UNIVERSITÉ DU LUXEMBOURG

Final Use-Case Study

Mics : Adaptive computing

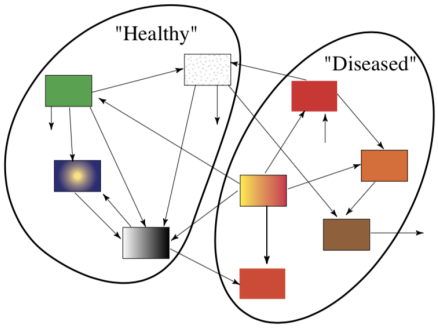
Big Data Analytics

COVID-19 : Modeling, Dynamics and Visualization in Epidemiology.

31/08/2020

Author : Moad Hani - 0190054306

Prof. Dr. Martin Theobald & Dr. Vinu Venugopal



2019/2020

Problem Definition

Our project tries to answer this question: How to better exploit data for modeling and monitoring the evolution of Covid-19?

The explosion in the amount of data being produced and the accompanying processing capacity is opening up previously unsuspected opportunities for all areas of research. To understand the change in scale of data production, Eric Shmidt, Google's CEO, said, "every two days we produce as much information as we generated from the dawn of civilization to 2003.

It is widely recognized that the analysis of large volumes of data, Big Data, could offer considerable information and opportunities to address societal challenges, such as health, safety, etc. As a result, data has become an indispensable tool and a strategic issue in today's world. The objective is to better understand and use available data to better act and guide collective decisions to improve the daily lives of citizens and create useful services that respond to issues or needs.

Certainly, scientific projects are a tremendous booster of research in data processing and this is one of the main reasons to address the Covid-19 threat through a rigorous analytical approach to identify the problem and show the advantages of data visualization in Big Data and combine their afferent technologies to analyze and make large and complex information afferent to Covid-19 datasets readable, condense information into a single graph, refine interpretation, memorize key elements using a graphical representation, optimize decision making, and foster innovation.

COVID-19 : Modeling, Dynamics and Visualization in Epidemiology.

Abstract

Our study on Covid-19 which is an academic project associated with the Big Data Analytics Module, will be divided in three parts:

The first part will deal in details with the SIR model which in general models well infectious diseases such as covid-19 with a python application (I practically translated a javascript code and I re-adapted it in python to demonstrate the impact of containment on propagation and the inverse impact of decontainment).

In addition, the second part will analyze the opinions of Twitter users in real time and give a simple visualization of the results obtained, which will allow to design a general idea at each moment when we want to make a study or initiate a fight against the virus.

Thus, the last part will be dedicated to the creation of a data pipeline to analyze covid using various tools and big data technologies such as Pyspark, and Kafka. For technical reasons we have chosen AWS instead of HCP. And here we will see real-time visualizations as output.

I also created a tracker with a public board and I was particularly interested in the case of the United States, although you can choose any country in the world in this tool and monitor its situation in real time.

Keywords: *SIR-type epidemiological models, First-order nonlinear random differential equations, Random variable transformation technique, Probability density function, Twitter API, Alyen API, PySpark, Kafka, Amazon Web Services, HDFS, NiFi, Tableau, Analytics, Visualization techniques, Python.*

List of Figures

Contents

Introduction

January 2020: The media reports the emergence of a mysterious viral pneumonia in Wuhan, China. Today, we know the culprit: the SARS-CoV-2 coronavirus. Since then, the whole world has been exposed to a continuous flow of information about its propagation capacities and its impact on mortality.

Although much of the mystery has been revealed, predicting the evolution of the spread of the virus remains tricky as many uncertainties remain about it. On the other hand, malaria (Malaria), a disease transmitted by a particular type of mosquito, still kills a child every 3 seconds in Africa and between 1 and 3 million people per year, 2 billion people, i.e. less than 40% of the world's population are exposed to this disease, according to estimates by the WHO (World Health Organization). Leprosy, avian flu, recently HIV..., are additional examples of contagious diseases that have worried or still worry society, because for such diseases the number of cases can suddenly increase in a given region at a given time and in a potentially uncontrollable way, we then talk about the study of epidemics. Studying, understanding and analyzing the development of an epidemic is a difficult task, but it is essential to predict this development and to act to slow down or even prevent it. These studies also make it possible to predict the consequences on the population the action of vaccination the quarantine or the distribution of screening tests.

[4].

In the fight against these viruses, mathematical models help us understand the spread of the virus and allow us to evaluate the effects of the interventions undertaken to limit its spread, with a major goal: to flatten the epidemic curve in order to keep it below the capacity of the health system. (Figure (1)).

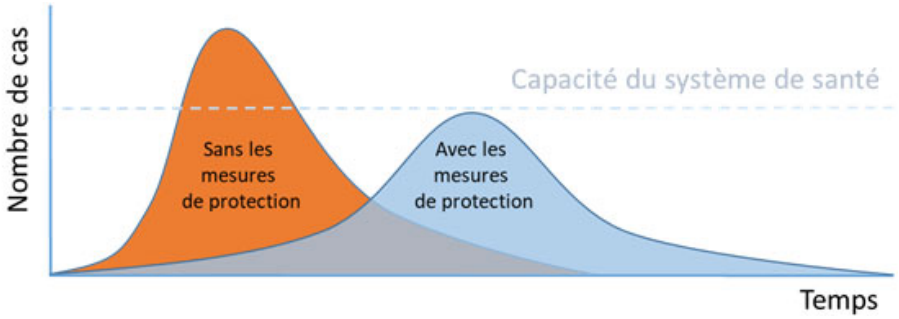


Figure 1: The evolution of an epidemic with or without preventive measures.

Mathematical modeling in epidemiology began in 1760 with Bernoulli's work to evaluate the effectiveness of inoculation against smallpox. [3].

Figure 2: Daniel Bernoulli (1700 –1782)

Bernouilli made the first mathematical model describing an infectious disease; it was the precursor to vaccination to prevent smallpox. Nowadays in the most modern journals his work and constantly cited, he is truly the precursor of mathematical epidemiology.
But there were other people, however one can say that the foundation of mathematical epidemiology based on compartmentalized models is the work of Sir Ronald Ross (1911) [9].

> Such calculation.... are useful, not for the numerical estimates yieldes by them, but because they give more precision to our ideas, and a guide for future investigation (Ronald ross)

Theorem. ***(Mosquito theorem Ronald Ross)*** :

1. *Whatever the original number of malaria cases in the locality may have been, the ultimate endemic malaria ratio will tend to settle down to*

a fixed figure, dependent on the number of Anophelines and the other factors – that is, if these factors remains constant all the time.

2. *If the number of Anophelines is sufficiently high, the ultimate malaria ratio (m) will become fixed at some figure between 0 and 1 (that is between 0 % and 100 %). If the number of Anophelines is sufficiently low (say below 40% per person), the ultimate malaria ratio will tend to zero – that is, the disease will tend to die out. (In this calculation a negative malaria ratio, that is, one which is less than nothing, must be interpreted as meaning zero).*

3. *A small change in the constants (e.g. the Anopheline factor) may produce a great change in the malaria.*

Ronald Ross gave the first mathematical model of malaria transmission where x_1 represents the proportion of infective humans and x_2 the proportion of infective mosquitoes and is written :

$$\begin{cases} \dot{x_1} = mab_1x_2(1-x_1) - \gamma x_1 \\ \dot{x_2} = b_2a(1-x_2)x_2 - \mu x_2 \end{cases}$$

Since then, mathematical modeling has become an essential tool in the analysis of the dynamics of infectious diseases. Indeed Ronald Ross used the above model to show that in order to eradicate malaria, the quantity of infectious mosquitoes just needs to be brought above a certain threshold, Ronald called what is today called mathematical epidemiology the theory of *happennings* or pathometry.

Modeling is now a research field in its own right, whose capabilities have been

increased tenfold by the tremendous computing power enabled by today's computers.

In the context of the Covid-19 pandemic, Ebola, HIV, or any other pandemic, epidemiologists develop, test and adjust models to simulate the spread of these infectious diseases in order to better understand them and optimize interventions to control them.

Recent models of CoV-2-SARS are often derived from the famous S.I.R model developed in 1927 by Kermack and McKendrik [1], which describes the transition between Susceptible (S), Infectious (I) and Recovered (R) populations. Susceptible" individuals are not immune to the contagious agent. Infectious" individuals are infected and, without necessarily being sick, may infect other "Susceptible" individuals. Recovered" individuals are immune to the disease after having fought it. In the case of CA-MRSA-CoV-2, it would be useful to add a population of Exposed (E) individuals to the model. Figure (3) describes the dynamics of transfers between individuals from different groups within the model.

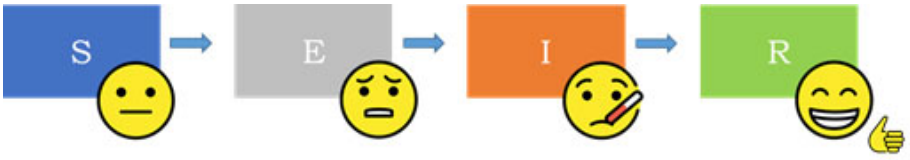


Figure 3: Representation of a simulation model with Susceptible (S), Exposed (E), Infectious (I) or Recovered (R) individuals.

This Figure (3) can also be described using equations, assigning a value to the arrows that move from one group to the other, called the transfer rate. Such a model can then be used to predict the number of infectious individuals (I), which can be compared with the numbers observed in reality. The model can also test the effect of interventions such as mandatory hand

washing, quarantine, or vaccination of individuals from at-risk populations. For example, Figure (4) describes how vaccination could slow down the current epidemic by preventing a shift from Susceptible (S) to Exposed (E). The model also allows calculation of the portion of the population that should be vaccinated to keep the total number of infected individuals below a specified threshold.

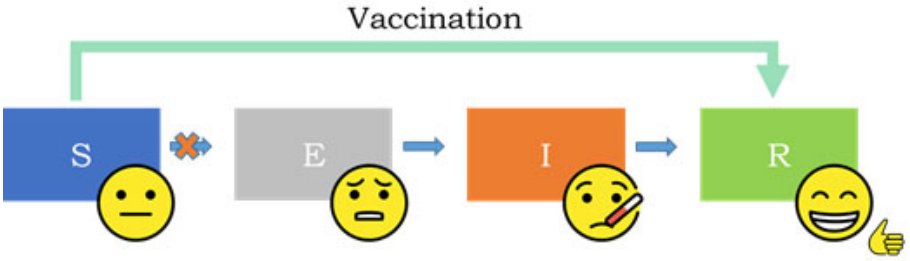


Figure 4: Representation of a simulation model with Susceptible (S), Exposed (E), Infectious (I) or Recovered (R) individuals showing how vaccination allows S to become R before being infected.

[5].

These simulation models allow to test virtually thousands of possible scenarios in a very short period of time. Uncertainties associated with certain model parameters limit the accuracy of predictions at the beginning of an epidemic. However, even in the absence of available data, the knowledge gained from previous epidemics can be used to build an initial model. The aim is to evaluate the impact of intervention programs in order to guide decision-makers in their choices on the one hand and scientists in their data collection activities on the other hand [5].

The arrival of new data leads to the refinement of models whose results can then influence intervention programs to keep the epidemic curve below the capacity of the health system (as recommended in Figure (1). Current

computing power allows for a substantial increase in model complexity. For example, a multi-agent model, which minimizes the movement of individuals and their interactions, has made it possible to predict the effects of quarantine measures on the Ebola epidemic in 2014.

Mathematical models can also be used to calculate certain parameters, such as the basic reproduction rate, R_0, which corresponds (in a simplified manner) to the average number of individuals that a carrier will infect for the duration of his or her infection. Under certain conditions, 1 - 1/R0 gives an indication of the proportion of the population likely to be infected during the epidemic. Currently, the R_0 of CoV-2 SARS is estimated to be approximately 2.5, which would suggest by applying the formula and, as stated in a simplified manner, that 60Mathematical models do not take into account all the parameters on which the epidemic depends, but they give an idea of the spread in the near future. The study focuses on asymptotic epidemic behavior in the presence of containment and isolation, confirming that in the Moroccan case, these two factors played an important role;

Mathematical models in epidemiology are very numerous! They target interacting human populations, their frequency, their distribution in time (and space), and propose predictions on these population densities, whether healthy, infected or cured (SIR- Susceptible, Infectious, Recovered).

The SIR-type mathematical models can be extended to study the dynamics of the spread of Covid 19 (or another pandemic), and incorporate terms that take into account the effect of containment of healthy individuals (S) and isolation of infected individuals (I). In order to achieve reasonable results, initial data on the densities of the different SIR compartments, an estimate of the contagion rate, and other elements are required. These deniers allow the calculation of the famous basic reproduction number R_0, describing the

average number of secondary cases produced by a typical infectious individual placed in a population consisting entirely of healthy individuals, during his or her period of infection [8].

Mathematically, in order to ensure that the epidemic is extinguished, it is necessary to control certain parameters on which R_0 depends, to make it smaller than 1 strictly; it is then understood that if $R_0 \leq 1$, the average number of secondary cases produced by an infectious individual tends gradually towards zero, which leads to the eradication of the epidemic.

Among these parameters, on which R_0 depends, those that measure population containment and isolation rates are fundamental. His results confirm that the more important these are, the more the amplitude of the epidemic decreases.

