Cluster Analysis Assignment

Nyasha Mashanda

2020-10-17

Contents

Contents

List of Figures

List of Tables

Table 1: Acronyms

Variable	Description
iso_code	ISO 3166-1 alpha-3 – three-letter country codes
continent	Continent of the geographical location
location	Geographical location
date	Date of observation
total_cases	Total confirmed cases of COVID-19
new_cases	New confirmed cases of COVID-19
new_cases_smoothed	New confirmed cases of COVID-19 (7-day smoothed)
total_deaths	Total deaths attributed to COVID-19
new_deaths	New deaths attributed to COVID-19
new_deaths_smoothed	New deaths attributed to COVID-19 (7-day smoothed)
total_cases_per_million	Total confirmed cases of COVID-19 per 1,000,000 people
new_cases_per_million	New confirmed cases of COVID-19 per 1,000,000 people
new_cases_smoothed_per_million	New confirmed cases of COVID-19 (7-day smoothed) per 1,000,000 people
total_deaths_per_million	Total deaths attributed to COVID-19 per 1,000,000 people
new_deaths_per_million	New deaths attributed to COVID-19 per 1,000,000 people
new_deaths_smoothed_per_million	New deaths attributed to COVID-19 (7-day smoothed) per 1,000,000 people
population_density	Number of people divided by land area, measured in square kilometers
median_age	Median age of the population, UN projection for 2020
aged_65_older	Share of the population that is 65 years and older, most recent year available
aged_70_older	Share of the population that is 70 years and older in 2015
gdp_per_capita	Gross domestic product at purchasing power parity (constant 2011 international dollars)
extreme_poverty	Share of the population living in extreme poverty, most recent year available since 2010
cardiovasc_death_rate	Death rate from cardiovascular disease in 2017 (annual number of deaths per 100,000 people)
diabetes_prevalence	Diabetes prevalence (% of population aged 20 to 79) in 2017
female_smokers	Share of women who smoke, most recent year available
male_smokers	Share of men who smoke, most recent year available
handwashing_facilities	Share of the population with basic handwashing facilities on premises
hospital_beds_per_thousand	Hospital beds per 1,000 people, most recent year available since 2010
life_expectancy	Life expectancy at birth in 2019

```
library(kableExtra)
```

```
##
## Attaching package: 'kableExtra'
```

```
## The following object is masked from 'package:dplyr':
##
##     group_rows
```

```
# Display this on the first page to see what and from where each variable was collected.
owid_covid_codebook <- read_csv("owid-covid-codebook.csv")
```

```
## Parsed with column specification:
## cols(
##   Variable = col_character(),
##   Description = col_character()
## )
```

```
knitr::kable(owid_covid_codebook, digits = 5, caption = "Acronyms") %>%
  kable_styling(full_width = F, font_size = 7) %>%
  column_spec(1, border_left = T) %>%
  column_spec(2, border_right = T)
```

```
#View(owid_covid_codebook)
```

Abstract

The novel COVID-19 corona virus is still not well understood and there are many open questions related to patterns in its spread. The goal of this assignment is to discover if there are any regional patterns that exist using cluster analysis.

The assignment uses COVD-19 pandemic data collected from the Our World In Data site (European Center for Disease Prevention and Control 2020). The data contains 29 indicators related to the COVID-19 cases for 208 countries. The data set is updated daily from when the pandemic started. For this assignment, a subset of the data will be used; this subset consists of all the information on the pandemic on the 02 of September 2020.

There are three kinds of clustering methods that will be explored in this analysis: hierarchical, partitioning and density based methods. In order to determine any regional patterns, the number of clusters will be limited to 6, resembling the six regions that are: Africa, Asia, Europe, North America, Oceania and South America.

1 What about standardising the data????

2 Exploratory Data Analysis

```
# Reading the data
df1 <- read_excel("owid-covid-data.xlsx", sheet = "Data")

# Check if data has been imported correctly
head(df1)
```

```
## # A tibble: 6 x 30
##   location iso_code continent date  total_cases new_cases new_cases_smoot~
##   <chr>    <chr>    <chr>     <chr>       <dbl>     <dbl>            <dbl>
## 1 Algeria  DZA      Africa    2020~       44833       339            372.
## 2 Angola   AGO      Africa    2020~        2654        30             53
## 3 Benin    BEN      Africa    2020~        2145         0              4.29
## 4 Botswana BWA      Africa    2020~        1733         9             24.4
## 5 Burkina~ BFA      Africa    2020~        1375         5              3.29
## 6 Burundi  BDI      Africa    2020~         445         0              2.14
## # ... with 23 more variables: total_deaths <dbl>, new_deaths <dbl>,
## #   new_deaths_smoothed <dbl>, total_cases_per_million <dbl>,
## #   new_cases_per_million <dbl>, new_cases_smoothed_per_million <dbl>,
## #   total_deaths_per_million <dbl>, new_deaths_per_million <dbl>,
## #   new_deaths_smoothed_per_million <dbl>, population <dbl>,
## #   population_density <dbl>, median_age <dbl>, aged_65_older <dbl>,
## #   aged_70_older <dbl>, gdp_per_capita <dbl>, extreme_poverty <dbl>,
## #   cardiovasc_death_rate <dbl>, diabetes_prevalence <dbl>,
## #   female_smokers <dbl>, male_smokers <dbl>, handwashing_facilities <dbl>,
## #   hospital_beds_per_thousand <dbl>, life_expectancy <dbl>
```

```
tail(df1)
```

```
## # A tibble: 6 x 30
##   location iso_code continent date  total_cases new_cases new_cases_smoot~
##   <chr>    <chr>    <chr>     <chr>       <dbl>     <dbl>            <dbl>
## 1 Guyana   GUY      South Am~ 2020~        1373        67             44.7
## 2 Paraguay PRY      South Am~ 2020~       18338       676            587.
## 3 Peru     PER      South Am~ 2020~      657129      5092           7107.
## 4 Suriname SUR      South Am~ 2020~        4089        55             55.9
## 5 Uruguay  URY      South Am~ 2020~        1611        16             10.7
## 6 Venezue~ VEN      South Am~ 2020~       47756      1888            943.
## # ... with 23 more variables: total_deaths <dbl>, new_deaths <dbl>,
## #   new_deaths_smoothed <dbl>, total_cases_per_million <dbl>,
## #   new_cases_per_million <dbl>, new_cases_smoothed_per_million <dbl>,
## #   total_deaths_per_million <dbl>, new_deaths_per_million <dbl>,
## #   new_deaths_smoothed_per_million <dbl>, population <dbl>,
## #   population_density <dbl>, median_age <dbl>, aged_65_older <dbl>,
## #   aged_70_older <dbl>, gdp_per_capita <dbl>, extreme_poverty <dbl>,
## #   cardiovasc_death_rate <dbl>, diabetes_prevalence <dbl>,
## #   female_smokers <dbl>, male_smokers <dbl>, handwashing_facilities <dbl>,
## #   hospital_beds_per_thousand <dbl>, life_expectancy <dbl>
```

```
# Look at the structure of the data
str(df1)
```

```
## tibble [208 x 30] (S3: tbl_df/tbl/data.frame)
##  $ location                      : chr [1:208] "Algeria" "Angola" "Benin" "Botswana" ...
##  $ iso_code                      : chr [1:208] "DZA" "AGO" "BEN" "BWA" ...
##  $ continent                     : chr [1:208] "Africa" "Africa" "Africa" "Africa" ...
```

```
##  $ date                           : chr [1:208] "2020-09-02" "2020-09-02" "2020-09-02" "2020-09-02" ...
##  $ total_cases                    : num [1:208] 44833 2654 2145 1733 1375 ...
##  $ new_cases                      : num [1:208] 339 30 0 9 5 0 267 86 0 4 ...
##  $ new_cases_smoothed             : num [1:208] 372.14 53 4.29 24.43 3.29 ...
##  $ total_deaths                   : num [1:208] 1518 108 40 6 55 ...
##  $ new_deaths                     : num [1:208] 8 1 0 0 0 0 3 0 0 0 ...
##  $ new_deaths_smoothed            : num [1:208] 8.857 0.857 0.143 0.429 0 ...
##  $ total_cases_per_million        : num [1:208] 1022.4 80.8 176.9 736.9 65.8 ...
##  $ new_cases_per_million          : num [1:208] 7.731 0.913 0 3.827 0.239 ...
##  $ new_cases_smoothed_per_million : num [1:208] 8.487 1.613 0.354 10.388 0.157 ...
##  $ total_deaths_per_million       : num [1:208] 34.62 3.29 3.3 2.55 2.63 ...
##  $ new_deaths_per_million         : num [1:208] 0.182 0.03 0 0 0 0 0.113 0 0 0 ...
##  $ new_deaths_smoothed_per_million: num [1:208] 0.202 0.026 0.012 0.182 0 0 0.022 0.771 0.03 0 ...
##  $ population                     : num [1:208] 43851043 32866268 12123198 2351625 20903278 ...
##  $ population_density             : num [1:208] 17.35 23.89 99.11 4.04 70.15 ...
##  $ median_age                     : num [1:208] 29.1 16.8 18.8 25.8 17.6 17.5 18.8 25.7 18.3 16.7 ...
##  $ aged_65_older                  : num [1:208] 6.21 2.4 3.24 3.94 2.41 ...
##  $ aged_70_older                  : num [1:208] 3.86 1.36 1.94 2.24 1.36 ...
##  $ gdp_per_capita                 : num [1:208] 13914 5819 2064 15807 1703 ...
##  $ extreme_poverty                : num [1:208] 0.5 NA 49.6 NA 43.7 71.7 23.8 NA NA 38.4 ...
##  $ cardiovasc_death_rate          : num [1:208] 278 276 236 237 269 ...
##  $ diabetes_prevalence            : num [1:208] 6.73 3.94 0.99 4.81 2.42 6.05 7.2 2.42 6.1 6.1 ...
##  $ female_smokers                 : num [1:208] 0.7 NA 0.6 5.7 1.6 NA NA 2.1 NA NA ...
##  $ male_smokers                   : num [1:208] 30.4 NA 12.3 34.4 23.9 NA NA 16.5 NA NA ...
##  $ handwashing_facilities         : num [1:208] 83.7 26.7 11 NA 11.9 ...
##  $ hospital_beds_per_thousand     : num [1:208] 1.9 NA 0.5 1.8 0.4 0.8 1.3 2.1 1 NA ...
##  $ life_expectancy                : num [1:208] 76.9 61.1 61.8 69.6 61.6 ...
```

