

# I. Les notions

Notre sujet porte sur **l'évitement de la propagation des épidémies grâce aux aspects mathématiques**. Il est donc important de revenir sur quelques notions essentielles du thème.

Tout d'abord, rappelons qu'une **épidémie** est l'apparition de la propagation d'une maladie infectieuse contagieuse qui frappe en même temps et au même endroit un grand nombre de personnes. Cette épidémie est toujours accompagnée de statistiques d'un point de vue infectieux (nombre d'infectés, âges touchés, taux de contamination, taux de mortalité, ...) ou vis à vis du remède (statistiques de soins, nombre de guéris, réinfectés ?, ...).

Pour nos travaux, nous étudierons essentiellement **les outils mathématiques tels que les graphes et les statistiques**. Premièrement, les graphes, vont nous permettre de modéliser le parcours de l'épidémie à travers une population.

Nous allons pouvoir grâce à cela étudier différentes configurations, en faisant varier le nombre de personnes contaminées, de personnes saines, de personnes guéries, ... **autant de paramètres qui vont faire varier les statistiques finales**. Nous en venons donc au deuxième outil mathématique présent ici, l'utilisation des statistiques. En effet, pour voir si nous pouvons prévoir et empêcher les différentes épidémies, nous allons utiliser les statistiques présentes lors des différentes simulations, en faisant varier les paramètres. Les statistiques vont donc jouer un rôle majeur pour anticiper une épidémie en approche.

Ensuite, l'intitulé exacte de notre sujet est « Comment éviter la propagation des épidémies grâce aux aspects mathématiques ? ». Il est donc important de comprendre ce que nous entendons par **éviter la propagation**. Ce terme va avoir le sens dans notre sujet de « endiguer la propagation d'un virus, d'une épidémie dans la population jusqu'à trouver une solution viable ». En effet, nous n'allons pas utiliser les mathématiques pour prévoir l'arrivée d'un nouveau virus, d'un variant d'une maladie, etc... mais nous allons essayer de voir comment nous pouvons faire pour qu'une pandémie soit la moins virulente possible jusqu'à l'obtention d'une solution permettant de l'éradiquer.

## II. Perspectives historiques

Dans de nombreux modèles, une valeur est calculée ou définie afin de comprendre l'évolution de l'épidémie c'est ce qu'on appelle  $R_0$  ou le **taux de reproduction du virus**.

Le concept est créé par les démographes à la fin du XIXe siècle, où il est appelé « taux net de reproduction ». Il désigne alors le taux d'accroissement ou de déclin d'une population.

Ce type de décompte est introduit en médecine par Sir Ronald Ross, médecin britannique et spécialiste du paludisme, qui démontre en 1911, avec un simple modèle, que la progression de la maladie peut être freinée efficacement une fois que la population de moustiques descend sous un seuil critique. **Ross est alors l'un des premiers épidémiologistes à proposer un modèle mathématique.**

Mais le concept du  $R_0$  est plus précisément défini en 1952 par George MacDonald, un médecin britannique lui aussi spécialiste du paludisme, pour décrire la transmission de cette maladie mortelle qui deviendra une priorité pour l'Organisation mondiale de la santé au milieu des années 1950.

Plus tard, en 1975, l'Allemand Klaus Dietz et l'Américain Herbert W. Hethcote vont contribuer à élargir la notion de  $R_0$  aux maladies contagieuses et à populariser son usage comme une « métrique » précieuse dans la compréhension et l'anticipation des épidémies.

Afin d'arriver à cette anticipation des virus et épidémies, de nombreux modèles ont servi de fondations afin d'arriver à ce que l'on a aujourd'hui.

Le modèle de Reed-Frost est un des plus vieux modèles mathématiques des épidémies, il date de 1929. Il est très simpliste, mais il permet d'introduire des notions essentielles et d'obtenir une formule mathématique importante.

Dans les années 1920, le mathématicien Lowell Reed et le médecin Wade Hampton Frost ont mis au point un modèle pour la propagation des maladies, utilisé dans leurs cours de biostatistique et d'épidémiologie à l'université Johns Hopkins. Ce n'est qu'en 1950 que la formulation mathématique a été publiée.

Le modèle est une extension de ce qui a été proposé par H.E. Soper en 1929 pour la rougeole. Le modèle de Soper était déterministe, c'est-à-dire que tous les membres de la population étaient également sensibles à la maladie et avaient la capacité de la transmettre. Le modèle est également basé sur la loi d'action de masse, de sorte que le taux d'infection à un moment donné était proportionnel au nombre de personnes sensibles et infectieuses à ce moment-là. Il est efficace pour des populations moyennement grandes, mais il ne tient pas compte des infectieux multiples qui entrent en contact avec le même individu. Par conséquent, dans les petites populations, le modèle surestime grandement le nombre de personnes sensibles qui deviennent infectées.

