Package ‘ltbayes’

October 29, 2016

Type Package

Title Simulation-Based Bayesian Inference for Latent Traits of Item Response Models

Version 0.4

Date 2016-10-28

Author Timothy R. Johnson

Maintainer Timothy R. Johnson <trjohns@uidaho.edu>

Description Functions for simulating realizations from the posterior distribution of a latent trait of an item response model. Distributions are conditional on one or a subset of response patterns (e.g., sum scores). Functions for computing likelihoods, Fisher and observed information, posterior modes, and profile likelihood confidence intervals are also included. These functions are designed to be easily amenable to user-specified models.

Depends R (>= 3.3.1), mcmc (>= 0.9-4), MHadaptive (>= 1.1-8), numDeriv (>= 2016.8-1)

Suggests coda, mcmcplots, ggmcmc, knitr, ggplot2, gridExtra, reshape2

License GPL (>= 2)

VignetteBuilder knitr

NeedsCompilation yes

Repository CRAN

Date/Publication 2016-10-29 00:06:34

R topics documented:

fmodel1pl 2
fmodel1pp 4
fmodel2pl 5
fmodel2pp 7
fmodel3pl 9
fmodel3pp 11
fmodel4pl 12

fmodel4pp . . . 14
fmodelgrl . . . 16
fmodelgrp . . . 18
fmodelnrm . . . 20
fmodelpcm . . . 22
fmodelrsm . . . 24
fmodelseq . . . 26
information . . . 27
patterns . . . 29
postmode . . . 30
postsamp . . . 32
posttrace . . . 33
profileci . . . 35

Index **37**

`fmodel1pl` *Latent Trait Posterior of the One-Parameter Binary Logistic Model*

Description

`fmodel1pl` evaluates the (unnormalized) posterior density of the latent trait of a one-parameter binary logistic item response model with given prior distribution, and computes the probabilities for each item and response category given the latent trait.

Usage

```
fmodel1pl(zeta, y, bpar, prior = dnorm, ...)
```

Arguments

`zeta`	Scalar or vector of latent trait values.
`y`	Vector of length m for a single response pattern, or matrix of size s by m of a set of s item response patterns. In the latter case the posterior is computed by conditioning on the event that the response pattern is one of the s response patterns. Elements of `y` should be 0 or 1.
`bpar`	Vector of m "difficulty" parameters.
`prior`	Function that evaluates the prior distribution of the latent trait. The default is a standard normal distribution.
`...`	Additional arguments to be passed to `prior`.

Details

The item response model is parameterized as

$$P(Y_{ij} = 1|\zeta_i) = 1/(1 + \exp(-(\zeta_i - \beta_j))),$$

where β_j is the difficulty parameter (`bpar`) and ζ_i is the latent trait (`zeta`).

Value

post	The log of the unnormalized posterior distribution evaluated at zeta.
prob	Matrix of size m by 2 array of item response probabilities.

Note

This function is designed to be called by other functions in the **ltbayes** package, but could be useful on its own. This function calls fmodel4pl since it is a special case.

Author(s)

Timothy R. Johnson

See Also

See fmodel2pl, fmodel3pl, and fmodel4pl for related models, and fmodel1pp for the probit variant of this model.

Examples

```
samp <- 5000 # samples from posterior distribution
burn <- 1000 # burn-in samples to discard

beta <- -2:2        # difficulty parameters

post <- postsamp(fmodel1pl, c(1,1,0,0,0),
bpar = beta, control = list(nbatch = samp + burn))

post <- data.frame(sample = 1:samp,
zeta = post$batch[(burn + 1):(samp + burn)])

with(post, plot(sample, zeta), type = "l")  # trace plot of sampled realizations
with(post, plot(density(zeta, adjust = 2))) # density estimate of posterior distribution

with(posttrace(fmodel1pl, c(1,1,0,0,0),
bpar = beta),plot(zeta, post, type = "l")) # profile of log-posterior density

information(fmodel1pl, c(1,1,0,0,0), bpar = beta) # Fisher information

with(post, mean(zeta)) # posterior mean
postmode(fmodel1pl, c(1,1,0,0,0), bpar = beta) # posterior mode

with(post, quantile(zeta, probs = c(0.025, 0.975))) # posterior credibility interval
profileci(fmodel1pl, c(1,1,0,0,0), bpar = beta) # profile likelihood confidence interval
```

fmodel1pp — *Latent Trait Posterior of the One-Parameter Binary Probit Model*

Description

fmodel1pp evaluates the (unnormalized) posterior density of the latent trait of a one-parameter binary probit item response model with given prior distribution, and computes the probabilities for each item and response category given the latent trait.

Usage

```
fmodel1pp(zeta, y, bpar, prior = dnorm, ...)
```

Arguments

zeta	Latent trait value.
y	Vector of length m for a single response pattern, or matrix of size s by m of a set of s item response patterns. In the latter case the posterior is computed by conditioning on the event that the response pattern is one of the s response patterns. Elements of y should be 0 or 1.
bpar	Vector of m "difficulty" parameters.
prior	Function that evaluates the prior distribution of the latent trait. The default is a standard normal distribution.
...	Additional arguments to be passed to prior.

Details

The item response model is parameterized as

$$P(Y_{ij} = 1|\zeta_i) = \Phi(\zeta_i - \beta_j),$$

where where β_j is the difficulty parameter (bpar), ζ_i is the latent trait (zeta), and Φ is the distribution function of a standard normal distribution.

Value

post	The log of the unnormalized posterior distribution evaluated at zeta.
prob	Matrix of size m by 2 array of item response probabilities.

Note

This function is designed to be called by other functions in the **ltbayes** package, but could be useful on its own. This function calls fmodel4pp since it is a special case.

Author(s)

Timothy R. Johnson

See Also

See fmodel2pp, fmodel3pp, and fmodel4pp for related models, and fmodel1pl for the logit variant of this model.

Examples

```
samp <- 5000 # samples from posterior distribution
burn <- 1000 # burn-in samples to discard

beta <- -2:2       # difficulty parameters

post <- postsamp(fmodel1pp, c(1,1,0,0,0),
bpar = beta, control = list(nbatch = samp + burn))

post <- data.frame(sample = 1:samp,
zeta = post$batch[(burn + 1):(samp + burn)])

with(post, plot(sample, zeta), type = "l")  # trace plot of sampled realizations
with(post, plot(density(zeta, adjust = 2))) # density estimate of posterior distribution

with(posttrace(fmodel1pp, c(1,1,0,0,0),
bpar = beta),plot(zeta, post, type = "l")) # profile of log-posterior density

information(fmodel1pp, c(1,1,0,0,0), bpar = beta) # Fisher information

with(post, mean(zeta)) # posterior mean
postmode(fmodel1pp, c(1,1,0,0,0), bpar = beta) # posterior mode

with(post, quantile(zeta, probs = c(0.025, 0.975))) # posterior credibility interval
profileci(fmodel1pp, c(1,1,0,0,0), bpar = beta) # profile likelihood confidence interval
```

---

fmodel2pl *Latent Trait Posterior of the Two-Parameter Binary Logistic Model*

Description

fmodel2pl evaluates the (unnormalized) posterior density of the latent trait of a two-parameter binary logistic item response model with given prior distribution, and computes the probabilities for each item and response category given the latent trait.

Usage

```
fmodel2pl(zeta, y, apar, bpar, prior = dnorm, ...)
```

Arguments

zeta Latent trait value.

y	Vector of length m for a single response pattern, or matrix of size s by m of a set of s item response patterns. In the latter case the posterior is computed by conditioning on the event that the response pattern is one of the s response patterns. Elements of y should be 0 or 1.
apar	Vector of m "discrimination" parameters.
bpar	Vector of m "difficulty" parameters.
prior	Function that evaluates the prior distribution of the latent trait. The default is a standard normal distribution.
...	Additional arguments to be passed to prior.

