

# Package ‘OSFD’

June 19, 2023

**Type** Package

**Title** Output Space-Filling Design

**Version** 1.0

**Date** 2023-06-01

**Description**

A method to generate a design in the input space that sequentially fills the output space of a black-box function. The output space-filling design will be helpful in inverse design or feature-based modeling problem.

Please see Wang et al.(2023) <[DOI:10.48550/arXiv.2305.07202](https://doi.org/10.48550/arXiv.2305.07202)> for details. This work is supported by U.S. National Foundation grant CMMI-1921646.

**License** GPL (>= 2)

**Depends** Rcpp (>= 1.0.8), lhs, twinning, dplyr

**LinkingTo** Rcpp, RcppArmadillo, twinning

**RoxygenNote** 7.2.3

**Encoding** UTF-8

**NeedsCompilation** yes

**Maintainer** Shangkun Wang <[sk\\_wang@gatech.edu](mailto:sk_wang@gatech.edu)>

**Repository** CRAN

**Author** Shangkun Wang [aut, cre],  
Roshan Joseph [aut]

**Date/Publication** 2023-06-19 15:30:17 UTC

## R topics documented:

OSFD-package . . . . .	2
ball_unif . . . . .	2
mMdist . . . . .	3
OSFD . . . . .	4

<b>Index</b>	<b>6</b>
--------------	----------

---

OSFD-package

*A sequential algorithm to generate designs that fill the output space*

---

### Description

A sequential method to generate a design that produces points filling the output space. The underlying mapping  $f$  from input space to output space is assumed to be a black-box function that can be evaluated in the forward direction. Please see Wang et al. (2023) for details.

### Author(s)

Shangkun Wang, V. Roshan Joseph

Maintainer: Shangkun Wang <sk\_wang@gatech.edu>

### References

Wang, Shangkun, Adam P. Generale, Surya R. Kalidindi, and V. Roshan Joseph. "Sequential Designs for Filling Output Spaces." *Technometrics*, to appear (2023).

---

ball\_unif

*(Quasi) uniform points in a p-dimensional ball*

---

### Description

ball\_unif generate random or quasi-random uniform points in a p-dimensional ball.

### Usage

```
ball_unif(cen, rad, n, rand = TRUE)
```

### Arguments

cen	a vector specifying the center of the ball.
rad	radius of the ball.
n	number of points.
rand	whether to generate random or quasi random points. Default value is TRUE.

### Details

ball\_unif generate random uniform points or quasi uniform points by twinning algorithm in a p-dimensional ball.

### Value

a matrix of the generated points.

## References

Vakayil, Akhil, and V. Roshan Joseph. "Data twinning." *Statistical Analysis and Data Mining: The ASA Data Science Journal* 15.5 (2022): 598-610.

Wang, Shangkun, Adam P. Generale, Surya R. Kalidindi, and V. Roshan Joseph. "Sequential Designs for Filling Output Spaces." *Technometrics*, to appear (2023).

## Examples

```
x = ball_unif(c(0,0),1,10,rand=FALSE)
plot(x,type='p')
```

---

mMdist

*Minimax distance*

---

## Description

mMdist computes the minimax distance of a design in a specified region. A large uniform sample from the specified region is needed to compute the minimax distance.

## Usage

```
mMdist(X, X_space)
```

## Arguments

X                    a matrix specifying the design.  
X\_space             a large sample of uniform points in the space of interest.

## Details

mMdist approximates the minimax distance of a set of points X by the large sample X\_space in the space of interest.

## Value

the minimax distance.

## References

Johnson, Mark E., Leslie M. Moore, and Donald Ylvisaker. "Minimax and maximin distance designs." *Journal of statistical planning and inference* 26.2 (1990): 131-148.

Wang, Shangkun, Adam P. Generale, Surya R. Kalidindi, and V. Roshan Joseph. "Sequential Designs for Filling Output Spaces." *Technometrics*, to appear (2023).

**Examples**

```
# the minimax distance of a random Latin hypercube design
D = randomLHS(5,2)
mMdist(D,replicate(2,runif(1e5)))
```

---

OSFD

*Output space-filling design*


---

**Description**

This function is for producing designs that fill the output space.

**Usage**

```
OSFD(
  D = NULL,
  f,
  p,
  q,
  n_ini = NA,
  n,
  scale = TRUE,
  method = "EI",
  CAND = NULL,
  rand_out = FALSE,
  rand_in = FALSE
)
```

**Arguments**

D	a matrix of the initial design. If not specified, a random Latin hypercube design of size <code>n_ini</code> and dimension <code>p</code> will be generated as initial design.
f	black-box function.
p	input dimension.
q	output dimension.
n_ini	the size of initial design. This initial size must be specified if D is not provided.
n	the size of the final design.
scale	whether to scale the output points to 0 to 1 for each dimension.
method	two choices: 'EI' or 'Greedy'; the default is 'EI'.
CAND	the candidate points in the input space. If Null, it will be automatically generated.

rand_out	whether to use random uniform points or quasi random points by twinning algorithm for generating points in spheres for output space approximation. The default value is FALSE.
rand_in	whether to use random uniform points or quasi random points by twinning algorithm for generating points in spheres for input space candidate sets. The default value is FALSE.

### Details

OSFD produces a design that fills the output space using the sequential algorithm by Wang et al. (2023).

### Value

D	the final design points in the input space
Y	the output points

### References

Wang, Shangkun, Adam P. Generale, Surya R. Kalidindi, and V. Roshan Joseph. "Sequential Designs for Filling Output Spaces." *Technometrics*, to appear (2023).

### Examples

```
# test function: inverse-radius function (Wang et.al 2023)
inverse_r = function(x){
  epsilon = 0.1
  y1=1/(x[1]^2+x[2]^2+epsilon^2)^(1/2)
  if (x[2]==0){
    y2 = 0
  }else if (x[1]==0) {
    y2 = pi/2}else{
    y2 = atan(x[2]/x[1])
  }
  return (c(y1=y1,y2=y2))
}

set.seed(2022)
p = 2
q = 2
f = inverse_r
n_ini = 10
n = 50
osfd = OSFD(f=f,p=p,q=q,n_ini=n_ini,n=n)
D = osfd$D
Y = osfd$Y
```

# Index

ball\_unif, [2](#)

mMdist, [3](#)

OSFD, [4](#)

OSFD-package, [2](#)