Package ‘funStatTest’

May 23, 2024

Title  Statistical Testing for Functional Data
Version  1.0.3
License  AGPL (>= 3)
Imports  checkmate, distr, dplyr, ggplot2, magrittr, Matrix, pbapply, stats, stringr, tibble, tidyr, tidyselect, utils
Suggests  knitr, rmarkdown, testthat, vdiff
VignetteBuilder  knitr
Config/fusen/version  0.6.0
Date/Publication  2024-05-23 09:00:03 UTC
Encoding  UTF-8
RoxygenNote  7.3.1
NeedsCompilation  no
Author  Zaineb Smida [aut] (<https://orcid.org/0000-0002-9974-299X>), Ghislain Durif [aut, cre] (<https://orcid.org/0000-0003-2567-1401>), Lionel Cucala [aut]
Maintainer  Ghislain Durif <gd.dev@libertymail.net>
Repository  CRAN
comp_stat

Description

Computation of the different statistics defined in the package. See Smida et al (2022) for more details.

Usage

```
comp_stat(MatX, MatY, stat = c("mo", "med"))
```

Arguments

- **MatX**: numeric matrix of dimension \(n_{\text{point}} \times n\) containing \(n\) trajectories (in columns) of size \(n_{\text{point}}\) (in rows).
- **MatY**: numeric matrix of dimension \(n_{\text{point}} \times m\) containing \(m\) trajectories (in columns) of size \(n_{\text{point}}\) (in rows).
- **stat**: character string or vector of character string, name of the statistics for which the p-values will be computed, among "mo", "med", "wmw", "hkr", "cff".

Details

For HKR statistics, only the values of the two statistics, namely \(\text{HKR1}\) and \(\text{HKR2}\) and not the eigen values (see \texttt{stat_hkr()} for more details).

Value

List of named numeric value corresponding to the statistic values listed in \texttt{stat} input.

Author(s)

Zaineb Smida, Ghislain DURIF, Lionel Cucala
permut_pval

References

See Also
stat_mo(), stat_med(), stat_wmw(), stat_hkr(), statcff()

Examples

```r
simu_data <- simul_data(
  n_point = 100, n_obs1 = 50, n_obs2 = 75, c_val = 10,
  delta_shape = "constant", distrib = "normal"
)
MatX <- simu_data$mat_sample1
MatY <- simu_data$mat_sample2

res <- comp_stat(MatX, MatY, stat = c("mo", "med", "wmw", "hkr", "cff"))
res
```

permut_pval

Permutation-based computation of p-values

Description
Computation of the p-values associated to any statistics described in the package with the permutation methods. See Smida et al (2022) for more details.

Usage

```r
permut_pval(MatX, MatY, n_perm = 100, stat = c("mo", "med"), verbose = FALSE)
```

Arguments

- **MatX**: numeric matrix of dimension \( n_{\text{point}} \times n \) containing \( n \) trajectories (in columns) of size \( n_{\text{point}} \) (in rows).
- **MatY**: numeric matrix of dimension \( n_{\text{point}} \times m \) containing \( m \) trajectories (in columns) of size \( n_{\text{point}} \) (in rows).
- **n_perm**: integer, number of permutation to compute the p-values.
- **stat**: character string or vector of character string, name of the statistics for which the p-values will be computed, among "mo", "med", "wmw", "hkr", "cff".
- **verbose**: boolean, if TRUE, enable verbosity.

Value
list of named numeric value corresponding to the p-values for each statistic listed in the stat input.
Author(s)
Zaineb Smida, Ghislain DURIF, Lionel Cucala

References

See Also
stat_mo(), stat_med(), stat_wmw(), stat_hkr(), stat_cff(), comp_stat()

Examples

# simulate small data for the example
simu_data <- simul_data(
  n_point = 20, n_obs1 = 4, n_obs2 = 5, c_val = 10,
  delta_shape = "constant", distrib = "normal"
)

MatX <- simu_data$mat_sample1
MatY <- simu_data$mat_sample2
res <- permut_pval(
  MatX, MatY, n_perm = 100, stat = c("mo", "med", "wmw", "hkr", "cff"),
  verbose = TRUE)
res

plot_simu(simu)

Description
Graphical representation of simulated data

Usage
plot_simu(simu)

Arguments
simu           list, output of simul_data()

Value
the ggplot2 graph of simulated trajectories.
power_exp

Author(s)
Zaineb Smida, Ghislain DURIF, Lionel Cucala

See Also
simul_data()

Examples

# constant delta
simu_data <- simul_data(
  n_point = 100, n_obs1 = 50, n_obs2 = 75, c_val = 5,
  delta_shape = "constant", distrib = "normal"
)
plot_simu(simu_data)
# linear delta
simu_data <- simul_data(
  n_point = 100, n_obs1 = 50, n_obs2 = 75, c_val = 5,
  delta_shape = "linear", distrib = "normal"
)
plot_simu(simu_data)
# quadratic delta
simu_data <- simul_data(
  n_point = 100, n_obs1 = 50, n_obs2 = 75, c_val = 5,
  delta_shape = "quadratic", distrib = "normal"
)
plot_simu(simu_data)

Description

Computation of the statistical power (i.e. risk to reject the null hypothesis when it is false) associated to any statistics described in the package based on simulation permutation-based p-values computations. See Smida et al (2022) for more details.