Details

The item response model is parameterized as

$$P(Y_{ij} = 1|\zeta_i) = 1/(1 + \exp(-\alpha_j(\zeta_i - \beta_j))),$$

where α_j is the discrimination parameter (apar), β_j is the difficulty parameter (bpar), and ζ_i is the latent trait (zeta).

Value

post	The log of the unnormalized posterior distribution evaluated at zeta.
prob	Matrix of size m by 2 array of item response probabilities.

Note

This function is designed to be called by other functions in the **ltbayes** package, but could be useful on its own. This function calls fmodel4pl since it is a special case.

Author(s)

Timothy R. Johnson

See Also

See fmodel1pl, fmodel3pl, and fmodel4pl for related models, and fmodel2pp for a probit variant of this model.

Examples

```
samp <- 5000 # samples from posterior distribution
burn <- 1000 # burn-in samples to discard

alph <- rep(1, 5)   # discrimination parameters
beta <- -2:2        # difficulty parameters

post <- postsamp(fmodel2pl, c(1,1,0,0,0),
apar = alph, bpar = beta,
control = list(nbatch = samp + burn))
```

```
post <- data.frame(sample = 1:samp,
zeta = post$batch[(burn + 1):(samp + burn)])

with(post, plot(sample, zeta), type = "l")  # trace plot of sampled realizations
with(post, plot(density(zeta, adjust = 2))) # density estimate of posterior distribution

with(posttrace(fmodel2pl, c(1,1,0,0,0),
apar = alph, bpar = beta),
plot(zeta, post, type = "l")) # profile of log-posterior density

information(fmodel2pl, c(1,1,0,0,0),
apar = alph, bpar = beta) # Fisher information

with(post, mean(zeta)) # posterior mean
postmode(fmodel2pl, c(1,1,0,0,0),
apar = alph, bpar = beta) # posterior mode

with(post, quantile(zeta, probs = c(0.025, 0.975))) # posterior credibility interval
profileci(fmodel2pl, c(1,1,0,0,0), apar = alph,
bpar = beta) # profile likelihood confidence interval
```

---

fmodel2pp *Latent Trait Posterior of the Two-Parameter Binary Probit Model*

Description

fmodel2pp evaluates the (unnormalized) posterior density of the latent trait of a two-parameter binary probit item response model with given prior distribution, and computes the probabilities for each item and response category given the latent trait.

Usage

```
fmodel2pp(zeta, y, apar, bpar, prior = dnorm, ...)
```

Arguments

zeta	Latent trait value.
y	Vector of length m for a single response pattern, or matrix of size s by m of a set of s item response patterns. In the latter case the posterior is computed by conditioning on the event that the response pattern is one of the s response patterns. Elements of y should be 0 or 1.
apar	Vector of m "discrimination" parameters.
bpar	Vector of m "difficulty" parameters.
prior	Function that evaluates the prior distribution of the latent trait. The default is a standard normal distribution.
...	Additional arguments to be passed to prior.

Details

The item response model is parameterized as

$$P(Y_{ij} = 1|\zeta_i) = \Phi(-\alpha_j(\zeta_i - \beta_j)),$$

where Φ is the distribution function of a standard normal distribution, α_j is the discrimination parameter (`apar`), β_j is the difficulty parameter (`bpar`), and ζ_i is the latent trait (`zeta`).

Value

`post`	The log of the unnormalized posterior distribution evaluated at `zeta`.
`prob`	Matrix of size m by 2 array of item response probabilities.

Note

This function is designed to be called by other functions in the **ltbayes** package, but could be useful on its own. This function calls `fmodel4pp` since it is a special case.

Author(s)

Timothy R. Johnson

See Also

See `fmodel1pp`, `fmodel3pp`, and `fmodel4pp` for related models, and `fmodel2pp` for a probit variant of this model.

Examples

```
samp <- 5000 # samples from posterior distribution
burn <- 1000 # burn-in samples to discard

alph <- rep(1, 5)   # discrimination parameters
beta <- -2:2        # difficulty parameters

post <- postsamp(fmodel2pp, c(1,1,0,0,0),
apar = alph, bpar = beta,
control = list(nbatch = samp + burn))

post <- data.frame(sample = 1:samp,
zeta = post$batch[(burn + 1):(samp + burn)])

with(post, plot(sample, zeta), type = "l")  # trace plot of sampled realizations
with(post, plot(density(zeta, adjust = 2))) # density estimate of posterior distribution

with(posttrace(fmodel2pp, c(1,1,0,0,0),
apar = alph, bpar = beta),
plot(zeta, post, type = "l")) # profile of log-posterior density

information(fmodel2pp, c(1,1,0,0,0),
apar = alph, bpar = beta) # Fisher information
```

```
with(post, mean(zeta)) # posterior mean
postmode(fmodel2pp, c(1,1,0,0,0),
apar = alph, bpar = beta) # posterior mode

with(post, quantile(zeta, probs = c(0.025, 0.975))) # posterior credibility interval
profileci(fmodel2pp, c(1,1,0,0,0), apar = alph,
bpar = beta) # profile likelihood confidence interval
```

fmodel3pl *Latent Trait Posterior of the Three-Parameter Binary Logistic Model*

Description

`fmodel3pl` evaluates the (unnormalized) posterior density of the latent trait of a three-parameter binary logistic item response model with given prior distribution, and computes the probabilities for each item and response category given the latent trait.

Usage

```
fmodel3pl(zeta, y, apar, bpar, cpar, prior = dnorm, ...)
```

Arguments

`zeta`	Latent trait value.
`y`	Vector of length m for a single response pattern, or matrix of size s by m of a set of s item response patterns. In the latter case the posterior is computed by conditioning on the event that the response pattern is one of the s response patterns. Elements of `y` should be 0 or 1.
`apar`	Vector of m "discrimination" parameters.
`bpar`	Vector of m "difficulty" parameters.
`cpar`	Vector of m lower asymptote (i.e., "guessing") parameters.
`prior`	Function that evaluates the prior distribution of the latent trait. The default is a standard normal distribution.
`...`	Additional arguments to be passed to `prior`.

Details

The item response model is parameterized as

$$P(Y_{ij} = 1|\zeta_i) = \gamma_j + (1 - \gamma_j)/(1 + \exp(-\alpha_j(\zeta_i - \beta_j))),$$

where α_j is the discrimination parameter (`apar`), β_j is the difficulty parameter (`bpar`), $0 \leq \gamma_j < 1$ is the lower asymptote parameter (`cpar`), and ζ_i is the latent trait (`zeta`).

Value

post	The log of the unnormalized posterior distribution evaluated at `zeta`.

prob	Matrix of size m by 2 array of item response probabilities.

Note

This function is designed to be called by other functions in the **ltbayes** package, but could be useful on its own. This function calls `fmodel4pl` since it is a special case.

Author(s)

Timothy R. Johnson

See Also

See `fmodel1pl`, `fmodel2pl`, and `fmodel4pl` for related models, and `fmodel3pp` for a probit variant of this model.

