Package ‘phenoDist’

October 8, 2014

Version 1.12.0

Title Phenotypic distance measures

Author Xian Zhang, Gregoire Pau, Wolfgang Huber, Michael Boutros

Maintainer Xian Zhang <xianzhang@gmail.com>

Depends R (>= 2.9.0), imageHTS, e1071

Suggests GOstats, MASS

Description PhenoDist is designed for measuring phenotypic distance in image-based high-throughput screening, in order to identify strong phenotypes and to group treatments into functional clusters.

Reference

License LGPL-2.1

URL http://www.dkfz.de/signaling,http://www.embl.de/research/units/genome_biology/huber/

biocViews CellBasedAssays

R topics documented:

clusterDist 2
ctlSeparatn 3
distToNeg 4
enrichAnalysis 5
PDMByFactorAnalysis 6
PDMByKS 8
PDMBySvmAccuracy 9
PDMBySvmWeightVector 10
PDMByWellAvg 11
repCorr 13
repDistRank 14

Index **15**

clusterDist *clusterDist*

Description

Clustering analysis based on a distance matrix.

Usage

```
clusterDist(x, distMatrix, clusterFun=hclust, ...)
```

Arguments

x	An imageHTS object.
distMatrix	A pair-wise distance matrix or a dist object.
clusterFun	A character string defining the cluster function.
...	Additional arguments to be passed to the cluster function.

Details

This function performs a clustering analysis based on a pair-wise distance matrix such as generated by PDMBySvmAccuracy.

Value

The return from the cluster function, such as an hclust object returned from the hclust function.

Author(s)

Xian Zhang

See Also

hclust, PDMBySvmAccuracy

Examples

```
library(phenoDist)

## load the imageHTS object
load(system.file(kimorph, kimorph.rda, package=phenoDist))
x@localPath <- file.path(tempdir(), kimorph)

## load sample phenotypic distance matrix
load(system.file(kimorph, svmAccPDM_Pl1.rda, package=phenoDist))

## phenotypic clustering
```

```
  phenoCluster <- clusterDist(x, distMatrix=svmAccPDM_Pl1, clusterFun=hclust, method=ward)

   ## Not run:
   require(GOstats)
  GOEnrich <- enrichAnalysis(x, cl=cutree(phenoCluster, k=5), terms=GO, annotation=org.Hs.eg.db, pvalueCutoff=0.0

## End(Not run)
```

`ctlSeparatn` *Calculate control separation*

Description

This function calculates the Z'-factor between negative and positive controls, based on phenotypic distance measurements.

Usage

```
ctlSeparatn(x, pheno, neg=rluc, pos=ubc, ...)
```

Arguments

`x`	An `imageHTS` object.
`pheno`	A numeric vector for distances to negative controls, as returned from `distToNeg`.
`neg`	A character string to identify negative controls.
`pos`	A character string to identify positive controls.
`...`	Additional arguments to be passed to the `Zprime` function of the `imageHTS` package.

Details

Phenotypes of negative and positive controls are defined as their phenotypic distance to negative controls. The Z'-factor is a metric measuring the separation between negative controls and positive controls (Zhang et al. 1999). This function calls the `Zprime` function of the `imageHTS` package to calculate the Z'-factor. Please see `Zprime` for detailed description of the definition and calculation of the Z'-factor.

Value

The Z'-factor value.

Author(s)

Xian Zhang

References

J. H. Zhang, T. D. Chung, K. R. Oldenburg. A Simple Statistical Parameter for Use in Evaluation and Validation of High Throughput Screening Assays. J Biomol Screening, 1999.

See Also

`distToNeg`, `Zprime`

Examples

```
## see distToNeg.
```

---

distToNeg *distToNeg*

Description

From a pair-wise distance matrix, this function extracts the corresponding distance between a sample and negative controls.

Usage

```
distToNeg(x, distMatrix, neg=rluc)
```

Arguments

`x`	An `imageHTS` object
`distMatrix`	A pair-wise distance matrix, as generated by `PDMBySvmAccuracy`.
`neg`	A character string to identify negative controls.

Details

This function averages the distance measurements between the sample and all negative controls on the same plate.

Value

A numeric vector with length equal to the dimension of the distance matrix.

