Article

Study of the impact of vaccination on the COVID-19 pandemic and the stocks of vaccine producing companies

António Santos [1,‡], Manuel Moreira [1,‡], Maria Jordão [1,‡] and Sara Dias [1,‡]

[1] University of Minho; antoniomsantos99@gmail.com (A.M.); manuelmoreira2000@gmail.com (M.M.); sofiamarques2408@gmail.com (M.J.); sarah.dias@outlook.pt (S.D.)

‡ These authors contributed equally to this work.

Abstract: The amount of data available nowadays only keeps getting bigger and more diverse, meaning that there is huge potential in studying it's real 'value' that will help uncover useful and hidden knowledge in this data. This paper intends to develop a scientific article, as well as the implementation and study of a Big Data system, and define a well reasoned and discussed architecture pipeline. Throughout this article, we will describe the work developed, including the description of a dataset given to us by our teachers, the two complementary datasets chosen by us and the definition of the use cases. In addition, we will choose several Big Data tools, studied previously by each member of the group, that we think are the most appropriate for the defined use cases, and the development of an architecture (including the chosen "tools") that, according to the final dataset, can achieve the desired results. Our use cases are the study of the impact of vaccination on the COVID-19 pandemic and the study of the impact of vaccination on the stocks of vaccine producing companies. What we could conclude on its study is that vaccination had a really big impact on the number of deaths, but not so much on the number of cases. We also concluded that the vaccination only influenced the stocks of *Moderna*, being that it is also a biotech company, not just pharmaceutical.

Keywords: Big Data, dataset, data processing, data storage, data visualization, Big Data frameworks

Citation: Santos, A.; Moreira, M.; Jordão, M.; Dias, S. Study of the impact of vaccination on the COVID-19 pandemic and the stocks of vaccine producing companies. *Journal Not Specified* **2022**, *1*, 0. https://doi.org/

Received:
Accepted:
Published:

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

1. Introduction

The amount of data being collected by most companies today is growing exponentially, making it impossible for traditional relational databases to remain a satisfactory method of data storage. One of the main causes is reflected in the fact that relational databases are neither horizontally scalable nor able to handle unstructured data.

Thus, most businesses involving large volumes of data are making the transition to **NoSQL** databases. These were developed with the goal of working with large amounts of data, meeting all the requirements of *Big Data*.

In the case of the **COVID-19** pandemic, which affected all services and industries, we have at our disposal a gigantic amount of databases related to its impact. In this work, we have been asked to find several *datasets* related to the pandemic and to create use cases, which find a link between all of them so that their impact can be studied. To do this, **Big Data** tools will be used.

Since *datasets* related to vaccination and the actions of companies will be used, then we will study two different use cases. The first will study the impact of vaccination on the number of infected and deaths from **COVID-19** and the second will study the impact of the pandemic and vaccination on the stock market of the various companies that produced the vaccines, such as Pfizer, Moderna and Astrazeneca.

In order to carry out the article, it was necessary to define how the research for the *datasets* and the various *Big Data* tools would be done. First we will mention the methodology used in the *datasets* research, followed by the methodology used in the tools research.

The search for *datasets* was done in groups during the practical classes of the curricular unit, in order to find *datasets* that could be used in the individual and group work. For this, we searched various *websites*, such as **Our World in Data**, **Kaggle** and **Worldbank**, for *datasets* directly related to the **COVID-19**, as in the *dataset* of vaccinations, but also by *datasets* that in a more discrete way could be related to the theme, as it happened with the *dataset* of the companies' shares.

For the research of *Big Data* tools, the work was done differently. First of all, it was a research done individually in order to create a ready-made architecture to gather, analyze and present the data needed to study both use cases. To do this, it was necessary to find tools that would handle the storage, processing, and visualization of the data.

With this goal in mind, research was done on the best tools for *Big Data*, individual research on each tool, and also comparisons between the various tools to study which would be the most efficient for the specific use cases.

After the various individual tasks were completed, it was necessary to bring together the knowledge that each member had acquired so that we could work together as a group. For this, we discussed which were the best tools to use and which was the best approach, within the architectures that each one defined.

2. Materials and Methods

2.1. Datasets study

2.1.1. Datasets decription

As mentioned earlier, a *dataset* was given regarding the number of cases and daily deaths from **COVID-19** in the various countries of the world, taken from the official *website* of the **World Health Organization – WHO**.

This *dataset* consists of 8 columns, which are as follows:

Table 1. Description of the first *dataset*

Column	**Description**	**Category**
Date_reported	a data referente à informação recolhida	Date
Country_code	the country code for the collected information	Categorical
Country	the country for the collected information	Categorical
WHO_region	region where the country is located, according to WHO	Categorical
New_cases	number of new cases of COVID-19 on the day under study	Numeric
Cumulative_cases	total number of COVID-19 cases up to the day in study	Numeric
New_deaths	number of new deaths by COVID-19 on the day in study	Numeric
$Cumulative_deaths$	total number of deaths by COVID-19 until the day under study	Numeric

To complement this *dataset*, it was necessary to find two other *datasets* with which it was possible to create a *use case*. For this, a *dataset* with data on the shares of companies that produced vaccines against the **COVID-19** and another *dataset* with data on vaccination by country was chosen.

Starting with the *dataset* concerning vaccination worldwide, including doses administered daily, it was found on the *website* **Our World in Data**. It contains 16 columns, which are detailed below:

It is worth noting that this *dataset* does not contain complete daily information, as for some countries some columns are only updated weekly.

The second complementary *dataset* contains information regarding the shares of the companies that produced the vaccines, which resulted from various data compiled from **Yahoo Finance**. This consists of 8 columns, which are as follows:

It is worth noting that the *stocks* market is closed at the weekends, so no information is available on those days.

2.1.2. Use cases definition

After transforming the *datasets* into a final *dataset*, the use cases to be studied were then defined. As mentioned before, the first use case is the study of the impact of vaccination on

Table 2. Description of the second *dataset*

Column	**Description**	**Category**
location	country or region referring to the data	Categorical
iso_code	ISO region code	Categorical
date	date information was collected	Date
total_vaccinations	total number of doses administered to date	Numeric
total_vaccinations_per_hundred	*total_vaccinations* per 100 inhabitants	Numeric
daily_vaccinations_raw	doses administered on the day in question (raw data)	Numeric
daily_vaccinations	doses administered on the day in question	Numeric
daily_vaccinations_per_million	*daily_vaccinations* per million inhabitants	Numeric
people_vaccinated	total number of people with at least one dose	Numeric
people_vaccinated_per_hundred	*people_vaccinated* per 100 inhabitants	Numeric
people_fully_vaccinated	total number of people with complete vaccination	Numeric
people_fully_vaccinated_per_hundred	*people_fully_vaccinated* per 100 inhabitants	Numeric
total_boosters	total number of booster doses administered	Numeric
total_boosters_per_hundred	*total_boosters* per 100 inhabitants	Numeric
daily_people_vaccinated	daily number of people to receive the first dose	Numeric
daily_people_vaccinated_per_hundred	*daily_people_vaccinated* per 100 inhabitants	Numeric

Table 3. Description of the third *dataset*

Column	**Description**	**Category**
Company	company related data	Categorical
Date	date regarding the data	Date
Open	stock price at the beginning of the *trading day*	Categorical
High	highest stock price on the day in question	Categorical
Low	lowest stock price on the day in question	Numeric
Close	stock price at the close of the exchange	Numeric
Adj Close	adjusted value of the shares at the close of the exchange	Numeric
Volume	number of actions moved	Numeric

the number of infected and deaths by the **COVID-19**. The second use case is the study of the impact of **COVID-19** and, mostly, of vaccination on the shares of the companies that produced the vaccines.

2.2. State of the art

In order to later be able to properly study the chosen case studies, it is necessary to make a choice of appropriate *big data* tools. For this purpose, a choice was made on several types of tools that deal with data processing, storage and/or visualization, within all those that were researched by the group members in their individual work. For the case of data processing we chose Apache Spark, for data storage we chose MongoDB and for data visualization we chose PowerBI.

2.2.1. Apache Spark

Apache Spark is an open-source framework that can rapidly process large data sets (Big Data) and distribute these tasks across multiple systems to ease the workload. Spark supports data streaming, graph processing, and machine learning techniques.

