Project Work - Elastic Search Information System Prof. Marco Brambilla

# Systems and Methods for Big and Unstructured Data

POLITECNICO
MILANO 1863

Curti Gabriele [10624502]
Cutrupi Lorenzo [10629494]
Samuele Mariani [10622653]
Alessandro Molteni [10623928]
Matteo Monti [10622780]

January 18, 2022

# Contents

# 1 Introduction

The project is about designing, storing and querying a database using technologies shown during the lessons. In this case we were asked to build an Elastic Search database to store a given dataset in order to implement some queries aimed at exploring the data statistics and to implement a visualization dashboard to better explore the results.

## 1.1 Problem specifications and hypothesis

In order to store the dataset the group has decided to keep the same data schema as the one provided from the professor available ***here***.

The following is a description of each one of the data fields:

| Field name | Data Type | Description |
|---|---|---|
| **Index** | Integer | The index of the record |
| **Area** | String | Acronyms of the region of delivery |
| **Supplier** | String | Complete name of the supplier of the vaccine |
| **Administration Date** | Datetime | Administration date of the vaccines |
| **Age Group** | String | Age group of the people administered with the vaccines |
| **Male Count** | Integer | Number of vaccinations administered to males |
| **Female Count** | Integer | Number of vaccinations administered to females |
| **First Doses** | Integer | Number of people administered with the first dose |
| **Second Doses** | Integer | Number of people administered with the second dose |
| **Post Infection Doses** | Integer | Number of people administered with a dose after they have been infected |
| **NUTS1 Code** | String | https://en.wikipedia.org |
| **NUTS2 Code** | String | https://en.wikipedia.org |
| **Region ISTAT Code** | Integer | ISTAT code of a region |
| **Region Name** | String | Name of the region (bilingual, when necessary) |

An example of document is the following:

| | |
|---|---|
| t _id | XEtB_H0B8RwFVmbcHWjl |
| t _index | c19-data |
| # _score | - |
| t _type | _doc |
| @timestamp | Dec 26, 2021 @ 00:00:00.000 |
| t area | ABR |
| t codice_NUTS1 | ITF |
| t codice_NUTS2 | ITF1 |
| # codice_regione_ISTAT | 13 |
| data_somministrazione | Dec 26, 2021 @ 01:00:00.000 |
| # dose_addizionale_booster | 17 |
| t fascia_anagrafica | 12-19 |
| t fornitore | Moderna |
| t nome_area | Abruzzo |
| # pregressa_infezione | 0 |
| # prima_dose | 1 |
| # seconda_dose | 11 |
| # sesso_femminile | 12 |
| # sesso_maschile | 17 |

# 2 Queries and Commands

In order to retrieve useful data from the dataset, from both user perspective and big data analysis perspective, some simple queries were designed with the intent of simulating some of the basic operations that such DataBase could be used for.
Additionally two commands were designed with the intent of demonstrating how the database could be modified and updated.

## 2.1 Queries

1. Counts the amount of somministrations on males and females in each region:

```
GET /c19-data/_search
{
    "size": 0,
    "aggs": {
        "Somministrations by region": {
            "terms": {
                "field": "nome_area"
            },
            "aggs": {
                "Number of somministrations on females": {
                    "sum": {
                        "field": "sesso_femminile"
                    }
                },
                "Number of somministrations on males": {
                    "sum": {
                        "field": "sesso_maschile"
}}}}}}
```

2. Returns the amount of somministrations for each number of dose:

```
GET /c19-data/_search
{
    "aggs": {
        "1st doses before a certain day": {
          "sum": {
            "field": "prima_dose"
            }
        },
        "2nd doses before a certain day": {
          "sum": {
            "field": "seconda_dose"
            }
        },
        "booster doses before a certain day": {
          "sum": {
            "field": "dose_addizionale_booster"
            }
        }
    }
}
```

3. Returns the amount of somministrations before a certain day for each number of dose:

```
GET /c19-data/_search
{
    "size": 0,
    "query": {
        "bool": {
            "must": [
                {"range": {
                    "data_somministrazione": {
                    "lte": "2021-04-29"
            }}]}}},
    "aggs": {
        "1st doses before a certain day": {
          "sum": {
            "field": "prima_dose"
            }
        },
        "2nd doses before a certain day": {
          "sum": {
            "field": "seconda_dose"
            }
        },
        "booster doses before a certain day": {
          "sum": {
            "field": "dose_addizionale_booster"
}}}}
```

