JOISIE Journal Of Information System And Informatics Engineering
Vol. 6, No.1, Juni 2022, Hlm 15-24

p- ISSN: 2503-5304
e- ISSN: 2527-3116

PENERAPAN *ALGORITMA PILLAR* UNTUK OPTIMASI PENENTUAN TITIK AWAL *CENTROID* PADA *ALGORITMA K-MEANS CLUSTERING*

Dewi Lestari[1)], Abd. Charis Fauzan[2*)], Harliana[3)].
[1,2,3] Fakultas Ilmu Eksakta, Program Studi Ilmu Komputer,Universitas Nahdlatul Ulama Blitar, Jl. Masjid No.22
email: [1]dsuprapto46@gmail.com, [2]**abdcharis@unublitar.ac.id**, [3]harliana@unublitar.ac.id

Abstract

Data sets are data that still need to be processed into information using data mining methods, one of which is the K-Means method. However, this method still has a weakness when the initial centroid is still done randomly, so it is necessary to optimize it with the Pillar algorithm. The dataset used comes from KAAGLE which takes a representation of a person's stress level using eight attributes and five classes. The dataset will be processed by pre-processing, Pillar algorithm calculation, and k-means method calculation. based on these stages, this research will produce the optimal centroid value, namely when the maximum euclidian distance of each centroid must be less than the same as the environmental limit value (nbdis) and more than equal to the maximum value (nmin) so that the optimal initial centroid is obtained with the Pillar algorithm. at id 336, 228, 35, 29, 506. Based on the results of the study, shows that the pillar algorithm can improve grouping by evaluating the Davies Bouldin Index (DBI) of 0.00000473 and speed up the iteration which was originally obtained in the 23rd iteration to the iteration of the 23rd iteration. 17 clustering results have been found.

Keywords: *Algoritma Pillar, Davies BouldinIndex(DBI), K-means.*

Abstrak

Data set merupakan data yang masih perlu diolah menjadi sebuah informasi dengan menggunakan metode data mining salah satunya yaitu metode K-Means. Namun metode ini masih mempunyai kelemahan pada saat pengambilan centroid awal masih dilakukan secara acak, sehingga perlu dilakukan optimalisasi dengan algoritma Pillar. Dataset yang digunakan berasal dari KAAGLE yang mengambil representasi tingkat stress seseorang dengan menggunakan delapan atribut dan lima class. Datset tersebut akan diolah dengan tahapan pre processing, perhitungan algoritma Pillar, dan perhitungan metode k-means. berdasarkan tahapan tersebut, maka penelitian ini akan menghasilkan nilai centroid yang optimal yaitu saat jarak euclidience distance maksimal setiap centroid harus kurang dari sama dengan nilai batas lingkungan (nbdis) dan lebih dari sama dengan nilai maksimal (nmin) sehingga diperoleh centroid awal optimal dengan algoritma Pillar pada id 336, 228, 35, 29, 506. Berdasarkan hasil penelitian menunjukkan bahwa algoritma pillar mampu meningkatkan pengelompokan dengan evaluasi Davies Bouldin Index (DBI) sebesar 0,00000473 dan mempercepat iterasi yang semula hasil diperoleh pada iterasi ke-23 menjadi iterasi ke-17 hasil pengklusteran sudah ditemukan.

Kata Kunci: *Algoritma Pillar, Davies BouldinIndex(DBI), K-means.*

1. PENDAHULUAN

Data set merupakan data yang masih perlu diolah menjadi sebuah informasi dengan melakukan Pengelompokan data atau biasa disebut dengan data *clustering*, dalam mengolah data hingga dapat menjadi sebuah informasi salah satunya menggunakan metode yang sudah tidak asing lagi yaitu metode *K-means*. Kelebihan dari metode *K-means* ini mudah untuk diterapkan karena menghitung nilai jarak *centroid* setiap *cluster* dengan menggunakan rumus *Euclidience distance* yang sangat umum digunakan (Rahayu et al., 2019). Akan tetapi kelemahan algoritma *K-means* awal perhitungan menggunakan nilai C *(centroid)* secara acak dan akan menyebabkan hasil pengelompokan anggota *cluster* yang acak pula. Ketergantungan pada nilai C *(centroid)* membuat akurasi pada algoritma *K-means* kurang maksimal (Primandana et al., 2019).

Menurut penelitian Dinata, hasil klusterisasi dengan *k-means* saja kurang

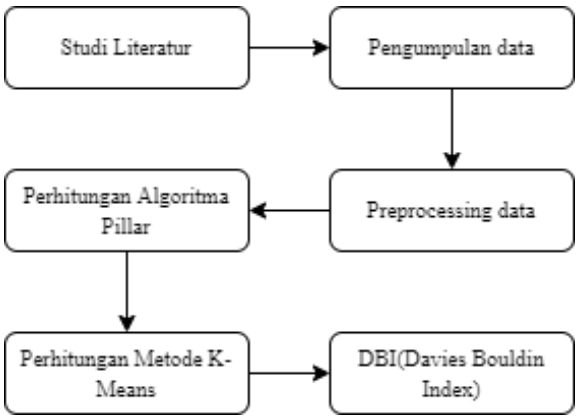
optimal, perlu dilakukan optimasi *centroid* menggunakan metode *Reduksi* atribut dengan *information gain* dengan memperoleh hasil evaluasi dengan Davies Bouldin Index (DBI) sebesar 1,8117. Sedangkan dihitung dengan *k-means* saja memperoleh nilai evaluasi DBI sebesar 2,1972 (Dinata et al., 2020).