The data was exported to a dataframe from an excel file "owid-covid-data.xlsx". The first step is to check if the data has been imported correctly using the head() and tail() functions. This was confirmed by running the code. The next step included checking the structure of the dataframe and also the variables in the data (See variables in Table 1). Location, iso_code, continent and date were found to be in character format while the rest of the variables are numerical. This is fine expcept for the date which should be in a date format. However, since this column has only one date, the column will not be used for ana;ysis and will be removed from the dataframe.

The next step is to visualize the distribution of the variables and this will be done using box and whisker plots.

2.1 Distribution

3 Cluster Analysis

3.1 K-means

3.2 K-mediods

3.3 Clara

3.4 DBSCAN

4 Dimension Reduction

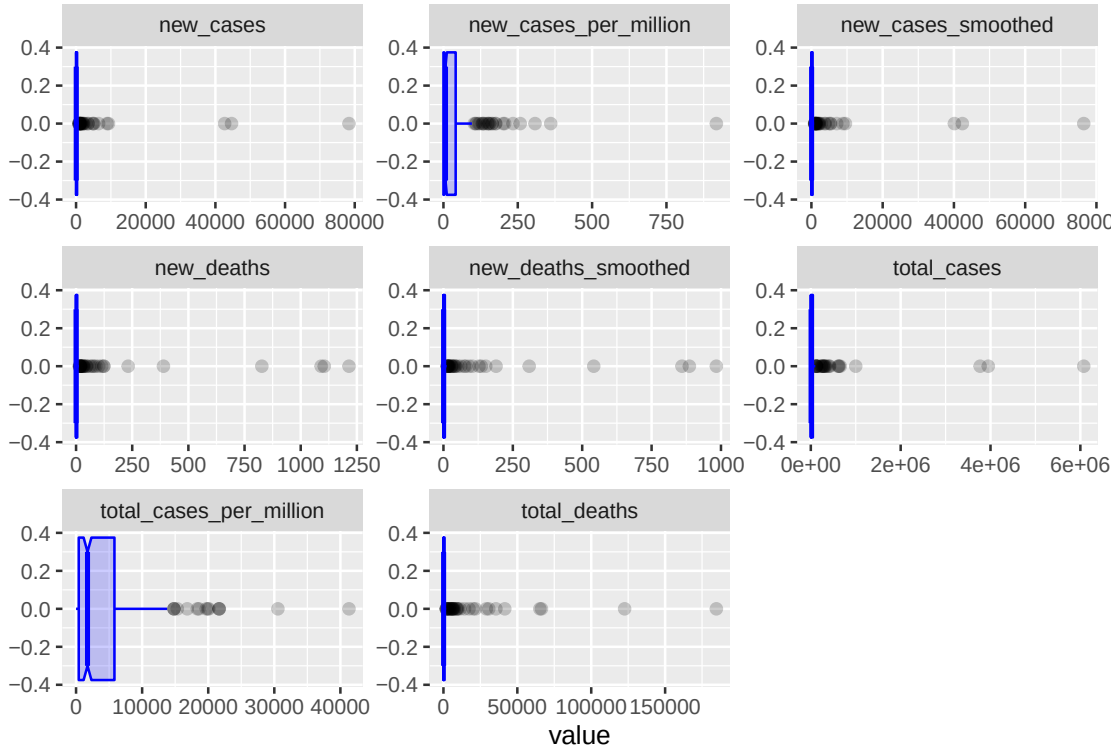
4.1 Results

The data shows that handwashing_facilties, extreme poverty, male smokers and female smokers are among the columns with the highest percentages of missing data.

```
# Plot box and whisker for all variables

df1 %>%
```

```
  keep(is.numeric) %>%
  select(c(1:8)) %>%
  gather() %>%
  ggplot(aes(value)) +
    facet_wrap(~ key, scales = "free") +
    geom_boxplot(
      # custom boxes
        color="blue",
        fill="blue",
        alpha=0.2,

        # Notch?
        notch=TRUE,
        notchwidth = 0.8,

        # custom outliers
        outlier.colour="black",
        outlier.fill="black",
        outlier.size=2
    )
```

```
## notch went outside hinges. Try setting notch=FALSE.
## notch went outside hinges. Try setting notch=FALSE.
```



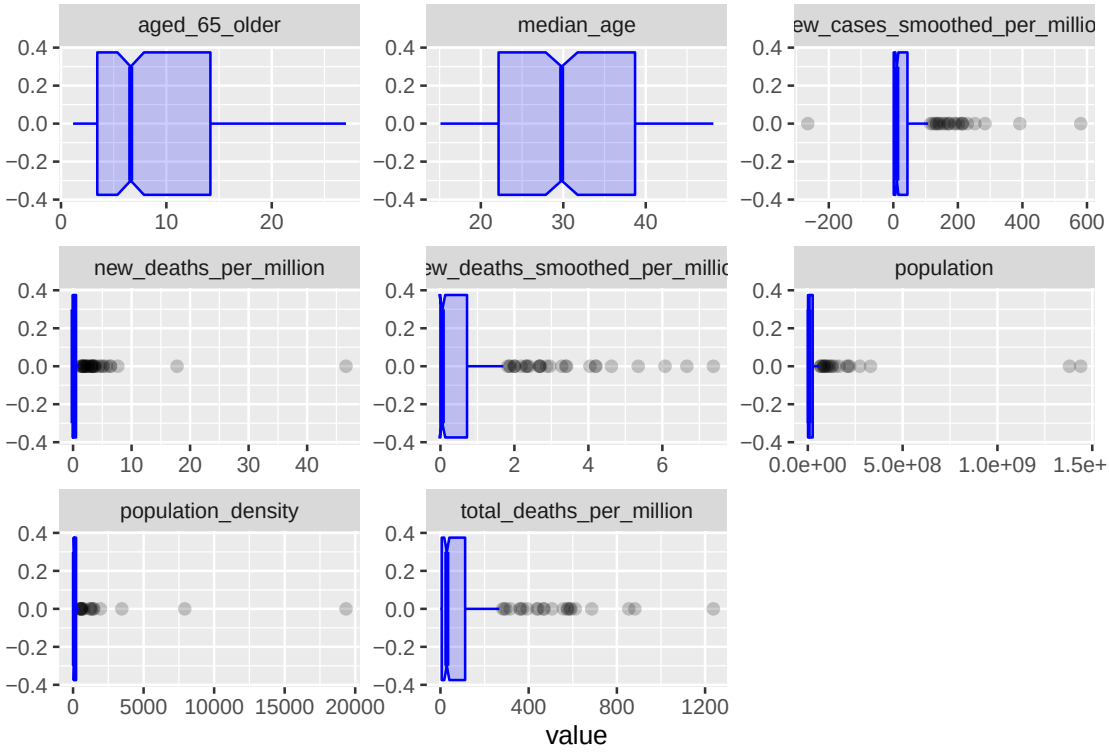
The first seven figures shpw that most of the data is concerntrated very close to less than a thousand except for total_cases_per_million which is more spread and between 0 and 5000. All the variables contain numerous outliers and this shows that we might need a clustering method that is robust and not affected by outliers.

```
df1 %>%
  keep(is.numeric) %>%
  select(c(9:16)) %>%
  gather() %>%
```

```
ggplot(aes(value)) +
  facet_wrap(~ key, scales = "free") +
  geom_boxplot(
          # custom boxes
       color="blue",
       fill="blue",
       alpha=0.2,

       # Notch?
       notch=TRUE,
       notchwidth = 0.8,

       # custom outliers
       outlier.colour="black",
       outlier.fill="black",
       outlier.size=2
  )
```

```
## Warning: Removed 62 rows containing non-finite values (stat_boxplot).
```

```
## notch went outside hinges. Try setting notch=FALSE.
## notch went outside hinges. Try setting notch=FALSE.
```



