Final project: Step 1 & 2

Danyu Zhang, Limingrui Wan, Daniel Alonso

December 14th, 2020

Contents

Importing libraries

```r
library(dplyr)
library(ggplot2)
library(reshape2)
library(PerformanceAnalytics)
library(gridExtra)
library(stringr)
library(foreach)
library(MASS)
library(andrews)
library(mice)
library(factoextra)
library(corrplot)
library(plotrix)
library(corpcor)
library(ggpubr)
library(ca)
library(tidyverse)
library(corpcor)
library(RSpectra)
library(factoextra)
library(cluster)
library(mclust)
```

Cluster Analysis

Pre-processing Data

We define colors for plots

```r
color_1 <- "deepskyblue2"
color_2 <- "seagreen2"
color_3 <- "orange2"
color_4 <- "darkorchid4"
color_5 <- "firebrick2"
color_6 <- 'red'
```

As we stated in *step 1*, there are some variables as they are redundant transformations of other columns. For different cases we may need to use standardized data and cases where the model only work with quantitative variables. we need to build a few subsets. And we need to impute the missing values.

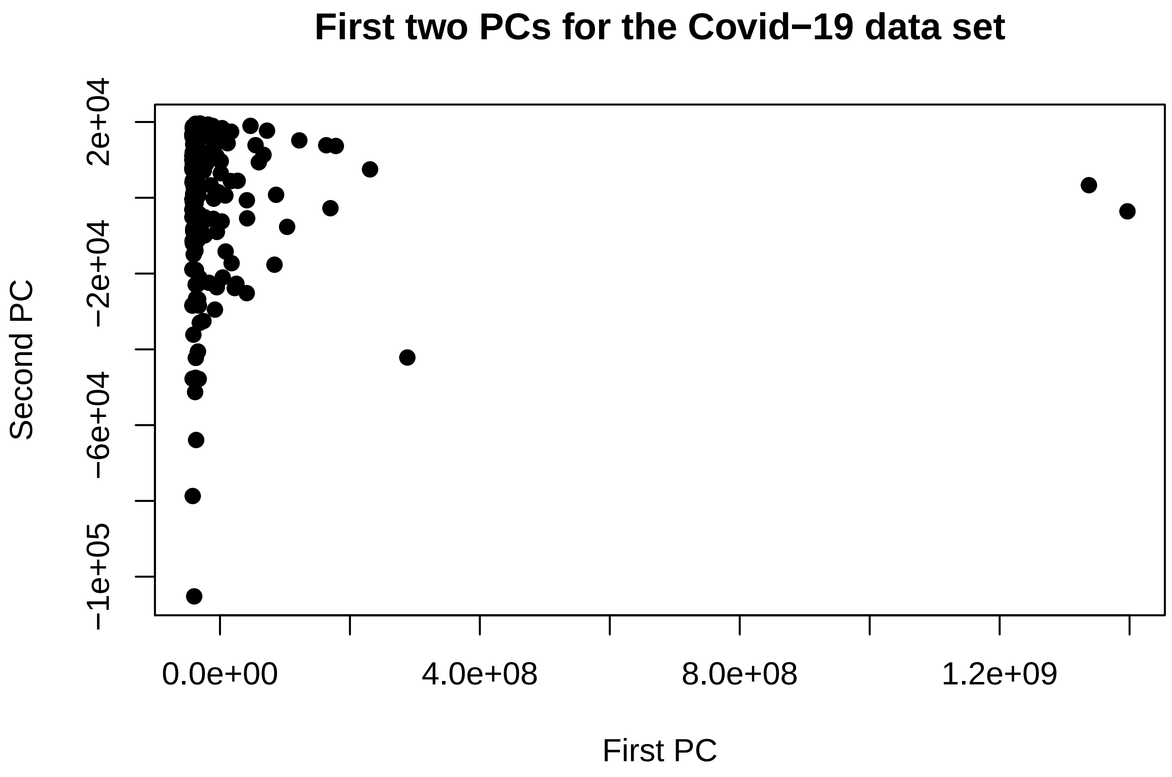
```r
data2 <- read.csv('./data/data_imp.csv', header=TRUE)
data <- data2[,2:length(names(data2))]
data$continent=as.factor(data$continent)
data$development=as.factor(data$development)
data_cate <- subset(data, select = c(continent,development,location))
data_quan <- subset(data, select = -c(continent,development,location))
```

PCA analysis

computing PCAs

To visualize the results, we need to obtain the first two PCAs.

```
#> Estimating optimal shrinkage intensity lambda.var (variance vector): 0.3941
#>
#> Estimating optimal shrinkage intensity lambda (correlation matrix): 0.0367
```



We can't tell how many groups from this picture, then we need to find that with multiple methods.

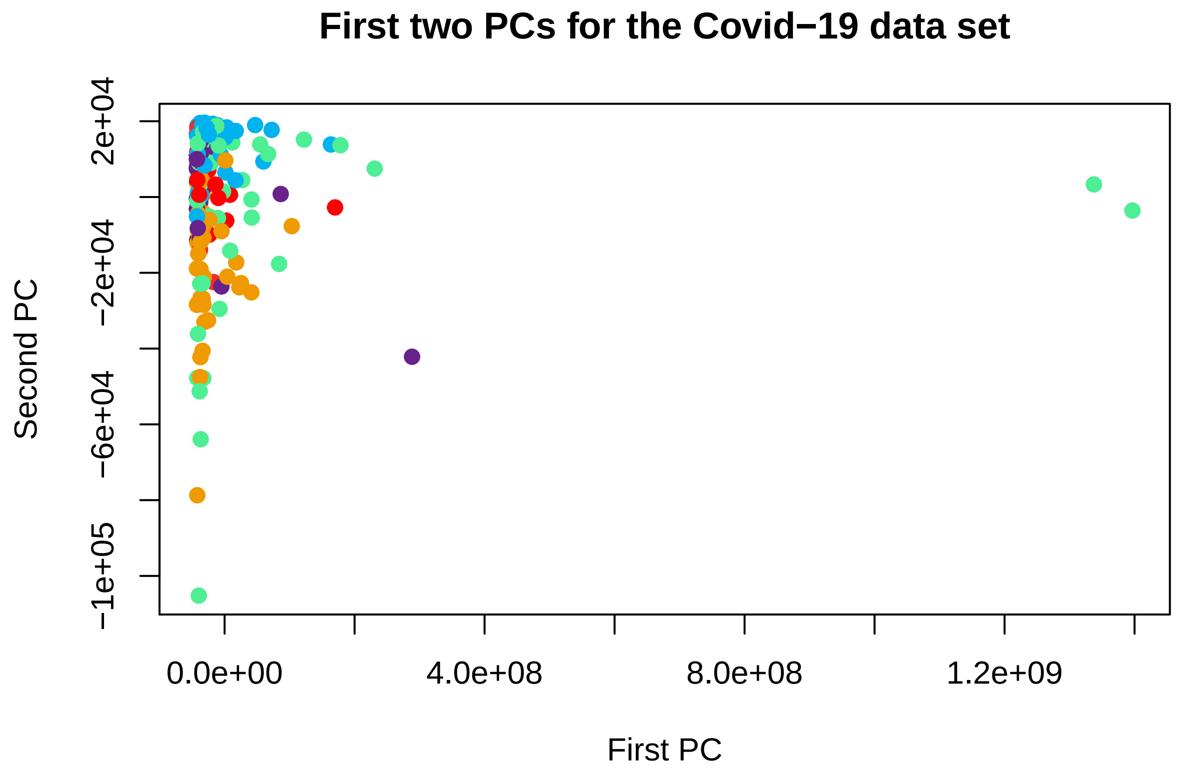
Partitional clustering

Partition of dataset

We firstly check how is the data grouped by the categorical variables.

Continent

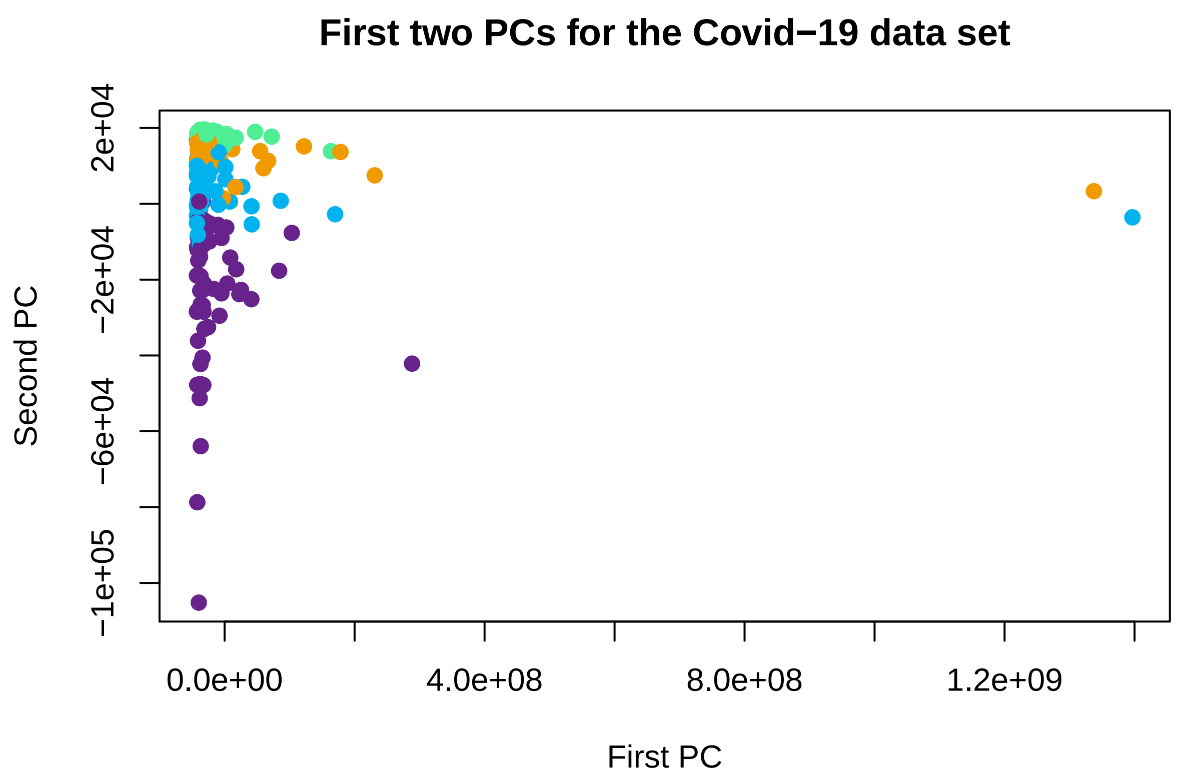
```
#>
#>        Africa          Asia        Europe North America       Oceania
#>            53            45            42            23             6
#> South America
#>            12
```



No sign of groups, i.t. we can't get information by knowing the location of a country.

