J. Math. and Its Appl.
E-ISSN: 2579-8936
P-ISSN: 1829-605X
Vol. 15, No. 1, Maret 2018, 1-15

Optimasi Kernel K-Means dalam Pengelompokan Kabupaten/Kota Berdasarkan Indeks Pembangunan Manusia di Indonesia

Kasiful Aprianto
Badan Pusat Statistik Provinsi Sulawesi Barat, Jl. RE Martadinata No. 10, Mamuju
e-mail: apriantokasiful@gmail.com

Abstrak

Kernel k-means (KKC) bekerja dengan mengubah data dari initial space ke dalam featured space dan k-means dijalankan menggunakan data featured space tersebut. Permasalahan utama dari KKC adalah inisialisasi *centroid* dimana posisi *centroid* sangat mempengaruhi hasil dari pengelompokan itu sendiri. Paper ini menjelaskan peran optimasi dalam pencarian titik *centroid* yang tepat untuk menemukan hasil yang baik dan stabil. Particle swarm optimization (PSO) dipilih karena mudah untuk diimplementasikan. Principal Component Analysis (PCA) digunakan untuk mereduksi dimensi tanpa mengurangi karakteristik data secara signifikan. Kemampuan PCA inilah yang kemudian digunakan untuk meningkatkan akurasi dari clustering. Keuntungan dari KKC-PSO dengan PCA adalah menemukan *centroid* yang tepat dengan waktu pencarian yang lebih singkat. Hasil percobaan menunjukkan bahwa KKC-PSO dengan PCA memberikan hasil yang optimal dilihat dari akumulasi within sum square yang kecil, dan juga stabil dilihat dari hasil perulangan yang selalu berhasil mendapatkan nilai optimal. Selanjutnya, algoritma ini digunakan untuk melihat pengelompokan kabupaten/kota di Indonesia berdasarkan indikator indeks pembangunan manusia (IPM) menggunakan data yang bersumber dari Badan Pusat Statistik. Kabupaten atau kota dapat dibagi menjadi 3 kelompok berdasarkan variabel IPM, yaitu kelompok dengan IPM yang rendah (kelompok 2), sedang (kelompok 3), dan tinggi (kelompok 1). Dari penelitian ini terlihat bahwa masih terdapat perbedaan IPM antar kabupaten, dimana perbedaan IPM ini cenderung terkelompok dan berdekatan antara satu dengan yang lainnya.
Kata kunci: kernel k-means, optimasi, particle swarm optimization, principal component analysis, Gaussian

Abstract

Kernel K-means (KKC) works by converting data from initial space into featured space and k-means is run using the featured space data. The main problem of KKC is the initialization of centroid where the position of the centroid will greatly affect the result of the grouping itself. This paper explains the role of optimization in the search for the right centroid point to find good and stable results. Particle swarm optimization (PSO) is chosen because it is easy to implement. Principal Component Analysis (PCA) is used to reduce dimensions without significantly reducing data characteristics. This PCA capability is then used to improve the

accuracy of clustering. The advantage of KKC-PSO with PCA is finding the right centroid with a shorter search time. The experimental results show that KKC-PSO with PCA gives optimal results seen from the accumulation within small summits, and also stable from the results of the loop that always managed to get the optimal value. Furthermore, this algorithm is used to see the grouping of districts / cities in Indonesia based on indicators of human development index (HDI) using data sourced from the Central Bureau of Statistics. Districts or cities can be divided into 3 groups based on HDI variables, those with low HDI (group 2), medium (group 3), and high (group 1). From this research, it can be seen that there are differences of HDI between districts, where the differences in HDI tend to be clustered and close to each other.
Keywords: *kernel k-means, particle swarm optimization, principal component analysis, Gaussian*

1 Pendahuluan

Klaster secara umum digunakan untuk melihat kesamaan antar objek dan mengelompokkannya ke dalam beberapa bagian. Peran klastering dalam ilmu pengetahuan menjadi semakin pesat berkembang seiring dengan kebutuhannya yang semakin meningkat, diantaranya berupa bioinformatika, bisnis dan pemasaran, pendidikan, dan pemerintahan. K-means merupakan salah satu algoritma populer dalam pembentukan klaster. Selain karena kemudahan, algoritma ini juga mampu memberikan hasil yang efektif [1]. Kelemahan dari algoritma ini adalah pembagian kelas yang dibentuk harus terpisah secara linear [2]. Hal ini membuat k-means tidak memberikan hasil yang baik ketika dihadapkan dengan permasalahan data dengan kondisi yang terpisah secara nonlinear. Untuk menjawab permasalahan tersebut, diperlukan adanya suatu perluasan dimensi yang terpetakan dengan menggunakan kernel, lalu menggunakan kernel tersebut untuk dijadikan sebagai variabel yang dicari kelasnya dengan k-means, atau disebut sebagai kernel k-means (KKC). Penelitian sebelumnya membuktikan bahwa penggunaan kernel ke dalam k-means, tidak hanya mampu menangkap fitur nonlinear dari data yang diinisialisasi, tetapi juga mampu mengekspresikan hasil dari kompleksnya pemetaan nonlinear [2].

K-means bekerja dengan menentukan terlebih dahulu posisi dari *centroid* secara random. Berdasarkan penelitian sebelumnya, penentuan secara random mempengaruhi hasil dari pengelompokan itu sendiri, dimana hasil yang diberikan belum stabil. Untuk beberapa kasus juga bisa menyebabkan hasil yang tidak optimal [3]. Untuk mendapatkan posisi awal klaster yang baik, perlu adanya beberapa serangkaian percobaan yang dilakukan. Terdapat beberapa cara untuk mengetahui posisi awal klaster yang optimal, diantaranya yaitu dengan melakukan pencarian optimasi. Pada penelitian ini menggunakan *particle swarm optimization* (PSO). PSO merupakan metode pencarian titik optimum yang terinspirasi dari fenomena alam dan perilaku burung [4]. Penggabungan K-means dan PSO pernah dilakukan sebelumnya oleh Ida, Asyrofa,

dan Wayan [5] untuk pencarian *centroid* yang tepat dalam klasterisasi nasabah bank berdasarkan tingkat likuiditas.

