Previsão de Epidemia do Coronavírus

Sérgio Bastos
sergbastos@yahoo.com

30 de março de 2020

Introdução

O presente documento visa apresentar um software desenvolvido em Python para a elaboração da curva epidêmica do Coronavírus na população da cidade a partir dos dados históricos da evolução do número de casos confirmados, servindo de ferramenta para o planejamento de ações visando achatar a curva e diminuir o impacto do alastramento da doença no sistema de saúde da cidade.

O software é basicamente composto de duas partes: a elaboração da curva epidêmica através da resolução de três equações diferenciais e o ajuste de uma curva de tendência exponencial sobre os dados históricos dos casos confirmados. Uma vez conhecida a curva de tendência é possível ajustar os parâmetros da curva epidêmica para que ambas coincidam, permitindo com relativa precisão prever o acréscimo de casos, a data em que ocorrerá o pico mais grave e a quantidade de pessoas infectadas.

A curva epidêmica

O modelo matemático que descreve a disseminação de uma doença em uma população é conhecido como S-I-R, que divide a população em três "compartimentos" que podem variar em função do tempo, t:

- **S(t)** são aqueles suscetíveis mas ainda não infectados com a doença;
- **I(t)** é o número de indivíduos infectados;
- **R(t)** são aqueles indivíduos que se recuperaram da doença e agora estão imunes à ela.

As curvas que descrevem o comportamento dos três compartimentos no modelo S-I-R são elaboradas a partir da solução das seguintes equações diferenciais, derivadas pela primeira vez por Kermack e McKendrick [1] em 1927:

$$\frac{dS}{dt} = -\frac{\beta SI}{N} \tag{1}$$

$$\frac{dI}{dt} = \frac{\beta SI}{N} - \gamma I \tag{2}$$

$$\frac{dR}{dt} = \gamma I \tag{3}$$

Este modelo descreve as alterações de cada um dos compartimentos em função do tempo a partir de dois parâmetros: β e γ. A solução das equações apresentadas acima e sua apresentação em função dos parâmetros β e γ, utilizando-se da linguagem Python foi apresentada por Christian Hill [3] e é a base para a solução aqui apresentada.

O parâmetro γ representa o inverso do período infeccioso, aqui considerado 2 dias para o Coronavírus, usando como referência o pré-artigo de Liangrong Peng [4], portanto $\gamma = 0,5$.

O parâmetro β descreve a efetiva taxa de contato da doença: um indivíduo infectado entra em contato com βN outros indivíduos por unidade de tempo. Ele é o responsável pelo formato da curva em termos de amplitude e período de tempo, ou seja, o "achatamento" da curva em função das ações preventivas e de cuidados para evitar o alastramento da doença é por ele determinado. Quanto menor o β, mais "achatada" será a curva, durando mais tempo e com menor amplitude (menor o número de doentes que precisam de cuidados médicos ao mesmo tempo).

A ideia do software aqui apresentado é parametrizar o β a partir da tendência exponencial apresentada em função dos dados históricos de casos já confirmados, como apresentado na sequência.

Outros parâmetros utilizados, dentre eles R_0, estão em conformidade com o recente artigo da revista The Lancet [2].

A evolução dos dados históricos

O número de casos confirmados da doença a cada dia forma uma série temporal de tendência exponencial. Essa curva exponencial tem o formato:

$$X(t) = X(0)e^{\lambda t} \tag{4}$$

A curva epidêmica, no seu início, também possui o formato exponencial caracterizado pelo seguinte formato:

$$I(t) = I(0)e^{(\beta-\gamma)t} \tag{5}$$

De onde podemos concluir que a relação entre as duas curvas pode ser descrita por:

$$\lambda = \beta - \gamma \tag{6}$$

Uma vez conhecida a relação entre a curva exponencial de tendência ajustada sobre os dados diários do total de casos confirmados, elaboramos o software de acordo com a seguinte sequência:

- A partir da série temporal histórica dos dados diários de casos confirmados (totais), determinar a equação da curva exponencial de tendência;
- Uma vez tendo a curva exponencial de tendência, determinar a partir da sua taxa de crescimento exponencial (λ), o valor de β, sabendo que o valor de γ, para o COVID-19 conforme estudos [4] é 0,5.
- Resolver as equações diferenciais que caracterizam a modelagem S-I-R, tendo como valores iniciais I_0 igual ao valor X_0 da curva de tendência exponencial e considerando:

$$R_0 = \frac{\beta}{\gamma} \tag{7}$$

Com base nisso temos definida a nossa curva epidêmica conforme a modelagem S-I-R [1].

O software como ferramenta de planejamento

Conhecida a curva epidêmica baseada nos valores diários do total de casos confirmados, temos uma previsão do número de pessoas que estarão infectadas e, assumindo que um percentual dessas pessoas irá necessitar de tratamento no sistema de Saúde da cidade, confrontando com o número de leitos disponíveis, saberemos se estes serão suficientes e quando ocorrerão as maiores demandas. A cada dia que passa, ao atualizarmos os dados históricos, a curva epidêmica será atualizada, ficando mais ou menos "achatada"refletindo assim as consequências das ações que foram tomadas para diminuir a velocidade de avanço da epidemia, servindo como uma ferramenta de planejamento dos recursos necessários, bem como para a avaliação das medidas preventivas adotadas.

A Figura 1 mostra a curva epidêmica S-I-R. A Figura 2 mostra somente a curva de Infectados frente à oferta de leitos disponíveis no sistema de Saúde. Sabendo que uma parcela das pessoas infectadas necessitará de cuidados especiais ocupando parte dos leitos disponíveis, é possível fazer uma avaliação da quantidade de recursos (leitos, respiradores, etc...) necessários. A escolha entre as opções de gráficos é feita apenas controlando o comentário (#) das linhas de plotagem. A escala do gráfico resultante é ajustada automaticamente. A Figura 3 mostra o ajuste da curva exponencial em cima dos dados históricos de casos confirmados.

