

Package: blockrand (via r-universe)

October 17, 2024

Type Package

Title Randomization for Block Random Clinical Trials

Version 1.5

Date 2020-04-01

Author Greg Snow <538280@gmail.com>

Maintainer Greg Snow <538280@gmail.com>

Description Create randomizations for block random clinical trials.
Can also produce a pdf file of randomization cards.

License GPL-2

Repository/R-Forge/Project blockrand

Repository/R-Forge/Revision 14

Repository/R-Forge/DateTimeStamp 2020-04-01 21:50:26

Date/Publication 2020-04-06 10:02:15 UTC

NeedsCompilation no

Repository <https://glsnow.r-universe.dev>

RemoteUrl <https://github.com/cran/blockrand>

RemoteRef HEAD

RemoteSha 13381efc2bc270d847ea4481bb675bb74fe82d6c

Contents

| | |
|-----------------------------|---|
| blockrand-package | 2 |
| blockrand | 3 |
| plotblockrand | 5 |

| | |
|--------------|----------|
| Index | 9 |
|--------------|----------|

blockrand-package *Generate block randomizations for clinical trials.*

Description

This package will create a block randomization for clinical trials and help with creating the randomization cards that the study coordinator can use to assign new subjects to their treatment.

Details

Package: blockrand
Type: Package
Version: 1.1
Date: 2008-02-02
License: Gnu Public License Ver. 2
Copyright: Greg Snow and Intermountain Healthcare

Currently there are 2 main functions and an optional list. The `blockrand` function is used to create a data frame with the block sequential treatment randomizations. When doing a stratified study you should run `blockrand` once for each stratum then optionally combine the different data frames with `rbind`. Save the data frame(s) and when the study is completed the data can be added to the data frame for analysis.

The `plotblockrand` function is used to create the randomization cards to be used when assigning subjects to treatment. The cards are printed out and sealed in envelopes, then when a new subject is enrolled the next envelope is opened and the subject assigned to the corresponding treatment.

You can optionally create a list named `blockrand.text` with optional elements `top`, `middle`, and `bottom`. If this list exists and you run `plotblockrand` without specifying these arguments, then the element of the `blockrand.text` list will be used instead.

Author(s)

Greg Snow <538280@gmail.com>

Maintainer: Greg Snow <538280@gmail.com>

Examples

```
## stratified by sex, 100 in stratum, 2 treatments
male <- blockrand(n=100, id.prefix='M', block.prefix='M',stratum='Male')
female <- blockrand(n=100, id.prefix='F', block.prefix='F',stratum='Female')

my.study <- rbind(male,female)

## Not run:
plotblockrand(my.study, 'mystudy.pdf',
  top=list(text=c('My Study', 'Patient: %ID%', 'Treatment: %TREAT%'),
```

```

        col=c('black','black','red'),font=c(1,1,4)),
middle=list(text=c("My Study","Sex: %STRAT%","Patient: %ID%"),
            col=c('black','blue','green'),font=c(1,2,3)),
bottom="Call 123-4567 to report patient entry",
cut.marks=TRUE)

### or

blockrand.txt <- list(
  top=list(text=c('My Study','Patient: %ID%','Treatment: %TREAT%'),
          col=c('black','black','red'),font=c(1,1,4)),
  middle=list(text=c("My Study","Sex: %STRAT%","Patient: %ID%"),
             col=c('black','blue','green'),font=c(1,2,3)),
  bottom="Call 123-4567 to report patient entry")

plotblockrand(my.study, 'mystudy.pdf', cut.marks=TRUE)

## End(Not run)

```

blockrand

Generate a block randomization for a clinical trial

Description

This function creates random assignments for clinical trials (or any experiment where the subjects come one at a time). The randomization is done within blocks so that the balance between treatments stays close to equal throughout the trial.

Usage

```

blockrand(n, num.levels = 2, levels = LETTERS[seq(length = num.levels)],
id.prefix, stratum, block.sizes = 1:4, block.prefix,
uneq.beg=FALSE, uneq.mid=FALSE, uneq.min=0, uneq.maxit=10)

```

Arguments

| | |
|--------------|--|
| n | The minimum number of subjects to randomize |
| num.levels | The number of treatments or factor levels to randomize between |
| levels | A character vector of labels for the different treatments or factor levels |
| id.prefix | Optional integer or character string to prefix the id column values |
| stratum | Optional character string specifying the stratum being generated |
| block.sizes | Vector of integers specifying the sizes of blocks to use |
| block.prefix | Optional integer or character string to prefix the block.id column |
| uneq.beg | Should an unequal block be used at the beginning of the randomization |
| uneq.mid | Should an unequal block be used in the middle |

| | |
|------------|---|
| uneq.min | what is the minimum difference between the most and least common levels in an unequal block |
| uneq.maxit | maximum number of tries to get uneq.min difference |

Details

This function will randomize subjects to the specified treatments within sequential blocks. The total number of randomizations may end up being more than n .

