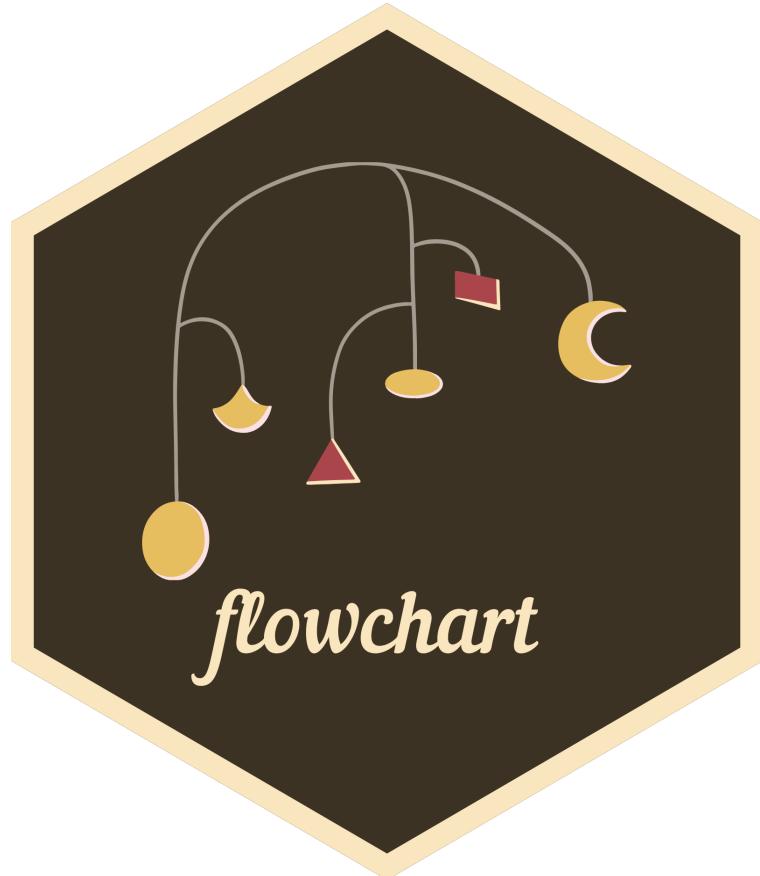


{flowchart}: an R package for creating participant flow diagrams integrated with tidyverse



Pau Satorra

Germans Trias i Pujol Research Institute
and Hospital (IGTP)
Badalona, Spain

July 11, 2024

Introduction

Motivation

- In any study, the flow of subjects must be clear and the process transparent, especially in health research studies
- The CONSORT, STROBE and ICH guidelines reflect this need
- The preferred way to present this patient flow through the different phases is a flowchart (also called flow diagram)
- The creation of the flowchart is a joint task between researchers, data management team and statisticians

Motivation

- There are several R packages dedicated to building flowcharts: {Gmisc}, {DiagrammeR}, {consort}, {ggflowchart}
- Complex programming and manual parameterization are often involved
- Some are designed for building other kind of diagrams

{flowchart} package

- Creates reproducible flowcharts from a dataset in an easy way
- Provides a set of functions that can be combined with the pipe operator (|> or %>%)

flowchart: Tidy Flowchart Generator

Creates participant flow diagrams directly from a data frame. Representing the flow of participants through each stage of a study, especially in clinical trials, is essential to assess the generalisability and validity of the results. This package provides a set of functions that can be combined with a pipe operator to create all kinds of flowcharts from a data frame in an easy way.

Version: 0.4.0
Depends: R (\geq 4.1.0)
Imports: [Gmisc](#), grid, [tidyverse](#), [purrr](#), [stringr](#), [tibble](#), [tidyselect](#), [rlang](#), grDevices
Suggests: [knitr](#), [rmarkdown](#)
Published: 2024-05-30
Author: Pau Satorra [aut, cre], João Carmezim [aut], Natália Pallarès [aut], Cristian Tebé [aut]
Maintainer: Pau Satorra <psatorra at igtp.cat>
BugReports: <https://github.com/bruijtp/flowchart/issues>
License: MIT + file [LICENSE](#)
URL: <https://bruijtp.github.io/flowchart/>
NeedsCompilation: no
Materials: [README NEWS](#)
CRAN checks: [flowchart results](#)

Documentation:

Reference manual: [flowchart.pdf](#)
Vignettes: [flowchart](#)

Downloads:

Package source: [flowchart_0.4.0.tar.gz](#)
Windows binaries: r-devel: [flowchart_0.3.0.zip](#), r-release: [flowchart_0.3.0.zip](#), r-oldrel: [flowchart_0.3.0.zip](#)
macOS binaries: r-release (arm64): [flowchart_0.3.0.tgz](#), r-oldrel (arm64): [flowchart_0.3.0.tgz](#), r-release (x86_64): [flowchart_0.3.0.tgz](#), r-oldrel (x86_64): [flowchart_0.3.0.tgz](#)
Old sources: [flowchart archive](#)

Linking:

Please use the canonical form <https://CRAN.R-project.org/package=flowchart> to link to this page.

flowchart CRAN page
useR! 2024 - {flowchart}

Overview

- Create a flowchart
 - `as_fc()`
 - `fc_draw()`
 - `fc_split()`
 - `fc_filter()`
- Customize flowcharts
 - `fc_modify()`
- Combine flowcharts
 - `fc_merge()`
 - `fc_stack()`
- Export flowcharts
 - `fc_export()`

safo dataset

- Built-in dataset
- Randomly generated dataset from the SAFO clinical trial¹

ID	did not meet inclusion criteria	met exclusion criteria	declined to participate	treatment received	intention to treat (ITT)	per protocol (PP)
1	Yes	No	NA	NA	NA	NA
2	No	No	Yes	NA	NA	NA
3	No	No	No	cloxacillin plus fosfomycin	Yes	Yes
4	No	Yes	NA	NA	NA	NA
5	No	No	No	cloxacillin plus fosfomycin	Yes	Yes
6	No	Yes	NA	NA	NA	NA

