

# COVID-19: Behind The Numbers

Data Mining and Machine Learning Project

Prof. Marcelloni Francesco

Prof. Ducange Pietro

*Rambod Rahmani*

Master's Degree in Artificial Intelligence and Data Engineering

July 21, 2021

## Contents

# 1 Introduction

**Decemebr 31, 2019:** *China, Wuhan Municipal Health Commission reported a cluster of cases of pneumonia in Wuhan, Hubei Province.*

**January 1, 2020:** *World Health Organization (WHO) had set up the Incident Management Support Team across the three levels of the organization.*

**January 5, 2020:** *WHO published the first Disease Outbreak News on the new virus. This was a flagship technical publication to the scientific community.*

**January 12, 2020:** *China publicly shared the genetic sequence of COVID-19.*

At the beginning of 2020, a new virus started spreading around in the capital of Central China's Hubei province: the city we all came to know as Wuhan. As it turned out, this was the start of a world-changing event with overwhelming extent: Coronavirus Disease 2019 (COVID-19). After the first and the second waves of the virus have passed over the entire world, while the number of deaths by COVID-19 infections is decreasing as a result of the vaccination campaigns, the aim of this work is to address the following questions:

- **Which countries have been affected the most by COVID-19?**
- **Is it possible to build personalized predictive models for symptomatic COVID-19 patients based on health preconditions?**

In order to fully answer these questions, first of all a reliable and big enough dataset is needed. Second, data mining and machine learning techniques can be applied in order to obtain statistically significant results that could help address the proposed questions. In the following pages the development of the project and the resulting Python software are presented. The software architecture is presented in the very last section in order to focus primarily on the dataset retrieval and preprocessing, and on the analysis techniques and results.

All the files related to this work are available in this GitHub repository: <https://github.com/rambodrahmani/covid19-behind-the-numbers>

## 2 Dataset

Two different datasets were used:

- the data on confirmed cases and confirmed deaths is updated daily and is published by the **Johns Hopkins University**; this is the best available dataset on the pandemic at global level;
- as far as it concerns medical preconditions of COVID-19 patients, the most detailed dataset I was able to retrieve is the one provided by **The Federal government of Mexico**.

The content of the datasets, the features of interest used in what follows and the preprocessing procedures applied on each of them are detailed in the following subsections.

### 2.1 COVID-19 Daily Data

The Johns Hopkins University Center for Systems Science and Engineering (JHU CSSE)<sup>1</sup> provides the best available global dataset on the COVID-19 pandemic. Multiple sources were used in the data set, since January 21, 2020:

- World Health Organization (WHO);
- European Centre for Disease Prevention and Control (ECDC);
- US Centers for Disease Control and Prevention (UCDC);
- Los Angeles Times;
- The Mercury News.

The JHU CSSE data is provided as a collection of daily `.csv` files which need to be merged to obtain the dataset with all the daily data for all the countries. Luckily, the **Our World in Data** organization — a collaborative effort between researchers at the University of Oxford, focused on "research and data to make progress against the world's largest problems" — provides the JHU CSSE data already merged<sup>2</sup> and updated to the latest second as a single `.csv` file. The choice was made to use this `.csv` file as dataset.

The dataset file is named `owid-covid-data.csv` with a size of 26.2 MiB.

This dataset was used for the first part of the work: finding the characteristic curves of each country and grouping together countries with comparable behavior by means of time series clustering algorithms. This will provide us a truthful overview of the countries most affected by COVID-19 and the ones that put in place appropriate policies to handle the pandemic widespread.

#### 2.1.1 Preprocessing

This dataset contains worldwide, per-country, daily data for COVID-19 for a total of 59 columns and 103349 entries. Among others, we are interested mainly in the following features:

- total confirmed cases;
- new confirmed cases;

---

<sup>1</sup><https://coronavirus.jhu.edu/map.html>

<sup>2</sup><https://ourworldindata.org/coronavirus-data>

- total deaths;
- new deaths;
- total confirmed cases per one million population;
- new confirmed cases per one million population;
- total deaths per one million population;
- new deaths per one million population.

**Each country is represented by different time series with daily granularity. One for each of the listed attributes.**

The main considerations to be made about this dataset as far as it concerns data pre-processing are:

- it contains redundant data, e.g. values aggregated by continent (Asia, Africa, Europe, America, North America etc...);
- some of the daily data values are missing (specially for the very initial months of the pandemic, i.e. from 2020-01-01 to 2020-03-01);
- some of the daily data values are negative;

as a result, the following preprocessing procedures were applied

- data aggregated by continents was removed;
- missing daily data values for countries were replaced<sup>3</sup> with the constant value 0; this seems the most reasonable choice since these missing values refer to the very beginning of the pandemic; using either the mean or the mode results a distorted dataset;
- negative values were replaced using the moving average<sup>4</sup> with a moving window of size 15 to approximate the negative values as the average of the 7 previous and successive values.

Taking into account also the fact that the daily data will mostly be used resampled with weekly frequency and that by visual inspection it appears that negative values are close to zero, there is no need to further preprocess this dataset.

## 2.2 COVID-19 Medical Preconditions Data

The second part of this work focused on frequent pattern analysis in order to be able to find interesting patterns as far as it concerns COVID-19 patients who had prior medical preconditions. Finding such patterns is a crucial task that might allow to best allocate very limited medical resources. The rapid global spread of the virus SARS-CoV-2 has provoked a spike in demand for hospital care. Hospital systems across the world have been over-extended, including the one case we are most familiar with, Northern Italy.

As a result, decisions on how to best allocate very limited medical resources have come to the forefront: who to test, who to admit into hospitals, who to treat in an Intensive Care Unit (ICU), who to support with a ventilator.

---

<sup>3</sup>The `scikit-learn SimpleImputer` was used to this end.

<sup>4</sup>The `pandas DataFrame.rolling` method was used to this end.

The dataset required for such type of analysis was not easy to find: usually COVID-19 datasets only contain daily values (numbers) of confirmed cases and deaths. They rarely come equipped with the medical preconditions of the patients. Luckily, the Federal government of Mexico[7] provided such a dataset. It is splitted in three main files:

- `210206COVID19MEXICO.csv`: the main file containing, among others,
  - patients sex, age, admission date, COVID-19 test result;
  - if the patient required hospitalization, intubation or Intensive care unit (ICU);
  - if the patient was affected by Pneumonia, Diabetes, Asthma, Hypertension, Cardiovascular disease (CVD), Immunosuppression, etc...;
  - death date (only for those patients who actually died).
- `201128_Catalogos.xlsx`, `201128_Descriptores.xlsx`: contain additional clarifications regarding each feature present in the main dataset file.

