Package 'bayesbio'

May 24, 2016

Title Miscellaneous Functions for Bioinformatics and Bayesian Statistics

Version 1.0.0

Description A hodgepodge of hopefully helpful functions. Two of these perform shrinkage estimation: one using a simple weighted method where the user can specify the degree of shrinkage required, and one using James-Stein shrinkage estimation for the case of unequal variances.

Depends R (>= 3.2.0)

Suggests ggplot2, RISmed, testthat

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 5.0.1

NeedsCompilation no

Author Andrew McKenzie [aut, cre]

Maintainer Andrew McKenzie <amckenz@gmail.com>

Repository CRAN

Date/Publication 2016-05-24 16:32:42

R topics documented:

`allDups` *Identify all duplicates values in a vector.*

Description

By default the base R function duplicated only identifies the duplicated values after the first in a vector as TRUE. This function identifies all of the duplicates as true.

Usage

```
allDups(x)
```

Arguments

`x`	The input vector.

Value

A logical vector.

`a_hat_mle` *Likelihood function of the James-Stein shrinkage factor.*

Description

To be used in MLE computation of the James-Stein shrinkage factor.

Usage

```
a_hat_mle(stat, vars, a_hat)
```

Arguments

`stat`	Input statistics to be shrinkage estimated.
`vars`	Corresponding variances of equal length.
`a_hat`	Shrinkage intensity to be estimated.

Value

The likelihood of the function given the parameters.

References

http://projecteuclid.org/euclid.ss/1331729986

`bayesbio` *bayesbio: Miscellaneous functions useful in bioinformatics and Bayesian statistics*

Description

A hodgepodge of hopefully helpful functions. Two of these perform shrinkage estimation: one using a simple weighted method where the user can specify the degree of shrinkage required, and one using James-Stein shrinkage estimation for the case of unequal variances.

`cbindFill` *cbind while converting missing entries to NA.*

Description

cbind usually malfunctions on vector of unequal lengths; this function allows vectors of unequal length to be combined, while filling the missing entries with NAs.

Usage

```
cbindFill(...)
```

Arguments

...	A set of vectors separated by commas.

Value

A matrix that combines the inputted vectors.

References

http://r.789695.n4.nabble.com/How-to-join-matrices-of-different-row-length-from-a-list-td3177212.html; http://stackoverflow.com/a/7962286/560791

createStrings *Creates random, unique character strings.*

Description

Makes them unique by randomly choosing the character strings; and, in case it is necessary, adding numbers to the end using make.unique.

Usage

```
createStrings(number, length, upper = FALSE)
```

Arguments

number	Specifies the number of character strings that should be created.
length	Specifies the length of each character string in letters.
upper	Binary parameter specifying whether the character strings should be uppercase. Default = FALSE, so the character strings are all lowercase.

References

http://stackoverflow.com/a/1439541/560791

ggHorizBar *Create a color-labeled horizontal bar plot in ggplot2.*

Description

This function takes a data frame and creates a horizontal (by default) bar plot from it while ordering the values.

Usage

```
ggHorizBar(data_df, dataCol, namesCol, labelsCol, decreasing = TRUE)
```

Arguments

data_df	Data frame with columns to specify the data values, the row names, and the fill colors of each of the bars.
dataCol	The column name that specifies the values to be plotted.
namesCol	The column name that specifies the corresponding names for each of the bar plots to be plotted.
labelsCol	The column name that specifies the groups of the labels.
decreasing	Logical specifying whether the values in dataCol should be in decreasing order.

Value

A ggplot2 object, which can be plotted via the plot() function or saved via the ggsave() function.

`jaccardSets` *Jaccard index of two character vectors.*

Description

This function compares the elements in two character vectors to find the Jaccard index, i.e. the number of intersections divided by the total number of elements in both sets.

Usage

```
jaccardSets(set1, set2)
```

Arguments

`set1`	Character vector.
`set2`	Character vector.

Value

A number (one-element numeric vector) specifying the Jaccard index from comparing the two sets.

References

https://en.wikipedia.org/wiki/Jaccard_index

`mgsub` *Multiple pattern gsub.*

Description

An extension to gsub that handles vectors of patterns and replacements, avoiding recursion problems associated with overlap at the extense of computation time.

Usage

```
mgsub(pattern, replacement, x, ...)
```

Arguments

`pattern`	Character vector of patterns to match.
`replacement`	Character vector of replacements for each pattern.
`x`	Character vector in which the gsub should be performed.
`...`	Additional arguments to grep.

References

http://stackoverflow.com/a/15254254/560791

`nearestTime` *Merge data frames based on the nearest datetime differences.*

Description

Takes two data frames each with time/date columns in date-time or date format (i.e., able to be compared using the function difftime), finds the rows of df2 that minimize the absolute value of the datetime for each of the rows in df1, and merges the corresponding rows of df2 into df1 for downstream processing.