Reed et Frost ont modifié le modèle de Soper pour tenir compte du fait qu'un seul nouveau cas serait produit si une personne sensible particulière entrait en contact avec

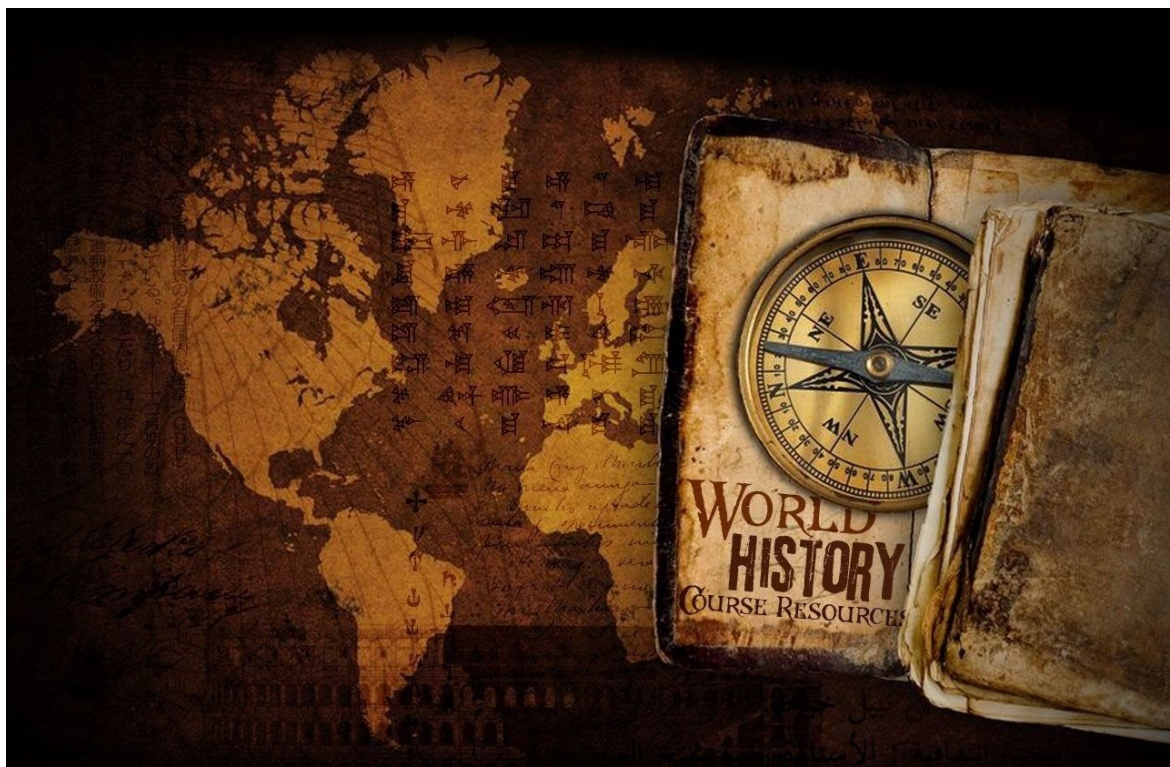
deux cas ou plus. Le modèle Reed-Frost a été largement utilisé et **a servi de base au développement d'études de simulation plus détaillées sur la propagation des maladies.**

Une autre approche fondatrice fut celle des modèles compartimentaux créés en 1927 par le médecin militaire et épidémiologiste Anderson Gray McKendrick et le biochimiste William Ogilvy Kermack.

Le principe est de diviser la population en **classes épidémiologiques** telles que les individus susceptibles d'être infectés, ceux qui sont infectieux, et ceux qui ont acquis une immunité à la suite de la guérison.

Depuis, cette approche est utilisée pour modéliser de nombreuses maladies, et continue d'être un sujet de recherche actif en prenant en compte de nouveaux éléments tels que les découvertes de la science des réseaux.

Il s'agit des premiers **modèles SIR** qui seront pleinement utilisés pour les épidémies qui verront le jour dès les années 1980.



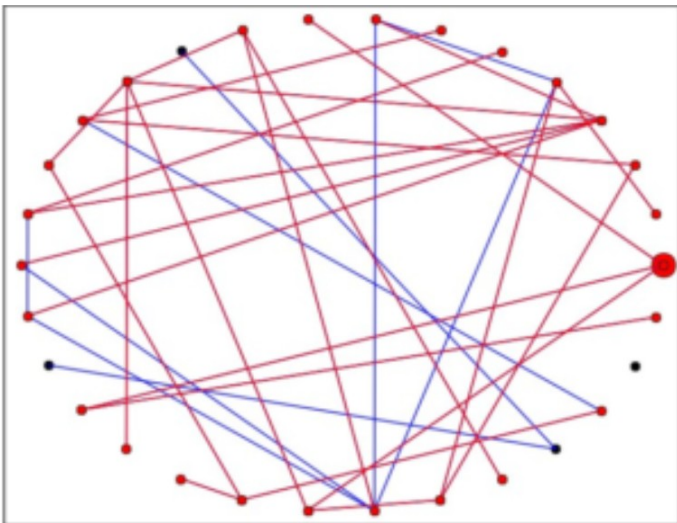
### III. Résultats mobilisés

On cherche à analyser des résultats sous forme de statistiques comme le nombre de contaminations. Ces résultats seront basés sur différents modèles (Erdős-Rényi, SIR, Reed-Frost,...).

