Title Maximum One-Factor-at-a-Time Designs
Version 1.0
Imports SLHD, stats

Description Identifying important factors from a large number of potentially important factors of a highly nonlinear and computationally expensive black box model is a difficult problem. Xiao, Joseph, and Ray (2022) doi:10.1080/00401706.2022.2141897 proposed Maximum One-Factor-at-a-Time (MOFAT) designs for doing this. A MOFAT design can be viewed as an improvement to the random one-factor-at-a-time (OFAT) design proposed by Morris (1991) doi:10.1080/00401706.1991.10484804. The improvement is achieved by exploiting the connection between Morris screening designs and Monte Carlo-based Sobol' designs, and optimizing the design using a space-filling criterion. This work is supported by a U.S. National Science Foundation (NSF) grant CMMI-1921646 https://www.nsf.gov/awardsearch/showAward?AWD_ID=1921646.

License GPL (>= 2)
Encoding UTF-8
RoxygenNote 7.2.1
NeedsCompilation no
Author Qian Xiao [aut], V. Roshan Joseph [aut, cre]
Maintainer V. Roshan Joseph <roshan@gatech.edu>
Repository CRAN
Date/Publication 2022-10-29 08:52:56 UTC

R topics documented:

measure ........................................................................ 2
mofat ........................................................................ 3
Index ........................................................................ 5
measure

**Screening measures**

**Description**
This function can be used for computing screening measures.

**Usage**

```r
measure(design, y)
```

**Arguments**

- `design` design matrix, which should have the Sobol’ design structure
- `y` response vector

**Details**

The `measure` function computes the screening measures such as the total Sobol’ indices (Sobol’ 1993) and $\mu^*$ measure of Campolongo et al. (2007). The design matrix should have the Sobol’ design structure. Please see Xiao et al. (2022) for details.

**Value**

- `t` Total Sobol’ index
- `mustar` $\mu^*$ measure

**Author(s)**

Qian Xiao and V. Roshan Joseph

**References**


Examples

```r
#Friedman function
fun <- function (X)
{
  return(Y)
}
design = mofat(p=10, l=3)
y = apply(design, 1, fun)

#Screening measures
measure(design, y)
```

## Description

This function can be used for generating MOFAT designs.

## Usage

```r
mofat(p, l, method = "best")
```

## Arguments

- `p`: number of factors
- `l`: number of base runs
- `method`: choose among "uniform", "projection", and "best"

## Details

The `mofat` function generates the MOFAT design for a given number of factors \( p \geq 2 \) and number of base runs \( l \geq 3 \). The total number of runs in the MOFAT design will be \( l(p+1) \). A MOFAT design can be viewed as an optimized version of Morris screening design (Morris 1991) by exploiting its connections with the Monte Carlo-based design of Sobol' (1993). Please see Xiao et al. (2022) for details.

Three choices for the `method` are given: "uniform", "projection", and "best". Option "uniform" gives 1 equally-spaced levels for the entire design, which are also balanced. "projection" option adjusts the levels of the two base matrices A and B such that there are \( 2l \) or \( 2l - 1 \) levels in the design depending on \( l \) is even or odd. Option "best" (default) chooses the best among the first two options using maximin distance criterion.

## Value

- `design`: MOFAT design
Author(s)

Qian Xiao and V. Roshan Joseph

References


Examples

# MOFAT with three base runs
mofat(p=10, l=3, method="uniform")
mofat(p=10, l=3, method="projection")

# MOFAT with five base runs
mofat(p=10, l=5)
dim(mofat(p=125, l=5))
Index

measure, 2
mofat, 3