UCL

Cluster validation: how to think and what to do

Christian Hennig

1. Introduction
   - A short introduction to clustering methods
   - Introduction to cluster validation

2. Approaches for cluster validation
   - Use of external information
   - Visual Exploration
   - Stability assessment
   - Internal validation indexes
   - Testing for clustering structure
   - Sensitivity analysis and comparing different clusterings

3. Discussion

1. Introduction

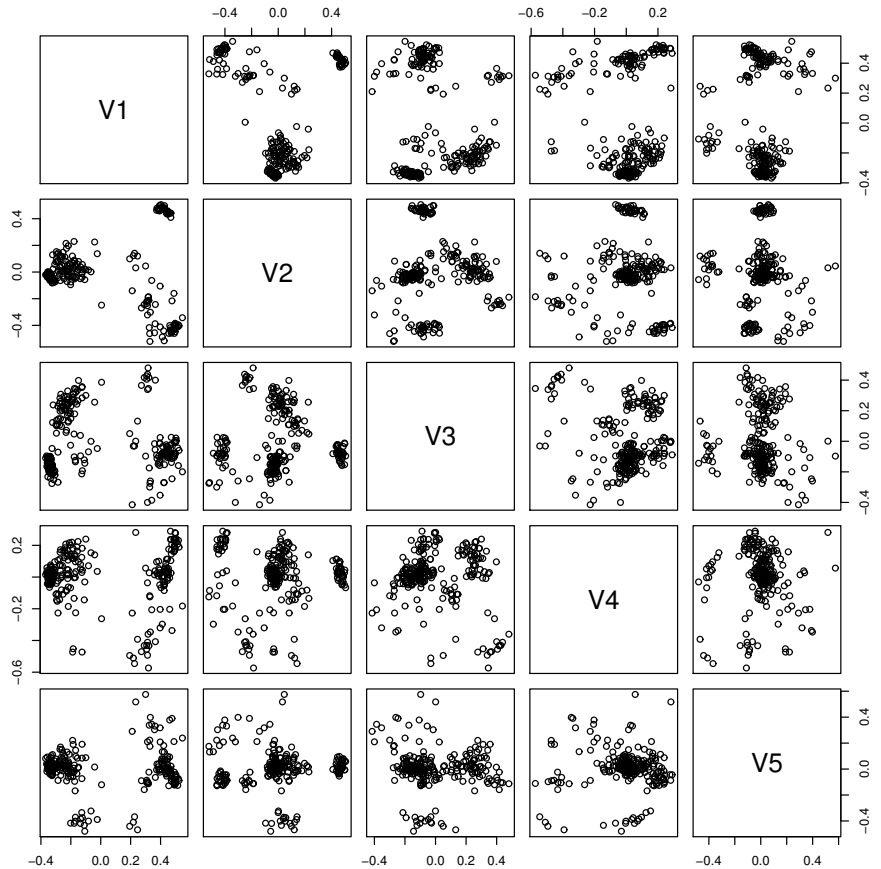
Cluster analysis: finding groups in data.

There are many cluster analysis methods,
and on many datasets these may produce
many different clusterings.

Cluster validation: clustering quality assessment,
either assessing a single clustering,
or comparing different clusterings
(i.e., with different numbers of clusters
for finding a best one).

1.1 A short introduction to clustering methods

Cluster analysis is about finding groups in data.



1.1.1 **k-means** (Steinhaus (1956))

$$\sum_{i=1}^{n} \|\mathbf{x}_i - \bar{\mathbf{x}}_{C(i)}\|^2 = \min!$$

1.1.1 **k-means** (Steinhaus (1956))

$$\sum_{i=1}^{n} \|\mathbf{x}_i - \bar{\mathbf{x}}_{C(i)}\|^2 = \min!$$

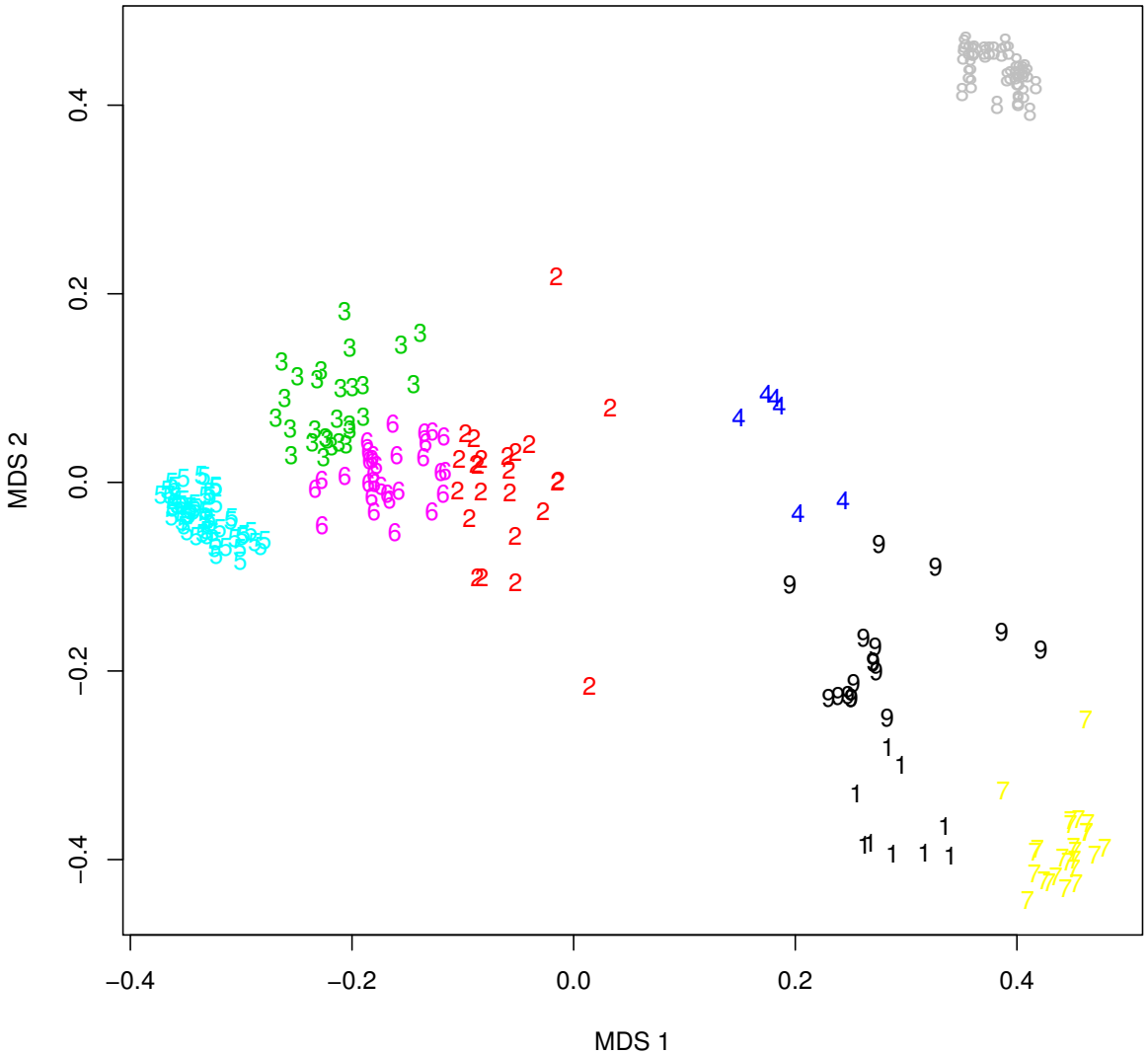
represents all objects by centroid,
"compact" clusters.

1.1.1 **k-means** (Steinhaus (1956))

$$\sum_{i=1}^{n} \|\mathbf{x}_i - \bar{\mathbf{x}}_{C(i)}\|^2 = \min!$$

represents all objects by centroid,
"compact" clusters.

Version: Don't square, other centroids than mean ("pam").

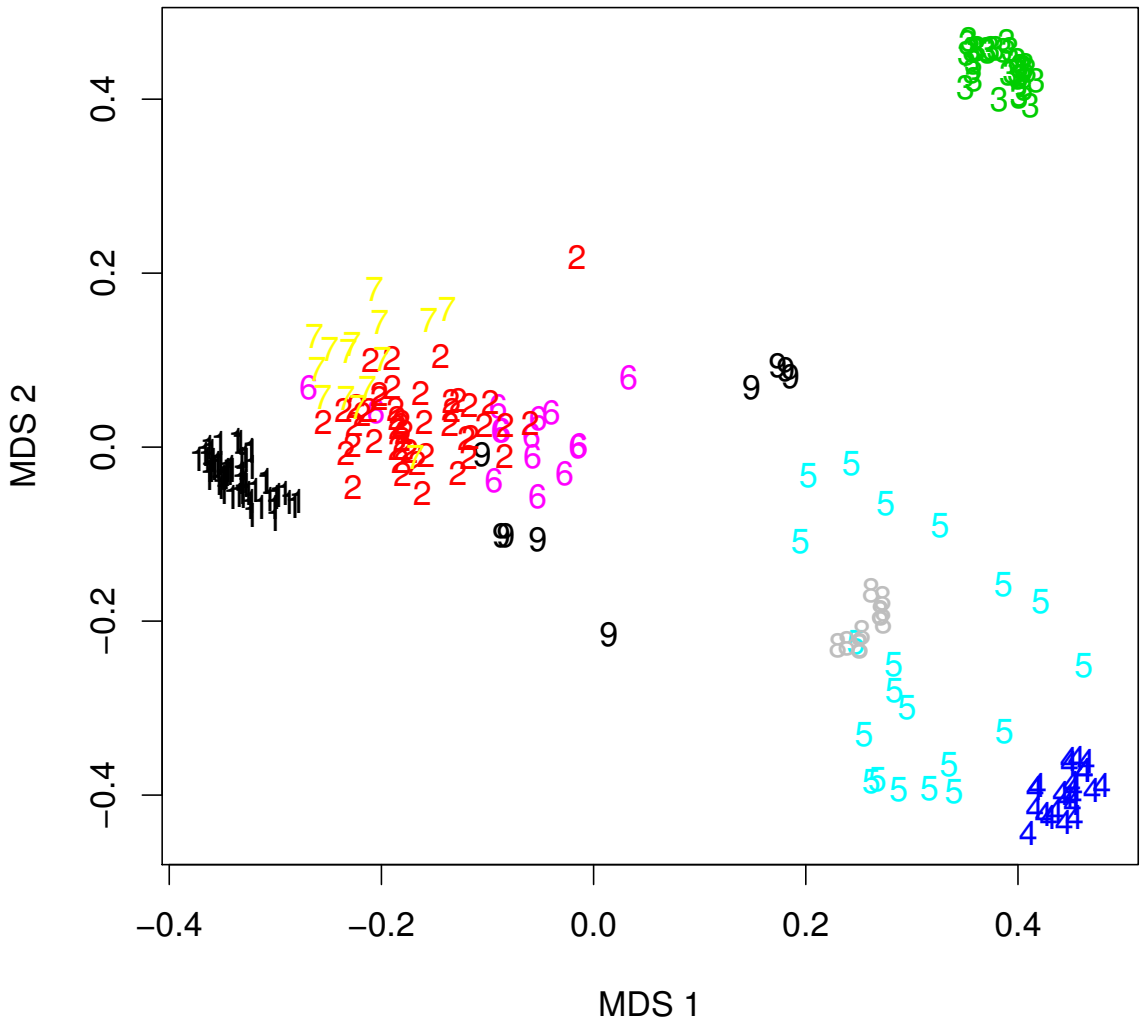

MDS 2
0.4
0.2
0.0
−0.2
−0.4
−0.4
−0.2
0.0
0.2
0.4
MDS 1

1.1.2 **Gaussian mixture model** (Pearson (1894))

$$f(\mathbf{x}) = \sum_{j=1}^{k} \pi_j \varphi_{\mathbf{a}_j, \Sigma_j}(\mathbf{x}).$$

Clusters are described by Gaussian distributions.
Elliptical clusters, flexible size and shape.

Mixtures of other distribution families exist, too.


MDS 2
0.4
0.2
0.0
−0.2
−0.4
−0.4
−0.2
0.0
0.2
0.4
MDS 1

1.1.3 Classical hierarchical methods

Operate on *dissimilarity matrices*;
compute dissimilarity measure for every pair of observations.

Can use Euclidean distance,
but also tailor-made distances for other data formats.

1.1.3 Classical hierarchical methods

Operate on *dissimilarity matrices*;
compute dissimilarity measure for every pair of observations.

Can use Euclidean distance,
but also tailor-made distances for other data formats.

"Cluster": a collection of similar objects,
dissimilar to the others.

Genetic data: 236 Tetragonula bees, 13 allele pairs

```
      [,1] [,2] [,3] [,4] [,5] [,6]  (...)
 [1,] "NO" "AA" "PP" "HH" "EH" "FF"
 [2,] "EO" "AA" "PP" "HH" "GH" "FF"
 [3,] "NQ" "AA" "PT" "HH" "GF" "EF"
 [4,] "OO" "AA" "PP" "GH" "GH" "EF"
 [5,] "OO" "AA" "PP" "GH" "GH" "EF"
 [6,] "LN" "AA" "PP" "HH" "EG" "FE"
(...)
```

Compute "shared allele distance".

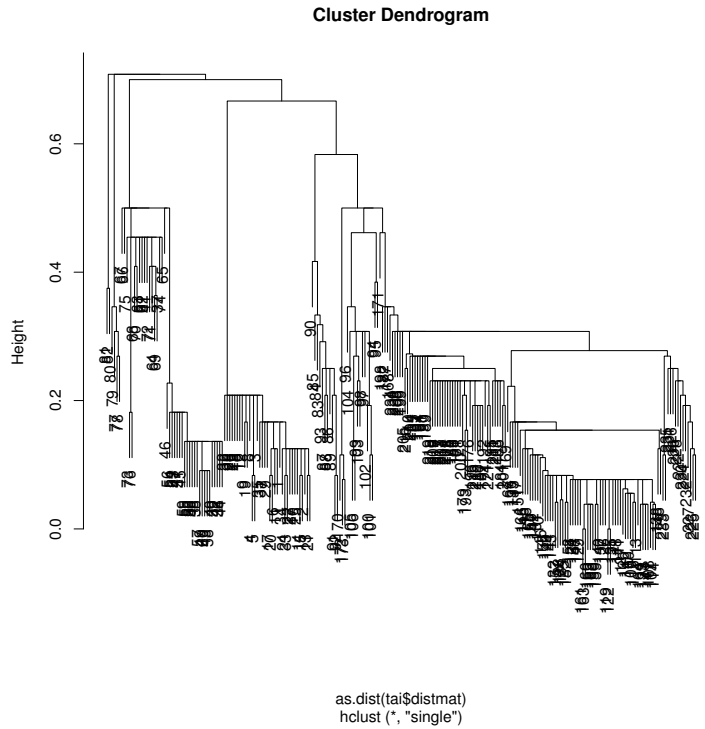
```
     [,1] [,2] [,3] [,4] [,5]
[1,] 0.00 0.21 0.33 0.29 0.25
[2,] 0.21 0.00 0.33 0.25 0.21
[3,] 0.33 0.33 0.00 0.29 0.33 (...)
[4,] 0.29 0.25 0.29 0.00 0.08
[5,] 0.25 0.21 0.33 0.08 0.00
(...)
```

Dataset seen before is a

Euclidean approximation ("MDS") of this.

1.1.3 Classical hierarchical methods

Operate on dissimilarities and produce hierarchical trees (originally motivated by biological classification).
Differ in definition of "dissimilarity between clusters".



Single Linkage: (Florek et al. (1951))

$$\tilde{d}(A, B) = \min_{a \in A, b \in B} d(a, b)$$

Complete Linkage:

$$\tilde{d}(A, B) = \max_{a \in A, b \in B} d(a, b)$$

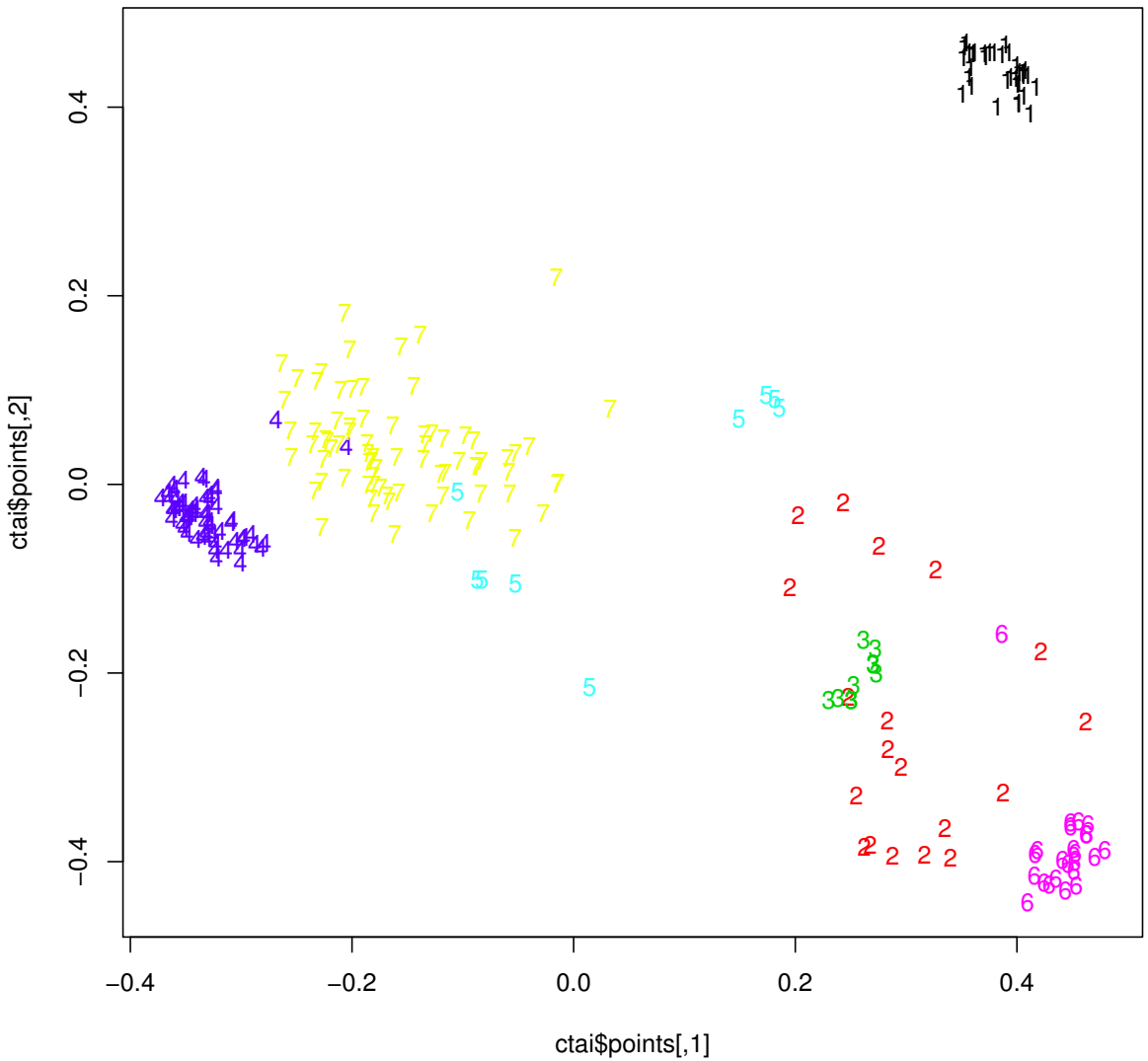
Average Linkage:

$$\tilde{d}(A, B) = \text{ave}_{a \in A, b \in B} d(a, b)$$

These can deliver quite different clusterings.
(Complete L. very compact,
Single L. separated but maybe widespread)

