Package 'CoClust'

April 11, 2025

Title A Copula-Based Clustering Algorithm

Date 2025-04-11

Version 1.0-0

Author Francesca Marta Lilja Di Lascio [aut, cre],
Simone Giannerini [aut]

Depends R (>= 2.15.1), methods, copula

Imports gtools

Description
A copula based clustering algorithm that finds clusters according to the complex multivariate dependence structure of the data generating process. The updated version of the algorithm is described in Di Lascio, F.M.L. and Giannerini, S. (2019). ``Clustering dependent observations with copula functions''. Statistical Papers, 60, p.35-51. <doi:10.1007/s00362-016-0822-3>.

Maintainer Francesca Marta Lilja Di Lascio <marta.dilascio@unibz.it>

License GPL (>= 2)

NeedsCompilation no

Repository CRAN

Date/Publication 2025-04-11 11:20:09 UTC

Contents

CoClust   *A copula-based clustering algorithm*

Description

Cluster analysis based on copula functions

Usage

```
CoClust(m, dimset = 2:50, noc = 4, copula = "frank", fun = median,
  method.ma = c("empirical", "pseudo"), method.c = c("ml", "mpl", "irho", "itau"),
    dfree = NULL, writeout = 5, penalty = c("BICk", "AICk", "LL"), ...)
```

Arguments

m	a data matrix.
dimset	the set of dimensions for which the function tries the clustering.
noc	sample size of the set for selecting the number of clusters.
copula	a copula model. This should be one of "normal", "t", "frank", "clayton" and "gumbel". See the Details section.
fun	combination function of the pairwise Spearman's rho used to select the k-plets. The default is median
method.ma	estimation method for margins. See the Details section.
method.c	estimation method for copula. See fitCopula.
dfree	degrees of freedom for the *t* copula.
writeout	writes a message on the number of allocated observations every writeout observations.
penalty	Specifies the likelihood criterion used for selecting the number of clusters.
...	further parameters for fitCopula.

Details

Usage for Frank copula: `CoClust(m, nmaxmarg = 2:50, noc = 4, copula = "frank", fun = median, method.ma=c("gaussian","empirical"), method.c = "mpl", penalty ="BICk", ...)`

CoClust is a clustering algorithm that, being based on copula functions, allows to group observations according to the multivariate dependence structure of the generating process without any assumptions on the margins.

For each k in dimset the algorithm builds a sample of noc observations (rows of the data matrix m) by using the matrix of Spearman's rho correlation coefficients which are combined by means of the function fun (median by default). The number of clusters K is selected by means of a criterion based on the likelihood of the copula fit. The switch penalty allows to select 3 different criteria; The choice LL corresponds to using the likelihood without penalty terms. Then, the remaining

observations are allocated to the clusters as follows: 1. selects a K-plet of observations on the basis of fun applied to the pairwise Spearman's rho; 2. allocates or discards the K-plet on the basis of the likelihood of the copula fit.

The estimation approach for the copula fit is semiparametric: a range of nonparametric margins and parametric copula models can be selected by the user. The CoClust algorithm does not require to set a priori the number of clusters nor it needs a starting classification.

Notice that the dependence structure for the Gaussian and the t copula is set to exchangeable. Non structured dependence structures will be allowed in a future version.

Value

An object of S4 class "CoClust", which is a list with the following elements:

Number.of.Clusters
> the number K of identified clusters.

Index.Matrix a *n.obs* by *(K+1)* matrix where *n.obs* is the number of observations put in each cluster. The matrix contains the row indexes of the observations of the data matrix m. The last column contains the log-likelihood of the copula fit.

Data.Clusters the matrix of the final clustering.

Dependence a list containing:

Model	the copula model used for the clustering.
Param	the estimated dependence parameter between clusters.
Std.Err	the standard error of Param.
P.val	the p-value associated to the null hypothesis H_0: theta=0.

LogLik the maximized log-likelihood copula fit.

Est.Method the estimation method used for the copula fit.

Opt.Method the optimization method used for the copula fit.

LLC the value of the LogLikelihood Criterion for each k in dimset.

Index.dimset a list that, for each k in dimset, contains the index matrix of the initial set of nk observations used for selecting the number of clusters, together with the associated loglikelihood.

Note

The final clustering is composed of K groups in which observations of the same group are independent whereas the observations that belong to different groups and that form a K-plet are dependent.

Author(s)

Francesca Marta Lilja Di Lascio <marta.dilascio@unibz.it>,

Simone Giannerini <simone.giannerini@unibo.it>

References

Di Lascio, F.M.L. and Giannerini, S. (2019). "Clustering dependent observations with copula functions". *Statistical Papers*, 60, p.35-51. DOI 10.1007/s00362-016-0822-3.

Di Lascio, F.M.L. (2018) "CoClust: An R Package for Copula-based Cluster Analysis", Recent Applications in Data Clustering, p.93-114. Ed. Harun Pirim, IntTech Publisher. ISBN 978-1-78923-527-2. DOI 10.5772/intechopen.74865.

Di Lascio, F.M.L., Durante, F. and Pappada', R. (2017). "Copula-based clustering methods", Copulas and Dependence Models with Applications, p.49-67. Eds Ubeda-Flores, M., de Amo, E., Durante, F. and Fernandez Sanchez, J., Springer International Publishing. ISBN 978-3-319-64220-8.