This dataset is perfect for the objective of the analysis we are interested in — develop personalized models that predict the following events:

1. hospitalization;
2. need for ICU;
3. need for a ventilator;
4. mortality.

### 2.2.1 Preprocessing

The preprocessing required by this second dataset is completely different from the one required by the first one. This is because the type of data is completely different. While the first dataset is primarily a sequence of numerical records, here we deal with categorical attributes. The main considerations to be made include:

- the dataset is split into 3 files: one with the main content, the remaining two files contain headers details;
- Spanish is the de facto national language spoken by the vast majority of Mexicans, all files are in Spanish;
- the dataset has a size of about 1.2 GiB, with a total of 40 columns and more than 7.950.230 entries.

The preprocessing stage included merging the 3 files into a single `.csv` file, in English, containing therefore patients details, medical preconditions, required medical care and if they survived SARS-CoV-2 or not.

Additionally, in the original dataset the categorical variables related to medical preconditions include additional information which are not useful for the frequent pattern mining process.

This table summarizes the binary encoding adopted:

Numeric Value	Meaning	Binary Encoding
1	Yes	True
2	No	False
97	Not applicable	False
98	Unknown	False
99	Not Specified	False

### 3 Analysis

As said in the introductory section, the analysis was carried out using data mining and machine learning techniques in order to answer the proposed questions. As we move on to build our models, in this very particular case, working with data from a pandemic such the COVID-19, we should do that by **taking a very critical look at models**. Keep in mind the aphorism "All models are wrong", often expanded as "All models are wrong, but some are useful". The modeller's paradox is even more true when crisis erupts: unfortunately, in such situations, data are lacking or of poor quality. What is worse is that models do not simulate the behaviours of citizens to stop seeking medical assistance in the case of a pandemic (because of fear), models do not capture the effects of sustained lock down as far as it concerns mental health and general social well being. Models can not simulate the social and economical consequences of business failures and economic depressions, nor do models simulate the effects of increase of violence and policing on cultural norms or attitudes towards democracy. And also importantly, is that models do not translate well necessarily across different cultural and political boundaries.

#### 3.1 Which countries have been affected the most by COVID-19?

To answer the very first question, we need to understand what is hidden behind the official numbers and charts of confirmed COVID-19 active cases and deaths. We are not doing anything clever at all, just plotting the data and trying to learn from it.

We are so used to watching COVID-19 numbers and charts nowadays and sometimes we think we might even understand how the pandemic is evolving as days goes by. For example, it is very common to consider the following plot:

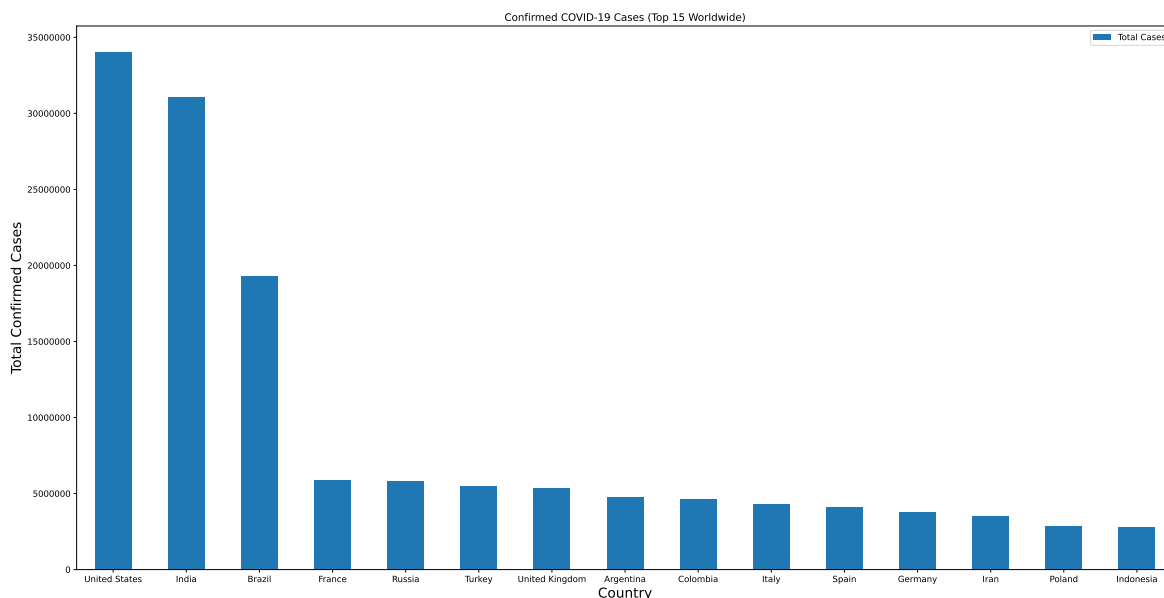


Figure 1: Top 15 Countries Confirmed COVID-19 Cases

Is this really the right choice? Does this ranking tells us anything meaningful about the current undergoing pandemic situation? From what we can observe in figure 1, clearly United States has higher confirmed COVID-19 cases than countries such as Spain or Italy. Taking into account that the US is a much bigger country, we can agree that this results

do not imply that the US is more affected than Spain, Italy or Germany. We can therefore think of a fairer comparison independent of the country size: the number of infections needs to be normalized to the population of each country. This provides a more coherent view of the countries most affected by COVID-19:

