Package ‘FactorCopula’

March 5, 2023

Version 0.9.3

Date 2023-03-4

Title Factor, Bi-Factor, Second-Order and Factor Tree Copula Models

Author Sayed H. Kadhem [aut],
Aristidis K. Nikoloulopoulos [aut, cre]

Maintainer Aristidis K. Nikoloulopoulos <a.nikoloulopoulos@uea.ac.uk>

Depends R (>= 3.5.0), statmod, abind, utils

Imports igraph, matlab, polycor, VineCopula

Description
Estimation, model selection and goodness-of-fit of (1) factor copula models for mixed continuous and discrete data in Kadhem and Nikoloulopoulos (2021) <doi:10.1111/bmsp.12231>; (2) bi-factor and second-order copula models for item response data in Kadhem and Nikoloulopoulos (2023) <doi:10.1007/s11336-022-09894-2>; (3) factor tree copula models for item response data in Kadhem and Nikoloulopoulos (2022) <arXiv:2201.00339>.

NeedsCompilation yes

Repository CRAN

Date/Publication 2023-03-05 20:00:02 UTC

R topics documented:

`FactorCopula-package` *Factor, bi-factor, second-order and factor tree copula models*

Description

Estimation, model selection and goodness-of-fit of (1) factor copula models for mixed continuous and discrete data in Kadhem and Nikoloulopoulos (2021); (2) bi-factor and second-order copula models for item response data in Kadhem and Nikoloulopoulos (2023); (3) factor tree copula models for item response data in Kadhem and Nikoloulopoulos (2022).

Details

This package contains R functions for:

- diagnostics based on semi-correlations (Kadhem and Nikoloulopoulos, 2021, 2023; Joe, 2014) to detect tail dependence or tail asymmetry;
- diagnostics to show that a dataset has a factor structure based on linear factor analysis (Kadhem and Nikoloulopoulos, 2021,2023; Joe, 2014);
- estimation of the factor copula models in Krupskii and Joe (2013), Nikoloulopoulos and Joe (2015), and Kadhem and Nikoloulopoulos (2021);
- estimation of the bi-factor and second-order copula models for item response data in Kadhem and Nikoloulopoulos (2023);
- estimation of the factor tree copula models for item response data in Kadhem and Nikoloulopoulos (2022);
- model selection of the factor copula models in Krupskii and Joe (2013), Nikoloulopoulos and Joe (2015);
- model selection of the bifactor and second-order copula models in Kadhem and Nikoloulopoulos (2023);
- model selection of the factor tree copula models in Kadhem and Nikoloulopoulos (2022);

- goodness-of-fit of the factor copula models in Krupskii and Joe (2013), Nikoloulopoulos and Joe (2015), and Kadhem and Nikoloulopoulos (2021) using the M_2 statistic (Maydeu-Olivares and Joe, 2006). Note that the continuous and count data have to be transformed to ordinal;
- goodness-of-fit of the bi-factor and second-order copula models in Kadhem and Nikoloulopoulos (2023) using the M_2 statistic (Maydeu-Olivares and Joe, 2006).

Author(s)

Sayed H. Kadhem
Aristidis K. Nikoloulopoulos <a.nikoloulopoulos@uea.ac.uk>

References

Joe, H. (2014). *Dependence Modelling with Copulas*. Chapman & Hall, London.

Maydeu-Olivares, A. and Joe, H. (2006). Limited information goodness-of-fit testing in multidimensional contingency tables. *Psychometrika*, **71**, 713–732. doi:10.1007/s1133600512959.

Kadhem, S.H. and Nikoloulopoulos, A.K. (2021) Factor copula models for mixed data. *British Journal of Mathematical and Statistical Psychology*, **74**, 365–403. doi:10.1111/bmsp.12231.

Kadhem, S.H. and Nikoloulopoulos, A.K. (2023) Bi-factor and second-order copula models for item response data. *Psychometrika*, **88**, 132–157. doi:10.1007/s11336022098942.

Kadhem, S.H. and Nikoloulopoulos, A.K. (2022) Factor tree copula models for item response data. *Arxiv e-prints*, <arXiv: 2201.00339>. `https://arxiv.org/abs/2201.00339`.

Krupskii, P. and Joe, H. (2013) Factor copula models for multivariate data. *Journal of Multivariate Analysis*, **120**, 85–101. doi:10.1016/j.jmva.2013.05.001.

Nikoloulopoulos, A.K. and Joe, H. (2015) Factor copula models with item response data. *Psychometrika*, **80**, 126–150. doi:10.1007/s1133601393874.

`discrepancy` *Diagnostics to detect a factor dependence structure*

Description

The diagnostic method in Joe (2014, pages 245-246) to show that each dataset has a factor structure based on linear factor analysis. The correlation matrix $\mathsf{R}_{\text{observed}}$ has been obtained based on the sample correlations from the bivariate pairs of the observed variables. These are the linear (when both variables are continuous), polychoric (when both variables are ordinal), and polyserial (when one variable is continuous and the other is ordinal) sample correlations among the observed variables. The resulting $\mathsf{R}_{\text{observed}}$ is generally positive definite if the sample size is not small enough; if not one has to convert it to positive definite. We calculate various measures of discrepancy between $\mathsf{R}_{\text{observed}}$ and $\mathsf{R}_{\text{model}}$ (the resulting correlation matrix of linear factor analysis), such as the maximum absolute correlation difference $D_1 = \max|\mathsf{R}_{\text{model}} - \mathsf{R}_{\text{observed}}|$, the average absolute correlation difference $D_2 = \text{avg}|\mathsf{R}_{\text{model}} - \mathsf{R}_{\text{observed}}|$, and the correlation matrix discrepancy measure $D_3 = \log\big(\det(\mathsf{R}_{\text{model}})\big) - \log\big(\det(\mathsf{R}_{\text{observed}})\big) + \text{tr}(\mathsf{R}_{\text{model}}^{-1}\mathsf{R}_{\text{observed}}) - d$.

Usage

```
discrepancy(cormat, n, f3)
```

Arguments

cormat	$\mathsf{R}_{\text{observed}}$.
n	Sample size.
f3	If TRUE, then the linear 3-factor analysis is fitted.

Value

A matrix with the calculated discrepancy measures for different number of factors.

Author(s)

Sayed H. Kadhem <s.kadhem@uea.ac.uk>
Aristidis K. Nikoloulopoulos <a.nikoloulopoulos@uea.ac.uk>

References

Joe, H. (2014). *Dependence Modelling with Copulas*. Chapman & Hall, London.

Kadhem, S.H. and Nikoloulopoulos, A.K. (2021) Factor copula models for mixed data. *British Journal of Mathematical and Statistical Psychology*, **74**, 365–403. doi:10.1111/bmsp.12231.

Examples

```
#------------------------------------------------
#                     PE Data
#------------------             -----------------
data(PE)
#correlation
continuous.PE1 <- -PE[,1]
continuous.PE <- cbind(continuous.PE1, PE[,2])
u.PE <- apply(continuous.PE, 2, rank)/(nrow(PE)+1)
z.PE <- qnorm(u.PE)
categorical.PE <- data.frame(apply(PE[, 3:5], 2, factor))
nPE <- cbind(z.PE, categorical.PE)

#-------------------------------------------------
# Discrepancy measures----------------------------
#-------------------------------------------------
#correlation matrix for mixed data
cormat.PE <- as.matrix(polycor::hetcor(nPE, std.err=FALSE))
#discrepancy measures
out.PE = discrepancy(cormat.PE, n = nrow(nPE), f3 = FALSE)

#------------------------------------------------
#------------------------------------------------
#                    GSS Data
#------------------             -----------------
```

```
data(GSS)
attach(GSS)
continuous.GSS <- cbind(INCOME,AGE)
continuous.GSS <- apply(continuous.GSS, 2, rank)/(nrow(GSS)+1)
z.GSS <- qnorm(continuous.GSS)
ordinal.GSS <- cbind(DEGREE,PINCOME,PDEGREE)
count.GSS <- cbind(CHILDREN,PCHILDREN)

# Transforming the count variables to ordinal
# count1 : CHILDREN
count1  = count.GSS[,1]
count1[count1 > 3] = 3

# count2: PCHILDREN
count2  = count.GSS[,2]
count2[count2 > 7] = 7

# Combining both transformed count variables
ncount.GSS = cbind(count1, count2)

# Combining ordinal and transformed count variables
categorical.GSS <- cbind(ordinal.GSS, ncount.GSS)
categorical.GSS <- data.frame(apply(categorical.GSS, 2, factor))

# combining continuous and categorical variables
nGSS = cbind(z.GSS, categorical.GSS)

#-------------------------------------------------
# Discrepancy measures----------------------------
#-------------------------------------------------
#correlation matrix for mixed data
cormat.GSS <- as.matrix(polycor::hetcor(nGSS, std.err=FALSE))
#discrepancy measures
out.GSS = discrepancy(cormat.GSS, n = nrow(nGSS), f3 = TRUE)
```

GSS *The 1994 General Social Survey*

Description

Hoff (2007) analysed seven demographic variables of 464 male respondents to the 1994 General Social Survey. Of these seven, two were continuous (income and age of the respondents), three were ordinal with 5 categories (highest degree of the survey respondent, income and highest degree of respondent's parents), and two were count variables (number of children of the survey respondent and respondent's parents).

Usage

```
data(GSS)
```

Format

A data frame with 464 observations on the following 7 variables:

`INCOME` Income of the respondent in 1000s of dollars, binned into 21 ordered categories.

`DEGREE` Highest degree ever obtained (0:None, 1:HS, 2:Associates, 3:Bachelors, 4:Graduate).

`CHILDREN` Number of children of the survey respondent.

`PINCOME` Financial status of respondent's parents when respondent was 16 (on a 5-point scale).

`PDEGREE` Highest degree of the survey respondent's parents (0:None, 1:HS, 2:Associates, 3:Bachelors, 4:Graduate).

`PCHILDREN` Number of children of the survey respondent's parents - 1.

`AGE` Age of the respondents in years.

Source

Hoff, P. D. (2007). Extending the rank likelihood for semiparametric copula estimation. *The Annals of Applied Statistics*, **1**, 265–283.

`M2.Factor` *Goodness-of-fit of factor copula models for mixed data*

Description

The limited information M_2 statistic (Maydeu-Olivares and Joe, 2006) of factor copula models for mixed continuous and discrete data.

