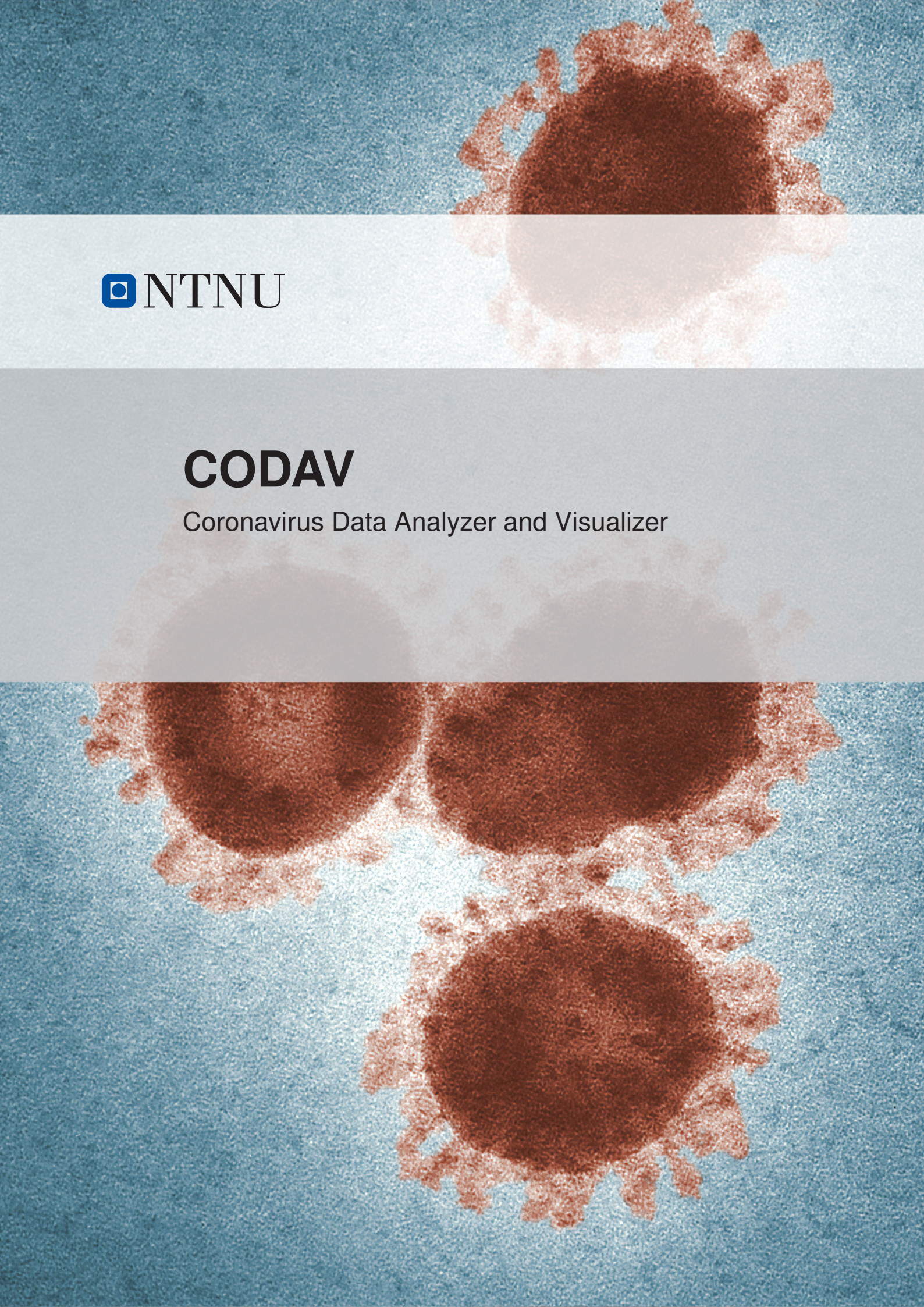


NTNU

CODAV

Coronavirus Data Analyzer and Visualizer

Contents

1

Introduction

This project's purpose is to analyze and visualize data related to the COVID-19 pandemic. In particular, the reference public consists of common people interested in having a better understanding of the pandemic evolution and future forecasts. Those people should already have some information about the pandemic evolution and should know the meaning of basic parameters, such as cases, tests and positive rate.

Users do not have a complete freedom in the selection of attributes, which means that they can select among some preset interactive graphs, on which they can also perform filtering operations, but they do not have the possibility, for example, to select custom columns. Since they are not supposed to be experts, indeed, the selection of a large number of parameters may have been confusing and the interface more difficult to navigate. However, this tool has been developed using a modular structure, which allows programmers to easily create new graphs using the existing data, and to make them available as preset graphs. This tool, therefore, is also thought for further improvements performed by experts in the field, which are supposed to be able to properly select which attributes to include in the preset graphs in order to provide meaningful representations.

The tool has been designed with the purpose of being intuitive to use with easily understandable results. This means, for example, that if different countries are displayed, those are represented using different colours, and if several attributes are associated to the same country, the colours of the attributes will be based on the one associated to the country, but they will slightly change for the different parameters. This process is performed automatically, and the programmer does not have to manually specify such shades nor, for instance, the position of subplots inside a bigger plot.

Another important part of the project consists of the predictions. I propose the use of some models, and compare them on the basis of the available data to date. Since I consider long term predictions, it is quite challenging to have a high accuracy, because there are also variables which have not been considered so far (for instance, the presence of a vaccine in the near future), and that will be analyzed more in depth in the present report.

2

Interface overview

Before going into more details in the application of theoretical principles, this section introduces the interface of CODAV and explains how to perform common operations. Figure 2.1 shows the top part of the tool interface, since the bottom part contains only the plots, which will be further analyzed subsequently.

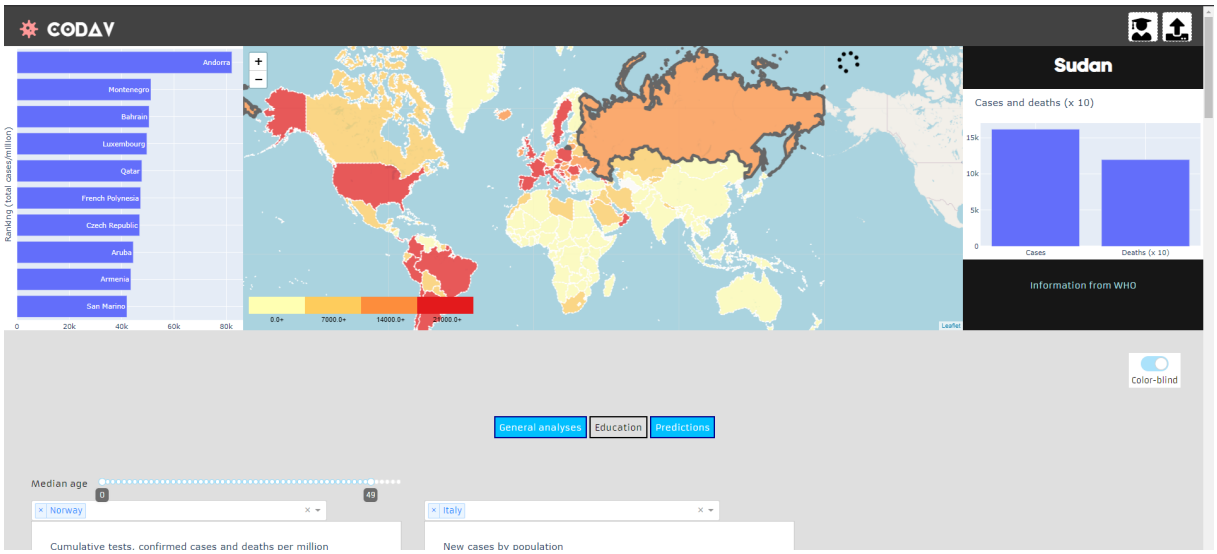


Figure 2.1: Top part of the interface

2.1 File uploading

The first step in the use of the tool consists of uploading the necessary files. In order to do so, the user can find two buttons in the top right corner of the interface. The button with the standard upload symbol (on the right) must be used to upload the COVID cases file from the dataset proposed in the assignment specifications, while the other button must be used to upload the UNESCO education dataset[1]. Once the data will be correctly uploaded, waiting symbols will appear where the elements of the interface will appear once computed.

In order to upload the files properly, the user can either drag and drop them on the buttons (one at the time) or click on the button and choose the file. When the file is on the chosen button, a green square will appear around the button.

When the interface will be completely loaded, the user will still be able to upload new files, which will replace the current files. If the user uploads only one new file, this will replace the existing file and the data will be re-computed consequently. There are no restrictions on the file format and name, but if the file is not compatible with the tool, the computations will stop. In that case, the user will still be able to upload a new file, consistent with the specifications.

The top bar will remain visible when the user scrolls down, since it does not take a consistent part of the interface and it is useful in case the user wants to upload new data.

[1] https://en.unesco.org/covid19/educationresponse

2.2 Top area

Below the top bar, the user can find an area which extends from left to right, covering the entire extent in terms of page width. This area contains information about the total amount of COVID cases to date. The central part consists of a Leaflet map, which represents the total number of COVID cases per million inhabitants. When the user hovers on a country (s)he can see the number of total cases per million in the top-right corner of the map. In this example, I show that this information is loading, since it may take some time depending on which other computations are performed at that time. The user can also perform other typical operations on the map, such as zooming and panning. When the user clicks on a country, the right part of the top area will show a plot comparing the number of cases and the number of deaths for that country and a link to reach the WHO website for that country. In this case, the user has previously clicked on Sudan and now is hovering on Russia. The scale of the plot is different for each country, since the purpose is to compare the cases and the deaths, while the comparison between different countries in terms of cases is visible from the map colours.

On the left, the user can see the top 10 countries per number of cases per million inhabitants updated to the date of the uploaded dataset. In this case, for example, it is possible to see that many small countries, such as Andorra, Luxembourg and San Marino are in the list. A possible reason is that some people have to move to contiguous countries to work and a few cases impact significantly on the overall number of cases per million. Moreover, when a few people get ill, they can spread the virus in the rest of the country more easily. Other small countries, such as the Vatican City, do not appear on the list, therefore I suppose that being a microstate is not necessarily a factor that impacts on the total number of cases per million.

2.3 Graph area

On the right side of the interface, the user can see a button to activate or de-activate the "color-blind" mode. If the switch is on (colored in blue), the colors will be color-blind friendly for all the plots, otherwise some plots may not be easily readable by people with color-blindness problems. The reason why I have not put the color-blind mode as the default option is that in that case the colorscale is reduced, and the users would be able to plot less differently coloured lines at the same time.

Below, the user can finds three buttons, which can be used to show (highlighted button) or hide specific groups of plots. The user can decide to activate multiple areas at the same time. In the example, the user has activated the General analyses and Predictions areas, while the Education area is hidden (even though the plots already exist).

Then, the remaining part of the interface is formed by plots, which are resizable, and whose content will be analyzed in further detail later in this report. The plots also allow interactions, but I will not discuss this point specifically since those interactions are the ones implemented in Plotly.

3

Data handling

Data pre-processing is an important phase to be completed before analyzing and visualizing them. In this case, I have used both the COVID dataset from OurWorldInData, suggested in the assignment specifications, and the UNESCO dataset[1] which studies the impact of the pandemic on education, and in particular to the opening of schools and universities, considered country by country and day by day from the beginning of the pandemic.

Both the datasets are not complete and contain several missing values. As an example, there are some missing data related to the number of COVID cases for the initial phases of the pandemic, when the data were collected mainly in China. Moreover, some of the COVID dataset contain comments, which can be useful for human readers, but that are not easily understandable by a machine, since the text is not presented in a standardized format.

An additional, but related, challenge in the COVID dataset consists of the fact that not all the countries record all the types of data. For example, the ICU (Intensive Care Units) availability is recorded mainly by some European countries, and the consequence is that the visualization has to focus on a specific area of the map in order to be meaningful. Moreover, in same cases the units of the data slightly change from country to country. This is the case of test units. However, even when testing is unclear, I am not giving much importance to the comparisons in terms of testing, therefore I assume that even when the unit is unclear I consider it as the number of tests performed. In particular, when I visualize this information, I show it as "new tests". The reason why I have made this choice is that "test units" may contain any kind of text, and it would be necessary to consider each case separately to have a more generic tool.

