

# Package: iRoCoDe (via r-universe)

June 10, 2026

**Type** Package

**Title** Incomplete Row-Column Designs

**Version** 1.0.1

**Maintainer** Md. Ashraful Haque <ashrafulhaque664@gmail.com>

**Description** The Row-column designs are widely recommended for experimental situations when there are two well-identified factors that are cross-classified representing known sources of variability. These designs are expected to result a gain in accuracy of estimating treatment comparisons in an experiment as they eliminate the effects of the row and column factors. However, these designs are not readily available when the number of treatments is more than the levels of row and column blocking factors. This package named 'iRoCoDe' generates row-column designs with incomplete rows and columns, by amalgamating two incomplete block designs (D1 and D2). The selection of D1 and D2 (the input designs) can be done from the available incomplete block designs, viz., balanced incomplete block designs/ partially balanced incomplete block designs/ t-designs. (Mcsorley, J.P., Phillips, N.C., Wallis, W.D. and Yucas, J.L. (2005).<doi:10.1007/s10623-003-6149-9>).

**License** GPL (>= 2)

**Encoding** UTF-8

**Author** Sayantani Karmakar [aut], Md. Ashraful Haque [aut, cre], Cini Varghese [aut], Seema Jaggi [aut], Eldho Varghese [aut], Mohd. Harun [aut]

**NeedsCompilation** no

**Repository** <https://mah-iasri.r-universe.dev>

**Date/Publication** 2022-01-25 15:02:43 UTC

**RemoteUrl** <https://github.com/cran/iRoCoDe>

**RemoteRef** HEAD

**RemoteSha** 4a14c4fd9cbe66e2e0cde09018f66077e32e09ec

## Contents

a_matrix . . . . .	2
get_rep . . . . .	3
get_treat . . . . .	3
iRoCoDe . . . . .	4
n_matrix . . . . .	5
rearrange_mat . . . . .	6
rmv_list . . . . .	7
row_unique . . . . .	7
<b>Index</b>	<b>9</b>

---

a_matrix	<i>Generates the 'a_matrix' (A) of a design</i>
----------	---

---

### Description

The a\_matrix function creates an 'a\_matrix' (A) with help of design D2 and 'n\_matrix' (N) of design D1.

### Usage

```
a_matrix(design2, n_matrix)
```

### Arguments

design2	A design in matrix format
n_matrix	An incidence matrix of a design

### Details

This functions reads the second design (D2) and the incidence matrix or 'n\_matrix' (N) of the first design (D1) as input in matrix format. The dimation of the 'a\_matrix' (A) is same as the input incidence matrix (N).

### Value

It returns matrix (A) with the same dimension as of 'n\_matrix' (N).

### Author(s)

1) Sayantani Karmakar 2) Md. Ashraful Haque 3) Cini Varghese 4) Seema Jaggi 5) Eldho Varghese  
6) Mohd. Harun

---

get_rep	<i>Generates number of replications in a design</i>
---------	---

---

**Description**

The get\_rep function finds the number of replications of the treatments in a given design.

**Usage**

```
get_rep(design)
```

**Arguments**

design	A design in matrix format
--------	---------------------------

**Details**

This function reads a design (D) in matrix format and obtain the number of replications (r) of the treatments (v) of the given design where rows are treated as blocks.

**Value**

It returns a scalar value denoting the number of replications of the treatments in the design.

**Author(s)**

1) Sayantani Karmakar 2) Md. Ashraful Haque 3) Cini Varghese 4) Seema Jaggi 5) Eldho Varghese  
6) Mohd. Harun

---

get_treat	<i>Generates number of treatments in a Design</i>
-----------	---

---

**Description**

The get\_treat function finds the number of treatments in a given design.

**Usage**

```
get_treat(design)
```

**Arguments**

design	A design in matrix format
--------	---------------------------

**Details**

This function reads a design (D) in matrix format and finds the number of treatments (v) in the given design where rows are treated as blocks.

**Value**

A list of treatments in a design.

**Author(s)**

1) Sayantani Karmakar 2) Md. Ashraful Haque 3) Cini Varghese 4) Seema Jaggi 5) Eldho Varghese  
6) Mohd. Harun

---

iRoCoDe

*Generate Incomplete Row-Column Designs*


---

**Description**

The 'iRoCoDe' function uses an algorithmic approach for constructing row-column designs with incomplete rows and columns by using two incomplete block designs. A wide range of incomplete block designs, viz., balanced incomplete block designs (BIBD)/ partially balanced incomplete block designs (PBIBD)/ t-designs can be used in this function to construct these designs.

**Usage**

```
iRoCoDe(design1, design2)
```

**Arguments**

design1	first design (D1) having parameters: no. of treatments = v1, No. of blocks = b1, block size = k1, and treatment-replications = r1
design2	second design (D2) having parameters: no. of treatments = v2, No. of blocks = b2, block size = k2, and treatment-replications = r2

**Details**

The function takes two incomplete block designs D1 and D2 (having parameters v1, b1, k1, r1 and v2, b2, k2, r2) as input in matrix format. The input designs (D1 and D2) must satisfy the parametric relationships:  $v_1 = r_2$ ,  $b_1 = b_2$  and  $k_1 = k_2$  for generating the new design using this function.

