Exploratory Data Analysis and Clustering: Globalization dataset

Thierry Denoeux

8/8/2022

Reading the data

We start by reading the data

```
data<-read.table('/Users/Thierry/Documents/R/Data/Economics/globalization/global2000_2014.txt',
                 header=TRUE)
dim(data)
```

```
## [1] 159   8
```

```
head(data)
```

```
##                 country code eco00 soc00 pol00 eco14 soc14 pol14
## 3           Afghanistan  AFG 28.44  7.84 36.50 35.45 18.64 46.07
## 4                Angola  AGO 74.55 13.00 47.98 52.54 21.86 48.27
## 5               Albania  ALB 29.02 36.02 59.50 66.99 46.24 71.80
## 7 United Arab Emirates  ARE 75.74 81.58 45.65 88.06 77.93 55.33
## 8             Argentina  ARG 61.56 52.21 91.43 38.26 52.60 92.61
## 9               Armenia  ARM 59.08 44.14 29.75 68.21 43.88 66.99
```

```
x<-data[,6:8]
row.names(x)<-data[,2]
```

Printing basic summary statistics:

```
summary(x)
```

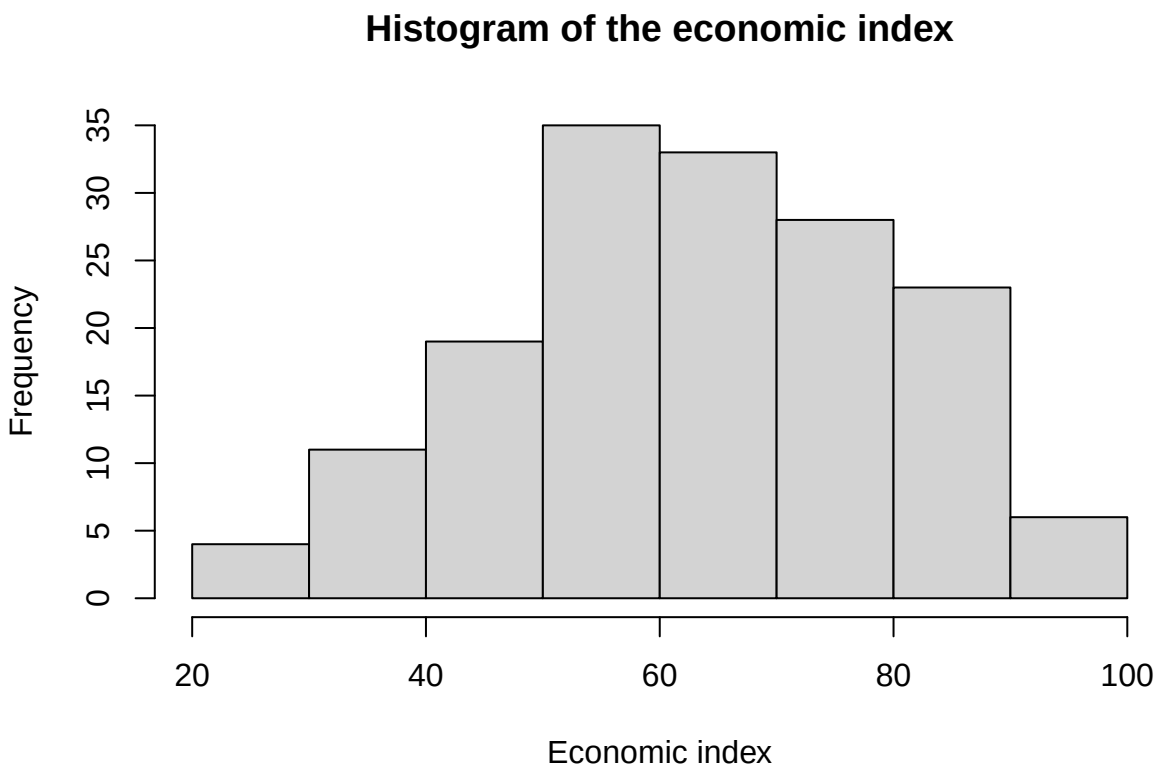
```
##      eco14           soc14           pol14
##  Min.   :24.72   Min.   :15.83   Min.   :21.11
##  1st Qu.:52.55   1st Qu.:29.53   1st Qu.:60.38
##  Median :62.83   Median :48.49   Median :72.93
##  Mean   :63.41   Mean   :50.86   Mean   :71.76
##  3rd Qu.:76.33   3rd Qu.:71.16   3rd Qu.:85.95
##  Max.   :97.77   Max.   :91.61   Max.   :97.29
```

Basic Plots

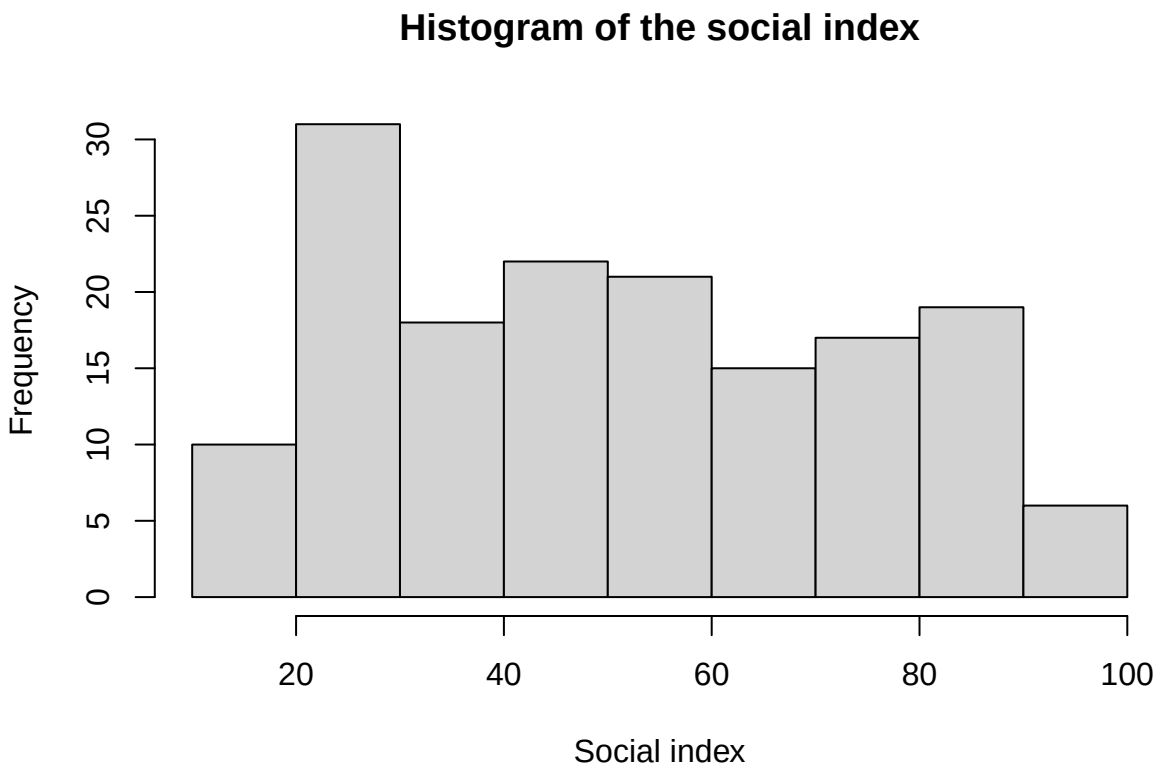
1D plots

Histograms:

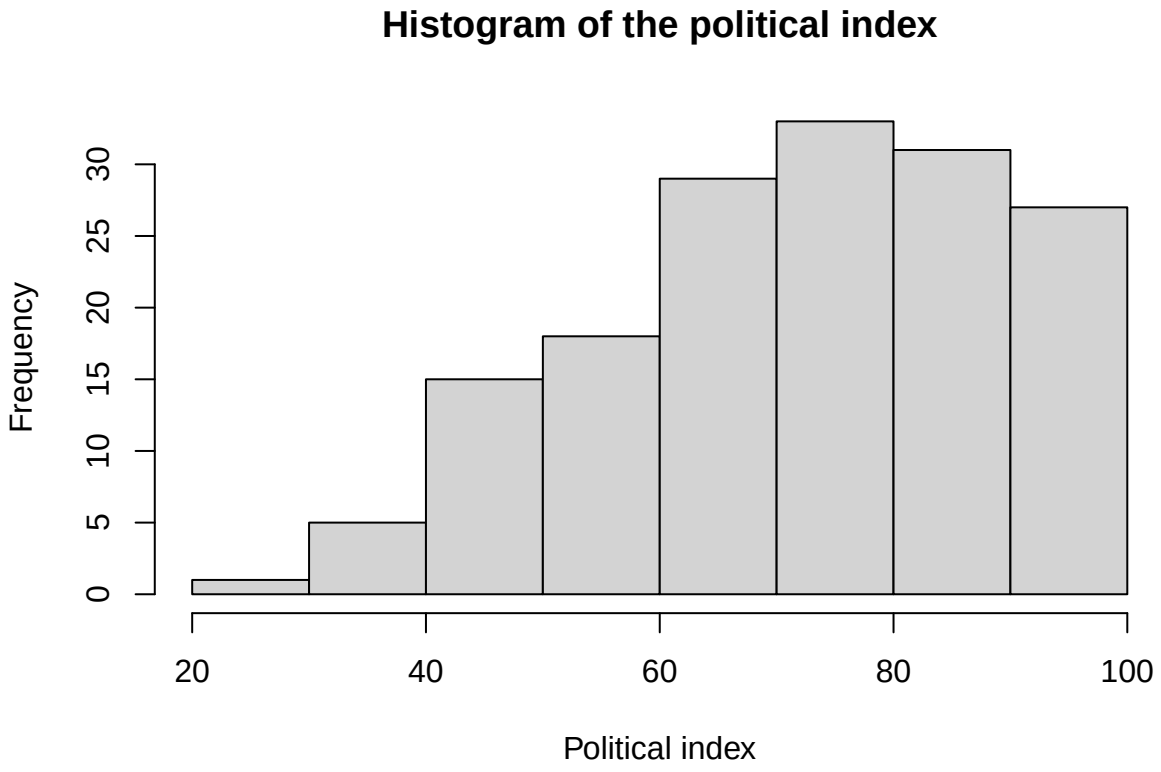
```
hist(x[,1],xlab='Economic index',main="Histogram of the economic index")
```




```
hist(x[,2],xlab='Social index',main="Histogram of the social index")
```




```
hist(x[,3],xlab='Political index',main="Histogram of the political index")
```