4. Counts the amount of second doses and first doses after infection for each age range:

```
GET /c19-data/_search
{
    "size": 0,
    "aggs": {
        "age range": {
            "terms": {
                "field": "fascia_anagrafica"
            },
            "aggs": {
                "2nd dose": {
                    "sum": {
                        "field": "seconda_dose"
                    }
                },
                "1st dose after infection": {
                    "sum": {
                         "field": "pregressa_infezione"
                    }
                }
            }
        }
    }
}
```

5. Counts the amount of males and females vaccinated for each supplier:

```
GET /c19-data/_search
{
    "size": 0,
    "aggs": {
        "supplier": {
            "terms": {
                "field": "fornitore"
            },
            "aggs": {
                "males": {
                    "sum": {
                        "field": "sesso_maschile"
                    }
                },
                "females": {
                    "sum": {
                        "field": "sesso_femminile"
}}}}}}
```

6. Counts the number of vaccinations on male and female children (under 11) for each region:

```
GET /c19-data/_search
{
    "size": 0,
    "query": {
        "bool": {
            "filter": [
                {
                "term": {
                   "fascia_anagrafica": "05-11"
            }}]}},
    "aggs": {
        "supplier": {
            "terms": {
                "field": "nome_area"
            },
            "aggs": {
                "males": {
                    "sum": {
                        "field": "sesso_maschile"
                    }
                },
                "females": {
                    "sum": {
                        "field": "sesso_femminile"
}}}}}}
```

7. Counts the amount of vaccination on a certain day with all type of supplier except Moderna:

```
GET /c19-data/_search
{
    "size": 0,
    "query": {
        "bool": {
            "must": [
                {"term": {
                    "data_somministrazione": {
                        "value": "2021-04-21"
            ]}}},
            "must_not": [
                {"term": {
                    "fornitore": {
                        "value": "Moderna"
    }}]}}},
    "aggs": {
        "amount of male vaccinations": {"sum":
            {"field":"sesso_maschile"}},
        "amount of female vaccinations": {"sum":
            {"field":"sesso_femminile"}}
    }
}
```

8. This query returns the documents concerning vaccinations administrated between the 24 and 29 of April only if they **are** from MODERNA:

```
GET /c19-data/_search
{
  "query": {
    "bool": {
      "must": [
        {"range": {
          "data_somministrazione": {
            "gte": "2021-04-24",
            "lte": "2021-04-29"
          }
        }},
        {
         "term": {
            "fornitore": {
              "value": "Moderna"
            }
          }
        }
      ]
    }
  }
}
```

9. Returns the documents concerning vaccinations administrated between the 24 and 29 of April only if they are **NOT** from MODERNA and with an higher score if they are from Pfizer/BioNTech:

```
GET /c19-data/_search
{
"query": {
    "bool": {
        "must": [
            {"range": {
                "data_somministrazione": {
                    "gte": "2021-04-24",
                    "lte": "2021-04-29"
                }
            }}
        ],
        "must_not": [
            {"term": {
                "fornitore": {
                    "value": "Moderna"
                }
            }}
        ],
        "should": [
            {"term": {
                "fornitore": {
                    "value": "Pfizer/BioNTech"
}}}]}}}
```

## 2.2 Commands

1. This command updates the number of first doses in a document

```
POST /c19-data/_update/X0tB_H0B8RwFVmbcHWjl
{
  "doc": {
    "prima_dose":10
  }
}
```

2. This command updates the vaccine provider in a document

```
POST /c19-data/_update/7EtB_H0B8RwFVmbcHWjl
{
  "doc": {
    "fornitore": "Moderna"
  }
}
```

