut ke seluruh neuron kepada lapisan atasnya (lapisan tersembunyi).

**Langkah 4** : Untuk *hidden* neuron ( $Z_i, j=1,2,3,\dots,p$ ) dihitung nilai input dengan menggunakan nilai bobotnya:

$$z\_in_j = V_{oj} + \sum_{i=1}^n X_i V_{ij} \quad (1)$$

Kemudian dihitung nilai output dengan menggunakan fungsi aktivasi yang digunakan:

$$z_j = f(z\_in_j) \quad (2)$$

Di mana fungsi aktivasi yang digunakan adalah fungsi *sigmoid biner* yang mempunyai persamaan:

$$f(x) = \frac{1}{1+\delta^x} \quad (3)$$

**Langkah 5** : Untuk tiap output neuron ( $Y_k, k=1,2,3,\dots,m$ ) dihitung nilai input dengan nilai bobotnya

$$\sum_{i=1}^p Z_i W_{jk} \quad (4)$$

Kemudian dihitung nilai *output* dengan menggunakan fungsi aktivasi:  $y_k = f(y\_ink)$

**Langkah 6** : Untuk tiap *output* neuron ( $Y_k, k=1,2,3,\dots,m$ ) menerima pola target yang bersesuaian dengan pola input dan kemudian dihitung informasi kesalahan:  $\Delta k = (t_k - y_k) f'(y\_ink)$  (5)

Kemudian dihitung koreksi nilai bobot yang kemudian akan digunakan untuk memperbaharui nilai  $\Delta w_{jk}$  :  $w_{jk} = \alpha \delta_k z_j$

Hitung koreksi nilai bias yang kemudian akan digunakan untuk memperbaharui nilai  $w_{0k}$  :  $\Delta w_{0k} = \alpha \delta_k$  Dan kemudian nilai  $\delta_k$  dikirim ke neuron pada lapisan sebelumnya.

**Langkah 7** : Untuk tiap *hidden neuron* ( $Z_j, j=1,2,3,\dots,p$ )

$$\delta\_in_j = \sum_{k=1}^m \delta_k W_{jk} \quad (6)$$

Kemudian nilai tersebut dikalikan dengan nilai turunan dari fungsi aktivasi untuk menghitung informasi kesalahan:  $\delta_j = \delta\_in_j f'(z\_in_j)$

Hitung koreksi nilai bobot yang kemudian digunakan untuk memperbaharui  $v_{ij}$  :

$$\Delta v_{ij} = \alpha \delta_j x_i$$

Dan hitung nilai koreksi bias yang kemudian digunakan untuk memperbaharui nilai

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

Memperbaharui nilai bobot dan nilai bias:

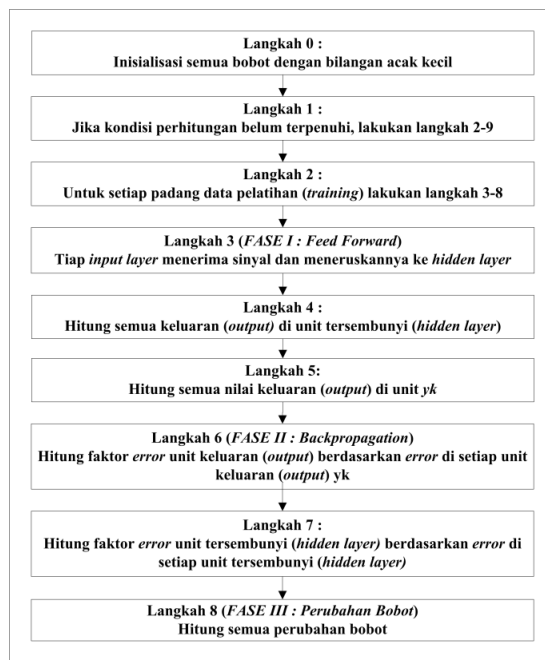
**Langkah 8** : Tiap nilai bias dan bobot ( $j=0,\dots,p$ ) pada output neuron ( $Y_k, k=1,2,3,\dots,m$ ) diperbaharui:

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

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

### 3. Metodologi Penelitian

Penelitian ini dilakukan di PT. PDAM Kota Duri, pengumpulan data dan informasi digunakan pendekatan deskriptif atau survey yaitu mengumpulkan data dari data pelanggan yaitu sosial umum, sosial khusus, rumah tangga 1, rumah tangga 2, rumah tangga 3, niaga 1, niaga 2 dan niaga 3 dari tahun 2016-2017. Data tersebut kemudian dianalisis dan digunakan sebagai data pelatihan (*training*) dan data pengujian (*testing*). Algoritma Jaringan Syaraf Tiruan *Backpropagation* untuk memprediksi volume penggunaan air di PT. PDAM Kota Duri adalah sebagai berikut :



Gambar 1: Alur *Backpropagation*

### Hasil dan Pembahasan

#### 4.1 Analisa Data

Pada tahapan ini, data yang telah diperoleh dari PT. PDAM Kota Duri akan dibagi menjadi 2 kelompok yaitu data pelatihan (*training*) dan data pengujian (*testing*) pada Jaringan Syaraf Tiruan (JST). Data yang dibutuhkan adalah data dari PT. PDAM Kota Duri dari tahun 2016 (pelatihan) dan 2017 (pengujian)