Some applications of Apache Spark include *Machine Learning*, due to its ability to store data in memory and execute queries repeatedly, which reduces the time needed to determine the best possible solution. Another application of Spark is **data integration**, and it is a very efficient tool for performing ETL operations on data. This means that it performs, extracts, transforms and loads operations to extract data from different sources, cleans it up and organizes it, making it ready to be loaded into another system for analysis.

In addition, Spark also features **interactive analysis**, through which users can perform real-time data analysis with the help of *Structured Streaming*. You can also run interactive queries in a live *web* session. Finally, Spark includes *Fog computing*, offering components such as *Spark Streaming*, *GraphX* and *MLlib*, which deal with processing data and decentralizing its storage across networks of interconnected devices and users, which need a distributed parallel processing system.

Apache Spark's many advantages include its speed and the fact that it can handle multiple *workloads*. In addition, it is very easy to use, offers support for various languages such as Scala, Java, Python, and R, and is very efficient. Finally, Spark has a large support community, it is one of the most used *open source* tools, and, as mentioned before, it can stream data in real time.

2.2.2. MongoDB

MongoDB is a **NoSQL** database written in C, C++ and JavaScript, which serves as an advanced alternative to current databases. It is one of the best *big data* analysis tools for working with *datasets* that vary and change frequently or are semi-structured or unstructured. It is a database used for large volumes of data, using documents and collections instead of traditional rows and columns. Documents consist of *key-value* pairs and collections have *sets* of documents.

Some applications of MongoDB include storing data from mobile applications, content management systems, product catalogs, and more. Its main features include aggregation, indexing, replication, *capped collections*, *ad-hoc queries*, *sharding*, *load balancing*, file storage, etc. MongoDB is used by Facebook, eBay, Google, and many others.

In terms of advantages, it is an easy-to-learn tool that supports many platforms and operating systems, it is reliable, fast and low cost. It is a very flexible tool, for storing data in documents, with support for many ad-hoc queries, such as searching by *field name*, by regular expressions or *queries* by range. In addition, all fields in the document can be indexed to improve the quality of searches. It is a great tool in load balancing, since it splits the data between the MongoDB instances, also duplicating the data, taking into account possible errors that may be made. MongoDB stores data of any type, including integers, *strings*, *booleans*, *arrays* and objects.

On the other hand, MongoDB can be slow in some cases and has limited analytics.

2.2.3. PowerBI

PowerBI is a data visualization tool from Microsoft that allows you to explore and extract information about data. It can help us get quick answers from data and also allows real-time data mapping and analysis. PowerBI handles almost any kind of data, such as data from *streaming*, *cloud services*, Excel spreadsheets and many others.

PowerBI is considered one of the best data visualization tools and is being used in all industries such as finance, sales and operations. The tool can be used for free, allowing you to analyze up to 1GB of data without a paid subscription.

PowerBI is made up of several components, each serving a certain purpose. The *Power Query* is a data linking tool that allows you to transform, combine and enhance data from various sources. The *Power Pivot* is a data modeling tool for creating data models. The *Power View* is a data visualization tool that generates interactive charts, graphs, maps and other visual elements. The *Power Map* is another visualization tool for creating immersive 3D images and the *Power QA* is a question and answer engine that allows you to ask questions about the data in plain language.

Some advantages of PowerBi include the fact that it integrates seamlessly with other existing applications, adopting analytical and reporting capabilities; or the existence of custom *dashboards*, which turns out to be one of its biggest advantages.

In addition, it has no memory or speed restrictions, ensuring that data is quickly analyzed; and it is a very intuitive, simple and easy-to-use tool, so you don't need to be an expert in the field to be able to use it.

3. System developed

With the study of the *datasets*, the definition of the use cases and the selection of the *Big Data* tools, we can now demonstrate the architecture adopted for the study of both use cases.

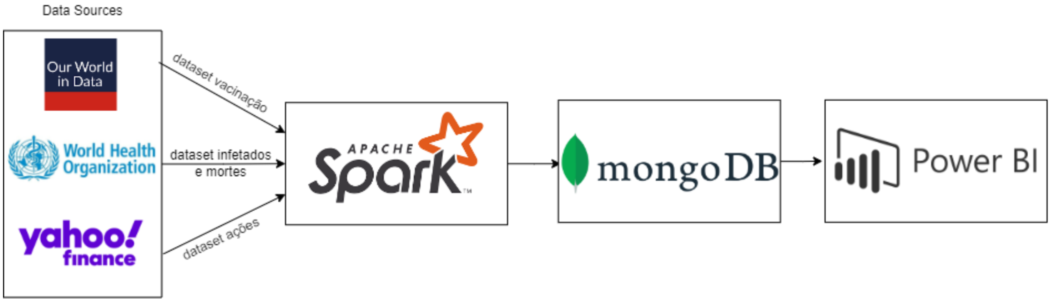


Figure 1. Defined architecture

As we can see in the figure above, the pipeline starts with the *data sources*, which in this case are the **Our World in Data**, the **Yahoo Finance** and the **World Health Organization**. These sources provide three *datasets* from which the data will be taken, and passed to the *data processing system*, **Apache Spark**, where it will be processed. This is where the data from the three *datasets* will be combined into one, creating the final *dataset*.

Looking at the base *dataset* and the one with the shares, we know that both have a column designated for the date regarding the data collected. Since in the first *dataset* the column is named "*Date_reported*" then we need to rename it to just "*Date*". Thus, we can proceed to *join* the first two *datasets* by the column referring to the date, resulting in a single *dataset* that will have daily information about the number of cases, deaths and information about the actions of each of the companies.

Looking now at the *dataset* for vaccinations, we know that it also has the column referring to the date of data collection, so we can also give the *join* by date. However, the *dataset* also contains information about the country in question, as did the first *dataset*. So, we can try to join this *dataset* with the previous one, with a *join* by the *"Date"* and *"Country"* columns.

Below is a summary diagram of the creation of the final *dataset* structure.

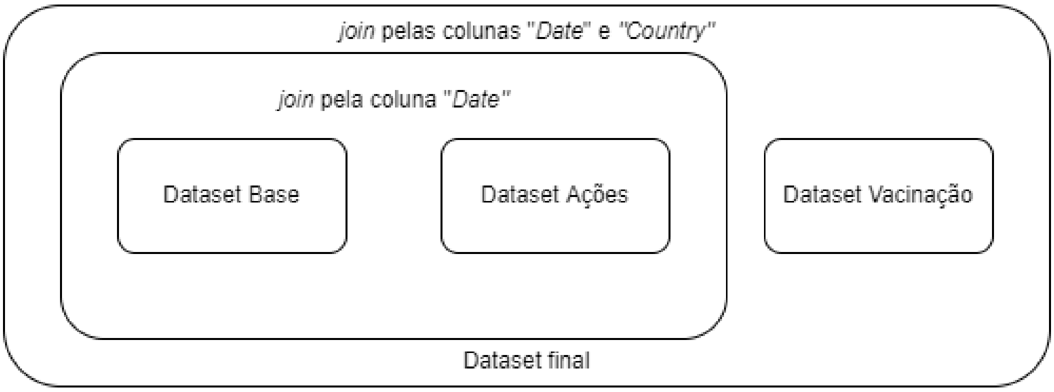


Figure 2. Structure of the final dataset

Next, the data is stored in the **MongoDB** and then it is sent to **PowerBI**, where they can be visualized. At this stage various types of graphs can be generated, and from the analysis of these, it will be possible to extract information, such as the identification of patterns or significant changes in the data, that can be related to the **COVID-19** pandemic. With the use of graphs we will be able to visualize and interpret the data in order to respond to our use cases.

4. Results

For our work, we decided to use **Google Colab** for data processing, with the support of Apache Spark, and for data storage, MongoDB.

4.1. Apache Spark

In order to be able to use Apache Spark within the Python and Google Colab environment we need the following dependencies:

- Java
- Apache Spark + Hadoop
- Pyspark

The following block of code downloads and installs the necessary dependencies in Google Colab, and also includes the code for importing the dependencies.

```
!apt-get install openjdk-8-jdk-headless -qq > /dev/null
!wget -q https://dlcdn.apache.org/spark/spark-3.2.1/spark-3.2.1-bin-hadoop3.2.tgz
!tar xf spark-3.2.1-bin-hadoop3.2.tgz
!pip install -q findspark
!pip install pyspark


from  pyspark.sql.functions import input_file_name
from pyspark.sql.types import *
import findspark
import pyspark

from pyspark.sql import SparkSession, SQLContext
from pyspark import SparkConf, SparkContext
```

After the import is done we finally move on to data processing. In order to use Apache Spark, it is necessary to create a *SparkSession* within our program. This object contains all the necessary settings for this *pipeline* step to work, as well as the necessary information that allows us to later store the data in MongoDB.

```
conf = SparkConf().set("spark.jars.packages",
"org.mongodb.spark:mongo-spark-connector_2.12:3.0.1")
sc = SparkContext(conf=conf)

spark = SparkSession.builder\
        .master("local")\
        .appName("Colab")\
        .config('spark.ui.port', '4050')\
        .config("spark.mongodb.input.uri",
"mongodb://localhost:27017/bigData.dados") \
        .config("spark.mongodb.output.uri",
"mongodb://localhost:27017/bigData.dados") \
        .getOrCreate()
```

To be able to read the *datasets*, it is necessary that they are present in our machine, and for that we have to download them so that Google Colab can access the data.

