

BAB II LANDASAN TEORI

2.1 Tinjauan Studi

Tinjauan studi berguna bagi peneliti untuk dijadikan pedoman dan pegangan penelitian yang selanjutnya akan dibuat, dengan adanya penelitian sebelumnya tersebut memudahkan peneliti sesuai topik pembahasan. Penelitian sebelumnya juga dapat dijadikan perbandingan penelitian sehingga menghasilkan penelitian baru yang lebih bermanfaat. Berikut ini adalah beberapa referensi sebagai dasar pelaksanaan penelitian, antara lain:

Menurut [7] yang menerapkan algoritma *K-Means clustering* untuk mengelompokkan jumlah penjualan ikan laut di TPI berdasarkan wilayah di Indonesia, hal tersebut dilakukan karena tidak stabilnya jumlah penjualan dan harga ikan yang ada di setiap daerah. Data dalam penelitian ini diperoleh dari BPS sebanyak 20 provinsi di Indonesia berdasarkan data produksi perikanan laut yang dijual di TPI pada tahun 2015-2018. Selanjutnya data tersebut diolah dengan menggunakan algoritma *K-Means* dan program *Rapid Miner* dengan membuat 2 *cluster* yaitu *cluster* tingkat tinggi dan *cluster* tingkat rendah. Hasil dari penelitian tersebut didapatkan 2 provinsi yang masuk ke dalam *cluster* tinggi dan 18 provinsi pada *cluster* rendah. Dengan hasil penelitian tersebut diharapkan dapat menjadi acuan pemerintah setiap daerah untuk menstabilkan harga ikan.

Menurut [8] yang menjelaskan mengenai klasterisasi hasil tangkap ikan yang berada di Pelabuhan Perikanan Nusantara (PPN) Ternate, hal tersebut dikarenakan belum diketahui jenis ikan hasil tangkap yang menjadi komoditi unggul pada setiap tahunnya. Data diperoleh dari Dinas PPN Ternate dengan data akomodasi hasil tangkap ikan setiap bulannya pada tahun 2015-2017. Dari data yang telah didapatkan, data tersebut kemudian diolah menggunakan algoritma *K-Means* dan dilakukan validasi dengan *Davies Bouldin Index* (DBI) untuk menentukan *cluster* terbaik yang terdapat dalam penelitian ini. Setelah dilakukan perhitungan maka didapatkan hasil, bahwa dengan menggunakan 2 *cluster* yaitu *cluster* hasil tangkap sedikit dan hasil tangkap banyak. Terdapat 16 jenis ikan

yang masuk kedalam *cluster* hasil tangkap sedikit dan 2 jenis ikan pada *cluster* hasil tangkap banyak. Kemudian dari perolehan klasterisasi tersebut dilakukan validasi dengan melakukan perhitungan beberapa *cluster* yaitu 2, 3, 4, dan 5 *cluster* untuk menentukan perbandingan *cluster* terbaik. Dari perbandingan validasi tersebut didapatkan bahwa pada 2 *cluster* dengan hasil DBI 0.54 dinyatakan sebagai *cluster* terbaik dibandingkan *cluster* yang lain, karena apabila nilai DBI mendekati 0 maka *cluster* dianggap baik.

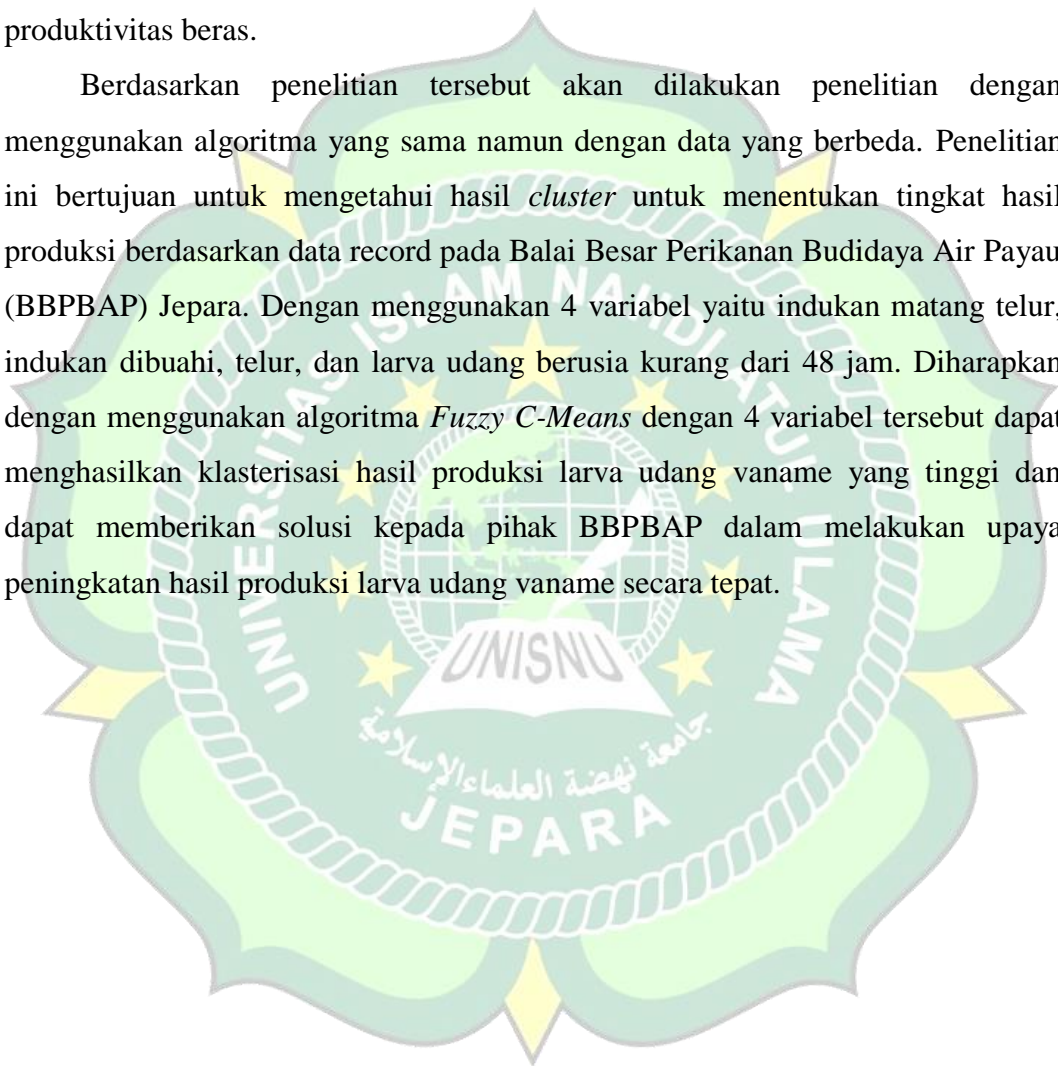
Menurut [9] menerapkan algoritma *Fuzzy C-Means* untuk mengelompokkan kecamatan di Tana Luwu berdasarkan produktivitas hasil perkebunan. Dalam melakukan *clustering*, data diperoleh dari Badan Pusat Statistik- dengan variable hasil produksi tanaman perkebunan yaitu produksi kelapa, kelapa sawit, kopi, lada, kakao, cengkeh dan luas lahan dari 45 kecamatan di Tana Lawu. Kemudian data diolah menggunakan metode *Fuzzy C-Means* dan dievaluasi menggunakan *partition coeeficient index* untuk mengevaluasi derajat keanggotaan dari data tersebut. Hasil penelitian didapatkan 2 *cluster* kategori yaitu kelompok kecamatan produktif dan kelompok kecamatan kurang produktif. Sehingga dari hasil penelitian tersebut dapat digunakan untuk mengontrol distribusi dari hasil tanaman perkebunan.

