PyData
Bratislava

GAP
DATA
INSTITUTE

How to create interactive maps in R & Python

(PyData Bratislava Meetup #5, Nervosa)

27. 11. 2017

Martin Bago, Instarea #PyDataBA

HOW TO CREATE INTERACTIVE MAPS IN R & PYTHON?

PyData Bratislava #5, 27.11.2017

Martin Bago, Instarea

Martin Bago

- Ing. @ Process Automation and Informatization in Industry (2016, MTF STU BA)
- Bc. @ Applied Informatics (2014, FEI STU BA)

2017-now	Data Scientist, Instarea s.r.o., Market Locator
2015-2016	Head of Analyst, News and Media Holding a.s.
2014-2015	SEO Analyst, Centrum Holdings a.s.
2011-2014	Automix.sk, Centrum Holdings a.s.
2010-2013	Editor-in-chief OKO Casopis (FEI STU BA)

- Passionate driver, beer&coffee&football lover

- Why an interactive map?
- Why to use Python or R?
- Which libraries are necessary
- What are the key issues
- How to integrate polygons of districts, how to color your map, etc…
- What kind of data do we use and what are the typical use cases we covered with interactive maps
- Q&A

Why an interactive map?

- Can show incredible amount of data
- Visual & very intuitive way to understanding data
- Available online
- No graphic design skills needed
- Easy to play with
- Porn for analysts

Why to use R or Python?

- Handling big data
- SQL select directly from code
- Saving as HTML
- Few lines of code
- No javascript needed

What do you need?

- R/R Studio
- Python – Jupyter notebook
- Basic coding skills
- Idea
- Data
- Internet

Libraries

```
install.packages(“…")  # installing packages in R
```

```
library(leaflet) # using Leaflet in R
```

```
library(readxl)   # reading excel in R
```

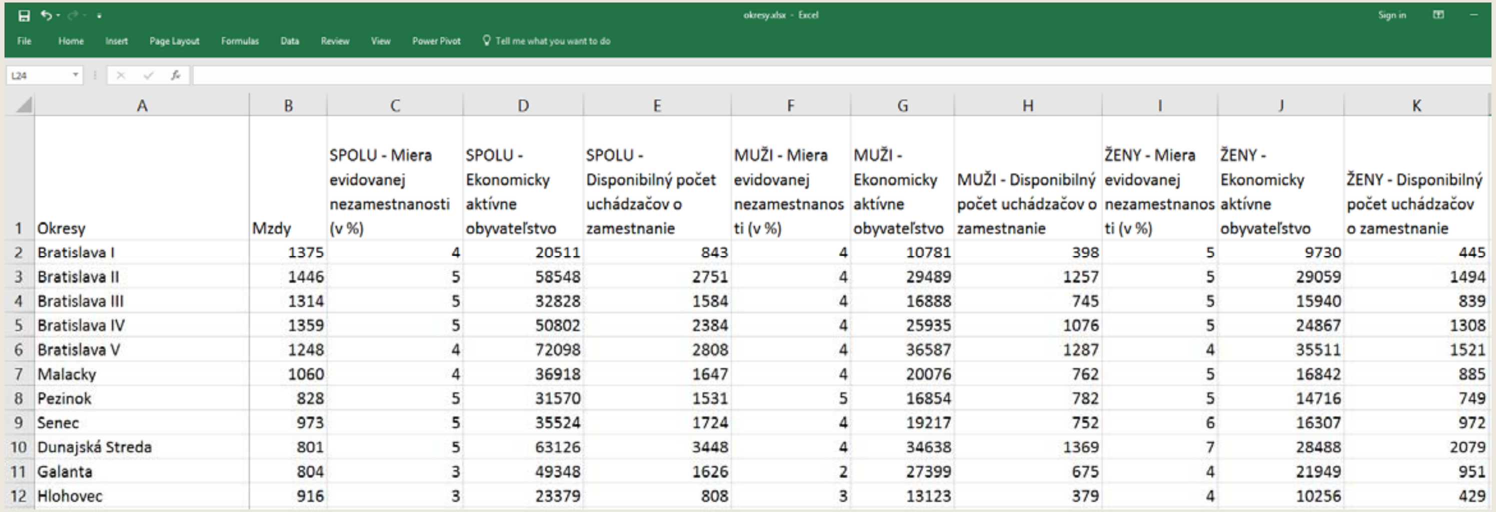
```
library(rgdal) # reading JSON in R
```

