Package 'bairt'

March 27, 2018

Type Package

Title Bayesian Analysis of Item Response Theory Models

Version 0.1.2

Maintainer Javier Martinez <martinezjavier243@gmail.com>

Description Bayesian estimation of the two and three parameter models of item response theory (IRT). Also, it is possible to use a web interactive application intended for the making of an MCMC estimation and model-fit of the IRT models.

Depends R (>= 3.1.0)

Imports coda, mvtnorm, stats, graphics, shiny, shinyjs

License GPL (>= 2)

Encoding UTF-8

RoxygenNote 6.0.1

NeedsCompilation no

Author Javier Martinez [aut, cre],
Irene Garcia Mosquera [ctb]

Repository CRAN

Date/Publication 2018-03-27 12:07:03 UTC

R topics documented:

`burning.mcmc` *Burning of MCMC objects*

Description

This function gives the number of the first discarded iterations for an MCMC object of class mcmc.2pnob or mcmc.3pnob.

Usage

```
burning.mcmc(mcmclist, ...)
```

Arguments

`mcmclist`	A *mcmc.2pnob* or *mcmc.3pnob* class object.
`...`	Further arguments.

Value

The burning number for a *mcmc.2pnob* or *mcmc.3pnob* object.

Author(s)

Javier Martínez

See Also

mcmc.2pnob, mcmc.3pnob and continue.mcmc.bairt.

Examples

```
# data for model
data("MathTest")

# Only for the first 500 examinees of the data MathTest
# Two-Parameter Normal Ogive Model
model2 <- mcmc.2pnob(MathTest[1:500,], iter = 100, burning = 0)
iter.mcmc(model2)
burning.mcmc(model2)
thin(model2)
parts.mcmc(model2)
model.mcmc(model2)
data.mcmc(model2)


# For all examinees of the data MathTest
# Three-Parameter Normal Ogive Model
model3 <- mcmc.3pnob(MathTest, iter = 3500, burning = 500)
iter.mcmc(model3)
burning.mcmc(model3)
thin(model3)
parts.mcmc(model3)
model.mcmc(model3)
data.mcmc(model3)


## End(Not run)
```

---

chain.study *Convergence graphs for the simulated values*

Description

Convergence graphs for the study of the simulated values for an MCMC marginal chain.

Usage

```
chain.study(mcmclist, ...)
```

Arguments

mcmclist	A *mcmc.2pnob* or *mcmc.3pnob* class object.
...	Further arguments.

Details

The top left graph displays the sequence of simulated values and the top right graph displays the lagged correlations of the sequence as a function of the lag value. The bottom left graph is an histogram of the simulated values and the bottom right graph is the box plot of the simulated values.

Value

Convergence graphs for the study of the simulated values for an MCMC marginal chain.

Author(s)

Javier Martínez

References

Johnson, V. E. & Albert, J. H. (1999). Ordinal Data Modeling. New York: Springer.

See Also

`mcmc.2pnob`, `mcmc.3pnob` and `continue.mcmc`.

`chain.study.bairt` *Convergence graphs for the simulated values*

Description

Convergence graphs for the study of the simulated values for an MCMC marginal chain.

Usage

```
## S3 method for class 'bairt'
chain.study(mcmclist, parameter = "a", chain = 1,
  line = TRUE, ...)
```

Arguments

`mcmclist`	A *mcmc.2pnob* or *mcmc.3pnob* class object.
`parameter`	The parameter (*a*, *b*, *c* or *theta*) for graphing.
`chain`	The number of the chain that will be graphed.
`line`	A red line that represent the posterior mean of the simulated values.
`...`	Further arguments.

Details

The top left graph displays the sequence of simulated values and the top right graph displays the lagged correlations of the sequence as a function of the lag value. The bottom left graph is an histogram of the simulated values and the bottom right graph is the box plot of the simulated values.

Value

Convergence graphs for the study of the simulated values for an MCMC marginal chain.

Author(s)

Javier Martínez

References

Johnson, V. E. & Albert, J. H. (1999). Ordinal Data Modeling. New York: Springer.

See Also

mcmc.2pnob, mcmc.3pnob and continue.mcmc.

Examples

```
# data for model
data("MathTest")

# Only for the first 500 examinees of the data MathTest
# Two-Parameter Normal Ogive Model
model2 <- mcmc.2pnob(MathTest[1:500,], iter = 400, burning = 100)
check.plot(model2)
chain.study(model2, parameter = "b", chain = 12)
chain.study(model2, parameter = "theta", chain = 10)


# For all examinees of the data MathTest
# Two-Parameter Normal Ogive Model
modelAll2 <- mcmc.2pnob(MathTest, iter = 3500, burning = 500, thin = 10)
check.plot(modelAll2)
chain.study(modelAll2, parameter = "b", chain = 14)
chain.study(modelAll2, parameter = "theta", chain = 10)

# Three-Parameter Normal Ogive Model
modelAll3 <- mcmc.3pnob(MathTest, iter = 3500, burning = 500, thin = 10)
check.plot(modelAll3)
chain.study(modelAll3, parameter = "b", chain = 12)
chain.study(modelAll3, parameter = "c", chain = 10)


## End(Not run)
```

`chain.study.default` *Convergence graphs for the simulated values*

Description

Convergence graphs for the study of the simulated values for an MCMC marginal chain.

Usage

```
## Default S3 method:
chain.study(mcmclist, ...)
```

Arguments

`mcmclist`	A *mcmc.2pnob* or *mcmc.3pnob* class object.
`...`	Further arguments.

Details

The top left graph displays the sequence of simulated values and the top right graph displays the lagged correlations of the sequence as a function of the lag value. The bottom left graph is an histogram of the simulated values and the bottom right graph is the box plot of the simulated values.

Value

Convergence graphs for the study of the simulated values for an MCMC marginal chain.

Author(s)

Javier Martínez

References

Johnson, V. E. & Albert, J. H. (1999). Ordinal Data Modeling. New York: Springer.

See Also

`mcmc.2pnob`, `mcmc.3pnob` and `continue.mcmc`.

`check.plot` *Plot of the discrimination marginal posterior means against difficulty marginal posterior means*

Description

Marginal Posterior means of b_j plotted against the marginal posterior means of a_j. Each point is labeled with the number of the corresponding Item.

For the Three-Parameter Normal Ogive Item Response Model *(3pno)*, the size of the numbers refers to the marginal posterior means of c_j.

