Package: basket (via r-universe)

November 17, 2026

Title Basket Trial Analysis

Version 0.12.2

Description Implementation of multisource exchangeability models for Bayesian analyses of prespecified subgroups arising in the context of basket trial design and monitoring. The R 'basket' package facilitates implementation of the binary, symmetric multi-source exchangeability model (MEM) with posterior inference arising from both exact computation and Markov chain Monte Carlo sampling. Analysis output includes full posterior samples as well as posterior probabilities, highest posterior density (HPD) interval boundaries, effective sample sizes (ESS), mean and median estimations, posterior exchangeability probability matrices, and maximum a posteriori MEMs. In addition to providing ``basketwise" analyses, the package includes similar calculations for ``clusterwise" analyses for which subgroups are combined into meta-baskets, or clusters, using graphical clustering algorithms that treat the posterior exchangeability probabilities as edge weights. In addition plotting tools are provided to visualize basket and cluster densities as well as their exchangeability. References include Hyman, D.M., Puzanov, I., Subbiah, V., Faris, J.E., Chau, I., Blay, J.Y., Wolf, J., Raje, N.S., Diamond, E.L., Hollebecque, A. and Gervais, R (2015) <doi:10.1056/NEJMoa1502309>; Hobbs, B.P. and Landin, R. (2018) <doi:10.1002/sim.7893>; Hobbs, B.P., Kane, M.J., Hong, D.S. and Landin, R. (2018) <doi:10.1093/annonc/mdy457>; and Kaizer, A.M., Koopmeiners, J.S. and Hobbs, B.P. (2017) <doi:10.1093/biostatistics/kxx031>.

Depends R (>= 3.5.0)

License LGPL-2

Maintainer Michael J. Kane <michael.kane@yale.edu>

Imports GenSA, foreach, ggplot2, stats, tibble, tidyr, dplyr, igraph, gridExtra, itertools, crayon, cli, RColorBrewer, tidygraph, parallel, doParallel, ggraph, partitions, pscl

LazyData true

Encoding UTF-8

RoxygenNote 7.2.3

URL https://github.com/kaneplusplus/basket

BugReports https://github.com/kaneplusplus/basket/issues

Suggests knitr, rmarkdown, testthat (>= 3.0.0),

VignetteBuilder knitr

Config/testthat/edition 3

Config/pak/sysreqs
libfontconfig1-dev libfreetype6-dev libglpk-dev libgmp3-dev libicu-dev libxml2-dev

Repository https://kaneplusplus.r-universe.dev

Date/Publication 2023-07-17 15:42:01 UTC

RemoteUrl https://github.com/kaneplusplus/basket

RemoteRef HEAD

RemoteSha a665c37a79a193dbc8af33fb40f750b12abefaf4

Contents

basket-package . . . 3
basket . . . 4
basket_map . . . 6
basket_name . . . 6
basket_pep . . . 7
cluster_baskets . . . 8
cluster_map . . . 9
cluster_membership . . . 9
cluster_pep . . . 10
get_seed . . . 11
mem_exact . . . 11
mem_mcmc . . . 12
plot_density . . . 14
plot_map . . . 15
plot_mem . . . 16
plot_pep . . . 16
plot_pep_graph . . . 17
sample_posterior . . . 18
update_p0 . . . 19
vemu . . . 20

Index **21**

basket-package *Basket Trial Analysis*

Description

The R basket package provides for the designs and analysis of basket trials for multi-source exchangeability models (MEM) <https://doi.org/10.1093/biostatistics/kxx031> allowing arms to "share" power with similar arms in a trial. The package is intended to perform the exact or MCMC computation of the operating characteristics of different scenarios. Calculations derived from these analyses include the posterior probabilities, HPD boundaries, effective sample sizes (ESS), mean and median estimations can be calculated with this package using the MEM method. Along with providing "basketwise" analyses, the package includes similar calculations for "clusterwise" analyses where a cluster a set of similar baskets. In addition plotting tools are provided to visualize basket and cluster density as well as their exchangeability. The package includes the following main functions:

- [basket_name()] Get the basket names in an analysis
- [basket_pep()] Get the Posterior Exchangeability Probability (PEP) matrix for the arms in the trial.
- [basket_map()] Get the Maximum A Posteriori (MAP) matrix for the arms in a trial.
- [cluster_pep()] Get the Posterior Exchangeability Probability (PEP) matrix for the arms in the trial.
- [cluster_map()] Get the Maximum A Posteriori (MAP) matrix for the arms in a trial.
- [sample_posterior()] Sample from the posterior distribution of the arms in the trial.
- [mem_exact()] Create a basket analysis using the exact method.
- [mem_mcmc()] Create a basket analysis using the exact method.
- [summary()] Summarize the results of an analysis.
- [update_p0()] Update the null that a basket response rate is above a specified value.
- [plot_density()] Plot the estimated densities of arms or clusters.
- [plot_pep()] Show the exchangeogram of the PEP matrix.
- [plot_mem()] Plot the arm prior, MAP, and PEP of a basket trial.

