

Package: binest (via r-universe)

June 8, 2026

Type Package

Title Estimation of Group Means and SDs from Binned Count Data

Version 0.2-1

Date 2026-05-27

Depends R (>= 3.5.0), splines, stats

Suggests knitr, rmarkdown, R2jags

Description Estimates group-level means and standard deviations from binned (coarsened) count data, where the within-bin scores are unobserved. The package implements three methods that share a common output structure: `bin_means()` (a fast estimator that assumes within-district normality and uses pooled bin proportions to derive bin-conditional truncated-normal expectations), `mle_hetop()` (maximum likelihood for the heteroskedastic ordered probit model of Reardon, Shear, Castellano and Ho 2017 <[doi:10.3102/1076998616666279](https://doi.org/10.3102/1076998616666279)>), and `fh_hetop()` (the Bayesian Fay-Herriot variant of Lockwood, Castellano and Shear 2018 <[doi:10.3102/1076998618795124](https://doi.org/10.3102/1076998618795124)>). The `mle_hetop()` and `fh_hetop()` functions are forked from the 'HETOP' package by J. R. Lockwood ('CRAN', last released 2019). `mle_hetop()` has been modified to speed up the runtime via a vectorized inner loop and to remove two user-facing arguments (`fixedcuts` and `svals`) that some users found confusing; cutpoints and starting values are now derived internally from the data.

License GPL (>= 2)

VignetteBuilder knitr

Encoding UTF-8

NeedsCompilation no

Author Paul T. von Hippel [aut, cre], David J. Hunter [aut], J.R. Lockwood [aut] (Original HETOP package author)

Maintainer Paul T. von Hippel <ph3828@eid.utexas.edu>

Repository <https://cran.r-universe.dev>

Date/Publication 2026-06-08 18:20:08 UTC

RemoteUrl <https://github.com/cran/binest>

RemoteRef HEAD

RemoteSha b1ddee8fa564b6f09bbc0f88ce52394b073b2685

Contents

| | |
|---------------------------|----|
| binest-package | 2 |
| bin_means | 3 |
| fh_hetop | 5 |
| gendata_hetop | 8 |
| mle_hetop | 10 |
| triple_goal | 13 |
| tx_g6_math_2018 | 14 |
| waic_hetop | 15 |

Index 17

| | |
|----------------|---|
| binest-package | <i>Estimation of Group Means and SDs from Binned Count Data</i> |
|----------------|---|

Description

Estimates group-level means and standard deviations from binned (coarsened) count data, where the within-bin scores are unobserved. The package implements three methods that share a common output structure:

- **bin_means**: a fast per-group estimator under within-group normality. Runs in time linear in the number of groups times the number of bins. This is the preferred estimator in the package.
- **mle_hetop**: maximum-likelihood fit of the heteroskedastic ordered probit (HETOP) model of Reardon, Shear, Castellano and Ho (2017).
- **fh_hetop**: the Bayesian Fay-Herriot variant of HETOP by Lockwood, Castellano and Shear (2018).

The `mle_hetop` and `fh_hetop` functions are forked from the HETOP package by J. R. Lockwood (CRAN, last released 2019). `mle_hetop` has been modified to speed up its runtime via a vectorized inner loop and to remove two user-facing arguments (`fixedcuts` and `svals`) that some users found confusing; cutpoints and starting values are now derived internally from the data.

`mle_hetop` and `fh_hetop` are superseded by `bin_means` and remain in the package for comparison purposes. See `vignette("binest")` for an empirical comparison on Texas STAAR Grade-6 mathematics data.

Bundled data

The package ships with `tx_g6_math_2018`, a district-level dataset of bin counts and reported mean scale scores from the 2017-18 administration of the State of Texas Assessments of Academic Readiness (STAAR) Grade-6 mathematics test. See the vignette for usage.

Author(s)

Paul T. von Hippel <ph3828@eid.utexas.edu>, David J. Hunter, and J. R. Lockwood.

References

Fisher, R. A. (1922). On the mathematical foundations of theoretical statistics. *Philosophical Transactions of the Royal Society of London A*, 222, 309-368.