development

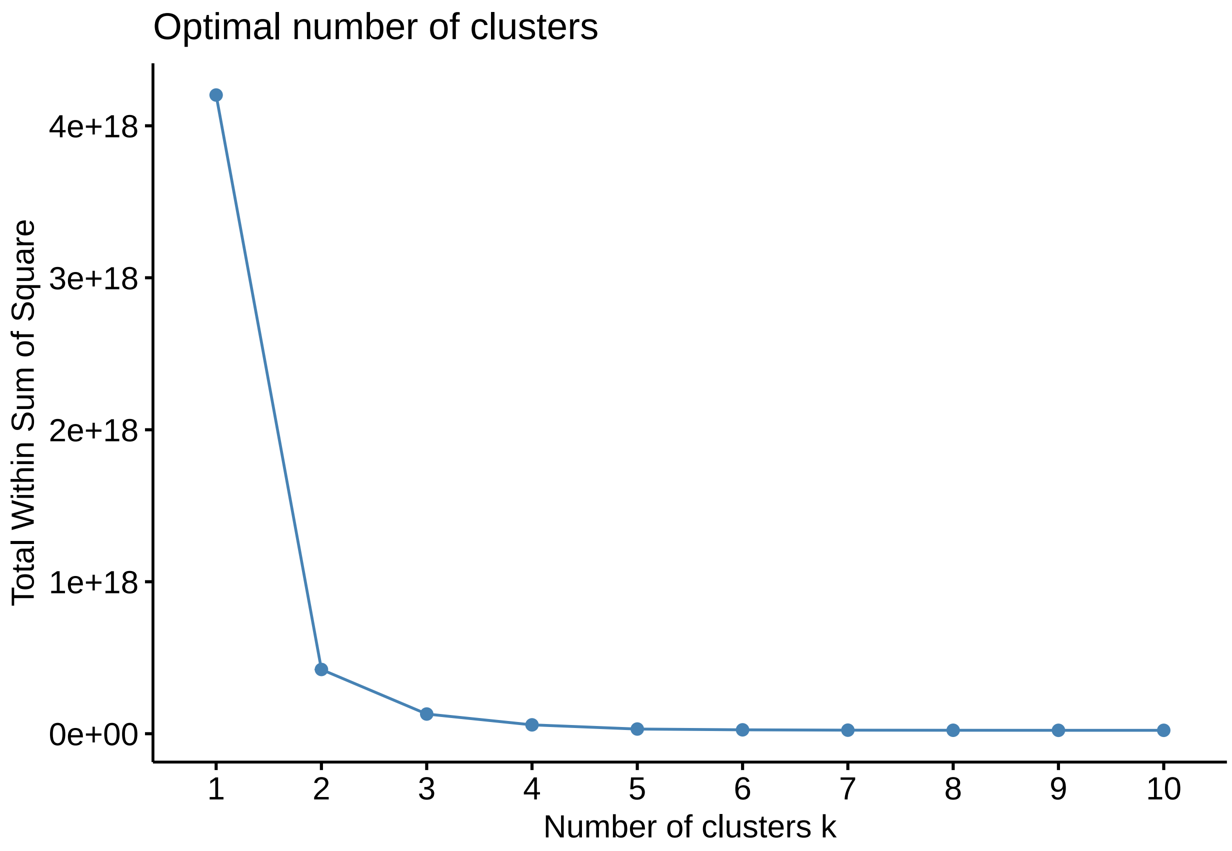
```
#>
#>      high       low    medium very high
#>        49        38        36        58
```



In this plot, groups are not separated well, the borders are not clear.

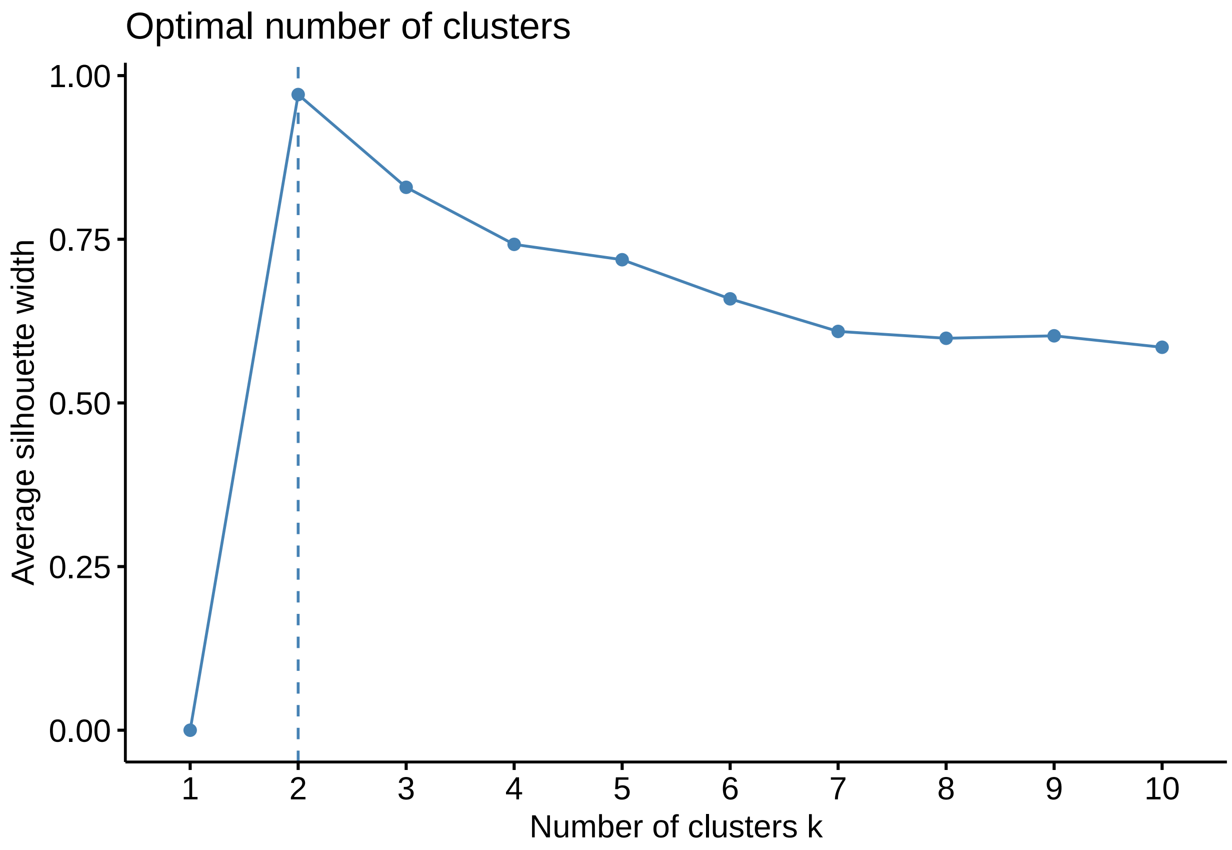
Select k

WSS



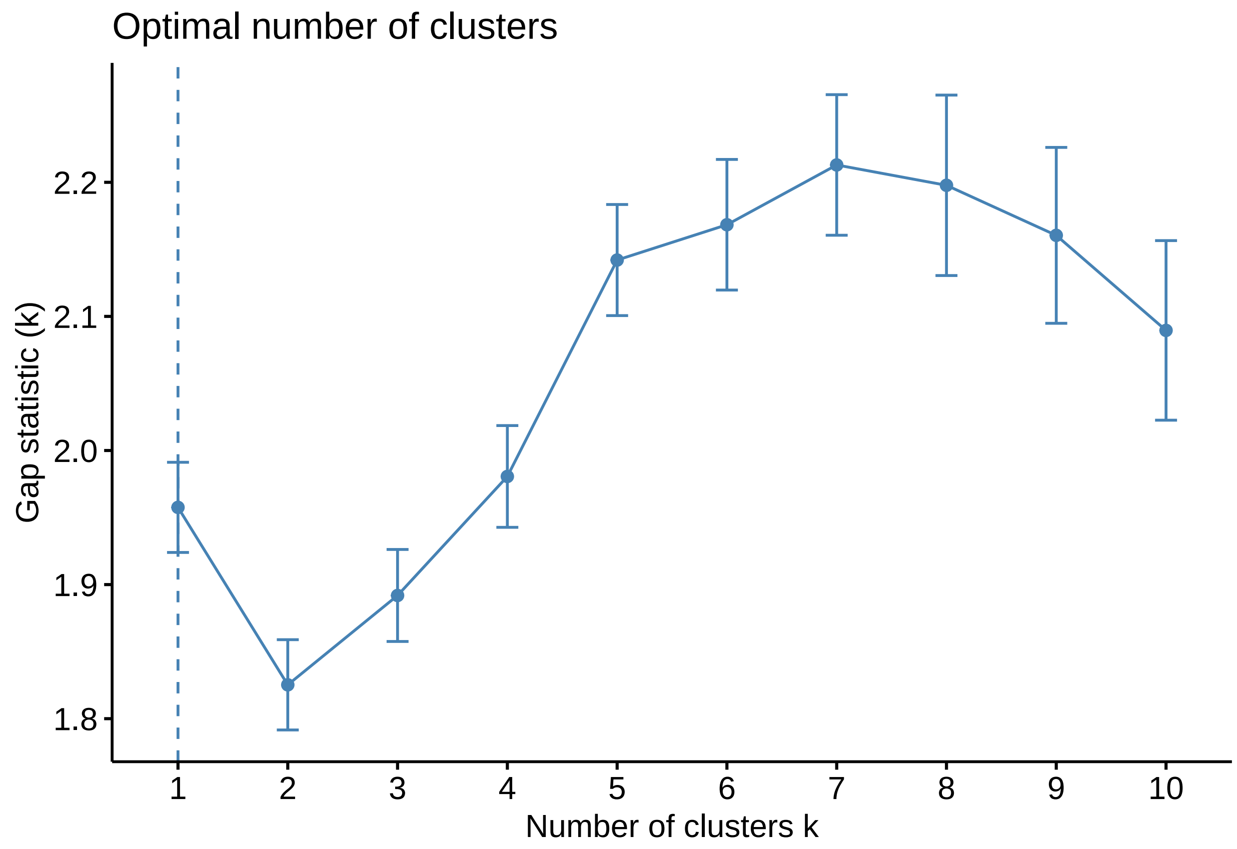
there is no optimal solution from WSS

Silhoutte



it sugguest us to set k into 2.

