

MODUL 12

ANALISIS KLASTER 2 (*SINGLE LINKAGE*)



CAPAIAN PEMBELAJARAN

1. Praktikan mampu mengimplementasikan penggunaan software R untuk analisis klaster dengan metode *single linkage*.



KEBUTUHAN ALAT/BAHAN/SOFTWARE

1. Komputer
2. Software R



DASAR TEORI

A. PENDAHULUAN

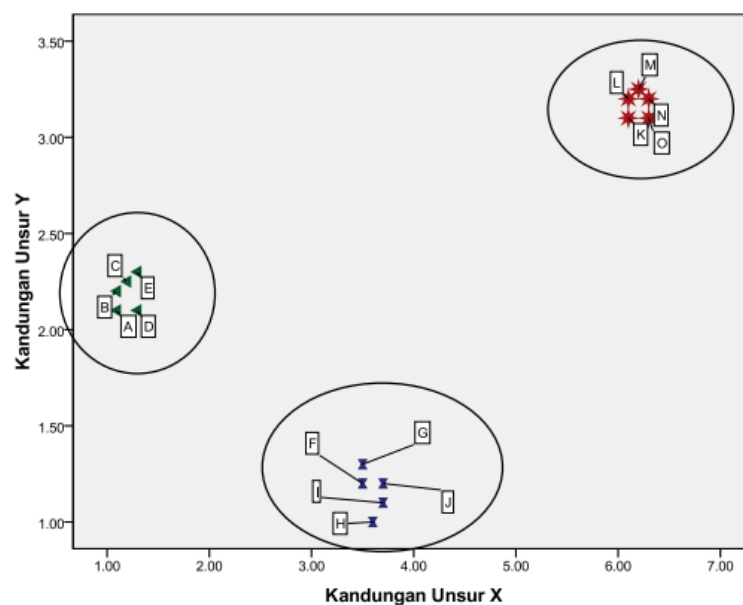
Analisis klaster (*cluster analysis*) merupakan suatu teknik statistika yang digunakan untuk mengelompokkan (*group*) sekumpulan objek (manusia, produk, tanaman, dan sebagainya) ke dalam beberapa klaster. Perhatikan bahwa suatu objek hanya bisa masuk atau bergabung dalam satu klaster. Beberapa objek yang berada dalam satu klaster cenderung saling mirip, namun cenderung berbeda terhadap objek-objek yang berada dalam klaster lainnya. Sebagai contoh perhatikan data pada Tabel A. Berdasarkan data pada Tabel A, objek yang diteliti adalah batu, sebanyak 15 batu. Masing-masing batu memiliki kadar X dan kadar Y.

Tabel A

Batu	kadar X	kadar Y
A	1.1	2.1
B	1.1	2.2
C	1.2	2.25
D	1.3	2.1
E	1.3	2.3
F	3.5	1.2
G	3.5	1.3
H	3.6	1
I	3.7	1.1
J	3.7	1.2
K	6.1	3.1
L	6.1	3.2
M	6.2	3.25
N	6.3	3.2
O	6.3	3.1

Gambar A memberikan gambaran yang cukup jelas untuk pengelompokkan (cluster). Berdasarkan Gambar B, jika dibentuk klaster sebanyak 3, maka:

1. Batu A, B, C, D, dan E berada dalam satu klaster, misalkan klaster pertama.
2. Batu F, G, H, I, dan J berada dalam satu klaster, misalkan klaster kedua.
3. Batu K, L, M, N, dan O berada dalam satu klaster, misalkan klaster ketiga.



Gambar A

Perhatikan bahwa batu A, B, C, D, dan E cenderung mirip, karena berada di dalam satu klaster, yakni klaster pertama, namun cenderung berbeda terhadap batu-batu yang berada dalam klaster yang berbeda. Tiga klaster yang tersaji dalam Gambar

