UNIVERSITÀ
DEGLI STUDI
FIRENZE

Data Mining and Organization – Covid–19

Armand Palla

Università degli Studi di Firenze

15 February 2022

UNIVERSITÀ DEGLI STUDI FIRENZE

Table of contents:

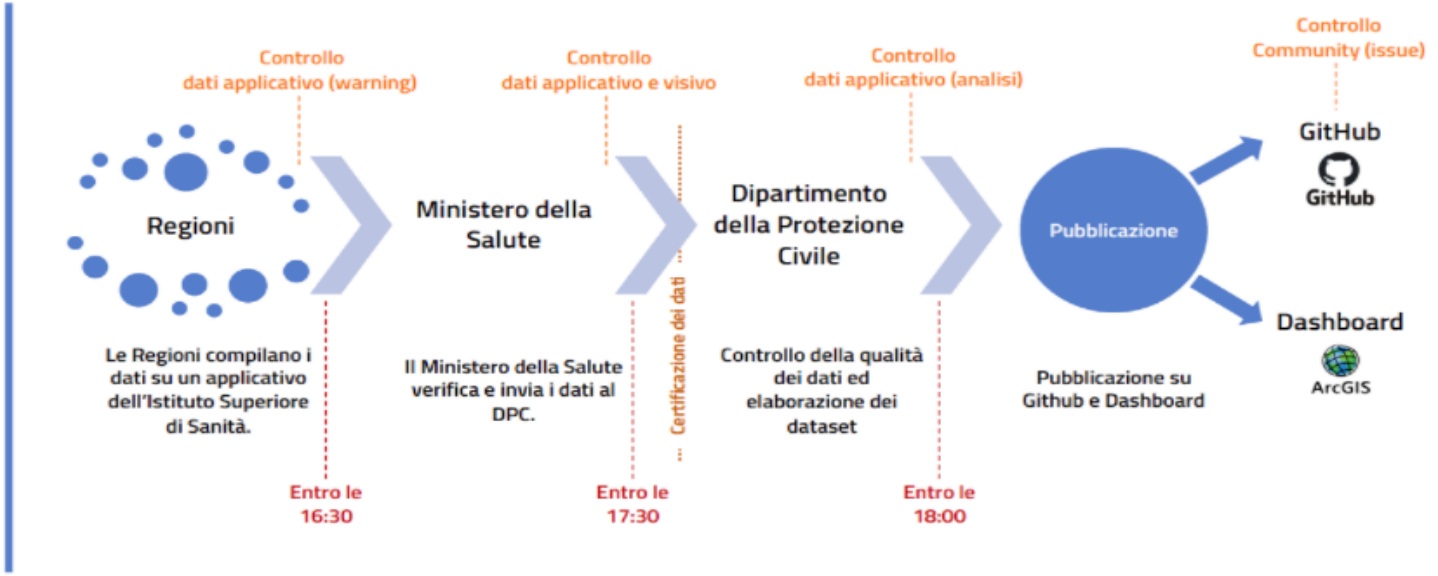
1. Understanding the data

- Civil Protection Department - Coronavirus emergency.
- The Italian Civil Protection Department receives daily data by the Italian Ministry of Health, analyzes them and updates the database.
- The database is freely accessible at github.com/pcm-dpc/COVID-19.
- This database contains data of integrated surveillance for the "Coronavirus disease 2019" in Italy.
- The additional value of these data relies on the real-time (daily update) integrated surveillance of COVID-19 in Italy and on their reliability due to their official source (Italian Civil Protection Department).

1. Understanding the data

- These data are useful because:
  1. they provide insight on the spread of SARS-CoV-2.
  2. to inform Italian and foreign citizens on the SARS-CoV-2 spread in Italy.
  3. to support organizations in the evaluation of the efficiency of current prevention and control measures.
  4. to support governments in the future prevention decisions

PROTEZIONE CIVILE
Presidenza del Consiglio dei Ministri
Dipartimento della Protezione Civile



1. Understanding the data

- The database consists of different folders such as: ‘aree’, ‘dati-andamentonazionale’, ‘dati-json’, ‘dati-province’, ‘dati-regioni’ ,‘schede-riepilogative etc, but we are using:

1. dpc-covid19-ita-andamento-nazionale.json
2. dpc-covid19-ita-regioni.json
3. dpc-covid19-ita-province.json

1. Understanding the data

1. dpc-covid19-ita-andamento-nazionale.json

The folder called ‘dati-andamento-nazionale’ contains data relating to the national trend of SARS-CoV-2 spread.

I have used **pandas.DataFrame** in order to elaborate with data in .json format.

Inside each file, data are structured in the 24 fields (one column per field).


```
[
  {
    "data":"2020-02-24T18:00:00",
    "stato":"ITA",
    "ricoverati_con_sintomi":101,
    "terapia_intensiva":26,
    "totale_ospedalizzati":127,
    "isolamento_domiciliare":94,
    "totale_positivi":221,
    "variazione_totale_positivi":0,
    "nuovi_positivi":221,
    "dimessi_guariti":1,
    "deceduti":7,
    "casi_da_sospetto_diagnostico":null,
    "casi_da_screening":null,
    "totale_casi":229,
    "tamponi":4324,
    "casi_testati":null,
    "note":null,
    "ingressi_terapia_intensiva":null,
    "note_test":null,
    "note_casi":null,
    "totale_positivi_test_molecolare":null,
    "totale_positivi_test_antigenico_rapido":null,
    "tamponi_test_molecolare":null,
    "tamponi_test_antigenico_rapido":null
  },
```

1. Understanding the data

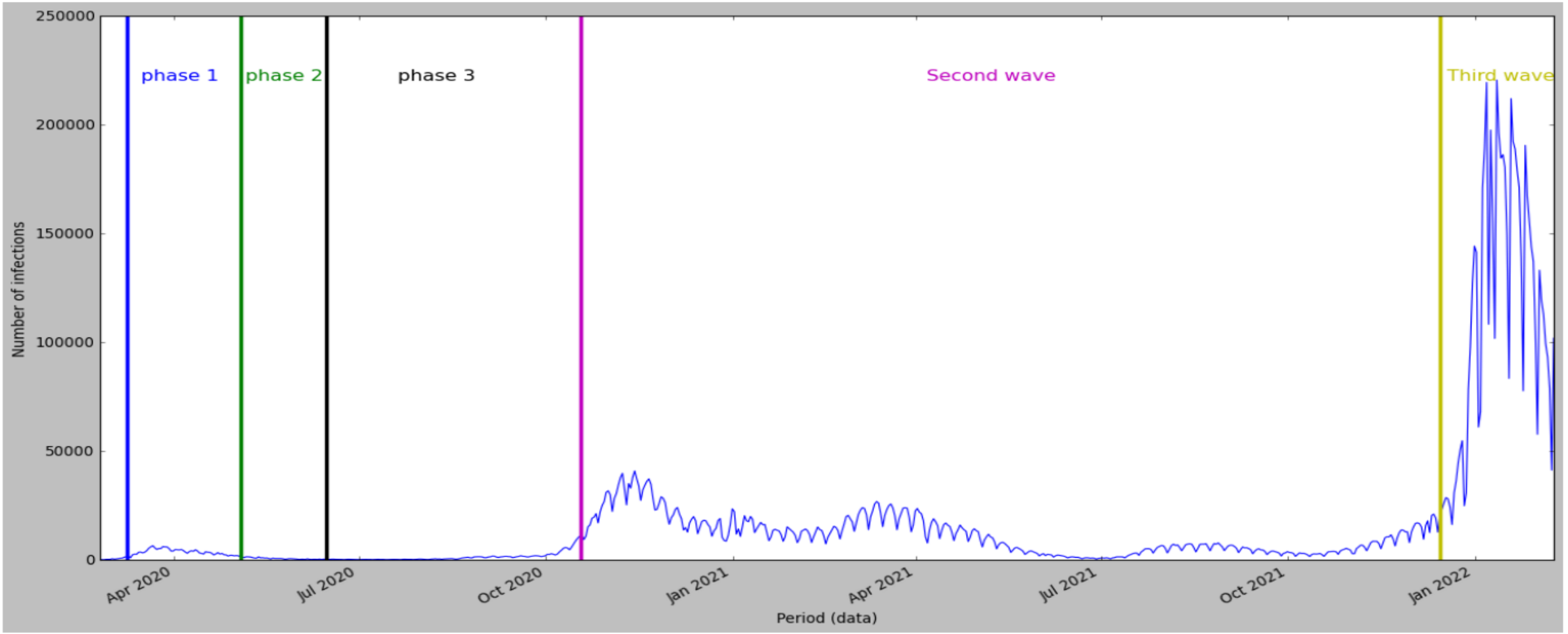
1. dpc-covid19-ita-andamento-nazionale.json

Then, we have done an analysis based on different **covid time intervals:**

- **Phase 1:** 09/03/2020 – 03/05/2020; Quarantine Period.
- **Phase 2:** 04/05/2020 - 14/06/2020 , Relaxation of containment measures.
- **Phase 3:** 15/06/2020 - 06/11/2020, living with COVID-19 .
- **Second wave:** 06/11/2020 - 14/12/2021, 06/11/2020-new DPCM that divides Italy into 3 zones(Yellow zone, Orange zone, Red zone).
- **Third wave:** 15/12/2021 - until now, the last period when we have seen an increase in positive cases.

