Politecnico di Milano

Systems and methods for big and unstructured data

SMBUD project: third delivery

Group 24

Basso Paolo 10783951

Aiello Andrea 10863133

Borsatto Andrea 10628989

Cavalli Dario 10820532

Petriconi Emanuele 10577000

Academic Year 2021–2022

Specifications

The scope of this project is to import and analyze the following dataset on Elasticsearch, somministrazioni-vaccini-latest. It contains information about the COVID-19 vaccine administrations in Italy.

Hypothesis

The only hypothesis made by our team and not by the authors of the dataset (covid19-opendata-vaccini) is that the numeric fields will not be bigger than $2^{31} - 1$. This hypothesis is reasonable knowing what each field represent.

Dataset Schema

After importing the dataset, an index called "**somministrazioni_vaccini**" was created with the following mappings:

Field name	Field type
data_somministrazione	date
fornitore	keyword
area	keyword
fascia_anagrafica	keyword
sesso_maschile	integer
sesso_femminile	integer
prima_dose	integer
seconda_dose	integer
pregressa_infezione	integer
dose_addizionale_booster	integer
codice_NUTS1	keyword
codice_NUTS2	keyword
codice_regione_ISTAT	keyword
nome_area	keyword

The following fields are of type *integer* because they represent an unbounded numeric value which is always less than $2^{31} - 1$ in the dataset, but some values are, or could be in the future, bigger than 32767 so they cannot be mapped to type *short*:

- sesso_maschile
- sesso_femminile
- prima_dose
- seconda_dose

- pregressa_infezione
- dose_addizionale_booster

The field "*data_somministrazione*" is of type *date* because it represents a date in the format YYYY-MM-DD (ISO8601).

The following fields are of type *keyword* because they are strings chosen from a limited number of possible values wich are going to be used in term-level queries:

- fornitore
- area
- fascia_anagrafica
- codice_NUTS1
- codice_NUTS2
- nome_area
- codice_regione_ISTAT

For "*codice_regione_ISTAT*" we decided to use the *keyword* type even though Kibana suggested the *numeric* type. We made this decision because, following the documentation, the possible numeric values are from a limited set and this field will not be used for range queries of any kind while it could be used for term-level queries.

Queries

1. **Total number of somministrated doses in Lombardia after 01/12/2021.**

```
GET /somministrazioni_vaccini/_search
{
    "size":0,
    "query":{
      "bool":{
        "must":[{
          "term":{
            "nome_area":{
              "value":"Lombardia"
            }
          }},
          {"range":{
            "data_somministrazione":{
              "gte": "2021-12-01"
            }
          }}]
      }
    },
    "aggs":{
      "vaccinazioni":{
        "sum":{
          "script": "doc['sesso_femminile'].value + doc['
              sesso_maschile'].value"
        }
      }
    }
}
```

2. **Number of injected doses in Italy during the last month to people under 19 years old.**

```
GET /somministrazioni_vaccini/_search
{
    "size":0,
    "query":{
      "bool":{
        "must":[
          {"range":{
            "data_somministrazione":{
              "gte": "now-1M/d"
            }
          }},
          {"bool":{
            "should":[
              {"term":{"fascia_anagrafica": "05-11"}},
              {"term":{"fascia_anagrafica": "12-19"}}
            ]}
          }]
      }
    },
    "aggs":{
      "vaccinazioni":{
        "sum":{
          "script": "doc['sesso_femminile'].value + doc['sesso_maschile'].value"
        }
      }
    }
}
```

3. **Average number of vaccinated people with previous infection during the last 7 days.**

```
GET /somministrazioni_vaccini/_search
{
    "size":0,
    "query":{
      "bool":{
        "must":{
          "range":{
              "data_somministrazione":{
                "gte": "now-7d/d"
              }
          }
        }
      }
    },
    "aggs":{
      "region":{
        "terms":{
          "field": "nome_area ,
             size    : 99
        },
      "aggs":{
        "avg_pregressa_infezione":{
          "avg": {
            "field": "pregressa_infezione"
          }
        }
      }}
    }
}
```

