

Package ‘clustvarel’

August 4, 2009

Title Variable Selection for Model-Based Clustering

Version 1.3

Author Nema Dean <nd29c@stats.gla.ac.uk> and Adrian E. Raftery <raftery@stat.washington.edu>

Description The selection method uses either a greedy search or headlong search. The greedy search at each step either checks all variables not currently included in the set of clustering variables singly for inclusion into the set or checks all variables in the set of clustering variables singly for exclusion. The headlong search only checks until a variable is included or excluded (i.e. does not necessarily check all possible variables for inclusion/exclusion at each step) and any variable with evidence of clustering below a certain level at any stage is removed from consideration for the remainder of the algorithm. Each variable’s evidence for being useful to the clustering given the currently selected clustering variables is given by the difference between the BIC for the model with clustering (allowed to vary over 2 to a maximum number of groups and any of the different covariance parameterizations allowed in mclust) using the set of clustering variables including the variable being checked and the sum of BICs for the model with clustering (allowed to vary over 2 to a maximum number of groups and any of the different covariance parameterizations allowed in mclust) using the set of clustering variables without the variable being checked and the model for the variable being checked being conditionally independent of the clustering given the other clustering variables (this is modeled as a regression of the variable being checked on the other clustering variables).

Maintainer Nema Dean <nd29c@stats.gla.ac.uk>

Depends mclust

License GPL

Repository CRAN

Date/Publication 2009-08-04 14:23:59

R topics documented:

clustvarel	2
clvarelnosampgr	5
clvarelnosamphl	7
clvarselsampgr	10
clvarselsamphl	13

clustvarel	<i>Variable selection for Model-Based Clustering</i>
------------	--

Description

A function which uses a greedy or headlong search to find the (locally) optimal subset of variables in a dataset that have group/cluster information.

Usage

```
clustvarel(X, G, emModels1 = c("E", "V"), emModels2 = c("EII", "VII", "EEI",
  "VEI", "EVI", "VVI", "EEE", "EEV", "VEV", "VVV"), samp=FALSE,
  sampsize=2000, allow.EEE=TRUE, forcetwo=TRUE, search="greedy",
  upper=0, lower=-10, itermax=100)
```

Arguments

X	A matrix of data with rows corresponding to observations and columns (at least 2) corresponding to variables. Categorical variables are not permitted.
G	A scalar specifying the maximum number of clusters believed to be present in X.
emModels1	A vector of character strings indicating the models to be fitted in the EM phase of univariate clustering. Possible models: “E” for spherical, equal variance “V” for spherical, variable variance The default is all of the above.
emModels2	A vector of character strings indicating the models to be fitted in the EM phase of multivariate clustering. Possible models: “EII”: spherical, equal volume “VII”: spherical, unequal volume “EEI”: diagonal, equal volume, equal shape “VEI”: diagonal, varying volume, equal shape “EVI”: diagonal, equal volume, varying shape “VVI”: diagonal, varying volume, varying shape “EEE”: ellipsoidal, equal volume, shape, and orientation “EEV”: ellipsoidal, equal volume and equal shape “VEV”: ellipsoidal, equal shape “VVV”: ellipsoidal, varying volume, shape, and orientation The default is all of the above.
samp	A logical value indicating whether or not a subset of observations is to be used in the hierarchical clustering phase used to get starting values for the EM algorithm.

sampsize	The number of observations to be used in the hierarchical clustering subset.
allow.EEE	A logical value indicating whether a new clustering will be run with equal variance hierarchical clustering starting values if the clusterings with variable variance hierarchical clustering starting values do not produce any viable BIC values.
forcetwo	A logical value indicating whether at least two variables will be forced to be selected initially (regardless of whether BIC evidence suggests bivariate clustering or not).
search	A character vector indicating whether a “greedy” or potentially quicker but less optimal “headlong” algorithm is used to search for clustering variables
upper	A scalar value indicating the minimum BIC difference between clustering and no clustering used to select a clustering variable in the headlong search. Default is 0.
lower	A scalar value indicating the level of BIC difference between clustering and no clustering below which a variable will be removed from consideration in the headlong algorithm. Default is -10.
itermax	A scalar value giving the maximum number of iterations (of addition and removal steps) the algorithm is allowed to run for.