Almost half of the variables Median_age and aged_65_older values are much more spread. The numbers seem resasonable given that a smaller percentage of the population is age above 65 and usually the median age of most countries is expected to be below 50 given that there usually more young people than older people in country. Countries like Japan and Italy seem to have the highest percentage of older people with a median age of 48.2 and 47.9. The Niger and Uganda have the youngest population with a median age of 16.1 and 15.4. In genral European countries seems to have an older society while Africa countries have a much younger population.

Monaco and Singapore have the highstest population densities while China has the highest population. As a result these countries stand out as outliers with regards to population variables.

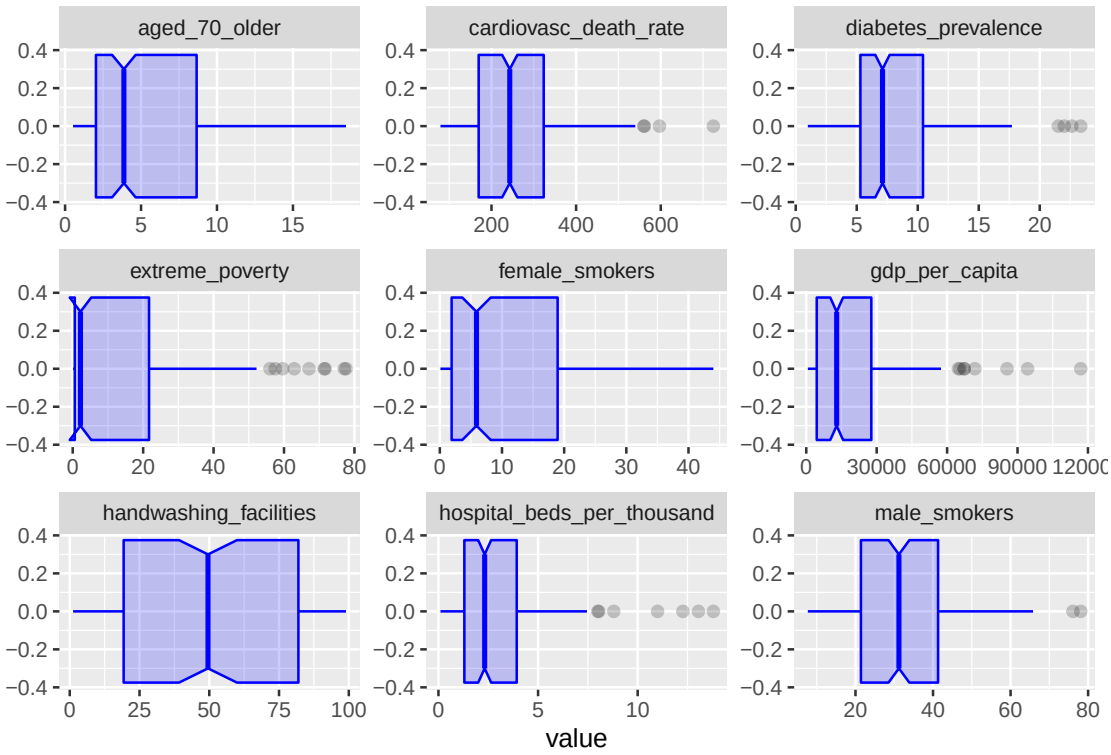
New Deaths per million are hightest in north and south america.

New cases smoothed per million show a negative number(outlier). This is most likely a entry error and the values will be replaced with a regional average.

```
df1 %>%
  keep(is.numeric) %>%
  select(c(17:25)) %>%
  gather() %>%
  ggplot(aes(value)) +
    facet_wrap(~ key, scales = "free") +
    geom_boxplot(
             # custom boxes
          color="blue",
          fill="blue",
          alpha=0.2,

          # Notch?
          notch=TRUE,
          notchwidth = 0.8,

          # custom outliers
          outlier.colour="black",
          outlier.fill="black",
          outlier.size=2
    )
```

```
## Warning: Removed 483 rows containing non-finite values (stat_boxplot).
```

```
## notch went outside hinges. Try setting notch=FALSE.
```

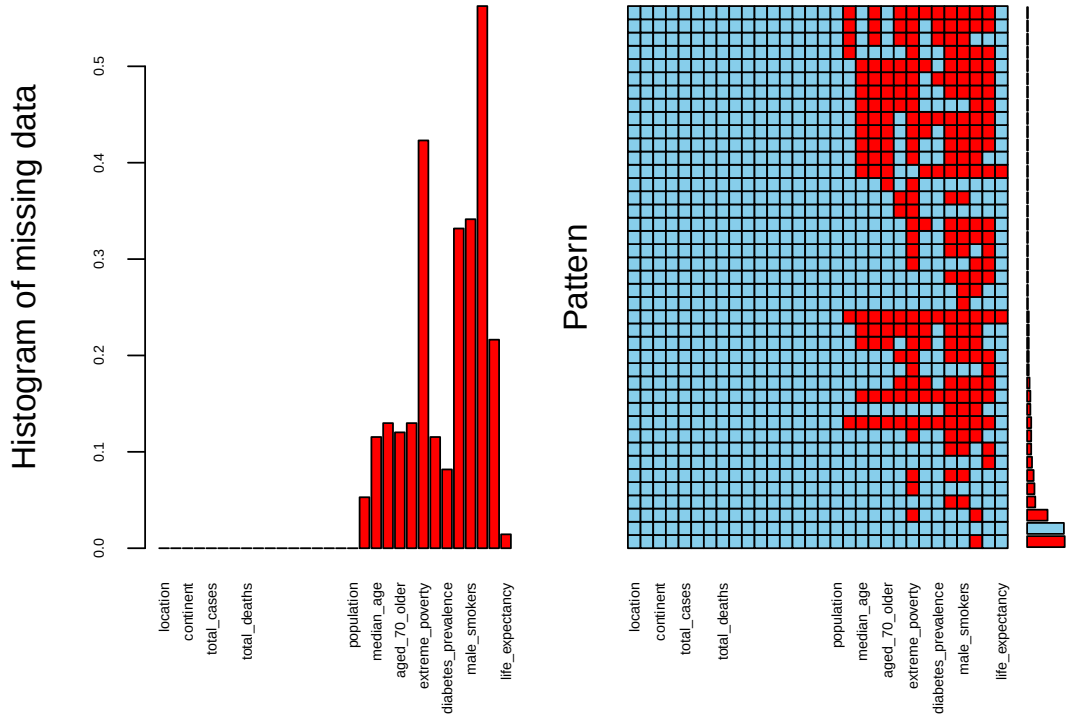


The observations for these variables look more spread with less outliers. The variables also suggest that there are generally more male smokers than female smokers. Futhermore, the data show that it is mostly european countries that are wealthy or less poor. The numbers seem to be in the expected ranges.

