Linguistic traits as heritable units? Spatial Bayesian clustering reveals Swiss German dialect regions

Supplementary materials

Noemi Romano* Sandro Bachmann† Peter Ranacher‡ Stéphane Joost§

Contents

This document contains some of the results from TESS analysis using data of the Syntactic Atlas of Swiss German (SADS).

- For raw results for all models, please visit: Raw results
- To explore the interactive maps, please visit: Mapping Swiss German dialects

*Laboratory of Geographic Information Systems (LASIG), School of Architecture,Civil and Environmental Engineering (ENAC), EPFL, Lausanne, Switzerland, +41 21 693 57 87, Route Cantonale Station 18 CH-1015 Lausanne
†German Department, University of Zurich (UZH), Zurich, Switzerland
‡Language and Space, University of Zurich (UZH), Zurich, Switzerland & Department of Geography, University of Zurich (UZH), Zurich, Switzerland
§Laboratory of Geographic Information Systems (LASIG), School of Architecture,Civil and Environmental Engineering (ENAC), EPFL, Lausanne, Switzerland, +41 21 693 57 87, Route Cantonale Station 18 CH-1015 Lausanne

Results with 6 populations

These results have been obtained with the Admixture model of TESS using the least correlated syntactic phenomena of the Syntactic Atlas of Swiss German (SADS). Choosing 6 populations (K=6), we obtain the following results.

Loading libraries

```
library(tmap)
library(sf)
library(pophelper)
library(dplyr)
library(RColorBrewer)
```

Reading data

Read combined results with CLUMPP and merge them

```
#Read clumpp results
K6 <- readQ("../../../Results/CLUMPP_output/pop_K6-combined-merged.txt")

#Merging results of 6 populations
merged_data <- mergeQ(K6)[[1]]

#Import coordinates
coord<-read.table('../../coordinates_wgs84.txt',header = T)

#Spatialize results
k6_results<-cbind(coord,merged_data)

#Convert data to sf format
point_data<-st_as_sf(k6_results, coords = c("X_1", "Y_1"), crs = 4326)

#Read municipalities
municipalities <- st_read("../../../shapefiles/municipalities_voronoi.shp")
```

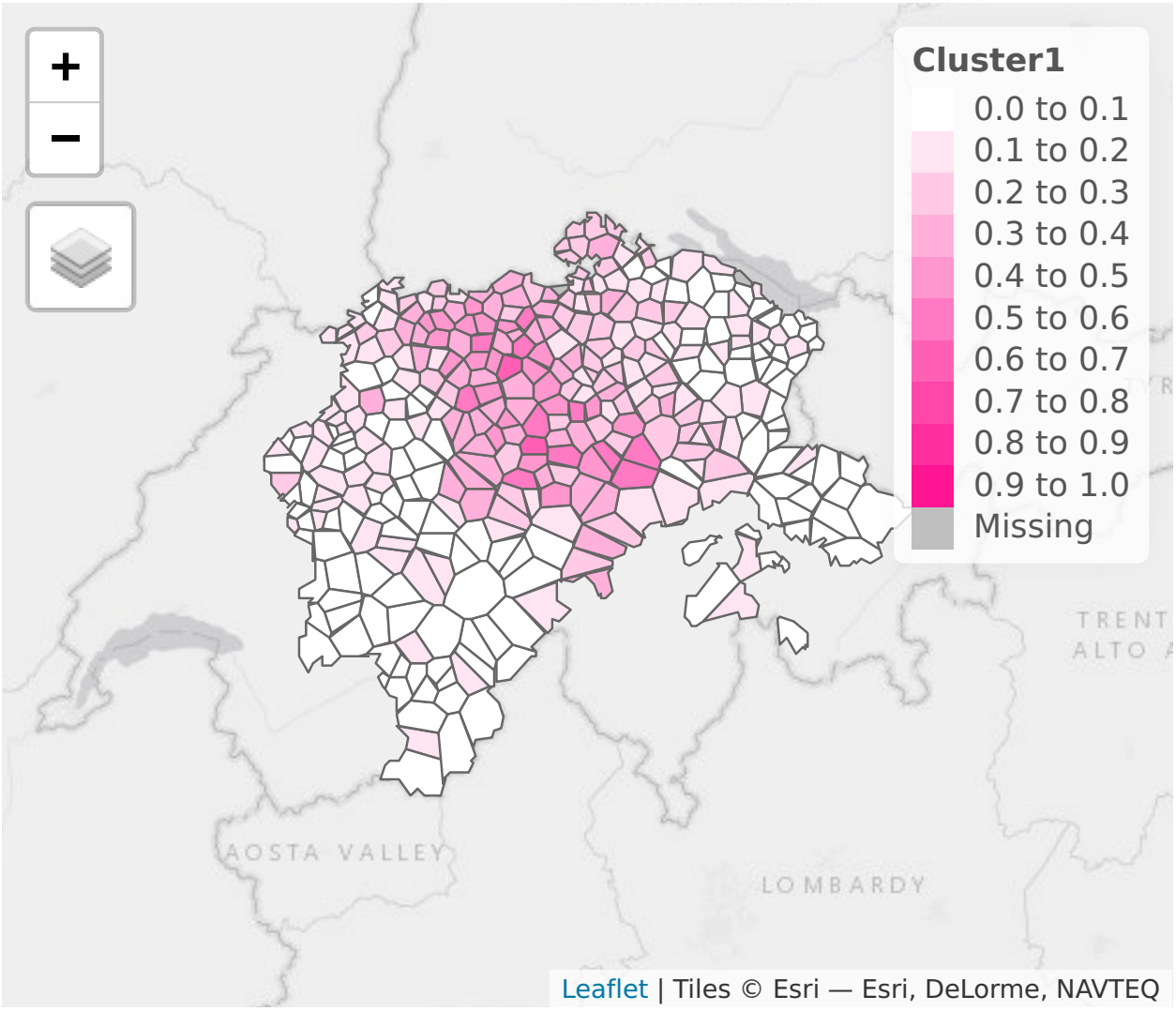
```
## Reading layer `municipalities_voronoi' from data source `/home/noe/Desktop/Paper/Romano_PdM/shapefil
## Simple feature collection with 356 features and 0 fields
## geometry type:  MULTIPOLYGON
## dimension:      XY
## bbox:           xmin: 7.025109 ymin: 45.91675 xmax: 10.09691 ymax: 47.80846
## CRS:            4326
```

```
#Create ID in municipalities
municipalities$id <- 1:nrow(municipalities)
```

