Package 'LAWBL'

May 16, 2022

Type Package

Title Latent (Variable) Analysis with Bayesian Learning

Version 1.5.0

Date 2022-05-13

Author Jinsong Chen [aut, cre, cph]

Maintainer Jinsong Chen <jinsong.chen@live.com>

Description A variety of models to analyze latent variables based on Bayesian learning: the partially CFA (Chen, Guo, Zhang, & Pan, 2020) <DOI:10.1037/met0000293>; generalized PCFA; partially confirmatory IRM (Chen, 2020) <DOI:10.1007/s11336-020-09724-3>; Bayesian regularized EFA <DOI:10.1080/10705511.2020.1854763>; Fully and partially EFA.

License GPL-3

Encoding UTF-8

LazyData true

Depends R (>= 3.6.0)

Imports stats, MASS, coda

RoxygenNote 7.2.0

URL https://github.com/Jinsong-Chen/LAWBL,

   https://jinsong-chen.github.io/LAWBL/

BugReports https://github.com/Jinsong-Chen/LAWBL/issues

Suggests knitr, rmarkdown, testthat

VignetteBuilder knitr

NeedsCompilation no

Repository CRAN

Date/Publication 2022-05-16 07:00:05 UTC

R topics documented:

LAWBL-package   *LAWBL: Latent (Variable) Analysis with Bayesian Learning*

Description

This package is to provide a variety of models to analyze latent variables based on Bayesian learning.

Details

LAWBL represents a partially confirmatory / exploratory approach to model latent variables based on Bayesian learning. Built on the power of statistical learning, it can address psychometric challenges such as parameter specification, local dependence, and factor extraction. Built on the scalability and flexibility of Bayesian inference and resampling techniques, it can accommodate modeling frameworks such as factor analysis, item response theory, cognitive diagnosis modeling and causal or explanatory modeling. The package can also handle different response formats or a mix of them, with or without missingness. The variety of models provide a partial approach covering a wide range of the exploratory-confirmatory continuum under the context of latent variable modeling.

Towards the confirmatory end, this package includes the Partially Confirmatory Factor Analysis (PCFA) model for continuous data (Chen, Guo, Zhang, & Pan, 2020), the generalized PCFA (GPCFA) model covering continuous, categorical, and mixed-type data, and the partially confirmatory item response model (PCIRM) for continuous and dichotomous data with intercept terms (Chen, 2020). For PCFA, GPCFA, and PCIRM, there are two major model variants with different constraints for identification. One assumes local independence (LI) with a more exploratory tendency, which can be also called the E-step. The other allows local dependence (LD) with a more confirmatory tendency, which can be also called the C-step.

Towards the exploratory end, the Bayesian regularized EFA (BREFA) with factor extraction and parameter estimation in one step (Chen 2021) is offered. It's further improved as the Fully and partially EFA with better performance and partial knowledge.

Parameters are obtained by sampling from the posterior distributions with the Markov chain Monte Carlo (MCMC) techniques. Different Bayesian learning methods are used to regularize the loading pattern, local dependence, and/or factor identification.

Note

This package is under development. You are very welcome to send me any comments or suggestions for improvements, and to share with me any problems you may encounter with the use of this package.

Author(s)

Jinsong Chen, <jinsong.chen@live.com>

References

Chen, J. (2020). A partially confirmatory approach to the multidimensional item response theory with the Bayesian Lasso. *Psychometrika*. 85(3), 738-774. DOI:10.1007/s11336-020-09724-3.

Chen, J., Guo, Z., Zhang, L., & Pan, J. (2021). A partially confirmatory approach to scale development with the Bayesian Lasso. *Psychological Methods*. 26(2), 210–235. DOI: 10.1037/met0000293.

Chen, J. (2021). A generalized partially confirmatory factor analysis framework with mixed Bayesian Lasso methods. *Multivariate Behavioral Research*. DOI: 10.1080/00273171.2021.1925520.

Chen, J. (2021). A Bayesian regularized approach to exploratory factor analysis in one step. *Structural Equation Modeling: A Multidisciplinary Journal*. DOI: 10.1080/10705511.2020.1854763.

