

# GIS in Archaeology

## 06 - Densities

Martin Hinz

Institut für Archäologische Wissenschaften, Universität Bern

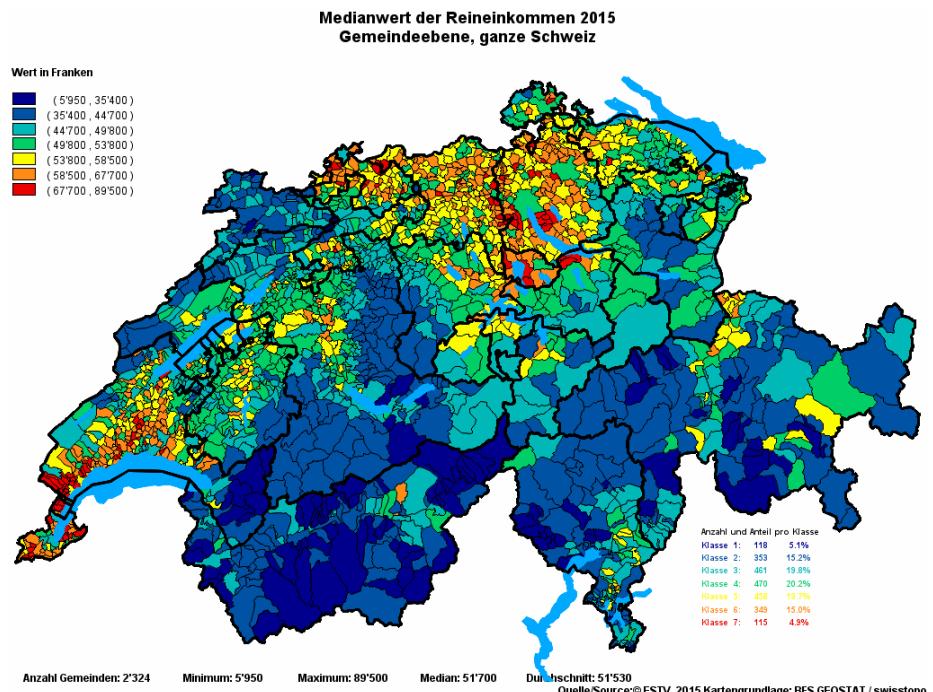
21/10/20

# Leftover from last session

- Spatial Join did not work last time
- Reason: Different projections of the involved layers
  - Seem to be a problem in older QGIS versions and/or Windows version!?
- Solution: Reprojecting the layers
- to start: Please download this [QGIS Project](#) and unzip it to a location of your choice

# Calculate mean 14C age per canton (1)

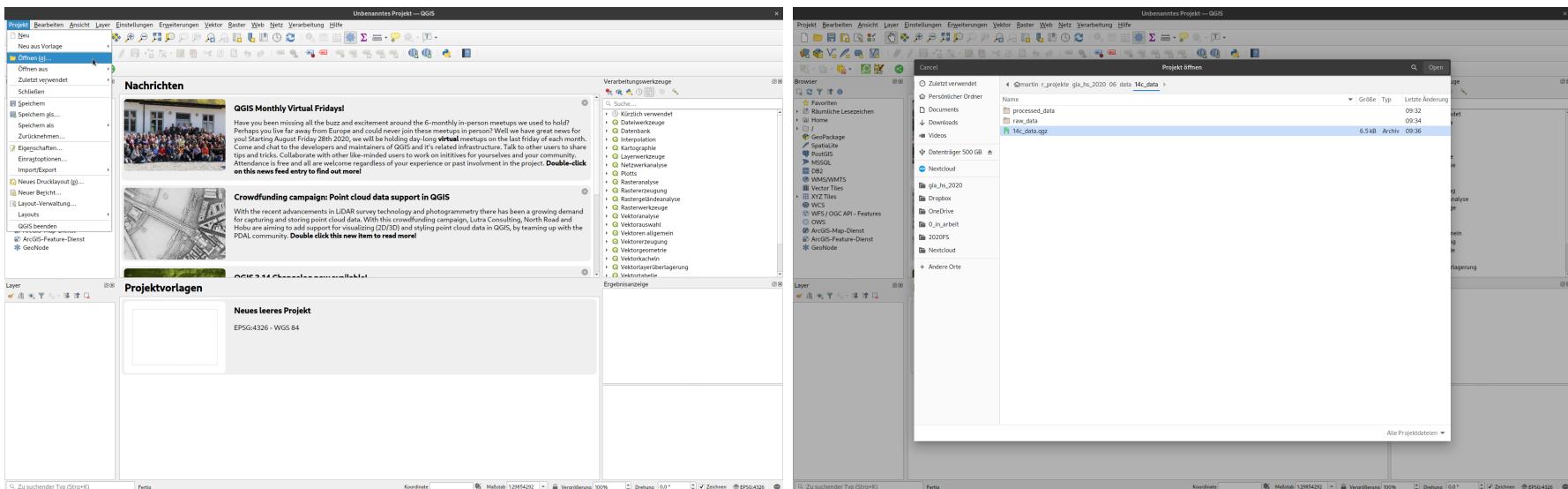
Using GIS, you can execute statistical procedures and visualise the result in a Choropleth.  
(the German term for this is also Choropleth...)



Source: <http://www.estv2.admin.ch/>

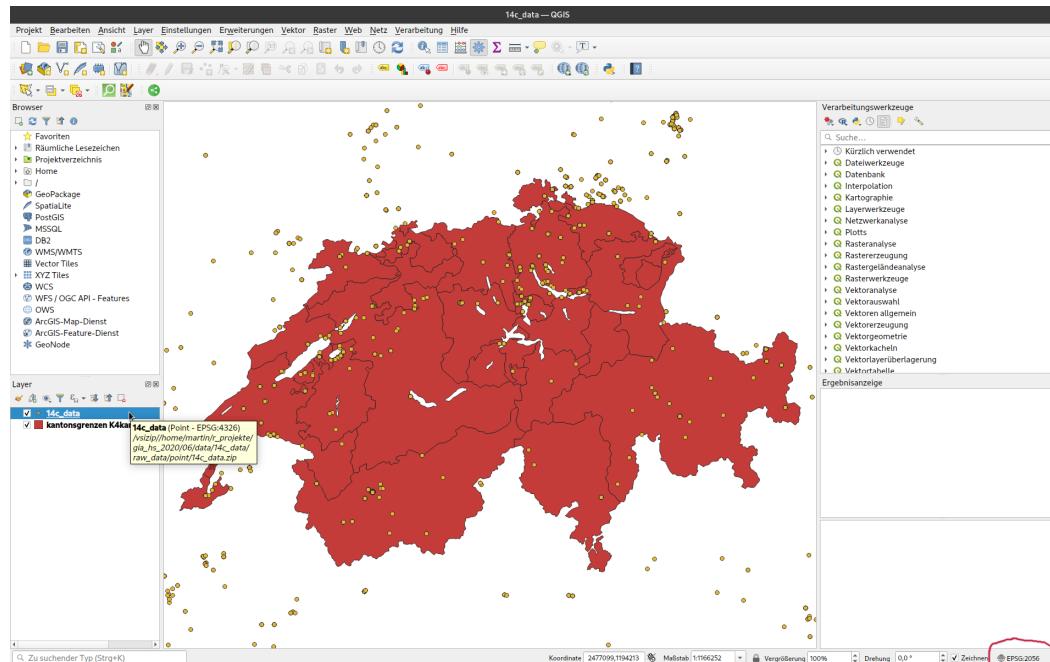
# Open a QGIS Project

- open the containing 14c\_data.qgz as QGIS Project
- all layers (if still at the correct relative path) should be loaded immediately



