

PERBANDINGAN FCM DAN HIERARCHICAL CLUSTERING PADA DATA UBI JALAR INDONESIA

Matthew Russel Paul ¹⁾ Teny Handhayani ²⁾

¹⁾²⁾ Teknik Informatika Universitas Tarumanagara
Jl. Letjen S. Parman No.1, Jakarta 11440 Indonesia
email : ¹⁾ matthew.535230100@stu.untar.ac.id, ²⁾ tenyh@fti.untar.ac.id

ABSTRAK

Penelitian ini membandingkan performa dua algoritma klasterisasi, yaitu *Fuzzy C-Means (FCM)* dan *Hierarchical Clustering*, dalam mengelompokkan data produksi ubi jalar di Indonesia berdasarkan luas panen, produksi, dan produktivitas. Data diperoleh dari situs resmi Kementerian Pertanian Indonesia dan mencakup rentang tahun 2010–2024. Setelah melalui tahap pra-pemrosesan, kedua algoritma diterapkan dan dievaluasi menggunakan metrik *Silhouette Score*, *Davies-Bouldin Index*, serta waktu komputasi. Hasil menunjukkan bahwa *Hierarchical Clustering* dengan metode *linkage Ward, Complete, dan Average* menghasilkan performa terbaik pada konfigurasi dua klaster dengan nilai *Silhouette* sebesar 0.9385 dan *DBI* sebesar 0.3878. Sementara itu, *FCM* memberikan hasil optimal pada dua klaster dengan *Silhouette* sebesar 0.9074, *DBI* 0.6446, dan waktu komputasi tercepat sebesar 0.032 detik. Namun, performa *FCM* menurun signifikan saat jumlah klaster bertambah. Penelitian ini menunjukkan bahwa *Hierarchical Clustering* lebih unggul dalam kestabilan dan struktur klaster, sedangkan *FCM* lebih efisien dalam waktu proses pada jumlah klaster kecil.

Key words

Fuzzy C-Means, Hierarchical Clustering, klasterisasi, ubi jalar, evaluasi klaster

1. Pendahuluan

Ubi jalar (*Ipomoea batatas*) merupakan salah satu komoditas pangan penting di Indonesia yang memiliki nilai strategis, baik sebagai sumber karbohidrat alternatif maupun sebagai bahan baku industri pangan dan pakan [1]. Produksi ubi jalar tersebar di berbagai wilayah dengan karakteristik geografis dan iklim yang beragam, menghasilkan variasi dalam produktivitas dan kualitas hasil panen [2]. Variabilitas ini menjadikan proses pengelompokan wilayah berdasarkan karakteristik produksi ubi jalar menjadi hal yang penting untuk keperluan perencanaan, distribusi, serta pengembangan komoditas ini secara optimal.

Dengan kemajuan teknologi informasi dan data terbuka, tersedia banyak data terkait sektor pertanian,

termasuk data produksi ubi jalar dari berbagai daerah di Indonesia [3]. Salah satu sumber data yang resmi dan relevan adalah situs Basis Data Statistik Pertanian yang mudah diakses. Data ini mencakup informasi kuantitatif yang dapat digunakan untuk menganalisis pola produksi antarwilayah, yang dalam konteks ini dapat dimanfaatkan untuk membentuk klaster atau kelompok wilayah dengan karakteristik produksi ubi jalar yang serupa.

Penelitian ini bertujuan untuk membandingkan performa dua metode klasterisasi data, yaitu *Fuzzy C-Means (FCM)* dan *Hierarchical Clustering*, dalam mengelompokkan data produksi ubi jalar dari berbagai wilayah di Indonesia. Metode *FCM* memungkinkan sebuah data untuk menjadi anggota dari lebih dari satu klaster dengan derajat keanggotaan tertentu, sedangkan *Hierarchical Clustering* membentuk struktur pohon dari data berdasarkan kemiripan antar data [4], [5]. Kedua metode memiliki pendekatan berbeda dalam membentuk klaster, sehingga perbandingan kinerjanya menjadi penting untuk diketahui dalam konteks data pertanian.

Dalam penelitian ini, data yang digunakan merupakan data produksi ubi jalar yang diunduh dari situs resmi Badan Data Statistik Pertanian. Data diproses dengan melakukan pembersihan. Selanjutnya, metode *FCM* dan *Hierarchical Clustering* diterapkan pada data tersebut. Evaluasi hasil klaster dilakukan menggunakan metrik seperti *Silhouette Score* dan *Davies-Bouldin Index* untuk menilai kualitas klaster yang terbentuk [6].

Penelitian terdahulu telah banyak menggunakan metode klasterisasi untuk analisis data pertanian [7], namun perbandingan langsung antara *FCM* dan *Hierarchical Clustering* dalam konteks komoditas ubi jalar masih jarang dilakukan. Beberapa studi menunjukkan bahwa *FCM* unggul dalam menangkap ketidakpastian data, sedangkan *Hierarchical Clustering* lebih mudah divisualisasikan [8], [9]. Oleh karena itu, kajian ini diharapkan dapat memberikan gambaran yang lebih komprehensif mengenai kelebihan dan kekurangan kedua metode tersebut ketika diterapkan pada data produksi pertanian.

Kontribusi utama dari makalah ini adalah memberikan dasar ilmiah dan praktis dalam pemilihan metode klasterisasi yang paling sesuai untuk menganalisis data produksi ubi jalar, sehingga dapat membantu pengambil kebijakan dalam menentukan wilayah prioritas

pengembangan. Selain itu, penelitian ini juga berkontribusi dalam memperkaya literatur di bidang pertanian presisi dan *data mining* dengan studi kasus berbasis data nasional yang aktual.

2. Metode Penelitian

Untuk menjelaskan tahapan penelitian ini secara sistematis, digunakan diagram untuk alur proses yang terdiri dari empat langkah utama, dimulai dari tahap mulai hingga selesai. Proses dimulai dengan pengumpulan data produksi ubi jalar dari sumber resmi. Setelah data terkumpul, dilakukan tahap pra-pemrosesan yang mencakup pembersihan data, serta penanganan data kosong agar sesuai untuk analisis. Tahap berikutnya adalah penerapan dua teknik klusterisasi, yakni *Fuzzy C-Means* dan *Hierarchical Clustering*, pada data yang telah diproses. Hasil kluster kemudian dianalisis menggunakan metrik evaluasi untuk membandingkan performa kedua metode. Setelah semua tahap tersebut dilalui, proses penelitian akan diselesaikan dengan penarikan kesimpulan berdasarkan hasil evaluasi.