Author(s)

Xian Zhang

See Also

`PDMBySvmAccuracy`

Examples

```
library(phenoDist)

## load the imageHTS object
load(system.file(kimorph, kimorph.rda, package=phenoDist))
x@localPath <- file.path(tempdir(), kimorph)

## load sample phenotypic distance matrix
load(system.file(kimorph, svmAccPDM_Pl1.rda, package=phenoDist))

## replicate ranking
ranking <- repDistRank(x, distMatrix=svmAccPDM_Pl1)
summary(ranking)

## phenotypic distance to negative control
pheno <- distToNeg(x, distMatrix=svmAccPDM_Pl1, neg=rluc)

## separation between negative and positive controls
ctlSeparatn(x, pheno, neg=rluc, pos=ubc, method=robust)

## replicate correlation coefficient
repCorr(x, pheno)
```

---

enrichAnalysis *enrichAnalysis*

Description

This function performs enrichment analysis on genes within each cluster, with genes from all clusters as the gene universe.

Usage

```
enrichAnalysis(x, cl, terms=c(GO,KEGG), verbose=FALSE, ...)
```

Arguments

x	An imageHTS object.
cl	A named numeric vector, with names being gene Entrez IDs and values indicating grouping.
terms	A character string indicating which annotation should be used. This must be (an abbreviation of) one of the strings 'GO' or 'KEGG'.
verbose	A logical scalar indicating whether progress should be reported.
...	Additional arguments to be passed to the hyperGTest function of the GOstats package.

Details

Gene enrichment analysis tests whether certain gene annotations (e.g., GO terms or KEGG ids) are enriched in genes of interest, compared with the gene universe. This function is designed to analyze gene enrichment in clusters, with genes in the tested cluster being genes of interest and genes in all clusters being the gene universe.

The Hypergeometric test is performed by the `hyperGTest` function of the `GOstats` package. Please refer to `hyperGTest` for additional arguments.

Value

A list of `HyperGResult` instances.

Author(s)

Xian Zhang

See Also

`GOstats`, `clusterDist`

Examples

```
## see clusterDist.
```

PDMByFactorAnalysis *Calculate phenotypic distance matrix by factor analysis*

Description

This function transforms the cell features by factor analysis and computes the phenotypic distance matrix.

Usage

```
PDMByFactorAnalysis(x, unames, selectedCellFtrs, distMethod=c(manhattan,euclidean, correlation,mahala
```

Arguments

`x`	An `imageHTS` object.
`unames`	A character vector, containing the well names from where to collect the cell features. See `getUnames` for details.
`selectedCellFtrs`	A character vector for cell features to be used in the calculation. If missing, all features are used.

distMethod	A character string indicating which distance method should be used. This must be (an abbreviation of) one of the strings 'manhattan', 'euclidean', 'correlation' or 'mahalanobis'.
nFactors	An integer scalar for the number of factors.
scores	A character string indicating the type of scores to be reported by factor analysis. This must be (an abbreviation of) one of the strings 'regression' or 'Bartlett'.
...	Additional arguments to be passed to the `factanal` function of the `stats` pacakge.

Details

This function first collects individual cell features in all wells (which could be time and memory consuming), performs factor analysis on cell features and transforms cell features into a certain number of factors, and then the factors are averaged by well and passed to `PDMByWellAvg` to calculate the phenotypic distance matrix.

Value

A symmetric distance matrix with dimensions equaling to the length of `unames`.

Author(s)

Xian Zhang

See Also

`factanal`, `PDMByWellAvg`

Examples

```
library(phenoDist)

## load the imageHTS object
load(system.file(kimorph, kimorph.rda, package=phenoDist))
x@localPath <- file.path(tempdir(), kimorph)

## segmentation and feature extraction
unames <- setdiff(getUnames(x), getUnames(x, content=empty))

## calculate pair-wise svm distance matrix
load(system.file(kimorph, selectedFtrs.rda, package=phenoDist))
pdm <- PDMByFactorAnalysis(x, unames=getUnames(x, plate=1, row=2:3, col=3), selectedCellFtrs, distMethod=euclide
pdm
```

PDMByKS *Compute phenotypic distance matrix by Kolmogorov-Smirnov statistics*

Description

This function calculates the Kolmogorov-Smirnov statistic between a given sample and the negative control, for each feature, and then computes the phenotypic distance matrix based on the Kolmogorov-Smirnov statistics.

