

Package ‘VEMIRT’

October 20, 2022

Type Package

Title Variational Expectation Maximization for High-dimensional IRT Models

Version 1.0

Date 2022-10-16

Maintainer Jiaying Xiao <jxiao6@uw.edu>

Description VEMIRT is created to assist researchers to conduct exploratory and confirmatory multidimensional item response theory (MIRT) analysis.

The core computation engine of VEMIRT is a family of Gaussian Variational EM algorithms that are considerably more efficient than currently available algorithms in other software packages, especially when the number of latent factors exceeds 4.

License GPL-3

Imports GPArotation, Matrix, polycor, psych, Rcpp (>= 1.0.9), testit

LinkingTo Rcpp, RcppArmadillo

RoxygenNote 7.2.1

Encoding UTF-8

Depends R (>= 2.10)

LazyData true

Author Jiaying Xiao [aut, cre],
Gongjun Xu [aut],
Chun Wang [aut]

Archs x64

R topics documented:

VEMIRT-package	2
exampleData_2pl	3
exampleData_3pl	3
exampleIndic_cfa2pl	3
exampleIndic_cfa3pl	4
exampleIndic_efa2pl_c1	4
exampleIndic_efa2pl_c2	4
exampleIndic_efa3pl_c1	5
exampleIndic_efa3pl_c2	5

gvem_2PLCFA	6
gvem_2PLEFA_adaptlasso	7
gvem_2PLEFA_lasso	8
gvem_2PLEFA_rot	10
pa_poly	11
sgvem_3PLCFA	11
sgvem_3PLEFA_adaptlasso	13
sgvem_3PLEFA_lasso	15
sgvem_3PLEFA_rot	17
Index	19

VEMIRT-package	<i>VEMIRT: Variational Expectation Maximization for High-dimensional IRT Models</i>
----------------	-------------------------------------------------------------------------------------

Description

VEMIRT is created to assist researchers to conduct exploratory and confirmatory multidimensional item response theory (MIRT) analysis. The core computation engine of VEMIRT is a family of Gaussian Variational EM algorithms that are considerably more efficient than currently available algorithms in other software packages, especially when the number of latent factors exceeds 4.

Details

The package includes three modules: parallel analysis, exploratory and confirmatory analysis for M2PL and M3PL models. The number of factors can be identified via parallel analysis using the [pa_poly](#) function. To conduct the exploratory analysis, the Gaussian Variational EM (GVEM) algorithms with post-hoc rotation (Promax & CF-Quartimax), Lasso, or adaptive Lasso are provided. The stochastic GVEM is implemented to further improve the computational efficacy when analyzing M3PL models. The package also supports the confirmatory analysis by using the [gvem_2PLCFA](#) and [sgvem_3PLCFA](#) functions.

Author(s)

- Jiaying Xiao <jxiao6@uw.edu>
- Gongjun Xu <gongjun@umich.edu>
- Chun Wang <wang4066@uw.edu>

References

- Cho, A. E., Wang, C., Zhang, X., & Xu, G. (2021). Gaussian variational estimation for multidimensional item response theory. *British Journal of Mathematical and Statistical Psychology*, 74, 52-85.
- Cho, A. E., Xiao, J., Wang, C., & Xu, G. (2022). Regularized Variational Estimation for Exploratory Item Factor Analysis. *Psychometrika*. <https://doi.org/10.1007/s11336-022-09874-6>

exampleData_2pl	<i>Response data set for M2PL</i>
-----------------	-----------------------------------

Description

The response data set is simulated based on a between-item M2PL model with 5 factors. The true factor correlations are set as 0.1.

Usage

```
exampleData_2pl
```

Format

A data frame with 2000 respondents and 75 items

exampleData_3pl	<i>Response data set for M3PL</i>
-----------------	-----------------------------------

Description

The response data set is simulated based on a within-item M3PL model with 3 factors. The true factor correlations are set as 0.1.

Usage

```
exampleData_3pl
```

Format

A data frame with 2000 respondents and 45 items

exampleIndic_cfa2pl	<i>Factor-loading indicator matrix for M2PL-CFA</i>
---------------------	-----------------------------------------------------

Description

The factor-loading indicator matrix can be used as an input for confirmatory factor analysis.

