Package 'LMest'

March 18, 2019

Type Package

Title Latent Markov Models with and without Covariates

Version 2.4.4

Date 2019-03-14

Author Francesco Bartolucci, Silvia Pandolfi - University of Perugia (IT)

Maintainer Francesco Bartolucci <bart@stat.unipg.it>

Description Fit certain versions of the Latent Markov model for longitudinal categorical data.

License GPL (>= 2)

Depends R (>= 2.0.0), MASS, MultiLCIRT, stats, mvtnorm, mmm, mclust (>= 5.0)

NeedsCompilation yes

Repository CRAN

Date/Publication 2019-03-18 13:03:23 UTC

R topics documented:

`LMest-package` *Fit latent Markov models*

Description

Set of functions to fit latent Markov models in the basic version and in the extended version with individual covariates.

Details

Package:	LMest
Type:	Package
Version:	2.4.4
Date:	2019-03-14
License:	Unlimited

The most important functions are `est_lm_basic`, `est_lm_cov_manifest`, and `est_lm_cov_latent` which estimate the basic LM model and its extensions including covariates

Author(s)

Francesco Bartolucci, Silvia Pandolfi - University of Perugia (IT) Maintainer: Francesco Bartolucci <bart@stat.unipg.it>

References

Bartolucci, F., Farcomeni, A. and Pennoni, F. (2013), *Latent Markov Models for Longitudinal Data*, Chapman and Hall/CRC press.

Bartolucci, F., Pandolfi, S. and Pennoni, F. (2017), LMest: An R Package for Latent Markov Models for Longitudinal Categorical Data, *Journal of Statistical Software*, **81**, 1-38, doi:10.18637/jss.v081.i04.

Examples

```
# Example of drug consumption data
# load data
data(data_drug)
```

```
data_drug = as.matrix(data_drug)
S = data_drug[,1:5]-1
yv = data_drug[,6]
n = sum(yv)
# fit of the Basic LM model
k = 3
out1 = est_lm_basic(S,yv,k,mod=1,out_se=TRUE)
## Not run:
out2 = bootstrap_lm_basic(out1$piv,out1$Pi,out1$Psi,n,mod=1,B=1000)

## End(Not run)
```

`blkdiag` *Build a block diagonal matrix.*

Description

Function to build a block diagonal matrix (internal function).

Usage

```
blkdiag(A, B)
```

Arguments

A	The first non-empty, numeric matrix
B	The second non-empty, numeric matrix

Value

C	A numeric block diagonal matrix

Author(s)

Francesco Bartolucci, Silvia Pandolfi - University of Perugia (IT)

`bootstrap_lm_basic` *Parametric bootstrap for the basic LM model*

Description

Function that performs bootstrap parametric resampling to compute standard errors for the parameter estimates.

Usage

```
bootstrap_lm_basic(piv, Pi, Psi, n, B = 100, start = 0, mod = 0, tol = 10^-6)
```

Arguments

piv	initial probability vector
Pi	probability transition matrices (k x k x TT)
Psi	matrix of conditional response probabilities (mb x k x r)
n	sample size
B	number of bootstrap samples
start	type of starting values (0 = deterministic, 1 = random)
mod	model on the transition probabilities (0 for time-heter., 1 for time-homog., from 2 to (TT-1) partial homog. of that order)
tol	tolerance level for convergence

Value

mPsi	average of bootstrap estimates of the conditional response probabilities
mpiv	average of bootstrap estimates of the initial probability vector
mPi	average of bootstrap estimates of the transition probability matrices
sePsi	standard errors for the conditional response probabilities
sepiv	standard errors for the initial probability vector
sePi	standard errors for the transition probability matrices

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

Examples

```
## Not run:
# Example of drug consumption data
# load data
data(data_drug)
data_drug = as.matrix(data_drug)
S = data_drug[,1:5]-1
yv = data_drug[,6]
n = sum(yv)
# fit of the Basic LM model
k = 3
out1 = est_lm_basic(S,yv,k,mod=1,out_se=TRUE)
out2 = bootstrap_lm_basic(out1$piv,out1$Pi,out1$Psi,n,mod=1,B=1000)

## End(Not run)
```

`bootstrap_lm_basic_cont` *Parametric bootstrap for the basic LM model for continuous outcomes*

Description

Function that performs bootstrap parametric resampling to compute standard errors for the parameter estimates.

Usage

```
bootstrap_lm_basic_cont(piv, Pi, Mu, Si, n, B = 100, start = 0, mod = 0, tol = 10^-6)
```

Arguments

`piv`	initial probability vector
`Pi`	probability transition matrices (k x k x TT)
`Mu`	matrix of conditional means for the response variables (r x k)
`Si`	var-cov matrix common to all states (r x r)
`n`	sample size
`B`	number of bootstrap samples
`start`	type of starting values (0 = deterministic, 1 = random)
`mod`	model on the transition probabilities (0 for time-heter., 1 for time-homog., from 2 to (TT-1) partial homog. of that order)
`tol`	tolerance level for convergence

Value

`mMu`	average of bootstrap estimates of the conditional means of the response variables
`mSi`	average of bootstrap estimates of the var-cov matrix
`mpiv`	average of bootstrap estimates of the initial probability vector
`mPi`	average of bootstrap estimates of the transition probability matrices
`seMu`	standard errors for the conditional means of the response variables
`seSi`	standard errors for the var-cov matrix
`sepiv`	standard errors for the initial probability vector
`sePi`	standard errors for the transition probability matrices

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

Examples

```
## Not run:
# Example based on multivariate longitudinal continuous data
# load data
require(mmm)
data(multiLongGaussian)
res = long2matrices(multiLongGaussian$ID,X=cbind(multiLongGaussian$X,multiLongGaussian$time),
      Y=cbind(multiLongGaussian$resp1, multiLongGaussian$resp2))
Y = res$YY
n = dim(Y)[1]
# fit of the Basic LM model for continuous outcomes
k = 3
out1 = est_lm_basic_cont(Y,k,mod=1)
out2 = bootstrap_lm_basic_cont(out1$piv,out1$Pi,out1$Mu,out1$Si,n,mod=1,B = 1000)

## End(Not run)
```

---

bootstrap_lm_cov_latent

Parametric bootstrap for LM models with individual covariates in the latent model

---

Description

Function that performs bootstrap parametric resampling to compute standard errors for the parameter estimates.

Usage

```
bootstrap_lm_cov_latent(X1, X2, param = "multilogit", Psi, Be, Ga, B = 100,
   fort = TRUE)
```

Arguments

X1	matrix of covariates affecting the initial probabilities (n x nc1)
X2	array of covariates affecting the transition probabilities (n x TT-1 x nc2)
param	type of parametrization for the transition probabilities ("multilogit" = standard multinomial logit for every row of the transition matrix, "difflogit" = multinomial logit based on the difference between two sets of parameters)
Psi	array of conditional response probabilities (mb x k x r)
Be	parameters affecting the logit for the initial probabilities
Ga	parametes affecting the logit for the transition probabilities
B	number of bootstrap samples
fort	to use fortran routine when possible (FALSE for not use fortran)

Value

mPsi	average of bootstrap estimates of the conditional response probabilities
mBe	average of bootstrap estimates of the parameters affecting the logit for the initial probabilities
mGa	average of bootstrap estimates of the parameters affecting the logit for the transition probabilities
sePsi	standard errors for the conditional response probabilities
seBe	standard errors for the parameters in Be
seGa	standard errors for the parameters in Ga

Author(s)

Francesco Bartolucci, Silvia Pandolfi - University of Perugia (IT)

Examples

```
## Not run:
# Example based on self-rated health status (SRHS) data
# load SRHS data
data(data_SRHS_long)
dataSRHS = data_SRHS_long

TT = 8
head(dataSRHS)
res = long2matrices(dataSRHS$id,X=cbind(dataSRHS$gender-1,
dataSRHS$race==2|dataSRHS$race==3, dataSRHS$education==4,
dataSRHS$education==5,dataSRHS$age-50,(dataSRHS$age-50)^2/100),
Y=dataSRHS$srhs)

# matrix of responses (with ordered categories from 0 to 4)
S = 5-res$YY

# matrix of covariates (for the first and the following occasions)
# colums are: gender,race,educational level (2 columns),age,age^2)
X1 =res$XX[,1,]
X2 =res$XX[,2:TT,]

# estimate the model
out1 = est_lm_cov_latent(S,X1,X2,k=2,output=TRUE,out_se=TRUE)

out2 = bootstrap_lm_cov_latent(X1,X2,Psi=out1$Psi,Be=out1$Be,Ga=out1$Ga,B=1000)

## End(Not run)
```

`bootstrap_lm_cov_latent_cont` *Parametric bootstrap for LM models for continuous outcomes with individual covariates in the latent model*

Description

Function that performs bootstrap parametric resampling to compute standard errors for the parameter estimates.