Usage

```
M2.1F(tcontinuous, ordinal, tcount, cpar, copF1, gl)
M2.2F(tcontinuous, ordinal, tcount, cpar, copF1, copF2, gl, SpC)
```

Arguments

`tcontinuous`	$n \times d_1$ matrix with the transformed continuous to ordinal reponse data, where n and d_1 is the number of observations and transformed continous variables, respectively.
`ordinal`	$n \times d_2$ matrix with the ordinal reponse data, where n and d_2 is the number of observations and ordinal variables, respectively.
`tcount`	$n \times d_3$ matrix with the transformed count to ordinal reponse data, where n and d_3 is the number of observations and transformed count variables, respectively.
`cpar`	A list of estimated copula parameters.

copF1 $(d_1 + d_2 + d_3)$-vector with the names of bivariate copulas that link the each of the oberved variabels with the 1st factor. Choices are “bvn” for BVN, “bvtν” with $\nu = \{1, \ldots, 9\}$ degrees of freedom for t-copula, “frk” for Frank, “gum” for Gumbel, “rgum” for reflected Gumbel, “1rgum” for 1-reflected Gumbel, “2rgum” for 2-reflected Gumbel, “joe” for Joe, “rjoe” for reflected Joe, “1rjoe” for 1-reflected Joe, “2rjoe” for 2-reflected Joe, “BB1” for BB1, “rBB1” for reflected BB1, “BB7” for BB7, “rBB7” for reflected BB7,“BB8” for BB8, “rBB8” for reflected BB8, “BB10” for BB10, “rBB10” for reflected BB10.

copF2 $(d_1 + d_2 + d_3)$-vector with the names of bivariate copulas that link the each of the oberved variabels with the 2nd factor. Choices are “bvn” for BVN, “bvtν” with $\nu = \{1, \ldots, 9\}$ degrees of freedom for t-copula, “frk” for Frank, “gum” for Gumbel, “rgum” for reflected Gumbel, “1rgum” for 1-reflected Gumbel, “2rgum” for 2-reflected Gumbel, “joe” for Joe, “rjoe” for reflected Joe, “1rjoe” for 1-reflected Joe, “2rjoe” for 2-reflected Joe, “BB1” for BB1, “rBB1” for reflected BB1, “BB7” for BB7, “rBB7” for reflected BB7,“BB8” for BB8, “rBB8” for reflected BB8, “BB10” for BB10, “rBB10” for reflected BB10.

gl Gauss legendre quardrature nodes and weights.

SpC Special case for the 2-factor copula model with BVN copulas. Select a bivariate copula at the 2nd factor to be fixed to independence. e.g. "SpC = 1" to set the first copula at the 2nd factor to independence.

Details

The M_2 statistic has been developed for goodness-of-fit testing in multidimensional contingency tables by Maydeu-Olivares and Joe (2006). Nikoloulopoulos and Joe (2015) have used the M_2 statistic to assess the goodness-of-fit of factor copula models for ordinal data. We build on the aforementioned papers and propose a methodology to assess the overall goodness-of-fit of factor copula models for mixed continuous and discrete responses. Since the M_2 statistic has been developed for multivariate ordinal data, we propose to first transform the continuous and count variables to ordinal and then calculate the M_2 statistic at the maximum likelihood estimate before transformation.

Value

A list containing the following components:

M2 The M_2 statistic which has a null asymptotic distribution that is χ^2 with $s - q$ degrees of freedom, where s is the number of univariate and bivariate margins that do not include the category 0 and q is the number of model parameters.

df $s - q$.

p-value The resultant p-value.

Author(s)

Sayed H. Kadhem <s.kadhem@uea.ac.uk>
Aristidis K. Nikoloulopoulos <a.nikoloulopoulos@uea.ac.uk>

References

Kadhem, S.H. and Nikoloulopoulos, A.K. (2021) Factor copula models for mixed data. *British Journal of Mathematical and Statistical Psychology*, **74**, 365–403. doi:10.1111/bmsp.12231.

Maydeu-Olivares, A. and Joe, H. (2006). Limited information goodness-of-fit testing in multidimensional contingency tables. *Psychometrika*, **71**, 713–732. doi:10.1007/s1133600512959.

Nikoloulopoulos, A.K. and Joe, H. (2015) Factor copula models with item response data. *Psychometrika*, **80**, 126–150. doi:10.1007/s1133601393874.

Examples

```
#------------------------------------------------
# Setting quadreture points
nq <- 25
gl <- gauss.quad.prob(nq)
#------------------------------------------------
#                    PE Data
#------------------             -----------------
data(PE)
continuous.PE1 = -PE[,1]
continuous.PE2 = PE[,2]
continuous.PE <- cbind(continuous.PE1, continuous.PE2)

categorical.PE <- PE[, 3:5]
#------------------------------------------------
#                  Estimation
#------------------             -----------------
#------------------ One-factor  -----------------
# one-factor copula model
cop1f.PE <- c("joe", "joe", "rjoe", "joe", "gum")
est1factor.PE <- mle1factor(continuous.PE, categorical.PE,
                            count=NULL, copF1=cop1f.PE, gl, hessian = TRUE)
#------------------------------------------------
#                       M2
#------------------------------------------------
#Transforming the continuous to ordinal data:
ncontinuous.PE = continuous2ordinal(continuous.PE, 5)
# M2 statistic for the one-factor copula model:

m2.1f.PE <- M2.1F(ncontinuous.PE, categorical.PE, tcount=NULL,
                  cpar=est1factor.PE$cpar, copF1=cop1f.PE, gl)
```

`M2.StructuredFactor` *Goodness-of-fit of bi-factor and second-order copula models for item response data*

Description

The limited information M_2 statistic (Maydeu-Olivares and Joe, 2006) of bi-factor and second-order copula models for item response data.

Usage

```
M2Bifactor(y,cpar, copnames1, copnames2, gl, ngrp, grpsize)
M2Second_order(y,cpar, copnames1, copnames2, gl, ngrp, grpsize)
```

Arguments

`y`	$n \times d$ matrix with the ordinal reponse data, where n and d is the number of observations and variables, respectively.
`cpar`	A list of estimated copula parameters.
`copnames1`	**For the bi-factor copula**: d-vector with the names of bivariate copulas that link each of the oberved variabels with the common factor. **For the second-order factor copula**: G-vector with the names of bivariate copulas that link the each of the group-specific factors with the common factor, where G is the number of groups of items. Choices are "bvn" for BVN, "bvtν" with $\nu = \{2, \ldots, 9\}$ degrees of freedom for t-copula, "frk" for Frank, "gum" for Gumbel, "rgum" for reflected Gumbel, "1rgum" for 1-reflected Gumbel, "2rgum" for 2-reflected Gumbel.
`copnames2`	**For the bi-factor copula**: d-vector with the names of bivariate copulas that link the each of the oberved variabels with the group-specific factor. **For the second-order factor copula**: d-vector with the names of bivariate copulas that link the each of the oberved variabels with the group-specific factor. Choices are "bvn" for BVN, "bvtν" with $\nu = \{2, \ldots, 9\}$ degrees of freedom for t-copula, "frk" for Frank, "gum" for Gumbel, "rgum" for reflected Gumbel, "1rgum" for 1-reflected Gumbel, "2rgum" for 2-reflected Gumbel.
`gl`	Gauss legendre quardrature nodes and weights.
`ngrp`	number of non-overlapping groups.
`grpsize`	vector indicating the size for each group, e.g., c(4,4,4) indicating four items in all three groups.

Details

The M_2 statistic has been developed for goodness-of-fit testing in multidimensional contingency tables by Maydeu-Olivares and Joe (2006). We use the M_2 to assess the overall fit for the bi-factor and second-order copula models for item resposne data (Kadhem & Nikoloulopoulos, 2021).

Value

A list containing the following components:

`M2`	The M_2 statistic which has a null asymptotic distribution that is χ^2 with $s - q$ degrees of freedom, where s is the number of univariate and bivariate margins that do not include the category 0 and q is the number of model parameters.

df $s - q$.

p-value The resultant p-value.

Author(s)

Sayed H. Kadhem <s.kadhem@uea.ac.uk>
Aristidis K. Nikoloulopoulos <a.nikoloulopoulos@uea.ac.uk>

References

Kadhem, S.H. and Nikoloulopoulos, A.K. (2023) Bi-factor and second-order copula models for item response data. *Psychometrika*, **88**, 132–157. doi:10.1007/s11336022098942.

Maydeu-Olivares, A. and Joe, H. (2006). Limited information goodness-of-fit testing in multidimensional contingency tables. *Psychometrika*, **71**, 713–732. doi:10.1007/s1133600512959.

Examples

```
#------------------------------------------------
# Setting quadreture points
nq <- 15
gl <- gauss.quad.prob(nq)
#------------------------------------------------
#                     TAS Data
#------------------             -----------------
data(TAS)
#using a subset of the data
#group1: 1,3,6,7,9,13,14
grp1=c(1,3,6)
#group2: 2,4,11,12,17
grp2=c(2,4,11)
#group3: 5,8,10,15,16,18,19,20
grp3=c(5,8,10)
#Rearrange items within testlets
set.seed(123)
i=sample(1:nrow(TAS),500)
ydat=TAS[i,c(grp1,grp2,grp3)]

d=ncol(ydat);d
n=nrow(ydat);n

#size of each group
g1=length(grp1)
g2=length(grp2)
g3=length(grp3)

grpsize=c(g1,g2,g3)#group size
#number of groups
ngrp=length(grpsize)

#------------------------------------------------
```

```
#                        M2
#-----------------------------------------------
#BI-FACTOR
tauX0 = c(0.49,0.16,0.29,#0.09,0.47,0.49,0.30,
          0.46,0.41,0.33,#0.29,0.24,
          0.10,0.16,0.14)#,0.12,0.03,0.03,0.10,0.10)
tauXg = c(0.09,0.37,0.23,#0.53,0.24,0.32,0.27,
          0.53,0.58,0.20,#0.23,0.25,0.34,0.33,
          0.30,0.19,0.24)#,0.29,0.43,0.26)
copX0 = rep("bvt2", d)
copXg = c(rep("rgum", g1), rep("bvt3", g2+g3))
#converting taus to cpars
cparX0=mapply(function(x,y) tau2par(x,y),x=copX0,y=tauX0)
cparXg=mapply(function(x,y) tau2par(x,y),x=copXg,y=tauXg)
cpar=c(cparX0,cparXg)

m2_Bifactor = M2Bifactor(y=ydat, cpar, copX0, copXg, gl, ngrp, grpsize)
```

---

mapping *Mapping of Kendall's tau and copula parameter*

Description

Bivariate copulas: mapping of Kendall's tau and copula parameter.

