



# Modelo SIR aplicado ao Covid-19

Projeto 1 – Cálculo Numérico  
Grupo 2

# Introdução

- Epidemias e pandemias ao longo da história
- SARS – Cov – 2
- Modelagem matemática
- Modelo SIR





# Cidade de Curitiba

- Primeiro caso: 11 de março
- Protocolo de Responsabilidade Sanitária e Social
- Cores e características das bandeiras

## PROTOCOLO DE RESPONSABILIDADE SANITÁRIA E SOCIAL



### Alerta

Iniciamos com a bandeira amarela que é sinal de alerta constante devido à pandemia e não termos segurança e situação de normalidade. Todos devem estar o tempo todo em alerta e adotando as medidas de precaução largamente anunciadas e orientadas e cumprir todas as orientações do protocolo de responsabilidade sanitária e social.

Saiba mais

Situação atual



### Risco moderado

Segundo nível de alerta onde haverá restrições a funcionamento de serviços e do comércio e áreas onde propiciam a aglomeração de pessoas.

Saiba mais



### Alto risco

Nível de alerta total (bandeira vermelha) onde haverá a restrição da circulação de pessoas, permitindo o funcionamento apenas dos serviços essenciais.

Saiba mais

<https://coronavirus.curitiba.pr.gov.br/>

# Objetivos

- Interferência da bandeira vigente
- Análise pelo modelo SIR
- Evidência da necessidade de restrições



# Modelo SIR

S = suscetíveis

I = infectados

R = recuperados

N = total de habitantes

$$\begin{cases} \frac{dS}{dt} = -\frac{\beta \cdot I \cdot S}{N} \\ \frac{dI}{dt} = \frac{\beta \cdot I \cdot S}{N} - \gamma \cdot I \\ \frac{dR}{dt} = \gamma \cdot I \end{cases}$$

- Com  $\beta$  sendo a taxa de crescimento da doença
- Com  $\gamma$  sendo a taxa de recuperação



# Método de Euler

**Problema de Valor Inicial**

$$\begin{cases} y' = f(x, y(x)) \\ y(x_0) = y_0 \end{cases}$$

**Método de Euler**

$$y_{n+1} = y_n + h \cdot f(x_n, y_n)$$

# Método de Runge Kutta

**Problema de Valor Inicial**

$$\begin{cases} y' = f(x, y(x)) \\ y(x_0) = y_0 \end{cases}$$

**Método de Runge Kutta**

$$\begin{cases} y_{n+1} = y_n + \frac{h}{6} \cdot (k_1 + 2k_2 + 2k_3 + k_4) \\ x_{n+1} = x_n + h \end{cases}$$

**onde:**

$$k_1 = f(x_n, y_n)$$

$$k_2 = f\left(x_n + \frac{h}{2}, y_n + \frac{h}{2} \cdot k_1\right)$$

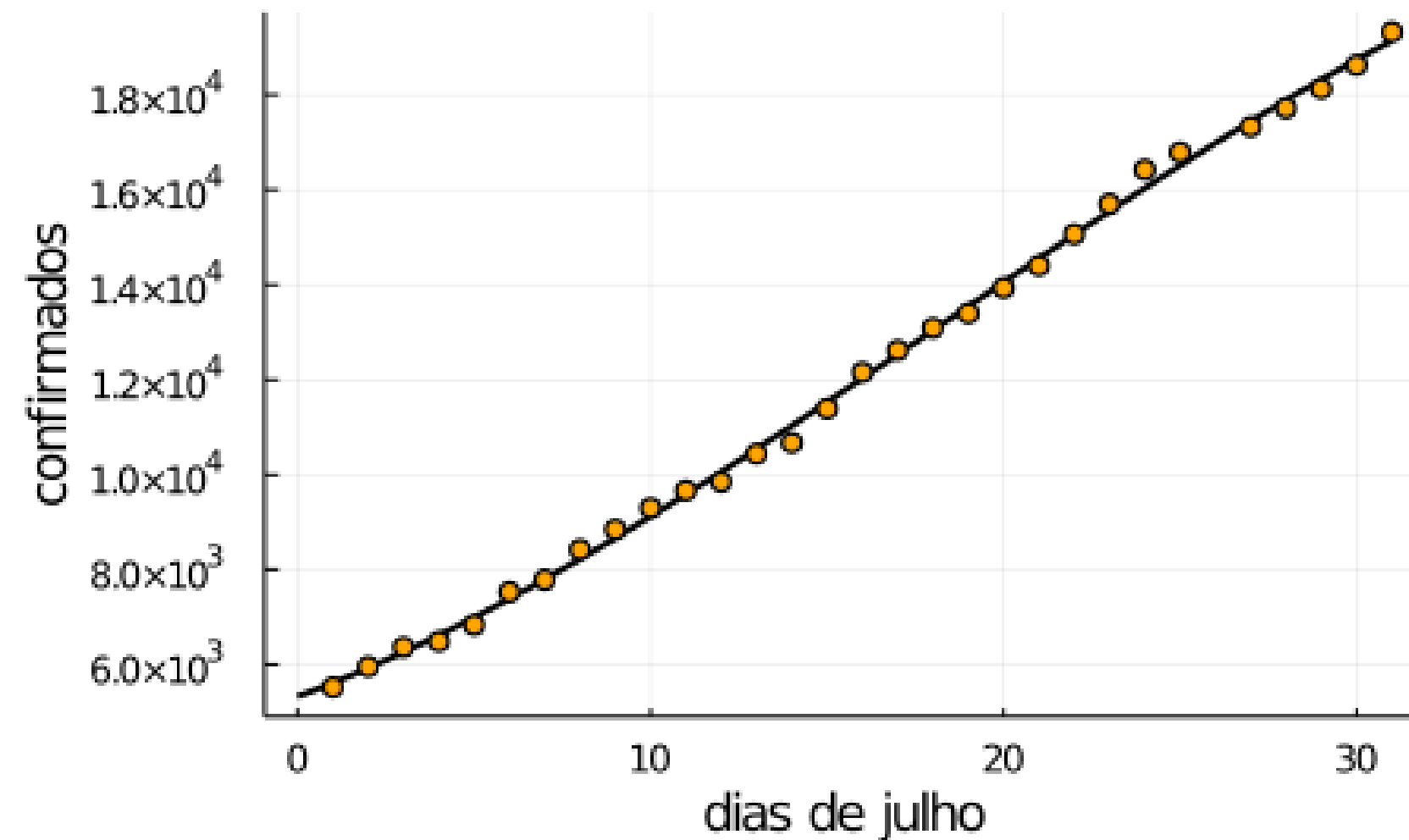
$$k_3 = f\left(x_n + \frac{h}{2}, y_n + \frac{h}{2} \cdot k_2\right)$$