Usage

```
bootstrap_lm_cov_latent_cont(X1, X2, param = "multilogit", Mu, Si, Be, Ga, B = 100)
```

Arguments

`X1`	matrix of covariates affecting the initial probabilities (n x nc1)
`X2`	array of covariates affecting the transition probabilities (n x TT-1 x nc2)
`param`	type of parametrization for the transition probabilities ("multilogit" = standard multinomial logit for every row of the transition matrix, "difflogit" = multinomial logit based on the difference between two sets of parameters)
`Mu`	matrix of conditional means for the response variables (r x k)
`Si`	var-cov matrix common to all states (r x r)
`Be`	parameters affecting the logit for the initial probabilities
`Ga`	parametes affecting the logit for the transition probabilities
`B`	number of bootstrap samples

Value

`mMu`	average of bootstrap estimates of the conditional means for the response variables
`mSi`	average of bootstrap estimates of the var-cov matrix
`mBe`	average of bootstrap estimates of the parameters affecting the logit for the initial probabilities
`mGa`	average of bootstrap estimates of the parameters affecting the logit for the transition probabilities
`seMu`	standard errors for the conditional means
`seSi`	standard errors for the var-cov matrix
`seBe`	standard errors for the parameters in Be
`seGa`	standard errors for the parameters in Ga

Author(s)

Francesco Bartolucci, Silvia Pandolfi - University of Perugia (IT)

Examples

```
## Not run:
# Example based on multivariate longitudinal continuous data
# load data
require(mmm)
data(multiLongGaussian)
TT=4
res = long2matrices(multiLongGaussian$ID,X=cbind(multiLongGaussian$X,multiLongGaussian$time),
                    Y=cbind(multiLongGaussian$resp1, multiLongGaussian$resp2))
Y = res$YY
X1 =res$XX[,1,]
X2 =res$XX[,2:TT,]

# estimate the model
est = est_lm_cov_latent_cont(Y,X1,X2,k=3,output=TRUE)
out = bootstrap_lm_cov_latent_cont(X1,X2,Mu=est$Mu,Si=est$Si,Be=est$Be,Ga=est$Ga,B=1000)


## End(Not run)
```

`complk` *Complete log-likelihood of the basic latent Markov model*

Description

Function that computes complete log-likelihood of the basic latent Markov model (internal use).

Usage

```
complk(S, R, yv, piv, Pi, Psi, k)
```

Arguments

`S`	matrix of distinct response configurations (n x TT x r)
`R`	matrix of missing response configurations
`yv`	corresponding vector of frequencies
`piv`	vector of initial probabilities
`Pi`	transition probability matrix
`Psi`	conditional response probabilities
`k`	number of latent classes

Value

`lk`	log-likelihood
`Phi`	matrix of the conditional probabilities of the observed response configurations
`L`	matrix of the forward probabilities
`pv`	vector of marginal probabilities

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

References

Baum, L. E., Petrie, T., Soules, G., and Weiss, N. (1970). A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. Annals of Mathematical Statistics, 41, 164-171.

---

complk_cont *Complete log-likelihood of the basic latent Markov model for continuous outcomes*

Description

Function that computes complete log-likelihood of the basic latent Markov model for continuous outcomes (internal use).

Usage

```
complk_cont(Y, piv, Pi, Mu, Si, k)
```

Arguments

Y	array of continuous outcomes (n x TT x r)
piv	vector of initial probabilities
Pi	transition probability matrix
Mu	matrix of conditional means for the response variables (r x k)
Si	var-cov matrix common to all states (r x r)
k	number of latent classes

Value

lk	log-likelihood
Phi	matrix of the conditional probabilities of the observed response configurations
L	matrix of the forward probabilities
pv	vector of marginal probabilities

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

References

Baum, L. E., Petrie, T., Soules, G., and Weiss, N. (1970). A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. Annals of Mathematical Statistics, 41, 164-171.

`data_criminal_sim` *Criminal dataset*

Description

Simulated dataset about crimes committed by a cohort of subjects.

Usage

```
data(data_criminal_sim)
```

Format

A data frame with 60000 observations on the following 13 variables.

`id` subject id

`sex` gender of the subject

`time` occasion of observation

`y1` crime of type 1

`y2` crime of type 2

`y3` crime of type 3

`y4` crime of type 4

`y5` crime of type 5

`y6` crime of type 6

`y7` crime of type 7

`y8` crime of type 8

`y9` crime of type 9

`y10` crime of type 10

References

Bartolucci, F., Pennoni, F. and Francis, B. (2007), A latent Markov model for detecting patterns of criminal activity, Journal of the Royal Statistical Society, series A, 170, pp. 115-132.

Examples

```
data(data_criminal_sim)
```

data_drug *Dataset about marijuana consumption*

Description

Longitudinal dataset about marijuana consumption measured by ordinal variables with 3 categories with increasing levels of consumption.

Usage

```
data(data_drug)
```

Format

A data frame with 51 observations on the following 6 variables.

`V1` reported drug use at the 1st occasion

`V2` reported drug use at the 2nd occasion

`V3` reported drug use at the 3rd occasion

`V4` reported drug use at the 4th occasion

`V5` reported drug use at the 5th occasion

`V6` frequency of the response configuration

Source

Elliot, D. S., Huizinga, D. and Menard, S. (1989) Multiple Problem Youth: Delinquency, Substance Use, and Mental Health Problems. New York: Springer.

References

Bartolucci, F. (2006). Likelihood inference for a class of latent markov models under linear hypotheses on the transition probabilities. Journal of the Royal Statistical Society, series B, 68, 155-178.

Examples

```
data(data_drug)
```

`data_SRHS_long` *Self-reported health status dataset*

Description

Dataset about self-reported health status derived from the Health and Retirement Study conducted by the University of Michigan.

Usage

```
data(data_SRHS_long)
```

Format

A data frame with 56592 observations on the following 6 variables.

`id` subject id

`gender` gender of the subject coded as 1 for "male", 2 for "female"

`race` race coded as 1 for "white", 2 for "black", 3 for "others"

`education` educational level coded as 1 for "high school", 2 for "general educational diploma", 3 for "high school graduate", 4 for "some college", 5 for "college and above"

`age` age at the different time occasions

`srhs` self-reported health status at the different time occasions coded as 1 for "excellent", 2 for "very good", 3 for "good", 4 for "fair", 5 for "poor"

References

Bartolucci, F., Bacci, S. and Pennoni, F. (2014), Longitudinal analysis of the self-reported health status by mixture latent autoregressive models, Journal of the Royal Statistical Society - series C, 63, pp. 267-288

Examples

```
data(data_SRHS_long)
```

decoding *Perform local and global decoding*

Description

Function that performs local and global decoding (Viterbi) from the output of `est_lm_basic`, `est_lm_cov_latent`, `est_lm_cov_manifest`, and `est_lm_mixed`.

Usage

```
decoding(est, Y, X1 = NULL, X2 = NULL, fort = TRUE)
```

Arguments

`est`	output from `est_lm_basic`, `est_lm_cov_latent`, `est_lm_cov_manifest`, or `est_lm_mixed`
`Y`	single vector or matrix of responses
`X1`	matrix of covariates on the initial probabilities (`est_lm_cov_latent`) or on the responses (`est_lm_cov_manifest`)
`X2`	array of covariates on the transition probabilites
`fort`	to use Fortran routines

Value

`Ul`	matrix of local decoded states corresponding to each row of Y
`Ug`	matrix of global decoded states corresponding to each row of Y

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

References

Viterbi A (1967). Error Bounds for Convolutional Codes and an Asymptotically Optimum Decoding Algorithm. IEEE Transactions on Information Theory, 13, 260-269.

Juan B., Rabiner L. (1991). Hidden Markov Models for Speech Recognition. Technometrics, 33, 251-272.

Examples

```
# example for the output from est_lm_basic
data(data_drug)
data_drug = as.matrix(data_drug)
S = data_drug[,1:5]-1
yv = data_drug[,6]
n = sum(yv)
```

```
# fit the Basic LM model
k = 3
est = est_lm_basic(S,yv,k,mod=1)
# decoding for a single sequence
out1 = decoding(est,S[1,])
# decoding for all sequences
out2 = decoding(est,S)


# example for the output from est_lm_cov_latent with difflogit parametrization
data(data_SRHS_long)
dataSRHS = data_SRHS_long[1:1600,]

TT = 8
head(dataSRHS)
res = long2matrices(dataSRHS$id,X=cbind(dataSRHS$gender-1,
dataSRHS$race==2| dataSRHS$race==3, dataSRHS$education==4,
dataSRHS$education==5, dataSRHS$age-50,(dataSRHS$age-50)^2/100),
Y= dataSRHS$srhs)

# matrix of responses (with ordered categories from 0 to 4)
S = 5-res$YY

# matrix of covariates (for the first and the following occasions)
# colums are: gender,race,educational level (2 columns),age,age^2)
X1 =res$XX[,1,]
X2 =res$XX[,2:TT,]

# estimate the model
est = est_lm_cov_latent(S,X1,X2,k=2,output=TRUE,param="difflogit")
# decoding for a single sequence
out1 = decoding(est,S[1,,],X1[1,],X2[1,,])
# decoding for all sequences
out2 = decoding(est,S,X1,X2)
```

`draw_lm_basic` *Draw samples from the basic LM model*

Description

Function that draws samples from the basic LM model with specific parameters.

Usage

```
draw_lm_basic(piv, Pi, Psi, n)
```

Arguments

`piv`	vector of initial probabilities of the latent Markov chain
`Pi`	set of transition probabilities matrices (k x k x TT)

Psi	array of conditional response probabitlies (mb x k x r)
n	sample size

Value

Y	matrix of response configurations unit by unit
S	matrix of distinct response configurations
yv	corresponding vector of frequencies

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

Examples

```
# draw a sample for 1000 units and only one response variable
n = 1000
TT = 6
k = 2
r = 1 #number of response variables
mb = 3 #maximum number of response categories

piv = c(0.7,0.3)
Pi = matrix(c(0.9,0.1,0.1,0.9),k,k)
Pi = array(Pi,c(k,k,TT))
Pi[,,1] = 0
Psi = matrix(c(0.7,0.2,0.1,0.5,0.4,0.1),mb,k)
Psi = array(Psi,c(mb,k,r))
out = draw_lm_basic(piv, Pi, Psi, n=1000)
```

---

draw_lm_basic_cont *Draw samples from the basic LM model for continuous outcomes*

Description

Function that draws samples from the basic LM model for continuous outcomes with specific parameters.