References

Vemurafenib in multiple nonmelanoma cancers with braf v600 mutations Hyman DM, Puzanov I, Subbiah V, Faris JE, Chau I, Blay JY, Wolf J, Raje NS, Diamond EL, Hollebecque A, et al. New England Journal of Medicine 2015; 373(8):726–736. <https://doi:10.1056/NEJMoa1502309>

Bayesian basket trial design with exchangeability monitoring BP Hobbs, R Landin Statistics in medicine 37 (25), 3557-357. <https://doi.org/10.1002/sim.7893>

Statistical challenges posed by uncontrolled master protocols: sensitivity analysis of the vemurafenib study BP Hobbs, MJ Kane, DS Hong, R Landin Annals of Oncology 29 (12), 2296-2301. <doi:10.1093/annonc/mdy457>

Bayesian hierarchical modeling based on multisource exchangeability AM Kaizer, JS Koopmeiners, BP Hobbs Biostatistics 19 (2), 169-184. <https://doi.org/10.1093/biostatistics/kxx031>

basket *Create a Basket Trial Analysis*

Description

This function creates an analysis modeling the exchangeability and distribution characteristics of cohorts in a basket trial, where a single therapy is given to multiple cohorts. The model is based on the multi-source exchangeability model. This is a generalization of the Bayesian hierarchical model and it accomodates multiple sets of latent factors shared combinations of cohorts.

Usage

```
basket(
  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",
  call = NULL,
  cluster_function = cluster_membership,
  method = c("mcmc", "lmem", "exact"),
  mcmc_iter = 250000,
  mcmc_burnin = 50000,
  initial_mem = round(prior - 0.001),
  cluster_analysis = FALSE,
  parallelRun = FALSE,
  seed = get_seed()
)
```

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	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. (default 0.05)

alternative	the alternative case definition (default "greater").
call	the call of the function. (default NULL)
cluster_function	a function to cluster baskets.
method	"mcmc", "lmem" (local MEM), or "exact". See details for an explanation. (default "mcmc")
mcmc_iter	if the method is "mcmc" then this spcifies the number of mcmc iterations. Otherwise, it is ignored. (default 200000)
mcmc_burnin	if the method is "mcmc" then this specifies the number of burn-in iterations. (default 50000)
initial_mem	if the method is "mcmc" then this spcifies the initial MEM matrix. Otherwise, it is ignored.
cluster_analysis	if the cluster analysis is conducted.
parallelRun	if the computation is conduected in parallel mode
seed	the random seed for the mcmc calculations. By default this is .Random.seed. If this value is not initialized, then it is first initialized with Sys.time() and then returned.

Details

The model may be fit using an exact calculation via mcmc, or a local version of MEM. The first conducts posterior inference through the entire set of exchangeability relationships in the sample domain. This approach is computationally feasible only when the number of cohorts is relatively small. As a rule of thumb this option should be used with a maximum of 20 cohorts. By default, the second(mcmc) is used and it is based on the Metropolis algorithm and it extends the model's implementation to larger collections of subpopulations. The algorithm initiates with a burn-in period (see mcmc_burnin), which are discarded from the analysis. The third is based on a recent paper by Wei et al. The calculation is computationally efficient and tends to borrow to a lesser extent than other implemented methods.

Examples

```
# 3 baskets, each with enrollement size 5
trial_sizes <- rep(5, 3)

# 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 = letters[1:3]
)

b = basket(trials$responses, trials$size, trials$name)
```

```
summary(b)
```

---

basket_map *Get the Basketwise Maximum A Posteriori Probability Matrix*

Description

MEM analyses include the maximum a posterior exchangeability probability (MAP) of included arms indicating whether two arms in the trial are exchangeable. This function returns the matrix of those relationships.

Usage

```
basket_map(x)
```

Arguments

x	either an exchangeability model or basket object.

