

**PREDIKSI CURAH HUJAN MENGGUNAKAN METODE
ADAPTIVE-EXPECTATION BASED MULTI-ATTRIBUTE FUZZY
TIME SERIES**

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:
Yulia Trianandi
NIM: 115090601111019



PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2018

PENGESAHAN

**PREDIKSI CURAH HUJAN MENGGUNAKAN METODE ADAPTIVE-EXPECTATION
BASED MULTI-ATTRIBUTE FUZZY TIME SERIES**

SKRIPSI

**Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer**

**Disusun Oleh :
Yulia Trianandi
NIM: 115090601111019**

**Skripsi ini telah diuji dan dinyatakan lulus pada
3 Agustus 2018
Telah diperiksa dan disetujui oleh:**

Dosen Pembimbing I



**Wayan F. Mahmudy, S.Si, M.T, Ph.D
NIP: 19720919 199702 1 001**

Dosen Pembimbing II



**Candra Dewi, S.Kom, M.Sc
NIP: 19771114 200312 2 001**

**Mengetahui
Ketua Jurusan Teknik Informatika**



**Tji Astoto Kurniawan, S.T, M.T, Ph.D
NIP: 19710518 200312 1 001**



IDENTITAS TIM PENGUJI

Informasi pelaksanaan ujian dan penguji.

JADWAL
Jadwal pelaksanaan ujian

Hari/Tanggal	JUMAT / 03 AGUSTUS 2018
Tempat	FILKOM UB / E2.4
Waktu	07:30 - 08:45

- PENGUJI**
Majelis penguji ujian skripsi
- 
Dr.Eng. Ahmad Afif Supianto, S.Si, M.Kom (ke I)
 * ketua majelis
 NIK. 2012018206231001
 - 
Randy Cahya Wihandika, S.ST., M.Kom (ke II)
 NIK. 201405 880206 1 001

Penguji 1



Ahmad Afif Supianto, Dr.Eng., S.Si, M.Kom

- ☑ Laboratory of Intelligent Computing and Visualization
- ☑ Master of Computer Science/Informatics
- ☑ Komputasi Cerdas
- ☑ Informatics Engineering
- ☑ FILKOM UB, 8 Veteran Road
A building, 1st floor, Room A1.9
- ☑ afif.supianto[at]ub[dot]ac[dot]id

Penguji 2



Randy Cahya Wihandika, S.ST., M.Kom

- ☑ Laboratory of Intelligent Computing and Visualization
- ☑ Komputasi Cerdas
- ☑ Informatics Engineering
- ☑ FILKOM UB, 8 Veteran Road
A building, 1st floor, Room A1.5
- ☑ rendicahya[at]ub[dot]ac[dot]id



PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 12 Maret 2018



Yulia Trianandi

NIM: 115090601111019





DAFTAR RIWAYAT HIDUP

Nama : Yulia Trianandi
Jenis Kelamin : Laki-Laki
Tempat/Tanggal lahir : Purworejo, 15 Juli 1993
Alamat : Kaliurip, RT 01 / RW 01, Bener, Purworejo
Kode Pos : 54183
Email : yuliatrianandi@gmail.com
Telepon : 082132140369
Agama : Islam
Kewarganegaraan : WNI
Pendidikan Formal :

1. TK RAHAYU (1998-1999)
2. SD NEGERI KALIURIP (1999-2005)
3. SMP NEGERI 2 PURWOREJO (2005-2008)
4. SMA NEGERI 1 PURWOREJO (2008-2011)



UCAPAN TERIMA KASIH

Skripsi ini merupakan bagian dari tugas akhir penulis selama mengikuti perkuliahan dan sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer di Fakultas Ilmu Komputer, Universitas Brawijaya. Keberadaan skripsi ini tidak terlepas dari bimbingan dan bantuan dari berbagai pihak, untuk itu pada kesempatan ini penulis ingin menyampaikan rasa hormat dan terima kasih yang sebesar-besarnya kepada:

1. Wayan Firdaus Mahmudy, S.Si, M.T, Ph.D selaku dosen pembimbing pertama yang telah bersedia untuk membantu dan membimbing penulis dalam pengerjaan tugas akhir ini.
2. Candra Dewi, S.Kom, M.Sc selaku dosen pembimbing kedua yang telah bersedia untuk membantu dan membimbing penulis dalam pengerjaan tugas akhir ini.
3. Dr.Eng. Ahmad Afif Supianto, S.Si, M.Kom selaku ketua majelis penguji pada ujian skripsi tugas akhir ini.
4. Randy Cahya Wihandika, S.ST., M.Kom selaku majelis penguji pada ujian skripsi tugas akhir ini.
5. Seluruh dosen di FILKOM Universitas Brawijaya atas kesediaannya membagikan ilmunya kepada penulis.
6. Seluruh pihak, teman, keluarga tanpa terkecuali yang tidak dapat penulis sebutkan satu persatu yang telah ikut terlibat membantu, mendukung dan mendoakan dalam menyelesaikan tugas akhir ini.

ABSTRAK

Prediksi curah hujan sangat dibutuhkan dalam meningkatkan hasil produksi tanaman pangan pada suatu wilayah, salah satunya pada area Tengger. Kesalahan dalam memprediksi curah hujan dapat mengakibatkan kesalahan saat menentukan masa tanam, dan jenis tanaman yang tepat. Agar menghasilkan prediksi curah hujan dengan tingkat kesalahan sedikit, penelitian ini menggunakan metode Adaptive-Expectation Based Multi-Attribute Fuzzy Time Series. Metode tersebut telah terbukti mampu memprediksi closing price pada Taiwan Stock Exchange Capitalization Weighted Stock Index (TAEIX) dengan tingkat kesalahan lebih sedikit dibandingkan metode Fuzzy Time Series Chen. Penelitian ini akan menghasilkan prediksi curah hujan di empat kecamatan area Tengger yaitu Puspo, Sumber, Tosari, dan Tukur. Dari hasil pengujian 36 data pada tahun 2014, didapatkan nilai Mean Square Error (MSE) terbaik senilai 28,0470 pada kecamatan Tosari.

Kata Kunci : prediksi, curah hujan, *fuzzy time series*, *adaptive expectation*

ABSTRACT

Prediction of rainfall is needed in increasing the production of food crops in a region, one of them is Tengger area. Errors in predicting rainfall can cause errors when determining the planting period, and the right type of plant. In order to produce rainfall predictions with a slight error rate, this study uses the Adaptive-Expectation Based Multi-Attribute Fuzzy Time Series method. The method has been proven to predict the closing price on the Taiwan Stock Exchange Capitalization Weighted Stock Index (TAEIX) with fewer error rates than Chen Fuzzy Time Series methods. This study will produce rainfall predictions in the four districts of Tengger area, namely Puspo, Sumber, Tosari, and Tukur. From the results of testing 36 data in 2014, obtained the best Mean Square Error (MSE) value of 28.0470 in Tosari district.

Keywords : forecasting, rainfall, *fuzzy time series*, *adaptive expectation*

KATA PENGANTAR

Segala puji dan syukur hanya bagi Tuhan Yang Maha Esa, karena petunjuk dan anugerah-Nya yang sungguh besar dan nyata sehingga penulis mampu menyelesaikan pengerjaan skripsi dengan judul **“Prediksi Curah Hujan Menggunakan Metode *Adaptive-Expectation Based Multi-Attribute Fuzzy Time Series*”**

Skripsi ini merupakan bagian dari tugas akhir penulis selama mengikuti perkuliahan dan sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer di Fakultas Ilmu Komputer, Universitas Brawijaya. Keberadaan skripsi ini tidak terlepas dari bimbingan dan bantuan dari berbagai pihak, untuk itu pada kesempatan ini penulis ingin menyampaikan rasa hormat dan terima kasih yang sebesar-besarnya kepada:

1. Wayan Firdaus Mahmudy, S.Si, M.T, Ph.D selaku dosen pembimbing pertama yang telah bersedia untuk membantu dan membimbing penulis dalam pengerjaan tugas akhir ini.
2. Candra Dewi, S.Kom, M.Sc selaku dosen pembimbing kedua yang telah bersedia untuk membantu dan membimbing penulis dalam pengerjaan tugas akhir ini.
3. Seluruh dosen di FILKOM Universitas Brawijaya atas kesediaannya membagikan ilmunya kepada penulis.
4. Seluruh pihak, teman, keluarga tanpa terkecuali yang tidak dapat penulis sebutkan satu persatu yang telah ikut terlibat membantu, mendukung dan mendoakan dalam menyelesaikan tugas akhir ini.

Malang, 12 Maret 2018

Yulia Trianandi

yuliatrianandi@gmail.com

DAFTAR ISI

PENGESAHAN	ii
IDENTITAS TIM PENGUJI	iii
PERNYATAAN ORISINALITAS	iv
DAFTAR RIWAYAT HIDUP	vi
UCAPAN TERIMA KASIH	vii
ABSTRAK.....	viii
KATA PENGANTAR.....	ix
DAFTAR ISI.....	x
DAFTAR TABEL.....	xiii
DAFTAR GAMBAR.....	xiv
DAFTAR LAMPIRAN	xiv
BAB 1 PENDAHULUAN.....	1
1.1 Latar belakang.....	1
1.2 Rumusan masalah.....	2
1.3 Tujuan	2
1.4 Manfaat.....	2
1.5 Batasan masalah	2
1.6 Sistematika pembahasan.....	3
BAB 2 LANDASAN KEPUSTAKAAN	4
2.1 Kajian pustaka	4
2.2 Dasar teori.....	5
2.2.1 Curah hujan	6
2.2.2 <i>Adaptive Expectation</i>	6
2.2.3 <i>Fuzzy Time Series (FTS)</i>	6
2.2.4 Fuzzy C-Means.....	7
2.2.5 <i>Adaptive-Expectation Based Multi-Attribute Fuzzy Time Series</i> ...	8
2.2.6 <i>Mean Square Error</i>	9
BAB 3 METODOLOGI	10
3.1 Tahapan penelitian	10
3.1.1 Studi literatur	10
3.1.2 Perancangan proses	11

3.1.3 Implementasi algoritma	11
3.1.4 Pengujian.....	11
3.1.5 Kesimpulan dan saran	11
3.2 Teknik pengumpulan data	12
3.3 Algoritma yang digunakan	12
3.4 Lingkungan pengujian	12
3.5 Skenario pengujian	12
3.5.1 Skenario pengujian pengaruh jumlah data latih terhadap MSE .	13
3.5.2 Skenario pengujian pengaruh jumlah data uji terhadap MSE	13
3.5.3 Skenario pengujian pengaruh bobot terhadap MSE.....	13
3.5.4 Skenario pengujian pengaruh iterasi maksimal terhadap MSE..	14
3.5.5 Pengujian validasi sistem	14
BAB 4 Perancangan	16
4.1 Formulasi permasalahan.....	16
4.2 Siklus algoritma <i>Adaptive-Expectation Based Multi-Attribute Fuzzy Time Series</i>	17
4.3 Siklus penyelesaian masalah Menggunakan Algoritma <i>Adaptive-Expectation Based Multi-Attribute Fuzzy Time Series</i>	19
4.3.1 Pembentukan <i>subinterval</i> menggunakan <i>Fuzzy C-Means</i>	19
4.3.2 Pembentukan Himpunan Fuzzy.....	30
4.3.3 Fuzzifikasi	32
4.3.4 Pembentukan Fuzzy Logic Relationship (FLR).....	33
4.3.5 Fitnes nilai α	35
4.3.6 Defuzzifikasi.....	40
4.3.7 <i>Adaptive Expectation</i>	42
4.4 Perancangan antarmuka	43
BAB 5 IMPLEMENTASI	44
5.1 Implementasi Algoritma	44
5.1.1 Implementasi pembentukan <i>subinterval</i> dengan <i>Fuzzy C-Means</i>	44
5.1.2 Implementasi pembentukan himpunan Fuzzy.....	49
5.1.3 Implementasi proses fuzzifikasi	50
5.1.4 Implementasi Proses Pembentukan FLR.....	52



5.1.5 Implementasi proses <i>fitnes</i> nilai α	53
5.1.6 Implementasi proses defuzzifikasi	55
5.1.7 Implementasi proses Adaptive-Expectation pada hasil defuzzifikasi	56
5.2 Implementasi antarmuka.....	57
5.2.1 Antarmuka halaman <i>input</i>	57
5.2.2 Antarmuka halaman proses	57
5.2.3 Antarmuka halaman pengujian.....	59
BAB 6 Pengujian Dan ANALISIS	61
6.1 Sistematika pengujian.....	61
6.2 Hasil dan analisis pengujian	61
6.2.1 Pengujian pengaruh jumlah data latih terhadap nilai MSE	61
6.2.2 Pengujian pengaruh jumlah data uji terhadap nilai MSE	62
6.2.3 Pengujian pengaruh bobot terhadap nilai MSE	63
6.2.4 Pengujian pengaruh iterasi maksimal terhadap nilai MSE	64
6.2.5 Pengujian validasi sistem	65
BAB 7 Penutup	67
7.1 Kesimpulan.....	67
7.2 Saran	67
DAFTAR PUSTAKA.....	68
LAMPIRAN A HASIL PREDIKSI DATA UJI KECAMATAN PUSPO TAHUN 2014.....	70
LAMPIRAN B HASIL PREDIKSI DATA UJI KECAMATAN SUMBER TAHUN 2014	72
LAMPIRAN C HASIL PREDIKSI DATA UJI KECAMATAN TOSARI TAHUN 2014	74
LAMPIRAN D HASIL PREDIKSI DATA UJI KECAMATAN TUTUR TAHUN 2014	76



DAFTAR TABEL

Tabel 2.1 Kajian pustaka	5
Tabel 3.1 Skenario pengujian pengaruh jumlah data latih terhadap MSE	13
Tabel 3.2 Skenario pengujian pengaruh jumlah data latih terhadap MSE	13
Tabel 3.3 Skenario pengujian pengaruh bobot terhadap MSE.....	14
Tabel 3.4 Skenario pengujian pengaruh iterasi maksimum terhadap MSE.....	14
Tabel 3.5 Skenario pengujian validasi sistem	15
Tabel 4.1 Data sampel masukan	16
Tabel 4.2 Nilai kuadrat dari selisih data ke- i dengan rata-rata.....	20
Tabel 4.3 Nilai maksimum, nilai minimum, dan standard deviasi	21
Tabel 4.4 <i>Universe of Discourse</i>	21
Tabel 4.5 Matriks partisi awal	22
Tabel 4.6 Matriks partisi ternormalisasi <i>attribute A</i>	23
Tabel 4.7 Kuadrat μ_k dari <i>attribute A</i>	23
Tabel 4.8 Perhitungan pusat kluster <i>attribute A</i>	24
Tabel 4.9 Pusat kluster <i>attribute A</i>	25
Tabel 4.10 Hasil $(X_i - V_j)^2$ dari <i>attribute A</i>	25
Tabel 4.11 Perhitungan nilai objektif <i>attribute A</i>	25
Tabel 4.12 Perhitungan perbaikan matriks partisi <i>attribute A</i>	26
Tabel 4.13 Matriks partisi perbaikan	27
Tabel 4.14 Pusat kluster dari semua <i>attribute</i>	27
Tabel 4.15 Penentuan batas subinterval	29
Tabel 4.16 Himpunan Fuzzy <i>attribute A</i>	31
Tabel 4.17 Derajat keanggotaan <i>attribute A</i>	32
Tabel 4.18 Hasil fuzzifikasi semua <i>attribute</i> data	33
Tabel 4.19 <i>Fuzzy Logic Relationship</i>	34
Tabel 4.20 Hasil prediksi data latih	35
Tabel 4.21 Fitnes α	36
Tabel 4.22 Frekuensi FLR pada pusat kluster atribut utama.....	41
Tabel 4.23 Perbandingan data aktual dengan hasil prediksi	42
Tabel 4.24 Perbandingan MSE <i>Adaptive Expectation</i>	42
Tabel 6.1 Pengujian jumlah data latih.....	61

Tabel 6.2 Pengujian jumlah data uji.....	62
Tabel 6.3 Pengujian bobot	63
Tabel 6.4 Pengujian Iterasi maksimal.....	64
Tabel 6.5 Pengujian validasi sistem	65

DAFTAR GAMBAR

Gambar 2.1 Fungsi keanggotaan.....	8
Gambar 3.1 Diagram blok metodologi penelitian	10
Gambar 4.1 Diagram alir prediksi curah hujan menggunakan metode <i>Adaptive-Expectation Based Multi-Attribute Fuzzy Time Series</i>	18
Gambar 4.2 Grafik himpunan fuzzy <i>attribute A</i>	31
Gambar 4.3 Perancangan antarmuka	43
Gambar 5.1 Antarmuka halaman input	57
Gambar 5.2 Antarmuka proses subinterval	57
Gambar 5.3 Antarmuka proses fuzzifikasi	58
Gambar 5.4 Antarmuka proses defuzzifikasi	58
Gambar 5.5 Antarmuka pengujian jumlah data latih	59
Gambar 5.6 Antarmuka pengujian jumlah data uji.....	59
Gambar 5.7 Antarmuka pengujian bobot	60
Gambar 5.8 Antarmuka pengujian iterasi maksimal	60
Gambar 6.1 Grafik hasil pengujian pengaruh jumlah data latih terhadap MSE ...	62
Gambar 6.2 Grafik hasil pengujian jumlah data uji terhadap MSE.....	63
Gambar 6.3 Grafik hasil pengujian bobot terhadap MSE	64
Gambar 6.4 Grafik hasil pengujian iterasi maksimal terhadap MSE.....	65
Gambar 6.5 Grafik hasil pengujian validasi sistem	66

DAFTAR LAMPIRAN

LAMPIRAN A HASIL PREDIKSI DATA UJI KECAMATAN PUSPO TAHUN 2014.....	70
LAMPIRAN B HASIL PREDIKSI DATA UJI KECAMATAN SUMBER TAHUN 2014.....	72
LAMPIRAN C HASIL PREDIKSI DATA UJI KECAMATAN TOSARI TAHUN 2014	74
LAMPIRAN D HASIL PREDIKSI DATA UJI KECAMATAN TUTUR TAHUN 2014.....	76



BAB 1 PENDAHULUAN

1.1 Latar belakang

Data curah hujan merupakan salah satu aspek penting dalam beberapa sektor terutama sektor pertanian. Dengan mengetahui curah hujan yang akan terjadi, petani dapat menentukan masa tanam yang tepat untuk menghindari terjadinya gagal panen. Selain menentukan masa tanam, prediksi curah hujan dapat membantu untuk menentukan jenis tanaman pangan agar mendapatkan hasil panen terbaik (Mardawilis & Ritonga, 2016). Sehingga prediksi curah hujan sangat dibutuhkan dalam meningkatkan hasil produksi tanaman pangan terutama pada iklim ekstrim seperti El-Nino dan La-Nina (Suciantini, 2015).

Pertanian di Indonesia memiliki kontribusi yang cukup besar dalam kebutuhan pangan, dan ekspor, salah satunya adalah kentang. Tanaman kentang sangat tergantung pada tingkat curah hujan yang terjadi pada suatu daerah. Salah satu daerah produksi tanaman kentang yang berkembang dengan pesat di Indonesia adalah Tengger di Jawa Timur (Billah, 2013). Daerah Tengger meliputi beberapa kecamatan yaitu kecamatan Sumber, Puspo, Tukur, dan Tosari. Sehingga penelitian tentang prediksi curah hujan pada kecamatan tersebut sangat dibutuhkan untuk meningkatkan produktivitas tanaman kentang di area Tengger.

Pendekatan tentang prediksi curah hujan pada empat tempat di area Tengger telah dilakukan oleh beberapa penelitian sebelumnya, salah satunya penelitian untuk meramalkan curah hujan menggunakan metode *Generalized Model Space Time Autoregressive-Seemingly Unrelated Regression* (Iriani, et al., 2015). Pada area tersebut juga pernah dilakukan beberapa penelitian untuk meramalkan curah hujan, diantaranya menggunakan metode *Fuzzy Neural Networks* (Utomo, et al., 2017), dan metode *Hybrid Adaptive Neuro Fuzzy Inference System* dengan Algoritma Genetika (Wahyuni & Mahmudy, 2017). Data pada ketiga penelitian tersebut akan digunakan ulang pada penelitian ini dengan tujuan pengembangan pendekatan mengenai curah hujan.

Selain di area Tengger, prediksi curah hujan juga pernah dilakukan di kota Samarinda menggunakan *Fuzzy Time Series Chen* (Fauziah, et al., 2016). Sedangkan dalam penelitian ini menggunakan metode *Adaptive-Expectation Based Multi-Attribute Fuzzy Time Series* untuk memprediksi curah hujan di area Sumber, Puspo, Tukur, dan Tosari. Metode ini pernah digunakan untuk meramalkan *Taiwan Stock Exchange Capitalization Weighted Stock Index* (TAEIX) dengan hasil *Average Error Percentage* (MAER) lebih kecil dibandingkan dengan metode *Fuzzy Time Series Chen* (Chen, 1996) (Liu, et al., 2010). Pada penelitian tersebut, *Adaptive-Expectation* digunakan untuk memperbaiki kesalahan dari hasil prediksi yang berasal dari data historis yang bersifat linguistik. *Adaptive-Expectation* juga pernah digunakan untuk memperbaiki kesalahan prediksi metode *Adaptive Neuro Fuzzy Inference System* pada *TAEIX Forecasting* (Wei, 2013). Oleh karena itu penulis diharapkan mampu meramalkan curah hujan

menggunakan metode *Adaptive-Expectation Based Multi-Attribute Fuzzy Time Series* dengan memperoleh tingkat kesalahan prediksi sedikit.

1.2 Rumusan masalah

Berdasarkan latar belakang penelitian ini, maka diperoleh beberapa rumusan masalah sebagai berikut:

1. Bagaimana mengimplementasikan metode *Adaptive-Expectation Based Multi-Attribute Fuzzy Time Series* untuk memprediksi curah hujan?
2. Bagaimana tingkat kesalahan dari hasil prediksi curah hujan menggunakan metode *Adaptive-Expectation Based Multi-Attribute Fuzzy Time Series*?

1.3 Tujuan

Tujuan:

1. Mengimplementasikan metode *Adaptive-Expectation Based Multi-Attribute Fuzzy Time Series* untuk memprediksi curah hujan.
2. Menguji tingkat kesalahan dari hasil prediksi curah hujan menggunakan metode *Adaptive-Expectation Based Multi-Attribute Fuzzy Time Series*.

1.4 Manfaat

Dengan adanya penelitian ini diharapkan memberi manfaat sebagai berikut:

1. Dapat membantu penduduk daerah Puspo, Sumber, Tosari, Tukur, Jawa Timur khususnya sektor pertanian agar hasil panen tanaman pangan menjadi lebih baik.
2. Dapat membantu Badan Meteorologi dan Geofisika dalam memprediksi curah hujan agar mendapatkan hasil yang lebih akurat.
3. Dapat membantu Badan Penelitian dan Pengembangan (Balitbang) dalam membuat Kalender Tanam (KATAM) dengan menggunakan data hasil prediksi curah hujan yang lebih akurat.

1.5 Batasan masalah

Penelitian ini memiliki beberapa batasan masalah sebagai berikut :

1. Faktor perubahan atmosfer dan gejala iklim ekstrim seperti El-Nino dan La-Nina tidak mempengaruhi hasil prediksi curah hujan pada penelitian ini.
2. Hasil yang didapat dari penelitian ini adalah prediksi jumlah debit curah hujan berdasarkan data curah hujan, kelembaban, suhu, dan lama penyinaran matahari selama sepuluh tahun.
3. Area yang digunakan sebagai studi kasus adalah empat kecamatan Sumber, Puspo, Tukur, dan Tosari.