In order to alleviate the problem of the number of resuscitation beds in hospitals, the available places, and the health personnel who must face an unexpected wave, it is therefore necessary to ensure (via containment and isolation for example, in the absence of a vaccine) that the peak of the disease is lower and flatter, and spread over a fairly wide time range, thus reducing the amplitude of this tsunami, and consequently facilitating the reception of patients in hospitals! Flattening the epidemiological curve is fundamental! . As mentioned and illustrated in the Figure (4), vaccinating the population would be an ideal solution. In the absence of available vaccine, containment (combined with other social distancing measures) will limit the proportion of people who are likely to become infectious. Simulation models are therefore a crucial tool in the fight against CoV-2-SARS, helping to understand the enormous effort required to combat the disease.

Chapter 1

Study of Epidemiological Models SIR Stochastic

1.1 SIR epidemiological models: Description and solution

The SIR model developed by Kermack-McKendrick in 1927 [1] is one of the simplest compartmentalized models, and many models are derivatives of this basic form. The model has three compartments: S for the number of susceptible persons, I for the number of infectious persons, and R for the number of persons cured or recovered (or immunized). (This compartment can also be called "resistant" or "withdrawn".) This model is reasonably predictive for infectious diseases where recovery confers lasting resistance, such as measles, mumps and rubella. These (S, I *and* R) variables represent the number of people in each compartment at a given time. To represent that the number of susceptible, infectious, and recovered individuals may vary over time (even if the total population size remains constant), we consider specific

numbers of populations as a function of $t(time), S(t),\ I(t)\ and\ R(t)$. For a specific disease in a specific population, these functions can be developed to predict possible epidemics, and to control them as implied by the variable function of t. The importance of this dynamic aspect is most evident in an endemic disease with a short infectious period, such as measles in the UK before the introduction of a vaccine in 1968. These diseases tend to occur in cycles of epidemics because of the variation in the number of susceptible $(S(t))$ over time. During an epidemic, the number of susceptible individuals decreases rapidly as more individuals become infected and enter the epidemic. thus in infectious and removed compartments. The disease can only reappear when the number of susceptible individuals has recovered, for example following the birth of offspring in the susceptible compartment.

Each member of the population generally progresses from susceptible to infectious to withdrawn. This can be represented by a flow chart in which the boxes represent the different compartments and the arrows represent the transition between compartments, i.e. :

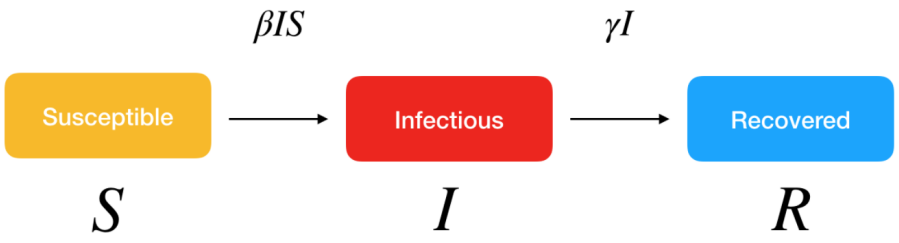


Figure 1.1: RIS model with dynamics and constant population

For complete model specification, the arrows must be labeled with the transmission rates between compartments. Between S and I, the transmission rate is $-\gamma SI$, γ is the average number of contacts per person per unit time, multiplied by the probability of disease transmission in a contact between

a susceptible and an infectious person, or otherwise γ is the actual contact rate, and SI/N is the fraction of those contacts between an infectious and susceptible individual that result in the infection of the susceptible person. (This is very similar to the law of mass action in chemistry in which random collisions between molecules result in a chemical reaction and the fractional rate is proportional to the concentration of the two reagents).

Between I and R, the transition rate is assumed to be proportional to the number of infectious individuals which is γI. It will not only be the number of individuals who will have recovered and acquired immunity, but also the number of deaths. If an individual is infectious for a period of time $\mathcal{D}$, then $\gamma = 1/\mathcal{D}$. The dynamics of this model is given by the following system:

If we assume that the total population size N, is fixed during the epidemic (i.e. birth and death rates are neglected), then this assumption is generally assumed when the incubation period of the disease is somewhat short:

$$S(t) + I(t) + R(t) = N, \qquad t \geq 0 \tag{1.1}$$

It is reasonable to treat the percentage or proportion of susceptible, infected and recovered people instead of the number of sizes. Therefore, the equation. (1.2) can be divided by N to obtain

$$S(t) + I(t) + R(t) = 1, \qquad t \geq 0 \tag{1.2}$$

Therefore, S (t), I (t) and R (t) are the percentages of susceptible, infected and recovered at any time t, respectively. Mathematically, standardized SIR models are considered as an initial value problem (IVP) of a system of nonlinear differential equations (NLDEs) that takes the form [7]:

$$\dot{S} = -\gamma I(t)S(t), \qquad t \geq 0. \tag{1.3}$$

$$\dot{I} = \gamma I(t)S(t) - \delta I(t), \qquad t \geq 0. \tag{1.4}$$

$$\dot{R} = \delta I(t), \qquad t \geq 0. \tag{1.5}$$

With initial conditions:

$$S(0) = S_0 \ , \quad I(0) = I_0, \quad R(0) = 0 \tag{1.6}$$

An approximate solution of the system of equations (1.3, 1.4, 1.5) with initial conditions $S(0) = S_0$ and $I(0) = I_0$ (1.6) is obtained as follows:
First of all we have:

$$\dot{S}/S = -\gamma I \quad \text{and} \quad \dot{I}/I = \gamma S - \delta \tag{1.7}$$

By dividing these 2 equations we obtain :

$$\frac{\dot{S}/S}{\dot{I}/I} = \frac{-\gamma I}{\gamma S - \delta} \tag{1.8}$$

Therefore :

$$\gamma \dot{S} - \delta \frac{\dot{S}}{S} = -\gamma \dot{I} \tag{1.9}$$

Integrating (1.9) with respect to time and applying the initial conditions, we have:

$$\gamma S - \delta ln(S) = -\gamma I + \gamma A \tag{1.10}$$

With constant A equal to :

$$A = 1 - \frac{\delta}{\gamma} ln(s_0) \tag{1.11}$$

We can eliminate the proportion of infected between equations (1.3) and (1.10) to obtain :

$$\frac{\dot{S}}{S} = \gamma S - \delta ln(S) - \gamma A \tag{1.12}$$

Replace $ln(S)$ by u (1.12) becomes:

$$\dot{u} = \gamma e^u - \delta u - \gamma A \tag{1.13}$$

In the radius $u < 1$, $e^u = 1 + u + ... \cong 1 + u$ and therefore (1.13) is approximated by :

$$\dot{u} \cong \gamma(1 - A) + (\gamma - \delta)u \tag{1.14}$$

The normalized susceptible S sub-population is between 0 and 1 (i.e. $0 < s < 1$). So for the radius $u < 1$ or also $|ln(s)| < 1$, this condition means that the analysis is valid for $1/e < s < 1$. The solution of the equation (1.14) with the initial condition $u(0) = u_0$ is the following :

$$u = \frac{1}{\gamma - \delta}\{[\gamma(1 - A) + (\gamma - \delta)u_0]e^{(\gamma-\delta)t} - \gamma(1 - A)\} \tag{1.15}$$

Replacing the value of u by $ln(s)$ and A in equation (1.11) and (1.15)

$$ln(S) = \frac{ln(S_0)}{\gamma - \delta}\{\gamma e^{(\gamma-\delta)t} - \delta\} \tag{1.16}$$

Thus

$$S = S_0^{\frac{\gamma e^{(\gamma-\delta)t} - \delta}{\gamma-\delta}}, \qquad t \geq 0. \tag{1.17}$$

From the equation (1.10) the proportion of infected people is:

$$I = A - S + \frac{\delta}{\gamma} ln(S). \tag{1.18}$$

Replacing A and S in equation (1.11) and (1.17) respectively in equation (1.18) :

$$I = 1 - S_0^{\frac{\gamma e^{(\gamma-\delta)t} - \delta}{\gamma-\delta}} + \frac{\delta}{\gamma}\left(-ln(S_0) + ln(S_0^{\frac{\gamma e^{(\gamma-\delta)t} - \delta}{)}})\right), \qquad t \geq 0. \tag{1.19}$$

The solution of the process $I(t)$, as given in equation (1.19), is not simple enough to use the RVT technique, so the approximation $|ln(s)| < 1$ can be used in equation (1.18) to obtain a suitable form, as follows: Replacing $ln(S)$ by u on (1.18) :

$$I = A - e^u + \frac{\delta}{\gamma}u \cong A - (1+u) + \frac{\delta}{\gamma}u \tag{1.20}$$

In terms of S becomes :

$$I \cong -\frac{\delta}{\gamma}ln(S_0) - \left(1 - \frac{\delta}{\gamma}\right) ln(S). \tag{1.21}$$

we replace $ln(S)$ in equation (1.16)

$$I \cong -\frac{\delta}{\gamma}ln(S_0) - \frac{\gamma - \delta}{\gamma(\gamma - \delta)} \left(\gamma e^{(\gamma-\delta)t} - \delta\right) ln(s_0) \tag{1.22}$$

or

$$I \cong -e^{(\gamma-\delta)t} ln(S_0) \tag{1.23}$$

the proportion of recovered $R(t)$ is obtained from equation (1.18)

$$R = 1 - S - I = 1 - A - \frac{\delta}{\gamma}ln(S) \tag{1.24}$$

Replacing A and $ln(S)$ with (1.11) and (1.16) gives :

$$R = \frac{\delta}{\gamma}\left(ln(S_0) - ln(S_0^{\frac{\gamma e^{(\gamma-\delta)t} - \delta}{\gamma - \delta}})\right), \qquad t \geq 0 \tag{1.25}$$

$$\text{or } R = \delta\left(\frac{1 - e^{(\gamma-\delta)t}}{\gamma - \delta}\right) ln(S_0) \tag{1.26}$$

And thus the approximate solution of the three previous equations (1.3, 1.4, 1.5), representing the epidemiological SIR model (1.1) with their initial conditions given in the equations (1.6), is given as follows :

$$S = S_0^{\frac{\gamma e^{(\gamma-\delta)t}-\delta}{\gamma-\delta}}, \qquad t \geq 0. \tag{1.27}$$

$$I = 1 - S_0^{\frac{\gamma e^{(\gamma-\delta)t}-\delta}{\gamma-\delta}} + \frac{\delta}{\gamma}\left(-ln(S_0) + ln(S_0^{\frac{\gamma e^{(\gamma-\delta)t}-\delta}{\gamma-\delta}})\right), \qquad t \geq 0 \tag{1.28}$$

$$R = \delta\left(\frac{1-e^{(\gamma-\delta)t}}{\gamma-\delta}\right) ln(S_0), \qquad t \geq 0. \tag{1.29}$$

1.2 The stochastic solution of the SIR random epidemiological models

In this section, we consider a general stochastic SIR modelwhere the initial conditions S_0, the contact rate γ, and the recovery rate δ are random variables with the following domains:

$$\begin{aligned} D_{S_0} &= \{s_0 = s_0(\omega), \omega \in \Omega{:}0 \leq s_{0,1} \leq s_{0,2} \leq 1\} \\ D_\gamma &= \{\gamma = \gamma(\omega), \omega \in \Omega{:}0 \ \leq \ \gamma_1 \ \leq \ \gamma_2 \ \leq \ 1\} \\ D_\delta &= \{\delta = \delta(\omega), \omega \in \Omega{:}0 \leq \delta_1 \leq \delta\delta_2 \leq 1\} \end{aligned} \tag{1.30}$$

We will develop a stochastic solution of this random SIR model by deriving the probability density function (PDF) of the stochastic processes of the solution,$S(t)$, $I(t)$ and $R(t)$. In addition, key statistical properties such as the mean, variance functions and confidence intervals are evaluated based on this PDF. In addition, the basic reproduction number R_0, which is an interesting parameter to control the spread of the epidemic, is interpreted in the stochastic case. Determining the PDF of this parameter is one of the objectives of this section.

In order to find the PDFs, $f_S(s,t)$, $f_I(i,t)$ and $f_R(r,t)$ for the solution of the stochastic process, $S(t)$, $I(t)$ and $R(t)$ of the initial value problem (IVP 1.1), respectively.

1.3 Calculation of the probability density of stochastic process solutions

For any fixed instant $t \geq 0$ let us consider the initial value problem(1.3, 1.4, 1.4) where the input parameters are assumed to be random variables with initial conditions (1.6), Applying the technique of random variable transformation with the equations (1.27, 1.28, 1.29) we obtain the probability density of susceptible, infectious, and recovered populations.