Le modèle de Erdős-Rényi consiste en un graphe aléatoire dont les nœuds représentent des personnes, et les liens des contacts entre les personnes.

L'objectif de ce modèle est de trouver l'instant 'T' où le seuil épidémique est atteint.

Le problème majeur de ce modèle est la répartition des liens qui est d'environ 10 liens par personnes. Or en réalité ces liens sont très homogènes mais très variables.



Les points noirs représentent les personnes, les liens en bleus, les contacts sains et en rouge, les contacts ayant entraînés des contaminations

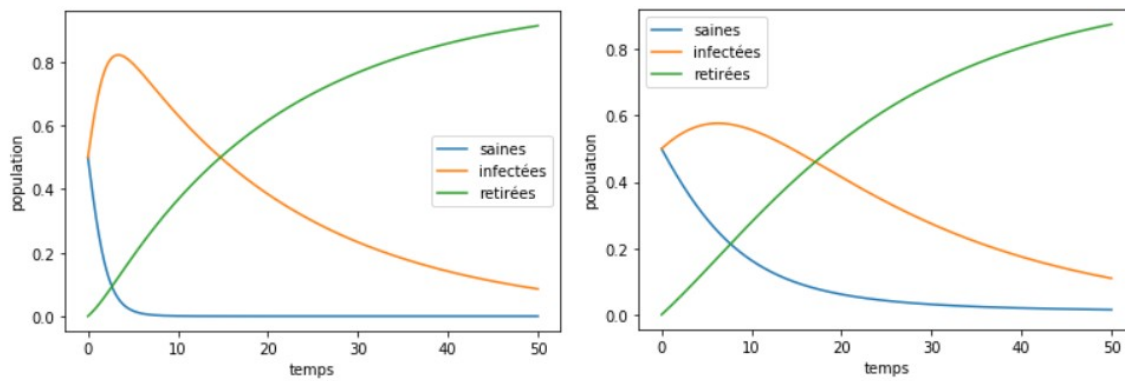
Le modèle de SIR est un graphique avec 3 courbes représentant le taux d'infectés, de sains et de retirés, c'est-à-dire le nombre de personnes atteintes du virus, le nombre de personnes pas encore atteintes et le nombre de patients guéris.

Ce modèle s'intéresse plus au taux de reproduction  $R_0$ .

Au début de l'épidémie, l'expression de  $R_0$  est  $\beta/\gamma$  puisque  $1/\gamma$  représente la durée moyenne de la maladie et qu'au début, les personnes rencontrées sont presque toutes saines.

$\beta$  représente le taux de transmission, et  $\gamma$  le taux de guérison.

Il est considéré avec le modèle SEIR (qui prend en compte les morts) comme les meilleurs modèles existants sans trop de limites.



Comparaison du pic en modifiant le taux de transmission

Le taux de transmission est de 0,9 à gauche et de 0,2 à droite. Le taux de guérison est fixé à 0,1.

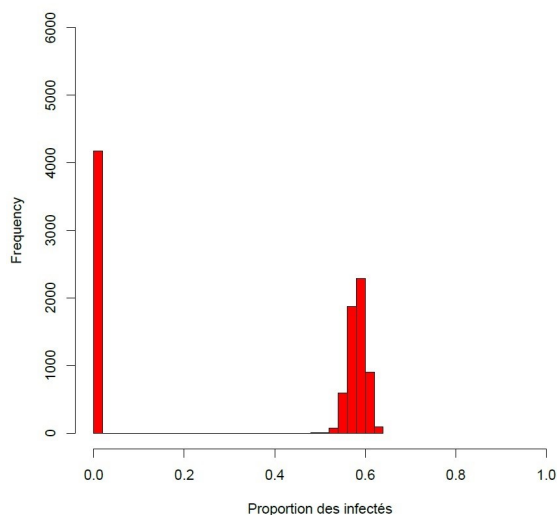
Nous avons trois courbes de couleurs différentes représentant les trois états possibles d'une personne.

La somme est alors égale à 1 car l'axe des ordonnées représente la proportion de population par états : soit Saines, Infectées ou encore Retirées. Ces courbes agissent en fonction du taux de transmission  $\beta$  et du taux de guérison  $\gamma$ .

