Package 'dcminfo'

November 6, 2017

Type Package

Title Information Matrix for Diagnostic Classification Models

Version 0.1.7

Maintainer Yanlou Liu <liuyanlou@163.com>

Description A set of asymptotic methods that can be used to directly estimate the expected (Fisher) information matrix by Liu, Tian, and Xin (2016) <doi:10.3102/1076998615621293> in diagnostic classification models or cognitive diagnostic models are provided when marginal maximum likelihood estimation is used. For these methods, both the item and structural model parameters are considered simultaneously. Specifically, the observed information matrix, the empirical cross-product information matrix and the sandwich-type co-variance matrix that can be used to estimate the asymptotic co-variance matrix (or the model parameter standard errors) within the context of diagnostic classification models are provided.

License GPL-3

Encoding UTF-8

LazyData true

Depends R (>= 3.1.0)

Imports CDM, utils, stats, methods

Collate 'dcminfo.R' 'amps.R' 'lcdmobsp2.R' 'lcdmxpd.R' 'sim_CRUM_N1000.R' 'sim_DINA_N1000.R' 'utils.R'

RoxygenNote 6.0.1

Suggests rmarkdown

NeedsCompilation no

Author Yanlou Liu [aut, cre], Tao Xin [aut]

Repository CRAN

Date/Publication 2017-11-06 12:05:45 UTC

R topics documented:

`amps` *Generate the possible attribute mastery profile*

Description

This function is used to generate the possible attribute mastery profile.

Usage

```
amps(q_matrix, ind_zero_probs = NULL)
```

Arguments

`q_matrix`	A $J \times K$ `matrix` (Q-matrix) defines which attributes are measured by which items.
`ind_zero_probs`	a `vector` of integers indicating which attribute mastery profiles have zero probability. The default is NULL (none of the attribute mastery profiles has zero probability).

Value

A `matrix` giving the generated attribute mastery profiles.

Author(s)

Yanlou Liu, Qufu Normal University, <liuyanlou@163.com>
Tao Xin, Beijing Normal University

References

George, A. C., Robitzsch, A., Kiefer, T., Gross, J., & Uenlue, A. (2016). The R Package CDM for cognitive diagnosis models. *Journal of Statistical Software*, 74(2), 1-24. doi:10.18637/jss.v074.i02

Ma, W. & de la Torre, J. (2017). GDINA: The generalized DINA model framework. *R package version 1.4.2*. Retrived from https://CRAN.R-project.org/package=GDINA

Robitzsch, A., Kiefer, T., George, A. C., & Uenlue, A. (2017). *CDM: Cognitive diagnosis modeling. R package version 5.9-27*. Retrived from https://CRAN.R-project.org/package=CDM

Examples

```
# Example 1.
simqmatrix <- sim_DINA_N1000$simqmatrix
simqmatrix
attr_mast_patt <- amps(q_matrix=simqmatrix)
attr_mast_patt

# Example 2.
ind_zero_probs <- c(3,7)
attr_mast_patt <- amps(q_matrix=simqmatrix, ind_zero_probs=ind_zero_probs)
attr_mast_patt
```

`dcminfo` *The information matrix for diagnostic classification models (or cognitive diagnostic models)*

Description

This function is used to estimate the information matrix for diagnostic classification models (DCMs; Rupp, Templin, & Henson, 2010) or cognitive diagnostic models, such as the the observed information matrix, the empirical cross-product information matrix and the sandwich-type covariance matrix that can be used to estimate the asymptotic covariance matrix or the model parameter standard errors.

