# Wavelet Secure Maps: enhancing privacy protected maps

Edwin de Jonge @edwindjonge

Statistics Netherlands Research & Development
@edwindjonge

2024, October 7 2024

# Wavelet Secure Maps: enhancing privacy protected maps

# sdcSpatial: Privacy protected maps

**Takeout message: `sdcSpatial` and Wavelets:**

- `sdcSpatial` helps to assess sensitiviy and create privacy protected density maps.
- `protect_wavelet` novel method for protecting density maps.
- takes care of rural vs urban spatial resolution.
- can be seen as combination of `protect_smooth` and `protect_quadtree`.
- *work in progress, on github, not yet on CRAN*

# Who am I and why these protection methods?

- Statistical consultant, Data Scientist @cbs.nl / Statistics NL
- Statistics Netherlands is producer main official statistics in the Netherlands:
  - Stats on Demographics, economy (GDP), education, environment, agriculture, Finance etc.
  - Part of the European Statistical System, ESS.

### Motivation for `sdcSpatial`

- ESS has European Code of Statistical Practice (predates GDPR, European law on Data Protection): **no individual information may be revealed**.

# Sdc in `sdcSpatial`?

SDC = “Statistical Disclosure Control”

**Collection of statistical methods to:**

- Check if data is safe to be published
- Protect data by slightly altering (aggregated) data
  - adding noise
  - shifting mass
- Most SDC methods operate on records.
- **`sdcSpatial` works upon locations.**

# Data

```
data(dwellings, package="sdcSpatial")
nrow(dwellings)
```

```
## [1] 90603
```

```
head(dwellings) # consumption/unemployed are simulated!
```

```
##        x      y consumption unemployed
## 1 149712 470104    2049.926      FALSE
## 2 149639 469906    1814.938      FALSE
## 3 149631 469888    2074.882      FALSE
## 4 149788 469831    1927.989      FALSE
## 5 149773 469834    2164.969      FALSE
## 6 149688 469898    1987.958      FALSE
```

# Let's create a sdc_raster

## Creation:

```r
library(sdcSpatial)
unemployed <- sdc_raster( dwellings[c("x", "y")] # realistic locations
                        , dwellings$unemployed # simulated data!
                        , r = 500 # raster resolution of 500m
                        , min_count = 10 # min support
                        )
```

## What has been created?

```r
print(unemployed)
```

```
## logical sdc_raster object:
##    resolution: 500 500 ,  max_risk: 0.95 , min_count: 10
##    mean sensitivity score [0,1]:  0.4249471
```

42% of the data on this map is sensitive and should be protected!

# Type of raster density maps:

(Stored in `unemployed$value`):

Density can be area-based:

- **number of people** per square (`$count`): population density.
- **(total) value** per square (`$sum`): number of unemployed per square.

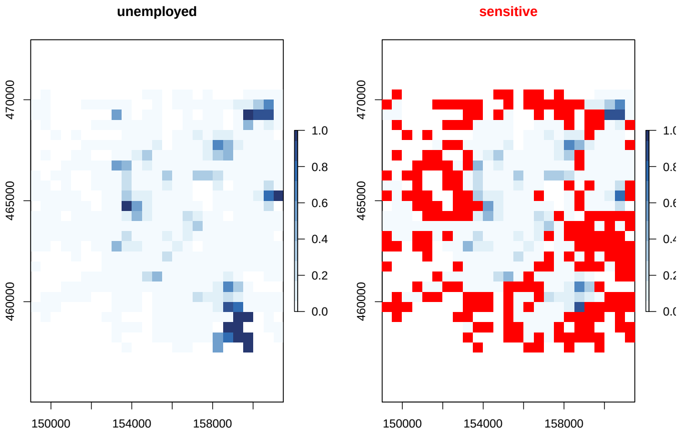
Or density can population-based:

- **Mean value** per square (`$mean`): unemployment rate per square.

*Note: All density types are valid, but (total) value per square strongly interacts with population density.*
*(e.g. https://xkcd.com/1138)*.

# Plotting a `sdc_raster`

```
plot(unemployed, "mean")
```



# How to reduce sensitivity?

**Options:**

a) Remove sensitive locations: **`remove_sensitive`**.

b) Use a coarser raster: `sdc_raster`.

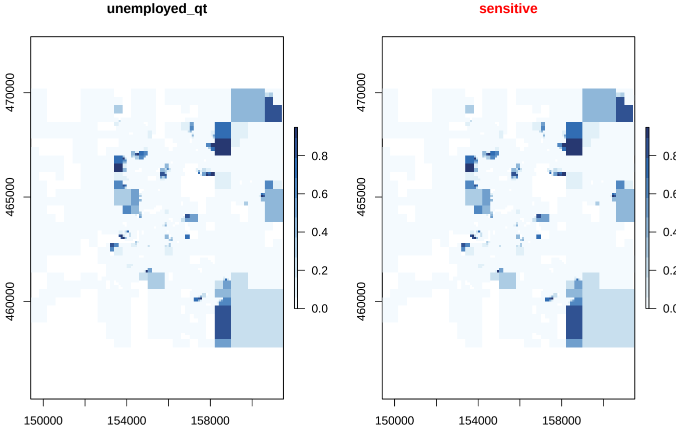
c) Aggregate sensitive cells hierarchically with a quad tree until not sensitive: **`protect_quadtree`** (Suñé et al. 2017).

d) Apply spatial smoothing: **`protect_smooth`** (Wolf and Jonge 2018; Jonge and Wolf 2016).