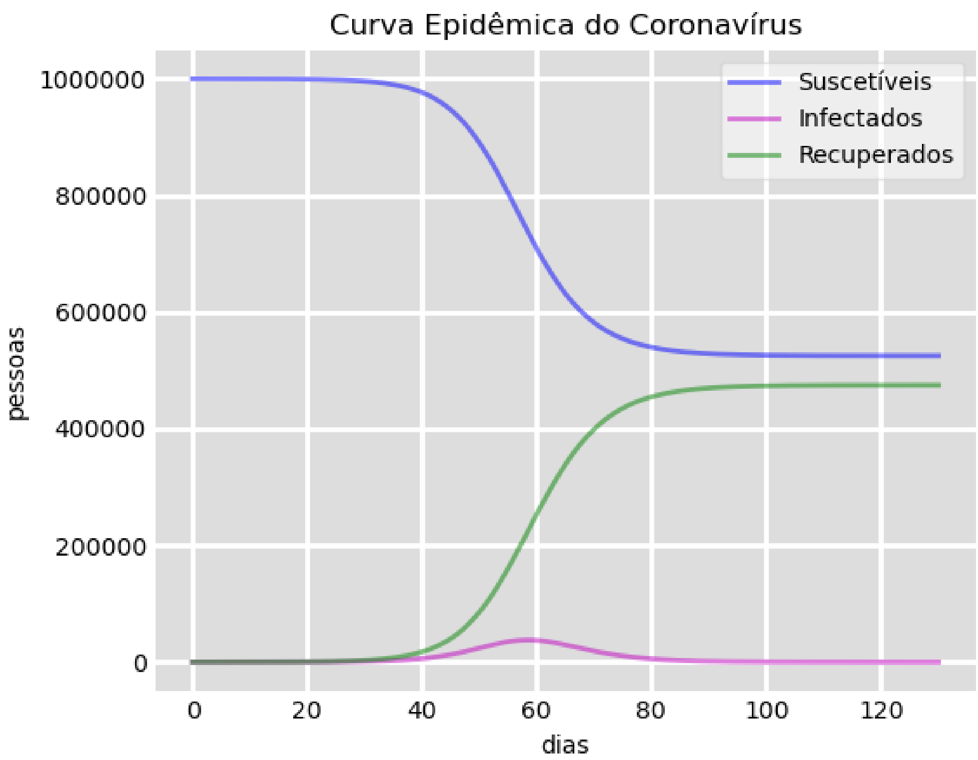

Figura 1: Curva Epidêmica S-I-R

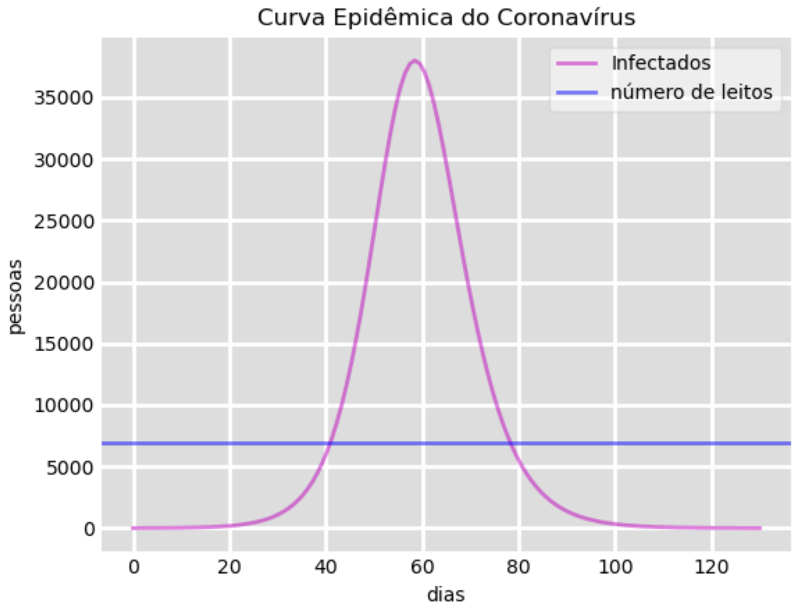

Figura 2: Curva de Infectados vs Leitos Disponíveis

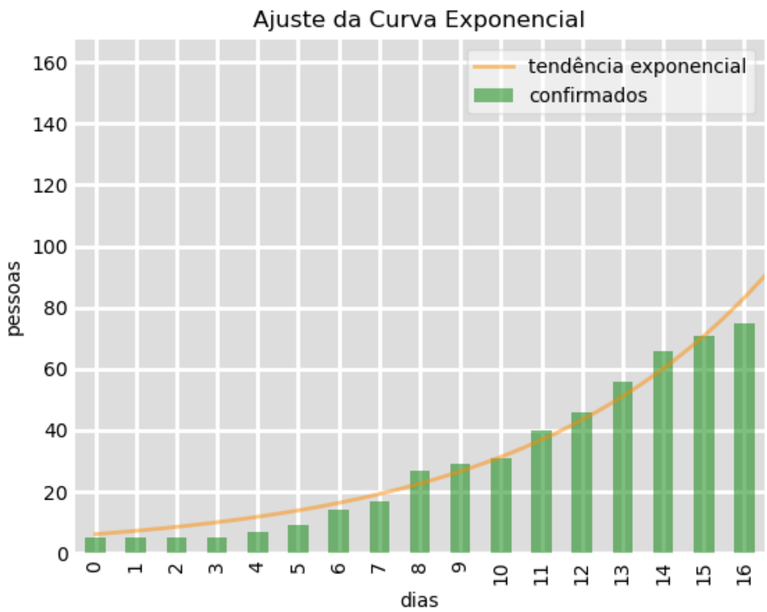

Figura 3: Ajuste da Curva Exponencial sobre os Dados Históricos

O software

Listing 1: Software para previsão da epidemia do Coronavírus

```
# -*- coding: latin-1 -*-
import numpy as np
import scipy
from scipy.integrate import odeint
import matplotlib.pyplot as plt
import pandas as pd

# ***************************************************************************
# * Código para estimativa de propagação do Coronavírus
# * Autor: Sérgio Bastos    28/03/2020
# ***************************************************************************

# dados históricos
hist = pd.Series([5, 5, 5, 5, 7, 9, 14, 17, 27, 29, 31, 40, 46, 56, 66, 71])
inicio = '12/03/2020'

# População total N, gamma
N = 1000000
gamma = 0.5

popt_exponential, pcov_exponential = \
    scipy.optimize.curve_fit(lambda t, a, b: a * np.exp(b * t),
                             hist.index, hist, p0=(5, 0.3))

# Parâmetros para o modelo S-I-R
I0 = popt_exponential[0]
beta = gamma + popt_exponential[1]
R0 = beta / gamma
S0 = N - I0 - R0

t = np.linspace(0, 130, 130)


# Equações diferenciais do modelo S-I-R.
def deriv(y, t, N, beta, gamma):
    S, I, R = y
    dSdt = -beta * S * I / N
    dIdt = beta * S * I / N - gamma * I
    dRdt = gamma * I
    return dSdt, dIdt, dRdt


# Vetor de condições iniciais
y0 = S0, I0, R0
# Integra as equações S-I-R ao longo do tempo, t.
ret = odeint(deriv, y0, t, args=(N, beta, gamma))
S, I, R = ret.T

