JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)
Journal homepage: https://jurnal.stkippgritulungagung.ac.id/index.php/jipi
ISSN: 2540-8984
Vol. 11, No. 1, March 2026, Pp.77-91

OPTIMASI PUSAT CLUSTER K-PROTOTYPES PADA PENGELOMPOKAN PENERIMAAN BANTUAN REHABILITASI RUTILAHU DI KOTA SURABAYA

Lisya Septyo Ningrum[1)], Kartika Maulida Hindrayani[*2)], Wahyu Syaifullah Jauharis Saputra[3)]
1. Sains Data, Fakultas Ilmu Komputer, Universitas Pembangunan Nasional Veteran Jawa Timur, Indonesia
2. Sains Data, Fakultas Ilmu Komputer, Universitas Pembangunan Nasional Veteran Jawa Timur, Indonesia
3. Sains Data, Fakultas Ilmu Komputer, Universitas Pembangunan Nasional Veteran Jawa Timur, Indonesia

Article Info
Kata Kunci: Flower Pollination Algorithm; Genetic Algorithm; K-Prototype Clustering; Particle Swarm Optimization; Rehabilitasi Rutilahu

Keywords: Flower Pollination Algorithm; Genetic Algorithm; K-Prototype Clustering; Particle Swarm Optimization; Rutilahu Rehabilitation;

Article history:
Received 15 April 2025
Revised 2 May 2025
Accepted 24 May 2025
Available online 1 March 2026

DOI :
https://doi.org/10.29100/jipi.v11i1.7656

* Corresponding author.
Corresponding Author
E-mail address:
kartika.maulida.ds@upnjatim.ac.id

ABSTRAK
Clustering merupakan teknik penting dalam data mining yang digunakan untuk mengelompokkan data berdasarkan kesamaan karakteristik. Algoritma *K-Prototypes* sering digunakan pada data bertipe campuran karena mengombinasikan *K-Means* untuk atribut numerik dan *K-Modes* untuk atribut kategorikal. Namun, kinerjanya sangat bergantung pada inisialisasi pusat klaster awal. Penelitian ini mengusulkan penerapan tiga algoritma optimasi yaitu *Particle Swarm Optimization* (PSO), *Genetic Algorithm* (GA), dan *Flower Pollination Algorithm* (FPA) untuk meningkatkan performa *K-Prototypes* dalam pengelompokan calon penerima program rehabilitasi Rumah Tidak Layak Huni (Rutilahu) di Kota Surabaya. Evaluasi dilakukan menggunakan *Davies-Bouldin Index* (DBI), *Silhouette Score*, dan waktu komputasi. Berdasarkan hasil penelitian menunjukkan bahwa PSO memberikan hasil terbaik dengan DBI terendah sebesar 0,6467, *Silhouette Score* tertinggi sebesar 0,5498, dan rata-rata waktu komputasi tercepat yaitu 18,0264 detik. GA menghasilkan DBI tertinggi sebesar 0,7134, *Silhouette Score* sebesar 0,5143, serta rata-rata waktu komputasi terlama yaitu 6875,6763 detik. FPA memiliki DBI 0,6467 dan *Silhouette Score* yang sama dengan PSO, tetapi dengan rata-rata waktu komputasi sebesar 3414,2101 detik. Dengan demikian, PSO terbukti paling efektif dalam meningkatkan akurasi dan efisiensi *clustering K-Prototypes*, serta mendukung distribusi bantuan yang lebih adil dan tepat sasaran.

ABSTRACT
Clustering is a crucial technique in data mining used to group data based on similar characteristics. The *K-Prototypes* algorithm is often applied to mixed-type data because it combines *K-Means* for numerical attributes and *K-Modes* for categorical attributes. However, its performance heavily depends on the initialization of cluster centers. This study proposes the application of three optimization algorithms, namely *Particle Swarm Optimization* (PSO), *Genetic Algorithm* (GA), and *Flower Pollination Algorithm* (FPA), to enhance the performance of *K-Prototypes* in clustering prospective beneficiaries of the Rehabilitation Program for Uninhabitable Houses (Rutilahu) in Surabaya City. The evaluation was conducted using the *Davies-Bouldin Index* (DBI), *Silhouette Score*, and computational time. The results show that PSO achieved the best performance, with the lowest DBI of 0.6467, the highest *Silhouette Score* of 0.5498, and the fastest average computational time of 18.0264 seconds. GA yielded the highest DBI of 0.7134, a *Silhouette Score* of 0.5143, and the longest average computational time of 6875.6763 seconds. FPA achieved a DBI of 0.6467 and the same *Silhouette Score* as PSO but with an average computational time of 3414.2101 seconds. Therefore, PSO is proven to be the most effective method in improving the accuracy and efficiency of *K-Prototypes clustering*, thereby supporting a fairer and more targeted distribution of aid.

JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)
Journal homepage: https://jurnal.stkippgritulungagung.ac.id/index.php/jipi
ISSN: 2540-8984
Vol. 11, No. 1, March 2026, Pp.77-91

I. PENDAHULUAN

CLUSTERING merupakan metode yang digunakan untuk mengelompokkan data berdasarkan kesamaan karakteristiknya [1]. Salah satu keunggulan utama pengelompokan data adalah kemampuannya mengenali pola secara otomatis tanpa memerlukan informasi sebelumnya [2]. Algoritma *K-Prototypes* adalah salah satu metode *clustering* yang dirancang untuk menangani data campuran, yaitu data numerikal dan kategorikal. Metode ini menggabungkan pendekatan algoritma *K-Means* yang biasa digunakan untuk menangani data numerik dan *K-Modes* yang biasa digunakan untuk menangani data kategorik [3]. Namun, performa algoritma *K-Prototypes* sangat bergantung pada inisialisasi awal pusat klaster. Jika inisialisasi kurang optimal, hasil *clustering* bisa tidak maksimal [4]. Oleh karena itu, diperlukan metode optimasi untuk menghasilkan pusat klaster yang optimal, sehingga dapat meningkatkan akurasi dan efektivitas algoritma ini.

Optimasi pada *clustering* bertujuan mengoptimalkan proses inisialisasi pusat klaster untuk meningkatkan kualitas hasil pengelompokan. Pada penelitian ini, tiga algoritma optimasi dipertimbangkan untuk mengatasi kelemahan *K-Prototypes*, di antaranya adalah *Particle Swarm Optimization* (PSO), *Genetic Algorithm* (GA), dan *Flower Pollination Algorithm* (FPA). PSO adalah algoritma optimasi yang terinspirasi oleh perilaku sosial burung dalam mencari makanan. Algoritma ini dapat digunakan untuk mengoptimalkan posisi pusat klaster dalam algoritma *clustering*, sehingga menghasilkan pengelompokan yang lebih akurat [5]. Sementara itu, GA merupakan metode optimasi yang didasarkan pada prinsip seleksi alam, di mana individu-individu dalam populasi akan berevolusi untuk menghasilkan solusi yang lebih baik [6]. Sedangkan, FPA mengambil inspirasi dari proses penyerbukan bunga, di mana pola penyerbukan yang efisien dapat digunakan untuk menemukan solusi optimal dalam pengelompokan data [7].

Penelitian ini berfokus pada tiga algoritma optimasi utama, yaitu PSO, GA, dan FPA dengan mempertimbangkan rekam jejak penggunaannya dalam konteks pengelompokan data campuran. Terdapat studi-studi terdahulu yang secara eksplisit menerapkan GA dan FPA untuk mengoptimalkan pusat klaster pada algoritma *K-Prototypes*. Hal ini menunjukkan potensi keduanya dalam menangani data dengan atribut numerik dan kategorikal secara bersamaan. Di sisi lain, PSO menunjukkan tingkat adopsi yang cukup tinggi dalam berbagai penelitian terkait segmentasi data sosial, termasuk pengelompokan calon penerima bantuan, meskipun belum banyak diintegrasikan secara langsung dengan *K-Prototypes*. Oleh sebab itu, eksplorasi lebih lanjut terhadap ketiga algoritma ini dalam skenario yang seragam dan berbasis data sosial campuran menjadi relevan untuk dilakukan, guna memperoleh pemahaman yang lebih mendalam mengenai karakteristik dan efektivitas masing-masing pendekatan dalam meningkatkan kualitas hasil klasterisasi.

Penelitian sebelumnya menunjukkan bahwa optimasi pusat klaster dengan algoritma seperti PSO, GA, dan FPA dapat meningkatkan kualitas *clustering*. Wijaya, Noersasongko dan Purwanto (2024) melakukan optimasi pusat klaster PSO. Pada penelitian ini, optimasi dengan PSO dapat meningkatkan kualitas *clustering*, terbukti dari nilai *Davies-Bouldin Index* (DBI) yang lebih rendah. Hasil penelitian menunjukkan bahwa DBI terbaik untuk *K-Medoids* adalah 0,283 dengan 3 klaster, sedangkan *PSO+K-Medoids* mencapai DBI 0,088 dengan 2 klaster, menandakan peningkatan kualitas pengelompokan [8]. Suwirmayati, Dharma Putra, dan Kumara (2020) melakukan optimasi pada pusat klaster *K-Prototype* menggunakan GA dalam menangani data *German Credit* yang bertipe campuran. Hasil penelitian mereka menunjukkan bahwa *K-Prototype* dengan optimasi GA memberikan hasil lebih baik dibandingkan *K-Prototype* konvensional dan *K-Means*, dengan nilai *cost* terendah sebesar 2622 dibandingkan 2706 untuk *K-Prototype* konvensional dan 9579 untuk *K-Means* [4]. Sementara itu, penelitian yang dilakukan oleh S. Karim, Panigoro, dan Nashar (2024) memanfaatkan FPA untuk mengoptimasi *K-Prototypes* pada dataset diabetes melitus tipe 2 di Kota Gorontalo, dengan hasil klaster terbaik sebanyak tiga klaster dan nilai *fitness* sebesar -0,501 [9].

Meskipun beberapa penelitian sebelumnya menunjukkan peningkatan kualitas *clustering* berdasarkan nilai *Davies-Bouldin Index* (DBI) atau *fitness*, perlu dikaji lebih lanjut alasan pemilihan metrik-metrik tersebut dalam konteks data campuran seperti data Rutilahu. DBI memang umum digunakan karena mengukur seberapa baik masing-masing klaster terpisah dan kompak [10]. Namun, dalam penelitian ini juga dipertimbangkan metrik evaluasi tambahan berupa *Silhouette Score*, yang tidak hanya mengukur kohesi dan separasi antar klaster, tetapi juga memberikan nilai evaluasi per data point serta memungkinkan visualisasi distribusi klaster [11]. Hal ini mendukung analisis yang lebih komprehensif dan interpretatif, terutama dalam konteks data sosial yang membutuhkan pemeriksaan granular terhadap hasil klasterisasi.

