

Pokok Bahasan XIV

Analisis Cluster

Kode Pokok Bahasan: TIK.RPL03.001.011.01

Deskripsi Pokok Bahasan:

Membahas tentang Association Rule pada R dengan dataset yang diberikan.

No	Elemen Kompetensi	Indikator Kinerja	Jml Jam	Hal
1	Memahami cara implementasi Analisis Cluster	1.1 Mampu memahami cara implementasi Analisis Cluster dengan Hierarchical Cluster, Dendogram, k-mean clustering	1	12

TUGAS PENDAHULUAN

Hal yang harus dilakukan dan acuan yang harus dibaca sebelum praktikum :

1. Menginstal R pada PC masing-masing praktikan.
2. Menginstal R Studio pada PC masing-masing praktikan.

DAFTAR PERTANYAAN

1. Berikan penjelasan mengenai apa itu “Analisis Cluster”?

Jawab : Analisis klaster adalah metode statistik dalam penelitian yang memungkinkan peneliti untuk mengelompokkan atau mengelompokkan sekumpulan objek ke dalam kluster-kluster kecil namun berbeda yang berbeda karakteristiknya dari kluster-kluster lain yang berbeda.

2. Jelaskan pengertian Hierarchical Cluster, Dendogram, k-mean clustering?

Jawab :

Hierarchical Cluster = metode pengelompokan data yang dimulai dengan setiap satu pengamatan sebagai clusternya sendiri kemudian terus mengelompokkan pengamatan ke dalam kelompok yang semakin besar.

Dendogram = sejenis diagram treelike yang digunakan secara hirarkis kekelompokan. Ini merupakan daftar semua peserta pada satu akhir dan kemudian mengarahkan cabang keluar dari peserta tersebut yang serupa dan menghubungkan mereka dengan simpul yang mewakili sebuah cluster.

k-mean clustering = Pengklasteran k rata-rata adalah algoritme untuk membagi n pengamatan menjadi k kelompok sedemikian hingga tiap pengamatan termasuk ke dalam kelompok dengan rata-rata terdekat. Hasilnya adalah pembagian pengamatan ke dalam sel-sel Voronoi. Pengklasteran k rata-rata meminimalkan ragam dalam klaster.



TEORI SINGKAT

Analisis cluster sering juga disebut analisis gerombol. Analisis cluster adalah analisis statistika yang bertujuan untuk mengelompokkan data sedemikian sehingga data yang berada dalam kelompok yang sama mempunyai sifat yang relatif homogen daripada data yang berada dalam kelompok yang berbeda.

Ditinjau dari hal-hal yang dikelompokkan, analisis cluster dibagi menjadi dua macam, yaitu pengelompokan observasi dan pengelompokan variabel. Dalam pembahasan ini, pengelompokan yang dilakukan adalah pengelompokan observasi.

LAB SETUP

Hal yang harus disiapkan dan dilakukan oleh praktikan untuk menjalankan praktikum modul ini.

1. Menginstall library yang dibutuhkan untuk mengerjakan modul.
2. Menjalankan R Studio.

ELEMEN KOMPETENSI I**Deskripsi:**

Memahami cara implementasi Analisis Cluster

Kompetensi Dasar:

1. Mampu memahami cara implementasi Analisis Cluster dengan Hierarchical Cluster, Dendogram, k-mean clustering.

Latihan 1.1.1**Penjelasan Singkat :**

Pada latihan ini anda akan diminta untuk menerapkan Analisi Cluster pada data yang diberikan.

Langkah-Langkah Praktikum:

Dataset 1 :

observation	Income	Education
s1	5	5
s2	6	6
s3	15	14
s4	16	15
s5	25	20
s6	30	19

Hierarchical Cluster :

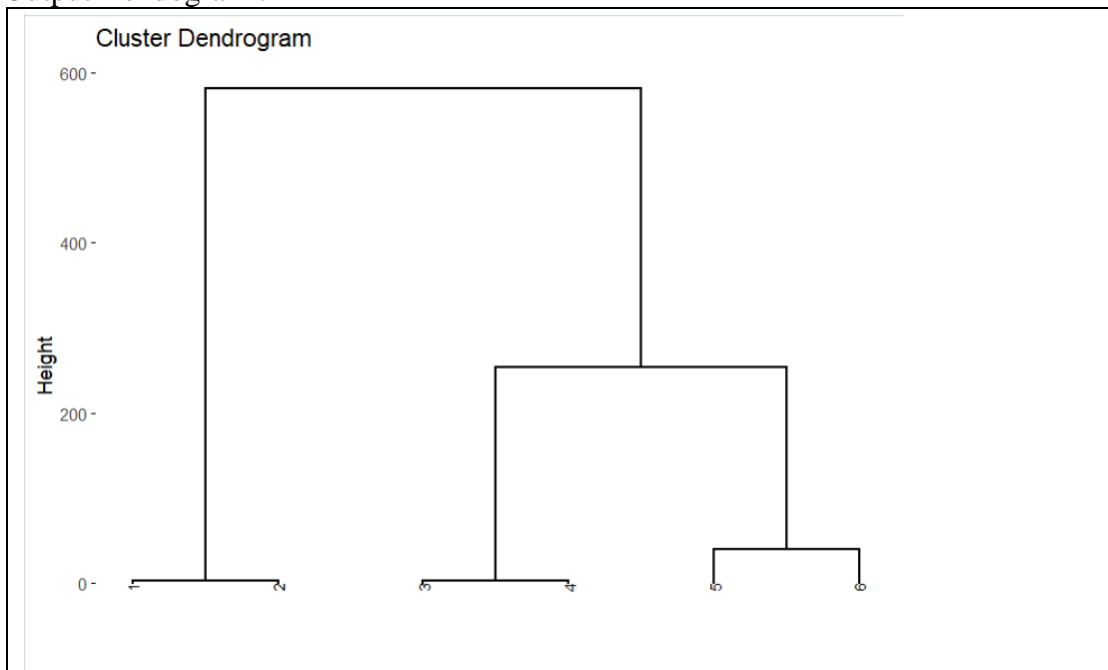
```
library("readxl")
#my_data <- read_excel(file.choose(), sheet = "", range = "")
my_data <- read_excel("E:/Cluster analysis/clusterPage127.xlsx", na
= "-")
print(my_data)
str(my_data)
d.euc <- dist(my_data)
d.sqeucl <- d.euc^2
cluster<- hclust(d = d.sqeucl, method = "centroid")
library("factoextra")
fviz_dend(cluster, cex = 0.6)

#if(!require(devtools)) install.packages("devtools")
#devtools::install_github("kassambara/factoextra")
```

