Package ‘DFP’

April 25, 2017

Type Package
Title Gene Selection
Version 1.34.0
Date 2009-07-22
Author R. Alvarez-Gonzalez, D. Glez-Pena, F. Diaz, F. Fdez-Riverola
Maintainer Rodrigo Alvarez-Glez <rodrigo.djv@uvigo.es>
Depends methods, Biobase (>= 2.5.5)
Collate AllClasses.R show-methods.R DFP.R
LazyLoad Yes
Description This package provides a supervised technique able to identify
differentially expressed genes, based on the construction of \textit{Fuzzy Patterns} (FPs). The Fuzzy Patterns are built by means of applying 3 Membership Functions to discretized gene expression values.
License GPL-2
biocViews Microarray, DifferentialExpression
NeedsCompilation no

R topics documented:

DFP-package ................................................. 2
calculateDiscriminantFuzzyPattern .......................... 3
calculateFuzzyPatterns ...................................... 4
calculateMembershipFunctions ......................... 5
discretizeExpressionValues ................................ 6
discriminantFuzzyPattern ................................ 7
ExpressionLevel-class ...................................... 9
HighExpressionLevel-class ................................. 10
LowExpressionLevel-class ................................ 11
MediumExpressionLevel-class .......................... 12
plotDiscriminantFuzzyPattern .......................... 13
plotMembershipFunctions ................................. 14
readCSV .................................................. 15
rmadataset .................................................. 16
show-methods ............................................... 16
showDiscreteValues ...................................... 17
showFuzzyPatterns ........................................ 18
Description

This package provides a supervised technique able to identify differentially expressed genes, based on the construction of Fuzzy Patterns (FPs). The Fuzzy Patterns are built by means of applying 3 Membership Functions to discretized gene expression values.

Details

<table>
<thead>
<tr>
<th>Package:</th>
<th>DFP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Package</td>
</tr>
<tr>
<td>Version:</td>
<td>1.0</td>
</tr>
<tr>
<td>Date:</td>
<td>2008-07-03</td>
</tr>
<tr>
<td>License:</td>
<td>GPL-2</td>
</tr>
</tbody>
</table>

The main functionality of the package is provided by the `discriminantFuzzyPattern` function, which works in a 4-step process:

1. Calculates the Membership Functions. These functions are used in the next step to discretize gene expression data.
2. Discretizes the gene expression data (float values) into ‘Low’, ‘Medium’ or ‘High’ labels.
3. Calculates a Fuzzy Pattern for each category. To do this, a given percentage of the samples belonging to a category must have the same label (‘Low’, ‘Medium’ or ‘High’).
4. Calculates the Discriminant Fuzzy Pattern (DFP) that includes those genes present in two or more FPs with different assigned labels.

Additional data classes: ExpressionSet, AnnotatedDataFrame.

Author(s)

Rodrigo Alvarez-Gonzalez
Daniel Glez-Pena
Fernando Diaz
Florentino Fdez-Riverola
Maintainer: Rodrigo Alvarez-Gonzalez <<rodrigo.djv@uvigo.es>>

References


Examples

###################################################
############ Get sample data ############
calculateDiscriminantFuzzyPattern

Calculates the Discriminant Fuzzy Pattern to select significative genes

Description

Calculates the Discriminant Fuzzy Pattern (DFP) that includes those genes present in two or more FPs with different assigned labels.

Usage

calculateDiscriminantFuzzyPattern(rmadataset, fps)

Arguments

rmadataset ExpressionSet with numeric values containing gene expression values (rows) of samples belonging to different categories (columns). The ExpressionSet also contains an AnnotatedDataFrame with metadata regarding the classes to which each sample belongs.

fps Genes belonging to each Fuzzy Patterns. There are one FP for each class. Includes an attribute ifs with the Impact Factor for each category.

Value

Genes belonging to the final DFP. Includes an attribute ifs with the Impact Factor for each category.

Author(s)

Rodrigo Alvarez-Gonzalez
Daniel Glez-Pena
Fernando Diaz
Florentino Fdez-Riverola
Maintainer: Rodrigo Alvarez-Gonzalez <<rodrigo.djv@uvigo.es>>
References


calculateFuzzyPatterns

*Calculates a Fuzzy Pattern for each category of the samples*

Description

Calculates a Fuzzy Pattern for each category. To do this, a given percentage of the samples belonging to a category must have the same label (‘Low’, ‘Medium’ or ‘High’).

