

L'activité va permettre d'implémenter le modèle S.I.R. et de l'appliquer à l'épidémie du Covid-19.

1. Introduction

Le film **Contagion**, sorti en 2011, relate une pandémie dévastatrice qui explose à l'échelle du globe...

Bande annonce : <https://www.youtube.com/watch?v=wMDoUFajq1I>

2. Le modèle S.I.R.

L'étude des épidémies s'appelle l'épidémiologie. Elle permet d'étudier, comprendre, prévoir l'évolution d'une épidémie afin d'en limiter l'impact et plus particulièrement la mortalité associée.

Le modèle S.I.R. a été mis au point par KERMACK et MCKENDRICK en 1927 afin d'expliquer la peste de Bombay, en Inde, ayant frappée de 1905 à 1906.

Les scientifiques ont décidé de suivre l'évolution au cours du temps (t) de trois catégories de population S, R et I :

- S (Susceptible) la population **Saine** n'ayant pas encore eu la maladie.
- I (Infected) la population **Infectée** et contagieuse.
- R (Recovered) la population **Rétablie**. Il s'agit des personnes immunisées ou décédées.

Le modèle comporte plusieurs règles :

- La population est fixe : pas de phénomènes démographiques (naissances, décès, immigration et émigration).
- Ne plus être sain veut forcément dire devenir infecté.
- Les personnes infectées sont toutes infectieuses.
- Chaque personne rétablie est rétablie à jamais.

Le modèle comporte plusieurs paramètres :

- β = Contacts / jours \times Probabilité de transmission du virus.
- λ = Jours d'infections.
- Il existe un coefficient R_0 équivalent au nombre de cas moyen qu'un infecté infectera.
 - R_0 = Contacts / jours \times Probabilité de transmission du virus \times Jours d'infection.
- λ est la durée d'infection moyenne où l'on peut transmettre le virus.
- Chaque jour $1 / \lambda$ Infectés deviennent rétablis après avoir passé λ jours infecté précédemment.
- Le modèle sous-entend que la personne n'est plus un cas possible d'infection. Il ne s'agit donc pas de la durée de la maladie mais de la durée où l'on est contagieux, sans le savoir.
- P la population totale en question (le plus souvent un Pays).

Le modèle définit la manière dont évolue chaque catégorie S, I et R. Etudier une évolution c'est étudier une variation. Or une variation en mathématiques se définit par la dérivée. Ici elle se fait en fonction du temps t.

Pour rappel $f'(x) = \frac{df}{dx}$

La variation de personnes rétablies ($\frac{dR}{dt}$) correspond au nombre de personnes ayant dépassé le temps d'infection λ (personnes immunisées ou décédées). Ainsi chaque jour on a une fraction $1 / \lambda$ de population qui passe de infecté à rétablie, soit :

$$\frac{dR}{dt} = \frac{1}{\lambda} \times I = \frac{I}{\lambda}$$

La variation de personnes infectées ($\frac{dI}{dt}$) correspond au nombre de nouveaux infectés auquel on soustrait ceux passés en rétablis.

Les nouveaux infectés correspondent à la population I aillant infecté les S multiplié par le coefficient d'infection journalier β : soit $S \times I \times \beta$. On divise alors par la population totale P.

$$\frac{dI}{dt} = \frac{\beta \times I \times S}{P} - \frac{I}{\lambda}$$

Enfin le nombre de personnes saines S diminue chaque jour du nombre d'infectés.

$$\frac{dS}{dt} = - \frac{\beta \times I \times S}{P}$$

Pour notre modèle, le nombre de S, I et R actuel dépendent de l'état précédent. Cela s'apparente à une suite puisque les termes $x + 1$ doivent être calculés à partir du terme précédent et de la variation : $f(x + 1) = f(x) + f'(x)$. Dans ce modèle, x est le temps t, ainsi :

$$R(t + 1) = R(t) + \frac{dR}{dt} = R + \frac{I}{\lambda} \quad I(t + 1) = I(t) + \frac{dI}{dt} = I + \frac{\beta \times I \times S}{P} - \frac{I}{\lambda} \quad S(t + 1) = S(t) + \frac{dS}{dt} = S - \frac{\beta \times I \times S}{P}$$

3. Les données pour le Covid-19 en France

Afin d'appliquer le modèle S.I.R. au Covid-19, il est nécessaire de définir les paramètres fixes (β , λ et P) et de départ (à $t = 0$) du modèle $S(0)$, $I(0)$ et $R(0)$.

Entre le moment où la personne est infectieuse et à un contact avec le monde médical (médical ou urgence), il s'écoule 5,8 jours. Après de délai, elle est mise en quarantaine et ne peut infecter personne.

$$\lambda = 5,8$$

Pour la France, le **RO** (Contacts / jours \times Probabilité de transmission du virus \times Jours d'infection) moyen sans confinement est de **2,3** et avec confinement respecté, il est de **1,4**.

D'après le modèle, le taux de transmission $\beta = \frac{RO}{\lambda}$.

- Hors confinement : $\beta = 2,3 / 5,8 = 0,40$
- Avec confinement : $\beta = 1,4 / 5,8 = 0,24$

La population **P** en France en 2020 est de **67 064 000**.

Il y a deux dates clés en France pour le Covid-19, le 24 janvier et le 16 mars.