Menurut Primandana, kelemahan *K-means* dapat diatasi dengan algoritma *Pillar* untuk pengambilan *centroid* awal yang optimal (Primandana et al., 2019). Algoritma *Pillar* menentukan titik pusat *cluster* dengan cara memilih data dengan nilai *euclidean* paling jauh dari titik pusat *cluster*. Namun pemilihan titik *cluster* tetap memperhatikan kemungkinan data *outlier*. Pengujian dilakukan dengan menetapkan satu buah *cluster* awal sebagai inisialisasi sekaligus sebagai *cluster* pembanding untuk menentukan kualitas *cluster* berikutnya (Ketut Agus Seputra, 2020).

Namun pada penelitian ini algoritma *Pillar* digunakan untuk menentukan posisi *centroid* awal dengan menghitung jarak *akumulasi metrik* antara setiap data dengan semua *centroid* sebelumnya. Pemilihan titik ditentukan oleh titik data yang memiliki jarak maksimum. Sehingga dapat menemukan semua *centroid* yang terpisah sejauh mungkin dari *centroid* awal pada *data set* (Primandana et al., 2019).

Pada penelitian ini untuk optimasi *centroid* pada perhitungan metode *K-means Clustering* menggunakan algoritma *Pillar* dikarenakan lebih memperhatikan titik pusat *centroid* terjauh sehingga dapat menemukan titik pusat *centroid* yang seimbang atau optimal dan jika menggunakan metode lain seperti algoritma *Genetika* untuk penentuan optimasi *centroid* awal kurang cocok dikarenakan cara kerja metode ini memanipulasi kode dari parameter yang telah ada (*kromosom)* dan pencarian titik bersifat seperti teori peluang (Kuntjoro et al., 2018). Sehingga dipilihlah algoritma *Pillar* dalam mengatasi kelemahan *K-means Clustering* untuk memaksimalkan pemilihan nilai *centroid* awal yang lebih optimal.

2. METODE PENELITIAN



Gambar 1. Alur Metode Penelitian

Gambar 1 merupakan rancangan metode penelitian yang akan dibuat langkah pertama mencari Studi Literatur, mempersiapkan *dataset* atau menormalisasi data agar data siap di terapkan dalam perhitungan algoritma *Pillar* maupun *K-means* (A L Ramdani & H B Firmansyah, 2019).

1. Studi Literatur

Studi literatur dalam pengembangan penelitian ini didapat dari journal, artikel ilmiah, maupun e-book studi literatur yang diperlukan sebagai berikut :

a. Metode *K-means Clustering*
b. Tahapan *K-means*
c. Algoritma *Pillar*
d. Davies Bouldin Index(DBI)

2. Pengumpulan Data

Pengumpulan data diperolaeh dari data *UCI Machine Learning* Berikut *data set* identifikasi tingkat level stress yang akan diolah dalam penelitian ini:

Tabel 1. Data Set Awal

Id Pasien	**sr**	**rr**	**t**	**lm**	**bo**
			Fitur		
Id 1	93,8	25,7	91,8	16,6	89,84
Id 2	91,64	25,1	91,6	15,9	89,55
Id 3	60	20	96	10	95
...					
Id 630	73,92	21,4	93,4	11,4	91,39
Id		**rem**	**sr2**	**hr**	**sl**
Pasien		Fitur			Class
Id 1		99,6	1,8	74	3

Id 2	98,9	1,6	73	3
Id 3	85	7	60	1
...				
Id 630	92	4,1	63	2

Tabel 1 merupakan *data set* awal yang belum diolah sama sekali dan *data set* ini akan diproses menggunakan metode yang sudah ditentukan, tabel diatas memiliki delapan kolom fitur dan satu kolom kelas. Tabel diatas merupakan potongan excel *dataset* yang sebenarnya datanya berjumlah 630 data dan memiliki empat kelas. Data yang akan diolah diatas merupakan data klasifikasi dan dapat digunakan untuk klustering dan sebaliknya jika data klustering dapat untuk klasifikasi dengan cara mereduksi keterbilangan nilai atribut secara bertingkat *(hirarki*) (Suyanto, 2019). Fitur dalam tabel diatas yaitu sr (dengkuran), rr (laju pernapasan), T (suhu tubuh), lm (laju gerakan anggota tubuh), bo (kadar oksigen darah), rem (pergerakan mata), sr (jumlah jam tidur), hr (detak jantung), dan sl (Tingkat Stres 0 - rendah/normal, 1 – sedang rendah, 2-sedang, 3-sedang tinggi, 4-tinggi), (Rachakonda et al., 2020).

