The German NGO [Open Knowledge Foundation Deutschland e.V.](https://okfn.de/) has made German Trade Resister data available via the project [OffeneRegister.de](https://www.offeneregister.de/), together with the British NGO [opencorporates](https://opencorporates.com/). While the data from German Trade Resister is publicly available in principle, retrieving the data is a case-by-case activity and is very cumbersome ([try for yourself if you like](https://www.handelsregisterbekanntmachungen.de/)). The data provided by OffeneRegister.de instead comes with an easy to navigate API, and, what is even more convenient, is available for bulk download (alternatively as a JSON or as a SQLite database file).

Having a research focus on corporate transparency, I could not resist the temptation to take a peak. I downloaded the SQLite database. Let’s access it with R.

library(DBI)

library(tidyverse)

library(ggmap)

library(rgdal)

library(rgeos)

tmp <- tempdir()

unzip("~/../Downloads/handelsregister.db.zip", exdir = tmp)

db <- file.path(tmp, "handelsregister.db")

con <- dbConnect(RSQLite::SQLite(), db)

While I have my reservations about using the individual data contained in the data base for data quality and privacy reasons, some aggregate analysis seems in order. As a proof of concept, I will try to visualize where German companies are located. Thus, I focus on the relation companies and try to extract some spatial variation.

sql <- "select id, current\_status, retrieved\_at, registered\_address from company"

res <- dbSendQuery(con, sql)

company <- dbFetch(res)

dbClearResult(res)

dbDisconnect(con)

An inspection of the registered\_address field shows that the data is relatively messy, but most non-empty cells contain five digit German post codes (*Postleitzahlen*, PLZ). I will focus on those.

company %>%

filter(current\_status == "currently registered",

!is.na(registered\_address)) %>%

mutate(plz = str\_extract(registered\_address, "\\d{5}")) %>%

filter(plz != "") %>%

select(id, plz) -> company\_plz

Time to ask our first question. What are the TOP 10 firm hosting PLZs?

companies\_plz <- company\_plz %>%

group\_by(plz) %>%

summarise(companies = n()) %>%

mutate(link = sprintf('[Link to map](https://www.suche-postleitzahl.org/plz-gebiet/%25s)', plz)) %>%

arrange(-companies)

kable(head(companies\_plz, 10), col.names = c("PLZ", "Registered companies", "Link to map"),

format.args = list(big.mark = ","))

| **PLZ** | **Registered companies** | **Link to map** |
| --- | --- | --- |
| 10117 | 5,858 | [Link to map](https://www.suche-postleitzahl.org/plz-gebiet/10117) |
| 20457 | 4,678 | [Link to map](https://www.suche-postleitzahl.org/plz-gebiet/20457) |
| 20354 | 4,677 | [Link to map](https://www.suche-postleitzahl.org/plz-gebiet/20354) |
| 20095 | 3,574 | [Link to map](https://www.suche-postleitzahl.org/plz-gebiet/20095) |
| 82031 | 3,334 | [Link to map](https://www.suche-postleitzahl.org/plz-gebiet/82031) |
| 10719 | 2,854 | [Link to map](https://www.suche-postleitzahl.org/plz-gebiet/10719) |
| 22767 | 2,593 | [Link to map](https://www.suche-postleitzahl.org/plz-gebiet/22767) |
| 60325 | 2,313 | [Link to map](https://www.suche-postleitzahl.org/plz-gebiet/60325) |
| 10707 | 2,309 | [Link to map](https://www.suche-postleitzahl.org/plz-gebiet/10707) |
| 55129 | 2,223 | [Link to map](https://www.suche-postleitzahl.org/plz-gebiet/55129) |

The top PLZ on the list looks oddly familiar… But the list is not really informative as PLZ vary in size. Let’s link this to some spatial information.