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.

Reardon, S. F., 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.

Sheppard, W. F. (1898). On the calculation of the most probable values of frequency-constants for data arranged according to equidistant divisions of a scale. *Proceedings of the London Mathematical Society*, 29, 353-380.

bin_means

Fast Estimation of Group Means and SDs from Binned Counts

Description

Estimates G group means and standard deviations from count data in K ordinal categories, under the assumption that within each group the underlying scores are normally distributed. Cutpoints are either supplied by the caller or derived from the pooled bin proportions.

`bin_means` is the preferred estimator in this package and supersedes `mle_hetop` and `fh_hetop`: it runs much faster than either HETOP variant, produces an estimate for every group with at least three populated bins, and on real data has been found to be at least as accurate as either HETOP variant. The HETOP functions remain in the package for comparison purposes.

Usage

```
bin_means(ngk, cutpoints = NULL, eb_shrink = FALSE,
          iterate = FALSE, tol = 1e-3, maxit = 100)
```

Arguments

| | |
|-----------|--|
| ngk | Numeric matrix of dimension $G \times K$ in which column k of row g indicates the number of units from group g falling into category k . |
| cutpoints | Optional numeric vector of length $K-1$. If NULL (default), cutpoints are derived from the pooled bin proportions via <code>qnorm(cumsum(colSums(ngk)/sum(ngk))[1:(K-1)])</code> and the output is on a standardized scale (latent population mean 0, total variance 1). If supplied, the cutpoints are treated as known on the test-score scale, and the output <code>est_raw</code> is on the test-score scale. |

| | |
|-----------|---|
| eb_shrink | Logical. If TRUE, applies empirical-Bayes shrinkage to the per-group estimates. Each group's log-SD is shrunk toward the population-weighted mean of log-SDs with weight equal to $\tau^2 / (\tau^2 + s_g^2)$, where τ^2 is the moment estimator of between-group log-SD variance and s_g^2 is the per-group sampling variance of the log-SD MLE, derived from the Fisher information of the binned-normal likelihood. The per-group mean is shrunk analogously, with the additional constraint of a per-group MAP step under the actual multinomial bin likelihood. Default FALSE. |
| iterate | Logical. If TRUE and cutpoints = NULL, the function refines the data-derived cutpoints by alternating between (a) fitting per-group means and SDs given current cutpoints and (b) matching the pooled bin proportions to the implied mixture-of-normals CDF. Iterates until cutpoints change by less than tol. Not valid when cutpoints are supplied. Default FALSE; iteration did not meaningfully improve per-group estimates in our tests on real or simulated data. |
| tol | Convergence tolerance (on the standardized scale) for the per-group MLE iteration and, when iterate = TRUE, the cutpoint-refinement EM. Default 1e-3. |
| maxit | Maximum number of iterations. Default 100. |

Details

The function derives $K-1$ cutpoints on a standardized scale (mean 0, SD 1) by applying `qnorm` to the cumulative pooled bin proportions `cumsum(colSums(ngk)/sum(ngk))[1:(K-1)]`. For each bin k with cutpoints $c_{\{k-1\}}$ and c_k , the function then computes the truncated-normal first and second moments:

$$p_k = \frac{\Phi(c_k) - \Phi(c_{\{k-1\}})}{E[Z|k]} = \frac{\phi(c_{\{k-1\}}) - \phi(c_k)}{p_k} \quad E[Z^2|k] = 1 + \frac{(c_{\{k-1\}} * \phi(c_{\{k-1\}}) - c_k * \phi(c_k))}{p_k}$$

For each group g , the estimated mean is the within-group bin-proportion-weighted average of $E[Z|k]$, and the estimated variance is the within-group bin-proportion-weighted average of $E[Z^2|k]$ minus the squared estimated mean.

Estimates are returned on two scales.