1.6 Sistematika pembahasan

Berikut sistematika penulisan yang diterapkan pada penelitian ini:

BAB I Pendahuluan

Bab ini membahas tentang latar belakang, rumusan masalah, tujuan, manfaat, batasan masalah, dan sistematika penulisan.

BAB II Landasan Kepustakaan

Bab ini membahas tentang dasar teori dan referensi penelitian sebelumnya untuk pemahaman permasalahan yang dibahas pada penelitian ini. Teori-teori yang dibahas pada bab ini adalah curah hujan, *Adaptive Expectation*, *Fuzzy Time Series*, *Fuzzy C-Means*, *Adaptive-Expectation based Multi-Attribute Fuzzy Time Series*, dan *Mean Square Error (MSE)*.

BAB III Metodologi

Bab ini membahas tentang langkah-langkah yang dilalui dalam menyelesaikan penelitian ini, meliputi studi literatur, perancangan proses, implementasi algoritma, pengujian, dan pengambilan kesimpulan. Selain itu bab ini juga membahas teknik pengumpulan data, algoritma yang digunakan, lingkungan pengujian dan skenario pengujian.

BAB IV Perancangan

Bab ini berisi tentang formulasi masalah yang harus diselesaikan, siklus algoritma *Adaptive-Expectation based Multi-Attribute Fuzzy Time Series*, siklus penyelesaian masalah metode *Adaptive-Expectation Based Multi-Attribute Fuzzy Time Series* dalam bentuk perhitungan manual, dan perancangan antarmuka.

BAB V Implementasi

Bab ini berisi tentang struktur class yang menyusun program beserta potongan kode program dalam penyelesaian masalah menggunakan metode *Adaptive-Expectation based Multi-Attribute Fuzzy Time Series* dan implementasi antarmuka.

BAB VI Pengujian dan Pembahasan

Bab ini berisi tentang pengujian metode yang digunakan dengan menghitung nilai *Mean Square Error (MSE)* sebagai tolak ukur pengujian. Pada bab ini menjelaskan hasil dan analisis dari perhitungan MSE pada setiap skenario pengujian.

BAB VII Penutup

Bab ini menguraikan kesimpulan akhir yang didapatkan dari penelitian dan saran-saran untuk pengembangan penelitian.

BAB 2 LANDASAN KEPUSTAKAAN

Dalam bab ini terdapat bagian kajian pustaka dan dasar teori yang berhubungan dengan prediksi curah hujan. Dalam kajian pustaka dibahas tentang penelitian-penelitian terdahulu yang menjadi perbandingan, dan dasar dalam penulisan penelitian ini. Dalam bagian dasar teori berisi tentang konsep, metode, model, dan teori dari berbagai sumber pustaka yang digunakan dalam penelitian.

2.1 Kajian pustaka

Dalam melakukan sebuah penelitian, beberapa temuan dari penelitian terdahulu sangat penting karena dapat dijadikan sebagai materi pendukung dalam penelitian yang akan dilakukan. Pada bagian ini akan dibahas secara singkat tentang beberapa penelitian terdahulu yang cukup relevan dengan permasalahan atau metode yang digunakan dalam penelitian ini.

Penelitian terdahulu pernah dilakukan oleh Utomo pada tahun 2017 pada area Tengger, yaitu melakukan penelitian terhadap prakiraan curah hujan *time series*, metode yang digunakan adalah kombinasi antara Logika Fuzzy dan Jaringan Syaraf Tiruan (Utomo, et al., 2017). Dalam penelitiannya menggunakan masukan berupa data curah hujan, dengan hasil evaluasi menggunakan *Root Mean of Square Error* (RMSE) 2,399.

Di kota Samarinda tahun 2016, Fauziah, Wahyuningsih, dan Nasution melakukan peramalan curah hujan menggunakan metode *Fuzzy Time Series* Chen. Hasil peramalan curah hujan tersebut mendapatkan nilai kesalahan RMSE sebesar 79,96 untuk data 2011 sampai 2016, 84,91 untuk data 2013 sampai 2016, dan 73,68 untuk data 2014 sampai 2016.

Pada tahun 2010, Liu, Cheng, dan Chen melakukan penelitian dengan objek *Taiwan Stock Exchange Capitalization Weighted Stock Index* (TAIEX) menggunakan metode *Adaptive-Expectation Based Multi-Attribute Fuzzy Time Series* (Liu, et al., 2010). Kesimpulan pada penelitian ini adalah metode ini dapat digunakan untuk memprediksi Indeks pada TAIEX dan mendapatkan nilai kesalahan MAER sebesar (1.36% untuk 2002, 0.94% untuk 2003, dan 1.051% untuk 2004) dalam setiap periode pengujiannya.

Pada penelitian ini membahas tentang prediksi curah hujan menggunakan metode *Adaptive-Expectation Based Multi-Attribute Fuzzy Time Series*. Penelitian ini menggunakan empat data masukan yaitu curah hujan, suhu, kelembaban, dan lama penyinaran dengan data curah hujan sebagai *attribute* utamanya. Pada penelitian ini akan menggunakan *Mean Square Error* (MSE) untuk mengukur tingkat kesalahan prediksi curah hujan yang diusulkan.

Setelah menganalisis dan mempelajari beberapa penelitian terdahulu, penulis kemudian membandingkan objek penelitian, metode yang digunakan dalam penelitian, dan hasil penelitian tersebut dapat dilihat pada Tabel 2.1.

Tabel 2.1 Kajian pustaka

No	Judul	Objek dan Input	Metode dan Nilai Evaluasi
1.	Kombinasi Logika Fuzzy Dan Jaringan Syaraf Tiruan Untuk Prakiraan Curah Hujan Timeseries Di Area Puspo – Jawa Timur (Utomo, et al., 2017)	<u>Objek</u> : Curah hujan <u>Input Penelitian</u> : Data curah hujan	<u>Metode</u> : Logika Fuzzy dan Jaringan Syaraf Tiruan <u>Nilai evaluasi</u> : <i>Root Mean of Square Error (RMSE)</i> dengan nilai 2,399.
2.	Peramalan Menggunakan Fuzzy Time Series Chen-Studi Kasus: Curah Hujan Kota Samarinda (Fauziah, et al., 2016)	<u>Objek</u> ; Curah hujan <u>Input Penelitian</u> : Data curah hujan	<u>Metode</u> : Fuzzy Time Series Chen <u>Nilai Evaluasi</u> : <i>Root Mean of Square Error (RMSE)</i> dengan nilai (79.96, 84.91, 73.68) untuk setiap periode testnya.
3.	<i>Adaptive-Expectation Based Multi-Attribute FTS Model For Forecasting TAIEX</i> (Liu, et al., 2010)	<u>Objek</u> : TAIEX <u>Input Penelitian</u> : <ul style="list-style-type: none"> • <i>high price</i> • <i>low price</i> • <i>close price</i> • <i>open price</i> 	<u>Metode</u> : <i>Adaptive-Expectation</i> dan <i>Multi-Attribute Fuzzy Time Series</i> <u>Nilai evaluasi</u> : <i>MAER</i> (1.36%, 0.94%, 1.051%) di setiap periode testnya.
4.	Prediksi Curah Hujan Menggunakan Metode <i>Adaptive-Expectation Based Multi-Attribute Fuzzy Time Series</i>	<u>Objek</u> : Curah hujan <u>Input Penelitian</u> : <ul style="list-style-type: none"> • curah hujan • suhu • kelembaban • lama penyinaran 	<u>Metode</u> : <i>Adaptive-Expectation</i> dan <i>Multi-Attribute Fuzzy Time Series</i>

2.2 Dasar teori

Dalam bagian ini membahas tentang konsep, metode, model, dan teori yang digunakan pada penelitian ini, meliputi curah hujan, *Fuzzy Time Series*, *Fuzzy C-Means*, metode *Adaptive-Expectation Based Multi-Attribute Fuzzy Time Series*, dan *Mean Square Error*.



2.2.1 Curah hujan

Hujan merupakan air yang jatuh ke bumi karena titik air yang tertampung di awan ataupun udara melebihi kapasitas awan. Sedangkan pengertian dari curah hujan adalah air hujan yang ditampung dan diukur pada permukaan dengan luas 1 meter persegi (m^2). Curah hujan menggunakan satuan milimeter (mm), sehingga curah hujan 1 mm berarti bahwa terkumpulnya air hujan pada tempat seluas $1m^2$ dengan kedalaman atau ketinggian air sebesar 1mm. Curah hujan yang terukur kurang dari 0,1 mm akan dicatat sebagai 0.

2.2.2 Adaptive Expectation

Adaptive Expectation merupakan sebuah teori dari ilmu ekonomi dengan dasar apa yang akan terjadi di masa yang akan datang sama dengan yang telah terjadi di masa lampau (Kmenta, 1971). *Adaptive Expectation* digunakan untuk memperbaiki kesalahan prediksi yang akan datang dengan mempelajari kesalahan prediksi sebelumnya. Untuk mendapatkan hasil prediksi yang akan datang, dibutuhkan data aktual dan kesalahan prediksi pada periode sebelumnya (Liu, et al., 2010). Misalkan $P(t)$ adalah data pada waktu t , $P(t - 1)$ adalah data pada waktu $(t - 1)$, maka untuk memprediksi data pada waktu t dapat dilihat pada persamaan (2.2).

$$P(t) = P(t - 1) + h_0 \times \varepsilon(t - 1) \quad (2.2)$$

Keterangan:

$\varepsilon(t - 1)$ = kesalahan prediksi.

h_0 = bobot.

2.2.3 Fuzzy Time Series (FTS)

Fuzzy Time Series diperkenalkan oleh Song, dan Chissom pada tahun 1993 untuk memprediksi masalah menggunakan data *time series* yang bersifat linguistik (Song & Chissom, 1993). Kemudian banyak peneliti mengembangkan *Fuzzy Time Series* dengan mengubah *interval* ataupun ordernya. Pada tahun 1996, Chen menemukan model *Fuzzy Time Series* dengan akurasi yang lebih baik dan dikenal dengan metode *Fuzzy Time Series* Chen (Chen, 1996). Perhitungan metode *Fuzzy Time Series* Chen termasuk paling mudah dan memiliki performa yang baik. Berikut langkah-langkah *Fuzzy Time Series* Chen :

- Langkah pertama yaitu menentukan *Universe of Discourse* dari bilangan aktual ke dalam subinterval dengan interval yang sama.
- Langkah kedua untuk menentukan *fuzzy set* menggunakan *Universe of Discourse* dan menghasilkan A_1, A_2, \dots, A_n dengan nilai linguistik sesuai dengan masing-masing *Universe of Discourse*.
- Langkah ketiga fuzzifikasi set data historis dari bilangan crisp ke bilangan fuzzy dalam bentuk derajat keanggotaan. Dari beberapa derajat keanggotaan kemudian dipilih satu *fuzzy set* yang memiliki derajat keanggotaan tertinggi.
- Langkah keempat membuat tabel *Fuzzy Logical Relationships* (FLR) berdasarkan data historis.

- e. Mengklasifikasikan FLR yang telah diperoleh dari tahap ke-3 ke dalam grup-grup sehingga terbentuk *Fuzzy Logical Relationship Group* (FLRG) dan mengkombinasikan hubungan yang sama.
- f. Defuzzifikasi nilai peramalan.

2.2.4 Fuzzy C-Means

Fuzzy klustering adalah salah satu teknik untuk menentukan kluster optimal dalam suatu ruang vektor yang didasarkan pada bentuk normal euclidean untuk jarak antar vektor. Fuzzy C-Means (FCM) adalah salah satu contoh metode yang didasari oleh fuzzy klasterisasi (Bezdek, 1981). Tujuan dari algoritma FCM yaitu untuk menemukan pusat kluster (*centroid*) dengan menggunakan fungsi objektif dan iterasi maksimal sebagai kondisi berhenti. Algoritma dari FCM dijelaskan sebagai berikut (Kusumadewi & Purnomo, 2010):

- a. Memasukkan data yang akan diklasterisasi.
- b. Menentukan nilai variabel jumlah kluster, bobot, iterasi maksimum, dan error terkecil yang diharapkan.
- c. Membangkitkan bilangan *random* μ_{ik} ; $i = 1, 2, 3, \dots, n$, dan $k = 1, 2, 3, \dots, c$ (n = jumlah data sampel, c = jumlah *cluster* (berupa kolom) yang akan dibentuk) sebagai elemen-elemen matriks partisi awal. Kemudian menghitung normalisasi matriks partisi. Sehingga di dapatkan matriks derajat keanggotaan dari bilangan acak, dengan jumlah setiap barisnya bernilai 1. Langkah c dapat dilihat pada Persamaan 2.3.

$$\begin{aligned} 0 \leq \mu_{ik} \leq 1, \quad \forall i, k, \\ 0 < \sum_{k=1}^c \mu_{ik} \leq n, \quad \forall i, \\ \sum_{i=1}^n \mu_{ik} = 1, \quad \forall k, \end{aligned} \tag{2.3}$$

Keterangan:

μ_{ik} = bilangan acak (jumlah dari semua kelas dalam satu data = 1).

n = jumlah data (berupa baris).

k = jumlah kluster (berupa kolom).

- d. Menghitung pusat kluster ke-k: V_{kj} , ($k = 1, 2, 3, \dots, c$ dan $j = 1, 2, 3, \dots, m$). Perhitungan langkah d dapat dilihat pada persamaan 2.4.

$$V_{kj} = \frac{\sum_{i=1}^n ((\mu_{ik})^w \times X_{ij})}{\sum_{i=1}^n \mu_{ik}^w} \tag{2.4}$$

Keterangan:

V_{kj} = pusat kluster pada kluster ke-k dan atribut ke-j.

μ_{ik} = derajat keanggotaan pada data ke-i, kluster ke-k.

X_{ij} = data sampel pada data ke-i, attribute ke-j.

w = pembobot.

- e. Menghitung fungsi objektif pada iterasi ke-t, menggunakan persamaan 2.5.

$$P_t = \sum_{i=1}^n \sum_{k=1}^c ([\sum_{j=1}^m (X_{ij} - V_{kj})^2] \times (\mu_{ik})^w) \tag{2.5}$$

- f. Menghitung perbaikan matriks partisi.
- g. Memeriksa kondisi berhenti, iterasi akan berhenti ketika sudah mencapai iterasi maksimal yang telah ditentukan atau jika selisih antar



nilai objektif (t) dikurangi nilai objektif ($t - 1$) kurang dari *error* minimal.

2.2.5 Adaptive-Expectation Based Multi-Attribute Fuzzy Time Series

Metode ini diperkenalkan oleh Liu, Chen, dan Cheng pada tahun 2010 (Liu, et al., 2010) dengan langkah-langkah sebagai berikut :

- a. Menentukan jumlah data dan attribute untuk dataset. Misalkan data *time series* S dengan n data dan m atribut. Maka dapat ditulis sebagai $S_j(t)$ dengan keterangan $t = 1, 2, \dots, n$ dan $j = 1, 2, \dots, m$.
- b. Melakukan klasterisasi pada data latih *time series* S menggunakan *Fuzzy C-Means* ke dalam jumlah klaster (c) yang ditentukan dengan aturan ($2 \leq c \leq n$). Berdasarkan keterbatasan manusia dalam memproses informasi pada penelitian Miller, angka tujuh digunakan sebagai jumlah klaster pada metode ini (Miller, 1956). Sehingga didapatkan tujuh pusat klaster V_{ij} dengan perhitungan *Fuzzy C-Means* secara *iterative* pada persamaan 2.3 dan persamaan 2.4. Setelah pusat setiap *cluster* didapatkan, maka terbentuklah matriks $\mu_{it}(t = 1, 2, \dots, n)$
- c. Melakukan perankingan secara *ascending* dari setiap klaster pada atribut utama dan mengubah data *time series* ke dalam *Fuzzy set* (fuzzifikasi). Perankingan pusat klaster L_r ($r = 1, 2, 3, \dots, c$) digunakan untuk menentukan nilai liguistik pada fungsi keanggotaan.



Gambar 2.1 Fungsi keanggotaan

Sumber: Liu (2010)

- d. Membentuk *Fuzzy Logical Relationship* (FLR) dari data latih. FLR pertama terbentuk pada data ketiga. FLR kiri tersusun dari *attribute* pendukung dari data ketiga itu sendiri dan 2 data sebelumnya, sedangkan FLR kanan berisi *attribute* utama data ketiga itu sendiri. Setiap FLR kanan menunjuk pada FLR kanan selanjutnya $A_i \rightarrow A_{(i+1)}$, $\forall (i = 1, 2, \dots, \text{jumlah klaster})$, kemudian dikelompokkan sesuai dengan ruas kiri dari kombinasi tersebut. Misalnya jika $A_1 \rightarrow A_1$, $A_1 \rightarrow A_3$, $A_1 \rightarrow A_6$, maka dapat dikelompokkan menjadi $A_1 \rightarrow A_1, A_3, A_6$. (2.6)
- e. Melakukan defuzzifikasi sesuai FLR. FLR kiri digunakan untuk mencari FLR yang paling cocok (terpilih) dengan menghitung selisihnya. Setelah mendapatkan FLR terpilih, defuzzifikasi dilakukan sesuai kombinasi FLR kanan dari FLR terpilih tersebut. Misal A_i merupakan FLR

kanan terpilih dari data uji F_t , dan P_i merupakan pusat klasternya ($t = 1, 2, \dots, n \forall n = \text{jumlah data}$), maka hasil prediksi ($F_{(t+1)}$) dilakukan sesuai kondisi sebagai berikut

Rule 1: $A_i \rightarrow \text{kosong}$, maka $F_{(t+1)} = P_i$
 (sesuai *Naive Forecasting Principle*) (2.7)

Rule 2: $A_i \rightarrow A_j$, maka $F_{(t+1)} = P_j$

Rule 3: $A_i \rightarrow A_1, A_2, \dots, A_k$,
 maka $F_{(t+1)} = (P_1 + P_2 + \dots + P_k)/k$

- f. Memperbaiki hasil prediksi dengan *Adaptive-Expectation* menggunakan parameter alpha (α) secara iteratif dari 0 sampai 1 dengan jarak 0.01. $0 < \alpha < 1$. Menyimpan nilai α terbaik dengan nilai MSE terkecil dari setiap iterasinya, dan menggunakan α tersebut untuk mengadaptasi hasil prediksi pada data uji. Langkah ini menggunakan persamaan 2.8.

$$Pn(t) = A(t - 1) + \alpha(P(t) - A(t - 1)) \tag{2.8}$$

Keterangan:

Pn = *Adapted Forecast* (hasil prediksi yang baru)

A = Data aktual

P = Hasil Prediksi

t = *time series* (urutan data)

2.2.6 Mean Square Error

Mean Square Error (MSE) merupakan salah satu metode untuk mengetahui tingkat kesalahan prediksi dengan mengkuadratkan selisih antara data aktual dengan hasil prediksi kemudian dilakukan perataan kesalahan prediksi tersebut untuk semua data (Suryaningrum & Wijaya, 2015). Perhitungan MSE dapat dilihat pada persamaan (2.7).

$$MSE = \frac{1}{n} \times \sum_{t=0}^n (X_t - F_t)^2 \tag{2.9}$$

Keterangan:

n = jumlah data

X = data aktual

F = hasil prediksi

t = waktu / urutan data

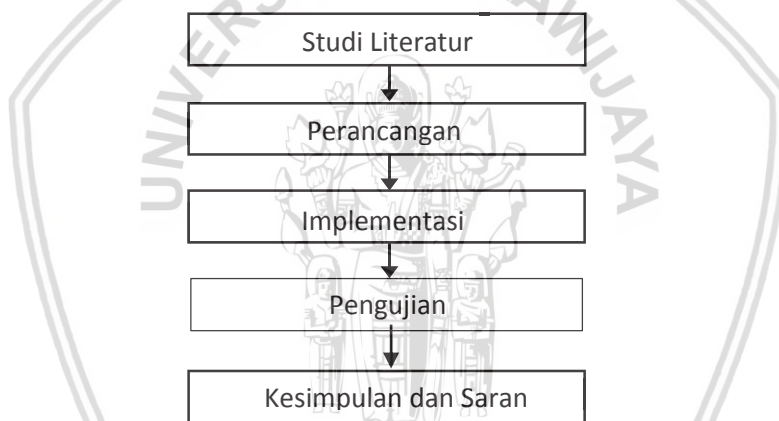


BAB 3 METODOLOGI

Bab ini membahas tentang tahapan atau proses pengerjaan penelitian untuk meramalkan curah hujan menggunakan metode *Adaptive-Expectation Based Multi-Attribute Fuzzy Time Series* di area Sumber, Puspo, Tukur, Dan Tosari – Jawa Timur. Penelitian ini termasuk dalam penelitian non-implementatif dengan pendekatan analitik (*analytic*). Penelitian ini akan menghasilkan perangkat lunak (*software*) yang dapat digunakan untuk meramalkan curah hujan.

3.1 Tahapan penelitian

Tahapan-tahapan yang dilakukan dalam penelitian ini meliputi: studi literatur, pengumpulan data, analisis kebutuhan sistem, perancangan sistem, implementasi sistem, pengujian sistem, dan pengambilan kesimpulan. Ilustrasi mengenai tahapan dalam penelitian ini dapat dilihat dalam bentuk diagram blok seperti pada Gambar 3.1.



Gambar 3.1 Diagram blok metodologi penelitian

3.1.1 Studi literatur

Studi literatur merupakan tahapan mempelajari literatur dari berbagai sumber pustaka sebagai pendukung dalam melakukan penelitian ini. Literatur berupa teori, maupun informasi yang berkaitan dengan penelitian diperoleh dari buku, internet, jurnal-jurnal penelitian terdahulu, serta dari proses bimbingan dengan dosen pembimbing. Informasi dan teori yang dipelajari antara lain:

- Curah hujan.
- Adaptive Expectation*.
- Fuzzy Time Series*.
- Fuzzy C-Means*.
- Metode *Adaptive-Expectation Based Multi-Attribute Fuzzy Time Series*.
- Mean Square Error*.

3.1.2 Perancangan proses

Pada tahap perancangan penelitian ini membahas tentang perancangan yang dilakukan untuk dapat melakukan implementasi. Pada sub-bab ini meliputi perancangan tahapan-tahapan yang didasari dari hasil pengumpulan data dan analisis kebutuhan yang sudah dilakukan. Pada sub-bab perancangan ini dijelaskan mengenai desain dan arsitektur sistem prediksi curah hujan dengan metode *Adaptive-Expectation Based Multi-Attribute Fuzzy Time Series*, meliputi perancangan flowchart tahapan-tahapan proses metode, perancangan perhitungan manualisasi data menggunakan aplikasi Excel, perancangan gambaran sederhana tentang antarmuka, dan skenario pengujian metode untuk mengetahui tingkat kesalahan sistem dalam mengimplementasikan metode untuk meramalkan curah hujan.

3.1.3 Implementasi algoritma

Sistem ini akan diimplementasikan menggunakan bahasa pemrograman Java. Berikut ini merupakan proses atau langkah-langkah pengimplementasian sistem :

- a. Memasukkan data yang berbentuk teks (.txt) yang nantinya akan digunakan sebagai masukan ke dalam sistem.
- b. Memasukkan proses perhitungan *Adaptive-Expectation Based Multi-Attribute Fuzzy Time Series* ke dalam sistem.
- c. Membuat antarmuka sistem

3.1.4 Pengujian

Pengujian perlu dilakukan untuk mengukur seberapa baik kemampuan sistem yang dibangun, dan menunjukkan pengaruh variabel inputan terhadap hasil prediksi. Proses pengujian pada penelitian ini yaitu pengujian variabel masukan dan validasi sistem dengan menggunakan tingkat akurasi sebagai tolak ukur. Perhitungan tingkat akurasi sistem menggunakan indikator *Mean Square Error* (MSE).