Usage

```
PDMByKS(x, unames, neg=rluc, selectedCellFtrs, distMethod=c(manhattan, euclidean, correlation, mahalan
```

Arguments

x	An imageHTS object.
unames	A character vector, containing the well names from where to collect the cell features. See getUnames for details.
neg	A character string to identify negative controls.
selectedCellFtrs	A character vector for cell features to be used in the calculation. If missing, all features are used.
distMethod	A character string indicating which distance method should be used. This must be (an abbreviation of) one of the strings 'manhattan', 'euclidean', 'correlation' or 'mahalanobis'.

Details

For each well, this function collects features of all cells, and performs a Kolmogorov-Smirnov test for each feature against the corresponding cell features from negative control wells, with the function ks.test of the stats package. The Kolmogorov-Smirnov statistics are collected for all wells and passed to PDMByWellAvg to calculate the phenotypic distance matrix.

Value

A symmetric distance matrix with the number of rows equaling to the length of unames.

Author(s)

Xian Zhang

See Also

ks.test, PDMByWellAvg

Examples

```
library(phenoDist)

## load the imageHTS object
load(system.file(kimorph, kimorph.rda, package=phenoDist))
x@localPath <- file.path(tempdir(), kimorph)

## calculate pair-wise svm distance matrix
load(system.file(kimorph, selectedFtrs.rda, package=phenoDist))
pdm <- PDMByKS(x, unames=getUnames(x,plate=1, row=2:3, col=3), neg=rluc, selectedCellFtrs=selectedCellFtrs, dist
pdm
```

PDMBySvmAccuracy *Compute phenotypic distance matrix by SVM classification accuracy*

Description

This function performs an SVM classification between two given samples, and calculates the classification accuracy via cross validation as the phenotypic distance between the two samples. For multiple samples, the function returns a pair-wise distance matrix.

Usage

```
PDMBySvmAccuracy(x, unames, selectedCellFtrs, cross=5, verbose=FALSE, ...)
```

Arguments

x	An imageHTS object.
unames	A character vector, containing the well names from where to collect the cell features. See getUnames for details.
selectedCellFtrs	A character vector for cell features to be used in the calculation. If missing, all features are used.
cross	An interger scalar indicating how many folds of cross validation should be performed.
verbose	A logical scalar indicating whether progress should be reported.
...	Additional arguments to be passed to the svm function of the e1071 package.

Details

For every pair of wells, this function collects features of all cells from both wells, and performs a bi-class classification using Support Vector Machine (SVM). The classification accuracy is defined as the phenotypic distance for the distance matrix.

Value

A symmetric distance matrix with the number of rows equaling to the length of `unames`.

Author(s)

Xian Zhang

See Also

`svm`

Examples

```
library(phenoDist)

## load the imageHTS object
load(system.file(kimorph, kimorph.rda, package=phenoDist))
x@localPath <- file.path(tempdir(), kimorph)

## calculate pair-wise svm distance matrix
load(system.file(kimorph, selectedFtrs.rda, package=phenoDist))
pdm <- PDMBySvmAccuracy(x, unames=getUnames(x, plate=1, row=2:3, col=3), selectedCellFtrs=selectedCellFtrs, cros
```

`PDMBySvmWeightVector` *Compute phenotypic distance matrix by SVM weight vector*

Description

This function performs an SVM classification between a given sample and the negative control, calculates the weight vector, and then computes the phenotypic distance matrix based on the weight vectors.

Usage

```
PDMBySvmWeightVector(x, unames, neg=rluc, selectedCellFtrs, distMethod=c(manhattan,euclidean, correla
```

Arguments

`x`	An `imageHTS` object.
`unames`	A character vector, containing the well names from where to collect the cell features. See `getUnames` for details.
`neg`	A character string to identify the negative controls.
`selectedCellFtrs`	A character vector for cell features to be used in the calculation. If missing, all features are used.

distMethod	A character string indicating which distance method should be used. This must be (an abbreviation of) one of the strings 'manhattan', 'euclidean', 'correlation' or 'mahalanobis'.
verbose	A logical scalar indicating whether progress should be reported.
kernel	The kernel argument for the svm function of the e1071 package.
...	Additional arguments to be passed to the svm function of the e1071 package.