Usage

```
par2tau(copulaname, cpar)
tau2par(copulaname, tau)
```

Arguments

copulaname	Choices are "bvn" for BVN, "bvtν" with $\nu = \{1, \dots, 9\}$ degrees of freedom for t-copula, "frk" for Frank, "gum" for Gumbel, "rgum" for reflected Gumbel, "1rgum" for 1-reflected Gumbel, "2rgum" for 2-reflected Gumbel, "joe" for Joe, "rjoe" for reflected Joe, "1rjoe" for 1-reflected Joe, "2rjoe" for 2-reflected Joe, "BB1" for BB1, "rBB1" for reflected BB1, "BB7" for BB7, "rBB7" for reflected BB7, "BB8" for BB8, "rBB8" for reflected BB8, "BB10" for BB10, "rBB10" for reflected BB10.
cpar	Copula parameter(s).
tau	Kendall's tau.

Value

Kendall's tau or copula parameter.

References

Joe H (1997) *Multivariate Models and Dependence Concepts*. Chapman & Hall, London.

Joe H (2014) *Dependence Modeling with Copulas*. Chapman & Hall, London.

Joe H (2014) *CopulaModel: Dependence Modeling with Copulas*. Software for book: *Dependence Modeling with Copulas*, Chapman & Hall, London 2014.

Examples

```
# 1-param copulas
#BVN copula
cpar.bvn = tau2par("bvn", 0.5)
tau.bvn = par2tau("bvn", cpar.bvn)

#Frank copula
cpar.frk = tau2par("frk", 0.5)
tau.frk = par2tau("frk", cpar.frk)

#Gumbel copula
cpar.gum = tau2par("gum", 0.5)
tau.gum = par2tau("gum", cpar.gum)

#Joe copula
cpar.joe = tau2par("joe", 0.5)
tau.joe = par2tau("joe", cpar.joe)

# 2-param copulas
#BB1 copula
tau.bb1 = par2tau("bb1", c(0.5,1.5))

#BB7 copula
tau.bb7 = par2tau("bb7", c(1.5,1))

#BB8 copula
tau.bb8 = par2tau("bb8", c(3,0.8))

#BB10 copula
tau.bb10 = par2tau("bb10", c(3,0.8))
```

`mle.Factor` *Maximum likelihood estimation of factor copula models for mixed data*

Description

We use a two-stage etimation approach toward the estimation of factor copula models for mixed continuous and discrete data.

Usage

```
mle1factor(continuous, ordinal, count, copF1, gl, hessian, print.level)
mle2factor(continuous, ordinal, count, copF1, copF2, gl, hessian, print.level)
mle2factor.bvn(continuous, ordinal, count, copF1, copF2, gl, SpC, print.level)
```

Arguments

`continuous`	$n \times d_1$ matrix with the continuous reponse data, where n and d_1 is the number of observations and continous variables, respectively.
`ordinal`	$n \times d_2$ matrix with the ordinal reponse data, where n and d_2 is the number of observations and ordinal variables, respectively.
`count`	$n \times d_3$ matrix with the count reponse data, where n and d_3 is the number of observations and count variables, respectively.
`copF1`	$(d_1 + d_2 + d_3)$-vector with the names of bivariate copulas that link the each of the oberved variabels with the 1st factor. Choices are "bvn" for BVN, "bvtν" with $\nu = \{1, \ldots, 9\}$ degrees of freedom for t-copula, "frk" for Frank, "gum" for Gumbel, "rgum" for reflected Gumbel, "1rgum" for 1-reflected Gumbel, "2rgum" for 2-reflected Gumbel, "joe" for Joe, "rjoe" for reflected Joe, "1rjoe" for 1-reflected Joe, "2rjoe" for 2-reflected Joe, "BB1" for BB1, "rBB1" for reflected BB1, "BB7" for BB7, "rBB7" for reflected BB7, "BB8" for BB8, "rBB8" for reflected BB8, "BB10" for BB10, "rBB10" for reflected BB10.
`copF2`	$(d_1 + d_2 + d_3)$-vector with the names of bivariate copulas that link the each of the oberved variabels with the 2nd factor. Choices are "bvn" for BVN, "bvtν" with $\nu = \{1, \ldots, 9\}$ degrees of freedom for t-copula, "frk" for Frank, "gum" for Gumbel, "rgum" for reflected Gumbel, "1rgum" for 1-reflected Gumbel, "2rgum" for 2-reflected Gumbel, "joe" for Joe, "rjoe" for reflected Joe, "1rjoe" for 1-reflected Joe, "2rjoe" for 2-reflected Joe, "BB1" for BB1, "rBB1" for reflected BB1, "BB7" for BB7, "rBB7" for reflected BB7, "BB8" for BB8, "rBB8" for reflected BB8, "BB10" for BB10, "rBB10" for reflected BB10.
`gl`	Gauss legendre quardrature nodes and weights.
`SpC`	Special case for the 2-factor copula model with BVN copulas. Select a bivariate copula at the 2nd factor to be fixed to independence. e.g. "SpC = 1" to set the first copula at the 2nd factor to independence.
`hessian`	If TRUE, the hessian of the negative log-likelihood is calculated during the minimization process.
`print.level`	Determines the level of printing which is done during the minimization process; same as in `nlm`.

Details

Estimation is achieved by maximizing the joint log-likelihood over the copula parameters with the univariate parameters/distributions fixed as estimated at the first step of the proposed two-step estimation approach.

Value

A list containing the following components:

`cutpoints`	The estimated univariate cutpoints.
`negbinest`	The estimated univariate parametes for the count responses (fitting the negative binomial distribution).
`loglik`	The maximized joint log-likelihood.
`cpar`	Estimated copula parameters in a list form.
`taus`	The estimated copula parameters in Kendall's tau scale.
`SEs`	The SEs of the Kendall's tau estimates.

Author(s)

Sayed H. Kadhem <s.kadhem@uea.ac.uk>
Aristidis K. Nikoloulopoulos <a.nikoloulopoulos@uea.ac.uk>

References

Kadhem, S.H. and Nikoloulopoulos, A.K. (2021) Factor copula models for mixed data. *British Journal of Mathematical and Statistical Psychology*, **74**, 365–403. doi:10.1111/bmsp.12231.

Krupskii, P. and Joe, H. (2013) Factor copula models for multivariate data. *Journal of Multivariate Analysis*, **120**, 85–101. doi:10.1016/j.jmva.2013.05.001.

Nikoloulopoulos, A.K. and Joe, H. (2015) Factor copula models with item response data. *Psychometrika*, **80**, 126–150. doi:10.1007/s1133601393874.

Examples

```
#------------------------------------------------
# Setting quadreture points
nq <- 25
gl <- gauss.quad.prob(nq)
#------------------------------------------------
#                     PE Data
#------------------             -----------------
data(PE)
continuous.PE1 = -PE[,1]
continuous.PE2 = PE[,2]
continuous.PE <- cbind(continuous.PE1, continuous.PE2)

categorical.PE <- PE[, 3:5]
#------------------------------------------------
#                   Estimation
#------------------             -----------------
#------------------ One-factor  -----------------
# one-factor copula model
cop1f.PE <- c("joe", "joe", "rjoe", "joe", "gum")
est1factor.PE <- mle1factor(continuous.PE, categorical.PE,
                            count=NULL, copF1=cop1f.PE, gl, hessian = TRUE)
```

```
est1factor.PE
#-----------------------------------------------
#-----------------------------------------------
#                    GSS Data
#------------------            -----------------
data(GSS)
attach(GSS)
continuous.GSS <- cbind(INCOME, AGE)
ordinal.GSS <- cbind(DEGREE, PINCOME, PDEGREE)
count.GSS <- cbind(CHILDREN, PCHILDREN)

#-----------------------------------------------
#                  Estimation
#------------------            -----------------
#------------------ One-factor  -----------------
# one-factor copula model
cop1f.GSS <- c("joe","2rjoe","bvt3","bvt3",
          "rgum","2rjoe","2rgum")
est1factor.GSS <- mle1factor(continuous.GSS, ordinal.GSS,
                        count.GSS, copF1 = cop1f.GSS, gl, hessian = TRUE)
```

`mle.FactorTree` *Maximum likelihood estimation of factor tree copula models*

Description

We use a two-stage estimation approach toward the estimation of factor tree copula models for item response data.

Usage

```
mle1FactorTree(y, A, cop, gl, hessian, print.level)
mle2FactorTree(y, A, cop, gl, hessian, print.level)
```

Arguments

`y`	$n \times d$ matrix with the ordinal reponse data, where n and d is the number of observations and ordinal variables, respectively.
`A`	$d \times d$ vine array with $1, ..., d$ on diagonal, note only the first row and diagnoal values are used for the 1-truncated vine model
`cop`	$(2d-1)$-vector with the names of bivariate copulas that link each of the oberved variabels with the 1st factor (1-factor part of the model), and conditional dependence of variables given the latent factor (1-truncated vine tree part of the model) . Choices are "bvn" for BVN, "bvtν" with $\nu = \{1, \ldots, 9\}$ degrees of freedom for t-copula, "frk" for Frank, "gum" for Gumbel, "rgum" for reflected Gumbel, "1rgum" for 1-reflected Gumbel, "2rgum" for 2-reflected Gumbel.

gl	Gauss legendre quardrature nodes and weights.
hessian	If TRUE, the hessian of the negative log-likelihood is calculated during the minimization process.
print.level	Determines the level of printing which is done during the minimization process; same as in nlm.

Details

Estimation is achieved by maximizing the joint log-likelihood over the copula parameters with the univariate cutpoints fixed as estimated at the first step of the proposed two-step estimation approach.

Value

A list containing the following components:

cutpoints	The estimated univariate cutpoints.
loglik	The maximized joint log-likelihood.
taus	The estimated copula parameters in Kendall's tau scale.
SEs	The SEs of the Kendall's tau estimates.

Author(s)

Sayed H. Kadhem
Aristidis K. Nikoloulopoulos <a.nikoloulopoulos@uea.ac.uk>

References

Joe, H. (2014). *Dependence Modelling with Copulas*. Chapman & Hall, London.

Kadhem, S.H. and Nikoloulopoulos, A.K. (2022b) Factor tree copula models for item response data. *Arxiv e-prints*, <arXiv: 2201.00339>. https://arxiv.org/abs/2201.00339.

