NbClust Package : finding the relevant number of clusters in a dataset

Malika Charrad, Nadia Ghazzali, Véronique Boiteau, Azam Niknafs

Laval University, Quebec, Canada

June 13th, 2012

Outline

Outline

Introduction

- Clustering is the task of assigning a set of objects into groups (clusters) so that the objects in the same cluster are more similar to each other than objects in other clusters.
- Most of the clustering algorithms depend on input parameters such as **the number of clusters**, the minimum number of objects in a cluster, or the diameter of a cluster ..
  $\Rightarrow$ The selection of different parameters leads to different clusters of data.

How many clusters are there in the dataset ?

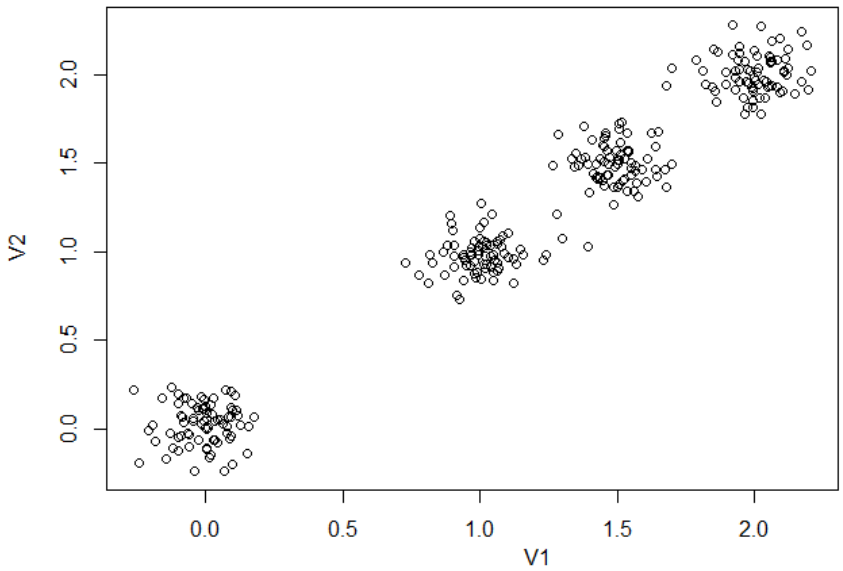


Fig.1

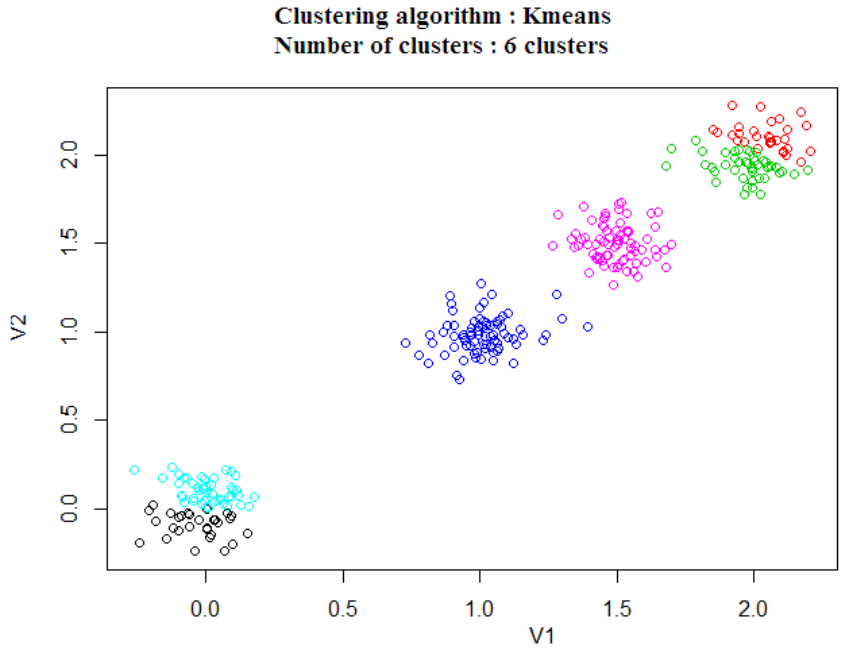


Fig.2

How to select the best number of clusters in a dataset ?

- If the clustering algorithm parameters are assigned improper values, the clustering method may result in a partitioning scheme that's not optimal $\Rightarrow$ Wrong decisions.
- The user is faced with the dilemma of selecting the number of clusters in the dataset.

- The problem of deciding the number of clusters better fitting a dataset as well as the evaluation of the clustering results is known under the term **cluster validity**.