$$k_4 = f(x_n + h, y_n + h \cdot k_3)$$

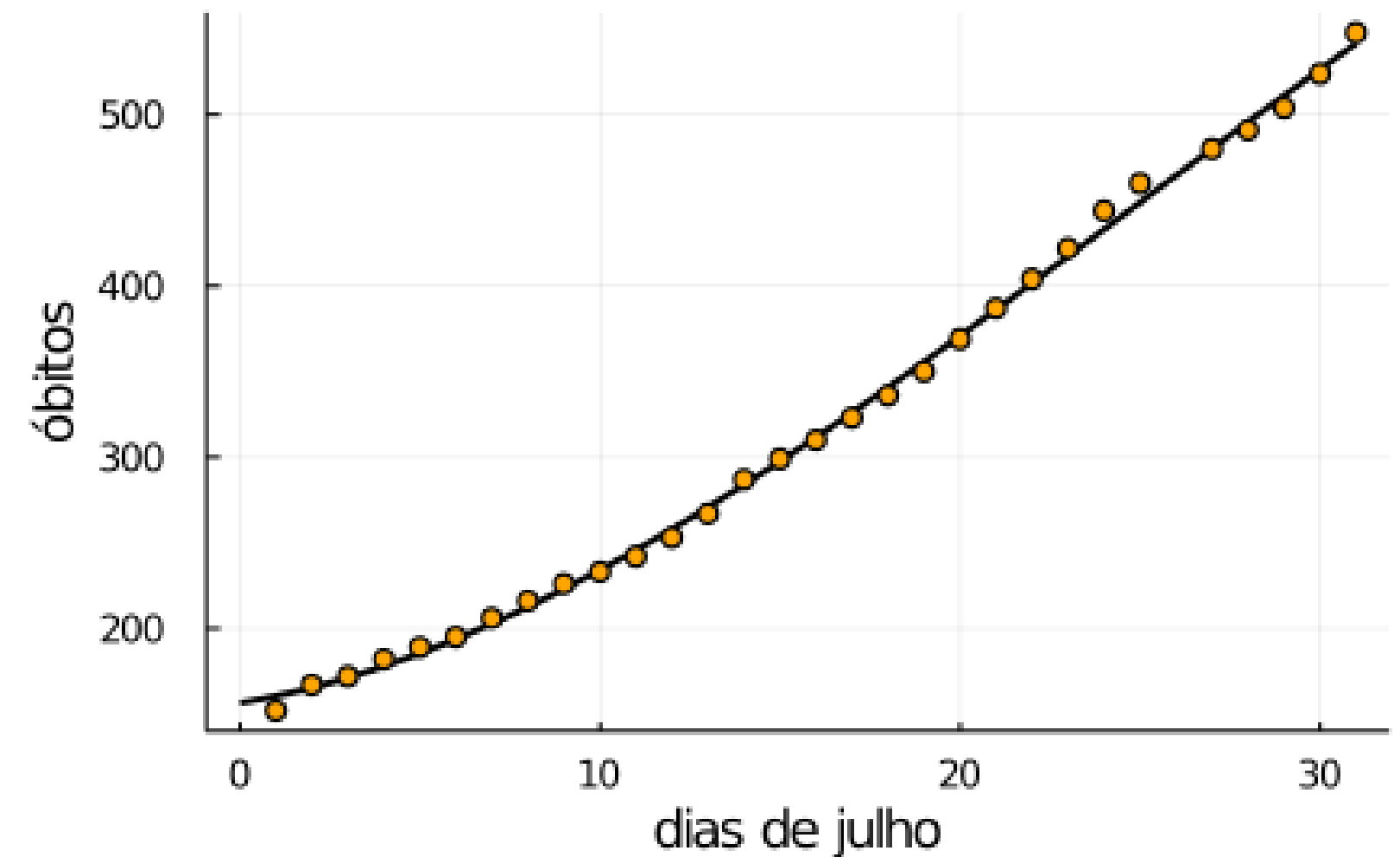


# Estudo de Julho

Casos confirmados até julho