4. **Average number of daily somministrations in Sardegna during tha last month.**

```
GET /somministrazioni_vaccini/_search {
    "size":0,
    "query":{
      "bool":{
        "must":[{
          "range":{
              "data_somministrazione":{
                "gte": "now-1M/d"
              }
          }
        },
        {"term":{
          "nome_area": "Sardegna"
        }}]
      }
    },
    "aggs":{
      "region":{
        "terms":{
          "field": "data_somministrazione"
        },
      "aggs":{
        "somministrazioni":{
          "avg":{
            "script": "doc['sesso_femminile'].value + doc['
                sesso_maschile'].value"
          }
        }
      }}
    }
}
```

5. **Number of vaccinated people for each kind of vaccine and for each age range from 22/12/2021.**

```
GET /somministrazioni_vaccini/_search
{
    "size":0,
    "query":{
        "bool":{
            "must":{
                "range":{
                "data_somministrazione":{
                    "gte": "2021-12-22"
                    }
                }
            }
        }
    },
    "aggs": {
        "fornitore&eta": {
            "multi_terms":{
                "terms": [
                {"field":"fornitore"},
                {"field":"fascia_anagrafica"}
                ],
                "size": 99
            },
            "aggs":{
                "somministrazioni":{
                    "sum":{
                    "script": "doc['sesso_femminile'].value + doc['
                       sesso_maschile'].value"
                    }
                }
            }
        }
    }
}
```

6. **Number of booster doses given to male and female for each different supplier during the last 3 months.**

```
GET /somministrazioni_vaccini/_search
{
  "size": 0,
    "query": {
      "bool": {
        "must": [
          {"range": {"data_somministrazione": {"gte": "now-3M/d"}}},
          {"range": {"dose_addizionale_booster": {"gte": "1"}}}
        ]
      }
    },
    "aggs": {
      "fornitore":{
        "terms": {
          "field": "fornitore"
        },
        "aggs": {
          "somministrazioni_donne": {
            "sum": {
              "field": "sesso_femminile"
            }
          },
          "somministrazioni_uomini": {
            "sum": {
              "field": "sesso_maschile"
            }
          }
        }
      }
   }
}
```

7. **People over 80 years old that have received the booster.**

```
GET /somministrazione_vaccini/_search {
  "size":0,
    "query": {
      "bool": {
        "must": [
          {"range":{"dose_addizionale_booster": {"gte": "1"}}},
          {"bool":{
            "should":[
              {"term":{"fascia_anagrafica":"80-89"}},
              {"term":{"fascia_anagrafica":"90+"}}
            ]}
          }
        ]
      }
    },
    "aggs":{
      "vaccinazioni": {
        "sum": {
          "script": "doc['sesso_femminile'].value + doc['sesso_maschile']
             ].value"
        }
       }
    }
}
```

8. **Day with the highest number of injected doses during the last 30 days.**

```
GET /somministrazioni_vaccini/_search
{
  "size": 0,
  "query": {
    "bool": {
      "must": [
        {
          "range": {
            "data_somministrazione": {
              "gte": "now-30d/d"
            }
          }
        }
      ]
    }
  },
  "aggs": {
    "days": {
      "terms": {
        "field": "data_somministrazione"
      },
      "aggs": {
        "vaccinations": {
          "sum": {
            "script": "doc['sesso_femminile'].value + doc['
                sesso_maschile'].value"
          }
        },
        "vaccination_sort":{
          "bucket_sort": {
            "sort": [{
              "vaccinations": {"order": "desc"}
            }],
            "size": 1
          }
        }
      }
    }
  }
}
```

Commands

1. **Add a new document to Elasticsearch. Use example values.**

```
POST /somministrazioni_vaccini/_doc
{
  "area": "BAS",
  "codice_regione_ISTAT": "17",
  "nome_area": "Basilicata",
  "data_somministrazione": "2022-01-11",
  "dose_addizionale_booster": 2,
  "codice_NUTS1": "ITF",
  "fascia_anagrafica": "12-19",
  "prima_dose": 1,
  "pregressa_infezione": 0,
  "fornitore": "Janssen",
  "seconda_dose": 0,
  "sesso_maschile": 3,
  "codice_NUTS2": "ITF5",
  "sesso_femminile": 0
}
```

