

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA KHOA HỌC - KỸ THUẬT MÁY TÍNH



MÔ HÌNH HÓA TOÁN HỌC ()

Nhóm: ... —- Bài tập lớn

Mô hình SIR trong dự báo COVID-19

GVHD: Nguyễn An Khương
Huỳnh Tường Nguyên
SV thực hiện: Tăng Minh Nhật – 1813380
Đỗ Lê Thiên Ân – 1810823
Lê Hồng Phong – 1813518
Đặng Thành Ngân – 1813187
Huỳnh Thiên Trình – 1810615

Tp. Hồ Chí Minh, Tháng 7/2020

Mục lục

1 Kiến thức chuẩn bị	3
1.1 Tổng quan về R và Python	3
1.2 Phương pháp xấp xỉ Euler trong giải hệ phương trình vi phân.	3
1.3 Giải thuật Metropolis-Hastings	5
1.3.1 Phương pháp Monte Carlo	5
1.3.1.a Sơ lược	5
1.3.1.b Phương pháp chung	5
1.3.2 Sơ lược về xích Markov	5
1.3.2.a Quá trình Markov	5
1.3.2.b Xích Markov	6
1.3.2.c Xích Markov Monte Carlo (MCMC)	6
1.3.3 Giải thuật Metropolis-Hastings	6
1.3.3.a Phương pháp	6
1.4 Một số kiến thức khác về xác suất và thống kê	7
1.4.1 Xác suất tiên nghiệm	7
1.4.2 Xác suất hậu nghiệm	7
1.4.3 Xác suất Bayes	8
1.4.3.a Lý thuyết	8
1.4.3.b Một số bài toán về xác suất tiên nghiệm	8
1.4.3.c Một số bài toán về xác suất hậu nghiệm	9
2 Cách xây dựng mô hình SIR	9
2.1 Giới thiệu chung về mô hình.	9
2.2 Trường hợp liên tục	9
2.3 Trường hợp rời rạc.	10
2.4 Những vấn đề liên quan	11
2.4.1 Phương pháp xấp xỉ Euler trong giải hệ SIR	11
2.4.2 Ước lượng hệ số β và γ .	11
3 Bài toán 2. Áp dụng thuật toán Euler để tìm nghiệm của hệ SIR	12
3.1 Giới thiệu	12
3.2 Tham số đầu vào	12
3.3 Kết quả đầu ra	13
3.4 Chương trình	13
3.5 Kết quả và phân tích	16
3.5.1 Trường hợp 1	16
3.5.2 Trường hợp 2	17
3.5.3 Trường hợp 3	18
3.5.4 Trường hợp 4	19
3.5.5 Trường hợp 5	20
3.6 Nhận xét	20
4 Bài toán 3. Lấy mẫu sử dụng thuật toán Metropolis-Hastings	21
5 Bài toán 4. Tìm hệ số R_0 của khu vực tự chọn	23
6 Nhật ký công việc - Log Assignment	23



7 Kết luận	25
Tài liệu	25

Bài báo cáo này trình bày lời giải một số bài toán về thống kê mô tả và một số bài toán liên quan đến các mô hình xác suất rời rạc cổ sử dụng R.

Trong phần đầu, chúng tôi nhắc lại một số kiến thức và kết quả cơ bản về thống kê mô tả, và các mô hình xác suất rời rạc thông dụng, cùng với một số hàm R thông dụng mà chúng tôi sử dụng để tính toán kết quả trong các lời giải của các bài toán trong Phần 2.

Đối với mỗi bài toán Phần 2, trước hết chúng tôi trình bày lời giải truyền thống, bằng cách dùng các công thức và kết quả đã được trình bày trong Phần 1. Sau đó chúng tôi tính toán lại kết quả bằng các hàm R. Các biểu đồ (nếu có) cũng sẽ được vẽ bằng R và nhúng vào LaTeX.

1 Kiến thức chuẩn bị

1.1 Tổng quan về R và Python

...

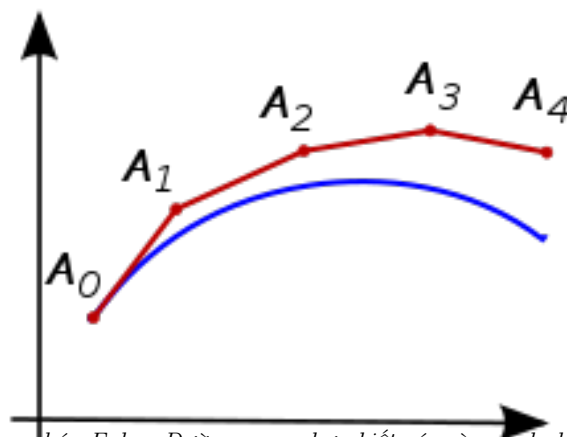
1.2 Phương pháp xấp xỉ Euler trong giải hệ phương trình vi phân.

Phương pháp Euler là một phương pháp bậc một thường được sử dụng trong việc giải các phương trình vi phân thường. Phương pháp được đặt tên theo Leonhard Euler, người đã giới thiệu phương pháp trong quyển sách Institutionum Calculi Integralis cùng tên xuất bản trong khoảng thời gian 1768 đến 1770.

Xem xét bài toán tính toán hình dạng của một đường cong chưa biết bắt đầu tại một điểm cho trước và thỏa mãn một phương trình vi phân nào đó. Ở đây, phương trình vi phân có thể được coi như là một công thức mà nhờ nó độ dốc của đường tiếp tuyến với đường cong có thể được tính toán tại điểm bất kỳ nào trên đường cong, một khi vị trí của điểm đó đã được tính toán.

Ý tưởng là trong khi đường cong ban đầu chưa biết, điểm khởi đầu của nó được biểu thị bởi A_0 là đã biết (xem hình phía trên bên phải). Thì, từ phương trình vi phân, độ dốc của đường cong tại có thể được tính toán, và vì vậy, có thể tìm được đường tiếp tuyến.

Đi một bước nhỏ dọc theo đường tiếp tuyến đến một điểm A_1 . Dọc bước nhỏ này, độ dốc không thay đổi quá nhiều, vì vậy A_1 sẽ gần với đường cong. Nếu chúng ta coi A_1 vẫn còn nằm trên đường cong, cùng một lý luận như đối với điểm A_0 có thể tính được độ dốc và tiếp tuyến của đường cong tại A_1 . Sau vài bước, một đường cong đa giác $A_0A_1A_2A_3\dots$ được tìm ra. Nói chung, đường cong này không phân kỳ quá xa khỏi đường cong chưa biết ban đầu, và sai số giữa hai đường cong có thể được làm nhỏ nếu kích thước bước là đủ nhỏ và khoảng thời gian tính toán là hữu hạn.



Hình 1: Minh họa phương pháp Euler. Đường cong chưa biết có màu xanh da trời và lời giải gần đúng của nó là đường nhiều cạnh màu đỏ.

Giả sử ta có phương trình vi phân bậc nhất

$$y'(t) = f(t, y(t)), \quad y(t_0) = y_0$$

Trên đoạn $[a, b]$ ta chia thành n đoạn bằng nhau với

$$\Delta t = \frac{b-a}{n}$$

Δt là kích thước của mỗi bước và đặt $t_n = t_0 + n\Delta t$. Xét phương trình vi phân $y'(t) = f(t, y(t))$. Phương pháp Euler thuận thu được bằng cách sử dụng sai phân xấp xỉ thuận

$$\frac{y_{n+1} - y_n}{\Delta t} \approx y'(t) = f(t, y(t)) \Rightarrow y_{n+1} = y_n + f(t, y(t))\Delta t.$$

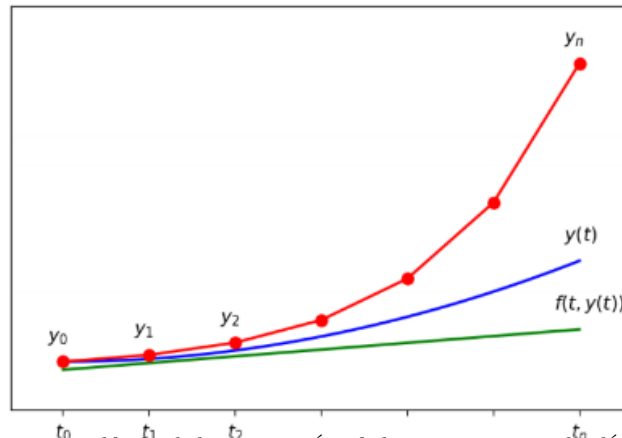
Ở dạng tổng quát, một hệ phương trình vi phân bậc một được viết dưới dạng

$$y_1 = y_0 + f(t_0, y(t_0))\Delta t.$$

$$y_2 = y_1 + f(t_1, y(t_1))\Delta t.$$

...