Other important libraries:
library(dplyr) - data operations
library(htmltools) - HTML popups/labels
library(geojsonio) – working with geoJSONs

Working with 2D maps (images):
library(ggplot2)
library(plotly)
library(maptools)
library(rnaturalearth)

Data formats

- .txt, .csv or .xls/.xlsx is fine, but check the encoding
- One of the benefits of using R/Python is easy use of data transformation
- You can use SQL select directly in R/Python and use data from your RAM



	A	B	C	D	E	F	G	H	I	J	K
1	Okresy	Mzdy	SPOLU - Miera evidovanej nezamestnanosti (v %)	SPOLU - Ekonomicky aktívne obyvateľstvo	SPOLU - Disponibilný počet uchádzačov o zamestnanie	MUŽI - Miera evidovanej nezamestnanosti (v %)	MUŽI - Ekonomicky aktívne obyvateľstvo	MUŽI - Disponibilný počet uchádzačov o zamestnanie	ŽENY - Miera evidovanej nezamestnanosti (v %)	ŽENY - Ekonomicky aktívne obyvateľstvo	ŽENY - Disponibilný počet uchádzačov o zamestnanie
2	Bratislava I	1375	4	20511	843	4	10781	398	5	9730	445
3	Bratislava II	1446	5	58548	2751	4	29489	1257	5	29059	1494
4	Bratislava III	1314	5	32828	1584	4	16888	745	5	15940	839
5	Bratislava IV	1359	5	50802	2384	4	25935	1076	5	24867	1308
6	Bratislava V	1248	4	72098	2808	4	36587	1287	4	35511	1521
7	Malacky	1060	4	36918	1647	4	20076	762	5	16842	885
8	Pezinok	828	5	31570	1531	5	16854	782	5	14716	749
9	Senec	973	5	35524	1724	4	19217	752	6	16307	972
10	Dunajská Streda	801	5	63126	3448	4	34638	1369	7	28488	2079
11	Galanta	804	3	49348	1626	2	27399	675	4	21949	951
12	Hlohovec	916	3	23379	808	3	13123	379	4	10256	429

Source: http://www.statistics.sk/pls/elisw/metainfo.explorer?obj=41&cmd=go&s=1002&sso=2&so=15