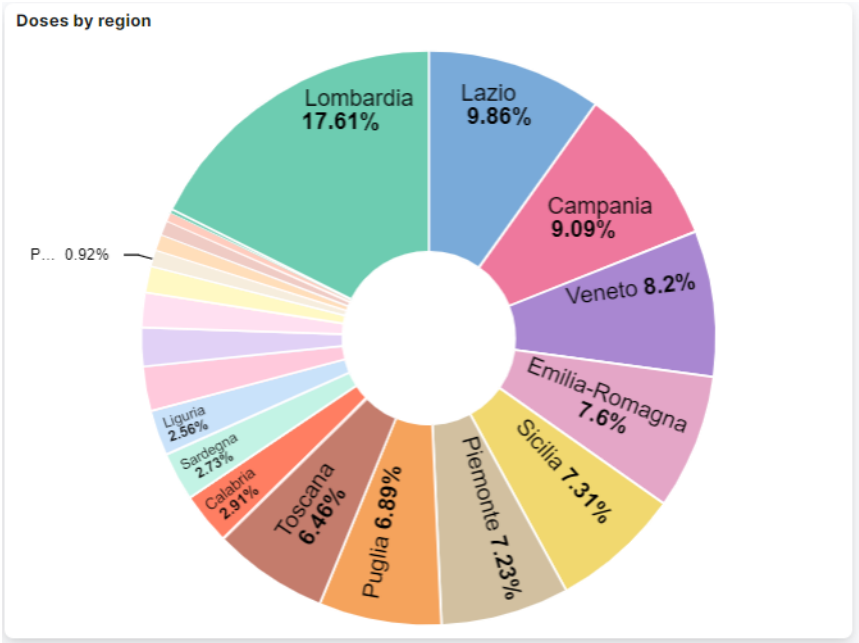
3. Inserts a new document in the index

```
POST /c19-data/_doc
{
        "area": "ABR",
        "codice_NUTS1": "ITF",
        "codice_NUTS2": "ITF1",
        "codice_regione_ISTAT": 13,
        "data_somministrazione": "2021-04-21",
        "fascia anagrafica": "12-19",
        "fornitore": "Moderna",
        "nome_area": "Abruzzo",
        "sesso_maschile": 38,
        "sesso_femminile": 12,
        "prima_dose": 20,
        "seconda_dose": 10,
        "pregressa_infezione": 15,
        "dose_addizionale_booster": 5
}
```

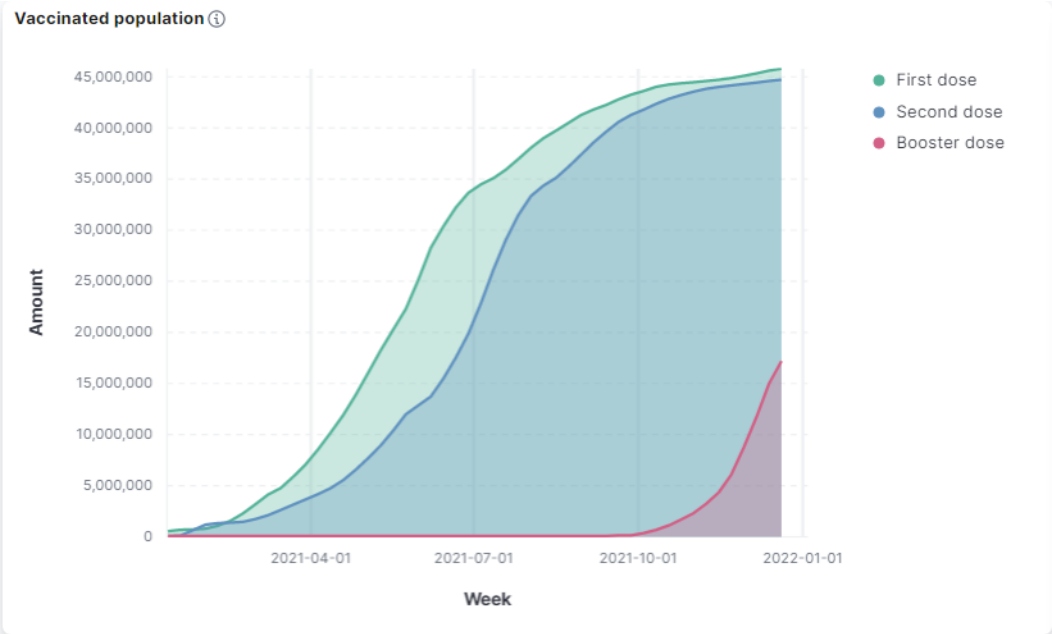
# 3 Kibana dashboard

To aid the data visualization task we implemented **kibana** in the elastic search stack and built a dashboard with some visualization:

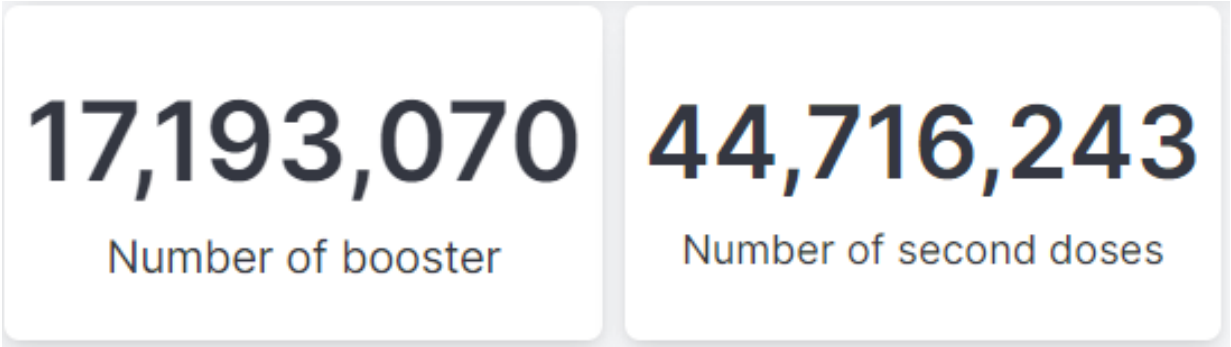
1. The diagram below shows the percentage of somministration in each region on the total somministrations, emphasising the ones with the most. It is strictly related to Query 1 since it requires the somministrations for each region.



