

# 新型コロナウイルス(CoVid-19)とSIRモデル について

Takeshi Nagae

2020年4月3日 (ver 2.2)

- 新型コロナウイルス感染症(CoVid-19)の拡大が止まりません。毎日のように感染発生のニュースが流れ、多くの方が、これからの生活に不安を感じ、保障もなくいつ終わるのか分からない自粛要請にウンザリしていると思います。
- この文章は、医学・疫学について**素人である僕**が、手元にある西浦先生の論文とか、数理生物学のテキストと首っ引きになりながら、**私よりも素人である方々**に向けて、感染症拡大メカニズムを記述・分析できる SIR (Susceptible-Infectious-Recovered) モデルをなるべく易しく解説しようというものです。
- あくまで**素人が書いている文章**ですから、以下のことに注意して下さい：
  - 学術的に間違っただけを言っていたり、医学・疫学分野で常識とされている知見が反映されていないかもしれません。
  - 分析の方法や結果については、あくまで「計算してみた」程度の例であり、CoVid-19の現状を正しく把握できていなかったり、将来を正しく予測できる保障はありません。
  - 本当は参考文献とかちゃんとつけないといけないんですが、後回しにしています。
- 更新履歴：
  - ver 1.0→2.0: ver1.0で推計した感染率 $\beta$ と回復/死亡率 $\gamma$ がやっぱり実態に合っていないので、データにフィッティングするのはいったん諦め、[新型コロナウイルス感染症対策専門家会議の見解など](#)(4月1日版)で示されている東京都での  $R_0 = 1.7$  と、[この辺](#)とか[この辺](#)で使われている平均感染日数 $\gamma^{-1} = 5$ から  $\beta = 0.34$  として書き直しました。
  - ver2.0→2.1: 「回復または死亡」を「隔離」と表現してましたが、政策としての「高齢者隔離」と混同しそうなので素直に「回復/死亡」と書くことにしました。
  - ver2.1→2.2: 83%の活動自粛という表現がわかりにくい(活動を83%にするのか83%減らすのか?)ので、83%レベルの自粛(17%の活動削減)と書くことにしました。

## ❖ SIRモデルとは

- SIRモデルというのは、対象とする集団を、**3種類の個体群**～感染症感受性個体(S:

susceptible)、感染個体(I: infectious)および隔離個体(R: recovered)~に分類し、そのダイナミクスを常微分方程式として記述するものです。

- そうですよ。寝ますよね。そんなん分かるかよ。アホかと。おっしゃるとおりです。路線を変えましょう。
- 少しイメージしやすいように、ある町に  $N = 1000$  人の住人がいたとして、そのうちの1人がCoVid-19に感染したとしましょう。感染した人を**感染者(I)**と呼ぶことにします。(I)と書くのは、infectious (感染性)の頭文字です。
- 残る999人はこれからCoVid-19にかかるかもしれない人で「感受性を有する」と言うようですが、わかりにくいので、**未感染者(S)**と呼ぶことにします。Sは susceptible (感受性)の頭文字です。
- 感染者(I)は、何日かすると回復するか亡くなります。どちらにしても、もうこの町でCoVid-19を広めることはありません。この人たちを**回復/死亡者(R)**と呼びます。Rは recovered (回復した) と removed (死亡によって除外された)の共通の頭文字です。CoVid-19によって亡くなる方がここまで多くなければ回復者だけでいいんですが。
- 日本で最初の感染者が確認されたのは2020年1月22日です。この日を0日目として、 $t$  日目の**未感染者**の数を  $S(t)$ 、**感染者**の数を  $I(t)$ 、**回復/死亡者**の数を  $R(t)$  で表すことにします。はい、記号が出ると目が痛くなる人いますね。しばらく目を休めましょう。

- 起きて下さい。

- 町の話に戻します。最初の日を  $t = 0$  日目とするとき、

- > **未感染者(S)** : 999人
- > **感染者(I)** : 1人
- > **回復/死亡者(R)** : 0人

ですから、 $S(0) = 999$ ,  $I(0) = 1$ ,  $R(0) = 0$  となります。

- さて、明日までの間に、感染者は増えるんでしょうか。順を追って考えていきます。まず、**未感染者(S)**の中の一人をAさんとします。Aさんは一日の間に、町の中の1人に会う(接触する)とします。**Aさんが会う相手は、100人の住人からランダムに選ばれる**とします。もし相手としてAさん自身が選ばれた場合は、Aさんは誰にも会わなかったということにしましょう。
- さて、Aさんが**未感染者(S)**に会ったのならAさんがCoVid-19にかかることはありませんが、もし相手が**感染者(I)**だったら感染するリスクがあります。**未感染者(S)が感染者(I)と会った時に感染する(感染者からCoVid-19を伝染される確率、感染率)**を、ギリシャ文字の  $\beta$  (ベータ)で表すことにします。
- なんだよ、また記号かよ、しかも何だよギリシャ文字って。いいじゃん  $b$  とかでって思いますね。泣きたくなりますよね。僕もそうです。ただ、僕が見た文献の中には**感染率**を  $\beta$  で表してるもの多くて、そういうのを見た時に「あ、**感染率**ね」って思えるように、ここでは  $\beta$  を使います。涙は自分で拭きましょう。

- $\beta$ を使うと「Aさんが明日までに**感染者(I)**になる確率」が表せます。え？それって  $\beta$  じゃないの？って思ったアナタ。残念、違います。Aさんが感染するのは、今日会った相手が**感染者(I)**だった場合に限られることを思い出して下さい。最初の日の**感染者(I)**は1名だけなので、Aさんが**感染者(I)**に会う確率は  $\frac{1}{1000} = \frac{I(0)}{N}$  です。ここで、 $N$ はすべての住民の数です。最初に言いましたよ。これに  $\beta$  を掛けたものが「Aさんが明日までに**感染者(I)**になる確率」です。
- 具体的な数字を入れると分かりやすいかもしれませんが、 $\beta = 0.34$ とすると、Aさんが明日までに**感染者(I)**になる確率は、 $0.001 \times 0.34 = 0.00034$  です。0.034% ならず伝染されることはなさそうです。しかし、他の  $S(0) = 999$ 人の**未感染者(S)**も同程度に**感染者(I)**になる可能性がありますから、町全体では、明日までに

$$S(0) \frac{I(0)}{N} \beta = 999 \times \frac{1}{1000} \times \beta$$

だけの**未感染者(S)**が感染することになります。 $\beta = 0.34$ だと0.340人ですね。「なんだ、じゃ0人でいいじゃん」と言いたいところですが、ここは0.340人だけが感染する、つまり**未感染者(S)**が0.340人だけ減って、**感染者(I)**が0.340人だけ増える、ということにします。