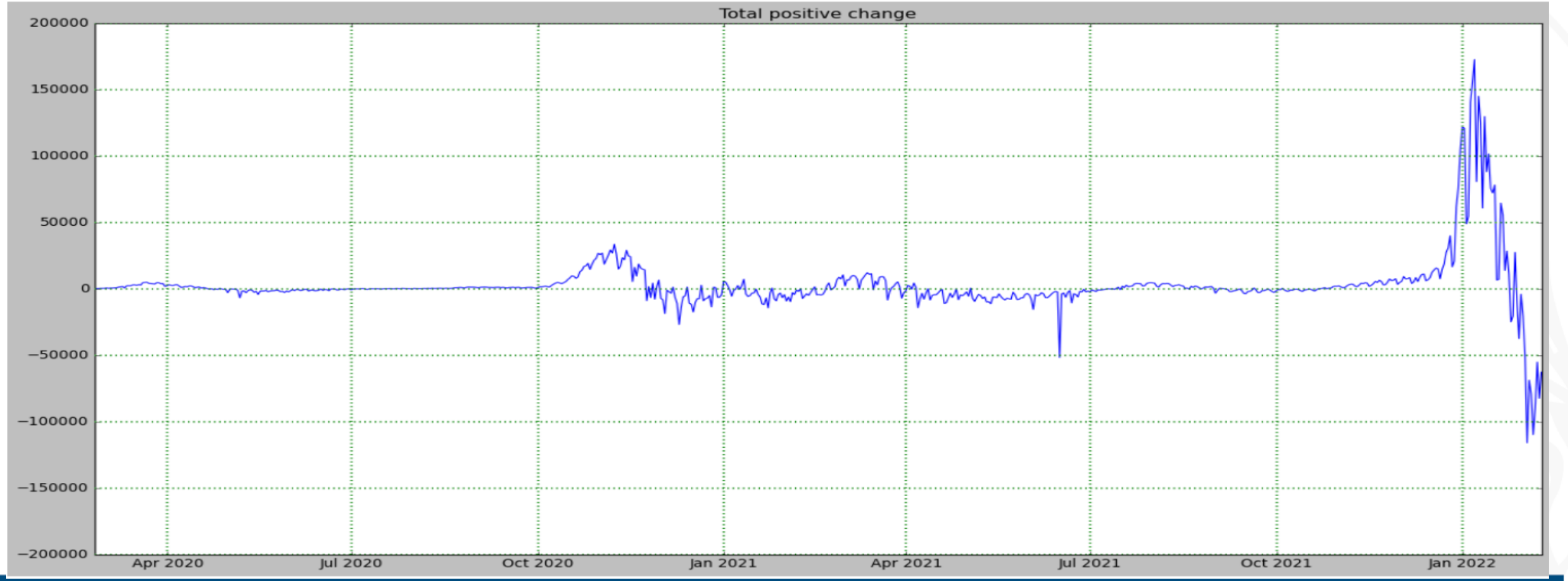
UNIVERSITÀ DEGLI STUDI FIRENZE

1. Understanding the data



1. Understanding the data

In another graph, we have represented the **total positives change** for every day from the initial until now and we can see that in the last week we have a decrease of positive cases.



2. Working with data

2. [dpc-covid19-ita-regioni.json]

The folder called ‘dati-regioni’ contains data relating to the regional trend of SARS-CoV-2 spread.

Inside each file, data are structured in the 24 fields (one column per field).


```
{
  "data":"2020-02-24T18:00:00",
  "stato":"ITA",
  "codice_regione":13,
  "denominazione_regione":"Abruzzo",
  "lat":42.35122196,
  "long":13.39843823,
  "ricoverati_con_sintomi":0,
  "terapia_intensiva":0,
  "totale_ospedalizzati":0,
  "isolamento_domiciliare":0,
  "totale_positivi":0,
  "variazione_totale_positivi":0,
  "nuovi_positivi":0,
  "dimessi_guariti":0,
  "deceduti":0,
  "casi_da_sospetto_diagnostico":null,
  "casi_da_screening":null,
  "totale_casi":0,
  "tamponi":5,
  "casi_testati":null,
  "note":null,
  "ingressi_terapia_intensiva":null,
  "note_test":null,
  "note_casi":null,
  "totale_positivi_test_molecolare":null,
  "totale_positivi_test_antigenico_rapido":null,
  "tamponi_test_molecolare":null,
  "tamponi_test_antigenico_rapido":null,
  "codice_nuts_1":null,
  "codice_nuts_2":null
},
```

2. Working with data

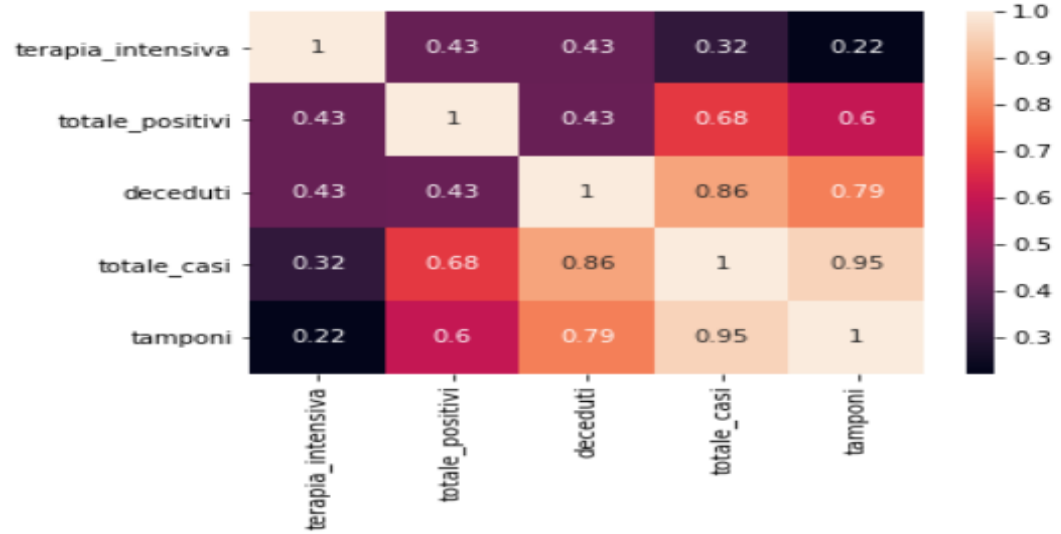
We have created another dictionary object called **covid**, in which we decided to add the fields that we think that are the most importants.

Then, we have plotted a heatmap using seaborn library

in order to represent the correlation between the choosen

variables.