Value

A list with the following components:

| | |
|-----------|---|
| est_raw | Estimates on the raw (test-score) scale when cutpoints were supplied, or on the standardized scale otherwise. A list with elements <code>group_mean_<suffix></code> , <code>group_sd_<suffix></code> , <code>cutpoints</code> , and <code>icc</code> (intracluster correlation), where <code><suffix></code> is <code>mle</code> when <code>eb_shrink = FALSE</code> and <code>eb</code> when <code>eb_shrink = TRUE</code> . |
| est_std | Estimates on the standardized scale where the population-weighted state mean is 0 and the total (within + between) state SD is 1. Same elements as <code>est_raw</code> . |
| gof | Per-group Pearson chi-square goodness-of-fit of the within-group normality assumption: a data frame with columns <code>chisq</code> , <code>df</code> , and <code>p</code> . NA when the test is unidentified (fewer than four populated bins). |
| iter_info | Diagnostic flags and tuning parameters from the fit, including <code>within</code> and <code>eb_shrink</code> . |

Author(s)

Paul T. von Hippel and David J. Hunter.

References

Sheppard W.F. (1898). On the calculation of the most probable values of frequency-constants for data arranged according to equidistant divisions of a scale. *Proceedings of the London Mathematical Society*, 29, 353-380.

Fisher R.A. (1922). On the mathematical foundations of theoretical statistics. *Philosophical Transactions of the Royal Society A*, 222, 309-368.

Examples

```
set.seed(1001)
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)
ng <- rep(1000, G)
ngk <- gendata_hetop(G, K = 4, ng, mug, sigmag, cutpoints)

bm <- bin_means(ngk)
print(cbind(true = mug, est = bm$est_raw$group_mean_mle))
print(cbind(true = sigmag, est = bm$est_raw$group_sd_mle))
print(cbind(true = cutpoints, est = bm$est_raw$cutpoints))
```

 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 the suggested package `R2jags`. Requires JAGS (a system binary, not an R package) to be installed; see <https://sourceforge.net/projects/mcmc-jags/>.

Note: `fh_hetop` has been superseded by `bin_means`, which runs much faster, requires no external dependencies, and on real data has been found to be at least as accurate as `fh_hetop`. `fh_hetop` is retained in the package for comparison purposes.

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 $G \times K$ 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 $K \geq 3$. |
| 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 <code>set.seed</code> . |
| 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 <code>R2jags::jags</code> . |

Details

The function is basically a wrapper for `R2jags::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 <code>BUGSoutput</code> element of the returned object. |
|-------|---|

| | |
|-----------------|--|
| Dinfo | A list containing information about the data used to 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

R2jags::jags

Examples

```
## Not run:
## fh_hetop() requires JAGS, an external system binary; see
## https://sourceforge.net/projects/mcmc-jags/. The example below
## is wrapped in \dontrun{} so that it is not executed by R CMD
## check, but should run interactively once JAGS is installed.

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)

## End(Not run)

```

| | |
|---------------|--|
| gendata_hetop | <i>Generate count data from Heteroskedastic Ordered Probit (HETOP) Model</i> |
|---------------|--|

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 $G \times K$ 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.

This implementation is forked from the HETOP package by J. R. Lockwood (CRAN, last released 2019). We have modified the original code in two ways: (1) the inner cell-probability loop is vectorized, which substantially speeds up the runtime per likelihood evaluation; and (2) the user-facing arguments `fixedcuts` and `svals` have been removed, because some users found them confusing and supplying incompatible values caused silent optimization failures. Cutpoints and starting values are now derived internally from the data.

Note: `mle_hetop` has been superseded by `bin_means`, which runs much faster, produces an estimate for every identified group, and on real data has been found to be at least as accurate as `mle_hetop`. `mle_hetop` is retained in the package for comparison purposes.

Usage

```
mle_hetop(ngk, iterlim = 1500, ...)
```

Arguments

| | |
|----------------------|--|
| <code>ngk</code> | Numeric matrix of dimension $G \times K$ in which column k of row g indicates the number of units from group g falling into category k . |
| <code>iterlim</code> | Maximum number of iterations used in optimization (passed to <code>nlm</code>). |
| <code>...</code> | Any other arguments for <code>nlm</code> . |

Details

This function requires $K \geq 3$. 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 group 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. This function derives the two fixed cutpoints internally from the pooled bin proportions via `qnorm(cumsum(colSums(ngk)/sum(ngk))[1:2])`, which places them on the same standardized scale as the internal starting values for the group means and log standard deviations. 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 intraclass 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:

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

Author(s)

J. R. Lockwood (original implementation); David J. Hunter and Paul T. von Hippel <ph3828@eid.utexas.edu> (vectorization and API simplification).