Usage

```
calculateFuzzyPatterns(rmadataset, dvs, piVal = 0.9, overlapping = 2)
```

Arguments

- **rmadataset**: `ExpressionSet` with numeric values containing gene expression values (rows) of samples belonging to different categories (columns). The `ExpressionSet` also contains an `AnnotatedDataFrame` with metadata regarding the classes to which each sample belongs.
- **dvs**: Matrix containing discrete values according to the overlapping parameter after discretizing the gene expression values. Includes an attribute `types` which determines the category of each sample.
- **piVal**: Controls the degree of exigency for selecting a gene as a member of a Fuzzy Pattern. Default value = 0.9. Range[0,1].
- **overlapping**: Modifies the number of membership functions used in the discretization process. Possible values:
  Default value = 2.

Value

Genes belonging to each Fuzzy Patterns. There are one FP for each class. Includes an attribute `ifs` with the Impact Factor for each category.

Author(s)

Rodrigo Alvarez-Gonzalez
Daniel Glez-Pena
Fernando Diaz
Florentino Fdez-Riverola
Maintainer: Rodrigo Alvarez-Gonzalez <<rodrigo.djv@uvigo.es>>
**calculateMembershipFunctions**

*Calculates Membership Functions*

**Description**

Calculates the Membership Functions. These functions are used in the next step (*discretizeExpressionValues*) to discretize gene expression data.

**Usage**

```r
calculateMembershipFunctions(rmadataset, skipFactor = 3)
```

**Arguments**

- **rmadataset**: `ExpressionSet` with numeric values containing gene expression values (rows) of samples belonging to different categories (columns). The `ExpressionSet` also contains an `AnnotatedDataFrame` with metadata regarding the classes to which each sample belongs.
- **skipFactor**: Numeric value to omit odd values (a way of normalization). Higher values imply that less samples of a gene are considered as odd. If `skipFactor=0` do NOT skip. Default value = 3. Range [0,).

**Value**

*Membrahip functions* to determine the discret value (linguistic label) corresponding to a given gene expression level.

**Author(s)**

Rodrigo Alvarez-Gonzalez  
Daniel Glez-Pena  
Fernando Diaz  
Florentino Fdez-Riverola  
Maintainer: Rodrigo Alvarez-Gonzalez <<rodrigo.djv@uvigo.es>>

**References**

discretizeExpressionValues

Function to discretize gene expression data

Description

Discretizes the gene expression data (float values) into ‘Low’, ‘Medium’ or ‘High’ labels.

Usage

discretizeExpressionValues(rmadataset, mfs, zeta = 0.5, overlapping = 2)

Arguments

- **rmadataset**: ExpressionSet with numeric values containing gene expression values (rows) of samples belonging to different categories (columns). The ExpressionSet also contains an AnnotatedDataFrame with metadata regarding the classes to which each sample belongs.
- **mfs**: Membership functions to determine the discret value (linguistic label) corresponding to a given gene expression level.
- **zeta**: Threshold value which controls the activation of a linguistic label ('Low', 'Medium' or 'High'). The lower, the less possibilities of having genes with more than one assigned linguistic label. Default value = 0.5. Range[0,1].
- **overlapping**: Modifies the number of membership functions used in the discretization process. Possible values:
  Default value = 2.

Value

Matrix containing discrete values according to the overlapping parameter after discretizing the gene expression values. Includes an attribute **types** which determines the category of each sample.

Author(s)

Rodrigo Alvarez-Gonzalez
Daniel Glez-Pena
Fernando Diaz
Florentino Fdez-Riverola
Maintainer: Rodrigo Alvarez-Gonzalez <<rodrigo.djv@uvigo.es>>
References


discriminantFuzzyPattern

discriminantFuzzyPattern

Discriminant Fuzzy Pattern to filter genes

Description

discriminantFuzzyPattern discovers significant genes based on the construction of Fuzzy Patterns (FPs). The Fuzzy Patterns are built by means of applying 3 Membership Functions to the gene expression values in the matrix rmadataset.

