

PREDIKSI pH AIR HUJAN DI KALIMANTAN BARAT DENGAN METODE BACKPROPAGATION

Fra Siskus Dian Arianto, Muhlasah Novitasari Mara, Naomi Nessyana Debataraaja

INTISARI

Hujan asam dipengaruhi oleh emisi gas Karbon Monoksida (CO), Sulfur Dioksida (SO₂), dan Nitrogen Dioksida (NO₂). Terjadinya hujan asam diakibatkan oleh gas CO, SO₂, dan NO₂ yang mencair bersamaan dengan air hujan sehingga membentuk asam pada air tersebut. Dikatakan hujan asam apabila pH air kurang dari 5,6 dan jika terjadi hujan asam maka dapat menyebabkan berbagai macam penyakit. Oleh karena itu, informasi hujan asam penting diketahui dengan cara memperkirakan pH air hujan. Salah satu upaya untuk memperkirakan permasalahan tersebut yaitu dengan cara membentuk model untuk mensimulasikan hujan asam dengan akurasi yang tinggi. Adapun data CO, SO₂, NO₂, dan pH air hujan yang digunakan dalam penelitian ini yaitu sebanyak 55 pengamatan. Berdasarkan data tersebut, digunakan metode Backpropagation untuk menentukan model terbaik pH air hujan. Dari hasil 40 percobaan terhadap neuron pada lapisan tersembunyi diperoleh struktur jaringan yang outputnya mendekati data aktual pH air hujan yaitu 3 unit neuron pada lapisan input, 2 unit neuron pada lapisan tersembunyi, dan 1 unit neuron pada lapisan output. Struktur jaringan tersebut menghasilkan MSE 0,06414, MAD 0,212, MAPE 0,041349, dan R -0,41573.

Kata kunci: Algoritma Backpropagation, Gradient Descent

PENDAHULUAN

Hujan asam berpengaruh terhadap kualitas air di Kalimantan Barat. pH menjadi salah satu faktor untuk mengetahui terjadinya hujan asam. Hujan asam terjadi saat pH yang dihasilkan dibawah pH normal yaitu 5,6. Dari tahun 2009 sampai 2013 rata-rata pH air hujan di Kalimantan Barat yaitu 5,2 berdasarkan Badan Meteorologi, Klimatologi dan Geofisika (BMKG) Stasiun Klimatologi Siantan Pontianak. Air yang terkontaminasi oleh hujan asam apabila dikonsumsi dapat menyebabkan berbagai macam gangguan kesehatan. Masyarakat Kalimantan Barat dalam kegiatan sehari-hari sangat bergantung pada air sehingga hujan asam merupakan suatu masalah lingkungan yang serius. Informasi hujan asam penting diketahui untuk mengetahui batasan faktor yang menyebabkan kejadian iklim, ketika terjadi hujan asam yang berlebihan maka dapat ditangani dengan memperhatikan kombinasi faktor penyebab kejadian iklim. Salah satu upaya untuk memperkirakan permasalahan tersebut yaitu dengan melalui pembentukan model simulasi pH air hujan dengan akurasi yang tinggi.

Hujan asam di Kalimantan Barat di pengaruhi oleh emisi bahan bakar berupa gas Karbon Monoksida (CO), Sulfur Dioksida (SO₂), dan Nitrogen Dioksida (NO₂). pH merupakan logaritma negatif konsentrasi ion hydrogen (H⁺). Karbon Monoksida (CO) hampir merupakan zat nonpolar yang tidak mudah larut didalam air, namun ketika gas CO ini berada di udara akan cepat teroksidasi menjadi CO₂. Karbon Dioksida (CO₂) merupakan molekul linear dan nonpolar karena strukturnya yang simetris, zat ini mudah larut didalam air dan larutannya bersifat asam karbonat. Sulfur Dioksida (SO₂) merupakan molekul nonlinear dan berbentuk polar, cairan SO₂ didalam air membentuk asam encer. Nitrogen dioksida (NO₂) merupakan zat *intermediate* dalam reaksi ammonia menjadi asam nitrat [1]. Gas CO, SO₂, dan NO₂ yang dikeluarkan oleh kendaraan, pabrik, dan dari alam terpapar ke udara, jika terjadi hujan, gas pencemar tersebut akan mencair menjadi asam sehingga pH air tersebut menjadi kurang dari 5,6.

Jaringan syaraf tiruan merupakan salah satu representasi buatan dari otak manusia yang selalu mencoba untuk mensimulasikan proses pembelajaran pada otak manusia tersebut. Istilah buatan disini digunakan karena jaringan syaraf ini diimplementasikan dengan menggunakan program komputer yang mampu menyelesaikan sejumlah proses perhitungan selama proses pembelajaran [2][3]. Terdapat

beberapa metode pada jaringan syaraf tiruan yaitu metode *perceptron*, *backpropagation*, *self organizing*, *learning vector quantization*. Metode *perceptron* merupakan jaringan syaraf sederhana yang digunakan untuk mengklasifikasikan suatu tipe pola tertentu yang sering dikenal dengan pemisahan secara linear. Metode *Backpropagation* merupakan algoritma pembelajaran yang terawasi dan metode ini sering digunakan untuk pembentukan model jaringan basis radial. Jaringan syaraf berbasis radial biasanya membutuhkan neuron lebih banyak dibanding dengan jaringan *feedword*. metode *Self organizing* yaitu jaringan pada lapisannya berisi neuron-neuron yang akan menyusun diri berdasarkan *input* nilai tertentu dalam suatu kelompok yang dikenal dengan istilah *cluster*. Kemudian *learning vector quantization* merupakan suatu metode untuk melakukan pelatihan terhadap lapisan-lapisan kompotitif yang terawasi [2][3]. Pada penelitian ini digunakan metode *backpropagation* sebagai metode penentuan model untuk memperkirakan pH air hujan di Kalimantan Barat pada masa mendatang.

