

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 (CO2011)

Mô hình SIR trong dự báo COVID-19

GVHD: Nguyễn Tiến Thịnh
Nguyễn An Khương

SV thực hiện: Phạm Quang Bình – 1811578 -LO1
Cao Ngọc Bảo – 1811492 -LO4
Võ Quốc Bảo – 1811553 -LO1
Nông Quốc Dũng – 1811794 -LO1

Tp. Hồ Chí Minh, Tháng 07/2020

Mục lục

1	Mô hình SIR	2
1.1	Trường hợp rời rạc	2
1.2	Trường hợp liên tục	3
2	Mô hình SIRD	3
3	Phương pháp Euler	4
3.1	Áp dụng cho mô hình SIR	4
3.2	Áp dụng cho mô hình SIRD	5
4	Phương pháp Runge-Kutta	6
4.1	Áp dụng cho mô hình SIR	6
4.2	Áp dụng cho mô hình SIRD	7
5	Phương pháp Markov Chain Monte Carlo (MCMC)	8
5.1	Thuật toán Metropolis-Hastings	8
5.2	Hiện thực Metropolis Hashing	9
6	Ước lượng giá trị trung bình hệ số R_0	12
7	Phân tích rõ chính sách hạn chế đi lại và cách ly đã ảnh hưởng đến hệ số R_0	13
	Tài liệu	15

1 Mô hình SIR

1.1 Trường hợp rời rạc

Chúng ta xem xét một căn bệnh đang lan rộng khắp Việt Nam, đó là bệnh cúm mới. Các trung tâm phòng chống và ngăn ngừa dịch bệnh đang rất chú tâm đến việc tìm hiểu và thử nghiệm các mô hình cho căn bệnh mới này trước khi nó thực sự trở thành một đại dịch. Chúng ta chia dân số ra thành 3 nhóm: Có nguy cơ nhiễm bệnh, đã nhiễm bệnh và đã hồi phục. Ta cũng đưa ra các giả định cho mô hình này như:

- Không ai có thể vào và rời khỏi cộng đồng cách ly này và không có gì liên quan với bên ngoài (không nhập cư).
- Mỗi người chỉ có thể thuộc 1 trong 3 nhóm như: nhóm có nguy cơ nhiễm bệnh **S** (những người có thể bị nhiễm bệnh); nhóm nhiễm bệnh **I** (đang nhiễm bệnh và có thể lây bệnh cho những người ở nhóm **S**; nhóm đã hồi phục **R** (Đã nhiễm bệnh nhưng đã khỏi và sẽ không thể bị nhiễm bệnh lại, bao gồm là trường hợp tử vong).
- Ban đầu thì chỉ có người thuộc nhóm **S** hoặc **I**.
- Khi ai đó đã bị nhiễm bệnh cúm trong năm nay thì không thể bị cúm trở lại.
- Thời gian trung bình cho quá trình nhiễm và khỏi bệnh là 2 tuần, đây cũng là thời gian để một người bị coi là nhiễm bệnh (thuộc nhóm **I**) và có thể truyền bệnh cho nhóm **S**.
- Mốc thời gian sẽ tính theo từng tuần.

Chúng ta sẽ giả định cho việc khai báo các biến giá trị:

$$\begin{aligned} S(n) &= \text{số lượng người thuộc nhóm có nguy cơ nhiễm bệnh } \mathbf{S} \text{ sau thời gian } n. \\ I(n) &= \text{số lượng người thuộc nhóm nhiễm bệnh } \mathbf{I} \text{ sau thời gian } n. \\ R(n) &= \text{số lượng người thuộc nhóm khỏi bệnh } \mathbf{R} \text{ sau thời gian } n. \end{aligned}$$

Chúng ta bắt đầu mô hình với $R(n)$. Theo giả định thì thời gian nhiễm bệnh của mỗi người là 2 tuần. Do đó sẽ có 50% số người mắc bệnh sẽ hồi phục mỗi tuần:

$$R(n+1) = R(n) + 0.5I(n)$$

Giá trị 0.5 được gọi là tỉ lệ hồi phục mỗi tuần, nó đại diện cho tỷ lệ người thuộc nhóm **I** khỏi bệnh mỗi tuần. $I(n)$ sẽ có 2 trường hợp tăng lên và giảm xuống. Nó sẽ được giảm bởi số lượng người khỏi bệnh mỗi tuần: $0.5I(n)$. Nó sẽ bị tăng lên bởi số lượng người có nguy cơ nhiễm bệnh tiếp xúc với những người nhiễm bệnh và rồi nhiễm bệnh: $a * S(n) * I(n)$.

Chúng ta khai báo a như một tốc độ truyền bệnh hay là hệ số lây nhiễm. Nói cách khác thì a là tỷ lệ số người trong nhóm **S** sẽ tiếp xúc với những người nhiễm bệnh **I**. Ta nhận ra đây là một hệ số xác suất và ban đầu nó được xem như có giá trị không đổi và được tìm thấy từ những điều kiện ban đầu.