Tabel 1. Data Penggunaan Air di PT. PDAM Kota Duri Tahun 2016/2017

Pelanggan Air Bersih PT. PDAM Kota Duri										
Tahun	Bulan	Sosial Umum M <sup>3</sup>	Sosial Khusus M <sup>3</sup>	Rumah Tangga 1 M <sup>3</sup>	Rumah Tangga 2 M <sup>3</sup>	Rumah Tangga 3 M <sup>3</sup>	Niaga 1 M <sup>3</sup>	Niaga 2 M <sup>3</sup>	Niaga 3 M <sup>3</sup>	Jumlah Air M <sup>3</sup>
2016	Jan	4.044	1.001	19.244	103.347	2.640	2.293	10.169	644	143.382
	Feb	3.175	982	17.855	90.029	2.298	1.895	8.941	532	125.707
	Mar	2.418	835	14.742	76.744	2.219	1.629	7.580	474	106.641
	Apr	3.361	1.023	19.653	106.867	2.890	2.278	9.775	459	146.306
	Mei	3.449	916	1.823	95.898	2.484	1.827	8.931	428	132.163
	Jun	3.277	669	16.158	84.361	2.041	1.600	8.544	1.156	117.806
	Jul	2.949	666	17.309	91.295	2.271	1.767	9.233	1.318	126.808
	Aug	2.641	1.068	17.937	93.900	2.219	1.909	9.539	1.177	130.390
	Sept	2.403	865	1.527	79.447	2.012	1.499	7.993	712	110.201
	Okt	2.865	1.017	17.581	91.133	2.213	1.665	8.615	1.205	126.294
	Nov	2.777	1.043	17.138	91.159	2.410	1.665	9.063	103	126.285
	Des	2.100	1.892	12.807	64.796	1.645	1.269	6.343	882	91.734
2017	Jan	2.380	2.856	18.487	94.359	2.252	1.933	10.450	710	133.427
	Feb	2.852	2.285	16.243	70.837	2.321	1.669	8.980	731	105.918
	Mar	2.710	2.716	14.789	77.271	1.734	1.737	9.345	656	110.958
	Apr	2.836	4.140	16.848	88.322	3.837	1.698	8.866	708	127.255
	Mei	2.859	3.021	17.023	88.001	5.444	2.275	8.693	1.322	128.638
	Jun	2.699	3.146	12.099	66.856	4.504	1.975	6.038	1.218	98.535
	Jul	2.907	3.950	13.564	81.069	5.410	2.133	7.524	527	117.084
	Aug	3.071	4.095	13.247	81.779	6.571	2.538	7.427	370	119.098
	Sept	3.072	4.049	11.267	83.687	7.694	3.159	8.036	296	121.260
	Okt	2.418	3.191	7.560	64.950	7.681	2.890	6.874	266	95.830
	Nov	2.182	1.280	7.146	69.607	9.248	2.945	6.210	391	99.009
	Des	2.287	1.197	6.484	68.749	9.677	3.326	7.232	484	99.436

Sumber: PT. PDAM Kota Duri Tahun 2016 s/d 2017

Berikut merupakan variabel- variabel yang akan *ditransformasikan* ke *interval* yang lebih kecil, digunakan dalam memprediksi jumlah penggunaan air misalnya pada *interval* [0.1, 0.8] adalah: bersih di PT. PDAM Kota Duri:

1. Sosial Umum
2. Sosial Khusus
3. Rumah Tangga 1
4. Rumah Tangga 2
5. Rumah Tangga 3
6. Niaga 1
7. Niaga 2
8. Niaga 3
9. Jumlah Penggunaan air (ditetapkan sebagai data Target).

Fungsi *aktivasi* yang digunakan untuk mengolah data pelatihan (*training*) dan pengujian (*testing*) di atas adalah fungsi *sigmoid* (biner), maka data harus *ditransformasikan* terlebih dahulu karena *range* (batas) keluaran fungsi *aktivasi sigmoid* adalah fungsi *asimtotik* (tidak pernah mencapai 0 ataupun 1). Data

$$X = \frac{0,8(x-a)}{b-a} + 0,1$$

Berikut merupakan variabel- variabel yang akan digunakan dalam memprediksi jumlah penggunaan air bersih di PT. PDAM Kota Duri:

1. Sosial Umum
2. Sosial Khusus
3. Rumah Tangga 1
4. Rumah Tangga 2
5. Rumah Tangga 3
6. Niaga 1
7. Niaga 2
8. Niaga 3
9. Jumlah Penggunaan air (ditetapkan sebagai data Target).

Fungsi *aktivasi* yang digunakan untuk mengolah data pelatihan (*training*) dan pengujian (*testing*) di atas adalah fungsi *sigmoid* (biner), maka data harus ditransformasikan terlebih dahulu karena *range* (batas) keluaran fungsi *aktivasi sigmoid* adalah fungsi *asimtotik* (tidak pernah mencapai 0 ataupun 1). Data ditransformasikan ke *interval* yang lebih kecil, misalnya pada *interval* [0.1, 0.8] adalah:

$$X = \frac{0,8(x-a)}{b-a} + 0,1$$

Keterangan :

0.8 = Ketetapan

X = Nilai data ke -n

A = Nilai data terendah (data terendah dari setiap input)

B = Nilai data tertinggi (data tertinggi dari setiap input)

Berikut merupakan data variabel dan inisialisasi data transformasi data dengan mengambil sampel data pada bulan januari pada setiap variabel :

Tabel 2. Pendataan Variabel dan Inisialisasi Data

No	Variabel	Inisialisasi	Nilai
1	Sosial	X1	A = 2100
	Umum		B = 4044
2	Sosial	X2	A = 666
	Khusus		B = 4140
3	Rumah	X3	A = 6484
	Tangga 1		B = 19653
4	Rumah	X4	A = 64796
	Tangga 2		B = 106867
5	Rumah	X5	A = 1645
	Tangga 3		B = 9677
6	Niaga 1	X6	A = 1269
			B = 3326
7	Niaga 2	X7	A = 6038
			B = 10450
8	Niaga 3	X8	A = 266
			B = 1322
9	Target	T	A = 91734
			B = 146306

Berikut merupakan sampel cara perhitungan transformasi data :

- Variabel pada Sosial Umum (X1)  
 $X1 = (0,8 * (4044 - 2100)/(4044 - 2100)) + 0,1 = 0,9000$
- Variabel pada Sosial Khusus (X2)  
 $X1 = (0,8 * (1001 - 666)/(4140 - 666)) + 0,1 = 0,1771$
- Variabel Rumah Tangga 1 (X3)

$$X2 = (0,8 * (19244 - 6484)/(19653 - 6484)) + 0,1 = 0,8752$$

$$4. \text{ Variabel Rumah Tangga 2 (X4)} \\ X3 = (0,8 * (103347 - 64796)/(106867 - 64796)) + 0,1 = 0,8331$$

$$5. \text{ Variabel Rumah Tangga 3 (X5)} \\ X4 = 0,8 * (2640 - 1645)/(9677 - 1645) + 0,1 = 0,1991$$

$$6. \text{ Variabel Niaga 1 (X6)} \\ X5 = (0,8 * (2293 - 1269)/(3326 - 1269)) + 0,1 = 0,4982$$

$$7. \text{ Variabel Niaga 2 (X7)} \\ X5 = (0,8 * (10169 - 6038)/(10450 - 6038)) + 0,1 = 0,8490$$

$$8. \text{ Variabel Niaga 3 (X8)} \\ X5 = (0,8 * (644 - 266)/(1322 - 266) + 0,1 = 0,3864$$

$$9. \text{ Variabel Penggunaan Air (Target)} \\ T = (0,8 * (143382 - 91734)/(146306 - 91734)) + 0,1 = 0,8571$$

Tabel 3 adalah hasil transformasi dari data pelatihan (*training*) tahun 2016 dan data pengujian (*testing*) tahun 2017 yang nantinya digunakan sebagai data untuk pengolahan menggunakan algoritma *backpropagation* serta tabel nilai bobot dengan inisial (V) dari *input layer* (lapisan masukan) ke *hidden layer* (lapisan tersembunyi) didapat dari hasil pengolahan di aplikasi *Matlab R2011*.

