626 hw6

Q1. Using PCA to analyze and reduce features, and provide one graph and describe which countries stand out in a negative way in terms of these variables that relate to the sustainable development goals set for 2030.

```
library(dplyr)
```

```
##
## 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
```

```
dat <- read.csv("C:/Users/linzh/Desktop/ECON 626/hw6/pc6_sdg_goals.csv")
dat <- dat[!(dat$indexreg == ""),] # drop regional data and keep national data

# group data by indexreg (region), and fill in na with group mean (regional mean)
df <- dat %>%  group_by(indexreg) %>%
  mutate_all(funs(ifelse(is.na(.), mean(., na.rm = TRUE),.)))
```

```
## `mutate_all()` ignored the following grouping variables:
## Column `indexreg`
## Use `mutate_at(df, vars(-group_cols()), myoperation)` to silence the message.
```

```
#alternative way: use mutate()
# df <- dat %>%
#   group_by(indexreg) %>%
#   mutate(sdg1_wpc = ifelse(is.na(sdg1_wpc), mean(sdg1_wpc, na.rm = TRUE), sdg1_wpc))

# group_mean <- aggregate(csdg1_wpc ~ indexreg, data = dat, FUN = mean) # calculate regional mean
```

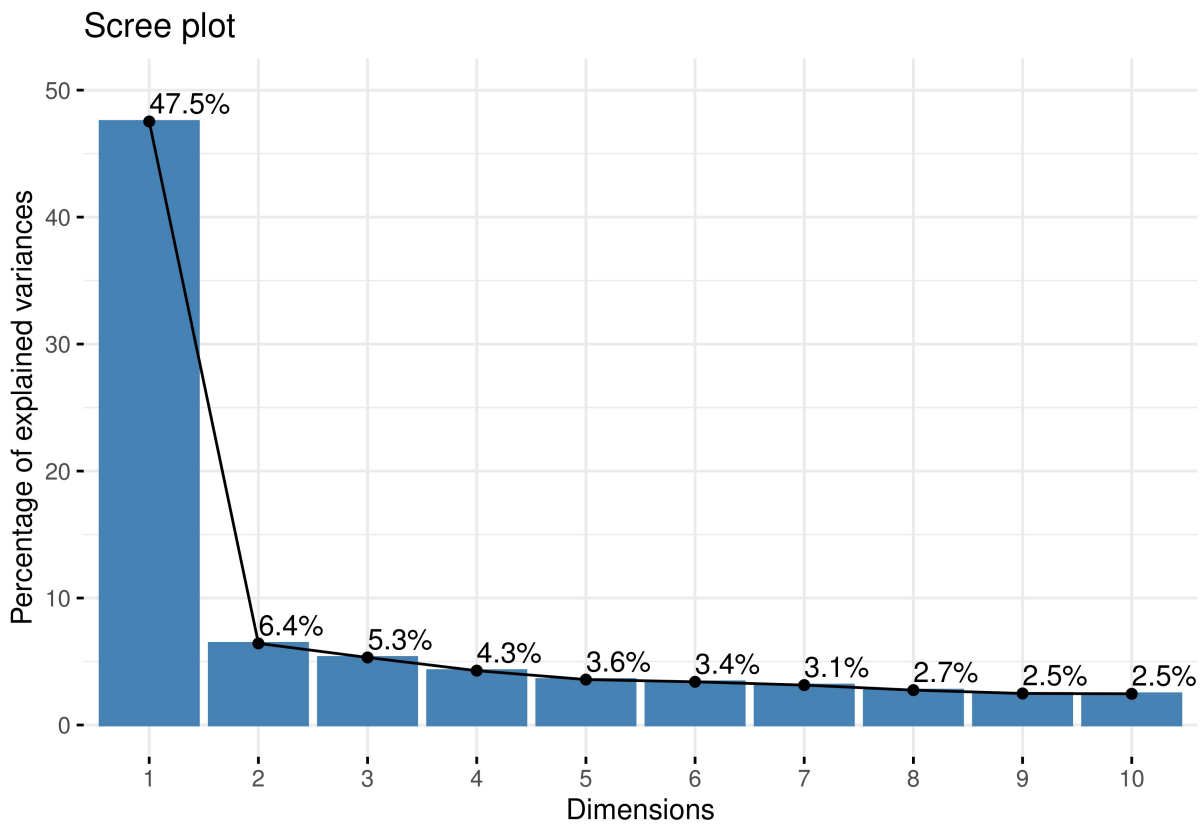
Step 2. Principal component analysis

```
pca <- prcomp(df[,5:36], scale=T)
comp <- data.frame(df[,1:3],pca$x[,1:2])
var <- pca$sdev^2/sum(pca$sdev^2)

#scree plot
library(factoextra)
```

```
## Loading required package: ggplot2
```

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

```
#scree plot
fviz_screeplot(pca, addlabels = TRUE, ylim = c(0, 50))
```




```
# Alternative way:
# library(factoextra)
# fviz_eig(pca, addlabels = TRUE)
# fviz_pca_ind(pca, col.ind = "cos2", # Color by the quality of representation
#   gradient.cols = c("#00AFBB", "#E7B800", "#FC4E07"), repel = TRUE ) # Avoid text overlapping
```

