PRISMA 2 (2019): 161-170
PRISMA, Prosiding Seminar Nasional Matematika
https://journal.unnes.ac.id/sju/index.php/prisma/
ISSN 2613-9189

Kajian Indeks Validitas pada Algoritma *K-Means Enhanced* dan *K-Means* MMCA

A F Khairati[a], A A Adlina[a], G F Hertono[a], B D Handari[a,*]

[a]*Departemen Matematika FMIPA Universitas Indonesia, Depok*

**Alamat Surel: bevina@sci.ui.ac.id*

Abstrak

Algoritma *K-Means*merupakan salah satu metode yang banyak digunakan dalam penyelesaian masalah *clustering* seperti masalah pengenalan pola, partisi dan pengelompokkan taksonomi pada tumbuhan. Algoritma*K-Means* memiliki ketergantungan terhadap pemilihan titik pusat awal klaster yang dilakukan secara acak. Hal ini dapat mempengaruhi hasil *clustering* karena adanya perubahan titik pusat awal klaster pada tiap simulasi. Metode *Enhanced*dan *Maximum Minimum CriterionAlgorithm* merupakan dua metode yang dapat diterapkan pada algoritma *K-Means*dalam pemilihan titik pusat awal klaster. Penerapan kedua metode tersebut pada algoritma *K-Means* menghasilkan hasil *clustering* yanglebih optimal. Hal tersebut ditunjukkan dengan jumlah iterasi yang sama pada tiap simulasi dalam mencapai kriteria konvergen dan nilai rata-rata similaritas terhadap data *benchmark* yang lebih baik. Selain itu, kesulitan algoritma *K-Means* adalah dalam menentukan jumlah klaster optimal suatu himpunan data.Indeks validitas merupakan metode yang dapat digunakan untuk menentukan hasil *clustering*dengan jumlah klaster optimal pada himpunan data. Pada makalah ini, dilakukan *clustering* menggunakan algoritma *K-Means*, *K-Means Enhanced* dan *K-Means Maximum Minimum Criterion Algorithm*. Selanjutnya, masing-masing hasil *clustering* tersebut dievaluasi oleh empat jenis indeks validitas, yaitu indeks *Silhouette*, *Davies-Bouldin*, *Dunn*, dan *Calinski-Harabasz*. Implementasi tersebut dilakukan padahimpunan data *benchmark* yang sudah diketahui jumlah klaster optimalnya yaitu himpunan data Iris, Ruspini, Seeds, dan Wine. Hasil implementasi dibandingkan untuk mengetahuiapakah keempat indeks validitas dapat memprediksi jumlah klaster dengan tepat. Dari hasil simulasi, indeks *Silhouette*, *Davies-Bouldin*, dan *Calinski-Harabasz* dapat memprediksi jumlah klaster optimal lebih baik dibandingkan dengan *Dunn*.

Kata kunci:

Clustering, *K-Means*, Indeks Validitas, Metode *Enhanced*, *Maximum Minimum Criterion Algorithm*.

1. Pendahuluan

Metode *clustering* merupakan metodeyang dapat digunakan untukpengelompokkan himpunan data menjadi beberapa klaster sehingga data dalam satu klaster memiliki karakteristik yang sama, sedangkan data dalam kelompok yang berbeda memiliki karakteristik yang berbeda (Nanda, Mahanty, & Tiwari, 2010).Algoritma *K-Means*termasuk dalam *partitional clustering*, yang merupakan suatu cara mempartisi himpunan data menjadi beberapa kelompok yang saling lepas dengan jumlah klaster yang sudah ditetapkan di awal (Nazeer & Sebastian, 2009).Beberapa kelebihan dari algoritma *K-Means*diantaranya adalah mudah diimplementasikan, memiliki tingkat konvergensi yang tinggi, dan menghasilkan klaster yang lebih padat jika dibandingkan dengan metode hirarki. Sedangkan, beberapa kekurangannya diantaranya adalah sensitif terhadap pemilihan titik pusat awal klaster dan sulit dalam menentukan jumlah klaster terbaik (Lasheng dan Yuqiang, 2017,Bakshi, Derakhshi dan Zafarani, 2012).

Banyak penelitian yang telah dilakukan untuk menyelesaikan masalah pemilihan titik pusat awal klaster, diantaranya adalah metode *Enhanced* yang diajukan oleh Yedla, Pathakota, & Srinivasa (2010) dan metode *Maximum Minimum Criterion Algorithm*(MMCA) oleh Lasheng & Yuqiang (2017).