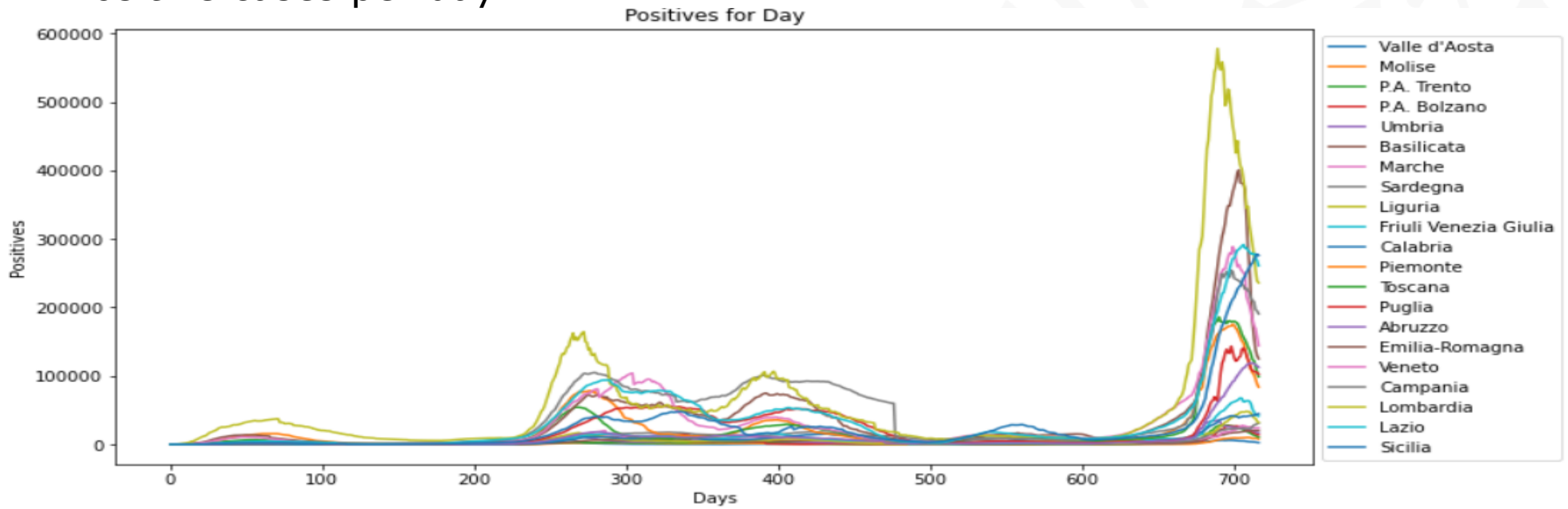
```
covid = {
    'date': json_data.data,
    'regioni': json_data.denominazione_regione,
    'terapia_intensiva': json_data.terapia_intensiva,
    'totale_positivi': json_data.totale_positivi,
    'deceduti': json_data.deceduti,
    'totale_casi': json_data.totale_casi,
    'tamponi': json_data.tamponi,
}
```



2. Working with data

After this, we started to plot some different graphs starting from:

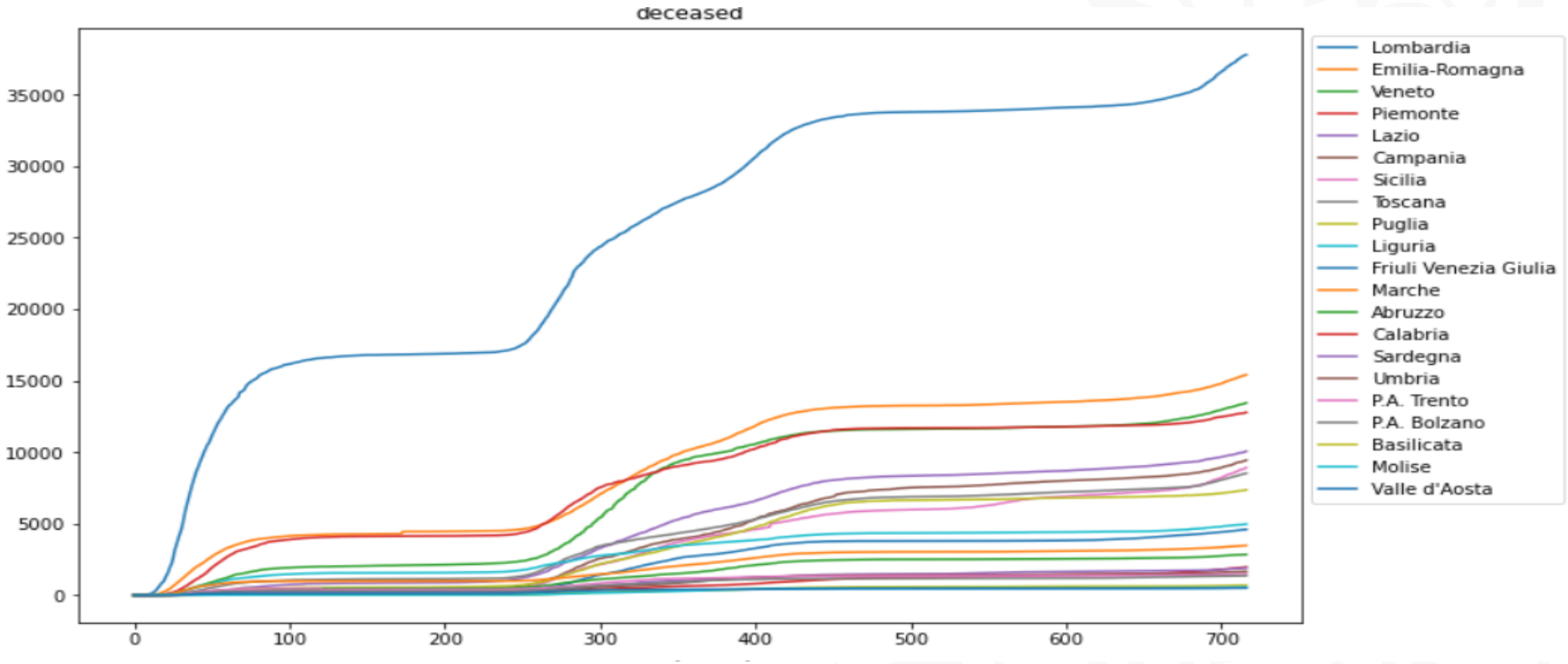
1. Positive cases per day



As we can see, the regions with the most number of positive cases are Lombardia, Emilia-Romagna and Veneto.