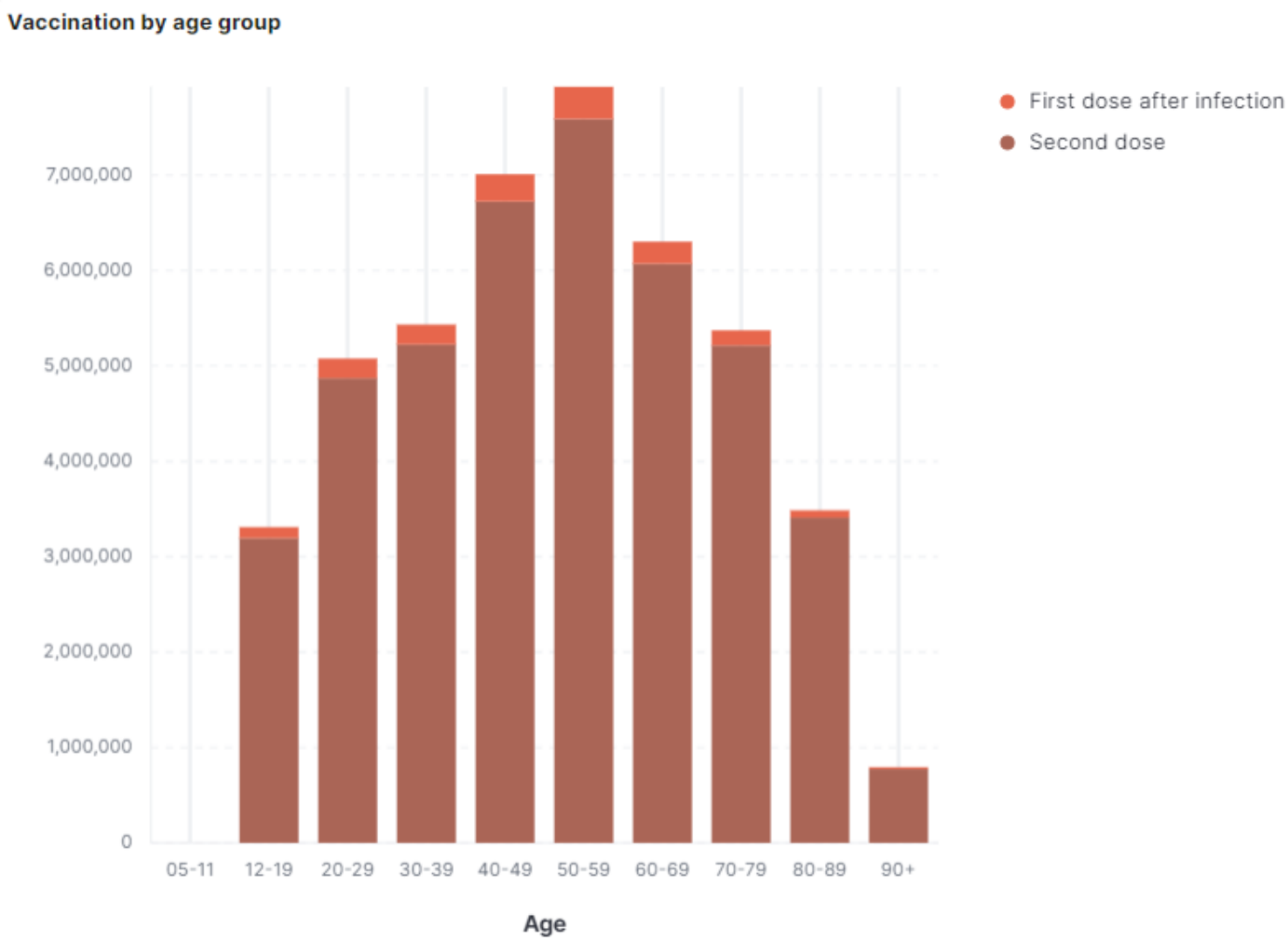

2. The diagram below shows the trend of the somministrations for each dose (first, second and booster). It is basically Query 3 done with all the days of the year.



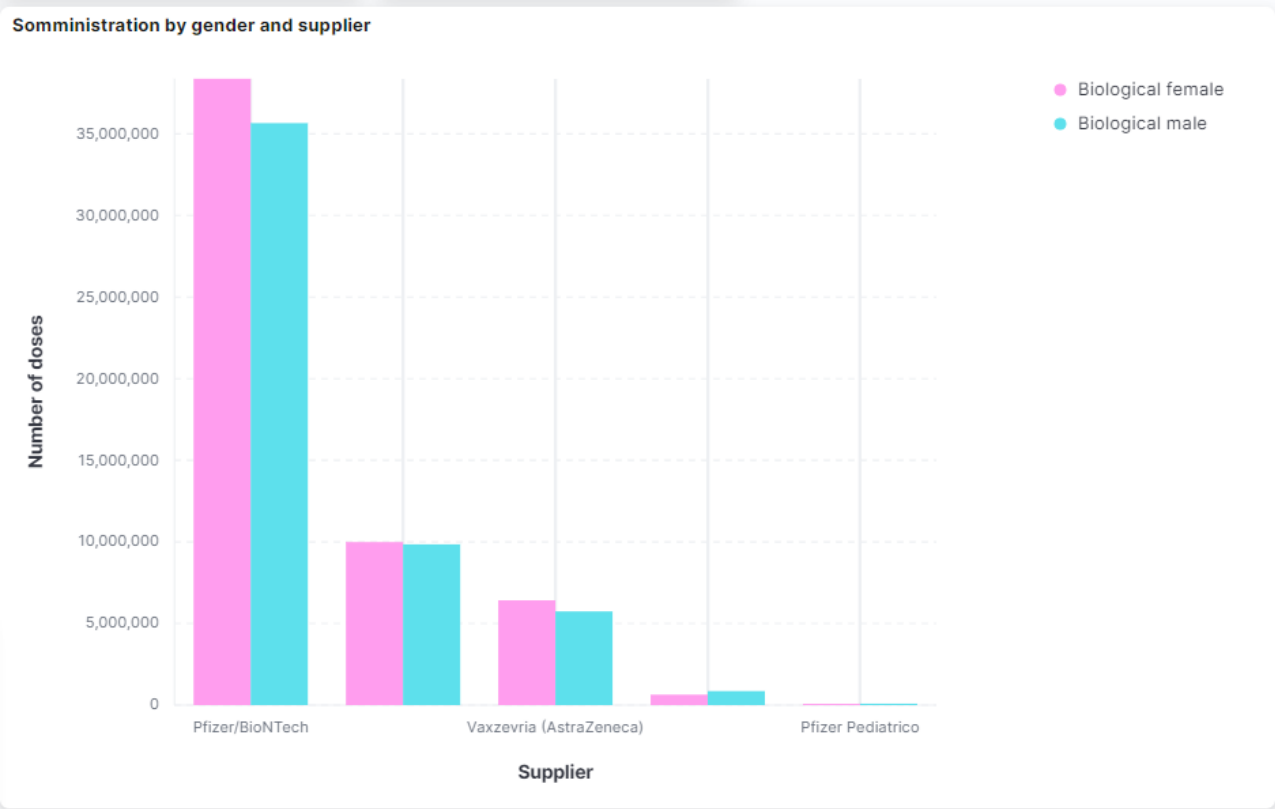
3. The image below describes the amount of people vaccinated with booster dose and second dose. It is done similarly to Query 2.