1.3.1 Calculation of the probability density of the susceptible proportion $S(t)$

The Section Theorem (**??**) can be used with the following choices

$$X = (S_0, \gamma, \delta)^T$$

;

$$Y = (U, V, W)^T$$

$$Y = f(X)$$

$$e : \mathbb{R}^3 \to \mathbb{R}^3,\ f(S_0, \gamma, \delta) = (f_1(S_0, \gamma, \delta), f_2(S_0, \gamma, \delta), f_3(S_0, \gamma, \delta))^T = (U, V, W)^T \tag{1.31}$$

$$where\ U = S_0^{\frac{\gamma e^{(\gamma-\delta)t}-\delta}{\gamma-\delta}}, \quad V = \gamma, \quad W = \delta$$

Extract S_0, γ, δ we get :

$$S_0 = U^{\frac{V-W}{Ve^{(V-W)t}-W}}, \qquad \gamma = V, \qquad delta = W, \tag{1.32}$$

Thus, according to the theorem (**??**), the joint probability density function of the vector (U, V, W) is given by:

$$f_{U,V,W}(u, v, w) = f_{S_0,\gamma,\delta}(u^{\frac{v-w}{ve^{(v-w)t}-w}}, v, w)|J| \tag{1.33}$$

where J is given by :

$$J = \frac{v-w}{u(ve^{(v-w)t}-w)} u^{\frac{v-w}{ve^{(v-w)t}-w}}$$

J is the Jacobian function obtained from the inverse of f, the vector of change of the variable is defined in (1.32) and therefore $g(U, V, W)$ the inverse function of f is :

$$g : \mathbb{R}^3 \to \mathbb{R}^3,\ g(U, V, W) = (g_1(U, V, W), g_2(U, V, W), g_3(U, V, W))^T = (S_0, \gamma, \delta)^T \tag{1.34}$$

Therefore, the PDF of the likely proportion, $S(t)$, is the marginal PDF (γ, δ) of the joint probability density (1.32) :

$$f_S(s;t) = \int_{D_\gamma}\int_{D_\delta} f_{S_0,\gamma,\delta}(u^{\frac{v-w}{ve^{(v-w)t}-w}}, v, w)|\frac{v-w}{u(ve^{(v-w)t}-w)}u^{\frac{v-w}{ve^{(v-w)t}-w}}|dvdw \tag{1.35}$$

1.3.2 Calculation of the probability density of the infected proportion I (t)

To calculate the PDF of the solution $I(t)$ as given on (1.23) we can define U, V, W as follows:

$$U = -e^{(\gamma-\delta)t}ln(S_0), \qquad V = \gamma, \qquad W = \delta$$

We extract S_0, γ, δ we obtain

$$S_0 = e^{-Ue^{-(V-W)t}}, \qquad \gamma = V, \qquad \delta = W \tag{1.36}$$

The vector $(U, V, W)^T$. is given by :

$$f_{U,V,W}(u, v, w) = f_{S_0,\gamma,\beta}(e^{-ue^{-(v-w)t}}, v, w)|J| \tag{1.37}$$

$$where\ J = -e^{-ue^{-(v-w)t}-(v-w)t}$$

J is the Jacobian function of the inverse of f. the variable change vector is defined in (1.36). And so PDF of the infected population $I(t)$, is the marginalization on δ, γ on the joint PDF (1.37)

$$f_I(i;t) = \int_{D_\gamma}\int_{D_\delta} f_{S_0,\gamma,\delta}(e^{-ie^{-(v-w)t}}, v, w)| - e^{-ie^{-(v-w)t}-(v-w)t}|dvdw \tag{1.38}$$

1.3.3 Calculation of the probability density of the recovered proportion R(t)

Always the PDF of the $R(t)$ solution as given on (1.29) can be obtained in terms of $U, V,\ and\ W$ by the same way

$$U = \delta(\frac{1-e^{(\gamma-\delta)t}}{\gamma-\delta})ln(S_0), \qquad V = \gamma, \qquad W = \delta$$

We get S_0, γ, δ by inverse :

$$S_0 = e^{\frac{U(V-W)}{W\left(1-e^{(V-W)t}\right)}}, \qquad \gamma = V, \qquad \delta = W \tag{1.39}$$

The joint PDF of the vector $(U, V, W)^T$ is given by :

$$f_{U,V,W}(u,v,w) = f_{S_0,\gamma,\delta}(e^{\frac{u(v-w)}{w\left(1-e^{(v-w)t}\right)}}, v, w)|J| \tag{1.40}$$

$$where\ J = \frac{v-w}{w(1-e^{(v-w)t})}e^{\frac{u(v-w)}{w\left(1-e^{(v-w)t}\right)}}$$

J is the Jacobian function of the inverse of f. the variable change vector is defined in (1.39). And so the population PDF of the recovered $R(t)$, is the marginalization on δ, γ on the joint PDF:

$$f_R(r;t) = \int_{D_\gamma}\int_{D_\delta} f_{S_0,\gamma,\delta}(e^{\frac{u(v-w)}{w\left(1-e^{(v-w)t}\right)}}, v, w)|\frac{v-w}{w(1-e^{(v-w)t})}e^{\frac{u(v-w)}{w\left(1-e^{(v-w)t}\right)}}|dvdw \tag{1.41}$$

Remark 1.1. *To ensure the existence of all probability densities derived above for the solutions $S(t), I(t); and R(t)$, given by the equations (1.35, 1.38, 1.41) all input random variables, S_0, γ, δ, are assumed to be absolutely continuous over their domains, defined by (1.30) .*

1.4 Mean, Variance, Confidence interval

The PDFs derived in the previous section allow us to compute the main statistical information related to the stochastic processes $S(t), I(t)$ *and* $R(t)$ such as the mean and variance functions defined by:

$$\mathbb{E}[Q(t)] = \mu_Q(t) = \int_{-\infty}^{\infty} q f_Q(q) dq \tag{1.42}$$

$$\mathbb{V}[Q(t)] = (\sigma_Q(t))^2 = \int_{-\infty}^{\infty} q^2 f_Q(q) dq - (\mu_Q(t))^2 \tag{1.43}$$

For the susceptible population :

$$\mathbb{E}[S(t)] = \mu_S(t) = \int_{s_{01}}^{s_{02}} s f_S(s) ds \; \mathbb{V}[S(t)] \;\; = (\sigma_S(t))^2 = \int_{-\infty}^{\infty} s^2 f_S(s) ds - (\mu_S(t))^2 \tag{1.44}$$

For the infected population :

$$\begin{aligned} \mathbb{E}[I(t)] = \mu_I(t) &= \int_{i_{01}}^{i_{02}} i f_I(i) di \\ \mathbb{V}[I(t)] = (\sigma_I(t))^2 &= \int_{-\infty}^{\infty} i^2 f_I(i) di - (\mu_I(t))^2 \end{aligned} \tag{1.45}$$

For the recovered population :

$$\begin{aligned} \mathbb{E}[R(t)] = \mu_R(t) &= \int_{r_{01}}^{r_{02}} r f_R(r) dr \\ \mathbb{V}[R(t)] = (\sigma_R(t))^2 &= \int_{-\infty}^{\infty} r^2 f_R(r) dr - (\mu_R(t))^2 \end{aligned} \tag{1.46}$$

The confidence interval is another important statistical property. It is the interval to which a random variable with a predefined probability belongs. This probability is called the confidence level. For our proposed SIR model, the probability that the percentage of susceptible, at a given time $t = \hat{t}$, belongs to a specific interval, say $[s_1(\hat{t}), s_2({}_t t)]$=$[\hat{s_1}, \hat{s_2}]$ can be determined by the following integral:

$$\mathbb{P}\left(\omega \in \Omega : S(\hat{t}; \omega)) \in [\hat{s_1}, \hat{s_2}]\right) = \int_{\hat{s_2}}^{\hat{s_1}} f_S(s; \hat{t}) ds \tag{1.47}$$

To construct probabilistic intervals for $(1-\theta)$ 100 % confidence level, the PDF can be used for each $\hat{t} \geq 0$ set to be determined $s_1(\hat{t})$ and $s_2(\hat{t})$ so that :

$$\int_0^{\hat{s_1}} f_S(s;\hat{t})ds = \frac{\theta}{2} = \int_{\hat{s_2}}^1 f_S(s;\hat{t})ds \tag{1.48}$$

where θ in [0, 1] *theta* is fixed, and

$$1-\theta = \mathbb{P}\left(\omega \in \Omega : S(\hat{t};\omega)) \in [\hat{s_1}, \hat{s_2}]\right) = \int_{\hat{s_2}}^{\hat{s_1}} f_S(s;\hat{t})ds \tag{1.49}$$

Generally confidence levels is 95%, or $1-\theta$ =0.95, is built. Similar expressions can be given for the percentage of infected and recovered proportions by exchanging $f_S(s;\hat{t})$ by $f_I(r,\hat{t})$ and $f_R.(r,\hat{t})$ in the equations. (1.47)- (1.49), respectively.

1.5 Basic Reproduction Rate

The basic reproduction rate R_0 is the most well known and important value in epidemiology, R_0 is defined as secondary infections occurring when an infectious is introduced into a population completely susceptible to [2]. [6]

. This is the most common value that gives the line %of demarcationbetween the continuation of the disease or the disease will disappear.

The R_0 helps to estimate the time it takes for the number of victims of an epidemic to double. It also helps to determine the minimum proportion of a population (P) that must be immunized by natural infection or vaccination (if available) to prevent the outbreak or persistence of an epidemic: $P = 1-\frac{1}{R_0}$. This is referred to as the herd immunity effect (gregarious immunity, *herd immunity*) to refer to the percentage of the population that would need to be immunized in order for the epidemic to stop thriving.

This selection is based on the relation given by:

$$R_0 = \frac{\gamma}{\delta} \begin{cases} \text{si } R_0 < 1 \,(\ i.e\ \gamma < \delta) \text{ the disease will disappear when } t \to \infty \\ \text{si } R_0 > 1 \,(\ i.e\ \gamma > \delta) \text{ the disease will spread when } t \to \infty \end{cases}$$

This result follows from equation (1.23), since ($lim_{t\to+\infty} I(t) = 0$). Note that this result is consistent with its obvious interpretation. The epidemic disappears when the infection rate is lower than the cure rate. Equally, when the condition $\gamma < \delta$ is verified. In our context, both γ and δ are assumed to be random variables, so that the requirement for epidemic extinction in a deterministic framework is $\gamma < \delta$. This means calculating the probability $\mathbb{P}$. in the stochastic scenario,

$$where\ \Lambda = \{\omega \in \Omega : \gamma(\omega) < \delta(\omega)\} = \{\omega \in \Omega : R_0(\omega) < 1\} \tag{1.50}$$

This key probability can be calculated by using the following identification between the SIR model notation and the one used in this proposal

$$X = (X_1, X_2)^T = (\gamma, \delta)^T, \qquad Y = \frac{X_1}{X_2} = \frac{\gamma}{\delta} = R_0 \tag{1.51}$$

We get

$$f_{R_0}(r_0) = \int_{D_\delta} f_{\gamma,\delta}(x, r_0 x)|x|dx \qquad r_0 > 0, \tag{1.52}$$

where $f_{\gamma,\delta}(.,.)$ denotes the marginal distribution on S_0 of the joint PDF, $f_{S_0,\gamma,\delta}(.,.,.)$. Consequently, as $R_0(\omega) > 0$ for any $\omega \in \Omega$ the target probability of the event described in (1.50) can be calculated as follows :

$$\mathbb{P}[\Lambda] = \int_0^1 \int_{D_\gamma} f_{\gamma,\delta}(x, r_0 x)|x|dxdr_0 \tag{1.53}$$

Chapter 2

Modeling Epidemic in Python

2.1 How to program the evolution of a SIR(M) model ?

Algorithms of the "game of life" type have been used for a long time in many fields. They also allow to simulate the spread of a virus in a given population and they offer interesting visual effects in the graphical representation of the results. They can be used to report on the evolution of a probabilistic SIR(M) model.

The program below has been built with the following assumptions :

- acquired or initial immunity is permanent

- an infected person infects his or her "neighbors" with a probContagion contagion rate

- the disease is lethal from the 8th day with a probability of death of probDeath

- possible states are Healthy, Infected, Immune (Remitted) or Dead

If I run the program with the following parameters :

- probContagion = 0.057

- probDeath = 0.005

- Days of infection = 14

I get this kind of simulation (the color represents the state: blue=healthy, red=infected, green=immune, black=deceased) and the dynamics of the model is as follows (the scale is logarithmic!) :

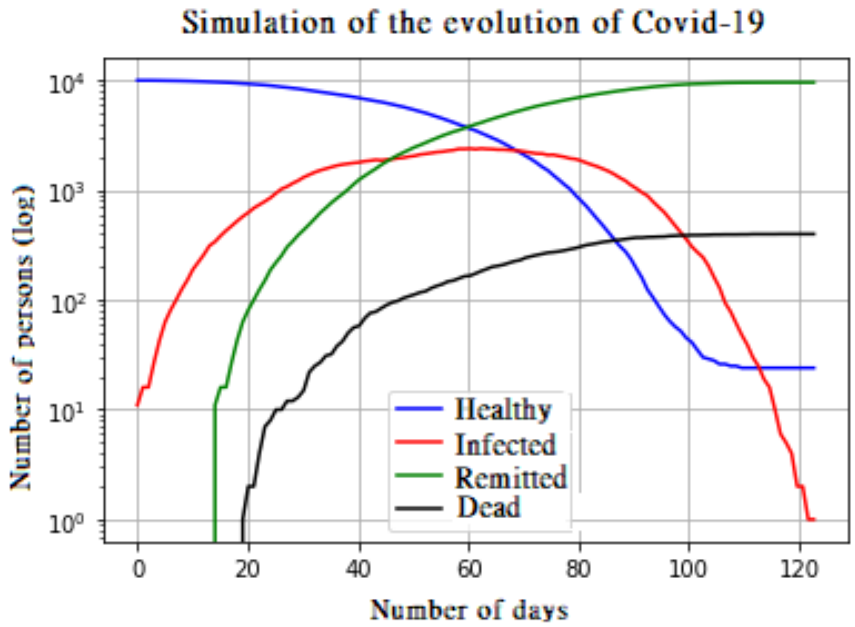


Figure 2.1: The evolution of COV-19 epidemic by a simple representation

This simulation represents a highly contagious epidemic; it spreads very quickly and the whole population becomes either immune or dead after 120 days.