Gambar 1 Metode Penelitian

2.1. Data

Data yang digunakan dalam penelitian ini diperoleh dari situs resmi milik Kementerian Pertanian Republik Indonesia melalui situs <https://bdsp2.pertanian.go.id/>, yang menyediakan informasi statistik sektor pertanian secara terbuka dan terstruktur. Rentang waktu pengumpulan data mencakup periode dari tahun 2010 hingga 2024, sehingga mencerminkan tren dan dinamika produksi ubi jalar dalam jangka panjang. Setiap entri data mencakup beberapa atribut utama, yakni luas panen (dalam hektar), total produksi (dalam ton), serta produktivitas (dalam satuan ton per hektar). Ketiga atribut tersebut dipilih karena mewakili indikator penting dalam menilai kinerja sektor budidaya ubi jalar di berbagai wilayah. Data ini kemudian menjadi dasar utama dalam proses analisis klusterisasi untuk mengelompokkan wilayah berdasarkan kemiripan karakteristik produksinya.

2.2. Pra-Pemrosesan

Pra-pemrosesan data merupakan tahapan krusial dalam proses analisis data yang bertujuan untuk meningkatkan kualitas dan konsistensi informasi sebelum dianalisis lebih lanjut [10]. Data mentah yang diperoleh dari sumber eksternal sering kali mengandung berbagai masalah, seperti nilai yang hilang, duplikasi, atau inkonsistensi dalam format [11]. Jika dibiarkan, kondisi ini dapat menyebabkan bias dalam hasil analisis dan

menurunkan performa algoritma yang diterapkan [12]. Oleh karena itu, sebelum dilakukan proses klusterisasi, data harus dibersihkan dan disesuaikan terlebih dahulu melalui serangkaian teknik seperti penghapusan entri tidak relevan, serta penanganan *missing value*. Tahap ini tidak hanya membantu menjaga integritas hasil, tetapi juga memastikan bahwa metode analisis yang digunakan dapat bekerja secara optimal dalam mengungkap pola tersembunyi dari data yang tersedia [13].

Tahap awal pra-pemrosesan dilakukan dengan menggunakan program yang dimulai dengan membaca data dari *file Excel* yang berisi informasi produksi ubi jalar di berbagai wilayah. Kolom-kolom yang merepresentasikan data untuk tahun 2023 dan 2024 dihapus karena dianggap belum stabil atau belum lengkap. Untuk menjaga integritas data asli, dibuat salinan data yang kemudian digunakan dalam proses pembersihan. Nilai nol dianggap sebagai indikator ketidakterisian data (*missing value*), sehingga seluruh nilai nol diubah menjadi NaN (*Not a Number*). Selanjutnya, dihitung persentase nilai yang hilang di setiap baris, dan baris-baris yang memiliki nilai hilang lebih dari 30% namun tidak sepenuhnya kosong disaring dan dihapus dari *dataset* utama. Setelah proses penyaringan, data kembali disalin untuk tahap *imputasi*. Untuk baris dengan nilai yang tidak sepenuhnya kosong, metode *forward fill* diikuti *backward fill* diterapkan secara horizontal guna mengisi kekosongan berdasarkan nilai terdekat dari tahun sebelumnya atau sesudahnya. Terakhir, seluruh nilai NaN yang tersisa dikembalikan menjadi nol agar konsisten dengan struktur awal data.

Langkah-langkah pra-pemrosesan ini dilakukan untuk memastikan kualitas dan validitas data sebelum digunakan dalam proses klusterisasi [14]. Keberadaan nilai nol atau kosong yang cukup signifikan dapat memengaruhi hasil analisis dan menurunkan akurasi model dalam mengenali pola [15]. Dengan menghapus data yang tidak informatif dan mengisi kekosongan secara sistematis, proses ini membantu menghasilkan data yang lebih bersih dan representatif. Selain itu, menghindari pengaruh bias dari baris-baris yang tidak lengkap juga mendukung evaluasi kinerja model secara adil dan akurat [16]. Dengan data yang telah diproses dengan baik, metode klusterisasi yang digunakan dalam penelitian ini dapat bekerja secara optimal dalam mengelompokkan wilayah berdasarkan kemiripan karakteristik produksinya [17].

2.3. Fuzzy C-Means

Dalam penelitian ini, digunakan algoritma *Fuzzy C-Means* (FCM) sebagai pendekatan klusterisasi berbasis partisi yang bersifat *soft clustering*, di mana setiap data tidak terbatas hanya menjadi anggota satu kluster, tetapi dapat memiliki derajat keanggotaan terhadap beberapa kluster sekaligus. Berbeda dengan algoritma klusterisasi konvensional seperti *K-Means* yang bersifat *hard clustering* (setiap data hanya masuk ke satu kluster secara mutlak), FCM memberikan fleksibilitas dengan

mengizinkan satu objek menjadi bagian dari beberapa kluster dengan tingkat keanggotaan yang bervariasi [18].

Pendekatan *fuzzy* ini menjadi lebih relevan terutama pada data yang memiliki batas kluster yang tidak jelas atau ketika pola distribusi data saling bertumpang tindih. Dalam konteks tersebut, FCM mampu menangkap kompleksitas dan ketidakpastian struktur data yang tidak bisa ditangani secara optimal oleh metode klasterisasi tradisional. Dibandingkan dengan metode seperti klasterisasi hierarkis, FCM menawarkan implementasi yang lebih sederhana serta efisiensi komputasi yang lebih baik, terutama saat diterapkan pada *dataset* berukuran sedang hingga besar.

FCM termasuk dalam kategori *unsupervised learning* dan dirancang untuk menangani *dataset* yang mungkin mengandung *outlier*, yaitu data yang posisinya jauh dari pusat-pusat kluster sehingga sulit untuk diklasifikasikan secara tegas [19]. Teknik ini merupakan pengembangan dari metode partisi dengan menambahkan konsep keanggotaan *fuzzy*, memungkinkan proses klasterisasi tetap berjalan meskipun distribusi antar kelompok tidak sepenuhnya terpisah.

Pada prinsipnya, setiap data dalam algoritma FCM akan diberi nilai derajat keanggotaan terhadap masing-masing kluster. Semakin dekat suatu titik data ke pusat suatu kluster, semakin tinggi pula nilai keanggotaannya terhadap kluster tersebut. Proses ini diulang secara iteratif hingga mencapai konvergensi [20].

Berikut merupakan penjelasan mengenai langkah-langkah algoritma FCM [21]:

1. Menentukan Data Masukan

Langkah awal adalah menyiapkan data *input* berupa matriks berukuran $n \times m$, di mana n menyatakan jumlah data (baris), dan m menunjukkan jumlah atribut (kolom). Setiap nilai pada matriks mewakili atribut tertentu dari satu entitas data.