Usage

discriminantFuzzyPattern(rmadataset, skipFactor = 3, zeta = 0.5, overlapping = 2, piVal = 0.9)

Arguments

rmadataset ExpressionSet with numeric values containing gene expression values (rows) of samples belonging to different categories (columns).
The ExpressionSet also contains an AnnotatedDataFrame with metadata regarding the classes to which each sample belongs.

skipFactor Numeric value to omit odd values (a way of normalization).
Higher values imply that less samples of a gene are considered as odd. If skipFactor=0 do NOT skip.
Default value = 3. Range[0,).

zeta Threshold value which controls the activation of a linguistic label (‘Low’, ‘Medium’ or ‘High’).
The lower, the less possibilities of having genes with more than one assigned linguistic label.
Default value = 0.5. Range[0,1].

overlapping Modifies the number of membership functions used in the discretization process.
Possible values:
Default value = 2.

piVal Controls the degree of exigency for selecting a gene as a member of a Fuzzy Pattern.
Default value = 0.9. Range[0,1].
**Details**

The `discriminantFuzzyPattern` function works in a 4-step process:

1. Calculates the *Membership Functions*. These functions are used in the next step to discretize gene expression data.
2. Discretizes the gene expression data (float values) into ‘Low’, ‘Medium’ or ‘High’ labels.
3. Calculates a *Fuzzy Pattern* for each category. To do this, a given percentage of the samples belonging to a category must have the same label (‘Low’, ‘Medium’ or ‘High’).
4. Calculates the *Discriminant Fuzzy Pattern* (DFP) that includes those genes present in two or more FPs with different assigned labels.

**Value**

- `membership.functions`
  
  *Membership functions* to determine the discret value corresponding to a given gene expression level.

- `discrete.values`
  
  Discrete values according to the overlapping parameter after discretizing the gene expression values. Includes an attribute `types` which determines the category of each sample.

- `fuzzy.patterns`
  
  Genes belonging to each *Fuzzy Patterns*. There are one FP for each class. Includes an attribute `ifs` with the *Impact Factor* for each category.

- `discriminant.fuzzy.pattern`
  
  Genes belonging to the final DFP. Includes an attribute `ifs` with the *Impact Factor* for each category.

- `params`
  
  The parameters used to tune the algorithm (as arguments in the function).

**Author(s)**

Rodrigo Alvarez-Gonzalez  
Daniel Glez-Pena  
Fernando Diaz  
Florentino Fdez-Riverola  
Maintainer: Rodrigo Alvarez-Gonzalez <<rodrigo.djv@uvigo.es>>

**References**


**Examples**

```
#########################################  
############ Get sample data ############  
#########################################  
library(DFP)  
data(rmadataset)
```

```
# Filters the most representative genes #  
```
ExpressionLevel-class

res <- discriminantFuzzyPattern(rmadataset)
summary(res)

ExpressionLevel-class  Class "ExpressionLevel"

Description
A virtual class which represents a generic Membership Function.

Objects from the Class
A virtual Class: No objects may be created from it.

Slots
center: Object of class "numeric". Represents the peak point in the function curve.
width: Object of class "numeric". Represents the length of values lower than 1 and greater than 0 in the function curve.

Methods
show signature(object = "ExpressionLevel"): Prints the ExpressionLevel subclass of the object.
setValues signature(object = "ExpressionLevel", values = "numeric"): Generic function to be implemented in the subclasses.
computeMembership signature(object = "ExpressionLevel", x = "numeric"): Generic function to be implemented in the subclasses.

Author(s)
Rodrigo Alvarez-Gonzalez
Daniel Glez-Pena
Fernando Diaz
Florentino Fdez-Riverola
Maintainer: Rodrigo Alvarez-Gonzalez <<rodrigo.djv@uvigo.es>>

References

Examples
showClass("ExpressionLevel")
HighExpressionLevel-class

Class "HighExpressionLevel"

Description

A class which represents a Membership Function to determine the membership of a numeric value to the ‘High’ discrete label. The result depends on the ‘center’ and ‘width’ values.

Objects from the Class

Objects can be created by calls of the form `new("HighExpressionLevel")`.

Slots

center: Object of class "numeric". Represents the peak point in the function curve.

width: Object of class "numeric". Represents the length of values lower than 1 and greater than 0 in the function curve.