Now if I run the program with the following parameters:

- probContagion = 0.014

- probDeath = 0.005

- Days of infection = 14

I get the following simulation and the dynamics of the model is as follows (the scale is logarithmic!) :

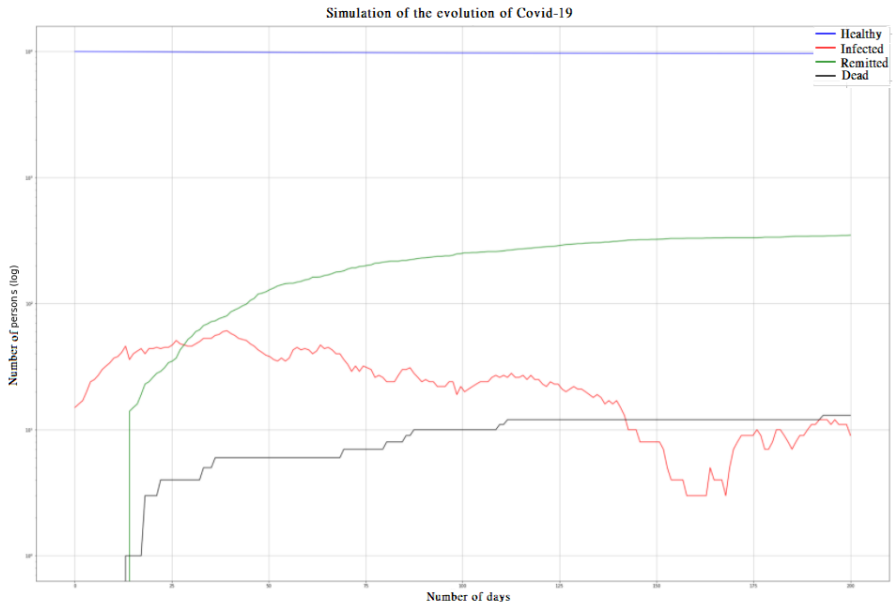


Figure 2.2: The evolution of COVID-19 epidemic with a lower probability of contagion

2.2 Probabilistic Model of Epidemic Spread: Impact of Containment and Decontainment

Containment is an excellent measure to counter an epidemic because it allows to limit the contacts and therefore the contagions of the epidemic.

The simulations below were generated with a Python program that simulates the spread of an epidemic via a "game of life" type algorithm. The rules of this "game" are as follows:

- initial population: 10,000 people

- initial infected cases: 5 people

- death rate: 0.005

- duration of the simulation: 200 days

- the duration of infection is 24 days

- the disease is lethal between the 10th day and the 24th day.

- Immunity is acquired for all remaining days after the 24th infection.

- three phases are defined: phase 1 of 40 days which corresponds to the initial contagion phase, phase 2 of containment which lasts 57 days and the deconfinement phase which lasts until the 200th day.

1st case: no containment, a strong epidemic lasting 200 days.

On the simulation, we get about 500 deaths or 5% of the population, which is very important (reduced to 67 million, that would be more than 3 million deaths). We get a typical curve of the bell-shaped SIR model, but after 200 days, we have not yet passed into the decreasing part and we stagnate in a flat part.

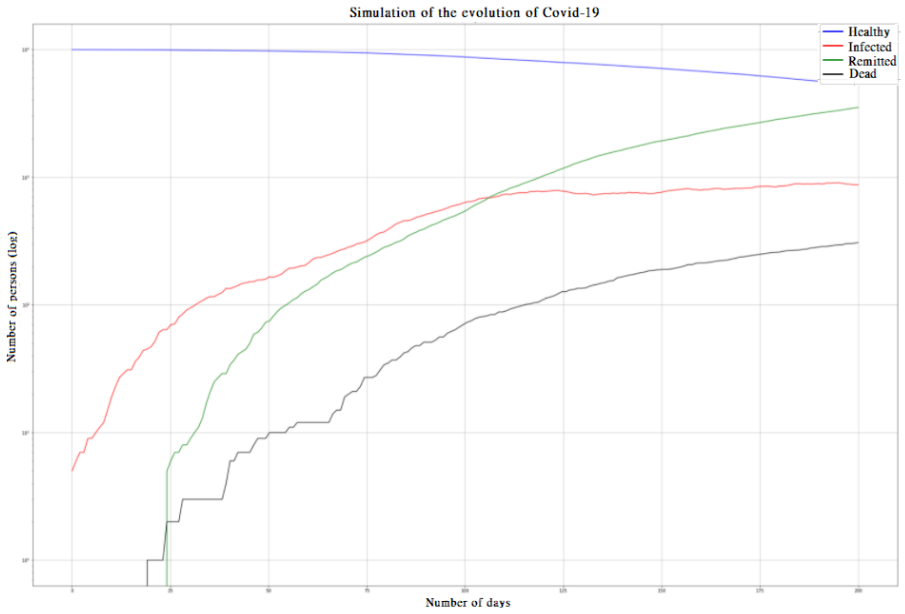


Figure 2.3: The evolution of COVID-19 epidemic - case 1

2nd case: confinement from day 41 to the end of the simulation (200 days) The simulation consists of 2 phases:

- an initial phase of rapid propagation for 40 days

- and 160 days of containment to slow the spread with a contagion rate that is three times lower.

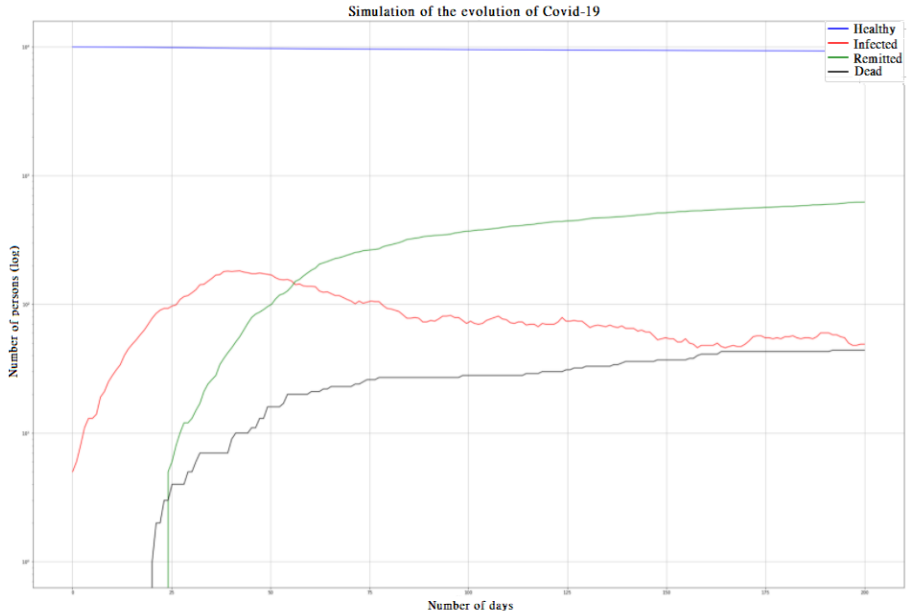


Figure 2.4: The evolution of COVID-19 epidemic - case 2

In the simulation, we obtain about 50 deaths or 0.5% of the population. A typical curve of the damped SIR model is obtained and the spread of the virus is gradually extinguished due to the lack of contact.

3rd case: initial propagation phase of 40 days then containment of 57 days then deconfinement from the 98th day to the end of the simulation (200 days).

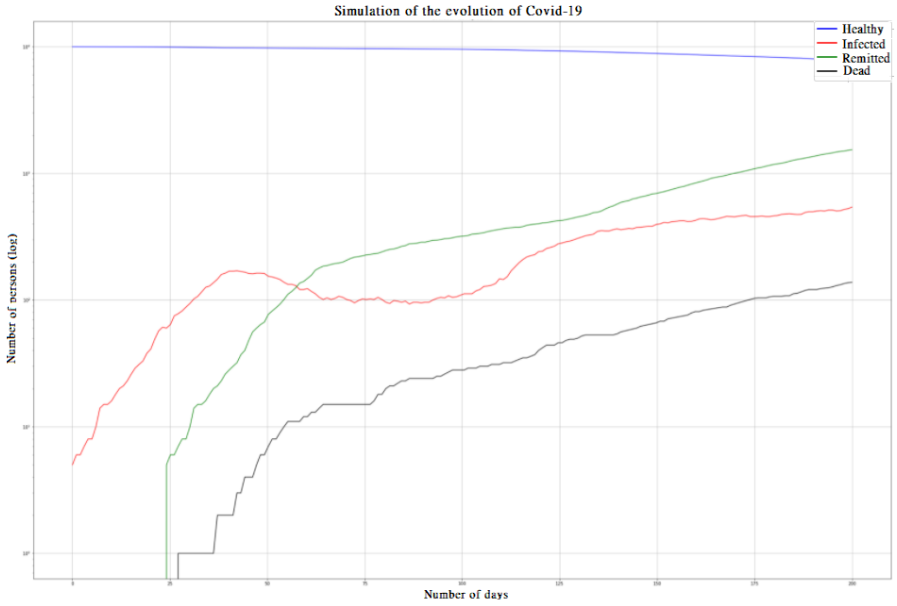


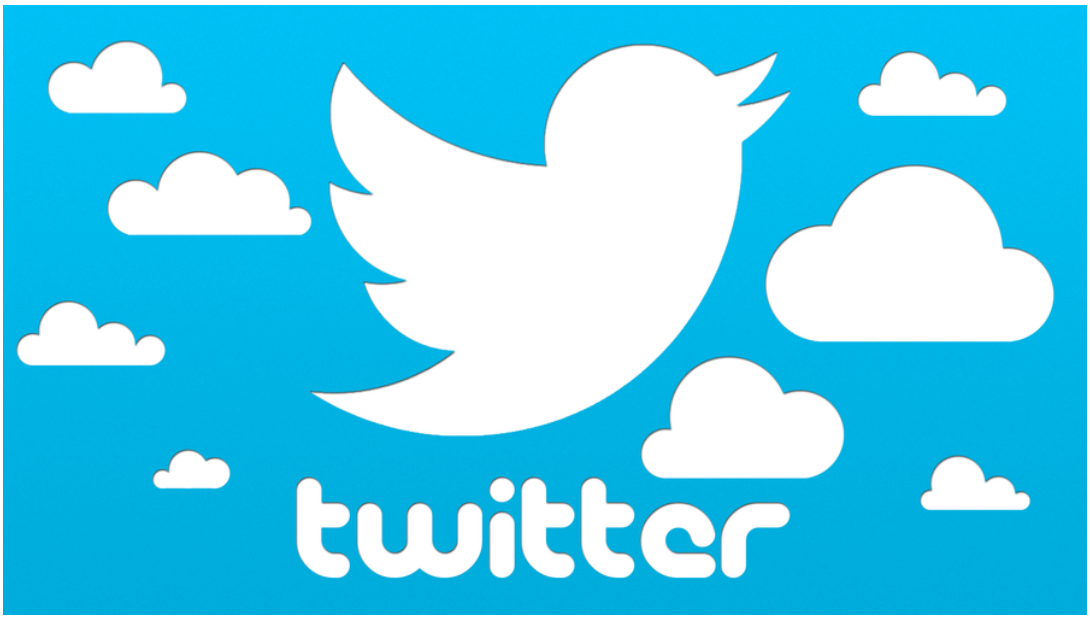
Figure 2.5: The evolution of COVID-19 epidemic - case 3

This would result in a greater number of deaths than the previous case (2 to 3 times more deaths, i.e. a rate of 1 to 1.5%). The dynamics are interesting because we can clearly see the impact of confinement on the curve of infected cases and the rebound (the "second wave") due to deconfinement after 97 days.

Conclusion: Containment is an efficient solution to fight epidemics because it allows a drastic decrease in the contagion rate. A simple "game of life" type model demonstrates this. Deconfinement always leads to an increase in the contagion rate (because there are more contacts). The models show that this deconfinement leads to a revival of the "second wave" type of epidemic.

Chapter 3

Sentiment Analysis - COVID-19 (Twitter)



Twitter users around the world send about 350,000 new Tweets every minute, creating 6,000 pieces of information of 140 characters per second. Twitter is now an extremely valuable resource from which we can extract information using text mining tools such as sentiment analysis.

Within the general social chatter every second, there are large amounts of extremely valuable information waiting to be extracted. Through sentiment analysis, we can generate reflections on consumer reactions to ads, opinions about products or brands,

and even track opinion about events as they unfold. For this reason, we often hear sentiment analysis referred to as "opinion mining".

With this in mind, we decided to implement a useful tool based on a Python script to start exploring the opinion of the general public, in the face of the Covid-19 virus.

What does this script do?

Using this script, we can gather Tweets with the Twitter API, analyze their feelings with the AYLIEN Text Analysis API, and visualize the results with Matplotlib. The script also provides visualization and saves the results in a CSV file to facilitate reporting and analysis. The 3 main objectives of its creation :

1 - Understand the audience's reaction to covid-19 news on Twitter

2 - Measure the opinions on covid-19 and know, in real time, the opinion of the Internet users.