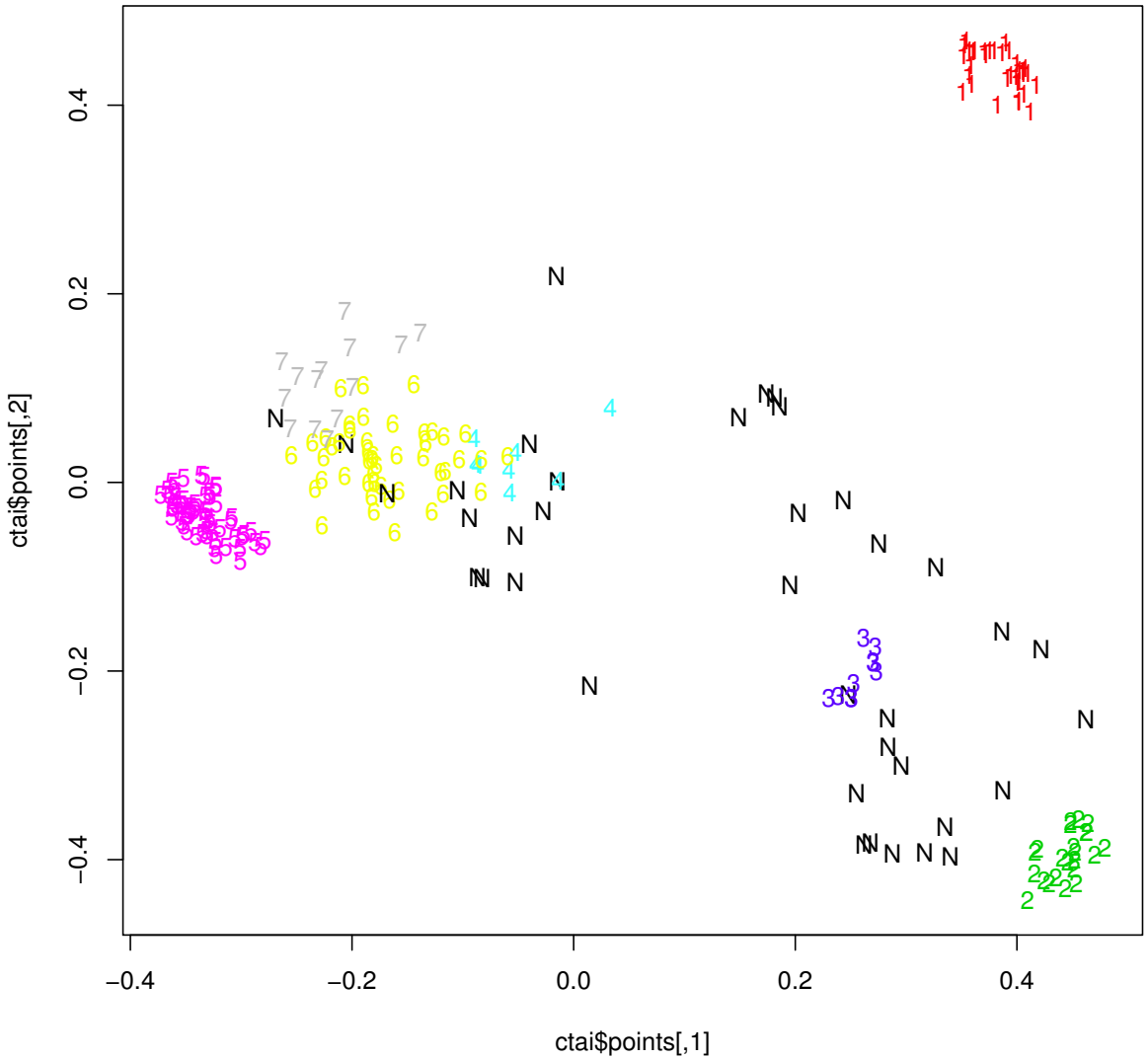
1.1.4 **Spectral clustering** (Shi and Malik (2000))

Dissimilarity-based nonlinear dimension reduction
for k-means.


ctai$points[,2]
0.4
0.2
0.0
−0.2
−0.4
−0.4
−0.2
0.0
0.2
0.4
ctai$points[,1]

1.1.5 Density-based methods

such as “DBSCAN” (Ester et al. (1996)),
joins observations with all neighbouring points,
and neighbourhoods if they share enough points.


ctai$points[,2]
0.4
0.2
0.0
−0.2
−0.4
−0.4
−0.2
0.0
0.2
0.4
ctai$points[,1]

1.1.6 Other approaches

- Models and approaches for data with specific structure, time series, categorical, spatial, and text data,...
- Overlapping and fuzzy clustering
- "Self-organising" algorithmic methods
- Semi-supervised and constrained clustering
- Multi-mode clustering (e.g., observations and variables)
- Big data algorithms, grids, dimension reduction,...

1.2 Introduction to cluster validation

Generally concerned with
evaluating the quality of clusterings.

Of interest for...

- Assessment of reliability/quality of given clustering
- Choice of clustering method
- Number of clusters (and other parameters)

1.2 Introduction to cluster validation

Generally concerned with
evaluating the quality of clusterings.

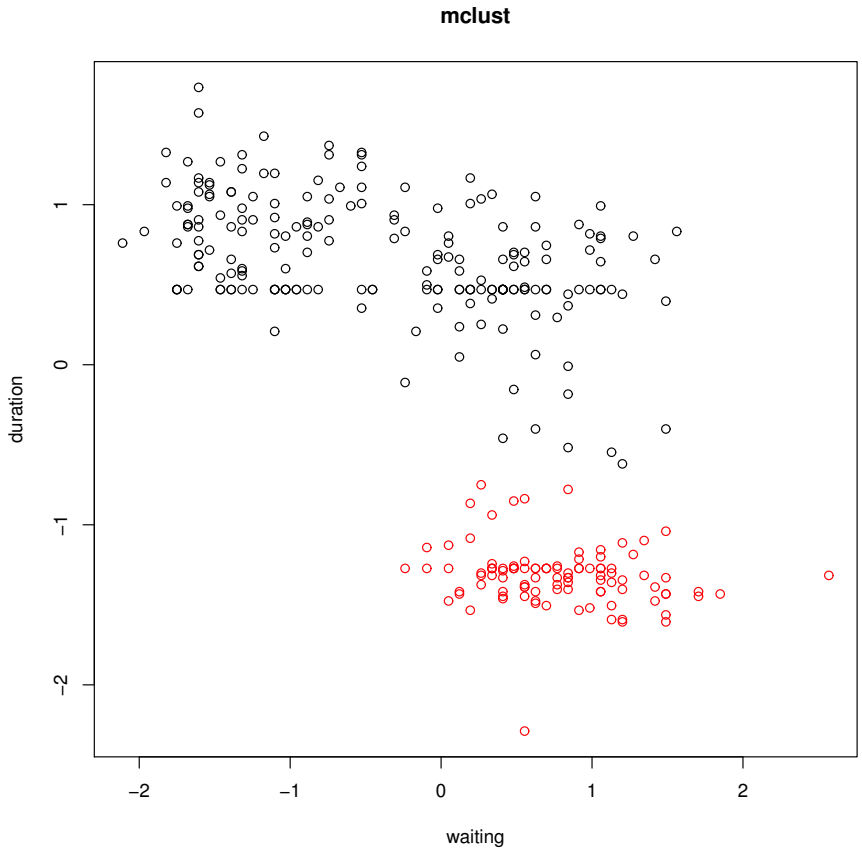
Of interest for...

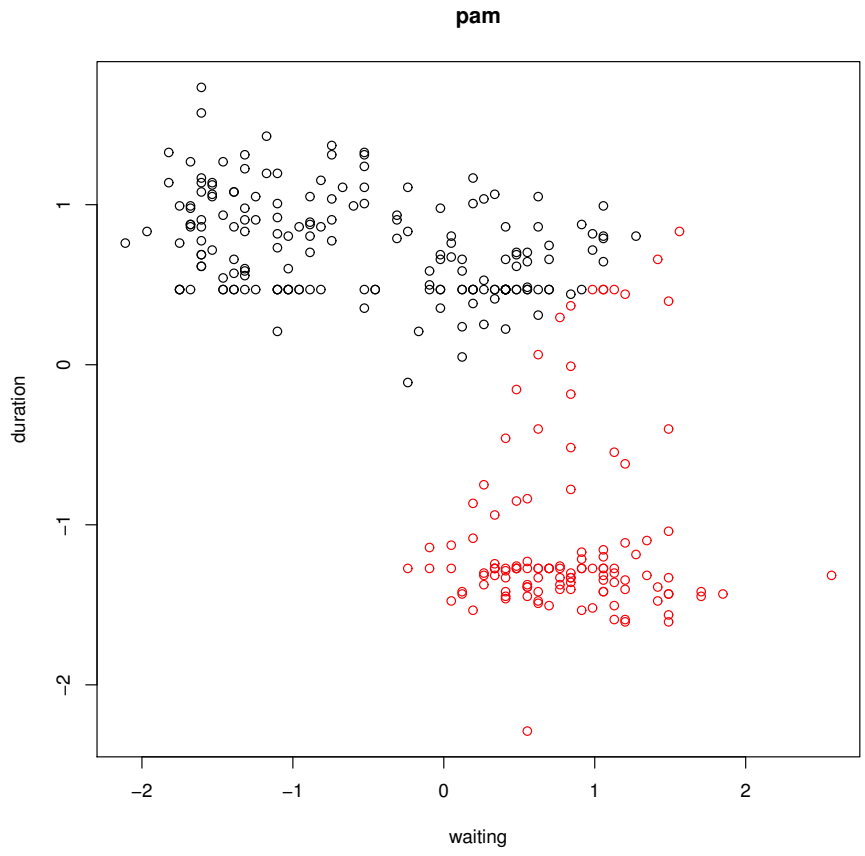
- Assessment of reliability/quality of given clustering
- Choice of clustering method
- Number of clusters (and other parameters)

In literature sometimes "cluster validation"
inappropriately refers to "number of clusters" exclusively.

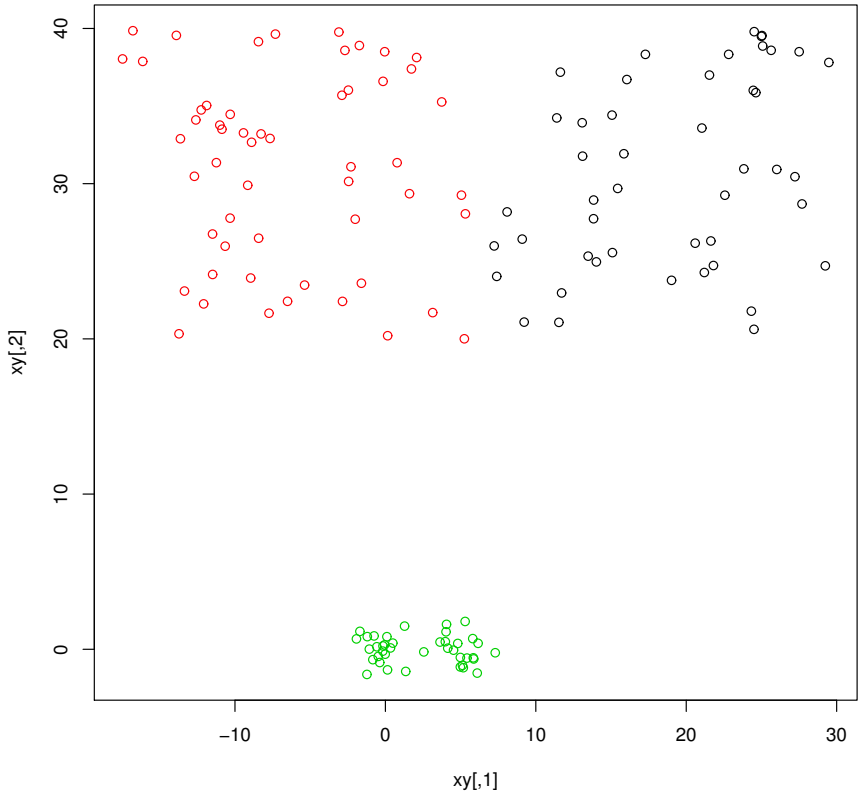
1.2.1 Which clustering is better?

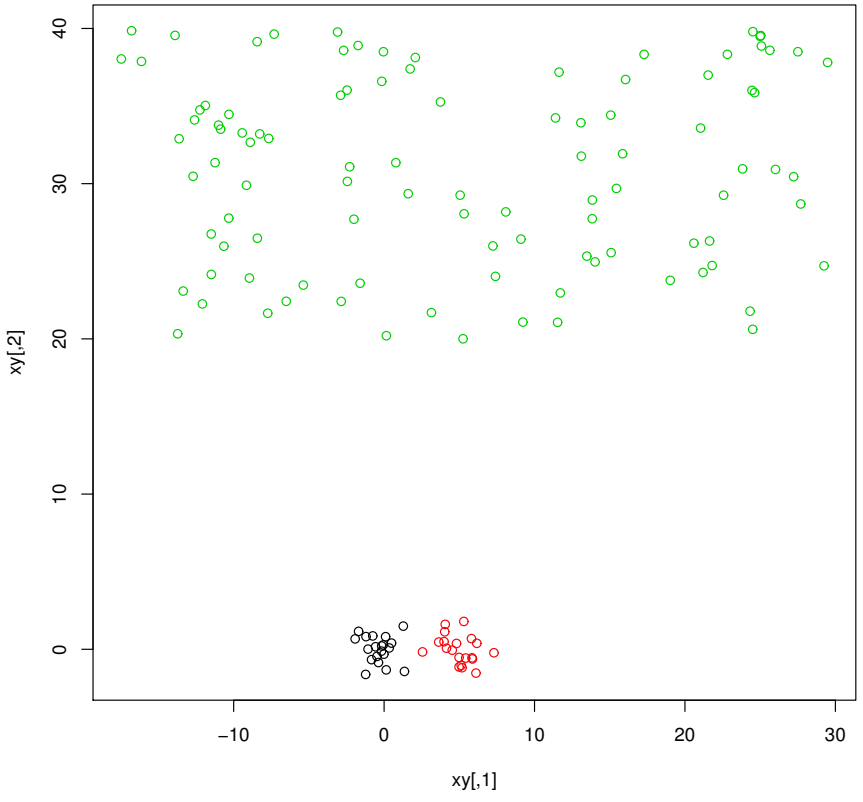
(Old faithful geyser data)





Which clustering is better?





. . . this depends on the aim of clustering.

Different aims of clustering
require different characteristics of a clustering,
and these can be conflicting.

E.g., Single Linkage emphasises between-cluster separation
at the expense of within-cluster homogeneity.

E.g., pattern recognition in images
requires separation,

E.g., pattern recognition in images
requires separation,

clustering for information reduction requires
good representation by centroids,

E.g., pattern recognition in images
requires separation,

clustering for information reduction requires
good representation by centroids,

groups in social network analysis shouldn't have
large within-cluster gaps,

E.g., pattern recognition in images
requires separation,

clustering for information reduction requires
good representation by centroids,

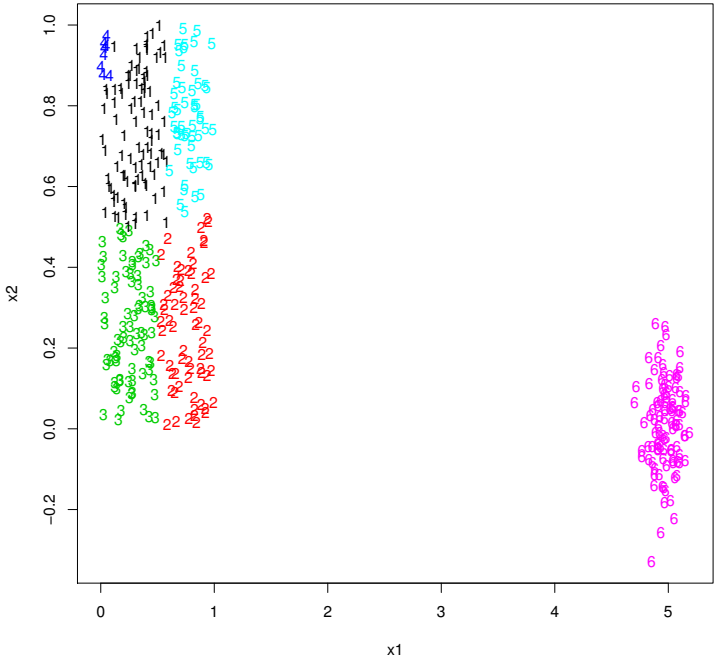
groups in social network analysis shouldn't have
large within-cluster gaps,

underlying "true" classes (biological species)
may lead to homogeneous distributional shapes.

First principle to choose a clustering method:
"Know what you need and know what the methods do."

First principle to choose a clustering method:
"Know what you need and know what the methods do."

Number of clusters:
"What granularity is needed?"



User needs to make decisions about
what qualifies a data subset to be a "cluster".
The data alone won't tell us.

User needs to make decisions about
what qualifies a data subset to be a "cluster".
The data alone won't tell us.

This should influence...

- data preprocessing decisions such as variable transformation and aggregation, dissimilarity definition etc.,
- choice of the clustering method,
- approach for cluster validation.

Clustering method and validation approach. . .
are both guided by required cluster concept,
so should suit one another;

but often clustering aims are multiple or unclear,
and clustering method focuses on narrow criterion,
so validation can complement clustering method.

1.2.2 Approaches for cluster validation

- Use of external information

1.2.2 Approaches for cluster validation

- Use of external information
- Visual exploration

1.2.2 Approaches for cluster validation

- Use of external information
- Visual exploration
- Stability assessment

1.2.2 Approaches for cluster validation

- Use of external information
- Visual exploration
- Stability assessment
- Internal validation indexes

1.2.2 Approaches for cluster validation

- Use of external information
- Visual exploration
- Stability assessment
- Internal validation indexes
- Testing for clustering structure

1.2.2 Approaches for cluster validation

- Use of external information
- Visual exploration
- Stability assessment
- Internal validation indexes
- Testing for clustering structure
- Sensitivity analysis and comparison of different clusterings on same dataset

2. Approaches for cluster validation

2.1 Use of external information

"External information" can mean various things.

- Variables that "should" be related to the clustering:
  - known "true" classification,
  - known related classification,
  - other (e.g. continuous) variables,
- External variable(s) to be explained/predicted by clustering,
- Informal expert assessment.

Known classifications

"True" classification known $\Rightarrow$
clustering not of real scientific interest.

However may be of interest for method benchmarking;
or "true" classification itself may be in some doubt
and in need of confirmation/validation.

Known classifications

"True" classification known $\Rightarrow$
clustering not of real scientific interest.

However may be of interest for method benchmarking;
or "true" classification itself may be in some doubt
and in need of confirmation/validation.

More often known classification is assumed to be
strongly related to the found clustering
(e.g., region/country for regional data).

Measure of similarity between two clusterings

Rand index (Rand (1971)): proportion of pairs of objects that are in same cluster in both clusterings.

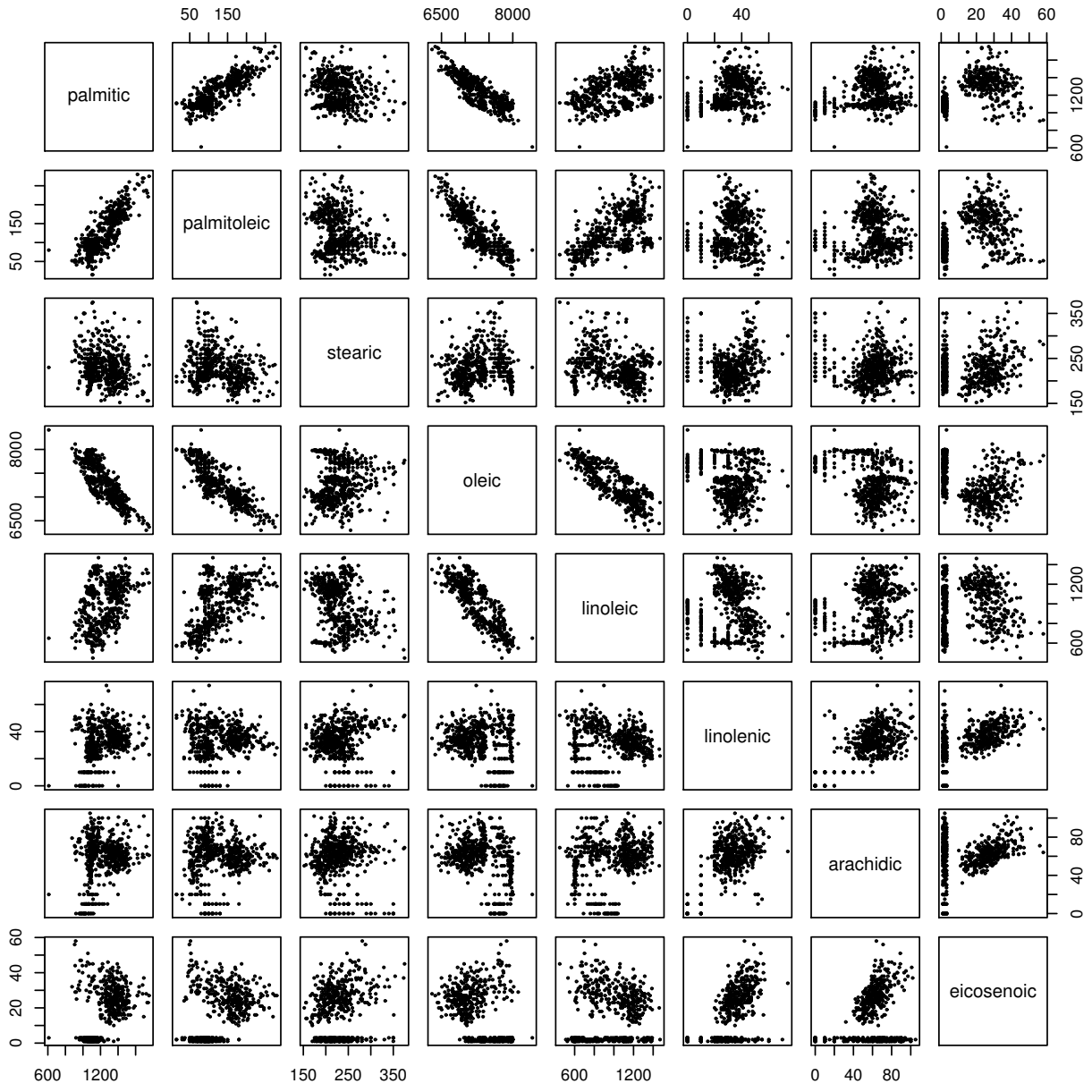
Can compare clusterings with different numbers of clusters.
Doesn't require matching of clusters.