Note that the *datasets* of cases and vaccinations against **COVID** are up to date until June 16, 2022, while the *stocks* does not contain completely updated information.

```
!mkdir dataset

!wget -P dataset/ https://raw.githubusercontent.com/owid/covid-19-data/-
master/public/data/vaccinations/vaccinations.csv
!wget -P dataset/ https://covid19.who.int/WHO-COVID-19-global-data.csv
!wget -P dataset/ https://cdn.discordapp.com/attachments/946357765754982440/-
948939040835657728/stocks.csv
```

With the *datasets* present we can start importing them into Apache Spark.

To ensure correct storage, we need to define an architecture (or *Schema*) for each *dataset*. This architecture contains the legend of each column as well as the type of data stored in it.

```
customSchema_CoVIDCases = StructType([ \
StructField("Date_Reported", DateType(), True), \
StructField("Country_code", StringType(), True), \
StructField("Country", StringType(), True), \
StructField("WHO_region", StringType(), True), \
StructField("New_cases", IntegerType(), True), \
StructField("Cumulative_cases", IntegerType(), True), \
StructField("New_deaths", IntegerType(), True), \
StructField("Cumulative_deaths", IntegerType(), True), \
])

customSchema_CoVIDVaccinations = StructType([ \
StructField("location", StringType(), True), \
StructField("iso_code", StringType(), True), \
StructField("date", DateType(), True), \
StructField("total_vaccinations", IntegerType(), True), \
StructField("people_vaccinated", IntegerType(), True), \
StructField("people_fully_vaccinated", IntegerType(), True), \
StructField("total_boosters", IntegerType(), True), \
StructField("daily_vaccinations_raw", IntegerType(), True), \
StructField("daily_vaccinations", IntegerType(), True), \
StructField("total_vaccinations_per_hundred", FloatType(), True), \
StructField("people_vaccinated_per_hundred", FloatType(), True), \
StructField("people_fully_vaccinated_per_hundred", FloatType(), True), \
StructField("total_boosters_per_hundred", FloatType(), True), \
StructField("daily_vaccinations_per_million", FloatType(), True), \
StructField("daily_people_vaccinated", IntegerType(), True), \
StructField("daily_people_vaccinated_per_hundred", FloatType(), True)
])

customSchema_stocks = StructType([ \
StructField("Company", StringType(), True), \
StructField("Date", DateType(), True), \
StructField("Open", FloatType(), True), \
StructField("High", FloatType(), True), \
StructField("Low", FloatType(), True), \
StructField("Close", FloatType(), True), \
StructField("Adj Close", FloatType(), True), \
StructField("Volume", IntegerType(), True), \
])
```

Now, importing the *datasets* with the *schemas*, we have:

```
df_CoVIDCases = spark.read.format("csv") \
   .option("header", "true") \
   .option("sep",",") \
   .schema(customSchema_CoVIDCases) \
   .load('/content/dataset/WHO-COVID-19-global-data.csv')

df_CoVIDVaccinations = spark.read.format("csv") \
   .option("header", "true") \
   .option("sep",",") \
   .schema(customSchema_CoVIDVaccinations) \
   .load('/content/dataset/vaccinations.csv')

df_stocks = spark.read.format("csv") \
   .option("header", "true") \
   .option("sep",",") \
   .schema(customSchema_stocks) \
   .load('/content/dataset/stocks.csv')

print("###CASOS###")
df_CoVIDCases.printSchema()
df_CoVIDCases.show(n=10)
print(df_CoVIDCases.count())
print("###VACINACOES###")
df_CoVIDVaccinations.printSchema()
df_CoVIDVaccinations.show(n=10)
print(df_CoVIDVaccinations.count())
print("###STOCKS###")
df_stocks.printSchema()
df_stocks.show(n=10)
```

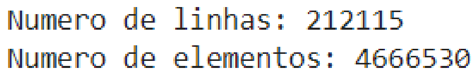
With the importation of the *datasets* done, we can then begin the study of their joins, starting by joining the base *dataset* with that of the vaccinations.

In order to facilitate the process we can equalize the name of the columns whose information contained in both *datasets* is the same. In this case, as we mentioned in the previous chapter, we will rename the *"Date_reported* column of the first *dataset* to just *"Date"*, just as in the *dataset* of vaccinations we will need to rename the *"date"* column to *"Date"*. However, Pyspark is not case sensitive and will therefore recognize the column regardless of this change. Finally, still in the *dataset* of vaccinations, we have to rename the *"location"* table to *"Country"*.

Next, we join both *datasets* using a *left outer join*, in order to maintain the format of the first *dataset*.

```
df_CoVIDCases = df_CoVIDCases.withColumnRenamed("Date_Reported","Date")
df_CoVIDVaccinations = df_CoVIDVaccinations.withColumnRenamed("location",
"Country")

```

```
partialDataset = df_CoVIDCases.join(df_CoVIDVaccinations,["Country","Date"],
'left')
partialDataset.show(n=10)
print(f"Numero de linhas: {partialDataset.count()}")
print(f"Numero de elementos: {(partialDataset.count() *
len(partialDataset.columns))}")
```



```
Numero de linhas: 212115
Numero de elementos: 4666530
```

Figure 3. Result of the *prints* for the number of lines

It is worth remembering that there are several lines in the *dataset* that do not contain any pertinent information related to the *dataset* of vaccinations. This is due to the fact that the introduction of the vaccine did not happen at the same time as the appearance of cases, so there is a large time interval where there is only information regarding the first *dataset*. In addition, once again we remind you that this *dataset* does not contain complete daily information, because for some countries, some columns are only updated weekly.

To deal with these *missing values* from the *dataset*, we will replace these same values with those of the closest preceding non-null element. However, this method will eventually cause problems since the information in the last rows for a certain country may contaminate the rows of another country if the first row of the latter contains nulls.

In order to get around this problem we will replace all the null values in the first row for each country with 0, which is factually correct since there was no vaccination on those dates. To achieve this goal we filter these rows, replace the values and join the filtered rows to the original *dataset*, using the **textitunion** table.

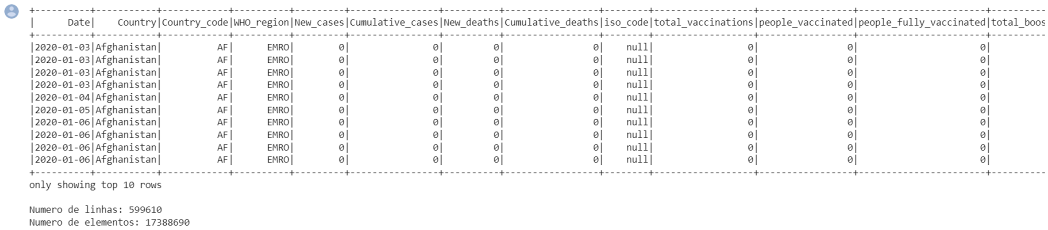
```
from pyspark.sql.functions import last,col, row_number
from pyspark.sql.window import Window
import sys

w2 = Window.partitionBy("Country").orderBy(col("Date"))
df=partialDataset.withColumn("row",row_number().over(w2)) \
  .filter(col("row") == 1).drop("row") \
  .fillna(0)

w2 = Window.partitionBy("Country").orderBy(col("Date"))
partialDataset=partialDataset.withColumn("row",row_number().over(w2)) \
  .filter(col("row") != 1).drop("row")

partialDataset = partialDataset.union(df)
```

With the merge executed, we can replace the null values of the entire *dataset* using the method mentioned earlier.

```
window_last = Window.orderBy(["Country","Date"])
for column in partialDataset.columns:
  partialDataset = partialDataset.withColumn(column, last(column,
ignorenulls=True).over(window_last))

partialDataset.show(n=10)
```