Step 3. Graph

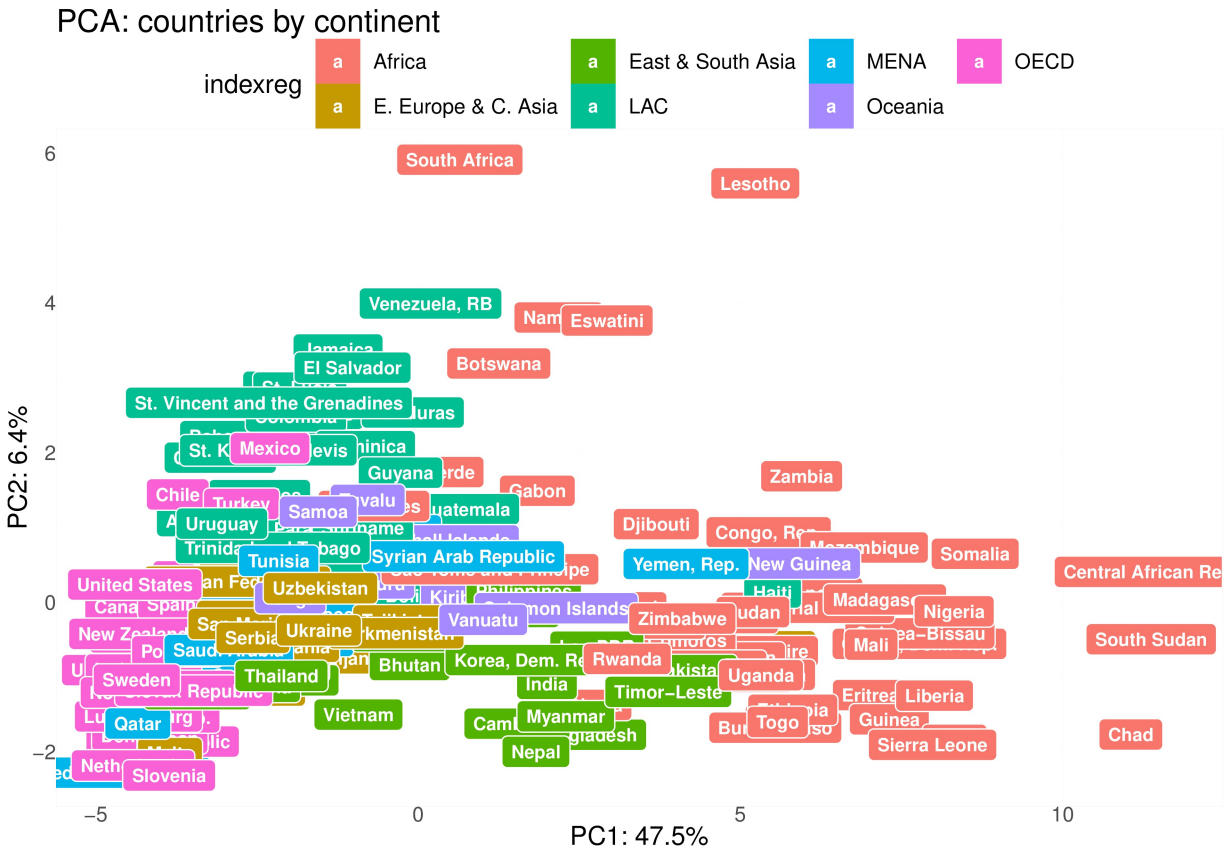
```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --
```

```
## v tibble  3.0.4     v purrr   0.3.4
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.1
```

```
## -- Conflicts ---------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
comp %>%
  as.data.frame %>%
  ggplot(aes(x=PC1,y=PC2, label=country, color=indexreg)) +
  geom_label(aes(fill = indexreg), colour = "white", fontface = "bold", cex = 2.5)+
  theme_bw(base_size=0.1) +
  labs(x=paste0("PC1: ",round(var[1]*100,1),"%"),
       y=paste0("PC2: ",round(var[2]*100,1),"%"))+
  labs(title="PCA: countries by continent")+
  theme(legend.position="top", text = element_text(size=10))
```



From the graph, South Africa, Lesotho, Central African Region, South Sudan, and Chad stand out in a negative way in terms of the sustainable development goals set for 2030. In specific, Central African Region, South Sudan, and Chad stand out in terms of PC1, and South Africa and Lesotho stand out in terms of PC2.

