

# Package ‘LSMjml’

May 7, 2026

**Type** Package

**Title** Fitting Latent Space Item Response Models using Joint Maximum Likelihood Estimation

**Version** 0.6.0

**Author** Dylan Molenaar [aut, cre]

**Maintainer** Dylan Molenaar <d.molenaar@uva.nl>

**Description** In Latent Space Item Response Models, subjects and items are embedded in a multidimensional Euclidean latent space.

As such, interactions among persons, items, and person-item combinations can be revealed that are unmodelled in more conventional item response theory models. This package implements the methods from Molenaar & Jeon (in press) and can be used to fit Latent Space Item Response Models to data using joint maximum likelihood estimation. The package can handle binary data, ordinal data, and data with mixed scales.

The package incorporates facilities for data simulation, rotation of the latent space, and K-fold cross-validation to select the number of dimensions of the latent space.

**License** GPL-3

**Encoding** UTF-8

**Imports** Rcpp (>= 1.0.12), lavaan, pROC, psych

**LinkingTo** Rcpp, RcppArmadillo

**NeedsCompilation** yes

**Repository** CRAN

**Date/Publication** 2025-12-19 15:10:08 UTC

## Contents

LSMfit . . . . .	2
LSMrotate . . . . .	6
LSMselect . . . . .	8
LSMsim . . . . .	10

<b>Index</b>	<b>13</b>
--------------	-----------

LSMfit

*Fitting Latent Space Item Response Models using Joint Maximum Likelihood Estimation*

## Description

This function fits a Latent Space Item Response Model (LSIRM) with an R dimensional latent space using penalized Joint Maximum Likelihood (pJML) or constrained Joint Maximum Likelihood (cJML) to observed binary or ordinal item scores.

## Usage

```
LSMfit(X, ndim_z, penalty=NULL, C=NULL, starts=NULL,
       tol=.1e-2, silent=FALSE)
```

## Arguments

X	A matrix of size N by n containing the binary or ordinal item scores, where N is the number of subjects and n is the number of items. The number of item score categories can be different across items as long as the lowest score is coded 0 for all items. NA's are allowed.
ndim_z	Number of dimensions of the latent space, R.
penalty	The weight for the L2 penalty of pJML. If both penalty and C (see below) are NULL (the default), a pJML is used with a weight of 1 (i.e., standard normal prior on all parameters).
C	The maximum size of the norm of the item parameter vectors. The resulting maximum norm for the person parameter vectors is $1/2 * C$ .
starts	Either a list containing starting values for the model parameters (see <b>details</b> ) or a character string indicating the method of starting value calculation. The options are: <p><b>"wls"</b> the starting values are determined from fitting a R+1 dimensional item factor model to the polychoric correlation matrix of X using weighted least squares</p> <p><b>"ml"</b> the starting values are determined from fitting a R+1 dimensional linear factor model to the polychoric correlation matrix of X using normal theory maximum likelihood (default)</p> <p><b>"random"</b> starting values are drawn from normal distributions.</p>
tol	Convergence criterion: Iterations stop if the difference in loglikelihoods between two subsequent iterations is smaller than this number. Default is .001.
silent	Logical. If FALSE, iterations details are printed to the screen during estimation.

## Details

LSMfit optimizes the joint likelihood function of the LSIRM described in Molenaar and Jeon (in press) using a variant of the alternating optimization algorithm by Chen et al. (2019) and using either a L2 regularization penalty similar to Bergner et al. (2022) or a constraint on the norms of the parameter vectors similar to Chen et al. (2019). For binary  $X_{pi}$ , the LSIRM by Molenaar and Jeon is given by:

$$\text{logit}(E(X_{pi})) = \theta_p + b_i - (\sum_{r=1}^R (z_{pr} - w_{ir})^2)^{1/2}$$

where  $\theta_p$  is a person intercept,  $b_i$  is an item intercept,  $R$  denotes the dimension of the latent space, and  $z_{pr}$  and  $w_{ir}$  are respectively the person and item coordinates in the latent space. The matrix  $\mathbf{W}$  containing the  $w_{ir}$  parameters is constrained to an echelon structure (i.e., all elements from the upper triangle of submatrix  $\mathbf{W}_{1:(R-1),1:(R-1)}$  are fixed to 0. Next, cJML estimation involves constraining the Euclidean norm of person parameter vector  $\tau_{1p} = [\theta_p, z_{p1}, z_{p2}, \dots, z_{pK}]$  to be equal to C, and the Euclidean norm of the item parameter vector  $\tau_{2i} = [b_i, w_{i1}, w_{i2}, \dots, w_{iK}]$  to be equal to  $1/2 * C$ . On the contrary, pJML estimation involves adding an L2 regularization penalty for all parameters to the joint likelihood function in such a way that the penalty parameter can be interpreted as the precision of a 0-centered normal prior on the parameters.

