# Package ‘BHTSpack’

February 8, 2021

**Type** Package

**Title** Bayesian Multi-Plate High-Throughput Screening of Compounds

**Version** 0.6

**Description** Can be used for joint identification of candidate compound hits from multiple assays, in drug discovery. This package implements the framework of I. D. Shterev, D. B. Dunson, C. Chan and G. D. Sempowski. “Bayesian Multi-Plate High-Throughput Screening of Compounds”. Scientific Reports 8(1):9551, 2018. This project was funded by the Division of Allergy, Immunology, and Transplantation, National Institute of Allergy and Infectious Diseases, National Institutes of Health, Department of Health and Human Services, under contract No. HHSN272201400054C entitled ‘`Adjuvant Discovery For Vaccines Against West Nile Virus and Influenza’’, awarded to Duke University and lead by Drs. Herman Staats and Soman Abraham.

**Depends** R (>= 3.2.3), R2HTML (>= 2.3.2), xtable (>= 1.8-2)

**VignetteBuilder** knitr

**Suggests** knitr

**License** GPL-3

**LazyLoad** yes

**NeedsCompilation** yes

**Author** Ivo D. Shterev [aut, cre],
            David B. Dunson [aut],
            Cliburn Chan [aut],
            Gregory D. Sempowski [aut]

**Maintainer** Ivo D. Shterev <i.shterev@gmail.com>

**Repository** CRAN

**Date/Publication** 2021-02-08 16:30:13 UTC

### R topics documented:

- BHTSpack-package ........................................... 2
- abfun ............................................................ 3
- alpha.u ....................................................... 4
Description

Can be used for joint identification of candidate hits from multiple assays, in drug discovery. This package implements the framework of I. D. Shterev, D. B. Dunson, C. Chan and G. D. Sempowski. "Bayesian Multi Plate High Throughput Screening of Compounds", arXiv:1709.10041, September 2017. This project was funded by the Division of Allergy, Immunology, and Transplantation, National Institute of Allergy and Infectious Diseases, National Institutes of Health, Department of Health and Human Services, under contract No. HHSN272201400054C entitled "Adjuvant Discovery For Vaccines Against West Nile Virus and Influenza", awarded to Duke University and lead by Drs. Herman Staats and Soman Abraham.

Details

The DESCRIPTION file:

Package: BHTSpack
Type: Package
Title: Bayesian Multi-Plate High-Throughput Screening of Compounds
Version: 0.6
Authors@R: c(person(c("Ivo", "D."), "Shterev", role = c("aut", "cre"), email = "i.shterev@gmail.com"), person(c("David", "B."), "Dunson", role = "aut"), person("Cliburn", "Chan", role = "aut"), person(c("Gregory", "D."), "Sempowski", role = "aut"))
Description: Can be used for joint identification of candidate compound hits from multiple assays, in drug discovery. This package implements the framework of I. D. Shterev, D. B. Dunson, C. Chan and G. D. Sempowski. "Bayesian Multi Plate High Throughput Screening of Compounds", arXiv:1709.10041, September 2017. This project was funded by the Division of Allergy, Immunology, and Transplantation, National Institute of Allergy and Infectious Diseases, National Institutes of Health, Department of Health and Human Services, under contract No. HHSN272201400054C entitled "Adjuvant Discovery For Vaccines Against West Nile Virus and Influenza", awarded to Duke University and lead by Drs. Herman Staats and Soman Abraham.
Depends: R (>= 3.2.3), R2HTML (>= 2.3.2), xtable (>= 1.8-2)
VignetteBuilder: knitr
Suggests: knitr
Index of help topics:

- **BHTSpack-package**: Bayesian Multi-Plate High-Throughput Screening of Compounds
- **abfun**: package internal function
- **alpha.u**: package internal function
- **b.u**: package internal function
- **bhts**: Bayesian High-Throughput Screening
- **bhts2HTML**: Convert to HTML
- **data.create**: Create Synthetic Data
- **fdr.r**: package internal function
- **h.pr.u**: package internal function
- **hatpai.u**: package internal function
- **ind.u**: package internal function
- **lambda.u**: package internal function
- **lg.mu.sig**: package internal function
- **mu.k.u**: package internal function
- **nu.u**: package internal function
- **pai.u**: package internal function
- **ptrace**: Trace (ACF) Plots
- **r.fdr**: Significant Hits
- **sig.k.u**: package internal function
- **tau.u**: package internal function
- **z.pr.u**: package internal function

**Author(s)**

I. D. Shterev, D. B. Dunson, C. Chan and G. D. Sempowski

Maintainer: I. D. Shterev <i.shterev@duke.edu>

**References**

Usage

\texttt{abfun(m, v)}

Arguments

\texttt{m} \hspace{1cm} \text{Description}
\texttt{v} \hspace{1cm} \text{Description}

