

**PREDIKSI CUACA MENGGUNAKAN JARINGAN SYARAF  
TIRUAN *BACKPROPAGATION***

**SKRIPSI**

oleh:

**AFIF NUR RAKHMAN**

**0610960002-96**



**PROGRAM STUDI ILMU KOMPUTER  
JURUSAN MATEMATIKA**

**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM**

**UNIVERSITAS BRAWIJAYA**

**MALANG**

**2012**

**PREDIKSI CUACA MENGGUNAKAN JARINGAN SYARAF  
TIRUAN *BACKPROPAGATION***

**SKRIPSI**

Sebagai salah satu syarat untuk memperoleh  
gelar Sarjana dalam bidang Ilmu Komputer

oleh:

**AFIF NUR RAKHMAN**

**0610960002-96**



**PROGRAM STUDI ILMU KOMPUTER**

**JURUSAN MATEMATIKA**

**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM**

**UNIVERSITAS BRAWIJAYA**

**MALANG**

**2012**

**LEMBAR PENGESAHAN SKRIPSI**  
**PREDIKSI CUACA MENGGUNAKAN JARINGAN SYARAF**  
**TIRUAN *BACKPROPAGATION***

Oleh:

**AFIF NUR RAKHMAN**

**0610960002-96**

Setelah dipertahankan di depan Majelis Penguji

pada tanggal 22 Mei 2012

dan dinyatakan memenuhi syarat untuk memperoleh gelar  
Sarjana Komputer dalam bidang Ilmu Komputer

**Pembimbing I,**

**Pembimbing II,**

**Candra Dewi, S.Kom.,MSc**  
**NIP. 197711142003122001**

**Dewi Yanti L.,S.Kom.,M.Kor**  
**NIP. 198111162005012004**

**Mengetahui,**

**Ketua Jurusan Matematika**

**Fakultas MIPA Universitas Brawijaya**

**Dr. Abdul Rouf Alghofari, M.Sc**

**NIP. 196709071992031001**

## LEMBAR PERNYATAAN

Saya yang bertanda tangan di bawah ini :

Nama : Afif Nur Rakhman

NIM : 0610960002

Jurusan : Matematika

Penulis Tugas Akhir berjudul : *Prediksi Cuaca Menggunakan Jaringan Syaraf Tiruan Backpropagation*

Dengan ini menyatakan bahwa :

1. Isi dari tugas Akhir yang saya buat adalah benar-benar karya sendiri dan tidak menjiplak karya orang lain, selain nama-nama yang tercatat di isi dan tertulis di daftar pustaka dalam Tugas Akhir ini.
2. Apabila dikemudian hari ternyata Tugas Akhir yang saya tulis terbukti hasil jiplakan, maka saya akan bersedia menanggung segala resiko yang akan saya terima.

Demikian pernyataan ini dibuat dengan segala kesadaran.

Malang, 22 Mei 2012

Yang menyatakan,

Afif Nur Rakhman

NIM. 0610960002

# PREDIKSI CUACA MENGGUNAKAN JARINGAN SYARAF TIRUAN *BACKPROPAGATION*

## ABSTRAK

Kondisi alam sudah banyak mengalami perubahan. Dibandingkan dengan masa silam, bila perubahan situasi berkenaan dengan cuaca dan iklim yang terjadi masih bisa diprediksi dan diketahui melalui tanda-tanda alam maupun melalui kalender yang berlangsung seperti biasa dalam setiap tahunnya, maka kondisi saat ini tidak lagi demikian. Hal ini telah menjadi fenomena global yang pada akhirnya tidak mudah di prediksi. Adanya kecerdasan buatan sangat membantu dalam proses prediksi cuaca dan salah satunya adalah Jaringan Syaraf Tiruan (JST). Metode JST yang digunakan dalam penelitian ini adalah metode *backpropagation*.

Metode *backpropagation* bekerja dengan memproses data parameter-parameter cuaca untuk dikenali polanya dalam proses pelatihan. Selama proses pelatihan, masing-masing bobot koneksi dipropagasikan ke seluruh unit dengan metode pelatihan perambatan balik (*backpropagation*). Data yang digunakan sebagai masukan adalah data parameter-parameter cuaca kota Malang, Jawa Timur selama 12 bulan atau 1 tahun. Untuk mengukur keberhasilan sistem prediksi cuaca ini adalah dengan menggunakan alat ukur berupa *Mean Square Error (MSE)* dan tingkat akurasi.

Penelitian ini menggunakan nilai *target error* 0,0001 dan jumlah *epoch* sebesar 15000 dapat menghasilkan struktur JST terbaik dengan kombinasi 5 *neuron input* dan 7 *neuron hidden*, dan dengan nilai *learning rate* 0,8 yang menghasilkan MSE sebesar 0,1050134. JST yang terbentuk mampu melakukan prediksi cuaca dengan tingkat akurasi sebesar 82,78689 %.

Kata Kunci : Jaringan Syaraf Tiruan, *Backpropagation*, prediksi cuaca.

# WEATHER PREDICTIONS USING BACKPROPAGATION NEURAL NETWORK

## ABSTRACT

Nowadays, natural conditions have undergone many changes. In the past, changes on weather and climate can be predicted based upon natural phenomena and events on calendar. It is quite difficult to predict weather changes using such conventional method. In addition, climate changes have become a global phenomenon that can not be easily predicted. Artificial intelligence is very helpful in the process of weather prediction, such as Artificial Neural Network (ANN) with the backpropagation method.

The method works by processing the data parameters of the weather for pattern recognition through the training process. During the training process, each of the connection weights are propagated to the entire unit. The data used as input data is the weather parameters in Malang, East Java, for over 1 year. The success rate of this system Mean Square Error (MSE) were used.

This study uses a target error value of 0.0001 and the number of epoch error of 15 000 to produce the best ANN structure with a combination of five input neurons and 7 hidden neurons, and the value of the yield learning rate 0.8 MSE of 0.1050134. ANN is capable of doing the weather forecast with accuracy rate of 82.78689%.

Keywords: Neural Network, Backpropagation, weather prediction.



## KATA PENGANTAR

*Alhamdulillah rabbil 'alamin.* Puji syukur penulis panjatkan kehadirat Allah SWT, karena atas segala rahmat dan limpahan hidayahnya, Skripsi yang berjudul “Prediksi cuaca Dengan Jaringan Syaraf Tiruan *Backpropagation* ” ini dapat diselesaikan. Skripsi ini disusun dan diajukan sebagai syarat untuk memperoleh gelar sarjana pada program studi Ilmu Komputer, jurusan Matematika, fakultas MIPA, universitas Brawijaya.

Semoga Allah SWT melimpahkan rahmat atas Nabi Muhammad SAW yang senantiasa memberikan cahaya petunjuk, dan atas keluarganya yang baik dan suci dengan rahmat yang berkah-Nya menyelamatkan kita pada hari akhirat.

Dalam penyelesaian tugas akhir ini, penulis telah mendapat begitu banyak bantuan baik moral maupun materiil dari banyak pihak. Atas bantuan yang telah diberikan, penulis ingin menyampaikan penghargaan dan ucapan terima kasih yang sedalam-dalamnya kepada:

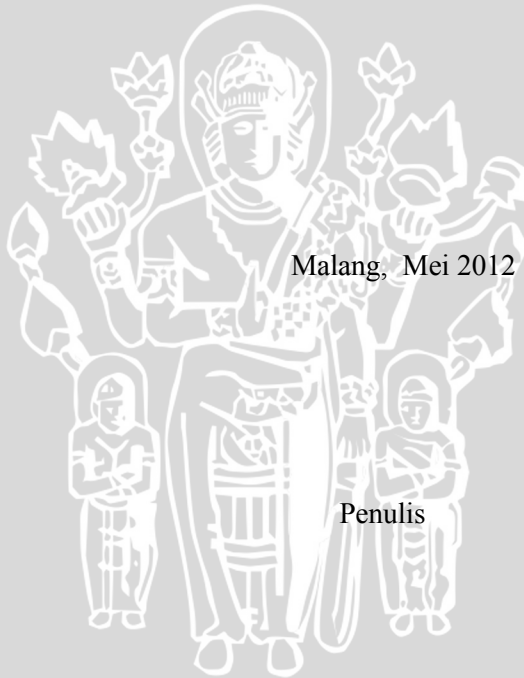
1. Candra Dewi, S.Kom., M.Sc., selaku Pembimbing Utama. Terima kasih atas semua saran, bantuan, kritikan, waktu, dorongan semangat dan bimbingannya.
2. Dewi Yanti L, S.Kom., M.Kom., selaku Pembimbing Kedua. Terima kasih atas semua saran, bantuan, kritikan, waktu, dorongan semangat dan bimbingannya.
3. Dr. Abdul Rouf Alghofari, M.Sc., selaku ketua Jurusan Matematika, Fakultas MIPA, Universitas Brawijaya.
4. Drs.Marji, MT., selaku ketua Program Studi Ilmu Komputer Universitas Brawijaya.
5. Drs. Muh. Arif Rahman, M.Kom., selaku Penasihat Akademik yang selama ini memberi saran, bimbingan dan nasehatnya.
6. Orang tua dan kedua kakak perempuan yang selalu memberi dukungan dan doa dalam kuliah dan penyelesaian tugas akhir ini. Terima kasih atas semuanya.
7. Tito Prasetyo terima kasih atas sharing dan dukungannya.
8. Teman–teman dan sahabat yang tidak bisa penulis sebutkan satu-persatu.

9. Semua teman-teman Ilmu Komputer angkatan 2006. Terima kasih atas semangat dan doanya.
10. Pihak lain yang tidak bisa penulis sebutkan satu-persatu.

Semoga penulisan laporan tugas akhir ini bermanfaat bagi pembaca sekalian. Dengan tidak lupa kodratnya sebagai manusia, penulis menyadari bahwa tugas akhir ini masih jauh dari kesempurnaan, dan mengandung banyak kekurangan, sehingga dengan segala kerendahan hati penulis mengharapkan kritik dan saran yang membangun dari pembaca.

Malang, Mei 2012

Penulis





## DAFTAR ISI

<b>HALAMAN JUDUL .....</b>	<b>i</b>
<b>HALAMAN PENGESAHAN .....</b>	<b>iii</b>
<b>HALAMAN PERNYATAAN .....</b>	<b>v</b>
<b>ABSTRAK .....</b>	<b>vii</b>
<b>ABSTRACT .....</b>	<b>ix</b>
<b>KATA PENGANTAR .....</b>	<b>xi</b>
<b>DAFTAR ISI .....</b>	<b>xiii</b>
<b>DAFTAR GAMBAR .....</b>	<b>xv</b>
<b>DAFTAR TABEL .....</b>	<b>xvii</b>
<b>DAFTAR LAMPIRAN .....</b>	<b>xix</b>

## **BAB I PENDAHULUAN**

1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	3
1.3 Batasan Masalah.....	3
1.4 Tujuan Penelitian.....	3
1.5 Manfaat Penelitian.....	3
1.6 Sistematika Penulisan.....	4

## **BAB II TINJAUAN PUSTAKA**

2.1 Pengertian cuaca.....	5
2.1.1 Cuaca.....	5
2.1.2 Unsur-unsur cuaca .....	5
2.1.3 kategori Cuaca.....	5
2.1.4 Meso Meteorologi.....	5

2.2. Peramalan Data Time Series .....	10
2.3 Jaringan Syaraf Tiruan.....	11
2.3.1 Defenisi Jaringan Syaraf Tiruan .....	11
2.3.2 Jaringan Syaraf Tiruan <i>Backpropagation</i> .....	14
2.3.2.1 Fungsi Aktivasi Pada <i>Backpropagation</i> .....	14
2.3.2.2 Pelatihan <i>Backpropagation</i> .....	16
2.3.2.3 Pengujian Jaringan Syaraf Tiruan.....	20
2.3.4 Optimalisasi Arsitektur <i>Backpropagation</i> ..	21
2.3.4.1 Inisialisasi Bobot Bias .....	21
2.3.3.2 Jumlah Unit Tersembunyi .....	22
2.3.3.3 Penentuan Jumlah Unit Pada Lapisan Tersembunyi.....	22
2.3.4.3 Normalisasi Data.....	23

### **BAB III METODOLGI DAN PERANCANGAN SISTEM**

3.1 Studi Literatur .....	26
3.2 Data Yang Digunakan.....	26
3.3 Menentukan Pola Data .....	26
3.4 Deskripsi Umum Sistem .....	28
3.5 Perancangan Proses .....	28
3.5.1. Proses Normalisasi Data.....	29
3.5.2. Proses Pelatihan Jaringan Syaraf Tiruan.....	30
3.5.3. Proses Inisialisasi Bobot dan Bias.....	31
3.5.4. Proses <i>Feedforward</i> .....	33
3.5.5. Proses Prediksi.....	33
3.6 Perhitungan Manual.....	34

3.7 Perancangan Uji Coba.....	44
3.7.1. Pengaruh Jumlah <i>Hidden Unit</i> dan <i>Learning Rate</i> .....	45
3.8 Perancangan Antarmuka.....	46

## **BAB IV HASIL DAN PEMBAHASAN**

4.1 Lingkungan Implementasi .....	49
4.1.1 Lingkungan Perangkat Keras.....	49
4.1.2 Lingkungan Perangkat Lunak .....	49
4.2 Implementasi Program .....	49
4.2.1 Struktur Data .....	50
4.2.2 Proses Ambil File Parameter Cuaca .....	50
4.2.3 Proses Normalisasi Data .....	52
4.2.4 Proses Pengkategorian Cuaca .....	54
4.2.5 Proses Training .....	52
4.2.5.1. Proses <i>Feedforward</i> .....	56
4.2.5.1.1. Prosedur Mencari $Z_{inj}$ .....	57
4.2.5.1.2. Prosedur Mencari $Y_{ink}$ .....	58
4.2.5.2. Proses <i>Backpropagation</i> .....	59
4.2.5.2.1. Prosedur Mencari Nilai $W_{jk}$ .....	59
4.2.5.2.2. Prosedur Mencari Nilai $d_{inj}$ .....	60
4.2.5.2.3. Prosedur Mencari $d_j$ .....	60
4.2.5.2.4. Prosedur Mencari $d_{Vij}$ .....	61
4.2.5.2.5. Prosedur <i>Update</i> Bobot.....	62
4.3 Implementasi Interface.....	63
4.3.1 Form Training.....	63
4.3.2 Form Prediction .....	64
4.3.3 Form Hasil Uji Coba .....	65

4.4 Uji Coba Sistem .....	66
4.4.1 Uji Pengaruh Jumlah <i>Neuron</i> dan <i>Learning Rate</i> Terhadap MSE .....	66
4.4.2 Uji Prediksi .....	74

**BAB V PENUTUP**

5.1 Kesimpulan.....	79
5.2 Saran.....	79

<b>DAFTAR PUSTAKA.....</b>	<b>81</b>
----------------------------	-----------

<b>LAMPIRAN.....</b>	<b>82</b>
----------------------	-----------



## DAFTAR GAMBAR

Gambar 2.1	Konsep Jaringan syaraf Manusia dan Jaringan syaraf tiruan .....	13
Gambar 2.2	Skema Neural Network.....	13
Gambar 3.1	Langkah-langkah Penelitian.....	25
Gambar 3.2	Arsitektur Jaringan.....	27
Gambar 3.3	Diagram Alir Sistem.....	29
Gambar 3.4	Diagram Alir proses Normalisasi.....	30
Gambar 3.5	Diagram Alir Pelatihan JST .....	31
Gambar 3.6	Diagram Alir inialisasi Bobot dan Bias .....	32
Gambar 3.7	Diagram Alir <i>feedforward</i> .....	33
Gambar 3.8	Diagram Alir Prediksi .....	34
Gambar 3.9	<i>Form Training</i> .....	46
Gambar 3.10	<i>Form Prediction</i> .....	47
Gambar 3.11	<i>Form Hasil Uji Coba</i> .....	48
<i>Sourcecode 4.1</i>	Struktur Data .....	50
<i>Sourcecode 4.2</i>	Proses Ambil file Parameter Cuaca .....	51
<i>Sourcecode 4.3</i>	Normalisasi Data .....	52
<i>Sourcecode 4.4</i>	Proses Pengkategorian Cuaca .....	54
<i>Sourcecode 4.5</i>	Prosedur Mencari Z_inj .....	57
<i>Sourcecode 4.6</i>	Prosedur Mencari Y_ink .....	58
<i>Sourcecode 4.7</i>	Prosedur Menghitung Wjk .....	59
<i>Sourcecode 4.8</i>	Prosedur Menghitung d_inj .....	60

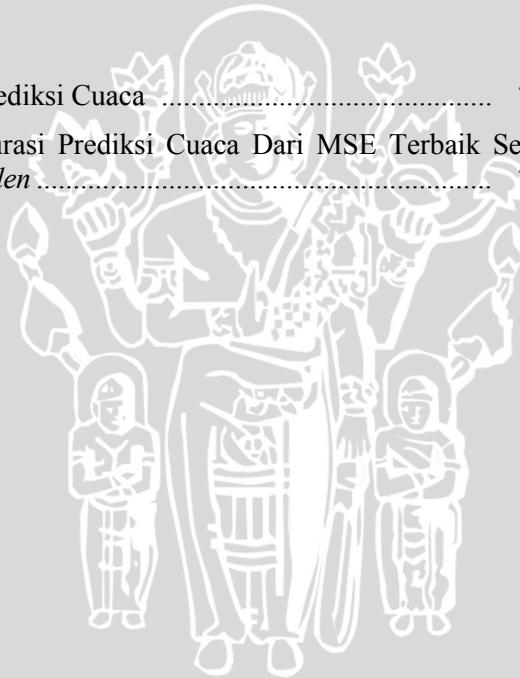


<i>Sourcecode</i> 4.9 Prosedur Mencari d <sub>j</sub> .....	61
<i>Sourcecode</i> 4.10 Prosedur Mencari d <sub>Vij</sub> .....	61
<i>Sourcecode</i> 4.11 Prosedur <i>Update</i> Bobot .....	62
Gambar 4.1 <i>Form Training</i> .....	63
Gambar 4.2 <i>Form Prediction</i> .....	64
Gambar 4.3 <i>Form Hasil Uji Coba</i> .....	65
Gambar 4.4 Grafik Pengaruh Jumlah 6 <i>Neuron Hidden</i> Terhadap MSE .....	67
Gambar 4.5 Grafik Pengaruh Jumlah 7 <i>Neuron Hidden</i> Terhadap MSE .....	69
Gambar 4.6 Grafik Pengaruh Jumlah 8 <i>Neuron Hidden</i> Terhadap MSE .....	70
Gambar 4.7 Grafik Pengaruh Jumlah 9 <i>Neuron Hidden</i> Terhadap MSE .....	72
Gambar 4.8 Grafik Pengaruh Jumlah 10 <i>Neuron Hidden</i> Terhadap MSE .....	73
Gambar 4.9 Grafik Pengaruh Jumlah <i>Neuron Hidden</i> Terhadap MSE .....	74

## DAFTAR TABEL

Tabel 3.1 Pola Data .....	26
Tabel 3.2 Definisi .....	27
Tabel 3.3 Pengaruh Jumlah Neuron Hidden Terhadap MSE .....	45
Tabel 3.4 Validasi Hasil Prediksi Cuaca .....	45
Tabel 3.6 Data Masukan .....	34
Tabel 3.7 Hasil Normalisasi Data Awal .....	35
Tabel 3.8 Data Nilai Awal Bobot $V_{ij}$ .....	35
Tabel 3.9 Data Nilai Awal Wjk .....	36
Tabel 3.10 Data $V_{ij}$ Dengan Nguyen Widrow .....	36
Tabel 3.11 Data Wjk Dengan Nguyen Widrow .....	37
Tabel 3.12 Nilai Bias $V_{oj}$ .....	37
Tabel 3.13 Nilai Bias Wok .....	37
Tabel 3.14 Operasi Pada Data $Z_{in}$ .....	38
Tabel 3.15 Aktivasi $Z_{in}$ .....	38
Tabel 3.16 Operasi Pada output ( $Y_{in}$ ) .....	39
Tabel 3.17 Hasil Aktivasi ( $Y_{in}$ ) .....	39
Tabel 3.18 Nilai Kemelesetan $y_k$ .....	40
Tabel 3.19 Tabel $\Delta W_{jk}$ .....	40
Tabel 3.20 Koreksi Nilai Bias Output (Wok) .....	40
Tabel 3.21 Tabel $\delta_{in_j}$ .....	41
Tabel 3.22 Tabel $\delta_{_j}$ .....	41
Tabel 3.23 Tabel $\Delta V_{ij}$ .....	42
Tabel 3.24 Tabel $\Delta V_{oj}$ .....	42
Tabel 3.25 Tabel $V_{ij}$ Baru .....	43
Tabel 3.23 Tabel $V_{oj}$ Baru .....	43

Tabel 3.23 Tabel Wjk Baru .....	44
Tabel 3.24 Tabel Wok Baru .....	44
Tabel 4.1 Pengaruh Jumlah 6 <i>Neuron Hidden</i> Terhadap MSE .	67
Tabel 4.2 Pengaruh Jumlah 7 <i>Neuron Hidden</i> Terhadap MSE .	68
Tabel 4.3 Pengaruh Jumlah 8 <i>Neuron Hidden</i> Terhadap MSE .	70
Tabel 4.4 Pengaruh Jumlah 9 <i>Neuron Hidden</i> Terhadap MSE .	71
Tabel 4.5 Pengaruh Jumlah 10 <i>Neuron Hidden</i> Terhadap MSE	72
Tabel 4.6 Hasil Prediksi Cuaca .....	75
Tabel 4.7 Akurasi Prediksi Cuaca .....	77
Tabel 4.8 Hasil Akurasi Prediksi Cuaca Dari MSE Terbaik Setiap Jumlah <i>Neuron Hidden</i> .....	78



# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Dewasa ini kondisi alam sudah banyak mengalami perubahan. Bila dimasa silam perubahan situasi berkenaan dengan cuaca dan iklim yang terjadi masih bisa diprediksi dan diketahui melalui tanda-tanda alam maupun melalui kalender yang berlangsung seperti biasa dalam setiap tahunnya, maka kondisi saat ini tidak lagi demikian. Hal ini telah menjadi fenomena global yang pada akhirnya tidak mudah diramal. Fenomena alam ini sulit untuk dikendalikan dan dimodifikasi, kecuali dalam skala kecil. Agar fenomena cuaca dapat dioptimalkan maka informasi tentang kondisi cuaca terutama peluang kejadian cuaca ekstrim (kemarau panjang dan kebanjiran) dan prediksi kondisi cuaca yang akan datang perlu diketahui sedini mungkin. Upaya ini bertujuan untuk menghindari atau meminimalisasi dampak yang ditimbulkan adanya iklim ekstrim tersebut (Sutikno, 2004).