The Potential Scale Reduction Factor (*Rhat*) is calculated for each chain, *bairt* generates a single MCMC and evaluates convergence by breaking the chain in three sub chains and comparing the between- and within-subchain variance.

The *black color suggests convergence* and *red items indicate convergence problems* (*Rhat greater than 1.1*).

Usage

```
check.plot(mcmclist, ...)
```

Arguments

`mcmclist`	A *mcmc.2pnob* or *mcmc.3pnob* class object.
`...`	Further arguments.

Details

If *converg.test = TRUE* the items with Rhat menor that 1.1 are print in red color. It is useful for quick check of the convergence.

Value

A plot of the discrimination marginal posterior means against difficulty marginal posterior means. For the Three-parameter model the guessing marginal posterior means are represented by the number size of the item.

Author(s)

Javier Martínez

References

Johnson, V. E. & Albert, J. H. (1999). Ordinal Data Modeling. New York: Springer.

Gelman, A., Carlin, J. B., Stern, H. S. & Rubin, B. (2004). Bayesian Data Analysis.New York: Chapman & Hall/CRC.

See Also

mcmc.2pnob, mcmc.3pnob and continue.mcmc.bairt.

`check.plot.default` *Plot of the discrimination marginal posterior means against difficulty marginal posterior means*

Description

Marginal Posterior means of b_j plotted against the marginal posterior means of a_j. Each point is labeled with the number of the corresponding Item.

For the Three-Parameter Normal Ogive Item Response Model *(3pno)*, the size of the numbers refers to the marginal posterior means of c_j.

The Potential Scale Reduction Factor (*Rhat*) is calculated for each chain, *bairt* generates a single MCMC and evaluates convergence by breaking the chain in three sub chains and comparing the between- and within-subchain variance.

The *black color suggests convergence* and *red items indicate convergence problems* (*Rhat greater than 1.1*).

Usage

```
## Default S3 method:
check.plot(mcmclist, ...)
```

Arguments

`mcmclist`	A *mcmc.2pnob* or *mcmc.3pnob* class object.
`...`	Further arguments.

Details

If *converg.test* = *TRUE* the items with Rhat menor that 1.1 are print in red color. It is useful for quick check of the convergence.

Value

A plot of the discrimination marginal posterior means against difficulty marginal posterior means. For the Three-parameter model the guessing marginal posterior means are represented by the number size of the item.

Author(s)

Javier Martínez

References

Johnson, V. E. & Albert, J. H. (1999). Ordinal Data Modeling. New York: Springer.

Gelman, A., Carlin, J. B., Stern, H. S. & Rubin, B. (2004). Bayesian Data Analysis.New York: Chapman & Hall/CRC.

See Also

`mcmc.2pnob`, `mcmc.3pnob` and `continue.mcmc.bairt`.

`check.plot.mcmc.2pnob` *Plot of the discrimination marginal posterior means against difficulty marginal posterior means*

Description

Marginal Posterior means of b_j plotted against the marginal posterior means of a_j. Each point is labeled with the number of the corresponding Item.

For the Three-Parameter Normal Ogive Item Response Model *(3pno)*, the size of the numbers refers to the marginal posterior means of c_j.

The Potential Scale Reduction Factor (*Rhat*) is calculated for each chain, *bairt* generates a single MCMC and evaluates convergence by breaking the chain in three sub chains and comparing the between- and within-subchain variance.

The *black color suggests convergence* and *red items indicate convergence problems* (*Rhat greater than 1.1*).

Usage

```
## S3 method for class 'mcmc.2pnob'
check.plot(mcmclist, converg.test = TRUE, ...)
```

Arguments

`mcmclist`	A *mcmc.2pnob* or *mcmc.3pnob* class object.
`converg.test`	Checking if Rhat is major that 1.1.
`...`	Further arguments.

Details

If *converg.test* = *TRUE* the items with Rhat menor that 1.1 are print in red color. It is useful for quick check of the convergence.

Value

A plot of the discrimination marginal posterior means against difficulty marginal posterior means. For the Three-parameter model the guessing marginal posterior means are represented by the number size of the item.

Author(s)

Javier Martínez

References

Johnson, V. E. & Albert, J. H. (1999). Ordinal Data Modeling. New York: Springer.

Gelman, A., Carlin, J. B., Stern, H. S. & Rubin, B. (2004). Bayesian Data Analysis.New York: Chapman & Hall/CRC.

See Also

mcmc.2pnob, mcmc.3pnob and continue.mcmc.bairt.

Examples

```
# data for model
data("MathTest")

# Only for the first 500 examinees of the data MathTest
# Two-Parameter Normal Ogive Model
model2 <- mcmc.2pnob(MathTest[1:500,], iter = 400, burning = 100)
check.plot(model2)
chain.study(model2, parameter = "b", chain = 12)
chain.study(model2, parameter = "theta", chain = 10)


# For all examinees of the data MathTest
# Two-Parameter Normal Ogive Model
modelAll2 <- mcmc.2pnob(MathTest, iter = 3500, burning = 500, thin = 10)
check.plot(modelAll2)
chain.study(modelAll2, parameter = "b", chain = 14)
chain.study(modelAll2, parameter = "theta", chain = 10)

# Three-Parameter Normal Ogive Model
modelAll3 <- mcmc.3pnob(MathTest, iter = 3500, burning = 500, thin = 10)
check.plot(modelAll3)
chain.study(modelAll3, parameter = "b", chain = 12)
chain.study(modelAll3, parameter = "c", chain = 10)


## End(Not run)
```

---

check.plot.mcmc.3pnob *Plot of the discrimination marginal posterior means against difficulty marginal posterior means*

Description

Marginal Posterior means of b_j plotted against the marginal posterior means of a_j. Each point is labeled with the number of the corresponding Item.

For the Three-Parameter Normal Ogive Item Response Model *(3pno)*, the size of the numbers refers to the marginal posterior means of c_j.

The Potential Scale Reduction Factor (*Rhat*) is calculated for each chain, *bairt* generates a single MCMC and evaluates convergence by breaking the chain in three sub chains and comparing the between- and within-subchain variance.

The *black color suggests convergence* and *red items indicate convergence problems* (*Rhat greater than 1.1*).

