The commute to my workplace is 90 minutes each way. Podcasts are my friend. I’m a long-time listener of [In Our Time](https://www.bbc.co.uk/programmes/b006qykl) and enjoyed the recent episode about [The Danelaw](https://www.bbc.co.uk/sounds/play/m0003jp7).

Melvyn and I hail from the same part of the world, and I learned as a child that many of the local place names there were derived from Old Norse or Danish. Notably: places ending in *-by* denote a farmstead, settlement or village; those ending in *-thwaite* mean a clearing or meadow.

So how local are those names? Time for some quick and dirty maps using R.

First, we’ll need a dataset of British place names. There are quite a few of these online, but top of my Google search was [Index of Place Names in Great Britain (July 2016)](http://geoportal1-ons.opendata.arcgis.com/datasets/a6c138d17ac54532b0ca8ee693922f10_0). It comes in several formats including CSV, easy to read into R like so:

library(tidyverse)

library(maps)

gbplaces <- read\_csv("https://opendata.arcgis.com/datasets/a6c138d17ac54532b0ca8ee693922f10\_0.csv?outSR=%7B%22latestWkid%22%3A27700%2C%22wkid%22%3A27700%7D")

A quick inspection of the data reveals that whilst there is a unique identifier, objectid\_1, each row is not as such a unique place (the dataset is based on grid locations). We can reduce the number of rows a little by taking distinct(placesort, lat, long\_), but that will still retain duplicate place names with slightly different coordinates. For our purposes, it doesn’t really matter – we just want an indication of distribution, rather than a highly-accurate map.

We’ll start by looking at places ending in *-by*. For this example, we’ll let the points themselves define the outline of Great Britain rather than drawing one. We’ll emphasise the *-by* places and try to de-emphasise the rest.

gbplaces %>%

distinct(placesort, lat, long\_) %>%

mutate(isBy = ifelse(grepl("^.+by$", placesort), TRUE, FALSE)) %>%

# not the territories!

filter(lat > 40) %>%

ggplot(aes(long\_, lat)) +

geom\_point(aes(color = isBy,

alpha = isBy),

size = 0.5) +

scale\_colour\_viridis\_d(direction = -1,

name = "ends in -by",

option = "inferno") +

scale\_alpha\_manual(values = c(0.3, 1)) +

theme(axis.title = element\_blank(),

axis.text = element\_blank(),

axis.ticks = element\_blank(),

panel.grid = element\_blank(),

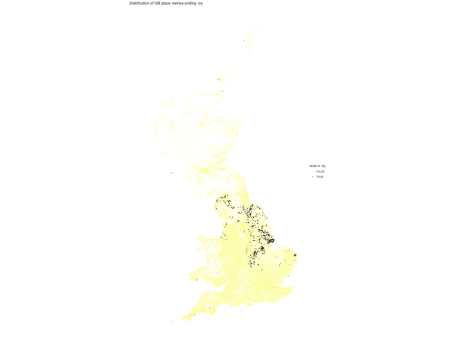
panel.border = element\_blank()) +

labs(title = "Distribution of GB place names ending -by") +

guides(alpha = FALSE) +

coord\_map()

Here’s the result – click for a larger version. Not bad. Lots of locations in Cumbria and eastern England. I like how the “plotting by points only” approach emphasises the empty mountainous regions in Scotland, Northern England and Wales.

[](https://nsaunders.files.wordpress.com/2019/04/ukby.png)

Now we’ll look at *-thwaite*. This time we’ll use map\_data() to pull an outline from the *maps* package.

# filter out N Ireland

ggplot(data = map\_data("world", "UK") %>% filter(group != 3),

aes(x = long, y = lat)) +

geom\_polygon(aes(group = group),

fill = "darkolivegreen") +

coord\_map() +

geom\_point(data = gbplaces %>% filter(grepl("^.+thwaite$", placesort),

lat > 40),

aes(long\_, lat),

color = "yellow",

size = 0.5) +

theme(axis.title = element\_blank(),

axis.text = element\_blank(),

axis.ticks = element\_blank(),

panel.grid = element\_blank(),

panel.border = element\_blank()) +

labs(title = "Distribution of GB place names ending -thwaite")

Result below. We see that *-thwaite* is much more localised to Cumbria and parts of Yorkshire.

[](https://nsaunders.files.wordpress.com/2019/04/gbthwaite.png)

**Summary**  
I find mapping languages quite fascinating, but of course it’s not an original idea. [Here’s an interactive map](https://www.britishmuseum.org/whats_on/exhibitions/vikings/vikings_live/old_norse_origins.aspx) of Norse-derived place names in the UK, developed for an exhibition at the British Museum. I’m sure there are many others.

If you want to put data on a map, R offers many options using base R, ggplot2 or interactive Javascript such as Leaflet.