Adjusted Rand index (ARI, Hubert and Arabie (1985)):
Standardise Rand index to $[-1, 1]$
with mean 0 for random clusterings:

$$\mathrm{ARI}(\mathcal{C}_1, \mathcal{C}_2) = \frac{\sum_{ij} \binom{n_{ij}}{2} - [\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}]/\binom{n}{2}}{\frac{1}{2}[\sum_i \binom{a_i}{2} + \sum_j \binom{b_j}{2}] - [\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}]/\binom{n}{2}},$$

i runs over $\mathbb{N}_{K_1}$, j runs over $\mathbb{N}_{K_2}$, $n_{ij} = |C_{1i} \cap C_{2j}|$, $a_i = \sum_{j=1}^{K_2} n_{ij}$ and $b_j = \sum_{i=1}^{K_1} n_{ij}$.

Olive oil data



```
library(pdfCluster)
data(oliveoil)

# Look at object:
str(oliveoil)
# 'data.frame': 572 obs. of  10 variables:
#  $ macro.area : Factor w/ 3 levels "South","Sardinia",..: 1 1 1 1 1 1 1 1 1 1 ...
#  $ region     : Factor w/ 9 levels "Apulia.north",..: 1 1 1 1 1 1 1 1 1 1 ...
#  $ palmitic   : int  1075 1088 911 966 1051 911 922 1100 1082 1037 ...
#  $ palmitoleic: int  75 73 54 57 67 49 66 61 60 55 ...
#  $ stearic    : int  226 224 246 240 259 268 264 235 239 213 ...
#  $ oleic      : int  7823 7709 8113 7952 7771 7924 7990 7728 7745 7944 ...
#  $ linoleic   : int  672 781 549 619 672 678 618 734 709 633 ...
#  $ linolenic  : int  36 31 31 50 50 51 49 39 46 26 ...
#  $ arachidic  : int  60 61 63 78 80 70 56 64 83 52 ...
#  $ eicosenoic : int  29 29 29 35 46 44 29 35 33 30 ...

olive <- oliveoil[,3:10] # Variables for clustering
```

```
library(mclust) # This has the adjustedRandIndex command.

solive <- scale(olive)
olive3 <- kmeans(olive,3,nstart=100)
olive3s <- kmeans(solive,3,nstart=100)
# k-means depends on the scale of the variables;
# scaling should help clustering

adjustedRandIndex(olive3$cluster,olive3s$cluster)
# 0.4587804, these are somewhat different.

adjustedRandIndex(olive3$cluster,oliveoil$macro.area)
# 0.3182057

adjustedRandIndex(olive3s$cluster,oliveoil$macro.area)
# 0.448355, both OK but not great, with scaling clearly better
```

Note: legitimate clusterings may be in data that are quite different from given classification.

External variables to be predicted/explained
by clustering

E.g., given 50 personality variables from
psychological questionnaire among other information
to explain individual's alcohol consumption *a*,
may want to reduce questionnaire information
to categorical variable with 5-10 personality types
for efficiently modelling *a*.

Clustering of personality variables can be
Assessed by measuring prediction quality of *a*.

Informal expert assessment

When doing clustering for subject matter experts,
sometimes experts object against a clustering.

Sometimes this is good information,
because clustering should have properties
(like being related to certain external variable)
that experts didn't specify in advance.

But it may also reflect expert's prejudice,
so find out the reasons and how good they are.

2.2 Visual exploration

Try to assess cluster quality by visual means.

Human intuition can spot methods' artifacts
and give clearer idea
of meanings and shortcomings of clusters.

But high dimensional data is hard to plot.

2.2.1 Multidimensional Scaling

Visualising dissimilarities in low-d Euclidean space

Aim: for $\{\mathbf{x}_1, \mathbf{x}_2, \ldots \mathbf{x}_n\}$
characterised by dissimilarity d find
$\mathbf{y}_1, \mathbf{y}_2, \ldots \mathbf{y}_n \in I\!R^p$ so that

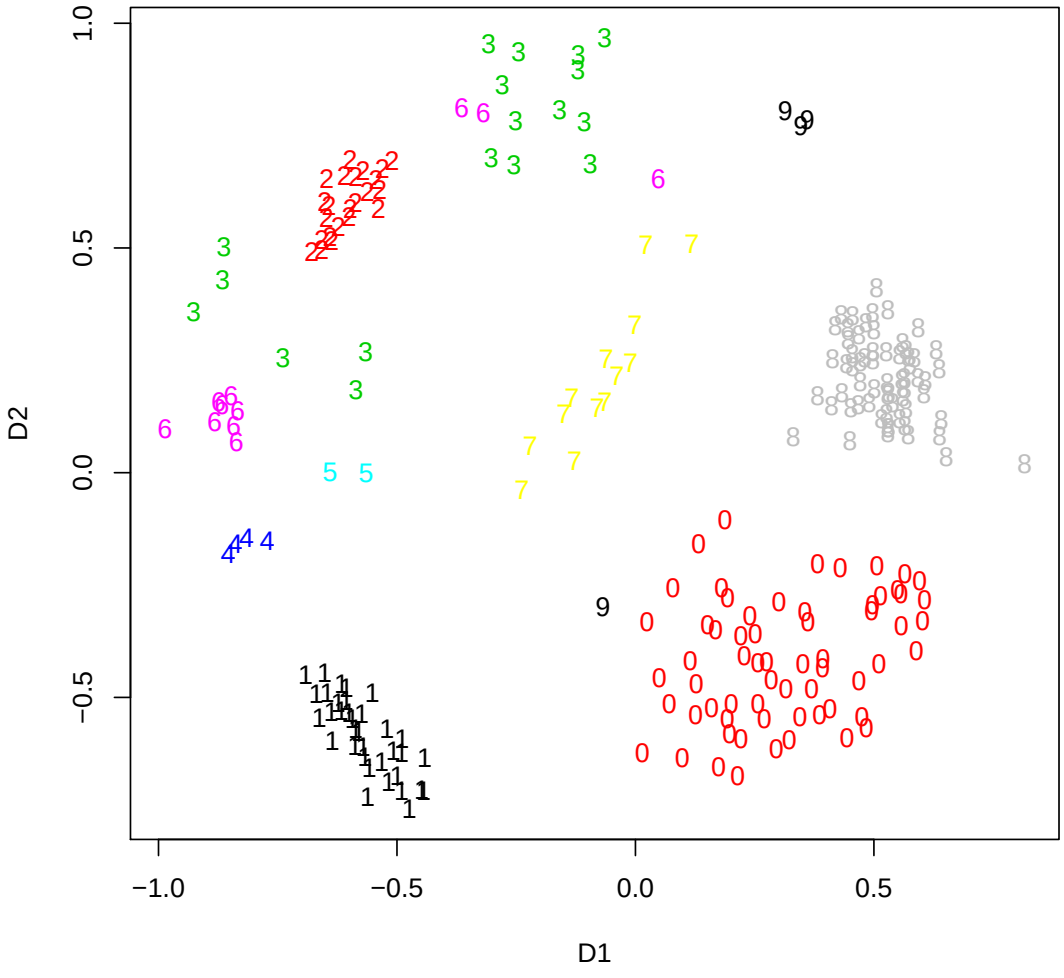
$$d(\mathbf{x}_i, \mathbf{x}_j) \approx d_{L2}(\mathbf{y}_i, \mathbf{y}_j).$$

There are several MDS methods, see Borg et al. (2012).
Ratio MDS (R-package smacof): minimise

$$\sqrt{\frac{\sum_{i<j}\left(bd(\mathbf{x}_i,\mathbf{x}_j)-d_{L2}(\mathbf{y}_i,\mathbf{y}_j)\right)^2}{n(n-1)/2}}$$

over b and $\mathbf{y}_1, \mathbf{y}_2, \ldots \mathbf{y}_n$
under $\sum_{i<j} bd(\mathbf{x}_i,\mathbf{x}_j)^2 = \frac{n(n-1)}{2}$.

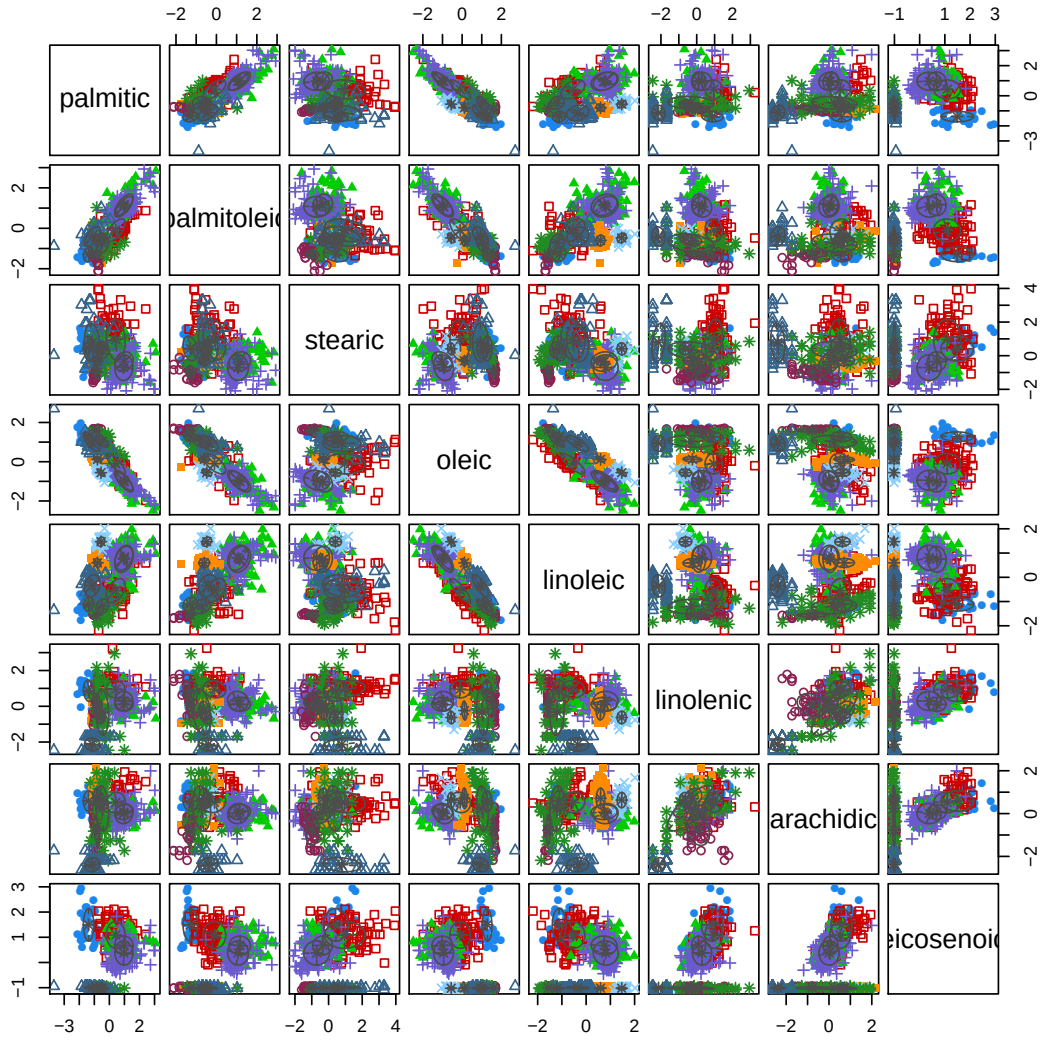
```
library(smacof) # for mds
library(prabclus) # for data
library(fpc) # provides clusym, see below
data(tetragonula)
ta <- alleleconvert(strmatrix=tetragonula)
tai <- alleleinit(allelematrix=ta)
# Tetragonula bees data as used in Sec. 1.1
# tai$distmat is dissimilarity matrix

atrigona <- hclust(as.dist(tai$distmat),method="average")
atrigona10 <- cutree(atrigona,k=10)
# Use Average Linkage clustering to split into 10 clusters

mdstrigona <- mds(tai$distmat)
plot(mdstrigona$conf,col=atrigona10,pch=clusym[atrigona10])
```


D2
D1
1.0
0.5
0.0
−0.5
−1.0
−0.5
0.0
0.5

Visualisation of Euclidean data

```
library(mclust)
molive <- Mclust(solive)
# Gaussian mixture fit for olive oil data
# Has G=9 clusters

plot(molive,what="classification")
# mclust plot method showing all dimensions
```

Looking at all dimensions is tough.



2.2.2 Projection methods

Given: $n \times p$-dataset $\mathbf{X}$.
Find $p \times s$-matrix $\mathbf{C}$ (eg, $s = 2$), so that
$\mathbf{Y} = \mathbf{XC}$ is optimally "informative".

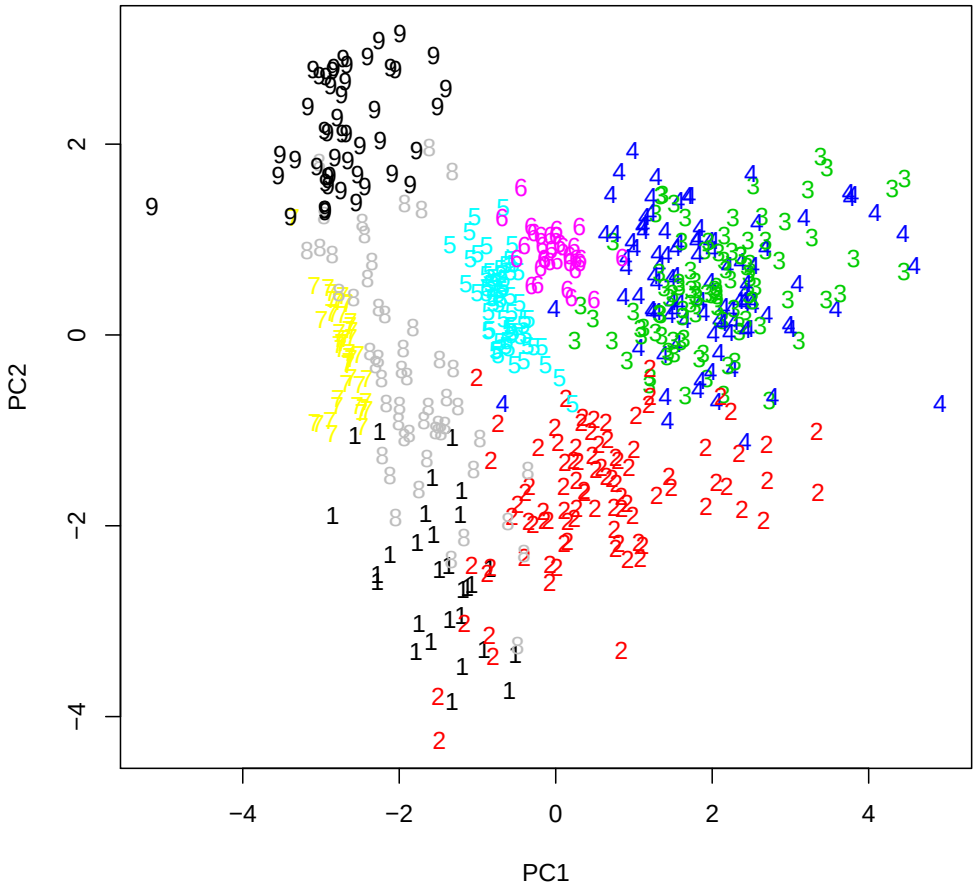
Definition. The first s projection vectors defined by the choice of $\mathbf{Q}$ and $\mathbf{R}$) $\mathbf{c}_1, \ldots \mathbf{c}_s$ are defined as the vectors maximising

$$F_{\mathbf{c}} = \frac{\mathbf{c}'\mathbf{Q}\mathbf{c}}{\mathbf{c}'\mathbf{R}\mathbf{c}}$$

subject to $\mathbf{c}_i'\mathbf{R}\mathbf{c}_j = \delta_{ij}$, where $\delta_{ij} = 1$ for $i = j$ and $\delta_{ij} = 0$ else.

Corollary. The first s projection vectors of $\mathbf{X}$ are the eigenvectors of $\mathbf{R}^{-1}\mathbf{Q}$ corresponding to the s largest eigenvalues.

Definition. PCA is defined by $\mathbf{Q} = Cov(\mathbf{X})$ and $\mathbf{R} = \mathbf{I}_p$.



PCA: "Information" = variance. Clusters ignored.

Define projection methods that optimally separate clusters.

Notation:
Let $\mathbf{x}_{i1}, \ldots, \mathbf{x}_{in_i}$ the p-dimensional points of group $i = 1, \ldots, k$, $n = \sum_{i=1}^{k} n_i$. Let $\mathbf{X}_i = (\mathbf{x}_{i1}, \ldots, \mathbf{x}_{in_i})'$, $i = 1, \ldots, k$, and $\mathbf{X} = (\mathbf{X}_1', \ldots, \mathbf{X}_k')'$. Let

$$\mathbf{m}_i = \frac{i}{n_i} \sum_{j=1}^{n_i} \mathbf{x}_{ij}, \ \mathbf{m} = \frac{1}{n} \sum_{i=1}^{k} \sum_{j=1}^{n_i} \mathbf{x}_{ij},$$
$$\mathbf{U}_i = \sum_{j=1}^{n_i} (\mathbf{x}_{ij} - \mathbf{m}_i)(\mathbf{x}_{ij} - \mathbf{m}_i)', \ \mathbf{U} = \sum_{i=1}^{s} \mathbf{U}_i,$$
$$\mathbf{S}_i = \frac{1}{n_i - 1} \mathbf{U}_i, \ \mathbf{W} = \frac{1}{n-k} \mathbf{U}, \ \mathbf{B} = \frac{1}{n(k-1)} \sum_{i=1}^{k} n_i (\mathbf{m}_i - \mathbf{m})(\mathbf{m}_i - \mathbf{m})',$$

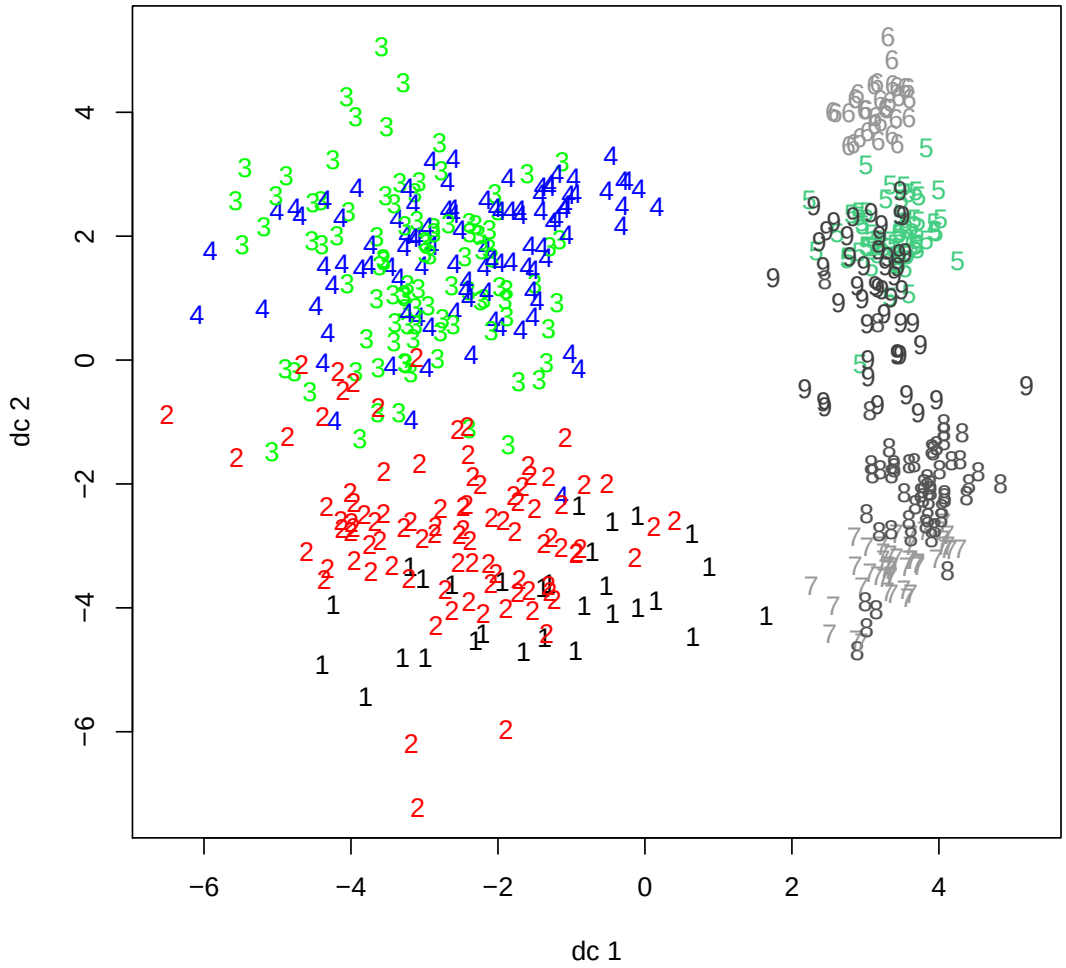
that is, $\mathbf{S}_i$ is the covariance matrix of group i with mean vector $\mathbf{m}_i$, $\mathbf{W}$ is the pooled within groups-scatter matrix and $\mathbf{B}$ is the between groups-scatter matrix.