3 - The most interesting thing in the script is that you can search for the desired tag and the script will execute the related tweets and through the same analysis pipeline, store the results in a CSV file and display a pie chart visualization.

```
In [42]: sentiment = client.Sentiment({'text': '

John is a very good football player!'})

In [44]: print(sentiment)

{'polarity': 'positive', 'subjectivity': 'objective',
'text': 'John is a very good football player!',
'polarity_confidence':
0.9940106272697449,  'subjectivity_confidence': 0.943706847162803}


In [17]: import sys
         import pandas
         import tweepy
         import csv
         import matplotlib.pyplot as plt
         from aylienapiclient import textapi
```

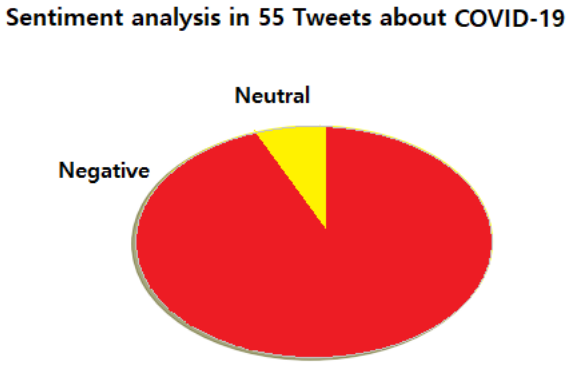
```
from collections import Counter
if sys.version_info[0] < 3:
    input = raw_input


## Informations related to Twitter identification
consumer_key = "a2TgI8DFAzeGlqklZtTIPgcHT"#"dLcHBP0Qo3IwggxQwFxMQ2Y3v"
consumer_secret = "6epE48RiTLJ8WzRrCnDsiBt3tbkKsx98kkafOoX5xFHON4RbeE"#""
access_token = "3335730178-zOS83eBBuv7dg3jfZFMypTBEI9o0mrXcAY62Eng" #"
access_token_secret = "gzlnL1tbQa0npB1YD0ys5fWLJ4MGsuYVcJdFnCKdvXqvd" #""


## Create and Configure an instance of Tweepy
auth = tweepy.OAuthHandler(consumer_key, consumer_secret)
auth.set_access_token(access_token, access_token_secret)
api = tweepy.API(auth)

## Informations of Alyen Identification

client = textapi.Client("2dbbb66d","97f9a67569cde2c9281a98877628d25a")

## Searching a tag - example of covid-19

query = input("What topic do you want to analyze for this example? \n")
number = input("How many Tweets do you want to analyze? \n")
results = api.search(lang="enfr", q=query + " -rt", count=number,

result_type="mixed")

print("--- --- Search for Tweets \n")
```

```
with open("test-COVID-19-8.csv", 'w', newline='') as csvfile:
    csv_writer = csv.DictWriter(f=csvfile, delimiter=',',quotechar='|',
    fieldnames=["Tweet", "Sentiment"])
    csv_writer.writeheader()
    print("--- Open a CSV file to store the results of your analysis")


## Tidy up the tweets set and send it to the Alyen API
    for c, result in enumerate(results, start=1):
        tweet = result.text
        tidy_tweet = tweet.strip().encode('ascii', 'ignore')

        if len(tweet) == 0:
                    print('Tweet vide')
                    continue
        response = client.Sentiment({'text': tidy_tweet})
        csv_writer.writerow({'Tweet': response['text'],'
        Sentiment': response['polarity']})
        print("Number of Tweets analyzed{}".format(c))


## Count the data in the Sentiment column of the CSV file
with open("test-COVID-19-8.csv", 'r') as data:
    counter = Counter()
    for row in csv.DictReader(data):
        counter[row['Sentiment']] += 1

    positive = counter['positive']
    negative = counter['negative']
    neutral = counter['neutral']

       ## Declares the variables for the pie chart
```

```
        colors = ['green', 'red', 'yellow']
        sizes = [positive, negative, neutral]
        labels = 'Positive', 'Negative', 'Neutral'

            ## We use matplotlib to draw graphs
        plt.pie(x=sizes, shadow=True, colors=colors, labels=labels, startangle=90)

        plt.title("Sentiment analysis in {} Tweets about {}".format(number, query))
        plt.show()

What topic do you want to analyze for this example?
COVID-19
How many Tweets do you want to analyze?
55
--- Search for Tweets

--- Open a CSV file to store the results of your analysis....
Number of Tweets analyzed 1
Number of Tweets analyzed 2
Number of Tweets analyzed 3
Number of Tweets analyzed 4
Number of Tweets analyzed 5
Number of Tweets analyzed 6
Number of Tweets analyzed 7
Number of Tweets analyzed 8
Number of Tweets analyzed 9
Number of Tweets analyzed 10
Number of Tweets analyzed 11
Number of Tweets analyzed 12
Number of Tweets analyzed 13
Number of Tweets analyzed 14
Number of Tweets analyzed 15
Number of Tweets analyzed 16
```

```
Number of Tweets analyzed 17
Number of Tweets analyzed 18
Number of Tweets analyzed 19
Number of Tweets analyzed 20
Number of Tweets analyzed 21
Number of Tweets analyzed 22
Number of Tweets analyzed 23
Number of Tweets analyzed 24
Number of Tweets analyzed 25
Number of Tweets analyzed 26
Number of Tweets analyzed 27
Number of Tweets analyzed 28
Number of Tweets analyzed 29
Number of Tweets analyzed 30
Number of Tweets analyzed 31
Number of Tweets analyzed 32
Number of Tweets analyzed 33
Number of Tweets analyzed 34
Number of Tweets analyzed 35
Number of Tweets analyzed 36
Number of Tweets analyzed 37
Number of Tweets analyzed 38
Number of Tweets analyzed 39
Number of Tweets analyzed 40
Number of Tweets analyzed 41
Number of Tweets analyzed 42
Number of Tweets analyzed 43
Number of Tweets analyzed 44
Number of Tweets analyzed 45
Number of Tweets analyzed 46
Number of Tweets analyzed 47
Number of Tweets analyzed 48
```

```
Number of Tweets analyzed 49
Number of Tweets analyzed 50
Number of Tweets analyzed 51
Number of Tweets analyzed 52
Number of Tweets analyzed 53
Number of Tweets analyzed 54
Number of Tweets analyzed 55
```



Conclusion : Given the mixed situation between alarmists and conspirators, and until the pin is drawn, the saying: prevention is better than cure will still have its interest.

Chapter 4

Creation of a data pipeline to analyse Covid-19

4.1 Overview

The purpose is to collect the real time streaming data from COVID19 open API for every 5 minutes into the ecosystem using NiFi and to process it and store it in the data lake on AWS.Data processing includes parsing the data from complex JSON format to csv format then publishing to Kafka for persistent delivery of messages into PySpark for further processing.The processed data is then fed into output Kafka topic which is inturn consumed by Nifi and stored in HDFS .A Hive external table is created on top of HDFS processed data for which the process is Orchestrated using Airflow to run for every time interval. Finally KPIs are visualised in Tableau.

4.2 Project execution guidelines

1. As soon as we login into the EC2 instance , we need to start the following jps services using the commands .
2. Note that everytime we stop and start the instance these services need to be restarted if they are not running as Daemon processes.

To run Hadoop

$root@ip-172-31-23-142://home/ubuntu/hadoop2.7.1sbin/start-all.sh$

Or

$root@ip-172-31-23-142:/home/ubuntu/hadoop2.7.1sbin/start-dfs.sh$

$root@ip-172-31-23-142:/home/ubuntuhadoop2.7.1sbin/start-yarn.sh$

To run NiFi

$root@ip-172-31-23-142:/home/ubuntubin/nifi.shstart$

To run NiFi in the Browser

After running the vncserver, open the browser Then type http://localhost:9999/nifi

To run Kafka

$root@ip-172-31-23-142:/home/ubuntu/kafka/bin/kafka-server-start.shconfig/server.properties$

Before running the NiFi GUI browser we start the vnc server.

To start vncserver

$root@ip-172-31-23-142:/home/ubuntuvncserver$:1

To run Airflow
Running Airflow needs two terminals to be opened in parallel.
Need to login into the ec-2 instance into both the terminals.

Then go into the root folder by specifying

root@ip-172-31-23-142 cd

Then into airflow folder by

root@ip-172-31-23-142 cd Airflow

Then run each of these commands in different terminals and keep them running
root@ip-172-31-23-142 airflow webserver -p 8080
root@ip-172-31-23-142 airflow scheduler

Once all the processes are started, we can check the command jps in another terminal

Figure 4.1: Checking all the processes

The next step is to open the VNC server and Firefox browser in it, and give localhost:9999/nifi to open the NiFi UI

Here, as a first step , we extract the data from Corona Open API endpoint using

InvokeHttp processor -Drag and drop a new processor and search for Invokehttp and click on configure and specify details as follows

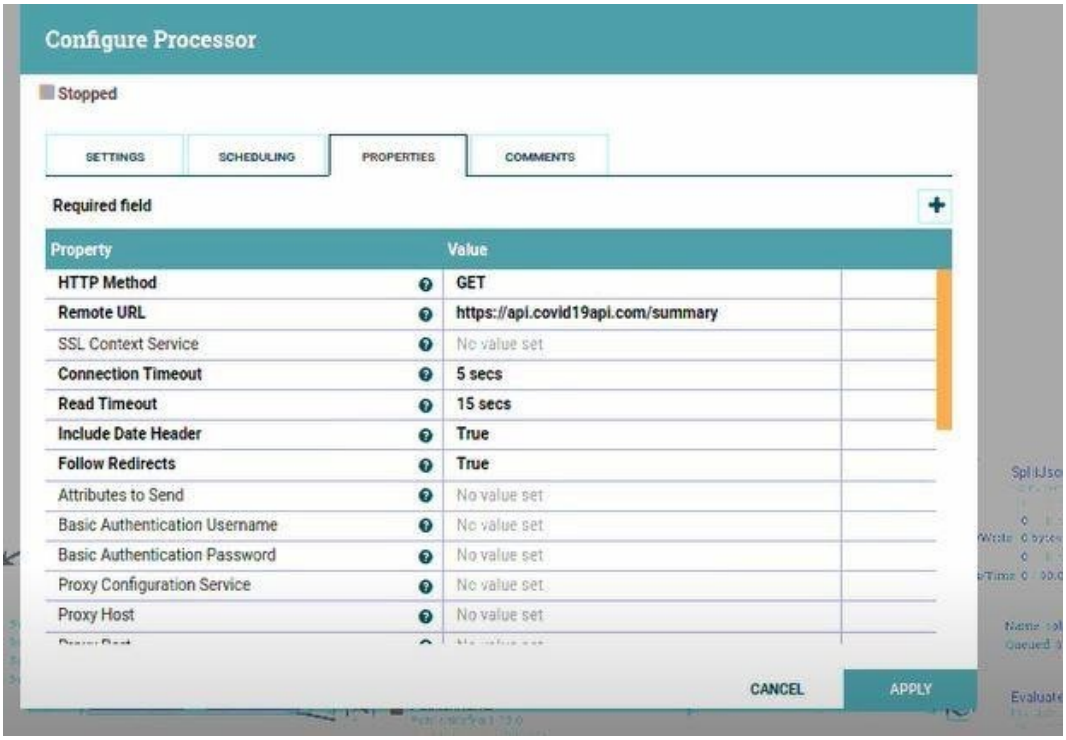


Figure 4.2: Extract the data from Corona Open API endpoint

Then run the processor and the raw data is received in nested JSON format (JSON array and JSON objects) from the COVID 19 API endpoint for every 5 minutes as scheduled in the processor.

Then, evaluate the fields which are as JSON object using EvaluateJSON processor with the following configuration