Examples

\texttt{abfun(0.26, 10^{-4})}

---

\texttt{alpha.u} \hspace{1cm} \textit{package internal function}

Description

package internal function

Usage

\texttt{alpha.u(nu, a, b, H)}

Arguments

\texttt{nu} \hspace{1cm} \text{Description}
\texttt{a} \hspace{1cm} \text{Description}
\texttt{b} \hspace{1cm} \text{Description}
\texttt{H} \hspace{1cm} \text{Description}

Examples

\texttt{M = 5}
\texttt{H = 10}
\texttt{a = 10^{-6}}
\texttt{b = 10^{-6}}
\texttt{nu = lapply(1:M, function(x){rbeta(H, a, b)})}
\texttt{alpha.u(nu, a, b, H)}
Description

package internal function

Usage

b.u(hatpai)

Arguments

hatpai

Examples

pai = 0.5
M = 10
H = 10
K = 5
n = 100

z = abs(rnorm(n))

sigma1 = abs(rnorm(K))
sigma0 = abs(rnorm(K))

mu1 = abs(rnorm(K))
mu0 = abs(rnorm(K))

hk0 = matrix(sample(K, M*H, replace=TRUE), M, H)
hk1 = matrix(sample(K, M*H, replace=TRUE), M, H)

nu.h0 = lapply(1:H, function(x){rbeta(1,5,5)})
uh1 = lapply(1:H, function(x){rbeta(1,5,5)})

ph0 = lapply(nu.h0, lambda.u)
ph1 = lapply(nu.h1, lambda.u)

b.u(hatpai.u(z, hk1, hk0, ph1, ph0, sigma1, sigma0, mu1, mu0, pai, H, n))
Bayesian High-Throughput Screening

Description

This is the package main function.

Usage

bhts(Z, iters, H, K, mu00=NULL, mu10=NULL, a.alpha, b.alpha, a.tau, b.tau,
pnorm=FALSE, s=NULL, store=FALSE)

Arguments

Z       A list of compounds.
itters  Number of iterations to perform.
H       Number of local DP components.
K       Number of global DP components.
mu00    Activity level (mean) of non-hit compounds
mu10    Activity level (mean) of hit compounds
a.alpha Gamma shape parameter specifying local DP concentration prior.
b.alpha Gamma rate parameter specifying local DP concentration prior.
a.tau   Gamma shape parameter specifying global DP concentration prior.
b.tau   Gamma rate parameter specifying global DP concentration prior.
pnorm   Plate normalization. If TRUE, each plate is normalized to zero mean and unit variance, prior to analysis. Default is FALSE.
s       Random seed (for reproducibility purposes). Default is NULL.
store   If TRUE, all samples of certain latent variables are stored in the output object. Default is FALSE.

Value

This function returns a list consisting of the following elements:

hatpai A list of vectors of posterior probabilities, estimating the probability of a compound being a hit.
dat.store If store = TRUE (default is FALSE), the output contains a list of iters x K matrices of samples. Each matrix contains the samples of a separate latent variable. At each iteration, the following six variables are stored in a different row of their corresponding matrix, (λ_1^0, ..., λ_K^0), (a_1^1, ..., λ_K^1), (μ_01, ..., μ_0K), (μ_11, ..., μ_1K), (σ_0^2_1, ..., σ_0^2_K) and (σ_1^2, ..., σ_K^2).
Examples

```r
set.seed(1234)
Nmax = 100
M = 100
n = sample(Nmax, M, replace=TRUE)
Z = lapply(n, function(x){abs(rnorm(x))})
bhts(Z, iters=100, H=10, K=5, mu00=0, mu10=10, a.alpha=10, b.alpha=5, a.tau=10, b.tau=5)
```

---

**bhts2HTML**

*Convert to HTML*

---

**Description**

This function creates an HTML file.

**Usage**

```r
bhts2HTML(dat, dir, fname, title=NULL, bgcolor="#BBBBEE")
```

**Arguments**

- `dat`: An object which is the output of `bhts()`.
- `dir`: Directory in which to store the file.
- `fname`: File name.
- `title`: The title of the html file.
- `bgcolor`: Color for the html background.

**Examples**

```r
#See package vignette
```

---

**data.create**

*Create Synthetic Data*

---

**Description**

This function generates synthetic compound data.

**Usage**

```r
data.create(N, nr, nc, M, p, s=NULL, covrow=NULL, covcol=NULL, c=0.0001, mat=FALSE)
```
Arguments

- N: Number of compounds per plate.
- nr: Number of plate rows.
- nc: Number of plate columns.
- M: Number of plates.
- p: Probability of a compound being a hit.
- s: Random seed (for reproducibility purposes). Default is NULL.
- covrow: Noise plate row-covariance matrix. Default is NULL.
- covcol: Noise plate column-covariance matrix. Default is NULL.
- c: Constant for scaling plate noise. Default is 0.0001.
- mat: Specifies a matrix (TRUE) or a vector (FALSE) plate format. Default is FALSE.

