Package ‘MIPHENO’

February 19, 2015

Version 1.2

Date 2011-11-01

Title Mutant Identification through Probabilistic High throughput
Enabled NOrmalization

Author Shannon M. Bell <bell.shannonm@gmail.com>, Lyle D. Burgoon
<burgoon.lyle@epa.gov>

Maintainer Shannon M. Bell <bell.shannonm@gmail.com>

Depends R (>= 2.12.1)

Imports doBy, gdata

LazyLoad yes

Description This package contains functions to carry out processing of high throughput data analysis and detection of putative hits/mutants. Contents include a function for post-hoc quality control for removal of outlier sample sets, a median-based normalization method for use in datasets where there are no explicit controls and where most of the responses are of the wildtype/no response class (see accompanying paper). The package also includes a way to prioritize individuals of interest using am empirical cumulative distribution function. Methods for generating synthetic data as well as data from the Chloroplast 2010 project are included.

Repository CRAN

Date/Publication 2012-01-27 11:27:41

NeedsCompilation no

R topics documented:

`cdf.pval` *Generate Empirical pvalues from Cumulative Distribution Function*

Description

Returns the pvalue for the probability of observing a response equal to the input data (/codecdf.data or /codesample.data) or more extreme (smaller) based on an empirical distribution function (ecdf) of the `cdf.data`. Observations with a high pvalue (-> 1) are also rare, thus calculating 1-pvalue or 1-F will return the probability at the other end of the distribution. See ecdf for details `ecdf`.

Usage

```
cdf.pval(cdf.data, sample.data=NULL, ...)
```

Arguments

`cdf.data` Dataframe or Matrix of inputs used to make the CDF (ie the NULL distribution or wt distribution, if known), or the sample data if a NULL distribution is unavailable.

`sample.data` Optional. Dataframe or Matrix of inputs for which a pvalue is to be determined in the event that a seperate NULL distribution is used.

`...` Other parameters.

Details

Data should be presorted if you are going to match it to labels (ie sample descriptors) as the labels need to be removed prior to processing. Only numeric and NA data are permitted. Columns in `cdf.data` and `sample.data` should be corresponding if using both.

Each column in the `cdf.data` is used for generating the CDF. For columns (assays or probes) where <2 observations were made, the column is omitted from the CDF calculation. This step generates a CDF function for each column. Data from /codecdf.data or /codesample.data, if supplied, is run through the corresponding CDF function by column (assay), where the probability of observing the response (row value) is calculated.

Value

`cdf.pval` returns a dataframe containing rows= observations (input order preserved) and columns = assays. The column order may have changed to match `sample.data`, if provided. Values will be between 0 and 1.

Author(s)

Shannon M. Bell

References

Shannon M. Bell, Lyle D. Burgoon, Robert L. Last. MIPHENO: Data normalization for high throughput metabolite analysis. *BMC Bioinformatics* 2012, 13(10)

See Also

ecdf

Examples

```
#See the sweave document in the corresponding paper for examples
```

find_hits *Identification of putative hits using Zvalues or MIPHENO empirical pval*

Description

Returns a dataframe containing all the 'hits' here 2 or more observations in source and/or in ID passing the threshold set by the supplied criteria.

Usage

```
find_hits(data=data, ID= 'LOCUS', source=NULL, values=list(start=11, stop=21),
 var.cuts=FALSE, low.cut=NULL, high.cut=NULL, cutoff=0.05, Z=NULL, ...)
```

Arguments

`data` Dataframe containing a column of identifiers and column(s) of assay data providing scores to determine if an individual is a putative hit.

`ID` The name of the column containing individual identifiers. Must contain same values or as source.

`source` A list of individuals (contained in data) to be tested to see if they are a hit.

`values` Values (or columns) in data that are to be used to determine if an observation is a putative hit.

`var.cuts` Logical, will variable cutoffs be used for each of the assays (columns)? Must provide high.cut and low.cut if TRUE

`low.cut` A list of values (same length as the number of assay columns) giving the MAXIMUM value for an observation to be considered BELOW 'normal'.

`high.cut` A list of values (same length as the number of assay columns) giving the MINIMUM value for an observation to be considered ABOVE 'normal'.

`cutoff` p value below which observations are considered a putative hit.

`Z` Z score which is considered a hit.

`...` Other parameters.

Details

This function uses data coming out of the cdf.pval function or data with Zscores. Suggestions for using pvalue data are given below. The whole data object can be used, including if there are additional descriptors. ID refers to the identifier for individuals. Does not need to be unique. source is optional and contains a list of identifiers to be test for putative hits. If there are multiple individuals with the same ID (ex, in the same test group) then over half of them need to meet the criteria to be a putative hit. values indicates the columns containing values to evaluate, with start = the position of the first column and stop = the position of the last column.