Usage

```
exampleIndic_cfa2pl
```

Format

A data frame with 75 items and 5 factors

exampleIndic_cfa3pl	<i>Factor-loading indicator matrix for M3PL-CFA</i>
---------------------	-----------------------------------------------------

Description

The factor-loading indicator matrix can be used as an input for confirmatory factor analysis.

Usage

```
exampleIndic_cfa3pl
```

Format

A data frame with 45 items and 3 factors

exampleIndic_efa2pl_c1	<i>Factor-loading indicator matrix for M2PL-EFA with lasso/ adaptive penalty under constraint 1</i>
------------------------	-----------------------------------------------------------------------------------------------------

Description

The factor-loading indicator matrix can be used as an input for exploratory factor analysis with lasso/ adaptive lasso penalty under constraint 1.

Usage

```
exampleIndic_efa2pl_c1
```

Format

A data frame with 75 items and 5 factors. Items 1, 16, 31, 46 and 61 can be combined as an identity matrix to satisfy constraint 1

exampleIndic_efa2pl_c2	<i>Factor-loading indicator matrix for M2PL-EFA with lasso/ adaptive penalty under constraint 2</i>
------------------------	-----------------------------------------------------------------------------------------------------

Description

The factor-loading indicator matrix can be used as an input for exploratory factor analysis with lasso/ adaptive lasso penalty for constraint 1.

Usage

```
exampleIndic_efa2pl_c2
```

Format

A data frame with 75 items and 5 factors. Items 1, 16, 31, 46 and 61 can be combined as a triangular matrix to satisfy constraint 2

`exampleIndic_efa3pl_c1`

Factor-loading indicator matrix for M3PL-EFA with lasso/ adaptive penalty under constraint 1

Description

The factor-loading indicator matrix can be used as an input for exploratory factor analysis with lasso/ adaptive lasso penalty under constraint 1.

Usage`exampleIndic_efa3pl_c1`**Format**

A data frame with 45 items and 3 factors. Items 1, 16, and 19 can be combined as an identity matrix to satisfy constraint 1

`exampleIndic_efa3pl_c2`

Factor-loading indicator matrix for M3PL-EFA with lasso/ adaptive penalty under constraint 2

Description

The factor-loading indicator matrix can be used as an input for exploratory factor analysis with lasso/ adaptive lasso penalty for constraint 1.

Usage`exampleIndic_efa3pl_c2`**Format**

A data frame with 45 items and 3 factors. Items 1, 16, and 19 can be combined as a triangular matrix to satisfy constraint 2

gvem_2PLCFA

Confirmatory M2PL Analysis

Description

Confirmatory M2PL Analysis

Usage

```
gvem_2PLCFA(u, indic, max.iter = 5000)
```

Arguments

<code>u</code>	a $N \times J$ matrix or a <code>data.frame</code> that consists of binary responses of N individuals to J items. The missing values are coded as NA
<code>indic</code>	a $J \times K$ matrix or a <code>data.frame</code> that describes the factor loading structure of J items to K factors. It consists of binary values where 0 refers to the item is irrelevant with this factor, 1 otherwise
<code>max.iter</code>	the maximum number of iterations for the EM cycle; default is 5000

Value

a list containing the following objects:

<code>ra</code>	item discrimination parameters, a $J \times K$ matrix
<code>rb</code>	item difficulty parameters, vector of length J
<code>reta</code>	variational parameters $\eta(\xi)$, a $N \times J$ matrix
<code>reps</code>	variational parameters ξ , a $N \times J$ matrix
<code>rsigma</code>	population variance-covariance matrix, a $K \times K$ matrix
<code>mu_i</code>	mean parameter for each person, a $K \times N$ matrix
<code>sig_i</code>	covariance matrix for each person, a $K \times K \times N$ array
<code>n</code>	the number of iterations for the EM cycle
<code>Q_mat</code>	factor loading structure, a $J \times K$ matrix
<code>GIC</code>	model fit index
<code>AIC</code>	model fit index
<code>BIC</code>	model fit index

See Also

[sgvem_3PLCFA](#)