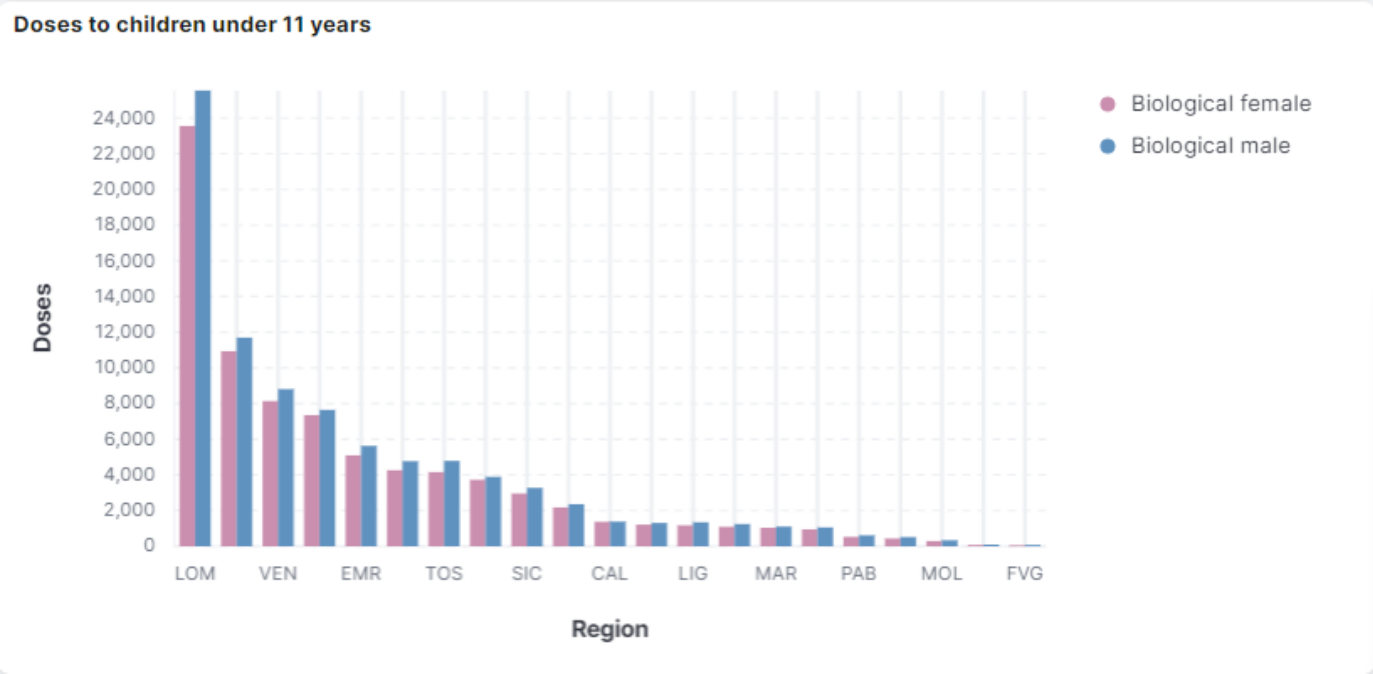

4. The graph below shows the amount of immunized (second dose or first dose after infection) people for each range of age. It is the result of Query 4.



