

Beispielanalyse der im Umkreis von Windrädern liegenden Hotels mit der freien Software „R“

Für die Analyse wurde folgendes Paket genutzt:

Installation:

```
@Manual{
  title = {sf: Simple Features for R},
  author = {Edzer Pebesma},
  year = {2017},
  note = {R package version 0.5-4},
  url = {https://CRAN.R-project.org/package=sf}, }
```

Quelle:

Paket: Edzer Pebesma (2017). sf: Simple Features for R.

R package version 0.5-4.

<https://CRAN.R-project.org/package=sf>

Analyse:

```
library(tidyverse)
library(sf)
library(rgeos)
# library(mapview)
library(tmap)

# Laden der Daten

# Am Ende muss ein Objekt (z.B. "hotels") mit den
Koordinaten der Hotels und (z.B. als Punkte)

# ein/mehrere Objekt(e) mit den Windpotenzialflächen,
z.B. "windpots" zur Verfügung stehen:

hotels <- ...

windpots0 <- ...

windpots1 <- ...

windpots2 <- ...
```

```
windpots3 <- ...
```

Außerdem wird ein Polygon mit den
Landkreisgrenzen "lk_aw" und Polygone mit den
Ortsgemeindegrenzen "ogs" benötigt:

```
lk_aw <- ...
```

```
ogs <- ...
```

Die folgende Funktion erledigt die ganze Arbeit

```
hotelsImPuffer <- function(windpots, label, radius) {
  windbuffer <- st_buffer(windpots, radius) %>%
  st_union

  hotels_in_buffer_indices <-
  st_intersects(windbuffer, hotels)[[1]]

  hotels_in_windbuffer <-
  hotels[hotels_in_buffer_indices, ]
```

```
chr <- sprintf("Radius %s m: %s Hotels im
Puffer.\r\n\r\n", radius, nrow(hotels_in_windbuffer))
```

```
print(chr)
```

```
if (nrow(hotels_in_windbuffer) > 0) {
```

```
  # In welcher OG wie viele Häuser?
```

```
  cover_indices <- ogs %>%
  st_covers(hotels_in_windbuffer)
```

```
  n_hotels_in_ogs <- map(cover_indices,
~length(.x))
```

```
  hotel_names_in_ogs <- map(cover_indices, ~{
```

```
    if (length(.x) > 0) sprintf("%s (%s %s, %s)",
hotels_in_windbuffer[.x, ]$name,
```

```
hotels_in_windbuffer[.x, ]$langname,
```

```
hotels_in_windbuffer[.x, ]$hausnummer,
```

```
hotels_in_windbuffer[.x, ]$ort)
```

```
  })
```

```
  nu <- n_hotels_in_ogs[n_hotels_in_ogs > 0] %>%
  unlist
```

```

na <- ogs$name[n_hotels_in_ogs > 0]
ho <- hotel_names_in_ogs[n_hotels_in_ogs > 0]

pwalk(list(x = na, y = nu, z = ho), function(x, y, z) {

  chr <-<- paste0(chr, sprintf("\r\n\r\nRadius %s m,
OG '%s': %s Hotels im Puffer:\r\n  %s", radius, x, y,
paste(z, collapse = "\r\n  ")))

  })

  chr <- paste0(chr, sprintf("\r\n\r\n%s  %s\r\n",
nrow(hotels_in_windbuffer), sum(nu)))

}

write_file(chr,
sprintf("diesen/pfad/bitte/nach/belieben/anpassen/win
dpuffer_%s_radius_%dm.txt", label,
as.integer(radius)))

# windbuffer <- sp::spTransform(windbuffer,
sp::CRS("+init=epsg:4326"))

# hotels_in_windbuffer <-
sp::spTransform(hotels_in_windbuffer,
sp::CRS("+init=epsg:4326"))

# osm_tiles <-
tmaptools::read_osm(bbox(lk_aw %>%
st_transform(crs = 4326)))

# tm_shape(osm_tiles) + tm_raster() +
tm_shape(windbuffer %>% st_transform(crs = 4326))
+ tm_shape(hotels_in_windbuffer %>%
st_transform(crs = 4326))

# plot(windbuffer)

# plot(hotels_in_windbuffer)

tm <- qtm(lk_aw, fill = NULL, borders = "blue") +

qtm(windpots, fill = "yellow", fill.alpha = 0.3) +

qtm(windbuffer, fill = "black", fill.alpha = 0.1)

if (nrow(hotels_in_windbuffer) > 0)

tm <- tm + qtm(hotels_in_windbuffer, fill = "black",
borders = "black")

```

```

tm <- tm + tm_format_World(title =
sprintf("Radius: %s m", radius))

save_tmap(tm,
sprintf("diesen/pfad/bitte/nach/belieben/anpassen/win
dpuffer_%s_radius_%dm.png", label,
as.integer(radius)),

width = 1920, height = 1080)

}

# Nun kann man wie gewünscht die Funktion auf
verschiedene Windpotenzialflächen und verschiedene
Radien anwenden, z.B.:

hotelsImPuffer(windpots0, "sz0", 500) # Szenario 0,
500 m Radius

hotelsImPuffer(windpots0, "sz0", 1000) # Szenario 0,
1000 m Radius

hotelsImPuffer(windpots0, "sz0", 1500) # ...

hotelsImPuffer(windpots0, "sz0", 2000)

hotelsImPuffer(windpots1, "sz1", 500) # Szenario 1,
500 m Radius

hotelsImPuffer(windpots1, "sz1", 1000) # Szenario 1,
1000 m Radius

hotelsImPuffer(windpots1, "sz1", 1500) # ...

hotelsImPuffer(windpots1, "sz1", 2000)

# usw.

```