Reading data&polygons

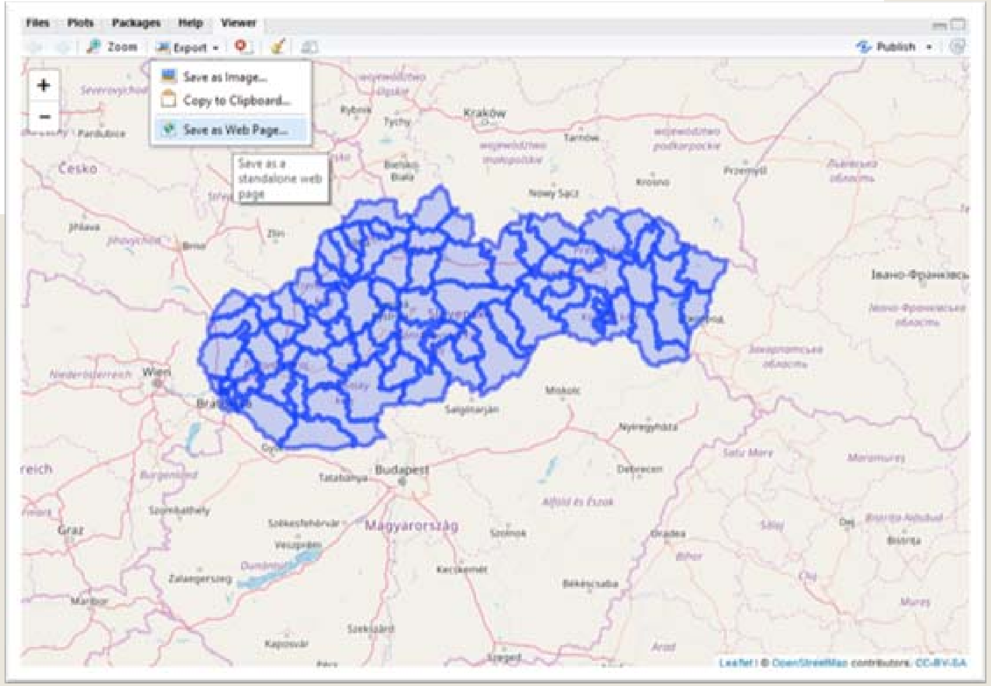
```
SVK_districts <- rgdal::readOGR("districts_epsg_4326.geojson", "OGRGeoJSON")
okresy <- read_excel(“okresy.xlsx")
```

All used files will be available to download

Districts of Slovakia in geoJSON at GITHUB (thx to drakh)

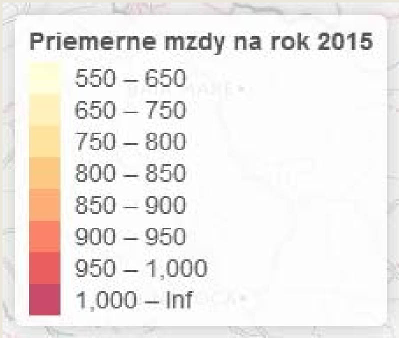
Generating map

```
leaflet() %>%
   addTiles()
   addPolygons(data = SVK_districts)
```



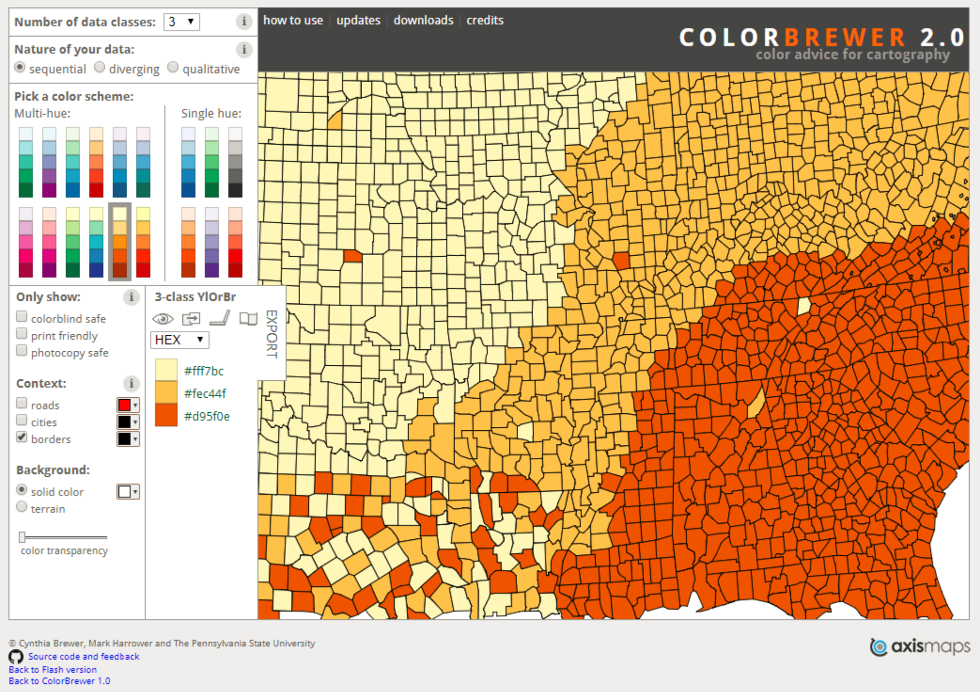
Setting bins & colors

```
bins <- c(550, 650, 750, 800, 850, 900, 950, 1000, Inf)
pal <- colorBin("YlOrRd", domain = okresy$Mzdy, bins = bins)  #ColorBrewer
```