Examples

```
# Create an MEM analysis of the Vemurafenib trial data.
data(vemu_wide)

mem_analysis <- mem_exact(
  vemu_wide$responders,
  vemu_wide$evaluable,
  vemu_wide$baskets
)

# Get basket MAPs.
basket_map(mem_analysis)
```

---

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

```
# 3 baskets, each with enrollement size 5
trial_sizes <- rep(5, 3)

# 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(3))
)

basket_name(mem_mcmc(trials$responses, trials$size, trials$basket))
```

---

basket_pep *The Basketwise Posterior Exchangeability Probability Matrix*

Description

MEM analyses include the posterior exchangeability probability (PEP) of included arms giving the probability that any two arms are exchangeable. This function returns the matrix of those probabilities.

Usage

```
basket_pep(x)
```

Arguments

x	either an exchangeability model or basket object.

Examples

```
# Create an MEM analysis of the Vemurafenib trial data.
data(vemu_wide)

mem_analysis <- mem_exact(
  vemu_wide$responders,
  vemu_wide$evaluable,
  vemu_wide$baskets
)
```

```
# Get the PEP for baskets.
basket_pep(mem_analysis)
```

`cluster_baskets` *Get the Cluster Members of MEM Models*

Description

Object returned by the ‘mem_mcmc()‘ and ‘mem_exact()‘ include information about the arms in the trials and the cluster composed of sets of similar arms. This function returns the name of each arm in a cluster.

Usage

```
cluster_baskets(x)
```

Arguments

`x`	either an exchangeability model or basket object.

Value

A named list is returned where the name is the cluster name and each element of the list is comprise do a character vector of the baskets in each cluster.

Examples

```
# Create an MEM analysis of the Vemurafenib trial data.
data(vemu_wide)

mem_analysis <- mem_exact(
  vemu_wide$responders,
  vemu_wide$evaluable,
  vemu_wide$baskets,
  cluster_analysis = TRUE
)

# Get the baskets in the clusters.
cluster_baskets(mem_analysis)
```

cluster_map *Get the Clusterwise Maximum A Posteriori Probability Matrix*

Description

MEM analyses include the maximum a posterior exchangeability probability (MAP) of included arms indicating whether two arms in the trial are exchangeable. This function returns the matrix of those relationships.

Usage

```
cluster_map(x)
```

Arguments

x	either an exchangeability model or basket object.

Examples

```
# Create an MEM analysis of the Vemurafenib trial data.
data(vemu_wide)

mem_analysis <- mem_exact(
  vemu_wide$responders,
  vemu_wide$evaluable,
  vemu_wide$baskets
)

# Get the cluster MAPs.
cluster_map(mem_analysis)
```

cluster_membership *Cluster Baskets Based on the Posterior Exchangeabilities*

Description

This is the default function used to cluster cohorts in the `basket`, `mem_mcmc`, and `mem_exact` functions. The approach creates a graph where each vertex is a cohort and the weight between two cohorts is determined by their posterior exchangeability probability. The graph is then clustered using **igraph**'s `louvain` function, which determines the number of clusters and the cluster memberships, and has been shown to perform well with real clinical data.

Usage

```
cluster_membership(m)
```

Arguments

m	the adjacency matrix.

Value

A factor variable with cluster memberships for each cohort in the study.

See Also

basket mem_mcmc mem_exact

---

cluster_pep *Get the Clusterwise Posterior Exchangeability Matrix*

Description

MEM analyses include the posterior exchangeability probability (PEP) of clusters of arms giving the probability that any two arms are exchangeable. This function returns the matrix of those probabilities.

Usage

```
cluster_pep(x)
```

Arguments

x	either an exchangeability model or basket object.

Examples

```
# Create an MEM analysis of the Vemurafenib trial data.
data(vemu_wide)

mem_analysis <- mem_exact(
  vemu_wide$responders,
  vemu_wide$evaluable,
  vemu_wide$baskets
)

# Get cluster PEPs.
basket_pep(mem_analysis)
```

get_seed *Get the Current Random Seed*

Description

Get the value of the current random seed. If one has not been initialized, then initialize it and return the new seed value.

Usage

```
get_seed()
```

mem_exact *Fit the Exact MEM Model*

Description

Fit the MEM model using full Bayesian inference.