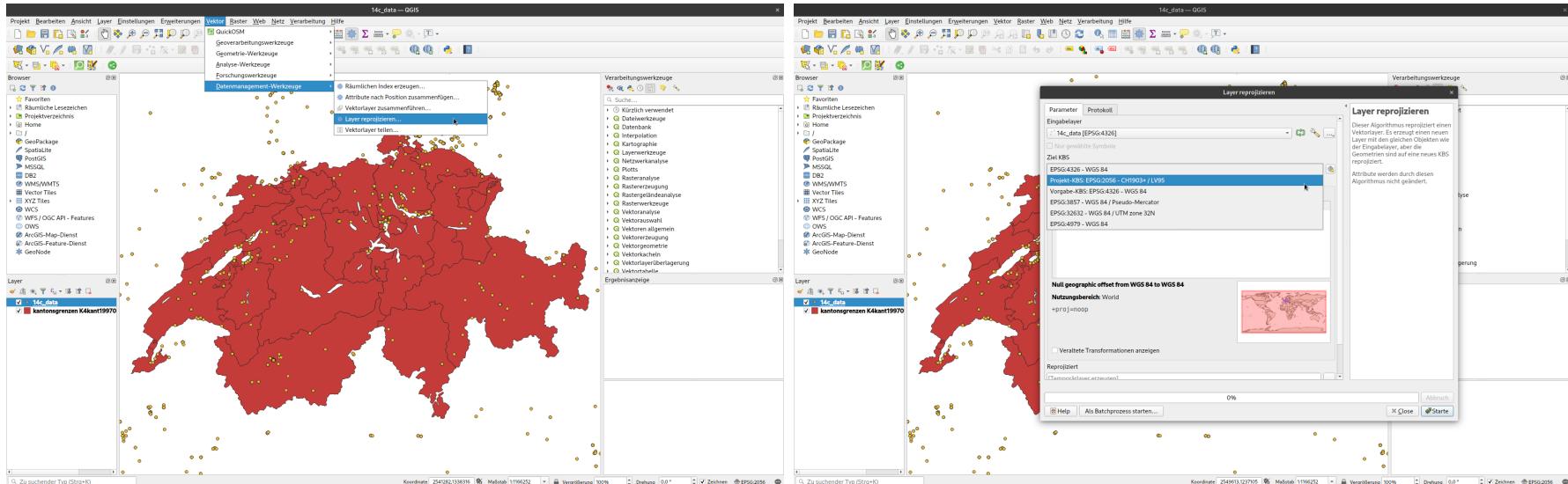
# Check the CRS

- first, set your project to the correct CRS (here: EPSG 2056)
- look to the individual layers to check if these layers have this projection
  - can be done by hovering the mouse over the layer in the layer pane
- it turns out, the point layer is in EPSG 4326



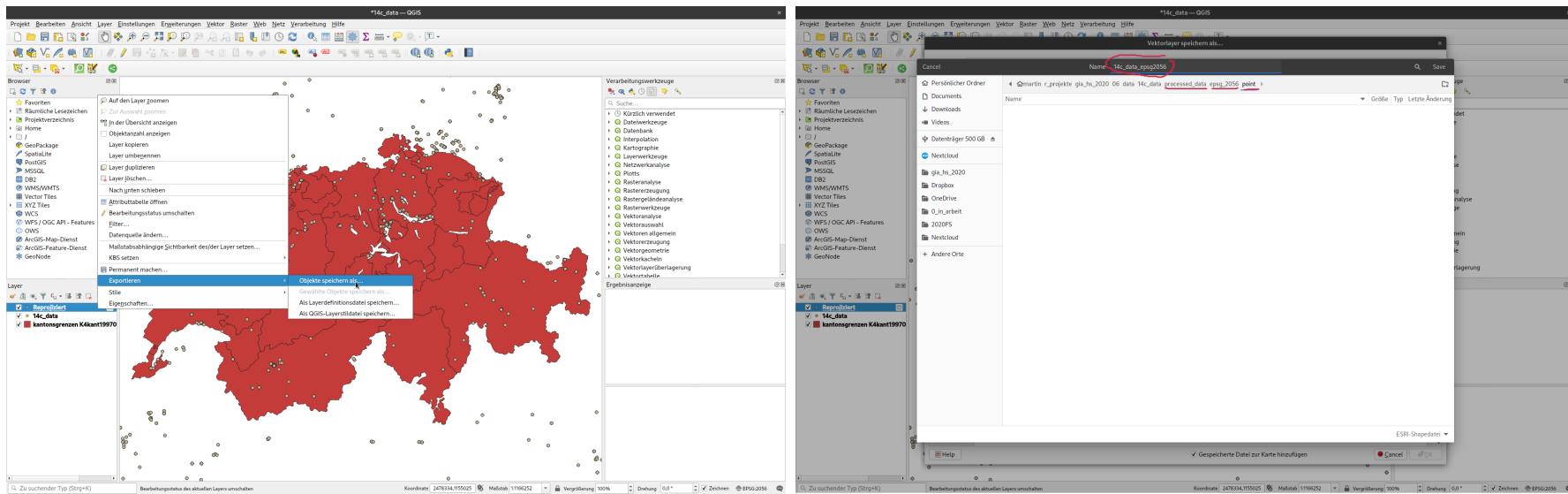
# Reproject Layer

- Select 'Reproject Layer' from 'Vector > Data management tools'
- Select your project CRS as target CRS
- Click on 'Run'



# Saving reproject layer

- Currently, the layer is a temporary layer
- to work with it, it is more save to export it to a shapefile
- please keep a reasonable file structure
  - eg. separating raw data from processed data
  - eg. separating raster from vector (also best separated)
  - indicate the projection system in file and folder names



# Calculate mean 14C age per canton (2)

*(Spatial Join)*

To start, we need the cantonal borders and the 14C data and the Toolbox .

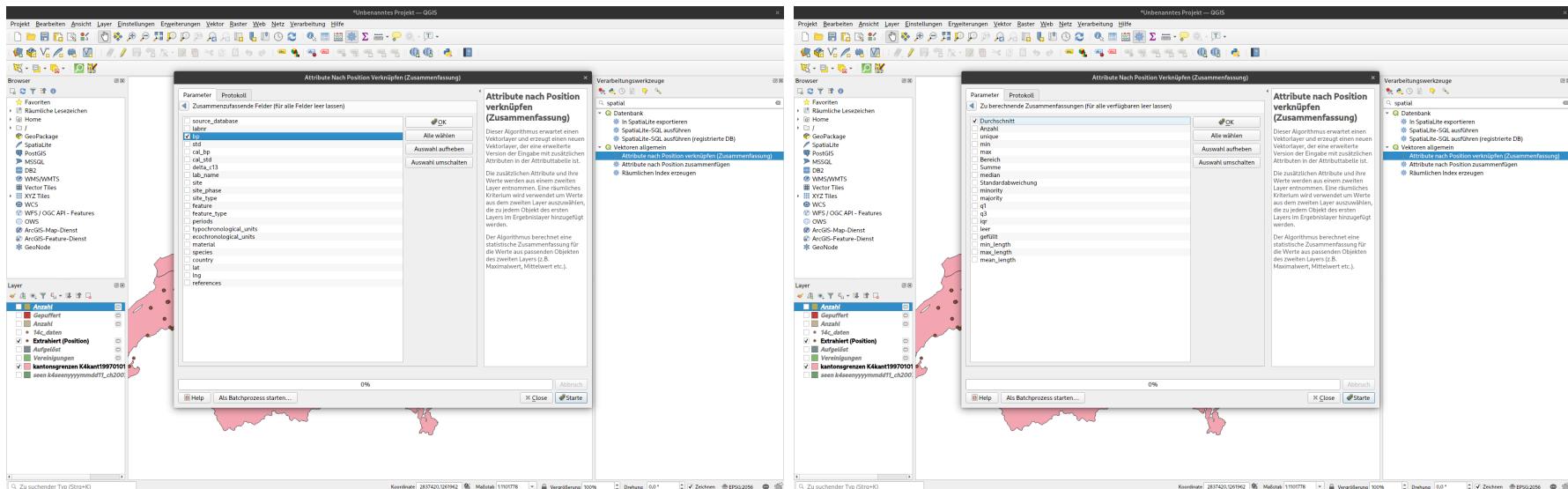
The command is 'Join attributes by location (summary)'

# Calculate mean 14C age per canton (3)

- Select the cantonal borders as input layer
- Select the 14C data as joined layer
- Select intersection as condition

# Calculate mean 14C age per canton (4)

- in Summary fields, select bp (uncalibrated 14C date) as field
- in Calculated Summaries, select 'mean' as statistic
- temporary layer might be ok
- press 'Run'