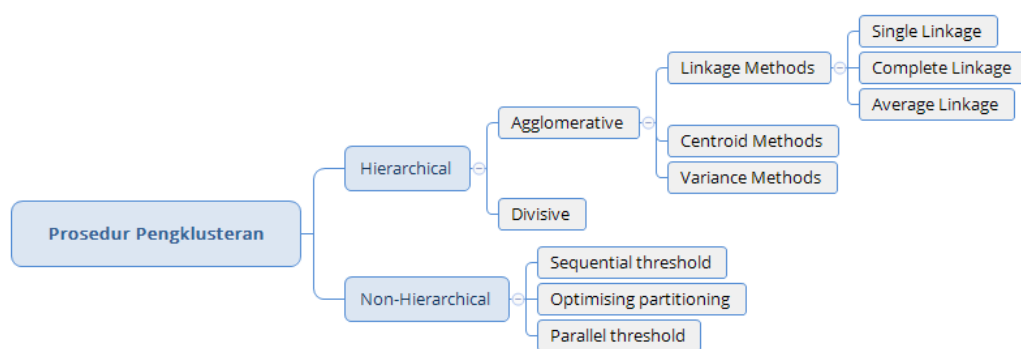
A melibatkan dua variabel klaster, yakni variabel kadar X (sumbu horizontal) dan kadar Y (sumbu vertikal).

B. UKURAN KEMIRIPAN (*Measure of Similarity*)

Ukuran kemiripan dilakukan dengan mencari jarak dari masing-masing objek. Dari jarak yang diperoleh tersebut selanjutnya dikelompokkan dengan cara memilih jarak yang paling dekat. Pencarian jarak dapat menggunakan *Euclidean Distance* yang dirumuskan sebagai berikut:

$$d_{1,2} = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

C. PROSEDUR PENKLASTERAN



Gambar C

Pada metode *hierarchical*, jumlah klaster belum atau tidak diketahui sebelumnya, sementara pada metode *non-hierarchical* jumlah klaster ditetapkan terlebih dahulu, sebelum melakukan pengklasteran objek. Modul ini akan mempraktikkan clustering menggunakan metode *hierarchical*, khususnya metode **Single Linkage**.



PRAKTIK

Praktik 1 (Input dan Import Data)

Andaikan diberikan data dari 15 jenis batu beserta kandungan unsur X dan Y pada Tabel 1 berikut:

Tabel 1

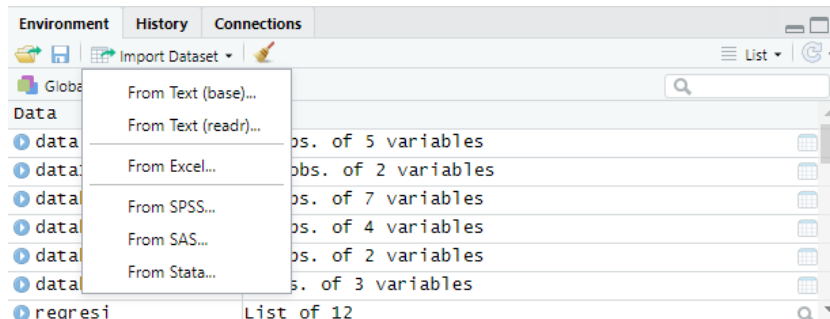
Batu	kadar X	kadar Y
A	1,1	2,1
B	1,1	2,2
C	1,2	2,25
D	1,3	2,1
E	1,3	2,3
F	3,5	1,2
G	3,5	1,3
H	3,6	1
I	3,7	1,1
J	3,7	1,2
K	6,1	3,1
L	6,1	3,2
M	6,2	3,25
N	6,3	3,2
O	6,3	3,1

Inputkan data Tabel 1 pada Ms. Excel, kemudian simpan dengan nama **data2.xlsx** (Gambar 1).

	A	B	C
1	Batu	kadar X	kadar Y
2	A	1,1	2,1
3	B	1,1	2,2
4	C	1,2	2,25
5	D	1,3	2,1
6	E	1,3	2,3
7	F	3,5	1,2
8	G	3,5	1,3
9	H	3,6	1
10	I	3,7	1,1
11	J	3,7	1,2
12	K	6,1	3,1
13	L	6,1	3,2
14	M	6,2	3,25
15	N	6,3	3,2
16	O	6,3	3,1

Gambar 1

Selanjutnya, pada Gambar 2, lakukan import file **data2.xlsx** pada R dengan cara lihat jendela **Environment**, pilih **Import Dataset**, pilih **From Excel**.



Gambar 2

Hasil dari langkah ini terlihat pada Gambar 3.