Definition. Discriminant Coordinates (DC; Rao (1952)) are defined by $\mathbf{Q} = \mathbf{B}$ and $\mathbf{R} = \mathbf{W}$.

Corollary. Only $k - 1$ eigenvalues of $\mathbf{W}^{-1}\mathbf{B}$ are larger than 0. The whole information about the mean differences can be displayed in $k - 1$ dimensions.

Use R-function `plotcluster` in fpc.

```
library(fpc)
plotcluster(solive,molive$classification)
```

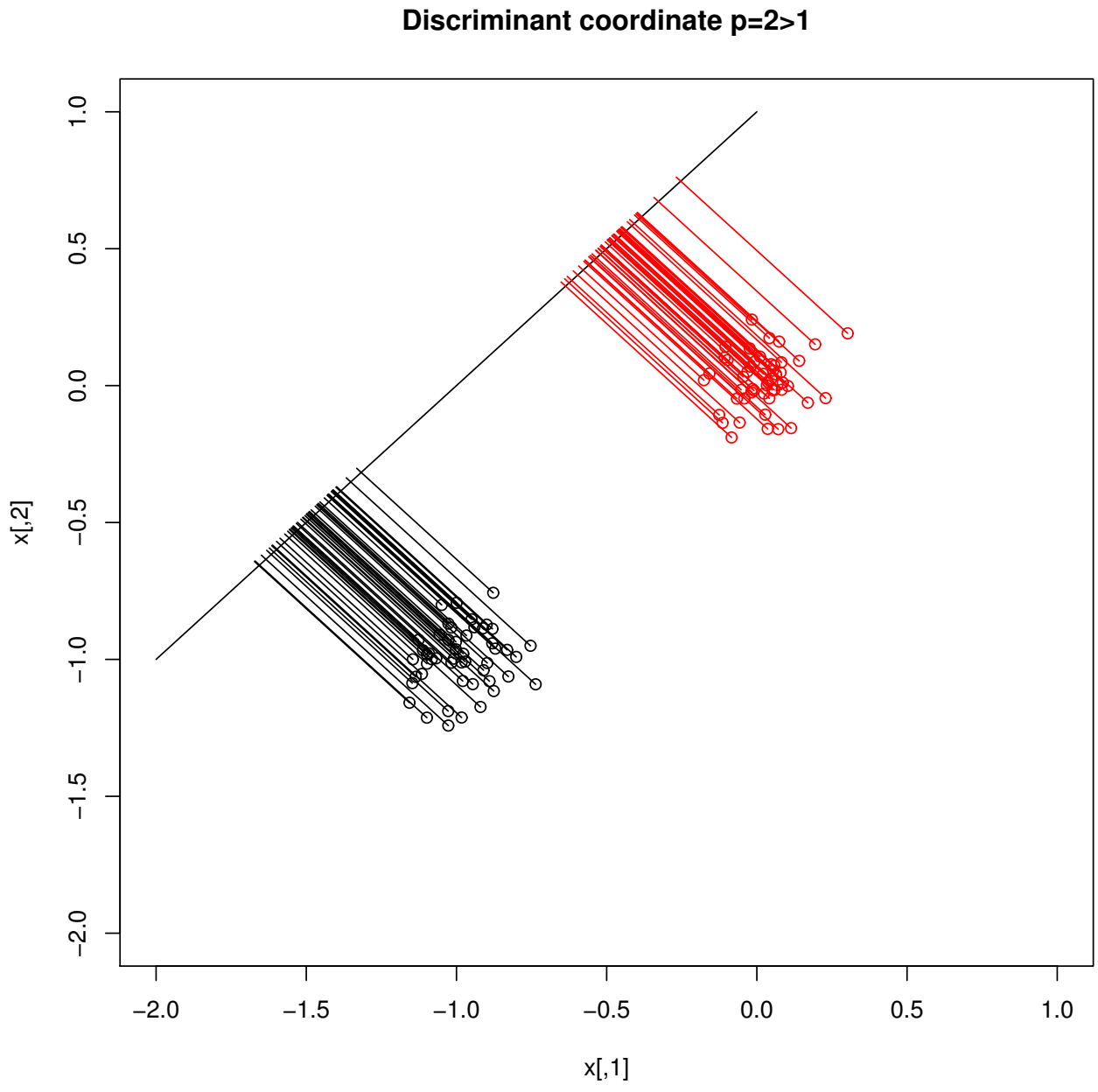


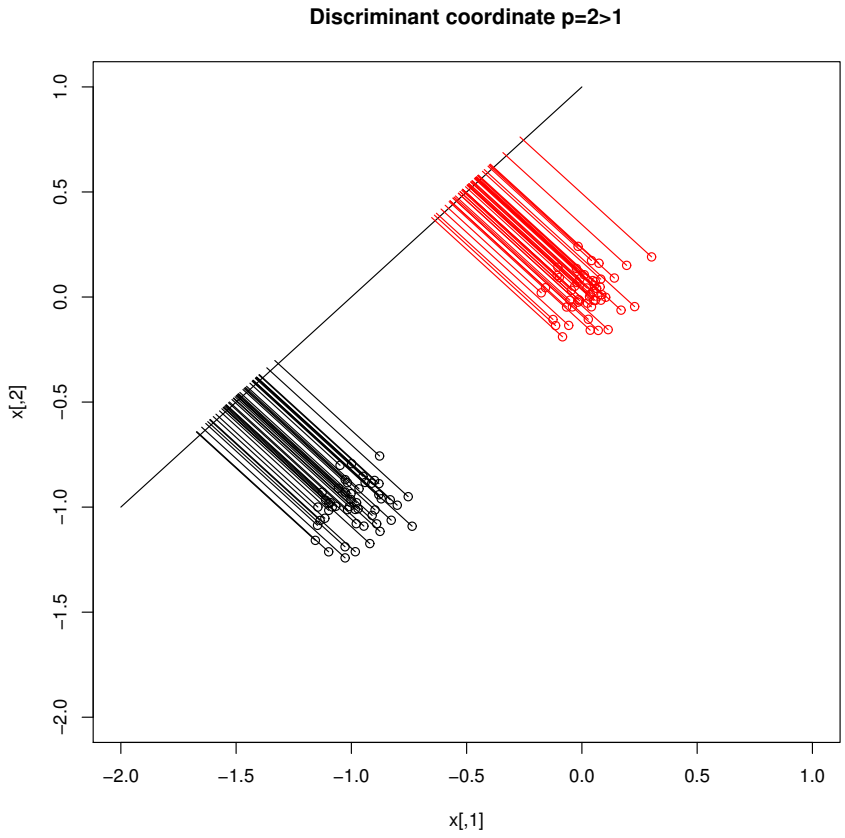
More than 3 clusters: cannot see everything in 2-d.

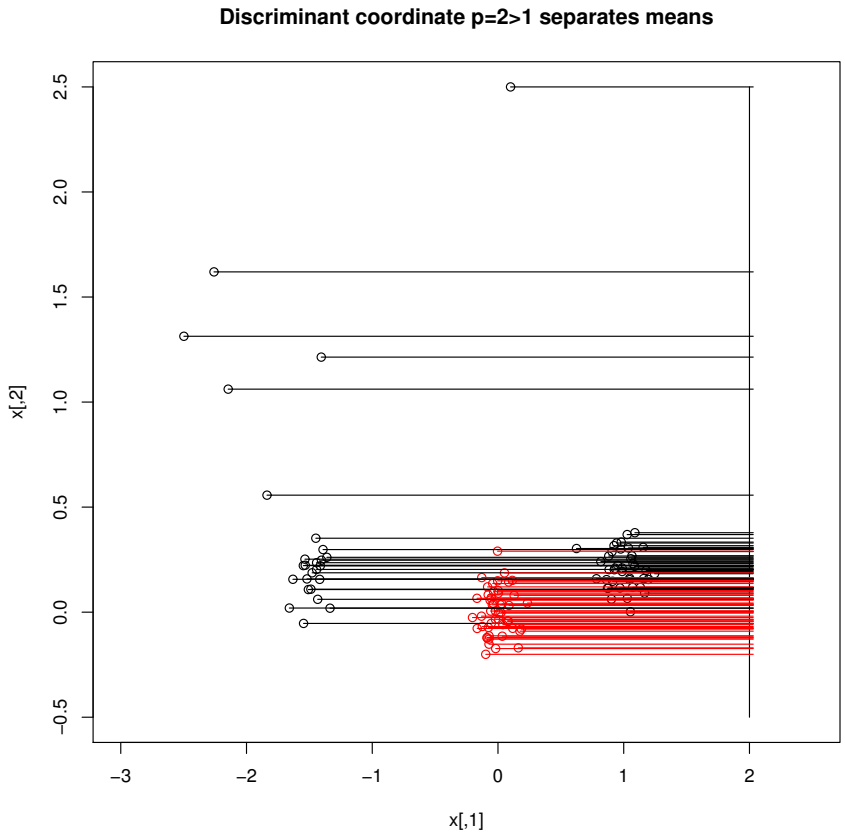
Difficulties with DC:

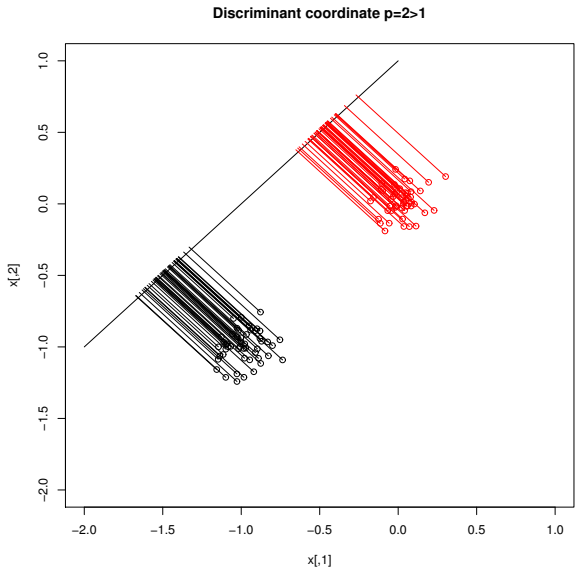
- Separation between cluster means is shown.
- All within-cluster cov-matrices equal implicitly assumed.
- More than 3 clusters: cannot see everything in 2-d.
- DCs may be dominated by outliers.

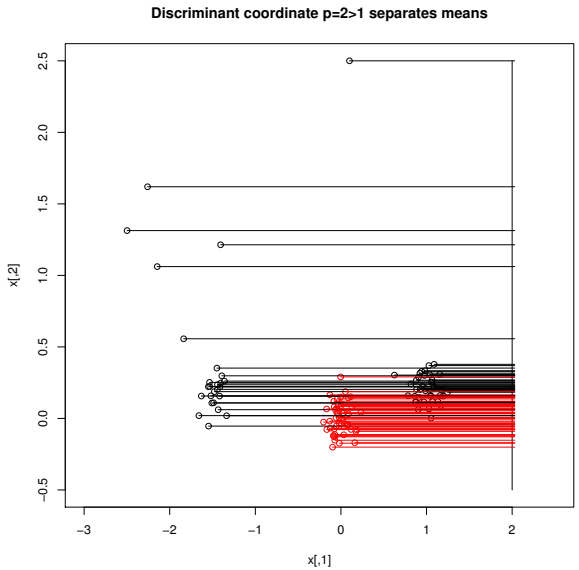
Cure: Projection method that separates single cluster from rest.

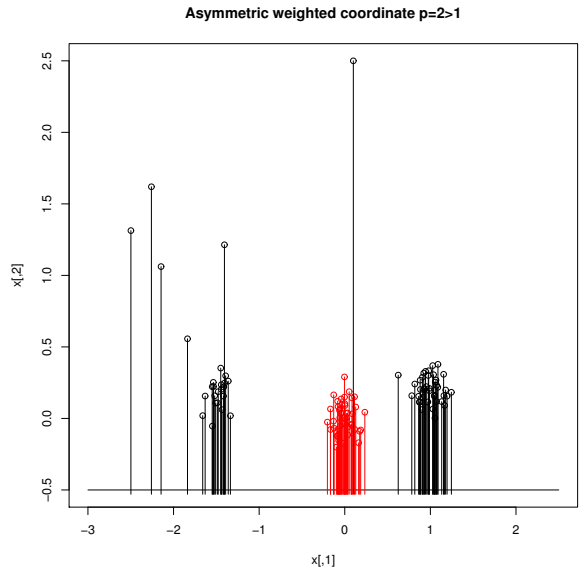

Discriminant coordinate p=2>1
x[,2]
x[,1]
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0


Discriminant coordinate p=2>1
x[,2]
x[,1]
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0


Discriminant coordinate p=2>1 separates means
x[,2]
x[,1]
2.5
2.0
1.5
1.0
0.5
0.0
−0.5
−3
−2
−1
0
1
2


Discriminant coordinate p=2>1
x[,2]
x[,1]
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0


Discriminant coordinate p=2>1 separates means
x[,2]
x[,1]
−3
−2
−1
0
1
2
−0.5
0.0
0.5
1.0
1.5
2.0
2.5


Asymmetric weighted coordinate p=2>1
x[,2]
x[,1]
−3
−2
−1
0
1
2
−0.5
0.0
0.5
1.0
1.5
2.0
2.5

Definition (Hennig (2004)) Let

$$\mathbf{B}^* = \frac{1}{n_1 n_2} \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} (\mathbf{x}_{1i} - \mathbf{x}_{2j})(\mathbf{x}_{1i} - \mathbf{x}_{2j})',$$

denoting now by $\mathbf{x}_{2j}$ all points that are not in cluster 1.
Asymmetric DCs for cluster 1 are defined by $\mathbf{Q} = \mathbf{B}^*$ and $\mathbf{R} = \mathbf{S}_1$.

Definition. Let

$$\mathbf{B}^{**} = \frac{1}{n_1 \sum_{j=1}^{n_2} w_j} \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} w_j (\mathbf{x}_{1i} - \mathbf{x}_{2j})(\mathbf{x}_{1i} - \mathbf{x}_{2j})', \text{ where}$$

$$w_j = \min\left(1, \frac{d}{(\mathbf{x}_{2j} - \mathbf{m}_1)' \mathbf{S}_1^{-1} (\mathbf{x}_{2j} - \mathbf{m}_1)}\right), \; j = 1, \ldots, n_2,$$

$d > 0$ being some constant, for example the 0.99-quantile of the χ^2_p-distribution.
Asymmetric weighted coordinates (AWC) for cluster 1
are defined by $\mathbf{Q} = \mathbf{B}^{**}$ and $\mathbf{R} = \mathbf{S}_1$.

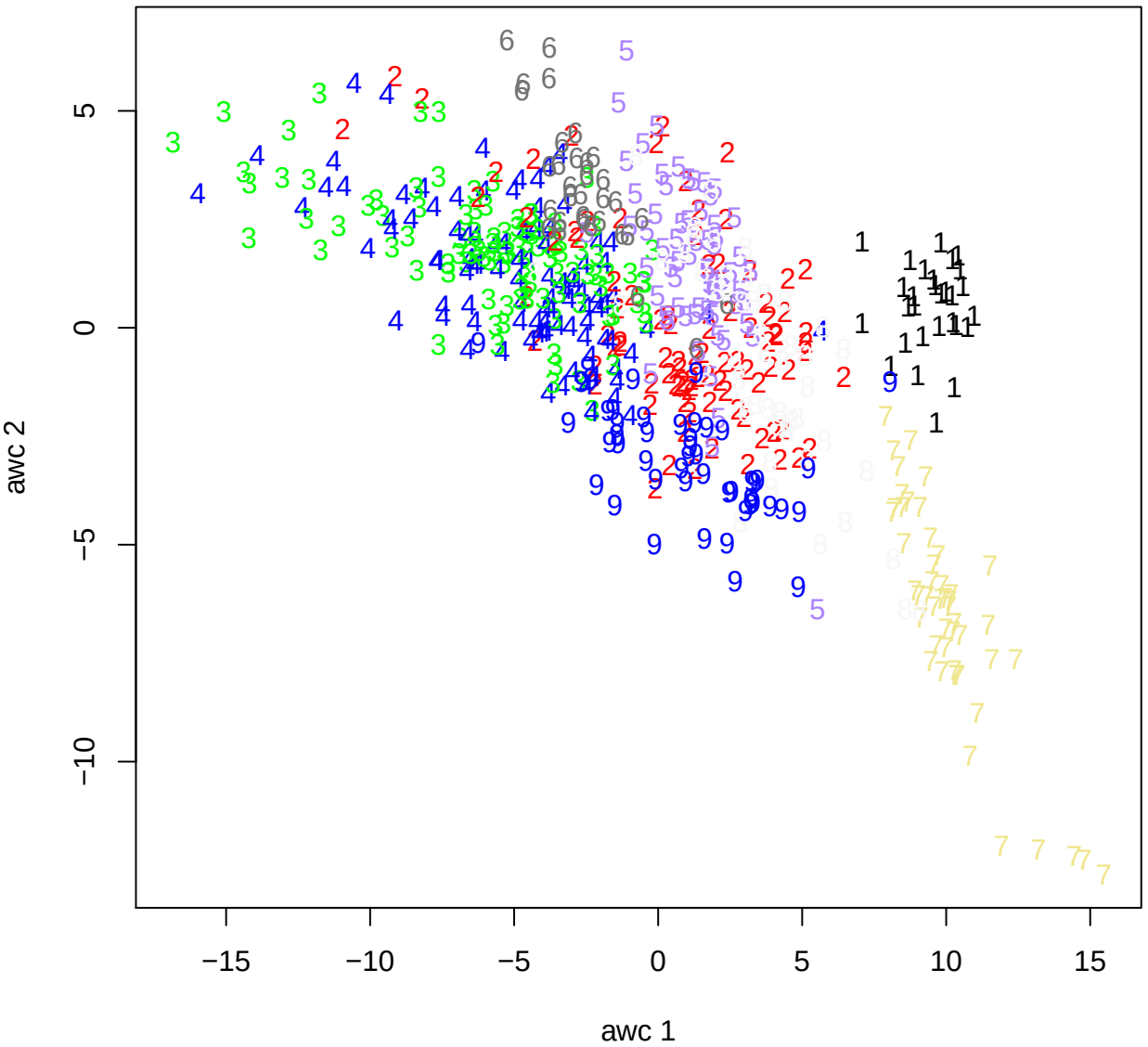
Motivation for weights: Consider $\mathbf{x}_{2j} = \mathbf{m}_1 + q\mathbf{v}$, where $\mathbf{v}$ is a unit vector w.r.t. $\mathbf{S}_1$ giving the direction of the deviation of x_{2j} from the mean $\mathbf{m}_1$ of cluster 1 and $q > 0$ is the amount of deviation. The contribution of $\mathbf{x}_{2j}$ to $\mathbf{B}^{**}$ is, for q large enough,

