

Package ‘CompExpDes’

September 18, 2025

Type Package

Title Designs for Computer Experimentations

Version 1.0.9

Maintainer Ashutosh Dalal <ashutosh.dalal197@gmail.com>

Description In computer experiments space-filling designs are having great impact. Most popularly used space-filling designs are Uniform designs (UDs), Latin hypercube designs (LHDs) etc. For further references one can see Mckay (1979) <DOI:10.1080/00401706.1979.10489755> and Fang (1980) <<https://cir.nii.ac.jp/crid/1570291225616774784>>. In this package, we have provided algorithms for generate efficient LHDs and UD. Here, generated LHDs are efficient as they possess lower value of Maxpro measure, Phi_p value and Maximum Absolute Correlation (MAC) value based on the weightage given to each criterion. On the other hand, the produced UD are having good space-filling property as they always attain the lower bound of Discrete Discrepancy measure. Further, some useful functions added in this package for adding more value to this package.

Suggests HadamardR

License GPL (>= 2)

Encoding UTF-8

RoxygenNote 7.3.2

NeedsCompilation no

Author Ashutosh Dalal [aut, cre],
Cini Varghese [aut, ctb],
Rajender Parsad [aut, ctb],
Mohd Harun [aut, ctb]

Repository CRAN

Date/Publication 2025-09-18 19:00:02 UTC

Contents

Best_Model	2
Discrete_Discrepancy	3
MAC	4

Maxpro_Measure	5
max_coincidence_number	5
NOLHDs	6
OLHDs_2F	7
PhipMeasure	8
SLHDs	9
UDesigns_I	9
UDesigns_II	10
UDesigns_III	11
wtLHDs	12
wtLHDs_prime	13

Index	15
--------------	-----------

Best_Model	<i>Find Best Model</i>
------------	------------------------

Description

This function will try to find out a significant model for each combinations based on adjusted R². Then user need to select which model they want to use.

Usage

```
Best_Model(model, data)
```

Arguments

model	Provide a vector that contains all the individual terms present in a full model
data	Provide data in a matrix or data frame format where you want to fit the model

Value

Generate a list of significant models for various combinations of factors.

Author(s)

Ashutosh Dalal, Cini Varghese, Rajender Parsad and Mohd Harun

Examples

```
## Not run:
library(CompExpDes)
# Sample data
data <- data.frame(
  x1 = c(1.0, 1.4, 1.8, 2.2, 2.6, 3.0, 3.4, 3.8, 4.2, 4.6, 5.0, 5.4),
  x2 = c(50, 25, 5, 30, 55, 45, 20, 10, 35, 60, 40, 15),
  x3 = c(2.5, 6.0, 4.0, 1.0, 5.5, 4.5, 3.0, 2.0, 6.5, 3.5, 1.5, 5.0),
  x4 = c(45, 25, 55, 35, 65, 15, 70, 20, 50, 30, 60, 40),
```

```
y = c(0.0795, 0.0118, 0.0109, 0.0991, 0.1266, 0.0717, 0.1319, 0.0900, 0.1739,  
0.1176, 0.1836, 0.1424)  
)  
# List of terms in the polynomial model  
model <- list('x1', 'x2', 'x3', 'x4', 'x1:x2', 'x1:x3', 'x1:x4',  
            'x2:x3', 'x2:x4', 'x3:x4', 'I(x1^2)',  
            'I(x2^2)', 'I(x3^2)', 'I(x4^2)')  
  
Best_Model(model,data)  
  
## End(Not run)
```

Discrete_Discrepancy *Measure of Discrete Discrepancy*

Description

Discrete Discrepancy is a measure of uniformity for any uniform design. Lesser the value of Discrete Discrepancy measure, better is the uniform design.

Usage

```
Discrete_Discrepancy(Design,a,b)
```

Arguments

Design	A matrix
a	Any value $a > b > 0$. By default it is set to 1.
b	Any value $a > b > 0$. By default it is set to 0.5.