Langkah – langkah Perhitungan sebagai berikut :  
 Langkah 0 : Inisialisasi semua bobot dengan bilangan acak kecil.

**Langkah 1** : Jika kondisi perhitungan belum terpenuhi, lakukan langkah 2-9.

**Langkah 2** : Untuk setiap pasang data pelatihan, lakukan 3-8.

**Fase I : Propagasi Maju (feed forward)**

**Langkah 3** : Tiap *input layer* menerima sinyal dan meneruskannya ke *hidden layer* di atasnya.

**Langkah 4** : Hitung semua keluaran (*output*) di unit tersembunyi (*hidden layer*)  $Z_j$  ( $j= 1,2,\dots,p$ )

$$Z_{\_netj} = \left[ v_{jo} \sum_{i=1}^n x_i v_{ji} \right]$$

$$Z_{\_net 1} = -9,0218 + (0,9000 * 4.0893) + (0,1771 * -12.6059) + (0,8572 * 3.8926) + (0,8331 * 2.6772) + (0,1991 * -5.1347) + (0,4982 * -5.8459) + (0,8490 * 3.7648) + (0,3864 * 5.2527) = -0,6467$$

$$Z_{\_net 2} = 3,1878 + (0,9000 * 3.8117) + (0,1771 * 12.5258) + (0,8572 * -8.3892) + (0,8331 * -4.3976) + (0,1991 * -7.1776) + (0,4982 * 3.3908) + (0,8490 * -2.6528) + (0,3864 * -2.7386) = -5,2182$$

$$Z_{\_net 3} = -9.6652 + (0,9000 * -4.1231) + (0,1771 * 14.3192) + (0,8572 * -7.4235) + (0,8331 * -2.4658) + (0,1991 * 18.9344) + (0,4982 * 3.4449) + (0,8490 * 0.0759) + (0,3864 * 4.1219) = 0,7459$$

Tabel 3. Data Hasil Transformasi Pola Pelatihan (*training*) dan Pengujian (*testing*) Tahun 2016/2017

Pelanggan Air Bersih PT. PDAM Kota Duri										
Tahun	Bulan	X1	X2	X3	X4	X5	X6	X7	X8	T
2016	Jan	0.9000	0.1771	0.8752	0.8331	0.1991	0.4982	0.8490	0.3864	0.8571
	Feb	0.5424	0.1728	0.7908	0.5798	0.1650	0.3435	0.6264	0.3015	0.5980
	Mar	0.2309	0.1389	0.6017	0.3272	0.1572	0.2400	0.3796	0.2576	0.3185
	Apr	0.6189	0.1822	0.9000	0.9000	0.2240	0.4924	0.7776	0.2462	0.9000
	Mei	0.6551	0.1576	0.8136	0.6914	0.1836	0.3170	0.6246	0.2227	0.6927
	Jun	0.5844	0.1007	0.6877	0.4720	0.1394	0.2287	0.5544	0.7742	0.4822
	Jul	0.4494	0.1000	0.7576	0.6039	0.1624	0.2937	0.6793	0.8970	0.6142
	Aug	0.3226	0.1926	0.7958	0.6534	0.1572	0.3489	0.7348	0.7902	0.6667
	Sept	0.2247	0.1458	0.6337	0.3786	0.1366	0.1895	0.4545	0.4379	0.3707
	Okt	0.4148	0.1808	0.7741	0.6008	0.1566	0.2540	0.5673	0.8114	0.6066
	Nov	0.3786	0.1868	0.7472	0.6013	0.1762	0.2540	0.6485	0.6788	0.6065
	Des	0.1000	0.3823	0.4841	0.1000	0.1000	0.1000	0.1553	0.5667	0.1000
2017	Jan	0.2152	0.6043	0.8292	0.6622	0.1605	0.3582	0.9000	0.4364	0.7112
	Feb	0.4095	0.4728	0.6928	0.2149	0.1673	0.2556	0.6335	0.4523	0.3079
	Mar	0.3510	0.5721	0.6045	0.3372	0.1089	0.2820	0.6996	0.3955	0.3818
	Apr	0.4029	0.9000	0.7296	0.5474	0.3183	0.2668	0.6128	0.4348	0.6207
	Mei	0.4123	0.6423	0.7402	0.5413	0.4784	0.4912	0.5814	0.9000	0.6410
	Jun	0.3465	0.6711	0.4411	0.1392	0.3848	0.3746	0.1000	0.8212	0.1997
	Jul	0.4321	0.8562	0.5301	0.4094	0.4750	0.4360	0.3694	0.2977	0.4716
	Aug	0.4996	0.8896	0.5108	0.4229	0.5906	0.5935	0.3519	0.1788	0.5011
	Sept	0.5000	0.8790	0.3906	0.4592	0.7025	0.8351	0.4623	0.1227	0.5328
	Okt	0.2309	0.6815	0.1654	0.1029	0.7012	0.7304	0.2516	0.1000	0.1600
	Nov	0.1337	0.2414	0.1402	0.1915	0.8573	0.7518	0.1312	0.1947	0.2066
	Des	0.1770	0.2223	0.1000	0.1752	0.9000	0.9000	0.3165	0.2652	0.2129

Tabel 4. Nilai Bobot Awal *Input Layer* ke *Hidden Layer* dari Matlab R2011

Bobot	Z1	Z2	Z3	Z4	Z5	Z6	Z7	Z8	Z9
V1	4.0893	3.8117	-4.1231	4.1327	1.3154	-4.5069	-3.0457	0.6456	5.0714
V2	-12.6059	12.5258	14.3192	-0.4143	8.4562	-11.3636	-3.0484	16.2216	9.1783
V3	3.8926	-8.3892	7.4235	8.3462	3.4169	5.5521	6.4309	-2.8547	3.3153
V4	2.6772	-4.3976	-2.4658	-4.5372	-4.0038	3.6222	2.6789	-2.5186	4.9907
V5	-5.1347	-7.1776	18.9344	19.0398	-20.0764	-0.7395	-4.8270	12.9980	14.9724
V6	-5.8459	3.3908	3.4439	-6.7757	-7.6063	-0.0368	12.6988	-4.4151	1.8987
V7	3.7648	-2.6528	0.0759	2.2951	4.4799	5.9311	0.7487	-5.7383	-4.4830
V8	5.2527	-2.7386	4.1219	-3.0424	5.0618	-1.9933	-4.9500	-4.0669	1.5263

Tabel 5. Nilai Bobot *Hidden Layer* ke *Output Layer* dari Matlab R2011

Bobot	Y
W1	-0,2967
W2	0,6617
W3	0,1705
W4	0,0994
W5	0,8344
W6	-0,4283
W7	0,5144
W8	0,5075
W9	-0,2391