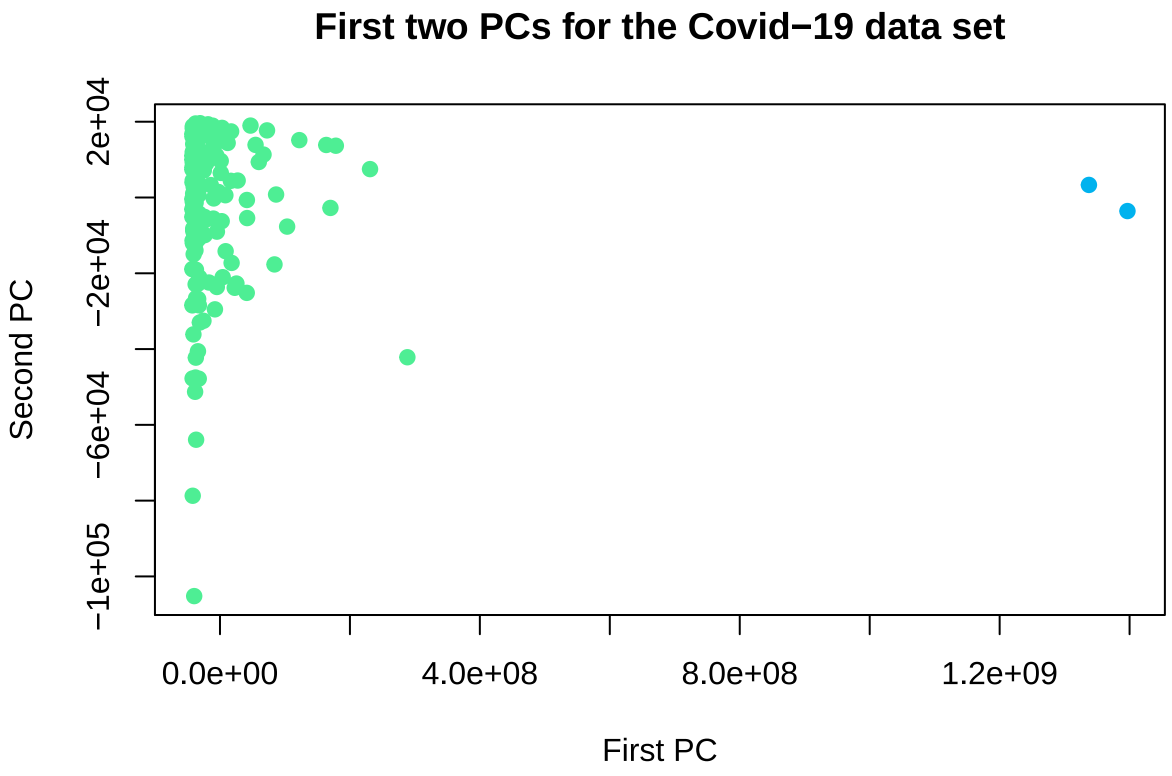
Gap Statistic



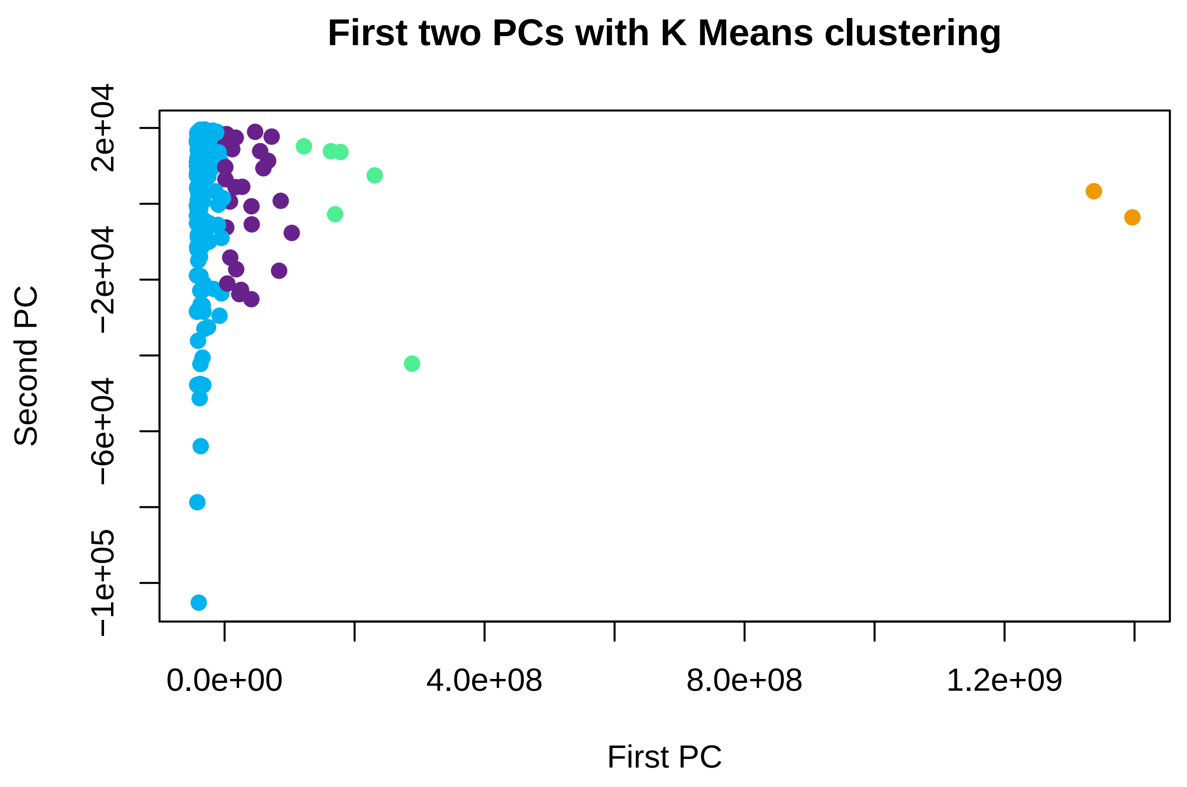
the result is 1, but we can't set the number of cluster to be 1 , otherwise, it makes no sense.

The K-means algorithm

Notice that in our data, there are 3 categorical variables: but one of them is the names;one of them is the continent, which is irrelevant;one is development, but it's simple determined by numerical variable *develop*. so we only choose the rest of variable which all are quantitative. and we have only 181, not need to apply **CLARA**.



we can try to increase k=4 because we have a categorical variable *development*, we can check the model with it.



we now have a better clustering result.

Kmeans Analysis

```
#>
#>    1   2
#>    2 179
```

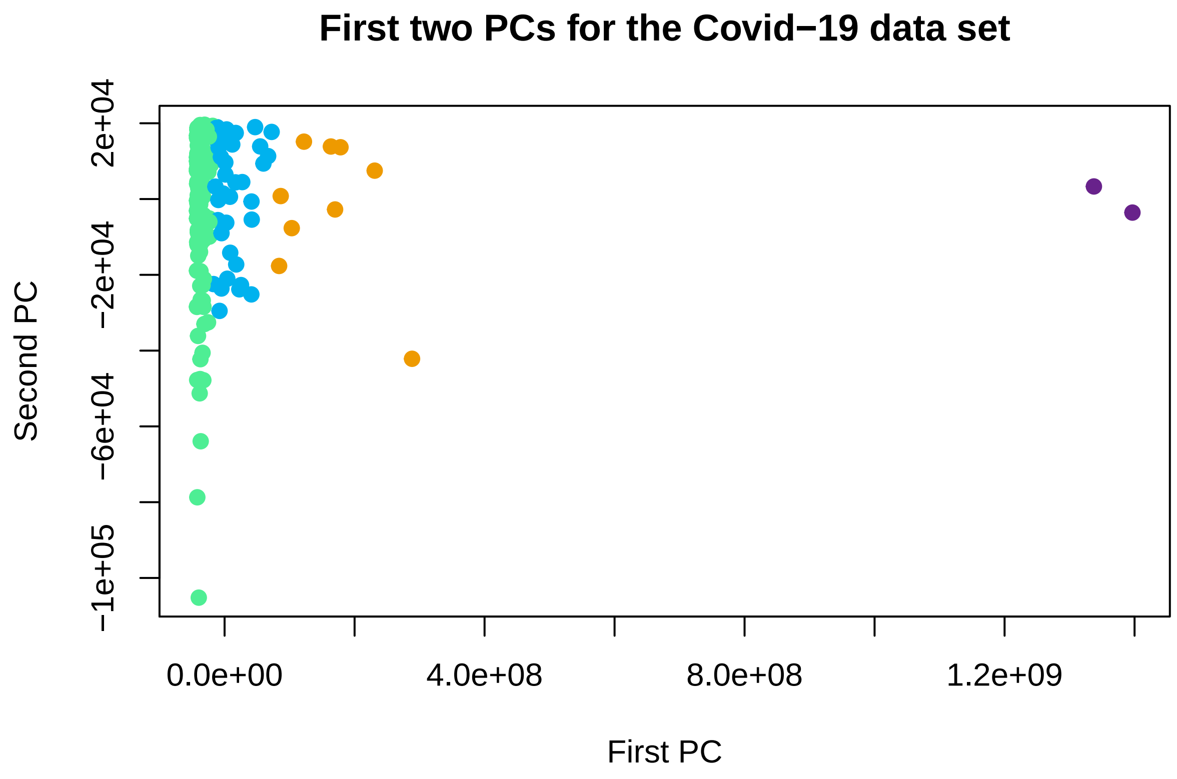
```
#>                                  [,1]         [,2]
#> total_cases_per_million    8.367026e+03 1.000707e+04
#> new_cases_per_million      1.307603e+02 5.319317e+01
#> total_deaths_per_million   1.365347e+02 2.629343e+02
#> stringency_index           5.315247e+01 5.918167e+01
#> population                 9.910324e+06 2.348012e+08
#> population_density         2.065009e+02 3.227617e+02
#> median_age                 3.015959e+01 2.836667e+01
#> aged_65_older              8.590082e+00 6.938000e+00
#> gdp_per_capita             1.877996e+04 1.556913e+04
#> extreme_poverty            1.203562e+01 1.101667e+01
#> cardiovasc_death_rate      2.658733e+02 2.623268e+02
#> diabetes_prevalence        7.972260e+00 7.395000e+00
#> hospital_beds_per_thousand 2.695055e+00 1.351667e+00
#> life_expectancy            7.259110e+01 7.016833e+01
#> human_development_index    7.060753e-01 6.798333e-01
```

when k = 2, we can see obvious difference between towo group.

After we tuning K into 4, it has a more interesting result, we can also characterize them with some features: from 1 to 4 means from lowest(fewest) to highest(most).

cluster	cases	death rate	economic	average age	medical resources	stringency
cluster1	1	1	1	1	1	1
cluster2	2	3	3	4	4	3
cluster3	3	2	4	2	2	2
cluster4	4	4	2	3	3	4

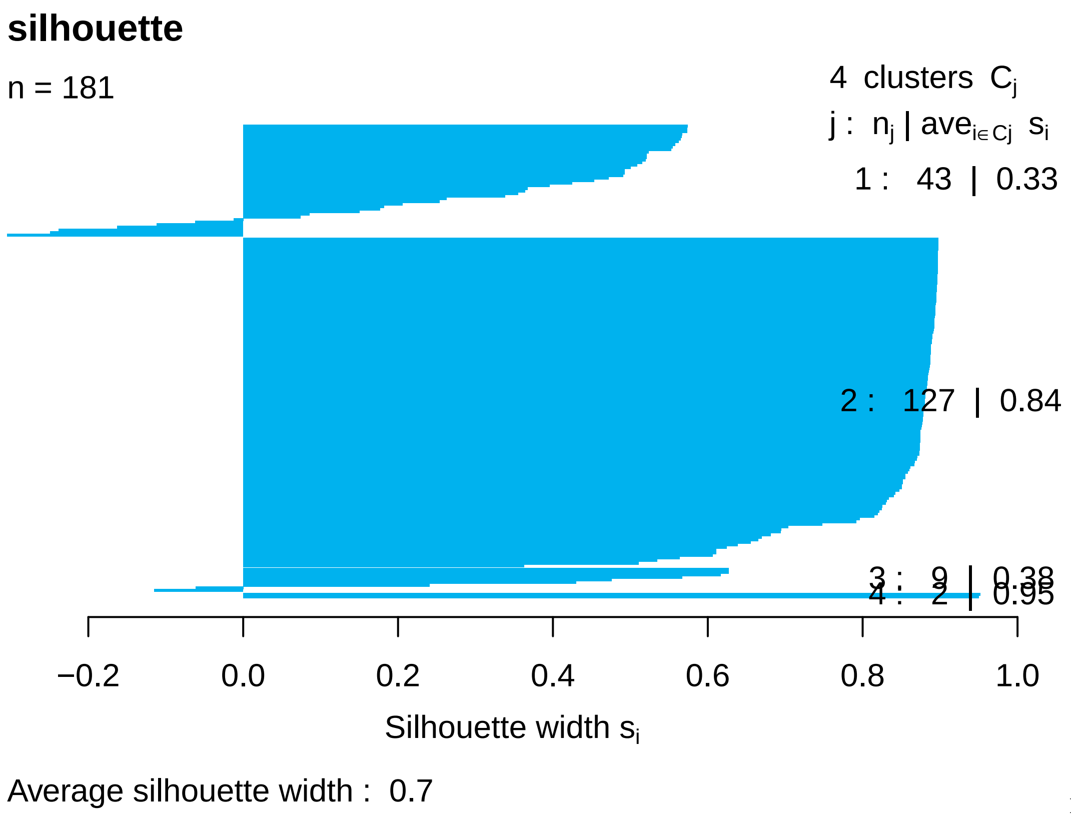
PAM



let's check the mean vector of the results of PAM.

we can also charactsize the clusters as following table: from 1 to 4 means from lowest(fewest) to highest(most).

cluster	cases	death rate	economic	average age	medical resources	stringency
cluster1	1	1	1	1	1	1
cluster2	2	3	2	3	3	4
cluster3	3	4	3	4	4	3
cluster4	4	2	4	2	2	2



Here is the plot of silhouette.