Using the starts argument, starting values can be provided in a list containing entries:

`z0` a N by R matrix with starting values for  $z_{pr}$

`w0` a n by R matrix with starting values for  $w_{ir}$

`b0` a n by 1 matrix with starting values for  $b_i$

`theta0` a N by 1 matrix with starting values for  $\theta_p$

Alternatively, starting values can be automatically determined by LSMfit. To this end, the following R+1 factor model will be fit (omitting the item intercept):

$$g(E(X_{pi})) = \eta_{p0} + \sum_{r=1}^R \lambda_{ir} \eta_{pr}$$

where the n by R matrix of  $\lambda_{ir}$  parameters follows the echelon structure above, and  $g(\cdot)$  is either the identity link or the probit link (see below). The model above is fit to the polychoric correlation matrix of  $\mathbf{X}$  using either weighted least squares (WLS) estimation with a probit link for  $g(\cdot)$  or normal theory maximum likelihood (ML) estimation with an identity link for  $g(\cdot)$  (i.e., the polychoric correlation matrix is treated as a pmcc matrix). In both cases, the thresholds of the polychoric correlation matrix are taken as the basis for the starting values of  $b_i$ , the factor score estimates of  $\eta_{p0}$  are taken as starts for  $\theta_p$ , the estimates of  $\eta_{pr}$  are taken as the starts for  $z_{pr}$ , and the estimates of  $\lambda_{ir}$  are taken as a basis for the starting values of  $w_{ir}$ . The WLS approach is statistically the most rigorous approach but can be time consuming, while the ML approach is an ad-hoc approach but which is fast and turns out to work well in practice. The ML approach is the default approach to obtain starting values if `starts=NULL`. Especially for models with  $R > 2$  fitting the factor model above may fail. In that case, LSMfit automatically switches to random starts.

Ordinal items are internally accomodated by dummy coding the items with more than 2 score levels into C-1 binary variables using a cummlatrive binary coding scheme (see Molenaar & Jeon, in press). Next, the dummy coded variables are submitted to the binary LSIRM above with the  $w_{ir}$  parameters equated for dummy coded variables that correspond to the same original items. In the resuling model, the estimates of  $b_i$  correspond to the category parameters of a sequential IRT model (Tutz, 1990) which are generally close to those of a graded response IRT model. The number of score levels can be different across items as long as the lowest score is coded 0 for all items

**Value**

An object of class `LSMfit` with values

<code>theta</code>	$\theta_p$ estimates
<code>b</code>	$b_i$ estimates
<code>z</code>	$z_{pr}$ estimates
<code>w</code>	$z_{ir}$ estimates
<code>logL</code>	value of the loglikelihood at convergence
<code>starts</code>	the starting values used
<code>as_starts</code>	a list containing the parameter estimates, suitable to be used as argument for <code>starts</code> in a new run
<code>internal</code>	various matrices used internally

**Author(s)**

Dylan Molenaar <d.molenaar@uva.nl>

**References**

Bergner, Y., Halpin, P., & Vie, J. J. (2022). Multidimensional Item Response Theory in the Style of Collaborative Filtering. *Psychometrika*, **87**(1), 266-288. <https://doi.org/10.1007/s11336-021-09788-9>

Chen, Y., Li, X., & Zhang, S. (2019). Joint maximum likelihood estimation for high-dimensional exploratory item factor analysis. *Psychometrika*, **84**(1), 124-146. <https://doi.org/10.1007/s11336-018-9646-5>

Molenaar, D., & Jeon, M.J. (in press). Joint maximum likelihood estimation of latent space item response models. *Psychometrika*.

Tutz, G. (1990). Sequential item response models with an ordered response. *British Journal of Mathematical and Statistical Psychology*, **43**(1), 39-55. <https://doi.org/10.1111/j.2044-8317.1990.tb00925.x>

**See Also**

[LSMselect](#) for selecting the number of latent space dimensions using cross-validation. [LSMsim](#) for simulating data according to the LSIRM. [LSMrotate](#) for rotating item and person coordinates.