Examples

```
samp <- 5000 # samples from posterior distribution
burn <- 1000 # burn-in samples to discard

alph <- c(1.27,1.34,1.14,1,0.67)   # discrimination parameters
beta <- c(1.19,0.59,0.15,-0.59,-2) # difficulty parameters
gamm <- c(0.1,0.15,0.15,0.2,0.01)  # lower asymptote parameters

post <- postsamp(fmodel3pl, c(0,0,1,1,1),
apar = alph, bpar = beta, cpar = gamm,
control = list(nbatch = samp + burn, scale = 3))

post <- data.frame(sample = 1:samp,
zeta = post$batch[(burn + 1):(samp + burn)])

with(post, plot(sample, zeta), type = "l")  # trace plot of sampled realizations
with(post, plot(density(zeta, adjust = 2))) # density estimate of posterior distribution

with(posttrace(fmodel3pl, c(0,0,1,1,1), apar = alph, bpar = beta, cpar = gamm),
plot(zeta, post, type = "l")) # profile of log-posterior density

information(fmodel3pl, c(0,0,1,1,1), apar = alph, bpar = beta, cpar = gamm) # Fisher information

with(post, mean(zeta)) # posterior mean
postmode(fmodel3pl, c(0,0,1,1,1), apar = alph, bpar = beta, cpar = gamm) # posterior mode

with(post, quantile(zeta, probs = c(0.025, 0.975))) # posterior credibility interval
profileci(fmodel3pl, c(0,0,1,1,1), apar = alph,
bpar = beta, cpar = gamm) # profile likelihood confidence interval
```

`fmodel3pp` *Latent Trait Posterior of the Three-Parameter Binary Probit Model*

Description

`fmodel3pp` evaluates the (unnormalized) posterior density of the latent trait of a three-parameter binary probit item response model with given prior distribution, and computes the probabilities for each item and response category given the latent trait.

Usage

```
fmodel3pp(zeta, y, apar, bpar, cpar, prior = dnorm, ...)
```

Arguments

`zeta`	Latent trait value.
`y`	Vector of length m for a single response pattern, or matrix of size s by m of a set of s item response patterns. In the latter case the posterior is computed by conditioning on the event that the response pattern is one of the s response patterns. Elements of `y` should be 0 or 1.
`apar`	Vector of m "discrimination" parameters.
`bpar`	Vector of m "difficulty" parameters.
`cpar`	Vector of m lower asymptote (i.e., "guessing") parameters.
`prior`	Function that evaluates the prior distribution of the latent trait. The default is a standard normal distribution.
`...`	Additional arguments to be passed to `prior`.

Details

The item response model is parameterized as

$$P(Y_{ij} = 1|\zeta_i) = \gamma_j + (1 - \gamma_j)\Phi(-\alpha_j(\zeta_i - \beta_j)),$$

where Φ is the distribution function of a standard normal distribution, α_j is the discrimination parameter (`apar`), β_j is the difficulty parameter (`bpar`), $0 \leq \gamma_j < 1$ is the lower asymptote parameter (`cpar`), and ζ_i is the latent trait (`zeta`).

Value

`post`	The log of the unnormalized posterior distribution evaluated at `zeta`.
`prob`	Matrix of size m by 2 array of item response probabilities.

Note

This function is designed to be called by other functions in the **ltbayes** package, but could be useful on its own. This function calls `fmodel4pp` since it is a special case.

Author(s)

Timothy R. Johnson

See Also

See fmodel1pp, fmodel2pp, and fmodel4pp for related models, and fmodel3pl for a logit variant of this model.

Examples

```
samp <- 5000 # samples from posterior distribution
burn <- 1000 # burn-in samples to discard

alph <- rep(1, 5)   # discrimination parameters
beta <- -2:2        # difficulty parameters
gamm <- rep(0.1, 5) # lower asymptote parameters

post <- postsamp(fmodel3pp, c(1,1,0,0,0),
apar = alph, bpar = beta, cpar = gamm,
control = list(nbatch = samp + burn))

post <- data.frame(sample = 1:samp,
zeta = post$batch[(burn + 1):(samp + burn)])

with(post, plot(sample, zeta), type = "l")  # trace plot of sampled realizations
with(post, plot(density(zeta, adjust = 2))) # density estimate of posterior distribution

with(posttrace(fmodel3pp, c(1,1,0,0,0),
apar = alph, bpar = beta, cpar = gamm),
plot(zeta, post, type = "l")) # profile of log-posterior density

information(fmodel3pp, c(1,1,0,0,0),
apar = alph, bpar = beta, cpar = gamm) # Fisher information

with(post, mean(zeta)) # posterior mean
postmode(fmodel3pp, c(1,1,0,0,0),
apar = alph, bpar = beta, cpar = gamm) # posterior mode

with(post, quantile(zeta, probs = c(0.025, 0.975))) # posterior credibility interval
profileci(fmodel3pp, c(1,1,0,0,0), apar = alph,
bpar = beta, cpar = gamm) # profile likelihood confidence interval
```

---

fmodel4pl *Latent Trait Posterior of the Four-Parameter Binary Logistic Model*

Description

fmodel4pl evaluates the (unnormalized) posterior density of the latent trait of a four-parameter binary logistic item response model with given prior distribution, and computes the probabilities for each item and response category given the latent trait.

Usage

```
fmodel4pl(zeta, y, apar, bpar, cpar, dpar, prior = dnorm, ...)
```

Arguments

`zeta`	Latent trait value.
`y`	Vector of length m for a single response pattern, or matrix of size s by m of a set of s item response patterns. In the latter case the posterior is computed by conditioning on the event that the response pattern is one of the s response patterns. Elements of `y` should be 0 or 1.
`apar`	Vector of m "discrimination" parameters.
`bpar`	Vector of m "difficulty" parameters.
`cpar`	Vector of m lower asymptote parameters.
`dpar`	Vector of m upper asymptote parameters.
`prior`	Function that evaluates the prior distribution of the latent trait. The default is a standard normal distribution.
`...`	Additional arguments to be passed to `prior`.

Details

The item response model is parameterized as

$$P(Y_{ij} = 1|\zeta_i) = \gamma_j + (\delta_j - \gamma_j)/(1 + \exp(-\alpha_j(\zeta_i - \beta_j))),$$

where α_j is the discrimination parameter (`apar`), β_j is the difficulty parameter (`bpar`), $0 \leq \gamma_j < 1$ is the lower asymptote parameter (`cpar`), $0 < \delta_j \leq 1$ is the upper asymptote parameter (`dpar`), and ζ_i is the latent trait (`zeta`). The four-parameter binary logistic model is discussed by Barton and Lord (1981).

Value

`post`	The log of the unnormalized posterior distribution evaluated at `zeta`.
`prob`	Matrix of size m by 2 array of item response probabilities.

Note

This function is designed to be called by other functions in the **ltbayes** package, but could be useful on its own.

Author(s)

Timothy R. Johnson

References

Barton, M. A. & Lord, R. M. (1981). An upper asymptote for the three-parameter logistic item-response model. New Jersey: Educational Testing Service.

See Also

See fmodel1pl, fmodel2pl, and fmodel3pl for related models, and fmodel4pp for a probit variant of this model.

Examples

```
samp <- 5000 # samples from posterior distribution
burn <- 1000 # burn-in samples to discard

alph <- rep(1, 5)   # discrimination parameters
beta <- -2:2        # difficulty parameters
gamm <- rep(0.1, 5) # lower asymptote parameters
delt <- rep(0.9, 5) # upper asymptote parameters

post <- postsamp(fmodel4pl, c(1,1,0,0,0),
apar = alph, bpar = beta, cpar = gamm, dpar = delt,
control = list(nbatch = samp + burn))

post <- data.frame(sample = 1:samp,
zeta = post$batch[(burn + 1):(samp + burn)])

with(post, plot(sample, zeta), type = "l")  # trace plot of sampled realizations
with(post, plot(density(zeta, adjust = 2))) # density estimate of posterior distribution

with(posttrace(fmodel4pl, c(1,1,0,0,0),
apar = alph, bpar = beta, cpar = gamm, dpar = delt),
plot(zeta, post, type = "l")) # profile of log-posterior density

information(fmodel4pl, c(1,1,0,0,0),
apar = alph, bpar = beta, cpar = gamm, dpar = delt) # Fisher information

with(post, mean(zeta)) # posterior mean
postmode(fmodel4pl, c(1,1,0,0,0),
apar = alph, bpar = beta, cpar = gamm, dpar = delt) # posterior mode

with(post, quantile(zeta, probs = c(0.025, 0.975))) # posterior credibility interval
profileci(fmodel4pl, c(1,1,0,0,0), apar = alph,
bpar = beta, cpar = gamm, dpar = delt) # profile likelihood confidence interval
```

---

fmodel4pp *Latent Trait Posterior of the Four-Parameter Binary Probit Model*

Description

fmodel4pp evaluates the (unnormalized) posterior density of the latent trait of a four-parameter binary probit item response model with given prior distribution, and computes the probabilities for each item and response category given the latent trait.