Hierarchical clustering

There are multiple choice in this section, we will only accept the models with reasonable clusters. i.t. not too few or too many observations in one cluster.

Agglomerative algorithms

Single linkage

```
man_dist <- daisy(data_quan,metric="manhattan",stand=FALSE)
single = hclust(man_dist,method="single")
cl_single = cutree(single,4)
table(cl_single)
#> cl_single
#>   1   2   3   4
#> 178   1   1   1
```

Single method is an obvious wrong choice.

Complete linkage

```
complete = hclust(man_dist,method="complete")
cl_complete<- cutree(complete,4)
table(cl_complete)
#> cl_complete
#>   1   2   3   4
#> 170   7   2   2
```

Still terrible, only a little bit better.

Average linkage

```
average<- hclust(man_dist,method="average")
cl_average <- cutree(average,4)
table(cl_average)
#> cl_average
#>   1   2   3   4
#> 162  12   5   2
```

Almost same as the previous one, 165 observations in cluster 1, and 16 in others, not a good result.

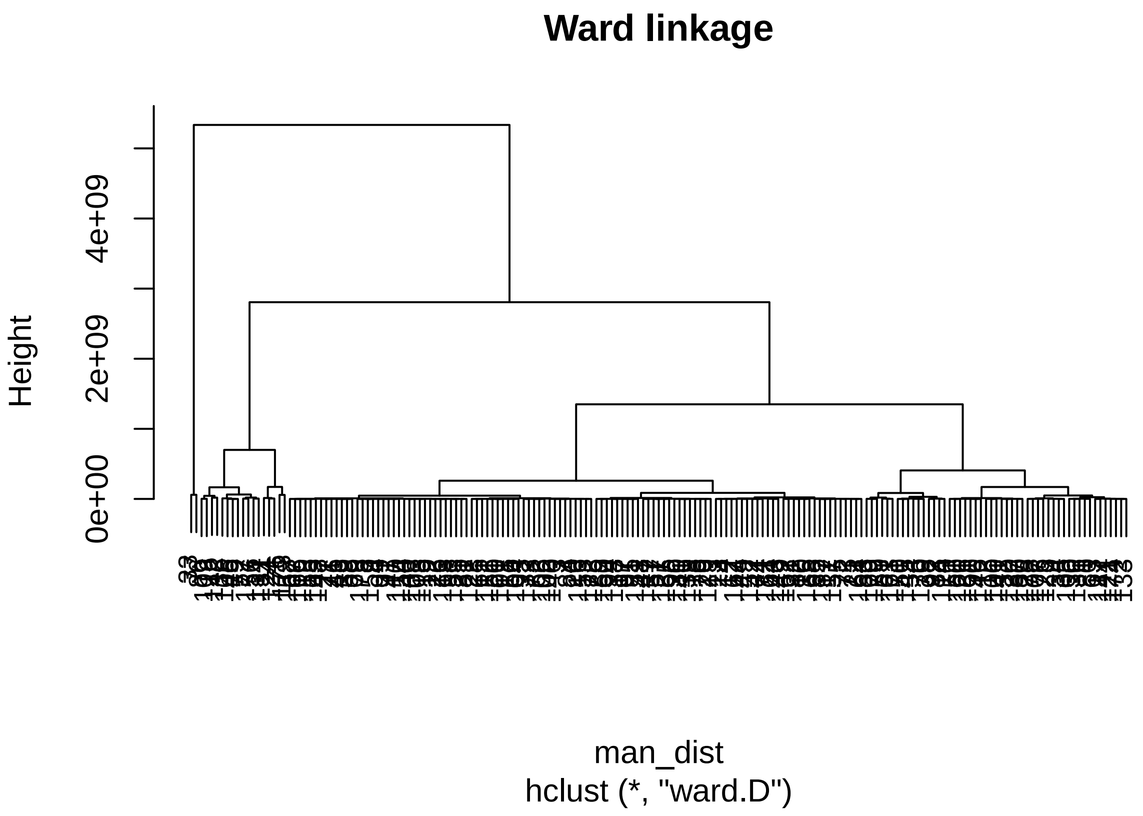
Ward linkage

```
ward <- hclust(man_dist,method="ward")
cl_ward <- cutree(ward,4)
table(cl_ward)
#> cl_ward
#>   1   2   3   4
#>  51 111  17   2
```

This one is acceptable. let's move on and analyze it.

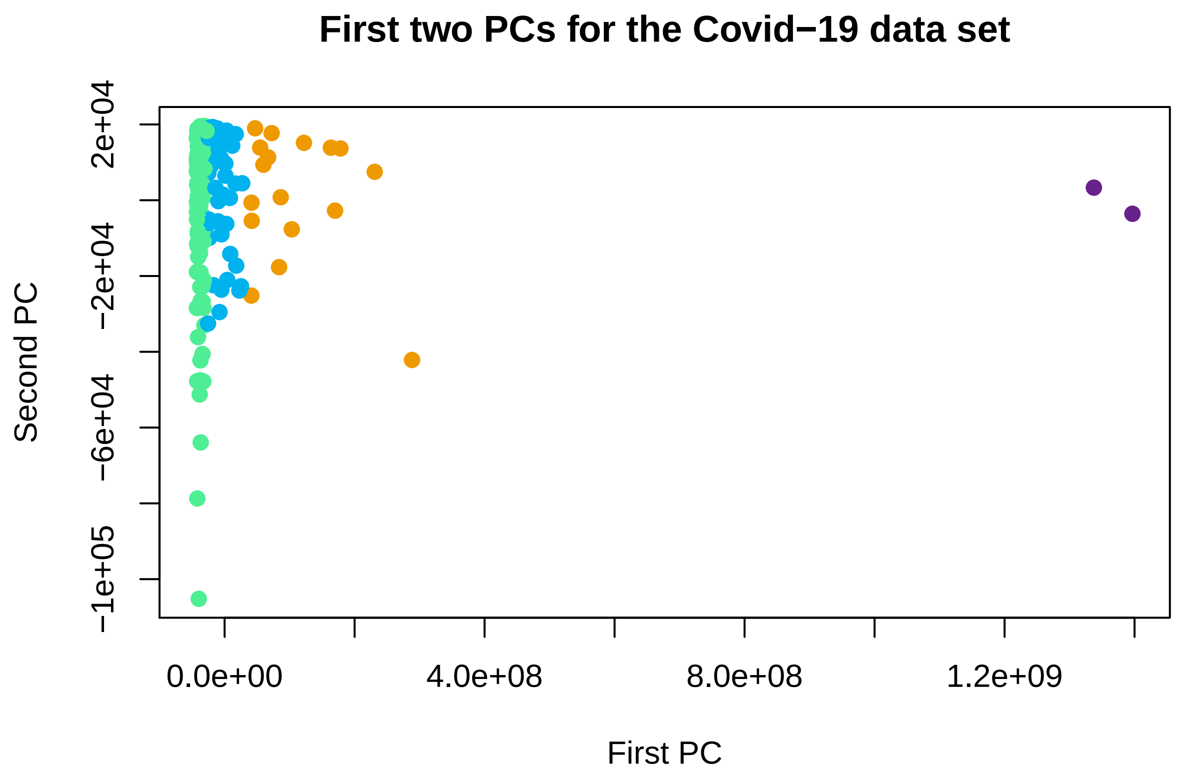
Analysis

```
plot(ward,main="Ward linkage",cex=0.8)
```



Since our assumed K is 4, we need to cut the highest connection, then we can have our clusters.

```
colors_ward <- c(color_1,color_2,color_3,color_4)[cl_ward]
plot(Z,pch=19,col=colors_ward,main="First two PCs for the Covid-19 data set",xlab="First PC",ylab="Second PC")
```