Results

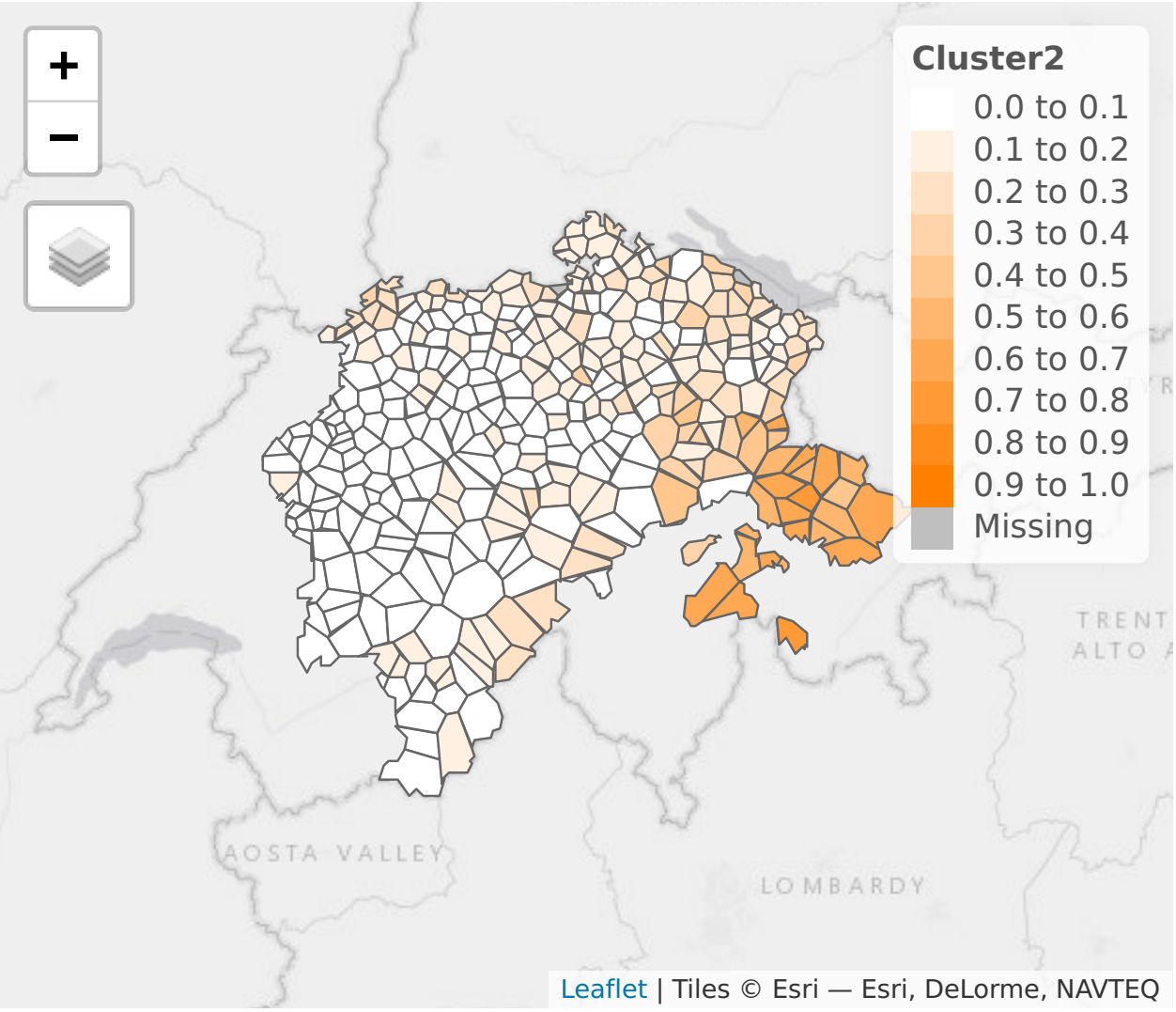
Population 1

```
#Defining my palet of colors
mypaletTESS<-c("#1F78B4","#33A02C","#E31A1C","#FF7F00","#6A3D9A",'deeppink1')
palette <-colorRampPalette(c("white", mypaletTESS[6]))

#Defining breaks
breaks <- c(0, 0.1, 0.2,0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1)

population_1_aggregated = municipalities %>%
  st_join(point_data[,c('Cluster1','geometry')]) %>%
  group_by(id) %>%
  summarize(Cluster1 = mean(Cluster1, na.rm = TRUE))

tmap_mode("view")
tm_shape(population_1_aggregated) +
tm_polygons("Cluster1", palette = palette(15), breaks = breaks)
```



