Package ‘CEoptim’

February 26, 2015

Type Package
Title Cross-Entropy R Package for Optimization
Version 1.0
Date 2015-02-28
Author Tim Benham and Qibin Duan and Dirk P. Kroese and Benoit Liquet
Maintainer Benoit Liquet <b.liquet@uq.edu.au>
Depends MASS, msm, stats
Description Optimization solver based on the Cross-Entropy method.
License GPL (>= 2.0)
NeedsCompilation no
Repository CRAN
Date/Publication 2015-02-26 16:46:44

R topics documented:

CEoptim-package ........................................ 2
CEoptim .................................................. 2
dirichletrnd ............................................. 5
FitzHugh ................................................... 5
lesmis ..................................................... 6
print ....................................................... 7
yt ......................................................... 8

Index 9
The CEoptim package provides an optimization solver based on the Cross-Entropy method. The main function `ceoptim` can be used to solve multi-extremal optimization problems involving discrete, continuous, and mixed variables. In addition, CEoptim implements linear constraints for continuous optimization.

Details

```
Package: CEoptim
Type: Package
Version: 1.0
Date: 2015-02-28
License: GPL (>=2.0)
LazyLoad: yes
```

Author(s)

Tim Benham, Qibin Duan, Dirk P. Kroese, Benoit Liquet <b.liquet@uq.edu.au>

References


See Also

 CEoptim

Description

CEopt is an optimization function based on the Cross-Entropy method

Usage

`CEoptim(f, f.arg=NULL, maximize=FALSE, continuous=NULL, discrete=NULL, N=100L, rho=0.1, iterThr=1e4L, noImproveThr=5, verbose=FALSE)`
Arguments

\( f \) Function to be optimized. Can have continuous and discrete arguments
\( f\_\text{arg} \) List of additional fixed arguments passed to function \( f \).
\( \text{maximize} \) Logical value determining whether to maximize or minimize the objective function
\( \text{continuous} \) List of arguments for the continuous optimization part consisting of:
  - mean Vector of initial means.
  - sd Vector of initial standard deviations.
  - smoothMean Smoothing parameter for the vector of means. Default value 1 (no smoothing).
  - smoothSd Smoothing parameter for the standard deviations. Default value 1 (no smoothing).
  - sdThr Positive numeric convergence threshold. Check whether the maximum standard deviation is smaller than \( \text{sdThr} \). Default value 0.001.
\( \text{discrete} \) List of arguments for the discrete optimization part, consisting of:
  - categories Integer vector which defines the allowed values of the categorical variables. The \( i \)th categorical variable takes values in the set \( \{0,1,\ldots,\text{categories}(i)-1\} \).
  - probs List of initial probabilities for the categorical variables. Defaults to equal (uniform) probabilities.
  - smoothProb Smoothing parameter for the probabilities of the categorical sampling distribution. Default value 1 (no smoothing).
  - probThr Positive numeric convergence threshold. Check whether all probabilities in the categorical sampling distributions deviate less than \( \text{probThr} \) from either 0 or 1. Default value 0.001.
\( N \) Integer representing the CE sample size.
\( \rho \) Value between 0 and 1 representing the elite proportion.
\( \text{iterThr} \) Termination threshold on the largest number of iterations.
\( \text{noImproveThr} \) Termination threshold on the largest number of iterations during which no improvement of the best function value is found.
\( \text{verbose} \) Logical value set for CE progress output.

Value

\textbf{CEoptim} returns an object of class "CEoptim" which is a list with the following components.

- **optimum** Optimal value of \( f \).
- **optimizer** List of the location of the optimal value, consisting of:
  - continuous Continuous part of the optimizer.
  - discrete Discrete part of the optimizer.
- **termination** List of termination information consisting of:
- **niter** Total number of iterations upon termination.
- **convergence** One of the following statements:
  * Not converged, if the number of iterations reaches `iterThr`;
  * The optimum did not change for `noImproveThr` iterations, if the best value has not improved for `noImproveThr` iterations;
  * Variances converged, otherwise.