It is also important to consider the data distribution. The tool displays the top 10 countries by number of cases per million of people on the left of the interface, and it is very clear from there that some countries have very high values. It is also possible to observe that many of those states are micro-nations in relatively densely populated areas (for example, Andorra and San Marino). The effect of this phenomenon is that if the numbers are displayed on a map using a linear scale for the colours, the majority of the countries will have a similar colour, and those micro-nations will be the only ones with different colours.

3.1 Types of data

In the considered datasets, most of the columns represent quantities, which can be measured and compared, and on which it may be possible to perform operations. I will now analyze the two datasets separately to understand the type of the different attributes.

The education dataset contains the dates and the ISO codes of the countries, which I have considered as the index of the dataset, since both of them are not unique, but if I take them together, they become unique. The name of a country in this case in not used anywhere, since when I consider the overall dataset I use the name from the overall dataset. The country, however, could be considered as part of the index as well, since the names are actually unique. In fact, this information is redundant if the ISO code is already considered. Dates can be considered as quantitative, since I perform operations on them when I make predictions later, while countries can be considered as categorical, since I cannot perform any arithmetic operation on them and they don't represent any quantity by themselves. In fact, dates could also be considered as ordinal if no operations were performed on them, but in this case operations are quite important.

[1] https://en.unesco.org/covid19/educationresponse

The status, which refers to the opening of schools and universities, is instead categorical, and in particular there exist four categories: fully open, partially open, closed due to COVID-19, Academic break.

As for the main dataset, here there are more attributes that can be considered. In particular, they can be classified as follows:

- Categorical: ISO code, continent, location, other textual columns
- Quantitative: the remaining non-textual columns

where the textual columns in general contain notes and comments about some attribute.

3.2 Data formatting

The data from the two datasets have been given using the CSV format, which can be easily read using Pandas Dataframes. In this particular case, the application has to use the same (pre-processed) data for all the visualizations. This has been challenging because of the structure I propose for this application. In particular, my purpose is to create a modular and easily extensible application, where the creation of plots with different kinds of filters and/or animations is particularly easy for the programmer. For this reason, I have some generic functions to build generic maps or plots, which take the current filter values and view of the dataframe as an input, and plot the graph as an output. In this way, in case the programmer wants to perform any change to the way in which graphs are visualized and managed, (s)he can do so by changing only a limited number of functions, and not all the possible graphs by hand. Moreover, some small changes to the code would allow the user to create custom plots on the basis of the features and the filters (s)he wants to use in the visualization. However, this has not been implemented since such a level of freedom would have created additional confusion for a non-expert user, which is the target audience of this application.

Since there is a generic function for the creation of the graphs, it is impossible to define Dash callbacks in the same way as if there were multiple separate plots, with hard-coded IDs. For this reason, it is necessary to make use of the concept of matching patterns. The IDs of the different graphs and filter, therefore, are organized in a rational way, with a fixed structure.

This introduction can also be useful to understand my decisions in terms of data management in terms of memory. Because of the described architecture, the imported data have to be shared across different graphs, and it is not possible to use just a global variable containing the dataset in the Pandas format. For this reason, after the dataset is imported, it is saved in a hidden div, which is one of the methods suggested by Dash documentation. Then, the data can be easily accessed from the hidden div, and it is also possible to use callbacks on the div itself, to detect any change in its content.

Having two datasets and the necessity of storing the content inside a div, I have decided to merge them before saving their content. For this reason, I have considered the ISO code and the date columns in the two datasets and used them as the index for merging, since their combination will always be unique and the ISO code is an international standard, shared across the two datasets.

This approach allows to use also updated versions of the datasets, as long as the dates are in the same format as now and the date and ISO code column names remain the same. However, if the other columns change, and in particular if some column will be removed or renamed with respect to the current situation, some plots will not be visualized properly because they will not find the necessary data.

3.2.1 Use of different formats to represent the data

The use of the CSV format for this project has presented some challenges with respect to other ways of organizing data, such as databases. In particular, in the context of this project, CSV is less standardized and does not allow to represent complex data structures, which can instead be easily modeled using traditional database structures. CSV, being a tabular format, has the problem of redundancy, and the same column may contain several copies of the same information, while databases allow the definition of a more complex structure, based on relations between tables and attributes. However, in the context of data analysis, Pandas can be used to easily read CSV data and to perform queries on the data.

There also exist some libraries which allow to write queries in a compact way, similarly to what can be observed in the case of SQL. In case the users were allowed to create custom graphs, this language could have been used to type custom interrogations, possibly combined with graphical filters. In this project,

however, I have chosen to avoid this degree of freedom to avoid making the interface more complex than necessary and, for this reason, counter-intuitive for a non-expert user.

When the data are placed inside the hidden div, as described above, they are transformed from Pandas to JSON, which has a syntax similar to the one of JavaScript, but different with respect to CSV. Such transformation happens automatically, and there is no manual parsing on the programming side. This solves one of the problems related to JSON, which is the fact that it is more difficult to be parsed, because its structure is more complex than the tabular one.

3.3 Signal sampling

This project has not involved any advanced signal sampling technique, since the time resolution was the same for both the dataset (one sample per day), and the original COVID dataset already contained some additional information which referred to the weekly number of cases, in addition to the daily cases.

Some parts of the project have required techniques related to signal sampling, and in particular the detection of peaks for the subdivision of the prediction data between the train and the test set. However, this has only required the use of an existing scipy module, without the need of reimplementing the peak-picking algorithm.

4

Visualization principles

In this project, I have taken several data types into account. In particular, the proposed dataset contains mainly temporal data, and for this reason a natural approach to the task consists of showing data that change through time. This is why most of the plots are line plots. Moreover, the proposed data are also related to locations, and for this reason (possibly animated) maps are also a good choice to represent them. In particular, animations can be used to give the user a visual feedback about the change in the number of cases through time across different countries. Maps can also be interactive, as shown in the map placed at the top of the interface.

In the case of the education dataset, I have assigned some arbitrary values to the opening status in order to visualize the school opening status trend through time. In particular, a full opening has a value of 1, a partial opening has a value of 1.5 and a full closing due to COVID-19 has a value of 2. Academic breaks are left as NaNs, and in this way the corresponding data are not displayed in the plots, since they are not particularly relevant in my opinion. I have decided to start from 1, and not from 0, because I use those data in combination with other data from the COVID dataset, and if I use the education status as the denominator of an index, I don't want it to go to infinity, since in that case the representation is not particularly intuitive for users, which would see some spikes, while the other values would be too small to be visualized.

4.1 Colour

An important part of the visualization consists of the choice of colours in the plots. I have tried to keep consistent choices through the entire application, and to have the same colours to represent the same patterns. For example, I have considered high values for the data using the red colour when I consider maps. The reason why I have made this choice is that high data, in the context of a pandemic, are considered as bad data, and red is traditionally associated to something negative and dangerous. For example, higher numbers for the number of cases are represented with darker shades of red.

As for the line plots, it is very important to have different colours for each trace. For example, if I consider multiple countries on the same plot, I represent them using different colours. Those colours should be different the ones from the others, because if they are similar or equal the user will be confused. For this reason, I have chosen the default colours proposed by Plotly. Those colours, in fact, are different only for the first ten traces (for example, ten different countries), and then the same colours start repeating. This is not a problem, in my opinion, since visualizing more than ten traces is confusing for a user anyway, independently on the chosen colours. Moreover, the user can choose the color-blind mode and reduce the scale to only three colors.

In some cases I visualize multiple traces for the same value of a filter (for example, multiple traces for the same country). In this case, I keep a base colour for that specific country, in such a way that it is recognizable, and I have implemented some small variations in terms of the colours of different quantities. For instance, if I consider the number of deaths, the number of cases and the number of tests for a specific country, I can use the default blue used by Plotly for one of the traces, and two different variations for the other two traces. There is no risk that there are overlaps between other countries' colours and those variations, because the default Plotly colours are very well separated (so they are visually very different). The same applies for the color-blind mode.

It is indeed also important to consider the problem of colour blindness and other illnesses. This problem is always taken into account for the maps where I use different shades of colour, such as in the case of the map placed on the top. On the other hand, this does not apply to every possible plot by default, but the color-blind mode can be activated at any moment for the currently visible plots. Moreover,

the prediction plots are color-blind safe because the prediction line is dashed and the actual data line is continuous.

4.2 Types of representations

In this project, I have focused mainly on some kinds of data representation. I have used maps to display the current situation of the pandemic across the world, and I have decided that choropleth maps are the best option to easily compare different quantities. I have used two different libraries for the maps: the top map uses Leaflet, while the ones in the graph area are the native choropleth maps in Plotly. The reason why I have taken this decision is that in smaller areas the Leaflet map is not clearly visible, since the country borders are thicker. Moreover, the Leaflet map is more detailed, and I think it is a better choice to display it as the main map.

The second type of plot I have used is the line plot. This is particularly appropriate for the representation of numerical values which change through time. In some particular cases, I have also used scatter plots. Those have been useful to visualize two channels in a bidimensional plot, by using points as markers. For example, it is possible to use such a plot to represent the population density and the number of cases per million. This scenario will be further analyzed in the use cases section.

From a visualization point of view, I have styled the scatter plots using points with the same colour, since in the examples I have considered they all belong to the same category, and I have placed labels on top of the points. I could have used other kinds of markers, such as flags for the countries, but this would have made the graph harder to read, in my opinion, since it is not immediate for the user to associate a flag to a given country (depending on where (s)he lives, (s)he may be more familiar with some regional flags, but not with other countries' flags) and they would take also more vertical space with respect to text. I believe that there exist no optimal solutions in this case, since if there are many countries, the texts will also overlap and the user will have to zoom on a specific area to read it properly. On the other hand, using arrows which point to the text would not be as easy to read as well, since the user should follow the arrow from a given point, and when there are many points this operation is difficult as well.

I have not used bubble charts for the representation of tabular data because in the cases I have analyzed bubbles would have taken more space and would have made the plot difficult to read. In particular, the challenge with COVID figures is that the involved sizes range across very wide intervals (for example, when considering the total number of cases, which ranges from 0 to hundreds of thousands). In this way, some bubbles should be almost invisible, while others would become very big, covering other bubbles. On the other hand, I think it is important to avoid the use of logarithms for the size, at least in this specific problem. Using logarithms may induce the user to think that two countries with a very different number of cases are in fact very similar, and draw wrong conclusions as a consequence. This may also be true for the line plots, but in that case the use of a logarithmic scale is justified because there is an exponential increase in the number of cases.

Finally, I have implemented some bar plots, which can be used, for example, to show rankings (for instance, the rankings of the countries with the highest number of cases per million) or to show cumulative values (for example, the days in which the schools and the universities have remained closed in different countries). Bar plots can also be used to display data belonging to different categories. For example, I think they can be very effective in the case of the representation of several measures associated to different countries, such as the school opening status (for instance, the number of days a school remains open, partially open or closed due to COVID-19 for different countries). I will analyze those example in more details as well in the use cases section.

For this project, I have not used heatmaps. The reason why I have taken this decision is that the available data are country-wide, therefore the use of different colours for different geographical areas is already performed when the choropleth maps are used. More detailed data would make the heatmap really useful, for example to display the number of cases for specific cities. In that way, it would be easier to compare the population density values with the total number of cases per million, as I have shown in a previous example for different countries.

From a programming point of view, it is possible to model a bar plot in the exact same way as a scatter plot, and whoever wishes to add new analyses could easily change the code in the main Python script. The reason why this possibility is not given to the user is that it may confuse him or her by giving too many possibilities at the same time. The strategy I have adopted, as already mentioned, consists in giving the user the possibility of choosing among a set of pre-defined plots, maps and predictions, on which (s)he can possibly apply filters and interact.

4.3 Details about spatial data

Maps are one of the most relevant representations in this project, since it is important not only to have an idea about the current situation in a country, but also about the distribution of the new cases and of the total cases. For this reason, it is possible to show the data using different colours, by means of choropleth maps, and also through animations, which are able give the user an idea about the time evolution of the cases.

On the other hand, other map-based visualizations are not particularly relevant in my opinion, since only country-wide data are available with this dataset. With more detailed data, it could be interesting to show the differences across countries, to see whether the number of cases per million people is higher or lower in urban areas, for example.

Volumetric visualizations in this case are not relevant as well, since they are usually used to represent tridimensional objects, for example in the medicine field. In this case, it could be possible to have a tridimensional map, for example using histograms, but the visualization would be much more complex from the user perspective, and it would be difficult to understand the meaning of many possibly overlapping bars. For this reason, I have avoided this type of visualization.