To join the third *dataset*, the process used was the same. 229

```
finalDataset = partialDataset.join(df_stocks,["Date"],'left')
finalDataset.show(n=10)

print(f"Numero de linhas: {finalDataset.count()}")
print(f"Numero de elementos: {(finalDataset.count() *
len(finalDataset.columns))}")
```



```
+----------+-----------+------------+----------+---------+----------------+----------+-----------------+--------+------------------+-----------------+-----------------------+----------
|      Date|    Country|Country_code|WHO_region|New_cases|Cumulative_cases|New_deaths|Cumulative_deaths|iso_code|total_vaccinations|people_vaccinated|people_fully_vaccinated|total_boos
+----------+-----------+------------+----------+---------+----------------+----------+-----------------+--------+------------------+-----------------+-----------------------+----------
|2020-01-03|Afghanistan|          AF|      EMRO|        0|               0|         0|                0|    null|                 0|                0|                      0|
|2020-01-03|Afghanistan|          AF|      EMRO|        0|               0|         0|                0|    null|                 0|                0|                      0|
|2020-01-03|Afghanistan|          AF|      EMRO|        0|               0|         0|                0|    null|                 0|                0|                      0|
|2020-01-03|Afghanistan|          AF|      EMRO|        0|               0|         0|                0|    null|                 0|                0|                      0|
|2020-01-04|Afghanistan|          AF|      EMRO|        0|               0|         0|                0|    null|                 0|                0|                      0|
|2020-01-05|Afghanistan|          AF|      EMRO|        0|               0|         0|                0|    null|                 0|                0|                      0|
|2020-01-06|Afghanistan|          AF|      EMRO|        0|               0|         0|                0|    null|                 0|                0|                      0|
|2020-01-06|Afghanistan|          AF|      EMRO|        0|               0|         0|                0|    null|                 0|                0|                      0|
|2020-01-06|Afghanistan|          AF|      EMRO|        0|               0|         0|                0|    null|                 0|                0|                      0|
|2020-01-06|Afghanistan|          AF|      EMRO|        0|               0|         0|                0|    null|                 0|                0|                      0|
+----------+-----------+------------+----------+---------+----------------+----------+-----------------+--------+------------------+-----------------+-----------------------+----------
only showing top 10 rows

Numero de linhas: 599610
Numero de elementos: 17388690
```

Figure 4. Result of joining the three *datasets*

As we can see, the number of rows and elements has increased considerably, compared 230 to the two *datasets* (which can be seen above). However, most of this new information is 231 redundant. Most of the rows in the *dataset* have been quadrupled to join the information 232 for each of the four companies present in the *dataset* of the *stocks*. 233

In order to solve this problem we can "rotate" the *dataset* of the *stocks* so that the 234 information for each company is stored by columns and not by rows. 235

The following code snippet "rotates" the *dataset* of the *stocks*, saving it in a new CSV 236 file. 237

```
with open("dataset/stocks.csv",'r') as f:
    dataset = f.readlines()
    titulo = dataset[0].split(',')
    dataset=dataset[1:]

new_dataset = {}
for linha in dataset:
    info = linha.split(',')
    if info[1] not in new_dataset:
        new_dataset[info[1]] = {}
    for index in range(2,len(info)):
        new_dataset[info[1]][f"{info[0]}_{titulo[index]}"] = info[index]

with open("dataset/new_stocks.csv",'w') as f:
    f.write(f"date,{','.join(new_dataset['2020-01-13'].keys())}".replace("\n","")
+"\n")
    for key,value in new_dataset.items():
        f.write(",".join([key]+list(value.values())).replace('\n','')+'\n')
```

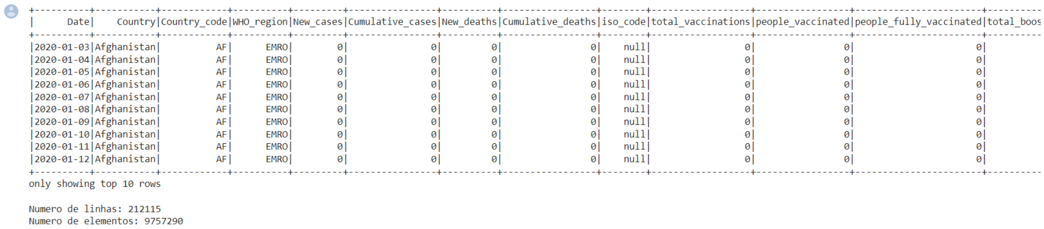
We now need to redefine the *schema* for the *stocks* and import it again. 238

```
customSchema_newstocks = StructType([ \
StructField("Date", DateType(), True), \
StructField("JNJ_Open", FloatType(), True), \
StructField("JNJ_High", FloatType(), True), \
StructField("JNJ_Low", FloatType(), True), \
StructField("JNJ_Close", FloatType(), True), \
StructField("JNJ_Adj Close", FloatType(), True), \
StructField("JNJ_Volume", IntegerType(), True), \
StructField("PFE_Open", FloatType(), True), \
StructField("PFE_High", FloatType(), True), \
StructField("PFE_Low", FloatType(), True), \
StructField("PFE_Close", FloatType(), True), \
StructField("PFE_Adj Close", FloatType(), True), \
StructField("PFE_Volume", IntegerType(), True), \
StructField("AZN_Open", FloatType(), True), \
StructField("AZN_High", FloatType(), True), \
StructField("AZN_Low", FloatType(), True), \
StructField("AZN_Close", FloatType(), True), \
StructField("AZN_Adj Close", FloatType(), True), \
StructField("AZN_Volume", IntegerType(), True), \
StructField("MRNA_Open", FloatType(), True), \
StructField("MRNA_High", FloatType(), True), \
StructField("MRNA_Low", FloatType(), True), \
StructField("MRNA_Close", FloatType(), True), \
StructField("MRNA_Adj Close", FloatType(), True), \
StructField("MRNA_Volume", IntegerType(), True), \
])

df_stocks = spark.read.format("csv") \
   .option("header", "true") \
   .option("sep",",") \
   .schema(customSchema_newstocks) \
   .load('/content/dataset/new_stocks.csv')

df_stocks.printSchema()
df_stocks.show(n=10)
```

Next, we put the various *datasets* back together, to confirm that the number of rows and elements has decreased as expected.

```
finalDataset = partialDataset.join(df_stocks,["Date"],'left')
finalDataset.show(n=10)

print(f"Numero de linhas: {finalDataset.count()}")
print(f"Numero de elementos: {(finalDataset.count() *
len(finalDataset.columns))}")
```

As we can see with the new *dataset* of *stocks*, the number of rows has dropped considerably, since we no longer have repeated rows. Although the *dataset* still contains some redundant information, in the *stocks* information, its abundance is more acceptable than in the previous solution.



```
+----------+-----------+------------+----------+---------+----------------+----------+-----------------+--------+------------------+-----------------+-----------------------+---------
|      Date|    Country|Country_code|WHO_region|New_cases|Cumulative_cases|New_deaths|Cumulative_deaths|iso_code|total_vaccinations|people_vaccinated|people_fully_vaccinated|total_boos
+----------+-----------+------------+----------+---------+----------------+----------+-----------------+--------+------------------+-----------------+-----------------------+---------
|2020-01-03|Afghanistan|          AF|      EMRO|        0|               0|         0|                0|    null|                 0|                0|                      0|
|2020-01-04|Afghanistan|          AF|      EMRO|        0|               0|         0|                0|    null|                 0|                0|                      0|
|2020-01-05|Afghanistan|          AF|      EMRO|        0|               0|         0|                0|    null|                 0|                0|                      0|
|2020-01-06|Afghanistan|          AF|      EMRO|        0|               0|         0|                0|    null|                 0|                0|                      0|
|2020-01-07|Afghanistan|          AF|      EMRO|        0|               0|         0|                0|    null|                 0|                0|                      0|
|2020-01-08|Afghanistan|          AF|      EMRO|        0|               0|         0|                0|    null|                 0|                0|                      0|
|2020-01-09|Afghanistan|          AF|      EMRO|        0|               0|         0|                0|    null|                 0|                0|                      0|
|2020-01-10|Afghanistan|          AF|      EMRO|        0|               0|         0|                0|    null|                 0|                0|                      0|
|2020-01-11|Afghanistan|          AF|      EMRO|        0|               0|         0|                0|    null|                 0|                0|                      0|
|2020-01-12|Afghanistan|          AF|      EMRO|        0|               0|         0|                0|    null|                 0|                0|                      0|
+----------+-----------+------------+----------+---------+----------------+----------+-----------------+--------+------------------+-----------------+-----------------------+---------
only showing top 10 rows

Numero de linhas: 212115
Numero de elementos: 9757290
```

