

BAB III

METODE PENELITIAN DAN PERANCANGAN

Pada bab ini akan dibahas metode dan perancangan yang digunakan dalam penelitian tentang prediksi cuaca menggunakan metode *Al-Alaoui Backpropagation*. Langkah-langkah yang akan dilakukan dalam penelitian ini, yaitu :

1. Melakukan studi literatur yang berkaitan dengan unsur-unsur cuaca dan metode *Al-Alaoui Backpropagation*.
2. Melakukan pengumpulan data tentang unsur-unsur cuaca.
3. Menganalisa dan melakukan perancangan perangkat lunak prediksi cuaca.
4. Mengimplementasi dalam bentuk program berdasarkan analisis dan perancangan yang telah dilakukan.
5. Melakukan uji coba perangkat lunak untuk prediksi cuaca.
6. Melakukan evaluasi hasil prediksi terhadap perangkat lunak.

Berdasarkan langkah-langkah tersebut akan dibuat diagram alir seperti gambar 3.1



Gambar 3.1 Diagram Alir Penelitian

Sumber : [Metode Penelitian dan Perancangan]

3.1 Studi Literatur

Dalam penelitian ini dibutuhkan literatur-literatur yang dapat membantu dalam menyelesaikan masalah. Teori mengenai unsur-unsur cuaca serta metode *Al-Alaoui Backpropagation* yang digunakan dalam penelitian diperoleh dari jurnal-jurnal penelitian, buku-buku referensi, dan internet.

3.2 Data Penelitian

Data yang digunakan dalam penelitian ini adalah data klimatologi yang berasal dari Badan Meteorologi Klimatologi dan Geofisika Stasiun Klimatologi Karangploso. Data yang diambil mulai bulan Januari 2011 sampai Maret 2012. Data unsur cuaca yang diambil adalah : suhu, tekanan udara, kelembaban udara, dan kecepatan angin.

3.3 Analisis dan Perancangan Sistem

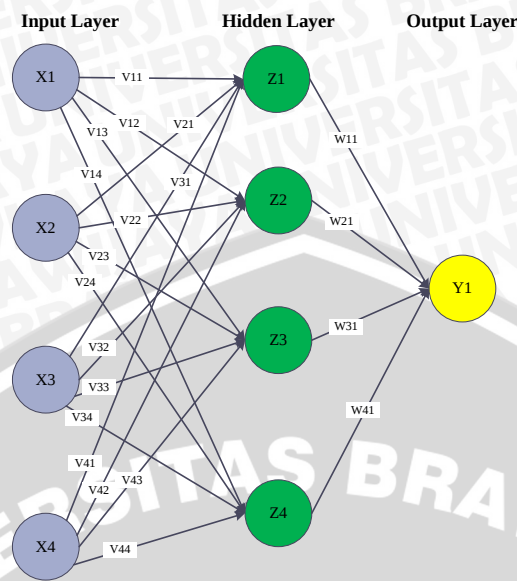
3.3.1 Deskripsi Sistem

Sistem yang akan dibuat dalam skripsi ini merupakan sistem yang digunakan untuk melakukan prediksi cuaca berdasarkan unsur-unsur cuaca yang meliputi suhu, tekanan udara, kelembaban udara, dan kecepatan angin. Hasil prediksi dari sistem ini berupa kondisi cuaca yang meliputi cuaca cerah, berawan, dan hujan.

Pada tahap awal, sistem harus memiliki data unsur-unsur cuaca. Data unsur-unsur cuaca yang menjadi *input* merupakan data-data numerik. Data kemudian dinormalisasi agar bisa digunakan dalam proses selanjutnya. Setelah data-data dinormalisasi, kemudian dilakukan proses Jaringan Syaraf Tiruan *Al-Alaoui Backpropagation*. Apabila hasil *output* jaringan tidak sama dengan *output* data maka akan dilakukan duplikasi, dan dilakukan proses JST lagi.

3.3.2 Arsitektur

Arsitektur sistem Jaringan Syaraf Tiruan yang akan digunakan dalam sistem ini dapat dilihat pada gambar 3.2.



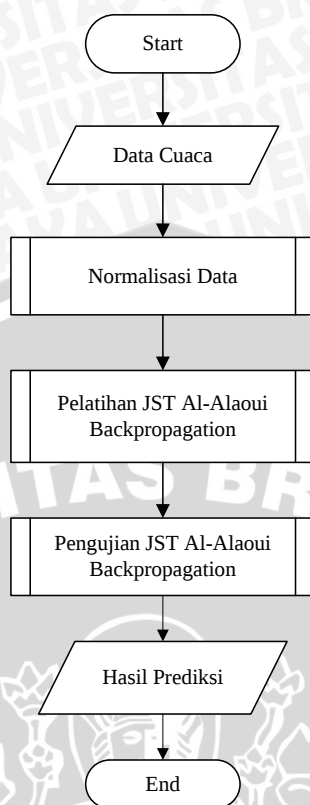
Gambar 3.2 Desain Arsitektur Jaringan Syaraf Tiruan

Sumber : [Perancangan]

Arsitektur Jaringan Syaraf Tiruan yang digunakan yaitu jaringan syaraf tiruan dengan satu *hidden layer*. Di dalam *input layer* terdapat empat buah *neuron input layer* sebagai parameter *input* yaitu x_1 (suhu), x_2 (tekanan udara), x_3 (kelembaban udara), x_4 (kecepatan angin). Sedangkan dalam *hidden layer* terdapat empat buah *neuron hidden layer* yaitu z_1 (*hidden layer* 1), z_2 (*hidden layer* 2), z_3 (*hidden layer* 3), z_4 (*hidden layer* 4). Serta satu buah *output layer* yaitu y_1 (kategori cuaca yang meliputi cerah, berawan, dan hujan).

3.3.3 Perancangan Proses

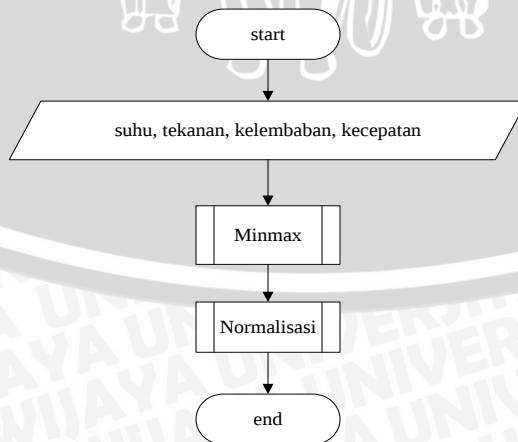
Tahapan proses dalam sistem prediksi cuaca ini yaitu pengambilan data dari unsur-unsur parameter cuaca yang digunakan antara lain : suhu, tekanan udara, kelembaban udara, dan kecepatan angin. Selanjutnya data-data tersebut dinormalisasi, kemudian data-data tersebut dilatihkan ke dalam jaringan syaraf tiruan *al-alaoui backpropagation*. Apabila kesalahan minimum telah terpenuhi maka sistem siap melakukan pengujian dengan menggunakan data latih. Pengujian tersebut akan menghasilkan *output* berupa keadaan cuaca yang terbagi dalam tiga kategori, yaitu cerah, berawan, hujan. Alur proses sistem secara umum dapat dilihat pada gambar 3.3.



