Package ‘LA’

May 11, 2022

Type Package
Title Lioness Algorithm (LA)
Version 2.1
Date 2022-05-06
Description Contains Lioness Algorithm (LA) for finding optimal designs over continuous design space, optimal Latin hypercube designs, and optimal order-of-addition designs. LA is a brand new nature-inspired meta-heuristic optimization algorithm. Detailed methodologies of LA and its implementation on numerical simulations can be found at Hongzhi Wang, Qian Xiao and Abhyuday Mandal (2021) <arXiv:2010.09154>.
License MIT + file LICENSE
Encoding UTF-8
Suggests testthat (>= 3.0.0), knitr, rmarkdown, devtools
Imports Rcpp
LinkingTo Rcpp, RcppArmadillo
RoxygenNote 7.1.2
NeedsCompilation yes
Author Hongzhi Wang [aut, cre], Qian Xiao [aut], Abhyuday Mandal [aut]
Maintainer Hongzhi Wang <hw34508@uga.edu>
Repository CRAN
Date/Publication 2022-05-11 15:30:07 UTC

R topics documented:

LA_LHDC ................................................................. 2
LA_OofAC ............................................................. 3
LA_OptC ............................................................... 4

Index 7
Description

LA_LHDC returns a n by k Latin hypercube design matrix generated by lioness algorithm (LA).

Usage

```r
LA_LHDC(n, k, m = 100L, N = 5000L, OC = "phi_p", p = 15L, q = 1L)
```

Arguments

- **n**: A positive integer, which stands for the number of rows (or run size).
- **k**: A positive integer, which stands for the number of columns (or factor size).
- **m**: A positive integer, which stands for the number of starting lionesses agents. The default is set to be 100.
- **N**: A positive integer, which stands for the number of iterations. The default is set to be 5000. A large value of N will result a high CPU time.
- **OC**: An optimality criterion. The default setting is "phi_p", and it could be one of the following: "phi_p", "AvgAbsCor", "MaxAbsCor", "MaxProCriterion".
- **p**: A positive integer, which is the parameter in the phi_p formula, and p is preferred to be large. The default is set to be 15.
- **q**: The default is set to be 1, and it could be either 1 or 2. If q is 1, dij is the Manhattan (rectangular) distance. If q is 2, dij is the Euclidean distance.
**x** is a vector.

**a** is the starting value of the sequence.

**b** is the ending value of the sequence.

**X** A matrix object. In general, **X** stands for a design matrix.

**i** A positive integer, which stands for the i\(^{th}\) row of **X**.

**j** A positive integer, which stands for the j\(^{th}\) column (or row) of **X**, and it should be within [1, ncol(\(X\))] (or [1, nrow(\(X\))]).

**y** is a vector.

**type** An exchange type. If **type** is "col" (the default setting), two random elements will be exchanged within column j. If **type** is "row", two random elements will be exchanged within row j.

**Value**

If all inputs are logical, then the output will be a n by k LHD.

**Examples**

```r
#generate a 6 by 3 maximin distance LHD with the default setting
try=LA_LHDC(n=6,k=3)
try

#Another example
#generate a 9 by 4 nearly orthogonal LHD
try2=LA_LHDC(n=9,k=4,OC="AvgAbsCor")
try2
```

---

**Description**

`LA_OofAC` returns a n by k D-optimal order-of-addition design matrix generated by lioness algorithm (LA)

**Usage**

```r
LA_OofAC(n, k, m = 100L, N = 5000L)
factorialC(x)
modC(a, b)
rOofAC(n, k)
```
**Arguments**

- **n**: A positive integer, which stands for the number of rows (or run size). Note that the maximum of $n$ cannot be greater than $k$ factorial.
- **k**: A positive integer, which stands for the number of columns (or factor size).
- **m**: A positive integer, which stands for the number of starting lionesses agents. The default is set to be 100.
- **N**: A positive integer, which stands for the number of iterations. The default is set to be 5000. A large value of $N$ will result in a high CPU time.
- **x**: is a positive integer.
- **a**: is a positive integer.
- **b**: is a positive integer.
- **X**: A matrix object. In general, $X$ stands for the design matrix.

**Value**

If all inputs are logical, then the output will be an $n$ by $k$ order-of-addition design.

**Examples**

```r
# generate a D-optimal full OofA with 4 factors.
try = LA_OofAC(n=24, k=4, m=10, N=50)
try

# Another example
# generate a D-optimal 11-run OofA with 4 factors.
try2 = LA_OofAC(n=11, k=4, m=10, N=50)
try2
```

---

**Description**

LA_OptC returns optimal designs with continuous input
Usage

LA_OptC(n, lb, ub, m = 100L, N = 5000L, OC = "D", alpha = 0.1)

D(X)
A(X)
GscoreC(X, x)
rSign(m = 2L)
G(Y)

Arguments

n  A positive integer, which stands for the number of rows (or run size) for a design.
lb  A vector contains the lower bounds of all the input variables. For example, if there are 3 input variables whose lower bounds are 0, 5, and 15, lb should be lb=c(0,5,15).
ub  A vector contains the upper bounds of all the input variables. For example, if there are 3 input variables whose upper bounds are 10, 15, and 25, ub should be ub=c(10,15,25).
m  A positive integer.
N  A positive integer, which stands for the number of iterations. The default is set to be 5000. A large value of N will result a high CPU time.
OC  An optimality criterion. The default setting is "D-optimality". It could be one of the following: "D", "A", and "G", which stands for "D-optimality", "A-optimality", and "G-optimality", respectively.
alpha  A tuning parameter in algorithm for controlling how big the change would be when updating elements in the step of avoiding local optimum. The default is set to be 0.1, which is the recommended value.
X  A matrix object. In general, X stands for the design matrix.
x  is a vector.
Y  A matrix object. In general, Y stands for the design matrix.

Value

If all inputs are logical, then the output will be either a n by length(lb) optimal design. Here, the length(lb) is assumed to be at least 2.

Examples

#Assume in a simple linear regression model, we want to find a D-optimal
#20-run design, where the input variable takes values between 0 and 24.
#In theory, we know the optimal design is the following:
#matrix(c(rep(1,20),rep(0,10),rep(24,10)),ncol=2,nrow=20,byrow=FALSE)
# Use LA with default setting to find the optimal design for above problem.

```r
try = LA_OptC(n=20, lb=c(1,0), ub=c(1,24))
round(try, 8)
```
Index

A (LA_OptC), 4
AvgAbsCorC (LA_LHDC), 2

corC (LA_LHDC), 2
D (LA_OptC), 4
dijC (LA_LHDC), 2
exchangeC (LA_LHDC), 2
factorialC (LA_OofAC), 3
G (LA_OptC), 4
GscoreC (LA_OptC), 4

LA_LHDC, 2
LA_OofAC, 3
LA_OptC, 4

MaxAbsCorC (LA_LHDC), 2
MaxProCriterionC (LA_LHDC), 2
modC (LA_OofAC), 3
MOMC (LA_OofAC), 3

permuC (LA_LHDC), 2
phi_pC (LA_LHDC), 2
PWOC (LA_OofAC), 3

rLHDC (LA_LHDC), 2
rOofAC (LA_OofAC), 3
rSign (LA_OptC), 4

seqC (LA_LHDC), 2
TC (LA_OofAC), 3