Metode *backpropagation* digunakan diberbagai bidang diantaranya yaitu, Anindita Septiani dan Nur Sya'baniah (2012) membahas tentang sistem peramalan jumlah produksi air PDAM Samarinda menggunakan jaringan syaraf tiruan *backpropagation*. Penelitian tersebut menemukan bahwa jumlah data yang digunakan pada penelitian berbanding lurus dengan kinerja sistem ini, artinya semakin banyak data latih yang digunakan untuk penelitian, hasil prediksinya akan semakin membaik [4]. Aprianti (2013) membahas tentang estimasi curah hujan bulanan Kota Pontianak dengan suhu permukaan laut sebagai estimator berdasarkan jaringan syaraf tiruan *backpropagation*. Diperoleh 3 koefisien korelasi tertinggi terhadap data pengujian yaitu 0,70, 0,71 dan 0,75, yang masing-masing berada di sekitar garis *equator*. Koefisien korelasi ini menunjukkan bahwa pola SPL mendekati pola curah hujan di Pontianak [5]. Tujuan dari penelitian ini adalah membuat model prediksi pH air hujan menggunakan metode *backpropagation* pada Jaringan Syaraf Tiruan. Pembentukan model untuk perkiraan pH air hujan yang ditentukan berdasarkan emisi bahan bakar berupa gas karbon monoksida (CO), sulfur dioksida (SO₂), dan nitrogen dioksida (NO₂).

METODE 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 tersembunyi. Algoritma *backpropagation* menggunakan eror *output* untuk mengubah nilai bobot-bobot dalam arah mundur (*backward*). Untuk mendapatkan eror ini, tahap perambatan maju (*forward propagation*) harus dikerjakan terlebih dahulu. Saat perambatan maju neuron-neuron diaktifkan dengan menggunakan fungsi aktivasi yang dapat dideferensiasikan, seperti sigmoid biner [2]:

$$f(x) = \frac{1}{1+e^{-x}} \quad (1)$$

dengan $f'(x) = f(x)[1 - f(x)]$

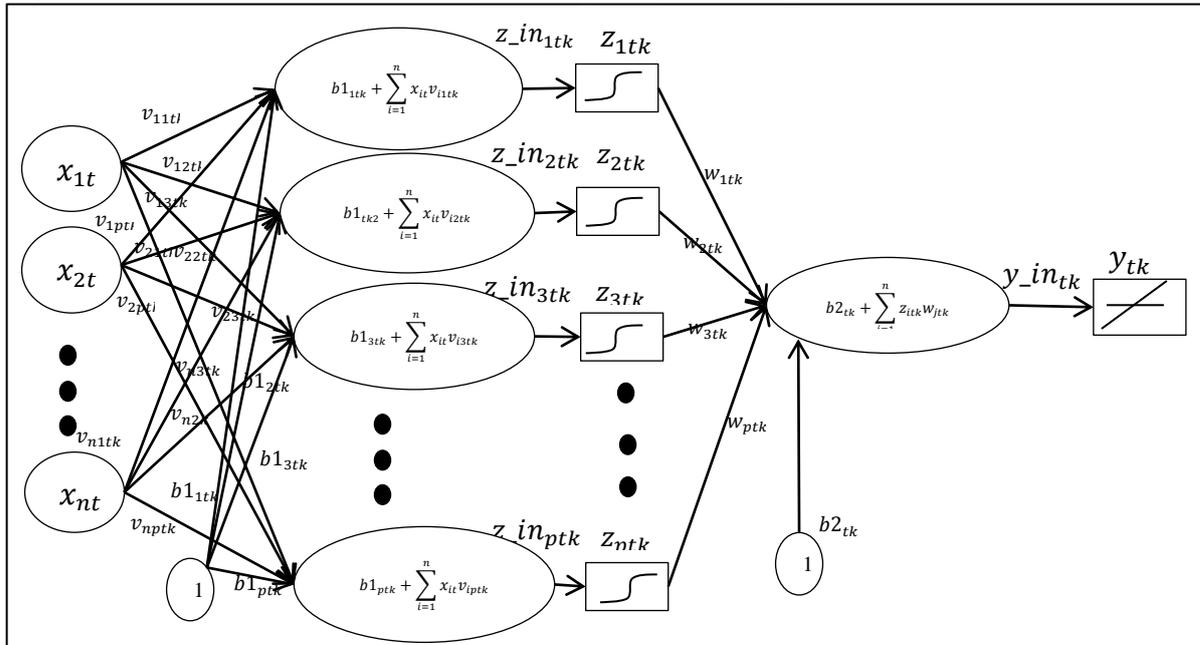
Atau fungsi linear

$$f(x) = x \quad (2)$$

dengan $f'(x) = 1$

Arsitektur jaringan *backpropagation* fungsi *gradient descent* dengan momentum disajikan pada Gambar 1. Pada Gambar 1, lapisan *input* terdiri atas n neuron dan l pengamatan, yang dinyatakan dengan x_{it} dimana x_{it} menunjukkan unit *input* untuk neuron ke- i pada pengamatan ke- t . Satu lapisan tersembunyi dengan p neuron, l pengamatan, dan m *epoch*, yang dinyatakan dengan z_{jtk} dimana z_{jtk} menunjukkan unit lapisan tersembunyi untuk neuron ke- j , pengamatan ke- t , pada *epoch* ke- k . Serta satu unit *output* yang dinyatakan dengan y_{tk} dimana unit *output* untuk pengamatan ke- t , pada *epoch* ke- k . Bobot yang menghubungkan x_{it} dengan unit neuron pada lapisan tersembunyi z_{jtk} adalah v_{ijtk} dimana v_{ijtk} merupakan bobot yang menghubungkan neuron *input* ke- i dengan unit neuron lapisan

tersembunyi ke- j pada pengamatan ke- t dan *epoch* ke- k . Bobot bias yang terhubung ke unit neuron pada lapisan tersembunyi z_{jtk} adalah b_{1jtk} dimana b_{1jtk} merupakan bobot bias neuron ke- j , pengamatan ke- t , pada *epoch* ke- k . Bobot yang menghubungkan unit neuron pada lapisan tersembunyi z_{jtk} dengan unit neuron pada lapisan *output* y_{tk} adalah w_{jtk} dengan j unit lapisan tersembunyi, t pengamatan, dan k *epoch*. Bobot bias b_{2tk} terhubung dengan lapisan *output*. Fungsi aktivasi yang digunakan pada lapisan tersembunyi yaitu fungsi aktivasi sigmoid biner (logsig) dan pada lapisan *output* fungsi aktivasi yang digunakan yaitu fungsi aktivasi linear (purelin).