Chen, J. (2022). Partially confirmatory approach to factor analysis with Bayesian learning: A LAWBL tutorial. *Structural Equation Modeling: A Multidisciplinary Journal*. DOI: 10.1080/00273171.2021.1925520.

Chen, J. (In Press). Fully and partially exploratory factor analysis with bi-level Bayesian regularization. *Behavior Research Methods*.

---

nlsy27 *National Longitudinal Survey of Youth 1997*

Description

A data set consisted of 3,458 individual responses to 27 mixed-type items, with a 1.12 percentage of missing data

Usage

nlsy27

Format

A list with components:

dat The response data

Q Intial design matrix with three factors and two to three specified loadings per factor

cati Indices of categorical (polytomous) items

---

pcfa   *(Generalized) Partially Confirmatory Factor Analysis*

Description

PCFA is a partially confirmatory approach covering a wide range of the exploratory-confirmatory continuum in factor analytic models (Chen, Guo, Zhang, & Pan, 2021). The PCFA is only for continuous data, while the generalized PCFA (GPCFA; Chen, 2021) covers both continuous and categorical data.

There are two major model variants with different constraints for identification. One assumes local independence (LI) with a more exploratory tendency, which can be also called the E-step. The other allows local dependence (LD) with a more confirmatory tendency, which can be also called the C-step. Parameters are obtained by sampling from the posterior distributions with the Markov chain Monte Carlo (MCMC) techniques. Different Bayesian Lasso methods are used to regularize the loading pattern and LD. The estimation results can be summarized with `summary.lawbl` and the factorial eigenvalue can be plotted with `plot_lawbl`.

Usage

```
pcfa(
  dat,
  Q,
  LD = TRUE,
  cati = NULL,
  cand_thd = 0.2,
  PPMC = FALSE,
  burn = 5000,
  iter = 5000,
  update = 1000,
  missing = NA,
  rfit = TRUE,
  sign_check = FALSE,
  sign_eps = -0.5,
  rs = FALSE,
  auto_stop = FALSE,
  max_conv = 10,
  rseed = 12345,
  digits = 4,
  alas = FALSE,
  verbose = FALSE
)
```

Arguments

dat   A $N \times J$ data *matrix* or *data.frame* consisting of the responses of N individuals to J items.

Q	A $J \times K$ design matrix for the loading pattern with K factors and J items. Elements are 1, -1, and 0 for specified, unspecified, and zero-fixed loadings, respectively. For models with LI or the E-step, one can specify a few (e.g., 2) loadings per factor. For models with LD or the C-step, the sufficient condition of one specified loading per item is suggested, although there can be a few items without any specified loading. See Examples.
LD	logical; TRUE for allowing LD (model with LD or C-step).
cati	The set of categorical (polytomous) items in sequence number (i.e., 1 to J); NULL for no and -1 for all items (default is NULL).
cand_thd	Candidate parameter for sampling the thresholds with the MH algorithm.
PPMC	logical; TRUE for conducting posterior predictive model checking.
burn	Number of burn-in iterations before posterior sampling.
iter	Number of formal iterations for posterior sampling (> 0).
update	Number of iterations to update the sampling information.
missing	Value for missing data (default is NA).
rfit	logical; TRUE for providing relative fit (DIC, BIC, AIC).
sign_check	logical; TRUE for checking sign switch of loading vector.
sign_eps	minimum value for switch sign of loading vector (if sign_check=TRUE).
rs	logical; TRUE for enabling recommendation system.
auto_stop	logical; TRUE for enabling auto stop based on EPSR<1.1.
max_conv	maximum consecutive number of convergence for auto stop.
rseed	An integer for the random seed.
digits	Number of significant digits to print when printing numeric values.
alas	logical; for adaptive Lasso or not. The default is FALSE.
verbose	logical; to display the sampling information every update or not.