Output :

```
> str(my_data)
'data.frame': 6 obs. of 3 variables:
 $ observation: chr "s1 " "s2 " "s3 " "s4 " ...
 $ Income : num 5 6 15 16 25 30
 $ Education : num 5 6 14 15 20 19
> d.euc <- dist(my_data)
Warning message:
In dist(my_data) : NAs introduced by coercion
> d.euc
      1      2      3      4      5
2  1.732051
3 16.477257 14.747881
4 18.207141 16.477257  1.732051
5 30.618622 28.905017 14.282857 12.609520
6 35.092734 33.429029 19.364917 17.832555  6.244998
> d.sqeuc <- d.euc^2
> d.sqeuc
      1      2      3      4      5
2    3.0
3 271.5 217.5
4 331.5 271.5    3.0
5 937.5 835.5 204.0 159.0
6 1231.5 1117.5 375.0 318.0    39.0
> cluster<- hclust(d = d.sqeuc, method = "centroid")
> fviz_dend(cluster, cex = 0.6)
> |
```

Output Dendrogram :



Hierarchical Cluster in Python :



```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

df = pd.read_excel("data1.xlsx")
df.head()

data= df.iloc[:,[1,2]]
data.head()

from sklearn.preprocessing import normalize
data_scaled = normalize(data)
data_scaled = pd.DataFrame(data_scaled, columns=data
.columns)
data_scaled.head()

import scipy.cluster.hierarchy as shc
plt.figure(figsize=(10,7))
plt.title("Dendrograms")
dend = shc.dendrogram(shc.linkage(data_scaled, metho
d = 'ward'))
```

Output :




✓ 0s [2] `df = pd.read_excel("data1.xlsx")`
`df.head()`

	observation	Income	Education
0	s1	5	5
1	s2	6	6
2	s3	15	14
3	s4	16	15
4	s5	25	20

✓ 0s [3] `data= df.iloc[:,[1,2]]`
`data.head()`

	Income	Education
0	5	5
1	6	6
2	15	14
3	16	15
4	25	20

✓ 0s  `from sklearn.preprocessing import normalize`
`data_scaled = normalize(data)`
`data_scaled = pd.DataFrame(data_scaled, columns=data.columns)`
`data_scaled.head()`

	Income	Education
0	0.707107	0.707107
1	0.707107	0.707107
2	0.731055	0.682318
3	0.729537	0.683941
4	0.780869	0.624695

Output Dendrogram :





Script k-mean clustering

```
> klaster <- kmeans(dataku[, 2:3], 3, nstart = 20)
> klaster
> table(klaster$cluster, dataku$X)
```

Output :

```
> klaster <- kmeans(my_data[, 2:3], 3, nstart = 20)
> klaster
K-means clustering with 3 clusters of sizes 2, 2, 2

Cluster means:
  Income Education
1  15.5      14.5
2   5.5       5.5
3  27.5      19.5

Clustering vector:
[1] 2 2 1 1 3 3

Within cluster sum of squares by cluster:
[1] 1 1 13
(between_SS / total_SS = 97.9 %)

Available components:
[1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss" "betweenss"
[7] "size"         "iter"         "ifault"

> table(klaster$cluster, my_data$observation)

   s1 s2 s3 s4 s5 s6
1   0  0  1  1  0  0
2   1  1  0  0  0  0
3   0  0  0  0  1  1
> |
```

Script k-mean clustering

```
import matplotlib.pyplot as plt
from sklearn.cluster import KMeans
import pandas as pd
```

```
df2 = pd.read_excel("data1.xlsx")
df2.head()

data2= df.iloc[:,[1,2]]
data2.head()

kmeans =KMeans(n_clusters=3).fit(data2)
centroids = kmeans.cluster_centers_
print(centroids)

plt.scatter(df.iloc[:,1], df.iloc[:,2], c = kmeans.labels_.astype(float), s =50, alpha = 0.5)
plt.scatter(centroids[:,0], centroids[:,1], c='red', s=50)
plt.show
```