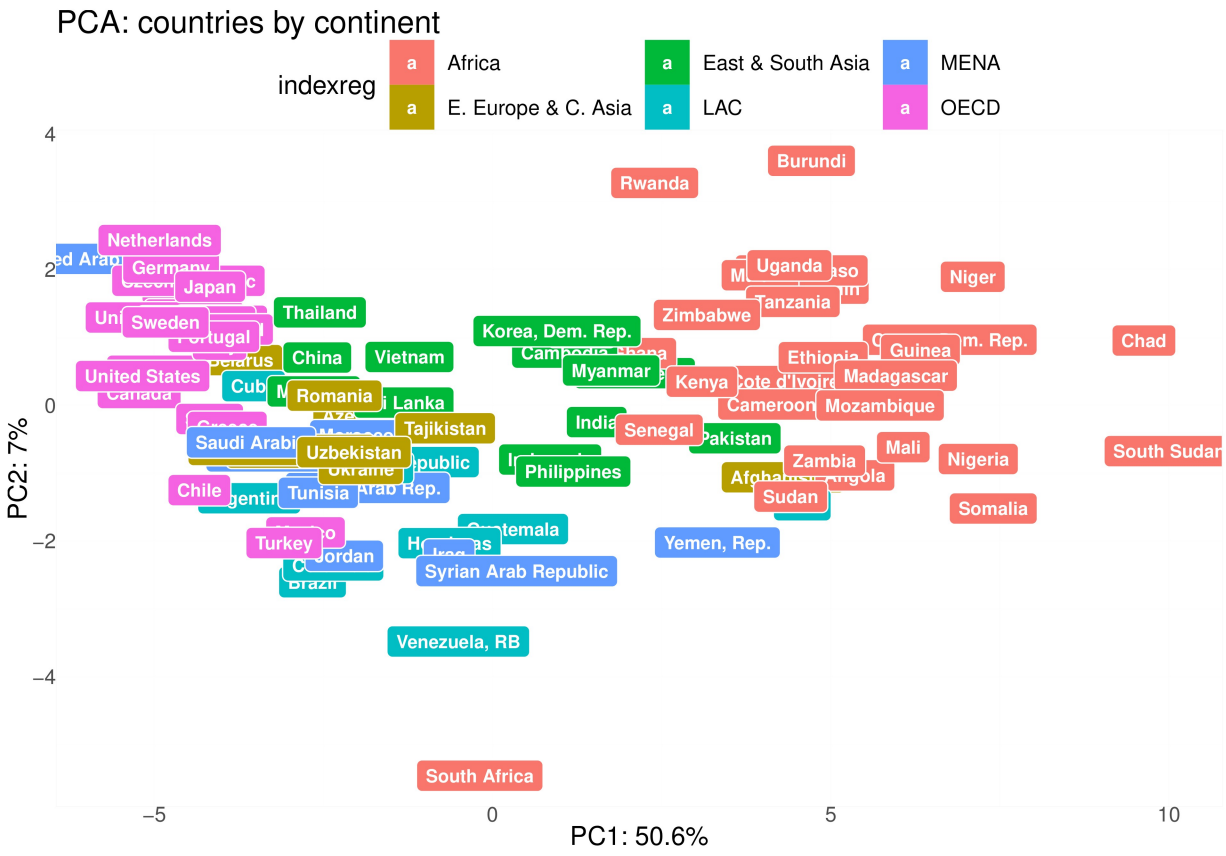
Q2. Repeat your analysis in Q2 separately for (1) countries with a population of more than 9 million people and (2) countries with a population of less than 9 million people.

```
# countries with a population of more than 9 million people: g1(group1)
g1 <- filter(dat, pop_2020 > 9000000)
g1 <- g1[!(g1$indexreg == ""),] # drop regional data and keep national data

# fill in na with group mean after data is grouped by indexreg
```

```
group1 <- g1 %>%  group_by(indexreg) %>%
  mutate_all(funs(ifelse(is.na(.), mean(., na.rm = TRUE),.)))
```

```
## `mutate_all()` ignored the following grouping variables:
## Column `indexreg`
## Use `mutate_at(df, vars(-group_cols()), myoperation)` to silence the message.
```

```
#pca
pca.group1 <- prcomp(group1[,5:36], scale=T)
comp.group1 <- data.frame(group1[,1:3],pca.group1$x[,1:2])
var.group1 <- pca.group1$sdev^2/sum(pca.group1$sdev^2)

#graph
comp.group1 %>%
  as.data.frame %>%
  ggplot(aes(x=PC1,y=PC2, label=country, color=indexreg)) +
  geom_label(aes(fill = indexreg), colour = "white", fontface = "bold", cex = 2.5)+
  theme_bw(base_size=0.1) +
  labs(x=paste0("PC1: ",round(var.group1[1]*100,1),"%"),
       y=paste0("PC2: ",round(var.group1[2]*100,1),"%"))+
  labs(title="PCA: countries by continent")+
  theme(legend.position="top", text = element_text(size=10))
```



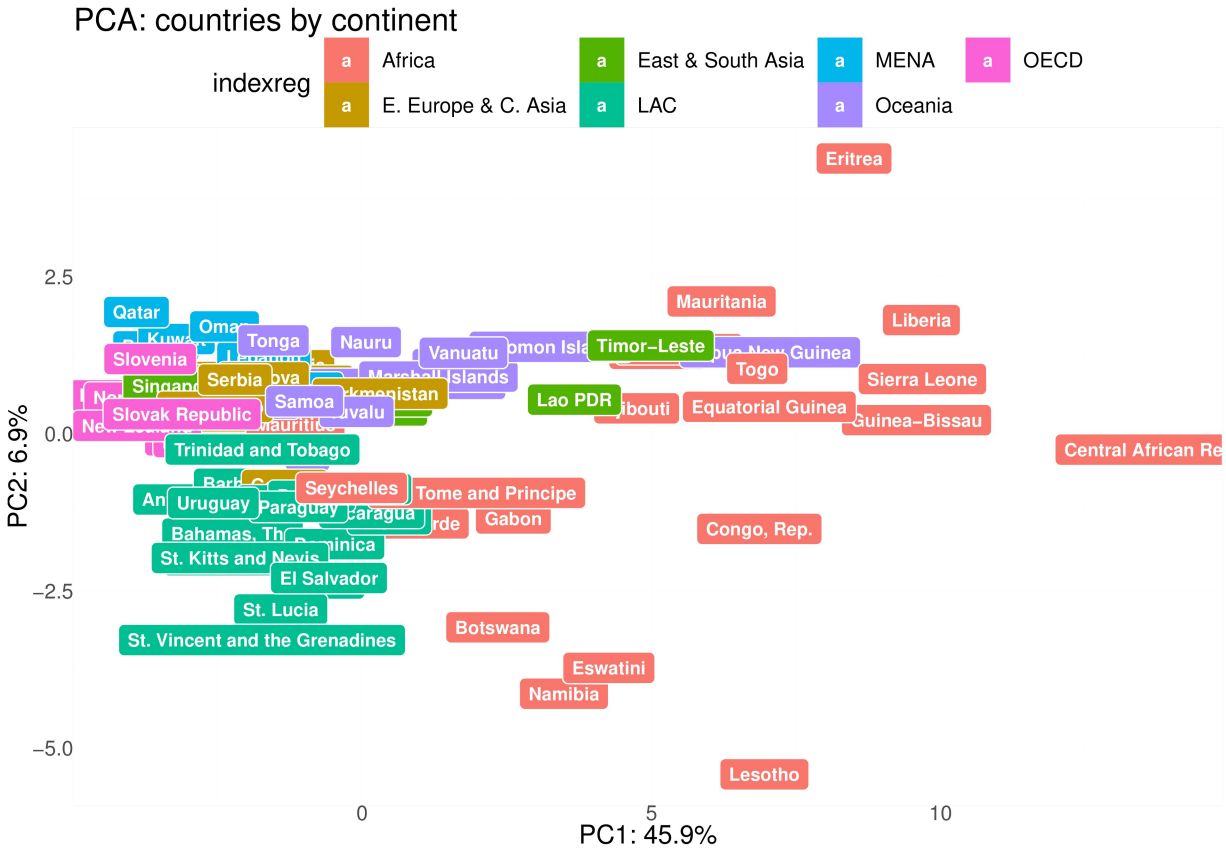
Rwanda, Burundi, South Africa, South Sudan, and Chad stand out in a negative way in terms of the sustainable development goals set for 2030 among countries with a population above 9 millions. In specific,

