Package 'LHD'

September 17, 2020

Type Package

Title Latin Hypercube Designs (LHDs) Algorithms

Version 1.2.0

Author Hongzhi Wang, Qian Xiao, Abhyuday Mandal

Maintainer Hongzhi Wang <hw34508@uga.edu>

Description Contains different algorithms for efficient Latin Hypercube Designs (LHDs) with flexible sizes. Our package is comprehensive since it is capable of generating maximin distance LHDs, maximum projection LHDs, and nearly orthogonal LHDs. Documentation for each algorithm includes useful information and explanation along with corresponding references. This package is particularly useful in the area of Design and Analysis of Experiments (DAE). More specifically, design of computer experiments.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1

Suggests testthat, knitr, rmarkdown, devtools

Imports stats, numbers

VignetteBuilder knitr

NeedsCompilation no

Repository CRAN

Date/Publication 2020-09-17 04:30:03 UTC

R topics documented:

AvgAbsCor 2
dij 3
exchange 4
FastMmLHD 4
GA 5
GLP 7
LaPSO 8

LOO 10
LPWT 11
MaxAbsCor 12
MaxProCriterion 12
MWT 13
OA2LHD 14
OASA 15
OLHD1998 16
phi_p 17
rLHD 18
SA 19
SA2008 20
SLHD 22
WT 24

Index **26**

`AvgAbsCor` *Calculate the Average Absolute Correlation*

Description

`AvgAbsCor` returns the average absolute correlation of a matrix

Usage

```
AvgAbsCor(X)
```

Arguments

`X`	A matrix object. In general, `X` stands for the design matrix.

Value

If all inputs are logical, then the output will be a positive number indicating the average absolute correlation of input matrix. `average absolute correlation = \frac{2 \sum_{i=1}^{k-1} \sum_{j=i+1}^{k}|q_{ij}|}{k(k-1)}`

References

Georgiou, S. D. (2009) Orthogonal Latin hypercube designs from generalized orthogonal designs. *Journal of Statistical Planning and Inference*, **139**, 1530-1540.

Examples

```
#create a toy LHD with 5 rows and 3 columns
toy=rLHD(n=5,k=3);toy

#Calculate the average absolute correlation of toy
AvgAbsCor(X=toy)
```

---

dij *Calculate the Inter-site Distance*

Description

dij returns the inter-site distance of two design points of an LHD

Usage

```
dij(X, i, j, q = 1)
```

Arguments

X	A matrix object. In general, X stands for the 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 row of X. Both i and j should be in [1,nrow(X)] and they should not be equal to each other.
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.

Value

If all inputs are logical, then the output will be a positive number indicating the distance. dij = \left\{ \sum_{k=1}^{m} \vert x_{ik}-x_{jk}\vert ^q \right\}^{1/q}

Examples

```
#create a toy LHD with 5 rows and 3 columns
toy=rLHD(n=5,k=3);toy

#Calculate the inter-site distance of the 2nd and the 4th row of toy (with default q)
dij(X=toy,i=2,j=4)

#Calculate the inter-site distance of the 2nd and the 4th row of toy (with q=2)
dij(X=toy,i=2,j=4,q=2)
```

exchange *Exchange two random elements*

Description

exchange returns a new design matrix after two randomly selected elements are switched from a user-defined column

Usage

```
exchange(X, j)
```

Arguments

X	A matrix object. In general, X stands for the design matrix.
j	A positive integer, which stands for the j^th column of X, and it should be in [1,ncol(X)].

Value

If all inputs are logical, then the output will be a new design matrix after the exchange.

Examples

```
#create a toy LHD with 5 rows and 3 columns
toy=rLHD(n=5,k=3);toy

#Choose the first column of toy and exchange two randomly selected elements.
toy_new=exchange(X=toy,j=1)
toy;toy_new
```

FastMmLHD *Fast Maximin Distance LHD*

Description

FastMmLHD returns a maximin distance LHD matrix generated by the construction method of Wang, L., Xiao, Q., and Xu, H. (2018)

Usage

```
FastMmLHD(n, k, method = "manhattan", t1 = 10)
```

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).
method	A distance measure method. The default setting is "manhattan", and it could be one of the following: "euclidean", "maximum", "manhattan", "canberra", "binary" or "minkowski". Any unambiguous substring can be given.
t1	A tunning parameter, which determines how many repeats will be implemented to search for the optimal design. The default is set to be 10.