$$y_n = y_{n-1} + f(t_{n-1}, y(t_{n-1}))\Delta t.$$



Hình 2: Các điểm tròn màu đỏ chính là giá trị xấp xỉ đường cong y có độ dốc $y'(t) = f(t, y)$ nghĩa là $y' = f(t, y)$. Sai khác giữa y_n và y_{n-1} đúng bằng $f(t_{n-1}, y_{n-1})(t_n - t_{n-1})$

trong đó y_i là các hàm số thực phụ thuộc vào biến $t \geq 0$ và f_i là các hàm số thực phi tuyến phụ thuộc vào biến $t \geq 0$ và các y_i 's với mọi $i \in 1, \dots, N$. Phương pháp Euler khi đó được áp dụng cho từng y_i .

1.3 Giải thuật Metropolis-Hastings

1.3.1 Phương pháp Monte Carlo

1.3.1.a Sơ lược

Các phương pháp Monte Carlo là một lớp các thuật toán để giải quyết nhiều bài toán trên máy tính theo kiểu không tất định, thường bằng cách sử dụng các số ngẫu nhiên (thường là các số giả ngẫu nhiên), ngược lại với các thuật toán tất định

1.3.1.b Phương pháp chung

Phương pháp Monte Carlo có thể có nhiều biến thể khác nhau, nhưng thường sẽ theo các bước sau:

- Bước 1: Chọn miền xác định các biến đầu vào
- Bước 2: Chọn các biến đầu vào, thuộc miền xác định, 1 cách ngẫu nhiên dựa trên 1 hàm phân phối xác suất trên miền xác định đã chọn ở câu 1
- Bước 3: Thực hiện 1 phép tính toán tất định trên các biến đầu vào đó
- Bước 4: Tổng hợp kết quả

1.3.2 Sơ lược về xích Markov

1.3.2.a Quá trình Markov

Quá trình Markov là một quá trình ngẫu nhiên, mà dự đoán về các trạng thái trong tương lai của nó phụ thuộc hoàn toàn vào trạng thái hiện tại, và dự đoán này chính xác như là nếu ta đã biết toàn quá trình trong tương lai lẫn quá khứ, nói cách khác, trạng thái trong quá khứ và tương lai là độc lập đôi một với nhau

1.3.2.b Xích Markov

Xích Markov thường được định nghĩa là 1 quá trình Markov mà không gian trạng thái, hoặc tập index (thường sử dụng thời gian), là rời rạc

1.3.2.c Xích Markov Monte Carlo (MCMC)

MCMC là một họ gồm nhiều thuật toán thường dùng để lấy mẫu phân bố xác suất nhiều chiều dựa trên việc xây dựng xích Markov có phân bố dừng tương ứng và kỹ thuật gieo điểm ngẫu nhiên Monte Carlo. Bằng cách xây dựng 1 xích Markov với phân phối mong muốn là phân phối cân bằng, ta có thể lấy mẫu của phân phối mong muốn đó bằng cách ghi lại các trạng thái từ xích. Có nhiều phương thức để xây dựng chuỗi, một trong số đó là giải thuật Metropolis-Hastings.

1.3.3 Giải thuật Metropolis-Hastings

1.3.3.a Phương pháp

Thuật toán Metropolis-Hastings được biết đến là một phương pháp xích Markov Monte Carlo có các bước như sau:

1. Khởi tạo β_0 và γ_0 từ phân bố xác suất tiên nghiệm $\pi(\beta, \gamma)$
2. Gán $\beta := \beta_0$ và $\gamma := \gamma_0$
3. Khởi tạo β^* và γ^* ngẫu nhiên từ phân phối xác suất bất kỳ $p(\beta, \gamma)$
4. Nếu $p(\beta, \gamma)$ là đối xứng, nghĩa là $p(\beta^*, \gamma^* | \beta, \gamma) = p(\beta, \gamma | \beta^*, \gamma^*)$, gán cho r là xác suất giữ lại β^* và γ^* bằng công thức:

$$r := \min(1, \frac{\pi(\beta^*, \gamma^*)}{\pi(\beta, \gamma)}) \quad (1)$$

và đi đến bước 6

5. Nếu không đối xứng, gán

$$r := \min(1, \frac{\pi(\beta^*, \gamma^*)p(\beta, \gamma | \beta^*, \gamma^*)}{\pi(\beta, \gamma)p(\beta^*, \gamma^* | \beta, \gamma)}) \quad (2)$$

6. Khởi tạo giá trị q ngẫu nhiên từ phân phối đều liên tục $U(0, 1)$.
7. Nếu $q < r$, tạo $\beta_{i+1} := \beta^*$ và $\gamma_{i+1} := \gamma^*$ với i là chỉ số phần tử trong mẫu và đi đến bước 9.
8. Ngược lại, tạo $\beta_{i+1} := \beta$ và $\gamma_{i+1} := \gamma$ và đi đến bước 9
9. Lặp lại từ bước 2 với $\beta := \beta_i$ và $\gamma := \gamma_i$ cho đến khi đủ kích cỡ mẫu.

1.4 Một số kiến thức khác về xác suất và thống kê

1.4.1 Xác suất tiên nghiệm

Xác suất tiên nghiệm trong tiếng anh là Priori Probability

Xác suất tiên nghiệm là một phương pháp dự đoán khả năng một biến cố sẽ xuất hiện theo một cách nhất định dựa trên tỉ lệ phần trăm, chứ không phải được xác định từ dữ liệu lịch sử.

Nói cách khác, xác suất tiên nghiệm nhìn vào tính logic hơn là dữ liệu trong quá khứ để xác định xác suất một biến cố có thể xảy ra.

Đặc điểm của Xác suất tiên nghiệm Xác suất tiên nghiệm thường được sử dụng trong phương pháp tính toán xác suất.

Điều này là do người quan sát phải sử dụng logic để xác định các kết quả có thể xảy ra của một biến cố để xác định các hình thức mà các kết quả này có thể xảy ra.

Xác suất tiên nghiệm thường được sử dụng để xác định khả năng xảy ra các biến cố độc lập ở đó khả năng xảy ra một biến cố nhất định sẽ không bị ảnh hưởng bởi các biến cố đã xảy ra trước đó.

Xác suất tiên nghiệm có thể được áp dụng cho tất cả các trò chơi ngẫu nhiên như roulette, ném xúc xắc, chơi xổ số, v.v.

Hạn chế lớn nhất của phương pháp xác định xác suất này là nó chỉ có thể được áp dụng cho một tập hợp các biến cố hữu hạn vì hầu hết các biến cố đều có xác suất có điều kiện.

1.4.2 Xác suất hậu nghiệm

Xác suất hậu nghiệm trong tiếng anh là Posterior Probability. Xác suất hậu nghiệm, theo thống kê của Bayes, là xác suất được điều chỉnh hoặc cập nhật của một biến cố xảy ra sau khi xem xét thông tin mới.

Xác suất hậu nghiệm được xác định bằng cách điều chỉnh xác suất tiên nghiệm bằng định lý Bayes.

Theo thuật ngữ thống kê, xác suất hậu nghiệm là xác suất của biến cố A xảy ra do biến cố B đã xảy ra.

Xác suất tiên nghiệm theo Bayes, là xác suất của một biến cố xảy ra trước khi dữ liệu mới được thu thập.

Xác suất tiên nghiệm là dự đoán đúng nhất về kết quả xác suất dựa trên kiến thức và tính logic ở thời điểm trước khi thử nghiệm được thực hiện.