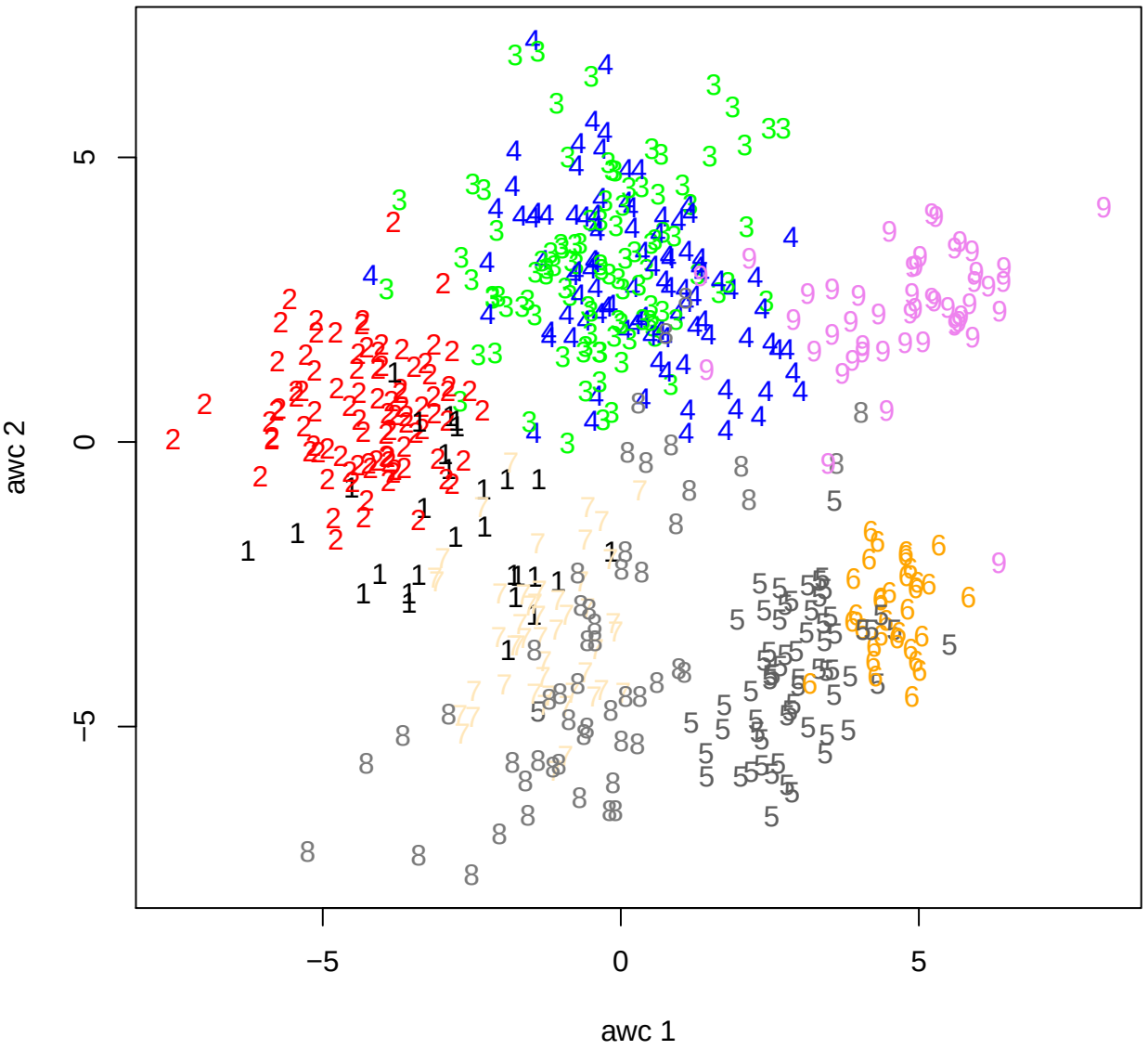
$$\sum_{i=1}^{n_1} \frac{d}{(\mathbf{x}_{2j} - \mathbf{m}_1)'\mathbf{S}_1^{-1}(\mathbf{x}_{2j} - \mathbf{m}_1)}(\mathbf{x}_{1i} - \mathbf{x}_{2j})(\mathbf{x}_{1i} - \mathbf{x}_{2j})',$$

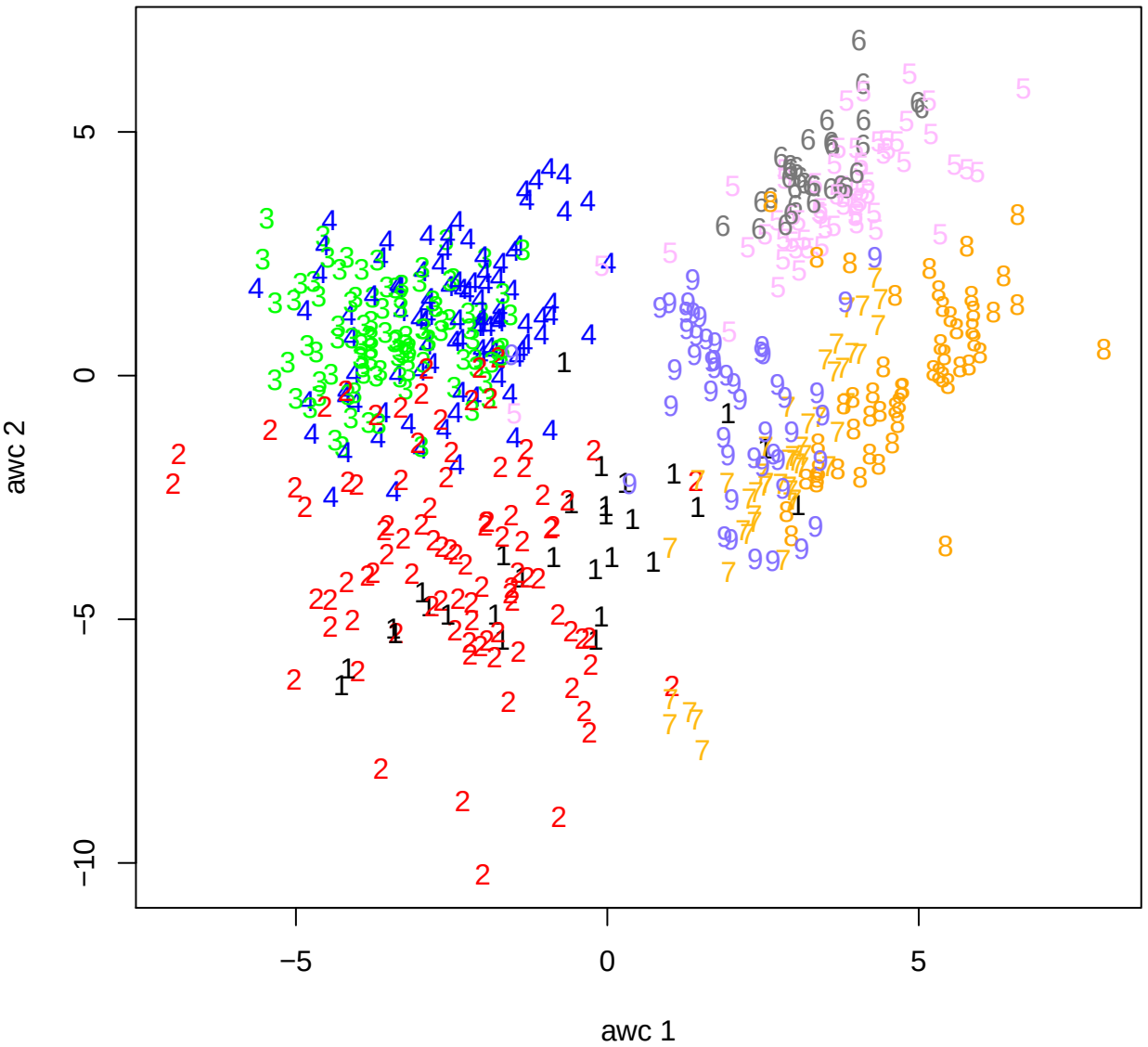
$\to n_1 d \frac{\mathbf{v}\mathbf{v}'}{\mathbf{v}'\mathbf{S}_1^{-1}\mathbf{v}}$ for $q \to \infty$.

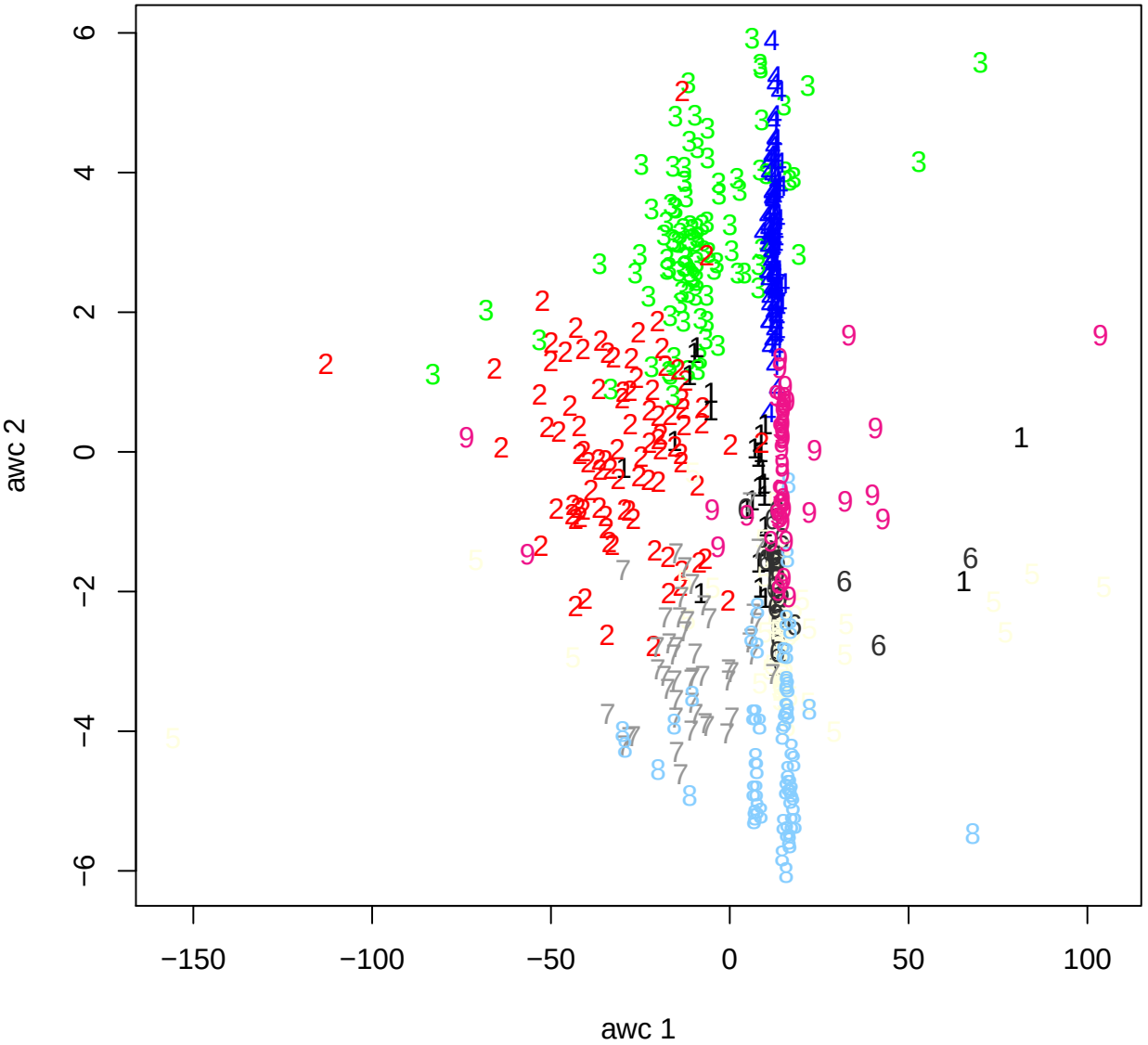
Look for a single cluster at a time.

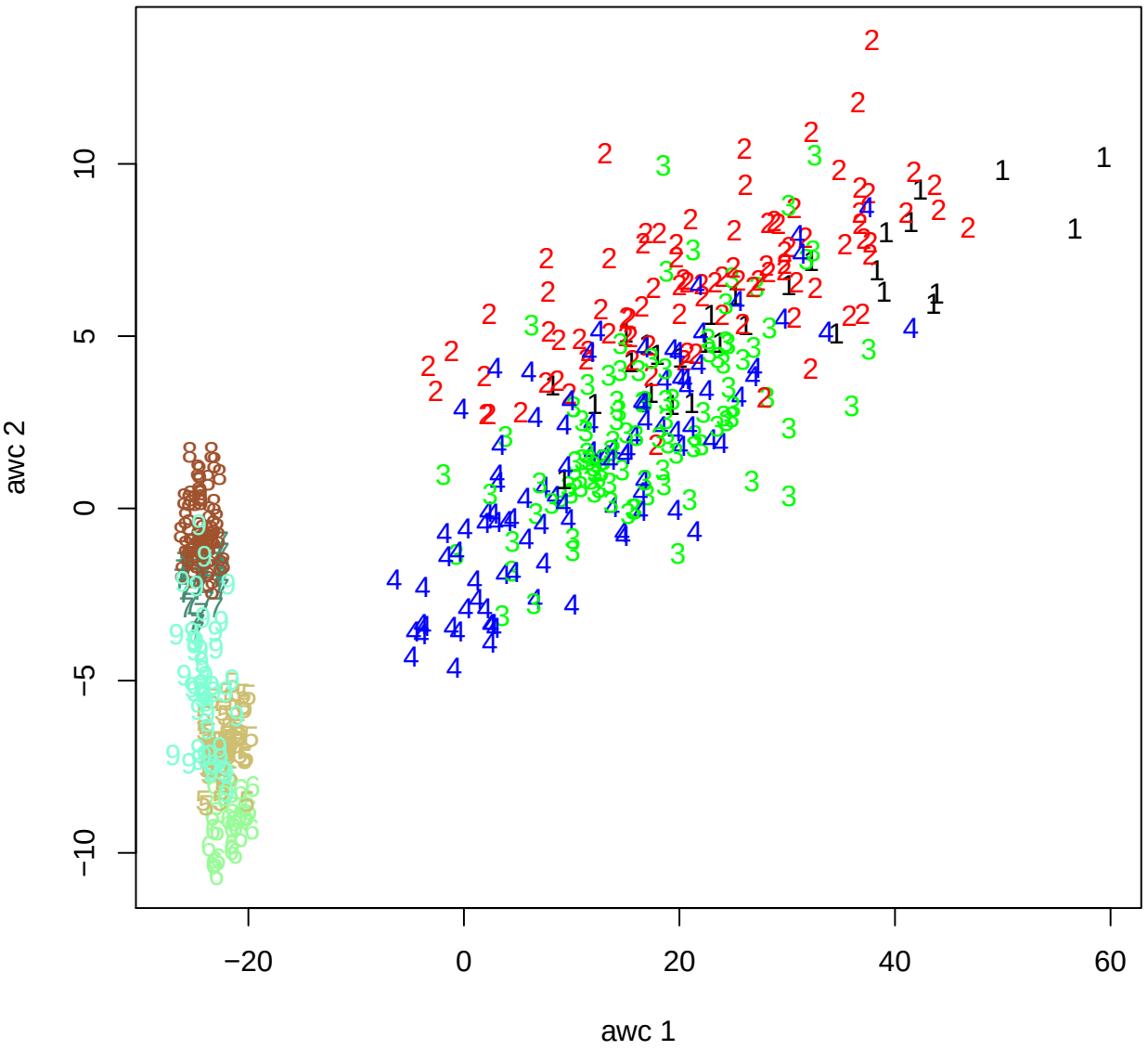
```
plotcluster(solive,smolive$classification,clnum=1)
plotcluster(solive,smolive$classification,clnum=2)
plotcluster(solive,smolive$classification,clnum=3)
plotcluster(solive,smolive$classification,clnum=4)
plotcluster(solive,smolive$classification,clnum=8)
```


awc 2
awc 1
5
0
−5
−10
−15
−10
−5
0
5
10
15


awc 2
5
0
−5
−5
0
5
awc 1


awc 2
5
0
−5
−10
−5
0
5
awc 1


awc 2
6
4
2
0
−2
−4
−6
−150
−100
−50
0
50
100
awc 1

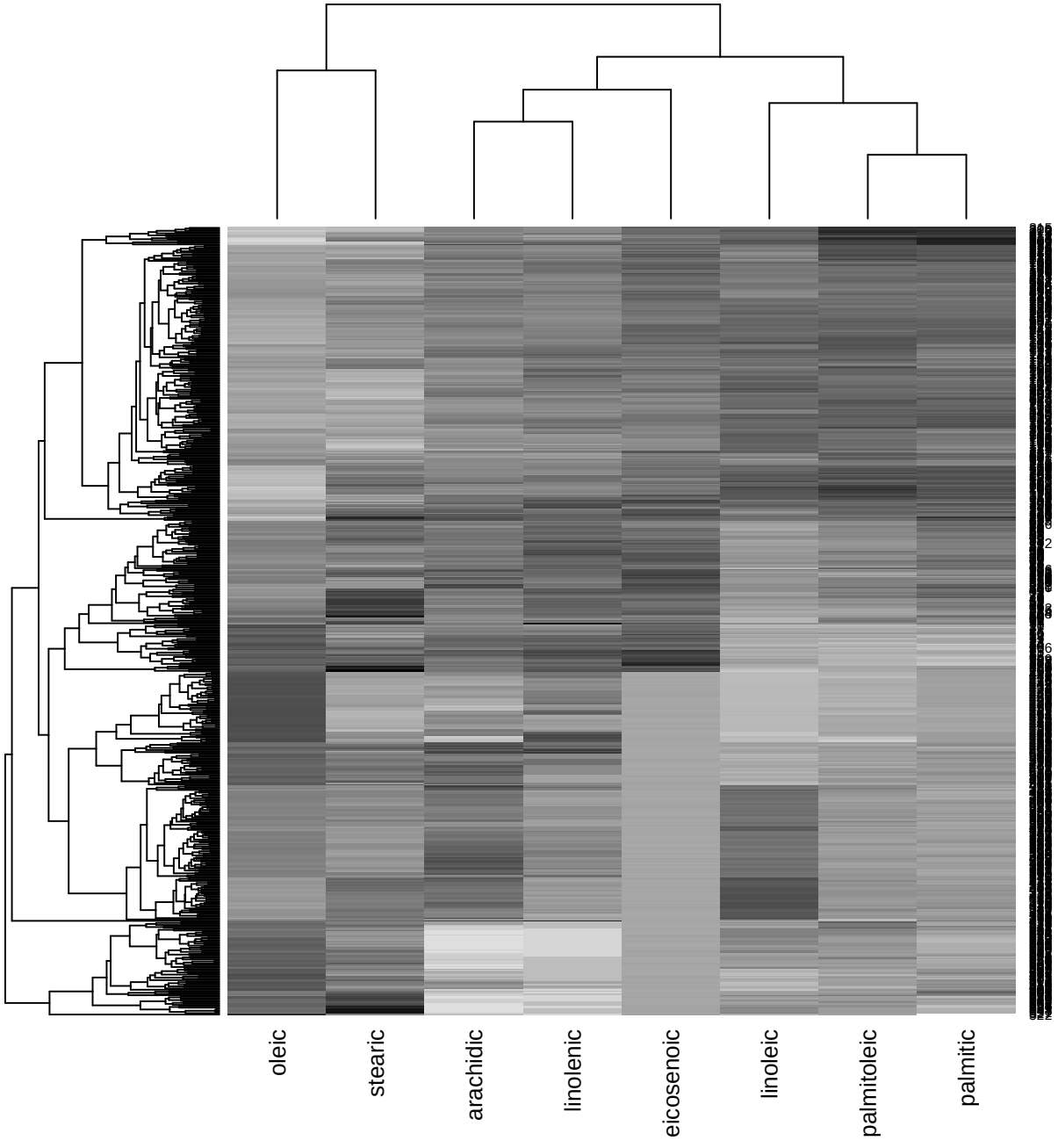

awc 2
10
5
0
−5
−10
−20
0
20
40
60
awc 1

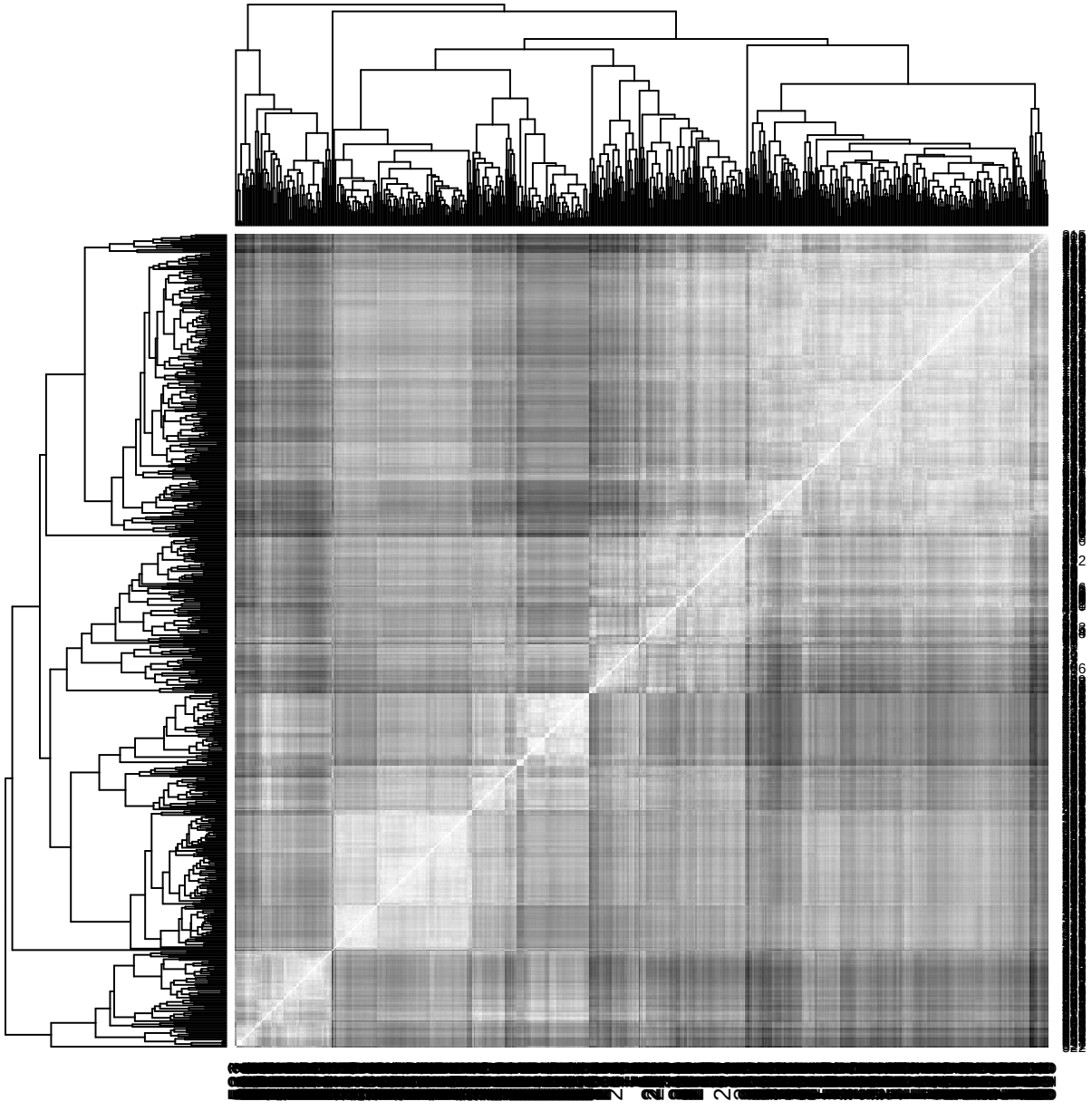
2.2.3 Heatmaps

. . . can be useful for visualising higher dimensions.