Menurut [10] menjelaskan pengelompokkan kecamatan di kabupaten Jember berdasarkan hasil produksi pertanian tanaman pangan pada 31 kecamatan di kabupaten Jember pada tahun 2018. Data yang telah diperoleh kemudian dilakukan perhitungan data menggunakan algoritma *Fuzzy C-Means* berdasarkan 4 variabel yang telah ditentukan, kemudian dilakukan proses optimasi dengan menggunakan metode *Elbow*. Dari proses perhitungan yang telah dilakukan dalam penelitian ini, maka dihasilkan 3 *cluster* yang berbeda dengan 12 kecamatan masuk dalam *cluster* 1, 14 kecamatan pada *cluster* 2, dan 5 kecamatan pada *cluster* 3. Dengan adanya hasil klasterisasi tersebut akan digunakan untuk menganalisa tingkat ketahanan pangan dan pemberian bantuan lebih tepat sasaran.

Menurut [11] menjelaskan klasterisasi potensi produksi beras di kabupaten Blitar menggunakan metode *Fuzzy C-Means*. Peneliti menggunakan data produksi beras di kabupaten Blitar tahun 2015 – 2019 yang diperoleh dari BPS. Kemudian

dari data tersebut dilakukan perhitungan menggunakan algoritma *Fuzzy C-Means* dengan membentuk *cluster* data yang berjumlah 3 *cluster* serta dilakukan evaluasi dengan menggunakan *partition coefficient index*. Hasil dari perhitungan yang telah dilakukan maka didapatkan *cluster* dengan karakteristik daerah dengan rata-rata area tanam padi yang luas, cukup luas, dan kurang luas. Dengan hasil klusterisasi dapat digunakan untuk pengambilan keputusan untuk meningkatkan produktivitas beras.

Berdasarkan penelitian tersebut akan dilakukan penelitian dengan menggunakan algoritma yang sama namun dengan data yang berbeda. Penelitian ini bertujuan untuk mengetahui hasil *cluster* untuk menentukan tingkat hasil produksi berdasarkan data record pada Balai Besar Perikanan Budidaya Air Payau (BBPBAP) Jepara. Dengan menggunakan 4 variabel yaitu indukan matang telur, indukan dibuahi, telur, dan larva udang berusia kurang dari 48 jam. Diharapkan dengan menggunakan algoritma *Fuzzy C-Means* dengan 4 variabel tersebut dapat menghasilkan klusterisasi hasil produksi larva udang vaname yang tinggi dan dapat memberikan solusi kepada pihak BBPBAP dalam melakukan upaya peningkatan hasil produksi larva udang vaname secara tepat.



2.2 Tinjauan Pustaka

2.2.1 Data Mining

Data mining merupakan sebuah istilah yang digunakan untuk menguraikan penemuan pengetahuan di dalam database. Data mining adalah proses yang menggunakan teknik *statistic*, matematika, kecerdasan buatan dan *machine learning* untuk mengekstraksi dan mengidentifikasi informasi yang besar [12].

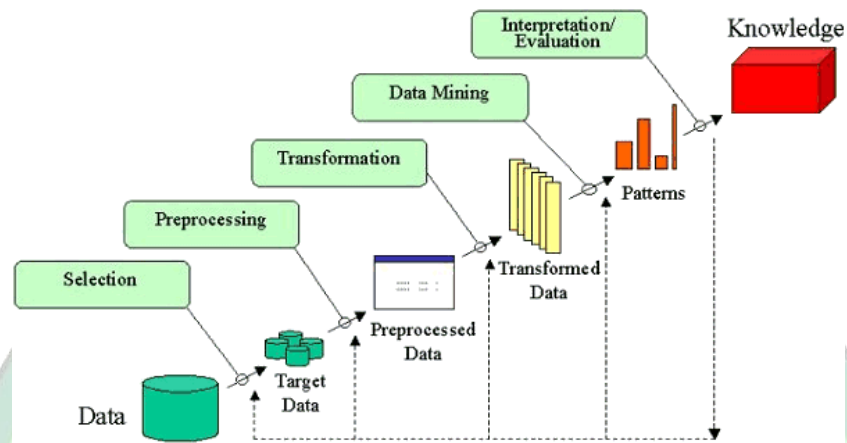
Data Mining merupakan proses penggalian informasi dan berguna dari set data besar yang melibatkan konsep *interdisipliner* yang relatif baru yang melibatkan analisis data dan penemuan pengetahuan dari *database* dan menggunakan pendekatan multi sisi yang mencakup analisis *statistic*, visualisasi data, penemuan pengetahuan, pengenalan pola dan manajemen basis data [13].

Data mining didefinisikan sebagai proses menemukan pola data. Data mining sendiri berhubungan dengan penemuan sesuatu yang tersembunyi dan pola data tertentu yang tidak diketahui sebelumnya dengan menggunakan data yang besar untuk membuat hasil yang lebih akurat. Data mining juga digunakan untuk membuat suatu keputusan yang kritis [14].

Dari uraian diatas dapat disimpulkan bahwa data mining merupakan suatu teknik untuk menggali informasi tersembunyi dari suatu basis data yang besar dengan aturan-aturan tertentu yang berguna untuk membantu dalam pengambilan sebuah keputusan. Selain itu, data mining juga dapat menjelaskan beberapa kondisi penelitian, mempertegas hipotesis, dan menganalisa data yang memiliki pola hubungan yang baru.