Create a flowchart

as_fc()

- Allows to initialize a dataset in the class `fc` created for this package
- Creates a flowchart with an initial box showing the total number of rows of the dataset

```
library(flowchart)

safo_fc <- safo |>
  as_fc()
```

as_fc()

- Allows to initialize a dataset in the class `fc` created for this package
- Creates a flowchart with an initial box showing the total number of rows of the dataset

```
library(flowchart)

safo_fc <- safo |>
  as_fc()

str(safo_fc, max.level = 1)
```

```
List of 2
$ data: tibble [925 × 21] (S3: tbl_df/tbl/data.frame)
$ fc   : tibble [1 × 16] (S3: tbl_df/tbl/data.frame)
- attr(*, "class")= chr "fc"
```

as_fc()

`safo_fc$fc`

id	x	y	n	N	perc	text	type	group	just	text_color	text_fs	text_fface	text_ffamily	bg_fill	border_color	
Initial																
1	0.5	0.5	925	925	100	dataframe	init	NA	center	black	8		1	NA	white	black
						925										

fc_draw()

- Allows to draw a previously created `fc` object

```
safo |>  
  as_fc()
```

fc_draw()

- Allows to draw a previously created `fc` object

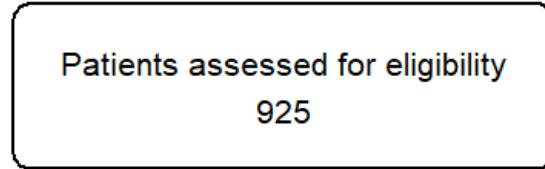
```
safo |>  
  as_fc() |>  
  fc_draw()
```

Initial dataframe
925

fc_draw()

- Allows to draw a previously created `fc` object

```
safo |>  
  as_fc(label = "Patients assessed for eligibility") |>  
  fc_draw()
```



Patients assessed for eligibility
925

- We can use the `label` argument to modify the box label

fc_filter()

- We can filter an existing flowchart specifying the logic in which the filter is to be applied

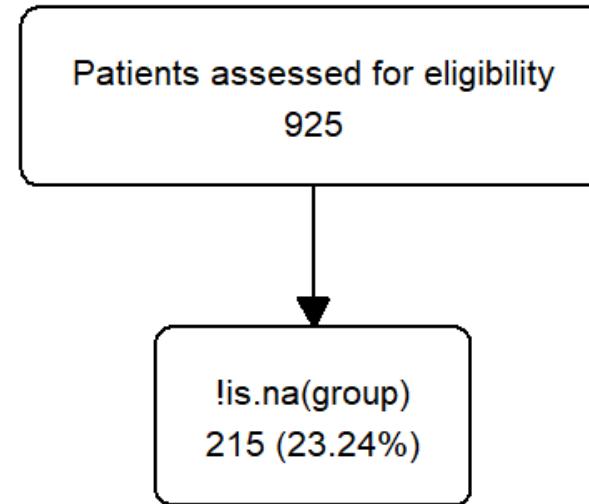
```
safo |>  
  as_fc(label = "Patients  
assessed for eligibility") |>  
  fc_draw()
```

Patients assessed for eligibility
925

fc_filter()

- We can filter an existing flowchart specifying the logic in which the filter is to be applied

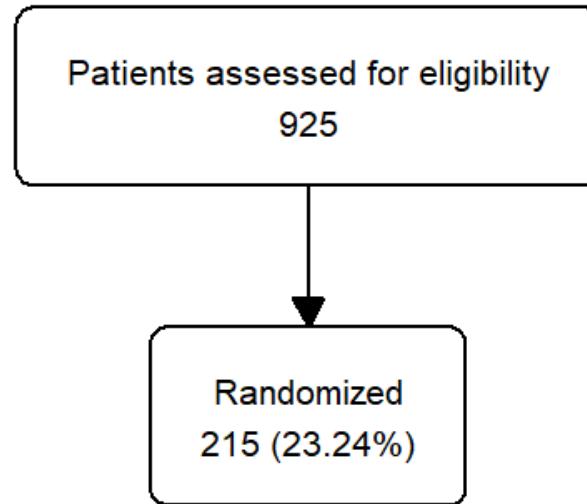
```
safo |>  
  as_fc(label = "Patients  
assessed for eligibility") |>  
    fc_filter(!is.na(group))  
  
|>  
  fc_draw()
```



fc_filter()

- We can filter an existing flowchart specifying the logic in which the filter is to be applied

```
safo |>  
  as_fc(label = "Patients  
assessed for eligibility") |>  
    fc_filter(!is.na(group),  
label = "Randomized") |>  
      fc_draw()
```

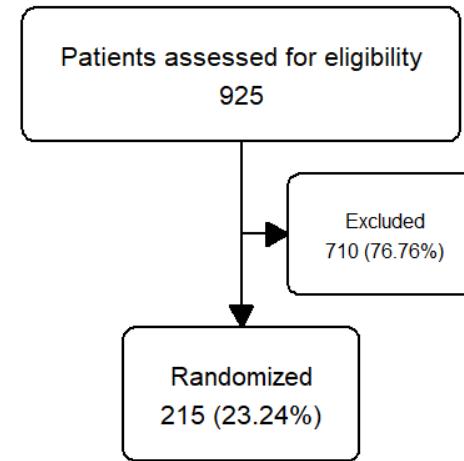


- We can change again the label

fc_filter()

