Package 'HETOP'

June 28, 2019

Type Package

Title MLE and Bayesian Estimation of Heteroskedastic Ordered Probit (HETOP) Model

Version 0.2-6

Date 2019-06-26

Depends R (>= 3.4.0), R2jags, splines, stats

Author J.R. Lockwood

Maintainer J.R. Lockwood <jrlockwood@ets.org>

Description Provides functions for maximum likelihood and Bayesian estimation of the Heteroskedastic Ordered Probit (HETOP) model, using methods described in Lockwood, Castellano and Shear (2018) <doi:10.3102/1076998618795124> and Reardon, Shear, Castellano and Ho (2017) <doi:10.3102/1076998616666279>. It also provides a general function to compute the triple-goal estimators of Shen and Louis (1998) <doi:10.1111/1467-9868.00135>.

License GPL (>= 2)

NeedsCompilation no

Repository CRAN

Date/Publication 2019-06-28 16:20:27 UTC

R topics documented:

fh_hetop *Fit Fay-Herriot Heteroskedastic Ordered Probit (FH-HETOP) Model using JAGS*

Description

Fits the FH-HETOP model described by Lockwood, Castellano and Shear (2018) using the jags function in R2jags.

Usage

```
fh_hetop(ngk, fixedcuts, p, m, gridL, gridU, Xm=NULL, Xs=NULL,
seed=12345, modelfileonly = FALSE, modloc=NULL, ...)
```

Arguments

ngk	Numeric matrix of dimension G x K in which column k of row g indicates the number of units from group g falling into category k.
fixedcuts	A vector of length 2 providing the first two cutpoints, to identify the location and scale of the group parameters. Note that this suffices for any K >= 3.
p	Vector of length 2 giving degrees of freedom for cubic spline basis to parameterize Efron priors for group means and group standard deviations; see References.
m	Vector of length 2 giving number of grid points to parameterize Efron priors for group means and group standard deviations; see References.
gridL	Vector of length 2 of lower bounds for grids to parameterize Efron priors for group means and group standard deviations; see References.
gridU	Vector of length 2 of upper bounds for grids to parameterize Efron priors for group means and group standard deviations; see References.
Xm	Optional matrix of covariates for the group means.
Xs	Optional matrix of covariates for the log group standard deviations.
seed	Passed to set.seed.
modelfileonly	If TRUE, function returns location of JAGS model file only, without running JAGS. Default is FALSE.
modloc	Optional character vector of length 1 providing the full path to the name of file where the JAGS model code will be written. Defaults to NULL, in which case the code will be written to a temporary file.
...	Additional arguments to jags.

Details

The function is basically a wrapper for jags, building model code depending on the specification of the Efron priors and any covariates for the group means and group standard deviations. Details on the FH-HETOP model are provided by Lockwood, Castellano and Shear (2018).

Covariates to predict the group means and group log standard deviations are optional. However, `Xm` and `Xs` must both be either NULL, or specified; the current version of this function cannot use covariates to predict one set of parameters but not use any covariates to predict the other set. While covariates in general must be present or absent simultaneously for the two sets of parameters, it is not necessary that the same covariates be used to predict the two sets of parameters. All covariates must be centered so that they sum to zero across groups.

Value

A object of class `rjags`, with additional information specific to the FH-HETOP model. The additional information is stored as a list called `fh_hetop_extras` with the following components:

`Finfo`	A list containing information used to estimate the population distribution of the residuals from the FH-HETOP model. Note that the posterior samples of the parameters defining the residual distribution can be found in the `BUGSoutput` element of the returned object.
`Dinfo`	A list containing information about the data used to the fit the model, including the counts, covariates and fixed cutpoints.
`waicinfo`	A list containing information about the WAIC for the estimated model; see help file for `waic_hetop`.
`est_star_samps`	A list with posterior samples of parameters with respect to the 'star' scale which defines the location and scale of the group means and standard deviations that corresponds to a marginal population mean of zero and marginal population standard deviation of 1. Additional details in help file for `mle_hetop`
`est_star_mug`	A dataframe containing various estimates of the group means on the 'star' scale, including posterior means, Constrained Bayes and Triple-Goal estimates. Additional details in help file for `triple_goal`.
`est_star_sigmag`	A dataframe containing various estimates of the group standard deviations on the 'star' scale, including posterior means, Constrained Bayes and Triple-Goal estimates. Additional details in help file for `triple_goal`.

Author(s)

J.R. Lockwood <jrlockwood@ets.org>

References

Efron B. (2016). "Empirical Bayes deconvolution estimates," *Biometrika* 103(1):1–20.

Lockwood J.R., Castellano K.E. and Shear B.R. (2018). "Flexible Bayesian models for inferences from coarsened, group-level achievement data," *Journal of Educational and Behavioral Statistics.* 43(6):663–692.