Data mining banyak berkaitan dengan bidang ilmu-ilmu seperti *database system*, *data warehousing*, *statistic*, *machine learning*, *information retrieval*, dan komputasi tingkat tinggi. Selain hal tersebut, data mining juga didukung oleh ilmu lain, seperti *neural network*, pengenalan pola, *spatial data analysis*, *image database*, *signal processing* [15]. Data mining juga biasa disebut dengan *knowledge-discovery in database* (KDD) atau *pattern recognition*. Istilah KDD atau disebut penemuan pengetahuan data karena tujuan utama dari data mining adalah untuk memanfaatkan data dalam basis data dengan mengolahnya sehingga menghasilkan sebuah informasi baru yang berguna. Sedangkan istilah *pattern*

recognition atau disebut pengenalan pola yang mempunyai tujuan sebagai pengetahuan yang akan dicari dari dalam sebuah bongkahan data atau dataset yang sedang dihadapi. Proses KDD secara garis besar dapat dijelaskan sebagai berikut [16]:



Gambar 2. 1 Proses *Knowledge Discovery in Databases* [16].

4.1.1 *Data Selection*

Data selection merupakan tahapan pertama proses data mining dimana dilakukan pemilihan data dari kumpulan data berskala besar. Tahapan ini memang harus dilakukan sebelum tahap pencarian informasi dalam KDD dilakukan. Data yang diperoleh dari proses seleksi nantinya akan diproses pada data mining. Proses ini dilakukan dengan memilih data tertentu yang dibutuhkan untuk selanjutnya diproses pada tahap berikutnya dan kemudian data akan disimpan dalam suatu tempat yang berbeda agar mempermudah pengguna dalam menggunakan data.

4.1.2 *Pre-processing (Cleaning)*

Pre-processing tahapan melakukan pembersihan data pada data yang kurang sesuai atau salah pada proses inputan. Dalam tahapan ini juga, apabila terdapat atribut yang digunakan sebagai parameter yang dirasa tidak sesuai mengalami penghapusan atribut. Hal tersebut dilakukan karena, apabila terdapat data ataupun atribut yang kurang relevan tetapi masih dicantumkan maka dapat mempengaruhi tingkat akurasi dari hasil yang diperoleh pada proses data mining. Karena *pre-processing* sangat penting dilakukan sebelum melanjutkan ke proses berikutnya.

4.1.3 Transformation

Pada proses data mining tidak dapat mengambil maupun memasukan data mentah secara sembarangan. Tetapi, diperlukannya format data yang sesuai sebelum data tersebut akan digunakan. Transformasi dalam pemilihan data juga dapat menentukan kualitas dari hasil perhitungan data mining. Hal tersebut dikarenakan, terdapat beberapa karakteristik yang berbeda dari masing-masing data mining yang digunakan.

4.1.4 Data Mining

Data mining merupakan proses pengumpulan data, baik dalam skala kecil maupun besar yang bertujuan untuk menemukan informasi serta pengetahuan di masa mendatang dengan menggunakan aturan tertentu. Dimana proses awal dari data mining dimulai dari database, data relevan, data siap hingga akhirnya data mining. Pada data mining terdapat berbagai macam teknik, metode maupun algoritma yang sangat bervariasi. Pemilihan algoritma yang tepat harus sesuai dengan tujuan dan proses KDD selanjutnya.

4.1.5 Interpretation (Evaluation)

Tahapan ini, ketika informasi sudah didapatkan dari proses data mining hasil tersebut ditampilkan dengan bentuk maupun bahasa yang lebih mudah dipahami oleh semua pihak. Karena pada tahap ini dilakukan penyesuaian informasi yang ditemukan sudah sesuai dengan fakta atau bertentangan dengan hipotesis sebelumnya.

Menurut Kusrini (2009) dalam data mining memiliki beberapa metode dan algoritma yang dapat digunakan untuk menganalisis suatu data, diantaranya sebagai berikut [12]:

1. Klastering (*Clustering*)

Klastering digunakan menentukan atau menganalisis objek data dimana label kelas tidak dikethau dengan cara mengelompokkan data untuk membentuk kelas baru. Contoh algoritma klastering yaitu *K-Means*, *K-Medoids*, *Fuzzy C-Means*, dan lain-lain.

2. Klasifikasi (*Classification*)

Klasifikasi merupakan proses menemukan model yang menjelaskan dan membedakan kelas-kelas dengan tujuan model tersebut dapat digunakan untuk memprediksi kelas atau objek yang memiliki label kelas yang tidak diketahui. Contoh algoritma klasifikasi yaitu C4.5, K-Nearest Neighbor (K-NN), dan lain-lain.

3. Asosiasi (*Association*)

Asosiasi merupakan pencarian aturan aturan asosiasi yang menunjukkan kondisi-kondisi nilai atribut yang sering terjadi secara bersamaan dalam sekumpulan data, yang sering digunakan untuk menganalisis *market basket* dan data transaksi. Contoh algoritma asosiasi yaitu Apriori, *Frequent Pattern Growth* (FP-Growth), dan lain-lain.

4. Estimasi (*Estimation*)

Estimasi pada dasarnya sama dengan metode klasifikasi, dimana memerlukan data testing yang sudah diberi label. Akan tetapi, output dari klasifikasi adalah nilai diskrit, sedangkan output dari estimasi adalah nilai kontinu, titik estimasi tersebut mencari model hubungan antar atribut prediktor dan atribut dependent, dimana atribut dependennya juga berupa nilai kontinu. Contoh algoritma estimasi yaitu *Regresi Linier*, *Confidence Interval Estimations*, dan lain-lain.

2.2.2 Algoritma *Fuzzy C-Means*

Metode *clustering* dengan menggunakan algoritma *Fuzzy C-Means* didasarkan pada teori logika *Fuzzy*. Teori tersebut pertama kali diperkenalkan oleh Lotfi A. Zadeh pada tahun 1965 dengan nama himpunan *Fuzzy* (*Fuzzy set*) [17]. Dalam teori fuzzy, keanggotaan sebuah data tidak diberikan nilai tegas dengan nilai 1 (menjadi anggota) dan 0 (tidak menjadi anggota), melainkan dengan nilai derajat keanggotaan yang jangkauan nilainya 0 sampai 1. Nilai keanggotaan suatu data dalam sebuah himpunan menjadi 0 ketika sama sekali tidak menjadi anggotanya, dan jika semakin kecil maka semakin rendah derajat keanggotaannya.