Priemerne mzdy na rok 2015

■	550 – 650
■	650 – 750
■	750 – 800
■	800 – 850
■	850 – 900
■	900 – 950
■	950 – 1,000
■	1,000 – Inf



Number of data classes: 3

Nature of your data: ● sequential ○ diverging ○ qualitative

Pick a color scheme:

Only show: □ colorblind safe □ print friendly □ photocopy safe

Context: □ roads □ cities ☑ borders

Background: ● solid color ○ terrain

3-class YlOrBr

HEX

■	#fff7bc
■	#fec44f
■	#d95f0e

© Cynthia Brewer, Mark Harrower and The Pennsylvania State University

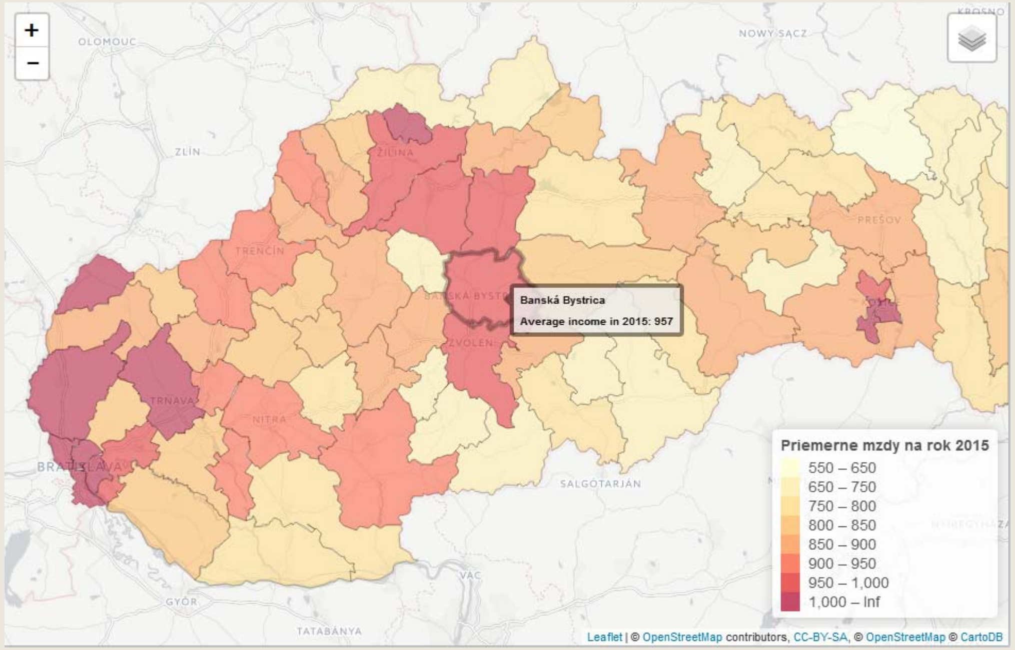
Source code and feedback

Back to Flash version

Back to ColorBrewer 1.0

Adding labels

```
library(htmltools)

labels <- sprintf(
  "<strong>%s</strong><br/>Average income in 2015: %g",
  okresy$Okresy,okresy$Mzdy) %>% lapply(htmltools::HTML)
```