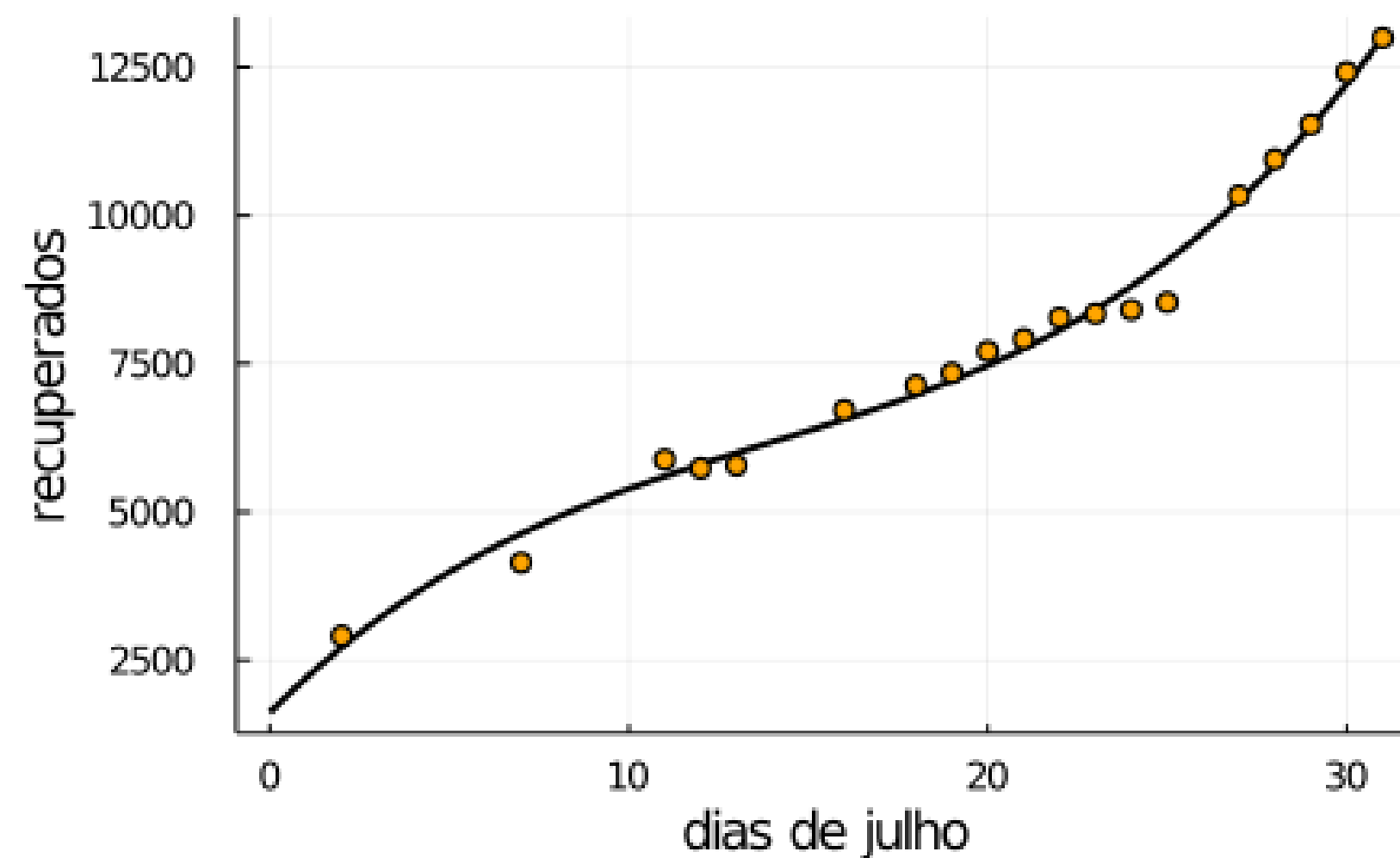


Óbitos até julho

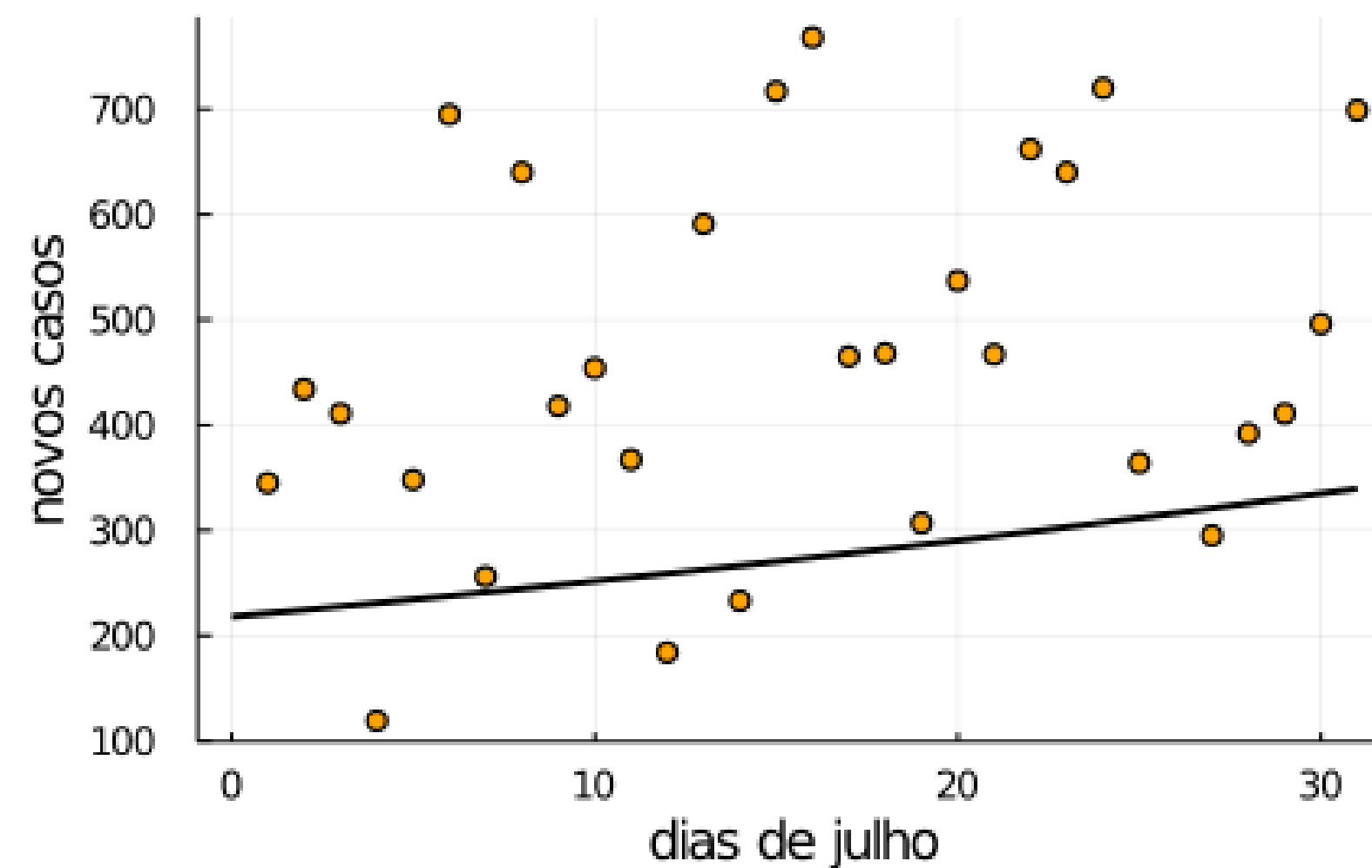