Chad and South Africa stand out in terms of PC1; Rwanda, Burundi, and South Africa stand out in terms of PC2.

```
# countries with a population of less than 9 million people: g2(group2)
g2 <- filter(dat, pop_2020 < 9000000)
g2 <- g2[!(g2$indexreg == ""),] # drop regional data and keep national data

# fill in na with group mean after data is grouped by indexreg
group2 <- g2 %>%  group_by(indexreg) %>%
  mutate_all(funs(ifelse(is.na(.), mean(., na.rm = TRUE),.)))
```

```
## `mutate_all()` ignored the following grouping variables:
## Column `indexreg`
## Use `mutate_at(df, vars(-group_cols()), myoperation)` to silence the message.
```

```
#pca
pca.group2 <- prcomp(group2[,5:36], scale=T)
comp.group2 <- data.frame(group2[,1:3],pca.group2$x[,1:2])
var.group2 <- pca.group2$sdev^2/sum(pca.group2$sdev^2)

#graph
comp.group2 %>%
  as.data.frame %>%
  ggplot(aes(x=PC1,y=PC2, label=country, color=indexreg)) +
  geom_label(aes(fill = indexreg), colour = "white", fontface = "bold", cex = 2.5)+
  theme_bw(base_size=0.1) +
  labs(x=paste0("PC1: ",round(var.group2[1]*100,1),"%"),
       y=paste0("PC2: ",round(var.group2[2]*100,1),"%"))+
  labs(title="PCA: countries by continent")+
  theme(legend.position="top", text = element_text(size=10))
```

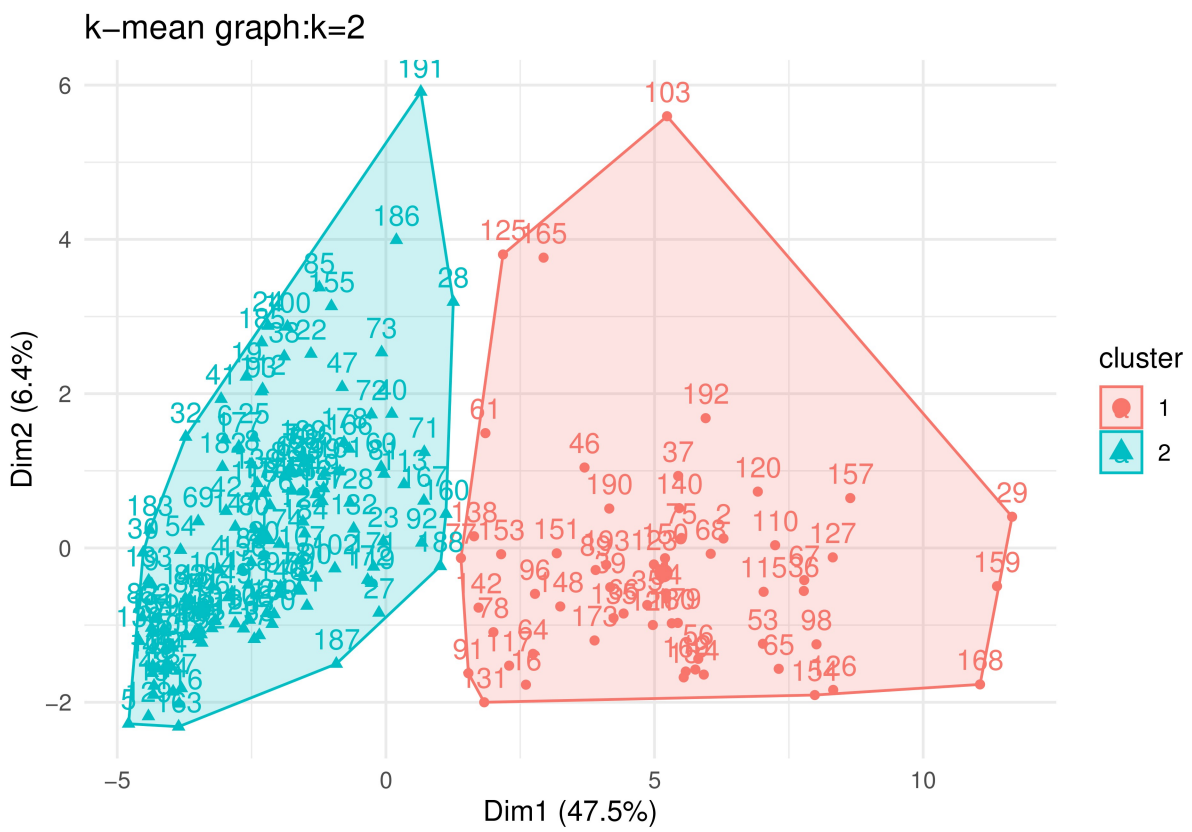


Eritrea, Central African Region, and Lesotho stand out in a negative way in terms of the sustainable development goals set for 2030 among countries with a population below 9 millions. In specific, Eritrea and Central African Region stand out in terms of PC1; Eritrea and Lesotho stand out in terms of PC2.