Generating map II.

```
leaflet() %>%
  addTiles() %>%
  setView(lng = 19.411622, lat = 48.696708, zoom = 7) %>%
  addPolygons(data = SVK_districts, opacity = 0.25, weight = 1, color =
        "black", fillColor = ~pal(okresy$Mzdy), fillOpacity =
        0.50,smoothFactor = 1, stroke = 0.1, label = labels, highlightOptions
        = highlightOptions(color = "black", weight = 5, bringToFront = TRUE))
        %>%
  addProviderTiles('CartoDB.Positron') %>%
  addProviderTiles(providers$Stamen.Toner, group = "Toner") %>%
  addProviderTiles(providers$Stamen.TonerLite, group = "Toner Lite") %>%
  addLayersControl( baseGroups = c("OSM (default)", "Toner", "Toner Lite"),
        options = layersControlOptions(collapsed = FALSE)) %>%
  addLegend(pal = pal, values = okresy$Mzdy, opacity = 0.7, title =
        "Priemerne mzdy na rok 2015",
         position = "bottomright")
```

AddPolylines & Add … & Add..

- https://rstudio.github.io/leaflet/

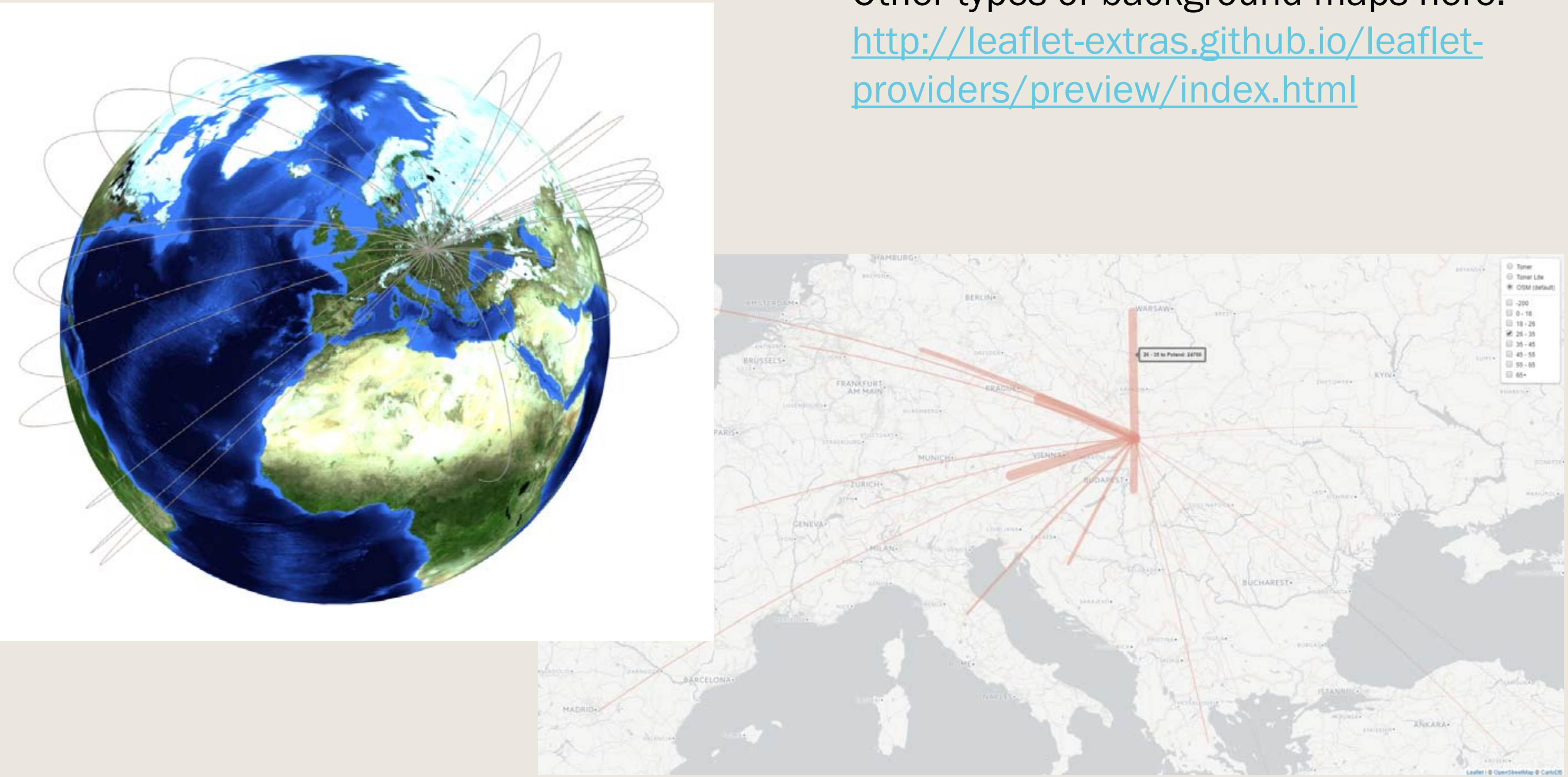
```
addMarkers(~long, ~lat, icon = …)
addMarkers(clusterOptions = marketClusterOptions())
addCircleMarkers()
addCircles(lng = ~Long, lat = ~lat, weight = 1, radius = ~sqrt(Pop) * 30)
addRectangles()
addPolylines()
```