The final block sizes will actually be the product of `num.levels` and `block.sizes` (e.g. if there are 2 levels and the default block sizes are used (1:4) then the actual block sizes will be randomly chosen from the set (2,4,6,8)).

If `id.prefix` is not specified then the `id` column of the output will be a sequence of integers from 1 to the number of rows. If `id.prefix` is numeric then the `id` column of the output will be a sequence of integers starting at the value of `id.prefix`. If `id.prefix` is a character string then the numbers will be converted to strings (zero padded) and have the prefix prepended.

The `block.prefix` will be treated in the same way as the `id.prefix` for identifying the blocks. The one difference being that the `block.id` will be converted to a factor in the final data frame.

If `uneq.beg` and/or `uneq.mid` are true then an additional block will be used at the beginning and/or inserted in the middle that is not balanced, this means that the final totals in each group may not be exactly equal (but still similar). This makes it more difficult to anticipate future assignments as the numbers will not return to equality at the end of each block.

For stratified studies the `blockrand` function should run once each for each stratum using the `stratum` argument to specify the current stratum (and using `id.prefix` and `block.prefix` to keep the id's unique). The separate data frames can then be combined using `rbind` if desired.

Value

A data frame with the following columns:

| | |
|-------------------------|--|
| <code>id</code> : | A unique identifier (number or character string) for each row |
| <code>stratum</code> : | Optional, if <code>stratum</code> argument is specified it will be replicated in this column |
| <code>block.id</code> : | An identifier for each block of the randomization, this column will be a factor |
| <code>block.size</code> | The size of each block |
| <code>treatment</code> | The treatment assignment for each subject |

Author(s)

Greg Snow <538280@gmail.com>

References

Schulz, K. and Grimes, D. (2002): Unequal group sizes in randomized trials: guarding against guessing, *The Lancet*, 359, pp 966–970.

See Also

[plotblockrand](#), [sample](#), [rbind](#)

Examples

```
## stratified by sex, 100 in stratum, 2 treatments
male <- blockrand(n=100, id.prefix='M', block.prefix='M',stratum='Male')
female <- blockrand(n=100, id.prefix='F', block.prefix='F',stratum='Female')

my.study <- rbind(male,female)

## Not run:
plotblockrand(my.study,'mystudy.pdf',
  top=list(text=c('My Study','Patient: %ID%','Treatment: %TREAT%'),
    col=c('black','black','red'),font=c(1,1,4)),
  middle=list(text=c("My Study","Sex: %STRAT%","Patient: %ID%"),
    col=c('black','blue','green'),font=c(1,2,3)),
  bottom="Call 123-4567 to report patient entry",
  cut.marks=TRUE)

## End(Not run)
```

plotblockrand

Create a pdf file of randomization cards

Description

Creates a pdf file of randomization cards based on the output from blockrand. This file can then be printed and the cards put into envelopes for use by a study coordinator for assigning subjects to treatment.

Usage

```
plotblockrand(x, file = "blockrand.pdf", top, middle, bottom,
  blockrand.text, width = 11, height = 8.5, par.args, id.col = "id",
  stratum.col = "stratum",
  treat.col = "treatment", cut.marks = FALSE, top.ho, top.vo, middle.ho,
  middle.vo, bottom.ho, bottom.vo, nrow=2, ncol=2, ...)
```

Arguments

| | |
|----------------|--|
| x | A data frame, usually the output from blockrand |
| file | The name of the pdf file to create (include the .pdf in the name) |
| top | A character vector or list (see details) with the template to be printed at the top of each card |
| middle | A character vector or list (see details) with the template to be printed in the middle of each card (positioned to be visible through the window of an envelope) |
| bottom | A single character string to be printed at the bottom of each card |
| blockrand.text | A list with default values to use for other options |

| | |
|--------------------------|---|
| <code>width</code> | Passed to pdf |
| <code>height</code> | Passed to pdf |
| <code>par.args</code> | A list containing additional arguments to <code>par</code> before plotting the text |
| <code>id.col</code> | Name or number of the column in <code>x</code> that contains the id's of the subjects |
| <code>stratum.col</code> | Name or number of the column in <code>x</code> that contains the names of the strata |
| <code>treat.col</code> | Name or number of the column in <code>x</code> that contains the treatment assignments |
| <code>cut.marks</code> | Logical, should cut marks be plotted as well (useful if printing on plain paper then cutting apart) |
| <code>top.ho</code> | Shift top text to the right(left) |
| <code>top.vo</code> | Shift top text up(down) |
| <code>middle.ho</code> | Shift middle text to the right(left) |
| <code>middle.vo</code> | Shift middle text up(down) |
| <code>bottom.ho</code> | Shift bottom text to the right(left) |
| <code>bottom.vo</code> | Shift bottom text up(down) |
| <code>nrow</code> | Number of rows of cards to print |
| <code>ncol</code> | Number of columns of cards to print |
| <code>...</code> | Optional arguments passed to pdf |

Details

This function creates a pdf file with randomization "cards". It puts 4 cards per page. You can either print the file onto perforated cards (Avery 8387) or onto regular paper then cut the cards apart. The top of each card can then be folded over (extra protection from someone trying to read the upcoming treatments) and the card placed in an envelope (letter size) with a window and sealed. The envelopes are then used by a study coordinator to assign subjects to treatments as they are enrolled into the trial.