Examples

```
gvem_2PLCFA(exampleData_2pl, exampleIndic_cfa2pl)
```

gvem_2PLEFA_adaptlasso

Exploratory M2PL Analysis with Adaptive Lasso Penalty

Description

Exploratory M2PL Analysis with Adaptive Lasso Penalty

Usage

```
gvem_2PLEFA_adaptlasso(
  u,
  indic,
  max.iter = 5000,
  constrain = "C1",
  non_pen = NULL,
  gamma = 2
)
```

Arguments

u	a $N \times J$ matrix or a data.frame that consists of binary responses of N individuals to J items. The missing values are coded as NA
indic	a $J \times K$ matrix or a data.frame that describes the factor loading structure of J items to K factors. It consists of binary values where 0 refers to the item is irrelevant with this factor, 1 otherwise. For exploratory factor analysis with adaptive lasso penalty, indic should be imposed certain constraints on the a $K \times K$ sub-matrix to ensure identifiability. The remaining parts do not assume any pre-specified zero structure but instead, the appropriate lasso penalty would recover the true zero structure. Also see constrain
max.iter	the maximum number of iterations for the EM cycle; default is 5000
constrain	the constraint setting: "C1" or "C2". To ensure identifiability, "C1" sets a $K \times K$ sub-matrix of indic to be an identity matrix, and the variance of the population variance-covariance matrix to be 1. Instead of setting all off-diagonals of a sub-matrix to be zero, "C2" keeps the sub-matrix to be a triangular matrix with the diagonal being ones. That is, there are test items associated with each factor for sure and they may be associated with other factors as well. Nonzero entries except for the diagonal entries of the sub-matrix are penalized during the estimation procedure. Although, "C2" is much weaker than "C1", it still ensures empirical identifiability. Default is "C1"
non_pen	the index of an item which is associated with each factor to satisfy "C2". For C1, the input can be NULL
gamma	a numerical value of adaptive lasso parameter. Zou (2006) recommended three values, 0.5, 1, and 2. The default value is 2.

Value

a list containing the following objects:

ra item discrimination parameters, a $J \times K$ matrix

rb	item difficulty parameters, vector of length J
reta	variational parameters $\eta(\xi)$, a $N \times J$ matrix
reps	variational parameters ξ , a $N \times J$ matrix
rsigma	population variance-covariance matrix, a $K \times K$ matrix
mu_i	mean parameter for each person, a $K \times N$ matrix
sig_i	covariance matrix for each person, a $K \times K \times N$ array
n	the number of iterations for the EM cycle
Q_mat	factor loading structure, a $J \times K$ matrix
GIC	model fit index
AIC	model fit index
BIC	model fit index
lbd	numerical value of lasso penalty parameter λ

References

Cho, A. E., Xiao, J., Wang, C., & Xu, G. (2022). Regularized Variational Estimation for Exploratory Item Factor Analysis. *Psychometrika*. <https://doi.org/10.1007/s11336-022-09874-6>

Zou, H. (2006). The adaptive LASSO and its oracle properties. *Journal of the American Statistical Association*, 7, 1011418–1429.

See Also

[gvem_2PLEFA_rot](#), [gvem_2PLEFA_lasso](#), [exampleIndic_efa2pl_c1](#), [exampleIndic_efa2pl_c2](#)

Examples

```
gvem_2PLEFA_adaptlasso(exampleData_2pl, exampleIndic_efa2pl_c1, constrain="C1", non_pen=NULL, gamma=2)
gvem_2PLEFA_adaptlasso(exampleData_2pl, exampleIndic_efa2pl_c2, constrain="C2", non_pen=61, gamma=2)
```

gvem_2PLEFA_lasso	<i>Exploratory M2PL Analysis with Lasso Penalty</i>
-------------------	-----------------------------------------------------

Description

Exploratory M2PL Analysis with Lasso Penalty

Usage

```
gvem_2PLEFA_lasso(u, indic, max.iter = 5000, constrain = "C1", non_pen = NULL)
```