Output :


```
[7] df2 = pd.read_excel("data1.xlsx")
df2.head()
```

	observation	Income	Education
0	s1	5	5
1	s2	6	6
2	s3	15	14
3	s4	16	15
4	s5	25	20

```
data2= df.iloc[:,[1,2]]
data2.head()
```

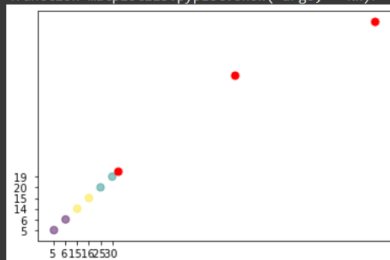
	Income	Education
0	5	5
1	6	6
2	15	14
3	16	15
4	25	20

```
[9] kmeans =KMeans(n_clusters=3).fit(data2)
centroids = kmeans.cluster_centers_
print(centroids)
```

```
[[ 5.5  5.5]
 [27.5 19.5]
 [15.5 14.5]]
```

```
plt.scatter(df.iloc[:,1], df.iloc[:,2], c = kmeans.labels_.astype(float), s =50, alpha = 0.5)
plt.scatter(centroids[:,0], centroids[:,1], c='red', s=50)
plt.show
```

```
<function matplotlib.pyplot.show(*args, **kw)>
```



Dataset 2 :

Supplier	DAR	DSR	Quality
S1	96.81	73.85	100
S2	99.64	65.79	100
S3	96.5	71.63	62.86
S4	99.349	79.38	96.86
S5	100	88.24	100
S6	71.4	60.7	71.4
S7	99.827	84.54	79.43
S8	99.9	82.98	88
S9	99.058	95.16	98
S10	98.1	90.77	81.43
S11	99.574	79.66	95.14
S12	99.606	80.4	71.1

Script k-mean clustering

```

dataku=read.delim("clipboard")
data_latih <-dataku[,c(2,3,4)]
data_latih_matrix <-
as.matrix(scale(data_latih))
library(factoextra) library(NbClust)
lnb <- NbClust(data_latih, distance = "euclidean", min.nc = 2,max.nc = 8, method
=
"complete", index ="all")

km.res=kmeans(data_latih, 3,
nstart=25) km.res library(dplyr)
fviz_cluster(km.res, data = data_latih, geom = "point",stand = FALSE, frame.type
=
"norm") fviz_cluster(km.res,
data = data_latih)

```

Output :



```

> dataku
  Supplier    DAR   DSR Quality
1       S1    96.810 73.85  100.00
2       S2    99.640 65.79  100.00
3       S3    96.500 71.63   62.86
4       S4    99.349 79.38   96.86
5       S5   100.000 88.24  100.00
6       S6    71.400 60.70   71.40
7       S7    99.827 84.54   79.43
8       S8    99.900 82.98   88.00
9       S9    99.058 95.16   98.00
10      S10    98.100 90.77   81.43
11      S11    99.574 79.66   95.14
12      S12    99.606 80.40   71.10

> data_latih <- dataku[,c(2,3,4)]
> data_latih_matrix <- as.matrix(scale(data_latih))
> lnb <- NbClust(data_latih, distance = "euclidean", min.nc = 2, max.nc = 8, method = "complete", index = "all")
*** : The Hubert index is a graphical method of determining the number of clusters.
      In the plot of Hubert index, we seek a significant knee that corresponds to a
      significant increase of the value of the measure i.e the significant peak in Hubert
      index second differences plot.

*** : The D index is a graphical method of determining the number of clusters.
      In the plot of D index, we seek a significant knee (the significant peak in Dindex
      second differences plot) that corresponds to a significant increase of the value of
      the measure.

*****
* Among all indices:
* 5 proposed 2 as the best number of clusters
* 7 proposed 3 as the best number of clusters
* 1 proposed 4 as the best number of clusters
* 4 proposed 5 as the best number of clusters
* 6 proposed 8 as the best number of clusters

***** Conclusion *****

* According to the majority rule, the best number of clusters is 3

*****
Plot Zoom