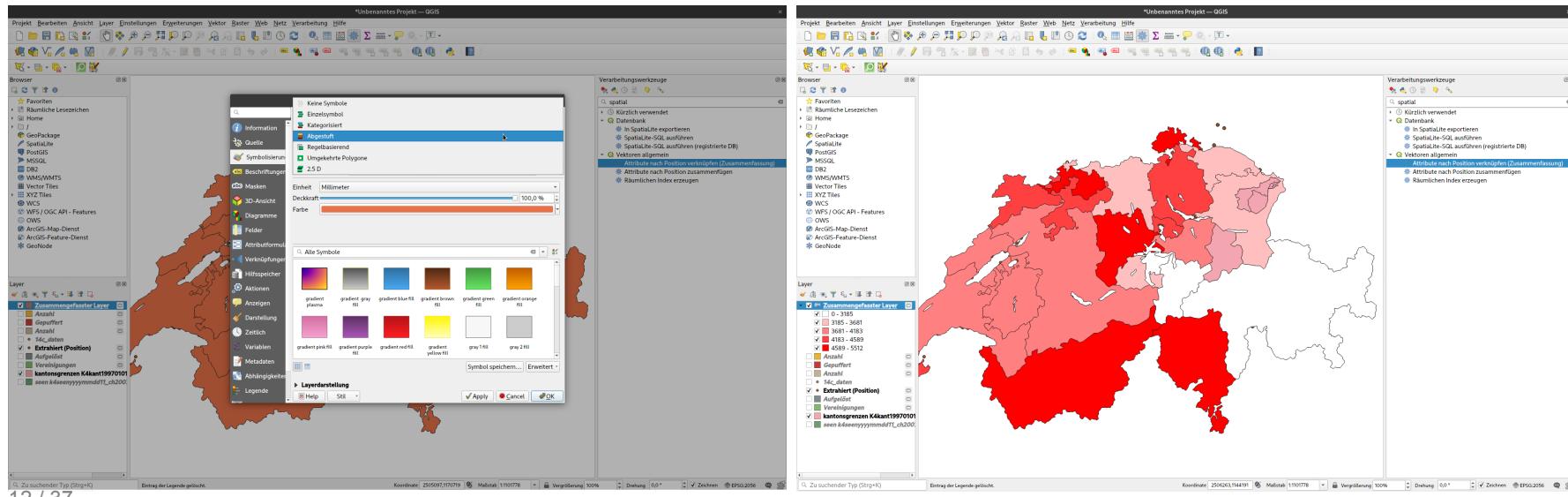
## Calculate mean 14C age per canton (5)

In the Attribute Table, now you can see the mean bp age per canton

# Visualise mean 14C age per canton

Now we can color the cantons according to the mean BP age

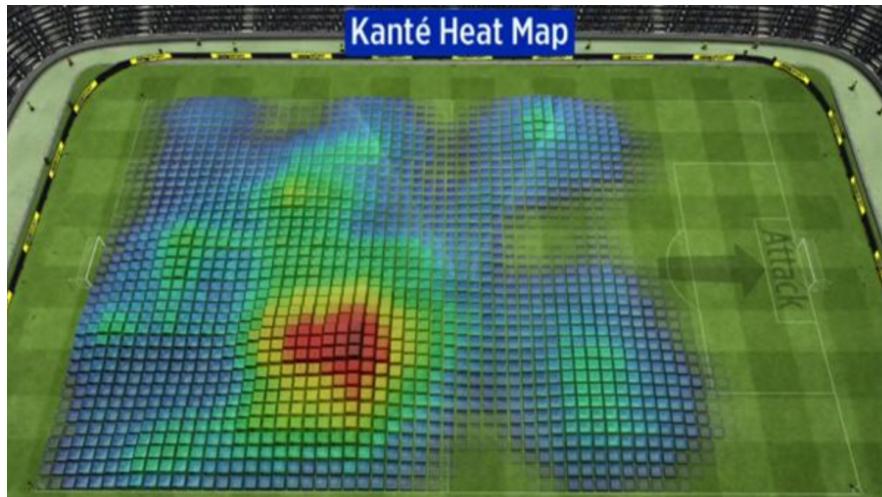
- Open the **Properties** of the layer (right click on the layer) and Select there the **Symbology** tab
- Select 'Graduated' as symbol and Classify the values
- click on 'OK'
- Now you should be able to see the age by color



# Visualising densities

## Heatmaps

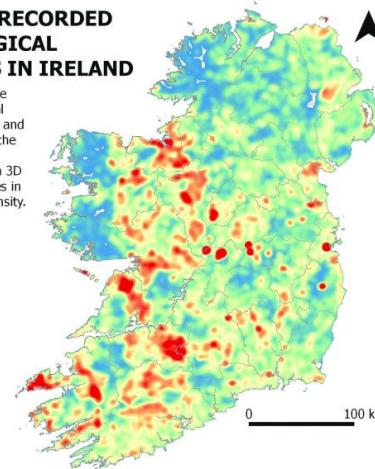
- Heatmaps are one of the best visualization tools for dense point data
- Heatmap is an interpolation technique that useful in determining density of input features
- Heatmaps are most commonly used to visualize crime data, traffic incidents etc.
- QGIS has a heatmap renderer that can be used to create an raster from a point layer.



### DENSITY OF RECORDED ARCHAEOLOGICAL MONUMENTS IN IRELAND

Map showing the relative density of archaeological monuments in the Sites and Monuments Record for the Republic of Ireland and Northern Ireland, with a 3D graphic illustrating spikes in recorded monument density.

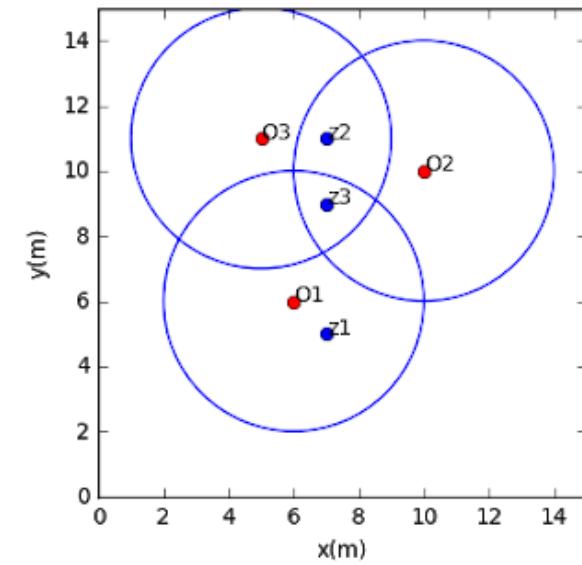
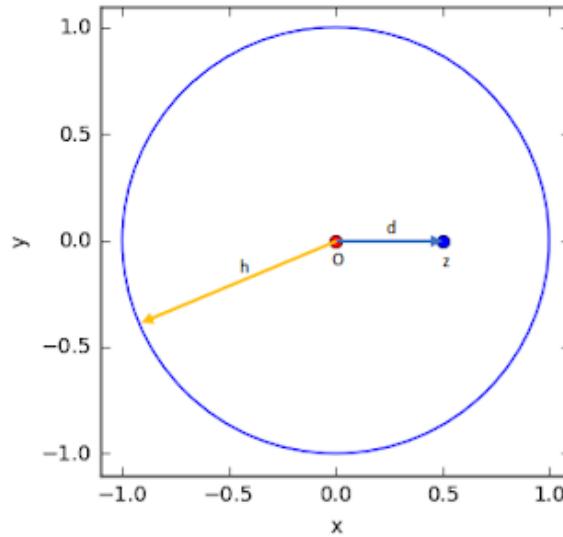
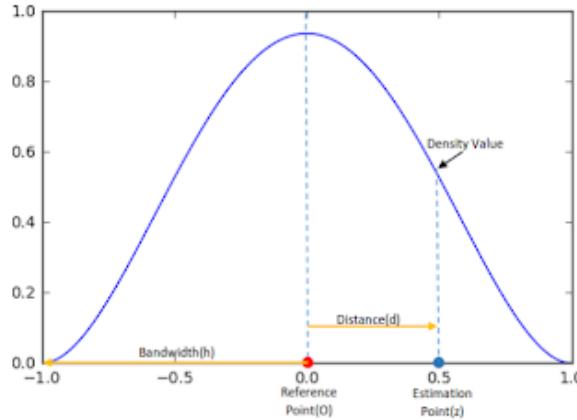
High density  
Low density



Source: Berner Zeitung; <https://archaeologyireland.ie/>

# Heatmaps

- technically, Heatmaps are produced using a technique called KDE (Kernel Density Estimation)
- individual points are 'blurred' using most of the time a gaussian distribution (Kernel) with specific parameters
- these 'blurred' points are combined
- the values are represented by a raster

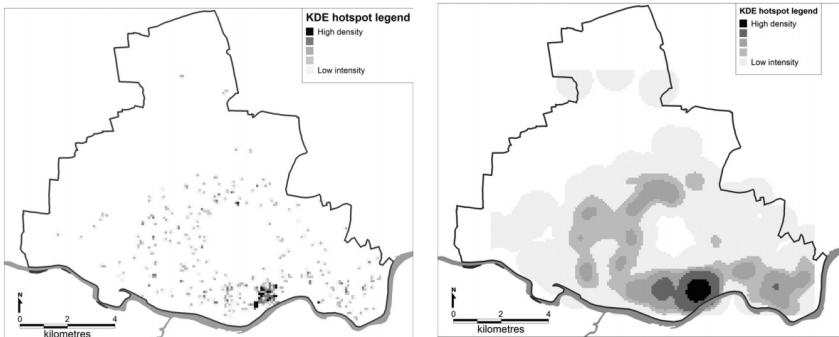


Source: <https://www.geodose.com>

# KDE - what to consider

- Settings to influence: bandwidth and resulting resolution
- Also, like most processing tools, it operates only reasonable in projected CRS (meter based)