If you wish to use a different cutoff for each column, then set var.cuts = TRUE and supply lists for both low.cut and high.cut that correspond to the largest value to be considered a hit on the low side (ex low abundance) and the smallest value to be considered a hit on the high side (ex high abundance), respectively. Alternatively, cutoff is used for data coming out of cdf.pval. cutoff=0.05 then values <=0.025 and values >= 0.975 will be considered putative hits. If Zscores are provided (or other criteria where values >= abs(x) are considered a hit), then Z should be used to define a cutoff.

data are subsetted based on the column (ID) either by all levels (e.g. group A, group B) or by source, if provided. Each column in values (e.g. assay) is evaluated to see if any individuals in that column meet the criteria for a putative hit. If more than half of the individuals meet the criteria to be a putative hit for that column, all the individuals belonging to that level are put into the output data frame. If not, then the remaining columns are evaluated or it moves to the next level. Individual responses that are low or high are evaluated separately.

Value

find_hits returns a dataframe containing putative hits and data for other individuals in their group.

Author(s)

Shannon M. Bell

References

Bell SM, Burgoon LD, Last RL. MIPHENO: Data normalization for high throughput metabolite analysis. *BMC Bioinformatics* 2012, 13(10)

Examples

```
#See the sweave document in the corresponding paper for examples
```

---

mad.scores *Calculates the mad score (zscore)*

Description

Returns a dataframe with the desired score (e.g. Zscore) for observation based on the algorithm for calculating Z scores described in Lu et al 2008. Calculations are done based on value of parameter.

Usage

```
mad.scores(data, parameter='FLATCODE', n=3, out=c('Zscore', 'label'), ...)
```

Arguments

data	Dataframe or Matrix of inputs for which the zscore is to be calculated
parameter	The parameter (given by column name) on which the mad score is to be calculated. This is the only non-numeric column allowed.
n	the number of Median Absolute Deviations (MAD) from the center which is considered a 'mutant' or putative hit.
out	The desired output value, either the 'Zscore' (quantitative) or 'label' (qualitative, 3 classes).
...	Other parameters.

Details

Data should be presorted according to any identifiers and `parameter` as labels not used in the calculations will be removed. `data` should be a dataframe with `parameter` as a factor (or character) and the rest of the values numeric.

Each column (other than `parameter`) is considered independent. Rows (individual responses) are extracted from `data` according to `parameter` and the MAD is calculated for each column (assay). If out = 'label', then a test is done to see if the result (Zscore) is greater or less than abs(n) and scored 'high' or 'low' accordingly.

Value

`mad.scores` returns a dataframe containing the Zscores or labels for the observations. The first column will contain the `parameter` label. If data will be returned in the same order as it was input so long as a sort based on `parameter` was completed ahead of time.

Author(s)

Shannon M. Bell

References

Bell SM, Burgoon LD, Last RL. MIPHENO: Data normalization for high throughput metabolite analysis. *BMC Bioinformatics* 2012, 13(10)

Lu Y, et al. New connections across pathways and cellular processes: Industrialized mutant screening reveals novel associations between diverse phenotypes in Arabidopsis. *Plant Physiol* 2010, 152(2):529-540

Examples

```
#See the sweave document in the corresponding paper for examples
```

rm.outliers *Post-Hoc outlier removal for high throughput data*

Description

Returns a dataframe with outlier groups removed. Note that each column (other than the parameter column) is treated as a separate assay. Therefore if one 'group' does not meet the criteria for inclusion in 1 assay (column value), but does for all others, only the data for the assay failing the quality control will be removed.

Usage

```
rm.outliers(data, parameter='FLATCODE', n=3, ...)
```

Arguments

data	Dataframe or Matrix of inputs for which outliers are to be removed
parameter	The parameter (given by column name) on which defines a subgroup. This is the only non-numeric column allowed.
n	the number of Median Absolute Deviations (MAD) from the global median a subgroup is allowed to be before it is considered an outlier.
...	Other parameters.

Details

Data should be presorted according to both identifier and parameter prior to running command. Order will be retained if this is followed. data should be a dataframe with a parameter serving as a label and the rest of the values numeric. Note that parameter should be the attribute where the most error is expected. A visual inspection using box and whisker plots may be helpful in determining the best variable to use. data is first broken down into groups based on parameter. Both the group median and the median of all groups (global median) is calculated. Groups where the absolute value of the difference between the group median and global median is greater than n * MAD(group medians) for a given attribute (column) have their values removed (ie set to NA). Data for the group is retained for columns that pass this criteria.

Value

rm.outliers returns a dataframe containing data with values not passing the filter converted to NA.

Author(s)

Shannon M. Bell

References

Bell SM, Burgoon LD, Last RL. MIPHENO: Data normalization for high throughput metabolite analysis. *BMC Bioinformatics* 2012, 13(10)

Examples

```
#See the sweave document in the corresponding paper for examples
```

Index