Le modèle de Reed-Frost, qui est un des premiers modèles existants, est en réalité un ancêtre du modèle SIR.

Il reprend les mêmes termes et arguments : Sains, Infectés, Retirées et  $R_0$ .

Néanmoins, il s'intéresse uniquement à la proportion de personnes infectées.



On a ici une représentation des résultats de 10 000 simulations du modèle de Reed-Frost, le nombre de simulations(ordonnée) et les résultats, soit la proportion des infectés (abscisse)

## IV. La démarche

Comment avons-nous travaillé ? :

Grâce à la partie précédente, c'est-à-dire les informations que nous avons pu récoltées, nous avons alors pu commencer à travailler sur notre sujet, et donc à chercher une solution à la question posée.

Nous avons donc en parallèle écrit le blog, décrypté et compris les informations vues antérieurement et nous avons également codé des modélisations.

### IV.1 Synthèses

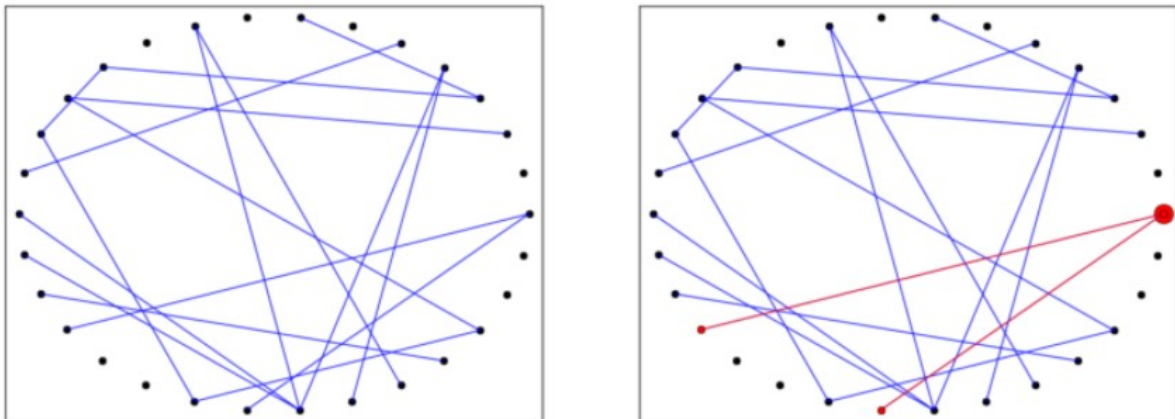
Propagation d'épidémies et graphes aléatoires :

Comprendre la propagation est d'une importance capitale pour traiter et contrôler l'épidémie.

On utilise des simulations / modèles appelés graphes aléatoires

2 exemples de modèles

Modèle de Erdős-Rényi (plus simple)



Chaque nœud = une personne.

- On part d'un nœud infecté
- Une proba d'être lié à un autre nœud (de le croiser à un moment)
- Une proba de propager l'infection quand on est lié
- Quand la majorité des nœuds deviennent infectés à la fin de la simulation on a atteint un seuil épidémique
- Ce seuil dépend de la probabilité de propagation



Ici le seuil a été atteint (probabilité de propagation de 8%)

On peut estimer que le seuil correspond à l'inverse de la taille de la population.

Donc pour 30 individus il vaudra 1/30, Ou ~3%.

Modèle de réseaux "scale-free" (plus réaliste)

Problème des modèles Erdős-Rényi: précis quand les individus ont en moyenne une dizaine de liens

Dans la réalité, on a une répartition hétérogène des liens (certains en ont très peu, et d'autres en ont

énormément), donc ce modèle est très limité.

La distribution dans un réseau homogène de type Erdős-Rényi suit une loi de Poisson, dans un

réseau réel il s'agit d'une loi de puissance, par exemple:

$$P(k) \frac{1}{k^\gamma}$$

où lambda vaut en général entre 2 et 3.

On appelle les réseaux suivant une distribution par loi de puissance les réseaux "scale-free"

Pour représenter ces réseaux on va les construire progressivement en suivant une idée simple: la

probabilité qu'un nouvel individu se connecte à un individu déjà présent est proportionnelle au

nombre de connexions ce dernier que possède déjà. Donc les individus avec le plus de connexions

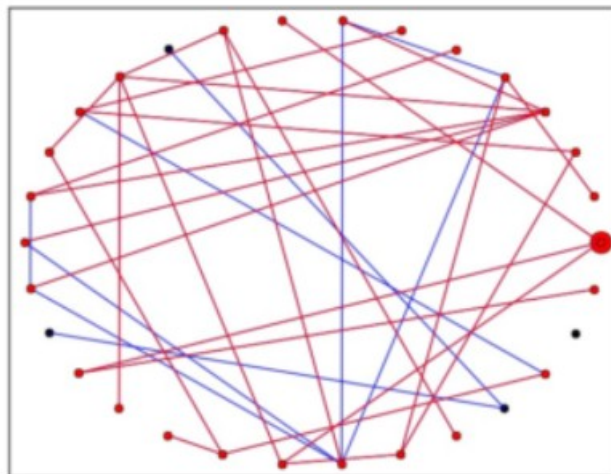
sont ceux avec le plus de désir de se connecter d'avantage.