3. *Pre Processing* Data

Sebelum data di terapkan pada metode *K-means Clustering* dilakukan preprocessing data dengan menormalisasi atau membuat ID pada *dataset* untuk mempermudah perhitungan. Dan dalam tahapan normalisasi data sebelum melalui tahapan perhitungan pada algoritma terdapat 4 tahap yaitu *cleaning* data (pembersihan data), *reduction* data (pengurangan volume data), *transformation* data (konversi data), serta *integrasi* data (kombinasi data). data yang diperlukan tahapan normalisasi sebelum perhitungan yaitu data yang tidak lengkap terdapat *noisy*, tidak konsisten, duplikat dan *over capacity* (Tommy, 2021).

4. Perhitungan Algoritma *Pillar*

Langkah selanjutnya menentukan *centroid* yang optimal dengan menggunakan algoritma *Pillar* sebelum diproses dengan Perhitungan *K-means* dengan tahapan perhitungan pengambilan *centroid* awal yang optimal sebagai berikut (Primandana et al., 2019):

1. Hitung m. m adalah titik tengah data atau rata-rata data.
2. Hitung jarak setiap data(x) terhadap m.
3. Menghitung nilai nmin pada Persamaan (1).

$$nmin = a\frac{n}{k} \quad (1)$$

4. dmax = data dengan jarak terjauh dari m.
5. Tentukan nbdis yaitu batas lingkungan pada Persamaan (2).

$$nbdis = \beta . dmax \quad (2)$$

6. Tentukan i = 1 sebagai penghitung untuk menentukan *centroid* awal ke-i
7. Menghitung nilai DM pada Persamaan (3).

$$DM = DM + D \quad (3)$$

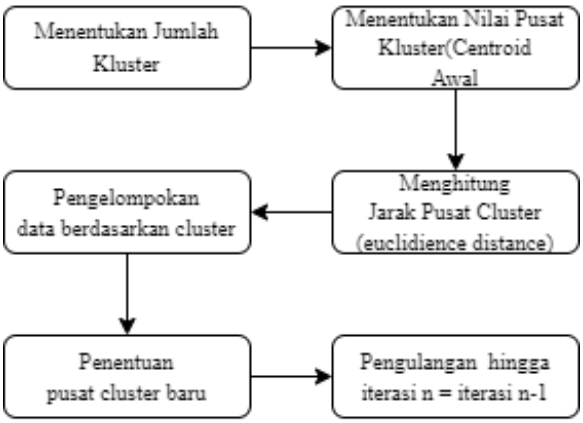
8. Pilih calon *centroid* (c) = data dengan nilai DM tertinggi
9. Menghitung nilai SX pada Persamaan (11).

$$SX = SX \cup C \quad (4)$$

10. D sebagai jarak setiap data(x) ke C.
11. Tentukan *no* = banyak data yang memenuhi D ≤ *nbdis*
12. Menetapkan nilai DM(C) = 0
13. Jika *no* < *nmin*, kembali ke langkah 8
14. Menetapkan nilai D(SX) = 0
15. Menentukan nilai C = C ∪ c
16. Menambahkan iterasi *i* = *i* + 1
17. Jika *i* ≤ k, kembali ke langkah 7
18. Selesai, dimana C adalah solusi sebagai *centroid* awal yang dioptimalkan.

5. Perhitungan Algoritma *K-means Clustering*

Setelah mendapat hasil *centroid* yang optimal lalu dihitung dengan metode *K-means* menggunakan rumus *Euclidience distance*. Dan hasil akhir Pengelompokan data berdasarkan *cluster* (Ariasa et al., 2020), (Rustam, 2020), (Baharuddin et al., 2019).



Gambar 2. Tahapan Metode *K-means*

Pada Gambar 2, tahapan dalam Perhitungan *K-means* sebelum ke-inti perhitungan yaitu menghitung jarak pusat *cluster* dengan rumus *Euclidience distance* pada Persamaan (5) (Ramadhan et al., 2020), (Syahputra et al., 2020).

$$d(x,y) = \sqrt{(\sum_{i=1}^{n}(xi - yi)^2)} \quad (5)$$

Keterangan :

Xi = sampel data

Yi = data uji

i = variabel data

6. Evaluasi

Menggunakan algoritma (DBI) Davies Bouldin Index evaluasi DBI ini mencari nilai dengan Menghitung nilai SSW yaitu untuk mengetahui matrik kohesi dalam sebuah *cluster* ke-*i* yang dirumuskan pada Persamaan (6)

$$SSW_i = \frac{1}{m_i}\sum_{j=1}^{m_i} d(x_j, c_i) \quad (6)$$

Keterangan :

SSW : sum of squere within *cluster*

$\boldsymbol{m_i}$: Jumlah data dalam kluster ke- *i*

$\boldsymbol{c_i}$: *Centroid cluster* ke-i

$\mathbf{d}(\boldsymbol{x_j}, \boldsymbol{c_i})$: Jarak dari data ke-i ke titik *cluster i*

untuk langkah selanjutnya menggunakan Persamaan (7) untuk mengetahui separasi ante *cluster*.