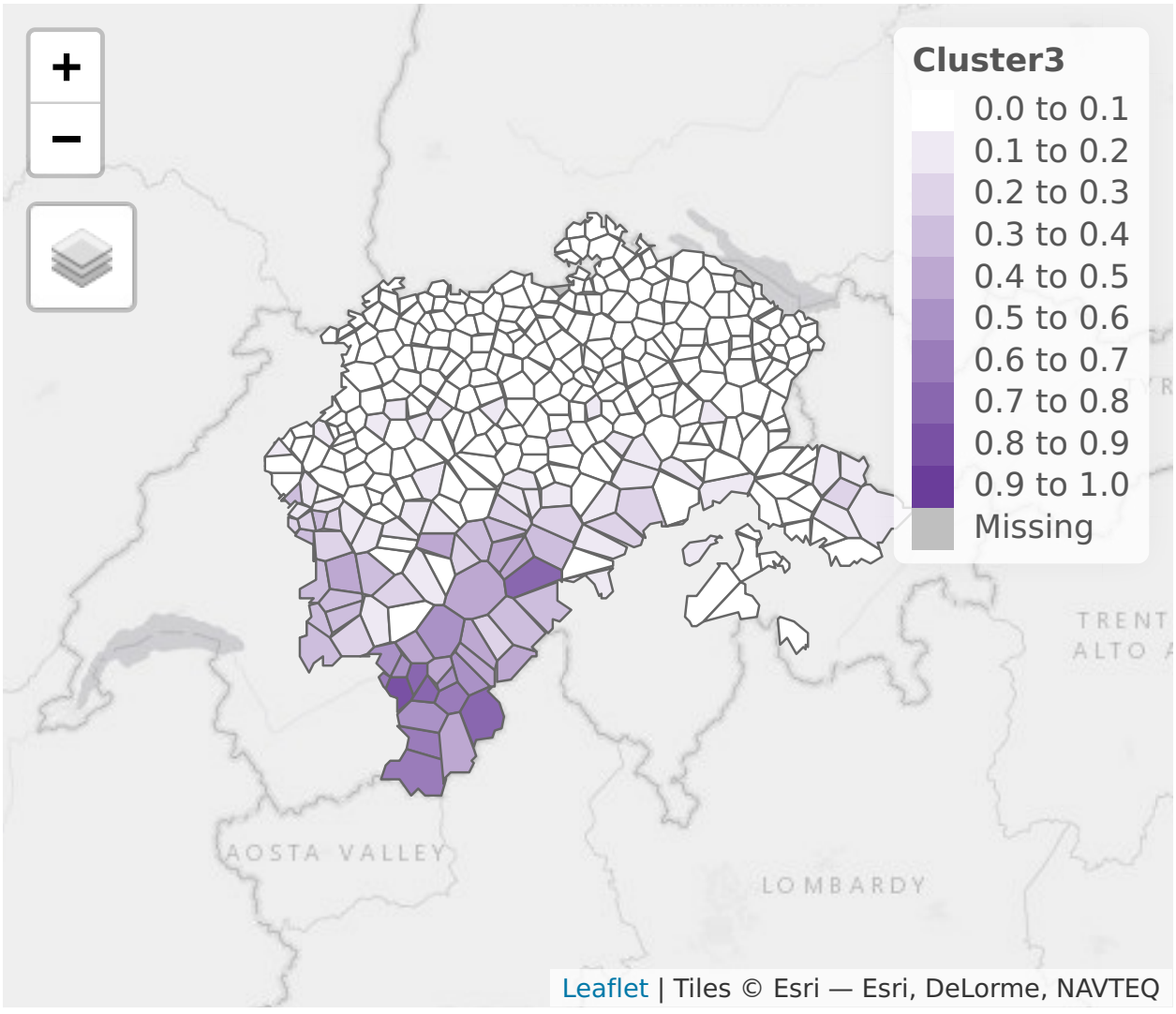
Population 2

```
palette <-colorRampPalette(c("white", mypaletTESS[4]))
population_2_aggregated = municipalities %>%
  st_join(point_data[,c('Cluster2','geometry')]) %>%
  group_by(id) %>%
  summarize(Cluster2 = mean(Cluster2, na.rm = TRUE))

tmap_mode("view")
tm_shape(population_2_aggregated) +
tm_polygons("Cluster2", palette = palette(15), breaks = breaks)
```