4.2 Missing values

Running a summary on the dataframe shows that there are various columns with missing data.

```
missing_data_visual <- vis_miss(df1, sort_miss = T) # Visualise to see rows/columns with missing data
#missing_data_visual

aggr_plot <- VIM::aggr(df1, col=c('navyblue','red'), numbers=TRUE, sortVars=TRUE, labels=names(df1), cex.axis=
plot(aggr_plot, cex.axis=.5, gap=3, ylab=c("Histogram of missing data","Pattern"))
```




```
# Creating table
missing_data_table = aggr_plot$missings %>%
  filter(Count > 0) %>%
  arrange(desc(Count)) %>%
  mutate(Percentage = Count/208)

knitr::kable(missing_data_table, digits = 5, caption = "Missing data visualisation") %>%
  kable_styling(full_width = F, font_size = 7) %>%
  column_spec(1, border_left = T) %>%
  column_spec(3, border_right = T)
```

\begin{table}

\caption{(#tab:missing_data_tab)Missing data visualisation}

Variable	Count	Percentage
handwashing_facilities	117	0.56250
extreme_poverty	88	0.42308
male_smokers	71	0.34135
female_smokers	69	0.33173
hospital_beds_per_thousand	45	0.21635
aged_65_older	27	0.12981
gdp_per_capita	27	0.12981
aged_70_older	25	0.12019
median_age	24	0.11538
cardiovasc_death_rate	24	0.11538
diabetes_prevalence	17	0.08173
population_density	11	0.05288
life_expectancy	3	0.01442

\end{table}

As show in Table @ref(missing_data_tab), a number of variables have a very high number of missing values including handwashing_facilities, extreme_poverty, male smokers and female smokers. It is generally not ggod working with data or columns that have a high percentage of missing data. Therefore various methods of imputing the missing data will be observed in the next section.

4.3 Imputation by data scrapping

There are various sources that were used to collect the data including the European Center for Disease Prevention and Control. However some of the data can be found on wikipedia. Although the data will not exactly be the same but the values will be very similar.

Over 78% of rows have missing data therefore simply omitting the rows with missing will lead to a loss a huge size of important data.

Population density missing values were imputed using the data from wikipedia (**???** Density)

https://en.wikipedia.org/wiki/List_of_countries_and_dependencies_by_population_density

It is important to indicate that the data was given in 2019 but given that population densities change very slowly, this is better than replacing with an average/median.

Median age missing values will be imputed using the data found from the following wikipedia site:

https://en.wikipedia.org/wiki/List_of_countries_by_median_age

Median ages data was added using data from the 2019 data

https://www.cia.gov/library/publications/the-world-factbook/fields/343rank.html

Only data for Syria was found.

Median age above 75 years old was ignored

Searching the source of the data maps to a site:

https://github.com/owid/covid-19-data/blob/master/public/data/owid-covid-codebook.csv which says the gdp_per_capita Gross domestic product at purchasing power parity (constant 2011 international dollars), most recent year available. Given that the most recent data from the world bank (https://databank.worldbank.org/source/jobs/Series/NY.GDP.PCAP.PP.KD#) which matches this data is for the year 2016, the data was used for imputing the data. Note the data is not available

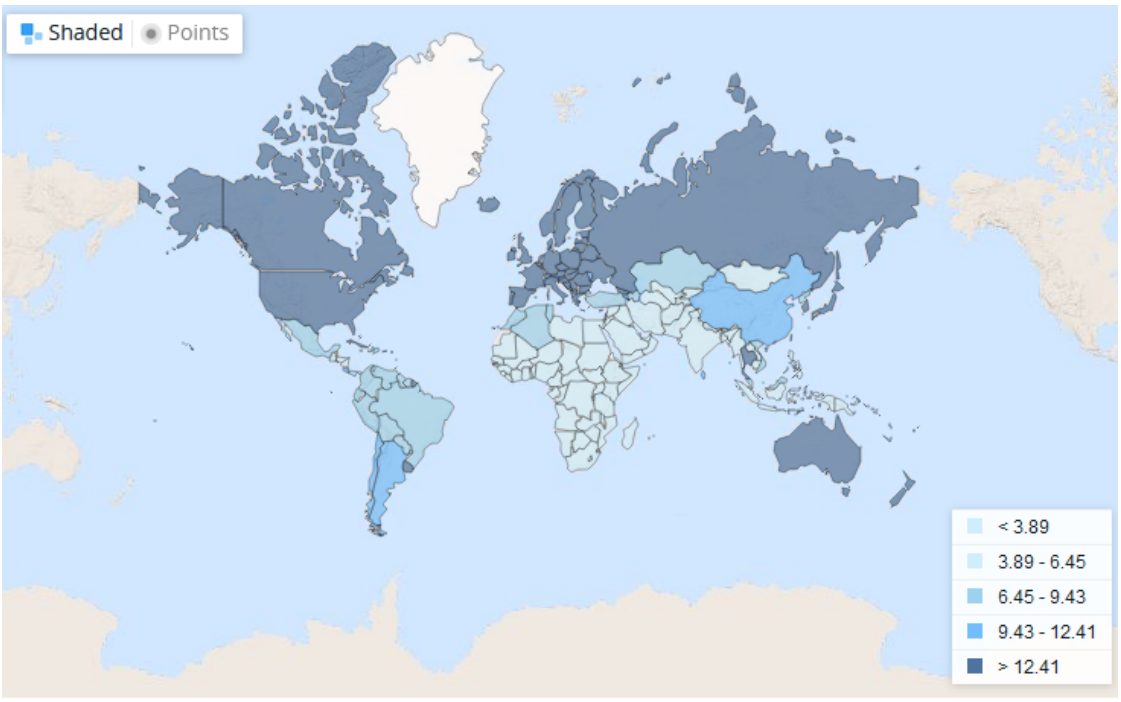
Extreme poverty is a population with an income of less than $1.90 a day. From the data the Share of the population living in extreme poverty, most recent year available since 2010. The data from the following site https://en.wikipedia.org/wiki/List_of_countries_by_percentage_of_population_living_in_poverty the data is very similar to that available without missing values, therefore it will be better to use the data for imputing.

Most countries with 0 percent were having missing values

This was skipped due to insufficient data to fill in the missing values.

female and male smokers info not enough handwashing facilities skipped because data is not enough. hospital bed per thousand informatiion was found to be too old for example Chad which had data for 2005. Given that there are many years between the time the data was collected and this year the data was ignored.

From the world bank data on life expectancy (https://data.worldbank.org/indicator/SP.DYN.LE00.IN), the life expectancy of Guernsey, Jersey and Kosovo are 82.6, 80.6, 71.95 years respectively.

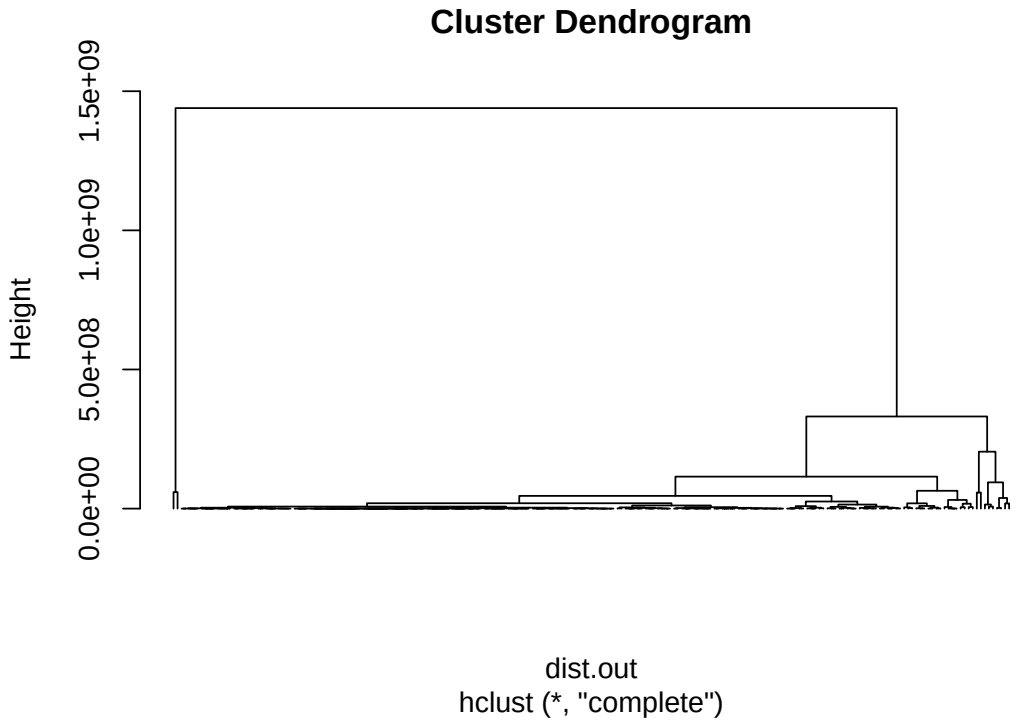


The results show that handwashing_facilites is missing 56.25% of the values followed by extreme_poverty missing 42.3% of the values. In total, 78.4% of the samples have missing values.
The histograms also suggest that there is a negative value in new_cases_smoothed_per_million.
Upon further investigation, Luxembourg seems to have negative values for new_cases_smoothed and new_cases_smoothed_per_million. This is probably because the new_cases_smoothed_per_million is derived from new_cases_smoothed.
The mice package allows multivariate imputation by chained equations
MICE assumes that the missing data are Missing at Random (MAR), which means that the propensity for a data point to be missing is not related to the missing data but is related to some of the observed data.
https://www.theanalysisfactor.com/mar-and-mcar-missing-data/
Therefore it is important to determine which columns have data points that may be missing at random. The imputation process is not a one size fits all method.
There is also the Amelia package which also makes an assumption that the missing data is random in nature (MSR)
Hmisc is a multiple purpose package useful for data analysis, high – level graphics, imputing missing values, advanced table making, model fitting & diagnostics (linear regression, logistic regression & cox regression) etc. Hmisc assumes linearity in the variables being predicted.
mi (Multiple imputation with diagnostics) package provides several features for dealing with missing values. Like other packages, it also builds multiple imputation models to approximate missing values. And, uses predictive mean matching method like the mice package.
The following variables need need impiutation using some of the methods.