Figure 5. Result of merging the three updated *datasets*

Before inserting the information into the database we can perform some more preprocessing in order to get cleaner data. We start by removing some unnecessary columns for our use case, which in this case are the columns *"iso_code"* and *"daily_vaccinations_raw*: 245 246 247

```
finalDataset=finalDataset.drop('iso_code','daily_vaccinations_raw')

finalDataset.show(n=10)
```

Previously, we replaced the null values in the *dataset* of vaccinations, and we now proceed to replace the null values in the *dataset* of *stocks*. Obviously, we cannot use the strategy of setting all values equal to 0, because this would mean that the companies lost all their value over the weekend, a very misleading piece of information. 248 249 250 251

Again, the solution we will use is to replace the null value with the closest non-null value to it. Although this method does not produce 100 percent correct data, it does show that the value of a company's *stocks* stays the same while the market is closed, which turns out to be a correct assumption to make. 252 253 254 255

We will also use the same strategy for the remaining columns since the *dataset* has already been correctly preprocessed. 256 257

```
window_last = Window.orderBy(["Country","Date"])
for column in finalDataset.columns:
  finalDataset = finalDataset.withColumn(column, last(column, ignorenulls=True)
.over(window_last))
```

4.2. MongoDB 258

Now that we have the data processed, we can start preparing the database storage, starting by installing **MongoDB** in Google Colab. 259 260

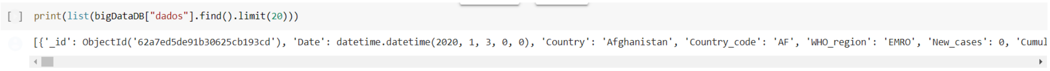
```
!apt install mongodb
!service mongodb start
```

Having MongoDB installed, we proceed to create a database and a collection to store the information. 261 262

```
from pymongo import MongoClient
client = MongoClient("mongodb://localhost:27017/")
bigDataDB = client["bigData"]
dados = bigDataDB["dados"]
```

Once the database is ready, all that is required is to import all the information that has been processed previously.

```
finalDataset.write.format("com.mongodb.spark.sql.DefaultSource").option(
"spark.mongodb.output.uri", "mongodb://127.0.0.1/bigData.dados").mode("append")
.save()
```



```
[ ] print(list(bigDataDB["dados"].find().limit(20)))

    [{'_id': ObjectId('62a7ed5de91b30625cb193cd'), 'Date': datetime.datetime(2020, 1, 3, 0, 0), 'Country': 'Afghanistan', 'Country_code': 'AF', 'WHO_region': 'EMRO', 'New_cases': 0, 'Cumul
```

Figure 6. Result of data warehousing

4.3. *PowerBI*

Having the data stored in MongoDB, we can now move on to the last phase of the work: the data visualization, using **PowerBI**.

Recalling our case studies, our goal is to study the influence of the Covid-19 pandemic on the stock market of vaccine developers as well as the study of the impact of vaccination on the number of people infected and dying from **COVID-19**.

First of all, it is necessary to mention that, for the sake of simplification of the analysis and explanation of the results given, we only focus our analysis on the data from Portugal, with the exception of the stock use case analysis.

Starting then with the first case study, several charts were developed with the support of the PowerBI tool for data analysis.

As we can see from the graph below, this shows the variation in the value of the shares of the companies *Johnson and Johnson, Moderna, Pfizer and Astrazeneca*, respectively, throughout the months of the year 2021. Along with this chart we have generated another that reflects the fluctuation of daily vaccinations over the months of 2021 worldwide. In this year, we can see that only the company *Moderna* suffered changes in the value of its shares, which peaked at the end of the summer, at the time of the peak of vaccinations that also happened during the summer.

We can see at first glance that when there is an increase in the number of vaccines administered daily, you can also see an increase in the value of the shares of *Moderna*. This can be explained because by the fact that this company is a biotechnology company that has developed a vaccine, so it is a company that has become highly valued in the market, suggesting that shareholders expect to see the company continue to innovate, as it has done with vaccines. However, we can see that the companies *Johnson and Johnson, Pfizer* and *Astrazeneca* have almost always maintained the value of their shares. This is because they are solely pharmaceutical companies (unlike *Moderna*), and so vaccines do not represent a large source of income for these companies, compared to, for example, drugs that they produce that are prescribed daily and represent a much larger profit than vaccines that have very limited doses.

Focusing our attention now on the second use case, the impact of vaccination on the number of people infected and dying from **COVID-19** we can see the following graphs:

The first graph represents the percentage of Portuguese people vaccinated partially or completely during 2021. In this graph we can see an almost linear increase in the number of people vaccinated with one dose between the months of March and August, reaching a *plateau* in the remaining months. In terms of the number of people completely vaccinated we can observe a similar behavior, where there is an increase from the month of March and September, then reaching a *plateau*. As we can read in the articles [**?**] [**?**] [**?**] we can justify these behaviors by the fact that, since the beginning of the vaccination process took place in December 2020, initially only priority groups, representing only a small portion of the Portuguese population (about 950 thousand people) were vaccinated. It was only in April 2021 that the 2nd vaccination phase began, opening the door to more than 2 million people,