2. Working with data

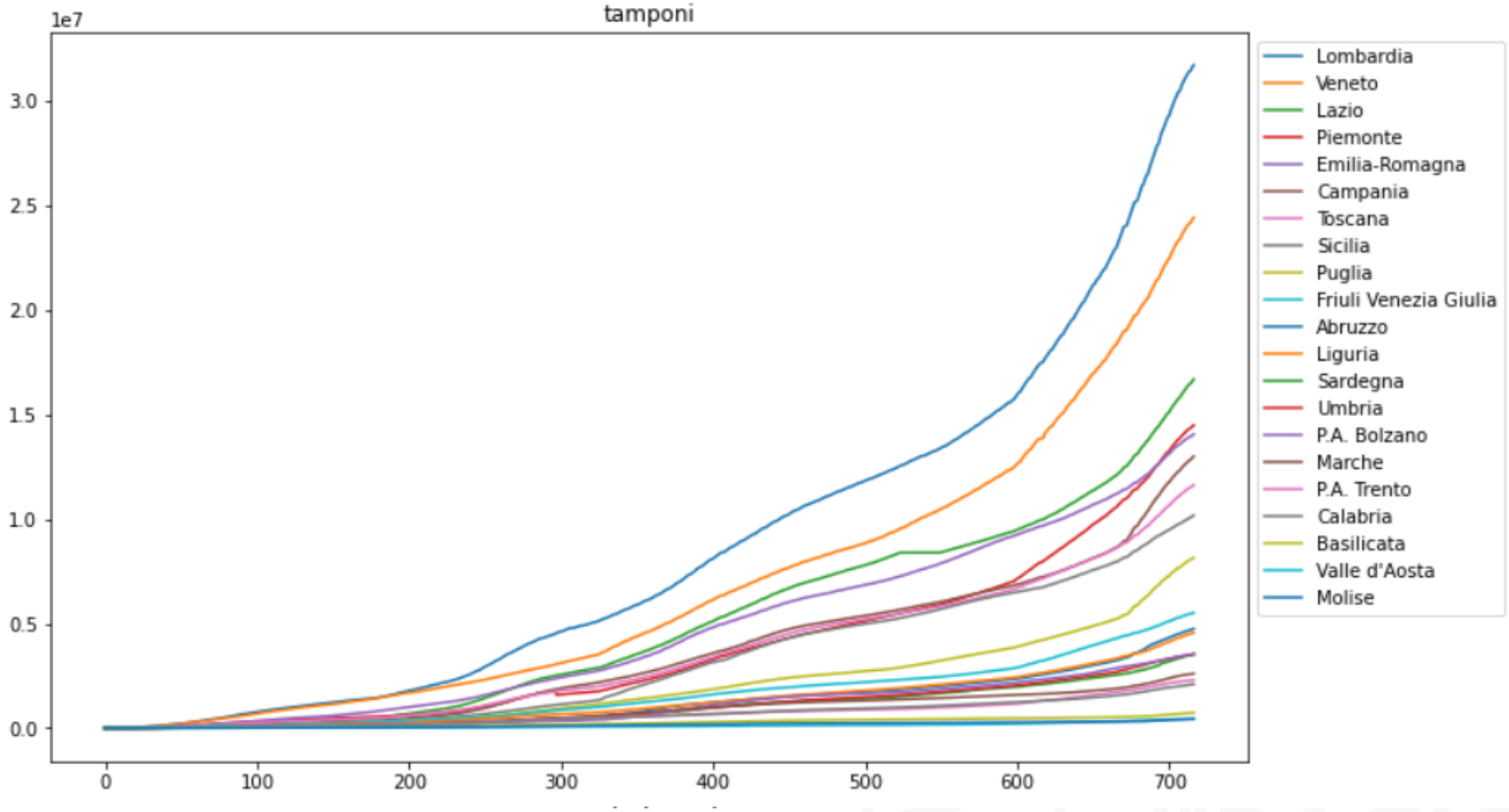
2. Number of deceased persons



UNIVERSITÀ DEGLI STUDI FIRENZE

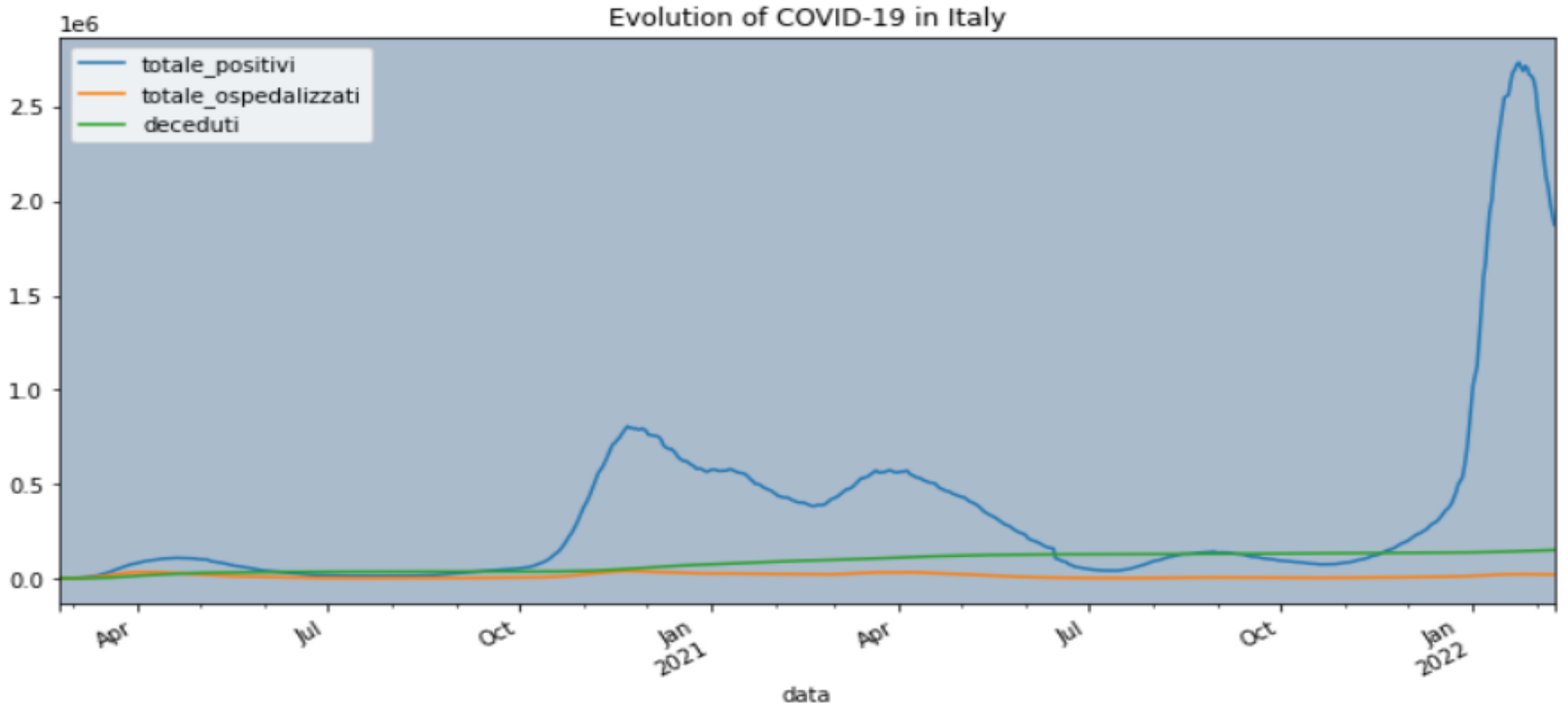
2. Working with data

3. Tampons for every regions



2. Working with data

4. Evolution of Covid-19, we make a comparison between (**totale_positivi, totale_ospedalizzati** e **deceduti**).



2. Working with data

3. dpc-covid19-ita-province.json

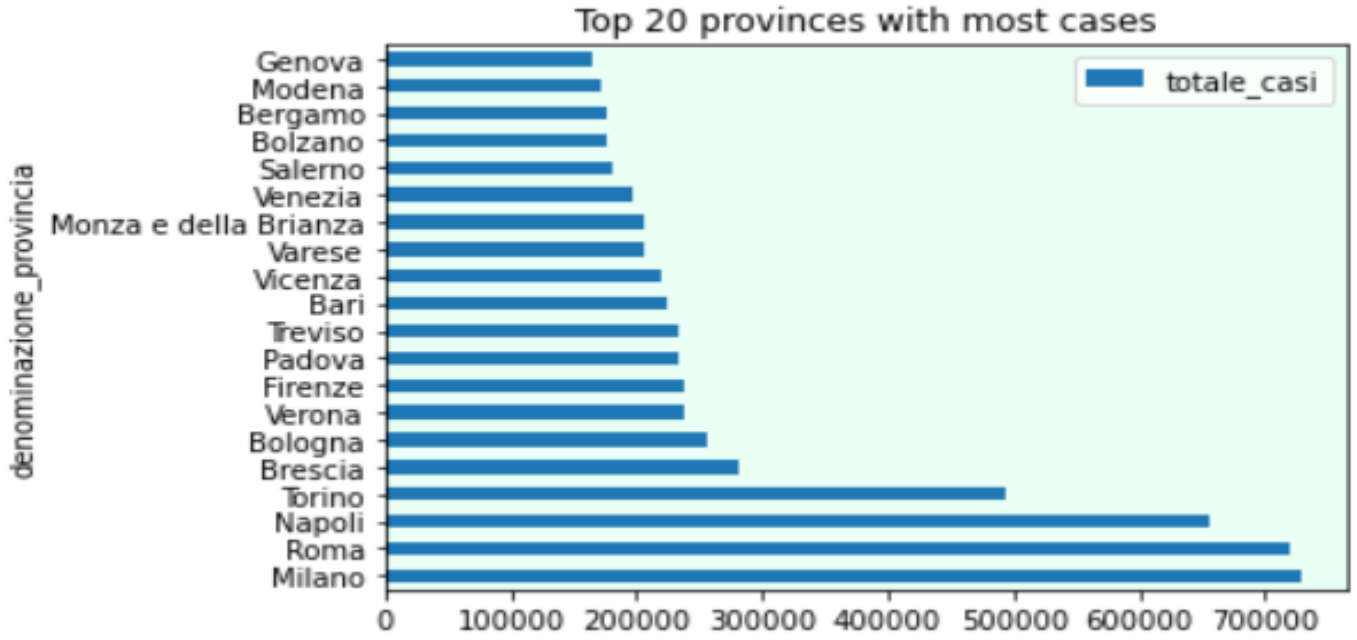
The folder called ‘dati-regioni’ contains data relating to the regional trend of SARS-CoV-2 spread.