Complete code

```
library(readxl) # reading excel
library(leaflet) # working with leaflet
library(rgdal) # working with json
library(htmltools) # adding labels / popups

setwd("C/…")  # set working directory

SVK_districts <- rgdal::readOGR("districts_epsg_4326.geojson", "OGRGeoJSON")  # reading JSON polygons
okresy <- read_excel(“okresy.xlsx")   # reading offline data

bins <- c(550, 650, 750, 800, 850, 900, 950, 1000, Inf)    # setting bins
pal <- colorBin("YlOrRd", domain = okresy$Mzdy, bins = bins)  # setting colors

labels <- sprintf(
  "<strong>%s</strong><br/>Average income in 2015: %g",
  okresy$Okresy,okresy$Mzdy) %>% lapply(htmltools::HTML)   # adding labels

leaflet() %>%             # map generating
  addTiles() %>%
  setView(lng = 19.411622, lat = 48.696708, zoom = 7) %>%        # set default view
  addPolygons(data = SVK_districts, opacity = 0.25, weight = 1, color = "black", # adding polygon tiles
              fillColor = ~pal(okresy$Mzdy), fillOpacity = 0.50,smoothFactor = 1,
              stroke = 0.1, label = labels,
              highlightOptions = highlightOptions(color = "black", weight = 5, bringToFront = TRUE)) %>%
  addProviderTiles('CartoDB.Positron') %>%       # adding other backgroung maps
  addProviderTiles(providers$Stamen.Toner, group = "Toner") %>%
  addProviderTiles(providers$Stamen.TonerLite, group = "Toner Lite") %>%
  addLayersControl( baseGroups = c("OSM (default)", "Toner", "Toner Lite),
                    options = layersControlOptions(collapsed = TRUE)) %>%
  addLegend(pal = pal, values = okresy$Mzdy, opacity = 0.7, title = “Average income in 2015",  # adding Legend
            position = "bottomright")
```

Alternatives

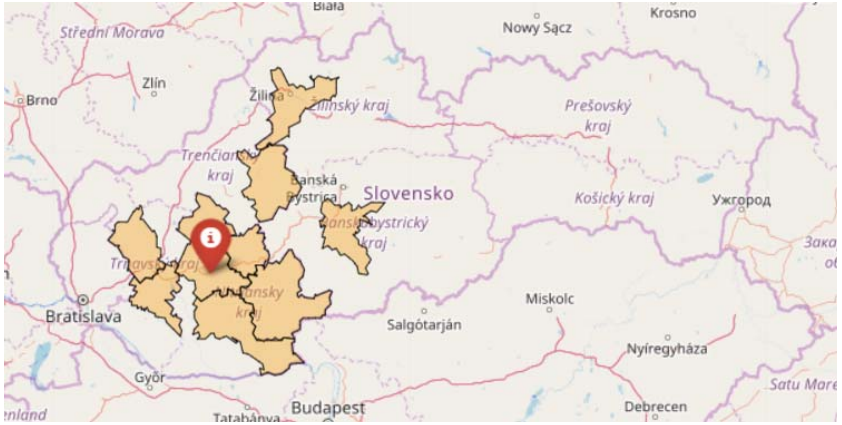
Other types of background maps here: http://leaflet-extras.github.io/leaflet-providers/preview/index.html



