

# Efektivitas Algoritma Fuzzy C-Means dalam Klasterisasi Bidang Minat Anak Usia Dini Menggunakan Skor *silhouette*

Rafli Aditya Pramana<sup>1</sup>, Yuni Yamasari<sup>2</sup>

<sup>1,2</sup> Program Studi Teknik Informatika, Fakultas Teknik, Universitas Negeri Surabaya

<sup>1</sup>[rafliaditya.21007@mhs.unesa.ac.id](mailto:rafliaditya.21007@mhs.unesa.ac.id)

<sup>2</sup>[yuniyamasari@unesa.ac.id](mailto:yuniyamasari@unesa.ac.id)

**Abstrak**— Penelitian ini menganalisis efektivitas algoritma Fuzzy C-Means dalam mendeteksi minat anak usia dini menggunakan metrik evaluasi skor *silhouette*. Fokus utama penelitian adalah menguji kemampuan algoritma dalam mengelompokkan data pendidikan anak usia dini serta menilai akurasi hasil klasterisasi yang dihasilkan. Berdasarkan data yang diambil dari Taman Pendidikan Anak Sholeh Jombang, algoritma Fuzzy C-Means diimplementasikan untuk mengidentifikasi pola dan kecenderungan minat siswa. Temuan penelitian mengungkapkan bahwa pendekatan ini mampu memberikan pemahaman lebih komprehensif terkait minat anak, yang dapat menjadi dasar pengembangan strategi pembelajaran berbasis personalisasi. Studi ini merekomendasikan penggunaan Fuzzy C-Means sebagai solusi efektif untuk pemetaan minat usia dini. Untuk penelitian lanjutan, disarankan untuk mengeksplorasi kombinasi algoritma ini dengan teknik klasterisasi alternatif serta mengintegrasikan metrik penilaian lain guna memperkaya interpretasi terhadap data pendidikan anak.

**Kata Kunci**— Minat anak usia dini, Fuzzy C-Means, Evaluasi Kualitas Klasterisasi, Skor *Silhouette*.

## I. PENDAHULUAN

Dalam mempersiapkan sumber daya manusia yang berkualitas di Indonesia, Pendidikan memiliki peranan penting dalam mewujudkan hal tersebut. Pendidikan merupakan usaha yang secara sadar dilakukan dalam mewujudkan suasana belajar dan proses pembelajaran peserta didik sehingga dapat secara aktif mengembangkan potensi dirinya [1]. Pendidikan dapat diberikan kepada anak sejak berusia dini.

Anak usia dini merupakan individu yang berada dalam masa pertumbuhan dan perkembangan secara pesat sehingga disebut sebagai “usia emas” atau “the golden age” [2]. Pendidikan anak usia dini (PAUD) adalah Pendidikan yang ditujukan kepada anak hingga usia 6 tahun yang melakukan pembelajaran menggunakan rangsangan dan stimulasi untuk mengoptimalkan kemampuan dan tumbuh kembang anak [3].

Hasil Survei Sosial Ekonomi Nasional (Susenas) yang diselenggarakan Badan Pusat Statistik pada Maret 2019-2023, berkisar pada nilai 26-27%, indeks tersebut sempat turun pada tahun 2021 dikarenakan pandemi Covid-19, dan kembali

membalik pada tahun setelahnya [4]. Mendeteksi minat pada anak usia dini penting karena dapat memberikan dan menyediakan lingkungan belajar yang lebih sesuai, merencanakan aktivitas belajar yang menarik, dan mengembangkan potensi anak menjadi lebih maksimal.

Lembaga Pendidikan umumnya akan menyelenggarakan tes yang merupakan wadah dalam mengevaluasi hasil belajar siswa. Tes yang dilakukan sangatlah bervariasi dan menghasilkan *output* yang banyak. Perkembangan teknologi saat ini dapat dimanfaatkan dalam melakukan analisis data hasil tes tersebut agar dapat mengidentifikasi secara objektif dan akurat, serta dapat mempermudah dan mempersingkat waktu dalam pengolahan data tersebut.

Analisis data berbasis *clustering* merupakan pilihan yang tepat karena dapat mengungkapkan dan mengidentifikasi pola tertentu dalam sebuah data besar dan kompleks yang bila dilakukan pada Analisa manual akan kurang efektif. Data mining merupakan sebuah proses pencarian informasi yang dilakukan secara otomatis dalam sebuah dataset [5].

*Clustering* merupakan suatu metode yang digunakan untuk mengelompokkan data ke dalam beberapa kelompok kecil (klaster) berdasarkan kesamaan karakteristik dan sifat yang dimiliki oleh data tersebut [6]. Proses pembentukan klaster didasarkan pada tingkat kemiripan data dalam satu kelompok. Semakin tinggi tingkat kesamaan atau homogenitas data dalam suatu klaster, semakin optimal hasil yang diperoleh dari proses clustering tersebut.