Usage

```
mem_exact(
  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",
  seed = 1000,
  cluster_analysis = FALSE,
  call = NULL,
  cluster_function = cluster_membership
)
```

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	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.
alternative	the alternative case definition (default greater)
seed	the random number seed.
cluster_analysis	if the cluster analysis is conducted.
call	the call of the function (default NULL).
cluster_function	a function to cluster baskets

See Also

cluster_membership

Examples

```
# 3 baskets, each with enrollement size 5
trial_sizes <- rep(5, 3)

# 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 = letters[1:3]
)

summary(mem_exact(trials$responses, trials$size, trials$name))
```

---

mem_mcmc *Fit the MEM Model using MCMC*

Description

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

Usage

```
mem_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",
  mcmc_iter = 250000,
  mcmc_burnin = 50000,
  initial_mem = round(prior - 0.001),
  seed = 1000,
  cluster_analysis = FALSE,
  call = NULL,
  cluster_function = cluster_membership,
  parallelRun = FALSE
)
```

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	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.
alternative	the alternative case definition (default greater)
mcmc_iter	the number of total MCMC iterations (includes the number of MCMC burn_in iterations).
mcmc_burnin	the number of MCMC Burn_in iterations.
initial_mem	the initial MEM matrix.
seed	the random number seed.
cluster_analysis	if the cluster analysis is conducted.
call	the call of the function.
cluster_function	a function to cluster baskets
parallelRun	if the computation is conduected in parallel mode

Examples

```
# 3 baskets, each with enrollement size 5
trial_sizes <- rep(5, 3)

# 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 = letters[1:3]
)
res <- mem_mcmc(trials$responses, trials$size)
```

---

plot_density *Plot the Response Densities in Basket Trials*

Description

The MEM analysis calculates the probability of exchangeability of baskets and clusters in a basket trial. This function creates density plots of the response rates of each basket or each cluster under the MEM design taking into account the extent to which power can be borrowed from similar trials.

Usage

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

Arguments

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

Details

The ... options can be used to specify the colors of the response density plot or, when plotting an object of class ‘exchangeability_model‘ the type can be specified. In this case, the default is ‘type = c("both", "basket", "cluster")‘.

Examples

```
# Create an MEM analysis of the Vemurafenib trial data.
data(vemu_wide)

mem_analysis <- mem_exact(
  vemu_wide$responders,
```

```
  vemu_wide$evaluable,
  vemu_wide$baskets
)

plot_density(mem_analysis)
```

`plot_map` *Plot the Map Exchangeability of a Basket Trial*

Description

The Maximum A Posteriori Probability (MAP) of an MEM is the estimate of the exchangeability structure of a basket trial. This function visualizes this matrix as an exchangeogram.

Usage

```
plot_map(x, ...)
```

Arguments

`x`	`basket` element of the exchangeability model.
`...`	other options passed to ggplot2 to alter the visual

Details

The `plot_map` function attempts to place the basket names to the left of the main diagonal in a way that makes it easy to read. However, for especially long basket names options are provided. Here is a list of all options available to "fine tune" the visualizations. These auxiliary options include:

- [palette]A color palette consisting of 3 colors: the first corresponds to a low degree of exchangeability, the second to 50 exchangeability, and the third to a high degree of exchangeability. Interpolation between these colors is performed for intermediary degrees of exchangeability.
- [text_color]A text string setting the color of the exchangeability values printed on the plot. (Default "white")
- [tile_color]A text string setting the color of the edges of the tiles. (Default "white")
- [expand] The proportion to expand the viewport (Default expand = c(0.3, 0.3))
- [text_size] The text size. (Default 4)
- [legend_position] The legend position. (Default legend_position = c(0.25, 0.8)
- [draw_legend] Should the legend be drawn? (Default TRUE)
- [basket_name_hoffset] The horizontal offset of the basket names.. (Default 0)
- [basket_name_hjust] The basket name justification.. (Default 1 - right justified)

Examples

```
# Create an MEM analysis of the Vemurafenib trial data.
data(vemu_wide)

mem_analysis <- mem_exact(
  vemu_wide$responders,
  vemu_wide$evaluable,
  vemu_wide$baskets
)

plot_map(mem_analysis$basket)
```

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

Description

: Plot the Prior, MAP, and PEP Matrices

Usage

```
plot_mem(x, type = c("prior", "map", "pep"), ...)
```

Arguments

`x`	the exchangeability model.
`type`	the plot type that will be plotted.
`...`	other options. See Details for more information.