Value

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

References

Wang, L., Xiao, Q., and Xu, H. (2018) Optimal maximin L_1-distance Latin hypercube designs based on good lattice point designs. *The Annals of Statistics*, **46**(6B), 3741-3766.

Examples

```
#n by n design when 2n+1 is prime
try=FastMmLHD(8,8)
try
phi_p(try)   #calculate the phi_p of "try".

#n by n design when n+1 is prime
try2=FastMmLHD(12,12)
try2
phi_p(try2)   #calculate the phi_p of "try2".

#n by n-1 design when n is prime
try3=FastMmLHD(7,6)
try3
phi_p(try3)   #calculate the phi_p of "try3".

#General cases
try4=FastMmLHD(24,8)
try4
phi_p(try4)   #calculate the phi_p of "try4".
```

GA *Genetic Algorithm for LHD*

Description

GA returns an LHD matrix generated by genetic algorithm (GA)

Usage

```
GA(
  n,
  k,
  m = 10,
  N = 10,
  pmut = 1/(k - 1),
  OC = "phi_p",
  p = 15,
  q = 1,
  maxtime = 5
)
```

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 even integer, which stands for the population size and it must be an even number. The default is set to be 10.
N	A positive integer, which stands for the number of iterations. The default is set to be 10.
pmut	A probability, which stands for the probability of mutation. The default is set to be 1/(k - 1).
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 prefered 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.
maxtime	A positive number, which indicates the expected maximum CPU time given by user, and it is measured by minutes. For example, maxtime=3.5 indicates the CPU time will be no greater than three and half minutes. The default is set to be 5.

Value

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

References

Liefvendahl, M., and Stocki, R. (2006) A study on algorithms for optimization of Latin hypercubes. *Journal of Statistical Planning and Inference*, **136**, 3231-3247.

Examples

```
#generate a 5 by 3 maximin distance LHD with the default setting
try=GA(n=5,k=3)
try
phi_p(try)   #calculate the phi_p of "try".

#Another example
#generate a 8 by 4 nearly orthogonal LHD
try2=GA(n=8,k=4,OC="AvgAbsCor")
try2
AvgAbsCor(try2)  #calculate the average absolute correlation.
```

GLP *Good Lattice Point Design*

Description

GLP returns a design matrix generated by good lattice point (GLP)

Usage

```
GLP(n, k, h = sample(seq(from = 1, to = (n - 1)), k))
```

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). k must be smaller than n. In GLP designs, k <= the total number of positive integers that are smaller than and coprime to n.
h	A vector whose length is k, with its elements that are smaller than and coprime to n. The default is set to be a random sample of k elements between 1 and n-1.

Value

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

References

Korobov, A.N. (1959) The approximate computation of multiple integrals. *Dokl. Akad. Nauk SSSR*, **124**, 1207-1210.

Examples

```
#generate a 5 by 3 GLP design with the default setting
try=GLP(n=5,k=3)
try

#Another example
#generate a 8 by 4 GLP design with given h vector
try2=GLP(n=8,k=4,h=c(1,3,5,7))
try2
```

LaPSO *Particle Swarm Optimization for LHD*

Description

LaPSO returns an LHD matrix generated by particle swarm optimization algorithm (PSO)

Usage

```
LaPSO(
  n,
  k,
  m = 10,
  N = 10,
  SameNumP = 0,
  SameNumG = n/4,
  p0 = 1/(k - 1),
  OC = "phi_p",
  p = 15,
  q = 1,
  maxtime = 5
)
```

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 particles. The default is set to be 10.
N	A positive integer, which stands for the number of iterations. The default is set to be 10.
SameNumP	A non-negative integer, which stands for how many elements in current column of current particle LHD should be the same as corresponding Personal Best. SameNumP=0, 1, 2, ..., n, and 0 means to skip the "exchange". The default is set to be 0.