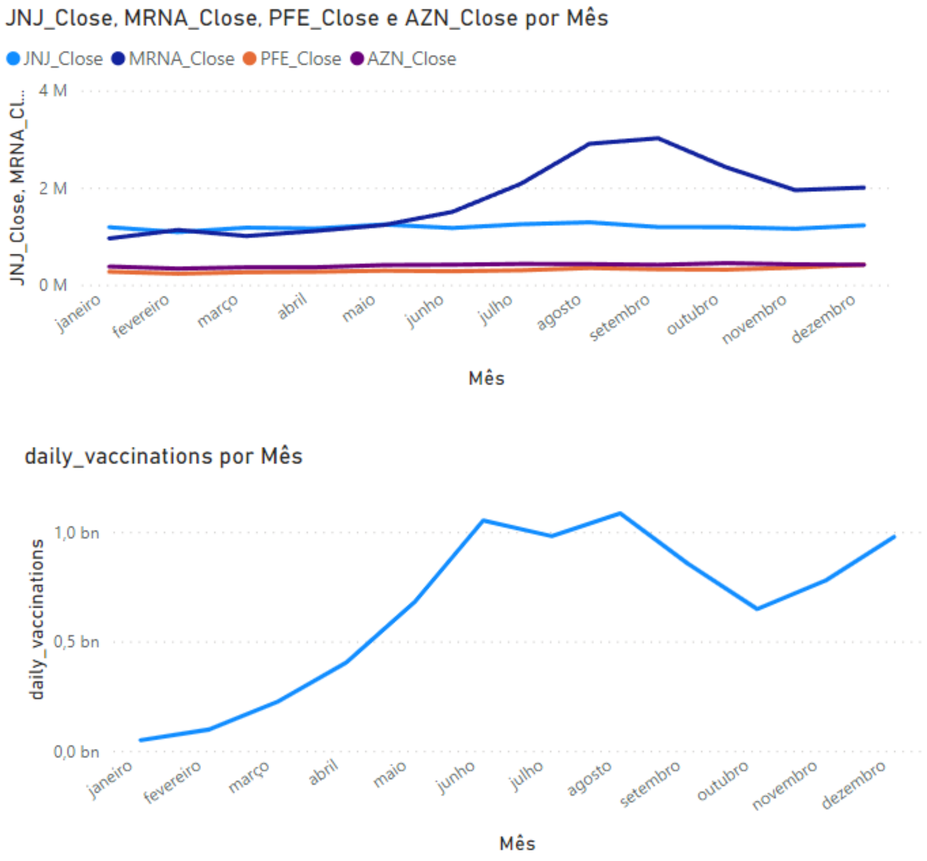


Figure 7. *Stock* of vaccine producers and daily vaccinations in 2021

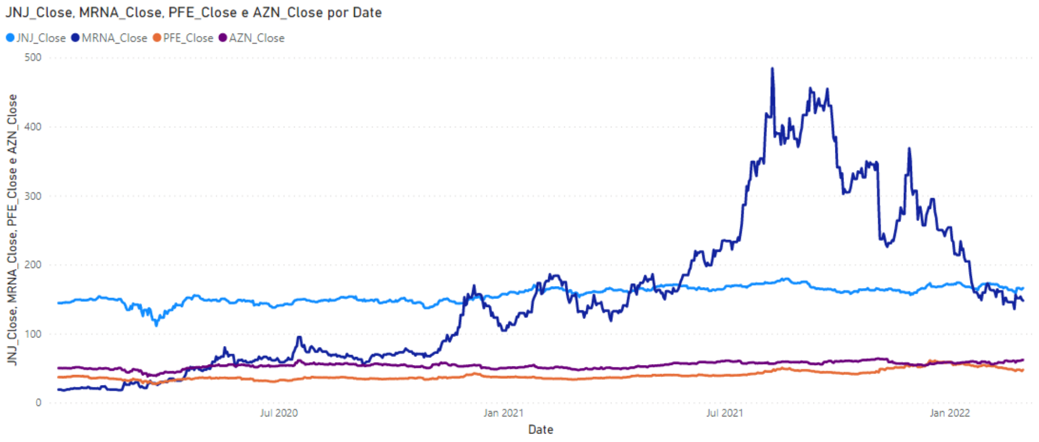


Figure 8. Fluctuation on the *stocks* of the companies

so it is possible to see a large increase in the number of people vaccinated compared with previous months. The number of people fully vaccinated also increases months later, since there is a mandatory waiting period between vaccine doses. Thus, after the first dose there is a delay in the line in the graph regarding the number of people fully vaccinated, because they have to wait a certain period of time before completing the vaccination plan. This can be seen by the lag between the curve of people vaccinated and the number of people completely vaccinated from January to April and August to September.

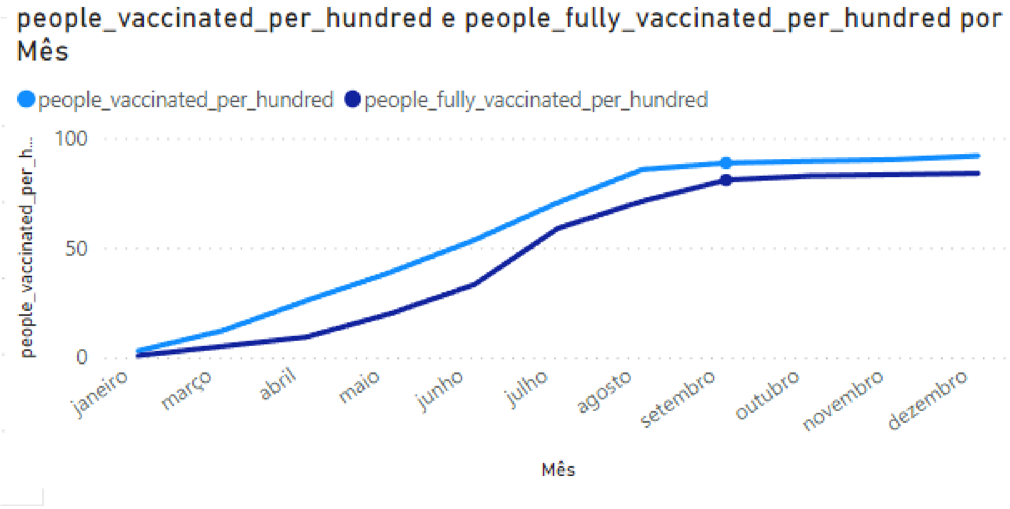


Figure 9. Percentage of Portuguese people vaccinated and fully vaccinated during 2021

A possible justification for the curve of fully vaccinated people being slightly inconstant lies in the fact that different vaccines have different waiting periods as well as different numbers of mandatory doses, hence we cannot expect linearity in the data.

Finally, starting in September we can then observe that we practically reach a *plateau*. This is due to the fact that at that instant around 87% of the Portuguese population had already been vaccinated, hence the rise becomes much more slight compared to what was observed before.

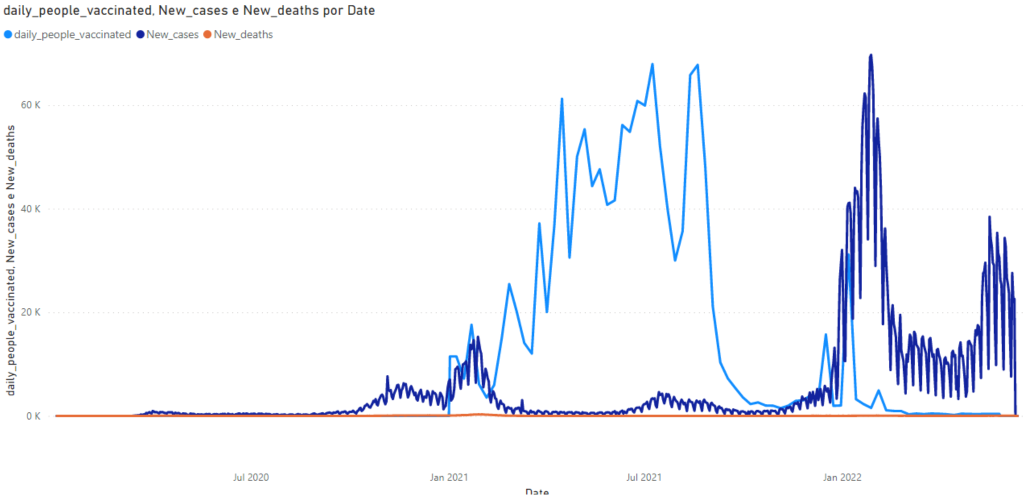


Figure 10. New cases, deaths and number of people vaccinated in Portugal in the last 3 years

From these two previous graphs we can then observe the variation between vaccinations and the number of new cases and deaths.

We can see that in the graph 9, the number of people vaccinated focuses mainly on the period between January 2021 and mid-September 2021, and there is no information about it much before January 2021 since vaccination only began in December 2020.

We can see that the number of cases before the start of vaccination never exceeded the 20,000 mark. As can be expected, it was the appearance of the first cases that led to the need for vaccination. We can see that during the vaccination period the number of cases was minimal, but nevertheless, we cannot say that this happened as a direct consequence of the vaccination, since in 2021 we experienced a period of general confinement in the first months of the year, as well as a set of measures to be taken in order to control contagion,

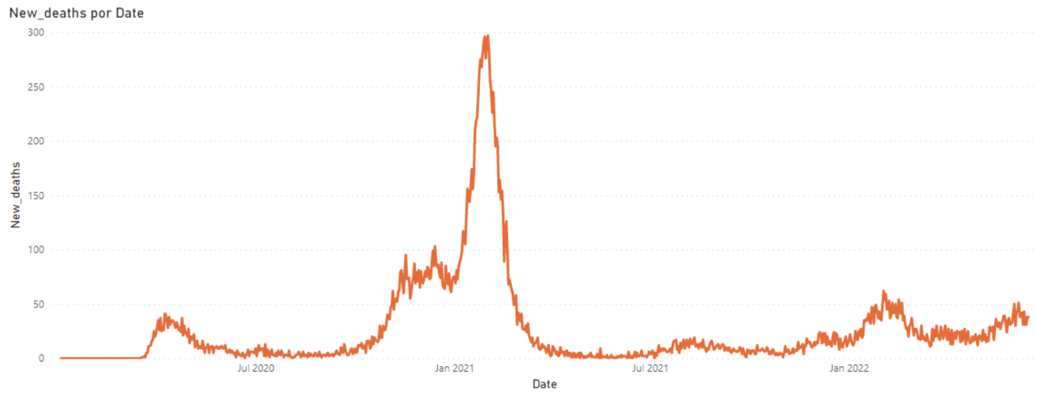


Figure 11. Number of deaths in Portugal in the last 3 years

such as masks and the banning of large events and nightclubs, which only began to be relieved at the end of the year.

Therefore, although it would be desirable, from the observed graphs, to conclude that the reason why the numbers of cases were minimal after vaccination was a direct consequence of it, the same cannot be concluded with such certainty. However, we can observe a steep decline after September 2021 and before January 2022 in the number of people vaccinated daily. This, as noted earlier, is due to the fact that the majority of the population was already vaccinated. However, this decrease did not result in an increase in cases. Nevertheless, there were still contingency and containment measures in place that prohibit us from making a direct connection between the two observed data. We can observe a slight increase in the number of cases in the month of December, since it is a festive season that leads to the grouping of people, increasing the possibility of contagion.