Chúng ta sẽ xét tiếp $S(n)$. Nó sẽ giảm duy nhất khi người thuộc nhóm **S** trở thành người bị bệnh thuộc nhóm **I**. Chúng ta có thể dùng tỷ lệ a trên:

$$S(n+1) = S(n) - a * S(n) * I(n)$$

Chúng ta thu được phương trình trên từ giả thuyết về tỷ lệ lây lan a , diễn dịch phương trình có nghĩa là số lượng người thuộc nhóm **S** giảm khi có $a * S(n)$ người tiếp xúc với số lượng người mắc bệnh ở tuần đó ($I(n)$). Tổng hợp lại ta sẽ có mô hình kết hợp như sau:

$$R(n+1) = R(n) + 0.5 * I(n)$$

Đây là phương trình của mô hình SIR, ta có thể giải quyết bằng việc tính toán lặp đi lặp lại và có thể nghiên cứu bằng biểu đồ. Từ đây chúng ta sẽ sử dụng các phương pháp xấp xỉ như Euler, Runge-Kutta để biết được tình hình dịch bệnh trong những tuần tới.

1.2 Trường hợp liên tục

Ta có công thức tổng quát:

$$\frac{dS}{dt} = -\frac{\beta}{N}IS \quad (1)$$

$$\frac{dI}{dt} = \frac{\beta}{N}IS - \gamma I \quad (2)$$

$$\frac{dR}{dt} = \gamma I \quad (3)$$

Bắt đầu với $\gamma = 0.5$. Giá trị 0.5 được gọi là tỷ lệ hồi phục mỗi tuần, nếu có dữ liệu thực thì ta có thể phân tích và tìm ra tỷ lệ này. $I(t)$ sẽ có 2 trường hợp tăng lên và giảm xuống. Nó sẽ được giảm bởi số lượng người khỏi bệnh mỗi tuần: $0.5 * I(t)$. Nó sẽ bị tăng lên bởi số lượng người có nguy cơ nhiễm bệnh tiếp xúc với một người nhiễm bệnh và rồi nhiễm bệnh: $a * S(t) * I(t)$. Chúng ta cũng khai báo a và lấy giá trị 0.000526 cho hệ số lây nhiễm a .

$$\frac{dR}{dt} = 0.5 * I(t)$$

Chúng ta sẽ xét tiếp $S(t)$. Nó sẽ giảm duy nhất khi người thuộc nhóm **S** trở thành người bị bệnh nhóm **I**. Chúng ta có thể dùng tỷ lệ a :

$$\frac{dS}{dt} = -0.000526 * S(t)I(t)$$

Tổng hợp lại ta sẽ có mô hình kết hợp thành các phương trình vi phân như sau:

$$\begin{aligned} \frac{dS}{dt} &= -0.000526I(t)S(t) \\ \frac{dI}{dt} &= 0.000526I(t)S(t) - 0.5I(t) \\ \frac{dR}{dt} &= 0.6I(t) \end{aligned}$$

2 Mô hình SIRD

Đối với các loại bệnh nguy hiểm chẳng hạn như COVID-19 có tồn tại tỉ lệ tử vong nên việc sử dụng mô hình SIR để dự đoán bệnh không còn phù hợp với hoàn cảnh thực tế. Vì vậy cần cải tiến mô hình SIR đó là mô hình SIRD.

Mô hình SIRD cũng được cấu thành tương tự như mô hình SIR, nhưng ở đây chúng ta quan tâm thêm 1 nhóm người đó là **Deaths(D)** - nhóm người tử vong do dịch bệnh. Căn bệnh sẽ tử vong khi họ nhiễm bệnh. Nhưng không thể tử vong ngay lập tức sau khi họ nhiễm bệnh. Vì thế chúng ta phải định nghĩa một hệ số khác gọi là μ để chỉ ra tốc độ tử vong do dịch bệnh. Và tốc độ phục hồi γ không bị ảnh hưởng. Trong mô hình SIRD thì $I(n)$ giảm bởi số lượng người hồi phục (theo tỷ lệ γ) và số lượng người tử vong (theo tỷ lệ μ) như sau:

$$I(n+1) = I(n) - \gamma * I(n) + \frac{\beta * I(n) * S(n)}{N}$$

Số ca tử vong $D(n)$ sẽ được tính tương tự như cách tính số lượng ca hồi phục:

$$D(n+1) = D(n) + \mu * D(n)$$

Đối với mô hình SIRD thì ta có thể áp dụng phương pháp xấp xỉ Euler như mô hình SIR, chỉ cần thêm biến số của số lượng người tử vong và đặc biệt dựa vào những dữ liệu, điều kiện ban đầu ta phải phân tích được tốc độ tử vong μ và tốc độ này là hằng số cho cả quá trình.