「なんだよ、おかしいじゃん。誰なの、その0.340人」って思いますよね。涙が出そうになりますよね。僕もそうです。でも、感染者が1人、2人って**整数の値しか取らない**(離散的と言います)モデルは直感的だけど、起こり得る場合の数がムチャクチャ多くなってしまふんです。例えば、住民が100人だった場合、起こり得る組み合わせを「未感染99・感染1・隔離0」「未感染98・感染2・隔離0」「未感染98・感染1・隔離1」...と数えていくと、全部で  $9.4 \times 10^{159}$  個くらいになります。94の後ろに0を158個かかないといけない。観測可能な宇宙の星の数が  $10^{22}$  個とか、全宇宙の原子の数が  $10^{80}$  個とか言われてますから、まあとんでもない数です。もちろん、100人じゃなくて1000人だと組み合わせはもっと増えます。その組み合わせを追いかけるよりも「0.340人」という**実数で表現するモデル**を受け入れた方がラクなので、いまはそうすることにしましょう。

- さて、もう一つ、明日までに**感染者(I)**がCoVid-19から回復する(もしくは亡くなる)可能性もあります。1日あたりに**感染者(I)**が**回復/死亡者(R)**になる確率(回復/死亡率)を、ギリシャ文字の  $\gamma$  (ガンマ)で表します。またギリシャ文字です。もう諦めはつきましたね？
- 回復/死亡率にも具体的な数値として  $\gamma = 0.200$ を入れておきます。 $t = 0$ 日目の**感染者(I)**は  $I(0) = 1$ 人ですから、 $R(0) \times \gamma = 1 \times \gamma$  で、明日までに0.200人が回復する、つまり、**感染者(I)**が0.200人だけ減って、**回復/死亡者(R)**が0.200人だけ増えることになります。

- ここまでの話をおさらいしておきましょう。この町では、 $t = 0$ 日目に、**未感染者(S)**、**感染者(I)**および**回復/死亡者(R)**が、それぞれ、 $[S(0), I(0), R(0)] = [99, 1, 0]$ だけいます。 $t = 1$ 日目までの間に、次のように変化します：

- **未感染者(S)**は  $\frac{S(0)I(0)}{N}\beta = 0.340$  だけ減る。
- **回復/死亡者(R)**は  $I(0)\gamma = 0.200$  だけ増える。
- **感染者(I)**は  $\frac{S(0)I(0)}{N}\beta = 0.340$  だけ増え、 $I(0)\gamma = 0.200$  だけ減る (差し引きで0.140増える)。

その結果、 $t = 1$ 日目の**未感染者(S)**、**感染者(I)**および**回復/死亡者(R)**は、それぞれ、 $[S(1), I(1), R(1)] = [998.66, 1.140, 0.200]$ となります。

- 次の日も同じように感染状況が変化します。初日に比べて**感染者(I)**が少し増えていいますから、**未感染者(S)**が**感染者(I)**に接触する確率が少し増え、それだけ**非感染者(S)**から**感染者(I)**になる人数が増えます。同じように**感染者(I)**が増えることで、回復・死亡する人も増えますから、**感染者(I)**から**回復/死亡者(R)**になる人数も増えます。
- 具体的に計算してみると、

- **未感染者(S)**は  $\frac{S(1)I(1)}{N}\beta = 0.387$  だけ減る。
- **回復/死亡者(R)**は  $I(1)\gamma = 0.228$  だけ増える。
- **感染者(I)**は  $\frac{S(1)I(1)}{N}\beta - I(1)\gamma = 0.159$  だけ増える。

となった結果、 $t = 2$ 日目の感染状況は

$[S(2), I(2), R(2)] = [998.273, 1.299, 0.428]$ となります。まだまだパンデミックと呼ばれる状態には、ほど遠いようです。都心部とは違い、たった1000人しかない町なので、このまま感染が広がらないかもしれません。

- ではこの町はCoVid-19拡大から免れるのでしょうか。これは、上の手続きを繰り返すことで分析できます。いま、 $t$ 日目の感染状況が  $S(t), I(t), R(t)$  で表されている時、 $t + 1$ 日目の感染状況は、

$$S(t+1) = S(t) - \frac{S(t)I(t)}{N}\beta$$

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

$$R(t+1) = R(t) + I(t)\gamma$$

と表せます。この数式を計算機に入れて計算すれば、何日分もの感染状況があつという間に求められます。もし、Python が使えるなら、次のコードを実行してみてください。18行目に上の計算式が実装されています。