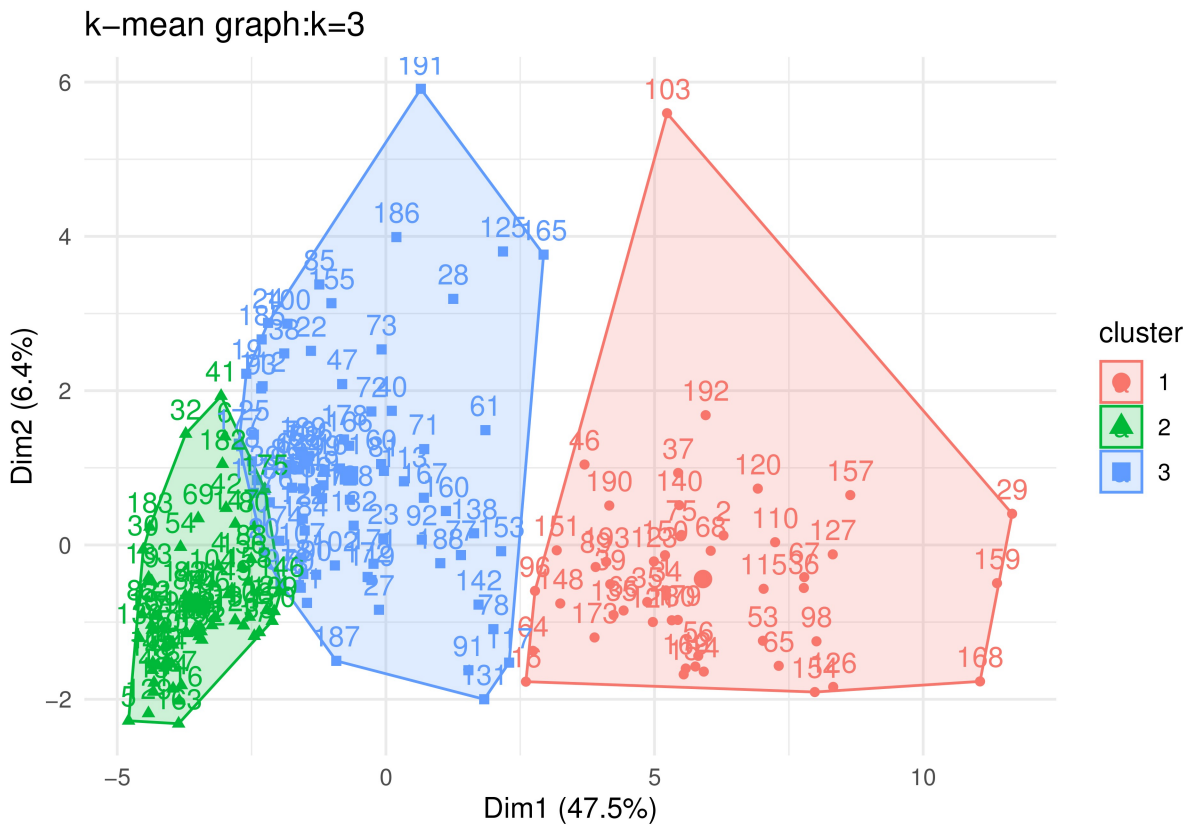
Analyzing 2 subgroups separately according to whether the national population is greater than 9 million changes the result, because the variation among countries is closely related to, or largely generated by, population structure. In specific, Li and Ralph (2019) demonstrated their findings that the systematic patterns among the variations in subjects were generated by population structure (p.289).

Q3.Use the sustainable development goals data and a clustering algorithm to examine whether cross-country variation in development is best captured by a model with 2, 3 or 4 clusters. Which number of clusters seems to best describe the pattern that is present in the data?

