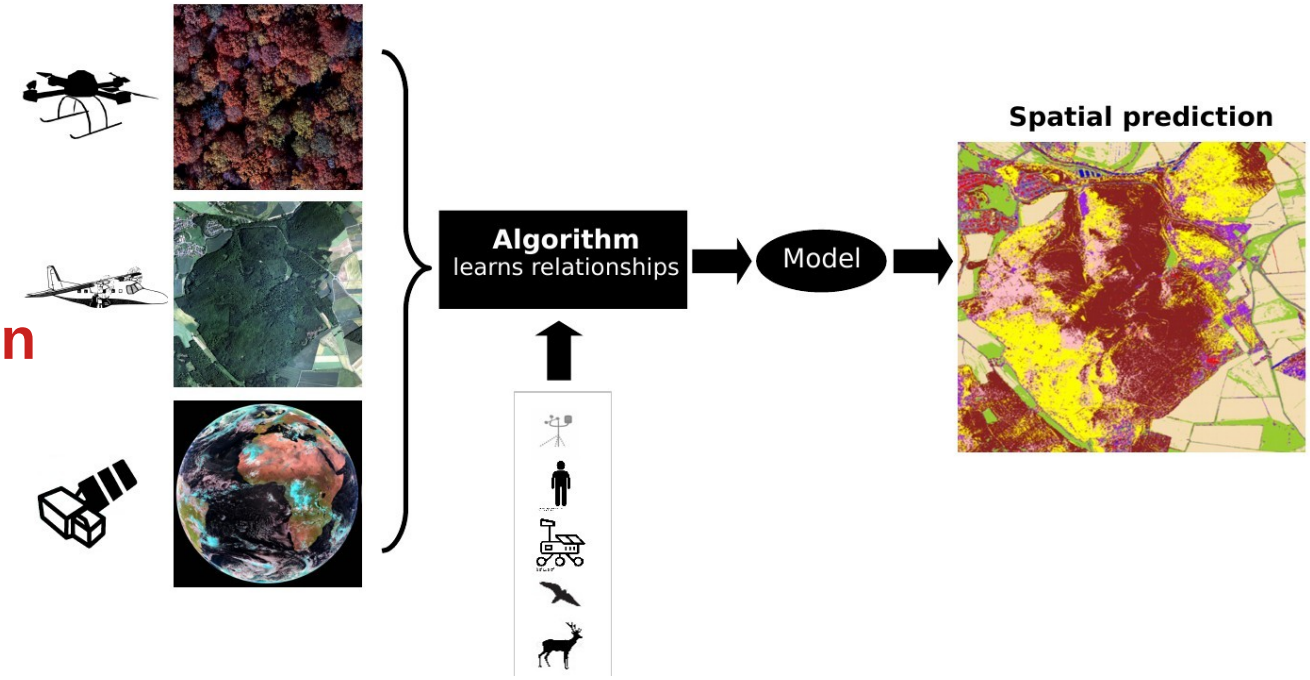
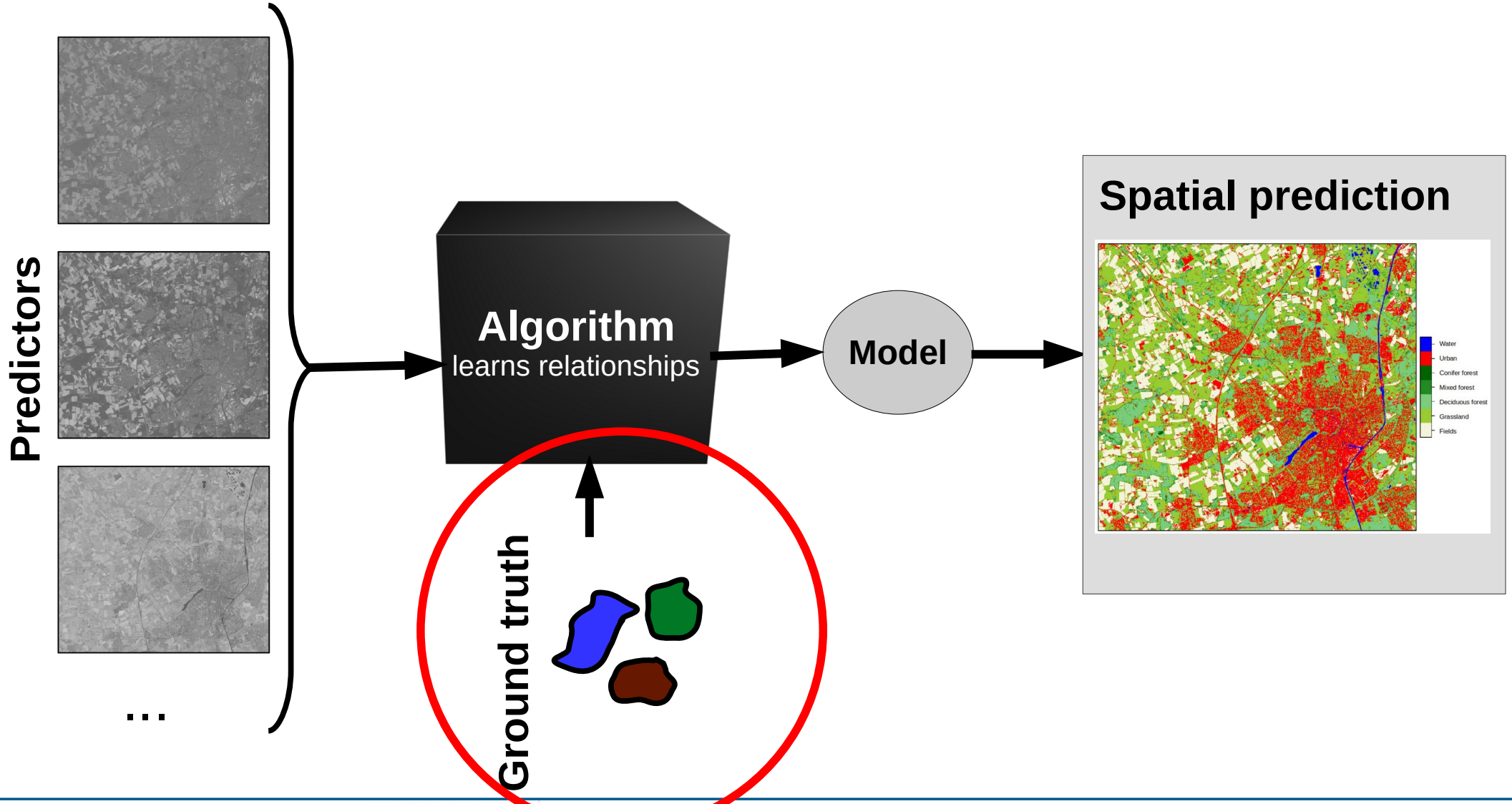
# Remote sensing and machine learning:
## Towards a spatio-temporal continuous monitoring of the environment

