

PENGCLUSTERAN DATA CURAH HUJAN KOTA BENGKULU MENGGUNAKAN *FUZZY CLUSTERING ALGORITMA MIXTURE*

Herlina Latipa Sari¹, Dimas Aulia Trianggana²

^{1,2}Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Dehasen Bengkulu
Jalan Meranti Raya No. 32 Sawah Lebar Kota Bengkulu website : <http://unived.ac.id> Telp (0736) 22027

¹herlinalatipasari@mail.com

²ezdroez@gmail.com

Abstrak: Penclusteran Data curah hujan Kota Bengkulu dengan *Fuzzy Clustering* menggunakan Algoritma *Mixture* dan mengevaluasi performasi Algoritma *Mixture* dalam menghasilkan tingkat keakuratan lokasi prakiraan curah hujan bulanan di Stasiun Klimatologi Pulau Baai Bengkulu. Data curah hujan Kota Bengkulu di analisis menggunakan algoritma *Mixture* dimana pengujian dilakukan menggunakan software matlab dan software SOCR. Dalam mengelompokkan data curah hujan Kota Bengkulu, Algoritma Gaussian Mixture Modelling berdasarkan dengan distribusi statistik yang dapat mengetahui overlapping data dalam sebuah cluster. Dari hasil pengujian Algoritma *Mixtures* untuk mendapatkan alternatif yang terbaik, dapat digunakan untuk memecahkan masalah dalam mengelompokkan data yang memiliki kesamaan jumlah data curah hujan yang sama atau mendekati, sehingga dapat digunakan sebagai pendukung pengambilan keputusan dalam mengelompokkan data. *Mixture Modelling* dalam mengelompokkan data curah hujan Kota Bengkulu mempunyai kemampuan untuk mendeteksi keberadaan suatu cluster yang *overlap* dengan cluster yang lain.

Kata Kunci: *Fuzzy Clustering, Algoritma Mixture.*

Abstract: *Grouping rainfall data Bengkulu City Clustering using Fuzzy Algorithm and evaluate performasi Mixture Mixture algorithm to generate the level of location accuracy in forecasting monthly rainfall Climatological Station Island Baai Bengkulu. Bengkulu city rainfall data were analyzed using Mixture algorithm where the testing is done using matlab software and software SOCR. In classifying rainfall data Bengkulu, Gaussian Mixture Modeling Algorithms based on the statistical distribution that can determine overlapping data in a cluster. Mixtures of the testing algorithm to get the best alternative, it can be used to solve problems in classifying the data that have the same amount of rainfall data are the same or close, so it can be used as decision support in classifying data. Mixture Modelling the rainfall data grouping City Bengkulu has the ability to detect the presence of a cluster overlap with other clusters.*

Keywords: *Fuzzy Clustering, Algoritma Mixture.*

I. PENDAHULUAN

Konvensi Internasional di seluruh dunia menyatakan bahwa curah hujan mempunyai peran yang sangat penting, untuk mendukung sektor

penerbangan, diberikan layanan jasa meteorologi untuk meningkatkan keamanan dan keselamatan penerbangan. Indonesia juga telah menerapkan hal ini diantaranya dalam peraturan Pemerintah No. 3 tahun 2001 tentang keselamatan penerbangan.

Berdasarkan data prakiraan curah hujan tahun 2009 pada Kota Bengkulu yang dilakukan pada empat pos pengamatan yang digunakan dalam proses prakiraan curah hujan yang terjadi dari masing-masing pos pengamatan didapat prakiraan curah hujan dengan rata-rata 200mm – 400 mm masuk dalam pengelompokan sifat prakiraan curah hujan normal.

Stasiun Klimatologi Pulau Baai Bengkulu adalah salah satu Unit Pelaksana Teknis Badan Meteorologi Klimatologi dan Geofisika yang mempunyai tugas pokok untuk melaksanakan pengamatan, pengolahan, dan penyebaran data unsur-unsur cuaca/iklim (angin, hujan, suhu,

tekanan udara, *visibility* dan lainnya) sehingga data curah hujan dapat dikelompokkan disesuaikan dengan kebutuhan masyarakat.

Metode fuzzy clustering, telah banyak diaplikasikan untuk mengelompokkan suatu data berdasarkan kesamaan/kemiripan yang dimiliki oleh suatu wilayah. Terdapat berbagai macam teknik fuzzy clustering misalnya Fuzzy Mixture Modelling. *Fuzzy Mixture Modelling* merupakan salah satu jenis data clustering dimana dalam pemodelannya, data dalam suatu kelompok diasumsikan terdistribusi sesuai dengan salah satu jenis distribusi statistik yang ada.

Beberapa contoh pemanfaatan Algoritma Clustering diantaranya dalam peranan pengelompokan samar dalam prediksi kekeringan di Indonesia [2], memanfaatkan klustering tingkat pemakaian pompa air tanah menggunakan model fuzzy clustering [7] juga memanfaatkan pembagian kelas peserta kuliah berdasarkan fuzzy clustering dan fuzzy C-Means untuk clustering data (Studi kasus : data performance mengajar dosen).

Tujuan penggunaan algoritma *Mixture* adalah untuk langkah awal mengelompokkan record-record data yang dianalisis sehingga terkelompok ke dalam interval-interval kelas yang lebih sedikit yang diharapkan dapat mempertinggi tingkat akurasi yang dihasilkan [1]. Dari data yang dihasilkan dari kedua algoritma ini nantinya akan dibandingkan sehingga dapat disimpulkan dari kedua algoritma ini akan dapat satu algoritma dimana hasilnya nantinya mendekati akurasi data.

