Package ‘FastCUB’

October 12, 2022

Title Fast EM and Best-Subset Selection for CUB Models for Rating Data

Version 0.0.2

Author Rosaria Simone <rosaria.simone@unina.it>

Maintainer Rosaria Simone <rosaria.simone@unina.it>

Description For ordinal rating data, consider the accelerated Expectation-Maximization algorithm to estimate and test models within the family of CUB models (where CUB stands for Combination of a discrete Uniform and a shifted Binomial distributions). The procedure is built upon Louis' identity for the observed information matrix. Best-subset variable selection for CUB regression models is then implemented on such basis. The methods here implemented are illustrated and discussed in the preprint available from Researchgate by Simone R. (2020) <https://tinyurl.com/vvk563e>.

Depends R (>= 2.15.2), Formula

Encoding UTF-8

Imports methods, utils, CUB

LazyData true

RoxygenNote 6.1.1

NeedsCompilation no

Repository CRAN

Date/Publication 2020-02-12 22:30:02 UTC

R topics documented:

`bestcub` *Best-subset variable selection for CUB models via fast EM algorithm*

Description

Perform a best-subset search for CUB models on the basis of the BIC index, by combining all possible covariates' specification for feeling and for uncertainty parameters

Usage

```
bestcub(ordinal,m,Y,W,toler=1e-4,maxiter=200,iterc=5,alpha=0.05,mix=FALSE,
tolmix=1e+2,fmix=NULL,invgen=TRUE,verbose=FALSE)
```

Arguments

`ordinal`	Vector of ordinal responses
`m`	Number of ordinal categories
`Y`	Matrix of selected covariates for the uncertainty parameter
`W`	Matrix of selected covariates for the feeling parameter
`toler`	Fixed convergence tolerance for final estimates
`maxiter`	Maximum number of iterations allowed for running the optimization algorithm
`iterc`	Iteration from which the acceleration strategy starts
`alpha`	Significant level for Wald test
`mix`	Logical: should a first preliminary standard EM be run at toler equal to tolmix? (default is FALSE)

tolmix	Convergence tolerance for first preliminary EM (if mix=TRUE).
fmix	Fraction of iteration needed for first preliminary EM (if mix=TRUE). Default is null.
invgen	Logical: should the recursive formula for the inverse of the information matrix be considered? (Default is TRUE)
verbose	Logical: should messages about acceleration steps be printed out? (Default is FALSE)

Value

A list containing the following results:

vsel	List of all estimated models (with the accelerated EM) as FastCUB objects
bestmodel	FastCUB object corresponding to the best CUB model (according to BIC), if not null
bestW	Names of covariates for feeling in the best model with all significant effect
bestY	Names of covariates for feeling in the best model with all significant effect
param	ML estimates of the best model
se	Estimated standard errors for the best model
bic	BIC index of the best (significant) model
mattime	Matrix of computational time for each of the estimated model
matiter	Matrix of number of iterations occurred for each of the estimated model

See Also

fastCUB

Examples

```
library(FastCUB)
data(univer)
ordinal<-univer$global
m<-7
Y<-univer[,c(2,3,4)]
W<-univer[,8:11]
## Search for the best CUB model with covariates only for feeling
best0q<-bestcub(ordinal,m,Y=NULL,W,toler=1e-4,maxiter=100,iterc=5,alpha=0.05,invgen=TRUE)
## Search for the best CUB model with covariates only for uncertainty
bestp0<-bestcub(ordinal,m,Y,W=NULL,toler=1e-4,maxiter=100,iterc=5,alpha=0.05,invgen=TRUE)
## Search for the best CUB model with covariates for both parameters
bestpq<-bestcub(ordinal,m,Y,W,toler=1e-4,maxiter=100,iterc=5,alpha=0.05,invgen=TRUE,
    mix=TRUE,tolmix=1e+3,fmix=1)
final<-bestpq$bestmodel; summary(final)
```

`BIC.fastCUB` *S3 BIC method for class "fastCUB"*

Description

S3 BIC method for objects of class `fastCUB`.

Usage

```
## S3 method for class 'fastCUB'
BIC(object, ...)
```

Arguments

`object`	An object of class "fastCUB"
`...`	Other arguments

Value

BIC index for the fitted model.

See Also

`logLik`, `fastCUB`

`bitcsi` *Shifted Binomial probabilities of ordinal responses*

Description

Compute the shifted Binomial probabilities of ordinal responses.