4.4 Other visualization choices

In this project, I have decided to avoid using tridimensional plots in general, since I think they may be confusing for an user and because I could not find any meaningful application for the visualization of COVID data. As I have already mentioned, I have focused mainly on line plots and maps.

Plotly natively allows further interactions with the graphs, and this is one of the reasons why I have chosen it. In particular, it allows the user to pan, zoom and select areas and intervals on the plots. Moreover, a button (the house) allows to restore the initial view of the plots. This is very important since the user may have performed several operations before getting to a certain area. Other relevant features which are useful in this context are the possibility to show and hide traces from the legend visualized on the right of the plot and the presence of interacting both with buttons and with commonly used gestures.

Responsiveness is also something that I have considered. My strategy is that when I start the application I load the necessary data, which are uploaded by the user, and compute the preset graphs and predictions. In this case, it is very important to consider that this initial phase may require some time, since many calculations are performed. However, I believe that the responsiveness should apply mainly to the subsequent interactive phase, and I have experienced no problems from this point of view, since the most expensive parts are the queries on the data and the predictions. Anyway, when results are being calculated, a loader is shown to the user.

Another choice I have made consists of not giving the user the possibility to select among different columns for the same graph. The reason why I have made this choice is that having, for example, a single map in which it is possible to choose among several columns does not allow by itself to compare those values, and it would be necessary to have another map, identical to the first one, where the user can select the other column to visualize. As an alternative, I may have given the user the possibility to select multiple columns and display them in the same plot area. This possibility, however, would have some drawbacks. First of all, all the plots of the tool are resizable, and resizing happens always in the same way. Having inner plots representing different quantities would require learning an additional way of resizing plots, without bringing a significant benefit. Moreover, to the best of my knowledge, implementing resizable subplots in such a way that the user can use the same procedure as in the resizing of the main plots is not natively possible, and it would be necessary to adapt Plotly code. Even though this should not be impossible, updates of Plotly may require to keep adjusting the additional code as Plotly updates, which would require programming time without having a significant advantage from the user point of view.

5

Predictions

Predicting the evolution of COVID-19, given the available data, is quite hard, since it is difficult to forecast how different governments will after the first phase, given that they are now more experienced and more prepared to the pandemic, and that they may give more importance to the country economy, while in the first phase hard lockdowns were implemented in many countries in the world. Moreover, the future development of an effective vaccine will hopefully interrupt the pandemic, but when this will be ready and available for a consistent number of people is still not clear.

The strategy I have adopted in the realization of the model has consisted of comparing several existing models (specifically, SARIMAX, Prophet and VAR) and make an analysis of how accurate they can be on the basis of the first phase behaviour. In those models, I have considered the prediction of the number of new cases per day by using different sets of data for the training. In the case of SARIMAX and Prophet, in particular, I have used the old values for the new cases and tried to make predictions based on that data. In the case of VAR, I have considered all the machine-readable variables (excluding, for example, the notes and other textual columns). The subdivision of the test and train dataset, instead, is defined by the beginning of the second phase (which has started in September, according to my observations). Then I have analyzed the situation for some specific countries.

5.1 Peak picking

My idea for the training of the different models is to identify the beginning of the last peak and to use the previous data as training and the subsequent data as the testing set. In order to identify the peaks correctly, I have used scipy, and set thresholds to avoid choosing small fluctuations that would be, in fact, noise.

Moreover, while in the case of SARIMAX and Prophet I have started the testing only at a certain point in the peak, while in VAR I start at its beginning. This difference exists since those methods consider different variables, and I have found that this particular settings yields to more promising results for the future, in accordance with the current observations, for the majority of the countries under consideration.

5.2 Differences between the models

In the three models I have considered, the prediction of the new cases per day may vary consistently for some countries. This depends on some factors:

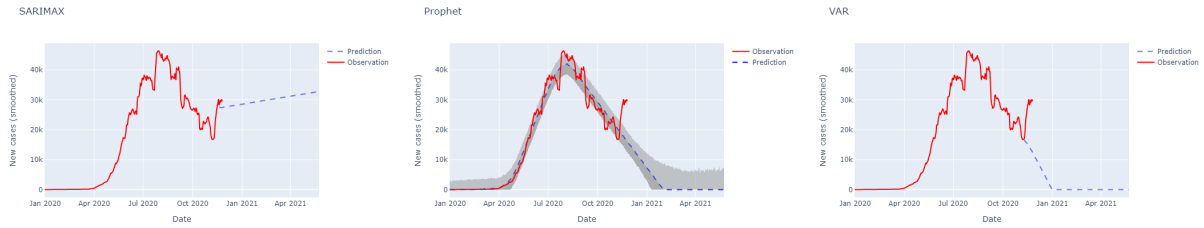
- Not all the models consider the same variables as an input, in order to avoid the user to wait a large amount of time to obtain the results
- The mathematical and statistical principles behind the different models are different, therefore the predictions work in a different way and have not been implemented specifically for the study of a pandemic, but for time series in general
- It is difficult to assess whether there are causality effects with some variables, such as the school opening, since different countries may have different policies inside the schools (compulsory masks, hand hygiene...)

Those factors may not be captured correctly by all the models. In the next section, I will analyze some case studies for some representative countries.

5.3 Case studies

5.3.1 Brazil

Brazil is an interesting case, since it is not possible to recognize different phases in the evolution of the pandemic. Differently from some European countries, which have experienced the effects of the pandemic a short time after China, Brazil has started having the first cases in April and has reached the peak in late July. In this case, Prophet and VAR find similar results, which are consistent with the idea that the one observed in late July is actually a peak, and that the curve will keep decreasing. SARIMAX, on the other hand, probably interprets the last increase in the cure as the beginning of a new peak and makes the prediction that the number of cases will increase. SARIMAX prediction may not be particularly accurate on a daily basis, since it usually finds a line for many countries, but the final point of such line may be a good prediction in case the curve will start to increase again.



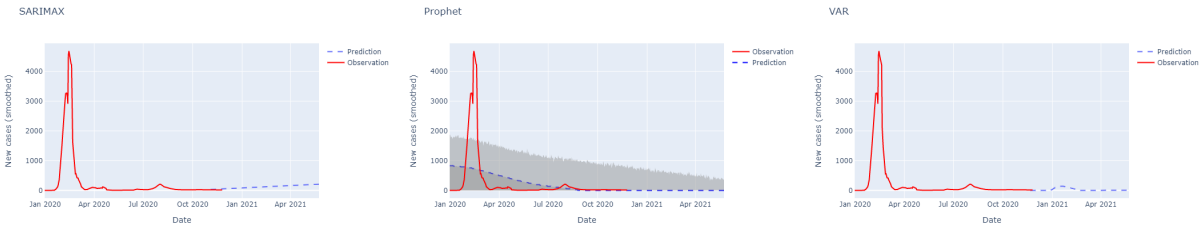
(a) SARIMAX (b) Prophet (c) VAR

Figure 5.1: Brazil

Overall, the results look convincing and the pattern looks similar to the one of other countries, where there has been an increase in the number of cases, the application of more restrictive measures (or the change in the people behaviour), and the subsequent decrease in the number of cases, at a slower rate with respect to the new cases increase.

5.3.2 China

China has been the first country to be hit by the pandemic, according to the official data, but is now one of the countries with the lowest number of registered cases in the world. Differently from other countries, China has applied very restrictive measures from the beginning, as I will analyze in the next sections about the comparison of different measures in different countries.



(a) SARIMAX (b) Prophet (c) VAR

Figure 5.2: China

In this case, there has been a relatively high peak at the beginning, and the curve has flattened afterwards. All the models predict that the number of cases will remain close to the current level, and this prediction is what I would expect: China has been able to avoid the second phase of new cases, which has hit Europe and other countries starting from the end of the summer, and since those measures are effective, changing them, according to those data, would not be a rational behaviour.

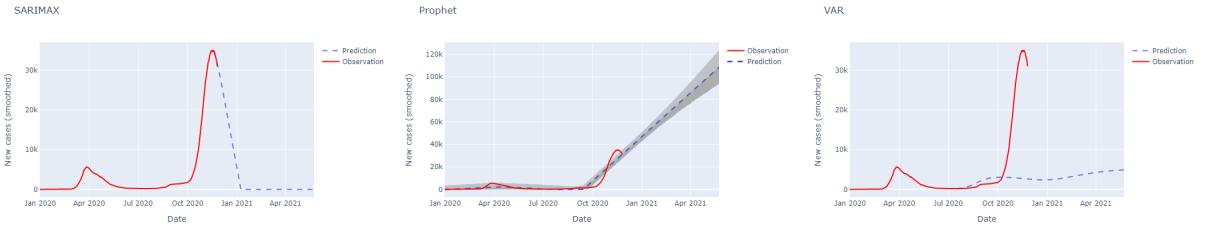
5.3.3 Italy

Italy has been the first European country with a high number of cases, especially in the northern part of the country, which is more densely populated than the southern part. In this case, the initial development

of the pandemic has progressed quite rapidly in March, and lockdown measure have been applied to some areas first and to the entire country later. For this reason, the curve has subsequently decreased, even if more slowly with respect to its increase. During the summer, the curve has remained almost flat and the measures have been lifted through time.

In this case, it is important to consider that many variables are involved, even though not all of them are included in the data in detail. First of all, life expectancy in Italy is particularly high, and older people, which may have a weaker immune system, may be hit more severely. Another factor that should be considered, and that is not captured by the data, is the fact that the pandemic has spread mainly in Lombardy, which has a higher population density with respect to other regions, and is considered one of the major industrial centers in the country. For this reason, the pandemic has severely affected the economy.

The aforementioned reasons may explain why the second phase has hit more people than before: differently from the first phase, the lockdown has been less restrictive and more localized, giving people more freedom of movement in the country.



(a) SARIMAX (b) Prophet (c) VAR

Figure 5.3: Italy

The predictions, in this case, show very different results. In this moment, SARIMAX looks rather accurate, if the current restrictions will be kept in place. However, this is not necessarily what will happen, since during the Christmas period the Italian government has announced less restrictions with respect to the current ones. This would mean that the number of cases is likely to increase again in December, following the same pattern it has already followed after the summer season. This information, however, cannot be captured using the current data, since lifting the restrictions during the Christmas period may favour the economy and the respect of traditions, even though it may not be the best possible choice in terms of minimizing the number of cases.

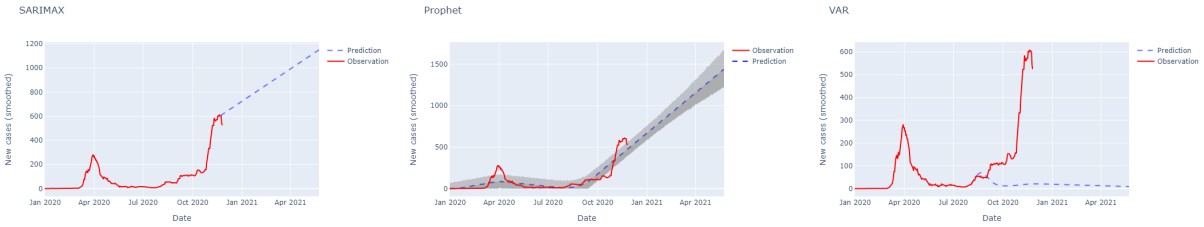
The Prophet model, on the other hand, shows a steady increase in the number of cases in the next months. This is, in my opinion, equally unlikely, since when the number of cases increases, new restrictive measures are applied, and the number of cases tends to decrease. As in other cases, I think this model is useful to forecast the overall trend of the pandemic, which can however be not very accurate on a daily basis.

Finally, VAR shows a different behaviour with respect to the actual results. I believe that this would be the consequence of completely "rational" decisions which consider only the medical data, and not economy. This pattern, indeed, can be observed in other countries as well, as I will show in the next case studies. In case the lockdown was applied earlier, the curve looks quite realistic, and it also shows the characteristic oscillatory behaviour obtained when the lockdowns are applied and lifted as the cases increase and decrease.

5.3.4 Norway

Differently from other European countries, Norway has a low population density, on average, and this factor might have affected the development of the pandemic. Moreover, the majority of the population is concentrated in some cities, such as Oslo, while some regions in the northern part of the country are less densely populated. Another difference, which is not captured by the data, is the different idea of sociability with respect to other countries, such as Italy. This factor, however, is very hard to quantify numerically.

