Exploring and presenting maps with tmap

Martijn Tennekes[1]

1. Statistics Netherlands

A map tells more than a thousand coordinates. Generally, people tend to like maps, because they are appealing, recognizable, and often easy to understand. Maps are not only useful for navigation, but also to explore, analyse, and present spatial data.

The **tmap** package offers a powerful engine to visualise maps, both static and interactive. It is based on the Grammar of Graphics, with a syntax similar to ggplot2, but tailored to spatial data. Layers from different sources can be stacked, map legends and attributes can be added, and small multiples can be created.

An example of a map is the following.

```
library(tmap)
data(World, metro)

tm_shape(World) +
  tm_polygons("HPI", id = "name") +
  tm_text("name", size = "AREA") +
tm_shape(metro) +
  tm_dots(id = "name")
```

Interaction with charts and maps is not considered as additional fancy feature anymore, of which users will say "wow, this is interactive!". To the contrary, users will expect charts and maps to be interactive, especially when published online. Also in *R*, interaction has become common ground, especially since the introduction of **shiny** and **htmlwidgets**.

However, the increase of interactive maps does not mean the end of static maps. Newspapers, journals, and posters still rely on printed maps. To design a thematic map that is appealing, informative, and yet simple is conisdered as a craft.

There are two modes in which maps can be visualized in **tmap**, `"plot"` for static plotting and `"view"` for interactive viewing. Users are able to switch between these modes without effort. The choropleth above is reproduced in interactive mode as follows:

```
ttm()
last_map()
```

The function `ttm` is a toggle between the modes. Alternatively, `tmap_mode("view")` can be used to swtich to the view mode.