Package 'topconfectsql'

June 7, 2018

Title Top Results by Confident Effect Size, Quasi-Likelihood branch

Version 1.0.1

Description Uses limma's treat or edgeR's glmTreat to rank genes (or other features) by confident log2 fold change.

Depends R (>= 3.3.0)

Imports methods,
utils,
stats,
assertthat,
magrittr,
dplyr,
purrr,
ggplot2,
limma,
edgeR,
nloptr,
memoise

Suggests NBPSeq,
testthat,
knitr,
rmarkdown,
reshape2,
tidyr,
readr,
org.At.tair.db,
org.Sc.sgd.db,
AnnotationDbi

License LGPL-2.1 | file LICENSE

URL https://github.com/pfh/topconfects

Encoding UTF-8

LazyData true

RoxygenNote 6.0.1

VignetteBuilder knitr

R topics documented:

`confects_plot` *Top confident effect sizes plot*

Description

Create a ggplot2 object showing the confect, effect, and average expression level of top features in a Topconfects object.

Usage

```
confects_plot(confects, n = 50, limits = NULL)
```

Arguments

`confects`	A "Topconfects" class object, as returned from limma_confects, edger_confects, etc.
`n`	Number if items to show.
`limits`	c(lower, upper) limits on x-axis.

Details

For each gene, the estimated effect is shown as a dot. The confidence bound is shown as a line to positive or negative infinity, showing the set of non-rejected effect sizes for the feature.

Value

A ggplot2 object. Working non-interactively, you must print() this for it to be displayed.

confects_plot_me *Mean-expression vs effect size plot*

Description

Like plotMD in limma, but shows "confect" on the y axis rather than "effect" ("effect" is shown underneath in grey). This may be useful for assessing whether effects are only being detected only in highly expressed genes.

Usage

```
confects_plot_me(confects)
```

Arguments

confects	A "Topconfects" class object, as returned from limma_confects, edger_confects, etc.

Value

A ggplot2 object. Working non-interactively, you must print() this for it to be displayed.

edger_confects *Confident effect sizes based on the edgeR Quasi-Likelihood method, both linear and non-linear*

Description

For all possible absolute log2 fold changes, which genes have at least this fold change at a specified False Discovery Rate?

Usage

```
edger_confects(fit, coef = NULL, contrast = NULL, effect = NULL,
  fdr = 0.05, step = 0.01, null = "worst.case")
```

Arguments

fit	An edgeR DGEGLM object produced using glmQLFit.
coef	Coefficient to test, as per glmTreat. Use either coef or contrast or effect.
contrast	Contrast to test, as per glmTreat. Use either coef or contrast or effect.
effect	A non-linear effect, created with one of the effect_... functions. Use either coef or contrast or effect.
fdr	False Discovery Rate to control for.
step	Granularity of log2 fold changes to test.
null	"null" parameter passed through to edger::glmTreat (if coef or contrast given). Choices are "worst.case" or "interval". Note that the default here is "worst.case", to be consistent with other functions in topconfects. This differs from the default for glmTreat.

Value

See nest_confects for details of how to interpret the result.

Technical note: when using a non-linear effect size: Signed confects are based on TREAT-style p-values. Unsigned confects (generally with df>1) are based on comparing the best fit within the H0 region to the best fit overall, which may up to double p-values.

`edger_group_confects` *Group confects (differential 5' or 3' end usage, etc)*

Description

Find differential exon usage, etc.

Usage

```
edger_group_confects(fit, group_id, group_effect, fdr = 0.05, step = 0.01)

limma_group_confects(object, group_id, group_effect, fdr = 0.05,
  step = 0.01, trend = FALSE)
```

Arguments

`fit`	For edgeR, an edgeR DGEGLM object.
`group_id`	A factor of length `nrow(fit)`, assigning items to groups (eg genes).
`group_effect`	A group effect object created by one of the A `group_effect_...` functions.
`fdr`	False Discovery Rate to maintain.
`step`	Step size when calculating confident effect sizes.
`object`	For limma, an expression matrix or EList object, or anything limma's lmFit will accept.
`trend`	For limma, should `eBayes(trend=TRUE)` be used?

Details

If the order of the members of a group is important, ensure the order of rows is correct in the original matrix.

Groups with less than two members will be ignored.

To construct a group design matrix, the design from `fit` will be repeated in a block diagonal matrix.

Value

See nest_confects for details of how to interpret the result.

effect_contrast *Simple linear contrast effect.*

Description

This is included for completeness and testing. limma_confects() and edger_confects() provide a faster version of this using limma's treat and edgeR's topTreat functions.

Usage

```
effect_contrast(contrast)
```

Arguments

contrast	A vector of the same length as the number of coefficients. The effect size is the dot product of this with the coefficients.

Value

An object defining how to calculate an effect size.

effect_link_log2 *Adapt an effect size object to work with coefficients estimated from log2 transformed data or with a log2 link function*

Description

Differential expression analysis usually fits coefficients to log2 transformed values, or with a log2 link function. This function adapts an effect size object to work with such log2-scale coefficients.

Usage

```
effect_link_log2(effect)
```

Arguments

effect	An object defininining an effect size on coefficients with untransformed scale.

Value

An object defining how to calculate an effect size from coefficients on a logarithmic scale.

effect_shift *Shift of mass effect*

Description

Detect a "shift of mass" between two conditions. For example expression might move later in a time series in an experimental condition vs a control. If all expression shifted to a later time in the experimental condition, this would be given an effect size of 1. Conversely if all expression shifted to an earlier time, the effect size would be -1.