Seperti yang diketahui, kernel mencoba memberikan *feature space* dengan melakukan serangkaian *initial product* untuk masing-masing vektor dalam suatu matriks. Jika partikel diberikan parameter pencarian sebanyak n, tentu memperkecil peluang untuk mendapatkan nilai optimum. Selain itu juga, ukuran yang besar akan memakan waktu komputasi yang besar. Dengan menggunakan PCA, kernel yang dimaksud kemudian diringkas dan dibentuk komponen baru yang bisa mewakili n dimensi dari kernel yang ada. Hasil dari PCA kemudian digunakan untuk mencari *centroid* terbaik dari kernel K-means dengan PSO. Menggunakan PCA untuk data hasil *feature space* telah dilakukan oleh Ding dan He [6] dimana dalam penelitian mereka hasil dari reduksi dimensi tersebut cukup baik digunakan ke dalam kasus pengelompokan menggunakan K-Means. Berdasarkan kebutuhan KKC dalam mengetahui inisialisasi *centroid* awalnya, maka paper ini meneliti efektifitas dari penggunaan PSO dalam pencarian inisialisasi awal *centroid* yang optimal dengan bantuan PCA. Untuk selanjutnya akan disingkat menjadi KKC-PSO.

Hasil dari algoritma yang dibuat untuk selanjutnya digunakan untuk mengelompokkan kabupaten/kota di Indonesia menggunakan data indeks pembangunan manusia (IPM). Berdasarkan situs resmi dari Badan Pusat Statistik, IPM menjelaskan bagaimana penduduk dapat mengakses hasil pembangunan dalam memperoleh pendapatan, kesehatan, pendidikan, dan sebagainya. IPM sendiri digunakan sebagai salah satu indikator penting untuk mengukur keberhasilan dalam upaya membangun kualitas hidup masyarakat.

Berdasarkan penelitian sebelumnya, Setiyono dan Mukhlash [7] berhasil menerapkan spatial clustering dalam mengelompokkan kesejahteraan penduduk kota Surabaya menggunakan algoritma GDBScan, Clarans, dan Cure. Hal ini menunjukkan bahwa proses spatial kenyataannya tidak selalu murni terjadi secara random [8]. Untuk penelitian ini selanjutnya menggunakan KKC-PSO dalam pengelompokan IPM di Indonesia, dimana hasil menunjukkan bahwa kabupaten/kota yang memiliki kemiripan antar cluster saling berdekatan, kemudian membagi kabupaten/kota menjadi beberapa jenis berdasarkan hasil yang diperoleh dari perhitungan.

2 Metode Penelitian

2.1 Dasar Teori

2.1.1 Kernel Gaussian

Kernel digunakan untuk mengatasi permasalahan dimensi, dimana kita dapat mendefinisikan kernel seperti pada persamaan (1),

$$K(\mathrm{x},\mathrm{x}') = \exp\left(-\frac{(\mathrm{x}-\mathrm{x}')^2}{2\sigma^2}\right) \quad (1)$$

dengan masing-masing x dan x' adalah vektor, dan $(\mathrm{x}-\mathrm{x}')^2$ dikenal dengan istilah jarak euclidean kuadrat antara kedua vektor. Sedangkan σ merupakan parameter bebas yang ditentukan di awal. Jika $\gamma = 1/(2\sigma^2)$, maka persamaan yang lebih sederhana dari kernel ditunjukkan sebagai persamaan (2) berikut:

$$K(\mathrm{x},\mathrm{x}') = \exp(-\gamma(x-x')^2) \quad (2)$$

Dari persamaan kernel Gaussian pada rumus diatas, terlihat bahwa nilai yang dihasilkan pada kernel tersebut berada pada rentang 1 $(\text{jika } \mathrm{x} = \mathrm{x}')$ hingga mendekati nol. Nilai yang diberikan tidak akan pernah tepat nol. Persamaan ini bisa dijelaskan menggunakan deret *Taylor* sebagai berikut (nilai $\gamma = 1$):

$$K(\mathrm{x},\mathrm{x}') = \exp(-(x-x')^2)$$

$$K(\mathrm{x},\mathrm{x}') = \exp(-\mathrm{x}^2)\exp(-\mathrm{x}'^2)\exp(-2\mathrm{x}\mathrm{x}')$$

$$K(\mathrm{x},\mathrm{x}') = \exp(-\mathrm{x}^2)\exp(-\mathrm{x}'^2)\sum_{k=0}^{\infty}\frac{2\mathrm{x}^k\mathrm{x}'^k}{k!} \quad (3)$$

Jika *feature space* yang dibentuk adalah matriks, maka membentuk *feature space* dengan ukuran matriks (nxn) seperti pada persamaan (4).

$$K(\mathbf{X}^{\mathbf{T}},\mathbf{X}) = \begin{bmatrix} K(\mathrm{x}_1,\mathrm{x}_1) & \cdots & K(\mathrm{x}_1,\mathrm{x}_n) \\ \vdots & \ddots & \vdots \\ K(\mathrm{x}_n,\mathrm{x}_1) & \cdots & K(\mathrm{x}_n,\mathrm{x}_n) \end{bmatrix} \quad (4)$$

