Package ‘SLHD’

February 19, 2015

Type Package
Title Maximin-Distance (Sliced) Latin Hypercube Designs
Version 2.1-1
Date 2015-01-26
Author Shan Ba
Maintainer Shan Ba <shanbatr@gmail.com>
Description Generate the optimal Latin Hypercube Designs (LHDs) for computer experiments with quantitative factors and the optimal Sliced Latin Hypercube Designs (SL-HDs) for computer experiments with both quantitative and qualitative factors. Details of the algorithm can be found in Ba, S., Brenneman, W. A. and Myers, W. R. (2015), "Optimal Sliced Latin Hypercube Designs," Technometrics. Important function in this package is "maximinSLHD".
License LGPL-2.1
NeedsCompilation yes
Repository CRAN
Date/Publication 2015-01-28 07:54:05

R topics documented:

<table>
<thead>
<tr>
<th>SLHD-package</th>
<th>maximinSLHD</th>
</tr>
</thead>
<tbody>
<tr>
<td>. . . . . . .</td>
<td>. . . . . . .</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>

Index

SLHD-package Sliced Latin hypercube design

Description

Generate the optimal Latin hypercube designs and the optimal sliced Latin hypercube designs for computer experiments.

Details
This package contains functions for generating the optimal Latin hypercube designs (LHDs) when \( t = 1 \) and the optimal sliced Latin hypercube designs (SLHDs) when \( t > 1 \). The maximin distance criterion is adopted as the optimality criterion.

When \( t = 1 \), the maximin-distance LHD is popularly used for designing computer experiments with quantitative factors.

When \( t > 1 \), the maximin-distance SLHD is a special class of LHD which can be partitioned into several slices (blocks), each of which is also a LHD of smaller size. The optimal SLHD structure guarantees the uniformity (space-filling property) in each slice as well as in the whole design. The SLHD is very important in designing computer experiments with quantitative and qualitative factors, where each slice is used as a design for quantitative factors under one of the \( t \) different level combinations of qualitative factors.

Important function in this package is `maximinslhd`.

**Author(s)**

Shan Ba

Maintainer: Shan Ba <shanbatr@gmail.com>

**References**


### maximinSLHD

**Maximin-Distance (Sliced) Latin Hypercube Designs**

**Description**

Generate the optimal Latin hypercube designs when \( t = 1 \) and the optimal sliced Latin hypercube designs when \( t > 1 \) for computer experiments. The maximin distance criterion is adopted as the optimality criterion.

**Usage**

```r
maximinSLHD(t, m, k, power = 15, nstarts = 1, itermax = 100, total_iter = 1e+06)
```
maximinSLHD

Arguments

- **t**: The number of slices. If \( t = 1 \), it leads to a standard space-filling Latin hypercube design.
- **m**: The number of design points (runs) in each slice. When \( t = 1 \), the number of design points for the whole design is just \( m \); when \( t > 1 \), the number of design points for the whole design is \( mt \).
- **k**: The number of input factors (variables).
- **power**: Optional, default is “15”. The power parameter \( r \) in the average reciprocal inter-point distance measure. When \( r \to \infty \), minimizing the average reciprocal inter-point distance measure is equivalent to maximizing the minimum distance among the design points.
- **nstarts**: Optional, default is “1”. The number of random starts.
- **itermax**: Optional, default is “100”. The maximum number of non-improving searches allowed under each temperature. Lower this parameter if you want the algorithm to converge faster.
- **total_iter**: Optional, default is “1e+06”. The maximum total number of iterations. Lower this number if the design is prohibitively large and you want to terminate the algorithm prematurely to report the best design found so far.

Details

This function utilizes a version of the simulated annealing algorithm and several computational shortcuts to efficiently generate the optimal Latin Hypercube Designs (LHDs) and the optimal Sliced Latin Hypercube Designs (SLHDs). The maximin distance criterion is adopted as the optimality criterion. Please refer to Ba et al. (2015) for details of the algorithm.

When \( t = 1 \), the maximin-distance LHD is popularly used for designing computer experiments with quantitative factors.

When \( t > 1 \), the maximin-distance SLHD is a special class of LHD which can be partitioned into several slices (blocks), each of which is also a LHD of smaller size. The optimal SLHD structure ensures the uniformity (space-filling property) in each slice as well as in the whole design. The SLHD is very important in designing computer experiments with quantitative and qualitative factors, where each slice is used as a design for quantitative factors under one of the \( t \) different level combinations of qualitative factors.

Value

The value returned from the function is a list containing the following components:

- **Design**: The optimal design matrix. When \( t = 1 \), it is a maximin-distance LHD; when \( t > 1 \), it is a maximin-distance SLHD with the first column representing the slice number.
- **measure**: The average reciprocal inter-point distance measure.
- **StandDesign**: The optimal design matrix after standardizing each continuous variable into \((0,1)\) scale.
- **temp0**: Initial temperature.
- **time_rec**: Time to complete the search.
Author(s)

Shan Ba<shanbatr@gmail.com>

References


Examples

#Maximin-distance Latin hypercube design
D1<-maximinSLHD(t = 1, m = 10, k = 3)
D1$Design
D1$StandDesign

#Maximin-distance sliced Latin hypercube designs
D2<-maximinSLHD(t = 3, m = 4, k = 2)
D2$Design
D2$StandDesign
Index

*Topic Computer Experiments
  maximinSLHD, 2

*Topic Design of Experiments
  maximinSLHD, 2

*Topic package
  SLHD-package, 1
maximinSLHD, 2, 2
SLHD (SLHD-package), 1
SLHD-package, 1