Stem and leaf plots:

```
stem(x[,1])
```

```
##
##   The decimal point is 1 digit(s) to the right of the |
##
##   2 |
##   2 | 5778
##   3 | 124
##   3 | 58999
##   4 | 000013344
##   4 | 5556777888899
##   5 | 0122233333344
##   5 | 55677777888889999999
##   6 | 00001112233333334444
##   6 | 55667788888999
##   7 | 00112223333
##   7 | 555666667788888999
##   8 | 011112333334
##   8 | 56677778889
##   9 | 0234
##   9 | 58
```

```
stem(x[,2])
```

```
##
##   The decimal point is 1 digit(s) to the right of the |
##
##   1 | 6888899999
##   2 | 111112223333444
```

```
##   2 | 555566677788899
##   3 | 000134
##   3 | 55556667799
##   4 | 00000012233333444
##   4 | 5667789
##   5 | 1112223344
##   5 | 55556677788
##   6 | 111234
##   6 | 55667789
##   7 | 001122444
##   7 | 67778999
##   8 | 00111233344
##   8 | 555667789
##   9 | 011112
```

```
stem(x[,3])
```

```
##
##   The decimal point is 1 digit(s) to the right of the |
##
##   2 | 1
##   2 |
##   3 |
##   3 | 78999
##   4 | 134
##   4 | 555668888999
##   5 | 124444
##   5 | 55555889999
##   6 | 0011222233444
##   6 | 5555566666778999
##   7 | 00112222223333334
##   7 | 5566666667778899
##   8 | 0011223333444444
##   8 | 5666677788999
##   9 | 0000001111112222233334
##   9 | 55556677
```

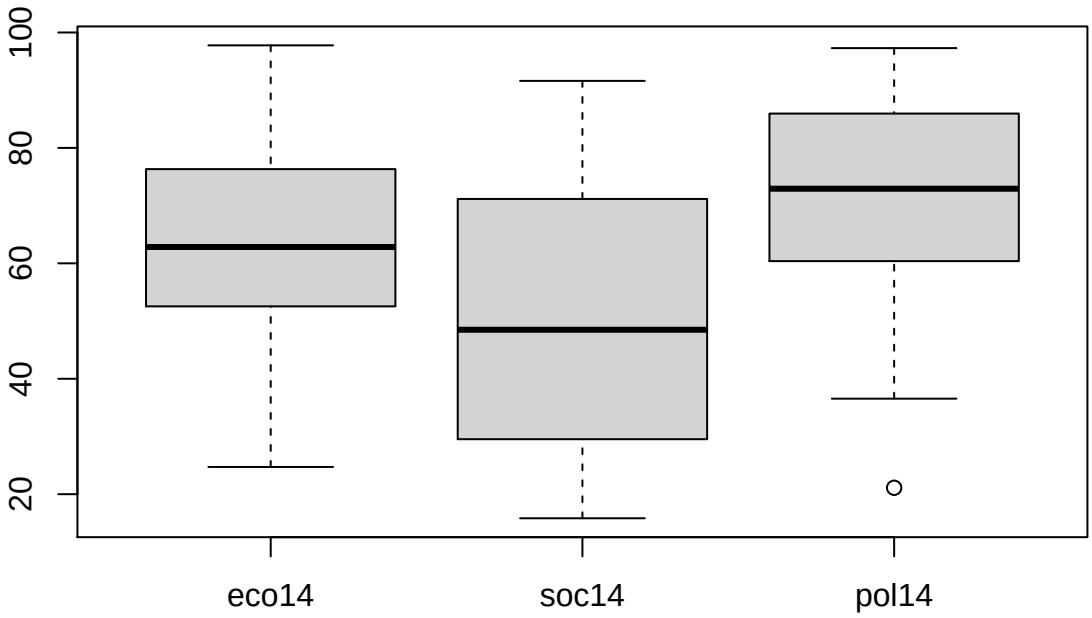
We clearly see one outlying observation for the political index:

```
ii<-which(x[,3]<22)
print(data[ii,1])
```

```
## [1] "Kiribati"
```

Boxplots:

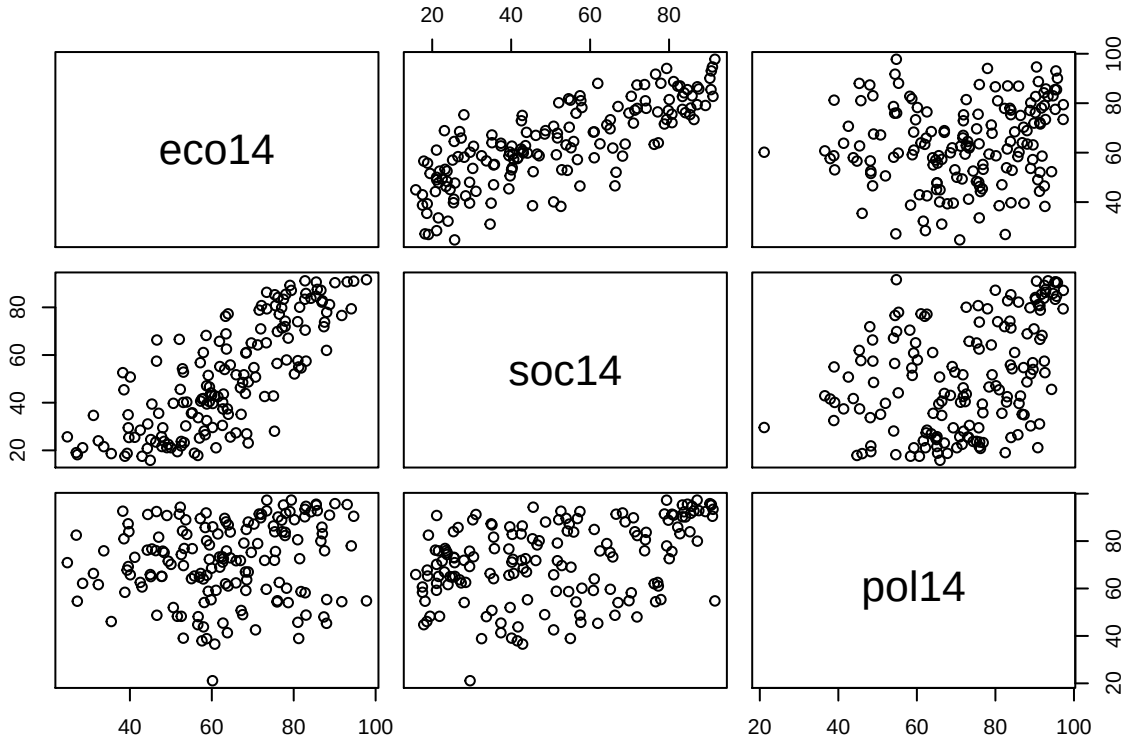
```
boxplot(x)
```



2D plots

Scatter plots:

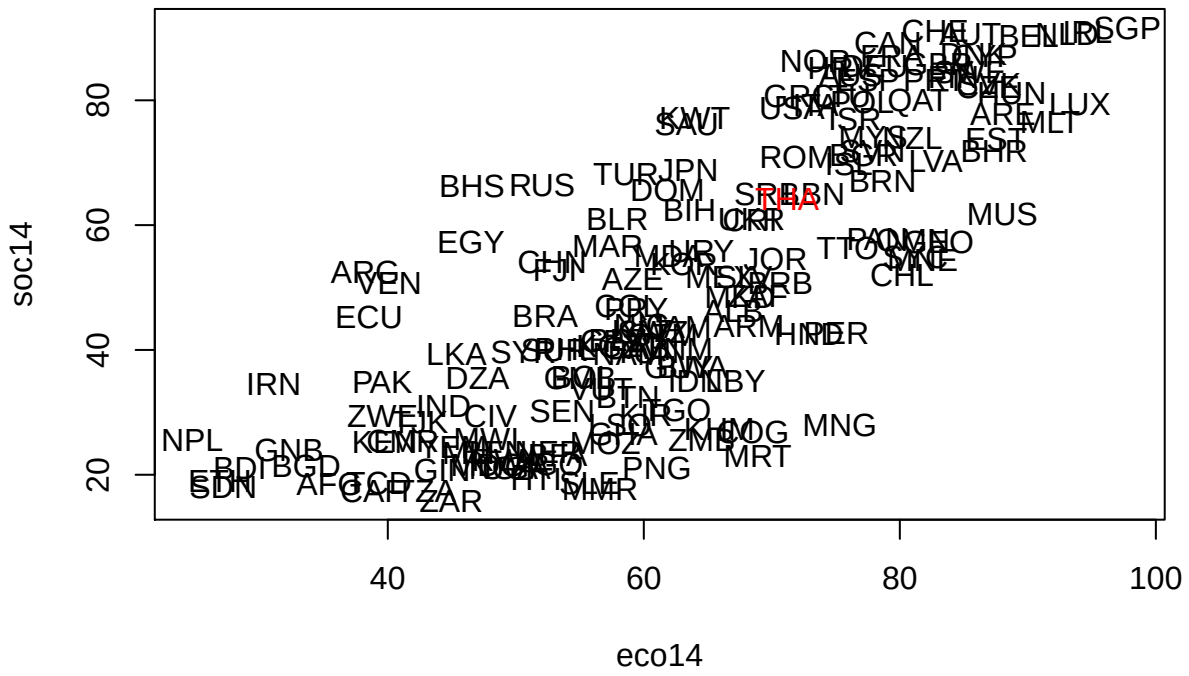
```
plot(x)
```



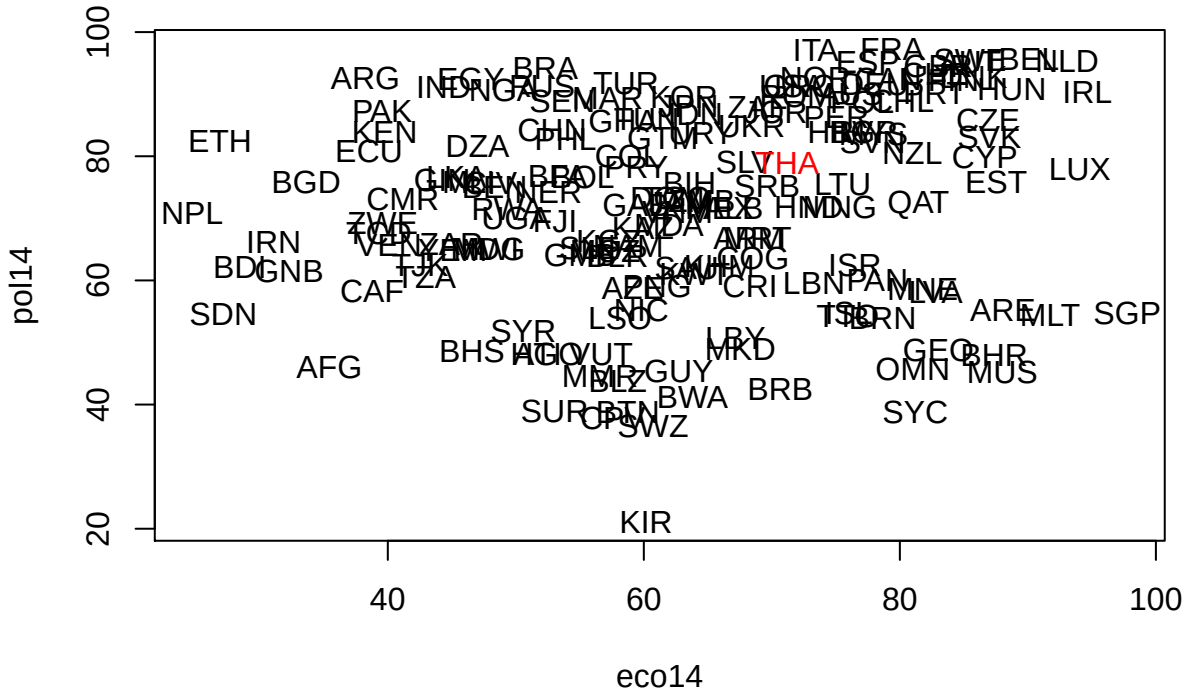
Scatter plots with country codes

```
ii<-which(row.names(x)=="THA")
plot(x[,1:2],type="n")
```

```
text(x[-ii,1:2],row.names(x[-ii,]))
text(x[ii,1:2],"THA",col="red")
```




```
plot(x[,c(1,3)],type="n")
text(x[-ii,c(1,3)],row.names(x[-ii,]))
text(x[ii,c(1,3)],"THA",col="red")
```



We clearly see a correlation between the economic and social indices:

```
cor(x)
```

```
##           eco14     soc14     pol14
```

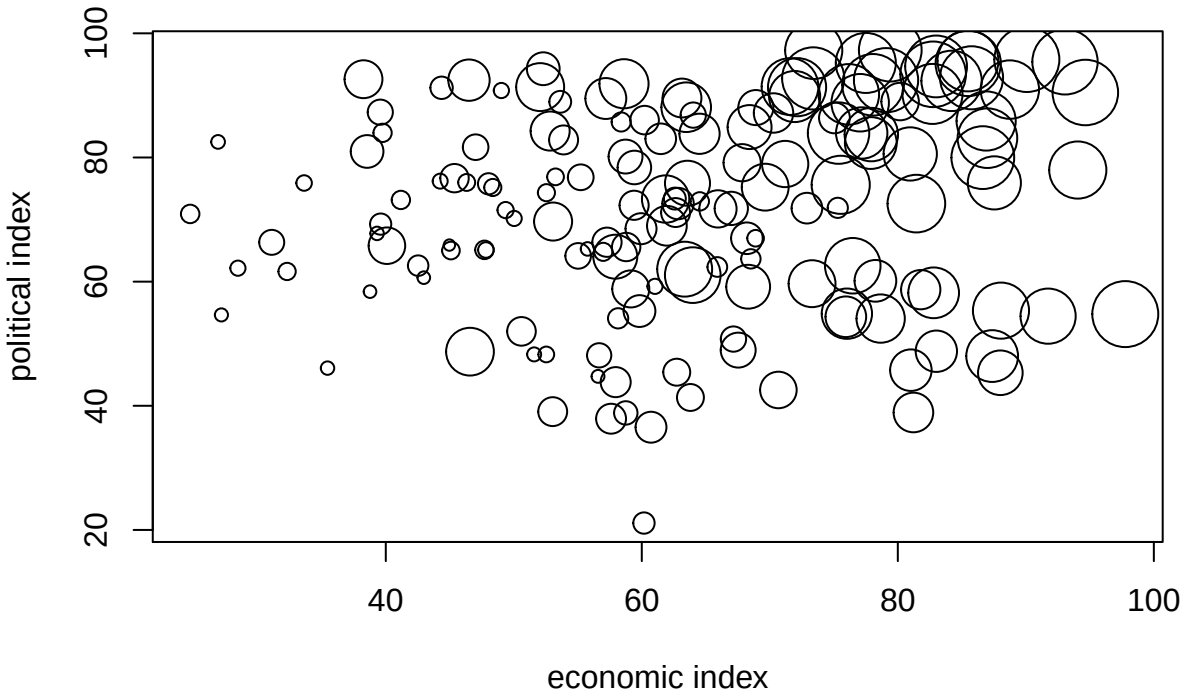
```
## eco14 1.0000000 0.7855921 0.1590361
## soc14 0.7855921 1.0000000 0.3967397
## pol14 0.1590361 0.3967397 1.0000000
```

3D plots

Plots in three dimensions are usually not very readable. Package `rgl` has function `plot3d` that draws interactive 3D plots that can be rotated.

Another solution is to represent the third dimension as the size of the symbols:

```
plot(x[,1],x[,3],pch=1,cex=x[,2]/20,xlab='economic index',ylab='political index')
```



Other visualization methods

Other visualization methods for arbitrary multidimensional data have been proposed. For instance, Chernoff faces display multivariate data as human faces. The motivation is that we easily notice small changes in human faces and assess the similarity between them.

Chernoff faces are available in package `DescTools`. We will first select a subset of countries to display:

```
ii<-which(row.names(x) %in% c("THA","MMR","CHN","USA","PHL","FRA","MYS",
                              "GBR","AZE","BRN","CRI","CMR"))
print(data[ii,1:2])
```

```
##               country code
## 14          Azerbaijan  AZE
## 30  Brunei Darussalam  BRN
## 38               China  CHN
## 40            Cameroon  CMR
## 45          Costa Rica  CRI
## 64              France  FRA
```

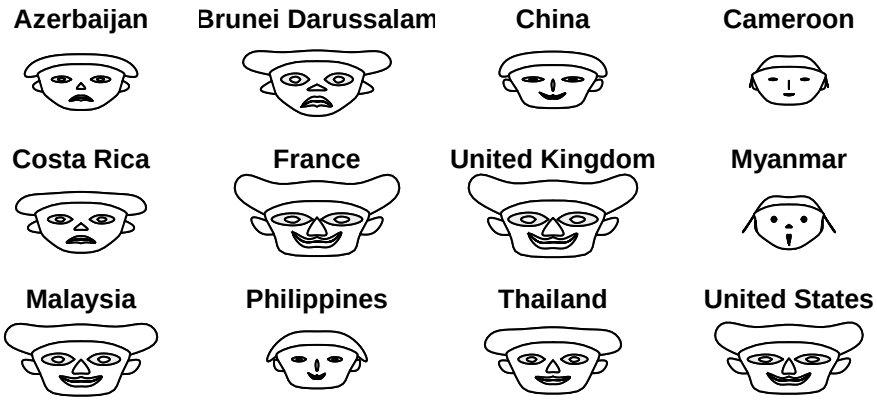
```
## 68     United Kingdom  GBR
## 127           Myanmar  MMR
## 135          Malaysia  MYS
## 149       Philippines  PHL
## 182          Thailand  THA
## 194     United States  USA
```

We then run function `PlotFaces`:

```
library(DescTools)
```

```
## Warning: package 'DescTools' was built under R version 4.0.2
```

```
PlotFaces(x[ii,],labels=data[ii,1])
```



We will study later more sophisticated techniques for plotting multi-dimensional data.

Clustering

Application of the HCM algorithm

We start by standardizing the data:

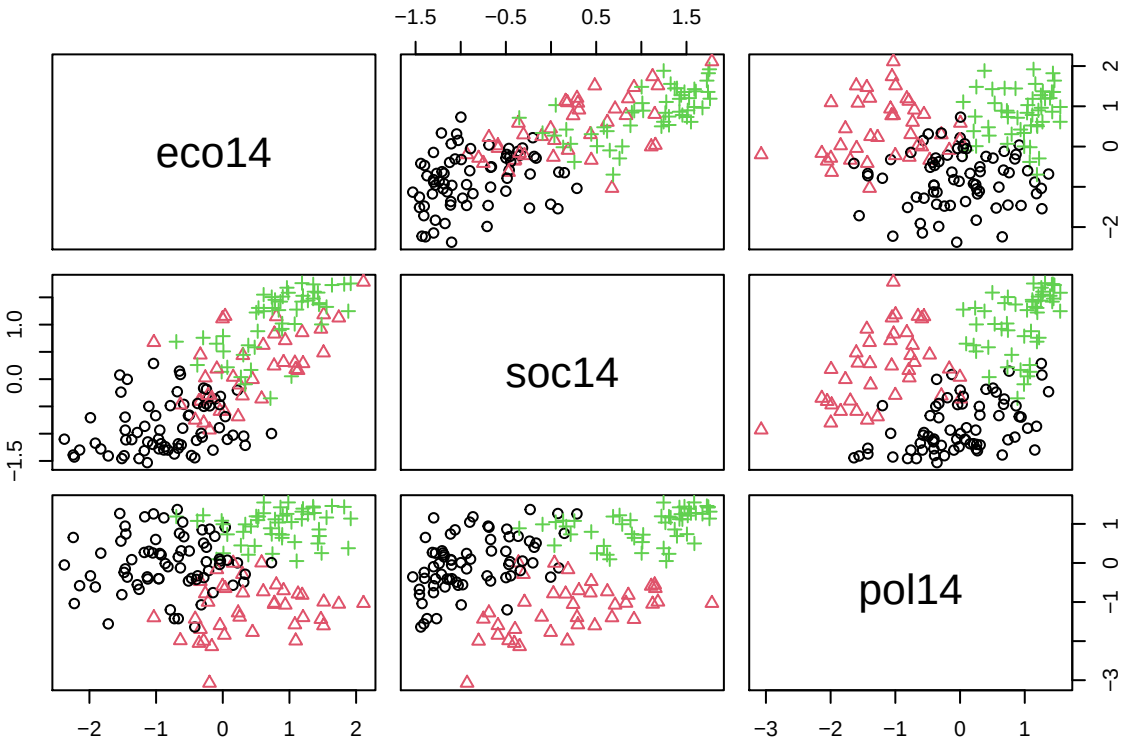
```
x<-scale(x)
```

We run the HCM algorithm with $c = 3$ clusters:

```
km <- kmeans(x,centers=3,nstart=10)
```