3.1.5 Kesimpulan dan saran

Proses pengambilan kesimpulan baru dapat dilakukan ketika seluruh tahapan dalam penelitian ini (perancangan, implementasi, dan pengujian) telah selesai dilakukan. Kesimpulan diambil melalui hasil pengujian yang dilakukan terhadap sistem. Tahap akhir dari penulisan penelitian ini adalah pemberian saran untuk memperbaiki kesalahan-kesalahan yang ada dan memberikan masukan untuk pengembangan sistem ini selanjutnya.

3.2 Teknik pengumpulan data

Data yang digunakan dalam penelitian ini diperoleh dari Badan Meteorologi Klimatologi dan Geofisika (BMKG) Karangploso. Data yang diambil merupakan data curah hujan, suhu, kelembaban udara, dan lama penyinaran matahari pada area Tengger, yaitu Sumber, Puspo, Tuter, dan Tosari. Data pada daerah tersebut telah digunakan pada penelitian-penelitian tentang curah hujan meliputi penelitian Utomo, dan Mahmudy (2016), Iriany, Mahmudy, Handoyo, Sulistiono, dan Nisak tahun 2015 (Iriani, et al., 2015), Utomo, Mahmudy, dan Anam tahun 2017 (Utomo, et al., 2017), dan Wahyuni, dan Mahmudy tahun 2017 (Wahyuni & Mahmudy, 2017). Data tersebut digunakan ulang dalam penelitian ini dengan tujuan pengembangan dari pendekatan sebelumnya mengenai prediksi curah hujan.

3.3 Algoritma yang digunakan

Penelitian ini menggunakan algoritma *Adaptive-Expectation Based Multi-Attribute Fuzzy Time Series* yang seperti telah diuraikan pada Bab 2 telah terbukti efektif digunakan untuk menghasilkan hasil prediksi atau peramalan. Implementasi algoritma menggunakan bahasa pemrograman Java karena dalam penelitian ini akan menghasilkan prototype aplikasi dekstop.

3.4 Lingkungan pengujian

Agar perangkat lunak berjalan dengan baik maka pengujian perangkat lunak dijalankan pada personal computer dengan spesifikasi sebagai berikut:

1. Kebutuhan hardware
 - a. Laptop dengan processor Intel core i7
 - b. Memori 4GB
2. Kebutuhan software
 - a. Sistem operasi Windows 10 64-bit
 - b. Database MySQL
 - c. Tool pemrograman Netbeans
 - d. Java JDK, JDBC

3.5 Skenario pengujian

Pada penelitian ini akan menggunakan 36 data dasarian setiap tahunnya dari tahun 2012 sampai dengan 2014 dalam proses pengujiannya. Proses pengujian pada penelitian ini dibagi menjadi 2 skenario pengujian yaitu skenario pengujian pengaruh variabel yang diuji, dan skenario pengujian dengan variabel terbaik menggunakan data set yang berbeda.

Skenario pengujian pertama bertujuan untuk mengetahui pengaruh nilai variabel-variabel yang digunakan di dalam perhitungan terhadap nilai MSE yang

dihasilkan. Variabel-variabel yang diuji pada skenario ini yaitu nilai jumlah data latih, jumlah data uji, bobot, dan iterasi maksimum.

3.5.1 Skenario pengujian pengaruh jumlah data latih terhadap MSE

Pengujian ini bertujuan untuk mengetahui pengaruh jumlah data latih terhadap MSE yang didapatkan dari hasil prediksi. Pada pengujian ini dilakukan 5 kali pengujian untuk setiap jumlah data latih dengan nilai variabel lainnya tetap, kemudian dihitung kesalahan prediksi menggunakan MSE.

Tabel 3.1 Skenario pengujian pengaruh jumlah data latih terhadap MSE

Jumlah data latih	Uji ke-					Rata-Rata	
	1	2	3	4	5	MSE	Lama Komputasi (millisecond)
36
72
108
144

3.5.2 Skenario pengujian pengaruh jumlah data uji terhadap MSE

Pengujian ini bertujuan untuk mengetahui pengaruh jumlah data uji terhadap MSE yang didapatkan dari hasil prediksi. Pada pengujian ini dilakukan 5 kali pengujian untuk setiap jumlah data uji dengan nilai variabel lainnya tetap, kemudian dihitung kesalahan prediksi menggunakan MSE.

Tabel 3.2 Skenario pengujian pengaruh jumlah data latih terhadap MSE

Jumlah data uji	Uji ke-					Rata-Rata	
	1	2	3	4	5	MSE	Lama Komputasi (millisecond)
36
72
108
144

3.5.3 Skenario pengujian pengaruh bobot terhadap MSE

Pengujian ini bertujuan untuk mengetahui pengaruh nilai bobot terhadap MSE yang dihasilkan. Variabel Bobot dibutuhkan untuk menghasilkan matriks partisi pada algoritma *Fuzzy C-Means*. Pada pengujian ini dilakukan 5 kali pengujian



untuk setiap bobot dengan nilai variabel lainnya tetap, kemudian dihitung kesalahan peramalan menggunakan MSE.

Tabel 3.3 Skenario pengujian pengaruh bobot terhadap MSE

Bobot	Uji ke-					Rata-Rata	
	1	2	3	4	5	MSE	Lama Komputasi (millisecond)
2
3
4
5

3.5.4 Skenario pengujian pengaruh iterasi maksimal terhadap MSE

Pengujian ini bertujuan untuk mengetahui pengaruh iterasi maksimal terhadap MSE yang didapatkan dari hasil prediksi. Iterasi maksimum dibutuhkan untuk menghasilkan pusat klaster pada algoritma *Fuzzy C-Means*. Pada pengujian ini dilakukan 5 kali pengujian untuk setiap nilai iterasi maksimum dengan nilai variabel lainnya tetap, kemudian dihitung kesalahan peramalan menggunakan MSE.

Tabel 3.4 Skenario pengujian pengaruh iterasi maksimum terhadap MSE

Iterasi Maksimal	Uji ke-					Rata-Rata	
	1	2	3	4	5	MSE	Lama Komputasi (millisecond)
3
4
5
6
7
8

3.5.5 Pengujian validasi sistem

Skenario pengujian yang ketiga dilakukan sebanyak 5 kali uji coba dengan 4 data yang berbeda untuk menguji kestabilan sistem dengan menggunakan kombinasi dari variabel yang telah didapatkan dari pengujian yang pertama. Keempat data tersebut adalah data tahun 2010 sampai 2014 dari kecamatan Puspo, Sumber, Tosari, dan Tutur.



Tabel 3.5 Skenario pengujian validasi sistem

Kecamatan	Uji ke-					Rata-Rata	
	1	2	3	4	5	MSE	Lama Kalkulasi (milisecond)
Puspo	
Sumber	
Tosari	
Tutur	



BAB 4 PERANCANGAN

Bab ini membahas tentang perancangan perangkat lunak prediksi curah hujan dengan menggunakan metode *Adaptive-Expectation Based Multi-Attribute Fuzzy Time Series*. Perancangan pada perangkat lunak ini terdiri dari formulasi permasalahan, siklus algoritma, siklus penyelesaian permasalahan, perancangan antarmuka, dan perancangan pengujian.

4.1 Formulasi permasalahan

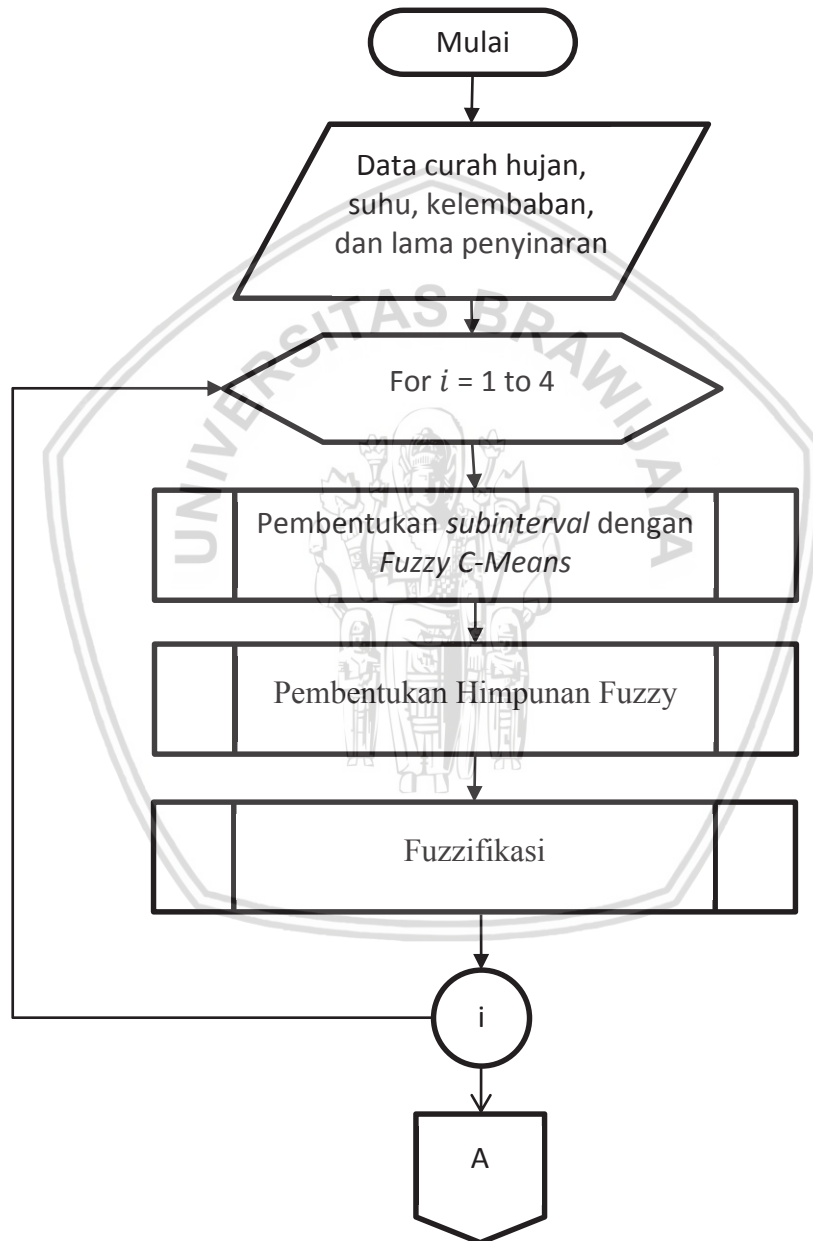
Penelitian ini menggunakan 4 *attribute* data yang mempengaruhi hasil prediksi curah hujan yaitu curah hujan, suhu, kelembaban udara, dan lama penyinaran matahari pada area Tengger (Sumber, Puspo, Tukur, dan Tosari). Empat *attribute* tersebut menjadi masukan untuk perangkat lunak pada implementasi penelitian ini. Contoh data masukan perangkat lunak pada penelitian ini dapat dilihat pada Tabel 4.1.

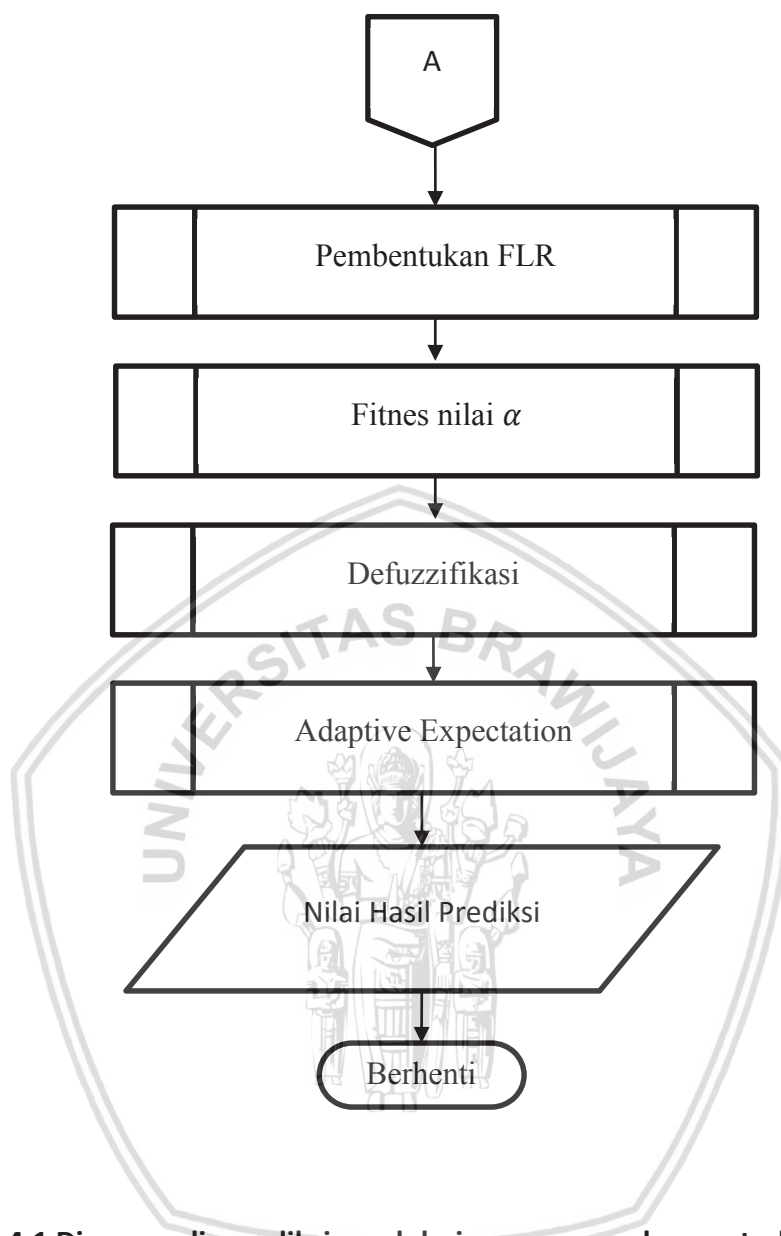
Tabel 4.1 Data sampel masukan

Dasarian/Bulan/Tahun	Curah hujan (A)	Suhu (B)	Kelembaban (C)	Lama Penyinaran (D)
1/1/2010	19.800	24.060	84.200	3.730
2/1/2010	6.400	24.556	77.000	3.611
3/1/2010	0.000	23.100	86.545	3.282
1/2/2010	19.700	23.670	85.700	3.440
2/2/2010	16.800	23.867	84.556	3.989
3/2/2010	11.625	24.450	81.375	5.575
1/3/2010	9.900	24.340	84.700	4.710
2/3/2010	18.300	24.278	80.556	5.456
3/3/2010	10.727	24.436	82.727	3.973
1/4/2010	9.100	23.950	85.200	4.470

4.2 Siklus algoritma *Adaptive-Expectation Based Multi-Attribute Fuzzy Time Series*

Penelitian ini menggunakan metode *Adaptive-Expectation Based Multi-Attribute Fuzzy Time Series* untuk memprediksi curah hujan. Metode *Multi-Attribute Fuzzy Time Series* digunakan untuk memperoleh hasil prediksi curah hujan, sedangkan metode *Adaptive-Expectation* digunakan untuk memperbaiki hasil prediksi curah hujan. Diagram alir (*flowchart*) dari proses prediksi perangkat lunak menggunakan metode ini ditunjukkan pada Gambar 4.2.





Gambar 4.1 Diagram alir prediksi curah hujan menggunakan metode *Adaptive-Expectation Based Multi-Attribute Fuzzy Time Series*.

Penjelasan dari diagram alir pada Gambar 4.1 adalah sebagai berikut:

1. Memasukkan data yang digunakan untuk prediksi dalam bentuk (.txt) ke dalam perangkat lunak, dan menginisialisasi jumlah data latih, jumlah data uji, bobot, dan iterasi maksimum. Sedangkan variable lain telah ditentukan didalam sistem yaitu error terkecil, dan jumlah kelas.
2. Pembentukan *subinterval* yang diawali dengan klusterisasi dengan menggunakan algoritma Fuzzy C-Means.
3. Pembentukan himpunan fuzzy berdasarkan subinterval yang telah terbentuk.

4. Fuzzifikasi data set berdasarkan derajat keanggotaan maksimum sebanyak 4 tahap sesuai *attribute* yang digunakan.
5. Pembentukan FLR menggunakan *Multi-Attribute Fuzzy Time Series*.
6. Melakukan fitness untuk mendapatkan nilai alpha terbaik.
7. Defuzzifikasi untuk mendapatkan hasil prediksi curah hujan.
8. Perbaiki populasi induk menjadi populasi baru dengan menggunakan *Adaptive-Expectation*.
9. Mendapatkan hasil prediksi curah hujan.

4.3 Siklus penyelesaian masalah Menggunakan Algoritma Adaptive-Expectation Based Multi-Attribute Fuzzy Time Series.

Data Sampel yang digunakan pada siklus penyelesaian masalah ini yaitu data dasarian pertama bulan Januari sampai dasarian pertama bulan April pada tahun 2010, data tersebut seperti ditunjukkan pada tabel 4.1. Data tersebut akan diolah menjadi hasil prediksi sementara menggunakan *multi-attribute fuzzy time series* yaitu meliputi proses pembentukan *subinterval*, pembentukan himpunan fuzzy, fuzzifikasi, pembentukan *Fuzzy Logic Relationship*, dan Defuzzifikasi. Hasil prediksi sementara tersebut kemudian diolah menggunakan *Adaptive-Expectation*. Dalam *Adaptive-Expectation* hasil prediksi tersebut digunakan sebagai *chromosome* dan dengan parameter tambahan α sebesar 0,1 dilakukan *fitness* sebanyak 100 kali. Nilai α akan berubah secara adaptive sebanyak 0,1 dalam setiap tahap *fitness*-nya. Nilai α terbaik yaitu memiliki nilai *Mean Square Error* terkecil akan disimpan. Nilai α tersebut akan menghasilkan hasil prediksi optimal menggunakan persamaan 2.8.

4.3.1 Pembentukan *subinterval* menggunakan Fuzzy C-Means

Dalam pembentukan *subinterval* menggunakan *Fuzzy C-Means* terdapat tiga tahap yaitu penentuan *Universe of Discourse* (U), penentuan pusat klaster dan penentuan batas *subinterval*.

4.3.1.1 Penentuan *Universe of Discourse* (U)

Dalam menentukan *universe of discourse* dibutuhkan beberapa proses yaitu variabel D_{min} , D_{max} , *mean*, dan standard deviasi. Nilai *mean* (rata-rata) dalam pengolahan data sampel curah hujan (A) dihitung dengan cara sebagai berikut.

$$\begin{aligned}
 \text{mean} &= \frac{(\text{data ke-1} + \text{data ke-2} + \dots + \text{data ke-10})}{10} \\
 &= \frac{(19.8 + 6.4 + 0 + 19.7 + 16.8 + 11.625 + 9.9 + 18.3 + 10.727 + 9.1)}{10} \\
 &= \frac{122.352}{10} \\
 &= 12,235
 \end{aligned}$$

Setelah mendapatkan nilai rata-rata, kemudian mencari standar deviasi. Untuk mempermudah dalam perhitungan, masing-masing data $ke-i$ ($i = 1, 2, \dots, jumlah\ data$) dikurangi rata-rata kemudian dikuadratkan. Perhitungan standar deviasi pada *attribute* A dapat dilihat pada Tabel 4.2.

Tabel 4.2 Nilai kuadrat dari selisih data $ke-i$ dengan rata-rata

Data ke-	data ke-i	(data ke-i)-(rata-rata)	((data ke-i) - (rata-rata)) ²
1	19.800	7.565	57.2262
2	6.400	-5.835	34.0496
3	0.000	-12.235	149.7001
4	19.700	7.465	55.7232
5	16.800	4.565	20.8374
6	11.625	-0.610	0.3723
7	9.900	-2.335	5.4532
8	18.300	6.065	36.7818
9	10.727	-1.508	2.2747
10	9.100	-3.135	9.8295
Jumlah			372.2480

Dengan menggunakan nilai jumlah pada Tabel 4.2, perhitungan standard deviasi dari setiap *attribute* sebagai berikut.

$$\begin{aligned}
 \text{Standard deviasi} &= \sqrt{\frac{1}{n-1} \sum_{i=1}^{10} (\text{data ke } i - \text{mean})^2} \\
 &= \sqrt{\frac{1}{10-1} \times 372.2480} \\
 &= 6.4312
 \end{aligned}$$

Perhitungan yang sama dilakukan juga pada *attribute* suhu (B), kelembaban (C), dan lama penyinaran (D). Nilai Dmin, Dmax, dan standard deviasi dari setiap *attribute* dapat dilihat pada Tabel 4.3



Tabel 4.3 Nilai maksimum, nilai minimum, dan standard deviasi

Attribute	Dmin	Dmax	standard deviasi
A	0.000	19.800	6.4312
B	23.100	24.556	0.4456
C	77.000	86.545	2.8960
D	3.282	5.575	0.8083

Langkah selanjutnya adalah menghitung nilai U dari *attribute* A sebagai berikut.

$$\begin{aligned}
 U &= [(D_{min} - \text{standard deviasi}), (D_{max} + \text{standard deviasi})] \\
 &= [(0.000 - 6.388), (19.800 + 6.388)] \\
 &= [(-6.388), (26.231)]
 \end{aligned}$$

Kemudian dengan melakukan perhitungan yang sama pada *attribute* B, C, dan D maka *Universe of Discourse* untuk setiap *attribute* dapat dilihat pada Tabel 4.4

Tabel 4.4 Universe of Discourse

Attribute	U	
	Batas Bawah	Batas Atas
A	-6.431	26.231
B	22.654	25.002
C	74.104	89.441
D	2.474	6.383

4.3.1.2 Penentuan pusat kluster

Proses penentuan pusat kluster dilakukan menggunakan algoritma *Fuzzy C-Means* yang terdiri dari beberapa tahap yaitu: pembentukan matriks partisi awal (u), normalisasi matriks partisi, dan perhitungan pusat kluster (V).

Variabel yang sudah ditentukan untuk melakukan proses penentuan pusat kluster sebagai berikut

- a. Banyak kluster yang akan dibentuk = 7
- b. Nilai bobot (w) = 2
- c. Iterasi awal = 1
- d. Iterasi maksimum = 3
- e. *Error* terkecil (ξ) = 0,01



Pembentukan matriks partisi awal (u)

Matriks partisi awal memiliki ukuran jumlah data (i) x jumlah kluster (k), sehingga matriks partisi awal untuk *attribute* A pada tabel 4.1 memiliki ukuran 10 x 7 yang berisi data acak atau *random*. Matriks partisi awal *attribute* A dapat dilihat pada Tabel 4.5.

Tabel 4.5 Matriks partisi awal

i	Cluster (u)						
	c1	c2	c3	c4	c5	c6	c7
1	214	68	189	135	205	70	215
2	142	224	238	120	84	68	15
3	76	34	133	47	238	21	5
4	58	32	63	125	215	87	219
5	44	167	165	204	229	191	137
6	68	154	68	17	142	70	180
7	237	163	148	11	96	134	100
8	181	20	12	83	253	127	162
9	178	137	168	32	146	107	202
10	228	108	207	234	130	1	174

Normalisasi matriks partisi

Normalisasi matriks partisi merupakan proses menormalisasi matriks partisi awal sehingga data random tersebut antara 0 sampai 1 dan jumlah dari setiap baris bernilai 1 sesuai dengan aturan persamaan 2.3. Langkah awal dalam melakukan normalisasi matriks partisi adalah mencari jumlah dari setiap baris pada matriks partisi awal.

$$Q_1 = 214 + 68 + 189 + 135 + 205 + 70 + 215 = 1096$$

$$Q_2 = 142 + 224 + 238 + 120 + 84 + 68 + 15 = 891$$

.....

$$Q_{10} = 228 + 108 + 207 + 234 + 130 + 1 + 174 = 1082$$

Langkah selanjutnya adalah mencari nilai dari setiap data dari matriks partisi yang sudah ternormalisasi (μ_{ik}) yaitu dengan membagi setiap datanya dengan jumlah data (Q_i) sesuai dengan baris datanya.

$$\mu_{11} = \frac{214}{1096} = 0.1952555$$



$$\mu_{12} = \frac{68}{1096} = 0.0620438$$

$$\mu_{17} = \frac{215}{1096} = 0.0620438$$

Sehingga $\sum_{i=1}^c \mu_{ik} = 0.1952555 + 0.0620438 + \dots + 0.0620438 = 1$

Perhitungan yang sama juga dilakukan untuk data berikutnya. Matriks partisi yang sudah dinormalisasi *attribute A* dapat dilihat pada Tabel 4.6.