```

```

> km.res=kmeans(data_latih, 3, nstart=25)
> km.res
K-means clustering with 3 clusters of sizes 4, 7, 1

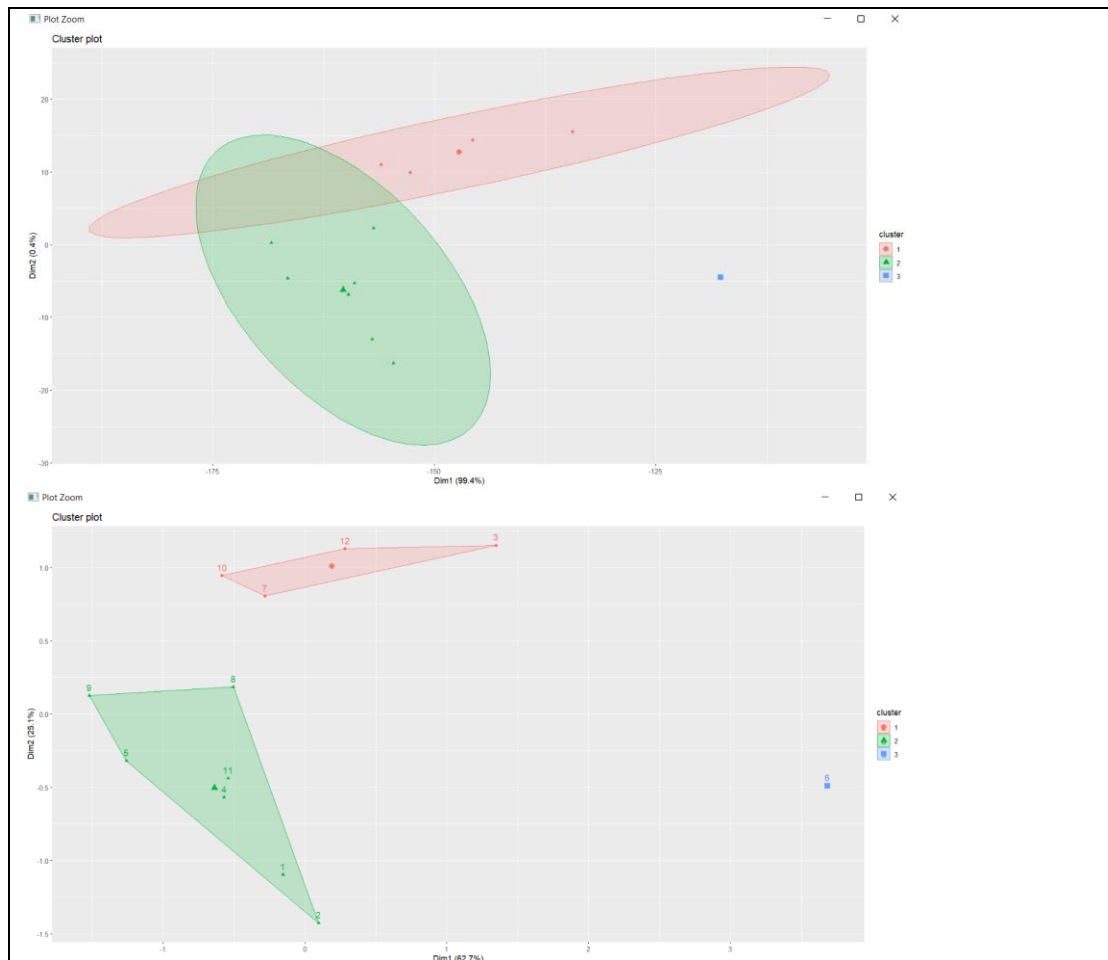
Cluster means:
      DAR   DSR Quality
1 98.50825 81.83500 73.70500
2 99.19014 80.72286 96.85714
3 71.40000 60.70000 71.40000

Clustering vector:
[1] 2 2 1 2 2 3 1 2 2 1 2 1

Within cluster sum of squares by cluster:
[1] 417.3477 662.7461 0.0000
 (between_SS / total_SS = 71.5 %)

Available components:
[1] "cluster" "centers" "totss" "withinss" "tot.withinss" "betweenss" "size"
[8] "iter" "ifault"
>

```



Script k-mean clustering in Python

```
import matplotlib.pyplot as plt
from sklearn.cluster import KMeans
import pandas as pd

df2 = pd.read_excel("data2.xlsx")
df2.head()

data2= df.iloc[:, [1,2,3]]
data2.head()

kmeans =KMeans(n_clusters=3).fit(data2)
centroids = kmeans.cluster_centers_
print(centroids)

plt.scatter(df.iloc[:,1], df.iloc[:,2], c = kmeans.labels_.astype(float), s =50, alpha = 0.5)
plt.scatter(centroids[:,0], centroids[:,1], c='red', s=50)
plt.show
```

Output :

```
✓ [12] df2 = pd.read_excel("data2.xlsx")
0s df2.head()
```

	Supplier	DAR	DSR	Quality
0	S1	96.81	73.85	100
1	S2	99.64	65.79	100
2	S3	96.5	71.63	62.86
3	S4	99.349	79.38	96.86
4	S5	100	88.24	100

```
✓ [14] data2= df2.iloc[:,[1,2,3]]
0s data2.head()
```

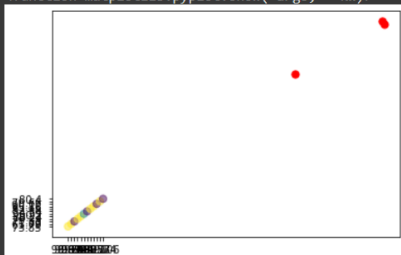
	DAR	DSR	Quality
0	96.81	73.85	100
1	99.64	65.79	100
2	96.5	71.63	62.86
3	99.349	79.38	96.86
4	100	88.24	100

```
✓ [15] kmeans =KMeans(n_clusters=3).fit(data2)
0s centroids = kmeans.cluster_centers_
print(centroids)
```

```
[[98.50825  81.835  73.705 ]
 [71.4      60.7    71.4   ]
 [99.19014286 80.72285714 96.85714286]]
```

```
✓ plt.scatter(df2.iloc[:,1], df2.iloc[:,2], c = kmeans.labels_.astype(float), s =50, alpha = 0.5)
0s plt.scatter(centroids[:,0], centroids[:,1], c='red', s=50)
plt.show
```

```
<function matplotlib.pyplot.show(*args, **kw)>
```