En application réelle on devra également avoir une propagation dynamique, c'est à dire une

transmission non-instantanée, la mort ou la guérison du patient, etc.

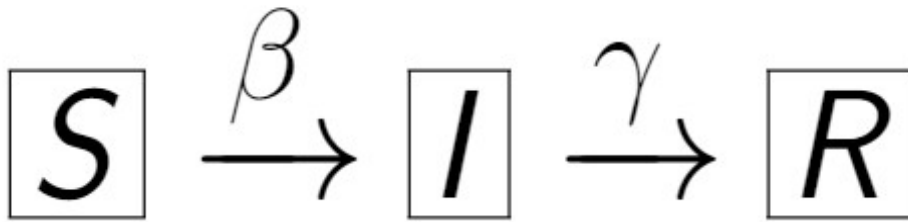
Il a été montré que dans un réseau scale-free, le seuil épidémique est nul, peu importe le contexte

l'épidémie finira toujours par dégénérer.



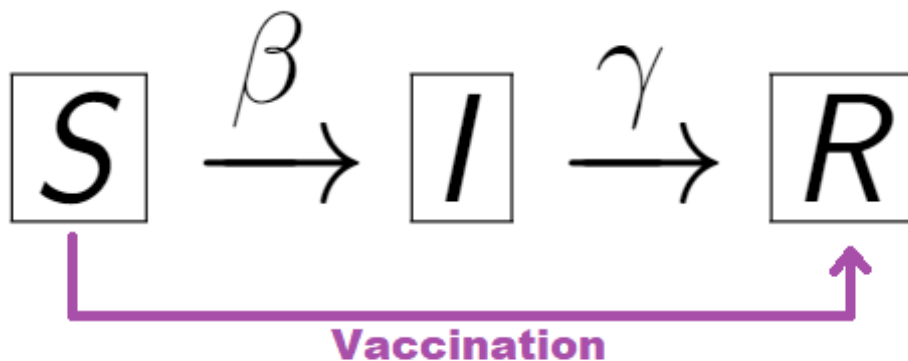
Par la suite, nous avons une fait une synthèse permettant d'expliquer comment nous allons pouvoir modéliser les différents états, sains, infectés et guéris, et ainsi de prédire l'évolution de la maladie au cours du temps, et elle a pour but principal de guider les dirigeants dans la prise de décision.

Le modèle SIR : modèle à compartiments, on divise la population en plusieurs catégories (Sain, Infectés, Retirées) le modèle SIR ne s'occupe pas directement de prédire la mortalité de l'épidémie, pour cela il faut un autre modèle : le modèle SEIR



Ici,  $\beta$  représente le taux de transmission, et  $\gamma$  le taux de guérison. Mathématiquement, le modèle SIR est donné par le système suivant :

$$\begin{cases} \frac{dS(t)}{dt} = -\beta S(t)I(t) & (1.1) \\ \frac{dI(t)}{dt} = \beta S(t)I(t) - \gamma I(t) & (1.2) \\ \frac{dR(t)}{dt} = \gamma I(t) & (1.3) \end{cases}$$



La politique de vaccination visant à vacciner 100% de la population est quasiment impossible : il faut donc trouver le bon équilibre pour vacciner une partie de la population assez importante pour ralentir, puis arrêter l'épidémie. C'est ce qu'a permis le modèle SIR.