2. **A document in the database has a small error that must be corrected. The document representing the vaccinations done in the "BAS" area with the "Janssen" vaccine on the "2021-12-22" (data_somministrazione) has reported less vaccinations that what actually happened. In particular sesso_maschile must be updated from 1 to 2 as well as prima_dose. An initial query found that the id of the document is "Btox530BZF4mmwtvcTdZ".**

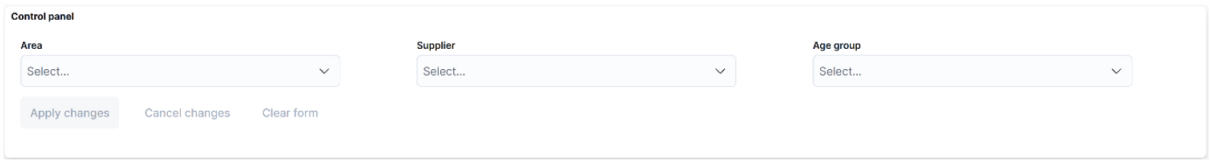
```
POST /somministrazioni_vaccini/_update/Btox530BZF4mmwtvcTdZ
{
  "doc": {
    "prima_dose" : 2,
    "sesso_maschile" : 2
  }
}
```

3. **Delete the document older than 6 months (since we do not want to analyse those).**

```
POST /somministrazioni_vaccini/_delete {
  "query": {
    "range": {
      "data_somministrazione": {
        "lte": "now-6M/d"
      }
    }
  }
}
```

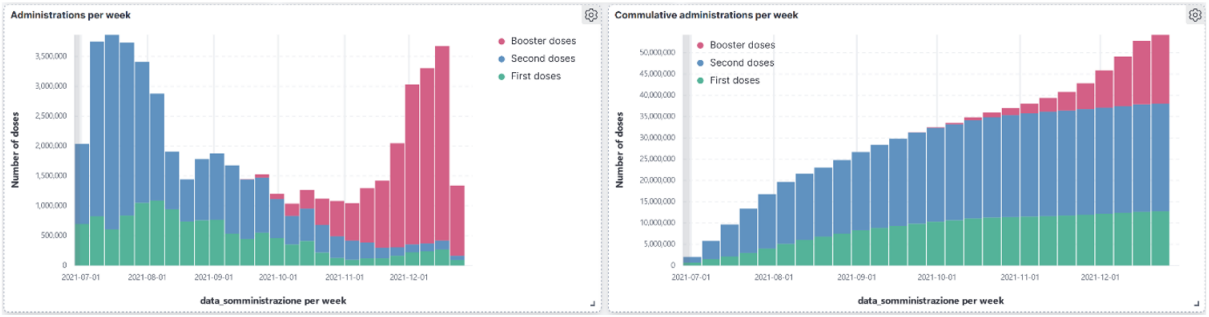
Kibana Dashboard

To help with the understanding of the data, we built a **Kibana Dashboard**. The dashboard is built as follows:

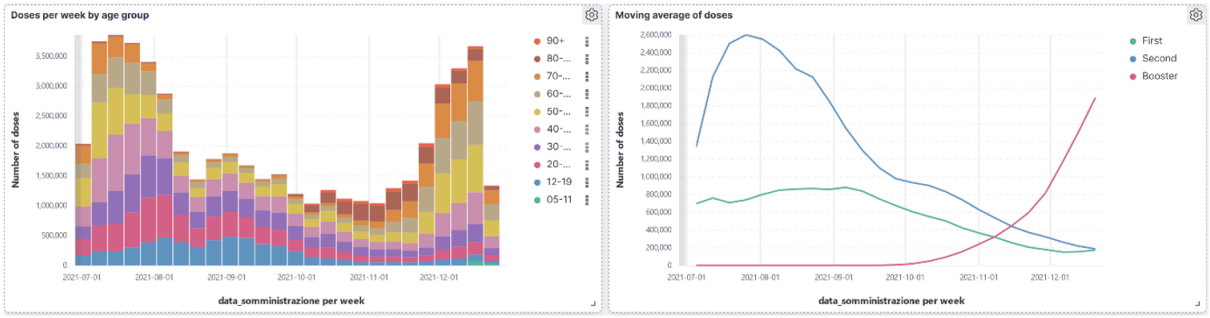


A control panel is used to filter the data according to the area, the supplier and the age group. In all three these fields multiple values can be chosen simultaneously.

