Modul 06 - Clustering

Roni Yunis

17/09/2024

Pengantar

K-Means Clustering adalah algoritma *Unsupervised Learning* yang mencoba mengelompokkan data berdasarkan kesamaannya. *Unsupervised Learning* salah satu paradigma dalam machine learning di mana algoritma diprogram untuk mengekstraksi pola atau informasi dari data tanpa adanya label atau petunjuk eksplicit tentang keluaran yang diinginkan. Dengan kata lain, dalam unsupervised learning, model tidak diberikan label atau target spesifik untuk diprediksi atau dipelajari. Unsupervised learning sering digunakan ketika data tidak memiliki label atau ketika tujuan analisis adalah untuk menemukan pola yang mungkin tidak diketahui sebelumnya:

Contoh penggunaan unsupervised learning meliputi:

1. Rekomendasi Produk: Mengidentifikasi pola konsumen berdasarkan perilaku pembelian untuk memberikan rekomendasi produk tanpa memerlukan label spesifik untuk setiap pengguna.
2. Segmentasi Pelanggan: Mengelompokkan pelanggan berdasarkan preferensi dan perilaku pembelian tanpa memerlukan informasi label kelompok.
3. Analisis Teks: Mengelompokkan dokumen atau teks berdasarkan tema atau topik tanpa memerlukan label kategori untuk setiap dokumen.

K-Means Clustering

Dalam K-Means clustering, kita telah menentukan jumlah cluster yang kita ingin datanya dikelompokkan. Algoritma secara acak menetapkan setiap observasi ke cluster, dan menemukan pusat data dari setiap cluster. Kemudian, algoritma melakukan iterasi melalui dua langkah:

1. Tetapkan ulang titik data ke cluster yang sentroidnya paling dekat.
2. Hitung sentroid baru dari setiap cluster.

Kedua langkah ini diulangi sampai variasi cluster tidak dapat dikurangi lebih jauh. Variasi dalam cluster dihitung sebagai jumlah dari jarak Euclid (Euclidean) antara titik data dan sentroid cluster masing-masing.

Dalam kasus ini, kita akan mengklasterisasi informasi dari COVID 19 yang ada pada negara ASIA. Tujuan dari analisis ini adalah untuk melihat apakah pandemi COVID-19 di Negara ASIA bisa di klasterisasi berdasarkan atribut yang sudah ada. Dataset dalam kasus ini bisa di lihat atau diunduh di halaman web berikut https://www.worldometers.info/coronavirus/

Library (tidyverse) = dplyr + ggplot2 + lubridate

Load Packages

```
# Manipulasi data
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 4.3.3
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
# Visualisasi data
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.3.3
```

```
# Untuk melakukan klasterisasi
library(cluster)

# fungsi tambahan untuk klasterisasi dan visualisasi
library(factoextra)
```

```
## Welcome! Want to learn more? See two factoextra-related books at https://goo.gl/ve3WBa
```

Data Preparation

Menyiapkan data, data dalam kasus ini sudah di unduh sebelumnya jadi tinggal digunakan

```
corona <- read.table("data/covid19.dat")
head(corona)
```

```
##             TotCases TotDeath Recovered ActCases Cases1M Deaths1M TotTests
## Afghanistan    14525      249      1303    12973  373.91     6.41    37348
## Armenia         8927      127      3317     5483 3013.04    42.87    57081
## Azerbaijan      5246       61      3327     1858  517.80     6.02   294264
## Bahrain        10793       17      5826     4950 6365.23    10.03   309573
## Bangladesh     44608      610      9375    34623  271.10     3.71   297054
## Bhutan            43        0         6       37   55.78     0.00    17038
##               Tests1M       Pop ASEAN
## Afghanistan    961.43  38846163     0
## Armenia      19265.99   2962785     0
```

```
## Azerbaijan   29045.09  10131281     0
## Bahrain     182572.48   1695617     0
## Bangladesh    1805.29 164546795     0
## Bhutan       22102.44    770865     0
```

```
glimpse(corona)
```

```
## Rows: 49
## Columns: 10
## $ TotCases  <int> 14525, 8927, 5246, 10793, 44608, 43, 141, 125, 83001, 944, 7~
## $ TotDeath  <int> 249, 127, 61, 17, 610, 0, 2, 0, 4634, 17, 12, 4, 5185, 1573,~
## $ Recovered <int> 1303, 3317, 3327, 5826, 9375, 6, 138, 123, 78304, 790, 600, ~
## $ ActCases  <int> 12973, 5483, 1858, 4950, 34623, 37, 1, 2, 63, 137, 145, 43, ~
## $ Cases1M   <dbl> 373.91, 3013.04, 517.80, 6365.23, 271.10, 55.78, 322.57, 7.4~
## $ Deaths1M  <dbl> 6.41, 42.87, 6.02, 10.03, 3.71, 0.00, 4.58, 0.00, 3.22, 14.0~
## $ TotTests  <int> 37348, 57081, 294264, 309573, 297054, 17038, 19130, 20406, 0~
## $ Tests1M   <dbl> 961.43, 19265.99, 29045.09, 182572.48, 1805.29, 22102.44, 43~
## $ Pop       <int> 38846163, 2962785, 10131281, 1695617, 164546795, 770865, 437~
## $ ASEAN     <int> 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, ~
```