Usage

```
dcminfo(dat, delta, attr_probs, q_matrix, Mj, Aj, attr_mast_patt = NULL,
  linkfct = "logit", info_type = "Sw")
```

Arguments

`dat`	A $N \times J$ binary data `matrix` consisting of the responses of N examinees to J items.
`delta`	A `list` of item parameter estimates.
`attr_probs`	A `vector` of the estimated attribute mastery profile probability.
`q_matrix`	A $J \times K$ `matrix` (Q-matrix) defines which attributes are measured by which items.
`Mj`	A `list` of the design matrices and labels for each item (see de la Torre, 2011).
`Aj`	A `list` of the possible combinations of the required attributes for each item (see de la Torre, 2011).
`attr_mast_patt`	A $L \times K$ binary `matrix` defines the attribute mastery profiles. see `amps`.
`linkfct`	Type of the link function for the DCMs. It can be "logit", "identity", or "log". In the current version, only the "logit" link function is now available.

`info_type` The returned information (or covariance) matrix type. The It can be "XPD" (the empirical cross-product information matrix), "Obs"(the observed information matrix), or "Sw" (the sandwich-type covariance matrix). The default is "Sw".

Value

A `matrix` giving information, or covariance matrix.

Author(s)

Yanlou Liu, Qufu Normal University, <liuyanlou@163.com>
Tao Xin, Beijing Normal University

References

Liu, Y., Tian, W., & Xin, T. (2016). An Application of M2 Statistic to Evaluate the Fit of Cognitive Diagnostic Models. *Journal of Educational and Behavioral Statistics, 41*, 3-26.

Liu, Y., Xin, T., Andersson, B. & Tian, W. (2017). Information Matrix Estimation Procedures for Cognitive Diagnostic Models. *under review*.

de la Torre, J. (2011). The generalized DINA model framework. *Psychometrika, 76*, 179-199.

Rupp, A. A., Templin, J., & Henson, R. A. (2010). *Diagnostic measurement: theory, methods, and applications*. New York, NY: Guilford.