***Hanna Meyer***
Remote Sensing & Spatial Modelling,
Institute of Landscape Ecology, WWU Münster

## Part 3: Training data for Land cover classification



# How to use the spectral properties to classify land cover?

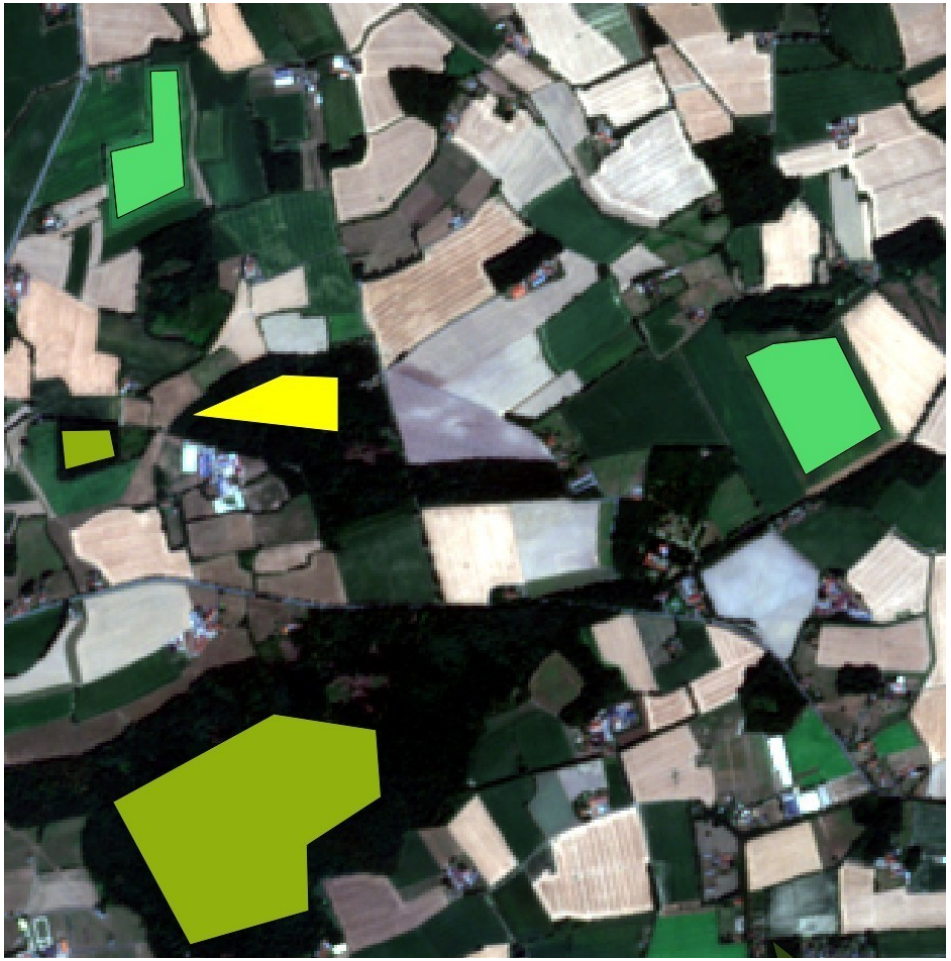


# Define the response variable

- What are the dominant land cover types?
- How much detail is required?
- Example:
  - Urban
  - Open soil
  - Grassland
  - Forest
  - Water

# How to use the spectral properties to classify land cover?

## Reference data are required



- Training data from field work, expert knowledge, existing databases,…
- Typically polygons. Why?

# Training polygons for land cover classification

- How many?
  - Hard to say ;)
  - Here: at least 3 polygons per class
- Where?
  - Try to cover spectral variability of the classes
  - Keep in mind: Each pixel will be handled as ONE training point. No need to produce huge polygons on homogeneous areas