Kemajuan teknologi informasi saat ini memberi peluang untuk bisa dilakukan prediksi terhadap kondisi cuaca dengan menggunakan data-data berbagai macam parameter cuaca yang ada. Melalui prediksi ini akan membantu masyarakat dalam mengambil keputusan diberbagai bidang seperti pertanian, perikanan, kehutanan, industri, kerajinan, pariwisata, perhubungan. Salah satu metode yang dapat diaplikasikan dengan baik dalam bidang prediksi adalah jaringan syaraf tiruan, yang merupakan cabang ilmu dari *Artificial Inteligent*. Secara umum prediksi yang dapat dilakukan oleh jaringan syaraf tiruan adalah prediksi runtut waktu (*time series*), sebagai *input* dan target dari *output* yang diinginkan pada proses pelatihan adalah data periode sebelum tahun yang akan diprediksi, data tersebut digunakan untuk menentukan bobot yang optimal. Setelah bobot optimal didapatkan dari proses pelatihan, bobot-bobot tersebut digunakan untuk menentukan nilai prediksi jika sistem diuji oleh data yang pernah masuk dalam sistem prediksi (Wahyudi, 2008).

*Backpropagation* adalah salah satu metode dari jaringan syaraf tiruan. Percobaan meramal cuaca menggunakan *Backpropagation* sudah pernah dilakukan sebelumnya, yaitu Prediksi cuaca berdasarkan fitur parameter cuaca dan periodenya menggunakan metode *Backpropagation* (Paras,dkk., 2007). Pada percobaan tersebut diketahui tingkat error rata-rata untuk beberapa fitur yang diujikan, yaitu sebesar kurang dari 3%. Selain itu, metode jaringan syaraf tiruan *Backpropagation* sudah sering dilakukan untuk meramal sesuatu dengan data *time series*. Seperti dalam percobaan prediksi suhu (*temperature forecasting*) (Mohsen Hayati dkk, 2007), percobaan prediksi harga saham (Wahyudi, 2008), yang dimana menghasilkan tingkat prediksi yang baik dan layak. Dan dengan parameter jaringan batas toleransi kesalahan (*target error*) 0.01, Laju Belajar 0.1 dan Momentum 0.2, maka teknik Jaringan Syaraf tiruan *Backpropagation* dapat cukup baik digunakan untuk memprediksi kebangkrutan bank di Indonesia dengan data satu tahun (Ruth C, 2002)

Hasil prediksi dari berbagai macam penelitian sebelumnya menunjukkan bahwa jaringan syaraf tiruan *Backpropagation* dapat memprediksi dengan baik. Maka dalam penelitian ini akan mencoba mengembangkan penelitian lebih lanjut mengenai pembuatan sistem yang dapat memprediksi cuaca dengan menggunakan jaringan syaraf tiruan *Backpropagation* dan data *time series*. Dengan menggunakan unsur-unsur cuaca yang ada, seperti radiasi matahari, temperatur, kelembaban udara, tekanan udara, kecepatan angin Diharap dengan metode *Backpropagation* akan mendapat hasil prediksi yang dapat mendekati hasil sesungguhnya.

## 1.2 Perumusan Masalah

Berdasarkan uraian pada latar belakang diatas, maka dapat di rumuskan masalah sebagai berikut :

1. Bagaimana mengimplementasikan metode jaringan syaraf tiruan *Backpropagation* untuk aplikasi memprediksi cuaca.
2. Berapa tingkat akurasi dari metode yang diterapkan.



### 1.3 Batasan Masalah

Dari perumusan masalah, diberikan batasan masalah untuk menghindari melebarnya masalah yang akan diselesaikan:

1. Penelitian pada skripsi ini menggunakan data cuaca harian selama satu tahun dari Fakultas Pertanian Brawijaya Kota Malang.
2. Menggunakan 5 parameter unsur cuaca yaitu, intensitas radiasi matahari (lamanya penyinaran), temperatur, kelembaban udara, tekanan udara, kecepatan angin

### 1.4 Tujuan Penelitian

Tujuan dari penelitian yang ingin dicapai adalah :

1. Membuat aplikasi untuk pemecahan masalah prediksi cuaca menggunakan jaringan syaraf tiruan *Backpropagation*.
2. Mengukur tingkat akurasi prediksi cuaca dari metode yang diterapkan.

### 1.5 Manfaat Penelitian

1. Dapat menerapkan metode jaringan syaraf tiruan *Backpropagation* untuk proses prediksi cuaca.
2. Sebagai tambahan informasi dan bahan masukan untuk memprediksi cuaca.

### 1.6 Sistematika Penulisan Tugas Akhir

#### BAB I PENDAHULUAN :

Bab ini berisi tentang latar belakang masalah, permasalahan, batasan masalah, tujuan penulisan, dan sistematika penulisan.

#### BAB II DASAR TEORI :

Bab ini berisi tentang teori-teori yang menjadi acuan untuk pelaksanaan penulisan tugas akhir yang meliputi teori tentang metode *Artificial Neural Network*, serta Algoritma *Backpropagation* dalam menyelesaikan masalah pada prediksi cuaca.

### BAB III PERANCANGAN DAN METODOLOGI

Bab ini berisi tentang aliran proses atau alur dari system yang akan dibuat serta algoritma dan flowchart dari program serta pembahasannya.

### BAB IV ANALISA DAN PEMBAHASAN

Bab ini membahas tentang metode-metode yang digunakan untuk merancang perangkat lunak prediksi cuaca menggunakan Jaringan Syaraf Tiruan *Backpropagation*.

### BAB V KESIMPULAN DAN SARAN

Bab ini berisi tentang kesimpulan dari system yang dirancang serta saran pengembangan dari keseluruhan tahapan pembuatan Tugas Akhir ini .



## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Pengertian Cuaca**

##### **2.1.1 Cuaca**

Cuaca merupakan keadaan atmosfer pada suatu tempat tertentu di bumi dengan waktu tertentu, biasanya diperhitungkan pada kondisi harian. Di Indonesia keadaan cuaca selalu diumumkan untuk jangka waktu sekitar 24 jam melalui prakiraan cuaca hasil analisis Badan Meteorologi dan Geofisika (BMG), Departemen Perhubungan. Untuk negara-negara yang sudah maju perubahan cuaca sudah diumumkan setiap jam dan sangat akurat (tepat). Cuaca terdiri dari seluruh fenomena yang terjadi di atmosfer Bumi atau sebuah planet lainnya. Cuaca biasanya merupakan sebuah aktivitas fenomena ini dalam waktu beberapa hari. Cuaca rata-rata dengan jangka waktu yang lebih lama dikenal sebagai iklim. Aspek cuaca ini diteliti lebih lanjut oleh ahli klimatologi, untuk tanda-tanda perubahan iklim. Iklim adalah keadaan cuaca rata-rata dalam waktu satu tahun yang penyelidikannya dilakukan dalam waktu yang lama (minimal 30 tahun) dan meliputi wilayah yang luas. Matahari adalah kendali iklim yang sangat penting dan sumber energi di bumi yang menimbulkan gerak udara dan arus laut. Kendali iklim yang lain, misalnya distribusi darat dan air, tekanan tinggi dan rendah, massa udara, pegunungan, arus laut dan badai. Perlu Anda ketahui bahwa ilmu yang mempelajari tentang iklim disebut Klimatologi, sedangkan ilmu yang mempelajari tentang keadaan cuaca disebut Meteorologi (Boucher.K, 1975).

##### **2.1.2 Unsur – unsur cuaca**

Unsur cuaca terdiri dari radiasi matahari, temperatur, kelembaban udara, tekanan udara, curah hujan, kecepatan angin (Boucher.K, 1975).

###### **a. Temperatur**

Temperatur adalah derajat panas dari aktivitas molekul dalam atmosfer. Lazimnya pengukuran temperatur dinyatakan dalam skala Celsius, Reamur, dan Fahrenheit. Temperatur di muka bumi tidaklah sama di berbagai tempat. Suhu udara diukur oleh derajat Celsius ( $^{\circ}\text{C}$ ), dengan alat termometer air raksa. Suhu bola kering

atau “suhu udara sebenarnya” adalah sebuah nilai yang diambil pada sebuah tempat yang terlindung. Diletakkan pada ketinggian 120 cm s/d 180 cm dari permukaan tanah. Bacaan dapat diambil pada waktu tertentu dai suatu hari. Suhu rata-rata bulanan adalah suhu yang diambil setiap bulanan. Rata-rata diambil dari suhu udara maksimum dan suhu udara minimum harian dan di rata-rata selama 30 hari. Untuk menunjukkan indikasi variasi diurnal, ini dapat dilengkapi dengan rata-rata maksimum dan minimum bulanan. (rata-rata maksimum bulanan adalah rata-rata dari suhu maksimum selama 30 hari). Selain itu juga baik untuk mengumpulkan suhu udara tertinggi dan terendah setiap bulannya (Hidayat.S, 2008).

## b. Kelembaban Udara

Kelembaban udara dapat digambarkan sebagai kelembaban mutlak (absolute humidity: AH), yaitu jumlah uap air yang sebenarnya ada dalam satuan massa atau unit volume udara, yaitu gram/ kilogram (g/kg) atau gram/ meter kubik (g/m<sup>3</sup>). Kelembaban relative (RH) adalah merupakan bentuk yang menunjukkan indikasi langsung dari potensi penguapan. Jumlah uap air udara yang dapat ditampung (saturation-point humidity SH) tergantung kepada suhunya. Kelembaban Relatif adalah rasio antara jumlah persentase uap air yang sebenarnya dengan jumlah uap air dalam udara yang dapat dikandung pada suhu tertentu. Dinyatakan dengan persen.

Kelembaban biasanya diukur dengan hygrometer bola basah dan kering. Alat ini biasanya terdiri dari dua thermometer yang digantungkan berdekatan. Thermometer yang satu mengukur suhu udara bola kering (*dry bulb*). Suhu bola kedua dibalut dengan kain basah. Hal ini dimaksudkan untuk memberikan kesan pendinginan dan ini yang menunjukkan suhu bola basah (WBT) yang selalu lebih rendah dari DBT. Untuk mendapatkan data kondisi kelembaban, adalah cukup untuk menentukan Kelembaban Relatif rata-rata bulanan maksimum dan minimum (maksimum selama 30 hari) dan Kelembaban Relatif minimum rata-rata selama 12 bulan. Kelembaban Relatif biasanya maksimum pada jam 6.00 pagi dan jam 15.00 minimum pada sore hari.

Kandungan uap air di atmosfer secara rata-rata cenderung konstan sekalipun ada perubahan musim dan variasi cuaca. Udara di atas Indonesia senantiasa lembap. Di dataran rendah, dekat pantai, rawa, hutan tropik, atau sungai-sungai besar, kelembaban udara



selalu tinggi, yaitu diatas 60 persen. Didaerah pedalaman atau di daerah-daerah yang tinggi di lereng gunung, kelembaban udara yang tinggi, lebih memungkinkan adanya turun hujan. Karena sifat kepulauannya, maka kelembaban udara diatas Indonesia selalu tinggi.

Penurunan suhu tadi menyebabkan kapasitas udara (jumlah uap air yang terkandung di udara) akan turun sampai kapasitas tetap sama dengan jumlah uap air yang sebenarnya terkandung di udara. Penurunan suhu merupakan panas alam yang penting dalam menjenuhkan udara. Bila suhu terus turun sampai dibawah suhu udara jenuh, maka terjadi kelebihan uap air dibandingkan dengan kapasitas udara pada suhu baru yang lebih rendah itu. Akibatnya, uap air akan berubah menjadi bentuk titik-titik air atau es. Suhu pada waktu kejenuhan itu terjadi menyebabkan perubahan uap air menjadi air disebut titik embun. Kecepatan angin diukur oleh anemometer, dan arahnya diukur oleh busur angin (*vane*) . alat anemograf dapat menghasilkan rekaman menerus dari kecepatan angin dan perubahan arah. Angin bebas normalnya diukur pada daerah yang datar pada ketinggian 10 m. Pengukuran pada daerah perkotaan biasanya diambil pada ketinggian antara 10 m sampai dengan 20 m untuk menghindari halangan. Kecepatan dekat permukaan tanah baik lebih rendah daripada angin bebas. Arah angin dapat dikelompokkan menjadi delapan atau enam belas kategori. Kecepatan angin diukur dengan m/det (m/s) (Hidayat.S, 2008).

### c. Radiasi Matahari

Meskipun hanya sebagian kecil dari radiasi yang dipancarkan matahari diterima permukaan bumi, namun radiasi matahari merupakan sumber energi utama untuk proses-proses fisika atmosfer. Proses-proses fisika atmosfer yang dapat menentukan proses terbentuknya cuaca di atmosfer bumi kita ini.

Radiasi matahari, yang merupakan gelombang elektromagnetik dibangkitkan dari proses fusi nuklir yang mengubah hidrogen menjadi helium. Permukaan matahari bersuhu 6000 K meskipun bagian dalamnya bersuhu jutaan derajat Kelvin. Dengan suhu permukaan tersebut, radiasi yang dipancarkan berupa gelombang elektromagnetik sebesar 73,5 juta watt setiap m<sup>2</sup> permukaan matahari. Dengan jarak rata-rata matahari-bumi sejauh 150 juta km (Trewartha & Horn, 1980), radiasi yang samapai di



puncak atmosfer rata-rata sebesar  $1360 \text{ Wm}^{-2}$ . Sedangkan rasdiasi matahari yang sampai dipermukaan bumi (daratan atau lautan) sekitar setengah dari yang diterima di puncak atmosfer, karena sebagian akan diserap dan dipantulkan kembali ke angkasa luar oleh atmosfer khususnya oleh awan. Rata-rata sebesar 30% radiasi matahari yang sampai di bumi dipantulkan kembali ke angkasa luar (Wallace & Hobbs, 1977).

d. Tekanan Udara

Tekanan udara ialah tekanan yang diberikan oleh udara karena bertanya kepada bidang horizontal yang luasnya 1 cm. Tekanan udara pada suatu tempat tidak tetap, demikian pula tekanan udara berbeda-beda dari tempat ke tempoat baik dalam jarak pendek maupun dalam areal yang luas. Tekanan udara ini merupakan gaya berat kolom udara dari permukaan tanah sampai puncak atmosfer per satuan luas gaya. Gaya ini ditimbulkan oleh percepatan ke bawah berupa gravitasi ( $g$ ) dan massa ( $m$ ). Hasil dari perkalian ini disebut berat ( $W$ ), oleh karena itu tekanan udara pada setiap titik ( $P$ ) merupakan berat total udara diatas titik tersebut per satuan luas ( $A$ ) (Wallace & Hobbs, 1977).

e. Angin

Angin adalah pergerakan udara pada arah horizontal atau hampir horizontal. Sedangkan pergerakan udara arah vertikal dinamakan aliran udara. Angin diberi nama berdasarkan arah dari mana angin itu bertiup. Arah angin datang disebut winward sedangkan tujuan angin disebut leeward. Angin selalu bertiup dari tempat yang bertekanan tionggi ke tempat yang bertekanan rendah mengikuti hukum buys-ballot yaitu di belahan bumi utara ke arah angin membelok ke kanan dan di belahan bumi selatan arah angin membelok ke kiri penyimpangan ini disebabkan oleh perputaran bumi pada porosnya (rotasi) (Holton, 1979).

### 2.1.3 Kategori Cuaca

Berdasarkan diskusi pada tanggal 7 januari 2011 dengan pakar Klimatologi dan meteorologi dari Fakultas Pertanian Brawijaya, Bapak Dr.Rudy Sulistyono, terdapat 3 kategori cuaca yang dapat di kelompokkan, yaitu cuaca cerah, cuaca mendung atau berawan dan hujan. Berikut terdapat penjelasan dari kategori cuaca tersebut :

#### a. Cuaca Cerah

Cuaca dapat dikatakan cerah apabila dalam satu hari terdapat lama penyinaran matahari lebih dari 70% dari lama penyinaran selama 6 jam, memiliki tekanan udara sebesar  $< 997\text{mbr}$ , mempunyai nilai kelembaban udara relatif (RH) sebesar 60-65%, mempunyai kecepatan angin  $\geq 4$ , dan memiliki temperatur  $< 24\text{ C}^\circ$ .

#### b. Cuaca mendung

Cuaca dapat dikatakan mendung apabila dalam satu hari terdapat lama penyinaran antara 60-70% dari lama penyinaran selama 6 jam, memiliki tekanan udara antara 997mbr, mempunyai nilai kelembaban udara relatif (RH) sebesar 66-79%, mempunyai kecepatan angin antara 3-4 km/jam, dan mempunyai temperatur sebesar  $24\text{ C}^\circ$ .

#### c. Hujan

Dikatakan hujan apabila lama penyinaran dalam sehari kurang dari 60% dari lama penyinaran selama 6 jam, memiliki tekanan udara  $\geq 998\text{mbr}$ , mempunyai nilai kelembaban relatif (RH) lebih besar dari 79%, mempunyai kecepatan angin 1-3 km/jam, dan mempunyai temperatur  $> 24\text{ C}^\circ$ .

### 2.1.4 Meso Meteorologi

Meso meteorologi adalah meteorologi yang membahas perilaku atmosfer dalam daerah yang relatif sempit, tetapi pola iklimnya sudah berbeda dari cuaca wilayah sekitarnya. Sebagai contoh adalah cuaca perkotaan. Skala cuaca meso sekitar 0 – 100 km. (Boucher.K, 1975)

## 2.2 Prediksi Data Time Series

Teknik prediksi terbagi menjadi dua kelompok yaitu analisis kualitatif dan analisis kuantitatif. Teknik kualitatif merupakan prediksi berdasarkan pendapat suatu pihak, dan datanya tidak bias direpresentasikan secara tegas menjadi suatu nilai. Teknik prediksi tersebut misalnya adalah prediksi pendapat (*judgement forecast*). Sebaliknya, teknik prediksi kuantitatif merupakan teknik prediksi yang berdasarkan pada masa lalu (*data histories*) dan dapat dibuat dalam bentuk angka yang biasa disebut sebagai data *time series* (Jumingan, 2009).

Prediksi data *time series* memprediksi apa yang terjadi berdasarkan data histories masa lalu. *Time series* adalah kumpulan dari pengamatan yang teratur pada sebuah variabel selama periode waktu yang sama dan suksesif. Dengan mempelajari bagaimana sebuah variabel berubah setiap waktu, sebuah relasi diantara kebutuhan dan waktu dapat diformulasikan dan digunakan untuk memprediksi tingkat kebutuhan yang akan datang (Jumingan, 2009)

Ada tiga tujuan dari metode *time series* yaitu sebagai berikut :

1. Prediksi (*forecasting*) berusaha memperkirakan perkembangan sistem untuk jangka pendek.
2. Pemodelan (*modeling*) berusaha menemukan gambaran tentang keakuratan sifat sistem untuk jangka panjang.
3. Penggolongan digunakan untuk menentukan sifat-sifat utama dari sistem.

Tiga tujuan tersebut berhubungan tetapi mungkin tidak sama. Pemodelan dapat digunakan untuk prediksi (*forecasting*) jangka pendek tetapi mungkin kemampuan untuk melakukan hal tersebut kurang maksimal dan sebaliknya. Langkah penting dalam memilih suatu metode runtun waktu (*time series*) yang tepat adalah dengan mempertimbangkan jenis pola data sehingga metode yang paling tepat dengan pola tersebut dapat diuji. Pola data dapat dibedakan menjadi empat yaitu sebagai berikut.

1. Pola horisontal terjadi bilamana nilai data berfluktuasi di sekitar nilai rata-rata yang tetap.
2. Pola musiman terjadi bilamana suatu runtun dipengaruhi oleh faktor musiman.
3. Pola siklis terjadi bilamana datanya dipengaruhi oleh faktor ekonomi jangka panjang seperti berhubungan dengan siklus bisnis.
4. Pola *trend* terjadi bilamana terdapat kenaikan atau penurunan sekuler jangka panjang dalam data.

Banyak runtun data yang mencakup kombinasi dari pola-pola diatas. Metode prediksi yang dapat membedakan setiap pola harus dipakai bila diinginkan adanya pemisahan komponen data tersebut. Demikian pula, metode prediksi alternatif dapat digunakan untuk mengenali pola dan mencocokkan data secara tepat sehingga nilai mendatang dapat diramalkan (Bose, 1996).

## **2.3 Jaringan Syaraf Tiruan**

### **2.3.1 Definisi Jaringan Syaraf Tiruan**

Jaringan syaraf tiruan merupakan sistem pemroses informasi yang memiliki karakteristik mirip dengan jaringan syaraf biologi. Jaringan syaraf tiruan merupakan sistem komputasi yang didasarkan atas pemodelan sistem syaraf biologis (*neurons*) melalui pendekatan dari sifat-sifat komputasi biologis (*biological computation*). Menurut Subiyanto (Subiyanto, 2002), jaringan syaraf tiruan adalah membuat model sistem komputasi yang dapat menirukan cara kerja jaringan syaraf biologi. jaringan syaraf tiruan dibentuk sebagai generalisasi model matematika dari jaringan syaraf biologi dengan asumsi sebagai berikut.

- a. Pemrosesan informasi terjadi pada banyak elemen sederhana (*neurons*).
- b. Sinyal dikirimkan diantara *neuron-neuron* melalui penghubung - penghubung.
- c. Penghubung antar *neuron* memiliki bobot yang akan memperkuat atau memperlemah sinyal.
- d. Untuk menentukan keluaran (*output*), setiap *neuron* menggunakan fungsi aktivasi yang dikenakan pada penjumlahan



masuk (input) yang diterima. Besarnya keluaran (output) ini selanjutnya dibandingkan dengan suatu batas ambang.

*Neuron* adalah unit pemroses informasi yang menjadi dasar dalam pengoperasian jaringan syaraf tiruan (Siang, 2005). *Neuron* terdiri dari 3 elemen pembentuk sebagai berikut :

- a. Himpunan unit-unit yang dihubungkan dengan jalur koneksi.
- b. Suatu unit penjumlah yang akan menjumlahkan masukan-masukan sinyal yang sudah dikalikan dengan bobotnya.
- c. Fungsi aktivasi yang akan menentukan apakah sinyal dari *input neuron* akan diteruskan ke *neuron* lain ataukah tidak.