Kaitannya dengan *K-Means*, sebenarnya *FCM* merupakan versi *fuzzy* dari *K-Means* dengan beberapa modifikasi yang membedakannya dengan *K-Means*.

Fuzzy C-Means merupakan sebuah teknik untuk mengklusterkan suatu data yang berada pada setiap titik data dalam suatu kluster dan ditentukan oleh derajat keanggotaan [18]. Konsep dasar dari algoritma *Fuzzy C-Means* itu sendiri, yang pertama dengan menentukan pusat *cluster* yang berfungsi sebagai penanda lokasi rata-rata untuk setiap kluster. Yang mana pada kondisi awal pusat kluster tersebut belum akurat. Setiap titik data memiliki derajat keanggotaan yang berbeda pada setiap klusternya. Maka dari itu, dilakukan perbaikan pusat kluster dan derajat keanggotaan setiap data secara berulang untuk melihat pusat kluster tersebut bergerak menuju titik yang tepat serta memiliki tingkat akurasi yang tinggi dan waktu komputasi yang cepat [19].

Perulangan tersebut berdasarkan minimisasi fungsi objektif yang menggambarkan jarak dari titik data yang menuju pusat kluster yang memiliki derajat keanggotaan titik data yang berbobot. Algoritma *Fuzzy C-Means* menghasilkan deretan pusat kluster dan beberapa derajat keanggotaan untuk setiap data.

Menurut Kusumadewi (2004), tahapan Algoritma *Fuzzy C-Means* sebagai berikut [20]:

1. Input data yang akan dicluster X , berupa matriks berukuran $n \times m$ (n = jumlah sample data, m = atribut setiap data). X_{ij} = data sample ke- i ($i = 1, 2, \dots, n$), atribut ke- j ($j = 1, 2, \dots, m$).
2. Menetapkan variabel yang diperlukan. Dengan ketentuan sebagai berikut:

Tentukan jumlah <i>cluster</i> (c)	= ($k > 2$)
Pangkat untuk matriks partisi (w)	= ($w > 1$)
Maksimum iterasi	= (MaxIter)
Error terkecil yang diharapkan	= (ϵ)
Fungsi objektif awal	= ($P_0 = 0$)
Iterasi awal	= ($t = 1$)

3. Membangkitkan bilangan random μ_{ik} ($i = 1, 2, \dots, n$; $k = 1, 2, \dots, c$), sebagai elemen-elemen matriks partisi awal μ_{ik} . μ_{ik} adalah derajat keanggotaan yang merujuk pada seberapa besar kemungkinan suatu data bisa menjadi anggota ke dalam suatu *cluster*. Posisi dan nilai matriks dibangun secara random. Dimana nilai keanggotaan terletak pada interval 0 sampai 1. Pada posisi awal matriks partisi U masih belum akurat begitu juga pusat *clusternya*. Sehingga kecenderungan data untuk masuk suatu *cluster* belum akurat.

$$\mu_{ik} = \begin{bmatrix} \mu_{11}[u_1] & \mu_{21}[u_1] & \dots & \dots & \mu_{c1}[u_1] \\ \mu_{12}[u_2] & \mu_{22}[u_2] & \dots & \dots & \mu_{c2}[u_2] \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \mu_{1N}[u_N] & \mu_{2N}[u_N] & \dots & \dots & \mu_{cN}[u_N] \end{bmatrix}$$

Dan,

$$\sum_{i=1}^k \mu_{ik} = \mathbf{1} \dots \dots \dots (2.1)$$

Dimana, $0 < \sum_{j=1}^k \mu_{ik} < 1$, yang berarti nilai acak adalah lebih besar dari 0 dan lebih kecil dari 1.

4. Menghitung pusat setiap *cluster* ke- k : V_{kj} dengan $k = 1, 2, \dots, c$ dan $j = 1, 2, \dots, m$ untuk matriks partisi. Dimana X_{ij} adalah variabel fuzzy yang digunakan dan w adalah bobot.

$$V_{kj} = \frac{\sum_{i=1}^n (\mu_{ik})^w * X_{ij}}{\sum_{i=1}^n (\mu_{ik})^w} \dots \dots \dots (2.2)$$

Dimana:

V_{kj} = nilai pusat *cluster* data ke- i pada variabel ke- j

μ_{ik} = nilai derajat keanggotaan *cluster* data ke- i pada kelompok ke- k

X_{ij} = data ke- i pada variabel ke- j

w = pangkat pembobot

n = jumlah data

5. Menghitung fungsi objektif pada iterasi ke- t , P_t . Fungsi objektif tersebut berfungsi sebagai syarat perulangan untuk mendapatkan pusat kluster yang tepat. Sehingga nantinya akan diperoleh kecenderungan data yang masuk pada kluster yang sesuai pada tahap terakhir.

$$P_t = \sum_{i=1}^n \sum_{k=1}^c \left(\left[\sum_{j=1}^m (X_{ij} - V_{kj})^2 (\mu_{ik})^w \right] \right) \dots\dots\dots (2.3)$$

Menghitung fungsi objektif P_t dengan mengurangi nilai variabel fuzzy X_{ij} dengan pusat *cluster* V_{kj} kemudian hasil pengurangannya di kuadratkan. Hasil dari setiap kuadrat dijumlahkan untuk dikali dengan kuadrat dari derajat keanggotaan μ_{ik} pada setiap *cluster*. Setelah itu, menjumlahkan semua nilai pada semua *cluster* untuk mendapatkan fungsi objektif P_t .

