

Package ‘flipr’

March 8, 2021

Title Flexible Inference via Permutations in R

Version 0.2.1

Description A flexible permutation framework for making inference such as point estimation, confidence intervals or hypothesis testing, on any kind of data, be it univariate, multivariate, or more complex such as network-valued data, topological data, functional data or density-valued data.

License GPL (>= 3)

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1

Imports purrr, rlang, magrittr, cli, ggplot2, tibble, withr, viridisLite

Suggests rmarkdown, knitr, covr, tidyverse, plotly, htmlwidgets, htmltools, pracma

VignetteBuilder knitr

URL <https://astamm.github.io/flipr/>, <https://github.com/astamm/flipr/>

BugReports <https://github.com/astamm/flipr/issues/>

Depends R (>= 2.10)

NeedsCompilation no

Author Alessia Pini [aut],
Aymeric Stamm [aut, cre] (<<https://orcid.org/0000-0002-8725-3654>>),
Simone Vantini [aut]

Maintainer Aymeric Stamm <aymeric.stamm@math.cnrs.fr>

Repository CRAN

Date/Publication 2021-03-08 14:42:06 UTC

R topics documented:

flipr	2
test-statistic	2
two_sample_ci	3
two_sample_pe	5
two_sample_pf	6
two_sample_test	8
two_sample_viz	10

Index	12
--------------	-----------

flipr	<i>flipr: Flexible inference via permutations in R</i>
-------	--

Description

The flipr package provides a flexible permutation framework for making inference such as point estimation, confidence intervals or hypothesis testing, on any kind of data, be it univariate, multivariate, or more complex such as network-valued data, topological data, functional data or density-valued data.

test-statistic	<i>Test Statistics for the Two-Sample Problem</i>
----------------	---

Description

This function implements the original Hotelling's T^2 statistic which is defined for multivariate data when the number n of observations is greater than the number p of variables.

Usage

```
stat_hotelling(data, indices, ...)
```

```
stat_t(data, indices, ...)
```

```
stat_f(data, indices, ...)
```

```
stat_mean(data, indices, ...)
```

Arguments

data	A list of the $n_1 + n_2$ concatenated observations with the original n_1 observations from the first sample on top and the original n_2 observations from the second sample below.
indices	An integer vector giving the indices in data that are considered to belong to the first sample.
...	Extra parameters (might be useful for other user-supplied <code>stat_*</code> () functions). Not used here.

Value

A real scalar giving the value of Hotelling's T^2 statistic.

Examples

```
n <- 10L
mx <- 0
sigma <- 1

# Two different models for the two populations
x <- rnorm(n = n, mean = mx, sd = sigma)
x <- as.list(x)
delta <- 10
my <- mx + delta
y <- rnorm(n = n, mean = my, sd = sigma)
y <- as.list(y)
stat_hotelling(c(x, y), 1:n)
stat_t(c(x, y), 1:n)
stat_mean(c(x, y), 1:n)
```

two_sample_ci

Two-Sample Permutation Confidence Interval

Description

This function calculates permutation confidence intervals for two-sample problems. This is done through the specification a point estimate of the parameter to be estimated and of a set of null hypotheses of the form $F_X = F_{\{g(Y, \text{parameters})\}}$ where g is a user-supplied function.

Usage

```
two_sample_ci(
  null_specification,
  x,
  y,
  alpha = 0.05,
  statistic = stat_hotelling,
```

```

B = 1000L,
alternative = "right_tail",
type = "exact",
point_estimate = NULL,
lower_bound = 0,
upper_bound = 1,
seed = NULL
)