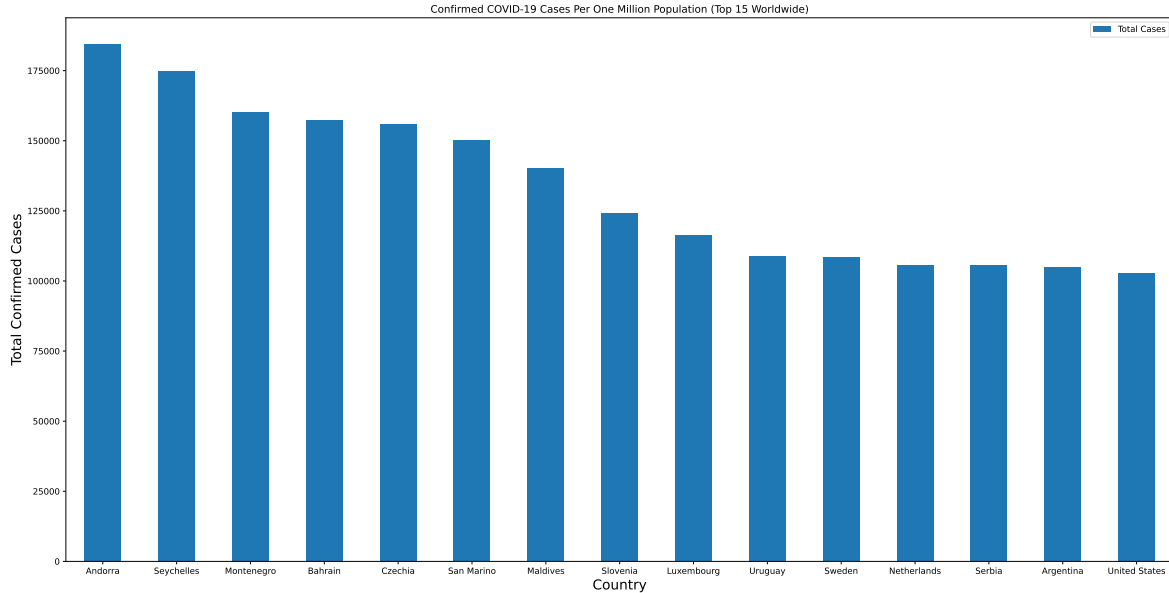


Figure 2: Top 15 Countries Confirmed COVID-19 Cases Per One Million Population

However it is not yet good enough: countries have different testing policies and a more intensive COVID-19 testing rate gives more confirmed cases while no testing at all would imply zero cases. We can all agree upon the fact that zero cases with no testing at all does not really mean that a given country is not affected by the pandemic. We therefore need a quantity unrelated to the rate of testing.

This quantity is the number of deaths: this value is unbiased by the testing rate. We will use the normalized number of daily deaths for comparing countries.

Before moving on, it is important to clarify that when we deal with COVID-19 deaths we refer to number of people who died in the COVID-19 time period (starting from January 2020 till today). It is important to point out that on the death certificate of these individuals there might be no reference at all to the COVID-19 pandemic. This is because, as a result of different policies in different countries worldwide, it is hard to obtain reliable datasets with daily deaths that differentiate between those labeled COVID-19 and those completely related to other causes. Additionally, sometimes there is no certainty about whether COVID-19 did or did not play a role in the death of an individual.

While the previous results were obtained using the original COVID-19 historical dataset[2], **this time some preprocessing and resampling is needed:**

- the daily values for some of the dates in the original dataset are missing:
  - countries with too many missing values (more than 150 days) were removed since no interpolation technique can really be effective in such cases;
  - where few missing values were present for the initial months of the pandemic, these were replaced using the constant value 0;

- missing and negative values in the middle of the time series were replaced using the moving window average;
- plotting the data with daily granularity results in a time series plot which is not smooth and hard to read; the data was therefore resampled with weekly granularity;

To impute the missing values, i.e., to infer them from the known part of the data, an univariate imputation algorithm was used: imputes values in the  $i$ -th feature dimension using only non-missing values in that feature dimension. To this end, the `SimpleImputer` class from the `scikit-learn` Python package and the `rolling` method from the `pandas.DataFrame` package were used.

This is the resulting time series plot for some of the countries:

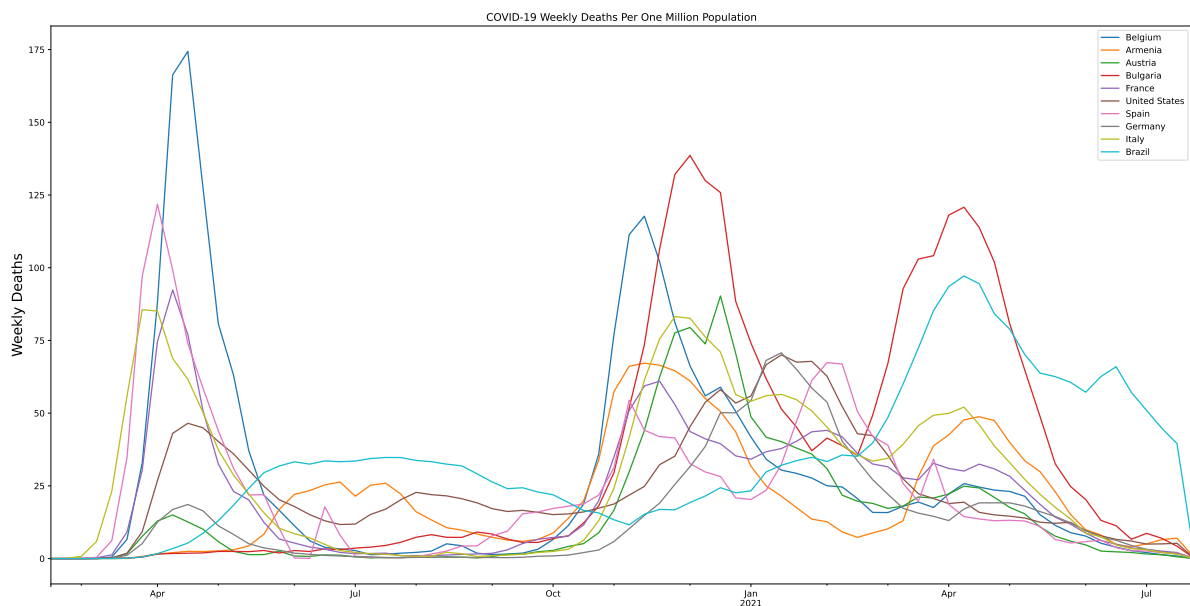


Figure 3: COVID-19 Weekly Deaths Per One Million Population

Daily deaths count appears to be a fair measure to compare how hard different countries have been hit by the virus. In addition, observing the plot one can immediately point out that