2.1.2 Kernel K-Means

Diketahui $K(\mathbf{X}^{\mathbf{T}},\mathbf{X}) = \mathbf{Y} = [\mathbf{Y}_1,\mathbf{Y}_2,\dots,\mathbf{Y}_N]^T$ adalah dataset awal yang digunakan dengan baris sebanyak *N* baris dengan setiap $\mathbf{Y_i} = [y_1,y_2,\dots,y_n]^T$ sebanyak *n* variabel. $\mathbf{C} = [\mathbf{C_1},\mathbf{C_2},\dots,\mathbf{C}_m]^T$ adalah titik awal *centroid* yang diinisialisasi secara acak sebanyak *m* *centroid* dengan setiap $\mathbf{C_j} = [c_1,c_2,\dots,c_n]^T$ sebanyak *n* variabel. Masing-masing $\mathbf{Y_i}$ dikumpulkan kedalam masing-masing klaster berdasarkan jarak yang paling dekat dengan *centroid* $\mathbf{C}$. Jarak antara variabel $\mathbf{Y_i}$ dengan $\mathbf{C_j}$ bisa dilakukan dengan menggunakan rumus Euclidean distance seperti pada persamaan (5).

$$d(\mathbf{Y_i}, \mathbf{C_j}) = \sqrt{\sum_{k=1}^{n} (y_{ik} - c_{ik})^2} \quad (5)$$

Untuk selanjutnya, masing-masing baris dikelompokkan ke dalam klaster berdasarkan jarak terdekat dengan semua *centroid* yang ada. Dari semua data yang dikelompokkan tadi, masing-masing di rata-rata kan untuk dijadikan *centroid* baru dalam perulangan yang dilakukan. Algoritma dasar dari k-means adalah sebagai berikut:

1. Tentukan jumlah klaster (m) dan inisialisasi setiap *klaster* dengan memberi nilai *centroid* secara acak.
2. Hitung jarak setiap data terhadap masing-masing *centroid*
3. Kelompokkan data ke dalam klaster berdasarkan jarak yang paling pendek
4. Hitung pusat klaster yang baru dengan memberikan nilai rata-rata dari masing-masing data yang terkelompokkan
5. Lakukan perulangan hingga mencapai iterasi tertentu

2.1.3 Normalisasi

Berhubung dengan Gaussian kernel yang digunakan menggunakan jarak euclidean, maka sebelum memasukkannya ke dalam kernel, perlu dilakukannya suatu normalisasi data. Normalisasi data digunakan untuk memastikan terdapat jarak yang setimbang dengan variabel yang berbeda. Sebagai contoh, ketika memiliki data antara berat badan dan ukuran sepatu. Satu satuan ukuran sepatu tidak bisa dibandingkan dengan satu kilogram berat badan. Hal ini membuat selisih antara satu titik dengan titik lainnya menjadi sangat besar. Terdapat beberapa cara untuk melakukan normalisasi data, salah satunya yaitu seperti pada persamaan (6).

$$x_{i(normalisasi)} = \frac{x_i - (\min) x}{(\max) x - (\min) x} \quad (6)$$

2.1.4 Particle Swarm Optimization (PSO)

PSO merupakan algoritma yang terinspirasi dari fenomena alam seperti perilaku ikan yang mengoptimalkan permasalahan dengan menganalisis kawanan secara acak, dimana masing-masing ikan melakukan pencarian yang disebut sebagai particle, dan populasi bergelombol yang disebut swarm [5]. Setiap partikel bergerak dan mencari posisi optimal yang pernah mereka lewati. Pergerakan mereka dipengaruhi oleh titik dengan nilai terbaik yang pernah dilewati oleh masing-masing partikel (partikel lokal) dan juga satu posisi terbaik dari semua yang pernah dilewati partikel (partikel global). Pada setiap iterasi, masing-masing partikel mengingat nilai dari setiap partikel lokal mereka sebelumnya (fitness lokal). Pada saat itu juga, mereka semua mengetahui posisi terbaik yang dimiliki teman-temannya sehingga setiap mereka mengetahui posisi terbaik dari yang pernah mereka semua lalui (fitness global) [9]. Partikel bergerak sedikit

demi sedikit dengan arah menuju partikel global dan dipengaruhi juga dengan partikel lokal dari masing-masing partikel. Persamaan yang menggambarkan ilustrasi diatas adalah sebagai berikut:

$$v_i(t) = v_i(t-1) + c_1 r_1 * (P_i - X_i(t-1) + c_2 r_2 * (G - X_i(t-1) \quad (7)$$

$$X_i(t) = X_i(t-1) + v_i(t) \quad (8)$$

dengan

i : Iterasi

v : Kecepatan partikel

c : Random value yang ditetapkan ($0 < c < 1$)

r : Random value yang berubah setiap iterasi ($0 < r < 1$)

P : Posisi lokal fitness partikel (vektor)

G : Posisi global fitness partikel (vektor)

2.1.5 Principal Component Analysis (PCA)

Jika $\mathbf{X}$ merupakan matriks berukuran nxp, n observasi dan p-variat variabel acak X, maka analisis komponen utama merupakan suatu metode untuk mereduksi p-variat dari X. Reduksi dilakukan dengan memberikan nilai p-variat baru, dalam contoh ini Y dimana $Y_i, i = 1 \text{ sd } p$ merupakan kombinasi linear dari p-variat lama tanpa meninggalkan banyak informasi dari p-variat sebelumnya. Algoritma dari PCA dijelaskan sebagai berikut:

1. Mencari rata-rata untuk setiap variabel, lalu hitung kovarian

   Misalkan Y dan Z adalah variabel acak, maka:

   $$cov(Y,Z) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \bar{y})(z_i - \bar{z}) \quad (9)$$