Inside each file, data are structured in the 14 fields (one column per field).


```
{
  "data":"2020-02-24T18:00:00",
  "stato":"ITA",
  "codice_regione":13,
  "denominazione_regione":"Abruzzo",
  "codice_provincia":66,
  "denominazione_provincia":"L'Aquila",
  "sigla_provincia":"AQ",
  "lat":42.35122196,
  "long":13.39843823,
  "totale_casi":0,
  "note":null,
  "codice_nuts_1":null,
  "codice_nuts_2":null,
  "codice_nuts_3":null
},
```

2. Working with data

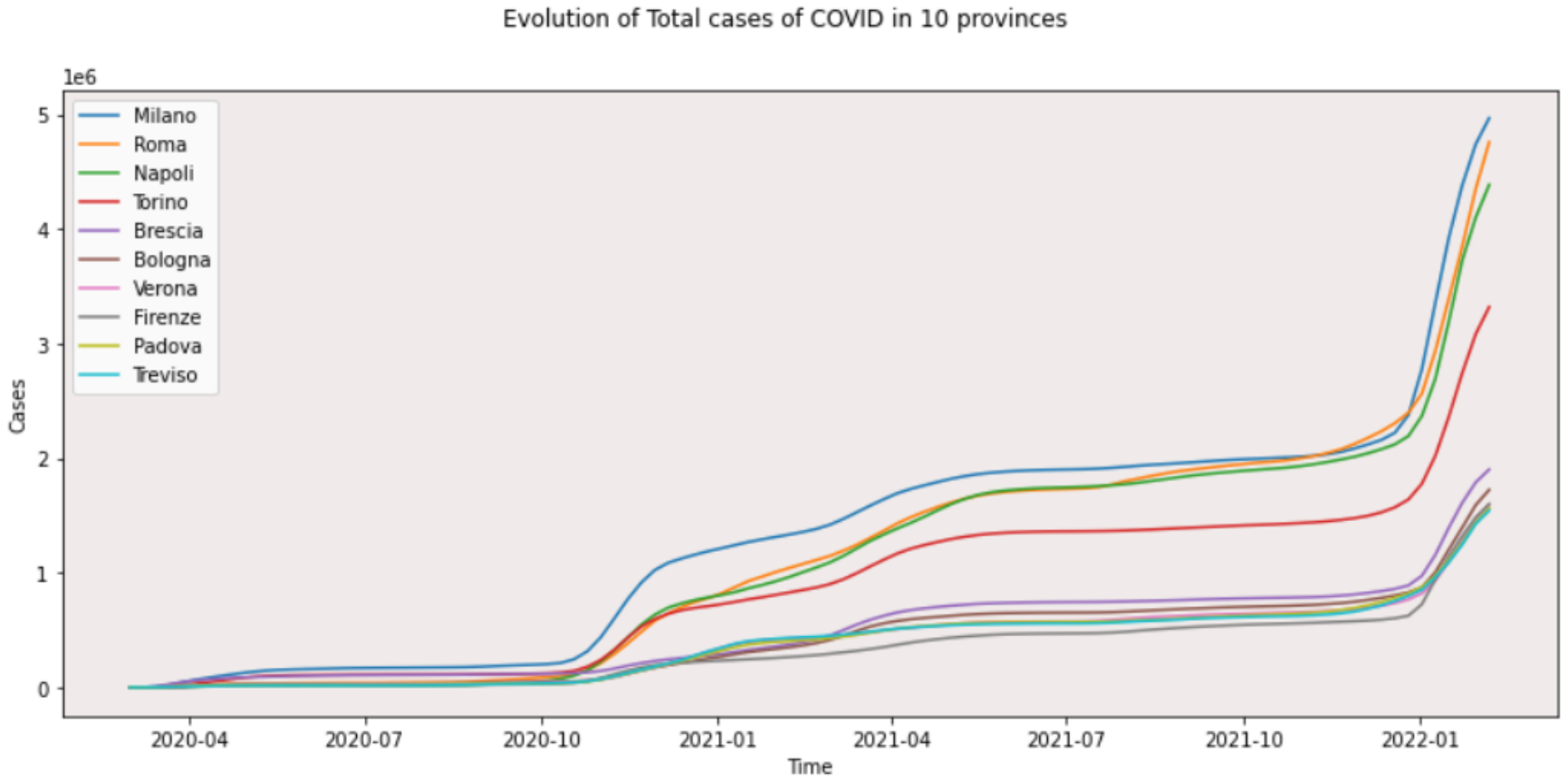
After this, we have displayed the Top 20 provinces with the most number of cases in an horizontal bar.



As, we can see from the results, the most infected provinces are Milano, Roma and Napoli.

2. Working with data

We have displayed the same results in a graph with 10 provinces.

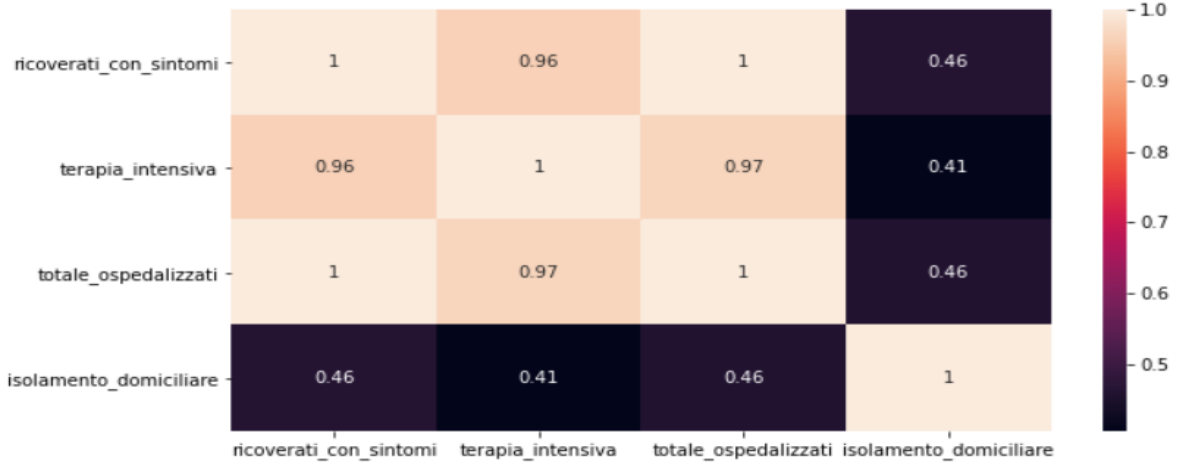


3. Clustering algorithms

We work again with dpc-covid19-ita-regioni.json dataset and we have taken in consideration these variables:

```
{'isolamento_domiciliare',
 'ricoverati_con_sintomi',
 'terapia_intensiva',
 'totale_ospedalizzati'}
```

Then, we have constructed a heatmap in order to control for NaN values and after this we have built the correlation matrix between the chosen variables.



3. Clustering algorithms

The algorithms that we have used in this project are:

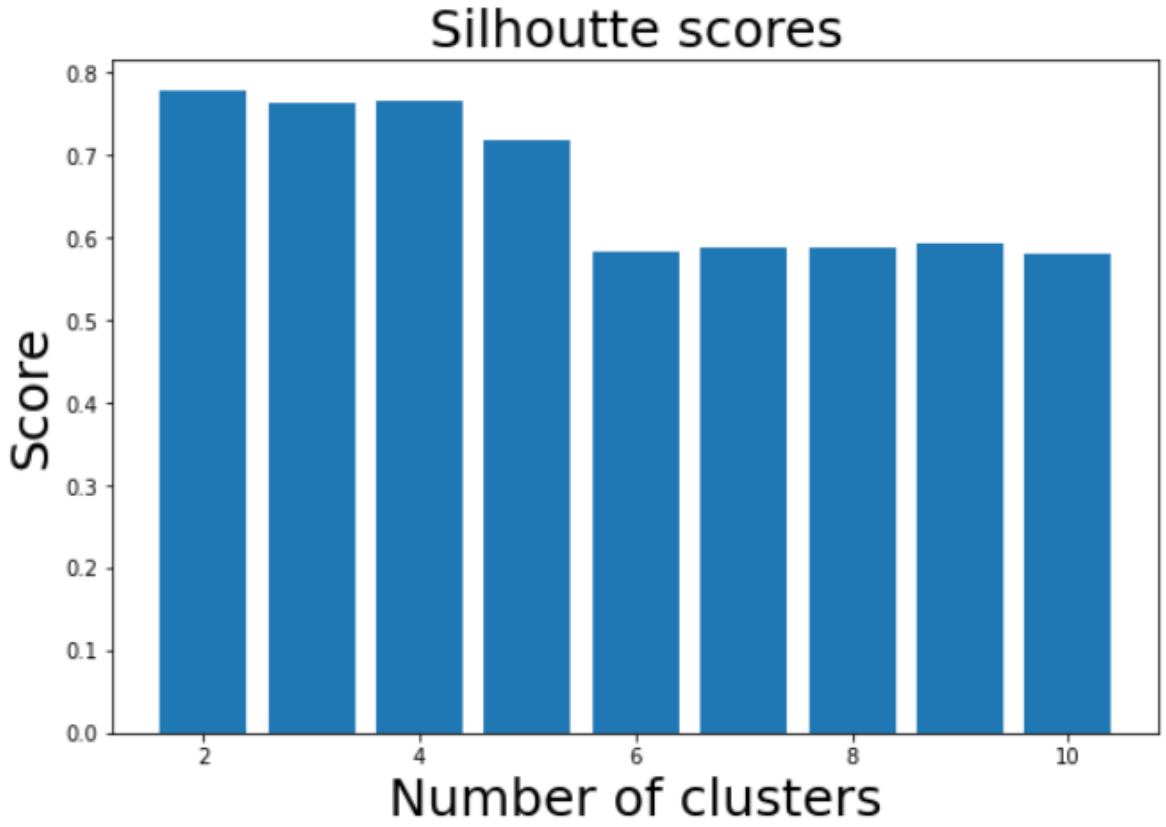
1. K-Means
2. Hierarchical Clustering
3. DBScan