Dataset berikut berisi 49 baris dan 10 buah kolom.

Pada bagian ini kita akan melakukan klasterisasi berdasarkan atribut `TotCases`, `TotDeath`, `Recovered`, `ActCases` dan `Pop`. Sebelum kita melakukan klastering, kita akan setup terlebih dahulu dengan pendekatan K-Means Clustering.

1. Elemen-elemen pada matriks jarak antar negara yang digunakan adalah Jarak Euclid;
2. Matriks data yang dianalisis distandarisasi mempertimbangkan adanya rentang nilai yang lebar pada atribut jumlah penduduk; dan
3. Jumlah klaster optimal diperiksa dengan menggunakan metode yang dapat digunakan untuk mengidentifikasi jumlah klaster. Ada beberapa metode yang dapat digunakan, seperti metode Siluet, Statistik Gap, Elbow Method, dan lain-lain.

Note: *Silahkan anda pelajari secara mandiri konsep teoritis dari metode-metode untuk mengindentifikasi jumlah klaster tsb*

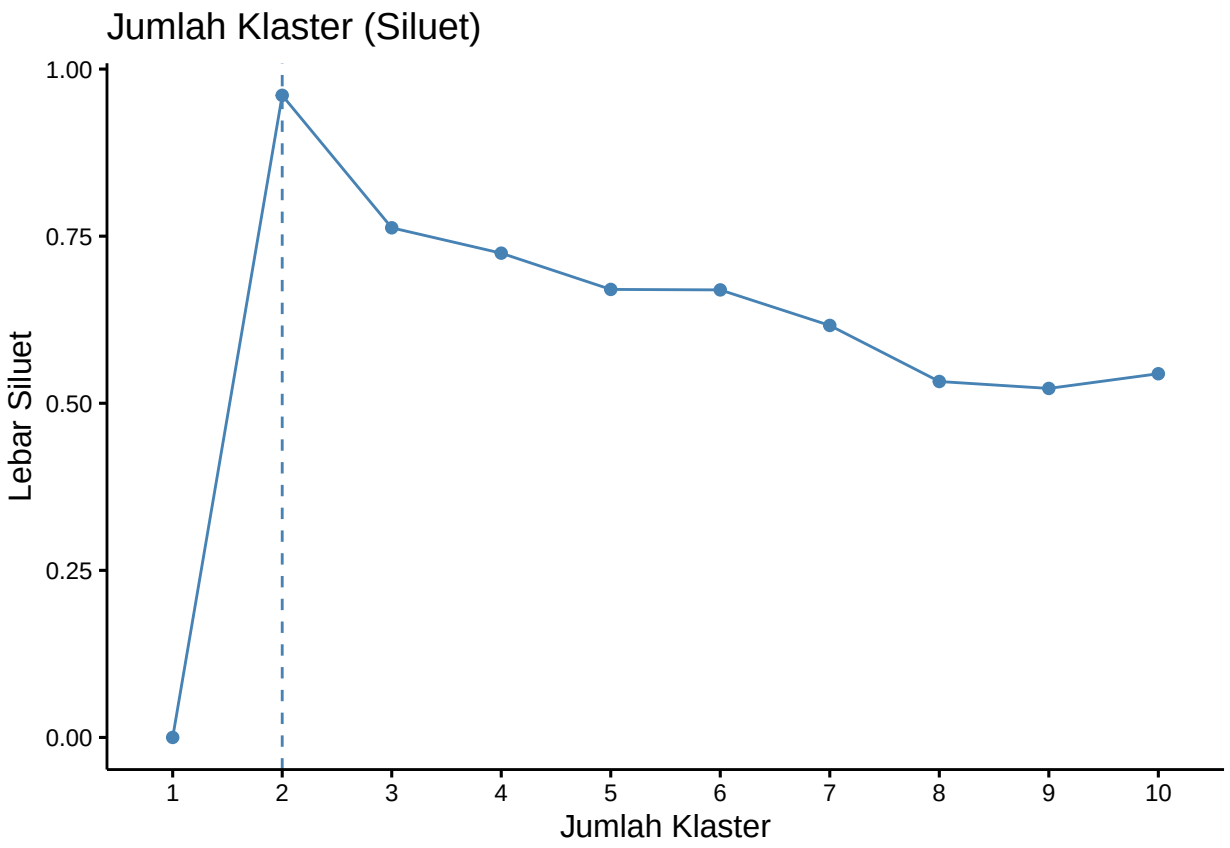
```
# Menampilkan nama atribut dalam objek corona yang akan di klasterisasi
set.seed(999) #memilih secara acak setiap observasi klustering
covid <- corona[c(1:4,9)] #mengambil atribut ke 1 s/d 4 dan atribut ke 9 (total ada 5 atribut)
colnames(covid)
```

```
## [1] "TotCases"  "TotDeath"  "Recovered" "ActCases"  "Pop"
```