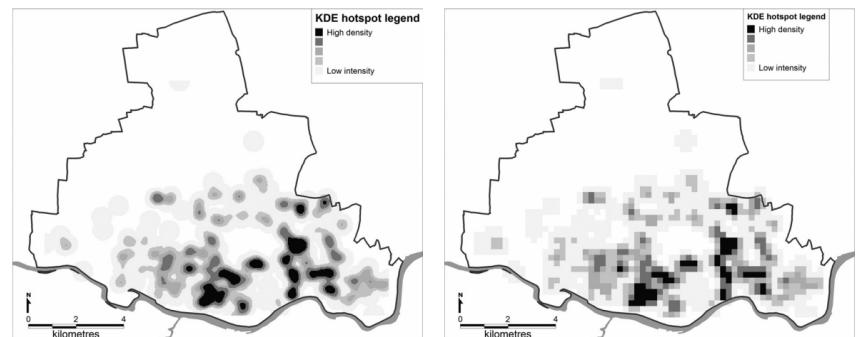
## Bandwith



**Figure 7.** A comparison of KDE hotspot maps generated using the same cell size but with different bandwidth sizes (a) 100 metres (PAI of 119.3) and (b) 800 metres (PAI of 40.4)

Different bandwidths result in more isolated vs. more homogeneous result

## Resolution



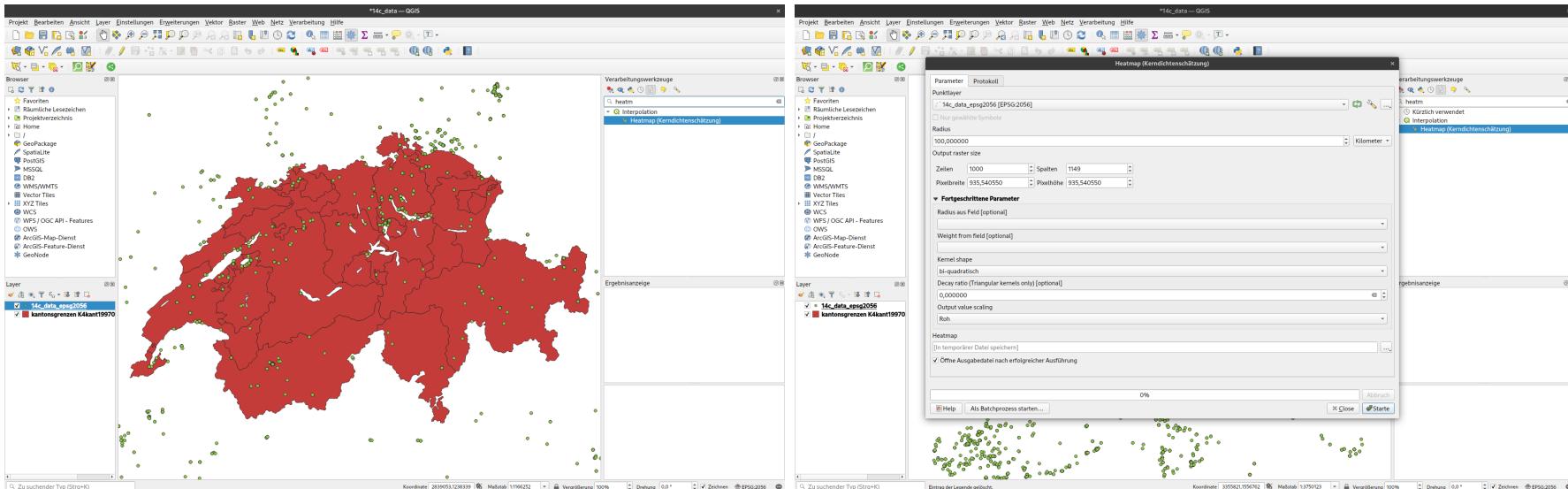
**Figure 6.** A comparison of KDE hotspot maps generated using the same bandwidth but with different cell sizes (a) 30 metres (PAI of 7.0) and (b) 240 metres (PAI of 7.0)

higher resolution results in smoother details, but are more memory, space and time consumptive

Source: Chainey 2013

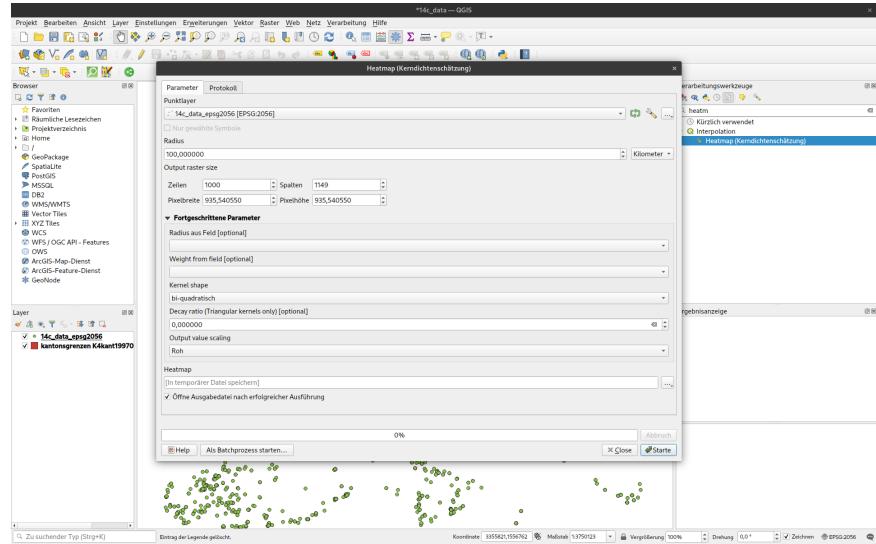
# Heatmap in QGIS (1)

- tool is 'hidden' in the toolbox
- type 'Heatmap' in the search field
- click on tool



# Heatmap in QGIS (2)

- select your point layer
- select the radius (ie. the bandwidth) in map units (meters, km, ...)
- select output size (in number of cells or with of pixels)
- advanced options:
  - individual radius and weighting can be taken from attribute table
  - shape of the kernel can be selected
  - output density can be scaled
- export to temporary folder
- add to map if finished
- click on 'Run'



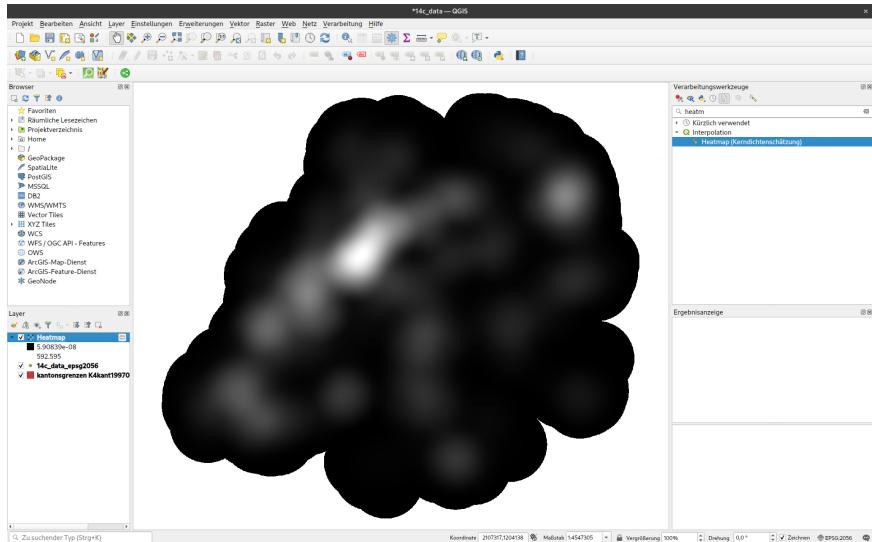
# Heatmap in QGIS (3)

## Result

- you should see a grayish blob over your points, lighter where more points are...