Gambar 1. Arsitektur Jaringan *Backpropagation* [2]

Pembelajaran menggunakan *backpropagation* memiliki beberapa tahap pengerjaan yaitu *feedforward* dimana pada tahap ini sinyal x_{it} dipropagasikan ke neuron tersembunyi menggunakan fungsi aktivasi sigmoid biner. Substitusikan pengamatan pertama terlebih dahulu dari masing-masing unit neuron *input*. Nilai dari z_{jtk} selanjutnya dipropagasi lagi ke neuron *output* dengan menggunakan fungsi aktivasi linear yaitu nilai dari y_{tk} . Nilai pada neuron *output* y_{tk} dibandingkan dengan target atau data aktual yaitu T_t , dengan mencari kesalahan dari peramalan yaitu $T_t - y_{tk}$ dengan begitu bobot-bobot akan dimodifikasi untuk mengurangi kesalahan yang terjadi pada peramalan pertama.

Propagasi mundur atau *backpropagation* dilakukan untuk memperbaiki kesalahan dengan menghitung δ_{tk} . Nilai δ_{tk} digunakan untuk memperbaiki nilai Δw_{jtk} yaitu koreksi bobot yang menghubungkan lapisan tersembunyi ke- j dengan lapisan *output* pada pengamatan ke- t di *epoch* ke- k . Nilai koreksi bobot tersebut digunakan untuk mendapatkan bobot-bobot w_{jtk} . menghitung nilai δ_{jtk} untuk memperbaiki Δv_{ijtk} yaitu koreksi bobot yang menghubungkan lapisan *input* ke- i dengan lapisan tersembunyi ke- j pada pengamatan ke- t di *epoch* ke- k . Nilai koreksi bobot tersebut digunakan untuk mendapatkan bobot-bobot v_{ijtk} . Setelah diperoleh perbaikan bobot-bobot pada proses perhitungan di pengamatan pertama bobot-bobot tersebut digunakan untuk proses perhitungan di pengamatan kedua, dan proses tersebut dilakukan sebanyak pengamatan sampai kondisi berhenti.

Algoritma *Backpropagation*:

Inisialisasi bobot awal dengan mengambil nilai acak yang umumnya terletak antara -0,5 sampai 0,5 atau -1 sampai 1. Nilai bobot yang terlalu kecil dari -1 akan menghasilkan nilai keluaran jaringan syaraf ke neuron tersembunyi maupun ke neuron keluaran mendekati nol. Hal ini dapat menyebabkan waktu yang panjang untuk proses pembelajaran, begitu juga sebaliknya untuk nilai bobot yang lebih besar dari 1. Penetapan maksimum *epoch*, target error, momentum (mc), dan *learning rate* (α) sebagai

batasan pembelajaran. Adapun algoritma pelatihan pada metode *backpropagation* yang dikerjakan yaitu tahap *feedforward* terlebih dahulu kemudian selanjutnya tahap *backpropagation*, tahap-tahap tersebut dapat dijelaskan sebagai berikut:

Feedforward:

- Setiap sinyal *input* x_{it} ($i = 1,2,3, \dots, n$ dan $t = 1,2,3, \dots, l$) diteruskan ke semua unit pada lapisan tersembunyi.
- Setiap unit pada suatu lapisan tersembunyi menerima sinyal dari lapisan *input* dan diproses dengan menggunakan persamaan sebagai berikut:

$$z_in_{jtk} = b1_{jtk} + \sum_{i=1}^n x_{it} v_{ijtk} \quad (3)$$

Nilai z_in_{jtk} ($(j = 1,2,3, \dots, p)$, $(t = 1,2,3, \dots, l)$, dan $(k = 1,2,3, \dots, m)$) diperoleh dengan mensubstitusikan x_{it} , v_{ijtk} , dan $b1_{jtk}$ ke dalam persamaan (3). Nilai z_in_{jtk} dari persamaan (3) diaktivasi untuk mendapatkan *output* dari unit lapisan tersembunyi dengan menggunakan persamaan sebagai berikut:

$$z_{jtk} = f(z_in_{jtk}) = \frac{1}{1+e^{-z_in_{jtk}}} \quad (4)$$

Nilai z_{jtk} ($(j = 1,2,3, \dots, p)$, $(t = 1,2,3, \dots, l)$, dan $(k = 1,2,3, \dots, m)$) diperoleh dengan mensubstitusikan nilai z_in_{jtk} dari persamaan (3) ke persamaan (4). Setiap sinyal di lapisan tersembunyi diteruskan ke semua unit di lapisan *output*.

- Setiap unit pada suatu lapisan *output* menerima sinyal dari lapisan tersembunyi dan diproses dengan menggunakan persamaan sebagai berikut:

$$y_in_{tk} = b2_{tk} + \sum_{j=1}^p z_{jtk} w_{jtk} \quad (5)$$

Nilai y_in_{tk} ($(t = 1,2,3, \dots, l)$ dan $(k = 1,2,3, \dots, m)$) diperoleh dengan mensubstitusikan nilai z_{jtk} pada persamaan (4), w_{jtk} , dan $b2_{tk}$ ke dalam persamaan (5). Nilai y_in_{tk} dari persamaan (5) diaktivasi untuk mendapatkan *output* dari unit lapisan *output* dengan menggunakan persamaan sebagai berikut:

$$y_{tk} = f(y_in_{tk}) \quad (6)$$

Backpropagation:

- Setiap unit *output* y_{tk} ($(t = 1,2,3, \dots, l)$ dan $(k = 1,2,3, \dots, m)$) menerima target pola yang berhubungan dengan pola *input* pembelajaran. Bobot-bobot baru yang menghubungkan antar lapisan tersembunyi dengan lapisan *output* diperoleh berdasarkan informasi error menggunakan persamaan sebagai berikut:

$$\delta_{tk} = (T_t - y_{tk}) f'(y_in_{tk}) \quad (7)$$

Nilai T_t dan y_{tk} pada persamaan (6) disubstitusikan ke dalam persamaan (7) untuk mendapatkan informasi error. Koreksi bobot yang digunakan untuk memperbaiki nilai w_{jtk} yaitu dengan persamaan sebagai berikut:

$$\Delta w_{jtk} = \alpha \delta_{tk} z_{jtk} \quad (8)$$

Nilai α , nilai δ_{tk} pada persamaan (7), dan nilai z_{jtk} pada persamaan (4) disubstitusikan ke dalam persamaan (8). Persamaan (8) tersebut digunakan untuk *epoch* ke-1 sedangkan untuk *epoch* lebih dari 1 menggunakan persamaan sebagai berikut:

$$\Delta w_{jtk} = mc (\Delta w_{jtk}(\text{epoch sebelumnya})) + \alpha \delta_{tk} z_{jtk} \quad (9)$$

Nilai mc , $\Delta w_{jtk}(\text{epoch sebelumnya})$, α , nilai δ_{tk} pada persamaan (7), dan nilai z_{jtk} pada persamaan (4) disubstitusikan ke dalam persamaan (9). Koreksi bias yang digunakan untuk memperbaiki nilai $b2_{tk}$ yaitu dengan persamaan sebagai berikut:

$$\Delta b2_{tk} = \alpha \delta_{tk} \quad (10)$$

Nilai δ_{tk} pada persamaan (7) dan α disubstitusikan ke dalam persamaan (10) untuk *epoch* ke-1, sedangkan untuk *epoch* lebih dari 1 menggunakan persamaan sebagai berikut:

$$\Delta b_{2_{tk}} = mc(\Delta b_{2_{tk}}(\text{epoch sebelumnya})) + \alpha \delta_{tk} \quad (11)$$

Nilai mc , $\Delta b_{2_{tk}}(\text{epoch sebelumnya})$, α , dan δ_{tk} pada persamaan (7) disubstitusikan ke dalam persamaan (11).

Langkah (a) pada tahap *backpropagation* ini dilakukan sebanyak jumlah lapisan tersembunyi, yaitu dengan menghitung informasi eror dari suatu lapisan tersembunyi ke lapisan tersembunyi sebelumnya.

- b. Tiap-tiap unit tersembunyi z_{jtk} ($(j = 1,2,3, \dots, p)$, $(t = 1,2,3, \dots, l)$ dan $(k = 1,2,3, \dots, m)$) mencari delta *input* untuk mendapatkan bobot-bobot baru yang menghubungkan antara lapisan *input* dengan lapisan tersembunyi dan diproses menggunakan persamaan sebagai berikut:

$$\delta_{in_{jtk}} = \sum_{j=1}^p \delta_{tk} w_{jtk} \quad (12)$$

Substitusikan nilai δ_{tk} pada persamaan (7) dan w_{jtk} ke dalam persamaan (12). Informasi eror yang digunakan untuk proses mendapatkan bobot-bobot baru diperoleh dengan menggunakan persamaan sebagai berikut:

$$\delta_{jtk} = \delta_{in_{jtk}} f'(z_{in_{jtk}}) \quad (13)$$

Substitusikan nilai $\delta_{in_{jtk}}$ pada persamaan (12) dan turunan dari fungsi aktivasi pada lapisan tersembunyi terhadap $z_{in_{jtk}}$ ke dalam persamaan (13). Koreksi bobot yang digunakan untuk memperbaiki nilai v_{ijtk} yaitu bobot-bobot yang menghubungkan antara lapisan input dan lapisan tersembunyi dengan persamaan sebagai berikut:

$$\Delta v_{ijtk} = \alpha \delta_{jtk} x_{it} \quad (14)$$

Substitusikan nilai α , nilai δ_{jtk} pada persamaan (13), dan x_{it} ke dalam persamaan (14) untuk *epoch* ke-1, sedangkan untuk *epoch* lebih dari 1 menggunakan persamaan sebagai berikut:

$$\Delta v_{ijtk} = mc(\Delta v_{ijtk}(\text{epoch sebelumnya})) + \alpha \delta_{jtk} x_{it} \quad (15)$$

Substitusikan nilai mc , $\Delta v_{ijtk}(\text{epoch sebelumnya})$, α , nilai δ_{jtk} pada persamaan (13), dan x_{it} ke dalam persamaan (15) tersebut. Koreksi bias yang digunakan untuk memperbaiki nilai $b_{1_{jtk}}$ yaitu bobot bias yang terhubung ke lapisan tersembunyi dengan menggunakan persamaan sebagai berikut:

$$\Delta b_{1_{jtk}} = \alpha \delta_{jtk} \quad (16)$$

Substitusikan nilai α dan nilai δ_{jtk} pada persamaan (13) ke dalam persamaan (16) tersebut untuk *epoch* ke-1, sedangkan untuk *epoch* lebih dari 1 menggunakan persamaan sebagai berikut:

$$\Delta b_{1_{jtk}} = mc(\Delta b_{1_{jtk}}(\text{epoch sebelumnya})) + \alpha \delta_{jtk} \quad (17)$$

Substitusikan nilai mc , $\Delta b_{1_{jtk}}(\text{epoch sebelumnya})$, α , dan nilai δ_{jtk} pada persamaan (13) ke dalam persamaan (17) tersebut.

