Package 'consort'

September 10, 2021

Type Package

Title Create Consort Diagram

Version 0.2.0

Description To make it easy to create CONSORT diagrams for the transparent reporting of participant allocation in randomized, controlled clinical trials. This is done by creating a standardized disposition data, and using this data as the source for the creation a standard CONSORT diagram. Human effort by supplying text labels on the node can also be achieved.

License MIT + file LICENSE

URL https://github.com/adayim/consort/

BugReports https://github.com/adayim/consort/issues

Encoding UTF-8

Imports Gmisc (>= 2.1.0), grid

Suggests testthat (>= 3.0.0), knitr, rmarkdown, covr

VignetteBuilder knitr

RoxygenNote 7.1.1

Config/testthat/edition 3

NeedsCompilation no

Author Alim Dayim [aut, cre] (<https://orcid.org/0000-0001-9998-7463>)

Maintainer Alim Dayim <ad938@cam.ac.uk>

Repository CRAN

Date/Publication 2021-09-10 04:20:02 UTC

R topics documented:

`consort-package` *Create Consort diagram*

Description

To make it easy to create CONSORT diagrams for the transparent reporting of participant allocation in randomized, controlled clinical trials. This is done by creating a standardized disposition data, and using this data as the source for the creation a standard CONSORT diagram. Human effort by supplying text labels on the node can also be achieved.

`add_box` *Add nodes*

Description

Create/add vertically aligned labeled nodes or side nodes.

Usage

```
add_box(prev_box = NULL, txt, just = "center", dist = 0.02)
```

Arguments

`prev_box`	Previous node object, the created new node will be vertically aligned with this node. Left this as 'NULL' if this is the first node. The first node will be aligned in the top center.
`txt`	Text in the node. If the 'prev_box' is a horizontally aligned multiple nodes, a vector of with the same length must be provided.
`just`	The justification for the text: left, center or right.
`dist`	Distance between previous node, including the distance between the side node.

Value

A `consort.list` or `consort` object.

See Also

`add_side_box`,`add_split`

Examples

```
txt1 <- "Population (n=300)"
txt1_side <- "Excluded (n=15): \n
              \u2022 MRI not collected (n=3)\n
              \u2022 Tissues not collected (n=4)\n
              \u2022 Other (n=8)"

node1 <- add_box(txt = txt1)

node3 <- add_side_box(node1, txt = txt1_side)

node4 <- add_box(node3, txt = "Randomized (n=200)")

node1_sp <- add_split(node4, txt = c("Arm A (n=100)", "Arm B (n=100")))
side1_sp <- add_side_box(node1_sp,
                         txt = c("Excluded (n=15):\n
                         \u2022 MRI not collected (n=3)\n
                         \u2022 Tissues not collected (n=4)\n
                         \u2022 Other (n=8)",
                         "Excluded (n=15):\n
                         \u2022 MRI not collected (n=3)\n
                         \u2022 Tissues not collected (n=4)"))

node2_sp <- add_box(side1_sp,
                    txt = c("Final analysis (n=100)",
                            "Final analysis (n=100"))

node1
node3
node4
node1_sp
side1_sp
node2_sp
```

`add_label_box` *Add a vertically aligned label nodes on the left side.*

Description

In a consort diagram, this can be used to indicate different stage.

Usage

```
add_label_box(ref_box, txt)
```

Arguments

`ref_box`	Reference node to which the label will be horizontally aligned.
`txt`	Text in the node.

Value

A consort object.

Examples

```
txt1 <- "Population (n=300)"
txt1_side <- "Excluded (n=15): \n
              \u2022 MRI not collected (n=3)\n
              \u2022 Tissues not collected (n=4)\n
              \u2022 Other (n=8)"

node1 <- add_box(txt = txt1)

node3 <- add_side_box(node1, txt = txt1_side)

node4 <- add_box(node3, txt = "Randomized (n=200)")

node1_sp <- add_split(node4, txt = c("Arm A (n=100)", "Arm B (n=100"))
side1_sp <- add_side_box(node1_sp,
                         txt = c("Excluded (n=15):\n
                         \u2022 MRI not collected (n=3)\n
                         \u2022 Tissues not collected (n=4)\n
                         \u2022 Other (n=8)",
                         "Excluded (n=15):\n
                         \u2022 MRI not collected (n=3)\n
                         \u2022 Tissues not collected (n=4)"))

node2_sp <- add_box(side1_sp,
                    txt = c("Final analysis (n=100)",
                            "Final analysis (n=100"))#'
lab1 <- add_label_box(node1, txt = "Screening")
lab2 <- add_label_box(node4, txt = "Randomized")
lab3 <- add_label_box(node2_sp, txt = "Final analysis")
build_consort(list(node1, node3, node4, node1_sp, side1_sp, node2_sp),
              list(lab1, lab2, lab3))
```

---

add_side_box *Add a side node*

Description

Add an exclusion node on the right side. If the length of text label is two, then the first one will be aligned on the left and the second on the right. Otherwise, all the side nodes will be aligned on the right.

