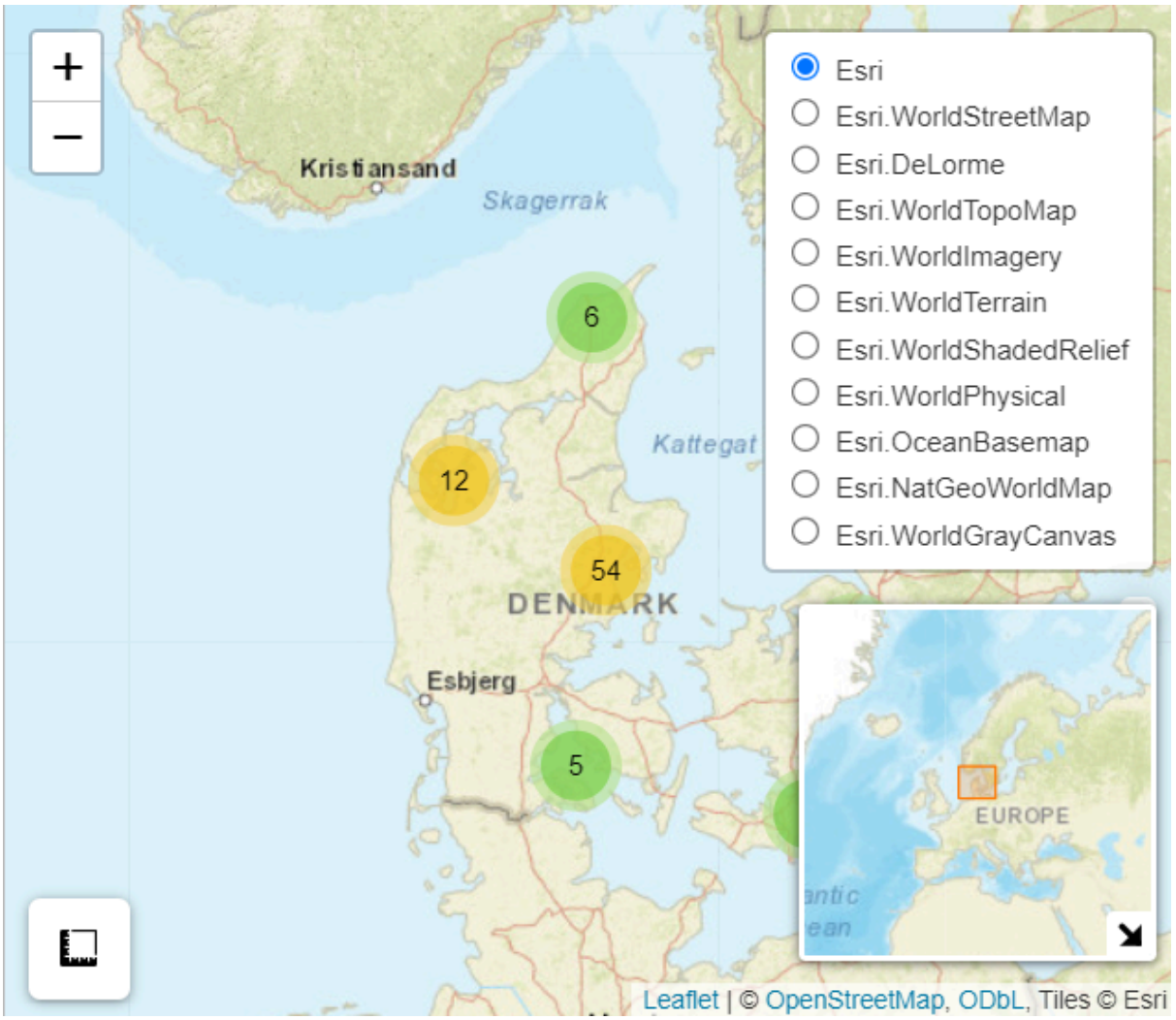
Interactive maps with R / Leaflet Maps

Toke Storgaard Guldfeldt, 21 / 03 / 2025

GitHub:

https://github.com/Digital-Methods-HASS/AU762186_Guldfeldt_Toke/tree/main/week12

visualisation, interactive map in GitHub:



answer(s):

Question 1

What is the order of longitude and latitude in the setView() function?

Answer: unless specified, the first argument will be interpreted as longitude and the second as latitude

Question 2

```
leaflet() %>%
  addTiles() %>%
  setView( lng = 2.34, lat = 48.85, zoom = 5 ) %>%  # let's use setView to navigate to our area
  addProviderTiles("Esri.WorldPhysical", group = "Physical") %>%  # add physical background
  addProviderTiles("Esri.WorldImagery", group = "Aerial") %>%  # add satellite image
  addProviderTiles("MtbMap", group = "Geo") %>%                # add geomorphic units map

addLayersControl(                                    # we are adding layers control to the maps
  baseGroups = c("Geo","Aerial", "Physical"),
  options = layersControlOptions(collapsed = F))  # replace T with F and back and run it
```

How does the map above change if you replace the T in the last line of code above with F?

