

Thematic maps with mapsf : : CHEAT SHEET

Create and integrate thematic maps in your workflow.

Base map

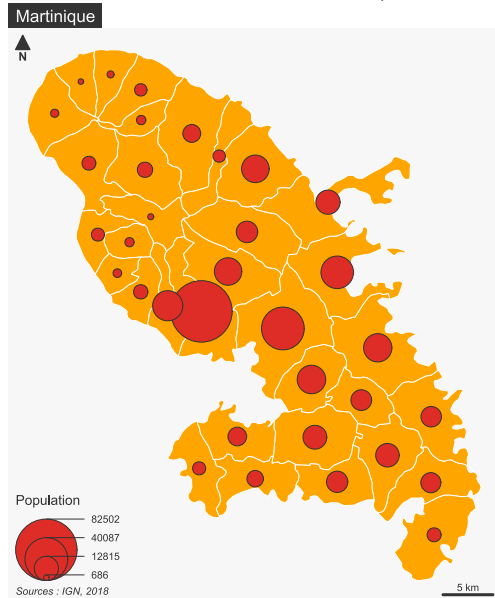
Import library
`library(mapsf)`

Import the sample data set
`mtq <- mf_get_mtg()`

Initiate a base map centered on a specific extent
`mf_map(x = mtq, col = "orange",
border = "white")`

Plot symbology
`mf_map(x = mtq, type = "prop", var = "POP",
leg_title = "Population", add = TRUE)`

Complete layout (credits, title, north, arrow, scale bar)
`mf_layout(title = "Martinique",
credits = "Sources: IGN, 2018")`



Colors

mapsf can use color palettes from `hcl.colors()`.
`mf_get_pal()` is useful to create well-balanced asymmetric diverging palettes

```
mf_get_pal(n = c(7, 2), pal = c("Burg", "Mint"))
```



Symbology

The x argument should be an sf object.
Input geometries can be polygons, lines or points.



Choropleth (ratios)
`mf_map(x = mtq, type = "choro",
var = "var", method = "quantile")`



Typology (categories)
`mf_map(x = mtq, type = "typo", var = "var")`



Proportional Symbols (stocks)
`mf_map(x = mtq, type = "prop", var = "var",
inches = 0.1, symbol = "circle")`



Graduated Symbols (stocks)
`mf_map(x = mtq, type = "grad", var = "var",
pch = 24)`



Symbols (categories)
`mf_map(x = mtq, type = "symb", var = "var",
pch = c(21:23))`



Choropleth proportional symbols (stocks - ratios)
`mf_map(x = mtq, type = "prop_choro",
var = c("var1", "var2"))`



Colorized proportional symbols (stocks - categ.)
`mf_map(x = mtq, type = "prop_typo",
var = c("var1", "var2"))`



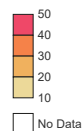
Choropleth symbols (ratios - categories)
`mf_map(x = mtq, type = "symb_choro",
var = c("var1", "var2"),
pch = c(21:23))`



Raster
`mf_raster(x = raster)`

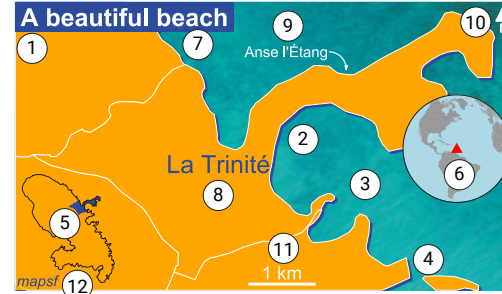
Legend

Default legends are plotted along maps.
Customization parameters are available with
`mf_legend()`



Map Layout

Along with cartographic functions, other functions are dedicated to customize the layout design.



1 Set a map theme (figure margins, colors, title options...)
`mf_theme(bg = "white", tab = TRUE,
mar = c(0,0,0,0), pos = "left")`

Builtin themes are available : default, ink, dark, agolalight, candy, darkula, iceberg, green, nevermind, jsk or barcelona.

2 Init a map centered on a specific area
`mf_init(x = mtq[30,])
mf_map(x = mtq, col = "orange",
border = "white", add = TRUE)`

3 Import external image for background
`mf_background(filename = "img/sea.jpg")`

4 Create a shadow effect
`mf_shadow(...)`

5 Create a custom inset
`mf_inset_on(x = mtq, pos = "bottomleft")
mf_map(...)
mf_inset_off()`

6 Create a world inset
`mf_inset_on(x = "worldmap", pos = "right")
mf_worldmap(mtg)
mf_inset_off()`

7 Plot title
`mf_title("A beautiful beach")`

8 Plot labels
`mf_label(...)`

9 Plot annotation (in specific locations)
`mf_annotation(...)`

10 North arrow
`mf_arrow(...)`

11 Scale (in km)
`mf_scale(...)`

12 Credits
`mf_credits(...)`

Export Maps

`mf_export()` exports maps in PNG or SVG formats.
The exported map width/height ratio will match the one of a spatial object.

Additionally, `mf_export()` can be used to set a theme, to extend the map space on one or several side of the figure, or to center a map on a specific area.

Simple export (PNG)
`mf_export(x = mtq, width = 500,
filename = "my_export.png")
mf_map(x = mtq, add = TRUE)
dev.off()`

Export with a theme (SVG)
`mf_export(x = mtq, width = 5, export = "svg",
filename = "my_export.svg",
theme = "nevermind")
mf_map(x = mtq, add = TRUE)
dev.off()`

Extra space on the figure (bottom, left, top, right)
`mf_export(x = mtq, width = 500,
filename = "my_export.png",
expandBB = c(0,0.6,0,0))
mf_map(x = mtq, add = TRUE)
dev.off()`

Export a map centered on a specific area
`mf_export(x = mtq[30,], height = 600,
filename = "my_export.png")
mf_map(x = mtq, add = TRUE)
dev.off()`

Further documentation

Vignettes on mapsf website: riatelab.github.io/mapsf

- > Get started
- > How to Use Themes
- > How to Export Maps
- > How to Create Inset Maps
- > How to Create Faceted Maps
- > How to Use a Custom Font Family