Arguments

<code>u</code>	a $N \times J$ matrix or a data.frame that consists of binary responses of N individuals to J items. The missing values are coded as NA
<code>indic</code>	a $J \times K$ matrix or a data.frame that describes the factor loading structure of J items to K factors. It consists of binary values where 0 refers to the item is irrelevant with this factor, 1 otherwise. For exploratory factor analysis with lasso penalty, <code>indic</code> should be imposed certain constraints on the a $K \times K$ sub-matrix to ensure identifiability. The remaining parts do not assume any pre-specified zero structure but instead, the appropriate lasso penalty would recover the true zero structure. Also see <code>constrain</code>
<code>max.iter</code>	the maximum number of iterations for the EM cycle; default is 5000
<code>constrain</code>	the constraint setting: "C1" or "C2". To ensure identifiability, "C1" sets a $K \times K$ sub-matrix of <code>indic</code> to be an identity matrix, and the variance of the population variance-covariance matrix to be 1. Instead of setting all off-diagonals of a sub-matrix to be zero, "C2" keeps the sub-matrix to be a triangular matrix with the diagonal being ones. That is, there are test items associated with each factor for sure and they may be associated with other factors as well. Nonzero entries except for the diagonal entries of the sub-matrix are penalized during the estimation procedure. Although, "C2" is much weaker than "C1", it still ensures empirical identifiability. Default is "C1"
<code>non_pen</code>	the index of an item which is associated with each factor to satisfy "C2". For C1, the input can be NULL

Value

a list containing the following objects:

<code>ra</code>	item discrimination parameters, a $J \times K$ matrix
<code>rb</code>	item difficulty parameters, vector of length J
<code>reta</code>	variational parameters $\eta(\xi)$, a $N \times J$ matrix
<code>reps</code>	variational parameters ξ , a $N \times J$ matrix
<code>rsigma</code>	population variance-covariance matrix, a $K \times K$ matrix
<code>mu_i</code>	mean parameter for each person, a $K \times N$ matrix
<code>sig_i</code>	covariance matrix for each person, a $K \times K \times N$ array
<code>n</code>	the number of iterations for the EM cycle
<code>Q_mat</code>	factor loading structure, a $J \times K$ matrix
<code>GIC</code>	model fit index
<code>AIC</code>	model fit index
<code>BIC</code>	model fit index
<code>lbd</code>	numerical value of lasso penalty parameter λ

References

Cho, A. E., Xiao, J., Wang, C., & Xu, G. (2022). Regularized Variational Estimation for Exploratory Item Factor Analysis. *Psychometrika*. <https://doi.org/10.1007/s11336-022-09874-6>

See Also

[gvem_2PLEFA_rot](#), [gvem_2PLEFA_adaptlasso](#), [exampleIndic_efa2pl_c1](#), [exampleIndic_efa2pl_c2](#)

Examples

```
gvem_2PLEFA_lasso(exampleData_2pl, exampleIndic_efa2pl_c1, constrain="C1")
gvem_2PLEFA_lasso(exampleData_2pl, exampleIndic_efa2pl_c2, constrain="C2", non_pen=61)
```

gvem_2PLEFA_rot

*Exploratory M2PL Analysis with Post-hoc Rotation***Description**

Exploratory M2PL Analysis with Post-hoc Rotation

Usage

```
gvem_2PLEFA_rot(u, domain, max.iter = 5000, rot = "Promax")
```

Arguments

u	a $N \times J$ matrix or a data.frame that consists of binary responses of N individuals to J items. The missing values are coded as NA
domain	the number of factors
max.iter	the maximum number of iterations for the EM cycle; default is 5000
rot	the post-hoc rotation method: Promax or CF-Quartimax; default is "Promax", but may also be "cfQ" for conducting the CF-Quartimax rotation

Value

a list containing the following objects:

ra	item discrimination parameters, a $J \times K$ matrix
rb	item difficulty parameters, vector of length J
reta	variational parameters $\eta(\xi)$, a $N \times J$ matrix
reps	variational parameters ξ , a $N \times J$ matrix
rsigma	population variance-covariance matrix, a $K \times K$ matrix
mu_i	mean parameter for each person, a $K \times N$ matrix
sig_i	covariance matrix for each person, a $K \times K \times N$ array
n	the number of iterations for the EM cycle
rk	factor loadings, a $J \times K$ matrix
Q_mat	factor loading structure, a $J \times K$ matrix
GIC	model fit index
AIC	model fit index
BIC	model fit index
ur_a	item discrimination parameters before conducting the rotation, a $J \times K$ matrix