Jaringan syaraf tiruan ditentukan oleh 3 hal sebagai berikut :

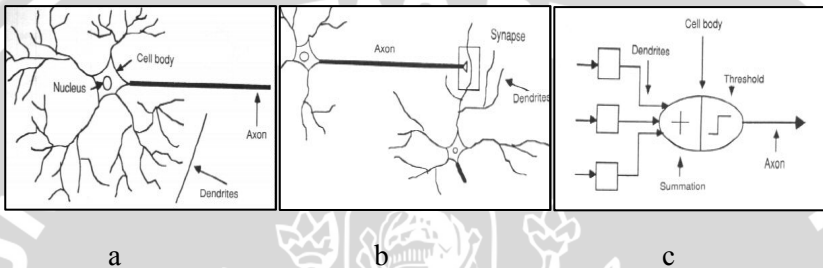
- a. Pola hubungan antar *neuron* (disebut arsitektur jaringan)
- b. Metode untuk menentukan bobot penghubung (disebut algoritma *training/learning*/pelatihan/belajar)
- c. Fungsi aktivasi.

Di dalam jaringan syaraf tiruan, istilah simpul (*node*) sering digunakan untuk menggantikan *neuron*.

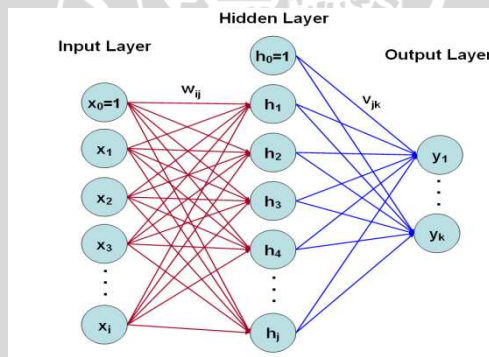
Setiap simpul pada jaringan menerima atau mengirim sinyal dari atau ke simpul-simpul lainnya. Pengiriman sinyal disampaikan melalui penghubung. Kekuatan hubungan yang terjadi antara setiap simpul yang saling terhubung dikenal dengan nama bobot. Model-model jaringan syaraf tiruan ditentukan oleh arsitektur jaringan serta algoritma pelatihan. Arsitektur biasanya menjelaskan arah perjalanan sinyal atau data di dalam jaringan. Sedangkan algoritma belajar menjelaskan bagaimana bobot koneksi harus diubah agar pasangan masukan-keluaran yang diinginkan dapat tercapai. Perubahan harga bobot koneksi dapat dilakukan dengan berbagai cara, tergantung pada jenis algoritma pelatihan yang digunakan. Dengan mengatur besarnya nilai bobot ini diharapkan bahwa kinerja jaringan dalam mempelajari berbagai macam pola yang dinyatakan oleh setiap pasangan masukan-keluaran akan meningkat.



Inspirasi awal terhadap teknik ini adalah berasal dari pengujian jaringan bioelektrik di dalam otak manusia yang dibentuk oleh *neurons* dan *synapses*. Dalam suatu model jaringan syaraf, *node* sederhana (disebut juga 'neuron', 'neurode', 'PE' atau unsur-unsur pengolahan, atau 'satuan') yang terhubung secara bersama membentuk suatu jaringan *node* yang disebut jaringan syaraf (*neural network*) (Gambar 2.1).



Gambar 2.1. Konsep jaringan syaraf manusia dan model jaringan syaraf tiruan. (a) komponen-komponen syaraf (neuron), (b) gambaran mengenai synapses, dan (c) model jaringan syaraf.



Gambar 2.2 Skema neural network.

Biasanya terdapat paling tidak tiga lapisan pada suatu jaringan langkah maju, terdiri dari lapisan input (*input layer*), suatu lapisan tersembunyi (*hidden layer*), dan suatu lapisan output (*output layer*). Lapisan input memberi umpan kepada lapisan tersembunyi, kemudian lapisan tersembunyi memberi umpan kepada lapisan

output. Pengolahan aktual dalam suatu jaringan terjadi didalam *node* pada lapisan tersembunyi dan lapisan output (Gambar 2).

Setiap koneksi antara syaraf memiliki bobot numerik. Apabila jaringan ini bekerja, suatu nilai akan diberikan pada setiap *node* – nilai tersebut akan diberikan oleh operator manusia, dari sensor lingkungan, ataupun dari beberapa program eksternal. Setiap *node* kemudian memberikan nilai tertentu pada suatu koneksi yang membawanya keluar, kemudian setiap koneksi mengalikannya dengan suatu pembobot. Setiap *node* pada lapisan berikutnya kemudian menerima nilai yang merupakan penjumlahan dari nilai yang dihasilkan dari setiap koneksi, dan dalam setiap *node* dilakukan perhitungan sederhana terhadap nilai tersebut. Secara khas fungsi ini merupakan fungsi sigmoid. Proses ini kemudian berulang, dengan hasil yang dilewatkan pada lapisan sub-sekuen dari *node-node* hingga mencapai *node* pada lapisan output.

### 2.3.2 Jaringan Syaraf Tiruan *Backpropagation*

*Backpropagation* merupakan model jaringan syaraf tiruan dengan *layer* jamak. Seperti halnya model jaringan syaraf tiruan lainnya, *backpropagation* melatih jaringan untuk mendapatkan keseimbangan antara kemampuan jaringan untuk mengenali pola yang digunakan selama pelatihan serta kemampuan jaringan untuk memberikan respon yang benar terhadap pola masukan yang serupa (tapi tidak sama) dengan pola yang dipakai selama pelatihan.

#### 2.3.2.1 Fungsi Aktivasi pada *Backpropagation*

Dalam JST, fungsi aktivasi digunakan untuk menentukan keluaran suatu *neuron*. Argumen fungsi aktivasi adalah net masukan (kombinasi linier masukan dan bobotnya yang ditunjukkan dengan persamaan 2.1) (Siang, 2004).

$$\text{jika net} = \sum x_i w_i \quad (2.1)$$

Maka fungsi aktivasinya ditunjukkan oleh persamaan :

$$f(\text{net}) = f\left(\sum x_i w_i\right) \quad (2.2)$$

Beberapa fungsi aktivasi yang dapat dipakai yaitu (Siang, 2005) :

a. Fungsi Sigmoid biner

adalah fungsi yang memiliki *range* (0,1). Fungsi sigmoid biner didefinisikan sebagai berikut.

$$f(x) = \frac{1}{1 + e^{-x}}, \quad -\infty \leq x \leq \infty \quad (2.3)$$

$$f'(x) = f(x)(1 - f(x)) \quad (2.4)$$

Fungsi lain yang sering dipakai adalah fungsi sigmoid bipolar dengan *range* (-1,1) yang didefinisikan sebagai berikut :

$$f(x) = \frac{2}{1 + e^{-x}} - 1, \quad -\infty \leq x \leq \infty \quad (2.5)$$

$$f(x) = \frac{(1 + f(x))(1 - f(x))}{2} \quad (2.6)$$

Fungsi sigmoid memiliki nilai maksimum 1. Untuk pola yang targetnya lebih dari 1, pola masukan dan keluaran harus terlebih dahulu ditransformasi sehingga semua polanya memiliki *range* yang sama seperti fungsi sigmoid yang dipakai. Alternatif lain adalah menggunakan fungsi aktivasi sigmoid hanya pada *layer* yang bukan *layer* keluaran. Pada *layer* keluaran, fungsi aktivasi yang dipakai adalah fungsi identitas  $f(x) = x$ .

b. Fungsi *threshold* (batas ambang)

$$y = \begin{cases} 0, & \text{jika } x < 0 \\ 1, & \text{jika } x \geq 0 \end{cases} \quad (2.7)$$

Fungsi *threshold* pada persamaan 2.7 merupakan fungsi *threshold* biner. Adakalanya dalam JST ditambahkan suatu unit masukan yang nilainya selalu 1. Unit tersebut dikenal dengan bias. Bias dapat dipandang sebagai sebuah *input* yang nilainya selalu 1. Bias berfungsi untuk mengubah *threshold* menjadi = 0.

c. Fungsi Sigmoid Bipolar (tansig)

Fungsi sigmoid bipolar hampir sama dengan fungsi sigmoid biner, hanya saja *output* dari fungsi ini memiliki *range* antara 1 sampai -1. Fungsi sigmoid bipolar ditunjukkan pada persamaan 2.7.

$$y = f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (2.8)$$

dengan persamaan 2.8 sebagai turunannya

$$f'(x) = [1 + f(x)][1 - f(x)] \quad (2.9)$$

d. Fungsi identitas

$$f(x) = x \quad (2.10)$$

Persamaan 2.10 digunakan jika keluaran yang dihasilkan oleh JST merupakan sembarang bilangan *riil* (bukan hanya pada *range* [0,1] atau [1,-1]).

### 2.3.2.2 Pelatihan *Backpropagation*

Pelatihan *backpropagation* meliputi 3 fase sebagai berikut :

1. Fase 1, yaitu propagasi maju.  
Pola masukan dihitung maju mulai dari *layer* masukan hingga *layer* keluaran menggunakan fungsi aktivasi yang ditentukan.
2. Fase 2, yaitu propagasi mundur.  
Selisih antara keluaran jaringan dengan target yang diinginkan merupakan kesalahan yang terjadi. Kesalahan yang terjadi itu dipropagasi mundur. Dimulai dari garis yang berhubungan langsung dengan unit-unit di *layer* keluaran.
3. Fase 3, yaitu perubahan bobot.  
Modifikasi bobot untuk menurunkan kesalahan yang terjadi. Ketiga fase tersebut diulang-ulang terus hingga kondisi penghentian dipenuhi. Algoritma pelatihan untuk jaringan *Backpropagation* dengan satu *layer* tersembunyi (dengan fungsi aktivasi sigmoid biner) adalah sebagai berikut :

- a. Langkah 0  
Inisialisasi semua bobot dengan bilangan acak kecil.
- b. Langkah 1  
Jika kondisi penghentian belum dipenuhi, lakukan langkah 2-8. Kondisi penghentian yaitu target *error* dan *epoch* (siklus tiap perubahan bobot).
- c. Langkah 2  
Untuk setiap pasang data pelatihan, lakukan langkah 3-8.
- d. Langkah 3 (langkah 3-5 merupakan fase 1, fase *Feedforward*)  
Tiap unit masukan menerima sinyal dan meneruskannya ke unit tersembunyi di atasnya.
- e. Langkah 4  
Hitung semua keluaran di unit tersembunyi  $z_j$  ( $j = 1, 2, \dots, p$ ).

$$z_{in\ j} = v_{oj} + \sum_{i=1}^n x_i v_{ij} \quad (2.11)$$

$$z_j = f(z_{in\ j}) = \frac{1}{1 + e^{-z_{in\ j}}} \quad (2.12)$$

- f. Langkah 5  
Hitung semua keluaran jaringan di unit keluaran  $y_k$  ( $k = 1, 2, \dots, m$ ).

$$y_{in\ k} = w_{ok} + \sum_{j=1}^p z_j w_{jk} \quad (2.13)$$

$$y_k = f(y_{in\ k}) = \frac{1}{1 + e^{-y_{in\ k}}} \quad (2.14)$$



- g. Langkah 6 (langkah 6-7 merupakan fase 2, fase *Backpropagation*)

Hitung faktor  $\delta$  unit keluaran berdasarkan kesalahan di setiap unit keluaran  $y_k$  ( $k = 1, 2, \dots, m$ ).

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

$t_k$  = target keluaran (2.15)

$\delta$  merupakan unit kesalahan yang akan dipakai dalam perubahan bobot *layer* dibawahnya.

Hitung perubahan bobot  $w_{kj}$  dengan laju pemahaman  $\alpha$ .

$$\Delta w_{kj} = \alpha \delta_k z_j, \quad k = 1, 2, \dots, m; \quad j = 0, 1, \dots, p \quad (2.16)$$

- h. Langkah 7

Hitung faktor  $\delta$  unit tersembunyi berdasarkan kesalahan di setiap unit tersembunyi  $z_j$  ( $j = 1, 2, \dots, p$ ):

$$\delta_{in_j} = \sum_{k=1}^n \delta_k w_{jk} \quad (2.17)$$

Faktor  $\delta$  tersembunyi :

$$\delta_j = \delta_{in_j} f'(z_{net\ j}) = \delta_{net\ j} z_j (1 - z_j) \quad (2.18)$$

Hitung suku perubahan bobot  $v_{ji}$  :

$$\Delta v_{ij} = \alpha \delta_i x_i, \quad j = 1, 2, \dots, p; \quad i = 1, 2, \dots, n \quad (2.19)$$

- i. Langkah 8 (fase 3, fase perubahan bobot)

Hitung semua perubahan bobot. Perubahan bobot garis yang menuju ke unit keluaran, yaitu:

$$W_{jk}(\text{baru}) = W_{jk}(\text{lama}) + \Delta w_k \quad (2.20)$$

; ( $k = 1, 2, \dots, m; j = 0, 1, \dots, p$ ).

Kemudian perbaiki bias di unit output dengan persamaan, yaitu :

$$W_{ok}(\text{baru}) = W_{ok}(\text{lama}) + \Delta w_{ok} \quad (2.21)$$

Perubahan bobot garis yang menuju ke unit tersembunyi, yaitu:

$$v_{ij}(\text{baru}) = v_{ij}(\text{lama}) + \Delta v_{ij} \\ ;(j = 1, 2, \dots, p ; i = 0, 1, \dots, n). \quad (2.22)$$

Kemudian perbaiki bias di unit output dengan persamaan, yaitu :

$$V_{oj}(\text{baru}) = V_{oj}(\text{lama}) + \Delta V_{oj} \quad (2.23)$$

Parameter  $\alpha$  merupakan laju pemahaman yang menentukan kecepatan iterasi. Nilai  $\alpha$  terletak antara 0 dan 1 ( $0 \leq \alpha \leq 1$ ). Semakin besar harga  $\alpha$ , semakin sedikit iterasi yang dipakai. Akan tetapi jika harga  $\alpha$  terlalu besar, maka akan merusak pola yang sudah benar sehingga pemahaman menjadi lambat. Satu siklus pelatihan yang melibatkan semua pola disebut *epoch*.

Pemilihan bobot awal sangat mempengaruhi jaringan syaraf tiruan dalam mencapai minimum global (atau mungkin lokal saja) terhadap nilai *error* (kesalahan) dan cepat tidaknya proses pelatihan menuju kekonvergenan.

Apabila bobot awal terlalu besar maka *input* (masukan) ke setiap lapisan tersembunyi atau lapisan *output* (keluaran) akan jatuh pada daerah dimana turunan fungsi sigmoidnya akan sangat kecil. Apabila bobot awal terlalu kecil, maka *input* (masukan) ke setiap lapisan tersembunyi atau lapisan *output* (keluaran) akan sangat kecil. Hal ini akan menyebabkan proses pelatihan berjalan sangat lambat. Biasanya bobot awal diinisialisasi secara random dengan nilai antara -0.5 sampai 0.5 (atau -1 sampai 1 atau interval yang lainnya).

Setelah pelatihan selesai dilakukan, jaringan dapat dipakai untuk pengenalan pola. Dalam hal ini, hanya propagasi maju (langkah 4 dan 5) saja yang dipakai untuk menentukan keluaran jaringan (Siang, 2005).

Seperti halnya jaringan syaraf yang lain, pada jaringan *feedforward* (umpan maju) pelatihan dilakukan dalam rangka perhitungan bobot sehingga pada akhir pelatihan akan diperoleh bobot-bobot yang baik. Selama proses pelatihan, bobot-bobot diatur secara iteratif untuk meminimumkan *error* (kesalahan) yang terjadi. *Error* (kesalahan) dihitung berdasarkan rata-rata kuadrat kesalahan (MSE). Rata-rata kuadrat kesalahan juga dijadikan dasar perhitungan

unjuk kerja fungsi aktivasi. Sebagian besar pelatihan untuk jaringan *feedforward* (umpan maju) menggunakan gradien dari fungsi aktivasi untuk menentukan bagaimana mengatur bobot-bobot dalam rangka meminimumkan kinerja. Gradien ini ditentukan dengan menggunakan suatu teknik yang disebut *backpropagation*. Pada dasarnya, algoritma pelatihan standar *backpropagation* akan menggerakkan bobot dengan arah gradien negatif. Prinsip dasar dari algoritma *backpropagation* adalah memperbaiki bobot-bobot jaringan dengan arah yang membuat fungsi aktivasi menjadi turun dengan cepat.

Tujuan dari perubahan bobot untuk setiap lapisan, bukan merupakan hal yang sangat penting. Perhitungan kesalahan merupakan pengukuran bagaimana jaringan dapat belajar dengan baik. Kesalahan pada keluaran dari jaringan merupakan selisih antara keluaran aktual (*current output*) dan keluaran target (*desired output*). Langkah berikutnya adalah menghitung nilai SSE (*Sum Square Error*) yang merupakan hasil penjumlahan nilai kuadrat *error neuron1* dan *neuron2* pada lapisan *output* tiap data, dimana hasil penjumlahan keseluruhan nilai SSE akan digunakan untuk menghitung nilai RMSE (*Root Mean Square Error*) tiap iterasi (Kusumadewi, 2003).

### 2.3.2.3 Pengujian Jaringan Syaraf Tiruan

Perhitungan galat merupakan pengukuran bagaimana jaringan saraf tiruan dapat belajar dengan baik. Perhitungan galat ini merupakan pengukuran ketepatan jaringan saraf tiruan terhadap data target pembelajaran. Galat pada keluaran jaringan saraf tiruan merupakan selisih antara keluaran sebenarnya (*current output*) dengan keluaran yang diinginkan (*desired output*) dari masukan data tertentu. Pada proses pembelajaran, data yang menjadi pembanding adalah data pembelajaran, sedangkan pada proses pengujian, data yang dipakai adalah data uji. Jumlah galat pada jaringan saraf tiruan dapat dihitung dengan menggunakan jumlah galat kuadrat (SSE—*sum of squared error*), atau rata-rata galat kuadrat (MSE—*mean square error*) Berikut ini adalah cara menghitung SSE, MSE untuk kumpulan data  $d$ , pada jaringan saraf tiruan dengan kumpulan neuron keluaran *outputs* (Kusumadewi, 2003):

$$SSE = \sum_{i=1}^n (t_i - o_i)^2 \quad (2.24)$$

$$MSE = \frac{\sum_{i=1}^n (t_i - o_i)^2}{n} \quad (2.25)$$

Dengan:

$t_i$  adalah nilai target pada neuron keluaran ke- $k$

$o_i$  adalah nilai output pada neuron keluaran ke- $k$

$n$  adalah jumlah neuron keluaran

### 2.3.3 Optimalisasi Arsitektur Backpropagation

#### 2.3.3.1 Inisialisasi bobot bias

Bobot awal akan mempengaruhi apakah jaringan mencapai titik maksimum lokal, atau maksimum global, dan seberapa cepat konvergensinya. Bobot yg menghasilkan nilai turunan fungsi aktivasi yang terlalu kecil sebaiknya dihindari, demikian pula nilai bobot awal tidak boleh terlalu besar karena nilai turunan fungsi aktivasinya menjadi sangat kecil

Nguyen Widrow (1990) dalam siang J.J (2005) mengusulkan cara membuat inisialisasi bobot dan bias ke unit tersembunyi sehingga menghasilkan iterasi lebih cepat.

$$\beta = 0.7 \sqrt{n} \quad (2.26)$$

$n$  = jumlah unit masukan (*input unit*)

$p$  = jumlah unit tersembunyi (*hidden unit*)

$\beta$  (beta) = faktor skala

Untuk setiap unit tersembunyi dari unit ke-1 sampai unit ke- $p$  :

1. Inisialisasi bobot-bobot antara unit *input* ke unit tersembunyi ( $j=1,2,\dots,p$ ) dengan cara:

- a. Menentukan bobot-bobot antara unit *input* ke unit tersembunyi ( $v_{ij}$ ) dengan bilangan acak antara  $-\beta$  dan  $\beta$ .
- b. Menghitung  $\|v_j\|$  dengan menggunakan persamaan 2.28.

$$\|v_j\| = \sqrt{v_{j1}^2 + v_{j2}^2 + \dots + v_{jn}^2} \quad (2.27)$$

- c. Menginisialisasikan kembali  $v_{ij}$  seperti yang ditunjukkan pada persamaan 2.29:

$$v_{ij} = \frac{\beta \cdot v_{ij}(\text{lama})}{\|v_j\|} \quad (2.28)$$

2. Menentukan bias antara *input* ke unit tersembunyi ( $j=1,2,\dots,p$ ) dengan  $v_{oj}$  di set dengan bilangan acak yang terletak pada skala antara  $-\beta$  dan  $\beta$ .

### 2.3.3.2 Jumlah Unit Tersembunyi

Menentukan jumlah lapisan tersembunyi (*hidden layer*) dan jumlah unitnya sangat sulit. Hasil teoritis yang didapat menunjukkan bahwa jaringan dengan 1 *hidden layer* saja sudah cukup bagi *Backpropagation* untuk mengenali hubungan antar masukan dan target dengan tingkat ketelitian yang ditentukan. Akan tetapi penambahan jumlah *hidden layer* kadang kala mampu membuat proses *learning* menjadi lebih mudah (Siang, 2005). Sedang jumlah unit tiap lapisan yang optimal dapat ditentukan dengan beberapa percobaan.

### 2.3.3.3 Penentuan jumlah unit pada lapisan tersembunyi

Menggunakan satu layer sudah cukup untuk menyelesaikan berbagai kasus. Sedangkan untuk menentukan jumlah neuron dalam satu hidden layer harus memenuhi aturan seperti persamaan:

$$N_{hidden} \approx \frac{2}{3} * N_{input} + N_{output} \quad (2.29)$$

Aturan bahwa jumlah neuron dalam satu hidden layer disarankan 2/3 dari jumlah input ditambah dengan jumlah output, dan jumlah neuron kurang dari dua kali jumlah neuron input (Heaton.J, 2010).



### 2.3.4 Normalisasi Data

Sebelum data diproses sebagai masukan dari suatu JST maka sebelumnya data juga perlu dilakukan normalisasi dengan menyesuaikan *range output* fungsi aktivasi. Misalkan menggunakan fungsi aktivasi sigmoid maka data masukan harus dirubah dengan *range*  $[0...1]$ , namun karena merupakan fungsi kontinu maka nilai 0 dan 1 tidak pernah dicapai. Maka dari itu *range* dirubah menjadi  $[0.1...0.9]$ , sehingga dapat dirumuskan pada persamaan 2.19 berikut (Siang, 2005).