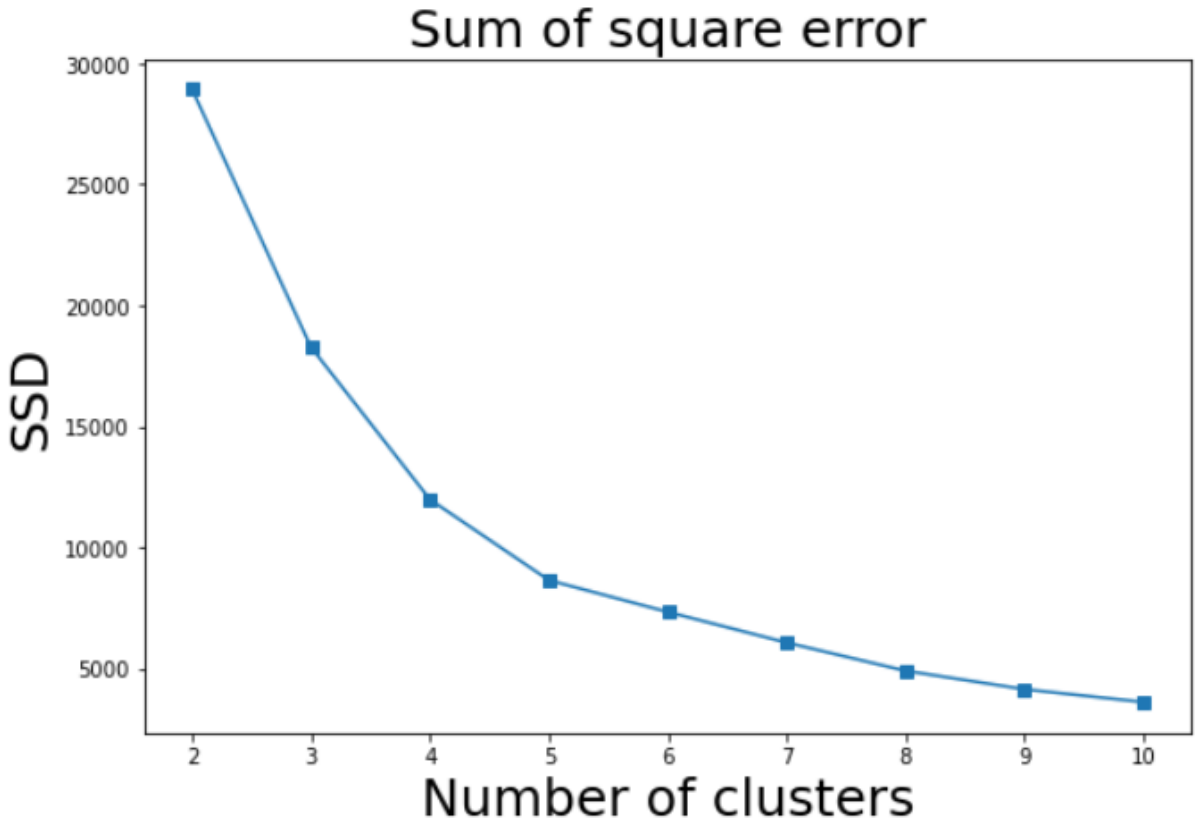
The methods that we have used in order to calculate the number of clusters in K-Means and Hierarchical algorithms are:

1. Silhouette Coefficient
2. Elbow Method

3. Clustering algorithms

Using Silhoutte coefficient and Elbow Method we can say that the number of clusters we need to use is k=2.