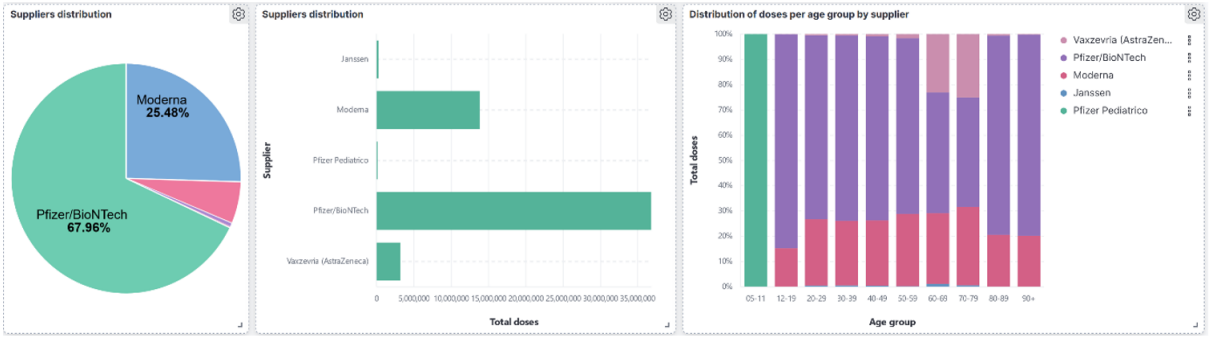
In the second row, we focus on the number of doses administered. The first graph shows the total number of doses, the second divides this number in first, second and third doses. The third graph shows how many doses have been administered to each age group and the graph is broken by dose type.



In the next row, we focus on the administrations. The first graph shows the number of administrations per week while the second graph shows the cumulative sum of administrations per week. Both graphs are broken by the dose type.



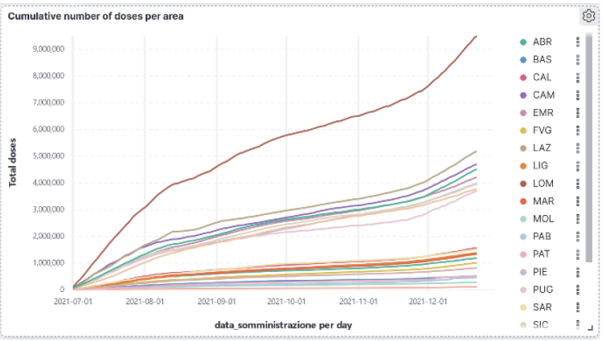
In the next row, we firstly show the number of doses administered per week but this time we break the graph by the age group. The second graph shows the moving average (window size = 7) of the administration per week broken by the type of dose.



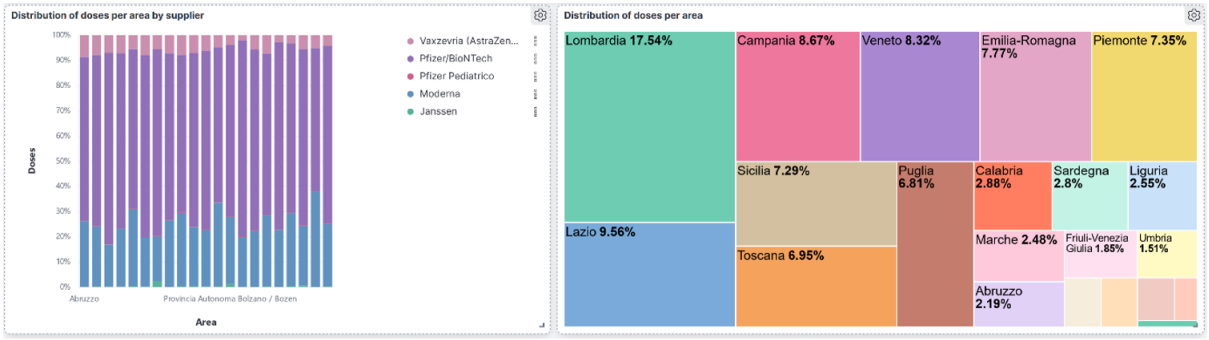
In the next row, we focus on the suppliers of the doses. The first graph is a pie chart that shows the suppliers distribution. The second graph is a horizontal bar chart which shows more clearly how many doses for each supplier have been administered. The last graph shows the distribution of the supplier used for each age group. This graph is interesting since it clearly shows that some suppliers are preferred for each age group.