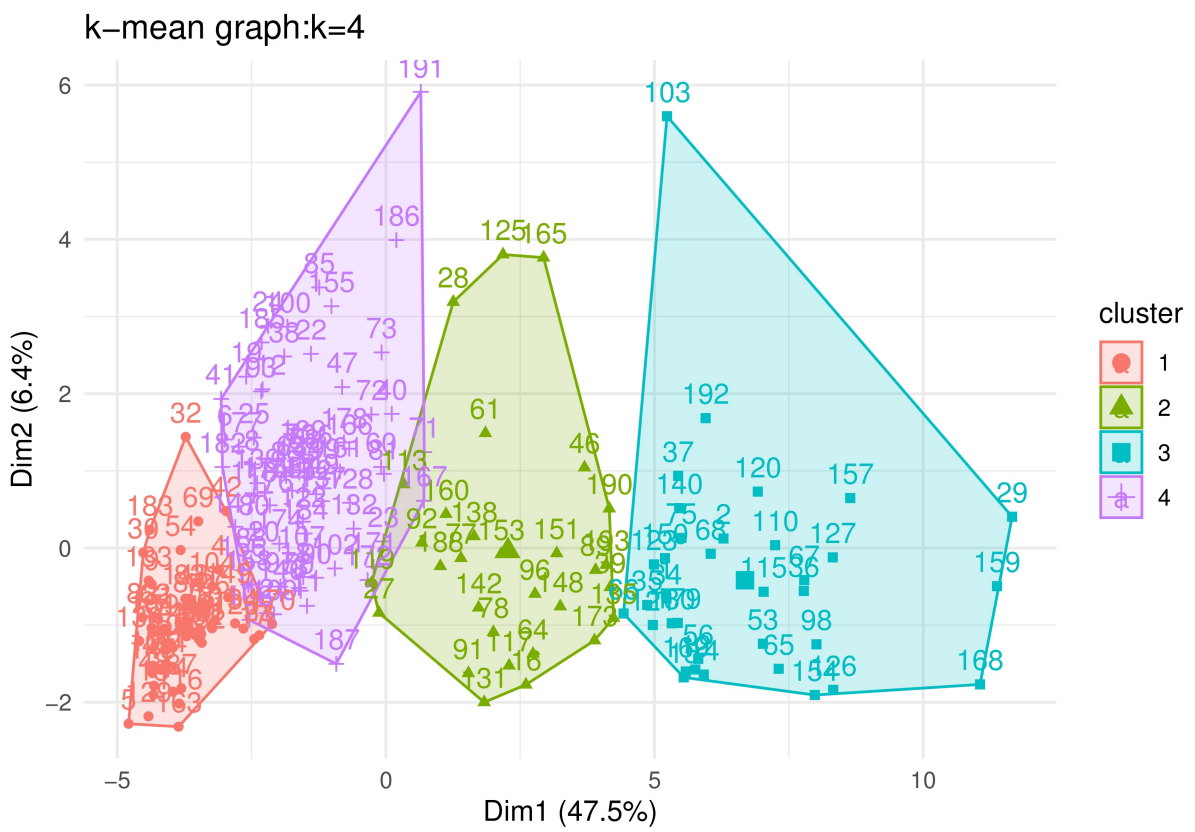
```
dat3 <- scale(df[5:36]) # Scaling the data

# Compute k-means
set.seed(123)
clusters2 <- kmeans(dat3, 2, nstart = 25)
clusters3 <- kmeans(dat3, 3, nstart = 25)
clusters4 <- kmeans(dat3, 4, nstart = 25)

# k-mean plots
fviz_cluster(clusters2, data = dat3, frame.type = "convex") + theme_minimal() + ggtitle("k-mean graph:k
```




```
fviz_cluster(clusters3, data = dat3, frame.type = "convex") + theme_minimal() + ggtitle("k-mean graph:k=
```




```
fviz_cluster(clusters4, data = dat3, frame.type = "convex") + theme_minimal() + ggtitle("k-mean graph:k=
```