Tabel 4.6 Matriks partisi ternormalisasi *attribute A*

<i>i</i>	μ						
	c1	c2	c3	c4	c5	c6	c7
1	0.1953	0.0620	0.1724	0.1232	0.1870	0.0639	0.1962
2	0.1594	0.2514	0.2671	0.1347	0.0943	0.0763	0.0168
3	0.1372	0.0614	0.2401	0.0848	0.4296	0.0379	0.0090
4	0.0726	0.0401	0.0788	0.1564	0.2691	0.1089	0.2741
5	0.0387	0.1469	0.1451	0.1794	0.2014	0.1680	0.1205
6	0.0973	0.2203	0.0973	0.0243	0.2031	0.1001	0.2575
7	0.2666	0.1834	0.1665	0.0124	0.1080	0.1507	0.1125
8	0.2160	0.0239	0.0143	0.0990	0.3019	0.1516	0.1933
9	0.1835	0.1412	0.1732	0.0330	0.1505	0.1103	0.2082
10	0.2107	0.0998	0.1913	0.2163	0.1201	0.0009	0.1608

Perhitungan Pusat klaster (V)

Perhitungan pusat klaster menggunakan Persamaan (2.4). Langkah pertama yaitu mencari nilai dari μ_{ik} pada Tabel 4.6 dipangkatkan dengan nilai bobot (*w*) kemudian dikalikan dengan nilai aktual pada *attribute A* Tabel 4.1. Hasil perkalian tersebut kemudian dijumlahkan setiap kolomnya kemudian dibagi dengan jumlah hasil kuadrat dari matriks partisi. Langkah tersebut dapat dilihat pada Tabel 4.7, dan Tabel 4.8.

Tabel 4.7 Kuadrat μ_{ik} dari *attribute A*

<i>i</i>	$(\mu_{ik})^w, w = 2$						
	c1	c2	c3	c4	c5	c6	c7
1	0.0381	0.0038	0.0297	0.0152	0.0350	0.0041	0.0385
2	0.0254	0.0632	0.0714	0.0181	0.0089	0.0058	0.0003
3	0.0188	0.0038	0.0576	0.0072	0.1846	0.0014	0.0001

Tabel 4.7 (lanjutan)

<i>i</i>	$(\mu_{ik})^w, w = 2$						
	c1	c2	c3	c4	c5	c6	c7
4	0.0053	0.0016	0.0062	0.0245	0.0724	0.0119	0.0751
5	0.0015	0.0216	0.0211	0.0322	0.0406	0.0282	0.0145
6	0.0095	0.0485	0.0095	0.0006	0.0413	0.0100	0.0663
7	0.0711	0.0336	0.0277	0.0002	0.0117	0.0227	0.0127
8	0.0467	0.0006	0.0002	0.0098	0.0911	0.0230	0.0374
9	0.0337	0.0199	0.0300	0.0011	0.0227	0.0122	0.0434
10	0.0444	0.0100	0.0366	0.0468	0.0144	0.0000	0.0259
Jumlah	0.2943	0.2066	0.2899	0.1555	0.5225	0.1193	0.3140

Tabel 4.8 Perhitungan pusat kluster *attribute A*

<i>i</i>	A_i	$(\mu_{ik})^w \times A_i$						
		c1	c2	c3	c4	c5	c6	c7
1	19.800	0.7549	0.0762	0.5888	0.3004	0.6927	0.0808	0.7619
2	6.400	0.1626	0.4045	0.4566	0.1161	0.0569	0.0373	0.0018
3	0.000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
4	19.700	0.1038	0.0316	0.1225	0.4822	1.4264	0.2336	1.4800
5	16.800	0.0252	0.3624	0.3538	0.5408	0.6815	0.4741	0.2439
6	11.625	0.1100	0.5643	0.1100	0.0069	0.4798	0.1166	0.7709
7	9.900	0.7036	0.3328	0.2744	0.0015	0.1154	0.2249	0.1253
8	18.300	0.8537	0.0104	0.0038	0.1795	1.6680	0.4203	0.6839
9	10.727	0.3612	0.2140	0.3218	0.0117	0.2430	0.1305	0.4652
10	9.100	0.4041	0.0907	0.3331	0.4256	0.1314	0.0000	0.2353
Jumlah		3.479	2.086	2.564	2.064	5.495	1.718	4.768
		0	9	7	7	1	1	2

Perhitungan pusat kluster attribute A dari V_1 dan V_2 menggunakan persamaan (2.4) dapat dilihat perhitungannya sebagai berikut:

$$V_1 = \frac{\sum_{i=1}^{10} ((\mu_{ik1})^2 \times X_i)}{\sum_{i=1}^{10} \mu_{ik1}^2} = \frac{3.4790}{0.2943} = 11.8184$$



$$V_2 = \frac{\sum_{i=1}^{10} ((\mu_{ik2})^2 \times X_i)}{\sum_{i=1}^{10} \mu_{ik2}^2} = \frac{2.0869}{0.2066} = 10.0995$$

Perhitungan serupa juga diterapkan untuk menghitung pusat kluster dari *attribute* B, C, dan D. Hasil perhitungan pusat kluster *attribute* A dapat dilihat pada Tabel 4.9.

Tabel 4.9 Pusat kluster *attribute* A

<i>Attribute</i>	c1	c2	c3	c4	c5	c6	c7
A	11.8184	10.0995	8.8444	13.2701	10.5155	14.4009	15.1827

Perhitungan fungsi objektif (P)

Setelah mendapatkan pusat kluster, kemudian menghitung fungsi objektif dengan menggunakan persamaan (2.5). Langkah pertama dalam menghitung fungsi objektif adalah dengan mengkuadratkan selisih dari data aktual (X_i) dikurangi pusat kluster (V_j). Hasil perhitungan dapat dilihat pada Tabel 4.10.

Tabel 4.10 Hasil $(X_i - V_j)^2$ dari *attribute* A

i	c1	c2	c3	c4	c5	c6	c7
1	63.7062	94.0998	120.0248	42.6397	86.2022	29.1499	21.3191
2	29.3589	13.6863	5.9752	47.1982	16.9372	64.0150	77.1365
3	139.6743	101.9998	78.2237	176.0954	110.5755	207.3870	230.5155
4	62.1198	92.1697	117.8437	41.3437	84.3553	28.0801	20.4057
5	24.8165	44.8967	63.2913	12.4602	39.4951	5.7555	2.6155
6	0.0374	2.3272	7.7316	2.7063	1.2310	7.7058	12.6575
7	3.6802	0.0398	1.1143	11.3575	0.3788	20.2584	27.9073
8	42.0113	67.2483	89.4080	25.2999	60.5987	15.2027	9.7173
9	1.1911	0.3938	3.5441	6.4673	0.0447	13.4978	19.8536
10	7.3896	0.9990	0.0653	17.3897	2.0036	28.0999	36.9997

Langkah selanjutnya menghitung nilai dari $(X_i - V_j)^2$ dikalikan dengan Data aktual X_{ik} ($i = \text{jumlah data}, k = \text{jumlah pusat kluster}$) pada *attribute* A, kemudian dijumlahkan untuk setiap kolomnya. Nilai Objektif didapatkan melalui jumlah dari jumlah setiap kolomnya. Hasil perhitungan tersebut dapat dilihat pada Tabel 4.11.

Tabel 4.11 Perhitungan nilai objektif *attribute* A

<i>i</i>	$(X_i - V_k)^2 \times X_{ij}$						
	c1	c2	c3	c4	c5	c6	c7
1	2.4288	0.3622	3.5692	0.6469	3.0158	0.1189	0.8204
2	0.7457	0.8650	0.4263	0.8561	0.1505	0.3729	0.0219
3	2.6286	0.3842	4.5084	1.2674	20.4077	0.2980	0.0188

Tabel 4.11 (lanjutan)

<i>i</i>	$(X_i - V_k)^2 \times X_{ij}$						
	c1	c2	c3	c4	c5	c6	c7
4	0.3273	0.1478	0.7326	1.0119	6.1080	0.3329	1.5330
5	0.0372	0.9686	1.3329	0.4011	1.6021	0.1624	0.0380
6	0.0004	0.1130	0.0732	0.0016	0.0508	0.0773	0.8393
7	0.2616	0.0013	0.0309	0.0017	0.0044	0.4603	0.3531
8	1.9599	0.0383	0.0183	0.2482	5.5235	0.3492	0.3632
9	0.0401	0.0079	0.1063	0.0070	0.0010	0.1642	0.8610
10	0.3281	0.0100	0.0024	0.8133	0.0289	0.0000	0.9568
Jumlah	8.7576	2.8982	10.8006	5.2554	36.8928	2.3361	5.8055
Fungsi Objektif	= 8.7576 + 2.8982 + 10.8006 + 5.2554 + 36.8928 + 2.3361 + 5.8055						
	=72.7461						

Perbaikan matriks partisi

Perbaikan matriks partisi dihitung dengan memangkatkan hasil pada Tabel 4.10 dengan $-1/1$ (karena $w-1=1$ ($w=2$)), kemudian dijumlahkan setiap barisnya untuk menghitung hasil perbaikan matriks partisi seperti ditunjukkan pada Tabel 4.12

Tabel 4.12 Perhitungan perbaikan matriks partisi attribute A

<i>i</i>	$((X_i - V_k)^2)^{-1}$							Jumlah
	c1	c2	c3	c4	c5	c6	c7	
1	0.0157	0.0106	0.0083	0.0235	0.0116	0.0343	0.0469	0.1509
2	0.0341	0.0731	0.1674	0.0212	0.0590	0.0156	0.0130	0.3833
3	0.0072	0.0098	0.0128	0.0057	0.0090	0.0048	0.0043	0.0536
4	0.0161	0.0108	0.0085	0.0242	0.0119	0.0356	0.0490	0.1561
5	0.0403	0.0223	0.0158	0.0803	0.0253	0.1737	0.3823	0.7400
6	26.7392	0.4297	0.1293	0.3695	0.8123	0.1298	0.0790	28.6888
7	0.2717	25.1264	0.8975	0.0880	2.6398	0.0494	0.0358	29.1085
8	0.0238	0.0149	0.0112	0.0395	0.0165	0.0658	0.1029	0.2746
9	0.8395	2.5396	0.2822	0.1546	22.3523	0.0741	0.0504	26.2927
10	0.1353	1.0010	15.3086	0.0575	0.4991	0.0356	0.0270	17.0642

Perhitungan selanjutnya yaitu dengan membagi setiap titik data pada Tabel 4.12 dengan jumlah setiap barisnya pada baris yang sama. Perhitungan dari Tabel 4.13 dapat dilihat sebagai berikut:

$$\mu_{11} = \frac{((X_1 - V_1)^2)^{-1}}{\sum_{k=1}^7 ((X_1 - V_k)^2)^{-1}} = \frac{0.0157}{0.1509} = 0.1040$$

$$\mu_{12} = \frac{((X_1 - V_2)^2)^{-1}}{\sum_{k=1}^7 ((X_1 - V_k)^2)^{-1}} = \frac{0.0106}{0.1509} = 0.0704$$

Perhitungan tersebut dilakukan untuk setiap elemen penyusun matriks partisi perbaikan. Hasil matriks perbaikan dapat dilihat pada Tabel 4.13.



Tabel 4.13 Matriks partisi perbaikan

<i>i</i>	μ_{ik}						
	c1	c2	c3	c4	c5	c6	c7
1	0.1040	0.0704	0.0552	0.1554	0.0769	0.2273	0.3108
2	0.0889	0.1906	0.4366	0.0553	0.1540	0.0408	0.0338
3	0.1335	0.1828	0.2384	0.1059	0.1686	0.0899	0.0809
4	0.1031	0.0695	0.0544	0.1550	0.0759	0.2281	0.3140
5	0.0545	0.0301	0.0214	0.1085	0.0342	0.2348	0.5166
6	0.9320	0.0150	0.0045	0.0129	0.0283	0.0045	0.0028
7	0.0093	0.8632	0.0308	0.0030	0.0907	0.0017	0.0012
8	0.0867	0.0542	0.0407	0.1440	0.0601	0.2396	0.3748
9	0.030 4	0.0048	0.0117	0.0022	0.2719	0.0015	0.6775
10	0.313 1	0.0292	0.5928	0.0068	0.0379	0.0041	0.0161

Langkah selanjutnya yaitu mengecek kondisi berhenti, iterasi akan berhenti apabila nilai objektif (*t*) dikurangi nilai objektif (*t-1*) lebih kecil dari nilai error terkecil (ξ) atau jumlah iterasi sudah mencapai iterasi maksimal yang telah ditentukan. Jika kondisi belum berhenti, maka akan diulang proses mulai dari Tabel 4.6 sampai Tabel 4.13 dengan catatan nilai tiap data pada matriks partisi ternormalisasi (*t+1*) pada Tabel 4.6 diganti dengan matriks partisi perbaikan (*t*) pada Tabel 4.13. Setelah mencapai kondisi berhenti, urutkan pusat kluster dari yang terkecil ke terbesar maka akan didapatkan Pusat Kluster dari setiap *attribute* seperti pada Tabel 4.14.

Tabel 4.14 Pusat kluster dari semua *attribute*

Pusat kluster	Attribute			
	A	B	C	D
c1	5.3988	23.5958	80.6509	3.5971
c2	8.8597	23.8894	80.6679	3.8893
c3	10.1051	23.9458	82.0039	3.9246
c4	11.3725	24.0199	82.3896	4.5347
c5	16.7926	24.0827	84.1763	5.2245
c6	18.9222	24.2777	84.6811	5.3521
c7	18.9567	24.4275	85.3017	5.3546



4.3.1.3 Penentuan batas subinterval

Batas subinterval terdiri dari 2 bagian yaitu batas bawah dan batas atas. Untuk batas bawah subinterval pertama menggunakan batas bawah *Universe of Discourse*, dan batas atas subinterval terakhir menggunakan batas atas *Universe of Discourse*. Selain dua nilai batas tersebut, nilai batas ditentukan dari nilai tengah diantara 2 pusat klaster berdekatan. Penentuan batas subinterval pada *attribute A* (Tabel 4.15) sebagai berikut:

Subinterval 1

Batas bawah = -6.4312 (batas bawah *Universe of Discourse*)

Batas atas = nilai tengah antara pusat klaster ke-1 dan ke-2 yaitu 7.1293

Subinterval 2

Batas bawah = nilai tengah antara pusat klaster ke-1 dan ke-2 yaitu 7.1293

Batas atas = nilai tengah antara pusat klaster ke-2 dan ke-3 yaitu 9.4824

Subinterval 3

Batas bawah = nilai tengah antara pusat klaster ke-2 dan ke-3 yaitu 9.4824

Batas atas = nilai tengah antara pusat klaster ke-3 dan ke-4 yaitu 10.7388

Subinterval 4

Batas bawah = nilai tengah antara pusat klaster ke-3 dan ke-4 yaitu 10.7388

Batas atas = nilai tengah antara pusat klaster ke-4 dan ke-5 yaitu 14.0825

Subinterval 5

Batas bawah = nilai tengah antara pusat klaster ke-4 dan ke-5 yaitu 14.0825

Batas atas = nilai tengah antara pusat klaster ke-5 dan ke-6 yaitu 17.8574

Subinterval 6

Batas bawah = nilai tengah antara pusat klaster ke-5 dan ke-6 yaitu 17.8574

Batas atas = nilai tengah antara pusat klaster ke-6 dan ke-7 yaitu 18.9395

Subinterval 7

Batas bawah = nilai tengah antara pusat klaster ke-6 dan ke-7 yaitu 18.9395

Batas atas = 26.2312 (batas atas *Universe of Discourse*)

Penentuan batas subinterval serupa juga dilakukan pada *attribute* B, C, dan D. Batas subinterval untuk semua *attribute* dapat dilihat pada Tabel 4.15.

Tabel 4.15 Penentuan batas subinterval

Sub interval	Batas	Attribute data			
		A	B	C	D
u1	bawah	-6.4312	22.6544	74.1040	2.4737
	atas	7.1293	23.7426	80.6594	3.7432
u2	bawah	7.1293	23.7426	80.6594	3.7432
	atas	9.4824	23.9176	81.3359	3.9070
u3	bawah	9.4824	23.9176	81.3359	3.9070
	atas	10.7388	23.9828	82.1967	4.2297
u4	bawah	10.7388	23.9828	82.1967	4.2297
	atas	14.0825	24.0513	83.2829	4.8796
u5	bawah	14.0825	24.0513	83.2829	4.8796
	atas	17.8574	24.1802	84.4287	5.2883
u6	bawah	17.8574	24.1802	84.4287	5.2883
	atas	18.9395	24.3526	84.9914	5.3534
u7	bawah	18.9395	24.3526	84.9914	5.3534
	atas	26.2312	25.0016	89.4410	6.3833

4.3.2 Pembentukan Himpunan Fuzzy

Himpunan fuzzy terbentuk berdasarkan subinterval yang telah terbentuk pada proses sebelumnya. Himpunan fuzzy dari *attribute* A (Tabel 4.16) dihitung menggunakan persamaan sebagai berikut:

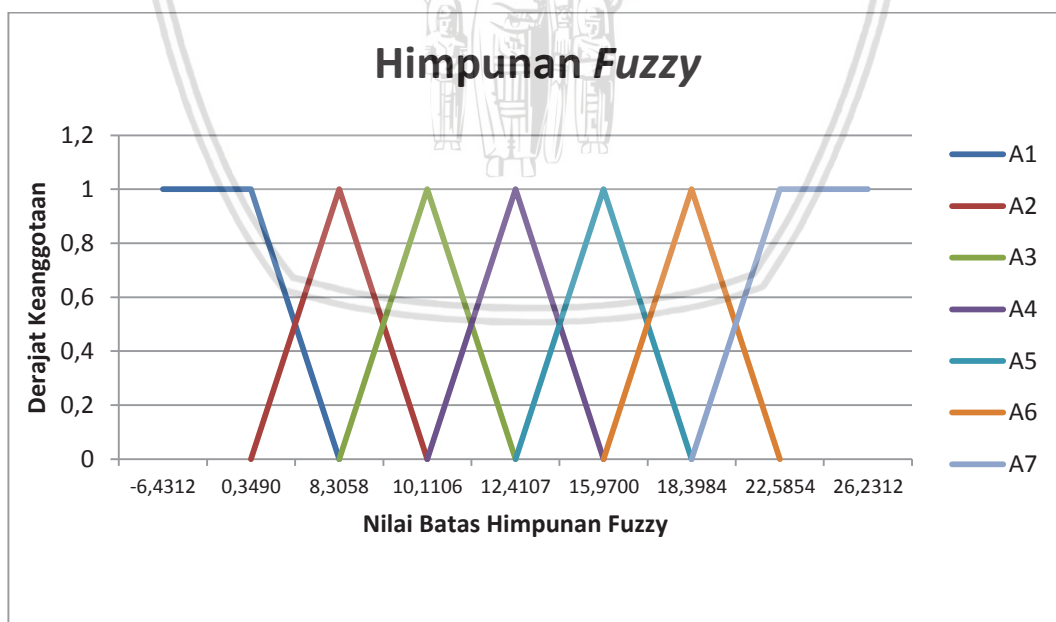
$$\begin{aligned}
 A_1 &= \frac{1}{u_1} + \frac{0.5}{u_2} + \frac{0}{u_3} + \frac{0}{u_4} + \frac{0}{u_5} + \frac{0}{u_6} + \frac{0}{u_7} \\
 A_2 &= \frac{0.5}{u_1} + \frac{1}{u_2} + \frac{0.5}{u_3} + \frac{0}{u_4} + \frac{0}{u_5} + \frac{0}{u_6} + \frac{0}{u_7} \\
 A_3 &= \frac{0}{u_1} + \frac{0}{u_2} + \frac{1}{u_3} + \frac{0.5}{u_4} + \frac{0}{u_5} + \frac{0}{u_6} + \frac{0}{u_7} \\
 A_4 &= \frac{0}{u_1} + \frac{0}{u_2} + \frac{0}{u_3} + \frac{0.5}{u_4} + \frac{1}{u_5} + \frac{0.5}{u_6} + \frac{0}{u_7} \\
 A_5 &= \frac{0}{u_1} + \frac{0}{u_2} + \frac{0}{u_3} + \frac{0}{u_4} + \frac{0.5}{u_5} + \frac{1}{u_6} + \frac{0.5}{u_7} \\
 A_6 &= \frac{0}{u_1} + \frac{0}{u_2} + \frac{0}{u_3} + \frac{0}{u_4} + \frac{0}{u_5} + \frac{0.5}{u_6} + \frac{1}{u_7} \\
 A_7 &= \frac{0}{u_1} + \frac{0}{u_2} + \frac{0}{u_3} + \frac{0}{u_4} + \frac{0}{u_5} + \frac{0}{u_6} + \frac{1}{u_7}
 \end{aligned}$$

Operator himpunan gabungan ditunjukkan menggunakan simbol “+” dan u_i merupakan subinterval ($i = 1, 2, 3, \dots, \text{jumlah kelas}$). Himpunan fuzzy A_1 terbentuk antara batas bawah u_1 yaitu -1.8882 dan nilai tengah dari u_2 ($\frac{0.5}{u_2}$) dengan nilai 6.7820 dari hasil jumlah kedua batas u_2 dibagi dua. Himpunan fuzzy A_2 terbentuk antara nilai tengah dari u_1 ($\frac{0.5}{u_1}$) yaitu 2.0395 (hasil jumlah kedua batas u_1 dibagi dua) dan nilai tengah dari u_3 ($\frac{0.5}{u_3}$) dengan nilai 8.6337 (hasil jumlah kedua batas u_3 dibagi dua). Begitu seterusnya menggunakan perulangan, misalkan himpunan (A_i) terbentuk antara nilai tengah u_{i-1} yaitu (jumlah kedua batas u_{i-1} dibagi dua) dan antara nilai tengah u_{i+1} yaitu (jumlah kedua batas u_{i+1} dibagi dua). Untuk himpunan fuzzy terakhir A_n terbentuk antara nilai tengah u_{n-1} dan batas atas subinterval (u_n). Hasil batas subinterval dari setiap himpunan fuzzy pada *attribute* A dapat dilihat pada Tabel 4.16.