The screenshot shows the R Environment window with the imported data displayed as a table. The table has 5 rows and 3 columns: 'Batu', 'kadar X', and 'kadar Y'. The data is as follows:

	Batu	kadar X	kadar Y
1	A	1.1	2.10
2	B	1.1	2.20
3	C	1.2	2.25
4	D	1.3	2.10
5	E	1.3	2.30

Showing 1 to 6 of 15 entries, 3 total columns

Gambar 3

Praktik 2 (Memprioritaskan Variabel)

Pada **data2** terdapat 3 variabel (Batu, kadar X, dan kadar Y), selanjutnya hanya akan diambil 2 variabel (kadar X dan kadar Y), dengan menghilangkan variabel Batu. Data yang terbentuk selanjutnya dinamai **simpan_data2**.

Input

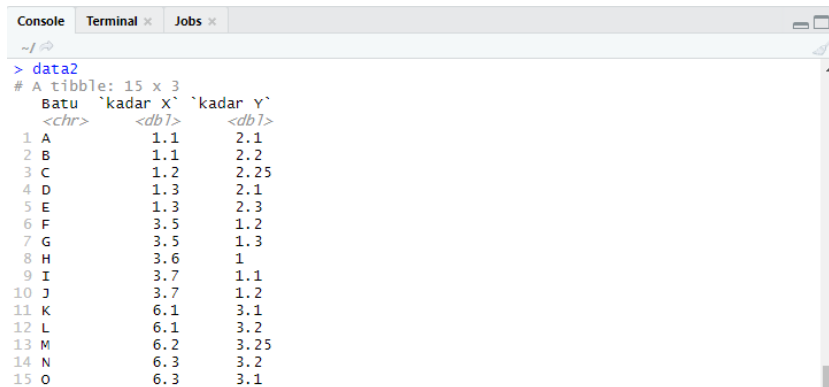
```

1 data2
2
3 #mengkonversi data2 menjadi simpan_data
4 simpan_data=data2
5 simpan_data
6
7 #menghilangkan variabel batu dan mengkonversi menjadi simpan_data2
8 variabel = c("kadar X","kadar Y")
9 simpan_data2=simpan_data[variabel]
10 simpan_data2

```

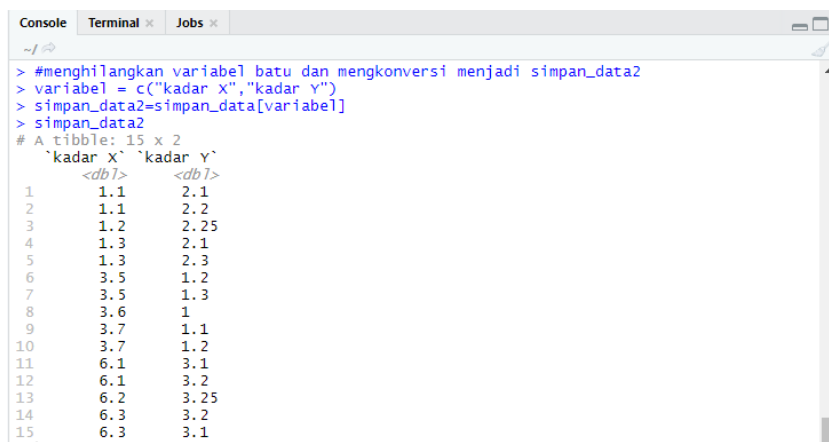
Gambar 4

Output



```
Console Terminal Jobs
~/
> data2
# A tibble: 15 x 3
  Batu   kadar_x` `kadar_y`
  <chr>   <dbl>   <dbl>
1 A       1.1     2.1
2 B       1.1     2.2
3 C       1.2     2.25
4 D       1.3     2.1
5 E       1.3     2.3
6 F       3.5     1.2
7 G       3.5     1.3
8 H       3.6     1
9 I       3.7     1.1
10 J      3.7     1.2
11 K      6.1     3.1
12 L      6.1     3.2
13 M      6.2     3.25
14 N      6.3     3.2
15 O      6.3     3.1
```