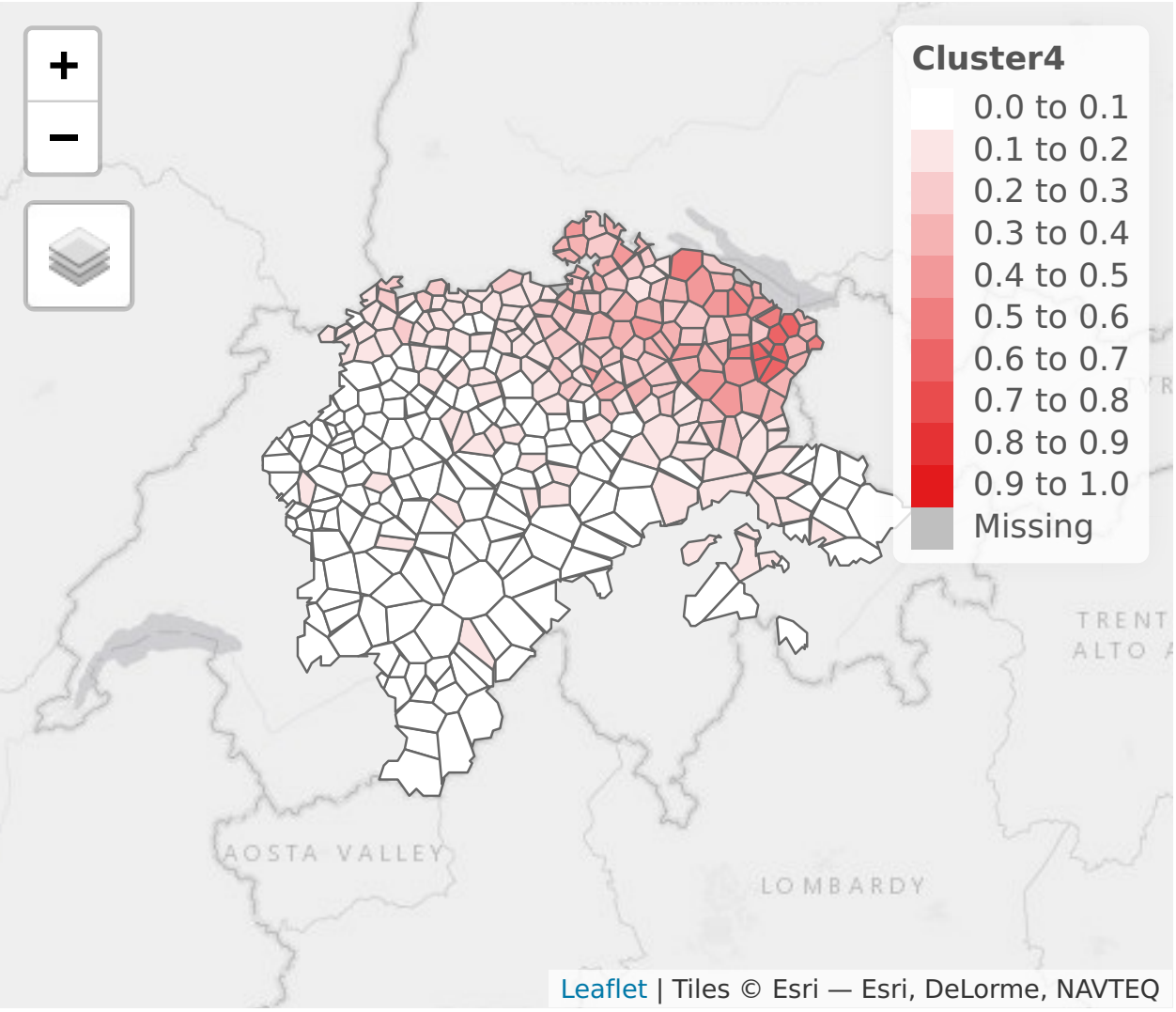
Population 3

```
palette <-colorRampPalette(c("white", mypaletTESS[5]))

population_3_aggregated = municipalities %>%
  st_join(point_data[,c('Cluster3','geometry')]) %>%
  group_by(id) %>%
  summarize(Cluster3 = mean(Cluster3, na.rm = TRUE))

tmap_mode("view")
tm_shape(population_3_aggregated) +
tm_polygons("Cluster3", palette = palette(15), breaks = breaks)
```