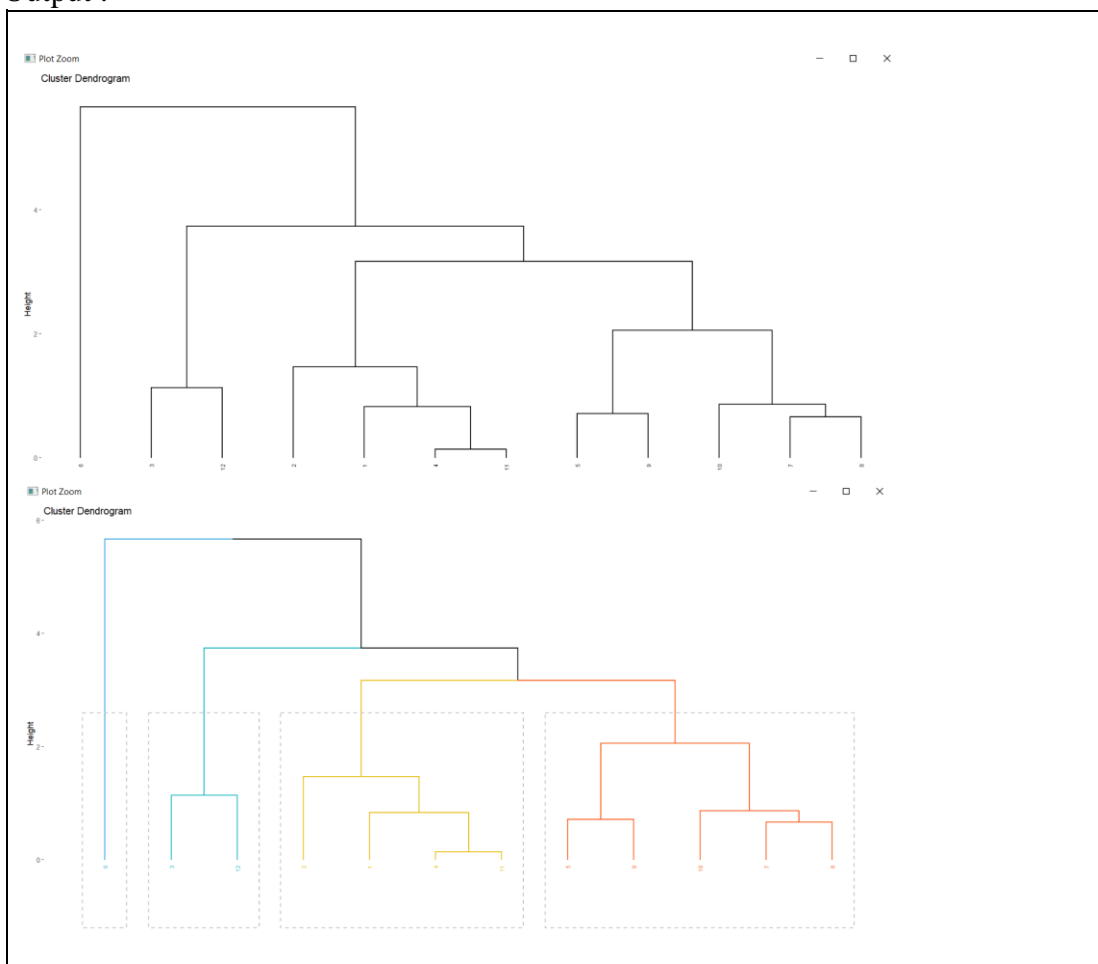
Script hierarchical clustering

```

dataku2=read.delim("clipboard")
df=scale(dataku2[,2:4]) res.dist <- dist(df,
method = "euclidean") res.hc <- hclust(d
= res.dist, method = "ward.D2")
library("factoextra") fviz_dend(res.hc, cex
= 0.5)
fviz_dend(res.hc, k = 4, cex = 0.5, k_colors = c("#2E9FDF", "#00AFBB",
"#E7B800", "#F
C4E07"), color_labels_by_k = TRUE, rect = TRUE)

```

Output :



Script hierarchical clustering in Python

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

df = pd.read_excel("data2.xlsx")
df.head()

data= df.iloc[:, [1,2,3]]
data.head()

from sklearn.preprocessing import normalize
data_scaled = normalize(data)
data_scaled = pd.DataFrame(data_scaled, columns=data.columns)
data_scaled.head()

import scipy.cluster.hierarchy as shc
plt.figure(figsize=(10,7))
plt.title("Dendrograms")
dend = shc.dendrogram(shc.linkage(data_scaled, method = 'ward')
)
```

Output :



```
[19] df = pd.read_excel("data2.xlsx")  
df.head()
```

	Supplier	DAR	DSR	Quality
0	S1	96.81	73.85	100
1	S2	99.64	65.79	100
2	S3	96.5	71.63	62.86
3	S4	99.349	79.38	96.86
4	S5	100	88.24	100

```
data= df.iloc[:,[1,2,3]]  
data.head()
```

	DAR	DSR	Quality
0	96.81	73.85	100
1	99.64	65.79	100
2	96.5	71.63	62.86
3	99.349	79.38	96.86
4	100	88.24	100