Details

The default value for ‘forcetwo’ is TRUE because often in practice there will be little evidence of clustering on the univariate or bivariate level although there is multivariate clustering present and these variables are used as starting points to attempt to find this clustering, if necessary being removed later in the algorithm.

The default value for ‘allow.EEE’ is TRUE but if necessary to speed up the algorithm it can be set to FALSE. Other speeding-up restrictions include reducing the ‘emModels1’ (to “E”, say) and the ‘emModels2’ to a smaller set of covariance parameterizations. Reducing the maximum possible number of clusters present in the data will also increase the speed of the algorithm. Another time-saving device is the ‘samp’ option which uses the same algorithm but uses only a subset of the observations in the expensive hierarchical phase of Mclust. The headlong search may be quicker than the greedy search option in data sets with large numbers of variables (depending on the values of the upper and lower bounds chosen for the BIC difference).

The defaults for the ‘eps’, ‘tol’ and ‘itmax’ options for the Mclust steps run in the algorithm can be changed by setting the variables .Mclust\$eps, .Mclust\$tol and .Mclust\$itmax respectively to new values.

Value

A list giving:

sel.var	The matrix of selected variables.
steps.info	A matrix with a row for each step of the algorithm giving: the name of the best variable proposed, the BIC of the clustering variables’ model at the end of the step, the BIC difference between clustering and not clustering for the variable,

the type of step (addition/removal),
the decision for the variable.

Author(s)

N. Dean and A. E. Raftery

References

A. E. Raftery and N. Dean (2006). Variable Selection for Model-Based Clustering, *Journal of the American Statistical Association*, Volume 101, no. 473, pp. 168-178 <http://www.stat.washington.edu/www/research/reports/2004/tr452.pdf>

J. H. Badsberg (1992). Model search in contingency tables by CoCo. In Y. Dodge and J. Whittaker (Eds.), *Computational Statistics*, Volume 1, pp. 251-256

See Also

[clvarselnosampgr](#), [clvarselsampgr](#), [clvarselnosamphl](#), [clvarselsamphl](#), [Mclust](#)

Examples

```
#Create 3-d data with 2 clusters in the first two variables and no
#clustering in the rest
X<-matrix(0,200,3)
colnames(X)<-1:3
#clusters have mixing proportion pro, means mu1 and mu2 and variances
#sigma1 and sigma2
pro<-0.5
mu1<-c(0,0)
mu2<-c(3,3)
sigma1<-matrix(c(1,0.5,0.5,1),2,2,byrow=TRUE)
sigma2<-matrix(c(1.5,-0.7,-0.7,1.5),2,2,byrow=TRUE)
u<-runif(200)
library(MASS)
for(i in 1:200)
{
  ifelse(u[i]<pro,X[i,1:2]<-mvrnorm(1,mu1,sigma1),X[i,1:2]<-mvrnorm(1,mu2,sigma2))
  X[i,3]<-rnorm(1,1.5,2)
}
#Find the clustering variables
m<-clustvarsel(X,G=3)
#Look at the names of the variables selected
colnames(m$sel.var)
m$steps.info
#look at the clustering produced by the variables selected
result<-Mclust(m$sel.var,1:3)
result
```

clvarselnosampgr *Greedy Search Variable Selection for Model-Based Clustering without hierarchical clustering sub-sampling*

Description

A function which uses a greedy search, without sub-sampling at the hierarchical clustering stage of Mclust, to find the (locally) optimal subset of variables in a dataset that have group/cluster information. This function is called by the clustvarsel function when the option ‘samp’ is set to FALSE and ‘search’ is set to “greedy”.

