

Package ‘DBNorm’

August 17, 2016

Type Package

Title Distribution-based normalisation

Version 1.0

Date 2016-07-05

Author Qinxue Meng, Paul J. Kennedy

Maintainer Qinxue Meng <Qinxue.Meng@uts.edu.au>

Description Distribution-based normalisation

RoxygenNote 5.0.1

Depends R (>= 3.0),
distr (>= 2.6),
nls2 (>= 0.2),
nlmrt (>= 2016.3.2),

Suggests devtools,
roxygen2

License GPL-3.0

URL <https://github.com/mengqinxue/DBNorm>

R topics documented:

conNormalizer	2
custFit	2
defineDist	3
disNormalizer	4
fourierFit	5
gaussianFit	5
genDistData	6
loadData	7
polyFit	8
visDistData	8
visFitting	9
Index	11

conNormalizer	<i>Normalizing a target data array to a basis array based on their distributions</i>
---------------	--

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
DArray1 = conNormalizer(DArray1, DArray3)
```

custFit	<i>fitting a distribution by a customised curve function</i>
---------	--

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.

Value

a customised curve fitting function

Author(s)

Qinxue Meng, Paul Kennedy

See Also

lm

Examples

```
# 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

```
defineDist(dist, min = 0, max = 1)
```

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

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

disNormalizer	<i>Normalizing a target data array to a basis array based on element positions</i>
---------------	--

Description

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

Usage

```
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

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

fourierFit	<i>fitting a distribution by fourier curve fitting</i>
------------	--

Description

fitting a distribution by fourier curve fitting

Usage

```
fourierFit(DBdata, n)
```

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

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

gaussianFit	<i>fitting a distribution by gaussian curve fitting</i>
-------------	---

Description

fitting a distribution by gaussian curve fitting

Usage

```
gaussianFit(DBdata)
```

Arguments

DBdata input distribution dataset

Details

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

Value

a gaussian curve fitting function

Author(s)

Qinxue Meng, Paul Kennedy

See Also

optim

Examples

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

genDistData	<i>Generating distribution dataset based on input data arrays.</i>
-------------	--

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

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

`loadData`*Loading build-in datasets*

Description

This function loads build-in data array for examples

Usage

```
loadData(n)
```

Arguments

<code>n</code>	n-th data array to load; if <code>n = 1</code> , <code>DArray1</code> is loaded; if <code>n = 2</code> , <code>DArray2</code> is loaded; if <code>n = 3</code> , <code>DArray3</code> is loaded; if <code>n = 4</code> , <code>DArray4</code> is loaded; if <code>n</code> 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()
```

Examples

```
# load DArray1
loadData(1)
# load all data arrays
loadData(5)
```

polyFit	<i>fitting a distribution by polynomial curve fitting</i>
---------	---

Description

fitting a distribution by polynomial curve fitting

Usage

```
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

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

visDistData	<i>Visualising distribution dataset</i>
-------------	---

Description

Visualising distribution dataset

Usage

```
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", "Frequency")
visDistData(DBdata1, "P", "DArray1", "Range", "Probability")
```

visFitting

Visualising fitting results on the input distribution

Description

Visualising fitting results on the input distribution

Usage

```
visFitting(DBdata, t, xl, yl)
```

Arguments

DBdata	a distribution dataset
t	title of plot
xl	description of x-axis
yl	description of y-axis

Details

The function visualizes data distribution and corresponding fitting function so as to provide an intuitive way to evaluate the performance of fitting function.

Author(s)

Qinxue Meng, Paul Kennedy

Examples

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

Index

conNormalizer, [2](#)
custFit, [2](#)

definedDist, [3](#)
disNormalizer, [4](#)

fourierFit, [5](#)

gaussianFit, [5](#)
genDistData, [6](#)

loadData, [7](#)

polyFit, [8](#)

visDistData, [8](#)
visFitting, [9](#)