Package ‘ifa’

February 20, 2015

Type Package

Title Independent Factor Analysis

Version 7.0

Date 2012-14-01

Author Cinzia Viroli

Maintainer Cinzia Viroli <cinzia.viroli@unibo.it>

Imports mvtnorm

Depends R(>= 1.9.0)

Description The package performes Independent Factor Analysis

License GPL-2

URL www2.stat.unibo.it/viroli

Repository CRAN

Date/Publication 2012-01-17 16:48:28

NeedsCompilation yes

R topics documented:

ifa.aic *A function to compute the AIC*

Description

A function to compute the Akaike Information Criterion (AIC) for the fitted IFA model, according to the formula `-2*log-likelihood + 2*npar`, where `npar` represents the number of parameters.

Usage

```
ifa.aic(output)
```

Arguments

`output`	The fitted IFA model object, a list including the log-likelihood and the number of parameters

Value

It returns a numeric value with the corresponding AIC.

Author(s)

Cinzia Viroli

References

Sakamoto, Y., Ishiguro, M., and Kitagawa G. (1986). Akaike Information Criterion Statistics. D. Reidel Publishing Company.

Viroli, C. (2005). Choosing the number of factors in Independent Factor Analysis model, Metodoloski Zvezki, Advances in Methodology and Statistics, Vol. II, N. 2, 219-229. Available at $www2.stat.unibo.it/viroli$.

See Also

ifa.bic

Examples

```
data(memory)
fit<-ifa.em(memory$x,c(2,2),it=50,eps=0.001)
ifa.aic(fit)
```

ifa.bic *A function to compute the BIC*

Description

A function to compute the Bayesian Information Criterion (BIC), also known as Schwarz's Bayesian criterion (SBC), for the fitted IFA model, according to the formula `-2*log-likelihood + npar*log(nobs)`, where `npar` represents the number of parameters and `nobs` the number of observations in the fitted model.

Usage

```
ifa.bic(output)
```

Arguments

`output`	The fitted IFA model object, a list including the log-likelihood, the number of observations and the number of parameters.

Value

It returns a numeric value with the corresponding BIC.

Author(s)

Cinzia Viroli

References

Schwarz, G. (1978) Estimating the Dimension of a Model, Annals of Statistics, 6, 461-464.

Viroli, C. (2005). Choosing the number of factors in Independent Factor Analysis model, Metodoloski Zvezki, Advances in Methodology and Statistics, Vol. II, N. 2, 219-229. Available at $www2.stat.unibo.it/viroli$.

See Also

ifa.aic

Examples

```
data(memory)
fit<-ifa.em(memory$x,c(2,2),it=50,eps=0.001)
ifa.bic(fit)
```

ifa.em	*Fitting an Independent Factor Analysis model by the EM algorithm*

Description

`ifa.em` is used to perform Independent Factor Analysis on a matrix of data by the Expectation Maximization algorithm.

Usage

```
ifa.em(y, ni, it = 15, eps = 0.001, init = NULL, scaling = TRUE)
```

Arguments

`y`	A data matrix with `n` rows representing observations and `p` columns representing variables.
`ni`	A vector containing the number of mixture components for modeling each factor. The number of factors is equal to the length of this vector.
`it`	The maximum number of iterations of the EM algorithm. By default it is set to 15.
`eps`	The lower bound for relative variation of the likelihood. It is used as alternative stopping rule for the EM algorithm: if the relative increment of the likelihood is lower than `eps` the fitting is stopped. The default is 0.001.
`init`	A list containing initial values for the factor loading matrix (specified by `H`) and the variance matrix of the noise term (specified by `psi`). If `NULL` (default) the algorithm starts from the ordinal factor model solution.
`scaling`	If TRUE (default) the data are scaled before fitting the IFA model

Details

Independent Factor Analysis is a latent variable model with independent and non gaussian factors. The `p` observed variables `x` are modelled in terms of a smaller set of `k` unobserved independent latent variables, `y`, and an additive specific term u: `x=Hy+u`, where `u` is assumed to be normally distributed with diagonal variance matrix `Psi` and the factor loading matrix `H` is also termed as `mixing matrix`. The density of each factor is modelled by a mixture of gaussians. The model is fitted by the EM algorithm. This version can be computationally slow in the complex cases. A faster R package for window which is based on fortran code can be downloaded at the home: $www2.stat.unibo.it/viroli$ in the section Software and Data.