Usage

```
fmodel4pp(zeta, y, apar, bpar, cpar, dpar, prior = dnorm, ...)
```

Arguments

`zeta`	Latent trait value.
`y`	Vector of length m for a single response pattern, or matrix of size s by m of a set of s item response patterns. In the latter case the posterior is computed by conditioning on the event that the response pattern is one of the s response patterns. Elements of `y` should be 0 or 1.
`apar`	Vector of m "discrimination" parameters.
`bpar`	Vector of m "difficulty" parameters.
`cpar`	Vector of m lower asymptote parameters.
`dpar`	Vector of m upper asymptote parameters.
`prior`	Function that evaluates the prior distribution of the latent trait. The default is a standard normal distribution.
`...`	Additional arguments to be passed to `prior`.

Details

The item response model is parameterized as

$$P(Y_{ij} = 1|\zeta_i) = \gamma_j + (\delta_j - \gamma_j)\Phi(\alpha_j(\zeta_i - \beta_j)),$$

where α_j is the discrimination parameter (`apar`), β_j is the difficulty parameter (`bpar`), $0 \leq \gamma_j < 1$ is the lower asymptote parameter (`cpar`), $0 < \delta_j \leq 1$ is the upper asymptote parameter (`dpar`), ζ_i is the latent trait (`zeta`), and Φ is the distribution function of the standard normal distribution. This model is a variant of the logistic model discussed by Barton and Lord (1981).

Value

`post`	The log of the unnormalized posterior distribution evaluated at `zeta`.
`prob`	Matrix of size m by 2 array of item response probabilities.

Note

This function is designed to be called by other functions in the **ltbayes** package, but could be useful on its own.

Author(s)

Timothy R. Johnson

References

Barton, M. A. & Lord, R. M. (1981). An upper asymptote for the three-parameter logistic item-response model. New Jersey: Educational Testing Service.

See Also

See fmodel1pp, fmodel2pp, and fmodel3pp for related models, and fmodel4pl for the logistic variant of this model.

Examples

```
samp <- 5000 # samples from posterior distribution
burn <- 1000 # burn-in samples to discard

alph <- rep(1, 5)   # discrimination parameters
beta <- -2:2        # difficulty parameters
gamm <- rep(0.1, 5) # lower asymptote parameters
delt <- rep(0.9, 5) # upper asymptote parameters

post <- postsamp(fmodel4pp, c(1,1,0,0,0),
apar = alph, bpar = beta, cpar = gamm, dpar = delt,
control = list(nbatch = samp + burn))

post <- data.frame(sample = 1:samp,
zeta = post$batch[(burn + 1):(samp + burn)])

with(post, plot(sample, zeta), type = "l")  # trace plot of sampled realizations
with(post, plot(density(zeta, adjust = 2))) # density estimate of posterior distribution

with(posttrace(fmodel4pp, c(1,1,0,0,0),
apar = alph, bpar = beta, cpar = gamm, dpar = delt),
plot(zeta, post, type = "l")) # profile of log-posterior density

information(fmodel4pp, c(1,1,0,0,0),
apar = alph, bpar = beta, cpar = gamm, dpar = delt) # Fisher information

with(post, mean(zeta)) # posterior mean
postmode(fmodel4pp, c(1,1,0,0,0),
apar = alph, bpar = beta, cpar = gamm, dpar = delt) # posterior mode

with(post, quantile(zeta, probs = c(0.025, 0.975))) # posterior credibility interval
profileci(fmodel4pp, c(1,1,0,0,0), apar = alph,
bpar = beta, cpar = gamm, dpar = delt) # profile likelihood confidence interval
```

---

fmodelgrl *Latent Trait Posterior of the Logistic Graded Response Model*

Description

fmodelgrl evaluates the (unnormalized) posterior density of the latent trait of a logistic graded response model with a given prior distribution, and computes the probability for each item and response category given the latent trait.

Usage

```
fmodelgrl(zeta, y, apar, bpar, prior = dnorm, ...)
```

Arguments

`zeta`	Latent trait value.
`y`	Vector of length m for a single response pattern, or matrix of size s by m of a set of s item response patterns. In the latter case the posterior is computed by conditioning on the event that the response pattern is one of the s response patterns. Elements of `y` should be integers from 0 to r-1 where r is the number of response categories.
`apar`	Vector of length m of "discrimination" parameters.
`bpar`	Matrix of size m by r-1 of "difficulty" parameters.
`prior`	Function that evaluates the prior distribution of the latent trait. The default is the standard normal distribution.
`...`	Additional arguments to be passed to `prior`.

Details

The parameterization of the graded response model used here is

$$P(Y_{ij} \geq y|\zeta_i) = 1/(1 + \exp(-\alpha_j(\zeta_i - \beta_{jy})))$$

for $y = 1, \ldots, r-1$, where α_j and β_{jk} are the "discrimination" and "difficulty" parameters, respectively, for the k-th cumulative item response function. Note that the difficulty parameters must meet the constraint that $\beta_{j,k+1} \geq \beta_{jk}$ for $k = 1, \ldots, r-1$ so that all item response probabilities are non-negative. This model was first proposed by Samejima (1969, 1972).

Value

`post`	The log of the unnormalized posterior distribution evaluated at `zeta`.
`prob`	Matrix of size m by 2 array of item response probabilities.

Author(s)

Timothy R. Johnson

References

Samejima, F. (1969). Estimation of ability using a response pattern of graded scores. *Psychometrikka Monograph*, No. 17.

Samejima, F. (1972). A general model for free-response data. *Psychometrika Monograph*, No. 18.

See Also

See `fmodelgrp` for the probit variant of this model.

Examples

```
samp <- 5000 # samples from posterior distribution
burn <- 1000 # burn-in samples to discard

alph <- rep(1, 3)                          # discrimination parameters
beta <- matrix(c(-1,1), 3, 2, byrow = TRUE) # difficulty parameters

post <- postsamp(fmodelgrl, c(0,1,2),
apar = alph, bpar = beta,
control = list(nbatch = samp + burn))

post <- data.frame(sample = 1:samp,
zeta = post$batch[(burn + 1):(samp + burn)])

with(post, plot(sample, zeta), type = "l")  # trace plot of sampled realizations
with(post, plot(density(zeta, adjust = 2))) # density estimate of posterior distribution

with(posttrace(fmodelgrl, c(0,1,2),
apar = alph, bpar = beta),
plot(zeta, post, type = "l")) # profile of log-posterior density

information(fmodelgrl, c(0,1,2),
apar = alph, bpar = beta) # Fisher information

with(post, mean(zeta)) # posterior mean
postmode(fmodelgrl, c(0,1,2),
apar = alph, bpar = beta) # posterior mode

with(post, quantile(zeta, probs = c(0.025, 0.975))) # posterior credibility interval
profileci(fmodelgrl, c(0,1,2), apar = alph,
bpar = beta) # profile likelihood confidence interval
```

`fmodelgrp` *Latent Trait Posterior of the Probit Graded Response Model*

Description

`fmodelgrp` evaluates the (unnormalized) posterior density of the latent trait of a probit graded response model with a given prior distribution, and computes the probability for each item and response category given the latent trait.

Usage

```
fmodelgrp(zeta, y, apar, bpar, prior = dnorm, ...)
```

Arguments

`zeta`	Latent trait value.

`y`	Vector of length m for a single response pattern, or matrix of size s by m of a set of s item response patterns. In the latter case the posterior is computed by conditioning on the event that the response pattern is one of the s response patterns. Elements of `y` should be integers from 0 to r-1 where r is the number of response categories.
`apar`	Vector of length m of "discrimination" parameters.
`bpar`	Matrix of size m by r-1 of "difficulty" parameters.
`prior`	Function that evaluates the prior distribution of the latent trait. The default is the standard normal distribution.
`...`	Additional arguments to be passed to `prior`.