# Estudo de Julho

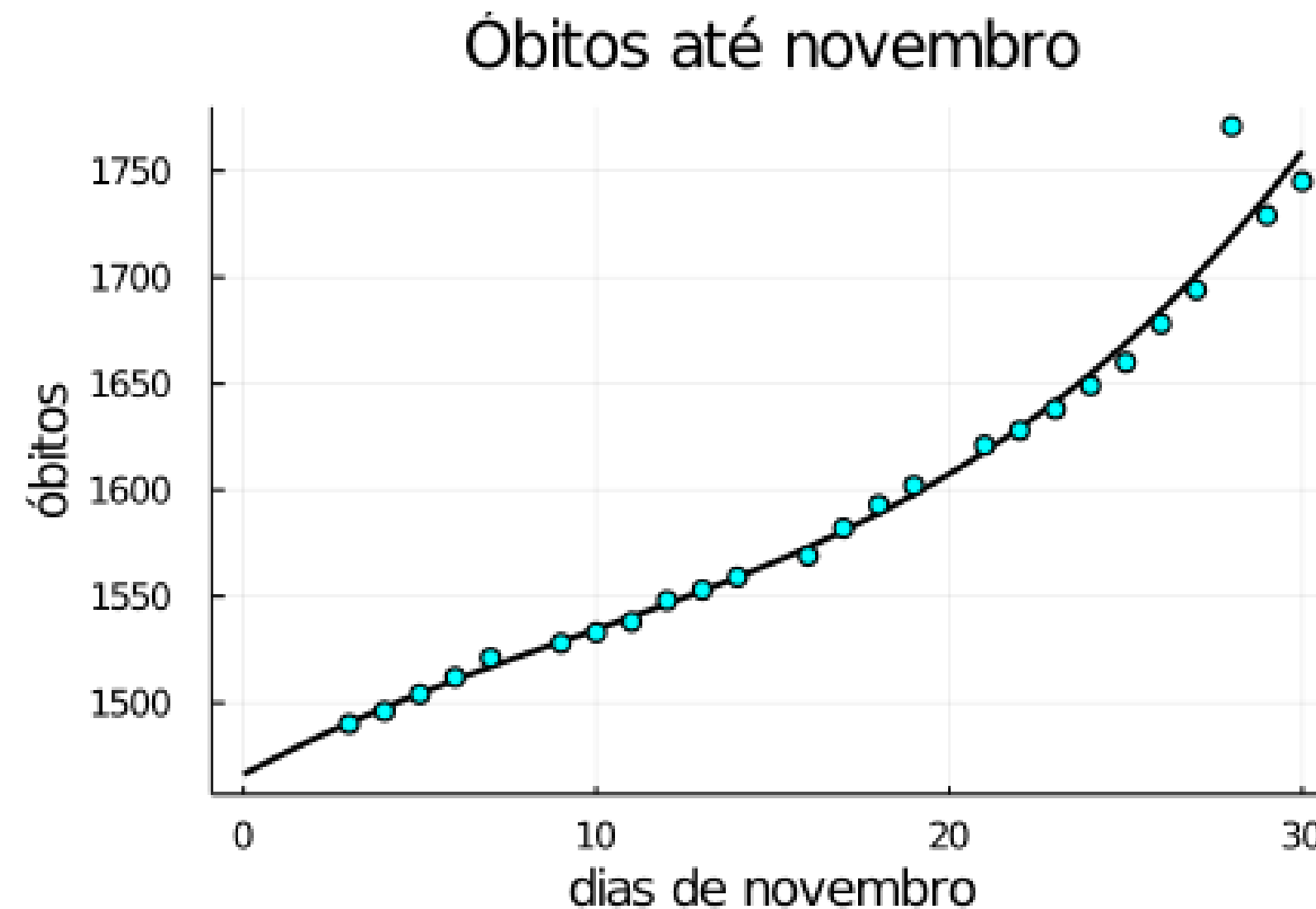
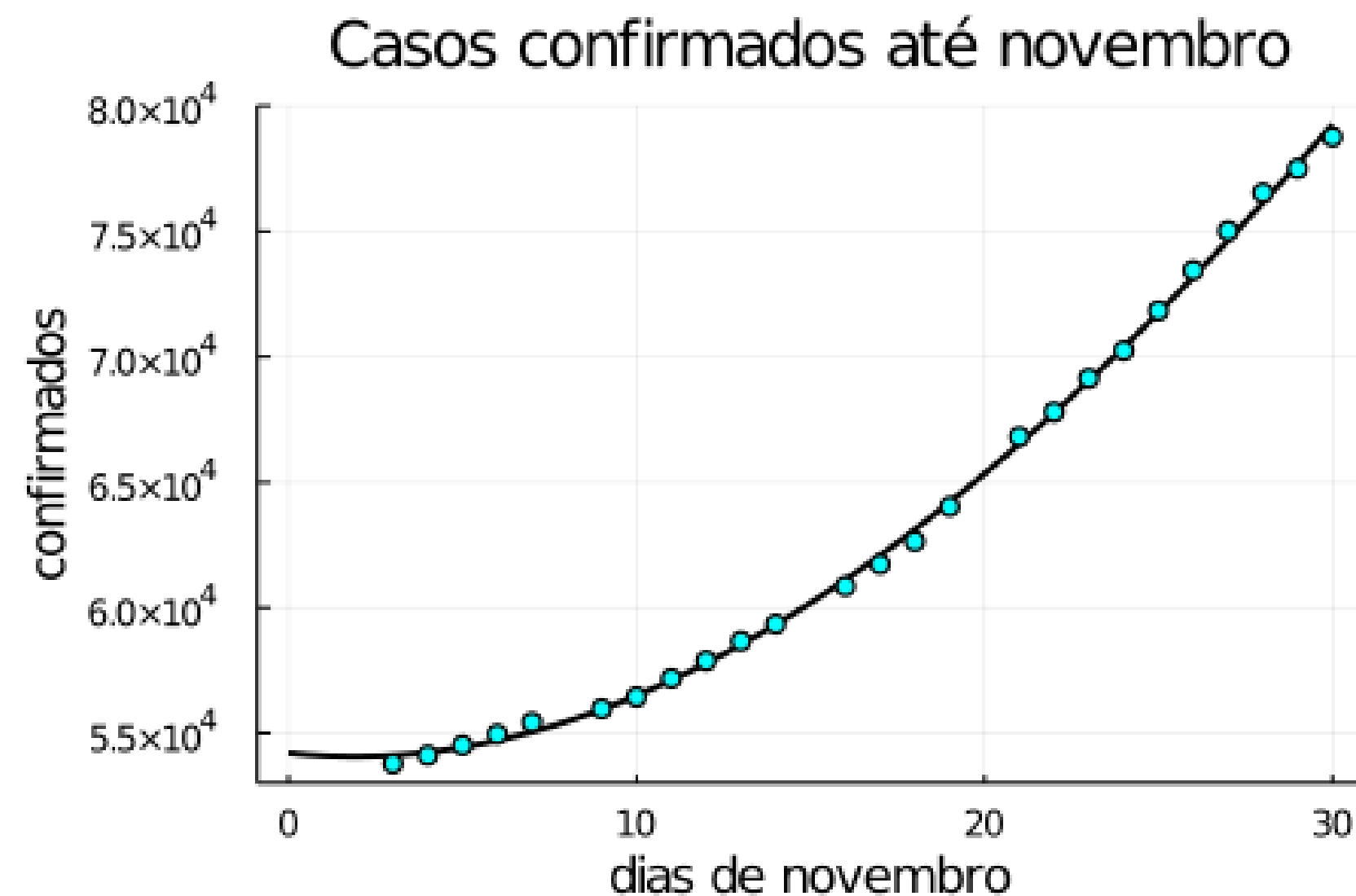
Recuperados até julho



