

Modeling the spreading of West Nile Virus in student campuses

Groupe D: Grégoire Collier, Timothé loos, Thierry Zhang, Nathan Alimi

CentraleSupélec

February 2, 2024

Outline

Introduction

Data

Model

- Model for Paris-Saclay Campuses

- Model for Rennes' campus

Interpretation and Strategy

Difficulties and Points to Improve

Conclusion

Annexes

Introduction

The main goal of our study was to model the evolution of a West Nile type pandemic through compartment model. We had access to data for the three campuses of Paris-Saclay: CS, HEC and Polytechnique for the period: 04/15/2030-06/21/2030.

- ▶ Compartment model (inspired by SIR and SEIR models)
- ▶ Parameters to fit (given, computed or taken from others papers)
- ▶ Number of persons, birds and mosquitoes were given.

What strategy should be adopted to slow down the epidemic that has just started in Rennes?

Data given for Paris-Saclay Campuses

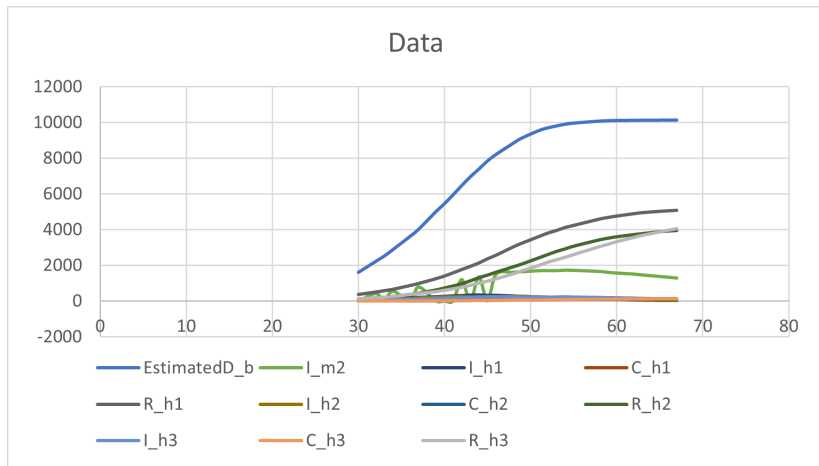
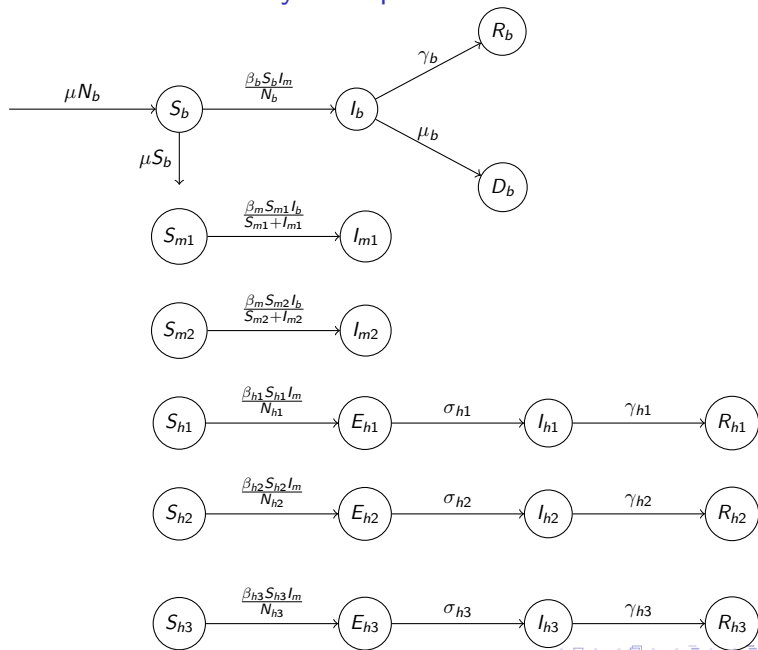


Figure: All the data for HEC(h1), Polytechnique(h2) and CS(h3)

Model

- ▶ First method: Perform regressions using the available data
- ▶ Then making a compartment model for each situation. The difficulty came from the high number of parameters too difficult to fit
- ▶ Some parameters were finally given, others were estimated and the last one have been found on other works.
- ▶ The model describing Paris-Saclay campuses contains 6 population: Birds, 2 populations of mosquitoes and 3 populations of humans (one for each campus: HEC,X,CS)

Model for Paris-Saclay Campuses



Parameters and simplifications

- ▶ $\beta_h, \beta_b, \beta_m$ represents the transmission rate of the virus respectively humans, birds and mosquitoes.
- ▶ γ_b, γ_h represents the recovery rate for birds and humans
- ▶ μ_b is the death rate due to the virus
- ▶ σ_h denotes the infection rate
- ▶ We considered that the total population for each group is constant though it was said that the population of mosquitoes rose. We chose this simplification to ease the model.
- ▶ We chose a frequency dependant model
- ▶ The term $\frac{\beta_{hi} S_{hi} I_m}{N_{hi}}$ captures the transmission from mosquitoes to humans, reflecting the interaction between susceptible humans and infected mosquitoes within the total human population.