Gambar 5



```
Console Terminal Jobs
~/
> #menghilangkan variabel batu dan mengkonversi menjadi simpan_data2
> variabel = c("kadar_x", "kadar_y")
> simpan_data2=simpan_data[variabel]
> simpan_data2
# A tibble: 15 x 2
  `kadar_x` `kadar_y`
  <dbl>   <dbl>
1       1.1     2.1
2       1.1     2.2
3       1.2     2.25
4       1.3     2.1
5       1.3     2.3
6       3.5     1.2
7       3.5     1.3
8       3.6     1
9       3.7     1.1
10      3.7     1.2
11      6.1     3.1
12      6.1     3.2
13      6.2     3.25
14      6.3     3.2
15      6.3     3.1
```

Gambar 6

Analisis

Gambar 5 adalah output/tampilan dari data **data2** (3 variabel), sementara Gambar 6 adalah output/tampilan dari data **simpan_data2** (2 variabel).

Praktik 3 (Mencari Euclid Distance dan Kuadratnya)

Pencarian nilai jarak Euclid dari data **simpan_data2** dan hasilnya ditampilkan dalam bentuk matriks. Kemudian dilanjutkan dengan mencari kuadrat jarak Euclid tersebut.

Input

```
12 #mencari euclid distance
13 jarak=dist(as.matrix(simpan_data2))
14 jarak
15
16 #mencari kuadrat euclid distance
17 jarak_pangkat_dua=jarak*jarak
18 jarak_pangkat_dua
19
34:1 (Top Level) R Script
```

Gambar 7

Output

```
Console Terminal Jobs
> #mencari euclid distance
> jarak=dist(as.matrix(simpan_data2))
> jarak
      1      2      3      4      5      6      7      8
2  0.1000000
3  0.1802776 0.1118034
4  0.2000000 0.2236068 0.1802776
5  0.2828427 0.2236068 0.1118034 0.2000000
6  2.5632011 2.6000000 2.5283394 2.3769729 2.4596748
7  2.5298221 2.5632011 2.4884734 2.3409400 2.4166092 0.1000000
8  2.7313001 2.7730849 2.7060118 2.5495098 2.6419690 0.2236068 0.3162278
9  2.7856777 2.8231188 2.7518176 2.6000000 2.6832816 0.2236068 0.2828427 0.1414214
10 2.7513633 2.7856777 2.7115494 2.5632011 2.6400758 0.2000000 0.2236068 0.2236068
11 5.0990195 5.0803543 4.9731781 4.9030603 4.8662100 3.2202484 3.1622777 3.2649655
12 5.1195703 5.0990195 4.9912423 4.9244289 4.8836462 3.2802439 3.2202484 3.3301652
13 5.2280493 5.2069665 5.0990195 5.0331402 4.9912423 3.3900590 3.3305405 3.4383862
14 5.3150729 5.2952809 5.1877259 5.1195703 5.0803543 3.4409301 3.3837849 3.4828150
15 5.2952809 5.2773099 5.1703482 5.0990195 5.0635956 3.3837849 3.3286634 3.4205263
      9      10      11      12      13      14
2
3
4
5
6
7
8
9
10 0.1000000
11 3.1240999 3.0610456
12 3.1890437 3.1240999 0.1000000
13 3.2973474 3.2330326 0.1802776 0.1118034
14 3.3421550 3.2802439 0.2236068 0.2000000 0.1118034
15 3.2802439 3.2202484 0.2000000 0.2236068 0.1802776 0.1000000
> |
```

Gambar 8

```
Console Terminal Jobs
> #mencari kuadrat euclid distance
> jarak_pangkat_dua=jarak*jarak
> jarak_pangkat_dua
      1      2      3      4      5      6      7      8      9     10
2  0.0100
3  0.0325 0.0125
4  0.0400 0.0500 0.0325
5  0.0800 0.0500 0.0125 0.0400
6  6.5700 6.7600 6.3925 5.6500 6.0500
7  6.4000 6.5700 6.1925 5.4800 5.8400 0.0100
8  7.4600 7.6900 7.3225 6.5000 6.9800 0.0500 0.1000
9  7.7600 7.9700 7.5725 6.7600 7.2000 0.0500 0.0800 0.0200
10 7.5700 7.7600 7.3525 6.5700 6.9700 0.0400 0.0500 0.0500 0.0100
11 26.0000 25.8100 24.7325 24.0400 23.6800 10.3700 10.0000 10.6600 9.7600 9.3700
12 26.2100 26.0000 24.9125 24.2500 23.8500 10.7600 10.3700 11.0900 10.1700 9.7600
13 27.3325 27.1125 26.0000 25.3325 24.9125 11.4925 11.0925 11.8225 10.8725 10.4525
14 28.2500 28.0400 26.9125 26.2100 25.8100 11.8400 11.4500 12.1300 11.1700 10.7600
15 28.0400 27.8500 26.7325 26.0000 25.6400 11.4500 11.0800 11.7000 10.7600 10.3700
      11      12      13      14
2
3
4
5
6
7
8
9
10
11
12 0.0100
13 0.0325 0.0125
14 0.0500 0.0400 0.0125
15 0.0400 0.0500 0.0325 0.0100
> |
```