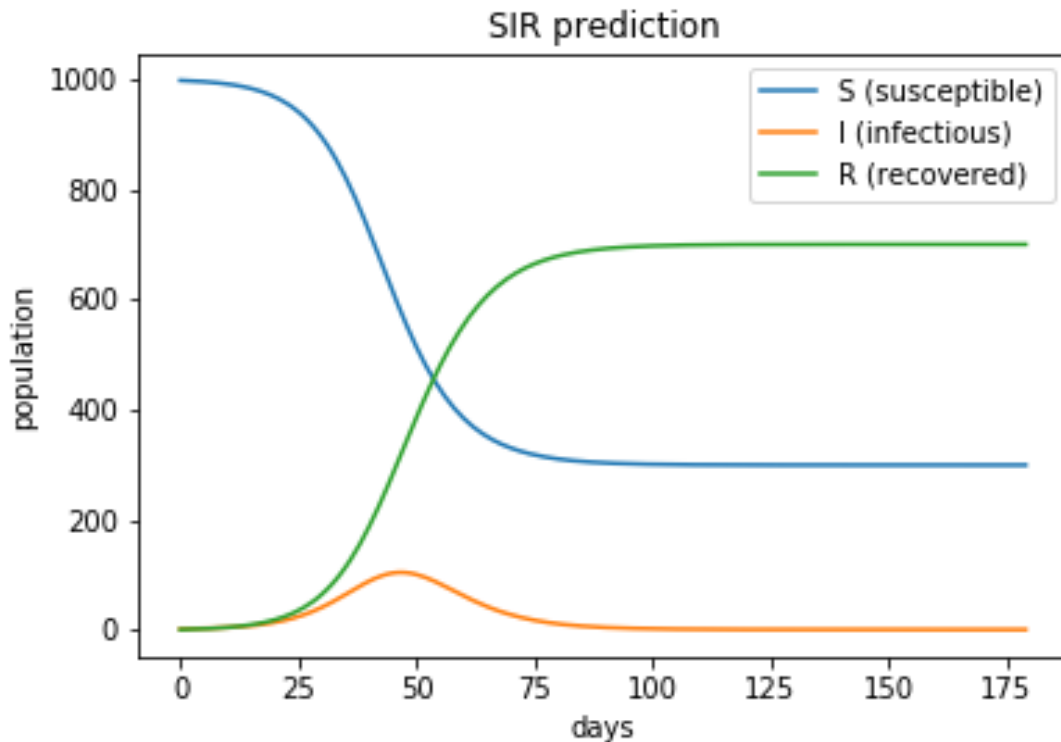
```
1 # モジュールのインポート
2 import numpy as np
```

```

3  from matplotlib import pyplot as plt
4  # パラメータ
5  R0=1.7
6  gamma = 0.2 # 回復/死亡率
7  beta = R0*gamma # 感染率
8  print("β :%4.2f γ :%4.2f R0:%4.2f" % (beta, gamma, R0))
9  N = 1000 # 全人口
10 IO = 1 # 初期感染者数
11 T = 180 # 分析日数
12 DT = 1
13 # SIRモデルで各日の感染状況を計算
14 X = np.zeros((T,3)) # 感染状況を格納する変数
15 X[0,:] = (N-IO, IO, 0) # 初期状態を与える
16 for i in range(T-1):
17     S, I, R = X[i,:]
18     X[i+1,:] = X[i,:] + [-S*I/N*beta, S*I/N*beta - I*gamma,
19                          I*gamma]
19 # 実行結果を表示
20 fig, ax = plt.subplots()
21 labels = ["S (susceptible)", "I (infectious)", "R (recovered)"]
22 for i in range(3):
23     ax.plot(range(T), X[:, i], label=labels[i])
24 plt.legend()
25 plt.title('SIR prediction')
26 plt.xlabel('days')
27 plt.ylabel('population')
28 plt.show()
29 fig.savefig('SIR_prediction.png')
30 print("感染ピーク : %3d日目, 感染者%4.2f人, 重篤者%4.2f人 "%
      (np.argmax(X[:, 1]), np.max(X[:, 1]), np.max(X[:, 1])*0.055))

```

- 最初の感染が確認されてから半年(180日)間の感染状況を実際に計算した結果が、こちらのグラフです。横軸が日付、縦軸が人数をとって感染状況を表しています。青の曲線が**未感染者(S)**、オレンジ色の曲線が**感染者(I)**、緑の曲線が**回復/死亡者(R)**を、それぞれ表しています。



**Figure 1:** SIRモデルを用いた将来予測。  $\beta = 0.34$ ,  $\gamma = 0.2$ ,  $R_0 = 1.7$ ,  $N = 1000$ 。  
 青が未感染者(S)、オレンジが感染者(I)、緑が回復/死亡者(R)

- これは...ちょっと**衝撃的**です。  $t = 0$ 日目での感染者はたった1人でした。  
 $t = 1$ ,  $t = 2$ でも 1.085, 1.180 と二人にも満たなかったはずですが、ところが、 $t = 16$ 日目以降は毎日1名以上が新たに感染しはじめ、 $t = 34 \sim 36$ 日目では5名以上が新規感染します。感染者は  $t = 47$ 日目にピークを迎え、その数は約104人となります。重篤化率が5.5%だとすると5~6名が重症化することになり、現在の日本の医療体制では、そのすべての命を救うことは**極めて困難**となります。

[東洋経済オンライン荻原氏のサイト](#)では、2020年4月1日現在、日本では感染者数2107名のうち、重症化しているのは60名、死亡者は57名ですから、ざっくり見積もって重篤化率は5.6%です。[日本集中治療医学会](#)によると、日本ではICUベッド数が**人口10万人あたりの5床程度**とのことですから、**人口1000人の町で5~6名もの重症者の手当ては到底できません**。

## ❖ 基本再生算数

- SIRモデルを特徴づける2つのパラメータ、**感染率** $\beta$ と回復/死亡率  $\gamma$  は、それぞれ、感染症の特徴を判りやすく説明してくれます。

- まず、回復/死亡率は「1人の感染者(I)が1日あたりに回復もしくは死亡する確率」ですから、その逆数  $\frac{1}{\gamma}$  は**平均的な感染期間**を表します。例えば、上の計算例で使った  $\gamma = 0.2$ の場合、平均感染期間は  $\frac{1}{\gamma} = 5$ 日となります。もうちょっと長いような気もしますが、武漢の症例などをもとに計算されているみたいです。
- 次に、感染率  $\beta$  は「1人の感染者(I)が1回の接触あたりに非感染者(S)を感染させる確率」を表しています。ここで、感染初期においては、感染者(I)が出会う相手がほぼ全員が非感染者(S)となることに注意しましょう。このことは、感染初期において、感染率  $\beta$  は「感染者(I)が増加する割合」を表すと言っていいでしょう。一方、回復/死亡率  $\gamma$  は「感染者(I)が減少する割合」を表します。そこで、この2つの割合の比：

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

を**基本再生算数** (basic reproduction number)といい「(感染初期において) 1人の感染者がどれだけの二次感染者を産むか」を表す数値として用いられます。

- 基本再生算数  $R_0$  は、感染が拡大するか否か、感染が拡大するとしたらどの程度の速さで拡大するのか、の指標となります。まず  $R_0 < 1$  の場合、感染者1人あたり2次感染者は1人未満ですから、感染は拡大しないことが期待できます。 $R_0 < 1$  ということは  $\beta < \gamma$  を意味していますから、新しく感染する人より回復していく(もしくは死亡する)人の方が多いわけです。一方、人類を悩ませてきた感染症のほとんどは  $R_0 > 1$  です。容易に想像できるように、 $R_0$  が大きいほど、感染は急速に拡大します。
- 上の計算例で用いた  $\beta = 0.24$ ,  $\gamma = 0.2$  から求められる基本再生算数は

$$R_0 = 1.7$$

となります。インフルエンザの基本再生算数は2~3だと言われています。欧米だと  $R_0 = 2\sim 4$  くらいだったりします。

- 基本再生算数  $R_0$  を用いると「集団のうち何人が免疫を持っていれば感染が拡大しないか」という**集団免疫(herd immunity)**の閾値を求めることができます。全員が未感染者(S)であるとき、1人の感染者が生み出す2次感染者が  $R_0$  でした。ここで、集団の中で  $p$  だけの割合が免疫を持っていたとしましょう。この時、未感染者(S)は全体の  $1 - p$  だけしかいませんから、2次感染者の数は  $R_0(1 - p)$  になります。これが1未満なら感染は拡大しないはずで、 $R_0(1 - p) < 1$  を満足するためには、免疫率  $p$  が