Visualising a hierarchical clustering:

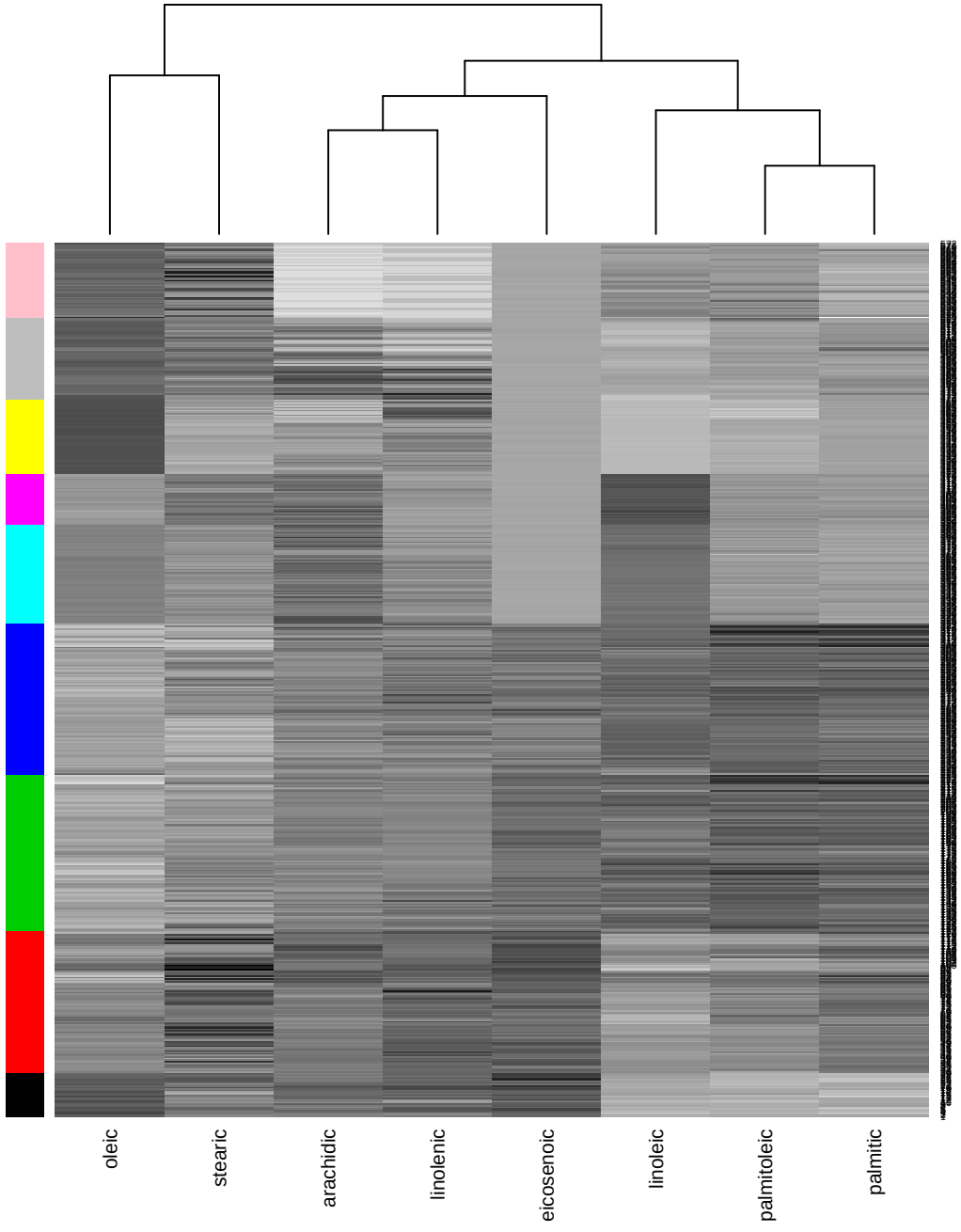
```
dolive <- dist(solive) # Euclidean distances
aveolive <- hclust(dolive,method="average")

heatmap(solive,Rowv=as.dendrogram(aveolive),
 col=grey(seq(1,0,-0.01)),scale="none",cexCol=1)
# Heatmap of observations

heatmap(as.matrix(dolive),Rowv=as.dendrogram(aveolive),
 Colv=as.dendrogram(aveolive),col=grey(seq(1,0,-0.01)),
 scale="none",cexCol=1)
# Heatmap of distances
```


oleic
stearic
arachidic
linolenic
eicosenoic
linoleic
palmitoleic
palmitic



Visualising a partition:

```
heatmap(solive[order(smolive$classification),],
 Rowv=NA,col=grey(seq(1,0,-0.01)),scale="none",
 cexCol=1,
 RowSideColors=c(palette(),"pink")[smolive$classification]
[order(smolive$classification)])
```


oleic
stearic
arachidic
linolenic
eicosenoic
linoleic
palmitoleic
palmitic

2.3 Stability assessment

General principle for stability assessment

- Generate several new datasets out of the original one.
- Cluster all these new datasets.
- Define statistic to formalise how similar new clusterings are to the original one.
- If they are very similar, it's stable.

2.3.1 Bootstrap stability assessment

Fang and Wang (2012): for cluster number G, $b = 1, \ldots, B$:

- Step 1 Draw two bootstrap subsamples.
- Step 2 Partition them both into G clusters.
- Step 3 Generalise these to produce two clusterings on whole dataset by suitable classifier, i.e., nearest centroid for k-means, QDA for Gaussian mixtures.
- Step 4 With cluster assignments c_{1bG}, c_{2bG}:

$$S_{bG} = \frac{1}{n^2} \sum_{i,j}^{n} |1\left[c_{1bG}(\mathbf{x}_i) = c_{1bG}(\mathbf{x}_j)\right] - 1\left[c_{2bG}(\mathbf{x}_i) = c_{2bG}(\mathbf{x}_j)\right]|.$$

Average dissimilarity between clusterings

$$S_G = \frac{1}{B}\sum_{b=1}^{B} S_{bG}$$

measures stability.

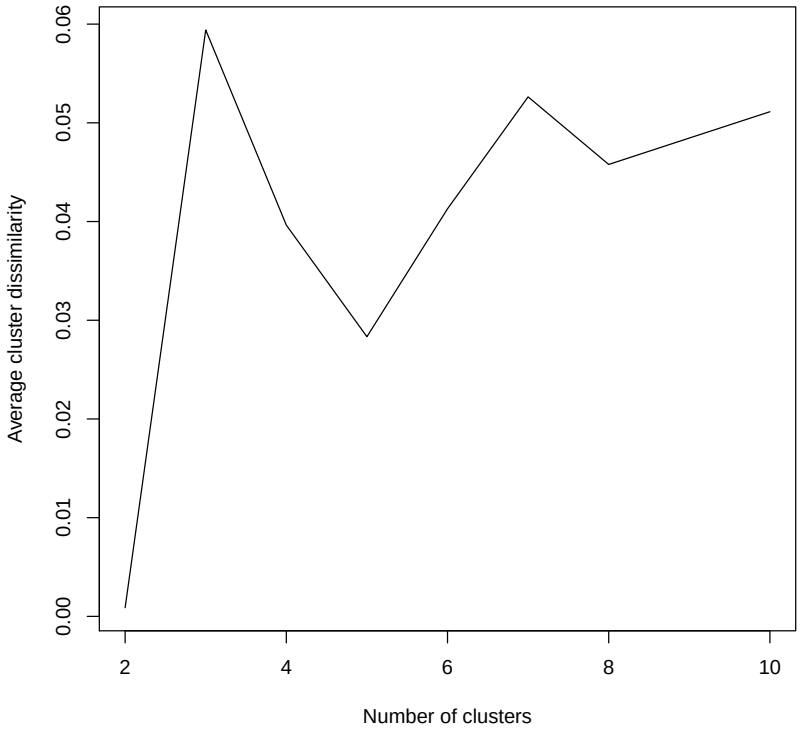
Selection of G: minimise S_G.

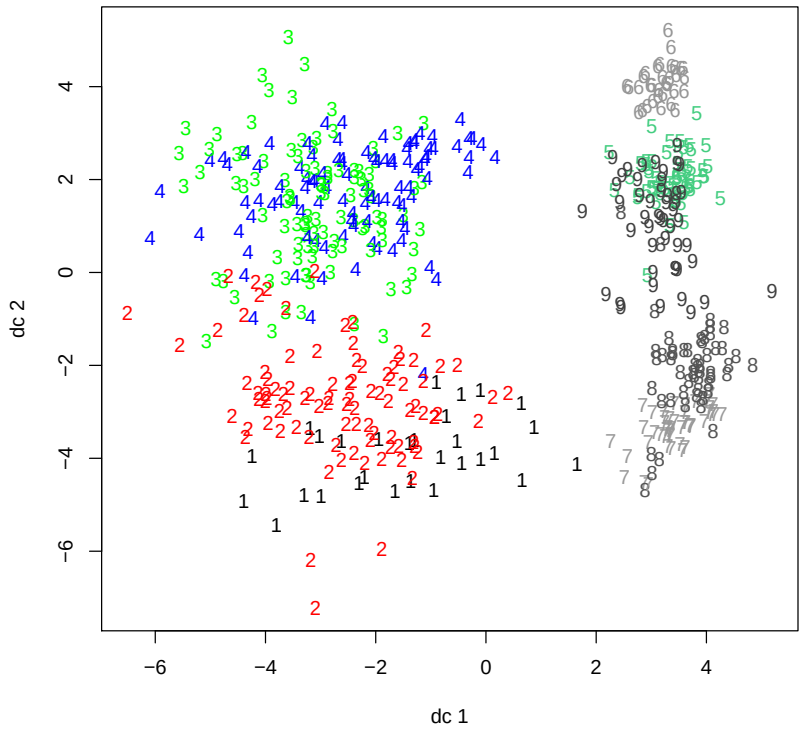
This cannot be used with $G = 1$!

Alternative method (both in fpc and flexclust):
"prediction strength" (Tibshirani and Walther (2005)).

```
library(fpc)
nsbolive <- nselectboot(solive,clustermethod=noisemclustCBI,
classification="qda",krange=2:10,nnk=0)

plot(2:10,nsbolive$stabk[2:10],xlab="Number of clusters",
ylab="Average cluster dissimilarity",type="l")
```

$G = 2$ is optimal here; makes some sense.





2.3.2 Cluster-wise stability assessment

Many clusterings are unstable in one way or another.

Want to know which clusters are stable

$\Rightarrow$ here *cluster-wise* methodology,

`clusterboot` in package fpc (Hennig (2007)).

1. Use the Jaccard coefficient

$$\gamma(C, D) = \frac{|C \cap D|}{|C \cup D|}.$$

   to measure similarity between two subsets of a set.
2. Repeat B times steps 2-4:
   draw bootstrap datasets from original one,
3. apply the same clustering method to them.
4. For $C \in \mathcal{C}$ record $m_i = \max_{D \neq C} \gamma(C, D)$
5. Use $\bar{\gamma} = \frac{1}{B} \sum_{i=1}^{B} m_i$ to assess stability of C.

Other resampling methods exist.

For computing γ for given original cluster and cluster in resampled dataset, use only points that are both in original dataset and in resampled one.

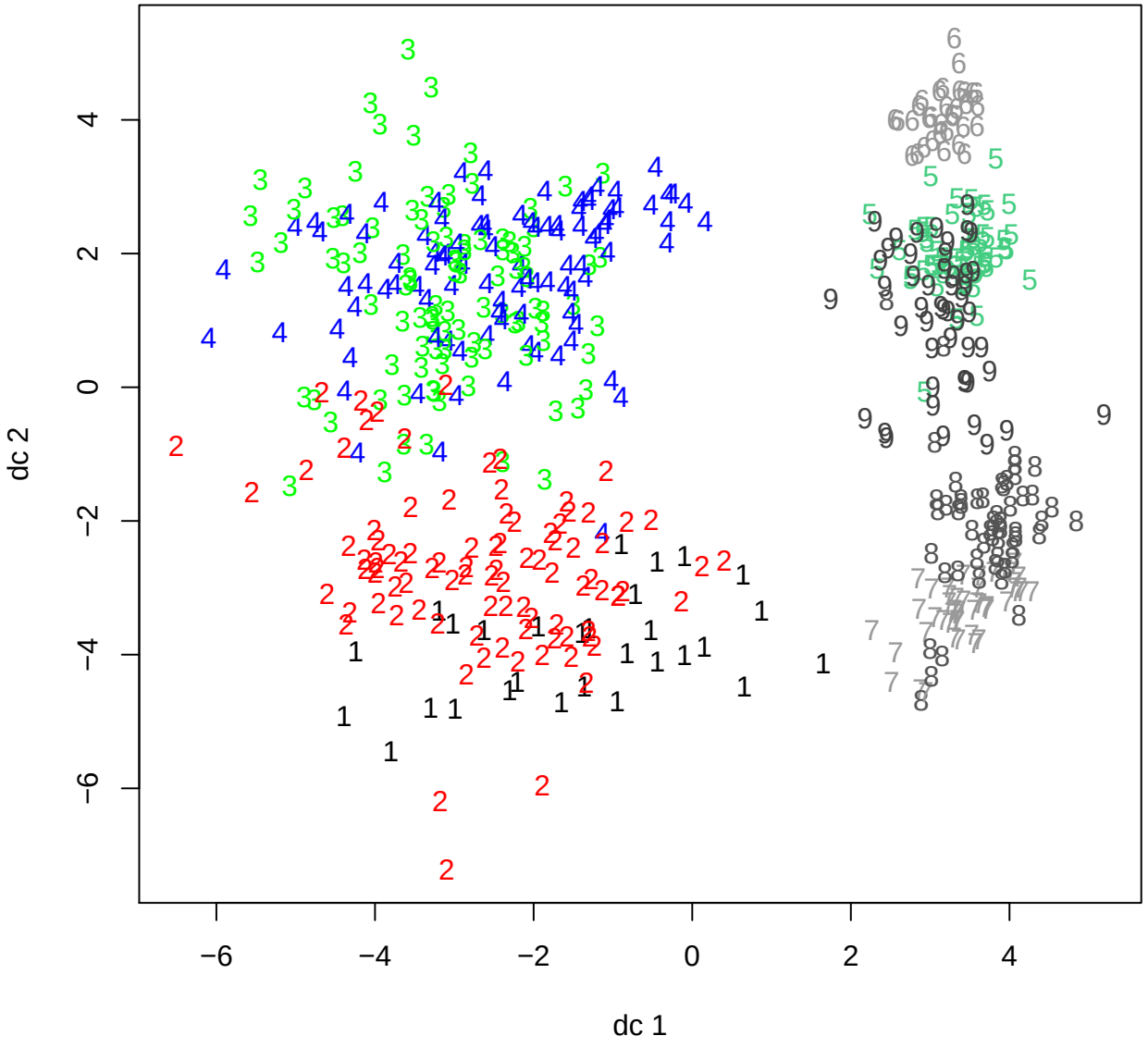
For computing γ for given original cluster and cluster in resampled dataset, use only points that are both in original dataset and in resampled one.

Interpretation:

- 0.5 is minimum v so that for given partition it's possible for every cluster to find another partition so that maximum γ is $\leq v$.
- New partition with m clusters, original one with $k > m \Rightarrow \exists$ at least $k - m$ clusters in original partition for which no $\gamma > v$.

Consider clusters with $\max \gamma \leq 0.5$ as "dissolved".
Demand $\bar{\gamma} >> 0.5$ for stability.

```
cbolive <-
clusterboot(solive,clustermethod=noisemclustCBI,seed=12345,G=1:9,nnk=0)
# B=100 takes some time.

> cbolive
* Cluster stability assessment *
Cluster method:  mclustBIC
Full clustering results are given as parameter result
of the clusterboot object, which also provides further statistics
of the resampling results.
Number of resampling runs:  100

Number of clusters found in data:  9
```


dc 2
dc 1
−6
−4
−2
0
2
4

```
 Clusterwise Jaccard bootstrap (omitting multiple points) mean:
[1] 0.5683592 0.7735421 0.5009002 0.4879526 0.9628937 0.9279366 0.9084475
[8] 0.7808512 0.8851998
dissolved:
[1] 49  2 71 73  0  4  1 13  4
recovered:
[1] 33 55  9 11 96 95 94 72 90
```

Instabilities can result from

- features of the data,
- instabilities of clustering method,
- mismatch between the two.

Stable clusters are not necessarily good.
(Fixing $G = 1$ is always stable.)
Unstable clusters can be tolerated if stability is not the aim or if only parts can be stably clustered.

2.4 Internal validation indexes

. . . attempt to quantify the quality of a clustering
just by using the data and the clustering,
not any external information (as ARI does).

Traditionally these are often used for
estimating G by optimisation,
and try to capture quality in a single number.

Overview: Halkidi et al. (2015).

Some examples

Many indexes are based on
the within clusters sum of squares:

$$W_G = \sum_{i=1}^{n} \|\mathbf{x}_i - \bar{\mathbf{x}}_{C(i)}\|^2$$

These are directly connected to k-means,
despite being often advertised as more general.

W_G will decrease with G.
Adjustments are needed to define $i(W_G)$ so that
optimum $i(W_G)$ can indicate optimum G.

One of these is the...

