conNormalizer

Normalizing a target data array to a basis array based on their distributions

Description
Normalizing a target data array to a basis array based on their distributions

Usage
conNormalizer(tg, bs)

Arguments

tg a target data array
bs a basis data array
Details

The function maps a target data array to a basis array based on their distributions and the basis data array can be an arbitrary data array or a standard distribution such as normal distribution.

Value

A normalized target data array with the same distribution with the basis data array

Author(s)

Qinxue Meng, Paul Kennedy

Examples

# Normalize DArray1 to DArray3
# load build-in data arrays
data(DArray1)
data(DArray3)

# Capturing distribution information
DBdata1 <- genDistData(DArray1, 500)
DBdata3 <- genDistData(DArray3, 500)

# Using Gaussian function to fit DBdata3
DBdata3 <- gaussianFit(DBdata3)

# Normalize DBdata1 to the Gaussian fitting function of DBdata3
DAArray1 = conNormalizer(DArray1, DArray3)
DAArray3DBdata <- genDistData(DAArray3DBdata, 500)
visDistData(DAArray3DBdata, "P", "DAArray3", "Range", "Probability")

custFit 

fitting a distribution by a customised curve function

Description

fitting a distribution by a customised curve function

Usage

custFit(DBdata, formula)

Arguments

DBdata 
input distribution dataset

formula 
a customised curve function

Details

The function fits distributions by a customised curve fitting and returns a customised curve fitting function.
**defineDist**

**Value**

a customised curve fitting function

**Author(s)**

Qinxue Meng, Paul Kennedy

**See Also**

lm

**Examples**

```r
# Calculating the customised curve fitting function of DArray1's distribution
DBdata1 = custFit(DBdata1)
```

---

**defineDist**

Generating distribution data based on predefined distribution

**Description**

Generating distribution data based on predefined distribution

**Usage**

```r
defineDist(dist)
```

**Arguments**

- `dist`: a predefined distribution
- `min`: the lower bound of data range and default value is 0
- `max`: the upper bound of data range and default value is 1

**Details**

This function generates distribution data based on predefined distribution. The purpose of this function is to enable to normalize arbitrary distributions into a standard distribution.

**Value**

a distribution dataset of the input predefined distribution

**Author(s)**

Qinxue Meng, Paul Kennedy

**See Also**

list()
Examples

```r
# generate distribution data of a normal distribution
DArray5 <- defineDist(Norm(mean=0, sd=1))
```

### Description

Normalizing a target data array to a basis array based on element positions

### Usage

```r
disNormalizer(tg, bs)
```

### Arguments

- **tg**: a target data array
- **bs**: a basis data array

### Details

The function normalize target data array to a basis array based on element positions. This method does not need to do fitting before normalization and works for discrete values as well.

### Value

A normalized target data array with the same distribution with the basis data array

### Author(s)

Qinxue Meng, Paul Kennedy

### Examples

```r
# Calculating the polynomial curve fitting function of DArray1's distribution
DArray1 = disNormalizer(DArray1, DArray3)
```
distrNormalizer

Normalizing a target data array to a standard distribution

Description
Normalizing a target data array to a standard distribution

Usage
distrNormalizer(tg, bs)

Arguments
- tg: a target data array
- bs: a standard distribution created by defineDist(dist)

Details
The function normalize target data array to a standard distribution.

Value
A normalized target data array with the same distribution with the standard distribution

Author(s)
Qinxue Meng, Paul Kennedy

Examples
```r
# Normalize a given data array into a normal distribution
loadData(0)
DBdata1 <- genDistData(DArray1, 500)
DBdata5 <- defineDist(Norm(mean=0, sd=1))
DA1toDA5 <- distrNormalizer(DBdata1, DBdata5)
DA1toDA5DBdata <- genDistData(DA1toDA5, 500)
visDistData(DA1toDA5DBdata, "P", "DA1toDA5", "Range", "Probability")
```

fourierFit
fitting a distribution by fourier curve fitting

Description
fitting a distribution by fourier curve fitting

Usage
fourierFit(DBdata, n)
gaussianFit

Arguments

- **DBdata**: input distribution dataset
- **n**: the degree of the fourier fitting function

Details

The function fits distributions by fourier curve fitting and returns a fourier curve fitting function.

Value

a fourier curve fitting function

Author(s)

Qinxue Meng, Paul Kennedy

See Also

lm

Examples

```r
# Calculating the fourier curve fitting function of DArray1's distribution
DBdata1 = fourierFit(DBdata1, 3)
```
**genDistData**

**Description**
Generating distribution dataset based on input data arrays.

**Usage**
```
genDistData(data, nbin)
```

**Arguments**
- **data**: input data array
- **nbin**: number of bins

**Details**
This function generates distribution dataset based on input data arrays for downstream analysis.

**Value**
a distribution dataset of a given input data array

**Author(s)**
Qinxue Meng, Paul Kennedy

**See Also**
- `list()`

**Examples**
```
# Calculating the gaussian curve fitting function of DArray1's distribution
DBdata1 = gaussianFit(DBdata1)

# load DArray1
DData1 <- genDistData(DArray1, 500)
```
Description

This function loads build-in data array for examples

Usage

loadData(n)

Arguments

n  n-th data array to load; if n = 1, DArray1 is loaded; if n = 2, DArray2 is loaded; if n = 3, DArray3 is loaded; if n = 4, DArray4 is loaded; if n is not 1, 2, 3, 4, all four data arrays are loaded.

Details

This function loads example data arrays for user to test

Value

None

Author(s)

Qinxue Meng

See Also

data()
**polyFit**

fitting a distribution by polynomial curve fitting

**Description**

fitting a distribution by polynomial curve fitting

**Usage**

```r
polyFit(DBdata, n)
```

**Arguments**

- `DBdata`: input distribution dataset
- `n`: the degree of polynomial functions

**Details**

The function fits distributions by polynomial curve fitting and returns a polynomial curve fitting function.

**Value**

a polynomial curve fitting function

**Author(s)**

Qinxue Meng, Paul Kennedy

**See Also**

`lm`

**Examples**

```r
# Calculating the polynomial curve fitting function of DArray1's distribution
DBdata1 = polyFit(DBdata1, 3)
```

---

**visDistData**

Visualising distribution dataset

**Description**

Visualising distribution dataset

**Usage**

```r
visDistData(DBdata, type, t, xl, yl)
```
Arguments

DBdata  a distribution dataset
type    plot by frequency / probability
t       title of plot
xl      description of x-axis
yl      description of y-axis

Details

This function generates distribution data based on predefined distribution. The purpose of this function is to enable to normalize arbitrary distributions into a standard distribution.

Author(s)

Qinxue Meng, Paul Kennedy

Examples

# visualising a distribution data
DBdata1 <- genDistData(DArray1, 500)
visDistData(DBdata1, "F", "DArray1", "Range", "Frequence")
visDistData(DBdata1, "P", "DArray1", "Range", "Probability")
Examples

# visualising fitting results on DArray1's distribution
visFitting(DBdata1, "DArray1", "Range", "Probability")
Index

conNormalizer, 1
custFit, 2
defineDist, 3
disNormalizer, 4
distrNormalizer, 5
fourierFit, 5
gaussianFit, 6
genDistData, 7
loadData, 8
polyFit, 9
visDistData, 9
visFitting, 10