Usage

```
draw_lm_basic_cont(piv, Pi, Mu, Si, n)
```

Arguments

piv	vector of initial probabilities of the latent Markov chain
Pi	set of transition probabilities matrices (k x k x TT)
Mu	matrix of conditional means for the response variables (r x k)
Si	var-cov matrix common to all states (r x r)
n	sample size

Value

Y	array of continuous outcomes (n x TT x r)

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

Examples

```
# draw a sample for 1000 units and 3 response variable
n = 1000
TT = 5
k = 2
r = 3 #number of response variables

piv = c(0.7,0.3)
Pi = matrix(c(0.9,0.1,0.1,0.9),k,k)
Pi = array(Pi,c(k,k,TT))
Pi[,,1] = 0
Mu  = matrix(c(-2,-2,0,0,2,2),r,k)
Si = diag(r)
out = draw_lm_basic_cont(piv,Pi,Mu,Si,n)
```

---

draw_lm_cov_latent *Draw samples from LM model with covariaates in the latent model*

Description

Function that draws samples from the LM model with individual covariates with specific parameters.

Usage

```
draw_lm_cov_latent(X1, X2, param = "multilogit", Psi, Be, Ga, fort = TRUE)
```

Arguments

X1	desing matrix for the covariates on the initial probabilities (n x nc1)
X2	desing matrix for the covariates on the transition probabilities (n x TT-1 x nc2)
param	type of parametrization for the transition probabilities ("multilogit" = standard multinomial logit for every row of the transition matrix, "difflogit" = multinomial logit based on the difference between two sets of parameters)
Psi	array of conditional response probabilities (mb x k x r)
Be	parameters affecting the logit for the initial probabilities
Ga	parametes affecting the logit for the transition probabilities
fort	to use fortran routine when possible (FALSE for not use fortran)

Value

Y	matrix of response configurations unit by unit (n x TT x r)
U	matrix containing the sequence of latent states (n x TT)

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

Examples

```
# draw a sample for 1000 units, 10 response variable and 2 covariates
n=1000
TT = 5
k=2
nc = 2 #number of covariates
r = 10 #number of response variables
mb = 2 #maximum number of response categories
fort=TRUE


Psi  = matrix(c(0.9,0.1,0.1,0.9),mb,k)
Psi = array(Psi,c(mb,k,r))
Ga = matrix(c(-log(0.9/0.1),0.5,1),(nc+1)*(k-1),k)
Be = array(c(0,0.5,1),(nc+1)*(k-1))
#Simulate covariates
X1 = matrix(0,n,nc)
for(j in 1:nc) X1[,j]=rnorm(n)
X2 = array(0,c(n,TT-1,nc))
for (t in 1:(TT-1)) for(j in 1:nc){
if(t==1){
X2[,t,j] = 0.5*X1[,j]+rnorm(n)
}else{
X2[,t,j] = 0.5*X2[,t-1,j]+rnorm(n)
}
}

out = draw_lm_cov_latent(X1,X2,Psi=Psi,Be=Be,Ga=Ga,fort=fort)
```

---

draw_lm_cov_latent_cont

Draw samples from LM model for continuous outcomes with covariaates in the latent model

---

Description

Function that draws samples from the LM model for continuous outcomes with individual covariates with specific parameters.

Usage

```
draw_lm_cov_latent_cont(X1, X2, param = "multilogit", Mu, Si, Be, Ga, fort = TRUE)
```

Arguments

X1	desing matrix for the covariates on the initial probabilities (n x nc1)
X2	desing matrix for the covariates on the transition probabilities (n x TT-1 x nc2)
param	type of parametrization for the transition probabilities ("multilogit" = standard multinomial logit for every row of the transition matrix, "difflogit" = multinomial logit based on the difference between two sets of parameters)
Mu	array of conditional means for the response variables (r x k)
Si	var-cov matrix common to all states (r x r)
Be	parameters affecting the logit for the initial probabilities
Ga	parametes affecting the logit for the transition probabilities
fort	to use fortran routine when possible (FALSE for not use fortran)

Value

Y	array of continuous outcomes (n x TT x r)
U	matrix containing the sequence of latent states (n x TT)

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

Examples

```
# draw a sample for 1000 units, 10 response variable and 2 covariates
n=1000
TT = 5
k=2
nc = 2 #number of covariates
r = 3 #number of response variables
fort=TRUE

Mu  = matrix(c(-2,-2,0,0,2,2),r,k)
Si = diag(r)
Ga = matrix(c(-log(0.9/0.1),0.5,1),(nc+1)*(k-1),k)
Be = array(c(0,0.5,1),(nc+1)*(k-1))

#Simulate covariates
X1 = matrix(0,n,nc)
for(j in 1:nc) X1[,j]=rnorm(n)
X2 = array(0,c(n,TT-1,nc))
for (t in 1:(TT-1)) for(j in 1:nc){
if(t==1){
X2[,t,j] = 0.5*X1[,j]+rnorm(n)
}else{
```

```
X2[,t,j] = 0.5*X2[,t-1,j]+rnorm(n)
}
}

out = draw_lm_cov_latent_cont(X1,X2,param="multilogit",Mu,Si,Be,Ga,fort=fort)
```

`draw_lm_mixed` *Draws samples from the mixed LM model*

Description

Function that draws samples from the mixed LM model with specific parameters.

Usage

```
draw_lm_mixed(la, Piv, Pi, Psi, n, TT)
```

Arguments

`la`	vector of mass probabilities for the first latent variable
`Piv`	matrix of initial probabilities of the latent Markov chain (k2 x k1)
`Pi`	set of transition matrices (k2 x k2 x k1)
`Psi`	array of conditional response probabitlies (mb x k2 x r)
`n`	sample size
`TT`	number of time occasions

Value

`Y`	matrix of response configurations unit by unit
`S`	matrix of distinct response configurations
`yv`	corresponding vector of frequencies

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

Examples

```
# draw a sample for 1000 units and only one response variable and 5 time occasions
k1 = 2; k2 = 3
la = rep(1/k1,k1)
Piv = matrix(1/k2,k2,k1)
Pi = array(0,c(k2,k2,k1))
Pi[,,1] = diag(k2)
Pi[,,2] = 1/k2
Psi = cbind(c(0.6,0.3,0.1),c(0.1,0.3,0.6),c(0.3,0.6,0.1))
out = draw_lm_mixed(la,Piv,Pi,Psi,n=1000,TT=5)
```

`est_lm_basic` *Estimate basic LM model*

Description

Main function for estimating the basic LM model.

Usage

```
est_lm_basic(S, yv, k, start = 0, mod = 0, tol = 10^-8, maxit = 1000,
   out_se=FALSE, piv=NULL, Pi=NULL, Psi = NULL)
```

Arguments

`S`	array of available configurations (n x TT x r) with categories starting from 0 (use NA for missing responses)
`yv`	vector of frequencies of the available configurations
`k`	number of latent states
`start`	type of starting values (0 = deterministic, 1 = random, 2 = initial values in input)
`mod`	model on the transition probabilities (0 for time-heter., 1 for time-homog., from 2 to (TT-1) partial homog. of that order)
`tol`	tolerance level for convergence
`maxit`	maximum number of iterations of the algorithm
`out_se`	to compute the information matrix and standard errors
`piv`	initial value of the initial probability vector (if start=2)
`Pi`	initial value of the transition probability matrices (k x k x TT) (if start=2)
`Psi`	initial value of the conditional response probabilities (mb x k x r) (if start=2)

Value

`lk`	maximum log-likelihood
`piv`	estimate of initial probability vector
`Pi`	estimate of transition probability matrices
`Psi`	estimate of conditional response probabilities
`np`	number of free parameters
`aic`	value of AIC for model selection
`bic`	value of BIC for model selection
`lkv`	log-likelihood trace at every step
`V`	array containing the posterior distribution of the latent states for each response configuration and time occasion
`sepiv`	standard errors for the initial probabilities
`sePi`	standard errors for the transition probabilities
`sePsi`	standard errors for the conditional response probabilities
`call`	command used to call the function

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

References

Bartolucci, F., Farcomeni, A. and Pennoni, F. (2013), Latent Markov Models for Longitudinal Data, Chapman and Hall/CRC press.

Examples

```
# Example of drug consumption data
# load data
data(data_drug)
data_drug = as.matrix(data_drug)
S = data_drug[,1:5]-1
yv = data_drug[,6]
# fit of the Basic LM model
k = 3
out = est_lm_basic(S,yv,k,mod=1)
summary(out)

## Not run:
# Example based on criminal data
# load criminal data
data(data_criminal_sim)
out = long2wide(data_criminal_sim,"id","time","sex",
c("y1","y2","y3","y4","y5","y6","y7","y8","y9","y10"),aggr=T,full=999)
XX = out$XX
YY = out$YY
freq = out$freq
# fit basic LM model with increasing number of states to select the most suitable
Res0 = vector("list",7)
for(k in 1:7){
    Res0[[k]] = est_lm_basic(YY,freq,k,mod=1,tol=10^-4)
    save(list = ls(),file="example_criminal_temp.RData")
}
out1 = Res0[[6]]

## End(Not run)
```

---

est_lm_basic_cont *Estimate basic LM model for continuous outcomes*

Description

Main function for estimating the basic LM model for continuous outcomes.