Tabel 4.16 Himpunan Fuzzy *attribute A*

Himpunan Fuzzy	Batas	Nilai
A1	bawah	-6.4312
	atas	8.3058
A2	bawah	0.3490
	atas	10.1106
A3	bawah	8.3058
	atas	12.4107
A4	bawah	10.1106
	atas	15.9700
A5	bawah	12.4107
	atas	18.3984
A6	bawah	15.9700
	atas	22.5854
A7	bawah	18.3984
	atas	26.2312



Gambar 4.2 Grafik himpunan fuzzy *attribute A*

4.3.3 Fuzzifikasi

Fuzzifikasi merupakan proses untuk mengubah data asli menjadi nilai *fuzzy* sesuai dengan nilai derajat keanggotaan tertinggi. Perhitungan derajat keanggotaan data curah hujan (A) pada baris pertama Tabel 4.1 yaitu sebagai berikut

$$\begin{aligned}
 \text{Curah hujan (A) } X_1 &= 19.8000 \\
 \text{Derajat keanggotaan } A_1 &= 0 \\
 \text{Derajat keanggotaan } A_2 &= 0 \\
 \text{Derajat keanggotaan } A_3 &= 0 \\
 \text{Derajat keanggotaan } A_4 &= 0 \\
 \text{Derajat keanggotaan } A_5 &= 0 \\
 \text{Derajat keanggotaan } A_6 &= \frac{(22.5853-19.800)}{(22.5853-18.3984)} = 0.6653 \\
 \text{Derajat keanggotaan } A_7 &= \frac{(19.800-18.3984)}{(22.5853-18.3984)} = 0.3347
 \end{aligned}$$

Dari hasil perhitungan tersebut, data X_1 pada *attribute* curah hujan (A) termasuk ke dalam himpunan fuzzy A_6 , karena nilai derajat keanggotaan A_6 senilai 0.6653 paling besar dibandingkan dengan derajat keanggotaan lainnya. Perhitungan derajat keanggotaan tersebut juga digunakan untuk menghitung data selanjutnya (X_i). Adapun hasil perhitungan semua data sampel pada *attribute* A dapat dilihat pada Tabel 4.17.

Tabel 4.17 Derajat keanggotaan *attribute* A

No	X(i)	A1	A2	A3	A4	A5	A6	A7	Hasil Fuzzy
1	19.800	0.000	0.000	0.000	0.000	0.000	0.665	0.335	A6
2	6.400	0.240	0.760	0.000	0.000	0.000	0.000	0.000	A2
3	0.000	1.000	0.000	0.000	0.000	0.000	0.000	0.000	A1
4	19.700	0.000	0.000	0.000	0.000	0.000	0.689	0.311	A6
5	16.800	0.000	0.000	0.000	0.000	0.658	0.342	0.000	A5
6	11.625	0.000	0.000	0.342	0.658	0.000	0.000	0.000	A4
7	9.900	0.000	0.117	0.883	0.000	0.000	0.000	0.000	A3
8	18.300	0.000	0.000	0.000	0.000	0.041	0.959	0.000	A6
9	10.727	0.000	0.000	0.732	0.268	0.000	0.000	0.000	A3
10	9.100	0.000	0.560	0.440	0.000	0.000	0.000	0.000	A2



Perhitungan nilai derajat keanggotaan pada *attribute* B, C, dan D sama dengan perhitungan pada Tabel 4.17. Hasil fuzzifikasi dari semua *attribute* dapat dilihat pada Tabel 4.18.

Tabel 4.18 Hasil fuzzifikasi semua *attribute* data

Data ke- <i>i</i>	A	B	C	D
1	A6	B4	C5	D2
2	A2	B7	C1	D2
3	A1	B1	C7	D1
4	A6	B2	C6	D1
5	A5	B2	C6	D3
6	A4	B6	C2	D6
7	A3	B6	C6	D4
8	A6	B6	C2	D6
9	A3	B6	C4	D3
10	A2	B3	C6	D4

4.3.4 Pembentukan Fuzzy Logic Relationship (FLR)

Pembentukan FLR berdasarkan data latih/*training* dengan runtun waktu tertentu (order). Misalkan order yang digunakan adalah tiga, maka dari 10 data pada Tabel 4.18 akan terbentuk 6 FLR (untuk uji 1 menggunakan data ke-9) dan 7 FLR (untuk data uji 2 menggunakan data ke-10) karena 2 data awal akan digunakan untuk membentuk FLR yang pertama (F_1). Berdasarkan hasil fuzzifikasi pada Tabel 4.18, Hasil pembentukan FLR dari 8 data latih dengan 3 order ditunjukkan pada Tabel 4.19.

Tabel 4.19 Fuzzy Logic Relationship

No	FLR	
	kiri	kanan
1	$\{ (B_{((t-2),4)}, C_{((t-2),5)}, D_{((t-2),2)}),$ $(B_{((t-1),7)}, C_{((t-1),1)}, D_{((t-1),2)}),$ $(B_{((t),1)}, C_{((t),7)}, D_{((t),1)}) \}$	$\rightarrow F_1 = A_1$
2	$\{ (B_{((t-2),7)}, C_{((t-2),1)}, D_{((t-2),2)}),$ $(B_{((t-1),1)}, C_{((t-1),7)}, D_{((t-1),1)}),$ $(B_{((t),2)}, C_{((t),6)}, D_{((t),1)}) \}$	$\rightarrow F_2 = A_6$
3	$\{ (B_{((t-2),1)}, C_{((t-2),7)}, D_{((t-2),1)}),$ $(B_{((t-1),2)}, C_{((t-1),6)}, D_{((t-1),1)}),$ $(B_{((t),2)}, C_{((t),6)}, D_{((t),3)}) \}$	$\rightarrow F_3 = A_5$
4	$\{ (B_{((t-2),2)}, C_{((t-2),6)}, D_{((t-2),1)}),$ $(B_{((t-1),2)}, C_{((t-1),6)}, D_{((t-1),3)}),$ $(B_{((t),6)}, C_{((t),2)}, D_{((t),6)}) \}$	$\rightarrow F_4 = A_4$
5	$\{ (B_{((t-2),2)}, C_{((t-2),6)}, D_{((t-2),3)}),$ $(B_{((t-1),6)}, C_{((t-1),2)}, D_{((t-1),6)}),$ $(B_{((t),6)}, C_{((t),6)}, D_{((t),4)}) \}$	$\rightarrow F_5 = A_3$
6	$\{ (B_{((t-2),6)}, C_{((t-2),2)}, D_{((t-2),6)}),$ $(B_{((t-1),6)}, C_{((t-1),6)}, D_{((t-1),4)}),$ $(B_{((t),6)}, C_{((t),2)}, D_{((t),6)}) \}$	$\rightarrow F_6 = A_6$
7	$\{ (B_{((t-2),6)}, C_{((t-2),6)}, D_{((t-2),4)}),$ $(B_{((t-1),6)}, C_{((t-1),2)}, D_{((t-1),6)}),$ $(B_{((t),6)}, C_{((t),4)}, D_{((t),3)}) \}$	$\rightarrow F_7 = A_3$



Dari Tabel 4.19 untuk pengujian pertama menggunakan F_1 sampai F_5 sebagai FLR latih dan F_6 sebagai FLR uji untuk memprediksi data periode 9. Sedangkan untuk pengujian kedua menggunakan F_1 sampai F_6 sebagai FLR latih dan F_7 sebagai FLR uji untuk memprediksi data periode 10.

Dari Tabel 4.19 FLR kanan dikombinasikan dengan FLR selanjutnya menggunakan aturan $F_i \rightarrow F_j \forall i \& j = 1, 2, \dots, k$. Kombinasi FLR kanan tersebut dapat dilihat sebagai berikut

$$F_1 = A_1 \rightarrow A_6$$

$$F_2 = A_6 \rightarrow A_5$$

$$F_3 = A_5 \rightarrow A_4$$

$$F_4 = A_4 \rightarrow A_3$$

$$F_5 = A_3 \rightarrow A_6$$

$$F_6 = A_6 \rightarrow A_3$$

kemudian dikelompokkan sesuai ruas kiri yang sama. Pengelompokan kombinasi FLR kanan dapat dilihat sebagai berikut

$$A_1 \rightarrow A_6$$

$$A_2 \rightarrow$$

$$A_3 \rightarrow A_6$$

$$A_4 \rightarrow A_3$$

$$A_5 \rightarrow A_4$$

$$A_6 \rightarrow A_3, A_5$$

Dari pengelompokan tersebut untuk pengujian pertama menggunakan A_1 sampai A_5 sebagai perhitungan frekuensi untuk memprediksi data periode 9. Sedangkan untuk pengujian kedua menggunakan A_1 sampai A_6 sebagai perhitungan frekuensi untuk memprediksi data periode 10.

4.3.5 Fitnes nilai α

Dari kombinasi FLR yang terbentuk dilakukan prediksi data latih untuk mencari nilai α , perhitungan defuzzifikasi dijelaskan pada subbab 4.3.6 Defizzifikasi. Hasil defuzzifikasi FLR latih sebagai berikut

Tabel 4.20 Hasil prediksi data latih

Dasarian/Bulan/Tahun	Data Aktual	Hasil Prediksi
1/1/2010	19.800	-
2/1/2010	6.400	-
3/1/2010	0.000	-
1/2/2010	19.700	5.3988



Tabel 4.20 (lanjutan)

Dasarian/Bulan/Tahun	Data Aktual	Hasil Prediksi
2/2/2010	16.800	18.9222
3/2/2010	11.625	16.7926
1/3/2010	9.900	11.3725
2/3/2010	18.300	10.1051

Dengan menggunakan data perbandingan pada Tabel 4.20, *Adaptive Expectation* mencari nilai parameter α terbaik yang memiliki nilai MSE terkecil dari 99 tahap perulangan sesuai dengan persamaan 2.6. Perhitungan dalam pembentukan generasi baru (prediksi baru) pada tahap pertama sebagai berikut.

$$\begin{aligned} \text{Prediksi (1/2/2010)} &= 0.000 + (0.01 \times (5.3988 - 0.000)) \\ &= 0.0539 \end{aligned}$$

$$\begin{aligned} \text{Prediksi (2/2/2010)} &= 19.700 + (0.01 \times (18.9222 - 10.700)) \\ &= 19.692 \end{aligned}$$

.....

.....

$$\begin{aligned} \text{Prediksi (2/3/2010)} &= 9.900 + (0.01 \times (10.1051 - 9.900)) \\ &= 9.902 \end{aligned}$$

Langkah selanjutnya menghitung nilai MSE pada setiap tahapnya. Contoh perhitungan MSE pada tahap pertama sebagai berikut.

$$\begin{aligned} \text{MSE} &= \frac{|19.700 - 0.0539|^2 + |16.800 - 19.693|^2 + \dots + |18.300 - 9.902|^2}{5} \\ &= 98.9206 \end{aligned}$$

Kedua tahap perhitungan tersebut dilakukan juga pada tahap-tahap berikutnya. Hasil dari kedua perhitungan tersebut pada 99 tahap dapat dilihat pada Tabel 4.21.

Tabel 4.21 Fitness α

α	1/2/2010	2/2/2010	3/2/2010	1/3/2010	2/3/2010	MSE
0.01	0.053988	19.692	16.800	11.622	9.902	98.92061
0.02	0.107977	19.684	16.800	11.620	9.904	98.47916
0.03	0.161965	19.677	16.800	11.617	9.906	98.03891
0.04	0.215953	19.669	16.800	11.615	9.908	97.59986
0.05	0.269942	19.661	16.800	11.612	9.910	97.16199
0.06	0.32393	19.653	16.800	11.610	9.912	96.72533
0.07	0.377918	19.646	16.799	11.607	9.914	96.28985



Tabel 4.21 (lanjutan)

α	1/2/2010	2/2/2010	3/2/2010	1/3/2010	2/3/2010	MSE
0.08	0.431907	19.638	16.799	11.605	9.916	95.85557
0.09	0.485895	19.630	16.799	11.602	9.918	95.42249
0.1	0.539883	19.622	16.799	11.600	9.921	94.9906
0.11	0.593872	19.614	16.799	11.597	9.923	94.5599
0.12	0.64786	19.607	16.799	11.595	9.925	94.1304
0.13	0.701848	19.599	16.799	11.592	9.927	93.70209
0.14	0.755837	19.591	16.799	11.590	9.929	93.27497
0.15	0.809825	19.583	16.799	11.587	9.931	92.84905
0.16	0.863813	19.576	16.799	11.585	9.933	92.42433
0.17	0.917802	19.568	16.799	11.582	9.935	92.0008
0.18	0.97179	19.560	16.799	11.580	9.937	91.57846
0.19	1.025778	19.552	16.799	11.577	9.939	91.15732
0.2	1.079767	19.544	16.799	11.574	9.941	90.73737
0.21	1.133755	19.537	16.798	11.572	9.943	90.31862
0.22	1.187743	19.529	16.798	11.569	9.945	89.90106
0.23	1.241732	19.521	16.798	11.567	9.947	89.4847
0.24	1.29572	19.513	16.798	11.564	9.949	89.06953
0.25	1.349708	19.506	16.798	11.562	9.951	88.65555
0.26	1.403697	19.498	16.798	11.559	9.953	88.24277
0.27	1.457685	19.490	16.798	11.557	9.955	87.83118
0.28	1.511673	19.482	16.798	11.554	9.957	87.42079
0.29	1.565662	19.474	16.798	11.552	9.959	87.01159
0.3	1.61965	19.467	16.798	11.549	9.962	86.60358
0.31	1.673639	19.459	16.798	11.547	9.964	86.19677
0.32	1.727627	19.451	16.798	11.544	9.966	85.79116
0.33	1.781615	19.443	16.798	11.542	9.968	85.38674
0.34	1.835604	19.436	16.797	11.539	9.970	84.98351
0.35	1.889592	19.428	16.797	11.537	9.972	84.58148
0.36	1.94358	19.420	16.797	11.534	9.974	84.18064
0.37	1.997569	19.412	16.797	11.532	9.976	83.781
0.38	2.051557	19.404	16.797	11.529	9.978	83.38255
0.39	2.105545	19.397	16.797	11.527	9.980	82.98529



Tabel 4.21 (lanjutan)

α	1/2/2010	2/2/2010	3/2/2010	1/3/2010	2/3/2010	MSE
0.4	2.159534	19.389	16.797	11.524	9.982	82.58923
0.41	2.213522	19.381	16.797	11.521	9.984	82.19436
0.42	2.26751	19.373	16.797	11.519	9.986	81.80069
0.43	2.321499	19.366	16.797	11.516	9.988	81.40821
0.44	2.375487	19.358	16.797	11.514	9.990	81.01693
0.45	2.429475	19.350	16.797	11.511	9.992	80.62684
0.46	2.483464	19.342	16.797	11.509	9.994	80.23794
0.47	2.537452	19.334	16.797	11.506	9.996	79.85024
0.48	2.59144	19.327	16.796	11.504	9.998	79.46374
0.49	2.645429	19.319	16.796	11.501	10.000	79.07842
0.5	2.699417	19.311	16.796	11.499	10.003	78.69431
0.51	2.753405	19.303	16.796	11.496	10.005	78.31138
0.52	2.807394	19.296	16.796	11.494	10.007	77.92965
0.53	2.861382	19.288	16.796	11.491	10.009	77.54912
0.54	2.91537	19.280	16.796	11.489	10.011	77.16978
0.55	2.969359	19.272	16.796	11.486	10.013	76.79163
0.56	3.023347	19.264	16.796	11.484	10.015	76.41468
0.57	3.077335	19.257	16.796	11.481	10.017	76.03892
0.58	3.131324	19.249	16.796	11.479	10.019	75.66436
0.59	3.185312	19.241	16.796	11.476	10.021	75.29099
0.6	3.2393	19.233	16.796	11.473	10.023	74.91882
0.61	3.293289	19.226	16.795	11.471	10.025	74.54784
0.62	3.347277	19.218	16.795	11.468	10.027	74.17805
0.63	3.401265	19.210	16.795	11.466	10.029	73.80946
0.64	3.455254	19.202	16.795	11.463	10.031	73.44206
0.65	3.509242	19.194	16.795	11.461	10.033	73.07586
0.66	3.56323	19.187	16.795	11.458	10.035	72.71085
0.67	3.617219	19.179	16.795	11.456	10.037	72.34704
0.68	3.671207	19.171	16.795	11.453	10.039	71.98442
0.69	3.725195	19.163	16.795	11.451	10.042	71.62299
0.7	3.779184	19.156	16.795	11.448	10.044	71.26276
0.71	3.833172	19.148	16.795	11.446	10.046	70.90372



Tabel 4.21 (lanjutan)

α	1/2/2010	2/2/2010	3/2/2010	1/3/2010	2/3/2010	MSE
0.72	3.88716	19.140	16.795	11.443	10.048	70.54588
0.73	3.941149	19.132	16.795	11.441	10.050	70.18923
0.74	3.995137	19.124	16.795	11.438	10.052	69.83378
0.75	4.049125	19.117	16.794	11.436	10.054	69.47952
0.76	4.103114	19.109	16.794	11.433	10.056	69.12645
0.77	4.157102	19.101	16.794	11.431	10.058	68.77458
0.78	4.21109	19.093	16.794	11.428	10.060	68.42391
0.79	4.265079	19.086	16.794	11.426	10.062	68.07443
0.8	4.319067	19.078	16.794	11.423	10.064	67.72614
0.81	4.373055	19.070	16.794	11.420	10.066	67.37904
0.82	4.427044	19.062	16.794	11.418	10.068	67.03315
0.83	4.481032	19.054	16.794	11.415	10.070	66.68844
0.84	4.53502	19.047	16.794	11.413	10.072	66.34493
0.85	4.589009	19.039	16.794	11.410	10.074	66.00261
0.86	4.642997	19.031	16.794	11.408	10.076	65.66149
0.87	4.696985	19.023	16.794	11.405	10.078	65.32156
0.88	4.750974	19.016	16.793	11.403	10.080	64.98283
0.89	4.804962	19.008	16.793	11.400	10.083	64.64529
0.9	4.858951	19.000	16.793	11.398	10.085	64.30895
0.91	4.912939	18.992	16.793	11.395	10.087	63.9738
0.92	4.966927	18.984	16.793	11.393	10.089	63.63984
0.93	5.020916	18.977	16.793	11.390	10.091	63.30708
0.94	5.074904	18.969	16.793	11.388	10.093	62.97551
0.95	5.128892	18.961	16.793	11.385	10.095	62.64514
0.96	5.182881	18.953	16.793	11.383	10.097	62.31596
0.97	5.236869	18.946	16.793	11.380	10.099	61.98798
0.98	5.290857	18.938	16.793	11.378	10.101	61.66119
0.99	5.344846	18.930	16.793	11.375	10.103	61.33559

Dari Tabel 4.21 telah didapatkan nilai α terbaik senilai (0.99) dengan MSE terkecil yaitu (61.3355). Nilai α yang didapatkan disimpan dan akan digunakan untuk memperbaiki hasil defuzzifikasi pada data uji.



4.3.6 Defuzzifikasi

Defuzzifikasi dilakukan dengan mencari selisih atribut pendukung (di ruas kiri pada tabel FLR) data uji dengan FLR. Perhitungan selisih nilai *attribute* pendukung FLR dengan data uji dilakukan untuk menemukan FLR yang paling cocok dengan data uji. FLR yang paling cocok akan disimpan berdasarkan atribut utama (di ruas kanan pada tabel FLR).

Dari hasil fuzzifikasi data pada Tabel 4.18 akan dibagi menjadi 2 tipe data yaitu data latih dan data uji, dengan data latih pada periode ke-1 sampai ke-8 sedangkan data uji pada periode ke-9 dan ke-10. Berikut merupakan contoh perhitungan selisih atribut pendukung data uji pertama (periode ke-9).

Nilai pangkat bawah atribut pendukung periode ke-9 pada Tabel 4.19 yaitu F_6 (FLR sebelum data uji) sebagai data uji pertama adalah sebagai berikut:

$$F_6 = (6, 2, 6), (6, 6, 4), (6, 2, 6)$$

Nilai pangkat bawah atribut pendukung periode ke-1 sampai ke-8 Tabel 4.18 dan F_1 sampai F_5 Tabel 4.19 sebagai data latih adalah sebagai berikut:

$$F_1 = (4, 5, 2), (7, 1, 2), (1, 7, 1)$$

$$F_2 = (7, 1, 2), (1, 7, 1), (2, 6, 1)$$

$$F_3 = (1, 7, 1), (2, 6, 1), (2, 6, 3)$$

$$F_4 = (2, 6, 1), (2, 6, 3), (6, 2, 6)$$

$$F_5 = (2, 6, 3), (6, 2, 6), (6, 6, 4)$$

Selisih atribut pendukung bernilai absolut dari selisih atribut pendukung data uji terhadap setiap data latih. Adapun cara menghitungnya sebagai berikut:

1. Selisih atribut pendukung F_6 (data uji) dan F_1 (data latih)

$$\begin{aligned} &= |6 - 4| + |2 - 5| + |6 - 2| \\ &+ |6 - 7| + |6 - 1| + |4 - 2| \\ &+ |6 - 1| + |2 - 7| + |6 - 1| \\ &= 32 \end{aligned}$$

2. Selisih atribut pendukung F_6 (data uji) dan F_2 (data latih)

$$\begin{aligned} &= |6 - 7| + |2 - 1| + |6 - 2| \\ &+ |6 - 1| + |6 - 7| + |4 - 1| \\ &+ |6 - 2| + |2 - 6| + |6 - 1| \\ &= 28 \end{aligned}$$

3. Selisih atribut pendukung F_6 (data uji) dan F_3 (data latih)

$$\begin{aligned} &= |6 - 1| + |2 - 7| + |6 - 1| \\ &+ |6 - 2| + |6 - 6| + |4 - 1| \end{aligned}$$



$$\begin{aligned}
 &+|6 - 2| + |2 - 6| + |6 - 3| \\
 &= 33
 \end{aligned}$$

4. Selisih atribut pendukung F_6 (data uji) dan F_4 (data latih)

$$\begin{aligned}
 &= |6 - 2| + |2 - 6| + |6 - 1| \\
 &+|6 - 2| + |6 - 6| + |4 - 3| \\
 &+|6 - 6| + |2 - 2| + |6 - 6| \\
 &= 18
 \end{aligned}$$

5. Selisih atribut pendukung F_6 (data uji) dan F_5 (data latih)

$$\begin{aligned}
 &= |6 - 2| + |2 - 6| + |6 - 3| \\
 &+|6 - 6| + |6 - 2| + |4 - 6| \\
 &+|6 - 6| + |2 - 6| + |6 - 4| \\
 &= 23
 \end{aligned}$$

FLR yang paling cocok dengan data uji adalah dengan selisih antar *attribute* pendukung paling sedikit, sehingga dari hasil perhitungan tersebut dapat disimpulkan bahwa FLR yang paling cocok untuk data uji pertama adalah F_4 dengan selisih terkecil senilai 18. Perhitungan yang sama juga dilakukan untuk data uji kedua dengan F_1 sampai F_6 sebagai FLR latih, dan F_7 sebagai FLR uji. Data uji kedua cocok dengan FLR F_6 dengan nilai selisih 5. Langkah selanjutnya menghitung frekuensi himpunan *fuzzy* yang tergabung dalam FLR terpilih terhadap setiap pusat kluster atribut utama pada data uji. Dari Tabel 4.19 dan aturan pengelompokkan, FLR terpilih dapat diuraikan pada atribut utama sebagai berikut

$$\text{Data uji ke-1 } F_4 = A_4 \rightarrow A_3$$

$$\text{Data uji ke-2 } F_6 = A_6 \rightarrow A_3, A_5$$

Hasil perhitungan frekuensi sesuai dengan uraian tersebut dapat dilihat pada Tabel 4.22.

Tabel 4.22 Frekuensi FLR pada pusat kluster atribut utama

Frekuensi (f)	Pusat kluster						
	A1	A2	A3	A4	A5	A6	A7
	5.3988	8.8597	10.1051	11.3725	16.7926	18.9222	18.9567
data uji 1	0	0	1	0	0	0	0
data uji 2	0	0	1	0	1	0	0

Dengan menggunakan pusat kluster dan frekuensi yang didapat maka perhitungan defuzzifikasi dapat diuraikan sebagai berikut:

$$\text{Data uji 1} = \frac{(10.1051)}{1} = 10.1051$$



$$Data\ uji\ 2 = \frac{(10.1051 + 16.7926)}{2} = 13.4488$$

Perbandingan data curah hujan (A) aktual dengan hasil prediksi dapat dilihat pada Tabel 2.23.