Usage

```
bitcsi(m,ordinal,csi)
```

Arguments

`m`	Number of ordinal categories
`ordinal`	Vector of ordinal responses
`csi`	Feeling parameter of the shifted Binomial distribution

Value

A vector of the same length as `ordinal`, where each entry is the shifted Binomial probability of the corresponding observation.

References

Piccolo D. (2003). On the moments of a mixture of uniform and shifted binomial random variables, *Quaderni di Statistica*, **5**, 85–104

See Also

probcub00, probcubp0, probcub0q

Examples

```
data(univer)
m<-7
csi<-0.7
ordinal<-univer$informat
pr<-bitcsi(m,ordinal,csi)
```

---

bitgama *Shifted Binomial distribution with covariates*

Description

Return the shifted Binomial probabilities of ordinal responses where the feeling component is explained by covariates via a logistic link.

Usage

```
bitgama(m,ordinal,W,gama)
```

Arguments

m	Number of ordinal categories
ordinal	Vector of ordinal responses
W	Matrix of covariates for the feeling component
gama	Vector of parameters for the feeling component, with length equal to NCOL(W)+1 to account for an intercept term (first entry of gama)

Value

A vector of the same length as ordinal, where each entry is the shifted Binomial probability for the corresponding observation and feeling value.

See Also

logis, probcub0q, probcubpq

Examples

```
n<-100
m<-7
W<-sample(c(0,1),n,replace=TRUE)
gama<-c(0.2,-0.2)
csivett<-logis(W,gama)
ordinal<-rbinom(n,m-1,csivett)+1
pr<-bitgama(m,ordinal,W,gama)
```

---

coef.fastCUB — *S3 Method: coef for class "fastCUB"*

Description

S3 method: coef for objects of class `fastCUB`.

Usage

```
## S3 method for class 'fastCUB'
coef(object, ...)
```

Arguments

`object`	An object of class `fastCUB`
`...`	Other arguments

Details

Returns estimated values of coefficients of the fitted model

Value

ML estimates of parameters of the fitted CUB model.

See Also

`fastCUB`, `summary`

`cormat` *Correlation matrix for estimated model*

Description

Compute parameter correlation matrix for estimated model as returned by an object of class "fastCUB".

Usage

```
cormat(object,digits=options()$digits)
```

Arguments

`object`	An object of class "fastCUB"
`digits`	Number of significant digits to be printed. Default is `options()$digits`

Value

Parameters correlation matrix for fitted fastCUB models.

See Also

`fastCUB`, `vcov`

`dissim` *Normalized dissimilarity measure*

Description

Compute the normalized dissimilarity measure between observed relative frequencies and estimated (theoretical) probabilities of a discrete distribution.

Usage

```
dissim(proba,probb)
```

Arguments

`proba`	Vector of observed relative frequencies
`probb`	Vector of estimated (theoretical) probabilities

Value

Numeric value of the dissimilarity index, assessing the distance to a perfect fit.

Examples

```
proba<-c(0.01,0.03,0.08,0.07,0.27,0.37,0.17)
probb<-c(0.04,0.04,0.05,0.10,0.21,0.32,0.24)
dissim(proba,probb)
```

---

fastCUB — *Main function for fast estimation CUB models*

Description

Main function to estimate and validate a CUB model for explaining uncertainty and feeling for given ratings, with or without covariates, on the basis of Louis' identity for the information matrix and the derived accelerated estimation.

Usage

```
fastCUB(Formula, data, mix=FALSE, tolmix=1e+2,fmix= NULL,...)
```

Arguments

`Formula`	Object of class Formula with two right-hand side: the first for uncertainty covariates, the second for feeling covariates.
`data`	Data frame from which model matrices and response variables are taken.
`mix`	Logical: should a first preliminary standard EM be run at toler equal to tolmix? (default is FALSE).
`tolmix`	Convergence tolerance for first preliminary EM (if mix=TRUE).
`fmix`	Fraction of iteration needed for preliminary EM (if mix=TRUE). Default is null.
`...`	Additional arguments to be passed for the specification of the model and the acceleration steps.

Details

This is the main function for CUB models, which calls for the corresponding functions whenever covariates are specified. It performs maximum likelihood estimation via the E-M algorithm for CUB models and extensions based on the Louis'identity for the observed information matrix.