- Feigen: Eigenvalue for each factor.
- NLA_le3: Number of Loading estimates >= .3 for each factor.
- Shrink: Shrinkage (or ave. shrinkage for each factor for adaptive Lasso).
- EPSR & NCOV: EPSR for each factor & # of convergence.
- Ave. Thd: Ave. thresholds for polytomous items.
- Acc Rate: Acceptance rate of threshold (MH algorithm).
- LD>.2 >.1 LD>.2 >.1: # of LD terms larger than .2 and .1, and LD's shrinkage parameter.
- #Sign_sw: Number of sign switch for each factor.

Value

pcfa returns an object of class lawbl without item intercepts. It contains a lot of information about the posteriors that can be summarized using summary.lawbl.

References

Chen, J., Guo, Z., Zhang, L., & Pan, J. (2021). A partially confirmatory approach to scale development with the Bayesian Lasso. *Psychological Methods*. 26(2), 210–235. DOI: 10.1037/met0000293.

Chen, J. (2021). A generalized partially confirmatory factor analysis framework with mixed Bayesian Lasso methods. *Multivariate Behavioral Research*. DOI: 10.1080/00273171.2021.1925520.

Examples

```
####################################################
#  Example 1: Estimation with continuous data & LD  #
####################################################

dat <- sim18cfa1$dat
J <- ncol(dat)
K <- 3
Q<-matrix(-1,J,K);
Q[1:6,1]<-Q[7:12,2]<-Q[13:18,3]<-1

m0 <- pcfa(dat = dat, Q = Q, LD = TRUE,burn = 2000, iter = 2000)
summary(m0) # summarize basic information
summary(m0, what = 'qlambda') #summarize significant loadings in pattern/Q-matrix format
summary(m0, what = 'offpsx') #summarize significant LD terms

####################################################
#  Example 2: Estimation with categorical data & LI  #
####################################################
dat <- sim18ccfa40$dat
J <- ncol(dat)
K <- 3
Q<-matrix(-1,J,K);
Q[1:2,1]<-Q[7:8,2]<-Q[13:14,3]<-1

m1 <- pcfa(dat = dat, Q = Q,LD = FALSE,cati=-1,burn = 2000, iter = 2000)
summary(m1) # summarize basic information
summary(m1, what = 'qlambda') #summarize significant loadings in pattern/Q-matrix format
summary(m1, what = 'offpsx') #summarize significant LD terms
summary(m1,what='thd') #thresholds for categorical items
```

---

pcirm   *Partially Confirmatory Item Response Model*

Description

pcirm is a partially confirmatory approach to item response models (Chen, 2020), which estimates the intercept for continuous and dichotomous data. Similar to PCFA and GPCFA, there are two major model variants with different constraints for identification. One assumes local independence (LI) with a more exploratory tendency, which can be also called the E-step. The other allows

local dependence (LD) with a more confirmatory tendency, which can be also called the C-step. Parameters are obtained by sampling from the posterior distributions with the Markov chain Monte Carlo (MCMC) techniques. Different Bayesian Lasso methods are used to regularize the loading pattern and LD. The estimation results can be summarized with `summary.lawbl` and the factorial eigenvalue can be plotted with `plot_lawbl`.

Usage

```
pcirm(
  dat,
  Q,
  LD = TRUE,
  cati = NULL,
  PPMC = FALSE,
  burn = 5000,
  iter = 5000,
  update = 1000,
  missing = NA,
  rseed = 12345,
  sign_check = FALSE,
  sign_eps = -0.5,
  auto_stop = FALSE,
  max_conv = 10,
  digits = 4,
  alas = FALSE,
  verbose = FALSE
)
```

Arguments

dat	A $N \times J$ data *matrix* or *data.frame* consisting of the responses of N individuals to J items. Only continuous and dichotomous data are supported.
Q	A $J \times K$ design matrix for the loading pattern with K factors and J items. Elements are 1, -1, and 0 for specified, unspecified, and zero-fixed loadings, respectively. For models with LI or the E-step, one can specify a few (e.g., 2) loadings per factor. For models with LD or the C-step, the sufficient condition of one specified loading per item is suggested, although there can be a few items without any specified loading. See `Examples`.
LD	logical; TRUE for allowing LD (model with LD or C-step).
cati	The set of dichotomous items in sequence number (i.e., 1 to J); NULL for no and -1 for all items (default is NULL).
PPMC	logical; TRUE for conducting posterior predictive model checking.
burn	Number of burn-in iterations before posterior sampling.
iter	Number of formal iterations for posterior sampling (> 0).
update	Number of iterations to update the sampling information.
missing	Value for missing data (default is NA).