- We can filter an existing flowchart specifying the logic in which the filter is to be applied

```
safo |>  
  as_fc(label = "Patients  
assessed for eligibility") |>  
    fc_filter(!is.na(group),  
label = "Randomized", show_exc =  
TRUE) |>  
    fc_draw()
```

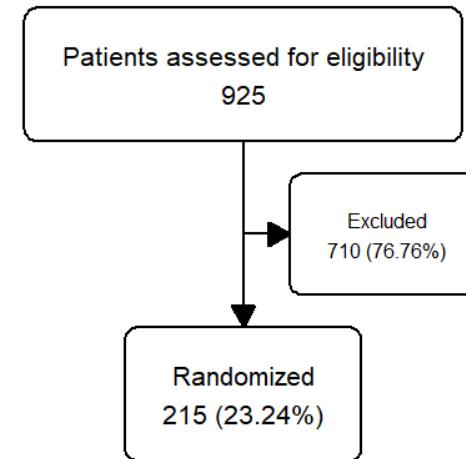


- We can change again the label
- We can use `show_exc=TRUE` to show the excluded rows

fc_split()

- We can split an existing flowchart according to the different values of a column

```
safo |>  
  as_fc(label = "Patients  
assessed for eligibility") |>  
    fc_filter(!is.na(group),  
label = "Randomized", show_exc =  
TRUE) |>  
    fc_draw()
```



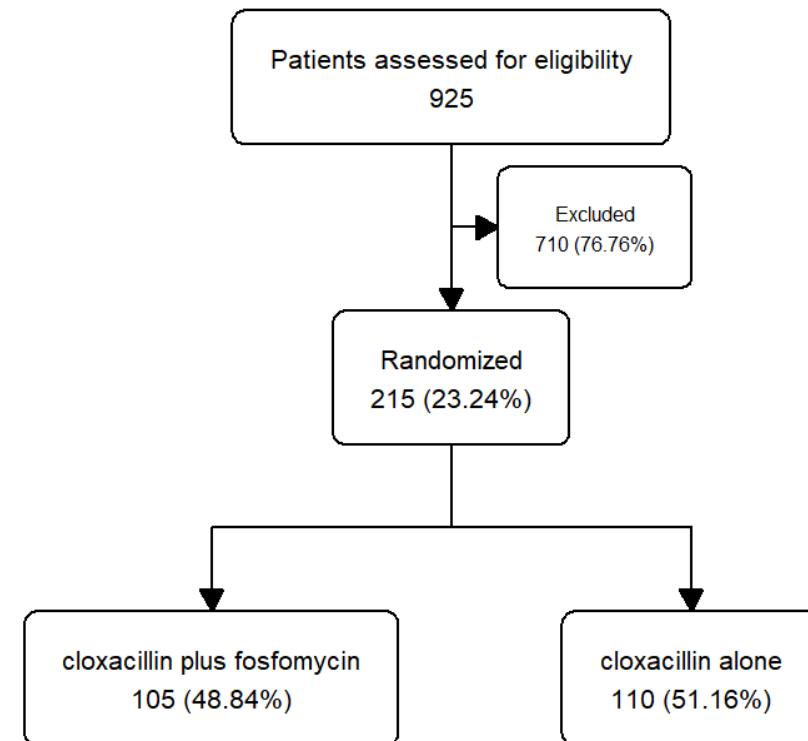
`fc_split()`

- We can split an existing flowchart according to the different values of a column

```

safo |>
  as_fc(label = "Patients
assessed for eligibility") |>
  fc_filter(!is.na(group),
label = "Randomized", show_exc =
TRUE) |>
  fc_split(group) |>
  fc_draw()

```



Customize flowcharts

Modify function arguments

- Some arguments common to `as_fc()`, `fc_filter()` and `fc_split()`, to customise the appearance of the boxes created at each step

`label=` modify the label.

`text_pattern=` modify the pattern of the text (e.g. `{label}\n {n} ({perc}%)`).

`just=` modify the justification for the text.

`text_color=` modify the color of the text.

`text_fs=` modify the font size of the text.

`bg_fill=` modify the background color of the box.

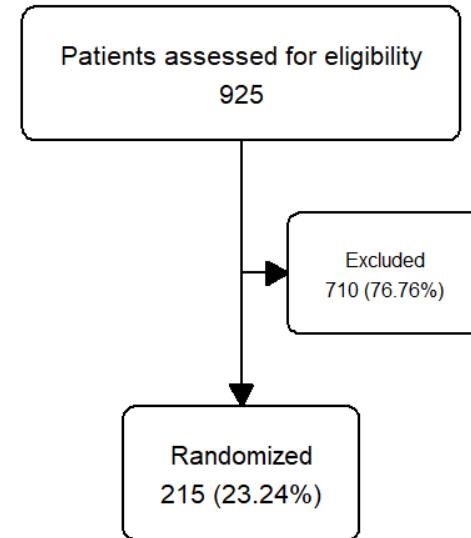
`border_color=` modify the border color of the box.