Tabel 6. Nilai Bias *Input Layer* ke *Hidden Layer* dari Matlab R2011

Bobot	Vj
W1	-9,0218
W2	3,1878
W3	-9,6652
W4	-7,2256
W5	-3,0306
W6	-2,8061
W7	5,3978
W8	5,4479
W9	-8,2315

Tabel 7. Nilai Bias *Hidden Layer* ke *Output Layer* dari *Matlab R2011*

Bias	Y
$W_j$	-0,8483

$$Z_{net\ 4} = -7.2256 + (0,9000 * 4.1327) + (0,1771 * -0.4143) + (0,8572 * 8.3462) + (0,8331 * -4.5372) + (0,1991 * 19.0398) + (0,4982 * -6.7757) + (0,8490 * 2.2951) + (0,3864 * -3.0424) = 1,1329$$

$$Z_{net\ 5} = -3,0306 + (0,9000 * 1.3154) + (0,1771 * 8.4562) + (0,8572 * 3.4169) + (0,8331 * -4.0038) + (0,1991 * -20.0764) + (0,4982 * -7.6063) + (0,8490 * 4.4799) + (0,3864 * 5.0618) = -2,7217$$

$$Z_{net\ 6} = -2,8061 + (0,9000 * -4.5069) + (0,1771 * -11.3636) + (0,8572 * 5.5521) + (0,8331 * 3.6222) + (0,1991 * -0.7395) + (0,4982 * -0.0368) + (0,8490 * -5.9311) + (0,3864 * -1.9933) = 3,1012$$

$$Z_{net\ 7} = -5,3978 + (0,9000 * -3.0457) + (0,1771 * -3.0484) + (0,8572 * 6.4309) + (0,8331 * 2.6789) + (0,1991 * -4.8270) + (0,4982 * 12.6988) + (0,8490 * 0.7487) + (0,3864 * -4.9500) = 3,2701$$

$$Z_{net\ 8} = 5.4476 + (0,9000 * 0.6456) + (0,1771 * 16.2216) + (0,8572 * -2.8547) + (0,8331 * -2.5186) + (0,1991 * 12.9980) + (0,4982 * -4.4151) + (0,8490 * -5.7383) + (0,3864 * -4.0669) = -1,7492$$

$$Z_{net\ 9} = -8.2315 + (0,9000 * 5.0714) + (0,1771 * 9.1783) + (0,8572 * 3.3153) + (0,8331 * 4.9907) + (0,1991 * 14.9724) + (0,4982 * 1.8987) + (0,8490 * -4.4830) + (0,3864 * 1.5263) = 5,7281$$

Kemudian mencari nilai sigmoid :

$$Z_1 = \text{sigmoid} [-0,6467] = \frac{1}{1+e^{-(-0,6467)}} = 0,3437$$

$$Z_2 = \text{sigmoid} [-5,2182] = \frac{1}{1+e^{-(-5,2182)}} = 0,0054$$

$$Z_3 = \text{sigmoid} [0,7459] = \frac{1}{1+e^{-(-0,7459)}} = 0,6783$$

$$Z_4 = \text{sigmoid} [1,1329] = \frac{1}{1+e^{-(-1,1329)}} = 0,7564$$

$$Z_5 = \text{sigmoid} [-2,7217] = \frac{1}{1+e^{-(-2,7217)}} = 0,0617$$

$$Z_6 = \text{sigmoid} [3,1012] = \frac{1}{1+e^{-(-3,1012)}} = 0,9569$$

$$Z_7 = \text{sigmoid} [3,2701] = \frac{1}{1+e^{-(-3,2701)}} = 0,9634$$

$$Z_8 = \text{sigmoid} [-1,7492] = \frac{1}{1+e^{-(-1,7492)}} = 0,1481$$

$$Z_9 = \text{sigmoid} [5,7281] = \frac{1}{1+e^{-(-5,7281)}} = 0,9968$$

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

Karena jaringannya hanya memiliki sebuah unit keluaran  $y$  maka  $y_{net\ k}$

$$y_{net\ k} = Y_{net\ k} = -0,8483 + (0,3437 * 0,2967) + (0,0054 * 0,6617) + (0,6783 * 0,1705) + (0,7564 * 0,0994) + (0,0617 * 0,8344) + (0,9569 * -0,4283) + (0,9634 * 0,5144) + (0,1481 * 0,5075) + (0,9968 * -0,2391) + (0,6848 * 0,1356) = -0,6889$$

(sesuai dengan nilai Y di Aplikasi Matlab R2011b)

$$= \text{sigmoid} [-0,6899]$$

$$= 1/(1+e^{(-(-0,6899))})$$

$$= 0,3343$$

**Fase 2 : Propagasi Mundur (backpropagation)**

**Langkah ke 6 :** Hitung faktor berdasarkan kesalahan setiap unit keluaran  $Y_k$

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

$$= (0,8571 - 0,3343) * 0,3343 * (1 - 0,3343)$$

$$= 0,1164$$

Hitung perubahan bobot  $W_{jk}$  (digunakan dalam mengubah bobot  $W_{jk}$ ) dengan laju pembelajaran (*learning rate*)  $\alpha = 0,5$  dengan persamaan:

$$\Delta w_{[1,1]} = \alpha * \delta_k * z_1$$

$$= 0,5 * (0,1164) * 0,3437 = 0,0200$$

$$\Delta w_{[1,2]} = \alpha * \delta_k * z_2$$

$$= 0,5 * (0,1164) * 0,0054 = 0,0003$$

$$\Delta w_{[1,3]} = \alpha * \delta_k * z_3$$

$$= 0,5 * (0,1164) * 0,6783 = 0,0395$$

$$\Delta w_{[1,4]} = \alpha * \delta_k * z_4$$

$$= 0,5 * (0,1164) * 0,7564 = 0,0440$$

$$\Delta w_{[1,5]} = \alpha * \delta_k * z_5$$

$$= 0,5 * (0,1164) * 0,0617 = 0,0036$$

$$\Delta w_{[1,6]} = \alpha * \delta_k * z_6$$

$$= 0,5 * (0,1164) * 0,9569 = 0,0096$$

$$\Delta w_{[1,7]} = \alpha * \delta_k * z_7$$

$$= 0,5 * (0,1164) * 0,9634 = 0,0002$$

$$\Delta w_{[1,8]} = \alpha * \delta_k * z_8$$

$$= 0,5 * (0,1164) * 0,1481 = 0,0029$$

$$\Delta w_{[1,9]} = \alpha * \delta_k * z_9$$

$$= 0,5 * (0,1164) * 0,9968 = 0,0219$$

$$\Delta w_{[0]} = \alpha * \delta$$

$$= 0,5 * (0,1164) = 0,0582$$

**Langkah 7 :** Hitung faktor *error*  $\delta$  unit tersembunyi (*hidden layer*) berdasarkan *error* di setiap unit tersembunyi (*hidden layer*)  $z_j$  ( $j=1,2,\dots,p$ ).