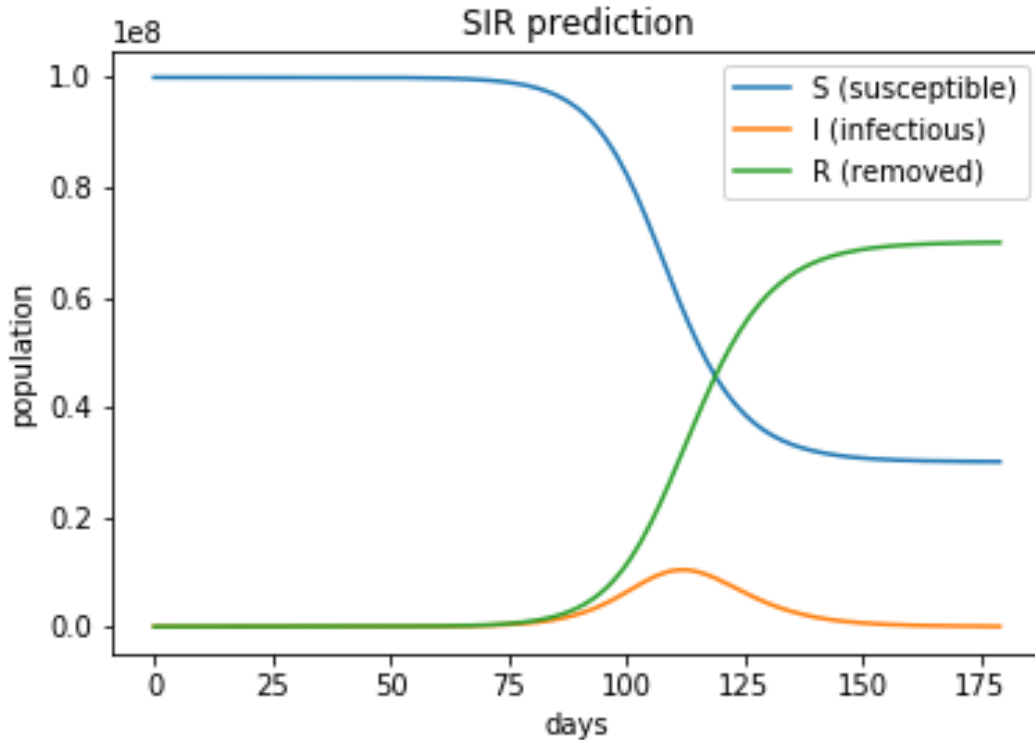
$$p > 1 - \frac{1}{R_0}$$

でなければなりません。上の計算例だと  $R_0 = 1.7$  なので  $p > 0.412$ 。**41.2%の日本人が免疫を持てばCoVid-19は国内では拡大しない勘定**になります。

## ❖ 集団免疫戦略と経済活動の自粛

- 日本全国の41.2%の人が免疫を獲得するまでに何日かかるんでしょう。上で使ったパラメータをそのままに、集団人口を1億人にして、初期感染者数を20人にして計算し

てみました(プログラムの9行目と10行目を書き換えれば計算できます)。計算してみると、41.2%の人が回復/死亡者(R)となるのは117日目です。ただし、その前の112日目に感染者(I)がピークに達し、ピーク時の総感染者数が**1033.2万人**、うち重篤者は**56.8万人**にもものぼるため、集団免疫を獲得する前に**医療破綻**が起きてしまいます。



**Figure 2:** SIRモデルを用いた日本全体の感染状況予測。

$$\beta = 0.34, \gamma = 0.2, R_0 = 1.7, N = 1 \times 10^8。$$

青が未感染者(S)、オレンジが感染者(I)、緑が回復/死亡者(R)

- そこで、日本政府がやっているのが経済活動の自粛です。SIRモデルの枠組みで考えると「未感染者と感染者が会う機会を減らす」ことに相当します。例えば、活動を自粛して、一日あたりに会う人を(普段を100%として) **76%**にまで減らす(24%削減する)とどうなるでしょう。上の例で説明してみます。 $t = 0$ 日目では、1000人の住人のうち999人が未感染者(S)、1人が感染者(I)でした。自粛前は、未感染者(S)のAさんが感染者(I)に出会う確率は $\frac{1}{1000}$ でした。 $t$ 日目に $N$ 人の住民のうち、未感染者(S)が $S(t)$ 人で感染者(I)が $I(t)$ 人だった時、1人の未感染者(S)が感染者に会う確率は $\frac{I(t)}{N}$ でした。自粛によってこれを76%したら感染状況はどう変わるでしょうか。難しい計算が必要ですか？そんな事ありません。未感染者(S)が感染者(I)に変わる確率が $\frac{S(t)I(t)}{N}\beta$



から  $\frac{S(t)I(t)}{N}\beta \times 0.76$  になるだけです。つまり、 $\beta$  を 0.76 倍すれば、自粛後の感染状況を予想できるわけです。

- 計算してみた結果、下記のようにになりました。ピーク時の総感染者数は**282万人**となり、何もせず放任した時(1033万人)の27%程度にまで減らせそうです。ただし、**重篤化数は15.5万人**。現況のICUベッド数から考えると気が遠くなるような数です。自粛によってピークの時期が117日目から234日目にずれているのがわずかな救いでしょうか、8ヶ月あれば、なんとかなるかもしれません。
- さて、活動自粛によって出会う機会を減らすことで、集団免疫を獲得するまでの時間が伸びます。この例だと、国民の41.2%にあたる4120万人が免疫を獲得するのは317日目。2020年1月22日から起算すると2020年12月4日。なんとか年内には通常生活を取り戻せるかもしれません。