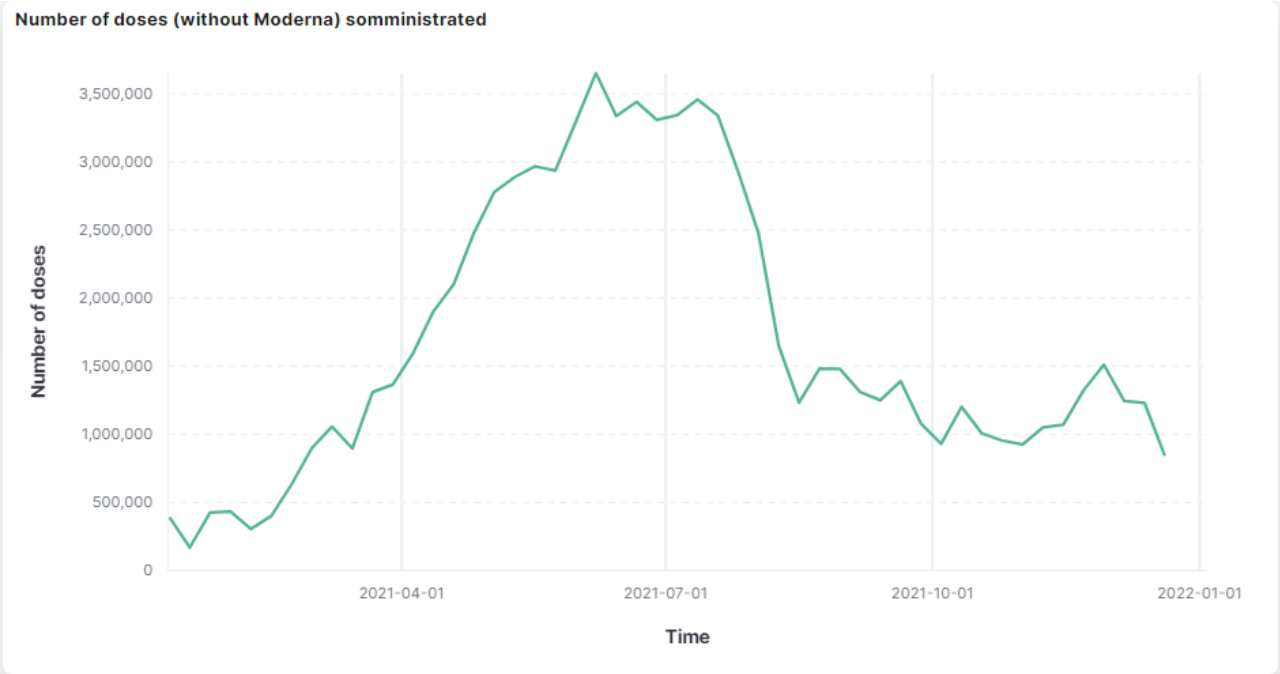
5. The diagram below shows the count of somministration on males and females done with each supplier. It is similar to Query 5.




6. The graph below shows the number of male and female children (under 11) who are vaccinated in each region. It is the result of Query 6.



7. The image below shows the number of daily doses somministrated with all suppliers except Modern. It is similar to Query 7, done for all the days of the last year




The exported Kibana dashboard is in the delivery folder, and can be imported using the index name "c19-data".

# 4 Worldwide Dataset Integration

An additional dataset named `vaccination-world.csv` as been integrated in the project and is available in the .zip delivery folder. This additional dataset is a collection of data (71993 entries) concerning vaccinations in the whole world.

A new kibana dashboard and few queries have been designed in order to retrieve analytics from the dataset. The new kibana dashboard is also available in the delivery folder under the name of `KibanaDashboardWorldwide.ndjson`, this dashboard can be easily imported by applying the index name "c19-data-worldwide".

## 4.1 Dataset structure

In the following table describes the dataset schema and data fields:

| Field name | Data Type | Description |
|---|---|---|
| **location** | String | The name of the state |
| **iso_code** | String | The international ISO code of the state |
| **date** | Datetime | Administration date of the vaccines |
| **total_vaccinations** | Integer | Total number of vaccination administered in the state |
| **people_vaccinated** | Integer | Total number of people administered with at least one dose of the vaccine in the state |
| **people_fully_vaccinated** | Integer | Total number of people who completed the vaccination cycle in the state |
| **total_boosters** | Integer | Total number of people administered with the booster dose in the state |
| **daily_vaccinations** | Integer | Number of doses administered daily |
| **total_vaccinations_per_hundred** | Double | Percentage of people vaccinated with at least one dose of the vaccine |
| **people_vaccinated_per_hundred** | Double | Percentage of people vaccinated with only one dose of the vaccine |
| **people_fully_vaccinated_per_hundred** | Double | Percentage of people vaccinated with two doses of the vaccine |
| **total_boosters_per_hundred** | Double | Percentage of people vaccinated with the booster dose of the vaccine |
| **daily_vaccinations_per_million** | Integer | Percentage *over million* of people vaccinated with at least one dose of the vaccine |
| **daily_people_vaccinated** | Integer | Number of people vaccinated daily with at least one dose of the vaccine |
| **daily_people_vaccinated_per_hundred** | Double | Percentage of people vaccinated daily with at least one dose of the vaccine |