Usage

```
clvarselnosampgr(X, G, emModels1 = c("E", "V"), emModels2 = c("EII", "VII", "EEI",
  "VEI", "EVI", "VVI", "EEE", "EEV", "VEV", "VVV"), allow.EEE=TRUE,
  forcetwo=TRUE, itermax=100)
```

Arguments

X	A matrix of data with rows corresponding to observations and columns (at least 2) corresponding to variables. Categorical variables are not permitted.
G	A scalar specifying the maximum number of clusters believed to be present in X.
emModels1	A vector of character strings indicating the models to be fitted in the EM phase of univariate clustering. Possible models: “E” for spherical, equal variance “V” for spherical, variable variance The default is all of the above.
emModels2	A vector of character strings indicating the models to be fitted in the EM phase of multivariate clustering. Possible models: “EII”: spherical, equal volume “VII”: spherical, unequal volume “EEI”: diagonal, equal volume, equal shape “VEI”: diagonal, varying volume, equal shape “EVI”: diagonal, equal volume, varying shape “VVI”: diagonal, varying volume, varying shape “EEE”: ellipsoidal, equal volume, shape, and orientation “EEV”: ellipsoidal, equal volume and equal shape “VEV”: ellipsoidal, equal shape “VVV”: ellipsoidal, varying volume, shape, and orientation The default is all of the above.
allow.EEE	A logical value indicating whether a new clustering will be run with equal variance hierarchical clustering starting values if the clusterings with variable variance hierarchical clustering starting values do not produce any viable BIC values.

<code>forcetwo</code>	A logical value indicating whether at least two variables will be forced to be selected initially (regardless of whether BIC evidence suggests bivariate clustering or not).
<code>itermax</code>	A scalar value giving the maximum number of iterations (of addition and removal steps) the algorithm is allowed to run for.

Details

This function is called by ‘clustvarsel’ when the option ‘samp’ is set to FALSE and ‘search’ is set to “greedy”.

The default value for ‘forcetwo’ is TRUE because often in practice there will be little evidence of clustering on the univariate or bivariate level although there is multivariate clustering present and these variables are used as starting points to attempt to find this clustering, if necessary being removed later in the algorithm.

The default value for ‘allow.EEE’ is TRUE but if necessary to speed up the algorithm it can be set to FALSE. Other speeding-up restrictions include reducing the ‘emModels1’ (to “E”, say) and the ‘emModels2’ to a smaller set of covariance parameterizations. Reducing the maximum possible number of clusters present in the data will also increase the speed of the algorithm. Another time-saving device is use the function ‘clvarselsampgr’ which uses the same algorithm but uses only a subset of the observations in the expensive hierarchical phase of Mclust. The headlong search may be quicker in larger data sets (depending on the values of the upper and lower bounds chosen for the BIC difference).

The defaults for the ‘eps’, ‘tol’ and ‘itmax’ options for the Mclust steps run in the algorithm can be changed by setting the variables `.Mclust$eps`, `.Mclust$tol` and `.Mclust$itmax` respectively to new values.

Value

A list giving:

<code>sel.var</code>	The matrix of selected variables.
<code>steps.info</code>	A matrix with a row for each step of the algorithm giving: the name of the best variable proposed, the BIC of the clustering variables’ model at the end of the step, the BIC difference between clustering and not clustering for the variable, the type of step (addition/removal), the decision for the variable.