Usage

```
add_side_box(prev_box, txt, side = NULL, dist = 0.02)
```

Arguments

prev_box	Previous node object, the created new node will be aligned at the right bottom of the ‘prev_box‘.
txt	Text in the node. If the ‘prev_box‘ is a horizontally aligned multiple nodes, a vector of with the same length must be provided.
side	Position of the side box, ‘left‘ or ‘right‘ side of the terminal box. Will be aligned on the left and right side if only two groups, right otherwise.
dist	Distance between previous node, including the distance between the side node.

Value

A `consort.list` or `consort` object.

See Also

add_box,add_split

Examples

```
txt1 <- "Population (n=300)"
txt1_side <- "Excluded (n=15): \n
              \u2022 MRI not collected (n=3)\n
              \u2022 Tissues not collected (n=4)\n
              \u2022 Other (n=8)"

node1 <- add_box(txt = txt1)

node3 <- add_side_box(node1, txt = txt1_side)

node4 <- add_box(node3, txt = "Randomized (n=200)")

node1_sp <- add_split(node4, txt = c("Arm A (n=100)", "Arm B (n=100")))
side1_sp <- add_side_box(node1_sp,
                         txt = c("Excluded (n=15):\n
                         \u2022 MRI not collected (n=3)\n
                         \u2022 Tissues not collected (n=4)\n
                         \u2022 Other (n=8)",
                         "Excluded (n=15):\n
                         \u2022 MRI not collected (n=3)\n
                         \u2022 Tissues not collected (n=4)"))

node2_sp <- add_box(side1_sp,
                    txt = c("Final analysis (n=100)",
                            "Final analysis (n=100"))
node1
node3
node4
node1_sp
side1_sp
node2_sp
```

add_split *Add a splitting box*

Description

This function will create a horizontally aligned nodes. The horizontal coordinate will be automatically calculated if the coordinates not provided.

Usage

```
add_split(prev_box, txt, coords = NULL, dist = 0.02)
```

Arguments

prev_box	Previous node that the newly created split box will be aligned.
txt	A vector of text labels for each nodes.
coords	The horizontal coordinates of the boxes, see details.
dist	Distance between previous node, including the distance between the side node.

Details

The ‘coords‘ will be used to set the horizontal coordinates of the nodes. The ‘coords‘ should be within 0 and 1 to avoid the nodes is aligned outside of the final figure. If the ‘coords‘ is ‘NULL‘, not given. The function will calculate the ‘coords‘. If the the length of the ‘txt‘ is two, then a coordinates of 0.35 and 0.65 will be used. Once the split box is added, all the following nodes will be split accordingly.

Value

A `consort.list` object.

See Also

add_box, add_side_box

Examples

```
txt1 <- "Population (n=300)"
txt1_side <- "Excluded (n=15): \n
              \u2022 MRI not collected (n=3)\n
              \u2022 Tissues not collected (n=4)\n
              \u2022 Other (n=8)"

node1 <- add_box(txt = txt1)

node3 <- add_side_box(node1, txt = txt1_side)

node4 <- add_box(node3, txt = "Randomized (n=200)")
```

```
node1_sp <- add_split(node4, txt = c("Arm A (n=100)", "Arm B (n=100"))
side1_sp <- add_side_box(node1_sp,
                         txt = c("Excluded (n=15):\n
                         \u2022 MRI not collected (n=3)\n
                         \u2022 Tissues not collected (n=4)\n
                         \u2022 Other (n=8)",
                         "Excluded (n=15):\n
                         \u2022 MRI not collected (n=3)\n
                         \u2022 Tissues not collected (n=4)"))

node2_sp <- add_box(side1_sp,
                    txt = c("Final analysis (n=100)",
                            "Final analysis (n=100"))#'
node1
node3
node4
node1_sp
side1_sp
node2_sp
```

---

box_text *Generate label and bullet points*

Description

This function use the data to generate label and bullet points for the box.