Value

An object of the class "fastCUB": returns a list containing the following results:

`estimates`	Maximum likelihood estimates of model parameters
`loglik`	Log-likelihood function at the final estimates
`varmat`	Variance-covariance matrix of final estimates
`niter`	Number of executed iterations
`BIC`	BIC index for the estimated model
`parnames`	Names for model parameters

See Also

probcub00, probcubp0, probcub0q, probcubpq

Examples

```
library(FastCUB)
data(univer)
ordinal<-univer$global
m<-7
effe<-with(univer, Formula(global~0|gender+freqserv+age +changefa))
cub0q<-fastCUB(effe,data=univer,m=7, maxiter=100,toler=1e-8,mix=TRUE,verbose=FALSE)
summary(cub0q)
## Fast EM for  CUB model with covariates only for uncertainty
effe<-with(univer, Formula(global~gender+freqserv+age +changefa|0))
cubp0<-fastCUB(effe,data=univer,m=7, maxiter=100,toler=1e-8,iterc=5,verbose=TRUE)
## Fast EM for  CUB model with covariates for both feeling and uncertainty
effe<-with(univer, Formula(global~gender+freqserv+age +changefa|gender+freqserv+age +changefa))
cubpq<-fastCUB(effe,data=univer,m=7, maxiter=100,toler=1e-8,iterc=5)
summary(cubpq)
BIC(cubpq)
```

`fastCUB_package` *fastCUB package*

Description

The package implements Louis' identity for CUB models for rating data, to retrieve the observed information matrix within the Expectation-Maximization algorithm. On this basis, an acceleration strategy is derived and then a best-subset search through the covariate space for feeling and uncertainty is implemented.

Details

Package:	fastCUB
Type:	Package
Version:	0.0.2
Date:	2019-03-05
License: GPL-2 | GPL-3	

Author(s)

Rosaria Simone

References

Louis T.A. (1982). Finding the Observed Information Matrix when Using the EM Algorithm, *Journal of the Royal Statistical Society, Series B*, **44**, 226–233
Simone R. (2020). An Accelerated EM algorithm for mixture models with uncertainty for rating data. *under Review*

`fitted.fastCUB` *S3 method "fitted" for class "fastCUB"*

Description

S3 method fitted for objects of class `fastCUB`.

Usage

```
## S3 method for class 'fastCUB'
fitted(object, ...)
```

Arguments

`object`	An object of class `fastCUB`
`...`	Other arguments

Details

Returns the fitted probability distribution for GEM models with no covariates. If only one dichotomous covariate is included in the model to explain some components, it returns the fitted probability distribution for each profile.

See Also

`fastCUB`

`inibest` *Preliminary estimators for CUB models without covariates*

Description

Compute preliminary parameter estimates of a CUB model without covariates for given ordinal responses. These preliminary estimators are used within the package code to start the E-M algorithm.

Usage

```
inibest(m,freq)
```

Arguments

`m`	Number of ordinal categories
`freq`	Vector of the absolute frequencies of given ordinal responses

Value

A vector (π, ξ) of the initial parameter estimates for a CUB model without covariates, given the absolute frequency distribution of ordinal responses

References

Iannario M. (2009). A comparison of preliminary estimators in a class of ordinal data models, *Statistica & Applicazioni*, **VII**, 25–44
Iannario M. (2012). Preliminary estimators for a mixture model of ordinal data, *Advances in Data Analysis and Classification*, **6**, 163–184

See Also

`inibestgama`

Examples

```
m<-9
freq<-c(10,24,28,36,50,43,23,12,5)
estim<-inibest(m,freq)
pai<-estim[1]
csi<-estim[2]
```

`inibestgama` *Preliminary parameter estimates of a CUB model with covariates for feeling*

Description

Compute preliminary parameter estimates for the feeling component of a CUB model fitted to ordinal responses These estimates are set as initial values for parameters to start the E-M algorithm.

Usage

```
inibestgama(m,ordinal,W)
```

Arguments

`m`	Number of ordinal categories
`ordinal`	Vector of ordinal responses
`W`	Matrix of selected covariates for explaining the feeling component

Value

A vector of length equal to NCOL(W)+1, whose entries are the preliminary estimates of the parameters for the feeling component, including an intercept term as first entry.