Dari beberapa penelitian terdahulu, menunjukkan efektivitas algoritma optimasi seperti PSO, GA, dan FPA dalam meningkatkan kualitas klasterisasi. Namun, penerapannya masih banyak difokuskan pada skenario dengan kompleksitas data yang terbatas. Dalam konteks sistem berskala besar, diperlukan pendekatan yang tidak hanya menghasilkan klasterisasi yang akurat, tetapi juga efisien secara komputasi dan adaptif terhadap keterbatasan

JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)
Journal homepage: https://jurnal.stkippgritulungagung.ac.id/index.php/jipi
ISSN: 2540-8984
Vol. 11, No. 1, March 2026, Pp.77-91

sumber daya. Penelitian ini mengusulkan strategi optimasi yang lebih efisien dan fleksibel terhadap variasi struktur data, guna mendukung penerapan yang lebih luas dalam lingkungan operasional yang kompleks.

Salah satu keunggulan utama *clustering* adalah kemampuannya untuk menangani berbagai jenis data, baik numerikal maupun kategorikal, sehingga sering dimanfaatkan dalam berbagai bidang seperti kesehatan, pemasaran, dan perencanaan sosial. Salah satunya adalah program Rehabilitasi Rutilahu. Program Rehabilitasi Rumah Tidak Layak Huni (Rutilahu) merupakan masalah sosial di Kota yang sering kali menghadapi kendala. Program ini merupakan salah satu upaya strategis Pemerintah Kota Surabaya untuk meningkatkan kualitas hidup masyarakat miskin melalui perbaikan tempat tinggal yang tidak memenuhi standar kelayakan. Berdasarkan data dari Dinas Perumahan Rakyat dan Kawasan Permukiman serta Pertanahan (DPRKPP) Kota Surabaya, banyak keluarga di Surabaya yang berhadapan dengan rumah di bawah standar dan memerlukan perbaikan besar. Terdapat sekitar 2,51 juta unit rumah tidak layak huni, dengan 2,18 juta rumah berisiko menjadi tidak layak huni dan 330.000 rumah sudah dianggap benar-benar tidak layak huni. Program rehabilitasi ini bertujuan untuk mengatasi permasalahan tersebut dan mendapat antusiasme tinggi dari masyarakat [12].

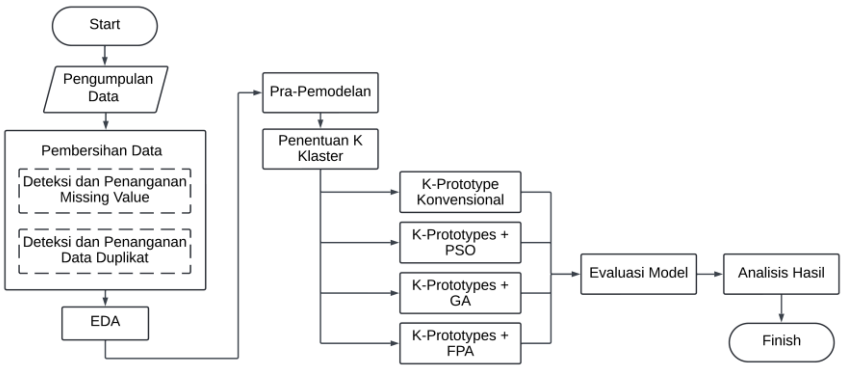
Proses penentuan penerima bantuan rehabilitasi Rutilahu masih menghadapi tantangan. Salah satu masalah utama adalah memastikan bantuan diberikan kepada pihak yang benar-benar membutuhkan. Subjektivitas dalam penilaian dan kriteria seleksi yang belum terstandarisasi sering menyebabkan bantuan tidak tepat sasaran [13]. Kondisi ini menimbulkan potensi kurangnya efisiensi, rumah tangga yang layak menerima bantuan terlewatkan, sementara pihak yang kurang prioritas justru mendapatkan alokasi. Oleh sebab itu, diperlukan sistem otomatisasi berbasis data untuk mengelompokkan masyarakat yang berhak menerima bantuan rehabilitasi Rutilahu.

Permasalahan rumah tidak layak huni merupakan isu sosial yang tidak hanya terbatas pada konteks lokal, tetapi juga menjadi perhatian global dalam upaya pencapaian *Sustainable Development Goals* (SDGs) yang menekankan pada kota dan permukiman yang inklusif, aman, dan berkelanjutan [14]. Dataset Rutilahu yang digunakan dalam penelitian ini mencerminkan upaya pemerintah dalam menanggulangi ketimpangan sosial melalui program perbaikan rumah. Secara teknis, dataset ini memiliki kombinasi atribut numerik dan kategorikal ordinal, menjadikannya tantangan tersendiri dalam proses klasterisasi. Kompleksitas ini memberikan ruang eksplorasi yang relevan untuk pengujian algoritma optimasi dalam konteks *clustering* data campuran.

Penelitian ini mengeksplorasi dan mengevaluasi performa ketiga algoritma optimasi yaitu PSO, GA, dan FPA dalam mengoptimalkan pusat klaster *K-Prototypes*. Berbeda dengan studi sebelumnya yang umumnya hanya berfokus pada satu pendekatan atau diterapkan pada data non-sosial. Penelitian ini menjadi salah satu yang membandingkan secara langsung ketiga algoritma tersebut dalam konteks pengelompokan penerima bantuan sosial Rutilahu, serta mengidentifikasi keunggulan dan keterbatasannya terhadap tantangan teknis seperti atribut campuran.

II. METODE PENELITIAN

Metode penelitian ini dirancang secara sistematis untuk mencapai tujuan penelitian. Tahapan penelitian pada Gambar 1 menjelaskan setiap langkah, mulai dari persiapan data hingga analisis hasil.



Gambar 1. Diagram Alir Penelitian

Adapun penjelasan dalam setiap tahapan yang ada pada diagram alir penelitian pada Gambar 1 sebagai berikut.

A. Pengumpulan dan Pembersihan Data

Tahap penelitian ini bertujuan untuk mengumpulkan variabel-variabel yang relevan [15]. Penelitian ini menggunakan data kependudukan dari warga Kota Surabaya yang mengajukan bantuan rehabilitasi Rutilahu tahun 2024. Data ini mencakup kondisi demografi, ekonomi, sosial, dan kesejahteraan. Setelah data terkumpul, data akan di bersihkan. Proses pembersihan data merupakan langkah krusial dalam memastikan kualitas data yang akan digunakan dalam analisis [16]. Pembersihan data dilakukan dengan deteksi dan penanganan *missing value*

menggunakan metode imputasi statistik serta penghapusan data duplikat guna menghindari bias dalam analisis.

B. Exploratory Data Analysis (EDA) dan Pra-Pemodelan

Tahap visualisasi data bertujuan untuk memahami struktur dataset, menemukan pola, mendeteksi anomali, memeriksa asumsi, dan memvisualisasikan data [17]. Analisis dilakukan melalui statistika deskriptif dan visualisasi sebaran data menggunakan grafik. Pada data numerik, dilakukan visualisasi *boxplot* untuk melihat keberadaan *outlier* pada data. *Outlier* adalah data yang menunjukkan perbedaan mencolok dibandingkan dengan data lainnya [18]. Selanjutnya, dilakukan proses standarisasi menggunakan *StandardScaler* untuk menyelaraskan skala fitur numerik dalam data agar berada pada rentang yang sama sehingga tidak menyebabkan bias dalam proses klasterisasi [19]. Pada data kategorik, dilakukan visualisasi menggunakan *barplot* untuk mengetahui frekuensi kategori dalam suatu variabel untuk memastikan tidak ada ketimpangan data yang dapat mempengaruhi hasil klasterisasi. Kemudian, dilakukan proses *encoding* untuk transformasi fitur kategorik menjadi representasi angka [20]. Proses *encoding* dilakukan menggunakan *OrdinalEncoder*. Hal ini dikarenakan data kategorik yang digunakan dalam penelitian bersifat ordinal, yaitu data yang memiliki urutan atau tingkatan, tetapi selisih antar tingkatannya tidak bisa diukur secara pasti.

C. K-Prototype Clustering

Pemodelan menggunakan *K-Prototype* dengan mengelompokkan data campuran, ke dalam k klaster berdasarkan jarak dissimilarity yang minimal [21]. Proses dimulai dengan inisialisasi prototipe secara acak yang terdiri atas kombinasi fitur numerik dan kategorik. Kemudian, prototipe di evaluasi dengan menghitung *dissimilarity* yang menggunakan kombinasi antara jarak numerik dan jarak kategorik. Rumus total *dissimilarity* diformulasikan pada persamaan (1) sebagai berikut.

$$d(x,y) = \sqrt{\sum_{i=1}^{p}(x_i - y_i)^2} + \gamma \cdot \sum_{i=1}^{q} \delta(x_i, y_i) \quad (1)$$

Keterangan:

x dan y	:	Dua vektor data yang memiliki n dimensi
x_i, y_i	:	Nilai fitur ke-i dari vektor x dan y
p	:	Jumlah fitur numerik dalam dataset
q	:	Jumlah fitur kategorik dalam dataset
γ	:	Parameter penyeimbang antara jarak numerik dan jarak kategorik
$\delta(x_i, y_i)$	:	Fungsi indikator, bernilai 1 jika $x_i \neq y_i$ dan bernilai 0 jika $x_i = y_i$

Setelah itu, setiap data di*assign* ke klaster dengan nilai *dissimilarity* terkecil. Prototipe diperbarui dengan menghitung rata-rata untuk fitur numerik dan modus untuk fitur kategorik. Proses ini berulang hingga prototipe stabil atau iterasi maksimum tercapai [22].

D. Particle Swarm Optimization (PSO)

Algoritma *Particle Swarm Optimization* (PSO) diperkenalkan oleh J. Kennedy dan R. C. Eberhart merupakan algoritma optimasi yang menyeimbangkan eksplorasi dan eksploitasi dalam pencarian solusi [23]. PSO diawali dengan inisialisasi populasi secara acak dalam ruang pencarian dengan posisi x_i dan kecepatan awal v_i. Setiap partikel kemudian dievaluasi menggunakan fungsi objektif untuk menentukan nilai *fitness*nya. Pada tahap pembaruan kecepatan dan posisi, setiap partikel mengadaptasi pergerakannya berdasarkan pengalaman individu ($pBest$) dan sosial ($gBest$). Kecepatan partikel diperbarui menggunakan persamaan (2) berikut.

$$v_i^{t+1} = \omega v_i^t + c_1 r_1 (pbest_i - x_i^t) + c_2 r_2 (gBest - x_i^t) \quad (2)$$

Keterangan:

v_i^{t+1}	:	Kecepatan partikel i pada iterasi ke-$(t+1)$
ω	:	Faktor inersia
v_i^t	:	Kecepatan partakel i pada iterasi sebelumnya
c_1 dan c_2	:	Koefisien pembelajaran
r_1 dan r_2	:	Bilangan acak dalam interval $[0, 1]$
$pbest_i$	:	Posisi terbaik yang pernah dicapai partikel i selama proses iterasi
$gBest$	:	Posisi terbaik yang pernah dicapai seluruh populasi partikel dalam iterasi yang telah berlangsung
x_i^t	:	Posisi partikel i pada iterasi ke-t

Setelah kecepatan diperbarui, posisi partikel diperbarui menggunakan persamaan (3) berikut.

$$x_i^{t+1} = x_i^t + v_i^{t+1} \quad (3)$$

Keterangan:
x_i^{t+1} : Posisi baru dari partikel i pada iterasi ke-$(t+1)$
x_i^t : Posisi partikel i pada iterasi sebelumnya
v_i^{t+1} : Kecepatan baru partikel yang telah diperbarui menggunakan persamaan sebelumnya

Proses iterasi ini berlanjut hingga kondisi penghentian terpenuhi, seperti jumlah iterasi maksimum atau konvergensi terhadap nilai optimal [8].

