Package ‘Tri.Hierarchical.IBDs’

August 10, 2023

Type Package

Title Tri-Hierarchical IBDs (Tri- Hierarchical Incomplete Block Designs)

Version 1.0.0

Maintainer Ashutosh Dalal <ashutosh.dalal97@gmail.com>

Description Tri-hierarchical incomplete block design is defined as an arrangement of v treatments each replicated r times in a three system of blocks if, each block of the first system contains m_1 blocks of second system and each block of the second system contains m_2 blocks of the third system. Ignoring the first and second system of blocks, it leaves an incomplete block design with b_3 blocks of size k_3i units; ignoring first and third system of blocks, it leaves an incomplete block design with b_2 blocks each of size k_2i units and ignoring the second and third system of blocks, it leaves an incomplete block design with b_1 blocks each of size k_1 units. For dealing with experimental circumstances where there are three nested sources of variation, a tri-hierarchical incomplete block design can be adopted. Tri - hierarchical incomplete block designs can find application potential in obtaining mating-environmental designs for breeding trials. To know more about nested block designs one can refer Preece (1967) <doi:10.1093/biomet/54.3-4.479>. This package includes series1(), series2(), series3() and series4() functions. This package generates tri-hierarchical designs with six component designs under certain parameter restrictions.

License GPL (>= 2)

Encoding UTF-8

Repository CRAN

RoxygenNote 7.2.3

NeedsCompilation no

Author Nehatai Agashe [aut, ctb],
Cini Varghese [aut, ctb],
Harun Mohammed [ctb],
Ashutosh Dalal [aut, cre]

Date/Publication 2023-08-10 10:10:02 UTC

R topics documented:

`Series1` *Tri-Hierarchical IBDs using Triangular Association Scheme*

Description

This function generates Tri-Hierarchical IBDs based on Triangular association scheme. Here, v=nC2, n >=5. We find balanced incomplete block designs (BIBD) at block level and triangular PBIB designs at sub-block level as well as sub-sub block level. Information matrix pertaining to the estimation of treatments effects, canonical efficiency factor in comparison to an orthogonal design and six component designs are provided.

Usage

```
Series1(
  v,
  D1 = FALSE,
  D2 = FALSE,
  D3 = FALSE,
  D4 = FALSE,
  D5 = FALSE,
  D6 = FALSE,
  Randomization = FALSE
)
```

Arguments

`v`	Number of treatments, v = nC2 where n>=5
`D1`	Bi-Hierarchical IBD by ignoring blocks
`D2`	Bi-Hierarchical IBD by ignoring sub-blocks
`D3`	Bi-Hierarchical IBD by ignoring sub-sub blocks
`D4`	IBD at block level
`D5`	IBD at sub block level
`D6`	IBD at sub-sub block level
`Randomization`	Randomization of layout of the designs if needed enter TRUE; by default it is FALSE.

Value

It gives Tri-HIB design and six component designs with canonical efficiency factor in comparison to an orthogonal design.

Note

Numbers in the outer most parentheses represents as block elements, second level parentheses as sub block elements and inner most parentheses as sub-sub block elements.

References

Preece, D.A. (1967) <https://doi.org/10.1093/biomet/54.3-4.479>."Nested balanced incomplete block designs".

Examples

```
library(Tri.Hierarchical.IBDs)
Series1(15,D1=TRUE,D2=TRUE,D3=TRUE,D4=TRUE,D5=FALSE,D6=TRUE,Randomization=FALSE)
```

Series2 *Tri-Hierarchical IBDs using Latin Square Association Scheme*

Description

It generates the Tri-Hierarchical IBDs based on Latin Square association scheme. Here, number of treatments (v) should be a perfect square. We find balanced incomplete block designs (BIBD) at block level and latin square PBIB designs at sub-block level as well as sub-sub block level. Information matrix pertaining to the estimation of treatments effects, canonical efficiency factor in comparison to an orthogonal design and six component designs are provided.

Usage

```
Series2(
  v,
  D1 = FALSE,
  D2 = FALSE,
  D3 = FALSE,
  D4 = FALSE,
  D5 = FALSE,
  D6 = FALSE,
  Randomization = FALSE
)
```

Arguments

`v`	Number of treatments, v (>=16) should be a square number
`D1`	Bi-Hierarchical IBD by ignoring blocks
`D2`	Bi-Hierarchical IBD by ignoring sub-blocks
`D3`	Bi-Hierarchical IBD by ignoring sub-sub blocks
`D4`	IBD at block level
`D5`	IBD at sub block level
`D6`	IBD at sub-sub block level
`Randomization`	Randomization of layout of the designs if needed enter TRUE; by default it is FALSE.