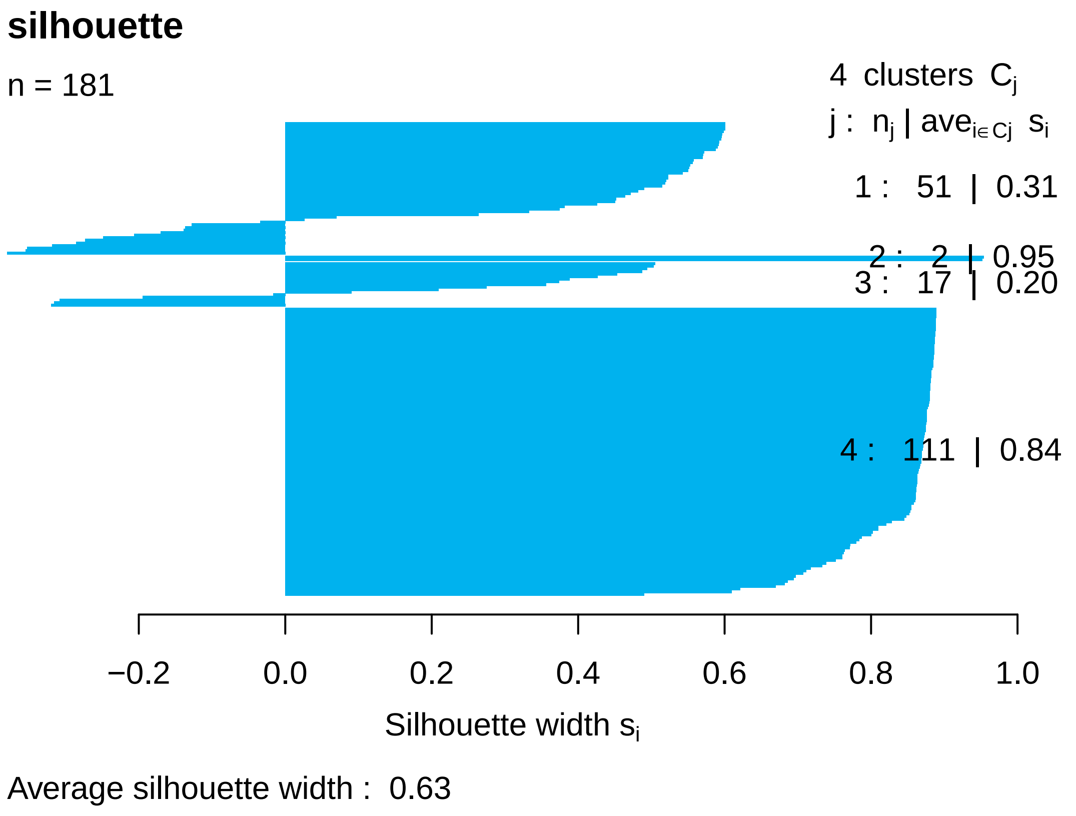


Through this plot, we can consider it's similar to the plot we get in *k means*, but the order of clusters has a different, we can do some adjustments then compare them.

```
which(cl_ward==2)
#>   3   4   5   7   8  10  11  12  13  14  17  18  19  20  21  22  23  25  26  27
#>   3   4   5   7   8  10  11  12  13  14  17  18  19  20  21  22  23  25  26  27
#>  28  29  31  37  39  40  41  42  43  44  46  47  48  49  53  55  57  58  60  62
#>  28  29  31  37  39  40  41  42  43  44  46  47  48  49  53  55  57  58  60  62
#>  64  65  66  67  68  69  71  72  73  74  75  78  81  82  84  85  89  91  93  94
#>  64  65  66  67  68  69  71  72  73  74  75  78  81  82  84  85  89  91  93  94
#>  95  96  97  98  99 101 102 103 104 106 108 110 111 113 115 116 118 119 122 125
#>  95  96  97  98  99 101 102 103 104 106 108 110 111 113 115 116 118 119 122 125
#> 127 129 130 132 135 137 138 139 140 143 147 148 149 150 151 152 153 154 155 156
#> 127 129 130 132 135 137 138 139 140 143 147 148 149 150 151 152 153 154 155 156
#> 157 158 159 162 164 165 166 167 172 175 181
#> 157 158 159 162 164 165 166 167 172 175 181
cl_ward[which(cl_ward==2)]=5
cl_ward[which(cl_ward==4)]=2
cl_ward[which(cl_ward==5)]=4
table(kmeans_2$cluster,cl_ward)
#>    cl_ward
#>       1   2   3   4
#>   1  35   0   0 111
#>   2   0   0   6   0
#>   3   0   2   0   0
#>   4  16   0  11   0
```

The results are almost same. Then we can check the silhouette plot:

```
sil_ward <- silhouette(cl_ward,man_dist)
plot(sil_ward,main='silhouette',col=color_1)
```



We can also charactsize the clusters as following table: from 1 to 4 means from lowest(fewest) to highest(most).

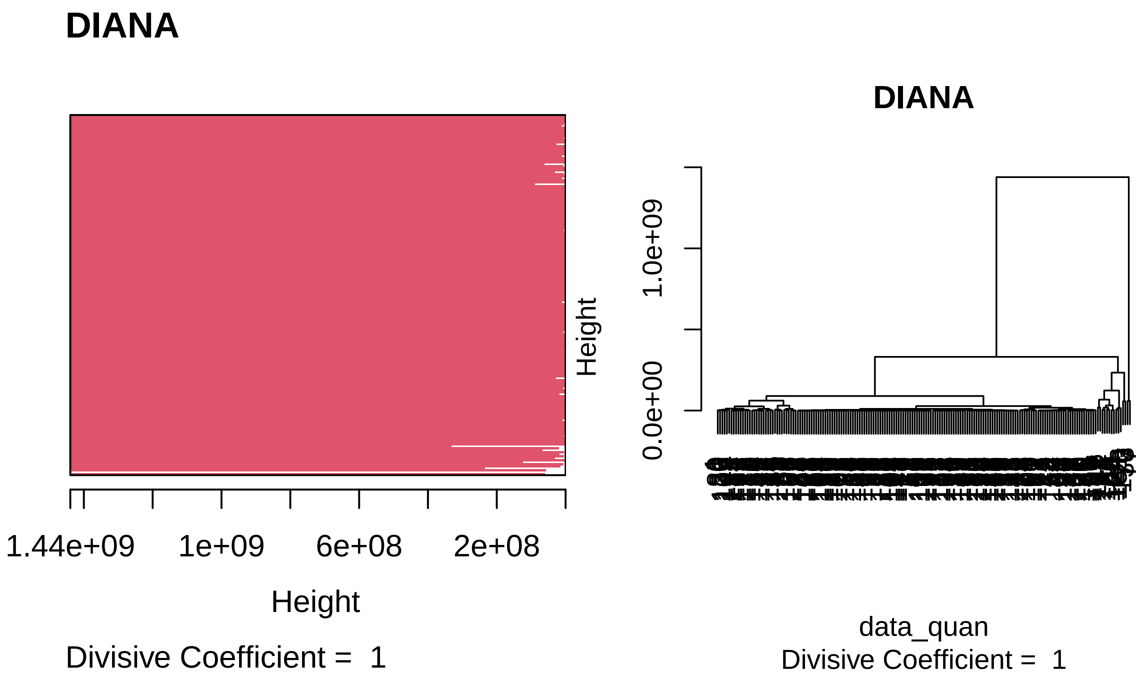
cluster	cases	death rate	economic	average age	medical resources	stringency
cluster1	1	1	1	1	1	1

cluster	cases	death rate	economic	average age	medical resources	stringency
cluster2	2	3	3	4	4	3
cluster3	3	2	4	2	2	2
cluster4	4	4	2	3	3	4

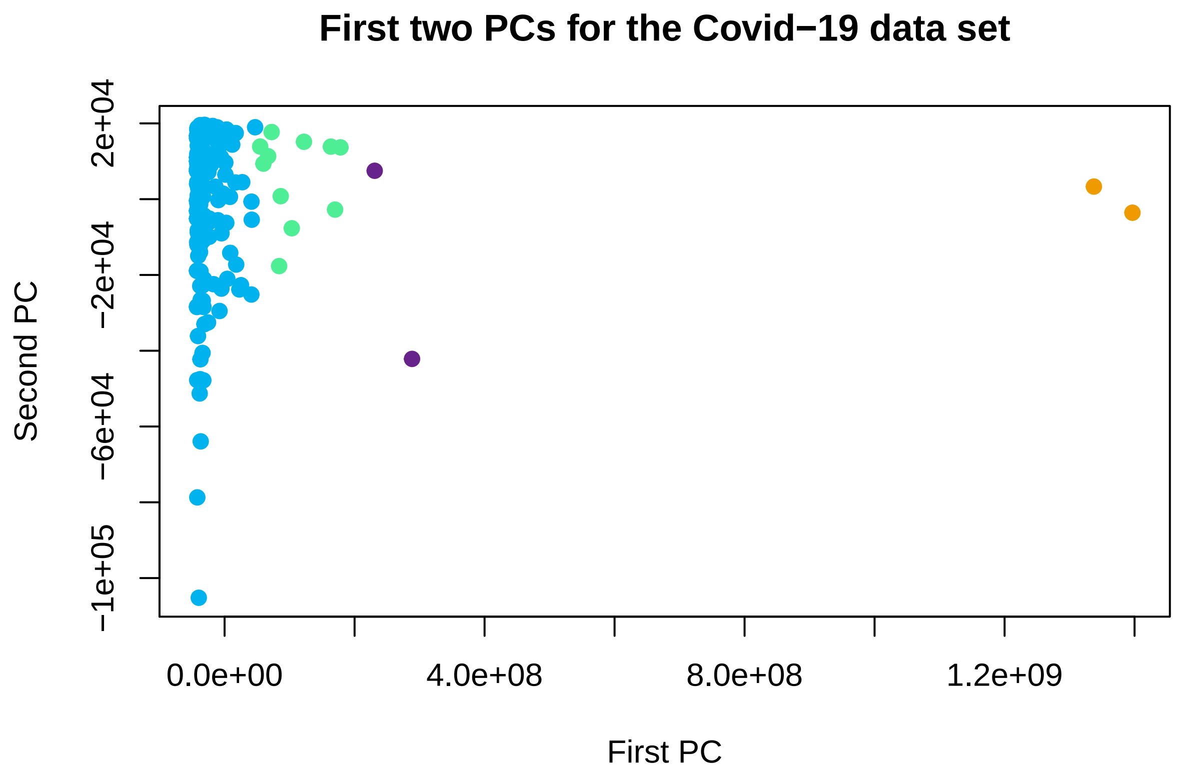
Divisive algorithms

```
diana <- diana(data_quan,metric="manhattan")
cl_diana <- cutree(diana,4)
table(cl_diana)
#> cl_diana
#>   1   2   3   4
#> 166  11   2   2
```

```
plot(diana,main="DIANA")
```



```
colors_diana <- c(color_1,color_2,color_3,color_4)[cl_diana]
plot(Z,pch=19,col=colors_diana,main="First two PCs for the Covid-19 data set",xlab="First PC",ylab="Second PC")
```



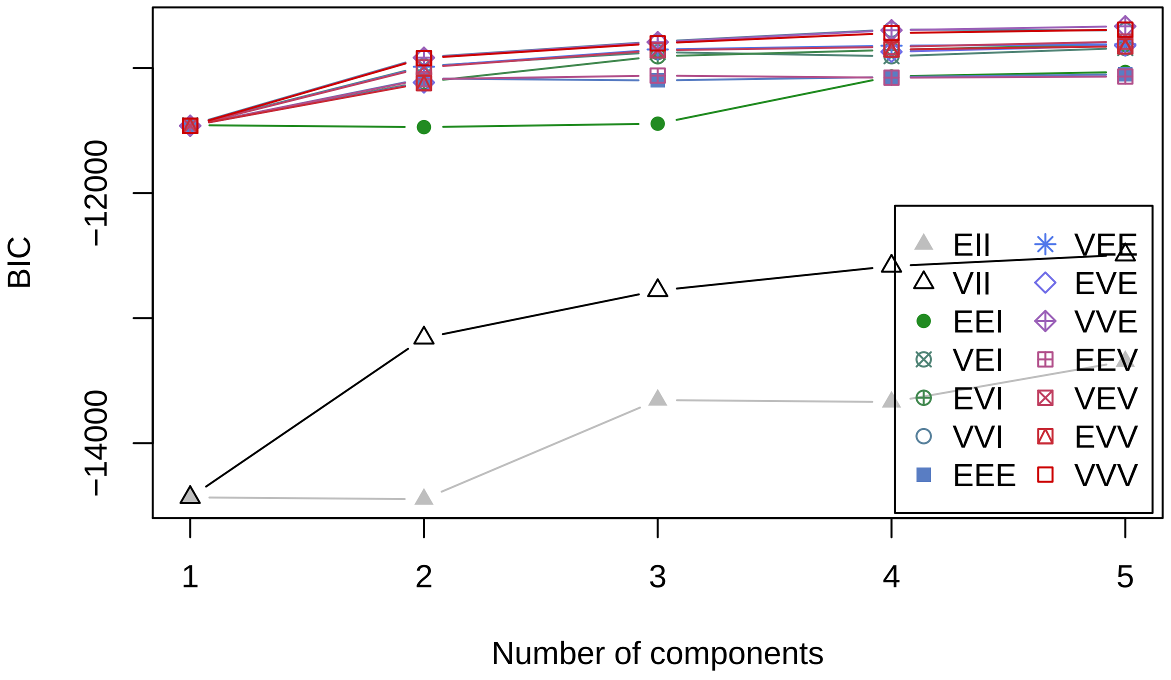
there are too many entries of cluster 1, we can hardly say that it a good one.

so among all **Hierarchical clustering**s we will choose the result of **Ward**.