References

Iannario M. (2008). Selecting feeling covariates in rating surveys, *Rivista di Statistica Applicata*, **20**, 103–116
Iannario M. (2009). A comparison of preliminary estimators in a class of ordinal data models, *Statistica & Applicazioni*, **VII**, 25–44
Iannario M. (2012). Preliminary estimators for a mixture model of ordinal data, *Advances in Data Analysis and Classification*, **6**, 163–184

See Also

inibest

Examples

```
data(univer)
m<-7; ordinal<-univer$global; cov<-univer$gender
ini<-inibestgama(m,ordinal,W=cov)
```

---

invmatgen *Recursive computation of the inverse of a matrix*

Description

Compute the variance-covariance matrix of the incomplete score vector involved in Louis' identity for the observed information matrix

Usage

```
invmatgen(G,H,listE)
```

Arguments

G	Primary matrix for the sum decomposition of $G+H$
H	Secondary matrix for the sum decomposition of $G+H$
listE	Auxiliary matrices that sum up to H

Value

The inverse of matrix G + H computed recursively thanks to matrices listed in listE

References

Miller K. (1981). On the inverse of the sum of matrices, *Mathematics Magazine*, **54**, 67–72

See Also

fastCUB

`logis` *The logistic transform*

Description

Create a matrix YY binding array `Y` with a vector of ones, placed as the first column of YY. It applies the logistic transform componentwise to the standard matrix multiplication between YY and `param`.

Usage

```
logis(Y,param)
```

Arguments

`Y`	A generic matrix or one dimensional array
`param`	Vector of coefficients, whose length is NCOL(Y) + 1 (to consider also an intercept term)

Value

Return a vector whose length is NROW(Y) and whose i-th component is the logistic function at the scalar product between the i-th row of YY and the vector `param`.

Examples

```
n<-50
Y<-sample(c(1,2,3),n,replace=TRUE)
param<-c(0.2,0.7)
logis(Y,param)
```

`logLik.fastCUB` *logLik S3 Method for class "fastCUB"*

Description

S3 method: logLik() for objects of class "fastCUB".

Usage

```
## S3 method for class 'fastCUB'
logLik(object, ...)
```

Arguments

`object`	An object of class "fastCUB"
`...`	Other arguments

Value

Log-likelihood at the final ML estimates for parameters of the fitted fastCUB model.

See Also

`fastCUB`

`makeplot` *Plot facilities for fastCUB objects*

Description

Plot facilities for objects of class "fastCUB".

Usage

```
makeplot(object)
```

Arguments

`object`	An object of class "fastCUB"

Details

Returns a plot comparing fitted probabilities and observed relative frequencies for GEM models without covariates. If only one explanatory dichotomous variable is included in the model for one or all components, then the function returns a plot comparing the distributions of the responses conditioned to the value of the covariate.

`print.fastCUB` *S3 method: print for class "fastCUB"*

Description

S3 method print for objects of class `fastCUB`.

Usage

```
## S3 method for class 'fastCUB'
print(x, ...)
```

Arguments

`x`	An object of class `fastCUB`
`...`	Other arguments

Value

Brief summary results of the fitting procedure, including parameter estimates, their standard errors and the executed call.

`probbit` *Probability distribution of a shifted Binomial random variable*

Description

Return the shifted Binomial probability distribution.

Usage

```
probbit(m,csi)
```

Arguments

`m`	Number of ordinal categories
`csi`	Feeling parameter

Value

The vector of the probability distribution of a shifted Binomial model.

See Also

`bitcsi`, `probcub00`

Examples

```
m<-7
csi<-0.7
pr<-probbit(m,csi)
plot(1:m,pr,type="h",main="Shifted Binomial probability distribution",xlab="Categories")
points(1:m,pr,pch=19)
```

---

probcub00 *Probability distribution of a CUB model without covariates*

Description

Compute the probability distribution of a CUB model without covariates.

Usage

```
probcub00(m,pai,csi)
```

Arguments

m	Number of ordinal categories
pai	Uncertainty parameter
csi	Feeling parameter

Value

The vector of the probability distribution of a CUB model.

References

Piccolo D. (2003). On the moments of a mixture of uniform and shifted binomial random variables. *Quaderni di Statistica*, **5**, 85–104

See Also

bitcsi, probcub0q, probcubp0, probcubpq

Examples

```
m<-9
pai<-0.3
csi<-0.8
pr<-probcub00(m,pai,csi)
plot(1:m,pr,type="h",main="CUB probability distribution",xlab="Ordinal categories")
points(1:m,pr,pch=19)
```

---

probcub0q — *Probability distribution of a CUB model with covariates for the feeling component*

Description

Compute the probability distribution of a CUB model with covariates for the feeling component.

