

# Package: flps (via r-universe)

October 18, 2024

**Type** Package

**Title** Fully-Latent Principal Stratification

**Version** 1.1.0

**Date** 2024-03-27

**Description** Simulation and analysis of Fully-Latent Principal Stratification (FLPS) with measurement models. Lee, Adam, Kang, & Whittaker (2023). <[doi:10.1007/978-3-031-27781-8\\_25](https://doi.org/10.1007/978-3-031-27781-8_25)>. This package is supported by the Institute of Education Sciences, U.S. Department of Education, through Grant R305D210036.

**License** GPL-3

**Depends** R (>= 3.5.0)

**Imports** rstan, Rcpp (>= 1.0.8.3), methods, mirt, MASS, utils, stats, mvtnorm, ggplot2, glue

**Suggests** testthat (>= 3.0.0), knitr, rmarkdown, lavaan, data.table

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.3.1

**VignetteBuilder** knitr

**Collate** 'flps-package.r' 'load\_rstan.r' 'make\_FLPSdata.r' 'run\_FLPS.r' 'priors.r' 'sim\_gen\_measurement.r' 'sim\_gen\_structure.r' 'sim\_gen\_standata.r' 'sim\_make\_data.r' 'sim\_utils.r' 'z\_utils.r' 'z\_validate.r' 'plot\_function.r' 'summary\_function.r' 'data\_desc.r' 'zzz.r'

**URL** <https://sooyong1.github.io/flps/>

**Repository** <https://sooyong1.r-universe.dev>

**RemoteUrl** <https://github.com/sooyong1/flps>

**RemoteRef** HEAD

**RemoteSha** a7f9e95e7145532b883227f65f7d6705daca767e

## Contents

flps-package . . . . .	2
binary . . . . .	3
continuous . . . . .	4
example0 . . . . .	5
example1 . . . . .	6
example2 . . . . .	7
example3 . . . . .	8
flps_plot . . . . .	9
graded . . . . .	9
importModel . . . . .	11
makeInpData . . . . .	11
makeSimData . . . . .	13
modelBuilder . . . . .	15
plot.flps . . . . .	15
print.flps . . . . .	16
print.summary.flps . . . . .	16
runFLPS . . . . .	17
summary.flps . . . . .	19
<b>Index</b>	<b>20</b>

---

flps-package	<i>Fully latent principal stratification</i>
--------------	--

---

## Description

The FLPS package conducts Bayesian analysis for fully latent principal stratification via rstan.

`_PACKAGE`

## Details

The 'flps' package.

## Author(s)

Sooyong Lee <sooyong109@gmail.com>

## References

Sales, A. C., & Pane, J. F. (2019). The role of mastery learning in an intelligent tutoring system: Principal stratification on a latent variable. *The Annals of Applied Statistics*, 13(1), 420-443. Lee, S., Adam, S., Kang, H.-A., & Whittaker, T. A. (2022). Fully latent principal stratification: Combining ps with model-based measurement models. In *The annual meeting of the psychometric society* (pp. 287-298).

---

binary

*binary.rda*

---

### Description

A data set containing binary items information. Example data regenerated from CTA1

### Usage

binary

### Format

A data frame with variables:

**schid** School ID

**id** Student ID

**sex** 0 = boys; 1 = girls

**race** 0 = White; 1 = Others

**pretest** Pre test scores

**stdscore** Standardized scores

**cm\_sex** Cluster-mean of sex

**cm\_race** Cluster-mean of race

**cm\_pretest** Cluster-mean of Pre test scores

**cm\_stdscore** Cluster-mean of of Standardized scores

**trt** Treatment assignment; 0 = control, 1 = treatment

**Y** Outcome

**q1** Binary item

**q2** Binary item

**q3** Binary item

**q4** Binary item

**q5** Binary item

**q6** Binary item

**q7** Binary item

**q8** Binary item

**q9** Binary item

**q10** Binary item

**q11** Binary item

**q12** Binary item

**q13** Binary item

**q14** Binary item  
**q15** Binary item  
**q16** Binary item  
**q17** Binary item  
**q18** Binary item  
**q19** Binary item  
**q20** Binary item

