

# Package ‘pipeGS’

February 2, 2018

**Version** 0.4

**Date** 2017-12-12

**Title** Permutation p-Value Estimation for Gene Set Tests

**Description** Code for various permutation p-values estimation methods for gene set test. The description of corresponding methods can be found in the dissertation of Yu He(2016) “Efficient permutation P-value estimation for gene set tests” <<https://searchworks.stanford.edu/view/11849351>>. One of the methods also corresponds to the paper “Permutation p-value approximation via generalized Stolarsky invariance” <[arXiv:1603.02757](https://arxiv.org/abs/1603.02757)>.

**URL** <https://searchworks.stanford.edu/view/11849351>,  
<https://arxiv.org/abs/1603.02757>

**License** GPL-2

**Imports** parallel

**Author** Hera He <[njheyu@gmail.com](mailto:njheyu@gmail.com)>

**Maintainer** Hera He <[njheyu@gmail.com](mailto:njheyu@gmail.com)>

**RoxygenNote** 6.0.1

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2018-02-02 18:30:00 UTC

## R topics documented:

get.phat.linear . . . . .	2
get.phat.quadratic . . . . .	3
pipegs . . . . .	4
<b>Index</b>	<b>5</b>

---

get.phat.linear      *Get the pvalue for the linear statistic*

---

### Description

Get the pvalue for the linear statistic

### Usage

```
get.phat.linear(x, y, method = c("phat2", "saddlepoint", "phat1", "phat3",
  "nested", "mc", "exact"), N.mc = 10^3, mc.cores = 1, N.level = 1000,
  q = 0.2, B = 5 * length(x), do.sd = FALSE)
```

### Arguments

x	binary vector of treatment assignment
y	gene expression measurement matrix
method	method to estimate pvalue for the linear statistic
N.mc	number of mc samples is method == "mc"
mc.cores	number of cores to use in mcapply
N.level	number of samples in each level if method == "nested"
q	progression quantile if method == "nested"
B	number of burn-ins to use if method == "nested"
do.sd	logical to indicate whether calculating sd or not

### Value

a list containing phat from different methods

### Examples

```
x <- c(rep(0, 4), rep(1, 4))
y <- c(rnorm(4, 0, 1), rnorm(4, 2, 1))
get.phat.linear(x, y, method = "saddlepoint")
get.phat.linear(x, y, method = "phat1")
get.phat.linear(x, y, method = "phat2")
get.phat.linear(x, y, method = "phat3")
get.phat.linear(x, y, method = "nested")
get.phat.linear(x, y, method = "mc")
get.phat.linear(x, y, method = "exact")
get.phat.linear(x, y, method = "saddlepoint", do.sd = TRUE)
get.phat.linear(x, y, method = "phat1", do.sd = TRUE)
get.phat.linear(x, y, method = "phat2", do.sd = TRUE)
get.phat.linear(x, y, method = "phat3", do.sd = TRUE)
get.phat.linear(x, y, method = "nested", do.sd = TRUE)
get.phat.linear(x, y, method = "mc", do.sd = TRUE)
get.phat.linear(x, y, method = "exact", do.sd = TRUE)
```

---

get.phat.quadratic      *Get the pvalue for the quadratic statistic*

---

### Description

Get the pvalue for the quadratic statistic

### Usage

```
get.phat.quadratic(x, y, method = c("is", "nested", "mc", "exact"),
  N.mc = 10^3, N.is = 10^3, N.level = 1000, q = 0.2, B = 5 *
  length(x), mc.cores = 1, do.sd = FALSE)
```

### Arguments

x	binary vector of treatment assignment
y	gene expression measurement matrix
method	method to estimate pvalue for the linear statistic
N.mc	number of mc samples is method == "mc"
N.is	number of mc samples is method == "is"
N.level	number of samples in each level if method == "nested"
q	progression quantile if method == "nested"
B	number of burn-ins to use if method == "nested"
mc.cores	number of cores to use in mcaply
do.sd	logical to indicate whether calculating sd or not

### Value

a list containing phat from different methods

### Examples

```
x <- c(rep(0, 4), rep(1, 4))
y <- matrix(c(rnorm(4*5, 0, 1), rnorm(4*5, 2, 1)), nrow = 8)
get.phat.quadratic(x, y, method = "is")
get.phat.quadratic(x, y, method = "nested")
get.phat.quadratic(x, y, method = "mc")
get.phat.quadratic(x, y, method = "exact")
get.phat.quadratic(x, y, method = "is", do.sd = TRUE)
get.phat.quadratic(x, y, method = "nested", do.sd = TRUE)
get.phat.quadratic(x, y, method = "mc", do.sd = TRUE)
get.phat.quadratic(x, y, method = "exact", do.sd = TRUE)
```

---

pipegs

*pvalue approximation with stolarsky invariance principal.*

---

**Description**

The package calculates the approximation for permutation p-values of linear statistic

# Index

`get.phat.linear`, [2](#)  
`get.phat.quadratic`, [3](#)  
`pipegs`, [4](#)  
`pipegs-package (pipegs)`, [4](#)