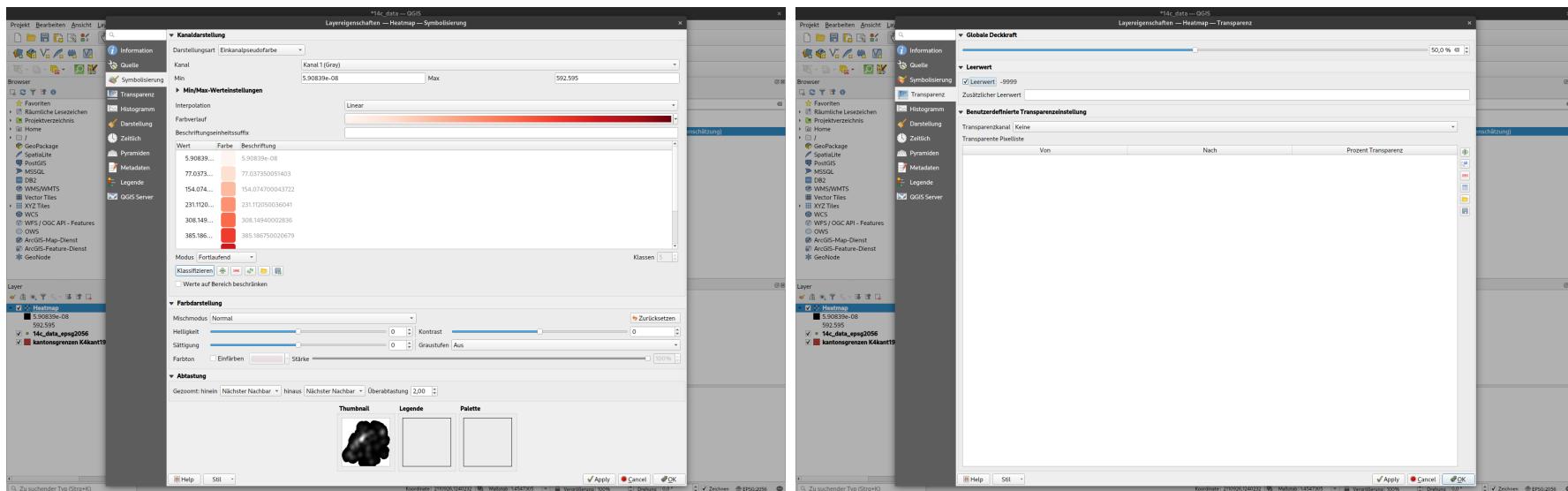
## Troubleshooting

- if you do not see anything, your bandwidth might have been to low...



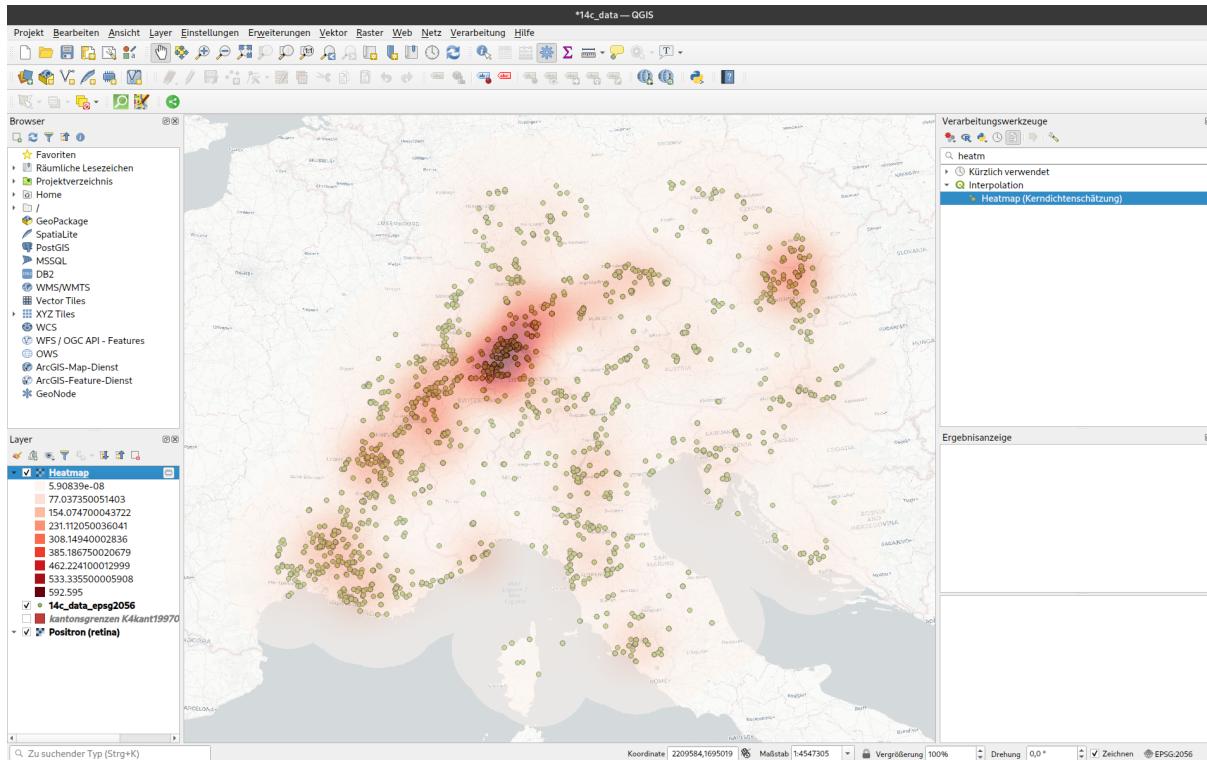
# Styling heatmaps in QGIS (1)

- heatmaps are best shown using a color ramp in pseudocolor and some transparency
- add layer properties > symbology
- select pseudocolor
- go to transparency and select eg. 50%



# Styling heatmaps in QGIS (2)

- alter color ramp and transparency until your satisfied
- you might like to add a background map and remove the cantonal borders

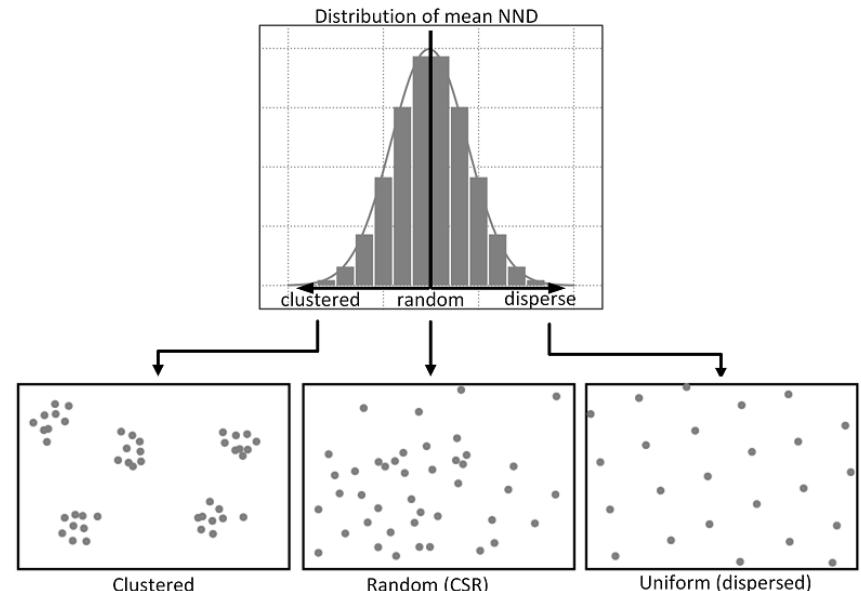


- enjoy the result

# Density (Point Pattern) Analysis

## Motivation

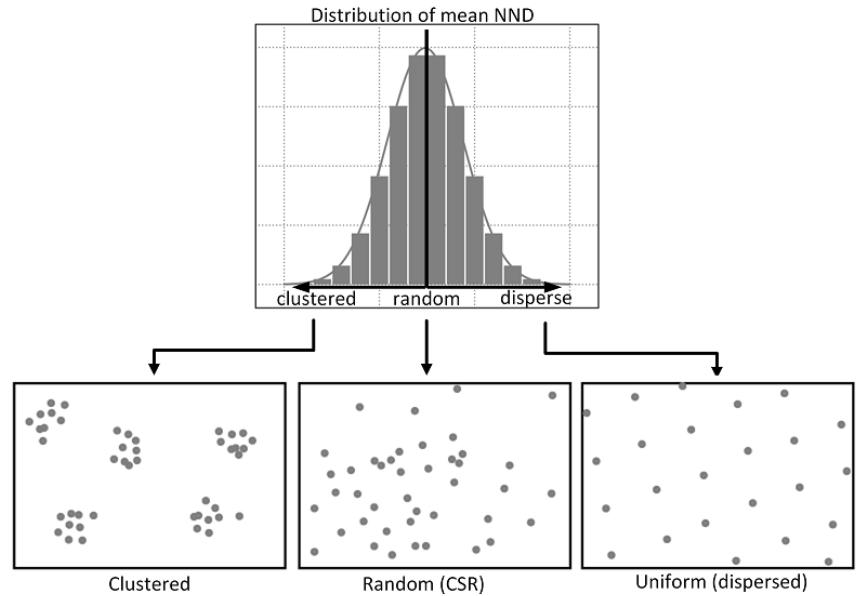
- most basically tries to answer the question if data are clustered, regular spaced or random
- uninfluenced spatial processes tend to produce random patterns
- Complete Spatial Randomness**