Usage

```
probcub0q(m,ordinal,W,pai,gama)
```

Arguments

`m`	Number of ordinal categories
`ordinal`	Vector of ordinal responses
`W`	Matrix of covariates for explaining the feeling component NCOL(Y)+1 to include an intercept term in the model (first entry)
`pai`	Uncertainty parameter
`gama`	Vector of parameters for the feeling component, whose length equals NCOL(W)+1 to include an intercept term in the model (first entry)

Value

A vector of the same length as `ordinal`, whose i-th component is the probability of the i-th observation according to a CUB distribution with the corresponding values of the covariates for the feeling component and coefficients specified in `gama`.

References

Piccolo D. (2006). Observed Information Matrix for MUB Models, *Quaderni di Statistica*, **8**, 33–78

Piccolo D. and D'Elia A. (2008). A new approach for modelling consumers' preferences, *Food Quality and Preference*, **18**, 247–259

Iannario M. and Piccolo D. (2012). CUB models: Statistical methods and empirical evidence, in: Kenett R. S. and Salini S. (eds.), *Modern Analysis of Customer Surveys: with applications using R*, J. Wiley and Sons, Chichester, 231–258

See Also

`bitgama`, `probcub00`, `probcubp0`, `probcubpq`

Examples

```
data(relgoods)
m<-10
naord<-which(is.na(relgoods$Physician))
nacov<-which(is.na(relgoods$Gender))
na<-union(naord,nacov)
ordinal<-relgoods$Physician[-na]
W<-relgoods$Gender[-na]
pai<-0.44; gama<-c(-0.91,-0.7)
pr<-probcub0q(m,ordinal,W,pai,gama)
```

---

probcubp0 *Probability distribution of a CUB model with covariates for the uncertainty component*

Description

Compute the probability distribution of a CUB model with covariates for the uncertainty component.

Usage

```
probcubp0(m,ordinal,Y,bet,csi)
```

Arguments

`m`	Number of ordinal categories
`ordinal`	Vector of ordinal responses
`Y`	Matrix of covariates for explaining the uncertainty component
`bet`	Vector of parameters for the uncertainty component, whose length equals NCOL(Y) + 1 to include an intercept term in the model (first entry)
`csi`	Feeling parameter

Value

A vector of the same length as `ordinal`, whose i-th component is the probability of the i-th observation according to a CUB model with the corresponding values of the covariates for the uncertainty component and coefficients for the covariates specified in `bet`.

References

Piccolo D. (2006). Observed Information Matrix for MUB Models, *Quaderni di Statistica*, **8**, 33–78

Piccolo D. and D'Elia A. (2008). A new approach for modelling consumers' preferences, *Food Quality and Preference*, **18**, 247–259

Iannario M. and Piccolo D. (2012). CUB models: Statistical methods and empirical evidence, in: Kenett R. S. and Salini S. (eds.), *Modern Analysis of Customer Surveys: with applications using R*, J. Wiley and Sons, Chichester, 231–258

See Also

bitgama, probcub00, probcubpq, probcub0q

Examples

```
data(relgoods)
m<-10
naord<-which(is.na(relgoods$Physician))
nacov<-which(is.na(relgoods$Gender))
na<-union(naord,nacov)
ordinal<-relgoods$Physician[-na]
Y<-relgoods$Gender[-na]
bet<-c(-0.81,0.93); csi<-0.20
probi<-probcubp0(m,ordinal,Y,bet,csi)
```

---

probcubpq *Probability distribution of a CUB model with covariates for both feeling and uncertainty*

Description

Compute the probability distribution of a CUB model with covariates for both the feeling and the uncertainty components.

Usage

```
probcubpq(m,ordinal,Y,W,bet,gama)
```

Arguments

m	Number of ordinal categories
ordinal	Vector of ordinal responses
Y	Matrix of covariates for explaining the uncertainty component
W	Matrix of covariates for explaining the feeling component
bet	Vector of parameters for the uncertainty component, whose length equals NCOL(Y) + 1 to include an intercept term in the model (first entry)
gama	Vector of parameters for the feeling component, whose length equals NCOL(W)+1 to include an intercept term in the model (first entry)

Value

A vector of the same length as `ordinal`, whose i-th component is the probability of the i-th rating according to a CUB distribution with given covariates for both uncertainty and feeling, and specified coefficients vectors `bet` and `gama`, respectively.