See Also

[gvem_2PLEFA_lasso](#), [gvem_2PLEFA_adaptlasso](#)

Examples

```
gvem_2PLCFA(exampleData_2pl, domain=5,max.iter=3000)
gvem_2PLCFA(exampleData_2pl, domain=5,rot="cfQ")
```

pa_poly	<i>Parallel analysis using polychoric correlation</i>
---------	-------------------------------------------------------

Description

Identify the number of factors

Usage

```
pa_poly(data, n.iter = 10, figure = TRUE)
```

Arguments

data	a $N \times J$ matrix or a data.frame that consists of the responses of N individuals to J items without any missing values. The responses are binary or polytomous.
n.iter	Number of simulated analyses to perform
figure	By default, pa_poly draws an eigenvalue plot. If FALSE, it suppresses the graphic output

Value

pa_poly returns a data.frame with the eigenvalues for the real data and the simulated data.

Examples

```
pa_poly(exampleData_2pl, n.iter=20)
```

sgvem_3PLCFA	<i>Stochastic GVEM for Confirmatory M3PL Analysis</i>
--------------	-------------------------------------------------------

Description

Stochastic GVEM for Confirmatory M3PL Analysis

Usage

```
sgvem_3PLCFA(
  u,
  indic,
  samp = 50,
  forgetrate = 0.51,
  mu_b,
  sigma2_b,
  Alpha,
  Beta,
  max.iter = 5000
)
```

Arguments

<code>u</code>	a $N \times J$ matrix or a <code>data.frame</code> that consists of binary responses of N individuals to J items. The missing values are coded as NA
<code>indic</code>	a $J \times K$ matrix or a <code>data.frame</code> that describes the factor loading structure of J items to K factors. It consists of binary values where 0 refers to the item is irrelevant with this factor, 1 otherwise
<code>samp</code>	a subsample for each iteration; default is 50
<code>forgetrate</code>	the forget rate for the stochastic algorithm. The value should be within the range from 0.5 to 1. Default is 0.51
<code>mu_b</code>	the mean parameter for the prior distribution of item difficulty parameters
<code>sigma2_b</code>	the variance parameter for the prior distribution of item difficulty parameters
<code>Alpha</code>	the α parameter for the prior distribution of guessing parameters
<code>Beta</code>	the β parameter for the prior distribution of guessing parameters
<code>max.iter</code>	the maximum number of iterations for the EM cycle; default is 5000

Value

a list containing the following objects:

<code>ra</code>	item discrimination parameters, a $J \times K$ matrix
<code>rb</code>	item difficulty parameters, vector of length J
<code>rc</code>	item guessing parameters, vector of length J
<code>rs</code>	variational parameters s , a $N \times J$ matrix
<code>reta</code>	variational parameters $\eta(\xi)$, a $N \times J$ matrix
<code>reps</code>	variational parameters ξ , a $N \times J$ matrix
<code>rsigma</code>	population variance-covariance matrix, a $K \times K$ matrix
<code>mu_i</code>	mean parameter for each person, a $K \times N$ matrix
<code>sig_i</code>	covariance matrix for each person, a $K \times K \times N$ array
<code>n</code>	the number of iterations for the EM cycle
<code>Q_mat</code>	factor loading structure, a $J \times K$ matrix
<code>GIC</code>	model fit index
<code>AIC</code>	model fit index
<code>BIC</code>	model fit index

References

- Cho, A. E., Wang, C., Zhang, X., & Xu, G. (2021). Gaussian variational estimation for multidimensional item response theory. *British Journal of Mathematical and Statistical Psychology*, 74, 52-85.
- Cho, A. E., Xiao, J., Wang, C., & Xu, G. (2022). Regularized Variational Estimation for Exploratory Item Factor Analysis. *Psychometrika*. <https://doi.org/10.1007/s11336-022-09874-6>

See Also

[gvem_2PLCFA](#)