Value

A list containing the components:

`H`	The estimated factor loading matrix
`lik`	The log-likelihood computed at each iteration of the EM algorithm

`w` A matrix with the estimated weights of the mixtures. Each row contains the weights of each factor

`mu` A matrix with the estimated component means of the mixtures. Each row contains the vector means for each factor

`vu` A matrix with the estimated component variances of the mixtures. Each row contains the vector variances for each factor

`psi` The noise diagonal variance matrix

`ni` The input vector specifying the number of components for each mixture

`L` The number of factors

`numvar` The number of observed variables

`numobs` The number of observations

Author(s)

Cinzia Viroli

References

Attias H. (1999), Independent Factor Analysis, Neural Computation, 11, 803–851.

Montanari A. and Viroli C. (2010), The Independent Factor Analysis approach to latent variable modeling, Statistics, 44, 397–416.

See Also

`ifa.init.random`, `ifa.init.pca`

Examples

```
data(memory)
init.values<-ifa.init.random(memory$x,2)
fit<-ifa.em(memory$x,c(2,2),it=50,eps=0.0001,init.values)

fit<-ifa.em(memory$x,c(2,2),it=50,eps=0.0001)
plot(ifa.predict(scale(memory$x),fit))
```

`ifa.init.pca` *Internal function to initialize the IFA parameters to ordinary factor solution*

Description

An internal function to initialize the IFA parameters to ordinary factor solution. This is not to be called by the user, since it is automatically called by the function `ifa.em` if the option `init` is `NULL` (default).

See Also

`ifa.em`,`ifa.init.random`

ifa.init.random *A function to randomly inizialize the IFA parameters*

Description

A function to initialize the IFA parameters to random values.

Usage

```
ifa.init.random(y,L)
```

Arguments

`y`	The data to be fitted
`L`	The number of factors

Value

A list containing 2 initialized parameters:

`H`	The factor loading matrix
`psi`	The noise diagonal variance matrix

See Also

`ifa.init.pca`

Examples

```
data(memory)
init.values<-ifa.init.random(memory$x,2)
fit<-ifa.em(memory$x,c(2,2),it=50,eps=0.0001,init.values)
```

ifa.predict *A function to predict the latent variables*

Description

A function to compute the reconstructed factors.

Usage

```
ifa.predict(y, output, method = ”lms”)
```

Arguments

`y`	The data matrix for which factor reconstructions are desired
`output`	The result of a call to ifa.em
`method`	The method for reconstructing the factors

Details

The function reconstructs the factor scores by different methods: `"bartlett"`, `"thompson"` or `"lms"` (least mean squares). The default is `"lms"`.

Value

A matrix with factors in columns.

Author(s)

Cinzia Viroli

`memory` *Memory Tests on Alzheimer and Depressed Patients*

Description

Data from study by Robert Hart, concerning ability of memory tests to distinguish between patients with mild Alzheimer's disease, depression, and controls.

Usage

```
data(memory)
```

Format

memory is a list of 2 components: `$y` is the group variable (1=Alzheimer; 2=Depressed; 3=Normal) and `$x` is a data.frame containing 45 obs. of 6 variables:

HIR : HI Imagery Recall

LIR : LO Imagery Recall

HIUM: HI Imagery Unreminded Memory

LIUM: LO Imagery Unreminded Memory

SM : Storage

RM : Recognition Memory

References

Hart, R.P., Kwentus, J.A., Taylor, J.R., \& Hamer, R.M. (1988), Productive naming and memory in depression and Alzheimer's type dementia, Archives of Clinical Neuropsychology, 3, 313-322.

Examples

```
data(memory)
```

Index