Tabel 4.23 Perbandingan data aktual dengan hasil prediksi

Dasarian / Bulan / Tahun	Data aktual	Hasil prediksi
2/3/2010	18.300	-
3/3/2010	10.727	10.1051
1/4/2010	9.100	13.4488

4.3.7 Adaptive Expectation

Adaptive Expectation memperbaiki hasil prediksi pada Tabel 4.23 dengan menggunakan nilai α terbaik sebesar (0.99) yang didapatkan dari Tabel 4.21. Perhitungan *Adaptive expectation* untuk memperbaiki hasil prediksi sebagai berikut

$$\begin{aligned} \text{Prediksi (3/3/2010)} &= 18.300 + (0.99 \times (10.1051 - 18.300)) \\ &= 10.1870 \end{aligned}$$

$$\begin{aligned} \text{Prediksi (1/4/2010)} &= 10.727 + (0.99 \times (13.4488 - 10.727)) \\ &= 13.4216 \end{aligned}$$

Setelah mendapatkan hasil prediksi akhir, melakukan perhitungan MSE. Perbandingan MSE sebelum dan setelah proses *Adaptive Expectation* dapat dilihat pada Tabel 2.24

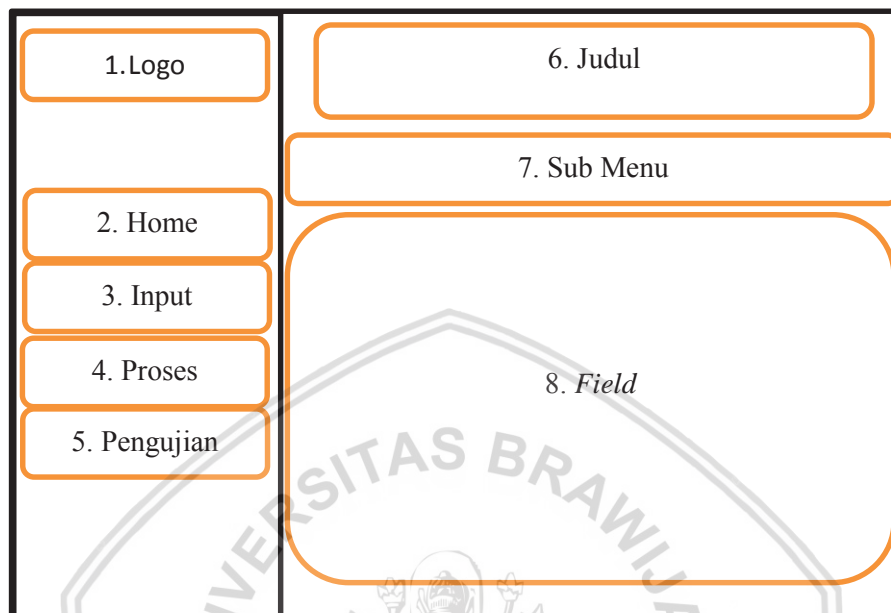
Tabel 4.24 Perbandingan MSE Adaptive Expectation

Dasarian / Bulan / Tahun	Data aktual	Hasil prediksi (Sebelum Adaptive Expectation)	Hasil prediksi (Setelah Adaptive Expectation)
2/3/2010	18.300	-	-
3/3/2010	10.727	10.1051	10.1870
1/4/2010	9.100	13.4488	13.4216
MSE		9.6494	9.4838



4.4 Perancangan antarmuka

Perancangan antarmuka penelitian ini bertujuan untuk memberi gambaran tampilan yang diharapkan pada aplikasi dan merancang letak tombol-tombol operasi beserta fungsinya.



Gambar 4.3 Perancangan antarmuka

Keterangan :

1. Logo : Tempat menampilkan logo fakultas.
2. Home : Tombol menuju halaman penjelasan dan dasar teori
3. Input : Tombol menuju halaman masukkan.
4. Proses : Tombol menuju halaman proses.
5. Pengujian : Tombol menuju halaman pengujian.
6. Judul : Tempat menampilkan judul penelitian.
7. *Sub Menu* : Tempat menampilkan judul halaman dan tombol halaman.
- Field* : Tempat menampilkan konten dari halaman.

BAB 5 IMPLEMENTASI

Hasil implementasi dari metode *Adaptive-Expectation based Multi-Attribute Time Series* dibahas pada bab ini. Implementasi dilakukan pada bahasa pemrograman Java. Bab ini terbagi menjadi dua subbab, yaitu meliputi implementasi Algoritma, Implementasi antarmuka.

5.1 Implementasi Algoritma

Dalam subbab implementasi algoritma ini membahas hasil implementasi dari prediksi curah hujan menggunakan metode *Adaptive-Expectation based Multi-Attribute Time Series*. Langkah-langkah dalam implementasi sesuai dengan bab sebelumnya yaitu bab perancangan.

5.1.1 Implementasi pembentukan *subinterval* dengan *Fuzzy C-Means*

Implementasi proses pembentukan *subinterval* menggunakan *Fuzzy C-Means* terdapat tiga tahap yaitu penentuan *Universe of Discourse* (U), penentuan pusat klaster dan penentuan batas *subinterval*.

5.1.1.1 Penentuan *Universe of Discourse*

Implementasi proses Penentuan *Universe of Discourse* terlampirkan pada Kode Program 5.1. Pada kode program tersebut baris 6, dan 10 dilakukan pemanggilan method mencari nilai *Minimum* dan *Maksimum*, dan mencari standard deviasi. Pada baris ke-7 dihitung jumlah dari semua data, dan pada baris ke-8 dihitung rata-rata dari semua data. Setelah mendapatkan hasil dari nilai minimum, maksimum, dan standar deviasi, hasil tersebut digunakan untuk menentukan Batas atas dan Bawah dari *Universe of Discourse* pada baris 11 dan 12.

Baris	Kode
1	<code>public void universeOfDiscourse() {</code>
2	<code> this.Dmin=9999;</code>
3	<code> this.Dmaks=0;</code>
4	<code> this.jumlah=0;</code>
5	<code> for(int i=0;i<this.data.length;i++){</code>
6	<code> setMinMax(i);</code>
7	<code> this.jumlah+=this.data[i];</code>
8	<code> }</code>
9	<code> rata=getJumlah()/data.length;</code>
10	<code> stdev();</code>
11	<code> this.universeAtas=getDmaks()+getStdev();</code>
12	<code> this.universeBawah=getDmin()-getStdev();</code>
	<code>}</code>

Kode Program 5.1 Penentuan *universe of discourse*

5.1.1.2 Implementasi proses perhitungan standar deviasi

Implementasi proses mencari standar deviasi dapat dilihat pada Kode Program 5.2. Implementasi proses tersebut dilakukan dari setiap atribut data yang akan digunakan untuk mencari batas dari *Universe of Discorse*. Pada baris 4 merupakan selisih antara data dengan rata-rata data, kemudian hasilnya dikuadratkan pada baris 5. Pada baris 5 juga menjumlahkan semua selisih kuadrat dari tiap data. Baris 4 dan 5 ini merupakan implementasi dari tabel 4.2 sebagai bahan menghitung rumus utama standar deviasi pada baris 7.

Baris	Kode
1	<code>private void stdev() {</code>
2	<code> double jmlks=0;</code>
3	<code> for(int i=0;i<data.length;i++){</code>
4	<code> double x=data[i]-getRata();</code>
5	<code> jmlks += Math.pow(x, 2);</code>
6	<code> }</code>
7	<code> stdev=Math.sqrt(jmlks/(data.length-1));</code>
8	<code>}</code>

Kode Program 5.2 Proses perhitungan standar deviasi

5.1.1.3 Implementasi proses penentuan pusat kluster

Setelah mendapatkan nilai standar deviasi dilakukan implementasi proses penentuan pusat kluster yang dapat dilihat pada Kode Program 5.3. Pada Baris 1 sampai 16 merupakan pembentukan matriks partisi awal yang dipanggil hanya saat pertama kali dijalankannya fungsi penentuan pusat kluster. Pada baris 2 dan 3 menunjukkan bahwa matriks partisi awal terbentuk dengan jumlah baris sebanyak jumlah data dan jumlah kolom sebanyak jumlah kluster. Pada baris 9 sampai 11 menunjukkan bahwa matriks partisi awal terbentuk dari bilangan random dan baris 13 menyimpan jumlah dari setiap barisnya yang akan digunakan sebagai pembagi pada normalisasi matriks partisi.

Normalisasi matriks partisi ditunjukkan pada baris 17 sampai 28. Pada baris 23 sampai 25, matriks partisi awal dinormalisasi yaitu dengan membagi data random matriks partisi awal dengan jumlah dari tiap barisnya. Setelah mendapatkan matriks partisi ternormalisasi, pada baris 29 sampai 60 dilakukan perhitungan pusat kluster secara *iterative* yang akan berhenti ketika kondisi berhenti terpenuhi. Baris 47 sampai 49 merupakan implementasi Tabel 4.5 dan Tabel 4.6 yaitu perhitungan pembilang dan penyebut untuk menghitung pusat kluster pada baris 54 dan 55. Langkah selanjutnya menghitung fungsi objektif dari hasil pusat kluster yang ditunjukkan pada baris 61 sampai 72 yaitu dengan jumlah semua (selisih dari data aktual dengan pusat kluster) dipangkatkan bobot. Implementasi dari Tabel 4.10 disimpan didalam array `temp[i][j]` pada baris 66 dan 67 disimpan dan digunakan untuk memperbaiki matriks partisi pada baris 72 sampai 97. Baris 98 sampai 111 menunjukkan kondisi berhenti perulangan method perhitunganPusatKluster() baris 29. Kondisi berhenti tersebut adalah

ketika selisih nilai objektif dengan nilai objektif sebelumnya lebih kecil dari minerror (baris 103 sampai 106) atau iterasi sudah mencapai iterasi maksimal yang ditentukan (baris 107 sampai 110). Setelah mencapai kondisi berhenti, pada baris 113-126 dilakukan pengurutan pusat kluster secara *ascending*.

Baris	Kode
1	<code>private void setMatriksPartisi() {</code>
2	<code> this.mPartisiAwal=new Integer</code>
3	<code> [this.data.length][this.jml_kluster];</code>
4	<code> this.jml_rowMP=new Integer</code>
5	<code> [this.data.length];</code>
6	<code> for(int i = 0; i < mPartisiAwal.length; i++){</code>
7	<code> this.jml_rowMP[i]=0;</code>
8	<code> for(int j=0;j<this.jml_kluster;j++){</code>
9	<code> this.mPartisiAwal[i][j]= 1 +</code>
10	<code> (int)(Math.random() * ((300 - 1) +</code>
11	<code>1));</code>
12	
13	<code> this.jml_rowMP[i]+=this.mPartisiAwal[i][j];</code>
14	<code> }</code>
15	<code> }</code>
16	<code>}</code>
17	<code>private void NormalisasiMP</code>
18	<code>(Integer[][] mPartisi, Integer[] jml_rowMP) {</code>
19	<code> this.mPartisi=new Double</code>
20	<code> [mPartisi.length][mPartisi[0].length];</code>
21	<code> for(int i=0;i<mPartisi.length;i++){</code>
22	<code> for(int j=0;j<mPartisi[0].length;j++){</code>
23	<code> this.mPartisi[i][j]=</code>
24	<code> ((double)mPartisi[i][j]/</code>
25	<code> (double)jml_rowMP[i]);</code>
26	<code> }</code>
27	<code> }</code>
28	<code> }</code>
29	<code>private void perhitunganPusatKluster() {</code>
30	<code> this.nilaiObjektif[this.iterasi]=0.0;</code>
31	<code> //pembilang pada rumus perhitungan pusat</code>
32	<code> Kluster yaitu jumlah dari</code>
33	<code> (MPnormalisasi dipangkatkan bobot) dikali</code>
34	<code> Data untuk setiap pusat kluster.</code>
35	<code> //penyebut pada rumus perhitungan pusat</code>
36	<code> kluster yaitu jumlah dari (MPnormalisasi</code>
37	<code> dipangkatkan bobot) untuk setiap pusat</code>
38	<code>kluster.</code>
38	<code> for(int j=0;j<this.jml_kluster;j++){</code>
40	<code> jml_penyebut[j]=0.0;</code>
41	<code> jml_pembilang[j]=0.0;</code>
42	<code> }</code>

Kode Program 5.3 Proses penentuan pusat kluster

Baris	Kode
43	for(int i=0;i<this.mPartisi.length;i++){
44	for(int j=0;j<this.jml_klaster;j++){
45	Xij[i][j]=Math.pow
46	this.mPartisi[i][j],this.bobot);
47	jml_penyebut[j]+=Xij[i][j];
48	jml_pembilang[j]+=
49	Xij[i][j]*this.data[i];
50	}
51	}
52	//set pusat klaster
53	for(int i=0;i<this.jml_klaster;i++){
54	this.pusatKlaster[i]=
55	jml_pembilang[i]/jml_penyebut[i];
56	}
57	perhitunganFungsiObjektif();
58	perbaikiMPartisi();
59	cekKondisiBerhenti();
60	}
61	private void perhitunganFungsiObjektif() {
62	for(int i=0;i<this.data.length;i++){
63	for(int
64	j=0;j<this.pusatKlaster.length;j++){
65	temp[i][j]=Math.pow(Math.abs(this.data[i]-
66	this.pusatKlaster[j]),this.bobot);
67	nilaiObjektif[this.iterasi]
68	+=temp[i][j]*Xij[i][j];
69	}
70	}
71	}
72	}
73	private void perbaikiMPartisi() {
74	int hitung=this.bobot-1;
75	double pangkat= (double) (-1/hitung);
76	this.temp_rowMatriks=new
77	Double[this.data.length];
78	for(int i=0;i<this.data.length;i++){
79	temp_rowMatriks[i]=0.0;
80	for(int
81	j=0;j<this.pusatKlaster.length;j++){
82	temp_rowMatriks[i]+=
83	Math.pow(this.temp[i][j],pangkat);
84	}
85	}
86	//memperbaiki matriks normalisasi
87	}

Kode Program 5.3 (lanjutan)

Baris	Kode
88	for(int i=0;i<this.data.length;i++){
89	for(int j=0;
90	j<this.pusatKlaster.length;j++){
91	this.mPartisi[i][j]=
92	Math.pow(temp[i][j],pangkat)/
93	this.temp_rowMatriks[i];
94	
95	}
96	}
97	}
98	private void cekKondisiBerhenti() {
99	if(this.iterasi==0){
100	this.iterasi+=1;
101	this.perhitunganPusatKlaster();
102	}else{
103	if((Math.abs(
104	this.nilaiObjektif[iterasi]-
105	this.nilaiObjektif[iterasi-1])
106	>this.minerror)
107	&&this.iterasi<this.maksiterasi-1){
108	this.iterasi+=1;
109	this.perhitunganPusatKlaster();
110	}
111	}
112	}
113	private void rankPusatCluster() {
114	for(int i=0;i<this.jml_klaster;i++){
115	for(int j=0;j<this.jml_klaster-1;j++){
116	if(this.pusatKlaster[j]>
117	this.pusatKlaster[j+1]){
118	double tempo =
119	this.pusatKlaster[j];
120	this.pusatKlaster[j]=
121	this.pusatKlaster[j+1];
122	this.pusatKlaster[j+1]=tempo;
123	}
124	}
125	}
126	}

Kode Program 5.3 (lanjutan)

5.1.1.4 Implementasi pembentukan batas *subinterval*

Dari implementasi sebelumnya didapatkan hasil pusat klaster masing-masing *attribute* yang digunakan dalam implementasi pembentukan *subinterval*. Implementasi pembentukan *subinterval* tersebut dapat dilihat pada Tabel 5.4. Pada baris 7 sampai 11 menunjukkan bahwa batas bawah subinterval pertama diisi dengan nilai batas bawah *universe of discourse*, dan batas atas subinterval pertama diisi dengan nilai tengah antara pusat pertama dan pusat kedua. Pada baris 11 sampai 18 menunjukkan bahwa batas atas subinterval terakhir diisi dengan nilai batas atas *universe of discourse*, dan batas bawah subinterval

terakhir diisi dengan nilai tengah antara pusat terakhir dan pusat sebelumnya. Pada baris 18 sampai 28 menunjukkan bahwa selain dua kondisi diatas yaitu subinterval diantara pertama dan terakhir, batas bawah subinterval berisi nilai tengah antara pusat tersebut dengan pusat sebelumnya, sedangkan batas atas subinterval berisi nilai tengah antara pusat tersebut dengan pusat selanjutnya.

Baris	Kode
1	<code>private void batasSubinterval() {</code>
2	<code> this.subIntervalBawah=new</code>
3	<code> Double[this.jml_klaster];</code>
4	<code> this.subIntervalAtas=new</code>
5	<code> Double[this.jml_klaster];</code>
6	<code> for(int i=0;i<this.jml_klaster;i++){</code>
7	<code> if(i==0){</code>
8	<code> this.subIntervalBawah[i]=this.universeBawah;</code>
9	<code> this.subIntervalAtas[i]=</code>
10	<code> (this.pusatKlaster[i]+this.pusatKlaster[i+1])/2;</code>
11	<code> }else{</code>
12	<code> if(i==this.jml_klaster-1){</code>
13	<code> this.subIntervalBawah[i]=</code>
14	<code> (this.pusatKlaster[i]+</code>
15	<code> this.pusatKlaster[i-1])/2;</code>
16	<code> this.subIntervalAtas[i]=</code>
17	<code> this.universeAtas;</code>
18	<code> }else{</code>
19	<code> this.subIntervalBawah[i]=</code>
20	<code> (this.pusatKlaster[i]+</code>
21	<code> this.pusatKlaster[i-1])/2;</code>
22	<code> this.subIntervalAtas[i]=</code>
23	<code> (this.pusatKlaster[i]+</code>
24	<code> this.pusatKlaster[i+1])/2;</code>
25	<code> }</code>
26	<code> }</code>
27	<code> }</code>
28	<code>}</code>

Kode Program 5.4 Penentuan batas subinterval

5.1.2 Implementasi pembentukan himpunan Fuzzy

Proses selanjutnya yaitu implementasi pembentukan himpunan fuzzy yang dapat dilihat pada Kode Program 5.5. Pembentukan himpunan fuzzy ini berdasarkan hasil batas atas dan bawah subinterval dari proses sebelumnya. Baris 3 sampai 7 menunjukkan himpunan fuzzy pertama berada di antara batas bawah subintervalnya (baris 4) sampai nilai tengah dari kedua batas subinterval himpunan selanjutnya (baris 5 dan 6). Baris 7 sampai 12 menunjukkan bahwa himpunan fuzzy terakhir berada di antara nilai tengah dari kedua batas subinterval himpunan sebelumnya (baris 9 dan 10) sampai batas atas subinterval himpunan terakhir itu sendiri (baris 11). Selain himpunan pertama dan terakhir, himpunan fuzzy terletak diantara nilai tengah dari kedua batas subinterval himpunan

sebelumnya (baris 14 dan 15) sampai nilai tengah dari kedua batas subinterval himpunan sesudahnya (baris 17 dan 18).

Baris	Kode
1	private void setBatas() {
2	for(int i=0;i<this.jml_klaster;i++){
3	if(i==0){
4	this.fuzzySetBawah[i]=this.subIntervalBawah[i];
5	this.fuzzySetAtas[i]=(this.subIntervalBawah[i+1]+
6	this.subIntervalAtas[i+1])/2;
7	}else if(i==this.jml_klaster-1){
8	this.fuzzySetBawah[i]=
9	(this.subIntervalBawah[i-1]+
10	this.subIntervalAtas[i-1])/2;
11	this.fuzzySetAtas[i]=this.subIntervalAtas[i];
12	}else{
13	this.fuzzySetBawah[i]=
14	(this.subIntervalBawah[i-1]+
15	this.subIntervalAtas[i-1])/2;
16	this.fuzzySetAtas[i]=
17	(this.subIntervalBawah[i+1]+
18	this.subIntervalAtas[i+1])/2;
19	}
20	}
21	}

Kode Program 5.5 Pembentukan Himpunan Fuzzy

5.1.3 Implementasi proses fuzzifikasi

Implementasi proses fuzzifikasi dilakukan berdasarkan himpunan fuzzy yang telah terbentuk dari proses sebelumnya. Implementasi proses fuzzifikasi dapat dilihat pada Kode Program 5.6. Baris 1 sampai 52 menunjukkan perhitungan fuzzifikasi data sesuai letak data dari batas atas dan bawah semua himpunan fuzzy.

Pada baris 6 sampai 19 menunjukkan perhitungan pada himpunan fuzzy pertama yang terbagi menjadi tiga kondisi. Kondisi pertama pada baris 7 sampai 10, jika data berada diantara batas bawah himpunan fuzzy pertama dan batas bawah himpunan fuzzy selanjutnya maka hasil fuzzifikasi bernilai 1. Kondisi kedua pada baris 11 sampai 17, jika data berada diantara batas atas himpunan fuzzy pertama (a) dan batas bawah himpunan fuzzy selanjutnya (b) maka hasil fuzzifikasi = $\frac{a-data}{a-b}$. Selain kedua kondisi tersebut hasil fuzzifikasi bernilai 0 (baris 18).

Baris 20 sampai 38 menunjukkan perhitungan pada himpunan fuzzy terakhir yang terbagi menjadi tiga kondisi. Kondisi pertama pada baris 21 sampai 29, jika data berada diantara batas bawah himpunan fuzzy terakhir (a) dengan batas atas fuzzy sebelumnya (b) maka hasil fuzzifikasi = $\frac{data-a}{b-a}$. Kondisi kedua pada baris 30 sampai 35, jika data berada diantara batas atas himpunan fuzzy terakhir dan

batas atas himpunan sebelumnya maka hasil fuzzifikasi = 1. Selain pada kedua kondisi tersebut (baris 35 sampai 37) hasil fuzzifikasi bernilai 0.

Baris 38 sampai 52 menunjukkan perhitungan pada himpunan fuzzy selain himpunan pertama dan kedua yang terbagi menjadi tiga kondisi. Kondisi pertama pada 38 sampai 44, jika data berada diantara batas bawah himpunan fuzzy sekarang (a) dan batas bawah himpunan selanjutnya (b) maka hasil fuzzifikasi = $\frac{data-a}{b-a}$. Kondisi kedua pada 44 sampai 50, jika data berada diantara batas atas himpunan fuzzy sekarang (a) dan batas bawah himpunan selanjutnya (b) maka hasil fuzzifikasi = $\frac{a-data}{a-b}$. Selain dua kondisi tersebut (baris 50 dan 51), hasil fuzzifikasi bernilai 0.

Setelah mendapatkan semua nilai hasil fuzzifikasi, pada baris 54-57 dilakukan pencarian nilai maksimal setiap baris datanya. Jika nilai maksimal berada pada himpunan fuzzy $A_j, j = 1, 2, \dots, jumlah\ klaster$ maka data tersebut ditandai sebagai A_j . Perhitungan fuzzifikasi pada proses ini dilakukan pada empat *attribute* sehingga menghasilkan fuzzifikasi seperti pada Tabel 4.18.