Menentukan Jumlah Klaster

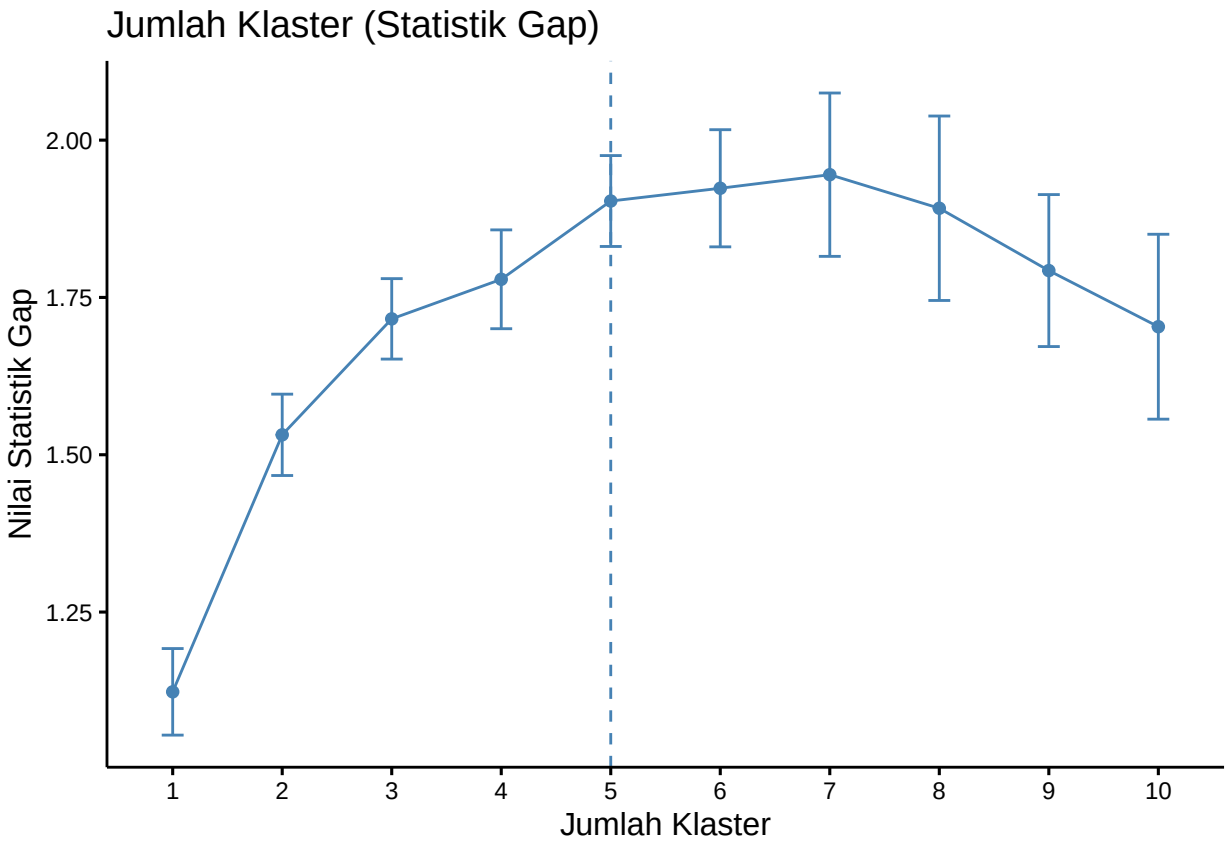
```
# Penentuan jumlah klaster dengan metode Siluet
klaster.Siluet <-  fviz_nbclust(covid, FUNcluster = kmeans, k.max = 10, method = "silhouette") +
theme(axis.text=element_text(size=9))
klaster.Siluet$labels$title = "Jumlah Klaster (Siluet)"
klaster.Siluet$labels$y = "Lebar Siluet"
klaster.Siluet$labels$x = "Jumlah Klaster"
```

```
klaster.Siluet
```