See Also

`jags`

Examples

```
set.seed(1001)

## define mean-centered covariates
G  <- 12
z1 <- sample(c(0,1), size=G, replace=TRUE)
z2 <- 0.5*z1 + rnorm(G)
Z  <- cbind(z1 - mean(z1), z2 = z2 - mean(z2))

## define true parameters dependent on covariates
beta_m    <- c(0.3,  0.8)
beta_s    <- c(0.1, -0.1)
mug       <- Z[,1]*beta_m[1] + Z[,2]*beta_m[2] + rnorm(G, sd=0.3)
sigmag    <- exp(0.3 + Z[,1]*beta_s[1] + Z[,2]*beta_s[2] + 0.2*rt(G, df=7))
cutpoints <- c(-1.0, 0.0, 1.2)

## generate data
ng   <- rep(200,G)
ngk  <- gendata_hetop(G, K = 4, ng, mug, sigmag, cutpoints)
print(ngk)

## fit FH-HETOP model including covariates
## NOTE: using an extremely small number of iterations for testing,
##       so that convergence is not expected
m <- fh_hetop(ngk, fixedcuts = c(-1.0, 0.0), p = c(10,10),
              m = c(100, 100), gridL = c(-5.0, log(0.10)),
              gridU = c(5.0, log(5.0)), Xm = Z, Xs = Z,
              n.iter = 100, n.burnin = 50)

print(m)
print(names(m$fh_hetop_extras))

s <- m$BUGSoutput$summary
print(data.frame(truth = c(beta_m, beta_s), s[grep("beta", rownames(s)),]))

print(cor(mug,    s[grep("mu",    rownames(s)),"mean"]))
print(cor(sigmag, s[grep("sigma", rownames(s)),"mean"]))

## manual calculation of WAIC (see help file for waic_hetop)
tmp <- waic_hetop(ngk, m$BUGSoutput$sims.matrix)
identical(tmp, m$fh_hetop_extras$waicinfo)
```

---

gendata_hetop *Generate count data from Heteroskedastic Ordered Probit (HETOP) Model*

Description

Generates count data for G groups and K ordinal categories under a heteroskedastic ordered probit model, given the total number of units in each group and parameters determining the category probabilities for each group.

Usage

```
gendata_hetop(G, K, ng, mug, sigmag, cutpoints)
```

Arguments

G	Number of groups.
K	Number of ordinal categories.
ng	Vector of length G providing the total number of units in each group.
mug	Vector of length G giving the latent variable mean for each group.
sigmag	Vector of length G giving the latent variable standard deviation for each group.
cutpoints	Vector of length (K-1) giving cutpoint locations, held constant across groups, that map the continuous latent variable to the observed categorical variable.

Details

For each group g, the function generates ng IID normal random variables with mean mug[g] and standard deviation sigmag[g], and then assigns each to one of K ordered groups, depending on cutpoints. The resulting data for a group is a table of category counts summing to ng[g].

Value

A G x K matrix where column k of row g provides the number of simulated units from group g falling into category k.

Author(s)

J.R. Lockwood <jrlockwood@ets.org>

References

Reardon S., Shear B.R., Castellano K.E. and Ho A.D. (2017). "Using heteroskedastic ordered probit models to recover moments of continuous test score distributions from coarsened data," *Journal of Educational and Behavioral Statistics* 42(1):3–45.

Lockwood J.R., Castellano K.E. and Shear B.R. (2018). "Flexible Bayesian models for inferences from coarsened, group-level achievement data," *Journal of Educational and Behavioral Statistics.* 43(6):663–692.

Examples

```
set.seed(1001)

## define true parameters
G         <- 10
mug       <- seq(from= -2.0, to= 2.0, length=G)
sigmag    <- seq(from=  2.0, to= 0.8, length=G)
cutpoints <- c(-1.0, 0.0, 0.8)

## generate data with large counts
```

```
ng   <- rep(100000,G)
ngk  <- gendata_hetop(G, K = 4, ng, mug, sigmag, cutpoints)
print(ngk)

## compare theoretical and empirical cell probabilities
phat  <- ngk / ng
ptrue <- t(sapply(1:G, function(g){
    tmp <- c(pnorm(cutpoints, mug[g], sigmag[g]), 1)
    c(tmp[1], diff(tmp))
}))
print(max(abs(phat - ptrue)))
```

`mle_hetop` *Maximum Likelihood Estimation of Heteroskedastic Ordered Probit (HETOP) Model*

Description

Computes MLEs of `G` group means and standard deviations using count data from `K` ordinal categories under a heteroskedastic ordered probit model. Estimation is conducted conditional on two fixed cutpoints, and additional constraints on group parameters are imposed if needed to achieve identification in the presence of sparse counts.