Something that we can also affirm, based on what has been observed, is a drastic increase in the number of cases in 2022. This is due to the fact that we have reached the end of confinement and the end of contingency measures, thus increasing the risk of contagion. So in a "blunt" way we can say that the number of people vaccinated and the number of new cases is not exactly related. Not only because a person can contract the virus more than once, but we also know that the vaccine only provides us with antibodies for a period of time. So based on what we have observed, there is very little we can say about the impact that vaccines have on new cases, especially since there are many other variables that play a role in contracting the virus.

In terms of the impact of vaccination on the number of deaths, we can see the graph 11. This graph is an enlargement of the graph above, since the previous scale did not favor the analysis of the number of deaths.

As we can see, the number of Covid-19 deaths increased dramatically from December 2020 to February 2021. In this period vaccination was still almost non-existent, so it is not yet possible to infer some kind of relationship between the two.

After February we can see a radical decrease in the number of deaths, which can be explained by the fact that the country is in general confinement until April, and older and more vulnerable people are starting to be vaccinated.

Between July and the end of the year 2021 we can see that the number of deaths practically didn't change, remaining always low, so we can deduce the existence of a causality between the fact that the majority of the population is already vaccinated, and the fact that they are more resistant to the virus.

A more interesting analysis can be made starting in the year 2022, where despite the number of cases increasing dramatically, the number of deaths, despite a slight increase in comparison to previous months, did not exceed the values that were observed at the beginning of the pandemic.

In other words, we can conclude that although the number of new cases of Covid-19 has reached record levels and we are now living without contingency rules and out of confinement, the number of deaths has not even come close to the pre-vaccination figures. This is something that would not be expected unless the vaccine actually had some kind of positive effect in protecting against the virus.

In summary, taking into consideration the use case regarding the impact of vaccination on the number of infected and deaths by the **COVID-19** and the analysis of the graphs obtained, we can conclude that the vaccine had little influence on the number of cases, although this conclusion is a bit contrived since many other variables played an active role in this analysis, such as confinement and contingency plans. As for the impact of vaccination on the number of deaths, we can deduce that the fact that the majority of the population was vaccinated made possible the drastic decrease in the number of deaths at a time when the number of cases reached record levels.

Finally, besides the graphs mentioned above, other interesting graphs were developed from the available data set, but these were not used for the explanation and justification of the use cases.

Independently, they still present information allusive to the theme, and can be visualized below:

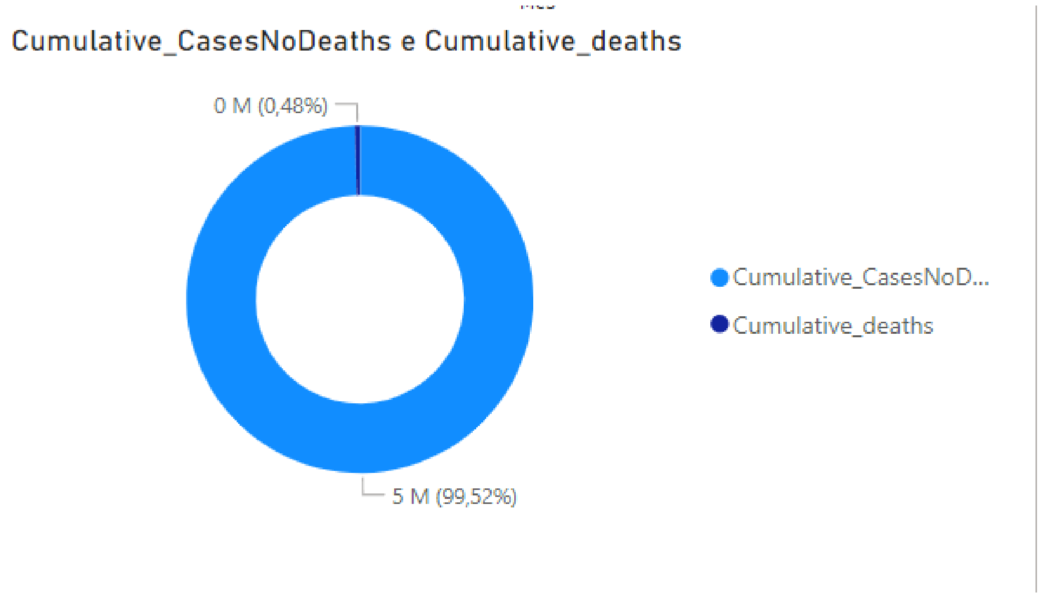


Figure 12. Comparison between the number of cases and deaths in Portugal

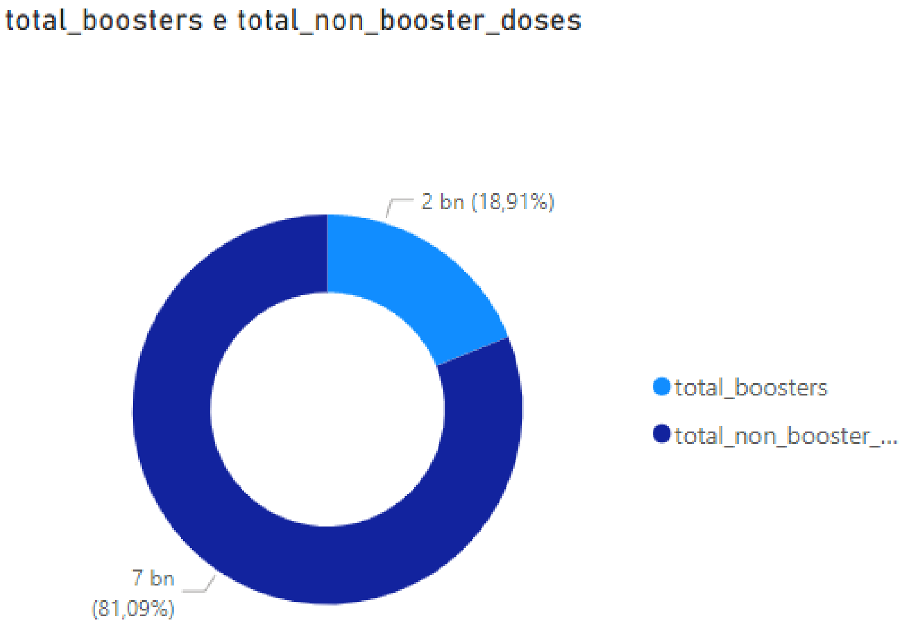


Figure 13. Comparison between the number of base and booster doses (*booster*)

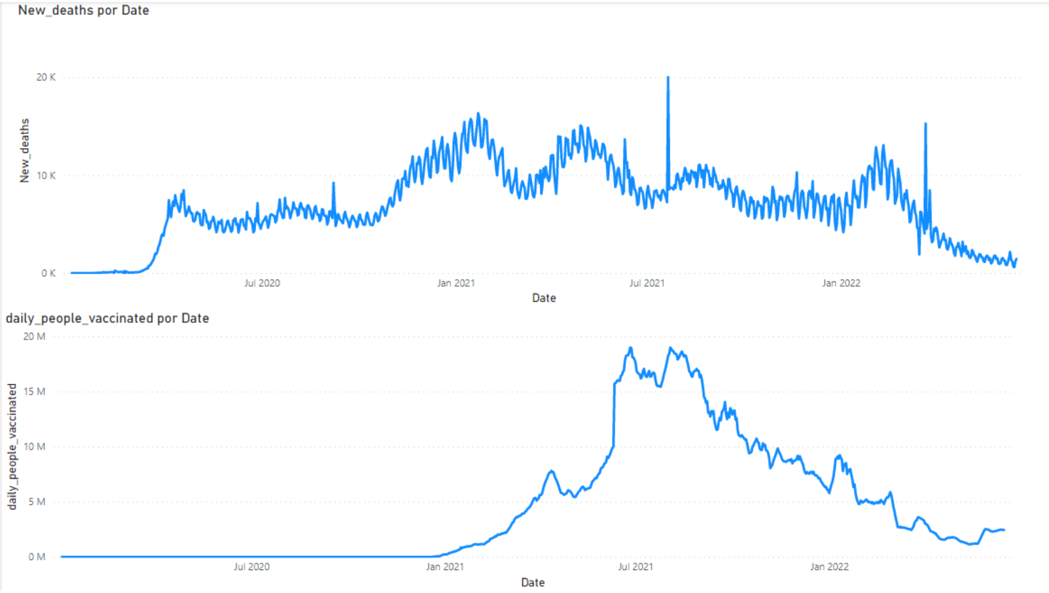


Figure 14. (1)Number of deaths in the last 3 years (2) Number of people vaccinated in the last 3 years

5. Conclusions

The objective of this paper was, based on the individual works developed by the group members, to implement a well grounded and discussed pipeline architecture and develop a supporting article for it.