Novos casos em julho

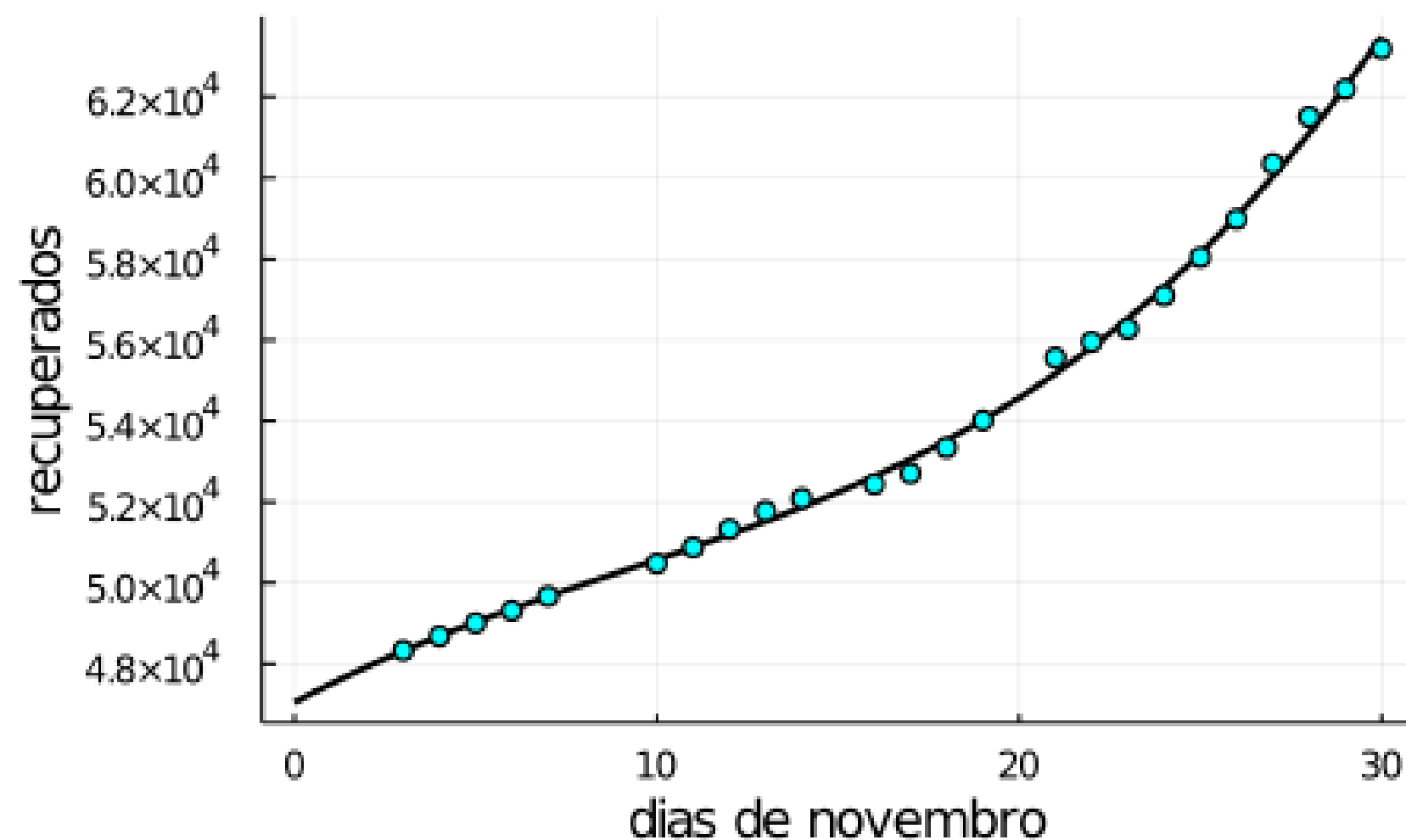


# Estudo de Novembro

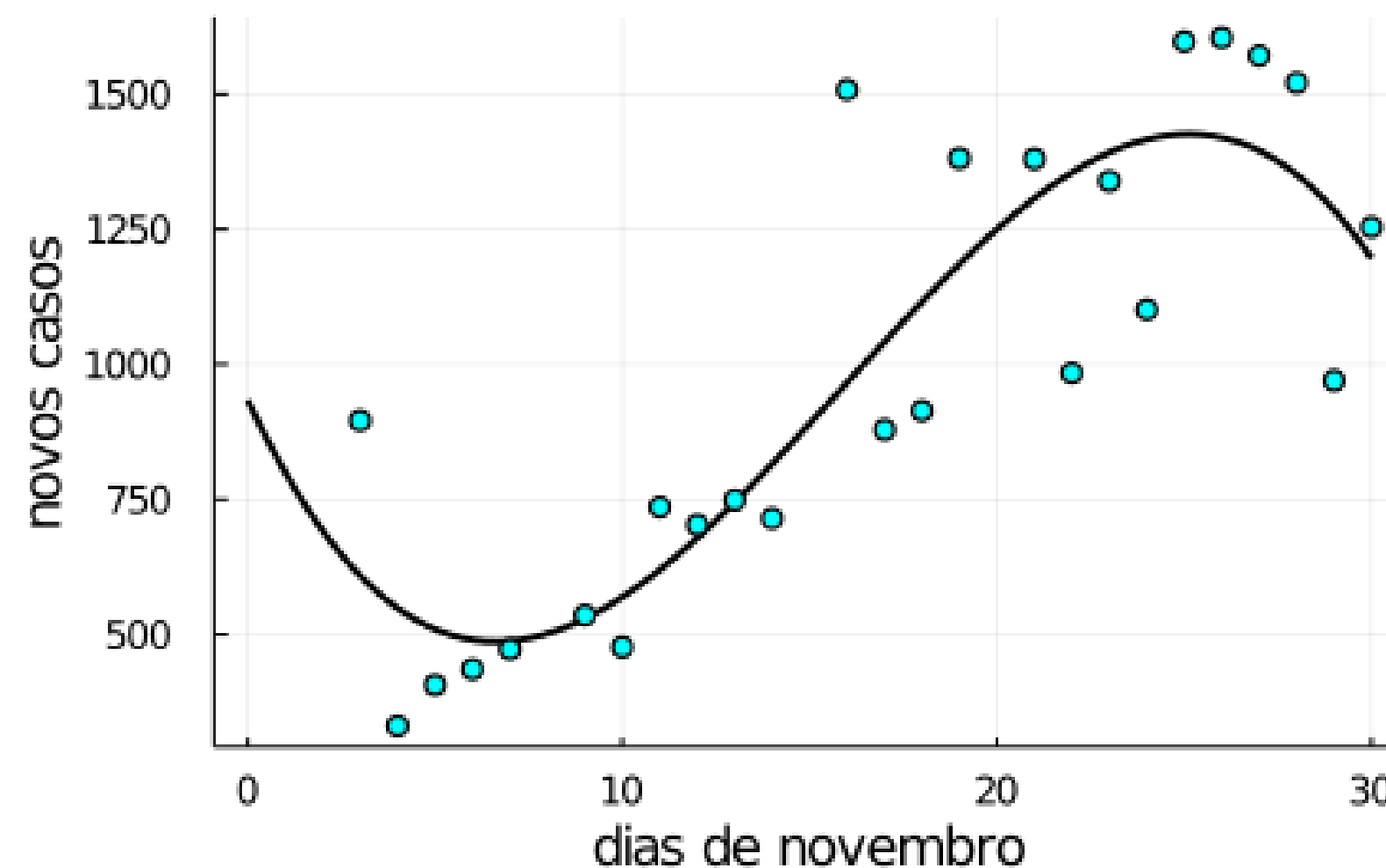


# Estudo de Novembro

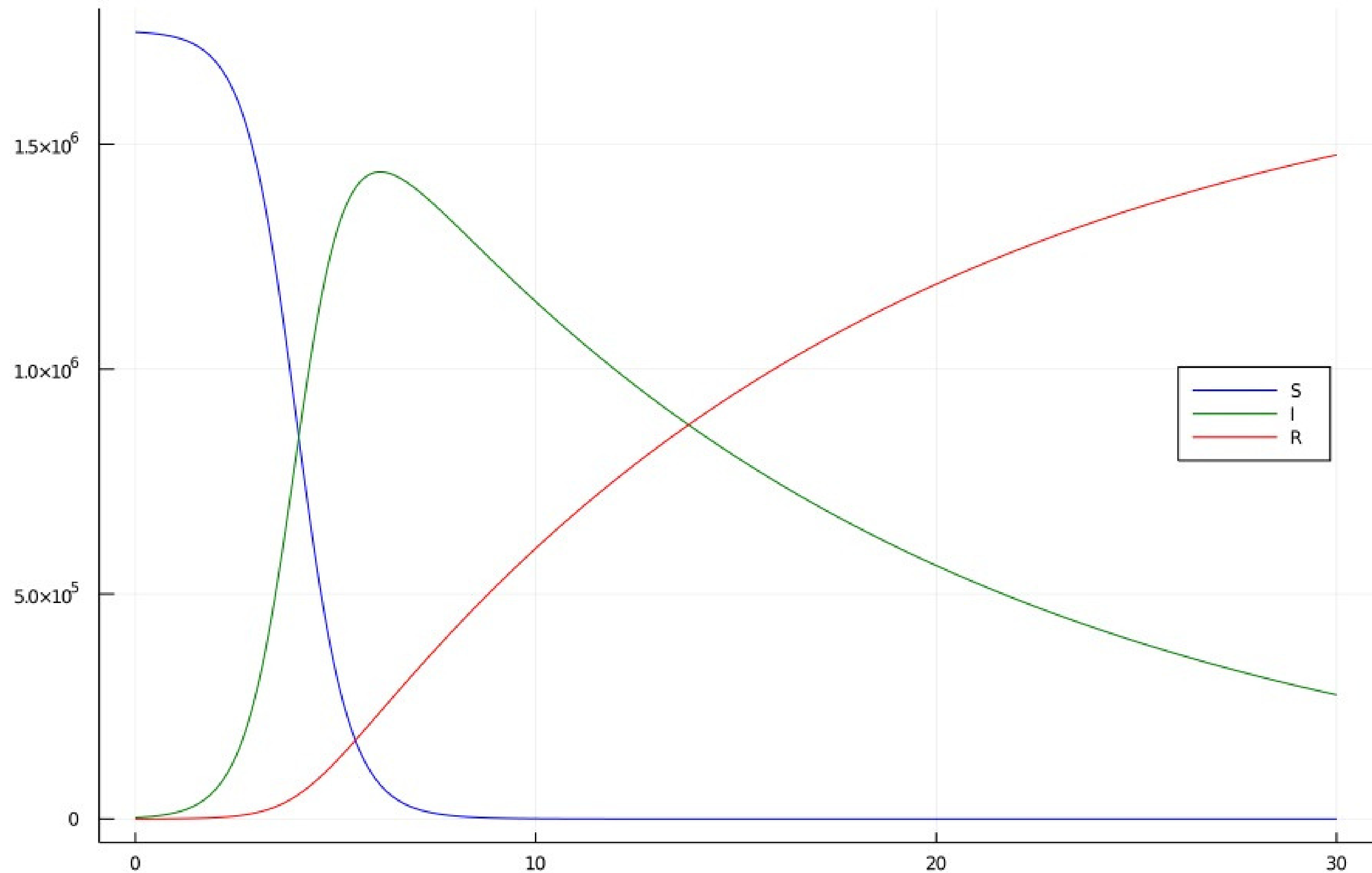
Recuperados até novembro



Novos casos em novembro

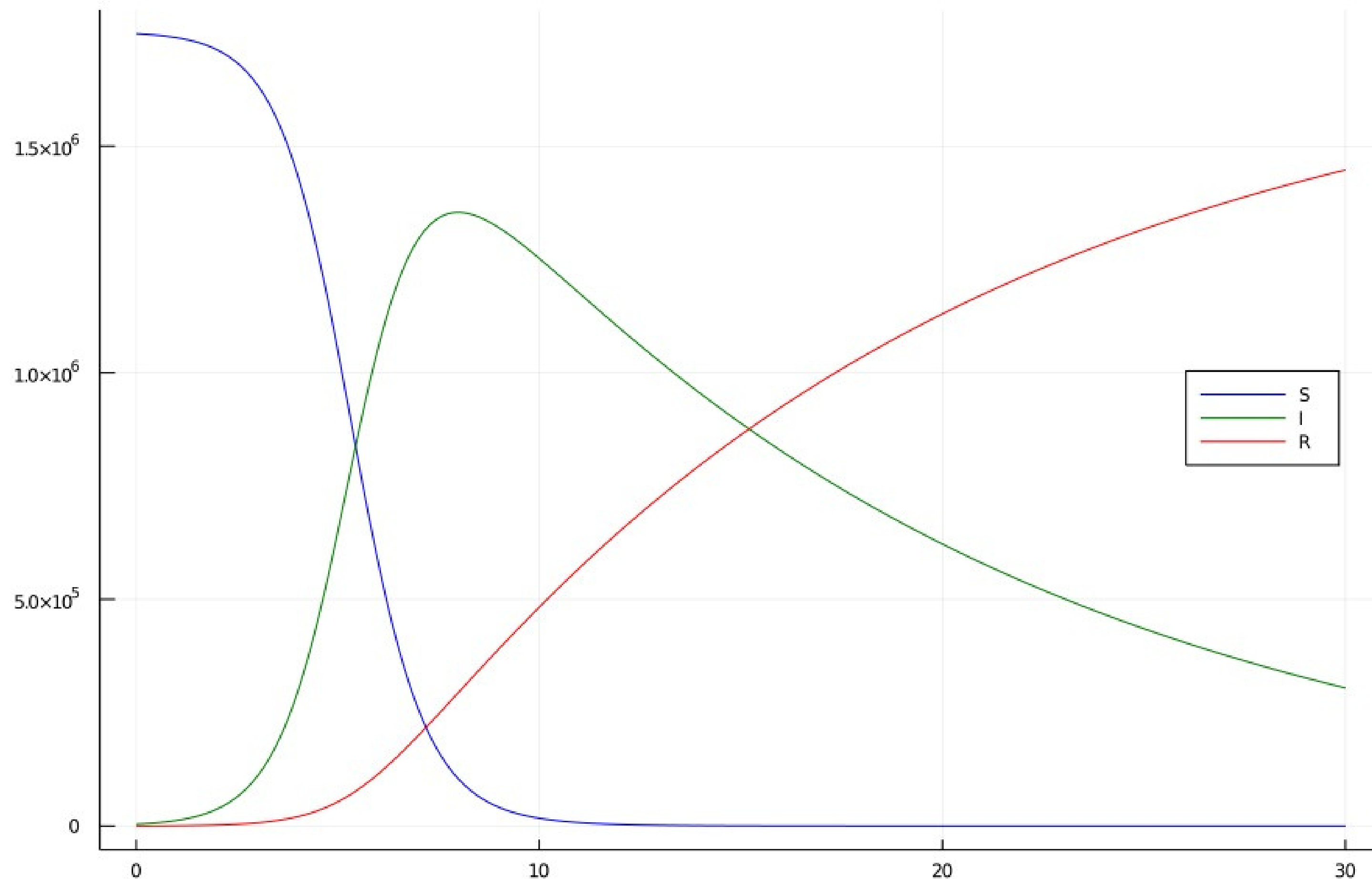


Modelo SIR - Julho



$\beta = 1,65$   
 $\gamma = 1/14$   
 $y_0 = 2881$

Modelo SIR Novembro



$$\beta = 1,20$$

$$\gamma = 1/14$$

$$y_0 = 3965$$

# Código

```
function metodo_de_euler(x0, y0 :: Vector, f, xN, N)
    h = (xN - x0) / N
    m = length(y0)
    y = zeros(m, N + 1)
    y[:,1] = y0
    x = range(x0, xN, length = N + 1)
    for k = 1:N
        y[:,k+1] = y[:,k] + h * f(x[k], y[:,k])
    end
    return x, y
end
```

```
function metodo_runge_kutta(x0, y0 :: Vector, f, xN, N)
    h = (xN - x0) / N
    m = length(y0)
    y = zeros(m, N + 1)
    y[:,1] = y0
    x = range(x0, xN, length=N+1)
    for k = 1:N
        k1 = f(x[k], y[:,k])
        k2 = f(x[k] + 0.5, y[:,k] + h * k1 / 2)
        k3 = f(x[k] + 0.5, y[:,k] + h * k2 / 2)
        k4 = f(x[k+1], y[:,k] + h * k3)
        y[:,k+1] = y[:,k] + h * (k1 + 2k2 + 2k3 + k4) / 6
    end
    return x, y
end
```

```

function sir_novembro(i, metodo)
    # i é o tempo de infecção. Ex.: i=14
    if metodo != :euler && metodo != :runge