$$SSB_{ij} = d(i,j) \quad (7)$$

Keterangan :

SSB : sum of squere between *cluster*

d(*i,j*) : Jarak *Euclidience distance* data ke i dan data ke-j

Langkah selanjutnya yaitu mencari Rasio dari hasil perhitungan SSW dan SSB seperti pada Persamaan (8)

$$R_{ij} = \frac{SSW_i + SSW_j}{SSB_{ij}} \quad (8)$$

Untuk proses perhitungan tahap akhir dalam mencari nilai DBI pada Persamaan (9)

$$DBI = \frac{1}{k}\sum_{i=1}^{k} \max i \neq j\,(R_{i,j}) \quad (9)$$

Keterangan :

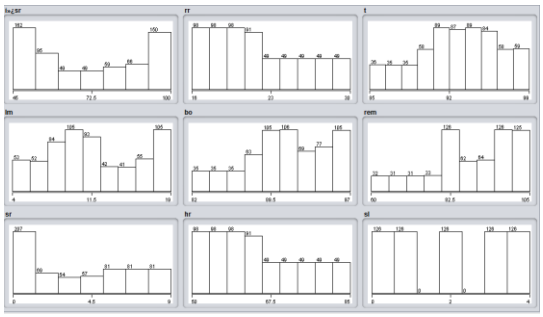
k : Jumlah *cluster* yang dihitung

Semakin kecil nilai DBI yang didapatkan yaitu nilai yang mendekati 0 atau sama dengan 0 berarti semakin baik *cluster* dari hasil pengelompokan.

3. HASIL DAN PEMBAHASAN

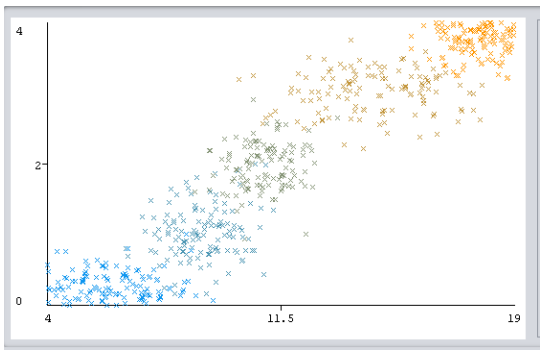
Algoritma *Pillar* dalam penelitian ini diharapkan dapat mengoptimalkan pemilihan *centroid* yang semula pengambilanya centoidnya acak dalam algoritma *k-means*. Sebelum ke tahap perhitungan *k-means* dilakukan terlebih dahulu *preprocessing* data set pada tabel 1 untuk memastikan data tidak terdapat *redudance* data maupun *noisy* yang dapat mengurangi keakuratan perhitungan. Dan untuk mengetahui seberapa padat pengelompokan data set dapat dilihat menggunakan weka.

1. Visualisasi Data Set pada Weka



Gambar 3. Grafik Visualisasi Data set

Melalui grafik pada Gambar 3, dapat dilihat nilai minimum dari setiap fitur salah satunya fitur hr yaitu 50 dan nilai data maksimum 85 serta nilai Mean yaitu 64,5.



Gambar 4. Grafik Plot Persebaran Data

Persebaran data pada Gambar 4 plot warna biru merupakan class ke-0 dan biru kehijauan class ke-1 untuk kelas ke-2 berwarna hijau selanjutnya warna hijau agak orange class ke-3 selanjutnya class ke-4 berwarna orange. Setelah dicek atribut menggunakan aplikasi Weka tidak terdapat missing, dan nilai unik nya 99% dapat dilihat pada Gambar 5.

Name: hr
Missing: 0 (0%) Distinct: 626 Type: Numeric
Unique: 622 (99%)

Statistic	Value
Minimum	50
Maximum	85
Mean	64.5
StdDev	9.915

Gambar 5. Pengecekan Atribut

2. Perhitungan Algoritma *Pillar*

Sebelum melakukan perhitungan *k-means* dilakukan perhitunga *Pillar* terlebih dahulu yaitu mencari jarak terjauh dengan rumus *euclidience distance* dari nilai titik tengah data. Langkah pertama untuk menghitung *Pillar* dengan mencari rata-rata nilai tengah dari keseluruhan data yang ada menggunakan Persamaan (10).

$$AVG = \frac{\sum x_i}{n_{x_i}} \quad (10)$$

Keterangan :

$\sum x_i$: Jumlah Keseluruhan nilai pada fitur ke-i

n_{x_i} : Jumlah Data

penerapan pada data set untuk mencari nilai rata-rata data pada fitur sr pada Persamaan (11).

$$AVG = \frac{45108}{630} = 71{,}6 \quad (11)$$

Untuk nilai m hasil perhitungan pada fitur lainya dapat dilihat ada Tabel 2.