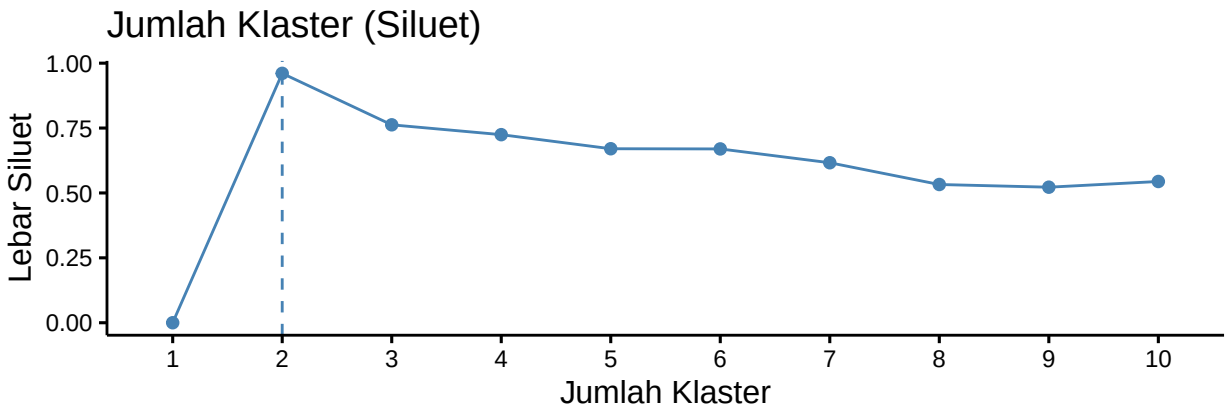


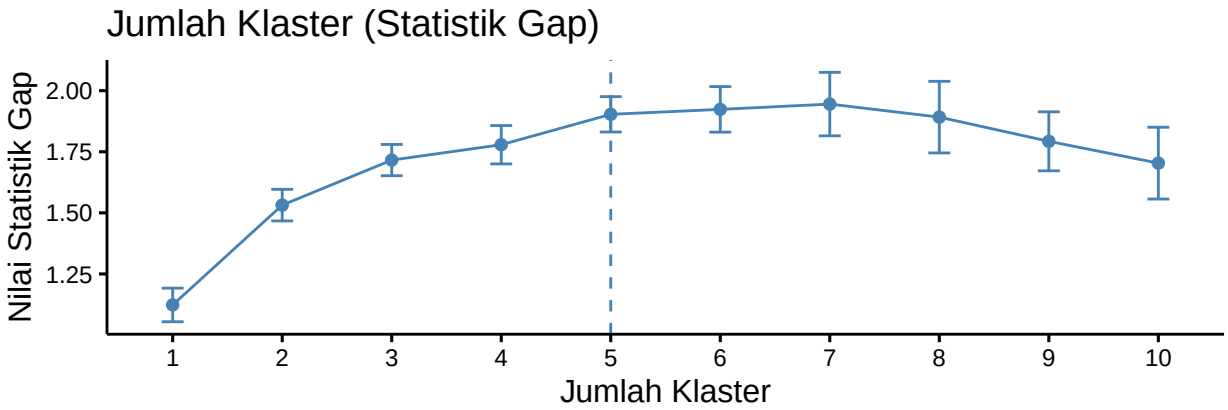

```
# Penentuan jumlah klaster dengan metode statistik gap
klaster.Gap = fviz_nbclust(covid, FUNcluster = kmeans, k.max = 10, method = "gap_stat") +
theme(axis.text=element_text(size=9))
klaster.Gap$labels$title = "Jumlah Klaster (Statistik Gap)"
klaster.Gap$labels$y = "Nilai Statistik Gap"
klaster.Gap$labels$x = "Jumlah Klaster"
```

```
klaster.Gap
```




```
# Visualisasi hasil penentuan klaster
gridExtra::grid.arrange(klaster.Siluet, klaster.Gap, nrow=2)
```