Usage

```
box_text(x, label = NULL, bullet = FALSE)
```

Arguments

x	A list or a vector to be used.
label	A character string as a label at the beginning of the text label. The count for each categories will be returned if no label is provided.
bullet	If shows bullet points. If the value is 'TRUE', the bullet points will be tabulated, default is 'FALSE'.

Value

A character string of vector.

Examples

```
val <- data.frame(am = factor(ifelse(mtcars$am == 1, "Automatic", "Manual")),
                  car = row.names(mtcars))

 box_text(val$car, label = "Cars in the data")
 box_text(val$car, label = "Cars in the data", bullet = FALSE)
 box_text(split(val$car, val$am), label = "Cars in the data")
 box_text(split(val$car, val$am), label = "Cars in the data", bullet = FALSE)
```

`build_consort` *Build consort diagram*

Description

Build consort diagram

Usage

```
build_consort(consort_list, label_list = NULL)
```

Arguments

`consort_list`	A list of nodes.
`label_list`	A list of label nodes.

Value

A `consort.plot` object.

See Also

`add_side_box`,`add_split`, `add_side_box`

Examples

```
txt1 <- "Population (n=300)"
txt1_side <- "Excluded (n=15): \n
              \u2022 MRI not collected (n=3)\n
              \u2022 Tissues not collected (n=4)\n
              \u2022 Other (n=8)"

node1 <- add_box(txt = txt1)

node3 <- add_side_box(node1, txt = txt1_side)

node4 <- add_box(node3, txt = "Randomized (n=200)")
```

```
node1_sp <- add_split(node4, txt = c("Arm A (n=100)", "Arm B (n=100"))
side1_sp <- add_side_box(node1_sp,
                         txt = c("Excluded (n=15):\n
                         \u2022 MRI not collected (n=3)\n
                         \u2022 Tissues not collected (n=4)\n
                         \u2022 Other (n=8)",
                         "Excluded (n=15):\n
                         \u2022 MRI not collected (n=3)\n
                         \u2022 Tissues not collected (n=4)"))

node2_sp <- add_box(side1_sp,
                    txt = c("Final analysis (n=100)",
                            "Final analysis (n=100"))

lab1 <- add_label_box(node1, txt = "Screening")
lab2 <- add_label_box(node4, txt = "Randomized")
lab3 <- add_label_box(node2_sp, txt = "Final analysis")
build_consort(list(node1, node3, node4, node1_sp, side1_sp, node2_sp),
              list(lab1, lab2, lab3))
```

`consort_plot` *Self generating consort diagram*

Description

Create CONSORT diagram from a participant disposition data.

Usage

```
consort_plot(
  data,
  orders,
  side_box,
  allocation = NULL,
  labels = NULL,
  coords = NULL,
  dist = 0.02,
  cex = 0.8
)
```

Arguments

`data`	Data set with disposition information for each participants.
`orders`	A named vector or a list, names as the variable in the dataset and values as labels in the box. The order of the diagram will be based on this.
`side_box`	Variable vector, appeared as side box in the diagram. The next box will be the subset of the missing values of these variables.

allocation	Name of the grouping/treatment variable (optional), the diagram will split into branches on this variables forward.
labels	Named vector, names is the location of the vertical node excluding the side box. The position location should plus 1 after the allocation variables if the allocation is defined.
coords	The horizontal coordinates of the boxes, see details.
dist	Optional, distance between boxes. Default is 0.02.
cex	Multiplier applied to font size, Default is 0.8

Details

The calculation of numbers is as in an analogous to Kirchhoff's Laws of electricity. The numbers in terminal nodes must sum to those in the ancestor nodes. All the drop outs will be populated as a side box. Which was different from the official CONSORT diagram template, which has dropout inside a vertical node.

Value

A `consort.plot` object.