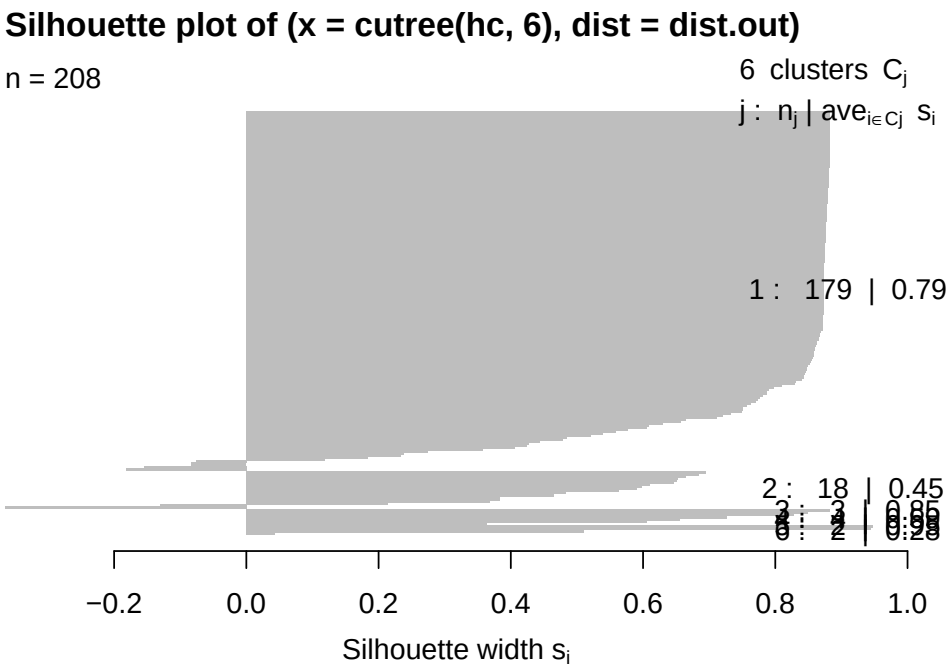
4.4 female_smokers and male_smokers, cardiovasc_death_rate and diabetes prevalence are MACR variables therefore the MICE package will be used on this data.

various packages will be used for imputation and the one that yields the best result will be chosen.
This will be support for dimension reduction.
It is not enough to try out one package for imputing data therefore try out more! The question is if any one value of the observation is missing, will it affect the missingness of a specific variable.
The percentage of population aged_65_older looks to have regional patterns therefore it is justified to use regional medians/averages for imputation. Since aged_70_older can is closely related to aged_65_older it is decided to use the same method.
The randomizing, re-running, averaging and final run is a very good advice.

```
# Hierarchical clustering
# Compute pairwise distance matrices
dist.out <- dist(data.out,
                 method = "euclidean")
# Hierarchical clustering results
hc <- hclust(dist.out,
             method = "complete")
# Visualization of hclust
plot(hc, labels = F,-1)
```

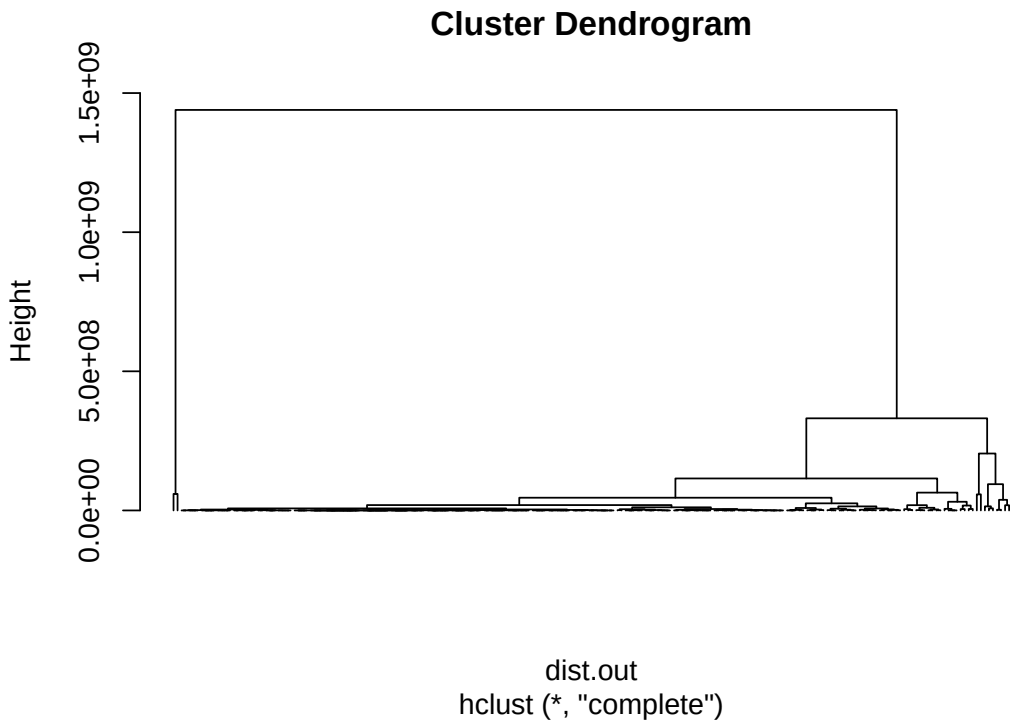



```
plot(silhouette(cutree(hc,6),dist.out))
```

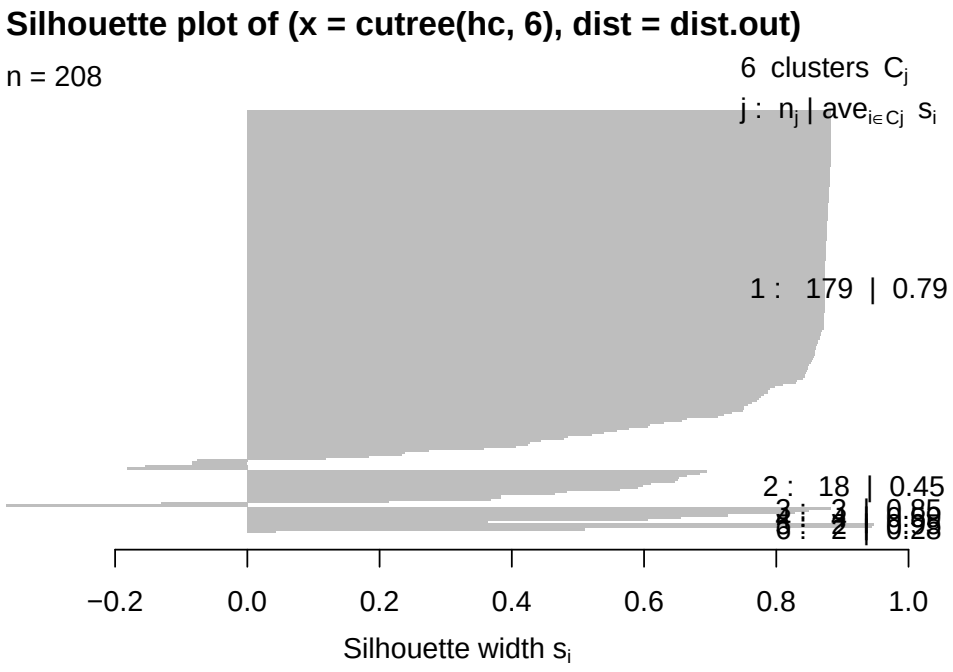


Average silhouette width : 0.75

```r
hc <- hclust(dist.out,
             method = "complete")
# Visualization of hclust
plot(hc, labels = F,-1)
```