`plot_pep` *Plot the Posterior Exchangeability of a Basket Trial*

Description

The posterior exchangeability of the baskets in a MEM analysis can be visualized via an exchangeogram using this function.

Usage

```
plot_pep(x, ...)
```

Arguments

`x`	`basket` element of the exchangeability model.
`...`	other options passed to ggplot2 to alter the visual characteristics of the plot. See Details for more information.

Details

The plot_pep function attempts to place the basket names to the left of the main diagonal in a way that makes it easy to read. However, for especially long basket names options are provided. Here is a list of all options available to "fine tune" the visualizations. These auxiliary options include:

- [palette] A color palette consisting of 3 colors: the first corresponds to a low degree of exchangeability, the second to 50 exchangeability, and the third to a high degree of exchangeability. Interpolation between these colors is performed for intermediary degrees of exchangeability. [text_color]A text string setting the color of the exchangeability values printed on the plot. (Default "white") [tile_color]A text string setting the color of the edges of the tiles. (Default "white") (Default RColorBrewer::brewer.pal(3, "BuGn"))
- [expand] The proportion to expand the viewport (Default expand = c(0.3, 0.3))
- [text_size] The text size. (Default 4)
- [legend_position] The legend position. (Default legend_position = c(0.25, 0.8)
- [draw_legend] Should the legend be drawn? (Default TRUE)
- [basket_name_hoffset] The horizontal offset of the basket names.. (Default 0)
- [basket_name_hjust] The basket name justification.. (Default 1 - right justified)

Examples

```
# Create an MEM analysis of the Vemurafenib trial data.
data(vemu_wide)

mem_analysis <- mem_exact(
  vemu_wide$responders,
  vemu_wide$evaluable,
  vemu_wide$baskets
)

plot_pep(mem_analysis$basket)
```

---

plot_pep_graph *Plot a Network Graph of the PEP Matrix*

Description

Plot a Network Graph of the PEP Matrix

Usage

```
plot_pep_graph(
  x,
  color_by = c("post_prob", "mean_est", "median_est"),
  layout = c("fr", "nicely", "kk", "drl"),
  pep_cutoff = 0
)
```

Arguments

x	the exchangeability model.
color_by	which variable to color by. One of "post_prob", "mean_est", "median_est".
layout	the layout algorithm to use for the graph. One of
pep_cutoff	a value between 0 and 1 indicating the cutoff for PEP above which edges of the graph will be drawn.

Examples

```
# Create an MEM analysis of the Vemurafenib trial data.
data(vemu_wide)

mem_analysis <- mem_exact(
  vemu_wide$responders,
  vemu_wide$evaluable,
  vemu_wide$baskets
)

plot_pep_graph(mem_analysis)
```

---

sample_posterior *Sample Posterior Samples from a Basket Trial*

Description

Sample Posterior Samples from a Basket Trial

Usage

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

Arguments

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

Examples

```
# 3 baskets, each with enrollement size 5
trial_sizes <- rep(5, 3)

# 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(3))
)
```

---

update_p0 *Update Full Bayes results with different p0 values*

Description

After running either 'mem_mcmc' or 'mem_exact', the test can be updated without rerunning the entire analysis. This function provides updating of both the null response rate along with the alternative rerunning relevant test.

Usage

```
update_p0(res, p0 = 0.15, alternative = "greater")
```

Arguments

res	the result of an mem analysis.
p0	the null response rate for the poster probability calculation (default 0.15).
alternative	the alternative case definition (default greater)

Examples

```
## Not run:
# Create an MEM analysis of the Vemurafenib trial data.
data(vemu_wide)

mem_analysis <- mem_exact(
  vemu_wide$responders,
  vemu_wide$evaluable,
  vemu_wide$baskets
)

# Update the null from p0 = 0.15 the default, to p = 0.25.
update_p0(mem_analysis, 0.20)

## End(Not run)
```

vemu *Summary Data from the Vemurafenib 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 Vemurafenib 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

Index

basket, 4
basket-package, 3
basket_map, 6
basket_name, 6
basket_pep, 7

cluster_baskets, 8
cluster_map, 9
cluster_membership, 9
cluster_pep, 10

get_seed, 11

mem_exact, 11
mem_mcmc, 12

plot_density, 14
plot_map, 15
plot_mem, 16
plot_pep, 16
plot_pep_graph, 17

sample_posterior, 18

update_p0, 19

vemu, 20
vemu_wide *(vemu)*, 20