Tabel 2. Nilai m pada setiap fitur

sr	**rr**	**t**	**lm**	**bo**	**rem**	**sr2**	**hr**
m data(rata rata data *cluster*)							
71,6	21,8	92,8	11,7	90,9	88,5	3,7	65

Setelah diketahui nilai rata-rata titik pusat setiap fitur langkah selanjutnya dilakukan perhitungan jarak *euclidience distance* setiap data terhadap m dan diperoleh *dmax* keseluruhan data bernilai 43,138. Dan jika Sebelum memilih jarak maksimum setiap *centroid* harus mencari terlebih dahulu nilai batas lingkungan (*nbdis)* dan nilai minimum *(nmin)* dan perhitungan dapat dilihat pada Persamaan (12) dan (13) dengan diketahui k bernilai 5 dan α yaitu 0,1 sedangkan nilai β yaitu 0,9.

$$nmin = 0{,}1\left(\frac{630}{5}\right) = 12{,}6 \quad (12)$$

$$nbdis = 0{,}9(43{,}138) = 38{,}8242 \quad (13)$$

Dalam memilih *centroid* awal yang optimal nilai D ≥ *nmin* dan D ≤ *nbdis* hasil pemilihan *centroid* awal dapat dilihat pada Tabel 3.

Tabel 3. *Centroid* awal optimal

id pasien	Class	*dmax*
Id 336	C0	38,695044
Id 228	C1	25,720226
Id 35	C2	13,324038
Id 29	C3	28,681582
Id 506	C4	38,775678

3. Perhitungan algoritma *K-means* ditambah algoritma *Pillar*

Setelah mendapatkan hasil *centroid* dari perhitungan algoritma *Pillar* dengan hasil *centroid* awal kluster 0 diambil data dengan id 336, kluster 1 pada data dengan id 228, kluster 2 data id 35, kluster 3 id 29 dan kluster 4 data dengan id 506 pada Tabel 4.

Tabel 4. Nilai *Centroid* Awal

Id Pasien	sr	rr	t	lm	bo
	Fitur				
Id 1	93,8	25,7	91,8	16,6	89,84
Id 2	91,64	25,1	91,6	15,9	89,55
Id 3	60	20	96	10	95
...					
Id 630	73,92	21,4	93,4	11,4	91,39
Id Pasien		**rem**	**sr2**	**hr**	**sl**
		Fitur			Class
Id 1		99,6	1,8	74	3
Id 2		98,9	1,6	73	3
Id 3		85	7	60	1
...					
Id 630		92	4,1	63	2

Setelah itu dilakukan perhitungan *k-means* menggunakan Persamaan (5) dan untuk hasil perhitungan jarak *euclidience distance* dapat dilihat pada Tabel 5.

Tabel 5. Hasil perhitungan jarak *Euclidience distance* Iterasi ke-1

Id Pasien	Jarak Antar *Centroid*				
	0	**1**	**2**	**3**	**4**
Id 1	69,7	53,1	40,4	1,28	14,8
Id 2	67,1	50,4	37,7	4,18	17,3
Id 3	31,7	13,2	8,12	42,4	54,2
...					
Id 630	46,4	28,5	16,3	26,2	38,5
Id Pasien				**Terdekat**	**Class Diikuti**
Id 1				1,284791	3
Id 2				4,175571	3
Id 3				8,124038	2
...					
Id 630				16,32265	2

Pada Tabel 5, diperoleh dari hasil perhitungan Persamaan (12) dengan data pada Id 1 diperoleh hasil pengelompokan kelas jarak terdekat diperoleh pada *cluster* 3.

$$d_0(x,y) = \sqrt{(45-93{,}80)^2 + (16-25{,}68)^2}$$
$$\sqrt{(96-91{,}84)^2 + (4-16{,}60)^2}$$
$$\sqrt{(95-89{,}84)^2 + (60-99{,}60)^2}$$
$$\sqrt{(7-1.84)^2 + (50-74{,}20)^2}$$
$$= 69{,}7 \qquad (12)$$

Untuk mengetahui jarak terdekat data terhadap setiap *cluster* dapat dilihat pada perhitungan Persamaan 12 yaitu mencari jarak *euclidience distance* pada *cluster* 0 diperoleh nilai 69,7 dan untuk hasil pada *cluster* 1, 2, 3, 4, dapat dilihat pada Tabel 5. Dan untuk mengetahui *cluster* mana yang diikuti dari setiap perhitungan jarak yaitu pilih jarak minimum dari setiap hasil perhitungan jarak *euclidience distance* setiap *cluster*. Dan untuk melanjutkan Iterasi hingga iterasi ke-(n-1) dan iterasi ke-n hasilnya sama dicari nilai *centroid* baru dari hasil perhitungan iterasi sebelumnya dan untuk rumus perhitunganya dapat dilihat pada Persamaan (13) (Suntoko, 2019), (Sari, 2021).

$$\mu_k = \frac{1}{N_k}\sum_{i=1}^{N_k} x_i \qquad (13)$$

Keterangan :

