

# Package ‘DFP’

June 23, 2022

**Type** Package

**Title** Gene Selection

**Version** 1.55.0

**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 `\emph{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

**git\_url** <https://git.bioconductor.org/packages/DFP>

**git\_branch** master

**git\_last\_commit** 8cf1240

**git\_last\_commit\_date** 2022-04-26

**Date/Publication** 2022-06-23

## R topics documented:

DFP-package . . . . .	2
calculateDiscriminantFuzzyPattern . . . . .	3
calculateFuzzyPatterns . . . . .	4
calculateMembershipFunctions . . . . .	5
discretizeExpressionValues . . . . .	6
discriminantFuzzyPattern . . . . .	8

ExpressionLevel-class . . . . .	10
HighExpressionLevel-class . . . . .	11
LowExpressionLevel-class . . . . .	12
MediumExpressionLevel-class . . . . .	13
plotDiscriminantFuzzyPattern . . . . .	14
plotMembershipFunctions . . . . .	15
readCSV . . . . .	16
rdataset . . . . .	17
show-methods . . . . .	18
showDiscreteValues . . . . .	18
showFuzzyPatterns . . . . .	19

<b>Index</b>	<b>20</b>
--------------	-----------

---

DFP-package	<i>DFP Package Overview</i>
-------------	-----------------------------

---

## 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

Package: DFP  
 Type: Package  
 Version: 1.0  
 Date: 2008-07-03  
 License: GPL-2

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**

F. Diaz; F. Fdez-Riverola; D. Glez-Pena; J.M. Corchado. Using Fuzzy Patterns for Gene Selection and Data Reduction on Microarray Data. 7th International Conference on Intelligent Data Engineering and Automated Learning: IDEAL 2006, (2006) pp. 1095-1102

**Examples**

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

#####
# Filter the most representative genes #
#####
res <- discriminantFuzzyPattern(rmadataset)

#####
##### Different result displays #####
#####
plotMembershipFunctions(rmadataset, res$membership.functions, featureNames(rmadataset)[1:2])
showDiscreteValues(res$discrete.values, featureNames(rmadataset)[1:10], c("healthy", "AML-inv"))
showFuzzyPatterns(res$fuzzy.patterns, "healthy")[21:50]
plotDiscriminantFuzzyPattern(res$discriminant.fuzzy.pattern)
```

---

calculateDiscriminantFuzzyPattern

*Calculates the Discriminant Fuzzy Pattern to select significant 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**

rmdatASET	<a href="#">ExpressionSet</a> with numeric values containing gene expression values (rows) of samples belonging to different categories (columns). The <a href="#">ExpressionSet</a> also contains an <a href="#">AnnotatedDataFrame</a> with metadata regarding the classes to which each sample belongs.
fps	Genes belonging to each <i>Fuzzy Patterns</i> . There are one FP for each class. Includes an attribute <i>ifs</i> with the <i>Impact Factor</i> 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**

F. Diaz; F. Fdez-Riverola; D. Glez-Pena; J.M. Corchado. Using Fuzzy Patterns for Gene Selection and Data Reduction on Microarray Data. 7th International Conference on Intelligent Data Engineering and Automated Learning: IDEAL 2006, (2006) pp. 1095-1102

---

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(rmdatASET, dvs, piVal = 0.9, overlapping = 2)
```

**Arguments**

rmdatASET	<a href="#">ExpressionSet</a> with numeric values containing gene expression values (rows) of samples belonging to different categories (columns). The <a href="#">ExpressionSet</a> also contains an <a href="#">AnnotatedDataFrame</a> 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 <i>types</i> which determines the category of each sample.
piVal	Controls the degree of exigency for selecting a gene as a member of a <i>Fuzzy Pattern</i> . Default value = 0.9. Range[0, 1].
overlapping	Modifies the number of membership functions used in the discretization process. Possible values: <ol style="list-style-type: none"><li>1. 'Low', 'Medium', 'High'.</li><li>2. 'Low', 'Low-Medium', 'Medium', 'Medium-High', 'High'.</li><li>3. 'Low', 'Low-Medium', 'Low-Medium-High', 'Medium', 'Medium-High', 'High'.</li></ol> 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>>