- some countries suffered more the first wave, some others the second and some others the third wave; some countries suffered all of them;
- countries such as [Italy](#) and [Belgium](#) were devastated by the first wave and reacted by completely shutting down social life; yet they did not do better with the second wave; finally, the vaccination campaign seems to have helped them control the pandemic (both in [Italy](#) and in [Belgium](#) almost 50% of the population has received the first vaccine dose);
- countries such as [Austria](#) and Germany have not been hit hard by the first COVID-19 wave; these states reacted fast and slowed down social life right at the beginning; as a result, the number of deaths went back to almost zero; yet they suffered from the second wave and got the situation under control using vaccines;



- countries such as **United States** and **Brazil** had a daily death toll with an almost constant trend during the first and second waves; while the situation has drastically improved in the **United States** thanks to the vaccines (50% of the population has received the first vaccine dose), in **Brazil** the number of daily deaths are increasing (16% of the population has received the first vaccine dose);

If we extend the same reasoning to all the countries, figure 4 shows the weekly deaths per one million population for 200 countries, from Afghanistan to Zimbabwe. It is humanly impossible to extract any useful information from such a plot. **However, in this mess, how many different characteristic curves can we find?**

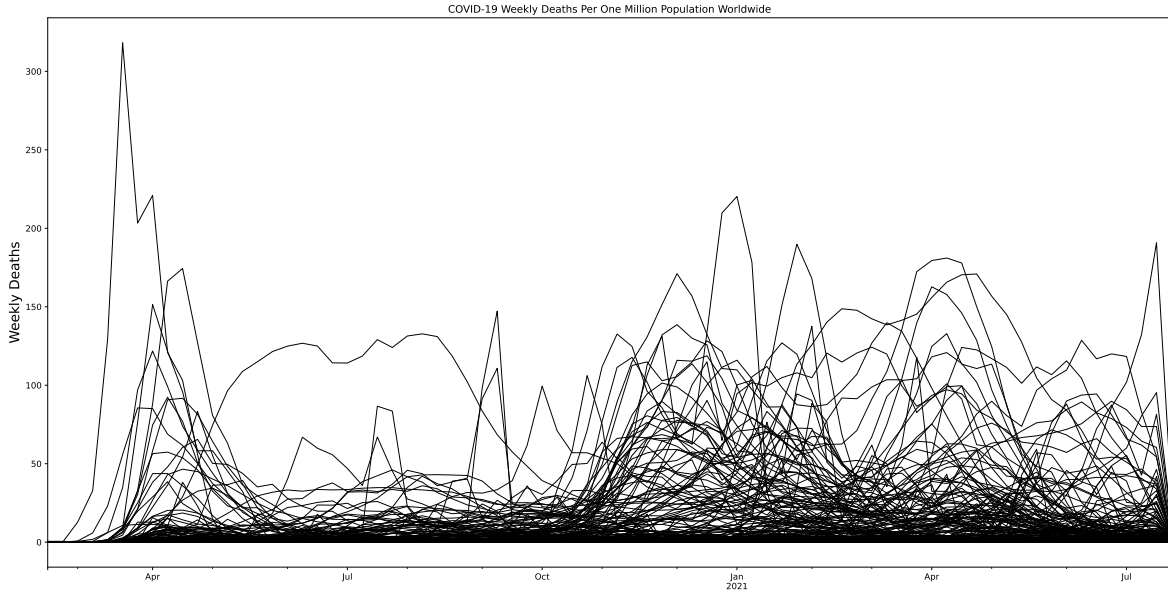


Figure 4: COVID-19 Weekly Deaths Per One Million Population Worldwide

To clean the mess, find patterns and extract the required knowledge, a clustering algorithm was used. This unsupervised learning technique groups similar data curves together.

### 3.1.1 Clustering time series data

When taking into account time series clustering algorithm, there have been several measures applied. For example there is probability-based distance, that takes into account the seasonality of data, Hausdorff distance defined as "the maximum of the distances from a point in any of the sets to the nearest point in the other set" (Rote (1991)) or Hidden Markov model based distance used in modelling complex time series (Zeng, Duan, and Wu (2010)). The most popular however is the Euclidean distance and Dynamic Time Warping distance. It has been proven that the Euclidean distance is the most efficient, but forces both time series to be the same length. The DTW method is however known to be the most accurate.

#### Euclidean vs. Dynamic Time Warping

The main difference between both distances can be best understood graphically. The picture below shows an example of matched points of two data vectors. They were connected based on the minimal distance between points based on DTW (black) and Euclidean (red) distance. It can be seen that with DTW the 11th blue point matches 4 green points. When taking into account the Euclidean distance it can be seen that the assigned points 9-to-9

and 10-to-10 are visibly further than 9-to-11 and 10-to-11. That can significantly impact the overall distance of series between each other. The DTW takes into account the shape of both time series much better. Dynamic Time warping is a method of calculating distance that is more accurate than Euclidean distance. It has an advantage over Euclidean if data points are shifted between each other and we want to look rather at its shape.

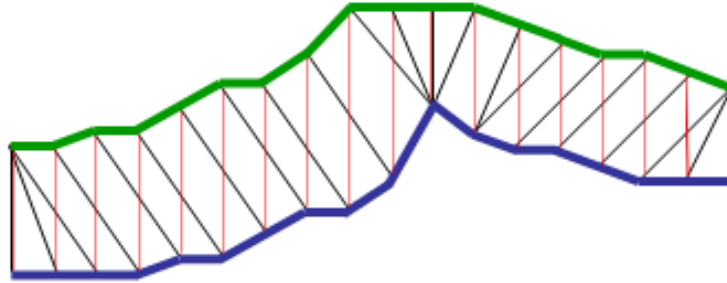


Figure 5: Visual comparison of matched points based on DTW (black) and Euclidean (red) distance

Additionally, the two time series do not have to be equal in length as required by the Euclidean distance. The Euclidean distance takes pairs of data points and compares them to each other. DTW calculates the smallest distance between all points — this enables a one-to-many match.

Two K-means clustering models were built: one using the euclidean distance and one using the DTW algorithm. To this end, `tslearn`, a Python package that provides machine learning tools for the analysis of time series, was used. This package builds on (and hence depends on) the `scikit-learn`, `numpy` and `scipy` libraries.