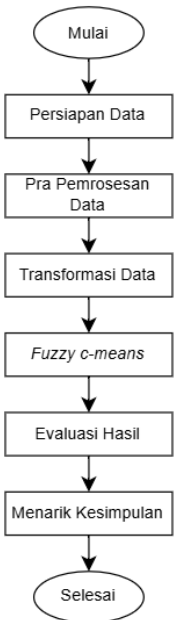
Dalam bidang Pendidikan, klasterisasi dapat digunakan dalam berbagai tujuan. Dalam penelitian yang dilakukan oleh Nafuri *et al.* (2022) melakukan teknik klasterisasi dalam mengklasifikasikan kinerja akademik siswa sehingga institusi pendidikan dapat menyediakan dukungan yang tepat dan efektif kepada siswa yang membutuhkan dan dapat meningkatkan kualitas pengajaran serta pengelolaan sumber daya dapat lebih efisien [7]. Penelitian yang dilakukan oleh Kurniawan *et al.* (2021) menggunakan metode *Clustering* dengan algoritma K-Means terhadap jumlah lulusan dari

berbagai jenjang dalam upaya mempermudah pemerintah untuk memberikan bantuan pendidikan [8]. Penelitian/1Saputra *et al.* (2023) juga menggunakan algoritma K-Means dalam mengklasterisasi nilai ujian siswa/1[9]. Dengan menggunakan metode *clustering* untuk mempermudah pemetaan data pendidikan siswa agar mendapatkan informasi untuk mengenali karakteristik siswa untuk dimanfaatkan sebagai acuan dalam pemberian pendidikan yang sesuai.

Penelitian ini mengacu pada data pendidikan yang telah dilakukan pada penelitian sebelumnya yaitu kelompok belajar Taman Pendidikan Anak Sholeh Jombang dimana merupakan sebuah penyelenggara Pendidikan anak berusia dini yang dilakukan oleh Amaliya, H. N., & Yamasari, Y. (2023) [10]. Penelitian ini juga bertujuan untuk mengevaluasi kinerja metode klasterisasi menggunakan algoritma fuzzy c-means menggunakan skor silhouette.

*Fuzzy c-means* (FCM) merupakan sebuah teknik pengklasteran data yang menentukan keberadaan data dalam suatu klaster oleh nilai keanggotaan [11]. *Silhouette score* merupakan sebuah metrik yang diperkenalkan oleh Peter J. Rousseeuw (1986) untuk mengukur seberapa baik suatu objek ditempatkan dalam klasternya dibandingkan dengan klaster lain [12]. Dengan demikian, hasil dari penelitian ini diharapkan dapat memberikan kontribusi dalam pengembangan minat pada anak usia dini..

II. METODE PENELITIAN



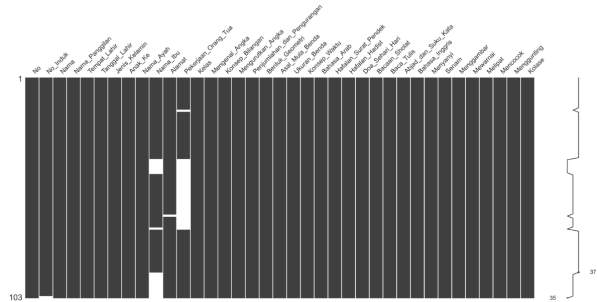
Gambar 1 Alur penelitian

Penelitian ini bertujuan untuk mengembangkan sistem evaluasi kualitas klasterisasi menggunakan algoritma Fuzzy C-Means dalam mendeteksi minat anak usia dini dengan memanfaatkan skor silhouette sebagai metrik evaluasi. Penelitian ini berfokus pada pengembangan suatu metode analisis data pendidikan, khususnya dalam mengidentifikasi pola dan minat anak usia dini. Alur tahapan yang akan dilaksanakan pada penelitian ini seperti pada Gambar 1.

A. Persiapan Data

Dalam proses ini, dataset didapat dari penelitian sebelumnya yang membahas tentang Clustering Bidang Keilmuan Menggunakan Kombinasi Metode Topsis dan Algoritma K-Means oleh Amaliya, H. N., & Yamasari, Y. (2023).

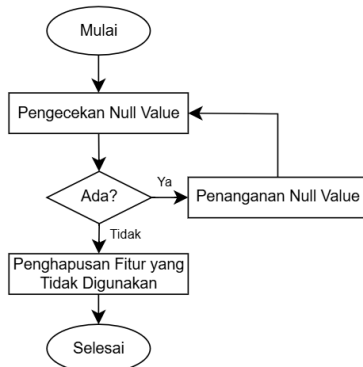
Populasi yang digunakan penulis merupakan seluruh data siswa, dengan sampel penelitian berupa data siswa yang berusia 4-6 tahun pada tahun ajaran 2022/2023. Pada tahap ini, peneliti akan melakukan eksplorasi data untuk memahami pola dan karakteristik data yang ada, sehingga dapat menentukan pendekatan yang tepat dalam proses klasterisasi.



Gambar 2 Dataset

Pada Gambar 2, merupakan sebuah dataset yang akan digunakan pada penelitian ini. Dataset tersebut memiliki 37 fitur yang mencakup berbagai informasi mengenai siswa, seperti: No\_Induk, Nama, Nama\_Panggilan, Tempat\_Lahir, Jenis\_Kelamin, hingga perolehan nilai siswa pada setiap mata pelajaran. Dengan melakukan analisis eksploratif terhadap data ini, peneliti dapat mengidentifikasi pola-pola atau karakteristik tertentu yang mungkin tidak terlihat pada penelitian sebelumnya, yang dapat digunakan untuk memperbaiki strategi pendidikan dan intervensi bagi siswa yang membutuhkan.

## B. Pra Pemrosesan Data



Gambar 3 Alur pra pemrosesan data

Tahapan kedua dalam penelitian ini yaitu dengan melakukan pra pemrosesan data. Setelah mengetahui informasi awal data, dilakukan pra pemrosesan pada data yang akan diimplementasikan kedalam sistem dengan menyesuaikan kebutuhan seperti tahapan pada Gambar 3.