Usage

```
est_lm_basic_cont(Y, k, start = 0, mod = 0, tol = 10^-8, maxit = 1000,
                  out_se=FALSE, piv = NULL, Pi = NULL, Mu = NULL, Si = NULL)
```

Arguments

Y	array of continuous outcomes (n x TT x r)
k	number of latent states
start	type of starting values (0 = deterministic, 1 = random, 2 = initial values in input)
mod	model on the transition probabilities (0 for time-heter., 1 for time-homog., from 2 to (TT-1) partial homog. of that order)
tol	tolerance level for convergence
maxit	maximum number of iterations of the algorithm
out_se	to compute the information matrix and standard errors
piv	initial value of the initial probability vector (if start=2)
Pi	initial value of the transition probability matrices (k x k x TT) (if start=2)
Mu	initial value of the conditional means (r x k) (if start=2)
Si	initial value of the var-cov matrix common to all states (r x r) (if start=2)

Value

lk	maximum log-likelihood
piv	estimate of initial probability vector
Pi	estimate of transition probability matrices
Mu	estimate of conditional means of the response variables
Si	estimate of var-cov matrix common to all states
np	number of free parameters
aic	value of AIC for model selection
bic	value of BIC for model selection
lkv	log-likelihood trace at every step
V	array containing the posterior distribution of the latent states for each units and time occasion
call	command used to call the function

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

References

Bartolucci, F., Farcomeni, A. and Pennoni, F. (2013), Latent Markov Models for Longitudinal Data, Chapman and Hall/CRC press.

Examples

```
# Example based on multivariate longitudinal continuous data
# load data
require(mmm)
data(multiLongGaussian)
res = long2matrices(multiLongGaussian$ID,X=cbind(multiLongGaussian$X,multiLongGaussian$time),
      Y=cbind(multiLongGaussian$resp1, multiLongGaussian$resp2))
Y = res$YY

# fit of the Basic LM model for continuous outcomes
k = 3
out = est_lm_basic_cont(Y,k,mod=1,tol=10^-5)
summary(out)
```

`est_lm_cov_latent` *Estimate LM model with covariates in the latent model*

Description

Main function for estimating the LM model with covariates in the latent model.

Usage

```
est_lm_cov_latent(S, X1=NULL, X2=NULL, yv = rep(1,nrow(S)), k, start = 0, tol = 10^-8,
   maxit = 1000, param = "multilogit", Psi, Be, Ga, fort = TRUE, output = FALSE,
   out_se = FALSE, fixPsi = FALSE)
```

Arguments

`S`	array of available configurations (n x TT x r) with categories starting from 0 (use NA for missing responses)
`X1`	matrix of covariates affecting the initial probabilities (n x nc1)
`X2`	array of covariates affecting the transition probabilities (n x TT-1 x nc2)
`yv`	vector of frequencies of the available configurations
`k`	number of latent states
`start`	type of starting values (0 = deterministic, 1 = random, 2 = initial values in input)
`tol`	tolerance level for checking convergence of the algorithm
`maxit`	maximum number of iterations of the algorithm
`param`	type of parametrization for the transition probabilities ("multilogit" = standard multinomial logit for every row of the transition matrix, "difflogit" = multinomial logit based on the difference between two sets of parameters)
`Psi`	intial value of the array of the conditional response probabilities (mb x k x r)
`Be`	intial value of the parameters affecting the logit for the initial probabilities (if start=2)

Ga	intial value of the parametes affecting the logit for the transition probabilities (if start=2)
fort	to use fortran routine when possible (FALSE for not use fortran)
output	to return additional output (V,PI,Piv,Ul)
out_se	to compute the information matrix and standard errors
fixPsi	TRUE if Psi is given in input and is not updated anymore

Value

lk	maximum log-likelihood
Be	estimated array of the parameters affecting the logit for the initial probabilities
Ga	estimated array of the parameters affecting the logit for the transition probabilities
Piv	estimate of initial probability matrix
PI	estimate of transition probability matrices
Psi	estimate of conditional response probabilities
np	number of free parameters
aic	value of AIC for model selection
bic	value of BIC for model selection
lkv	log-likelihood trace at every step
V	array containing the posterior distribution of the latent states for each response configuration and time occasion
Ul	matrix containing the predicted sequence of latent states by the local decoding method
sePsi	standard errors for the conditional response matrix
seBe	standard errors for Be
seGa	standard errors for Ga
call	command used to call the function

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia, http://www.stat.unipg.it/bartolucci

References

Bartolucci, F., Farcomeni, A. and Pennoni, F. (2013), Latent Markov Models for Longitudinal Data, Chapman and Hall/CRC press.

Examples

```
## Not run:
# Example based on self-rated health status (SRHS) data
# load SRHS data
data(data_SRHS_long)
dataSRHS = data_SRHS_long

TT = 8
head(dataSRHS)
res = long2matrices(dataSRHS$id,X=cbind(dataSRHS$gender-1,
dataSRHS$race==2|dataSRHS$race==3, dataSRHS$education==4,
dataSRHS$education==5,dataSRHS$age-50,(dataSRHS$age-50)^2/100),
Y=dataSRHS$srhs)

# matrix of responses (with ordered categories from 0 to 4)
S = 5-res$YY
n = dim(S)[1]
# matrix of covariates (for the first and the following occasions)
# colums are: gender,race,educational level (2 columns),age,age^2)
X1 =res$XX[,1,]
X2 =res$XX[,2:TT,]

# estimate the model
est2f = est_lm_cov_latent(S,X1,X2,k=2,output=TRUE,out_se=TRUE)
summary(est2f)

# average transition probability matrix
PI = round(apply(est2f$PI[,,,2:TT],c(1,2),mean),4)

# Transition probability matrix for white females with high educational level
ind1 = (X1[,1]==1 & X1[,2]==0 & X1[,4]==1)
PI1 = round(apply(est2f$PI[,,ind1,2:TT],c(1,2),mean),4)

# Transition probability matrix for non-white male, low educational level
ind2 = (X1[,1]==0 & X1[,2]==1& X1[,3]==0 & X1[,4]==0)
PI2 = round(apply(est2f$PI[,,ind2,2:TT],c(1,2),mean),4)

## End(Not run)
```

---

est_lm_cov_latent_cont

Estimate LM model for continuous outcomes with covariates in the latent model

---

Description

Main function for estimating the LM model for continuous outcomes with covariates in the latent model.

Usage

```
est_lm_cov_latent_cont(Y, X1 = NULL, X2 = NULL, yv = rep(1,nrow(Y)), k, start = 0,
              tol = 10^-8, maxit = 1000, param = "multilogit", Mu = NULL, Si = NULL,
                 Be = NULL, Ga = NULL, output = FALSE)
```

Arguments

Y	array of continuous outcomes (n x TT x r)
X1	matrix of covariates affecting the initial probabilities (n x nc1)
X2	array of covariates affecting the transition probabilities (n x TT-1 x nc2)
yv	vector of frequencies of the available configurations
k	number of latent states
start	type of starting values (0 = deterministic, 1 = random, 2 = initial values in input)
tol	tolerance level for checking convergence of the algorithm
maxit	maximum number of iterations of the algorithm
param	type of parametrization for the transition probabilities ("multilogit" = standard multinomial logit for every row of the transition matrix, "difflogit" = multinomial logit based on the difference between two sets of parameters)
Mu	initial value of the conditional means (r x k) (if start=2)
Si	initial value of the var-cov matrix common to all states (r x r) (if start=2)
Be	intial value of the parameters affecting the logit for the initial probabilities (if start=2)
Ga	intial value of the parametes affecting the logit for the transition probabilities (if start=2)
output	to return additional output (V,PI,Piv,Ul)

Value

lk	maximum log-likelihood
Be	estimated array of the parameters affecting the logit for the initial probabilities
Ga	estimated array of the parameters affecting the logit for the transition probabilities
Mu	estimate of conditional means of the response variables
Si	estimate of var-cov matrix common to all states
np	number of free parameters
aic	value of AIC for model selection
bic	value of BIC for model selection
lkv	log-likelihood trace at every step
Piv	estimate of initial probability matrix
PI	estimate of transition probability matrices
Ul	matrix containing the predicted sequence of latent states by the local decoding method
call	command used to call the function

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia, http://www.stat.unipg.it/bartolucci

References

Bartolucci, F., Farcomeni, A. and Pennoni, F. (2013), Latent Markov Models for Longitudinal Data, Chapman and Hall/CRC press.

Examples

```
## Not run:
# Example based on multivariate longitudinal continuous data
# load data
require(mmm)
data(multiLongGaussian)
TT=4
res = long2matrices(multiLongGaussian$ID,X=cbind(multiLongGaussian$X,multiLongGaussian$time),
      Y=cbind(multiLongGaussian$resp1, multiLongGaussian$resp2))
Y = res$YY
X1 =res$XX[,1,]
X2 =res$XX[,2:TT,]

# estimate the model
est = est_lm_cov_latent_cont(Y,X1,X2,k=3,output=TRUE)
summary(est)

# average transition probability matrix
PI = round(apply(est$PI[,,,2:TT],c(1,2),mean),4); PI

## End(Not run)
```

---

est_lm_cov_manifest *Estimate LM model with covariates in the measurement model*

Description

Main function for estimating LM model with covariates in the measurement model based on a global logit parameterization.