Number of doses per area

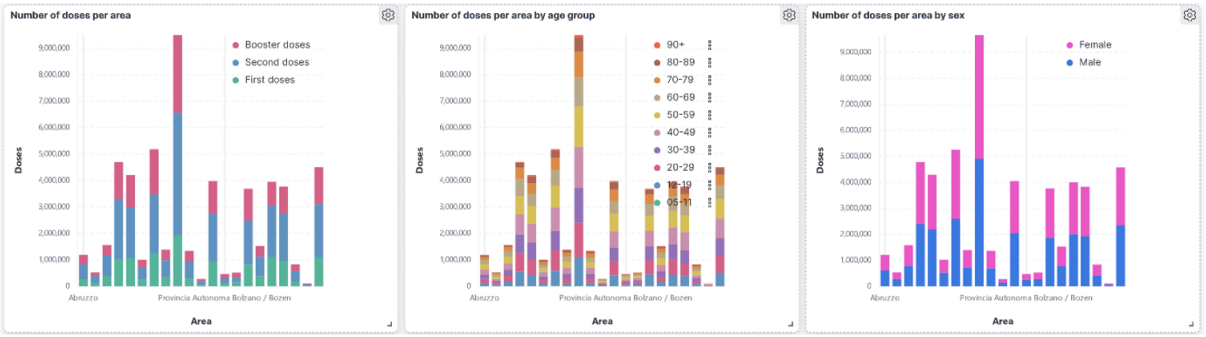
Area	Total dose	First doses	Second do	Booster do	Male	Female
Abruzzo	1,185,953	264,008	571,282	350,663	606,602	595,085
Basilicata	523,214	139,818	244,466	138,930	268,165	262,474
Calabria	1,562,932	387,490	775,206	400,236	790,394	787,236
Campania	4,699,041	1,023,372	2,268,474	1,407,195	2,390,311	2,389,656
Emilia-Rom...	4,208,742	1,058,500	1,929,347	1,220,895	2,182,229	2,108,965
Friuli-Vene...	1,003,322	248,762	457,772	296,788	521,512	501,639
Lazio	5,181,026	1,238,427	2,244,166	1,698,433	2,600,495	2,651,018
Liguria	1,382,419	342,220	641,475	398,724	700,079	695,316
Lombardia	9,503,025	1,923,659	4,641,903	2,937,463	4,904,092	4,763,786
Marche	1,341,474	287,776	637,266	416,432	690,308	674,927



We then focus on the administrations per region. We include a table with the more salient data for each region. On the left there is a graph which shows the cumulative number of doses administered in each area.



We then include a graph which for each region shows the suppliers distribution. On the left we included another graph which shows the distribution of doses per area in a different and maybe clearer way.



In the last row we included three graphs which shows some useful metrics about the number of doses administered for each region. The first graph focuses on the type of doses, the second on the age groups and the third on the sex.

Optional 1: Integrating other datasets

Specifications

Import in Elasticsearch the following dataset dpc-covid19-ita-andamento-nazionale. It contains information about tests, hospitalizations, intensive care, home confinement, positive cases, recoveries, and deaths regarding the COVID-19 pandemic in Italy. A description of the fields is available here.
Import only these fields (which are the ones still updated):

1. data
2. ricoverati_con_sintomi
3. terapia_intensiva
4. totale_ospedalizzati
5. isolamento_dominciliare
6. totale_positivi
7. variazione_totale_positivi
8. nuovi_positivi
9. dimessi_guariti
10. deceduti
11. totale_casi
12. tamponi