# Plota os dados
fig = plt.figure(facecolor='w')
ax = fig.add_subplot(111, facecolor='#dddddd', axisbelow=True)

# As 3 curvas S(t), I(t) e R(t)
ax.plot(t, S, 'b', alpha=0.5, lw=2, label='Suscetíveis')
ax.plot(t, I, 'm', alpha=0.5, lw=2, label='Infectados')
ax.plot(t, R, 'g', alpha=0.5, lw=2, label='Recuperados')
```

```
# plt.axhline(y=7000, color='b', alpha=0.5, lw=2, label='número de leitos')

ax.set_xlabel('dias')
ax.set_ylabel('pessoas')
plt.title('Curva Epidêmica do Coronavírus')
ax.yaxis.set_tick_params(length=0)
ax.xaxis.set_tick_params(length=0)
plt.ticklabel_format(axis="y", style="plain")
ax.grid(b=True, which='major', c='w', lw=2, ls='-')
legend = ax.legend()
legend.get_frame().set_alpha(0.5)
for spine in ('top', 'right', 'bottom', 'left'):
    ax.spines[spine].set_visible(False)
plt.show()
```

Referências

[1] W. O. Kermack e A. G. McKendrick, editor. *A Contribution to The Mathematical Theory of Epidemics*, volume 115 of *A*. Royal Society of London, August 1927.

[2] Joel Hellewell, Sam Abbott, Amy Gimma, Nikos I Bosse, Christopher I Jarvis, Timothy W Russell, James D Munday, Adam J Kucharski, W John Edmunds, Sebastian Funk, and Rosalind M Eggo. Feasibility of controlling covid-19 outbreaks by isolation of cases and contacts. *The Lancet*, 8:488–496, April 2020.

[3] Christian Hill. *Learning Scientific Programming with Python*. Cambridge University Press, 2016.

[4] Liangrong Peng, Wuyue Yang, Dongyan Zhang, Changjing Zhuge, and Liu Hong. Epidemic analysis of covid-19 in china by dynamical modeling. *https://www.medrxiv.org/content/10.1101/2020.02.16.20023465v1.full.pdf*, 2020.