6. Menghitung perubahan matriks partisi U .

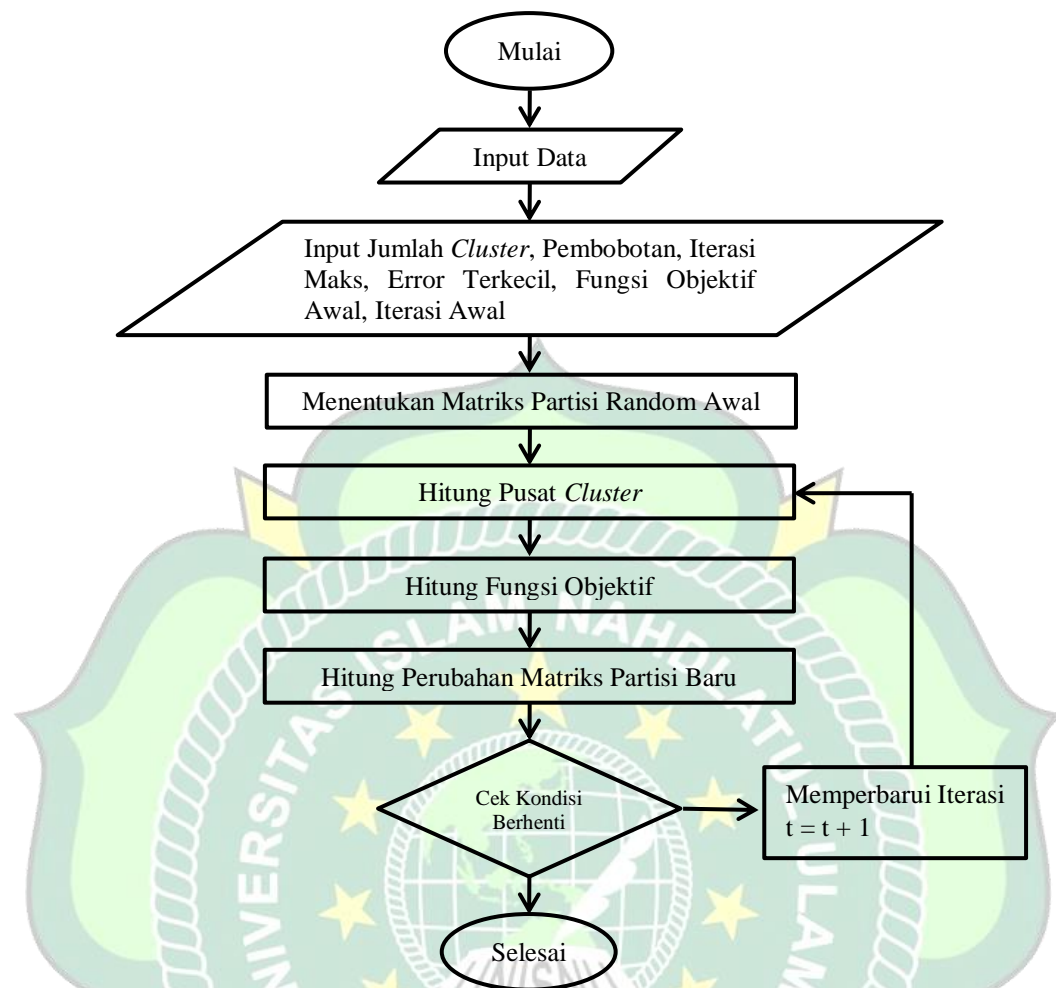
$$\mu_{ik} = \frac{\left[\sum_{j=1}^m ((X_{ij} - V_{kj})^2)^{\frac{-1}{w-1}} \right]}{\sum_{k=1}^c \left[\sum_{j=1}^m ((X_{ij} - V_{kj})^2)^{\frac{-1}{w-1}} \right]} \dots\dots\dots (2.4)$$

Dengan, $i = 1, 2, \dots, n$ dan $k = 1, 2, \dots, c$.

Dalam mencari perubahan dari matriks partisi μ_{ik} , melakukan pengurangan nilai variabel fuzzy X_{ij} terhadap pusat *cluster* V_{kj} kemudian dikuadratkan. Selanjutnya dijumlahkan dan dipangkatkan dengan $\frac{-1}{w-1}$. Setelah melakukan proses perhitungan, normalisasikan semua data derajat keanggotaan baru dengan cara menjumlahkan derajat keanggotaan baru $k = 1, \dots, c$, hasilnya dari perhitungan tersebut kemudian dibagi dengan derajat keanggotaan yang baru. Proses ini dilakukan agar derajat keanggotaan yang baru mempunyai rentang antara 0 dan tidak lebih dari 1.

7. Mengecek kondisi untuk berhenti, dengan ketentuan:

- Jika ($|P_t - P_{t-1}| < \zeta$) atau ($t > \maxIter$) maka berhenti.
- Jika tidak, $t = t + 1$, ulangi langkah ke - 4.



Gambar 2. 2 Flowchart *Fuzzy C-Means* [20].

Seperti algoritma pada umumnya, algoritma *Fuzzy C-Means* memiliki kelebihan dan kelemahan yang dijabarkan sebagai berikut [21]:

1. Kelebihan

- a. Dalam implementasi penyelesaian masalah dengan menggunakan algoritma *Fuzzy C-Means* dapat memahami karakteristik data yang tidak dapat didefinisikan.
- b. Mempunyai kemampuan dalam mengelompokkan data yang besar.
- c. Lebih kuat terhadap data *outlier* atau data dengan karakter yang berbeda atau *value* yang berbeda dalam satu atau beberapa variable.
- d. Penentuan titik *cluster* yang optimal.
- e. Dapat melakukan *clustering* lebih dari satu variable secara bersamaan.

2. Kelemahan

- a. Algoritma *Fuzzy C-Means* memerlukan waktu yang panjang untuk melakukan proses perhitungan komputasi dalam menentukan *cluster* pada setiap anggota dalam sebuah dataset.
- b. Masih terpengaruh terhadap cara pembagian data pada data yang sama dan sangat sensitive terhadap kondisi awal seperti jumlah *cluster* dan titik pusat *cluster* pada pengelompokan data.

Untuk memahami lebih lanjut tentang algoritma *Fuzzy C-Means*, berikut contoh penerapan perhitungan *FCM* yang digunakan sebagai data sampel perhitungan, yaitu :

1. Input data hasil produksi larva udang vaname, data yang masukkan adalah 10 data sampel dengan 4 parameter.

Tabel 2. 1 Data Hasil Produksi Larva Udang Vaname

No	Tanggal Panen	Induk Udang Matang Telur	Induk Udang Dibuahi	Telur	Larva Udang Usia < 48 jam
1	01-07-2020	193 ekor	108 ekor	9 juta	7 juta
2	02-07-2020	161 ekor	97 ekor	8 juta	7 juta
3	03-07-2020	188 ekor	119 ekor	9.8 juta	8 juta
4	04-07-2020	142 ekor	67 ekor	7 juta	6 juta
5	05-07-2020	229 ekor	122 ekor	9.8 juta	8 juta
6	06-07-2020	129 ekor	90 ekor	7.6 juta	5.6 juta
7	07-07-2020	234 ekor	125	7.8 juta	6.8 juta
8	08-07-2020	145 ekor	91	8 juta	7 juta
9	09-07-2020	172 ekor	87	8 juta	7 juta
10	10-07-2020	182 ekor	106	8 juta	7 juta