μ_k : Titik *centroid* dari *cluster* ke-k

N_k : Banyaknya data pada *cluster* ke-k

x_i : Data ke-i pada *cluster* ke-k

Tabel 6. *Centroid* baru pada Iterasi ke-17

C	sr	rr	t	lm	bo	rem	sr2	hr
	Fitur							
0	47,2	16,9	97,3	5,76	95,9	68,8	7,88	52,2
1	55,03	19	95,1	8,98	93,5	82,4	5,932	57,5
2	71,12	21,1	93,1	11,1	91,1	90,6	3,668	62,8
3	87,14	23,9	91	14,4	89	97,4	0,952	69,8
4	97,85	27,9	87,7	18	85,2	102	0,089	79,8

Pada Tabel 6 merupakan *centroid* akhir dari hasil *cluster*isasi akhir pada iterasi ke-17.

Tabel 7. Hasil Perhitungan *K-means* Akhir Iterasi ke-17

Id Pasien	Jarak Antar *Centroid*				
	0	1	2	3	4
Id 1	62,467	47,129	28,001	8,9132	10,166
Id 2	59,882	44,461	25,257	6,0225	12,153
Id 3	22,762	6,6198	14,118	33,439	49,631
...					
Id 630	38,466	22,496	3,2833	16,75	33,44

Id Pasien	Terdekat	Class Diikuti
Id 1	8,913	3
Id 2	6,022	3
Id 3	6,62	1
...		
Id 630	3,283	2

Dari hasil perhitungan iterasi akhir pada Tabel 7 didapat hasil pengelompokan *cluster*isasi data set tingkat stress dapat dilihat pada Tabel 8.

Tabel 8. Hasil Perhitungan *Cluster*isasi optimasi *k-means*

Jumlah	C0	C1	C2	C3	C4
	111	155	112	120	132

4. Hasil Perhitungan *K-means* Tanpa *Pillar*

Untuk perhitungan *k-means* tanpa *Pillar* pengambilan *centroid* awal diambil acak dapat dilihat pada Tabel (9).

Tabel 9. Pengambilan *Centroid* Awal Acak

C	Id Pasien	sr	rr	t	lm
		Fitur			
0	Id 9	45,3	16,1	96,2	4,22
1	Id 26	53,7	18,7	94,7	8,74
2	Id 41	61	20,1	92,1	10,1
3	Id 62	81,1	22,3	90,1	12,4
4	Id 130	97,6	27,6	87	17,8
C	**Id Pasien**	**bo**	**rem**	**sr2**	**hr**
		Fitur			
0	Id 9	95,1	61,12	7,1	50,3
1	Id 26	93,1	81,84	5,7	56,8
2	Id 41	90,1	85,48	2,1	60,2
3	Id 62	88,1	95,36	0,1	65,7
4	Id 130	84,4	102	0	78,9

Setelah pengambilan *centroid* awal dilakukan perhitungan jarak dan mencari nilai *centroid* baru hingga menemukan hasil akhir pengklusteran dan didapat hingga iterasi ke-23 dengan nilai *centroid* baru pada Tabel 10.

Tabel 10. *Centroid* baru iterasi ke-23

C	sr	rr	t	lm	bo	rem	sr2	hr
	Fitur							
0	47,18	16,88	97,3	5,76	95,88	68,8	7,88	52,2
1	54,953	18,99	95,1	8,98	93,52	82,4	5,93	57,5
2	71,04	21,11	93,1	11,1	91,11	90,6	3,67	62,8
3	87,14	23,9	91	14,4	88,95	97,4	0,95	69,8
4	97,85	27,91	87,7	18	85,23	102	0,09	79,8

Dan untuk hasil pengklusteran jumlah nya sama dengan Tabel 7 karena hasil sama oleh karena itu perlu dilakukan evaluasi dengan acuan hasil *centroid* awal yaitu dengan Davies Bouldin Index(DBI).

5. Evaluasi DBI

Untuk mencari hasil evaluasi langkah pertama mencari nilai SSW dengan rumus pada Persamaan (6) dan perhitungan untuk data id 1 dengan claster 0 dapat dilihat pada Persamaan (14)

$$SSW_0 = \sqrt{(48{,}12 - 47{,}20)^2 +}$$

$$\sqrt{(17{,}248 - 16{,}88)^2 + (97{,}872 - 97{,}32)^2}$$

$$\sqrt{(6{,}496 - 5{,}76)^2 + (96{,}248 - 95{,}88)^2}$$

$$\sqrt{(72{,}48 - 68{,}80)^2 + (8{,}248 - 7{,}88)^2}$$

$$\sqrt{(3{,}12 - 52{,}20)^2}$$

$$= 4{,}065104 \qquad (14)$$

Dan hasil penjumlahan nilai jarak *euclidience distance cluster* 0 yaitu 543,7873 dengan jumlah data *cluster* 0 yaitu 111 sehingga SSW C0 bernilai 4,898715358 untuk nilai SSW *cluster* lain dapat dilihat pada Tabel 11.