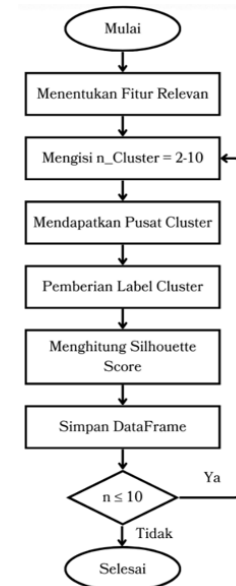
Pengecekan null value dilakukan dengan memvisualisasikan dataset untuk menampilkan sebaran null value dalam data. Setelah dilakukan penghapusan fitur yang tidak relevan dan penanganan null value pada fitur. Dilakukan pengecekan untuk memastikan integritas data sebelum melanjutkan ke tahap berikutnya. Dengan dataset yang sudah bersih, peneliti siap untuk memasuki tahapan transformasi data guna mempersiapkan data tersebut untuk analisis lebih lanjut dan proses klasterisasi.

## C. Transformasi Data

Tahapan ketiga dalam penelitian ini yaitu dilakukannya transformasi data. Transformasi data dilakukan agar data yang telah bersih dapat diproses dengan lebih efektif dan efisien. Dalam tahapan ini dilakukan beberapa langkah, dengan melakukan transformasi value pada fitur yang telah ditentukan dengan mengubah kategori data kategorikal menjadi numerik, agar dapat dilakukan komputasi.

## D. Fuzzy C-Means

Fuzzy C-Means (FCM) adalah sebuah algoritma pengklasteran data yang menentukan keberadaan setiap data dalam suatu klaster berdasarkan nilai keanggotaan. Prinsip kerja dari algoritma ini adalah dengan mengukur jarak antara titik data dengan pusat klaster yang bersangkutan. Algoritma FCM beroperasi dengan cara meminimalkan fungsi objektif  $J_m$  melalui serangkaian pembaruan iteratif terhadap nilai keanggotaan  $u_{ij}$  dan pusat klaster  $c_j$  [13].



Gambar 4 Alur fuzzy c-means

Pada Gambar 4 proses *fuzzy c-means* melibatkan langkah-langkah iteratif di mana nilai keanggotaan dan pusat klaster terus diperbarui hingga perubahan yang terjadi antara iterasi menjadi sangat kecil atau mencapai ambang batas yang telah ditentukan. Dengan demikian, proses iteratif ini berlanjut hingga nilai keanggotaan dan pusat klaster mencapai kondisi stabil tanpa perubahan signifikan, menandakan bahwa algoritma telah mencapai konvergensi yang diharapkan. Alur kerja *fuzzy c-means* dalam mengelompokkan data dengan cara:

1. Memasukkan fitur yang akan dikelompokkan ke dalam matriks  $X$  dengan ukuran  $n \times m$ . ( $n$  = banyak sampel data, dan  $m$  = jumlah variabel setiap data).
2. Tentukan jumlah klaster ( $C$ ), nilai pangkat bobot ( $w > 1$ ), maksimum iterasi ( $maxIter$ ), batas kesalahan terkecil ( $\xi$ ), Inisialisasi fungsi objektif awal ( $p$ ) dan iterasi awal ( $t=0$ ).
3. Buat praktisi awal  $U_{nc} = [\mu_{ik}]$  dimana, merupakan bilangan acak yang menyatakan derajat keanggotaan data  $i$  pada Cluster  $k$ , dengan kondisi:

$$0 \leq \mu_{ik} \leq 1 \text{ dan } \sum_{k=1}^c \mu_{ik} = 1 \forall i \quad (5)$$

4. Menghitung pusat Cluster  $V_{kj}$  untuk Klaster dan variabel  $j$  dengan persamaan:

$$V_k = \frac{\sum_{i=1}^n (\mu_{ik}^w) X_i}{\sum_{i=1}^n (\mu_{ik}^w)} \quad (6)$$

Dimana  $V_k$  merupakan pusat kluster ke-k,  $X_i$  merupakan titik data ke-I dan  $w$  = pangkat bobot.

5. Menghitung iterasi ke-t, dengan menghitung fungsi objektif  $P_t$  yang menggambarkan jumlah jarak antara data dan pusat kluster dengan persamaan:

$$P_t = \sum_{i=1}^n \sum_{k=1}^n (\mu_{ik}^w) ||X_i - V_k||^2 \quad 7)$$

6. Memperbaiki derajat keanggotaan  $\mu_{ik}$  dalam bentuk matriks partisi menggunakan persamaan:

$$\mu_{ik} = \frac{[\sum_{j=1}^m (X_{ij} - V_{kj})^2]^{\frac{1}{w-1}}}{\sum_{k=1}^c [\sum_{j=1}^m (X_{ij} - V_{kj})^2]^{\frac{1}{w-1}}} \quad 8)$$

Dimana:

$i=1,2,\dots,n$

$k=1,2,\dots,c$

$X_{ij}$  = sampel data ke-i, variabel ke-j

$V_{kj}$  = pusat kluster ke-k untuk variabel ke-j

$w$  = pangkat pembobotan

7. Mengecek kondisi berhenti pada iterasi dengan syarat jika  $(-P_t - P_{t-1} - \xi)$  maka berhenti sebaliknya lanjutkan iterasi  $t=t+1$ .

Setelah didapatkan hasil pengklasteran dengan algoritma *Fuzzy C-Means*, lakukan perhitungan *Silhouette Score* pada setiap data kluster kemudian simpan data kluster untuk digunakan pada analisis selanjutnya.