        error("insira ':runge' ou ':euler'")
    end

    T = 1_752_000 #Total de habitantes na cidade de Curitiba
    γ = 1/i #Taxa de recuperação
    β = 1.20 #Taxa de crescimento

    f(x, y) = [-β / T * y[2] * y[1];
               β / T * y[2] * y[1] - γ * y[2];
               γ * y[2]]

    x0 = 0.0 #Onde se inicia, dado pelo tempo
    y0 = [T - 3965.0; 3965.0; 0.0] #Onde se inicia, número dos casos
    xf = 30.0 #Onde termina, ou seja, o tempo final
    N = 1000 #Quantos passos a função dará

```



```
if metodo == :euler
    x, y = metodo_de_euler(x0, y0, f, xf, N)
else
    x, y = metodo_runge_kutta(x0, y0, f, xf, N)
end

#Plotando os gráficos de Suscetíveis, Infectados e Recuperados
p = plot(x, y[1,:], c=:blue, lab="S", legend=:right, title= "Modelo SIR - Julho")
plot!(x, y[2,:], c=:green, lab="I")
plot!(x, y[3,:], c=:red, lab="R")

# Salvando os gráficos
png(p, "gráficos_modelo_sir_julho")

end
```

```

dias_julho2 = [2, 7, 11, 12, 13, 16, 18, 19, 20, 21, 22, 23, 24, 25, 27, 28, 29, 30, 31]
recuperados_julho = [2909, 4145, 5883, 5735, 5784, 6717, 7134, 7343, 7710, 7915, 8275, 8341, 8407, 8533, 10334, 10944, 11533, 12413, 12991]

f(x) = -0.2321 * x^3 + 12.6913 * x^2 + 275.1272 * x + 5329.4509 #Aproximação de função - Casos confirmados até julho