3 Phương pháp Euler

3.1 Áp dụng cho mô hình SIR

Đây là một phương pháp xấp xỉ xét các giá trị của các mốc thời gian của một quá trình, các mốc thời gian cách nhau bằng bước nhảy thời gian Δt . Có dạng tổng quát là:

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

Áp dụng cho bài toán trên thì tính toán theo Euler ta sẽ có các phương trình sau:

$$\begin{aligned} S_{n+1} &= S_n + \Delta t * (a * I_n * S_n) \\ I_{n+1} &= I_n + \Delta t * ((a * I_n * S_n) - 0.5 * I_n) \\ R_{n+1} &= R_n + \Delta t * (0.5 * I_n) \end{aligned}$$

Hệ số lây nhiễm a có thể tính được khi biết số lượng người mắc 2 tuần đầu (trường hợp rời rạc), hay là biết được tỉ lệ tiếp xúc giữa 2 nhóm người **S** và **I** (trường hợp liên tục).

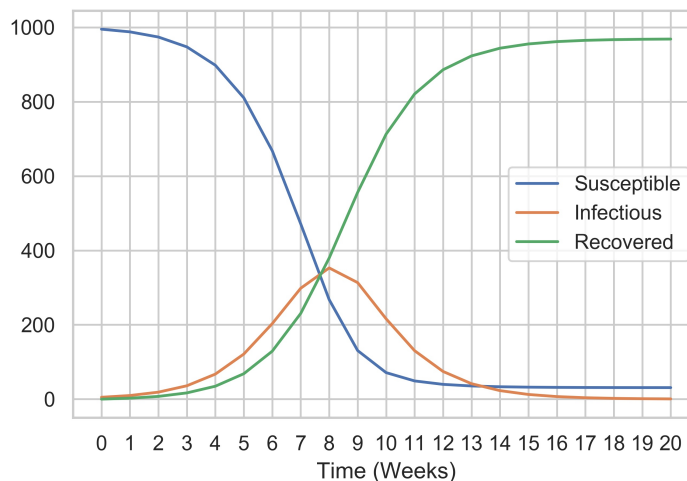
Giả sử, một cộng đồng có 1000 dân cư. Dịch bệnh xuất hiện với 5 người có triệu chứng và có tỉ lệ tiếp xúc (β) là 1.45, tỉ lệ phục hồi (γ) 0.6. Dựa vào mô hình SIR ta suy ra được:

$$\frac{dS}{dt} = -0.00145IS \quad (4)$$

$$\frac{dI}{dt} = 0.00145IS - 0.6I \quad (5)$$

$$\frac{dR}{dt} = 0.6I \quad (6)$$

Việc giải hệ phương trình vi phân trên ta sử dụng công thức Euler và chia thành 1000 đoạn nhỏ ($N=1000$) ta thu được hình 1 và bảng 1.



Hình 1: Biểu đồ biểu hiện dịch bệnh trong 20 tuần đầu tiên sử dụng công thức Bayes ($N=1000$, $\beta = 1.45$, $\gamma = 0.6$).

Week	Susceptible	Infectious	Recovered
0	995	5	0
1	987.78625	9.71375	2.5
2	973.87334	18.76978	7.35687
3	947.36823	35.89001	16.74177
4	898.0667	67.24653	34.68677
5	810.49849	121.19148	68.31004
6	668.0715	203.02273	128.90577
7	471.40264	298.18023	230.41714
8	267.58637	352.90638	379.50725
9	130.65861	313.38095	555.96044
10	71.28703	216.06206	712.65092
11	48.95352	130.36454	820.68194
12	39.6999	74.43588	885.86421
13	35.41501	41.50283	923.08216
14	33.28377	22.88266	943.83357
15	32.17942	12.54568	955.2749
16	31.59403	6.85822	961.54774
17	31.27985	3.7433	964.97686
18	31.11007	2.04143	966.8485
19	31.01798	1.1128	967.86922
20	30.96793	0.60645	968.42562

Bảng 1: Số người có nguy cơ mắc, số người mắc và phục hồi trong 20 tuần tính theo công thức Euler

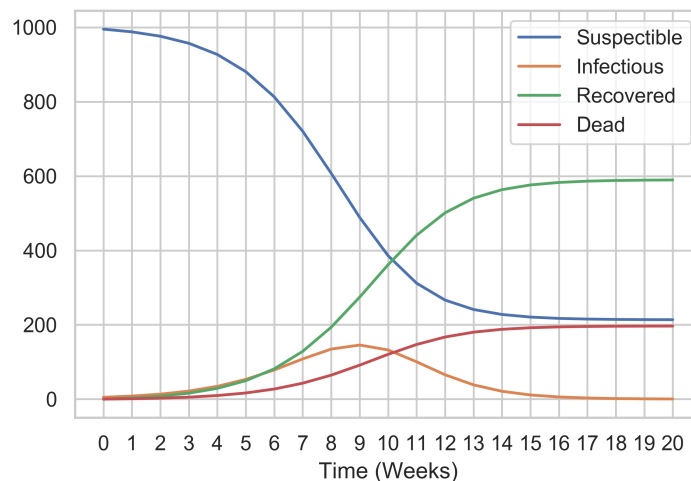
3.2 Áp dụng cho mô hình SIRD

Đối với mô hình SIRD, ta cần phải điều chỉnh công thức (5),(6) và thêm một công thức để diễn tả số ca tử vong (giả sử tỉ lệ tử vong là 0.2). Ta sẽ thu được hình 2 và bảng 2:

$$\frac{dI}{dt} = 0.00145IS - 0.6(\gamma + \mu) \quad (7)$$

$$\frac{dR}{dt} = 0.6I \quad (8)$$

$$\frac{dD}{dt} = 0.2I \quad (9)$$



Hình 2: Biểu đồ biểu hiện dịch bệnh trong 20 tuần đầu tiên sử dụng công thức Bayes ($N=1000$, $\beta = 1.45$, $\gamma = 0.6$, $\mu = 0.2$).

Tuy nhiên, nhược điểm của việc sử dụng phương pháp Euler cho cả hai mô hình trên chính là cho kết quả không chính xác nếu bước nhảy quá nhỏ và tốn nhiều thời gian cho vòng lặp với bước nhảy quá lớn. Nên ta sẽ sử dụng một phương pháp tối ưu và cho kết quả chính xác hơn. Đó là phương pháp Runge-Kutta.

Week	Susceptible	Infectious	Recovered	Dead
0	995	5	0	0
1	984.95949	9.44483	4.19676	1.39892
2	966.46908	17.47964	12.03847	4.01282
3	933.77169	31.18929	26.27927	8.75976
4	879.83284	52.305	50.89662	16.96554
5	799.62475	79.78832	90.4402	30.14673
6	698.84219	106.27836	146.15959	48.71986
7	590.62485	121.73496	215.73015	71.91005
8	495.30154	120.03269	288.49933	96.16644
9	420.32094	104.54127	356.35334	118.78445
10	366.27361	82.73536	413.24327	137.74776
11	329.88961	61.46818	456.48165	152.16055
12	305.94658	43.90056	487.61465	162.53822
13	289.81151	30.19674	509.99381	169.99794
14	279.39218	20.46405	525.10783	175.03594
15	272.65201	13.77769	535.17772	178.39257
16	268.15789	9.14754	542.02093	180.67364
17	265.16845	5.99683	546.62604	182.20868
18	263.26692	3.97247	549.57046	183.19015
19	261.96635	2.58539	551.5862	183.86207
20	261.12124	1.6924	552.88977	184.29659

Bảng 2: Số người có nguy cơ mắc, số người mắc, số người phục hồi và tử vong trong 20 tuần tính theo công thức Euler

4 Phương pháp Runge-Kutta

4.1 Áp dụng cho mô hình SIR

Đây là một phương pháp xấp xỉ cho kết quả chính xác hơn Euler mà ta đã đề cập phía trên. Phương pháp này xét các giá trị của các mốc thời gian của một quá trình, các mốc thời gian cách nhau bằng bước nhảy thời gian h . Có dạng tổng quát là:

$$\begin{aligned} y_{n+1} &= y_n + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4) \\ t_{n+1} &= t_n + h \end{aligned}$$

Các số gia k_1, k_2, k_3, k_4 sẽ được tính như sau:

$$\begin{aligned} k_1 &= f(t_n, y_n) \\ k_2 &= f(t_n + \frac{h}{2}, y_n + \frac{h}{2}k_1) \\ k_3 &= f(t_n + \frac{h}{2}, y_n + \frac{h}{2}k_2) \\ k_4 &= f(t_n + h, y_n + hk_3) \end{aligned}$$

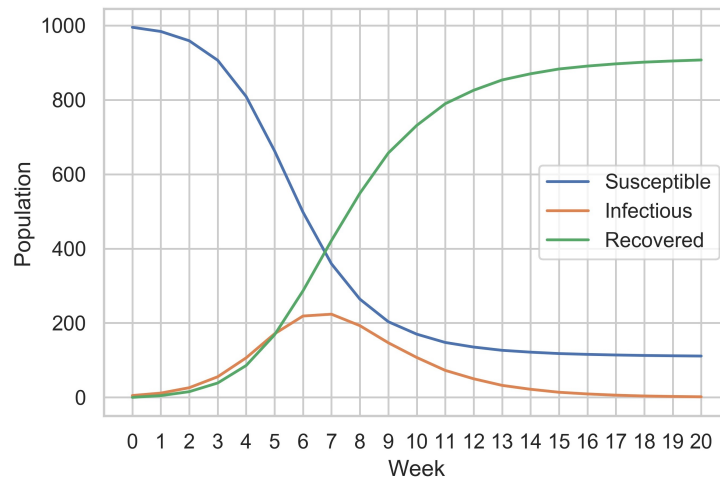
Áp dụng cho bài toán của chúng ta thì theo phương pháp Runge-Kutta bậc 4 ta sẽ ví dụ cho phương trình tính số lượng người của nhóm có nguy cơ nhiễm bệnh **S**, ta được hàm *Susceptible*:

$$(-\text{beta}/N) * I * S$$

Tính các số gia và S qua các mốc thời gian:

$$\begin{aligned} k1S &= h * \text{Susceptible}(\text{beta}, I0, S0, N) \\ k2S &= h * \text{Susceptible}(\text{beta}, I0, S0 + k1S/2, N) \\ k3S &= h * \text{Susceptible}(\text{beta}, I0, S0 + k2S/2, N) \\ k4S &= h * \text{Susceptible}(\text{beta}, I0, S0 + k3S, N) \\ S &= S0 + (k1S + 2 * k2S + 2 * k3S + k4S)/6 \end{aligned}$$

Tương tự như thuật toán Euler, ta sẽ giải hệ phương trình (4),(5),(6) bằng phương pháp Runge-Kutta (bậc 4) với bước chia là 200 và thu được hình 3 và bảng 3 dưới đây:



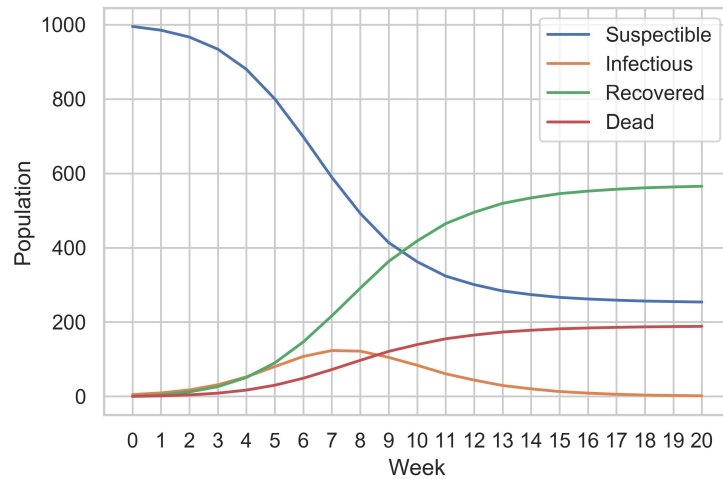
Hình 3: Biểu đồ biểu hiện dịch bệnh trong 20 tuần đầu tiên sử dụng công thức Runge-Kutta (bậc 4) ($\beta = 1.45$, $\gamma = 0.6$).

Week	Susceptible	Infectious	Recovered
0	995	5	0
1	983.79452	11.53295	4.68649
2	958.80644	25.95057	15.33249
3	906.36652	55.35009	38.60647
4	809.00637	106.23796	85.62867
5	662.69954	170.97057	168.17422
6	497.931	218.71657	286.46119
7	359.13827	223.63518	421.66976
8	264.11292	192.8222	548.84111
9	203.34112	146.75003	657.04459
10	169.94828	107.21303	731.27526
11	147.6319	72.58668	789.52582
12	135.26497	49.97431	825.72709
13	126.51646	32.26135	853.39463
14	121.53205	21.75383	870.02679
15	117.80775	13.72826	882.90566
16	115.59788	9.19424	890.74144
17	113.93339	5.9418	896.74293
18	112.67465	3.68217	901.33999
19	111.82314	2.46322	904.47898
20	111.10343	1.58154	907.15082

Bảng 3: Số lượng người có nguy cơ mắc, số người mắc và phục hồi trong 20 tuần đầu tiên tính theo công thức Runge-Kutta.

4.2 Áp dụng cho mô hình SIRD

Tương tự như thuật toán Euler, ta sẽ giải hệ phương trình (4),(7),(8),(9) bằng phương pháp Runge-Kutta (bậc 4) với bước chia là 200 và thu được hình 4 và bảng 4:



Hình 4: Biểu đồ biểu hiện dịch bệnh trong 20 tuần đầu tiên sử dụng công thức Runge-Kutta (bậc 4) ($\beta = 1.45$, $\gamma = 0.6$, $\mu = 0.2$).

Week	Susceptible	Infectious	Recovered	Dead
0	995	5	0	0
1	984.97093	9.44709	4.19197	1.39732
2	966.52287	17.49746	12.01563	4.00521
3	933.89771	31.26633	26.22448	8.74149
4	880.00219	52.55177	50.82138	16.94046
5	800.09594	80.25892	90.21148	30.07049
6	698.25704	107.44004	146.54708	48.84903
7	589.72322	123.31522	216.4508	72.15027
8	492.48117	121.73062	291.01511	97.00504
9	413.6393	104.9553	363.20624	121.06875
10	362.21129	83.76452	418.14415	139.38138
11	323.66086	60.89262	464.7089	154.90297
12	300.69634	43.87418	495.16214	165.05405
13	283.68736	29.37758	519.25652	173.08551
14	273.81296	20.27633	533.9162	177.97207
15	266.34823	13.05309	545.35372	181.78457
16	262.01009	8.85878	552.14885	184.04962
17	258.8049	5.79173	557.24203	185.74734
18	256.46145	3.62694	561.00595	187.00198
19	255.01049	2.44492	563.35368	187.78456
20	253.84767	1.58173	565.24485	188.41495

Bảng 4: Số lượng người có nguy cơ mắc, số người mắc và phục hồi trong 20 tuần đầu tiên tính theo công thức Runge-Kutta.