…and how does it look in Python?

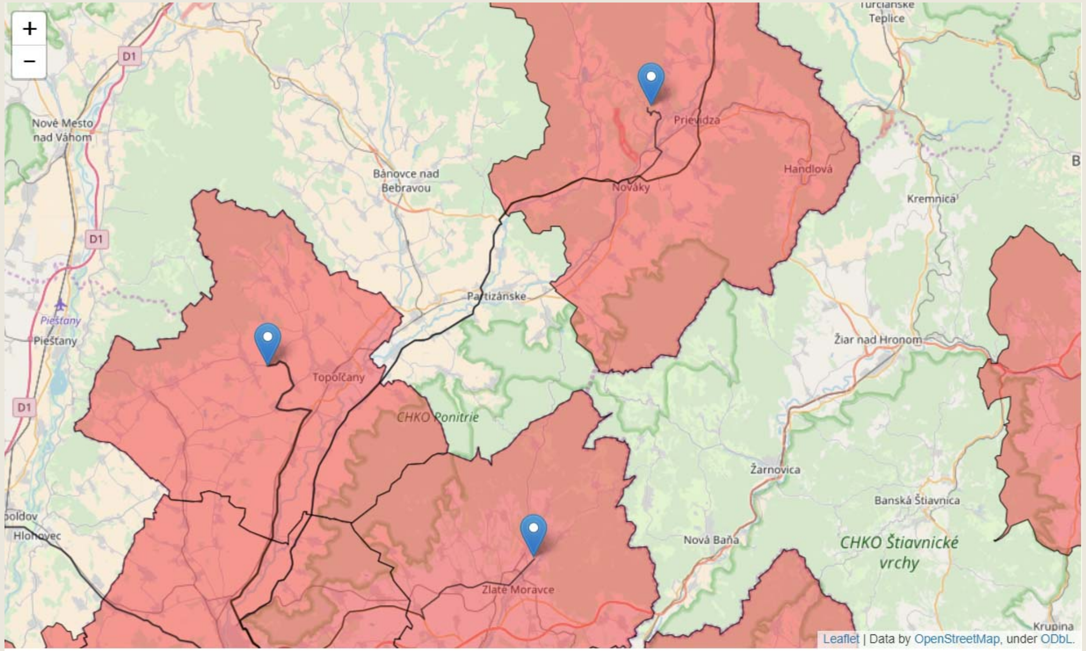
- Folium - Python Data, Leaflet.js Maps
- https://github.com/python-visualization/folium

```
import folium
import json
import pandas as pd
import numpy as np
import colorbrewer


.
.
.
with open('districts2.geojson') as f:
    js = json.load(f)

m = folium.Map(location=[48.5, 17.5], zoom_start=7)

folium.Marker(
        [lat, long]
        , popup= ‘TEXT'
        , icon=folium.Icon(color='red',icon='info-sign')
    ).add_to(m)

m

m.save('mapa_final.html')
```



Other options

- Getting routes from Google Maps (library googlemaps)
- Adding HTML popups with images (library htmlwidgets)
- Distance measurements



Find more at:

analyzy.marketlocator.sk

THANKS FOR YOUR ATTENTION

FEEL FREE TO ASK

martin.bago@gmail.com

martin.bago@instarea.com

https://www.linkedin.com/in/martinbago/

https://www.facebook.com/martin.bago