

# **PENERAPAN FUZZY C-MEANS PADA DATA KARTU KREDIT NASABAH**

**Rizky Akhmad Fahreza<sup>1</sup>**

<sup>1</sup>Program Studi Informatika, Universitas Diponegoro;  
[rafahreza@students.undip.ac.id](mailto:rafahreza@students.undip.ac.id)



## **Abstrak**

Penelitian ini membahas penerapan metode Fuzzy C-Means (FCM) dalam pengelompokan data, dengan fokus pada efektivitasnya melalui metrik evaluasi clustering. FCM memungkinkan setiap titik data memiliki keanggotaan ganda dalam beberapa cluster, memberikan fleksibilitas dalam pengelompokan data yang kompleks. Dengan menggunakan data kartu kredit nasabah, penelitian ini menerapkan algoritma FCM untuk mengidentifikasi struktur cluster. Hasil analisis menunjukkan bahwa jumlah cluster optimal, berdasarkan metrik Final Partition Coefficient, Partition Entropy, dan Xie-Beni Index, adalah 2 cluster. Metrik-metrik ini menunjukkan bahwa dua cluster mampu merepresentasikan data dengan baik, menyeimbangkan kejelasan pemisahan cluster dan keanggotaan fuzzy. Penelitian ini diharapkan dapat memberikan wawasan lebih dalam mengenai penerapan FCM dan kontribusinya dalam analisis data serta rekomendasi untuk penelitian lebih lanjut.

**Kata Kunci:** Fuzzy C-Means, Clustering, Final Partition Coefficient, Partition Entropy, Xie-Beni Index, keanggotaan fuzzy.

# **1. Pendahuluan**

## **1.1. Latar belakang**

Penggunaan metode pengelompokan dalam analisis data semakin penting, terutama dalam konteks pengolahan data yang kompleks. Fuzzy C-Means (FCM) adalah salah satu metode yang memungkinkan data untuk memiliki keanggotaan ganda dalam beberapa cluster, yang berbeda dari metode K-Means yang lebih tradisional. Penelitian ini bertujuan untuk mengeksplorasi penerapan FCM dalam konteks tertentu, serta manfaat dan tantangan yang dihadapi dalam penggunaannya.

## **1.2. Permasalahan**

Rumusan masalah dalam penelitian ini mencakup tiga aspek utama. Pertama, bagaimana cara kerja algoritma Fuzzy C-Means (FCM) dalam mengelompokkan data dengan fleksibilitas keanggotaan fuzzy. Kedua, apa saja aplikasi FCM yang dapat diterapkan dalam berbagai bidang untuk menangani data yang kompleks. Ketiga, bagaimana hasil penerapan FCM dibandingkan dengan metode pengelompokan lainnya dalam hal efektivitas dan kualitas pengelompokan data. Penelitian ini bertujuan untuk menjawab ketiga pertanyaan tersebut melalui studi teoritis dan implementasi praktis algoritma FCM.

## **1.3. Tujuan Penelitian**

Tujuan dari penelitian ini adalah untuk memahami konsep dasar dan cara kerja algoritma Fuzzy C-Means (FCM), serta menerapkannya pada studi kasus untuk menganalisis hasil pengelompokan yang dihasilkan. Selain itu, penelitian ini juga bertujuan memberikan rekomendasi bagi penelitian selanjutnya berdasarkan temuan yang diperoleh, sehingga dapat memperluas wawasan tentang penggunaan FCM dalam analisis data.

## **1.4. Manfaat Penelitian**

Penelitian ini diharapkan dapat memberikan kontribusi terhadap pemahaman yang lebih baik tentang teknik pengelompokan fuzzy, khususnya algoritma Fuzzy C-Means (FCM). Selain itu, penelitian ini juga memberikan wawasan mengenai aplikasi praktis FCM di bidang yang relevan, serta rekomendasi untuk pengembangan metode pengelompokan di masa depan guna meningkatkan efektivitas dan fleksibilitas analisis data.

