Package 'SCCDdesign'

September 21, 2026

Title Construction of Screening Designs for Mixed Level Continuous and Categorical Factors

Version 0.2.0

Description Constructs screening designs for experiments involving continuous and categorical factors with multiple levels. The package implements methods for constructing mixed-level screening designs, involving factors with more than two levels. It also evaluates the statistical performance of screening designs throughdev power to identify active effects and Type I error rates.
The package implements three methods proposed by Jones, B., Lekivetz, R., Majumdar, D. and Nachtsheim, C. (2025) <doi:10.1080/00401706.2024.2362149> for generating efficient screening designs involving three-level continuous and two-level categorical factors for even run sizes.
It also includes Paley Type I and Type II constructions for conference matrices and pseudo conference matrices obtained using the coordinate exchange algorithm by Jones, B. and Nachtsheim, C. J. (2011) <doi:10.1080/00224065.2011.11917841> which are used in the development of these screening designs.

License GPL-3

Encoding UTF-8

RoxygenNote 8.0.0

Imports pracma, AlgDesign, leaps, stats

NeedsCompilation no

Author Vyshna I C [aut, cre],
Cini Varghese [aut, ctb],
Safeela Nasrin [aut],
Boyina Devi Priyanka [aut, ctb],
Mohd Harun [aut, ctb],
Anindita Datta [aut, ctb]

Maintainer Vyshna I C <vyshnaic@gmail.com>

Repository CRAN

Date/Publication 2026-09-21 10:00:41 UTC

Contents

`conferenceMatrix` *Constructs a conference matrix of a specified order using a direct construction for order 2 and Paley-based constructions for supported odd prime or prime-power values of $q = n - 1$.*

Description

Constructs a conference matrix of a specified order using a direct construction for order 2 and Paley-based constructions for supported odd prime or prime-power values of $q = n - 1$.

Usage

```
conferenceMatrix(n)
```

Arguments

`n`	A single finite integer specifying the order of the conference matrix. The value must be at least 2.

Details

For `n = 2`, the conference matrix is constructed directly. For larger orders, the function sets $q = n-1$ and determines whether q is a supported odd prime or odd prime power. Depending on the value of q modulo 4, the corresponding Paley construction is used. An error is returned when the supplied order does not satisfy the requirements of the implemented construction.

Value

A numeric $n \times n$ conference matrix.

Examples

```
conferenceMatrix(6)
```

`dOptimalTwoLevel` *Constructs a D-optimal two-level design with* `m` *runs and* `m - 1` *factors. The factor levels are coded as -1 and +1.*

Description

Constructs a D-optimal two-level design with `m` runs and `m - 1` factors. The factor levels are coded as -1 and +1.

Usage

```
dOptimalTwoLevel(m)
```

Arguments

`m`	A positive integer specifying the number of runs. The minimum value is 3.

Details

The candidate set is generated using all possible combinations of -1 and +1 for the factors. The D-optimal design is selected using `AlgDesign::optFederov()`.

Value

A numeric matrix containing the D-optimal two-level design.

Examples

```
dOptimalTwoLevel(4)
dOptimalTwoLevel(8)
```

`hadamardMatrix` *Constructs a Hadamard matrix of order* `n` *using* `pracma::hadamard()`.

Description

Constructs a Hadamard matrix of order `n` using `pracma::hadamard()`.

Usage

```
hadamardMatrix(n)
```

Arguments

`n`	A positive integer specifying the order of the matrix.

Details

A Hadamard matrix contains only +1 and -1, and its rows are mutually orthogonal. The supported orders are of the forms 2^e, $12(2^e)$, and $20(2^e)$.

Value

A Hadamard matrix of order `n`.