Importing the dataset to Elasticsearch

As specifications we need to whitelist some fields. A small python script is used to get only the wanted fields as a csv.

```
import pandas as pd

df = pd.read_csv("https://github.com/pcm-dpc/COVID-19/raw/master/dati-andamento-nazionale/dpc-covid19-ita-andamento-nazionale.csv")

df = df [[
  "data",
  "ricoverati_con_sintomi",
  "terapia_intensiva",
  "totale_ospedalizzati",
  "isolamento_domiciliare",
  "totale_positivi",
  "variazione_totale_positivi",
  "nuovi_positivi",
  "dimessi_guariti",
  "deceduti",
  "totale_casi",
  "tamponi",
]]
print(df.to_csv(index=False))
```

Data Mappings

We imported the data and created an index **"covid19_italy"** with the following mappings:

Field name	Field type
data	date
ricoverati_con_sintomi	integer
terapia_intensiva	integer
totale_ospedalizzati	integer
isolamento_domiciliare	Integer
totale_positivi	Integer
variazione_totale_positivi	Integer
nuovi_positivi	Integer
dimessi_guariti	Integer
deceduti	Integer
totale_casi	Integer
tamponi	long

The following fields are of type *integer* because they represent an unbounded numeric value which is always less than $2^{31} - 1$ in the dataset, but some values are, or could be in the future, bigger than 32767 so they cannot be mapped to type *short*:

- ricoverati_con_sintomi
- terapia_intensiva
- totale_ospedalizzati
- isolamento_domiciliare
- totale_positivi
- variazione_totale_positivi
- nuovi_positivi
- dimessi_guariti
- deceduti
- totale_casi

Since "*tamponi*" is a cumulative value that could become big we decided to map this field to *long* to be sure that the upper bound will not be a problem in the future.

The field "*data*" is of type *date* because it represents a date in the format YYYY-MM-DDTHH:MM:SS (ISO 8601).

Queries

1. **Daily average of new positive cases from 20/12/22.**

```
GET /covid19_italy/_search
{
  "size":0,
  "query":{
  "bool":{
    "must":{
      "range":{
          "data":{
            "gte": "2021-12-22"
          }
      }
    }
  }
},
  "aggs": {
    "media_positivi":{
      "avg":{
        "field": "nuovi_positivi"
      }
    }
  }
}
```

2. **Highest record of new positive cases.**

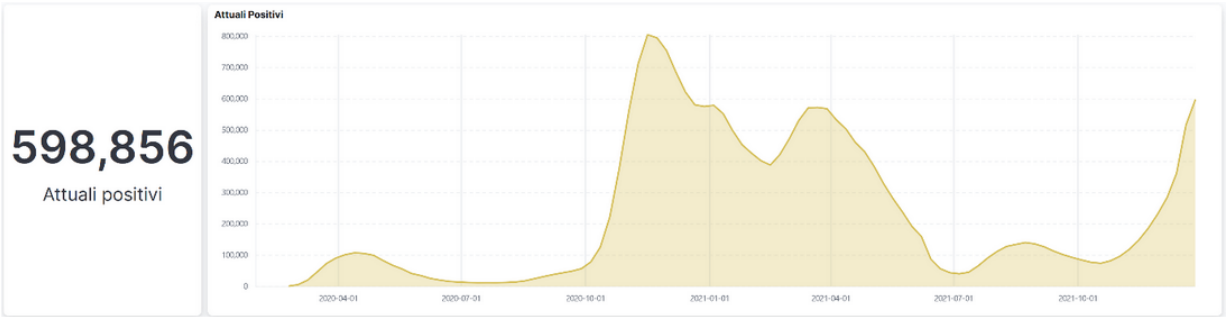
```
GET /covid19_italy/_search
{
    "size": 0,
    "aggs":{
      "max_nuovi_positivi":{
        "max": { "field": "nuovi_positivi"}
      }
    }
}
```

3. **Day with the highest number of hospitalizations.**

```
GET /covid19_italy/_search
{
    "size": 5,
    "sort": [
      {"totale_ospedalizzati": {"order": "desc"}}
    ],
    "fields":[
      "data",
      "totale_ospedalizzati"
      ],
    "_source": false
}
```