In this case, the predictions show different results, and the same pattern as the other analyzed countries can be observed: SARIMAX tends to make a rough predictions which may be accurate in the long term, even though not very realistic in the short term, and is usually rather pessimistic. Prophet, on the other hand, considers a more "smoothed" version of the data and shows a similar behaviour with respect to



(a) SARIMAX (b) Prophet (c) VAR

Figure 5.4: Norway

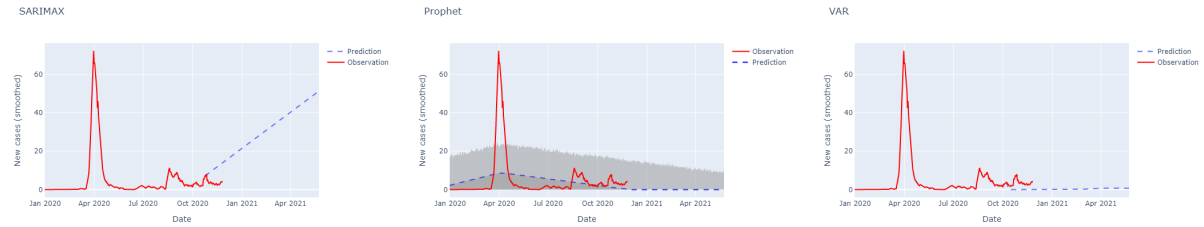
SARIMAX. Finally, VAR shows a prediction based on a "rational" behaviour, which corresponds to the the application of restrictions when a peak starts.

It is also important to notice that Norway has implemented, especially in some regions, effective tracking systems, which are able to identify the possible infected persons on the basis of their previous contacts. This system, on the other hand, has stopped working in other countries, such as Italy, where the number of cases has increased too quickly to apply this method, even in the second phase, which could be expected as a consequence of the restrictive measures lifting.

5.3.5 New Zealand

Similarly to Norway, New Zealand has a low population density and, even if this information is not captured by the data, it has the advantage of being separated by other countries by the ocean. This means that is easier to track anyone entering the country from abroad, differently from what happens in European countries, such as Italy, which have less control on the borders because of the freedom of movement rules in the European Union.

New Zealand, indeed, has had a low number of cases, and the predictions show that this number is likely to remain very low through time. Even SARIMAX, which is the most pessimistic model, shows that the number of new cases per day should not be higher than the previous peak.



(a) SARIMAX (b) Prophet (c) VAR

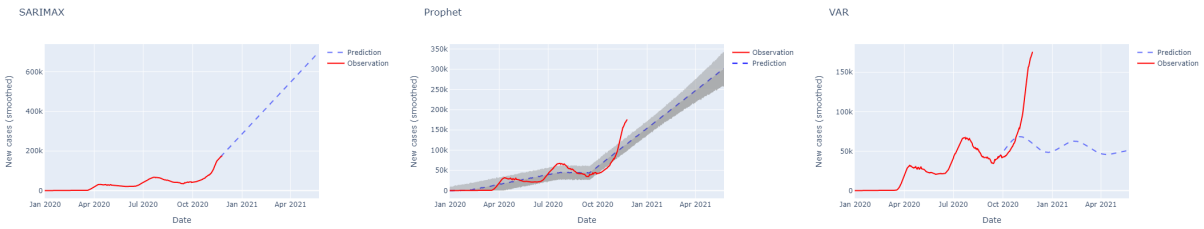
Figure 5.5: New Zealand

5.3.6 United States

The USA have shown a peculiar behaviour, in which there is not a first phase clearly separable from the second phase, since the number of cases has never went down to zero. For this reason, analyzing the predictions can be particularly interesting.

As observed in the previous analyses, SARIMAX and Prophet show a similar behaviour and are rather pessimistic, assuming that the curve will keep increasing, on average, during the next semester. VAR, on the other hand, captures the oscillatory behaviour, which should represent, again, the application of the same behaviour as in the previous phases of the pandemic.

In this case, the situation is made more difficult to predict because of the change in the President of the USA, which may propose different policies with respect to his predecessor. Those policies could go towards more restrictive measures, which would correspond to the situation predicted by VAR or, in a more optimistic case, in the same behaviour observed in other countries, where once the peak is reached the curve tends to reach values close to zero.



(a) SARIMAX (b) Prophet (c) VAR

Figure 5.6: USA

Moreover, the USA are a federal country, which means that the individual states may take some decisions independently on the others. This means that a more detailed dataset would probably be able to make better predictions. Finally, it is important to consider the impact that political leaders have on the people opinions, independently by their institutional role. This effect, again, cannot be studied in detail given the proposed data.

6

User interface

In this part of the report, I will describe the principles behind the development of the user interface, with a particular focus on consistency.

The interface is organized in sections. The top part of the interface includes a big central interactive choropleth map, where the user can select a country, compare the cases and the deaths for that selection and obtain the link for receiving more information about the current situation (from the WHO website) and display the ranking of the countries with the highest number of cases per million people, that is the same quantity shown in the map. The reason why this map is central is that the user will be attracted to it, and that is quite an important part of the tool, since the user can get updated information about the cases evolution through the WHO link, which not only contains numerical data, but also news and important information which are not included in the current dataset.

Below the map, the user can see three buttons ("General analyses", "Education" and "Predictions"). They show or hide the corresponding sections. In particular, the Education section shows the plots obtained using the UNESCO education dataset, the Predictions section shows the predictions in terms of new cases per day and the General analyses sections shows the other analyses which use the COVID dataset (but not the UNESCO dataset). When a button is highlighted, the corresponding section is active, and more sections can be activated at the same time.

6.1 Consistency

One of the principles behind the realization of an effective interface is consistency. This mean that it is important to be self-consistent and also to be consistent with what the user is used to see. In this case, I have focused on designing an intuitive interface, which should be used without a specific training by any user.

In particular, the first step the user has to perform consists of uploading the necessary data. Those data are the ones related to education (contained in a csv file), and the ones related to the COVID cases (in a csv file as well). In order to upload them the user can either choose to drag and drop the file of interest over the corresponding icon on the top right corner of the interface (the upload button for the COVID file and the education button for the education file) or click on the button and select the file from the computer. Those buttons have a dashed border to differentiate them from the other buttons in the interface, and when a file is dragged on one of them, a visual feedback, consisting of a green box, is shown. The user can upload the files in any order and can also upload a new file afterwards, which will upload the old file for that typology. The other buttons have a different style, which allows the user to see which ones have been selected.

6.2 Structure and Gestalt principles

The interface I have proposed is, in my opinion, not particularly complex, and I have designed it on this way on purpose, in order to make it easier to use. For this reason, there aren't many controls which are shown together. The most relevant case in which this happens is the disposition of the filters.

In particular, the filters are always displayed on top of the graph they refer to, and they are displayed on different lines. This guarantees that each filter is clearly distinguishable from the others and that the controls for that filter (for example, the dropdown menu button and the clear button) are placed all on the same aligned and are aligned with the plot.

The structure of the bottom part of the interface, by default, is very repetitive, which means that all the plots will have the same size will be organized as in a grid, which is adapted when the browser window

is resized. There is, for this reason, a simple, regular and predictable pattern, in such a way that the user does not have to understand how the different plots work in different cases. An advantage of having a modular code is that this pattern is automatically preserved even in the case any change is performed in the layout, for future developments of the tool.

Another important principle in the interface development is the clear distinction between the background and the other elements. I have chosen a light gray background in order to have a different colour with respect to the one of the plots and of the buttons. At the same time, I have chosen it to be light in order to avoid attracting the attention on a secondary part of the interface. On the other hand, the plots have a white background since the user may want to export them as images and, for example, include them in a document. This guarantees a better integration with the white background many documents have.

6.3 Attention and memory

It is very important to avoid asking the user to remember previous information. For this reason, I never hide the current status of interface elements. The first example is the one of the three buttons which allow the user to select the group of plots to visualize: the currently active ones are highlighted, while the other ones are not. In this way, the user will know which plots (s)he is currently visualizing and may choose to hide some of them.

Moreover, when the user selects something in a filter, this information is kept visible in the filter, so that the user knows, at a given moment, what is currently displayed. This applies for all the kinds of filters I have considered.

Finally, I have avoided the use of menus, apart from the ones in the filters. The filter menus, however, are dropdown menu with only one level of depth, and are easier to navigate for a typical user. Since I have avoided implementing a complex and hierarchical interface, the use of the so-called "Breadcrumbs" has not been necessary. On the other hand, the three buttons for the graph group selection have actually the same purpose, but for a shallow interface.

6.4 Responsiveness

It is important that the interface is responsive when the user tries to navigate it and when (s)he performs operations of the graphs. In particular, when the files are updated, the interface requires some time before the data are correctly processed. The main reason why this happens is that the predictions are computed, and even the simplest versions can take a relatively long time.

For this reason, the user can see the plots when they are calculated, and the predictions are usually the last plots to appear in the interface. However, the user can still interact with the interface during the loading phase, and even change the input files, if necessary. In this case, the calculations will start again with the new data.

This means that during the loading phase, the user has to wait for a longer time, while later the interactions with the graphs happens almost immediately, unless new predictions are performed. While the user is waiting, I have decided to have a small loader icon where the element will be. The reason why I have not used a progress bar is that the only plots which require multiple steps and more time to load are the predictions, but in that case their API does not expose any information about the residual time, and the duration of the operations may vary. A progress bar, then, would be not only more difficult to implement, but also misleading for the final user. Another decision I have taken consists of not showing any plot while the figure is being updated, in order to avoid showing misleading information to the user.

7

Use cases

This section presents some of the plots that can be obtained using this tool and analyzes their meaning in the context of COVID-19.

7.1 Population density vs cases per million

This plot is important to compare the effectiveness of the policies of several countries with different population densities. In particular, effective policies would correspond to a high population density and a low number of cases, since as the population density increases, it is more difficult to apply social distancing effectively. On the other hand, ineffective policies would correspond to countries with a low population density but a high number of cases per million people.

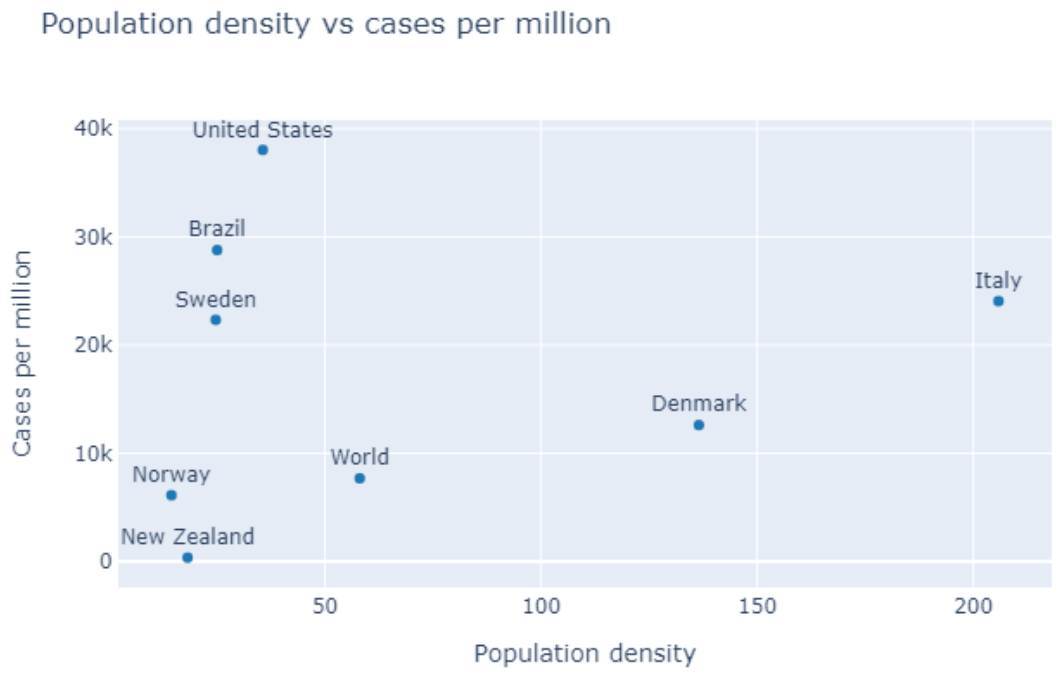


Figure 7.1: Population density vs cases per million

Considering the countries represented in Figure 7.1, it is possible to observe that some countries have a low population density and have performed really well. This is the case of New Zealand, which, as already discussed previously, has the advantage of being an island and to have full control on the people who come from overseas.