Details

For each well, this function collects features of all cells from the well and all cells from the negative control wells, and performs a bi-class classification using Support Vector Machine (SVM). The classification weight vectors are calculated for all wells passed to PDMByWellAvg to compute the phenotypic distance matrix.

Value

A symmetric distance matrix with dimensions equaling to the length of unames.

Author(s)

Xian Zhang

See Also

svm, PDMByWellAvg

Examples

```
library(phenoDist)

## load the imageHTS object
load(system.file(kimorph, kimorph.rda, package=phenoDist))
x@localPath <- file.path(tempdir(), kimorph)

## calculate pair-wise svm distance matrix
load(system.file(kimorph, selectedFtrs.rda, package=phenoDist))
pdm <- PDMBySvmWeightVector(x, unames=getUnames(x,plate=1, row=2:3, col=3), neg=rluc, selectedCellFtrs=selected
```

PDMByWellAvg *Compute phenotypic distance matrix with well features*

Description

This function computes the phenotypic distance matrix, with cell features averaged by well.

Usage

```
PDMByWellAvg(profiles, selectedWellFtrs, transformMethod=c(none, scale, PCA), distMethod=c(manhattan
```

Arguments

`profiles` A data frame, containing the phenotypic profiles, as returned from the `summarizeWells` function of the `imageHTS` package.

`selectedWellFtrs` A character vector indicating well features to be used in the calculation. If missing, all features are used.

`transformMethod` A character string indicating which transformation method should be used. This must be (an abbreviation of) one of the strings 'none', 'scale' or 'PCA'.

`distMethod` A character string indicating which distance method should be used. This must be (an abbreviation of) one of the strings 'manhattan', 'euclidean', 'correlation' or 'mahalanobis'.

`nPCA` An integer scalar for the number of PCA dimensions to be used in the calculation.

Details

Pair-wise phenotypic distance measurements of the treatments in screen results in a phenotypic distance matrix. The features stored in `profiles` are transformed with the `transformMethod` and the distance matrix is calculated with the `distMethod`.

Value

A symmetric distance matrix with dimensions equaling to the number of rows of `profiles`.

Author(s)

Xian Zhang

See Also

`summarizeWells`

Examples

```
library(phenoDist)

## load the imageHTS object
load(system.file(kimorph, kimorph.rda, package=phenoDist))
x@localPath <- file.path(tempdir(), kimorph)

## calculate pair-wise svm distance matrix
load(system.file(kimorph, selectedFtrs.rda, package=phenoDist))
pdm <- PDMByWellAvg(profiles=summarizeWells(x, getUnames(x,plate=1, row=2:3, col=3), conf/featurepar.txt), sele
pdm
```

repCorr *repCorr*

Description

This function calculates the correlation between replicate phenotypes.

Usage

```
repCorr(x, pheno, ...)
```

Arguments

x	An imageHTS object.
pheno	A numeric vector for distance measures to negative control, as returned from distToNeg.
...	Additional arguments to be passed to the cor function of the stats pacakge.

Value

Correlation coefficient returned from the cor function of the stats pacakge.

Author(s)

Xian Zhang

See Also

cor, distToNeg

Examples

```
## see distToNeg.
```

`repDistRank` *repDistRank*

Description

For each treatment, this function calculates the ranking of its replicate in terms of phenotypic distance.

Usage

```
repDistRank(x, distMatrix)
```

Arguments

`x`	An `imageHTS` object.
`distMatrix`	A pair-wise distance matrix or `dist` object.

Details

For each treatment, this function ranks the distance measure to its replicate among the distance measurements to all other treatments.

Value

A numeric vector with length equal to the dimension of the distance matrix.

Author(s)

Xian Zhang

See Also

`PDMByWellAvg`

Examples

```
## see distToNeg.
```

Index

clusterDist, 2
ctlSeparatn, 3

distToNeg, 4

enrichAnalysis, 5

PDMByFactorAnalysis, 6
PDMByKS, 8
PDMBySvmAccuracy, 9
PDMBySvmWeightVector, 10
PDMByWellAvg, 11

repCorr, 13
repDistRank, 14