Examples

```
#------------------------------------------------
# Setting quadreture points
nq <- 5
gl <- gauss.quad.prob(nq)
#------------------------------------------------
#                    PTSD Data
#------------------             -----------------
data(PTSD)
ydat=PTSD
n=nrow(ydat)
d=ncol(ydat)
#------------------------------------------------
#                   Estimation
#------------------             -----------------
#selecting vine tree based on polychoric
```

```
rmat=polychoric0(ydat)$p
A.polychoric=selectFactorTrVine(y=ydat,rmat,alg=3)

#---------------- 1-factor tree  ----------------
# 1-factor tree copula model
copf1 <- rep("frk",d)
coptree <- rep("frk",d-1)
cop <- c(copf1,coptree)
est1factortree <- mle1FactorTree(y=ydat, A=A.polychoric$VineTreeA, cop,
gl, hessian=FALSE, print.level=2)

#---------------- 2-factor tree  ----------------
# 2-factor tree copula model
copf1 <- rep("frk",d)
copf2 <- rep("frk",d)
coptree <- rep("frk",d-1)
cop <- c(copf1,copf2,coptree)

est2factortree <- mle2FactorTree(y=ydat, A=A.polychoric$VineTreeA,
cop, gl, hessian=FALSE, print.level=2)
```

`mle.StructuredFactor` *Maximum likelihood estimation of the bi-factor and second-order copula models for item response data*

Description

We approach the estimation of the bi-factor and second-order copula models for item response data with the IFM method of Joe (2005).

Usage

```
mleBifactor(y, copnames1, copnames2, gl, ngrp, grpsize,
hessian, print.level)
mleSecond_order(y, copnames1, copnames2, gl, ngrp, grpsize,
hessian, print.level)
```

Arguments

`y` $n \times d$ matrix with the item reponse data, where n and d is the number of observations and variables, respectively.

`copnames1` **For the bi-factor copula**: d-vector with the names of bivariate copulas that link the each of the oberved variabels with the common factor. **For the second-order factor copula**: G-vector with the names of bivariate copulas that link the each of the group-specific factors with the common factor, where G is the number of groups of items. Choices are “bvn” for BVN, “bvtν” with $\nu = \{1, \ldots, 9\}$ degrees of freedom for t-copula, “frk” for Frank, “gum” for Gumbel, “rgum” for reflected Gumbel, "1rgum" for 1-reflected Gumbel, "2rgum" for 2-reflected Gumbel.

`copnames2`	**For the bi-factor copula**: d-vector with the names of bivariate copulas that link the each of the oberved variabels with the group-specific factor. **For the second-order factor copula**: d-vector with the names of bivariate copulas that link the each of the oberved variabels with the group-specific factor. Choices are "bvn" for BVN, "bvtν" with $\nu = \{1, \ldots, 9\}$ degrees of freedom for t-copula, "frk" for Frank, "gum" for Gumbel, "rgum" for reflected Gumbel, "1rgum" for 1-reflected Gumbel, "2rgum" for 2-reflected Gumbel.
`gl`	Gauss legendre quardrature nodes and weights.
`ngrp`	number of non-overlapping groups.
`grpsize`	vector indicating the size for each group, e.g., c(4,4,4) indicating four items in all three groups.
`hessian`	If TRUE, the hessian of the negative log-likelihood is calculated during the minimization process.
`print.level`	Determines the level of printing which is done during the minimization process; same as in `nlm`.

Details

Estimation is achieved by maximizing the joint log-likelihood over the copula parameters with the univariate cutpoints fixed as estimated at the first step of the proposed two-step estimation approach.

Value

A list containing the following components:

`cutpoints`	The estimated univariate cutpoints.
`taus`	The estimated copula parameters in Kendall's tau scale.
`SEs`	The SEs of the Kendall's tau estimates.
`loglik`	The maximized joint log-likelihood.

Author(s)

Sayed H. Kadhem <s.kadhem@uea.ac.uk>
Aristidis K. Nikoloulopoulos <a.nikoloulopoulos@uea.ac.uk>

References

Joe, H. (2005) Asymptotic efficiency of the two-stage estimation method for copula-based models. *Journal of Multivariate Analysis*, **94**, 401–419. doi:10.1016/j.jmva.2004.06.003.

Kadhem, S.H. and Nikoloulopoulos, A.K. (2023) Bi-factor and second-order copula models for item response data. *Psychometrika*, **88**, 132–157. doi:10.1007/s11336022098942.

Examples

```
#------------------------------------------------
# Setting quadreture points
nq <- 25
gl <- gauss.quad.prob(nq)
#------------------------------------------------
#                    TAS Data
#------------------             -----------------
data(TAS)
#using a subset of the data
#group1: 1,3,6,7,9,13,14
grp1=c(1,3,6)
#group2: 2,4,11,12,17
grp2=c(2,4,11)
#group3: 5,8,10,15,16,18,19,20
grp3=c(5,8,10)
#Rearrange items within testlets
set.seed(123)
i=sample(1:nrow(TAS),500)
ydat=TAS[i,c(grp1,grp2,grp3)]

d=ncol(ydat);d
n=nrow(ydat);n

#size of each group
g1=length(grp1)
g2=length(grp2)
g3=length(grp3)

grpsize=c(g1,g2,g3)#group size
#number of groups
ngrp=length(grpsize)

#BI-FACTOR
copX0 = rep("bvt2", d)
copXg = c(rep("rgum", g1), rep("bvt3", g2+g3))
mle_Bifactor = mleBifactor(y = ydat, copX0, copXg, gl, ngrp, grpsize, hessian=FALSE, print.level=2)
```

---

pbnorm *Bivariate normal and Student cdfs with vectorized inputs*

Description

Bivariate normal and Student cdfs with vectorized inputs

Usage

```
pbnorm(z1,z2,rho,icheck=FALSE)
pbvt(z1,z2,param,icheck=FALSE)
```

Arguments

z1	scalar or vector of reals
z2	scalar or vector of reals
rho	scalar or vector parameter in (-1,1); vectors cannot have different lengths if larger than 1, each of z1,z2,rho either has length 1 or a constant n greater than 1
param	vector of length 2, or matrix with 2 columns; vectors and number of rows of matrix cannot be different if larger than 1; for param, first column is rho, second column is df.
icheck	T if checks are made for proper inputs, default of F

Details

Donnelly's code can be inaccurate in the tail when the tail probability is 2.e-9 or less (it sometimes returns 0). In the case the exchmvn code is used with dimension 2. Alternatively a user can use vectorized function pbivnorm() in the library pbivnorm, and write a function pbvncop based on it.

Value

cdf value(s)

References

Donnelly TG (1973). Algorithm 462: bivariate normal distribution, Communications of the Association for Computing Machinery, 16, 638;

Joe H (2014) *Dependence Modeling with Copulas*. Chapman & Hall/CRC.

Joe H (2014) *CopulaModel: Dependence Modeling with Copulas*. Software for book: *Dependence Modeling with Copulas*, Chapman & Hall/CRC, 2014.

Examples

```
cat("\n pbnorm rho changing\n")
z1=.3; z2=.4; rho=seq(-1,1,.1)
out1=pbnorm(z1,z2,rho)
print(cbind(rho,out1))

cat("\n pbnorm matrix inputs for z1, z2\n")
rho=.4
z1=c(-.5,.5,10.)
z2=c(-.4,.6,10.)
z1=matrix(z1,3,3)
z2=matrix(z2,3,3,byrow=TRUE)
out3=pbnorm(z1,z2,rho)
print(out3)
```

```
cdf2=rbind(rep(0,3),out3)
cdf2=cbind(rep(0,4),cdf2)
pmf=apply(cdf2,2,diff)
pmf2=apply(t(pmf),2,diff)
pmf2=t(pmf2)  # rectangle probabilities
print(pmf2)

cat("\n pbvt rho changing\n")
z1=.3; z2=.4; rho=seq(-.9,.9,.1); nu=2
param=cbind(rho,rep(nu,length(rho)))
out1=pbvt(z1,z2,param)
print(cbind(rho,out1))
cat("\n pbvt z1 changing\n")
z1=seq(-2,2,.4)
z2=.4; rho=.5; nu=2
out2=pbvt(z1,z2,c(rho,nu))
print(cbind(z1,out2))
```

PE *Political-economic risk of 62 countries for the year 1987*

Description

Quinn (2004) used 5 mixed variables, namely the continuous variable black-market premium in each country (used as a proxy for illegal economic activity), the continuous variable productivity as measured by real gross domestic product per worker in 1985 international prices, the binary variable independence of the national judiciary (1 if the judiciary is judged to be independent and 0 otherwise), and the ordinal variables measuring the lack of expropriation risk and lack of corruption.

Usage

```
data(PE)
```

Format

A data frame with 62 observations (countries) on the following 5 variables:

`BM` Black-market premium.

`GDP` Gross domestic product.

`IJ` Independent judiciary.

`XPR` Lack of expropriation risk.

`CPR` Lack of corruption.

Source

Quinn, K. M. (2004). Bayesian factor analysis for mixed ordinal and continuous responses. *Political Analysis*, **12**, 338–353.

polychoric0 *Polychoric correlation*

Description

Polychoric correlation

Usage

```
polychoric0(odat,iprint=FALSE,prlevel=0)
```

Arguments

odat	nxd matrix of ordinal responses in 0,...,(ncat-1) or 1,...,ncat
iprint	flag for printing of intermediate results, including bivariate tables for observed versus expected assuming discretized bivariate Gaussian
prlevel	print.level for nlm for numerical optimization

Details

Polychoric correlation for ordinal random variables. The number of categories can vary.

Value

$polych is a polychoric correlation matrix based on two-stage estimate; $iposdef is an indicator if the 2-stage correlation matrix estimate is positive definite.

Examples

```
data(PTSD)
ydat=PTSD
rmat=polychoric0(ydat)$p
```

PTSD *The Post-traumatic stress disorder (PTSD)*

Description

The measure consists of 20 categorical items divided into four domains: (1) intrusions (e.g., repeated, dis- turbing, and unwanted memories), (2) avoidance (e.g., avoiding external reminder of the stressful experience), (3) cognition and mood alterations (e.g., trouble remembering important parts of the stressful experience) and (4) reactivity alterations (e.g., taking too many risks or doing things that could cause you harm). Each item is answered in a five-point ordinal scale: “0 = Not at all”, “1 = A little bit”, “2 = Moderately”, “3 = Quite a bit”, and “4 = Extremely”.

Usage

```
data(PTSD)
```

Format

A data frame with 221 observations and 20 items.

Source

Williams, D. and Mulder, J. (2020). *BGGM: Bayesian Gaussian Graphical Models*. R package version 1.0.0.

`rfactor` *Simulation of factor copula models for mixed continuous and discrete data*

Description

Simulating standard uniform and ordinal response data from factor copula models.