Usage

```
## S3 method for class 'mcmc.3pnob'
check.plot(mcmclist, converg.test = T, c.probs = c(0,
  0.2, 0.5, 1), legen = "topleft", ...)
```

Arguments

`mcmclist`	A *mcmc.2pnob* or *mcmc.3pnob* class object.
`converg.test`	Checking if Rhat is major that 1.1.
`c.probs`	Vector for assignment of intervals the Guessing (c).
`legen`	Coordinates to be used to position the Guessing (c) legend.
`...`	Further arguments.

Details

If *converg.test = TRUE* the items with Rhat menor that 1.1 are print in red color. It is useful for quick check of the convergence.

Value

A plot of the discrimination marginal posterior means against difficulty marginal posterior means. For the Three-parameter model the guessing marginal posterior means are represented by the number size of the item.

Author(s)

Javier Martínez

References

Johnson, V. E. & Albert, J. H. (1999). Ordinal Data Modeling. New York: Springer.

Gelman, A., Carlin, J. B., Stern, H. S. & Rubin, B. (2004). Bayesian Data Analysis.New York: Chapman & Hall/CRC.

See Also

mcmc.2pnob, mcmc.3pnob and continue.mcmc.bairt.

Examples

```
# data for model
data("MathTest")

# Only for the first 500 examinees of the data MathTest
# Two-Parameter Normal Ogive Model
model2 <- mcmc.2pnob(MathTest[1:500,], iter = 400, burning = 100)
check.plot(model2)
chain.study(model2, parameter = "b", chain = 12)
chain.study(model2, parameter = "theta", chain = 10)


# For all examinees of the data
# Two-Parameter Normal Ogive Model
modelAll2 <- mcmc.2pnob(MathTest, iter = 3500, burning = 500, thin = 10)
check.plot(modelAll2)
chain.study(modelAll2, parameter = "b", chain = 14)
chain.study(modelAll2, parameter = "theta", chain = 10)

# Three-Parameter Normal Ogive Model
modelAll3 <- mcmc.3pnob(MathTest, iter = 3500, burning = 500, thin = 10)
check.plot(modelAll3)
chain.study(modelAll3, parameter = "b", chain = 12)
chain.study(modelAll3, parameter = "c", chain = 10)


## End(Not run)
```

`continue.mcmc` *Continue MCMC for the Estimation of the Two-Parameter or Three-Parameter Normal Ogive Model*

Description

This is a function for *bairt* objects. You can use *continue.mcmc* for continue the MCMC the Two-Parameter or Three-Parameter normal ogive item response model.

Usage

```
continue.mcmc(mcmclist, ...)
```

Arguments

`mcmclist`	A *bairt* class object (*mcmc.2pnob* or *mcmc.3pnob*).
`...`	Further arguments.

Details

If any argument (*final.values*, *c.prior*, *iter*, *burning*, *thin* or *parts*) is NULL, *continue.mcmc* take the value of the *mcmclist*.

Value

An *mcmc.2pnob or mcmc.3pnob* object.

Author(s)

Javier Martínez

References

Johnson, V. E., & Albert, J. H. (1999). Ordinal Data Modeling. New York: Springer.

A.A. Beguin, A, A & Glas, C.A.W. (2001). MCMC Estimation and Some Model-Fit Analysis of Multidimensional IRT Models. Psychometrika, 66, 541-562.

See Also

`mcmc.2pnob` and `mcmc.3pnob`.

---

continue.mcmc.bairt *Continue MCMC for the Estimation of the Two-Parameter or Three-Parameter Normal Ogive Model*

Description

This is a function for *bairt* objects. You can use *continue.mcmc* for continue the MCMC the Two-Parameter or Three-Parameter normal ogive item response model.

Usage

```
## S3 method for class 'bairt'
continue.mcmc(mcmclist, initial.value = NULL,
  c.prior = NULL, iter = NULL, burning = NULL, thin = NULL,
  parts = NULL, ...)
```

Arguments

`mcmclist`	A *bairt* class object (*mcmc.2pnob* or *mcmc.3pnob*).
`initial.value`	List with initial values.
`c.prior`	A two dimensional vector which defines the beta prior distribution of guessing parameters. The default is a non-informative prior, *Beta(1,1)*.
`iter`	Total number of iterations.
`burning`	Number of burnin iterations.

thin	The thinning interval between consecutive observations.
parts	Number of splits for MCMC chain.
...	Further arguments.

Details

If any argument (*final.values*, *c.prior*, *iter*, *burning*, *thin* or *parts*) is NULL, *continue.mcmc* take the value of the *mcmclist*.

Value

An *mcmc.2pnob or mcmc.3pnob* object.

Author(s)

Javier Martínez

References

Johnson, V. E., & Albert, J. H. (1999). Ordinal Data Modeling. New York: Springer.

A.A. Beguin, A, A & Glas, C.A.W. (2001). MCMC Estimation and Some Model-Fit Analysis of Multidimensional IRT Models. Psychometrika, 66, 541-562.

See Also

mcmc.2pnob and mcmc.3pnob.

Examples

```
# data for model
data("MathTest")

# Only for the first 500 examinees of the data MathTest
# Two-Parameter Normal Ogive Model
model2 <- mcmc.2pnob(MathTest[1:500,], iter = 100, burning = 0)

# continue the  MCMC for the Two-Parameter Normal Ogive Model
model21 <- continue.mcmc(model2, iter = 100, burning = 0)


# For all examinees of the data MathTest
# Three-Parameter Normal Ogive Model
# selection of the prior for 5 response options
cprior <- select.c.prior(5)
modelAll3 <- mcmc.3pnob(MathTest, iter = 1000, burning = 0,
                    c.prior = cprior)

#continue the  MCMC for the Three-Parameter Normal Ogive Model
# form 1
initialValues2 <- final.values.mcmc(modelAll3)
modelAll31 <- mcmc.3pnob(MathTest, initial.value = initialValues2,
```

```
                    iter = 2000, burning = 0, c.prior = cprior)
# form 2
modelAll32 <- continue.mcmc(modelAll3, iter = 2000, burning = 0)


## End(Not run)
```

`continue.mcmc.default` *Continue MCMC for the Estimation of the Two-Parameter or Three-Parameter Normal Ogive Model*

Description

This is a function for *bairt* objects. You can use *continue.mcmc* for continue the MCMC the Two-Parameter or Three-Parameter normal ogive item response model.

