Package ‘EMDomics’

October 5, 2015

Title Earth Mover's Distance for Differential Analysis of Genomics Data

Description The EMDomics algorithm is used to perform a supervised two-class analysis to measure the magnitude and statistical significance of observed continuous genomics data between two groups. Usually the data will be gene expression values from array-based or sequence-based experiments, but data from other types of experiments can also be analyzed (e.g. copy number variation). Traditional methods like Significance Analysis of Microarrays (SAM) and Linear Models for Microarray Data (LIMMA) use significance tests based on summary statistics (mean and standard deviation) of the two distributions. This approach lacks power to identify expression differences between groups that show high levels of intra-group heterogeneity. The Earth Mover's Distance (EMD) algorithm instead computes the ``work'' needed to transform one distribution into the other, thus providing a metric of the overall difference in shape between two distributions. Permutation of sample labels is used to generate q-values for the observed EMD scores.

Version 1.0.0

biocViews Software, DifferentialExpression, GeneExpression, Microarray

Maintainer Daniel Schmolze <emd@schmolze.com>

Depends R (>= 3.2.0)

Imports emdist, BiocParallel, matrixStats, ggplot2

Suggests knitr

License MIT + file LICENSE

LazyData true

VignetteBuilder knitr

NeedsCompilation no

Author Daniel Schmolze [aut, cre], Andrew Beck [aut], Sheida Nabavi [aut]

R topics documented:

`emdomics-package` *Earth Mover's Distance algorithm for differential analysis of genomics data.*

Description

`calculate_emd` will usually be the only function needed.

`calculate_emd` *Earth Mover's Distance for differential analysis of genomics data*

Description

This is the main user interface to the **EMDomics** package, and will usually the only function needed.

The algorithm is used to compare genomics data between two groups, refered to herein as "group A" and "group B". Usually the data will be gene expression values from array-based or sequence-based experiments, but data from other types of experiments can also be analyzed (i.e. copy number variation).

Traditional methods like Significance Analysis of Microarrays (SAM) and Linear Models for Microarray Data (LIMMA) use significance tests based on summary statistics (mean and standard deviation) of the two distributions. This approach tends to give non-significant results if the two distributions are highly heterogeneous, which can be the case in many biological circumstances (e.g sensitive vs. resistant tumor samples).

The Earth Mover's Distance algorithm instead computes the "work" needed to transform one distribution into the other, thus capturing possibly valuable information relating to the overall difference in shape between two heterogeneous distributions.

The EMD-based algorithm implemented in **EMDomics** has two main steps. First, a matrix (e.g. of expression data) is divided into data for "group A" and "group B", and the EMD score is calculated using the two groups for each gene in the data set. Next, the labels for group A and group B are randomly permuted a specified number of times, and an EMD score for each permutation is calculated. The median of the permuted scores for each gene is used as the null distribution, and the False Discovery Rate (FDR) is computed for a range of permissive to restrictive significance thresholds. The threshold that minimizes the FDR is defined as the q-value, and is used to interpret the significance of the EMD score analogously to a p-value (e.g. q-value < 0.05 = significant.)

Note that q-values of 0 are adjusted to 1/(nperm+1). For this reason, the `nperm` parameter should not be too low (the default of 100 is reasonable).

Usage

```
calculate_emd(data, samplesA, samplesB, binSize = 0.2, nperm = 100,
  verbose = TRUE, parallel = TRUE)
```

Arguments

`data` A matrix containing genomics data (e.g. gene expression levels). The rownames should contain gene identifiers, while the column names should contain sample identifiers.

`samplesA` A vector of sample names identifying samples in `data` that belong to "group A". The names must corresponding to column names in `data`.

`samplesB` A vector of sample names identifying samples in `data` that belong to "group B". The names must corresponding to column names in `data`.

`binSize` The bin size to be used when generating histograms of the data for "group A" and "group B". Defaults to 0.2.

`nperm` An integer specifying the number of randomly permuted EMD scores to be computed. Defaults to 100.

`verbose` Boolean specifying whether to display progress messages.

`parallel` Boolean specifying whether to use parallel processing via the **BiocParallel** package. Defaults to `TRUE`.

Value

The function returns an `EMDomics` object.

See Also

`EMDomics` `emd2d`

Examples

```
# 100 genes, 100 samples
dat <- matrix(rnorm(10000), nrow=100, ncol=100)
rownames(dat) <- paste("gene", 1:100, sep="")
colnames(dat) <- paste("sample", 1:100, sep="")

# "group A" = first 50, "group B" = second 50
groupA <- colnames(dat)[1:50]
groupB <- colnames(dat)[51:100]
results <- calculate_emd(dat, groupA, groupB, nperm=10, parallel=FALSE)
head(results$emd)
```

calculate_emd_gene *Calculate EMD score for a single gene*

Description

Calculate EMD score for a single gene

Usage

```
calculate_emd_gene(vec, samplesA, samplesB, binSize = 0.2)
```

Arguments

vec	A named vector containing data (e.g. expression data) for a single gene.
samplesA	A vector of sample names identifying samples in vec that belong to "group A".
samplesB	A vector of sample names identifying samples in vec that belong to "group B".
binSize	The bin size to be used when generating histograms for "group A" and "group B".