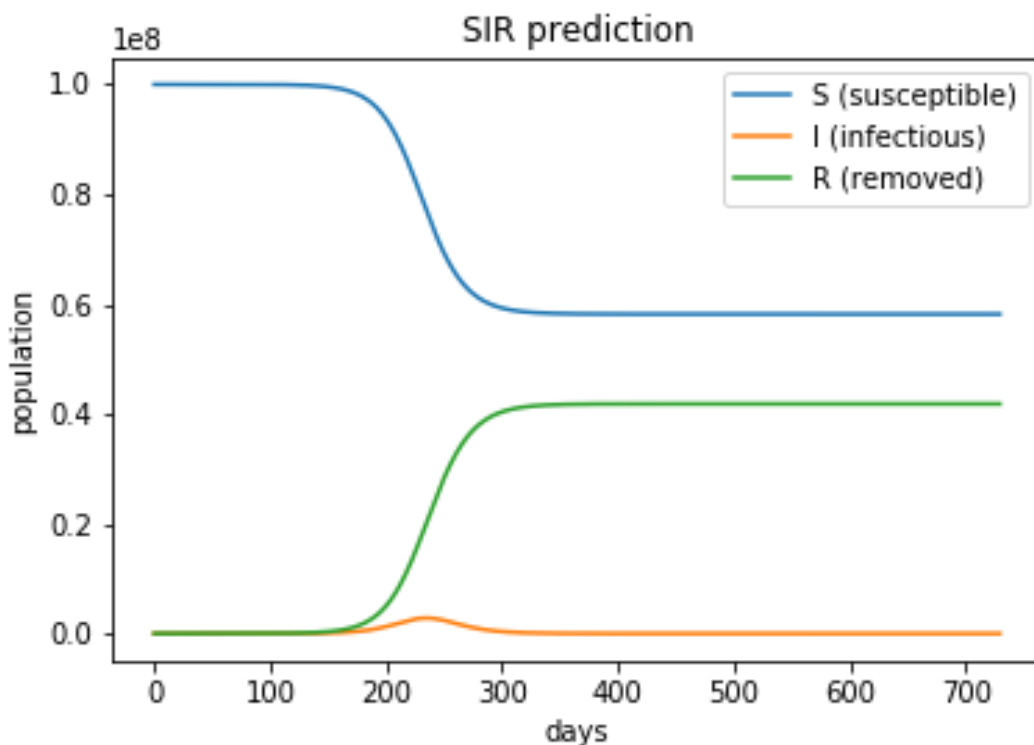


Figure 3: 自粛政策の下での感染状況。最初の分析に比べて横軸が2年分になっていることに注意。

- もっと自粛したらどうなるかやってみたんですが、 $\beta$  を 0.76 より小さくすると、感染が抑えられ過ぎて、回復/死亡者(R)が 41.2% に届かず、集団免疫が獲得できないという結果になりました。自粛しすぎると集団免疫も獲得できず、どこかで特效薬が開発されるまでは永遠に自粛を解除できない、という悲しい結果になりそうです。

❖ **活動自粛だけでは日本を救えない！！もう一**

## 歩、その先へ

- 上の分析で見たように、活動自粛だけでは、集団免疫を獲得するまでに15.5万人もの重篤者を覚悟しなければいけません。現在の医療体制では、その多くの命を救うことはできないでしょう。そこで、もう一方踏み込んでみます。[東京経済オンラインの荻原のサイト](#)のデータを見ると、日本では、2020年4月1日12:00までに亡くなった57名のうち、54名が70代以上で占められています。同様の傾向は世界共通に見られ、[Ferguson Report 9](#) では、30代までの感染者死亡率が0.1%未満であるのに対し、40代で0.15%、50代で0.6%、60代で2.2%、70代で5.1%、80代以上で9.3%と報告されています(英国/米国の場合)。これを利用しない手はないでしょう。
- そこで、ちょっと荒っぽいですが、感染死亡率が高い70代以上の高齢者を、徹底的に保護することを考えてみましょう。ざっくり2000万人のお年寄りと、それ以外の世代との交流を完全に隔絶する「**高齢輪中政策**」もしくは「**逆姥捨て山政策**」です(語彙)。「いや無茶だろそんな」と思うかもしれませんが、僕だって嫌ですよ、お婆ちゃんに会えなくなるのは。でも、それで守れる命があるかもしれないし、何より思考実験するだけなら誰にも迷惑はかかりません。とにかく、やるだけやってみましょう。
- 計算は簡単です。人口を**1億人**から**8000万人**に減らすだけ。1つ困ったことに、70代未満の人の重篤化率がよく分かりません。上述の荻原さんのデータだと、2020年4月1日12:00時点で70代未満の感染者1581名に対して重篤化したのは36人。そのまま計算すると重篤化率は2.28%です。でも、ご存知の日本は検査数が少ないので、無症状・軽症な"隠れ感染者"がもっと居るでしょう。たとえば、実際の感染者が検査陽性数の10倍だとしたら、重篤化率は0.228%になります。
- とりあえずは、[Ferguson Report 9](#) で報告されている世代別の「入院を必要とする比率」と「入院後に集中治療が必要な比率」の積のうち、70代未満の平均値である0.91%を使うことにします。米英のデータだということには注意が必要ですけど。
- 目指したい免疫保持者数はあくまで4200万人のままですから、78%レベルよりもっと活動してもらわないといけません。色々試したところ、**83%レベルの活動**(17%の活動削減)が下限で、これより活動を減らすと若齢者だけでは4200万人の免疫を獲得できません。
- 活動レベルを83%にした時の若齢者の感染状況の推移を示したのが下記のグラフです。4200万人が免疫を獲得するのは231日目。感染者がピークを迎えるのは173日目で、その時の若齢感染者数は387万人。そのうち0.91%が重篤化すると3.5万人。うわ、**微妙**!! もうちょっと重症者が少なくなることを予想してたんですが...