Related work

(Milligan and Cooper, 1985) examined 30 indices with simulated data.
There are other criteria which were not examined in Milligan and Cooper study such as :

- Dunn index (Dunn, 1974)
- Silhouette statistic (Rousseeuw, 1987)
- Gap statistic (Tibshirani, 2001)
- Dindex (Lebart, 2000)
- SD index and SDbw index (Halkidi et al., 2000, 2001)
- Statistic of Hubert ((Hubert and Arabie, 1985))

Related work

$\Rightarrow$ 19 among all existing indices are implemented in SAS and R packages : **cclust**, **clusterSim**, **clv** and **clvalid**.

SAS	R			
Cluster	**clusterSim**	**cclust**	**clv**	**clvalid**
1. CH index (Pseudo-F or index.G1)		**8.** Ratkowsky	19. dunn	
2. CCC («Cubic clustering criterion»)	**4.** KL (Krzanowski-Lay)	**9.** Scott		
3. pseudo-t2	**5.** Gamma (or index.G2)	**10.** Marriot		
	6. Gap	**11.** Ball		
	7. Silhouette	**12.** trcovw		
		13. tracew		
		14. Friedman		
		15. Rubin		
	16. Hartigan			
	17. C-index (index.G1 or Hubert &Levine index)			
	18. DB (Davies-Bouldin)			

NbClust package

1. **NbClust** package provides **30 indices** to determine the number of clusters :
   - 11 other indices :
     - **"duda"** Duda and Hart (1973)
     - **"beale"** Beale (1969)
     - **"gplus"** Rohlf (1974), Milligan (1981)
     - **"frey"** Frey and Van Groenewoud (1972)
     - **"tau"** Rohlf (1974), Milligan (1981)
     - **"mcclain"** McClain and Rao (1975),
     - **"gap"** Tibshirani (2001),
     - **"dindex"** Lebart (2000),
     - **"hubert"** Hubert and Arabie (1985),
     - **"sdindex"** Halkidi et al. (2000),
     - **"sdbw"** Halkidi et al. (2001).
2. **NbClust** offers the user the best clustering scheme among different results.

NbClust function

NbClust(data, diss="NULL", distance="euclidean", min.nc=2, max.nc=15, method="ward", index="all", alphaBeale=0.1)

Arguments :

data matrix or data set

diss dissimilarity matrix to be used. By default, diss="NULL", but if it is replaced by a dissimilarity matrix, *distance* should be "NULL".

distance the distance measure to be used to compute the dissimilarity matrix. This must be one of : **"euclidean"**, **"maximum"**, **"manhattan"**, **"canberra"**, **"binary"**, **"minkowski"** or "NULL".

NbClust function

NbClust(data, diss="NULL", distance="euclidean", min.nc=2, max.nc=15, method="ward", index="all", alphaBeale=0.1)

Arguments :

min.nc minimum number of clusters, between 2 and (number of objects - 1).

max.nc maximum number of clusters, between 2 and (number of objects - 1), greater or equal to min.nc.

method the cluster analysis method to be used. Available methods are :

- **"ward"**, **"single"**, **"complete"**, **"average"**, **"mcquitty"**, **"median"**, **"centroid"**
- **"kmeans"**

NbClust function

Nb.clusters(data, diss="NULL", distance="euclidean", min.nc=2, max.nc=15, method="ward", index="all", alphaBeale=0.1)