Examples

```
sgvem_3PLCFA(exampleData_3pl, exampleIndic_cfa3pl, samp=50, forgetrate=0.51, mu_b=0, sigma2_b=4, Alpha=10, Beta=4)
```

sgvem_3PLEFA_adaptlasso

Stochastic GVEM with Adaptive Lasso Penalty for Exploratory M3PL Analysis

Description

Stochastic GVEM with Adaptive Lasso Penalty for Exploratory M3PL Analysis

Usage

```
sgvem_3PLEFA_adaptlasso(
  u,
  indic,
  samp = 50,
  forgetrate = 0.51,
  mu_b,
  sigma2_b,
  Alpha,
  Beta,
  max.iter = 5000,
  constrain = "C1",
  non_pen = NULL,
  gamma = 2
)
```

Arguments

u	a $N \times J$ matrix or a data.frame that consists of binary responses of N individuals to J items. The missing values are coded as NA
indic	a $J \times K$ matrix or a data.frame that describes the factor loading structure of J items to K factors. It consists of binary values where 0 refers to the item is irrelevant with this factor, 1 otherwise. For exploratory factor analysis with lasso penalty, indic should be imposed certain constraints on the a $K \times K$ sub-matrix to ensure identifiability. The remaining parts do not assume any pre-specified zero structure but instead, the appropriate lasso penalty would recover the true zero structure. Also see constrain
samp	a subsample for each iteration; default is 50
forgetrate	the forget rate for the stochastic algorithm. The value should be within the range from 0.5 to 1. Default is 0.51
mu_b	the mean parameter for the prior distribution of item difficulty parameters
sigma2_b	the variance parameter for the prior distribution of item difficulty parameters
Alpha	the α parameter for the prior distribution of guessing parameters
Beta	the β parameter for the prior distribution of guessing parameters
max.iter	the maximum number of iterations for the EM cycle; default is 5000
constrain	the constraint setting: "C1" or "C2". To ensure identifiability, "C1" sets a $K \times K$ sub-matrix of indic to be an identity matrix, and the variance of the population

	variance-covariance matrix to be 1. Instead of setting all off-diagonals of a sub-matrix to be zero, "C2" keeps the sub-matrix to be a triangular matrix with the diagonal being ones. That is, there are test items associated with each factor for sure and they may be associated with other factors as well. Nonzero entries except for the diagonal entries of the sub-matrix are penalized during the estimation procedure. Although, "C2" is much weaker than "C1", it still ensures empirical identifiability. Default is "C1"
non_pen	the index of an item which is associated with each factor to satisfy "C2". For C1, the input can be NULL
gamma	a numerical value of adaptive lasso parameter. Zou (2006) recommended three values, 0.5, 1, and 2. The default value is 2.

Value

a list containing the following objects:

ra	item discrimination parameters, a $J \times K$ matrix
rb	item difficulty parameters, vector of length J
rc	item guessing parameters, vector of length J
rs	variational parameters s , a $N \times J$ matrix
reta	variational parameters $\eta(\xi)$, a $N \times J$ matrix
reps	variational parameters ξ , a $N \times J$ matrix
rsigma	population variance-covariance matrix, a $K \times K$ matrix
mu_i	mean parameter for each person, a $K \times N$ matrix
sig_i	covariance matrix for each person, a $K \times K \times N$ array
n	the number of iterations for the EM cycle
Q_mat	factor loading structure, a $J \times K$ matrix
GIC	model fit index
AIC	model fit index
BIC	model fit index
lbd	numerical value of lasso penalty parameter λ

References

- Cho, A. E., Xiao, J., Wang, C., & Xu, G. (2022). Regularized Variational Estimation for Exploratory Item Factor Analysis. *Psychometrika*. <https://doi.org/10.1007/s11336-022-09874-6>
- Zou, H. (2006). The adaptive LASSO and its oracle properties. *Journal of the American Statistical Association*, 7, 1011418–1429.