Usage

```
## Default S3 method:
continue.mcmc(mcmclist, ...)
```

Arguments

`mcmclist`	A *bairt* class object (*mcmc.2pnob* or *mcmc.3pnob*).
`...`	Further arguments.

Details

If any argument (*final.values*, *c.prior*, *iter*, *burning*, *thin* or *parts*) is NULL, *continue.mcmc* take the value of the *mcmclist*.

Value

An *mcmc.2pnob or mcmc.3pnob* object.

Author(s)

Javier Martínez

References

Johnson, V. E., & Albert, J. H. (1999). Ordinal Data Modeling. New York: Springer.

A.A. Beguin, A, A & Glas, C.A.W. (2001). MCMC Estimation and Some Model-Fit Analysis of Multidimensional IRT Models. Psychometrika, 66, 541-562.

See Also

`mcmc.2pnob` and `mcmc.3pnob`.

`data.mcmc` *MCMC object data*

Description

This function gives the data for an MCMC object.

Usage

```
data.mcmc(mcmclist, ...)
```

Arguments

`mcmclist`	A *mcmc.2pnob* or *mcmc.3pnob* class object.
`...`	Further arguments.

Value

The data for an MCMC object.

Author(s)

Javier Martínez

See Also

`mcmc.2pnob`, `mcmc.3pnob` and `continue.mcmc.bairt`.

Examples

```
# data for model
data("MathTest")

# Only for the first 500 examinees of the data MathTest
# Two-Parameter Normal Ogive Model
model2 <- mcmc.2pnob(MathTest[1:500,], iter = 100, burning = 0)
iter.mcmc(model2)
burning.mcmc(model2)
thin(model2)
parts.mcmc(model2)
model.mcmc(model2)
data.mcmc(model2)


# For all examinees of the data MathTest
# Three-Parameter Normal Ogive Model
model3 <- mcmc.3pnob(MathTest, iter = 3500, burning = 500)
iter.mcmc(model3)
burning.mcmc(model3)
```

```
thin(model3)
parts.mcmc(model3)
model.mcmc(model3)
data.mcmc(model3)


## End(Not run)
```

`diagnostic.mcmc` *Diagnostic of* mcmc.2pnob *or* mcmc.3pnob *object*

Description

This function gives the summary for all MCMC chains. It including calculus of Rhat, posterior mean, posterior standard deviation and posterior quartiles.

Usage

```
diagnostic.mcmc(mcmclist, ...)
```

Arguments

`mcmclist`	A *mcmc.2pnob* or *mcmc.3pnob* class object.
`...`	Further arguments.

Value

Data frame with the summary. It including calculus of Rhat, posterior mean, posterior standard deviation and posterior quartiles.

Author(s)

Javier Martínez

References

Gelman, A., Carlin, J. B., Stern, H. S. & Rubin, B. (2004). Bayesian Data Analysis.New York: Chapman & Hall/CRC.

See Also

`mcmc.2pnob`, `mcmc.3pnob` and `continue.mcmc.bairt`.

Examples

```
# data for model
data("MathTest")

# Only for the first 500 examinees of the data MathTest
# Two-Parameter Normal Ogive Model
model2 <- mcmc.2pnob(MathTest[1:500,], iter = 100, burning = 0)
diagnostic.mcmc(model2)


# For all examinees of the data MathTest
# Three-Parameter Normal Ogive Model
model3 <- mcmc.3pnob(MathTest, iter = 3500, burning = 500)
diagnostic.mcmc(model3)


## End(Not run)
```

---

final.values.mcmc *Values of the last iteration for each chain*

Description

This function gives the values of the last iteration for each chain. This is useful for assign the initial values from new MCMC models.

Usage

```
final.values.mcmc(mcmclist, ...)
```

Arguments

mcmclist	A *mcmc.2pnob* or *mcmc.3pnob* class object.
...	Further arguments.

Value

A list with the last values simulated from a *mcmc.2pnob* or *mcmc.3pnob* class object.

Author(s)

Javier Martínez

See Also

mcmc.2pnob, mcmc.3pnob and continue.mcmc.bairt.

Examples

```
# data for model
data("MathTest")

# Only for the first 500 examinees of the data MathTest
# Two-Parameter Normal Ogive Model
model2 <- mcmc.2pnob(MathTest[1:500,], iter = 100, burning = 0)

# continue the  MCMC for the Two-Parameter Normal Ogive Model
initialValues2 <- final.values.mcmc(model2)
model121 <- mcmc.2pnob(MathTest[1:500,], initial.value = initialValues2,
iter = 100, burning = 0)


# For all examinees of the data MathTest
# Three-Parameter Normal Ogive Model
model3 <- mcmc.3pnob(MathTest, iter = 3500, burning = 500)

# continue the  MCMC for the Three-Parameter Normal Ogive Model
initialValues3 <- final.values.mcmc(model3)
model131 <- mcmc.3pnob(MathTest, initial.value = initialValues3,
iter = 3000, burning = 0)


## End(Not run)
```

---

irc *Plot of posterior density of the item response curve*

Description

Plot of the *j-th* `item` response curve.

Usage

```
irc(mcmclist, ...)
```

Arguments

`mcmclist`	A *mcmc.2pnob* or *mcmc.3pnob* class object.
`...`	Further arguments.

Details

The solid line corresponds to the location of the posterior mean and the points correspond to the percentiles determined by *prob*. *prob = c(0.05, 0.95)* corresponds to the 5th and 95th percentils of the posterior density respectively.

Value

Plot of the *j-th* `item` response curve.

Author(s)

Javier Martínez

References

Johnson, V. E. & Albert, J. H. (1999). Ordinal Data Modeling. New York: Springer.

See Also

`mcmc.2pnob`, `mcmc.3pnob` and `continue.mcmc.bairt`.

`irc.bairt` *Plot of posterior density of the item response curve*

Description

Plot of the *j-th* `item` response curve.

Usage

```
## S3 method for class 'bairt'
irc(mcmclist, item = 1, color = "red", prob = c(0.05,
  0.95), ...)
```

Arguments

`mcmclist`	A *mcmc.2pnob* or *mcmc.3pnob* class object.
`item`	The number of *j-th* item.
`color`	Item response curve color.
`prob`	A vector of length two for defined the percentiles of the posterior density.
`...`	Further arguments.

Details

The solid line corresponds to the location of the posterior mean and the points correspond to the percentiles determined by *prob*. *prob = c(0.05, 0.95)* corresponds to the 5th and 95th percentils of the posterior density respectively.