Value

The function calculates the value of Discrete Discrepancy measure and its lower bound for a given design.

Author(s)

Ashutosh Dalal, Cini Varghese, Rajender Parsad and Mohd Harun

References

Qin H, Fang KT (2004)<DOI:10.1007/s001840300296> Discrete discrepancy in factorial designs. *Metrika*, 60, 59-72.

Examples

```
library(CompExpDes)
lhd1<-matrix(c(1,5,7,3,4,2,6,2,1,4,5,3,7,6,4,5,6,1,2,3,7),nrow=7,ncol=3,byrow=FALSE)
lhd2<-cbind(lhd1[,3],lhd1[,2],lhd1[,1])
ud<-rbind(lhd1,lhd2)
Discrete_Discrepancy(ud, 1, 0.5)
```

MAC

Maximum Absolute Correlation

Description

Maximum Absolute Correlation (MAC) is the maximum absolute value among off diagonal values of a correlation matrix.

Usage

```
MAC(matrix)
```

Arguments

matrix Input a matrix

Value

It returns a maximum absolute correlation value for a given matrix.

Author(s)

Ashutosh Dalal, Cini Varghese, Rajender Parsad and Mohd Harun

References

Jones, B. and Nachtsheim, C. J. (2011). A class of three-level designs for definitive screening in the presence of second-order effects. *Journal of Quality Technology*, 43(1), 1-15.

Examples

```
library(CompExpDes)
lhd<-matrix(c(1,5,7,3,4,2,6,2,1,4,5,3,7,6,4,5,6,1,2,3,7),nrow=7,ncol=3,byrow=FALSE)
MAC(lhd)
```

Maxpro_Measure	<i>Measure of Maxpro criterion</i>
----------------	------------------------------------

Description

This function calculates Maxpro criterion for a given space-filling design. Lesser the value of it better the design, in the sense that the design has maximum spread in higher dimensional spaces.

Usage

```
Maxpro_Measure(Design)
```

Arguments

Design Provide a design in a matrix format

Value

Provides Maxpro criterion value given by Joseph et al. (2015).

Author(s)

Ashutosh Dalal, Cini Varghese, Rajender Parsad and Mohd Harun

References

Joseph, V.R., Gul, E. and Ba, S. (2015). Maximum projection designs for computer experiments. *Biometrika*, 102 (2), 371-380.

Examples

```
library(CompExpDes)
lhd<-matrix(c(1,5,7,3,4,2,6,2,1,4,5,3,7,6,4,5,6,1,2,3,7),nrow=7,ncol=3,byrow=FALSE)
Maxpro_Measure(lhd)
```

max_coincidence_number

Maximum Coincidence (or Meeting) numbers between rows

Description

Finding out Maximum coincidence (or Meeting) number between unique pair of rows.

Usage

```
max_coincidence_number(matrix)
```

Arguments

matrix Provide any matrix

Value

This function provides the maximum coincidence number between any pair of rows of for a given matrix.

Author(s)

Ashutosh Dalal, Cini Varghese, Rajender Parsad and Mohd Harun

Examples

```
library(CompExpDes)
mat<-matrix(c(1,2,3,3,2,1,4,2,1),nrow=3,byrow=TRUE)
max_coincidence_number(mat)
```

NOLHDs

Nearly Orthogonal Latin Hypercube Designs for Flexible Levels and Factors

Description

This NOLHDs are generated using a new algorithm for a flexible parameter range which possesses a good space-filling property.

Usage

```
NOLHDs(factors, levels)
```

Arguments

factors Number of factors(≥ 2)
 levels Number of levels(\geq factors+3)

Value

Generates NOLHDs along with its parameters and maximum absolute correlation (MAC) value.

Author(s)

Ashutosh Dalal, Cini varghese, Rajender Parsad and Mohd Harun

References

Cioppa, T.M. and Lucas, T.W.: Efficient nearly orthogonal and space-filling latin hypercubes. *Technometrics*, 49(1), 45–55 (2007).