Memperhatikan jumlah klaster yang direkomendasikan dari kedua metode tersebut, berkisar antar 2 sampai 5 klaster. Dalam kasus ini kita akan melakukan klastering terhadap kasus COVID-19 tersebut diantara nilai rekomendasi klaster tersebut yaitu menjadi 3 dan 4 klaster saja.

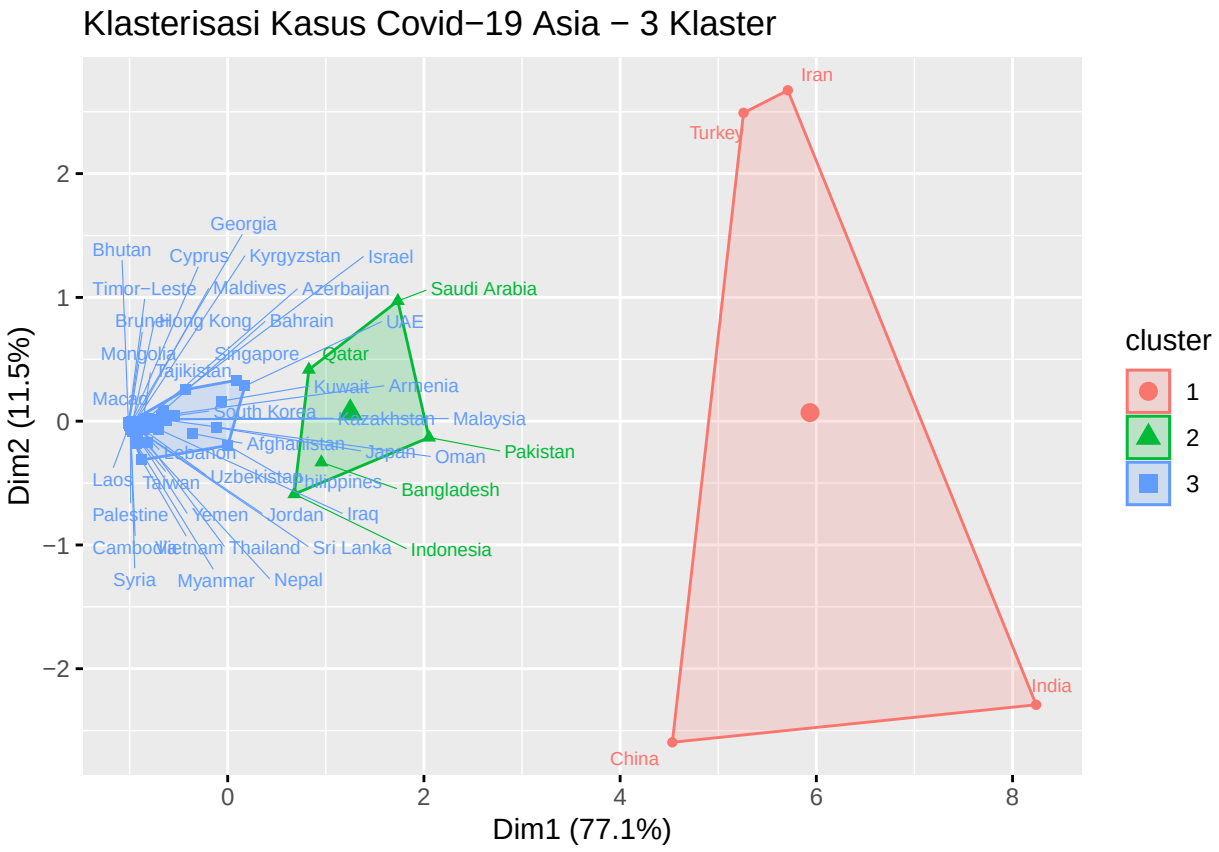
Latihan: *Sebagai perbandingan silahkan anda lakukan dengan menentukan jumlah klaster berdasarkan rekomendasi yang diberikan*