- Other arguments specific to each function (vignette)

fc_modify()

- We can modify the parameters of the created flowchart

```
safo |>  
  as_fc(label = "Patients  
assessed for eligibility") |>  
  fc_filter(!is.na(group),  
label = "Randomized", show_exc =  
TRUE) |>  
  fc_draw()
```



fc_modify()

- We can modify the parameters of the created flowchart

```
safo_fc <- safo |>  
  as_fc(label = "Patients assessed for eligibility") |>  
  fc_filter(!is.na(group), label = "Randomized", show_exc = TRUE)  
  
safo_fc$fc
```

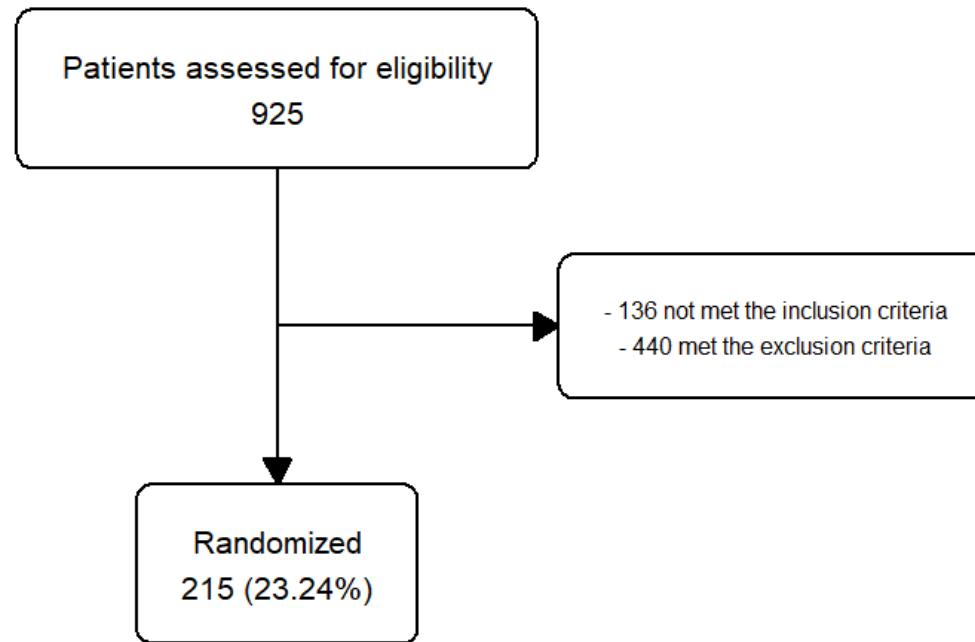
id	x	y	n	N	perc	text	type	group	just	text_color	text_fs	text_fface	text_ffamily	bg_fill	border_color
1	0.50	0.6666667	925	925	100	Patients assessed for eligibility 925	init	NA	center	black	8	1	NA	white	black
2	0.50	0.3333333	215	925	23.24	Randomized 215 (23.24%)	filter	NA	center	black	8	1	NA	white	black
3	0.65	0.5000000	710	925	76.76	Excluded 710 (76.76%)	exclude	NA	center	black	6	1	NA	white	black

fc_modify()

- We can modify the parameters of the created flowchart

```
safo |>
  as_fc(label = "Patients assessed for eligibility") |>
  fc_filter(!is.na(group), label = "Randomized", show_exc = TRUE) |>
  fc_modify(
    ~ . |>
      mutate(
        text = ifelse(id == 3, str_glue("- {sum(safo$inclusion_crit
== 'Yes')} not met the inclusion criteria\n- {sum(safo$exclusion_crit ==
'Yes')} met the exclusion criteria"), text),
        x = ifelse(id == 3, 0.75, x)
      )
  ) |>
  fc_draw()
```

fc_modify()

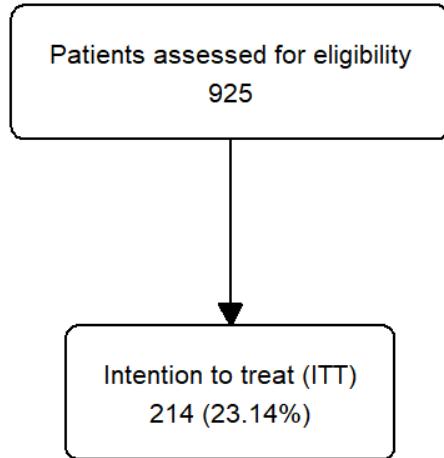


Combine flowcharts

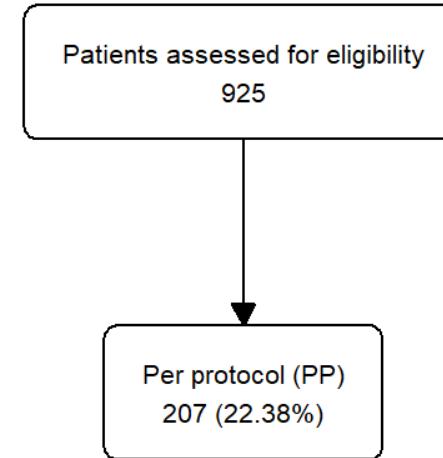
`fc_merge()`

- We can combine different flowcharts horizontally

```
fc1 <- safo |>  
  as_fc(label = "Patients  
assessed for eligibility") |>  
  fc_filter(itt == "Yes",  
label = "Intention to treat (ITT)")  
  
fc_draw(fc1)
```



```
fc2 <- safo |>  
  as_fc(label = "Patients  
assessed for eligibility") |>  
  fc_filter(pp == "Yes",  
label = "Per protocol (PP)")  
  
fc_draw(fc2)
```



fc_merge()

