Package ‘fourPNO’

September 24, 2019

Type Package

Title Bayesian 4 Parameter Item Response Model

Version 1.1.0

Description Estimate Barton & Lord's (1981) <doi:10.1002/j.2333-8504.1981.tb01255.x> four parameter IRT model with lower and upper asymptotes using Bayesian formulation described by Culpepper (2016) <doi:10.1007/s11336-015-9477-6>.

URL https://github.com/tmsalab/fourPNO

BugReports https://github.com/tmsalab/fourPNO/issues

License GPL (>= 2)

Depends R (>= 3.5.0)

Imports Rcpp (>= 1.0.0)

LinkingTo Rcpp, RcppArmadillo (>= 0.9.200)

RoxygenNote 6.1.1

Encoding UTF-8

NeedsCompilation yes

Author Steven Andrew Culpepper [aut, cre, cph] (<https://orcid.org/0000-0003-4226-6176>)

Maintainer Steven Andrew Culpepper <sculpepp@illinois.edu>

Repository CRAN

Date/Publication 2019-09-24 04:40:02 UTC

R topics documented:

Gibbs_2PNO *Gibbs Implementation of 2PNO*

Description

Implement Gibbs 2PNO Sampler

Usage

```
Gibbs_2PNO(Y, mu_xi, Sigma_xi_inv, mu_theta, Sigma_theta_inv, burnin,
  chain_length = 10000L)
```

Arguments

`Y`	A N by J `matrix` of item responses.
`mu_xi`	A two dimensional `vector` of prior item parameter means.
`Sigma_xi_inv`	A two dimensional identity `matrix` of prior item parameter VC matrix.
`mu_theta`	The prior mean for theta.
`Sigma_theta_inv`	The prior inverse variance for theta.
`burnin`	The number of MCMC samples to discard.
`chain_length`	The number of MCMC samples.

Value

Samples from posterior.

Author(s)

Steven Andrew Culpepper

Examples

```
# simulate small 2PNO dataset to demonstrate function
J = 5
N = 100

# Population item parameters
as_t = rnorm(J,mean=2,sd=.5)
bs_t = rnorm(J,mean=0,sd=.5)

# Sampling gs and ss with truncation
gs_t = rbeta(J,1,8)
ps_g = pbeta(1-gs_t,1,8)
ss_t = qbeta(runif(J)*ps_g,1,8)
theta_t = rnorm(N)
Y_t = Y_4pno_simulate(N,J,as=as_t,bs=bs_t,gs=gs_t,ss=ss_t,theta=theta_t)
```

```
# Setting prior parameters
mu_theta = 0
Sigma_theta_inv = 1
mu_xi = c(0,0)
alpha_c = alpha_s = beta_c = beta_s = 1
Sigma_xi_inv = solve(2*matrix(c(1,0,0,1), 2, 2))
burnin = 1000

# Execute Gibbs sampler. This should take about 15.5 minutes
out_t = Gibbs_4PNO(Y_t,mu_xi,Sigma_xi_inv,mu_theta,Sigma_theta_inv,
                    alpha_c,beta_c,alpha_s, beta_s,burnin,
                    rep(1,J),rep(1,J),gwg_reps=5,chain_length=burnin*2)

# Summarizing posterior distribution
OUT = cbind(
    apply(out_t$AS[, -c(1:burnin)], 1, mean),
    apply(out_t$BS[, -c(1:burnin)], 1, mean),
    apply(out_t$GS[, -c(1:burnin)], 1, mean),
    apply(out_t$SS[, -c(1:burnin)], 1, mean),
    apply(out_t$AS[, -c(1:burnin)], 1, sd),
    apply(out_t$BS[, -c(1:burnin)], 1, sd),
    apply(out_t$GS[, -c(1:burnin)], 1, sd),
    apply(out_t$SS[, -c(1:burnin)], 1, sd)
)
OUT = cbind(1:J, OUT)
colnames(OUT) = c('Item','as','bs','gs','ss','as_sd','bs_sd',
                  'gs_sd','ss_sd')
print(OUT, digits = 3)
```

Gibbs_4PNO *Gibbs Implementation of 4PNO*

Description

Internal function to -2LL

Usage

```
Gibbs_4PNO(Y, mu_xi, Sigma_xi_inv, mu_theta, Sigma_theta_inv, alpha_c,
  beta_c, alpha_s, beta_s, burnin, cTF, sTF, gwg_reps,
  chain_length = 10000L)
```

Arguments

Y	A N by J `matrix` of item responses.
mu_xi	A two dimensional `vector` of prior item parameter means.
Sigma_xi_inv	A two dimensional identity `matrix` of prior item parameter VC matrix.
mu_theta	The prior mean for theta.

`Sigma_theta_inv`
: The prior inverse variance for theta.

`alpha_c`
: The lower asymptote prior 'a' parameter.

`beta_c`
: The lower asymptote prior 'b' parameter.

`alpha_s`
: The upper asymptote prior 'a' parameter.

`beta_s`
: The upper asymptote prior 'b' parameter.

`burnin`
: The number of MCMC samples to discard.