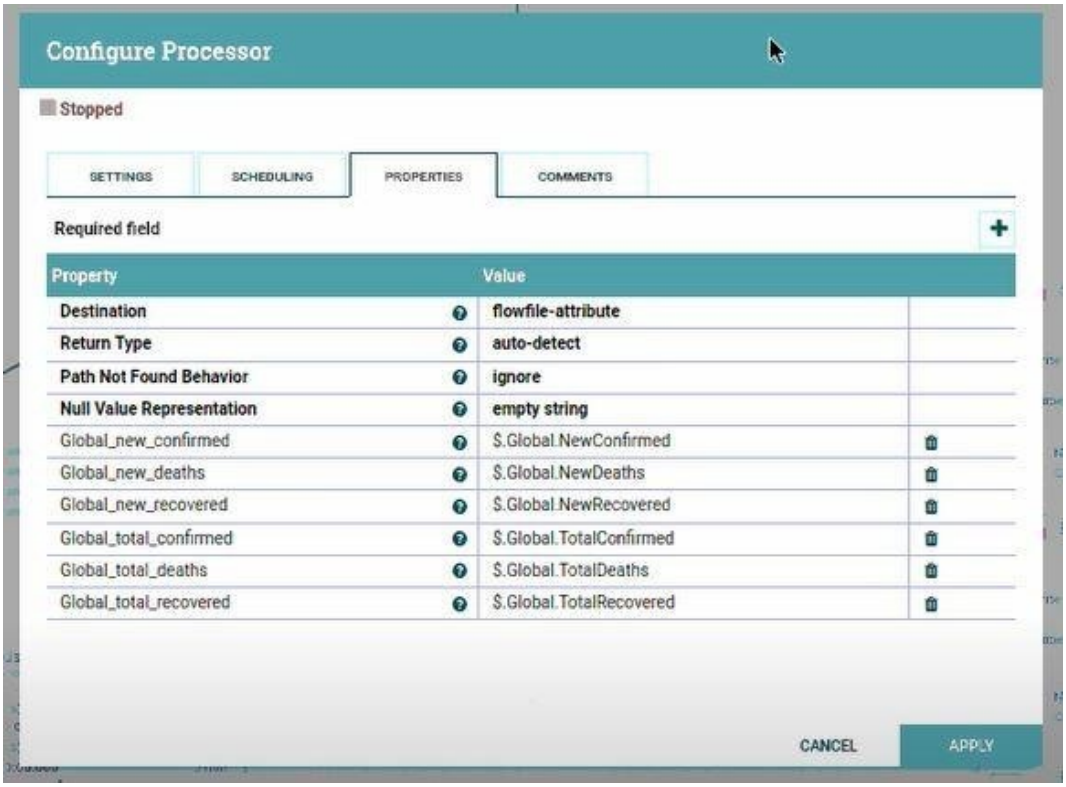


Figure 4.3: EvaluateJSON to find JSON Object

To evaluate fields which are as JSON arrays , splitJSON processor is used

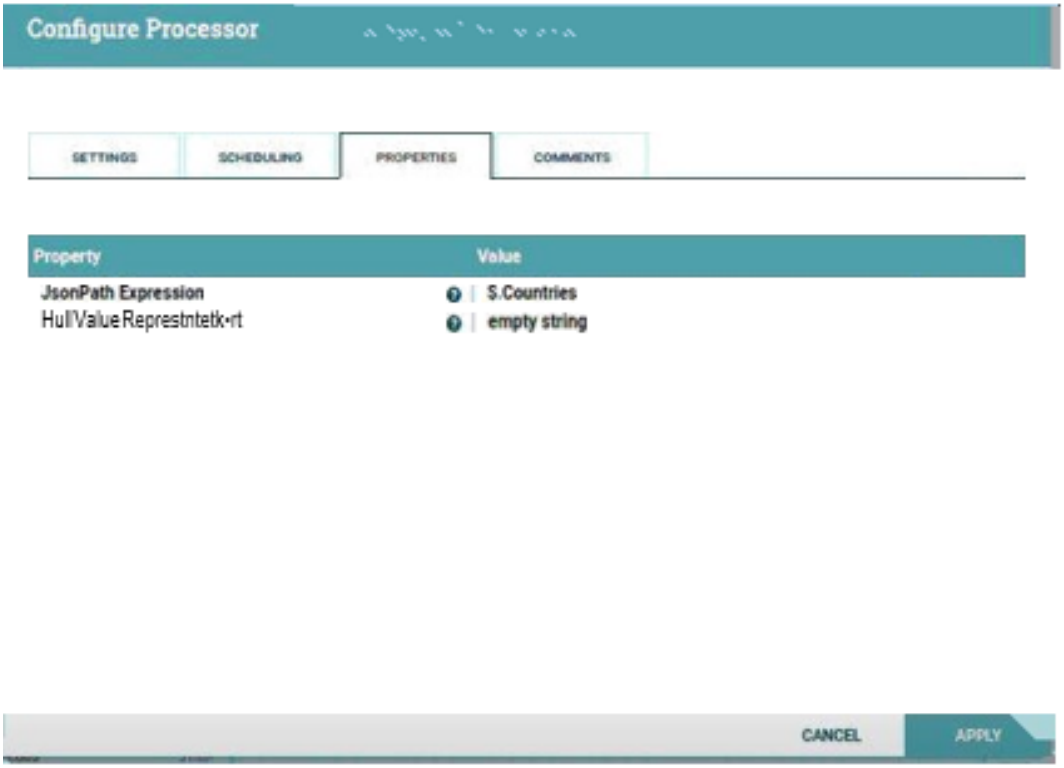


Figure 4.4: splitJSON processor to look for JSON arrays

Then we use EvaluateJSON again to extract the individual fields from JSON array which was split by the array

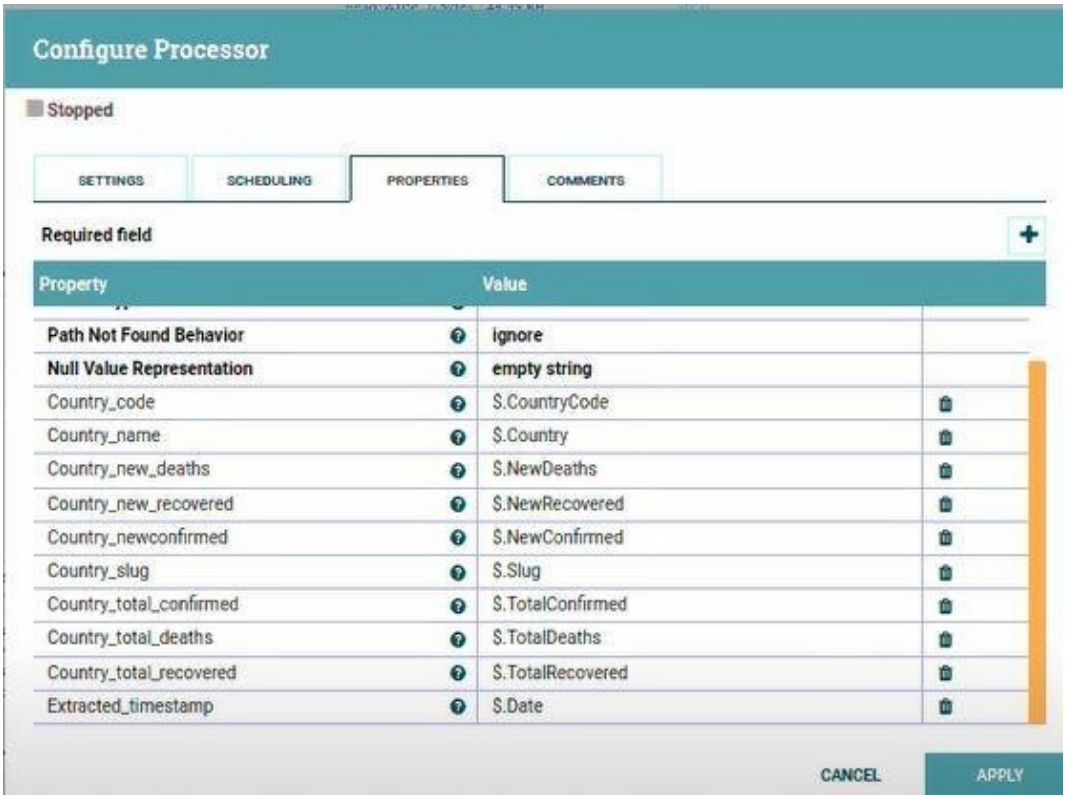


Figure 4.5: EvaluateJSON to extract the individual fields from JSON array

4.3 Encryption of Pii fields in the Data using Nifi

Data encryption of Pii fields is necessary for Data security purposes in large scale distribution environments.So, we've encrypted one of the fields as a sample to show how the encryption can be carried out using NiFi.

We would need three processors for the encryption purpose

1 . AttributestoEncrpyt

2. Cryptographic Hash Attribute

3. EncryptContent

Encryption template which needs to be created using set of processors for encrypting the data which needs to be imported as XML into NiFi from settings− >templates− >upload template and we need to upload the given XML here− >drag and drop the template icon in NiFi and choose the latest uploaded template .

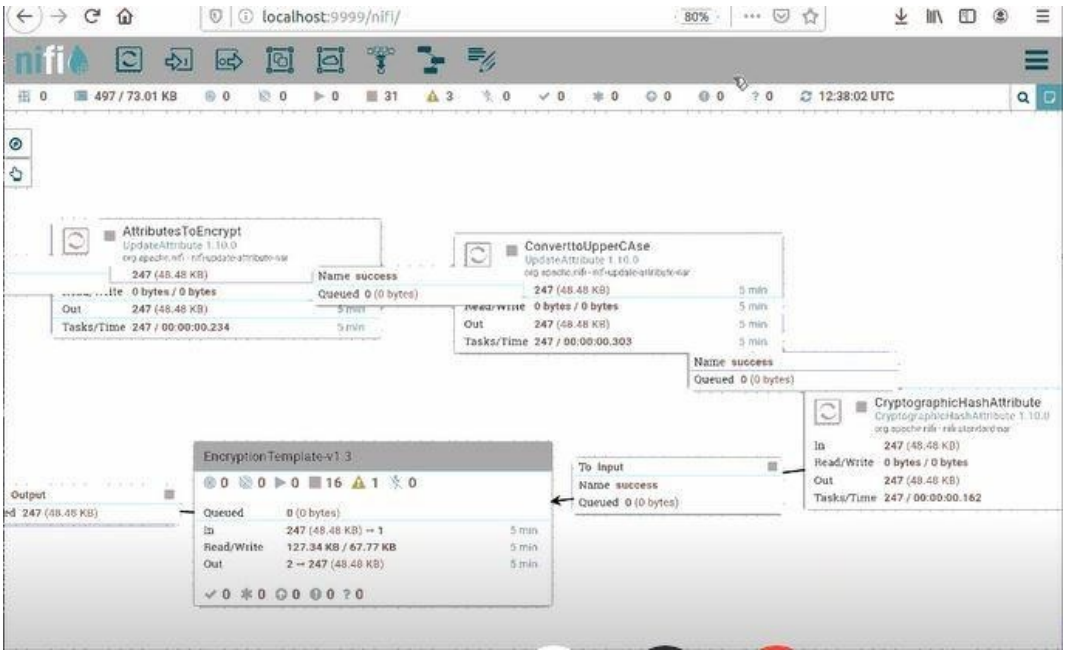


Figure 4.6: Schema of Encryption

Now, parse the values into CSV format using ReplaceText processor with commas as follows

Figure 4.7: Parsing values into CSV format

Now the parsed data in parallel goes into HDFS as well as Kafka for ease of use . For that we use PublishKafka and PutHDFS processors and are configured as follows

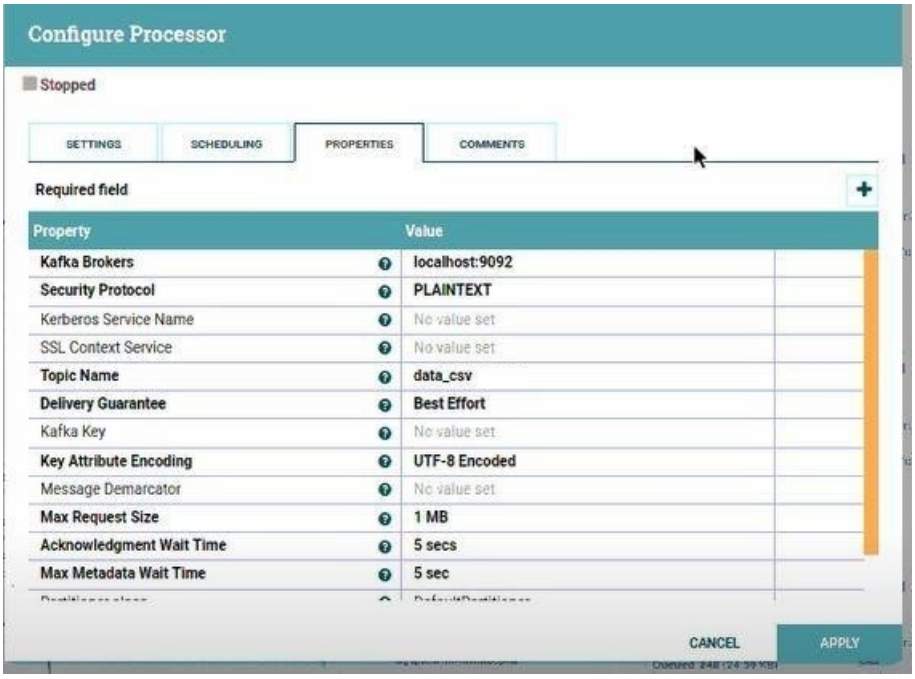


Figure 4.8: PublishKafka

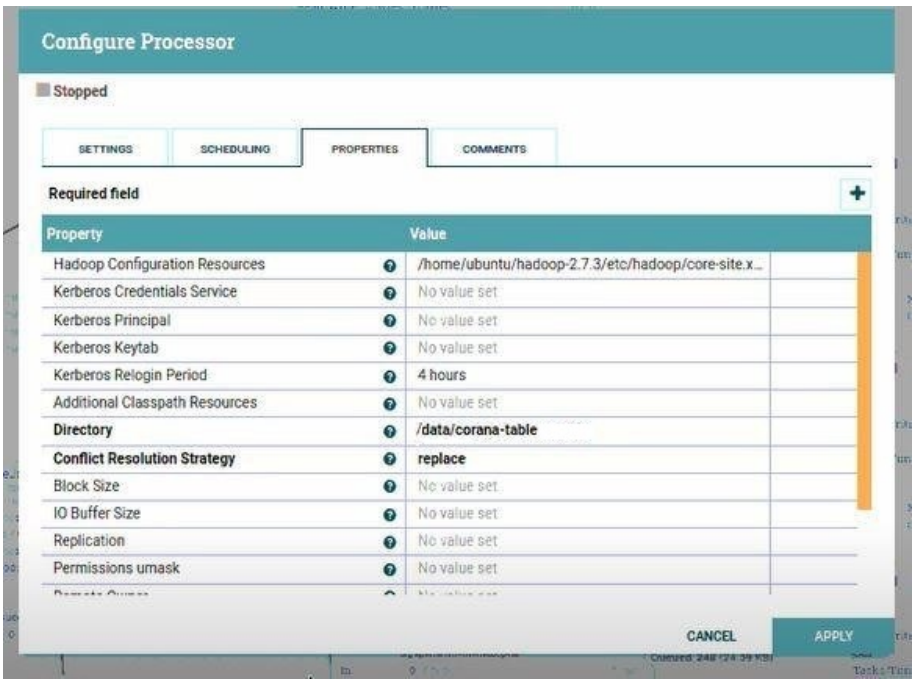


Figure 4.9: PutHDFS

Once the data is published to Kafka topic , we can view the data in the console using the following command: Navigate to bin folder where Kafka was installed .Then run the following command

bin/kafka-console-consumer.sh –bootstrap-server localhost:9092 –from- beginning –topic topic name