# **2. Tinjauan Pustaka**

## **2.1. Konsep Dasar FCM (Fuzzy C-Means)**

Fuzzy C-Means (FCM), yang juga dikenal sebagai metode soft clustering, memungkinkan setiap titik data memiliki keanggotaan dalam semua cluster dengan nilai antara 0 dan 1. Semakin dekat sebuah titik data dengan pusat cluster, semakin

tinggi nilai keanggotaannya terhadap cluster tersebut (Kapoor & Singhal, 2017). Pada setiap iterasi, pusat cluster dan nilai keanggotaan diperbarui berdasarkan hasil perhitungan. Selain itu, FCM dapat dikombinasikan dengan algoritma genetika untuk mengembangkan sistem rekomendasi yang didasarkan pada kesamaan atribut (Kapoor & Singhal, 2017).

## 2.2. Algoritma FCM

Algoritma FCM bekerja dengan memperbarui pusat cluster dan matriks keanggotaan secara iteratif hingga hasilnya konvergen. Matriks keanggotaan memungkinkan setiap data memiliki derajat keanggotaan ke dalam beberapa cluster, menjadikannya metode yang fleksibel untuk mengelompokkan data kompleks.

Dikutip dari Ross (2010), terdapat 4 langkah utama bagaimana algoritma FCM bekerja:

1. Inisialisasi
  - a. Menetapkan jumlah cluster ( $c$ ) dengan nilai  $2 \leq c < n$ , di mana  $n$  adalah jumlah data
  - b. Tentukan parameter *fuzziness* ( $m$ ), biasanya  $m > 1$  (semakin besar nilai  $m$ , semakin *fuzzy* pembagian cluster)
  - c. Menetapkan ambang batas toleransi konvergensi ( $\epsilon$ ).
  - d. Inisialisasi matriks partisi ( $U^{(0)}$ ), di mana  $U$  adalah matriks keanggotaan (*membership*) yang menentukan tingkat keanggotaan setiap data dalam setiap cluster. Pada langkah ini, kita mengatur nilai keanggotaan  $u_{ik}^{(0)}$  untuk setiap data  $k$  terhadap setiap cluster  $i$ , yang nilainya berada antara 0 dan 1. Untuk setiap data  $k$ , jumlah total keanggotaan terhadap semua cluster harus 1. Proses ini dilakukan secara acak, dengan tujuan membagi keanggotaan data terhadap setiap cluster sedemikian rupa sehingga jumlahnya 1. Matriks ini kemudian diperbarui pada setiap iterasi algoritma hingga konvergen. Langkah ini diberi indeks iterasi  $r=0,1,2,\dots$

2. Menghitung *cluster centroid*

Menghitung *centroid cluster* ( $v_i^{(r)}$ ) pada setiap iterasi menggunakan matriks keanggotaan ( $U$ ) yang diperbarui pada langkah sebelumnya. Rumus umum untuk pusat cluster adalah:

$$v_i^{(r)} = \frac{\sum_{k=1}^n u_{ik}^{(m)} \cdot x_k}{\sum_{k=1}^n \mu_{ik}^{(m)}} \quad (1)$$

Di mana:

- $\mu_{ik}$  adalah keanggotaan data ke- $k$  dalam cluster  $i$
- $x_k$  adalah titik data ke- $k$

### 3. Memperbarui Matriks Partisi

Untuk setiap iterasi iterasi ( $r$ ), matriks kanggotaan ( $U$ ) diperbarui menggunakan rumus:

$$\mu_{ik}^{(r+1)} = \left[ \sum_{j=1}^c \left( \frac{d_{ik}}{d_{jk}} \right)^{\frac{2}{(m-1)}} \right]^{-1} \quad (2)$$

Di mana:

- $d_{ik}$  adalah jarak antara titik data  $x_k$  dan pusat cluster  $v_i^{(r)}$
- $m$  adalah parameter *fuzziness* yang mengontrol derajat keanggotaan fuzzy

### 4. Validasi konvergensi

- Hitung perbedaan antara matriks partisi saat ini ( $U^{(r+1)}$ ) dengan matriks partisi pada iterasi sebelumnya ( $U^{(r)}$ )
- Jika selisih lebih kecil dari atau sama dengan  $\varepsilon$ , maka algoritma berhenti karena hasil telah konvergen.
- Jika tidak, tingkatkan iterasi ( $r = r + 1$ ) dan kembali ulang langkah 2 hingga 4.