**Examples**

```
#
# only binary items
#

# data sim with 1000 subjects and 20 binary items
# according to 2 dimensional latent space model (R=2)
set.seed(1111)
N=1000
nit=20
ndim_z=2
```

```

dat_obj=LSMsim(N,nit,ndim_z)
X=dat_obj$X
zt=dat_obj$par$zt      # rotated true z, see ?LSMsim and ?LSMrotate
wt=dat_obj$par$wt      # rotated true w

#fit model
results=LSMfit(X,2)

#plot the parameter recovery results
oldpar=par(mfrow=c(2,2))

s_p=sign(cor(results$z,zt))      # to correct for sign switches in the plots
s_i=sign(cor(results$w,wt))

plot(s_p[1,1]*zt[,1],results$z[,1]); abline(0,1)
plot(s_p[2,2]*zt[,2],results$z[,2]); abline(0,1)
plot(s_i[1,1]*wt[,1],results$w[,1]); abline(0,1)
plot(s_i[2,2]*wt[,2],results$w[,2]); abline(0,1)

par(oldpar)

#
# mixed scale items
#

# data sim with 1000 subjects and 20 mixed scale items
# according to 2 dimensional latent space model (R=2)
set.seed(1111)
N=1000
nit=20
ndim_z=2
nc=rpois(nit,2)+2      # number of response categories
                        # (between 2 and 7 for this seed)
dat_obj=LSMsim(N,nit,ndim_z,nc=nc)
X=dat_obj$X
zt=dat_obj$par$zt      # rotated true z, see ?LSMsim and ?LSMrotate
wt=dat_obj$par$wt      # rotated true w

#fit model
results=LSMfit(X,2)

#plot the parameter recovery results
oldpar=par(mfrow=c(2,2))

s_p=sign(cor(results$z,zt))      # to correct for sign switches in the plots
s_i=sign(cor(results$w,wt))

plot(s_p[1,1]*zt[,1],results$z[,1]); abline(0,1)
plot(s_p[2,2]*zt[,2],results$z[,2]); abline(0,1)
plot(s_i[1,1]*wt[,1],results$w[,1]); abline(0,1)
plot(s_i[2,2]*wt[,2],results$w[,2]); abline(0,1)

```

```
par(oldpar)
```

---

LSMrotate	<i>Rotate the person and item latent space parameter matrices to an echeleon structure</i>
-----------	--

---

### Description

This function rotates the person and item latent space parameter matrices to an echeleon structure.

### Usage

```
LSMrotate(z, w, method="chol")
```

### Arguments

z	The N by ndim_z matrix of person coordinates $z_{pr}$ to be rotated.
w	The nit by ndim_z matrix of item coordinates $w_{ir}$ to be rotated.
method	Character string, either "stepwise" or "chol"

### Details

[LSMfit](#) constrains the matrix of item coordinates  $w_{ir}$  to an echeleon structure in fitting the LSIRM to data. Therefore, to compare to results to other results (e.g., obtained using MCMC) or to the true values used to generate the data, it is necessary to rotate those other results/values to the same echeleon structure. This rotation can be performed using `LSMrotate`. Following Wansbeek & Meijer (2006), the function uses a Cholesky decomposition (method="chol", the default) which works for an arbitrary number of latent space dimensions R (except 1). For method="stepwise", the function performs the explicit rotation steps by determining the angle of rotation as described in Molenaar and Jeon (submitted). This method is only implemented for 2 or 3 latent space dimensions.

### Value

A list containing

zt	The rotate matrix of $z_{pr}$ parameters.
wt	The rotate matrix of $w_{ir}$ parameters.
rotMat	The rotation matrix.

### Author(s)

Dylan Molenaar <d.molenaar@uva.nl>

## References

- Molenaar, D., & Jeon, M.J. (in press). Joint maximum likelihood estimation of latent space item response models. *Psychometrika*.
- Wansbeek, T. & Meijer, E. (2000). *Measurement Error and Latent Variables in Econometrics*. Amsterdam: North-Holland.

## See Also

[LSMfit](#) to fit LSIRM models.