Gambar 3.3 Diagram Alir Sistem
Sumber : [Perancangan]

3.3.3.1 Proses Normalisasi Data

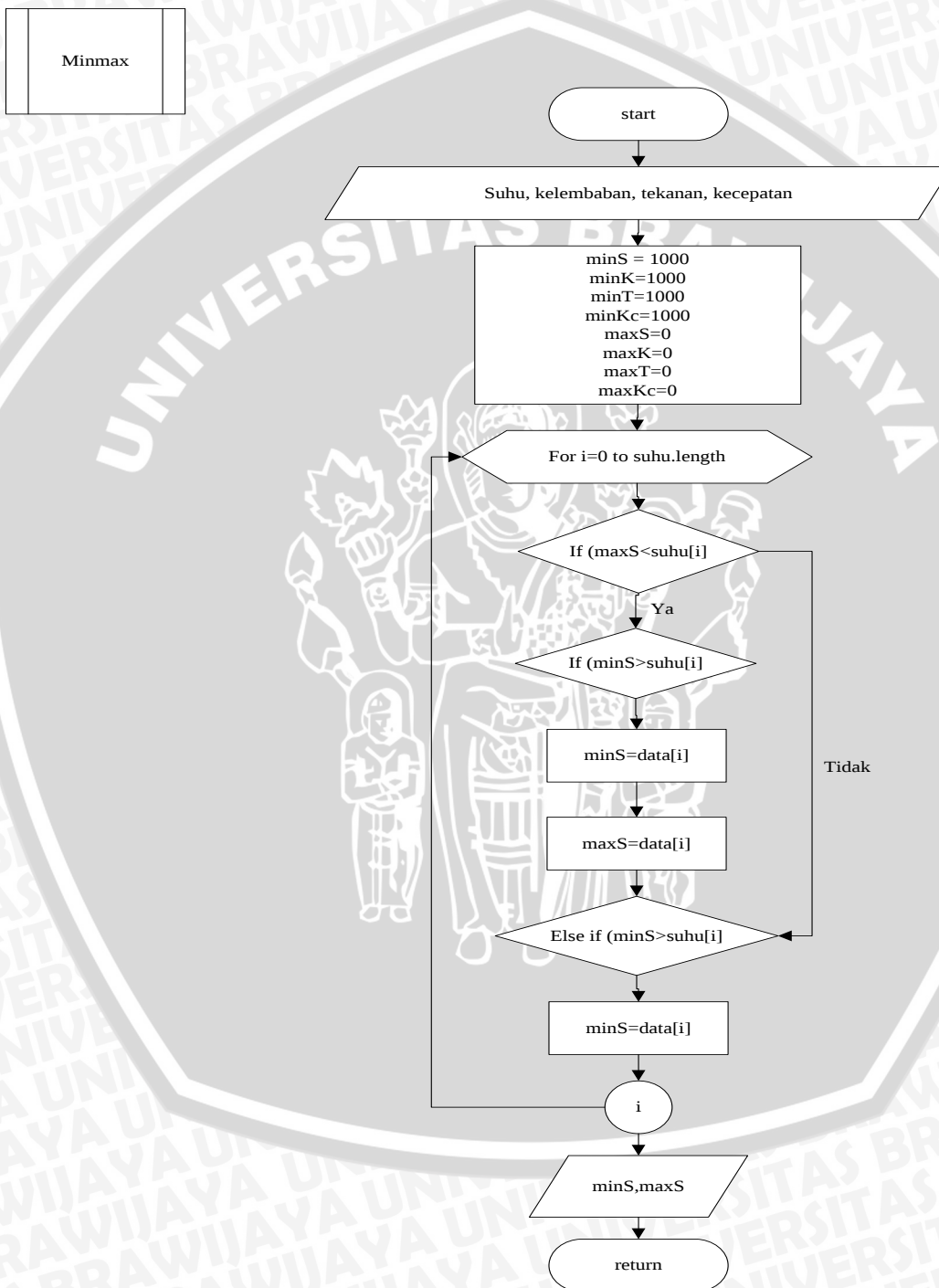
Proses normalisasi terdiri dari dua tahapan, yaitu : proses minmax dan proses normalisasi. Langkah-langkah normalisasi data ditunjukkan pada gambar 3.4.



Gambar 3.4 Diagram Alir Normalisasi Data
Sumber : [Perancangan]

3.3.3.2 Proses Minmax

Proses minmax merupakan proses untuk mencari nilai minimum dan maksimum data yang akan digunakan dalam proses normalisasi. Gambar 3.5 merupakan langkah-langkah proses minmax.

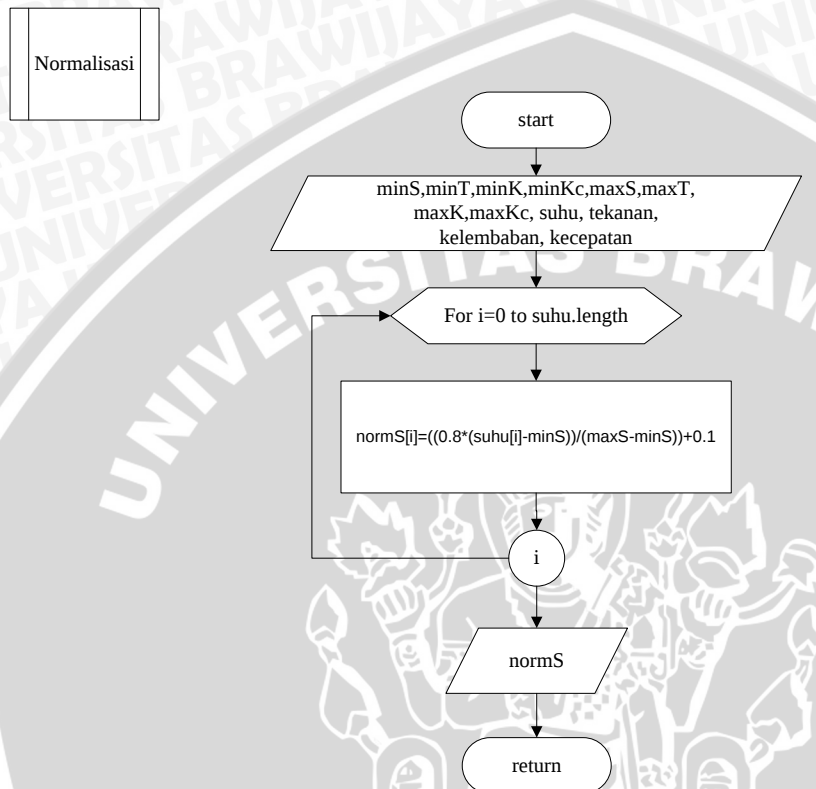


Gambar 3.5 Diagram Alir Minmax

Sumber : [Perancangan]

3.3.3.3 Proses Normalisasi

Proses ini bertujuan untuk mengubah data ke dalam domain fungsi aktivasi *sigmoid biner* langkah-langkah normalisasi data ditunjukkan pada gambar 3.6.



Gambar 3.6 Diagram Alir Normalisasi

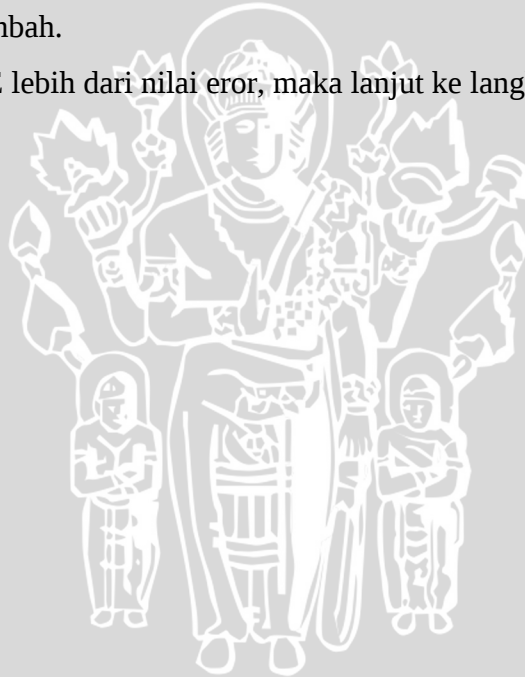
Sumber : [Perancangan]