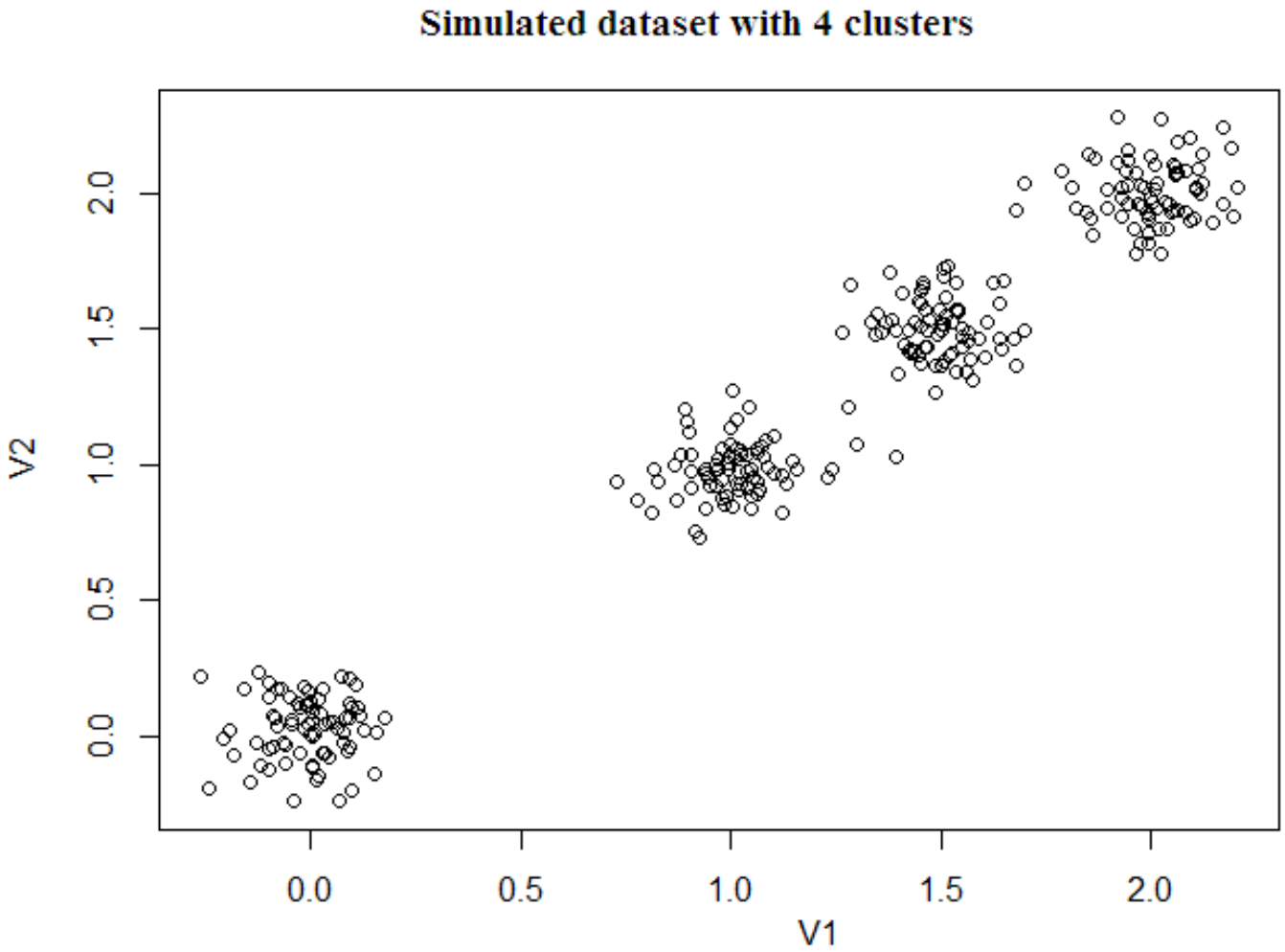
Arguments :

index the index to be calculated. This should be one of :

- **"kl", "ch", "hartigan", "ccc", "scott", "marriot", "trcovw", "tracew", "friedman", "rubin", "cindex", "db", "silhouette", "duda", "pseudot2", "beale", "ratkowsky", "ball", "ptbiserial", "gap", "frey", "mcclain", "gamma", "gplus", "tau", "dunn", "hubert", "sdindex", "dindex", "sdbw",**
- **"alllong"** : all indices included
- **"all"** : all indices except GAP, Gamma, Gplus and Tau.

alphaBeale significance value for Beale's index.

Example1 : Simulated dataset with 2 variables and 4 clusters



NbClust output : Gap index

```
> NbClust(data, diss="NULL", distance = "euclidean",
+         min.nc=2, max.nc=8, method = "ward",
+         index = "gap", alphaBeale = 0.1)

[1] "All 300 observations were used."

$All.index
  nc.Ward index.Gap
2       2 0.9025077
3       3 1.6141331
4       4 2.5794795
5       5 2.4893056
6       6 2.4039888
7       7 2.3342145
8       8 2.3002805


$All.CriticalValues
  nc.CritValue CritValue_Gap
2            2   -0.69921760
3            3   -0.92360136
4            4    0.14138852
5            5    0.12416949
6            6    0.07796550
7            7    0.09005784
8            8    0.05640620

$Best.nc
                [,1]
Number_clusters 4.0000
Value_Index     2.5795
```

NbClust output : "alllong" option

[All.index] Values of indices for each partition of the dataset obtained with a number of clusters between *min.nc* and *max.nc*.