Value

Plot of the *j-th* `item` response curve.

Author(s)

Javier Martínez

References

Johnson, V. E. & Albert, J. H. (1999). Ordinal Data Modeling. New York: Springer.

See Also

mcmc.2pnob, mcmc.3pnob and continue.mcmc.bairt.

Examples

```
# data for model
data("MathTest")

# Only for the first 500 examinees of the data MathTest
# Two-Parameter Normal Ogive Model
model2 <- mcmc.2pnob(MathTest[1:500,], iter = 400, burning = 100)
check.plot(model2)
irc(model2, item = 3)


# For all examinees of the data MathTest
# Three-Parameter Normal Ogive Model
# selection of the prior for 5 response options
cprior <- select.c.prior(5)
modelAll3 <- mcmc.3pnob(MathTest, iter = 1000, burning = 0,
                    c.prior = cprior)
irc(modelAll3 , item = 1, color = "blue")
irc(modelAll3 , item = 1, color = "blue", prob = c(0.1, 0.9))


## End(Not run)
```

`irc.default` *Plot of posterior density of the item response curve*

Description

Plot of the *j-th* `item` response curve.

Usage

```
## Default S3 method:
irc(mcmclist, ...)
```

Arguments

`mcmclist`	A *mcmc.2pnob* or *mcmc.3pnob* class object.
`...`	Further arguments.

Details

The solid line corresponds to the location of the posterior mean and the points correspond to the percentiles determined by *prob*. *prob = c(0.05, 0.95)* corresponds to the 5th and 95th percentils of the posterior density respectively.

Value

Plot of the *j-th* `item` response curve.

Author(s)

Javier Martínez

References

Johnson, V. E. & Albert, J. H. (1999). Ordinal Data Modeling. New York: Springer.

See Also

`mcmc.2pnob`, `mcmc.3pnob` and `continue.mcmc.bairt`.

`iter.mcmc` *Number of Iterations for an MCMC object.*

Description

This function gives the number of Iterations for a class object *mcmc.2pnob* or *mcmc.3pnob*.

Usage

```
iter.mcmc(mcmclist, ...)
```

Arguments

`mcmclist`	A *mcmc.2pnob* or *mcmc.3pnob* class object.
`...`	Further arguments.

Value

The number of iterations for an *mcmc.2pnob* or *mcmc.3pnob* object.

Author(s)

Javier Martínez

See Also

mcmc.2pnob, mcmc.3pnob and continue.mcmc.bairt.

Examples

```
# data for model
data("MathTest")

# Only for the first 500 examinees of the data MathTest
# Two-Parameter Normal Ogive Model
model2 <- mcmc.2pnob(MathTest[1:500,], iter = 100, burning = 0)
iter.mcmc(model2)
burning.mcmc(model2)
thin(model2)
parts.mcmc(model2)
model.mcmc(model2)
data.mcmc(model2)


# For all examinees of the data MathTest
# Three-Parameter Normal Ogive Model
model3 <- mcmc.3pnob(MathTest, iter = 3500, burning = 500)
iter.mcmc(model3)
burning.mcmc(model3)
thin(model3)
parts.mcmc(model3)
model.mcmc(model3)
data.mcmc(model3)


## End(Not run)
```

MathTest *Data from an math test applied at USB in 2012.*

Description

Observed data for a math test designed by Simon Bolivar University (USB) in 2012. It is represented by an 100×15 matrix of 1's and 0's.

Usage

```
data(MathTest)
```

Format

The *i-th* row of this matrix represents the answers from the *i-th* examinee, whereas the elements in *j-th* column represents the answers from the examinees to the *j-th* test item.

Examples

```
# data for model
data("MathTest")

# Only for the first 500 examinees of the data MathTest
# Two-Parameter Normal Ogive Model
model2 <- mcmc.2pnob(MathTest[1:500,], iter = 400, burning = 100)
check.plot(model2)


# For all examinees of the data MathTest
# Three-Parameter Normal Ogive Model
# selection of the prior for 5 response options
cprior <- select.c.prior(5)
model3 <- mcmc.3pnob(MathTest, iter = 3500, burning = 500,
                     c.prior = cprior)
check.plot(model3)
irc(model3, item = 11)
chain.study(model3, chain = 11, parameter = "a")
parameter.plot(model3)


## End(Not run)
```

---

mcmc.2pnob *MCMC Estimation of the Two-Parameter Normal Ogive Model*

Description

This function estimates the Two-Parameter normal ogive item response model by MCMC sampling (Johnson & Albert, 1999, p. 195). It is a modification of the function *mcmc.2pno* of *sirt* package.

Usage

```
mcmc.2pnob(data, initial.value = NULL, iter = 1000,
           burning = 500, thin = 1, parts = 3, ...)
```

Arguments

data	Data frame with dichotomous item responses.
initial.value	List with initial values
iter	Total number of iterations.
burning	Number of burnin iterations.

`thin`	The thinning interval between consecutive observations.
`parts`	Number of splits for MCMC chain.
`...`	Further arguments.

Details

For the two-parameter normal ogive item response model, we assume that the performance of the *i-th* examine depends on an unknown latent variable θ_i, and we let $\theta_1, ..., \theta_n$ respectively denotes the latent traits for all the *n* individuals taking the test.

We also assume that the probability of right answer depends only on the latent trait value and on the characteristics of the item. Specifically, for the *i-th* individual and *j-th* item, we model this probability as:

$$Pr(Y_{ij} = 1|\theta_i, a_j, b_j) = \Phi(a_j\theta_i - b_j)$$

where Φ is the standard normal cdf, and a_j and b_j are the item discrimination and item difficulty parameters associated with the *j-th* item (Johnson & Albert, 1999, p. 188).

Value

An object of class *mcmc.2pnob*. This is a list with the following elements:

`mcmcobj`	A list with the *a*, *b*, y *theta* chains.
`diagnostic`	A list with the *diag* matrix (it is a summary whit Rhat included) and the residual *deviance*.
`information`	A list with the *final.values* (values of the last iteration for each chain), and the arguments *iter*, *burning*, *data*, *thin*, *parts* and *model*, respectively.

Author(s)

Javier Martínez

The code is adapted from an R script of Alexander Robitzsch. (https://github.com/alexanderrobitzsch/sirt/blob/master/R/mcmc.2pno.R)

References

Johnson, V. E. & Albert, J. H. (1999). Ordinal Data Modeling. New York: Springer.