First of all, the optimal number of clusters was evaluated using the mean silhouette coefficient for all samples. The different silhouette values obtained for the K-means Model are listed below:

- 2 Clusters silhouette:
  - Euclidean distance: 0.606 — Dynamic Time Warping: 0.689
- 3 Clusters silhouette:
  - Euclidean distance: 0.585 — Dynamic Time Warping: 0.627
- 4 Clusters silhouette:
  - Euclidean distance: 0.582 — Dynamic Time Warping: 0.620
- 5 Clusters silhouette:
  - Euclidean distance: 0.597 — Dynamic Time Warping: 0.619

The best value for the mean silhouette coefficient is obtained with only 2 clusters. However, taking into account our knowledge of the phenomenon, we expect to find 3 clusters of countries:

1. those hardly hit by both the first and the second wave, that implemented a proper vaccination campaign;
2. those that have been hit by either the first or the second wave, that implemented a proper vaccination campaign;
3. and those suffering COVID-19 deaths constantly, that have not the resources to implement an adequate vaccination campaign.

Two K-means models with 3 clusters were therefore evaluated: one trained using the euclidean distance and the other using the dynamic time warping algorithm.

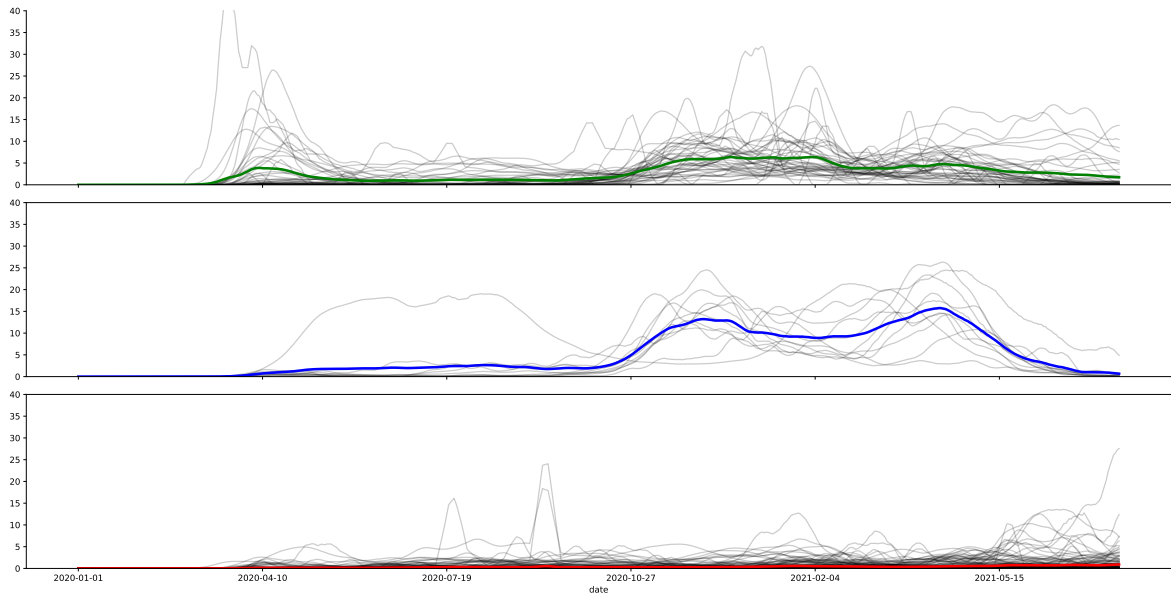


Figure 6: Time series K-means Clustering Model using Euclidean distance

Figure 6 shows the clusters obtained by applying the K-means Clustering algorithm provided by the `TimeSeriesKMeans` class of the `tslearn` Python package. In this model, the metric chosen to be used for both cluster assignment and barycenter computation is the euclidean distance:

- the **green cluster** groups 167 countries with a low number of daily deaths (close to zero);
- the **blue cluster** groups 11 countries which suffered either the first or the second SARS-CoV-2 wave;
- the **red cluster** groups 44 countries with a huge number of daily deaths, that managed to get the situation under control for a while after the first wave, but suffered for the second wave as well;
- mean silhouette coefficient for all samples: 0.58.

The time series K-means clustering model using the Dynamic Time Warping algorithm provides different clusters. As shown in figure 7,

- the **green cluster** groups 140 countries;

- the **blue cluster** groups 28 countries;
- the **red cluster** groups 54 countries;
- mean silhouette coefficient for all samples: 0.62.

We should keep in mind that the DTW algorithm is quite CPU-intensive: as a matter of fact, with 50 iterations, the same number used to obtain the model with the euclidean distance, it takes almost 2 minutes<sup>5</sup> to compute the clusters shown in figure 7.

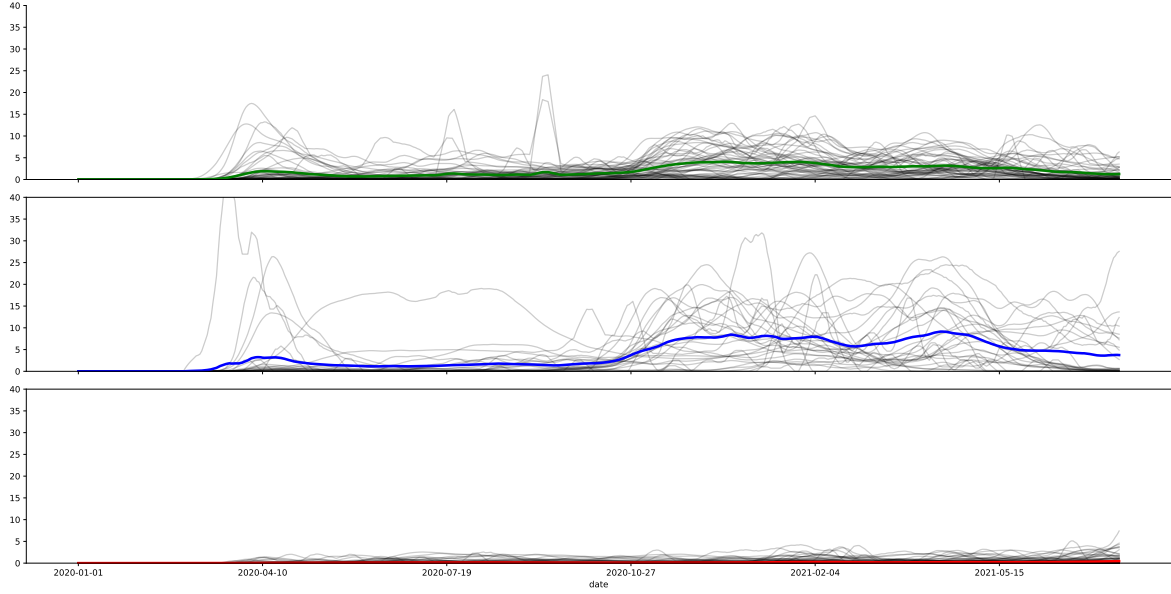


Figure 7: Time series K-means Clustering Model using Dynamic Time Warping algorithm

Simply looking at the characteristic curves of each of the clusters identified in the two proposed models, it looks like that the DTW-based K-means clustering performed a better grouping.