Dimana

$x$  : *input*

$a$  : nilai minimum dari data

$b$  : nilai maksimum dari data

$x'$  : *input* yang telah ternormalisasi

0.8 merupakan jarak dari skala 0,9-0,1



UNIVERSITAS BRAWIJAYA

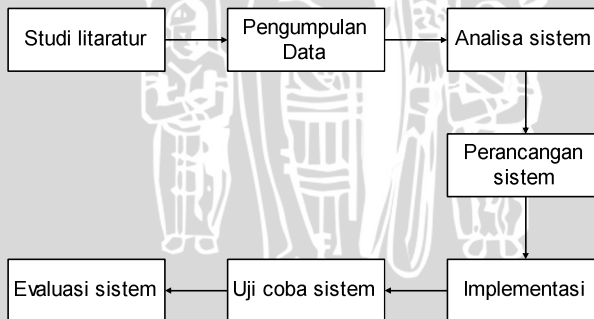


### BAB III METODOLOGI DAN PERANCANGAN

Pada bab metodologi dan perancangan ini akan dibahas metode, rancangan yang digunakan dan langkah-langkah yang dilakukan dalam penelitian tentang Jaringan syaraf tiruan *Backpropagation* dengan data *Time Series* untuk memprediksi cuaca. Penelitian dilakukan dengan tahapan sebagai berikut:

1. Mempelajari literatur yang berhubungan dengan data time series cuaca dan metode prediksi data time series dengan menggunakan jaringan syaraf *Backpropagation time series*.
2. Mengumpulkan data *time series* cuaca dari Fakultas Pertanian Brawijaya selama satu tahun yaitu tahun 2009.
3. Menganalisa dan merancang Jaringan syaraf tiruan *Backpropagation time series* untuk memprediksi cuaca.
4. Membuat perangkat lunak berdasarkan system yang telah dianalisa dan dirancang.
5. Menguji coba perangkat lunak jaringan syaraf tiruan *Backpropagation Time Series* untuk memprediksi cuaca.
6. Mengevaluasi hasil prediksi yang telah dibuat oleh system.

Alur penelitian yang dilakukan dapat digambarkan dalam bentuk diagram alir yang ditunjukkan pada Gambar 3.1.



**Gambar 3.1 Langkah-langkah Penelitian**

### 3.1 Studi Literatur

Untuk lebih memantapkan pengetahuan dalam menghasilkan tujuan dan pemecahan masalah penelitian ini dilakukan dengan studi literatur. Teori-teori mengenai jaringan syaraf tiruan, dan konsep perkiraan cuaca yang digunakan sebagai dasar penelitian ini diperoleh dari buku, jurnal, dan sumber lain.

### 3.2 Data Yang Digunakan

Data yang digunakan untuk penelitian ini adalah rekap data cuaca dari Fakultas Pertanian Universitas Brawijaya selama 1 tahun, data mulai dari januari 2009 sampai desember 2009. Data berisi dari 5 parameter cuaca yaitu, intensitas radiasi matahari, temperatur(C), kelembaban udara(RH), tekanan udara, kecepatan angin.

### 3.3 Menentukan Pola Data

Dari data yang didapat dapat disusun suatu pola/ *pattern* data yang nantinya digunakan sebagai masukan dalam pelatihan jaringan syaraf tiruan. Pola dari input terdiri dari  $x_1$  adalah temperatur,  $x_2$  adalah lama penyinaran,  $x_3$  adalah kelembaban udara,  $x_4$  adalah tekanan udara,  $x_5$  adalah kecepatan angin, serta pola keluaran/target yaitu cuaca cerah sebagai  $t_1$ , cuaca mendung sebagai  $t_2$  dan cuaca hujan sebagai  $t_3$ . Tabel pola data ditunjukkan pada tabel 3.1 dan arsitektur jaringan dapat dilihat pada gambar 3.2.

**Tabel 3.1 Pola Data**

Parameter X					Kejadian	output		
X1	X2	X3	X4	X5	Cerah	0	0	1
X1	X2	X3	X4	X5	Mendung	0	1	0
X1	X2	X3	X4	X5	Hujan	1	0	0

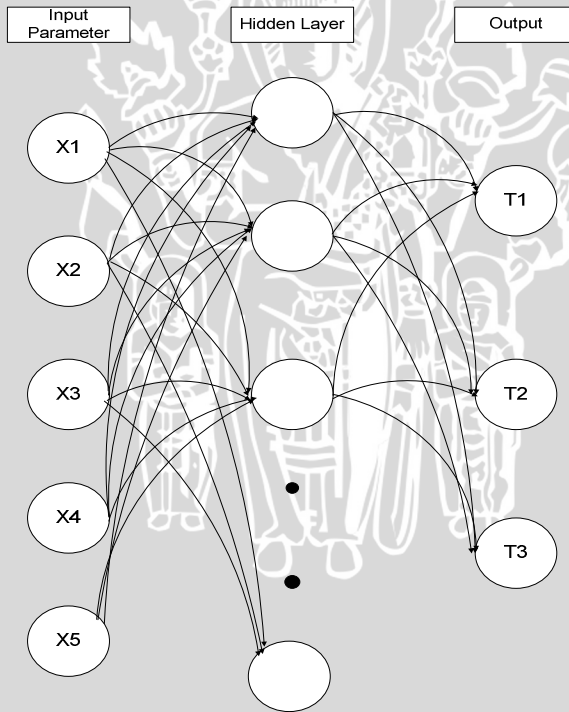
Data output akan didefinisikan agar lebih jelas dalam pengkategorianya. Dan ciri-ciri yang ada didapatkan didapatkan definisi sebagai berikut. Defenisi dari kelas output dapat dilihat pada tabel 3.2.

**Tabel 3.2 Definisi**

Kejadian	Output			Definisi				
				X1	X2	X3	X4	X5
Cerah	0	0	1	<24	>70	=60-65	<997	>=4
Mendung	0	1	0	24	=60-70	=66-79	=997	=3-4
Hujan	1	0	0	>24	<60	>79	>=99 8	=1-3

\*keterangan: 75 = 6 jam dari prosentase lama penyinaran 8 jam dalam satu hari

**Arsitektur Jaringan**



**Gambar 3.2 Arsitektur Jaringan**

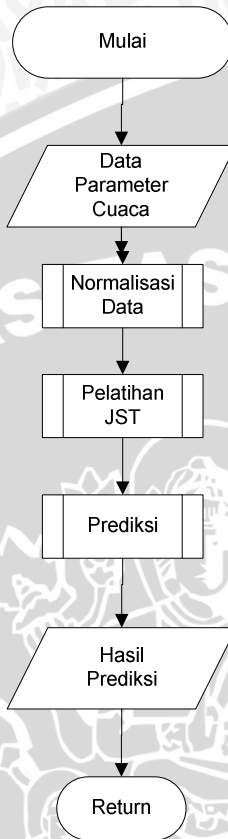


### 3.4 Deskripsi Umum Sistem

Secara umum sistem yang dibangun adalah perangkat lunak untuk menentukan kategori cuaca yang mengimplementasikan jaringan syaraf tiruan *Backpropagation*. Jaringan syaraf tiruan berfungsi sebagai pengidentifikasian pola *output*. Sistem bertujuan untuk menentukan kategori keadaan cuaca dari data hasil pengamatan selama 1 tahun, seperti yang disebutkan pada sub bab 3.2.

### 3.5 Perancangan Proses

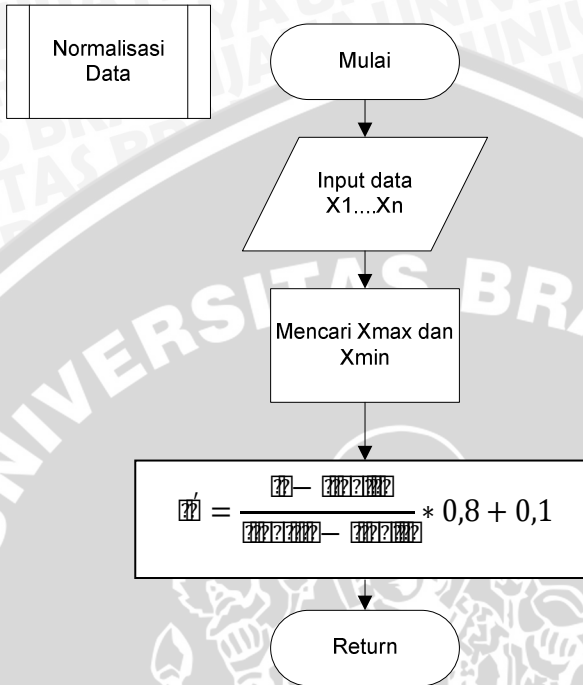
Tahapan proses dari sistem itu yaitu pengambilan data dari unsur-unsur parameter cuaca yang selanjutnya dinormalisasi dengan persamaan 2.30 . Setelah data dinormalisasi maka data siap untuk dilatihkan dalam jaringan syaraf tiruan. Apabila error minimum telah didapatkan maka sistem siap melakukan *testing* atau prediksi dengan data yang saling asing dengan data *training* sehingga menghasilkan *output* berupa keadaan cuaca yang terbagi dalam tiga kategori. Hasil dari validasi masih berupa data ternormalisasi sehingga perlu didenormalisasi agar didapatkan nilai output yang nyata (tidak dalam *range activation function*). Alur proses dalam sistem secara umum dapat dilihat pada gambar 3.2



**Gambar 3.3 Diagram Alir Sistem**

### 3.5.1 Proses Normalisasi Data

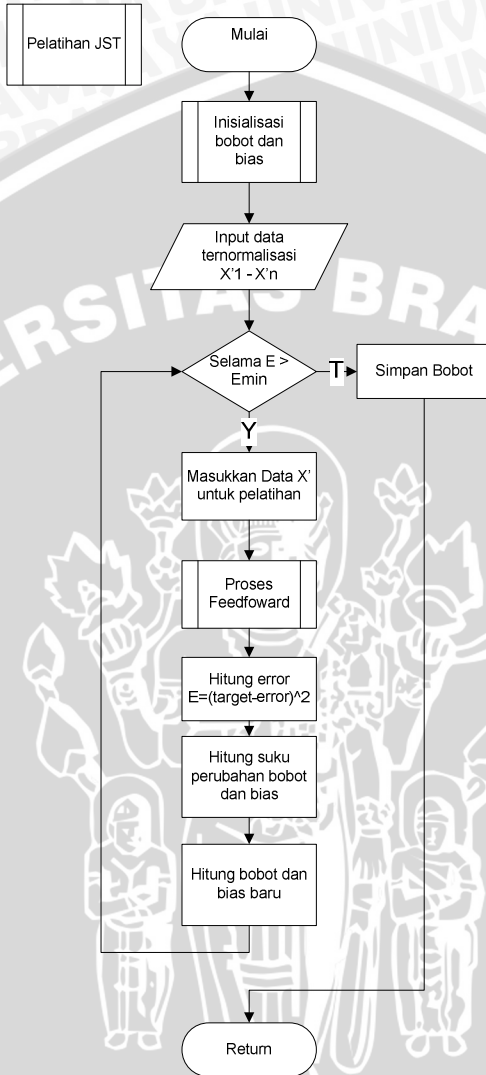
Proses ini bertujuan untuk merubah data kedalam domain fungsi aktivasi *sigmoid* yaitu dalam rentang nilai 0 sampai 1. Pada proses normalisasi ini, diperlukan input berupa data parameter cuaca ( $X_1$  sampai  $X_n$ ). Berdasarkan data input tersebut, dicari nilai maksimum ( $X_{max}$ ) dan minimum ( $X_{min}$ ). Kemudian dilakukan normalisasi sesuai dengan persamaan 2.30. Karena nilai 0 dan 1 tidak akan pernah dicapai oleh *output* maka digunakan 0,1 dan 0,9 untuk merepresentasikan nilai terkecil dan terbesar dari data (Freeman dan Skapura, 1991). Langkah-langkah proses normalisasi data ditunjukkan pada gambar 3.3.



**Gambar 3.4 Diagram Alir Proses Normalisasi**

### 3.5.2 Proses Pelatihan Jaringan Syaraf Tiruan

Proses pelatihan diawali dengan inisialisasi bobot awal untuk lapisan tersembunyi dan lapisan *output* dengan nilai random yang kecil. Kemudian memasukkan data training yang sudah dinormalisasi sebelumnya. Selama *error* masih lebih besar dari *error* minimum maka dilakukan langkah 3 sampai 9 yang sudah dijabarkan pada sub bab 2.3.2.2 mengenai pelatihan *Backpropagation*.

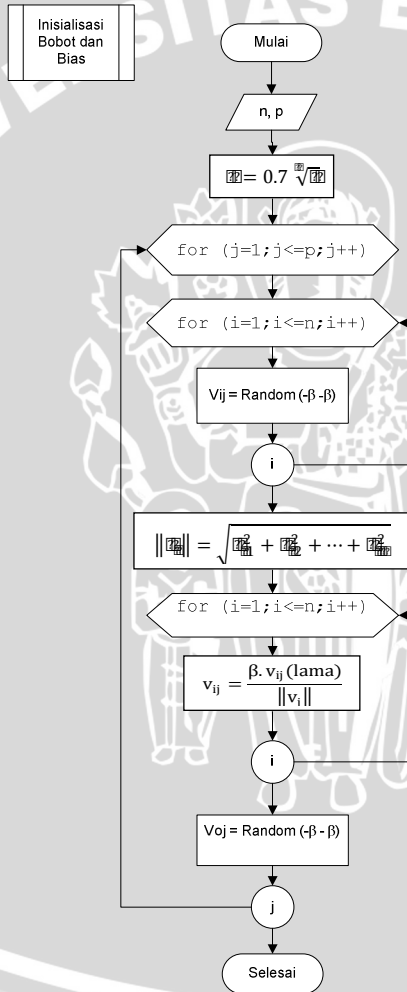


**Gambar 3.5 Diagram Alir Pelatihan JST**

### 3.5.3 Proses Inisialisasi Bobot dan Bias

Pada gambar 3.5 ditunjukkan diagram alir proses inisialisasi bobot dan bias menggunakan metode Nguyen-Widrow, masukkan berupa  $n$  (jumlah *input unit*) dan  $p$  (jumlah *hidden unit*). Nilai input tersebut digunakan untuk menghitung  $\beta$  menggunakan persamaan

2.26. Setelah diperoleh  $\beta$  kemudian dicari  $V_{ij}$  (bobot awal) secara random dengan kisaran nilai antara  $-\beta$  sampai  $\beta$ , dimana  $i$  merupakan jumlah iterasi ke-1 sampai  $n$  dan  $j$  merupakan iterasi ke-1 sampai  $p$ . Kemudian dicari  $v_j$  dengan persamaan 2.27. Berdasarkan nilai  $v_j$  yang telah diperoleh, maka dihitung  $V_{ij}$  yang baru dengan persamaan 2.28. proses dilanjutkan dengan mencari  $V_{oj}$  dengan nilai random antara  $-\beta$  sampai  $\beta$ . Langkah-langkah tersebut secara rinci dijelaskan pada sub bab 2.3.3 dan ditampilkan pada Gambar 3.6.

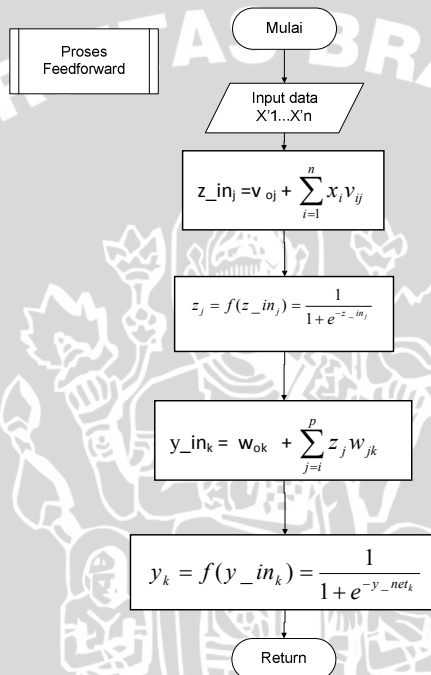


**Gambar 3.6 Diagram Alir Inisialisasi Bobot dan Bias**



### 3.5.4 Proses Feedforward

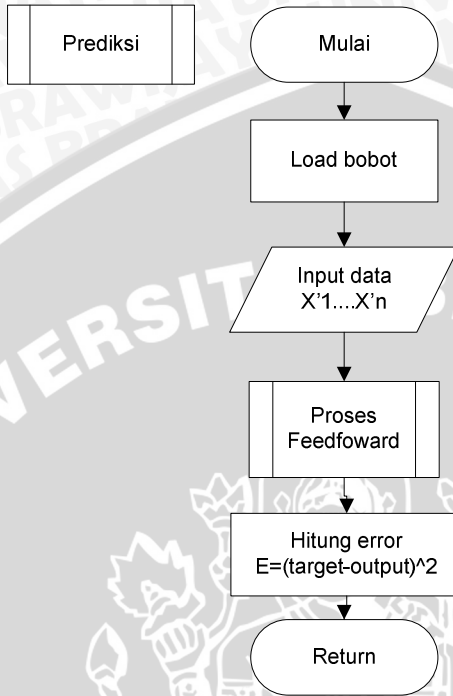
Pada proses ini, data input dirambatkan maju hingga *hidden layer* dan diaktivasi menggunakan fungsi sigmoid seperti yang ditunjukkan pada persamaan 2.3. Hasil aktivasi tersebut dirambatkan lagi menuju lapisan *output*. Untuk langkah-langkah lebih rinci telah dijelaskan pada sub bab 2.3.2.2.



Gambar 3.7 Diagram Alir Feedforward

### 3.5.5 Proses Prediksi

Proses prediksi merupakan pengecekan tingkat kebenaran sistem dalam menentukan prediksi cuaca. Proses ini diawali dengan mengambil bobot dari hasil *training*. Kemudian memasukkan data, yang akan digunakan untuk proses *feedforward*. Data tersebut merupakan data selain dari yang digunakan untuk *training*. Setelah itu dihitung tingkat kebenarannya dengan menghitung selisih antara target dan output. Langkah-langkah validasi ditunjukkan oleh diagram alir pada Gambar 3.6.



**Gambar 3.8 Diagram Alir Prediksi**

### 3.6 Perhitungan Manual

Diketahui data sampel dari penelitian 5 parameter tentang cuaca pada tabel 3.6.

**Tabel 3.6 Data Masukan**

	Temperatur (C°)	Lama penyinaran (%)	Kelembaban (RH)	Tekanan udara (mbr)	Kecepatan angin (km/jam)	T1	T2	T3
1	22,8	25	85,5	998	4	0	0	1
2	23,8	13	76,5	998	2	0	0	1
3	23,6	6	78,5	997	2,1	0	0	1

Langkah yang pertama yaitu untuk menentukan pola data pelatihan yaitu  $x_1$  adalah temperatur,  $x_2$  adalah lama penyinaran,  $x_3$  adalah kelembaban udara,  $x_4$  adalah tekanan udara,  $x_5$  adalah kecepatan angin, sedangkan targetnya adalah cuaca cerah ( $t_1$ ), cuaca mendung ( $t_2$ ), cuaca hujan ( $t_3$ ). Kemudian dilakukan normalisasi data input dengan persamaan 2.30. Hasil dari normalisasi data ditunjukkan pada tabel 3.7.

**Tabel 3.7 Tabel Hasil Normalisasi Data Awal**

no	X1	X2	X3	X4	X5	t1	t2	t3
1	0,1	0,9	0,9	0,9	0,9	0	0	1
2	0,9	0,395	0,1	0,1	0,5	0	0	1
3	0,9	0,1	0,278	0,101	0,48	0	0	1

Parameter awal untuk pelatihan yaitu

$$\alpha = 0.1$$

$$\varepsilon = 0.2$$

jumlah unit *input* = 6

jumlah unit di *hidden layer* = 6

jumlah unit *output* = 3

Inisialisasi bobot awal dengan metode Nguyen Widrow. Pertama-tama nilai faktor  $\beta$  dihitung dengan menggunakan persamaan 2.26, maka didapat :

$$\beta = 0.7 * 6^{1/6} = 0.94$$

Kemudian Nilai bobot awal diberi nilai random yang kecil (antara -  $\beta$  sampai  $\beta$ ) kemudian dihitung  $\|v_j\|$  dan  $\|w_k\|$  dengan persamaan 2.27 dan hasilnya pada tabel 3.8 dan 3.9 berikut:

**Tabel 3.8 Tabel Data Nilai Bobot Awal  $V_{ij}$**

	Z1	Z2	Z3	Z4	Z5	Z6
X1	0,34	0,87	-0,62	-0,42	-0,66	-0,69
X2	0,45	0,83	-0,49	-0,83	0,48	-0,30
X3	-0,32	0,38	-0,82	-0,54	0,91	-0,4

<b>X4</b>	0,21	0,32	-0,75	-0,75	-0,51	-0,74
<b>X5</b>	0,81	-0,44	-0,91	0,63	-0,68	0,10
<b>  vj  </b>	1,0586	1,3734	1,6394	1,4548	1,4888	1,1330

**Tabel 3.9 Tabel Data Nilai Bobot Awal  $W_{jk}$**

	<b>T1</b>	<b>T2</b>	<b>T3</b>
<b>Z1</b>	-0,81	-0,94	-0,74
<b>Z2</b>	0,42	-0,62	-0,23
<b>Z3</b>	0,83	-0,4	0,74
<b>Z4</b>	-0,21	-0,1	0,21
<b>Z5</b>	0,23	0,60	-0,32
<b>Z6</b>	-0,58	-0,35	0,86
<b>  wk  </b>	1,3981	1,3858	1,4262

Setelah itu dilakukan perhitungan  $V_{ij}$  dan  $W_{jk}$  baru dengan persamaan 2.28, dan hasilnya pada tabel 3.10 dan 3.11

**Tabel 3.10 Tabel data  $V_{ij}$  Dengan Nguyen Widrow**

	<b>Z1</b>	<b>Z2</b>	<b>Z3</b>	<b>Z4</b>	<b>Z5</b>	<b>Z6</b>
<b>X1</b>	0,3019	0,5955	-0,3555	-0,2714	-0,4167	-0,5725
<b>X2</b>	0,3996	0,5681	-0,2810	-0,5363	0,3031	-0,2489
<b>X3</b>	-0,2841	0,2601	-0,4702	-0,3489	0,5746	-0,3319
<b>X4</b>	0,1865	0,2190	-0,4300	-0,4846	-0,3220	-0,6139
<b>X5</b>	0,7193	-0,3012	-0,5218	0,4071	-0,4293	0,0830

**Tabel 3.11** Tabel data  $W_{jk}$  Dengan Nguyen Widrow

	<b>T1</b>	<b>T2</b>	<b>T3</b>
<b>Z1</b>	-0,5445	-0,6376	-0,4877
<b>Z2</b>	0,2823	-0,4205	-0,1515
<b>Z3</b>	0,5580	-0,2713	0,4877
<b>Z4</b>	-0,1411	-0,0678	0,1384
<b>Z5</b>	0,1546	0,4069	-0,2109
<b>Z6</b>	-0,3899	-0,2374	0,5667

Nilai bias  $V_{oj}$  dan  $W_{ok}$  adalah bilangan acak antara  $-\beta$  sampai  $\beta$ . Dan nilai  $V_{oj}$  dan  $W_{ok}$  dapat dilihat pada tabel 3.12 dan 3.13

**Tabel 3.12** Tabel Nilai Bias  $V_{oj}$

<b>J</b>	<b><math>V_{oj}</math></b>
<b>1</b>	-0,55
<b>2</b>	0,73
<b>3</b>	-0,66
<b>4</b>	0,92
<b>5</b>	-0,52
<b>6</b>	-0,73

**Tabel 3.13** Tabel Nilai Bias  $W_{ok}$

<b>K</b>	<b><math>W_{ok}</math></b>
<b>1</b>	-0,77
<b>2</b>	-0,69
<b>3</b>	0,83



## Proses Feedforward

### Pola data 1

Menghitung keluaran di unit tersembunyi( $z_j$ ) dengan persamaan 2.11 dan dilanjutkan dengan perhitungan aktivasinya dengan persamaan 2.12. Hasil perhitungannya ditunjukkan pada tabel 3.14 dan 3.15.