See Also

`mcmc.3pnob`, `continue.mcmc.bairt`, `gelman.diag` and `as.mcmc`.

Examples

```
# data for model
data("MathTest")

# estimate model only for the first 500 examinees of the data MathTest
model2 <- mcmc.2pnob(MathTest[1:500,], iter = 400, burning = 100)
# study of chains convergence
check.plot(model2)
diagnostic.mcmc(model2)
parameter.plot(model2)
chain.study(model2, parameter = "b", chain = 14)
irc(model2, item = 3)


# continue the  MCMC
# form 1
initialValues <- final.values.mcmc(model2)
model21 <- mcmc.2pnob(MathTest[1:500,], initial.value = initialValues,
iter = 3000, burning = 0)

# form 2
model22 <- continue.mcmc(model2, iter = 3000, burning = 0)


## End(Not run)
```

`mcmc.3pnob` *MCMC Estimation of the Three-Parameter Normal Ogive Model*

Description

This function estimates the Three-Parameter normal ogive item response model by MCMC sampling (Beguin & Glas, 2001, p. 542). It is a modification of the function *mcmc.3pno.testlet* of the *sirt* package.

Usage

```
mcmc.3pnob(data, initial.value = NULL, c.prior = c(1, 1), iter = 1000,
           burning = 500, thin = 1, parts = 3, ...)
```

Arguments

`data`	Data frame with dichotomous item responses.
`initial.value`	List with initial values
`c.prior`	A vector of length two which defines the beta prior distribution of guessing parameters. The default is a non-informative prior, *Beta(1,1)*.
`iter`	Total number of iterations.

burning	Number of burnin iterations.
thin	The thinning interval between consecutive observations.
parts	Number of splits for MCMC chain.
...	Further arguments.

Details

For the Three-parameter normal ogive item response model, we assume that the performance of the *i-th* examine depends on an unknown latent variable θ_i, and we let $\theta_1, ..., \theta_n$ respectively denotes the latent traits for all the *n* individuals taking the test.

We also assume that the probability of right answer depends only on the latent trait value and on the characteristics of the item. Specifically, for the *i-th* individual and *j-th* item, we model this probability as:

$$Pr(Y_{ij} = 1|\theta_i, a_j, b_j, c_j) = c_j + (1 - c_j)\Phi(a_j\theta_i - b_j)$$

where Φ is the standard normal cdf, and a_j, b_j and c_j are the item discrimination, item difficulty and item guessing parameters associated with the *j-th* item (Beguin & Glas, 2001, p. 542).

Value

An object of class *mcmc.3pnob*. This is a list with the following elements:

mcmcobj	A list with the *a*, *b*, y *theta* chains.
diagnostic	A list with the *diag* matrix (it is a summary whit Rhat included) and the residual *deviance*.
information	A list with the *final.values* (values of the last iteration for each chain), and the arguments *c.prior*, *iter*, *burning*, *data*, *thin*, *parts* and *model*, respectively.

Author(s)

Javier Martínez

The code is adapted from an R script of Alexander Robitzsch. (https://github.com/alexanderrobitzsch/sirt/blob/master/R/mcmc.3pno.testlet.R)

References

Beguin, A, A. & Glas, C.A.W. (2001). MCMC Estimation and Some Model-Fit Analysis of Multidimensional IRT Models. Psychometrika, 66, 541-562.

See Also

mcmc.2pnob, continue.mcmc.bairt, gelman.diag and as.mcmc.

Examples

```
# data for model
data("MathTest")

# estimate model only for the first 500 examinees of the data MathTest
# selection of the prior for 5 response options
cprior <- select.c.prior(5)
# estimate model only for the first 500 examinees of the data MathTest
model3 <- mcmc.3pnob(MathTest[1:500,], iter = 300, burning = 0,
                     c.prior = cprior)

# study of chains convergence model3
check.plot(model3)
diagnostic.mcmc(model3)
parameter.plot(model3)
chain.study(model3, parameter = "a", chain = 15)
irc(model3, item = 1)


# continue the  MCMC
# form 1
initialValues2 <- final.values.mcmc(model3)
model31 <- mcmc.3pnob(MathTest[1:500,], initial.value = initialValues2,
                      iter = 3000, burning = 0, c.prior = cprior)
# form 2
model32 <- continue.mcmc(model3, iter = 3000, burning = 0)


## End(Not run)
```

`model.mcmc` *MCMC object model*

Description

This function gives the model from MCMC object.

Usage

```
model.mcmc(mcmclist, ...)
```

Arguments

`mcmclist`	A *mcmc.2pnob* or *mcmc.3pnob* class object.
`...`	Further arguments.

Value

The model from MCMC object.

Author(s)

Javier Martínez

See Also

mcmc.2pnob, mcmc.3pnob and continue.mcmc.bairt.

Examples

```
# data for model
data("MathTest")

# Only for the first 500 examinees of the data MathTest
# Two-Parameter Normal Ogive Model
model2 <- mcmc.2pnob(MathTest[1:500,], iter = 100, burning = 0)
iter.mcmc(model2)
burning.mcmc(model2)
thin(model2)
parts.mcmc(model2)
model.mcmc(model2)
data.mcmc(model2)


# For all examinees of the data MathTest
# Three-Parameter Normal Ogive Model
model3 <- mcmc.3pnob(MathTest, iter = 3500, burning = 500)
iter.mcmc(model3)
burning.mcmc(model3)
thin(model3)
parts.mcmc(model3)
model.mcmc(model3)
data.mcmc(model3)


## End(Not run)
```

---

object.coda *Creating an mcmc.list for coda package*

Description

The function *object.coda* create a *mcmc.list* object. With this is possible to study the chain using the coda packet.

Usage

```
object.coda(mcmclist, ...)
```

Arguments

mcmclist	A *mcmc.2pnob* or *mcmc.3pnob* class object.
...	Further arguments.

Details

The function *object.coda* create a *mcmc.list* object of the marginal chain selectionated. The marginal chain is splited in subchains determined by *parts*. The aim is represent parallel chains with different starting values (Beguin & Glas, 2001, p. 547).

Value

A *mcmc.list* coda packet object.