   Dengan $\bar{y}$ dan $\bar{z}$ merupakan rataan sampel dari variabel Y dan Z dan n merupakan jumlah sampel.
2. Mencari nilai eigen dan vektor eigen dari matriks kovariansi empirik yang diperoleh
3. Menghitung proporsi viariansi dari masing-masing PC beserta nilai akumulasi untuk q-PC pertama

2.1.6 Indeks Pembangunan Manusia (IPM)

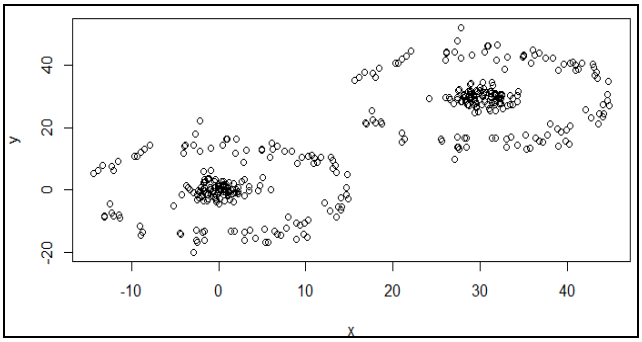
IPM merupakan indikator tingkat pembangunan manusia suatu wilayah, yang dihitung melalui perbandingan dari angka harapan hidup, pendidikan dan standar hidup layak [10]. Dijadikannya IPM sebagai indikator pembangunan dilandasi dari terdapatnya ketidakakuratan pendapatan perkapita dalam melihat pembangunan suatu wilayah [11].

Berdasarkan situs resmi dari Badan Pusat Statistik, IPM menjelaskan bagaimana penduduk dapat mengakses hasil pembangunan dalam memperoleh pendapatan, kesehatan, pendidikan, dan

sebagainya. IPM sendiri memiliki 3 dimensi dengan 4 indikator. Dimensi tersebut yaitu dimensi umur panjang dan hidup sehat, pengetahuan, dan standar layak hidup. Sedangkan indikatornya yaitu angka harapan hidup, harapan lama sekolah, rata-rata lama sekolah, dan pengeluaran per kapita yang disesuaikan. IPM di Indonesia digunakan sebagai indikator untuk mengukur keberhasilan dalam membangun kualitas hidup masyarakat. Dalam hal ini, IPM mampu menentukan level pembangunan suatu wilayah di Indonesia.

2.2 Metodologi Penelitian

Untuk memastikan algoritma dibangun dengan benar, dilakukan terlebih dahulu serangkaian percobaan dengan data buatan. Data di generate sebanyak 400 baris dengan 2 variabel X dan Y. Dari data tersebut, secara visual dapat dilihat bahwa klaster yang diharapkan adalah sebanyak 4 klaster seperti yang ditunjukkan pada Gambar 1.



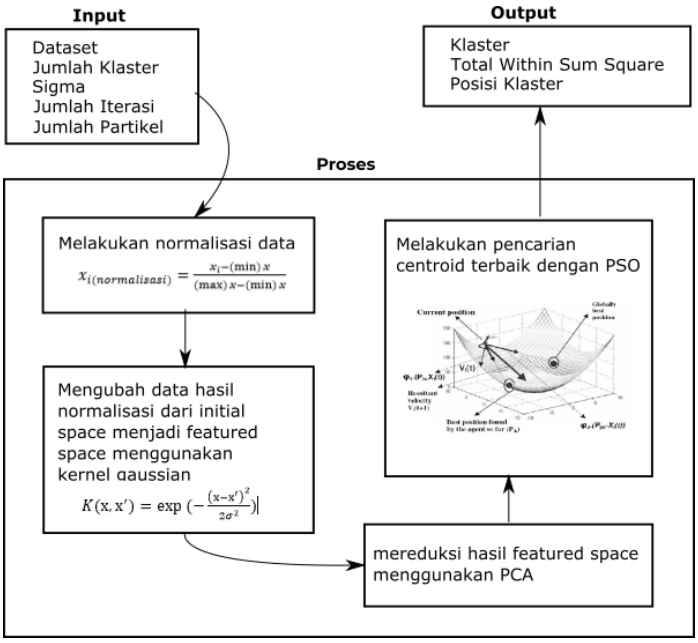
Gambar 1 Dataset awal dengan 4 klaster

Dari percobaan yang dilakukan, dilihat apakah KKC yang dibuat berhasil mengelompokkan dataset dengan baik. Percobaan dilakukan berulang kali untuk melihat hasil yang diberikan KKC belum optimal. Hipotesis awal penelitian ini adalah hasil yang diberikan KKC setelah direduksi menggunakan PCA lebih baik daripada KKC tanpa reduksi data.

Percobaan kedua menggunakan hybrid KKC-PSO dengan data hasil reduksi secara berulang dengan membandingkan hasilnya dengan KKC. Hipotesis kedua pada penelitian ini adalah KKC-PSO menghasilkan posisi klaster yang lebih stabil dan lebih akurat dibandingkan dengan KKC.