Population 4

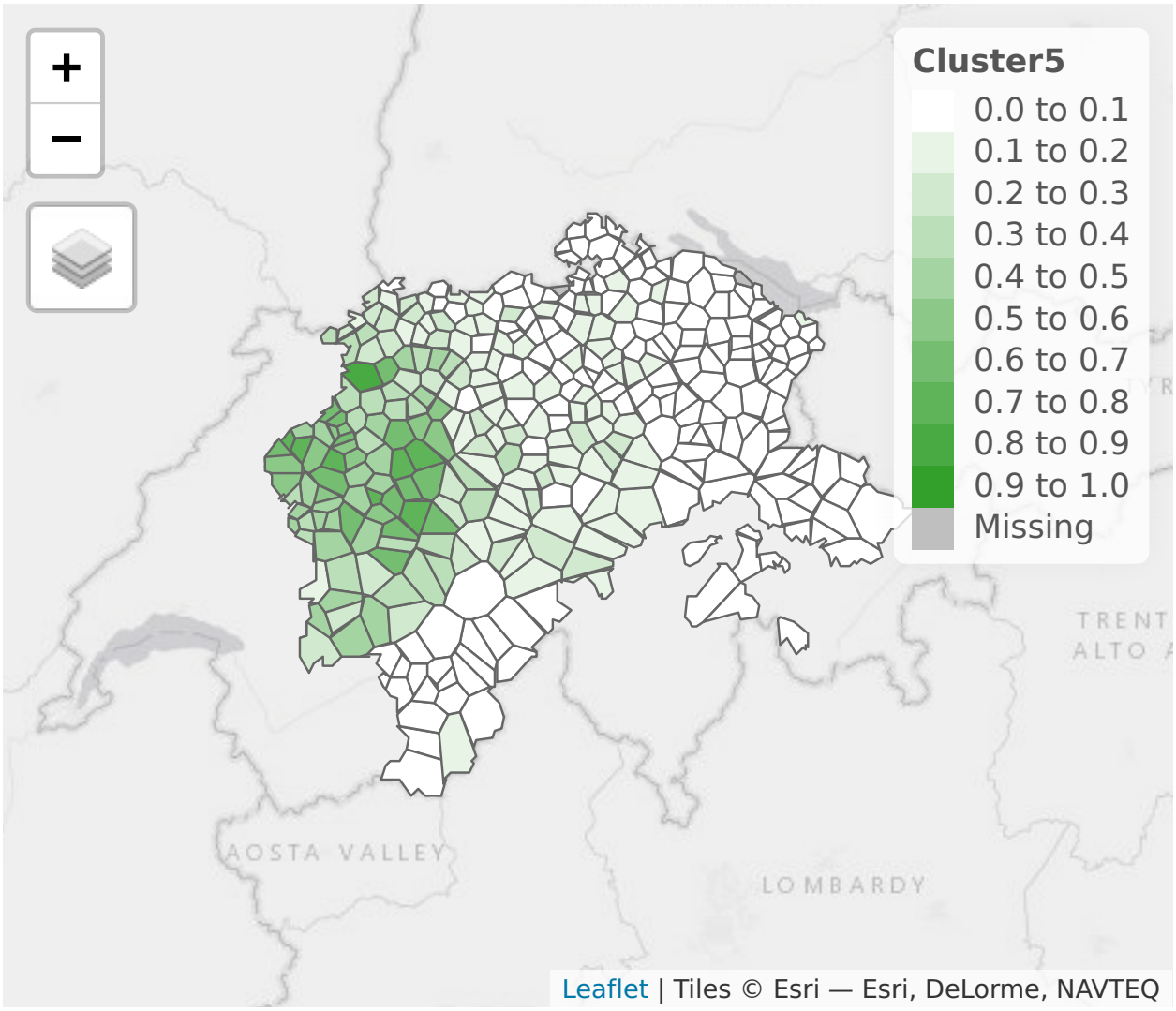
```
palette <-colorRampPalette(c("white", mypaletTESS[3]))

population_4_aggregated = municipalities %>%
  st_join(point_data[,c('Cluster4','geometry')]) %>%
  group_by(id) %>%
  summarize(Cluster4 = mean(Cluster4, na.rm = TRUE))

tmap_mode("view")
tm_shape(population_4_aggregated) +
tm_polygons("Cluster4", palette = palette(15), breaks = breaks)
```



