

# Package ‘harmonious’

July 13, 2024

**Type** Package

**Title** Explanatory Item Response modeling in CmdStanR

**Version** 0.1.0

**Author** Nathan DePuy

**Maintainer** Nathan DePuy <depy@uiowa.edu>

**Description** A package for performing Bayesian inference on Explanatory Item Repsonse models in CmdStanR

**License** MIT + file LICENSE

**Encoding** UTF-8

**LazyData** true

**Depends** bayesplot,  
Cairo,  
ggplot2,  
pander,  
mathjaxr,  
loo,  
cmdstanr

**Suggests** testthat (>= 3.0.0)

**Config/testthat/edition** 3

**RoxygenNote** 7.3.2

**Roxygen** list(markdown=TRUE)

**RdMacros** mathjaxr

## Contents

.onLoad . . . . .	2
CreateMod . . . . .	2
genData . . . . .	3
genReport . . . . .	4
hello . . . . .	5
initialize . . . . .	5
rCorr . . . . .	6
rhatCheck . . . . .	6
sample . . . . .	7
<b>Index</b>	<b>8</b>

---

<code>.onLoad</code>	<i>Load/Install the cmdstanr package</i>
----------------------	--

---

### Description

Validates installation of the cmdstanr package and, if not found, prompts for its installation from the recommended source

### Usage

```
.onLoad(...)
```

### Details

See <https://mc-stan.org/cmdstanr/>

---

CreateMod	<i>Create Adjunct CmdStan Model Environment</i>
-----------	---

---

### Description

Creates an environment to estimate the  $p \times i$  interaction model with CmdStan.

### Usage

```
CreateMod(
  initFile,
  runFile,
  coef_hyper,
  sd_hyper,
  nWarmup_init,
  nSamples_init,
  nWarmup_run,
  nSamples_run,
  aux_envir
)
```

### Arguments

<code>initFile</code>	Path to .stan file with fixed $\theta$ model configuration
<code>runFile</code>	Path to .stan file with the free $\theta$ model configuration
<code>coef_hyper</code>	Hyperparameter value for the standard deviation of normally distributed parameters
<code>sd_hyper</code>	Hyperparameter value for the shape parameter of gamma distributed parameters
<code>nWarmup_init</code>	Number of burn-in draws for the fixed $\theta$ model
<code>nWarmup_run</code>	Number of burn-in draws for the free $\theta$ model
<code>nSamples_run</code>	Number of sampled posterior values for the free $\theta$ model (after burn-in has completed)

aux_envir	An environment object that contains objects to be loaded into the CreateMod environment (such as is returned by the <a href="#">genData</a> function)
nSampels_init	Number of posterior draws for the fixed $\theta$ model (after burn-in has completed)

### Details

Creates an environment with methods for fitting the fixed  $\theta$  and free  $\theta$  models. Additionally, CreateMod includes a method for recursively checking  $\hat{R}$  convergence.

### Value

an environment object containing fixed  $\theta$  and free  $\theta$  model results

### See Also

[initialize](#), [sample](#), [rhatCheck](#)

---

genData	<i>Data Generating Function for the <math>p \times i</math> Interaction Model</i>
---------	---

---

### Description

Generates data for a  $p \times i$  Interaction Model under the Explanatory Item Response theoretical framework.

### Usage

```
genData(P, I, J, K, seed = NULL, isCorrI = TRUE)
```

### Arguments

P	Number of examinees
I	Number of items
J	Number of potential item feature categories
K	Number of potential person feature categories
seed	Integer seed for replication (if NULL, a random seed will be generated)

### Details

Let a model of the log-odds transformed outcome ( $\eta_{pi}$ ) be defined as:

$$\eta_{pi} = \theta_p \lambda_i + \tau_i + \sum_{k=1}^K \sum_{j=1}^J \beta_{jk(\eta)} x_{ik} z_{pj}$$

where the linear predictor of latent trait measurements  $\theta_p$  is defined as:

$$\theta_p = \sum_{j=1}^J \beta_{j(\theta)} z_{pj} + u_{p(\theta)}$$

where the linear predictor of item slope/discrimination terms ( $\lambda_i$ ) is defined as:

$$\lambda_i = \sum_{k=1}^K \beta_{k(\lambda)} x_{ik} + u_{i(\lambda)}$$

and where the linear predictor of item intercept/easiness terms ( $\tau_i$ ) is defined as:

$$\tau_i = \sum_{k=1}^K \beta_{k(\tau)} x_{ik} + u_{i(\tau)}$$

For the linear predictor of latent trait measurements, the following assumption is made on its residual terms for the purposes of model identification:

$$u_{p(\theta)} \sim \mathcal{N}(0, 1)$$

If an assumption is made that there is no structured dependency between the residual terms for the linear predictors of item intercepts and item slopes (`isCorrI = FALSE`), then:

$$u_{i(\lambda)} \sim \mathcal{N}(0, \sigma_\lambda^2)$$

$$u_{i(\tau)} \sim \mathcal{N}(0, \sigma_\tau^2)$$

Whereas if it is assumed that the residuals of the linear predictors of item intercepts and item slopes is assumed to have a predictable underlying structure (`isCorrI = TRUE`), then:

$$\begin{bmatrix} u_{i(\lambda)} \\ u_{i(\tau)} \end{bmatrix} \sim \mathcal{MVN}\left(\mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \Sigma = \begin{bmatrix} \sigma_\lambda^2 & \sigma_\lambda \sigma_\tau \\ \sigma_\tau \sigma_\lambda & \sigma_\tau^2 \end{bmatrix}\right)$$

## Value

an environment containing simulated data

---

genReport

*Generate Text Report of Free  $\theta$  Model Results*

---

## Description

Helper function that generates a text document in the user-specified save directory

## Usage

```
genReport(saveDir, fileDetails, ...)
```

## Arguments

saveDir	Path to directory where report is saved
fileDetails	string identifier for name of saved report
...	additional arguments passed from parent frame

## Details

Model results include posterior descriptives and simulated data information

## Value

a .txt file of the free  $\theta$  results

---

hello	<i>Hello, World!</i>
-------	----------------------

---

**Description**

Prints 'Hello, world!'.

**Usage**

```
hello()
```

**Examples**

```
hello()
```

---

initialize	<i>Run Fixed <math>\theta</math> Model</i>
------------	--

---

**Description**

Runs the first step in the two-step estimation framework

**Usage**

```
initialize(...)
```

**Arguments**

... arguments passed from parent frame

**Details**

The first step in the two-step estimation framework for the  $p \times i$  model

**Value**

Model results for the standardized  $\theta$  model added to the environment rendered by [CreateMod](#)

**See Also**

[CreateMod](#)

rCorr

*Generation of Data for the Person-by-Item Interaction Model***Description**

randomly samples a lower triangular, Cholesky-factorized matrix  $L$  from a Lewandowski-Kurowicka-Joe (LKJ) distribution given  $\eta$ .

**Usage**

```
rCorr(nDim, eta = 1)
```

**Arguments**

nDim	desired dimension of the sampled $n \times n$ matrix
eta	concentration hyperparameter

**Details**

Using the onion method, samples a Cholesky factorized lower-triangular matrix  $L$  from a Lewandowski-Kurowicka-Joe (LKJ) distribution given the concentration hyperparameter  $\eta$ , such that:

$$A = LL^\top$$

where  $A$  is a  $n \times n$  matrix and is a member of the set of all symmetric, positive-definite matrices.

**Value**

a Choleksy-factorized, lower triangular matrix

rhatCheck

*Recursive Validation for Rhat Convergence***Description**

Helper function that validates  $\hat{R}$  convergence.

**Usage**

```
rhatCheck(...)
```

**Arguments**

...	arguments passed from parent frame
-----	------------------------------------

Details

```
Input: CmdStan model,  $N_{\max}$ ,  $\hat{R}_1$ 
Sample CmdStan model
while  $\hat{R}_n \leq \hat{R}_{thresh}$  or  $n \leq N_{\max}$  do
  Sample free  $\theta$  model
   $n += 1$ 
end do
end while
```

Value

modified free  $\theta$  model that is added to an environment created by [CreateMod](#)

See Also

[CreateMod](#)

---

sample	<i>Run Free <math>\theta</math> Model</i>
--------	---

---

Description

Runs the second step of the two-step estimation framework

Usage

```
sample(...)
```

Arguments

... arguments passed from parent frame

Details

The second step treating latent trait measurements ( $\theta_p$ ) as freely estimated parameters in the two-step estimation framework for the  $p \times i$  model

Value

Model results for the full model added to the environment rendered by [CreateMod](#)

See Also

[CreateMod](#)

# Index

.onLoad, [2](#)

CreateMod, [2](#), [5](#), [7](#)

genData, [3](#), [3](#)

genReport, [4](#)

hello, [5](#)

initialize, [3](#), [5](#)

rCorr, [6](#)

rhatCheck, [3](#), [6](#)

sample, [3](#), [7](#)