On the other hand, United States have the double of New Zealand population density, but a much higher number of cases per million. Given only this plot, it is not possible to draw conclusions on why this has happened, but in my opinion the different geographical location of the United States and possibly different cultural attitudes and large cities may have contributed negatively to a wider spread of the virus.

Brazil and Sweden show a similar situation, even though they may have different cultures. On the other hand, Sweden and Norway are in the same geographical area, but while Norway has had fewer cases per million people, Sweden has shown a wider spread of the virus. A possible reason could be the presence of a lower stringency level in Sweden.

On the right side of the plot, it is possible to see countries with a higher population density, such as Denmark, Italy and China. Denmark shows an intermediate situation with respect to Norway and Sweden in terms of cases. The situation may be worse than Norway, for example, because Denmark is more easily reachable from central European countries, such as Germany, France, Belgium and the Netherlands. Norway, on the other hand, is less central, from a geographical point of view.

China has been the first country to be hit by the pandemic but the number of cases has rapidly decreased after the first phase of the pandemic, since restrictive measures have been quickly applied. For this reason, the total number of cases is lower than the ones in New Zealand notwithstanding the high population density.

Italy has been one of the countries hit most severely by the pandemic when it has reached Europe. It has spread mainly in the Northern part of the country, where the population density is higher than in the South, and I think this is one of the reasons why the number of cases per million is particularly high. It is interesting to note that Italy has had a comparable number of cases per million with respect to Sweden, even though it has a significantly higher number of cases. This may suggest that Italian policies have been more effective. The same comparison can be performed with the United States, and in this case it is even more likely that Italian policies have been more effective than United States' ones, since the number of cases per million is lower notwithstanding the population density.

7.2 School opening status

Here I present the school status plot, which analyzes the education institutes opening from the beginning of the pandemic to the last available date in the UNESCO education dataset (the numbers then may slightly change in case of an updated version of the dataset).

In this case, I have used different shades of the same colour to represent different attributes for the same country. The darkest colour refers to the fully open status, while the lightest refers to the fully closed status because of COVID-19. The sum of the three bars may not be the same for different countries because academic breaks are not displayed, since in that case there is not direct correlation with the number of COVID cases.

In this case, it is possible to observe that Norway, Sweden and New Zealand have kept schools opened for most of the time. This phenomenon can have several explanations, related to the discussion related to the previous graph, as I will also analyze more in depth subsequently. In the case of Norway and New Zealand, the number of cases has been particularly low, therefore schools have not closed if not during the most problematic phases. Sweden, on the other hand, has never implemented full closures but its policies have been less restrictive than the ones in other countries. China and Italy, on the other hand, have both kept the schools closed or partially closed for most of the time.

According to those examples, which have been selected to be representative in terms of different policies and number of registered cases, it looks like the schools closure is the result of the application of stringency measures, rather than of the number of cases per million people.

From the visualization point of view, the colours of this plot are not a strict necessity, as long as their shade changes for different attributes. Colour blind people can easily read the plot even without the use of colours, since each colour refers to a different country, and each country is separated from the other by a space. However, they can also choose the color-blind version, if they prefer.

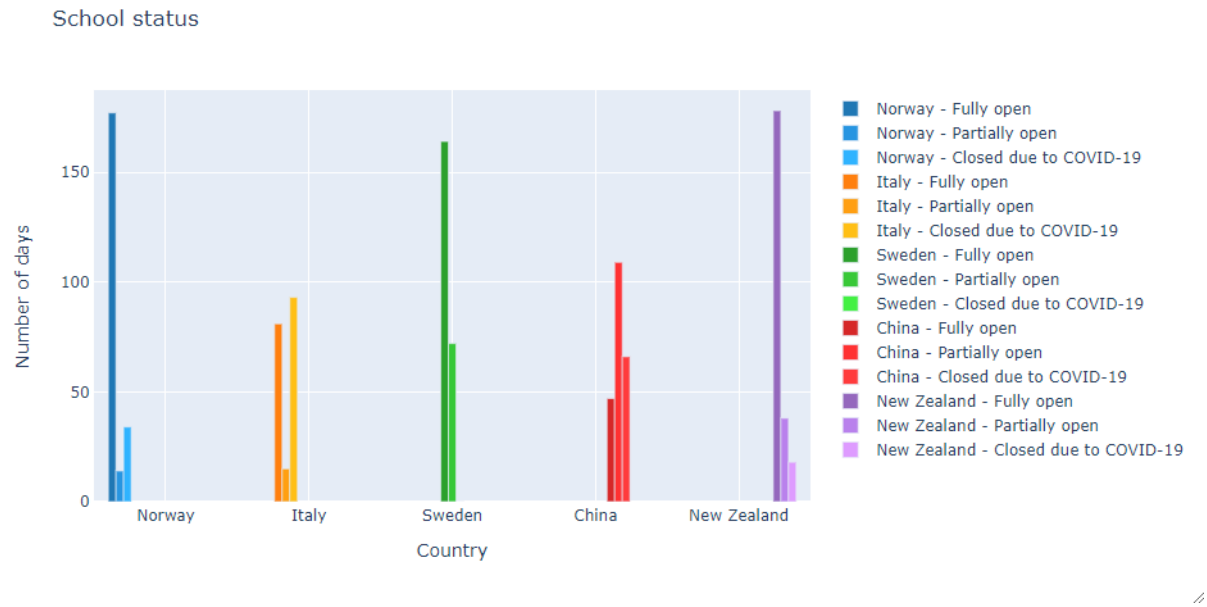


Figure 7.2: School status

7.3 Education stringency coherency (ESCO)

The purpose of this index is to show the different behaviour of countries in terms of education with respect to the overall restrictive measures. It is computed by assigning a value of 100 to fully closed school, of 50 to partially open schools and of 0 to open schools. Then, ESCO is computed as the difference between the stringency and the opening status, expressed with the aforementioned numbers. Negative values indicate that the stringency in schools has been higher than the global stringency, while positive values indicate that stringency in schools has been lower than the global stringency.

There are some interesting insights that can be obtained by those plots. It is possible to notice, for example, that after the school break some countries have decided to have a less stringent behaviour in schools. This applies to Italy and Finland, for example. This phenomenon may be explained by considering that the summer break has allowed schools to better organize the spaces and to define new protocols to guarantee safety for everybody in the school environment.

Another pattern, that can be observed in several countries, is that in the first phase of the pandemic, the response in terms of education has been in general quick and oriented towards stringent measures, which have been in some cases lifted (with respect to the overall pandemic evolution) after some time (for example, in Norway, Sweden, and Denmark). However, the behaviour of the countries taken into consideration shows that some of them have not had a consistent behaviour through time, since the curve oscillates. This may indicate the fact that, in this extraordinary situation, governments were not always prepared to face the pandemic. The countries which have shown a more consistent behaviour for the ESCO index are United States, Brazil and Iceland. This does not necessarily mean that their policies have been effective, but rather that they have probably taken their decisions by considering the new cases impact in the same way through time.

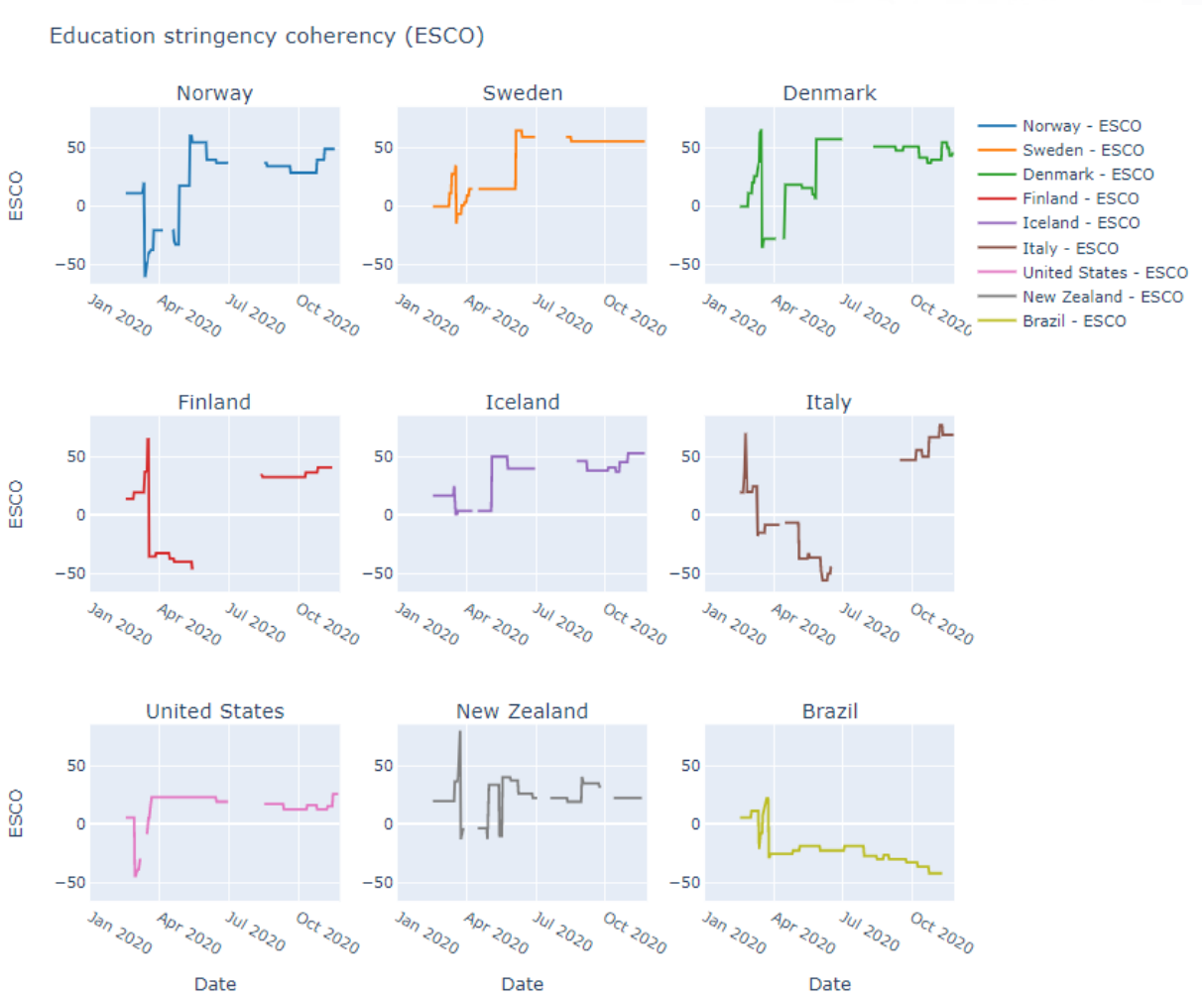


Figure 7.3: ESCO index

7.4 School stringency

This plot presents the stringency applied in the schools through time, where 1 refers to fully open schools, 1.5 to partially open schools and 2 to fully closed schools due to COVID-19. This measure is useful to see how different countries have behaved in terms of education and can be complementary with respect to the ESCO index.

In particular, it is possible to observe that some countries, even though they have not had many cases, have decided to completely close schools. This is the case of New Zealand, for example, even though the fully closure has not lasted very long. Brazil, on the other hand, is still keeping the schools closed, even though the new cases curve is decreasing, as shown in the prediction analysis performed before. This is a prudent behaviour, as the ESCO index also shows.

Italy, on the other hand, has behaved in the same way as Brazil at the beginning, and has now restored fully open schools, according to the analyzed data, which in this example last only until the 23rd of November. However, it is possible to upload the updated data in the tool to see whether there are relevant changes.

Finally, other two interesting cases are Sweden, Iceland and United States. The ESCO index has shown a consistent behaviour through time, and the closing status is also consistent. This means that the overall stringency through time has not varied consistently. Those countries have also never completely closed the schools. A possible explanation of this phenomenon can be the related to a low number of cases, to the implementation of effective protection measures or the overall implementation of more permissive policies.