Population 5

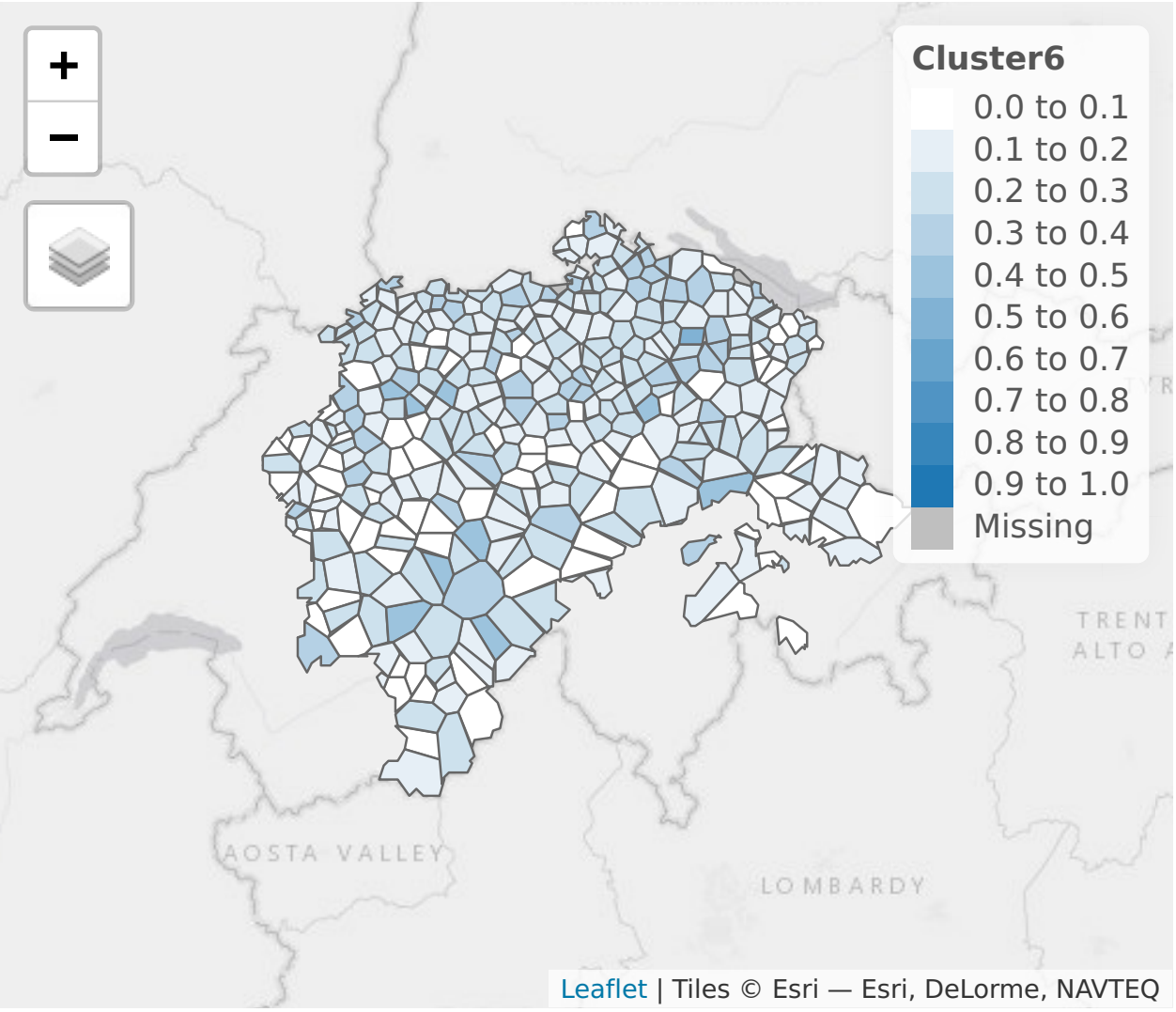
```
palette <-colorRampPalette(c("white", mypaletTESS[2]))

population_5_aggregated = municipalities %>%
  st_join(point_data[,c('Cluster5','geometry')]) %>%
```

```
  group_by(id) %>%
  summarize(Cluster5 = mean(Cluster5, na.rm = TRUE))

tmap_mode("view")
tm_shape(population_5_aggregated) +
tm_polygons("Cluster5", palette = palette(15), breaks = breaks)
```