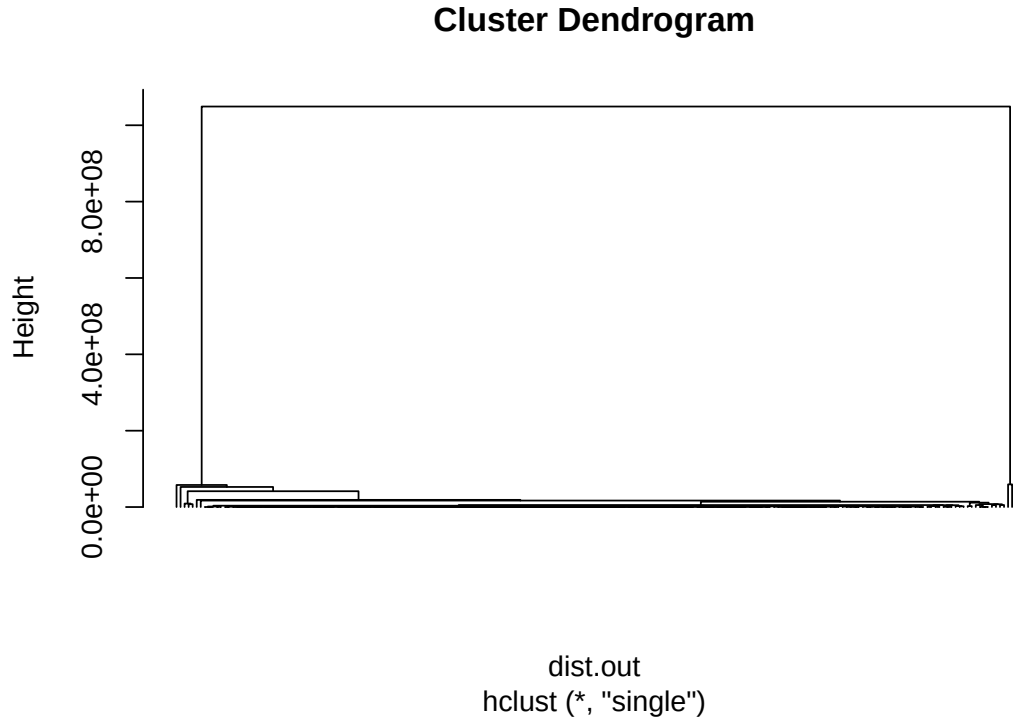


```
plot(silhouette(cutree(hc,6),dist.out))
```



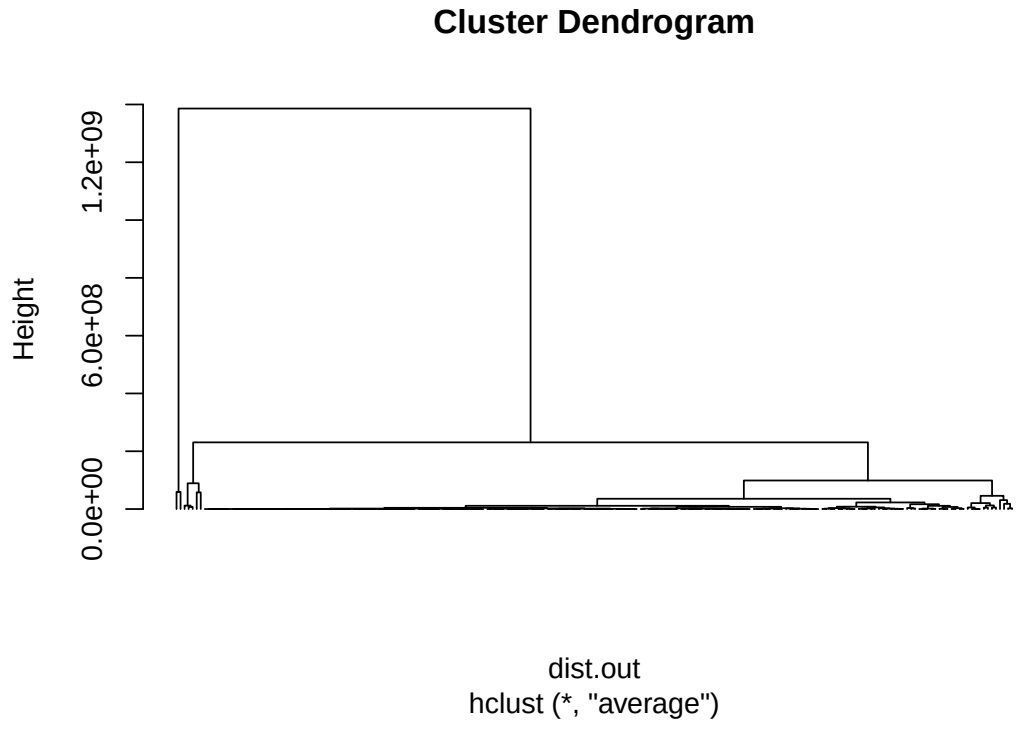
Average silhouette width : 0.75

```
hc <- hclust(dist.out,
             method = "single")
# Visualization of hclust
plot(hc, labels = F,-1)
```

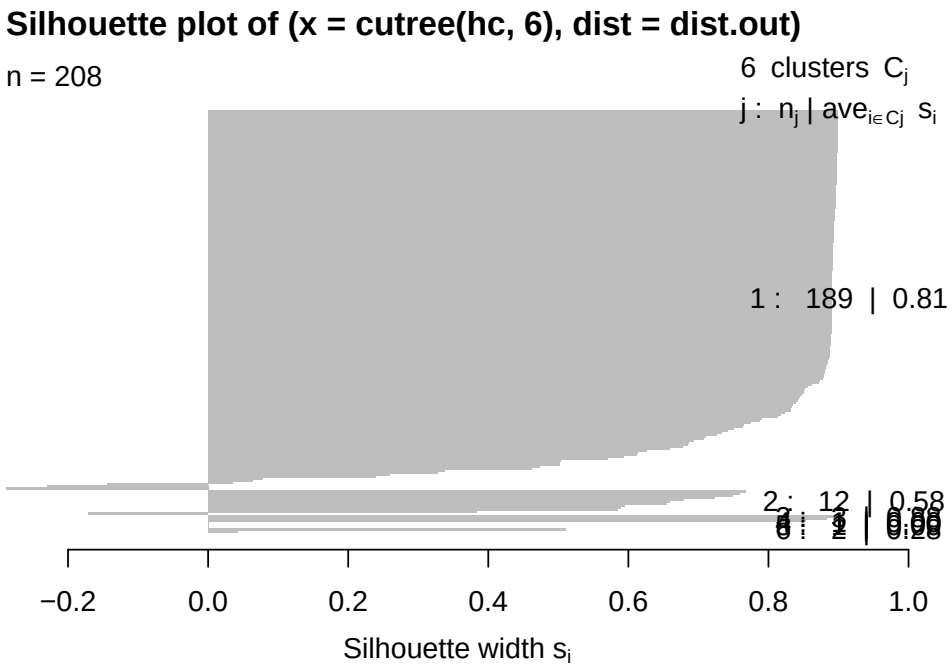


```
plot(silhouette(cutree(hc,6),dist.out))
```


```
hc <- hclust(dist.out,
             method = "average")
# Visualization of hclust
plot(hc, labels = F,-1)
```

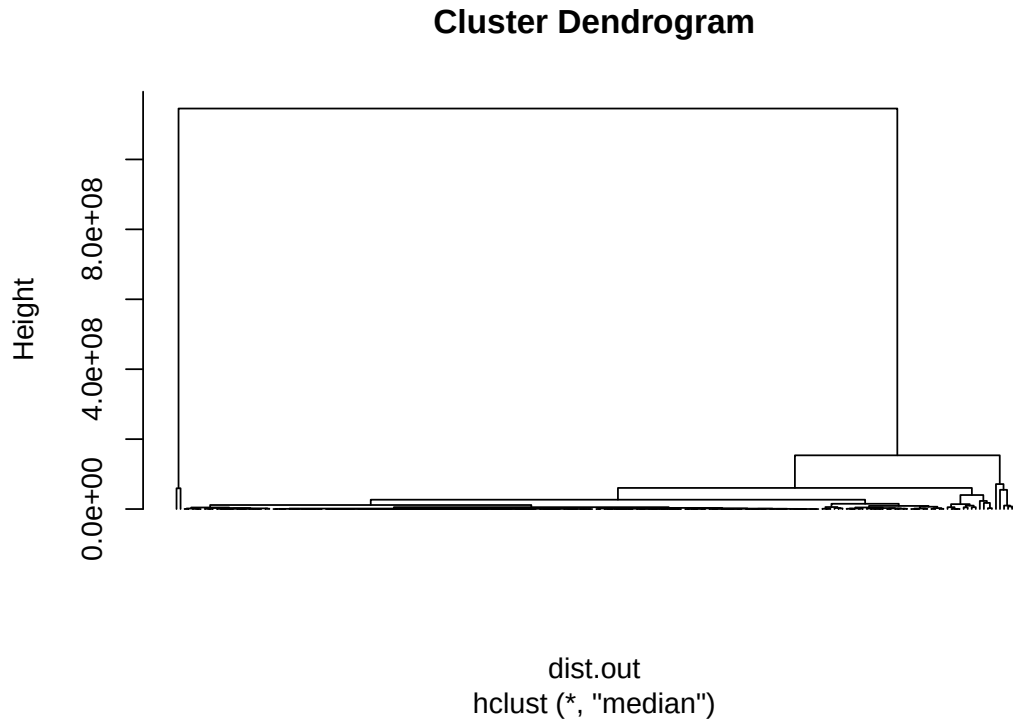


```
plot(silhouette(cutree(hc,6),dist.out))
```