To cite this article:
Khairati, A.F., Adlina, A.A, Hertono, G.F, & Handari, B.D. (2019). Kajian Indeks Validitas pada Algoritma *K-Means Enhanced* dan *K-Means* MMCA. *PRISMA, Prosiding Seminar Nasional Matematika 2*, 161-170

Selain masalah pemilihan titik pusat awal, banyak penelitian yang dilakukan untuk menyelesaikan masalah penentuan jumlah klaster optimummenggunakan metode indeks validitas. Terdapat tiga jenis kriteria indeks validitas, yaitu kriteria eksternal dan kriteria internal yang menggunakan uji statistik, dan kriteria relatif yang menggunakan kriteria yang spesifik. Kriteria relatif lebih sesuai digunakan untuk metode *crisp clustering* dan *fuzzy clustering*. Berdasarkan Baarsch, J., & Celebi, M. E. (2012), contoh indeks validitas dengan kriteria relatif adalahindeks *Dunn*, *Silhouette*, *Davies-Bouldin*, *Calinski-Harabasz*, *Point Bi Seral*, *Sum of Square*, dan *PBM*. Pada makalah ini dibahas empat jenis indeks validitas, yaitu *Dunn*, *Silhouette*, *Davies-Bouldin*, dan *Calinski-Harabasz*.

Pada makalah ini, keempat indeks tersebut diimplementasikan pada himpunan data *benchmark*, yaitu himpunan data Iris, Ruspini, Seeds, dan Wine,yang telah melalui proses *clustering* menggunakan algoritma *K-Means*, *K-MeansEnhanced*, dan *K-Means* MMCA. Setelah itu, hasil implementasi indeks validitastersebut dibandingkan untuk mengetahui indeks validitas yang lebih baik dalam menentukan jumlah klaster optimal pada himpunan data *benchmark*.

2. Metode Penelitian

Berikut dijelaskan teori-teori dasar yang dibutuhkan dalam makalah ini, yaitu algoritma *K-Means*, *K-MeansEnhanced*, dan *K-Means*MMCA. Sertaempat indeks validitas,yaitu *Dunn*, *Silhouette*, *Davies-Bouldin*, *Calinski-Harabasz*.

2.1. K-Means Clustering

Misal diberikan himpunan data berisi data yang akan diklaster menjadi klaster. Tujuan dari algoritma*K-Means* adalah meminimumkan jumlah *squared error* dari seluruh klaster(Jain, 2010).Misalkan adalah data ke- , adalah nilai rata-rata klaster yang didapatkan dengan cara menghitung rata-rata jarak antara data terhadap titik pusat klaster menggunakan jarak *Euclidean* dengan $k = 1,.,$ dan adalah himpunan klaster-klaster pada himpunan data . Sehingga jumlah *squared error*seluruh klaster pada himpunan data adalah

$$J(C) = \sum_{k=1}^{K} \sum_{x_i \in c_k} \|x_i - \mu_k\|^2.$$

Berikut adalah algoritma *K-Means*(Jain, Murty, & Flynn, 1999) :

Input: = Himpunan data, = Jumlah klaster

Output: Himpunan klaster

Langkah-langkah:

1. Tentukan titik pusat awal, dengan adalah jumlah klaster yang diinginkan
2. Tentukan titik pusat terdekat dari masing-masingdata, dan masukkan data ke klaster terdekat
3. Tentukan ulang letak titik pusat berdasarkan posisi anggota klaster yang terbentuk.

Jika belum memenuhi kriteria konvergen, ulangi langkah 2 dan 3.

Kriteria konvergen adalah ketikapada sebuah iterasi tidak ada perubahan anggota pada tiap klaster dibandingkan iterasi sebelumnya,atau nilai jumlah *squared error* telah mencapai nilai minimum(Jain, Murty, & Flynn, 1999). Tidak ada perubahan anggota pada tiap klaster ekuivalen dengan tidak adanya perubahan titik pusat klaster(Yedla, Pathakota, & Srinivasa, 2010).

2.2. K-Means Enhanced

Dalam menyelesaikan masalah pemilihan titik pusat awal klaster, metode *Enhanced*menentukan titik pusat awal klaster dengan cara menghitung jarak antara data terhadap titik asal (diasumsikan sebagai titik O dengan koordinat (0,0) di R^2. Jumlah atribut tanpa atribut kelas pada himpunan data menyatakan dimensi dari suatu himpunan data.

Berikut adalah algoritma *K-Means Enhanced* (Yedla, Pathakota, & Srinivasa, 2010):
Input: = Himpunan data, = Jumlah klaster
Output: Himpunan data yang terdiri dari beberapa klaster
Langkah-langkah:

1. Misalkan diberikan sebuah himpunan data D yang berisi $n$data d_i, dengan sejumlah atribut $i = 1, \dots, n$ dan n = jumlah data pada himpunan data
2. Periksa nilai atribut-atribut pada himpunan data. Jika terdapat nilai atribut negatif, lakukan transformasi dengan cara menentukan nilai atribut terkecil, kemudian semua atribut dikurangi dengan nilai atribut terkecil tersebut
3. Hitung jarak semua data terhadap titik asal
4. Lakukan pengurutan jarak dari urutan terkecil hingga terbesar
5. Partisi data ke dalam $[K]$klaster
6. Pada setiap klastertentukan titik tengahdengan menghitung nilai rata-rata atribut seluruh data pada setiap klaster,yang menjadi titik pusat awal klaster (c_j)
7. Hitung jarak semua data d_i terhadap semua titik pusat awal klaster(c_j)
8. Untuk semua data d_i, tentukan titik pusat awal klaster(c_j) terdekat dan masukkan d_i ke dalam klaster$j$
9. Simpan ClusterId$[i]$ sebagai klaster$j$ dimana data d_i saat ini berada dan NearestDist$[i]$ sebagai jarak data d_i dengan titik pusat klaster$j$saat ini
10. Hitung kembali titik pusat yang baru dalam klaster *j* dengan menghitung rata-rata atribut data d_i dalam klaster *j*
11. Hitung NewDist$[i]$ yang merupakan jarak antara seluruh data d_i dalam klaster *j* dengan titik pusat yang baru
12. Jika jarak yang baru lebih kecil atau sama dengan data yang terkait di NearestDist$[i]$ maka data d_i tetap pada klastersebelumnya
13. Jika jarak yang baru lebih besar dari data yang terkait dengan NearestDist$[i]$ maka untuk data $d_i$akan dihitung jaraknya terhadap seluruh titik pusat klaster$c_j$lainnya. Lalu, tentukan kembali titik pusat klasterc_j terdekat dan masukkan d_i ke dalam klaster j

Ulangi langkah 6-13 hingga kriteria konvergen terpenuhi.

2.3. K-Means MMCA

Pada metode MMCAdilakukan perhitungan jarak antar 2 data dalam himpunan data dan jarak antara himpunan data terhadap himpuan data lainnya. Definisi jarak antara 2 data, jarak minimum dari data ke himpunan data dan jarak maksimum 2 himpunan data, dapat dilihat pada (Lasheng & Yuqiang, 2017).

Berikut penerapanMMCA(Lasheng & Yuqiang, 2017):
Input: = Himpunan data, = Jumlah klaster, = Himpunan kosong
Output: himpunan data yang berisi data yang menjadi titik pusat klaster
Langkah-langkah:

1. Misalkan diberikan sebuah himpunan data yang berisi data (dengan sejumlah atribut.
2. Hitung jarak antar data dalam himpunan data
3. Kemudian tentukan dua data yang memiliki jarak paling besar. Masukkan kedua data tersebut ke dalam himpunan kosong
4. Jika jumlah anggota dalam masih kurang dari , maka dicari satu data pada himpunan

5. Hitung jarak seluruh data dalam himpunan data *D* terhadap himpunan . Sehingga didapatkan satu data yang memiliki jarak paling jauh terhadap himpunan kemudian masukkan data tersebut ke dalam himpunan

Ulangi langkah 4 dan 5 hingga jumlah anggota himpunan sama dengan nilai . Himpunan berisi data-data yang merupakan titik pusat awal klaster.

Hasil implementasi MMCAdigunakan sebagai *input* dalam algoritma *K-Means*.

Berikut langkah-langkah*K-Means* MMCA(Lasheng & Yuqiang, 2017):
Input: = Himpunan data, = Jumlah klaster, = Himpunan data hasil dari MMCA
Output: Himpunan data yang terdiri dari beberapa klaster
Langkah-langkah:

1. Hitung jarak data dalam himpunan data terhadap semua anggota himpunan
2. Tentukan titik pusat awal klaster terdekat dan masukkan data ke dalam klaster *j*
3. Tentukan titik pusat baru dalam klaster dengan meghitung nilai rata-rata dari atribut setiap data dalam klaster .

Jika terjadi perubahan titik pusat, ulangi langkah 2 dan 3 hingga tidak ada perubahan pada titik pusat klaster.

2.4. Uji Similaritas

Setelah melakukan *clustering* menggunakan *K-Means*, *K-MeansEnhanced*dan *K-means*MMCA, kemudian dilakukan uji nilai similaritas dari hasil klaster yang dihasilkan pada setiap himpunan data terhadap himpunan data terkait yang telah diklasifikasi sebelumnya.Uji similaritas dilakukan untuk melihat kesamaan hasil *clustering* menggunakan 3 metode di atas terhadap hasil *clustering* pada himpunan data *benchmark* terkait. Misalkan r_i adalah kelas data ke-i pada hasil klaster yang dihasilkan dan s_i adalah kelas data ke-i pada klasteryang sudah ada dan n adalah jumlah data pada himpunan data, maka similaritasnya didefinisikan sebagai berikut:

$$S = \sum_{i=1}^{n} \frac{\delta(r_i, s_i)}{n}, \qquad \delta(r_i, s_i) = \begin{cases} 1, r_i = s_i \\ 0, r_i = s_i \end{cases}.$$

2.5. Indeks Validitas

Untuk mengatasi kesulitan algoritma *K-Means* dalam menentukan jumlah klaster yang tepat berdasarkan data yang digunakan, maka digunakan indeks validitas yang merupakan metode untuk mengevaluasi hasil algoritma *clustering* dengan tujuan mendapatkan jumlah klaster terbaik. Menurut Liu, Li, Xiong, Gao, & Wu (2010), indeks validitas dihitung berdasarkan dua hal, yaitu:

1. *Compactness*: merupakan tingkat similaritas objek dalam klaster yang sama.
2. *Separation*: merupakan tingkat perbedaan objek dalam klaster yang berbeda.

