Package 'plotFPOP'

November 9, 2017

Type Package

Title plot of the functional cost FPOP 1d 2d

Version 0.1

Date 2017-10-22

Author Vincent Runge

Maintainer Vincent Runge <runge.vincent@gmail.com>

Description This package provides plotting functions for dynamic programming changepoint algorithms with penalty using functional pruning. The package contains simple (written in R) functions and algorithms to plot the functional cost in 1d and 2d at consecutive times. The goal is to give an easy and simple access to the shape of the functional cost in order to better understand its updates. Simulation functions are also available in order to simulate segmented data with a gaussian distribution in dimension 1 and 2.

License GPL-2

RoxygenNote 6.0.1

Imports plot3D, cumstats, grDevices, graphics, stats

R topics documented:

dataAll *data in dimension 1 with no pruned last changepoint candidate*

Description

Generating data in dimension 1 with no pruning

Usage

```
dataAll(n, beta = 1)
```

Arguments

n	length of the vector
beta	the penalty coefficient (use the same beta in function fpop1d)

Value

A vector of data of length n

Examples

```
data <- dataAll(n = 10, beta = 2)
fpop1d(data, beta = 2)
```

dataG1 *One-dimensional gaussian segmentation*

Description

Simulating a one-dimenstional segmentation with a given number of segments

Usage

```
dataG1(mean = c(0.5, 0, -0.5, 0), tau = c(0.25, 0.5, 0.75, 1), sigma = 1,
  n = 100)
```

Arguments

mean	Vector of means (mean of consecutive segments)
tau	relative position (between 0 and 1) of the changepoints (the last position is always 1)
sigma	A positive number: the value (unique) of the standard deviation for all the Gaussian laws in the simulation
n	The number of data point to simulate

Value

A vector of data of length n generated by the simulated segmentation

Examples

```
data <- dataG1(sigma = 0.05, n=20)
plot(1:20,data)
```

---

dataG2 *Two-dimensional gaussian segmentation*

Description

Simulating a two-dimenstional segmentation with a given number of segments

Usage

```
dataG2(mean1 = c(0, 1, 1, 0), mean2 = c(0, 0, 1, 1), tau = c(0.25, 0.5,
  0.75, 1), sigma = 1, n = 100)
```

Arguments

mean1	Vector of means for the first dimension
mean2	Vector of means for the second dimension
tau	relative position (between 0 and 1) of the changepoints (the last position is always 1)
sigma	A positive number: the value (unique) of the standard deviation for all the Gaussian laws in the simulation #' @param n The number of data point to simulate
n	The number of data point to simulate

Value

A matrix of data of dimension 2 x n generated by the simulated segmentation

Examples

```
data <- dataG2(sigma = 0.05, n=50)
max = max(data);min = min(data)
plot(1:50,data[1,],ylim = c(min,max))
par(new = TRUE)
plot(1:50,data[2,],ylim = c(min,max),col=2)
par(new = FALSE)
```

fpop1d *Functional cost 1d display*

Description

Displaying the functional cost at consecutive times

Usage

```
fpop1d(data1, beta, nb = 2000, order = FALSE)
```

Arguments

data1	Vector of data
beta	penalty coefficient, positive number
nb	An integer : the number of elements in the vector for the x-axix of the plot
order	a boolean. If true,it displays the labels on the real line from left to right

Value

plots of the functional cost (in the plot window) and the present labels (in the console) at consecutive times. The vertical gray line shows the position of the current added data on the real line

Examples

```
data <- dataG1(sigma = 0.5, n=15)
fpop1d(data,1)
```

fpop2d *Functional cost 2d display at level beta*

Description

Displaying the functional cost at consecutive times

Usage

```
fpop2d(data2, beta, nb = 300, circle = FALSE, select = 0)
```

Arguments

data2	matrix of data of dimension 2 x n
beta	penalty coefficent, positive number
nb	An integer : the number of rows in the matrix used for the plots
circle	A boolean to decide to draw the circles of intersection (green if the region stays, red otherwise)
select	An integer. Choose an index of data point and follow all its associated disks (even if the point is already pruned)

Value

Plots of the functional cost (in the plot window) and the present labels (in the console) at consecutive times

Examples

```
data <- dataG2(mean1 = 0, mean2 = 0, tau = 1, sigma = 0.5, n=10)
fpop2d(data, n = 300, 1, circle = TRUE)
```

Index