R Console

Fichier Edition Misc Packages Fenêtres Aide

```
> library(NbClust)
> donnees300obs <- read.table("C:/Users/CRSNG-INAL-3/Desktop/Malika/Dropbox/jeuxDonnees/donnees300obs.txt", header=T, quote="\"")
> NbClust(donnees300obs, diss="NULL", distance = "euclidean", min.nc=2, max.nc=8, method = "complete", index = "alllong", alphaBeale = 0.1)
[1] "*** : 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 co
[1] "*** : 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
[1] "All 300 observations were used."
$All.index
  nc.Ward   index.KL  index.CH index.Hartigan index.CCC index.Scott index.Marriot index.TrCovW index.TraceW index.Friedman index.Rubin index.Cindex
2       2  1.8651459  895.6139      668.70156  20.64425    792.1787      996.4123    11.978858    82.786541       167.5615    13.17912    0.3321567
3       3  1.7811088 1781.0241      843.43401  22.06843   1174.2771      627.3033    10.635141    25.520171       195.0933    42.75261    0.3067675
4       4 13.0962656 4824.0849       27.55568  31.03811   1729.3586      175.3041     9.823584     6.646146       328.9135   164.16339    0.2754854
5       5  0.9214796 3948.3865       16.55901  25.77902   1782.2822      229.6135     9.825536     6.080126       359.1440   179.44594    0.3077053
6       6  2.7710786 3328.0067       34.77563  21.72565   1818.1044      293.4283     8.492045     5.756974       383.3275   189.51864    0.3121072
7       7  0.5380998 3096.6092       17.89684  19.37835   1880.4782      324.4145     6.498847     5.148040       422.1872   211.93575    0.3091138
8       8  4.3801704 2809.2966       31.21993  17.03546   1917.8071      374.1494     5.099828     4.851693       450.7319   224.88108    0.3192735
   index.DB index.Silhouette index.Duda index.Pseudot2 index.Beale index.Ratkowsky index.Ball index.ptbiserial index.Gap index.Frey index.McClain
2 0.3564597        0.7010239  0.2940155      535.46336   2.3904614       0.6124279 41.3932706        0.8007461 0.8734741  1.4312724     0.1873365
3 0.4363813        0.6805459  0.1452106      871.20957   5.8470441       0.5546613  8.5067237        0.7312520 1.6401954  0.8983988     0.3643208
4 0.3627973        0.7603273  0.6808057       34.69475   0.4625966       0.4949585  1.6615365        0.6616059 2.6384985  6.9213029     0.3707219
5 0.7931844        0.6479575  0.8173179       16.31653   0.2204936       0.4430897  1.2160251        0.6161204 2.4991749  3.1934640     0.4306399
6 0.9250651        0.5465349  0.6355700       41.85753   0.5656423       0.4046820  0.9594956        0.6020981 2.3657402  4.1613665     0.4479539
7 1.0916315        0.4207703  0.4790411       23.92508   1.0402207       0.3750175  0.7354344        0.5621825 2.3064778  2.6922174     0.5091377
8 0.9642459        0.4231145  0.6618982       34.22403   0.5032946       0.3509555  0.6064616        0.5580119 2.2739200  3.1464080     0.5151708
  index.Gamma index.Gplus index.Tau index.Dunn index.Hubert index.SDindex index.Dindex index.SDbw
2   0.9245509   397.07768  9731.555 0.47994404  0.003840629      3.265763    0.4250368  0.3378019
3   0.9336786   347.79206  9792.501 0.22037303  0.004223950      2.968122    0.2523720  0.3742208
4   0.9987684     5.14408  8343.060 0.25663837  0.004250598      3.951622    0.1305675  0.1139560
5   0.9742847    98.85485  7490.706 0.04926211  0.004244624     16.499060    0.1255381  1.9385865
6   0.9705429   109.76444  7232.967 0.05123236  0.004264460     15.759647    0.1223859  3.2308449
7   0.9570589   145.75324  6497.016 0.05348293  0.004342222     13.663780    0.1153809  3.1578010
8   0.9563996   146.35469  6420.749 0.05570796  0.004338352     13.516486    0.1123511  5.7292797
```

NbClust output : Critical values

[All.CriticalValues] Critical values of some indices for each partition obtained with a number of clusters between *min.nc* and *max.nc*.

```
$All.CriticalValues
  nc.CritValue CritValue_Duda CritValue_PseudoT2 Fvalue_Beale CritValue_Gap
2            2      0.5171722            208.1910  0.092759985    -0.6948868
3            3      0.4801976            160.2064  0.003232501    -0.8791693
4            4      0.3986176            111.6416  0.630553857     0.2082191
5            5      0.3967367            111.0011  0.802389350     0.2141258
6            6      0.3967367            111.0011  0.569234871     0.1100320
7            7      0.1779589            101.6241  0.361903692     0.1288987
8            8      0.3846056            107.2044  0.605673835     0.1262526
```

NbClust output : Best number of clusters

[Best.nc] Best number of clusters proposed by each index and the corresponding index value.

```
$Best.nc
                 index.KL index.CH index.Hartigan index.CCC index.Scott index.Marriot index.TrCovW index.TraceW
Number_clusters   4.0000    4.000         4.0000    4.0000      4.0000        4.0000       7.0000       3.0000
Value_Index      13.0963 4824.085       815.8783   31.0381    555.0816      506.3085       1.9932      38.3923

Number_clusters index.Friedman index.Rubin index.Cindex    index.DB index.Silhouette index.Duda index.PseudoT2
Value_Index             4.0000      4.0000       4.0000      2.0000           4.0000     4.0000         4.0000
                      133.8202   -106.1282       0.2755      0.3565           0.7603     0.6808        34.6947

Number_clusters index.Beale index.Ratkowsky index.Ball index.PtBiserial index.Gap index.Frey     index.McClain
Value_Index          4.0000          2.0000     3.0000           2.0000    4.0000     2.0000            2.0000
                     0.4626          0.6124    32.8865           0.8007    2.6385     1.4313            0.1873

Number_clusters index.Gamma index.Gplus index.Tau index.Dunn index.Hubert index.SDindex index.Dindex index.SDbw
Value_Index          4.0000      4.0000     3.000     2.0000            0        3.0000            0      4.000
                     0.9988      5.1441  9792.501     0.4799            0        2.9681            0      0.114
```