## 2.3. Literatur Terkait

Kapoor & Singhal (2017) melakukan analisa komparatif yang membandingkan performa model algoritma *clustering* K-Means, K-Means++, dan Fuzzy C-Means. Pada penelitian ini, Kapoor & Singhal (2017) mengaplikasikan data tersusun dan tidak tersusun untuk melihat pengaruh pada jumlah iterasi yang dibutuhkan dan juga waktu waktu tempuh yang dialami model tersebut. Hasil dari penelitian ini menunjukkan bahwa susunan data yang sudah tersusun membutuhkan jumlah iterasi dan waktu tempuh yang lebih sedikit untuk masing-masing algoritma dibandingkan dengan susunan data tidak tersusun. Fuzzy C-Means membutuhkan jumlah iterasi dan waktu tempuh paling sedikit, disusul oleh K-Means dan K-Means++ menempati posisi akhir. Hal ini membuat penerapan algoritma Fuzzy C-Means menjadi menarik karena dengan kecilnya kebutuhan jumlah iterasi dan waktu tempuh dalam pelatihan model, akan menghemat komputasi dan waktu yang diperlukan untuk melatih model.

## 3. Metodologi Penelitian

### 3.1. Data dan Sumber Data

Data yang digunakan pada penelitian ini diambil dari *platform* Kaggle. Data ini memiliki 7 atribut dan berjumlah 660 baris. Deskripsi lengkap atribut data yang digunakan dijelaskan pada Tabel 1 berikut.

Tabel 1. Tabel atribut *dataset Credit Card Customer*

Credit Card Customer Attributes		
No	Label	Deskripsi
1	SI_No	Nomor Identifikasi Pelanggan: Berfungsi sebagai indeks untuk nilai-nilai data.
2	Customer Key	Kunci Pelanggan
3	AvgCreditLimit	Rata-rata Limit Kartu Kredit Pelanggan
4	TotalCreditCards	Total Jumlah Kartu Kredit yang Dimiliki Pelanggan
5	Totalvisitsbank	Total Jumlah Kunjungan ke Bank oleh Pelanggan
6	Totalvisitsonline	Total Kunjungan Online oleh Pelanggan Bank
7	Totalcallsmade	Total Panggilan yang Dilakukan Pelanggan ke Bank

### 3.2. Langkah-langkah penerapan FCM

Pada penelitian ini, langkah langkah penerapa FCM adalah sebagai berikut:

1. Memuat data.
2. Memahami data.
3. Membersihkan data dan pengaturan fitur.
4. Membangun model FCM.
5. Mencari nilai  $c$  terbaik dari rentang  $2 \leq c < 10$  dengan menggunakan metrik evaluasi Final Partition Coefficient, Partition Entropy, dan Xie-Beni Index.
6. Melakukan *clustering*.
7. Melakukan reduksi dimensi dengan Principal Component Analysis (PCA) untuk memudahkan visualisasi.
8. Visualisasi hasil *clustering*.

### 3.3. Perhitungan Manual

Pada perhitungan manual ini akan dicontohkan untuk mencari klaster dari 1 baris data dari *dataset* yang memiliki 5 fitur (penelitian ini menggunakan 5 dari 7 fitur yang tersedia berdasarkan hasil analisa pemahaman data).