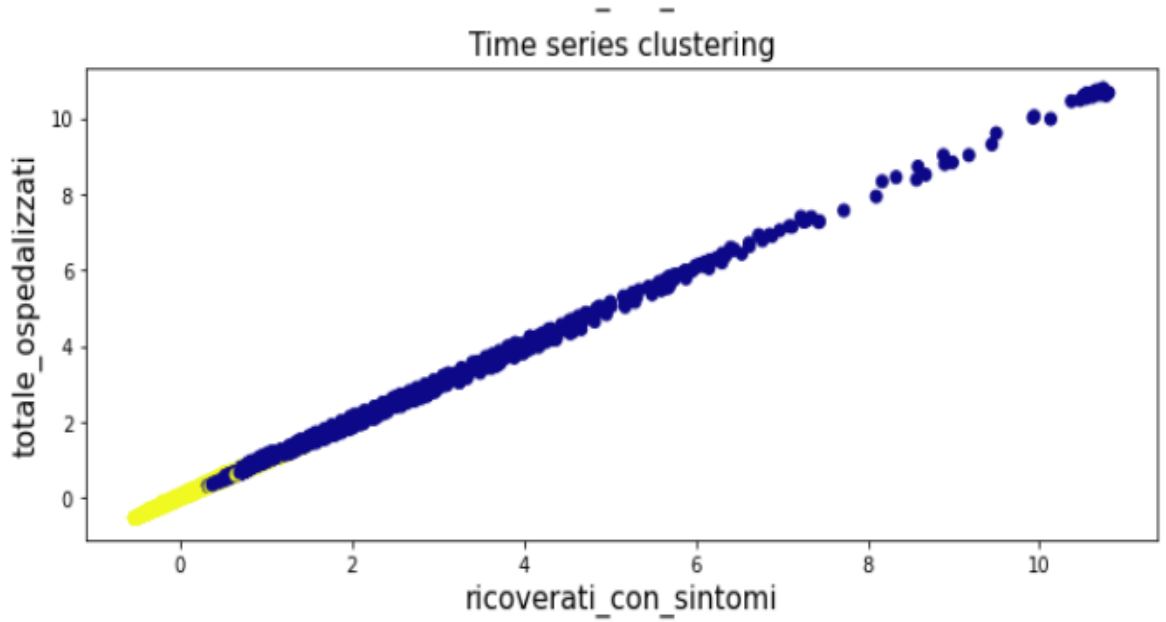


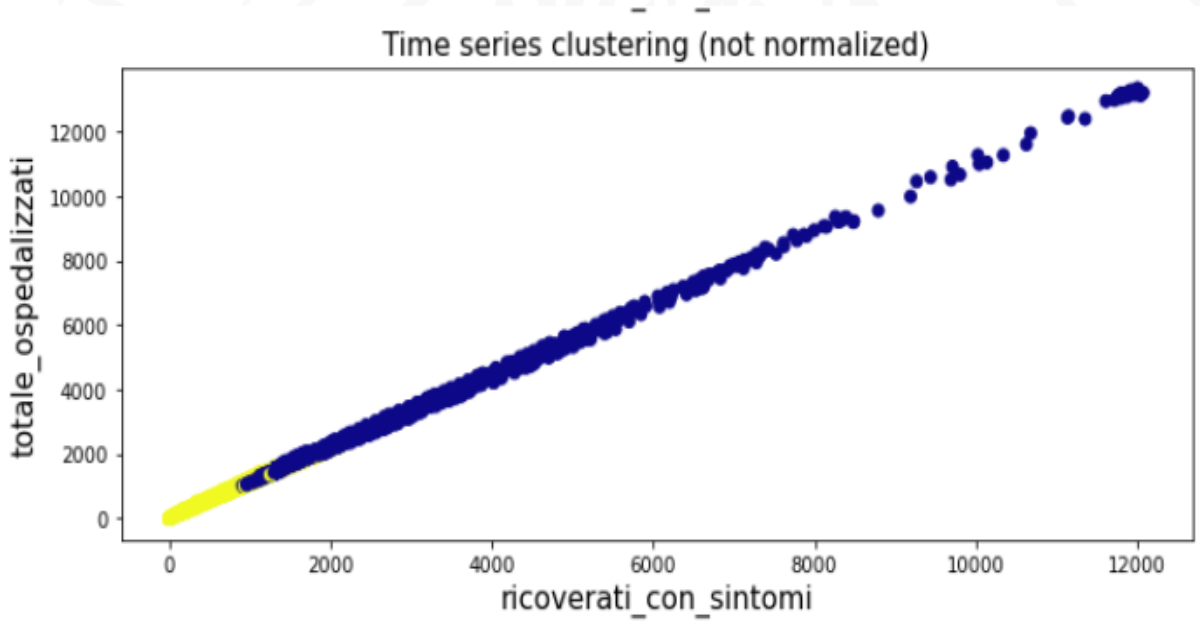


3. Clustering algorithms

A time series is a collection of observations made sequentially in time.

Then we are using a scatter chart in order to see the correlation of our variables with change over time.



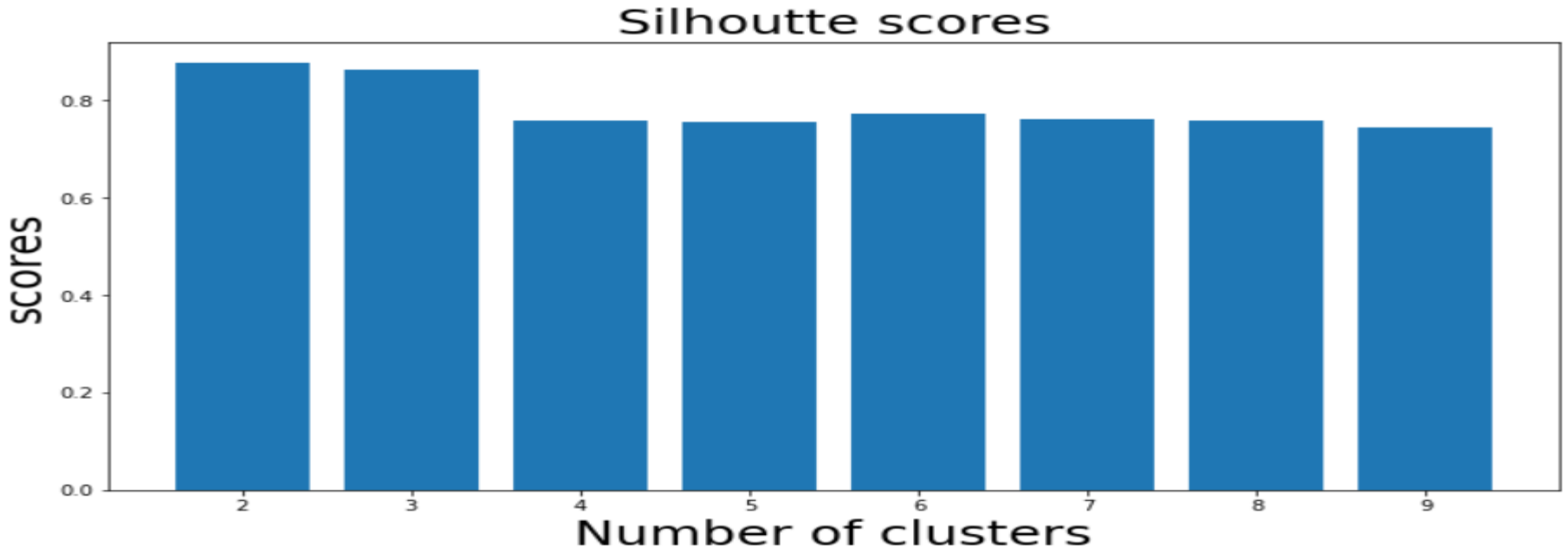


3. Clustering algorithms

Hierarchical clustering is a general family of clustering algorithms that build nested clusters by merging or splitting them successively.

This hierarchy of clusters is represented as a tree (or dendrogram).

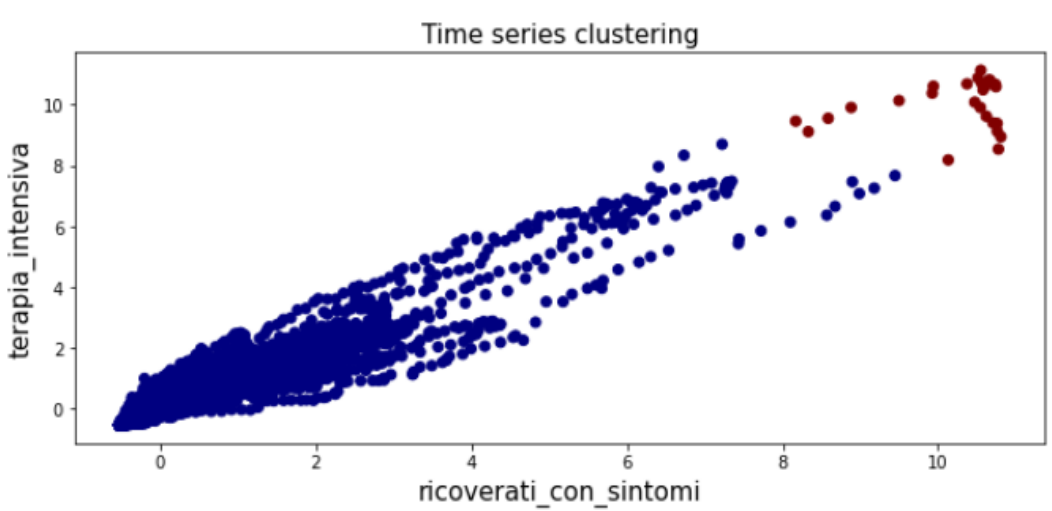
Using Silhoutte coefficient the ideal number of cluster also for the hierarchical clustering is equal to 2.

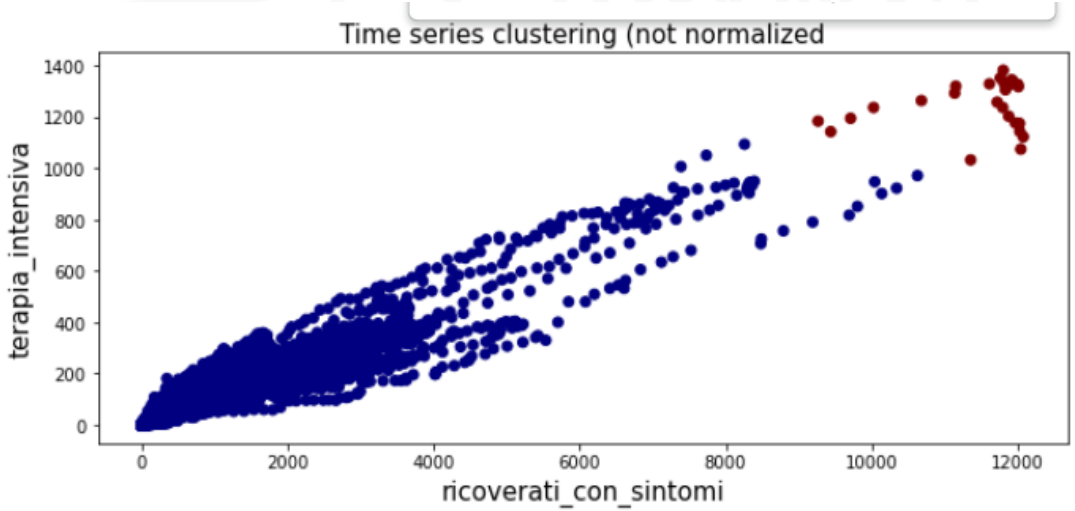


3. Clustering algorithms

In our example we have used agglomerative clustering that start with the points as individual clusters, than at each step it merges the closest pair of clusters until one cluster left.