## 4.2 Worldwide additional queries

1. This query returns the number of states present in the dataset

```
GET c19-data-worldwide/_search
{
  "size": 0,
  "aggs": {
          "number_of_states": {
                  "cardinality": {
                          "field": "location.keyword"
                  }
          }
  }
}
```

2. This query returns the number of records for each state

```
GET c19-data-worldwide/_search
{
  "size": 0,
  "aggs": {
          "records_per_state": {
                  "terms": {
                          "field": "location.keyword"
                  }
          }
  }
}
```

3. This query returns the maximum number of daily vaccinations reached in every state

```
{
  "size": 0,
  "aggs": {
   "records_per_state": {
      "terms": {
        "field": "location.keyword"
      },
      "aggs": {
        "highest_vaccination_number": {
          "max": {
            "field": "daily_vaccinations"
          }
        }
      }
    }
  }
}
```

# 5 Cassandra implementation

An additional implementation of the solution to the given problem has been realised using **Cassandra**, an alternative `noSql` platform.

The following sections describe the creation of the Cassandra's tables and the dataset storing. Three queries were also designed to show some basics features.

## 5.1 Keyspace and Table creation

In order to create the keyspace the following command has been used:

```
CREATE KEYSPACE Vaccinations_data WITH
    replication = {'class': 'SimpleStrategy', 'replication_factor': 1};
```

In order to select the `Vaccinations_data` keyspace it's necessary to prompt the command:

```
USE Vaccinations_data;
```

The next command has been used to create the Table:

```
CREATE TABLE Vaccinations
(Administration_Date date, Supplier text, Area text, Age_Group text,
Male_Count int, Female_Count int, First_Doses int, Second_Doses int,
Post_Infection_Doses int, Booster_Doses int, NUTS1 text, NUTS2 text,
ISTAT_Code int, Region text,
PRIMARY KEY(Area, Supplier, Administration_Date, Age_Group));
```

The table `PRIMARY KEY` contains the `Partition Key` **Area**, so that the data is correctly partitioned and organized within the cassandra nodes by region.

## 5.2 Storing the Dataset

In order to populate the DB the following command was used:

```
COPY Vaccinations
(Administration_Date, Supplier, Area, Age_Group, Male_Count,
Female_Count, First_Doses, Second_Doses, Post_Infection_Doses,
Booster_Doses, NUTS1, NUTS2, ISTAT_Code, Region)
FROM '"C:\Users\Pc\Desktop\somministrazioni-vaccini-latest.csv"'
WITH DELIMITER=',' AND HEADER=TRUE;
```

Figure 1: Note that the file path has to be changed in order to correctly update the data

## 5.3 Queries

1. The first proposed query retrieves the vaccinations administered in the "Abruzzo" region:

```
SELECT *
FROM Vaccinations
WHERE area = 'ABR' AND Supplier = 'Moderna';
```

2. This query returns the number of first doses administered in the "Lombardia" region and supplied by "Pfizer/Biontech":

```
SELECT sum(First_Doses)
FROM Vaccinations
WHERE area = 'LOM' AND Supplier = 'Pfizer/BioNTech';


---------output---------

system.sum(first_doses)
------------------------
            5415450

(1 rows)
```

3. This query returns the number of vaccines administered in the "Piemonte" differentiating between male and female population:

```
SELECT sum(Male_Count), sum(Female_Count)
FROM Vaccinations
WHERE area = 'PIE';


---------output---------

system.sum(male_count)  | system.sum(female_count)
------------------------+-------------------------
                3772889 |                  4067812

(1 rows)
```

# 6 Team composition and Sources

## 6.1 Team Composition

The project was realized by:

- Curti Gabriele, 10624502
- Cutrupi Lorenzo, 10629494
- Samuele Mariani, 10622653
- Alessandro Molteni, 10623928
- Matteo Monti, 10622780

## 6.2 Sources

- Slides from the lessons and exercise session.
- elastic.co/elasticsearch documentation for Elastic Search and Kibana.