Usage

```
r1factor(n, d1, d2, categ, theta, copF1)
r2factor(n, d1, d2, categ, theta, delta, copF1, copF2)
```

Arguments

`n`	Sample size.
`d1`	Number of standard uniform variables.
`d2`	Number of ordinal variables.
`categ`	A vector of categories for the ordinal variables.
`theta`	Copula parameters for the 1st factor.
`delta`	Copula parameters for the 2nd factor.
`copF1`	$(d_1 + d_2)$-vector with the names of bivariate copulas that link the each of the oberved variabels with the 1st factor. Choices are “bvn” for BVN, “bvtν” with $\nu = \{1, \ldots, 9\}$ degrees of freedom for t-copula, “frk” for Frank, “gum” for Gumbel, “rgum” for reflected Gumbel, “1rgum” for 1-reflected Gumbel, “2rgum” for 2-reflected Gumbel, “joe” for Joe, “rjoe” for reflected Joe, “1rjoe” for 1-reflected Joe, “2rjoe” for 2-reflected Joe, “BB1” for BB1, “rBB1” for reflected BB1, “BB7” for BB7, “rBB7” for reflected BB7, “BB8” for BB8, “rBB8” for reflected BB8, “BB10” for BB10, “rBB10” for reflected BB10.

copF2 — $(d_1 + d_2)$-vector with the names of bivariate copulas that link the each of the oberved variabels with the 2nd factor. Choices are “bvn” for BVN, “bvt[ν]” with $\nu = \{1, \ldots, 9\}$ degrees of freedom for t-copula, “frk” for Frank, “gum” for Gumbel, “rgum” for reflected Gumbel, “1rgum” for 1-reflected Gumbel, “2rgum” for 2-reflected Gumbel, “joe” for Joe, “rjoe” for reflected Joe, “1rjoe” for 1-reflected Joe, “2rjoe” for 2-reflected Joe, “BB1” for BB1, “rBB1” for reflected BB1, “BB7” for BB7, “rBB7” for reflected BB7, “BB8” for BB8, “rBB8” for reflected BB8, “BB10” for BB10, “rBB10” for reflected BB10.

Value

Data matrix of dimension $n \times d$, where n is the sample size, and $d = d_1 + d_2$ is the total number of variables.

Author(s)

Sayed H. Kadhem <s.kadhem@uea.ac.uk>
Aristidis K. Nikoloulopoulos <a.nikoloulopoulos@uea.ac.uk>

References

Kadhem, S.H. and Nikoloulopoulos, A.K. (2021) Factor copula models for mixed data. *British Journal of Mathematical and Statistical Psychology*, **74**, 365–403. doi:10.1111/bmsp.12231.

Examples

```
# ----------------------------------------------------
# ----------------------------------------------------
#              One-factor copula model
# ----------------------------------------------------
# ----------------------------------------------------
#Sample size ----------------------------------------
n = 100

#Continuous Variables  ------------------------------
d1 = 5

#Ordinal Variables  ---------------------------------
d2 = 3

#Categories for ordinal  ----------------------------
categ = c(3,4,5)

#Copula parameters  ---------------------------------
theta = rep(2, d1+d2)

#Copula names  --------------------------------------
copnamesF1 = rep("gum", d1+d2)
```

```
#----------------- Simulating data ------------------
datF1 = r1factor(n, d1=d1, d2=d2, categ, theta, copnamesF1)

#------------  Plotting continuous data -------------
pairs(qnorm(datF1[, 1:d1]))

# ---------------------------------------------------
# ---------------------------------------------------
#              Two-factor copula model
# ---------------------------------------------------
# ---------------------------------------------------
#Sample size ---------------------------------------
n = 100

#Continuous Variables  ------------------------------
d1 = 5

#Ordinal Variables  ---------------------------------
d2 = 3

#Categories for ordinal  ----------------------------
categ = c(3,4,5)

#Copula parameters  ---------------------------------
theta = rep(2.5, d1+d2)
delta = rep(1.5, d1+d2)

#Copula names  -------------------------------------
copnamesF1 = rep("gum", d1+d2)
copnamesF2 = rep("gum", d1+d2)

#----------------- Simulating data ------------------
datF2 = r2factor(n, d1=d1, d2=d2, categ, theta, delta,
                copnamesF1, copnamesF2)

#-----------------  Plotting  data ------------------
pairs(qnorm(datF2[,1:d1]))
```

`rFactorTree` *Simulation of 1- and 2-factor tree copula models for item response data*

Description

Simulating item response data from the 1- and 2-factor tree copula models.

Usage

```
r1factortree(n, d, A, copname1, copnametree, theta1, delta,K)
r2factortree(n, d, A, copname1, copname2, copnametree,theta1, theta2, delta,K)
```

Arguments

n	Sample size.
d	Number of observed variables/items.
A	$d \times d$ vine array with $1, ..., d$ on diagonal, note only the first row and diagnoal values are used for the 1-truncated vine model
theta1	copula parameter vector of size d for items with the first factor.
theta2	copula parameter vector of size d for items with the second factor.
delta	copula parameter vector of size $d - 1$ for the 1-truncated vine tree (conditional dependence).
copname1	A name of a bivariate copula that link each of the oberved variabels with the first factor (note only a single copula family for all items with the factor). Choices are “bvn” for BVN, “bvtν” with $\nu = \{1, \ldots, 9\}$ degrees of freedom for t-copula, “frk” for Frank, “gum” for Gumbel, “rgum” for reflected Gumbel, “1rgum” for 1-reflected Gumbel, “2rgum” for 2-reflected Gumbel.
copname2	A name of a bivariate copula that link each of the oberved variabels with the second factor (note only a single copula family for all items with the factor). Choices are “bvn” for BVN, “bvtν” with $\nu = \{1, \ldots, 9\}$ degrees of freedom for t-copula, “frk” for Frank, “gum” for Gumbel, “rgum” for reflected Gumbel, “1rgum” for 1-reflected Gumbel, “2rgum” for 2-reflected Gumbel.
copnametree	A name of a bivariate copula that link each of the oberved variabels with one another given the factors in the 1-truncated vine (note only a single copula family for all tree). Choices are “bvn” for BVN, “bvtν” with $\nu = \{1, \ldots, 9\}$ degrees of freedom for t-copula, “frk” for Frank, “gum” for Gumbel, “rgum” for reflected Gumbel, “1rgum” for 1-reflected Gumbel, “2rgum” for 2-reflected Gumbel.
K	Number of categories for the observed variables/items.

Value

Data matrix of dimension $n \times d$, where n is the sample size, and d is the total number of observed variables/items.

Author(s)

Sayed H. Kadhem
Aristidis K. Nikoloulopoulos <a.nikoloulopoulos@uea.ac.uk>

References

Joe, H. (2014). *Dependence Modelling with Copulas*. Chapman & Hall, London.

Kadhem, S.H. and Nikoloulopoulos, A.K. (2022b) Factor tree copula models for item response data. *Arxiv e-prints*, <arXiv: 2201.00339>. https://arxiv.org/abs/2201.00339.

Examples

```
# -----------------------------------------------------
# -----------------------------------------------------
#Sample size
n = 500

#Ordinal Variables  ---------------------------------
d = 5

#Categories for ordinal  ----------------------------
K = 5
# -----------------------------------------------------
#              1-2-factor tree copula model
# -----------------------------------------------------
#Copula parameters
theta1 = rep(3, d)
theta2 = rep(2, d)
delta = rep(1.5, d-1)

#Copula names
copulaname_1f = "gum"
copulaname_2f = "gum"
copulaname_vine = "gum"

#vine array
#Dvine
d=5
A=matrix(0,d,d)
A[1,]=c(1,c(1:(d-1)))
diag(A)=1:d



#----------------- Simulating data ------------------
#1-factor tree copula
data_1ft = r1factortree(n, d, A, copulaname_1f, copulaname_vine,
theta1, delta,K)
#2-factor tree copula
data_2ft = r2factortree(n, d, A, copulaname_1f, copulaname_2f,
copulaname_vine, theta1,theta2, delta,K)
```

`rStructuredFactor` *Simulation of bi-factor and second-order copula models for item response data*

Description

Simulating item response data from the bi-factor and second-order copula models for item response data.

Usage

```
rBifactor(n, d, grpsize, categ, copnames1,copnames2, theta1, theta2)
rSecond_order(n, d, grpsize, categ, copnames1, copnames2, theta1, theta2)
```

Arguments

`n`	Sample size.
`d`	Number of observed variables/items.
`grpsize`	vector indicating the size for each group, e.g., c(4,4,4) indicating four items in all three groups.
`categ`	A vector of categories for the observed variables/items.
`theta1`	**For the bi-factor model**: copula parameter vector of size d for items with the common factor. **For the second-order copulas**: copula parameter vector of size G for the common factor and group-specific factors.
`theta2`	**For the bi-factor model**: copula parameter vector of size d for items with the group-specific factor. **For the second-order copulas**: copula parameter vector of size d for items with the group-specific factor.
`copnames1`	**For the bi-factor copula**: d-vector with the names of bivariate copulas that link the each of the oberved variabels with the common factor. **For the second-order factor copula**: G-vector with the names of bivariate copulas that link the each of the group-specific factors with the common factor, where G is the number of groups of items. Choices are "bvn" for BVN, "bvtν" with $\nu = \{1, \ldots, 9\}$ degrees of freedom for t-copula, "frk" for Frank, "gum" for Gumbel, "rgum" for reflected Gumbel, "1rgum" for 1-reflected Gumbel, "2rgum" for 2-reflected Gumbel.
`copnames2`	**For the bi-factor copula**: d-vector with the names of bivariate copulas that link the each of the oberved variabels with the group-specific factor. **For the second-order factor copula**: d-vector with the names of bivariate copulas that link the each of the oberved variabels with the group-specific factor. Choices are "bvn" for BVN, "bvtν" with $\nu = \{1, \ldots, 9\}$ degrees of freedom for t-copula, "frk" for Frank, "gum" for Gumbel, "rgum" for reflected Gumbel, "1rgum" for 1-reflected Gumbel, "2rgum" for 2-reflected Gumbel.

Value

Data matrix of dimension $n \times d$, where n is the sample size, and d is the total number of observed variables/items.

Author(s)

Sayed H. Kadhem `<s.kadhem@uea.ac.uk>`
Aristidis K. Nikoloulopoulos `<a.nikoloulopoulos@uea.ac.uk>`

References

Kadhem, S.H. and Nikoloulopoulos, A.K. (2023) Bi-factor and second-order copula models for item response data. *Psychometrika*, **88**, 132–157. doi:10.1007/s11336022098942.