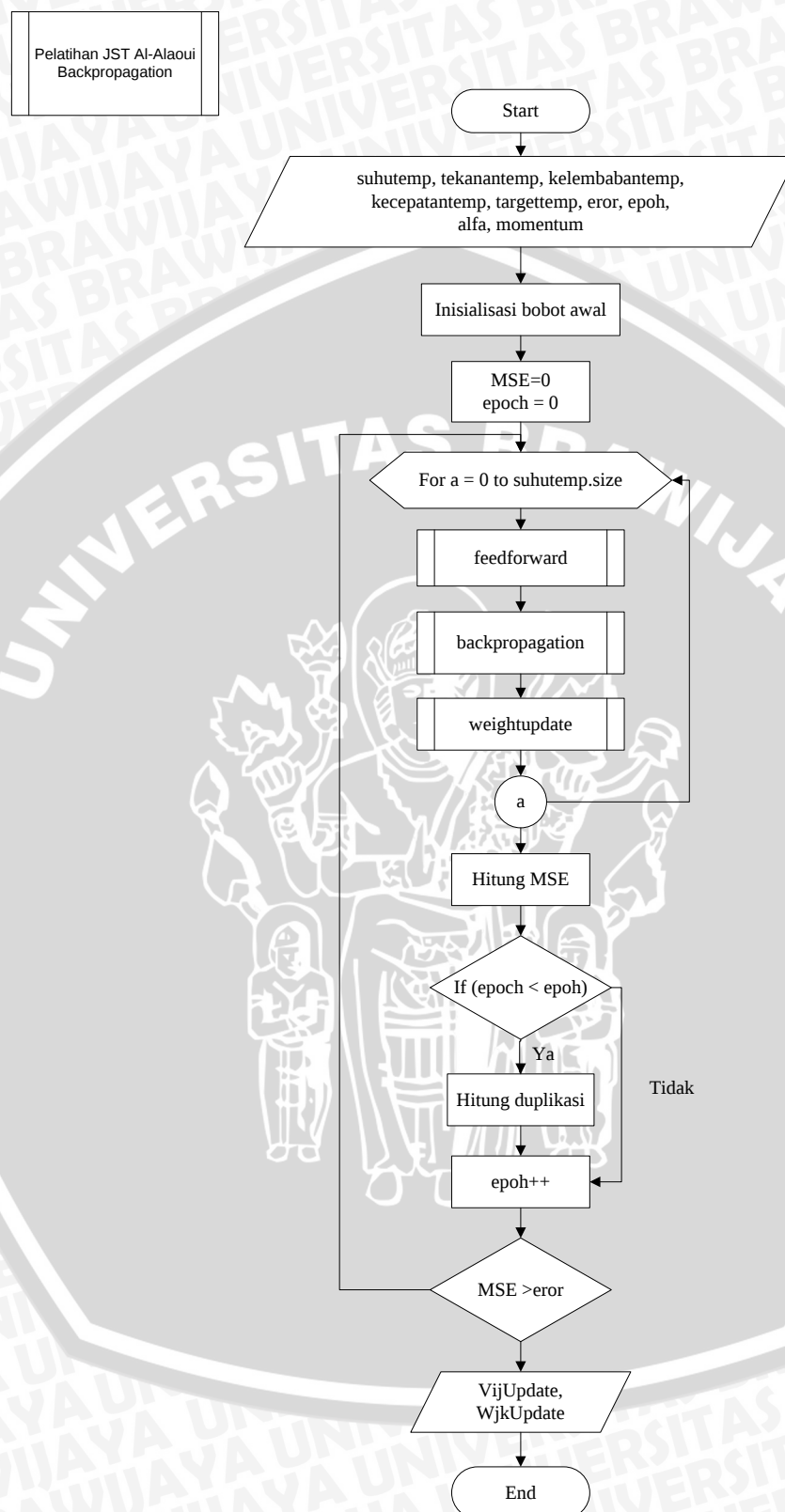
3.3.3.4 Proses Pelatihan JST Al-Alaoui Backpropagation

Algoritma *backpropagation* pada dasarnya terdiri dari tiga langkah, yaitu : *Feedforward*, *backpropagation*, dan *weight update*. Secara keseluruhan langkah-langkah pelatihan JST Al-Alaoui *Backpropagation* diilustrasikan pada gambar 3.7 dan langkah-langkahnya adalah sebagai berikut:

1. Mulai.
2. Masukkan data suhutemp, tekanantemp, kelembabantemp, kecepatanemp, targettemp, eror, epoh, alfa, momentum
3. Inisialisasi bobot awal secara *random*.
4. Atur parameter epoch awal sama dengan 0.

5. Lakukan perulangan untuk setiap a sama dengan 0 hingga jumlah data *input* yang dilatih.
6. Untuk setiap data *input*, lakukan langkah 7 hingga 15.
7. Lakukan proses *feedforward*.
8. Lakukan proses *backpropagation*.
9. Lakukan proses *weightupdate*.
10. Ulangi langkah 6 hingga 9 sampai nilai a sama dengan jumlah data *input*.
11. Hitung nilai MSE.
12. Apabila nilai epoch kurang dari nilai epoh, maka lanjut ke langkah 13. Namun apabila nilai epoch lebih dari nilai epoh, maka lanjut ke langkah 14.
13. Hitung duplikasi data.
14. Nilai epoch bertambah.
15. Apabila nilai MSE lebih dari nilai eror, maka lanjut ke langkah 5.



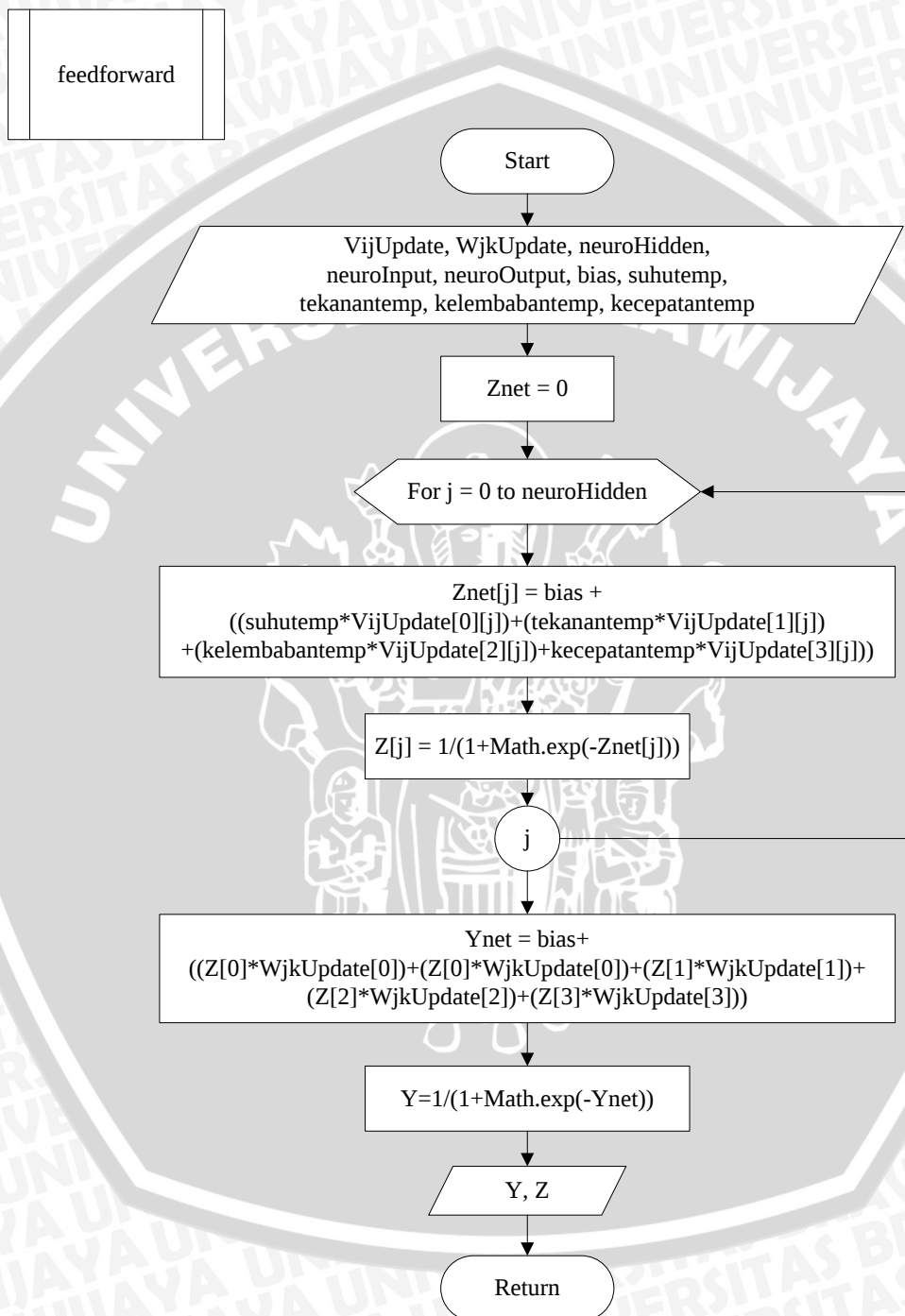


Gambar 3.7 Diagram Alir Pelatihan JST Al-Alaoui Backpropagation

Sumber : [Perancangan]