Details

The parameterization of the graded response model used here is

$$P(Y_{ij} \geq y|\zeta_i) = \Phi(\alpha_j(\zeta_i - \beta_{jy}))$$

for $y = 1, \ldots, r-1$, where α_j and β_{jk} are the "discrimination" and "difficulty" parameters, respectively, for the k-th cumulative item response function, and Φ is the distribution function of a standard normal distribution. Note that the difficulty parameters must meet the constraint that $\beta_{j,k+1} \geq \beta_{jk}$ for $k = 1, \ldots, r-1$ so that all item response probabilities are non-negative. This model was first proposed by Samejima (1969, 1972).

Value

`post`	The log of the unnormalized posterior distribution evaluated at `zeta`.
`prob`	Matrix of size m by 2 array of item response probabilities.

Note

This function is designed to be called by other functions in the **ltbayes** package, but could be useful on its own.

Author(s)

Timothy R. Johnson

References

Samejima, F. (1969). Estimation of ability using a response pattern of graded scores. *Psychometrikka Monograph*, No. 17.

Samejima, F. (1972). A general model for free-response data. *Psychometrika Monograph*, No. 18.

See Also

See `fmodelgrl` for the logit variant of this model.

Examples

```
samp <- 5000 # samples from posterior distribution
burn <- 1000 # burn-in samples to discard

alph <- rep(1, 3)                          # discrimination parameters
beta <- matrix(c(-1,1), 3, 2, byrow = TRUE) # difficulty parameters

post <- postsamp(fmodelgrp, c(0,1,2),
apar = alph, bpar = beta,
control = list(nbatch = samp + burn))

post <- data.frame(sample = 1:samp,
zeta = post$batch[(burn + 1):(samp + burn)])

with(post, plot(sample, zeta), type = "l")  # trace plot of sampled realizations
with(post, plot(density(zeta, adjust = 2))) # density estimate of posterior distribution

with(posttrace(fmodelgrp, c(0,1,2),
apar = alph, bpar = beta),
plot(zeta, post, type = "l")) # profile of log-posterior density

information(fmodelgrp, c(0,1,2),
apar = alph, bpar = beta) # Fisher information

with(post, mean(zeta)) # posterior mean
postmode(fmodelgrp, c(0,1,2),
apar = alph, bpar = beta) # posterior mode

with(post, quantile(zeta, probs = c(0.025, 0.975))) # posterior credibility interval
profileci(fmodelgrp, c(0,1,2), apar = alph,
bpar = beta) # profile likelihood confidence interval
```

---

fmodelnrm *Latent Trait Posterior of the Nominal Response Model*

Description

`fmodelnrm` evaluates the (unnormalized) posterior density of the latent trait of the nominal response model with given prior distribution, and computes the probabilities for each item and response category given the latent trait.

Usage

```
fmodelnrm(zeta, y, apar, bpar, prior = dnorm, ...)
```

Arguments

`zeta` Latent trait value.

y	Matrix of size s by m of response patterns such that the posterior is computed by conditioning on the event that the response pattern is one of the s response patterns. For conditioning on a single response pattern s = 1 and so the matrix is 1 by m. Elements of y should be integers from 0 to r-1 where r is the number of response categories.
apar	Matrix of size m by r of "slope" parameters.
bpar	Matrix of size m by r of "intercept" parameters.
prior	Function that evaluates the prior distribution of the latent trait. The default is the standard normal distribution.
...	Additional arguments to be passed to the prior distribution.

Details

The nominal response model is parameterized here as

$$P(Y_{ij} = y|\zeta_i) \propto \exp(\alpha_{jy}\zeta_i + \beta_{jy})$$

where $Y_{ij} = 0, 1, \ldots, r - 1$ and α_{jk} and β_{jk} are the "slope" (apar) and "intercept" (bpar) parameters, respectively. The nominal response model is also sometimes called the nominal categories model and was first proposed by Bock (1972).

Value

post	The log of the unnormalized posterior distribution evaluated at zeta.
prob	Matrix of size m by 2 array of item response probabilities.

Note

When estimating the item parameters, constraints on α_{jk} and β_{jk} are necessary for identification, such as $\alpha_{j0} = 0$ and $\beta_{j0} = 0$, but these are not reflected here since a variety of constraints can be used.

Author(s)

Timothy R. Johnson

References

Bock, R. D. (1972). Estimating item parameters and latent ability when responses are scored in two or more nominal categories. *Psychometrika*, *37*, 29-51.

Examples

```
samp <- 5000 # samples from posterior distribution
burn <- 1000 # burn-in samples to discard

alph <- matrix(c(-1, 0, 1), 5, 3, byrow = TRUE)
beta <- matrix(0, 5, 3)
```

```
post <- postsamp(fmodelnrm, c(0,1,2,1,0),
apar = alph, bpar = beta, control = list(nbatch = samp + burn))

post <- data.frame(sample = 1:samp,
zeta = post$batch[(burn + 1):(samp + burn)])

with(post, plot(sample, zeta), type = "l")  # trace plot of sampled realizations
with(post, plot(density(zeta, adjust = 2))) # density estimate of posterior distribution

with(posttrace(fmodelnrm, c(0,1,2,1,0), apar = alph, bpar = beta),
plot(zeta, post, type = "l")) # profile of log-posterior density

information(fmodelnrm, c(0,1,2,1,0), apar = alph, bpar = beta) # Fisher information

with(post, mean(zeta)) # posterior mean
postmode(fmodelnrm, c(0,1,2,1,0), apar = alph, bpar = beta) # posterior mode

with(post, quantile(zeta, probs = c(0.025, 0.975))) # posterior credibility interval
profileci(fmodelnrm, c(0,1,2,1,0),
   apar = alph, bpar = beta) # profile likelihood confidence interval
```

---

fmodelpcm *Latent Trait Posterior for the Partial Credit Model*

Description

fmodelpcm evaluates the (unnormalized) posterior density of the latent trait of a partial credit item response model with a given prior distribution, and computes the probability for each item and response category given the latent trait.

Usage

```
fmodelpcm(zeta, y, bpar, prior = dnorm, ...)
```

Arguments

zeta	Latent trait value.
y	Vector of length m for a single response pattern, or matrix of size s by m of a set of s item response patterns. In the latter case the posterior is computed by conditioning on the event that the response pattern is one of the s response patterns. Elements of y should be integers from 0 to r-1 where r is the number of response categories.
bpar	Matrix of size m by r-1 of "difficulty" parameters.
prior	Function that evaluates the prior distribution of the latent trait. The default is the standard normal distribution.
...	Additional arguments to be passed to prior.

Details

The parameterization of the partial credit model used here is

$$P(Y_{ij} = y|\zeta_i) \propto \exp(y\zeta_i - \sum_{k=0}^{y} \beta_{jk})$$

for $y = 0, 1, \ldots, r-1$ where $\beta_{j0} = 0$. The β_{jk} are the item "difficulty" parameters and ζ_i is the latent trait. This model was proposed by Masters (1982).

Value

`post`	The log of the unnormalized posterior distribution evaluated at `zeta`.
`prob`	Matrix of size m by 2 array of item response probabilities.

Note

The number of response categories (r) is inferred from the number of columns in `bpar`, not from the maximum value in `y`.

Author(s)

Timothy R. Johnson

References

Masters, G. N. (1982). A Rasch model for partial credit scoring. *Psychometrika*, *47*, 149-174.

See Also

For the rating scale model as a special case use the function `fmodelrsm`.

