Package 'basket'

February 17, 2019

Title Basket Trial Analysis

Version 0.0.18

Description
Implemention of multisource exchangeability models for basket trial design and monitoring.

License LGPL-2

Maintainer Michael J. Kane <mkane@presagia.ai>

Encoding UTF-8

Imports foreach, GenSA, rjags, ggplot2, stats, GGally, tibble, tidyr, dplyr, igraph, gridExtra, itertools

LazyData true

RoxygenNote 6.1.1

Suggests knitr, rmarkdown, testthat,

VignetteBuilder knitr

Author Brian Hobbs [aut], Michael Kane [aut, cre]

R topics documented:

basket *Fit Basket Trial Data*

Description

TODO FINISH THIS

Usage

```
basket(formula, data, p0, method = mem_empirical_bayes, ...)
```

Arguments

formula	the number of responders, enrollees, and possibly basket names for a basket trial.
data	the basket trial data
p0	the null (or failure probability) for the basket trial.
method	the method for fitting the basket trial. This could be mem_empirical_bayes, mem_full_bayes, sem, or a custom fitting function.
...	other parameters to be passed to the supplied method argument.

Examples

```
# WRITE THIS
# fit <- basket(responders ~ evaluable | basket, data = vemu_wide)
```

basket_name *The Names of the Baskets*

Description

Retrieve the basket names in an exchangeability model.

Usage

```
basket_name(model)
```

Arguments

model	the model to retrieve the basket names of

Examples

```
#' # 5 baskets, each with enrollement size 5
trial_sizes <- rep(5, 5)

# The response rates for the baskets.
resp_rate <- 0.15

# The trials: a column of the number of responses and a column of the
# the size of each trial.
trials <- data.frame(
  responses = rbinom(trial_sizes, trial_sizes, resp_rate),
  size = trial_sizes,
  name = paste("Basket", seq_len(5))
)

mem_empirical_bayes(trials$responses, trials$size, trials$basket)
```

---

cluster_mean *Calculate the Exchangeability Cluster Means*

Description

TODO FINISH THIS

Usage

```
cluster_mean(x, method = "maximizer")
```

Arguments

x	the exchangeability model
method	the exchangeability model to get the clusters for. This can be either "maximzer" or "pep" if a model includes a posterior exchangeability probability.

Examples

```
# WRITE THIS
```

---

eb_reference *Reference Outputs for Testing*

Description

There are three reference data sets used for testing: eb_reference, eb_reference ub, and fb_reference. These are the the outputs generated by the reference code (in inst/code-from-brian) and are used to detect changes in the output of our implementation.

kbeta *The K-Beta Disribution*

Description

Density, distribution function, quantile function, and random geration for the K-Beta distribution with shape parameters shape1, shape2, a, and b. The shape parameters are the same as those of the beta distribution. The a parameter defines the smallest value in the support of the distribution and the b parameter defines the largest.

Usage

```
dkbeta(x, shape1, shape2, a = -1, b = 1, log = FALSE)
pkbeta(q, shape1, shape2, a = -1, b = 1, lower.tail = TRUE, log.p = FALSE)
qkbeta(p, shape1, shape2, a = -1, b = 1, lower.tail = TRUE, log.p = FALSE)
rkbeta(n, shape1, shape2, a = -1, b = 1)
```

Arguments

x	vector of values in the support.
shape1	the first non-negative parameters of the Beta distribution.
shape2	the second non-negative parameters of the Beta distribution.
a	the lower bound of the support of the distribution (default -1).
b	the upper bound of the support of the distribution (default 1).
log	should the distribution value be shown on the log scale? (default FALSE)
q	vector of quantiles.
p	vector of probabilities.
n	number of observations. If 'length(n) > 1', the length is taken to be the number required.
log.p	logical; if TRUE, probabilities p are given as log(p).
lower.tail	logical; if TRUE (default), probabilities are P[X <= x], otherwise, P[X > x].

Examples

```
library(ggplot2)
a <- 1
b <- 3
shape1 <- 0.5
shape2 <- 0.5
x <- seq(a, b, by = 0.01)
d <- data.frame(x = x, y = dkbeta(x, 0.5, 0.5, a, b))
ggplot(d, aes(x = x, y = y)) +
  geom_area(fill = "black", alpha = 0.7) +
  xlim(0, 3)
```

`mem_empirical_bayes` *MEM Empirical Bayes*

Description

Fit the MEM model using empirical Bayesian inference.