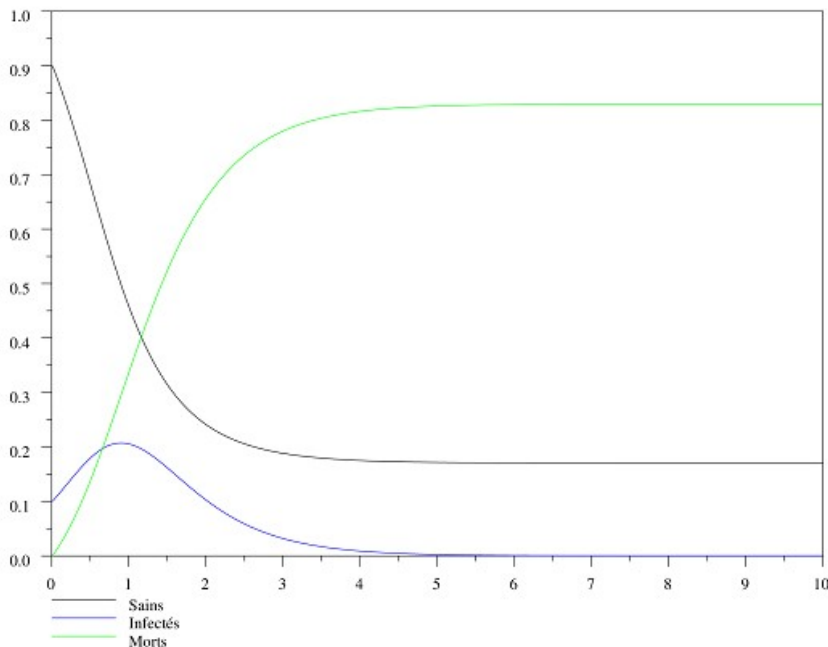
Le taux de reproduction  $R_0$  est le nombre moyen de cas secondaires produits par un individu infectieux au cours de sa période d'infection. Au début de l'épidémie, l'expression de  $R_0$  est  $\beta/\gamma$  puisque  $1/\gamma$  représente la durée moyenne de la maladie et qu'au début, les personnes rencontrées sont presque toutes saines. Ce taux peut être abaissé par des mesures comme la distanciation sociale, le confinement ou la fermeture de certains lieux.

Comparaison du pic en modifiant le taux de transmission Le taux de transmission est de 0,9 à gauche et de 0,2 à droite. Le taux de guérison est fixé à 0,1.



- 2 simulation interactive (à coder dans notre sujet)

Un petit programme SIR permet de tracer l'évolution des populations S, I et R en fonction du temps. Selon les valeurs des paramètres  $\alpha$  et  $r$ , les comportements sont complètement différents : si  $\alpha$  est trop élevé par exemple, les individus infectés meurent presque tout de suite et n'ont pas le temps d'en infecter beaucoup d'autres. Voici un exemple d'évolution des populations au cours du temps, obtenu avec SIR pour les valeurs  $r=4$  et  $\alpha=2$ .



Par la suite, avec tous ces résultats et ces explications qui nous ont permis de mieux comprendre le sujet, nous avons alors pu réaliser notre propre modélisation, en la codant.