**Tabel 3.14 Tabel operasi pada hidden ( $Z_{in}$ )**

J	$Z_{in}$
1	0,3992
2	-0,9113
3	-1,1482
4	0,8929
5	-0,5617
6	-0,7872

**Tabel 3.15 Tabel Hasil Aktivasi  $Z_{in}$**

J	Z
1	0,4015
2	0,2868
3	0,7592
4	0,2905
5	0,6368
6	0,6872

Keluaran unit output dihitung dengan menggunakan persamaan 2.13 dan 2.14, sehingga menghasilkan  $y_{in}$  pada tabel 3.16 dan menghasilkan hasil aktivasi  $y_{in}$  menghasilkan  $y$  yang ditunjukkan pada tabel 3.17.

**Tabel 3.16** Tabel operasi pada output ( $Y_{in}$ )

K	$Y_{in}$
1	-0,67949
2	-1,19156
3	1,2580952

**Tabel 3.17** Tabel Hasil Aktivasi ( $Y_{in}$ )

K	Y
1	0,663609
2	0,766998
3	0,221324

Setelah diperoleh nilai Y (hasil aktivasi) kemudian dicari *error* dengan mencari selisih antara  $(Y - \text{target})^2$ . Jika nilai *error* masih diatas nilai target *error*, maka dilakukan perubahan bobot dengan proses *Backpropagation*.

### **Proses Backpropagation**

Hitung nilai kemelesetan  $y_k$  dengan persamaan 2.15 dan hasil nilai kemelesetan dapat dilihat pada tabel 3.18.

$$\delta_k = (t_k - y_k) f'(y_{in_k}) = (t_k - y_k) y_k (1 - y_k)$$

$$\begin{aligned} \delta_1 &= (t_k - y_k) f'(y_{in_k}) \\ &= (0 - 0,663609) * 0,663609 (1 - 0,663609) = -0,148138 \end{aligned}$$

**Tabel 3.18** Tabel Nilai Kemelesetan  $y_k$

K	$\delta$
1	-0,148138
2	-0,137071
3	0,1341968

Hitung Nilai koreksi bobotnya menggunakan persamaan 2.16 dengan  $\alpha = 0,1$ , nilai koreksi bobot pada unit output dapat dilihat pada tabel 3.19.

**Tabel 3.19** Tabel  $\Delta W_{jk}$

$\Delta W_{jk}$	$\Delta W_1$	$\Delta W_2$	$\Delta W_3$
Z1	-0,005914	-0,005472	0,005358
Z2	-0,013499	-0,012491	0,012229
Z3	0,017010	0,015739	-0,015409
Z4	-0,013227	-0,012239	0,011982
Z5	0,008321	0,007699	-0,007537
Z6	0,011662	0,010791	-0,010565

Koreksi bias bobot  $\Delta W_{ok}$  dengan persamaan  $\Delta W_{01} = \alpha * \delta_k$ , Nilai koreksi bias pada unit *output* dapat dilihat pada tabel 3.20

**Tabel 3.20** Tabel Koreksi nilai bias *output* ( $W_{ok}$ )

K	$W_{ok}$
1	-0,0148138
2	-0,0137071
3	0,01341968

Hitung nilai kemelesatan  $Z_j$  dengan persamaan 2.17, dan tabel nilai kemelesatan  $y_j$  dapat dilihat pada tabel 3.21.

**Tabel 3.21** Tabel  $\delta_{in_j}$

J	$\delta_{in_j}$
1	0,002345
2	0,005353
3	-0,006745
4	0,005245
5	-0,003299
6	-0,004624

Hitung faktor kesalahan  $\delta$  di *hidden* unit dengan persamaan 2.18. Faktor kesalahan  $\delta$  dapat dilihat pada tabel 3.22.

**Tabel 3.22** Tabel  $\delta_j$

J	$\delta$
1	0,000564
2	0,001095
3	-0,001233
4	0,001081
5	-0,000763
6	-0,000994

Pakai nilai koreksi kemelesatan untuk memperbaharui nilai  $V_{ij}$  dengan persamaan 2.19. Nilai koreksi pada unit hidden dapat dilihat pada tabel 3.23.

**Tabel 3.23** Tabel  $\Delta V_{ij}$ 

$\Delta V_{ij}$	$\Delta V_1$	$\Delta V_2$	$\Delta V_3$	$\Delta V_4$	$\Delta V_5$
<b>Z1</b>	0,000006	0,000051	0,000051	0,000051	0,000051
<b>Z2</b>	0,000011	0,000099	0,000099	0,000099	0,000099
<b>Z3</b>	-0,000012	-0,000111	-0,000111	-0,000111	-0,000111
<b>Z4</b>	0,000011	0,000097	0,000097	0,000097	0,000097
<b>Z5</b>	-0,000008	-0,000069	-0,000069	-0,000069	-0,000069
<b>Z6</b>	-0,000010	-0,000089	-0,000089	-0,000089	-0,000089

Koreksi bias bobot  $\Delta V_{oj}$  dengan persamaan  $\Delta V_{oj} = \alpha \delta_j$ . Dan nilai koreksi bobot bias dapat dilihat pada tabel 3.24.

**Tabel 3.24** Tabel  $\Delta V_{oj}$ 

<b>J</b>	$\Delta V_{oj}$
<b>1</b>	0,0000564
<b>2</b>	0,0001095
<b>3</b>	-0,0001233
<b>4</b>	0,0001081
<b>5</b>	-0,0000763
<b>6</b>	-0,0000994

Perbaharui bobot di unit *hidden* dengan persamaan 2.22. Bobot baru di unit *hidden* dapat dilihat pada tabel 3.25.



**Tabel 3.25 Tabel  $V_{ij}$  Baru**

$\Delta V_{ij}$	$\Delta V_1$	$\Delta V_2$	$\Delta V_3$	$\Delta V_4$	$\Delta V_5$
<b>Z1</b>	0,3019	0,3996	-0,2841	0,1865	0,7193
<b>Z2</b>	0,5955	0,5682	0,2602	0,2191	-0,3011
<b>Z3</b>	-0,3555	-0,2811	-0,4703	-0,4301	-0,5219
<b>Z4</b>	-0,2714	-0,5362	-0,3488	-0,4845	0,4072
<b>Z5</b>	-0,4167	0,3030	0,5745	-0,3221	-0,4294
<b>Z6</b>	-0,5725	-0,2490	-0,3320	-0,6140	0,0829

Kemudian perbaiki nilai bias  $V_{oj}$ , Dengan persamaan 2.23. Nilai bias  $V_{oj}$  dapat dilihat pada tabel 3.26.

**Tabel 3.26 Tabel  $V_{oj}$  Baru**

<b>J</b>	<b><math>V_{oj}</math> Baru</b>
<b>1</b>	-0,550056
<b>2</b>	0,729891
<b>3</b>	-0,659877
<b>4</b>	0,919892
<b>5</b>	-0,519924
<b>6</b>	-0,729901

Perbaharui bobot di unit *output* dengan persamaan 2.20. Bobot baru di unit output dapat dilihat pada tabel 3.27.

**Tabel 3.27 Tabel  $W_{jk}$  Baru**

$W_{jk}$	$W_1$	$W_2$	$W_3$
<b>Z1</b>	-0,55041	-0,64307	-0,48234
<b>Z2</b>	0,26880	-0,43299	-0,13927
<b>Z3</b>	0,57501	-0,25556	0,47229
<b>Z4</b>	-0,15433	-0,08004	0,15038
<b>Z5</b>	0,16292	0,41460	-0,21844
<b>Z6</b>	-0,37824	-0,22661	0,55614

Kemudian perbaiki nilai bias  $W_{ok}$  dengan persamaan 2.21. Nilai bias  $W_{ok}$  dapat dilihat pada tabel 3.28.

**Tabel 3.28 Tabel  $W_{ok}$  Baru**

<b>K</b>	$W_{ok}$
<b>1</b>	-0,918148
<b>2</b>	-0,826877
<b>3</b>	0,964274

Setelah dilakukan perbaikan bobot sampai didapatkan error minimal atau max epoch, maka hasilnya perubahan bobot akan disimpan untuk pada saat ada data baru yang akan dilakukan prediksi.

### 3.7 Perancangan Uji Coba

Pada bagian perancangan uji coba ini akan dijelaskan mengenai pengujian data. Dalam uji coba ini menggunakan data *input* cuaca selama satu tahun dengan rentan waktu satu hari dan sebagai *training data* adalah data selama 6 bulan pertama, kemudian dilakukan validasi dengan data dengan data 6 bulan berikutnya. Dengan data *training* yang sama dilakukan pengulangan uji coba dengan perubahan jumlah *hidden* unit dan *learning rate*.

### 3.7.1.1 Pengaruh Jumlah *Hidden Unit* dan *Learning Rate*

Untuk mengetahui kecepatan jaringan dalam mencapai *error* minimal yang ditentukan berdasarkan jumlah *hidden unit*. Pengujian dilakukan dengan merubah-ubah jumlah *hidden unit* dengan jumlah *epoch* maksimal yang tetap dan *learning rate* yang diujikan yaitu 0.1 sampai 0,9. Bentuk data hasil percobaan tersebut akan disajikan pada tabel 3.3. Jumlah *hidden unit* dapat direpresentasikan dalam 1 *hidden layer*.

**Tabel 3.3 Pengaruh Jumlah *Neuron Hidden* dan *Learning Rate* Terhadap MSE**

NO	Jumlah Neuron Hidden	Learning rate	MSE

Setelah *training* selesai dilakukan maka tahap selanjutnya yaitu dilakukan proses prediksi. Dari data hasil prediksi akan dibandingkan dengan data *real* atau data asli parameter-parameter cuaca. Format hasil prediksi dapat ditunjukkan pada tabel 3.5.

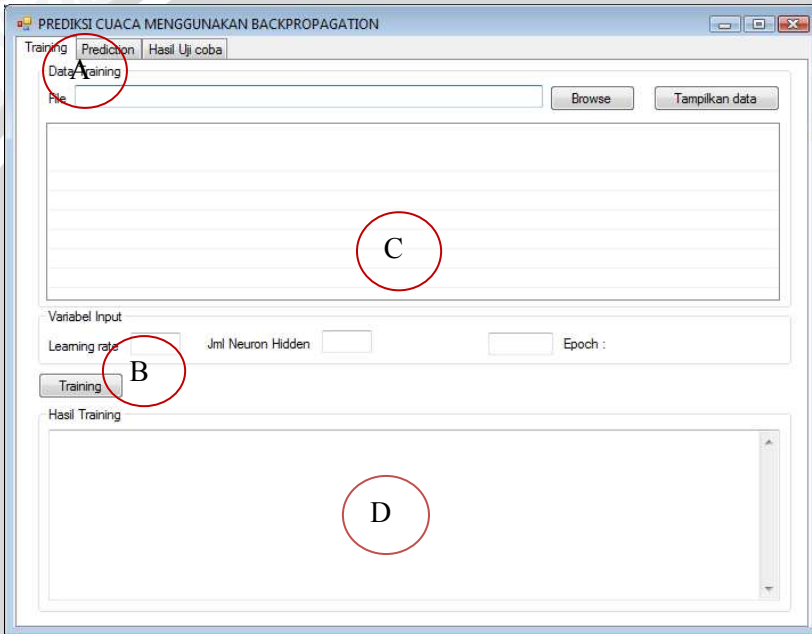
**Tabel 3.4 Validasi hasil Prediksi Cuaca**

Data (NO)	Hasil Prediksi	Real

### 3.8 Perancangan Antarmuka

Untuk perancangan antar muka prediksi cuaca ini terdiri dari 2 form, form *training*, dan form *forecast*. Untuk form *training* dapat dilihat pada gambar 3.1, dan form *forecast* dapat dilihat pada gambar 3.2.

#### Form *Training*



**Gambar 3.9 Form *Training***

Form training merupakan form untuk melakukan proses training dari data parameter cuaca. Pada form *training* ini terdapat 4 bagian, yaitu:

- A : Menu navigasi
- B : Menu yang berisi pilihan untuk mengatur arsitektur jaringan syaraf tiruan dan mengatur nilai parameter yang akan digunakan dalam proses training.
- C : Menampilkan data parameter cuaca
- D : Menampilkan hasil *training*

Form ini ditunjukkan oleh gambar 3.1.

## Form Prediction

PREDIKSI CUNCA MENGGUNAKAN BACKPROPAGATION

Training Prediction Hasil Uji coba

Data Uji

file uji  Browse Tampilkan data Uji

Prediksi

Hasil Ramalan

No	Hari ke-	Data real	Data peramalan JST

**Gambar 3.10 Form Prediction**

*Form forecast* merupakan form untuk melakukan proses prediksi (*forecast*) dari data yang telah di *training* sebelumnya. Pada *form forecast* ini terdapat 4 bagian, yaitu :

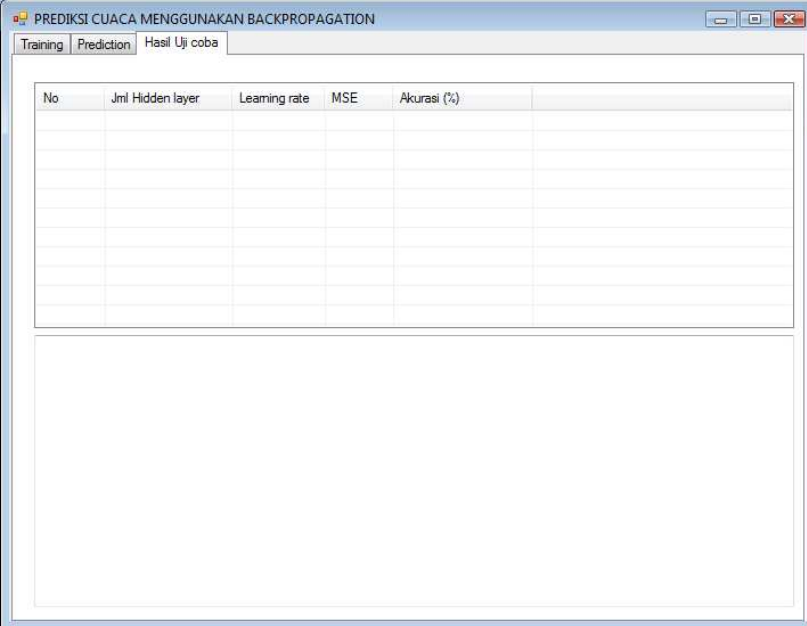
Pada *form forecast* terdapat 4 bagian yaitu :

- A : Menu navigasi
- B : Mengambil dan menampilkan data uji
- C : Data inputan yang berupa data uji
- D : Hasil prediksi

Form ini ditunjukkan oleh gambar 3.2.



## Form Hasil Uji Coba



The screenshot shows a software window with the title "PREDIKSI CUACA MENGGUNAKAN BACKPROPAGATION". It has three tabs: "Training", "Prediction", and "Hasil Uji coba". The "Hasil Uji coba" tab is active and contains a table with the following columns: "No", "Jml Hidden layer", "Leaming rate", "MSE", and "Akurasi (%)". The table is currently empty, with only the header row visible. Below the table is a large empty rectangular area.

No	Jml Hidden layer	Leaming rate	MSE	Akurasi (%)

**Gambar 3.11** *Form Hasil Uji Coba*

Pada *form* ini terdapat tampilan hasil dari beberapa percobaan prediksi cuaca dengan perubahan nilai inputan *learning rate* dan jumlah *neuron hidden*. Dan *form* menu hasil uji coba dapat dilihat pada gambar 3.3.

## BAB IV

### IMPLEMENTASI DAN PEMBAHASAN

#### 4.1 Lingkungan Implementasi

Lingkungan implementasi meliputi lingkungan perangkat keras serta lingkungan perangkat lunak.

##### 4.1.1 Lingkungan perangkat keras

Perangkat keras yang digunakan dalam pembangunan sistem prediksi cuaca menggunakan jaringan syaraf tiruan *Backpropagation* ini adalah:

1. Prosesor Intel(R) Centrino Duo(R) 1.50 GHz.
2. RAM 1014 MB
3. *Harddisk* dengan kapasitas 120 GB
4. Monitor
5. *Keyboard*
6. *Mouse*

##### 4.1.2 Lingkungan perangkat lunak

Perangkat lunak yang digunakan dalam pembangunan sistem prediksi cuaca ini adalah :

1. Sistem Operasi *Microsoft Windows Vista*
2. *Microsoft Visual C# 2008*

#### 4.2 Implementasi Program

Pada sub bab ini akan dibahas mengenai implementasi dari sistem prediksi cuaca menggunakan jaringan syaraf tiruan *Backpropagation*.

## 4.2.1 Struktur Data

Struktur data yang digunakan untuk menyimpan data-data parameter cuaca, data tanggal dan bulan, dan kondisi cuaca kedalam sistem adalah *array*. Struktur data tersebut seperti pada *sourcecode* 4.1.

```
class StructRawData
{
    public int[]    dateMonth;    //index pertama
    bulan, index kedua tanggal

    public double[] rawX;        //data real
    parameter cuaca

    public int[]   weather;      //weather
    condition; 0 : cerah, 1 : mendung, 2 : hujan
}
}
```

### *sourcecode* 4.1 Struktur Data

Dari struktur data tersebut dapat dijelaskan bahwa penyimpanan tanggal dan bulan merupakan *array* dari tipe data *integer*. Penyimpanan data setiap parameter cuaca juga merupakan *array* dari tipe data *real*. Untuk menyimpan data kondisi cuaca juga menggunakan *array* dengan tipe data *integer*.

## 4.2.2 Proses Ambil File Data Parameter Cuaca

Fungsi ini adalah untuk mengambil file data asli parameter-parameter cuaca yaitu, temperatur, lama penyinaran, kelembaban, tekanan udara, kecepatan angin ke aplikasi. Proses ini terdapat pada class Form1. Prosedur pengambilan data mentah parameter-parameter cuaca ini lebih lanjut dapat dilihat pada *sourcecode* 4.4.

```

private void btTampil_Click(object sender, EventArgs e)
{
    lvRawData.Items.Clear();
    if (!tbFileTrain.Text.Equals(""))
    {
        try
        {
            //IoFile instance
            IoFile inputFile = new IoFile(tbFileTrain.Text);
            List<StructRawData> dataFromFile = new
            List<StructRawData>();

            dataFromFile = inputFile.openFile3();
            rawData = dataFromFile;

            //menampilkan data dari file ke listview
            HeaderDataColumn();

            for (int i = 0; i < dataFromFile.Count; i++)
            {
                StructRawData bufRawData = new StructRawData();
                bufRawData = dataFromFile[i];

                ListViewItem lvItem = new
                ListViewItem((i+1).ToString());

                lvItem.SubItems.Add(bufRawData.dateMonth[0].ToString());
                lvItem.SubItems.Add(bufRawData.dateMonth[1].ToString());
                lvItem.SubItems.Add(string.Format("{0:0.0}",
                bufRawData.rawX[0]));
                lvItem.SubItems.Add(bufRawData.rawX[1].ToString());
                lvItem.SubItems.Add(string.Format("{0:0.0}",
                bufRawData.rawX[2]));
                lvItem.SubItems.Add(bufRawData.rawX[3].ToString());
                lvItem.SubItems.Add(string.Format("{0:0.0}",
                bufRawData.rawX[4]));
            }
        }
    }
}

```

```

        if (bufRawData.weather[0] == 1)
            lvItem.SubItems.Add("sunny");
        else if (bufRawData.weather[1] == 1)

            lvItem.SubItems.Add("cloudy");
        else lvItem.SubItems.Add("rain");
        lvRawData.Items.Add(lvItem);
    }

```

## Sourcecode 4.2 Proses Ambil File Data Parameter Cuaca

### 4.2.3 Proses Normalisasi

Prosedur normalisasi digunakan untuk melakukan normalisasi terhadap data mentah dari parameter cuaca agar *range* data yang didapatkan sebagai hasil berkisar dari 0,1 – 0,9.