3.3.3.5 Proses *Feedforward*

Proses *Feedforward* merupakan tahap pertama dari pelatihan *backpropagation*. Proses *Feedforward* dapat dilihat pada gambar 3.8

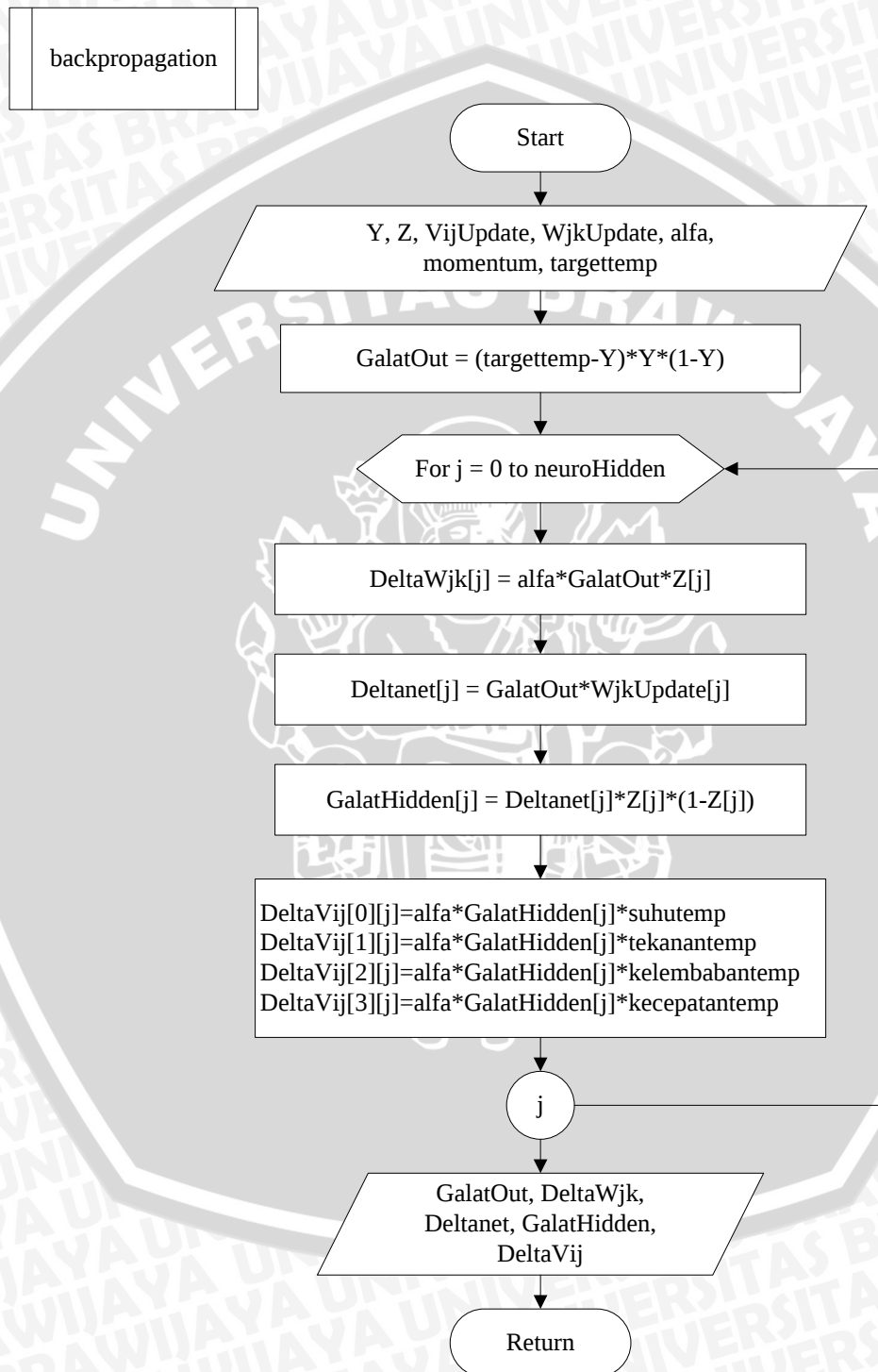


Gambar 3.8 Diagram Alir *feedforward*

Sumber : [Perancangan]

3.3.3.6 Proses *Backpropagation*

Proses *backpropagation* merupakan tahap kedua dalam pelatihan *backpropagation*. Proses *backpropagation* dapat dilihat pada gambar 3.9.

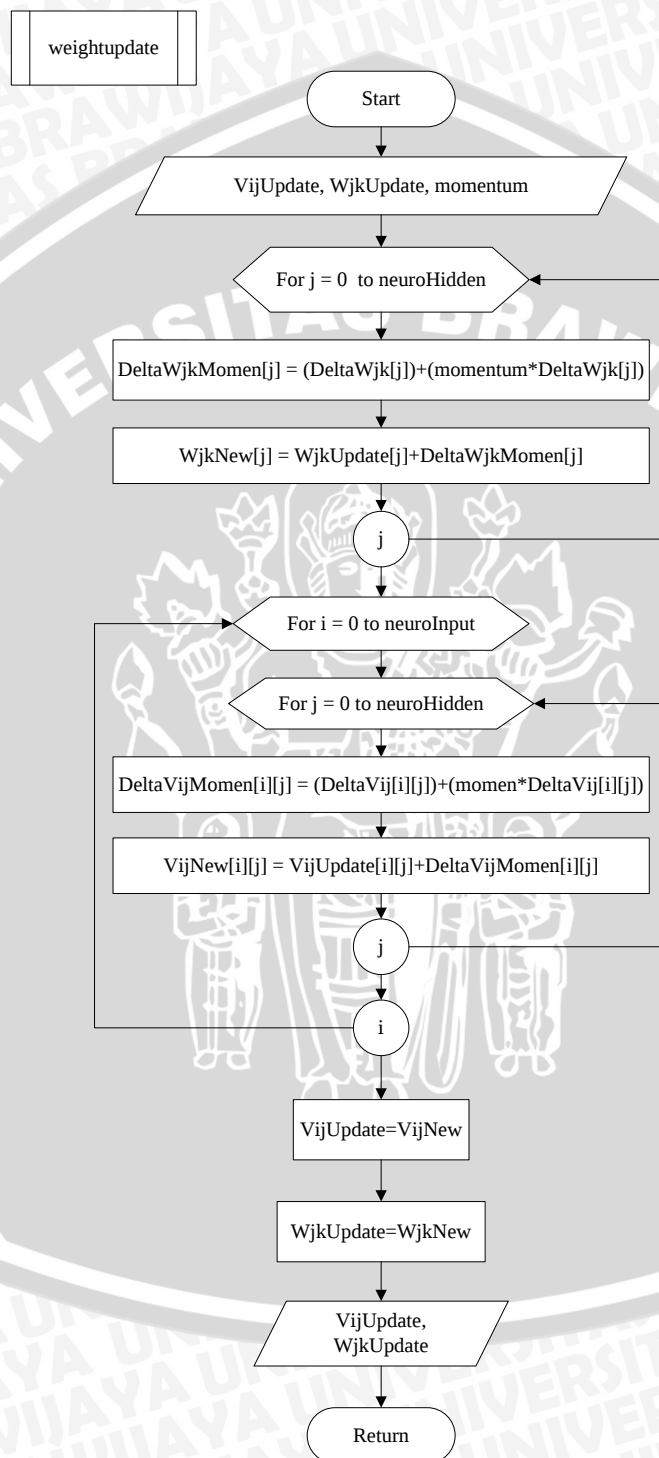


Gambar 3.9 Diagram Alir *backpropagation*

Sumber : [Perancangan]

3.3.3.7 Proses Weight Update

Proses *weight update* merupakan tahap ketiga dalam pelatihan *backpropagation*. Proses *weight update* dapat dilihat pada gambar 3.10.