SameNumG	A non-negative integer, which stands for how many elements in current column of current particle LHD should be the same as corresponding Global Best. SameNumP=0, 1, 2, ..., n, and 0 means to skip the "exchange". The default is set to be n/4. SameNumP and SameNumG cannot be 0 at the same time.
p0	A probability of exchanging two randomly selected elements in current column of current particle LHD. The default is set to be 1/(k - 1).
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 prefered 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.
maxtime	A positive number, which indicates the expected maximum CPU time given by user, and it is measured by minutes. For example, maxtime=3.5 indicates the CPU time will be no greater than three and half minutes. The default is set to be 5.

Value

If all inputs are logical, then the output will be a n by k LHD. Here are some general suggestions about the parameters:

- SameNumP is approximately n/2 when SameNumG is 0.
- SameNumG is approximately n/4 when SameNumP is 0.
- p0 * (k - 1) = 1 to 2 is often sufficient. So p0 = 1/(k - 1) to 2/(k - 1).

References

Chen, R.-B., Hsieh, D.-N., Hung, Y., and Wang, W. (2013) Optimizing Latin hypercube designs by particle swarm. *Stat. Comput.*, **23**, 663-676.

Examples

```
#generate a 5 by 3 maximin distance LHD with the default setting
try=LaPSO(n=5,k=3)
try
phi_p(try)   #calculate the phi_p of "try".

#Another example
#generate a 8 by 4 nearly orthogonal LHD
try2=LaPSO(n=8,k=4,OC="AvgAbsCor")
try2
AvgAbsCor(try2)  #calculate the average absolute correlation.
```

LOO *Leave-one-out Method*

Description

LOO returns a maximin distance LHD matrix generated by Leave-one-out method, along with its maximum minimum L_1 distance

Usage

```
LOO(n, k, h = sample(seq(from = 1, to = n), k), method = "manhattan")
```

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). k must be smaller than n+1.
h	A vector whose length is k, with its elements that are smaller than and coprime to n+1. The default is set to be a random sample of k elements between 1 and n.
method	A distance measure method. The default setting is "manhattan", and it could be one of the following: "euclidean", "maximum", "manhattan", "canberra", "binary" or "minkowski". Any unambiguous substring can be given.

Value

If all inputs are logical, then the output will be a list, which contains a n by k LHD and its maximum minimum L_1 distance. This is the construction method in section 2.2 of the reference, and it is particularly useful when n+1 is prime and k equals n.

References

Wang, L., Xiao, Q., and Xu, H. (2018) Optimal maximin L_1-distance Latin hypercube designs based on good lattice point designs. *The Annals of Statistics*, **46**(6B), 3741-3766.

Examples

```
#generate a 10 by 10 maximin distance LHD with default setting
try=LOO(n=10,k=10)
try
phi_p(try[[1]])   #calculate the phi_p of "try".

#Another example
#generate a 12 by 12 maximin distance LHD with given h vector
try2=LOO(n=12,k=12,h=1:12)
try2
phi_p(try2[[1]])   #calculate the phi_p of "try2".
```

LPWT *Linear Permuted Williams Transformation*

Description

LPWT returns a maximin distance LHD matrix generated by linear permuted williams transformation, along with its maximum minimum L_1 distance

Usage

```
LPWT(n, k, h = sample(seq(from = 1, to = (n - 1)), k), method = "manhattan")
```

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). k must be smaller than n. In GLP designs, k <= the total number of positive integers that are smaller than and coprime to n.
h	A vector whose length is k, with its elements that are smaller than and coprime to n. The default is set to be a random sample of k elements between 1 and n-1.
method	A distance measure method. The default setting is "manhattan", and it could be one of the following: "euclidean", "maximum", "manhattan", "canberra", "binary" or "minkowski". Any unambiguous substring can be given.

Value

If all inputs are logical, then the output will be a list, which contains a n by k LHD and its maximum minimum L_1 distance. This is the construction method in section 2.1 of the reference, and it is particularly useful when n is prime and k equals n-1.

References

Wang, L., Xiao, Q., and Xu, H. (2018) Optimal maximin L_1-distance Latin hypercube designs based on good lattice point designs. *The Annals of Statistics*, **46**(6B), 3741-3766.

Examples

```
#generate a 11 by 10 maximin distance LHD with default setting
try=LPWT(n=11,k=10)
try
phi_p(try[[1]])   #calculate the phi_p of "try".

#Another example
#generate a 13 by 12 maximin distance LHD with given h vector
try2=LPWT(n=13,k=12,h=1:12)
try2
phi_p(try2[[1]])   #calculate the phi_p of "try2".
```

MaxAbsCor *Calculate the Maximum Absolute Correlation*

Description

MaxAbsCor returns the maximum absolute correlation of a matrix

Usage

```
MaxAbsCor(X)
```

Arguments

X	A matrix object. In general, X stands for the design matrix.