Usage

```
mle_hetop(ngk, fixedcuts, svals=NULL, iterlim = 1500, ...)
```

Arguments

`ngk`	Numeric matrix of dimension `G x K` in which column `k` of row `g` indicates the number of units from group `g` falling into category `k`.
`fixedcuts`	A vector of length 2 providing the first two cutpoints, to identify the location and scale of the group parameters. Note that this suffices for any `K >= 3`.
`svals`	Optional vector of starting values. Its length is `2G + (K-3)` when no groups have sparse counts that affect identifiability; otherwise it must be smaller. See Details.
`iterlim`	Maximum number of iterations used in optimization (passed to `nlm`).
`...`	Any other arguments for `nlm`.

Details

This function requires `K >= 3`. If `ngk` has all nonzero counts, all model parameters are identified. Alternatively, arbitrary identification rules are required to ensure the existence of the MLE when there are one or more groups with nonzero counts in fewer than three categories. This function adopts the following rules. For any group with nonzero counts in fewer than three categories, the log of the group standard deviation is constrained to equal the mean of the log standard deviations for the remaining groups. Further constraints are imposed to handle groups for which all data fall

into either the lowest or highest category. Let S be the set of groups for which it is not the case that all data fall into an extreme category. Then for any group with all data in the lowest category, the mean for that group is constrained to be the minimum of the group means over S. Similarly, for any group with all data in the highest category, the mean for that grou is constrained to be the maximum of the group means over S.

The location and scale of the group means are identified for the purpose of conducting the estimation by fixing two of the cutpoints. However in practice it may be desirable to express the group means and standard deviations on a scale that is more easily interpreted; see Reardon et al. (2017) for details. This function reports estimates on four different scales: (1) the original estimation scale with two fixed cutpoints; (2) a scale defined by forcing the group means and log group standard deviations each to have weighted mean of zero, where weights are proportional to the total count for each group; (3) a scale where the population mean of the latent variable is zero and the population standard deviation is one; and (4) a scale similar to (3) but where a bias correction is applied. See Reardon et al. (2017) for details on this bias correction.

The function also returns an estimated intracluster correlation (ICC) of the latent variable, defined as the ratio of the between-group variance of the latent variable to its marginal variance. Scales (1)-(3) above lead to the same estimated ICC; scale (4) uses a bias-corrected estimate of the ICC which will not in general equal the estimate from scales (1)-(3).

Value

A list with the following components:

`est_fc`	A list of estimated group means, group standard deviations, cutpoints and ICC on scale (1).
`est_zero`	A list of estimated group means, group standard deviations, cutpoints and ICC on scale (2).
`est_star`	A list of estimated group means, group standard deviations, cutpoints and ICC on scale (3).
`est_starbc`	A list of estimated group means, group standard deviations, cutpoints and ICC on scale (4).
`nlmdetails`	The object returned by `nlm` that summarizes detailed of the optimization.
`pstatus`	A dataframe, with one row for each group, summarizing the estimation status of the mean and standard deviation for each group. A value of `est` means that the parameter was estimated without constraints. A value of `mean`, used for the group standard deviations, indicates that the parameter was constrained. Values of `min` or `max`, used for the group means, indicate that the parameter was constrained.

Author(s)

J.R. Lockwood <jrlockwood@ets.org>

References

Reardon S., Shear B.R., Castellano K.E. and Ho A.D. (2017). "Using heteroskedastic ordered probit models to recover moments of continuous test score distributions from coarsened data," *Journal of Educational and Behavioral Statistics* 42(1):3–45.

Lockwood J.R., Castellano K.E. and Shear B.R. (2018). "Flexible Bayesian models for inferences from coarsened, group-level achievement data," *Journal of Educational and Behavioral Statistics.* 43(6):663–692.

Examples

```
set.seed(1001)

## define true parameters
G         <- 10
mug       <- seq(from= -2.0, to= 2.0, length=G)
sigmag    <- seq(from=  2.0, to= 0.8, length=G)
cutpoints <- c(-1.0, 0.0, 0.8)

## generate data with large counts
ng   <- rep(100000,G)
ngk  <- gendata_hetop(G, K = 4, ng, mug, sigmag, cutpoints)
print(ngk)

## compute MLE and check parameter recovery:
m    <- mle_hetop(ngk, fixedcuts = c(-1.0, 0.0))
print(cbind(true = mug,       est = m$est_fc$mug))
print(cbind(true = sigmag,    est = m$est_fc$sigmag))
print(cbind(true = cutpoints, est = m$est_fc$cutpoints))

## estimates on other scales:
p    <- ng/sum(ng)
print(sum(p * m$est_zero$mug))
print(sum(p * log(m$est_zero$sigmag)))

print(sum(p * m$est_star$mug))
print(sum(p * (m$est_star$mug^2 + m$est_star$sigmag^2)))

## dealing with sparse counts
ngk_sparse <- matrix(rpois(G*4, lambda=5), ncol=4)
ngk_sparse[1,] <- c(5,8,0,0)
ngk_sparse[2,] <- c(0,10,10,0)
ngk_sparse[3,] <- c(12,0,0,0)
ngk_sparse[4,] <- c(0,0,0,10)
print(ngk_sparse)

m    <- mle_hetop(ngk_sparse, fixedcuts = c(-1.0, 0.0))
print(m$pstatus)
print(unique(m$est_fc$sigmag[1:4]))
print(exp(mean(log(m$est_fc$sigmag[5:10]))))
print(m$est_fc$mug[3])
print(min(m$est_fc$mug[-3]))
print(m$est_fc$mug[4])
print(max(m$est_fc$mug[-4]))
```