Oleh karenanya akan sangat menarik bagaimana menggunakan algoritma *Mixture* sehingga nantinya dapat dijadikan sebagai salah satu teknik dalam pengclusteran data curah hujan, sehingga penulis berniat untuk mengangkat topik ini dalam judul "*Pengclusteran Data Curah*

Hujan Kota Bengkulu Menggunakan Fuzzy Clustering Algoritma Mixture" yang diharapkan dapat menyelesaikan permasalahan dalam pengelompokkan atau penclustering berdasarkan dengan data hujan dan sifat hujan.

II. LANDASAN TEORI

A. Prakiraan Curah Hujan

Prakiraan adalah suatu proses yang memperkirakan sesuatu secara sistematis tentang sesuatu yang paling mungkin terjadi di masa depan berdasarkan informasi di masa lalu dan sekarang yang dimilikinya agar kesalahan (selisih antara hasil pendugaan dengan kenyataannya) dapat diperkecil [3].

Curah Hujan (mm) merupakan ketinggian air hujan yang jatuh pada tempat yang datar dengan asumsi tidak menguap, tidak meresap dan tidak mengalir. Curah hujan 1 (satu) mm adalah air hujan setinggi 1 (satu) mm yang jatuh (tertampung) pada tempat yang datar seluas 1 m² dengan asumsi tidak ada yang menguap, mengalir dan meresap [3]. Curah hujan kumulatif 1 (satu) bulan adalah jumlah curah hujan yang terkumpul selama 28 atau 29 hari untuk bulan februari dan 30 atau 31 hari untuk bulan-bulan lainnya. Sifat hujan merupakan perbandingan antara jumlah curah hujan selama rentang waktu yang ditetapkan (satu periode musim kemarau) dengan jumlah curah hujan normalnya (rata-rata selama 30 tahun (1971-2000) pada bulan dan tempat yang sama. Sifat hujan dibagi menjadi 3 (tiga) kategori, yaitu:

1. Atas Normal (AN): jika nilai curah hujan lebih dari 115% terhadap rata-ratanya.
2. Normal (N): jika nilai curah hujan antara 85%-115% terhadap rata-ratanya.
3. Bawah Normal (BN): jika nilai curah hujan kurang dari 85% terhadap rata-ratanya.

B. Fuzzy Clustering

Logika fuzzy pertama kali diperkenalkan oleh Lotfi Zadeh (UC Barkeley) pada tahun 1965, sebagai suatu cara matematis untuk menyatakan keadaan yang tidak menentu (samara) dalam kehidupan sehari-hari. Ide ini didasarkan pada kenyataan bahwa didunia ini suatu kondisi sering di interpretasikan dengan ketidakpastian atau tidak memiliki ketetapan secara kuantitatif, misalnya: panas, dingin dan cepat. Dengan logika fuzzy, kita dapat menyatakan informasi-informasi yang samara tersebut (kurang spesifik), kemudian memanipulasinya, dan menarik suatu kesimpulan dari informasi tersebut [4].

Logika fuzzy telah banyak diterapkan dalam proses pengendalian suatu sistem, hal itu karena fuzzy dapat menyimpan dan mengaplikasikan pengetahuan-pengetahuan tentang suatu masalah. Pengetahuan-pengetahuan ini terdapat pada aturan-aturan logika fuzzy. Aturan-aturan inilah yang dijadikan. Implementasi logika fuzzy dalam sistem kendali [5]. Kemunculan fuzzy clustering dilatarbelakangi adanya masalah curse of dimensionality, yaitu jumlah rule yang begitu cepat membesar dengan bertambahnya jumlah variabel input FIS. Jumlah rule yang terlalu besar jelas akan membuat komputasi menjadi berat dan juga optimasi parameter-parameter rule menjadi lebih sulit [6].

Dengan fuzzy clustering, suatu data input output akan dikelompokkan dalam beberapa grup atau cluster. Informasi cluster ini akan membantu dalam FIS tipe Sugeno terbaik yang bisa memodelkan kelakuan hubungan data input-output dengan jumlah rule minimum. Definisi sebuah rule diasosiasikan dengan suatu cluster data. Pembangunan FIS demikian bisa secara otomatis dilakukan menggunakan fungsi **genfis2**. Fuzzy clustering adalah salah satu teknik untuk

menentukan cluster optimal dalam suatu ruang vektor yang didasarkan pada bentuk normal Euclidian untuk jarak antar vektor.

1) Partisi Klasik (hard partition): Konsep partisi menjadi bagian yang sangat penting bagi proses pengclusteran. Tujuan proses pengclusteran pada partisi klasik adalah membagi himpunan data ke x ke dalam c kelompok (grup atau kelas) dengan asumsi bahwa c diketahui, partisi klasik x dapat didefinisikan sebagai suatu keluarga dari himpunan bagian- himpunan bagian $(A_i \mid 1 \leq i \leq c)$ $A_i \mid 1 \leq i \leq c \} \subset P(X), P(X)$ adalah power set dari X, dengan properti sebagai berikut :

$$\bigcup_{i=1}^c A_i = X \dots\dots\dots(1)$$

$$A_i \cap A_j = \phi; 1 \leq i \neq j \leq c \dots\dots\dots(2)$$

$$\phi \subset A_j \subset X; 1 \leq i \leq c \dots\dots\dots(3)$$

Persamaan (1) menunjukkan bahwa union himpunan bagian A_i berisi semua data. Himpunan bagian – himpunan bagian harus bersifat disjoint (2), dan tidak boleh ada yang berupa himpunan kosong (3). Dalam bentuk fungsi keanggotaan, suatu partisi dapat dipresentasikan sebagai matriks partisi $U = [\mu_{ik}]_{c \times n}$. Baris ke-i pada matriks tersebut berisi nilai keanggotaan μ_i pada himpunan bagian A_i . Berdasarkan persamaan (1) maka elemen-elemen pada matriks U harus memenuhi kondisi sebagai berikut :

