Package ‘lvmcomp’

December 31, 2018

Type Package

Title Stochastic EM Algorithms for Latent Variable Models with a High-Dimensional Latent Space

Version 1.2

Date 2018-12-19

Author Siliang Zhang [aut, cre], Yunxiao Chen [aut], Jorge Nocedal [cph], Naoaki Okazaki [cph]

Description Provides stochastic EM algorithms for latent variable models with a high-dimensional latent space. So far, we provide functions for confirmatory item factor analysis based on the multidimensional two parameter logistic (M2PL) model and the generalized multidimensional partial credit model. These functions scale well for problems with many latent traits (e.g., thirty or even more) and are virtually tuning-free. The computation is facilitated by multiprocessing 'OpenMP' API.

For more information, please refer to:

Maintainer Siliang Zhang <zhangsiliang123@gmail.com>

URL https://github.com/slzhang-fd/lvmcomp

BugReports https://github.com/slzhang-fd/lvmcomp/issues

License GPL-3

LazyData true

Depends R (>= 3.1)

Imports Rcpp (>= 0.12.17), coda (>= 0.19-1), stats

LinkingTo Rcpp, RcppArmadillo

RoxygenNote 6.1.0

Encoding UTF-8

NeedsCompilation yes

Repository CRAN

Date/Publication 2018-12-30 23:00:02 UTC
**R topics documented:**

- `data_sim_mirt` ........................................... 2
- `data_sim_pcirt` ........................................... 2
- `StEM_mirt` ............................................... 3
- `StEM_pcirt` ............................................... 4

## `data_sim_mirt`

**Simulated dataset for multivariate item response theory model.**

### Description

The dataset contains the simulation setting and the response data.

### Usage

`data_sim_mirt`

### Format

An object of class `list` of length 9.

## `data_sim_pcirt`

**Simulated dataset for generalized partial credit model.**

### Description

The dataset contains the simulation setting and the response data.

### Usage

`data_sim_pcirt`

### Format

An object of class `list` of length 10.
StEM_mirt

Stochastic EM algorithm for solving multivariate item response theory model

Description

Stochastic EM algorithm for solving multivariate item response theory model

Usage

\texttt{StEM\_mirt(response, Q, A0, d0, theta0, sigma0, m = 200, TT = 20, max\_attempt = 40, tol = 1.5, precision = 0.01, parallel = FALSE)}

Arguments

- \texttt{response} N by J matrix containing 0/1 responses, where N is the number of respondents and J is the number of items.
- \texttt{Q} J by K matrix containing 0/1 entries, where J is the number of items and K is the number of latent traits. Each entry indicates whether an item measures a certain latent trait.
- \texttt{A0} J by K matrix, the initial value of loading matrix, satisfying the constraints given by Q.
- \texttt{d0} Length J vector, the initial value of intercept parameters.
- \texttt{theta0} N by K matrix, the initial value of latent traits for each respondent.
- \texttt{sigma0} K by K matrix, the initial value of correlations among the latent traits.
- \texttt{m} The length of Markov chain window for choosing burn-in size with a default value 200.
- \texttt{TT} The batch size with a default value 20.
- \texttt{max\_attempt} The maximum number of batches before stopping.
- \texttt{tol} The tolerance of geweke statistic used for determining burn-in size with a default value 1.5.
- \texttt{precision} The precision value for determining the stopping of the algorithm with a default value 1e-2.
- \texttt{parallel} Whether or not enable the parallel computing with a default value \texttt{FALSE}.

Value

The function returns a list with the following components:

- \texttt{A\_hat} The estimated loading matrix.
- \texttt{d\_hat} The estimated value of intercept parameters.
- \texttt{sigma\_hat} The estimated value of correlation matrix of latent traits.
- \texttt{burn\_in\_T} The length of burn-in size.
References


Examples

# run a toy example based on the M2PL model

# load a simulated dataset
attach(data_sim_mirt)

# generate starting values for the algorithm
A0 <- Q
d0 <- rep(0, J)
theta0 <- matrix(rnorm(N*K, 0, 1), N)
sigma0 <- diag(1, K)

# do the confirmatory MIRT analysis
# to enable multicore processing, set parallel = T
mirt_res <- StEM_mirt(response, Q, A0, d0, theta0, sigma0)

StEM_pcirt

Stochastic EM algorithm for solving generalized partial credit model

Description

Stochastic EM algorithm for solving generalized partial credit model

Usage

StEM_pcirt(response, Q, A0, D0, theta0, sigma0, m = 200, TT = 20,
max_attempt = 40, tol = 1.5, precision = 0.015, parallel = F)

Arguments

response N by J matrix containing 0,1,...,M-1 responses, where N is the number of respondents and J is the number of items.

Q J by K matrix containing 0/1 entries, where J is the number of items and K is the number of latent traits. Each entry indicates whether an item measures a certain latent trait.

A0 J by K matrix, the initial value of loading matrix.

D0 J by M matrix containing the initial value of intercept parameters, where M is the number of response categories.

theta0 N by K matrix, the initial value of latent traits for each respondent
StEM_pcirt

sigma0  K by K matrix, the initial value of correlations among latent traits.
m  The length of Markov chain window for choosing burn-in size with a default value 200.
T  The batch size with a default value 20.
max_attempt  The maximum attempt times if the precision criterion is not meet.
tol  The tolerance of geweke statistic used for determining burn in size, default value is 1.5.
precision  The pre-set precision value for determining the length of Markov chain, default value is 0.015.
parallel  Whether or not enable the parallel computing with a default value FALSE.

Value
The function returns a list with the following components:

A_hat  The estimated loading matrix
D_hat  The estimated value of intercept parameters.
sigma_hat  The estimated value of correlation matrix of latent traits.
burn_in_T  The length of burn in size.

References

Examples

# run a toy example based on the partial credit model

# load a simulated dataset
attach(data_sim_pcirt)

# generate starting values for the algorithm
A0 <- Q
D0 <- matrix(1, J, M)
D0[,1] <- 0
theta0 <- matrix(rnorm(N*K), N, K)
sigma0 <- diag(1, K)

# do the confirmatory partial credit model analysis
# to enable multicore processing, set parallel = T
pcirt_res <- StEM_pcirt(response, Q, A0, D0, theta0, sigma0)
Index

*Topic datasets
    data_sim_mirt, 2
    data_sim_pcirt, 2

data_sim_mirt, 2
data_sim_pcirt, 2

StEM_mirt, 3
StEM_pcirt, 4