```
from sklearn.preprocessing import normalize  
data_scaled = normalize(data)  
data_scaled = pd.DataFrame(data_scaled, columns=data.columns)  
data_scaled.head()
```

	DAR	DSR	Quality
0	0.614422	0.468702	0.634668
1	0.639764	0.422422	0.642076
2	0.711514	0.528142	0.463480
3	0.621499	0.496578	0.605928
4	0.599908	0.529359	0.599908



Interpretasi :

Dendrogram menggambarkan bagaimana data-data yang disediakan bisa digabungkan menjadi suatu dataset yang lebih mudah dibaca, semisal ada data yang berhubungan langsung digambarkan dari mulai data primer sampai data sekunder

Tugas :

Diberikan data harga beberapa komoditas berbagai pasar di Jakarta. Lakukan analisis cluster menggunakan metode hierarchical clustering dan k-means serta interpretasikan maknanya.

Pasar	Beras	Jeruk	Minyak
Pasar Senen Blok III - VI	13000	18000	13000
Pasar Jembatan Merah	11000	25000	11500
Pasar Sunter Podomoro	12000	30000	12000
Pasar Rawa Badak	12000	23000	12000
Pasar Grogol	11000	25000	11000

Pasar Minggu	13125	25000	10000
--------------	-------	-------	-------



Pasar Mayestik	12000	20000	13000
Pasar Pramuka	12000	25000	14000
Pasar Kramat Jati	11700	25000	12000
Pasar Jatinegara	11000	22000	12000
Pasar Perumnas Klender	10900	20000	13000
Pasar Pulo Gadung	10700	22000	12000
Pasar Pal Meriam	10000	20000	12000
Pasar Ciplak	11200	17000	12000
Pasar Cijantung	11000	23000	12000
Pasar Cibubur	11500	20000	12500
Pasar Ujung Menteng	11500	30000	11000
Pasar Tanah Abang Blok A-G	11000	25000	12000
Pasar Petojo Ilir	11000	23000	12000
Pasar Gondangdia	13000	20000	12000
Pasar Paseban	12000	20000	11000
Pasar Cempaka Putih	12000	19000	12000
Pasar Johar Baru	12000	25000	13000
Pasar Baru Metro Atom	11500	20000	11000
Pasar Kebayoran Lama	12500	23000	13000
Pasar Cipete	10500	25000	12500
Pasar Pondok Labu	12000	25000	13000
Pasar Lenteng Agung	11500	28000	11500
Pasar Mampang Prapatan	13000	17000	12000
Pasar Tebet Barat	12000	20000	11000

Pasar Rumpit	13000	20000	13000
Pasar Tomang Barat	13000	24000	12000
Pasar Pos Pengumben	11400	22000	12000
Pasar Pal Merah	10800	25000	12000
Pasar Jembatan Lima	10000	35000	12500
Pasar Kelapa Gading	12000	25000	12500
Pasar Pademangan Timur	12000	23000	12000
Pasar Kalibaru	12000	20000	11000
Pasar Koja Baru	12000	18000	12500

K-means Clustering

```
library(factoextra)
library(NbClust)
library(dplyr)

dataku=read.delim("clipboard")
dataku
data_latih <-dataku[,c(2,3,4)]
data_latih_matrix <- as.matrix(scale(data_latih))
lnc <- NbClust(data_latih, distance = "euclidean", min.nc = 2,max.nc = 8,
method = "complete", index ="all")

km.res=kmeans(data_latih, 3, nstart=25)
km.res

fviz_cluster(km.res, data = data_latih, geom = "point",stand = FALSE,
frame.type = "norm")
fviz_cluster(km.res, data = data_latih)
```

```

> data_latih <- dataku[,c(2,3,4)]
> data_latih_matrix <- as.matrix(scale(data_latih))
> lnb <- NbClust(data_latih, distance = "euclidean", min.nc = 2, max.nc = 8, method = "complete", index = "all")
*** : The Hubert index is a graphical method of determining the number of clusters.
      In the plot of Hubert index, we seek a significant knee that corresponds to a
      significant increase of the value of the measure i.e the significant peak in Hubert
      index second differences plot.

*** : The D index is a graphical method of determining the number of clusters.
      In the plot of D index, we seek a significant knee (the significant peak in Dindex
      second differences plot) that corresponds to a significant increase of the value of
      the measure.

*****
* Among all indices:
* 5 proposed 2 as the best number of clusters
* 12 proposed 3 as the best number of clusters
* 2 proposed 4 as the best number of clusters
* 3 proposed 5 as the best number of clusters
* 2 proposed 8 as the best number of clusters

***** Conclusion *****

* According to the majority rule, the best number of clusters is 3

*****
> km.res=kmeans(data_latih, 3, nstart=25)
> km.res
K-means clustering with 3 clusters of sizes 20, 15, 4

Cluster means:
      Beras      Jeruk      Minyak
1 11586.25 24000.00 12125.00
2 11940.00 19266.67 12066.67
3 11250.00 30750.00 11750.00

Clustering vector:
[1] 2 1 3 1 1 1 2 1 1 1 2 1 2 2 1 2 3 1 1 2 2 2 1 2 1 1 1 3 2 2 2 1 1 1 3 1 1 2 2

Within cluster sum of squares by cluster:
[1] 51359344 37662667 30250000
(between_SS / total_SS = 79.8 %)