Model-based clustering

BIC

```
BIC <- mclustBIC(Z,G=1:5)
#> fitting ...
#>   |                                                                          |
```



Model

```
Mclust <- Mclust(Z,x=BIC)
summary(Mclust)
#> ----------------------------------------------------
#> Gaussian finite mixture model fitted by EM algorithm
#> ----------------------------------------------------
#>
#> Mclust VVE (ellipsoidal, equal orientation) model with 5 components:
#>
#>  log-likelihood   n df       BIC      ICL
#>         -5267.9 181 25 -10665.76 -10710.8
#>
#> Clustering table:
#>  1  2  3  4  5
#> 36 46 49  3 47
Mclust$classification
#>   1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18  19  20
#>   1   1   2   3   3   5   2   2   5   3   3   1   3   1   1   5   3   3   2   2
#>  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40
#>   3   2   3   5   2   3   2   2   1   5   3   5   4   1   1   5   1   5   2   2
#>  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60
#>   3   3   2   3   5   2   2   3   3   5   5   5   1   5   2   5   3   2   5   2
#>  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80
#>   5   2   1   1   2   2   2   3   2   5   2   3   3   1   3   5   4   3   5   5
#>  81  82  83  84  85  86  87  88  89  90  91  92  93  94  95  96  97  98  99 100
#>   3   3   5   2   3   5   5   5   3   1   2   5   3   3   3   1   3   2   3   5
#> 101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120
#>   2   2   3   2   5   3   1   2   5   2   2   1   3   5   2   2   1   1   2   1
#> 121 122 123 124 125 126 127 128 129 130 131 132 133 134 135 136 137 138 139 140
#>   5   2   1   5   1   3   3   1   3   3   5   3   5   5   1   5   3   3   3   3
#> 141 142 143 144 145 146 147 148 149 150 151 152 153 154 155 156 157 158 159 160
#>   5   5   1   5   5   1   3   2   1   3   3   1   2   2   3   2   3   2   2   1
#> 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180
#>   1   1   5   1   2   2   3   5   5   1   5   2   4   5   2   5   5   1   5   1
#> 181
#>   1
```

Parameters

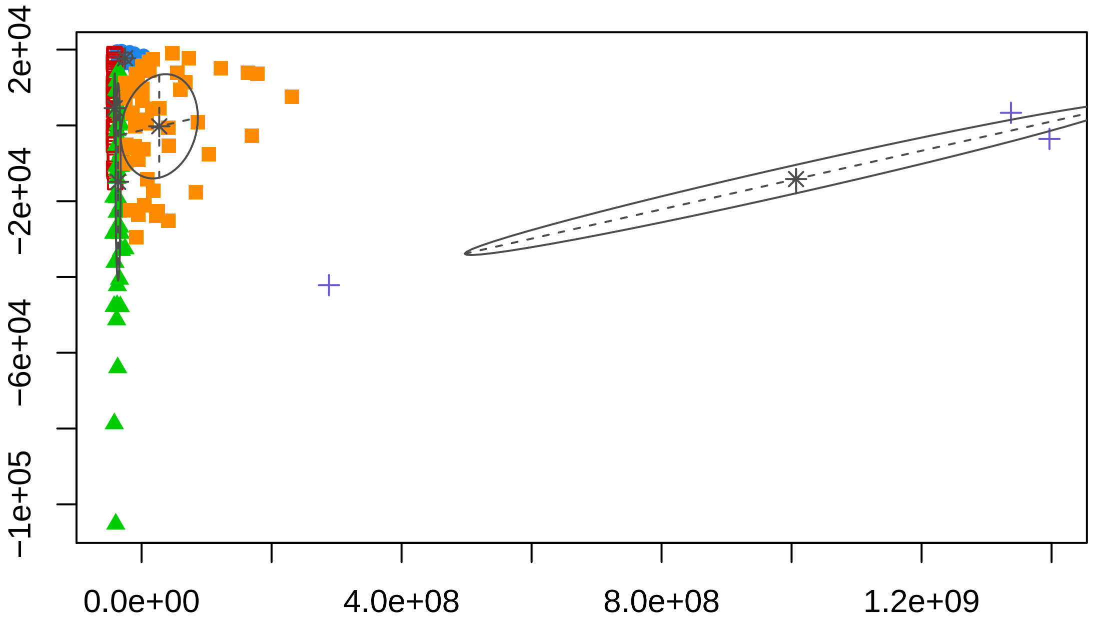
Here is the parameters' probability and mean vector

```
Mclust$parameters$pro
#> [1] 0.19896536 0.22020697 0.28388838 0.01658465 0.28035465
```

```
Mclust$parameters$mean
#>               [,1]          [,2]         [,3]          [,4]          [,5]
#> [1,] -25285823.36 -41187533.664 -36080501.91 1006910620.68 27266760.2101
#> [2,]      17724.62      4555.783     -14897.01     -14156.83     -235.1567
```

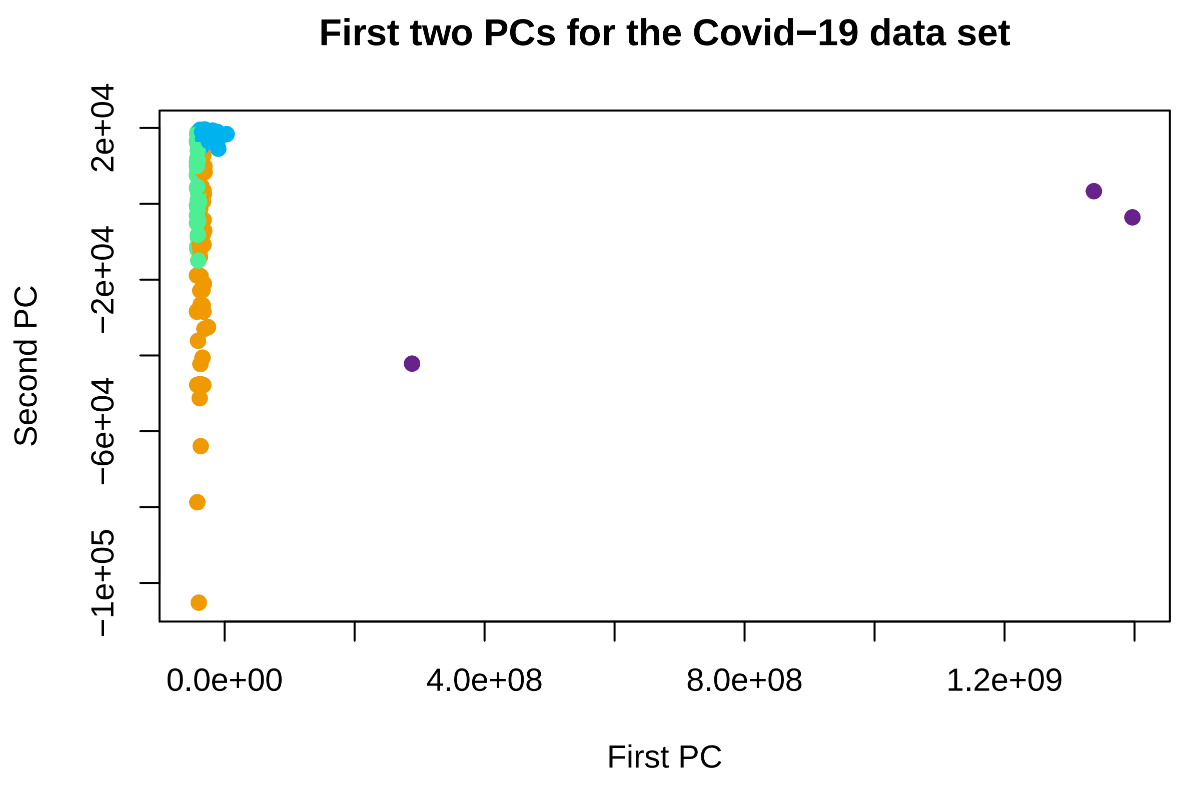
Mclust plot

```
plot(Mclust,what="classification")
```

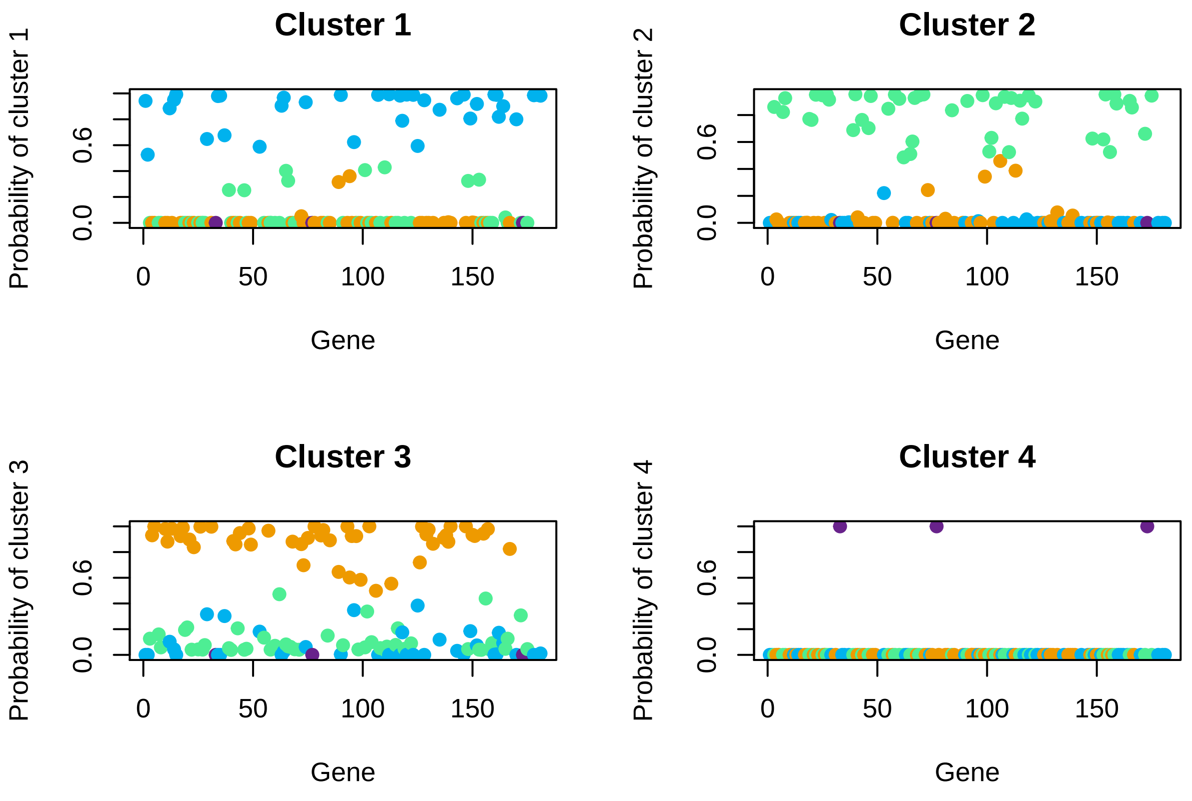


PCA plot

```
colors_Mclust <- c(color_1,color_2,color_3,color_4)[Mclust$classification]
plot(Z,pch=19,col=colors_Mclust,main="First two PCs for the Covid-19 data set",xlab="First PC",ylab="Second PC")
```