Value

If all inputs are logical, then the output will be a positive number indicating maximum absolute correlation. maximum absolute correlation = max_{ij} |q_{ij}|

References

Georgiou, S. D. (2009) Orthogonal Latin hypercube designs from generalized orthogonal designs. *Journal of Statistical Planning and Inference*, **139**, 1530-1540.

Examples

```
#create a toy LHD with 5 rows and 3 columns
toy=rLHD(n=5,k=3);toy

#Calculate the maximum absolute correlation of toy
MaxAbsCor(X=toy)
```

MaxProCriterion *Calculate the Maximum Projection Criterion*

Description

MaxProCriterion returns the maximum projection criterion of an LHD

Usage

```
MaxProCriterion(X)
```

Arguments

X	A matrix object. In general, X stands for the design matrix.

Value

If all inputs are logical, then the output will be a positive number indicating maximum projection criterion. maximum projection criterion = \Bigg{ \frac{1}{{n \choose 2}} \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \frac{1}{\Pi_{l=1}^{k}(x_{il}-x_{jl})^2} \Bigg}^{1/k}

References

Joseph, V. R., Gul, E., and Ba, S. (2015) Maximum projection designs for computer experiments. *Biometrika*, **102**, 371-380.

Examples

```
#create a toy LHD with 5 rows and 3 columns
toy=rLHD(n=5,k=3);toy

#Calculate the maximum projection criterion of toy
MaxProCriterion(X=toy)
```

MWT *Modified Williams Transformation*

Description

MWT returns a matrix after implementing the modified Williams transformation

Usage

```
MWT(X)
```

Arguments

X	A matrix object. In general, X stands for the design matrix, e.g. an LHD or a GLP design.

Value

If all inputs are logical, then the output will be a matrix whose sizes are the same as input matrix. Note that the output matrix is not an LHD.

References

Wang, L., Xiao, Q., and Xu, H. (2018) Optimal maximin L_1-distance Latin hypercube designs based on good lattice point designs. *The Annals of Statistics*, **46**(6B), 3741-3766.

Examples

```
#create a toy LHD with 5 rows and 3 columns
toy=rLHD(n=5,k=3);toy

#Implementing the modified Williams transformation on toy:
MWT(toy)
```

OA2LHD *Transfer an Orthogonal Array (OA) into an LHD*

Description

OA2LHD transfers an OA into an LHD with corresponding size

Usage

```
OA2LHD(OA)
```

Arguments

OA	An orthogonal array matrix.

Value

If the input is logical, then the output will be an LHD whose sizes are the same as input OA. The assumption is that the elements of OAs must be positive.

References

Tang, B. (1993) Orthogonal-array-based latin hypercubes. *Journal of the Americal Statistical Association*, **88**, 1392-1397.

Examples

```
#create an OA(9,2,3,2)
OA=matrix(c(rep(1:3,each=3),rep(1:3,times=3)),ncol=2,nrow=9,byrow = FALSE);OA

#Transfer the "OA" above into a LHD according to Tang (1993)
tryOA=OA2LHD(OA)
OA;tryOA
```

OASA *Orthogonal-Array-Based Simulated Annealing*

Description

OASA returns an LHD matrix generated by orthogonal-array-based simulated annealing algorithm (OASA)

Usage

```
OASA(
  OA,
  N = 10,
  T0 = 10,
  rate = 0.1,
  Tmin = 1,
  Imax = 5,
  OC = "phi_p",
  p = 15,
  q = 1,
  maxtime = 5
)
```

Arguments

OA	An orthogonal array matrix.
N	A positive integer, which stands for the number of iterations. The default is set to be 10.
T0	A positive number, which stands for the user-defined initial temperature. The default is set to be 10.
rate	A positive percentage, which stands for temperature decrease rate, and it should be in (0,1). For example, rate=0.25 means the temperature decreases by 25% each time. The default is set to be 10%.
Tmin	A positive number, which stands for the minimium temperature allowed. When current temperature becomes smaller or equal to Tmin, the stopping criterion for current loop is met. The default is set to be 1.
Imax	A positive integer, which stands for the maximum perturbations the algorithm will try without improvements before temperature is reduced. For the computation complexity consideration, Imax is recommended to be smaller or equal to 5, which is the default setting.
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 prefered 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.
maxtime	A positive number, which indicates the expected maximum CPU time given by user, and it is measured by minutes. For example, maxtime=3.5 indicates the CPU time will be no greater than three and half minutes. The default is set to be 5.