2. Menentukan Parameter Awal

- c : jumlah kluster yang ingin dibentuk
- m : nilai pengaruh derajat keanggotaan
- \max_iter : batas maksimum jumlah iterasi
- ϵ : toleransi konvergensi (perubahan minimal fungsi objektif)
- t : indeks iterasi yang diawali dari 1
- P_0 : nilai awal fungsi objektif, biasanya diinisialisasi 0

3. Inisialisasi Derajat Keanggotaan Acak

$$\sum_{i=1}^c \mu_{ik} = 1 \quad \dots\dots\dots(1)$$

Keterangan:

c = jumlah kluster

k = indeks data dalam *dataset*

Bilangan acak μ_{ik} dibangkitkan untuk membentuk sebuah matriks partisi awal (U), dengan ketentuan bahwa jumlah nilai keanggotaan suatu data terhadap semua kluster harus berjumlah satu. Nilai ini akan

digunakan sebagai dasar dalam perhitungan berikutnya.

4. Menghitung Pusat Kluster (*Centroid*)

$$V_{kj} = \frac{\sum_{i=1}^n (\mu_{ik})^w * X_{ij}}{\sum_{i=1}^n (\mu_{ik})^w} \quad \dots\dots\dots(2)$$

Keterangan:

n = jumlah anggota dalam *dataset*

j = indeks fitur atau atribut ke- j

w = parameter pembobotan

Untuk tiap kluster, dilakukan perhitungan pusat kluster berdasarkan nilai-nilai keanggotaan *fuzzy*. Perhitungan ini dilakukan dengan mengambil rata-rata tertimbang dari setiap atribut data terhadap derajat keanggotaannya pada kluster tersebut.

5. Menghitung Fungsi Objektif

$$P_t = \sum_{i=1}^n \sum_{k=1}^c \left(\left[\sum_{j=1}^m (X_{ij} - V_{kj})^2 \right] (X_{ij})^w \right) \quad \dots\dots(3)$$

Pada setiap iterasi, dihitung nilai fungsi objektif sebagai ukuran total jarak antara data dengan pusat kluster, yang telah diberi bobot berdasarkan nilai keanggotaan *fuzzy*. Nilai ini akan digunakan untuk memeriksa konvergensi.

6. Menghitung Perubahan Matriks Partisi dan Memperbarui Derajat Keanggotaan

$$\mu_{ik} = \frac{\left[\sum_{j=1}^m (X_{ij} - V_{kj})^2 \right]^{\frac{-1}{w-1}}}{\sum_{k=1}^c \left[\sum_{j=1}^m (X_{ij} - V_{kj})^2 \right]^{\frac{-1}{w-1}}} \quad \dots\dots\dots(4)$$

Matriks partisi U diperbarui berdasarkan jarak setiap data ke pusat kluster. Semakin dekat suatu data ke sebuah pusat kluster, maka semakin tinggi pula nilai keanggotaannya pada kluster tersebut.

7. Periksa Konvergensi

Jika selisih nilai fungsi objektif antar iterasi kurang dari nilai ϵ , atau jumlah iterasi telah mencapai batas maksimal, maka proses dihentikan. Jika tidak, iterasi dilanjutkan kembali dari langkah perhitungan pusat kluster.

2.4. Hierarchical Clustering

Digunakan juga *Hierarchical Clustering* yaitu metode klasterisasi yang membentuk struktur bertingkat atau hierarki dari data tanpa perlu menentukan jumlah kluster di awal. Metode ini sangat berguna ketika pengguna ingin mengeksplorasi struktur alami dari data atau melihat bagaimana kelompok-kelompok terbentuk secara bertahap [22]. Berbeda dengan *Fuzzy C-Means* atau *K-Means* yang termasuk dalam *partitioning method*, *Hierarchical Clustering* menghasilkan model berupa *dendrogram*, yaitu representasi visual yang

menggambarkan proses penggabungan atau pemisahan data secara berurutan. Meskipun cenderung lebih mahal dari sisi komputasi, terutama pada *dataset* besar, kelebihan utamanya terletak pada transparansi dan kemudahan interpretasi hasil, karena setiap proses pengelompokan dapat ditelusuri dari awal hingga akhir.

Hierarchical Clustering memiliki dua pendekatan utama, yaitu *agglomerative* (penggabungan dari bawah ke atas) dan *divisive* (pemecahan dari atas ke bawah). Pendekatan yang paling umum dan akan digunakan di penelitian ini adalah *agglomerative*, di mana setiap data awalnya dianggap sebagai kluster tersendiri, kemudian secara bertahap digabung berdasarkan tingkat kemiripan hingga seluruh data menjadi satu kluster besar.

Ukuran kemiripan atau jarak antar kluster dapat dihitung menggunakan berbagai metode seperti *single linkage* (jarak terdekat), *complete linkage* (jarak terjauh), *average linkage* (rata-rata jarak), atau *ward linkage* (berdasarkan peningkatan variansi total) [23]:

- *Single Linkage*
Mengukur jarak antara dua kluster berdasarkan pasangan data terdekat dari masing-masing kluster, dengan rumus:

$$d(A, B) = \min_{ij} \{d(X_i, Y_j)\} \quad \dots\dots\dots(5)$$

Keterangan:

A = elemen kluster A

B = elemen kluster B

$d(X_i, Y_j)$ = jarak antara titik

Metode ini cocok untuk bentuk kluster memanjang, namun sensitif terhadap *outlier*.

- *Complete Linkage*
Menentukan jarak berdasarkan pasangan data terjauh antara dua kluster:

$$d(A, B) = \max_{ij} \{d(X_i, Y_j)\} \quad \dots\dots\dots(6)$$

Hasil kluster cenderung lebih kompak, namun bisa terdistorsi oleh data yang tersebar.

- *Average Linkage*
Mengukur jarak sebagai rata-rata semua jarak pasangan data dari dua kluster:

$$d(A, B) = \frac{1}{k \cdot l} \sum_{i=1}^k \sum_{j=1}^l d(X_i, Y_j) \quad \dots\dots\dots(7)$$

Keterangan:

k = jumlah elemen kluster A

l = jumlah elemen kluster B

Memberikan hasil yang lebih seimbang antara jarak dekat dan jauh.

- *Ward Linkage*
Metode *Ward linkage* merupakan salah satu pendekatan dalam teknik klusterisasi hierarki yang tidak hanya mempertimbangkan jarak antar objek, tetapi juga memperhatikan seberapa besar peningkatan variansi total dalam suatu kluster setelah

dua kluster digabungkan. Tujuannya adalah meminimalkan pertambahan *sum of squared errors* (SSE) akibat penggabungan tersebut.

Secara matematis, total SSE dalam satu kluster dapat dinyatakan sebagai:

$$SSE = \sum_{j=1}^p \left(\sum_{i=1}^n X_{ij}^2 - \frac{1}{n} \left(\sum_{i=1}^n X_{ij} \right)^2 \right) \quad \dots\dots\dots(8)$$

Keterangan:

X_{ij} = nilai fitur ke-j dari data ke-i

n = jumlah data dalam satu kluster

p = jumlah fitur

Dengan kata lain, metode ini menggabungkan dua kluster yang menghasilkan kenaikan terkecil dalam SSE. Strategi ini cenderung mempertahankan struktur internal kluster dengan baik, sehingga sering dianggap lebih stabil dan akurat dibandingkan metode linkage lainnya.