Menurut Halkidi, Batistakis, & Vazirgiannis (2001), terdapat tiga jenis kriteria indeks validitas, yaitu:

1. Kriteria Eksternal: Mengevaluasi hasil algoritma *clustering* berdasarkan struktur yang sudah ditentukan.
2. Kriteria Internal: Mengevaluasi hasil algoritma *clustering* secara kuantitatif yang mencakup vektor dari himpunan data tersebut.
3. Kriteria Relatif: Mengevaluasi hasil algoritma *clustering* dengan membandingkan algoritma tersebut dengan skema *clustering* lainnya.

Pada penelitian ini akan digunakan empat indeks validitas dengan kriteria relatif, yaitu indeks *Silhouette*, indeks*Davies-Bouldin*, indeks*Dunn*, dan indeks*Calinski-Harabasz*.

2.5.1. Indeks Silhouette

Secara umum, indeks validitas *Silhouette* menghitung rata-rata nilai setiap titik pada himpunan data. Lebih spesifik, perhitungan nilai setiap titik adalah selisih nilai *separation* dan *compactness* yang dibagi dengan maksimum antara keduanya. Jumlah klaster yang terbaik ditunjukkan dengan nilai *Silhouette* yang semakin mendekati 1 (Rosseeuw, 1987). Misalkan terdapat N buah titik pada suatu himpunan data, terdapat pula di dalamnya klaster p dan klaster q dengan x_i adalah titik pada klaster p dan y_j adalah titik pada klaster q, sehingga $a_{p,i}$ adalah rata-rata jarak titik x_i ke setiap titik pada klaster p, dan $d_{q,i}$ adalah rata-rata jarak titik x_i ke setiap titik pada klaster q. Maka rumus perhitungan indeks validitas *Silhouette*dapat dilihat pada Tabel 1.

2.5.2. Indeks Davies-Bouldin

Indeks validitas *Davies-Bouldin* (DB) menghitung rata-rata nilai setiap titik pada himpunan data. Perhitungan nilai setiap titik adalah jumlah nilai *compactness* yang dibagi dengan jarak antara kedua titik pusat klaster sebagai *separation*. Jumlah klaster terbaik ditunjukkan dengan nilai DB yang semakin kecil (Davies & Bouldin, 1979).Misalkan terdapat suatu himpunan data dengan k buah klaster, terdapat n_p buah titik pada klaster p dan n_q buah titik pada klaster q dengan titik pusatnya masing-masing adalah c_p dan c_q, sehingga M_{pq} adalah jarak antara titik pusat klaster p dan klaster q, S_p dan S_q berturut-turut merupakan rata-rata jarak setiap titik pada klaster p dan q ke titik pusatnya pada klaster yang terkait, yaitu c_p dan c_q, dengan perhitungan indeks validitas DB dapat dilihat pada Tabel 1.

Tabel 1. Perhitungan indeks *Silhouette* dan indeks *Davies-Bouldin*

Indeks Silhouette	Indeks Davies-Bouldin
$SIL = \frac{1}{N}\sum_{i=0}^{N} s_{x_i},$ $s_{x_i} = \frac{(b_{q,i} - a_{p,i})}{\max\{a_{p,i}, b_{q,i}\}}, p \neq q,$ $b_{q,i} = \min d_{q,i} \; ; q = 1, \dots, k,$ $d_{q,i} = \frac{1}{n_q}\sum_{j=1}^{n_q} d(x_i, y_j),$ $a_{p,i} = \frac{1}{n_p}\sum_{k=1}^{n_p} d(x_i, x_k).$	$DB = \frac{1}{k}\sum_{p=1}^{k} R_p,$ $R_p = \max R_{p,q}, \; p \neq q,$ $R_{p,q} = \frac{(S_p + S_q)}{M_{pq}},$ $S_p = \frac{1}{n_p}\sum_{i=1}^{n_p} d(x_i, c_p),$ $S_q = \frac{1}{n_q}\sum_{j=1}^{n_q} d(y_j, c_q),$ $M_{pq} = d(c_p, c_q).$

2.5.3. Indeks Dunn

Indeks validitas *Dunn* (DN) menghitung nilai minimum dari perbandingan antara nilai fungsi dissimilaritas antara dua klaster sebagai *separation* dan nilai maksimum dari diameter klaster sebagai *compactness*. Jumlah klaster terbaik ditunjukkan dengan semakin besar nilai DN(Dunn, J.C., 1974).Misalkan terdapat suatu himpunan data dengan k buah klaster yang di dalamnya terdapat klaster p, klaster q, dan klaster r. Misal $x_i$adalah titik ke-i pada klaster p, $y_i$adalah titik ke-i pada klaster q, serta z_i dan $z_j$berturut-turut merupakan titik ke-$i$dan titik ke-$j$pada klaster r,sehingga perhitungan indeks validitas DNdapat dilihatpada Tabel 2.