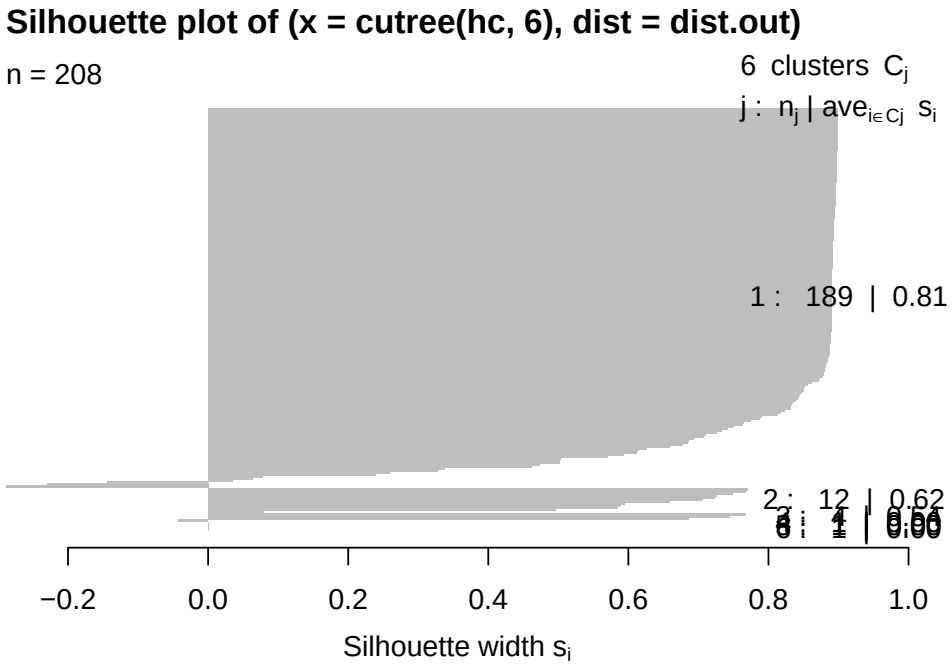


Average silhouette width : 0.79

```
hc <- hclust(dist.out,
             method = "median")
# Visualization of hclust
plot(hc, labels = F,-1)
```

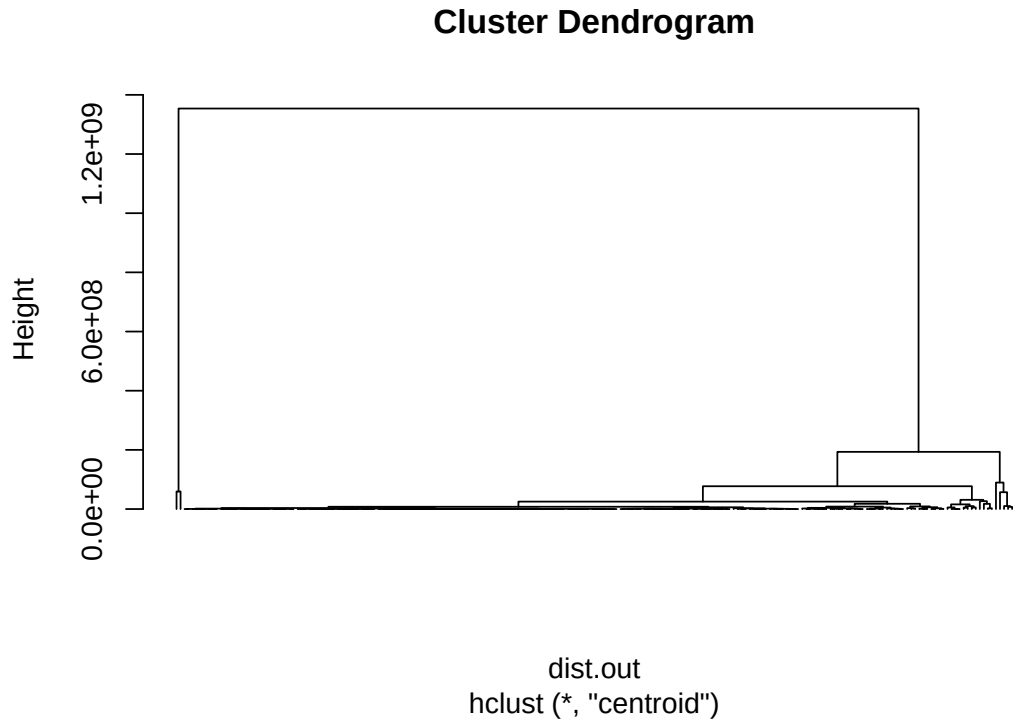


```r
plot(silhouette(cutree(hc,6),dist.out))
```

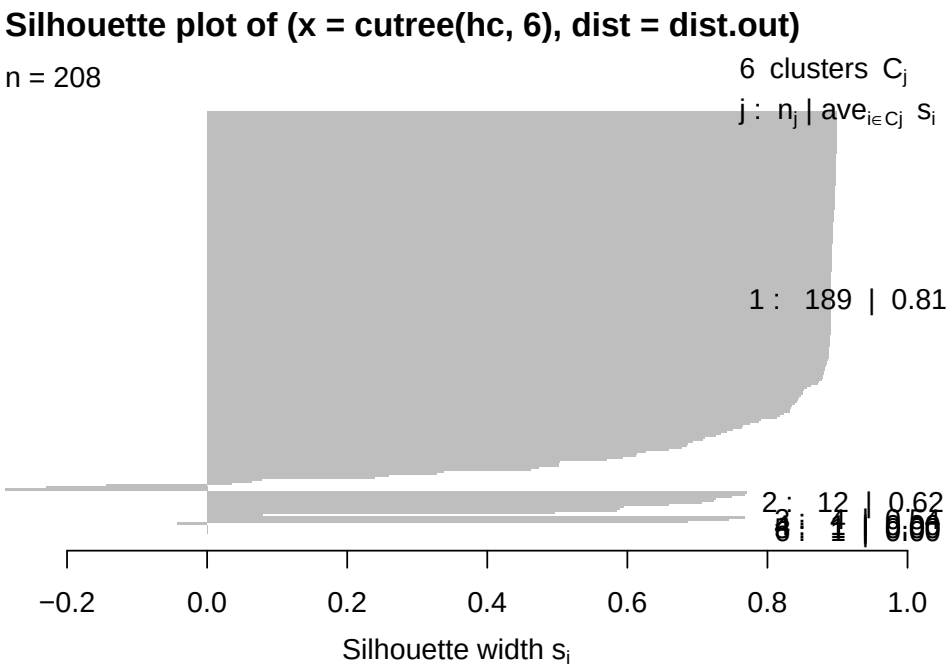


Average silhouette width : 0.78

```r
hc <- hclust(dist.out,
             method = "centroid")
# Visualization of hclust
plot(hc, labels = F,-1)
```