Fungsi proses normalisasi yang terdapat pada *class* NormalizeData dapat dilihat pada gambar 4.5.

```

private List<StructRawData>
NormalizingData(List<StructRawData> rawData)
{
    List<StructRawData>result = new List<StructRawData>();
    for (int i = 0; i < rawData.Count; i++)
    {
        //data untuk normalisasi
        StructRawData oldBufData = new StructRawData();
        //data normalisasi
        StructRawData newBufData = new StructRawData();
        oldBufData = rawData[i];

        newBufData.dateMonth = new int[2];
        newBufData.dateMonth[0] = oldBufData.dateMonth[0];
        newBufData.dateMonth[1] = oldBufData.dateMonth[1];
    }
}

```



```

newBufData.weather = new Int32[3];
newBufData.weather = oldBufData.weather;
//normalisasi data
newBufData.rawX = new double[oldBufData.rawX.Length];
for (int j = 0; j < newBufData.rawX.Length; j++)
{
    if ((oldBufData.rawX.Length == 5) &&
(oldBufData.rawX.Length == maxValue.Length))
    {
        try
        {
newBufData.rawX[j] = (oldBufData.rawX[j] - minValue[j])
/ (maxValue[j] - minValue[j]) * upperLimit + lowerLimit;
        }
        catch(Exception ex)
        {
            MessageBox.Show("Kemungkinan panjang data
tidak sama\n" + ex.ToString());
        }
    }
    result.Add(newBufData);
}

return result;
}

```

### ***Sourcecode 4.3 Proses Normalisasi Data***

#### 4.2.4 Proses Pengkategorian Cuaca

Pengkategorian cuaca bertujuan untuk menentukan suatu kondisi cuaca pada suatu hari dengan berbagai paramater cuaca yang ada. Dan fungsi proses pengkategorian cuaca yang terdapat pada class IoFile dapat dilihat pada gambar 4.6.

```
private int[] DefineWeather(double[] param)
{
    int[] weather = new int[3];
    int cerah = 0;
    int mendung = 0;
    int hujan = 0;
    int x2 = 0;

    //decide temperatur
    if (param[0] < 24)
        cerah++;
    else if (param[0] == 24)
        mendung++;
    else hujan++;

    //decide lama penyinaran
    if (param[1] > 70)
    {
        cerah++;
        x2 = 0;
    }
    else if (param[1] < 60)
    {
        hujan++;
        x2 = 2;
    }
    else
    {
        mendung++;
        x2 = 1;
    }

    //decide kelembaban
    if (param[2] < 66)
        cerah++;
    else if (param[2] > 79)
        hujan++;
    else mendung++;
}
```

```

//decide tekanan udara
if (param[3] < 997)
    cerah++;
else if (param[3] == 997)
    mendung++;
else hujan++;

//decide kecepatan angin
if (param[4] >= 4)
    cerah++;
else if (param[4] <= 3)
    hujan++;
else mendung++;

if ((cerah > mendung) && (cerah > hujan))
{
    weather[0] = 1;
    weather[1] = 0;
    weather[2] = 0;
}
else if ((mendung > cerah) && (mendung > hujan))
{
    weather[0] = 0;
    weather[1] = 1;
    weather[2] = 0;
}
else if ((hujan > cerah) && (hujan > mendung))
{
    weather[0] = 0;
    weather[1] = 0;
    weather[2] = 1;
}

else
{
    if ((cerah == 2) && (x2 == 0))
    {
        weather[0] = 1;
        weather[1] = 0;
        weather[2] = 0;
    }
}

```

```

else if ((hujan == 2) && (x2 == 2))
    {
        weather[0] = 0;
        weather[1] = 0;
        weather[2] = 1;
    }
else if ((mendung == 2) && (x2 == 1))
    {
        weather[0] = 0;
        weather[1] = 1;
        weather[2] = 0;
    }
else
    {
        weather[0] = 0;
        weather[1] = 1;
        weather[2] = 0;
    }
}

return weather;
}

```

#### **Sourcecode 4.4 Proses Pengkategorian Cuaca**

### **4.2.5 Proses Training**

Proses training terdapat 2 sub proses, yaitu sub proses *feedforward* dan sub proses *backpropagation*.

#### **4.2.5.1 Proses Feedforward**

Prosedur *feedforward* terdiri dari prosedur  $Z_{inj}$  yang digunakan untuk mencari nilai *input* dari *neuron hidden* dan *neuron output* dan  $Z_j$  yang digunakan untuk mengaktifasi nilai  $Z_{inj}$  dengan menggunakan fungsi *sigmoid*. Fungsi dari proses *feedforward* ini terdapat pada class *Backpropagation*.

#### 4.2.5.1.1 Prosedur Mencari Z\_inj

Prosedur Z\_inj ini adalah prosedur yang digunakan untuk mencari nilai *neuron hidden* dari *input* yang nilai tersebut akan aktifasi dengan menggunakan fungsi *sigmoid*. Prosedur lebih lanjut dapat dilihat pada sourcecode 4.5.

```
public double DoFeedForward(StructRawData dailyData)
{
    double[] bufOutput = new double[3];
    input = new double[5];
    outputNetworkAfterActivation = new double[3];
    double[] deltaTargetOutput = new double[3];

    try
    {
        StructRawData bufData = dailyData;

        double[] bufInput = new double[bufData.rawX.Length];
        input = bufInput;
        double[] bufHidden = new double[nodeHidden];
        outputHiddenBeforeActvivation = new double[nodeHidden];
        double[] hiddenActivation = new double[nodeHidden];
        double[] outputActvivation = new double[3];

        //total hidden
        for (int j = 0; j < input2Hidden.GetLength(1); j++)
        {
            //total input + 1 bias
            for (int i = 0; i < input2Hidden.GetLength(0); i++)
            {
                if (i < input2Hidden.GetLength(0) - 1)
                    bufHidden[j] += input2Hidden[i, j] * bufData.rawX[i];
                else
                    bufHidden[j] += input2Hidden[i, j];
            }
            outputHiddenBeforeActvivation[j] = bufHidden[j];
            hiddenActivation[j] = 1 / (1 + Math.Pow(2.718, -
                bufHidden[j]));
        }
    }
}
```

**Sourcecode 4.5 Prosedur Mencari Z\_inj**



#### 4.2.5.1.2 Prosedur Mencari $Y_{ink}$

Prosedur ini adalah prosedur untuk mencari nilai *output* dari *neuron hidden*, yang nantinya keluaran tersebut akan dilakukan aktifasi menggunakan fungsi *sigmoid*. Prosedur lebih lanjut dapat dilihat pada *sourcecode* 4.6.

```
for (int j = 0; j < hidden2Output.GetLength(1); j++)
//total output
{
//total hidden + 1 bias
for (int i = 0; i < hidden2Output.GetLength(0); i++)
{
if (i < hidden2Output.GetLength(0) - 1)
    bufOutput[j] += hidden2Output[i, j] *
hiddenActivation[i];
else
    bufOutput[j] += hidden2Output[i, j];
}
outputActivation[j] = 1 / (1 + Math.Pow(2.718, -
bufOutput[j]));
deltaTargetOutput[j] = Math.Abs(bufData.weather[j] -
outputActivation[j]);
}
outputNetworkAfterActivation = outputActivation;
//menentukan hasil prediksi
double maxVal = 0;
int idx = 0;
for (int i = 0; i <
outputActivation.Length; i++)
{
    if (i == 0)
    {
        maxVal = outputActivation[i];
    }
    else
    {
        if (maxVal < outputActivation[i])
        {
            maxVal =
outputActivation[i];
            idx = i;
        }
    }
}
```

**Sourcecode 4.6** Prosedur Mencari  $Z_{inj}$

#### 4.2.5.2 Proses *Backpropagation*

Fungsi dari proses *backpropagation* ini terdapat pada class *Backpropagation* yang terdiri dari prosedur *wjk*, *d\_inj*, *d\_j*, *d\_vij*, dan prosedur *update bobot*.

##### 4.2.5.2.1 Prosedur mencari nilai *Wjk*

Prosedur *Wjk* adalah untuk mencari nilai koreksi bobot dari *neuron hidden* ke *neuron output*. Dan prosedur lebih lanjut dapat dilihat pada *sourcecode* 4.6.

```
public void doBackProp ()
{
    double[,] outputWeightCorrection = new
double[hidden2Output.GetLength(0),
    hidden2Output.GetLength(1)];

//delta Wjk
for (int j = 0; j < outputWeightCorrection.GetLength(1);
j++) //total output
    {
for (int i = 0; i < outputWeightCorrection.GetLength(0);
i++) //total hidden + 1 bias
    {
if (i < outputWeightCorrection.GetLength(0)- 1)
        outputWeightCorrection[i, j] +=
            alpha * this.deltaTarget[j]
* outputHiddenBeforeActvivation[i];
        else
            outputWeightCorrection[i, j] +=
alpha * this.deltaTarget[j];
    }
    }
}
```

**Sourcecode 4.7** Prosedur menghitung *Wjk*

#### 4.2.5.2.2 Prosedur Mencari nilai $d_{inj}$

Prosedur  $d_{inj}$  adalah untuk mencari nilai kesalahan pada unit tersembunyi. Dan prosedur lebih lanjut dapat dilihat pada *sourcecode* 4.7.

```
//d_inj
double[] deltaTargetHidden = new double[nodeHidden];
for (int j = 0; j < outputWeightCorrection.GetLength(1);
j++)
{
for (int i = 0; i < outputWeightCorrection.GetLength(0) -
1; i++)
{
deltaTargetHidden[i] += outputWeightCorrection[i, j] *
deltaTarget[j];
}
}
}
```

#### *Sourcecode* 4.8 Prosedur menghitung $d_{inj}$

#### 4.2.5.2.3 Prosedur mencari nilai $d_j$

Prosedur  $d_j$  adalah untuk mencari nilai kesalahan pada unit tersembunyi. Prosedur selebihnya dapat dilihat pada *sourcecode* 4.8.

```

//d_j
double[] faktorKesalahan = new double[nodeHidden];
for (int i = 0; i < deltaTargetHidden.Length; i++)
{
    faktorKesalahan[i] = deltaTargetHidden[i] *
    input2Hidden[input2Hidden.GetLength(0)-1,i]
    * (1 - input2Hidden[input2Hidden.GetLength(0)-1,i]);
}

double[,] inputWeightCorrection = new
double[input2Hidden.GetLength(0),
input2Hidden.GetLength(1)];

```

### **Sourcecode 4.9** Prosedur mencari $d_j$

#### **4.2.5.2.4** Prosedur mencari $d_{Vij}$

Prosedur  $d_{Vij}$  adalah prosedur untuk mencari nilai koreksi bobot dari *neuron input* ke *neuron hidden*. Untuk prosedur lebih lanjut dapat dilihat pada *sourcecode 4.9*.

```

//delta Vij
//total hidden
for (int j = 0; j < inputWeightCorrection.GetLength(1);
j++)
{
    //total input + 1 bias
    for (int i = 0; i <
inputWeightCorrection.GetLength(0); i++)
    {
        if (i < inputWeightCorrection.GetLength(0) - 1)
            inputWeightCorrection[i, j] +=
            alpha * faktorKesalahan[j] * input[i];
        else
            inputWeightCorrection[i, j] += alpha *
            faktorKesalahan[j];
    }
}

```

### **Sourcecode 4.10** Prosedur mencari $d_{Vij}$

#### 4.2.5.2.5 Prosedur *Update Bobot*

Prosedur *update* bobot ini digunakan untuk mendapatkan nilai bobot yang baru. Selebihnya prosedur dapat dilihat pada *sourcecode* 4.8.

```
//lakukan update bobot

//total hidden
for (int j = 0; j < input2Hidden.GetLength(1); j++)
{
    //total input + 1 bias
    for (int i = 0; i < input2Hidden.GetLength(0); i++)
    {
        input2Hidden[i, j] = input2Hidden[i, j] +
        inputWeightCorrection[i, j];
    }
}

//total output
for (int j = 0; j < hidden2Output.GetLength(1); j++)
{
    //total hidden + 1 bias
    for (int i = 0; i < hidden2Output.GetLength(0); i++)
    {
        hidden2Output[i, j] = hidden2Output[i, j] +
        outputWeightCorrection[i, j];
    }
}
}
```

#### ***Sourcecode 4.11* Prosedur *Update Bobot***

Secara keseluruhan prosedur *backpropagation* ini digunakan untuk memperbaiki bobot pada jaringan syaraf tiruan agar nilai *error* yang dihasilkan dapat berkurang.



## 4.3 Implementasi Interface

### 4.3.1 Form Training

*Form Training* yaitu *form* yang digunakan untuk memasukkan data asli parameter cuaca kedalam sistem. *Form training* memiliki 3 sub menu yaitu sub menu *browse* untuk memasukkan data asli cuaca. Sedangkan sub menu *tampilkan data* digunakan untuk menampilkan data parameter cuaca ke dalam tabel. Sub menu *training* digunakan untuk proses pelatihan data parameter cuaca yang telah di normalisasi sebelumnya dan dengan memasukkan parameter pelatihan berupa *learnig rate* dan jumlah *neuron hidden* yang di inginkan terlebih dahulu. Dan *form training* dapat dilihat pada gambar 4.1.

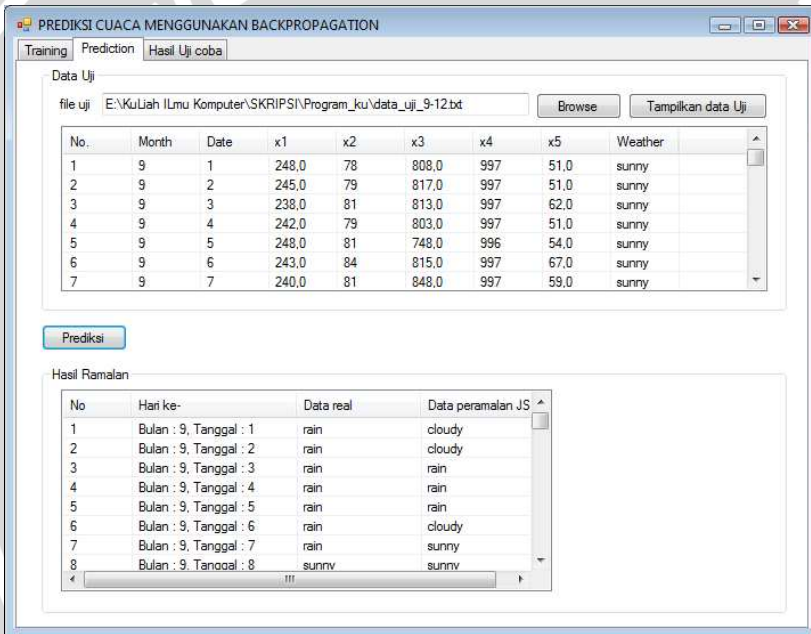
No.	Month	Date	x1	x2	x3	x4	x5	Weather
1	1	1	228,0	25	855,0	998	40,0	rain
2	1	2	238,0	13	763,0	997	20,0	rain
3	1	3	236,0	6	785,0	997	21,0	rain
4	1	4	210,0	25	828,0	999	24,0	rain
5	1	5	237,0	6	832,0	997	29,0	rain
6	1	6	233,0	88	821,0	998	23,0	rain
7	1	7	223,0	88	858,0	998	28,0	rain
8	1	8	221,0	21	840,0	998	18,0	rain
9	1	9	221,0	21	840,0	998	18,0	rain

Variabel Input  
Learning rate: 0,1    Jml Neuron Hidden: 6    1000    Epoch:   
  
Training  
Hasil Training  
-0,195269 | 0,181321 | 0,230139 | -0,411460 | 0,355669 | -0,265008 |  
-0,163824 | -0,415862 | 0,409561 | 0,151223 | -0,138621 | -0,283542 |  
0,042012 | 0,010503 | 0,157543 | -0,525144 | 0,273075 | 0,336092 |  
0,372279 | -0,218988 | -0,313882 | -0,343081 | -0,189789 | 0,226287 |  
-0,502853 | -0,359181 | -0,161631 | 0,170611 | -0,143672 | -0,179591 |  
-0,079014 | 0,105351 | -0,342392 | -0,401652 | -0,072429 | 0,434575 |  
  
0,307842 | 0,368072 | -0,220843 | 0,327918 | 0,254304 | -0,194074 | 0,033461 |  
-0,056950 | -0,162714 | 0,317292 | 0,227799 | 0,195256 | -0,301020 | -0,423056 |  
0,168668 | 0,238946 | 0,091362 | -0,442752 | -0,231918 | -0,358419 | -0,133528 |

Gambar 4.1 *Form Training*

### 4.3.1 Form Prediction

*Form Prediction* memiliki 3 sub menu yaitu sub menu *browse*, sub menu tampilkan data uji dan sub menu prediksi. Pada sub menu *browse*, digunakan untuk memasukkan data cuaca yang akan di lakukan prediksi. Pada sub menu tampilkan data digunakan untuk menapmpilkan data ke dalam tabel. Dan *form prediction* ini digunakan untuk proses prediksi cuaca. Menu *prediction* ini dapat dilihat pada gambar 4.2.



Gambar 4.2 Form Forecasting



## 4.4 Uji Coba Sistem

Dari sistem yang telah dibuat sebelumnya, maka pada sub bab ini akan dijelaskan pembahasan mengenai proses pengujian yang telah dilakukan pada sistem dan hasil proses pengujian tersebut. Pengujian dilakukan dengan cara melatih jaringan syaraf tiruan dengan parameter-parameter yang berbeda.

Data yang digunakan dalam uji coba sistem ini adalah data parameter-parameter cuaca harian kota malang selama satu tahun. Data tersebut meliputi 8 bulan atau sebanyak 244 data untuk pelatihan, dan 3 bulan atau sebanyak 122 data untuk data pengujian.

### 4.4.1 Uji Pengaruh Jumlah *Neuron Hidden* dan *Learning Rate* Terhadap MSE

Pengujian ini adalah menguji jumlah *neuron hidden* dan *learning rate* yang baik untuk sistem. Pengujian ini dilakukan dengan cara membandingkan nilai MSE yang diperoleh pada tiap kombinasi jumlah *neuron hidden* dan nilai *learning rate* yang terbaik nantinya akan menghasilkan MSE yang terkecil.

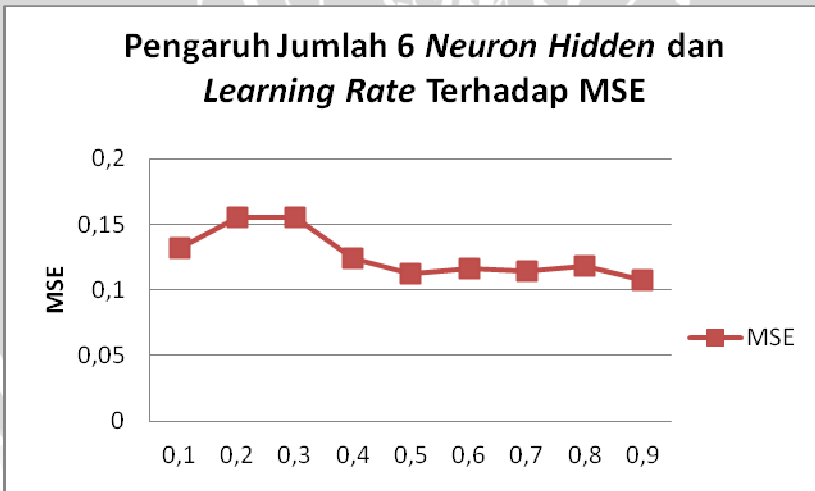
Pengujian dilakukan sebanyak 45 kali dari data *training* selama 8 bulan atau 244 data, dengan kombinasi dari rentang nilai *learning rate* yaitu 0,1 – 0,9 dan banyak *neuron hidden*. Penggunaan *neuron hidden* sendiri yang di gunakan yaitu antara 6 sampai 10 neuron. Sebab, Menurut Heaton (2010), untuk menentukan jumlah *neuron* dalam satu *hidden layer* harus memenuhi aturan bahwa jumlah *neuron* disarankan  $\frac{2}{3}$  dari jumlah *neuron input* ditambah jumlah *output*, dan jumlah *neuron* kurang dari dua kali jumlah *input*. Maka apabila jumlah *neuron* diluar rentang tersebut, akan sulit mendapatkan MSE yang kecil.

Kemudian pengujian dilakukan dengan 15000 kali iterasi dan target *error* sebesar 0,0001. Dalam pengujian ini digunakan iterasi 15000 karena *error* yang didapat sudah mendekati konvergen. Dari semua kombinasi *neuron hidden* dan *learning rate* pengujian tersebut dilakukan berkali-kali (lebih dari satu kali) dan di ambil MSE terbaik dari satu kombinasi. Dan hasil dari pengujian masing-masing dari jumlah *neuron hidden* dapat dilihat pada tabel 4.1 sampai tabel 4.5.

**Tabel 4.1 Pengaruh Jumlah 6 *Neuron Hidden* dan *Learning Rate* Terhadap MSE**

Jumlah <i>Neuron Hidden</i>	<i>Learning rate</i>	MSE
6	0,1	0,1314728
	0,2	0,1556272
	0,3	0,1556229
	0,4	0,1241942
	0,5	0,1124191
	0,6	0,1163344
	0,7	0,1145551
	0,8	0,1184644
	0,9	0,1071071

Pada tabel 4.1 dapat dilihat dari 9 kali percobaan dengan perubahan learning rate 0,1 – 0,9 di dapat MSE terkecil dari jumlah 6 neuron hidden adalah 0,1071071, yaitu terdapat pada *learning rate* 0,9. Dan grafik pengaruh jumlah 6 neuron hidden dan learning rate dapat dilihat pada gambar grafik 4.4.



**Gambar 4.4 Grafik Pengaruh Jumlah 6 *Neuron Hidden* dan *Learning Rate* Terhadap MSE**

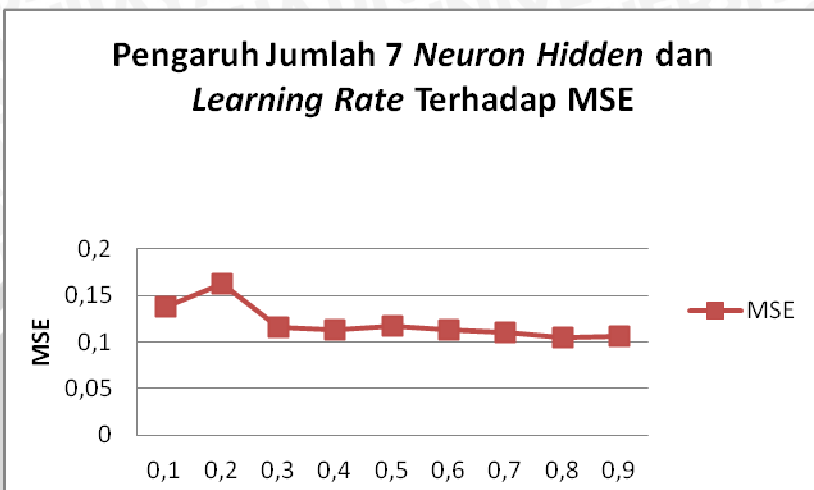


Grafik 4.4 menunjukkan MSE yang dihasilkan dari pengujian dengan menggunakan 6 neuron *hidden* mengalami penaikan dan penurunan MSE, dalam hal ini berarti MSE tidak terlalu stabil karena terdapat perubahan MSE yang signifikan. Di dapat MSE terkecil atau terendah terdapat pada learning rate 0,9.