Examples

```
# --------------------------------------------------
# --------------------------------------------------
#Sample size
n = 500

#Ordinal Variables  ---------------------------------
d = 9
grpsize=c(3,3,3)
ngrp=length(grpsize)

#Categories for ordinal  ----------------------------
categ = rep(3,d)

# --------------------------------------------------
# --------------------------------------------------
#              Bi-factor copula model
# --------------------------------------------------
# --------------------------------------------------
#Copula parameters
theta = rep(2.5, d)
delta = rep(1.5, d)

#Copula names
copulanames1 = rep("gum", d)
copulanames2 = rep("gum", d)

#----------------- Simulating data ------------------
data_Bifactor = rBifactor(n, d, grpsize, categ, copulanames1,
copulanames2, theta, delta)

# --------------------------------------------------
# --------------------------------------------------
#              Second-order copula model
# --------------------------------------------------
# --------------------------------------------------
#Copula parameters
theta= rep(1.5, ngrp)
delta = rep(2.5, d)

#Copula names
copulanames1 = rep("gum", ngrp)
copulanames2 = rep("gum", d)

data_Second_order = rSecond_order(n, d, grpsize, categ,
copulanames1, copulanames2, theta, delta)
```

`Select.Factor` *Model selection of the factor copula models for mixed data*

Description

A heuristic algorithm that automatically selects the bivariate parametric copula families that link the observed to the latent variables.

Usage

```
select1F(continuous, ordinal, count, copnamesF1, gl)
select2F(continuous, ordinal, count, copnamesF1, copnamesF2, gl)
```

Arguments

`continuous`	$n \times d_1$ matrix with the continuous reponse data, where n and d_1 is the number of observations and continous variables, respectively.
`ordinal`	$n \times d_2$ matrix with the ordinal reponse data, where n and d_2 is the number of observations and ordinal variables, respectively.
`count`	$n \times d_3$ matrix with the count reponse data, where n and d_3 is the number of observations and count variables, respectively.
`copnamesF1`	A vector with the names of possible candidates of bivariate copulas that link the each of the oberved variabels with the 1st factor. Choices are "bvn" for BVN, "bvtν" with $\nu = \{1, \ldots, 9\}$ degrees of freedom for t-copula, "frk" for Frank, "gum" for Gumbel, "rgum" for reflected Gumbel, "1rgum" for 1-reflected Gumbel, "2rgum" for 2-reflected Gumbel, "joe" for Joe, "rjoe" for reflected Joe, "1rjoe" for 1-reflected Joe, "2rjoe" for 2-reflected Joe, "BB1" for BB1, "rBB1" for reflected BB1, "BB7" for BB7, "rBB7" for reflected BB7, "BB8" for BB8, "rBB8" for reflected BB8, "BB10" for BB10, "rBB10" for reflected BB10.
`copnamesF2`	A list with the names of possible candidates of bivariate copulas that link the each of the oberved variabels with the 1st and 2nd factors. Choices are "bvn" for BVN, "bvtν" with $\nu = \{1, \ldots, 9\}$ degrees of freedom for t-copula, "frk" for Frank, "gum" for Gumbel, "rgum" for reflected Gumbel, "1rgum" for 1-reflected Gumbel, "2rgum" for 2-reflected Gumbel, "joe" for Joe, "rjoe" for reflected Joe, "1rjoe" for 1-reflected Joe, "2rjoe" for 2-reflected Joe, "BB1" for BB1, "rBB1" for reflected BB1, "BB7" for BB7, "rBB7" for reflected BB7, "BB8" for BB8, "rBB8" for reflected BB8, "BB10" for BB10, "rBB10" for reflected BB10.
`gl`	Gauss legendre quardrature nodes and weights.

Details

The linking copulas at each factor are selected with a sequential algorithm under the initial assumption that linking copulas are Frank, and then sequentially copulas with non-tail quadrant independence are assigned to any of pairs where necessary to account for tail asymmetry (discrete data) or tail dependence (continuous data).

Value

A list containing the following components:

`''1st factor''`	The selected bivariate linking copulas for the 1st factor.

``2nd factor''	The selected bivariate linking copulas for the 2nd factor.
AIC	Akaike information criterion.
taus	The estimated copula parameters in Kendall's tau scale.

Author(s)

Sayed H. Kadhem <s.kadhem@uea.ac.uk>
Aristidis K. Nikoloulopoulos <a.nikoloulopoulos@uea.ac.uk>

References

Kadhem, S.H. and Nikoloulopoulos, A.K. (2021) Factor copula models for mixed data. *British Journal of Mathematical and Statistical Psychology*, **74**, 365–403. doi:10.1111/bmsp.12231.

Examples

```
#------------------------------------------------
#                   Estimation
#------------------              -----------------
# Setting quadreture points
nq<-25
gl<-gauss.quad.prob(nq)
#------------------------------------------------
#                     PE Data
#------------------              -----------------
data(PE)
continuous.PE1 = -PE[,1]
continuous.PE <- cbind(continuous.PE1, PE[,2])
categorical.PE <- PE[, 3:5]

#------------------ One-factor  -----------------
# listing the possible copula candidates:
d <- ncol(PE)
copulasF1 <- rep(list(c("bvn", "bvt3", "bvt5", "frk", "gum",
"rgum", "rjoe","joe", "1rjoe","2rjoe", "1rgum","2rgum")), d)
out1F.PE <- select1F(continuous.PE, categorical.PE,
count=NULL, copulasF1, gl)
```

Select.FactorTree *Model selection of the 1- and 2-factor tree copula models for item response data*

Description

Heuristic algorithms that automatically select the bivariate parametric copula families and 1-truncated vine tree structure for the 1- and 2-factor tree copula models for item response data.

Usage

```
selectFactorTrVine(y,rmat,alg)
copselect1FactorTree(y, A, copnames, gl)
copselect2FactorTree(y, A, copnames, gl)
```

Arguments

`y`	$n \times d$ matrix with the item reponse data, where n and d is the number of observations and variables, respectively.
`rmat`	Polychoric correlation matrix of y.
`alg`	1-truncated vine selection using partial and polychoric correlations. Choose a number from (1,2,3). **1**: Partial correlation selection algorithm for 1-factor tree copula. **2**: Partial correlation selection algorithm for 2-factor tree copula. **3**: Polychoric correlation selection algorithm for 1-factor/2-factor tree copulas, here any other correlation matrix (rmat) can be used.
`A`	$d \times d$ vine array with $1, ..., d$ on diagonal, note only the first row and diagnoal values are used for the 1-truncated vine model
`copnames`	A vector with the names of possible candidates of bivariate copulas to be selected for the 1-factor/2-factor copula models for item response data. Choices are "bvn" for BVN, "bvtν" with $\nu = \{1, \dots, 9\}$ degrees of freedom for t-copula, "frk" for Frank, "gum" for Gumbel, "rgum" for reflected Gumbel, "1rgum" for 1-reflected Gumbel, "2rgum" for 2-reflected Gumbel.
`gl`	Gauss legendre quardrature nodes and weights.

Details

The model selection involves two separate steps. At the first step we select the 1-truncated vine or Markov tree structure assuming our models are constructed with BVN copulas. We find the maximum spanning tree which is a tree on all nodes that maximizes the pairwise dependencies using the well-known algorithm of Prim (1957). That is we find the tree with $d-1$ edges $\mathcal{E}$ that minimizes $\sum_{\mathcal{E}} \log(1-r_{jk}^2)$. The minimum spanning tree algorithm of Prim (1957) guarantees to find the optimal solution when edge weights between nodes are given by $\log(1-r_{jk}^2)$. We use two different measures of pairwise dependence r_{jk}. The first measure is the estimated polychoric correlation (Olsson, 1979). The second measures of pairwise dependence is the partial correlation which is based on the factor copula models in Nikoloulopoulos and Joe (2015) with BVN copulas; for more details see Kadhem and Nikoloulopoulos (2022). We call polychoric and partial correlation selection algorithm when the pairwise dependence is the polychoric and partial correlation, respectively.

At the the second step, we sequentially select suitable bivariate copulas to account for any tail dependence/asymmetry. We start from the 1st tree that includes copulas connecting the observed variables to the first factor and then we iterate over a set of copula candidates and select suitable bivariate copulas in the subsequent trees. We select only one copula family for all the edges in each tree to ease interpretation.

Value

A list containing the following components:

``1Factor copulas''
: The selected bivariate linking copulas for the first factor.

``2Factor copulas''
: The selected bivariate linking copulas for the second factor.

``Vine tree copulas''
: The selected bivariate linking copulas for the 1-truncated vine tree.

``log-likelihood''
: The maximized joint log-likelihood.

``estimated taus''
: The estimated copula parameters in Kendall's tau scale.

`F1treeA` Vine array for the 1-factor tree copula.

`F2treeA` Vine array for the 2-factor tree copula.

`pmat_f1` Partial correlation matrix for the 1-factor tree copula.

`pmat_f2` Partial correlation matrix for the 2-factor tree copula.

Author(s)

Sayed H. Kadhem
Aristidis K. Nikoloulopoulos <a.nikoloulopoulos@uea.ac.uk>

References

Fontenla, M. (2014). *optrees: Optimal Trees in Weighted Graphs*. R package version 1.0.

Kadhem, S.H. and Nikoloulopoulos, A.K. (2022) Factor tree copula models for item response data. *Arxiv e-prints*, <arXiv: 2201.00339>. https://arxiv.org/abs/2201.00339.

Nikoloulopoulos, A.K. and Joe, H. (2015) Factor copula models with item response data. *Psychometrika*, **80**, 126–150. doi:10.1007/s1133601393874.

Olsson, F. (1979) Maximum likelihood estimation of the polychoric correlation coefficient. *Psychometrika*, **44**, 443–460.

Prim, R. C. (1957) Shortest connection networks and some generalizations. *The Bell System Technical Journal*, **36**, 1389–1401.

Examples

```
#------------------------------------------------
# Setting quadreture points
nq <- 5
gl <- gauss.quad.prob(nq)
#------------------------------------------------
```

```
#                    PTSD Data
#------------------             -----------------
data(PTSD)
ydat=PTSD

#-----------------------------------------------
#vine tree structure
#selecting vine tree based on polychoric corr
rmat=polychoric0(ydat)$p
A.polychoric=selectFactorTrVine(y=ydat,rmat,alg=3)

#selecting bivariate copulas for the 1-2-factor tree
# listing the possible copula candidates:
copnames=c("gum", "rgum")

f1tree_model = copselect1FactorTree(ydat, A.polychoric$VineTreeA,
copnames, gl)
f2tree_model = copselect2FactorTree(ydat, A.polychoric$VineTreeA,
copnames, gl)
```

Select.StructuredFactor *Model selection of the bi-factor and second-order copula models for item response data*

Description

A heuristic algorithm that automatically selects the bivariate parametric copula families for the bi-factor and second-order copula models for item response data.