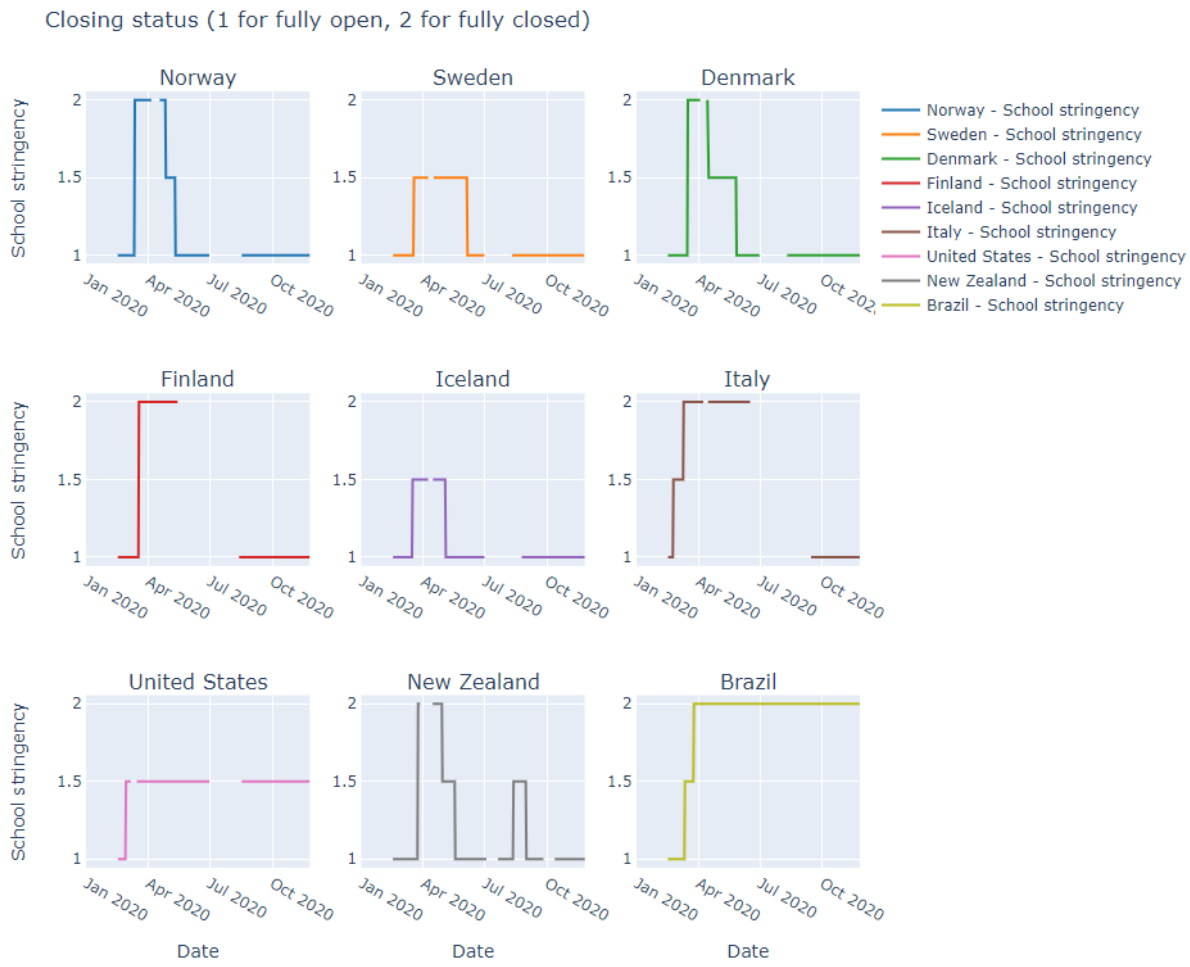


Figure 7.4: Closing status (school stringency)

7.5 Total cases for Nordic countries with a median age greater or equal than 40 years old

This plot allows the user to choose among different measure to visualize. In particular, the user can input the countries of interest and apply the median age filter. The two filters (countries and median age) will act together on the data, therefore, among the selected countries, only the ones with a median age that is greater or equal than 40 will be displayed.

Then, the user can also select the quantity to visualize by means of the legend on the right side. In this case, for example, I have selected the total cases per million people, and Sweden is the country with the highest number of cases. Even if Norway was selected, the median age is lower than 40, and for this reason it is not displayed.

The reason why filtering by median age is important is to allow the user to select all the states (s)he prefer and then analyze the impact of COVID-19 for different median age ranges, to see for example if countries with a similar median age show a similar behaviour. The plot is logarithmic to allow the user to possibly compare, for example, different parameters easily (for example, the total cases and the total deaths, which are in general much lower than the total cases).

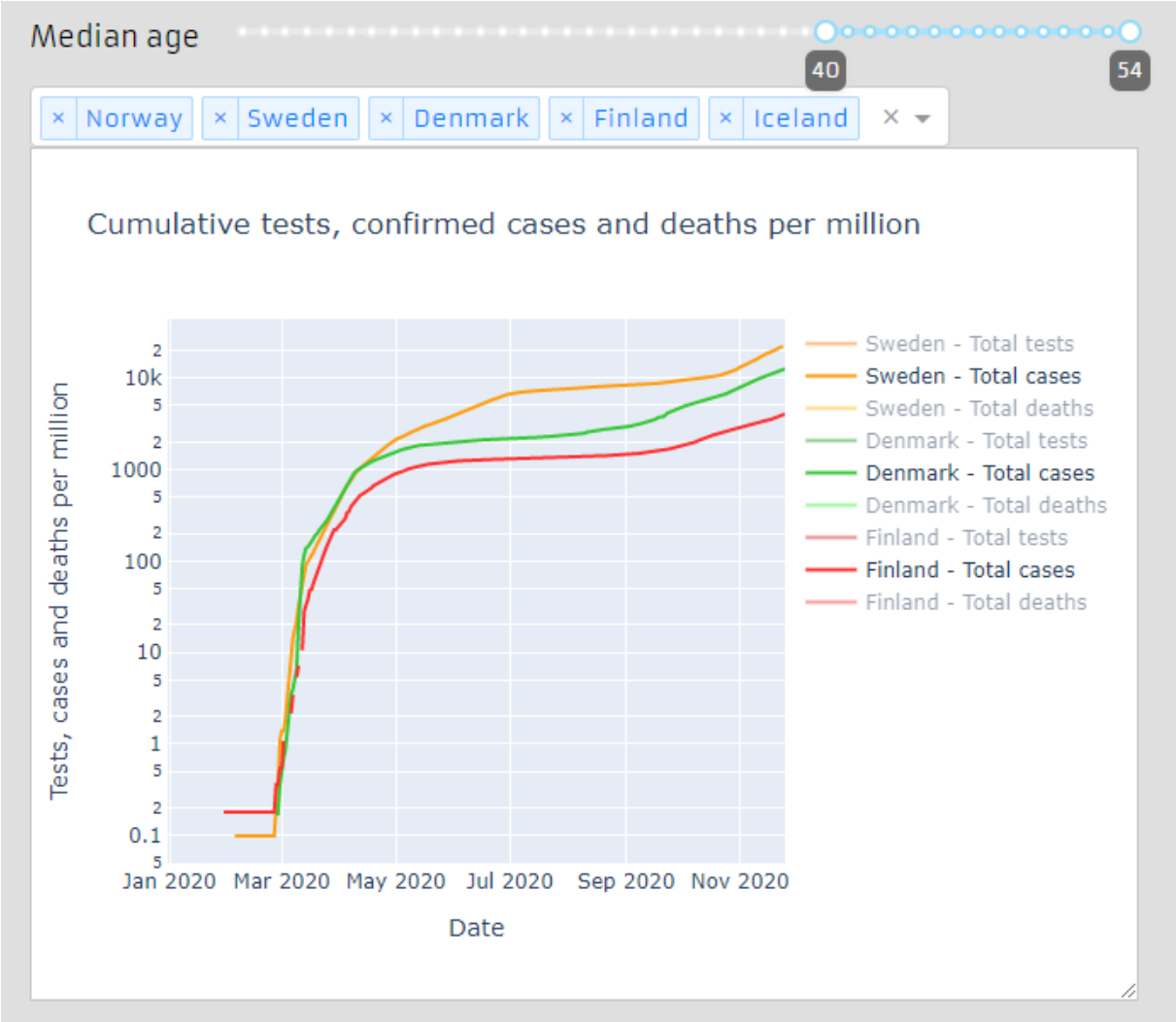


Figure 7.5: Total cases for Nordic countries with an average age higher than 40 years old

7.6 New ICUs with respect to the previous day

This plot is important to understand how the situation in the Intensive Care Units changes on a daily basis. In particular, a peak in this plot represents a very quick increase in the number of patients admitted in the ICUs. This spike can be observed at the beginning of the pandemic, when stringency measures were probably not particularly effective. Later, most of the countries have a more stable situation, and in particular the end of the first phase of the pandemic corresponds to negative values in this plot, which means that there are less people in the ICUs. A value of zero should not mislead the observer, since it means that the new admissions to the ICUs are the same as the previous days. Unfortunately, not all the countries have shared data about ICUs, therefore only some of them can be analyzed.

Those data are useful because they all show that the evolution of the ICUs admissions has been similar across countries with different policies, even though some of them (Denmark and United States) have had spikes in the ICUs admissions even after the first phase. This may indicate either inconsistencies in how the data have been collected or new big outbreaks. For example, the United States show a peak in the Summer, which may be related to more people meeting together with respect to the winter season. This information, however, is not given by the dataset, and this remains an hypothesis.

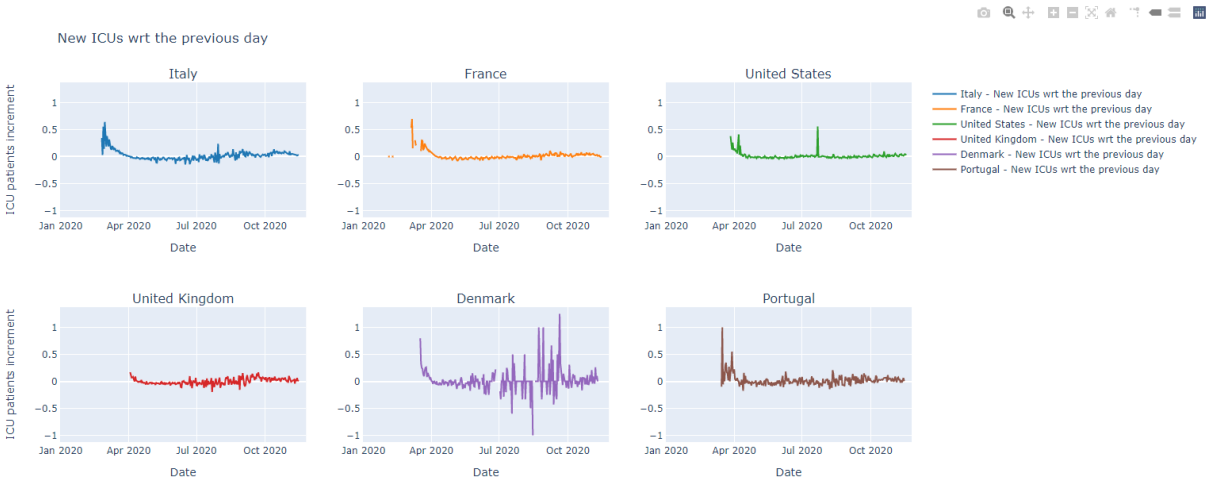


Figure 7.6: New ICUs admissions with respect to the previous day

7.7 New hospitalizations with respect to the previous day

This plot is useful for reasons similar to the previous one. In this case, however, all the people admitted to hospitals are considered, and not only people admitted to ICUs. It is important to observe, in particular, that a similar pattern can be observed in the considered data. However, the initial spike is higher for some plots. For this reason, here I show only a specific interval for the y axis, which the user can select in the graph by using coordinate views.

Two interesting facts in this case are the presence of a spike in July in the case of the United States, that can be observed also in the case of ICUs, and, in the case of Denmark, we can observe the same patterns as well, and my hypothesis is that measurements were not taken consistently through time (or that there have been localized outbreaks).