Population 6

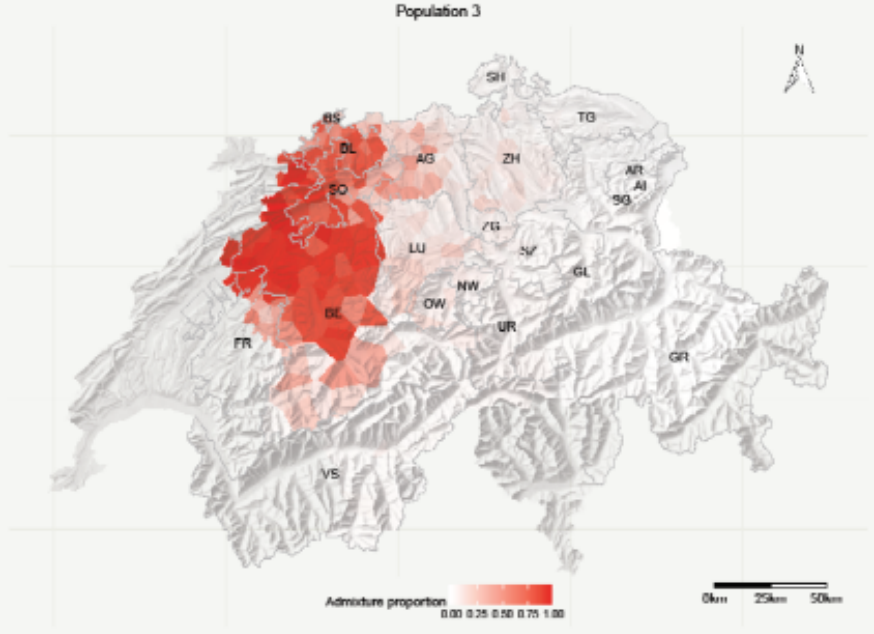
```
palette <-colorRampPalette(c("white", mypaletTESS[1]))

population_6_aggregated = municipalities %>%
  st_join(point_data[,c('Cluster6','geometry')]) %>%
  group_by(id) %>%
  summarize(Cluster6 = mean(Cluster6, na.rm = TRUE))

tmap_mode("view")
tm_shape(population_6_aggregated) +
tm_polygons("Cluster6", palette = palette(15), breaks = breaks)
```



Results with raw data

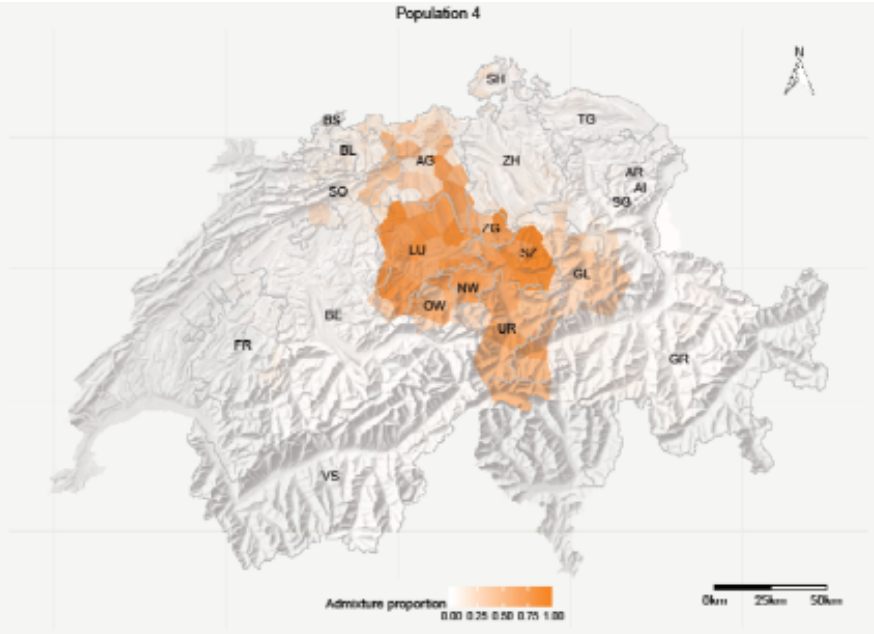
These results have been obtained with the Admixture model of TESS using raw data of the Syntactic Atlas of Swiss German (SADS).