2. Menetapkan variable yang diperlukan berdasarkan tabel 2.1, dengan ketentuan sebagai berikut :

Tabel 2. 2 Komponen Perhitungan

No	Komponen Perhitungan	Keterangan
1	Banyaknya <i>cluster</i> yang diinginkan	$c = 2$
2	Pangkat (pembobotan)	$w = 2$
3	Maksimum Iterasi	$T = 1$
4	Error Terkecil	$e = 0,1$
5	Fungsi Objektif Awal	$P_0 = 0$
6	Iterasi Awal	iter = 1

3. Membangkitkan bilangan random μ_{ik} , dengan i = banyaknya data dan k = banyak *cluster* (memiliki nilai acak 0-1).

$$\mu_{ik} = \begin{bmatrix} 0.55 & 0.45 \\ 0.62 & 0.38 \\ 0.31 & 0.69 \\ 0.51 & 0.49 \\ 0.65 & 0.35 \\ 0.44 & 0.56 \\ 0.14 & 0.86 \\ 0.45 & 0.55 \\ 0.44 & 0.56 \\ 0.73 & 0.27 \end{bmatrix}$$

4. Melakukan perhitungan manual pusat *cluster* ke-*k*Tabel 2. 3 Perhitungan Pusat *Cluster* ke-1

No	μ_{ik1}	Data yang akan di <i>cluster</i>				μ_{ik1}^2	μ_{ik}^2 * X_1	μ_{ik}^2 * X_2	μ_{ik}^2 * X_3	μ_{ik}^2 * X_4
		X_1	X_2	X_3	X_4					
1	0.55	193	108	9	7	0.303	58.38	32.67	2.723	2.118
2	0.62	161	97	8	7	0.384	61.89	37.29	3.075	2.691
3	0.31	188	119	9.8	8	0.096	18.07	11.44	0.942	0.769
4	0.51	142	67	7	6	0.26	36.93	17.43	1.821	1.561
5	0.65	229	122	9.8	8	0.423	96.75	51.55	4.141	3.38
6	0.44	129	90	7.6	5.6	0.194	24.97	17.42	1.471	1.084
7	0.14	234	125	7.8	6.8	0.02	4.586	2.45	0.153	0.133
8	0.45	145	91	8	7	0.203	29.36	18.43	1.62	1.418
9	0.44	172	87	8	7	0.194	33.3	16.84	1.549	1.355
10	0.73	182	106	8	7	0.533	96.99	56.49	4.263	3.73
Jumlah						2.608	461.2	262	21.76	18.24
							176.9	100.5	8.343	6.994

Tabel 2. 4 Pehitungan Pusat *Cluster* ke-2

No	μ_{ik2}	Data yang akan di <i>cluster</i>				μ_{ik2}^2	μ_{ik}^2 * X_1	μ_{ik}^2 * X_2	μ_{ik}^2 * X_3	μ_{ik}^2 * X_4
		X_1	X_2	X_3	X_4					
1	0.45	193	108	9	7	0.203	39.08	21.87	1.823	1.418
2	0.38	161	97	8	7	0.144	23.25	14.01	1.155	1.011
3	0.69	188	119	9.8	8	0.476	89.51	56.66	4.666	3.809
4	0.49	142	67	7	6	0.24	34.09	16.09	1.681	1.441
5	0.35	229	122	9.8	8	0.123	28.05	14.95	1.201	0.98
6	0.56	129	90	7.6	5.6	0.314	40.45	28.22	2.383	1.756
7	0.86	234	125	7.8	6.8	0.74	173.1	92.45	5.769	5.029
8	0.55	145	91	8	7	0.303	43.86	27.53	2.42	2.118
9	0.56	172	87	8	7	0.314	53.94	27.28	2.509	2.195
10	0.27	182	106	8	7	0.073	13.27	7.727	0.583	0.51
Jumlah						2.928	538.6	306.8	24.19	20.27
							184	104.8	8.262	6.922

Dari perhitungan pusat *cluster* pada tabel 2.3 dan 2.4, maka didapatkan pusat *cluster* baru, yang ditunjukkan sebagai berikut :

Tabel 2. 5 Pusat *Cluster* Baru

V_{kj}	1	2	3	4
1	176.8674	100.4665	8.343017	6.993688
2	183.952	104.7806	8.261808	6.921969

5. Setelah melakukan perhitungan pusat *cluster*. Langkah selanjutnya ialah melakukan perhitungan fungsi objektif dengan mengurangi nilai variabel fuzzy X_{ij} dengan pusat *cluster* V_{kj} kemudian hasil pengurangannya di kuadratkan. Hasil dari setiap kuadrat dijumlahkan untuk dikali dengan kuadrat dari derajat keanggotaan μ_{ik} pada setiap *cluster*. Setelah itu, menjumlahkan semua nilai pada semua *cluster* untuk mendapatkan fungsi objektif P_t , yang dijabarkan sebagai berikut :

Tabel 2. 6 Perhitungan Fungsi Objektif

No	Kuadrat keanggotaan data ke-i		L_1	L_2	$L_1 + L_2$
	μ_{ik1}^2	μ_{ik2}^2			
1	0.3025	0.2025	96.0278	18.7883	114.8162
2	0.3844	0.1444	101.4459	84.8214	186.2674
3	0.0961	0.4761	45.2210	105.7451	150.9662
4	0.2601	0.2401	608.2515	765.8670	1374.1185
5	0.4225	0.1225	1345.5103	285.3464	1630.8568
6	0.1936	0.3136	465.2838	1016.1812	1481.4651
7	0.0196	0.7396	75.7807	2155.0874	2230.8681
8	0.2025	0.3025	223.8152	516.4393	740.2546
9	0.1936	0.3136	39.7180	143.9653	183.6833
10	0.5329	0.0729	30.4187	0.39162	30.8103
Fungsi Objektif					8124.1069

Dari proses perhitungan fungsi objektif diatas, maka didapatkan nilai fungsi objektif sebesar 8124.10698.

