Package 'jaccard'

May 8, 2026

Type Package

Title Testing Similarity Between Binary Datasets using Jaccard/Tanimoto Coefficients

Version 0.1.2

Date 2026-02-02

Maintainer Neo Christopher Chung <nchchung@gmail.com>

Description Calculate statistical significance of Jaccard/Tanimoto similarity coefficients.

License GPL-2

Encoding UTF-8

Imports Rcpp (>= 0.12.6), qvalue, shiny

LinkingTo Rcpp

NeedsCompilation yes

RoxygenNote 7.3.3

Author Neo Christopher Chung [aut, cre],
Błażej Miasojedow [aut],
Michał Startek [aut],
Anna Gambin [aut]

Repository CRAN

Date/Publication 2026-02-03 10:00:11 UTC

Contents

`jaccard` *Compute a Jaccard/Tanimoto similarity coefficient*

Description

Compute a Jaccard/Tanimoto similarity coefficient

Usage

```
jaccard(x, y, center = FALSE, px = NULL, py = NULL)
```

Arguments

`x`	a binary vector (e.g., fingerprint)
`y`	a binary vector (e.g., fingerprint)
`center`	whether to center the Jaccard/Tanimoto coefficient by its expectation
`px`	probability of successes in `x` (optional)
`py`	probability of successes in `y` (optional)

Value

`jaccard.test.bootstrap` returns an expected value.

Examples

```
set.seed(1234)
x = rbinom(100,1,.5)
y = rbinom(100,1,.5)
jaccard(x,y)
```

`jaccard.ev` *Compute an expected Jaccard/Tanimoto similarity coefficient under independence*

Description

Compute an expected Jaccard/Tanimoto similarity coefficient under independence

Usage

```
jaccard.ev(x, y, px = NULL, py = NULL)
```

Arguments

x	a binary vector (e.g., fingerprint)
y	a binary vector (e.g., fingerprint)
px	probability of successes in x (optional)
py	probability of successes in y (optional)

Value

`jaccard.test.bootstrap` returns an expected value.

Examples

```
set.seed(1234)
x = rbinom(100,1,.5)
y = rbinom(100,1,.5)
jaccard.ev(x,y)
```

`jaccard.rahman` *Compute p-value using the EC-BLAST method*

Description

In the EC-BLAST paper, Rahman et al. (2014) provide the following description: The mean (μ) and s.d. (σ) of the similarity scores are used to define the z score, $z = (Tw - \mu)/\sigma$. For the purpose of calculating the P value, only hits with T > 0 are considered. The P value w is derived from the z score using an extreme value distribution $P = 1 - \exp\left(-e^{-z\pi/\sqrt{6}-\Gamma'(1)}\right)$, where the Euler-Mascheroni constant $\Gamma'(1) \approx 0.577215665$.

Usage

```
jaccard.rahman(j)
```

Arguments

j	a numeric vector of observed Jaccard coefficients (uncentered)

Value

`jaccard.rahman` returns a numeric vector of p-values

References

Rahman, Cuesta, Furnham, Holliday, and Thornton (2014) EC-BLAST: a tool to automatically search and compare enzyme reactions. Nature Methods, 11(2) https://www.nature.com/articles/nmeth.2803

jaccard.test *Test for Jaccard/Tanimoto similarity coefficients*

Description

Compute statistical significance of Jaccard/Tanimoto similarity coefficients between binary vectors, using four different methods.

Usage

```
jaccard.test(x, y, method = "mca", px = NULL, py = NULL, verbose = TRUE, ...)
```

Arguments

x	a binary vector (e.g., fingerprint)
y	a binary vector (e.g., fingerprint)
method	a method to compute a p-value ("mca", "bootstrap", "asymptotic", or "exact")
px	probability of successes in x (optional)
py	probability of successes in y (optional)
verbose	whether to print progress messages
...	optional arguments for specific computational methods

Details

There exist four methods to compute p-values of Jaccard/Tanimoto similarity coefficients: mca, bootstrap, asymptotic, and exact. This is simply a wrapper function for corresponding four functions in this package: jaccard.test.mca, jaccard.test.bootstrap, jaccard.test.asymptotic, and jaccard.test.exact.

We recommand using either mca or bootstrap methods, since the exact solution is slow for a moderately large vector and asymptotic approximation may be inaccurate depending on the input vector size. The bootstrap method uses resampling with replacement binary vectors to compute a p-value (see optional arguments). The mca method uses the measure concentration algorithm that estimates the multinomial distribution with a known error bound (specified by an optional argument accuracy).

Value

jaccard.test returns a list mainly consisting of

statistics	centered Jaccard/Tanimoto similarity coefficient
pvalue	p-value
expectation	expectation

Optional arguments for `method="bootstrap"`

fix whether to fix (i.e., not resample) `x` and/or `y`

B a total bootstrap iteration

seed a seed for a random number generator

Optional arguments for `method="mca"`

accuracy an error bound on approximating a multinomial distribution

error.type an error type on approximating a multinomial distribution (`"average"`, `"upper"`, `"lower"`)

seed a seed for the random number generator.