Membuat Model Klastering dengan K-Means Clustering

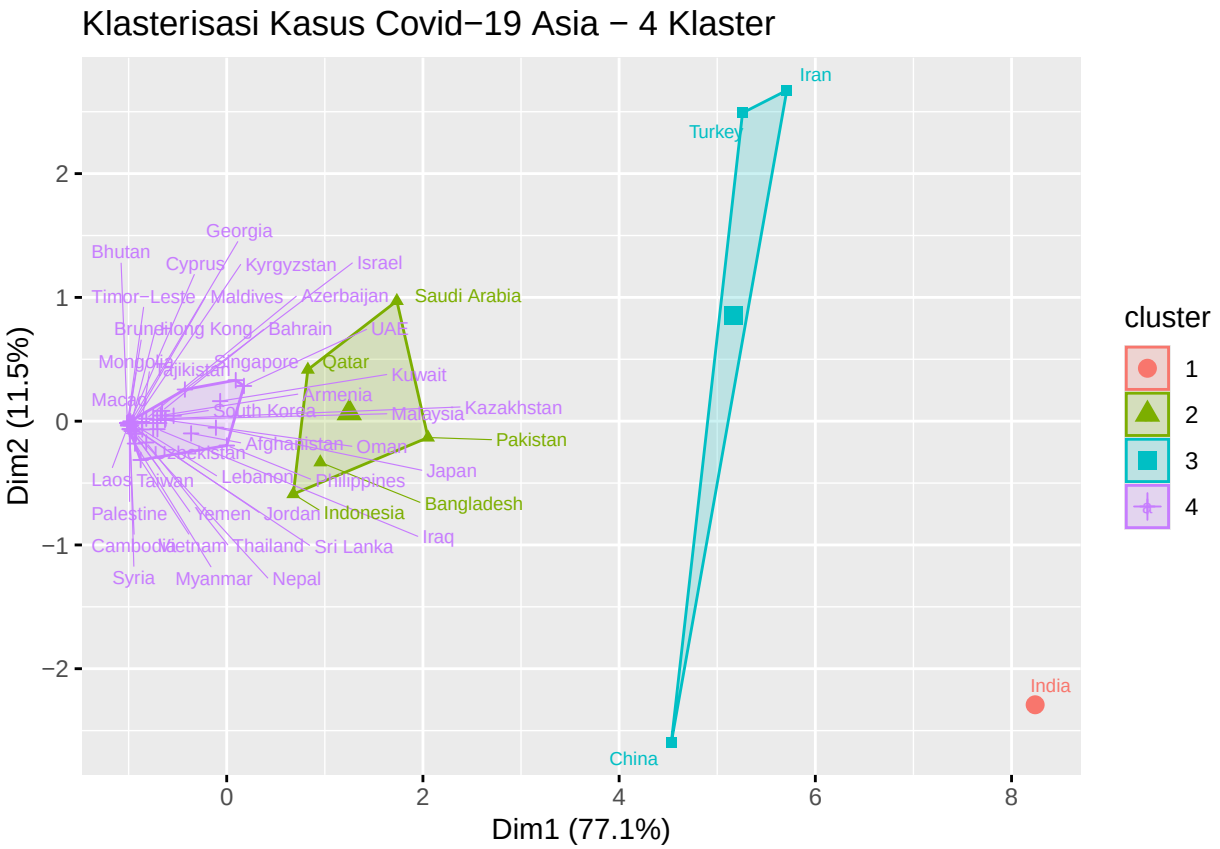
```
# Menentukan klasterisasi dengan K-Means Clustering
klasterCovid <-  get_dist(covid, stand = TRUE)
k3 <-  kmeans(klasterCovid, centers = 3, nstart = 25)
k4 <-  kmeans(klasterCovid, centers = 4, nstart = 25)
```

```
# Mendefinisikan plot diagram dari klasterisasi
Plot3 <-  fviz_cluster(k3, data = covid, repel = TRUE, labelsize = 7,
main = "Klasterisasi Kasus Covid-19 Asia - 3 Klaster")
Plot4 <-  fviz_cluster(k4, data = covid, repel = TRUE, labelsize = 7,
main = "Klasterisasi Kasus Covid-19 Asia - 4 Klaster")
```

```
# Visualisasi Plot klaster 3
Plot3
```




```
# Visualisasi Plot klaster 4
Plot4
```