Author(s)

Javier Martínez

References

A.A. Beguin, A, A. & Glas, C.A.W. (2001). MCMC Estimation and Some Model-Fit Analysis of Multidimensional IRT Models. Psychometrika, 66, 541-562.

See Also

as.mcmc.list and as.mcmc.

---

object.coda.bairt *Creating an mcmc.list for coda package*

Description

The function *object.coda* create a *mcmc.list* object. With this is possible to study the chain using the coda packet.

Usage

```
## S3 method for class 'bairt'
object.coda(mcmclist, parameter = "a", chain = 1,
  parts = NULL, ...)
```

Arguments

mcmclist	A *mcmc.2pnob* or *mcmc.3pnob* class object.
parameter	The parameter (a, b, c or theta) for graphing.
chain	The parameter's chain that will be graphed.
parts	Number of splits for MCMC chain.
...	Further arguments.

Details

The function *object.coda* create a *mcmc.list* object of the marginal chain selectionated. The marginal chain is splited in subchains determined by *parts*. The aim is represent parallel chains with different starting values (Beguin & Glas, 2001, p. 547).

Value

A *mcmc.list* coda packet object.

Author(s)

Javier Martínez

References

A.A. Beguin, A, A. & Glas, C.A.W. (2001). MCMC Estimation and Some Model-Fit Analysis of Multidimensional IRT Models. Psychometrika, 66, 541-562.

See Also

`as.mcmc.list` and `as.mcmc`.

Examples

```
# data for model
data("MathTest")

# Only for the first 500 examinees of the data MathTest
# Two-Parameter Normal Ogive Model
model2 <- mcmc.2pnob(MathTest[1:500,], iter = 400, burning = 100)

chain_a1 <- object.coda(model2, parameter = "a", chain = 1)
coda::gelman.plot(chain_a1)
coda::gelman.diag(chain_a1)
plot(chain_a1)


# For all examinees of the data MathTest
# Three-Parameter Normal Ogive Model
# selection of the prior for 5 response options
cprior <- select.c.prior(5)
model3 <- mcmc.3pnob(MathTest, iter = 3500, burning = 500,
                     c.prior = cprior, parts = 3)

chain_c1 <- object.coda(model3, parameter = "c", chain = 1)
coda::gelman.plot(chain_c1)
coda::gelman.diag(chain_c1)
plot(chain_c1)


## End(Not run)
```

`object.coda.default` *Creating an mcmc.list for coda package*

Description

The function *object.coda* create a *mcmc.list* object. With this is possible to study the chain using the coda packet.

Usage

```
## Default S3 method:
object.coda(mcmclist, ...)
```

Arguments

`mcmclist`	A *mcmc.2pnob* or *mcmc.3pnob* class object.
`...`	Further arguments.

Details

The function *object.coda* create a *mcmc.list* object of the marginal chain selectionated. The marginal chain is splited in subchains determined by *parts*. The aim is represent parallel chains with different starting values (Beguin & Glas, 2001, p. 547).

Value

A *mcmc.list* coda packet object.

Author(s)

Javier Martínez

References

A.A. Beguin, A, A. & Glas, C.A.W. (2001). MCMC Estimation and Some Model-Fit Analysis of Multidimensional IRT Models. Psychometrika, 66, 541-562.

See Also

`as.mcmc.list` and `as.mcmc`.

`parameter.plot` *Graph of marginal posterior densities*

Description

Graph of marginal posterior densities for the item parameters (*a*, *b* or *c*).

Usage

```
parameter.plot(mcmclist, ...)
```

Arguments

`mcmclist`	A *mcmc.2pnob* or *mcmc.3pnob* class object.
`...`	Further arguments.

Details

Graph of marginal posterior densities of the item parameter *a, b* for *mcmc.2pnob* object or *a, b, c* for *mcmc.3pnob object*. The center of error bar corresponds to the marginal posterior mean and the extremes correspond to percentiles of the marginal posterior density (These are delimited by *prob*). For example, *prob = c(0.05, 0.95)* is equivalent to the 5th and 95th percentiles of the marginal posterior density.

Value

Graph of marginal posterior densities for the item parameters (*a*, *b* or *c*).

Author(s)

Javier Martínez

References

Johnson, V. E. & Albert, J. H. (1999). Ordinal Data Modeling. New York: Springer.

See Also

`mcmc.2pnob`, `mcmc.3pnob` and `continue.mcmc.bairt`.

`parameter.plot.bairt` *Graph of marginal posterior densities*

Description

Graph of marginal posterior densities for the item parameters (*a*, *b* or *c*).

Usage

```
## S3 method for class 'bairt'
parameter.plot(mcmclist, items = NULL, parameter = NULL,
  prob = c(0.05, 0.95), ...)
```

Arguments

`mcmclist`	A *mcmc.2pnob* or *mcmc.3pnob* class object.
`items`	A vector to indicate the item to be plotted.
`parameter`	The parameter (*a*, *b*, *c* or *theta*) for graphing.
`prob`	A vector of length two for defined the percentiles of the posterior density.
`...`	Further arguments.

Details

Graph of marginal posterior densities of the item parameter *a, b* for *mcmc.2pnob* object or *a, b, c* for *mcmc.3pnob object*. The center of error bar corresponds to the marginal posterior mean and the extremes correspond to percentiles of the marginal posterior density (These are delimited by *prob*). For example, *prob = c(0.05, 0.95)* is equivalent to the 5th and 95th percentiles of the marginal posterior density.

Value

Graph of posterior densities of the item parameter (*a*, *b* or *c*).

Author(s)

Javier Martínez

References

Johnson, V. E. & Albert, J. H. (1999). Ordinal Data Modeling. New York: Springer.

See Also

`mcmc.2pnob`, `mcmc.3pnob` and `continue.mcmc.bairt`.

Examples

```
# data for model
data("MathTest")

# Only for the first 500 examinees of the data MathTest
# Two-Parameter Normal Ogive Model
model2 <- mcmc.2pnob(MathTest[1:500,], iter = 400, burning = 100)
parameter.plot(model2)
parameter.plot(model2, items = c(2, 10:15))
parameter.plot(model2, items = 1:100, parameter = "theta" )


# For all examinees of the data MathTest
# Three-Parameter Normal Ogive Model
model3 <- mcmc.3pnob(MathTest, iter = 3500, burning = 500)
parameter.plot(model3)
parameter.plot(model3, items = c(2, 10:15))
parameter.plot(model3, items = 1:100, parameter = c("c", "theta"))


## End(Not run)
```

`parameter.plot.default` *Graph of marginal posterior densities*

Description

Graph of marginal posterior densities for the item parameters (*a*, *b* or *c*).