## Examples

```

set.seed(1111)
N=1000
nit=20
ndim_z=2

#some true values not following the echeleon structure
z=matrix(rnorm(N*ndim_z),N,ndim_z)
w=matrix(rnorm(nit*ndim_z),nit,ndim_z)

# simulate data using these true values
dat_obj=LSMsim(N,nit,ndim_z,z=z,w=w)
X=dat_obj$X

#fit model
results=LSMfit(X,2)

#plot the parameter recovery results using the *unrotated* true values
#spoiler: will look like nothing

oldpar=par(mfrow=c(2,2))

s_p=sign(cor(results$z,z))      # to correct for sign switches in the plots
s_i=sign(cor(results$w,w))

plot(s_p[1,1]*z[,1],results$z[,1]); abline(0,1)
plot(s_p[2,2]*z[,2],results$z[,2]); abline(0,1)
plot(s_i[1,1]*w[,1],results$w[,1]); abline(0,1)
plot(s_i[2,2]*w[,2],results$w[,2]); abline(0,1)

#plot the parameter recovery results using the *rotated* true values
#spoiler: will look better

zt=dat_obj$par$zt      # rotated true z, see ?LSMsim and ?LSMrotate
wt=dat_obj$par$wt      # rotated true w

s_p=sign(cor(results$z,zt))      # to correct for sign switches in the plots
s_i=sign(cor(results$w,wt))

plot(s_p[1,1]*zt[,1],results$z[,1]); abline(0,1)

```

```

plot(s_p[2,2]*zt[,2],results$z[,2]); abline(0,1)
plot(s_i[1,1]*wt[,1],results$w[,1]); abline(0,1)
plot(s_i[2,2]*wt[,2],results$w[,2]); abline(0,1)

par(oldpar)

```

---

LSMselect	<i>Selecting the Latent Space Dimensionality using K-fold Cross-Validation</i>
-----------	--

---

### Description

This function perform a K-fold cross validation to select the number of dimensions,  $R$ , of the latent space in the Latent Space Item Response Model (LSIRM). Model performance is evaluated using metrics based on the out-of-sample prediction accuracy, the area under the ROC curve, and the mean squared error.

### Usage

```

LSMselect(X, maxDims=3, n folds=5, penalty=NULL, C=NULL,
          starts=NULL, tol=.1, silent=TRUE)

```

### Arguments

$X$	A matrix of size $N$ by $n$ containing the binary or ordinal item scores, where $N$ is the number of subjects and $n$ is the number of items. The number of item score categories can be different across items as long as the lowest score is coded 0 for all items. NA's are allowed.
maxDims	The maximum number of dimensions $R$ for the latent space to be considered. Should be at least 1 so that at least $R=0$ and $R=1$ are considered.
n folds	The number of folds $K$ .
penalty	The weight for the L2 penalty of pJML. If <code>penalty</code> is NULL (the default), a pJML is used with a weight of 1 (i.e., standard normal prior on all parameters).
C	The maximum size of the norm of the person parameter vectors. Not available for cross-validation yet.
starts	Either a list containing starting values for the model parameters or a character string indicating the method of starting value calculation, see <a href="#">LSMfit</a>
tol	Convergence criterion: Iterations stop if the difference in loglikelihoods between two subsequent iterations is smaller than this number. Default is .1.
silent	Logical. If FALSE, iterations details are printed to the screen during estimation.

## Details

LSMselect assigns the non-missing elements of the  $N$  by  $n$  matrix  $X$  randomly to one of the  $K$  folds making sure that the folds are (close to) equally sized. Then,  $\max\text{Dims}+1$  models (i.e.,  $R=0$ ,  $R=1$ ,  $\dots$ ,  $R=\max\text{Dims}$ ) are fit leaving out the data of the first fold. Next, using the parameter estimates for each of the models, the data in the first fold are predicted. Using these predictions and the actual observations in the fold, the three metrics below are calculated. This scheme is repeated for all folds, so that each fold is held out of estimation once.

The metrics calculated are respectively based on the well known prediction accuracy, area under the ROC curve, and mean squared error. However, the metrics are unnormalized and -for accuracy and the ROC curve- are complements so that for all metrics lower values indicate a better model fit. This results in the following metrics:

**Unnormalized Classification Error (UCE)** The UCE is the number of incorrectly predicted item scores summed over folds

**Unnormalized ROC Error (URE)** The URE is the complement of the area under the ROC curve multiplied by the fold size and summed over folds

**Residual Sum of Squares (RSS)** The RSS is the sum of the squared residuals over folds For more details see Molenaar and Jeon (submitted).

## Value

An object of class LSMselect with values

tot_metrics	The overall metrics (summed over folds)
fold_metrics	A list with separate entries for each metric containing the results for each fold separately

## Author(s)

Dylan Molenaar <d.molenaar@uva.nl>

## References

Molenaar, D., & Jeon, M.J. (in press). Joint maximum likelihood estimation of latent space item response models. *Psychometrika*.