Examples

```
library(CompExpDes)
NOLHDs(2, 5)
```

OLHDs_2F

Two Factor Orthogonal Latin Hypercube Designs

Description

This OLHDs are generated using a new unique algorithm which possesses a good space-filling property.

Usage

```
OLHDs_2F(levels)
```

Arguments

levels Number of levels, $4t-1$, where $t=3,4,\dots$

Value

Generates two factor OLHDs along with its parameters.

Author(s)

Ashutosh Dalal, Cini varghese, Rajender Parsad and Mohd Harun

References

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

Examples

```
library(CompExpDes)
OLHDs_2F(11)
```

PhipMeasure *Phi_p criterion*

Description

For a given design Phi_p criterion (Morris and Mitchell, 1995) is calculated using this function. Lesser the value of Phi_p criterion better the design in terms of space-filling.

Usage

```
PhipMeasure(design,p=15,q=1)
```

Arguments

design	A design matrix is needed
p	Any positive integer. Default value of p = 15.
q	Any positive integer. Default value of q = 1. This implies that we are considering here Euclidean distance.

Value

Generates Phi_p criterion value

Author(s)

Ashutosh Dalal, Cini Varghese, Rajender Parsad and Mohd Harun

References

Morris, M.D. and Mitchell, T.J. (1995). Exploratory designs for computer experiments. Journal of Statistical Planning and Inference, 43, 38-402.

Examples

```
library(CompExpDes)
lhd<-matrix(c(1,5,7,3,4,2,6,2,1,4,5,3,7,6,4,5,6,1,2,3,7),nrow=7,ncol=3,byrow=FALSE)
PhipMeasure(lhd,p=15,q=2)
```

 SLHDs

Sliced Latin Hypercube Designs with Equal Size of Slices

Description

This sliced LHDs are generated using a different new algorithm for a flexible parameter range which possesses a good space-filling property as whole design as well as for the slices.

Usage

SLHDs(slices, factors, levels)

Arguments

slices	Number of slices(≥ 2)
factors	Number of factors(≥ 2)
levels	Number of levels(≥ 3)

Value

Generates a Sliced LHD along with its parameters.

Author(s)

Ashutosh Dalal, Cini varghese, Rajender Parsad and Mohd Harun

References

Qian, P.Z.G. and Wu, C.F.J. (2009). Sliced space-filling designs. *Biometrika*, 96(4), 945–956.

Examples

```
library(CompExpDes)
SLHDs(3, 3, 3)
```

 UDesigns_I

Orthogonal Uniform Designs with Two Factors

Description

This series of UD's can be obtained for a composite number of levels, L with always two factors, F. Further, "Excellent" type UD's are Excellent in space-filling with larger number of runs available for $L \geq 6$. On the other hand, "Good" type UD's are good in space-filling and lesser the number of runs, available for $L \geq 9$. Generated designs are UD's under discrete discrepancy measure, as all designs will attain the lower bound value of discrete discrepancy.

Usage

```
UDesigns_I(levels, type)
```

Arguments

levels	Any composite number ≥ 6 (if "Excellent") or ≥ 9 (if "Good")
type	"Excellent" or "Good"

Details

Type "Excellent" or type "Good" both can exist for a same parameter range. For type "Excellent" it will require more runs than designs generated by type "Good". But type "Excellent" provides designs which are having more spread than type "Good" series designs.

Value

Returns a uniform designs along with number of factors, levels, runs, maximum absolute correlation (MAC) value and discrete discrepancy measure along with its lower bound value.

Author(s)

Ashutosh Dalal, Cini Varghese, Rajender Parsad and Mohd Harun

References

Fang, K.T. (1980). The uniform design: application of number-theoretic methods in experimental design. *Acta Math Appl Sin*, 3, 363-372.