Usage

```
nearestTime(df1, df2, timeCol1, timeCol2)
```

Arguments

`df1`	Data frame containing the dates for which the differences between the other data frame's date column should be minimized for each row.
`df2`	Data frame containing the dates which should be compared to, as well as other values that should be merged to df1 per minimized date time.
`timeCol1`	Character vector specifying the date/time column in df1.
`timeCol2`	Character vector specifying the date/time column in df2.

Value

A merged data frame that minimizes datetime differences.

`nearestTimeandID` *Merge data frames based on the nearest datetime differences and an ID column. Also removes duplicate column names from the result.*

Description

Takes two data frames each with time/date columns in date-time or date format (i.e., able to be compared using the function difftime), finds the rows of df2 that minimize the absolute value of the datetime for each of the rows in df1, and merges the corresponding rows of df2 into df1 for downstream processing.

Usage

```
nearestTimeandID(df1, df2, timeCol1, timeCol2, IDcol)
```

Arguments

df1	Data frame containing the dates for which the differences between the other data frame's date column should be minimized for each row.
df2	Data frame containing the dates which should be compared to, as well as other values that should be merged to df1 per minimized date time.
timeCol1	Character vector specifying the date/time column in df1.
timeCol2	Character vector specifying the date/time column in df2.
IDcol	Must be unique by row in df1. Multiple versions are allowed (and expected at least in some rows, as that is the point of the function) in df2.

Value

A merged data frame that minimizes datetime differences.

---

p.adjust.nlp *Adjust p-values where n is less than p.*

Description

This function recapitulates p.adjust but allows the number of hypothesis tests n to be less than the number of p-values p. Statistical properties of the p-value adjustments may not hold.

Usage

```
p.adjust.nlp(p, method = p.adjust.methods, n = length(p))
```

Arguments

p	Numeric vector of p-values.
method	Correction method.
n	Number of comparisons to be made.

References

http://stackoverflow.com/a/30110186/560791

pubmedQuery *Perform PubMed queries on 2x2 combinations of term vectors.*

Description

Perform PubMed queries on the intersections of two character vectors. This function is a wrapper to RISmed::EUtilsSummary with type = 'esearch', db = 'pubmed'.

Usage

```
pubmedQuery(rowTerms, colTerms, sleepTime = 0.01)
```

Arguments

rowTerms	Character vector of terms that should make up the rows of the resulting mention count data frame.
colTerms	Character vector of terms for the columns.
sleepTime	How much time (in seconds) to sleep between successive PubMed queries. If you set this too low, PubMed may shut down your connection to prevent overloading their servers.

Value

A data frame of the number of mentions for each combination of terms.

subsupDiag *Add values to the super- and sub-diagonals of a matrix.*

Description

Takes a matrix and adds values to the values that are one above the diagonal (ie the superdiagonal) and the values that are one below the diagonal (ie the subdiagonal).

Usage

```
subsupDiag(matrix, x)
```

Arguments

matrix	Matrix whose super- and sub-diagonals values should be replaced.
x	Numeric vector used to replace values in the matrix. If the inputted vector is not of the same length as both the super- and sub-diagonals of the matrix, then short vector recycling will occur (e.g., x can be one value to replace all of the super- and sub-diagonals of the matrix with that one value).

Value

The original matrix with the values added.

References

http://stackoverflow.com/a/9885186/560791

`unequalVarShrink` *Perform James-Stein shrinkage estimation using unequal variances*

Description

Traditional JS shrinkage estimation assumes equal variances for each of the data points, while this algorithm extends JS shrinkage estimation to entries with different variances.

Usage

```
unequalVarShrink(stat, vars, verbose = TRUE)
```

Arguments

`stat`	Input statistics to be shrinkage estimated.
`vars`	Corresponding variances of equal length.
`verbose`	Whether information about the algorithm should be reported.

Value

A data frame containing the shrinkage estimated statistics.

References

http://projecteuclid.org/euclid.ss/1331729986

`weightedShrink` *Weighted shrinkage estimation.*

Description

Shrink values towards the mean (in the sample or the overall cohort) to an inverse degree to the confidence you assign to that observation.

Usage

```
weightedShrink(x, n, m = NULL, meanVal = NULL)
```

Arguments

x	Numeric vector of values to be shrunken towards the mean.
n	Numeric vector with corresponding entries to x, specifying the number of observations used to calculate x, or some other confidence weight to associate with x.
m	Number specifying weight of the shrinkage estimation, relative to the number of observations in the input vector n. Defaults to the minimum of n, but this is an arbitrary value and should be explored to find an optimal value for your use case.
meanVal	Number specifying the overall mean towards which the values should be shrunken. Defaults to NULL, in which case it is calculated as the (non-weighted) arithmetic mean of the values in the inputted vector x.

Value

A numeric vector with shrunken data values.

References

http://math.stackexchange.com/a/41513

Index