```

Arguments

null_specification	A function with two arguments y and parameters s such that $F_X = F_{\{null_specification(Y, parameters)\}}$ under the null hypothesis.
x	A list or matrix representing the 1st sample.
y	A list or matrix representing the 2nd sample.
alpha	A scalar specifying the desired significance level. Default is 0.05 .
statistic	A character vector specifying the chosen test statistic(s). These can be stat_hotelling or user-specified functions that define desired statistics. See the section <i>User-supplied statistic function</i> for more information on how these user-supplied functions should be structured for compatibility with the flipr framework. Default is stat_hotelling .
B	The number of sampled permutation. Default is $1000L$.
alternative	A string specifying whether the p-value is right-tailed, left-tailed or two-tailed. Choices are "right_tail", "left_tail" and "two_tail". Default is "right_tail". Obviously, if the test statistic used in argument <code>statistic</code> is positive, all alternatives will lead to the two-tailed p-value.
type	A string specifying if performing an exact test through the use of Phipson-Smyth estimate of the p-value or an approximate test through a Monte-Carlo estimate of the p-value. Default is "exact".
point_estimate	A scalar providing a point estimate of the parameter under investigation. Default is NULL, in which case the point estimate is computed by maximizing the p-value function via the two_sample_pe function.
lower_bound	A scalar value specifying a rough lower bound of the interval over which searching the maximum of the p-value function. Default is 0 .
upper_bound	A scalar value specifying a rough upper bound of the interval over which searching the maximum of the p-value function. Default is 1 .
seed	An integer specifying the seed of the random generator useful for result reproducibility or method comparisons. Default is NULL.

Value

A length-2 vector providing the confidence interval at the required level.

Examples

```

x1 <- rnorm(10)
x2 <- rnorm(10, 3)
null_spec <- function(y, parameters) {y - parameters[1]}
two_sample_ci(
  null_specification = null_spec,
  x = x1,
  y = x2,
  statistic = stat_t,
  alternative = "two_tail",
  lower_bound = 2,
  upper_bound = 4,
  B = 100L
)

```

two_sample_pe

*Two-Sample Permutation Point Estimate***Description**

This function produces a point estimate of the parameters under investigation using the permutation framework.

Usage

```

two_sample_pe(
  null_specification,
  x,
  y,
  statistic = stat_hotelling,
  B = 1000,
  alternative = "two_tail",
  type = "exact",
  lower_bound = 0,
  upper_bound = 1,
  seed = NULL,
  verbose = FALSE
)

```

Arguments

null_specification	A function with two arguments y and $parameters$ such that $F_X = F_{\{null_specification(Y, parameters)\}}$ under the null hypothesis.
x	A list or matrix representing the 1st sample.
y	A list or matrix representing the 2nd sample.

statistic	A character vector specifying the chosen test statistic(s). These can be <code>stat_hotelling</code> or user-specified functions that define desired statistics. See the section <i>User-supplied statistic function</i> for more information on how these user-supplied functions should be structured for compatibility with the flpr framework. Default is <code>stat_hotelling</code> .
B	The number of sampled permutation. Default is 1000L.
alternative	A string specifying whether the p-value is right-tailed, left-tailed or two-tailed. Choices are "right_tail", "left_tail" and "two_tail". Default is "right_tail". Obviously, if the test statistic used in argument <code>statistic</code> is positive, all alternatives will lead to the two-tailed p-value.
type	A string specifying if performing an exact test through the use of Phipson-Smyth estimate of the p-value or an approximate test through a Monte-Carlo estimate of the p-value. Default is "exact".
lower_bound	A scalar value specifying a rough lower bound of the interval over which searching the maximum of the p-value function. Default is 0.
upper_bound	A scalar value specifying a rough upper bound of the interval over which searching the maximum of the p-value function. Default is 1.
seed	An integer specifying the seed of the random generator useful for result reproducibility or method comparisons. Default is NULL.
verbose	A boolean specifying whether to display some information about the resolution of the maximization problem. Default is FALSE.

Value

A numerical vector providing a point estimate of the parameters.

Examples

```
x1 <- rnorm(10)
x2 <- rnorm(10, mean = 3)
null_spec <- function(y, parameters) {y - parameters[1]}
two_sample_pe(
  null_specification = null_spec,
  x = x1, y = x2,
  statistic = stat_t,
  lower_bound = 2,
  upper_bound = 4
)
```

two_sample_pf

Two-Sample Permutation P-Value Function

Description

This function calculates the permutation p-value function for two-sample problems. This is done through the specification of a set of null hypotheses of the form $F_X = F_{\{g(Y, \text{parameters})\}}$ where g is a user-supplied function.