<https://gistbok.ucgis.org/>

# Archaeological Motivation

- clustered patterns result from 'attractive' processes
  - if the presence of one object makes other objects more **likely**
  - eg. burial sites: usually, we have burial grounds/areas, so that burials tend to be clustered in the landscape
- regular patterns result from 'repulsive' processes
  - if the presence of one object makes other objects more **unlikely**
  - eg. settlement sites: usually, settlements need some area/hinterland, so that settlements tend to be more regular in the landscape

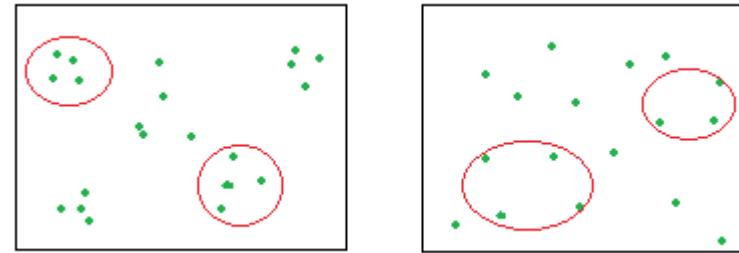


<https://gistbok.ucgis.org/>

# Nearest Neighbor Analysis

## Average Nearest Neighbor (ANN)

measures the average distance of neighboring data points from a given observation. If it is compared with a theoretical random average distance, it tells much about whether data points are clustered or dispersed.



Average distance of nearest neighbors are shorter in the left. Source:  
<https://towardsdatascience.com>

$$\text{NearestNeighborDistance} = d(\text{NN}) = \frac{\sum_{i=1}^n \min(d_{ij})}{N}$$

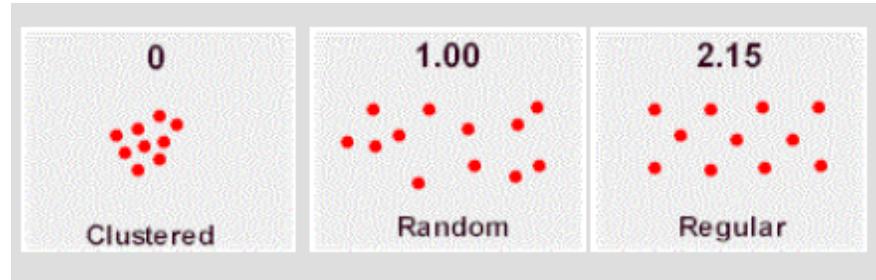
$$\text{TheoreticalRandomDistance} = d(\text{ran}) = 0.5 * \sqrt{\frac{A}{N}}$$

$$\text{NearestNeighborIndex} = \text{NNI} = \frac{d(\text{NN})}{d(\text{ran})}$$

# Nearest Neighbor Analysis

$$\text{NearestNeighborIndex} = NNI = \frac{d(NN)}{d(ran)}$$

- if the index is **below** 1, the points tend to be clustered
- if the index is **above** 1, the points tend to be regular
- if the index is **around** 1, the points tend to be random

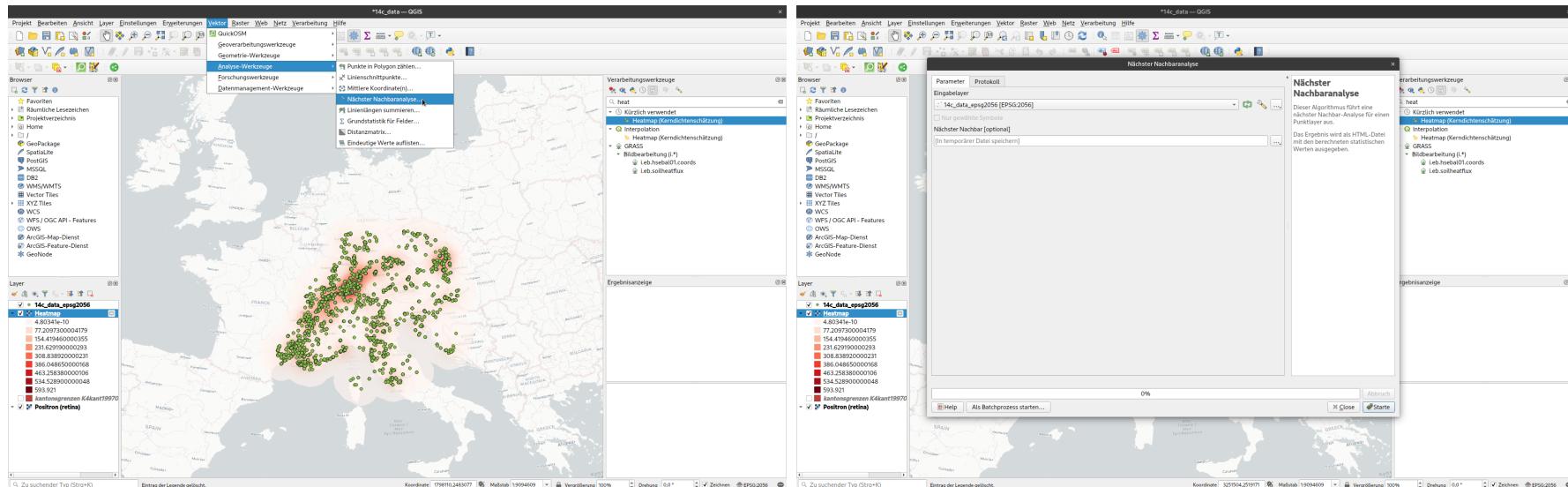


Average distance of nearest neighbors are shorter in the left. Source:  
<https://towardsdatascience.com>

# Nearest Neighbor Analysis in QGIS (1)

*(not the most developed function in QGIS yet)*

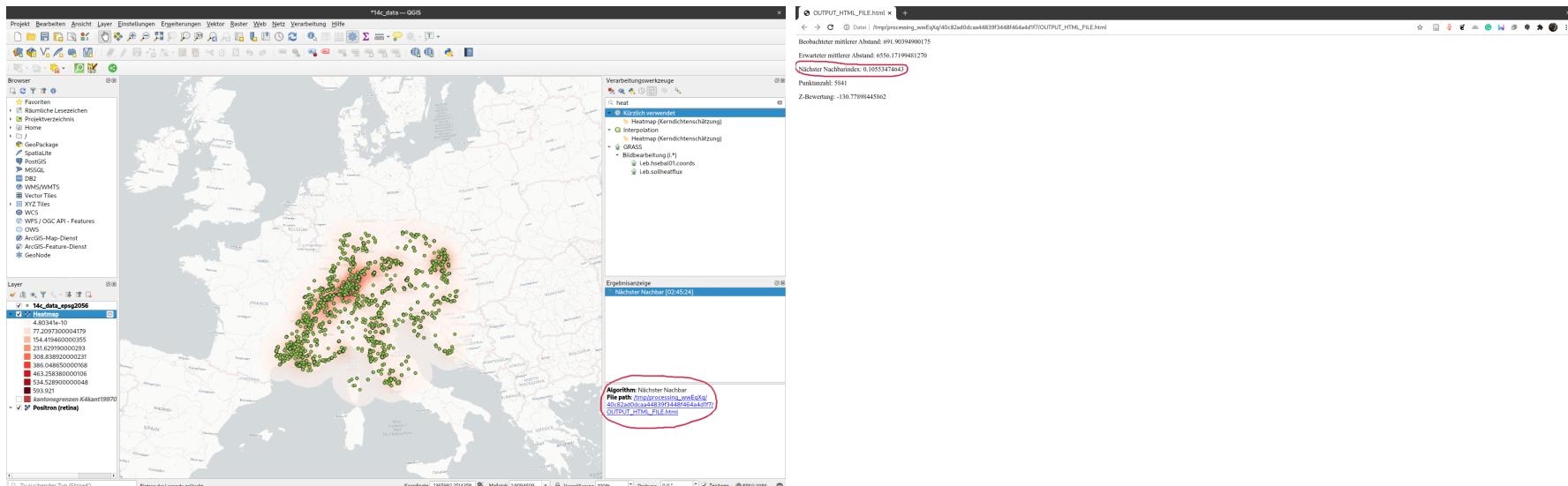
- Select 'Nearest Neighbor Analysis' from 'Vector > Analytical Tools'
- Select input Layer and press 'Run'



# Nearest Neighbor Analysis in QGIS (2)

(not the most developed function in QGIS yet)