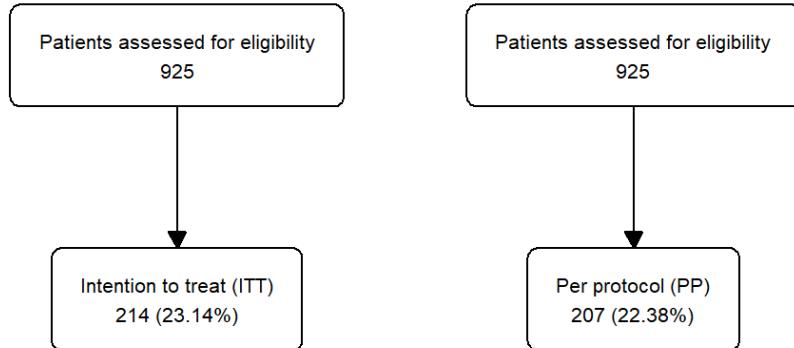
- We can combine different flowcharts horizontally

```
list(fc1, fc2) |>  
  fc_merge()
```

fc_merge()

- We can combine different flowcharts horizontally

```
list(fc1, fc2) |>  
  fc_merge() |>  
  fc_draw()
```



Export flowcharts

fc_export()

- We can export the drawn flowchart to some of the most popular graphic devices: **png**, **jpeg**, **tiff** and **bmp**

```
safo |>
  as_fc(label = "Patients assessed for eligibility") |>
  fc_filter(!is.na(group), label = "Randomized", show_exc = TRUE) |>
  fc_draw()
```

fc_export()

- We can export the drawn flowchart to some of the most popular graphic devices: `png`, `jpeg`, `tiff` and `bmp`

```
safo |>
  as_fc(label = "Patients assessed for eligibility") |>
  fc_filter(!is.na(group), label = "Randomized", show_exc = TRUE) |>
  fc_draw() |>
  fc_export("flowchart.png")
```

fc_export()

- We can export the drawn flowchart to some of the most popular graphic devices: **png**, **jpeg**, **tiff** and **bmp**

```
safo |>
  as_fc(label = "Patients assessed for eligibility") |>
  fc_filter(!is.na(group), label = "Randomized", show_exc = TRUE) |>
  fc_draw() |>
  fc_export("flowchart.png", width = 2500, height = 2000, res = 700)
```

- We can customize the size and resolution of the image to save

Hands-on examples

Example 1

- We will try to build a flowchart for the complete participant flow of the SAFO study trial

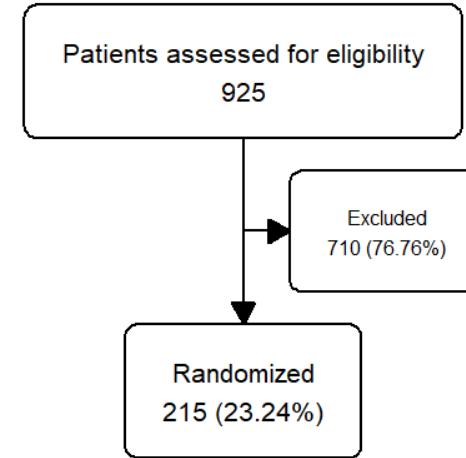
Example 1

```
safo |>  
  as_fc(label = "Patients  
assessed for eligibility") |>  
  fc_draw()
```

Patients assessed for eligibility
925

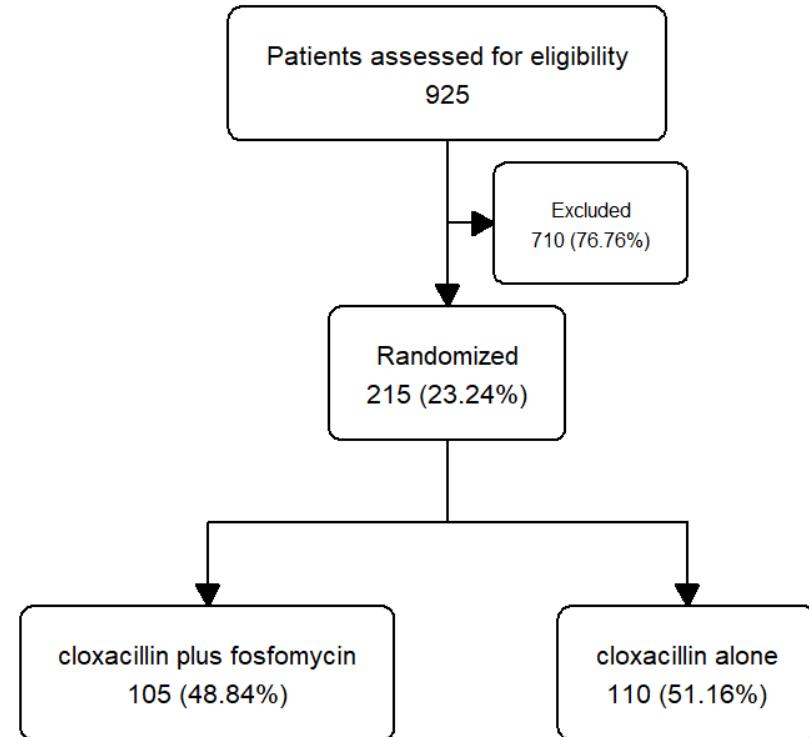
Example 1

```
safo |>  
  as_fc(label = "Patients  
assessed for eligibility") |>  
  fc_filter(!is.na(group),  
label = "Randomized", show_exc =  
TRUE) |>  
  fc_draw()
```



Example 1

```
safo |>  
  as_fc(label = "Patients  
assessed for eligibility") |>  
  fc_filter(!is.na(group),  
label = "Randomized", show_exc =  
TRUE) |>  
  fc_split(group) |>  
  fc_draw()
```