NbClust output : Best number of clusters

$Best.nc

	index.KL	index.CH	index.Hartigan	index.CCC	index.Scott	index.Marriot	index.TrCovW	index.TraceW
Number_clusters	4.0000	4.000	4.0000	4.0000	4.0000	4.0000	7.0000	3.0000
Value_Index	13.0963	4824.085	815.8783	31.0381	555.0816	506.3085	1.9932	38.3923

	index.Friedman	index.Rubin	index.Cindex	index.DB	index.Silhouette	index.Duda	index.PseudoT2
Number_clusters	4.0000	4.0000	4.0000	2.0000	4.0000	4.0000	4.0000
Value_Index	133.8202	-106.1282	0.2755	0.3565	0.7603	0.6808	34.6947

	index.Beale	index.Ratkowsky	index.Ball	index.PtBiserial	index.Gap	index.Frey	index.McClain
Number_clusters	4.0000	2.0000	3.0000	2.0000	4.0000	2.0000	2.0000
Value_Index	0.4626	0.6124	32.8865	0.8007	2.6385	1.4313	0.1873

	index.Gamma	index.Gplus	index.Tau	index.Dunn	index.Hubert	index.SDindex	index.Dindex	index.SDbw
Number_clusters	4.0000	4.0000	3.000	2.0000	0	3.0000	0	4.000
Value_Index	0.9988	5.1441	9792.501	0.4799	0	2.9681	0	0.114

17/28

NbClust output : Hubert index and Dindex

$Best.nc

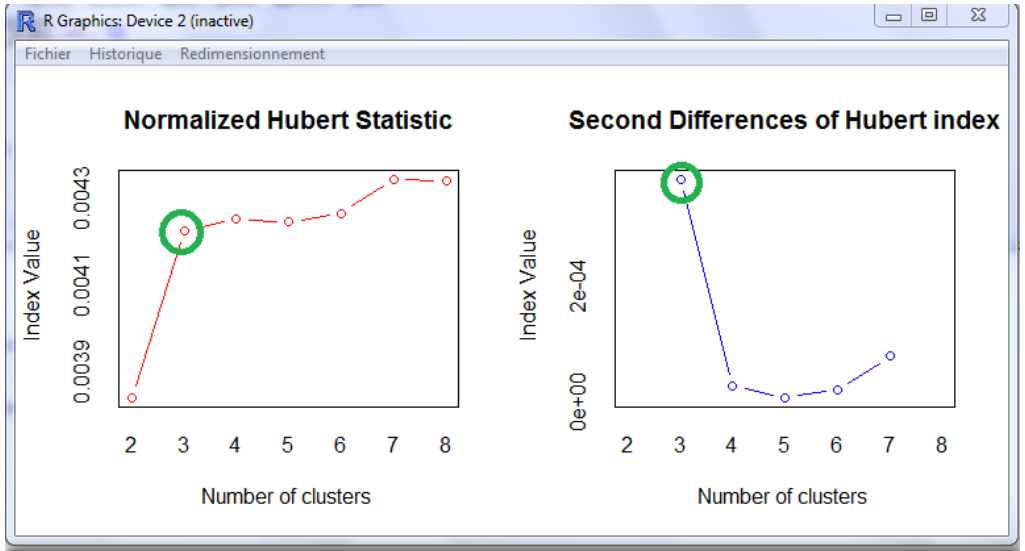
	index.KL	index.CH	index.Hartigan	index.CCC	index.Scott	index.Marriot	index.TrCovW	index.TraceW
Number_clusters	4.0000	4.000	4.0000	4.0000	4.0000	4.0000	7.0000	3.0000
Value_Index	13.0963	4824.085	815.8783	31.0381	555.0816	506.3085	1.9932	38.3923

	index.Friedman	index.Rubin	index.Cindex	index.DB	index.Silhouette	index.Duda	index.PseudoT2
Number_clusters	4.0000	4.0000	4.0000	2.0000	4.0000	4.0000	4.0000
Value_Index	133.8202	-106.1282	0.2755	0.3565	0.7603	0.6808	34.6947

	index.Beale	index.Ratkowsky	index.Ball	index.PtBiserial	index.Gap	index.Frey	index.McClain
Number_clusters	4.0000	2.0000	3.0000	2.0000	4.0000	2.0000	2.0000
Value_Index	0.4626	0.6124	32.8865	0.8007	2.6385	1.4313	0.1873