Equations

► For Birds :

$$\frac{dS_b}{dt} = \mu N_b - \mu S_b - \frac{\beta_b S_b I_m}{N_b}$$

$$\frac{dI_b}{dt} = \frac{\beta_b S_b I_m}{N_b} - \gamma_b I_b - \mu I_b$$

$$\frac{dR_b}{dt} = \gamma_b I_b$$

$$\frac{dD_b}{dt} = \mu I_b$$

► For Mosquitoes :

$$\frac{dS_{m1}}{dt} = \mu_m N_{m1} - \mu_m S_{m1} - \frac{\beta_m S_{m1} I_b}{S_{m1} + I_{m1}}$$

$$\frac{dI_{m1}}{dt} = \frac{\beta_m S_{m1} I_b}{S_{m1} + I_{m1}} - \mu_m I_{m1}$$

$$\frac{dS_{m2}}{dt} = \mu_m N_{m2} - \mu_m S_{m2} - \frac{\beta_m S_{m2} I_b}{S_{m2} + I_{m2}}$$

$$\frac{dI_{m2}}{dt} = \frac{\beta_m S_{m2} I_b}{S_{m2} + I_{m2}} - \mu_m I_{m2}$$

► For humans :

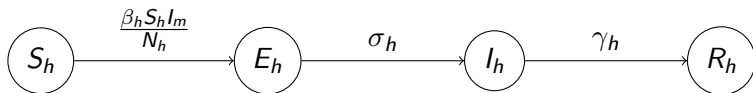
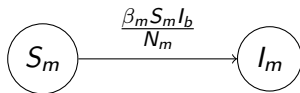
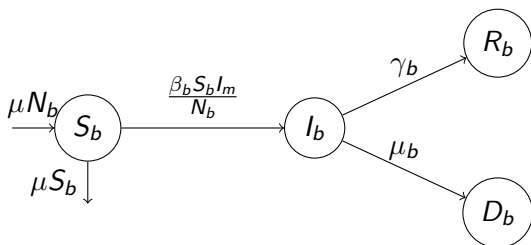
$$\frac{dS_{hi}}{dt} = \mu_h N_{hi} - \mu_h S_{hi} - \frac{\beta_{hi} S_{hi} I_m}{N_{hi}}$$

$$\frac{dE_{hi}}{dt} = \frac{\beta_{h1} S_{hi} I_m}{N_{hi}} - \sigma_{hi} E_{hi}$$