2.5.4. Indeks Calinski-Harabasz

Indeks validitas *Calinski-Harabasz* (CH) menghitung perbandingan antara nilai *Sum of Square between cluster* (SSB) sebagai *separation* dan nilai *Sum of Square within-cluster*(SSW) sebagai *compactness* yang dikalikan dengan faktor normalisasi, yaitu selisih jumlah data dengan jumlah klaster dibagi dengan jumlah klaster dikurang satu. Jumlah klaster terbaik ditunjukkan dengan semakin besar nilai CH (Baarsch & Celebi, 2012).Misalkan terdapat suatu himpunan data dengan k buah klaster dan N buah titik data, misal C_l adalah klaster ke-l dengan x_i adalah titik ke- i pada klaster ke- l, N_l adalah jumlah titik pada klaster ke- l, dan $\bar{x}_l$ adalah titik pusat klasterke- l, maka perhitungan indeks validitas CH dapat dilihatpada Tabel 2.

Tabel 2. Perhitungan indeks *Dunn* dan indeks *Calinski-Harabasz*

Indeks Dunn	Indeks Calinski-Harabasz
$DN = \min_{p=1,\dots,k}\left\{\min_{q=i+1,\dots,k}\left(\frac{d(c_p,c_q)}{\max_{r=1,\dots,k} diam(c_r)}\right)\right\},$ $d(c_p,c_q) = \min_{x_i\in C_p,\, y_i\in C_q} d(x_i,y_i),$ $diam(c_r) = \max_{z_i,z_j\in C_r} d(z_i,z_j).$	$CH = \frac{trace(SSB)}{trace(SSW)} \times \frac{N-k}{k-1},$ $SSW = \sum_{l=1}^{k}\sum_{x_i\in C_l}(x_i-\bar{x}_l)(x_i-\bar{x}_l)^T,$ $SSB = \sum_{l=1}^{k} N_l(\bar{x}_l-\bar{x})(\bar{x}_l-\bar{x})^T,$ $\bar{x}_l = \frac{1}{N_l}\sum_{x_i\in C_l} x_i,$ $\bar{x} = \frac{1}{N}\sum_{i=1}^{N} x_i.$

3. Hasil dan Pembahasan

Pada bagian ini ditunjukkan hasil simulasi yang dilakukan terhadap 3 himpunan data yaitu Iris, Wine dan Ruspiniyang diperoleh dari penelitian Maitra dan Melnykov (2012) serta himpunan data Seeds yang diperoleh dari https://archive.ics.uci.edu/ml/index.php.Penjelasan mengenai masing-masing himpunan data dapat dilihat pada Tabel 3.

Tabel 3. Data *Benchmark*

Himpunan Data	Jumlah Data	Dimensi	Banyak Kelas
Ruspini	75	2	4
Iris	150	4	3
Seeds	210	7	3
Wine	178	13	3

Pertama dilakukan proses *clustering* pada himpunan data *benchmark* menggunakan algoritma *K-Means, K-MeansEnhanced*dan *K-Means*MMCA.Untuk melihat jumlah iterasi maksimum yang dibutuhkan dalam mencapai kriteria konvergen,maka pada tiap himpunan data dilakukan 7 kali simulasi. Tabel 4 menunjukkan hasil dari simulasi ini.

Tabel 4. Jumlah Iterasi pada Himpunan Data

Simulasi ke-	Jumlah Iterasi											
	Iris			Wine			Ruspini			Seeds		
	K-MEANS	ENHANCED	MMCA	K-MEANS	ENHANCED	MMCA	K-MEANS	ENHANCED	MMCA	K-MEANS	ENHANCED	MMCA
1	4	6	4	12	5	8	4	4	2	15	5	7
2	5	6	4	9	5	8	4	4	2	5	5	7
3	5	6	4	7	5	8	2	4	2	13	5	7
4	8	6	4	7	5	8	3	4	2	8	5	7
5	8	6	4	5	5	8	5	4	2	10	5	7
6	9	6	4	12	5	8	3	4	2	7	5	7
7	4	6	4	4	5	8	2	4	2	8	5	7

Hasil simulasi pada tiap himpunan data menunjukkan bahwa jumlah iterasi yang dibutuhkan untuk mencapai kriteria konvergen dengan algoritma *K-Means* selalu berbeda pada setiap simulasi. Sedangkan algoritma *K-MeansEnhanced*dan *K-Means*MMCA selalu membutuhkan jumlah iterasi yang sama.

Kemudian dilakukan simulasi untuk menentukan nilai rata-rata similaritas himpunan data terhadap data *benchmark*.Pada simulasi ini jumlahiterasi ditentukan hingga iterasi maksimum pada hasil simulasi sebelumnya. Sebagai contoh, untuk himpunan data Iris dilakukan hingga 9 iterasi. Hasil rata-rata similaritas dapat dilihat pada Tabel 5.