E. *Genetic Algorithm (GA)*

Genetic Algorithm (GA) adalah metode optimasi dan pencarian berbasis evolusi yang meniru proses seleksi alam, seperti yang dijelaskan dalam teori Darwin [24]. Algoritma ini dimulai dengan inisialisasi populasi individu (kromosom) yang dihasilkan secara acak dalam ruang pencarian. Setiap individu dievaluasi menggunakan fungsi *fitness* untuk mengukur kualitas solusinya. Seleksi individu dilakukan menggunakan *Tournament Selection*, di mana sekelompok individu dipilih secara acak, lalu individu dengan *fitness* terbaik dipilih sebagai induk untuk reproduksi. Pada tahap *crossover*, dua individu dikombinasikan menggunakan *One-Point Crossover* dengan memilih titik potong secara acak dalam kromosom dan menukar bagian setelah titik tersebut. Setelah itu, dilakukan mutasi untuk menjaga keberagaman populasi dengan mengubah sebagian kecil kromosom secara acak. Pada mutasi adaptif, probabilitas mutasi (P_m) menyesuaikan kondisi evolusi, dengan nilai yang lebih besar pada generasi awal dan semakin kecil seiring mendekati konvergensi. Ketika gen X_i mengalami mutasi, maka nilai barunya ditentukan berdasarkan probabilitas mutasi (P_m). Jika bilangan acak r dalam interval $[0,1] >$ dari P_m, maka nilai gen tetap. Sebaliknya, jika $r \leq P_m$, maka nilai gen diubah secara acak untuk meningkatkan keberagaman solusi dan mencegah konvergensi prematur. Individu baru dievaluasi kembali menggunakan fungsi *fitness*. Populasi kemudian diperbarui dengan individu terbaik untuk memasuki iterasi berikutnya. Proses ini berlanjut hingga kondisi penghentian tercapai, seperti jumlah generasi maksimum atau konvergensi terhadap solusi optimal [25].

F. *Flower Pollination Algorithm (FPA)*

Flower Pollination Algorithm (FPA) adalah salah algoritma optimasi metaheuristik yang dikembangkan oleh Xin-She Yang (2012), terinspirasi dari mekanisme penyerbukan bunga di alam [26]. Diawali dengan inisialisasi sejumlah solusi awal yang direpresentasikan sebagai bunga (x_i) secara acak dalam ruang pencarian. Setiap bunga memiliki nilai *fitness* yang mencerminkan kualitas solusinya. Proses penyerbukan dalam FPA terbagi menjadi dua jenis, yaitu penyerbukan silang global dan penyerbukan sendiri lokal. Pada penyerbukan silang global, serbuk sari dapat berpindah dalam jarak jauh dengan bantuan *Lévy flight* untuk eksplorasi optimal dalam ruang pencarian, yang diformulasikan pada persamaan (4) sebagai berikut.

$$x_i^{t+1} = x_i^t + \gamma \cdot L(\lambda) \cdot (gBest - x_i^t) \quad (4)$$

Keterangan:
x_i^{t+1} : Posisi baru bunga i pada iterasi ke-$(t+1)$
x_i^t : Posisi bunga saat ini pada saat iterasi ke-t
γ : Faktor skala yang mengontrol langkah pergerakan serbuk sari
$gBest$: Posisi bunga dengan nilai *fitness* terbaik dalam populasi.
$L(\lambda)$: Distribusi *Lévy flight*

Sedangkan pada penyerbukan sendiri lokal terjadi ketika serbuk sari berpindah dalam jarak pendek antara bunga yang berdekatan, menyerupai penyerbukan sendiri atau penyerbukan dalam kelompok kecil. Model ini dapat dinyatakan pada persamaan (5) sebagai berikut.

$$x_i^{t+1} = x_i^t + \epsilon \cdot (x_j^t - x_k^t) \quad (5)$$

Keterangan:
x_j^t dan x_k^t : Dua solusi acak dari populasi yang dipilih secara lokal
ϵ : Bilangan acak dalam interval $[0,1]$ yang mengontrol tingkat pergerakan dalam pencarian lokal

Algoritma ini menerapkan parameter probabilitas p yang mengatur kecenderungan antara penyerbukan silang global dan penyerbukan sendiri lokal. Nilai r adalah bilangan acak dalam interval $[0,1]$. Jika nilai $r < p$, maka dilakukan penyerbukan silang global. Sebaliknya, jika nilai $r \geq p$, maka diterapkan penyerbukan sendiri lokal. Selanjutnya, mekanisme seleksi memilih solusi terbaik berdasarkan nilai *fitness*. Proses ini berulang hingga mencapai batas iterasi maksimum atau konvergensi terhadap solusi terbaik [27].

G. Evaluasi Model

Pada proses penentuan jumlah klaster optimal dan evaluasi model menggunakan *Elbow Method* dan *Silhouette Score*. *Elbow Method* digunakan untuk menentukan jumlah klaster (k) optimal dengan melihat *Sum of Squared Errors (SSE)*, yaitu jumlah kuadrat jarak setiap titik dalam klaster terhadap pusat klaster. Proses ini dilakukan dengan menjalankan algoritma klasterisasi untuk berbagai nilai k, lalu memplot grafik antara k dan SSE. Titik "siku" (*elbow*) pada grafik menunjukkan nilai k optimal, di mana setelah titik ini, penurunan *SSE* mulai melambat secara signifikan, sehingga penambahan klaster tidak memberikan peningkatan signifikan dalam pemisahan data [28]. Setelah jumlah klaster ditentukan, kualitas klaster dievaluasi menggunakan *Davies-Bouldin Index* (DBI) dan *Silhouette Score*. *Davies-Bouldin Index* (DBI) adalah metrik evaluasi *clustering* yang mengukur rata-rata kesamaan antar klaster, berdasarkan rasio antara sebaran dalam klaster dan jarak antar klaster. Semakin rendah nilai DBI, semakin baik kualitas *clustering* yang dihasilkan [29]. *Silhouette Score* merupakan metrik yang mengukur sejauh mana setiap titik data berada dalam klaster yang sesuai dibandingkan dengan klaster lain. Skor ini dihitung berdasarkan perbedaan antara jarak rata-rata titik ke klaster yang sama dan jarak rata-rata titik ke klaster terdekat lainnya [30]. Nilai *Silhouette Score* berkisar antara -1 hingga 1. Berdasarkan nilai tersebut, semakin ke arah kiri, kualitas klaster yang terbentuk semakin buruk, sedangkan semakin ke arah kanan, kualitas klaster yang terbentuk semakin baik [31].

H. Analisis Hasil

Pada tahap ini dilakukan perbandingan model *clustering* untuk memahami sejauh mana model berhasil memisahkan data dengan karakteristik yang berbeda. Analisis ini membantu mengidentifikasi kecenderungan pola dalam data serta memberikan wawasan awal sebelum masuk ke tahap evaluasi model secara kuantitatif.

III. Hasil Dan Pembahasan

Adapun hasil penelitian yang diperoleh dari setiap tahapan *clustering* mulai dari pengumpulan data hingga analisis hasil sebagai berikut.

A. Pengumpulan dan Pembersihan Data

Penelitian ini menggunakan data pengajuan bantuan Rehabilitasi Rutilahu Kota Surabaya tahun 2024, yang terdiri dari 1.360 baris data. Adapun variabel yang digunakan meliputi Pendapatan, Jumlah Tanggungan, Luas Rumah, Kondisi Atap, Kondisi Dinding, dan Kondisi Lantai. Setelah dilakukan pengecekan untuk mendeteksi *missing value* dan data duplikat, dataset tidak terindikasi adanya *missing value* dan data duplikat. Tabel 1 berikut adalah sampel data bersih yang digunakan dalam penelitian.

Tabel I
Data Bersih

No.	Nama	Pendapatan	Jumlah Tanggungan	Luas Rumah (m^2)	Kondisi Atap	Kondisi Dinding	Kondisi Lantai
1	P_1	4400000	3	41	Baik	Baik	Sama Rata dengan Jalanan
2	P_2	2600000	3	30	Sedang	Baik	Sama Rata dengan Jalanan
3	P_3	1700000	5	16	Sedang	Buruk	Lebih Rendah dari Jalanan
4	P_4	1800000	7	19	Sedang	Buruk	Lebih Rendah dari Jalanan
…	…	…	…	…	…	…	…
1360	P_{1360}	2300000	7	15	Sedang	Buruk	Sama Rata dengan Jalanan

B. Exploratory Data Analysis (EDA) dan Pra-Pemodelan

Pada fitur numerik, dapat diketahui sebaran data melalui statistika deskriptif. Pada Tabel 2 berikut merupakan ringkasan statistika deskriptif dari Pendapatan, Jumlah Tanggungan, dan Luas Rumah.

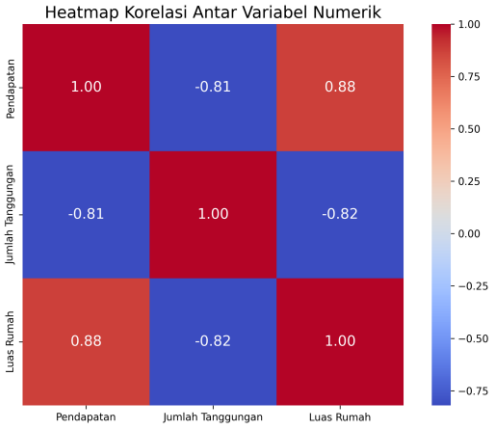
Tabel II
Statistika Deskriptif Fitur Numerik

Statistika Deskriptif	Pendapatan	Jumlah Tanggungan	Luas Rumah (m^2)
Minimum	1200000	2	15
Kuartil 1	2100000	3	21
Median	2700000	5	28
Mean	2807206	5	28,49
Kuartil 3	3500000	6	35
Maksimum	4500000	7	45

Berdasarkan Tabel II, dapat dilihat bahwa pendapatan memiliki rentang yang cukup luas, mulai dari 1.200.000 Rupiah hingga 4.500.000 Rupiah, dengan median 2.700.000 Rupiah dan rata-rata 2.807.206 Rupiah, menunjukkan kemungkinan *outlier* berpenghasilan tinggi. Jumlah tanggungan bervariasi dari 2 hingga 7 orang, dengan median

JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)
Journal homepage: https://jurnal.stkippgritulungagung.ac.id/index.php/jipi
ISSN: 2540-8984
Vol. 11, No. 1, March 2026, Pp.77-91