$$\mu_{ik} \in \{0,1\}; 1 \leq i \leq c; 1 \leq k \leq n \dots\dots\dots(4)$$

$$\sum_{i=1}^c \mu_{ik} = 1; 1 \leq k \leq n \dots\dots\dots(5)$$

$$0 < \sum_{k=1}^n \mu_{ik} < n; 1 \leq i \leq c \dots\dots\dots(6)$$

Semua kemungkinan partisi matriks X disebut dengan hard partition space, yang definisikan berikut :

$$M_{fc} = \left\{ U \in \mathcal{R}^{cn} \mid \mu_{ik} \in (0,1], \forall i,k; \sum_{i=1}^c \mu_{ik} = 1, \forall k; 0 < \sum_{k=1}^n \mu_{ik} < n, \forall i \right\} \dots(7)$$

2) *Partisi Fuzzy (fuzzy partition)*: Jika pada partisi klasik, suatu data secara eksklusif menjadi anggota hanya pada satu *cluster* saja, tidak demikian halnya dengan partisi fuzzy. Pada partisi fuzzy, nilai keanggotaan suatu data pada suatu cluster μ_{ik} terletak pada interval $[0,1]$. Matriks partisi pada partisi fuzzy memenuhi kondisi sebagai berikut :

$$\mu_{ik} \in [0,1]; 1 \leq i \leq c; 1 \leq k \leq n \dots\dots\dots(8)$$

$$\sum_{i=1}^c \mu_{ik} = 1; 1 \leq k \leq n \dots\dots\dots(9)$$

$$0 < \sum_{k=1}^n \mu_{ik} < n; 1 \leq i \leq c \dots\dots\dots(10)$$

Baris ke-*i* pada matriks partisi U berisi nilai keanggotaan data pada himpunan bagian fuzzy A_i . Jumlah derajat keanggotaan setiap data pada semua cluster (jumlah setiap kolom) bernilai 1 (persamaan 8). Semua kemungkinan partisi dari matriks X disebut dengan *fuzzy partitioning space*, yang didefinisikan sebagai :

$$M_{hc} = \left\{ U \in \mathcal{R}^{cn} \mid \mu_{ik} \in (0,1], \forall i,k; \sum_{i=1}^c \mu_{ik} = 1, \forall k; 0 < \sum_{k=1}^n \mu_{ik} < n, \forall i \right\} \dots\dots(11)$$

3) *Partisi Possibilistik (possibilistic partition)*: Tidak seperti halnya kedua partisi diatas, pada partisi possibilistic jumlah nilai keanggotaan suatu data pada semua cluster tidak harus 1, namun untuk menjamin suatu data menjadi anggota dari (paling tidak) suatu cluster, maka diharuskan ada nilai keanggotaan yang bernilai lebih dari 0.

Matriks partisi pada partisi fuzzy memenuhi kondisi sebagai berikut:

$$\mu_{ik} \in [0,1]; 1 \leq i \leq c; 1 \leq k \leq n \dots\dots\dots(12)$$

$$\exists i, \mu_{ik} > 0; \forall k \dots\dots\dots(13)$$

$$0 < \sum_{k=1}^n \mu_{ik} < n; 1 \leq i \leq c \dots\dots\dots(14)$$

Semua kemungkinan partisi matriksX disebut dengan *Possibilistic partitioning space*, yang didefinisikan sebagai :

$$M_{pc} = \left\{ U \in \mathcal{R}^{cn} \mid \mu_{ik} \in (0,1], \forall i,k; \exists i, \mu_{ik} > 0; \forall k; 0 < \sum_{k=1}^n \mu_{ik} < n, \forall i \right\} \dots\dots(15)$$

C. *Algoritma Mixture*

Algoritma Mixture merupakan salah satu jenis data clustering dimana dalam pemodelannya, data dalam satu kelompok diasumsikan terdistribusi sesuai dengan salah satu jenis distribusi sesuai dengan salah satu jenis distribusi statistik yang ada. *Algoritma mixture* merupakan metode yang mempunyai optimasi yang sama dengan algoritma *C-Means* melalui proses optimization dan Maximization. Berbeda dengan algoritma *Hard C-Means* dan *Fuzzy C-Means*, perbandingan jumlah data yang tercakup didalam masing-masing cluster juga mempengaruhi hasil akhir dari suatu proses data clustering. Perbandingan jumlah data yang terdapat didalam masing-masing cluster sering diistilahkan dengan nama *relative abundance* [1].

Distribusi statistik yang paling sering digunakan dalam data clustering menggunakan metode *mixture* adalah distribusi *Gaussian/Normal*. Disamping karena kemudahan penurunan berbagai rumus yang diperlukan, kecenderungan umum yang ada pada saat melakukan observasi adalah bahwa data yang didapatkan umumnya dalam keadaan terdistribusi secara normal. Berbeda dengan *C-Means, distance*

space yang digunakan didalam *mixture* berbasis distribusi *Gaussian/Normal* adalah *Mahalanobis distance space* yang sering dikaitkan dengan distribusi *multivariate Gaussian* menghitung jarak dengan rumus sebagai berikut :

$$D_{Mahalanobis}(x_2, x_1) = \|x_2 - x_1\|_{mahalanobis} = (x_2 - x_1)^T \Sigma^{-1} (x_2 - x_1) \quad (16)$$

dimana:

$(x_2 - x_1)^T$: *Transpose* dari sebuah matriks

$(\Sigma)^{-1}$: *Inverse* dari sebuah matriks

Σ : *Variance Covariance* matriks

Proses pengalokasian kembali data ke masing-masing cluster menggunakan Algoritma *mixture modeling*. Perbedaannya terletak pada cara penghitungan nilai keanggotaan (*Fuzzy C-Means*) dan nilai probabilitas data (*Mixture Modelling*) untuk menjadi anggota suatu cluster. Penghitungan nilai pada suatu cluster dikalikan dengan *relative abundance* dari cluster yang bersangkutan seperti berikut ini

$$P_{ik} = \pi_i \times f_i(x_k | \theta_i) \dots \dots \dots (17)$$

dimana:

p_{ik} : Probabilitas data ke-*k* menjadi anggota cluster ke-*i*

π_i : *Relative abundances* cluster ke-*i*

$f_i(x_k | \theta_i)$: Distribusi probabilitas cluster ke-*i*

θ_i : Parameter yang tercakup di dalam distribusi yang diasumsikan untuk cluster ke-*i*

Berbagai algoritma memungkinkan untuk digunakan dalam memecahkan proses optimasi *mixture modelling* termasuk di antaranya *random search, simulated annealing, Markov Chain Monte Carlo (MCMC)* maupun algoritma genetika.. Algoritma yang digunakan adalah sebagai berikut :

a. Tentukan jumlah cluster

b. Alokasikan data secara *random* ke masing-masing cluster yang telah ditentukan

1. Hitung *means* (sama dengan *centroid* pada *K-Means*) dari masing-masing cluster
2. Hitung standar deviasi/*variance covariance* dari masing-masing cluster
3. Hitung nilai probabilitas masing-masing data ke masing-masing cluster
4. Kembali ke Step b.1, apabila perubahan nilai probabilitas masih di atas nilai *threshold* yang ditentukan, atau apabila perubahan pada nilai *centroid* masih di atas nilai *threshold* yang ditentukan, atau apabila perubahan pada nilai *objective function* masih di atas nilai *threshold* yang ditentukan.

c. Kembali ke Step a. apabila masih ada jumlah cluster yang ingin dianalisa.

Dengan asumsi bahwa data terdistribusi secara normal, *means* cluster ke-*i*, μ_i , dihitung dengan menggunakan rumus sama dengan metode *Fuzzy K-Means* dengan u_{ik} merupakan nilai probabilitas data tersebut termasuk di dalam cluster ke-*i*. Sedangkan standar deviasi/*variance covariance* cluster ke-*i*, $\sigma_i \Sigma_i$, dihitung dengan menggunakan rumus sebagai berikut:

$$\sigma_i = \sqrt{\frac{\sum_{k=1}^N (x_k - \mu_i)^2}{N - 1}} \dots \dots \dots (18)$$

$$\Sigma_i = \frac{\sum_{k=1}^N (x_k - \mu_i)(x_k - \mu_i)^T}{N - 1} \quad (19)$$

dimana:

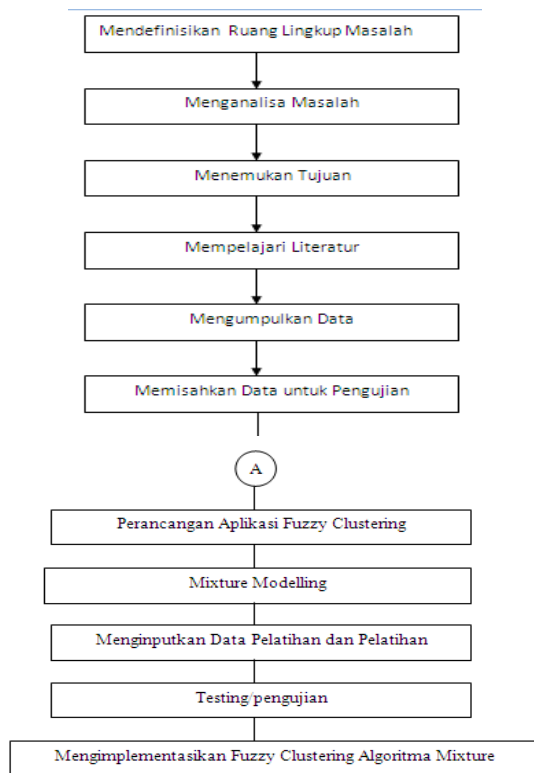
N : Jumlah data

μ_i : *Means* cluster ke-*i*

sedangkan untuk menghitung nilai probabilitas data ke-*k* ke cluster ke-*i* digunakan rumus penghitungan probabilitas.

III. METODOLOGI

Kerangka kerja (*frame work*) merupakan langkah-langkah yang dilakukan dalam penulisan. Adapun kerangka kerja yang digunakan dalam penulisan ini adalah seperti gambar 1.



Gambar 1. Kerangka kerja penelitian

Berdasarkan kerangka kerja pada gambar 1 maka masing-masing langkahnya diuraikan sebagai berikut :

1. Definisi Ruang lingkup masalah

Ruang lingkup masalah yang akan diteliti harus ditentukan terlebih dahulu karena tanpa mampu menentukan serta mendefinisikan rumusan dan batasan masalah yang akan diteliti, maka tidak akan pernah didapat solusi yang terbaik dari masalah tersebut. Jadi langkah ini adalah langkah awal yang terpenting dalam penulisan ini.

2. Analisa Masalah

Analisa masalah adalah langkah yang dilakukan untuk memahami masalah yang telah ditentukan ruang lingkup atau batasannya. Dengan menganalisa masalah yang telah ditentukan tersebut, maka diharapkan masalah dapat dipahami dengan baik.

3. Menentukan Tujuan

Berdasarkan pemahaman dari masala, maka ditentukan tujuan yang akan dicapai dari penelitian ini. Pada tujuan ini ditentukan target yang akan dicapai, terutama yang dapat mengatasi masalah-masalah yang ada.