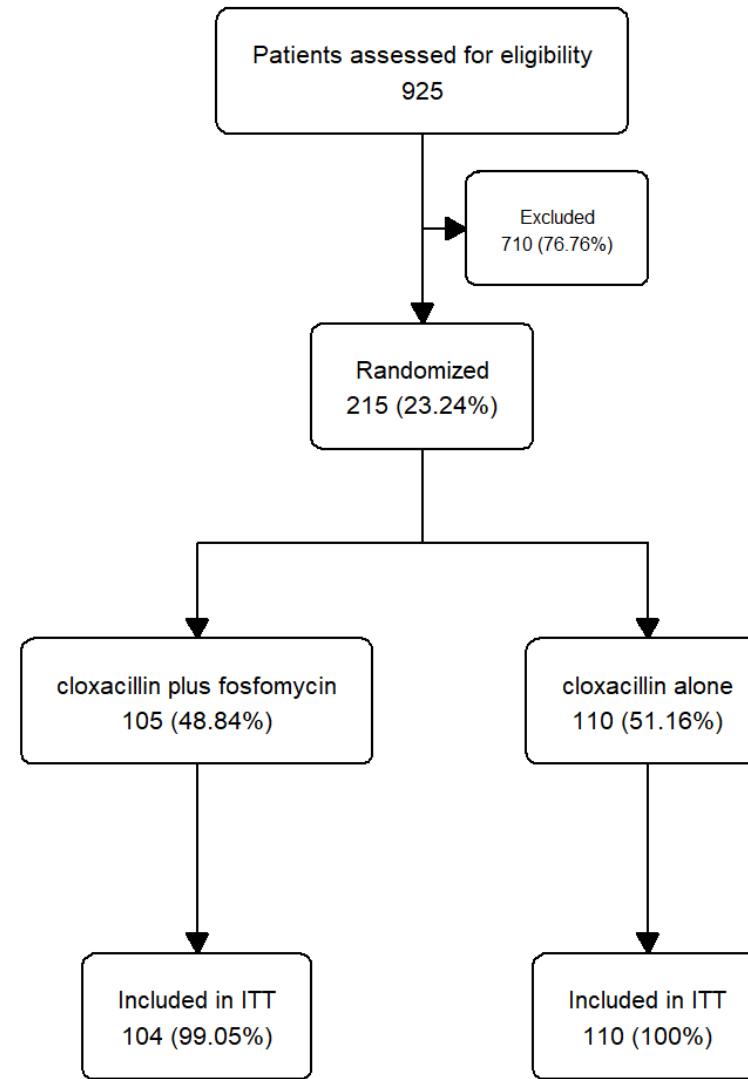


Example 1

```

safo |>
  as_fc(label = "Patients
assessed for eligibility") |>
  fc_filter(!is.na(group),
label = "Randomized", show_exc =
TRUE) |>
  fc_split(group) |>
  fc_filter(itt == "Yes",
label = "Included in ITT") |>
  fc_draw()

```

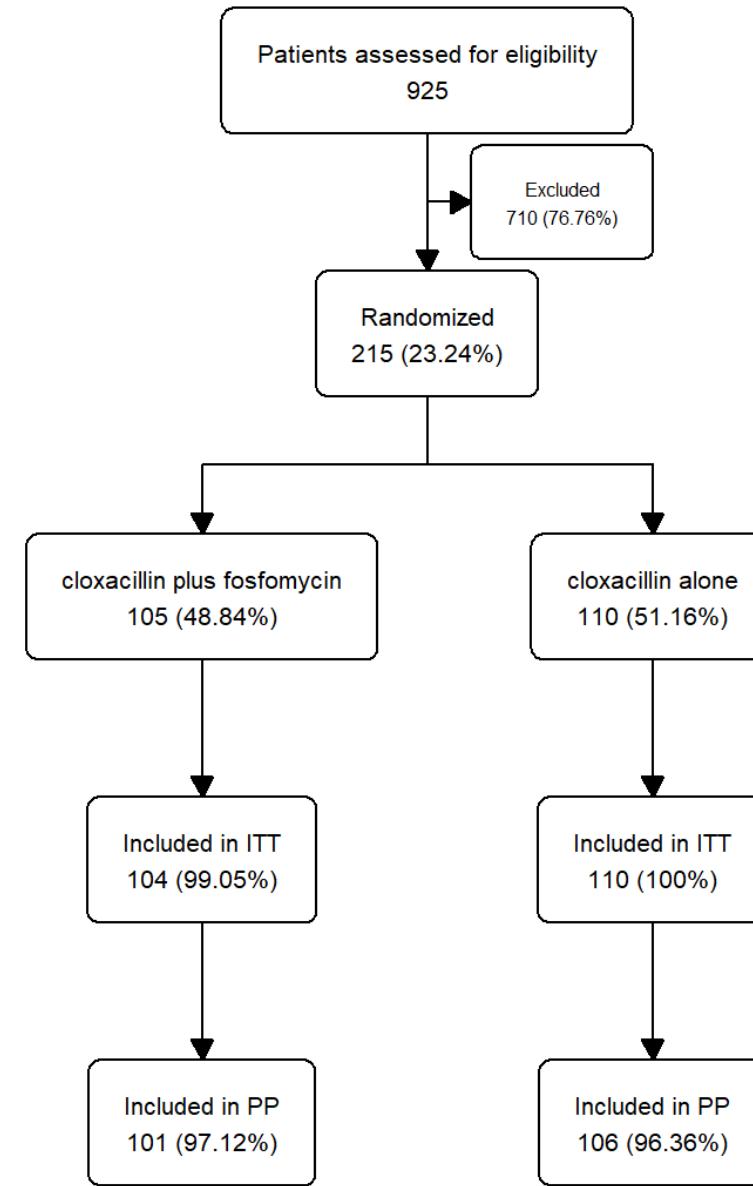


Example 1

```

safo |>
  as_fc(label = "Patients
assessed for eligibility") |>
  fc_filter(!is.na(group),
label = "Randomized", show_exc =
TRUE) |>
  fc_split(group) |>
  fc_filter(itt == "Yes",
label = "Included in ITT") |>
  fc_filter(pp == "Yes",
label = "Included in PP") |>
  fc_draw()