- **states** List of intermediate results computed at each iteration. It consists of the iteration number (`iter`), the best overall value (`optimum`) and the worst value of the elite samples, (`gammat`). The means (`mean`) and maximum standard deviations (`maxsd`) of the elite set are also included for continuous cases, and the maximum deviations (`maxprobs`) of the sampling probabilities to either 0 or 1 are included for discrete cases.

- **states.probs** List of categorical sampling probabilities computed at each iteration. Will only be returned for discrete and mixed cases.

**Note**

Although partial parameter passing is allowed outside lists, it is recommended that parameters names are specified in full. Parameters inside lists have to specified completely.

Because `CEoptim` is a random function it is useful to (1) set the seed for the random number generator (for testing purposes), and (2) investigate the quality of the results by repeating the optimization a number of times.

**Author(s)**

Tim Benham, Qibin Duan, Dirk P. Kroese, Benoit Liquet

**References**


**Examples**

```r
## Maximizing the Peaks Function

fun <- function(x){
  return(3*(1-x[1])^2*exp(-x[1]^2 - (x[2]+1)^2)
         -1/3*exp(-(x[1]+1)^2 - x[2]^2))
}

set.seed(1234)

mu0 <- c(-3,-3); sigma0 <- c(10,10)

res <- CEoptim(fun,continuous=list(mean=mu0, sd=sigma0), maximize=TRUE)

print(res$optimum); print(res$optimizer)
```
**dirichletrnd**

*Dirichlet generator*

**Description**

Random generation for the Dirichlet distribution

**Usage**

```r
dirichletrnd(a, n)
```

**Arguments**

- `a` numeric vector for the concentration parameters
- `n` number of observations

**Value**

dirichletrnd generates `n` random observations from a Dirichlet distribution

**Author(s)**

Tim Benham, Qibin Duan, Dirk P. Kroese, Benoit Liquet

**References**


---

**FitzHugh**

*Simulated data from FitzHugh-Nagumo differential equations*

**Description**

The data correspond to the values V(t) of the FitzHugh-Nagumo differential equations

\[
V'(t) = c^*(V(t) - (V(t)^3)/3 + R(t))
\]

\[
R'(t) = -(1/c)^*(V(t) - a + b*R(t))
\]

at times 0, 0.05,...,20.0, with parameters a = 0.2, b = 0.2, c = 3 and initial conditions V(0) = -1, R(0)=1, and adding gaussian noise with standard deviation 0.5.

**Usage**

```r
data(FitzHugh)
```
**Format**

A numeric vector of length 401

**References**


---

**lesmis**

*Network data from Les Miserables*

**Description**

An R implementation of Donald Knuth’s social network graph describing the interaction of characters in Victor Hugo’s novel Les Miserables. Each node represents a character, and edges connect any pair of characters that coappear. The weights of the edges are the number of such coappearances.

**Usage**

```r
data(lesmis)
```

**Format**

Matrix of weights (77x77)

**References**


**print**  
*Print method for the CEoptim object*

### Description

Produce print method for class "CEoptim"

### Usage

```r
## S3 method for class 'CEoptim'
print(x,...)
```

### Arguments

- `x`  
  object of class inheriting from "CEoptim"

- `...`  
  additional arguments: optimizer; optimum; termination; states; states.probs

### Details

Print method for "CEoptim" class, returns by default the main description of the x object including: optimizer; optimum; termination. To get the states and states.probs outputs, one should specify the corresponding argument to "TRUE".

### Author(s)

Tim Benham, Qibin Duan, Dirk P. Kroese, Benoit Liquet

### References


### See Also

CEoptim
Simulated cumulative data from an AR(1) model with regime switching

**Description**

$y_t$ represents the added value of a stock at time $t$, at day $t=1,2,\ldots,300$; that is, the increase (which may be negative) in stock price relative to the price at time $t=0$.

**Usage**

`data(yt)`

**Format**

Numeric vector of length 300

**References**

Index

*Topic datasets
  FitzHugh, 5
  lesmis, 6
  yt, 8
*Topic package
  CEoptim-package, 2

CEoptim, 2, 2, 7
CEoptim-package, 2
dirichletrnd, 5
FitzHugh, 5
lesmis, 6
print, 7
yt, 8