5 Phương pháp Markov Chain Monte Carlo (MCMC)

MCMC cho phép chúng ta suy ra được tập mẫu từ một phân phối xác suất bất kì mặc dù chúng ta không thể tính toán nó. Nó là một thuật toán quan trọng trong thống kê Bayes vì nó cho phép ta tính toán xác suất của phân bố hậu nghiệm (cái mà thường rất khó để tính hiển ra trong trường hợp nhiều tham số).

5.1 Thuật toán Metropolis-Hastings

MCMC là một lớp các phương pháp mà trong đó Metropolis-Hastings là một thuật toán cụ thể của nó. Metropolis-Hastings được ứng dụng tốt trong không gian nhiều chiều, gồm các bước:

1. Khởi tạo giá trị 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 kì $p(\beta, \gamma)$.
4. Nếu $p(\beta, \gamma)$ là đối xứng, nghĩa là $p(\beta^*, \gamma^* | \beta, \gamma) = p(\beta, \gamma | \beta^*, \gamma^*)$, ta gán r là xác suất giữ lại

β^* và γ^* bằng công thức

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

và đi đến bước 6.

5. Nếu không đối xứng, gán

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

và đi đến bước 6.

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 $\beta_{i+q} := \beta_i$ và $\gamma_{i+1} := \gamma_i$ 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.

5.2 Hiện thực Metropolis Hashing

Giả sử:

- Hàm mật độ xác suất $p(x)$ của phân bố cần lấy là

$$p(x) = N(x|mean, cov) \text{ hay } x \sim N(mean, cov)$$

trong đó:

x là một mảng gồm 2 phần tử với $x[0] = \beta$; $x[1] = \gamma$ và

$$\begin{aligned} mean &= \begin{bmatrix} 3 & 4 \end{bmatrix} \\ cov &= \begin{bmatrix} 1.0 & 0.7 \\ 0.7 & 1.2 \end{bmatrix} \end{aligned}$$

- Hàm lấy mẫu phân bố dự định $q(x)$ có phân bố chuẩn mean (trung bình) = x , covariance (hiệp phương sai) = cov :

$$q(x'|x) = N(x'|x, cov) \text{ hay } x'|x \sim N(x, cov)$$

x' là mẫu lấy từ tham số đầu vào x

```
def Metropolis_Hashing(x, N):
    sample = np.zeros(shape=(N, 2))
    sample[0] = x
    for i in range(N-1):
        x_cur = sample[i]
        x_star = proposal_distribution(x_cur)
        rate_anpha = (prior_function(x_star)*p_pdf(x_cur,
            x_star))/(prior_function(x_cur)*p_pdf(x_star, x_cur))
        if(min(1, rate_anpha) >= np.random.uniform(0.0, 1.0)):
            # r>=random(0,1)
            sample[i+1] = x_star
        else:
            sample[i+1] = x_cur
    return sample
```

Gọi hàm *Metropolis – Hasting*(x, N) với $x = [1.2, 0.6]$ (tức là $\beta_0 = 1.2$ và $\gamma_0 = 0.6$) và N (số lần lặp) = 10000

```
intt = [1.2, 0.6]
S = Metropolis_Hashing(init, 10000)
```

Ban đầu $x_{cur} = init = [1.2, 0.6]$, ta gán $x_{star} = proposal_distribution(x_{cur})$, trong đó hàm *proposal_distribution*(x_{cur}) được định nghĩa như sau:

```
def proposal_distribution(x_cur): #sampler_from_q
    cov = np.array([[1.0, 0.],
                    [0., 1.0]])
    x_proposed = np.random.multivariate_normal(x_cur, cov)
    return x_proposed
```

Tiếp theo ta tính *rate_alpha*.

$$rate_alpha = \frac{prior_function(x_{star}) * p_pdf(x_{cur}, x_{star})}{prior_function(x_{cur}) * p_pdf(x_{star}, x_{cur})} \quad (10)$$

Trong đó, các hàm *prior_function* và *p_pdf* được định nghĩa như sau:

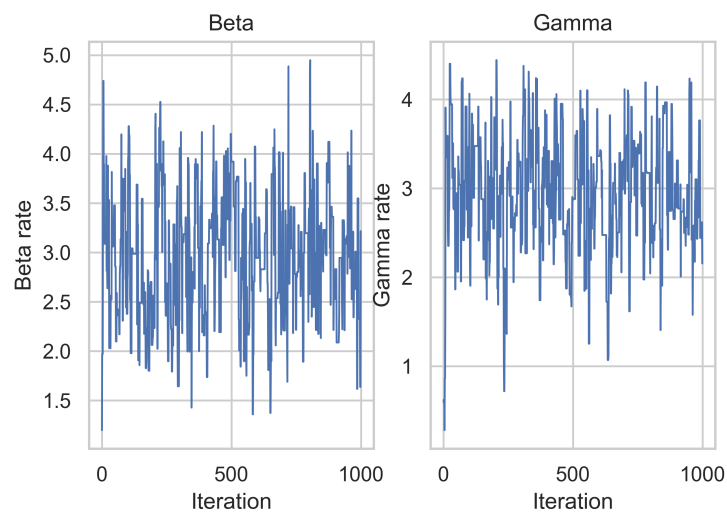
```
def prior_function(x): #p_pdf
    # pi(B,Y)
    mean = np.array([3.0, 3.0])
    cov = np.array([[0.5, 0],
                    [0, 0.5]])
    return stats.multivariate_normal.pdf(x, mean, cov)
def p_pdf(x_star, x_cur): #q_pdf
    cov = np.array([[1.0, 0.],
                    [0., 1.0]])
```

Sau khi có kết quả *rate_alpha*, ta thực hiện bước 5. Sau đó ta khởi tạo q ngẫu nhiên từ hàm phân phối đều liên tục $U(0, 1)$ và kiểm tra xem nếu r (từ bước 5) $\geq q$ thì thêm x_{star} vào tập mẫu, nếu không thì thêm x_{cur} vào tập mẫu.

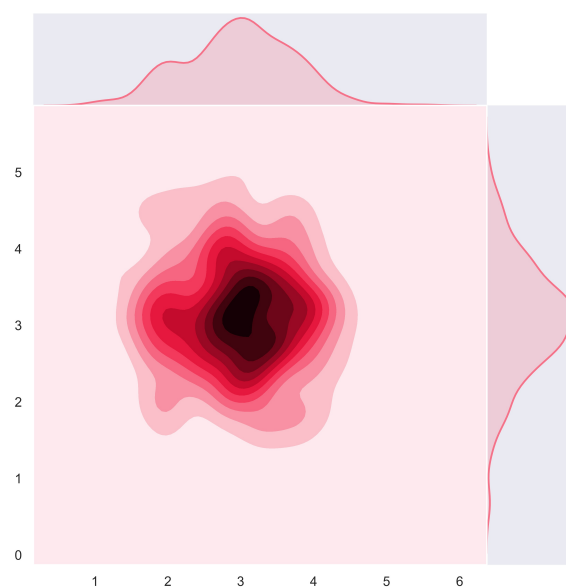
Cuối cùng ta có kết quả của các cặp (β, γ) như sau:

Beta	Gamma
1.2	0.6
1.970689263	0.622809345
1.970689263	0.622809345
1.970689263	0.622809345
3.494700343	0.282366611
4.738235995	0.866065951
4.738235995	0.866065951
...	...
2.327645077	2.442879437
2.327645077	2.442879437
2.327645077	2.442879437
1.63768991	2.618803339
1.63768991	2.618803339
3.215767129	2.155656508

Bảng 5: Giá trị các cặp mẫu (β, γ) được trả về.



Hình 5: Tỷ lệ Beta và Gamma



Hình 6: Phân bố Beta và Gamma

6 Ước lượng giá trị trung bình hệ số R_0

Ta có:

$$R_0 := \frac{\beta}{\gamma}$$

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:

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

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:

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

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

$$E(R_0) \approx \sum_{i=1}^m \pi(X | \beta_i, \gamma_i) \frac{\beta_i}{\gamma_i}$$

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 và $\pi(X | \beta, \gamma)$ được tính bằng công thức:

$$\pi(X | \beta, \gamma) = \prod_{i=1}^n \frac{\gamma^\beta}{\Gamma(\beta)} X(t_i)^{\beta-1} \exp(-\gamma X(t_i))$$

với X là phân bố xác suất về số ca mắc bệnh và phục hồi quan sát được tại một khu vực gồm một số quốc gia.

```
def x_pdf(x, mean, cov):
    return stats.multivariate_normal.pdf(x, mean, cov)

def _function(a, b, y):
    return (pow(y, b) * pow(a, b-1) * math.exp(-y*a)) / gms(b)

def likelihood_function(data, B, Y, m):
    data_arr = data
    mean = np.mean(data_arr[:, [0, 1]], axis=0)
    cov = np.cov(data_arr[:, [0, 1]].T)
    b = 1.2
    y = 0.6
    re = 1
    for i in range(m):
        a = x_pdf(data_arr[:, [0, 1]][i, :], mean, cov)
        re = re * _function(a, b, y)
    return re

def e_function(data, x, n, m):
    re = 0
    for i in range(n):
        a = x[i, 0] / x[i, 1]
        re = re + likelihood_function(data, x[i, 0], x[i, 1], m) * a
    return re
```