Usage

```
effect_shift(coef1, coef2)

effect_shift_log2(coef1, coef2)
```

Arguments

coef1	Column numbers in the design matrix for the first condition, in some meaningful order.
coef2	Corresponding column numbers for the second condition.

Details

This can be viewed as similar to Somers' D.

effect_shift_log2 is adapted to work with log2 scaled coefficients. This is almost certainly the version you want.

Note that this effect size is not symmetric: effect_shift_log2(c(1,2),c(3,4)) and effect_shift_log2(c(1,3),c(will give different results.

Value

An object defining how to calculate an effect size.

group_effect_shift *Group effect for shifts in 5' or 3' end usage*

Description

Create a group effect object to detect a shift in usage of features which are ordered within a group.

Usage

```
group_effect_shift(design, coef1, coef2, design_common = NULL)

group_effect_shift_log2(design, coef1, coef2, design_common = NULL)
```

Arguments

design	Design matrix.
coef1	Column number of coefficient for first condition in design matrix.
coef2	Column number of coefficient for second condition in design matrix.
design_common	Experimental! Optional sample-level design matrix. For example, this can be used to account for a batch effect or matched samples.

Details

group_effect_shift_log2 is almost certainly the version you want.

The coefficients should represent the expression levels in two different conditions.

Value

A group effect object.

See Also

effect_shift, effect_shift_log2

---

limma_confects *Confident log2 fold changes using limma's treat function, linear effects only*

Description

For all possible absolute log2 fold changes, which genes have at least this fold change at a specified False Discovery Rate?

Usage

```
limma_confects(fit, coef = NULL, fdr = 0.05, step = 0.01, trend = FALSE)
```

Arguments

fit	A limma MArrayLM object.
coef	Column number of coefficient or contrast to test.
fdr	False Discovery Rate to control for.
step	Granularity of log2 fold changes to test.
trend	Should `treat(..., trend=TRUE)` be used?

Details

`fit` should be produced using `lmFit`, and optionally `contrasts.fit` if a contrast is needed. It is not necessary to use `eBayes`, this function calls `eBayes` itself.

Value

See nest_confects for details of how to interpret the result.

limma_nonlinear_confects

Confident non-linear effect sizes using limma

Description

Confident non-linear effect sizes using limma

Usage

```
limma_nonlinear_confects(object, design, effect, fdr = 0.05, step = 0.01,
  trend = FALSE)
```

Arguments

object	An expression matrix or EList object, or anything limma's lmFit will accept.
design	Design matrix.
effect	A non-linear effect, created with one of the effect_... functions.
fdr	False Discovery Rate to control for.
step	Granularity of log2 fold changes to test.
trend	Should `eBayes(trend=TRUE)` be used?

Value

Technical note: Signed confects are based on TREAT-style p-values. Unsigned confects (generally with df>1) are based on comparing the best fit within the H0 region to the best fit overall, which may up to double p-values.

See `nest_confects` for details of how to interpret the result.

nest_confects

General purpose function to find sets of discoveries for a range of effect sizes, controlling FDR

Description

Find sets of discoveries for a range of effect sizes, controlling the False Discovery Rate for each set.

Usage

```
nest_confects(n, pfunc, fdr = 0.05, step = 0.01)
```

Arguments

n	Number of items being tested.
pfunc	A function(indices, effect_size) to calculate p-values. Indices is a subset of 1:n giving the p-values to be computed.
fdr	False Discovery Rate to control for.
step	Granularity of effect sizes to test.

Value

A "Topconfects" object, containing a table of results and various associated information.

The most important part of this object is the $table element, a data frame with the following columns:

- `rank` - Ranking by `confect` and for equal `confect` by p-value at that effect size.
- `index` - Number of the test, between 1 and n.
- `confect` - CONfident efFECT size.

The usage is as follows: To find a set of tests which have effect size at least `x` with the specified FDR, take the rows with `abs(confect) >= x`.

Some tests may have been given the same `confect`. To maintain the FDR, all or none should be chosen.

With this caveat understood, one may essentially take the top however many rows of the data frame and these will be the best set of results of that size to dependably have an effect size that is as large as possible.

Some wrappers around this function may add a sign to the `confect` column, if it makes sense to do so. They will also generally add an `effect` column, containing an estimate of the effect size that aims to be unbiassed rather than a conservative lower bound.

`normal_confects` *Confident effect sizes from from normal or t distributions*

Description

A general purpose confident effect size function for where a normal or t distribution can be assumed. Calculates confident effect sizes based on a mean and standard deviation (normal distribution) or mean and scale (t distribution).

Usage

```
normal_confects(mean, sd = 1, df = Inf, fdr = 0.05, step = 0.01)
```

Arguments

`mean`	A vector of means.
`sd`	A single or vector of standard deviations (or if t distribution, scales).
`df`	A single or vector of degrees of freedom, for t-distribution. Inf for normal distribution.
`fdr`	False Discovery Rate to control for.
`step`	Granularity of effect sizes to test.

Value

See `nest_confects` for details of how to interpret the result.

`rank_rank_plot` *A plot to compare two rankings*

Description

This is useful, for example, when comparing different methods of ranking potentially interesting differentially expressed genes.

Usage

```
rank_rank_plot(vec1, vec2, label1 = "First ranking",
  label2 = "Second ranking", n = 40)
```

Arguments

`vec1`	A vector of names.
`vec2`	Another vector of names.
`label1`	A label to go along with vec1.
`label2`	A label to go along with vec2.
`n`	Show at most the first n names in vec1 and vec2.

Value

A ggplot2 object. Working non-interactively, you must print() this for it to be displayed.

Index