F(x) = -0.0067 * x^3 + 0.4968 * x^2 + 3.4556 * x + 156.5620 #Aproximação de função - Óbitos até julho
g(x) = 0.7263 * x^3 - 30.2455 * x^2 + 607.0170 * x + 1616.3719 #Aproximação de função - Recuperados até julho
h(x) = e^(5.3849 + 0.0143 * x) #Aproximação de função - Novos casos em julho
w = range(0, 31, length=100)

#plotando os gráficos numa grade 2x2
layout = grid(2, 2)
p = plot(layout=layout, leg=false)

#Plotando as funções
plot!(p[1], f, w, c=:black, lw=2)
plot!(p[2], F, w, c=:black, lw=2)
plot!(p[3], g, w, c=:black, lw=2)
plot!(p[4], h, w, c=:black, lw=2)

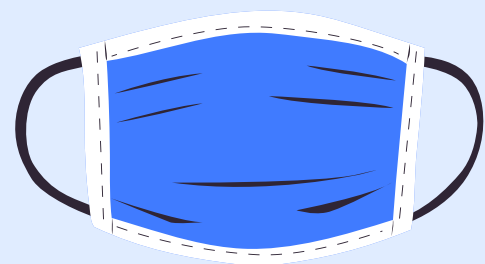
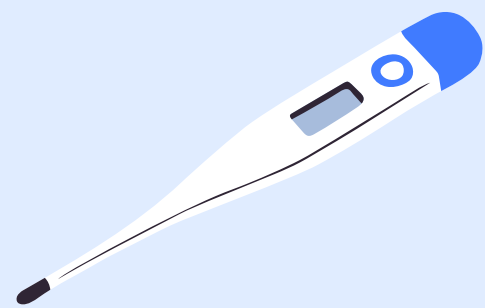
scatter!(p[1], dias_julho, confirmados_julho, title = "Casos confirmados até julho", xlabel = "dias de julho", ylabel = "confirmados", c=:orange)
scatter!(p[2], dias_julho, obitos_julho, title = "Óbitos até julho", xlabel = "dias de julho", ylabel = "óbitos", c=:orange)
scatter!(p[3], dias_julho2, recuperados_julho, title = "Recuperados até julho", xlabel = "dias de julho", ylabel = "recuperados", c=:orange)
scatter!(p[4], dias_julho, novos_casos_julho, title = "Novos casos em julho", xlabel = "dias de julho", ylabel = "novos casos", c=:orange)

#salvando a imagem com os gráficos
png(p, "graficos_julho")

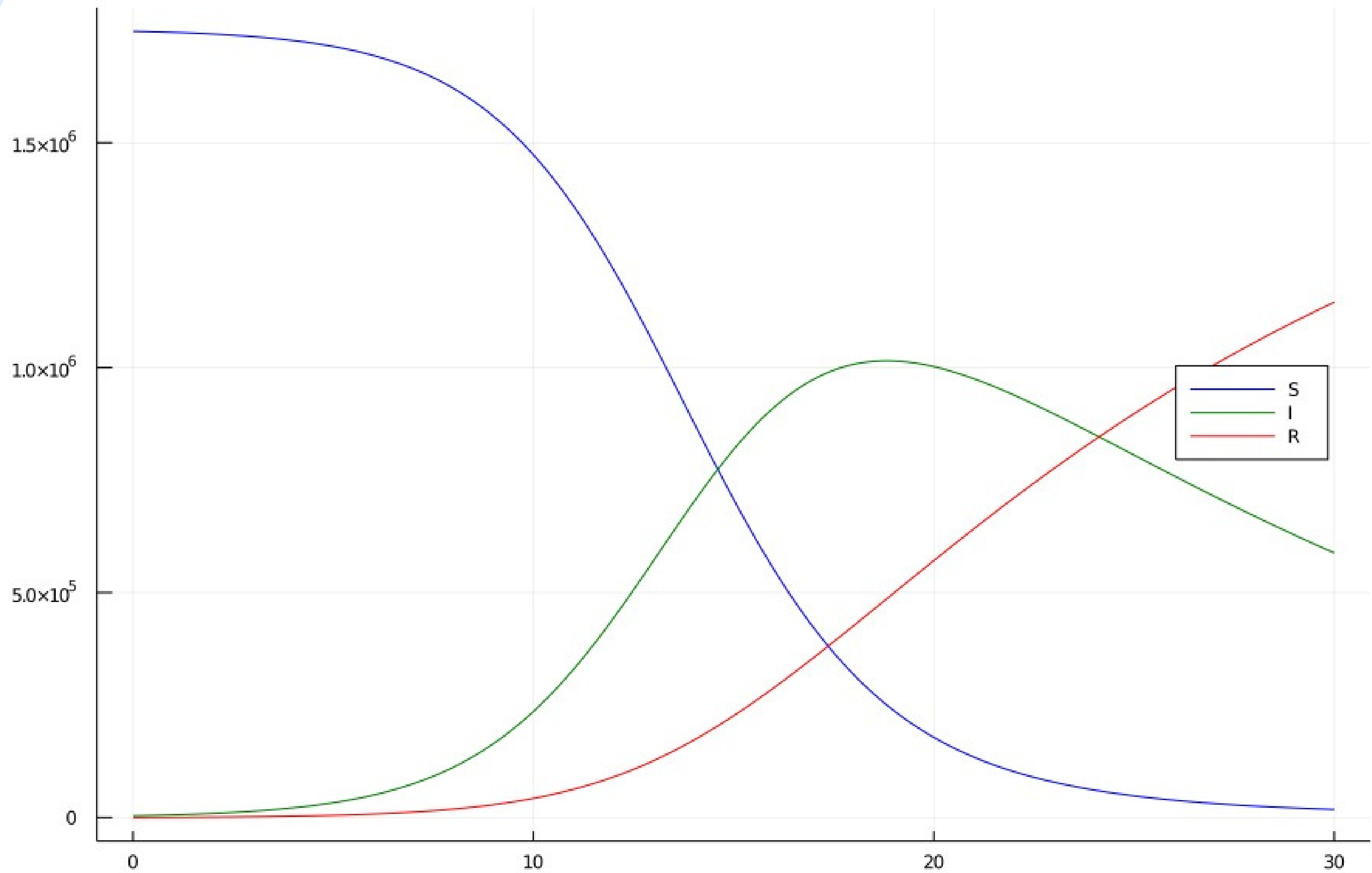
```