See Also

`add_side_box`,`add_split`, `add_side_box`,`build_consort`

Examples

```
## Prepare test data
set.seed(1001)
N <- 300

trialno <- sample(c(1000:2000), N)
exc1 <- rep(NA, N)
exc1[sample(1:N, 15)] <- sample(c("Sample not collected", "MRI not collected", "Other"),
                                15,
                                replace = TRUE, prob = c(0.4, 0.4, 0.2))

induc <- rep(NA, N)
induc[is.na(exc1)] <- trialno[is.na(exc1)]

exc2 <- rep(NA, N)
exc2[sample(1:N, 20)] <- sample(c("Sample not collected", "Dead",
                                  "Other"), 20, replace = TRUE,
                                  prob = c(0.4, 0.4, 0.2))
exc2[is.na(induc)] <- NA

exc <- ifelse(is.na(exc2), exc1, exc2)

arm <- rep(NA, N)
arm[is.na(exc)] <- sample(c("Conc", "Seq"), sum(is.na(exc)), replace = TRUE)
arm3 <- sample(c("Trt A", "Trt B", "Trt C"), N, replace = TRUE)
```

```
arm3[is.na(arm)] <- NA

fow1 <- rep(NA, N)
fow1[!is.na(arm)] <- sample(c("Withdraw", "Discontinued", "Death", "Other", NA),
                            sum(!is.na(arm)), replace = TRUE,
                            prob = c(0.05, 0.05, 0.05, 0.05, 0.8))
fow2 <- rep(NA, N)
fow2[!is.na(arm) & is.na(fow1)] <- sample(c("Protocol deviation", "Outcome missing", NA),
                                          sum(!is.na(arm) & is.na(fow1)), replace = TRUE,
                                           prob = c(0.05, 0.05, 0.9))


df <- data.frame(trialno, exc1, induc, exc2, exc, arm, arm3, fow1, fow2)
rm(trialno, exc1, induc, exc2, exc, arm, arm3, fow1, fow2, N)

## Single arm
out <- consort_plot(data = df,
order = c(trialno = "Population",
          exc1    = "Excluded",
          arm     = "Allocated",
          fow1    = "Lost of Follow-up",
          trialno = "Finished Followup",
          fow2    = "Not evaluable for the final analysis",
          trialno = "Final Analysis"),
side_box = c("exc1", "fow1", "fow2"),
cex = 0.9)

## Two arms
out <- consort_plot(data = df,
             order = c(trialno = "Population",
                       exc    = "Excluded",
                       arm     = "Randomized patient",
                       fow1    = "Lost of Follow-up",
                       trialno = "Finished Followup",
                       fow2    = "Not evaluable",
                       trialno = "Final Analysis"),
             side_box = c("exc", "fow1", "fow2"),
             allocation = "arm",
             labels = c("1" = "Screening", "2" = "Randomization",
                        "5" = "Final"))
## Three arms
consort_plot(data = df,
             order = c(trialno = "Population",
                       exc    = "Excluded",
                       arm3     = "Randomized patient",
                       fow1    = "Lost of Follow-up",
                       trialno = "Finished Followup",
                       fow2    = "Not evaluable",
                       trialno = "Final Analysis"),
             side_box = c("exc", "fow1", "fow2"),
             allocation = "arm3",
             labels = c("1" = "Screening", "2" = "Randomization",
                        "5" = "Final"))
```

```
## Multiple phase
consort_plot(data = df,
             order = list(trialno = "Population",
                          exc1    = "Excluded",
                          induc   = "Induction",
                          exc2    = "Excluded",
                          arm3     = "Randomized patient",
                          fow1    = "Lost of Follow-up",
                          trialno = "Finished Followup",
                          fow2    = "Not evaluable",
                          trialno = "Final Analysis"),
             side_box = c("exc1", "exc2", "fow1", "fow2"),
             allocation = "arm3",
             labels = c("1" = "Screening", "2" = "Month 4",
                        "3" = "Randomization", "5" = "Month 24",
                        "6" = "End of study"),
             dist = 0.02,
             cex = 0.7)
```

`print.consort` *Print Consort Plots*

Description

This method is to support saving plots with 'ggplot2::ggsave'.

Usage

```
## S3 method for class 'consort.list'
print(x, ...)

## S3 method for class 'consort'
print(x, ...)

## S3 method for class 'consort.plot'
grid.draw(x)
```

Arguments

`x`	A consort.plot object.
`...`	further arguments passed to `print` or `grid::grid.draw`.

Value

None

`print.consort.plot` *Add methods to print function*

Description

Method for create viewports for the `consort.list`, `consort.plot` or `consort` objects and display the output in on a grid device.

Usage

```
## S3 method for class 'consort.plot'
print(x, ...)
```

Arguments

`x`	A `consort.list`, `consort.plot` or `consort`.
`...`	Not used.

Value

None.

See Also

`add_side_box`,`add_split`, `add_side_box`, `consort_plot`,`build_consort`

Index