6. Menghitung perubahan matriks partisi baru, dengan proses perhitungan sebagai berikut :

Tabel 2. 7 Pehitungan Matriks Partisi Baru

No	L_1	L_2	L_t	μ_{ik1}	μ_{ik2}
				$\frac{L_1}{L_t}$	$\frac{L_2}{L_t}$
1	0.0031	0.01077796	0.01392808	0.22617082	0.77382918
2	0.0037	0.0017024	0.00549161	0.68999995	0.31000005
3	0.0021	0.00450233	0.00662745	0.32065383	0.67934617
4	0.0004	0.0003135	0.00074112	0.5769904	0.4230096
5	0.0003	0.0004293	0.00074331	0.42244457	0.57755543
6	0.0004	0.00030861	0.0007247	0.57415774	0.42584226
7	0.00025864	0.00034319	0.00060183	0.42975825	0.57024175
8	0.00090476	0.00058574	0.00149051	0.60701817	0.39298183
9	0.00487436	0.0021783	0.00705266	0.69113768	0.30886232
10	0.01751882	0.18614918	0.203668	0.08601653	0.91398347

Dengan melakukan perhitungan matriks partisi baru, maka didapatkan matriks partisi baru sebagai berikut:

Tabel 2. 8 Matriks Partisi Baru

Matriks Partisi Baru	
0.226170817	0.77382918
0.689999948	0.31000005
0.320653826	0.67934617
0.576990401	0.4230096
0.422444568	0.57755543
0.574157741	0.42584226
0.42975825	0.57024175
0.607018174	0.39298183
0.69113768	0.30886232
0.086016534	0.91398347

7. Mengecek kondisi berhenti, dengan melakukan pengurangan antar fungsi objektif. Dengan proses perhitungan sebagai berikut : $P_0= 0$ dan $P_1= 8124.10698$, sehingga didapatkan hasil pengurangan adalah 8124.10698. Dalam proses awal penentuan komponen dituliskan bahwa error terkecil adalah 0.1, dan dalam pengurangan antara fungsi objektif awal dan fungsi objektif pada iterasi 1 belum mencapai ketentuan. Maka dilakukan kembali proses perhitungan pada langkah ke 4 sampai nantinya memperoleh nilai eror terkecil sesuai dengan ketentuan komponen.

Setelah dilakukan proses perhitungan *fuzzy c-means*, didapatkan hasil akhir perhitungan sebagai berikut :

Tabel 2. 9 Hasil Akhir Perhitungan *Fuzzy C-Means*

C1	C2
0.9286	0.8699
0.9350	0.8458
0.9395	0.9453
0.9915	0.9240
0.8273	0.6144

2.2.3 Davies Bouldin Index (DBI)

Matrik *Davies Bouldin Index* (DBI) diperkenalkan oleh David L. Davies dan Donald W. Bouldin pada tahun 1979 yang digunakan untuk mengevaluasi *cluster* yang ditinjau dari kohesi dan separasi. Kohesi merupakan nilai jarak antar data yang berada dalam *cluster* yang sama. Sedangkan separasi adalah nilai jarak untuk mengukur jarak antar data dalam *cluster* yang berbeda. Validitas yang dilakukan adalah seberapa baik *clustering* yang sudah dilakukan dengan cara menghitung kuantitas dan fitur turunan dari set data[22].

Langkah-langkah untuk menghitung nilai *Davies Bouldin Index* adalah sebagai berikut [23]:

1. *Sum of Square within cluster* (SSW) bertujuan untuk mengetahui matriks kohesi dalam sebuah *cluster* ke-*i*, yang dirumuskan pada persamaan :

$$SSW = \frac{1}{m_i} \sum_{j=1}^{m_i} d(x_j, c_i) \dots \dots \dots (2.5)$$

m_i adalah jumlah data yang berada dalam *cluster* ke-*i*, sedangkan c_i adalah centroid ke-*i*.

2. *Sum Of Square Between Cluster* (SSB) untuk mengetahui matrik separasi atau jarak antar dua *cluster*, misalnya *i* dan *j*, digunakan formula *Sum Of Square Between Cluster* (SSB) dengan mengukur jarak antar centroid c_i dan c_j seperti pada persamaan berikut :

$$SBB_{i,j} = d(c_i, c_j) \dots \dots \dots (2.6)$$

3. Perhitungan rasio R_{ij} yang digunakan untuk mengetahui nilai perbandingan antar *cluster* ke-*i* dan *cluster* ke-*j* dengan menghitung nilai rasio yang dimiliki dari masing-masing *cluster*. Nilainya didapatkan dari komponen kohesi dan separasi. *Cluster* yang baik adalah yang mempunyai kohesi yang sekecil mungkin dan separasi yang sebesar mungkin. R_{ij} diformulasikan oleh persamaan berikut :

$$R_{i,j} = \frac{SSW_i SSW_j}{SSB_{i,j}} \dots \dots \dots (2.7)$$

Sifat-sifat yang dimiliki R_{ij} sebagai berikut :

- $R_{ij} \geq 0$
- $R_{i,j} = R_{j,i}$
- Jika $SSW_j \geq SSW_r$ dan $SSB_{i,j} = SSB_{i,r}$ maka $R_{i,j} > R_{i,r}$
- Jika $SSW_j = SSW_r$ dan $SSB_{i,j} \leq SSB_{i,r}$ maka $R_{i,j} > R_{i,r}$

4. Nilai *Davies Bouldin Index* (DBI) didapatkan dari persamaan berikut :

$$DBI = \frac{1}{K} \sum_{i=1}^K \frac{\max_{i \neq j} (R_{i,j})}{\dots} \dots \dots \dots (2.8)$$

K adalah jumlah *cluster* yang digunakan.

Dari syarat-syarat perhitungan yang didefinisikan di atas, dapat diamati bahwa semakin kecil nilai SSW maka hasil *clustering* yang didapat juga akan baik. Secara esensial, DBI menginginkan nilai dengan sekecil (non-negatif ≥ 0) mungkin untuk menentukan baik atau tidaknya *cluster* yang didapat. Indeks tersebut diperoleh dari rata-rata semua indeks *cluster*, dan nilai yang didapat dapat digunakan sebagai pendukung keputusan untuk penentuan jumlah *cluster* yang cocok digunakan.