Gambar 9

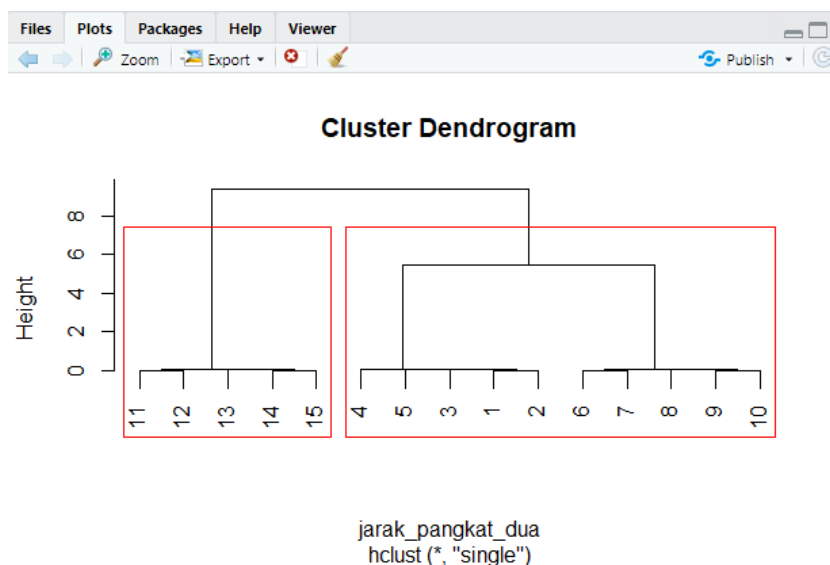
Praktik 4 (Membagi Data Menjadi 2 Klaster dengan Metode Single Linkage)

Input

```
19 | k1aster1=hclust(jarak_pangkat_dua,method = "single")
20 | plot(k1aster1)
21 | dua_kluster=cutree(k1aster1, k=2)
22 | rect.hclust(k1aster1, k=2, border = "red")
23 |
```

Gambar 10

Output



Gambar 11

Anlisis

Dari hasil output pada Gambar 11, terlihat bahwa jika data dibagi menjadi 2 klaster, maka: klaster 1 (11, 12, 13, 14, 15) dan klaster 2 (4, 5, 3, 1, 2, 6, 7, 8, 9, 10)

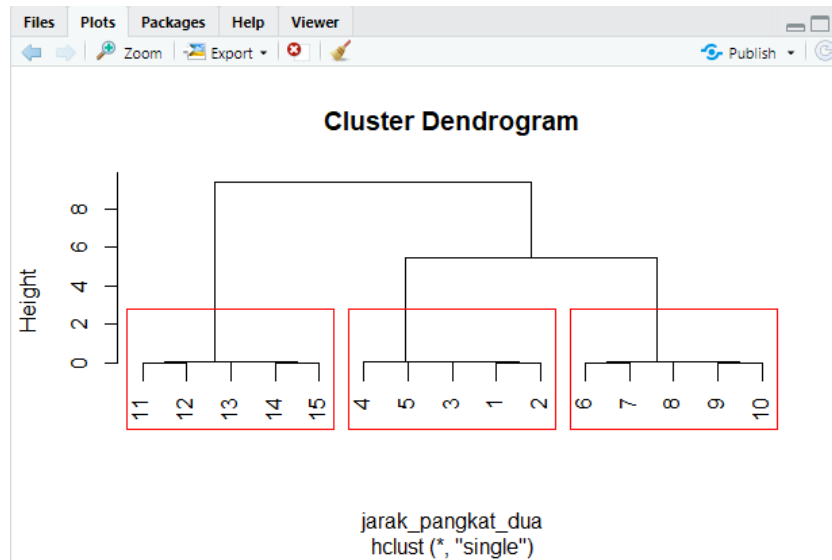
Praktik 5 (Membagi Data Menjadi 3 Klaster dengan Metode Single Linkage)