`triple_goal` *Shen and Louis (1998) Triple Goal Estimators*

Description

`triple_goal` implements the "Triple Goal" estimates of Shen and Louis (1998) for a vector of parameters given a sample from the posterior distribution of those parameters. Also computes "constrained Bayes" estimators of Ghosh (1992).

Usage

```
triple_goal(s, stop.if.ties = FALSE, quantile.type = 7)
```

Arguments

`s`	A `(n x K)` matrix of `n` samples of `K` group parameters with no missing values.
`stop.if.ties`	logical; if TRUE, function stops if any units have identical posterior mean ranks; otherwise breaks ties at random.
`quantile.type`	`type` argument to `quantile` function for different methods of computing quantiles.

Details

In typical applications, the matrix `s` will be a sample of size `n` from the joint posterior distribution of a vector of `K` group-specific parameters. Both the triple goal and constrained Bayes estimators are designed to mitigate problems arising from underdispersion of posterior means; see references.

Value

A dataframe with `K` rows with fields:

`theta_pm`	Posterior mean estimates of group parameters.
`theta_psd`	Posterior standard deviation estimates of group parameters.
`theta_cb`	"Constrained Bayes" estimates of group parameters using formula in Shen and Louis (1998).
`theta_gr`	"Triple Goal" estimates of group parameters using algorithm defined in Shen and Louis (1998).
`rbar`	Posterior means of ranks of group parameters (1=lowest).
`rhat`	Integer ranks of group parameters (=rank(rbar)).

Author(s)

J.R. Lockwood <jrlockwood@ets.org>

References

Shen W. and Louis T.A. (1998). "Triple-goal estimates in two-stage hierarchical models," *Journal of the Royal Statistical Society, Series B* 60(2):455-471.

Ghosh M. (1992). "Constrained Bayes estimation with applications," *Journal of the American Statistical Association* 87(418):533-540.

Examples

```
set.seed(1001)
.K <- 50
.nsamp <- 500
.theta_true <- rnorm(.K)
.s <- matrix(.theta_true, ncol=.K, nrow=.nsamp, byrow=TRUE) +
      matrix(rnorm(.K*.nsamp, sd=0.4), ncol=.K, nrow=.nsamp)
.e <- triple_goal(.s)
str(.e)
head(.e)
```

---

waic_hetop *WAIC for FH-HETOP model*

Description

Computes the Watanabe-Akaike information criterion (WAIC) for the FH-HETOP model using the data and posterior samples of the group means, group standard deviations and cutpoints.

Usage

```
waic_hetop(ngk, samps)
```

Arguments

ngk	Numeric matrix of dimension `G x K` in which column `k` of row `g` indicates the number of units from group `g` falling into category `k`.
samps	A matrix of posterior samples that includes at least the group means, group standard deviations and the cutpoints. Column names for these three collections of parameters must contain the strings 'mu', 'sigma' and 'cuts', respectively.

Details

Although this function can be called directly by the user, it is primarily intended to be used to compute WAIC as part of the function `fh_hetop`. Details on the WAIC calculation are provided by Vehtari and Gelman (2017).

Value

A list with the following components:

lpd_hat	Part 1 of the WAIC calculation: the estimated log pointwise predictive density, summed across groups.
phat_waic	Part 2 of the WAIC calculation: the effective number of parameters.
waic	The WAIC criterion: -2 times (lpd_hat - phat_waic).

Author(s)

J.R. Lockwood <jrlockwood@ets.org>

References

Lockwood J.R., Castellano K.E. and Shear B.R. (2018). "Flexible Bayesian models for inferences from coarsened, group-level achievement data," *Journal of Educational and Behavioral Statistics.* 43(6):663–692.

Vehtari A., Gelman A. and Gabry J. (2017). "Practical Bayesian model evaluation using leave-one-out cross-validation and WAIC," *Statistics and Computing.* 27(5):1413–1432.

Examples

```
## example call using data 'ngk' and FH-HETOP model object 'm'
## (demonstrated in examples for fh_hetop):
##
## waic_hetop(ngk, m$BUGSoutput$sims.matrix)
```

Index