Each card is split into 3 parts, top, middle, and bottom.

The top part is printed flush left and is the part that will be folded over for better security. Information on the treatment assignment goes here along with any other information you want.

The middle part is printed centered so that it will appear through the window of the envelope. The subject ID number and stratification information should go here.

The bottom part is limited to a single line that will be printed flush right at the bottom of the card. This can be used for additional instructions to the study coordinator (e.g. call the statistician at 123-4567 to record assignment).

The top, middle, and bottom templates can be vectors or lists. If the vectors have length greater than 1, then each element of the vector will be printed on a separate line (if there are 3 elements in top then there will be 3 lines at the top, etc.), bottom should only have a single element. If top, middle, or bottom are lists then they should have an element named "text" that consists of a character vector containing the template. The lists can then also have optional elements named "font" and "col", these vectors should be the same length as the "text" vector and represent the fonts and colors to use for the corresponding lines of text (for example if font is `c(1, 2, 1)` then the 2nd line will be printed bold).

If the template in top or middle contains "%ID%" (not including the quotes, but including the percent signs) then this string will be replaced with the contents of the ID column for each card. If they contain "%STRAT%" then it will be replaced with the contents of the stratum column. If top contains "%TREAT%" then it will be replaced with the contents of the treatment column (note that this is not available in the middle template).

If any of the arguments top, middle, or bottom are missing then the function will look for a corresponding element in the blockrand.text argument (a list) to use as the template. If the list does not exist, or the list does not have a corresponding element, then that portion of the card will be blank. Specifying the argument when calling the function will override the blockrand.text list.

The arguments top.ho, middle.ho, and bottom.ho move the corresponding parts to the right (left if negative). The units are approximately strwidth("W") so specifying a value of 0.5 will move the section about half a character to the right. The arguments top.vo, middle.vo, and bottom.vo move the corresponding parts up (down if negative). The units are approximately 1.5*strheight("Wj").

If any of the offset arguments are not specified then the corresponding element of the list "blockrand.text" is used if it exists otherwise they are 0.

The idea of the "blockrand.text" list is to set common defaults for your system (the default positions work for me, but you may want to tweak things for your system) including templates that are commonly used in your institution. Individual pieces can then be overridden with the function arguments. You can have a list saved with your defaults and pass that list to the blockrand.text argument.

Value

This function does not return anything useful, it is run for the side effect of creating a pdf file. The pdf file will have 4 cards per page and 1 card for each line of x.

Note

Adobe Acrobat (and possibly other pdf viewers) will often try to rescale the page when printing, for best results turn this feature off before printing.

Author(s)

Greg Snow <538280@gmail.com >

Examples

```
## stratified by sex, 100 in stratum, 2 treatments
male <- blockrand(n=100, id.prefix='M', block.prefix='M',stratum='Male')
female <- blockrand(n=100, id.prefix='F', block.prefix='F',stratum='Female')

my.study <- rbind(male,female)

## Not run:
plotblockrand(my.study,'mystudy.pdf',
  top=list(text=c('My Study','Patient: %ID%', 'Treatment: %TREAT%'),
    col=c('black','black','red'),font=c(1,1,4)),
  middle=list(text=c("My Study", "Sex: %STRAT%", "Patient: %ID%"),
```

```
        col=c('black','blue','green'),font=c(1,2,3)),
bottom="Call 123-4567 to report patient entry",
cut.marks=TRUE)

### or

my.blockrand.text <- list(
  top=list(text=c('My Study','Patient: %ID%','Treatment: %TREAT%'),
    col=c('black','black','red'),font=c(1,1,4)),
  middle=list(text=c("My Study","Sex: %STRAT%","Patient: %ID%"),
    col=c('black','blue','green'),font=c(1,2,3)),
  bottom="Call 123-4567 to report patient entry")

plotblockrand(my.study, 'mystudy.pdf', blockrand.text=my.blockrand.text,
  cut.marks=TRUE)

## End(Not run)
```


Index

* **datagen**

blockrand, 3

plotblockrand, 5

* **design**

blockrand, 3

plotblockrand, 5

* **distribution**

blockrand, 3

* **package**

blockrand-package, 2

blockrand, 3

blockrand-package, 2

plotblockrand, 4, 5