Usage

```
est_lm_cov_manifest(S, X, yv = rep(1,nrow(S)), k, q = NULL, mod = c("LM", "FM"),
      tol = 10^-8, maxit = 1000, start = 0, mu = NULL, al = NULL, be = NULL, si = NULL,
       rho = NULL, la = NULL, PI = NULL, output = FALSE, out_se = FALSE)
```

Arguments

S	array of available configurations (n x TT) with categories starting from 0
X	array (n x TT x nc) of covariates with eventually includes lagged response (nc = number of covariates)
yv	vector of frequencies of the available configurations
k	number of latent states
q	number of support points for the AR(1) process
mod	model ("LM" = Latent Markov with stationary transition, "FM" = finite mixture)
tol	tolerance for the convergence (optional) and tolerance of conditional probability if tol>1 then return
maxit	maximum number of iterations of the algorithm
start	type of starting values (0 = deterministic, 1 = random, 2 = initial values in input)
mu	starting value for mu (optional)
al	starting value for al (optional)
be	starting value for be (optional)
si	starting value for si when mod="FM" (optional)
rho	starting value for rho when mod="FM" (optional)
la	starting value for la (optional)
PI	starting value for PI (optional)
output	to return additional output (PRED0, PRED1)
out_se	TRUE for computing information matrix and standard errors

Value

mu	vector of cutpoints
al	support points for the latent states
be	estimate of the vector of regression parameters
si	sigma of the AR(1) process (mod = "FM")
rho	parameter vector for AR(1) process (mod = "FM")
la	vector of initial probabilities
PI	transition matrix
lk	maximum log-likelihood
np	number of parameters
aic	value of AIC index
bic	value of BIC index
PRED0	prediction of latent state
PRED1	prediction of the overall latent effect
sebe	standard errors for the regression parameters be
selrho	standard errors for logit type transformation of rho
J1	information matrix
call	command used to call the function

Author(s)

Francesco Bartolucci, Silvia Pandolfi - University of Perugia (IT)

References

Bartolucci, F., Farcomeni, A. and Pennoni, F. (2013), Latent Markov Models for Longitudinal Data, Chapman and Hall/CRC press.

Bartolucci, F., Bacci, S. and Pennoni, F. (2014), Longitudinal analysis of the self-reported health status by mixture latent autoregressive models, Journal of the Royal Statistical Society - series C, 63, pp. 267-288

Examples

```
## Not run:
# Example based on self-rated health status (SRHS) data

# load SRHS data
data(data_SRHS_long)
dataSRHS = data_SRHS_long
head(dataSRHS)

res = long2matrices(dataSRHS$id,X=cbind(dataSRHS$gender-1,
dataSRHS$race==2|dataSRHS$race==3,dataSRHS$education==4,
dataSRHS$education==5,dataSRHS$age-50,(dataSRHS$age-50)^2/100),
Y=dataSRHS$srhs)

X =res$XX
S = 5-res$YY

# *** fit stationary LM model
res0 = vector("list",10); tol = 10^-6;
for(k in 1:10){
  res0[[k]] = est_lm_cov_manifest(S,X,k,1,mod="LM",tol)
   save.image("example_SRHS.RData")
}

# *** fit the mixture latent auto-regressive model
tol = 0.005
res = vector("list",4)
k=1
q = 51
res[[k]]=est_lm_cov_manifest(S,X,k,q,mod="FM",tol,output=TRUE)
for(k in 2:4) res[[k]]=est_lm_cov_manifest(S,X,k,q=61,mod="FM",tol,output=TRUE)

## End(Not run)
```

`est_lm_mixed` *Estimate mixed LM model*

Description

Main function for estimating the mixed LM model with discrete random effect in the latent model.

Usage

```
est_lm_mixed(S, yv = rep(1,nrow(S)), k1, k2, start = 0, tol = 10^-8, maxit = 1000,
   out_se = FALSE)
```

Arguments

`S`	array of available response configurations (n x TT x r) with categories starting from 0
`yv`	vector of frequencies of the available configurations
`k1`	number of latent classes
`k2`	number of latent states
`start`	type of starting values (0 = deterministic, 1 = random)
`tol`	tolerance level for convergence
`maxit`	maximum number of iterations of the algorithm
`out_se`	to compute standard errors

Value

`la`	estimate of the mass probability vector (distribution of the random effects)
`Piv`	estimate of initial probabilities
`Pi`	estimate of transition probability matrices
`Psi`	estimate of conditional response probabilities
`lk`	maximum log-likelihood
`W`	posterior probabilities of the random effect
`np`	number of free parameters
`bic`	value of BIC for model selection
`call`	command used to call the function

Author(s)

Francesco Bartolucci, Silvia Pandolfi - University of Perugia (IT)

References

Bartolucci, F., Farcomeni, A. and Pennoni, F. (2013), Latent Markov Models for Longitudinal Data, Chapman and Hall/CRC press.

Examples

```
## Not run:
# Example based of criminal data
# load data
data(data_criminal_sim)
out = long2wide(data_criminal_sim,"id","time","sex",
 c("y1","y2","y3","y4","y5","y6","y7","y8","y9","y10"),aggr=T,full=999)

XX = out$XX
YY = out$YY
freq = out$freq
n1 = sum(freq[XX[,1]==1])
n2 = sum(freq[XX[,1]==2])
n = sum(freq)

# fit mixed LM model only for females
YY = YY[XX[,1]==2,,]; freq = freq[XX[,1]==2]
k1 = 2; k2 = 2
res = est_lm_mixed(YY,freq,k1,k2,tol=10^-8)
summary(res)

## End(Not run)
```

---

est_mc_basic *Estimate basic Markov chain (MC) model*

Description

Main function for estimating the basic MC model.

Usage

```
est_mc_basic(S, yv, mod = 0, tol = 10^-8, maxit = 1000, out_se = FALSE)
```

Arguments

S	matrix (n x TT) of available configurations of the response variable with categories starting from 0
yv	vector of frequencies of the available configurations
mod	model on the transition probabilities (0 for time-heter., 1 for time-homog., from 2 to (TT-1) partial homog. of that order)
tol	tolerance level for convergence
maxit	maximum number of iterations of the algorithm
out_se	to compute the information matrix and standard errors

Value

lk	maximum log-likelihood
piv	estimate of initial probability vector
Pi	estimate of transition probability matrices
np	number of free parameters
aic	value of AIC for model selection
bic	value of BIC for model selection
Fy	estimated marginal distribution of the response variable for each time occasion
sepiv	standard errors for the initial probabilities
sePi	standard errors for the transition probabilities
call	command used to call the function

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

References

Bartolucci, F., Farcomeni, A. and Pennoni, F. (2013), Latent Markov Models for Longitudinal Data, Chapman and Hall/CRC press.

Examples

```
# Example of drug consumption data
# load data
data(data_drug)
data_drug = as.matrix(data_drug)
S = data_drug[,1:5]-1
yv = data_drug[,6]
# fit of the Basic MC model
out = est_mc_basic(S,yv,mod=1,out_se=TRUE)
summary(out)
```

---

est_mc_cov_latent *Estimate Markov chain (MC) model with covariates*

Description

Main function for estimating the MC model with covariates.

Usage

```
est_mc_cov_latent(S, X1 = NULL, X2 = NULL, yv = rep(1,nrow(S)), start = 0, tol = 10^-8,
   maxit = 1000, out_se = FALSE, output = FALSE, fort = TRUE)
```

Arguments

S	matrix of available configurations of the response variable (n x TT) with categories starting from 0
X1	matrix of covariates affecting the initial probabilities (n x nc1)
X2	array of covariates affecting the transition probabilities (n x TT-1 x nc2)
yv	vector of frequencies of the available configurations
start	type of starting values (0 = deterministic, 1 = random)
tol	tolerance level for checking convergence of the algorithm
maxit	maximum number of iterations of the algorithm
out_se	to compute the information matrix and standard errors
output	to return additional output (PI,Piv)
fort	to use fortran routine when possible (FALSE for not use fortran)

Value

lk	maximum log-likelihood
Be	estimated array of the parameters affecting the logit for the initial probabilities
Ga	estimated array of the parameters affecting the logit for the transition probabilities
np	number of free parameters
aic	value of AIC for model selection
bic	value of BIC for model selection
seBe	standard errors for Be
seGa	standard errors for Ga
Piv	estimate of initial probability matrix
PI	estimate of transition probability matrices
call	command used to call the function

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia, http://www.stat.unipg.it/bartolucci

References

Bartolucci, F., Farcomeni, A. and Pennoni, F. (2013), Latent Markov Models for Longitudinal Data, Chapman and Hall/CRC press.

Examples

```
## Not run:

# Example based on criminal data
# load criminal data
data(data_criminal_sim)

#We consider the response variable referring of crime of type 5

out = long2wide(data_criminal_sim,"id","time","sex",
"y5",aggr=T,full=999)
XX = out$XX-1
YY = out$YY
freq = out$freq
TT=6

X1 = as.matrix(XX[,1])
X2 = as.matrix(XX[,2:TT])
# estimate the model
res = est_mc_cov_latent(S=YY,yv=freq,X1=X1,X2=X2,output=TRUE)
summary(res)

# Initial probability for female
Piv0 = round(colMeans(res$Piv[X1==0,]),4)

# Initial probability for male
Piv1 = round(colMeans(res$Piv[X1==1,]),4)


## End(Not run)
```

`est_multilogit` *Estimate multilogit model*

Description

The function performs maximum likelihood estimation of the multilogit model (internal function).