Example of “open soil” with various spectral properties

# Training polygons for land cover classification

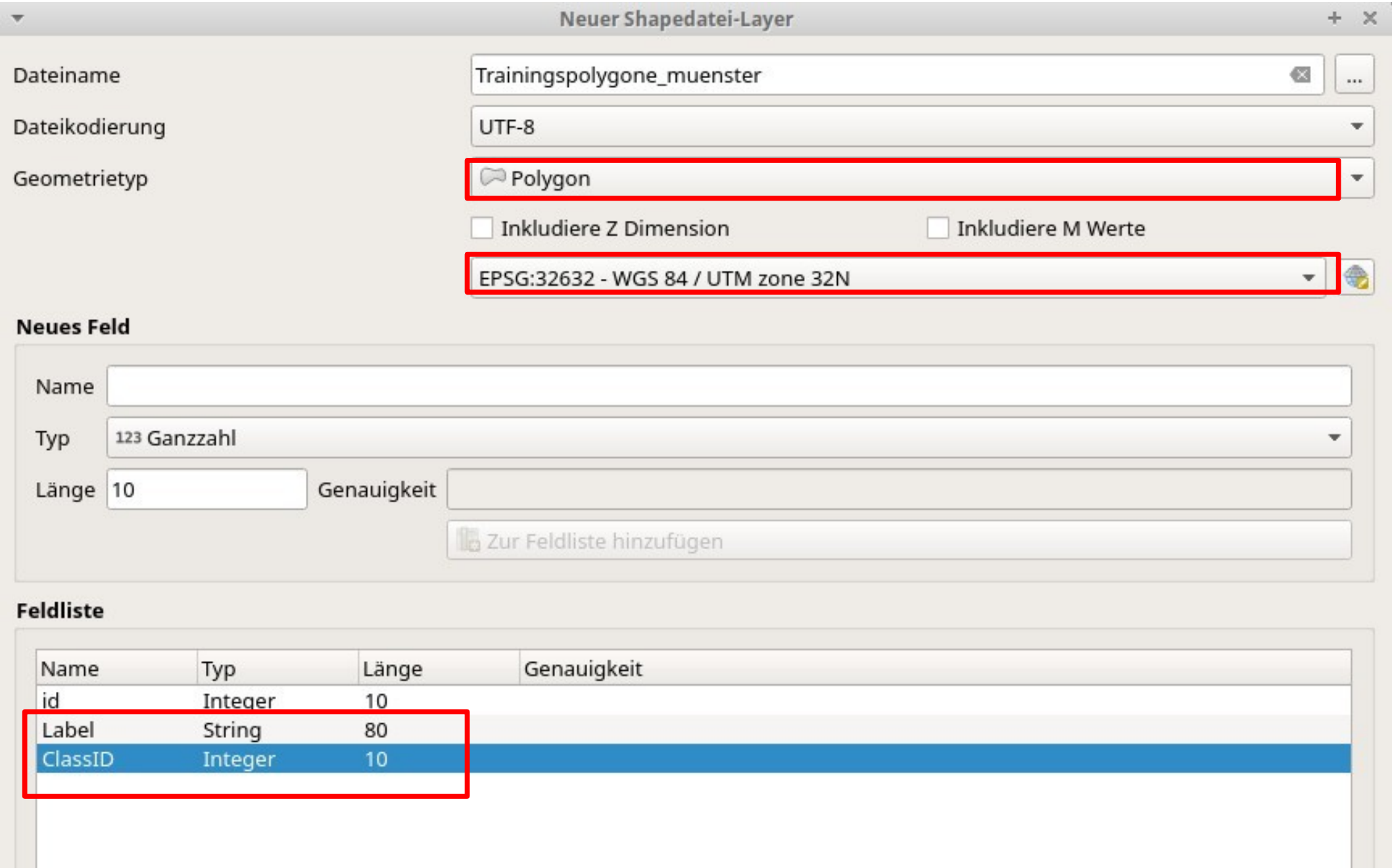
- Keep in mind:
  - Use only “pure” pixels
  - Google Earth as background useful but check for spatial and temporal differences

Get Basemaps:
https://gis.stackexchange.com/questions/20191/adding-basemaps-from-google-or-bing-in-qgis