## See Also

[LSMfit](#) for fitting LSIRM models and for details about the model.

## Examples

```
# Toy example: compare between R=0 and R=1 for data that follows one dimensional
# latent space model (R=1) using only 2 folds

set.seed(1111)
N=1000
nit=20
ndim_z=1
```

```
dat_obj=LSMsim(N,nit,ndim_z)
X=dat_obj$X
LSMselect(X,1,nfolds=2)
```

LSMsim

*Simulating Data according to the Latent Space Item Response Model***Description**

This function simulates data according to the Latent Space Item Response Model (LSIRM) with an R dimensional latent space and binary and/or ordinal item scores.

**Usage**

```
LSMsim(N, nit, ndim_z, nc=NULL, theta=NULL, b=NULL, z=NULL, w=NULL, gamma=NULL)
```

**Arguments**

N	Sample size
nit	Number of items
ndim_z	Number of dimensions of the latent space, R.
nc	Vector of length nit containing the number of response categories for each item. If NULL (the default) all items are simulated to be binary
theta	N-dimensional vector of true person intercepts $\theta_p$ , if NULL these are drawn from a standard normal distribution
b	nit-dimensional vector of true item intercepts $b_p$ , if NULL these are drawn from a uniform distribution
z	N by ndim_z matrix of true latent space person coordinates $z_{pr}$ , if NULL these are drawn from a standard normal distribution
w	nit by ndim_z matrix of true latent space item coordinates $w_{ir}$ , if NULL these are drawn from a standard normal distribution
gamma	a weight parameter for the Euclidean distances (see <b>details</b> ), if NULL gamma=1

**Details**

Data is simulated according to the original LSIRM by Jeon et al. (2021):

$$\text{logit}(E(X_{pi})) = \theta_p + b_i - \gamma(\sum_{r=1}^R (z_{pr} - w_{ir})^2)^{1/2}$$

In **LSMfit**,  $\gamma$  is fixed to one as it is not identified in a joint maximum likelihood framework (see Molenaar & Jeon, submitted). However, for data simulation,  $\gamma$  can be used to change the strength of the effect of  $z_{pr}$  and  $w_{ir}$ .

**Value**

An list containing:

X	the simulated data
par	a list containing the true parameter values, including wt and zt, the rotated matrices of $w_{ir}$ and $z_{pr}$ parameters.

**Author(s)**

Dylan Molenaar <d.molenaar@uva.nl>

**References**

Jeon, M., Jin, I. H., Schweinberger, M., & Baugh, S. (2021). Mapping unobserved item–respondent interactions: A latent space item response model with interaction map. *Psychometrika*, 86(2), 378-403. doi:10.1007/s11336-021-09762-5

Molenaar, D., & Jeon, M.J. (in press). Joint maximum likelihood estimation of latent space item response models. *Psychometrika*.

**See Also**

[LSMfit](#) for fitting LSIRM models using joint maximum likelihood.

**Examples**

```
# data sim with 1000 subjects and 20 items according to 2 dimensional
# latent space model (R=2) with both binary and ordinal items
set.seed(1111)
N=1000
nit=20
ndim_z=2
nc=sample(c(2,3,5),nit,replace=TRUE) # mix of 2, 3, and 5 point scales
dat_obj=LSMsim(N,nit,ndim_z,nc=nc)
X=dat_obj$X
zt=dat_obj$par$zt # rotated z
wt=dat_obj$par$wt # rotated w

#fit model
results=LSMfit(X,2)

#plot the parameter recovery results
oldpar=par(mfrow=c(2,2))

s_p=sign(cor(results$z,zt)) # to correct for sign switches in the plots
s_i=sign(cor(results$w,wt))

plot(s_p[1,1]*zt[,1],results$z[,1]); abline(0,1)
plot(s_p[2,2]*zt[,2],results$z[,2]); abline(0,1)
plot(s_i[1,1]*wt[,1],results$w[,1]); abline(0,1)
plot(s_i[2,2]*wt[,2],results$w[,2]); abline(0,1)
```

par(oldpar)

# Index

\* **models**

LSMfit, 2

LSMrotate, 6

LSMselect, 8

LSMsim, 10

\* **multivariate**

LSMfit, 2

LSMrotate, 6

LSMselect, 8

LSMsim, 10

LSMfit, 2, 6–11

LSMrotate, 4, 6

LSMselect, 4, 8

LSMsim, 4, 10