**Tabel 4.2 Pengaruh Jumlah 7 Neuron Hidden dan Learning Rate Terhadap MSE**

Jumlah <i>Neuron Hidden</i>	<i>Learning rate</i>	MSE
7	0,1	0,1381401
	0,2	0,1622059
	0,3	0,1162362
	0,4	0,1126895
	0,5	0,1169523
	0,6	0,1128324
	0,7	0,1101227
	0,8	<b>0,1050134</b>
	0,9	0,1056618

Pada tabel 4.2 dapat dilihat dari pengujian ke-2, yaitu dengan jumlah *neuron hidden* 7 dengan 9 kali percobaan dengan perubahan learning rate 0,1 – 0,9. Di dapat MSE terkecil dari jumlah 7 neuron hidden ini adalah 0,1050134, yaitu terdapat pada *learning rate* 0,8. Grafik pengaruh jumlah 7 *neuron hidden* dan *learning rate* dapat dilihat pada gambar grafik 4.5.



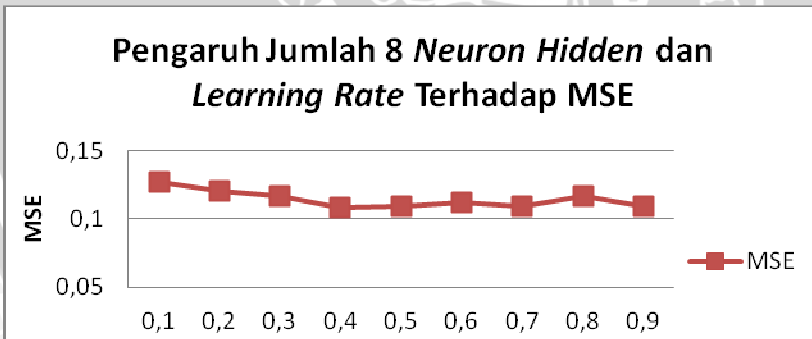
**Gambar 4.5 Grafik Pengaruh Jumlah 7 *Neuron Hidden* dan *Learning Rate* Terhadap MSE**

Dari grafik 4.5 menunjukkan MSE yang dihasilkan dari pengujian dengan menggunakan 7 *neuron hidden* mengalami kenaikan dan penurunan MSE di awal pengujian 3 learning pertama. Dalam hal ini berarti MSE tidak terlalu stabil karena terdapat perubahan MSE yang signifikan. Akan tetapi perubahan MSE selanjutnya relatif stabil dan tidak terjadi perubahan yang terlalu signifikan. Terlihat di dapat MSE terkecil atau terendah yaitu terdapat pada learning rate 0,8.

**Tabel 4.3 Pengaruh Jumlah 8 *Neuron Hidden* dan *Learning Rate* Terhadap MSE**

Jumlah <i>Neuron Hidden</i>	<i>Learning rate</i>	MSE
8	0,1	0,1265474
	0,2	0,1201096
	0,3	0,1165689
	0,4	0,1079739
	0,5	0,1087717
	0,6	0,1119673
	0,7	0,1087973
	0,8	0,1165997
	0,9	0,1088324

Pada tabel 4.3 dapat dilihat dari pengujian ke-3, yaitu dengan jumlah *neuron hidden* 8 dengan 9 kali percobaan dengan perubahan *learning rate* 0,1 – 0,9. Di dapat MSE terkecil dari jumlah 8 *neuron hidden* ini adalah 0,1079739, yaitu terdapat pada *learning rate* 0,4. Grafik pengaruh jumlah 8 *neuron hidden* dan *learning rate* dapat dilihat pada gambar grafik 4.6.



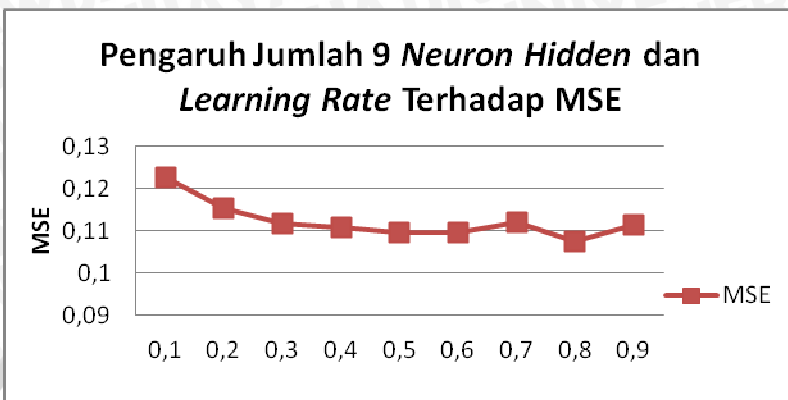
**Gambar 4.6 Grafik Pengaruh Jumlah 8 *Neuron Hidden* dan *Learning Rate* Terhadap MSE**

Dari grafik 4.6 menunjukkan MSE yang dihasilkan dari pengujian dengan menggunakan 8 *neuron hidden* mengalami kenaikan dan penurunan MSE. Akan tetapi dalam hal ini perubahan MSE masih relatif stabil karena perubahan MSE tidak terlalu signifikan. Dari grafik terlihat di dapat MSE terkecil atau terendah yaitu terdapat pada learning rate 0,8.

**Tabel 4.4 Pengaruh Jumlah 9 Neuron Hidden dan Learning Rate Terhadap MSE**

Jumlah <i>Hidden</i>	<i>Neuron</i>	<i>Learning rate</i>	MSE
9	9	0,1	0,1225359
		0,2	0,1152055
		0,3	0,1115941
		0,4	0,1108526
		0,5	0,1094377
		0,6	0,1095872
		0,7	0,1119656
		0,8	<b>0,1073144</b>
		0,9	0,1114698

Pada tabel 4.4 dapat dilihat dari pengujian ke-4, yaitu dengan kombinasi jumlah *neuron hidden* 9 dengan 9 kali percobaan dengan perubahan learning rate 0,1 – 0,9, Di dapat MSE terkecil dari jumlah 9 neuron hidden ini adalah 0,1073144, yaitu terdapat pada *learning rate* 0,8. Grafik pengaruh jumlah 9 *neuron hidden* dan *learning rate* dapat dilihat pada gambar grafik 4.6.



**Gambar 4.7** Grafik Pengaruh Jumlah 9 *Neuron Hidden* dan *Learning Rate* Terhadap MSE

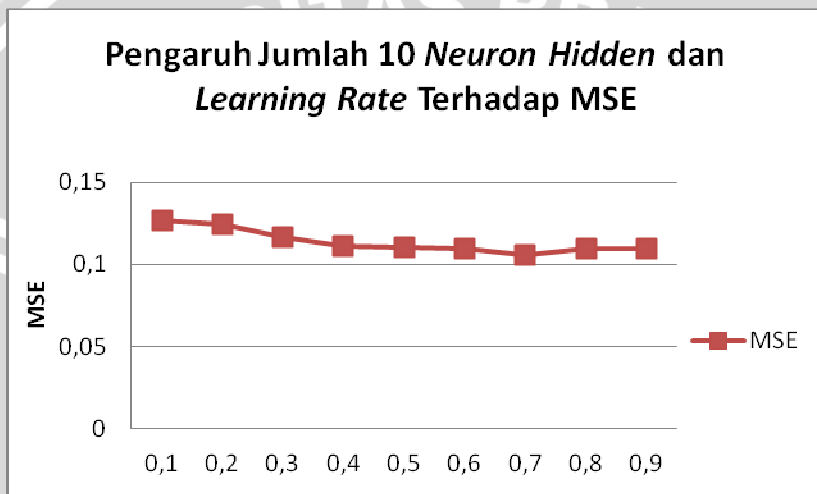
Dapat dilihat dari gambar grafik 4.7 menunjukkan MSE yang dihasilkan dari 9 kali pengujian *learning rate* dengan range 0,1 – 0,9 dan dengan menggunakan 9 *neuron hidden* mengalami kenaikan dan penurunan MSE. Tetapi MSE yang dihasilkan di antara *learning rate* 0,3 samapi *leaning rate* 0,7 masih dalam keadaan stabil. Kemudian terjadi penurunan MSE pada *learning rate* 8 yang juga menjadi MSE terkecil tau terendah pada pengujian di *neuron hidden* 9 ini.

**Tabel 4.5** Pengaruh Jumlah 10 *Neuron Hidden* dan *Learning Rate* Terhadap MSE

Jumlah <i>Neuron Hidden</i>	<i>Learning rate</i>	MSE
10	0,1	0,1268862
	0,2	0,1245618
	0,3	0,1166114
	0,4	0,1108458
	0,5	0,1099982
	0,6	0,1096203
	0,7	0,1054847
	0,8	0,1095126
	0,9	0,1094254



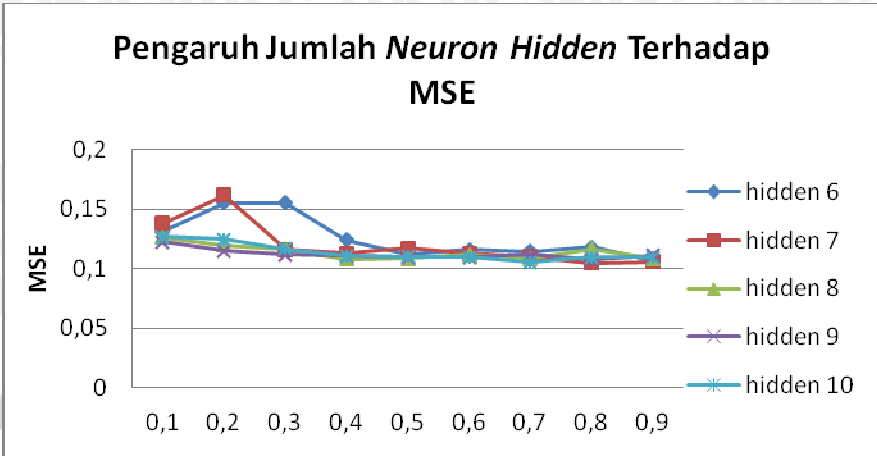
Pada tabel 4.5 dapat dilihat dari pengujian terakhir, yaitu dengan kombinasi jumlah *neuron hidden* 10 yang dilakukan sebanyak 9 kali percobaan dengan perubahan *learning rate* 0,1 – 0,9, di dapat MSE terkecil dari jumlah 10 *neuron hidden* ini adalah 0,1054847, yaitu terdapat pada *learning rate* 0,7. Grafik pengaruh jumlah 10 *neuron hidden* dan *learning rate* dapat dilihat pada gambar grafik 4.6.



**Gambar 4.8 Grafik Pengaruh Jumlah 10 *Neuron Hidden* dan *Learning Rate* Terhadap MSE**

Dapat dilihat dari gambar grafik 4.8 menunjukkan MSE yang dihasilkan dari 9 kali pengujian *learning rate* dengan range 0,1 – 0,9 dan dengan menggunakan 10 *neuron hidden* mengalami kenaikan dan penurunan MSE. Akan tetapi MSE yang dihasilkan masih dalam keadaan stabil karena tidak terlalu banyak perubahan MSE yang signifikan. Dan MSE terkecil atau terendah terdapat pada *learning rate* 0,7.

Pada tabel 4.1 sampai tabel 4.5 dapat dilihat bahwa nilai MSE yang paling kecil yang dihasilkan oleh semua pengujian ini adalah sebesar 0,1050134 yaitu terletak pada kombinasi 7 *neuron hidden* dan pada *learning rate* 0,8. Dari keseluruhan pengujian dari kombinasi-kombinasi tersebut dapat dilihat pada gambar grafik pengujian 4.9.



**Gambar 4.9 Pengaruh Jumlah Neuron Hidden Terhadap MSE**

Pada grafik gambar 4.9 menunjukkan gabungan keseluruhan dari 5 macam jumlah *hidden neuron*. Dapat dilihat dari grafik perubahan nilai MSE yang kecil tidak berpengaruh pada nilai besar atau kecilnya *learning rate*. Itu bukan berarti semakin kecil nilai *learning rate* semakin kecil pula MSE yang dihasilkan. Di dapat jumlah *hidden neuron* 8, 9, dan 10 yang memiliki kestabilan nilai MSE, karena tidak memiliki begitu banyak perubahan nilai MSE yang sangat signifikan. Sedangkan nilai MSE yang terkecil dari keseluruhan MSE yang ada terdapat pada jumlah *hidden neuron* ke 7, yaitu pada *learning rate* 0,8.

#### 4.4.2 Uji Prediksi

Dalam pengujian ini menggunakan data uji sebanyak 3 bulan atau sebanyak 122 data dan yang digunakan adalah kombinasi 5 *neuron input* dan 3 *neuron output* dan menggunakan bobot terbaik yaitu terdapat pada jumlah *neuron hidden* 7. Sebagian hasil dari prediksi dapat dilihat pada tabel 4.2, untuk hasil lengkap terdapat pada lampiran 1.

**Tabel 4.2 Tabel Hasil Prediksi Cuaca**

No	Data Real	Data Prediksi JST
1	hujan	Mendung
2	hujan	Mendung
3	hujan	Mendung
4	hujan	Mendung
5	hujan	Hujan
6	hujan	Mendung
7	hujan	Mendung
8	cerah	Cerah
9	cerah	Cerah
10	hujan	Hujan
11	hujan	Hujan
12	hujan	Hujan
13	hujan	Hujan
14	hujan	hujan
15	hujan	hujan
16	hujan	hujan
17	hujan	hujan
18	hujan	hujan
19	hujan	hujan
20	hujan	hujan
21	hujan	hujan
22	hujan	hujan
23	hujan	hujan
24	hujan	hujan
25	hujan	hujan
26	hujan	hujan
27	hujan	hujan
28	hujan	hujan

29	hujan	Hujan
30	hujan	Hujan
31	hujan	hujan
32	mendung	hujan
33	hujan	hujan
34	mendung	hujan
35	hujan	hujan
36	hujan	hujan
37	hujan	hujan
38	hujan	hujan
39	mendung	cerah
40	cerah	cerah
41	mendung	cerah
42	mendung	hujan
43	mendung	hujan
44	hujan	hujan
45	hujan	hujan
46	hujan	hujan
47	cerah	hujan
48	cerah	hujan
49	cerah	hujan
50	hujan	hujan
51	hujan	hujan
52	cerah	hujan
53	hujan	hujan
54	hujan	hujan
55	mendung	hujan
56	cerah	hujan
57	cerah	cerah
58	cerah	cerah
59	mendung	hujan
60	hujan	hujan

61	hujan	hujan
62	mendung	mendung
63	hujan	hujan
64	Cerah	Cerah
65	Cerah	Cerah

Dari pengujian sistem diatas didapatkan data hasil pengujian yang disajikan seperti pada tabel 4.3.

**Tabel 4.3 Tabel Akurasi Prediksi Cuaca**

Jumlah Data	Hasil	Tingkat Akurasi
Benar	101	$\frac{101}{122} \times 100\% = 82,78689$
Salah	21	
<b>Total</b>	<b>122</b>	

Pada pengujian menggunakan bobot terbaik tersebut dengan menggunakan data selain data pelatihan, di dapat hasil seperti pada tabel 4.2, dimana jumlah data hasil prediksi sistem yang sama dengan data *real* sebanyak 101 data dari jumlah total 122 data yang diujikan. Maka dari hasil tersebut bisa didapat tingkat akurasi pada sistem jaringan syaraf tiruan ini adalah sebesar 82,78689 %. Sedangkan sebanyak 21 data yang tidak dikenali oleh sistem kemungkinan disebabkan oleh pola data yang tidak dikenali selama pelatihan. Hasil prediksi dipengaruhi oleh berbagai pola data uji. Ketika pola telah dikenali oleh sistem, maka akurasi prediksi juga semakin tinggi.

UNIVERSITAS BRAWIJAYA





## BAB V

### PENUTUP

#### 5.1 Kesimpulan

- Implementasi Jaringan syaraf tiruan *Backpropagation* pada prediksi cuaca digunakan untuk melatih dan mengenali pola dari data cuaca sehingga mampu digunakan untuk memprediksi cuaca.
- Bobot optimal yang dihasilkan selama 45 kali pelatihan yaitu dari kombinasi *neuron hidden* 7 dan *learning rate* 0,8 dan menghasilkan MSE sebesar 0,1050134.
- Hasil dari bobot optimal digunakan untuk proses prediksi cuaca dan mampu menghasilkan tingkat akurasi sebesar 82,78689 %.
- Hasil prediksi dipengaruhi oleh berbagai pola data uji. Ketika pola telah dikenali oleh sistem, maka akurasi prediksi juga semakin tinggi.

#### 5.2 Saran

Dalam penelitian Jaringan Syaraf Tiruan ini perlu dilakukan perbandingan dengan algoritma lainnya agar diketahui metode mana yang menghasilkan prediksi lebih baik.

Pada pengujian lebih lanjut, dapat ditambahkan parameter lain yang dapat mempengaruhi cuaca agar dapat dihasilkan prediksi yang memiliki akurasi lebih baik.

UNIVERSITAS BRAWIJAYA



## DAFTAR PUSTAKA

- Boucher, K. 1975. *Global Climate*. The University Press. Ltd. London.
- Bose, N. K., dan P. Liang. 1996. *Neural Network Fundamentals with Graphs, Algorithms and Applications*. New York: McGraw-Hill.
- Bharath, Ramachandran; Drosen, James.; *Neural Network Computing*, Windcrest, 1994
- Prita, M. 2009. *Prediksi Cuaca Menggunakan Logika Fuzzy Untuk Pelayaran Pelabuhan Tanjung Perak Surabaya*
- Holton, J. R. 1979. *An Introduction to Dynamic Meteorology*. 2nd ed. Academic Press. Inc. London. 391 p.
- Heaton, J. 2010. *Introduction to Neural Network for C#, Second Edition*. St. Louis: Neatton Research, Inc.
- Hidayat, Syarif. 2008. *Unsur-unsur iklim (pdf)*.  
[http://pksm.mercubuana.ac.id/new/elearning/files\\_modul/12036-3-947795834327/](http://pksm.mercubuana.ac.id/new/elearning/files_modul/12036-3-947795834327/). Tanggal akses 10-11-2010
- Maqsood , Imran .2004. *Weather Forecasting Models Using Ensembles of Neural Networks*. Journal Faculty of Engineering, University of Regina, Regina, SK S4S 0A2, Canada.
- Jumingan. 2009. *Studi Kelayakan Bisnis – Teori dan Pembuatan Proposal Kelayakan*. Bumi Aksara. Jakarta
- Kusumadewi, S. (2003). *Artificial Intelligence ( Teknik dan Aplikasinya)*. Yogyakarta: Graha Ilmu.

- Mohsen, Hayati, Mohebi Zahra.2007. *Application of Artificial Neural Networks for Temperature Forecasting*. Journal World Academy of Science, Engineering and Technology.
- Sutikno, Boer R. 2004. Model Prediksi Curah Hujan dengan Regresi Splines Adaptif Berganda . Jurnal Agromet Indonesia.
- Syarif Hidayat, *Pengembangan Sumber Daya Air*, Pusat Pengembangan Bahan Ajar – UMB, Jakarta, 2008.
- Siang, Jong Jek. 2005. *Jaringan Syaraf Tiruan dan Pemogramannya Menggunakan Matlab*. Yogyakarta: Andi Offset.
- Setiawan, Wahyudi. 2008. *Prediksi Harga Saham Dengan Menggunakan Jaringan Syaraf Tiruan Multilayer Feedforward Network Dengan Algoritma Backpropagation*. Jurnal Manajemen Informatika, Fakultas Teknik, Universitas Trunojoyo Madura.
- Subiyanto, 2000. *Aplikasi Jaringan Syaraf Tiruan sebagai Metode Alternatif Prakiraan Beban Jangka Pendek*. <http://www.elektroindonesia.com/elektro/ener29.html-42k-> . (Tanggal Akses 13-11-2010).
- Chrestanti, Ruth. 2002. *Implementasi Backpropagation dalam Memprediksi Kebangkrutan Bank di Indonesia*. Jurnal Teknik Informatika, Fakultas Teknologi Industri, Universitas Atma Jaya Yogyakarta.
- Wallace. J.M. and Hobbs, P.V. 1977. *Atmospheric Science: An Introductory Survey*. Academic Press. New York. 467p.