2.5. Metode Evaluasi

Dalam proses evaluasi hasil klusterisasi, penting untuk menggunakan metrik yang dapat mengukur seberapa baik objek data dikelompokkan dalam kluster yang telah terbentuk. Salah satu metrik yang efektif dan sering digunakan adalah *silhouette score*. Metrik ini mengukur kualitas kluster berdasarkan konsistensi internal dan pemisahan antar kluster.

Prinsip dasar *silhouette score* adalah membandingkan jarak rata-rata antar objek dalam satu kluster dengan jarak rata-rata objek tersebut ke kluster terdekat berikutnya. Nilai *silhouette* untuk sebuah titik data i dihitung menggunakan rumus [24]:

$$s(i) = \frac{b(i) - a(i)}{\max(a(i), b(i))} \quad \dots\dots\dots(9)$$

Keterangan:

$a(i)$ = rata-rata jarak antara titik data i dengan semua titik lain dalam kluster yang sama,

$b(i)$ = rata-rata jarak antara titik data i dengan titik-titik pada kluster terdekat yang berbeda.

Nilai $s(i)$ berada dalam rentang -1 hingga 1, di mana nilai mendekati +1 menunjukkan bahwa objek tersebut ditempatkan dengan baik dalam klusternya karena jaraknya ke kluster lain jauh lebih besar dibandingkan jarak ke objek dalam klusternya sendiri. Sebaliknya, nilai yang mendekati -1 menandakan bahwa objek tersebut mungkin salah klasifikasi atau berada di batas antar kluster [25].

Untuk menilai keseluruhan kualitas hasil klusterisasi, nilai *silhouette* rata-rata dari semua titik data digunakan. Nilai rata-rata ini memberikan gambaran seberapa baik klusterisasi data secara keseluruhan, di mana nilai tinggi menandakan pembentukan kluster yang baik dan jelas.

Penggunaan *silhouette score* sangat relevan terutama dalam konteks metode *soft clustering* seperti *Fuzzy C-Means* maupun metode *hard clustering* lainnya, karena dapat membantu menentukan jumlah kluster optimal dan mengukur efektivitas pemisahan antar kluster.

Selain *silhouette score*, salah satu metrik yang umum digunakan untuk mengevaluasi hasil klusterisasi adalah *Davies-Bouldin Index* (DBI). Indeks ini memberikan ukuran kuantitatif terhadap kualitas pembentukan kluster dengan mempertimbangkan baik jarak antar kluster maupun penyebaran data di dalam kluster itu sendiri.

Secara garis besar, *Davies-Bouldin Index* mengukur seberapa baik kluster-kluster yang terbentuk terpisah satu sama lain sekaligus menjaga agar anggota dalam kluster tetap kompak. Nilai DBI dihitung berdasarkan rata-rata rasio antara jarak dalam kluster (*intra-cluster distance*) dan jarak antar pusat kluster (*inter-cluster distance*) untuk setiap pasang kluster.

Rumus umum DBI adalah sebagai berikut [24]:

$$DBI = \frac{1}{c} \sum_{i=1}^c \max_{j \neq i} \left(\frac{S_i + S_j}{M_{ij}} \right) \dots\dots\dots(10)$$

Keterangan:

c = jumlah kluster,

S_i = ukuran penyebaran data (seperti rata-rata jarak titik data ke *centroid*) dalam kluster i,

M_{ij} = jarak antara *centroid* kluster i dan kluster j.

Indeks ini menilai, untuk setiap kluster, nilai maksimum dari rasio penyebaran gabungan dua kluster terhadap jarak antar pusat kluster mereka. Semakin kecil nilai DBI, semakin baik kualitas kluster karena menunjukkan kluster yang kompak dan terpisah jelas [26].

Kelebihan menggunakan *Davies-Bouldin Index* adalah kemampuannya memberikan gambaran evaluasi tanpa perlu label data, sehingga cocok dipakai untuk berbagai algoritma klusterisasi seperti *Fuzzy C-Means*, maupun *hierarchical clustering*. Penggunaan DBI juga efektif untuk membantu menentukan jumlah kluster optimal dengan membandingkan nilai indeks dari berbagai konfigurasi kluster.

3. Hasil Percobaan

Dalam penelitian ini, semua pengujian dilakukan di dalam *Google Colab*. Dilakukan pengujian terhadap dua metode klusterisasi, yaitu *Fuzzy C-Means* (FCM) dan *Hierarchical Clustering* (HC), untuk menentukan kinerja terbaik dalam mengelompokkan data produksi ubi jalar di Indonesia. Penilaian dilakukan menggunakan tiga indikator utama, yakni *Silhouette Score* yang mengukur kesesuaian internal antar objek dalam satu kluster, *Davies-Bouldin Index* (DBI) yang menilai pemisahan antar kluster, serta waktu komputasi mencerminkan

efisiensi algoritma dalam menyelesaikan proses klusterisasi.

Untuk menilai performa metode *Fuzzy C-Means* dalam proses klusterisasi data produksi ubi jalar, dilakukan serangkaian eksperimen dengan variasi jumlah kluster dari dua hingga sepuluh. Setiap konfigurasi diuji sebanyak lima kali guna memastikan konsistensi hasil.

Pada Tabel 1, Hasil menunjukkan bahwa konfigurasi dengan dua kluster memberikan performa terbaik secara konsisten. Rata-rata nilai *Silhouette* mencapai 0.9074, yang merupakan skor tertinggi di antara seluruh konfigurasi yang diuji. Angka ini mengindikasikan bahwa mayoritas data berada pada posisi yang tepat dalam klusternya dan memiliki jarak yang cukup jauh dari kluster lain. Selain itu, nilai DBI untuk kluster 2 sebesar 0.6446 tergolong rendah, menandakan bahwa hasil kluster tidak hanya padat secara internal tetapi juga cukup terpisah antar kelompok. Waktu komputasi untuk konfigurasi ini juga paling efisien, dengan rata-rata hanya 0.032 detik per percobaan.

Tabel 1 Fuzzy C-Means 2 Cluster

Eksperimen ke	Silhouette	Davies-Bouldin Index	Waktu Komputasi (detik)
1	0.9074	0.6446	0.05
2	0.9074	0.6446	0.03
3	0.9074	0.6446	0.02
4	0.9074	0.6446	0.03
5	0.9074	0.6446	0.03
Rata-Rata	0.9074	0.6446	0.032

Ketika jumlah kluster ditambah menjadi tiga pada Tabel 2, performa FCM mengalami penurunan meskipun tidak signifikan. Rata-rata *Silhouette* turun menjadi 0.8119 dan DBI naik ke angka 0.7085. Meskipun nilai-nilai tersebut masih dalam kategori baik, perbandingan dengan konfigurasi dua kluster menunjukkan bahwa pembentukan tiga kelompok mulai menghadapi tantangan dalam menjaga keseimbangan antara kepadatan dan pemisahan kluster. Waktu komputasi pun meningkat menjadi 0.056 detik, namun tetap dalam rentang yang dapat diterima.