**References**

F. Diaz; F. Fdez-Riverola; D. Glez-Pena; J.M. Corchado. Using Fuzzy Patterns for Gene Selection and Data Reduction on Microarray Data. 7th International Conference on Intelligent Data Engineering and Automated Learning: IDEAL 2006, (2006) pp. 1095-1102

---

calculateMembershipFunctions

*Calculates Membership Functions*

---

**Description**

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

**Usage**

```
calculateMembershipFunctions(rmadataSet, skipFactor = 3)
```

**Arguments**

- `rmdatASET` [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**

*Membership 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**

F. Diaz; F. Fdez-Riverola; D. Glez-Pena; J.M. Corchado. Using Fuzzy Patterns for Gene Selection and Data Reduction on Microarray Data. 7th International Conference on Intelligent Data Engineering and Automated Learning: IDEAL 2006, (2006) pp. 1095-1102

---

discretizeExpressionValues

*Function to discretize gene expression data*

---

**Description**

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

**Usage**

```
discretizeExpressionValues(rmdatASET, mfs, zeta = 0.5, overlapping = 2)
```

**Arguments**

rmadataset	<a href="#">ExpressionSet</a> with numeric values containing gene expression values (rows) of samples belonging to different categories (columns). The <a href="#">ExpressionSet</a> also contains an <a href="#">AnnotatedDataFrame</a> with metadata regarding the classes to which each sample belongs.
mfs	<i>Membership functions</i> 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: <ol style="list-style-type: none"> <li>1. 'Low', 'Medium', 'High'.</li> <li>2. 'Low', 'Low-Medium', 'Medium', 'Medium-High', 'High'.</li> <li>3. 'Low', 'Low-Medium', 'Low-Medium-High', 'Medium', 'Medium-High', 'High'.</li> </ol> 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**

F. Diaz; F. Fdez-Riverola; D. Glez-Pena; J.M. Corchado. Using Fuzzy Patterns for Gene Selection and Data Reduction on Microarray Data. 7th International Conference on Intelligent Data Engineering and Automated Learning: IDEAL 2006, (2006) pp. 1095-1102

---

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 *rmdatASET*.

### Usage

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

### Arguments

rmdatASET	<a href="#">ExpressionSet</a> with numeric values containing gene expression values (rows) of samples belonging to different categories (columns). The <a href="#">ExpressionSet</a> also contains an <a href="#">AnnotatedDataFrame</a> 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 <i>skipFactor=0</i> do <b>NOT</b> 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: <ol style="list-style-type: none"> <li>'Low', 'Medium', 'High'.</li> <li>'Low', 'Low-Medium', 'Medium', 'Medium-High', 'High'.</li> <li>'Low', 'Low-Medium', 'Low-Medium-High', 'Medium', 'Medium-High', 'High'.</li> </ol> Default value = 2.
piVal	Controls the degree of exigency for selecting a gene as a member of a <i>Fuzzy Pattern</i> . 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	<i>Membership functions</i> 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 <i>types</i> which determines the category of each sample.
fuzzy.patterns	Genes belonging to each <i>Fuzzy Patterns</i> . There are one FP for each class. Includes an attribute <i>ifs</i> with the <i>Impact Factor</i> for each category.
discriminant.fuzzy.pattern	Genes belonging to the final DFP. Includes an attribute <i>ifs</i> with the <i>Impact Factor</i> 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

F. Diaz; F. Fdez-Riverola; D. Glez-Pena; J.M. Corchado. Using Fuzzy Patterns for Gene Selection and Data Reduction on Microarray Data. 7th International Conference on Intelligent Data Engineering and Automated Learning: IDEAL 2006, (2006) pp. 1095-1102

### Examples

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

#####
# Filters the most representative genes #
#####
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**

F. Diaz; F. Fdez-Riverola; D. Glez-Pena; J.M. Corchado. Using Fuzzy Patterns for Gene Selection and Data Reduction on Microarray Data. 7th International Conference on Intelligent Data Engineering and Automated Learning: IDEAL 2006, (2006) pp. 1095-1102