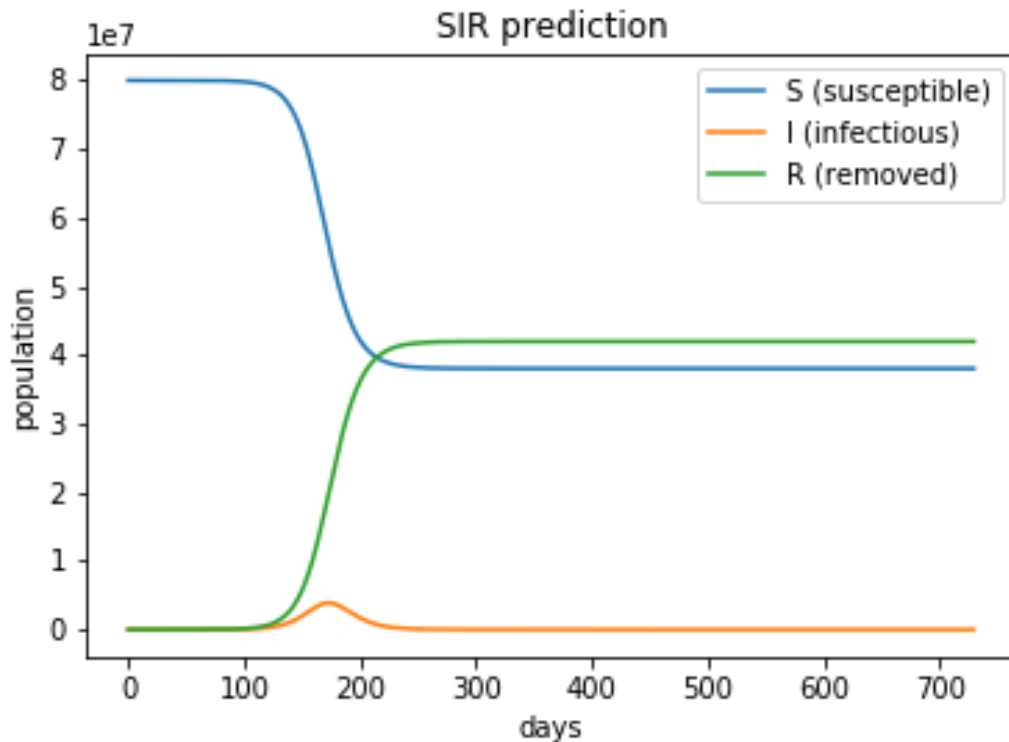


Figure 4: 高齢者隔離+緩い自粛政策の組み合わせ。

- 仕方ありません。多分、60代以下の重篤化率は0.91%より小さいはずですが。試しに計算してみると、重篤化率は50代以下で0.30%、40代以下で0.11%、40代未満では0.06%となりました。もっとも、この重篤化率は基本再生産数 $R_0$ が2を超えているような米英のデータであることには注意が必要です。ここで、無理やりですが、日本での重篤化率を仮に0.25%としてみましよう。このとき、**重篤者は1万人**。これならなんとか許容できそうです。
- ところで、83%の活動レベルまでの自粛(17%の活動削減)ってどのくらいなんでしょう。**6日に1回テレワーク**して外出を抑えると、だいたい83%の活動レベルになります。あれ？意外にいけそうじゃない？「6日に1回、テレワーク」って語呂もいいし。高齢者を完全に隔離するという荒っぽい政策との組み合わせではありますが、今よりもずっと緩い活動自粛で1年以内に集団免疫を獲得する可能性を示すことはできました。

【ご注意下さい！！】

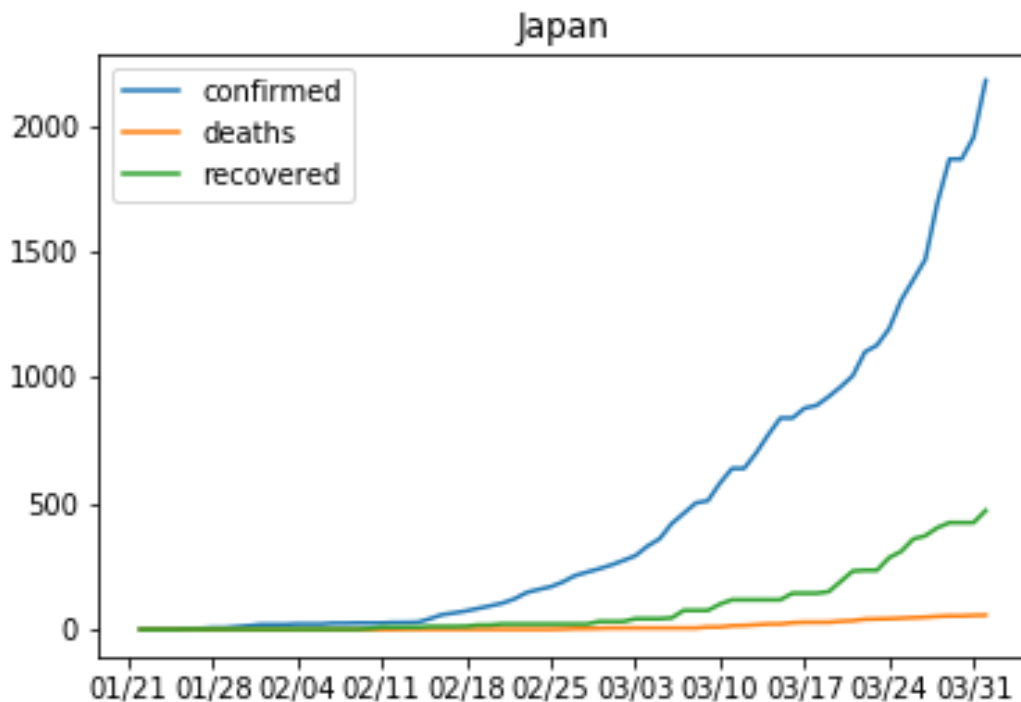
以上の分析は、あくまで、

- ・感染状況の変化がSIRモデルで適切に表現できる。
- ・SIRモデルのパラメータ  $\beta$  と  $\gamma$  が適切に推計できている。特に、 $R_0 = 1.7$  とするのが適切である。

という仮定に基づいています。素人の僕は1日で計算しましたが、世界中の人が、最新のデータを用いて、最新のモデルと最新の手法で、日々必死になって将来を予測し、CoVid-19と戦う手立てを考えまくっています。この文章は、そうした方々が一生懸命やっってることを、僕よりも素人な皆さんに伝えたいと思って書きました。