rseed	An integer for the random seed.
sign_check	logical; TRUE for checking sign switch of loading vector.
sign_eps	minimum value for switch sign of loading vector (if sign_check=TRUE).
auto_stop	logical; TRUE for enabling auto stop based on EPSR<1.1.
max_conv	maximum consecutive number of convergence for auto stop.
digits	Number of significant digits to print when printing numeric values.
alas	logical; for adaptive Lasso or not. The default is FALSE.
verbose	logical; to display the sampling information every update or not.

- Feigen: Eigenvalue for each factor.
- NLA_le3: Number of Loading estimates >= .3 for each factor.
- Shrink: Shrinkage (or ave. shrinkage for each factor for adaptive Lasso).
- EPSR & NCOV: EPSR for each factor & # of convergence.
- Ave. Int.: Ave. item intercept.
- LD>.2 >.1 LD>.2 >.1: # of LD terms larger than .2 and .1, and LD's shrinkage parameter.
- #Sign_sw: Number of sign switch for each factor.

Value

pcirm returns an object of class lawbl with item intercepts. It contains a lot of information about the posteriors that can be summarized using summary.lawbl.

References

Chen, J. (2020). A partially confirmatory approach to the multidimensional item response theory with the Bayesian Lasso. *Psychometrika*. 85(3), 738-774. DOI:10.1007/s11336-020-09724-3.

Examples

```
####################################
#  Example 1: Estimation with LD   #
####################################

dat <- sim24ccfa21$dat
J <- ncol(dat)
K <- 3
Q<-matrix(-1,J,K);
Q[1:8,1]<-Q[9:16,2]<-Q[17:24,3]<-1

m0 <- pcirm(dat = dat, Q = Q, LD = TRUE, cati = -1, burn = 2000,iter = 2000)
summary(m0) # summarize basic information
summary(m0, what = 'qlambda') #summarize significant loadings in pattern/Q-matrix format
summary(m0, what = 'offpsx') #summarize significant LD terms

####################################
#  Example 2: Estimation with LD   #
```

```
####################################

Q<-cbind(Q,-1);
Q[15:16,4]<-1

m1 <- pcirm(dat = dat, Q = Q, LD = FALSE, cati = -1, burn = 2000,iter = 2000)
summary(m1) # summarize basic information
summary(m1, what = 'qlambda') #summarize significant loadings in pattern/Q-matrix format
summary(m1, what = 'offpsx') #summarize significant LD terms
```

---

pefa	*Partially Exploratory Factor Analysis*

Description

PEFA is a partially exploratory approach to factor analysis, which can incorporate partial knowledge together with unknown number of factors, using bi-level Bayesian regularization. When partial knowledge is not needed, it reduces to the fully exploratory factor analysis (FEFA; Chen, 2021). A large number of factors can be imposed for selection where true factors will be identified against spurious factors. The loading vector is reparameterized to tackle model sparsity at the factor and loading levels with the multivariate spike and slab priors. Parameters are obtained by sampling from the posterior distributions with the Markov chain Monte Carlo (MCMC) techniques. The estimation results can be summarized with summary.lawbl and the trace or density of the posterior can be plotted with plot_lawbl.