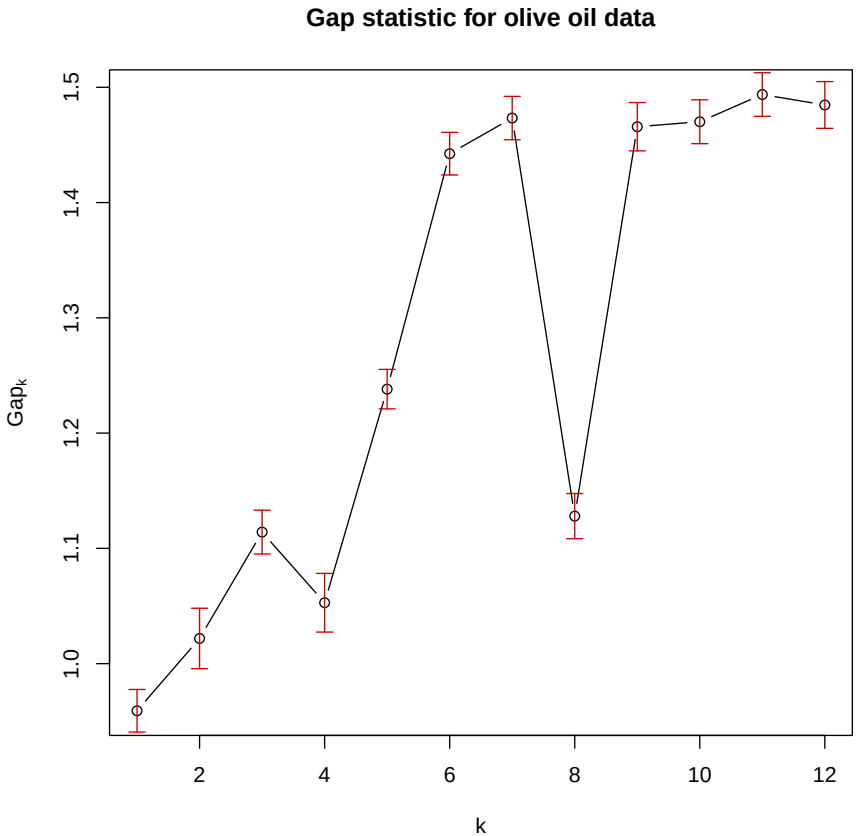
2.4.1 Gap statistic (Tibshirani et al. (2001))

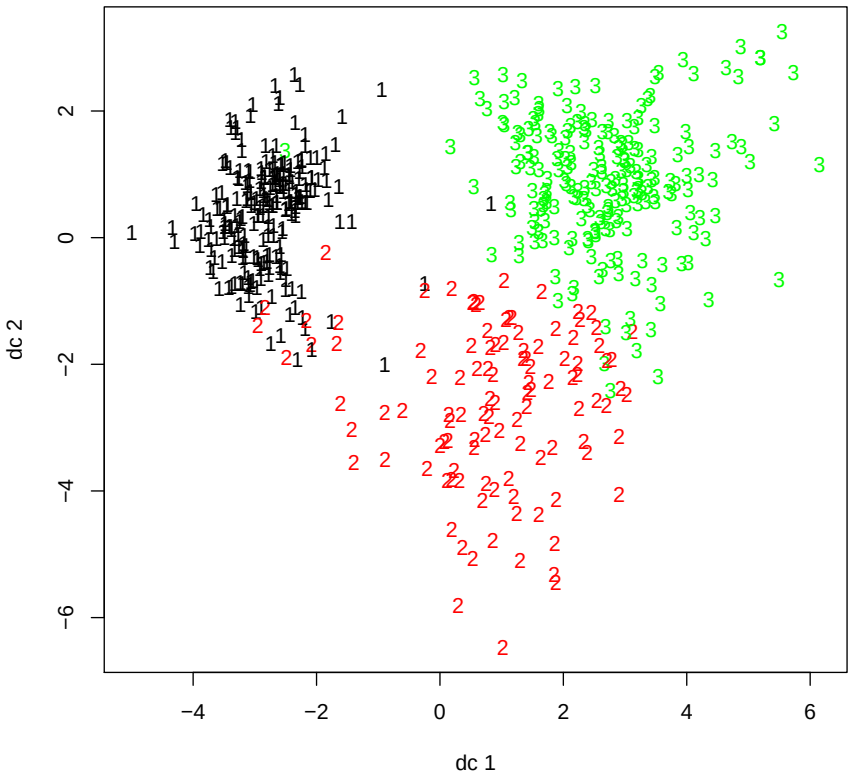
Simulate $\hat{E}(\log W_G)$, $\hat{\mathrm{sd}}(\log W_G)$ under uniform distribution,
Choose smallest G so that

$$\begin{aligned} \mathrm{gap}(G) &> \mathrm{gap}(G+1) - \hat{\mathrm{sd}}(\log W_{G+1}), \\ \text{where } \mathrm{gap}(G) &= \hat{E}(\log W_G) - \log W_G. \end{aligned}$$

```
library(cluster)
set.seed(998877)
gapolive <- clusGap(solive,kmeans,12,d.power=2)
# kmeans up to 12 clusters

print(gapolive,method="Tibs2001SEmax")
```

```
Clustering Gap statistic ["clusGap"].
B=100 simulated reference sets, k = 1..12
 --> Number of clusters (method 'Tibs2001SEmax', SE.factor=1): 3
          logW   E.logW       gap     SE.sim
 [1,] 7.733684 8.692806 0.9591225 0.01847567
 [2,] 7.308810 8.330658 1.0218475 0.02621990
 [3,] 7.069895 8.184027 1.1141317 0.01900816
 [4,] 7.006405 8.059251 1.0528461 0.02544953
 [5,] 6.748384 7.986522 1.2381383 0.01714330
 [6,] 6.482607 7.925032 1.4424247 0.01849207
 [7,] 6.394681 7.867982 1.4733010 0.01879081
 [8,] 6.690916 7.818907 1.1279906 0.01959830
 [9,] 6.312400 7.778200 1.4658006 0.02095239
[10,] 6.267933 7.738035 1.4701023 0.01902356
[11,] 6.210579 7.704309 1.4937294 0.01888529
[12,] 6.189764 7.674423 1.4846593 0.02026871
```





This isn't terribly convincing. . . ,
but then k-means isn't really appropriate here.

2.4.2 Indexes for general dissimilarity data

Can be used with Euclidean for $n \times p$-data.

Average silhouette width (ASW)

(Kaufman and Rousseeuw (1990))

$sw(i, \mathcal{C}) = \frac{b(i,\mathcal{C}) - a(i,\mathcal{C})}{\max(a(i,\mathcal{C}), b(i,\mathcal{C}))}$,

$$a(i, \mathcal{C}) = \frac{1}{|C_j| - 1} \sum_{x \in C_j} d(x_i, x), \;\; b(i, \mathcal{C}) = \min_{x_i \notin C_l} \frac{1}{|C_l|} \sum_{x \in C_l} d(x_i, x).$$

Maximum average $sw \Rightarrow$ good $\mathcal{C}$.

This contrasts within-cluster homogeneity
with separation from neighbouring clusters.

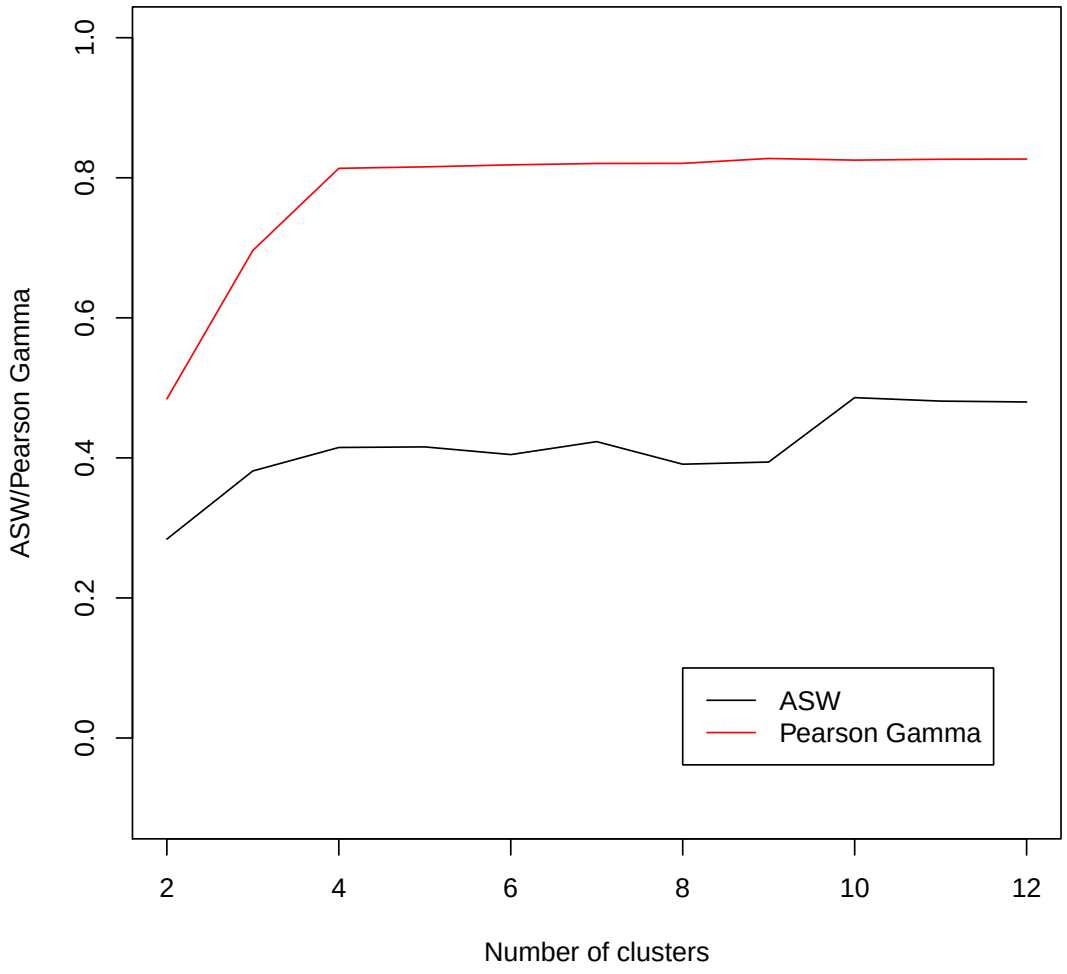
Pearson correlation version of Hubert's Γ
(Hubert and Schultz (1976))

$$\Gamma_P(G) = \mathrm{cor}(\mathrm{vec}(D), \mathrm{vec}(D_{\mathcal{C}_G})),$$

$\mathrm{vec}(D)$ vector of dissimilarities $d(\mathbf{x}_i, \mathbf{x}_j)$,
$\mathrm{vec}(D_{\mathcal{C}_G})$: 0 if $\mathbf{x}_i, \mathbf{x}_j$ in same cluster, 1 otherwise.

This focuses on approximation
of dissimilarity structure by clustering.

```
library(fpc)
cstrigona <- list()
atrigona <- hclust(as.dist(tai$distmat),method="average")
asw <- pg <- numeric(0)
for (i in 2:12){
  cli <- cutree(atrigona,k=i)
  cstrigona[[i]] <- cluster.stats(tai$distmat,cli)
  asw[i] <- cstrigona[[i]]$avg.silwidth
  pg[i] <- cstrigona[[i]]$pearsongamma
}

plot(2:12,asw[2:12],ylim=c(-0.1,1),xlab="Number of clusters",
 type="l",ylab="ASW/Pearson Gamma")
points(2:12,pg[2:12],type="l",col=2)
legend(8,0.1,c("ASW","Pearson Gamma"),lty=1,col=1:2)
```

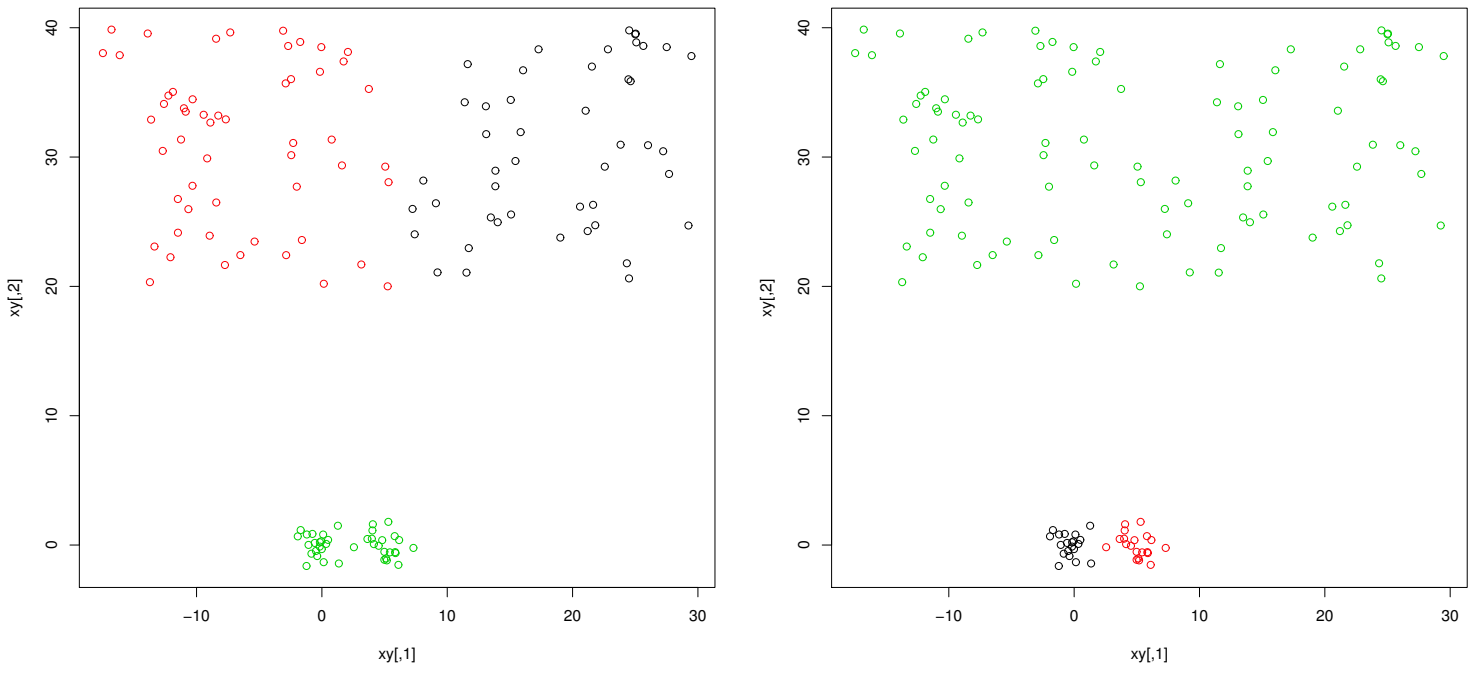


ASW picks $G = 10$, Γ_P picks $G = 9$, just.

All these suffer from "one size fits it all"-approach.

All these suffer from "one size fits it all"-approach.

Homogeneity will normally dominate here:



2.4.3 Measuring specific aspects of cluster quality

There are various different aims of clustering.

Measure them separately
to characterise a clustering,
instead of producing a single ranking of clusterings.

Current research project of mine.

Typical clustering aims

- Between-cluster separation

Typical clustering aims

- Between-cluster separation
- Within-cluster homogeneity (low distances)

Typical clustering aims

- Between-cluster separation
- Within-cluster homogeneity (low distances)
- Within-cluster homogeneous distributional shape

Typical clustering aims

- Between-cluster separation
- Within-cluster homogeneity (low distances)
- Within-cluster homogeneous distributional shape
- Good representation of data by centroids

Typical clustering aims

- Between-cluster separation
- Within-cluster homogeneity (low distances)
- Within-cluster homogeneous distributional shape
- Good representation of data by centroids
- Little loss of information
  from original distance between objects.

Typical clustering aims

- Between-cluster separation
- Within-cluster homogeneity (low distances)
- Within-cluster homogeneous distributional shape
- Good representation of data by centroids
- Little loss of information
  from original distance between objects.
- Clusters are regions of high density
  without within-cluster gaps

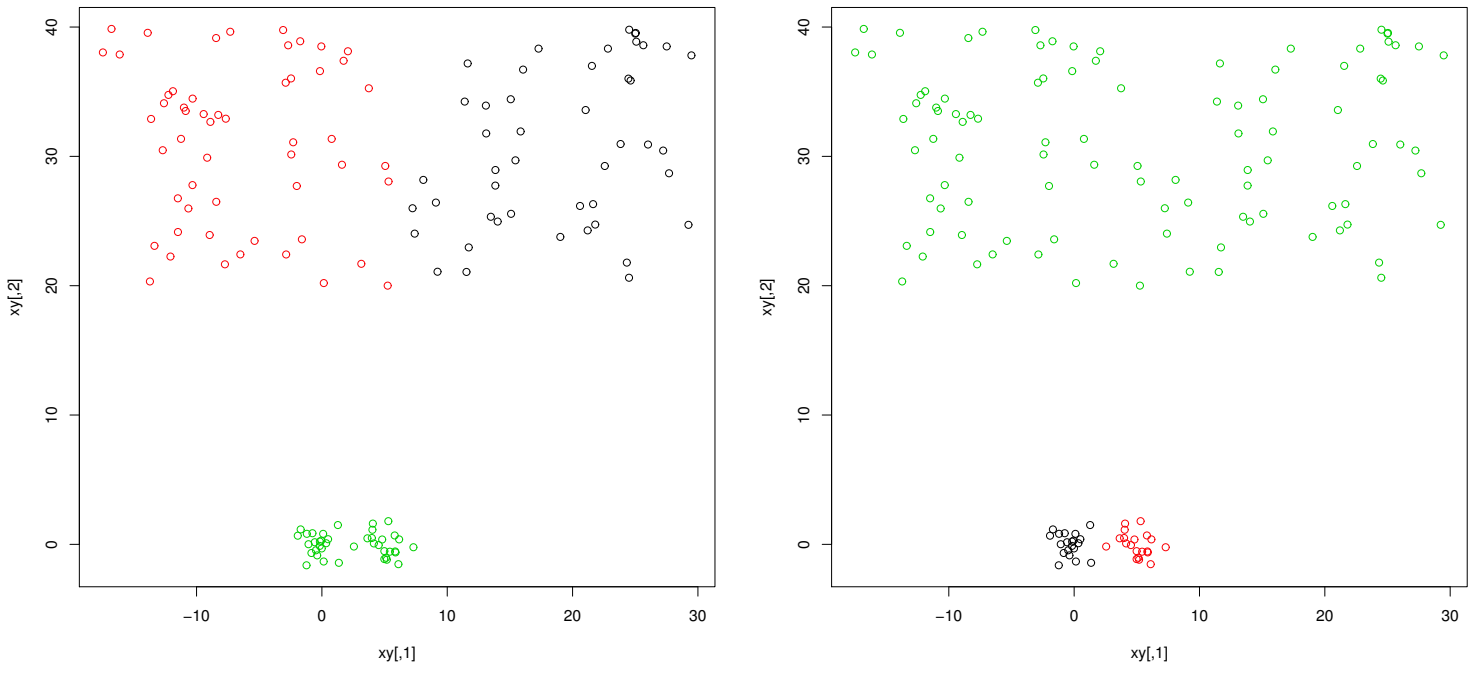
Typical clustering aims

- Between-cluster separation
- Within-cluster homogeneity (low distances)
- Within-cluster homogeneous distributional shape
- Good representation of data by centroids
- Little loss of information
  from original distance between objects.
- Clusters are regions of high density
  without within-cluster gaps
- Uniform cluster sizes

Typical clustering aims

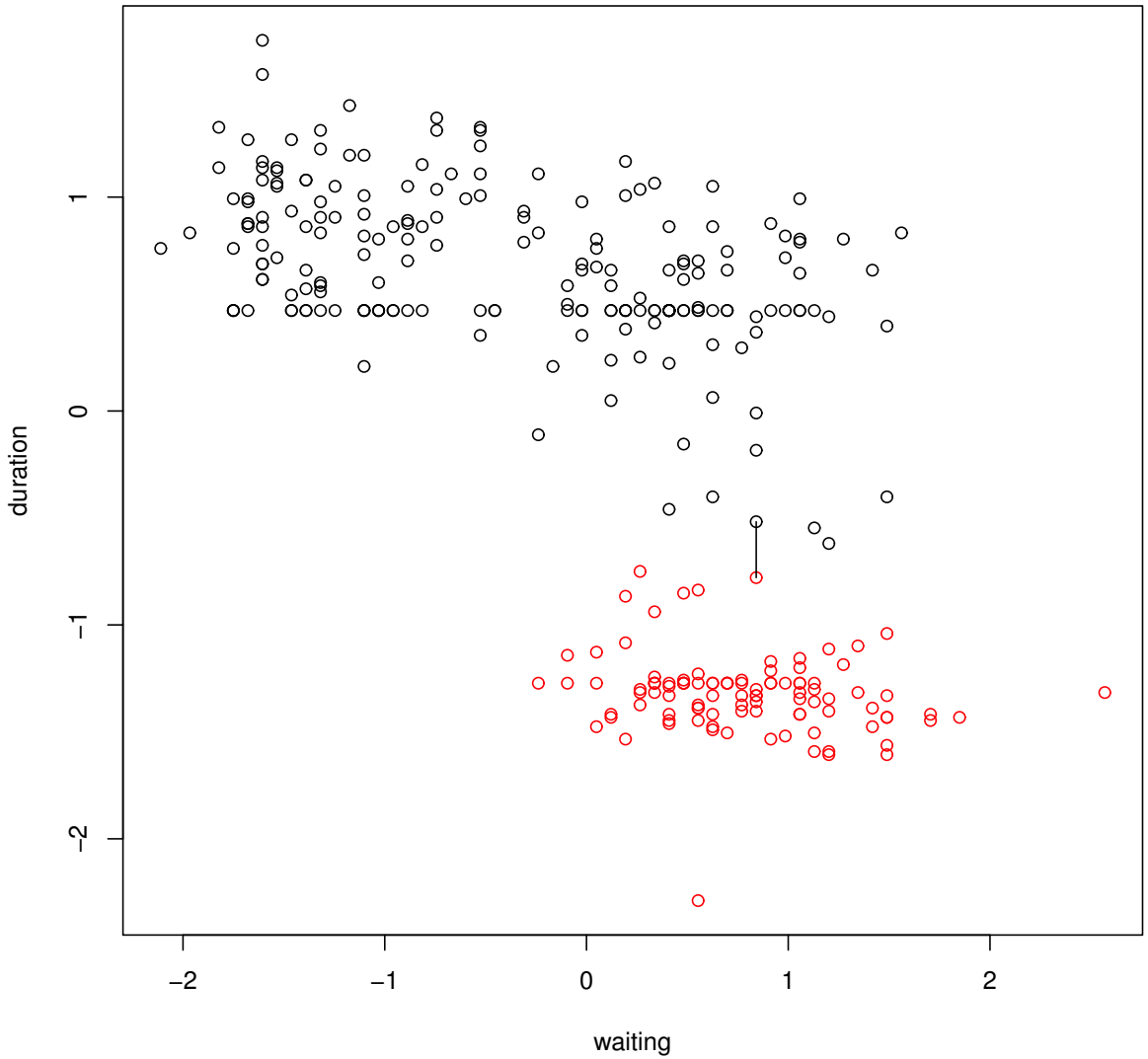
- Between-cluster separation
- Within-cluster homogeneity (low distances)
- Within-cluster homogeneous distributional shape
- Good representation of data by centroids
- Little loss of information
  from original distance between objects.
- Clusters are regions of high density
  without within-cluster gaps
- Uniform cluster sizes
- Stability