Keeping in mind that

1. the tools available in the **tslearn** Python package for working with time series clustering are quite limited;
2. as a result the silhouette coefficient was used at the beginning in absence of other more robust clustering tendency statistical indicator;
3. there is no real ground truth when dealing with clustering problems in general;

to compare the two models, I represented the clusters obtained by the two models on the world map. Surprisingly enough, as shown in figure 8 and in figure 9, those groups form local clusters on the world map as well. The nature of the phenomenon we are analyzing, the spread of the SARS-CoV-2 pandemic, tells us that how well this local clusters are grouped in the world map, in terms of neighboring countries can be used as a measure of how well the algorithm performed:

---

<sup>5</sup>Please take this statistic as a very general one — it is relative to an Intel Core i7-7700K Processor (4 Cores 8 Threads @ 4.20 GHz).

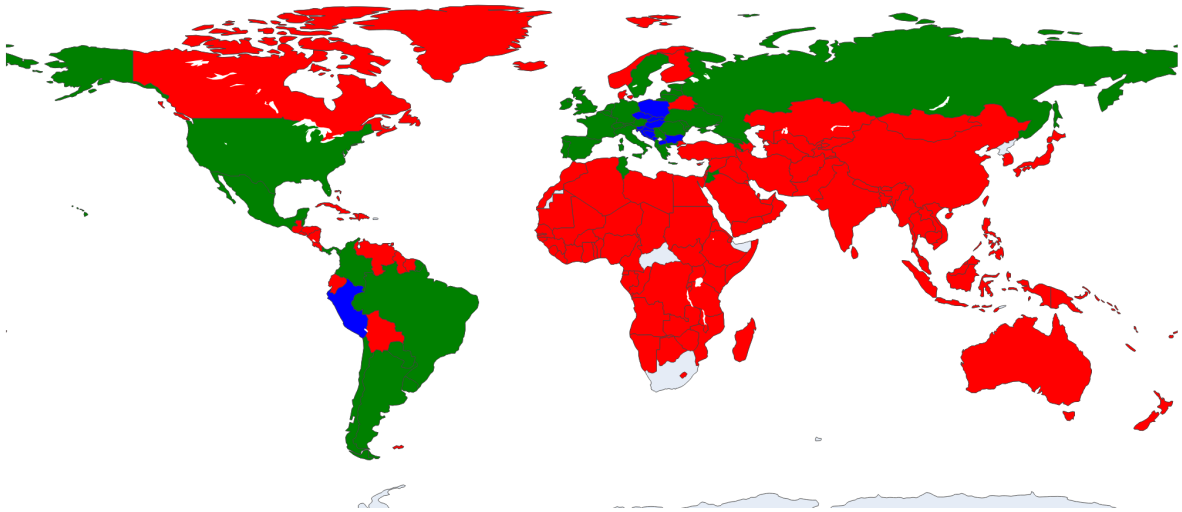


Figure 8: K-means Clustering Model World Map (Euclidean Distance)

Given the nature of the phenomenon under analysis, it is very hard to say which is better. One important observation to be made is that changing from the euclidean distance to the DTW algorithm, Canada moves from the **green cluster** to the **red cluster**. Also Russia moved from the **green cluster** to the **blue cluster**.

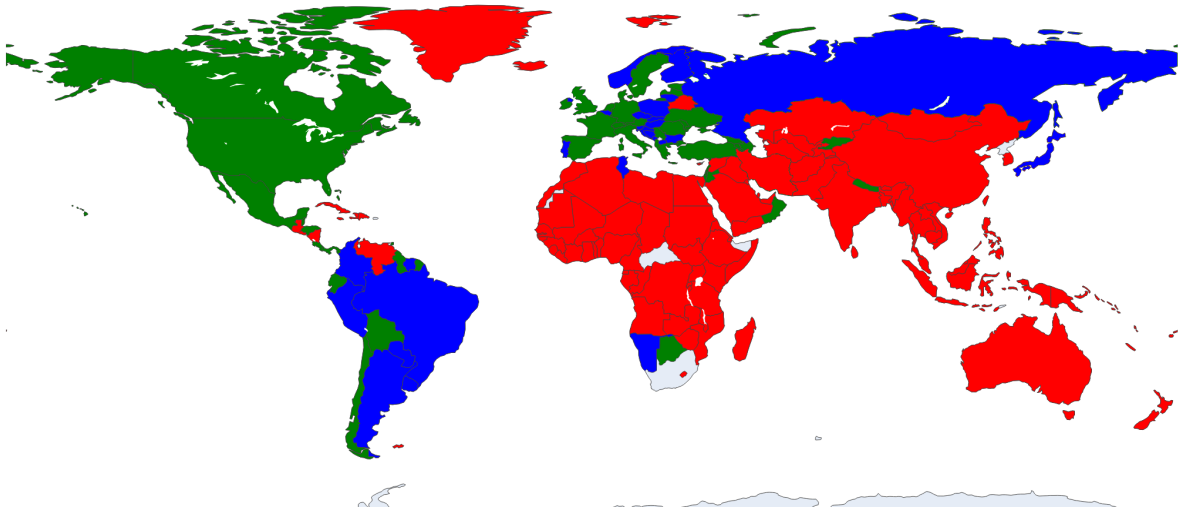


Figure 9: K-means Clustering Model World Map (DTW Algorithm)

- The **green cluster** (countries) corresponds to a two-wave profile, with the first peak at around Day 140, and values reaching 7 cases per 100,000 of the population. The second wave started around Day 240 and seems not to have peaked yet at Day 250. These are also countries which started an adequate vaccination campaign in order to prevent a third wave: on average, almost 60% of the population has received the second dose of the COVID-19 vaccine;
- The **blue cluster** groups countries which had a daily death toll with an almost constant trend during the first and second waves and did not succeed in running reactive vac-

cination campaign: on average, almost 20% of the population has received the second dose of the COVID-19 vaccine;

- The **red cluster** groups African and Middle East countries for which concern grows over third wave of infections. The number of deaths recorded for the past week has risen by more than 40% compared with the previous week, according to the World Health Organization (WHO): on average, less than 10% of the population has received the second dose of the COVID-19 vaccine.