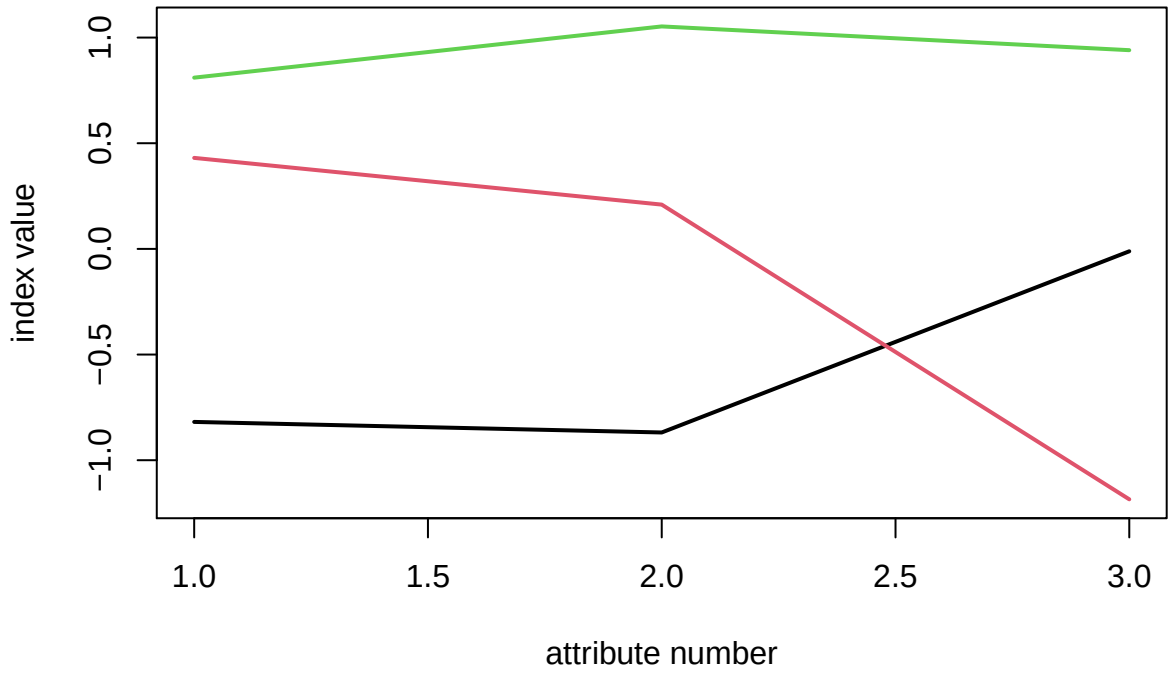
The partition is coded in vector `km$cluster`. We can plot the data with different groups in different colors:

```
pairs(x,col=km$cluster,pch=km$cluster)
```



The prototypes are in matrix `km$centers`. We can plot them as profiles as a help to interpret the clusters:

```
plot(km$centers[1,],type="l",ylim=range(km$centers),lwd=2,xlab="attribute number",
ylab="index value")
lines(km$centers[2,],lwd=2,col=2)
lines(km$centers[3,],lwd=2,col=3)
```



We can now look at the list of countries in each group. We will order the countries by decreasing distance to the center of their group. We start with group 1:

```
ii<-which(km$cluster==1)
n1<-length(ii)
D<-rep(0,n1)
for(i in 1:n1) D[i]<-sqrt(sum((x[ii[i],]-km$centers[1,])^2))
Ds<-sort(D,index.return=TRUE)
print(data.frame(data[ii[Ds$ix],1],Ds))
```

```
##          data.ii.Ds.ix...1.         x ix
## 1             Cote d'Ivoire 0.2912906 13
## 2                    Rwanda 0.3725872 54
## 3                     Niger 0.3887498 47
## 4                     Benin 0.3975134  6
## 5                    Uganda 0.4401962 64
## 6                    Malawi 0.4762658 45
## 7                      Mali 0.4818469 40
## 8                   Bolivia 0.4948631  9
## 9              Burkina Faso 0.5063239  7
## 10                Sri Lanka 0.5615465 37
## 11             Yemen, Rep. 0.5760377 67
## 12             Gambia, The 0.5787060 25
## 13              Madagascar 0.5844393 39
## 14                Cameroon 0.6048000 14
## 15                  Guinea 0.6359734 24
## 16              Mozambique 0.6432973 43
## 17                Zimbabwe 0.6617405 70
## 18                 Algeria 0.6708282 17
## 19          Kyrgyz Republic 0.6886049 35
## 20                   Gabon 0.7078570 22
## 21              Tajikistan 0.7242036 61
## 22            Sierra Leone 0.7420657 57
## 23                 Namibia 0.7448231 46
## 24                    Togo 0.7760861 60
## 25         Congo, Dem. Rep. 0.8098805 68
## 26               Kazakhstan 0.8246667 33
## 27              Philippines 0.8285641 51
## 28                  Vietnam 0.8704495 66
## 29                     Chad 0.8809460 59
## 30                   Zambia 0.9223014 69
## 31                  Jamaica 0.9606502 32
## 32                     Fiji 0.9787336 21
## 33                 Paraguay 0.9828150 53
## 34                 Tanzania 0.9882907 63
## 35                    Kenya 1.0140599 34
## 36                    Ghana 1.0176499 23
## 37                 Colombia 1.0242992 16
## 38                  Senegal 1.0786687 56
## 39         Papua New Guinea 1.0953235 52
## 40                  Ecuador 1.1068387 18
## 41                Guatemala 1.1075879 27
## 42            Venezuela, RB 1.1167687 65
## 43               Bangladesh 1.1225225  8
## 44                 Cambodia 1.1323251 36
## 45                  Tunisia 1.1405540 62
## 46                 Pakistan 1.1650592 50
```

```
## 47                 Lesotho 1.1775688 38
## 48 Central African Republic 1.2112664 11
## 49       Iran, Islamic Rep. 1.2193975 31
## 50                  Albania 1.2344916  3
## 51               Mauritania 1.2375158 44
## 52              Congo, Rep. 1.2426503 15
## 53                    India 1.2436406 30
## 54                  Nigeria 1.2449926 48
## 55     Syrian Arab Republic 1.2486333 58
## 56                Indonesia 1.2743401 29
## 57            Guinea-Bissau 1.2843911 26
## 58                    China 1.2861831 12
## 59                   Angola 1.4767766  2
## 60                    Haiti 1.5033377 28
## 61                  Burundi 1.5086259  5
## 62                   Brazil 1.5258540 10
## 63                 Mongolia 1.5552960 42
## 64                    Nepal 1.5751204 49
## 65                 Ethiopia 1.6551912 20
## 66         Egypt, Arab Rep. 1.7293026 19
## 67                Argentina 1.7463170  4
## 68                  Myanmar 1.7716188 41
## 69                    Sudan 1.8300804 55
## 70              Afghanistan 1.8688006  1
```

We repeat the same operations for groups 2 and 3:

```
ii<-which(km$cluster==2)
n1<-length(ii)
D<-rep(0,n1)
for(i in 1:n1) D[i]<-sqrt(sum((x[ii[i],]-km$centers[2,])^2))
Ds<-sort(D,index.return=TRUE)
print(data.frame(data[ii[Ds$ix],1],Ds))
```

```
##      data.ii.Ds.ix...1.         x ix
## 1   Trinidad and Tobago 0.3641290 38
## 2        Macedonia, FYR 0.4106638 26
## 3            Costa Rica 0.4959244 13
## 4              Barbados 0.6252093  8
## 5               Lebanon 0.6375158 21
## 6                Panama 0.6863144 32
## 7     Brunei Darussalam 0.7195832  9
## 8               Iceland 0.7269537 17
## 9                  Oman 0.7658230 31
## 10           Montenegro 0.8029299 28
## 11             Georgia 0.8047236 14
## 12           Azerbaijan 0.8211125  3
## 13            Nicaragua 0.8434586 30
## 14                Libya 0.9239727 22
## 15               Guyana 1.0149702 15
## 16              Armenia 1.0419734  2
## 17           Seychelles 1.0454510 37
## 18               Latvia 1.0614351 23
## 19              Belarus 1.0752238  6
## 20               Belize 1.1031713  7
```

```
## 21             Botswana 1.1128594 11
## 22              Moldova 1.1429160 24
## 23               Kuwait 1.1560985 20
## 24         Saudi Arabia 1.1621354 33
## 25               Israel 1.1882746 18
## 26            Mauritius 1.1909749 29
## 27               Mexico 1.2263614 25
## 28            Swaziland 1.2533123 36
## 29              Bahrain 1.2854391  4
## 30              Vanuatu 1.3016156 39
## 31           Cape Verde 1.3241240 12
## 32             Honduras 1.3285684 16
## 33 United Arab Emirates 1.4687939  1
## 34               Bhutan 1.4819362 10
## 35             Suriname 1.4965153 35
## 36         Bahamas, The 1.5510553  5
## 37                Malta 1.6031725 27
## 38             Kiribati 2.2934347 19
## 39            Singapore 2.3055738 34
```

```
ii<-which(km$cluster==3)
n1<-length(ii)
D<-rep(0,n1)
for(i in 1:n1) D[i]<-sqrt(sum((x[ii[i],]-km$centers[3,])^2))
Ds<-sort(D,index.return=TRUE)
print(data.frame(data[ii[Ds$ix],1],Ds))
```

```
##        data.ii.Ds.ix...1.         x ix
## 1                Malaysia 0.2334809 30
## 2                  Poland 0.2350240 35
## 3                Bulgaria 0.2574874  4
## 4                Slovenia 0.3360564 43
## 5                 Romania 0.3692977 38
## 6           United States 0.4336163 49
## 7               Australia 0.4395632  1
## 8                  Greece 0.4480726 19
## 9             New Zealand 0.4881365 33
## 10               Croatia 0.5009557 20
## 11               Germany 0.5418430 11
## 12              Portugal 0.5566317 36
## 13                 Spain 0.6108992 14
## 14                Norway 0.6296991 32
## 15                 Italy 0.6657524 23
## 16               Finland 0.6805648 16
## 17         Czech Republic 0.7122721 10
## 18                Canada 0.7154920  6
## 19             Lithuania 0.7537957 27
## 20         United Kingdom 0.7634155 18
## 21               Thailand 0.7637907 45
## 22        Slovak Republic 0.7667131 42
## 23                Ukraine 0.8011103 47
## 24                Hungary 0.8249343 21
## 25                 France 0.8266394 17
## 26                  Japan 0.8503993 25
## 27                 Sweden 0.8553706 44
```