The data which is now in Kafka topic , we extract this streaming data messages into PySpark for streaming data processing
Before submitting the python code , ensure that you have downloaded the required jars in the below command and copy them into your ec2-instance from your local using scp command .
Then run the python code in the spark bin folder using

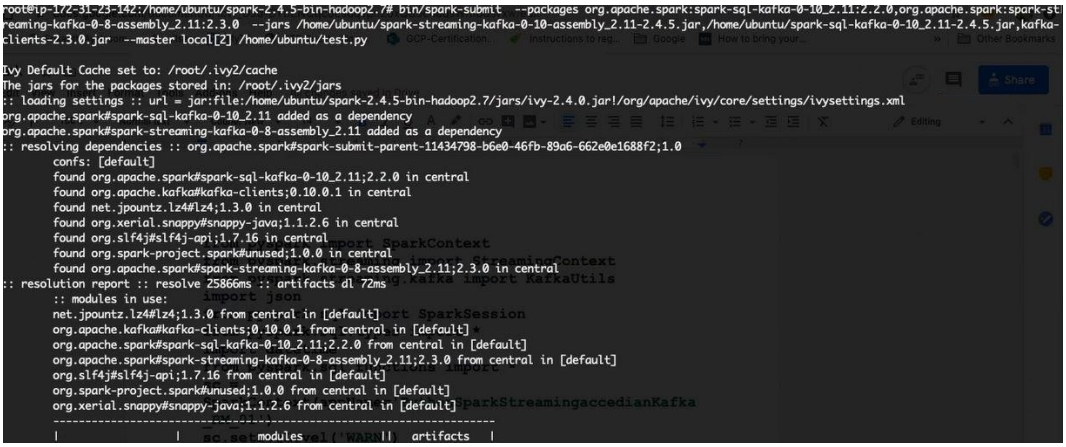


Figure 4.10: Running the code in Spark

The above python code is writing data to Kafka output topic in JSON format and it can be verified in console via Kafka console consumer

bin/kafka-console-consumer.sh –bootstrap-server localhost:9092 –from- beginning –topic topic name

Now the data in the output Kafka topic is ready to consume by consumers like Nifi. We use ConsumeKafka processor to get the data for further processing

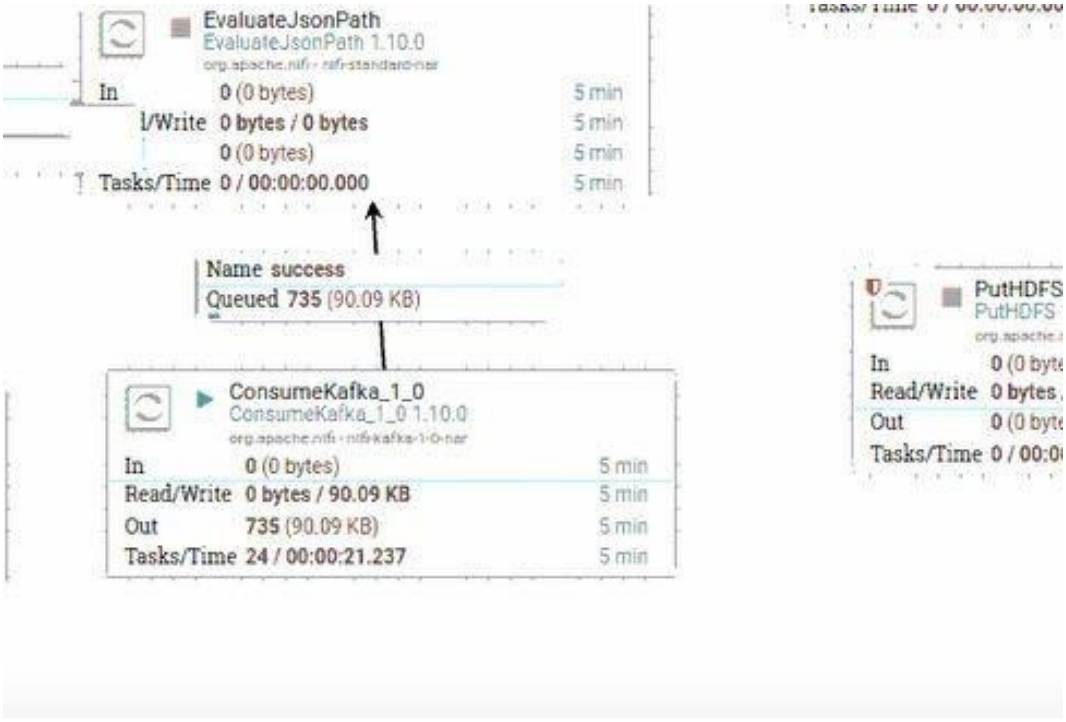


Figure 4.11: ConsumeKafka processor

Once the data is consumed by Nifi , We parse the data into CSV again and store it in HDFS. Then create a Hive external table using the below script in hive.

We login into Hive by entering into hive terminal by going into the folder

where apache- hive is installed and give the following command bin/hive which enters into Hive shell

4.4 Visualization of Data in AWS Quicksight

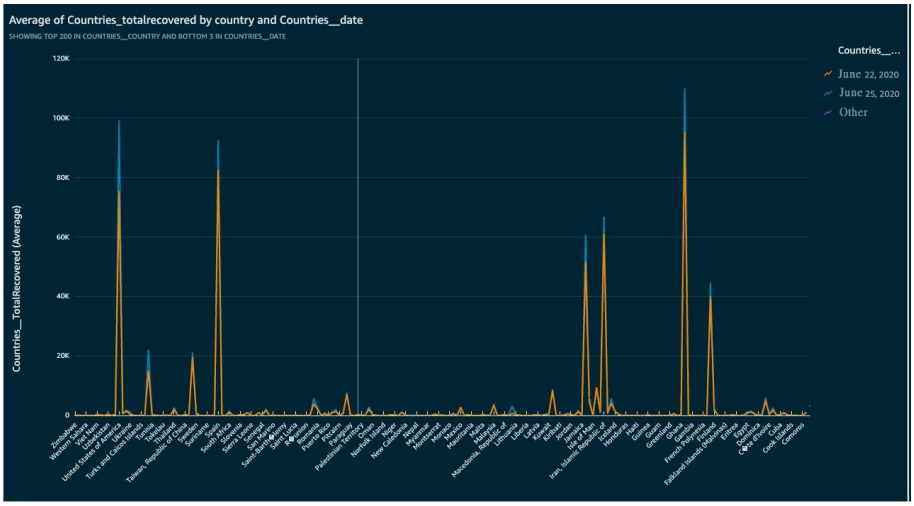


Figure 4.12: Visualization of Data in AWS Quicksight - Part 1

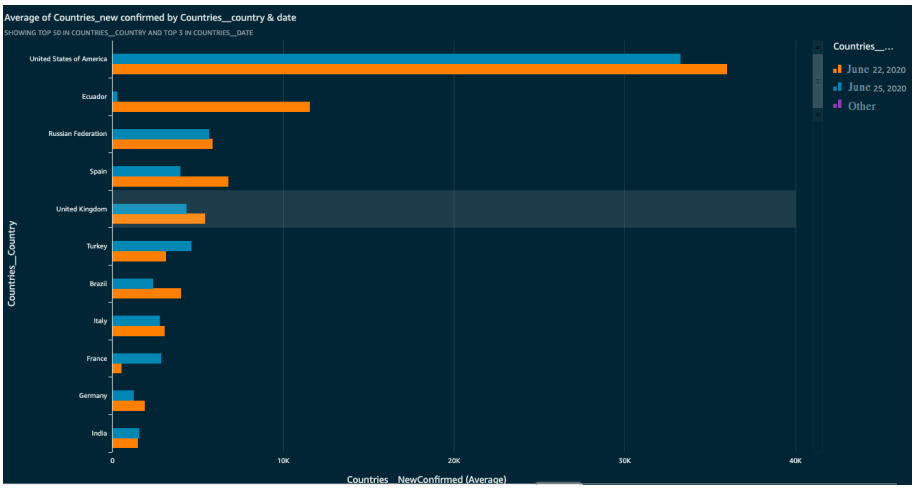


Figure 4.13: Visualization of Data in AWS Quicksight - Part 2

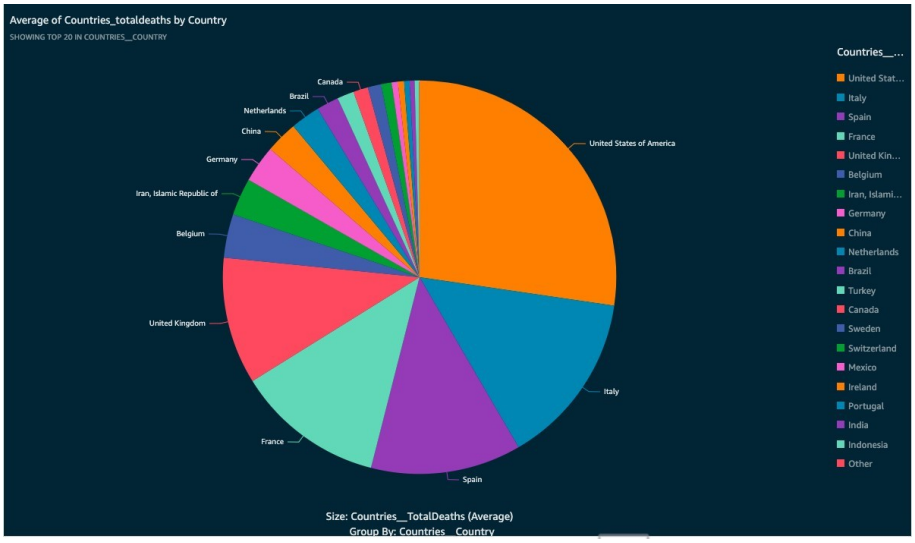


Figure 4.14: Visualization of Data in AWS Quicksight - Part 3

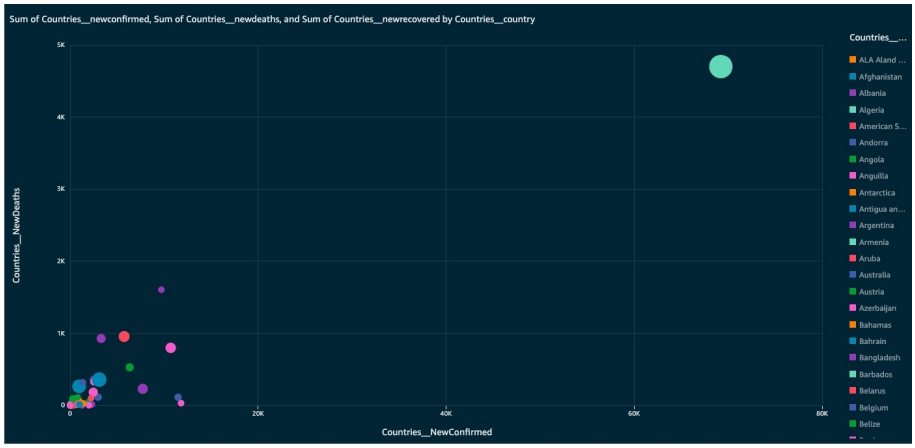


Figure 4.15: Visualization of Data in AWS Quicksight - Part 4

4.5 Visualisation of Data in Tableau online

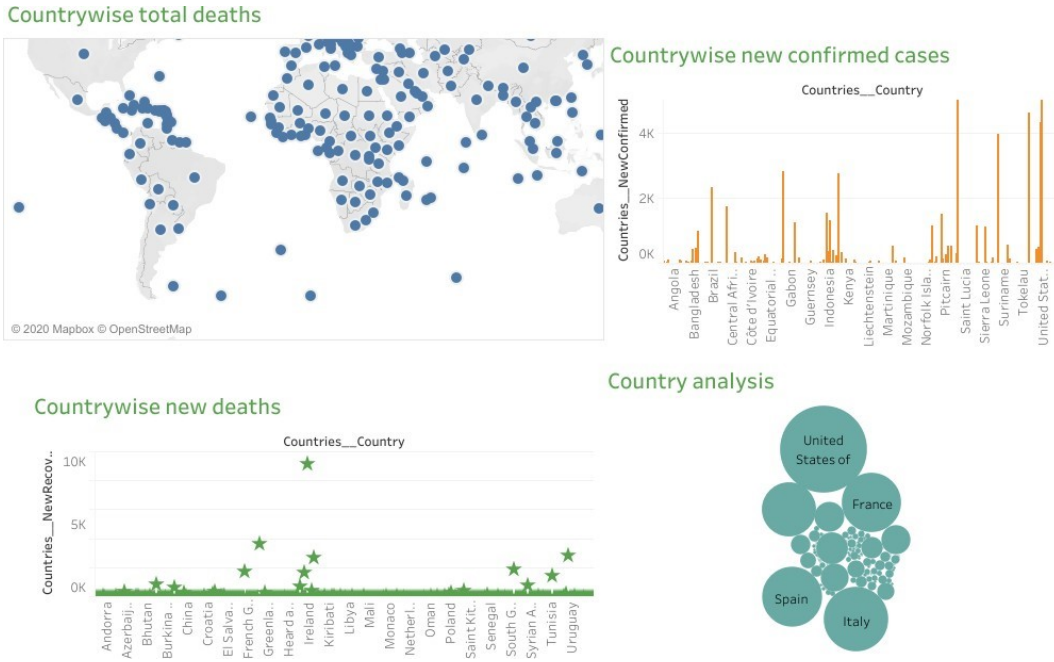


Figure 4.16: Data Visualization : New confirmed cases, country-wise new and total deaths