4. Mempelajari Literatur

Untuk mencapai tujuan maka dipelajari beberapa literature-;iteratur yang dapat dijadikan dasar atau rujukan dalam penelitian ini.

5. Mengumpulkan Data

Dalam pengumpulan data dilakukan observasi yaitu pengamatan secara langsung ditempat penelitian sehingga permasalahan yang ada dapat diketahui dengan jelas. Kemudian dilakukan wawancara yang bertujuan untuk mendapatkan informasi atau data yang dibutuhkan. Selain itu, juga dilakukan studi kepustakaan yaitu dengan membaca buku-buku yang menunjang dalam melakukan penganalisaan terhadap data dan informasi yang didapat.

6. Memisahkan Data untuk pelatihan dan pengujian

Setelah dilakukan pengumpulan data, maka selanjutnya data yang diperoleh dipisahkan menjadi dua bagian. Bagian pertama dipergunakan untuk melatih Fuzzy Clustering dan bagian kedua digunakan untuk menguji algoritma Mixture.

7. Perancangan Aplikasi Fuzzy Clustering

Bagian ini merupakan perancangan aplikasi fuzzy clustering dimana dalam penelitian ini terdapat dua algoritma yang digunakan untuk mengujikan data curah hujan menggunakan algoritma *mixture*..

Dimana untuk algoritma *Mixture* melakukan urutan algoritma yaitu rumus (17),(18) dan (19).

8. Mengumpulkan data untuk pelatihan

Langkah ini dilakukan sebelum melakukan pelatihan agar dalam melakukan pelatihan data yang di gunakan lengkap.

9. Pelatihan

Langkah ini dilakukan untuk melatih algoritma *Mixture* untuk menemukan akurasi data sehingga terjadi pengelompokan atau pengklusteran data curah hujan.

10. Pengujian / Testing

Bagian ini dilakukan untuk menguji dari kedua metode algoritma yang digunakan manakah yang menghasilkan cluster yang akurasi data menjadi cluster sesuai dengan cluster yang ditentukan.

11. Mengimplementasikan Fuzzy Clustering

Setelah fuzzy clustering dengan algoritma fuzzy c-means dan *mixture* diuji maka selanjutnya Fuzzy clustering dengan kedua algoritma tersebut diimplementasikan sebagai sebuah sistem yang siap untuk menyelesaikan masalah menggunakan software Matlab 6.5.

hambatan-hambatan yang terjadi dan kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan suatu perbaikan. Data yang digunakan adalah data curah hujan bulanan Kota Bengkulu selama lima tahun dimana data berasal dari alat pengukur curah hujan Ambrometer atau Type Hellman. Program bantu yang digunakan dalam pengclusteran data curah hujan Kota Bengkulu menggunakan Algoritma *Mixture* adalah Matlab 6.5 yang mana dalam program Matlab versi 6.5 terdapat Toolbox Fuzzy Cluster yang mempunyai kemampuan dalam mengelompokkan data sesuai dengan kelompok data yang telah ditentukan.

B. Langkah Komputasi Dalam Pengclusteran Data Curah Hujan Kota Bengkulu Menggunakan Algoritma Mixture.

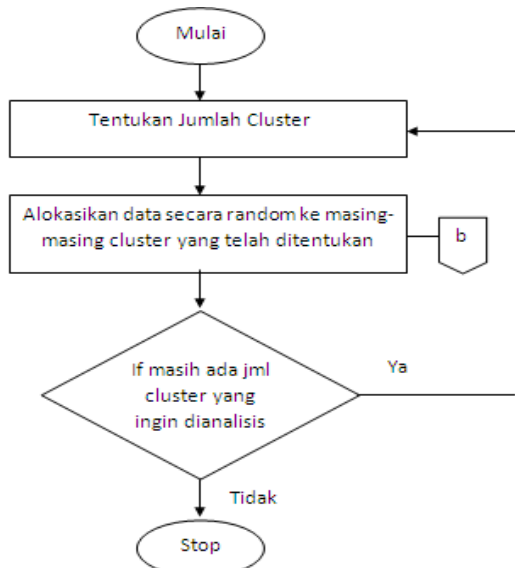
Langkah-langkah komputasi dalam pengclusteran data curah hujan ini membandingkan dua algoritma dimana masing-masing algoritma memiliki langkah-langkah yang harus ditelusuri sehingga dapat dibandingkan, yaitu :

1) *Langkah-langkah Komputasi dalam Algoritma Mixture:* Langkah-langkah dalam pengclusteran data curah hujan dengan algoritma *Mixture* ini terdiri dari beberapa langkah yaitu:

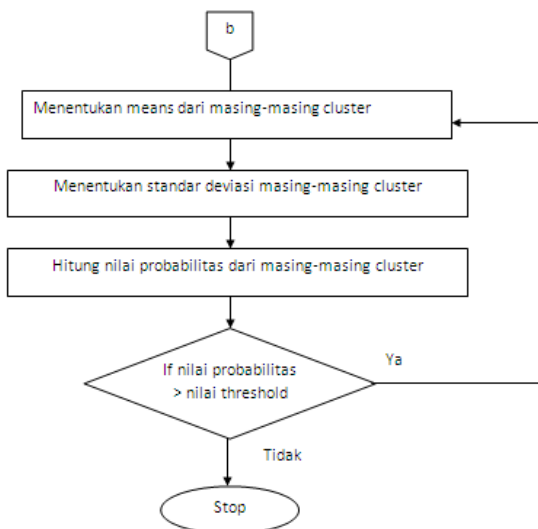
IV. ANALISA DAN PERANCANGAN SISTEM

A. Analisa Kebutuhan Sistem

Analisa sistem adalah penguraian dari suatu sistem utuh ke dalam bagian komponen-komponen dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan-permasalahan,



Gambar 2. Langkah-langkah Algoritma Mixture



Gambar 3. Langkah-langkah Algoritma Mixture dalam Menentukan Means

cluster, banyaknya iterasi, toleransi(ξ), dan pangkat/pembobot telah ditentukan. Titik awal, banyaknya iterasi, toleransi(ξ), dan pangkat/pembobot di input pada saat *running* program. Dalam proses ini akan dihitung *Centroid* dan *Membership Function* dengan menggunakan algoritma *Mixture*.