Tabel 5. Nilai Rata-Rata Similaritas pada Himpunan Data *Benchmark*

Jumlah Iterasi	Wine			Seeds			Iris			Ruspini		
	K-MEANS	ENHANCED	MMCA	K-MEANS	ENHANCED	MMCA	K-MEANS	ENHANCED	MMCA	K-MEANS	ENHANCED	MMCA
1	0.6610	0.7079	0.6067	0.7681	0.7762	0.7333	0.6933	0.9467	0.8533	0.6853	0.7733	1.0000
2	0.6689	0.7135	0.5618	0.7857	0.8667	0.8000	0.8660	0.9133	0.9000	0.8613	0.9600	1.0000
3	0.6625	0.7079	0.5618	0.7900	0.8857	0.8429	0.8327	0.9000	0.8933	0.7249	1.0000	1.0000
4	0.6757	0.7022	0.5674	0.8429	0.8905	0.8714	0.8567	0.8933	0.8933	0.8933	1.0000	1.0000
5	0.6824	0.7022	0.5730	0.8871	0.8905	0.8857	0.7840	0.8867	0.8933	0.8960	1.0000	1.0000
6	0.6891	0.7022	0.5674	0.8919	0.8905	0.8905	0.7927	0.8867	0.8933			
7	0.6933	0.7022	0.5730	0.8910	0.8905	0.8905	0.8533	0.8867	0.8933			
8	0.6757	0.7022	0.5730	0.8938	0.8905	0.8905	0.7433	0.8867	0.8933			
9	0.6850	0.7022	0.5730	0.8919	0.8905	0.8905	0.7947	0.8867	0.8933			
10	0.6674	0.7022	0.5730	0.8924	0.8905	0.8905						
11	0.6592	0.7022	0.5730									
12	0.6678	0.7022	0.5730									

Pada Tabel 5 dapat dilihat bahwa pada himpunan data Wine, menggunakan algoritma*K-MeansEnhanced*menunjukkan nilai similaritas yang lebih baik ,yaitu 0.7022, dibandingkan dengan *K-Means*MMCA dan *K-Means*. Dalam hal ini,*K-MeansEnhanced*mencapai kriteria konvergen (tidak ada

perubahan nilai similaritas) lebih cepat dibandingkan dengan *K-Means*MMCAyaitu pada iterasi ke-4.Pada himpunan data Seeds, algoritma *K-MeansEnhanced*mencapai kriteria konvergen lebih cepat dibandingkan 2 algoritma lainnya yaitu pada iterasi ke-5 dengan nilai similaritas 0.8905.Pada himpunan data Iris, *K-Means*sangat fluktuatif dan mencapai nilai terendah ketika jumlah iterasi yang digunakan adalah 1 sedangkan algoritma*K-MeansEnhanced*dan *K-Means*MMCAmenghasilkan nilai similaritas yang lebih stabil. Algoritma*K-Means*MMCA mencapai kriteria konvergen lebih cepat yaitu ketika iterasi ke-3 dengan nilai similaritas yaitu 0.8933. Pada himpunan data Ruspini, *K-Means*juga menunjukkan hasil yang tidak stabil, sedangkan *K-MeansEnhanced*dan *K-Means*MMCAmencapai nilai similaritas tertinggi yaitu 1 pada iterasi ke-3 untuk *K-MeansEnhanced*dan iterasi ke-1 untuk *K-Means*MMCA.

Setelah dilakukan *clustering* pada tiap himpunan data, hitung nilai indeks validitas menggunakan perhitungan indeks *Silhouette*, *Calinski-Harabasz*, *Davies-Bouldin*, dan *Dunn*untuk masing-masing himpunan data dengan jumlah klaster3 sampai 7. Nilai tersebut dibandingkan sehingga didapatkan jumlah klaster optimal untuk himpunan data Iris, Ruspini, Seeds, dan Wine berdasarkan kriteria masing-masing indeks validitas. Nilai masing-masing indeks untuk keempat himpunan data menggunakan algoritma *K-MeansEnhanced*, *K-Means*MMCA, dan *K-Means*dapat dilihat pada Tabel 6 sampai Tabel 9.

Tabel 6. Nilai Indeks Validitas pada Himpunan Data Ruspini

K	ENHANCED				MMCA				KMEANS			
	SL	CH	DB	DUNN	SL	CH	DB	DUNN	SL	CH	DB	DUNN
3	0.633	136.285	2.500	0.927	0.633	136.285	2.500	0.927	0.633	605.577	2.500	0.927
4	**0.738**	**425.327**	**2.122**	**1.986**	**0.738**	**425.327**	**2.122**	**1.986**	**0.738**	**852.956**	**2.122**	**1.986**
5	0.629	356.590	2.129	1.986	0.702	404.803	3.410	1.986	0.702	711.655	3.410	1.183
6	0.515	285.841	6.358	0.589	0.652	366.875	4.105	1.183	0.594	611.456	5.820	0.910
7	0.563	343.968	5.205	0.965	0.585	347.624	7.005	0.878	0.489	552.294	5.196	1.036

Jumlah klaster optimal untuk himpunan data Ruspini menggunakan algoritma *K-MeansEnhanced*, *K-Means*MMCA, dan *K-Means* dapat dilihat pada Tabel 6. Jumlah klaster optimal dari himpunan data Ruspini berdasarkan keempat kriteria tersebut menggunakan algoritma *K-MeansEnhanced*, *K-Means*MMCA, dan *K-Means* adalah 4. Hal tersebut sesuai dengan jumlah kelas himpunan data Ruspini berdasarkan data *benchmark* pada Tabel 3, yaitu sebanyak 4 kelas.