**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

F. Diaz; F. Fdez-Riverola; D. Glez-Pena; J.M. Corchado. Using Fuzzy Patterns for Gene Selection and Data Reduction on Microarray Data. 7th International Conference on Intelligent Data Engineering and Automated Learning: IDEAL 2006, (2006) pp. 1095-1102

### Examples

```
showClass("HighExpressionLevel")
```

---

LowExpressionLevel-class

*Class "LowExpressionLevel"*

---

### 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

F. Diaz; F. Fdez-Riverola; D. Glez-Pena; J.M. Corchado. Using Fuzzy Patterns for Gene Selection and Data Reduction on Microarray Data. 7th International Conference on Intelligent Data Engineering and Automated Learning: IDEAL 2006, (2006) pp. 1095-1102

### 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

F. Diaz; F. Fdez-Riverola; D. Glez-Pena; J.M. Corchado. Using Fuzzy Patterns for Gene Selection and Data Reduction on Microarray Data. 7th International Conference on Intelligent Data Engineering and Automated Learning: IDEAL 2006, (2006) pp. 1095-1102

### 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

```
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: <ol style="list-style-type: none"><li>1. 'Low', 'Medium', 'High'.</li><li>2. 'Low', 'Low-Medium', 'Medium', 'Medium-High', 'High'.</li><li>3. 'Low', 'Low-Medium', 'Low-Medium-High', 'Medium', 'Medium-High', 'High'.</li></ol> 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

F. Diaz; F. Fdez-Riverola; D. Glez-Pena; J.M. Corchado. Using Fuzzy Patterns for Gene Selection and Data Reduction on Microarray Data. 7th International Conference on Intelligent Data Engineering and Automated Learning: IDEAL 2006, (2006) pp. 1095-1102

---

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

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

## Arguments

rmadataset	An <a href="#">ExpressionSet</a> object with <a href="#">AnnotatedDataFrame</a> metadata.
mfs	A list of 3 <a href="#">ExpressionLevel</a> 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 <<rodrigo.djv@uvigo.es>>

## References

F. Diaz; F. Fdez-Riverola; D. Glez-Pena; J.M. Corchado. Using Fuzzy Patterns for Gene Selection and Data Reduction on Microarray Data. 7th International Conference on Intelligent Data Engineering and Automated Learning: IDEAL 2006, (2006) pp. 1095-1102

---

readCSV	<i>Creates an ExpressionSet with an AnnotatedDataFrame from CSV files</i>
---------	---

---

### 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

F. Diaz; F. Fdez-Riverola; D. Glez-Pena; J.M. Corchado. Using Fuzzy Patterns for Gene Selection and Data Reduction on Microarray Data. 7th International Conference on Intelligent Data Engineering and Automated Learning: IDEAL 2006, (2006) pp. 1095-1102

### Examples

```
dataDir <- system.file("extdata", package="DFP"); dataDir  
fileExprs <- file.path(dataDir, "exprsData.csv"); fileExprs  
filePhenodata <- file.path(dataDir, "pData.csv"); filePhenodata  
rmdatASET <- readCSV(fileExprs, filePhenodata); rmdatASET  
pData(phenoData(rmdatASET))  
exprs(rmdatASET)[1:10,1:5]
```



---

rmdatASET	<i>A sample ExpressionSet object</i>
-----------	--------------------------------------

---

### 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(rmdatASET)
```

### Format

```
ExpressionSet      str(pData(phenoData(rmdatASET)))  
AnnotatedDataFrame str(exprs(rmdatASET))
```

### Author(s)

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

### References

F. Diaz; F. Fdez-Riverola; D. Glez-Pena; J.M. Corchado. Using Fuzzy Patterns for Gene Selection and Data Reduction on Microarray Data. 7th International Conference on Intelligent Data Engineering and Automated Learning: IDEAL 2006, (2006) pp. 1095-1102

