Tugas 2 Pembelajaran Mesin

Fauzan Firdaus – 1301164317 – IF 40-04

1. Soal Nomor 1

Jawaban :

Berikut adalah contoh kasus dari K-Means Clustering. Jumlah cluster yang digunakan adalah 2 cluster (C1 dan C2), dengan 4 contoh sample data sebagai berikut.

Sample Data	Atribut 1	Atribut 2	Cluster
1	70	50	?
2	50	60	?
3	40	70	?
4	80	20	?

Pertama-tama, yang harus dilakukan adalah menghitung centroid dengan memilih masing-masing 1 sampel data tersebut secara random untuk setiap cluster. Misalkan dipilih data 1 untuk cluster 1 dan 2 untuk cluster 2. Dengan menggunakan *Euclidean Distance*, didapatkan centroid sebagai berikut.

Cluster / Centroid	C1	C2
1 (70,50)	0	22,36
2 (50,60)	23,36	0

Jarak data terdekat dari centroid 1 ke cluster C1 bernilai 0, dan centroid 2 cluster C2 bernilai 0, maka dari itu centroid C1 adalah data 1 dan centroid C2 adalah C2. Hal yang selanjutnya harus dilakukan adalah menghitung jarak pada data selanjutnya (data ke 3) dengan centroid yang telah ada. Dengan menggunakan *Euclidean Distance*, didapatkan kelompok cluster untuk data ke 3 sebagai berikut.

Sample Data	Euclidean Distance		Kelompok Cluster
	Cluster 1	Cluster 2	
3 (40, 70)	36.06	***14.14***	2

Selanjutnya adalah melakukan update nilai centroid yang baru dengan persamaan sebagai berikut

$$Atribut1CentroidBaru = \frac{Atribut1CentroidLama + Atribut1CurrentData}{2}$$

$$Atribut2CentroidBaru = \frac{Atribut2CentroidLama + Atribut2CurrentData}{2}$$

CurrentData yang dimaksud adalah atribut pada sample data terakhir yang telah dicari clusternya. Contohnya, karena data 3 adalah data terakhir yang telah dicari clusternya, maka data 3 lah yang dimaksud *CurrentData*. Didapatkan atribut centroid yang baru sebagai berikut.

Cluster	X	Y
C1	70	50
C2	40	65

Tahap selanjutnya adalah melakukan proses clustering lagi untuk sisa data yang ada seperti contoh dalam mencari cluster untuk data sampel ke 3. Didapatkan lah cluster ke 4 data sebagai berikut.

Sample Data	X	Y	Cluster
1	70	50	C1
2	50	60	C2
3	40	70	C2
4	80	20	C1

Kesimpulan dari contoh kasus
Kelebihan : Mengelompokkan data relatif efisien dan cepat, dan mudah diimplementasikan.
Kekurangan : Inisialisasi centroid dilakukan secara random, namun berdampak hasil cluster yang tidak terjamin benar benar optimal. Meskipun cepat, tidak terjamin akurat.

2. Soal Nomor 2

Jawaban :

Agglomerative Hierarchical Clustering adalah salah satu metode clustering dengan mengelompokan data ke dalam sebuah hirarki cluster dengan tujuan merangkum dan merepresentasikan data secara ringkas agar mudah di visualisasikan. Metode ini juga bekerja dengan menganggap setiap data memiliki 1 cluster yang berbeda, kemudian secara iterative isi dari tiap cluster bertambah (penggabungan klaster) sesuai ketentuan yang ada. Contoh kasus nya adalah terdapat 3 data sampel sebagai berikut.

Sample Data	Atribut 1	Atribut 2
1	3	2
2	1	1
3	1	3
4	2	2

Langkah pertama adalah menghitung jarak antar data ke setiap data yang ada. Bisa menggunakan *Euclidean Distance.* Didapatkan tabel jarak setiap data sebagai berikut.

Sample Data	1	2	3	4
1	0	2,24	2,24	1
2	2,24	0	2	1,42
3	2,24	2	0	1,42
4	1	1,42	1,42	0

Setelah mendapatkan jarak masing-masing dari setiap data, langkah selanjutnya adalah melakukan pengelompokkan (clustering) dengan menggabungkan data dalam 1 klaster. Contoh pertama proses pengelompokkan adalah sebagai berikut.

Cluster	Sample Data
C1	Data 1 & Data 4
C2	Data 2
C3	Data 3

Berikut adalah contoh hasil akhir dari proses pengelompokkan dengan jumlah klaster 2.

Cluster	Sample Data
C1	Data 1, Data 2, & Data 4
C2	Data 3

Laporan Analisis Sistem Clustering untuk Mencari Jumlah Klaster yang Optimal Menggunakan Metode Algoritma Self-Organizing Maps.