Input

```
24 | k1aster2=hclust(jarak_pangkat_dua,method = "single")
25 | plot(k1aster2)
26 | dua_kluster=cutree(k1aster2, k=3)
27 | rect.hclust(k1aster2, k=3, border = "red")
28 |
```

Gambar 12

Output



Gambar 13

Anlisis

Dari hasil output pada Gambar 13, terlihat bahwa jika data dibagi menjadi 3 klaster, maka: klaster 1 (11, 12, 13, 14, 15), klaster 2 (4, 5, 3, 1, 2), dan klaster 3 (6, 7, 8, 9, 10)

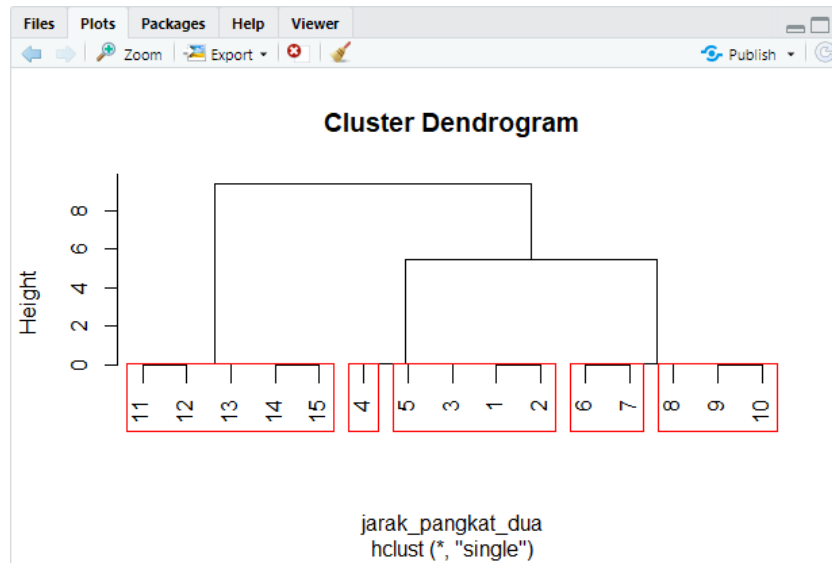
Praktik 6 (Membagi Data Menjadi 5 Klaster dengan Metode Single Linkage)

Input

```
29  
30 k1aster4=hclust(jarak_pangkat_dua,method = "single")  
31 plot(k1aster4)  
32 dua_kluster=cutree(k1aster4, k=5)  
33 rect.hclust(k1aster4, k=5, border = "red")
```

Gambar 14

Output



Gambar 15

Analisis

Dari hasil output pada Gambar 15, terlihat bahwa jika data dibagi menjadi 3 kluster, maka: kluster 1 (11, 12, 13, 14, 15), kluster 2 (4), kluster 3 (5, 3, 1, 2), kluster 4 (6, 7), dan kluster 5 (8, 9, 10)



LATIHAN

DATA	X	Y
M1	2	5,0
M2	2	5,5
M3	5	3,5
M4	6,5	2,2
M5	7	3,3
M6	3,5	4,8
M7	4	4,5

Lakukan clustering data tersebut menggunakan metode **Single Linkage** menjadi 2 dan 3 kluster, kemudian analisislah!



TUGAS

DATA	X	Y
1	1,57	5,38
2	0,47	4,28
3	2,04	1,78
4	5,64	1,84
5	3,15	0,73
6	3,78	0,54
7	2,74	1,08

Lakukan clustering data tersebut menggunakan metode **Single Linkage** menjadi 2 dan 3 kluster, kemudian analisislah!



REFERENSI

- [1] Gio, P.U., Effendie, A.R. 2017. Belajar Bahasan Pemrograman R (Dilengkapi Cara Membuat Aplikasi Olah Data Sederhana dengan R Shiny). Medan: USU Press.