Also, some **outliers** can be identified with reference to this map:

- Lebanon, Oman, Nepal, Kyrgyzstan, Tunisia, Botswana and Namibia which are classified as **green countries** but should be part of the **red countries**: this is highly related to the fact that the data provided by such countries might not be reliable;
- Israel which is the country with the one of the highest percentages of vaccinated citizens and is classified as a **red country**.

In conclusion, I believe that it is not easy to model a complex phenomenon such as the COVID-19 just by using 3 clusters based on the count of weekly deaths per one million population.

We also have to take into account that the clustering results are also influenced by vaccinations campaigns. Such campaigns (where actually active) are affected by geopolitics as well: politics and international relations play a major role.

### 3.2 Personalized predictive models for symptomatic COVID-19 patients

Currently, the world is facing a health and economic crisis due to the spread of the virus SARS-CoV-2 which causes a disease referred to as COVID-19. By the end of April 2020, the virus has spread to over 191 million people worldwide and has killed over 4,1 million. During this pandemic, governments and hospitals have struggled to allocate scarce resources, including tests, treatment in intensive care units (ICUs) and ventilators.

The rapid global spread of the virus SARS-CoV-2 has provoked a spike in demand for hospital care. Hospital systems across the world have been over-extended, including in Northern Italy, Ecuador, and New York City, and many other systems face similar challenges. As a result, decisions on how to best allocate very limited medical resources have come to the forefront. Specifically, under consideration are decisions on who to test, who to admit into hospitals, who to treat in an Intensive Care Unit (ICU), and who to support with a ventilator. Given today's ability to gather, share, analyze and process data, personalized predictive models based on demographics and information regarding prior conditions can be used to

1. help decision-makers allocate limited resources;
2. advise individuals how to better protect themselves given their risk profile;
3. using risk profiles to inform decisions on who should be tested (for the virus and/or antibodies) and at which frequency;
4. providing more accurate estimates of who is more likely to be hospitalized and the type of care they may need;
5. informing plans for staffing, resources, and prioritizing the level of care in extremely resource-constrained settings.

6. to develop personalized models that predict the following events:

- (a) hospitalization;
- (b) mortality;
- (c) need for ICU
- (d) need for a ventilator.

Equally importantly, as societies adapt to the pandemic, predictive models can

- 1. assess individual risk so that social distancing measures can transition from "blanket" to more targeted (e.g., deciding who can return to work, who is advised to stay at home, who should be tested, etc.);
- 2. direct policy decisions on who should receive priority for vaccination, which will be critical as initial vaccine production may not suffice to vaccinate everybody.

Using the FPGrowth (Frequent Pattern Growth) algorithm<sup>6</sup> allowed to mine the association rules listed in the following table ordered by support value:

Antecedents	Consequents	Support	Confidence	Lift	Kulczynski	IR
Pneumonia	COVID-19, Hospitalization	0.116372	0.859268	4.797504	0.0	0.0
Pneumonia	COVID-19, Hospitalization	0.116372	0.859268	4.797504	0.0	0.0

---

<sup>6</sup>The `frequent_patterns` module of the `mlxtend` Python package was used to this end.

## 4 Conclusion



## 5 Software Architecture

The subsequent analysis results were implemented as functionalities of a small Python3 software:

```
$ python3 main.py
Welcome to the COVID-19 Toolbox.

Type help or ? to list commands.

> ?

Documented commands (type help <topic>):
```

<b>exit</b>	plot_weekly_deaths_per_million
<b>help</b>	print_historical_ds_info
personalized_predictive_models	print_preconditions_ds_info
plot_total_cases	timeseries_clustering_dtw
plot_total_cases_per_million	timeseries_clustering_euclidean
plot_weekly_deaths_all_countries	update_historical_ds

The following is a brief description of each of the available commands:

- **update\_historical\_ds**: updates COVID-19 historical dataset to the latest available version;
- **print\_historical\_ds\_info**: prints COVID-19 historical dataset info;
- **print\_preconditions\_ds\_info**: prints COVID-19 preconditions dataset info;
- **plot\_total\_cases**: plot COVID-19 total confirmed cases histogram of the TOP 15 countries;
- **plot\_total\_cases\_per\_million**: plot COVID-19 total confirmed cases per one million population histogram of the TOP 15 countries;
- **plot\_weekly\_deaths\_per\_million**: plot COVID-19 weekly deaths per one million population time series of random countries;
- **plot\_weekly\_deaths\_all\_countries**: plot COVID-19 weekly deaths per one million population time series of all countries;
- **timeseries\_clustering\_euclidean**: plot COVID-19 daily deaths per one million population time series euclidean-based K-means clusters;
- **timeseries\_clustering\_dtw**: plot COVID-19 daily deaths per one million population time series DTW-based K-means clusters;
- **personalized\_predictive\_models**: build personalized predictive models for symptomatic COVID-19 patients using medical preconditions;
- **exit**: exit COVID-19 Toolbox.