Usage

```
pefa(
  dat,
  Q = NULL,
  K = 8,
  mjf = 3,
  PPMC = FALSE,
  burn = 5000,
  iter = 5000,
  missing = NA,
  eig_eps = 1,
  sign_eps = 0,
  rfit = TRUE,
  rs = FALSE,
  update = 1000,
  rseed = 12345,
  verbose = FALSE,
  auto_stop = FALSE,
  max_conv = 10,
  digits = 4
)
```

Arguments

dat	A $N \times J$ data *matrix* or *data.frame* consisting of the responses of N individuals to J items.
Q	A $J \times K$ design matrix for the loading pattern with K factors and J items for PEFA. Elements are 1, -1, and 0 for specified, unspecified, and zero-fixed loadings, respectively. It's not needed for FEFA, which is the default. See Examples.
K	Maximum number of factors for selection under FEFA. Not used for PEFA.
mjf	Minimum number of items per factor.
PPMC	logical; TRUE for conducting posterior predictive model checking.
burn	Number of burn-in iterations before posterior sampling.
iter	Number of formal iterations for posterior sampling (> 0).
missing	Value for missing data (default is NA).
eig_eps	minimum eigenvalue for factor extraction.
sign_eps	minimum value for switch sign of loading vector.
rfit	logical; TRUE for providing relative fit (DIC, BIC, AIC).
rs	logical; TRUE for enabling recommendation system.
update	Number of iterations to update the sampling information.
rseed	An integer for the random seed.
verbose	logical; to display the sampling information every update or not.
	• Feigen: Eigenvalue for each factor.
	• NLA_lg0: Number of Loading magnitudes > 0 for each factor.
	• iShrink: Inverted shrinkage parameter for each factor.
	• True Fa: Is the factor identified as true or not.
	• EPSR & NCOV: EPSR for each factor & # of convergence.
	• ROW: LA overflow,sign switch,bk=0, <eig_eps: Loading overflow, sign switch, vector bk=0 and eigenvalue<eig_eps.
auto_stop	logical; TRUE for enabling auto stop based on EPSR.
max_conv	maximum consecutive number of convergence for auto stop.
digits	Number of significant digits to print when printing numeric values.

Value

pcfa returns an object of class lawbl without item intercepts. It contains a lot of information about the posteriors that can be summarized using summary.lawbl.

References

Chen, J. (2021). A Bayesian regularized approach to exploratory factor analysis in one step. *Structural Equation Modeling: A Multidisciplinary Journal*, 28(4), 518-528. DOI: 10.1080/10705511.2020.1854763.

Chen, J. (In Press). Fully and partially exploratory factor analysis with bi-level Bayesian regularization. *Behavior Research Methods*.

Examples

```
######################################################
#  Example 1: Fully EFA                              #
######################################################

dat <- sim18cfa0$dat

m0 <- pefa(dat = dat, K=5, burn = 2000, iter = 2000,verbose = TRUE)
summary(m0) # summarize basic information
summary(m0, what = 'qlambda') #summarize significant loadings in pattern/Q-matrix format
summary(m0, what = 'phi') #summarize factorial correlations
summary(m0, what = 'eigen') #summarize factorial eigenvalue

#########################################################
#  Example 2: PEFA with two factors partially specified  #
#########################################################

J <- ncol(dat)
K <- 5
Q<-matrix(-1,J,K);
Q[1:2,1]<-Q[7:8,2]<-1
Q

m1 <- pefa(dat = dat, Q = Q,burn = 2000, iter = 2000,verbose = TRUE)
summary(m1)
summary(m1, what = 'qlambda')
summary(m1, what = 'phi')
summary(m1,what='eigen')
```

---

plot_lawbl *Posterior plots for* lawbl *object*

Description

Provide posterior plots based on the factorial eigenvalues of a lawbl object. For PEFA or FEFA, only true factors will be plotted.

Usage

```
plot_lawbl(object, what = "trace", istart = 1, iend = -1)
```

Arguments

object   A lawbl object

what   A list of options for what to plot.

         • trace: The trace of each factor's eigenvalue.

- density: The trace of each factor's eigenvalue.
- EPSR: Estimated Potential Scale Reduction (Gelman-Rubin diagnostics) for each factor.

istart   Starting point of the Markov chain for plotting.

iend   Ending point of the Markov chain for plotting; -1 for the actual final point.