Extends

Class "ExpressionLevel", directly.

Methods

`setValues` signature(object = "HighExpressionLevel", values = "numeric"): Establishes the ‘center’ and ‘width’ slots of the object, given a vector of numeric values.

`computeMembership` signature(object = "HighExpressionLevel", x = "numeric"): Returns a value in the [0,1] interval, which represents the membership to the ‘High’ discrete label.

Author(s)

Rodrigo Alvarez-Gonzalez
Daniel Glez-Pena
Fernando Diaz
Florentino Fdez-Riverola
Maintainer: Rodrigo Alvarez-Gonzalez <rodrigo.djv@uvigo.es>

References


Examples

`showClass("HighExpressionLevel")`
Description

A class which represents a Membership Function to determine the membership of a numeric value to the 'Low' discrete label. The result depends on the 'center' and 'width' values.

Objects from the Class

Objects can be created by calls of the form new("LowExpressionLevel").

Slots

center: Object of class "numeric". Represents the peak point in the function curve.

width: Object of class "numeric". Represents the length of values lower than 1 and greater than 0 in the function curve.

Extends

Class "ExpressionLevel", directly.

Methods

setValues signature(object = "LowExpressionLevel", values = "numeric"): Establishes the 'center' and 'width' slots of the object, given a vector of numeric values.

computeMembership signature(object = "LowExpressionLevel", x = "numeric"): Returns a value in the [0,1] interval, which represents the membership to the 'Low' discrete label.

Author(s)

Rodrigo Alvarez-Gonzalez
Daniel Glez-Pena
Fernando Diaz
Florentino Fdez-Riverola
Maintainer: Rodrigo Alvarez-Gonzalez <<rodrigo.djv@uvigo.es>>

References


Examples

showClass("LowExpressionLevel")
MediumExpressionLevel-class

Class "MediumExpressionLevel"

Description

A class which represents a Membership Function to determine the membership of a numeric value to the ‘Medium’ discrete label. The result depends on the ‘center’ and ‘width’ values.

Objects from the Class

Objects can be created by calls of the form `new("MediumExpressionLevel")`.

Slots

- center: Object of class "numeric". Represents the peak point in the function curve.
- width: Object of class "numeric". Represents the length of values lower than 1 and greater than 0 in the function curve.

Extends

Class "ExpressionLevel", directly.

Methods

- `setValues` signature(object = "MediumExpressionLevel", values = "numeric"): Establishes the ‘center’ and ‘width’ slots of the object, given a vector of numeric values.
- `computeMembership` signature(object = "MediumExpressionLevel", x = "numeric"): Returns a value in the [0,1] interval, which represents the membership to the ‘Medium’ discrete label.

Author(s)

Rodrigo Alvarez-Gonzalez
Daniel Glez-Pena
Fernando Diaz
Florentino Fdez-Riverola
Maintainer: Rodrigo Alvarez-Gonzalez <<rodrigo.djv@uvigo.es>>

References


Examples

`showClass("MediumExpressionLevel")`
**plotDiscriminantFuzzyPattern**

*Plots the Discriminant Fuzzy Pattern of the relevant genes*

**Description**

This function plots the *Discriminant Fuzzy Pattern* of the relevant genes (in rows) for the sample classes (in columns), as well as the impact factor which determines if a gene belongs to a *Fuzzy Pattern* in a class (if its value is higher than the *piVal*).

The relevant genes are those which are present in almost two different *Fuzzy Patterns* with different linguistic labels.

The plotting is made in both graphical and text mode.

**Usage**

```r
plotDiscriminantFuzzyPattern(dfp, overlapping = 2)
```

**Arguments**

- `dfp` A matrix with the fuzzy patterns and impact factors for the relevant genes.
- `overlapping` Modifies the number of membership functions used in the discretization process. Possible values:

  Default value = 2.

**Value**

A matrix with the discriminant genes in rows, along with the *Fuzzy Pattern* for each class (in columns).

This object contains an attribute (*ifs*) which stores the *Impact Factors* used to determine if a gene belongs to a *Fuzzy Pattern* in a class (if the value is higher than the *piVal*).