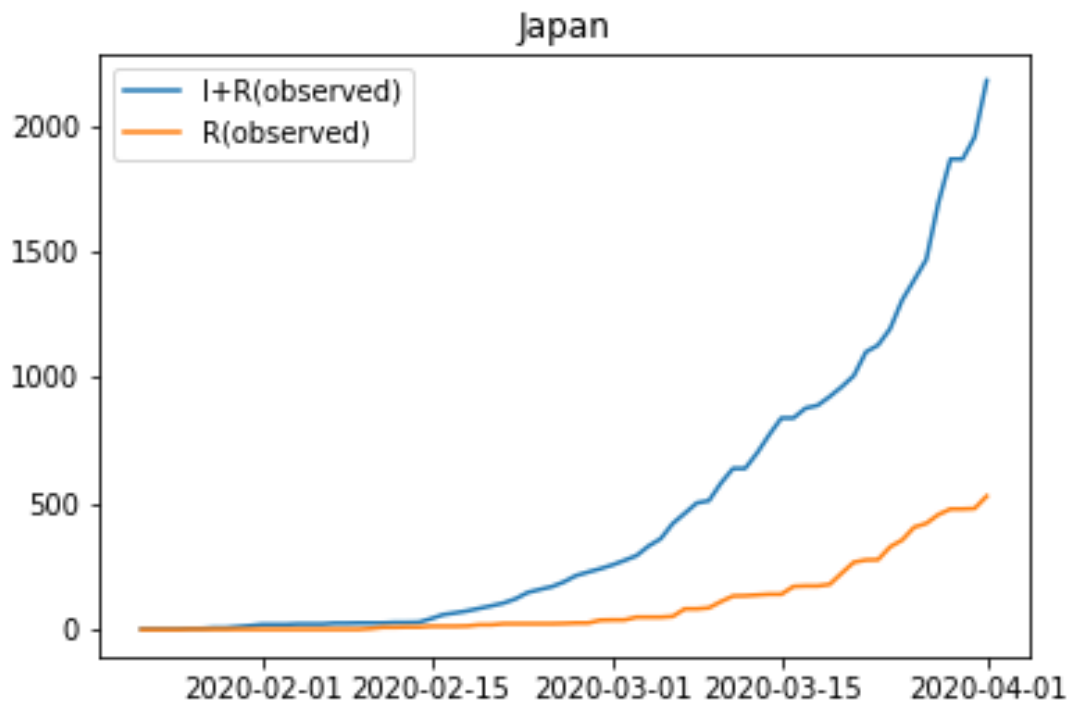
## ❖ 【おまけ】 SIRモデルの"お手軽"なパラメータ推計

- ここから先は、ver1.0 で使った  $\beta$  と  $\gamma$  をどうやって推計したかを書いておきます。少しだけ難しいので、読まなくても構いません。
- SIRモデルは、感染率  $\beta$  と回復/死亡率  $\gamma$  の2つのパラメータだけで感染状況の変化を記述できる、とてもシンプルなモデルです。実際の CoVid-19 は2週間程度の潜伏期間があり、年齢によって重篤化率が大きく変わることも知られています。あと、やっぱり感染者のすべてを、感染したと同時に検出できるわけじゃないし、その時々でその国がとった政策によっても変化しちゃうんですよね。ここで紹介する方法を使うと、どの国でも基本再生産数が  $R_0 > 3$  とかになり、もう世界中で医療崩壊みたいな感じになっちゃいます。
- さらに、現実を考えると、都心部と地方部、満員電車で通勤するサラリーマンと自宅でお子さんを抱えて家から出られない主婦とでは、一日に接触する機会も違います。PCR検査の精度や検査数の制限によって、感染が確定することないまま自宅療養している人も少なくないでしょう。SIRモデルは、そういう複雑で現実的な状況を正確に表すことはできません。僕が見たところ、SIRモデルをそのまま予測に使っている例はあまりなくて、少なくとも潜伏期間(Exposed)を入れて拡張したSEIRモデルを使っていることが多いようです。
- ただ、シンプルなだけに、観測されているデータからパラメータを推定することも以外に簡単です。僕は専門家ではないですが、HDX(*Human Data Exchange*)で[公開されているデータ](#)と、[Kai Sasaki 氏のブログ](#)で紹介されている方法を使って、**二乗平均平方根誤差(RMSE: root mean square error)**を最小化するようにパラメータを推計してみることができます。
- HDXで公開されている2020年4月1日までのデータを用いて、日本の感染が確認された者(confirmed)、死亡者(death)および回復者(recovered)の累積値をプロットしたものが以下の図です。



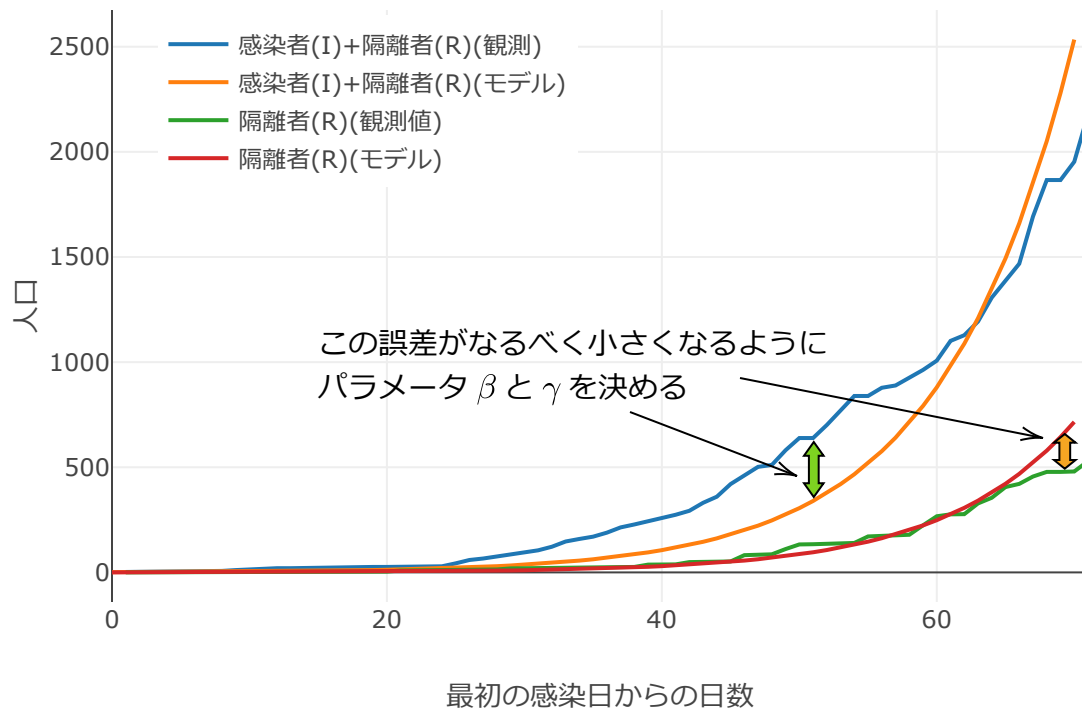
**Figure 5:** HDXによる日本の感染確認者数(confirmed)、死亡者数(death)および回復者数(recovered)。いずれも累積値。