Examples

```
samp <- 5000 # samples from posterior distribution
burn <- 1000 # burn-in samples to discard

beta <- matrix(0, 5, 2)

post <- postsamp(fmodelpcm, c(0,1,2,1,0), bpar = beta,
control = list(nbatch = samp + burn))

post <- data.frame(sample = 1:samp,
zeta = post$batch[(burn + 1):(samp + burn)])

with(post, plot(sample, zeta), type = "l")  # trace plot of sampled realizations
with(post, plot(density(zeta, adjust = 2))) # density estimate of posterior distribution

with(posttrace(fmodelpcm, c(0,1,2,1,0), bpar = beta),
plot(zeta, post, type = "l")) # profile of log-posterior density
```

```
information(fmodelpcm, c(0,1,2,1,0), bpar = beta) # Fisher information

with(post, mean(zeta)) # posterior mean
postmode(fmodelpcm, c(0,1,2,1,0), bpar = beta) # posterior mode

with(post, quantile(zeta, probs = c(0.025, 0.975))) # posterior credibility interval
profileci(fmodelpcm, c(0,1,2,1,0), bpar = beta) # profile likelihood confidence interval
```

---

fmodelrsm *Latent Trait Posterior of the Rating Scale Model*

Description

fmodelpcm evaluates the (unnormalized) posterior density of the latent trait of a rating scale item response model with a given prior distribution, and computes the probability for each item and response category given the latent trait.

Usage

```
fmodelrsm(zeta, y, cpar, dpar, prior = dnorm, ...)
```

Arguments

zeta	Latent trait value.
y	Vector of length m for a single response pattern, or matrix of size s by m of a set of s item response patterns. In the latter case the posterior is computed by conditioning on the event that the response pattern is one of the s response patterns. Elements of y should be integers from 0 to r-1 where r is the number of response categories.
cpar	Vector of length r-1 of category "threshold" parameters.
dpar	Vector of length m of item "difficulty" parameters.
prior	Function that evaluates the prior distribution of the latent trait. The default is the standard normal distribution.
...	Additional arguments to be passed to prior.

Details

The rating scale model is parameterized here as a special case of the partial credit model where

$$P(Y_{ij} = y|\zeta_i) \propto \exp(y\zeta_i - \sum_{k=0}^{y} \beta_{jk})$$

for $y = 0, 1, \ldots, r-1$ where $\beta_{jk} = \delta_j + \gamma_k$ and $\gamma_0 = 0$. The parameters δ_j and γ_k are the 'difficulty' (dpar) and 'threshold' (cpar) parameters, respectively. This model was proposed by Andersen (1977) and Andrich (1978a, 1978b).

Value

`post`	The log of the unnormalized posterior distribution evaluated at `zeta`.
`prob`	Matrix of size m by 2 array of item response probabilities.

Note

The number of response categories (r) is inferred from the number of columns in `dpar`, not from the maximum value in `y`.

Because the rating scale model can be viewed as a special case of the partial credit model, the latter can be used with `fmodelpcm` to specify more general models.

Author(s)

Timothy R. Johnson

References

Andersen, E. B. (1977). Sufficient statistics and latent trait models. *Psychometrika*, *42*, 69-81.

Andrich, D. (1978a). A rating formulation for ordered response categories. *Psychometrika*, *43*, 561-573.

Andrich, D. (1978b). Application of a psychometric rating scale model to ordered categories which are scored with successive integers. *Applied Psychological Measurement*, *2*, 581-594.

Examples

```
samp <- 5000 # samples from posterior distribution
burn <- 1000 # burn-in samples to discard

gamm <- c(0, 0)
delt <- rep(0, 5)

post <- postsamp(fmodelrsm, c(0,1,2,1,0), cpar = gamm, dpar = delt,
control = list(nbatch = samp + burn))

post <- data.frame(sample = 1:samp,
zeta = post$batch[(burn + 1):(samp + burn)])

with(post, plot(sample, zeta), type = "l")  # trace plot of sampled realizations
with(post, plot(density(zeta, adjust = 2))) # density estimate of posterior distribution

with(posttrace(fmodelrsm, c(0,1,2,1,0), cpar = gamm, dpar = delt),
plot(zeta, post, type = "l")) # profile of log-posterior density

information(fmodelrsm, c(0,1,2,1,0), cpar = gamm, dpar = delt) # Fisher information

with(post, mean(zeta)) # posterior mean
postmode(fmodelrsm, c(0,1,2,1,0), cpar = gamm, dpar = delt) # posterior mode

with(post, quantile(zeta, probs = c(0.025, 0.975))) # posterior credibility interval
profileci(fmodelrsm, c(0,1,2,1,0),
```

```
cpar = gamm, dpar = delt) # profile likelihood confidence interval
```

---

fmodelseq *Latent Trait Posterior for the Sequential Response Model*

Description

fmodelseq evaluates the (unnormalized) posterior density of the latent trait of a sequential item response model with a given prior distribution, and computes the probability for each item and response category given the latent trait.

Usage

```
fmodelseq(zeta, y, bpar, prior = dnorm, ...)
```

Arguments

zeta	Latent trait value.
y	Vector of length m for a single response pattern, or matrix of size s by m of a set of s item response patterns. In the latter case the posterior is computed by conditioning on the event that the response pattern is one of the s response patterns. Elements of y should be integers from 0 to r-1 where r is the number of response categories.
bpar	Matrix of size m by r-1 of step "difficulty" parameters.
prior	Function that evaluates the prior distribution of the latent trait. The default is the standard normal distribution.
...	Additional arguments to be passed to prior.

Details

The parameterization of the sequential model is such that

$$P(Y_{ij} > y|Y_{ij} \geq y, \zeta_i) = 1/(1 + \exp(-(\zeta_i - \beta_{j,y+1})))$$

for $y = 0, 1, \ldots, r-2$. This model is dicussed by Tutz (1990, 1997) and Verhelst, Glas, and de Vries (1997).

Value

post	The log of the unnormalized posterior distribution evaluated at zeta.
prob	Matrix of size m by 2 array of item response probabilities.

Author(s)

Timothy R. Johnson

References

Tutz, G. (1990). Sequential item response models with an ordered response. *British Journal of Mathematical and Statistical Psychology*, *43*, 39-55.

Tutz, G. (1997). Sequential models for ordered responses. In W. J. van der Linden & R. K. Hambleton (Eds.), *Handbook of item response theory* (pp. 139-152). New York: Springer-Verlag.

Verhelst, N. D., Glas, C. A. W., \& de Vries, H. H. (1997). A steps model to analyze partial credit. In W. J. van der Linden & R. K. Hambleton (Eds.), *Handbook of item response theory* (pp. 123-138). New York: Springer-Verlag.

Examples

```
samp <- 5000 # samples from posterior distribution
burn <- 1000 # burn-in samples to discard

beta <- matrix(0, 5, 2)

post <- postsamp(fmodelseq, c(0,1,2,1,0), bpar = beta,
control = list(nbatch = samp + burn))

post <- data.frame(sample = 1:samp,
zeta = post$batch[(burn + 1):(samp + burn)])

with(post, plot(sample, zeta), type = "l")  # trace plot of sampled realizations
with(post, plot(density(zeta, adjust = 2))) # density estimate of posterior distribution

with(posttrace(fmodelseq, c(0,1,2,1,0), bpar = beta),
plot(zeta, post, type = "l")) # profile of log-posterior density

information(fmodelseq, c(0,1,2,1,0), bpar = beta) # Fisher information

with(post, mean(zeta)) # posterior mean
postmode(fmodelseq, c(0,1,2,1,0), bpar = beta) # posterior mode

with(post, quantile(zeta, probs = c(0.025, 0.975))) # posterior credibility interval
profileci(fmodelseq, c(0,1,2,1,0), bpar = beta) # profile likelihood confidence interval
```

`information` *Fisher and Observed Information for an Item Response Model*

Description

`information` computes the Fisher (test, item, and category) and observed (test only) information for an item response model. Fisher information can be computed at any specified value of `zeta` but observed information is computed only at the MLE.