Examples

```
dat <- sim18cfa0$dat
J <- ncol(dat)
K <- 3
Q<-matrix(-1,J,K);
Q[1:2,1]<-Q[7:8,2]<-Q[13:14,3]<-1


m0 <- pcfa(dat = dat, Q = Q, LD = FALSE,burn = 1000, iter = 1000)
plot_lawbl(m0) # trace
plot_lawbl(m0, what='density')
plot_lawbl(m0, what='EPSR')
```

---

sim18ccfa40   *Simulated CCFA data with LI and missingness*

Description

Categorical CFA data simulated based on 18 items, 3 factors, and 4 categories with local independence and 10 percent missingness at random; factorial correlation $\Phi = .3$.

Usage

```
sim18ccfa40
```

Format

A list with components:

dat A dataset with simulated responses of 1000 individuals to 18 items

qlam Loading pattern and values used to simulated the data

---

sim18ccfa41 *Simulated CCFA data with LD and missingness*

Description

Categorical CFA data simulated based on 18 items, 3 factors, and 4 categories with local dependence and 10 percent missingness at random; factorial correlation $\Phi = .3$.

Usage

```
sim18ccfa41
```

Format

A list with components:

dat A dataset with simulated responses of 1000 individuals to 18 items

qlam Loading pattern and values used to simulated the data

LD Local dependence between items (LD effect = .3)

---

sim18cfa0 *Simulated CFA data with LI*

Description

CFA data simulated based on 18 items, 3 factors and local independence; factorial correlation $\Phi = .3$.

Usage

```
sim18cfa0
```

Format

A list with components:

dat A dataset with simulated responses of 1000 individuals to 18 items

qlam Loading pattern and values used to simulated the data

| sim18cfa1 | *Simulated CFA data with LD* |

Description

CFA data simulated based on 18 items, 3 factors and local dependence; factorial correlation $\Phi = .3$.

Usage

```
sim18cfa1
```

Format

A list with components:

dat A dataset with simulated responses of 1000 individuals to 18 items

qlam Loading pattern and values used to simulated the data

LD Local dependence between items (LD effect = .3)

| sim18mcfa41 | *Simulated MCFA data with LD and Missingness* |

Description

CFA data mixed with continuous and categorical responses simulated based on 3 factors, 6 4-category items, 12 continuous items, local dependence, and 10 percent missigness at random; factorial correlation $\Phi = .3$.

Usage

```
sim18mcfa41
```

Format

A list with components:

dat A dataset with simulated responses of 1000 individuals to 18 items

qlam Loading pattern and values used to simulated the data

LD Local dependence between items (LD effect = .3)

---

sim24ccfa21   *Simulated CCFA data (dichotomous) with LD and a minor factor/trait*

Description

Categorical CFA data simulated based on 24 items, 4 factors, 2 categories and local dependence; factorial correlation $\Phi = .3$. The last factor/trait is minor (measured by cross-loadings only).

Usage

```
sim24ccfa21
```

Format

A list with components:

dat A dataset with simulated responses of 1000 individuals to 24 items

qlam Loading pattern and values used to simulated the data

LD Local dependence between items (LD effect = .3)

---

sim_lvm   *Simulating data with Latent Variable Modeling*

Description

sim_lvm can simulate data based on factor analysis or item response models with different response formats (continuous or categorical), loading patterns and residual covariance (local dependence) structures.

Usage

```
sim_lvm(
  N = 1000,
  mla = NULL,
  K = 3,
  J = 18,
  cpf = 0,
  lam = 0.7,
  lac = 0.3,
  phi = 0.3,
  ph12 = -1,
  ecr = 0,
  P = 0,
  b = 0.3,
  K1 = 0,
```

```
    ph1 = 0.2,
    b1 = 0.3,
    ilvl = NULL,
    cati = NULL,
    noc = c(4),
    misp = 0,
    ome_out = FALSE,
    necw = K,
    necb = K,
    add_ind = c(),
    add_la = 0.5,
    add_phi = 0,
    zero_it = 0,
    rseed = 333,
    digits = 4
)
```