- c. Tiap-tiap unit *output* y_{tk} ($(t = 1,2,3, \dots, l)$ dan $(k = 1,2,3, \dots, m)$) mencari bias dan bobot baru yang menghubungkan antara lapisan tersembunyi dan lapisan *output* dengan persamaan sebagai berikut:

$$w_{jtk}(\text{baru}) = w_{jtk}(\text{lama}) + \Delta w_{jtk} \quad (18)$$

$$b_{2_{tk}}(\text{baru}) = b_{2_{tk}}(\text{lama}) + \Delta b_{2_{tk}} \quad (19)$$

Bobot-bobot baru tersebut diperoleh dengan mensubstitusikan nilai $w_{jtk}(\text{lama})$, nilai Δw_{jtk} pada persamaan (8) atau nilai Δw_{jtk} pada persamaan (9), $b_{2_{tk}}(\text{lama})$, dan nilai $\Delta b_{2_{tk}}$ pada persamaan (10) ke dalam persamaan (18) dan ke dalam persamaan (19). Tiap-tiap unit tersembunyi z_{jtk} ($(j = 1,2,3, \dots, p)$, $(t = 1,2,3, \dots, l)$, dan $(k = 1,2,3, \dots, m)$) mencari bias dan bobot baru yang menghubungkan antara lapisan *input* dan lapisan tersembunyi dengan persamaan

$$v_{ijtk}(\text{baru}) = v_{ijtk}(\text{lama}) + \Delta v_{ijtk} \quad (20)$$

$$b_{1_{jtk}}(\text{baru}) = b_{1_{jtk}}(\text{lama}) + \Delta b_{1_{jtk}} \quad (21)$$

Bobot-bobot baru tersebut diperoleh dengan mensubstitusikan nilai $v_{ijtk}(\text{lama})$, nilai Δv_{ijtk} pada

persamaan (14) atau nilai Δv_{ijtk} pada persamaan (15), $b_{1jtk}(lama)$, dan nilai Δb_{1jtk} pada persamaan (16) atau nilai Δb_{1jtk} pada persamaan (17) ke dalam persamaan (20) dan ke dalam persamaan (21). Langkah dari persamaan (3) sampai persamaan (21) dilakukan sebanyak data yang digunakan untuk pembelajaran. Setelah langkah-langkah tersebut dikerjakan dilakukan perhitungan MSE dari suatu pembelajaran yang menunjukkan besarnya kesalahan rata-rata kuadrat, dengan rumus perhitungan sebagai berikut:

$$e_t = (T_t - y_{tk}) \tag{22}$$

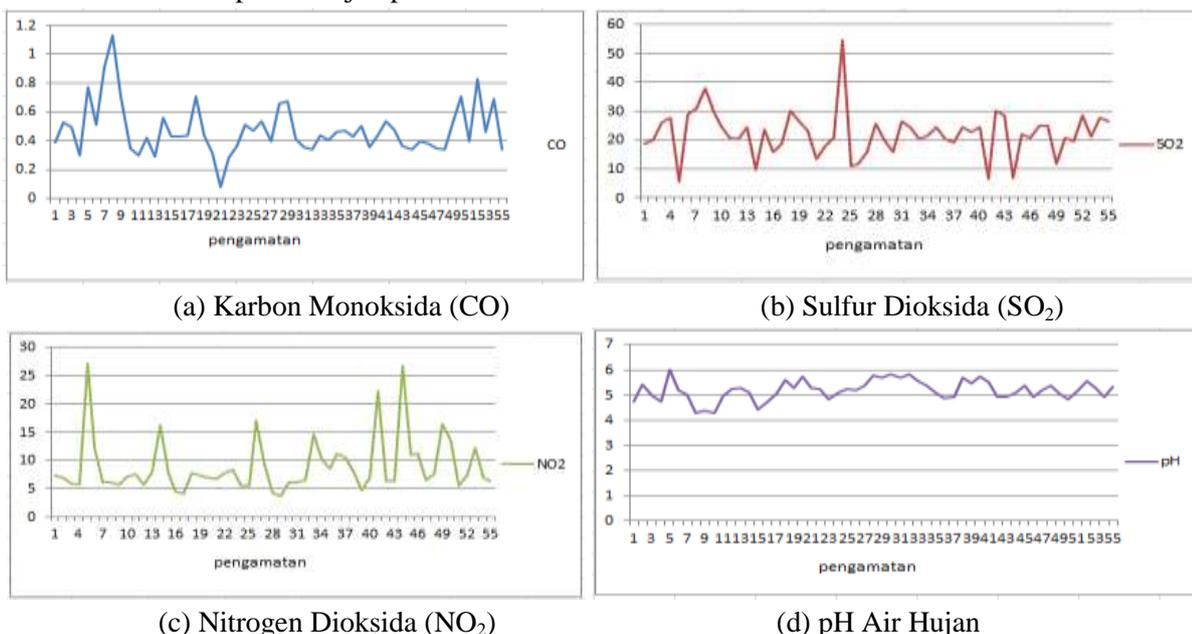
$$MSE = \frac{\sum_{t=1}^l e_t^2}{l} \tag{23}$$

dimana l menyatakan jumlah data dan e_t yaitu nilai kesalahan hasil ramalan. Dalam hal ini T_t adalah nilai data aktual yang merupakan target pembelajaran, dan y_{tk} adalah nilai ramalan [2][3]. Langkah-langkah pada persamaan (3) sampai (23) diatas dikerjakan selama *epoch* dalam pembelajaran kurang dari maksimum *epoch* yang ditetapkan dan MSE lebih besar dari target eror yang ditetapkan.

PREDIKSI pH AIR HUJAN DI KALIMANTAN BARAT DENGAN METODE BACKPROPAGATION

Tingkat keasaman air hujan dapat diketahui melalui pengukuran nilai pH dari air hujan tersebut. Dikatakan hujan asam apabila pH air hujan tersebut dibawah pH normal yaitu 5,6. Untuk dapat mengetahui tingkat keasaman suatu air hujan perlu juga diketahui faktor-faktor terjadinya hujan asam. Permasalahan yang dihadapi wilayah Kalimantan Barat saat ini adalah bagaimana memperkirakan pH air hujan dengan akurasi yang tinggi, dengan begitu wilayah tersebut dapat mengantisipasi menurunnya kesehatan masyarakat yang disebabkan hujan asam. Berdasarkan hal inilah yang menjadikan penentuan model pH air hujan di wilayah Kalimantan Barat sebagai kasus dalam penelitian ini. Data yang diperoleh yaitu emisi bahan bakar gas CO, SO₂, NO₂, dan pH air hujan.