Dashboard

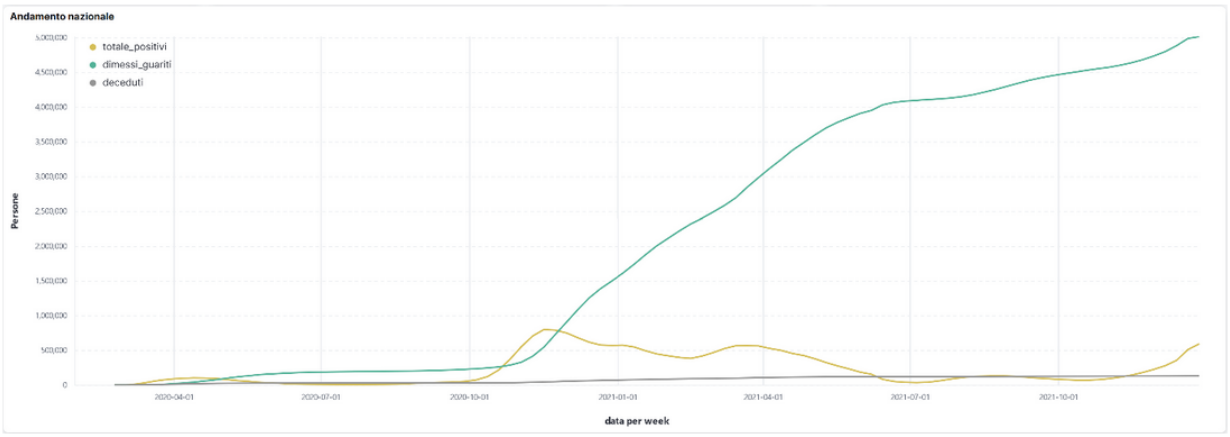
We built a dashboard named "**Optional COVID Italy dashboard**" and we included it in the delivery.



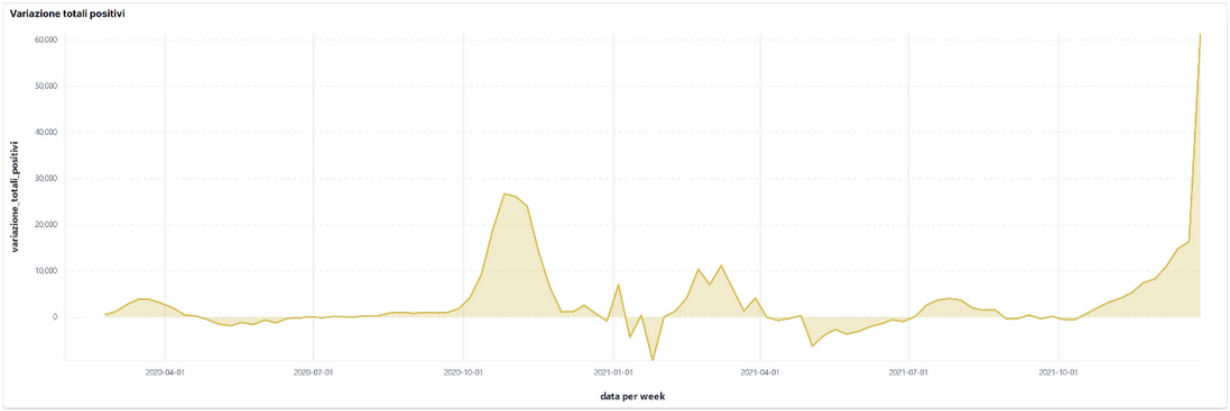
In the first row, we show the total number of positive people now and over time.

In the second row, we first show the cumulative number of healed people now and overtime. Then we show the number of new cases today and every day over time.

In the third row on the right, we show the total cumulative number of cases now and over time while on the left we show the number of cumulative number of deaths now and overtime.



We then include a graph which shows the number of positives, the cumulative number of healed and death over time.



In the last row we show the variation of the number of positives over time.

Optional 2: Implementing another NoSQL platform

We decided to implement the dataset in **Cassandra**.