```



Example 1

- In the vignette there is the full example to exactly reproduce the flowchart found in the SAFO article:

Grillo, S., Pujol, M., Miró, J.M. et al. Cloxacillin plus fosfomycin versus cloxacillin alone for methicillin-susceptible *Staphylococcus aureus* bacteremia: a randomized trial. Nat Med 29, 2518–2525 (2023).

<https://doi.org/10.1038/s41591-023-02569-0>

Example 2

- Now, we will create a flowchart without any dataset using the `N=` argument

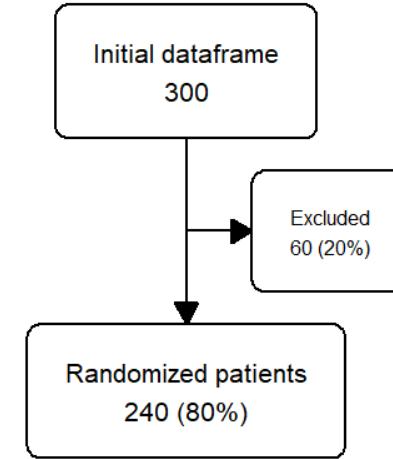
Example 2

```
as_fc(N = 300) |>  
  fc_draw()
```

Initial dataframe
300

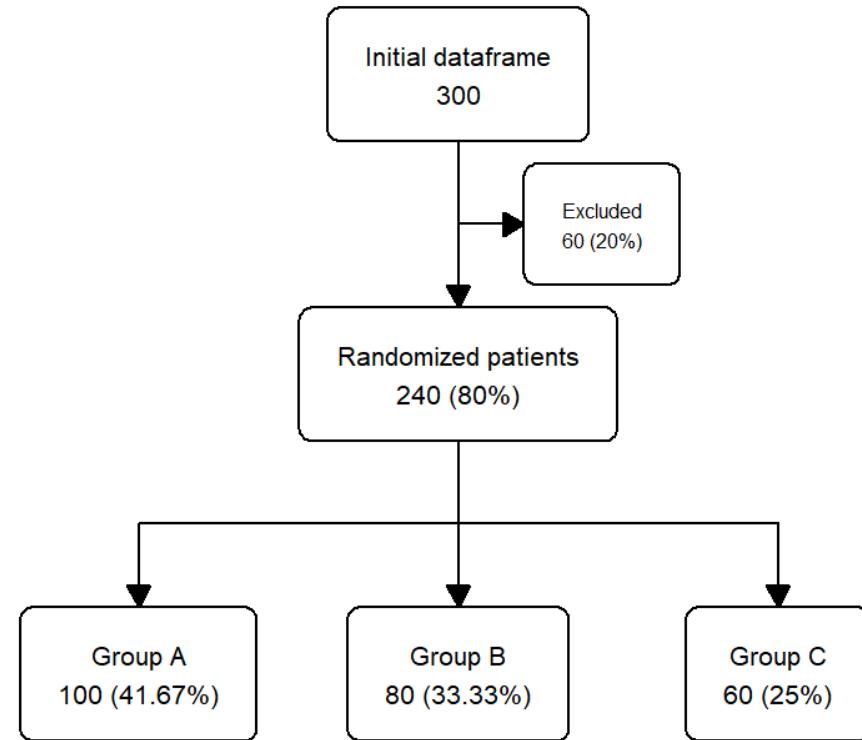
Example 2

```
as_fc(N = 300) |>  
  fc_filter(N = 240, label =  
"Randomized patients", show_exc =  
TRUE) |>  
  fc_draw()
```



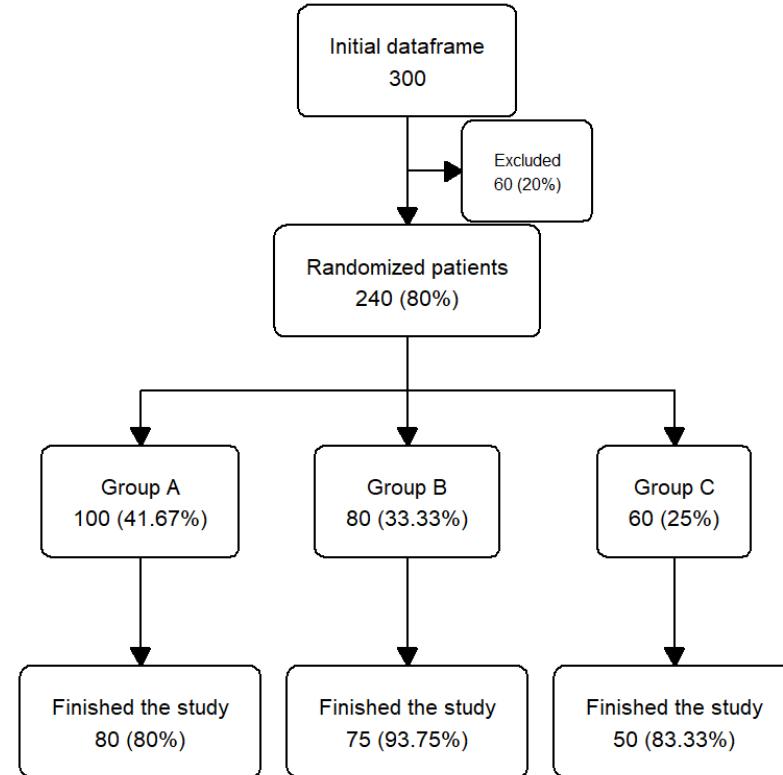
Example 2

```
as_fc(N = 300) |>  
  fc_filter(N = 240, label =  
"Randomized patients", show_exc =  
TRUE) |>  
  fc_split(N = c(100, 80,  
60), label = c("Group A", "Group B",  
"Group C")) |>  
  fc_draw()
```