For this paper, we took advantage of the research done earlier, during the individual papers, where we decided the various datasets we would use that were related to the COVID-19 pandemic. Also in the individual work, we defined use cases, which we decided to study in the group work, and we did research on the Big Data tools we could use, and made our most appropriate choice here. Once the best tools were chosen, we defined our architecture and proceeded to its implementation. After its implementation we performed analysis on the results obtained in order to answer the proposed use cases.

We concluded that, for the use case of the shares, vaccination only had an impact on the shares of *Modern*, the only biotechnology company, and that in the solely pharmaceutical companies its impact was very small, with the shares remaining constant for most of the time. For the use case of the impact of vaccination on the number of cases and deaths, we conclude that vaccination did not have much influence on the number of cases in the long term, with the peak of infections in Portugal being after vaccination. However, since vaccination started, the number of deaths never reached the pre-vaccination values, which shows its positive impact.

To conclude, we think that the realization of this work allowed the consolidation of concepts taught in the theoretical classes at university as well as allowed us to gain some experience on how to work with Big Data tools. Furthermore we believe that the objectives proposed by the teaching team were successfully met, leaving as future work a deeper analysis of the data where, in addition to Portugal, would be studied the impact of Covid-19 on a global scale and also the development of a dashboard to display the knowledge taken from the dataset.

6. Patents

This section is not mandatory, but may be added if there are patents resulting from the work reported in this manuscript.

Author Contributions: For research articles with several authors, a short paragraph specifying their individual contributions must be provided. The following statements should be used "Conceptualization, X.X. and Y.Y.; methodology, X.X.; software, X.X.; validation, X.X., Y.Y. and Z.Z.; formal analysis, X.X.; investigation, X.X.; resources, X.X.; data curation, X.X.; writing—original draft preparation, X.X.; writing—review and editing, X.X.; visualization, X.X.; supervision, X.X.; project administration, X.X.; funding acquisition, Y.Y. All authors have read and agreed to the published version of the manuscript.", please turn to the CRediT taxonomy for the term explanation. Authorship must be limited to those who have contributed substantially to the work reported.

Funding: Please add: "This research received no external funding" or "This research was funded by NAME OF FUNDER grant number XXX." and and "The APC was funded by XXX". Check carefully that the details given are accurate and use the standard spelling of funding agency names at https://search.crossref.org/funding, any errors may affect your future funding.

Institutional Review Board Statement: In this section, you should add the Institutional Review Board Statement and approval number, if relevant to your study. You might choose to exclude this statement if the study did not require ethical approval. Please note that the Editorial Office might ask you for further information. Please add "The study was conducted in accordance with the Declaration of Helsinki, and approved by the Institutional Review Board (or Ethics Committee) of NAME OF INSTITUTE (protocol code XXX and date of approval)." for studies involving humans. OR "The animal study protocol was approved by the Institutional Review Board (or Ethics Committee) of NAME OF INSTITUTE (protocol code XXX and date of approval)." for studies involving animals. OR "Ethical review and approval were waived for this study due to REASON (please provide a detailed justification)." OR "Not applicable" for studies not involving humans or animals.

Informed Consent Statement: Any research article describing a study involving humans should contain this statement. Please add "Informed consent was obtained from all subjects involved in the study." OR "Patient consent was waived due to REASON (please provide a detailed justification)." OR "Not applicable" for studies not involving humans. You might also choose to exclude this statement if the study did not involve humans.

Written informed consent for publication must be obtained from participating patients who can be identified (including by the patients themselves). Please state "Written informed consent has been obtained from the patient(s) to publish this paper" if applicable.

Data Availability Statement: In this section, please provide details regarding where data supporting reported results can be found, including links to publicly archived datasets analyzed or generated during the study. Please refer to suggested Data Availability Statements in section "MDPI Research

Data Policies" at https://www.mdpi.com/ethics. If the study did not report any data, you might add "Not applicable" here.

Acknowledgments: In this section you can acknowledge any support given which is not covered by the author contribution or funding sections. This may include administrative and technical support, or donations in kind (e.g., materials used for experiments).

Conflicts of Interest: Declare conflicts of interest or state "The authors declare no conflict of interest." Authors must identify and declare any personal circumstances or interest that may be perceived as inappropriately influencing the representation or interpretation of reported research results. Any role of the funders in the design of the study; in the collection, analyses or interpretation of data; in the writing of the manuscript; or in the decision to publish the results must be declared in this section. If there is no role, please state "The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results".

Sample Availability: Samples of the compounds ... are available from the authors.

Abbreviations

The following abbreviations are used in this manuscript:

MDPI	Multidisciplinary Digital Publishing Institute
DOAJ	Directory of open access journals
TLA	Three letter acronym
LD	Linear dichroism

Appendix A

Appendix A.1

The appendix is an optional section that can contain details and data supplemental to the main text—for example, explanations of experimental details that would disrupt the flow of the main text but nonetheless remain crucial to understanding and reproducing the research shown; figures of replicates for experiments of which representative data are shown in the main text can be added here if brief, or as Supplementary Data. Mathematical proofs of results not central to the paper can be added as an appendix.

Table A1. This is a table caption.

Title 1	Title 2	Title 3
Entry 1	Data	Data
Entry 2	Data	Data

Appendix B

All appendix sections must be cited in the main text. In the appendices, Figures, Tables, etc. should be labeled, starting with "A"—e.g., Figure A1, Figure A2, etc.

References

12. Plano Vacinação COVID-19 - Covid-19 Estamos ON. Available online: https://covid19estamoson.gov.pt/plano-vacinacao-covid-19/ (accessed on 5 July 2022).
12. Vacinação | Saúde Valongo. Available online: https://saude.cm-valongo.pt/saber-mais/vacinacao-51 (accessed on 5 July 2022).
12. Plano de Vacinação COVID-19. Available online: https://www.sns.gov.pt/wp-content/uploads/2021/01/Apresentacao_PlanoVacinacao_2 (accessed on 5 July 2022).
12. Author 1, R.B.; Author 2, M.D.; Author 3, F.L. Como está a correr a vacinação da covid-19? Compare Portugal com os outros países. Available online: https://www.publico.pt/interactivo/vacina-covid-19 (accessed on 5 July 2022).
12. Mapas e Números do Coronavírus - Dashboard da Renascença. Available online: https://coronavirus.rr.sapo.pt/ (accessed on 5 July 2022).
12. WHO Coronavirus (COVID-19) Dashboard. Available online: https://covid19.who.int/data (accessed on 5 July 2022).
12. Coronavirus Pandemic (COVID-19). Available online: https://ourworldindata.org/coronavirus (accessed on 5 July 2022).

12. Yahoo Finance - Stock Market Live, Quotes, Business Finance News. Available online: https://finance.yahoo.com/ (accessed on 5 July 2022).
12. Apache Hadoop vs Apache Storm. Available online: https://www.educba.com/apache-hadoop-vs-apache-storm/ (accessed on 5 July 2022).
11. Author 1, W.R; Author 2, E.D.; Author 3, R.K.; Author 4, A.Z. Immunizing markets against the pandemic: COVID-19 vaccinations and stock volatility around the world. *Int. Rev. Financ. Anal.*, **2021**, *77*, Article 101819. Available online: https://reader.elsevier.com/reader/sd/pii/S1057521921001538?token=D569CFFA963C0DF25E2DD792B52EBD8369A98264D0D347BA9E6 west-1originCreation=20220617163857 (accessed on 5 July 2022).
11. Author 1, K.F.C.; Author 2, Z.C. COVID-19 vaccines and global stock markets. *Finance Research Letters. in press*. Available online: https://reader.elsevier.com/reader/sd/pii/S1544612322000873?token=24CCBFDF2E64B08FF12BE4EE962BDAC387FE73BAD370C93EA78 west-1originCreation=20220617163940 (accessed on 5 July 2022).
12. Why has Pfizer's stock not sky rocketed after the vaccine announcement?. Available online: https://www.quora.com/Why-has-Pfizer-s-stock-not-sky-rocketed-after-the-vaccine-announcement (accessed on 5 July 2022).