Value

It gives Tri-HIB design and six component designs with canonical efficiency factor in comparison to an orthogonal design.

Note

Numbers in the outer most parentheses represents as block elements, second level parentheses as sub block elements and inner most parentheses as sub-sub block elements.

References

Preece, D.A. (1967) <https://doi.org/10.1093/biomet/54.3-4.479>."Nested balanced incomplete block designs".

Examples

```
library(Tri.Hierarchical.IBDs)
Series2(16,D1=TRUE,D2=TRUE,D3=FALSE,D4=FALSE,D5=FALSE,D6=FALSE,Randomization=FALSE)
```

`Series3` *Tri-Hierarchical IBDs using Rectangular Association Scheme*

Description

This function provides the Tri-Hierarchical IBDs based on Rectangular association scheme. Here, v= m*n, v should be composite number and (m,n)>=3. We find balanced incomplete block designs (BIBD) at block level and rectangular PBIB designs at sub-block level as well as sub-sub block level. Information matrix pertaining to the estimation of treatments effects, canonical efficiency factor in comparison to an orthogonal design and six component designs are provided.

Usage

```
Series3(
  m,
  n,
  D1 = FALSE,
  D2 = FALSE,
  D3 = FALSE,
  D4 = FALSE,
  D5 = FALSE,
  D6 = FALSE,
  Randomization = FALSE
)
```

Arguments

`m`	Any integer >=3
`n`	Any integer >=3
`D1`	Bi-Hierarchical IBD by ignoring blocks
`D2`	Bi-Hierarchical IBD by ignoring sub-blocks
`D3`	Bi-Hierarchical IBD by ignoring sub-sub blocks
`D4`	IBD at block level
`D5`	IBD at sub block level
`D6`	IBD at sub-sub block level
`Randomization`	Randomization of layout of the designs if needed enter TRUE; by default it is FALSE.

Value

It gives Tri-HIB design and six component designs with canonical efficiency factor in comparison to an orthogonal design.

Note

Numbers in the outer most parentheses represents as block elements, second level parentheses as sub block elements and inner most parentheses as sub-sub block elements.

References

Preece, D.A. (1967) <https://doi.org/10.1093/biomet/54.3-4.479>."Nested balanced incomplete block designs".

Examples

```
library(Tri.Hierarchical.IBDs)
Series3(4,3,D1=TRUE,D2=TRUE, D3=TRUE, D4=TRUE,D5=FALSE,D6=TRUE,Randomization=TRUE)
```

Series4 *Tri-Hierarchical IBDs using Initial Block Solution*

Description

This function gives Tri-Hierarchical IBDs using initial sequences. Here, v = 4t+1 or 4t+3, where t is an integer and v should be a prime number, using primitive element of Galois field designs are generated. We find balanced incomplete block designs (BIBD) at block level and PBIB designs at sub-block level as well as sub-sub block level with circular association scheme. Information matrix pertaining to the estimation of treatments effects, canonical efficiency factor in comparison to an orthogonal design and six component designs are provided.

Usage

```
Series4(
  v,
  D1 = FALSE,
  D2 = FALSE,
  D3 = FALSE,
  D4 = FALSE,
  D5 = FALSE,
  D6 = FALSE,
  Randomization = FALSE
)
```

Arguments

`v`	Number of treatments, (11 <= v < 200) a prime number
`D1`	Bi-Hierarchical IBD by ignoring blocks
`D2`	Bi-Hierarchical IBD by ignoring sub-blocks
`D3`	Bi-Hierarchical IBD by ignoring sub-sub blocks
`D4`	IBD at block level
`D5`	IBD at sub block level
`D6`	IBD at sub-sub block level
`Randomization`	Randomization of layout of the designs if needed enter TRUE; by default it is FALSE.

Value

It gives Tri-HIB design and six component designs with canonical efficiency factor in comparison to an orthogonal design.

Note

Numbers in the outer most parentheses represents as block elements, second level parentheses as sub block elements and inner most parentheses as sub-sub block elements.

References

Preece, D.A. (1967) <https://doi.org/10.1093/biomet/54.3-4.479>."Nested balanced incomplete block designs".

Examples

```
library(Tri.Hierarchical.IBDs)
Series4(13,D1=FALSE,D2=FALSE,D3=TRUE,D4=TRUE,D5=FALSE,D6=TRUE,Randomization=TRUE)
```

Index