Di Lascio, F.M.L. and Giannerini, S. (2016). "Clustering dependent observations with copula functions". *Statistical Papers*, p.1-17. DOI 10.1007/s00362-016-0822-3.

Di Lascio, F.M.L. and Giannerini, S. (2012). "A Copula-Based Algorithm for Discovering Patterns of Dependent Observations", *Journal of Classification*, 29(1), p.50-75.

Di Lascio, F.M.L. (2008). "Analyzing the dependence structure of microarray data: a copula-based approach". *PhD thesis*, Dipartimento di Scienze Statistiche, Universita' di Bologna, Italy.

Examples

```
## ********************************************************************
## 1. builds a 3-variate copula with different margins
##    (Gaussian, Gamma, Beta)
##
## 2. generates a data matrix xm with 15 rows and 21 columns and
##    builds the matrix of the true cluster indexes
##
## 3. applies the CoClust to the rows of xm and recovers the
##    multivariate dependence structure of the data
## ********************************************************************

## Step 1. **********************************************************
n     <- 105            # total number of observations
n.col <- 21             # number of columns of the data matrix m
n.marg <- 3             # dimension of the copula
n.row <- n*n.marg/n.col # number of rows of the data matrix m

theta <- 10
copula <- frankCopula(theta, dim = n.marg)
mymvdc <- mvdc(copula, c("norm", "gamma", "beta"),list(list(mean=7, sd=2),
              list(shape=3, rate=4), list(shape1=2, shape2=1)))

## Step 2. **********************************************************
set.seed(11)
x.samp <- rMvdc(n, mymvdc)
xm     <- matrix(x.samp, nrow = n.row, ncol = n.col, byrow=TRUE)

index.true <-  matrix(1:15,5,3)
colnames(index.true) <- c("Cluster 1","Cluster 2", "Cluster 3")
```

```
## Step 3. *************************************************************

clust <- CoClust(xm, dimset = 2:4, noc=2, copula="frank",
                 method.ma="empirical", method.c="ml",writeout=1)
clust
clust@"Number.of.Clusters"
clust@"Dependence"$Param
clust@"Data.Clusters"
index.clust <- clust@"Index.Matrix"

## compare with index.true
index.clust
index.true
##
```

CoClust-class   *Class "CoClust"*

Description

A class for `CoClust` and its extensions

Objects from the Class

Objects can be created by calls of the form `new("CoClust", ...)`.

Slots

Number.of.Clusters: Object of class `"integer"`. The number K of identified clusters.

Index.Matrix: Object of class `"matrix"`. A *n.obs* by *(K+1)* matrix where *n.obs* is the number of observations put in each cluster. The matrix contains the row indexes of the observations of the data matrix `m`. The last column contains the log-likelihood of the copula fit.

Data.Clusters: Object of class `"matrix"`. The matrix of the final clustering.

Dependence: Object of class `"list"`. The list contains:

   Model the copula model used for the clustering.

   Param the estimated dependence parameter between clusters.

   Std.Err the standard error of Param.

   P.val the p-value associated to the null hypothesis H_0: theta=0.

LogLik: Object of class `"numeric"`. The maximized log-likelihood copula fit.

Est.Method: Object of class `"character"`. The estimation method used for the copula fit.

Opt.Method: Object of class `"character"`. The optimization method used for the copula fit.

LLC: Object of class `"numeric"`. The value of the LogLikelihood Criterion for each k in dimset.

Index.dimset: Object of class `"list"`. A list that, for each k in `dimset`, contains the index matrix of the initial set of nk observations used for selecting the number of clusters, together with the associated loglikelihood.

Methods

No methods defined with class "CoClust" in the signature.

Author(s)

Francesca Marta Lilja Di Lascio <marta.dilascio@unibz.it>,

Simone Giannerini <simone.giannerini@unibo.it>

References

Di Lascio, F.M.L. and Giannerini, S. (2019). "Clustering dependent observations with copula functions". *Statistical Papers*, 60, p.35-51. DOI 10.1007/s00362-016-0822-3.

Di Lascio, F.M.L. (2018) "CoClust: An R Package for Copula-based Cluster Analysis", Recent Applications in Data Clustering, p.93-114. Ed. Harun Pirim, IntTech Publisher. ISBN 978-1-78923-527-2. DOI 10.5772/intechopen.74865.

Di Lascio, F.M.L., Durante, F. and Pappada', R. (2017). "Copula-based clustering methods", Copulas and Dependence Models with Applications, p.49-67. Eds Ubeda-Flores, M., de Amo, E., Durante, F. and Fernandez Sanchez, J., Springer International Publishing. ISBN: 978-3-319-64220-8.

Di Lascio, F.M.L. and Disegna, M. (2017). "A copula-based clustering algorithm to analyse EU country diets". *Knowledge-Based Systems*, 132, p.72-84. DOI: 10.1016/j.knosys.2017.06.004.

Di Lascio, F.M.L. and Giannerini, S. (2012). "A Copula-Based Algorithm for Discovering Patterns of Dependent Observations", *Journal of Classification*, 29(1), p.50-75.

Di Lascio, F.M.L. (2008). "Analyzing the dependence structure of microarray data: a copula-based approach". *PhD thesis*, Dipartimento di Scienze Statistiche, Universita' di Bologna, Italy.

See Also

See Also `CoClust`.

Examples

```
showClass("CoClust")
```

Index