Author(s)

N. Dean and A. E. Raftery

References

A. E. Raftery and N. Dean (2006). Variable Selection for Model-Based Clustering, *Journal of the American Statistical Association*, Volume 101, no. 473, pp. 168-178 <http://www.stat.washington.edu/www/research/reports/2004/tr452.pdf>

See Also

[clustvarsel](#), [clvarselsampgr](#), [clvarselnosamphl](#), [clvarselsamphl](#), [Mclust](#)

Examples

```
#Create 3-d data with 2 clusters in the first two variables and no
#clustering in the rest
X<-matrix(0,200,3)
colnames(X)<-1:3
#clusters have mixing proportion pro, means mu1 and mu2 and variances
#sigma1 and sigma2
pro<-0.5
mu1<-c(0,0)
mu2<-c(3,3)
sigma1<-matrix(c(1,0.5,0.5,1),2,2,byrow=TRUE)
sigma2<-matrix(c(1.5,-0.7,-0.7,1.5),2,2,byrow=TRUE)
u<-runif(200)
library(MASS)
for(i in 1:200)
{
  ifelse(u[i]<pro,X[i,1:2]<-mvrnorm(1,mu1,sigma1),X[i,1:2]<-mvrnorm(1,mu2,sigma2))
  X[i,3]<-rnorm(1,1.5,2)
}
#Find the clustering variables
m<-clvarselnosampgr(X,G=3)
#Look at the names of the variables selected
colnames(m$sel.var)
m$steps.info
#look at the clustering produced by the variables selected
result<-Mclust(m$sel.var,1:3)
result
```

clvarselnosamphl	<i>Headlong Search Variable Selection for Model-Based Clustering without hierarchical clustering sub-sampling</i>
------------------	---

Description

A function which uses a headlong search, without sub-sampling at the hierarchical clustering stage of Mclust, to find the (locally) optimal subset of variables in a dataset that have group/cluster information. This function is called by the clustvarsel function when the option 'samp' is set to FALSE and 'search' is set to "headlong".

Usage

```
clvarselnosamphl(X, G, emModels1 = c("E","V"), emModels2 = c("EII","VII","EEI",
  "VEI","EVI","VVI","EEE","EEV","VEV","VVV"), allow.EEE=TRUE,
  forcetwo=TRUE, upper=0, lower=-10, itermax=100)
```

Arguments

<code>X</code>	A matrix of data with rows corresponding to observations and columns (at least 2) corresponding to variables. Categorical variables are not permitted.
<code>G</code>	A scalar specifying the maximum number of clusters believed to be present in <code>X</code> .
<code>emModels1</code>	A vector of character strings indicating the models to be fitted in the EM phase of univariate clustering. Possible models: “E” for spherical, equal variance “V” for spherical, variable variance The default is all of the above.
<code>emModels2</code>	A vector of character strings indicating the models to be fitted in the EM phase of multivariate clustering. Possible models: “EII”: spherical, equal volume “VII”: spherical, unequal volume “EEI”: diagonal, equal volume, equal shape “VEI”: diagonal, varying volume, equal shape “EVI”: diagonal, equal volume, varying shape “VVI”: diagonal, varying volume, varying shape “EEE”: ellipsoidal, equal volume, shape, and orientation “EEV”: ellipsoidal, equal volume and equal shape “VEV”: ellipsoidal, equal shape “VVV”: ellipsoidal, varying volume, shape, and orientation The default is all of the above.
<code>allow.EEE</code>	A logical value indicating whether a new clustering will be run with equal variance hierarchical clustering starting values if the clusterings with variable variance hierarchical clustering starting values do not produce any viable BIC values.
<code>forcetwo</code>	A logical value indicating whether at least two variables will be forced to be selected initially (regardless of whether BIC evidence suggests bivariate clustering or not).
<code>upper</code>	A scalar value indicating the minimum BIC difference between clustering and no clustering used to select a clustering variable. Default is 0.
<code>lower</code>	A scalar value indicating the level of BIC difference between clustering and no clustering below which a variable will be removed from consideration in the algorithm. Default is -10.
<code>itermax</code>	A scalar value giving the maximum number of iterations (of addition and removal steps) the algorithm is allowed to run for.