Usage

```
mem_empirical_bayes(responses, size, name = NULL, p0 = 0.25,
  shape1 = 0.5, shape2 = 0.5, hpd_alpha = 0.05, upper_bound = 1,
  lower_bound = 0, call = NULL)
```

Arguments

`responses`	the number of responses in each basket.
`size`	the size of each basket.
`name`	the name of each basket (default: NULL - no basket names).
`p0`	the null response rate for the poster probability calculation (default 0.25).
`shape1`	the first shape parameter(s) for the prior of each basket (default 0.5).
`shape2`	the second shape parameter(s) for the prior of each basket (default 0.5).
`hpd_alpha`	the highest posterior density trial significance.
`upper_bound`	for constrained empirical Bayes, the upper bound on the inclusion probability (default 1).
`lower_bound`	for constrained empirical Bayes, the lower bound on the inclusion probability (default 0).
`call`	the call of the function (default NULL).

Examples

```
# 5 baskets, each with enrollement size 5
trial_sizes <- rep(5, 5)

# The response rates for the baskets.
resp_rate <- 0.15

# The trials: a column of the number of responses and a column of the
# the size of each trial.
trials <- data.frame(
  responses = rbinom(trial_sizes, trial_sizes, resp_rate),
  size = trial_sizes
)

mem_empirical_bayes(trials$responses, trials$size)
```

`mem_full_bayes_exact` *MEM Full Bayes Exact*

Description

Fit the MEM model using full Bayesian inference.

Usage

```
mem_full_bayes_exact(responses, size, name, p0 = 0.15, shape1 = 0.5,
  shape2 = 0.5, prior_inclusion = diag(length(responses))/2 +
  matrix(0.5, nrow = length(responses), ncol = length(responses)),
  hpd_alpha = 0.05, alternative = "greater", seed = 1000,
  call = NULL)
```

Arguments

`responses`	the number of responses in each basket.
`size`	the size of each basket.
`name`	the name of each basket.
`p0`	the null response rate for the poster probability calculation (default 0.15).
`shape1`	the first shape parameter(s) for the prior of each basket (default 0.5).
`shape2`	the second shape parameter(s) for the prior of each basket (default 0.5).
`prior_inclusion`	the matrix giving the prior inclusion probability for each pair of baskets. The default is on on the main diagonal and 0.5 elsewhere.
`hpd_alpha`	the highest posterior density trial significance.
`call`	the call of the function (default NULL).

Examples

```
# 5 baskets, each with enrollement size 5
trial_sizes <- rep(5, 5)

# The response rates for the baskets.
resp_rate <- 0.15

# The trials: a column of the number of responses and a column of the
# the size of each trial.
trials <- data.frame(
  responses = rbinom(trial_sizes, trial_sizes, resp_rate),
  size = trial_sizes
)

mem_full_bayes(trials$responses, trials$size)
```

`mem_full_bayes_mcmc` *MEM Full Bayes MCMC method*

Description

Fit the MEM model using full Bayesian Metropolis-Hasting MCMC inference.

Usage

```
mem_full_bayes_mcmc(responses, size, name, p0 = 0.15, shape1 = 0.5,
  shape2 = 0.5, Prior = diag(length(responses))/2 + matrix(0.5, nrow =
  length(responses), ncol = length(responses)), HPD.alpha = 0.05,
  alternative = "greater", niter.MCMC = 10000, Initial = NA,
  seed = 1000, call = NULL)
```

Arguments

`responses`	the number of responses in each basket.
`size`	the size of each basket.
`name`	the name of each basket.
`p0`	the null response rate for the poster probability calculation (default 0.15).
`shape1`	the first shape parameter(s) for the prior of each basket (default 0.5).
`shape2`	the second shape parameter(s) for the prior of each basket (default 0.5).
`alternative`	the alternative case defination (default greater)
`niter.MCMC`	the number of MCMC iterations.
`Initial`	the initial MEM matrix.
`seed`	the random number seed.
`call`	the call of the function.
`prior`	the matrix giving the prior inclusion probability for each pair of baskets. The default is on on the main diagonal and 0.5 elsewhere.
`hpd_alpha`	the highest posterior density trial significance.

Examples

```
# 5 baskets, each with enrollement size 5
trial_sizes <- rep(5, 5)

# The response rates for the baskets.
resp_rate <- 0.15

# The trials: a column of the number of responses and a column of the
# the size of each trial.
trials <- data.frame(
  responses = rbinom(trial_sizes, trial_sizes, resp_rate),
  size = trial_sizes
)
mem_full_bayes_mcmc(trials$responses, trials$size)
```

plot_all_exchangeability *Plot the Prior, MAP, and PEP of a Basket Trial*

Description

Plot the Prior, MAP, and PEP of a Basket Trial

Usage

```
plot_all_exchangeability(x, plotList, ...)
```

Arguments

x	the exchangeability model.
...	other options. See Details for more information.