Usage

```
two_sample_pf(
  parameters,
  null_specification,
  x,
  y,
  statistic = stat_hotelling,
  B = 1000L,
  alternative = "right_tail",
  combine_with = "tippett",
  type = "exact",
  seed = NULL
)
```

Arguments

parameters	A list of vectors specifying a set of candidate parameter values under the null hypothesis.
null_specification	A function with two arguments <code>y</code> and <code>parameters</code> such that $F_X = F_{\{\text{null_specification}(Y, \text{parameters})\}}$ under the null hypothesis.
x	A list or matrix representing the 1st sample.
y	A list or matrix representing the 2nd sample.
statistic	A character vector specifying the chosen test statistic(s). These can be stat_hotelling or user-specified functions that define desired statistics. See the section <i>User-supplied statistic function</i> for more information on how these user-supplied functions should be structured for compatibility with the flipr framework. Default is stat_hotelling .
B	The number of sampled permutation. Default is 1000L.
alternative	A string specifying whether the p-value is right-tailed, left-tailed or two-tailed. Choices are "right_tail", "left_tail" and "two_tail". Default is "right_tail". Obviously, if the test statistic used in argument <code>statistic</code> is positive, all alternatives will lead to the two-tailed p-value.
combine_with	A string specifying the combining function to be used to compute the single test statistic value from the set of p-value estimates obtained during the non-parametric combination testing procedure. Default is "tippett", which picks Tippett's function.
type	A string specifying if performing an exact test through the use of Phipson-Smyth estimate of the p-value or an approximate test through a Monte-Carlo estimate of the p-value. Default is "exact".
seed	An integer specifying the seed of the random generator useful for result reproducibility or method comparisons. Default is NULL.

Value

A vector of p-values of the permutation test for each parameter values specified through the first argument.

Examples

```
x1 <- rnorm(10)
x2 <- rnorm(10, 3)
null_spec <- function(y, parameters) {y - parameters[1]}
two_sample_pf(
  parameters = 3,
  null_specification = null_spec,
  x = x1,
  y = x2,
  statistic = stat_t,
  seed = 1234,
  B = 1000,
  alternative = "two_tail"
)
```

two_sample_test

Two-Sample Permutation Test

Description

This function carries out an hypothesis test where the null hypothesis is that the two samples are ruled by the same underlying generative probability distribution against the alternative hypothesis that they are ruled by two separate generative probability distributions.

Usage

```
two_sample_test(
  x,
  y,
  statistic = stat_hotelling,
  B = 1000L,
  alternative = "right_tail",
  combine_with = "tippett",
  type = "exact",
  seed = NULL
)
```

Arguments

- | | |
|-----------|---|
| x | A list or matrix representing the 1st sample. |
| y | A list or matrix representing the 2nd sample. |
| statistic | A character vector specifying the chosen test statistic(s). These can be stat_hotelling or user-specified functions that define desired statistics. See the section <i>User-supplied statistic function</i> for more information on how these user-supplied functions should be structured for compatibility with the flipr framework. Default is stat_hotelling . |
| B | The number of sampled permutation. Default is 1000L. |

alternative	A string specifying whether the p-value is right-tailed, left-tailed or two-tailed. Choices are "right_tail", "left_tail" and "two_tail". Default is "right_tail". Obviously, if the test statistic used in argument <code>statistic</code> is positive, all alternatives will lead to the two-tailed p-value.
combine_with	A string specifying the combining function to be used to compute the single test statistic value from the set of p-value estimates obtained during the non-parametric combination testing procedure. Default is "tippett", which picks Tippett's function.
type	A string specifying if performing an exact test through the use of Phipson-Smyth estimate of the p-value or an approximate test through a Monte-Carlo estimate of the p-value. Default is "exact".
seed	An integer specifying the seed of the random generator useful for result reproducibility or method comparisons. Default is NULL.

Value

A [list](#) with three components: the value of the statistic for the original two samples, the p-value of the resulting permutation test and a numeric vector storing the values of the permuted statistics.

User-supplied statistic function

A user-specified function should have at least two arguments:

- the first argument is `data` which should be a list of the $n_1 + n_2$ concatenated observations with the original n_1 observations from the first sample on top and the original n_2 observations from the second sample below;
- the second argument is `indices` which should be an integer vector giving the indices in `data` that are considered to belong to the first sample.

See the [stat_hotelling](#) function for an example.