**Value**

It returns a matrix with dimension  $v_1 \times v_2$ .

**Author(s)**

1) Sayantani Karmakar 2) Md. Ashraful Haque 3) Cini Varghese 4) Seema Jaggi 5) Eldho Varghese  
6) Mohd. Harun

## References

McSorley, J.P., Phillips, N.C., Wallis, W.D. & Yucas, J.L. (2005). Double arrays, triple arrays and balanced grids. *Designs, Codes and Cryptography*, 35(1), 21 - 45

## Examples

```
library(iRoCoDe)
D1= matrix(c(1, 2, 5,
             2, 3, 6,
             3, 4, 1,
             4, 5, 6,
             5, 1, 3,
             6, 1, 4,
             2, 3, 4,
             3, 5, 6,
             6, 1, 2,
             2, 4, 5), nrow = 10, ncol = 3, byrow = TRUE)
D2 = matrix(c(1, 4, 5,
             2, 4, 5,
             1, 3, 4,
             1, 2, 5,
             2, 3, 5,
             1, 3, 5,
             1, 2, 4,
             1, 2, 3,
             2, 3, 4,
             3, 4, 5), nrow = 10, ncol = 3, byrow = TRUE)

D = iRoCoDe(D1,D2)
## Output:
#[1] "Incomplete Row-Column Design"
#      [,1] [,2] [,3] [,4] [,5]
#[1,]    6    9    3    1    5
#[2,]    7    2   10    9    1
#[3,]    8    7    5    3    2
#[4,]    3    4    6    7   10
#[5,]    1    5    8   10    4
#[6,]    4    8    9    2    6
```

---

n\_matrix

*Generates the Incidence Matrix (N) of a Design*


---

## Description

The n\_matrix functions creates an incidence matrix/'n\_matrix' (N) from a given design.

## Usage

```
n_matrix(design, treatments, blocks)
```

**Arguments**

design	A design in matrix format
treatments	Number of treatments in the given design
blocks	Number of blocks in the given design

**Details**

This functions reads a design (D1) in matrix format, its no. of treatments (v1) and its no. of blocks (b1) as input. Here the rows of design D1 are treated as blocks. The generated incidence matrix (N) will have the dimension of (v1 x b1).

**Value**

It returns a matrix (N) with the dimension of (v1 x b1)

**Author(s)**

1) Sayantani Karmakar 2) Md. Ashrafal Haque 3) Cini Varghese 4) Seema Jaggi 5) Eldho Varghese  
6) Mohd. Harun

---

rearrange_mat	<i>Rearranges the given matrix</i>
---------------	------------------------------------

---

**Description**

The rearrange\_mat function rearranges the column elements of a given 'a\_matrix'.

**Usage**

```
rearrange_mat(a_matrix, treatments)
```

**Arguments**

a_matrix	A a_matrix of the first design (D1)
treatments	The number of treatments of the second design (D2)

**Details**

This function reads an 'a\_matrix' (A) generated from two designs (D1 and D2) and rearranges the column elements based on the treatments numbers in the second design (D2). It iteratively rearrange the items row-wise until the new A is generated with each row containing all the treatments of D2 exactly once.

**Value**

It returns new 'a\_matrix' (A) with each row containing all the treatments of D2 exactly once.

**Author(s)**

1) Sayantani Karmakar 2) Md. Ashraful Haque 3) Cini Varghese 4) Seema Jaggi 5) Eldho Varghese  
6) Mohd. Harun

---

rmv_list	<i>Removes a item from a list</i>
----------	-----------------------------------

---

**Description**

The rmv\_list function removes the specified item from the given list.

**Usage**

```
rmv_list(list, val)
```

**Arguments**

list	A list of values
val	A specific value

**Details**

This function reads a list of values and removes the specified value from that list.

**Value**

It returns a list.

**Author(s)**

1) Sayantani Karmakar 2) Md. Ashraful Haque 3) Cini Varghese 4) Seema Jaggi 5) Eldho Varghese  
6) Mohd. Harun

---

row_unique	<i>Checks uniqueness of elements in the rows of a matrix</i>
------------	--

---

**Description**

The row\_unique function checks whether each row contains unique treatments.

**Usage**

```
row_unique(a_matrix, treatments, rows, cols)
```

**Arguments**

<code>a_matrix</code>	An 'a_matrix' of the first design (D1)
<code>treatments</code>	The treatments of the second design (D2)
<code>rows</code>	Number of rows in the given 'a_matrix'
<code>cols</code>	Number of columns in the given 'a_matrix'

**Details**

This function reads an 'a\_matrix' (A) generated from two designs (D1 and D2) and checks whether each row contains all the treatments from D2 exactly once.

**Value**

It returns boolean value (either True or False).

**Author(s)**

1) Sayantani Karmakar 2) Md. Ashrafal Haque 3) Cini Varghese 4) Seema Jaggi 5) Eldho Varghese  
6) Mohd. Harun

# Index

`a_matrix`, 2

`get_rep`, 3

`get_treat`, 3

`iRoCoDe`, 4

`n_matrix`, 5

`rearrange_mat`, 6

`rmv_list`, 7

`row_unique`, 7