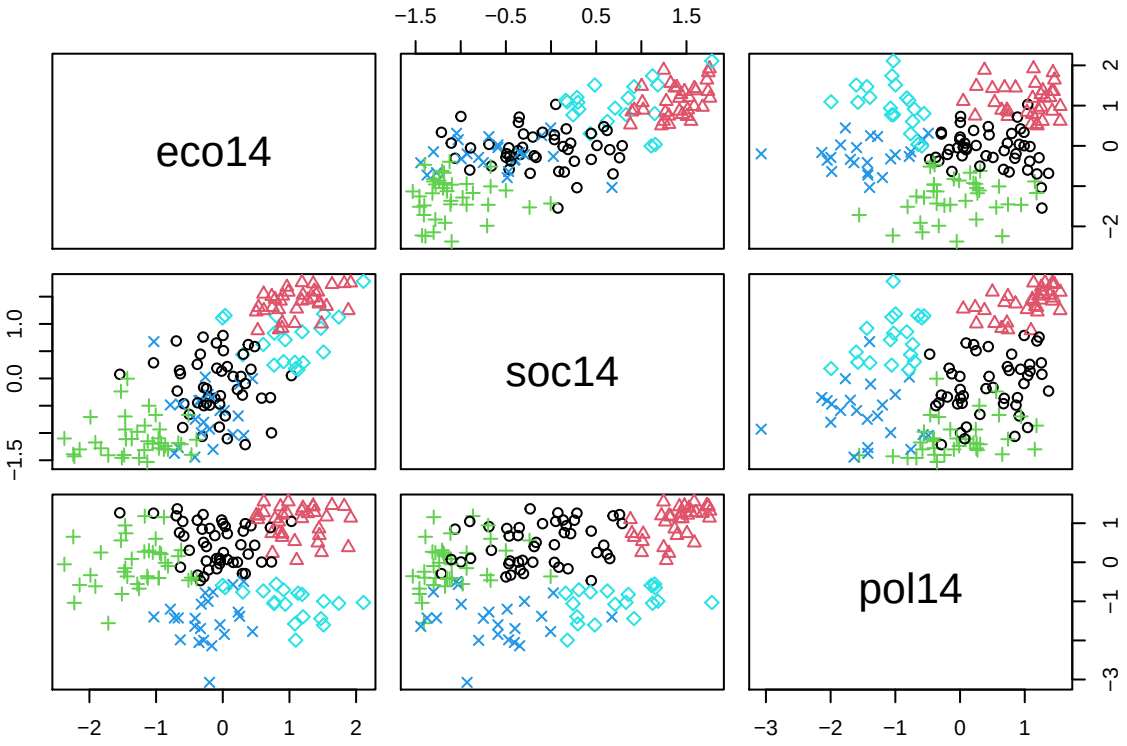
```
## 28                 Denmark 0.8569536 12
## 29             Switzerland 0.8846726  7
## 30                  Cyprus 0.9276809  9
## 31                  Serbia 0.9501800 41
## 32                 Estonia 0.9631855 15
## 33                  Jordan 0.9652743 24
## 34                   Qatar 0.9656954 37
## 35                 Austria 0.9988439  2
## 36                   Chile 1.0300587  8
## 37                 Uruguay 1.1371164 48
## 38                  Turkey 1.1774013 46
## 39                 Belgium 1.1859081  3
## 40 Bosnia and Herzegovina 1.1897442  5
## 41              Luxembourg 1.2264899 28
## 42            South Africa 1.2411096 50
## 43             Korea, Rep. 1.2438709 26
## 44             El Salvador 1.2499070 40
## 45      Dominican Republic 1.3029753 13
## 46             Netherlands 1.3196888 31
## 47                 Ireland 1.3261776 22
## 48                    Peru 1.4088093 34
## 49                 Morocco 1.4388894 29
## 50      Russian Federation 1.5707273 39
```

In the above analysis, the number of clusters was set arbitrarily to 3. let us repeat the analysis with $c = 5$ clusters:

```
km <- kmeans(x,centers=5,nstart=10)
```

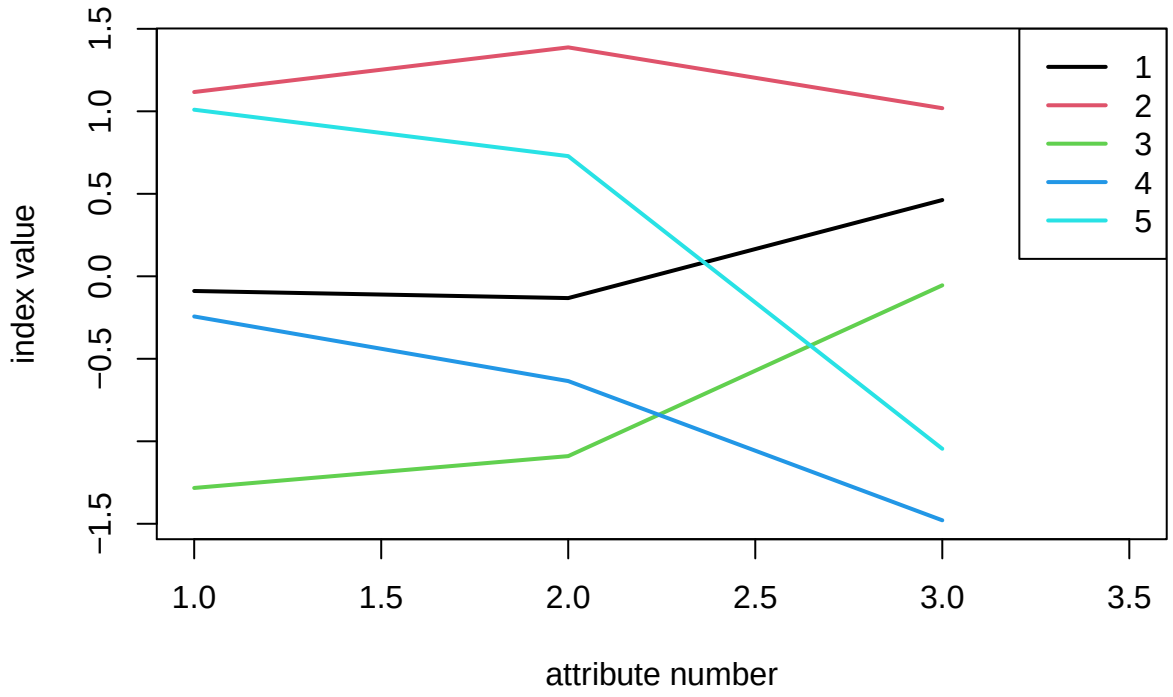
Plot of the data:

```
pairs(x,col=km$cluster,pch=km$cluster)
```



Cluster profiles:

```
plot(km$centers[1,],type="l",ylim=range(km$centers),lwd=2,xlab="attribute number",
ylab="index value",xlim=c(1,3.5))
for(k in 2:5) lines(km$centers[k,],lwd=2,col=k)
legend("topright",legend = 1:5,col=1:5,lty=1,lwd=2)
```



