

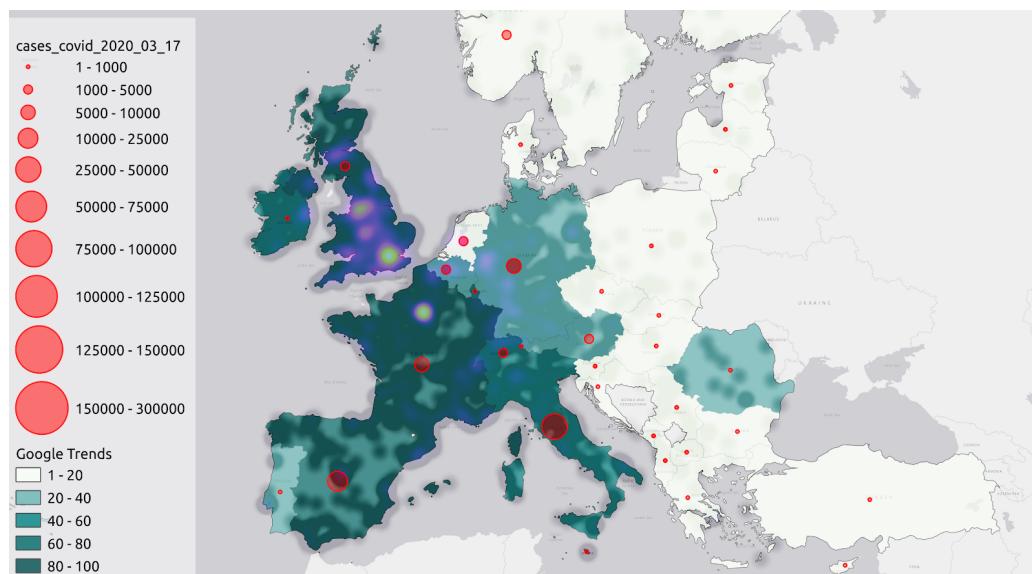
Data viz - Milestone 2

The goal of our visualization is to link the spread of the COVID-19 pandemic with its information flow on social network and web search.

You can see what our current website looks like [here](#).

In order to visualize the three main information of our project, namely the number of google searches and tweets related to Covid-19 and the reported cases of Covid-19 in the same map and without “overloading” the visualization, we adopted three different visualization that will be overlaid in our dashboard.

An example of the visual rendering of the three visualizations was implemented for a specific day (17/03/20) using the QGIS software (see below). We may change the colors for our final product, especially for the heatmap. Furthermore, the legend corresponding to the heatmap was not added here but we will in our dashboard.



We break down this goal in different independent pieces to implement :

Static Europe map website

Variable: Geojson to acquire geographic data and construct the map (source: [NUTS 2016](#), [Eurostat](#)).

Implementation: D3.js, JQuery, Bootstrap (to have a nice base website quickly)

Related lectures: To set up the map, we will first need the courses of week 4 - D3 Basics and week 8 - Maps

Choropleth map of Google searches related to Covid-19 by country and regions

Variable: Google Trends score [0-100] by country and region that we geocoded to a geojson representing administrative borders of European countries (source: [NUTS 2016, Eurostat](#)).

Implementation: D3.js, JQuery

Related lectures: Lecture on Maps (slide 19) and D3 Basics (week 4)

Extra ideas: Maybe try a cartogram representation of the Google Trends index. For now we are using GeoJSON data for the world map so we would have to change to TopoJSON data and try not to mess the entire visualization up.

Bubble map of Covid-19 reported cases

Variable: Covid-19 reported cases (source: John Hopkins) geocoded at the countries' centroid. We might add the number of cases at the center of the bubble for completeness similarly to [this map](#).

Implementation: D3.js, JQuery and this excellent [template](#) from Yan Holtz.

Related lectures: Lecture on Maps (slide 19) and D3 Basics (week 4)

Extra ideas: Transform the bubble in a pie chart with the number of confirmed cases and the number of deceased. But the risk is that we cannot distinguish countries with a higher death ratio because numbers are usually around 10%.