Value

If all inputs are logical, then the output will be an LHD whose sizes are the same as input OA. The assumption is that the elements of OAs must be positive.

References

Leary, S., Bhaskar, A., and Keane, A. (2003) Optimal orthogonal-array-based latin hypercubes. *Journal of Applied Statistics*, **30**, 585-598.

Examples

```
#create an OA(9,2,3,2)
OA=matrix(c(rep(1:3,each=3),rep(1:3,times=3)),ncol=2,nrow=9,byrow = FALSE);OA

#Use above "OA" as the input OA to generate a 9 by 2 maximin distance LHD
#with the default setting
tryOASA=OASA(OA=OA)
tryOASA
phi_p(tryOASA)   #calculate the phi_p of "tryOASA".

#Another example
#generate a 9 by 2 nearly orthogonal LHD
tryOASA2=OASA(OA=OA,OC="MaxAbsCor")
tryOASA2
MaxAbsCor(tryOASA2)  #calculate the maximum absolute correlation.
```

OLHD1998 *Orthogonal Latin Hypercube Design*

Description

OLHD1998 returns an orthogonal Latin hypercube design generated by the construction method of Ye (1998)

Usage

```
OLHD1998(m)
```

Arguments

m	A positive integer, and it must be greater than or equal to 2.

Value

If all inputs are logical, then the output will be an orthogonal LHD with the following run size: n=2^m+1 and the following factor size: k=2m-2.

References

Ye, K. Q. (1998) Orthogonal column Latin hypercubes and their application in computer experiments. *Journal of the American Statistical Association*, **93**(444), 1430-1439.

Examples

```
#create an orthogonal LHD with m=3. So n=2^m+1=9 and k=2*m-2=4
OLHD1998(m=3)

#create an orthogonal LHD with m=4. So n=2^m+1=17 and k=2*m-2=6
OLHD1998(m=4)
```

---

phi_p *Calculate the phi_p Criterion*

Description

phi_p returns the phi_p criterion of an LHD

Usage

```
phi_p(X, p = 15, q = 1)
```

Arguments

X	A matrix object. In general, X stands for the design matrix.
p	A positive integer, which is the parameter in the phi_p formula, and p is prefered 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.

Value

If all inputs are logical, then the output will be a positive number indicating phi_p. \phi_p = (\sum_{i=1}^{n-1}\sum_{j=i+1}^{n}dij^{-p})^{1/p}, where dij = \left\{ \sum_{k=1}^{m} \vert x_{ik}-x_{jk}\vert ^q \right\}^{1/q}

References

Jin, R., Chen, W., and Sudjianto, A. (2005) An efficient algorithm for constructing optimal design of computer experiments. *Journal of Statistical Planning and Inference*, **134**, 268-287.

Examples

```
#create a toy LHD with 5 rows and 3 columns
toy=rLHD(n=5,k=3);toy

#Calculate the phi_p criterion of toy with default setting
phi_p(X=toy)

#Calculate the phi_p criterion of toy with p=50 and q=2
phi_p(X=toy,p=50,q=2)
```

---

rLHD *Generate a random Latin Hypercube Design (LHD)*

Description

rLHD returns a random Latin hypercube design matrix with user-defined dimention

Usage

```
rLHD(n, k)
```

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).

Value

If all inputs are positive integer, then the output will be a n by k design matrix.