Tabel 7. Nilai Indeks Validitas pada Himpunan Data Iris

K	ENHANCED				MMCA				KMEANS			
	SL	CH	DB	DUNN	SL	CH	DB	DUNN	SL	CH	DB	DUNN
3	**0.551**	**560.366**	**3.521**	**1.872**	**0.553**	**560.400**	**3.520**	**1.807**	**0.553**	**2228.427**	**3.520**	**1.807**
4	0.497	529.121	5.019	1.154	0.495	527.835	4.973	1.286	0.498	1786.575	5.073	1.154
5	0.373	459.747	4.627	1.171	0.367	455.484	4.410	1.171	0.489	1485.157	6.400	0.921
6	0.437	400.114	6.678	0.702	0.366	470.406	5.800	1.311	0.371	1292.156	5.803	1.311
7	0.351	449.237	7.244	1.002	0.326	436.061	6.463	1.086	0.351	1148.481	7.002	1.002

Pada Tabel 7, dapat dilihat bahwa jumlah klaster optimal dari himpunan data Iris menggunakan algoritma *K-MeansEnhanced*, *K-Means*MMCAdan *K-Means*berdasarkan kriteria indeks *Silhouette*, *Calinski-Harabasz*, *Davies-Bouldin*, dan *Dunn* adalah 3. Hal tersebut sesuai dengan jumlah kelas himpunan data Iris berdasarkandata *benchmark* pada Tabel 3, yaitu sebanyak 3 kelas.

Tabel 8. Nilai Indeks Validitas pada Himpunan Data Seeds

K	ENHANCED				MMCA				KMEANS			
	SL	CH	DB	DUNN	SL	CH	DB	DUNN	SL	CH	DB	DUNN
3	**0.468**	**374.614**	**1.428**	1.224	**0.468**	**374.614**	**1.428**	1.224	**0.472**	**2588.931**	**1.438**	**1.338**
4	0.395	327.835	2.285	**1.293**	0.413	327.439	2.225	1.060	0.395	1992.934	2.285	1.293

5	0.363	308.483	2.537	1.164	0.362	310.218	2.712	1.267	0.360	1683.348	2.700	1.267
6	0.326	268.272	2.893	1.059	0.365	302.394	3.303	1.161	0.366	1478.705	3.292	1.161
7	0.356	293.776	3.619	1.187	0.349	293.296	3.687	**1.321**	0.356	1335.965	3.520	1.284

Jumlah klaster optimal dari himpunan data Seeds berdasarkan kriteria masing-masing indeks validitas, adalah 3 setelah dilakukan *clustering* menggunakan algoritma *K-Means*. Selanjutnya, jumlah klaster optimal berdasarkan kriteria *Silhouette*, *Calinski-Harabasz*, *Davies-Bouldin*, dan *Dunn*menggunakan algoritma *K-MeansEnhanced* berturut-turut adalah 3, 3, 3, dan 4. Terakhir, jumlah klaster optimal dari himpunan data Seedsberdasarkan kriteria keempat indeks validitasmenggunakan algoritma *K-Means*MMCAberturut-turut adalah 3, 3, 3, dan 7. Sedangkan berdasarkan data *benchmark* pada Tabel 3, jumlah kelas untuk himpunan data Seeds adalah 3. Pernyataan tersebut tidak sesuai dengan jumlah klaster optimal yang diprediksi oleh indeks validitas *Dunn*setelah dilakukan *clustering* pada himpunan data Seeds menggunakan algoritma *K-MeansEnhanced* dan *K-Means*MMCA. Hal ini dapat disebabkan karena indeks validitas *Dunn* sangat sensitif pada data *noise*, *outliers*, atau terdapat dua klaster yang sangat berdekatan (Baarsch, J., & Celebi, M. E., 2012).

Tabel 9. Nilai Indeks Validitas pada Himpunan Data Wine

K	ENHANCED				MMCA				KMEANS			
	SL	CH	DB	DUNN	SL	CH	DB	DUNN	SL	CH	DB	DUNN
3	**0.571**	561.816	3.405	0.934	0.560	497.005	**2.065**	1.477	**0.571**	**2396.546**	**3.405**	0.934
4	0.559	702.674	**2.770**	**0.977**	0.555	700.983	4.167	1.147	0.561	2303.098	3.627	1.046
5	0.517	710.010	5.548	0.681	0.558	725.894	4.167	**1.874**	0.549	2085.445	4.572	0.883
6	0.542	**850.293**	6.034	0.651	0.533	725.894	4.167	1.131	0.566	1973.548	4.389	**1.557**
7	0.520	813.783	7.505	0.464	**0.562**	**1187.546**	7.361	1.358	0.562	2049.276	7.462	1.322

Jumlah klaster optimal dari himpunan data Wineberdasarkan kriteria *Silhouette*, *Calinski-Harabasz*, *Davies-Bouldin*, dan *Dunn*menggunakan algoritma *K-MeansEnhanced* berturut-turut adalah 3, 6, 4, dan 4. Jumlah klaster optimal berdasarkan kriteria keempat validitas tersebut menggunakan algoritma *K-Means*MMCAberturut-turut adalah 7, 7, 3, dan 5. Sedangkan, menggunakan algoritma *K-Means* berturut-turut adalah 3, 3, 3, dan 6. Terdapat banyak perbedaan jumlah klaster optimal setelah dilakukan *clustering* menggunakan ketiga algoritma, dimana jumlah kelas berdasarkan data *benchmark* pada Tabel 3, untuk himpunan data Wineadalah 3 kelas.