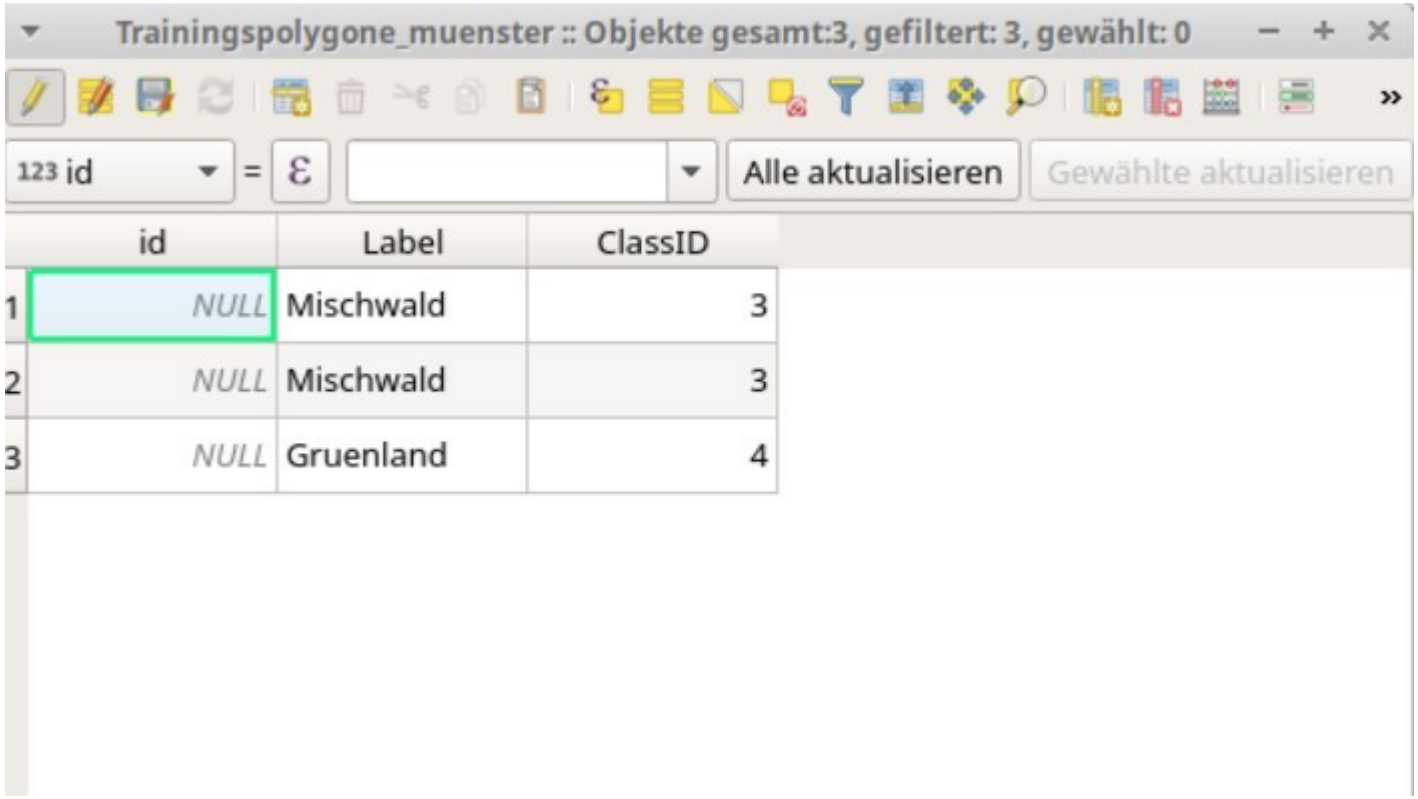
# Create training polygons

- In R: Mapedit
- ...or use QGIS to create a new vector layer
  - On the basis of the satellite image
  - Make use of high resolution background maps

# Create training polygons (QGIS)



# Create training polygons (QGIS)

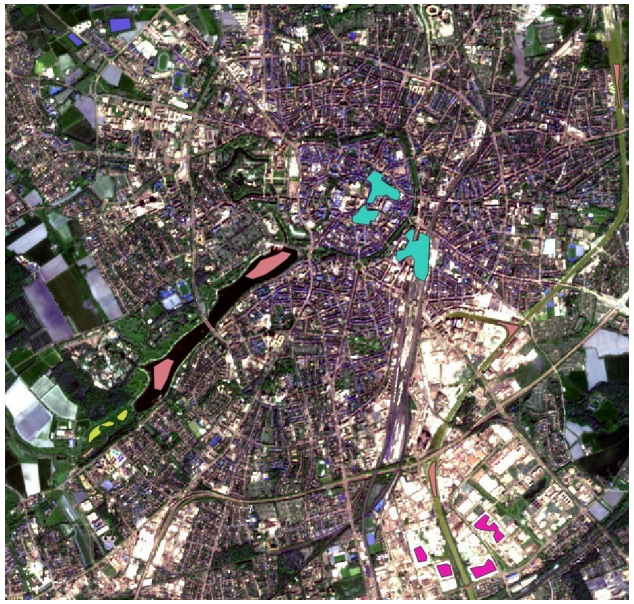



| | id | Label | ClassID |
|---|---|---|---|
| 1 | NULL | Mischwald | 3 |
| 2 | NULL | Mischwald | 3 |
| 3 | NULL | Gruenland | 4 |

# Create training polygons

- Digitize training polygons for your region
  - Cover all relevant land cover classes
  - At least 3 polygons per class
  - Use the projection of the satellite image
  - Save the polygons as geopackage (.gpkg)
- Load the data into R (?sf::read_sf)

# Combine predictors and response



| | B02 | B03 | B04 | B08 | ... | Class |
|---|---|---|---|---|---|---|
| 1 | 857 | 632 | 387 | 308 | | Water |
| 2 | 848 | 633 | 389 | 312 | | Water |
| 3 | 843 | 624 | 357 | 343 | | Water |
| 4 | 854 | 630 | 360 | 333 | | Water |
| 5 | 854 | 628 | 376 | 302 | | Water |
| 6 | 859 | 615 | 364 | 350 | | Water |

**How to do it in R**

```
extr <- extract(sen,trainingsites)
trainingsites$PolyID <- 1:nrow(trainingsites)
extr <- merge(extr,trainingsites,by.x="ID",by.y="PolyID")
```