Baris	Kode
1	private void hitungFuzzifikasi() {
2	double maks;
3	for(int i=0;i<this.data.length;i++){
4	maks=0.0;
5	for(int j=0;j<this.jml_klaster;j++){
6	if(j==0){
7	if(this.data[i]>this.fuzzySetBawah[j]&&
8	this.data[i]<this.fuzzySetBawah[j+1]){
9	this.hitungFuzzy[i][j]=1.0;
10	}else
11	if(this.data[i]<this.fuzzySetAtas[j]&&
12	this.data[i]>this.fuzzySetBawah[j+1]){
13	this.hitungFuzzy[i][j]=
14	(this.fuzzySetAtas[j]-this.data[i])/
15	(this.fuzzySetAtas[j]-
16	this.fuzzySetBawah[j+1]);
17	}else{
18	this.hitungFuzzy[i][j]=0.0;
19	}
20	}else if(j==this.jml_klaster-1){
21	if(this.data[i]>
22	this.fuzzySetBawah[jml_klaster-1]&&
23	this.data[i]<
24	this.fuzzySetAtas[jml_klaster-2]){
25	this.hitungFuzzy[i][j]=(this.data[i]-
26	this.fuzzySetBawah[jml_klaster-1])/
27	(this.fuzzySetAtas[jml_klaster-2]-
28	this.fuzzySetBawah[jml_klaster-1]);
29	}else

Kode Program 5.6 Proses fuzzifikasi



Baris	Kode
30	if(this.data[i]>
31	this.fuzzySetAtas[jml_klaster-2]
32	&&this.data[i]<
33	this.fuzzySetAtas[jml_klaster-1]){
34	this.hitungFuzzy[i][j]=1.0;
35	}else{
36	this.hitungFuzzy[i][j]=0.0;
37	}
38	}else{if(this.data[i]>this.fuzzySetBawah[j]
39	&&this.data[i]<this.fuzzySetBawah[j+1]){
40	this.hitungFuzzy[i][j]=(this.data[i]-
41	this.fuzzySetBawah[j])/
42	(this.fuzzySetBawah[j+1]-
43	this.fuzzySetBawah[j]);
44	}else if(this.data[i]>this.fuzzySetBawah[j+1]
45	&&this.data[i]<this.fuzzySetAtas[j]){
46	this.hitungFuzzy[i][j]=
47	(this.fuzzySetAtas[j]-this.data[i])/
48	(this.fuzzySetAtas[j]-
49	this.fuzzySetBawah[j+1]);
50	}else {this.hitungFuzzy[i][j]=0.0;
51	}
52	}
53	//mencari maksimal dalam satu baris
54	if(maks<this.hitungFuzzy[i][j]){
55	maks=this.hitungFuzzy[i][j];
56	this.hasilFuzzifikasi[i]=j+1;
57	}
58	}
59	}
60	}

Kode Program 5.6 (lanjutan)

5.1.4 Implementasi Proses Pembentukan FLR

Proses pembentukan *Fuzzy Logic Relationship* menggunakan data hasil fuzzifikasi dari proses sebelumnya. Adapun proses pembentukan FLR dapat dilihat pada Kode Program 5.7. Baris 6 dan 7 menunjukkan pembentukan FLR kiri, sedangkan baris 10 menunjukkan pembentukan FLR kanan. Kemudian FLR kanan dikelompokkan sesuai dengan persamaan 2.6 yang ditunjukkan pada baris 15 dan 16.

Baris	Kode
1	public class FLR {
2	private void setFLRlatih() {
3	for(int i=0;i<flr_kiri.length;i++){
4	for(int j=0;j<this.order;j++){

Kode Program 5.7 Proses pembentukan *Fuzzy Logic Relationship*



Baris	Kode
5	
6	for(int k=0;k<this.attribute-1;k++){
7	this.flr_kiri[i][j][k]=
8	this.fuzz[i+j][k+1];
9	}
10	}
11	flr_kanan[i]=fuzz[i+2][0];
12	}
13	for(int i=0;i<this.jml_klaster;i++){
14	for(int j=0;j<this.flr_kanan_latih.length-1;j++){
15	if(this.flr_kanan_latih[j]==i){
16	this.flr_kanan_def[i][flr_kanan_latih[j+1]]=
17	this.flr_kanan_latih[j+1];
18	}
19	}
20	}

Kode Program 5.7 (lanjutan)

5.1.5 Implementasi proses *fitness* nilai α

Setelah mendapatkan *Fuzzy Logic Relationship* (FLR) langkah selanjutnya yaitu melakukan pelatihan atau pencarian nilai alpha terbaik dari prediksi data latih itu sendiri. Proses pencarian nilai alpha terbaik dapat dilihat pada Kode Program 5.7. Baris 1 sampai 26 merupakan proses memprediksi curah hujan pada data latih yang nantinya digunakan untuk mencari nilai alpha yaitu pada baris 38 sebagai (this.hasil prediksi). Baris 27 sampai 52 merupakan suatu fungsi untuk mencari nilai alpha terbaik dengan melakukan *fitness* sebanyak 99 kali. Baris 44 sampai 47 melakukan pencarian nilai MSE paling kecil, dan nilai alpha yang memiliki nilai MSE terkecil akan dipilih dan disimpan pada baris 45. Nilai alpha terbaik akan digunakan untuk melakukan adaptasi pada proses prediksi data uji.

Baris	Kode
1	private void prediksiTraining() {
2	int flr_terpilih,count;
3	double jumlah_defuzzifikasi;
4	for(int i=0;i<this.flr_kanan.length-1;i++){
5	jumlah_defuzzifikasi=0;
6	count=0;
7	flr_terpilih=this.flr_kanan[i];
8	}

Kode Program 5.8 Proses *Adaptive-Expectation*



Baris	Kode
9	for(int j=0;j<this.jml_klaster;j++){
10	
11	if(this.flr_kanan_def[flr_terpilih][j]!=null){
12	
13	jumlah_defuzzifikasi+=this.pusatKlaster[j];
14	count+=1;
15	}
16	}
17	if(jumlah_defuzzifikasi==0){
18	
19	this.hasilPrediksi[i]=this.pusatKlaster[flr_terpilih
20];
21	}else{
22	
23	this.hasilPrediksi[i]=jumlah_defuzzifikasi/count;
24	}
25	}
26	}
27	private void adaptiveEx() {
28	double alpha=0.01,temp,MSE,minMSE=99999.9;
29	int count;
30	for(int i=0;i<99;i++){
31	temp=0;
32	count=0;
33	for(int
34	t=0;t<this.hasilPrediksi.length;t++){
35	// (t) pada data sama waktunya dengan (t-1) pada
36	rumus hasil prediksi.
37	this.Adapted_forecast[t]=data[t]+
38	alpha*(this.hasilPrediksi[t]-data[t]);
39	
40	temp+=Math.pow(this.Adapted_forecast[t]-
41	data[t+1],2);
42	count+=1;
43	MSE=temp/count;
44	if(minMSE>MSE){
45	this.fixAlpha=alpha;
46	minMSE=MSE;
47	}
48	alpha+=0.01;
49	}
50	}

Kode Program 5.8 Proses Adaptive-Expectation

5.1.6 Implementasi proses defuzzifikasi

Implementasi proses defuzzifikasi dapat dilihat pada Kode Program 5.9. Pada baris 9 sampai 31 dilakukan perhitungan selisih FLR kiri data latih dengan FLR kiri data uji. Baris 20 sampai 22 menunjukkan bahwa semua selisih FLR kiri data latih dengan uji dijumlahkan dan disimpan pada variabel (*hitung_selisih*). Setelah menghitung selisih data uji dengan semua FLR latih, pada baris 25 sampai 29 dilakukan pencarian selisih terkecil dan menyimpannya sebagai FLR terpilih. Pada baris 39 penjumlahan pusat klaster dari frekuensi FLR terpilih, baris 40 menghitung frekuensi kemunculan kombinasi. Jika kombinasi FLR kanan kosong yaitu pada baris 43 sampai 46, maka sesuai *Naive Forecasting Principle* hasil prediksi sama dengan nilai pusat klaster pada FLR terpilih itu sendiri. Jika kombinasi FLR kanan tidak kosong, maka Pada baris 48 dilakukan perhitungan defuzzifikasi menggunakan jumlah pusat klaster dibagi frekuensi (sesuai dengan aturan pada persamaan 2.7).

Baris	Kode
1	<code>Defuzzifikasi(Integer[][][] flr_kiri,</code>
2	<code>Integer[] flr_kanan, int makslatih, Double[]</code>
3	<code>pusatKlaster) {</code>
4	<code> this.makslatih=makslatih;</code>
5	<code> this.jml_klaster=7;</code>
6	<code> menghitungSelisih();</code>
7	<code> prediksi();</code>
8	<code> }</code>
9	<code> private void menghitungSelisih() {</code>
10	<code> int hitung_selisih;</code>
11	<code> int min=9999;</code>
12	<code> for(int i=0;i<</code>
13	<code> this.flr_kiri_latih.length;i++){</code>
14	<code> hitung_selisih=0;</code>
15	<code> for(int</code>
16	<code> j=0;j<this.flr_kiri_latih[0].length;j++){</code>
17	<code> for(int</code>
18	<code> k=0;k<this.flr_kiri_latih[0][0].length;k++){</code>
19	<code> hitung_selisih+=Math.abs(this.flr_kiri_latih[i][j][k]</code>
20	<code> -this.flr_kiri_uji[0][j][k]);</code>
21	<code> }</code>
22	<code> }</code>
23	<code> }</code>
24	<code> if(hitung_selisih<min) {</code>
25	<code> this.flr_terpilih=this.flr_kanan_latih[i];</code>
26	<code> min=hitung_selisih;</code>
27	<code> }</code>
28	<code> }</code>
29	<code> }</code>
30	<code> }</code>
31	<code>}</code>

Kode Program 5.9 Proses defuzzifikasi

Baris	Kode
32	<code>private void prediksi() {</code>
33	<code>double jumlah_defuzzifikasi=0;</code>
34	<code>int count=0;</code>
35	<code>for(int i=0;i<this.jml_klaster;i++){</code>
36	
37	<code>if(this.flr_kanan_def[this.flr_terpilih][i]!=null){</code>
38	
39	<code>jumlah_defuzzifikasi+=this.pusatKlaster[i];</code>
40	<code>count+=1;</code>
41	<code>}</code>
42	<code>}</code>
43	<code>if(jumlah_defuzzifikasi==0){</code>
44	
45	<code>this.hasilPrediksi=this.pusatKlaster[flr_terpilih];</code>
46	<code>}else{</code>
47	
48	<code>this.hasilPrediksi=jumlah_defuzzifikasi/count;</code>
49	<code>}</code>
50	<code>}</code>
51	<code>public double getHasilPrediksi() {</code>
52	<code>return hasilPrediksi;</code>
53	<code>}</code>
54	<code>}</code>
55	

Kode Program 5.9 (lanjutan)

5.1.7 Implementasi proses Adaptive-Expectation pada hasil defuzzifikasi

Implementasi proses adaptive-expectation pada hasil defuzzifikasi dapat dilihat pada Kode Program 5.10. Pada baris 3 dan 4 merupakan proses menghitung hasil prediksi akhir setelah diterapkan *Adaptive-Expectation* menggunakan nilai *alpha* terpilih.

Baris	Kode
1	<code>private void hitung() {</code>
2	<code> this.prediksiAkhir=</code>
3	<code> this.data_tmin1+(this.fixAlpha*</code>
4	<code> (this.hasilPrediksi-this.data_tmin1));</code>
5	<code> }</code>
6	<code>}</code>

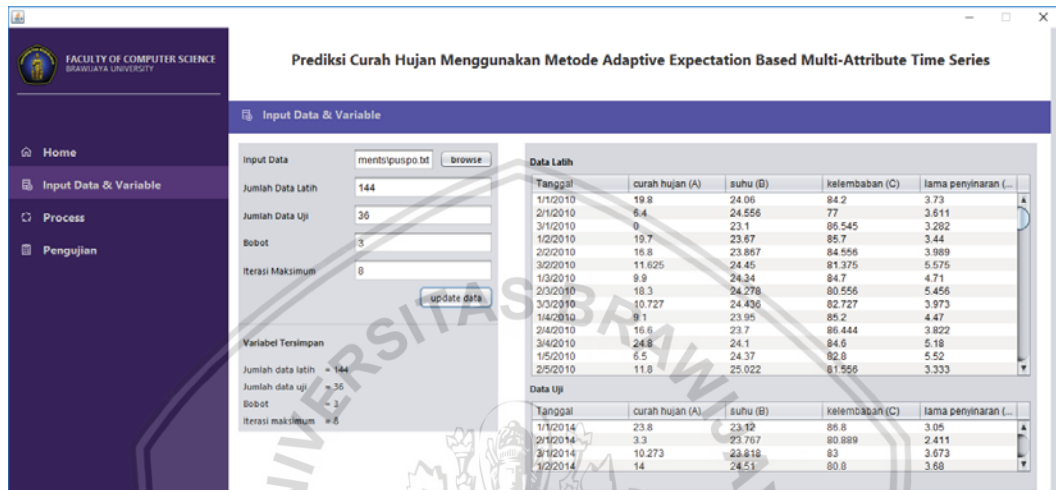
Kode Program 5.10 Proses adaptasi hasil prediksi



5.2 Implementasi antarmuka

5.2.1 Antarmuka halaman *input*

Halaman *input* atau masukan digunakan untuk melakukan pengisian data dan variabel yang dibutuhkan sistem untuk melakukan prediksi curah hujan. Data yang berupa (.txt) dimasukkan yang akan digunakan sebagai data latih dan uji. Variabel yang dibutuhkan sistem diantaranya, jumlah data latih, jumlah data uji, bobot, dan iterasi maksimal. Antarmuka halaman *input* dapat dilihat pada Gambar 5.1.

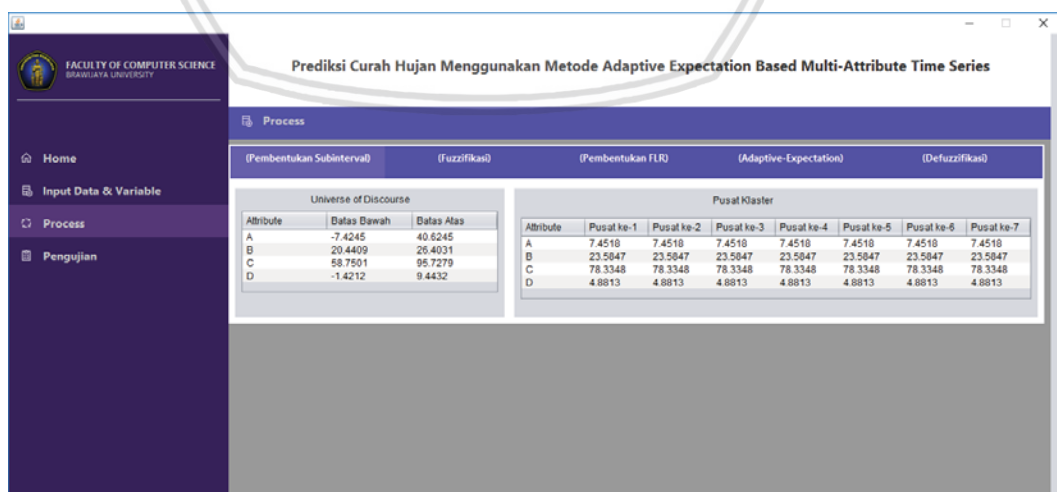


Gambar 5.1 Antarmuka halaman input

5.2.2 Antarmuka halaman proses

5.2.2.1 Antarmuka proses *subinterval*

Halaman proses subinterval digunakan untuk menampilkan hasil perhitungan subinterval berupa *Universe of Discourse* dan Pusat Klaster. Antarmuka halaman proses subinterval dapat dilihat pada Gambar 5.2.



Gambar 5.2 Antarmuka proses subinterval



5.2.2.2 Antarmuka proses fuzzifikasi

Halaman proses fuzzifikasi digunakan untuk menampilkan hasil perhitungan fuzzifikasi berupa hasil fuzzifikasi sesuai dengan derajat keanggotaannya. Antarmuka halaman proses subinterval (A) dapat dilihat pada Gambar 5.3.

Data ke-	Curah Hujan (A)	Suhu (B)	Kelembaban (C)	Lama Penyinaran (D)
1	A7	B6	C7	D2
2	A2	B7	C2	D2
3	A1	B2	C7	D1
4	A7	B6	C7	D2
5	A7	B6	C7	D2
6	A6	B7	C6	D6
7	A6	B7	C7	D2
8	A7	B6	C6	D6
9	A6	B7	C7	D2
10	A6	B6	C7	D2
11	A7	B6	C7	D2
12	A7	B6	C7	D6
13	A2	B7	C7	D6
14	A6	B7	C6	D2
15	A6	B7	C7	D2
16	A7	B6	C7	D2
17	A7	B6	C6	D6

Gambar 5.3 Antarmuka proses fuzzifikasi

5.2.2.3 Antarmuka proses defuzzifikasi

Halaman proses defuzzifikasi digunakan untuk menampilkan hasil perhitungan defuzzifikasi berupa hasil prediksi dan MSE. Hasil prediksi, variabel-variabel input, dan MSE dapat disimpan sebagai data dalam pengujian. Antarmuka halaman proses defuzzifikasi dapat dilihat pada Gambar 5.4.

Data ke-	Dasaran	Data Aktual (Curah Hujan)	Hasil Prediksi (Curah hujan)
1	1/1/2014	23.8000	11.6573
2	2/1/2014	3.3000	13.6276
3	3/1/2014	10.2730	6.9156
4	1/2/2014	14.0000	8.4674
5	2/2/2014	6.0000	9.8746
6	3/2/2014	10.3750	6.9291
7	1/3/2014	6.5000	6.5041
8	2/3/2014	16.6000	7.1091
9	3/3/2014	3.2730	10.3851
10	1/4/2014	6.3000	5.9892
11	2/4/2014	9.3000	7.0487
12	3/4/2014	14.7000	8.0987

MSE = 37.3148527777778

Simpan sebagai pengujian

Jumlah Data Latih Bobot

Jumlah Data Uji Iterasi Maksimum

Gambar 5.4 Antarmuka proses defuzzifikasi

5.2.3 Antarmuka halaman pengujian

5.2.3.1 Antarmuka pengujian jumlah data latih

Halaman pengujian jumlah data latih digunakan untuk menampilkan hasil pengujian data latih berupa Tabel. Antarmuka halaman pengujian data latih dapat dilihat pada Gambar 5.5.

Pengujian Data Lath							
Jumlah Data Lath	Uji 1 (MSE)	Uji 2 (MSE)	Uji 3 (MSE)	Uji 4 (MSE)	Uji 5 (MSE)	Rata-rata (MSE)	Lama Komputasi...
36	38.9717	41.6193	37.3833	37.6674	41.4859	39.4265	15
72	35.2229	35.5787	34.9783	36.1413	36.3099	35.6456	15
108	35.6980	35.4993	34.8572	36.0295	34.8605	35.3885	15
144	34.9400	34.8634	35.5108	33.6846	35.3609	34.8719	18

Variabel tetap =
 Jumlah Data Uji = 36
 Bobot = 2
 Iterasi Maksimum = 3

Gambar 5.5 Antarmuka pengujian jumlah data latih

5.2.3.2 Antarmuka pengujian jumlah data uji

Halaman pengujian jumlah data uji digunakan untuk menampilkan hasil pengujian data uji berupa Tabel. Antarmuka halaman pengujian data uji dapat dilihat pada Gambar 5.5.

Pengujian Data Uji							
Jumlah Data Uji	Uji 1 (MSE)	Uji 2 (MSE)	Uji 3 (MSE)	Uji 4 (MSE)	Uji 5 (MSE)	Rata-rata (MSE)	Lama Komputasi...
36	38.7497	36.8399	38.1875	39.9771	38.3978	38.4304	15
72	52.8787	52.4699	52.8255	52.4488	53.1873	52.7622	24
108	45.8043	45.5167	47.1702	45.7323	46.3659	46.1179	31
144	49.6014	45.9292	48.2430	48.7319	47.5943	48.0200	62

Variabel tetap =
 Jumlah Data Lath = 36
 Bobot = 2
 Iterasi Maksimum = 3

Gambar 5.6 Antarmuka pengujian jumlah data uji

5.2.3.3 Antarmuka pengujian bobot

Halaman pengujian bobot digunakan untuk menampilkan hasil pengujian bobot berupa Tabel. Antarmuka halaman pengujian bobot dapat dilihat pada Gambar 5.7.

Pengujian Bobot							
Bobot	Uji 1 (MSE)	Uji 2 (MSE)	Uji 3 (MSE)	Uji 4 (MSE)	Uji 5 (MSE)	Rata-rata (MSE)	Lama komputasi (...)
2	40.4382	35.1802	39.7594	38.0630	38.4633	38.3808	15
3	36.0433	36.0433	36.0433	36.0433	36.0433	36.0433	15
4	36.0433	36.0433	36.0433	36.0433	36.0433	36.0433	15
5	36.0433	36.0433	36.0433	36.0433	36.0433	36.0433	15

Variabel tetap =
 Jumlah Data Lath = 36
 Jumlah Data Uji = 36
 Iterasi Maksimum = 3

Gambar 5.7 Antarmuka pengujian bobot

5.2.3.4 Antarmuka pengujian iterasi maksimal

Halaman pengujian iterasi maksimal digunakan untuk menampilkan hasil pengujian iterasi maksimal berupa Tabel. Antarmuka halaman pengujian data uji dapat dilihat pada Gambar 5.8.

Pengujian Iterasi							
Iterasi Maksimum	Uji 1 (MSE)	Uji 2 (MSE)	Uji 3 (MSE)	Uji 4 (MSE)	Uji 5 (MSE)	Rata-rata (MSE)	Lama Komputasi(...)
3	39.2882	39.6670	41.8719	38.0235	37.5501	39.2821	16
4	33.4753	39.8843	38.4321	35.2511	39.8719	37.3849	15
5	36.9787	39.7610	34.1729	35.6863	36.6258	36.6451	15
6	35.5624	35.2772	32.7952	39.5499	38.7175	36.3804	15
7	38.0049	36.5020	33.6889	35.1651	32.5700	35.1958	15
8	34.3021	35.7113	32.1073	36.5221	35.0140	34.7474	18

Variabel tetap =
 Jumlah Data Lath = 36
 Jumlah Data Uji = 36
 Bobot = 2

Gambar 5.8 Antarmuka pengujian iterasi maksimal

BAB 6 PENGUJIAN DAN ANALISIS

Pembahasan tentang hasil pengujian beserta analisis dari implementasi prediksi curah hujan menggunakan metode *Adaptive-Expectation based Multi-Attribute Fuzzy Time Series* berada pada bab ini.

6.1 Sistematika pengujian

Sistematika pengujian penelitian ini berdasarkan atas skenario pengujian yang tercantum pada bab 3, yaitu pengujian pada 4 variabel masukan. Variabel masukan yang diuji yaitu pengujian jumlah data latih, pengujian jumlah data uji, pengujian bobot, dan pengujian iterasi maksimal. Pengujian tersebut menggunakan nilai *Mean Square Error* (MSE) sebagai tolak ukur pengaruh variabel-variabel tersebut terhadap hasil prediksi.

6.2 Hasil dan analisis pengujian

Untuk mengetahui pengaruh suatu variabel masukan terhadap nilai MSE dari hasil prediksi, pengujian dilakukan menggunakan nilai yang berbeda dari variabel tersebut dan variabel lainnya bernilai tetap pada setiap pengujiannya. Pengujian dari suatu nilai variabel dilakukan sebanyak 5 kali dan mengambil nilai rata-rata MSE dari pengujian nilai tersebut. Berikut merupakan hasil dan analisis pengujian setiap variabel sesuai dengan skenario pengujian.