List of countries in each cluster:

```
for(k in 1:5){
  ii<-which(km$cluster==k)
  n1<-length(ii)
  D<-rep(0,n1)
  for(i in 1:n1) D[i]<-sqrt(sum((x[ii[i],]-km$centers[k,])^2))
  Ds<-sort(D,index.return=TRUE)
  cat("\n")
  print(paste("Cluster",k))
  print(data.frame(data[ii[Ds$ix],1],Ds))
}
```

```
##
## [1] "Cluster 1"
##        data.ii.Ds.ix...1.         x ix
## 1                Paraguay 0.1756151 33
## 2                Colombia 0.2063238 10
## 3               Guatemala 0.3217621 16
## 4             El Salvador 0.3991937 36
## 5                 Jamaica 0.4547010 19
## 6                 Uruguay 0.4683414 43
## 7                 Tunisia 0.5492981 40
## 8                  Mexico 0.5520304 27
## 9                 Albania 0.5583420  1
## 10                  Gabon 0.5597393 14
```

```
## 11                Vietnam 0.6034592 44
## 12            Philippines 0.6315691 32
## 13            Korea, Rep. 0.6779434 24
## 14           South Africa 0.6784020 45
## 15 Bosnia and Herzegovina 0.6828657  4
## 16                Bolivia 0.6882437  6
## 17                  China 0.6924498  9
## 18             Kazakhstan 0.7013718 22
## 19                Moldova 0.7062832 26
## 20              Indonesia 0.7271341 18
## 21                 Jordan 0.7590191 20
## 22                Ukraine 0.7768084 42
## 23                Morocco 0.7848199 25
## 24                   Fiji 0.8315821 13
## 25                   Togo 0.8427083 38
## 26               Honduras 0.8445223 17
## 27                Armenia 0.8615568  3
## 28     Dominican Republic 0.8647648 11
## 29        Kyrgyz Republic 0.8979290 23
## 30               Thailand 0.9144871 39
## 31                 Serbia 0.9235328 37
## 32                Namibia 0.9345127 30
## 33                   Peru 0.9357293 31
## 34                  Ghana 1.0305637 15
## 35                 Zambia 1.0604217 46
## 36                  Japan 1.0656663 21
## 37                 Brazil 1.0870748  7
## 38                Senegal 1.0909734 35
## 39                Belarus 1.1238085  5
## 40                 Turkey 1.1881590 41
## 41     Russian Federation 1.2520935 34
## 42                  Chile 1.2760582  8
## 43               Mongolia 1.2761428 28
## 44       Egypt, Arab Rep. 1.3065741 12
## 45             Mauritania 1.3843194 29
## 46              Argentina 1.6753078  2
##
## [1] "Cluster 2"
##    data.ii.Ds.ix...1.         x ix
## 1            Portugal 0.1283477 28
## 2             Finland 0.2854024 13
## 3              Poland 0.3042176 27
## 4             Germany 0.3161021  9
## 5           Australia 0.3537360  1
## 6      Czech Republic 0.3643215  8
## 7              Canada 0.4045990  5
## 8      United Kingdom 0.4067155 15
## 9             Denmark 0.4243201 10
## 10            Hungary 0.4672257 18
## 11    Slovak Republic 0.4681143 31
## 12              Spain 0.4784197 11
## 13        Switzerland 0.4805685  6
## 14            Croatia 0.4882197 17
## 15             Sweden 0.4962680 33
```

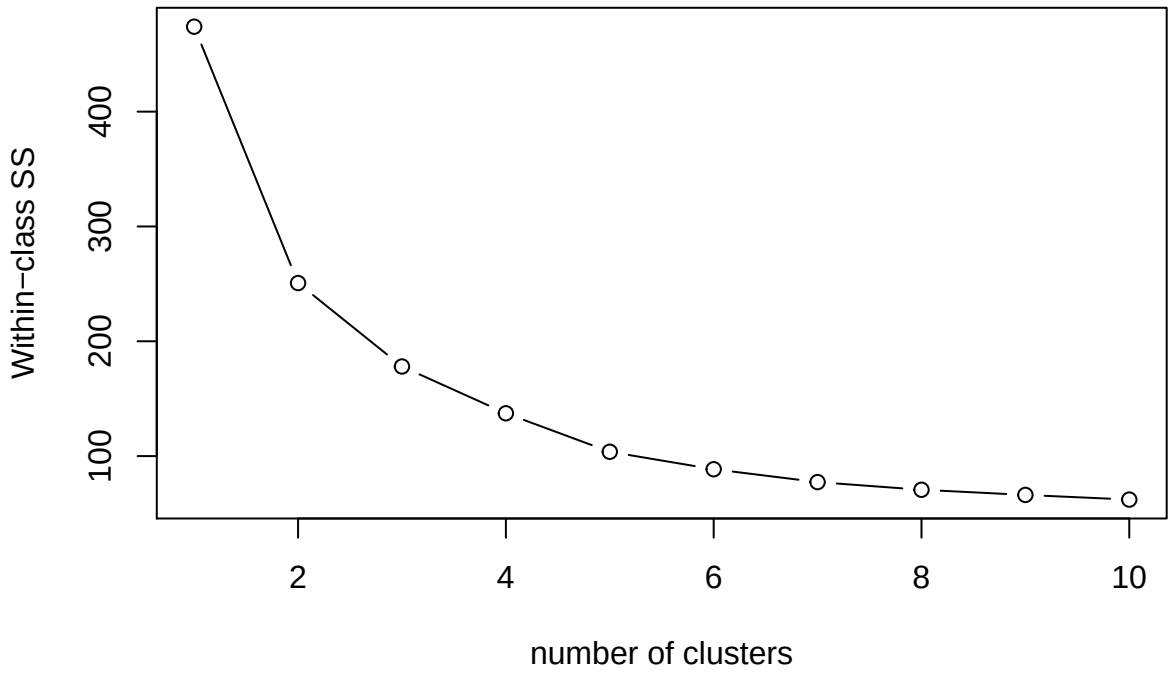
```
## 16           Malaysia 0.5217476 23
## 17             France 0.5817287 14
## 18            Austria 0.5840418  2
## 19             Norway 0.5872981 25
## 20             Greece 0.6106987 16
## 21        New Zealand 0.6142532 26
## 22           Bulgaria 0.6272496  4
## 23             Cyprus 0.6373118  7
## 24           Slovenia 0.6414646 32
## 25      United States 0.6604038 34
## 26              Italy 0.7418075 20
## 27            Belgium 0.7612125  3
## 28            Romania 0.7847767 30
## 29          Lithuania 0.8729587 21
## 30            Ireland 0.8895352 19
## 31        Netherlands 0.8917519 24
## 32            Estonia 0.9321559 12
## 33              Qatar 0.9758440 29
## 34         Luxembourg 1.0084901 22
##
## [1] "Cluster 3"
##          data.ii.Ds.ix...1.         x ix
## 1                  Cameroon 0.1633949  8
## 2                  Zimbabwe 0.2559504 37
## 3              Yemen, Rep. 0.3930346 35
## 4                      Mali 0.4029539 20
## 5                    Guinea 0.4073676 12
## 6                      Chad 0.4151584 30
## 7                    Rwanda 0.4474275 27
## 8                     Benin 0.4537042  3
## 9                    Malawi 0.4720130 22
## 10            Cote d'Ivoire 0.4764420  7
## 11                   Uganda 0.5087150 33
## 12               Madagascar 0.5134626 19
## 13               Tajikistan 0.5144474 31
## 14         Congo, Dem. Rep. 0.5550700 36
## 15               Bangladesh 0.6546624  5
## 16                    Niger 0.6578699 23
## 17                Sri Lanka 0.7092589 18
## 18                 Tanzania 0.7211523 32
## 19             Burkina Faso 0.7630242  4
## 20                    Kenya 0.8140562 17
## 21                  Algeria 0.8265043  9
## 22       Iran, Islamic Rep. 0.8453521 16
## 23            Guinea-Bissau 0.8454677 14
## 24 Central African Republic 0.8716083  6
## 25             Sierra Leone 0.9340420 29
## 26               Mozambique 0.9622093 21
## 27              Gambia, The 0.9656676 13
## 28                  Burundi 1.0339870  2
## 29                  Ecuador 1.0779127 10
## 30                 Pakistan 1.0878515 26
## 31                    Nepal 1.0934361 25
## 32            Venezuela, RB 1.1385725 34
```

```
## 33                 Ethiopia 1.2292886 11
## 34                    India 1.2616995 15
## 35                  Nigeria 1.2910130 24
## 36                    Sudan 1.4045925 28
## 37              Afghanistan 1.5976122  1
##
## [1] "Cluster 4"
##      data.ii.Ds.ix...1.         x ix
## 1               Vanuatu 0.2100052 23
## 2                Guyana 0.2394058 10
## 3                Belize 0.3334209  4
## 4              Botswana 0.4564628  7
## 5                 Libya 0.5185879 14
## 6                Bhutan 0.5459418  6
## 7               Lesotho 0.5484944 15
## 8             Nicaragua 0.5873985 18
## 9            Cape Verde 0.6286024  9
## 10 Syrian Arab Republic 0.6288184 22
## 11             Suriname 0.6613754 20
## 12            Swaziland 0.7218434 21
## 13       Macedonia, FYR 0.7337052 16
## 14               Angola 0.7633854  1
## 15              Myanmar 0.8413024 17
## 16                Haiti 0.8832135 11
## 17           Azerbaijan 0.9588293  2
## 18             Barbados 0.9794318  5
## 19     Papua New Guinea 0.9854354 19
## 20              Cambodia 1.0663298 12
## 21          Congo, Rep. 1.2063797  8
## 22         Bahamas, The 1.5320393  3
## 23             Kiribati 1.6221130 13
##
## [1] "Cluster 5"
##      data.ii.Ds.ix...1.          x ix
## 1     Brunei Darussalam 0.08080155  3
## 2               Iceland 0.25678355  6
## 3                Latvia 0.31412727 10
## 4               Lebanon 0.51960207  9
## 5   Trinidad and Tobago 0.54029315 19
## 6                Panama 0.54887712 15
## 7               Georgia 0.59436014  5
## 8            Montenegro 0.63215080 12
## 9               Bahrain 0.63743731  2
## 10               Israel 0.67543702  7
## 11 United Arab Emirates 0.68104290  1
## 12                 Oman 0.68983296 14
## 13            Mauritius 0.78946027 13
## 14           Costa Rica 0.81684987  4
## 15                Malta 0.83111926 11
## 16           Seychelles 1.09722406 18
## 17               Kuwait 1.13495212  8
## 18         Saudi Arabia 1.17380107 16
## 19            Singapore 1.52337453 17
```