Gambar 3.10 Diagram Alir *weightupdate*

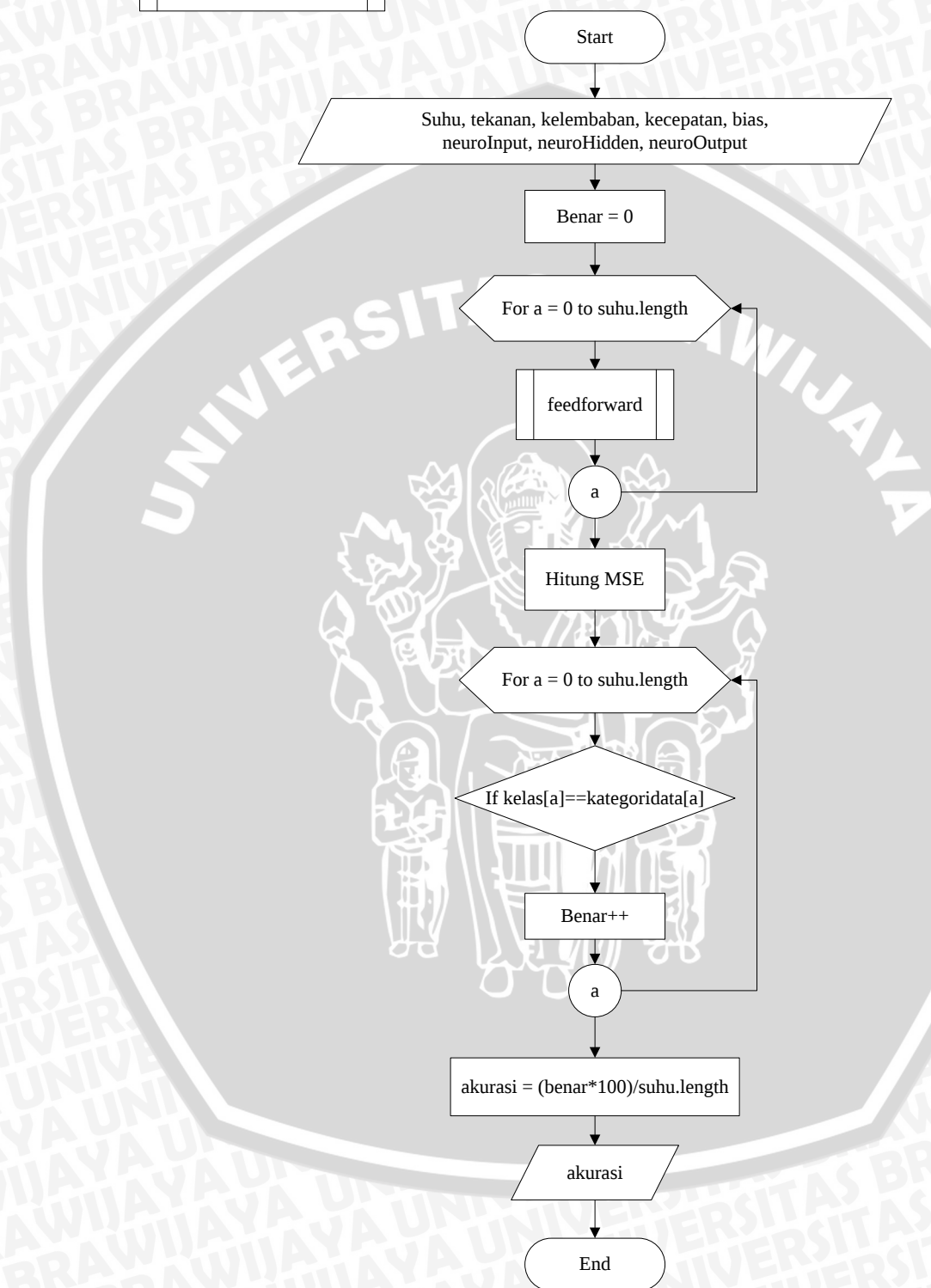
Sumber : [Perancangan]

3.3.3.8 Proses Pengujian JST *Al-Alaoui Backpropagation*

Proses pengujian JST *Al-Alaoui Backpropagation* dapat dilihat pada gambar 3.11 dan langkah-langkahnya adalah sebagai berikut :

1. Mulai.
2. Masukkan data cuaca yang ingin diuji, bias, jumlah *neuroHidden*, *neuroOutput*, *neuroInput*..
3. Inisialisasi awal nilai benar adalah 0.
4. Lakukan perulangan untuk setiap *a* sama dengan 0 hingga jumlah data yang diuji.
5. Untuk setiap data yang diuji, lakukan langkah 6 hingga 14.
6. Lakukan *Feedforward*.
7. Ulangi langkah 6 sampai nilai *a* sama dengan jumlah data yang diuji.
8. Hitung nilai MSE.
9. Lakukan perulangan untuk setiap *a* sama dengan 0 hingga jumlah data yang diuji.
10. Untuk setiap data yang diuji, lakukan langkah 11 hingga 14.
11. Apabila nilai kelas sama dengan kategori, maka lakukan langkah 12.
12. Nilai benar bertambah 1.
13. Lakukan perulangan untuk setiap *a* sama dengan 0 hingga jumlah data yang diuji.
14. Hitung akurasi data.

Pengujian JST Al-Alaoui
Backpropagation



Gambar 3.11 Diagram Alir Pengujian JST Al-Alaoui Backpropagation

Sumber : [Perancangan]

3.4 Perhitungan Manual

Dalam subbab ini akan menampilkan contoh perhitungan manual untuk prediksi cuaca dengan menggunakan Jaringan Syaraf *Al-Alaoui Backpropagation*.

3.4.1 Perhitungan Pelatihan Jaringan Syaraf Tiruan *Backpropagation*

Data cuaca yang digunakan menggunakan empat buah variabel *input*, yaitu: x_1 (suhu), x_2 (tekanan udara), x_3 (kelembaban udara), x_4 (kecepatan angin), dan t (kategori) sebagai target *output*. Berikut ini tabel 3.1 yang merupakan data yang akan digunakan.

Tabel 3.1 Data Cuaca

No	Suhu	Tekanan Udara	Kelembaban Udara	Kecepatan Angin	Target	Kategori
1	23.0	943.2	88	6	0.9	Hujan
2	24.8	945.3	75	3	0.9	Hujan
3	23.8	946.1	84	4	0.9	Hujan
4	24.2	944.5	81	5	0.6	Cerah
5	24.4	944.0	81	5	0.9	Hujan

Dari 5 data yang ada pertama-tama data akan dinormalisasi dengan persamaan 2.15. Hasil dari normalisasi data ditunjukkan pada tabel 3.2.

Tabel 3.2 Data Cuaca yang telah dinormalisasi

No	x_1	x_2	x_3	x_4	t
1	0.1	0.1	0.9	0.9	0.9
2	0.9	0.679310345	0.1	0.1	0.9
3	0.455555556	0.9	0.653846154	0.366666667	0.9
4	0.633333333	0.45862069	0.469230769	0.633333333	0.6
5	0.948695652	0.320689655	0.469230769	0.633333333	0.9

Setelah data dinormalisasi, kemudian ditentukan parameter awal yang digunakan dalam pelatihan, yaitu :

Learning rate = 0,1

Momentum = 0,5

Minimal Error = 0.01

Maksimal Duplikasi = 10

3.4.1.1 Proses Inisialisasi Bobot

Inisialisasi bobot-bobot dari *input layer* ke *hidden layer* mula-mula diberi nilai acak. Hasil inisialisasi bobot dari *input layer* ke *hidden layer* ditunjukkan pada tabel 3.3.

Tabel 3.3 Bobot antara *inputlayer* dengan *hiddenlayer* (v_{ij})

i	j			
	1	2	3	4
1	0.024831744	-0.06300484	0.05405725	-0.048277822
2	0.172158088	0.084678986	-0.023561685	0.030660433
3	0.055605825	-0.045611132	0.084625872	0.090552988
4	-0.012987446	-0.041231143	0.070805627	-0.13541528

Sama halnya dengan bobot-bobot antara *hidden layer* dengan *output layer* mula-mula diberi nilai acak antara 0 hingga 1. Hasil inisialisasi bobot antara *hidden layer* dengan *output layer* ditunjukkan pada tabel 3.4.