Tabel 1. Data Curah Hujan

X1	X2	Y1	Y2
10.0	10.0	10.0	50.0
20.0	40.0	20.0	70.0
30.0	30.0	30.0	60.0
40.0	50.0	40.0	80.0
50.0	50.0	50.0	40.0
60.0	70.0	60.0	40.0
70.0	70.0	70.0	20.0
80.0	80.0	80.0	10.0

Akan dilakukan dengan *Gaussian Mixture Modelling* (GMM).

Langkah 1 :

Tentukan jumlah cluster dimana $Kernel/C = 2$

Langkah 2 :

a. Hitung Means (sama dengan centroid pada FCM) dari masing-masing cluster.

Inisialisasi matriks partisi acak dari data

$$U = \begin{pmatrix} 0.3233 & 0.5268 & 0.6171 & 0.2317 & 0.9771 & 0.6423 & 0.6233 & 0.6232 \\ 0.6767 & 0.4732 & 0.3829 & 0.7683 & 0.0229 & 0.3577 & 0.3767 & 0.3768 \end{pmatrix}$$

Maka didapat means :

$$V = \begin{pmatrix} 0.5980 & 0.6407 & 0.3637 & 0.2759 \\ 0.5615 & 0.6613 & 0.7555 & 0.3431 \end{pmatrix}$$

a. Kriteria Penghentian iterasi Algoritma Mixture

Kriteria yang digunakan untuk menghentikan proses iterasi dari algoritma *Mixture* adalah

$P_{ik} = \pi_i \times f_i(x_k | \theta_i)$, apabila $P_{ik} < \epsilon$ (toleransi) maka iterasi dihentikan.

b. Proses Pengujian Algoritma Mixture

Proses Pengujian yang dilakukan adalah penerapan algoritma *Mixture* untuk pengclusteran dengan data curah hujan, jumlah

b. Hitung Standar Deviasi/variance covariance dari masing-masing cluster

$$S_{XX} = 21.97402830259$$

$$S_{XY} = 83.81626342614$$

$$S_{YX} = 83.81628342614$$

$$S_{YY} = 8.133669424448$$

c. Kembali ke *step* a apabila perubahan nilai probabilitas masih diatas nilai threshold yang ditentukan, atau apabila perubahan pada nilai centroid masih diatas nilai threshold yang

ditentukan, atau apabila perubahan pada nilai *objective function* masih diatas nilai *threshold* yang ditentukan.

Dari hasil percobaan, parameter yang akan dihitung dalam metode itu adalah matriks partisi, pusat cluster (*centroid*), standar deviasi, dan nilai probabilitas.

Dalam percobaan yang dilakukan dalam pengujian sistem ini dengan menggunakan data yang tersimpan dalam Microsoft Excel dengan nama DATA CURAH HUJAN.xls. Dimana data yang digunakan terdapat 12 sample data (dimana untuk data selama satu tahun/12 bulan), dalam dua variabel yaitu X1 : tempat pemeriksaan Diperta Padang Harapan dan X2: Stasiun Klomatologi Pulau Baii. Dimana data yang tabel ini nantinya yang akan diimplementasikan ke dalam Algoritma Mixture.

Tabel 2. Sample Data Curah Hujan selama 12 bulan

DATA KE :	VARIABEL	
	X1	X2
1	225	174
2	163	205
3	407	475
4	310	226
5	129	84
6	34	79
7	61	72
8	189	227
9	129	190
10	164	295
11	580	757
12	717	794

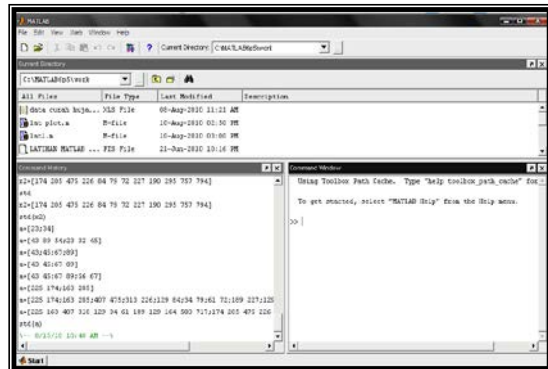
V. HASIL DAN PEMBAHASAN

A. Pengujian Hasil Pengclusteran dalam Algoritma Mixture

Pengujian hasil pengclusteran dalam algoritma Mixture dimana dengan mengikuti algoritma yang sudah dibahas dibab sebelumnya. Dalam Algoritma Mixture ini terdapat nilai standar deviasi dan nilai probabilitas yang masuk dalam proses algoritma ini . Berikut langkah-langkah yang

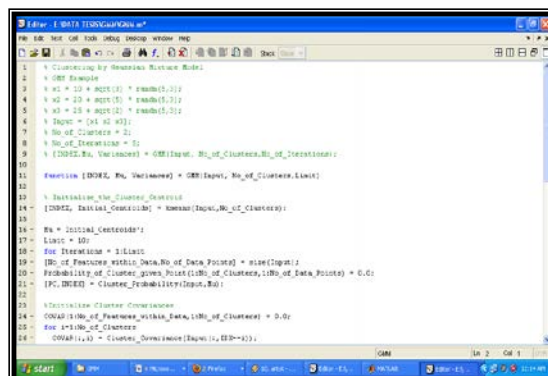
dilakukan dalam pengujian data tersebut dengan menggunakan Matlab 6.5, dalam tahap proses ini sebagai berikut :

- a. Aktifkan Software Matlab, tampilan windows utama Matlab akan muncul seperti gambar 5.10. Untuk menuliskan skrip program yang telah dibuat, harus dipanggil terlebih dahulu Matlab Editor dengan cara; ketik "Edit" pada Prompt Matlab yang ada pada Command Windows. Tampilan Matlab Editor dapat dilihat pada gambar 4.