Data gas CO, SO₂, NO₂, dan pH merupakan data bulanan dari Januari 2009 sampai November 2013. Data CO, SO₂, dan NO₂ diperoleh dari Badan Lingkungan Hidup Kota Pontianak sedangkan data pH diperoleh dari Badan Meteorologi, Klimatologi dan Geofisika (BMKG) Stasiun Klimatologi Siantan Pontianak. Pengambilan nilai CO, SO₂, dan NO₂ dilakukan setiap 30 menit dengan jumlah data yang diperoleh setiap hari yaitu 48 data. Masing-masing dari gas tersebut dicari nilai rata-rata perbulannya. Pengambilan Data pH air hujan tersebut dilakukan setiap 7 hari sekali dan kemudian dicari nilai rata-rata pH air hujan perbulan.



Gambar 2. Kadar Emisi Karbon Monoksida (CO), Sulfur Dioksida (SO₂), Nitrogen Dioksida (NO₂), dan pH Air Hujan Di Kota Pontianak Tahun 2009 Sampai 2013

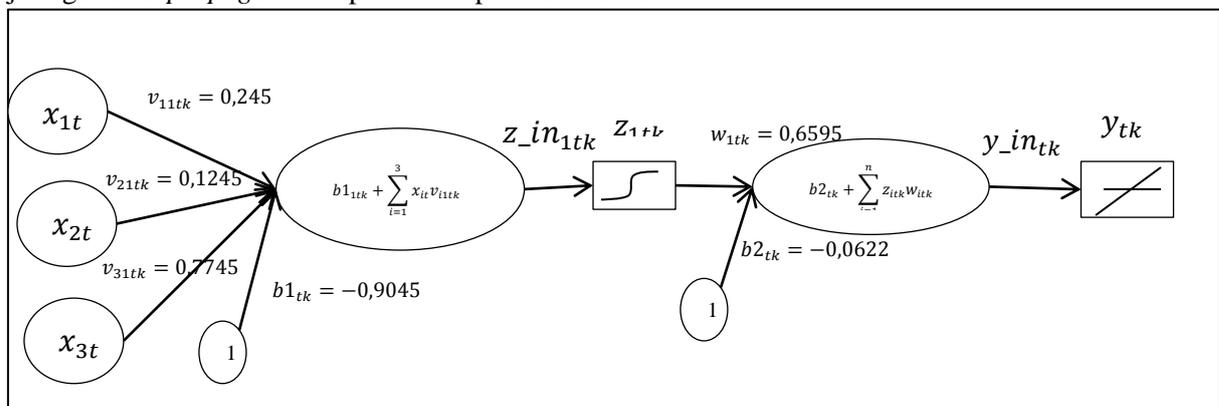
Gambar 2. merupakan data pengamatan dan kadar emisi CO, SO₂, NO₂, dan pH yang berada di udara Kota Pontianak dengan 55 pengamatan. Kadar emisi CO memiliki nilai rata-rata 0,47 mg/m³, kadar emisi terendah yaitu sebesar 0,079 mg/m³ dan kadar emisi tertinggi yaitu sebesar 1,132 mg/m³. kadar emisi SO₂ memiliki rata-rata 22,1436 μg/m³, data terendah yaitu sebesar 5,525 μg/m³, dan kadar emisi tertinggi yaitu sebesar 54,608 μg/m³. Kadar emisi NO₂ memiliki rata-rata 8,95407 μg/m³, kadar emisi terendah yaitu sebesar 3,562 μg/m³, dan data tertinggi yaitu sebesar 27,177 μg/m³. Data pH memiliki rata-rata 5,2, pH air hujan terendah yaitu sebesar 4,29, dan pH air hujan tertinggi yaitu sebesar 6,01. Deskriptif data dari keempat data tersebut di sajikan pada tabel 1.

Tabel 1. Deskriptif Data

	CO	SO ₂	NO ₂	pH
Rata-rata	0,47 mg/m ³	22,1436 μg/m ³	8,95407 μg/m ³	5,2
Minimum	0,079 mg/m ³	5,525 μg/m ³	3,562 μg/m ³	4,29
Maksimum	1,132 mg/m ³	54,608 μg/m ³	27,177 μg/m ³	6,01

Data pada pengamatan pertama sampai data pada pengamatan ke-45 tersebut digunakan untuk membentuk model pH air hujan dengan pembelajaran *backpropagation* berdasarkan fungsi *gradient descent* dengan momentum, sedangkan data pada pengamatan ke-46 sampai data pada pengamatan ke-55 digunakan untuk proses pengujian terhadap keakuratan model pH air hujan pada metode *backpropagation*.

Variabel yang terlibat dalam penentuan model prediksi pH air hujan (T_t) pada penelitian ini meliputi 3 variabel yaitu x_{1t} = karbon monoksida (CO), x_{2t} =sulfur dioksida (SO₂), dan x_{3t} =nitrogen dioksida (NO₂). Variabel x_{1t} , x_{2t} , dan x_{3t} sebagai *input* dan T_t sebagai target. Penelitian ini dilakukan sebanyak 40 percobaan untuk menentukan neuron di lapisan tersembunyi agar didapatkan suatu model pH air hujan terbaik berdasarkan MSE, MAD, dan MAPE. Percobaan pertama dengan arsitektur jaringan *backpropagation* dapat dilihat pada Gambar 3.



Gambar 3. Arsitektur Jaringan Backpropagation percobaan ke 1

Percobaan 1 pada arsitektur jaringan dengan satu neuron di lapisan tersembunyi, adapun langkah-langkahnya sebagai berikut:

Langkah 1: data yang digunakan untuk membuat model distandarisasi dengan menggunakan rata-rata dan simpangan baku.

Langkah 2: pembentukan jaringan syaraf *backpropagation* dengan 3 lapisan neuron, lapisan neuron ke 1 yaitu lapisan *input* terdiri dari 3 neuron, lapisan neuron ke 2 yaitu lapisan tersembunyi (*hidden layer*) terdiri dari 1 neuron dengan fungsi aktivasi sigmoid biner (logsig), dan lapisan neuron ke 3 yaitu lapisan *output* terdiri dari 1 neuron dengan fungsi aktivasi linear (purelin).