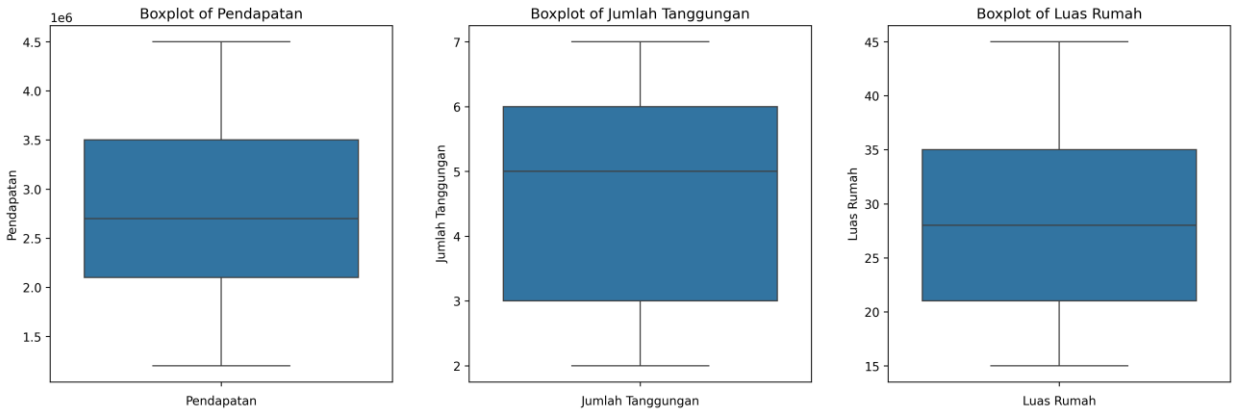
5 dan mayoritas berada dalam rentang 3–6 orang. Luas rumah berkisar antara 15 m² hingga 45 m², dengan median 28 m² dan rata-rata 28,49 m², menunjukkan distribusi yang relatif seragam meskipun ada beberapa rumah lebih luas. Untuk memperoleh insight yang lebih mendalam mengenai hubungan antar variabel numerik, berikut adalah visualisasi *heatmap* pada Gambar 2 yang menggambarkan korelasi antar fitur tersebut.



Gambar 2. Visualisasi Heatmap Fitur Numerik

Berdasarkan visualisais heatmap pada Gambar 2, hasil analisis korelasi antar variabel numerik menunjukkan hubungan yang signifikan antara pendapatan, jumlah tanggungan, dan luas rumah. Terdapat korelasi positif kuat antara pendapatan dan luas rumah, yaittu r sebesar 0,878, yang mengindikasikan bahwa semakin tinggi pendapatan, maka cenderung memiliki rumah dengan luas yang lebih besar. Hal ini konsisten dengan asumsi umum bahwa kondisi ekonomi keluarga berdampak langsung pada kemampuan dalam memiliki atau membangun rumah yang layak.

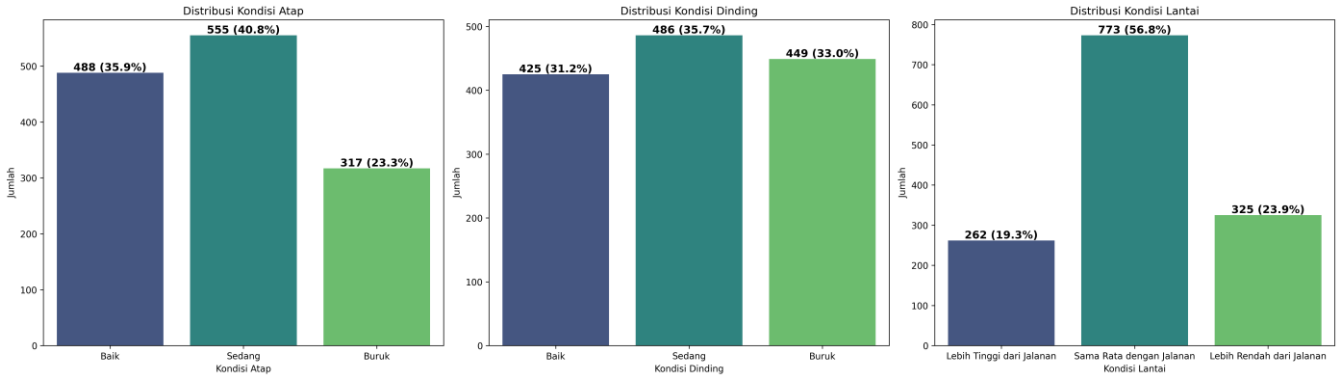
Sebaliknya, jumlah tanggungan memiliki korelasi negatif yang kuat terhadap pendapatan, yaitu r sebesar -0,814 dan luas rumah, yaitu r sebesar -0,820. Korelasi negatif ini menggambarkan bahwa semakin banyak jumlah tanggungan dalam sebuah rumah tangga, maka semakin kecil kemungkinan rumah tangga tersebut memiliki pendapatan yang tinggi dan luas rumah yang memadai. Temuan ini memperkuat hipotesis bahwa rumah tangga dengan jumlah anggota keluarga yang besar memiliki risiko lebih tinggi untuk berada dalam kondisi sosial ekonomi yang lebih rendah. Secara keseluruhan, hasil korelasi ini memberikan indikasi awal bahwa keluarga dengan pendapatan rendah, jumlah tanggungan tinggi, dan rumah sempit memiliki karakteristik yang saling terkait, dan oleh karena itu, dapat menjadi sasaran prioritas dalam program bantuan rehabilitasi rumah tidak layak huni.



Gambar 3. Boxplot Fitur Numerik

Berdasarkan visualisasi boxplot pada Gambar 3, tidak ditemukan keberadaan *outlier* pada ketiga variabel numerik, yaitu Pendapatan, Jumlah Tanggungan, dan Luas Rumah. Hal ini terlihat dari tidak adanya titik individu yang berada di luar batas *whisker* pada masing-masing plot. Oleh karena tidak terdapat *outlier* pada fitur numerik, maka fitur numerik di standarisasi.

JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)
Journal homepage: https://jurnal.stkippgritulungagung.ac.id/index.php/jipi
ISSN: 2540-8984
Vol. 11, No. 1, March 2026, Pp.77-91



Gambar 4. Barplot Fitur Kategorik

Berdasarkan visualisasi *barplot* pada Gambar 4, menunjukkan distribusi kondisi atap, dinding, dan lantai dalam dataset. Distribusi kondisi atap dalam dataset menunjukkan dominasi pada kategori Sedang, yang mencakup 535 data (40,8%), diikuti oleh kategori Baik dengan 488 data (35,9%) dan kategori Buruk dengan 317 data (23,3%). Hasil ini mengindikasikan bahwa sebagian besar rumah dalam dataset memiliki kondisi atap yang relatif baik, meskipun terdapat proporsi rumah dengan kondisi atap yang sedang atau buruk, yang membutuhkan perhatian dalam program rehabilitasi. Dominasi kategori Sedang menggambarkan adanya potensi kebutuhan perbaikan, meskipun tidak dalam kondisi darurat. Persentase distribusi antar kategori yang relatif merata menunjukkan adanya variasi kondisi atap yang cukup signifikan, yang menjadikannya fitur penting dalam pengelompokan dan analisis keputusan, seperti penggunaan algoritma *K-Prototypes*.

Pada Kondisi Dinding, distribusi kategori menunjukkan keseimbangan yang cukup baik, dengan kategori Sedang berjumlah 486 data (35,7%), kategori Baik sebanyak 449 data (31,2%), dan kategori Buruk sebanyak 425 data (33,0%). Variasi yang cukup signifikan pada kondisi dinding rumah mencerminkan keragaman tingkat kerusakan yang ada, di mana sebagian besar rumah membutuhkan perbaikan pada dindingnya. Penyebaran data yang hampir merata di antara ketiga kategori menunjukkan pentingnya fitur ini dalam pengelompokan penerima bantuan, khususnya dalam model *unsupervised* seperti *K-Prototypes*, yang memanfaatkan keragaman data untuk membedakan kelompok rumah dengan lebih efektif.

Adapun pada Kondisi Lantai, distribusi kategori menunjukkan dominasi pada kategori Sama Rata dengan Jalanan, yang mencakup 773 data (56,8%), diikuti oleh kategori Lebih Rendah dari Jalanan dengan 325 data (23,9%) dan kategori Lebih Tinggi dari Jalanan dengan 262 data (19,3%). Meskipun kategori Sama Rata dengan Jalanan lebih dominan, distribusi antar kategori tetap menunjukkan variasi yang signifikan. Kondisi lantai yang lebih rendah atau lebih tinggi dari jalanan dapat menjadi indikator penting terkait kerentanannya terhadap banjir atau genangan air. Oleh karena itu, fitur Kondisi Lantai sangat relevan untuk dipertimbangkan dalam konteks rehabilitasi rumah dan analisis prioritas penerima bantuan, mengingat kondisi lantai yang dapat mempengaruhi kelayakan rumah serta kerentanannya terhadap risiko banjir.

Secara keseluruhan, ketiga fitur tersebut memperlihatkan distribusi yang bervariasi, dengan beberapa kategori yang dominan namun tetap mencerminkan keragaman kondisi rumah. Variasi ini menunjukkan pentingnya mempertimbangkan Kondisi Atap, Kondisi Dinding, dan Kondisi Lantai dalam penentuan prioritas penerima bantuan rehabilitasi rumah, karena mereka memberikan gambaran yang lebih menyeluruh tentang kebutuhan perbaikan rumah yang bersangkutan.Selanjutnya, fitur kategorik dilakukan *encoding* menggunakan *OrdinalEncoder*. Adapun hasil dari proses standarisasi dan *encoding* pada pra-pemodelan pada Tabel 3 sebagai berikut.

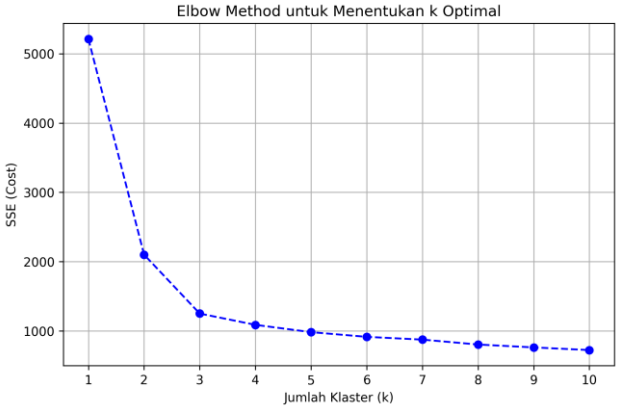
TABEL III
DATA HASIL PRA-PEMODELAN

No.	Pendapatan	Jumlah Tanggungan	Luas Rumah (m^2)	Kondisi Atap	Kondisi Dinding	Kondisi Lantai
1	1,942830	-0,940909	1,453806	2	2	1
2	-0,252742	-0,940909	0,175209	1	2	1
3	-1,350528	0,311181	-1,452097	1	0	0
4	-1,228552	1,563271	-1,103389	1	0	0
…	…	…	…	…	…	…
1360	-0,618670	1,563271	-1,568333	1	0	1

C. *Pemodelan K-Prototype Clustering*

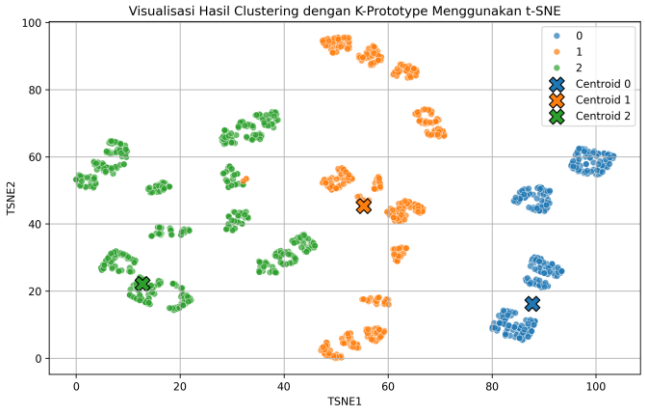
Setelah melakukan tahap pembersihan dan eksplorasi pada, tahap selanjutnya adalah pemodelan data. Terdapat 1360 baris data dan 6 variabel (3 variabel numerik telah di standarisasi dan 3 variabel kategorik telah di *OrdinalEncoding*) yang akan digunakan dalam tahap ini. Sebelum melakukan pemodelan, tahap yang paling krusial dalam *clustering* adalah menentukan jumlah *k* klaster.

JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)
Journal homepage: https://jurnal.stkippgritulungagung.ac.id/index.php/jipi
ISSN: 2540-8984
Vol. 11, No. 1, March 2026, Pp.77-91



Gambar 5. Grafik Elbow Method

Berdasarkan hasil *Elbow Method* yang ditampilkan pada grafik Gambar 5, nilai *SSE (Sum of Squared Errors)* mengalami penurunan tajam dari $k = 1$ ke $k = 2$, dan berlanjut menurun secara signifikan hingga $k = 3$. Setelah $k = 3$, penurunan *SSE* mulai melambat. Hal ini menunjukkan bahwa keuntungan dari menambahkan klaster baru semakin kecil. Titik elbow yang paling jelas terlihat berada di sekitar $k = 3$, karena setelah itu penurunan *SSE* tidak terlalu drastis. Kemudian, dilakukan pengelompokan menggunakan algoritma *K-Prototype*. Adapun hasil pengelompokan dapat dilihat melalui visualisasi menggunakan t-SNE (*t-distributed Stochastic Neighbor Embedding*) pada Gambar 6 berikut.



Gambar 6. Visualisasi PCA Pemodelan K-Prototype Konvensional

Hasil implementasi *K-Prototypes* konvensional menunjukkan bahwa dataset terbagi ke dalam tiga klaster dengan karakteristik yang berbeda. Klaster 0 dengan 307 anggota, memiliki nilai numerik yang tinggi dan didominasi oleh kategori Baik. Klaster 1 terdiri dari 475 anggota dengan nilai numerik yang mendekati rata-rata seluruh data serta dominasi kategori Sedang. Sementara itu, klaster 2 merupakan yang terbesar dengan 578 anggota, memiliki nilai numerik yang lebih rendah dan didominasi oleh kategori Buruk. Berdasarkan hasil visualisasi pada Gambar 6, terdapat beberapa data yang seharusnya masuk ke dalam klaster 2, tetapi justru terkelompok ke dalam klaster 1. Hal ini mengindikasikan kemungkinan *overlap* antar klaster atau kurang optimalnya pemisahan yang dilakukan oleh metode konvensional.

D. Particle Swarm Optimization (PSO)

Setiap partikel dalam PSO merepresentasikan sekumpulan pusat klaster yang diperbarui berdasarkan kecepatan dan posisi terbaiknya ($pBest$) serta posisi terbaik global ($gBest$). Pendekatan *hybrid* PSO diimplementasikan menggunakan parameter 1360 partikel, $k = 3$, 100 iterasi, $\omega = 0{,}6$, $c_1 = 2.0$, dan $c_2 = 2{,}0$. Parameter ini dipilih berdasarkan serangkaian uji coba untuk mencapai konvergensi optimal. Nilai parameter diuji dalam rentang tertentu untuk mengamati pengaruhnya terhadap hasil *clustering*. Kombinasi parameter ini membantu PSO menemukan pusat klaster yang lebih stabil dan akurat, sehingga meningkatkan kualitas pengelompokan penerima bantuan Rutilahu. Dalam proses pengujian stabilitas algoritma, dilakukan sebanyak lima kali *running* untuk memastikan konsistensi hasil pengelompokan. Adapun hasil dari lima kali pengujian tersebut disajikan pada Tabel 4 berikut.

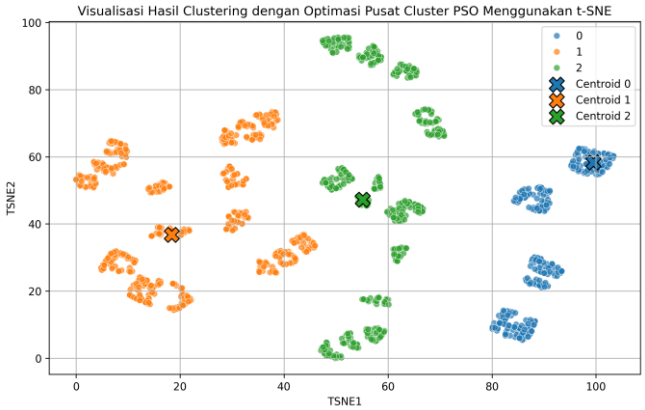
JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)
Journal homepage: https://jurnal.stkippgritulungagung.ac.id/index.php/jipi
ISSN: 2540-8984
Vol. 11, No. 1, March 2026, Pp.77-91

TABEL IV
HASIL PENGUJIAN STABILITAS ALGORITMA OPTIMASI PSO DALAM 5 KALI RUNNING

Running Ke-	Iterasi Maksimal	DBI	Silhouette Score	Waktu Komputasi (s)
1	48	0,6467	0,5498	18,4909
2	47	0,6467	0,5498	17,9540
3	44	0,6467	0,5498	16,4669
4	46	0,6467	0,5498	17,8370
5	52	0,6467	0,5498	19,3832

Hasil pengujian stabilitas algoritma PSO dalam lima kali *running* pada Tabel 4, menunjukkan konsistensi yang sangat baik. Nilai *Davies-Bouldin Index* (DBI) dan *Silhouette Score* tetap stabil pada masing-masing 0,6467 dan 0,5498. Hal ini menunjukkan bahwa kualitas pengelompokan yang dihasilkan sangat konsisten. Jumlah iterasi maksimal untuk mencapai konvergensi berada dalam rentang 44 hingga 52 iterasi, sedangkan waktu komputasi berkisar antara 16,47 hingga 19,38 detik. Hal ini menunjukkan bahwa algoritma PSO tidak hanya andal dalam menghasilkan klaster yang stabil, tetapi juga efisien dari segi waktu dan proses komputasi.

Pusat klaster optimal PSO tersebut, kemudian digunakan sebagai inisialisasi untuk melakukan *clustering* menggunakan algoritma *K-Prototype*. Adapun hasil *clustering K-Prototype* dengan pusat klaster optimal hasil PSO dapat dilihat pada visualisasi menggunakan t-SNE pada Gambar 7 berikut.



Gambar 7. Visualisasi PCA Pemodelan *K-Prototype* dengan Optimasi PSO

Hasil implementasi *K-Prototypes* dengan optimasi pusat klaster menggunakan PSO dataset terbagi kedalam 3 klaster. Klaster 0 dengan 307 anggota, memiliki nilai numerik yang lebih tinggi dan didominasi oleh kategori Baik. Klaster 1 dengan 581 anggota, berada di tengah dengan nilai numerik yang mendekati rata-rata dan didominasi oleh kategori Sedang. Sedangkan, klaster 2 dengan 472 anggota, menunjukkan nilai numerik yang lebih rendah dan didominasi oleh kategori Buruk. Berdasarkan visualisasi PCA pada Gambar 7, metode optimasi PSO mampu menghasilkan pemisahan klaster yang lebih jelas dan terstruktur dibandingkan metode konvensional. Distribusi data yang lebih terorganisir menunjukkan bahwa PSO berhasil meningkatkan akurasi segmentasi dan sepenuhnya mengelompokkan data ke dalam klaster yang sesuai, tanpa adanya kesalahan pemetaan.

E. Genetic Algorithm (GA)

Pada algoritma GA, setiap individu dalam populasi merepresentasikan sekumpulan pusat klaster yang diperbarui melalui seleksi, *crossover*, dan mutasi. Pendekatan *hybrid* GA diterapkan untuk mengoptimalkan algoritma *K-Prototypes* dalam proses *clustering*, dengan parameter 1360 individu, $k = 3$, 100 iterasi, laju mutasi 0,2, serta ukuran turnamen 50. Parameter ini dipilih berdasarkan serangkaian uji coba untuk mencapai konvergensi optimal. Nilai parameter diuji dalam rentang tertentu untuk mengamati pengaruhnya terhadap hasil *clustering*. Kombinasi parameter ini membantu GA menemukan pusat klaster yang lebih stabil dan akurat, sehingga meningkatkan kualitas pengelompokan penerima bantuan Rutilahu. Dalam proses pengujian stabilitas algoritma, dilakukan sebanyak lima kali *running* untuk memastikan konsistensi hasil pengelompokan. Adapun hasil dari lima kali pengujian tersebut disajikan pada Tabel 5 berikut.

TABEL V
HASIL PENGUJIAN STABILITAS ALGORITMA OPTIMASI GA DALAM 5 KALI RUNNING

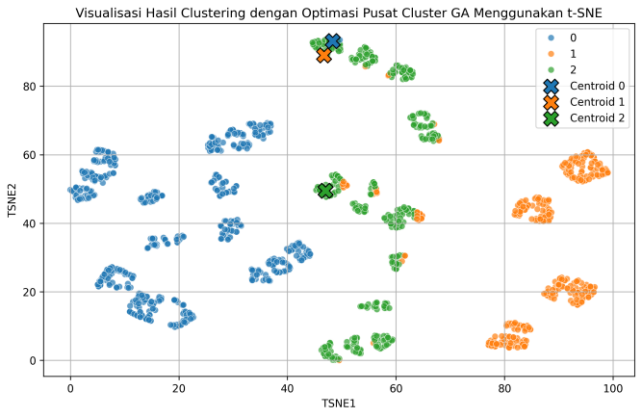
Running Ke-	Iterasi Maksimal	DBI	Silhouette Score	Waktu Komputasi (s)
1	53	0,7134	0,5143	6878,7123
2	48	0,7134	0,5143	6865,0623
3	49	0,7134	0,5143	6865,1825
4	47	0,7134	0,5143	6885,0485
5	50	0,7134	0,5143	6884,3758

Hasil pengujian stabilitas algoritma GA dalam lima kali *running* pada Tabel 5, menunjukkan bahwa algoritma

JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)
Journal homepage: https://jurnal.stkippgritulungagung.ac.id/index.php/jipi
ISSN: 2540-8984
Vol. 11, No. 1, March 2026, Pp.77-91

ini mampu menghasilkan kualitas pengelompokan yang konsisten, dengan nilai *Davies-Bouldin Index* (DBI) sebesar 0,7134 dan *Silhouette Score* sebesar 0,5143 pada seluruh percobaan. Iterasi maksimal yang dibutuhkan untuk mencapai konvergensi berkisar antara 47 hingga 53 iterasi, yang menunjukkan stabilitas proses evolusi individu dalam GA. Namun, waktu komputasi relatif tinggi, berada di kisaran 6865 hingga 6885 detik, menandakan bahwa meskipun GA memberikan hasil yang stabil, ia memerlukan sumber daya komputasi yang jauh lebih besar dibandingkan metode lainnya.