Examples

```
#Example 1.
#The sandwich-type covariance matrix, the empirical cross-product information matrix,
#and the observed information matrix for the DINA model

simresp <- sim_DINA_N1000$simresp
head(simresp)

simdelta <- sim_DINA_N1000$simdelta
simdelta

simqmatrix <- sim_DINA_N1000$simqmatrix
simqmatrix

simAj <- sim_DINA_N1000$simAj
simAj

simMj <- sim_DINA_N1000$simMj
simMj

simAttrProbs <- sim_DINA_N1000$simAttrProbs
simAttrProbs

# The number of the item parameters
N_delta <- length(unlist(simdelta))
N_delta
```

```
#Example 1.1 The sandwich-type covariance matrix

Sw_res <- dcminfo(dat=simresp, delta=simdelta, attr_probs=simAttrProbs,
                  q_matrix=simqmatrix, Mj=simMj, Aj=simAj)


Sw_se_delta <- sqrt(diag(Sw_res))[1:N_delta]

Sw_est_delta_se <- data.frame(delta_est= unlist(simdelta), se_delta=Sw_se_delta)
Sw_est_delta_se


#Example 1.2 The empirical cross-product information matrix

XPD_res <- dcminfo(dat=simresp, delta=simdelta, attr_probs=simAttrProbs,
                   q_matrix=simqmatrix, Mj=simMj, Aj=simAj, info_type = "XPD")

# Calculate the covariance matrix of the model parameters based on the XPD matrix
inv_XPD_res <- solve(XPD_res)

XPD_se_delta <- sqrt(diag(inv_XPD_res))[1:N_delta]
XPD_est_delta_se <- data.frame(delta_est= unlist(simdelta), se_delta=XPD_se_delta)
XPD_est_delta_se


#Example 1.3 The observed information matrix

Obs_res <- dcminfo(dat=simresp, delta=simdelta, attr_probs=simAttrProbs,
                   q_matrix=simqmatrix, Mj=simMj, Aj=simAj, info_type = "Obs")

# Calculate the covariance matrix of the model parameters based on the Obs matrix
inv_Obs_res <- solve(Obs_res)

Obs_se_delta <- sqrt(diag(inv_Obs_res))[1:N_delta]
Obs_est_delta_se <- data.frame(delta_est= unlist(simdelta), se_delta=Obs_se_delta)
Obs_est_delta_se


# Example 2.
#The sandwich-type covariance matrix, the empirical cross-product information matrix,
#and the observed information matrix for the C-RUM

simresp <- sim_CRUM_N1000$simresp
head(simresp)

simdelta <- sim_CRUM_N1000$simdelta
simdelta

simqmatrix <- sim_CRUM_N1000$simqmatrix
simqmatrix
```

```
simAj <- sim_CRUM_N1000$simAj
simAj

simMj <- sim_CRUM_N1000$simMj
simMj

simAttrProbs <- sim_CRUM_N1000$simAttrProbs
simAttrProbs

# The number of the item parameters
N_delta <- length(unlist(simdelta))
N_delta


#Example 2.1 The sandwich-type covariance matrix

Sw_res <- dcminfo(dat=simresp, delta=simdelta, attr_probs=simAttrProbs,
                  q_matrix=simqmatrix, Mj=simMj, Aj=simAj)


Sw_se_delta <- sqrt(diag(Sw_res))[1:N_delta]

Sw_est_delta_se <- data.frame(delta_est= unlist(simdelta), se_delta=Sw_se_delta)
Sw_est_delta_se


#Example 2.2 The empirical cross-product information matrix

XPD_res <- dcminfo(dat=simresp, delta=simdelta, attr_probs=simAttrProbs,
                   q_matrix=simqmatrix, Mj=simMj, Aj=simAj, info_type = "XPD")

# Calculate the covariance matrix of the model parameters based on the XPD matrix
inv_XPD_res <- solve(XPD_res)

XPD_se_delta <- sqrt(diag(inv_XPD_res))[1:N_delta]
XPD_est_delta_se <- data.frame(delta_est= unlist(simdelta), se_delta=XPD_se_delta)
XPD_est_delta_se


#Example 2.3 The observed information matrix

Obs_res <- dcminfo(dat=simresp, delta=simdelta, attr_probs=simAttrProbs,
                   q_matrix=simqmatrix, Mj=simMj, Aj=simAj, info_type = "Obs")

# Calculate the covariance matrix of the model parameters based on the Obs matrix
inv_Obs_res <- solve(Obs_res)

Obs_se_delta <- sqrt(diag(inv_Obs_res))[1:N_delta]
Obs_est_delta_se <- data.frame(delta_est= unlist(simdelta), se_delta=Obs_se_delta)
Obs_est_delta_se


#Example 3. User-specified attribute mastery patterns
```

```
attr_mast_patt <- amps(q_matrix=simqmatrix)

Sw_res <- dcminfo(dat=simresp, delta=simdelta, attr_probs=simAttrProbs,
attr_mast_patt = attr_mast_patt, q_matrix=simqmatrix, Mj=simMj, Aj=simAj)

Sw_se_delta <- sqrt(diag(Sw_res))[1:N_delta]

Sw_est_delta_se <- data.frame(delta_est= unlist(simdelta), se_delta=Sw_se_delta)
Sw_est_delta_se


#Example 4. Using the gdina function from the CDM package
library("CDM")
d1 <- CDM::gdina(data = sim_DINA_N1000$simresp, q.matrix = sim_DINA_N1000$simqmatrix,
maxit= 1000, rule="DINA", linkfct = "logit", calc.se=FALSE)
delta <- d1$delta
N_delta <- length(unlist(delta))
attr_probs <- d1$control$attr.prob[,1]
attr_mast_patt <- amps(q_matrix=simqmatrix)
Mj <- d1$Mj
Aj <- d1$Aj


#Example 4.1 The sandwich-type covariance matrix

Sw_res <- dcminfo(dat=sim_DINA_N1000$simresp, delta=delta, attr_probs=attr_probs,
                  q_matrix=sim_DINA_N1000$simqmatrix, Mj=Mj, Aj=Aj)

Sw_se_delta <- sqrt(diag(Sw_res))[1:N_delta]

Sw_est_delta_se <- data.frame(delta_est= unlist(delta), se_delta=Sw_se_delta)
Sw_est_delta_se


#Example 4.2 The empirical cross-product information matrix

XPD_res <- dcminfo(dat=sim_DINA_N1000$simresp, delta=delta, attr_probs=attr_probs,
                   q_matrix=simqmatrix, Mj=Mj, Aj=Aj, info_type = "XPD")

# Calculate the covariance matrix of the model parameters based on the XPD matrix
inv_XPD_res <- solve(XPD_res)

XPD_se_delta <- sqrt(diag(inv_XPD_res))[1:N_delta]
XPD_est_delta_se <- data.frame(delta_est= unlist(delta), se_delta=XPD_se_delta)
XPD_est_delta_se


#Example 4.3 The observed information matrix

Obs_res <- dcminfo(dat=sim_DINA_N1000$simresp, delta=delta, attr_probs=attr_probs,
                   q_matrix=simqmatrix, Mj=Mj, Aj=Aj, info_type = "Obs")
```

```
# Calculate the covariance matrix of the model parameters based on the Obs matrix
inv_Obs_res <- solve(Obs_res)

Obs_se_delta <- sqrt(diag(inv_Obs_res))[1:N_delta]
Obs_est_delta_se <- data.frame(delta_est= unlist(delta), se_delta=Obs_se_delta)
Obs_est_delta_se
```