Geographical heatmap of geolocalized tweets related to Covid-19

Variable: Number of tweets per day geocoded at the exact location (Lat/Lon coordinates). We will use a geographical heatmap to visualize the density of tweets across Europe by day. Basically, the heat map represents the intensity of points (with dark and high colors representing low and high intensity respectively). It is done using a Kernel Density Estimation interpolation, where we first specify the radius on which we want to interpolate values around each point (in our case, 100km) and the additional weights we want to give to each point (in our case, the number of tweets).

Implementation: D3.js, JQuery, [heatmap.js](#)

Related lectures: Lecture on Perception color (slide 20) and Mark and Channels (slide 19) and Maps

Space-time visualization

Goal: Since we are working with time data, we have set up a slider to change the dates within this time period: 22-01-2020 / 17-03-2020.

Related tools: D3.js, JQuery, [ion.RangeSlider.js](#)

Extra ideas: Expand the time period of the study.

Our Minimal Viable Product is what is described in our feature except the “Extra ideas” part, those are the core features. The “Extra ideas” part is what we will implement if we have time.

Extra features

- **Aggregation:** Because we are dealing with a lot of precise geolocation data our visualization is currently computationally heavy and takes time to load. We could aggregate some of it to reduce the loading time because we don't need a high degree of precision. To achieve this we could make use of the Lecture on Interaction (week 5)
- **Zoom on countries:** We also want to allow the user to zoom in on each country to get a more accurate view and also to get more insights about that country through a small toolbox with graphics and other information.
- **Basic plot on the right panel:** For now the right panel is not working, it always show the same graph. We need to find meaningful information to display there per country such as the detailed number of infected and deceased, the daily confirmed case increase, etc...
- **Basic world informations:** Currently there is a blank left panel that we might use to display world infection data in order to better identify the context especially in relation to the Chinese situation.
- **Correlation plot of several covariates by country:** The central map will allow the user to explore visually the link between Covid-19 cases, tweets and google searches. However, in order to quantify this correlation globally (i.e. for the European countries) and for the entire time period, we could add a plot quantifying the correlation between our three main variables and other covariates specific to the country that could influence the spatial distribution of Covid-19 cases (e.g. total population, GDP) and social network activity (e.g. proportion of young adults, total population). The resulting correlation plot was first implemented in R (see left figure) but we will translate it in d3.js to integrate the results to our final product. As an extra-extra idea we could add some variables and change the correlation plot design by implementing a solar correlation plot with the number of death from Covid-19 at the centre, Figure on the right.

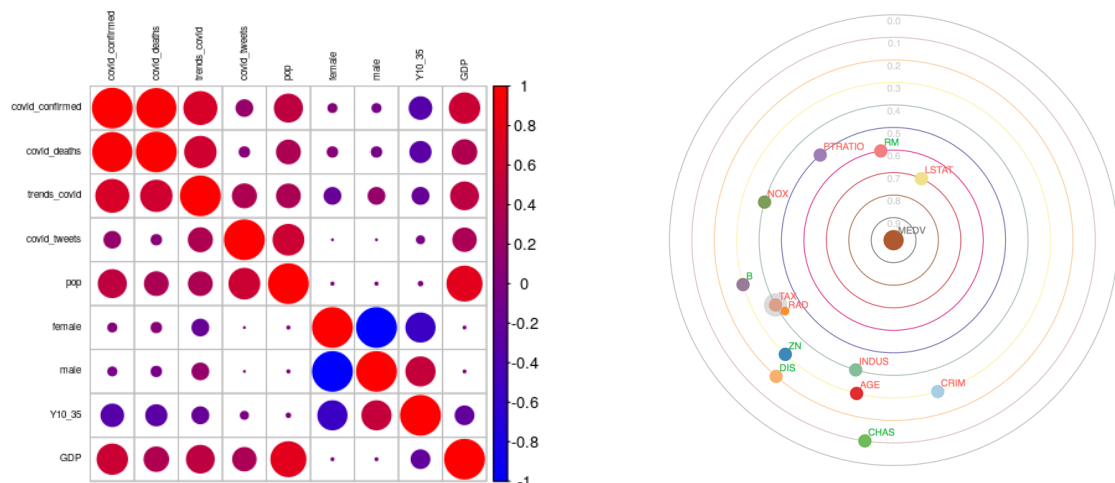


Figure 2. Credit: Stefan Zapf and Christopher Kraushaar