Tabel 11. Nilai SSW

SSW C0	SSW C1	SSW C2	SSW C3	SSW C4
4,898 71536	4,4745 73097	5,2532 65651	4,8179 66376	4,0646 87911

Setelah diketahui nilai SSW dilanjutkan mencari nilai SSB dengan rumus pada Persamaan (7) dan hasil perhitungan SSB pada Tabel 12.

Tabel 12. Nilai SSB

SSB	0	1	2	3	4
0	0	17,43	35,53	54,59	70,6
1	17,43	0	19,42	39,01	55,2
2	35,53	19,42	0	19,67	36,2
3	54,59	39,01	19,67	0	17,1
4	70,64	55,19	36,23	17,14	0

Selanjutnya mencari nilai Rasio dengan rumus pada Persamaan (8) untuk hasil nilai Rasio dapat dilihat pada Tabel 13.

Tabel 13. Nilai Rasio

R	0	1	2	3	4
0	0	0,537 7	0,285 7	0,178	0,12 7
1	0,537 7	0	0,501	0,238 2	0,15 5
2	0,285 7	0,501	0	0,511 9	0,25 7
3	0,178	0,238 2	0,511 9	0	0,51 8
4	0,126 9	0,154 7	0,257 2	0,518 3	0

Langkah selanjutnya mencari nilai Rasio Maksimum dan diperoleh R maksimum pada Tabel 14.

Tabel 14. Rasio Maksimum

R	R max
0	0,5376695
1	0,5376695
2	0,5119124
3	0,5182752
4	0,5182752

Setelah mengetahui nilai SSW, SSB dan Rasio dapat dihitung evaluasi DBI nya perhitngan DBI dapat dilihat pada Persamaan (15).

$$DBI = \frac{1}{5}\,(0{,}5376695 + 0{,}5376695 +$$

$$0{,}5119124 + 0{,}5182752 + 0{,}5182752)$$

$$= \frac{1}{5}(2{,}62380178)$$

$$= 0{,}52476036 \qquad (15)$$

Sehingga diperoleh hasil evaluasi DBI algoritma *k-means* ditambah algoritma *Pillar* pada Persamaan (15).

6. Perbandingan Hasil Perhitungan

Tabel 15. Perbandingan hasil perhitungan *k-means* dan *k-means* dengan algoritma *Pillar*

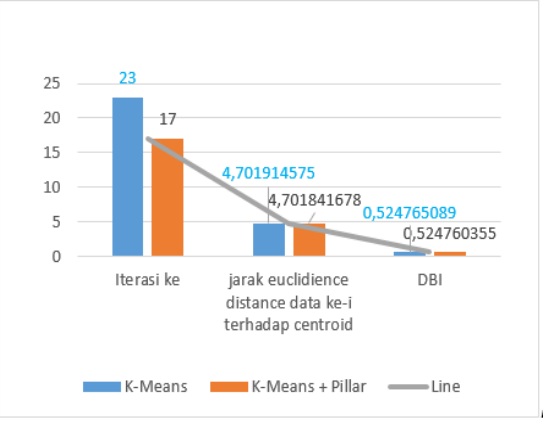
Metode	$\bar{x}$ *Euclidience distance* data ke-i terhadap *centroid*	Jumlah Iterasi	DBI
K-means	4,701914575	23	0,524765089

K-means + Algoritma *Pillar*	4,701841678	17	0,524760355

Dari Tabel 15 dapat diketahui bahwa dengan menggunakan algoritma *Pillar* dapat meningkatkan nilai evaluasi DBI sebesar 0,00000473 perhitungan *k-means* serta hasil akhir rata-rata *centroid* lebih kecil dibandingkan dengan menggunakan rumus *k-means* saja. Dan mempercepat perhitungan yang semula hingga pada iterasi ke-23 menjadi iterasi ke-17 sudah dapat menemukan hasil akhir pengelompokan.

4. SIMPULAN

Kesimpulan dari hasil penelitian yaitu pencarian nilai *centroid* yang optimal dengan algoritma *Pillar* yaitu jarak *euclidience distance* maksimal setiap *centroid* harus kurang dari sama dengan nilai batas lingkungan (*nbdis)* dan lebih dari sama dengan nilai maksimal*(nmin)* sehingga diperoleh *centroid* awal optimal pada id 336, 228, 35, 29, 506.



Gambar 5. Grafik hasil optimalisasi K-Means dengan Algoritma Pillar

Pada Gambar 5 dapat disimpulkan Hasil dari perhitungan *k-means* ditambah algoritma *Pillar* dapat meningkatkan nilai evaluasi pengklusteran dengan metode DBI dari semula nilai evaluasi DBI *k-means* 0,524765089 dan dengan *Pillar* evaluasi meningkat sebesar 0,00000473 dan menghemat waktu perhitungan dilihat dengan berkurangnya iterasi yang semula acak 23 iterasi setelah menggunakan *Pillar* berkurang menjadi 17 iterasi.

5. UCAPAN TERIMAKASIH

Terimakasih Kepada Program Studi Ilmu Komputer Fakultas Ilmu Eksakta Universitas Nahdlatul Ulama Blitar yang telah mendukung penelitian ini.