Determining the optimal number of clusters

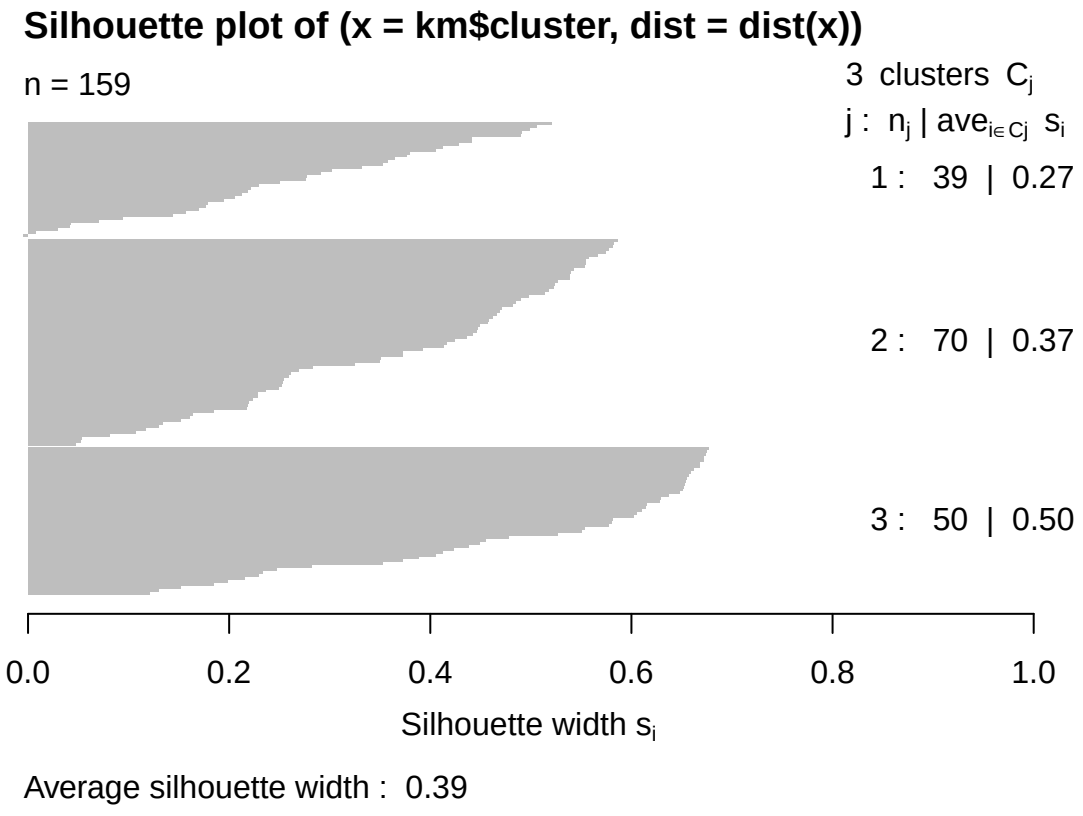
We can first try the "knee" method:

```
C<- 1:10
N<-length(C)
J<-rep(0,N)
for(i in 1:N){
  km<-kmeans(x,centers=C[i],nstart=10)
  J[i]<-km$tot.withinss
}
plot(C,J,type="b",xlab="number of clusters",ylab="Within-class SS")
```



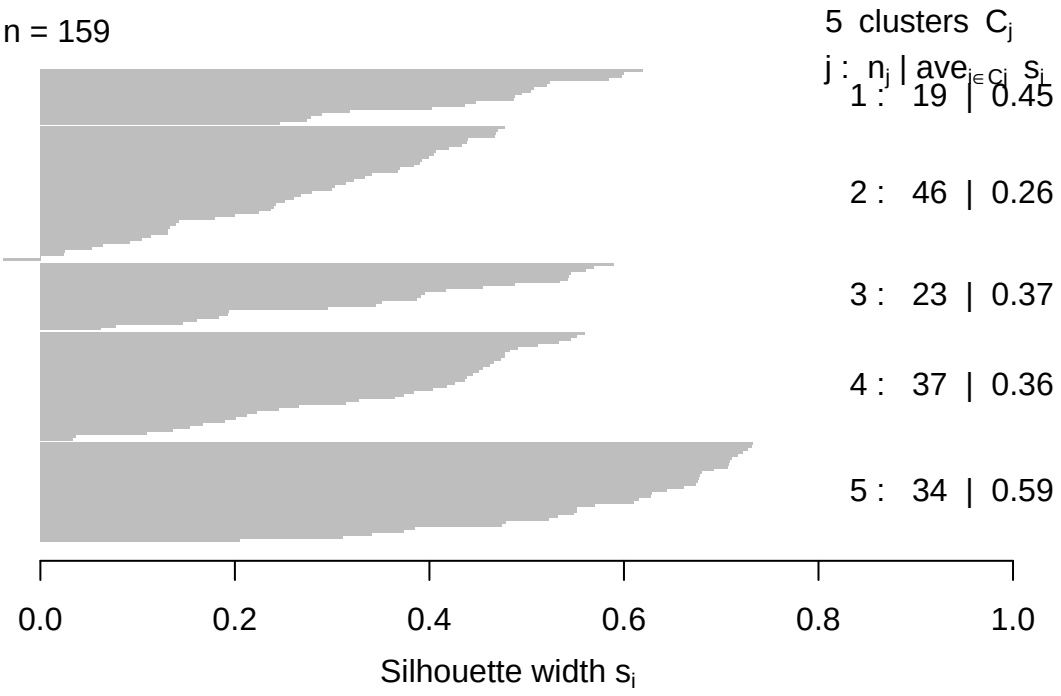
It does not help us to choose the number of clusters. Let us try the silhouette plot, for 3 clusters first:

```
library(cluster)
km<-kmeans(x,centers=3,nstart=10)
sil<-silhouette(km$cluster,dist(x))
plot(sil)
```



Then, for 5 clusters:

```
km<-kmeans(x,centers=5,nstart=10)
sil<-silhouette(km$cluster,dist(x))
plot(sil)
```



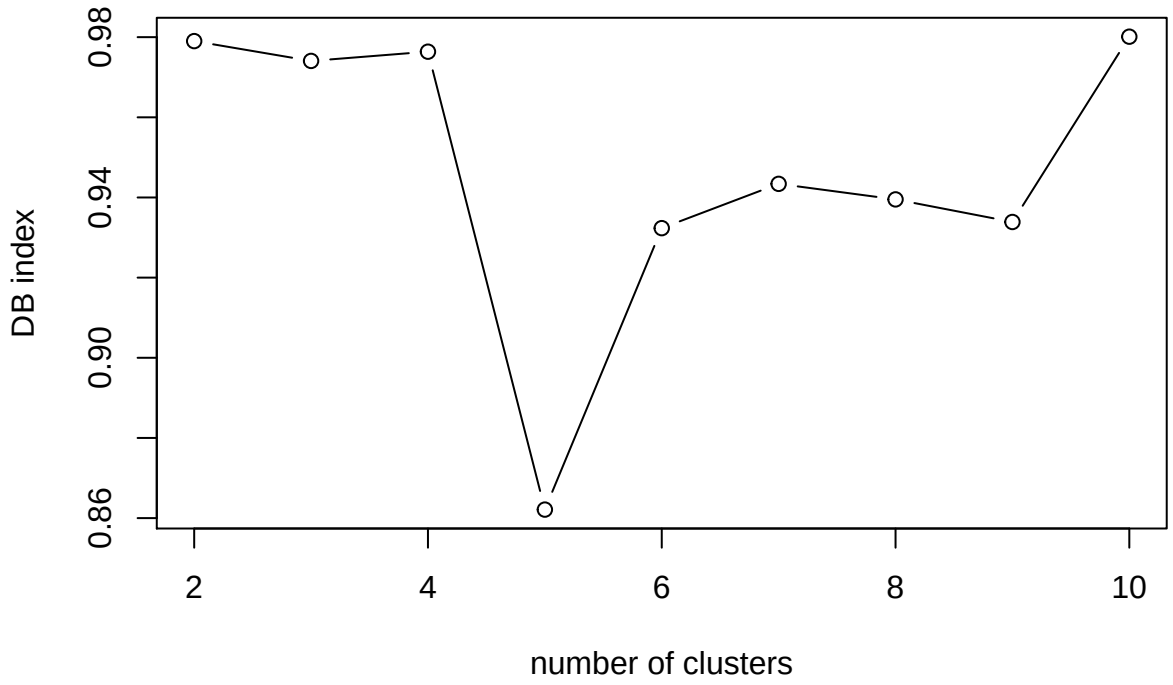
Average silhouette width : 0.39

Let us now plot the silhouette, Davis-Bouldin and Dunn indices for differents numbers of clusters:

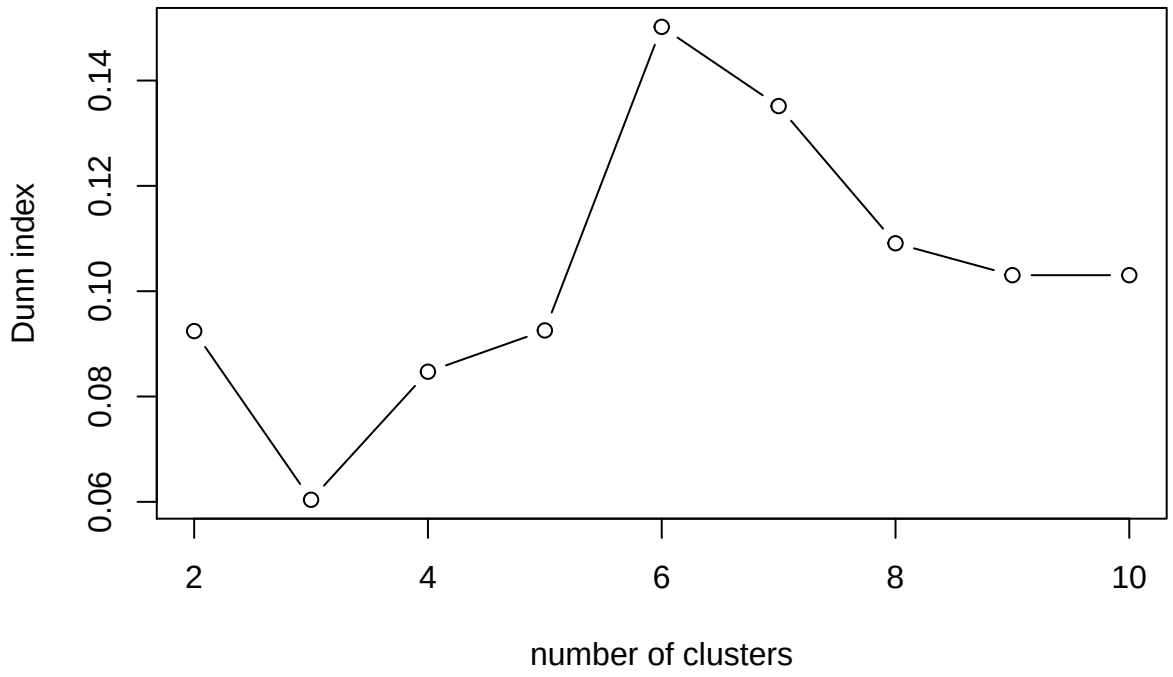
```
library(clusterCrit)
```

```
## Warning: package 'clusterCrit' was built under R version 4.0.2
```