Jika dilakukan dengan data nonlinear, KKC-PSO berhasil menentukan klaster dengan baik, ditunjukkan dengan nilai *within sum square* yang kecil. Penelitian selanjutnya yaitu menguji KKC-PSO dengan data yang linear. Hipotesis ketiga yaitu KKC-PSO tidak hanya baik untuk nonlinear, tetapi juga bisa digunakan untuk kasus linear. Jika ketiga hipotesis diatas terpenuhi, maka KKC-PSO bisa digunakan untuk kondisi data apapun, baik terpisah secara linear ataupun nonlinear. Untuk selanjutnya algoritma ini digunakan untuk menentukan klaster terbaik pada

data IPM yang terdiri dari angka harapan hidup, harapan lama sekolah, rata-rata lama sekolah, dan pengeluaran per kapita yang disesuaikan pada tahun 2016 yang diperoleh dari situs resmi Badan Pusat Statistik. Kabupaten/Kota dikelompokkan berdasarkan hasil klaster berdasarkan IPM sebanyak 3 klaster, yaitu klaster dalam kategori IPM rendah, sedang, dan tinggi. Untuk lebih jelasnya dijelaskan pada Gambar 2 dibawah

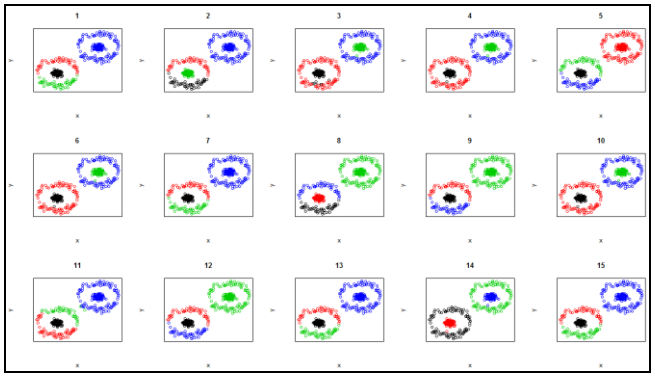


Gambar 2 Blok diagram KKC-PSO

3 Hasil dan Pembahasan

3.1 Data Buatan

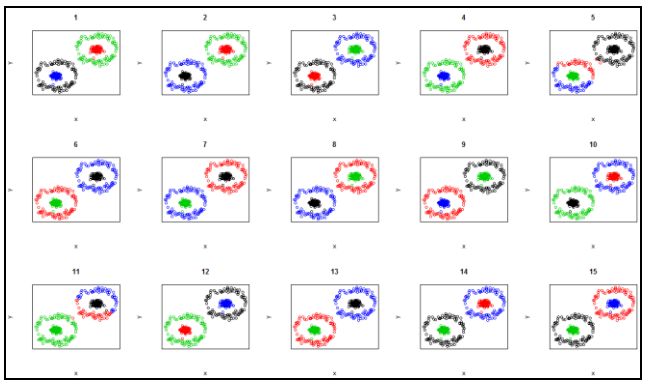
Hasil Percobaan untuk hipotesis pertama yaitu KKC dengan PCA lebih baik daripada KKC tanpa PCA. Data percobaan diharapkan bisa menemukan 4 klaster dengan masing-masing variabel dilakukan normalisasi terlebih dahulu. Untuk K-means sendiri menggunakan 10 iterasi. Kernel Gaussian menggunakan sigma sebesar varian maksimum dari data percobaan. Pada percobaan yang dilakukan sebanyak 15 kali untuk KKC tanpa PCA, diperoleh hasil yang ditunjukkan pada Gambar 3.



Gambar 3 Hasil KKC sebelum menggunakan PCA

Jika dilihat sekilas dari hasil plot pada Gambar 3, hanya 5 dari 15 plot yang sesuai dengan harapan, yaitu plot ke 3, 4, 6, 10, dan 14. Hal ini dikarenakan ukuran data setelah diubah ke dalam feature space menggunakan kernel Gaussian, ukuran variabel yang semula hanya 2, kini menjadi 400. Ketika menggunakan 10 iterasi tentu hal ini membuat pergerakan *centroid* semakin besar, ditambah dengan penetapan *centroid* yang random sebelum dilakukan K-means.

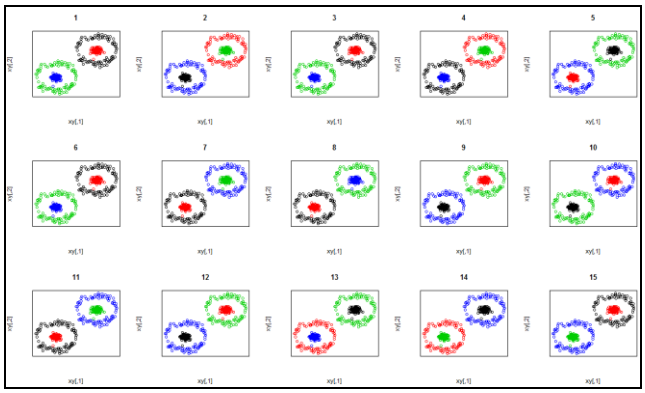
Untuk menjawab permasalahan tersebut, dilakukan peringkasan variabel yang semula 400 kini menjadi 3 variabel dengan tidak mengurangi secara signifikan variabel yang direduksi, sehingga kombinasi linear dari feature space telah terangkum ke dalam 3 variabel yang baru hasil dari PCA. Hasil dari KKC setelah direduksi seperti pada Gambar 4.



Gambar 4 Hasil KKC setelah menggunakan PCA

Terlihat dari Gambar 4 bahwa terdapat 13 dari 15 plot yang sesuai dengan harapan. Dengan demikian hipotesis pertama memberikan kesimpulan bahwa KKC dengan PCA memberikan hasil yang lebih baik dari KKC tanpa PCA.

Percobaan selanjutnya melihat apakah KKC-PSO memberikan hasil yang lebih stabil daripada KKC setelah keduanya direduksi menggunakan PCA. Untuk PSO sendiri menggunakan 20 partikel dengan iterasi pencarian sebanyak 20 kali. Hasil yang diperoleh ditunjukkan pada Gambar 5.