`cTF`
: A J dimensional `vector` indicating which lower asymptotes to estimate. 0 = exclude lower asymptote and 1 = include lower asymptote.

`sTF`
: A J dimensional `vector` indicating which upper asymptotes to estimate. 0 = exclude upper asymptote and 1 = include upper asymptote.

`gwg_reps`
: The number of Gibbs within Gibbs MCMC samples for marginal distribution of gamma. Values between 5 to 10 are adequate.

`chain_length`
: The number of MCMC samples.

Value

Samples from posterior.

Author(s)

Steven Andrew Culpepper

Examples

```
# Simulate small 4PNO dataset to demonstrate function
J = 5
N = 100

# Population item parameters
as_t = rnorm(J,mean=2,sd=.5)
bs_t = rnorm(J,mean=0,sd=.5)

# Sampling gs and ss with truncation
gs_t = rbeta(J,1,8)
ps_g = pbeta(1-gs_t,1,8)
ss_t = qbeta(runif(J)*ps_g,1,8)
theta_t <- rnorm(N)
Y_t = Y_4pno_simulate(N,J,as=as_t,bs=bs_t,gs=gs_t,ss=ss_t,theta=theta_t)

# Setting prior parameters
mu_theta=0
Sigma_theta_inv=1
mu_xi = c(0,0)
alpha_c=alpha_s=beta_c=beta_s=1
Sigma_xi_inv = solve(2*matrix(c(1,0,0,1),2,2))
burnin = 1000

# Execute Gibbs sampler
```

```
out_t = Gibbs_4PNO(Y_t,mu_xi,Sigma_xi_inv,mu_theta,
                   Sigma_theta_inv,alpha_c,beta_c,alpha_s,
                   beta_s,burnin,rep(1,J),rep(1,J),
                   gwg_reps=5,chain_length=burnin*2)

# Summarizing posterior distribution
OUT = cbind(apply(out_t$AS[,-c(1:burnin)],1,mean),
            apply(out_t$BS[,-c(1:burnin)],1,mean),
            apply(out_t$GS[,-c(1:burnin)],1,mean),
            apply(out_t$SS[,-c(1:burnin)],1,mean),
            apply(out_t$AS[,-c(1:burnin)],1,sd),
            apply(out_t$BS[,-c(1:burnin)],1,sd),
            apply(out_t$GS[,-c(1:burnin)],1,sd),
            apply(out_t$SS[,-c(1:burnin)],1,sd) )

OUT = cbind(1:J,OUT)
colnames(OUT) = c('Item', 'as', 'bs', 'gs', 'ss', 'as_sd', 'bs_sd',
                  'gs_sd', 'ss_sd')
print(OUT, digits = 3)
```

`min2LL_4pno` *Compute 4PNO Deviance*

Description

Internal function to -2LL

Usage

```
min2LL_4pno(N, J, Y, as, bs, gs, ss, theta)
```

Arguments

`N`	An `int`, which gives the number of observations. (> 0)
`J`	An `int`, which gives the number of items. (> 0)
`Y`	A N by J `matrix` of item responses.
`as`	A `vector` of item discrimination parameters.
`bs`	A `vector` of item threshold parameters.
`gs`	A `vector` of item lower asymptote parameters.
`ss`	A `vector` of item upper asymptote parameters.
`theta`	A `vector` of prior thetas.

Value

-2LL.

Author(s)

Steven Andrew Culpepper

See Also

`Gibbs_4PNO()`

`rmvnorm` *Generate Random Multivariate Normal Distribution*

Description

Creates a random Multivariate Normal when given number of obs, mean, and sigma.

Usage

```
rmvnorm(n, mu, sigma)
```

Arguments

`n`	An `int`, which gives the number of observations. (> 0)
`mu`	A `vector` length m that represents the means of the normals.
`sigma`	A `matrix` with dimensions m x m that provides the covariance matrix.

Value

A `matrix` that is a Multivariate Normal distribution

Author(s)

James J Balamuta

Examples

```
# Call with the following data:
rmvnorm(2, c(0,0), diag(2))
```

Total_Tabulate — *Calculate Tabulated Total Scores*

Description

Internal function to -2LL

Usage

```
Total_Tabulate(N, J, Y)
```

Arguments

N	An `int`, which gives the number of observations. (> 0)
J	An `int`, which gives the number of items. (> 0)
Y	A N by J `matrix` of item responses.

Value

A vector of tabulated total scores.

Author(s)

Steven Andrew Culpepper

See Also

`Gibbs_4PNO()`

Y_4pno_simulate — *Simulate from 4PNO Model*

Description

Generate item responses under the 4PNO

Usage

```
Y_4pno_simulate(N, J, as, bs, gs, ss, theta)
```

Arguments

N	An int, which gives the number of observations. (> 0)
J	An int, which gives the number of items. (> 0)
as	A vector of item discrimination parameters.
bs	A vector of item threshold parameters.
gs	A vector of item lower asymptote parameters.
ss	A vector of item upper asymptote parameters.
theta	A vector of prior thetas.

Value

A N by J matrix of dichotomous item responses.

Author(s)

Steven Andrew Culpepper

See Also

Gibbs_4PNO()

Index