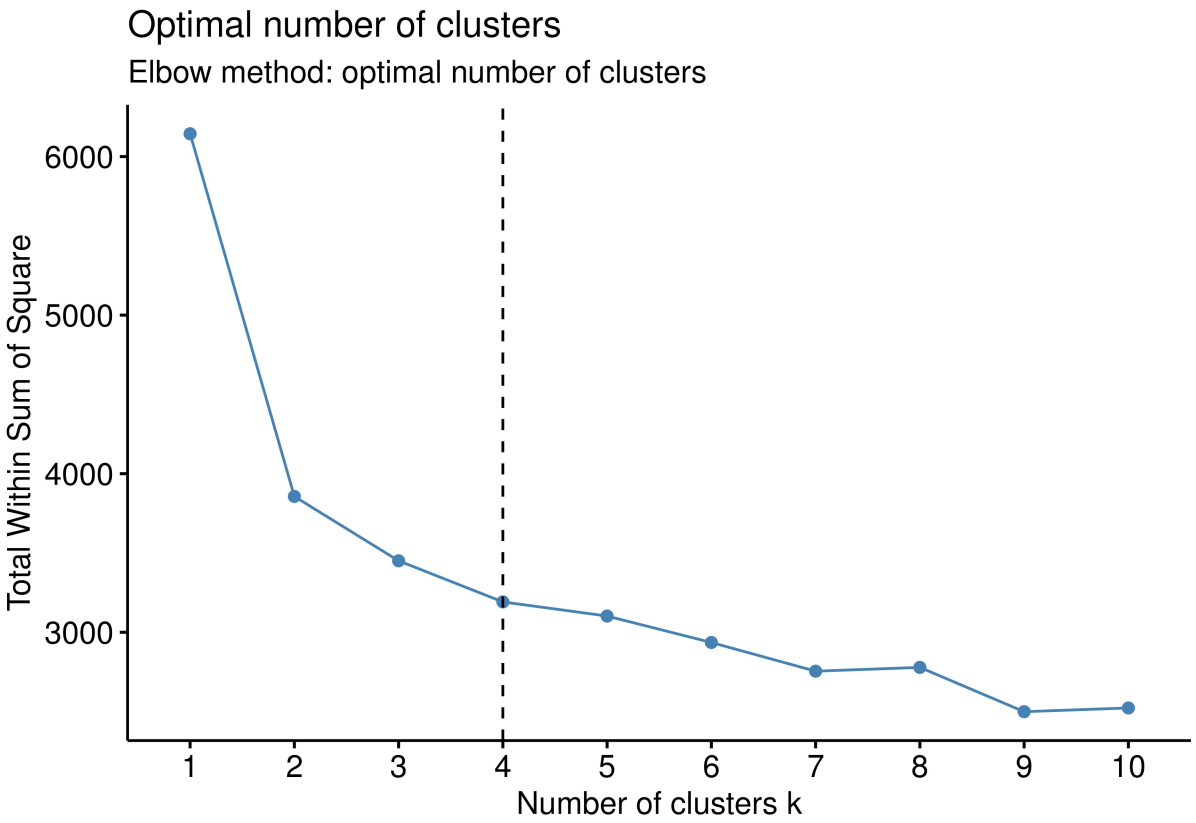


a. the total within-cluster variation

```
within_ss <- c(clusters2$tot.withinss, clusters3$tot.withinss, clusters4$tot.withinss)
within_ss # 3856.691, 3448.04, 3191.588
```

```
## [1] 3856.691 3448.040 3191.588
```

```
# graph of the total within-cluster variation: Elbow method
library(factoextra)
library(NbClust)

fviz_nbclust(dat3, kmeans, method = "wss") + geom_vline(xintercept = 4, linetype = 2) +
  labs(subtitle = "Elbow method: optimal number of clusters") # optimal k = 4
```



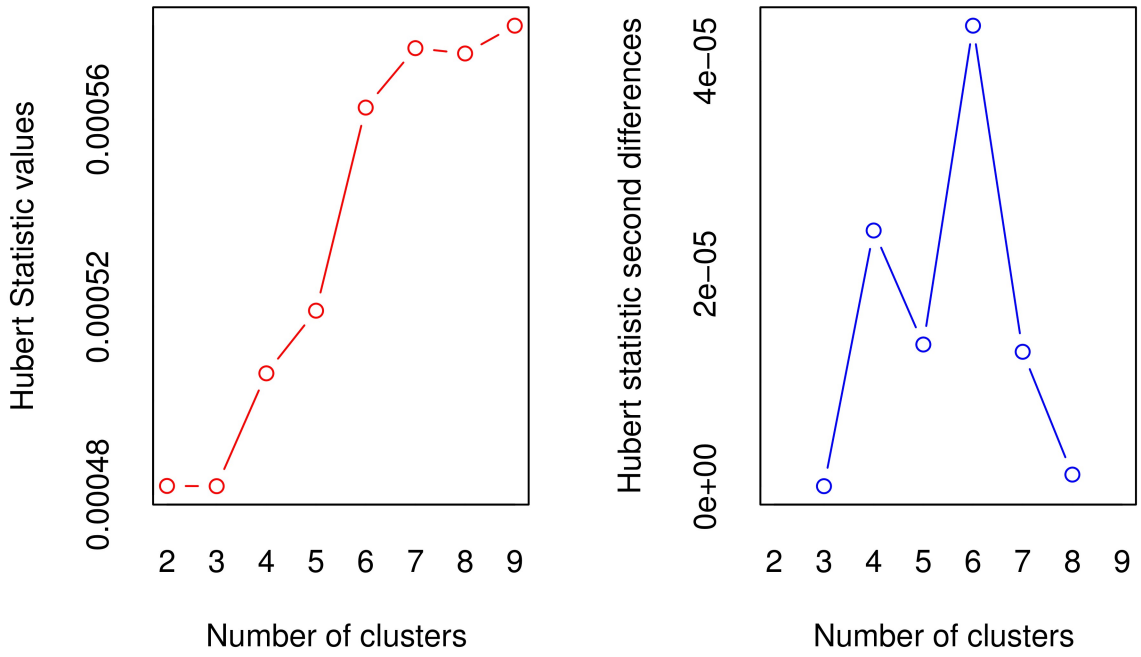
The results show the "elbow" locates at k = 4 indicates 4 is the appropriate number of clusters based on the total within-cluster variation.

b. the between-cluster variation

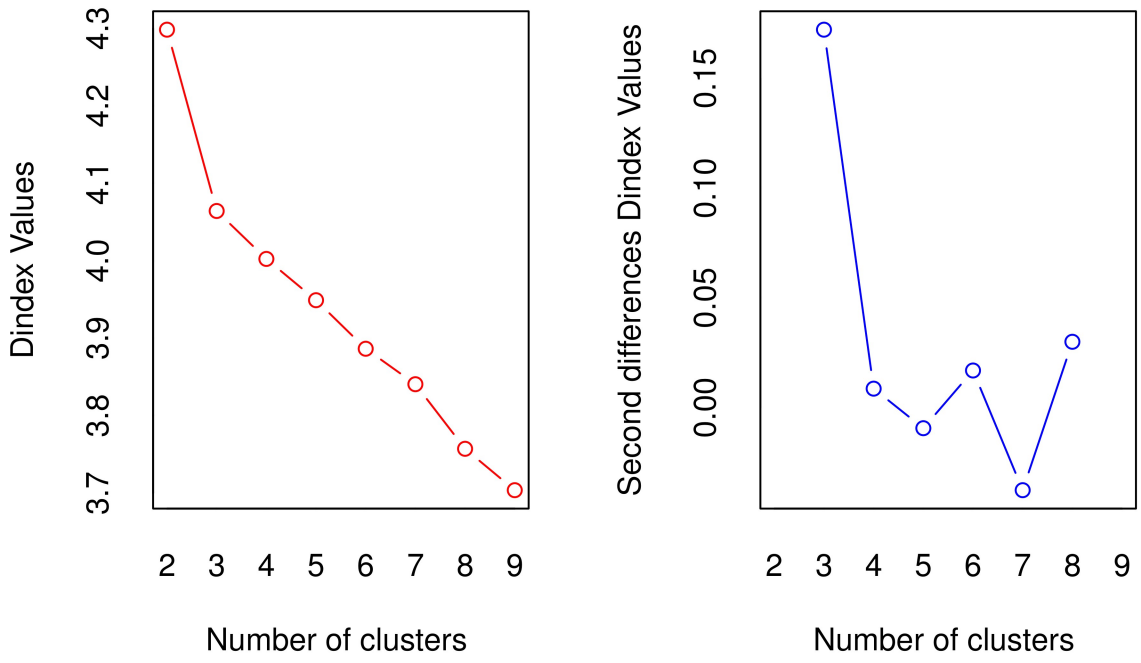
```
between_ss = c(clusters2$betweenss, clusters3$betweenss, clusters4$betweenss)
between_ss # 2287.309 2695.960 2952.412
```

```
## [1] 2287.309 2695.960 2952.412
```

```
# graph of the total between-cluster variation
nbclust <- NbClust(dat3, distance = "euclidean", min.nc = 2, max.nc = 9,
                   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.
##
```