Usage

```
## Default S3 method:
parameter.plot(mcmclist, ...)
```

Arguments

`mcmclist`	A *mcmc.2pnob* or *mcmc.3pnob* class object.
`...`	Further arguments.

Details

Graph of marginal posterior densities of the item parameter *a, b* for *mcmc.2pnob* object or *a, b, c* for *mcmc.3pnob object*. The center of error bar corresponds to the marginal posterior mean and the extremes correspond to percentiles of the marginal posterior density (These are delimited by *prob*). For example, *prob = c(0.05, 0.95)* is equivalent to the 5th and 95th percentiles of the marginal posterior density.

Value

Graph of posterior densities of the item parameter (*a*, *b* or *c*).

Author(s)

Javier Martínez

References

Johnson, V. E. & Albert, J. H. (1999). Ordinal Data Modeling. New York: Springer.

See Also

`mcmc.2pnob`, `mcmc.3pnob` and `continue.mcmc.bairt`.

`parts.mcmc` *Number of splits for MCMC chain*

Description

This function gives the splits number for a *mcmc.2pnob* or *mcmc.3pnob* object.

Usage

```
parts.mcmc(mcmclist, ...)
```

Arguments

`mcmclist`	A *mcmc.2pnob* or *mcmc.3pnob* class object.
`...`	Further arguments.

Value

The splits number for a *mcmc.2pnob* or *mcmc.3pnob* object.

Author(s)

Javier Martínez

See Also

`mcmc.2pnob`, `mcmc.3pnob` and `continue.mcmc.bairt`.

Examples

```
# data for model
data("MathTest")

# Only for the first 500 examinees of the data MathTest
# Two-Parameter Normal Ogive Model
model2 <- mcmc.2pnob(MathTest[1:500,], iter = 100, burning = 0)
iter.mcmc(model2)
burning.mcmc(model2)
thin(model2)
parts.mcmc(model2)
model.mcmc(model2)
data.mcmc(model2)


# For all examinees of the data MathTest
# Three-Parameter Normal Ogive Model
model3 <- mcmc.3pnob(MathTest, iter = 3500, burning = 500)
iter.mcmc(model3)
burning.mcmc(model3)
thin(model3)
parts.mcmc(model3)
model.mcmc(model3)
data.mcmc(model3)


## End(Not run)
```

---

sabirt *Shiny App for Bayesian Item Response Theory (SABIRT)*

Description

A web interactive application intended for the making of an MCMC (Markov Chain Monte Carlo Methods) estimation and model-fit of the item response models designed by Johnson and Albert (2pno, 1999) and Glas and Beguim (3pno, 2001). The outcome are the items parameters (difficulties and discrimination for 2pno, and additionally the chance to guess the right answers for 3pno) and also the latent abilities of each examinee.

Usage

```
sabirt()
```

Author(s)

Javier Martínez and Irene Garcia Mosquera

References

Beguin, A, A. & Glas, C.A.W. (2001). MCMC Estimation and Some Model-Fit Analysis of Multidimensional IRT Models. Psychometrika, 66, 541-562.

Harwell, M. R, & Baker, F. B. (1991). The use of Prior Distributions in Marginalized Bayesian Item Parameter Estimation: A Didactic. Psychometrika, 15, 375-389.

Johnson, V. E. & Albert, J. H. (1999). Ordinal Data Modeling. New York: Springer.

`select.c.prior` *Select the c prior for the Three-Parameter Normal Ogive Model*

Description

Select the *c* (guessing parameter) prior for *mcmc.3pnob*, through the application of Bayes Modal Estimation Equations.

Usage

```
select.c.prior(nitem, m = 20, ...)
```

Arguments

`nitem`	Number of alternatives for each item.
`m`	It is a priori weight assigned to the prior information. $m = 20$ by default.
`...`	Further arguments.

Details

Because *c* (guessing parameter) is bounded by 0 and 1, a $Beta(\alpha, \beta)$ prior distribution was proposed by Swaminathan and Gifford (1986). These parameters are defined as $\alpha = mp + 1$ and $\beta = m(p-1) + 1$, where $p = 1/n$ with *n* = number of alternatives for each item (Harwell & Baker, 1991, p.386)

Value

A vector length 2, this indicate the *c* (guessing parameter) prior for *mcmc.3pnob*.

Author(s)

Javier Martínez

References

Harwell, M. R, & Baker, F. B. (1991). The use of Prior Distributions in Marginalized Bayesian Item Parameter Estimation: A Didactic. Psychometrika, 15, 375-389.

See Also

mcmc.3pnob and continue.mcmc.bairt.

Examples

```
# data for model
data("MathTest")

# selection of the prior for 5 response options
cprior <- select.c.prior(5)

# estimate model only for the first 500 examinees of the data MathTest
model3 <- mcmc.3pnob(MathTest[1:500,], iter = 300, burning = 0,
                     c.prior = cprior)

## End(Not run)
```

---

thin *Thinning interval*

Description

This function gives the thinning interval for a *mcmc.2pnob* or *mcmc.3pnob* object.

Usage

```
thin(mcmclist, ...)
```

Arguments

mcmclist	A *mcmc.2pnob* or *mcmc.3pnob* object class object.
...	Further arguments.

Value

The thinning interval for a *mcmc.2pnob* or *mcmc.3pnob* object.

Author(s)

Javier Martínez

See Also

mcmc.2pnob, mcmc.3pnob and continue.mcmc.bairt.

Examples

```
# data for model
data("MathTest")

# Only for the first 500 examinees of the data MathTest
# Two-Parameter Normal Ogive Model
model2 <- mcmc.2pnob(MathTest[1:500,], iter = 100, burning = 0)
iter.mcmc(model2)
burning.mcmc(model2)
thin(model2)
parts.mcmc(model2)
model.mcmc(model2)
data.mcmc(model2)


# For all examinees of the data MathTest
# Three-Parameter Normal Ogive Model
model3 <- mcmc.3pnob(MathTest, iter = 3500, burning = 500)
iter.mcmc(model3)
burning.mcmc(model3)
thin(model3)
parts.mcmc(model3)
model.mcmc(model3)
data.mcmc(model3)


## End(Not run)
```

Index