Công thức tính Xác suất hậu nghiệm Công thức tính xác suất hậu nghiệm của biến cố A xảy ra khi biến cố B đã xảy ra:

Xác suất hậu nghiệm (Posterior Probability) là gì? Công thức tính Xác suất hậu nghiệm:

$$P(A|B) = \frac{P(B|A) * P(A)}{P(B)} = \frac{P(A \cap B)}{P(B)} \quad (3)$$

Trong đó:

- A,B là các biến cố.
- P(B) và P(A) là xác suất của biến cố B xảy ra và biến cố A xảy ra.
- P(B|A) là xác suất biến cố B xảy ra khi biến cố A đúng.
- P(A|B) là kết quả của phương trình, là xác suất biến cố A xảy ra khi biến cố B đã xảy ra.

Chức năng của Xác suất hậu nghiệm Định lý Bayes được ứng dụng trong nhiều lĩnh vực khác nhau như y học, tài chính và kinh tế.

Trong tài chính, công thức tính xác suất hậu nghiệm của định lý Bayes có thể được sử dụng để cập nhật các niềm tin trước đó một khi có thông tin mới.

Xác suất tiên nghiệm đại diện cho những niềm tin ban đầu, trước khi các thông tin hay bằng chứng mới được đưa ra và xác suất hậu nghiệm là xác suất đã được cập nhật theo thông tin mới này.

Phân phối xác suất hậu nghiệm phản ánh tốt hơn về quá trình tạo dữ liệu so với xác suất tiên nghiệm vì nó tính xác suất sau khi biến cố đã xảy ra, vì vậy, bao hàm nhiều thông tin hơn.

1.4.3 Xác suất Bayes

1.4.3.a Lý thuyết

Định lý Bayes cho phép tính xác suất xảy ra của một sự kiện ngẫu nhiên A khi biết sự kiện liên quan B đã xảy ra. Xác suất này được ký hiệu là $P(A|B)$, và đọc là "xác suất của A nếu có B". Đại lượng này được gọi xác suất có điều kiện hay xác suất hậu nghiệm vì nó được rút ra từ giá trị được cho của B hoặc phụ thuộc vào giá trị đó. Theo định lý Bayes, xác suất xảy ra A khi biết B sẽ phụ thuộc vào 3 yếu tố: - Xác suất xảy ra A của riêng nó, không quan tâm đến B. Ký hiệu là $P(A)$ và đọc là xác suất của A. Đây được gọi là xác suất biên duyên hay xác suất tiên nghiệm, nó là "tiên nghiệm" theo nghĩa rằng nó không quan tâm đến bất kỳ thông tin nào về B.

- Xác suất xảy ra B của riêng nó, không quan tâm đến A. Ký hiệu là $P(B)$ và đọc là "xác suất của B". Đại lượng này còn gọi là hằng số chuẩn hóa (normalising constant), vì nó luôn giống nhau, không phụ thuộc vào sự kiện A đang muốn biết.

- Xác suất xảy ra B khi biết A xảy ra. Ký hiệu là $P(B|A)$ và đọc là "xác suất của B nếu có A". Đại lượng này gọi là khả năng (likelihood) xảy ra B khi biết A đã xảy ra. Chú ý không nhầm lẫn giữa khả năng xảy ra B khi biết A và xác suất xảy ra A khi biết B.

Khi biết ba đại lượng này, xác suất của A khi biết B cho bởi công thức:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} = \frac{P(A \cap B)}{P(B)} \quad (4)$$

1.4.3.b Một số bài toán về xác suất tiên nghiệm

- Bài toán 1: một ví dụ điển hình về xác suất tiên nghiệm là khi tung một đồng xu.

Một đồng xu đồng chất có hai mặt là mặt sấp và mặt ngửa, mỗi lần tung đồng xu khả năng xuất hiện mặt sấp và mặt ngửa là bằng nhau bất kể kết quả ném trước đó như thế nào.

Xác suất tiên nghiệm xuất hiện mặt ngửa khi tung đồng xu là 50%, tương tự, xác suất tiên nghiệm xuất hiện mặt sấp khi tung đồng xu cũng là 50%.

- Bài toán 2: ví dụ khác phổ biến hơn là khi giá của một cổ phiếu thay đổi.

- Giá của nó có thể tăng, giảm hoặc giữ nguyên. Do đó, theo xác suất tiên nghiệm, chúng ta có khả năng giá tăng giảm giữ nguyên là $\frac{1}{3}$ hay 33%. Khả năng xảy ra một trong các kết quả được xác định này đều bằng nhau.

- Bài toán 3: giả sử cổ phiếu của công ty XYZ đóng cửa ở mức 10 đô ngày hôm qua.

Dựa trên vài ngày cuối cùng trong lịch sử giao dịch, cổ phiếu có vẻ như đang có xu hướng tăng, vì vậy các nhà đầu tư cho xác suất khả năng giá cổ phiếu tăng là 66% vào ngày mai.

Tuy nhiên, với xác suất tiên nghiệm, giá cổ phiếu của công ty XYZ chỉ có thể xảy ra một trong ba kịch bản sau: đi lên, đi xuống hoặc giữ nguyên. Vì vậy, chỉ có 33% cơ hội cổ phiếu sẽ tăng giá vào ngày mai.

Nhược điểm của xác suất tiên nghiệm là chỉ dựa trên lí trí do các kết quả được xác định bởi người quan sát chứ không phải từ hiểu biết, nhận thức hay dữ liệu lịch sử.

1.4.3.c Một số bài toán về xác suất hậu nghiệm

Một ví dụ đơn giản để hình dung xác suất hậu nghiệm là giả sử có ba mẫu đất có nhãn A, B và C. Trong đó, một mẫu đất trong ba mẫu A,B,C có một trữ lượng dầu bên dưới bề mặt của nó, trong khi hai mẫu còn lại thì không.

Xác suất tiên nghiệm của khả năng trữ lượng dầu nằm trong mẫu đất C là $\frac{1}{3}$ hay 33%.

Sau đó thử nghiệm khoan được tiến hành trên mẫu đất B và kết quả chỉ ra rằng không có dầu ở mẫu đất này.

Do mẫu đất B đã bị loại bỏ, xác suất hậu nghiệm của khả năng mẫu đất C chứa dầu trở thành $\frac{1}{2}$ hay 50%.

2 Cách xây dựng mô hình SIR

2.1 Giới thiệu chung về mô hình.

Mô hình SIR (Susceptible - Infectious - Recovered) là một trong các mô hình cách ly cơ bản được sử dụng nhiều nhất trong quá khứ và hiện tại để mô tả dịch bệnh. Mô hình chia dân số trong một khu vực thành ba loại - Có khả năng bị mắc bệnh - Bị mắc bệnh - Đã phục hồi cho một nhóm người được cách ly với giả thiết rằng sẽ miễn dịch với bệnh nếu đã phục hồi.

2.2 Trường hợp liên tục

Trường hợp liên tục của mô hình SIR là một hệ động lực gồm 3 phương trình vi phân sau :

$$\frac{dS}{dt} = -\frac{\beta}{N}SI \quad (5)$$

$$\frac{dI}{dt} = \frac{\beta}{N}SI - \gamma I \quad (6)$$

$$\frac{dR}{dt} = \gamma I \quad (7)$$

trong đó tại mỗi thời điểm $t \geq t_0 \geq 0$ với t_0 là thời điểm đầu ghi nhận,

- $S(t)$: Số người có nguy cơ mắc bệnh;
- $I(t)$: Số người bị nhiễm bệnh;
- $R(t)$: Số người hồi phục sau bệnh;
- $\beta(t)$: Tỷ lệ tiếp xúc của người trong nhóm $S(t)$ với người trong nhóm $I(t)$, được tính dựa trên các yếu tố thực tế như bán kính đi lại, mức độ tiếp xúc giữa người mắc bệnh với người có khả năng mắc bệnh, tính chất của virus,...;
- $\gamma(t)$: Tỷ lệ hồi phục khi mắc bệnh, được tính bằng cách lấy 1 chia cho khoảng thời gian trung bình mà một người phục hồi;
- $N(t)$: Tổng số người trong cộng đồng cách ly được tính bằng :

$$N(t) := S(t) + I(t) + R(t) \quad (8)$$

Hệ phương trình vi phân trên có thể được hiểu như sau :

- Phương trình (5) thể hiện sự suy giảm số người có nguy cơ mắc bệnh tại thời điểm $t \geq t_0$. Sự suy giảm được tính theo xác suất lây bệnh khi có tiếp xúc giữa nhóm $S(t)$ và nhóm $I(t)$;
- Phương trình (6) thể hiện độ biến thiên số người mắc bệnh tại thời điểm $t \geq t_0$. Sự biến thiên này được tính bằng cách lấy số người ở nhóm $S(t)$ đã bị lây nhiễm sau khi tiếp xúc với người bệnh nhóm $I(t)$ và trừ đi số người ở nhóm $I(t)$ đã phục hồi với tỷ lệ $\gamma I(t)$;
- Phương trình (7) thể hiện số người đã hồi phục từ nhóm $I(t)$ theo tỷ lệ hồi phục là γ .