#### E. Analisis Hasil

Tahapan selanjutnya dalam proses ini adalah menganalisa hasil yang telah di uji sebelumnya. Setelah mengumpulkan nilai *Silhouette Score* dari semua kluster yang digunakan, langkah selanjutnya dalam analisis hasil adalah memilih hasil klusterisasi terbaik untuk setiap algoritma berdasarkan nilai *Silhouette Score* tertinggi. Dengan kata lain, untuk setiap algoritma klusterisasi yang telah diterapkan, hasil klusterisasi yang memiliki nilai *Silhouette Score* paling tinggi akan dipilih sebagai hasil terbaik.

Proses seleksi ini sangat penting untuk memastikan bahwa hasil akhir dari klusterisasi adalah yang paling optimal. Dengan memilih klusterisasi yang memiliki nilai *Silhouette Score* tertinggi, agar dapat memastikan bahwa objek-objek dalam setiap kluster memiliki kesamaan yang tinggi dan terpisah dengan baik dari objek-objek di kluster lainnya. Ini memberikan gambaran yang jelas dan akurat tentang struktur data yang telah dianalisis, membantu kita memahami lebih baik pola dan hubungan dalam data.

#### F. Menarik Kesimpulan

Tahapan terakhir dalam penelitian ini yaitu dengan menarik kesimpulan berdasarkan hasil analisa data pengujian yang telah dilakukan. Dengan demikian, tahapan menarik kesimpulan mencakup analisis dan interpretasi hasil dari serangkaian tahapan yang dilakukan dalam penelitian, serta memberikan rekomendasi atau saran berdasarkan temuan dan kesimpulan yang diperoleh.

### III. HASIL DAN PEMBAHASAN

Subbab ini menjelaskan hasil dari setiap proses implementasi yang sudah dijabarkan peneliti pada bab 2 sebelumnya. Yang mana, proses dalam mengevaluasi klusterisasi algoritma *fuzzy c-means* dalam pada minat anak usia dini sebagai berikut.

#### A. Persiapan Data

Pada Gambar 5 dilakukan *import library*, diantaranya *pandas* berfungsi untuk memanipulasi data dalam bentuk *DataFrame*, memudahkan dalam analisis dan pengolahan *dataset* yang berkaitan dengan minat keilmuan anak. *Numpy* menyediakan dukungan untuk operasi matematis yang penting, sementara *matplotlib* dan *seaborn* memungkinkan visualisasi hasil *clustering*, sehingga pola minat anak dapat dipahami dengan lebih baik. Untuk menangani nilai yang hilang, *missingno* digunakan untuk memvisualisasikan pola *missing value*, yang penting untuk integritas data. *Scipy.stats* menyediakan fungsi statistik untuk pengujian hipotesis, mendukung analisis yang lebih mendalam.

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import math
import missingno as mso
import scipy.stats as stats
import skfuzzy as skfuzzy
import warnings
warnings.filterwarnings("ignore")

from sklearn.cluster import KMeans, DBSCAN, AgglomerativeClustering
from sklearn.preprocessing import LabelEncoder
from sklearn.preprocessing import StandardScaler
from scipy.spatial.distance import cdist
from sklearn import datasets
from sklearn.metrics import silhouette_samples, silhouette_score,
davies_bouldin_score
from yellowbrick.cluster import SilhouetteVisualizer
from pandas.plotting import scatter_matrix
from fcmmeans import FCM
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler, MinMaxScaler
from sklearn.cluster import DBSCAN
from sklearn.metrics import silhouette_score, silhouette_samples
```

Gambar 5 Import Library

Metode *Fuzzy C-Means (FCM)* dari *skfuzzy* diterapkan untuk mengelompokkan data dengan mempertimbangkan derajat keanggotaan, yang sangat relevan dalam konteks minat yang tidak selalu jelas. *yellowbrick.klaster* menyediakan *SilhouetteVisualizer*, yang memungkinkan visualisasi grafik *silhouette* untuk memberikan gambaran tentang seberapa baik klastering dilakukan. *pandas.plotting* memiliki fungsi *scatter\_matrix* yang membuat *scatter plots* antara semua pasangan kolom dalam *DataFrame*, memungkinkan analisis hubungan antara berbagai fitur dalam *dataset*.

*fcmeans import FCM* digunakan untuk menerapkan algoritme *clustering* berbasis *Fuzzy C-Means*, di mana setiap titik data dapat memiliki kecenderungan untuk menjadi anggota dari lebih dari satu klaster. Ini memberikan fleksibilitas dalam pengelompokan data yang kompleks dan tidak pasti. Selanjutnya dilakukan *import dataset*, seperti pada Gambar 6.