References

Piccolo D. (2006). Observed Information Matrix for MUB Models, *Quaderni di Statistica*, **8**, 33–78
Piccolo D. and D'Elia A. (2008). A new approach for modelling consumers' preferences, *Food Quality and Preference*, **18**, 247–259
Iannario M. and Piccolo D. (2012). CUB models: Statistical methods and empirical evidence, in: Kenett R. S. and Salini S. (eds.), *Modern Analysis of Customer Surveys: with applications using R*, J. Wiley and Sons, Chichester, 231–258

See Also

`bitgama`, `probcub00`, `probcubp0`, `probcub0q`

Examples

```
data(relgoods)
m<-10
naord<-which(is.na(relgoods$Physician))
nacov<-which(is.na(relgoods$Gender))
na<-union(naord,nacov)
ordinal<-relgoods$Physician[-na]
W<-Y<-relgoods$Gender[-na]
gama<-c(-0.91,-0.7); bet<-c(-0.81,0.93)
probi<-probcubpq(m,ordinal,Y,W,bet,gama)
```

---

relgoods *Relational goods and Leisure time dataset*

Description

Dataset consists of the results of a survey aimed at measuring the evaluation of people living in the metropolitan area of Naples, Italy, with respect to of relational goods and leisure time collected in December 2014. Every participant was asked to assess on a 10 point ordinal scale his/her personal score for several relational goods (for instance, time dedicated to friends and family) and to leisure time. In addition, the survey asked respondents to self-evaluate their level of happiness by marking a sign along a horizontal line of 110 millimeters according to their feeling, with the left-most extremity standing for "extremely unhappy", and the right-most extremity corresponding to the status "extremely happy".

Usage

```
data(relgoods)
```

Format

The description of subjects' covariates is the following:

`ID` An identification number

`Gender` A factor with levels: 0 = man, 1 = woman

`BirthMonth` A variable indicating the month of birth of the respondent

`BirthYear` A variable indicating the year of birth of the respondent

`Family` A factor variable indicating the number of members of the family

`Year.12` A factor with levels: 1 = if there is any child aged less than 12 in the family, 0 = otherwise

`EducationDegree` A factor with levels: 1 = compulsory school, 2 = high school diploma, 3 = Graduated-Bachelor degree, 4 = Graduated-Master degree, 5 = Post graduated

`MaritalStatus` A factor with levels: 1 = Unmarried, 2 = Married/Cohabitee, 3 = Separated/Divorced, 4 = Widower

`Residence` A factor with levels: 1 = City of Naples, 2 = District of Naples, 3 = Others Campania, 4 = Others Italia, 5 = Foreign countries

`Glasses` A factor with levels: 1 = wearing glasses or contact lenses, 0 = otherwise

`RightHand` A factor with levels: 1 = right-handed, 0 = left-handed

`Smoking` A factor with levels: 1 = smoker, 0 = not smoker

`WalkAlone` A factor with levels: 1 = usually walking alone, 0 = usually walking in company

`job` A factor with levels: 1 = Not working, 2 = Retired, 3 = occasionally, 4 = fixed-term job, 5 = permanent job

`PlaySport` A factor with levels: 1 = Not playing any sport, 2 = Yes, individual sport, 3 = Yes, team sport

`Pets` A factor with levels: 1 = owning a pet, 0 = not owning any pet

1) Respondents were asked to evaluate the following items on a 10 point Likert scale, ranging from 1 = "never, at all" to 10 = "always, a lot":

`WalkOut` How often the respondent goes out for a walk

`Parents` How often respondent talks at least to one of his/her parents

`MeetRelatives` How often respondent meets his/her relatives

`Association` Frequency of involvement in volunteering or different kinds of associations/parties, etc

`RelFriends` Quality of respondent's relationships with friends

`RelNeighbours` Quality of the relationships with neighbors

`NeedHelp` Easiness in asking help whenever in need

`Environment` Level of comfort with the surrounding environment

`Safety` Level of safety in the streets

`EndofMonth` Family making ends meet

`MeetFriend` Number of times the respondent met his/her friends during the month preceding the interview

`Physician` Importance of the kindness/simpathy in the selection of respondent's physician

`Happiness` Each respondent was asked to mark a sign on a 110mm horizontal line according to his/her feeling of happiness (left endpoint corresponding to completely unhappy, right-most endpoint corresponding to extremely happy

2) The same respondents were asked to score the activities for leisure time listed below, according to their involvement/degree of amusement, on a 10 point Likert scale ranging from 1 = "At all, nothing, never" to 10 = "Totally, extremely important, always":

`Videogames`

`Reading`

`Cinema`

`Drawing`

`Shopping`

`Writing`

`Bicycle`

`Tv`

`StayWFriend` Spending time with friends

`Groups` Taking part to associations, meetings, etc.