See Also

[sgvem_3PLEFA_rot](#), [sgvem_3PLEFA_lasso](#), [exampleIndic_efa3pl_c1](#), [exampleIndic_efa3pl_c2](#)

Examples

```
sgvem_3PLEFA_adaptlasso(exampleData_3pl, exampleIndic_efa3pl_c1, samp=50, forgetrate=0.51, mu_b=0, sigma2_b=4, A
sgvem_3PLEFA_adaptlasso(exampleData_3pl, exampleIndic_efa3pl_c2, samp=50, forgetrate=0.51, mu_b=0, sigma2_b=4, A
```

sgvem_3PLEFA_lasso *Stochastic GVEM with Lasso Penalty for Exploratory M3PL Analysis*

Description

Stochastic GVEM with Lasso Penalty for Exploratory M3PL Analysis

Usage

```
sgvem_3PLEFA_lasso(
  u,
  indic,
  samp = 50,
  forgetrate = 0.51,
  mu_b,
  sigma2_b,
  Alpha,
  Beta,
  max.iter = 5000,
  constrain = "C1",
  non_pen = NULL
)
```

Arguments

u	a $N \times J$ matrix or a data.frame that consists of binary responses of N individuals to J items. The missing values are coded as NA
indic	a $J \times K$ matrix or a data.frame that describes the factor loading structure of J items to K factors. It consists of binary values where 0 refers to the item is irrelevant with this factor, 1 otherwise. For exploratory factor analysis with lasso penalty, indic should be imposed certain constraints on the a $K \times K$ sub-matrix to ensure identifiability. The remaining parts do not assume any pre-specified zero structure but instead, the appropriate lasso penalty would recover the true zero structure. Also see constrain
samp	a subsample for each iteration; default is 50
forgetrate	the forget rate for the stochastic algorithm. The value should be within the range from 0.5 to 1. Default is 0.51
mu_b	the mean parameter for the prior distribution of item difficulty parameters
sigma2_b	the variance parameter for the prior distribution of item difficulty parameters
Alpha	the α parameter for the prior distribution of guessing parameters
Beta	the β parameter for the prior distribution of guessing parameters
max.iter	the maximum number of iterations for the EM cycle; default is 5000
constrain	the constraint setting: "C1" or "C2". To ensure identifiability, "C1" sets a $K \times K$ sub-matrix of indic to be an identity matrix, and the variance of the population variance-covariance matrix to be 1. Instead of setting all off-diagonals of a sub-matrix to be zero, "C2" keeps the sub-matrix to be a triangular matrix with the diagonal being ones. That is, there are test items associated with each factor for sure and they may be associated with other factors as well. Nonzero entries

	except for the diagonal entries of the sub-matrix are penalized during the estimation procedure. Although, "C2" is much weaker than "C1", it still ensures empirical identifiability. Default is "C1"
non_pen	the index of an item which is associated with each factor to satisfy "C2". For C1, the input can be NULL

Value

a list containing the following objects:

ra	item discrimination parameters, a $J \times K$ matrix
rb	item difficulty parameters, vector of length J
rc	item guessing parameters, vector of length J
rs	variational parameters s , a $N \times J$ matrix
reta	variational parameters $\eta(\xi)$, a $N \times J$ matrix
reps	variational parameters ξ , a $N \times J$ matrix
rsigma	population variance-covariance matrix, a $K \times K$ matrix
mu_i	mean parameter for each person, a $K \times N$ matrix
sig_i	covariance matrix for each person, a $K \times K \times N$ array
n	the number of iterations for the EM cycle
Q_mat	factor loading structure, a $J \times K$ matrix
GIC	model fit index
AIC	model fit index
BIC	model fit index
lbd	numerical value of lasso penalty parameter λ

References

Cho, A. E., Xiao, J., Wang, C., & Xu, G. (2022). Regularized Variational Estimation for Exploratory Item Factor Analysis. *Psychometrika*. <https://doi.org/10.1007/s11336-022-09874-6>

See Also

[sgvem_3PLEFA_rot](#), [sgvem_3PLEFA_adaptlasso](#), [exampleIndic_efa3pl_c1](#), [exampleIndic_efa3pl_c2](#)

Examples

```
sgvem_3PLEFA_lasso(exampleData_3pl, exampleIndic_efa3pl_c1, samp=50, forgetrate=0.51, mu_b=0, sigma2_b=4, Alpha=
sgvem_3PLEFA_lasso(exampleData_3pl, exampleIndic_efa3pl_c2, samp=50, forgetrate=0.51, mu_b=0, sigma2_b=4, Alpha=
```


Description

Stochastic GVEM for Exploratory M3PL Analysis

Usage

```
sgvem_3PLEFA_rot(
  u,
  domain,
  samp = 50,
  forgetrate = 0.51,
  mu_b,
  sigma2_b,
  Alpha,
  Beta,
  max.iter = 5000,
  rot = "Promax"
)
```