Tabel 2 Fuzzy C-Means 3 Cluster

Eksperimen ke	Silhouette	Davies-Bouldin Index	Waktu Komputasi (detik)
1	0.8118	0.7084	0.04
2	0.8118	0.7084	0.07
3	0.8118	0.7084	0.06
4	0.8118	0.7084	0.06
5	0.8118	0.7084	0.05
Rata-Rata	0.8119	0.7085	0.056

Penurunan kualitas klusterisasi semakin terlihat pada konfigurasi empat kluster yang terlihat pada Tabel 3. Rata-rata *Silhouette* berada di angka 0.7578, sementara

nilai DBI meningkat menjadi 0.8242. Ini menunjukkan bahwa pembentukan empat kluster mulai menghasilkan kelompok yang kurang terdefinisi dengan baik dan lebih tumpang tindih. Hal ini dapat disebabkan oleh data yang sebenarnya hanya membentuk dua hingga tiga kelompok dominan secara alami, sehingga pemaksaan jumlah kluster yang lebih banyak justru menurunkan kualitas segmentasi.

Tabel 3 Fuzzy C-Means 4 Cluster

Eksperimen ke	Silhouette	Davies-Bouldin Index	Waktu Komputasi (detik)
1	0.7578	0.8242	0.15
2	0.7578	0.8242	0.14
3	0.7578	0.8242	0.13
4	0.7578	0.8242	0.10
5	0.7578	0.8242	0.09
Rata-Rata	0.7578	0.8242	0.122

Pada konfigurasi lima hingga tujuh kluster yang ditunjukkan pada Tabel 4, Tabel 5, dan Tabel 6, performa FCM semakin memburuk. Terlihat dari menurunnya *Silhouette Score* secara bertahap (0.6052, 0.5767, dan 0.5263) serta meningkatnya nilai DBI (0.9742, 1.0959, dan 1.2295). Angka-angka ini menunjukkan bahwa kluster yang terbentuk mulai kehilangan kohesi internal dan semakin saling beririsan. Waktu komputasi juga menunjukkan tren meningkat, terutama pada kluster tujuh, yang mencapai rata-rata 0.154 detik. Peningkatan jumlah kluster tidak hanya menambah kompleksitas, tetapi juga mengurangi kualitas kluster secara umum.

Tabel 4 Fuzzy C-Means 5 Cluster

Eksperimen ke	Silhouette	Davies-Bouldin Index	Waktu Komputasi (detik)
1	0.6052	0.9741	0.07
2	0.6052	0.9741	0.04
3	0.6052	0.9741	0.05
4	0.6052	0.9741	0.09
5	0.6052	0.9741	0.06
Rata-Rata	0.6052	0.9742	0.062

Tabel 5 Fuzzy C-Means 6 Cluster

Eksperimen ke	Silhouette	Davies-Bouldin Index	Waktu Komputasi (detik)
1	0.5766	1.0958	0.07
2	0.5766	1.0958	0.16
3	0.5766	1.0958	0.07
4	0.5766	1.0958	0.12
5	0.5766	1.0958	0.04
Rata-Rata	0.5767	1.0959	0.092

Tabel 6 Fuzzy C-Means 7 Cluster

Eksperimen ke	Silhouette	Davies-Bouldin Index	Waktu Komputasi (detik)
1	0.5263	1.2295	0.17
2	0.5263	1.2295	0.14
3	0.5263	1.2295	0.11
4	0.5263	1.2295	0.22
5	0.5263	1.2295	0.13
Rata-Rata	0.5263	1.2295	0.154

Eksperimen dengan delapan hingga sepuluh kluster yang ditampilkan pada Tabel 8, Tabel 9, dan Tabel 10 memperkuat kesimpulan tersebut. Rata-rata *Silhouette* menurun drastis hingga di bawah 0.45, dengan nilai terendah pada konfigurasi 10 kluster (0.3783). Sementara itu, DBI pada konfigurasi sembilan mencapai angka tertinggi sebesar 1.3526, menunjukkan bahwa sebagian besar kluster tidak hanya tidak padat, tetapi juga memiliki posisi yang berdekatan satu sama lain. Kombinasi nilai *silhouette* yang rendah dan DBI yang tinggi mencerminkan buruknya struktur kluster yang terbentuk. Waktu komputasi pun meningkat, dengan puncaknya pada konfigurasi sepuluh kluster yang membutuhkan rata-rata 0.166 detik.

Tabel 7 Fuzzy C-Means 8 Cluster

Eksperimen ke	Silhouette	Davies-Bouldin Index	Waktu Komputasi (detik)
1	0.4938	1.0441	0.30
2	0.4051	1.3203	0.07
3	0.4051	1.3203	0.04
4	0.4051	1.3203	0.05
5	0.4560	1.0374	0.20
Rata-Rata	0.4331	1.1491	0.132

Tabel 8 Fuzzy C-Means 9 Cluster

Eksperimen ke	Silhouette	Davies-Bouldin Index	Waktu Komputasi (detik)
1	0.3970	1.2255	0.08
2	0.3831	1.5210	0.05
3	0.3917	1.2697	0.08
4	0.3831	1.5210	0.10
5	0.3970	1.2255	0.15
Rata-Rata	0.3904	1.3526	0.092

Tabel 9 Fuzzy C-Means 10 Cluster

Eksperimen ke	Silhouette	Davies-Bouldin Index	Waktu Komputasi (detik)
1	0.3756	1.3018	0.11
2	0.3757	1.2135	0.25
3	0.3757	1.2135	0.19

4	0.3770	1.3927	0.17
5	0.3892	0.3892	0.11
Rata-Rata	0.3783	1.1022	0.166

Berdasarkan hasil percobaan tersebut, dapat disimpulkan bahwa *Fuzzy C-Means* memberikan performa optimal ketika digunakan dengan dua atau tiga klaster. Jumlah klaster yang lebih dari tiga justru menyebabkan penurunan kualitas klasterisasi, baik dari sisi validitas klaster maupun efisiensi waktu. Hal ini mengindikasikan bahwa data produksi ubi jalar Indonesia cenderung membentuk dua hingga tiga kelompok yang dominan secara alami, yang mencerminkan pola geografis atau produktivitas yang serupa di beberapa wilayah.