A. Abstrak

Diketahui sebuah dataset tanpa label sebanyak 600 data. Di dalamnya terdapat 2 atribut. Dari 600 data tersebut, akan dicari jumlah klaster / pembagian data yang optimal. Untuk menentukan jumlah klaster yang optimal, dicari nilai ***Within-cluster Sum of Squares*** **(WCSS)** dalam setiap K klaster. Penentuan jumlah klaster yang optimal menggunakan metode ***elbow***. Algoritma clustering yang digunakan pada program ini adalah algoritma ***Self-Organizing Maps*** **(SOM).** Bahasa yang digunakan untuk program ini adalah bahasa **Matlab**, dengan versi tools adalah **R2018A**.

B. Self-Organizing Maps

Sedikit bahasan mengenai Self-Organizing Maps, Self-Organizing Maps atau yang sering disingkat sebagai SOM adalah salah satu algoritma / metode yang ada pada ANN (*Artificial Neural Network*) yang digunakan untuk clustering data dengan tujuan untuk mereduksi dimensionalitas mengguna-kan diskrit dari *input space* yang disebut peta / map.

Dalam program ini, SOM digunakan untuk clustering dataset sebanyak 600 data, untuk dapat menghasilkan jumlah klaster yang optimal.

C. Within-cluster Sum of Square (WCSS)

WCSS adalah ukuran variabilitas pengamatan dalam setiap klaster. Nilai WCSS sendiri didapatkan dari jumlah kuadrat dari masing-masing jarak antara titik pada input space dengan neuron / centroid klasternya. Persamaan WCSS:

$$WCSS = \sum jarak(P_i, C_1)^2 + \sum jarak(P_i, C_1)^2 + \cdots + \sum jarak(P_i, C_n)^2$$

Dengan *P* adalah titik pada input space, C adalah satu klaster K pada sebuah data, *i* adalah iterasi data input space, dan *n* adalah jumlah klaster yang ada pada suatu data.

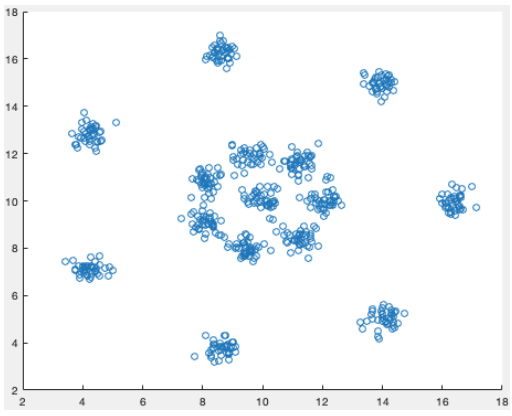
D. Metode Elbow

Metode Elbow adalah suatu metode yang digunakan untuk menghasilkan informasi dalam menentukan jumlah cluster terbaik dengan cara melihat persentase hasil perbandingan antara jumlah cluster yang akan membentuk siku pada suatu titik. Metode ini memberikan ide/gagasan dengan cara memilih nilai cluster dan kemudian menambah nilai cluster tersebut untuk dijadikan model data dalam penentuan cluster terbaik. Untuk persentase perbandingan yang digunakan adalah berdasarkan *Within-cluster Sum of Squares* (WCSS) yang telah didapatkan sebelumnya pada setiap klaster.

E. Mekanisme Program

Program ini dibuat dalam bahasa **Matlab.** Penulis tidak menyarankan untuk mengeksusi program yang telah penulis bangun dengan versi tools matlab dibawah **R2018A.**

Program clustering ini bertujuan untuk mencari jumlah klaster yang optimum pada sebuah data yang berjumlah 600 *input space*. Dari setiap data tersebut terdapat 2 atribut. Berikut adalah grafik persebaran data pada 600 data tersebut.



Secara garis besar, program ini mencari nilai WCSS dari setiap hasil proses klasterisasi. Maksudnya adalah program ini melakukan proses klaster ketika total klasternya = 1, 2, 3, dst. Jadi, dari setiap total klaster memiliki nilai WCSS yang berbeda. Penulis melakukan proses klaster sampai dengan total klasternya berjumlah 15. Hal ini berdasarkan kebutuhan datasetnya. Terlihat bahwa persebaran data memiliki total 15 area titik-titik yang berkumpul. Jika hasil dari proses klaster dengan total klaster 1 sampai 15 dinyatakan masih kurang optimal, maka penulis akan melakukan proses klaster lebih dari 15.