DAFTAR PUSTAKA

A L Ramdani & H B Firmansyah. (2019). Pillar K-Means Clustering Algorithm Using MapReduce Framework. *IOP Conference Series: Earth and Environmental Science*. https://doi.org/10.1088/1755-1315/258/1/012031

Ariasa, K., Gunadi, I. G. A., & Candiasa, I. M. (2020). Optimasi Algoritma Klaster Dinamis Pada K-Means Dalam (Studi Kasus : Universitas Pendidikan Ganesha). *Jurnal Nasional Pendidikan Teknik Informatika : JANAPATI* |, *9*, 181–193.

Baharuddin, M. M., Azis, H., & Hasanuddin, T. (2019). Analisis Performa Metode K-Nearest Neighbor Untuk Identifikasi Jenis Kaca. *ILKOM Jurnal Ilmiah*, *11*(3), 269–274. https://doi.org/10.33096/ilkom.v11i3.489.269-274

Dinata, R. K., Novriando, H., Hasdyna, N., & Retno, S. (2020). Reduksi Atribut Menggunakan Information Gain untuk Optimasi Cluster Algoritma K-Means. *Jurnal Edukasi Dan Penelitian Informatika*, *6*(1), 48–53. https://scholar.google.co.id/scholar?hl=id&as_sdt=0%2C5&q=optimasi+centroid+evaluasi+DBI&btnG=

Ketut Agus Seputra, I. N. S. W. W. (2020). Penerapan Algoritma Pillar Untuk Inisialisasi Titik Pusat K- Means Klaster Dinamis. *Jurnal Teknologi Informasi Dan Ilmu Komputer (JTIIK)*, *7*(6), 1213–1220. https://doi.org/10.25126/jtiik.202072538

Kuntjoro, D. A., Setiawan, B. D., & Perdana, R. S. (2018). Algoritme Genetika Untuk Optimasi K-Means Clustering Dalam

Pengelompokan Data Tsunami. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, *2*(10), 3865–3872. http://j-ptiik.ub.ac.id

Primandana, A., Adinugroho, S., & Dewi, C. (2019). Optimasi Penentuan Centroid pada Algoritme K-Means Menggunakan Algoritme Pillar (Studi Kasus : Penyandang Masalah Kesejahteraan Sosial di Provinsi Jawa Timur). *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, *3*(11), 10678–10683. http://j-ptiik.ub.ac.id

Rachakonda, L., Member, S., Bapatla, A. K., Member, S., & Mohanty, S. P. (2020). SaYoPillow : Blockchain-Integrated Privacy-Assured IoMT Framework for Stress Management Considering Sleeping Habits. *Kaagle*, 1–10. https://www.kaggle.com/laavanya/human-stress-detection-in-and-through-sleep/version/2?select=IEEE-TCE_2020-08-0175_SaYoPillow.pdf

Rahayu, A. E., Hikmah, K., Ningsih, N. Y., & Fauzan, A. C. (2019). Penerapan K-Means Clustering Untuk Penentuan Klasterisasi Beasiswa Bidikmisi Mahasiswa. *IILKOMNIKA*, *1*(2), 82–86.

Ramadhan, M. R., Fauzan, A. C., Aziz, N., & Wahyuni, T. (2020). K - Means Clustering for Determining Quality of Outdoor Temperature Based on BMKG Datasets. *Journal of Development Research*, *4*(May), 12–17. http://journal.unublitar.ac.id/jdr

Rustam, S. (2020). Penerapan Optimasi Jumlah Kluster Pada Kmeans Untuk Pengelompokan Kelas Mata Kuliah Kosentrasi Mahasiswa Semester Akhir. *Jurnal Sistem Informasi Dan Teknik Komputer*, *5*(1), 1–4.

Sari, D. P. (2021). Pengelompokkan Penyakit Berdasarkan Lingkungan Dengan Algoritma K-Means Pada Puskesmas Sungai Tarab 2. *JOISIE Journal Of Information System And Informatics Engineering*, *5*(2), 75–81. https://doi.org/https://doi.org/10.35145/joisie.v5i2.1700

Suntoko, J. (2019). *Data Mining Algoritma dan Implementasi dengan Pemrograman PHP* (kedua). PT Elex Media Komputindo.

Suyanto, D. (2019). Data Mining Untuk Klasifikasi dan Klasterisasi Data. In *teknik informatika* (Edisi Revi, p. 414). Informatika Bandung. www.biobses.com

Syahputra, S., Ramadani, S., Manaor, A., & Pardede, H. (2020). Menentukan Strategi Promosi Menggunakan Algoritma Clustering K-Means. *JOISIE Journal Of Information System And Informatics Engineering*, *4*(1), 7–14. https://doi.org/https://doi.org/10.35145/joisie.v4i1.510

Tommy, tommy. (2021). *Tahap Preprocessing Data Mining*. Wesite. https://id.linkedin.com/pulse/tahap-preprocessing-data-mining-tommy-