Arguments

N	Sample size.
mla	Population loading matrix.
K	Number of factors (if `mla=NULL`).
J	Number of items (if `mla=NULL`).
cpf	Number of cross-loadings per factor (if `mla=NULL`).
lam	Number of formal iterations for posterior sampling.
lac	Number of iterations to update the sampling information.
phi	Homogeneous correlations between any two factors.
ph12	Correlation between factor 1 and 2 (if it's different from `phi`.
ecr	Residual correlation (local dependence).
P	Number of observable predictors (for MIMIC model).
b	Coefficients of observable predictors (for MIMIC model).
K1	Number of latent predictors (for MIMIC model).
ph1	Correlations between latent predictors (for MIMIC model).
b1	Coefficients of latent predictors (for MIMIC model).
ilvl	Specified levels of all items (i.e., need to specify Item 1 to $J + P$); Any value smaller than 2 is considered as continuous item.
cati	The set of polytomous items in sequence number (i.e., can be any number set in between 1 and $J + P$); NULL for no and -1 for all (if `ilvl=NULL`).
noc	Number of levels for polytomous items.
misp	Proportion of missingness.
ome_out	Output factor score or not.
necw	Number of within-factor local dependence.

necb	Number of between-factor local dependence.
add_ind	(Additional) minor factor with cross-loadings.
add_la	Value of cross-loadings on (Additional) minor factor.
add_phi	Correlations between (Additional) minor factor and other factors.
zero_it	Surplus items with zero loading.
rseed	An integer for the random seed.
digits	Number of significant digits to print when printing numeric values.

Value

An object of class `list` containing the data, loading, and factorial correlation matrix.

Examples

```
# for continuous data with cross-loadings and local dependence effect .3
out <- sim_lvm(N=1000,K=3,J=18,lam = .7, lac=.3,ecr=.3)
summary(out$dat)
out$MLA
out$ofd_ind

# for categorical data with cross-loadings .4 and 10% missingness
out <- sim_lvm(N=1000,K=3,J=18,lam = .7, lac=.4,cati=-1,noc=4,misp=.1)
summary(out$dat)
out$MLA
out$ofd_ind
```

---

summary.lawbl	*Summary method for* lawbl *objects*

Description

Provide summaries of posterior information for a `lawbl` object, .

Usage

```
## S3 method for class 'lawbl'
summary(
  object,
  what = "basic",
  med = FALSE,
  SL = 0.05,
  detail = FALSE,
  digits = 4,
  istart = 1,
  iend = -1,
  ...
)
```

Arguments

`object`	A `lawbl` object
`what`	A list of options for what to summarize.

- `basic`: Basic information about the model and posteriors.
- `lambda`: Loading estimates.
- `qlambda`: Loading estimates in pattern/Q-matrix format.
- `eigen`: Factorial eigen value.
- `dpsx`: Diagonal elements in the residual covariance matrix PSX.
- `offpsx`: Off-diagonal elements in PSX; local dependence terms.
- `phi`: Factorial correlations.
- `thd`: Threshold estimates.
- `int`: Intercept estimates (for pcirm only).
- `shrink`: (Ave) shrinkage for each factor's loadings and LD (if `LD` in `pcfa` = T).
- `factor`: Are the factors true of not (for EFA).
- `all`: All above information.

`med`	logical; if the posterior median (TRUE) or mean (FALSE) is used as the estimate.
`SL`	Significance level for interval estimate. The default is .05.
`detail`	logical; if only significant (FALSE) or all (TRUE) estimates are presented.
`digits`	Number of significant digits to print when printing numeric values.
`istart`	Starting point of the Markov chain for summary.
`iend`	Ending point of the Markov chain for summary; -1 for the actual final point.
`...`	additional arguments

Value

A list or matrix containing the summarized information based on the option `what`.

Examples

```
dat <- sim18cfa0$dat
J <- ncol(dat)
K <- 3
Q<-matrix(-1,J,K);
Q[1:2,1]<-Q[7:8,2]<-Q[13:14,3]<-1

m0 <- pcfa(dat = dat, Q = Q, LD = FALSE,burn = 1000, iter = 1000)
summary(m0) # summarize basic information
summary(m0, what = 'lambda') #summarize significant loadings
summary(m0, what = 'qlambda') #summarize significant loadings in pattern/Q-matrix format
summary(m0, what = 'offpsx') #summarize significant LD terms
```

Index