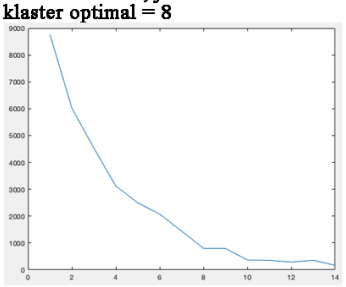
Dalam pencarian WCSS untuk setiap total klaster yang berbeda, terdapat proses klaster menggunakan metode SOM. Proses klaster dilakukan dengan tujuan untuk menghasilkan 3 nilai sebagai outputnya. Yang pertama adalah bobot atau *weight* dari setiap neuron. Yang kedua adalah *grid* yang merepresentasikan posisi akhir neuron tersebut berada pada grafik. Dan yang terakhir adalah WCSS dari hasil proses klaster tersebut.

Setelah mendapatkan WCSS dari setiap total klaster, maka yang dilakukan selanjutnya adalah membuat visualisasi data dari WCSS yang telah didapatkan. Dalam program ini, terdapat 15 nilai WCSS untuk divisualkan dalam bentuk grafik. Setelah grafik tersebut muncul, dilakukanlah analisis oleh user sendiri bagaimana dan berapa total klaster yang optimal berdasarkan metode elbow. Seperti yang telah didefinisikan sebelumnya, cara kerja atau teknik dalam metode elbow adalah menganalisis grafik dan titik optimalnya berada pada lengkungan garis yang membentuk siku (perubahan data yang ekstrim).

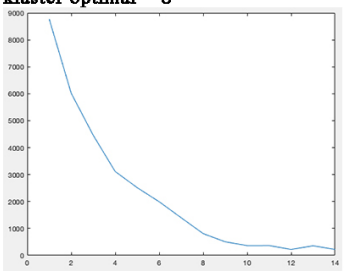
F. Hasil atau Output Program

Penulis melakukan 5 percobaan running/eksekusi program dengan hasil grafik WCSS sebagai berikut.

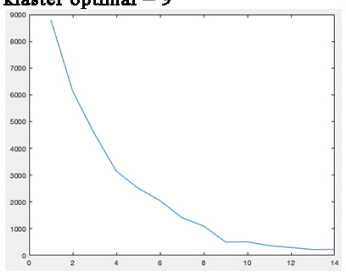
Percobaan ke-1, jumlah klaster optimal = 8



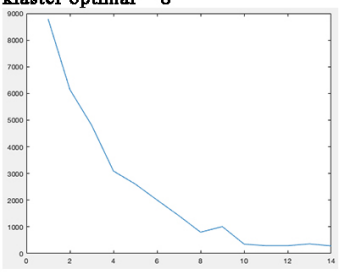
Percobaan ke-2, jumlah klaster optimal = 8



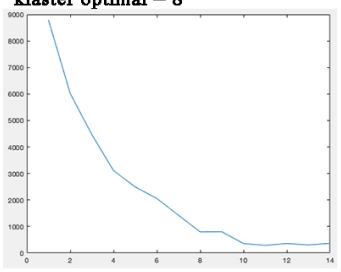
Percobaan ke-3, jumlah klaster optimal = 9



Percobaan ke-4, jumlah klaster optimal = 8



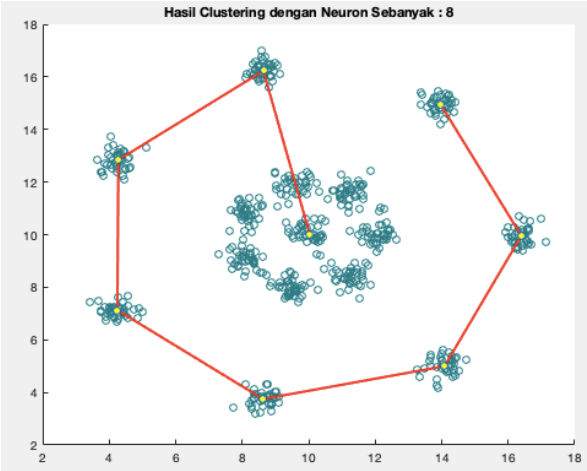
Percobaan ke-5, jumlah klaster optimal = 8



Dalam 5 percobaan tersebut, dapat diringkas menjadi sebagai berikut :

- Percobaan 1 : 8 total klaster
- Percobaan 2 : 8 total klaster
- Percobaan 3 : 9 total klaster
- Percobaan 4 : 8 total klaster
- Percobaan 5 : 8 total klaster

Total klaster dari setiap percobaan yang sering muncul adalah 8. Maka dari itu, dapat disimpulkan bahwa untuk dataset ini, total klaster yang optimal adalah sebanyak 8 klaster. Berikut adalah contoh dari hasil proses klasterisasi dengan total klaster sebanyak 8.



CATATAN : *Dalam laporan ini tidak terdapat penjelasan setiap baris kode program karena penjelasannya sudah berada dalam program tersebut.*