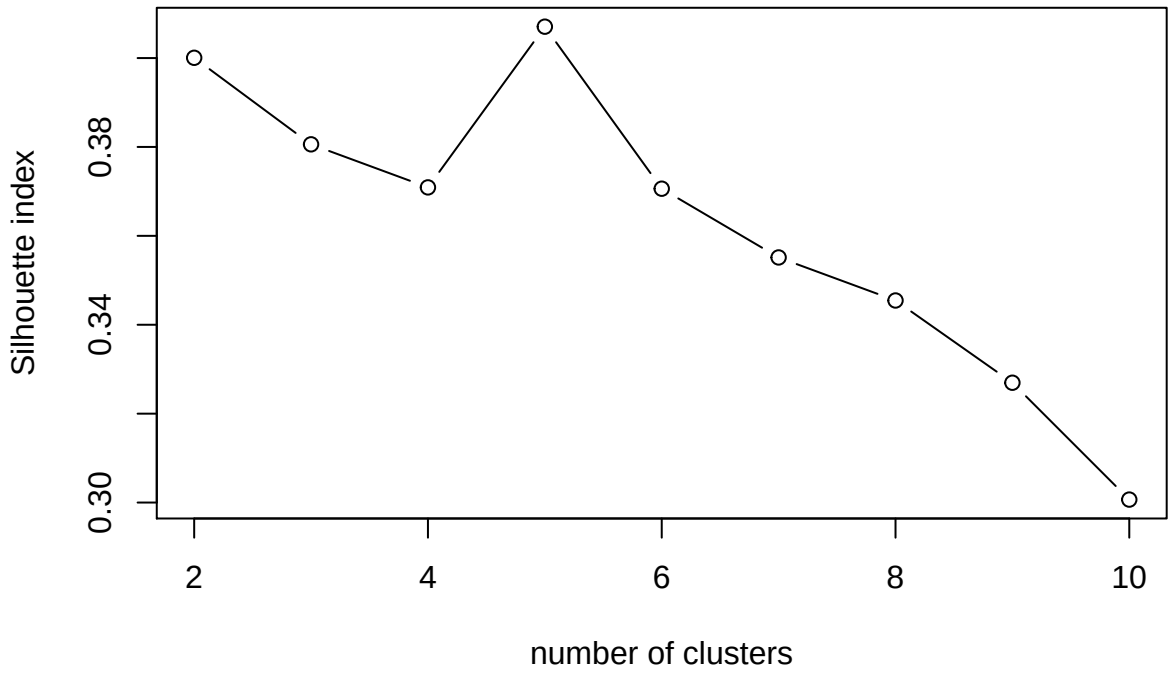
```
C<- 2:10
N<-length(C)
DB<-rep(0,N)
Du<-rep(0,N)
Si<-rep(0,N)
for(i in 1:N){
  km<-kmeans(x,centers=C[i],nstart=50)
  DB[i]<-intCriteria(as.matrix(x), km$cluster, crit="Davies_Bouldin")
  Du[i]<-intCriteria(as.matrix(x), km$cluster, crit="Dunn")
  Si[i]<-intCriteria(as.matrix(x), km$cluster, crit="Silhouette")
}
plot(C,DB,type="b",xlab="number of clusters",ylab="DB index")
```




```
plot(C,Du,type="b",xlab="number of clusters",ylab="Dunn index")
```




```
plot(C,Si,type="b",xlab="number of clusters",ylab="Silhouette index")
```



The DB and silhouette indices suggest a partition in 5 clusters, while the Dunn index is maximum for 6 clusters.

Fuzzy clustering

We apply the FCM algorithm with 5 clusters:

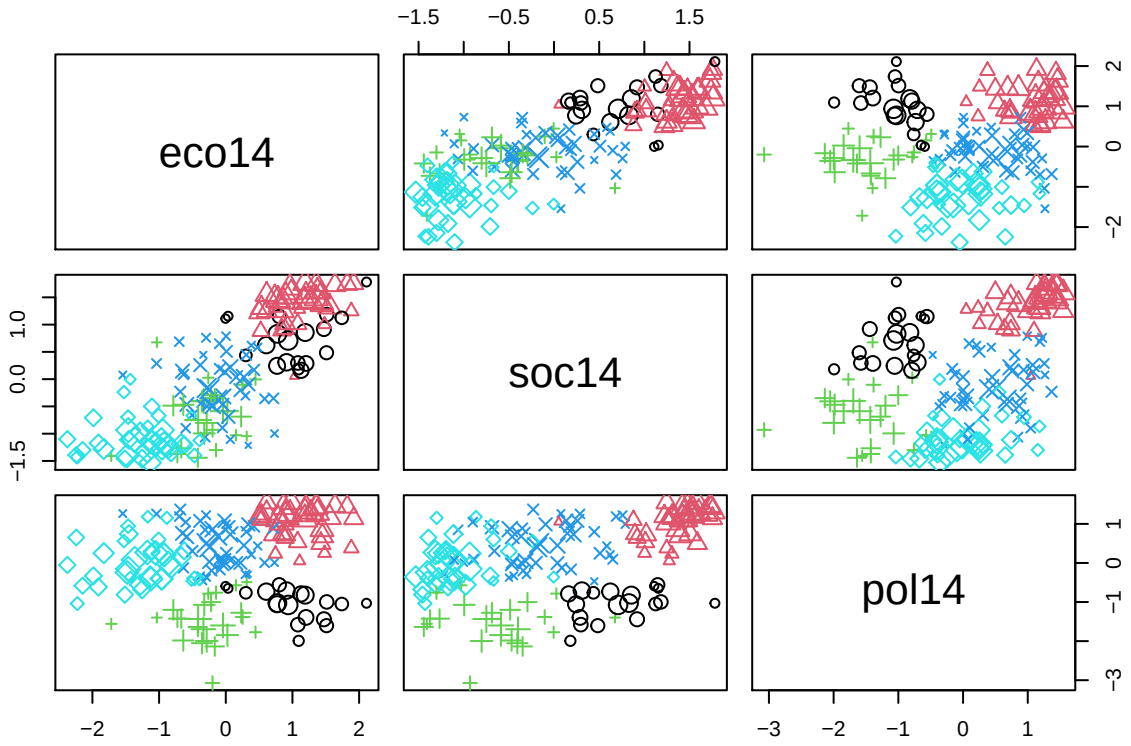
```
library(fclust)
```

```
## Registered S3 method overwritten by 'fclust':
##   method       from
##   print.fclust e1071
```

```
fkm<-FKM(x,k=5,RS=10)
```

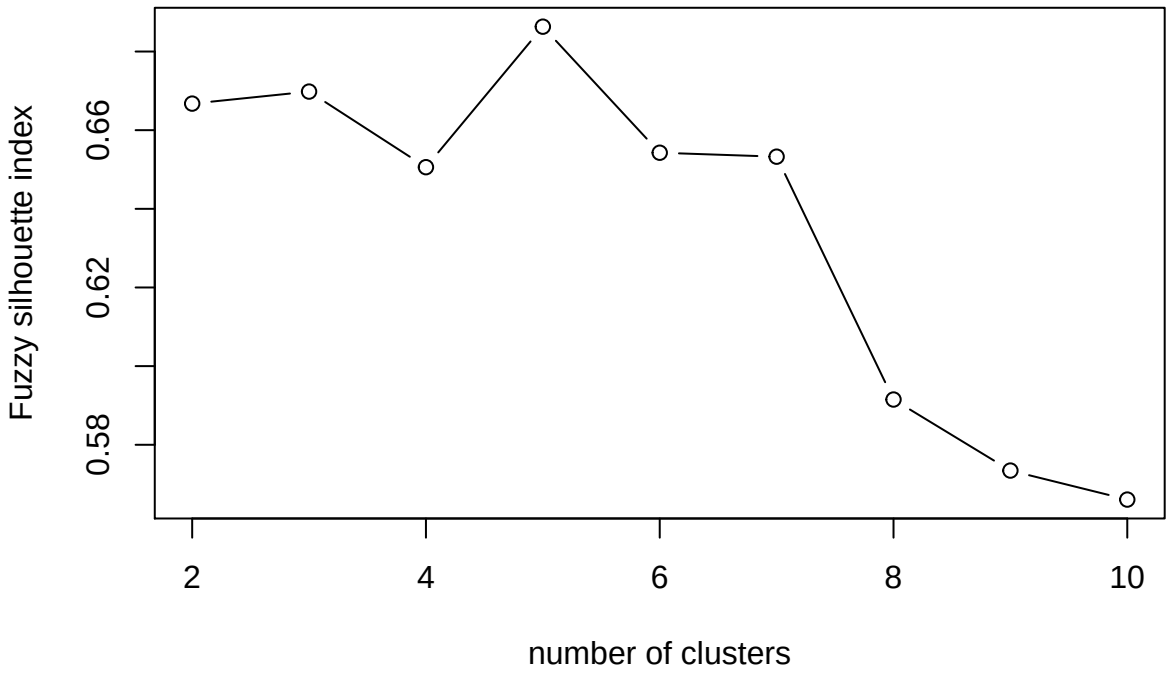
We plot the result, making the size of the symbol proportional to the maximum membership degree:

```
pairs(x,col=fkm$clus[,1],pch=fkm$clus[,1],cex=fkm$clus[,2]*2)
```



We can now plot the fuzzy silhouette index for different numbers of cluster:

```
C<- 2:10
N<-length(C)
FSI<-rep(0,N)
for(i in 1:N){
  fkm<-FKM(x,C[i],RS=10,index="SIL.F")
  FSI[i]<-fkm$criterion
}
plot(C,FSI,type="b",xlab="number of clusters",ylab="Fuzzy silhouette index")
```


Fuzzy silhouette index
0.58
0.62
0.66
2
4
6
8
10
number of clusters