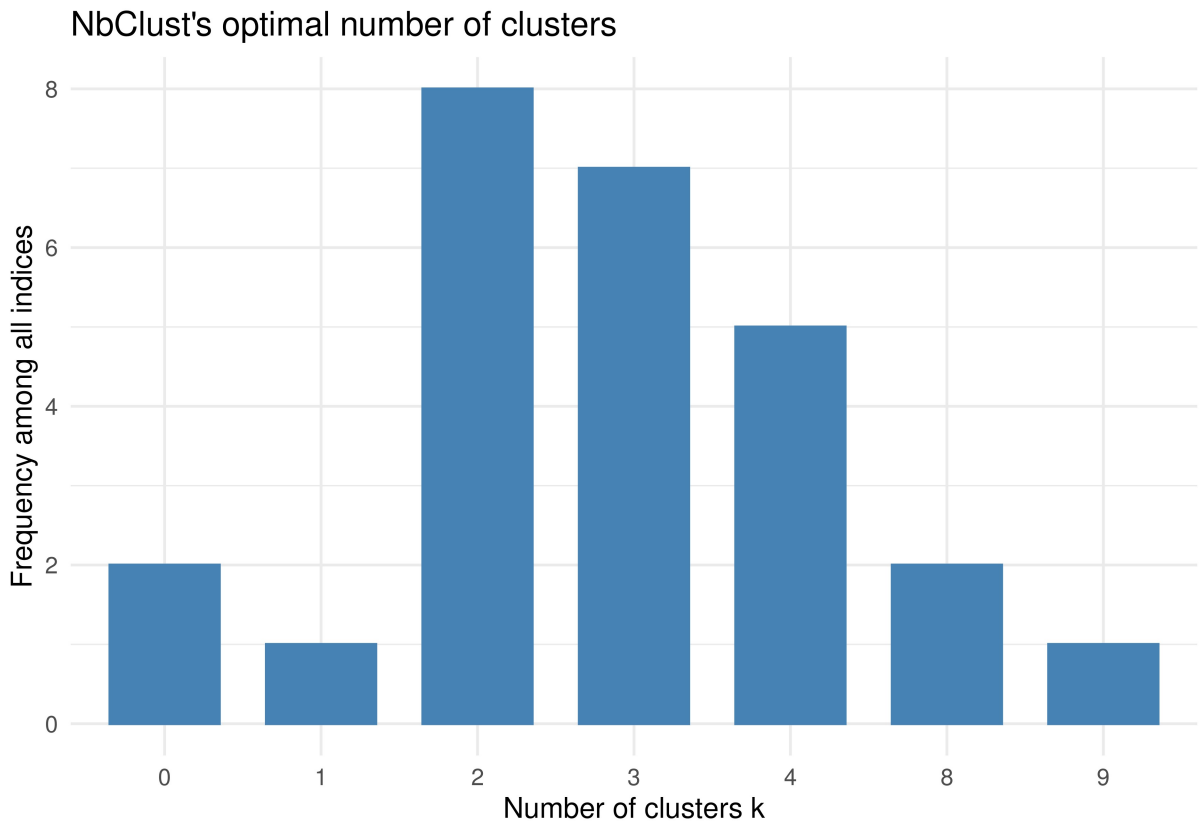



```
## *** : The D index is a graphical method of determining the number of clusters.
##                 In the plot of D index, we seek a significant knee (the significant peak in Dindex
##                 second differences plot) that corresponds to a significant increase of the value of
##                 the measure.
##
## *******************************************************************
## * Among all indices:
## * 8 proposed 2 as the best number of clusters
## * 7 proposed 3 as the best number of clusters
## * 5 proposed 4 as the best number of clusters
## * 2 proposed 8 as the best number of clusters
## * 1 proposed 9 as the best number of clusters
##
##                   ***** Conclusion *****
##
## * According to the majority rule, the best number of clusters is  2
##
##
## *******************************************************************
```

```
factoextra::fviz_nbclust(nbclust) + theme_minimal() +
  ggtitle("NbClust's optimal number of clusters") # optimal k = 2
```

```
## Among all indices:
## ===================
```

```
## * 2 proposed  0 as the best number of clusters
## * 1 proposed  1 as the best number of clusters
## * 8 proposed  2 as the best number of clusters
## * 7 proposed  3 as the best number of clusters
## * 5 proposed  4 as the best number of clusters
## * 2 proposed  8 as the best number of clusters
## * 1 proposed  9 as the best number of clusters
##
## Conclusion
## =========================
## * According to the majority rule, the best number of clusters is  2 .
```



2 clusters best describe the data according to the majority rules based on the between-cluster variation.

References: Li, H., & Ralph, P. (2019). Local PCA Shows How the Effect of Population Structure Differs Along the Genome. Genetics (Austin), 211(1), 289–304. https://doi.org/10.1534/genetics.118.301747