Gambar 5 Hasil KKC-PSO setelah menggunakan PCA

Tabel 1 *Within sum square* masing-masing kegiatan

Iterasi	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
KKC sebelum PCA	18995.6	18995.5	18644.7	18628.9	18995.5	18628.9	18995.5	18995.6	18995.5	18628.9	18991.7	18995.5	18995.6	18628.9	18995.6
KKC setelah PCA	18628.9	18628.9	18628.9	18628.9	18991.7	18628.9	18628.9	18630.1	18628.9	18628.9	18966.8	18628.9	18628.9	18628.9	18628.9
KKC-PSO setelah PCA	18628.9	18628.9	18628.9	18628.9	18628.9	18628.9	18628.9	18628.9	18628.9	18628.9	18628.9	18628.9	18628.9	18628.9	18628.9

Dari Gambar 5 beserta Tabel 1 diatas, terlihat bahwa semua plot menunjukkan hasil yang stabil dan konstan, dimana PSO mampu menemukan titik optimum dari *centroid* yang diberikan. Dengan demikian, hipotesis kedua memberikan kesimpulan bahwa KKC-PSO mampu memberikan hasil yang lebih stabil yang ditunjukkan seperti pada Gambar 6 daripada KKC tanpa PSO seperti pada Gambar 5 setelah keduanya dilakukan PCA pada *feature space*. Stabil yang dimaksud adalah ketika hasil yang diberikan selalu berada pada nilai optimum untuk setiap perulangan. Mengingat kecepatan perlu menjadi pertimbangan dalam melakukan klaster, Tabel 2 menggambarkan waktu yang dibutuhkan dalam melakukan klaster.

Tabel 2 Waktu masing-masing kegiatan (detik)

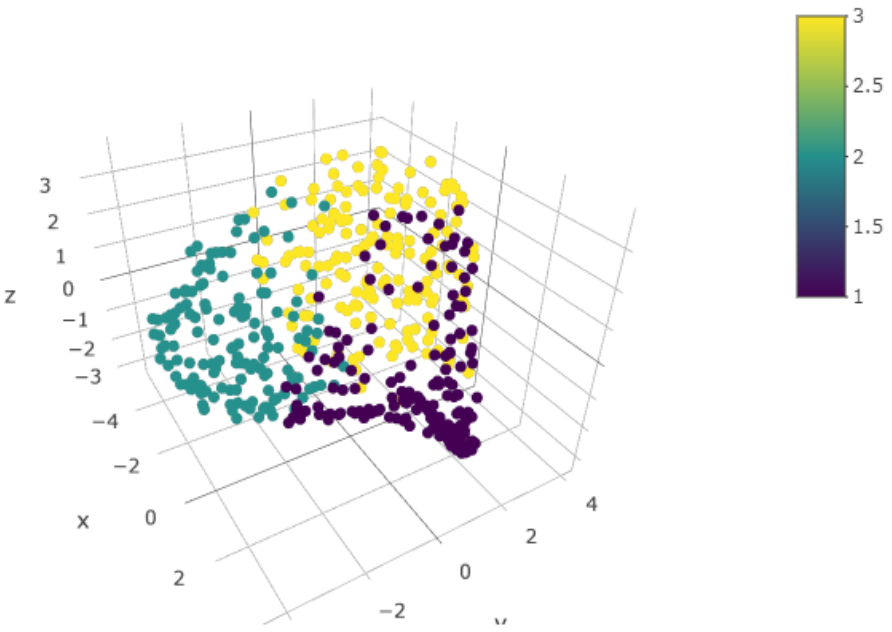
Iterasi	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
KKC sebelum PCA	0.34	0.23	0.27	0.39	0.3	0.36	0.3	0.43	0.21	0.34	0.47	0.38	0.38	0.35	0.37
KKC setelah PCA	0.03	0.03	0.06	0.03	0.01	0.12	0.11	0.01	0.01	0.17	0.02	0.03	0.15	0.01	0.17
KKC-PSO setelah PCA	8.89	8.76	8.88	9.12	8.77	8.98	8.8	8.73	8.95	8.97	8.88	9.15	9.24	9.13	8.83

Dari Tabel 2 terlihat bahwa dalam penggunaan KKC-PSO membutuhkan waktu yang besar. Ini dikarenakan pencarian yang dilakukan melakukan perhitungan KKC sebanyak 400 kali (20 iterasi x 20 partikel). Meskipun demikian, hasil yang diperoleh sepadan dimana dalam menggunakan KKC-PSO diperoleh hasil yang optimum di setiap perulangan yang telah dilakukan sehingga hasilnya bisa digunakan untuk melakukan analisis klaster.

3.2 Data Indeks Pembangunan Manusia (IPM)

Proses klaster dihitung menggunakan KKC-PSO dengan PCA. Untuk selanjutnya, setiap kabupaten/kota dikelompokkan ke dalam 3 bagian. Karena pengelompokan dilakukan dengan melihat feature space dari data IPM, maka untuk bisa melihat karakteristik dari pengelompokan

yang diberikan selain menggunakan rata-rata, juga bisa dilihat menggunakan visualisasi dari PCA, diambil 3 komponen utama dari PCA hasil dari kernel seperti pada Gambar 6. Secara visual dapat dilihat bahwa pengelompokan terpisah jelas pada kelompok berwarna ungu, namun untuk kelompok hijau dan kuning terpisah dan masih dapat dikelompokkan meskipun jarak antar kedua kelompok berdekatan.



