# Week 1 - Leaflet

Christoffer M. Kramer

2/2/2021

## 1

**Describe a problem or question in your field for which spatial analysis could be applicable.**
Cultural boundaries is a much discussed topic within anthropology. These are questions such as: Does close proximity to between cultures strengthen or weaken cultural differences? What role does geographical and ecological differences play in the creation of cultural boundaries? How does the state mantain, create, alter or remove cultural boundaries? What influence does immigration have on cultural boundaries? How and why do minorities maintain cultural boundaries towards the majority population (is it discriminatory practices? is it forced upon them? is it an active choice?)

## 2

**List 5 data layers that you think are necessary to answer your question/solve your problem. Find on the internet github and then describe examples of two or three of your listed layers.**
1. Migration patterns layer 2. cultural boundaries layer 3. Terrain layer 4. National boundaries layer 5. Layer that shows minority populations within nation states.

A big problem with cultural boundaries is that it is hard to define. Therefore cultural boundaries tend to be studied through fieldwork. However, it is probably possible to combine fieldwork with statistical analysis in order to create a map of them. But I haven't been able to find any on the internet.

## 3

**Your colleague has found some ruins during a hike in the Blue Mountains and recorded the coordinates of structures on her phone(RCFeature.csv). She would like to map her points but has no computer or mapping skills. Can you make a map that she can work with using only a browser? She needs an interactive map that she can download to her computer and use straightaway.**

## 3.1

**Create a standalone .html map in Leaflet showing at least basic topography and relief, and load in the table of points. Make sure she can see the FeatureID, FeatureType and Description attributes when she hovers over the point marker**

Basic data-preparation

```
data_RCFeature <- read.csv("data/RCFeature.csv")
data_RCFeature <- tibble(data_RCFeature)
view(data_RCFeature)
```

Assign the base location to an object. I'm basing the location on one of the recorded coordinates and chosing zoom based on trial and error.

```
location_blue_mountains <- leaflet() %>%
  setView(150.2923, -33.75892, zoom = 20)
```

Choose maps (typography and relief, an image map and a streetmap) and add them as tiles to the base location

```
#esri <- grep("^Esri", providers, value = TRUE)
maps <- c("OpenTopoMap", "Esri.WorldImagery", "Esri.WorldStreetMap")
for (map in maps) {
  location_blue_mountains <- location_blue_mountains %>% addProviderTiles(map, gro
up = map)
}
```

Create map

```
map_artifacts <- location_blue_mountains %>%
  addLayersControl(baseGroups = maps,
                   options = layersControlOptions(collapsed = FALSE))
map_artifacts
```

Filter out NA's and add points to map

```
places <- data_RCFeature %>% filter(!is.na(Longitude))
map_artifacts <- map_artifacts %>%
  addCircleMarkers(lng = places$Longitude,
                   lat = places$Latitude,
                   label = places$FeatureID,
                   popup = paste0("Feature Type: ", places$FeatureType, ". Descrip
tion: ", places$Description, sep = " "))
map_artifacts
```

## 3.2

**Consider adding elements such as minimap() and measure() for easier map interaction**

Add minimap

```
map_artifacts <- map_artifacts %>%
  addLayersControl(baseGroups = maps,
                   options = layersControlOptions(collapsed = FALSE)) %>%
  addMiniMap(tiles = maps[[1]], toggleDisplay = TRUE,
             position = "bottomright")

map_artifacts
```

Add measure

```
map_artifacts <- map_artifacts %>%
 addMeasure(
    position = "bottomleft",
    primaryLengthUnit = "meters",
    primaryAreaUnit = "sqmeters",
    activeColor = "#3D535D",
    completedColor = "#7D4479")

map_artifacts
```

## 3.3

**Explore differentiating the markers (e.g. by size using Accuracy field)**

Let radius be determined by accuracy.

```
places <- data_RCFeature %>% filter(!is.na(Longitude))
map_artifacts <- map_artifacts %>%
  addCircleMarkers(lng = places$Longitude,
                   lat = places$Latitude,
                   label = places$FeatureID,
                   popup = paste0("Feature Type: ", places$FeatureType, ". Descrip
tion: ", places$Description, sep = " "),
                   radius = places$Accuracy)
map_artifacts
```

## 3.4

**Explore the option of clustering markers with addMarkers(clusterOptions = markerClusterOptions()). Do you recommend marker clustering here?**

```
places <- data_RCFeature %>% filter(!is.na(Longitude))
map_artifacts <- map_artifacts %>%
  addCircleMarkers(lng = places$Longitude,
                   lat = places$Latitude,
                   label = places$FeatureID,
                   popup = paste0("Feature Type: ", places$FeatureType, ". Descrip
```

```
tion: ", places$Description, sep = " "),
                  clusterOptions = markerClusterOptions(),
                  radius = places$Accuracy)
map_artifacts
```

Yes in this case I recommend clusters since some areas are densely packed with artifacts.

Save as HTML

```
map_artifacts <- map_artifacts %>%
htmlwidgets::onRender("function(el, x) {
                        var myMap = this;
                        myMap.on('baselayerchange',
                        function (e) {
                        myMap.minimap.changeLayer(L.tileLayer.provider(e.name));
                        })
                        }")
saveWidget(map_artifacts, "map_artifact.html", selfcontained = TRUE)
```