- ?において、青の曲線で表される「確認された感染数(confirmed)」は、SIRモデルでは**感染者(I)**と**回復/死亡者(R)**の和に相当します。一方、SIRモデルでは回復者と死亡者はどちらも**回復/死亡者(R)**として取り扱われますから、?においてオレンジの曲線で表される「死亡者(death)」と緑の曲線で表される「回復者(recoverd)」の和が、SIRモデルでは**回復/死亡者(R)**に相当します。次の図のように、confirmedを「感染者(I)と回復/死亡者(R)の和(観測値)」、deathとrecoveredの和を「回復/死亡者(R) (観測値)」と表しておくといいかもしれません。



**Figure 6:** HDXのデータをSIRモデルにフィッティングさせるために調整。  
 I+R(observed)は感染者と回復/死亡者の和(観測値)、R(observed)は回復/死亡者(観測値)を表す。

- そして、SIRモデルで得られた**感染者(I)**、**回復/死亡者(R)**がこの観測値になるべく近くなるように、感染率  $\beta$  と回復/死亡率  $\gamma$  を求めます。具体的には、下の図のように、各日について観測値とモデルで得られる値との誤差を求め、その誤差を2乗したものの**平均の平方根**を取ったもの(2乗平均平方根誤差)を「離れ具合」の指標とし、これが最小になるように  $\beta$  と  $\gamma$  を決めてあげます。これも計算機を使うと、割と簡単にできます。



### Drawing 1: SIRモデルのフィッティングのイメージ

- ちょっと長いですが、計算のためのコードを書いておきますね。これも Python から実行できます。HDXから次の3つのファイルをダウンロードして、data という名前のディレクトリに入れてから実行して下さい。

```
import numpy as np
from scipy.optimize import minimize
import pandas as pd
from datetime import datetime

country_name = 'Japan' # ここを変えると他の国のSIRモデルのパラメータが推計できる

# data というディレクトリの中に、HDXからダウンロードした下記のファイルが置いてあることが前提
# 'data/time_series_covid19_confirmed_global.csv'
# 'data/time_series_covid19_deaths_global.csv'
# 'data/time_series_covid19_recovered_global.csv'

###
# 推計プログラムで呼び出す関数の定義
###
# データファイルから累積感染者数、累積死者数、累積回復者数を求める関数
def read_data(country_name='Japan'):
    # 国名から最初の感染者が報告された日付を抽出する
```

```

df = pd.read_csv('data/time_series_covid19_confirmed_global.csv')
df = df[df['Country/Region'] == country_name]
df = df[df['Province/State'].isna()]
cf = df.iloc[0, 4:] # 4列目以降が感染者数
start_date_str = cf[cf>0].first_valid_index() # 最初の感染者が報告された日
confirmed = df.iloc[0].loc[start_date_str:]
dates = [datetime.strptime(d, '%m/%d/%y') for d in confirmed.index]

# 死者数, 回復者数を対応するファイルから読み込む
def load(fname):
    df = pd.read_csv('data/time_series_covid19_%s_global.csv'%fname)
    df = df[df['Country/Region'] == country_name]
    df = df[df['Province/State'].isna()]
    return df.iloc[0].loc[start_date_str:]
deaths, recovered = load('deaths'), load('recovered')
#
return dates, confirmed.values, deaths.values, recovered.values
# SIRモデルを用いて将来のT日目までの予測を行う
# beta: 感染率
# gamma: 回復率/死亡率
# T: 分析期間(日)
# DT: 離散時点間隔(日) - 1日を4つの離散時点に分割するなら DT=0.25
# init: 初期状態(S, I, Rの初期値からなる3つ組)
def predict_SIR(beta, gamma, T, DT=1E-3, init = (1, -1E-6, 1E-6, 0.)):
    # SIRモデルの各状態変数の単位時間あたりの増分
    def SIR_diff(X):
        S, I, R = X
        N = sum(X)
        return(np.array([-beta*S*I / N, beta*S*I / N-gamma*I, gamma*I]))

    # 時点集合
    t_set = np.linspace(0, T, int(T/DT) + 1)
    # SIRの各状態を格納する行列
    X = np.zeros((len(t_set), 3), dtype='float')
    # 初期値から順に増分を加えることで状態変数を生成
    X[0] = init
    for tid, f in enumerate(t_set[:-1]):
        X[tid+1] = X[tid] + SIR_diff(X[tid])*DT
    return X, t_set
# パラメータvarの下で得られるSIRモデルの値と観測された値との二乗平均平方根誤差を求める
def rmse(var, C_obs, R_obs, DT, init):
    size = len(C_obs)
    beta, gamma = var
    X, t_set = predict_SIR(beta, gamma, size, DT, init)
    X = X[t_set >= 1]
    t_set = t_set[t_set>=1]
    S = X[t_set%1==0, 0]
    I = X[t_set%1==0, 1]
    R = X[t_set%1==0, 2]
    ret = np.sqrt( np.average((C_obs - I - R)**2) + np.average((R_obs - R)**2) )
    return ret

##
# 推計プログラム本体
##
dates_obs, Cfm, Dth, Rcv = read_data('Japan')
C_obs = Cfm
R_obs= Dth+Rcv

```



```

# RMSEが最小となる beta, gamma および基本再生産数 R0 を求める
N = 1e8
I0 = 1 # 初期時点での感染者数(初期感染者数を4にすると誤差が大きくなり, R0が1.16くらいになる)
DT = .01 # 離散時点の時間幅
init = (N-I0, I0, 0) # 初期解
# scipy.optimize.minimize 関数を用いて
opt = minimize(rmse,
               [0.48, 0.2],
               args=(C_obs, R_obs, DT, init),
               method='L-BFGS-B',
               bounds=[(0, 10), (0, 10)]
               )
beta, gamma = opt.x
R0 = beta/gamma
print("beta:%6.5f, gamma:%6.5f, R0:%6.5f, RMSE:%f" % (beta, gamma, R0, rmse((beta, gamma), C_obs, R_obs, DT,
init)))

```

- 2020年3月31日までのデータから推計された  $\beta$  と  $\gamma$  は、それぞれ、以下の通りです：

$$\beta = 0.12428$$

$$\gamma = 0.03427$$

$$R_0 = 3.62637$$

- 2021年4月1日までのデータを使うと、 $\beta$  と  $\gamma$  は、それぞれ、以下のように変わりました：

$$\beta = 0.14741$$

$$\gamma = 0.04164$$

$$R_0 = 3.54006$$