```
datanilai = pd.read_excel('Data_Siswa_Tapas_Nilai (1).xlsx')
datanilai
```

Gambar 6 Import Dataset

Data yang akan dianalisis disimpan dalam variabel *datanilai*, memudahkan peneliti untuk melakukan manipulasi dan analisis data. *DataFrame* digunakan sebagai struktur ideal untuk menyimpan data tabular, yang mencakup atribut siswa seperti nama, kelas, dan nilai. Fungsi *datanilai()* digunakan untuk menampilkan isi *DataFrame*, memastikan data terbaca dengan benar dan tidak ada nilai yang hilang atau format yang salah. Melihat data secara langsung membantu peneliti memahami konteks dan isi dataset dengan lebih baik.

## B. Pra Proses Data

Pada tahap ini, dilakukan pengecekan informasi pada data dengan menampilkan informasi penting seperti jumlah total entri, jumlah kolom, tipe data masing-masing kolom, dan jumlah nilai *non-null* dalam setiap kolom yang ditampilkan pada Gambar 7.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 103 entries, 0 to 102
Data columns (total 37 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   No                                     103 non-null    int64
1   No_induk                             102 non-null    float64
2   Nama                                  103 non-null    object
3   Nama_Panggilan                       103 non-null    object
4   Tempat_Lahir                         103 non-null    object
5   Tanggal_Lahir                       103 non-null    object
6   Jenis_Kelamin                       103 non-null    object
7   Anak_Ke                              103 non-null    int64
8   Nama_Ayah                            103 non-null    object
9   Nama_Ibu                             83 non-null     object
10  Alamat                               102 non-null    object
11  Pekerjaan_Orang_Tua                 69 non-null     object
12  Kelas                               103 non-null    object
13  Mengetahui_Angka                   103 non-null    float64
14  Konsep_Bilangan                   103 non-null    float64
15  Mengurutkan_Angka                 103 non-null    float64
16  Penjumlahan_dan_Pengurangan       103 non-null    float64
17  Bentuk_Geometri                   103 non-null    float64
18  Area_Dua_Benda                    103 non-null    float64
19  Ukuratan_Benda                    103 non-null    float64
20  Konsep_Waktu                      103 non-null    float64
21  Bahasa_Arab                       103 non-null    float64
22  Hafalan_Surat_Pendek              103 non-null    float64
23  Hafalan_Hadis                     103 non-null    float64
24  Dosa_Sebagai_Hak                  103 non-null    float64
25  Rakaat_Shalat                     103 non-null    float64
26  Rakaat_Tilas                       103 non-null    float64
27  Abjad_dan_Suku_Kata               103 non-null    float64
28  Bahasa_Inggria                    103 non-null    float64
29  Menyanyi                          103 non-null    float64
30  Senam                              103 non-null    float64
31  Menggambar                        103 non-null    float64
32  Mengetik                          103 non-null    float64
33  Mengetik                           104 non-null    float64
34  Mengetik                           103 non-null    float64
35  Menggunting                       103 non-null    float64
36  Kelas                             103 non-null    float64
dtypes: float64(25), int64(2), object(10)
memory usage: 29.9+ KB
```

Gambar 7 Informasi Dataset

Langkah selanjutnya dalam pra-pemrosesan data, adalah melakukan pembersihan. Tahap ini memeriksa nilai yang hilang karena sangat penting untuk memastikan kualitas data sebelum diterapkan dalam algoritma klasterisasi. Data yang telah diproses akan digunakan untuk memetakan minat keilmuan anak dengan mengidentifikasi pola dari nilai yang ada. Melalui analisis klasterisasi, peneliti dapat mengelompokkan siswa berdasarkan karakteristik nilai mereka, sehingga membantu dalam merancang strategi pendidikan yang lebih sesuai dan efektif. Dilakukan pengecekan persebaran *null value* dalam setiap fitur, hal tersebut berguna untuk mengambil sebuah keputusan dalam penanganan data agar hasil analisis yang dilakukan akan berkualitas. Persebaran *null value* dapat dilihat pada Gambar 2 atau Gambar 7.

Peneliti melakukan pembersihan dengan menghapus fitur No, Nama, Nama\_Panggilan, Tempat\_Lahir, Tanggal\_Lahir, Anak\_Ke, Nama\_Ayah, Nama\_Ibu, Alamat, Pekerjaan\_Orang\_Tua. Fitur tersebut dihapus karena tidak diperlukan dalam analisa yang akan dilakukan pada tahap selanjutnya dan menambahkan nilai pada baris yang mengandung nilai hilang (*NaN*) dalam *DataFrame*.

## C. Transformasi Data

Proses transformasi nilai dalam data merupakan langkah penting dalam persiapan data untuk analisis dalam penelitian. Dalam penelitian ini, konsistensi dan kualitas data menjadi kunci untuk menghasilkan analisis yang valid. Misalnya, jika nilai-nilai dalam *DataFrame* mewakili skor

atau kategori, memiliki tipe *integer* membantu dalam pengelompokan dan interpretasi data yang lebih jelas.

Peneliti melakukan transformasi nilai pada fitur Jenis\_Kelamin dengan menggantikan nilai yang semula "Perempuan" dan "Laki-Laki" menjadi 1 dan 2 seperti pada Gambar 8.

```
datanilai['Jenis_Kelamin'] = datanilai['Jenis_Kelamin'].replace({
    'Perempuan' : 1,
    'Laki-Laki' : 2
}).astype(int)
datanilai
```

Gambar 8 Transformasi Fitur Jenis\_Kelamin

Peneliti juga melakukan transformasi nilai pada fitur Kelas dengan menggantikan nilai semula 'PRA TK KELOMPOK A' diubah menjadi 1, 'PRA TK KELOMPOK B' menjadi 2, dan seterusnya hingga kelas 'TK B' menjadi 4 seperti pada Gambar 9.

```
datanilai['Kelas'] = datanilai['Kelas'].replace({
    'PRA TK KELOMPOK A' : 1,
    'PRA TK KELOMPOK B' : 2,
    'TK A' : 3,
    'TK B' : 4
}).astype(int)
datanilai
```

Gambar 9 Transformasi Fitur Kelas

Kemudian peneliti mengubah tipe data yang semula *float64* menjadi *int*. hal ini dilakukan agar memastikan semua fitur memiliki tipe data yang konsisten agar hasil pemrosesannya menjadi lebih akurat dengan cara seperti pada Gambar 10.

```
# Mengubah kolom float64 menjadi int
float_columns = datanilai.select_dtypes(include=['float64']).columns
datanilai[float_columns] = datanilai[float_columns].astype(int)

# Menampilkan tipe data setelah diubah
print(datanilai.dtypes)
```