Western population



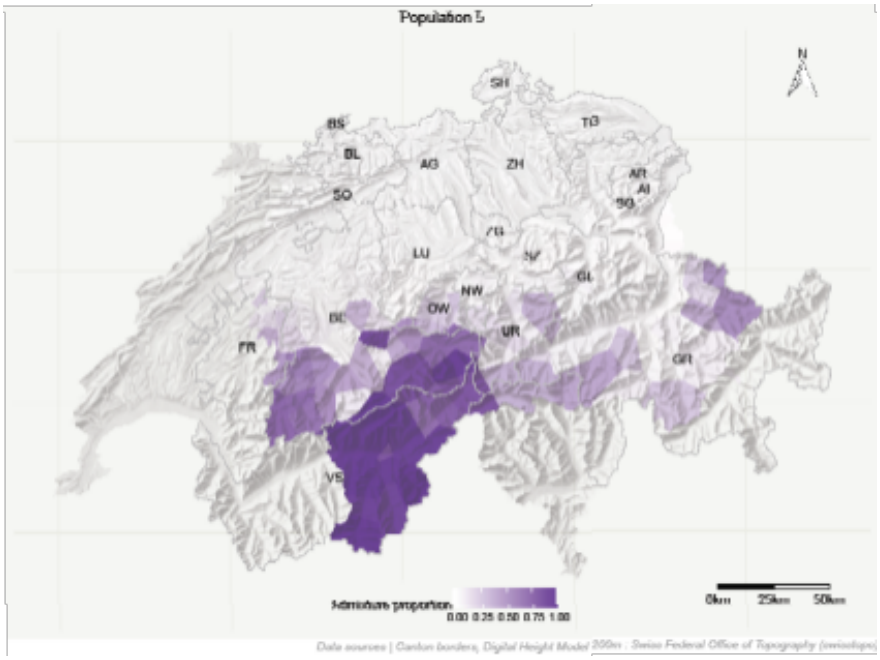
Data sources | Canton borders, Digital Height Model 200m : Swiss Federal Office of Topography (swisstopo)

Central population

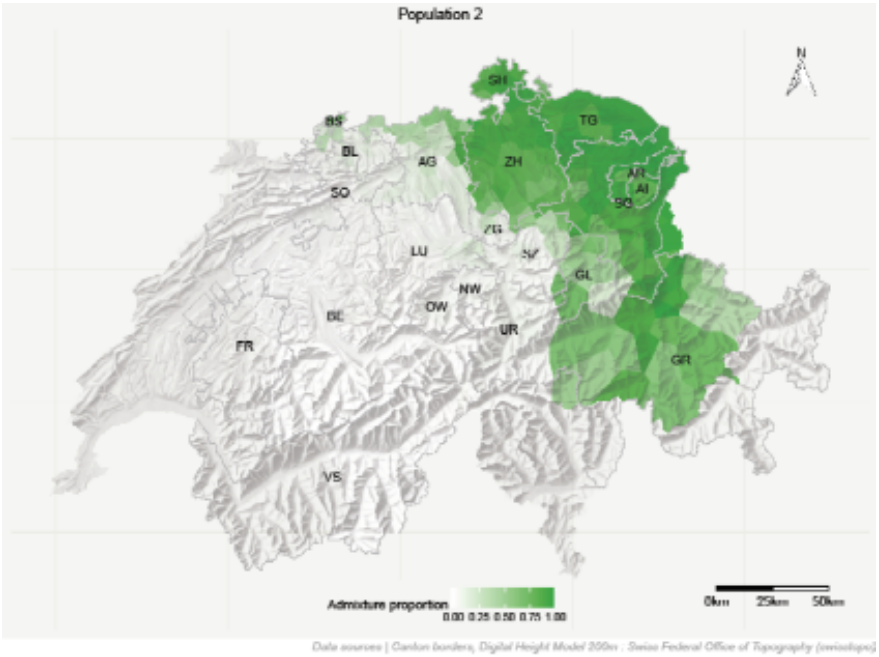


Data sources | Canton borders, Digital Height Model 200m : Swiss Federal Office of Topography (swisstopo)

Walser population



Eastern population



Ground population