### Source

CTA1

### Examples

```
data(binary)
summary(binary)
```

---

continuous

*continuous.rda*

---

### Description

A data set containing continuous items information. Example data regenerated from CTA1

### Usage

```
continuous
```

### Format

A data frame with variables:

**schid** School ID  
**id** Student ID  
**sex** 0 = boys; 1 = girls  
**race** 0 = White; 1 = Others  
**pretest** Pre test scores  
**stdscore** Standardized scores  
**cm\_sex** Cluster-mean of sex  
**cm\_race** Cluster-mean of race  
**cm\_pretest** Cluster-mean of Pre test scores  
**cm\_stdscore** Cluster-mean of of Standardized scores  
**trt** Treatment assignment; 0 = control, 1 = treatment

- Y** Outcome
- q1** Continuous item
- q2** Continuous item
- q3** Continuous item
- q4** Continuous item
- q5** Continuous item
- q6** Continuous item
- q7** Continuous item
- q8** Continuous item
- q9** Continuous item
- q10** Continuous item
- q11** Continuous item
- q12** Continuous item
- q13** Continuous item
- q14** Continuous item
- q15** Continuous item
- q16** Continuous item
- q17** Continuous item
- q18** Continuous item
- q19** Continuous item
- q20** Continuous item

**Source**

CTA1

**Examples**

```
data(continuous)
summary(continuous)
```

---

example0

*example0.rda*

---

**Description**

A generated data set for rasch FLPS to mimic CTA1 data.

**Usage**

```
example0
```

**Format**

A data frame with variables:

**Y** Outcome

**trt** Treatment assignment; 0 = control, 1 = treatment

**sex** 0 = boys; 1 = girls

**race** 0 = White; 1 = Others

**pretest** Pre test scores

**stdscore** Standardized scores

**V1** Binary item

**V2** Binary item

**V3** Binary item

**V4** Binary item

**V5** Binary item

**V6** Binary item

**V7** Binary item

**V8** Binary item

**V9** Binary item

**V10** Binary item

**V11** Binary item

**V12** Binary item

---

example1

*example1.rda*

---

**Description**

A generated data set containing binary items information.

**Usage**

example1

**Format**

A data frame with variables:

**Y** Outcome

**trt** Treatment assignment; 0 = control, 1 = treatment

**X1** Continuous covariate 1

**X2** Continuous covariate 2

**q1** Binary item  
**q2** Binary item  
**q3** Binary item  
**q4** Binary item  
**q5** Binary item  
**q6** Binary item  
**q7** Binary item  
**q8** Binary item  
**q9** Binary item  
**q10** Binary item

---

example2

*example2.rda*

---

### **Description**

A generated data set containing continuous items information.

### **Usage**

example2

### **Format**

A data frame with variables:

**Y** Outcome

**trt** Treatment assignment; 0 = control, 1 = treatment

**X1** Continuous covariate 1

**X2** Continuous covariate 2

**q1** Continuous item

**q2** Continuous item

**q3** Continuous item

**q4** Continuous item

**q5** Continuous item

**q6** Continuous item

**q7** Continuous item

**q8** Continuous item

**q9** Continuous item

**q10** Continuous item

---

example3

*example3.rda*

---

### **Description**

A generated data set for multidimensional FLPS.