Kalau dilihat dari kedua sumbu pada Grafik yaitu sumbu x dan y, didapatkan komponen-komponen utama yang terbentuk dari kelima atribut yang ada pada objek covid. Komponen utama yang pertama atau bisa dilihat dari Dim1 sebesar 77,1% dan komponen kedua sebesar 11,5%, sehingga dari kedua komponen tersebut dapat memformulasikan nilai matrik sebesar 88,6%. Untuk mengetahui besaran dari kelima komponen utama tersebut bisa dilakukan beberapa hal berikut ini.

```
# Membentuk 5 komponen utama dari objek Covid yang distandarisasi dan disimpan pada objek clus dengan m
clus <-  princomp(scale(covid))

# Menampilkan pusat dari masing-masing atribut
clus$center
```

```
##      TotCases      TotDeath     Recovered      ActCases           Pop
##  9.974660e-18 -4.064403e-17  7.155734e-18 -2.818926e-17 -2.577691e-17
```

```
# Menampilkan loading values (korelasi) antara komponen utama dengan atribut asal
clus$loadings
```

```
##
## Loadings:
##           Comp.1 Comp.2 Comp.3 Comp.4 Comp.5
## TotCases   0.499  0.224  0.125  0.259  0.786
## TotDeath   0.471  0.165 -0.408 -0.764
## Recovered  0.474  0.374 -0.252  0.528 -0.541
## ActCases   0.411 -0.103  0.831 -0.202 -0.297
```

```
## Pop        0.369 -0.879 -0.250  0.171
##
##                Comp.1 Comp.2 Comp.3 Comp.4 Comp.5
## SS loadings       1.0    1.0    1.0    1.0    1.0
## Proportion Var    0.2    0.2    0.2    0.2    0.2
## Cumulative Var    0.2    0.4    0.6    0.8    1.0
```

```
# Menampilkan nilai Eigen atau nilai karakteristik dari suatu matriks
eigen <- get_eig(clus)
eigen
```

```
##       eigenvalue variance.percent cumulative.variance.percent
## Dim.1 3.77453252        77.063372                    77.06337
## Dim.2 0.56083442        11.450369                    88.51374
## Dim.3 0.48102053         9.820836                    98.33458
## Dim.4 0.08157172         1.665423                   100.00000
## Dim.5 0.00000000         0.000000                   100.00000
```

Akurasi Model

Akurasi klaster bisa diketahui dengan menghitung rasio dari jumlah kuadrat antar klaster dengan jumlah kuadrat total. Sehingga bisa dihitung akurasi dari klaster 3 dan klaster 4 sebagai berikut:

```
Akurasi_klaster3 <-  (k3$betweenss/k3$totss)*100
Akurasi_klaster4 <-  (k4$betweenss/k4$totss)*100
```

```
# Lihat hasil akurasi
Akurasi_klaster3
```

```
## [1] 92.62079
```

```
Akurasi_klaster4
```

```
## [1] 95.77452
```

Kalau dilihat dari hasil diatas, ada peningkatan keakuratan (selisih antara akurasi 4 - akurasi 3). Sehingga bisa simpulkan bahwa ukuran peningkatan akurasinya relatif kecil. Berdasarkan hal tersebut klaster 3 nampaknya menjadi pilihan yang terbaik untuk digunakan.

Interpretasi hasil akurasi

Berdasarkan hasil akurasi pada data yang sudah dilakukan dengan melihat 5 atribut utama, maka didapatkan 3 klaster utama tentang kondisi COVID 19 di Asia, yang bisa diuraikan sebagai berikut:

1. Klaster 1, yaitu negara dengan tingkat kasus dan kematian yang sangat tinggi yaitu ada 4 negara; China, India, Iran, dan Turki
2. Klaster 2, yaitu negara dengan tingkat kasus dan kematin yang relatif tinggi, yaitu ada 5 negara; Indonesia, Banglades, Qatar, Saudi Arabia, dan Pakistan
3. Klaster 3, yaitu negara dengan tingkat kasus dan kematian yang kecil, yaitu ada 40 negara selain negara yang ada pada klaster 1 dan klaster 2

Referensi

1. K-Means Cluster Analysis,https://uc-r.github.io/kmeans_clustering