Setelah dilakukan evaluasi terhadap *Fuzzy C-Means*, analisis selanjutnya ditujukan untuk metode *Hierarchical Clustering* dengan variasi jumlah klaster dari dua hingga sepuluh. Berbeda dengan beberapa metode *clustering* lain yang memanfaatkan *random state* untuk inialisasi, *Hierarchical Clustering* tidak bergantung pada parameter acak tersebut, sehingga evaluasi dilakukan melalui variasi metode *linkage* sebagai bentuk variasi pengelompokan. Empat jenis *linkage* yang digunakan adalah *Ward*, *Complete*, *Average*, dan *Single*. Evaluasi dilakukan menggunakan metrik yang sama dengan metode sebelumnya, yaitu *Silhouette Score*, *Davies-Bouldin Index* (DBI), dan waktu komputasi. Hal ini bertujuan untuk memastikan konsistensi dalam perbandingan performa antar metode.

Pada konfigurasi dua klaster yang tercantum pada Tabel 10, *Hierarchical Clustering* menunjukkan performa yang sangat unggul. Seluruh metode *linkage* kecuali *Single* menghasilkan nilai *Silhouette* yang identik, yakni sebesar 0.9385 dengan DBI sebesar 0.3878. Sementara itu, *Single Linkage* mencatatkan nilai *Silhouette* sedikit lebih rendah (0.9206), tetapi justru menghasilkan DBI yang paling kecil di antara semua pengujian, yakni 0.0543. Hal ini menandakan bahwa pendekatan *Single Linkage* mampu membentuk klaster yang sangat padat dan jelas terpisah pada konfigurasi dua klaster, meskipun secara visual struktur *dendrogram* mungkin tidak serapi metode lain.

Tabel 10 Hierarchical Clustering 2 Cluster

Eksperimen	Silhouette	Davies-Bouldin Index	Waktu Komputasi (detik)
Ward	0.9385	0.3878	0.03
Complete	0.9385	0.3878	0.06
Average	0.9385	0.3878	0.03
Single	0.9206	0.0543	0.03

Ketika jumlah klaster ditingkatkan menjadi tiga seperti yang terlihat pada Tabel 11, performa tertinggi dicapai oleh *Complete*, *Average*, dan *Single Linkage* yang semuanya menghasilkan *Silhouette Score* sebesar 0.9310 dan DBI sebesar 0.2218. Hasil ini menunjukkan bahwa metode-metode tersebut berhasil mempertahankan pemisahan antar klaster dengan struktur yang relatif

stabil. *Ward Linkage* mulai menunjukkan penurunan performa dengan *Silhouette* sebesar 0.8124 dan DBI 0.7086. Meskipun demikian, waktu komputasi untuk semua metode tetap efisien, berada pada kisaran 0.03–0.04 detik.

Tabel 11 Hierarchical Clustering 3 Cluster

Eksperimen	Silhouette	Davies-Bouldin Index	Waktu Komputasi (detik)
Ward	0.8124	0.7086	0.03
Complete	0.9310	0.2217	0.04
Average	0.9310	0.2217	0.04
Single	0.9310	0.2217	0.04

Pada konfigurasi empat klaster di Tabel 12, *Single Linkage* kembali menunjukkan keunggulan dalam hal *Davies-Bouldin Index* yang sangat rendah (0.0410), jauh lebih baik dibandingkan *Ward* (0.5247), *Complete* (0.4142), maupun *Average* (0.4652). Namun secara *Silhouette Score*, *Complete* dan *Average Linkage* tetap unggul dibanding *Ward*. Ini menunjukkan bahwa pada jumlah klaster yang lebih banyak, pendekatan *linkage* selain *Ward* cenderung memberikan struktur klaster yang lebih terdefinisi.

Tabel 12 Hierarchical Clustering 4 Cluster

Eksperimen	Silhouette	Davies-Bouldin Index	Waktu Komputasi (detik)
Ward	0.8127	0.5247	0.05
Complete	0.8752	0.4142	0.08
Average	0.8640	0.4652	0.03
Single	0.9276	0.0410	0.03

Ketika eksperimen mencapai lima hingga tujuh klaster yang tercantum pada Tabel 13, Tabel 14, dan Tabel 15, performa metode *Ward* cenderung semakin menurun, ditandai dengan merosotnya *Silhouette Score* ke angka 0.7839, 0.6296, dan 0.6295 berturut-turut. Sebaliknya, *Complete* dan *Average Linkage* secara konsisten mencatatkan skor yang tinggi, dengan nilai *Silhouette* tetap di atas 0.84 untuk klaster lima hingga tujuh. *Single Linkage* tetap unggul dari sisi *Davies-Bouldin Index*, yang berarti klaster yang dihasilkan cenderung lebih terpisah dan padat. Akan tetapi, waktu prosesnya sedikit lebih tinggi, terutama saat mencapai tujuh klaster (0.11 detik).

Tabel 13 Hierarchical Clustering 5 Cluster

Eksperimen	Silhouette	Davies-Bouldin Index	Waktu Komputasi (detik)
Ward	0.7839	0.6634	0.06
Complete	0.8703	0.4198	0.10
Average	0.8610	0.3726	0.05
Single	0.8532	0.0615	0.11

Tabel 14 Hierarchical Clustering 6 Cluster

Eksperimen	Silhouette	Davies-Bouldin Index	Waktu Komputasi (detik)
Ward	0.6296	0.8358	0.18
Complete	0.8675	0.3397	0.16
Average	0.8553	0.3988	0.11
Single	0.7887	0.0863	0.10

Tabel 15 Hierarchical Clustering 7 Cluster

Eksperimen	Silhouette	Davies-Bouldin Index	Waktu Komputasi (detik)
Ward	0.6295	0.7696	0.04
Complete	0.8477	0.4683	0.03
Average	0.8381	0.4705	0.03
Single	0.7894	0.0849	0.04

Pada eksperimen dengan delapan dan sembilan kluster yang terlihat pada Tabel 16 dan Tabel 17, performa metode mulai menurun secara bertahap. *Ward Linkage* mengalami penurunan *Silhouette* hingga 0.6268 dan DBI 0.6659 pada kluster delapan. Sementara *Complete* dan *Average* masih mencatatkan nilai di atas 0.77 dan 0.82 dengan DBI lebih rendah dari *Ward*. Hal menarik adalah bahwa *Average Linkage* memberikan keseimbangan performa terbaik di beberapa konfigurasi, dengan nilai *Silhouette* yang konsisten tinggi dan DBI yang tidak ekstrem, meskipun waktu prosesnya sedikit lebih tinggi pada beberapa konfigurasi seperti kluster delapan dan sembilan.

Tabel 16 Hierarchical Clustering 8 Cluster

Eksperimen	Silhouette	Davies-Bouldin Index	Waktu Komputasi (detik)
Ward	0.6268	0.6659	0.05
Complete	0.7735	0.5585	0.04
Average	0.8204	0.4680	0.12
Single	0.7956	0.0776	0.08

Tabel 17 Hierarchical Clustering 9 Cluster

Eksperimen	Silhouette	Davies-Bouldin Index	Waktu Komputasi (detik)
Ward	0.6291	0.6095	0.12
Complete	0.7770	0.5331	0.14
Average	0.8169	0.4588	0.20
Single	0.8099	0.3030	0.15

Eksperimen dengan sepuluh kluster yang tercantum pada Tabel 18, menegaskan tren yang telah diamati sebelumnya, yaitu kinerja *Hierarchical Clustering* secara umum menurun seiring bertambahnya jumlah kluster. *Silhouette Score* dari *Ward Linkage* hanya sebesar 0.6327,

sementara *Single Linkage* yang sebelumnya unggul mulai mengalami penurunan performa dengan *Silhouette* 0.8069 dan DBI 0.2325. Waktu komputasi tetap efisien, dengan nilai maksimum sekitar 0.12 detik.