Details

TODO: WRITE THIS

plot_density *Plot the Densities of Baskets in a Trial*

Description

TODO: WRITE THIS

Usage

```
plot_density(x, ...)
```

Arguments

x	the exchangeability model.
...	other options. See Details for more information.

Details

TODO WRITE THIS TALK ABOUT ... OPTIONS

Examples

```
# TODO: WRITE THIS
```

plot_exchangeability *Plot the Map Exchangeability of a Basket Trial*

Description

TODO: WRITE THIS

Usage

```
plot_exchangeability(x, ...)
```

Arguments

x	the exchangeability model.
...	other options. See Details for more information.

Details

TODO: WRITE THIS

Examples

```
# WRITE THIS
```

plot_posterior_exchangeability *Plot the Posterior Exchangeability of a Basket Trial*

Description

TODO: WRITE THIS

Usage

```
plot_posterior_exchangeability(x, ...)
```

Arguments

x	the exchangeability model.
...	other options. See Details for more information.

Details

TODO: WRITE THIS

Examples

```
# WRITE THIS
```

sample_posterior *Sample Posterior Samples from a Basket Trial*

Description

TODO FINISH THIS

Usage

```
sample_posterior(model, num_samples = 10000)
```

Arguments

model	the exchangeability model
num_samples	the number of samples to draw. Default 10000

Examples

```
# WRITE THIS
```

sem *Single-Source Model*

Description

Fit the single source model.

Usage

```
sem(responses, size, name, p0 = 0.25, mu = 0, M = 0, inv_sig = 1,
  num_chains = 3, quiet = TRUE, num_iterations = 3000,
  progress_bar = interactive())
```

Arguments

responses	the number of responses in each basket.
size	the size of each basket.
name	the name of each basket (default: NULL - no basket names).
p0	the null response rate for the poster probability calculation (default 0.25).
mu	WHAT IS THIS?
M	WHAT IS THIS?
inv_sig	WHAT IS THIS?
num_chains	the number of MCMC chains to run.
quiet	should the function be run with extra output messages? Default: FALSE.
num_iterations	the number MCMC steps to take per chain.
progress_bar	shoud the progress bar be shown during execution? Default: interactive()

References

Berry SM, Broglio KR, Groshen S, Berry DA. Bayesian hierarchical modeling of patient subpopulations: efficient designs of phase II oncology clinical trials. Clinical Trials. 2013 Oct;10(5):720-34.

`summary.full_bayes` *Make the summary table*

Description

From the input full_bayes class object, summarize the CDF, HPD, ESS, Mean, and Median results.

Usage

```
## S3 method for class 'full_bayes'
summary(res)
```

Arguments

`res`	the full_bayes class object..

`update_result` *Update Full Bayes results with different p0 values*

Description

Update Full Bayes results with different p0 values and alternative

Usage

```
update_result(res, p0 = 0.15, alternative = "greater")
```

Arguments

`p0`	the null response rate for the poster probability calculation (default 0.15).
`alternative`	the alternative case defination (default greater)

Examples

```
MHResult1New <- updateResult(MHResult1, 0.25)
```

vemu *Summary Data from the Vemurafinib Study*

Description

The 'vemu' and 'vemu_wide' data sets provides response information taken from the "Vemurafenib in multiple nonmelanoma cancers with braf v600 mutations" study where, in total, 18 responders were observed among the 84 patients contributing evaluable outcomes for statistical estimation. Observed response rates varied from $42%$ and $43%$ for baskets of NSCLC and ECD or LCH to 0 and $4%,$ for CRC with vemurafenib mono and combination therapies, respectively. Two responders of seven patients, ATC was associated with a $29%$ response rate, while one responder of eight patients was observed in the cholangiocarcinoma basket. Contrasting favorable results for preliminary vemurafenib activity among NSCLC and ECD or LCH patients with less favorable results for CRC patients, the authors concluded that nonmelanoma tumor types harboring $BRAF^V600$ mutations failed to respond uniformly to BRAF-targeted therapy giving credence to more conventional organ-specific nosology when compared to molecular tumor nosology.

Later, in the "Statistical challenges posed by basket trials: sensitivity analysis of the Vemurafenib study" it was shown that patient-enrollment types we likely drove the negative results for several targets, rather than Vemurafinib itself.

References

Hyman DM, Puzanov I, Subbiah V, Faris JE, Chau I, Blay JY, Wolf J, Raje NS, Diamond EL, Hollebecque A, et al. Vemurafenib in multiple nonmelanoma cancers with braf v600 mutations. New England Journal of Medicine 2015; **373**(8):726–736.

Hobbs BP, Kane MJ, Hong DS, and Landin R. Statistical challenges posed by basket trials: sensitivity analysis of the Vemurafenib study. *Accepted to the Annals of Clinical Oncology* 2018.

Index