- Les trois premiers cas sont apparus le 24 janvier
 - $S(0) = P - I(0) - R(0) = P - 3 - 0 = P - 3$
 - $I(0) = 3$
 - $R(0) = 0$
- À la date de prise de décision du confinement, le 16 mars.
 - $S(0) = P - I(0) - R(0) = P - 96\,360 - 85\,214 = P - 181\,574$
 - $I(0) = 96\,360$
 - $R(0) = 85\,214$

4. Implémentation du modèle

4.1. Implémentation des calculs de $S(t + 1)$, $I(t + 1)$ et $R(t + 1)$.

La fonction `calcul_s(i, b, s, p)` calcule et retourne le nouveau nombre de personnes saines (`new_s`) à partir du nombre d'infectés (`i`), du taux de transmission β (`b`), du nombre de personnes saines courant (`s`) et de la population (`p`).

```
def calcul_s(i, b, s, p):  
    new_s = s + -(b * s * i) / p  
    return new_s
```

✍ Implémenter la fonction `calcul_i(i, b, s, l, p)` qui calcule et retourne le nouveau nombre d'infectés (`new_i`) à partir du nombre d'infecté courant (`i`), du taux de transmission β (`b`), du nombre de sains (`s`), du nombre de jours d'infections λ (`l`) et de la population (`p`).

✍ Implémenter la fonction `calcul_r(i, l, r)` qui calcule et retourne le nouveau nombre de rétablis (`new_r`) à partir du nombre d'infecté (`i`), du nombre de jours d'infections λ (`l`) et du nombre de rétablis courant (`r`).

4.2. Rentrer les données de départ

✍ Implémenter les lignes d'interfaces afin de demander à l'utilisateur les données de départ comme ci-dessous.

- Convertir en entier ou en flottant les valeurs saisies.
 - `val = int(input("texte"))`
 - `val = float(input("texte"))`

```
Veillez entrer le nombre d'infectés initial 96360  
Veillez entrer le nombre de rétablis initial 85214  
Veillez entrer la population totale 67064000  
Veillez entrer le taux de transmission 0.24  
Veillez entrer la durée moyenne d'infection en jour 5.8  
Veillez entrer la durée en jour maximum de l'étude 120
```

✍ Implémenter le calcul du nombre de personnes saines (`s`) en fonction de la population (`p`), du nombre d'infectés (`i`) et du nombre de rétablis (`r`).

4.3. Calculer $S(t)$, $I(t)$ et $R(t)$ et stocker les valeurs

✍ En prenant modèle sur le calcul et le stockage de $S(t)$ donnés dans le programme, compléter dans le programme :

- La déclaration des tableaux pour les valeurs de $I(t)$ et de $R(t)$;
- Le calcul de $I(t)$ et de $R(t)$ et l'ajout dans le tableau.

4.4. Tracé des données

Les données sont maintenant calculées. Il faut les afficher grâce à deux bibliothèques.

```
import matplotlib.pyplot as plt  
import numpy as np
```

- `np` (alias pour `numpy`) permet de créer des tableaux à plusieurs dimensions.
- `plt` (alias pour `matplotlib.pyplot`) permet de créer des courbes graphiques à partir de données sous forme `numpy`.

On va dans un premier temps affecter à une variable `x` les valeurs des abscisses, c'est-à-dire la durée de θ à t_{\max} :

- `x = np.array(range(θ , t_{\max})).`

Maintenant on va affecter à des variables les valeurs des ordonnées des trois courbes que l'on souhaite afficher. Le modèle pour la courbe des sains est donné :

- `y_s = np.array(tab_s)`.

✍ Sur le même modèle que ci-dessus, implémenter les lignes qui permettent d'affecter `y_i` et `y_r`.

Maintenant nous allons dire à `plt` de tracer les courbes avec `x` souhaité, `y` souhaité et un label voulu. Ainsi nous traçons la première courbe des sains avec comme `x`, `x` et comme `y`, `y_s` et comme label "saines".

- `plt.plot(x, y_s, label = "saines")`.

✍ Sur le même modèle, implémenter les lignes qui permettent de tracer les courbes infectés et rétablis.

Enfin le code fournit les dernières lignes propres à `plt` qui permettent respectivement de :

- `plt.legend()` : ajouter les légendes définies précédemment.
- `plt.title("Modèle S.I.R.")` : ajouter un titre au graphe.
- `plt.xlabel("Jours")` : ajouter un titre à l'axe des abscisses.
- `plt.ylabel("Nombre de personnes")` : ajouter un titre à l'axe des ordonnées.
- `nom = "SIR.png"` : définir une chaîne de caractère qui fera office de nom de fichier image dans lequel sera sauvegardé le graphique.
- `plt.savefig(nom)` : on sauvegarde les tracés, le graphique.
- `plt.show()` : affichage du graphique
- `plt.close()` : on ferme l'objet qui contient les tracés.

5. Exploitation du modèle

À partir du programme et des données :

- ✍ Réaliser un graphe à partir de l'arrivée du Covid-19 en France et sur 120 jours.
- ✍ Réaliser un graphe à partir de la date de confinement du 16 mars sur 120 jours et **sans** mesure de confinement.
- ✍ Réaliser un graphe à partir de la date de confinement du 16 mars sur 120 jours et **avec** mesure de confinement.
- ✍ Comparer les deux derniers graphes et conclure quant aux mesures prises par le gouvernement.

Sources : <https://ibmathsresources.com/2020/01/28/modelling-the-spread-of-wuhan-flu-coronavirus/>
<https://www.nejm.org/doi/pdf/10.1056/NEJMoa2001316?articleTools=true>
<https://gisanddata.maps.arcgis.com/apps/opsdashboard/index.html#/bda7594740fd40299423467b48e9ecf6>
<http://www.python-simple.com/python-matplotlib/matplotlib-intro.php>
<https://matplotlib.org/index.html>
<https://tribu.phm.education.gouv.fr/>