2.3 Trường hợp rời rạc.

Trường hợp rời rạc của mô hình SIR là hệ phương trình sai phân sau:

$$R(n+1) = R(n) + \gamma(\Delta t)I(n) \quad (9)$$

$$I(n+1) = I(n) - \gamma(\Delta t)I(n) + \beta(\Delta t)I(n)S(n) \quad (10)$$

$$S(n+1) = S(n) - \beta(\Delta t)S(n)I(n) \quad (11)$$

Với $S_0 > 0, I_0 > 0, R_0 > 0$, ta có thỏa mãn tính chất sau: $S_0 + I_0 + R_0 = N$

- $S(n)$: Số người có nguy cơ mắc bệnh;
- $I(n)$: Số người bị nhiễm bệnh;
- $R(n)$: Số người hồi phục sau bệnh;
- $\beta(n)$: Tỷ lệ tiếp xúc của người trong nhóm $S(t)$ với người trong nhóm $I(n)$, được tính dựa trên các yếu tố thực tế như bán kính đi lại, mức độ tiếp xúc giữa người mắc bệnh với người có khả năng mắc bệnh, tính chất của virus,...;
- $\gamma(n)$: Tỷ lệ hồi phục khi mắc bệnh, được tính bằng cách lấy 1 chia cho khoảng thời gian trung bình mà một người phục hồi;
- $N(n)$: Tổng số người trong cộng đồng cách ly;

Giả sử rằng β, γ là các giá trị không đổi và có thể xác định từ các dữ liệu ban đầu. Các cá nhân phục hồi từ căn bệnh sẽ trở miễn dịch vĩnh viễn.

Trong hệ rời rạc chỉ xét số nguyên dương cho $n=1,2,\dots$.

Ta có hệ các phương trình thể hiện tốc độ thay đổi S, I, R :

$$R' = \gamma(\Delta t)I \quad (12)$$

$$I' = -\gamma(\Delta t)I + \beta(\Delta t)IS \quad (13)$$

$$S' = -\beta(\Delta t)SI \quad (14)$$

Giới hạn của S thì phụ thuộc vào các điều kiện ban đầu

Nếu $S_0 \leq \frac{\gamma}{\beta}$, thì $I_1 \leq I_0$ và S_n giảm dần $I_{n+1} \leq I_n$ hay $I' < 0$; Không còn dịch bệnh.

Ngược lại, nếu $S_0 > \frac{\gamma}{\beta}$, thì $I_1 > I_0$ hay $I' > 0$ nhóm người bị nhiễm bệnh đang gia tăng.

2.4 Những vấn đề liên quan

2.4.1 Phương pháp xấp xỉ Euler trong giải hệ SIR

Phương pháp Euler là một phương pháp bậc một thường được sử dụng trong việc giải các phương trình vi phân thường. Phương pháp được đặt tên theo Leonhard Euler, người đã giới thiệu phương pháp trong quyển sách Institutionum Calculi Integralis cùng tên xuất bản trong khoảng thời gian 1768 đến 1770.

Giả sử ta có phương trình vi phân bậc nhất :

$$y' = f(t, y(t)) \quad (15)$$

Khi đó, ý tưởng của phương pháp Euler là xấp xỉ nghiệm y bằng dãy y_n sao cho

$$y_{n+1} := y_n + f(t_n, y_n)\Delta t \quad (16)$$

Với Δt là bước xấp xỉ đủ nhỏ và $f(t, y(t))$ là độ dốc của đường cong y tính tại thời điểm t .

Ở dạng tổng quát, một hệ phương trình vi phân bậc một được viết dưới dạng

$$y'_1 = f_1(t, y_1, \dots, y_N) \quad (17)$$

$$\vdots \quad (18)$$

$$y'_N = f_N(t, y_1, \dots, y_N). \quad (19)$$

trong đó y_i là các hàm số thực phụ thuộc vào biến $t \geq 0$ và f_i là các hàm số thực phi tuyến phụ thuộc vào biến $t \geq 0$ và các y_i với mọi $i \in \{1, \dots, N\}$. Phương pháp Euler khi đó được áp dụng cho từng y_i .

2.4.2 Ước lượng hệ số β và γ .

Việc ước lượng các hệ số β và γ phụ thuộc vào dữ liệu về COVID-19 đã được công bố. Cụ thể là số ca mắc bệnh và phục hồi tích lũy theo thời gian. Ở đây, ta sẽ sử dụng phương pháp suy luận Bayes. Gọi

- X : biến ngẫu nhiên quan sát số ca mắc bệnh và số ca hồi phục tại từng thời điểm $t \geq t_0$
- $\pi(\beta, \gamma|X)$: phân bố xác suất hậu nghiệm của β và γ khi có dữ liệu quan sát;
- $\pi(X|\beta, \gamma)$: phân bố xác suất của số ca mắc bệnh và số ca phục hồi khi β và γ
- $\pi(\beta, \gamma)$: phân bố xác suất tiên nghiệm khi chưa có dữ liệu ghi nhận về số ca mắc bệnh và số ca phục hồi.

Định lý Bayes được phát biểu như sau

$$\pi(\beta, \gamma) \propto \pi(X|\beta, \gamma)\pi(\beta, \gamma) \quad (20)$$

Nghĩa là phân bố xác suất hậu nghiệm của β và γ có thể được tính bằng cách lấy phân bố xác suất của số ca mắc bệnh và số ca phục hồi khi β và γ cho trước nhân với phân bố xác suất tiên nghiệm của β và γ .

Ước lượng này có vai trò rất quan trọng và được thể hiện trên hệ số

$$R_0 := \frac{\beta}{\gamma}. \quad (21)$$

Khi hệ số $R_0 < 1$ thì không có đợt bùng phát dịch bệnh xảy ra vì tỷ lệ tiếp xúc người mắc bệnh β nhỏ hơn tốc độ phục hồi. Khi hệ số $R_0 > 1$ thì các đợt bùng phát dịch bệnh sẽ xảy ra trong tương lai vì tỷ lệ tiếp xúc với người mắc bệnh cao hơn tốc độ phục hồi sau bệnh. Đặc biệt giá trị trung bình của hệ số này đúng bằng

$$E(R_0) = \int \pi(\beta, \gamma | X) R_0(\beta, \gamma) d(\beta, \gamma) \quad (22)$$

Trong đó X là dữ liệu về số ca mắc bệnh và phục hồi quan sát được. Giá trị trung bình này có thể ước lượng được vì phân bố xác suất $\pi(\beta, \gamma | X)$ có thể tính được nhờ vào công thức Bayes (20)

Tuy nhiên, tích phân (22) không thể được tính toán một cách trực tiếp. Thay vào đó chúng ta sẽ sử dụng công thức xấp xỉ

$$E(R_0) = \int \pi(\beta, \gamma | X) R_0(\beta, \gamma) \propto \int \pi(X | \beta, \gamma) R_0 d(\beta, \gamma) \approx \sum_{i=1}^m \pi(X | \beta_i, \gamma_i) \frac{\beta_i}{\gamma_i}, \quad (23)$$

Trong đó, (β_i, γ_i) được lấy ra dựa trên phân bố xác suất tiên nghiệm $\pi(\beta, \gamma)$ và m là kích cỡ mẫu.

Nghĩa là, chúng ta có thể lấy một mẫu kích cỡ m đủ lớn gồm các giá trị của biến ngẫu nhiên (β, γ) dựa trên phân phối xác suất tiên nghiệm $\pi(\beta, \gamma)$ và tính tổng các giá trị $\pi(X | \beta, \gamma) \frac{\beta}{\gamma}$ trên mẫu này.