$$\delta_{net(j)} = \sum_{k=1}^m (\delta_k * w_{kj})$$

$$\delta_{in\ j} = \delta_k * w_{jk}$$

$$\delta_{net1} = (0,1164) * 0,2967 = 0,0345$$

$$\delta_{net2} = (0,1164) * 0,6617 = 0,0770$$

$$\delta_{net3} = (0,1164) * 0,1705 = 0,0198$$

$$\delta_{net4} = (0,1164) * 0,0994 = 0,0116$$

$$\begin{aligned}\delta_{net5} &= (0,1164) * 0,8344 = -0,0971 \\ \delta_{net6} &= (0,1164) * -0,4283 = -0,0498 \\ \delta_{net7} &= (0,1164) * 0,5144 = 0,0599 \\ \delta_{net8} &= (0,1164) * 0,5075 = 0,0591 \\ \delta_{net9} &= (0,1164) * -0,2391 = -0,0278\end{aligned}$$

Kemudian hitung faktor *error*  $\delta$  di unit tersembunyi (*hidden layer*) dengan rumus persamaan:

$$\delta_k = \delta_{netj} f'(z_{netj}) = \delta_{netj} * Z_j * (1 - z_j)$$

$$\begin{aligned}\delta_1 &= (0,0345) * 0,3437 * (1 - 0,3439) = -0,0078 \\ \delta_2 &= (0,0770) * 0,0054 * (1 - 0,0054) = 0,0004 \\ \delta_3 &= (0,0198) * 0,6783 * (1 - 0,6784) = 0,0043 \\ \delta_4 &= (0,0116) * 0,7564 * (1 - 0,7564) = 0,0021 \\ \delta_5 &= (-0,0971) * 0,0617 * (1 - 0,0617) = 0,0056 \\ \delta_6 &= (-0,0498) * 0,9569 * (1 - 0,9569) = -0,0021 \\ \delta_7 &= (0,0599) * 0,9634 * (1 - 0,9634) = 0,0021 \\ \delta_8 &= (0,0591) * 0,1481 * (1 - 0,1481) = 0,0075 \\ \delta_9 &= (-0,0278) * 0,9968 * (1 - 0,9968) = -0,0001\end{aligned}$$

Hitung suku perubahan bobot (baru) ke unit tersembunyi (*hidden layer*) dengan rumus persamaan:

$$\Delta v_{0,j} = \alpha * \delta_j * x_i \text{ dimana } \alpha = 0,5$$

X1 = 0,9000

$$\begin{aligned}\Delta v_{1.1} &= 0,5 * (-0,0078) * 0,9000 = -0,0035 \\ \Delta v_{1.2} &= 0,5 * (0,0004) * 0,9000 = 0,0002 \\ \Delta v_{1.3} &= 0,5 * (0,0043) * 0,9000 = 0,0019 \\ \Delta v_{1.4} &= 0,5 * (0,0021) * 0,9000 = 0,0010 \\ \Delta v_{1.5} &= 0,5 * (0,0056) * 0,9000 = 0,0025 \\ \Delta v_{1.6} &= 0,5 * (-0,0021) * 0,9000 = -0,0009 \\ \Delta v_{1.7} &= 0,5 * (0,0021) * 0,9000 = 0,0010 \\ \Delta v_{1.8} &= 0,5 * (0,0075) * 0,9000 = 0,0034 \\ \Delta v_{1.9} &= 0,5 * (-0,0001) * 0,9000 = -0,0000 \\ \Delta v_{1.10} &= 0,5 * (0,0034) * 0,9000 = 0,0015\end{aligned}$$

X2 = 0,1771

$$\begin{aligned}\Delta v_{2.1} &= 0,5 * (0,0078) * 0,1771 = -0,0007 \\ \Delta v_{2.2} &= 0,5 * (0,0004) * 0,1771 = 0,0000 \\ \Delta v_{2.3} &= 0,5 * (0,0043) * 0,1771 = 0,0004 \\ \Delta v_{2.4} &= 0,5 * (0,0021) * 0,1771 = 0,0002 \\ \Delta v_{2.5} &= 0,5 * (0,0056) * 0,1771 = 0,0005 \\ \Delta v_{2.6} &= 0,5 * (-0,0021) * 0,1771 = -0,0002 \\ \Delta v_{2.7} &= 0,5 * (0,0021) * 0,1771 = 0,0002 \\ \Delta v_{2.8} &= 0,5 * (0,0075) * 0,1771 = 0,0007 \\ \Delta v_{2.9} &= 0,5 * (-0,0001) * 0,1771 = 0,0000 \\ \Delta v_{2.10} &= 0,5 * (0,0034) * 0,1771 = 0,0003\end{aligned}$$

X3 = 0,8752

$$\begin{aligned}\Delta v_{3.1} &= 0,5 * (0,0078) * 0,8752 = -0,0034 \\ \Delta v_{3.2} &= 0,5 * (0,0004) * 0,8752 = 0,0002 \\ \Delta v_{3.3} &= 0,5 * (0,0043) * 0,8752 = 0,0019 \\ \Delta v_{3.4} &= 0,5 * (0,0021) * 0,8752 = 0,0009 \\ \Delta v_{3.5} &= 0,5 * (0,0056) * 0,8752 = 0,0025 \\ \Delta v_{3.6} &= 0,5 * (-0,0021) * 0,8752 = -0,0009 \\ \Delta v_{3.7} &= 0,5 * (0,0021) * 0,8752 = 0,0009\end{aligned}$$

$$\begin{aligned}\Delta v_{3.8} &= 0,5 * (0,0075) * 0,8752 = 0,0033 \\ \Delta v_{3.9} &= 0,5 * (-0,0001) * 0,8752 = 0,0000 \\ \Delta v_{3.10} &= 0,5 * (0,0034) * 0,8752 = 0,0015\end{aligned}$$

X4 = 0,8331

$$\begin{aligned}\Delta v_{4.1} &= 0,5 * (0,0078) * 0,8331 = -0,0032 \\ \Delta v_{4.2} &= 0,5 * (0,0004) * 0,8331 = 0,0002 \\ \Delta v_{4.3} &= 0,5 * (0,0042) * 0,8331 = 0,0018 \\ \Delta v_{4.4} &= 0,5 * (0,0021) * 0,8331 = 0,0009 \\ \Delta v_{4.5} &= 0,5 * (0,0054) * 0,8331 = 0,0023 \\ \Delta v_{4.6} &= 0,5 * (-0,0021) * 0,8331 = -0,0009 \\ \Delta v_{4.7} &= 0,5 * (0,0021) * 0,8331 = 0,0009 \\ \Delta v_{4.8} &= 0,5 * (0,0075) * 0,8331 = 0,0031 \\ \Delta v_{4.9} &= 0,5 * (-0,0001) * 0,8331 = 0,0000 \\ \Delta v_{4.10} &= 0,5 * (0,0034) * 0,8331 = 0,0014\end{aligned}$$