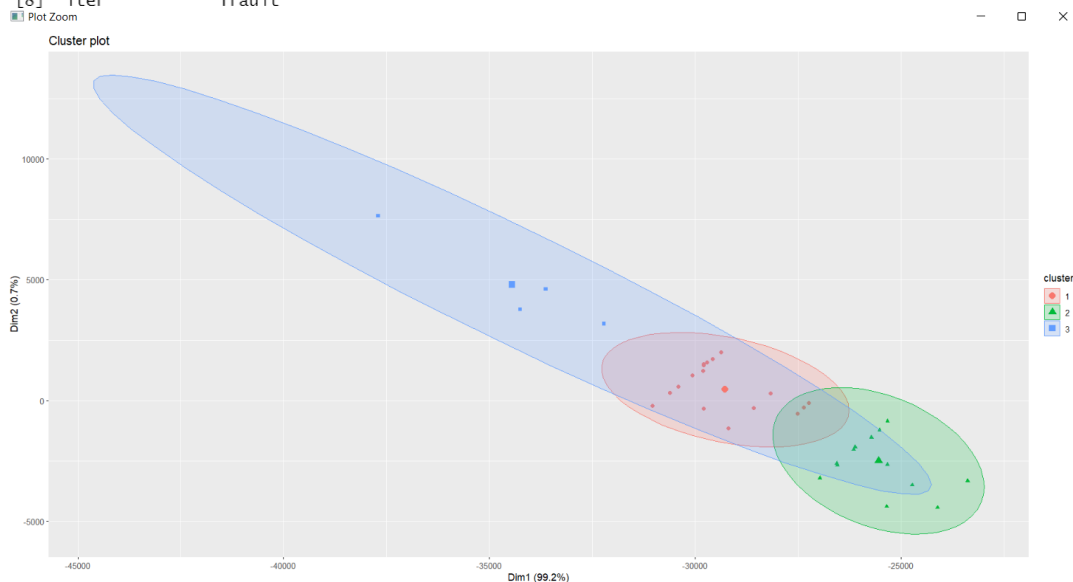
```

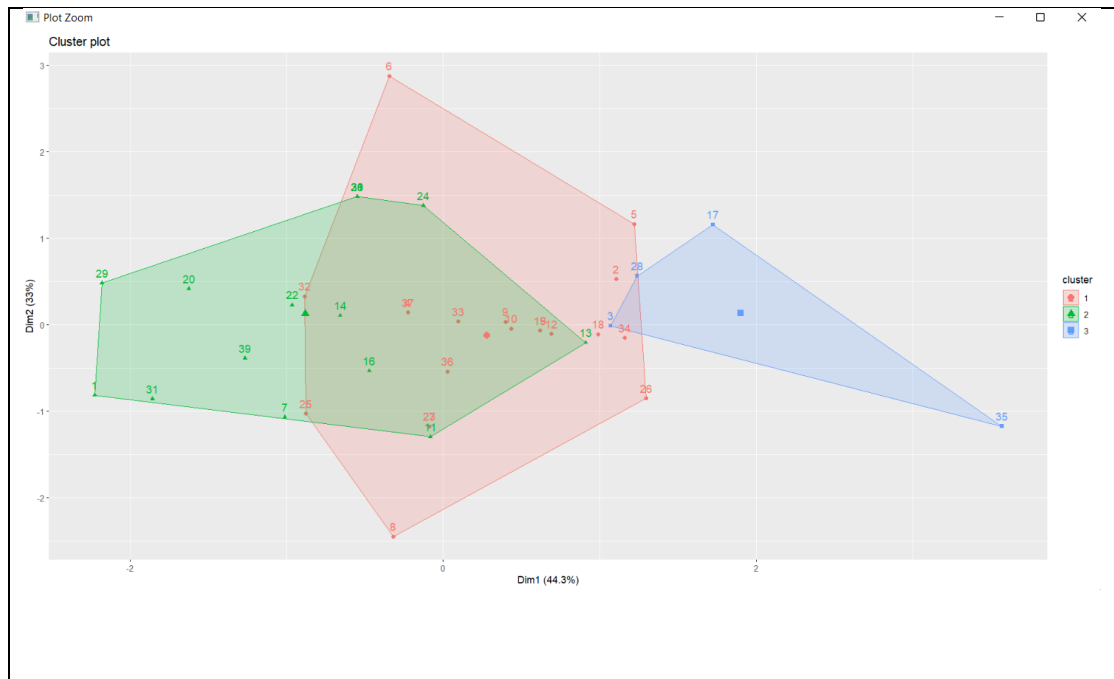
Available components:

```

[1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss" "betweenss"    "size"
[8] "iter"         "ifault"