See Also

jaccard.test.bootstrap jaccard.test.mca jaccard.test.exact jaccard.test.asymptotic

Examples

```
set.seed(1234)
x = rbinom(100,1,.5)
y = rbinom(100,1,.5)
jaccard.test(x,y,method="bootstrap")
jaccard.test(x,y,method="mca")
jaccard.test(x,y,method="exact")
jaccard.test(x,y,method="asymptotic")
```

`jaccard.test.asymptotic`

Compute p-value using an asymptotic approximation

Description

Compute statistical significance of Jaccard/Tanimoto similarity coefficients.

Usage

```
jaccard.test.asymptotic(x, y, px = NULL, py = NULL, verbose = TRUE)
```

Arguments

`x`	a binary vector (e.g., fingerprint)
`y`	a binary vector (e.g., fingerprint)
`px`	probability of successes in `x` (optional)
`py`	probability of successes in `y` (optional)
`verbose`	whether to print progress messages

Value

`jaccard.test.asymptotic` returns a list consisting of

`statistics`	centered Jaccard/Tanimoto similarity coefficient
`pvalue`	p-value
`expectation`	expectation

Examples

```
set.seed(1234)
x = rbinom(100,1,.5)
y = rbinom(100,1,.5)
jaccard.test.asymptotic(x,y)
```

`jaccard.test.bootstrap` *Compute p-value using the bootstrap procedure*

Description

Compute statistical significance of Jaccard/Tanimoto similarity coefficients.

Usage

```
jaccard.test.bootstrap(
  x,
  y,
  px = NULL,
  py = NULL,
  verbose = TRUE,
  fix = "x",
  B = 1000,
  seed = NULL
)
```

Arguments

`x`	a binary vector (e.g., fingerprint)
`y`	a binary vector (e.g., fingerprint)
`px`	probability of successes in `x` (optional)
`py`	probability of successes in `y` (optional)
`verbose`	whether to print progress messages
`fix`	whether to fix (i.e., not resample) `x` and/or `y`
`B`	a total bootstrap iteration
`seed`	a seed for a random number generator

Value

`jaccard.test.bootstrap` returns a list consisting of

`statistics`	centered Jaccard/Tanimoto similarity coefficient
`pvalue`	p-value
`expectation`	expectation

Examples

```
set.seed(1234)
x = rbinom(100,1,.5)
y = rbinom(100,1,.5)
jaccard.test.bootstrap(x,y,B=500)
```

`jaccard.test.exact` *Compute p-value using the exact solution*

Description

Compute statistical significance of Jaccard/Tanimoto similarity coefficients.

Usage

```
jaccard.test.exact(x, y, px = NULL, py = NULL, verbose = TRUE)
```

Arguments

`x`	a binary vector (e.g., fingerprint)
`y`	a binary vector (e.g., fingerprint)
`px`	probability of successes in `x` (optional)
`py`	probability of successes in `y` (optional)
`verbose`	whether to print progress messages

Value

`jaccard.test.exact` returns a list consisting of

`statistics`	centered Jaccard/Tanimoto similarity coefficient
`pvalue`	p-value
`expectation`	expectation

Examples

```
set.seed(1234)
x = rbinom(100,1,.5)
y = rbinom(100,1,.5)
jaccard.test.exact(x,y)
```

`jaccard.test.mca` *Compute p-value using the Measure Concentration Algorithm*

Description

Compute statistical significance of Jaccard/Tanimoto similarity coefficients.

Usage

```
jaccard.test.mca(
  x,
  y,
  px = NULL,
  py = NULL,
  accuracy = 1e-05,
  error.type = "average",
  verbose = TRUE
)
```

Arguments

`x`	a binary vector (e.g., fingerprint)
`y`	a binary vector (e.g., fingerprint)
`px`	probability of successes in `x` (optional)
`py`	probability of successes in `y` (optional)
`accuracy`	an error bound on approximating a multinomial distribution
`error.type`	an error type on approximating a multinomial distribution ("average", "upper", "lower")
`verbose`	whether to print progress messages

Value

`jaccard.test.mca` returns a list consisting of

`statistics`	centered Jaccard/Tanimoto similarity coefficient
`pvalue`	p-value
`expectation`	expectation

Examples

```
set.seed(1234)
x = rbinom(100,1,.5)
y = rbinom(100,1,.5)
jaccard.test.mca(x,y,accuracy = 1e-05)
```

`jaccard.test.pairwise` *Pair-wise tests for Jaccard/Tanimoto similarity coefficients*

Description

Given a data matrix, it computes pair-wise Jaccard/Tanimoto similarity coefficients and p-values among rows (variables). Only for testing due to its use of a for-loop.

Usage

```
jaccard.test.pairwise(
  dat,
  method = "mca",
  verbose = TRUE,
  compute.qvalue = TRUE,
  ...
)
```

Arguments

`dat`	a data matrix
`method`	a method to compute a p-value (`"mca"`, `"bootstrap"`, `"asymptotic"`, or `"exact"`)
`verbose`	whether to print progress messages
`compute.qvalue`	whether to compute q-values
`...`	optional arguments for specific computational methods

Value

`jaccard.test.pairwise` returns a list of matrices

`statistics`	Jaccard/Tanimoto similarity coefficients
`pvalues`	p-values
`qvalues`	q-values

See Also

jaccard.test

`runJaccardApp` *Launch an interactive Shiny app on a local network*

Description

Launch an interactive Shiny app on a local network

Usage

```
runJaccardApp()
```

Index