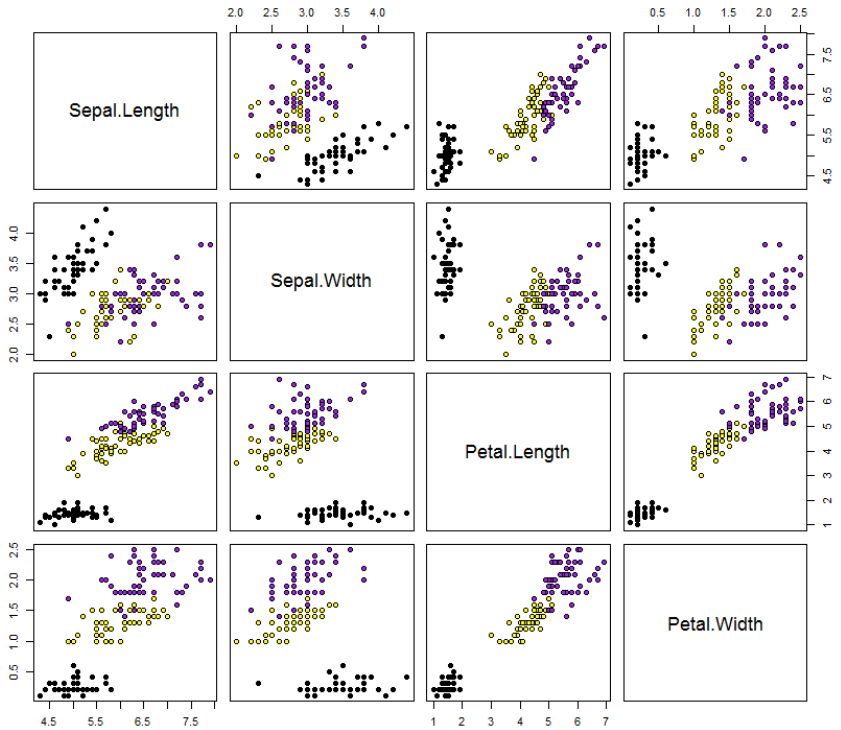
	index.Gamma	index.Gplus	index.Tau	index.Dunn	index.Hubert	index.SDindex	index.Dindex	index.SDbw
Number_clusters	4.0000	4.0000	3.000	2.0000	0	3.0000	0	4.000
Value_Index	0.9988	5.1441	9792.501	0.4799	0	2.9681	0	0.114

NbClust output : Hubert index and Dindex



Example2 : Iris dataset (Fisher 1936)

Iris dataset is composed of 3 species : "Setosa", "Virginica" and "Versicolor"