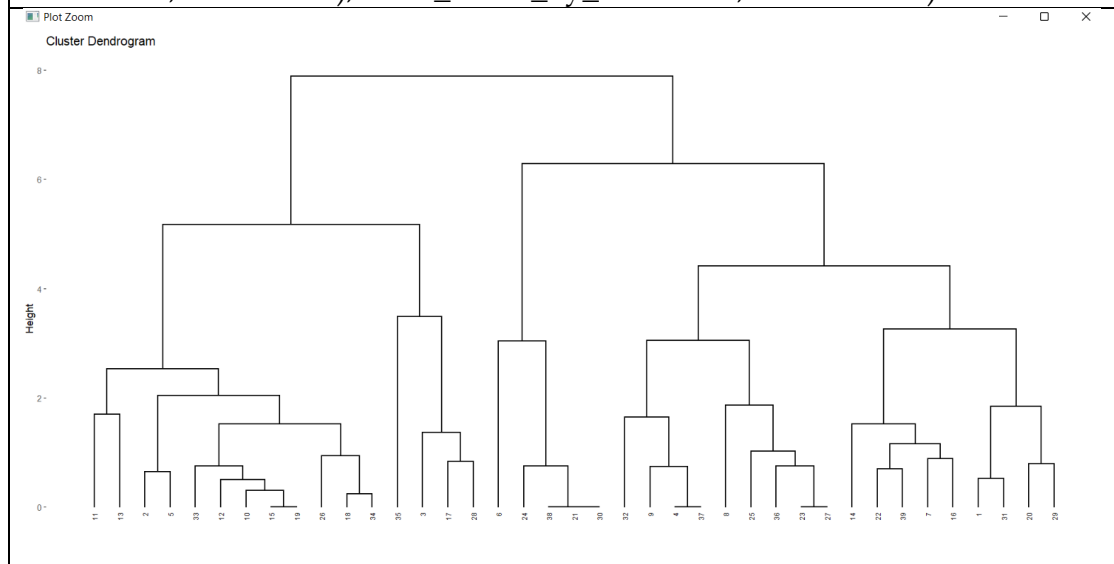
```

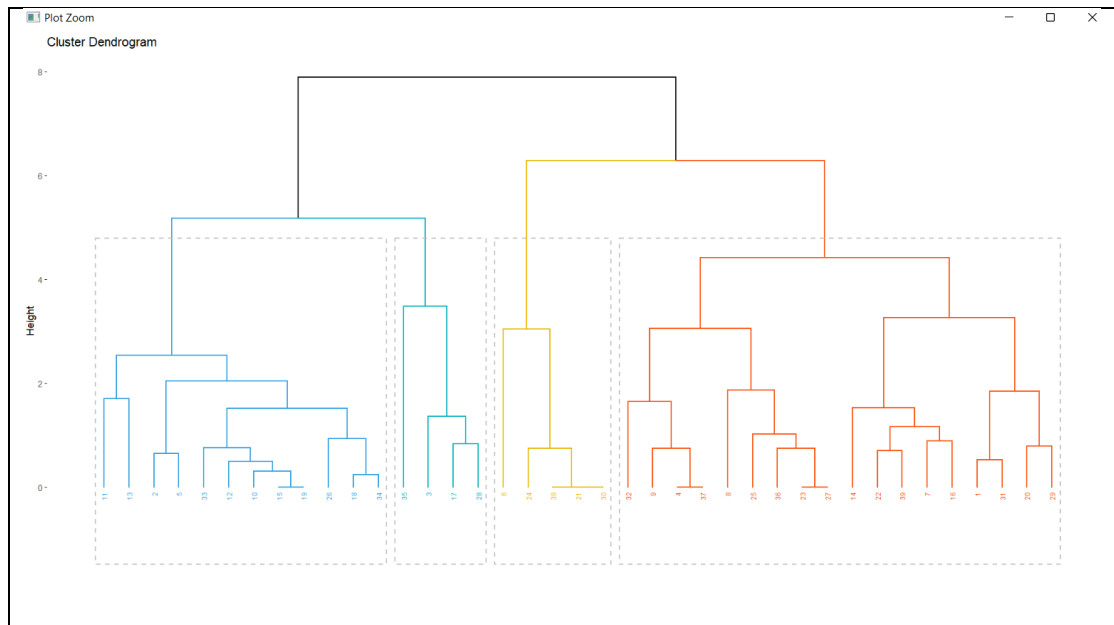




Hierarchical Clustering

```
df=scale(dataku[,2:4])
res.dist <- dist(df, method = "euclidean")
res.hc <- hclust(d = res.dist, method = "ward.D2")
library("factoextra")
fviz_dend(res.hc, cex = 0.5)
fviz_dend(res.hc, k = 4, cex = 0.5, k_colors = c("#2E9FDF", "#00AFBB",
"#E7B800", "#FC4E07"), color_labels_by_k = TRUE, rect = TRUE)
```





Intepretasi

Interpretasi dari data clustering diatas adalah dari sleuruh pasar yang ada bisa dilihat bahwa harga mayoritas mengikuti harga pasar yang ada, mulai dari beras, jagung, dan minyak semuanya dari pasar ke pasar masih mempunyai harga pasaran pada umumnya

CEK LIST

Elemen Kompetensi	No Latihan	Penyelesaian	
		Selesai	Tidak selesai
1	1.1.1	✓	
	1.1.2	✓	

FORM UMPAN BALIK

Elemen Kompetensi	Tingkat Kesulitan	Tingkat Ketertarikan	Waktu Penyelesaian dalam menit
Memahami cara implementasi Analisa Cluster	<input type="checkbox"/> Sangat Mudah	<input type="checkbox"/> Tidak Tertarik	30 menit
	<input type="checkbox"/> Mudah	<input type="checkbox"/> Cukup Tertarik	
	<input checked="" type="checkbox"/> ✓ Biasa	<input checked="" type="checkbox"/> ✓ Tertarik	