Examples

```
#create a toy LHD with 5 rows and 3 columns
toy=rLHD(n=5,k=3);toy

#another example with 9 rows and 2 columns
rLHD(9,2)
```

SA *Simulated Annealing for LHD*

Description

SA returns an LHD matrix generated by simulated annealing algorithm (SA)

Usage

```
SA(
  n,
  k,
  N = 10,
  T0 = 10,
  rate = 0.1,
  Tmin = 1,
  Imax = 5,
  OC = "phi_p",
  p = 15,
  q = 1,
  maxtime = 5
)
```

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).
N	A positive integer, which stands for the number of iterations. The default is set to be 10.
T0	A positive number, which stands for the user-defined initial temperature. The default is set to be 10.
rate	A positive percentage, which stands for temperature decrease rate, and it should be in (0,1). For example, rate=0.25 means the temperature decreases by 25% each time. The default is set to be 10%.
Tmin	A positive number, which stands for the minimium temperature allowed. When current temperature becomes smaller or equal to Tmin, the stopping criterion for current loop is met. The default is set to be 1.
Imax	A positive integer, which stands for the maximum perturbations the algorithm will try without improvements before temperature is reduced. For the computation complexity consideration, Imax is recommended to be smaller or equal to 5, which is the default setting.
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 prefered 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.
maxtime	A positive number, which indicates the expected maximum CPU time given by user, and it is measured by minutes. For example, maxtime=3.5 indicates the CPU time will be no greater than three and half minutes. The default is set to be 5.

Value

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

References

Morris, M.D., and Mitchell, T.J. (1995) Exploratory designs for computer experiments. *Journal of Statistical Planning and Inference*, **43**, 381-402.

Examples

```
#generate a 5 by 3 maximin distance LHD with the default setting
try=SA(n=5,k=3)
try
phi_p(try)   #calculate the phi_p of "try".

#Another example
#generate a 8 by 4 nearly orthogonal LHD
try2=SA(n=8,k=4,OC="AvgAbsCor")
try2
AvgAbsCor(try2)  #calculate the average absolute correlation.
```

SA2008 *Simulated Annealing for LHD with Multi-objective Optimization Approach*

Description

SA2008 returns an LHD matrix generated by simulated annealing algorithm with multi-objective optimization approach

Usage

```
SA2008(
  n,
  k,
  N = 10,
  T0 = 10,
  rate = 0.1,
  Tmin = 1,
```

```
  Imax = 5,
  OC = "phi_p",
  p = 15,
  q = 1,
  maxtime = 5
)
```

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).
N	A positive integer, which stands for the number of iterations. The default is set to be 10.
T0	A positive number, which stands for the user-defined initial temperature. The default is set to be 10.
rate	A positive percentage, which stands for temperature decrease rate, and it should be in (0,1). For example, rate=0.25 means the temperature decreases by 25% each time. The default is set to be 10%.
Tmin	A positive number, which stands for the minimium temperature allowed. When current temperature becomes smaller or equal to Tmin, the stopping criterion for current loop is met. The default is set to be 1.
Imax	A positive integer, which stands for the maximum perturbations the algorithm will try without improvements before temperature is reduced. For the computation complexity consideration, Imax is recommended to be smaller or equal to 5, which is the default setting.
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 prefered 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.
maxtime	A positive number, which indicates the expected maximum CPU time given by user, and it is measured by minutes. For example, maxtime=3.5 indicates the CPU time will be no greater than three and half minutes. The default is set to be 5.

Value

If all inputs are logical, then the output will be a n by k LHD. This modified simulated annealing algorithm reduces columnwise correlations and maximizes minimum distance between design points simultaneously, with a cost of more computational complexity.

References

Joseph, V.R., and Hung, Y. (2008) Orthogonal-maximin Latin hypercube designs. *Statistica Sinica*, **18**, 171-186.

Examples

```
#generate a 5 by 3 maximin distance LHD with the default setting
try=SA2008(n=5,k=3)
try
phi_p(try)   #calculate the phi_p of "try".

#Another example
#generate a 8 by 4 nearly orthogonal LHD
try2=SA2008(n=8,k=4,OC="AvgAbsCor")
try2
AvgAbsCor(try2)  #calculate the average absolute correlation.
```

SLHD *Sliced Latin Hypercube Design (SLHD)*

Description

SLHD returns an LHD matrix generated by improved two-stage algorithm

Usage

```
SLHD(
  n,
  k,
  t = 1,
  N = 10,
  T0 = 10,
  rate = 0.1,
  Tmin = 1,
  Imax = 3,
  OC = "phi_p",
  p = 15,
  q = 1,
  stage2 = FALSE,
  maxtime = 5
)
```

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).
t	A positive integer, which stands for the number of slices. n/t must be a positive integer, that is, n is divisible by t. t must be smaller or equal to k when n is 9 or larger. t must be smaller than k when n is smaller than 9. Otherwise, the funtion will never stop. The default is set to be 1.
N	A positive integer, which stands for the number of iterations. The default is set to be 10.