Details

This function is called by ‘`clustvarsel`’ when the option ‘`samp`’ is set to `FALSE` and ‘`search`’ is set to “`headlong`”.

The default value for ‘`forcetwo`’ is `TRUE` because often in practice there will be little evidence of clustering on the univariate or bivariate level although there is multivariate clustering present

and these variables are used as starting points to attempt to find this clustering, if necessary being removed later in the algorithm.

The default value for 'allow.EEE' is TRUE but if necessary to speed up the algorithm it can be set to FALSE. Other speeding-up restrictions include reducing the 'emModels1' (to "E", say) and the 'emModels2' to a smaller set of covariance parameterizations. Reducing the maximum possible number of clusters present in the data will also increase the speed of the algorithm. Another time-saving device is use the function 'clvarselsamphl' which uses the same algorithm but uses only a subsample of the observations in the expensive hierarchical phase of Mclust. The headlong search may be quicker than the greedy search in larger data sets (depending on the values of the upper and lower bounds chosen for the BIC difference). Lower values of 'upper' and higher values of 'lower' will possibly speed up the search (although they may make the solution found less optimal).

The defaults for the 'eps', 'tol' and 'itmax' options for the Mclust steps run in the algorithm can be changed by setting the variables .Mclust\$eps, .Mclust\$tol and .Mclust\$itmax respectively to new values.

Value

A list giving:

<code>sel.var</code>	The matrix of selected variables.
<code>steps.info</code>	A matrix with a row for each step of the algorithm giving: the name of the best variable proposed, the BIC of the clustering variables' model at the end of the step, the BIC difference between clustering and not clustering for the variable, the type of step (addition/removal), the decision for the variable.

Author(s)

N. Dean and A. E. Raftery

References

A. E. Raftery and N. Dean (2006). Variable Selection for Model-Based Clustering, Journal of the American Statistical Association, Volume 101, no. 473, pp. 168-178 <http://www.stat.washington.edu/www/research/reports/2004/tr452.pdf>

J. H. Badsberg (1992). Model search in contingency tables by CoCo. In Y. Dodge and J. Whittaker (Eds.), Computational Statistics, Volume 1, pp. 251-256

See Also

[clustvarsel](#), [clvarselsamphl](#), [clvarselnosampgr](#), [clvarselsampgr](#), [Mclust](#)

Examples

```
#Create 3-d data with 2 clusters in the first two variables and no
#clustering in the rest
X<-matrix(0,200,3)
```

```

colnames(X)<-1:3
#clusters have mixing proportion pro, means mu1 and mu2 and variances
#sigma1 and sigma2
pro<-0.5
mu1<-c(0,0)
mu2<-c(3,3)
sigma1<-matrix(c(1,0.5,0.5,1),2,2,byrow=TRUE)
sigma2<-matrix(c(1.5,-0.7,-0.7,1.5),2,2,byrow=TRUE)
u<-runif(200)
library(MASS)
for(i in 1:200)
{
  ifelse(u[i]<pro,X[i,1:2]<-mvrnorm(1,mu1,sigma1),X[i,1:2]<-mvrnorm(1,mu2,sigma2))
  X[i,3]<-rnorm(1,1.5,2)
}
#Find the clustering variables
m<-clvarselnosamphl(X,G=3)
#Look at the names of the variables selected
colnames(m$sel.var)
m$steps.info
#look at the clustering produced by the variables selected
result<-Mclust(m$sel.var,1:3)
result

```

clvarselsampgr	<i>Greedy Search Variable Selection for Model-Based Clustering with hierarchical clustering sub-sampling</i>
----------------	--

Description

A function which uses a greedy search, with sub-sampling at the hierarchical clustering stage of Mclust, to find the (locally) optimal subset of variables in a dataset that have group/cluster information. This function is called by the clustvarel function when the option ‘samp’ is set to TRUE and ‘search’ is set to “greedy”.