Nous nous sommes inspirés du code suivant afin de réaliser notre propre travail, disponible en annexe et dans la partie V.

```

import matplotlib.pyplot as plt
import numpy as np
import datetime
def calcul_sain(infecte, b, sain, pop):
    snew = ((-b * sain * infecte) / pop) + sain
    return snew

def calcul_infecte(infecte, b, sain, l, pop):
    inew = (((b*sain*infecte)/pop) - ((1/l)*infecte))+infecte
    return inew

def calcul_retabli(infecte, l, retabli):
    rn timer = ((1/l)*infecte)+retabli
    return rn timer

#Déclaration des variables de départ

infecte=int(input("Infectés initiaux = "))
retabli=int(input("Rétablis initiaux = "))
pop=int(input("Population totale = "))
sain=pop-retabli-infecte

beta=float(input("Veuillez entrer le taux de transmission "))
lamb=float(input("Veuillez entrer la durée moyenne d'infection en jour "))
tmax=int(input("Veuillez entrer la durée en jour maximum de l'étude "))
#Déclaration des tableaux contenant le nombre de sains, infectés, rétablis, au cours du te
tabs = []
tabi = []
tabr = []

#Calcul de S(t), I(t), R(t) et ajout dans le tableau

for t in range(0,tmax):
    # On ajoute les valeurs de s, i et r
    tabs.append(sain)
    tabi.append(infecte)
    tabr.append(retabli)
    # On calcul grace aux fonctions fonction 1, fonction 2 et fonction 3, les nouvelles valeurs
    sain = calcul_sain(infecte, beta, sain, pop)
    infecte = calcul_infecte(infecte, beta, sain, lamb, pop)
    retabli = calcul_retabli(infecte, lamb, retabli)

#On trace les courbes

#On définit les valeurs de x pour toutes les courbes
x = np.array(range(0,tmax))

#On définit la suite de valeur en y pour chacune des courbes
ys = np.array(tabs)
yi = np.array(tabi)
yr = np.array(tabr)

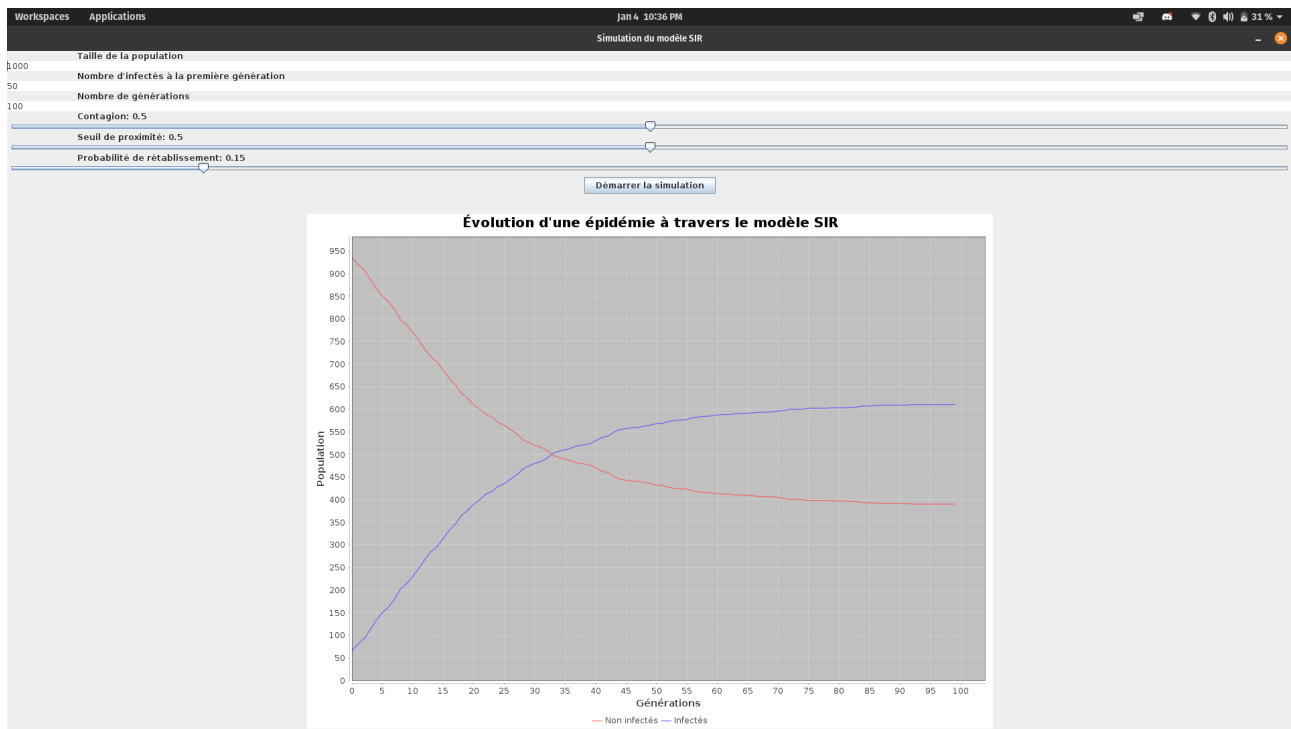
#On ajoute les trois courbes
plt.plot(x, ys, label="sains")
plt.plot(x, yi, label="infectés")
plt.plot(x, yr, label="rétablis")
plt.legend()

#On ajoute un titre et un label aux axes
plt.title("Modèle S.I.R")
plt.xlabel("Durée totale (en jours)")
plt.ylabel("Taux (en %)")

#On crée le nom du fichier et on sauvegarde notre courbe dans une image .png
nom = "SIR"+str(datetime.datetime.now())+".png"
plt.show(nom)
plt.close()

```

Après de nombreuses séances de travail, notre code à fonctionner et nous a permis de réaliser la modélisation suivante :



Voici comment nous avons procédé afin de répondre à notre problématique. Nous allons voir dans notre prochaine partie comment fonctionne notre programme, et quelles sont les conclusions que nous avons pu offrir à notre sujet.

## V. Résultats

Comme expliqué précédemment, nous avons créé un programme permettant de modéliser n'importe quelle épidémie, en nous appuyant sur la théorie des graphes, sur l'étude des statistiques, sur les modèles mathématiques, et plus précisément le modèle SIR. Et enfin, ce qui nous a été le plus utile est forcément les formules, permettant de calculer le nombre de personnes saines, infectées et guéries, sachant que ces populations sont reliées entre elles (cf IV.)