Create Database structure

Firstly we create the keyspace:

```
CREATE KEYSPACE IF NOT EXISTS population WITH replication = {"class": "
    SimpleStrategy", "replication_factor": 3};

//  Replace the double quotes with single ones.
```

Then we create the table:

```
USE population;

CREATE TABLE vaccinations (data_somministrazione date, fornitore text, area
    text, fascia_anagrafica text, sesso_maschile int, sesso_femminile int,
    prima_dose int, seconda_dose int, pregressa_infezione int,
    dose_addizionale_booster int, codice_NUTS1 text, codice_NUTS2 text,
    codice_regione_ISTAT int, nome_area text, PRIMARY KEY((
    data_somministrazione, fornitore, area, fascia_anagrafica)) );
```

Importing the data

We can now import the data:

```
// Import data from a csv file located in the specified folder (e.g: 'C:\
    OneDrive\Desktop\prova.csv')

// The same dataset used for ElasticSearch can be used here

COPY vaccinations (data_somministrazione, fornitore, area, fascia_anagrafica
    , sesso_maschile, sesso_femminile, prima_dose, seconda_dose,
    pregressa_infezione, dose_addizionale_booster, codice_NUTS1, codice_NUTS2
    , codice_regione_ISTAT, nome_area) FROM 'path-to-csv-file\file.csv' WITH
    DELIMITER=',' AND HEADER=TRUE;
```

Indexes

```
// for performance and avoiding ALLOW FILTERING in the queries

CREATE INDEX region_area ON vaccinations(area);

CREATE INDEX date_index ON vaccinations(data_somministrazione);
```

User defined functions

We define a user custom function that we will use in the queries:

```
// User defined functions must be enabled in path-to-cassandra-folder\
   cassandra\conf\cassandra.yaml, setting the variable
   enable_user_defined_functions to true

CREATE FUNCTION sum_different_columns ( arg1 int, arg2 int )

    RETURNS NULL ON NULL INPUT

    RETURNS int

    LANGUAGE java

    AS $$ return (arg1+arg2); $$;
```

Queries

1. **Query data about vaccinations on 28 december 2021 using Moderna on people aged between 40-49 years old.**

```
    SELECT * FROM vaccinations WHERE data_somministrazione='2021-12-28'
       AND fornitore='Moderna' AND fascia_anagrafica='40-49' ALLOW
       FILTERING;
```

2. **Find the total number of doses inoculated in Lombardia.**

```
    // Use ALLOW FILTERING if the index is not defined

    SELECT area, sum_different_columns(SUM(sesso_maschile), SUM(
       sesso_femminile)) AS total_vaccination FROM vaccinations WHERE
       area='LOM';
```

3. **Find the total number of doses inoculated in Lombardia on men and women.**

```
    // Use ALLOW FILTERING if the index is not defined

    SELECT SUM(sesso_maschile) AS man_total, SUM(sesso_femminile) AS
       women_total FROM vaccinations WHERE area='LOM';
```

4. **Find for which vaccine type and on which region and age group were inoculated the most doses of booster on 28 december 2021.**

```
    SELECT MAX(dose_addizionale_booster) AS max_booster, area,
       fascia_anagrafica, fornitore FROM vaccinations WHERE
       data_somministrazione = '2021-12-28';
```

Commands

1. **Insert a new record on the database.**

```
INSERT INTO vaccinations(data_somministrazione, fornitore, area,
    fascia_anagrafica, sesso_maschile, sesso_femminile, prima_dose,
    seconda_dose, pregressa_infezione, dose_addizionale_booster,
    codice_NUTS1, codice_NUTS2, codice_regione_ISTAT, nome_area)

VALUES ('2021-12-29','Pfizer/BioNTech','EMR','20-29',1,0,0,0,0,1,'
    ITH','ITH5',8,'Emilia-Romagna');
```

2. **Update a record on the database using "data_somministrazione", fornitore, area and "fascia_anagrafica".**

```
UPDATE vaccinations

 SET sesso_maschile= 2, dose_addizionale_booster=2

WHERE data_somministrazione = '2021-12-29' AND fornitore='Pfizer/
    BioNTech' AND area ='EMR' AND fascia_anagrafica='20-29';
```