Usage

```

clvarselsampgr(X, G, emModels1 = c("E","V"), emModels2 = c("EII","VII","EEI",
  "VEI","EVI","VVI","EEE","EEV","VEV","VVV"), sampsize=2000,
  allow.EEE=TRUE, forcetwo=TRUE, itermax=100)

```

Arguments

X	A matrix of data with rows corresponding to observations and columns (at least 2) corresponding to variables. Categorical variables are not permitted.
G	A scalar specifying the maximum number of clusters believed to be present in X.

<code>emModels1</code>	A vector of character strings indicating the models to be fitted in the EM phase of univariate clustering. Possible models: “E” for spherical, equal variance “V” for spherical, variable variance The default is all of the above.
<code>emModels2</code>	A vector of character strings indicating the models to be fitted in the EM phase of multivariate clustering. Possible models: “EII”: spherical, equal volume “VII”: spherical, unequal volume “EEI”: diagonal, equal volume, equal shape “VEI”: diagonal, varying volume, equal shape “EVI”: diagonal, equal volume, varying shape “VVI”: diagonal, varying volume, varying shape “EEE”: ellipsoidal, equal volume, shape, and orientation “EEV”: ellipsoidal, equal volume and equal shape “VEV”: ellipsoidal, equal shape “VVV”: ellipsoidal, varying volume, shape, and orientation The default is all of the above.
<code>sampsize</code>	The number of observations to be used in the hierarchical clustering subset.
<code>allow.EEE</code>	A logical value indicating whether a new clustering will be run with equal variance hierarchical clustering starting values if the clusterings with variable variance hierarchical clustering starting values do not produce any viable BIC values.
<code>forcetwo</code>	A logical value indicating whether at least two variables will be forced to be selected initially (regardless of whether BIC evidence suggests bivariate clustering or not).
<code>itermax</code>	A scalar value giving the maximum number of iterations (of addition and removal steps) the algorithm is allowed to run for.

Details

This function is called by ‘clustvarsel’ when the option ‘samp’ is set to TRUE and ‘search’ is set to “greedy”.

The default value for ‘forcetwo’ is TRUE because often in practice there will be little evidence of clustering on the univariate or bivariate level although there is multivariate clustering present and these variables are used as starting points to attempt to find this clustering, if necessary being removed later in the algorithm.

The default value for ‘allow.EEE’ is TRUE but if necessary to speed up the algorithm it can be set to FALSE. Other speeding-up restrictions include reducing the ‘emModels1’ (to “E”, say) and the ‘emModels2’ to a smaller set of covariance parameterizations. Reducing the maximum possible number of clusters present in the data will also increase the speed of the algorithm. The headlong search may be quicker than the greedy search in larger data sets (depending on the values of the upper and lower bounds chosen for the BIC difference).

The defaults for the ‘eps’, ‘tol’ and ‘itmax’ options for the Mclust steps run in the algorithm can be changed by setting the variables `.Mclust$eps`, `.Mclust$tol` and `.Mclust$itmax` respectively to new values.

Value

A list giving

<code>sel.var</code>	The matrix of selected variables.
<code>steps.info</code>	A matrix with a row for each step of the algorithm giving: the name of the best variable proposed, the BIC of the clustering variables' model at the end of the step, the BIC difference between clustering and not clustering for the variable, the type of step (addition/removal), the decision for the variable.