Data :

- $x_1 = (0.2, 0.6, 0.2, 0.2, 0.4)$
- $x_2 = (0.5, 0.5, 0.0, 0.8, 0.3)$

Parameter:

- $c = 2$
- $m = 2$
- $\varepsilon = 0,01$

Langkah – langkah:

1. Inisialisasi matriks keanggotaan  $U^{(0)}$ :

Misalkan kita inisialisasi nilai keanggotaan awal  $U^{(0)}$  secara acak, dengan syarat bahwa setiap data  $x_k$ , jumlah keanggotaan terhadap semua cluster adalah 1. Maka:

Table 1. Matriks Keanggotaan

Cluster	$x_1$	$x_2$
C1	0.6	0.5
C2	0.4	0.5

2. Hitung *centroid* kluster

Berdasarkan rumus (1) maka perhitungan *centroid* masing-masing kluster adalah:

- Untuk  $v_1$ :

$$v_1 = \frac{(0.6^2)(x_1) + (0.5^2)(x_3)}{0.6^2 + 0.5^2}$$

$$v_1 = \frac{(0.6^2)(0.2, 0.6, 0.2, 0.2, 0.4) + (0.5^2)(0.5, 0.5, 0.0, 0.8, 0.3)}{0.6^2 + 0.5^2}$$

$$v_1 = (0.323, 0.559, 0.118, 0.446, 0.359)$$

- Untuk  $v_2$ :

$$v_2 = \frac{(0.4^2)(x_1) + (0.5^2)(x_3)}{0.4^2 + 0.5^2}$$

$$v_2 = \frac{(0.4^2)(0.2, 0.6, 0.2, 0.2, 0.4) + (0.5^2)(0.5, 0.5, 0.0, 0.8, 0.3)}{0.4^2 + 0.5^2}$$

$$v_2 = (0.383, 0.539, 0.078, 0.566, 0.339)$$

Jadi dari hasil perhitungan, didapatkan *centroid* klaster 1 adalah (0.323, 0.559, 0.118, 0.446, 0.359) dan *centroid* klaster 2 adalah (0.383, 0.539, 0.078, 0.566, 0.339).

### 3. Memperbarui matriks keanggotaan

Berdasarkan rumus (2), maka perhitungan untuk memperbarui matriks keanggotaan adalah sebagai berikut:

#### a. Hitung jarak masing-masing data terhadap *centroid* klaster.

Untuk  $x_1 = (0.2, 0.6, 0.2, 0.2, 0.4)$

- Jarak  $x_1$  ke klaster 1

$$d_{11} = \|x_1 - v_1\| = \sqrt{(0.2 - 0.323)^2 + (0.6 - 0.559)^2 + \dots}$$

$$d_{11} = \sqrt{0.0857} \approx 0.293$$

- Jarak  $x_1$  ke klaster 2

$$d_{12} = \|x_1 - v_2\| = \sqrt{(0.2 - 0.383)^2 + (0.6 - 0.539)^2 + \dots}$$

$$d_{12} = \sqrt{0.1902} \approx 0.436$$

Untuk  $x_2 = (0.5, 0.5, 0.0, 0.8, 0.3)$

- Jarak  $x_2$  ke klaster 1

$$d_{21} = \|x_2 - v_1\| = \sqrt{(0.5 - 0.323)^2 + (0.5 - 0.559)^2 + \dots}$$

$$d_{21} = \sqrt{0.1777} \approx 0.421$$

- Jarak  $x_2$  ke klaster 2

$$d_{22} = \|x_2 - v_2\| = \sqrt{(0.5 - 0.383)^2 + (0.5 - 0.539)^2 + \dots}$$

$$d_{22} = \sqrt{0.0774} \approx 0.278$$

#### b. Hitung keanggotaan baru

- Untuk  $\mu_{11}$ :

$$\mu_{11} = \frac{1}{\left(\frac{d_{11}}{d_{11}}\right)^2 + \left(\frac{d_{11}}{d_{12}}\right)^2}$$

$$\mu_{11} = \frac{1}{1 + \left(\frac{0.293}{0.436}\right)^2} = \frac{1}{1 + 0.451} = \frac{1}{1.451} \approx 0.689$$