3 Bài toán 2. Áp dụng thuật toán Euler để tìm nghiệm của hệ SIR

3.1 Giới thiệu

Trong bài toán này, ta sẽ tìm nghiệm của hệ SIR sử dụng thuật toán Euler bằng chương trình viết trong Python và biểu diễn nghiệm thu được dùng đồ thị.

3.2 Tham số đầu vào

t : Khoảng thời gian ta đang xét (ngày) Chọn bước nhảy Δt là 1 ngày.

β : Hệ số tiếp xúc

γ : Hệ số phục hồi

$I(t_0)$: Số ca mắc bệnh tại thời điểm lần đầu ghi nhận được ca nhiễm bệnh $S(t_0)$: Số người chưa mắc bệnh có khả năng nhiễm tại thời điểm lần đầu ghi nhận được ca nhiễm bệnh $R(t_0)$: Số người hồi phục tại thời điểm lần đầu ghi nhận được ca nhiễm bệnh

3.3 Kết quả đầu ra

Kết quả thu được từ chương trình sẽ là 1 mảng, biểu diễn bằng đồ thị, với các kết quả sau:

$I(t)$: Số ca mắc bệnh tại thời điểm $t \geq t_0$

$S(t)$: Số người chưa mắc bệnh có khả năng nhiễm tại thời điểm $t \geq t_0$

$R(t)$: Số người hồi phục tại thời điểm $t \geq t_0$

3.4 Chương trình

```
import numpy as np
import scipy.integrate
import matplotlib.pyplot as pyplot
import random

#The number of days in the model (Assuming that t0 equals zero)
PERIOD = 20

#Array (S, I, R) of the results
SIR_out = []

#Arrays of S, I, R, respectively
S = []
I = []
R = []

#Initial values
S0 = 1000.0
I0 = 7.0
R0 = 0.0
beta = 0.0020
gamma = 0.50

#The number of people in the model
N = S0 + I0 + R0

#Return the slopes of the variables at the of the day
#Using formulas from SIR model to calculate the scope
#y: The values of S, I, R of the day before
def SIR_atom(y, beta, gamma):
    S_, I_, R_ = y

    dS_dt = -(beta)*I_*S_
    dR_dt = gamma*I_

    #Limits the number of S and R people so that 0<S, I, R<N at all times
    if(y[0] + dS_dt < 0):
        dS_dt = -y[0]
    if(y[2] + dR_dt > N):
```

```
dR_dt = N-y[2]

dI_dt = - (dS_dt + dR_dt)

return ([dS_dt, dI_dt, dR_dt])

#Get results and store them in SIR_out
# y is the array of S, I, R
def SIR_model(y, beta, gamma):

    #Instantiate values
    S.append(y[0])
    I.append(y[1])
    R.append(y[2])

    #Run over a period of (PERIOD) days
    for i in range(PERIOD):
        dS_dt, dI_dt, dR_dt = SIR_atom(y, beta, gamma)
        y[0] += dS_dt
        y[1] += dI_dt
        y[2] += dR_dt

        # Store values
        S.append(y[0])
        I.append(y[1])
        R.append(y[2])

    # Store values
    SIR_out.append(S)
    SIR_out.append(I)
    SIR_out.append(R)

if __name__ == "__main__":

    SIR_model([S0, I0, R0], beta, gamma)

    #Prints out all the results in (S I R) form
    for col in range(PERIOD):
        print(SIR_out[0][col], SIR_out[1][col], SIR_out[2][col], sep = " ")

    t = range(PERIOD + 1)

    #Arrays to store values to draw plot
    S_t = []
    I_t = []
    R_t = []

    #Store values to draw plot
    for e in SIR_out[0]:
```



```
S_t.append(e)

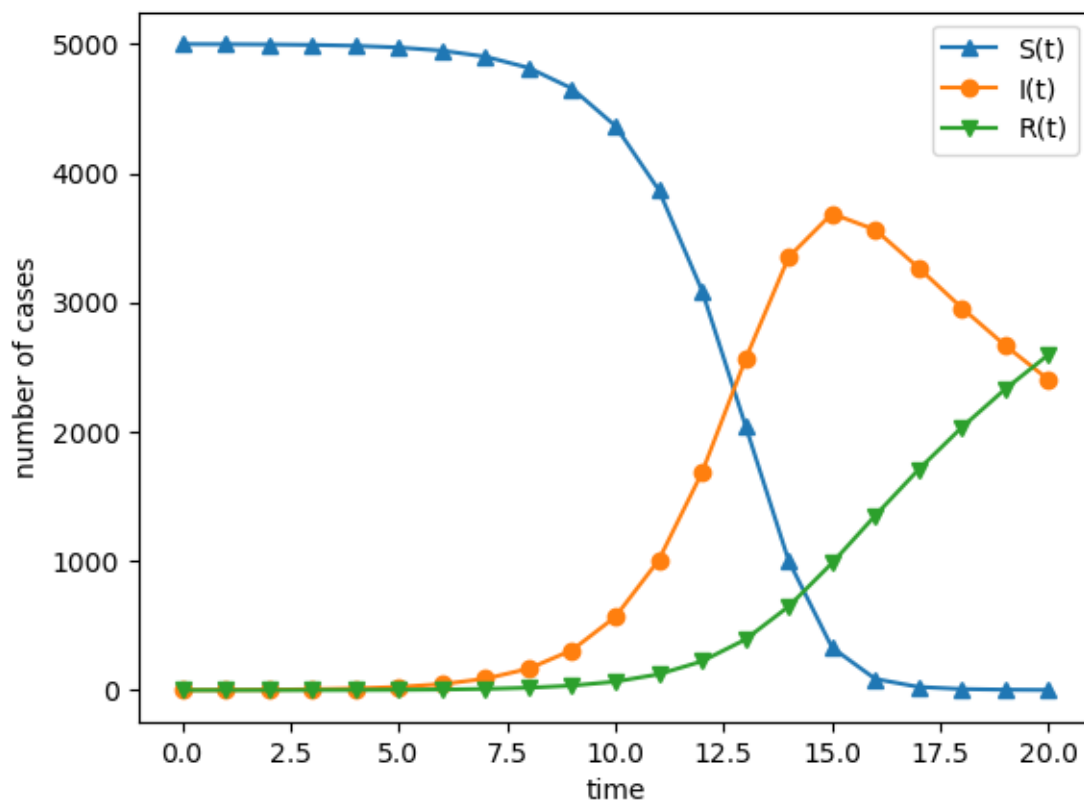
for e in SIR_out[1]:
    I_t.append(e)

for e in SIR_out[2]:
    R_t.append(e)

#Draw the plot that shows the result
pyplot.plot(t, S_t, label = "S(t)", marker = '^')
pyplot.plot(t, I_t, label = "I(t)", marker = 'o')
pyplot.plot(t, R_t, label = "R(t)", marker = 'v')
pyplot.xlabel("time")
pyplot.ylabel("number of cases")
pyplot.legend()
pyplot.show()
```