X5 = 0,1991

$$\begin{aligned}\Delta v_{5.1} &= 0,5 * (0,0078) * 0,1991 = -0,0008 \\ \Delta v_{5.2} &= 0,5 * (0,0004) * 0,1991 = 0,0000 \\ \Delta v_{5.3} &= 0,5 * (0,0043) * 0,1991 = 0,0004 \\ \Delta v_{5.4} &= 0,5 * (0,0021) * 0,1991 = 0,0002 \\ \Delta v_{5.5} &= 0,5 * (0,0056) * 0,1991 = 0,0006 \\ \Delta v_{5.6} &= 0,5 * (-0,0021) * 0,1991 = -0,0002 \\ \Delta v_{5.7} &= 0,5 * (0,0020) * 0,1991 = 0,0002 \\ \Delta v_{5.8} &= 0,5 * (0,0075) * 0,1991 = 0,0007 \\ \Delta v_{5.9} &= 0,5 * (-0,0001) * 0,1991 = 0,0000 \\ \Delta v_{5.10} &= 0,5 * (0,0034) * 0,1991 = 0,0003\end{aligned}$$

X6 = 0,4982

$$\begin{aligned}\Delta v_{6.1} &= 0,5 * (0,0078) * 0,4982 = -0,0019 \\ \Delta v_{6.2} &= 0,5 * (0,0004) * 0,4982 = 0,0001 \\ \Delta v_{6.3} &= 0,5 * (0,0043) * 0,4982 = 0,0011 \\ \Delta v_{6.4} &= 0,5 * (0,0021) * 0,4982 = 0,0005 \\ \Delta v_{6.5} &= 0,5 * (0,0054) * 0,4982 = 0,0014 \\ \Delta v_{6.6} &= 0,5 * (-0,0021) * 0,4982 = -0,0005 \\ \Delta v_{6.7} &= 0,5 * (0,0021) * 0,4982 = 0,0005 \\ \Delta v_{6.8} &= 0,5 * (0,0075) * 0,4982 = 0,0019 \\ \Delta v_{6.9} &= 0,5 * (-0,0001) * 0,4982 = 0,0000 \\ \Delta v_{6.10} &= 0,5 * (0,0034) * 0,4982 = 0,0008\end{aligned}$$

X7 = 0,8490

$$\begin{aligned}\Delta v_{7.1} &= 0,5 * (0,0078) * 0,8490 = -0,0033 \\ \Delta v_{7.2} &= 0,5 * (0,0004) * 0,8490 = 0,0002 \\ \Delta v_{7.3} &= 0,5 * (0,0043) * 0,8490 = 0,0018 \\ \Delta v_{7.4} &= 0,5 * (0,0021) * 0,8490 = 0,0009 \\ \Delta v_{7.5} &= 0,5 * (0,0056) * 0,8490 = 0,0024 \\ \Delta v_{7.6} &= 0,5 * (-0,0021) * 0,8490 = -0,0009 \\ \Delta v_{7.7} &= 0,5 * (0,0021) * 0,8490 = 0,0009 \\ \Delta v_{7.8} &= 0,5 * (0,0075) * 0,8490 = 0,0032 \\ \Delta v_{7.9} &= 0,5 * (-0,0001) * 0,8490 = 0,0000 \\ \Delta v_{7.10} &= 0,5 * (0,0034) * 0,8490 = 0,0014\end{aligned}$$

X8 = 0,3864

$$\begin{aligned}\Delta v_{8.1} &= 0,5 * (0,0078) * 0,3864 = -0,0015 \\ \Delta v_{8.2} &= 0,5 * (0,0004) * 0,3864 = 0,0001 \\ \Delta v_{8.3} &= 0,5 * (0,0043) * 0,3864 = 0,0008 \\ \Delta v_{8.4} &= 0,5 * (0,0021) * 0,3864 = 0,0004\end{aligned}$$



$$\begin{aligned}\Delta v_{8,5} &= 0,5 * (0,0056) * 0,3864 = 0,0011 \\ \Delta v_{8,6} &= 0,5 * (-0,0021) * 0,3864 = -0,0004 \\ \Delta v_{8,7} &= 0,5 * (0,0021) * 0,3864 = 0,0004 \\ \Delta v_{8,8} &= 0,5 * (0,0075) * 0,3864 = 0,0014 \\ \Delta v_{8,9} &= 0,5 * (-0,0001) * 0,3864 = 0,0000 \\ \Delta v_{8,10} &= 0,5 * (0,0034) * 0,3864 = 0,0007\end{aligned}$$

Langkah selanjutnya hitung koreksi nilai Bias dari unit tersembunyi (*hidden layer*).

$$\Delta V_{j0} = \alpha * \delta_j$$

$$\begin{aligned}\Delta V_{j1} &= 0,5 * (0,0078) = -0,0039 \\ \Delta V_{j2} &= 0,5 * (0,0004) = 0,0002 \\ \Delta V_{j3} &= 0,5 * (-0,0043) = 0,0022 \\ \Delta V_{j4} &= 0,5 * (0,0021) = 0,0011 \\ \Delta V_{j5} &= 0,5 * (0,0056) = 0,0028 \\ \Delta V_{j6} &= 0,5 * (-0,0021) = -0,0011 \\ \Delta V_{j7} &= 0,5 * (0,0021) = 0,0011 \\ \Delta V_{j8} &= 0,5 * (0,0075) = 0,0037 \\ \Delta V_{j9} &= 0,5 * (-0,0001) = 0,0000\end{aligned}$$