e) Do a multi-resolution analysis **`protect_wavelet`**

# Option: protect_quadtree

```
unemployed_100m <- sdc_raster( dwellings[c("x","y")], dwellings$unemployed
                             , r = 100) # use a finer raster
unemployed_qt <- protect_quadtree(unemployed_100m)
plot(unemployed_qt)
```



# Option: `protect_quadtree`

### Pro

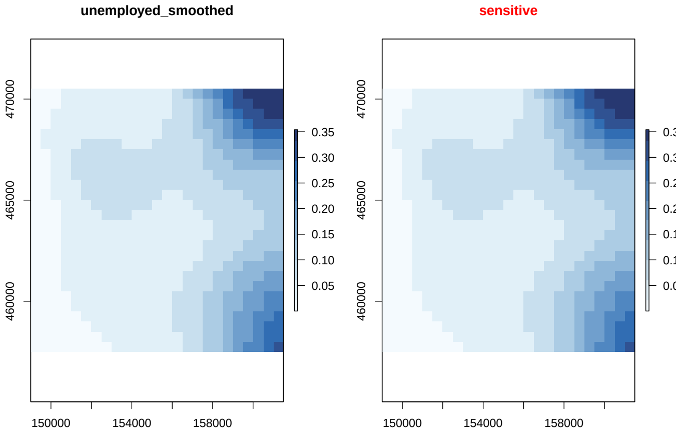
- Adapts to data density
- Adjusts until no sensitive data is left.
- It just works. . .

### Cons

- Crude, it just works :-)
- Visually: “Blocky” / “Mondrian-like” / Minecraft-result
- Is not translation invariant (basis)
- Isn’t there something smoother?

# Option: protect_smooth

```
unemployed_smoothed <- protect_smooth(unemployed, bw = 1500)
plot(unemployed_smoothed, "mean")
```



# Option: `protect_smooth`

## Pro's

- Often enhances spatial pattern visualization, removing spatial noise.
- Makes it a density map and used as source for e.g. contour map.

## Con's

- Does not remove all sensitive values (depends on bandwidth `bw`)
- A fixed band width is used for all locations: may remove detailed patterns. . .
  spatial processes often have location dependent band widths.

# Problem: smooth and adaptive

We need both a smooth and adaptive method!

### Wavelets

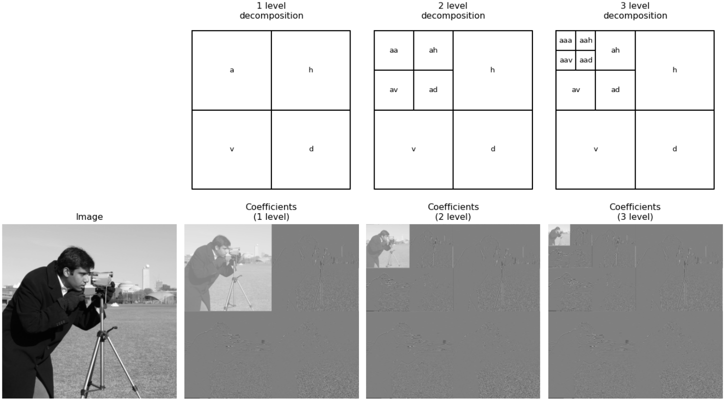
- Used for multi-resolution analysis (MRA), decompose signal / image at multiple resolutions.
- Used for denoising images: ("did I hear smoothing"?)

Also: - Used for lossy compression of images (e.g. JPEG!)

### We skip the math

- Except: wavelet decomposition can have different base functions, e.g. Haar, Daubechies etc.

# Wavelet and images (e.g. JPEG)



# sdcSpatial and Wavelets:

- Enter: `protect_wavelet`
- Using `dwt.2d`and `waveslim::idwt.2d` of R package `waveslim` (Whitcher 2024)
- Builds a multi-resolution version of density map
- Checks sensitivity (privacy) multiple resolutions

```r
unemployed_wvlt <- protect_wavelet(
  unemployed,
  wf = "la8", # wavelet transform / base functions
  depth = 4, # resolution depth
  ... # denoising parameters
)
```

# The end

- `protect_wavelet` wip on github
- "raw version", testing make it user friendly, in September on CRAN.

**Thank you for your attention!**

**Questions?**

**Curious?**

```
install.packages("sdcSpatial")
```

**Feedback and suggestions?**

https://github.com/edwindj/sdcSpatial

# References

Jonge, Edwin de, and Peter-Paul de Wolf. 2016. "Spatial Smoothing and Statistical Disclosure Control." In *Privacy in Statistical Databases*, edited by Josep Domingo-Ferrer and Mirjana Pejić-Bach, 107–17. Springer.

Suñé, E., C. Rovira, D. Ibáñez, and M. Farré. 2017. "Statistical Disclosure Control on Visualising Geocoded Population Data Using Quadtrees." http://nt17.pg2.at/data/x_abstracts/x_abstract_286.docx.

Whitcher, Brandon. 2024. *Waveslim: Basic Wavelet Routines for One-, Two-, and Three-Dimensional Signal Processing*. https://CRAN.R-project.org/package=waveslim.

Wolf, Peter-Paul de, and Edwin de Jonge. 2018. "Spatial Smoothing and Statistical Disclosure Control." In *Privacy in Statistical Databases - PSD 2018*, edited by Josep Domingo-Ferrer and Francisco Montes Suay. Springer.