Clustering of Iris dataset (1)

```
> data<-iris[,-c(5)]
> NbClust(data, diss="NULL", distance="euclidean", min.nc=2, max.nc=8, method="complete",
+         index="alllong", alphaBeale = 0.1)
[1] "All 150 observations were used."
$All.index
  nc.Ward    index.KL index.CH index.Hartigan index.CCC index.Scott index.Marriot index.TrCovW index.TraceW index.Friedman index.Rubin
2       2  1.96520688 280.8392      240.747826  30.44413     933.9084       977604.0    6868.54015     235.15306       715.2826     40.5663
3       3  5.35979604 485.9050       68.836295  35.86679    1210.7629       347351.8     304.17913      89.52501       804.1705    106.5545
4       4 54.03770717 495.1816       16.416748  35.60363    1346.7582       249402.3     135.74320      60.97295       955.5312    156.4512
5       5  0.02629515 414.3925       51.137078  33.06976    1387.9419       296129.2     121.50441      54.80993       991.9852    174.0431
6       6  7.16532289 455.4931       16.807553  33.98704    1506.5585       193380.9      96.99085      40.51983      1070.1736    235.4228
7       7  0.53083307 423.7198       20.295977  32.90627    1560.0089       184311.4      93.20045      36.28471      1171.9307    262.9011
8       8  2.40710655 414.7146        4.465337  32.48725    1628.7974       152185.5      60.93930      31.77490      1251.1704    300.2146
  index.Cindex  index.DB index.Silhouette index.Duda index.Pseudot2 index.Beale index.Ratkowsky index.Ball index.ptbiserial index.Gap
2    0.3722945 0.7027251        0.5159830  0.1460185      444.48211   13.936036       0.4728782 117.576528        0.6368838 0.6277765
3    0.3162886 0.7024549        0.5135953  0.5581868       55.40605    1.883972       0.4921923  29.841669        0.7203480 1.4185722
4    0.3465136 0.7289093        0.4998128  0.5932269       32.91340    1.621632       0.4386587  15.243238        0.6947522 1.6397339
5    0.3758156 0.9837589        0.3461740  0.5451570       48.39138    1.980121       0.4025504  10.961987        0.6072829 1.6245746
6    0.4031740 1.0523666        0.3382031  0.5655970       19.96913    1.785546       0.3737888   6.753304        0.5295023 1.8082840
7    0.3982187 1.0030440        0.3297649  0.6480041       19.55520    1.275958       0.3481775   5.183530        0.5211682 1.8374555
8    0.4118430 1.0738058        0.3240250  2.1862561      -11.93713   -1.252991       0.3275245   3.971862        0.4752821 1.9367040
   index.Frey index.McClain index.Gamma index.Gplus index.Tau index.Dunn index.Hubert index.SDindex index.Dindex index.SDbw
2   0.2674701     0.4228305   0.7471823   353.10899  2475.495 0.08240221  0.001525609      1.788563    1.1446419 0.89762762
3   0.8589346     0.4964345   0.8928129   139.92841  2649.841 0.10329208  0.001964297      1.610001    0.6722057 0.66355981
4 134.6913164     0.5734492   0.9261200    87.93423  2495.851 0.13654328  0.002156494      1.901151    0.5832114 0.81698159
5   1.1447822     0.7935639   0.8588780   149.09512  2206.153 0.10000000  0.002242624      3.452041    0.5513315 7.25547666
6   0.6883484     1.0742029   0.8919052    88.52519  1728.103 0.13108063  0.002328929      3.527890    0.4777923 6.71261138
7   1.2624045     1.1036575   0.9020268    77.17181  1664.993 0.13459548  0.002334359      3.605433    0.4529604 6.89511041
8   0.5934017     1.3191402   0.9114965    58.77808  1384.061 0.15285446  0.002342696      3.905173    0.4238946 0.03570134

$All.CriticalValues
  nc.CritValue CritValue_Duda CritValue_PseudoT2 Fvalue_Beale CritValue_Gap
2            2      0.6120671           48.16940 1.887317e-10   -0.72358300
3            3      0.6027254           46.13913 1.133647e-01   -0.17988855
4            4      0.5551014           38.47068 1.703935e-01    0.07072484
5            5      0.5799980           42.00034 9.833288e-02   -0.12784469
6            6      0.4590041           30.64438 1.373030e-01   -0.01378387
7            7      0.5130746           34.16523 2.821982e-01   -0.03020929
8            8      0.4284097           29.35272 1.000000e+00    0.07035865
```

Clustering of Iris dataset (2)

$Best.nc

	index.KL	index.CH	index.Hartigan	index.CCC	index.Scott	index.Marriot	index.TrCovW	index.TraceW	index.Friedman
Number_clusters	4.0000	4.0000	3.0000	3.0000	3.0000	3.0	3.000	3.000	4.0000
Value_Index	54.0377	495.1816	171.9115	35.8668	276.8545	532302.7	6564.361	117.076	151.3607

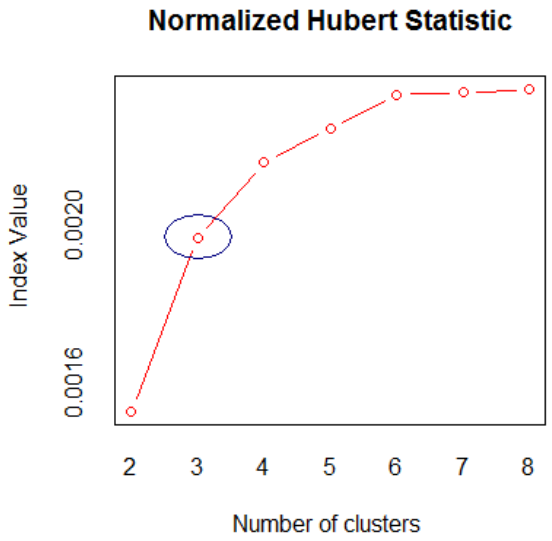
	index.Rubin	index.Cindex	index.DB	index.Silhouette	index.Duda	index.PseudoT2	index.Beale	index.Ratkowsky	index.Ball
Number_clusters	6.0000	3.0000	3.0000	2.000	4.0000	4.0000	3.000	3.0000	3.0000
Value_Index	-33.9014	0.3163	0.7025	0.516	0.5932	32.9134	1.884	0.4922	87.7349