### Fase III : Perubahan Bobot

**Langkah 8 :** Hitung semua perubahan bobot.

Langkah setelah ini adalah menghitung nilai bobot unit tersembunyi (*hidden*) dengan persamaan:

$$V_{oj}(\text{baru}) = V_{oj}(\text{lama}) + \Delta V_{oj}$$

$$\begin{aligned}V_{1.1}(\text{baru}) &= V_{1.1}(\text{lama}) + \Delta v_{1.1} \\ &= 4,0893 + (-0,0035) = 4,0858 \\ V_{1.2}(\text{baru}) &= V_{1.2}(\text{lama}) + \Delta v_{1.2} \\ &= -12,6059 + (0,0007) = -12,6066 \\ V_{1.3}(\text{baru}) &= V_{1.3}(\text{lama}) + \Delta v_{1.3} \\ &= 3,8926 + (0,0034) = 3,8892 \\ V_{1.4}(\text{baru}) &= V_{1.4}(\text{lama}) + \Delta v_{1.4} \\ &= 2,6772 + (0,0032) = 2,6740 \\ V_{1.5}(\text{baru}) &= V_{1.5}(\text{lama}) + \Delta v_{1.5} \\ &= -5,1347 + (0,0008) = -5,1355 \\ V_{1.6}(\text{baru}) &= V_{1.6}(\text{lama}) + \Delta v_{1.6} \\ &= -5,8449 + (-0,0019) = -5,8478 \\ V_{1.7}(\text{baru}) &= V_{1.7}(\text{lama}) + \Delta v_{1.7} \\ &= 3,7648 + (0,0033) = 3,7615 \\ V_{1.8}(\text{baru}) &= V_{1.8}(\text{lama}) + \Delta v_{1.8} \\ &= 5,2527 + (0,0015) = 5,2512\end{aligned}$$

hitung perubahan nilai *bobot* garis yang menuju ke unit keluaran (*output*) dengan rumus persamaan :

$$W_{oj}(\text{baru}) = W_{oj}(\text{lama}) + \Delta W_{oj}$$

$$\begin{aligned}W_1(\text{baru}) &= w_1(\text{lama}) + \Delta w_1 \\ &= (-0,2967) + (0,0200) = 0,2767 \\ W_2(\text{baru}) &= w_2(\text{lama}) + \Delta w_2 \\ &= (0,6617) + (0,0003) = 0,6620 \\ W_3(\text{baru}) &= w_3(\text{lama}) + \Delta w_3\end{aligned}$$

$$\begin{aligned}&= (0,1705) + (0,0395) = 0,2100 \\ W_4(\text{baru}) &= w_4(\text{lama}) + \Delta w_4 \\ &= (0,0994) + (0,0440) = 0,1434 \\ W_5(\text{baru}) &= w_5(\text{lama}) + \Delta w_5 \\ &= (0,8344) + (0,0036) = 0,8380 \\ W_6(\text{baru}) &= w_6(\text{lama}) + \Delta w_6 \\ &= (-0,4283) + (0,0096) = -0,4187 \\ W_7(\text{baru}) &= w_7(\text{lama}) + \Delta w_7 \\ &= (0,5144) + (0,0002) = 0,5146 \\ W_8(\text{baru}) &= w_8(\text{lama}) + \Delta w_8 \\ &= (0,5075) + (0,0029) = 0,5104 \\ W_9(\text{baru}) &= w_9(\text{lama}) + \Delta w_9 \\ &= (-0,2391) + (0,0219) = -0,2172\end{aligned}$$

Kemudian hitung perubahan nilai *bias* dari unit tersembunyi (*hidden layer*) ke *output*:

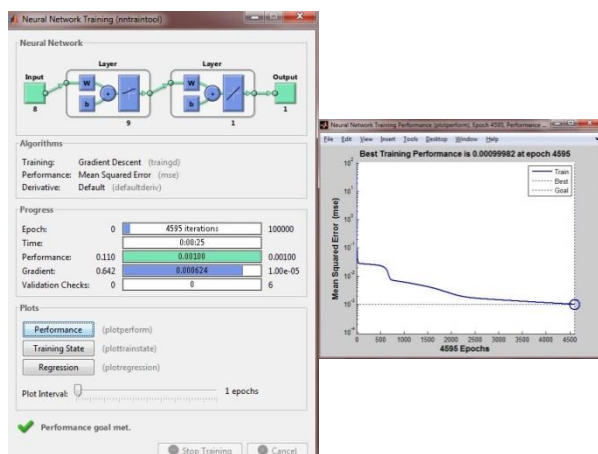
$$\begin{aligned}W_{j0}(\text{bias baru}) &= w_{j0}(\text{lama}) + \Delta w_{j0} \\ &= -0,8483 + 0,0582 = -0,7901\end{aligned}$$

Setelah proses pelatihan selesai, data keluaran (*output*) yang sudah didapatkan dari pelatihan jaringan yang masih dalam bentuk normalisasi, sehingga mesti dilakukannya proses *denormalisasi* data yang bertujuan untuk mengembalikan nilai hasil keluaran data yang masih dalam bentuk normalisasi ke dalam bentuk data yang sebenarnya, dengan persamaan :

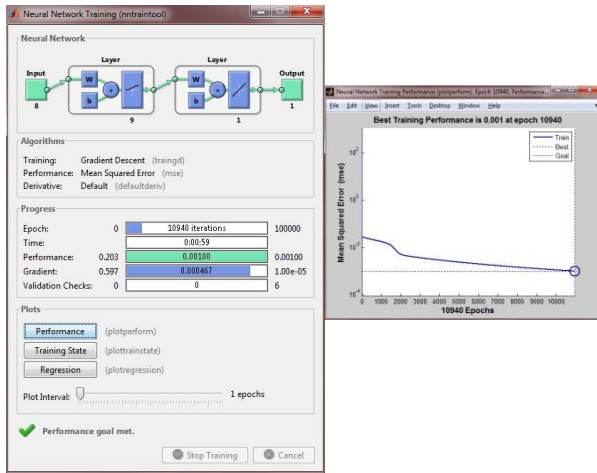
$$\begin{aligned}X' &= x(x_{\max} - x_{\min}) + x_{\min} \\ X' &= -0,6889 (146306 - 91734) + 91734 \\ &= 54135,60132\end{aligned}$$

Hasil *denormalisasi* dari keluaran target -0,6889 yang dihasilkan adalah 54135,60132. Hasil ini dinilai masih mengandung nilai *error* yang besar, karena target yang harus dicapai adalah 2840,165 (Target).

## 4.2 Implementasi Pada Matlab R2011b



Gambar 2 : Pola dan Grafik Goal Pelatihan 8-9-1



Gambar 3 : Goal Pengujian Data dengan Pola 8-9-1

Tabel 8. Perbandingan Terhadap Pola Pengujian

No	Pola	Epoch (Pelatihan)	MSE (Pengujian)	Akurasi (100-MSE)
1	8-7-1	1845	0,00099986	99,99900014%
2	8-8-1	7850	0,00099987	99,99900014%
3	<b>8-9-1</b>	<b>4595</b>	<b>0,001</b>	<b>99,99900000%</b>
4	8-10-1	6517	0,00099982	99,99900018%

Tabel 9. Hasil Prediksi JST Pola 8-9-1

Tahun	Bulan	Data Aktual	Data JST
2016	Jan	143382	138409
	Feb	125707	124052
	Mar	106641	110278
	Apr	146306	139403
	Mei	132163	129923
	Jun	117806	116919
	Jul	126808	124390
	Aug	130390	125039
	Sept	110201	111942
	Okt	126294	129236
	Nov	126285	125923
	Des	91734	97088
2017	Jan	133427	127964
	Feb	105918	107691
	Mar	110958	115085
	Apr	127255	125667
	Mei	128638	127413
	Jun	98535	102015
	Jul	117084	120073
	Aug	119098	117159
	Sept	121260	120892
	Okt	95830	103129
	Nov	99009	101830
	Des	99436	101868
<b>Total</b>		<b>2.840.165</b>	<b>2.843.388</b>
Rata -Rata		118.340	118.475

Dapat dilihat jumlah data aktual adalah 2.840.165 hampir mendekati dengan hasil prediksi menggunakan pola 8-9-1 dengan hasil perhitungan yaitu **2.843.388**.