### **Usage**

example3

### **Format**

A data frame with variables:

**Y** Outcome

**Z** Treatment assignment; 0 = control, 1 = treatment

**X1** Continuous covariate 1

**X2** Continuous covariate 2

**X3** Continuous covariate 2

**X4** Continuous covariate 2

**V1** Continuous item

**V2** Continuous item

**V3** Continuous item

**V4** Continuous item

**V5** Continuous item

**V6** Continuous item

**V7** Continuous item

**V8** Continuous item

**V9** Continuous item

**V10** Continuous item

**V11** Continuous item

**V12** Continuous item



---

flps_plot	<i>Make plots related to FLPS models</i>
-----------	--

---

**Description**

Make plots related to FLPS models

**Usage**

```
flps_plot(object, type = "causal", ...)
```

**Arguments**

object	a <a href="#">flps</a> object
type	a character indicating the type of plots
...	Additional features related to plots

**Value**

A [ggplot](#) object that can be further customized using the **ggplot2** package.

---

graded	<i>graded.rda</i>
--------	-------------------

---

**Description**

A data set containing graded response items information. Example data regenerated from CTA1

**Usage**

```
graded
```

**Format**

A data frame with variables:

**schid** School ID

**id** Student ID

**sex** 0 = boys; 1 = girls

**race** 0 = White; 1 = Others

**pretest** Pre test scores

**stdscore** Standardized scores

**cm\_sex** Cluster-mean of sex

**cm\_race** Cluster-mean of race  
**cm\_pretest** Cluster-mean of Pre test scores  
**cm\_stdscore** Cluster-mean of of Standardized scores  
**trt** Treatment assignment; 0 = control, 1 = treatment  
**Y** Outcome  
**q1** Graded response item  
**q2** Graded response item  
**q3** Graded response item  
**q4** Graded response item  
**q5** Graded response item  
**q6** Graded response item  
**q7** Graded response item  
**q8** Graded response item  
**q9** Graded response item  
**q10** Graded response item  
**q11** Graded response item  
**q12** Graded response item  
**q13** Graded response item  
**q14** Graded response item  
**q15** Graded response item  
**q16** Graded response item  
**q17** Graded response item  
**q18** Graded response item  
**q19** Graded response item  
**q20** Graded response item

**Source**

CTA1

**Examples**

```
data(graded)  
summary(graded)
```

---

importModel	<i>Import compiled Stan object</i>
-------------	------------------------------------

---

**Description**

Import compiled Stan object

**Usage**

```
importModel(lv_type, multilevel = FALSE, lv_randomeffect = FALSE)
```

**Arguments**

lv_type	a character indicating the type of FLPS model.
multilevel	a logical indicating multilevel Stan model.
lv_randomeffect	A logical indicating whether to estimate random effects for latent variables.

**Value**

a Stan compiled stanmodel object generated by modelBuilder

---

makeInpData	<i>Generate a matrix style data for simulation</i>
-------------	--

---

**Description**

[makeInpData](#) is a function for generating a data based on the given information.

**Usage**

```
makeInpData(
  N,
  R2Y,
  R2eta,
  omega,
  tau0,
  tau1,
  betaL,
  betaY,
  linear = TRUE,
  ydist = "n",
  lambda,
  nitem,
  nfac = 1,
```

```

    lvmodel,
    fcovmat,
    item.missing = TRUE,
    misspec = FALSE,
    cov.res = 0,
    relsize = 0.6
)

```

### Arguments

N	a numeric indicating sample size.
R2Y	a numeric indicating predictive power of covariates.
R2eta	a numeric indicating Predictive power of latent variable
omega	a numeric indicating the size of effect of latent factor on the outcome.
tau0	a numeric indicating the size of difference in the outcome between the treatment and the control.
tau1	a numeric indicating the principal effect
betaL	a numeric vector indicating the effects of covariates on the latent factor
betaY	a numeric vector indicating the effects of covariates on the outcome
linear	a logical whether the relationship between the outcome and covariates is linear (default is TRUE).
ydist	a character indicating the outcome distribution (default is n).
lambda	a numeric indicating the mean of Worked problems/person. (extent to which covariates predict eta).
nitem	a numeric indicating the number of maximum measurement items given to students.
nfac	a numeric indicating the number of latent factors
lvmodel	a character specifying a type of latent variable model.
fcovmat	a matrix indicating the variance-covariance matrix of latent factors when nfac > 1
item.missing	a logical to make the measurement item data missing for the control group (default is TRUE).
misspec	a logical to allow cross-loadings across latent factors when nfac > 1 (default is FALSE).
cov.res	a logical to allow for residual correlations (only for CFA model) (default is 0).
relsize	a numeric indicating the degree to which the latent factor explain the variances of continuous items (only for CFA model) (default is 0.6).