Usage

```
selectBifactor(y, grpsize, copnames, gl)
selectSecond_order(y, grpsize, copnames, gl)
```

Arguments

y	$n \times d$ matrix with the item reponse data, where n and d is the number of observations and variables, respectively.
grpsize	vector indicating the size for each group, e.g., c(4,4,4) indicating four items in all three groups.
copnames	A vector with the names of possible candidates of bivariate copulas to be selected for the bi-factor and second-order copula models for item response data. Choices are "bvn" for BVN, "bvtν" with $\nu = \{1, \ldots, 9\}$ degrees of freedom for t-copula, "frk" for Frank, "gum" for Gumbel, "rgum" for reflected Gumbel, "1rgum" for 1-reflected Gumbel, "2rgum" for 2-reflected Gumbel.
gl	Gauss legendre quardrature nodes and weights.

Details

The linking copulas at each factor are selected with a sequential algorithm under the initial assumption that linking copulas are BVN, and then sequentially copulas with tail dependence are assigned to any of pairs where necessary to account for tail asymmetry.

Value

A list containing the following components:

`‘‘common factor’’`
: The selected bivariate linking copulas for the common factor (Bi-factor: copulas link items with the common factor. Second-order: copulas link group-specific factors with the common factor).

`‘‘group-specific factor’’`
: The selected bivariate linking copulas for the items with group-speicifc factors.

`log-likelihood`
: The maximized joint log-likelihood.

`taus`
: The estimated copula parameters in Kendall’s tau scale.

Author(s)

Sayed H. Kadhem <s.kadhem@uea.ac.uk>
Aristidis K. Nikoloulopoulos <a.nikoloulopoulos@uea.ac.uk>

References

Kadhem, S.H. and Nikoloulopoulos, A.K. (2023) Bi-factor and second-order copula models for item response data. *Psychometrika*, **88**, 132–157. doi:10.1007/s11336022098942.

Examples

```
#------------------------------------------------
# Setting quadreture points
nq <- 15
gl <- gauss.quad.prob(nq)
#------------------------------------------------
#                     TAS Data
#------------------             -----------------
data(TAS)
#using a subset of the data
#group1: 1,3,6,7,9,13,14
grp1=c(1,3)
#group2: 2,4,11,12,17
grp2=c(2,4)
#group3: 5,8,10,15,16,18,19,20
grp3=c(5,8)
#Rearrange items within testlets
set.seed(123)
i=sample(1:nrow(TAS),500)
ydat=TAS[i,c(grp1,grp2,grp3)]
```

```
#size of each group
g1=length(grp1)
g2=length(grp2)
g3=length(grp3)
grpsize=c(g1,g2,g3)

# listing possible copula candidates:
copnames=c("bvt2","bvt3")

Bifactor_model = selectBifactor(ydat, grpsize, copnames, gl)
```

---

semicorr *Diagnostics to detect tail dependence or tail asymmetry.*

Description

The sample versions of the correlation ρ_N, upper semi-correlation ρ_N^+ (correlation in the joint upper quadrant) and lower semi-correlation ρ_N^- (correlation in the joint lower quadrant). These are sample linear (when both variables are continuous), polychoric (when both variables are ordinal), and polyserial (when one variable is continuous and the other is ordinal) correlations.

Usage

```
semicorr(dat,type)
```

Arguments

dat	Data frame of mixed continuous and ordinal data.
type	A vector with 1's for the location of continuous data and 2's for the location of ordinal data.

Value

A matrix containing the following components for semicorr():

rho	ρ_N.
lrho	ρ_N^-.
urho	ρ_N^+.

Author(s)

Sayed H. Kadhem <s.kadhem@uea.ac.uk>
Aristidis K. Nikoloulopoulos <a.nikoloulopoulos@uea.ac.uk>

References

Joe, H. (2014). *Dependence Modelling with Copulas*. Chapman and Hall/CRC.

Kadhem, S.H. and Nikoloulopoulos, A.K. (2021) Factor copula models for mixed data. *British Journal of Mathematical and Statistical Psychology*, **74**, 365–403. doi:10.1111/bmsp.12231.

Examples

```
#-------------------------------------------------
#                     PE Data
#------------------            ------------------
data(PE)
#correlation
continuous.PE1 <- -PE[,1]
continuous.PE <- cbind(continuous.PE1, PE[,2])
categorical.PE <- data.frame(apply(PE[, 3:5], 2, factor))
nPE <- cbind(continuous.PE, categorical.PE)

#-------------------------------------------------
# Semi-correlations-------------------------------
#-------------------------------------------------
# Exclude the dichotomous variable
sem.PE = nPE[,-3]
semicorr.PE = semicorr(dat = sem.PE, type = c(1,1,2,2))
#-------------------------------------------------
#-------------------------------------------------
#                    GSS Data
#------------------            ------------------
data(GSS)
attach(GSS)
continuous.GSS <- cbind(INCOME,AGE)
ordinal.GSS <- cbind(DEGREE,PINCOME,PDEGREE)
count.GSS <- cbind(CHILDREN,PCHILDREN)

# Transforming the COUNT variables to ordinal
# count1 : CHILDREN
count1  = count.GSS[,1]
count1[count1 > 3] = 3

# count2: PCHILDREN
count2  = count.GSS[,2]
count2[count2 > 7] = 7

# Combining both transformed count variables
ncount.GSS = cbind(count1, count2)

# Combining ordinal and transformed count variables
categorical.GSS <- cbind(ordinal.GSS, ncount.GSS)
categorical.GSS <- data.frame(apply(categorical.GSS, 2, factor))

# combining continuous and categorical variables
nGSS = cbind(continuous.GSS, categorical.GSS)
```

```
#-----------------------------------------------
# Semi-correlations------------------------------
#-----------------------------------------------
semicorr.GSS = semicorr(dat = nGSS, type = c(1, 1, rep(2,5)))
```

TAS *Toronto Alexithymia Scale (TAS)*

Description

The Toronto Alexithymia Scale is the most utilized measure of alexithymia in empirical research and is composed of $d = 20$ items that can be subdivided into $G = 3$ non-overlapping groups: $d_1 = 7$ items to assess difficulty identifying feelings (DIF), $d_2 = 5$ items to assess difficulty describing feelings (DDF) and $d_3 = 8$ items to assess externally oriented thinking (EOT). Students were 17 to 25 years old and 58% of them were female and 42% were male. They were asked to respond to each item using one of K = 5 categories: “1 = completely disagree”, “2 = disagree”, “3 = neutral”, “4 = agree”, “5 = completely agree”.

Usage

```
data(TAS)
```

Format

A data frame with 1925 observations on the following 20 items:

`DIF` items: 1,3,6,7,9,13,14.

`DDF` items: 2,4,11,12,17.

`EOT` items: 5,8,10,15,16,18,19,20.

Source

Briganti, G. and Linkowski, P. (2020). Network approach to items and domains from the toronto alexithymia scale. *Psychological Reports*, **123**, 2038–2052.

Williams, D. and Mulder, J. (2020). *BGGM: Bayesian Gaussian Graphical Models*. R package version 1.0.0.

transformation *Continuous/count to ordinal responses*

Description

Transforming a continuous/count to ordinal variable with K categories.

Usage

```
continuous2ordinal(continuous, categ)
count2ordinal(count, categ)
```

Arguments

continuous	Matrix of continuous data.
count	Matrix of count data.
categ	The number of categories K.

Examples

```
#-----------------------------------------------
#                    PE Data
#------------------            -----------------
data(PE)
continuous.PE <- PE[, 1:2]

#Transforming the continuous to ordinal data :
tcontinuous = continuous2ordinal(continuous.PE, categ=5)
table(tcontinuous)

#Transforming the count to ordinal data:
set.seed(12345)
count.data = rpois(1000, 3)
tcount = count2ordinal(count.data, 5)
table(tcount)
```

Vuong.Factor *Vuong's test for the comparison of factor copula models*

Description

Vuong (1989)'s test for the comparison of non-nested factor copula models for mixed data. We compute the Vuong's test between the factor copula model with BVN copulas (that is the standard factor model) and a competing factor copula model to reveal if the latter provides better fit than the standard factor model.

Usage

```
vuong.1f(cpar.bvn, cpar, copF1, continuous, ordinal, count, gl, param)
vuong.2f(cpar.bvn, cpar, copF1, copF2, continuous, ordinal, count, gl, param)
```

Arguments

`cpar.bvn`	copula parameters of the factor copula model with BVN copulas.
`cpar`	copula parameters of the competing factor copula model.
`copF1`	copula names for the first factor of the competing factor copula model.
`copF2`	copula names for the second factor of the competing factor copula model.
`continuous`	matrix of continuous data.
`ordinal`	matrix of ordinal data.
`count`	matrix of count data.
`gl`	gauss-legendre quardature points.
`param`	parameterization of estimated copula parameters. If FALSE, then cpar are the actual copula parameters without any transformation/reparamterization.

Value

A vector containing the following components:

`z`	the test statistic.
`p.value`	the p-value.
`CI.left`	lower/left endpoint of 95% confidence interval.
`CI.right`	upper/right endpoint of 95% confidence interval.

Author(s)

Sayed H. Kadhem <s.kadhem@uea.ac.uk>
Aristidis K. Nikoloulopoulos <a.nikoloulopoulos@uea.ac.uk>

References

Kadhem, S.H. and Nikoloulopoulos, A.K. (2021) Factor copula models for mixed data. *British Journal of Mathematical and Statistical Psychology*, **74**, 365–403. doi:10.1111/bmsp.12231.

Vuong, Q.H. (1989). Likelihood ratio tests for model selection and non-nested hypotheses. *Econometrica*, **57**, 307–333.