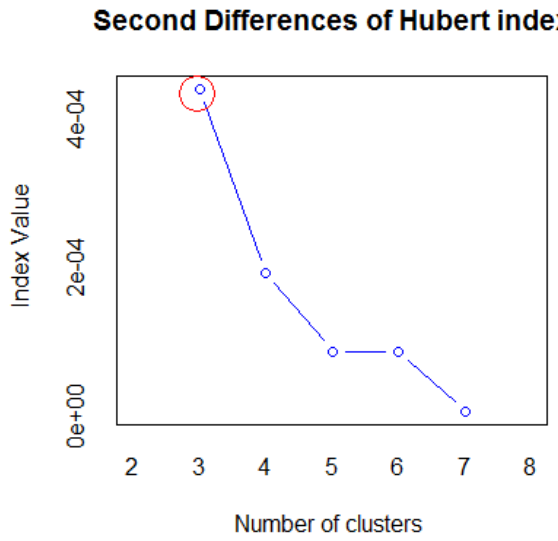
	index.PtBiserial	index.Gap	index.Frey	index.McClain	index.Gamma	index.Gplus	index.Tau	index.Dunn	index.Hubert
Number_clusters	3.0000	4.0000	2.0000	2.0000	4.0000	8.0000	3.00	8.0000	0
Value_Index	0.7203	1.6397	0.2675	0.4228	0.9261	58.7781	2649.84	0.1529	0

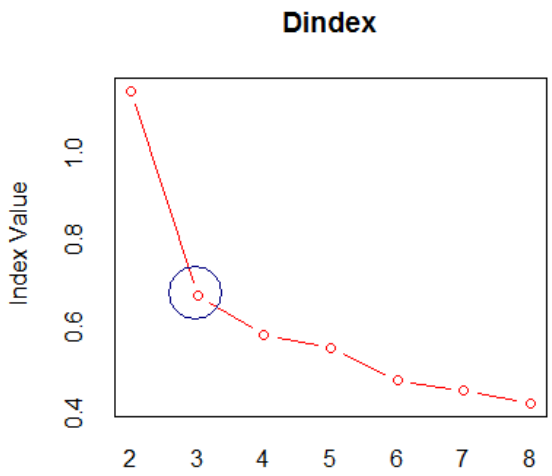
	index.SDindex	index.Dindex	index.SDbw
Number_clusters	3.00	0	8.0000
Value_Index	1.61	0	0.0357

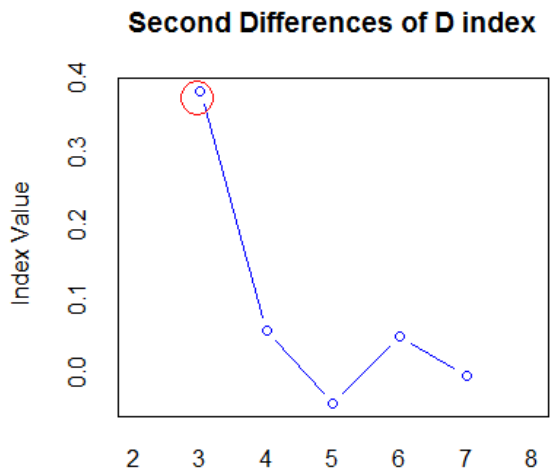
14/28

Clustering of Iris dataset (3)









How to decide on the correct number of clusters ?

$Best.nc

	index.KL	index.CH	index.Hartigan	index.CCC	index.Scott	index.Marriot	index.TrCovW	index.TraceW
Number_clusters	4.0000	4.000	4.0000	4.0000	4.0000	4.0000	7.0000	3.0000
Value_Index	13.0963	4824.085	815.8783	31.0381	555.0816	506.3085	1.9932	38.3923

	index.Friedman	index.Rubin	index.Cindex	index.DB	index.Silhouette	index.Duda	index.PseudoT2
Number_clusters	4.0000	4.0000	4.0000	2.0000	4.0000	4.0000	4.0000
Value_Index	133.8202	-106.1282	0.2755	0.3565	0.7603	0.6808	34.6947

	index.Beale	index.Ratkowsky	index.Ball	index.PtBiserial	index.Gap	index.Frey	index.McClain
Number_clusters	4.0000	2.0000	3.0000	2.0000	4.0000	2.0000	2.0000
Value_Index	0.4626	0.6124	32.8865	0.8007	2.6385	1.4313	0.1873

	index.Gamma	index.Gplus	index.Tau	index.Dunn	index.Hubert	index.SDindex	index.Dindex	index.SDbw
Number_clusters	4.0000	4.0000	3.000	2.0000	0	3.0000	0	4.000
Value_Index	0.9988	5.1441	9792.501	0.4799	0	2.9681	0	0.114

How to decide on the correct number of clusters ?

1. Majority rule : User can select the number of clusters proposed by the majority of indices. ex : 4 in 1st example and 3 in 2nd example.

2. User can consider only indices that performed best in simulations studies. Top-5 indices in Milligan and Cooper study are : CH index, Duda index, Cindex, Gamma and Beale index.

Conclusion

- **NbClust** package provides a large list of indices, many of them are not implemented anywhere. The current version contains up to 30 indices.
- **NbClust** package permits the user to simultaneously vary the number of clusters, the clustering method and the indices to decide how best to group observations in his dataset or to compare all indices or clustering methods.
- **NbClust** package is available at http ://cran.r-project.org/web/packages/NbClust/index.html

Thank you !

Questions

malika.charrad.1@ulaval.ca