**Author(s)**

Rodrigo Alvarez-Gonzalez  
Daniel Glez-Pena  
Fernando Diaz  
Florentino Fdez-Riverola  
Maintainer: Rodrigo Alvarez-Gonzalez <<rodrigo.djv@uvigo.es>>

**References**

**plotMembershipFunctions**

*Plots the Membership Functions (Low, Medium, High) used to discretize gene expression values*

**Description**

Each gene has 3 *Membership Functions* (‘Low’, ‘Medium’ and ‘High’) which can be plotted as curves in graphical mode.

In the text mode a membership function is represented with its *center* and *width*.

This function receives one or more gene names and plots the results in both graphical and text mode.

If a set of genes containing more than 36 elements is provided, only the text mode is available.

**Usage**

```r
plotMembershipFunctions(rmadataset, mfs, genes)
```

**Arguments**

- `rmadataset` An `ExpressionSet` object with `AnnotatedDataFrame` metadata.
- `mfs` A list of 3 `ExpressionLevel` objects (‘Low’, ‘Medium’ and ‘High’) for each gene (a list of lists).
- `genes` The set of genes to plot (a vector).

**Value**

A dataframe with the values of the membership functions (‘Low’, ‘Medium’ and ‘High’) for each gene (in rows) received as a parameter.

**Author(s)**

Rodrigo Alvarez-Gonzalez  
Daniel Glez-Pena  
Fernando Diaz  
Florentino Fdez-Riverola  
Maintainer: Rodrigo Alvarez-Gonzalez &lt;rodrigo.djv@uvigo.es&gt;

**References**

readCSV

Creates an ExpressionSet with an AnnotatedDataFrame from CSV files

Description

This function creates an ExpressionSet with an AnnotatedDataFrame. To do this, it requires two CSV files in a predefined format:

1. ‘exprsData’ with the expression values of genes (in rows) of different samples (in columns).
2. ‘pData’ with the samples (in columns) and the metadata ‘class’ (the most important for the algorithm discriminantFuzzyPattern), ‘age’ and ‘sex’.

Usage

readCSV(fileExprs, filePhenodata)

Arguments

fileExprs The path to the exprsData file.
filePhenodata The path to the pData file.

Value

An ExpressionSet object with an AnnotatedDataFrame storing ‘class’, ‘age’ and ‘sex’ information.

Author(s)

Rodrigo Alvarez-Gonzalez
Daniel Glez-Pena
Fernando Diaz
Florentino Fdez-Riverola
Maintainer: Rodrigo Alvarez-Gonzalez <<rodrigo.djv@uvigo.es>>

References


Examples

dataDir <- system.file("extdata", package="DFP"); dataDir
texprs <- file.path(dataDir, "exprsData.csv"); fileExprs
tphenodata <- file.path(dataDir, "pData.csv"); filePhenodata
rmadataset <- readCSV(fileExprs, filePhenodata); rmadataset
pData(phenoData(rmadataset))
texprs(rmadataset)[1:10,1:5]
rmadataset  A sample ExpressionSet object

Description

This ExpressionSet object includes an AnnotatedDataFrame with metadata about ‘Disease type’ (the most important for the algorithm), ‘Patient age’ and ‘Patient gender’.

This data set gives the expression values of 500 genes in 35 samples.

Usage

data(rmadataset)

Format

<table>
<thead>
<tr>
<th>ExpressionSet</th>
<th>str(pData(phenodata(rmadataset)))</th>
</tr>
</thead>
<tbody>
<tr>
<td>AnnotatedDataFrame</td>
<td>str(exprs(rmadataset))</td>
</tr>
</tbody>
</table>

Author(s)

Rodrigo Alvarez-Gonzalez  
Daniel Glez-Pena  
Fernando Diaz  
Florentino Fdez-Riverola  
Maintainer: Rodrigo Alvarez-Gonzalez <<rodrigo.djv@uvigo.es>>

References


Examples

data(rmadataset)  
featureNames(rmadataset)[1:20]  
sampleNames(rmadataset)  
varLabels(rmadataset)  
pData(phenodata(rmadataset))  
exprs(rmadataset)[1:10,1:5]

show-methods  Prints the slots (attributes) of an ExpressionLevel object

Description

Prints the slots (center and width) of an ExpressionLevel object.
**showDiscreteValues**