Tabel 3.4 Bobot antara *hiddenlayer* dengan *outputlayer* (w_{kj})

k	j			
	1	2	3	4
1	-0.223302631	-0.28987938	-0.165404641	-0.053623036

3.4.1.2 Proses *Feedforward*

Proses *Feedforward* berfungsi untuk menerima sinyal dari *input layer* dan meneruskannya pada *hidden layer*.

Menghitung nilai *hidden layer* (z_j)

Untuk menghitung nilai di *hidden layer* (z_j), pertama-tama menghitung nilai z_{netj} dengan persamaan 2.1. Hasil dari perhitungan z_{netj} dapat dilihat pada tabel 3.5.

Contoh perhitungan :

$$z_{netj} = 1 + \sum_{i=1}^n x_i \cdot v_{ji}$$

$$z_{net1} = 1 + ((0.1 \cdot 0.024831744048628) + (0.1 \cdot 0.172158088404691) + (0.9 \cdot 0.0556058249065137) + (0.9 \cdot -0.0129874461140024))$$

$$z_{net1} = 1.058055524$$

Tabel 3.5 Nilai z_{netj}

No	z_{netj}			
	1	2	3	4
1	1.058055524	0.924009368	1.142937905	0.957862199
2	1.14341217	0.991931795	1.048084617	0.972854694
3	1.196947457	1.001308793	1.084049006	1.014937813
4	1.111226216	0.949544884	1.106997106	0.939908282
5	1.095272222	0.917942143	1.127262459	0.920442101

Setelah nilai z_{netj} diketahui, kemudian dilanjutkan menghitung fungsi aktivasi dengan menggunakan persamaan 2.2. Hasil dari perhitungan z_j dapat dilihat pada tabel 3.6

Contoh perhitungan :

$$z_j = f(z_{netj}) = \frac{1}{1 + \exp(-z_{netj})}$$

$$z_1 = \frac{1}{1 + \exp(-1.058055524)}$$

$$z_1 = 0.742318778$$

Tabel 3.6 Nilai z_j

No	z_j			
	1	2	3	4
1	0.742318778	0.715858337	0.758218634	0.722693578
2	0.758305567	0.729469319	0.740406923	0.725688133
3	0.767981308	0.731315825	0.747259455	0.733985375
4	0.752357645	0.721023641	0.751568855	0.71908113
5	0.749373215	0.714622618	0.755333342	0.715132178

Menghitung nilai *output layer* (y_k)

Pertama-tama menghitung nilai y_{net_k} dengan persamaan 2.3 tanpa bias.

Hasil dari perhitungan y_{net_k} dapat dilihat pada tabel 3.7.

Contoh perhitungan :

$$y_{net_k} = 1 + \sum_{j=1}^p z_j \cdot w_{kj}$$

$$y_{net_1} = 1 + ((0.742318778 * (-0.22330263107321)) + (0.715858337 * (-0.289879379820957)) + (0.758218634 * (-0.165404640836926)) + (0.722693578 * -0.0536230358079067)))$$

$$y_{net_1} = 0.462559789$$

Tabel 3.7 Nilai y_{net_k}

No	y_{netk}
1	0.462559789
2	0.479940174
3	0.497879922
4	0.525418856
5	0.525119384

Setelah nilai y_{net_k} diketahui, kemudian dilanjutkan menghitung fungsi aktivasi dengan menggunakan persamaan 2.4. Hasil dari perhitungan y_k dapat dilihat pada tabel 3.8

Contoh perhitungan :

$$y_k = f(y_{\text{net}_k}) = \frac{1}{1 + \exp(-y_{\text{net}_k})}$$

$$y_k = \frac{1}{1 + \exp(-0.462559789)}$$

$$y_k = 0.613621253$$

Tabel 3.8 Nilai y_k

No	Y_k
1	0.613621253
2	0.617733748
3	0.621960976
4	0.628414001
5	0.628344068

3.4.1.3 Proses *Backpropagation*

Menghitung nilai galat *output* (δ_k)

Untuk menghitung δ_k digunakan persamaan 2.5. Hasil dari perhitungan δ_k dapat dilihat pada tabel 3.9.

Contoh perhitungan :

$$\delta_k = (t_k - y_k) f'(y_{\text{net}_k}) = (t_k - y_k) y_k (1 - y_k)$$

$$\delta_1 = (0,9 - 0.613621253) * 0.613621253 * (1 - 0.613621253)$$

$$\delta_1 = 0.067897597$$

Tabel 3.9 Nilai δ_k

No	δ_k
1	0.067897597
2	0.066654004
3	0.06537407
4	-0.006634949
5	0.063439212

Menghitung koreksi bobot (Δw_{kj})

Untuk menghitung koreksi bobot digunakan persamaan 2.6. hasil dari perhitungan Δw_{kj} dapat dilihat pada tabel 3.10.

Contoh perhitungan :

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

$$\Delta w_{11} = 0,1 * 0.067897597 * 0.742318778$$

$$\Delta w_{11} = 0.005040166$$

Tabel 3.10 Nilai Δw_{kj}

No	j			
	1	2	3	4
1	0.005040166	0.004860506	0.005148122	0.004906916
2	0.00505441	0.004862205	0.004935109	0.004837002
3	0.005020606	0.004780909	0.004885139	0.004798361
4	-0.000499185	-0.000526235	-0.000498662	-0.000477107
5	0.004753965	0.00453351	0.004791775	0.004536742

Menghitung nilai galat hidden layer (δ_j)

Untuk menghitung δ_j pertama-tama hitung δ_{netj} terlebih dahulu dengan menggunakan persamaan 2.7. Hasil perhitungan δ_j dapat dilihat pada tabel 3.11.

Contoh perhitungan :

$$\delta_{netj} = \sum_{k=1}^m \delta_k \cdot w_{kj}$$

$$\delta_{net_1} = 0.067897597 * -0.22330263107321$$

$$\delta_{net_1} = -0.015161712$$

Tabel 3.11 Nilai δ_{netj}

No	j			
	1	2	3	4
1	-0.015161712	-0.019682113	-0.011230578	-0.003640875
2	-0.014380094	-0.018835663	-0.010510167	-0.003083592
3	-0.013608317	-0.017997175	-0.009824402	-0.002550057
4	0.001331169	0.001778989	0.00094848	0.000211055
5	-0.012775302	-0.017059648	-0.00911622	-0.002063378

Setelah nilai δ_{netj} diketahui, kemudian dilanjutkan menghitung δ_j dengan menggunakan persamaan 2.8. Hasil dari perhitungan δ_j dapat dilihat pada tabel 3.12.

Contoh perhitungan :

$$\delta_j = \delta_{netj} \cdot f'(z_{netj}) = \delta_{netj} \cdot z_j(1 - z_j)$$

$$\delta_1 = -0.015161712 * 0.742318778 * (1 - 0.742318778)$$

$$\delta_1 = -0.002900157$$

Tabel 3.12 Nilai δ_j

No	j			
	1	2	3	4
1	-0.002900157	-0.004003444	-0.002058825	-0.000729659
2	-0.002635558	-0.003717102	-0.002020102	-0.000613835
3	-0.002424812	-0.003536319	-0.001855464	-0.000497901
4	0.000248018	0.000357841	0.000177094	4.26339E-05
5	-0.002399368	-0.003479096	-0.001684722	-0.000420348

Menghitung koreksi bobot (Δv_{ij})

Untuk menghitung koreksi bobot digunakan persamaan 2.9. hasil dari perhitungan Δv_{ij} dapat dilihat pada tabel 3.13.