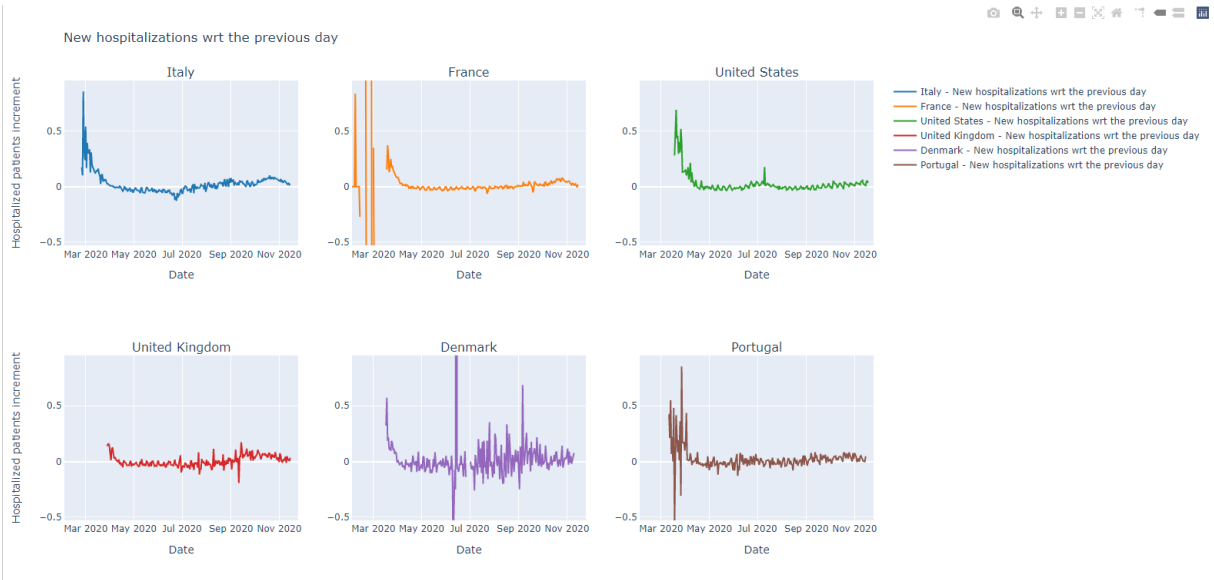


Figure 7.7: New hospital admissions with respect to the previous day

7.8 Deaths to cases animated map

In this case, I use animated map to show the evolution of the mortality rate so far, that is defined as the number of deaths divided by the number of cases to date. Higher numbers mean higher mortality, and they are represented with red colours. This map is useful because it visually shows the evolution of the

deaths to cases ratio through time. This may help the user in visualising which countries are handling the emergency in the best way in terms of the effectiveness of the cures in the hospitals.

It is possible to observe, for example, that Italy has a high mortality rate in the first phase, while later the mortality decreases, which may mean that the health system has probably improved through time, or that the load on the system has became lower as time passed.

Here I show two pictures for two different cases. It is possible to observe that the second date, later in time, shows a lower mortality rate, which is a good indicator of the effectiveness of the health system. However, updated data don't arrive from all the countries on the same day, and for this reason some countries are gray in case of the second plot.

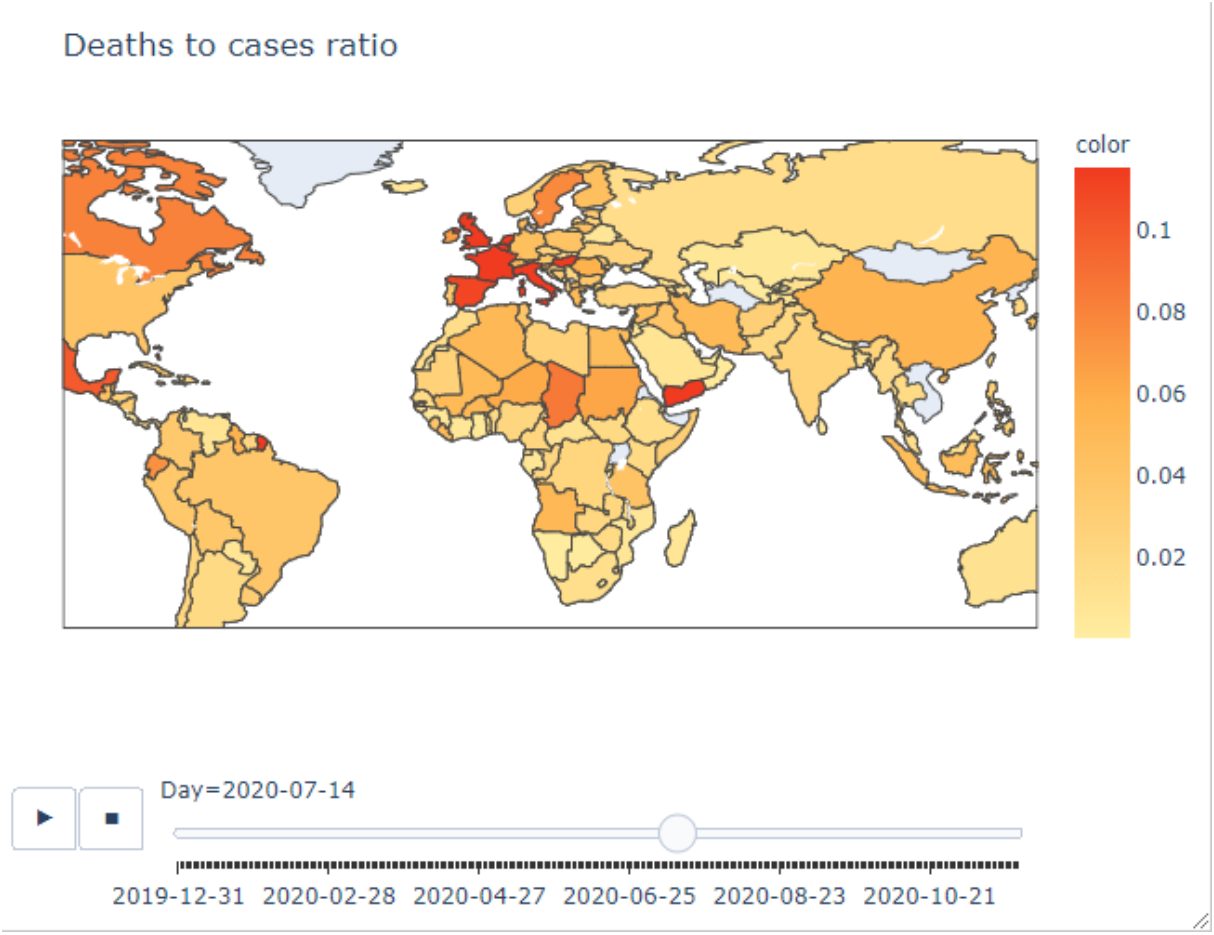


Figure 7.8: Deaths to cases (14.07.2020)

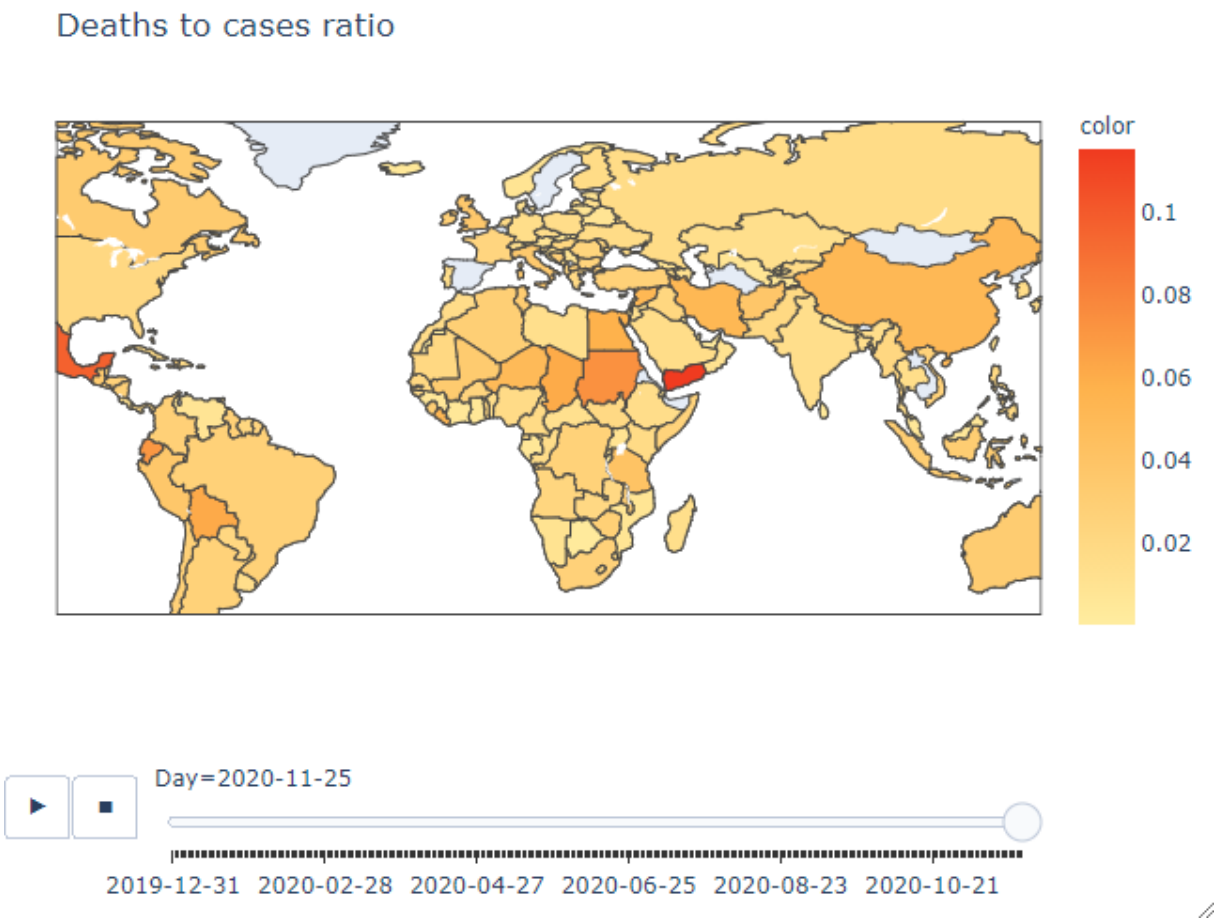


Figure 7.9: Deaths to cases (25.11.2020)

7.9 Other animated maps

The tool also allows to visualize other animated maps, which follow the same principles as the one presented in the previous section. For example, it is possible to visualize the number of cases and the number of deaths through time. Those maps can be particularly useful if the user wants to analyze the how the pandemic has spread in the world both in terms of cases and deaths. However, I am not describing those maps in detail here, since this would be repetitive with respect to the previous analyses, in my opinion.

7.10 Deaths to stringency ratio

Deaths to stringency ratio (DSR) can be used to measure how important the deaths have been for the determination of more stringent measures. I analyze both the "immediate response" (that is, considering the decisions with respect to the same day as the number of deaths) and the "delayed response", which is instead important since governments also need to evaluate the data before taking new decisions.

In both cases, increasing values indicate a lower importance of the number of deaths for the determination of new stringent measures, while decreasing values indicate that the deaths are considered more determinant than in the past. For this analysis, it is more meaningful to consider countries with a consistent number of deaths, therefore New Zealand, for example, should be excluded.

In both the cases, the behaviour is similar, but shifted, and this suggests that both the indexes can be used to measure this ratio and they will give the same information, from a qualitative point of view. This may also indicate that the implementation of restrictive measures is in general based on a preliminary evaluation and prediction of the number of deaths, which yields to similar values of stringency at a distance of one week.