Gambar 6 Visualisasi klaster menggunakan PCA

Agar hasil pengelompokan lebih meyakinkan, dilakukan uji Wilcoxon test untuk mengetahui apakah terdapat perbedaan yang signifikan antar klaster seperti pada Tabel 3. Terdapat perbedaan yang signifikan untuk angka harapan hidup antara klaster 1 dengan 2, dan 2 dengan 3. Untuk harapan lama sekolah, perbedaan yang signifikan terdapat pada klaster 1 dengan 2, dan 2 dengan 3. Untuk pengeluaran per kapita, terdapat perbedaan signifikan antara klaster 1 dengan 2, dan 2 dengan 3. Sedangkan untuk rata-rata lama sekolah, semua klaster terkelompok dan berbeda secara signifikan antara satu dengan lainnya. Dari uji Wilcoxon ini, dapat disimpulkan bahwa klaster yang dihasilkan sudah cukup baik dalam mengelompokkan masing-masing kabupaten kota kedalam 3 kategori, terlihat dari terdapatnya minimal satu variabel yang berbeda antara satu klaster dengan klaster lainnya.

Tabel 3 Hasil uji Wilcoxon dengan derajat kemaknaan 0.05

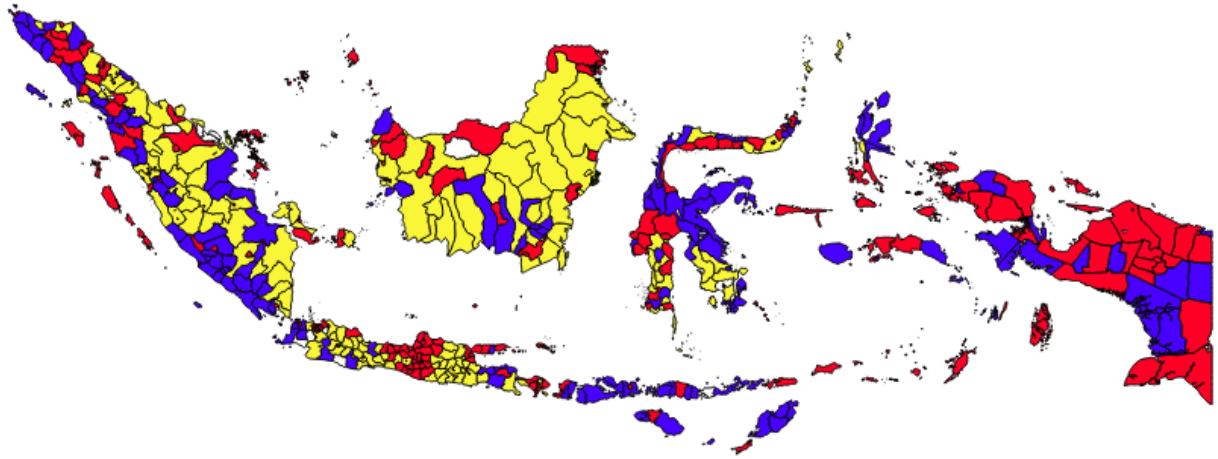
Var	Klaster (p-value)		
	1 dan 2	1 dan 3	2 dan 3
AHH	beda (2e-13)	sama (1)	beda (2e-16)
HLS	beda (0.008)	beda (0.004)	sama (0.9)
PPK	beda (2e-7)	sama (0.4)	beda (3e-13)
RLS	beda (0.05)	beda (0.002)	beda (0.02)

Dengan melihat rata-rata dan median dari setiap klaster, hal ini sudah sesuai dengan uji Wilcoxon. Berdasarkan angka harapan hidup, kelompok 2 memiliki rata-rata yang lebih rendah dibandingkan dengan dua kelompok lainnya. Untuk harapan lama sekolah, kelompok 2 dan 3 tidak jauh berbeda namun keduanya berbeda secara signifikan dibandingkan kelompok 1, dimana 2 dan 3 memiliki harapan lama sekolah yang lebih kecil dari kelompok 1. Untuk pengeluaran per kapita, kelompok 2 lebih rendah dibandingkan dengan kedua kelompok lainnya yang cenderung sama. Sedangkan untuk rata-rata lama sekolah, urutan dari yang tertinggi ke terendah yaitu pada kelompok 1, kelompok 3, dan kelompok 2. Dengan demikian, dapat disimpulkan bahwa hasil klaster yang dilakukan telah berhasil membagi kabupaten dan kota menjadi 3 bagian, yaitu kategori rendah (kelompok 2), sedang (kelompok 3), dan tinggi (kelompok 1).

Tabel 4 Rata-rata, median, dan kuartil dari masing-masing klaster terhadap variable

Klaster 1	AHH	HLS	PPK	RLS
	Min. :54.5	Min. : 2.34	Min. : 3725	Min. : 0.70
	Median :70.7	Median :12.98	Median :10672	Median : 9.04
	Mean :69.5	Mean :12.79	Mean :10440	Mean : 8.37
	Max. :77.5	Max. :17.03	Max. :22932	Max. :12.57
Klaster 2	AHH	HLS	PPK	RLS
	Min. :62.7	Min. :10.7	Min. : 5379	Min. :5.47
	Median :66.4	Median :12.4	Median : 8136	Median :7.45
	Mean :66.4	Mean :12.5	Mean : 8265	Mean :7.37
	Max. :69.8	Max. :14.6	Max. :11623	Max. :9.11
Klaster 3	AHH	HLS	PPK	RLS
	Min. :67.2	Min. :11.1	Min. : 6919	Min. :6.05
	Median :70.0	Median :12.4	Median : 9877	Median :7.68
	Mean :70.2	Mean :12.4	Mean : 9768	Mean :7.67
	Max. :73.4	Max. :14.5	Max. :12248	Max. :9.43

Hasil dari klaster yang dibentuk ditunjukkan pada Gambar 7. Dari gambar ini, terlihat bahwa klaster 1 yang terdiri dari warna merah, klaster 2 dengan warna biru, klaster 3 dengan warna kuning. Dari Gambar 7 juga terlihat bahwa hasil dari pengelompokan untuk setiap klaster cenderung terkelompok berdekatan satu dengan yang lainnya.