Gambar 10 Transformasi Tipe Data

#### D. Pengolahan Data

Pada implementasi algoritma fcm dilakukan seperti pada Gambar 11 dengan tahapan pertama, fitur yang relevan dipilih dari *DataFrame* *datanilai*, memastikan bahwa hanya variabel signifikan yang digunakan untuk analisis. Setelah itu, data tersebut distandarasi menggunakan *StandardScaler*, yang menjamin bahwa semua fitur memiliki rata-rata 0 dan deviasi standar 1, sehingga mengurangi potensi bias yang diakibatkan oleh perbedaan skala antar fitur.

```
# Memilih fitur yang relevan
selected_features = [] # Masukkan Fitur yg diinginkan
datanilai_selected = datanilai[selected_features]
# Inisialisasi FCM dengan jumlah cluster yang diinginkan
n_clusters = x # Jumlah kluster yang diinginkan
fcm = FCM(n_clusters=n_clusters)
# Fit model pada data
fcm.fit(datanilai_standardized)
# Menghitung silhouette score
silhouette_avg = silhouette_score(datanilai_standardized, labels)
print(f'Silhouette Score (overall): {silhouette_avg}')
# Menyimpan ke file Excel
data_with_silhouette.to_excel('Nama File.xlsx', index=False)
```

Gambar 11 Algoritma FCM

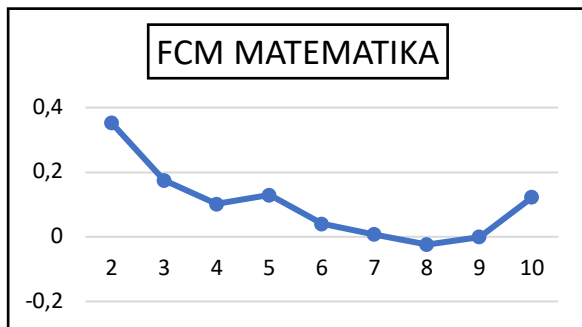
Selanjutnya, model *FCM* diinisialisasi dengan jumlah kluster yang diinginkan, dan model tersebut dilatih menggunakan data yang telah distandarasi. Setelah pelatihan, model menghasilkan label kluster untuk setiap data point dan posisi *centroid* untuk setiap kluster. Evaluasi kualitas *clustering* dilakukan dengan menghitung *silhouette score*, yang memberikan gambaran tentang seberapa baik pemisahan antara kluster. Rata-rata *silhouette score* diukur dan ditampilkan, serta *silhouette score* dihitung untuk setiap data poin, memberikan informasi lebih dalam mengenai distribusi data dalam kluster.

Hasil *clustering*, termasuk label kluster dan *silhouette score*, disimpan dalam *DataFrame* baru dan diekspor ke file Excel untuk analisis lebih lanjut. Visualisasi hasil dilakukan dengan membuat scatter plot, di mana data diplot dalam grafik 2D dengan warna yang berbeda untuk setiap kluster, dan centroid ditandai dengan simbol 'X' merah.

Pada tahap ini dilakukan pemrosesan algoritma yang akan terbagi menjadi beberapa bidang minat untuk melihat kecocokan minat pada anak usia dini. Pembagian bidang minat akan dibagi kedalam bidang berikut:

##### 1. Bidang Matematika

Pada percobaan untuk bidang matematika dilakukan menggunakan fitur 'Mengenal\_Angka', 'Konsep\_Bilangan', 'Mengurutkan\_Angka', 'Penjumlahan\_dan\_Pengurangan', 'Bentuk\_Geometri', 'Asal\_Mula\_Benda', 'Ukuran\_Benda', 'Konsep\_Waktu'.

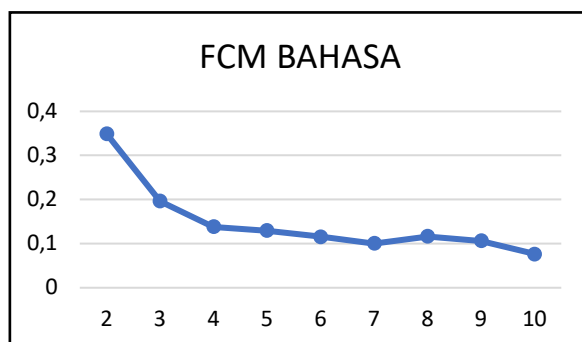


Gambar 12 Grafik Skor *Silhouette* Bidang Matematika

Pada Gambar 12 *fuzzy c-means* (FCM) menunjukkan bahwa skor *silhouette* bervariasi tergantung pada jumlah kluster yang diuji. Nilai tertinggi, 0.352, dicapai dengan pembagian ke dalam 2 kluster, yang menandakan pemisahan yang baik. Namun, nilai *silhouette* menurun seiring dengan peningkatan jumlah kluster, mencapai nilai terendah -0.024 pada pembagian ke dalam 8 kluster.

#### 2. Bidang Bahasa

Pada percobaan untuk bidang matematika dilakukan menggunakan fitur 'Bahasa\_Arab', 'Bahasa\_Ingggris', 'Hafalan\_Surat\_Pendek', 'Hafalan\_Hadist', 'Doa\_Sehari\_Ha ri', 'Bacaan\_Sholat', 'Baca\_Tulis', 'Abjad\_dan\_Suku\_Kata'.