Examples

```
n <- 10L
mx <- 0
sigma <- 1

# Two different models for the two populations
x <- rnorm(n = n, mean = mx, sd = sigma)
delta <- 10
my <- mx + delta
y <- rnorm(n = n, mean = my, sd = sigma)
t1 <- two_sample_test(x, y)
t1$pvalue

# Same model for the two populations
x <- rnorm(n = n, mean = mx, sd = sigma)
delta <- 0
my <- mx + delta
y <- rnorm(n = n, mean = my, sd = sigma)
```

```
t2 <- two_sample_test(x, y)
t2$pvalue
```

two_sample_viz

Two-Sample Permutation P-Value Function Graph

Description

This function plots the p-value function for a single parameter of interest, estimated via permutations.

Usage

```
two_sample_viz(
  null_specification,
  x,
  y,
  alpha = 0.05,
  statistic = stat_hotelling,
  B = 1000L,
  alternative = "two_tail",
  type = "exact",
  point_estimate = NULL,
  confidence_interval = NULL,
  lower_bound = 0,
  upper_bound = 1,
  n_grid_in = 20L,
  n_grid_out = 100L,
  seed = NULL
)
```

Arguments

null_specification	A function with two arguments y and parameters s such that $F_X = F_{\{\text{null_specification}(Y, \text{parameters})\}}$ under the null hypothesis.
x	A list or matrix representing the 1st sample.
y	A list or matrix representing the 2nd sample.
alpha	A scalar specifying the desired significance level. Default is 0.05.
statistic	A character vector specifying the chosen test statistic(s). These can be stat_hotelling or user-specified functions that define desired statistics. See the section <i>User-supplied statistic function</i> for more information on how these user-supplied functions should be structured for compatibility with the flipr framework. Default is stat_hotelling .
B	The number of sampled permutation. Default is 1000L.

alternative	A string specifying whether the p-value is right-tailed, left-tailed or two-tailed. Choices are "right_tail", "left_tail" and "two_tail". Default is "right_tail". Obviously, if the test statistic used in argument <code>statistic</code> is positive, all alternatives will lead to the two-tailed p-value.
type	A string specifying if performing an exact test through the use of Phipson-Smyth estimate of the p-value or an approximate test through a Monte-Carlo estimate of the p-value. Default is "exact".
point_estimate	A scalar providing a point estimate of the parameter under investigation. Default is NULL, in which case the point estimate is computed by maximizing the p-value function via the <code>two_sample_pe</code> function.
confidence_interval	A length-2 numerical vector providing a confidence interval for the parameter under investigation to be used for setting the boundaries of the grid onto which the p-value will be displayed. Default is NULL, in which case the confidence interval is computed via the <code>two_sample_ci</code> function.
lower_bound	A scalar value specifying a rough lower bound of the interval over which searching the maximum of the p-value function. Default is 0.
upper_bound	A scalar value specifying a rough upper bound of the interval over which searching the maximum of the p-value function. Default is 1.
n_grid_in	An integer specifying the size of the grid onto which the p-value function will actually be evaluated. Default is 20L.
n_grid_out	An integer specifying the size of the grid onto which the p-value function will be extrapolated for display. Default is 100L.
seed	An integer specifying the seed of the random generator useful for result reproducibility or method comparisons. Default is NULL.

Value

A `ggplot` object.

Examples

```
x1 <- rnorm(10)
x2 <- rnorm(10, mean = 3)
null_spec <- function(y, parameters) {y - parameters[1]}
## Not run:
two_sample_viz(
  null_specification = null_spec,
  x = x1,
  y = x2,
  statistic = stat_t,
  B = 10000
)
## End(Not run)
```

Index

flipr, [2](#)

ggplot, [11](#)

list, [9](#)

stat_f(test-statistic), [2](#)

stat_hotelling, [4](#), [6-10](#)

stat_hotelling(test-statistic), [2](#)

stat_mean(test-statistic), [2](#)

stat_t(test-statistic), [2](#)

test-statistic, [2](#)

two_sample_ci, [3](#), [11](#)

two_sample_pe, [4](#), [5](#), [11](#)

two_sample_pf, [6](#)

two_sample_test, [8](#)

two_sample_viz, [10](#)