`sim_CRUM_N1000` *Simulated observed response data, Q-matrix and model parameters (compensatory reparameterized unified model)*

Description

Artificial Q-matrix for 14 items with 3 attributes. A list of observed response data for 14 items, that was generated using the compensatory reparameterized unified model (C-RUM; Hartz, 2002).

Usage

```
sim_CRUM_N1000
```

Format

A list of observed response data Q-matrix, model parameters for 14 items with components:

`simresp` simulated responses data matrix of 1000 examinees response to 14 items.

`simdelta` A list of simulated item parameters for 14 items.

`simqmatrix` Artificial Q-matrix specifies the relationship between 14 items and 3 attributes.

`simAj` A `list` of the possible combinations of the required attributes for 14 items (see de la Torre, 2011).

`simMj` A `list` of the design matrices and labels for 14 items (see de la Torre, 2011).

`simAttrProbs` A Simulated `vector` of the probabilities of attribute mastery profile of 2^3.

References

de la Torre, J. (2011). The generalized DINA model framework. *Psychometrika, 76*, 179-199.

Hartz, S. M. (2002). A bayesian framework for the unified model for assessing cognitive abilities: Blending theory with practicality (Unpublished doctoral dissertation). University of Illinois at Urbana-Champaign.

`sim_DINA_N1000` *Simulated observed response data, Q-matrix and model parameters (the deterministic inputs, noisy "and" gate model)*

Description

Artificial Q-matrix for 14 items with 3 attributes. A list of observed response data for 14 items, that was generated using the the deterministic inputs, noisy "and" gate model (DINA; Haertel, 1989; Junker & Sijtsma, 2001) .

Usage

```
sim_DINA_N1000
```

Format

A list of observed response data Q-matrix, model parameters for 14 items with components:

`simresp` simulated responses data matrix of 1000 examinees response to 14 items.

`simdelta` A list of simulated item parameters for 14 items.

`simqmatrix` Artificial Q-matrix specifies the relationship between 14 items and 3 attributes.

`simAj` A `list` of the possible combinations of the required attributes for 14 items (see de la Torre, 2011).

`simMj` A `list` of the design matrices and labels for 14 items (see de la Torre, 2011).

`simAttrProbs` A Simulated `vector` of the probabilities of attribute mastery profile of 2^3.

References

de la Torre, J. (2011). The generalized DINA model framework. *Psychometrika, 76*, 179-199.

Haertel, E. H. (1989). Using restricted latent class models to map the skill structure of achievement items. *Journal of Educational Measurement, 26*, 301-321.

Junker, B. W., & Sijtsma, K. (2001). Cognitive assessment models with few assumptions, and connections with nonparametric item response theory. *Applied Psychological Measurement, 25*, 258-272.

Index