Usage

power_exp(
  n_simu = 100,
  alpha = 0.05,
  n_perm = 100,
  stat = c("mo", "med"),
  n_point = 100,
  n_obs1 = 50,
  n_obs2 = 50,
  c_val = 1,
delta_shape = "constant",
distrib = "normal",
max_iter = 10000,
verbose = FALSE
)

Arguments

n_simu integer value, number of simulations to compute the statistical power.
alpha numerical value, between 0 and 1, type I risk level to reject the null hypothesis in the simulation. Default value is 5%.
n_perm integer, number of permutation to compute the p-values.
stat character string or vector of character string, name of the statistics for which the p-values will be computed, among "mo", "med", "wmw", "hkr", "cff".
n_point integer value, number of points in the trajectory.
n_obs1 integer value, number of trajectories in the first sample.
n_obs2 integer value, number of trajectories in the second sample.
c_val numeric value, level of divergence between the two samples.
delta_shape character string, shape of the divergence between the two samples, among "constant", "linear", "quadratic".
distrib character string, type of probability distribution used to simulate the data among "normal", "cauchy", "dexp", "student".
max_iter integer, maximum number of iteration for the iterative simulation process.
verbose boolean, if TRUE, enable verbosity.

Details

The c_val input argument should be strictly positive so that the null hypothesis is not verified when simulating the data (i.e. so that the two sample are not generated from the same probability distribution).

Value

a list with the following elements:

- power_res: a list of named numeric value corresponding to the power values for each statistic listed in stat input.
- pval_res: a list of named numeric values corresponding to the p-values for each simulation and each statistic listed in the stat input.
- simu_config: information about input parameters used for simulation, including n_simu, c_val, distrib, delta_shape, n_point, n_obs1, n_obs2.

Author(s)

Zaineb Smida, Ghislain DURIF, Lionel Cucala
References


See Also

stat_mo(), stat_med(), stat_wmw(), stat_hkr(), stat_cff(), comp_stat()

Examples

# simulate a few small data for the example
res <- power_exp(
  n_simu = 20, alpha = 0.05, n_perm = 100,
  stat = c("mo", "med", "wmw", "hkr", "cff"),
  n_point = 25, n_obs1 = 4, n_obs2 = 5, c_val = 10, delta_shape = "constant",
  distrib = "normal", max_iter = 10000, verbose = FALSE
)
res$power_res
delta_shape character string, shape of the divergence between the two samples, among "constant", "linear", "quadratic".

distrib character string, type of probability distribution used to simulate the data among "normal", "cauchy", "dexp", "student".

max_iter integer, maximum number of iteration for the iterative simulation process.

Value

A list with the following elements

- mat_sample1: numeric matrix of dimension n_point x n_obs1 containing n_obs1 trajectories (in columns) of size n_point (in rows) corresponding to sample 1.
- mat_sample2: numeric matrix of dimension n_point x n_obs2 containing n_obs2 trajectories (in columns) of size n_point (in rows) corresponding to sample 2.

Author(s)

Zaineb Smida, Ghislain DURIF, Lionel Cucala

References


See Also

plot_simu(), simul_traj()

Examples

```r
simu_data <- simul_data(
  n_point = 100, n_obs1 = 50, n_obs2 = 75, c_val = 10, 
  delta_shape = "constant", distrib = "normal"
)
str(simu_data)
```

---

### simul_traj

*Single trajectory simulation*

Description

Simulate a trajectory of length n_point using a random generator associated to different probability distribution.

Usage

```r
simul_traj(n_point, distrib = "normal", max_iter = 10000)
```
Arguments

n_point  integer value, number of points in the trajectory.
distrib  character string, type of probability distribution used to simulate the data among "normal", "cauchy", "dexp", "student".
max_iter integer, maximum number of iteration for the iterative simulation process.

Value

Vector of size n_point with the trajectory values.

Author(s)

Zaineb Smida, Ghislain DURIF, Lionel Cucala

References


See Also

simul_data()

Examples

```r
simu_vec <- simul_traj(100)
plot(simu_vec, xlab = "point", ylab = "value")
```

stat_cff

Cuevas-Febrero-Fraiman statistic

Description


Usage

```
stat_cff(MatX, MatY)
```

Arguments

MatX  numeric matrix of dimension n_point x n containing n trajectories (in columns) of size n_point (in rows).
MatY  numeric matrix of dimension n_point x m containing m trajectories (in columns) of size n_point (in rows).
**Value**

numeric value corresponding to the WMW statistic value

**Author(s)**

Zaineb Smida, Ghislain Durif, Lionel Cucala

**References**


**See Also**

comp_stat(), permut_pval()

**Examples**

```r
simu_data <- simul_data(
    n_point = 100, n_obs1 = 50, n_obs2 = 75, c_val = 10,
    delta_shape = "constant", distrib = "normal"
)

MatX <- simu_data$mat_sample1
MatY <- simu_data$mat_sample2

stat_cff(MatX, MatY)
```

**Description**

The Horváth-Kokoszka-Reeder statistics defined in Chakraborty & Chaudhuri (2015) (and noted HKR1 and HKR2 in Smida et al 2022) are computed to compare two sets of functional trajectories.