Tabel 18 Hierarchical Clustering 10 Cluster

Eksperimen	Silhouette	Davies-Bouldin Index	Waktu Komputasi (detik)
Ward	0.6327	0.6235	0.09
Complete	0.7769	0.4210	0.12
Average	0.8163	0.4689	0.12
Single	0.8069	0.2325	0.10

Secara keseluruhan, *Hierarchical Clustering* dengan dua hingga tiga kluster memberikan hasil terbaik, terutama ketika menggunakan *Complete*, *Average*, atau *Single Linkage*. Ketiganya menunjukkan nilai *Silhouette* yang tinggi dan DBI yang rendah, yang menandakan kualitas kluster yang sangat baik. Bila dibandingkan dengan *Fuzzy C-Means*, metode *Hierarchical Clustering* unggul dalam hal struktur kluster yang lebih stabil, terutama pada konfigurasi kluster kecil. Namun, keunggulan *Fuzzy C-Means* terletak pada fleksibilitas dan kecepatan proses, terutama untuk data berukuran besar atau dengan struktur tumpang tindih.

4. Kesimpulan

Berdasarkan seluruh rangkaian pengujian yang dilakukan terhadap algoritma *Fuzzy C-Means* (FCM) dan *Hierarchical Clustering* (HC) untuk klusterisasi data produksi ubi jalar di Indonesia, diperoleh beberapa poin kesimpulan sebagai berikut:

1. *Hierarchical Clustering* menunjukkan performa terbaik secara keseluruhan, terutama pada konfigurasi dua kluster dengan metode *linkage Ward*, *Complete*, dan *Average*, yang semuanya menghasilkan nilai *Silhouette* sebesar 0.9385 dan *Davies-Bouldin Index* (DBI) sebesar 0.3878. Sementara itu, metode *Single Linkage* menghasilkan DBI paling rendah sebesar 0.0543, walaupun nilai *Silhouette*-nya sedikit lebih kecil yaitu 0.9206. Ini menandakan bahwa struktur kluster yang dibentuk sangat padat dan terpisah dengan baik.
2. *Fuzzy C-Means* bekerja optimal pada dua kluster, dengan nilai *Silhouette* rata-rata sebesar 0.9074, DBI sebesar 0.6446, dan waktu komputasi tercepat yaitu 0.032 detik. Namun, ketika jumlah kluster ditingkatkan menjadi tiga, performa FCM mulai menurun, dengan *Silhouette* menurun ke 0.8119 dan DBI meningkat menjadi 0.7085.
3. Kinerja FCM terus menurun signifikan pada jumlah kluster lebih besar, seperti pada konfigurasi 6 kluster (*Silhouette* = 0.5767, DBI = 1.0959) dan 9 kluster (*Silhouette* = 0.3904, DBI = 1.3526). Pada konfigurasi 10 kluster, *Silhouette* bahkan hanya sebesar 0.3783 dan DBI sebesar 1.1022, yang

menunjukkan pembentukan kluster yang semakin tumpang tindih dan tidak kompak.

4. Sebaliknya, *Hierarchical Clustering* menunjukkan kestabilan performa yang tinggi, bahkan pada jumlah kluster yang lebih besar. Sebagai contoh, pada konfigurasi 6 kluster, metode *Complete* menghasilkan *Silhouette* sebesar 0.8676 dan DBI sebesar 0.3398, jauh lebih baik daripada nilai pada FCM dengan jumlah kluster yang sama.
5. Dari sisi efisiensi waktu, FCM memiliki waktu komputasi yang cepat pada jumlah kluster rendah, seperti 0.03 detik pada 2 dan 3 kluster. Namun, pada konfigurasi 10 kluster, waktu ini meningkat menjadi 0.166 detik. Sementara itu, *Hierarchical Clustering* mempertahankan waktu proses yang relatif stabil, seperti 0.03–0.12 detik pada hampir semua konfigurasi.

Kemungkinan pengembangan selanjutnya dari penelitian ini mencakup penerapan metode klusterisasi lain yang mampu menangani distribusi data yang tidak beraturan. Selain itu, pemanfaatan fitur tambahan seperti kondisi iklim, jenis tanah, atau penggunaan varietas ubi jalar di tiap wilayah dapat memperkaya dimensi analisis dan menghasilkan kluster yang lebih bermakna. Terakhir, pengujian algoritma klusterisasi dalam skala waktu yang lebih sempit, misalnya per musim tanam, dapat memberikan wawasan terhadap perubahan pola produksi dan dinamika pertanian secara periodik.