Examples

```
hadamardMatrix(4)
hadamardMatrix(8)
hadamardMatrix(12)
```

`mixed_level_design` *Constructs an orthogonal design for two types of factors with different numbers of levels. Three construction cases are considered: even-even level factors, odd-even level factors and odd-odd level factors. The appropriate construction is selected automatically according to the parity of the numbers of levels of the two factors. Hadamard matrices are used for even-level factors, while conference matrices are used for odd-level factors. The order of both the Hadamard and conference matrices is* `m`*. For the even-even case, the design contains* `m - 1` *factors of each type. For the odd-even case, the design contains* `m` *odd-level factors and* `m - 1` *even-level factors. For the odd-odd case, the design contains* `m` *factors of each type.*

Description

Constructs an orthogonal design for two types of factors with different numbers of levels. Three construction cases are considered: even-even level factors, odd-even level factors and odd-odd level factors. The appropriate construction is selected automatically according to the parity of the numbers of levels of the two factors. Hadamard matrices are used for even-level factors, while conference matrices are used for odd-level factors. The order of both the Hadamard and conference matrices is `m`. For the even-even case, the design contains `m - 1` factors of each type. For the odd-even case, the design contains `m` odd-level factors and `m - 1` even-level factors. For the odd-odd case, the design contains `m` factors of each type.

Usage

```
mixed_level_design(a, b, m)
```

Arguments

`a`	Number of levels of the first factor.
`b`	Number of levels of the second factor.
`m`	Order of the Hadamard and conference matrices used in the construction. Must be a multiple of 4.

Value

A matrix containing the constructed orthogonal design. The total number of factors depends on the construction case: `2(m - 1)` for the even-even case, `2m - 1` for the odd-even case, and `2m` for the odd-odd case.

Examples

```
mixed_level_design(a = 4, b = 2, m = 4)
mixed_level_design(a = 3, b = 2, m = 8)
mixed_level_design(a = 5, b = 3, m = 4)
```

---

pseudoConference *Constructs a pseudo-conference matrix of order* `n` *using a determinant-based search procedure.*

Description

Constructs a pseudo-conference matrix of order `n` using a determinant-based search procedure.

Usage

```
pseudoConference(n, starts = 1000, max.iter = 100)
```

Arguments

`n`	A positive integer specifying the order of the matrix.
`starts`	Number of random starting matrices used in the search. Default is 1000.
`max.iter`	Maximum number of iterations for each starting matrix. Default is 100.

Details

The matrix has zero diagonal elements and off-diagonal elements equal to -1 or +1. The search selects a matrix by maximizing the determinant criterion.

Value

A list containing the pseudo-conference matrix and the corresponding log-determinant value.

Examples

```
result <- pseudoConference(6, starts = 10, max.iter = 10)
result$Matrix
result$LogDet
```

SCCD *Constructs an SCCD for a specified number of runs (n=2m). The construction method is selected automatically according to the number of runs.*

Description

Constructs an SCCD for a specified number of runs (n=2m). The construction method is selected automatically according to the number of runs.

Usage

```
SCCD(n)
```

Arguments

n	A positive even integer specifying the number of runs.

Details

Method I is used when n is divisible by 8. Method II is used when n is divisible by 4 but not by 8. Method III is used when n is even but not divisible by 4.

Value

A numeric SCCD matrix containing m continuous and m-1 categorical factors.

Examples

```
SCCD(16)
```

simulation *Performs a simulation study to evaluate the ability of a design to identify active main effects using best-subsets regression and the AICc criterion.*

Description

Performs a simulation study to evaluate the ability of a design to identify active main effects using best-subsets regression and the AICc criterion.

Usage

```
simulation(X, SNR, n.active, nrep = 1000)
```

Arguments

X	A design matrix containing the mixed-level factors.
SNR	A numeric value specifying the signal-to-noise ratio.
n.active	An integer specifying the number of active factors.
nrep	An integer specifying the number of simulation replications. Default is 1000.

Value

A data frame containing the SNR, number of active factors, power, and Type I error rate.

Examples

```
## Not run:
X <- matrix(rnorm(20 * 5), nrow = 20, ncol = 5)
simulation(X, SNR = 1, n.active = 2, nrep = 100)

## End(Not run)
```

Index