Usage

```
est_multilogit(Y, Xdis, label = 1:n, be = NULL, Pdis = NULL, dis = FALSE,
   fort = TRUE, ex = FALSE)
```

Arguments

`Y`	matrix of all responses
`Xdis`	array of all covariates (two- or three-dimensional)
`label`	label associated to every covariate configuration

be	initial value of the parameter vector
Pdis	probability configurations as initial values
dis	to display partial results
fort	to use fortran routines when possible
ex	TRUE if the function exits without running the estimation algorithm providing the score vector and the information matrix

Value

be	estimated vector regression coefficients
P	estimated matrix of probabilities
Pdis	estimated matrix of distinct probabilities
sc	score vector
Fi	information matrix

Author(s)

Francesco Bartolucci, Silvia Pandolfi - University of Perugia (IT)

---

expit *Compute the expit function.*

Description

Function to compute the inverse of the logit function.

Usage

```
expit(x)
```

Arguments

x	a vector of matrix which contains the values to be transformed

Value

y	The transformed sample

Author(s)

Francesco Bartolucci, Silvia Pandolfi - University of Perugia (IT)

expit1 *Compute the expit function with respect to a reference category.*

Description

Function to compute the inverse of the logit function with respect to a reference category.

Usage

```
expit1(lp,ref=1)
```

Arguments

lp	a vector which contains the values to be transformed
ref	the reference category

Value

p	The transformed vector
Der	The matrix of derivatives

Author(s)

Francesco Bartolucci, Silvia Pandolfi - University of Perugia (IT)

invglob *Invert vector of global logits.*

Description

Function to invert the marginal parametrization based on global logits (internal funtion).

Usage

```
invglob(eta)
```

Arguments

eta	vector of global logits

Value

p	vector of joint probabilties

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

lk_ar_rho *Compute complete log-likelihood for AR(1) latent process*

Description

Compute the complete log-likelihood for the transtion and intial probabilties under the constraint of AR(1) latent process (internal use).

Usage

```
lk_ar_rho(lrho, SUP, V, outp = FALSE)
```

Arguments

lrho	Fisher transformation of the autocorrelation coefficient
SUP	matrix of the support points
V	array containing the posterior distribution of the latent states for each response configuration and time occasion
outp	to return addtional output (Wei, rho)

Value

flk	Minus the complete log-likelihood

Author(s)

Francesco Bartolucci, Silvia Pandolfi - University of Perugia (IT)

lk_comp_latent *Complete log-likelihood of the latent Markov model with covariates*

Description

Function that computes complete log-likelihood of the latent Markov model with covariates in the distribution of the latent process (internal use).

Usage

```
lk_comp_latent(S, R, yv, Piv, PI, Psi, k, fort = TRUE, der = FALSE,
       dlPsi = NULL, dlPiv = NULL, dlPI = NULL)
```

Arguments

S	matrix of distinct response configurations
R	matrix of missing response configurations
yv	corresponding vector of frequencies
Piv	initial probability matrix
PI	transition probability matrices
Psi	conditional response probability matrix
k	number of latent classes
fort	to use fortran routine when possible
der	to compute derivatives
dlPsi	matrix of derivatives of the logarithm of the conditional response probabilities
dlPiv	matrix of derivatives of the logarithm of the intial probabilities
dlPI	matrix of derivatives of the logarithm of the transition probabilities

Value

lk	log-likelihood
Phi	matrix of the conditional probabilities of the observed response configurations
L	matrix of the forward probabilities
pv	vector of marginal probabilities
dlk	derivatives of the log-likelihood
dlPhi	matrix of derivatives of the log-conditional probabilities of the observed response configurations
dlL	matrix of derivatives of the log-forward probabilities
dlL2	matrix of second derivatives of the log-forward probabilities
dlpv	matrix of derivatives of the log-marginal probabilities

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

References

Baum, L. E., Petrie, T., Soules, G., and Weiss, N. (1970). A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. Annals of Mathematical Statistics, 41, 164-171.

`lk_comp_latent_cont` *Complete log-likelihood of the latent Markov model for continuous outcomes with covariates*

Description

Function that computes complete log-likelihood of the latent Markov model for continuous outcomes with covariates in the distribution of the latent process (internal use).

Usage

```
lk_comp_latent_cont(Y, yv, Piv, PI, Mu, Si, k)
```

Arguments

`Y`	array of continuous response outcomes
`yv`	corresponding vector of frequencies
`Piv`	initial probability matrix
`PI`	transition probability matrices
`Mu`	matrix of conditional means of the response variables
`Si`	var-cov matrix common to all states
`k`	number of latent classes

Value

`lk`	log-likelihood
`Phi`	matrix of the conditional probabilities of the observed response configurations
`L`	matrix of the forward probabilities
`pv`	vector of marginal probabilities

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

References

Baum, L. E., Petrie, T., Soules, G., and Weiss, N. (1970). A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. Annals of Mathematical Statistics, 41, 164-171.

`lk_obs` *Compute the observable log-likelihood of the basic LM model*

Description

Function that computes the observable log-likelihood of the basic LM model (internal use).

Usage

```
lk_obs(th, m, Bm, Cm, bv, k, S, R, yv, TT, r, mod)
```

Arguments

`th`	vector of parameters
`m`	list of design matrices for the logits
`Bm`	design matrix for the logits
`Cm`	design matrix for the logits
`bv`	number of response categories
`k`	number of states
`S`	matrix of distinct response configurations
`R`	matrix of missing response configurations
`yv`	corresponding vector of frequencies
`TT`	number of times occasions
`r`	number of response variabels
`mod`	type of model

Value

`lk`	log-likelihood
`sc`	score vector

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

`lk_obs_latent` *Compute the observable log-likelihood of the LM model with covariates in the latent model*

Description

Function that computes the observable log-likelihood of the LM model with covariates in the latent model (internal use).

Usage

```
lk_obs_latent(th, S, R, b, yv, Am, XXdis, Xlab, ZZdis, Zlab, param, fort = TRUE)
```

Arguments

`th`	vector of parameters
`S`	matrix of distinct response configurations
`R`	matrix of missing response configurations
`b`	number of response categories
`yv`	corresponding vector of frequencies
`Am`	design matrix for the logits
`XXdis`	design matrix used for estimation of the initial probabilities
`Xlab`	list of labels used for estimation of the initial probabilities
`ZZdis`	design matrix used for estimation of the transition probabilities
`Zlab`	list of labels used for estimation of the transition probabilities
`param`	type of parametrization for the transition probabilities ("multilogit" = standard multinomial logit for every row of the transition matrix, "difflogit" = multinomial logit based on the difference between two sets of parameters)
`fort`	to use fortran routine when possible

Value

`lk`	log-likelihood
`sc`	score vector
`Psi`	conditional response probabilities
`be`	parameters on initial probabilities
`Ga`	parameters on transition probabilities
`Piv`	initial probabilities
`PI`	transition probabilities
`dlPsi`	matrix of derivatives of the logarithm of the conditional response probabilities
`dlPiv`	matrix of derivatives of the logarithm of the intial probabilities
`dlPI`	matrix of derivatives of the logarithm of the transition probabilities

Author(s)

Francesco Bartolucci, Silvia Pandolfi - University of Perugia (IT)

`lk_obs_manifest` *Compute the observable log-likelihood of the LM model with covariates in the measurement model*

Description

Function that computes the observable log-likelihood of the LM model with covariates in the measurement model (internal use).

Usage

```
lk_obs_manifest(par, Y, Xd, yv, indn, lev, k, sup, G2, IPI, mod, outp = FALSE)
```

Arguments

`par`	vector of regression parameters
`Y`	matrix of response variables
`Xd`	matrix of covariates
`yv`	vector of frequencies of response configurations
`indn`	index of the design matrix for each unit
`lev`	vector containing the number of levels of each variable
`k`	number of latent states
`sup`	support points
`G2`	design matrix
`IPI`	index of the transition matrix elements
`mod`	model (0 = LM with stationary transition, 1 = finite mixture)
`outp`	to compute the score of the observable log-likelihood

Value

`lk`	log-likelihood
`U`	array containing the posterior distribution of the latent states for each response configuration and each pair of consecutive time occasions
`s`	score of the observable log-likelihood

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

`lk_obs_mixed` *Compute the observable log-likelihood of the mixed LM model*

Description

Function that computes the observable log-likelihood of the mixed LM model with discrete random effect in the latent model (internal use).

Usage

```
lk_obs_mixed(th, nla, nPiv, nPi, nPsi, S, yv, r, k1, k2)
```

Arguments

`th`	vector of parameters
`nla`	number of logits for the mass probability vector
`nPiv`	number of logits for the intial probabilities
`nPi`	number of logits for the transition probabilities
`nPsi`	number of logits for the conditional response probabilities
`S`	matrix of distinct response configurations
`yv`	corresponding vector of frequencies
`r`	number of response variables
`k1`	number of latent classes
`k2`	number of latent states

Value

`lk`	log-likelihood
`sc`	score vector

Author(s)

Francesco Bartolucci, Silvia Pandolfi - University of Perugia (IT)

lk_sta *Compute the stationary log-likelihood*

Description

Function that computes the stationary log-likelihood (internal use).

Usage

```
lk_sta(tau, u, V, G2, outl = TRUE)
```

Arguments

tau	vector of parameters
u	vector of initial posterior probabilities
V	sum of the posterior probabilities
G2	design matrix
outl	to return additional output (la,PI)

Value

flk	log-likelihood

Author(s)

Francesco Bartolucci, Silvia Pandolfi - University of Perugia (IT)

logit1 *Compute the logit function with respect to a reference category.*

Description

Function to compute the logit function with respect to a reference category.