T0	A positive number, which stands for the user-defined initial temperature. The default is set to be 10.
rate	A positive percentage, which stands for temperature decrease rate, and it should be in (0,1). For example, rate=0.25 means the temperature decreases by 25% each time. The default is set to be 10%.
Tmin	A positive number, which stands for the minimium temperature allowed. When current temperature becomes smaller or equal to Tmin, the stopping criterion for current loop is met. The default is set to be 1.
Imax	A positive integer, which stands for the maximum perturbations the algorithm will try without improvements before temperature is reduced. For the computation complexity consideration, Imax is recommended to be smaller or equal to 5, which is the default setting.
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 prefered 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.
stage2	A logic input argument, and it could be either FALSE or TRUE. If stage2 is FALSE (the default setting), SLHD will only implement the first stage of the algorithm. If stage2 is TRUE, SLHD will implement the whole algorithm.
maxtime	A positive number, which indicates the expected maximum CPU time given by user, and it is measured by minutes. For example, maxtime=3.5 indicates the CPU time will be no greater than three and half minutes. The default is set to be 5.

Value

If all inputs are logical, then the output will be a n by k LHD. As mentioned from the original paper, the first stage plays a much more important role since it optimizes the slice level. More resources should be given to the first stage if computational budgets are limited. Let m=n/t, where m is the number of rows for each slice, if (m)^k » n, the second stage becomes optional. That is the reason why we add a stage2 parameter to let users decide if the second stage is needed.

References

Ba, S., Myers, W.R., and Brenneman, W.A. (2015) Optimal Sliced Latin Hypercube Designs. *Technometrics*, **57**, 479-487.

Examples

```
#generate a 5 by 3 maximin distance LHD with the default setting
trySLHD1=SLHD(n=5,k=3)
trySLHD1
phi_p(trySLHD1)   #calculate the phi_p of "trySLHD1".

#generate a 5 by 3 maximin distance LHD with stage II
```

```
#let stage2=TRUE and other input are the same as above
trySLHD2=SLHD(n=5,k=3,stage2=TRUE)
trySLHD2
phi_p(trySLHD2)   #calculate the phi_p of "trySLHD2".

#Another example
#generate a 8 by 4 nearly orthogonal LHD
trySLHD3=SLHD(n=8,k=4,OC="AvgAbsCor",stage2=TRUE)
trySLHD3
AvgAbsCor(trySLHD3)  #calculate the average absolute correlation.
```

WT *Williams Transformation*

Description

WT returns a matrix after implementing the Williams transformation

Usage

```
WT(X, baseline = 1)
```

Arguments

X	A matrix object. In general, X stands for the design matrix, e.g. an LHD or a GLP design.
baseline	A integer, which defines the minimum value for each column of the matrix. The default is set to be 1.

Value

If all inputs are logical, then the output will be a matrix whose sizes are the same as input matrix.

References

Williams, E. J. (1949) Experimental designs balanced for the estimation of residual effects of treatments. *Australian Journal of Chemistry*, **2**, 149-168.

Examples

```
#create a toy LHD with 5 rows and 3 columns
toy=rLHD(n=5,k=3);toy
toy2=toy-1;toy2  #make elements of "toy" become 0,1,2,3,4

#Implementing Williams transformation on both toy and toy2:
#The result shows that "WT" function is able to detect the
#elements of input matrix and make adjustments.
WT(toy)
WT(toy2)
```

```
#Change the baseline
WT(toy,baseline=5)
WT(toy,baseline=10)
```

Index

AvgAbsCor, 2

dij, 3

exchange, 4

FastMmLHD, 4

GA, 5
GLP, 7

LaPSO, 8
LOO, 10
LPWT, 11

MaxAbsCor, 12
MaxProCriterion, 12
MWT, 13

OA2LHD, 14
OASA, 15
OLHD1998, 16

phi_p, 17

rLHD, 18

SA, 19
SA2008, 20
SLHD, 22

WT, 24