Examples

```
library(CompExpDes)
UDesigns_I(levels=6, type="Excellent")
```

UDesigns_II

Uniform Designs with Multiple Factors with Minimal Runs

Description

In this series, the Uniform Designs (UDs) are high dimensional with lesser number of runs will always attain lower bound of discrete discrepancy. They are available for any even number of factors, $F (\geq 4)$ with $F(F+1)$ levels each.

Usage

```
UDesigns_II(factors)
```

Arguments

factors any even number ≥ 4

Value

Returns a series of high dimensional UDs along with number of factors, levels, runs, MAC value and discrete discrepancy measure along with its lower bound value.

Author(s)

Ashutosh Dalal, Cini Varghese, Rajender Parsad and Mohd Harun

References

Fang, K.T. (1980). The uniform design: application of number-theoretic methods in experimental design. Acta Math Appl Sin, 3, 363-372.

Examples

```
library(CompExpDes)
UDesigns_II(4)
```

UDesigns_III

Nearly Orthogonal Uniform Designs for Two and Four Factors

Description

This function will provide nearly orthogonal uniform designs (UDs) for number of factors, $F = 2$ and 4 but a flexible number of levels, $L \geq 3$.

Usage

```
UDesigns_III(levels, factors)
```

Arguments

levels Number of levels, $L \geq 3$
factors Number of factors 2 or 4

Value

This function will generate 3 Uniform Designs along with the number of levels, factors, runs, MAC value and discrete discrepancy value along with its lower bound value.

Author(s)

Ashutosh Dalal, Cini varghese, Rajender Parsad and Mohd Harun

References

Fang, K.T. (1980). The uniform design: application of number-theoretic methods in experimental design. *Acta Math Appl Sin*, 3, 363-372.

Examples

```
## Not run:
library(CompExpDes)
UDesigns_III(3)

## End(Not run)
```

wtLHDs	<i>Weighted Criteria-Based Latin Hypercube Designs (LHDs) for Any Numbers of Factors (≥ 2)</i>
--------	--

Description

Weighted Criteria-Based Latin Hypercube Designs (LHDs) for Any Numbers of Factors (≥ 2)

Usage

```
wtLHDs(
  levels,
  factors,
  w1,
  w2,
  w3,
  pop_size = 30,
  generations = 100,
  mut_prob = 1/(factors - 1)
)
```

Arguments

levels	Range of levels, L is $F \leq L \leq \text{choose}(F+2, 2)$, where, F is number of factors.
factors	Any number of factors, $F \geq 2$.
w1	Weight of maximum absolute correlation. Between 0 to 1. So that $w1+w2+w3=1$.
w2	Weight of Phi_p criterion. Between 0 to 1. So that $w1+w2+w3=1$.
w3	Weight of Maxpro criterion. Between 0 to 1. So that $w1+w2+w3=1$.
pop_size	Default population size is 30.
generations	Default number of generations is 100.
mut_prob	Mutation probability, by default it is $1/(F-1)$.

Value

Generates Latin hypercube designs for a given number of factor-level combinations.

Examples

```
## Not run:
library(CompExpDes)
wtLHDs(9, 3, 0.5, 0.5, 0)

## End(Not run)
```

wtLHDs_prime	<i>Weighted Criteria-Based Latin Hypercube Designs (LHDs) for Prime Numbers</i>
--------------	---

Description

Weighted Criteria-Based Latin Hypercube Designs (LHDs) for Prime Numbers

Usage

```
wtLHDs_prime(
  levels,
  factors,
  w1,
  w2,
  w3,
  pop_size = 30,
  generations = 100,
  mut_prob = 1/(factors - 1)
)
```

Arguments

levels	Range of levels, L is $F \leq L \leq F^2$, where, F is number of factors.
factors	Any number of prime factors, $F \geq 3$.
w1	Weight of maximum absolute correlation. Between 0 to 1. So that $w1+w2+w3=1$.
w2	Weight of Phi_p criterion. Between 0 to 1. So that $w1+w2+w3=1$.
w3	Weight of Maxpro criterion. Between 0 to 1. So that $w1+w2+w3=1$.
pop_size	Default population size is 30.
generations	Default number of generations is 100.
mut_prob	Mutation probability, by default it is $1/(F-1)$.