Pusat klaster optimal GA tersebut, kemudian digunakan sebagai inisialisasi untuk melakukan *clustering* menggunakan algoritma *K-Prototype*. Adapun hasil *clustering K-Prototype* dengan pusat klaster optimal hasil GA dapat dilihat pada visualisasi menggunakan t-SNE pada Gambar 8 berikut.



Gambar 8. Visualisasi PCA Pemodelan *K-Prototype* dengan Optimasi GA

Hasil implementasi *K-Prototypes* dengan optimasi pusat klaster menggunakan GA menunjukkan dataset dibagi kedalam 3 klaster. Klaster 0 dengan 582 anggota memiliki nilai numerik yang lebih rendah dan didominasi oleh kategori Buruk. Klaster 1 dengan 354 anggota memiliki nilai numerik yang lebih tinggi dibandingkan klaster lainnya dan didominasi oleh Kategori Baik. Sementara itu, klaster 2 dengan 424 anggota berada di tengah dengan nilai numerik yang mendekati rata-rata dan didominasi oleh kategori Sedang. Berdasarkan visualisasi PCA pada Gambar 8, metode ini menghasilkan pemisahan klaster yang kurang optimal. Hal ini terlihat masih banyak data yang tumpang tindih. Terdapat banyak data yang seharusnya masuk ke dalam klaster 2, tetapi justru terkelompok ke dalam klaster 1. Hal ini mengindikasikan bahwa meskipun optimasi algoritma genetika belum optimal dalam mengurangi *overlap* antar klaster, masih ada kemungkinan kesalahan dalam pemetaan beberapa data, terutama dalam membedakan karakteristik antara klaster 1 dan klaster 2.

F. Flower Pollination Algorithm (FPA)

Pada proses mengoptimalkan penentuan pusat klaster *K-Prorotype*, digunakan FPA untuk menemukan pusat klaster yang lebih stabil dan akurat dengan mempertimbangkan fitur numerik dan kategorik secara bersamaan. Pendekatan *hybrid* FPA diterapkan untuk mengoptimalkan algoritma *K-Prototypes* dalam proses *clustering*, dengan parameter populasi sebanyak 1360 individu, jumlah klaster $(k) = 3$, jumlah iterasi maksimal 100, dan nilai *switching probability* (p) sebesar 0,5. Parameter ini dipilih berdasarkan serangkaian uji coba untuk mencapai konvergensi optimal. Nilai parameter diuji dalam rentang tertentu untuk mengamati pengaruhnya terhadap hasil *clustering*. Kombinasi parameter ini membantu FPA menemukan pusat klaster yang lebih stabil dan akurat, sehingga meningkatkan kualitas pengelompokan penerima bantuan Rutilahu. Dalam proses pengujian stabilitas algoritma, dilakukan sebanyak lima kali running untuk memastikan konsistensi hasil pengelompokan. Adapun hasil dari lima kali pengujian tersebut disajikan pada Tabel 6 berikut.

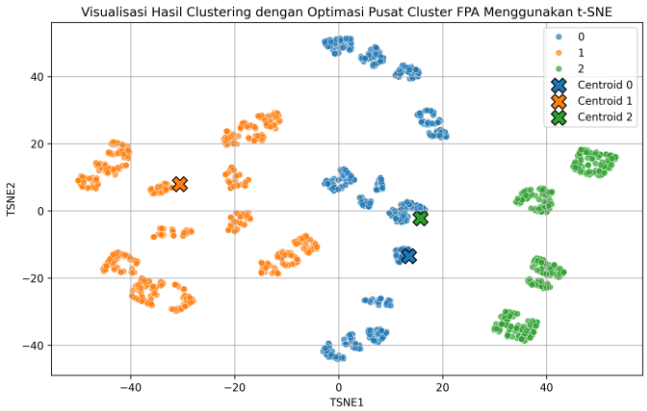
TABEL VI
HASIL PENGUJIAN STABILITAS ALGORITMA OPTIMASI FPA DALAM 5 KALI RUNNING

Running Ke-	Iterasi Maksimal	DBI	Silhouette Score	Waktu Komputasi (s)
1	32	0,6467	0,5498	3422,0102
2	25	0,6467	0,5498	3405,0983
3	27	0,6467	0,5498	3410,1420
4	30	0,6467	0,5498	3417,8825
5	28	0,6467	0,5498	3415,9175

Hasil pengujian stabilitas algoritma FPA dalam lima kali *running* pada Tabel 6, menunjukkan performa yang stabil dan konsisten. Nilai *Davies-Bouldin Index* (DBI) dan *Silhouette Score* tetap konstan pada 0,6467 dan 0,5498, menunjukkan bahwa kualitas pengelompokan yang dihasilkan sangat baik dan tidak terpengaruh oleh inisialisasi awal. Iterasi maksimal yang dibutuhkan untuk mencapai konvergensi berkisar antara 25 hingga 32 iterasi, menunjukkan efisiensi dalam proses optimasi. Waktu komputasi berada di kisaran 3405 hingga 3422 detik, yang relatif

lebih rendah dibandingkan GA namun masih lebih tinggi dibandingkan PSO. Secara keseluruhan, FPA menunjukkan kinerja yang stabil dan efisien dalam menemukan solusi optimal untuk pengelompokan data.

Pusat klaster optimal hasil FPA tersebut, kemudian digunakan sebagai inisialisasi untuk melakukan *clustering K-Prototype*. Adapun hasil *clustering K-Prototype* dengan pusat klaster optimal hasil FPA divisualisasikan menggunakan t-SNE pada Gambar 9 berikut.



Gambar 9.Visualisasi PCA Pemodelan *K-Prototype* dengan Optimasi FPA

Hasil implementasi *K-Prototypes* dengan optimasi pusat klaster menggunakan FPA menunjukkan bahwa dataset terbagi ke dalam 3 klaster. Klaster 0 dengan 472 anggota memiliki nilai numerik yang mendekati rata-rata dan didominasi oleh kategori Sedang. Klaster 1 dengan 581 anggota memiliki nilai numerik yang lebih rendah dibandingkan klaster lainnya dan didominasi oleh kategori Buruk. Sementara itu, klaster 2 dengan 307 anggota dengan nilai numerik yang lebih tinggi disbanding klaster lainnya dan memiliki karakteristik kategori Baik. Berdasarkan visualisasi hasil *clustering* menggunakan PCA pada Gambar 9, optimasi FPA mampu menghasilkan pemisahan klaster yang lebih jelas dan terstruktur dibandingkan metode konvensional. Distribusi data yang lebih terorganisir menunjukkan bahwa PSO berhasil meningkatkan akurasi segmentasi dan sepenuhnya mengelompokkan data ke dalam klaster yang sesuai, tanpa adanya kesalahan pemetaan.

G. *Evaluasi Model*

Evaluasi model dilakukan dengan membandingkan performa model *K-Prototype* konvensional dengan model optimasi pusat klaster *K-Prototype* menggunakan algoritma PSO, GA, dan FPA. Perbandingan dilakukan berdasarkan tiga metrik utama, yaitu nilai *Davies-Bouldin Index* (DBI), *Silhouette Score,* dan waktu komputasi. Adapun hasil evaluasi model dapat dilihat pada Tabel 7 berikut.

TABEL VII
EVALUASI MODEL

Model	DBI	Silhouette Score	μ Waktu Komputasi (s)
K-Prototype Konvensional	0,6468	0,5477	45,4343
K-Prototype + PSO	0,6467	0,5498	18,0264
K-Prototype + GA	0,7134	0,5143	6875,6763
K-Prototype + FPA	0,6467	0,5498	3414,2101

Berdasarkan hasil evaluasi model pada Tabel 7, model *K-Prototype* konvensional memperoleh nilai DBI sebesar 0,6468 dan *Silhouette Score* sebesar 0,5477, dengan waktu komputasi selama 45,4343 detik. Penerapan optimasi *Particle Swarm Optimization* (PSO) pada penentuan pusat klaster *K-Prototype* menghasilkan performa terbaik, ditandai dengan DBI terendah sebesar 0,6467, *Silhouette Score* tertinggi sebesar 0,5498, serta rata-rata waktu komputasi paling efisien, yaitu 18,0264 detik. Model dengan optimasi pusat klaster menggunakan *Genetic Algorithm* (GA) menunjukkan performa terendah, dengan DBI tertinggi sebesar 0,7134, *Silhouette Score* sebesar 0,5143, serta rata-rata waktu komputasi yang jauh lebih lama, yakni 6875,6763 detik, sehingga dinilai kurang optimal dari sisi akurasi maupun efisiensi. Sementara itu, model dengan optimasi menggunakan *Flower Pollination Algorithm* (FPA) mencatat DBI sebesar 0,6467 dan *Silhouette Score* sebesar 0,5498, sama seperti model PSO, namun memiliki rata-rata waktu komputasi yang jauh lebih tinggi, yaitu 3414,2101 detik.

Secara keseluruhan, hasil ini mengindikasikan bahwa PSO merupakan metode optimasi yang paling efektif, karena mampu meningkatkan kualitas hasil *clustering*. Hal ini ditunjukkan oleh nilai DBI dan *Silhouette Score* yang baik, serta mempercepat proses komputasi. Meskipun FPA menghasilkan akurasi serupa, efisiensi waktunya masih menjadi kendala. Sebaliknya, GA memiliki performa paling rendah dibandingkan metode lainnya.

H. Analisis Hasil

Dalam menghadapi tantangan akibat subjektivitas dalam proses penerimaan bantuan rehabilitasi Rutilahu, penelitian ini mengusulkan pendekatan berbasis data dengan *clustering K-Prototype* guna mengelompokkan penerima bantuan secara lebih objektif. Namun, performa *K-Prototype* bergantung pada inisialisasi pusat klaster. Oleh karena itu, dilakukan optimasi menggunakan PSO, GA, dan FPA untuk meningkatkan efektivitas dan efisiensi *clustering*.