3.5 Kết quả và phân tích

3.5.1 Trường hợp 1



$$S_0 = 5000;$$

$$I_0 = 1;$$

$$R_0 = 0;$$

$$\beta = 0.0002;$$

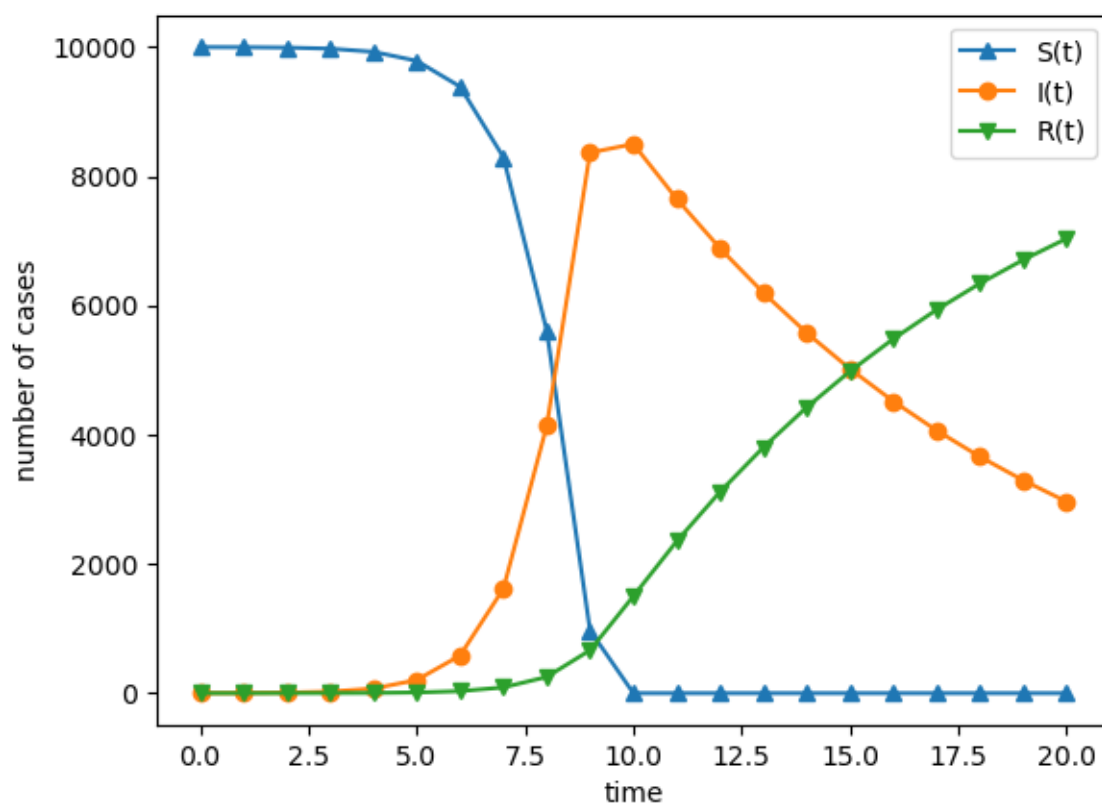
$$\gamma = 0.1;$$

Ta sẽ lấy trường hợp 1 làm tiêu chuẩn để so sánh và phân tích với các trường hợp khác

Tốc độ gia tăng số người nhiễm trong trường hợp này tăng nhanh vào từ ngày đầu đến khoảng ngày 14, với đồ thị bên trái cực đại giống với đồ thị hàm số mũ, đạt cực đại vào ngày 15 và sau đó giảm dần do số người khỏe mạnh giảm xuống 0.

Số người khỏe mạnh giảm dần và vào khoảng ngày 18, số người khỏe mạnh đã giảm xuống 0.

3.5.2 Trường hợp 2



$S_0 = 10000$ - Tăng 2 lần so với trường hợp 1;

$I_0 = 1$;

$R_0 = 0$;

$\beta = 0.0002$;

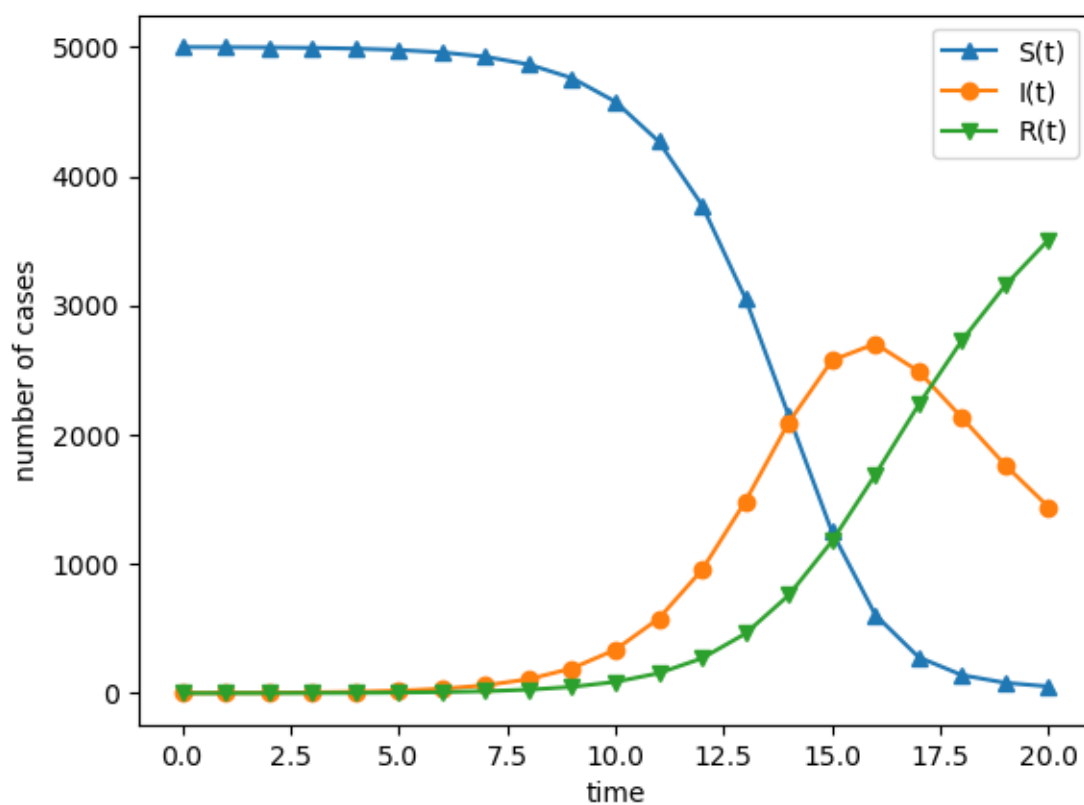
$\gamma = 0.1$;

Ta có thể thấy tốc độ gia tăng của số người bị nhiễm giảm đi rất nhiều so với trường hợp 1, đạt cực đại vào ngày 10, so với trường hợp 1 là ngày 15 và số lượng người khỏe mạnh cũng giảm nhanh hơn nhiều.

Số người hồi phục trong trường hợp này cũng tăng nhanh hơn so với trường hợp 1, nhưng đây cũng là kết quả từ việc số người nhiễm tăng nhanh hơn (Do số người hồi phục tăng nhanh hay chậm phụ thuộc vào γ và số người bị nhiễm)

→ Ta có thể thấy tốc độ gia tăng số người nhiễm bệnh phụ thuộc rất nhiều vào số người dân, việc này trong thực tế có thể hiểu là những biện pháp cách ly, hạn chế đi lại... ở trường hợp 1 để giảm số người dân trong từng không gian kín, đã không áp dụng ở trường hợp 2 làm tăng đáng kể số người nhiễm bệnh và tốc độ lây lan của dịch.