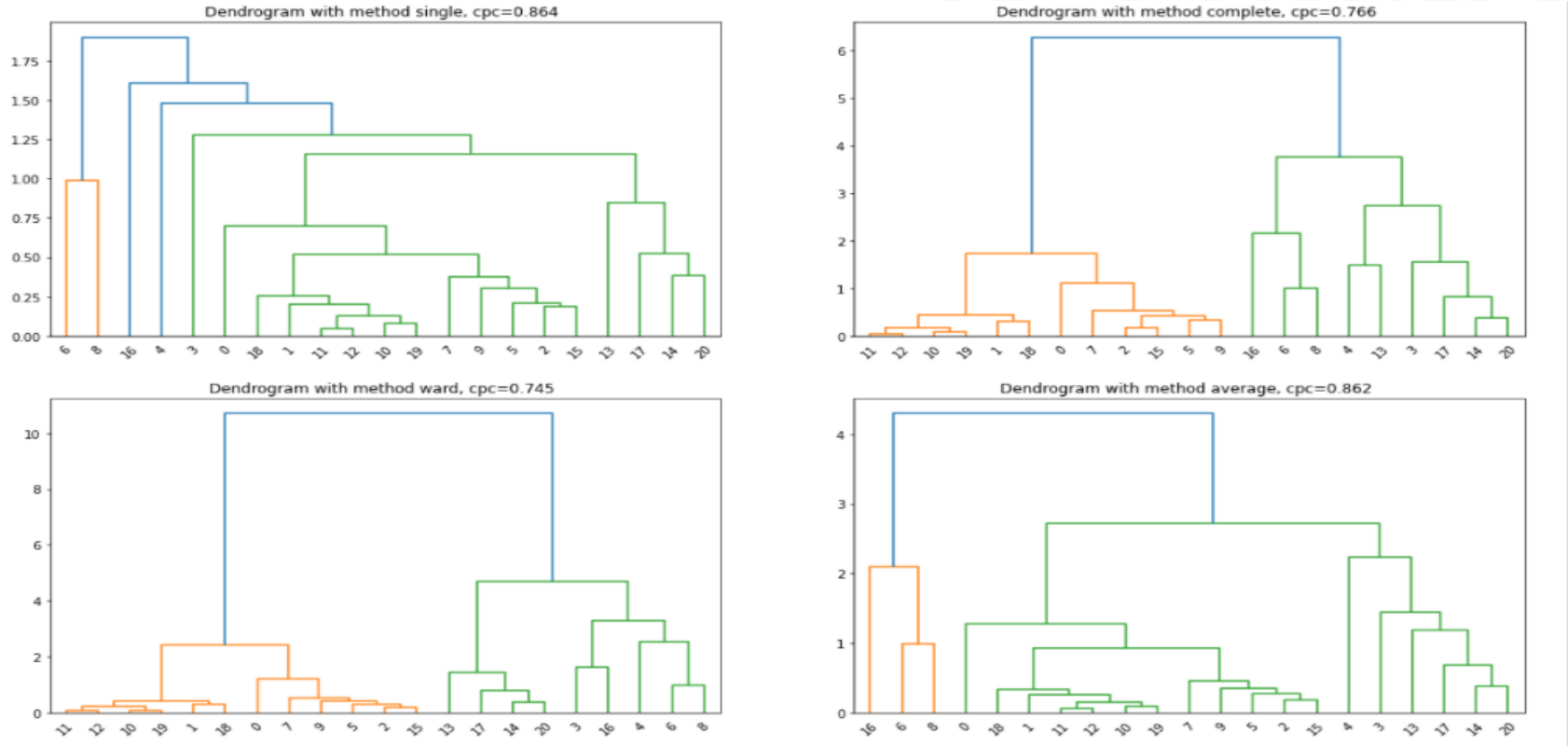
In our example the linkage criterion that we have used is **'average':**





3. Clustering algorithms

A **dendrogram** is a type of tree diagram showing hierarchical clustering or relationships between similar sets of data. In our example we have constructed a dendrogram just for the last day of [dpc-covid19-ita-regioni.json] file in order to see how this diagram works.



3. Clustering algorithms

DBSCAN is one of the most common clustering algorithms, it groups together points that are closely packed together (points with many nearby neighbors), marking as outlier points that lie alone in low-density regions.

In our example the DBScan algorithm is not applied due to the different density present in the data(dpc-covid19-ita-regioni.json) but he have implemented another example in order to represent how it functions.

The DBSCAN algorithm uses two parameters:

minPts: The minimum number of points (a threshold) clustered together for a region to be considered dense.

eps (ε): A distance measure that will be used to locate the points in the neighborhood of any point.

3. Clustering algorithms

In our example we have used another example of DBScan algorithm due to the different density present in the data(dpc-covid19-ita-regioni.json).

The dataset is generated by **sklearn.datasets.make_blobs**.

The **make_blobs()** function draws samples from a special **Gaussian mixture model**.

A **Gaussian mixture model** is a probabilistic model that assumes all the data points are generated from a mixture of a finite number of Gaussian distributions with unknown parameters.

A general Gaussian mixture model with k clusters has a density of the form:

$$p(x) = \sum_{i=1}^{k} \pi_i \mathcal{N}(\mu_i, \Sigma_i)$$

3. Clustering algorithms

Isotropic refers to the fact that the covariance matrices will all be diagonal

$$\Sigma_i = \begin{bmatrix} \sigma_i^2 & 0 \\ 0 & \sigma_i^2 \end{bmatrix}$$

with σi being the standard deviation that is passed in. By default, all clusters will have the same standard deviation.

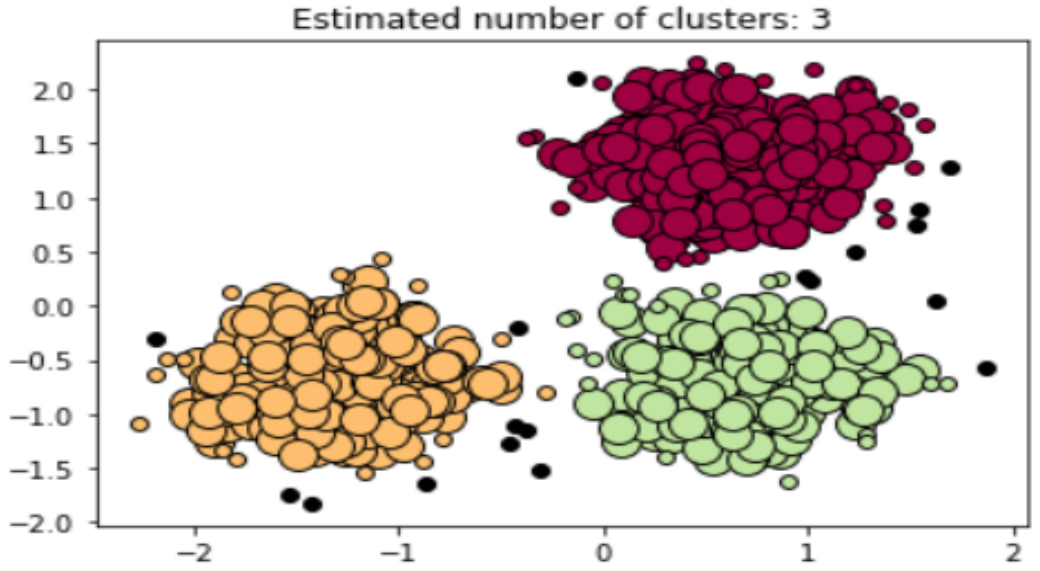
A Gaussian mixture model is not Gaussian unless there is only one cluster, but rather a combination of Gaussians.

3. Clustering algorithms

In our example we will have **eps =0.3**, **minPts=10**, **cluster_std=0.4** that means how tightly data are clustered around the mean, **n_samples=750** total number of points equally divided among clusters.

The results will be:

```
Estimated number of clusters: 3
Estimated number of noise points: 18
Homogeneity: 0.953
Silhouette Coefficient: 0.626
```



4. Conclusions

- **Data Understanding:** We have worked with github.com/pcm-dpc/COVID-19 dataset and we have analyzed the structure and main files of this directory.
- **Data Preprocessing:** We have normalized and standardized the data, dropping the unncessary data, we have represented different graphs in order to understand better the persistence of the virus in various provinces and regions of Italy.
- **Clustering algorithms:** The dataset in our possession is very aggregated and since it is a dataset without labels, we have applied K-means clustering and hierarchical clustering for classifying the data.
- We have used **Gaussian mixture model** as an example for understanding better DBScan algorithm.