Author(s)

N. Dean and A. E. Raftery

References

A. E. Raftery and N. Dean (2006). Variable Selection for Model-Based Clustering, *Journal of the American Statistical Association*, Volume 101, no. 473, pp. 168-178 <http://www.stat.washington.edu/www/research/reports/2004/tr452.pdf>

See Also

[clustvarsel](#), [clvarselnosampgr](#), [clvarselsamphl](#), [clvarselnosamphl](#), [Mclust](#)

Examples

```
#Create 3-d data with 2 clusters in the first two variables and no
#clustering in the rest
X<-matrix(0,200,3)
colnames(X)<-1:3
#clusters have mixing proportion pro, means mu1 and mu2 and variances
#sigma1 and sigma2
pro<-0.5
mu1<-c(0,0)
mu2<-c(3,3)
sigma1<-matrix(c(1,0.5,0.5,1),2,2,byrow=TRUE)
sigma2<-matrix(c(1.5,-0.7,-0.7,1.5),2,2,byrow=TRUE)
u<-runif(200)
library(MASS)
for(i in 1:200)
{
  ifelse(u[i]<pro,X[i,1:2]<-mvrnorm(1,mu1,sigma1),X[i,1:2]<-mvrnorm(1,mu2,sigma2))
  X[i,3]<-rnorm(1,1.5,2)
}
#look at the data
#pairs(X)
#Find the clustering variables
m<-clvarselsampgr(X,G=3,sampsize=100)
#Look at the names of the variables selected
```

```

colnames(m$sel.var)
m$steps.info
#look at the clustering produced by the variables selected
result<-Mclust(m$sel.var,1:3)
result

```

clvarselsamphl	<i>Headlong Search Variable Selection for Model-Based Clustering with hierarchical clustering sub-sampling</i>
----------------	--

Description

A function which uses a headlong search, with sub-sampling at the hierarchical clustering stage of Mclust, to find the (locally) optimal subset of variables in a dataset that have group/cluster information. This function is called by the clustvarel function when the option ‘samp’ is set to TRUE and ‘search’ is set to “headlong”.

Usage

```

clvarselsamphl(X, G, emModels1 = c("E", "V"), emModels2 = c("EII", "VII", "EEI",
  "VEI", "EVI", "VVI", "EEE", "EEV", "VEV", "VVV"), sampsize=2000,
  allow.EEE=TRUE, forcetwo=TRUE, upper=0, lower=-10, itermax=100)

```

Arguments

X	A matrix of data with rows corresponding to observations and columns (at least 2) corresponding to variables. Categorical variables are not permitted.
G	A scalar specifying the maximum number of clusters believed to be present in X.
emModels1	A vector of character strings indicating the models to be fitted in the EM phase of univariate clustering. Possible models: “E” for spherical, equal variance “V” for spherical, variable variance The default is all of the above.
emModels2	A vector of character strings indicating the models to be fitted in the EM phase of multivariate clustering. Possible models: “EII”: spherical, equal volume “VII”: spherical, unequal volume “EEI”: diagonal, equal volume, equal shape “VEI”: diagonal, varying volume, equal shape “EVI”: diagonal, equal volume, varying shape “VVI”: diagonal, varying volume, varying shape “EEE”: ellipsoidal, equal volume, shape, and orientation “EEV”: ellipsoidal, equal volume and equal shape “VEV”: ellipsoidal, equal shape

	“VVV”: ellipsoidal, varying volume, shape, and orientation The default is all of the above.
sampsize	The number of observations to be used in the hierarchical clustering subset.
allow.EEE	A logical value indicating whether a new clustering will be run with equal variance hierarchical clustering starting values if the clusterings with variable variance hierarchical clustering starting values do not produce any viable BIC values.
forcetwo	A logical value indicating whether at least two variables will be forced to be selected initially (regardless of whether BIC evidence suggests bivariate clustering or not).
upper	A scalar value indicating the minimum BIC difference between clustering and no clustering used to select a clustering variable. Default is 0.
lower	A scalar value indicating the level of BIC difference between clustering and no clustering below which a variable will be removed from consideration in the algorithm. Default is -10.
itermax	A scalar value giving the maximum number of iterations (of addition and removal steps) the algorithm is allowed to run for.

Details

This function is called by ‘clustvarsel’ when the option ‘samp’ is set to TRUE and ‘search’ is set to “headlong”.