Berdasarkan hasil evaluasi model pada Tabel IV, dapat diketahui bahwa model *K-Prototype* konvensional memperoleh DBI sebesar 0,6468 dan *Silhouette Score* sebesar 0,5477, dengan waktu komputasi selama 45,4343 detik. Sementara itu, penerapan algoritma PSO sebagai metode optimasi menghasilkan performa terbaik, ditandai dengan DBI terendah yaitu 0,6467, *Silhouette Score* tertinggi yaitu 0,5498, serta rata-rata waktu komputasi tercepat sebesar 18,0264 detik. Model dengan optimasi menggunakan GA menunjukkan performa terendah, dengan DBI tertinggi sebesar 0,7134 dan *Silhouette Score* terendah sebesar 0,5143, serta rata-rata waktu komputasi yang sangat tinggi, yakni 6875,6763 detik. Hal ini menunjukkan bahwa model cenderung menghasilkan klaster yang kurang kompak dan tidak terpisah dengan baik. Di sisi lain, FPA menunjukkan hasil yang serupa dengan PSO dari segi kualitas klaster dengan DBI sebesar 0,6467 dan *Silhouette Score* sebesar 0,5498, tetapi memerlukan waktu komputasi jauh lebih besar, yaitu 3414,2101 detik.

Hasil ini menunjukkan bahwa PSO merupakan metode optimasi yang paling efektif, baik dari segi akurasi hasil pengelompokan (DBI dan *Silhouette Score*) maupun efisiensi waktu komputasi. Keunggulan PSO dibandingkan dua metode lainnya tidak hanya berasal dari kerangka algoritmiknya yang sederhana dan efisien, tetapi juga dari kecocokan karakteristik algoritma dengan struktur data yang digunakan. Distribusi data Rutilahu yang digunakan dalam penelitian ini memiliki beberapa karakteristik penting yang mendukung performa PSO secara optimal. Berdasarkan hasil eksplorasi data, diketahui bahwa terdapat korelasi positif yang kuat antara pendapatan dan luas rumah, dengan r = 0,878. Terdapat korelasi negatif yang signifikan antara jumlah tanggungan dan pendapatan, dengan r = -0,814 serta jumlah tanggungan dan luas rumah, dengan r = -0,820. Hal ini menunjukkan bahwa data numerik memiliki pola linear yang kuat dan konsisten, yang dapat dengan mudah dimanfaatkan oleh PSO dalam mencari posisi pusat klaster yang optimal. Selain itu, distribusi fitur kategorik seperti kondisi atap, dinding, dan lantai juga menunjukkan variasi yang cukup merata antar kategori, dengan dominasi yang tidak terlalu ekstrem. Variasi ini memungkinkan proses pencarian solusi dalam PSO untuk lebih cepat menemukan konvergensi optimal, karena setiap partikel memiliki representasi yang cukup lengkap terhadap kondisi kategori.

PSO bekerja dengan prinsip iteratif di mana setiap partikel diperbarui berdasarkan pengalaman terbaik pribadi dan kolektif dalam ruang pencarian solusi. Mekanisme ini mempercepat proses menuju solusi yang optimal tanpa eksplorasi berlebihan seperti pada GA (dengan seleksi, *crossover*, dan mutasi) atau FPA (dengan kombinasi *Lévy flight* dan penyerbukan lokal). Karena dataset Rutilahu memiliki dimensi yang cukup besar yaitu 1.360 baris dan atribut campuran yang cukup kompleks, pendekatan berbasis adaptasi posisi seperti pada PSO menjadi lebih efisien dibanding eksplorasi acak yang cenderung memerlukan iterasi lebih banyak untuk konvergensi.

Di sisi lain, GA menunjukkan performa yang lebih rendah karena proses evolusinya melibatkan eksplorasi yang lebih luas, namun berisiko menghasilkan individu yang tidak stabil dalam jangka pendek. Mutasi dan crossover yang bersifat acak sering kali menghasilkan pusat klaster yang kurang representatif terhadap struktur data sebenarnya, sehingga mengurangi efektivitas pemisahan klaster. FPA, meskipun mampu memberikan hasil yang sama dengan PSO dari segi evaluasi DBI dan *Silhouette Score*. Namun, FPA memiliki waktu komputasi yang jauh lebih besar. Hal ini kemungkinan besar disebabkan oleh mekanisme Lévy flight yang memperluas ruang pencarian, namun memerlukan waktu lebih lama untuk konvergensi terhadap pusat klaster yang stabil.

Pada proses pengujian yang dilakukan dengan lima kali *running* di masing-masing algoritma optimasi. Stabilitas ketiga algoritma optimasi menunjukkan konsistensi yang berbeda-beda. Variasi waktu komputasi pada masing-masing metode disebabkan oleh perbedaan jumlah iterasi konvergensi yang dicapai setiap kali algoritma dijalankan. Hal ini mencerminkan karakteristik stokastik dari ketiga algoritma, di mana proses inisialisasi acak dan jalannya optimasi dapat mempengaruhi kecepatan konvergensi menuju solusi optimal. PSO mencatat iterasi antara 44 hingga 52 dengan waktu komputasi 16-19 detik, GA antara 47 hingga 53 iterasi dengan waktu sangat tinggi sekitar 6865–6885 detik, dan FPA antara 25 hingga 32 iterasi dengan waktu 3405–3422 detik. Meski iterasinya bervariasi, ketiganya menunjukkan hasil evaluasi klaster yang konsisten di setiap percobaan.

Berdasarkan pengujian stabilitas algoritma optimasi tersebut, PSO menunjukkan stabilitas paling tinggi, dengan nilai DBI dan *Silhouette Score* yang konsisten, serta waktu komputasi yang relatif singkat dan stabil. Hal ini mencerminkan keandalan PSO dalam menghasilkan klaster yang tidak hanya akurat, tetapi juga replikatif dalam kondisi inisialisasi yang berbeda. FPA juga menunjukkan performa yang konsisten, dengan nilai evaluasi yang sama di semua percobaan, meskipun waktu komputasinya tetap tinggi. Sementara itu, GA meskipun juga menunjukkan hasil evaluasi yang konsisten, memiliki kelemahan utama pada waktu komputasi yang sangat tinggi dan tidak

efisien. Oleh karena itu, PSO tidak hanya unggul dalam akurasi dan efisiensi, tetapi juga dalam hal stabilitas performa, menjadikannya pilihan utama dalam pengembangan sistem pendukung keputusan berbasis *clustering* untuk alokasi bantuan sosial yang lebih adil dan objektif.

Hasil klasterisasi dari model PSO menghasilkan tiga klaster utama yang secara praktis dapat dikategorikan sebagai "Buruk", "Sedang", dan "Baik", berdasarkan kombinasi variabel kondisi fisik rumah seperti atap, dinding, dan lantai serta indikator sosial ekonomi seperti pendapatan dan jumlah tanggungan. Klaster "Buruk" mencerminkan kelompok penerima bantuan dengan kondisi rumah sangat tidak layak, penghasilan rumah tangga yang rendah, serta jumlah tanggungan yang tinggi. Oleh karena itu, kelompok ini direkomendasikan untuk mendapatkan prioritas utama dalam distribusi bantuan Rehabilitasi Rumah Tidak Layak Huni (Rutilahu), karena tingkat kerentanannya paling tinggi. Dalam implementasinya, pemerintah daerah dapat menetapkan ambang batas prioritas berdasarkan skor gabungan antar-individu di dalam klaster tersebut, misalnya menggunakan nilai rata-rata atau rentang nilai *Davies-Bouldin Index* (DBI) yang menggambarkan kedekatan antar-anggota dalam satu klaster dan jarak antar klaster.

Sementara itu, klaster "Sedang" menggambarkan kelompok dengan kondisi rumah yang cukup layak tetapi masih memiliki aspek kerentanan tertentu, seperti pendapatan yang belum stabil atau jumlah tanggungan yang relatif banyak. Untuk kelompok ini, kebijakan yang dapat diterapkan dengan dimasukkan ke dalam daftar tunggu untuk gelombang bantuan berikutnya dalam satu periode. Klaster ini tetap perlu dipantau karena memiliki potensi untuk masuk ke dalam kategori prioritas jika terjadi perubahan kondisi ekonomi rumah tangga.

Adapun klaster "Baik" mencakup rumah tangga dengan kondisi rumah yang relatif layak, pendapatan yang cukup memadai, dan jumlah tanggungan yang rendah. Kelompok ini tidak direkomendasikan untuk menerima bantuan dalam periode berjalan, tetapi dapat diberikan intervensi dalam bentuk monitoring berkala atau program peningkatan kualitas hidup non-fisik, seperti pelatihan kewirausahaan atau akses ke layanan sosial lainnya. Dengan pendekatan ini, hasil klasterisasi tidak hanya menjadi alat analisis evaluatif, tetapi juga dapat menjadi dasar formulasi kebijakan penyaluran bantuan yang lebih objektif, adil, dan berbasis data. Pendekatan ini sekaligus mendukung prinsip transparansi dan akuntabilitas dalam tata kelola program sosial di tingkat daerah.

Meskipun penelitian ini difokuskan pada data Rutilahu, metodologi yang digunakan, terutama kombinasi *K-Prototype* dan optimasi PSO, memiliki potensi generalisasi yang tinggi terhadap jenis data sosial lainnya, seperti bantuan pendidikan, bantuan pangan, atau program jaminan kesehatan. Hal ini disebabkan oleh kesamaan karakteristik data, yakni adanya atribut campuran numerik dan kategorik, serta kebutuhan untuk mengelompokkan individu atau rumah tangga berdasarkan kondisi multi-dimensional. Oleh karena itu, hasil dan pendekatan dari penelitian ini dapat direplikasi dan disesuaikan untuk konteks sosial lainnya, guna memperkuat pengambilan keputusan berbasis data dalam berbagai program kesejahteraan masyarakat.

IV. KESIMPULAN

Penelitian ini membandingkan efektivitas optimasi pusat klaster algoritma *K-Prototypes* menggunakan *Particle Swarm Optimization* (PSO), *Genetic Algorithm* (GA), dan *Flower Pollination Algorithm* (FPA) dalam meningkatkan akurasi dan efisiensi pengelompokan data penerima bantuan rehabilitasi Rumah Tidak Layak Huni (Rutilahu) di Kota Surabaya. Evaluasi performa model dilakukan menggunakan tiga metrik utama yaitu *Davies-Bouldin Index* (DBI), *Silhouette Score*, dan waktu komputasi. Hasil evaluasi menunjukkan bahwa model *K-Prototypes* + PSO memberikan performa terbaik, dengan DBI terendah sebesar 0,6467, *Silhouette Score* tertinggi sebesar 0,5498, serta rata-rata waktu komputasi tercepat yaitu 18,0264 detik. Nilai DBI yang lebih rendah menunjukkan bahwa klaster yang terbentuk lebih kompak dan terpisah dengan baik, menandakan kualitas pengelompokan yang optimal. Sebaliknya, model dengan optimasi GA menunjukkan performa terendah, dengan DBI tertinggi sebesar 0,7134, *Silhouette Score* sebesar 0,5143, dan rata-rata waktu komputasi yang sangat tinggi yaitu 6875,6763 detik. Hal ini menunjukkan bahwa pengelompokan menggunakan GA menghasilkan klaster yang kurang optimal dan membutuhkan sumber daya komputasi yang besar. Sementara itu, FPA menghasilkan performa yang setara dengan PSO dalam hal DBI sebesar 0,6467 dan *Silhouette Score* sebesar 0,5498, namun dengan waktu komputasi yang jauh lebih lama sebesar 3414,2101 detik. Hal ini menjadikan PSO lebih unggul dari sisi efisiensi. Secara keseluruhan, implementasi PSO terbukti paling optimal dalam meningkatkan kualitas pengelompokan dan efisiensi proses komputasi. Hasil ini menunjukkan bahwa PSO adalah metode optimasi yang lebih sesuai untuk diterapkan dalam pengembangan sistem pendukung keputusan berbasis *clustering*, guna meningkatkan ketepatan seleksi penerima bantuan dan memastikan distribusi bantuan yang lebih adil serta objektif.