Value

Generates Latin hypercube designs for a given number of factor-level combinations.

Examples

```
## Not run:  
library(CompExpDes)  
wtLHDs_prime(9,3,1,0,0)  
  
## End(Not run)
```

Index

- * **Coincidence number**
 - max_coincidence_number, 5
- * **CompExpDes**
 - Discrete_Discrepancy, 3
 - MAC, 4
 - max_coincidence_number, 5
 - Maxpro_Measure, 5
 - NOLHDs, 6
 - OLHDs_2F, 7
 - PhipMeasure, 8
 - SLHDs, 9
 - UDesigns_I, 9
 - UDesigns_II, 10
 - UDesigns_III, 11
- * **Computer Experiments**
 - Discrete_Discrepancy, 3
 - MAC, 4
 - NOLHDs, 6
 - OLHDs_2F, 7
 - SLHDs, 9
 - UDesigns_I, 9
 - UDesigns_II, 10
 - UDesigns_III, 11
- * **Computer experiments**
 - Maxpro_Measure, 5
 - PhipMeasure, 8
- * **Discrete Discrepancy**
 - Discrete_Discrepancy, 3
 - UDesigns_I, 9
 - UDesigns_II, 10
 - UDesigns_III, 11
- * **LHD**
 - Discrete_Discrepancy, 3
 - MAC, 4
 - Maxpro_Measure, 5
 - NOLHDs, 6
 - OLHDs_2F, 7
 - PhipMeasure, 8
 - SLHDs, 9
 - UDesigns_I, 9
 - UDesigns_II, 10
- * **Latin Hypercube Designs**
 - Discrete_Discrepancy, 3
 - MAC, 4
 - Maxpro_Measure, 5
 - NOLHDs, 6
 - OLHDs_2F, 7
 - PhipMeasure, 8
 - SLHDs, 9
 - UDesigns_I, 9
 - UDesigns_II, 10
- * **Meeting number**
 - max_coincidence_number, 5
- * **Multi dimensional uniform designs**
 - UDesigns_II, 10
- * **Orthogonal uniform designs**
 - UDesigns_I, 9
 - UDesigns_III, 11
- * **Space-Filling Designs**
 - Discrete_Discrepancy, 3
 - MAC, 4
 - Maxpro_Measure, 5
 - NOLHDs, 6
 - OLHDs_2F, 7
 - PhipMeasure, 8
 - SLHDs, 9
 - UDesigns_I, 9
 - UDesigns_II, 10
 - UDesigns_III, 11
- * **Space-filling measure**
 - Maxpro_Measure, 5
 - PhipMeasure, 8
- * **Two factor orthogonal uniform designs**
 - UDesigns_I, 9
 - UDesigns_III, 11
- * **UD**
 - UDesigns_I, 9
 - UDesigns_II, 10

- U Designs_III, 11
- * **Uniform designs**
 - U Designs_I, 9
 - U Designs_II, 10
 - U Designs_III, 11
- * **Uniformity measure**
 - Discrete_Discrepancy, 3
- * **maximum absolute correlation**
 - MAC, 4
- * **maxpro**
 - Maxpro_Measure, 5
- * **phip**
 - PhipMeasure, 8
- *
 - Discrete_Discrepancy, 3
 - MAC, 4

- Best_Model, 2

- Discrete_Discrepancy, 3

- MAC, 4
- max_coincidence_number, 5
- Maxpro_Measure, 5

- NOLHDs, 6

- OLHDs_2F, 7

- PhipMeasure, 8

- SLHDs, 9

- U Designs_I, 9
- U Designs_II, 10
- U Designs_III, 11

- wtLHDs, 12
- wtLHDs_prime, 13