Details

The data in vec is divided into "group A" and "group B" by the identifiers given in samplesA and samplesB. The hist function is used to generate histograms for the two resulting groups, and the densities are retrieved and passed to emd2d to compute the EMD score.

Value

The emd score is returned.

See Also

emd2d

Examples

```
# 100 samples
vec <- rnorm(100)
names(vec) <- paste("sample", 1:100, sep="")

# "group A" = first 50, "group B" = second 50
groupA <- names(vec)[1:50]
groupB <- names(vec)[51:100]

calculate_emd_gene(vec, groupA, groupB)
```

EMDomics *Create an EMDomics object*

Description

This is the constructor for objects of class 'EMDomics'. It is used in calculate_emd to construct the return value.

Usage

```
EMDomics(data, samplesA, samplesB, emd, emd.perm)
```

Arguments

data
: A matrix containing genomics data (e.g. gene expression levels). The rownames should contain gene identifiers, while the column names should contain sample identifiers.

samplesA
: A vector of sample names identifying samples in data that belong to "group A". The names must corresponding to column names in data.

samplesB
: A vector of sample names identifying samples in data that belong to "group B". The names must corresponding to column names in data.

emd
: A matrix containing a row for each gene in data, and with the following columns:

- emd The calculated emd score.
- fc The log2 fold change of "group A" samples relative to "group B" samples.
- q-value The calculated q-value.

The row names should specify the gene identifiers for each row.

emd.perm
: A matrix containing a row for each gene in data, and with a column containing emd scores for each random permutation calculated via calculate_emd.

Value

The function combines it's arguments in a list, which is assigned class 'EMDomics'. The resulting object is returned.

See Also

calculate_emd

plot_density *Plot distributions and EMD score for a gene.*

Description

The data for the specified gene is retrieved from emdobj$emd. emdobj$samplesA and emdobj$samplesB are used to divide the data into two distributions, which are then visualized as density distributions. The calculated EMD score for the specified gene is displayed in the plot title.

Usage

```
plot_density(emdobj, gene_name)
```

Arguments

emdobj	An EMDomics object, typically returned via a call to calculate_emd.
gene_name	The gene to visualize. The name should be defined as a row name in emdobj$emd.

Value

A ggplot object is returned. If the value is not assigned, a plot will be drawn.

See Also

calculate_emd ggplot

Examples

```
# 100 genes, 100 samples
dat <- matrix(rnorm(10000), nrow=100, ncol=100)
rownames(dat) <- paste("gene", 1:100, sep="")
colnames(dat) <- paste("sample", 1:100, sep="")

# "group A" = first 50, "group B" = second 50
groupA <- colnames(dat)[1:50]
groupB <- colnames(dat)[51:100]

results <- calculate_emd(dat, groupA, groupB, nperm=10)
plot_density(results, "gene5")
```

plot_emdnull *Plot null distribution of permuted EMD scores vs. calculated EMD scores.*

Description

The median of the randomly permuted EMD scores (i.e. the null distribution) is plotted on the x-axis, vs. the observed EMD scores on the y-axis. The line y=x is superimposed.

Usage

```
plot_emdnull(emdobj)
```

Arguments

emdobj	An EMDomics object, typically returned via a call to calculate_emd.

Value

A ggplot object is returned. If the value is not assigned, a plot will be drawn.

See Also

calculate_emd ggplot

Examples

```
# 100 genes, 100 samples
dat <- matrix(rnorm(10000), nrow=100, ncol=100)
rownames(dat) <- paste("gene", 1:100, sep="")
colnames(dat) <- paste("sample", 1:100, sep="")

# "group A" = first 50, "group B" = second 50
groupA <- colnames(dat)[1:50]
groupB <- colnames(dat)[51:100]

results <- calculate_emd(dat, groupA, groupB, nperm=10)
plot_emdnull(results)
```

plot_perms — *Plot histogram of EMD scores calculated via random permutation.*

Description

The permuted EMD scores stored in `emdobj$emd.perm` are plotted as a histogram.

Usage

```
plot_perms(emdobj)
```

Arguments

`emdobj`	An `EMDomics` object, typically returned via a call to `calculate_emd`.

Value

A `ggplot` object is returned. If the value is not assigned, a plot will be drawn.

See Also

`calculate_emd` `ggplot`

Examples

```
# 100 genes, 100 samples
dat <- matrix(rnorm(10000), nrow=100, ncol=100)
rownames(dat) <- paste("gene", 1:100, sep="")
colnames(dat) <- paste("sample", 1:100, sep="")

# "group A" = first 50, "group B" = second 50
groupA <- colnames(dat)[1:50]
groupB <- colnames(dat)[51:100]

results <- calculate_emd(dat, groupA, groupB, nperm=10)
plot_perms(results)
```

Index