3.5.3 Trường hợp 3



$$S_0 = 5000;$$

$$I_0 = 1;$$

$$R_0 = 0;$$

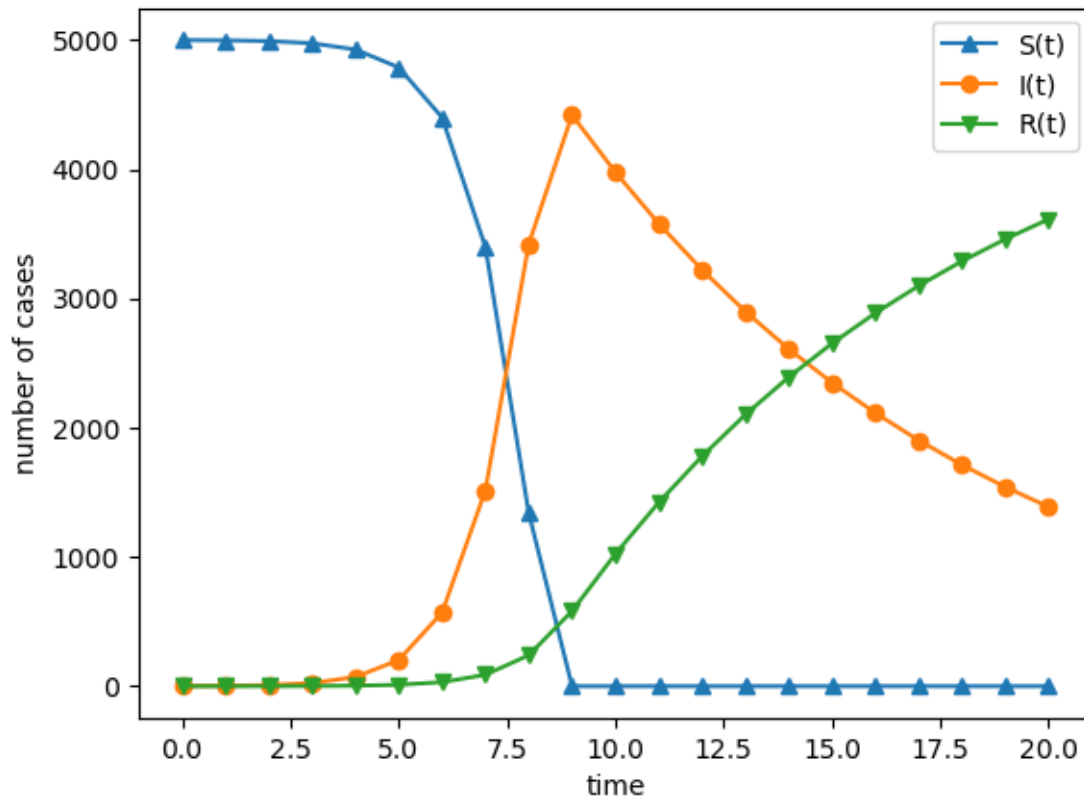
$$\beta = 0.0002;$$

$$\gamma = 0.2 - \text{Tăng 2 lần so với trường hợp 1};$$

Với S_0 bằng trường hợp 1, thì ta có thể thấy 2 trường hợp 1 và 3 có số người bị nhiễm và số người khỏe mạnh có đường biểu diễn giống nhau và các phân tích ở trường hợp 1 có thể áp dụng vào trường hợp 3.

Sự khác biệt ở trường hợp 1 và trường hợp 3 là ở số người hồi phục, do γ tăng nên tốc độ của số người hồi phục tăng cũng cao hơn so với trường hợp 1.

3.5.4 Trường hợp 4



$$S_0 = 5000;$$

$$I_0 = 1;$$

$$R_0 = 0;$$

$$\beta = 0.0004 - \text{Tăng 2 lần so với trường hợp 1;}$$

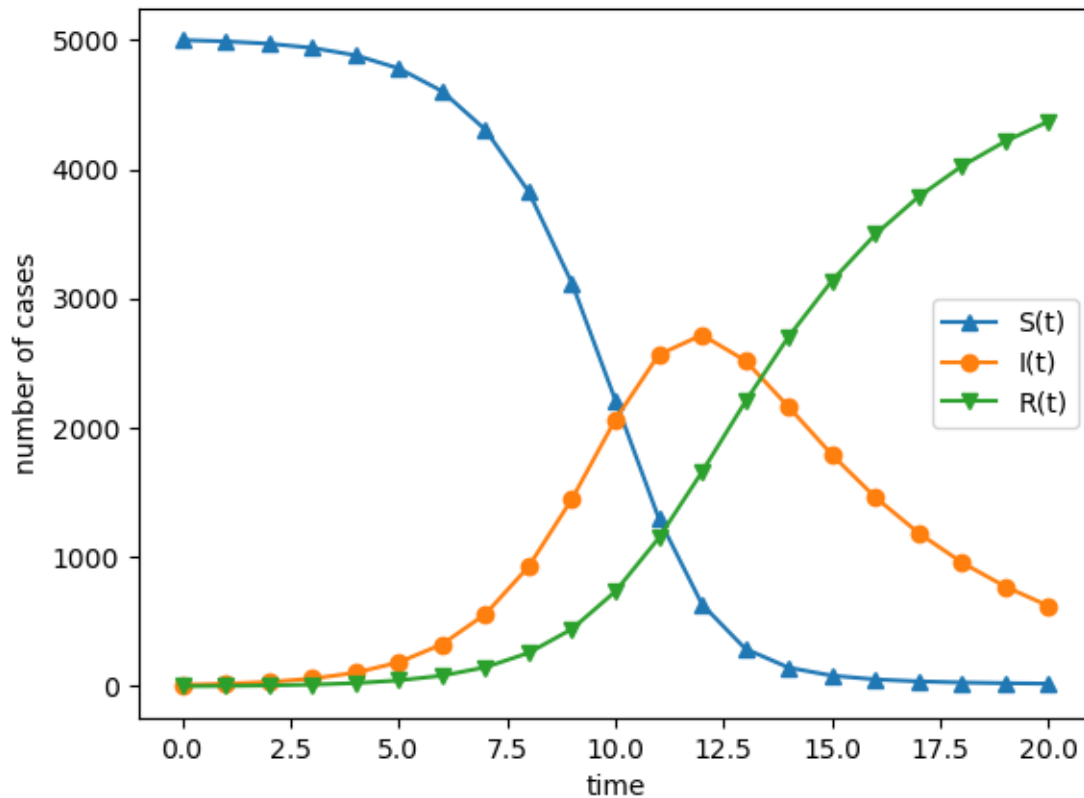
$$\gamma = 0.1;$$

Số người bị nhiễm tăng nhanh hơn rất nhiều so với trường hợp 1, và tăng nhanh hơn cả trường hợp 2, đạt cực đại vào khoảng ngày 9.

Số người hồi phục trong trường hợp này cũng tăng nhanh hơn so với trường hợp 1, nhưng đây cũng là kết quả từ việc số người nhiễm tăng nhanh hơn (Do số người hồi phục tăng nhanh hay chậm phụ thuộc vào γ và số người bị nhiễm).

Số người khỏe mạnh giảm rất nhanh, và xuống 0 vào ngày 9 \rightarrow Trong thực tế, β tăng 2 lần có thể được hiểu là các cá thể trong mô hình đã không vệ sinh tốt, đeo khẩu trang nơi công cộng, ... như ở trường hợp 1, và điều này dẫn đến số người nhiễm bệnh tăng rất nhanh.

3.5.5 Trường hợp 5



$$S_0 = 5000;$$

$$I_0 = 10; \text{ - Tăng 10 lần so với trường hợp 1}$$

$$R_0 = 0;$$

$$\alpha = 0.0002;$$

$$\beta = 0.2;$$

Số người bị nhiễm tăng và số người khỏe mạnh giảm nhanh hơn so với trường hợp 1 nhưng không nhanh bằng trường hợp 4, mặc dù số người bị nhiễm ban đầu tăng gấp 10 lần

3.6 Nhận xét

Từ những số liệu thu được từ các trường hợp trong 3.5, ta rút ra được kết luận như sau về mô hình SIR:

Số người khỏe mạnh và số người bị nhiễm có liên quan chặt chẽ đến nhau, với tốc độ tăng của số người bị nhiễm sẽ làm tăng tốc độ giảm của số người khỏe mạnh, và tốc độ gia tăng của số người bị nhiễm tỷ lệ thuận với số người khỏe mạnh.

Số người bị nhiễm tăng nhanh khi trong cộng đồng có nhiều người khỏe mạnh, β cao, trong đó, số người bị nhiễm tăng nhanh hay chậm phụ thuộc nhiều vào β .

4 Bài toán 3. Lấy mẫu sử dụng thuật toán Metropolis-Hastings

Nhóm chọn ngôn ngữ thống kê R để thực hiện bài toán.

Mô tả cách thiết lập và chạy Metropolis-Hastings dựa trên Markov chain Monte Carlo (MCMC) trong phạm vi mô hình SIR.
Ví dụ code sử dụng dữ liệu Eyam.

```
library(MultiBD)

data(Eyam)

print(Eyam)

printsamle_size <- 177
```

Tính log của phân bố xác suất số ca mắc bệnh và số người có khả năng mắc bệnh với param cho trước. Ma trận xác suất được trả về bởi `ddb_prob()` tương ứng với giá trị S từ a tới a_0 và I từ 0 tới B .

a là cận dưới, B là cận trên.

`ddb_prob()` Function là Xác suất chuyển tiếp của một quá trình chết/ sinh - chết..

`drates1` Tỷ lệ chết đi của S , bằng 0 vì S chỉ chuyển số lượng sang cho I .

`brates2` Mô hình không sinh thêm phần tử.