Contoh perhitungan :

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

$$\Delta v_{11} = 0,1 * -0.002900157 * 0.1$$

$$\Delta v_{11} = -2.90016E-05$$

Tabel 3.13 Nilai Δv_{ij}

No	i	j			
		1	2	3	4
1	1	-2.90016E-05	-4.00344E-05	-2.05882E-05	-7E-06
	2	-2.90016E-05	-4.00344E-05	-2.05882E-05	-7E-06
	3	-0.000261014	-0.00036031	-0.000185294	-7E-05
	4	-0.000261014	-0.00036031	-0.000185294	-7E-05

2	1	-0.0002372	-0.000334539	-0.000181809	-6E-05
	2	-0.000179036	-0.000252507	-0.000137228	-4E-05
	3	-2.63556E-05	-3.7171E-05	-2.0201E-05	-6E-06
	4	-2.63556E-05	-3.7171E-05	-2.0201E-05	-6E-06
3	1	-0.000110464	-0.000161099	-8.45267E-05	-2E-05
	2	-0.000218233	-0.000318269	-0.000166992	-4E-05
	3	-0.000158545	-0.000231221	-0.000121319	-3E-05
	4	-8.89098E-05	-0.000129665	-6.80337E-05	-2E-05
4	1	1.57078E-05	2.26633E-05	1.12159E-05	2.7E-06
	2	1.13746E-05	1.64113E-05	8.12188E-06	2E-06
	3	1.16377E-05	1.6791E-05	8.30978E-06	2E-06
	4	1.57078E-05	2.26633E-05	1.12159E-05	2.7E-06
5	1	-0.000227627	-0.00033006	-0.000159829	-4E-05
	2	-7.69452E-05	-0.000111571	-5.40273E-05	-1E-05
	3	-0.000112586	-0.00016325	-7.90523E-05	-2E-05
	4	-0.00015196	-0.000220343	-0.000106699	-3E-05

3.4.1.4 Proses Weight Update

Menghitung perubahan bobot garis baru menuju *output layer* (w_{kj})

Untuk menghitung w_{kj} , pertama-tama menghitung $\Delta w_{kj}(t)$ dengan variasi *backpropagation*, yaitu momentum. Persamaan yang digunakan adalah persamaan 2.12. Hasil dari perhitungan $\Delta w_{kj}(t)$ dapat dilihat pada tabel 3.14.

Contoh perhitungan:

$$\Delta w_{kj}(t+1) = \alpha \delta_k z_j + \mu \Delta w_{kj}(t)$$

$$\Delta w_{11}(t+1) = (0,1 * 0.067897597 * 0.742318778) + (0,5 * 0.004991634)$$

$$\Delta w_{11}(t+1) = 0.007560249$$

Tabel 3.14 Nilai $w_{kj}(t)$

No	j			
	1	2	3	4
1	0.007560249	0.007290759	0.007722184	0.007360374
2	0.007581615	0.007293308	0.007402663	0.007255503
3	0.00753091	0.007171364	0.007327709	0.007197542
4	-0.000748778	-0.000789353	-0.000747993	-0.00071566
5	0.007130947	0.006800264	0.007187663	0.006805113

Setelah nilai w_{kj} (t) diketahui, kemudian dilanjutkan menghitung w_{kj} dengan menggunakan persamaan 2.10. Hasil dari perhitungan w_{kj} dapat dilihat pada tabel 3.15.

Contoh perhitungan:

$$w_{kj}(\text{baru}) = w_{kj}(\text{lama}) + w_{kj}(t)$$

$$w_{11}(\text{baru}) = -0.22330263107321 + 0.00898494$$

$$w_{11}(\text{baru}) = -0.215742382$$

Tabel 3.15 Nilai w_{kj}

No	j			
	1	2	3	4
1	-0.215742382	-0.282588621	-0.157682457	-0.046262662
2	-0.208160766	-0.275295313	-0.150279794	-0.039007159
3	-0.200629857	-0.268123949	-0.142952086	-0.031809617
4	-0.201378635	-0.268913302	-0.143700079	-0.032525277
5	-0.194247688	-0.262113037	-0.136512416	-0.025720164

Menghitung perubahan bobot garis baru menuju *hidden layer* (v_{ij})

Untuk menghitung v_{ij} , pertama-tama menghitung Δv_{ij} (t) dengan variasi backpropagation, yaitu momentum. Persamaan yang digunakan adalah persamaan 2.13. Hasil dari perhitungan Δv_{ij} (t) dapat dilihat pada tabel 3.16.

Contoh perhitungan :

$$\Delta v_{ij}(t+1) = \alpha \delta_j x_i + \mu \Delta v_{ij}(t)$$

$$\Delta v_{11}(t+1) = (0.1 * -0.002900157 * 0.1) + (0.5 * 0.000150093)$$

$$\Delta v_{11}(t+1) = -4.35024E-05$$

Tabel 3.16 Nilai v_{ij} (t)

No	i	j			
		1	2	3	4
1	1	-4.35024E-05	-6.00517E-05	-3.08824E-05	-1E-05
	2	-4.35024E-05	-6.00517E-05	-3.08824E-05	-1E-05
	3	-0.000391521	-0.000540465	-0.000277941	-1E-04
	4	-0.000391521	-0.000540465	-0.000277941	-1E-04

2	1	-0.0003558	-0.000501809	-0.000272714	-8E-05
	2	-0.000268554	-0.00037876	-0.000205841	-6E-05
	3	-3.95334E-05	-5.57565E-05	-3.03015E-05	-9E-06
	4	-3.95334E-05	-5.57565E-05	-3.03015E-05	-9E-06
3	1	-0.000165695	-0.000241648	-0.00012679	-3E-05
	2	-0.00032735	-0.000477403	-0.000250488	-7E-05
	3	-0.000237818	-0.000346831	-0.000181978	-5E-05
	4	-0.000133365	-0.000194498	-0.000102051	-3E-05
4	1	2.35617E-05	3.39949E-05	1.68239E-05	4.1E-06
	2	1.70619E-05	2.4617E-05	1.21828E-05	2.9E-06
	3	1.74566E-05	2.51865E-05	1.24647E-05	3E-06
	4	2.35617E-05	3.39949E-05	1.68239E-05	4.1E-06
5	1	-0.00034144	-0.00049509	-0.000239743	-6E-05
	2	-0.000115418	-0.000167356	-8.10409E-05	-2E-05
	3	-0.000168879	-0.000244875	-0.000118579	-3E-05
	4	-0.00022794	-0.000330514	-0.000160049	-4E-05

Setelah nilai $v_{ij}(t)$ diketahui, kemudian dilanjutkan menghitung v_{ij} dengan menggunakan persamaan 2.11. Hasil dari perhitungan v_{ij} dapat dilihat pada tabel 3.17.

Contoh perhitungan :

$$v_{ji}(\text{baru}) = v_{ji}(\text{lama}) + v_{ji}(t)$$

$$v_{11}(\text{baru}) = 0.024831744048628 + -4.35024E-05$$

$$v_{11}(\text{baru}) = 0.024788242$$

Tabel 3.17 Nilai v_{ij}

No	i	j			
		1	2	3	4
1	1	0.024788242	-0.063064891	0.054026367	-0.0483
	2	0.172114586	0.084618934	-0.023592568	0.03065
	3	0.055214304	-0.046151597	0.084347931	0.09045
	4	-0.013378967	-0.041771608	0.070527685	-0.1355
2	1	0.024432441	-0.0635667	0.053753653	-0.0484
	2	0.171846032	0.084240174	-0.023798409	0.03059
	3	0.05517477	-0.046207353	0.084317629	0.09045
	4	-0.013418501	-0.041827364	0.070497384	-0.1355

3	1	0.024266746	-0.063808349	0.053626863	-0.0484
	2	0.171518682	0.083762771	-0.024048897	0.03052
	3	0.054936952	-0.046554184	0.084135651	0.0904
	4	-0.013551865	-0.042021862	0.070395333	-0.1356
4	1	0.024290308	-0.063774354	0.053643687	-0.0484
	2	0.171535744	0.083787388	-0.024036714	0.03052
	3	0.054954409	-0.046528998	0.084148116	0.0904
	4	-0.013528304	-0.041987867	0.070412157	-0.1355
5	1	0.023948867	-0.064269444	0.053403944	-0.0485
	2	0.171420326	0.083620032	-0.024117755	0.0305
	3	0.05478553	-0.046773873	0.084029537	0.09037
	4	-0.013756244	-0.042318381	0.070252109	-0.1356

Menghitung MSE

MSE dihitung dengan persamaan 2.17.