Dengan membandingkan karakteristik dari keempat himpunan data yang digunakan, himpunan data Wine memiliki dimensi data yang paling besar, yaitu 13. Sedangkan himpunan data Iris, Ruspini, dan Seeds memiliki jumlah dimensi data berturut-turut adalah 2, 4, dan 7. Untuk itu perlu dilakukan penelitian lebih lanjut mengenai pengaruh ukuran dimensi data yang besar terhadap penggunaan indeks validitas pada masalah *clustering*.

4. Simpulan

Metode *Enhanced* dan MMCA merupakan metode yang dapat digunakan dalam pemilihan titik pusat awal klaster pada algoritma *K-Means*. Berdasarkan hasil simulasi, jumlah iterasi yang dibutuhkanuntuk mencapai kriteria konvergen oleh algoritma *K-Means* selalu berubah pada setiap simulasi. Sedangkan algoritma *K-Means Enhanced* dan *K-Means*MMCA membutuhkan jumlah iterasi yang tetap pada setiap simulasi. Selain itu, nilai rata-rata similaritas yang dihasilkan dengan menggunakan algoritma *K-Means* selalu berubah-ubah tidak menuju ke suatu nilai tertentu sedangkan 2 algoritma lainnyaselalu menuju ke suatu nilai tertentu. Selanjutnya, hasil kajiankinerja indeks validitas *Silhouette*, *Davies-Bouldin*, *Dunn*, dan *Calinski-Harabasz* pada data *benchmark*, yaitu himpunan dataIris, Ruspini, Seeds, dan Wine, setelah dilakukan *clustering* menggunakan algoritma *K-Means*, *K-MeansEnhanced*, dan *K-Means*MMCAmenunjukkan bahwa indeks validitas *Silhouette*, *Davies-Bouldin*, dan *Calinski-Harabasz* memiliki hasil yang lebih baik dibandingkan dengan indeks validitas *Dunn*.

Daftar Pustaka

Baarsch, J., & Celebi, M. E. (2012). Investigation of Internal Validity Measures for K-Means Clustering. *International Multiconference of engineers and computer scientists 1* (hal. 14-16). LA: Louisiana Board of Regents.

Bakshi, M., Derakhshi, M. R., & Zafarani, E. (2012, June). Review and Comparison between Clustering Algorithms with Duplicate Entities Detection Purpose. *Computer Science & Emerging Technologies, 3*, 108-114.

Dunn, J. C. (1973-09-01). Well-Separated Clusters and Optimal Fuzzy Partitions. *Journal of Cybernetics* (published 1974). **4** (1): 95–104.

Davies, D. L., & Bouldin, D. W. (1979, May). A Cluster Separation Measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 224-227.

Halkidi, M., Batistakis, Y., & Vazirgiannis, M. (2001). On Clustering Validation Techniques. *Intelligent Information Systems*, 107-145.

Jain, A. K. (2010). Data Clustering: 50 years beyond K-means. *Pattern Recognition Letters 31*, 651-666.

Jain, A., Murty, M., & Flynn, P. (1999). Data Clustering: A Review. *ACM Computing Surveys*, 264-323.

Lasheng, C., & Yuqiang, L. (2017). Improved Initial Clustering Center Selection Algorithm. *SIGNAL PROCESSING Algorithm, Architecture, Arrangements and Application (SPA).* Poznan: IEEE.

Liu, Y., Li, Z., Xiong, H., Gao, X., & Wu, J. (2010). Understanding of Internal Clustering Validation Measures. *IEEE International Conference on Data Mining*, (hal. 911-916).

Maitra, R., & Melnykov, V. (2012). Simulating Data to Study Performance of Finite Mixture Modeling and Clustering Algorithms. *Computational and Graphical Statistics*, 1-26.

Nanda, S., Mahanty, B., & Tiwari, M. (2010). Clustering Indian stock market data for portfolio management. *Expert Systems with Applications*, 8793–8798.

Nazeer, K. A., & Sebastian, M. (2009). Improving the Accuracy and Efficiency of the K-means Clustering Algorithm. *World Congress on Engineering*, *1*. London.

Rosseeuw, P. J. (1987). Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Computational and Applied Mathematics*, 53-65.

Yedla, M., Pathakota, S. R., & Srinivasa, T. M. (2010). Enhanching K-Means Clustering Algorithm with Improved Initial Center. *International Journal of Computer Science and Information Technologies, Vol 1(2)*, 121-125