`drates2` Tỷ lệ chết của I là có và bằng $\alpha \cdot b$.

`trans12` Ở đây là mô hình chuyển từ trạng thái S sang trạng thái I

```
loglik_sir <- function(param, data) {

  alpha <- exp(param[1]) # Rates must be non-negative
  beta <- exp(param[2])
  # Set-up SIR model
  drates1 <- function(a, b) { 0 }
  brates2 <- function(a, b) { 0 }
  drates2 <- function(a, b) { alpha * b }
  trans12 <- function(a, b) { beta * a * b }
  sum(sapply(1:(nrow(data) - 1), # Sum across all time steps k
    function(k) {
      log( # sums the logs of logs that change from T to T + 1 state
        dbd_prob( # Compute the transition probability matrix
          t = data$time[k + 1] - data$time[k], # Time increment
          a0 = data$S[k], b0 = data$I[k], # From: S(t_k), I(t_k)
          drates1, brates2, drates2, trans12,
          a = data$S[k + 1], B = data$S[k] + data$I[k] - data$S[k + 1],
          computeMode = 4, nblocks = 80 # Compute using 4 threads
        )[1, data$I[k + 1] + 1] # To: S(t_(k+1)), I(t_(k+1))
      )
    }
  )
}
```



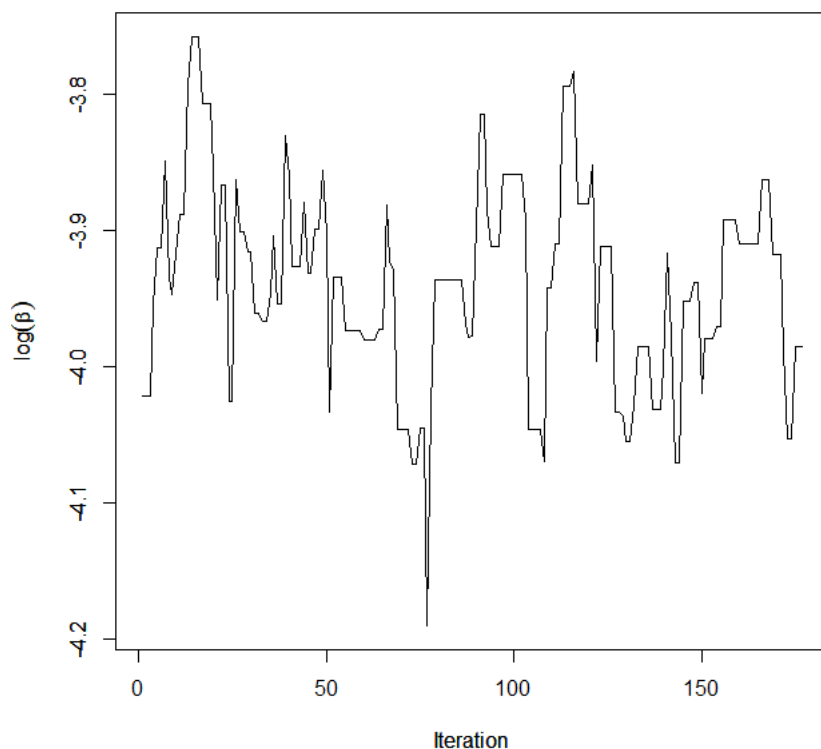
```
    }))  
}
```

Phân bố xác suất tiên nghiệm của số ca mắc bệnh và số ca phục hồi dựa trên phân bố chuẩn.

```
logprior <- function(param) {  
  log_alpha <- param[1]  
  log_beta <- param[2]  
  dnorm(log_alpha, mean = 0, sd = 100, log = TRUE) +  
    dnorm(log_beta, mean = 0, sd = 100, log = TRUE)  
}  
  
library(MCMCpack)  
  
alpha0 <- 3.39  
beta0 <- 0.0212  
  
post_sample <- MCMCmetrop1R(fun = function(param) { loglik_sir(param, Eyam) +  
  logprior(param) },  
  theta.init = log(c(alpha0, beta0)),  
  mcmc = 177, burnin = 50)  
# Sampling of beta, gamm
```

Vẽ đồ thị minh họa:

```
plot(as.vector(post_sample[,1]), type = "l", xlab = "Iteration", ylab =  
  expression(log(alpha)))  
  
plot(as.vector(post_sample[,2]), type = "l", xlab = "Iteration", ylab =  
  expression(log(beta)))  
  
R_0 <- 0  
for(i in 1:8){  
  R_0 <- R_0 + exp(loglik_sir(c(post_sample[i,1], post_sample[i,2]),  
    Eyam))*(exp(post_sample[i,1])/exp(post_sample[i,2]))  
}  
  
print(R_0)
```



Kết quả tính được với ví dụ trên là : The Metropolis acceptance rate was 0.51542

5 Bài toán 4. Tìm hệ số R_0 của khu vực tự chọn

...

6 Nhật ký công việc - Log Assignment

Tuần 1:

Thời gian	Địa điểm	Nội dung công việc	Kết quả
Thứ 5(02/07/2020)	Tại thư viện H1 trường ĐH Bách khoa tpHCM	Tìm hiểu đề tài bài tập lớn	Hiểu được tổng quan yêu cầu bài tập lớn
Thứ 7(04/07/2020)	Tại thư viện H1 trường ĐH Bách khoa tpHCM	Tìm hiểu ngôn ngữ R, ngôn ngữ Python, mô hình SIR	Hiểu được tổng quan và cách làm việc cơ bản với các ngôn ngữ



Tuần 2:

Thời gian	Địa điểm	Nội dung công việc	Kết quả
Thứ 4(08/07/2020)	Tại thư viện H1 trường ĐH Bách khoa tpHCM	Tạo git làm việc nhóm, phân chia công việc	Hiểu được các bước tiến hành của bài tập lớn
Thứ 7(11/07/2020)	Tại thư viện H1 trường ĐH Bách khoa tpHCM	Trao đổi các phần của mô hình , thảo luận tìm giải pháp	Hỗ trợ nhau thực hiện các phần việc đã phân công

Tuần 3:

Thời gian	Địa điểm	Nội dung công việc	Kết quả
Thứ 4(15/07/2020)	Tại thư viện H1 trường ĐH Bách khoa tpHCM	Kiểm tra tiến độ công việc	Các thành viên tiến triển công việc đúng yêu cầu
Thứ 7(18/07/2020)	Tại thư viện H1 trường ĐH Bách khoa tpHCM	Kiểm tra tiến độ công việc	Hoàn thành cơ sở lý thuyết, bài 1,2 và 3. Bài 4 gặp khó khăn trong việc xử lý số liệu
Chủ Nhật(19/07/2020)	Tại thư viện H1 trường ĐH Bách khoa tpHCM	Thảo luận, tìm kiếm giải pháp cho bài 4	Giải quyết được bài 4, tuy chưa là cách tốt nhất

Tuần 4:

Thời gian	Địa điểm	Nội dung công việc	Kết quả
Thứ 3(21/07/2020)	Tại thư viện H1 trường ĐH Bách khoa tpHCM	Tổng kết bài tập lớn -viết báo cáo	Hoàn thành 4/5 bài toán bài tập lớn

Phân chia công việc:



Họ tên	Phân công công việc	Đánh giá
Lê Hồng Phong	Xây dựng mô hình SIR + Bài 3	95%
Tăng Minh Nhật	Xây dựng mô hình SIR + Bài 4	100%
Đặng Thành Ngân	Tổng hợp lý thuyết về xác suất thống kê + Bài 4	95%
Đỗ Lê Thiên Ân	Lý thuyết thuật toán Metropolis- Hastings + Bài 2	90%
Huỳnh Thiên Trình	Lý thuyết phương pháp xấp xỉ Euler + Bài 3	95%

7 Kết luận

Tài liệu

- [1] Atkinson, Kendall A. (1989), *An Introduction to Numerical Analysis* (ấn bản 2), New York: John Wiley & Sons.
- [2] Butcher, John C. (2003), *Numerical Methods for Ordinary Differential Equations*, New York: John Wiley & Sons.
- [3] Tính toán khoa học - Chương 6: Bài toán giá trị ban đầu với phương trình vi phân thông thường by Chien Dang on www.slideshare.net.