Arguments

<code>u</code>	a $N \times J$ matrix or a data.frame that consists of binary responses of N individuals to J items. The missing values are coded as NA
<code>domain</code>	the number of factors
<code>samp</code>	a subsample for each iteration; default is 50
<code>forgetrate</code>	the forget rate for the stochastic algorithm. The value should be within the range from 0.5 to 1. Default is 0.51
<code>mu_b</code>	the mean parameter for the prior distribution of item difficulty parameters
<code>sigma2_b</code>	the variance parameter for the prior distribution of item difficulty parameters
<code>Alpha</code>	the α parameter for the prior distribution of guessing parameters
<code>Beta</code>	the β parameter for the prior distribution of guessing parameters
<code>max.iter</code>	the maximum number of iterations for the EM cycle; default is 5000
<code>rot</code>	the post-hoc rotation method: Promax or CF-Quartimax; default is "Promax", but may also be "cfQ" for conducting the CF-Quartimax rotation

Value

a list containing the following objects:

<code>ra</code>	item discrimination parameters, a $J \times K$ matrix
<code>rb</code>	item difficulty parameters, vector of length J
<code>rc</code>	item guessing parameters, vector of length J
<code>rs</code>	variational parameters s , a $N \times J$ matrix
<code>reta</code>	variational parameters $\eta(\xi)$, a $N \times J$ matrix

reps	variational parameters ξ , a $N \times J$ matrix
rsigma	population variance-covariance matrix, a $K \times K$ matrix
mu_i	mean parameter for each person, a $K \times N$ matrix
sig_i	covariance matrix for each person, a $K \times K \times N$ array
n	the number of iterations for the EM cycle
Q_mat	factor loading structure, a $J \times K$ matrix
rk	factor loadings, a $J \times K$ matrix
GIC	model fit index
AIC	model fit index
BIC	model fit index
ur_a	item discrimination parameters before conducting the rotation, a $J \times K$ matrix

See Also

[sgvem_3PLEFA_lasso](#), [sgvem_3PLEFA_adaptlasso](#)

Examples

```
sgvem_3PLEFA_rot(exampleData_3pl, 3, samp=50, forgetrate=0.51, mu_b=0, sigma2_b=4, Alpha=10, Beta=40, max.iter=500)
```

Index

* datasets

- exampleData_2pl, [3](#)
- exampleData_3pl, [3](#)
- exampleIndic_cfa2pl, [3](#)
- exampleIndic_cfa3pl, [4](#)
- exampleIndic_efa2pl_c1, [4](#)
- exampleIndic_efa2pl_c2, [4](#)
- exampleIndic_efa3pl_c1, [5](#)
- exampleIndic_efa3pl_c2, [5](#)

* package

- VEMIRT-package, [2](#)

- exampleData_2pl, [3](#)
- exampleData_3pl, [3](#)
- exampleIndic_cfa2pl, [3](#)
- exampleIndic_cfa3pl, [4](#)
- exampleIndic_efa2pl_c1, [4](#), [8](#), [9](#)
- exampleIndic_efa2pl_c2, [4](#), [8](#), [9](#)
- exampleIndic_efa3pl_c1, [5](#), [14](#), [16](#)
- exampleIndic_efa3pl_c2, [5](#), [14](#), [16](#)

- gvem_2PLCFA, [2](#), [6](#), [12](#)
- gvem_2PLEFA_adaptlasso, [7](#), [9](#), [10](#)
- gvem_2PLEFA_lasso, [8](#), [8](#), [10](#)
- gvem_2PLEFA_rot, [8](#), [9](#), [10](#)

- pa_poly, [2](#), [11](#)

- sgvem_3PLCFA, [2](#), [6](#), [11](#)
- sgvem_3PLEFA_adaptlasso, [13](#), [16](#), [18](#)
- sgvem_3PLEFA_lasso, [14](#), [15](#), [18](#)
- sgvem_3PLEFA_rot, [14](#), [16](#), [17](#)

- VEMIRT (VEMIRT-package), [2](#)
- VEMIRT-package, [2](#)