### Examples

```
data(rmdatASET)  
featureNames(rmdatASET)[1:20]  
sampleNames(rmdatASET)  
varLabels(rmdatASET)  
pData(phenoData(rmdatASET))  
exprs(rmdatASET)[1:10,1:5]
```

---

show-methods	<i>Prints the slots (attributes) of an ExpressionLevel object</i>
--------------	---

---

**Description**

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

**Methods**

**object** = "ExpressionLevel" See "ExpressionLevel".

---

showDiscreteValues	<i>Prints the labels to which the algorithm converts the gene expression values</i>
--------------------	---

---

**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 <code>ExpressionSet</code> 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

F. Diaz; F. Fdez-Riverola; D. Glez-Pena; J.M. Corchado. Using Fuzzy Patterns for Gene Selection and Data Reduction on Microarray Data. 7th International Conference on Intelligent Data Engineering and Automated Learning: IDEAL 2006, (2006) pp. 1095-1102

---

showFuzzyPatterns      *Plots the Fuzzy Patterns corresponding to a class*

---

## Description

This functions 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 <i>Fuzzy Patterns</i> (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 <a href="#">ExpressionSet</a> <i>rmadataset</i> 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

F. Diaz; F. Fdez-Riverola; D. Glez-Pena; J.M. Corchado. Using Fuzzy Patterns for Gene Selection and Data Reduction on Microarray Data. 7th International Conference on Intelligent Data Engineering and Automated Learning: IDEAL 2006, (2006) pp. 1095-1102

# Index

- \* **classes**
  - ExpressionLevel-class, [10](#)
  - HighExpressionLevel-class, [11](#)
  - LowExpressionLevel-class, [12](#)
  - MediumExpressionLevel-class, [13](#)
- \* **datasets**
  - rmdatataset, [17](#)
- \* **manip**
  - calculateDiscriminantFuzzyPattern, [3](#)
  - calculateFuzzyPatterns, [4](#)
  - calculateMembershipFunctions, [5](#)
  - discretizeExpressionValues, [6](#)
  - discriminantFuzzyPattern, [8](#)
  - plotDiscriminantFuzzyPattern, [14](#)
  - plotMembershipFunctions, [15](#)
  - readCSV, [16](#)
  - showDiscreteValues, [18](#)
  - showFuzzyPatterns, [19](#)
- \* **methods**
  - show-methods, [18](#)
- \* **package**
  - DFP-package, [2](#)
- AnnotatedDataFrame, [2](#), [4](#), [6–8](#), [15–17](#)
- calculateDiscriminantFuzzyPattern, [3](#)
- calculateFuzzyPatterns, [4](#)
- calculateMembershipFunctions, [5](#)
- computeMembership,HighExpressionLevel-method (HighExpressionLevel-class), [11](#)
- computeMembership,LowExpressionLevel-method (LowExpressionLevel-class), [12](#)
- computeMembership,MediumExpressionLevel-method (MediumExpressionLevel-class), [13](#)
- DFP (DFP-package), [2](#)
- DFP-package, [2](#)
- discretizeExpressionValues, [5](#), [6](#)
- discriminantFuzzyPattern, [2](#), [8](#), [8](#), [16](#), [18](#)
- ExpressionLevel, [11–13](#), [18](#)
- ExpressionLevel-class, [10](#)
- ExpressionSet, [2](#), [4](#), [6–8](#), [15–19](#)
- HighExpressionLevel-class, [11](#)
- LowExpressionLevel-class, [12](#)
- MediumExpressionLevel-class, [13](#)
- plotDiscriminantFuzzyPattern, [14](#)
- plotMembershipFunctions, [15](#)
- readCSV, [16](#)
- rmdatataset, [17](#)
- setValues,HighExpressionLevel-method (HighExpressionLevel-class), [11](#)
- setValues,LowExpressionLevel-method (LowExpressionLevel-class), [12](#)
- setValues,MediumExpressionLevel-method (MediumExpressionLevel-class), [13](#)
- show,ExpressionLevel-method (ExpressionLevel-class), [10](#)
- show-methods, [18](#)
- showDiscreteValues, [18](#)
- showFuzzyPatterns, [19](#)