**Methods**

```object = "ExpressionLevel"` See "ExpressionLevel".`

---

**showDiscreteValues**  *Prints the labels to which the algorithm converts the gene expression values*

**Description**

In an intermediate step, the algorithm `discriminantFuzzyPattern` converts the gene expression values into discrete labels (combining 'Low', 'Medium' and 'High', depending on the value of the param 'overlapping'). This function permits printing these labels, specifying a set of genes (a vector) and/or classes of samples.

**Usage**

```showDiscreteValues(dvs, genes, classes)```

**Arguments**

- **dvs**: A matrix with discrete labels for a set of genes (in rows) of several samples (in columns).
- **genes** [optional]: The set of genes to plot.
- **classes** [optional]: A set of classes to which the samples belong. It must be one of the classes stored in the phenoData of the original `ExpressionSet` object.

**Value**

A subset of the matrix `dvs` determined by the restrictions (genes and/or classes).

**Author(s)**

Rodrigo Alvarez-Gonzalez  
Daniel Glez-Pena  
Fernando Diaz  
Florentino Fdez-Riverola  
Maintainer: Rodrigo Alvarez-Gonzalez <<rodrigo.djv@uvigo.es>>

**References**

showFuzzyPatterns  
Plots the Fuzzy Patterns corresponding to a class

Description
This function prints (in text mode) the Fuzzy Patterns (discrete labels) calculated for a single class of samples.

Usage
showFuzzyPatterns(fps, class)

Arguments
- fps: A matrix with the Fuzzy Patterns (discrete labels) for all the samples and genes.
- class: A class to which the samples belong. It must be one of the classes stored in the phenoData of the original ExpressionSet rmadataset object.

Value
A vector of Fuzzy Patterns (discrete labels) for a single class of samples, with the genes associated.

Author(s)
Rodrigo Alvarez-Gonzalez
Daniel Glez-Pena
Fernando Diaz
Florentino Fdez-Riverola
Maintainer: Rodrigo Alvarez-Gonzalez <<rodrigo.djv@uvigo.es>>

References
Index

*Topic classes
  ExpressionLevel-class, 9
  HighExpressionLevel-class, 10
  LowExpressionLevel-class, 11
  MediumExpressionLevel-class, 12

*Topic datasets
  rmadataset, 16

*Topic manip
  calculateDiscriminantFuzzyPattern, 3
  calculateFuzzyPatterns, 4
  calculateMembershipFunctions, 5
  discretizeExpressionValues, 6
  discriminantFuzzyPattern, 7
  plotDiscriminantFuzzyPattern, 13
  plotMembershipFunctions, 14
  readCSV, 15
  showDiscreteValues, 17
  showFuzzyPatterns, 18

*Topic methods
  show-methods, 16

*Topic package
  DFP-package, 2

AnnotatedDataFrame, 2–7, 14–16
  calculateDiscriminantFuzzyPattern, 3
  calculateFuzzyPatterns, 4
  calculateMembershipFunctions, 5
  computeMembership,HighExpressionLevel-method
    (HighExpressionLevel-class), 10
  computeMembership,LowExpressionLevel-method
    (LowExpressionLevel-class), 11
  computeMembership,MediumExpressionLevel-method
    (MediumExpressionLevel-class), 12

  DFP (DFP-package), 2
  DFP-package, 2
  discretizeExpressionValues, 5, 6
  discriminantFuzzyPattern, 2, 7, 8, 15, 17

  ExpressionLevel, 10–12, 16, 17
  ExpressionLevel-class, 9

ExpressionSet, 2–7, 14–18
  HighExpressionLevel-class, 10
  LowExpressionLevel-class, 11
  MediumExpressionLevel-class, 12

plotDiscriminantFuzzyPattern, 13
plotMembershipFunctions, 14
readCSV, 15
rmadataset, 16
setValues,HighExpressionLevel-method
  (HighExpressionLevel-class), 10
setValues,LowExpressionLevel-method
  (LowExpressionLevel-class), 11
setValues,MediumExpressionLevel-method
  (MediumExpressionLevel-class), 12
show,ExpressionLevel-method
  (ExpressionLevel-class), 9
show-methods, 16
showDiscreteValues, 17
showFuzzyPatterns, 18