Gambar 4. Tampilan Windows Utama Matlab

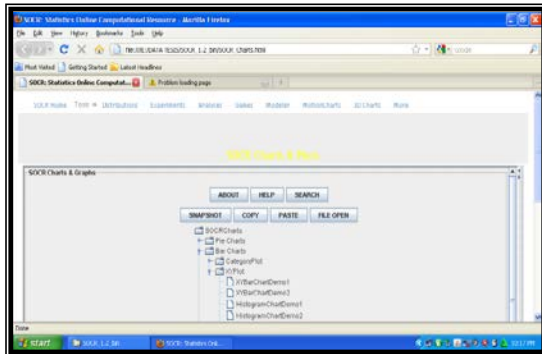
- b. Setelah Matlab Editor muncul, skrip program / kode program siap untuk diketik. Kode program untuk menyelesaikan algoritma mixture, dimana dalam kode program dalam algoritma mixture dibagi menjadi beberapa fungsipada gambar 5.



Gambar 5. Kode Program GMM dalam Matlab

- c. Interface pemanggilan algoritma Mixture. Dalam Algoritma Mixture menggunakan

software SOCR dalam pengclusterannya, dimana Software SOCR merupakan salah satu software yang terdapat fasilitas statistik untuk mixture model chart yang dapat mendukung AlgoritmaMixture. Software SOCR bekerja dilocalhost seperti pada gambar 6 yang menampilkan tampilan utama dari Software SOCR.



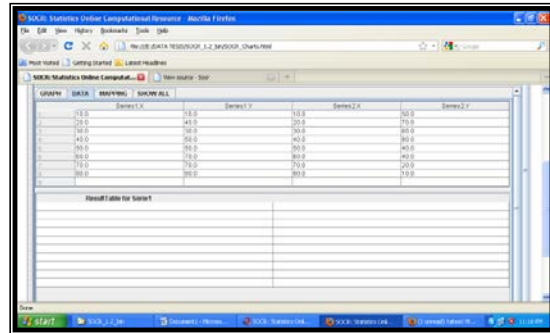
Gambar 6. Tampilan Utama Software SOCR

Pilih Linechart, kemudian pilih SOCR_EM Mixture Model Chart, kemudian lakukan penelusuran kesamping tampilan menu utama. Dimana dalam software SOCR ini terdapat beberapa pengaturan sesuai dengan urutan algoritma sebeum kita menjalankan software ini.



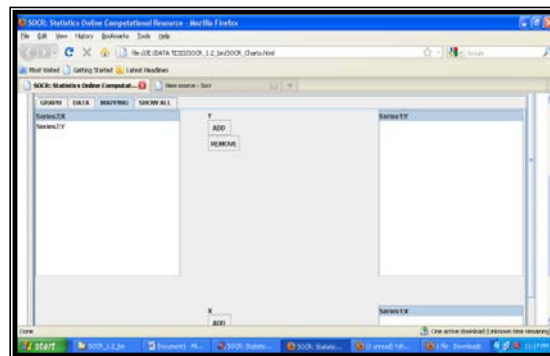
Gambar 7. Menu Input Sebelum Melakukan Clustering Mixture

Dari gambar 7 silahkan kita memberikan input terhadap data yang diberikan nantinya dengan memilih mixture model dengan gaussian, dengan speed normal, dan jumlah cluster atau number of kernel. Setelah itu kita melakukan proses entri data seperti gambar 8



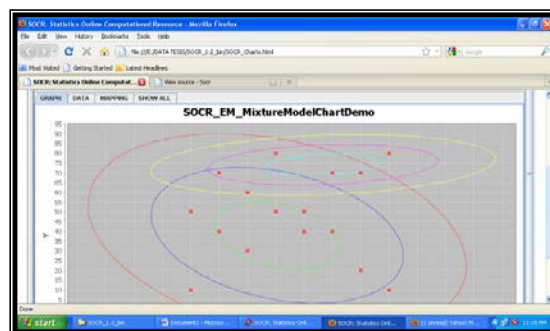
Gambar 8. Tampilan Proses Input Data pada Software SOCR

Dari data yang dientrikan setelah itu melakukan proses mapping dimana kita memilah data yang akan dimasukkan atau di add ke dalam koordinat sumbu x dan y dan juga dapat di remove seperti pada gambar 9.



Gambar 9. Tampilan Proses Mapping dari Input Data Software SOCR

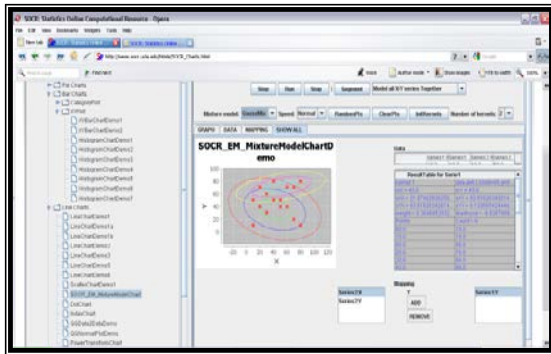
Setelah proses input data dilakukan kita dapat melakukan pilihan **Run** yang ada pada menu pilihan pada gambar 9 maka dapat dilihat Graph nya pada gambar 10.