Value

This function returns a list consisting of the following elements:

- Z: A list of matrices (mat = TRUE) or vectors (mat = FALSE) of compounds.
- B: A list of compound indicators specifying a hit (1) or a non-hit (0).
- I: A list of compound indicators specifying the mixture component (from 1 to K).

Examples

#See package vignette

---

**Description**

package internal function

**Usage**

fdr.r(r, hatpai, fdr)

**Arguments**

- r: Description
- hatpai: Description
- fdr: Description

**Examples**

#See package vignette
**h.pr.u**  
*package internal function*

**Description**

package internal function

**Usage**

```
h.pr.u(z, ih, mu, sigma, pk, K, H, n)
```

**Arguments**

- `z`
- `ih`
- `mu`
- `sigma`
- `pk`
- `K`
- `H`
- `n`

**Examples**

```r
Nmax = 100
K = 5
H = 10
M = 20
n = sample(Nmax, M, replace=TRUE)

z = abs(rnorm(sum(n)))
mu = abs(rnorm(K))
sigma = 1/rgamma(n=K, shape=10, rate=10)

ih = sample(H,sum(n), replace=TRUE)

pk = lambda.u(rbeta(K, 1, 1))

h.pr.u(z, ih, mu, sigma, pk, K, H, n)
```
hatpai.u

package internal function

Description

package internal function

Usage

hatpai.u(z, hk1, hk0, ph1, ph0, sigma1, sigma0, mu1, mu0, pai, H, n)

Arguments

- z
- hk1
- hk0
- ph1
- ph0
- sigma1
- sigma0
- mu1
- mu0
- pai
- H
- n

Value

value

Examples

pai = 0.5
M = 10
H = 10
K = 5
n = 100

z = abs(rnorm(n))

sigma1 = abs(rnorm(K))
sigma0 = abs(rnorm(K))

mu1 = abs(rnorm(K))
mu0 = abs(rnorm(K))
\begin{verbatim}
hk0 = matrix(sample(K, M*H, replace=TRUE), M, H)
hk1 = matrix(sample(K, M*H, replace=TRUE), M, H)

nu.h0 = lapply(1:H, function(x){rbeta(1,5,5)})
nu.h1 = lapply(1:H, function(x){rbeta(1,5,5)})

ph0 = lapply(nu.h0, lambda.u)
ph1 = lapply(nu.h1, lambda.u)

hatpai.u(z, hk1, hk0, ph1, ph0, sigma1, sigma0, mu1, mu0, pai, H, n)
\end{verbatim}

---

\textbf{ind.u}  

\textit{package internal function}

\section*{Description}

package internal function

\section*{Usage}

\texttt{ind.u(pr)}

\section*{Arguments}

\texttt{pr}

\section*{Examples}

\begin{verbatim}
Nmax = 100
K = 5
H = 10
M = 20
n = sample(Nmax, M, replace=TRUE)

z = abs(rnorm(sum(n)))
mu = abs(rnorm(K))
sigma = 1/rgamma(n=K, shape=10, rate=10)

hk = sample(K,M*H, replace=TRUE)

ph = as.vector(sapply(1:M, function(x){lambda.u(rbeta(H, 1, 1))}))

ind.u(z.pr.u(z, hk, mu, sigma, ph, H, n))
\end{verbatim}
### lambda.u

#### Description
package internal function

#### Usage

```r
lambda.u(nu)
```

#### Arguments

- `nu`

#### Examples

```r
H = 5
nu = rbeta(H, 1, 1)
lambda.u(nu)
```

### lg.mu.sig

#### Description
package internal function

#### Usage

```r
lg.mu.sig(m, v)
```

#### Arguments

- `m` Description
- `v` Description

#### Examples

```r
#See package vignette
```
**mu.k.u**

*package internal function*

**Description**

package internal function

**Usage**

```r
mu.k.u(k, ik, z, sigma, mu0)
```

**Arguments**

- `k`
- `ik`
- `z`
- `sigma`
- `mu0`

**Value**

Describe

**Examples**

```r
K = 5
n = 100
z = abs(rnorm(n))
sigma = sapply(1:K, function(x){1/rgamma(n=1, shape=10, rate=10)})
mu0 = 0
ik = sample(K, n, replace=TRUE)
sapply(1:K, mu.k.u, ik, z, sigma, mu0)
```

---

**nu.u**

*package internal function*

**Description**

package internal function

**Usage**

```r
nu.u(ind, tau, H)
```
Arguments

  ind
  tau
  H

Value

  Describe

Examples

  H = 5
  n = 100
  tau = rgamma(1, 1, 1)
  ind = sample(H, n, replace=TRUE)
  nu.u(ind, tau, H)

---

package internal function

Description

  package internal function

Usage

  pai.u(b, a.pai, b.pai)

Arguments

  b
  a.pai
  b.pai

Value

  Describe

Examples

  n = 100
  b = rbinom(n, 1, 0.5)
  a.pai = 10^-6
  b.pai = 10^-6
  pai.u(b, a.pai, b.pai)
$$\text{ptrace}$$  

Trace (ACF) Plots

Description

This function outputs trace plots of certain latent variables.