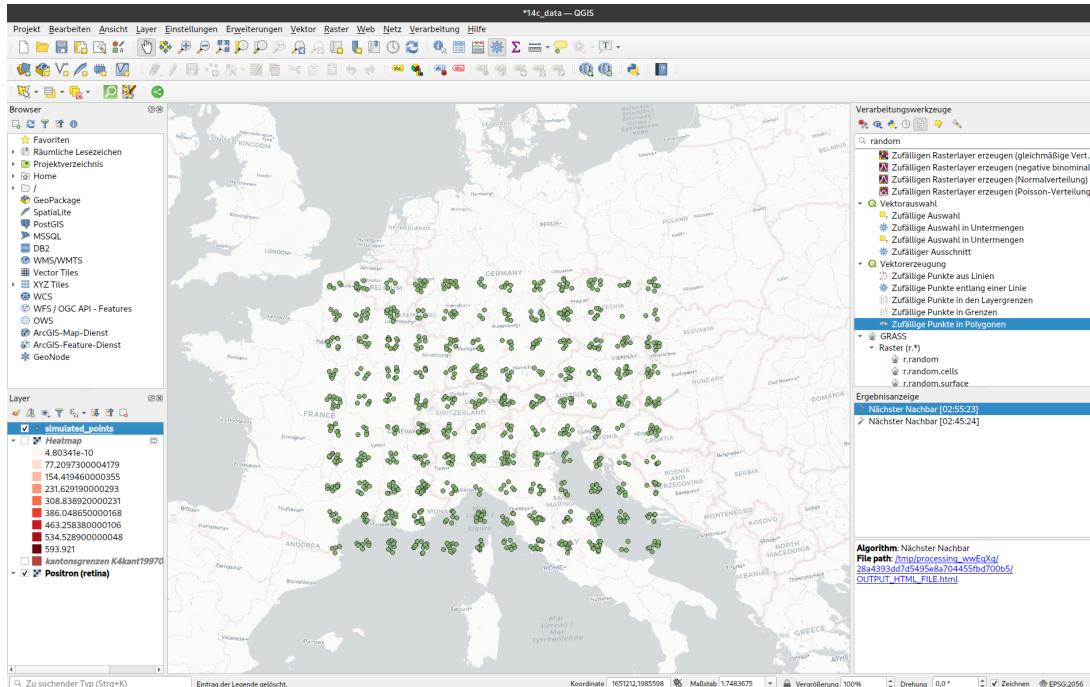
- Result not obviously present...
- Click on Link in the 'Result' pane
- result should pop up in a browser window. Here: Strongly clustered!



# Nearest Neighbor Analysis in QGIS - try it out

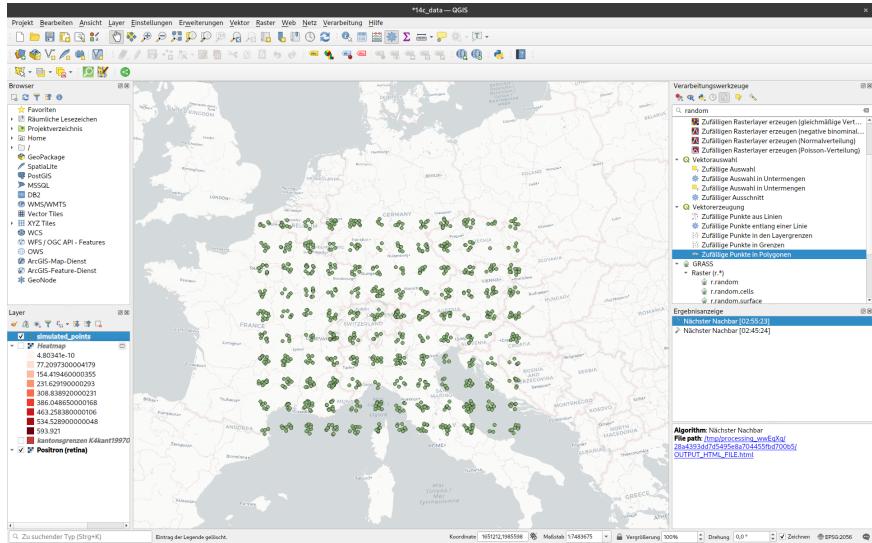
(not the most developed function in QGIS yet)

- Load the **this point layer** and add it to your map
- Run Nearest Neighbor Analysis
- Is the result consistent with your expectations?



# Nearest Neighbor Analysis - Restrictions

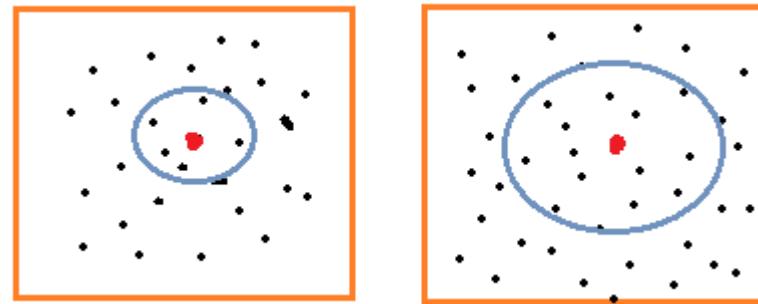
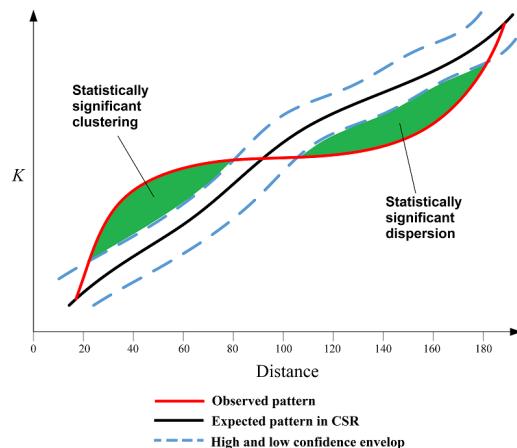
- what we consider regular is strongly scale dependend
- also, some processes might be regular in one scale, while clustered in another
- eg. burials around settlements:
  - Settlements might be regular spaced
  - Burials might be clustered around settlements
  - if we have only the burials, in the landscape they might have a similar structure like the example



# Neighbor Analysis over multiple scales

## The K function

The **K-Function** (or **Ripley's K**) measures distances between all points in space rather than just the neighbors as in ANN. It also helps to understand how clustering or dispersion of data occurs in different distances from its center (the centroid).



K-Function describing how clustering occurs in different scanning horizon from the center. Source: <https://towardsdatascience.com>

# K-function in QGIS

There is none...

Sorry!

But we can implement one ourself:

Using R within QGIS

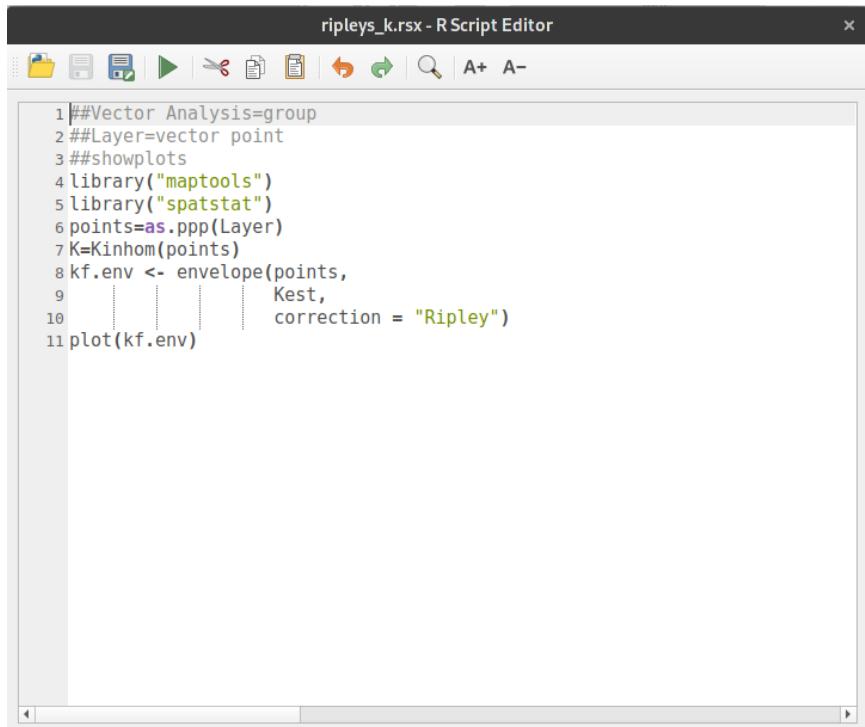
*(R is a very powerful statistical environment/language! Learn more next semester in my statistics course...)*