Usage

```
logit1(p, ref = 1)
```

Arguments

p	a vector which contains the values to be transformed
ref	the reference category

Value

lp	The transformed vector

Author(s)

Francesco Bartolucci, Silvia Pandolfi - University of Perugia (IT)

---

long2matrices *From data in the long format to data in array format*

Description

Function that transforms data in the long format to data in array format.

Usage

```
long2matrices(id, time = NULL, X = NULL, Y)
```

Arguments

id	vector of subjects id
time	vector of time occasions
X	matrix of covariates in long format
Y	matrix of responses in long format

Value

XX	array of covariates (n x TT x nc)
YY	array of responses (n x TT x r)

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

Examples

```
# Example based on SRHS data
# load SRHS data
data(data_SRHS_long)
dataSRHS = data_SRHS_long[1:1600,]
head(dataSRHS)
X = cbind(dataSRHS$gender-1,dataSRHS$race==2|dataSRHS$race==3,
dataSRHS$education==4,dataSRHS$education==5,dataSRHS$age-50,
(dataSRHS$age-50)^2/100)
Y = dataSRHS$srhs
res = long2matrices(dataSRHS$id,X=X,Y=Y)
```

long2wide *From data in the long format to data in the wide format*

Description

Function that transforms data in the long format to data in the wide format.

Usage

```
long2wide(data, nameid, namet, colx, coly, aggr = T, full = 999)
```

Arguments

data	matrix of data
nameid	name of the id column
namet	name of the t column
colx	vector of the names of the columns of the covariates
coly	vector of the names of the columns of the responses
aggr	if wide aggregated format is required
full	number to use for missing data

Value

listid	list of id for every unit
listt	list of the time occasions
data_wide	data in wide format
XX	array of the covariates
YY	array of the responses
freq	vector of the corresponding frequencies

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

Examples

```
# Example based on criminal data
# load criminal data
data(data_criminal_sim)
# consider only the first 1000 records to shorten time
out = long2wide(data_criminal_sim[1:1000,],"id","time","sex",
c("y1","y2","y3","y4","y5","y6","y7","y8","y9","y10"),aggr=TRUE,full=999)
```

marg_param *Compute marginal parametrization*

Description

Function that creates matrices C and M for the marginal parametrization of the probability vector for a vector of categorical variables (internal use)

Usage

```
marg_param(lev, type)
```

Arguments

lev	vector containing the number of levels of each variable
type	vector with elements "l", "g", "c", "r" indicating the type of logit

Value

C	matrix of constrats (the first sum(lev)-length(r) elements are referred to univariate logits)
M	marginalization matrix with elements 0 and 1
G	corresponding design matrix for the corresponding log-linear model

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

print.LMbasic *Print the output of LMbasic object*

Description

Given the output from LMbasic, it is written in a readable form

Usage

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

Arguments

x	output from LMbasic
...	further arguments passed to or from other methods

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

---

print.LMbasiccont *Print the output of LMbasiccont object*

Description

Given the output from LMbasiccont, it is written in a readable form

Usage

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

Arguments

x	output from LMbasiccont
...	further arguments passed to or from other methods

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

---

print.LMlatent *Print the output of LMlatent object*

Description

Given the output from LMlatent, it is written in a readable form

Usage

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

Arguments

x	output from LMlatent
...	further arguments passed to or from other methods

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

print.LMlatentcont *Print the output of LMlatentcont object*

Description

Given the output from LMlatentcont, it is written in a readable form

Usage

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

Arguments

x	output from LMlatentcont
...	further arguments passed to or from other methods

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

print.LMmanifest *Print the output of LMmanifest object*

Description

Given the output from LMmanifest, it is written in a readable form

Usage

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

Arguments

x	output from LMmanifest
...	further arguments passed to or from other methods

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

print.LMmixed *Print the output of LMmixed object*

Description

Given the output from LMmixed, it is written in a readable form

Usage

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

Arguments

x	output from LMmixed
...	further arguments passed to or from other methods

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

print.LMsearch *Print the output of LMsearch object*

Description

Given the output from LMsearch, it is written in a readable form

Usage

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

Arguments

x	output from LMsearch
...	further arguments passed to or from other methods

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

print.MCbasic *Print the output of MCbasic object*

Description

Given the output from MCbasic, it is written in a readable form

Usage

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

Arguments

x	output from MCbasic
...	further arguments passed to or from other methods

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

print.MClatent *Print the output of MClatent object*

Description

Given the output from MClatent, it is written in a readable form

Usage

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

Arguments

x	output from MClatent
...	further arguments passed to or from other methods

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

prob_multilogit *Compute multinomial probabilities*

Description

The function computes multinomial probabilities (internal function).

Usage

```
prob_multilogit(Xdis, be, label, fort = TRUE,der = FALSE)
```

Arguments

Xdis	array of all covariates (two- or three-dimensional)
be	initial value of the parameter vector
label	label associated to every covariate configuration
fort	to use fortran routines when possible
der	to compute derivatives

Value

P	estimated matrix of probabilities
Pdis	estimated matrix of distinct probabilities
dP	derivatives of the estimated matrix of probabilities
dPdis	derivatives of the estimated matrix of distinct probabilities

Author(s)

Francesco Bartolucci, Silvia Pandolfi - University of Perugia (IT)

prob_post_cov *Compute posterior probabilities.*

Description

Function that uses backward recursion to compute posterior probabilities (internal funtion).

Usage

```
prob_post_cov(S, yv, Psi, Piv, PI, Phi, L, pv, der = FALSE, fort = TRUE,
      dlPhi = NULL, dlPiv = NULL, dlPI = NULL, dlL = NULL, dlL2 = NULL, dlpv = NULL)
```

Arguments

S	matrix of distinct response configurations
yv	corresponding vector of frequencies
Psi	conditional response probabilities
Piv	initial probability matrix
PI	transition probability matrices
Phi	matrix of the conditional probabilities of the observed response configurations
L	matrix of the forward probabilities
pv	vector of marginal probabilities
der	to compute derivatives
fort	to use fortran routine when possible
dlPhi	matrix of derivatives of the log-conditional probabilities of the observed response configurations
dlPiv	matrix of derivatives of the logarithm of the intial probabilities
dlPI	matrix of derivatives of the logarithm of the transition probabilities
dlL	matrix of derivatives of the log-forward probabilities
dlL2	matrix of second derivatives of the log-forward probabilities
dlpv	matrix of derivatives of the log-marginal probabilities

Value

U	array containing the posterior distribution of the latent states for each response configuration and each pair of consecutive time occasions
V	array containing the posterior distribution of the latent states for each response configuration and time occasion
dlU	matrix of derivatives of the logarithm of U
dlV	matrix of derivatives of the logarithm of V

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

prob_post_cov_cont *Compute posterior probabilities.*

Description

Function that uses backward recursion to compute posterior probabilities (internal funtion).

Usage

```
prob_post_cov_cont(Y, yv, Mu, Si, Piv, PI, Phi, L, pv, der = FALSE,
      dlPhi = NULL, dlPiv = NULL, dlPI = NULL, dlL = NULL, dlL2 = NULL, dlpv = NULL)
```

Arguments

Y	array of continuous outcomes
yv	corresponding vector of frequencies
Mu	matrix of conditional means for the response variables
Si	var-cov matrix common to all states
Piv	initial probability matrix
PI	transition probability matrices
Phi	matrix of the conditional probabilities of the observed response configurations
L	matrix of the forward probabilities
pv	vector of marginal probabilities
der	to compute derivatives
dlPhi	matrix of derivatives of the log-conditional probabilities of the observed response configurations
dlPiv	matrix of derivatives of the logarithm of the intial probabilities
dlPI	matrix of derivatives of the logarithm of the transition probabilities
dlL	matrix of derivatives of the log-forward probabilities
dlL2	matrix of second derivatives of the log-forward probabilities
dlpv	matrix of derivatives of the log-marginal probabilities

Value

U	array containing the posterior distribution of the latent states for each response configuration and each pair of consecutive time occasions
V	array containing the posterior distribution of the latent states for each response configuration and time occasion
dlU	matrix of derivatives of the logarithm of U
dlV	matrix of derivatives of the logarithm of V

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

prod_array *Compute the product of array and vector*

Description

Function that compute the product of each matrix in X by the vector y (internal funtion).

Usage

```
prod_array(X, y)
```

Arguments

X	a given array
y	a given vector

Value

Z	the resulting matrix

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

rec1 *Recursions used by est_lm_cov_manifest*

Description

Implementation of the forward recursions to compute posterior distribution of the latent states (internal function).

Usage

```
rec1(Pio, las, PI)
```

Arguments

Pio	matrix of manifest probabilities
las	stationary initial probability
PI	transition probability matrices

Value

Q	matrices of posterior probabilities

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

References

Baum, L. E., Petrie, T., Soules, G., and Weiss, N. (1970). A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. Annals of Mathematical Statistics, 41, 164-171.

---

rec3 *Recursions used by est_lm_cov_manifest*

Description

Implementation of the backward recursions to compute posterior distribution of the latent states (internal function).