$$\frac{dI_{hi}}{dt} = \sigma_{hi} E_{hi} - \gamma_{hi} I_{hi}$$

$$\frac{dR_{hi}}{dt} = \gamma_{hi} I_{hi}$$

Model for Rennes' campus



Interpretation and Strategy

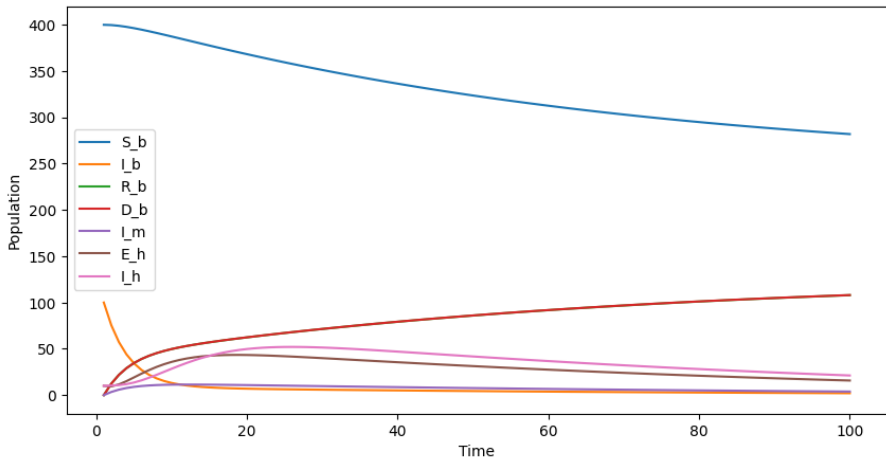


Figure: Results without any strategy

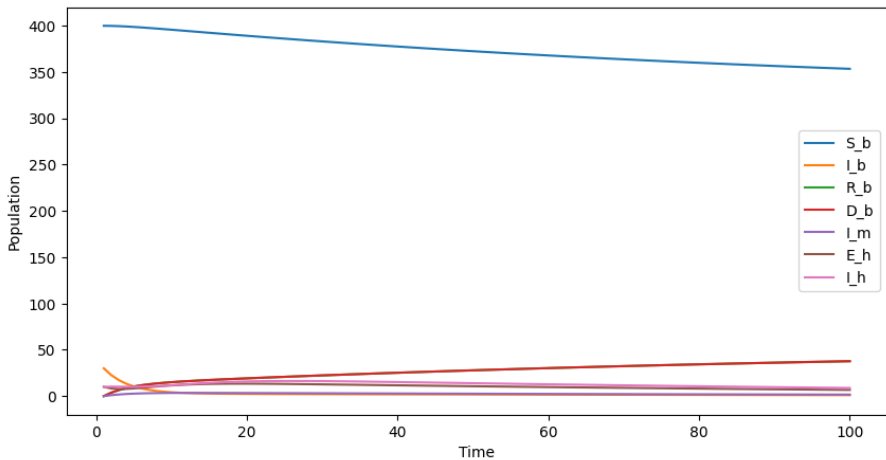


Figure: Results after killing 70 percents of the Canada Goos

- ▶ We use the simulation to adopt an efficient strategy
- ▶ Quarantine a certain proportion of the infected migratory geese
 - ▶ **Pros:** Effective solution
 - ▶ **Cons:** Expensive, complicated to realize
- ▶ Kill a certain proportion of migratory geese
 - ▶ **Pros:** Effective solution
 - ▶ **Cons:** Expensive, complicated to realize and impact on the local ecosystem
- ▶ Buy a mosquito net for each human.
 - ▶ **Pros:** Focus on the moment of infection and avoid it.
 - ▶ **Cons:** Expensive, difficult to ensure that people put it.
- ▶ Relocate students.
 - ▶ **Pros:** Efficient.
 - ▶ **Cons:** Socially unacceptable
- ▶ Quarantine 30 percents of infected geese , Kill 20 percents of infected geese ,provide all students with a mosquito net and relocate 14 percents of students allow to reduce the infected by 75 percents.

Difficulties and Points to Improve

- ▶ Difficulty to make a simple compartmental model with few data and knowledge of the disease.
- ▶ Difficulty to fit the high number of parameters
- ▶ Better organization and team work
- ▶ Difficulties to share tasks knowing the toughness of the model(fitting the parametres)
- ▶ For each task, we divided the work between the members of the group, most of the time, we tried to work by group of 2
- ▶ It was quite efficient but sometimes it was tough to collaborate between the groups.
- ▶ Given the difficulty of the task, we should have improved the model rather than giving up or attempting to fit parameters

Conclusion

- ▶ Our study aimed to model the spread of the West Nile Virus on the Paris-Saclay campuses and propose effective strategies to mitigate its impact.
- ▶ We developed compartmental models for bird, mosquito, and human populations, integrating various parameters and data available for the region.
- ▶ Simulation results highlighted the importance of targeted interventions, such as bird population reduction, in controlling the virus spread.
- ▶ Our work underscores the significance of mathematical modeling in understanding and combating infectious diseases and provides insights for future research and public health strategies.
- ▶ It improved our team work capabilities and allowed to face difficulties in search

Python's script

```
# dY = model_class(Y,t,paramètres)
def model(Y, t, beta_b, gamma_b, mu_b, beta_m, mu_m, beta_h, sigma_h, gamma_h):
    S_b, I_b, R_b, D_b, S_m, I_m, S_h, E_h, I_h, R_h = Y
    dS_b = -beta_b*S_b*I_m/(S_b+I_b+R_b)
    dI_b = beta_b*S_b*I_m/(S_b+I_b+R_b) - mu_b*I_b - gamma_b*I_b
    dR_b = gamma_b*I_b
    dD_b = mu_b*I_b
    dS_m = -beta_m*S_m*I_b/(S_m+I_m)
    dI_m = beta_m*S_m*I_b/(S_m+I_m) - mu_m*I_m
    dS_h = -beta_h*S_h*I_m/(S_h+E_h+I_h+R_h)
    dE_h = beta_h*S_h*I_m/(S_h+E_h+I_h+R_h) - sigma_h*E_h
    dI_h = sigma_h*E_h - gamma_h*I_h
    dR_h = gamma_h*I_h
    return [dS_b, dI_b, dR_b, dD_b, dS_m, dI_m, dS_h, dE_h, dI_h, dR_h]
```

✓ 0.0s


```
# Initialisation
t_array = np.linspace(1, 100, 100)

model = integrate.odeint(model, initial_conditions, t_array, args=(beta_b, gamma_b,

# Tracer des résultats
plt.figure(figsize=(10,5))
plt.plot(t_array, model[:, 0], label='S_b')
plt.plot(t_array, model[:, 1], label='I_b')
plt.plot(t_array, model[:, 2], label='R_b')
plt.plot(t_array, model[:, 3], label='D_b')
#plt.plot(t_array, model[:, 4], label='S_m')
plt.plot(t_array, model[:, 5], label='I_m')
#plt.plot(t_array, model[:, 6], label='S_h')
plt.plot(t_array, model[:, 7], label='E_h')
plt.plot(t_array, model[:, 8], label='I_h')
#plt.plot(t_array, model[:, 9], label='R_h')
```