- Untuk  $\mu_{12}$

$$\mu_{12} = 1 - \mu_{11} = 1 - 0.689 = 0.311$$

- Untuk  $\mu_{21}$

$$\mu_{21} = \frac{1}{\left(\frac{d_{21}}{d_{21}}\right)^2 + \left(\frac{d_{21}}{d_{22}}\right)^2}$$

$$\mu_{21} = \frac{1}{1 + \left(\frac{0.421}{0.278}\right)^2} = \frac{1}{1 + 2.296} = \frac{1}{3.296} \approx 0.303$$

- Untuk  $\mu_{22}$

$$\mu_{22} = 1 - \mu_{21} = 1 - 0.303 = 0.697$$

Maka matriks keanggotaan baru  $U^{(1)}$  adalah:

Table 2. Tabel keanggotaan baru setelah iterasi 1

Cluster	x <sub>1</sub>	x <sub>2</sub>
C1	0.689	0.303
C2	0.311	0.697

#### 4. Cek konvergensi

Untuk mengecek apakah perubahan sudah termasuk konvergen maka dilakukan perhitungan seperti berikut:

$$\|U^{(1)} - U^{(0)}\| = \sqrt{(0.689 - 0.6)^2 + (0.311 - 0.4)^2 + (0.303 - 0.5)^2 + (0.697 - 0.5)^2}$$

$$\|U^{(1)} - U^{(0)}\| = \sqrt{(0.089)^2 + (-0.089)^2 + (-0.197)^2 + (0.197)^2}$$

$$\|U^{(1)} - U^{(0)}\| = \sqrt{0.093460} \approx 0.3057$$

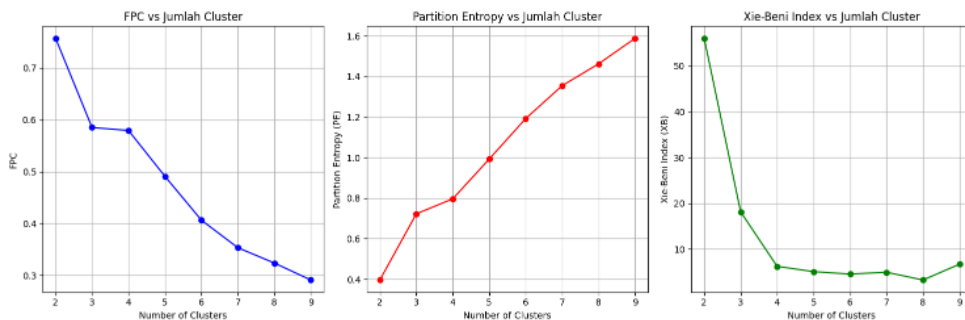


Hasil perhitungan selisih antara kedua matriks menghasilkan nilai 0.3057 yang dimana masih lebih besar dari  $\varepsilon = 0.01$ . Sehingga kita perlu melakukan iterasi selanjutnya dengan mengulangi langkah 2 – 4 hingga hasil selisih pengurangan matriks mencapai dibawah batas toleransi ( $\varepsilon = 0.01$ ).

## 4. Hasil dan Pembahasan

### 4.1. Hasil Penerapan

Pada penelitian ini, dilakukan pencarian nilai  $c$  terbaik dengan menerapkan algoritma Grid Search untuk  $2 \leq c < 10$  terhadap model Fuzzy C-Mean dengan parameter  $m$  (derajat fuzzy) = 2 error = 0.01, dan iterasi = 1000. Hasil yang didapatkan adalah sebagai berikut:



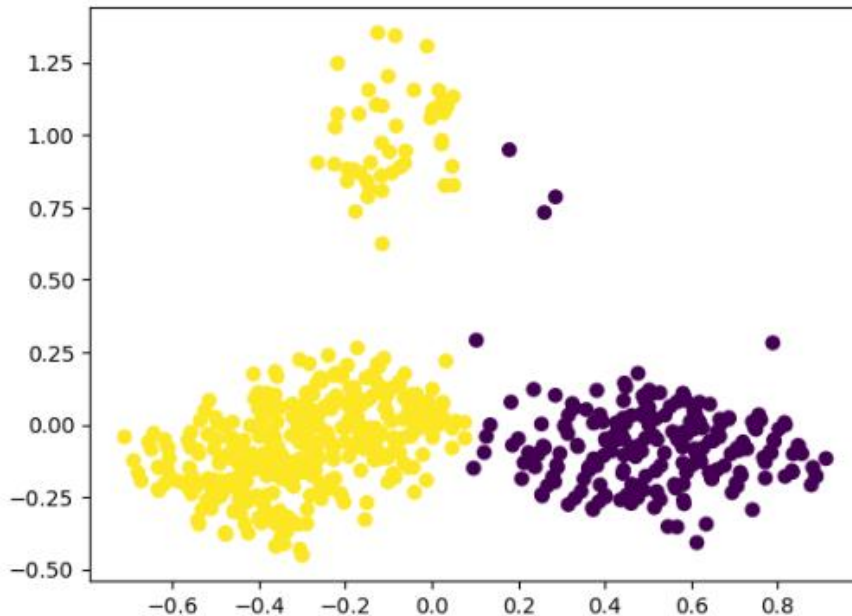
Jumlah cluster terbaik berdasarkan FPC: 2

Jumlah cluster terbaik berdasarkan PE: 2

Jumlah cluster terbaik berdasarkan XB: 8

Gambar 1. Hasil metode Grid Search untuk mencari nilai  $c$

Berdasarkan hasil Grid Search, maka ditentukan nilai  $c = 2$  untuk mendapatkan hasil yang optimal. Maka dilakukanlah *clustering* dengan nilai  $c = 2$  yang visualisasinya dapat dilihat pada Gambar 2.



Gambar 2. Visualisasi hasil clustering,  $c = 2$

Hasil *clustering* tersebut mendapatkan nilai Final Partition Coefficient = 0.757 (semakin mendekati 1.0, semakin baik).

## 4.2. Pembahasan Hasil

Dari eksperimen yang diterapkan pada penelitian ini, ditemukan bahwa nilai kluster yang optimum untuk data kartu kredit nasabah adalah 2. Nilai Final Partition Coefficient yang didapat dengan nilai kluster = 2 adalah 0,757. Nilai yang dapat terbilang cukup untuk sebuah studi kasus *clustering*.

## 5. Kesimpulan

### 5.1. Kesimpulan

Penelitian ini menunjukkan bahwa metode *clustering* Fuzzy C-Means merupakan metode yang cukup efektif untuk mengelompokan data. Dengan karakter fuzzy yang menawarkan keuntungan dibandingkan metode tradisional dalam situasi tertentu.

### 5.2. Rekomendasi Untuk Penelitian Selanjutnya

Penelitian selanjutnya direkomendasikan untuk melakukan eksplorasi lebih terhadap data sebelum melakukan *clustering*, dengan harapan dapat memahami lebih lanjut terkait data yang akan di-*clustering* dan meningkatkan validitas *cluster*.

## Daftar Pustaka

- Kapoor, A., & Singhal, A. (2017). A comparative study of K-Means, K-Means++ and Fuzzy C-Means clustering algorithms. *3rd IEEE International Conference On* .  
<https://doi.org/10.1109/CIACT.2017.7977272>
- Ross, T. J. (2010). Fuzzy Logic with Engineering Applications: Third Edition. In *Fuzzy Logic with Engineering Applications: Third Edition*.  
<https://doi.org/10.1002/9781119994374>

## Lampiran

- Kodingan Kaggle : <https://www.kaggle.com/code/rizkyfahreza/tugas-besar-komputasi-lunak-fuzzy-c-means>
- Dataset : <https://www.kaggle.com/datasets/aryashah2k/credit-card-customer-data>
- Link Youtube Presentasi: <https://youtu.be/v4yXrj7RVXA>

-