**Usage**

```r
stat_hkr(MatX, MatY)
```

**Arguments**

- **MatX** numeric matrix of dimension \( n_{\text{point}} \times n \) containing \( n \) trajectories (in columns) of size \( n_{\text{point}} \) (in rows).
- **MatY** numeric matrix of dimension \( n_{\text{point}} \times m \) containing \( m \) trajectories (in columns) of size \( n_{\text{point}} \) (in rows).
**Value**

A list with the following elements

- **T1**: numeric value corresponding to the HKR1 statistic value
- **T2**: numeric value corresponding to the HKR2 statistic value
- **eigenval**: numeric vector of eigen values from the empirical pooled covariance matrix of MatX and MatY (see Smida et al, 2022, for more details)

**Author(s)**

Zaineb Smida, Ghislain DURIF, Lionel Cucala

**References**


**See Also**

comp_stat(), permut_pval()

**Examples**

```r
simu_data <- simul_data(
  n_point = 100, n_obs1 = 50, n_obs2 = 75, c_val = 10,
  delta_shape = "constant", distrib = "normal"
)

MatX <- simu_data$mat_sample1
MatY <- simu_data$mat_sample2

stat_hkr(MatX, MatY)
```

---

**Description**

The MED median statistics defined in Smida et al (2022) is computed to compare two sets of functional trajectories.

**Usage**

```r
stat_med(MatX, MatY)
```
Arguments

MatX: numeric matrix of dimension n_point x n containing n trajectories (in columns) of size n_point (in rows).
MatY: numeric matrix of dimension n_point x m containing m trajectories (in columns) of size n_point (in rows).

Value

numeric value corresponding to the MED median statistic value

Author(s)

Zaineb Smida, Ghislain Durif, Lionel Cucala

References


See Also

comp_stat(), permut_pval()

Examples

```r
simu_data <- simul_data(
  n_point = 100, n_obs1 = 50, n_obs2 = 75, c_val = 10,
  delta_shape = "constant", distrib = "normal"
)
MatX <- simu_data$mat_sample1
MatY <- simu_data$mat_sample2
stat_med(MatX, MatY)
```

Description

The MO median statistics defined in Smida et al (2022) is computed to compare two sets of functional trajectories.

Usage

```r
stat_mo(MatX, MatY)
```
stat_wmw

Arguments

MatX numeric matrix of dimension n_point \times n containing n trajectories (in columns) of size n_point (in rows).

MatY numeric matrix of dimension n_point \times m containing m trajectories (in columns) of size n_point (in rows).

Value

umeric value corresponding to the MO median statistic value

Author(s)

Zaineb Smida, Ghislain DURIF, Lionel Cucala

References


See Also

comp_stat(), permut_pval()

Examples

```r
simu_data <- simul_data(
  n_point = 100, n_obs1 = 50, n_obs2 = 75, c_val = 10,
  delta_shape = "constant", distrib = "normal"
)
MatX <- simu_data$mat_sample1
MatY <- simu_data$mat_sample2
stat_mo(MatX, MatY)
```

---

**stat_wmw** **Wilcoxon-Mann-Whitney (WMW) statistic**

Description


Usage

```r
stat_wmw(MatX, MatY)
```
Arguments

MatX numeric matrix of dimension \( n_{\text{point}} \times n \) containing \( n \) trajectories (in columns) of size \( n_{\text{point}} \) (in rows).

MatY numeric matrix of dimension \( n_{\text{point}} \times m \) containing \( m \) trajectories (in columns) of size \( n_{\text{point}} \) (in rows).

Value

numeric value corresponding to the WMW statistic value

Author(s)

Zaineb Smida, Ghislain DURIF, Lionel Cucala

References


See Also

comp_stat(), permut_pval()

Examples

```r
simu_data <- simul_data(
  n_point = 100, n_obs1 = 50, n_obs2 = 75, c_val = 10,
  delta_shape = "constant", distrib = "normal"
)

MatX <- simu_data$mat_sample1
MatY <- simu_data$mat_sample2

stat_wmw(MatX, MatY)
```
Index

comp_stat, 2
comp_stat(). 4, 7, 10–14

permut_pval, 3
permut_pval(). 10–14
plot_simu, 4
plot_simu(). 8
power_exp, 5

simul_data, 7
simul_data(). 4, 5, 9
simul_traj, 8
simul_traj(). 8
stat_cff, 9
stat_cff(). 3, 4, 7
stat_hkr, 10
stat_hkr(). 2–4, 7
stat_med, 11
stat_med(). 3, 4, 7
stat_mo, 12
stat_mo(). 3, 4, 7
stat_wmm, 13
stat_wmm(). 3, 4, 7