`Walking`

`HandWork` Hobby, gardening, sewing, etc.

`Internet`

`Sport`

`SocialNetwork`

`Gym`

`Quiz` Crosswords, sudoku, etc.

`MusicInstr` Playing a musical instrument

`GoAroundCar` Hanging out by car

`Dog` Walking out the dog

`GoOutEat` Go to restaurants/pubs

Details

Period of data collection: December 2014
Mode of collection: questionnaire
Number of observations: 2459
Number of subjects' covariates: 16
Number of analyzed items: 34
Warning: with a limited number of missing values

Source

http://www.labstat.it/home/wp-content/uploads/2015/09/relgoods.txt `summary.fastCUB` *S3 method: summary for class "fastCUB"*

Description

S3 method summary for objects of class `fastCUB`.

Usage

```
## S3 method for class 'fastCUB'
summary(object, correlation = FALSE,
  pnames = object$parnames, ...)
```

Arguments

`object`	An object of class `fastCUB`
`correlation`	Logical: should the estimated correlation matrix be returned? Default is FALSE
`pnames`	A vector of character for names to be assigned to estimated parameters
`...`	Other arguments

Value

Extended summary results of the fitting procedure, including parameter estimates, their standard errors and Wald statistics, maximized log-likelihood compared with that of the saturated model and of a Uniform sample. AIC, BIC and ICOMP indeces are also displayed for model selection. Execution time and number of exectued iterations for the fitting procedure are aslo returned.

`univer` *Evaluation of the Orientation Services 2002*

Description

A sample survey on students evaluation of the Orientation services was conducted across the 13 Faculties of University of Naples Federico II in five waves: participants were asked to express their ratings on a 7 point scale (1 = "very unsatisfied", 7 = "extremely satisfied"). Here dataset collected during 2002 is loaded.

Usage

```
data(univer)
```

Format

The description of subjects' covariates is:

- `Faculty` A factor variable, with levels ranging from 1 to 13 indicating the coding for the different university faculties
- `Freqserv` A factor with levels: 0 = for not regular users, 1 = for regular users
- `Age` Variable indicating the age of the respondent in years
- `Gender` A factor with levels: 0 = man, 1 = woman
- `Diploma` A factor with levels: 1 = classic studies, 2 = scientific studies, 3 = linguistic, 4 = Professional, 5 = Technical/Accountancy, 6 = others
- `Residence` A factor with levels: 1 = city NA, 2 = district NA, 3 = others
- `ChangeFa` A factor with levels: 1 = changed faculty, 2 = not changed faculty

Analyzed ordinal variables (Likert ordinal scale): 1 = "extremely unsatisfied", 2 = "very unsatisfied", 3 = "unsatisfied", 4 = "indifferent", 5 = "satisfied", 6 = "very satisfied", 7 = "extremely satisfied"

- `Informat` Level of satisfaction about the collected information
- `Willingn` Level of satisfaction about the willingness of the staff
- `Officeho` Judgment about the Office hours
- `Competen` Judgement about the competence of the staff
- `Global` Global satisfaction

Details

Period of data collection: 2002
Mode of collection: questionnaire
Number of observations: 2179
Number of subjects' covariates: 7
Number of analyzed items: 5

Source

`http://www.labstat.it/home/wp-content/uploads/2016/12/univer2002.txt`

`vcov.fastCUB` *S3 method vcov() for class "fastCUB"*

Description

S3 method: vcov for objects of class `fastCUB`.

Usage

```
## S3 method for class 'fastCUB'
vcov(object, ...)
```

Arguments

object	An object of class fastCUB
...	Other arguments

Value

Variance-covariance matrix of the final ML estimates for parameters of the fitted CUB model, according to the logit transform also when covariates are not included for uncertainty or feeling parameters. It is computed on the basis of Louis' identity within the EM algorithm.

See Also

fastCUB

Index