**Nous avons alors par la suite pu nous appuyer sur un programme déjà réalisé trouvé sur internet, via le site [scipython \(lien en annexe\)](#).** Après nous être appuyé sur le code, nous avons créé notre propre programme, ayant le code suivant :

```
import java.lang.reflect.Array;
import java.util.ArrayList;
import java.util.concurrent.ThreadLocalRandom;

public class Terrain {

    ArrayList<Individu> population = new ArrayList<>();
    int infectes;

    public Terrain(int taille, int infectes) {
        for (int i = 0; i < taille; i++) {
            population.add(new Individu());
            population.get(i).setDistance(Math.random());
        }
        for (int j = 0; j < taille; j++) {
            int randomNum = ThreadLocalRandom.current().nextInt(0,
taille);
            population.get(j).setPlusProche(population.get(randomNum));
        }
        for (int k = 0; k < infectes; k++) {
            int randomNum = ThreadLocalRandom.current().nextInt(0,
taille);
            population.get(randomNum).setStatut("I");
        }
    }

    public ArrayList<Individu> getPopulation() {
        return population;
    }

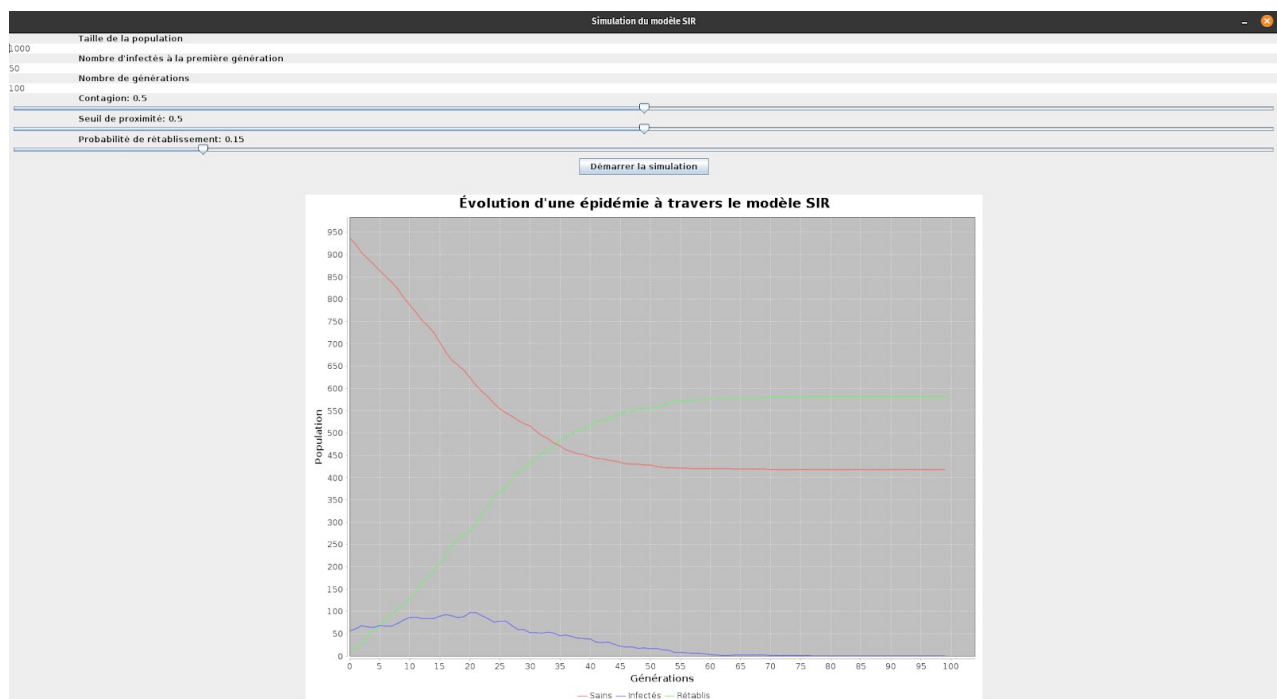
    public void setPopulation(ArrayList<Individu> population) {
        this.population = population;
    }
}
```

```

        public void nextGen(Terrain t, double contagion, double
seuilProx, double probRetablissement) {
            ArrayList<Individu> newPopulation = new
ArrayList<Individu>();
            newPopulation = t.getPopulation();
            int popSize = t.getPopulation().size();
            for (int i = 0; i<popSize; i++) {
                int randomNum = ThreadLocalRandom.current().nextInt(0,
newPopulation.size());
                Individu e = newPopulation.get(i);
                if (e.getStatut() == "S" && e.getDistance() <
seuilProx && e.getPlusProche().getStatut() == "I") {
                    if (Math.random() < contagion) {
                        e.setStatut("I");
                    }
                }
                if (e.getStatut() == "I") {
                    if (Math.random() < probRetablissement) {
                        e.setStatut("R");
                    }
                }
                e.setDistance(Math.random());
                e.setPlusProche(newPopulation.get(randomNum));
                newPopulation.set(i, e);
            }
            t.setPopulation(newPopulation);
        }
    }
}

```

En exécutant ce programme, on obtient alors des courbes comme celle-ci :



Notre programme prend en paramètre la taille de la population, le nombre d'infectés au départ, le taux de contagion, le seuil de proximité, ainsi que la probabilité de rétablissement. Autant de paramètre à faire varier et de résultats différents possibles qui peuvent alors être étudiés.

Avec ce programme, très simple puisque façonné par des élèves d'IUT, **nous pouvons déjà faire des prévisions statistiques sur des épidémies**, et cela nous permet donc d'affirmer, afin de répondre à notre problématique : « Comment éviter la propagation des épidémies grâce aux aspects mathématiques ? », que la propagation des épidémies peut donc être mesurée et contrôlée, comme ce qui est actuellement fait avec la Covid-19.