Ta lấy dữ liệu của 3 quốc gia: Việt Nam, Lào, Campuchia ¹
Sau khi chạy đoạn code:

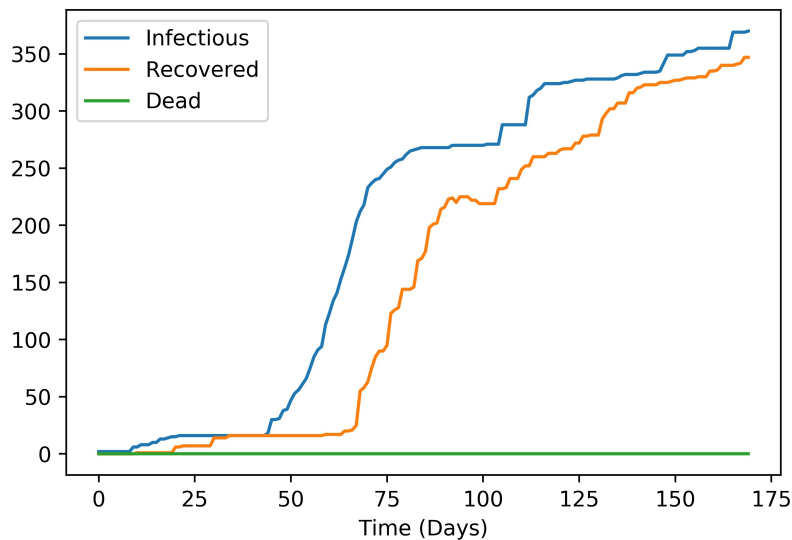
```
e_function(data,c,1000,507)
```

với *data* là dữ liệu số ca mắc bệnh và phục hồi quan sát được của 3 quốc gia trên, *c* là mẫu đã lấy bằng thuật toán **Metropolis-Hastings** và 1000 là kích thước mẫu còn 507 là độ lớn mảng dữ liệu, ta thu được:

1.196896611261779e-188

Hình 7: Giá trị trung bình hệ số R_0

7 Phân tích rõ chính sách hạn chế đi lại và cách ly đã ảnh hưởng đến hệ số R_0



Hình 8: Biểu đồ các ca nhiễm bệnh và phục hồi theo thời gian ở Việt Nam bắt đầu từ ngày 22-01-2020.

Nếu chính sách hạn chế đi lại và cách ly có hiệu quả thì tỷ lệ hồi phục (γ) sẽ ngày càng lớn hơn tỷ lệ mắc bệnh (β), dẫn tới R_0 ngày càng nhỏ. Nếu quản lý tốt công tác này ($R_0 < 1$) thì sẽ không có đợt bùng phát dịch xảy ra.

Việt Nam là một trong những quốc gia thực hiện rất tốt việc này. Nguyên nhân là do:

- Hạn chế tối đa việc ra bên ngoài, chỉ ra ngoài khi thực sự cần thiết.
- Nếu buộc phải ra ngoài luôn luôn đeo khẩu trang, giữ khoảng cách tiếp xúc là 2m.
- Thực hiện khai báo y tế, cập nhật tình hình sức khỏe hàng ngày, giữ liên hệ thường xuyên với cán bộ y tế, cơ sở y tế.

¹Dữ liệu được lấy từ trang web: https://github.com/CSSEGISandData/COVID-19/tree/master/csse_covid_19_data/csse_covid_19_time_series - thời gian là từ ngày 22-1-2020 đến ngày 9-7-2020.

- Và đặc việc là thực hiện chính sách cách ly bắt buộc 14 ngày đối với mọi người nhập cảnh.

Tính R_0 cho Việt Nam theo công thức trên:

```
e_function(vietnamese,c,1000,169)
```

với *vietnamese* là dữ liệu số ca mắc bệnh và phục hồi quan sát được của Việt Nam, *c* là mẫu đã lấy bằng thuật toán **Metropolis-Hastings** và *1000* là kích thước mẫu còn *169* là độ lớn mảng dữ liệu, ta thu được:

```
1.0088595177089162e-205
```

Hình 9: Giá trị trung bình hệ số R_0 của Việt Nam

Nhờ thực hiện tốt công tác trên, R_0 của Việt Nam rất nhỏ ($R_0 < 1$) và sẽ không có đợt bùng phát dịch.



Tài liệu

- [1] Frank Giordano, William P Fox, Steven Horton *A First Course in Mathematical Modeling* Nelson Education, 2013.
- [2] Steve Brooks et al. *Handbook of Markov Chain Monte Carlo*. CRC press, 2011.
- [3] W. K. Hastings. "Monte Carlo Sampling Methods Using Markov Chains and Their Applications". In: *Biometrika* 57 (1) (1970), pp. 97-109