Contoh Perhitungan :

$$MSE = \frac{1}{n} \sum_{t=1}^n (Y_t - Y'_t)^2$$

$$MSE = \frac{1}{5} * ((0,9 - 0.613621253190826)^2 + (0,9 - 0.617733747677569)^2 + (0,9 - 0.621960975726803)^2 + (0,6 - 0.628414000581653)^2 + (0,9 - 0.628344068221948)^2)$$

$$MSE = 0.062719405$$

Karena nilai MSE lebih besar daripada nilai minimal *error* yaitu 0,01 maka perulangan akan dilakukan sampai nilai MSE kurang dari minimal *error*. Setelah menghitung nilai MSE, langkah selanjutnya yaitu melakukan duplikasi data. Sebelum melakukan proses duplikasi, pertama-tama *output* jaringan diklasifikasikan ke dalam *range* kategori yang telah ditentukan seperti tabel 3.18. *Range* kategori diperoleh dari perhitungan rata-rata kelas. Kemudian *output* jaringan dibandingkan dengan *output* data yang ada seperti pada tabel 3.19. Apabila terdapat *misclassification* data yang salah akan dilakukan duplikasi.

Tabel 3.18 Range Kategori

Output Jaringan	Kategori
$y_k > 0,6$	Hujan
$0,3 < y_k \leq 0,6$	Cerah
$y_k \leq 0,3$	Berawan

Tabel 3.19 Perbandingan *Output* Jaringan dan *Output* Data

No	Output Jaringan	Kategori	Output Data	Kategori
1	0.613621253190826	Hujan	0.9	Hujan
2	0.617733747677569	Hujan	0.9	Hujan
3	0.621960975726803	Hujan	0.9	Hujan
4	0.628414000581653	Hujan	0.6	Cerah
5	0.628344068221948	Hujan	0.9	Hujan

Karena terdapat *misclassification* data seperti yang terlihat pada tabel 3.19, maka dilakukan duplikasi data sampai jumlah minimal *error* terpenuhi.

Proses Duplikasi Data

Data yang salah dalam klasifikasi akan ditempatkan ke dalam index terakhir. Seperti tabel 3.19 terlihat bahwa data ke empat mengalami kesalahan dalam klasifikasi data, maka data ke empat akan di duplikasi dan ditempatkan ke dalam index terakhir. Data yang akan diduplikasi ditunjukkan pada tabel 3.20.

No	x1	x2	x3	x4	t
1	0.407692308	0.325641026	0.751162791	0.536363636	0.9
2	0.629230769	0.541025641	0.509302326	0.318181818	0.9
3	0.506153846	0.623076923	0.676744186	0.390909091	0.9
4	0.555384615	0.458974359	0.620930233	0.463636364	0.6
5	0.58	0.407692308	0.620930233	0.463636364	0.9
6	0.555384615	0.458974359	0.620930233	0.463636364	0.6

Proses akan diulangi dari *feedforward*, *backpropagation*, *weight update*, dan duplikasi hingga nilai MSE kurang dari nilai minimal *error*.

3.5 Perancangan Interface

Rancangan *interface* prediksi cuaca terdiri dari 2 *form*, yaitu *form* pelatihan dan *form* pengujian. Untuk *form* pelatihan dapat dilihat pada gambar 3.12, dan *form* pengujian dapat dilihat pada gambar 3.13.

Form Pelatihan

Gambar 3.12 Interface Form Pelatihan

Penjelasan gambar 3.12 :

1. Memilih data yang akan digunakan dalam pelatihan.
2. Menampilkan alamat data yang digunakan dalam pelatihan.
3. Melakukan *load* data yang akan digunakan.
4. Menampilkan data pelatihan yang digunakan.
5. Melakukan proses normalisasi data yang digunakan dalam pelatihan.
6. Menampilkan data yang telah dinormalisasi.
7. Mengisi *minimal error* yang digunakan.
8. Mengisi *maksimal duplikasi* yang akan digunakan.
9. Mengisi *learning rate* yang digunakan.
10. Mengisi *momentum* yang akan digunakan dalam pelatihan.
11. Melakukan proses pelatihan.
12. Menampilkan bobot baru (v_{ij}) yang terbentuk setelah dilakukan pelatihan.

13. Menampilkan bobot baru (w_{kj}) yang terbentuk setelah dilakukan pelatihan.
14. Menampilkan *output* jaringan setelah dilakukan pelatihan.
15. Menampilkan nilai MSE dan jumlah duplikasi data setiap *epoch*.

Form Pengujian

PREDIKSI CUACA MENGGUNAKAN METODE AL-ALAOUI BACKPROPAGATION

PELATIHAN **PENGUJIAN**

LOAD DATA

Title 1	Title 2	Title 3	Title 4	Title 5	Title 6	Title 7	Title 8

NORMALISASI

Title 1	Title 2	Title 3	Title 4	Title 5	Title 6	Title 7	Title 8

PROSES

Nilai MSE

Hasil Akurasi Data

Pengujian JST

Output

Title 1	Title 2	Title 3

Output Data dan Output Jaringan

Title 1	Title 2	Title 3	Title 4	Title 5	Title 6

Gambar 3.13 Interface Form Pengujian

Penjelasan Gambar 3.13 :

1. Memilih data yang akan digunakan dalam pengujian.
2. Menampilkan alamat data yang digunakan dalam pengujian.
3. Melakukan *load* data yang akan digunakan dalam pengujian.
4. Menampilkan data yang digunakan dalam pengujian.
5. Melakukan normalisasi data.
6. Menampilkan data yang telah dinormalisasi.
7. Melakukan proses pengujian.
8. Menampilkan *output* jaringan.

9. Menampilkan perbandingan antara *output* jaringan dengan *output* data.
10. Menampilkan nilai MSE dari pengujian.
11. Menampilkan hasil tingkat akurasi dalam pengujian.

3.6 Rancangan Pengujian

Pada bagian ini akan dijelaskan mengenai pengujian data. Dalam pengujian ini menggunakan data 6 bulan (Januari hingga Juni 2011) sebagai data pelatihan dengan interval waktu 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 bulan dan beberapa data uji, yaitu data bulan Juli, bulan Juli hingga Agustus, bulan Juli hingga September, bulan Juli hingga Oktober, bulan Juli hingga November, dan bulan Juli hingga Desember. Kemudian bobot hasil pelatihan digunakan untuk melakukan prediksi terhadap data uji yang berbeda dengan data pelatihan. Hasil prediksi lalu dibandingkan dengan data asli untuk dihitung tingkat akurasinya dengan menggunakan MSE. Dengan data latih yang sama dilakukan pengulangan uji coba dengan merubah *learning rate*, momentum, dan jumlah duplikasi.

3.6.1 Pengaruh *Learning Rate* dan Momentum Terhadap MSE

Learning rate dan momentum berpengaruh pada kecepatan terjadinya konvergensi selama proses pelatihan. Nilai *learning rate* berkisar antara 0,1 hingga 0,9, sedangkan momentum berkisar antara 0,5 hingga 0,9. Pelatihan dilakukan dengan menggunakan nilai dari yang terkecil hingga terbesar. Hasil pengujian akan disajikan pada tabel 3.34.

Tabel 3.34 Pengaruh *Learning Rate* dan Momentum terhadap MSE

No.	<i>Learning Rate</i>	Momentum	MSE

3.6.2 Pengaruh Jumlah Data Latih Terhadap Tingkat Akurasi dan MSE

Setelah didapatkan struktur JST yang optimal, langkah selanjutnya adalah menghitung tingkat akurasi dan MSE dari sistem. Agar hasil pengujian, maka proses pelatihan dan pengujian harus dilakukan dalam kondisi yang sama. Percobaan akan dilakukan sebanyak 10 kali dengan data latih yang berbeda, data uji yang sama, dan parameter pelatihan yang sama. Hasil pengujian dimasukkan ke dalam tabel 3.35.

Tabel 3.35 Pengaruh Jumlah Data Latih Terhadap Tingkat Akurasi dan MSE

No.	Data Bulan	Tingkat Akurasi	MSE

3.6.3 Pengaruh Jumlah Data Uji Terhadap Tingkat Akurasi

Setelah didapatkan struktur JST yang optimal, langkah selanjutnya adalah menghitung tingkat akurasi dan MSE dari sistem. Agar hasil pengujian, maka proses pelatihan dan pengujian harus dilakukan dalam kondisi yang sama. Percobaan akan dilakukan sebanyak 6 kali dengan data latih yang sama, data uji yang berbeda, dan parameter pelatihan yang sama. Hasil pengujian dimasukkan ke dalam tabel 3.36.

Tabel 3.36 Pengaruh Jumlah Data Uji Terhadap Tingkat Akurasi dan MSE

No.	Data Bulan	Tingkat Akurasi	MSE