It is possible then to compare the behaviours of different countries. In particular, many countries present a peak in the initial phase, which indicates new measures taken in an emergency situation and

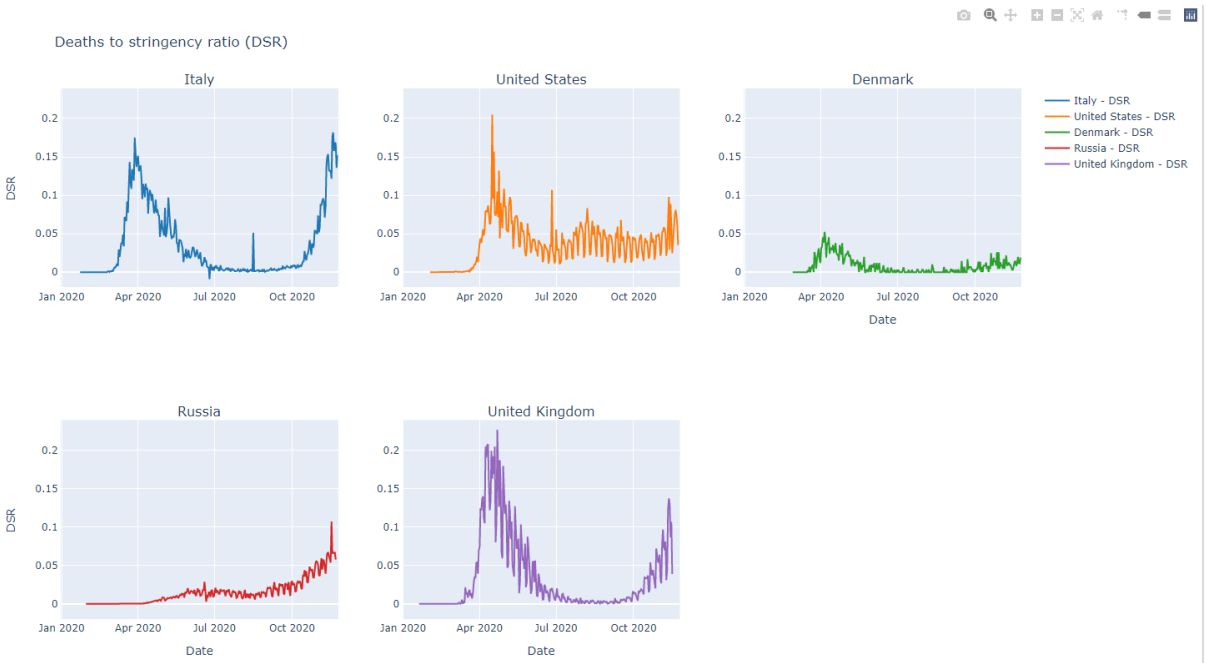


Figure 7.10: DSR (no shift)

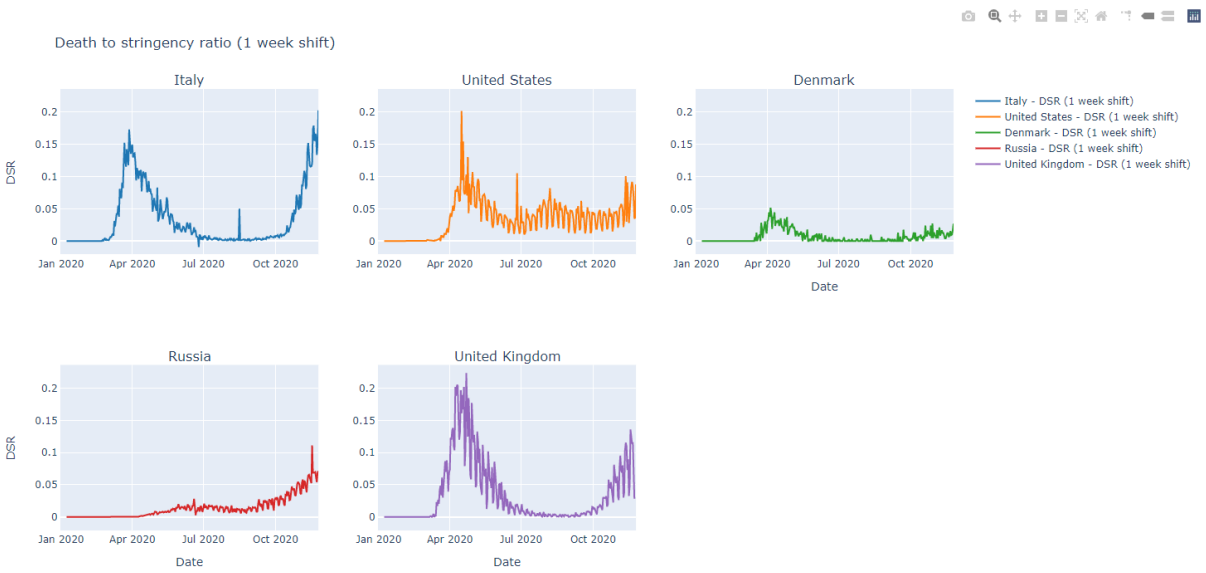


Figure 7.11: DSR (1 week shift)

based on the sudden increase of the number of deaths, which the country was not able to face immediately. In the case of Italy, this has happened for both the COVID phases. Other countries, such as Russia, have taken more restrictive measures at the beginning and are now being less restrictive in relation to the number of deaths. Denmark, on the other hand, has shown a comparable behaviour with respect to Italy, but the low peak indicates a better response, since the deaths are evaluated per million. United States, on the other hand, have never really applied tougher measures as much as other countries, since the central part of the plot is lower than the initial part, but still higher than in other countries. This may also be determined by the geography and population density of this country, which is less densely population than Italy and United Kingdom, for instance.

The tool also allows to compute, for example, the new deaths to new cases ratio, which is based on the same principles as this example. In that case, the ideal situation is to have lower values, which indicate a lower mortality rate.

7.11 New cases per square kilometer

It is important to consider the number of cases per square kilometer for a user who has to travel. In particular, one of the possibilities to move in Europe consists of using the train, which can replace the plane, especially for short journeys and in case the planes are cancelled because of COVID-19 restrictions. In that case, the user would probably like to minimize meetings with infected people. In this case, knowing the number of cases per million is not so important, since what really matters is the number of people met during the journey.

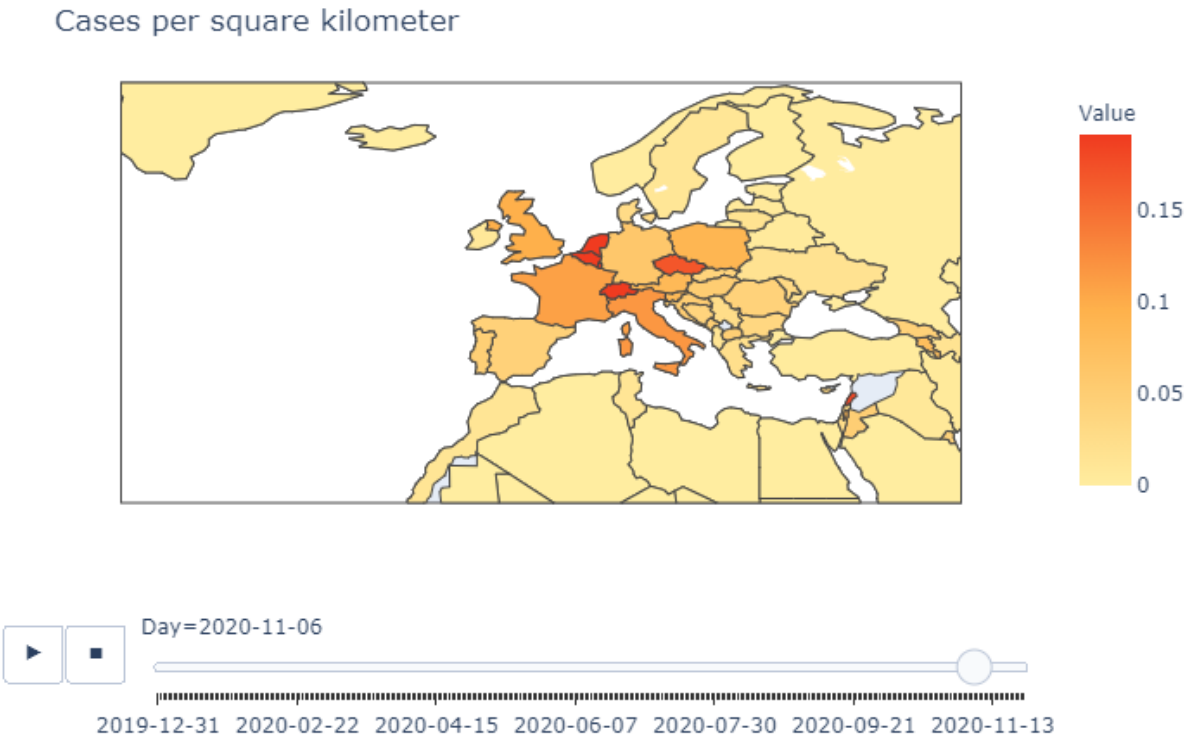


Figure 7.12: New cases per square kilometer

An animated map shows the evolution of the cases through time, and allows the user to determine, on the basis of this, which countries are safer to cross, and which ones are instead more dangerous. Considering the example represented in this case, let's suppose that the user wants to travel from Norway to Italy. Then (s)he will probably try to avoid Czech Republic, Belgium, France, the Netherlands and Switzerland, and would rather prefer, for instance, Sweden, Denmark, Germany and Austria. This information, then, can be used by a user also while (s)he is travelling, for example using flexible train tickets, to change his/her plans as the situation evolves. However, this kind of information may not be particularly useful for plane travellers, since they don't have to cross a country, but to stop at a certain (hopefully safer) location.

7.12 Total cases per million people with multiple filters

This plot gives the user the possibility to study the total number of cases per million people by using multiple filter. In particular, I allow filtering by median age, population density, GDP per capita, population, life expectancy, human development index and countries. The initial values of all the filters but the countries is set to the range reported in the dataset, while there is initially only one country to avoid having a long list of names in that filter. I think that the user is more likely to be interested in some of the countries, and not in every country in the world. The overall plot shows the data for the countries which satisfy all the other filters, among the ones selected.

An example of a possible result obtained for the considered filters is presented in Figure 7.13. In this case, for instance, it is possible to observe that there is not a common behaviour in terms of total cases, which suggests that it is necessary either to adjust the filters, or to consider new filters.

However, it is very easy to add new filters in the tool code. For example, if a programmer wants to filter by population (s)he can write, before the countries filter and after `axes = []`:

```
filters.append(Filter(filter_name = "Population", filter_type="RangeSlider", column_name = "
    population", n_steps = 40))
```

where the overall plot is defined in the following way:

```
# Multi-filter
axes = []
axes.append(Axis("Date", 'data["date"]'))
axes.append(Axis("Total cases per million", ['data["total_cases_per_million"]'], ["Cases per million"
    ]))

filters = []
filters.append(Filter(filter_name = "Median age", filter_type="RangeSlider", column_name = "
    median_age"))
filters.append(Filter(filter_name = "Population density", filter_type="RangeSlider", column_name = "
    population_density", n_steps = 100))
filters.append(Filter(filter_name = "GDP per capita", filter_type="RangeSlider", column_name = "
    gdp_per_capita", n_steps = 20))
filters.append(Filter(filter_name = "Life expectancy", filter_type="RangeSlider", column_name = "
    life_expectancy", n_steps = 100))
filters.append(Filter(filter_name = "Human development index", filter_type="RangeSlider", column_name
     = "human_development_index", n_steps = 100, prec = 2))

filters.append(Filter(default_value = ["Norway"], column_name = "location", multi = True))

graph_infos.append(GraphInfo(dataset = jsonified_data,  divide_traces = True, title = "Cases per
    million (with multiple filters)", axes = axes, filters = filters))

graph_divs.append(new_custom_graph())
```

This modular structure has allowed me to write more generic code, which can possibly be adapted to any kind of filter.

Figure 7.13: Total cases per million people with multiple filters

8

Course feedback

8.1 Challenges

In this project, the main challenge has consisted of using Dash callbacks in an effective way, which for me means having a modular structure and avoiding repeating the same code several times, since such a code would be difficult to modify subsequently. Another challenge has been related to the understanding of the amount of detail for the report and for the project. I hope I have not written too many analyses or viceversa. Another challenge is related to the quality of the data. Not all the countries collect the data in the same way, and some of them don't share their data, or share only part of them. Making predictions, then, becomes quite difficult.

8.2 Time demand

While the first part of the course has been relatively straightforward and the workload has been balanced, the project has taken more time. This may depend on the way in which I have worked on it, which may have been too programming-oriented. An estimate of the time spent on this project is about 60 hours. This workload, in my opinion, may be reduced by giving more examples of what is expected in the assignment description, and the daily workload may be decreased by publishing the assignment at the beginning of the semester, so that the students can implement each part when they study the corresponding topics.

8.3 Missing parts

I don't feel there is something missing, given the topic, but I would have preferred a free choice for the topic, since this would have allowed to study particular topics of interest, apart from the COVID-19 pandemic. In this assignment, on the other hand, the topic was determined, while the specific requirements were quite free. I would have preferred the opposite. For example, a topic of our choice with some constraints on the visualization. This may have changed the workload, but it would have been something on which everybody could be really passionate about.