Langkah 3: penetapan bobot-bobot awal dilakukan secara acak dengan mengambil nilai yang umumnya terletak antara -0,5 sampai 0,5 atau -1 sampai 1.

Langkah 4: penetapan maksimum *epoch* = 5000, target eror = 0, momentum (*mc*) = 0.5, dan *learning rate* (α) = 0.01 yang digunakan dalam pembelajaran.

Langkah 5: melakukan pembelajaran berdasarkan persamaan (3) sampai (23) pada algoritma *backpropagation* terhadap data pada pengamatan pertama sampai data pengamatan ke-45 berdasarkan bobot-bobot yang baru. Perulangan berhenti pada *epoch* ke-5000 dengan MSE 0,92 pada data standarisasi dan diperoleh bobot akhir yaitu $v_{11(45)(5000)} = -0,0945$, $v_{21(45)(5000)} = -0,5259$, $v_{31(45)(5000)} = 0,5253$, $b_{1(45)(5000)} = -0,8975$, $w_{1(45)(5000)} = 1,2094$, $b_{2(45)(5000)} = -0,3715$. Bobot-bobot akhir yang diperoleh tersebut dibuat model pH air hujan untuk percobaan pertama yaitu sinyal-sinyal *input* x_{1t} , x_{2t} , dan x_{3t} dengan bobot akhir yang menghubungkan lapisan *input* dan lapisan tersembunyi disubstitusikan ke persamaan z_in_{1tk} berikut

$$z_in_{1tk} = -0,8975 + (-0,0945)(x_{1tk}) + (-0,5259)(x_{2tk}) + (0,5253)(x_{3tk})$$

Nilai z_in_{1tk} tersebut digunakan fungsi aktivasi sigmoid biner untuk menghitung sinyal lapisan tersembunyi dengan mensubstitusikan nilai z_in_{1tk} ke persamaan z_{1tk} berikut

$$z_{1tk} = f(z_in_{1tk}) = \frac{1}{1 + e^{-z_in_{1tk}}}$$

Penjumlahan sinyal-sinyal lapisan tersembunyi z_{1tk} dengan bobot akhir yang menghubungkan lapisan tersembunyi dan lapisan *output* disubstitusikan ke persamaan y_in_{tk} berikut

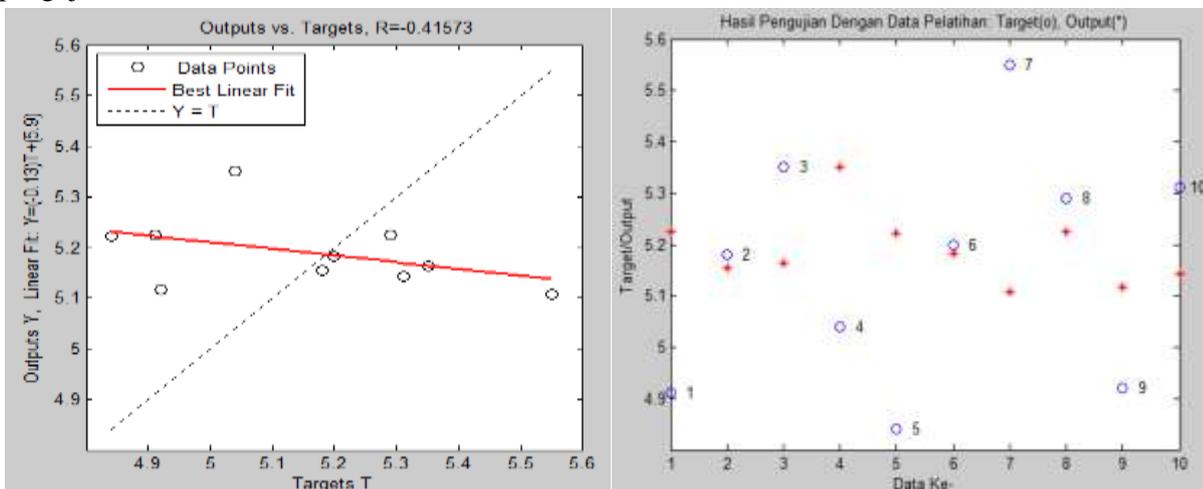
$$y_in_{tk} = -0,3715 + (1,2094)(z_{1tk})$$

Nilai y_in_{tk} tersebut digunakan fungsi aktivasi linier untuk menghitung sinyal lapisan *output* dengan mensubstitusikan nilai y_in_{tk} ke persamaan y_{tk} berikut

$$y_{tk} = f(y_in_{tk}) = y_in_{tk}$$

Langkah 6: dilakukan pengujian terhadap data-data baru yang tidak ikut dilatih namun terlebih dahulu data tersebut distandarisasi yaitu data dari pengamatan ke-46 sampai data pada pengamatan ke-55. Data yang telah distandarisasi digunakan untuk mencari variabel y_{tk} (prediksi pH air hujan di pengamatan ke-*t* pada *epoch* ke-*k*). Berdasarkan model pH air hujan untuk percobaan pertama dari pembelajaran menggunakan metode *backpropagation*, maka diperoleh MSE 0,06463, MAD 0,213, dan MAPE 0,041598 pada data pengujian.

Dari 40 percobaan terhadap arsitektur jaringan dalam menentukan neuron di lapisan tersembunyi untuk mendapatkan model terbaik prediksi pH air hujan. Dari penelitian tersebut diperoleh bahwa percobaan ke-2 dengan banyak neuron pada lapisan tersembunyi 2, *learning rate* 0,01, momentum 0,5, MSE pada pembelajaran 0,919, dan banyak *epoch* 5000 merupakan model terbaik pH air hujan *backpropagation* dengan MSE 0,06414, MAD 0,212, MAPE 0,041349, dan R -0,41573 pada data pengujian.