# Shape file is based on OSM data.

# Source: https://osm.wno-edv-service.de/pcboundaries/

unzip("../data/alle\_plz\_shapes\_deu.zip", exdir = tmp)

plz\_polys <- readOGR(file.path(tmp, "alle\_plz\_shapes\_deu"), "alle\_plz\_deu")

plz\_polys <- plz\_polys[plz\_polys$PCLEVEL == "8", ]

plz\_polys$area\_sqkm <- raster::area(plz\_polys) / 1000000

# Population data is based on the 100m raster data from Zensus 2011.

# Downloaded from: https://www.suche-postleitzahl.org/downloads

# Further info:

# http://blog.suche-postleitzahl.org/post/132153774751/einwohnerzahl-auf-plz-gebiete-abbilden

# https://www.zensus2011.de/SharedDocs/Aktuelles/Ergebnisse/DemografischeGrunddaten.html?nn=3065474

pop <- readxl::read\_xls("../data/plz\_einwohner.xls")

pop\_by\_plz <- data.frame(

plz = as.character(plz\_polys$PCODE),

area\_sqkm = plz\_polys$area\_sqkm,

stringsAsFactors = FALSE

)

pop\_by\_plz %>%

left\_join(pop) %>%

left\_join(companies\_plz) %>%

replace\_na(list(companies = 0)) %>%

rename(population = einwohner) %>%

mutate(comp\_by\_sqkm = companies/area\_sqkm,

comp\_by\_1000pop = 1000\*companies/population) -> comp\_by\_plz

How many companies are included in this data?

tab <- comp\_by\_plz %>%

mutate(pa = substr(plz, 1,1 )) %>%

group\_by(pa) %>%

summarise(companies = sum(companies))

tab <- rbind(tab, data.frame(pa = "Total", companies = sum(tab$companies)))

kable(tab, col.names = c("Post Area", "Registered companies"),

format.args = list(big.mark = ","))

| **Post Area** | **Registered companies** |
| --- | --- |
| 0 | 44,955 |
| 1 | 124,408 |
| 2 | 137,019 |
| 3 | 79,518 |
| 4 | 121,718 |
| 5 | 110,496 |
| 6 | 112,305 |
| 7 | 97,356 |
| 8 | 101,587 |
| 9 | 67,897 |
| Total | 997,259 |

Quite a few. Which reasonably populated PLZs are home to more firms than people?

kable(comp\_by\_plz %>%

filter(population > 100,

comp\_by\_1000pop > 1000) %>%

arrange(-comp\_by\_1000pop),

col.names = c("PLZ", "Area (km²)", "Population", "Registered companies", "Link to map",

"Companies by km²", "Companies by 1,000 inhabitants"),

format.args = list(big.mark = ",", digits = 2))

| **PLZ** | **Area (km²)** | **Population** | **Registered companies** | **Link to map** | **Companies by km²** | **Companies by 1,000 inhabitants** |
| --- | --- | --- | --- | --- | --- | --- |
| 40212 | 0.43 | 543 | 1,390 | [Link to map](https://www.suche-postleitzahl.org/plz-gebiet/40212) | 3,216 | 2,560 |
| 20354 | 1.30 | 2,273 | 4,677 | [Link to map](https://www.suche-postleitzahl.org/plz-gebiet/20354) | 3,602 | 2,058 |
| 20457 | 14.76 | 2,566 | 4,678 | [Link to map](https://www.suche-postleitzahl.org/plz-gebiet/20457) | 317 | 1,823 |
| 20095 | 0.77 | 3,172 | 3,574 | [Link to map](https://www.suche-postleitzahl.org/plz-gebiet/20095) | 4,613 | 1,127 |

Interesting, Düsseldorf and Hamburg make the cut. Maps. We want to see maps. First: Registered companies by square kilometer.

# For visualization, values are log transformed. Set zero values to

# be 80 % of non-zero minimum to make them plottable.

min\_1000pop <- 0.8 \* min(comp\_by\_plz$comp\_by\_1000pop[comp\_by\_plz$comp\_by\_1000pop > 0], na.rm = TRUE)

min\_sqkm <- 0.8 \* min(comp\_by\_plz$comp\_by\_sqkm[comp\_by\_plz$comp\_by\_sqkm > 0], na.rm = TRUE)