Usage

\[
\text{ptrace}(\text{res, var, ndisc, nr, nc, type} = \text{"trace"})
\]

Arguments

- \text{res}: An output object from \textit{bhts}. 
- \text{var}: Variable for which to display convergence diagnostic plots. Current options are \text{mu0} (displaying \(\mu_{01}, \ldots, \mu_{0K}\)), \text{mu1} (displaying \(\mu_{11}, \ldots, \mu_{1K}\)), \text{sigma0} (displaying \(\sigma_{01}^2, \ldots, \sigma_{0K}^2\)), \text{sigma1} (displaying \(\sigma_{11}^2, \ldots, \sigma_{1K}^2\)), \text{pk0} (displaying \(\lambda_{01}(0), \ldots, \lambda_{0K}(0)\)) and \text{pk1} (displaying \(\lambda_{11}(1), \ldots, \lambda_{1K}(1)\)).
- \text{ndisc}: Number of iterations for which to discard samples.
- \text{nr}: Number of rows in the resulting composite plot.
- \text{nc}: Number of columns in the resulting composite plot.
- \text{type}: Type of convergence diagnostic. Currently implemented are trace plots (default \text{type} = \text{"trace"}) and ACF plots (\text{type} = \text{"acf"}).

Examples

```
#See package vignette
```

\[
r.fdr
\]

Significant Hits

Description

This function determines significant hits, based on a specified expected FDR.

Usage

\[
r.fdr(\text{res, fdr} = 0.05)
\]

Arguments

- \text{res}: An output object from \textit{bhts}. 
- \text{fdr}: Expected FDR (default is 0.05).
Value

This function returns a list consisting of the following elements:

- `res` A data frame containing significant hits and their probabilities.
- `r` The computed significant hit probability threshold.

Examples

```r
# See package vignette
```

### sig.k.u

**package internal function**

Description

package internal function

Usage

```r
sig.k.u(k, ik, z, mu0, a0, b0)
```

Arguments

- `k`
- `ik`
- `z`
- `mu0`
- `a0`
- `b0`

Examples

```r
K = 5
n = 100
z = abs(rnorm(n))
mu0 = 0
ik = sample(K, n, replace=TRUE)
a0 = 5
b0 = 5
sapply(1:K, sig.k.u, ik, z, mu0, a0, b0)
```
**tau.u**

*package internal function*

**Description**

package internal function

**Usage**

\[\text{tau.u}(\nu, a\theta, b\theta)\]

**Arguments**

- \(\nu\)  
  Description
- \(a\theta\)  
  Description
- \(b\theta\)  
  Description

**Examples**

\[K = 5\]
\[a = 10^{-6}\]
\[b = 10^{-6}\]
\[\nu = \text{rbeta}(K, a, b)\]
\[\text{tau.u} (\nu, a, b)\]

---

**z.pr.u**

*package internal function*

**Description**

package internal function

**Usage**

\[\text{z.pr.u}(z, hk, mu, sigma, ph, H, n)\]

**Arguments**

- \(z\)
- \(hk\)
- \(mu\)
- \(sigma\)
- \(ph\)
- \(H\)
- \(n\)
Examples

\begin{verbatim}
Nmax = 100
K = 5
H = 10
M = 20
n = sample(Nmax, M, replace=TRUE)

z = abs(rnorm(sum(n)))
mu = abs(rnorm(K))
sigma = 1/rgamma(n=K, shape=10, rate=10)

hk = sample(K, M*H, replace=TRUE)

ph = as.vector(sapply(1:M, function(x){lambda.u(rbeta(H, 1, 1))}))

z.pr.u(z, hk, mu, sigma, ph, H, n)
\end{verbatim}
Index

* package
  BHTSpack-package, 2

abfun, 3
alpha.u, 4
b.u, 5
bhts, 6
bhts2HTML, 7
BHTSpack (BHTSpack-package), 2
BHTSpack-package, 2
data.create, 7
fdr.r, 8
h.pr.u, 9
hatpai.u, 10
ind.u, 11
lambda.u, 12
lg.mu.sig, 12
mu.k.u, 13
nu.u, 13
pai.u, 14
ptrace, 15
r.fdr, 15
sig.k.u, 16
tau.u, 17
z.pr.u, 17