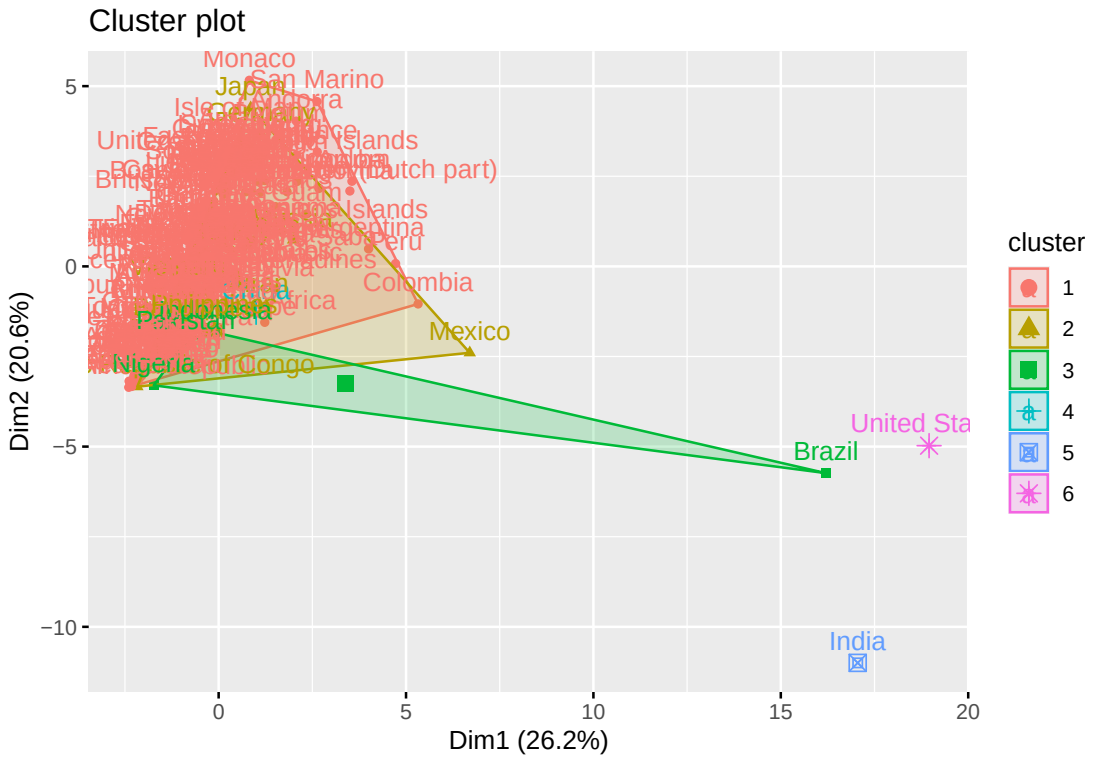


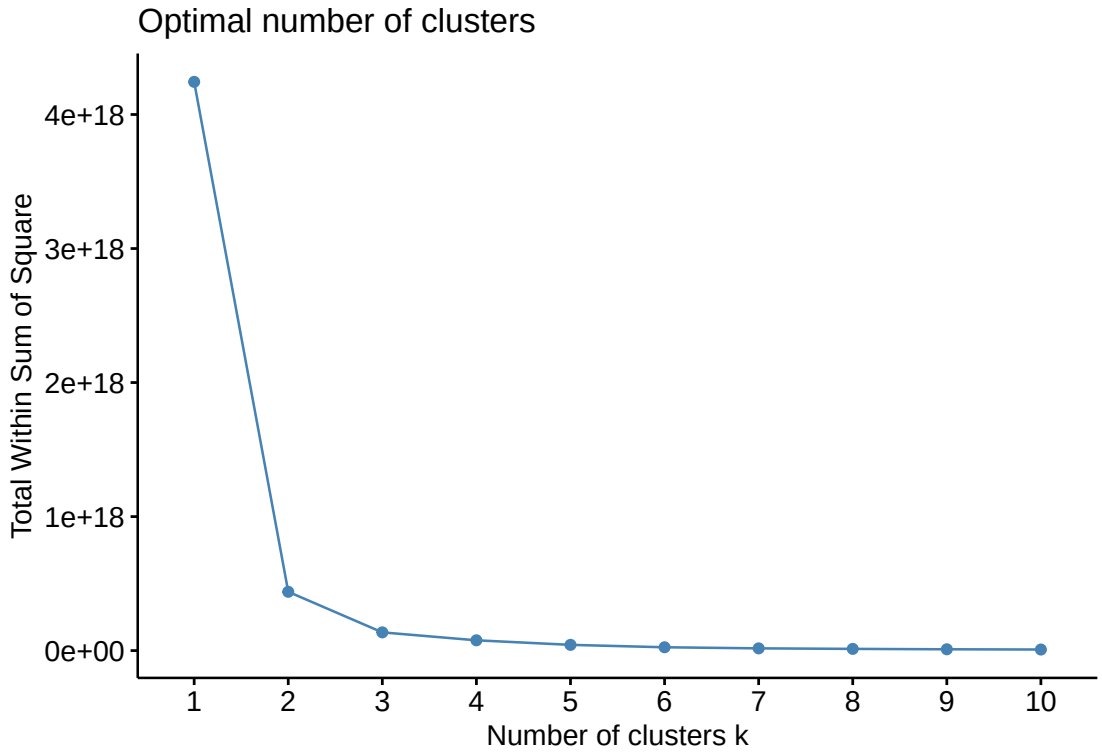
```r
plot(silhouette(cutree(hc,6),dist.out))
```



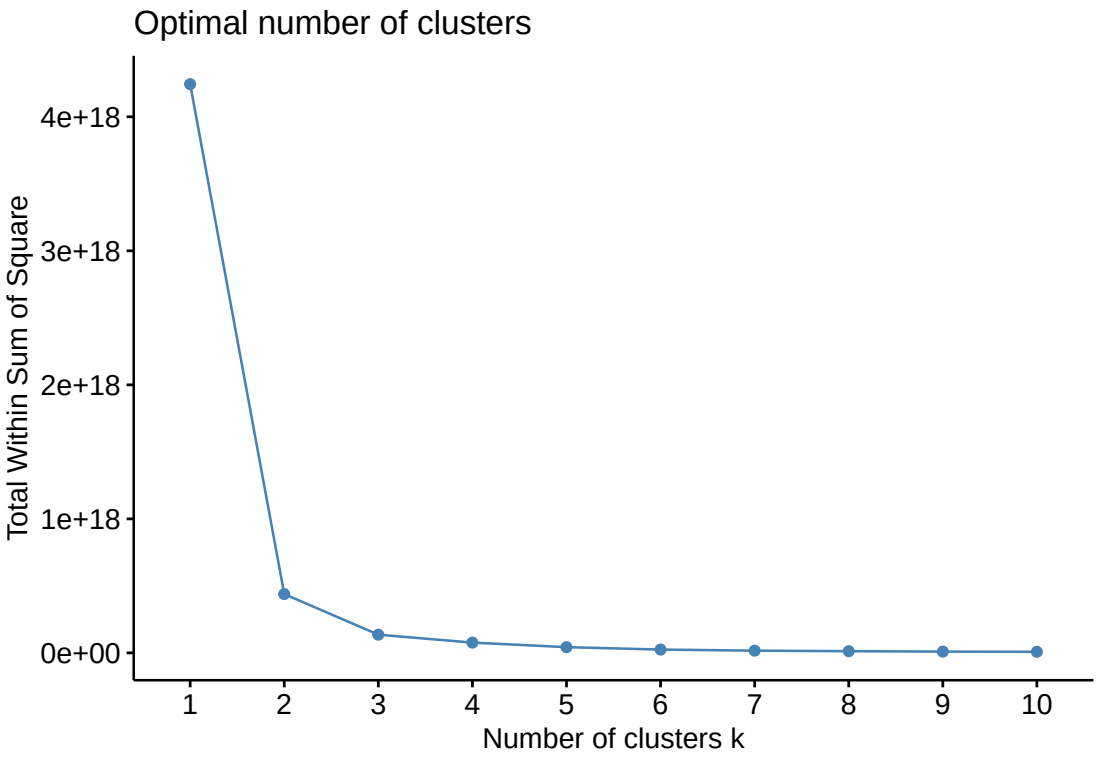
Average silhouette width : 0.78

```r
# #
# rect.hclust(hc, k = 6, border = 2:3)
#
# hcd <- as.dendrogram(hc)
# # Define nodePar
# nodePar <- list(lab.cex = 0.6,
#                 pch = c(20, 19),
#                 cex = 0.7,
#                 col = c("green","yellow"))
# plot(hcd,
#      xlab = "Height",
#      nodePar = nodePar,
#      main = "Cluster dendrogram",
#      edgePar = list(col = c("red","blue"), lwd = 2:1),
#      horiz = TRUE)
```





The dataset is very small and this results in Clara performing worse than PAM.
The randomizing, re-running, averaging and final run is a very good advice.



The dataset is very small and this results in Clara performing worse than PAM.
This code was optimised for reading
European Center for Disease Prevention and Control. 2020. *Coronavirus Source Data.*
https://ourworldindata.org/coronavirus-source-data.