Example 2

```
as_fc(N = 300) |>  
  fc_filter(N = 240, label =  
"Randomized patients", show_exc =  
TRUE) |>  
  fc_split(N = c(100, 80,  
60), label = c("Group A", "Group B",  
"Group C")) |>  
  fc_filter(N = c(80, 75,  
50), label = "Finished the study")  
|>  
  fc_draw()
```



Summary

Conclusions

- A clear and detailed reporting of the flow of participants in health research studies is required and recommended
- With this package, flowchart programming in R is made easier and accessible within the tidyverse workflow
- Flowchart reproducibility is assured
- As a limitation, we have not considered all possible scenarios and study designs, although it is highly customizable
- As future developments:
 - Define style themes
 - Shiny application

More information

- You can visit the {flowchart} website: <https://bruigtp.github.io/flowchart/>

flowchart 0.4.0 Get started Reference Changelog

flowchart



Links
[View on CRAN](#)
[Browse source code](#)
[Report a bug](#)

License
[Full license](#)
[MIT + file LICENSE](#)

Citation
[Citing flowchart](#)

Developers

Pau Satorra	Author, maintainer
João Carmezim	Author
Natàlia Pallarès	Author
Cristian Tebé	Author

CRAN 0.4.0 R-CMD-check passing downloads 627/month downloads 1868

Tidy Flowchart Generator

`flowchart` is an R package for drawing participant flow diagrams directly from a dataframe using tidyverse. It provides a set of functions that can be combined with a pipe operator to create all kinds of flowcharts from a dataframe in an easy way.

You can see the package in action in: <https://bruigtp.github.io/flowchart/>

How to install it?

The package is available on CRAN: <https://cran.r-project.org/web/packages/flowchart/index.html>.

```
install.packages("flowchart")
```

We can download the development version from the github repository:

```
# install.packages("remotes")
remotes::install_github('bruigtp/flowchart')
```

Contact

IGTP Biostatistics Support and Research Unit:

Pau Satorra

Author, maintainer

psatorra@igtp.cat

João Carmezim

Author

Natàlia Pallarès

Author

Cristian Tebé

Author

 github.com/bruigtp

[flowchart](#)

Public

flowchart is an R package for drawing participant flow diagrams directly from a dataframe using tidyverse.

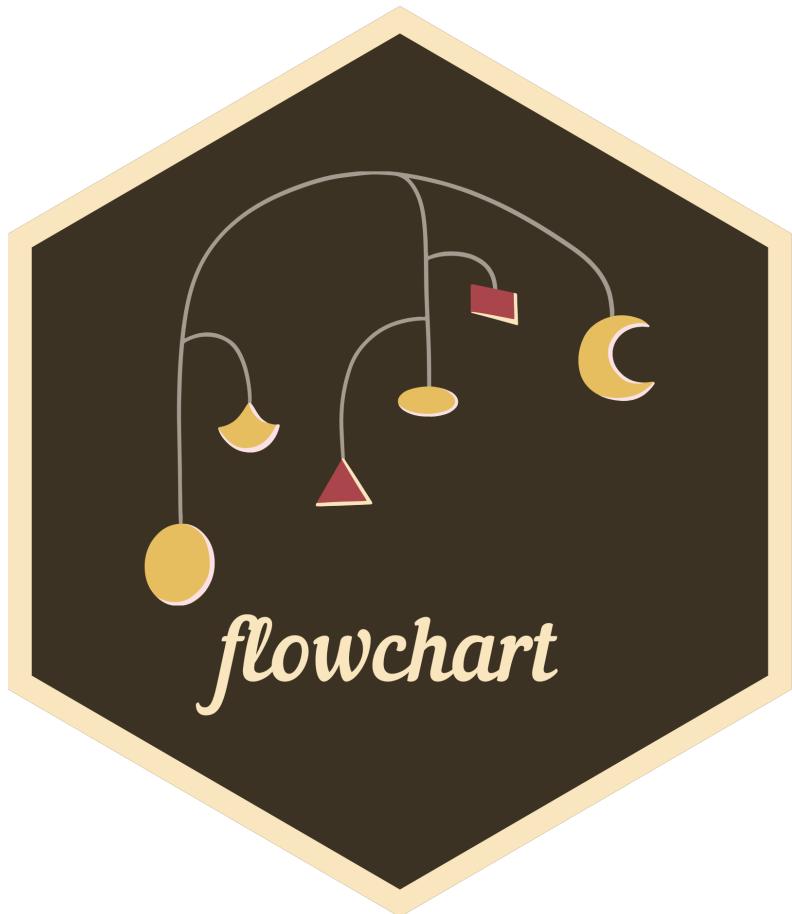
 R  34  1

[flowchart-an-R-package-for-creating-participant-flow-diagrams-integrated-with-tidyverse](#)

Public

Contains the code and the slides of the presentation: '{flowchart}: an R package for creating participant flow diagrams integrated with tidyverse'

 HTML  2  1



Thank you!