## 5.1 Class Diagram

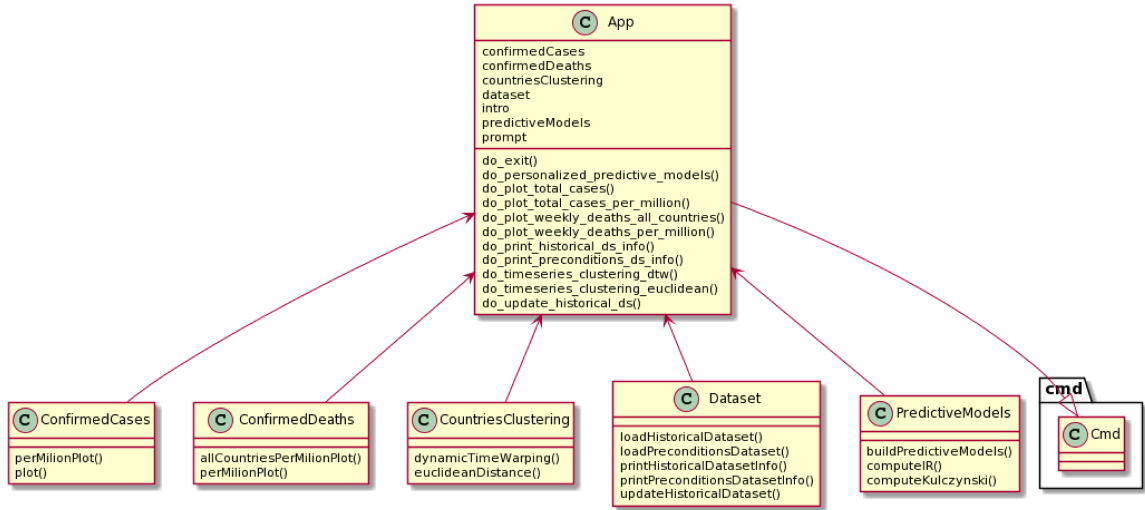


Figure 10: Python Class Diagram

The following is a brief description of each of the classes:

- **App**: application entry point; it implements the command line menu logic using an instance of the `Cmd` Python3 class;
- **Cmd**: this Python3 class provides a simple framework for writing line-oriented command interpreters. These are often useful for test harnesses, administrative tools, and prototypes that will later be wrapped in a more sophisticated interface;
- **ConfirmedCases**: implements all the functions related to plotting Confirmed COVID-19 cases;
- **ConfirmedDeaths**: implements all the functions related to plotting Confirmed COVID-19 cases;
- **CountriesClustering**: implements all the functions related to computing and plotting countries K-means clustering based on COVID-19 confirmed deaths;
- **Dataset**: implements all the functions related to loading, preprocessing, saving and printing the content of the different datasets;
- **PredictiveModels**: implements all the functions related to loading, preprocessing, saving and printing the content of the different datasets.

## 5.2 Python Packages

The following Python3 packages and libraries were used in the development:

- `matplotlib.pyplot`: is a state-based interface to `matplotlib`; it provides a MATLAB-like way of plotting; mainly intended for interactive plots and simple cases of programmatic plot generation;

- **pandas**: is a fast, powerful, flexible and easy to use open source data analysis and manipulation tool, built on top of the Python programming language;
- **numpy**: is the fundamental package for scientific computing in Python; it is a Python library that provides a multidimensional array object, various derived objects (such as masked arrays and matrices), and an assortment of routines for fast operations on arrays, including mathematical, logical, basic statistical operations, random simulation and much more;
- **plotly**: is a high-level, declarative charting library;
- **plotly.graph\_objs**: the figures created, manipulated and rendered by the **plotly** Python library are represented by tree-like data structures which are automatically serialized to JSON for rendering by the **Plotly.js** JavaScript library.
- **sklearn**: Scikit-learn (formerly **scikits.learn**) is a free software machine learning library for the Python programming language;
- **tslearn**: is a Python package that provides machine learning tools for the analysis of time series;
- **tslearn.utils**: is a which module includes various time series conversion utilities;
- **tslearn.clustering**: K-means clustering for time-series data;
- **mlxtend**: Mlxtend (machine learning extensions) is a Python library of useful tools for the day-to-day data science tasks;
- **mlxtend.frequent\_patterns**: this module contains the Apriori and FPGrowth algorithm implementations.

### 5.3 Kulczynski measure and Imbalance Ratio

The **frequent\_patterns** module of the **mlxtend** Python package used to mine the association rules, provides the following scoring metrics: "antecedent support", "consequent support", "support", "confidence", "lift", "leverage" and "conviction".

The Kulczynski measure was computed using the provided metrics according to the following reasoning:

$$\begin{aligned}
 Kulc(A, B) &= \frac{1}{2} (P(A|B) + P(B|A)) = \\
 &= \frac{1}{2} \left( \frac{support(B \cup A)}{support(B)} + \frac{support(A \cup B)}{support(A)} \right) = \\
 &= \frac{1}{2} \left( \frac{"support"}{"consequent support"} + \frac{"support"}{"antecedent support"} \right).
 \end{aligned}$$

The Imbalance Ratio was computed using the provided metrics according to the following reasoning:

$$\begin{aligned}
 IR(A, B) &= \frac{|support(A) - support(B)|}{support(A) - support(B) + support(A \cup B)} = \\
 &= \frac{|"antecedent support" - "consequent support"|}{"antecedent support" - "consequent support" + "support"}.
 \end{aligned}$$

## References

- [1] World Health Organization. *Archived: WHO Timeline - COVID-19*. 2021. URL: <https://www.who.int/news/item/27-04-2020-who-timeline---covid-19>.
- [2] Our World in Data. *Coronavirus Pandemic (COVID-19) – the data*. 2021. URL: <https://ourworldindata.org/coronavirus-data>.
- [3] University Of Cambridge. *Cambridge Conversations: COVID-19 behind the numbers*. 2020. URL: <https://www.alumni.cam.ac.uk/events/cambridge-conversations-events-in-2020/cambridge-conversations-covid-19-behind-the-numbers>.
- [4] Towards Data Science - Robert Biele. *COVID-19: What Is Hidden Behind the Official Numbers?* 2020. URL: <https://towardsdatascience.com/which-countries-are-affected-the-most-by-covid-19-4d4570852e31>.
- [5] Ewa Sobolewska. *Dynamic Time Warping (DTW) as a mean to cluster time series*. 2019. URL: [https://rstudio-pubs-static.s3.amazonaws.com/474160\\_0761428e8683401e941a92a4177254.html](https://rstudio-pubs-static.s3.amazonaws.com/474160_0761428e8683401e941a92a4177254.html).
- [6] Ioannis Ch. Paschalidis Salomon Wollenstein-Betech Christos G. Cassandras. “Personalized Predictive Models for Symptomatic COVID-19 Patients Using Basic Preconditions”. In: *International Journal of Medical Informatics* 142 (2020). DOI: <https://doi.org/10.1016/j.ijmedinf.2020.104258>.
- [7] Gobierno de Mexico - Direccion General de Epidemiologia. *COVID-19 Datos Abiertos*. 2021. URL: <https://www.gob.mx/salud/documentos/datos-abiertos-152127>.