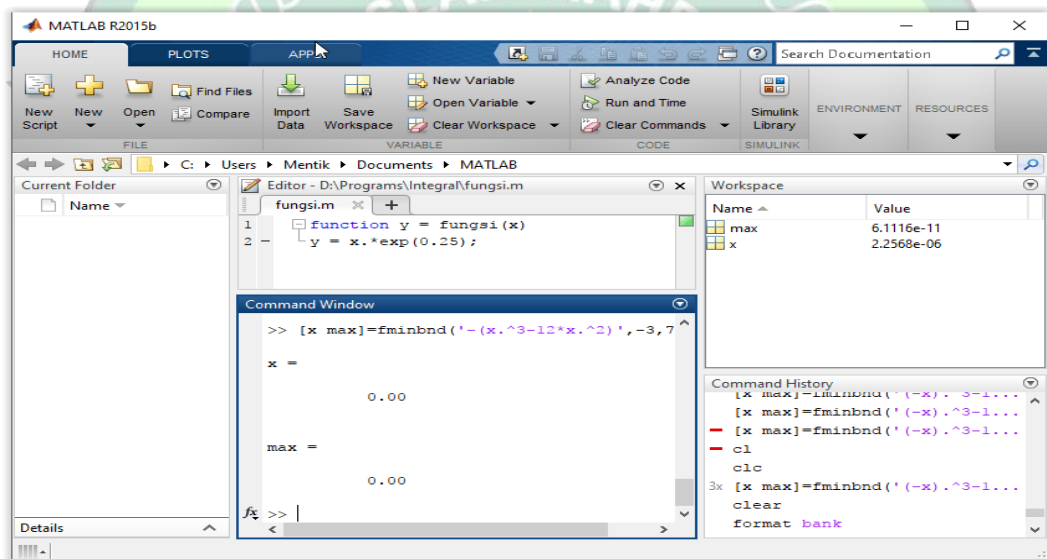
2.2.4 MATLAB (*Matrix Laboratory*)

MATLAB atau *Matrix Laboratory* adalah sebuah software yang digunakan untuk pemrograman, analisis, dan komputasi teknis serta matematika yang berbasis matriks. MATLAB pertama kali diperkenalkan pada tahun 1970 oleh Cleve Moler. Pada awalnya, software ini dirancang untuk menyelesaikan masalah-masalah pada persamaan aljabar linear. Akan tetapi dalam perkembangannya, MATLAB sendiri mengalami banyak perubahan baik dalam segi fungsi dan performa komputasi [19].

Saat ini MATLAB dikembangkan oleh Math Works Inc. mengintegrasikan komputasi matematik, visualisasi dan bahasa pemrograman untuk memberikan lingkungan kerja yang fleksibel bagi komputasi teknik. MATLAB juga memberikan kemudahan untuk pengguna dalam mengeksplorasi data, menciptakan algoritma, dan menciptakan beberapa perangkat grafis (GUI) [24].

Dikarenakan dapat memberikan perhitungan vector dan matriks dengan kecepatan tinggi. Maka, MATLAB banyak digunakan untuk komputasi numeric dan pengembangan algoritma, komputasi simbolik dengan menggunakan fungsi-fungsi pustaka *Symbolic Math*. Selain itu juga digunakan sebagai pemodelan, simulasi, penciptaan prototype, analisis data, pemrosesan sinyal atau citra, visualisasi saintifik dan grafik rekayasa [25].

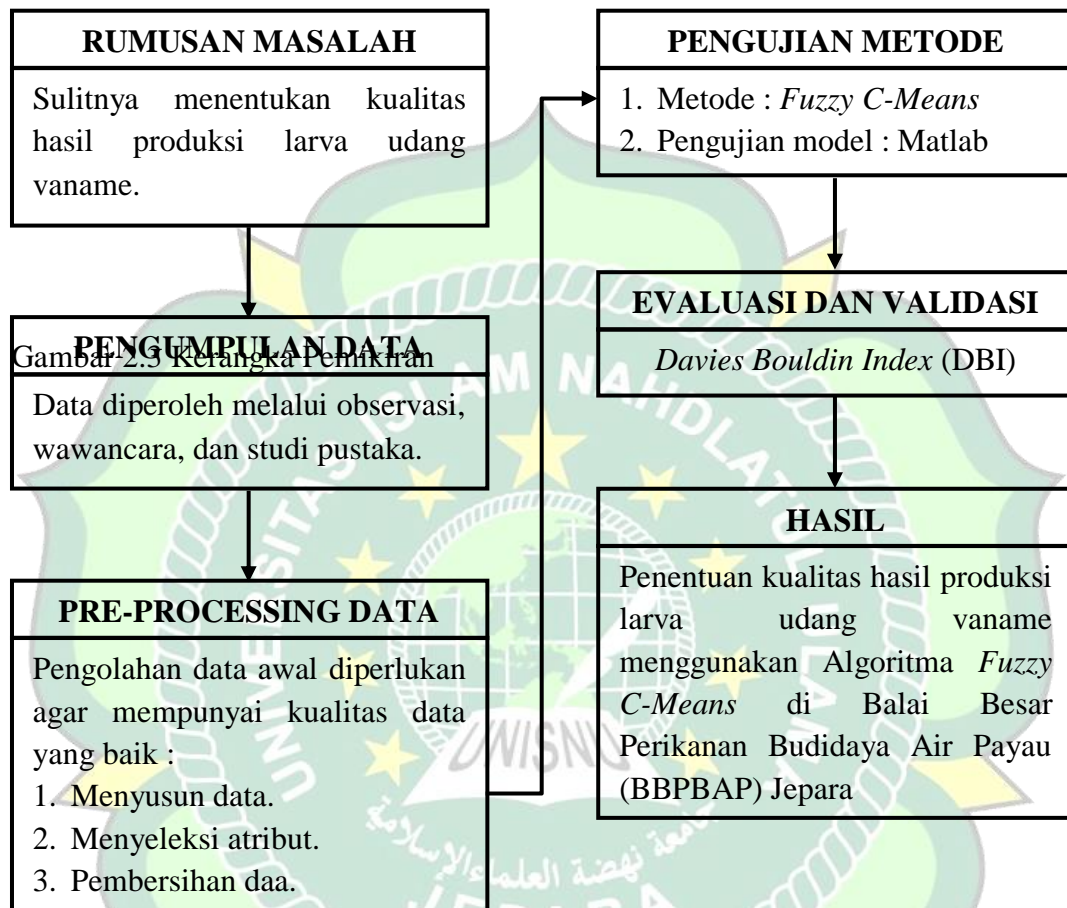
MATLAB dapat dioperasikan dalam sistem operasi Linux, Windows dan macOS serta dapat dihubungkan pada bahasa pemrograman seperti C, Java, dan Microsoft Excel. Dalam MATLAB terdapat *toolbox* yang dapat digunakan dalam pengolahan sinyal, sistem control, logika *fuzzy*, jaringan saraf tiruan, optimasi, dan berbagai teknologi lainnya.



Gambar 2. 3 Tampilan MATLAB

2.3 Kerangka Pemikiran

Pada tahap ini peneliti membuat suatu kerangka pemikiran secara bertahap. Kerangka pemikiran ini merupakan pola pikir peneliti dari awal sampai dalam melakukan penelitian.



Gambar 2. 4 Kerangka Pemikiran