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 (cutpoints derived from data):
m <- mle_hetop(ngk)
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)
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)
```

 tx_g6_math_2018

Texas STAAR Grade-6 Mathematics, 2017-18: District-Level Bin Counts

Description

District-level counts of students in each of four proficiency categories on the Texas State of Texas Assessments of Academic Readiness (STAAR) Grade-6 mathematics test, 2017-18 administration. For each district the dataset also reports the average scale score across all tested students, which can be used as ground truth for evaluating estimators that recover district means from binned counts.

Usage

```
data(tx_g6_math_2018)
```

Format

A data frame with 1151 rows and 8 columns:

district_id Sequential integer identifier (1 to 1151).

district_name District name (character).

n_tested Total students tested in the district.

unsatisfactory Students with scale score below 1536 (proficiency category "Did Not Meet Grade Level").

approaches Students with scale score in [1536, 1653) ("Approaches Grade Level").

meets Students with scale score in [1653, 1772) ("Meets Grade Level").

masters Students with scale score \geq 1772 ("Masters Grade Level").

reported_mean District average scale score, computed by the Texas Education Agency from individual student scores.

Details

The three published cut scores defining the bin boundaries are 1536, 1653, and 1772. The administrative floor of the STAAR scale is 1062 and the ceiling is 2143. Of the 1151 districts, 1014 have nonzero counts in all four bins, 120 have nonzero counts in three bins, and 17 have nonzero counts in two bins.

Source

Texas Education Agency, Academic Performance Reports (TAPR), 2017-18. Compiled by D.\ J.\ Hunter and P.\ T.\ von Hippel.

Examples

```
data(tx_g6_math_2018)
str(tx_g6_math_2018)

## Recover district means using bin means with known cutpoints.
ngk <- with(tx_g6_math_2018,
            cbind(unsatisfactory, approaches, meets, masters))
fit <- bin_means(ngk, cutpoints = c(1536, 1653, 1772))

## Correlation with reported truth on the test-score scale.
## (Districts with fewer than three populated bins are NA-coded by
## bin_means; use complete.obs for the comparison.)
cor(fit$est_raw$group_mean_mle, tx_g6_math_2018$reported_mean,
    use = "complete.obs")
```

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 $G \times K$ 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:

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

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

```
if (requireNamespace("R2jags", quietly = TRUE)) {
  set.seed(42)
  G <- 10
  ngk <- gendata_hetop(G = G, K = 4, nj = rep(50, G),
    mug = rnorm(G), sigmag = exp(rnorm(G, 0, 0.2)),
    cutpoints = c(-1, 0, 1))$ngk
  m <- fh_hetop(ngk, fixedcuts = c(-1, 0),
    p = c(10, 10), m = c(100, 100),
    gridL = c(-5, log(0.10)), gridU = c(5, log(5.0)),
    n.iter = 200, n.burnin = 100, seed = 1)
  waic <- waic_hetop(ngk, m$BUGSoutput$sims.matrix)
  print(waic)
}
```

Index

- * **datasets**
 - tx_g6_math_2018, 14
- * **models**
 - bin_means, 3
 - fh_hetop, 5
 - mle_hetop, 10
- * **package**
 - binest-package, 2
- * **utilities**
 - gendata_hetop, 8
 - triple_goal, 13
 - waic_hetop, 15

bin_means, 2, 3, 5, 10

binest (binest-package), 2

binest-package, 2

fh_hetop, 2, 3, 5

gendata_hetop, 8

mle_hetop, 2, 3, 7, 10

nlm, 10, 11

quantile, 13

set.seed, 6

triple_goal, 7, 13

tx_g6_math_2018, 2, 14

waic_hetop, 7, 15