Usage

```
information(fmodel, y, zeta, observed = FALSE, ...)
```

Arguments

- `fmodel` Function with first argument `zeta` which returns a list of the (unnormalized) natural logarithm of the posterior distribution evaluted at `zeta` and a m by r matrix of item category response probabilities. These must be named `post` and `prob`, respectively. The posterior should assume an (improper) uniform prior for `zeta`, and will impose this proper if `fmodel` has a `prior` argument.
- `y` Vector of length m for a single response pattern, or matrix of size s by m of a set of s item response patterns. In the latter case the posterior is computed by conditioning on the event that the response pattern is one of the s response patterns. Elements of `y` should be integers from 0 to r-1 where r is the number of response categories.
- `zeta` The value of the latent trait at which to compute Fisher information. Observed information is always computed at the MLE regardless of `zeta`. The default is the MLE.
- `observed` Logical to determine if the observed information is computed. The default is FALSE, but only the observed information can be computed if `y` has two or more rows.
- `...` Additional arguments to be passed to `fmodel` such as item parameters, or to the numerical routines for calculating first- and second-order (parital) derivatives.

Details

The Fisher information is defined here as the negative of the expected value of the second-order derivative of the log-likelihood function for ζ_i. This is the test information function. The item and category Fisher information functions are defined by decomposing this quantity by item and category, respectively (see Baker & Kim, 2004). The observed information is the second-order derivative of the log-likelihood evaluated evaluated at the MLE of ζ_i which is computed using `postmode`. The observed information function is only computed here for the test. The Fisher information cannot be computed by `information` if `y` has more than one row (i.e., more than one response pattern).

Value

- `test` Test information at `zeta`.
- `item` Item information at `zeta` (NA if `observed = FALSE`).
- `category` Category informtation at `zeta` (NA if `observed = FALSE`).

Note

For generality `information` computes Fisher and observed information using numerical (partial) differentiation even when closed-form solutions exist. Thus even though it does not depend on `y` that argument must still be provided for computational purposes. General and some model-specific closed-form formulas for test/item/category Fisher information are given by Baker and Kim (2004).

Author(s)

Timothy R. Johnson

References

Baker, F. B. & Kim, S. H. (2004). *Item response theory: Parameter estimation techniques* (2nd ed.). New York, NY: Marcel-Dekker.

Examples

```
alph <- c(1.27,1.34,1.14,1,0.67)   # discrimination parameters
beta <- c(1.19,0.59,0.15,-0.59,-2) # difficulty parameters
gamm <- c(0.1,0.15,0.15,0.2,0.01)   # lower asymptote parameters

# Fisher information of a three-parameter logistic binary model
information(fmodel3pl, y = c(0,1,1,1,1), apar = alph, bpar = beta, cpar = gamm)

# plot of Fisher information functions for each item
zeta <- seq(-5, 5, length = 100)
info <- matrix(NA, 100, 5)
for (j in 1:100) {
info[j,] <- information(fmodel3pl, c(0,1,1,1,1), zeta = zeta[j],
apar = alph, bpar = beta, cpar = gamm)$item
}
matplot(zeta, info, type = "l", ylab = "Information", bty = "n", xlab = expression(zeta))
legend(-3, 0.3, paste("Item", 1:5), lty = 1:5, col = 1:5)

# observed information given a sum score of 4
information(fmodel3pl, patterns(5, 2, 4), apar = alph, bpar = beta, cpar = gamm,
observed = TRUE)
```

---

patterns *Response Patterns for Given Total Score(s)*

Description

`patterns` determines the response patterns that yield one or one of several total scores.

Usage

```
patterns(m, r, total)
```

Arguments

`m`	Number of items.
`r`	Number of response categories.
`total`	Vector of total score or scores. Note that item responses are assumed to be integer-valued from 0 to r-1, so total scores should be between 0 and m(r-1).

Details

This is essentially a utility function to be used to prepare input for `postsamp`, `postmode`, `posttrace`, `profileci`, or `information`.

Value

`y`	Matrix of size s by m of the s response patterns that would yield (one of) the total score(s) in `total`.

Author(s)

Timothy R. Johnson

See Also

`postsamp`, `postmode`, `posttrace`, `profileci`, `information`

Examples

```
# response patterns for five binary items that
# yield a total score of 3

y <- patterns(5, 2, 3)

# response patterns for 3 5-category items that
# yield a total score of at least 3

y <- patterns(3, 5, c(3:12))
```

`postmode` *Posterior Mode of the Posterior Distribution of the Latent Trait of an Item Response Model*

Description

`postmode` numerically computes the posterior mode for a specified posterior distribution of the latent of an item response model.

Usage

```
postmode(fmodel, y, zmin = -5, zmax = 5, ...)
```

Arguments

`fmodel`	Function with first argument `zeta` which returns a list of the (unnormalized) natural logarithm of the posterior distribution evaluted at `zeta` and a m by r matrix of item category response probabilities. These must be named `post` and `prob`, respectively. The posterior should assume an (improper) uniform prior for `zeta`, and will impose this proper if `fmodel` has a `prior` argument.
`y`	Vector of length m for a single response pattern, or matrix of size s by m of a set of s item response patterns. In the latter case the posterior is computed by conditioning on the event that the response pattern is one of the s response patterns. Elements of `y` should be integers from 0 to r-1 where r is the number of response categories.

zmin	Minimum value of ζ to use in searching for the mode.
zmax	Maximum value of ζ to use in searching for the mode.
...	Additional arguments to pass to fmodel or optimize.

Details

This is basically wrapper for optimize to find the maximum of a log-posterior or log-likelihood function of a latent trait. Note that this is the MAP estimator, and also the MLE if the prior distribution is uniform.

Value

zeta	The value of ζ that maximizes the posterior distribution.
post	The value of the log-posterior distribution at the mode.

Warning

Finding the mode is not guaranteed. Inspection of the profile of the posterior (perhaps by using posttrace) is recommended to verify that zmin and zmax are set appropriately. Problems can arise for posterior distributions that are multimodal or where no (finite) mode exists.

Author(s)

Timothy R. Johnson

See Also

See optimize for information on using this function.

Examples

```
alph <- c(1.27,1.34,1.14,1,0.67)   # discrimination parameters
beta <- c(1.19,0.59,0.15,-0.59,-2) # difficulty parameters
gamm <- c(0.1,0.15,0.15,0.2,0.01)  # lower asymptote parameters

# MAP estimate given a sum score of 3
postmode(fmodel3pl, patterns(5, 2, 3), apar = alph, bpar = beta, cpar = gamm)

# MLE given a sum score of 3
postmode(fmodel3pl, patterns(5, 2, 3), apar = alph, bpar = beta, cpar = gamm,
prior = function(z) 1)
```

postsamp *MCMC Simulation from the Posterior Distribution of the Latent Trait of an Item Response Model*

Description

postsamp implements Markov chain Monte Carlo (MCMC) algorithms to simulate realizations from the posterior distribution of the latent trait of an item response model. The distribution is conditional on a single response vector or a subset of response vectors.

Usage

```
postsamp(fmodel, y, method = "metrop", control = list(), ...)
```

Arguments

fmodel	Function with first argument zeta which returns a list of the (unnormalized) natural logarithm of the posterior distribution evaluted at zeta. This must be returned as a list with the named object post.
y	A m-dimensional vector or a s by m matrix of item responses, where in the latter case the posterior is computed by conditioning on the event that the response pattern is one of the s response patterns in y. .
method	The sampler used. Currently the options are a random-walk Metropolis sampler (method = 'metrop') or an adaptive random-walk Metropolis sampler (method = 'adapt'). Both samplers use a normal proposal distribution. The default is method = 'mcmc'.
control	List of options for the sampler. See the help file for the sampler used for details. For method = 'metrop' see metrop and for method = 'adapt' see Metro_Hastings.
...	Additional arguments to be pased to fmodel.