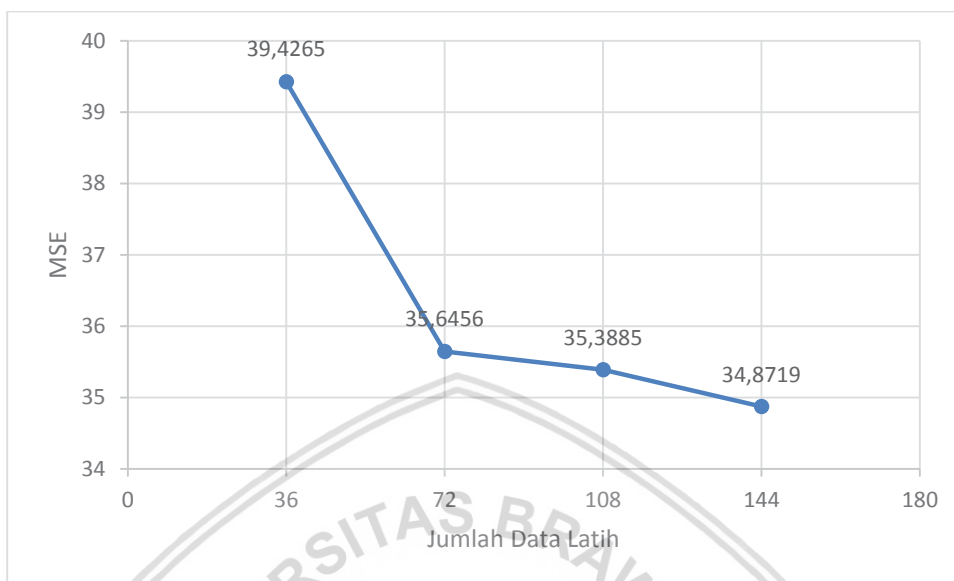
6.2.1 Pengujian pengaruh jumlah data latih terhadap nilai MSE

Pengujian jumlah data latih ini menggunakan data dasarian dimulai dari tahun 2010 sampai 2013 sebagai jumlah data latih yang berubah. Sedangkan variabel lain bernilai tetap yaitu jumlah data uji sebesar 36 data tahun 2014, bobot bernilai 2, dan iterasi maksimal bernilai 3. Penambahan jumlah data latih yaitu 1 tahun, 2 tahun, 3 tahun, dan 4 tahun sebelumnya dari data uji tahun 2014. Penambahan data tersebut sebesar 36 data karena dalam setahun terdapat 36 data dasarian. Hasil pengujian data latih dapat dilihat pada Tabel 6.1.

Tabel 6.1 Pengujian jumlah data latih

Jumlah Data Latih	Pengujian ke- (MSE)					Rata-Rata	
	1	2	3	4	5	MSE	Lama Komputasi (milidetik)
36	38.971	41.619	37.388	37.667	41.485	39.4265	15
72	35.222	35.578	34.978	36.141	36.306	35.6456	15
108	35.696	35.499	34.857	36.029	34.860	35.3885	15
144	34.941	34.863	35.510	33.684	35.360	34.8719	18

Dari hasil pengujian yang ditunjukkan pada Tabel 6.1, terjadi perubahan MSE ketika jumlah data latih bertambah. Perubahan nilai MSE pada pengujian data latih dapat dilihat dalam bentuk grafik yang ditunjukkan Gambar 6.1.



Gambar 6.1 Grafik hasil pengujian pengaruh jumlah data latih terhadap MSE

Dari grafik menunjukkan nilai MSE terhadap jumlah data latih memiliki *trendline* menurun. Semakin besar jumlah data latih yang digunakan, semakin kecil nilai MSE yang dihasilkan. Dari grafik 6.1 didapatkan nilai MSE terbaik dengan menggunakan 144 data latih.

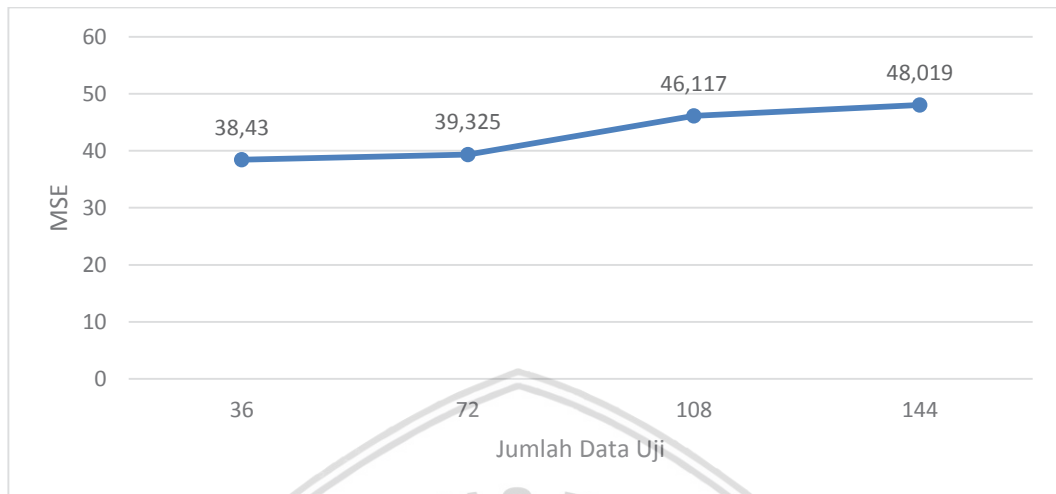
6.2.2 Pengujian pengaruh jumlah data uji terhadap nilai MSE

Pengujian jumlah data uji ini menggunakan data dasarian dimulai dari 36 sampai 144 data uji dengan penambahan sebesar 36 data pada setiap tahap ujinya. Sedangkan variabel lain bernilai tetap yaitu jumlah data latih sebesar 36 data, bobot bernilai 2, dan iterasi maksimal bernilai 3. Penambahan setiap tahap pengujiannya sebesar 36 data karena dalam setahun terdapat 36 data dasarian. Hasil pengujian data uji dapat dilihat pada Tabel 6.2.

Tabel 6.2 Pengujian jumlah data uji

Jumlah Data Uji	Pengujian ke- (MSE)					Rata-Rata	
	1	2	3	4	5	MSE	Lama Komputasi (milidetik)
36	38.750	36.840	38.188	39.977	38.398	38.430	15
72	37.287	37.566	38.870	41.384	41.518	39.325	24
108	45.804	45.517	47.170	45.732	46.366	46.117	31
144	49.601	45.929	48.243	48.732	47.594	48.019	62

Dari hasil pengujian yang ditunjukkan pada Tabel 6.2, terjadi perubahan MSE ketika jumlah data uji bertambah. Perubahan nilai MSE pada pengujian data uji dapat dilihat dalam bentuk grafik yang ditunjukkan Gambar 6.2.



Gambar 6.2 Grafik hasil pengujian jumlah data uji terhadap MSE

Dari grafik menunjukkan nilai MSE terhadap jumlah data uji memiliki *trendline* naik namun tidak selalu naik dalam setiap penambahan jumlah data. Hal ini disebabkan oleh berubahnya jumlah data uji, data aktual, dan hasil prediksi pada perhitungan MSE. Sedangkan pada pengujian variabel lain, hanya hasil prediksi yang berubah pada proses perhitungan MSE. Dari hasil pengujian jumlah data uji didapatkan 36 data terbaik sebagai data uji, yang ditunjukkan dengan nilai MSE paling sedikit dan lama komputasi terkecil.

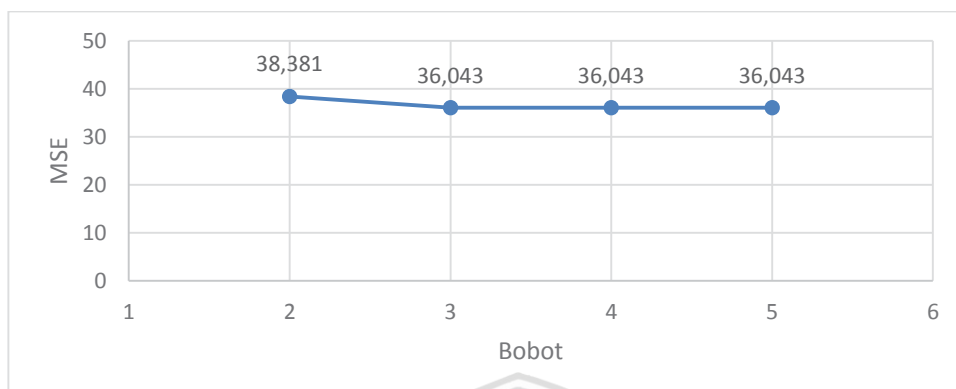
6.2.3 Pengujian pengaruh bobot terhadap nilai MSE

Pengujian variabel bobot ini menggunakan nilai yang berbeda yaitu dimulai dari 2 sampai 5 dengan penambahan sebesar 1 pada setiap tahap ujiannya. Sedangkan variabel lain bernilai tetap yaitu 36 data latih, 36 data uji, dan iterasi maksimal bernilai 3. Hasil pengujian bobot dapat dilihat pada Tabel 6.3.

Tabel 6.3 Pengujian bobot

Bobot	Pengujian ke- (MSE)					Rata-Rata	
	1	2	3	4	5	MSE	Lama Komputasi (milidetik)
2	40.438	35.180	39.759	38.063	38.463	38.381	15
3	36.043	36.043	36.043	36.043	36.043	36.043	15
4	36.043	36.043	36.043	36.043	36.043	36.043	15
5	36.043	36.043	36.043	36.043	36.043	36.043	15

Dari hasil pengujian yang ditunjukkan pada Tabel 6.3, terjadi perubahan MSE ketika bobot bertambah. Perubahan nilai MSE pada pengujian bobot dapat dilihat dalam bentuk grafik yang ditunjukkan Gambar 6.3.



Gambar 6.3 Grafik hasil pengujian bobot terhadap MSE

Dari grafik menunjukkan nilai MSE terhadap bobot memiliki *trendline* menurun dan bernilai tetap yaitu 36,043 pada bobot bernilai 3 ke atas. Semakin besar jumlah bobot yang digunakan, semakin kecil nilai MSE yang dihasilkan. Dari grafik 6.3 didapatkan nilai MSE terbaik menggunakan bobot dengan nilai 3.

6.2.4 Pengujian pengaruh iterasi maksimal terhadap nilai MSE

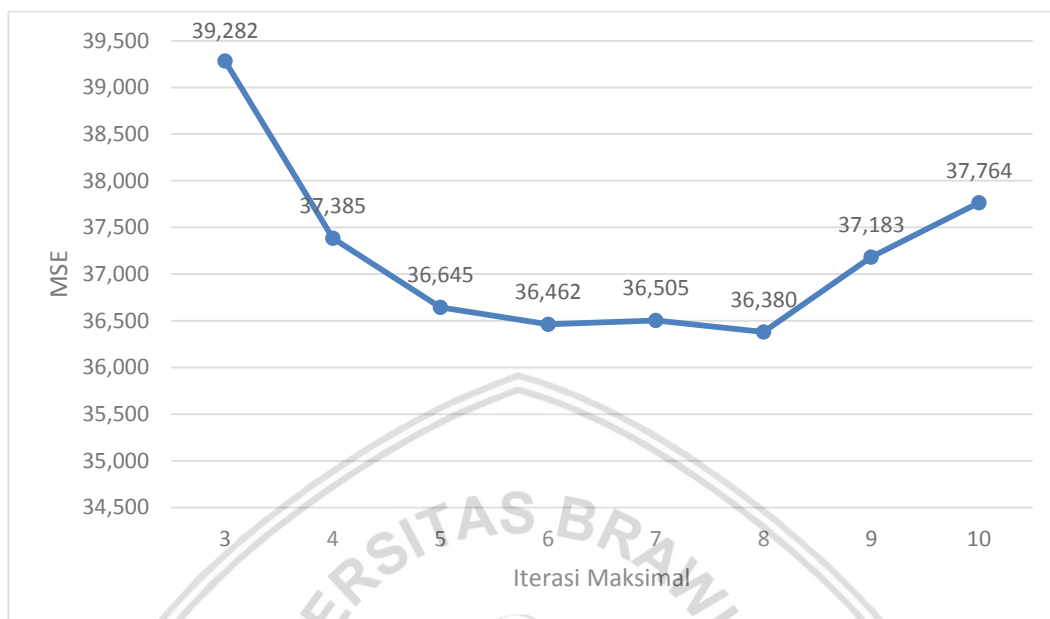
Pengujian variabel iterasi maksimal ini menggunakan nilai yang berbeda yaitu dimulai dari 3 sampai 10 dengan penambahan sebesar 1 pada setiap tahap ujiannya. Sedangkan variabel lain bernilai tetap yaitu 36 data latih, 36 data uji, dan bobot bernilai 2. Hasil pengujian bobot dapat dilihat pada Tabel 6.4.

Tabel 6.4 Pengujian Iterasi maksimal

Iterasi Maksimal	Pengujian ke- (MSE)					Rata-Rata	
	1	2	3	4	5	MSE	Lama Komputasi (millisecond)
3	39.288	39.667	41.872	38.024	37.560	39.2822	16
4	33.475	39.884	38.432	35.261	39.872	37.3848	15
5	36.980	39.761	34.173	35.686	36.626	36.6452	15
6	35.622	35.377	32.845	39.640	38.828	36.4624	15
7	36.881	39.660	34.074	35.085	36.823	36.5046	15
8	35.562	35.277	32.795	39.550	38.718	36.3804	18
9	33.352	39.482	38.321	35.140	39.621	37.1832	18
10	39.974	39.987	38.422	36.863	33.576	37.7644	18



Dari hasil pengujian yang ditunjukkan pada Tabel 6.4, terjadi perubahan MSE ketika iterasi maksimal bertambah. Perubahan nilai MSE pada pengujian bobot dapat dilihat dalam bentuk grafik yang ditunjukkan Gambar 6.4.



Gambar 6.4 Grafik hasil pengujian iterasi maksimal terhadap MSE

Dari grafik menunjukkan nilai MSE terhadap iterasi maksimal memiliki *trendline* menurun dan mendekati konvergen pada data ke 6 sampai 8, sedangkan pada iterasi ke 9 sampai 10 menunjukkan *trendline* naik. Dari grafik 6.3 didapatkan nilai MSE terbaik menggunakan iterasi maksimal dengan nilai 8 karena iterasi selanjutnya menunjukkan *trendline* nilai MSE bertambah.

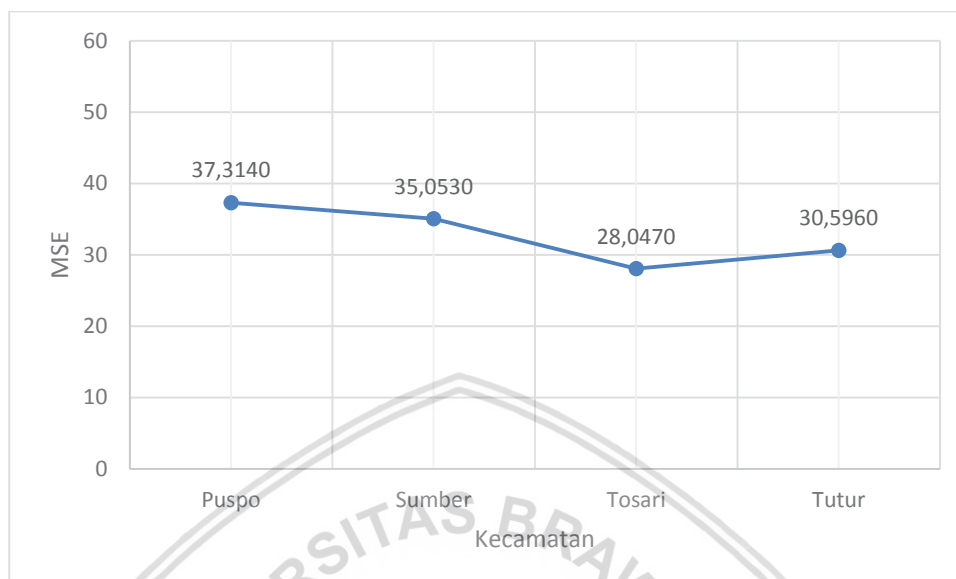
6.2.5 Pengujian validasi sistem

Pengujian validasi sistem dilakukan sebanyak 5 kali uji coba dengan 4 data yang berbeda menggunakan kombinasi dari variable terbaik yang telah didapatkan dari pengujian sebelumnya. Keempat data tersebut adalah data tahun 2010 sampai 2014 dari kecamatan Puspo, Sumber, Tosari, dan Tukur. Hasil pengujian validasi sistem dapat dilihat pada Tabel 6.5.

Tabel 6.5 Pengujian validasi sistem

Kecamatan	Uji ke-					Rata-Rata	
	1	2	3	4	5	MSE	Lama Komputasi (milisecond)
Puspo	37.314	37.314	37.314	37.314	37.314	37,314	38
Sumber	35.053	35.053	35.053	35.053	35.053	35.053	39
Tosari	28.047	28.047	28.047	28.047	28.047	28.047	37
Tukur	30.596	30.596	30.596	30.596	30.596	30.596	36

Dari hasil pengujian yang ditunjukkan pada Tabel 6.5, terjadi perubahan MSE ketika data yang digunakan berbeda. Perubahan nilai MSE pada pengujian bobot dapat dilihat dalam bentuk grafik yang ditunjukkan Gambar 6.5.



Gambar 6.5 Grafik hasil pengujian validasi sistem

Dari grafik pada Gambar 6.5 menunjukkan nilai MSE berbeda-beda untuk setiap kecamatan. Perbedaan MSE yang ditunjukkan grafik cukup stabil dengan jarak tidak terlalu jauh walaupun menggunakan data yang berbeda. Oleh karena itu sistem prediksi curah hujan menggunakan *Adaptive-Expectation based Multi-Attribute Fuzzy Time Series* sudah cukup baik.

BAB 7 PENUTUP

Bab penutup terdiri dari sub bab kesimpulan dan saran. Sub bab kesimpulan berisi hasil penelitian yang dilakukan dengan menjawab semua rumusan masalah yang telah ditetapkan pada bab satu. Sedangkan saran berisi tentang pertimbangan untuk tujuan pengembangan lebih lanjut terkait dengan penelitian ini.

7.1 Kesimpulan

Kesimpulan yang dapat diambil dari penelitian prediksi curah hujan menggunakan metode *Adaptive-Expectation based Multi-Attribute Fuzzy Time Series* adalah sebagai berikut.

1. Berdasarkan hasil implementasi prediksi curah hujan menggunakan metode *Adaptive-Expectation based Multi-Attribute Fuzzy Time Series* menggunakan urutan langkah sebagai berikut, langkah pertama membentuk *subinterval* dengan mencari nilai *Universe of Discourse*, standar deviasi, perhitungan pusat kluster, dan menentukan batas subinterval, langkah kedua yaitu membentuk himpunan fuzzy, langkah ketiga yaitu proses fuzzifikasi data, langkah keempat yaitu membentuk *Fuzzy Logic Relationship*, langkah kelima yaitu mencari nilai alpha terbaik dari prediksi data latih, langkah keenam yaitu defuzzifikasi data uji, dan langkah ketujuh yaitu memperbaiki prediksi data uji menggunakan nilai alpha *Adaptive-Expectation*.
2. Berdasarkan hasil pengujian prediksi curah hujan menggunakan metode *Adaptive-Expectation based Multi-Attribute Fuzzy Time Series*, maka disimpulkan bahwa sistem cukup baik untuk memprediksi curah hujan tahun 2014 di empat kecamatan yaitu Puspo dengan nilai MSE (37.3140), Sumber dengan nilai MSE (35.0530), Tosari dengan nilai MSE (28.0470), dan Tukur dengan nilai MSE (30.5960).

7.2 Saran

Prediksi curah hujan menggunakan *Adaptive-Expectation based Multi-Attribute Fuzzy Time Series* pada penelitian ini masih memiliki ruang untuk pengembangan ataupun variasi untuk penelitian selanjutnya yaitu sebagai berikut

1. Menambahkan atribut pendukung yang mempengaruhi hasil prediksi curah hujan. Pada penelitian ini menggunakan empat atribut curah hujan, suhu, kelembaban udara, dan lama penyinaran matahari.
2. Menggunakan jumlah data set yang berbeda dan data dari daerah yang berbeda.



DAFTAR PUSTAKA

- Bezdek, J. C., 1981. *Pattern Recognition with Fuzzy Objective Function Algorithms*. New York: Plenum Press.
- Billah, T., 2013. *Outlook Komoditi Kentang*. Jakarta: Pusat Data dan Sistem Informasi Pertanian.
- Cheng, C. H., Cheng, G. W. & Wang, J. W., 2008. Expert Systems with Applications. *Multi-attribute fuzzy time series method based on fuzzy clustering*, 34(2), pp. 1235-1242.
- Chen, S. M., 1996. Fuzzy Sets and System. *Forecasting Enrollments Based On Fuzzy Time Series*, Volume 81, pp. 311-319.
- Fauziah, N., Wahyuningsih, S. & Nasution, Y. N., 2016. Peramalan Menggunakan Fuzzy Time Series Chen (Studi Kasus: Curah Hujan Kota Samarinda). *Statistika*, pp. 52-61.
- Iriani, A., Mahmudy, W. F. & Nisak, C. S., 2015. GSTAR-SUR Model for Rainfall Forecasting in Tengger Region, East Java. *1st Int. Conf. Pure Appl. Res. Univ. Muhammadiyah Malang*, 21-22 August. pp. 1-8.
- Kmenta, J., 1971. *Elements of econometrics*. New York: Macmillan.
- Kusumadewi, S. & Purnomo, H., 2010. *Aplikasi Logika Fuzzy untuk Pendukung Keputusan*. Yogyakarta: Graha Ilmu Yogyakarta.
- Liu, J. W., Chen, T. L., Cheng, C. H. & Chen, Y. H., 2010. Computers & Mathematics with Applications. *Adaptive-Expectation Based Multi-Attribute FTS Model For Forecasting TAIEX*, 59(2), pp. 795-802.
- Mardawilis & Ritonga, E., 2016. Pengaruh Curah Hujan Terhadap Produksi Tanaman Pangan. *The Effect Of Rainfall On Food Crop Production In The Regency Of*, p. 289.
- Miller, G., 1956. Psychological Review. *The Magical Number Seven, Plus Or Minus Two: Some Limits On Our Capacity For Processing Information*, Volume 63, pp. 81-97.
- Song, Q. & Chissom, B. S., 1993. Fuzzy Sets and Systems. *Forecasting enrollments with fuzzy time series—Part I*, 54(1), pp. 1-9.
- Suciantini, 2015. Interaksi iklim (curah hujan) terhadap produksi tanaman pangan di Kabupaten Pacitan. *Relationship between climate (rainfall) and crop production in Pacitan*, Volume 1, pp. 358-365.
- Suryaningrum, K. M. & Wijaya, S. P., 2015. Analisa Dan Penerapan Metode Single Exponential Smoothing Untuk Prediksi Penjualan Pada Periode Tertentu. *Prosiding SNATIF*, pp. 259-266.
- Utomo, M. C. C., Mahmudy, W. F. & Anam, S., 2017. Optimasi Metode Fuzzy Neural Networks Untuk Prakiraan Trend Curah Hujan Di Area Puspo, Sumber, Tosari, Dan Tutur - Jawa Timur. *Jurnal Teknologi Informasi dan Ilmu Komputer*,

Volume vol. 4, no. 10. Tersedia pada:<<http://jtiik.ub.ac.id/index.php/jtiik/article/view/299>>. Tanggal akses: 31 mei 2018.

Wahyuni, I. & Mahmudy, W. F., 2017. Rainfall prediction using hybrid adaptive neuro-fuzzy inference system (ANFIS) and genetic algorithm. *Journal of Telecommunication, Electronic and Computer Engineering*, Volume vol. 9, no. 2-8. Tersedia pada: <<http://journal.utem.edu.my/index.php/jtec/article/view/2627>>. Tanggal akses: 31 Mei 2018.

Wei, L. Y., 2013. Economic Modelling. *A hybrid model based on ANFIS and adaptive expectation genetic algorithm to forecast TAIEX*, Volume 33, pp. 893-899.

Yu, H. K., 2005. *Physica A: Statistical Mechanics and its Applications. Weighted fuzzy time series models for TAIEX forecasting*, 349(3-4), pp. 609-624.