4.6 Creation of a Global Covid Tracker

Covid-19 On-line Worldwide Tracker

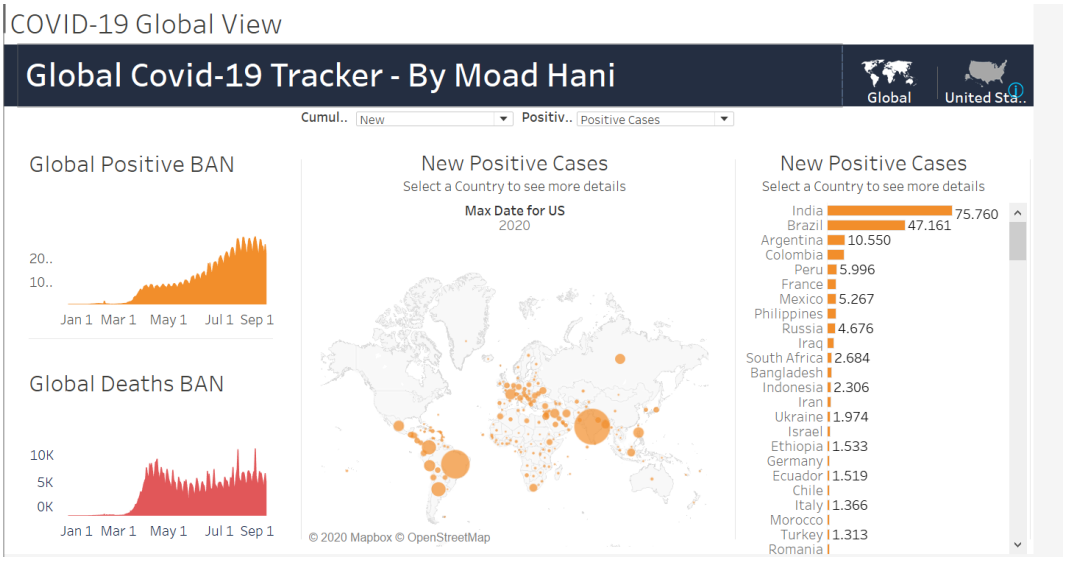


Figure 4.17: Global Covid-19 Tracker

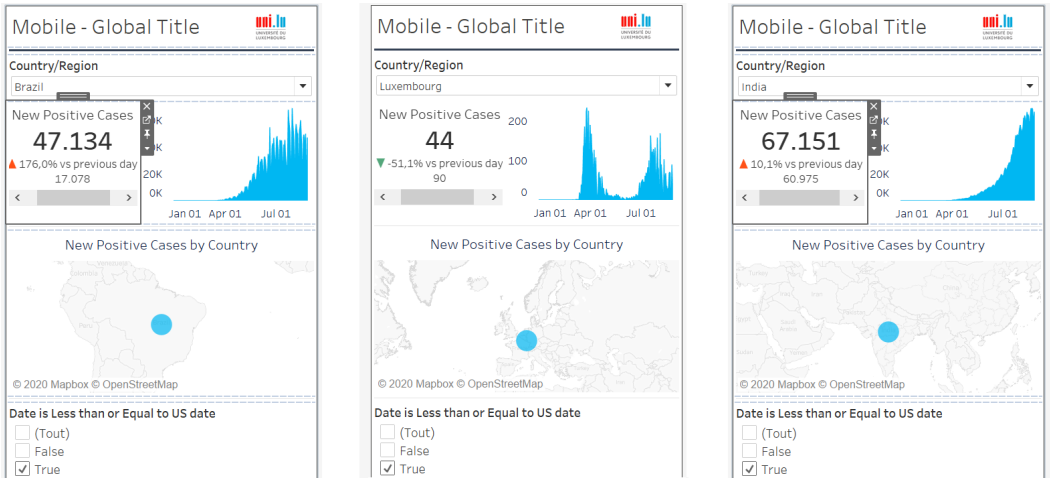


Figure 4.18: Covid-19 Tracking in Brazil, Luxembourg and India

Covid-19 On-line USA Tracker

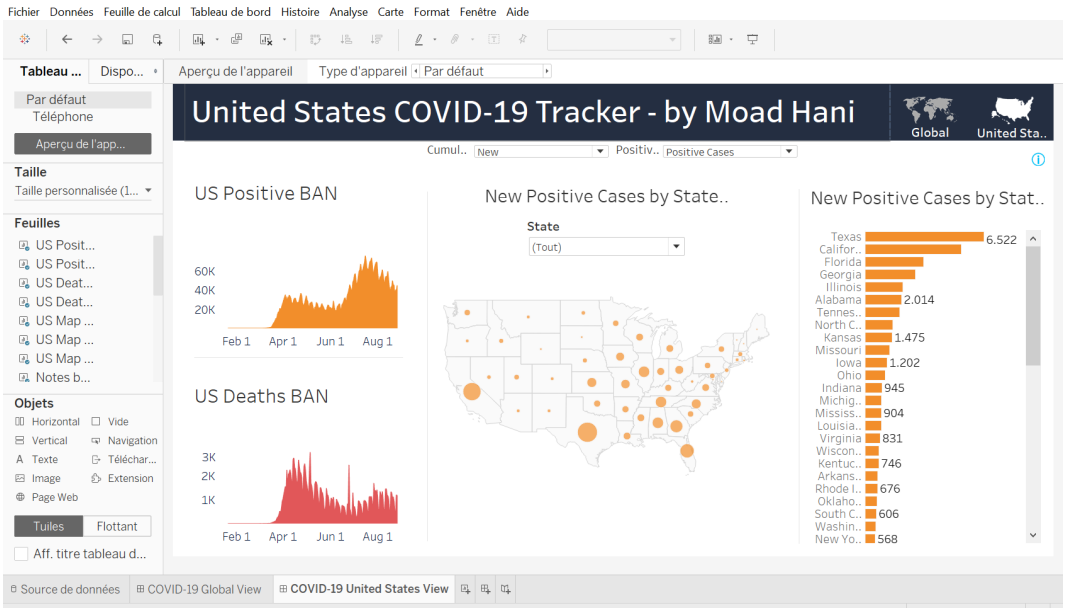


Figure 4.19: USA - Recent Data Visualization (28 August)

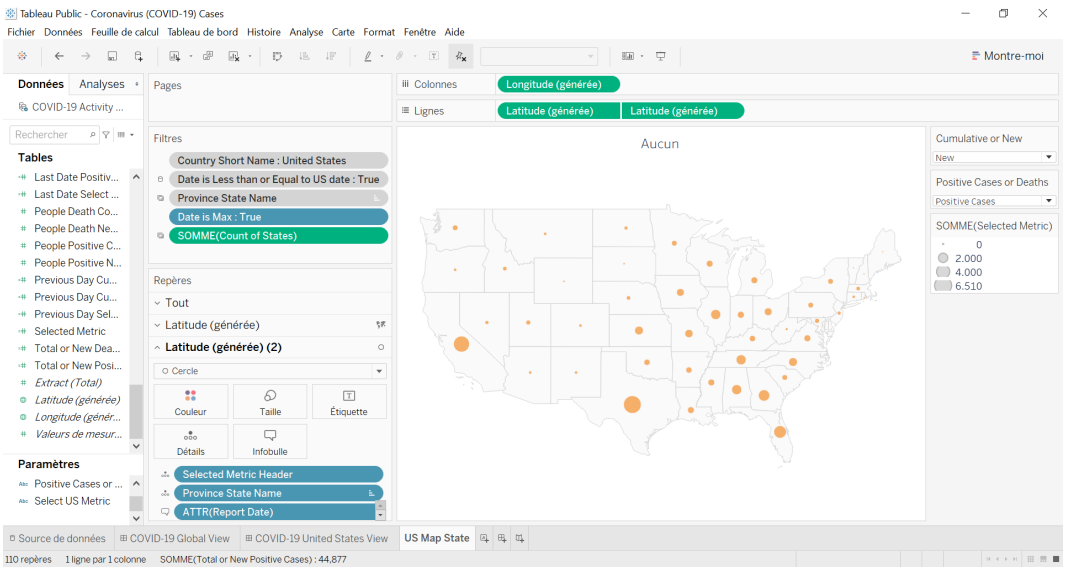


Figure 4.20: Creation of many metrics and parameters afferent to USA Data

4.7 Tools used

Nifi -nifi-1.10.0

Hadoop -hadoop-2.7.3

Hive-apache-hive-2.1.0

Spark-spark-2.4.5

Zookeeper-zookeeper-2.3.5 6

Kafka-kafka-2.11-2.4.0

Airflow-airflow-1.8.1

Tableau

4.8 Known errors and resolutions

Everytime consume kafka processor is run, the group id attribute needs to be changed and can be given some random text value.

When we see the below error while starting Hive.

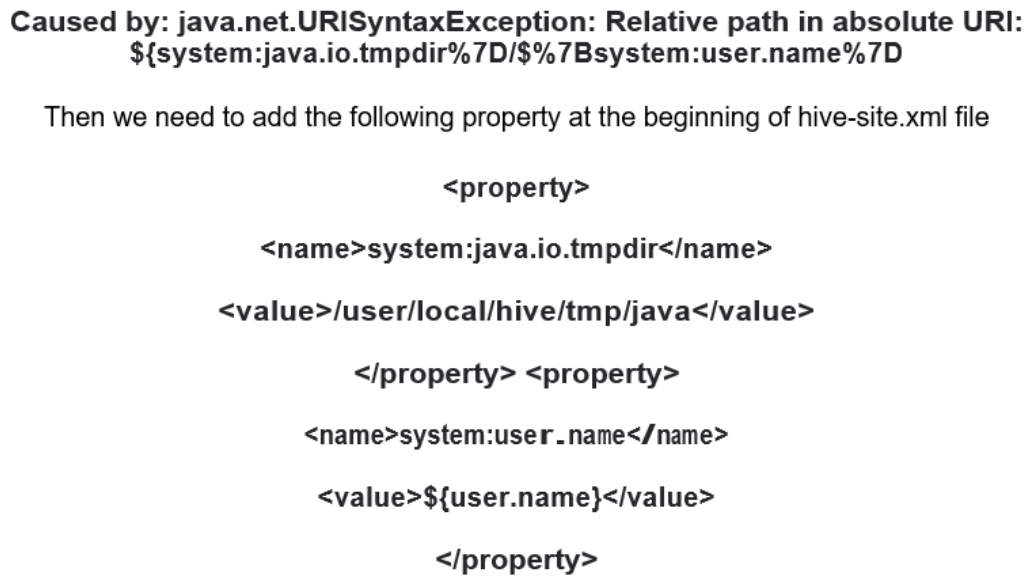


Figure 4.21: Adding a property in hive-site.xml to avoid common error

Conclusion

This project was an opportunity for me to put into practice my knowledge in math and programming and use technologies such as Hive, Spark, Kafka and AWS, etc.

The study, based on deterministic modeling (explaining effects by their underlying or latent causes), allowed us to draw the following conclusions:

1 - One cannot approach a scientific problem in depth without understanding it by using mathematics and algorithmics to model the phenomene by associating specific parameters to it.

2 - Big data analytics technologies are key elements that any data scientist is supposed to master to solve scientific problems centered on structured massive data (such as CSV files) or unstructured data such as complex objects like tweets. (They have several attributes).

3 - Visualization tools have also allowed us to have a better visibility of the present and future evolution of the disease, as well as its characterization.

Bibliography

[1] "A contribution to the mathematical theory of epidemics". In: *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character* 115.772 (Aug. 1927), pp. 700–721. DOI: 10.1098/rspa.1927.0118. URL: https://doi.org/10.1098/rspa.1927.0118.

[2] G. Chowell et al. "The basic reproductive number of Ebola and the effects of public health measures: the cases of Congo and Uganda". In: *Journal of Theoretical Biology* 229.1 (July 2004), pp. 119–126. DOI: 10.1016/j.jtbi.2004.03.006. URL: https://doi.org/10.1016/j.jtbi.2004.03.006.

[3] "Histoire et Mémoires de l'Académie Royale des Sciences de Paris, p. 1-45, 1760 (1766) ; édité dans Die Werke von Daniel Bernoulli, Band 2, Birkhäuser, 1982, p. 235-267." In: ().

[4] Matt Keeling. "The mathematics of diseases". In: *in Plus Magazine (revue mathématique soutenue par l'Université de Cambridge)* (). URL: https://plus.maths.org/content/os/issue14/features/diseases/index.

[5] *La modélisation au temps du COVID-19*. 2020. URL: https://uclouvain.be/fr/decouvrir/la-modelisation-au-temps-du-covid-19.html.

[6] Mingming Li and Xianning Liu. “An SIR Epidemic Model with Time Delay and General Nonlinear Incidence Rate”. In: *Abstract and Applied Analysis* 2014 (2014), pp. 1–7. DOI: 10.1155/2014/131257. URL: https://doi.org/10.1155/2014/131257.

[7] Maia Martcheva. “Introduction to Epidemic Modeling”. In: *Texts in Applied Mathematics*. Springer US, 2015, pp. 9–31. DOI: 10.1007/978-1-4899-7612-3_2. URL: https://doi.org/10.1007/978-1-4899-7612-3_2.

[8] “Radouan Yafia, Media24 :Covid-19 les conclusions du modèle SIR du Pr Radouan Yafia”. In: (). URL: https://www.medias24.com/covid-19-voici-les-conclusions-du-modele-sir-applique-au-maroc-etude-9841.html.

[9] “Ross, Ronald. “THE MATHEMATICS OF MALARIA.” British Medical Journal vol. 1,2626 (1911): 1023.” In: ().