### Value

a list containing all the data related to population values and running FLPS.

**Examples**

```

sdat <- makeInpData(
  N      = 200, # sample size
  R2Y    = 0.2, # r^2 of outcome
  R2eta  = 0.5, # r^2 of eta by one covariates
  omega  = 0.2, # the effect of eta
  tau0   = 0.13, # direct effect
  tau1   = -0.06, # interaction effect between Z and eta
  betaL  = 0.2,
  betaY  = 0.4,
  lambda = 0.8, # the proportion of administered items
  nitem  = 10, # the total number of items
  nfac   = 1, # the number of latent factors
  lvmodel = '2p1' )

```

---

makeSimData

*Generate Fully Latent Principal Stratification data for simulation*


---

**Description**

[makeInpData](#) is a function for generating a data based on the given information.

**Usage**

```

makeSimData(
  N,
  R2Y,
  R2eta,
  omega,
  tau0,
  tau1,
  betaL,
  betaY,
  linear = TRUE,
  ydist = "n",
  lambda,
  nitem,
  nfac,
  lvmodel,
  fcovmat,
  item.missing = TRUE,
  misspec = FALSE,
  cov.res = 0,
  relsize = 0.6
)

```

**Arguments**

N	a numeric indicating sample size.
R2Y	a numeric indicating predictive power of covariates.
R2eta	a numeric indicating Predictive power of latent variable
omega	a numeric indicating the size of effect of latent factor on the outcome.
tau0	a numeric indicating the size of difference in the outcome between the treatment and the control.
tau1	a numeric indicating the principal effect
betaL	a numeric vector indicating the effects of covariates on the latent factor
betaY	a numeric vector indicating the effects of covariates on the outcome
linear	a logical whether the relationship between the outcome and covariates is linear (default is TRUE).
ydist	a character indicating the outcome distribution (default is n).
lambda	a numeric indicating the mean of Worked problems/person. (extent to which covariates predict eta).
nitem	a numeric indicating the number of maximum measurement items given to students.
nfac	a numeric indicating the number of latent factors
lvmodel	a character specifying a type of latent variable model.
fcovmat	a matrix indicating the variance-covariance matrix of latent factors when nfac > 1
item.missing	a logical to make the measurement item data missing for the control group (default is TRUE).
misspec	a logical to allow cross-loadings across latent factors when nfac > 1 (default is FALSE).
cov.res	a logical to allow for residual correlations (only for CFA model) (default is 0).
resize	a numeric indicating the degree to which the latent factor explain the variances of continuous items (only for CFA model) (default is 0.6).

**Value**

a list containing all the data related to population values and running FLPS.

**Examples**

```
sdat <- makeSimData(
  N       = 200, # sample size
  R2Y     = 0.2, # r^2 of outcome
  R2eta   = 0.5, # r^2 of eta by one covariates
  omega   = 0.2, # the effect of eta
  tau0    = 0.13, # direct effect
  tau1    = -0.06, # interaction effect between Z and eta
  betaL   = 0.2,
  betaY   = 0.4,
```

```

lambda = 0.8, # the proportion of administered items
nitem   = 10, # the total number of items
nfac    = 1,  # the number of latent factors
lvmodel = '2pl' )

```

---

modelBuilder

*Generate compiled Stan object to facilitate the analysis*


---

### Description

Generate compiled Stan object to facilitate the analysis

### Usage

```
modelBuilder(lv_type, multilevel = FALSE, lv_randomeffect = FALSE)
```

### Arguments

lv\_type            A character string specifying the type of FLPS model  
multilevel        a logical indicating multilevel Stan model.  
lv\_randomeffect   A logical indicating whether to estimate random effects for latent variables.