Gambar 7 Hasil pengelompokan Kabupaten/Kota di Indonesia

4 Simpulan

Berdasarkan serangkaian percobaan yang dilakukan, KKC dengan PCA mampu memberikan hasil yang lebih baik daripada KKC tanpa PCA. Hal ini dikarenakan ukuran dari feature space yang sangat besar sehingga pergerakan dari cenroid dalam mencari titik *centroid* menjadi semakin besar. Disamping itu, waktu yang diberikan menjadi sangat cepat karena mampu mengurangi beban komputasi dari perhitungan. Meskipun efektifitas KKC dengan PCA memberikan waktu yang singkat dengan hasil yang baik, namun tetap masih membawa peluang untuk terpilihnya titik yang tidak optimum, karena inisialisasi *centroid* awal secara random. Penggunaan PSO dalam melakukan pencarian nilai optimum dierapkan pada penelitian ini. Hasil yang diperoleh menunjukkan bahwa dengan menggunakan PSO, diperoleh hasil yang optimum dan stabil setelah dilakukan serangkaian perulangan. Sehingga dapat disimpulkan beberapa tahapan untuk bisa mendapatkan nilai yang optimal dari KKC adalah sebagai berikut:

1. Melakukan normalisasi data untuk setiap variabel
2. Mengubah data menjadi feature space menggunakan kernel
3. Melakukan reduksi dimensi hasil dari feature space dengan PCA
4. Menggunakan feature space hasil reduksi untuk dilakukan perhitungan kmeans dengan menggunakan PSO agar bisa menemukan hasil terbaik. Indikator baik tidaknya posisi klaster dapat menggunakan nilai akumulasi dari *within sum square* untuk setiap klaster dimana semakin kecil nilai yang diperoleh maka semakin baik posisi yang diberikan.

Melihat potensi dari KKC-PSO menggunakan PCA, maka kemampuan ini digunakan dalam mengelompokkan kabupaten/kota se Indonesia menggunakan data IPM menjadi 3 kelompok. Dari hasil pengelompokan, setelah melakukan pengujian beda rata-rata menggunakan Wilcoxon test, hasil menunjukkan bahwa terdapat perbedaan yang signifikan setidaknya satu

variabel dari masing-masing kelompok. Dari hasil Wilcoxon tadi, bisa dilihat dan ditarik kesimpulan bahwa kabupaten atau kota dapat dibagi menjadi 3 kelompok berdasarkan variabel IPM, yaitu kelompok dengan IPM yang rendah (kelompok 2), sedang (kelompok 3), dan tinggi (kelompok 1). Dari penelitian ini terlihat bahwa masih terdapat perbedaan IPM antar kabupaten, dimana perbedaan IPM ini cenderung terkelompok dan berdekatan antara satu dengan yang lainnya.

5 Daftar Pustaka

[1] M. P. Cosmin, C. M. Marian and M. Mihai, "An optimized version of the K-means clustering algorithm", *Proceedings of the 2014 Federated Conference on Computer Science and Information Systems*, vol. 5, no. 2, (2014), pp. 695-699.

[2] N. Chen, and H. Zhang, "An Optimizing Algorithm of Non-Linear K-Means Clustering", *International Journal of Database Theory and Application*, vol. 9, No.4 (2016), pp.97-108.

[3] F. T. Grigorios and C. L. Aristidis, "The Global Kernel k-Means Algorithm for Clustering in Feature Space", *IEEE Transactions on Neural Networks*, vol. 20, No. 7, (2009), pp. 1181-1194.

[4] G. Armano and M. R. Farmani, "Multiobjective clustering analysis using particle swarm optimization," Expert Syst. Appl., vol. 55, pp. 184–193, 2016.

[5] W. Ida, A. A. Yudha, R. Asyrofa, F. M. Wayan, "Klastering Nasabah Bank Berdasarkan Tingkat Likuiditas Menggunakan Hybrid Particle Swarm Optimization dengan K-Means", *Jurnal Ilmiah Teknologi dan Informasi ASIA (JITIKA)*, vol. 10, No. 2, (2016).

[6] Ding. C, He. X, "K-means Clustering via Principal Component Analysis", *ICML '04 Proceedings of the twenty-first international conference on Machine learning*, page 29.

[7] Setiyono. B, Mukhlas. I, "Kajian Algoritma GDBScan, Clarans dan Cure untuk Spatial Clustering", *LIMITS - Journal of Mathematics and its Applications*, Vol. 2, No. 2, (2005).

[8] Bin Jiang, "Spatial Clustering for Mining Knowledge in Support of Generalization Process in GIS ICA", *Workshop of Geographic Infoemation Science*, (2004).

[9] Rosita. A, Purnanto. Y, Soelaiman. R, "Implementasi Algoritma Particle Swarm untuk Menyelesaikan Sistem Persamaan Nonlinear", *Jurnal Teknik ITS* Vol. 1, (Sept, 2012).

[10] Eka. P. L and Paidi. H, "Analisis Pertumbuhan Ekonomi dan Indeks Pembangunan Manusia (IPM) Provinsi-provinsi di Indonesia (Metode Kointegrasi)", *Jurnal Ekonomi dan Pembangunan, Fakultas Ekonomi, Universitas Sumatera Utara*, vol 3, No. 7, (2015).

[11] Setiawan. M. B, Hakim. A, "Indeks Pembangunan Manusia Indonesia", *Jurnal Economia*

- *Kajian Ilmiah Ekonomi dan Bisnis,* Vol 9 No 1 (2013).

[12] Badan Pusat Statistik, "Indeks Peembangunan Manusia 2014 - Metode Baru", Jakarta: Badan Pusat Statistik (2014).