What You're Doing Is Rather Desperate

Notes from the life of a [data] scientist

Infographic-style charts using the R waffle package

September 8, 2017September 8, 2017 / nsaunders

Infographics. I've seen good examples. I've seen more bad examples. In general, I prefer a good chart to an infographic. That said, there's a "genre" of infographic that I do think is useful, which I'll call "if X were 100 Y". A good example: if the world were 100 people (http://floda31.com/2011/05/if-the-world-were-100-people/).

That method of showing proportions has been called a *waffle chart* and for extra "infographic-i-ness", the squares can be replaced by icons. You want to do this using R? Of course you do. Here's how.

There's not much more here than you'll find at the Github home of the R packages, waffle (https://github.com/hrbrmstr/waffle) and extrafont (https://github.com/wch/extrafont). I've just made it a little more step-by-step.

1. Install the R packages

You need *waffle* to create waffle charts and *extrafont* to use icons in the charts.

```
install.packages(c("waffle", "extrafont"))

library(waffle)
library(extrafont)
```

2. Install Font Awesome

The icons that *waffle* can use are supplied by Font Awesome (http://fontawesome.io/).

Download the zip file by clicking the Download link at their website. You may want to check out the paid option later but for now, just select the free download.

Unzip the file and navigate to the *fonts* directory. To install the font, use the file named `fontawesome-webfont.ttf`. For Windows or Mac OS X users, installation is as simple as double-clicking the file and choosing "Install". I haven't done this under Linux for a while, it may even be the same procedure these days.

3. Import and register fonts

In R, fonts have to be imported to the *extrafont* database. You need to do this whenever a new font is installed that you want to use in R.

```
font_import()

# check that Font Awesome is imported
fonts()[grep("Awesome", fonts())]
# [1] "FontAwesome"
```

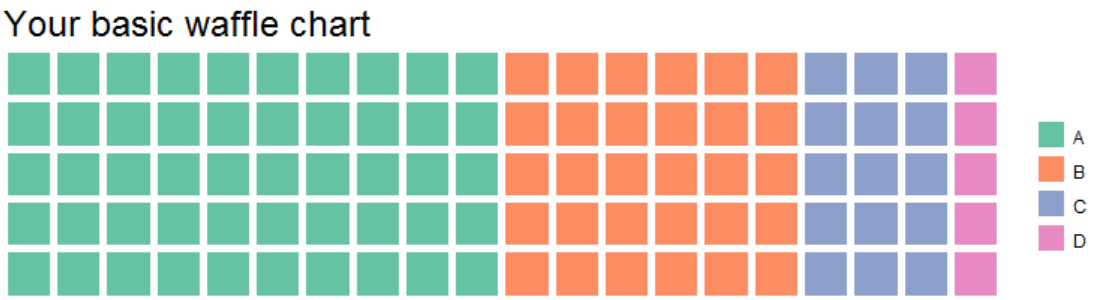
If you run `font_import()` in a session, you need to register the fonts with the R output device. Otherwise, this occurs when the package is loaded. If you're using the Windows OS and want to plot to the screen in RStudio, an additional argument is required:

```
# this should be fine for Mac OSX
loadfonts()
# use this if things look odd in RStudio under Windows
loadfonts(device = "win")
```

4. Create charts

Now we are ready to waffle. First, your basic waffle chart. The default colours are Colour Brewer "Set2".

```
waffle(c(50, 30, 15, 5), rows = 5, title = "Your basic waffle chart")
```



(https://nsaunders.wordpress.com/?attachment_id=4939)

Next, using icons. You can browse or search for available icon names on this page (http://fontawesome.io/icons/). You may need to fiddle with the size.

```
waffle(c(50, 30, 15, 5), rows = 5, use_glyph = "child", glyph_size = 6,
title = "Look I made an infographic using R!")
```

(https://nsaunders.wordpress.com/?attachment_id=4942)

You can use the `iron()` function to append waffle charts, which might be useful in comparisons. Here's some made-up data involving some aspect of cars in two unnamed countries:

```
iron(
  waffle(c(no = 80, yes = 20), rows = 5, use_glyph = "car", glyph_size = 6,
         colors = c("#c7d4b6", "#a3aabd"), title = "Country A"),
  waffle(c(no = 70, yes = 30), rows = 5, use_glyph = "car", glyph_size = 6,
         colors = c("#c7d4b6", "#a3aabd"), title = "Country B")
)
```

(https://nsaunders.wordpress.com/?attachment_id=4945)

Conclusion
That's it, more or less. Are the icons any more effective than the squares? In certain settings, perhaps. You be the judge.

R, statistics
packages, rstats, visualization, waffle

7 thoughts on "Infographic-style charts using the R waffle package"

1. *Matt*
   September 13, 2017 at 02:58
   Thank you, useful!

2. *damcoutinho*
   September 13, 2017 at 13:21
   I really enjoyed it! I'm learning R and tutorials like this are a really pleasant way to explore the language. Thank you for posting.

   About Font Awesome and Linux: double clicking the file didn't work for me on Ubuntu. The installation was successful with these commands:
   sudo apt-get update
   sudo apt-get install fonts-font-awesome

3. Pingback: Using Waffle Charts in R to Analyze Visits to the Grand Canyon - Real Life <- Code

4. *ingrid_b*
   September 16, 2017 at 00:58
   Super cool! Thanks for sharing :)

5. *Mike*
   September 25, 2017 at 21:08
   Fantastic mini tutorial. All worked as described. Thankyou! I found the cheat sheet at http://fontawesome.io/cheatsheet/ really useful as it gives pictures of all the glyphs available.

6. *Alexander Levashov*
   October 3, 2017 at 20:19
   Great tutorial Neil, thanks for writing it.

   I had a bit of problems importing FontAwesome on Win 10.

   Explicit import with font_import(pattern = "fontawesome-webfont.ttf") helped.

7. *acangros*
   October 9, 2017 at 04:17
   Amazing. It would nice also to be able to stack data in vertical

Comments are closed.