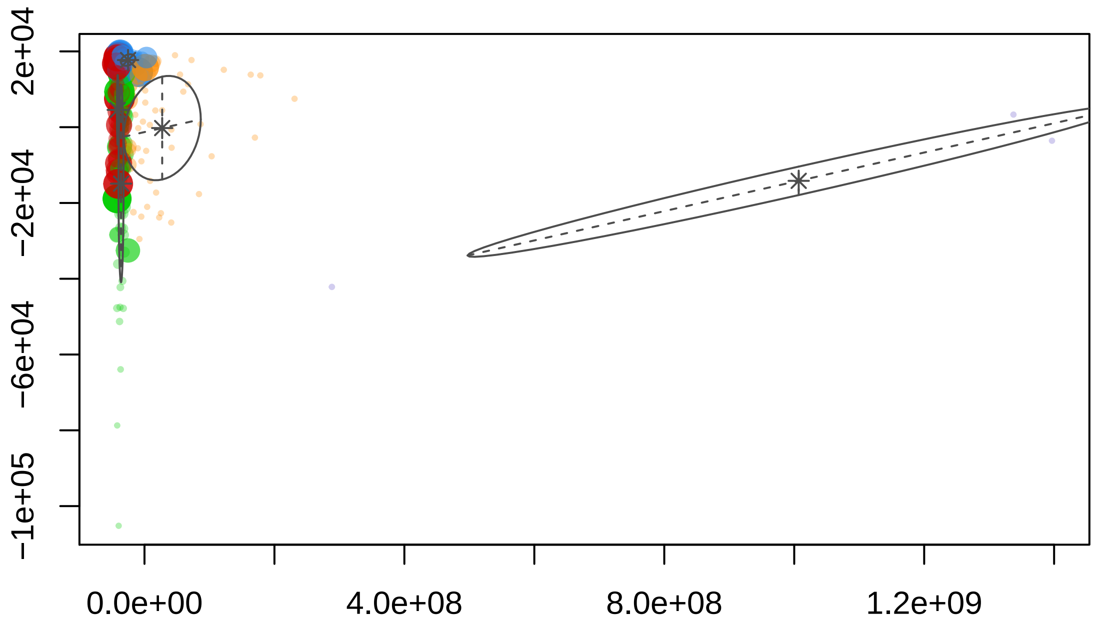


Probability plot



These four plots show the probability of the observations locate in the specific cluster. We can see that each cluster has a fairly good performance. it is reliable.

```r
par(mfrow=c(1,1))
plot(Mclust,what="uncertainty")
```



And here we can check the plot of those observations labeled with **uncertainty**

Analysis

We can also characterize the clusters as following table:

From 1 to 4 means from lowest(fewest) to highest(most).

cluster	cases	death rate	economic	average age	medical resources	stringency
cluster1	2	2	1	1	1	1
cluster2	3	3	2	3	3	4
cluster3	4	4	3	4	4	3
cluster4	1	1	4	2	2	2

Analysis of the results

We set the K into 4, i.t. we wish the algorithm can split the dataset into 4 clusters with clear border with the others. And there shouldn't be too many or too few observations in one cluster.

Hence we present the result from *K-Means*,*PAM*,*Agglomerative algorithms with ward linkage*,*Model-based*. And here we can put all mean vectors together.

1. K-Means:

We can check the cluster number and which countries are in the same cluster, but the table would be to long to show it.

from 1 to 4 means from lowest(fewest) to highest(most).

cluster	cases	death rate	economic	average age	medical resources	stringency
cluster1	1	1	1	1	1	1
cluster2	2	3	3	4	4	3
cluster3	3	2	4	2	2	2
cluster4	4	4	2	3	3	4

2. PAM:

cluster	cases	death rate	economic	average age	medical resources	stringency
cluster1	1	1	1	1	1	1
cluster2	2	3	2	3	3	4
cluster3	3	4	3	4	4	3
cluster4	4	2	4	2	2	2

3.

cluster	cases	death rate	economic	average age	medical resources	stringency
cluster1	1	1	1	1	1	1
cluster2	2	3	3	4	4	3
cluster3	3	2	4	2	2	2
cluster4	4	4	2	3	3	4

4. Model_based

cluster	cases	death rate	economic	average age	medical resources	stringency
cluster1	2	2	1	1	1	1
cluster2	3	3	2	3	3	4
cluster3	4	4	3	4	4	3
cluster4	1	1	4	2	2	2

Factor Analysis

Multidimensional Scaling

Dataset: Similarity of cocktails' popularity

The dataset contains how similar cocktails are in terms of popularity, the higher the similarity (between 1 and 0) the most similarly popular 2 drinks are.

A similarity of 1 = the cocktails are equally popular, similarity closer to 0 = one of the cocktails is significantly more popular than the other.

Correspondence analysis

Given the following contigency table of each pair of classes corresponding to each variable (age group and health status), we will perform correspondence analysis:

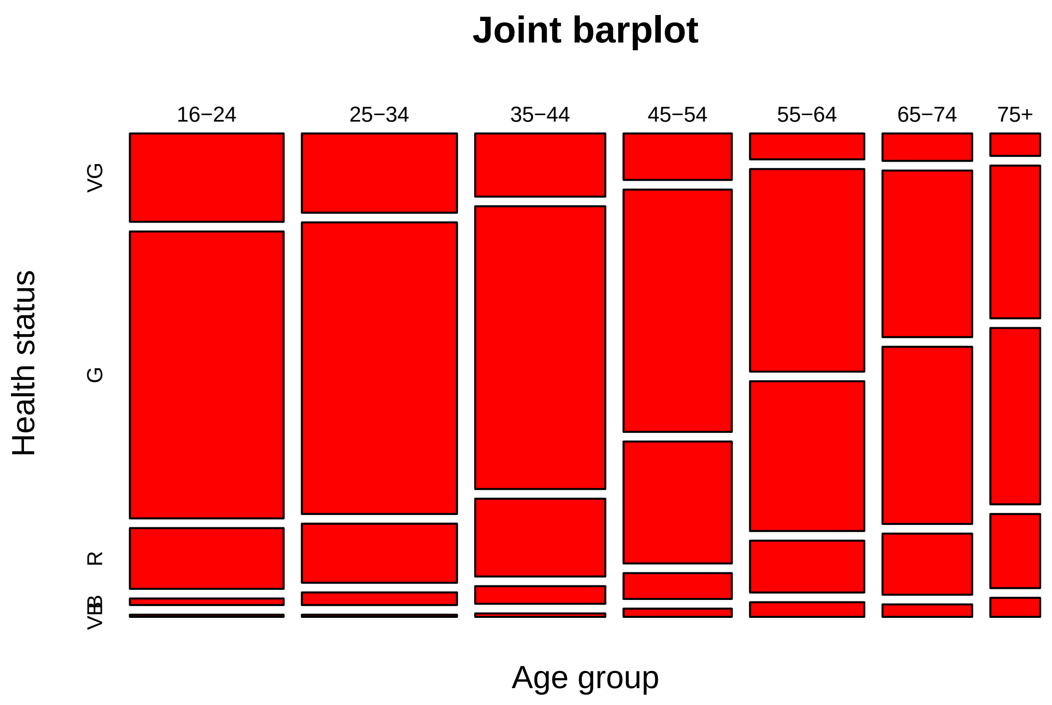
Table 9: health table

	VG	G	R	B	VB	Sum
16-24	243	789	167	18	6	1223
25-34	220	809	164	35	6	1234
35-44	147	658	181	41	8	1035
45-54	90	469	236	50	16	861
55-64	53	414	306	106	30	909
65-74	44	267	284	98	20	713
75+	20	136	157	66	17	396
Sum	817	3542	1495	414	103	6371

Visual analysis of the data

We can see a graphical representation of the contingency table as follows:

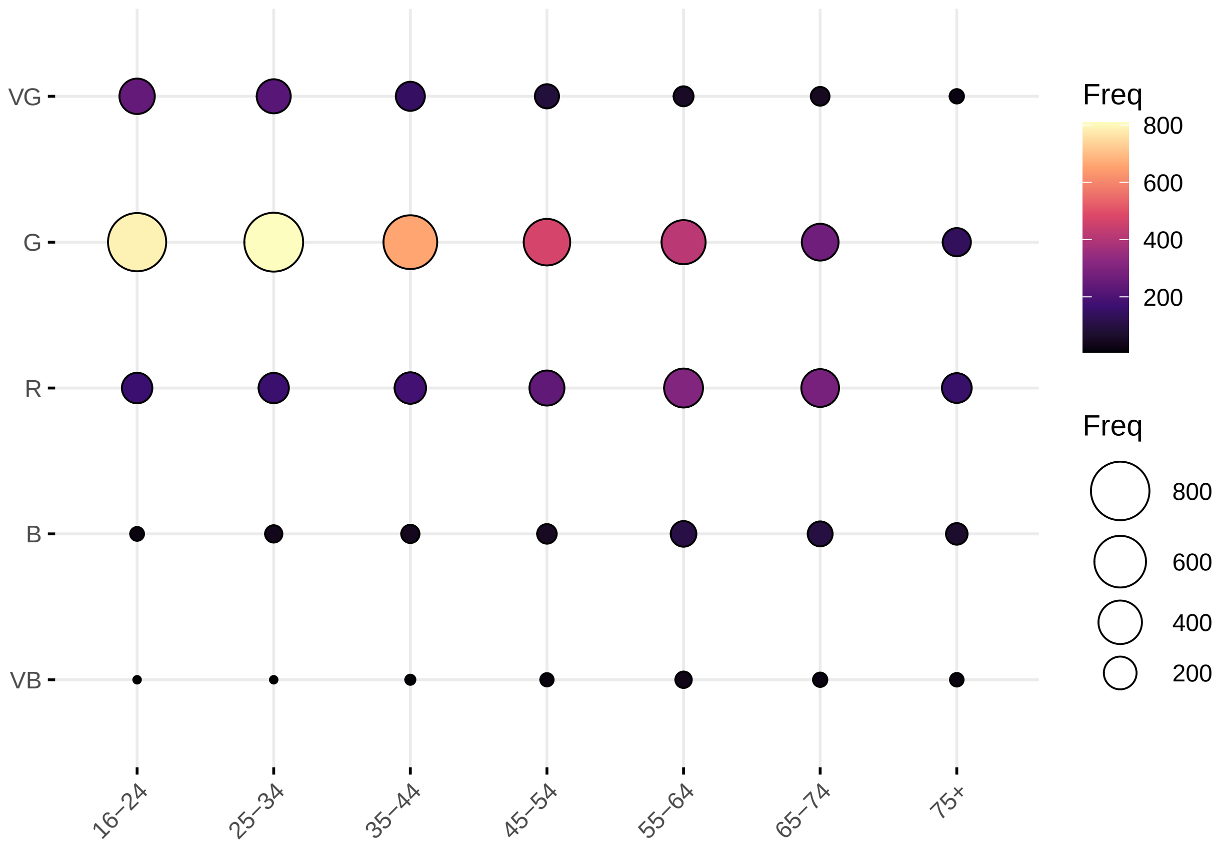
```r
plot(health,xlab="Age group",ylab="Health status",col='red',main="Joint barplot")
```



In this joint barplot we can notice that the age groups are all more or less the same, with a small trend, where younger age groups tend to have a larger amount of individuals than older age groups.