Examples

```
#------------------------------------------------
# Setting quadreture points
nq <- 25
gl <- gauss.quad.prob(nq)
#------------------------------------------------
```

```
#                     PE Data
#------------------             -----------------
data(PE)
continuous.PE1 = -PE[,1]
continuous.PE2 = PE[,2]
continuous.PE <- cbind(continuous.PE1, continuous.PE2)
categorical.PE <- PE[, 3:5]
d <- ncol(PE)
#-----------------------------------------------
#                   Estimation
#------------------             -----------------
# factor copula model with BVN copulas
cop1f.PE.bvn <- rep("bvn", d)
PE.bvn1f <- mle1factor(continuous.PE, categorical.PE,
count=NULL, copF1=cop1f.PE.bvn, gl, hessian = TRUE)

# Selected factor copula model
cop1f.PE <- c("joe", "joe", "rjoe", "joe", "gum")
PE.selected1f <- mle1factor(continuous.PE, categorical.PE,
count=NULL, copF1=cop1f.PE, gl, hessian = TRUE)
#-----------------------------------------------
#                   Vuong's test
#------------------             -----------------
v1f.PE.selected <- vuong.1f(PE.bvn1f$cpar$f1,
PE.selected1f$cpar$f1,cop1f.PE, continuous.PE,
categorical.PE, count=NULL, gl, param=FALSE)
```

`Vuong.FactorTree` *Vuong's test for the comparison of 1- and 2-factor tree copula models for item response data*

Description

The Vuong's test (Vuong,1989) is the sample version of the difference in Kullback-Leibler divergence between two models and can be used to differentiate two parametric models which could be non-nested. For the Vuong's test we provide the 95% confidence interval of the Vuong's test statistic (Joe, 2014, page 258). If the interval does not contain 0, then the best fitted model according to the AICs is better if the interval is completely above 0.

Usage

```
vuong_FactorTree(models, y, A.m1, tau.m1, copnames.m1,
tau.m2, copnames.m2,A.m2,nq)
```

Arguments

`models` choose a number from (1,2,3,4,5,6,7). **1**: Model 1 is 1-factor tree copula, Model 2 is 1-factor tree copula. **2**: Model 1 is 2-factor tree copula, Model 2 is 2-factor

	tree copula. **3**: Model 1 is 1-factor tree copula, Model 2 is 2-factor tree copula. **4**: Model 1 is 1-factor copula, Model 2 is 1-factor tree copula. **5**: Model 1 is 1-factor copula, Model 2 is 2-factor tree copula. **6**: Model 1 is 2-factor copula, Model 2 is 1-factor tree copula. **7**: Model 1 is 2-factor copula, Model 2 is 2-factor tree copula.
`y`	$n \times d$ matrix with the item reponse data, where n and d is the number of observations and variables, respectively.
`A.m1`	$d \times d$ vine array for Model 1, if it is 1-factor tree, or 2-factor tree otherwise keep as NULL.
`A.m2`	$d \times d$ vine array for Model 2, note only the first row and diagnoal values are used for the 1-truncated vine model.
`tau.m1`	vector of copula paramters in Kendall's τ for model 1.
`tau.m2`	vector of copula paramters in Kendall's τ for model 2.
`copnames.m1`	vector of names of copula families for model 1.
`copnames.m2`	vector of names of copula families for model 2.
`nq`	Number of Gauss legendre quardrature points.

Value

A vector containing the following components:

`z`	the test statistic.
`p.value`	the p-value.
`CI.left`	lower/left endpoint of 95% confidence interval.
`CI.right`	upper/right endpoint of 95% confidence interval.

Author(s)

Sayed H. Kadhem
Aristidis K. Nikoloulopoulos <a.nikoloulopoulos@uea.ac.uk>

References

Joe, H. (2014). *Dependence Modelling with Copulas*. Chapman and Hall/CRC.

Kadhem, S.H. and Nikoloulopoulos, A.K. (2022b) Factor tree copula models for item response data. *Arxiv e-prints*, <arXiv: 2201.00339>. https://arxiv.org/abs/2201.00339.

Vuong, Q.H. (1989). Likelihood ratio tests for model selection and non-nested hypotheses. *Econometrica*, **57**, 307–333.

Examples

```
#-------------------------------------------------
# Setting quadreture points
nq <- 25
#-------------------------------------------------
```

```
#                    PTSD Data
#------------------             -----------------
data(PTSD)
ydat=PTSD

d=ncol(ydat);d
n=nrow(ydat);n

#vine tree structure
#selecting vine tree based on polychoric corr
rmat=polychoric0(ydat)$p
A.polychoric=selectFactorTrVine(y=ydat,rmat,alg=3)
A.polychoric=A.polychoric$VineTreeA
#-------------------------------------------------
# M1 1-factor tree - M2 1-factor tree
tau.m1 = rep(0.4,d*2-1)
copnames.m1 = rep("bvn",d*2-1)
tau.m2 = rep(0.4,d*2-1)
copnames.m2 = rep("rgum",d*2-1)
vuong.1factortree = vuong_FactorTree(models=1, ydat,
A.m1=A.polychoric, tau.m1, copnames.m1, tau.m2,
copnames.m2,A.m2=A.polychoric,nq)
#-------------------------------------------------
# M1 2-factor tree - M2 2-factor tree
tau.m1 = rep(0.4,d*3-1)
copnames.m1 = rep("bvn",d*3-1)
tau.m2 = rep(0.4,d*3-1)
copnames.m2 = rep("rgum",d*3-1)

vuong.2factortree = vuong_FactorTree(models=2, ydat,
A.m1=A.polychoric, tau.m1, copnames.m1, tau.m2,
copnames.m2,A.m2=A.polychoric,nq)

#-------------------------------------------------
# M1 1-factor tree - M2 2-factor tree
tau.m1 = rep(0.4,d*2-1)
copnames.m1 = rep("bvn",d*2-1)

tau.m2 = rep(0.4,d*3-1)
copnames.m2 = rep("rgum",d*3-1)

vuong.12factortree = vuong_FactorTree(models=3, ydat,
A.m1=A.polychoric, tau.m1, copnames.m1, tau.m2,
copnames.m2,A.m2=A.polychoric,nq)
```

Vuong.StructuredFactor *Vuong's test for the comparison of bi-factor and second-order copula models*

Description

The Vuong's test (Vuong,1989) is the sample version of the difference in Kullback-Leibler divergence between two models and can be used to differentiate two parametric models which could be non-nested. For the Vuong's test we provide the 95% confidence interval of the Vuong's test statistic (Joe, 2014, page 258). If the interval does not contain 0, then the best fitted model according to the AICs is better if the interval is completely above 0.

Usage

```
vuong_structured(models, cpar.m1, copnames.m1, cpar.m2,
copnames.m2, y, grpsize)
```

Arguments

`models`	choose a number from (1,2,3,4). **1**: Model1 is bifactor, Model2 is bifactor. **2**: Model1 is second-order, Model2 is second-order. **3**: Model1 is second-order, Model2 is bifactor. **4**: Model1 is bifactor, Model2 is nested.
`cpar.m1`	vector of copula paramters for model 1, starting with copula parameters that link the items with common factor (bifactor), or group factors with common factor (second-order).
`cpar.m2`	vector of copula paramters for model 2, starting with copula parameters that link the items with common factor (bifactor), or group factors with common factor (second-order).
`copnames.m1`	vector of names of copula families for model 1, starting with copulas that link the items with common factor (bifactor), or group factors with common factor (second-order).
`copnames.m2`	vector of names of copula families for model 2, starting with copulas that link the items with common factor (bifactor), or group factors with common factor (second-order).
`y`	matrix of ordinal data.
`grpsize`	vector indicating the size for each group, e.g., c(4,4,4) indicating four items in all three groups.

Value

A vector containing the following components:

`z`	the test statistic.
`p.value`	the p-value.
`CI.left`	lower/left endpoint of 95% confidence interval.
`CI.right`	upper/right endpoint of 95% confidence interval.

Author(s)

Sayed H. Kadhem <s.kadhem@uea.ac.uk>
Aristidis K. Nikoloulopoulos <a.nikoloulopoulos@uea.ac.uk>

References

Joe, H. (2014). *Dependence Modelling with Copulas*. Chapman and Hall/CRC.

Kadhem, S.H. and Nikoloulopoulos, A.K. (2023) Bi-factor and second-order copula models for item response data. *Psychometrika*, **88**, 132–157. doi:10.1007/s11336022098942.

Vuong, Q.H. (1989). Likelihood ratio tests for model selection and non-nested hypotheses. *Econometrica*, **57**, 307–333.

Examples

```
#-------------------------------------------------
# Setting quadreture points
nq <- 25
gl <- gauss.quad.prob(nq)
#-------------------------------------------------
#                    TAS Data
#------------------             -----------------
data(TAS)
grp1=c(1,3,6,7,9,13,14)
grp2=c(2,4,11,12,17)
grp3=c(5,8,10,15,16,18,19,20)
ydat=TAS[,c(grp1,grp2,grp3)]

d=ncol(ydat);d
n=nrow(ydat);n

#Rearrange items within testlets
g1=length(grp1)
g2=length(grp2)
g3=length(grp3)

grpsize=c(g1,g2,g3)#group size
#number of groups
ngrp=length(grpsize)
#-------------------------------------------------
# M1 bifactor - M2 bifactor
cpar.m1 = rep(0.6,d*2)
copnames.m1 = rep("bvn",d*2)
cpar.m2 = rep(1.6,d*2)
copnames.m2 = rep("rgum",d*2)
vuong.bifactor = vuong_structured(models=1, cpar.m1, copnames.m1,
                 cpar.m2, copnames.m2,
                 y=ydat, grpsize)

# M1 seconod-order - M2 seconod-order
cpar.m1 = rep(0.6,d+ngrp)
copnames.m1 = rep("bvn",d+ngrp)
cpar.m2 = rep(1.6,d+ngrp)
copnames.m2 = rep("rgum",d+ngrp)
vuong.second_order = vuong_structured(models=2, cpar.m1,
    copnames.m1, cpar.m2, copnames.m2, y=ydat, grpsize)
```

```
# M1 seconod-order - M2 bifactor
cpar.m1 = rep(0.6,d+ngrp)
copnames.m1 = rep("bvn",d+ngrp)
cpar.m2 = rep(1.6,d*2)
copnames.m2 = rep("rgum",d*2)
vuong.2ndO_bif = vuong_structured(models=3, cpar.m1, copnames.m1,
                 cpar.m2, copnames.m2,
                 y=ydat, grpsize)

# M1 bifactor - M2 seconod-order
cpar.m1 = rep(0.6,d*2)
copnames.m1 = rep("bvn",d*2)
cpar.m2 = rep(1.6,d+ngrp)
copnames.m2 = rep("rgum",d+ngrp)
vuong.bif_2ndO = vuong_structured(models=4, cpar.m1, copnames.m1,
                 cpar.m2, copnames.m2,
                 y=ydat, grpsize)
```

Index