### Value

There's no return, but the compiled objects are saved in the package root directory.

---

plot.flps

*Plot*


---

### Description

Plot

### Usage

```
## S3 method for class 'flps'
plot(x, type = NULL, pars = c("tau0", "tau1"), ...)
```

### Arguments

x                    an object of class `flps`  
type                 a string for the type of plot  
pars                 a character vector indicating the target parameters  
...                  additional options for `stan_plot`

**Value**

A `ggplot` object that can be further customized using the **ggplot2** package.

---

<code>print.flps</code>	<i>Print results</i>
-------------------------	----------------------

---

**Description**

Print results

**Usage**

```
## S3 method for class 'flps'
print(x, ...)
```

**Arguments**

<code>x</code>	an object of class <code>flps</code>
<code>...</code>	additional options for future development

**Value**

Summary of FLPS model are printed.

---

<code>print.summary.flps</code>	<i>Print summary of results</i>
---------------------------------	---------------------------------

---

**Description**

Print summary of results

**Usage**

```
## S3 method for class 'summary.flps'
print(x, type = "structures", ...)
```

**Arguments**

<code>x</code>	an object of class <code>flps</code>
<code>type</code>	a string for the part of FLPS model <ul style="list-style-type: none"> <li>• <code>structures</code> : prints the results of structural parts.</li> <li>• <code>measurement</code> : prints the results of measurement parts.</li> <li>• <code>latent</code> : prints the information of individual latent scores</li> <li>• <code>raw</code> : prints the results via the summary function of <b>rstan</b> package..</li> </ul>
<code>...</code>	additional options for future development



**Value**

Summary of FLPS model are printed.

---

runFLPS	<i>Conduct fully latent principal stratification</i>
---------	--

---

**Description**

Conduct fully latent principal stratification

**Usage**

```
runFLPS(
  inp_data = NULL,
  compiled_stan = NULL,
  outcome = NULL,
  trt = NULL,
  covariate = NULL,
  lv_model = NULL,
  lv_type = NULL,
  priors_input = NULL,
  stan_options = list(),
  ...
)
```

**Arguments**

<code>inp_data</code>	A matrix or data frame containing the input data.
<code>compiled_stan</code>	An object of S4 class <code>stanmodel</code> produced by the <code>modelBuilder</code> function.
<code>outcome</code>	A character string specifying the outcome variable's name.
<code>trt</code>	A character string specifying the treatment or control group variable's name.
<code>covariate</code>	A character string specifying the covariate variable names.
<code>lv_model</code>	A description of the latent variable model using syntax akin to the <b>lavaan</b> package. Key operators include: <ul style="list-style-type: none"> <li>• <code>=~</code> : Denotes associations between factors and indicators (e.g., <math>F1 \sim v1 + v2 + v3</math>). All indicators associated with the corresponding factor should be written in the same line with <code>+</code>.</li> <li>• <code>+</code> : Specifies a series of indicators.</li> </ul>
<code>lv_type</code>	A character string indicating the type of latent variable models.
<code>priors_input</code>	A list specifying the priors or defaults to $N(0, 5)$ if not provided. Relevant parameters: <code>tau0</code> (group difference), <code>tau1</code> (principal effects), and <code>omega</code> (effect of latent factors on outcome). Ensure that the lengths of <code>tau1</code> and <code>omega</code> match the number of factors. Examples:

	<ul style="list-style-type: none"> <li>• <code>list(tau0 = c(0, 1), tau1 = c(0.5, 1))</code> : Mean and variance for normal priors.</li> <li>• <code>list(tau1 = list(c(0.5, 1), c(-0.4, 1)))</code> : For two factors.</li> </ul>
<code>stan_options</code>	A list of options for <code>[rstan::stan()]</code> , specified as 'name = value'.
<code>...</code>	Additional parameters for the latent variable models <ul style="list-style-type: none"> <li>• <code>nclass</code> A number specifying the number of latent classes.</li> <li>• <code>multilevel</code> A logical indicating if a multilevel structure is present.</li> <li>• <code>lv_randomeffect</code> A logical indicating whether to estimate random effects for latent variables.</li> <li>• <code>group_id</code> A string for grouping variable for multilevel structure.</li> </ul>

### Value

An object of class `flps` encompassing a `stanfit` object. Components include:

<code>call</code>	Function call with arguments.
<code>inp_data</code>	The input data frame provided.
<code>flps_model</code>	The Stan syntax used in <code>[rstan::stan()]</code> .
<code>flps_data</code>	Data list used for <code>[rstan::stan()]</code> .
<code>flps_fit</code>	Resulting <code>stanfit</code> object.
<code>time</code>	A numeric; Time taken for computation

### See Also

`[rstan::stan()]`

### Examples

```
inp_data <- flps::makeInpData(
  N       = 200,
  R2Y    = 0.2,
  R2eta  = 0.5,
  omega  = 0.2,
  tau0   = 0.23,
  tau1   = -0.16,
  betaL  = 0.1,
  betaY  = 0.2,
  lambda = 0.8,
  nitem  = 10,
  nfac   = 1,
  lvmodel = 'rasch' )
```

```
res <- runFLPS(
  inp_data = inp_data,
  outcome  = "Y",
  trt      = "Z",
  covariate = c("X1"),
  lv_type  = "rasch",
```

```
lv_model = "F =~ v1 + v2 + v3 + v4 + v5 + v6 + v7 + v8 + v9 + v10",
stan_options = list(iter = 1000, warmup = 500, cores = 1, chains = 2)
)
```

---

summary.flps

*Summarize the results*


---

### Description

Summarize the results

### Usage

```
## S3 method for class 'flps'
summary(object, type = "structures", ...)
```

### Arguments

object	an object of class <code>flps</code>
type	a string for the part of FLPS model <ul style="list-style-type: none"> <li>• <code>structures</code> : prints the results of structural parts.</li> <li>• <code>measurement</code> : prints the results of measurement parts.</li> <li>• <code>latent</code> : prints the information of individual latent scores</li> <li>• <code>raw</code> : prints the results via the summary function of <b>rstan</b> package..</li> </ul>
...	additional options for future development

### Value

Summary of FLPS model are printed.

# Index

## \* datasets

- binary, [3](#)
- continuous, [4](#)
- example0, [5](#)
- example1, [6](#)
- example2, [7](#)
- example3, [8](#)
- graded, [9](#)

## \* package

- flps-package, [2](#)

## \* rstan

- runFLPS, [17](#)

[binary](#), [3](#)

[continuous](#), [4](#)

[example0](#), [5](#)

[example1](#), [6](#)

[example2](#), [7](#)

[example3](#), [8](#)

[flps](#), [9](#), [15](#), [16](#), [19](#)

[flps \(flps-package\)](#), [2](#)

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

[flps\\_plot](#), [9](#)

[ggplot](#), [9](#), [16](#)

[graded](#), [9](#)

[importModel](#), [11](#)

[makeInpData](#), [11](#), [11](#), [13](#)

[makeSimData](#), [13](#)

[modelBuilder](#), [15](#)

[plot.flps](#), [15](#)

[print.flps](#), [16](#)

[print.summary.flps](#), [16](#)

[runFLPS](#), [17](#)

[stanfit](#), [18](#)

[summary.flps](#), [19](#)