## LAMPIRAN

### Lampiran 1 : Hasil Prediksi Cuaca

<b>NO</b>	<b>data real</b>	<b>data prediksi JST</b>
1	hujan	mendung
2	hujan	mendung
3	hujan	mendung
4	hujan	mendung
5	hujan	hujan
6	hujan	mendung
7	hujan	mendung
8	cerah	cerah
9	cerah	cerah
10	hujan	hujan
11	hujan	hujan
12	hujan	hujan
13	hujan	hujan
14	hujan	hujan
15	hujan	hujan
16	hujan	hujan
17	hujan	hujan
18	hujan	hujan
19	hujan	hujan
20	hujan	hujan
21	hujan	hujan
22	hujan	hujan
23	hujan	hujan
24	hujan	hujan
25	hujan	hujan
26	hujan	hujan
27	hujan	hujan

28	hujan	hujan
29	hujan	hujan
30	hujan	hujan
31	hujan	hujan
32	mendung	hujan
33	hujan	hujan
34	mendung	hujan
35	hujan	hujan
36	hujan	hujan
37	hujan	hujan
38	hujan	hujan
39	mendung	cerah
40	cerah	cerah
41	mendung	cerah
42	mendung	hujan
43	mendung	hujan
44	hujan	hujan
45	hujan	hujan
46	hujan	hujan
47	cerah	hujan
48	cerah	hujan
49	cerah	hujan
50	hujan	hujan
51	hujan	hujan
52	cerah	hujan
53	hujan	hujan
54	hujan	hujan
55	mendung	hujan
56	cerah	hujan
57	cerah	cerah
58	cerah	cerah
59	mendung	hujan



60	hujan	hujan
61	hujan	hujan
62	mendung	mendung
63	hujan	hujan
64	cerah	cerah
65	cerah	cerah
66	cerah	cerah
67	cerah	cerah
68	cerah	cerah
69	hujan	hujan
70	hujan	hujan
71	hujan	mendung
72	hujan	hujan
73	hujan	hujan
74	cerah	cerah
75	cerah	cerah
76	mendung	hujan
77	cerah	cerah
78	cerah	cerah
79	cerah	cerah
80	cerah	cerah
81	cerah	cerah
82	cerah	cerah
83	cerah	cerah
84	cerah	cerah
85	cerah	cerah
86	cerah	cerah
87	cerah	cerah
88	cerah	cerah
89	cerah	cerah
90	hujan	hujan
91	hujan	hujan

92	hujan	hujan
93	hujan	hujan
94	hujan	hujan
95	hujan	hujan
96	hujan	hujan
97	hujan	hujan
98	cerah	cerah
99	cerah	cerah
100	cerah	cerah
101	cerah	cerah
102	cerah	cerah
103	cerah	cerah
104	cerah	cerah
105	cerah	cerah
106	cerah	cerah
107	cerah	cerah
108	cerah	cerah
109	cerah	cerah
110	cerah	cerah
111	cerah	cerah
112	cerah	cerah
113	cerah	cerah
114	cerah	cerah
115	cerah	cerah
116	cerah	cerah
117	cerah	cerah
118	cerah	cerah
119	cerah	cerah
120	cerah	cerah
121	cerah	cerah
122	cerah	cerah

**Lampiran 2 : Data Parameter-parameter Cuaca Untuk Pelatihan**

<b>Bln</b>	<b>tgl</b>	<b>X1</b>	<b>X2</b>	<b>X3</b>	<b>X4</b>	<b>X5</b>	<b>kondisi</b>
1	1	22.8	25	85.5	998	4.0	hujan
1	2	23.8	13	76.3	997	2.0	hujan
1	3	23.6	6	78.5	997	2.1	hujan
1	4	21.0	25	82.8	999	2.4	hujan
1	5	23.7	6	83.2	997	2.9	hujan
1	6	23.3	88	82.1	998	2.3	hujan
1	7	22.3	88	85.8	998	2.8	hujan
1	8	23.1	31	84.0	998	1.8	hujan
1	9	23.3	44	83.3	997	2.9	mendung
1	10	23.8	63	78.3	996	3.1	cerah
1	11	23.9	80	78.0	997	3.2	cerah
1	12	24.6	90	78.1	997	3.8	cerah
1	13	25.1	93	76.9	997	4.2	hujan
1	14	23.3	98	87.9	998	1.8	hujan
1	15	23.7	14	83.2	997	2.7	cerah
1	16	22.3	80	87.6	997	3.4	hujan
1	17	23.5	50	82.2	997	3.2	mendung
1	18	23.6	66	83.7	997	2.3	cerah
1	19	24.0	75	80.7	997	2.3	cerah
1	20	24.3	94	81.2	997	2.6	cerah
1	21	24.5	95	79.3	997	1.9	cerah
1	22	24.3	95	79.3	996	2.9	cerah
1	23	24.3	90	78.2	996	3.0	cerah
1	24	24.7	94	76.3	996	2.8	cerah
1	25	24.1	96	81.5	996	1.8	cerah

1	26	22.8	79	85.7	997	1.6	hujan
1	27	23.4	50	82.5	997	1.5	cerah
1	28	23.6	75	79.8	997	1.6	hujan
1	29	23.4	81	81.1	998	1.8	hujan
1	30	23.3	63	82.2	998	1.8	mendung
1	31	23.4	63	76.3	997	1.7	mendung
2	1	24.2	66	82.8	997	1.7	cerah
2	2	23.3	79	88.9	997	1.9	hujan
2	3	23.4	50	85.0	997	2.0	hujan
2	4	22.7	50	85.7	998	1.8	hujan
2	5	23.3	50	86.3	998	3.7	hujan
2	6	23.7	75	84.0	998	3.2	hujan
2	7	23.0	46	85.9	998	2.7	hujan
2	8	23.0	50	83.5	998	2.5	cerah
2	9	23.0	68	85.0	997	3.2	cerah
2	10	23.3	80	85.0	997	2.9	hujan
2	11	23.5	66	78.0	998	2.5	hujan
2	12	23.2	48	84.0	998	2.9	hujan
2	13	23.5	50	80.2	998	3.6	hujan
2	14	24.0	63	75.4	998	4.0	hujan
2	15	23.2	38	81.7	998	3.1	hujan
2	16	23.5	41	79.7	998	2.1	hujan
2	17	23.7	56	83.9	998	2.4	hujan
2	18	23.7	70	82.7	998	4.3	hujan
2	19	24.2	56	79.8	998	5.9	hujan
2	20	24.3	63	76.3	998	4.4	hujan
2	21	23.8	65	82.5	998	3.1	hujan
2	22	23.5	38	85.3	998	2.9	hujan

2	23	23.9	38	78.1	998	2.0	hujan
2	24	24.1	75	81.6	998	2.2	hujan
2	25	23.3	81	85.1	998	1.5	hujan
2	26	23.2	25	87.4	998	1.3	hujan
2	27	23.2	25	86.2	998	1.4	hujan
2	28	23.3	38	85.5	998	1.6	hujan
2	29	23.1	19	89.7	998	1.5	hujan
3	1	23.9	19	83.3	998	1.6	hujan
3	2	23.8	44	88.2	998	1.8	hujan
3	3	23.4	50	85.0	998	2.5	hujan
3	4	23.4	38	85.4	997	2.5	hujan
3	5	22.9	38	93.0	998	1.8	hujan
3	6	22.5	44	88.2	998	2.0	hujan
3	7	22.5	44	86.3	998	2.2	hujan
3	8	23.9	50	82.9	998	1.3	hujan
3	9	22.8	81	86.6	998	1.2	hujan
3	10	23.6	43	84.3	997	1.8	hujan
3	11	24.3	76	78.0	997	1.6	cerah
3	12	22.7	75	80.9	997	1.3	hujan
3	13	20.2	50	90.5	998	1.6	hujan
3	14	23.6	38	80.6	998	2.6	hujan
3	15	21.9	63	89.0	998	1.7	hujan
3	16	23.0	44	83.1	998	1.8	hujan
3	17	22.3	50	87.9	998	1.4	hujan
3	18	22.5	63	85.1	998	1.3	hujan
3	19	22.9	68	84.4	998	1.3	hujan
3	20	21.8	19	88.8	998	1.5	hujan
3	21	21.8	25	88.9	998	1.2	hujan

3	22	23.4	31	80.6	998	0.9	hujan
3	23	22.6	25	90.3	998	0.8	hujan
3	24	23.4	25	83.6	998	2.2	hujan
3	25	23.7	25	80.8	997	2.1	hujan
3	26	22.6	38	90.5	998	2.7	hujan
3	27	21.9	25	85.3	998	1.9	hujan
3	28	22.3	55	83.7	998	2.0	hujan
3	29	23.5	31	87.4	998	2.2	hujan
3	30	23.1	45	88.0	998	2.2	hujan
3	31	22.5	44	91.8	998	1.8	hujan
4	1	21.5	19	89.3	998	1.8	mendung
4	2	23.4	65	77.2	997	2.0	hujan
4	3	24.1	75	81.0	998	2.2	hujan
4	4	23.5	64	86.8	998	2.0	hujan
4	5	24.1	63	81.7	998	1.9	hujan
4	6	23.7	78	86.9	998	2.2	hujan
4	7	23.5	75	85.0	998	2.2	hujan
4	8	24.2	50	81.5	997	1.6	mendung
4	9	23.9	63	85.8	997	1.9	hujan
4	10	23.6	53	81.9	998	2.8	hujan
4	11	23.8	83	78.0	998	2.9	hujan
4	12	23.5	90	83.6	998	3.0	cerah
4	13	23.6	76	83.3	997	2.7	hujan
4	14	24.3	54	78.5	997	2.2	cerah
4	15	22.3	81	79.1	997	2.5	hujan
4	16	21.4	88	84.7	998	2.8	mendung
4	17	24.3	63	79.6	997	3.2	cerah
4	18	23.4	88	82.2	997	2.2	mendung



4	19	23.3	63	83.2	997	2.5	cerah
4	20	23.8	75	83.4	997	2.1	mendung
4	21	24.3	63	86.1	996	2.6	cerah
4	22	24.7	84	80.9	996	3.4	cerah
4	23	23.8	89	83.8	997	5.1	cerah
4	24	23.8	85	81.0	997	2.8	mendung
4	25	24.0	69	80.8	996	2.7	cerah
4	26	23.8	75	82.3	997	3.2	cerah
4	27	23.8	75	79.2	997	3.9	cerah
4	28	24.3	75	79.1	996	3.3	cerah
4	29	24.3	73	81.7	997	2.3	hujan
4	30	23.2	69	86.5	997	2.6	mendung
5	1	23.6	66	85.1	997	2.2	mendung
5	2	23.8	65	85.3	997	2.0	hujan
5	3	23.3	60	87.4	998	2.0	hujan
5	4	23.3	53	88.0	998	1.8	hujan
5	5	23.8	58	83.8	998	2.0	mendung
5	6	23.4	68	82.4	997	3.2	mendung
5	7	23.8	69	79.1	997	6.1	mendung
5	8	22.3	65	78.1	997	6.9	hujan
5	9	24.4	73	74.8	998	6.5	hujan
5	10	24.0	89	73.1	998	7.1	hujan
5	11	22.8	93	78.0	998	5.8	cerah
5	12	23.4	94	79.8	997	3.1	cerah
5	13	24.2	75	75.6	997	3.2	cerah
5	14	23.5	76	76.9	997	3.0	cerah
5	15	23.7	75	79.6	997	2.4	cerah
5	16	23.8	75	80.1	997	3.2	mendung

5	17	24.0	70	79.6	997	2.7	mendung
5	18	24.4	70	79.9	996	4.0	mendung
5	19	24.9	69	81.8	996	4.1	mendung
5	20	23.9	68	82.6	996	3.6	mendung
5	21	24.6	68	77.6	996	3.8	mendung
5	22	24.3	69	79.0	996	4.2	hujan
5	23	23.5	56	80.2	996	5.0	mendung
5	24	23.7	63	81.5	996	5.0	mendung
5	25	24.0	66	79.5	996	6.8	mendung
5	26	24.3	68	84.2	996	6.0	mendung
5	27	24.4	68	83.2	996	4.2	mendung
5	28	23.8	65	83.2	997	4.8	mendung
5	29	23.3	63	84.3	997	4.0	mendung
5	30	23.1	64	85.0	997	3.5	mendung
5	31	23.5	65	82.8	997	4.1	mendung
6	1	23.6	63	83.6	997	4.3	mendung
6	2	23.8	65	79.6	997	4.7	mendung
6	3	23.3	68	78.0	997	4.0	mendung
6	4	23.3	66	80.0	997	4.7	mendung
6	5	23.8	66	79.0	997	4.4	mendung
6	6	23.4	68	83.0	997	2.7	mendung
6	7	23.8	58	76.8	997	2.9	mendung
6	8	22.3	58	76.8	996	2.7	mendung
6	9	24.4	65	82.4	998	2.0	mendung
6	10	24.0	65	83.7	998	2.3	mendung
6	11	22.8	46	78.0	997	2.5	mendung
6	12	23.4	60	81.3	998	2.6	mendung
6	13	24.2	45	77.3	997	3.7	mendung

6	14	23.5	69	81.8	997	3.8	mendung
6	15	23.7	70	82.4	997	4.4	mendung
6	16	23.8	69	80.4	997	2.8	mendung
6	17	24.0	68	82.5	997	2.8	mendung
6	18	24.4	69	79.3	997	3.1	mendung
6	19	24.9	68	82.1	997	2.1	mendung
6	20	23.9	68	75.4	997	2.3	mendung
6	21	24.6	63	79.3	997	3.5	hujan
6	22	24.3	66	79.1	998	3.1	mendung
6	23	23.5	69	77.2	997	3.7	mendung
6	24	23.7	65	75.8	997	3.2	mendung
6	25	24.0	66	77.7	997	2.1	mendung
6	26	24.3	63	78.7	997	1.4	mendung
6	27	24.4	64	81.6	997	1.4	mendung
6	28	23.8	65	78.0	997	1.1	hujan
6	29	23.3	66	80.7	998	3.1	mendung
6	30	23.1	65	81.7	997	3.0	mendung
7	1	22.0	68	80.8	997	4.2	mendung
7	2	23.3	69	80.3	997	3.8	mendung
7	3	23.6	69	83.3	997	4.9	hujan
7	4	23.8	50	77.2	997	6.2	mendung
7	5	23.2	69	79.5	997	5.3	hujan
7	6	22.0	65	81.1	999	5.9	hujan
7	7	22.4	55	84.5	998	3.8	hujan
7	8	23.6	53	78.8	997	4.2	mendung
7	9	22.9	65	83.5	997	5.1	hujan
7	10	23.0	63	80.8	998	3.9	hujan
7	11	23.3	60	78.0	998	4.6	hujan

7	12	23.5	63	80.9	998	6.4	hujan
7	13	23.6	66	80.9	998	4.5	hujan
7	14	23.1	68	82.3	998	5.9	hujan
7	15	21.9	65	82.9	998	6.6	hujan
7	16	21.1	68	81.7	998	2.7	hujan
7	17	22.8	63	80.5	998	4.1	mendung
7	18	23.6	66	81.8	997	6.4	mendung
7	19	22.7	66	80.3	997	6.5	mendung
7	20	22.6	65	78.9	997	5.4	hujan
7	21	23.3	66	78.8	998	7.2	hujan
7	22	23.0	65	80.5	998	5.8	hujan
7	23	22.0	68	82.0	998	4.5	hujan
7	24	22.0	69	82.0	998	3.0	hujan
7	25	21.6	65	83.7	998	3.6	hujan
7	26	21.4	68	81.5	998	3.8	hujan
7	27	21.1	69	82.9	998	3.3	hujan
7	28	21.0	69	80.5	998	2.9	mendung
7	29	22.6	66	81.1	997	2.8	mendung
7	30	23.6	68	79.6	997	3.1	mendung
7	31	24.3	69	73.5	997	3.8	mendung
8	1	23.4	70	80.7	997	3.5	mendung
8	2	22.8	69	81.3	997	2.7	hujan
8	3	22.3	68	80.3	998	3.2	hujan
8	4	22.3	69	80.7	998	4.9	hujan
8	5	23.0	70	83.8	998	4.3	hujan
8	6	22.8	71	81.3	998	4.1	hujan
8	7	23.2	65	80.7	998	3.9	mendung
8	8	23.2	69	78.7	997	4.3	cerah

8	9	24.2	73	78.3	997	3.3	mendung
8	10	23.9	66	79.0	997	2.5	mendung
8	11	23.9	64	82.8	997	2.7	hujan
8	12	24.8	38	77.2	996	3.0	mendung
8	13	24.3	70	79.5	997	4.1	hujan
8	14	23.8	50	84.3	997	3.9	hujan
8	15	24.1	51	79.6	997	4.9	mendung
8	16	24.3	69	78.4	996	6.0	cerah
8	17	24.2	73	78.2	997	5.5	mendung
8	18	24.3	70	80.4	997	6.1	cerah
8	19	23.3	71	81.5	997	4.5	hujan
8	20	23.0	43	84.3	997	5.3	hujan
8	21	23.7	40	81.1	997	5.1	mendung
8	22	23.9	63	82.9	997	3.7	mendung
8	23	24.3	69	82.2	997	4.8	mendung
8	24	24.4	63	79.2	997	4.7	mendung
8	25	23.6	69	81.2	997	2.6	hujan
8	26	23.6	54	80.9	998	3.5	mendung
8	27	23.8	63	79.3	997	3.2	mendung
8	28	24.3	65	81.0	997	2.3	cerah
8	29	24.3	75	78.7	997	3.2	cerah
8	30	24.6	75	78.5	997	4.4	cerah
8	31	24.9	75	82.2	997	4.4	cerah

**Keterangan :**

X1 = Temperatur

X2 = Lama Penyinaran

X3 = Kelembaban Udara

X4 = Tekanan Udara

X5 = Kecepatan Angin

UNIVERSITAS BRAWIJAYA





**Lampiran 3 : Data Parameter-parameter Cuaca Untuk Pengujian**

9	1	24.8	78	80.8	997	5.1	cerah
9	2	24.5	79	81.7	997	5.1	cerah
9	3	23.8	81	81.3	997	6.2	cerah
9	4	24.2	79	80.3	997	5.1	cerah
9	5	24.8	81	74.8	996	5.4	cerah
9	6	24.3	84	81.5	997	6.7	cerah
9	7	24.0	81	84.8	997	5.9	hujan
9	8	23.3	78	83.0	998	2.9	hujan
9	9	23.8	53	81.1	997	4.0	cerah
9	10	24.2	75	79.0	997	3.3	cerah
9	11	25.0	76	78.1	996	3.7	cerah
9	12	25.1	79	79.2	997	4.2	cerah
9	13	25.2	83	77.8	997	3.0	cerah
9	14	25.3	85	74.7	997	2.7	cerah
9	15	23.6	88	76.4	997	4.2	cerah
9	16	24.8	84	75.9	997	3.7	cerah
9	17	24.7	95	77.1	997	5.1	cerah
9	18	25.2	91	76.0	996	5.7	cerah
9	19	24.2	98	77.6	996	5.5	cerah
9	20	24.8	95	76.0	996	5.7	cerah
9	21	23.0	100	78.0	997	6.0	cerah
9	22	22.8	94	79.5	996	4.8	cerah
9	23	24.2	95	79.9	997	4.9	cerah
9	24	24.0	94	78.8	997	3.6	cerah
9	25	24.6	93	78.3	996	4.3	cerah

9	26	25.4	95	74.6	996	3.9	cerah
9	27	24.8	93	76.8	996	5.2	cerah
9	28	24.3	91	74.3	996	4.8	cerah
9	29	25.4	94	78.9	996	4,8	cerah
9	30	24.8	96	77,7	997	5,1	cerah
10	1	24,5	80	78,3	996	4,2	mendung
10	2	23,6	70	80	997	3,8	cerah
10	3	22,4	79	78,3	996	3,1	mendung
10	4	25,8	81	76,6	997	3,3	cerah
10	5	26,4	83	78,8	996	3,5	cerah
10	6	25,3	85	79,2	996	4,2	cerah
10	7	24,7	83	83,2	996	4,3	cerah
10	8	23,7	81	87,5	997	2,3	mendung
10	9	23,8	64	85,1	997	2,5	hujan
10	10	23,3	63	82,6	998	3	mendung
10	11	23,3	63	81,5	997	3,4	mendung
10	12	24,3	79	82,7	997	3,5	mendung
10	13	24,7	79	80,3	997	3,5	cerah
10	14	25,7	85	77,5	996	3,2	cerah
10	15	26,1	88	78,6	996	4,1	cerah
10	16	26,1	93	77,2	996	4,3	hujan
10	17	26,2	91	80,9	996	2,6	hujan
10	18	26,2	89	82,4	996	2,8	hujan
10	19	26,4	90	82,3	996	2,3	cerah
10	20	26,6	88	79,7	996	3,6	cerah
10	21	26	90	79,8	996	3,8	hujan
10	22	25,5	91	83,3	996	1,8	cerah
10	23	26,3	88	77,4	996	2,3	cerah

10	24	25,8	75	76,1	996	4,6	mendung
10	25	25,4	63	78,3	996	4,5	hujan
10	26	24,7	70	84	997	1,9	hujan
10	27	22,9	20	88,8	998	1,1	hujan
10	28	24,3	13	75,4	997	2	mendung
10	29	25,3	78	71,8	997	3,6	cerah
10	30	25,5	96	73,7	996	3	cerah
10	31	25,3	99	77,9	996	1,9	mendung
11	1	25,3	63	76,8	996	1,7	cerah
11	2	24,9	78	83,8	997	1,3	hujan
11	3	23,7	63	96,3	998	1,2	hujan
11	4	24,6	21	84,4	997	1,3	hujan
11	5	25,4	31	81,9	997	1,6	hujan
11	6	24,6	50	85,0	997	2,1	hujan
11	7	25,4	38	75,7	997	1,7	cerah
11	8	25,6	76	75,5	996	2,3	cerah
11	9	25,3	78	79,9	996	2,7	cerah
11	10	24,8	76	84,0	997	2,2	cerah
11	11	24,7	91	80,6	997	2,3	cerah
11	12	24,3	94	81,2	997	1,9	hujan
11	13	25,6	50	82,5	997	1,6	hujan
11	14	24,9	75	85,5	998	1,6	mendung
11	15	25,9	69	79,4	997	1,7	hujan
11	16	25,8	84	81,8	998	1,7	hujan
11	17	25,0	75	84,3	998	2,9	hujan
11	18	25,0	20	83,2	997	4,1	hujan
11	19	24,3	65	79,6	999	2,3	hujan
11	20	23,8	30	89,3	998	0,9	hujan

11	21	23.8	33	86.3	998	1.4	hujan
11	22	23.6	35	87.4	998	1.2	hujan
11	23	23.9	31	87.0	998	1.7	hujan
11	24	23.5	63	89.2	999	1.7	hujan
11	25	23.9	4	90.3	999	1.7	hujan
11	26	24.0	30	84.2	998	4.3	hujan
11	27	23.5	69	88.3	998	3.0	hujan
11	28	23.4	34	81.6	997	3.7	cerah
11	29	23.8	90	82.1	997	3.5	cerah
11	30	23.9	98	81.1	997	3.1	cerah
12	1	24.8	95	81.4	997	3.1	cerah
12	2	24.9	94	79.9	996	2.6	cerah
12	3	24.5	91	79.8	997	2.7	cerah
12	4	24.7	95	79.8	997	3.3	cerah
12	5	25.2	90	80.3	997	2.7	cerah
12	6	25.3	94	82.9	997	2.8	hujan
12	7	23.8	83	89.8	998	2.6	hujan
12	8	23.8	58	89.8	998	2.6	hujan
12	9	23.3	41	90.2	998	1.6	hujan
12	10	23.1	5	91.5	998	1.0	hujan
12	11	23.3	10	92.0	998	1.1	hujan
12	12	23.7	23	88.5	998	1.2	hujan
12	13	23.3	46	90.3	999	0.8	hujan
12	14	24.2	33	87.4	998	1.1	hujan
12	15	24.6	51	80.7	998	2.0	hujan
12	16	25.0	58	79.6	997	2.0	hujan
12	17	24.0	78	84.1	998	2.8	hujan
12	18	24.2	84	84.2	998	2.4	hujan

12	19	23.7	65	84.4	998	2.9	hujan
12	20	24.6	58	85.0	998	3.1	hujan
12	21	24.9	65	86.1	998	3.2	hujan
12	22	23.8	28	89.8	998	2.6	hujan
12	23	24.3	26	86.9	998	2.6	hujan
12	24	23.7	29	88.2	998	1.6	hujan
12	25	24.1	63	82.4	998	2.0	hujan
12	26	24.7	79	82.8	998	2.1	hujan
12	27	24.7	81	85.8	998	2.8	hujan
12	28	24.3	64	88.8	998	3.1	hujan
12	29	24.4	54	86.6	998	2.0	hujan
12	30	22.5	51	95.5	998	2.0	hujan
12	31	23.7	34	89.1	998	1.8	hujan

Keterangan :

X1 = Temperatur

X2 = Lama Penyinaran

X3 = Kelembaban Udara

X4 = Tekanan Udara

X5 = Kecepatan Angin