(a) Hubungan Target dan Output

(b) Perbandingan Target dan Output

Gambar 4. Hubungan dan Perbandingan antara Target dan Output Jaringan untuk Data Pengujian

Berdasarkan pembelajaran yang dilakukan terhadap data pH air hujan diperoleh bobot-bobot akhir yaitu $v_{11(45)(5000)} = -0,0609$, $v_{21(45)(5000)} = -0,4496$, $v_{31(45)(5000)} = 0,5529$, $v_{12(45)(5000)} = -0,3420$, $v_{22(45)(5000)} = 0,8509$, $v_{32(45)(5000)} = 0,3419$, $b_{11(45)(5000)} = -0,9347$, $b_{12(45)(5000)} = 0,7055$, $w_{1(45)(5000)} = 1,1795$, $w_{2(45)(5000)} = -0,252$, $b_{2(45)(5000)} = -0,1914$. Bobot ini dibentuk model pH air hujan yaitu untuk penjumlahan sinyal-sinyal *input* x_{1t} , x_{2t} , dan x_{3t} dengan bobot akhir yang menghubungkan lapisan *input* dan lapisan tersembunyi di substitusikan ke persamaan z_in_{1tk} dan z_in_{2tk} berikut

$$z_in_{1tk} = -0,9347 + (-0,0609)(x_{1tk}) + (-0,4496)(x_{2tk}) + (0,5529)(x_{3tk})$$

$$z_in_{2tk} = 0,7055 + (-0,3420)(x_{1tk}) + (0,8509)(x_{2tk}) + (0,3419)(x_{3tk})$$

Nilai z_in_{1tk} dan z_in_{2tk} tersebut digunakan fungsi aktivasi sigmoid biner untuk menghitung sinyal lapisan tersembunyi dengan mensubstitusikan nilai z_in_{1tk} dan z_in_{2tk} ke persamaan z_{1tk} dan z_{2tk} berikut

$$z_{1tk} = f(z_in_{1tk}) = \frac{1}{1 + e^{-z_in_{1tk}}}$$

$$z_{2tk} = f(z_in_{2tk}) = \frac{1}{1 + e^{-z_in_{2tk}}}$$

Sinyal-sinyal lapisan tersembunyi z_{1tk} dan z_{2tk} tersebut dengan bobot akhir yang menghubungkan lapisan tersembunyi dan lapisan *output* disubstitusikan ke persamaan y_in_{tk} berikut

$$y_in_{tk} = -0,1914 + (1,1795)(z_{1tk}) + (-0,252)(z_{2tk})$$

Nilai y_in_{tk} tersebut digunakan fungsi aktivasi linier untuk menghitung sinyal lapisan *output* dengan mensubstitusikan nilai y_in_{tk} ke persamaan y_{tk} berikut

$$y_{tk} = f(y_in_{tk}) = y_in_{tk}$$

PENUTUP

Dari hasil penelitian diperoleh model untuk prediksi pH air hujan terbaik berdasarkan pembelajaran menggunakan metode *backpropagation* untuk menentukan neuron pada lapisan tersembunyi yaitu penjumlahan sinyal-sinyal *input* x_{1t} , x_{2t} , dan x_{3t} dengan bobot akhir yang menghubungkan lapisan *input* dan lapisan tersembunyi di substitusikan ke persamaan z_in_{1tk} dan z_in_{2tk} berikut

$$z_in_{1tk} = -0,9347 + (-0,0609)(x_{1tk}) + (-0,4496)(x_{2tk}) + (0,5529)(x_{3tk})$$

$$z_in_{2tk} = 0,7055 + (-0,3420)(x_{1tk}) + (0,8509)(x_{2tk}) + (0,3419)(x_{3tk})$$

Nilai z_in_{1tk} dan z_in_{2tk} tersebut digunakan fungsi aktivasi sigmoid biner untuk menghitung sinyal lapisan tersembunyi dengan mensubstitusikan nilai z_in_{1tk} dan z_in_{2tk} ke persamaan z_{1tk} dan z_{2tk} berikut

$$z_{1tk} = f(z_in_{1tk}) = \frac{1}{1 + e^{-z_in_{1tk}}}$$

$$z_{2tk} = f(z_in_{2tk}) = \frac{1}{1 + e^{-z_in_{2tk}}}$$

Sinyal-sinyal lapisan tersembunyi z_{1tk} dan z_{2tk} tersebut dengan bobot akhir yang menghubungkan lapisan tersembunyi dan lapisan *output* disubstitusikan ke persamaan y_in_{tk} berikut

$$y_in_{tk} = -0,1914 + (1,1795)(z_{1tk}) + (-0,252)(z_{2tk})$$

Nilai y_in_{tk} tersebut digunakan fungsi aktivasi linier untuk menghitung sinyal lapisan *output* dengan mensubstitusikan nilai y_in_{tk} ke persamaan y_{tk} berikut

$$y_{tk} = f(y_in_{tk}) = y_in_{tk}$$

DAFTAR PUSTAKA

- [1]. Brady JE. *Kimia Universitas Asas dan Struktur*. Tangerang: Binarupa Aksara Publisher; 2008.
- [2]. Kusumadewi S. *Membangun Jaringan Syaraf Tiruan Menggunakan MATLAB & Excel Link*. Yogyakarta: Graha Ilmu; 2004.

- [3]. Kusumadewi S. *Artificial Intelligence (Teknik dan Aplikasinya)*. Yogyakarta: Graha Ilmu; 2003.
- [4]. Septiarini A. dan Sya'baniah N. *Sistem Peramalan Jumlah Produksi Air PDAM Samarinda Menggunakan Jaringan Syaraf Tiruan Backpropagation*. *Jurnal EKSPONENSIAL*. 2012; 3:43-48.
- [5]. Aprianti HF. Ihwan A. dan Jumarang MI. *Estimasi Curah Hujan Bulanan Kota Pontianak Dengan Suhu Permukaan Laut Sebagai Estimator Berdasarkan Jaringan Syaraf Tiruan Backpropagation*. *PRISMA FISIKA*. 2013; 3:16-19.

FRA SISKUS DIAN ARIANTO : FMIPA UNTAN, Pontianak, dian.arianto10@yahoo.com
MUHLASAH NOVITASARI MARA : FMIPA UNTAN, Pontianak, noveemara@gmail.com
NAOMI NESSYANA DEBATARAJA : FMIPA UNTAN, Pontianak, naominessyana@gmail.com