REFERENSI

- [1] R. Aisy *et al.*, “Pemanfaatan Ubi Jalar sebagai Alternatif Karbohidrat yang Meningkatkan Ekonomi Warga Banten,” *SEMAR (Jurnal Ilmu Pengetahuan, Teknologi, dan Seni bagi Masyarakat)*, vol. 12, no. 1, p. 47, Jun. 2023, doi: 10.20961/semar.v12i1.62162.
- [2] Muhammad Irsan Idrus, Trisnawaty AR, Muhanniah, and Mansur, “Pertumbuhan Dan Produksi Ubi Jalar (*Ipomoea batatas* L.) Varietas Lokal Soppeng Pada Perlakuan Sumber Benih Dan Ketinggian Bedengan,” *Jurnal Penelitian Pertanian Terapan*, vol. 2, no. 1, 2024.
- [3] M. Rojun and N. Nadziroh, “PERAN SEKTOR PERTANIAN DALAM PERTUMBUHAN EKONOMI DI KABUPATEN MAGETAN THE ROLE OF THE AGRICULTURAL SECTOR IN ECONOMIC GROWTH IN MAGETAN DISTRIC,” 2020.
- [4] D. Nurmin, M. N. Hayati, and R. Goejantoro, “Application of the Fuzzy C-Means Method in the Grouping of Regencies/Cities in Kalimantan Island Based on People’s Welfare Indicators in 2020,” *EKSPONENSIAL*, vol. 13, no. 2, Nov. 2022.
- [5] A. Fikri, B. F. Hutabarat, and U. Khaira, “Komparasi Antara Metode K-Means Clustering Dan Complete Linkage Dalam Pengelompokan Penyaluran Pinjaman Oleh Financial Technology,” *Jurnal Ilmiah Media Sisfo*, vol. 17, no. 2, pp. 228–239, Oct. 2023, doi: 10.33998/mediasisfo.2023.17.2.1373.
- [6] Y. Hasan, “Pengukuran Silhouette Score dan Davies-Bouldin Index pada Hasil Cluster K-Means dan DbSCAN,” 2024.
- [7] M. A. Septianto, A. Faqih, and A. R. Rinaldi, “KLAUSTERISASI DATA PRODUKSI PERTANIAN DI KABUPATEN CIREBON DENGAN ALGORITMA K-MEANS,” *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 13, no. 2, Apr. 2025, doi: 10.23960/jitet.v13i2.6174.
- [8] G. Marín Díaz, R. Gómez Medina, and J. A. Aijón Jiménez, “Integrating Fuzzy C-Means Clustering and Explainable AI for Robust Galaxy Classification,” *Mathematics*, vol. 12, no. 18, p. 2797, Sep. 2024, doi: 10.3390/math12182797.
- [9] R. Heppy Ria Sibarani, D. Lestari, D. Ananda, and I. Paul Hamonangan Marbun, “ANALISIS PENGELOMPOKAN KEBIASAAN PENGGUNAAN MEDIA SOSIAL MENGGUNAKAN METODE HIERARCHICAL CLUSTERING : PENDEKATAN DIVISIVE,” vol. 7, no. 10, 2024.
- [10] F. Alghifari and D. Juardi, “Penerapan Data Mining Pada Penjualan Makanan Dan Minuman Menggunakan Metode Algoritma Naïve Bayes,” 2021.
- [11] I. Gede, I. Sudipa, and M. Darmawiguna, *BUKU AJAR DATA MINING*. 2024. [Online]. Available: <https://www.researchgate.net/publication/377415198>
- [12] V. Gupta, J. Broughton, A. Rukundo, and L. J. Pinky, “Learning unbiased risk prediction based algorithms in healthcare: A case study with primary care patients,” *Inform Med Unlocked*, vol. 54, Jan. 2025, doi: 10.1016/j.imu.2025.101627.
- [13] U. Sri Sulistyawati, “Decoding Big Data: Mengubah Data Menjadi Keunggulan Kompetitif dalam Pengambilan Keputusan Bisnis,” *Jurnal Manajemen dan Teknologi (JMT)*, vol. 1, no. 2, 2024, doi: 10.35870/jmt.vxix.1114.
- [14] K. Maharana, S. Mondal, and B. Nemade, “A review: Data pre-processing and data augmentation techniques,” *Global Transitions Proceedings*, vol. 3, no. 1, pp. 91–99, Jun. 2022, doi: 10.1016/j.gltp.2022.04.020.
- [15] M. Farhan Mahfuzh, “IMPLEMENTASI DEEP LEARNING DALAM MENDETEKSI MATA UANG RUPIAH EMISI 2022 UNTUK MEMBANTU DISABILITAS NETRA DENGAN GOOGLE TEXT TO SPEECH SEBAGAI OUTPUT SUARA,” *Repositori Mahasiswa Untidar*, 2023.
- [16] V. N. Dang *et al.*, “Fairness and bias correction in machine learning for depression prediction across four study populations,” *Sci Rep*, vol. 14, no. 1, Dec. 2024, doi: 10.1038/s41598-024-58427-7.

- [17] M. Khaerul Rafli and U. Budiyanto, "KLAUSTERISASI TINGKAT KEMISKINAN MENGGUNAKAN METODE K-MEANS DI DKI JAKARTA," 2024.
- [18] S. W. Nur Aulia and P. K. Intan, "Klasterisasi Produksi Tanaman Perkebunan di Provinsi Jawa Timur Menggunakan Algoritma Fuzzy C-Means," *Jurnal Sains Matematika dan Statistika*, vol. 9, no. 2, p. 119, Aug. 2023, doi: 10.24014/jsms.v9i2.22735.
- [19] N. Rakhmawaty, Y. Novia Nasution, F. Deny, and T. Amijaya, "Perbandingan Metode K-Means Dan Metode Fuzzy C-Means (FCM) Pada Analisis Kinerja Pegawai PT. Cemara Khatulistiwa Persada Bontang," vol. 13, no. 1, May 2022.
- [20] G. S. Nugraha, R. Dwiyanaputra, F. Bimantoro, and A. Aranta, "Implementasi Fuzzy C-Means untuk Pengelompokan Daerah berdasarkan Persebaran Penularan Covid-19," *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 10, no. 1, pp. 97–104, Feb. 2023, doi: 10.25126/jtiik.2023105796.
- [21] I. M. Nur, A. N. L. Syifa, M. Kharis, and S. Heidy Permatasari, "IMPLEMENTASI METODE FUZZY C-MEANS DALAM PENGELOMPOKKAN HASIL PANEN PADI DI PROVINSI BALI," *VARIANCE: Journal of Statistics and Its Applications*, vol. 5, no. 1, pp. 13–24, Sep. 2023, doi: 10.30598/variancevol5iss1page13-24.
- [22] A. Alif *et al.*, "Penerapan Algoritma K-means Clustering dan Hierarchical Clustering dalam Mengelompokkan Data Pengangguran di Karawang," *Jurnal Algoritma*, 2024, doi: 10.33364/algoritma/v.21-2.2155.
- [23] B. M. Kumarahadi, H. Pratiwi, and S. Subanti, "Penerapan Metode Hierarchical Clustering Untuk Pengelompokan Kota/Kabupaten di Indonesia Berdasarkan Indikator Kemiskinan," *Jurnal Teknologi Informasi dan Komunikasi (TIKomSiN)*, vol. 11, no. 2, p. 13, Oct. 2023, doi: 10.30646/tikomsin.v11i2.754.
- [24] Z. Alamtaha, I. Djakaria, N. I. Yahya, J. Matematika, and F. Mipa, "Implementasi Algoritma Hierarchical Clustering dan Non-Hierarchical Clustering untuk Pengelompokkan Pengguna Media Sosial," *Estimasi: Journal of Statistics and Its Application*, vol. 4, no. 1, pp. 2721–379, 2023, doi: 10.20956/ejsa.vi.24830.
- [25] N. Nugroho and F. D. Adhinata, "Penggunaan Metode K-Means dan K-Means++ Sebagai Clustering Data Covid-19 di Pulau Jawa," *Teknika*, vol. 11, no. 3, pp. 170–179, Oct. 2022, doi: 10.34148/teknika.v11i3.502.
- [26] W. Warisa and N. Nurahman, "Perbandingan Performa Cluster Model Algoritma K-Means Dalam Mengelompokkan Penerima Bantuan Program Keluarga Harapan," *J. Sistem Info.*

Bisnis, vol. 13, no. 1, pp. 20–28, Jun. 2023, doi: 10.21456/vol13iss1pp20-28.

Matthew Russel Paul, Saat ini sebagai mahasiswa program studi Teknik Informatika Universitas Tarumanagara.