Gambar 10. Tampilan Graph dari Software SOCR

Chart Graph yang dihasilkan dari proses input data dan proses *mapping* yang sudah dilakukan dan juga dapat dengan fasilitas *Show All* karena pada

tampilan ini akan menampilkan semua chart dengan tabel yang ada.



Gambar 11. Tampilan Show All pada Software SOCR

Dari hasil tampilan Show All terdapat result tabel for series dimana menampilkan informasi nilai means dari sumbu x dan y, nilai standar deviasi, bobot/weight, count/iterasi yang dilakukan berdasarkan dengan proses run yang dijalankan dan epsilon.

B. Hasil Analisa Algoritma Mixture (Gaussian Mixture Modeling/GMM)

Data yang digunakan adalah data curah hujan tahun 2008 dengan dua pos pengamatan yaitu X1=Stasiun Klimatologi Pulau Baii dan X2=Diperta Padang Harapan. Dimana data diambil dari bulan januari sampai dengan desember 2008. Berdasarkan dengan penelusuran algoritma Fuzzy Clustering Means (FCM) dan *Gaussian Mixture Modelling* (GMM) dilakukan proses penginputan dan perhitungan data dengan menggunakan program Matlab 7.1 dengan jumlah cluster, matriks X, matriks partisi, centroid/means, standar deviasi, nilai probabilitas, pembobot/pangkat (w), maksimum iterasi dan kriteria penghentian. Hasil perhitungan dapat disajikan pada tabel berikut:

Tabel 3 Hasil Analisa perhitungan algoritma GMM

No	Unit Input	Gaussian Mixture Modelling (GMM)
1	Jumlah Cluster	Kernel = 3
2	Matriks X (mxn)	12 x 2
3	Matriks Partisi	Hasil dari proses Matriks X

4	Pusat Cluster	Hasil dari Proses Matriks Partisi
5	Standar Deviasi	Ada
6	Nilai Probabilitas	Ada
7	Pembobot	Weight = 2
8	Maksimum Iterasi	Iterasi berjalan apabila di run dan iterasi berhenti apabila distop
9	Kriteria Penghentian	$P_{ik} = \pi_i \times f_i(x_k \theta_i)$, apabila $P_{ik} < \epsilon$ (toleransi)
10	Interface Cluster Demo	Software SOCR

VI. KESIMPULAN DAN SARAN

Berdasarkan analisis, pembahasan dan pengujian sistem mengenai Fuzzy Clustering Algoritma Mixture dalam mengelompokkan data curah hujan Kota Bengkulu yang telah dibangun, maka terdapat beberapa hal yang dapat disimpulkan yaitu :

- Dari hasil pengujian Algoritma Mixtures untuk mendapatkan alternatif yang terbaik, dapat digunakan untuk memecahkan masalah dalam mengelompokkan data yang memiliki kesamaan jumlah data curah hujan yang sama atau mendekati, sehingga dapat digunakan sebagai pendukung pengambilan keputusan dalam mengelompokkan data.
- Mixture Modelling* dalam mengelompokkan data curah hujan Kota Bengkulu mempunyai kemampuan untuk mendeteksi keberadaan suatu cluster yang overlap dengan cluster yang lain.

Berdasarkan dengan kesimpulan dan hasil pengujian sistem dalam penclusteran data curah hujan Kota Bengkulu menggunakan Algoritma Mixtures / Gaussian Mixture Modelling (GMM) dapat membantu Badan Meteorologi, Klimatologi dan Geofisika Stasiun Klimatologi Pulau Baii Bengkulu dalam mengelompokkan atau

menclusterkan data berdasarkan dengan sifat hujan dan distribusi statistik memungkinkan untuk ditambahkan dalam mengakomodasi pendeteksian *outliers* ataupun menangani bentuk-bentuk cluster yang tidak normal.

REFERENSI

- [1] Agusta Yudi, (2007), "*K-Means Penerapan, Permasalahan dan Metode Terkait*", Journal Sistem dan Informatika Vol.3(Februari 2007) Halaman 47-60
- [2] Aqil M, Firmasyah, Prabowo A, Macalinao M, 2007, "*Klustering Tingkat Pemakaian Pompa AirTanah Menggunakan Model Fuzzy*", Jurnal Informatika Pertanian Volume 16 No.1.
- [3] Jaya,2005," *Prediksi Curah Hujan dengan Logika Fuzzy*" Jurnal BMKG Jakarta
- [4] Kusumadewi S, (2002), "*Analisis Desain Sistem Fuzzy Menggunakan Toolbox Matlab*", Yogyakarta Graha Ilmu.
- [5] Kusumadewi S dan Purnomo H, (2004), "*Aplikasi Logika Fuzzy Untuk Pengambilan Keputusan*", Yogyakarta, Graha Ilmu.
- [6] Naba Agus, (2009), "*Belajar Cepat Fuzzy Logic Menggunakan Matlab*", Yogyakarta, Andi.
- [7] Susanto dan Ernawati, (2005), "*Pembagian Kelas Peserta Kuliah Berdasarkan Fuzzy Clustering dan Partition Coefficient and Exponential Separation (PCAES) Index*"
- [8] Lutfi Taufic, (2007), "*Fuzzy C-Means untuk Clustering Data (studi Kasus : Data Performance Mengajar Dosen)*", Seminar Nasional Teknologi Tahun 2007, Yogyakarta.