# Using R in QGIS (1)

- In the Toolbox , click on the R Symbol in the upper toolbar
- Select 'Create New R Script'

# Using R in QGIS (2)

- in the small window, paste in the code on the right side
- save it as 'ripleys\_k'



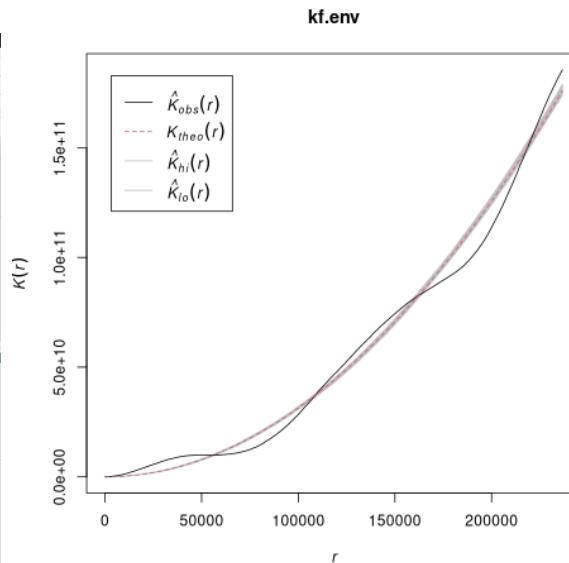
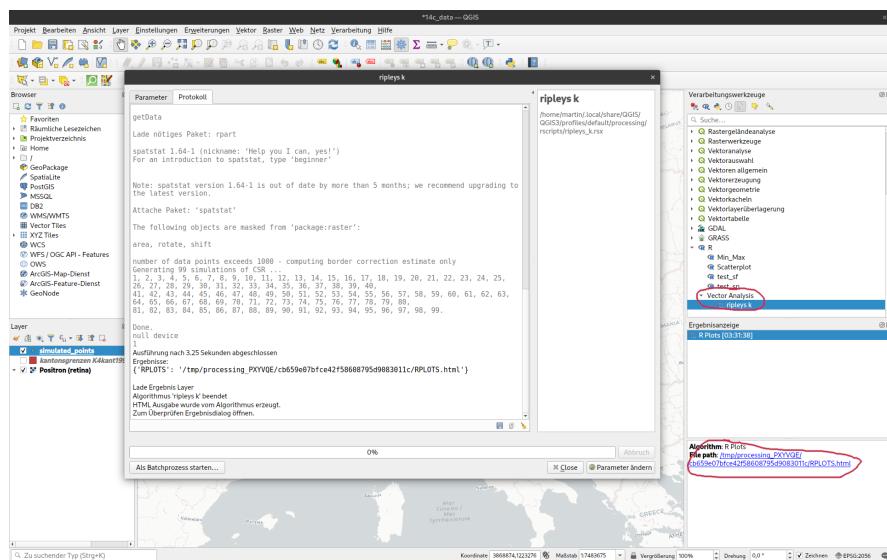
```
ripleys_k.rsx - R Script Editor
| File | New | Open | Save | Run | Stop | Help | A+ | A-
1 ##Vector Analysis=group
2 ##Layer=vector point
3 ##showplots
4 library("maptools")
5 library("spatstat")
6 points=as.ppp(Layer)
7 K=Kinhom(points)
8 kf.env <- envelope(points,
9           Kest,
10          correction = "Ripley")
11 plot(kf.env)
```

```
##Vector Analysis=group
##Layer=vector point
##showplots
library("maptools")
library("spatstat")
points=as.ppp(Layer)
K=Kinhom(points)
kf.env <- envelope(points,
                     Kest,
                     correction = "Ripley")
plot(kf.env)
```

# Our own K-function in QGIS

Now, we can use this function to analyse our layer:

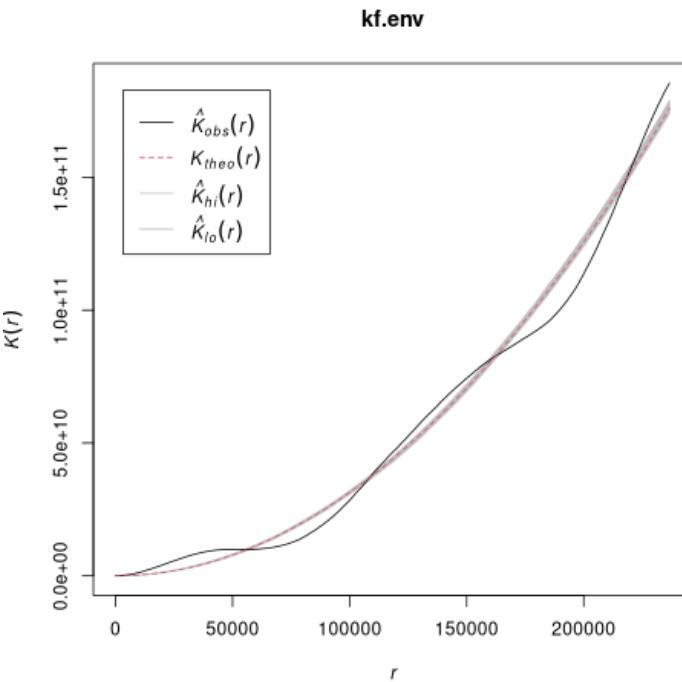
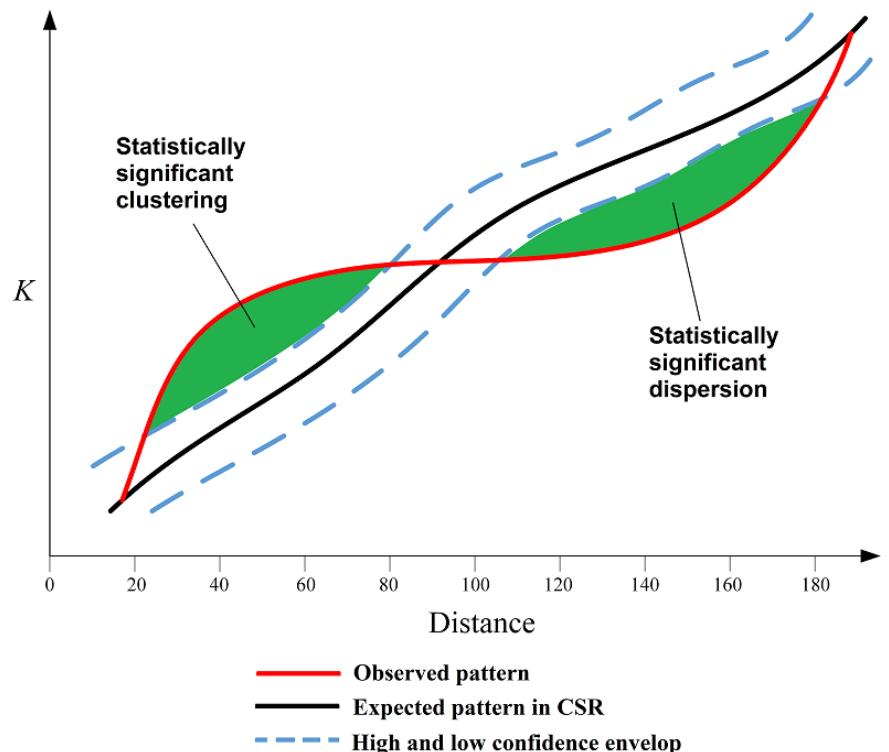
- in the toolbox, find the script under 'R > Vector Analysis > ripleys k'
- Open it, select the simulated points as layer
- 'Run'
- Inspect the result in by clicking the link in the Result Pane



# Result Ripley's K

- below 50 km Distance, we have clustering
- above 50 km Distance, we have a regular pattern

(I simulated the data accordingly...)



# What We've Covered

- Reprojecting Vector Layers
- Creating Heatmaps
- Nearest Neighbor Analysis to identify global Regularity/Clustering
- Ripley's K function to identify Regularity/Clustering on different spatial scales

*(this is already quite advanced stuff)*

# Homework

- Select a arbitrary point layer (the more archaeological the better)
- Make a heatmap
- Conduct a Nearest Neighbor Analysis
- Send me the shapefile of the points, a map of the resulting heatmap and the result of the Nearest Neighbor Analysis

# Any questions?



You might find the course material (including the presentations) at

[https://github.com/MartinHinz/gia\\_hs\\_2020](https://github.com/MartinHinz/gia_hs_2020)

You can see the rendered presentations at

[http://martinhinz.github.io/gia\\_hs\\_2020](http://martinhinz.github.io/gia_hs_2020)

You can contact me at

[martin.hinz@iaw.unibe.ch](mailto:martin.hinz@iaw.unibe.ch)

Source: <https://www.instagram.com/sadtopographies>