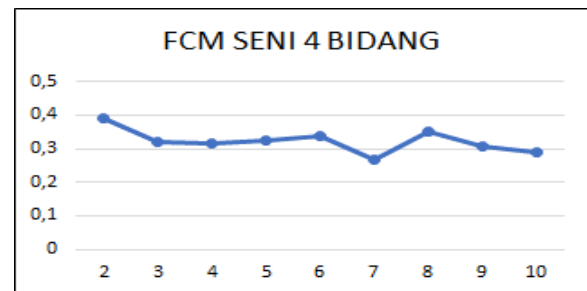


Gambar 13 Grafik Skor *Silhouette* Bidang Bahasa

Pada Gambar 13 algoritma *Fuzzy C-Means* (FCM), nilai *Silhouette* menunjukkan bahwa performa terbaik dicapai dengan pembagian 2 kluster, dengan nilai 0.357. Namun, seiring bertambahnya jumlah kluster, nilai *Silhouette* cenderung menurun, namun mencapai nilai terendah 0.068 pada pembagian 10 kluster.

#### 3. Bidang Seni

Pada percobaan untuk bidang seni dilakukan menggunakan fitur 'Menyanyi', 'Menggambar', 'Mewarnai', 'Kolase'.

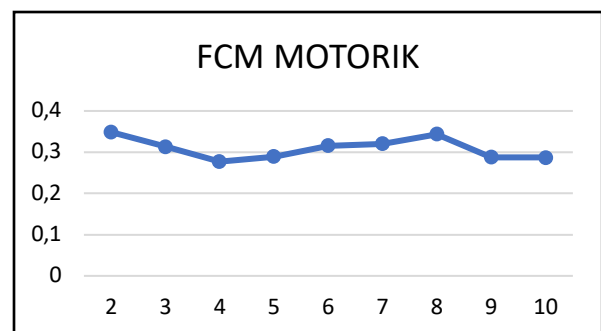


Gambar 14 Grafik Skor *Silhouette* Bidang Seni

Pada Gambar 14, algoritma *fuzzy c-means* (FCM) mencatat nilai *silhouette* terbaik pada pembagian 2 kluster, yaitu sebesar 0,390. Secara umum, nilai *silhouette* cenderung stabil untuk jumlah kluster yang lebih sedikit, meskipun terjadi penurunan nilai seiring dengan peningkatan jumlah kluster, mencapai titik terendah 0,269 pada pembagian 7 kluster dan terjadi peningkatan kembali pada 8 kluster namun skor *silhouette* tetap menurun.

#### 4. Bidang Motorik

Pada percobaan untuk bidang motorik dilakukan menggunakan fitur 'Senam', 'Melipat', 'Mencocok', 'Menggunting'.



Gambar 15 Grafik Skor *Silhouette* Bidang Motorik

Pada algoritma *Fuzzy C-Means* (FCM), nilai *Silhouette* tertinggi dicapai pada 2 kluster dengan nilai 0.349, menunjukkan pemisahan yang baik. Nilai *Silhouette* cenderung menurun dengan bertambahnya jumlah kluster, mencapai 0.277 pada 4 kluster. Ini menunjukkan bahwa FCM efektif untuk jumlah kluster yang lebih sedikit tetapi kurang optimal untuk jumlah kluster yang lebih tinggi.

#### E. Analisis Hasil

Tahapan analisis hasil dilakukan dengan mengumpulkan dan mengevaluasi nilai *Silhouette Score* dari setiap percobaan yang telah dilakukan pada empat bidang, yaitu Matematika, Bahasa, Seni, dan Motorik.

Pada setiap bidang, nilai *Silhouette Score* tertinggi diidentifikasi sebagai hasil klusterisasi terbaik. Misalnya,

pada bidang Matematika, nilai tertinggi adalah 0,352 pada 2 klaster; pada bidang Bahasa, nilai tertinggi adalah 0,357 pada 2 klaster; pada bidang Seni, nilai tertinggi adalah 0,390 pada 2 klaster; dan pada bidang Motorik, nilai tertinggi adalah 0,349 pada 2 klaster.

Selain mengidentifikasi nilai tertinggi, tren nilai Silhouette Score terhadap peningkatan jumlah klaster juga dianalisis. Pada semua bidang, terlihat bahwa nilai Silhouette cenderung menurun seiring dengan peningkatan jumlah klaster. Misalnya, pada bidang Matematika, nilai Silhouette turun hingga -0,024 pada 8 klaster; pada bidang Bahasa, nilai turun hingga 0,068 pada 10 klaster; pada bidang Seni, nilai turun hingga 0,269 pada 7 klaster; dan pada bidang Motorik, nilai turun hingga 0,277 pada 4 klaster.

Berdasarkan nilai Silhouette Score tertinggi, hasil klasterisasi terbaik dipilih untuk setiap bidang. Hasil ini menunjukkan bahwa algoritma Fuzzy C-Means (FCM) paling efektif ketika digunakan dengan jumlah klaster yang lebih kecil, khususnya 2 klaster. Hasil analisis menunjukkan bahwa pembagian data menjadi 2 klaster memberikan pemisahan yang paling optimal di semua bidang. Hal ini mengindikasikan bahwa struktur data pada keempat bidang lebih cocok untuk dikelompokkan dalam jumlah klaster yang sedikit, di mana objek-objek dalam klaster memiliki kesamaan yang tinggi dan terpisah dengan baik dari klaster lainnya.