DAFTAR PUSTAKA

[1] S. Hidayati, A. T. Darmaliana, and R. Riski, "Comparison of K-Means, Fuzzy C-Means, Fuzzy Gustafson Kessel, and DBSCAN for Village Grouping in Surabaya Based on Poverty Indicators," *Jurnal Pendidikan Matematika (Kudus)*, vol. 5, no. 2, 2022, doi: 10.21043/jpmk.v5i2.16552.

[2] A. Anilshi, R. Thimotius Abineno, and A. Aha Pekuwali, "Penerapan Algoritma K-Means Clustering Nilai Kemampuan Siswa Dalam Pelajaran Matematika," in *Prosiding Seminar Nasional Sustainable Agricultural Technology Innovation (SATI)*, Fakultas Sains dan Teknologi Universitas Kristen Wira Wacana Sumba, Aug. 2024, pp. 720–743. doi: https://ojs.unkriswina.ac.id/index.php/semnas-FST.

[3] A. F. Purba, M. Mustafid, and P. Kartikasari, "Penerapan Algoritma K-Prototype Untuk Pengelompokan Desa Di Kabupaten Bekasi Berdasarkan Infrastruktur Digital," *Jurnal Gaussian*, vol. 13, no. 2, pp. 479–489, Jan. 2025, doi: 10.14710/j.gauss.13.2.479-489.

[4] P. Suwirmayanti, G. Darma Putra, and S. Kumara, "Optimasi Pusat Cluster K-Prototype dengan Algoritma Genetika," *Majalah Ilmiah Teknologi Elektro*, vol. 13, no. 2, Dec. 2020, doi: 10.24843/10.24843/MITE.2015.v13i02p02.

[5] H. Harliana, R. M. Herdian Bhakti, O. Saeful Bachri, and F. Sofian Efendi, "Optimasi K-Means dengan Particle Swarm Optimization pada Pengelompokkan Daerah Stunting," *Jurnal Ilmiah Intech : Information Technology Journal of UMUS*, vol. 3, no. 02, 2021, doi: 10.46772/intech.v3i02.457.

[6] M. I. Hanafi, Billy Coster Junior, and Yesy Diah Rosita, "Optimasi Rute Penyebaran Brosur Bimbingan Belajar Menggunakan Algoritma Genetika," *Jurnal Informatika Teknologi dan Sains (Jinteks)*, vol. 5, no. 2, 2023, doi: 10.51401/jinteks.v5i2.2319.

[7] R. Karim and Salim, "Penerapan Flower Pollination Algorithm dengan Teknik Clustering dalam Penyelesaian Masalah Diophantine," *Techno.Com*, vol. 22, no. 2, 2023, doi: 10.33633/tc.v22i2.7174.

[8] D. B. Wijaya, E. Noersasongko, and P. Purwanto, "Optimasi Centroid Awal Algoritma K-Medoids Menggunakan Particle Swarm Optimization Untuk Segmentasi Customer," *Techno.Com*, vol. 23, no. 1, 2024, doi: 10.62411/tc.v23i1.9516.

[9] T. M. S. Karim, H. S. Panigoro, and L. O. Nashar, "Implementasi Metode K-Prototype Menggunakan Optimasi Flower Pollination Algorithm Pada Data Penderita Diabetes Melitus Tipe 2 di Kota Gorontalo," Universitas Negeri Gorontalo, Gorontalo, 2024.

[10] S. Azahra, Zilrahmi, D. Vionanda, and F. Fitri, "K-Modes Analysis with Validation of the DBI in Grouping Provinces in Indonesia based on Indicators of Poor Households," *UNP Journal of Statistics and Data Science*, vol. 2, no. 2, pp. 173–178, May 2024, doi: 10.24036/ujsds/vol2-iss2/165.

[11] T. Rahmawati, Y. Wilandari, and P. Kartikasari, "Analisis Perbandingan Silhouette Coefficient Dan Metode Elbow Pada Pengelompokkan Provinsi Di Indonesia Berdasarkan Indikator Ipm Dengan K-Medoids," *Jurnal Gaussian*, vol. 13, no. 1, pp. 13–24, Aug. 2024, doi: 10.14710/j.gauss.13.1.13-24.

[12] J. A. P. Setyawan and A. Prathama, "Implementasi Kebijakan Program Rehabilitasi Rumah Tidak Layak Huni Di Kota Surabaya (Studi Kasus Kelurahan Sawunggaling)," *Jdp (Jurnal Dinamika Pemerintahan)*, vol. 7, no. 1, 2024, doi: 10.36341/jdp.v7i1.3905.

[13] Y. Kusnadi and M. S. Putri, "Clustering Menggunakan Metode K-Means Untuk Menentukan Prioritas Penerima Bantuan Bedah Rumah (Studi Kasus : Desa Ciomas Bogor)," *Jurnal Teknologi Informatika dan Komputer*, vol. 7, no. 1, 2021, doi: 10.37012/jtik.v7i1.498.

[14] N. Haya Zen, L. Mageiasti, and Yulhendri, "Analisis Penerapan Sdgs Dalam Pembangunan Berkelanjutan Di Indonesia: Tinjauan Literatur Dan Tantangan Implementasi," *Gudang Jurnal Multidisiplin Ilmu*, vol. 3, no. 1, pp. 775–785, Jan. 2025, doi: 10.59435/gjmi.v3i1.1316.

[15] A. Muhaimin *et al.*, "Sentiment Analysis in Social Media: Case Study in Indonesia," in *Nusantara Science and Technology Proceedings*, Galaxy Science, May 2024, pp. 27–30. doi: 10.11594/nstp.2024.4106.

[16] T. M. Fahrudin, P. A. Riyantoko, and K. M. Hindrayani, "Implementation of Web Scraping on Google Search Engine for Text Collection Into Structured 2D List," *Telematika*, vol. 20, no. 2, 2023, doi: 10.31315/telematika.v20i2.9575.

[17] P. A. Riyantoko, T. M. Fahrudin, K. M. Hindrayani, and M. Idhom, "Exploratory Data Analysis and Machine Learning Algorithms to Classifying Stroke Disease," *IJCONSIST JOURNALS*, vol. 2, no. 02, 2021, doi: 10.33005/ijconsist.v2i02.49.

[18] S. Renaldi Sahat, P. Dwi Arman, and M. Amri, "Analisis Klaster Partitioning Around Medoids dengan Gower Distance untuk Rekomendasi Indekos (Studi Kasus: Indekos di Sekitar Kampus UPNVJT)," *G-Tech : Jurnal Teknologi Terapan*, vol. 8, no. 3, pp. 2060–2068, Jul. 2024.

[19] A. Agung, A. Daniswara, I. Kadek, and D. Nuryana, "Data Preprocessing Pola Pada Penilaian Mahasiswa Program Profesi Guru," *Journal of Informatics and Computer Science*, vol. 05, 2023, doi: https://doi.org/10.26740/jinacs.v5n01.p97-100.

[20] N. Widaad and D. Anggraini, "Sentiment Analysis Of Chatgpt App User Reviews Using Svm And Cnn Methods," *Jurnal Teknik Informatika (JUTIF)*, vol. 5, no. 6, pp. 1687–1700, 2024, doi: 10.52436/1.jutif.2024.5.6.4010.

[21] N. Saniyah, N. Suarna, and W. Prihartono, "Clustering Pelanggaran Lalu Lintas Pada Kendaraan Bermotor Menggunkan Algoritma K-Prototype (Studi Kasus: Pengadilan Negeri Cirebon)," 2024. doi: https://doi.org/10.36040/jati.v8i1.8420.

[22] M. Ganmanah and A. Kudus, "Penerapan Algoritme K-Prototypes untuk Pengelompokkan Desa-Desa di Provinsi Jawa Barat Berdasarkan Indikator Indeks Desa Membangun Tahun 2020," *Prosiding Statistika*, vol. 7, no. 2, 2021.

[23] D. Arman Prasetya, A. P. Sari, M. Idhom, and A. Lisanthoni, "Optimizing Clustering Analysis to Identify High-Potential Markets for Indonesian Tuber Exports," *IJEEEMI : Indonesian Journal of Electronics, Electromedical Engineering, and Medical Informatics*, vol. 7, no. 1, pp. 113–122, 2025, doi: 10.35882/ijeeemi.v7i1.55.

[24] Y. Istianto and S. 'Uyun, "Klasifikasi Kebutuhan Jumlah Produk Makanan Customer Menggunakan K-Means Clustering dengan Optimasi Pusat Awal Cluster Algoritma Genetika," *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 8, no. 5, 2021, doi: 10.25126/jtiik.2021842990.

[25] D. Ghaniyyu Muqsit, S. Achmadi, and Y. Agus Pranoto, "Penerapan Algoritma Genetika Untuk Penjadwalan Mata Pelajaran," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 7, no. 5, 2024, doi: 10.36040/jati.v7i5.7569.

[26] P. E. Mergos and X. S. Yang, "Flower pollination algorithm parameters tuning," *Soft comput*, vol. 25, no. 22, 2021, doi: 10.1007/s00500-021-06230-1.

[27] R. Karim, K. A. Sidarto, and S. Bantun, "Optimasi Fungsi Multimodal Menggunakan Flower Pollination Algorithm Dengan Teknik Clustering," *Techno.Com*, vol. 19, no. 2, 2020, doi: 10.33633/tc.v19i2.3216.

[28] N. A. Maori and E. Evanita, "Metode Elbow dalam Optimasi Jumlah Cluster pada K-Means Clustering," *Simetris: Jurnal Teknik Mesin, Elektro dan Ilmu Komputer*, vol. 14, no. 2, 2023, doi: 10.24176/simet.v14i2.9630.

[29] Y. A. Wijaya, D. A. Kurniady, E. Setyanto, W. S. Tarihoran, D. Rusmana, and R. Rahim, "Davies Bouldin Index Algorithm for Optimizing Clustering Case Studies Mapping School Facilities," *TEM Journal*, vol. 10, no. 3, 2021, doi: 10.18421/TEM103-13.

[30] K. R. Shahapure and C. Nicholas, "Cluster quality analysis using silhouette score," in *Proceedings - 2020 IEEE 7th International Conference on Data Science and Advanced Analytics, DSAA 2020*, 2020. doi: 10.1109/DSAA49011.2020.00096.

[31] T. M. Fahrudin, P. A. Riyantoko, K. M. Hindrayani, and M. H. P. Swari, "Cluster Analysis of Hospital Inpatient Service Efficiency Based on BOR, BTO, TOI, AvLOS Indicators using Agglomerative Hierarchical Clustering," *Telematika*, vol. 18, no. 2, 2021, doi: 10.31315/telematika.v18i2.4786.