Usage

```
rec3(Q, yv, PI, Pio, pim)
```

Arguments

Q	posterior probabilities from the forward recursion
yv	vector of frequencies of response configurations
PI	transition probability matrices
Pio	matrix of manifest probabilities
pim	vector of marginal probabilities

Value

U	array containing the posterior distribution of the latent states for each response configuration and time occasion
V	matrix containing the posterior distribution of the latent states

Author(s)

Francesco Bartolucci, Silvia Pandolfi - University of Perugia (IT)

References

Baum, L. E., Petrie, T., Soules, G., and Weiss, N. (1970). A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. Annals of Mathematical Statistics, 41, 164-171.

recursions *Recursions used by est_lm_basic*

Description

Implementation of the recursions to compute manifest probability of the responses and posterior distribution of the latent states (internal function).

Usage

```
recursions(S, R, yv, Psi, piv, Pi, k, lth, matr, Bm, Cm, bv, mod)
```

Arguments

S	matrix of distinct response configurations
R	matrix of missing response configurations
yv	corresponding vector of frequencies
Psi	matrix of conditional response probabilties
piv	initial probability vector
Pi	transition probability matrices
k	number of latent states
lth	internal argument
matr	internal argument
Bm	internal argument
Cm	internal argument
bv	internal argument
mod	model on the transition probabilities (0 for time-heter., 1 for time-homog., from 2 to (TT-1) partial homog. of that order)

Value

lk	log-likelihood
sc	score vector
F1	internal argument
F2	internal argument
F1d	internal argument
F2d	internal argument

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

References

Baum, L. E., Petrie, T., Soules, G., and Weiss, N. (1970). A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. Annals of Mathematical Statistics, 41, 164-171.

RLMSdat *Dataset about job satisfaction*

Description

Longitudinal dataset deriving from the Russia Longitudinal Monitoring Survey (RLMS) about job satisfaction measured by an ordinal variable with 5 categories.

Usage

```
data(RLMSdat)
```

Format

A data frame with 1718 observations on the following 7 variables.

`IKSJQ` reported job satisfaction at the 1st occasion

`IKSJR` reported job satisfaction at the 2nd occasion

`IKSJS` reported job satisfaction at the 3rd occasion

`IKSJT` reported job satisfaction at the 4th occasion

`IKSJU` reported job satisfaction at the 5th occasion

`IKSJV` reported job satisfaction at the 6th occasion

`IKSJW` reported job satisfaction at the 7th occasion

Source

http://www.cpc.unc.edu/projects/rlms-hse, http://www.hse.ru/org/hse/rlms

References

Russia Longitudinal Monitoring survey, RLMS-HSE, conducted by Higher School of Economics and ZAO "Demoscope" together with Carolina Population Center, University of North Carolina at Chapel Hill and the Institute of Sociology RAS

Examples

```
data(RLMSdat)
```

search.model.LM *Search for the global maximum of the log-likelihood*

Description

Function that searches for the global maximum of the log-likelihood of different models given a vector of possible number of states to try for

Usage

```
search.model.LM(version = c("basic","latent","manifest","basic.cont",
      "latent.cont"), kv, ..., nrep=2, tol1 = 10^-5, tol2 = 10^-10,out_se=FALSE)
```

Arguments

version	model to be estimated ("basic" = basic LM model (est_lm_basic function); "latent" = LM model with covariates in the distribution of the latent process (est_lm_cov_latent function); "manifest" = LM model with covariates in the measurement model (est_lm_cov_maifest function),"basic.cont" = basic LM model for continuous outcomes (est_lm_basic_cont function); "latent.cont" = LM model for continuous outcomes with covariates in the distribution of the latent process (est_lm_cov_latent_cont function))
kv	vector of possible number of latent states
...	additional arguments to be passed based on the model to be estimated (see details)
nrep	number of repetitions of each random initialization
tol1	tolerance level for checking convergence of the algorithm in the random initializations
tol2	tolerance level for checking convergence of the algorithm in the last deterministic initialization
out_se	TRUE for computing information matrix and standard errors

Details

The function combines deterministic and random initializations strategy to reach the global maximum of the model log-likelihood. It uses one deterministic initialization (start=0) and a number of random initializations (start=1) proportional to the number of latent states. The tolerance level is set equal to 10^-5. Starting from the best solution obtained in this way, a final run is performed (start=2) with a default tolerance level equal to 10^-10.

Arguments in ... depend on the model to be estimated. They match the arguments to be passed to functions `est_lm_basic`, `est_lm_cov_latent`, `est_lm_cov_manifest`, `est_lm_basic_cont`, or `est_lm_cov_latent_cont`.

Value

out.single	output of each single model (as from est_lm_basic, est_lm_cov_latent or est_lm_cov_manifest) for each k in kv
aicv	value of AIC index for each k in kv
bicv	value of BIC index for each k in kv
lkv	value of log-likelihood for each k in kv

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

Examples

```
# example for est_lm_basic
data(data_drug)
data_drug = as.matrix(data_drug)
S = data_drug[,1:5]-1
yv = data_drug[,6]
n = sum(yv)
# Search Basic LM model

res = search.model.LM("basic",kv=1:4,S,yv,mod=1)
summary(res)
```

---

sq *Create a matrix with the combination of vectors of (1,0)*

Description

Function that creates a matrix with the combination of all possible row vectors with 't' elements equal to 1 and 'J-t' elements equal to 0. If 't' is NULL the function generates all the possible combinations of vectors of length 'J' with elements (1,0).

Usage

```
sq(J,t = NULL)
```

Arguments

J	an integer
t	an integer

Value

M	the resulting matrix

Author(s)

Francesco Bartolucci, Silvia Pandolfi - University of Perugia (IT)

`stationary` *Stationary*

Description

Function that computes the derivatives of the log-likelihood of a stationary model (internal function).

Usage

```
stationary(tau, k, G2, IPI)
```

Arguments

`tau`	regression parameters
`k`	number of latent states
`G2`	internal argument
`IPI`	index of the transition matrix elements

Value

`d0`	first derivatives of the log-likelihood
`d1`	second derivatives of the log-likelihood

Author(s)

Francesco Bartolucci, Silvia Pandolfi - University of Perugia (IT)

`summary.LMbasic` *Print the output of LMbasic object*

Description

Given the output from LMbasic, it is written in a readable form

Usage

```
## S3 method for class 'LMbasic'
summary(object, ...)
```

Arguments

object	output from LMbasic
...	further arguments passed to or from other methods

Value

table	summary of all the results

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

---

summary.LMbasiccont *Print the output of LMbasiccont object*

Description

Given the output from LMbasiccont, it is written in a readable form

Usage

```
## S3 method for class 'LMbasiccont'
summary(object, ...)
```

Arguments

object	output from LMbasiccont
...	further arguments passed to or from other methods

Value

table	summary of all the results

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

summary.LMlatent *Print the output of LMlatent object*

Description

Given the output from LMlatent, it is written in a readable form

Usage

```
## S3 method for class 'LMlatent'
summary(object, ...)
```

Arguments

object	output from LMlatent
...	further arguments passed to or from other methods

Value

table	summary of all the results

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

summary.LMlatentcont *Print the output of LMlatentcont object*

Description

Given the output from LMlatentcont, it is written in a readable form

Usage

```
## S3 method for class 'LMlatentcont'
summary(object, ...)
```

Arguments

object	output from LMlatentcont
...	further arguments passed to or from other methods

Value

table	summary of all the results

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

---

summary.LMmanifest *Print the output of LMmanifest object*

Description

Given the output from LMmanifest, it is written in a readable form

Usage

```
## S3 method for class 'LMmanifest'
summary(object, ...)
```

Arguments

object	output from LMmanifest
...	further arguments passed to or from other methods

Value

table	summary of all the results

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

---

summary.LMmixed *Print the output of LMmixed object*

Description

Given the output from LMmixed, it is written in a readable form

Usage

```
## S3 method for class 'LMmixed'
summary(object, ...)
```

Arguments

object	output from LMmixed
...	further arguments passed to or from other methods

Value

table	summary of all the results

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

---

summary.LMsearch *Print the output of LMsearch object*

Description

Given the output from LMsearch, it is written in a readable form

Usage

```
## S3 method for class 'LMsearch'
summary(object, ...)
```

Arguments

object	output from LMsearch
...	further arguments passed to or from other methods

Value

table	summary of all the results

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

summary.MCbasic *Print the output of MCbasic object*

Description

Given the output from MCbasic, it is written in a readable form

Usage

```
## S3 method for class 'MCbasic'
summary(object, ...)
```

Arguments

object	output from MCbasic
...	further arguments passed to or from other methods

Value

table	summary of all the results

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

summary.MClatent *Print the output of MClatent object*

Description

Given the output from MClatent, it is written in a readable form

Usage

```
## S3 method for class 'MClatent'
summary(object, ...)
```

Arguments

object	output from MClatent
...	further arguments passed to or from other methods

Value

table	summary of all the results

Author(s)

Francesco Bartolucci, Silvia Pandolfi, University of Perugia (IT), http://www.stat.unipg.it/bartolucci

---

trans_par *Convert matrix parametrization*

Description

Function that converts matrix parametrization (internal function).

Usage

```
trans_par(par,lev,k,sup,G2,IPI,mod)
```

Arguments

par	regression parameters
lev	vector containing the number of levels of each variable
k	number of latent states
sup	vectors of support points
G2	internal argument
IPI	internal argument
mod	model (0 = LM with stationary transition, 1 = finite mixture)

Value

la	vector of initial probabilities
PI	transition matrix
rho	parameter vector for AR
si	sigma of the AR process
par	regression parameters
lrho	Fisher transformation of the autocorrelation coefficient
tau	regression parameters

Author(s)

Francesco Bartolucci, Silvia Pandolfi - University of Perugia (IT)

Index