Answer: if TRUE (T), layers control is automatically collapsed, of course accessible if you hover your cursor over it, good for when you have a lot of layers

Question 3

```
> glimpse(places)
Rows: 113
Columns: 8
$ f                          <chr> "Stud", "Aalborg", "Nørrebro", "Sunes Bodega", "Kristian F. Møller", "BAZAR VE…
$ Type                       <chr> "Religious place of worship", "cultural moment", "area in copenhagen", NA, NA,…
$ CoordinatesFILLTHIS        <chr> "56.1712119,10.1997894", "57.0482555,9.913457", "55.6943, 12.5488", "56.468691…
$ Latitude                   <dbl> 56.17121, 57.04826, 55.69430, 56.46869, 56.15717, 56.16304, 56.17255, 56.15689…
$ Longitude                  <dbl> 10.199789, 9.913457, 12.548800, 10.031576, 10.206187, 10.134230, 10.205510, 10…
$ Description                <chr> "cheap beer, very nice:) Men pas på evil twink i baren, han kaster øl på dig n…
$ `Greatness(1_ok_5_ecstatic)` <dbl> NA, 3, 5, 1, NA, 5, 5, 3, 4, 5, NA, NA, NA, 10, 3, NA, 5, 5, 3, 2, 4, NA, 4, N…
$ Notes                      <chr> NA, "Adela's example", NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, "please vis…
```

Are the Latitude and Longitude columns present?
Do they contain numeric decimal degrees?

Answer: yes and yes

Task 1: making a basic map of Denmark

```
l_dan <- leaflet()%>%
  setView(10.05,56.46,zoom = 6)%>%
  addTiles()

l_dan

for (provider in esri) {
  l_dan <- l_dan %>% addProviderTiles(provider, group = provider)
}
l_dan

DANmap <- l_dan %>%
  addLayersControl(baseGroups = names(esri),
                   options = layersControlOptions(collapsed = FALSE)) %>% #adds layer controls
  addMiniMap(tiles = esri[[1]], toggleDisplay = TRUE,
             position = "bottomright") %>% #adds minimap
  addMeasure(
    position = "bottomleft",
    primaryLengthUnit = "meters",
    primaryAreaUnit = "sqmeters",
    activeColor = "#3D535D",
    completedColor = "#7D4479") %>%
  htmlwidgets::onRender("
                        function(el, x) {
                        var myMap = this;
                        myMap.on('baselayerchange',
                        function (e) {
                        myMap.minimap.changeLayer(L.tileLayer.provider(e.name));
                        })
                        }") %>%
  addControl("", position = "topright")
```

This bit of code creates a basic version of a map of Denmark

Task 2: adding markers / points of interest

```
DANmap_markers <- l_dan %>%
  addLayersControl(baseGroups = names(esri),
                   options = layersControlOptions(collapsed = FALSE)) %>% #adds layer controls
  addMiniMap(tiles = esri[[1]], toggleDisplay = TRUE,
             position = "bottomright") %>% #adds minimap
  addMeasure(
    position = "bottomleft",
    primaryLengthUnit = "meters",
    primaryAreaUnit = "sqmeters",
    activeColor = "#3D535D",
    completedColor = "#7D4479") %>%
  htmlwidgets::onRender("
                        function(el, x) {
                        var myMap = this;
                        myMap.on('baselayerchange',
                        function (e) {
                        myMap.minimap.changeLayer(L.tileLayer.provider(e.name));
                        })
                        }") %>%
  addControl("", position = "topright")%>%
  addMarkers(lng = places$Longitude,
             lat = places$Latitude,
             )
```

This bit of of code adds the markers that we've made wednesday the 19th of march 2025 with the line: "addMarkers(lng=places$Longitude, lat=places$Latitude"

Task 3: clustering markers

```
DANmap_markers_cluster <- l_dan %>%
  addLayersControl(baseGroups = names(esri),
                   options = layersControlOptions(collapsed = FALSE)) %>% #adds layer controls
  addMiniMap(tiles = esri[[1]], toggleDisplay = TRUE,
             position = "bottomright") %>% #adds minimap
  addMeasure(
    position = "bottomleft",
    primaryLengthUnit = "meters",
    primaryAreaUnit = "sqmeters",
    activeColor = "#3D535D",
    completedColor = "#7D4479") %>%
  htmlwidgets::onRender("
                        function(el, x) {
                        var myMap = this;
                        myMap.on('baselayerchange',
                        function (e) {
                        myMap.minimap.changeLayer(L.tileLayer.provider(e.name));
                        })
                        }") %>%
  addControl("", position = "topright")%>%
  addMarkers(lng = places$Longitude,
             lat = places$Latitude,
             clusterOptions = markerClusterOptions())
```

Adding the argument: “clusterOptions=markerClusterOptions())” into the line: “addMarkers” clusters markers together

Task 4: to cluster or not to cluster?

Look at the two maps (with and without clustering) and consider what each is good for and what not.

Answer: if there is a lot of markers in a small area, then the unclustered map is borderline unuseable, as all the markers are too close together makes the map barely readable / navigable, if clustered however, the viewer can have a good overview of what area is marked and can then afterwards zoom into a given area to see what is interesting

Task 5: adding classification and notes to the markers

```
DANmap_markers_cluster_popup <- l_dan %>%
  addLayersControl(baseGroups = names(esri),
                   options = layersControlOptions(collapsed = T)) %>% #adds layer controls
  addMiniMap(tiles = esri[[1]], toggleDisplay = TRUE,
             position = "bottomright") %>% #adds minimap
  addMeasure(
    position = "bottomleft",
    primaryLengthUnit = "meters",
    primaryAreaUnit = "sqmeters",
    activeColor = "#3D535D",
    completedColor = "#7D4479") %>%
  htmlwidgets::onRender("
                        function(el, x) {
                        var myMap = this;
                        myMap.on('baselayerchange',
                        function (e) {
                        myMap.minimap.changeLayer(L.tileLayer.provider(e.name));
                        })
                        }") %>%
  addControl("", position = "topright")%>%
  addMarkers(lng = places$Longitude,
             lat = places$Latitude,
             popup = paste(places$Description, "<br>", places$Type),clusterOptions = markerClusterOptions())
```

Adding the argument: “popup=paste(places$Description,”
”,places$Type)” to the line: “addMarkers()” makes it so you can click on the marker and see its classification as well as any notes