log\_safe <- comp\_by\_plz %>%

mutate(comp\_by\_1000pop = replace(comp\_by\_1000pop,

comp\_by\_1000pop == 0, min\_1000pop),

comp\_by\_sqkm = replace(comp\_by\_sqkm,

comp\_by\_sqkm == 0, min\_sqkm))

plz\_map <-

fortify(plz\_polys, region = "PCODE") %>%

left\_join(log\_safe, by = c("id" = "plz"))

ggplot(plz\_map, aes(x = long, y = lat, group = group, fill = comp\_by\_sqkm)) +

geom\_polygon(colour = NA, lwd=0, aes(group = group)) +

scale\_fill\_gradient2(name = "Registered companies per km²",

low = "red", mid = "gray90", high = "blue", trans = "log",

midpoint = log(median(log\_safe$comp\_by\_sqkm, na.rm = TRUE)),

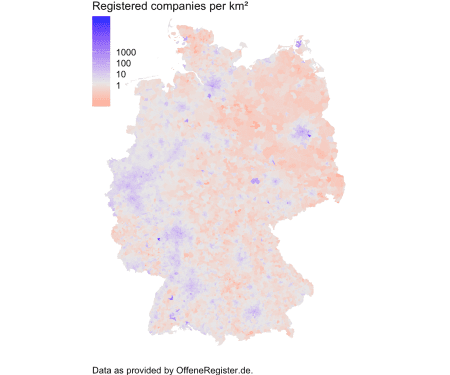
breaks = c(1, 10, 100, 1000)) +

coord\_map() +

theme\_void() +

theme(legend.justification=c(0,1), legend.position=c(0,1), plot.caption = element\_text(hjust = 0)) +

labs(caption = "Data as provided by OffeneRegister.de.")



Nice but not really surprising. Some areas stand out (Ost-Westfalen Lippe and the Rhein valley) but in general companies are where people live (meaning: in cities). Do we get better insights when we plot registered companies relative to the population?

# As we have seen above, we have some few PLZs with rather extreme companies

# by population ratios. I set them to 1000 so that they do not mess up the scale

plz\_map$comp\_by\_1000pop[plz\_map$comp\_by\_1000pop > 1000] <- 1000

ggplot(plz\_map, aes(x = long, y = lat, group = group, fill = comp\_by\_1000pop)) +

geom\_polygon(colour = NA, lwd=0, aes(group = group)) +

scale\_fill\_gradient2(name = "Registered companies per 1,000 inhabitants",

low = "red", mid = "gray90", high = "blue", trans = "log",

midpoint = log(median(log\_safe$comp\_by\_1000pop, na.rm = TRUE)),

breaks = c(1, 10, 100)) +

coord\_map() +

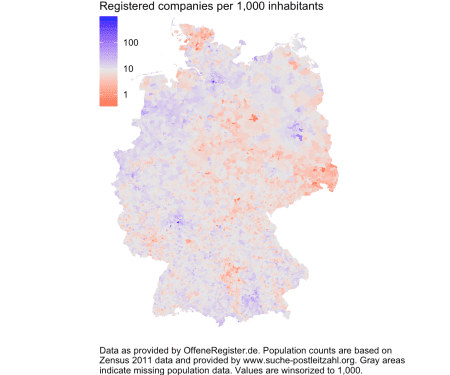
theme\_void() +

theme(legend.justification=c(0,1), legend.position=c(0,1), plot.caption = element\_text(hjust = 0)) +

labs(caption = paste("Data as provided by OffeneRegister.de. Population counts are based on",

"Zensus 2011 data and provided by www.suche-postleitzahl.org. Gray areas",

"indicate missing population data. Values are winsorized to 1,000.", sep = "\n"))



Now this is interesting. You can see relatively strong regional patterns. Some cities are not as company heavy as others (compare Leipzig to Dresden and see the Ruhr area). Some rural areas show higher levels of corporate activity (North Brandenburg, Mecklenburg) while others have relatively low levels (North Schleswig-Holstein, Saxonia, Swabia in Bavaria).

So, the data provided by OffeneRegister.de, while being somewhat messy at the individual level, can be used to generate informative insights at the aggregate level. Let’s hope that the initiative helps to trigger a public debate about the right way to host public data.