We also notice that people with good health status are more abundant than the rest.

```r
ggballoonplot(as.data.frame(health),fill="value")+scale_fill_viridis_c(option="A")
```



Our balloon plot tells us much of the same, the larger age groups are hte younger ones and, in proportion, there's a significantly larger amount of people with good/very good health status in younger age groups, than those with worse status within the same age group. The older the age group gets, the lesser the amount of individuals with good/very good health status, and the more with regular/bad health status.

Very bad health status individuals, while a very small subset of the general sample, are increasingly more common the older the age group is.

Therefore, there are differences in the sizes of groups, in general. But the differences are not too dramatic for age group sizes, the differences are more significant for health status groups. And there's definitely some relationship between the variables, or so we can infer from the plots.

Testing for independency between the variables

Relative proportion table (observed):

Table 10: health table

	VG	G	R	B	VB	Sum
16-24	0.0381416	0.1238424	0.0262125	0.0028253	0.0009418	0.1919636
25-34	0.0345315	0.1269816	0.0257416	0.0054936	0.0009418	0.1936902
35-44	0.0230733	0.1032805	0.0284100	0.0064354	0.0012557	0.1624549
45-54	0.0141265	0.0736148	0.0370429	0.0078481	0.0025114	0.1351436
55-64	0.0083189	0.0649819	0.0480301	0.0166379	0.0047088	0.1426778
65-74	0.0069063	0.0419086	0.0445770	0.0153822	0.0031392	0.1119134
75+	0.0031392	0.0213467	0.0246429	0.0103594	0.0026683	0.0621566
Sum	0.1282373	0.5559567	0.2346570	0.0649819	0.0161670	1.0000000

Here we can see how different groups are, the distribution of age groups is more even, however, for health status groups, "good" and "regular" gobble up over 70% of the observations.

Chi squared test (observed vs expected):

```
chisq.test(health)
#>
#>  Pearson's Chi-squared test
#>
#> data:  health
#> X-squared = 894.86, df = 24, p-value < 2.2e-16
```

We get a p-value of <2e-16, which means that there's a significant dependence between the age group and health status variables.

Correspondence analysis for the data matrix

First of all we calculate the total relative frequencies for rows/cols:

```
rel_freq_rows <- rowSums(health_rf)
rel_freq_cols <- colSums(health_rf)
```

We create a matrix of zeros where the diagonal is the sum of the rows of our relative frequency matrix and we do the same for the columns.

```
diag_rs <- diag(rel_freq_rows)
diag_cs <- diag(rel_freq_cols)
```

We the compute the matrices of row and column profiles:

```
prof_rs <- solve(diag_rs) %*% health_rf
apply(prof_rs, 1, sum)
#> [1] 1 1 1 1 1 1 1
prof_cs <- solve(diag_cs) %*% t(health_rf)
apply(prof_cs, 1, sum)
#> [1] 1 1 1 1 1
```

We compute the matrix M and its SVD:

```
M <- diag(1/sqrt(rel_freq_rows)) %*% (health_rf - rel_freq_rows %*% t(rel_freq_cols)) %*% diag(1/sqrt(rel_freq_cols))
M_svd <- svd(M)
```

We then define the Lambda, Gamma and Theta matrices:

```
Lambda_M <- diag(M_svd$d)
Gamma_M <- M_svd$u
Theta_M <- M_svd$v
```

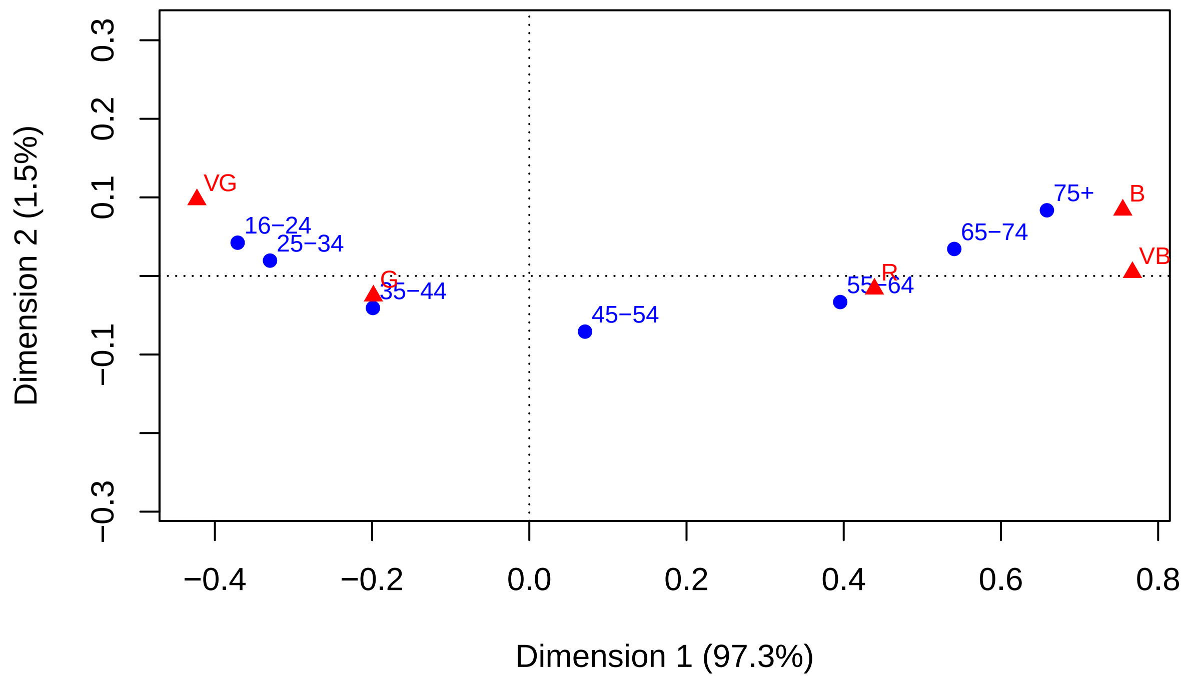
And we obtain each matrix:

```
X_r <- diag(1/sqrt(rel_freq_rows)) %*% Gamma_M[,1:2] %*% Lambda_M[1:2,1:2]
X_r
#>             [,1]        [,2]
#> [1,] -0.37107411  0.04230757
#> [2,] -0.32988430  0.01951487
#> [3,] -0.19895401 -0.04075134
#> [4,]  0.07091332 -0.07085805
#> [5,]  0.39551813 -0.03324647
#> [6,]  0.54063511  0.03434953
#> [7,]  0.65849263  0.08356749
X_c <- diag(1/sqrt(rel_freq_cols)) %*% Theta_M[,1:2] %*% Lambda_M[1:2,1:2]
X_c
#>            [,1]         [,2]
#> [1,] -0.4228656  0.097118969
#> [2,] -0.1983459 -0.025360769
#> [3,]  0.4390676 -0.016546296
#> [4,]  0.7550362  0.083935611
#> [5,]  0.7672942  0.004553535
```

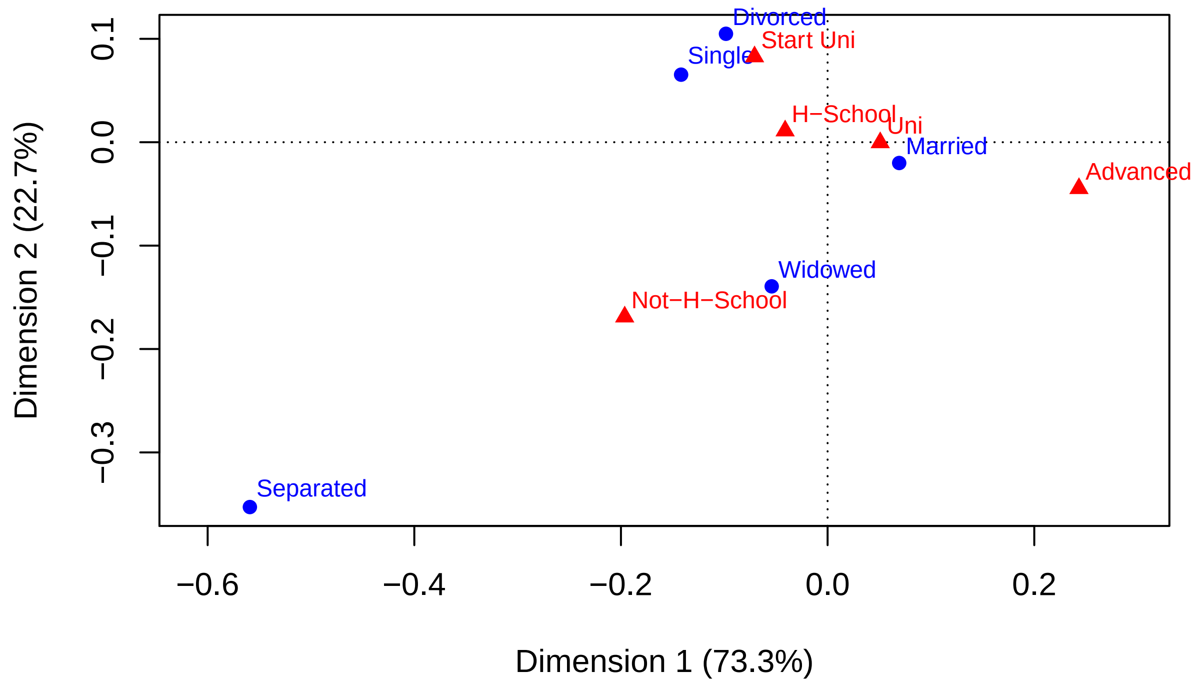
Library 'ca' and conclusions

Utilizing the library 'ca' we can perform the same analysis in a more speedy manner:

```
ca_ages_status <- ca(health)
plot(ca_ages_status)
```



```
Studies_Marital_Status <- prop.table(table(read.csv('../../datasets/Studies_Marital_Status.csv')))
plot(ca(Studies_Marital_Status))
```



From this we can see a few things:

- Clearly, most of these classes are dependent on each other
- Very good health status is strongly dependent on the respondant being younger (16-24 and 25-34)
- Good health status is also strongly dependent on people being relatively younger, but perhaps more than anything it's closer to group 35-44. Either way though, we can't underestimate good/very good's health status' dependence on youth overall.
- Bad and very bad health status are often strongly dependent on older ages, especially 75+
- Regular health status has a significant dependence on the respondants being 55-64 years of age. It seems like a decent way for this group to differentiate itself from the rest, where we can especulate that the respondants are not confident

on their health status enough to say that they're in good or bad condition. We can also infer that many long-term health conditions that are mildly deteriorating are already somewhat developed by this age, conditions like vision issues (i.e. developed myopia), arthritis, osteoporosis and some heart conditions are either starting to be developed around this age or are already developed to significantly developed, therefore maybe skewing the individuals' perspective of their own health status.

- Age group 45-54 seems to be in a midpoint where no particular health status is dependent on it in any significant way, these people may or may not consider themselves in good health, but overall, it's a bit of a tossup between people with regular health status and good health status among individuals in this group.