#### IV. KESIMPULAN

Berdasarkan penelitian yang telah dilakukan, berikut adalah kesimpulan dari penelitian ini:

1. Algoritma Fuzzy C-Means (FCM) menunjukkan performa terbaik ketika digunakan dengan 2 klaster di semua bidang yang diuji (Matematika, Bahasa, Seni, dan Motorik). Nilai Silhouette Score tertinggi dicapai pada 2 klaster, yaitu 0,352 (Matematika), 0,357 (Bahasa), 0,390 (Seni), dan 0,349 (Motorik).
2. Nilai Silhouette Score cenderung menurun seiring dengan peningkatan jumlah klaster. Hal ini menunjukkan bahwa kompleksitas pengelompokan yang lebih tinggi (dengan jumlah klaster lebih banyak) tidak selalu menghasilkan pemisahan data yang lebih baik. Penurunan nilai Silhouette yang signifikan terjadi pada jumlah klaster yang lebih besar
3. Data pada keempat bidang lebih cocok untuk dikelompokkan dalam jumlah klaster yang sedikit (2 klaster). Hal ini menunjukkan bahwa struktur data yang dianalisis memiliki karakteristik yang lebih sederhana dan

dapat dijelaskan dengan baik melalui pembagian klaster yang minimal.

#### DAFTAR PUSTAKA

- [1] A. Rahman, S. A. Munandar, A. Fitriani, Y. Karlina, and Yumriani, "Pengertian Pendidikan, Ilmu Pendidikan dan Unsur-Unsur Pendidikan," *Al Urwatul Wutsqa Kaji. Pendidik. Islam*, vol. 2, no. 1, pp. 1–8, 2022.
- [2] D. L. Trenggonowati and K. Kulsum, "Analisis Faktor Optimalisasi Golden Age Anak Usia Dini Studi Kasus Di Kota Cilegon," *J. Ind. Serv.*, vol. 4, no. 1, pp. 48–56, 2018, doi: 10.36055/jiss.v4i1.4088.
- [3] I. K. TANU, "Pentingnya Pendidikan Anak Usia Dini Agar Dapat Tumbuh Dan Berkembang Sebagai Generasi Bangsa Harapan Di Masa Depan," *Adi Widya J. Pendidik. Dasar*, vol. 2, no. 2, p. 19, 2019, doi: 10.25078/aw.v2i2.960.
- [4] Badan Pusat Statistik, "Statistik Pendidikan 2023," *Badan Pus. Stat.*, vol. 12, pp. i–242, 2023, [Online]. Available: <https://www.bps.go.id/id/publication/2022/11/25/a80bdf8c85bc28a4e6566661/statistik-pendidikan-2022.html>
- [5] R. Sebastian, A. Pandu Kusuma, and W. Dwi Puspitasari, "Penggunaan Metode K-Nearest Neighbor Untuk Mendiagnosa Kerusakan Laptop Dengan Teknik Data Mining," *JATI (Jurnal Mhs. Tek. Inform.)*, vol. 7, no. 2, pp. 1191–1196, 2023, doi: 10.36040/jati.v7i2.6776.
- [6] W. Lestari, "Clustering Data Mahasiswa Menggunakan Algoritma K-Means Untuk Menunjang Strategi Promosi (Studi Kasus: STMIK Bina Bangsa Kendari)," *Simkom*, vol. 4, no. 2, pp. 35–48, 2019, doi: 10.51717/simkom.v4i2.37.
- [7] T. D. Khang, N. D. Vuong, M. K. Tran, and M. Fowler, "Fuzzy C-means clustering algorithm with multiple fuzzification coefficients," *Algorithms*, vol. 13, no. 13, pp. 1–11, 2020, doi: 10.3390/A13070158.
- [8] R. Kurniawan, M. M. M. Mukarrob, and M. Mahradianur, "Klasterisasi Tingkat Pendidikan Di DKI Jakarta Pada Tingkat Kecamatan Menggunakan Algoritma K-Means," *Technol. J. Ilm.*, vol. 12, no. 4, pp. 234–239, 2021, doi: 10.31602/tji.v12i4.5633.
- [9] A. Z. Saputra, N. Suarna, and G. D. Lestari, "Klasterisasi Nilai Ujian Sekolah Menggunakan Metode Algoritma K-Means," *J. Janitra Inform. dan Sist. Inf.*, vol. 3, no. 1, pp. 1–9, 2023, doi: 10.25008/janitra.v3i1.153.
- [10] H. N. Amaliya and Y. Yamasari, "Clustering Bidang Keilmuan Menggunakan Kombinasi Metode Topsis dan Algoritma K Means," *JINACS J. Informatics adn Comput. Sci.*, vol. 04, pp. 405–413, 2023.
- [11] E. B. Diyah Puspitaningrum, "Komparasi Algoritma Fuzzy C-Means Dan Algoritma K-Nearest Neighbor



- Dalam Pengelompokan Rombongan Belajar Siswa Baru (Studi Kasus: Siswa Baru Madrasah Aliyah Negeri 01 Kota Bengkulu),” *J. Tek. Inform.*, vol. 7, no. 2, pp. 16–23, 2014, doi: 10.15408/jti.v7i2.1951.
- [12] P. J. Rousseeuw, “Silhouettes: A graphical aid to the interpretation and validation of cluster analysis,” *J. Comput. Appl. Math.*, vol. 20, no. C, pp. 53–65, 1987, doi: 10.1016/0377-0427(87)90125-7.
- [13] J. Tamaela, E. Sedyono, and A. Setiawan, “Cluster Analysis Menggunakan Algoritma Fuzzy C-means dan K-means Untuk Klasterisasi dan Pemetaan Lahan Pertanian di Minahasa Tenggara,” *J. Buana Inform.*, vol. 8, no. 3, pp. 151–160, 2017, doi: 10.24002/jbi.v8i3.1317.