Dari Tabel dan gambar yang ada dapat dilihat perbandingan data prediksi yang dilakukan dengan Jaringan Syaraf Tiruan tidak berbeda jauh dari data aktual PT. PDAM Kota Duri. Hasil akhir dari penelitian ini adalah Jaringan syaraf tiruan menggunakan metode *backpropagation* dapat memprediksi penggunaan konsumsi air di PT. PDAM Kota Duri untuk tahun selanjutnya.

Maka dapat disimpulkan bahwa model terbaik adalah pola 8-9-1 dengan proses epoch pelatihan = 4595 dan pencapaian *MSE* pada saat pengujian dengan *MSE* = 0,001 dengan akurasi 99,99900000% (100-MSE).

## 5. Kesimpulan

### 5.1 Simpulan

Berdasarkan analisa data pada pelatihan (*training*) dan pengujian (*testing*) tentang prediksi penggunaan air di PT. PDAM Kota Duri, maka dari penelitian ini dapat diambil beberapa kesimpulan sebagai berikut :

1. Penerapan metode Jaringan Syaraf Tiruan (JST) dengan algoritma *backpropagation* dapat memprediksi jumlah volume penggunaan air di masa mendatang pada PT. PDAM Kota Duri dengan menggunakan *Matlab R2011b*. Dengan menerapkan arsitektur jaringan yang paling tepat untuk digunakan dalam memprediksi penggunaan konsumsi air adalah dengan pola arsitektur 8-9-1 dengan hasil persentasi 99,99900000%
2. Penentuan dan pendataan variabel dapat diterapkan pada metode Jaringan Syaraf Tiruan (JST) dan dapat membantu mengatasi jumlah produksi air di PT. PDAM Kota Duri.

### 5.2 Saran

Berdasarkan hasil penelitian yang telah penulis lakukan melalui proses pelatihan data dan pengujian data dengan algoritma *backpropagation*, saran yang dapat diberikan adalah sebagai berikut.

1. Perlu dilakukan proses prapengolahan (*preprocessing*) yang lebih kompleks dengan data yang lebih banyak karena jumlah data mempengaruhi tingkat akurasi data.
2. Jumlah lapisan tersembunyi (*hidden layer*) yang digunakan ditambah lebih banyak untuk meneliti keakuratan jaringan dalam memprediksi penggunaan konsumsi air di PT. PDAM Kota Duri .
3. Diharapkan adanya pengembangan lebih lanjut agar dapat lebih meningkatkan pemahaman terhadap algoritma *backpropagation* dalam melakukan prediksi. Berisi saran-saran perbaikan dan penelitian lanjutan.

**Daftar Rujukan**

- [1] Monica, Mesran, Suginam, & Karim. (2017). Jambu Biji Menggunakan Metode Bayes, 2(1), 78–81.
- [2] Kung, J., Kim, D., & Mukhopadhyay, S. (2015). A power-aware digital feedforward neural network platform with backpropagation driven approximate synapses. *Proceedings of the International Symposium on Low Power Electronics and Design, 2015–Septe*, 85–90. <https://doi.org/10.1109/ISLPED.2015.7273495>
- [3] Pan, X., Lee, B., & Zhang, C. (2013). A comparison of neural network backpropagation algorithms for electricity load forecasting. In *Proceedings - 2013 IEEE International Workshop on Intelligent Energy Systems, IWIES 2013* (pp. 22–27). <https://doi.org/10.1109/IWIES.2013.6698556>
- [4] Huang, D., & Wu, Z. (2017). Forecasting outpatient visits using empirical mode decomposition coupled with backpropagation artificial neural networks optimized by particle swarm optimization. *PLoS ONE*, 12(2). <https://doi.org/10.1371/journal.pone.0172539>
- [5] Hrasko, R., Pacheco, A. G. C., & Krohling, R. A. (2015). Time Series Prediction Using Restricted Boltzmann Machines and Backpropagation. *Procedia Computer Science*, 55, 990–999. <https://doi.org/10.1016/j.procs.2015.07.104>
- [6] Lahmiri, S. (2014). Wavelet low- and high-frequency components as features for predicting stock prices with backpropagation neural networks. *Journal of King Saud University - Computer and Information Sciences*, 26(2), 218–227. <https://doi.org/10.1016/j.jksuci.2013.12.001>
- [7] Nugraha, A. P. (2015). Aplikasi Sistem Pakar Untuk Mendiagnosa Penyakit Dalam Menggunakan Metode Forward Chaining Dan Certainty Factor Berbasis Web ( Studi Kasus : Poliklinik Pt Pos Indonesia Bandung ) Application of Expert System to Diagnose Internal Disease Using Forward Chai, 2(2), 3676–3683.
- [8] Rahayu, S. (2013). Sistem pakar untuk mendiagnosa penyakit gagal ginjal dengan menggunakan metode bayes. *Pelita Informatika Budi Darma*, IV(3), 129–134
- [9] Ihwan, A. (2013). Metode Jaringan Saraf Tiruan Propagasi Balik Untuk Estimasi Curah Hujan Bulanan di Ketapang Kalimantan Barat, 243–247.
- [10] Zekson Arizona Matondang. (2013). “Jaringan Syaraf Tiruan dengan Algoritma Backpropagation untuk Penentuan Kelulusan Sidang Skripsi”. *Pelita Informatika Budi Darma*, 1 Agustus
- [11] Abidah, S. (2013). Analisis komparasi metode tsukamoto dan sugeno dalam prediksi jumlah siswa baru. *Jurnal Teknologi Informasi Dan Komunikasi*, 8(2), 57–63.
- [12] Simangunsong, F. R. D. B., & Nasution, S. D. (2015). Aplikasi Jaringan Saraf Tiruan Untuk Memprediksi Metode Back Propagation, 2(6), 43–47.
- [13] Ujianto, Y., dan Isa, M. (2015). Perbandingan Performansi Metode Peramalan Fuzzy Time Series yang Dimodifikasi dan Jaringan Syaraf Tiruan Backpropagation (Studi Kasus: Penutupan Harga IHSG), 4(2).
- [14] Dar, M. H. (2017). Penerapan Metode Backpropagation Neural Network Untuk Memprediksi Produksi Air, 12, 203–208