Details

This function uses MCMC samplers from the **mcmc** and **MHadaptive** packages, using the functions metrop and Metro_Hastings for random-walk (method = 'metrop') and adaptive random-walk (method = 'adapt') Metropolis samplers, respectively. Default options are used by postsamp except that the default burn-in samples is set to zero and the number of samples is set to 1000.

Value

out	List of the output from the sampler. See the help file for the sampler specified in method for details.

Warning

Sampling from the posterior distribution of the latent trait of an item response model typically exhibits quick convergence and efficient mixing. However as with any MCMC algorithm it is important to check the performance of the algorithm. In some cases adjustment of the scale of the proposal density may be necessary to achieve a suitable acceptance rate.

Author(s)

Timothy R. Johnson

See Also

See `metrop` and `Metro_Hastings` for details on the samplers.

Examples

```
# simulating 10000 realizations from the posterior distribution from
# a partial credit model using a non-adaptive Metropolis algorithm
zeta <- postsamp(fmodelpcm, c(0,1,2,1,0), bpar = matrix(0, 5, 2),
   control = list(nbatch = 10000))
plot(zeta$batch, type = "l") # trace plot

# simulating 10000 realizations from the posterior distribution from
# a partial credit model using an adaptive Metropolis algorithm
zeta <- postsamp(fmodelpcm, c(0,1,2,1,0), bpar = matrix(0, 5, 2),
   control = list(iterations = 10000), method = 'adapt')
plot(zeta$trace, type = "l") # trace plot
```

`posttrace` *Trace the Profile of the Log-Posterior Distribution of the Latent Trait of an Item Response Model*

Description

`posttrace` computes the (unnormalized) log-posterior distribution of the latent trait of an item response model in a given interval.

Usage

```
posttrace(fmodel, y, zeta = seq(zmin, zmax, length = length),
   zmin = -3, zmax = 3, length = 100, ...)
```

Arguments

`fmodel`	Function with first argument `zeta` which returns a list of the (unnormalized) natural logarithm of the posterior distribution evaluted at `zeta`. This must be returned as a list with the named object `post`.

`y`	Vector of length m for a single response pattern, or matrix of size s by m of a set of s item response patterns. In the latter case the posterior is computed by conditioning on the event that the response pattern is one of the s response patterns. Elements of `y` should be integers from 0 to r-1 where r is the number of response categories.
`zeta`	Vector of values of the latent trait at which to compute the log-posterior density. By default this is a sequence of `length` values from `zmin` to `zmax`.
`zmin`	Minimum value of the latent trait at which to compute the log-posterior density.
`zmax`	Maximum value of the latent trait at which to compute the log-posterior density.
`length`	Length of vector of latent trait values between `zmin` and `zmax`.
`...`	Additional arguments to be passed to `fmodel`.

Details

The primary purpose of this function is in preparing data for plotting the profile of the posterior density or likelihood (if the prior is uniform) for the latent trait. This can be useful for visual inspection of the profile posterior/liklelihood for modes and curvature.

Value

`zeta`	Returns the argument `zeta`.
`post`	The log-posterior density at each each value of `zeta`.

Note

Note that the posterior is the unnormalized posterior distribution. The normalized posterior distribution can be approximated by a density estimate based on a sample of realizations generated using `postsamp`.

Author(s)

Timothy R. Johnson

See Also

See `postsamp` for simulating realizations from the posterior distribution to use to estimate the normalized posterior density.

Examples

```
alph <- c(1.27,1.34,1.14,1,0.67)   # discrimination parameters
beta <- c(1.19,0.59,0.15,-0.59,-2) # difficulty parameters
gamm <- c(0.1,0.15,0.15,0.2,0.01)  # lower asymptote parameters

# profile of log-likelihood distribution given a sum score of 3
tmp <- posttrace(fmodel3pl, patterns(5, 2, 3), apar = alph, bpar = beta, cpar = gamm,
prior = function(z) 1)
with(tmp, plot(zeta, post, type = "l"))
```

```
# profile of log-posterior distribution (normal prior) given a sum score of 3
tmp <- posttrace(fmodel3pl, patterns(5, 2, 3), apar = alph, bpar = beta, cpar = gamm)
with(tmp, plot(zeta, post, type = "l"))
```

`profileci` *Profile Likelihood Confidence Interval of the Latent Trait of an Item Response Model*

Description

`profileci` numerically computes the profile likelihood confidence interval of the latent trait of an item response model for a given response vector or sum score(s).

Usage

```
profileci(fmodel, y, zmin = -5, zmax = 5, lower = TRUE, upper = TRUE, level = 0.95, ...)
```

Arguments

`fmodel`	Either a function with first argument `zeta` which returns the log-likelihood function or the log of the (unnormalized) posterior distribuion as a named object `post` in a list. In the latter case the prior should be specified as uniform, and if the function includes an argument `prior` for specifying the prior distribution this will be set to a uniform distribution.
`y`	A m-dimensional vector or a s by m matrix of item responses, where in the latter case the posterior is computed by conditioning on the event that the response pattern is one of the s response patterns in `y`.
`zmin`	Minimum value of the latent trait when searching for the MLE.
`zmax`	Maximum value of the latent trait when searching for the MLE.
`lower`	Logical for whether to compute the lower bound of the confidence interval (default is TRUE).
`upper`	Logical for whether to compute the upper bound of the confidence interval (default is TRUE).
`level`	Confidence level as a value in the open unit interval (default is 0.95).
`...`	Additional arguments to pass to `fmodel`.

Details

This function solves for the profile likelihood confidence interval using a root-finding approach. This can be used as an alternative to using the Fisher or observed information to compute a Wald confidence interval for the latent trait.

Value

zeta	Maximum likelihood estimate of the latent trait.
post	Value of the log-likelihood function at the maximum likelihood estimate of the latent trait.
lower	Lower bound of the confidence interval.
f.lower	Value of the log-likelihood function at lower.
upper	Upper bound of the confidence interval.
f.upper	Value of the log-likelihood function at upper.

Warning

Finding the confidence interval is not guaranteed. Inspection of the profile of the posterior (perhaps by using posttrace) is recommended to verify that zmin and zmax are set appropriately. Problems can arise for posterior distributions that are multimodal or where no (finite) mode exists.

Author(s)

Timothy R. Johnson

See Also

See uniroot for details on the root-finding function, and postsamp for the function that finds the MLE.

Examples

```
alph <- c(1.27,1.34,1.14,1,0.67)   # discrimination parameters
beta <- c(1.19,0.59,0.15,-0.59,-2) # difficulty parameters
gamm <- c(0.1,0.15,0.15,0.2,0.01)  # lower asymptote parameters

# profile confidence interval given a sum score of 3
profileci(fmodel3pl, patterns(5, 2, 3), apar = alph, bpar = beta, cpar = gamm)
```

Index

fmodel1pl, 2, *5, 6*, *10*, *14*
fmodel1pp, *3*, 4, *8*, *12*, *16*
fmodel2pl, *3*, 5, *10*, *14*
fmodel2pp, *5, 6*, 7, *8*, *12*, *16*
fmodel3pl, *3*, *6*, 9, *12*, *14*
fmodel3pp, *5*, *8*, *10*, 11, *16*
fmodel4pl, *3*, *6*, *10*, 12, *16*
fmodel4pp, *4, 5*, *8*, *11, 12*, *14*, 14
fmodelgrl, 16, *19*
fmodelgrp, *17*, 18
fmodelnrm, 20
fmodelpcm, 22, *25*
fmodelrsm, *23*, 24
fmodelseq, 26

information, 27, *29, 30*

Metro_Hastings, *32, 33*
metrop, *32, 33*

optimize, *31*

patterns, 29
postmode, *28–30*, 30
postsamp, *29, 30*, 32, *34*, *36*
posttrace, *29–31*, 33, *36*
profileci, *29, 30*, 35

uniroot, *36*