The default value for ‘forcetwo’ is TRUE because often in practice there will be little evidence of clustering on the univariate or bivariate level although there is multivariate clustering present and these variables are used as starting points to attempt to find this clustering, if necessary being removed later in the algorithm.

The default value for ‘allow.EEE’ is TRUE but if necessary to speed up the algorithm it can be set to FALSE. Other speeding-up restrictions include reducing the ‘emModels1’ (to “E”, say) and the ‘emModels2’ to a smaller set of covariance parameterizations. Reducing the maximum possible number of clusters present in the data will also increase the speed of the algorithm. The headlong search may be quicker than the greedy search in larger data sets (depending on the values of the upper and lower bounds chosen for the BIC difference). Lower values of ‘upper’ and higher values of ‘lower’ will possibly speed up the search (although they may make the solution found less optimal).

The defaults for the ‘eps’, ‘tol’ and ‘itmax’ options for the Mclust steps run in the algorithm can be changed by setting the variables .Mclust\$eps, .Mclust\$tol and .Mclust\$itmax respectively to new values.

Value

A list giving:

sel.var	The matrix of selected variables.
steps.info	A matrix with a row for each step of the algorithm giving: the name of the best variable proposed, the BIC of the clustering variables’ model at the end of the step, the BIC difference between clustering and not clustering for the variable,

the type of step (addition/removal),
the decision for the variable.

Author(s)

N. Dean and A. E. Raftery

References

A. E. Raftery and N. Dean (2006). Variable Selection for Model-Based Clustering, *Journal of the American Statistical Association*, Volume 101, no. 473, pp. 168-178 <http://www.stat.washington.edu/www/research/reports/2004/tr452.pdf>

J. H. Badsberg (1992). Model search in contingency tables by CoCo. In Y. Dodge and J. Whittaker (Eds.), *Computational Statistics*, Volume 1, pp. 251-256

See Also

[clustvarsel](#), [clvarselnosamphl](#), [clvarselsampgr](#), [clvarselnosampgr](#), [Mclust](#)

Examples

```
#Create 3-d data with 2 clusters in the first two variables and no
#clustering in the rest
X<-matrix(0,200,3)
colnames(X)<-1:3
#clusters have mixing proportion pro, means mu1 and mu2 and variances
#sigma1 and sigma2
pro<-0.5
mu1<-c(0,0)
mu2<-c(3,3)
sigma1<-matrix(c(1,0.5,0.5,1),2,2,byrow=TRUE)
sigma2<-matrix(c(1.5,-0.7,-0.7,1.5),2,2,byrow=TRUE)
u<-runif(200)
library(MASS)
for(i in 1:200)
{
  ifelse(u[i]<pro,X[i,1:2]<-mvrnorm(1,mu1,sigma1),X[i,1:2]<-mvrnorm(1,mu2,sigma2))
  X[i,3]<-rnorm(1,1.5,2)
}
#look at the data
#pairs(X)
#Find the clustering variables
m<-clvarselsamphl(X,G=3,sampsize=100)
#Look at the names of the variables selected
colnames(m$sel.var)
m$steps.info
#look at the clustering produced by the variables selected
result<-Mclust(m$sel.var,1:3)
result
```

Index

*Topic **cluster**

clustvarsel, [2](#)

clvarselnosampgr, [5](#)

clvarselnosamphl, [7](#)

clvarselsampgr, [10](#)

clvarselsamphl, [13](#)

clustvarsel, [2](#), [7](#), [9](#), [12](#), [15](#)

clvarselnosampgr, [4](#), [5](#), [9](#), [12](#), [15](#)

clvarselnosamphl, [4](#), [7](#), [7](#), [12](#), [15](#)

clvarselsampgr, [4](#), [7](#), [9](#), [10](#), [15](#)

clvarselsamphl, [4](#), [7](#), [9](#), [12](#), [13](#)

Mclust, [4](#), [7](#), [9](#), [12](#), [15](#)