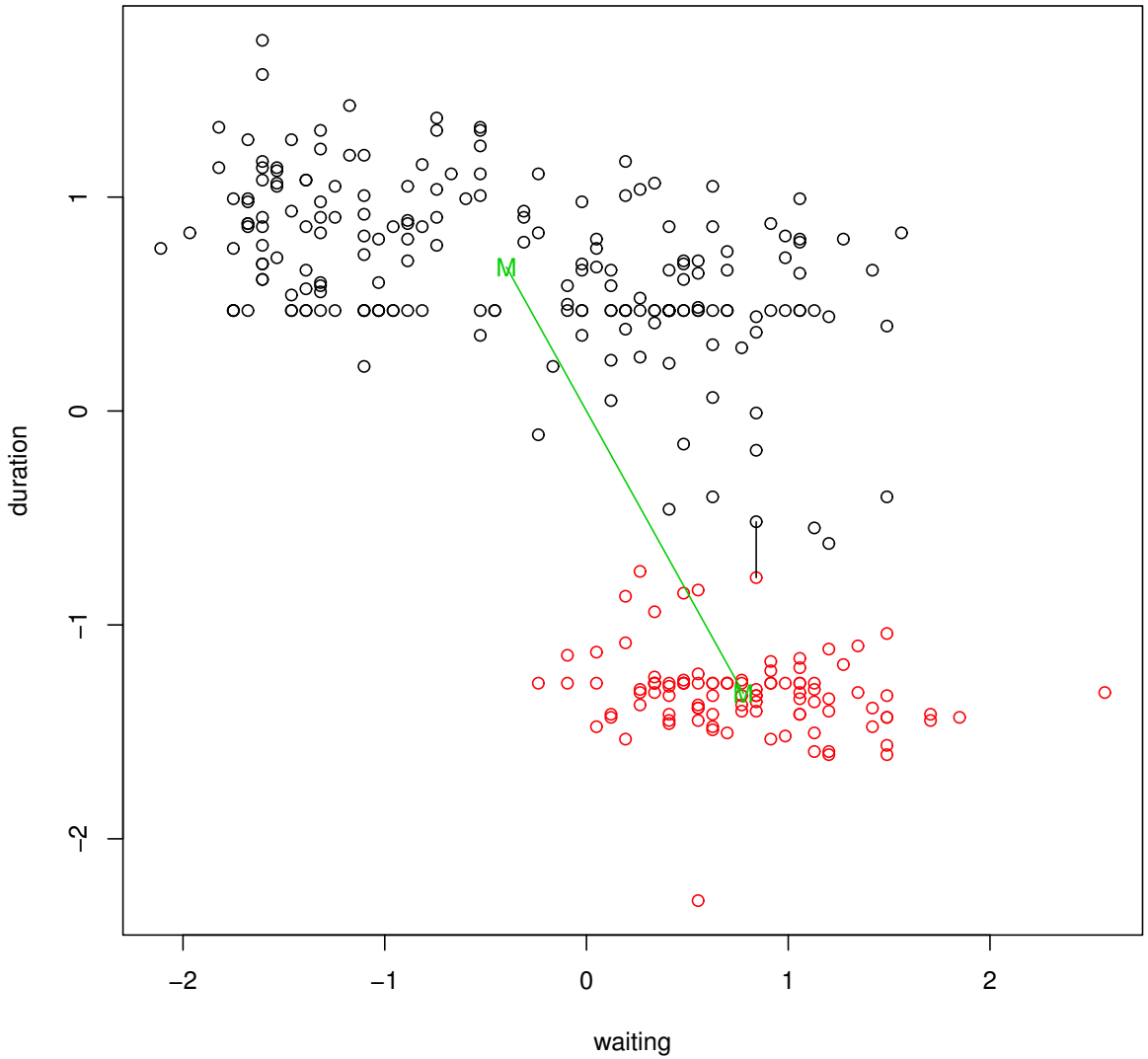
These may be in conflict with each other.



Measuring between-cluster separation

$\exists$ several ways measuring separation (as for other aims).
Straightforward: min distance between any two clusters, or distance between centroids (e.g., k-means).


duration
waiting
−2
−1
0
1
2
−2
−1
0
1


duration
waiting
M
M
−2
−1
0
1
2
−2
−1
0
1

Measuring between-cluster separation

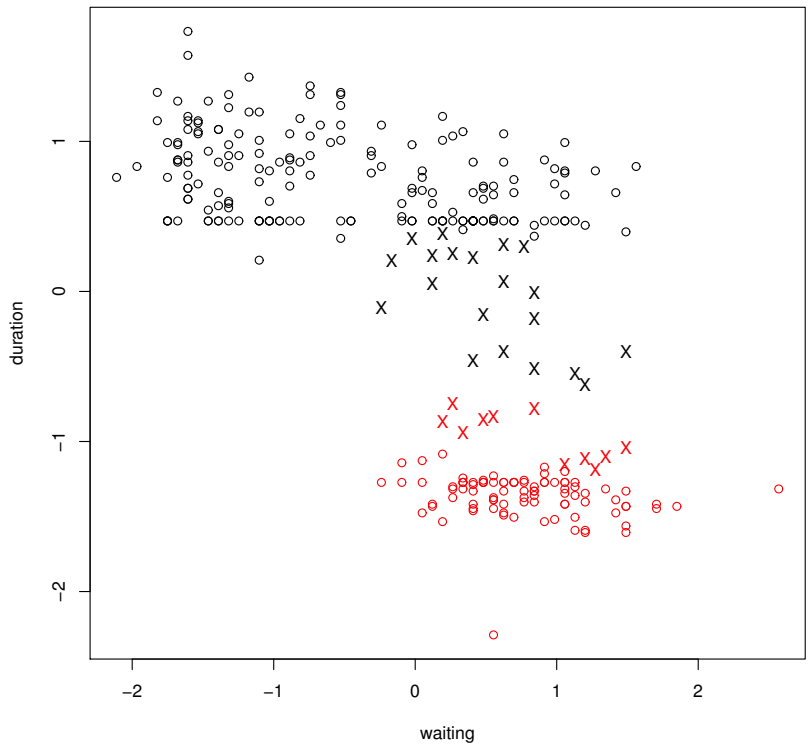
$\exists$ several ways measuring separation (as for other aims).
Straightforward: min distance between any two clusters,
or distance between centroids (e.g., k-means).

These measure quite different concepts of separation.
(min distance relies on only two points;
centroid distance ignores what goes on at border.)

***p*-separation index**:

More stable version of "min distance":

Average distance to nearest point in different cluster for $p = 10\%$ "border" points in any cluster.



Function cluster.stats in fpc computes many measures.

```
> cstrigona[[9]]
$n
[1] 236

$cluster.number
[1] 9

$cluster.size
[1]  35  23  18   4   2  11  13 126   4

$min.cluster.size
[1] 2

$noisen
[1] 0

$diameter
[1] 0.5000000 0.4090909 0.8181818 0.3846154 0.3750000 0.6250000 0.5769231
[8] 0.8461538 0.5769231

$average.distance
[1] 0.2907563 0.2648221 0.5805110 0.3461538 0.3750000 0.3679752 0.3979290
[8] 0.4790426 0.3333333
```

```
$median.distance
[1] 0.2916667 0.2727273 0.5909091 0.3653846 0.3750000 0.3333333 0.3846154
[8] 0.5384615 0.3461538

$separation
[1] 0.6666667 0.5000000 0.5000000 0.7083333 0.7083333 0.5833333 0.5000000
[8] 0.4615385 0.4615385

$average.toother
[1] 0.8922699 0.9025520 0.8876931 0.8884509 0.8508126 0.8993941 0.7739322
[8] 0.8712690 0.8242225
```

```
$separation.matrix
          [,1]      [,2]      [,3]      [,4]      [,5]      [,6]      [,7]
 [1,] 0.0000000 0.8000000 0.7000000 0.7916667 0.7272727 0.8500000 0.6666667
 [2,] 0.8000000 0.0000000 0.5000000 0.9545455 0.7727273 0.7727273 0.7272727
 [3,] 0.7000000 0.5000000 0.0000000 0.8181818 0.7272727 0.7272727 0.7272727
 [4,] 0.7916667 0.9545455 0.8181818 0.0000000 0.7916667 0.7083333 0.7307692
 [5,] 0.7272727 0.7727273 0.7272727 0.7916667 0.0000000 0.9166667 0.7500000
 [6,] 0.8500000 0.7727273 0.7272727 0.7083333 0.9166667 0.0000000 0.7083333
 [7,] 0.6666667 0.7272727 0.7272727 0.7307692 0.7500000 0.7083333 0.0000000
 [8,] 0.6666667 0.7727273 0.7272727 0.7307692 0.7083333 0.5833333 0.5000000
 [9,] 0.8750000 0.9545455 0.7272727 0.8461538 0.8750000 0.5833333 0.6538462
          [,8]      [,9]
 [1,] 0.6666667 0.8750000
 [2,] 0.7727273 0.9545455
 [3,] 0.7272727 0.7272727
 [4,] 0.7307692 0.8461538
 [5,] 0.7083333 0.8750000
 [6,] 0.5833333 0.5833333
 [7,] 0.5000000 0.6538462
 [8,] 0.0000000 0.4615385
 [9,] 0.4615385 0.0000000
```

```
$ave.between.matrix
           [,1]      [,2]      [,3]      [,4]      [,5]      [,6]      [,7]
 [1,] 0.0000000 0.9078882 0.9158730 0.8532738 0.8220779 0.9618536 0.8549451
 [2,] 0.9078882 0.0000000 0.7134387 0.9876482 0.8221344 0.8521739 0.8754941
 [3,] 0.9158730 0.7134387 0.0000000 0.9438131 0.7992424 0.8790863 0.9186092
 [4,] 0.8532738 0.9876482 0.9438131 0.0000000 0.8645833 0.7912362 0.7973373
 [5,] 0.8220779 0.8221344 0.7992424 0.8645833 0.0000000 0.9619490 0.8878205
 [6,] 0.9618536 0.8521739 0.8790863 0.7912362 0.9619490 0.0000000 0.9060712
 [7,] 0.8549451 0.8754941 0.9186092 0.7973373 0.8878205 0.9060712 0.0000000
 [8,] 0.8834451 0.9310238 0.9077375 0.8909757 0.8550174 0.9014968 0.6944096
 [9,] 0.9782738 0.9916008 0.9229798 0.8725962 0.9270833 0.7047176 0.8912722
           [,8]      [,9]
 [1,] 0.8834451 0.9782738
 [2,] 0.9310238 0.9916008
 [3,] 0.9077375 0.9229798
 [4,] 0.8909757 0.8725962
 [5,] 0.8550174 0.9270833
 [6,] 0.9014968 0.7047176
 [7,] 0.6944096 0.8912722
 [8,] 0.0000000 0.7371159
 [9,] 0.7371159 0.0000000
```

```
$average.between
[1] 0.8743582

$average.within
[1] 0.4607634

(...)

$max.diameter
[1] 0.8461538

$min.separation
[1] 0.4615385

$within.cluster.ss
[1] 24.09794
```

```
$avg.silwidth
[1] 0.3940591

$pearsongamma
[1] 0.8275561

$dunn
[1] 0.5454545

$dunn2
[1] 1.196204

$entropy
[1] 1.52252

$wb.ratio
[1] 0.5269732

$ch
[1] 54.93135

$cwidegap
[1] 0.2083333 0.2272727 0.5000000 0.3461538 0.3750000 0.4166667 0.4615385
[8] 0.3461538 0.5000000

$widestgap
[1] 0.5

$sindex
[1] 0.5083612
```

	ave.link-9	ave.link-10
ASW	0.394	0.486
Γ_P	0.828	0.825
ave.within	0.461	0.330
sindex	0.508	0.365
widestgap	0.5	0.5

Homogeneity: 10 clusters.

Separation, dissimilarity representation: 9 clusters.

2.5 Testing for clustering structure

Is a dataset *significantly* clustered?
Is a clustering with $G + 1$ clusters
significantly better than one with G?

Want to make sure that we don't cluster
something truly homogeneous.

This is however not so easy.

Testing for homogeneity

H_0: no clustering structure,
H_1: data are clustered.

Various approaches in literature (Huang et al. (2015))

Can make gap statistic into formal significance test:
reject uniform null model if

$$\begin{aligned} \mathrm{gap}(2) - \mathrm{gap}(1) &> 2\hat{\mathrm{sd}}(\log W_1 - \log W_2), \\ \hat{\mathrm{sd}}(\log W_1 - \log W_2) &= \sqrt{\hat{\mathrm{sd}}(\log W_1)^2 + \hat{\mathrm{sd}}(\log W_2)^2}. \end{aligned}$$

This is just not the case for the olive oil data:

$$\mathrm{gap}(2) - \mathrm{gap}(1) = 0.062, \ \hat{\mathrm{sd}}(\log W_1 - \log W_2) = 0.032,$$

but with $G > 2$, many $\mathrm{gap}(G) - \mathrm{gap}(1)$ are significant.

Can simulate E and sd
from simple uniform or Gaussian null models
for many validity statistics.

Issues:

- Real data are not uniform or Gaussian;
  may reject homogeneity even if still not "clustered".
- Most tests depend on alternative G,
  multiple testing issues when trying out many G.

Hennig and Lin (2015) construct more flexible null models,
adapted to specific situation.

Multiple significance tests ($G+1$ against G)
have been used in literature for estimating G
and for testing the "significance of each cluster".

Warning: issues of multiple testing
and data dependent hypotheses make
most of these approaches theoretically invalid.

If anything they're of exploratory value.

2.6 Sensitivity analyses and comparison of different clusterings on same dataset

Could apply different clustering methods
on same dataset and see whether
and which clusters coincide.

Sometimes done in literature but hard to interpret.
Issue: which methods to include and why.

Still useful for validity assessment to see whether different methods give similar results that are expected to give similar results.

Also, clustering and data preprocessing require many decisions.

Could investigate how much of a difference these decisions make to the clustering.

External validation indexes such as ARI can be used.

3. Discussion

- Clustering quality has many aspects.

3. Discussion

- Clustering quality has many aspects.
- Validity assessment is multivariate (and can be partly informal).

3. Discussion

- Clustering quality has many aspects.
- Validity assessment is multivariate
  (and can be partly informal).
- Need to decide what matters in application.

3. Discussion

- Clustering quality has many aspects.
- Validity assessment is multivariate
  (and can be partly informal).
- Need to decide what matters in application.
- Both method selection and validation
  require such decisions.

3. Discussion

- Clustering quality has many aspects.
- Validity assessment is multivariate
  (and can be partly informal).
- Need to decide what matters in application.
- Both method selection and validation
  require such decisions.
- Not all kinds of validation make sense
  to combine with all clustering methods
  (i.e., sum of squares criteria connected to k-means)
  but usually more than one criterion of interest.

Reasons for poor (invalid) clustering:

- data genuinely hard to cluster,
- wrong method or number of clusters,
- mismatch between clustering and validation method,
- bad preprocessing choices or non-choices
  (dissimilarity, variable transformation, standardisation etc.)

Could choose other methods or
change preprocessing to improve matters,
but be careful!

Preprocessing and clustering method
need to reflect clustering aim and meaning of data.

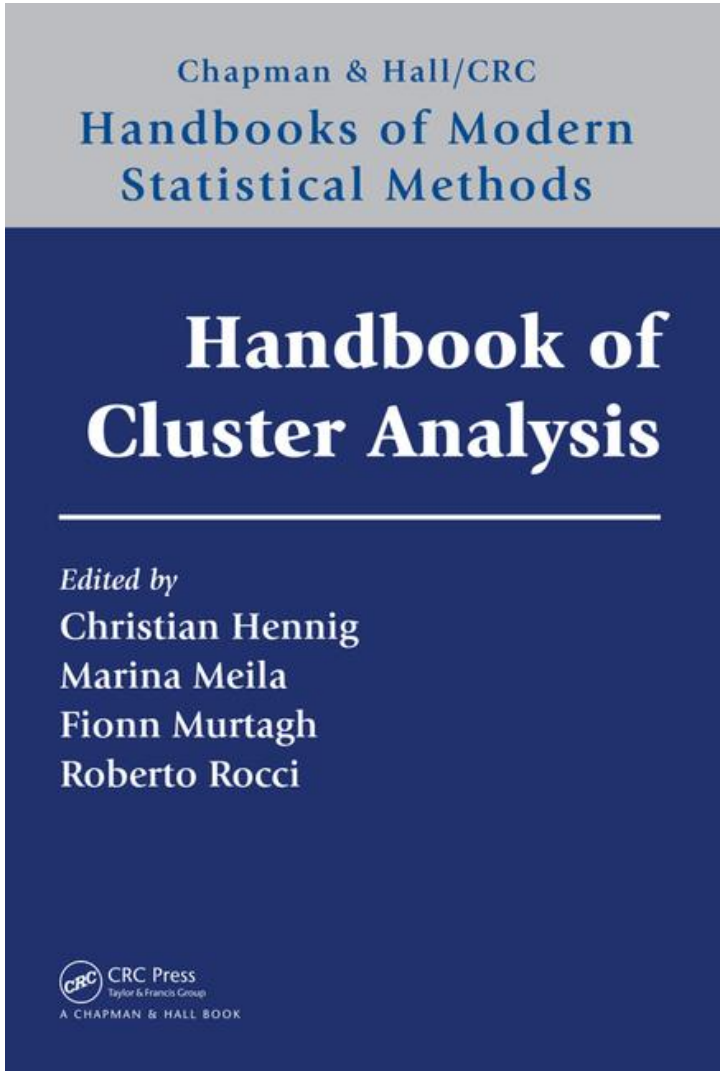
It's useless to make decisions by optimising
validation values if this is not respected.

Sometimes better accept that clusters in data
are unclear and uncertain,
rather than making them clearer by dodgy manipulations
(e.g. can induce clustering easily by some certain transformations;
k-means is often pretty stable even if cluster are not separated).

CLUSTER BENCHMARK DATA REPOSITORY

HTTP://IFCS.BOKU.AC.AT/REPOSITORY

Some more marketing:



This work is supported by EPSRC Grant EP/K033972/1.

References I

Borg, I., P. J. Groenen, and P. Mair (2012). *Applied Multidimensional Scaling*. Springer, New York.

Ester, M., H.-P. Kriegel, J. Sander, and X. Xu (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of KDD-96*, pp. 226–231. AAAI Press.

Fang, Y. and J. Wang (2012, March). Selection of the number of clusters via the bootstrap method. *Computational Statistics and Data Analysis 56*(3), 468–477.

Florek, K., J. Lukaszewicz, J. Perkal, and S. Zubrzycki (1951). Sur la liaison et la division des points dun ensemble fini. *Colloquium Mathematicae 2*, 282–285.

Halkidi, M., M. Vazirgiannis, and C. Hennig (2015). Method-independent indices for cluster validation and estimating the number of clusters. In C. Hennig, M. Meila, F. Murtagh, and R. Rocci (Eds.), *Handbook of Cluster Analysis*, Chapter 26, pp. 595–618. Chapman & Hall/CRC, Boca Raton FL.

Hennig, C. (2004). Asymmetric linear dimension reduction for classification. *Journal of Computational and Graphical Statistics 13*(4), 930–945.

Hennig, C. (2007). Cluster-wise assessment of cluster stability. *Computational Statistics and Data Analysis 52*(1), 258–271.

References II

Hennig, C. and C.-J. Lin (2015). Flexible parametric bootstrap for testing homogeneity against clustering and assessing the number of clusters. *Statistics and Computing 25*(4), 821–833.

Huang, H., Y. Liu, D. N. Hayes, A. Nobel, J. S. Marron, and C. Hennig (2015). Significance testing in clustering. In C. Hennig, M. Meila, F. Murtagh, and R. Rocci (Eds.), *Handbook of Cluster Analysis*, Chapter 15, pp. 337–360. Chapman & Hall/CRC, Boca Raton FL.

Hubert, L. and P. Arabie (1985). Comparing partitions. *Journal of Classification 2*(2), 193–218.

Hubert, L. J. and J. Schultz (1976). Quadratic assignment as a general data analysis strategy. *British Journal of Mathematical and Statistical Psychology 29*, 190–241.

Kaufman, L. and P. Rousseeuw (1990). *Finding Groups in Data*. Wiley, New York.

Pearson, K. (1894). Contributions to the mathematical theory of evolution. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences 185*, 71–110.

Rand, W. M. (1971). Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association 66*, 846–850.

References III

Rao, C. R. (1952). *Advanced Statistical Methods in Biometric Research*. Wiley, New York.

Shi, J. and J. Malik (2000). Normalized cuts and image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell. 22*(8), 888–905.

Steinhaus, H. (1956). Sur la division des corps matriels en parties. *Bulletin of the Polish Academy of Sciences 4*, 801–804.

Tibshirani, R. and G. Walther (2005). Cluster validation by prediction strength. *Journal of Computational and Graphical Statistics 14*, 511–528.

Tibshirani, R., G. Walther, and T. Hastie (2001). Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 63*(2), 411–423.