# Conclusões

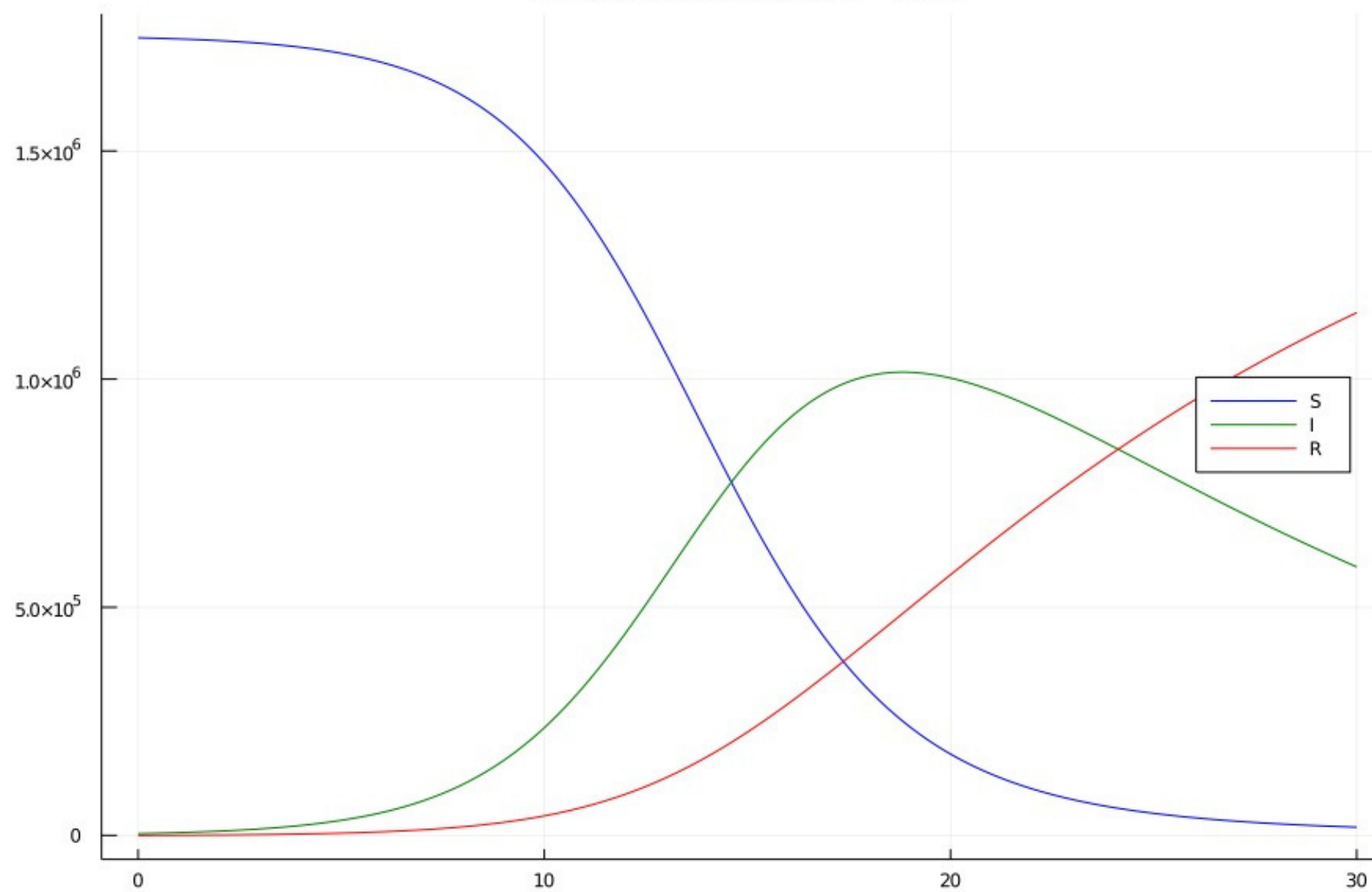
- Comparação com  $\beta = 0,5$
- Há necessidade de restrições
- Comparação entre bandeiras



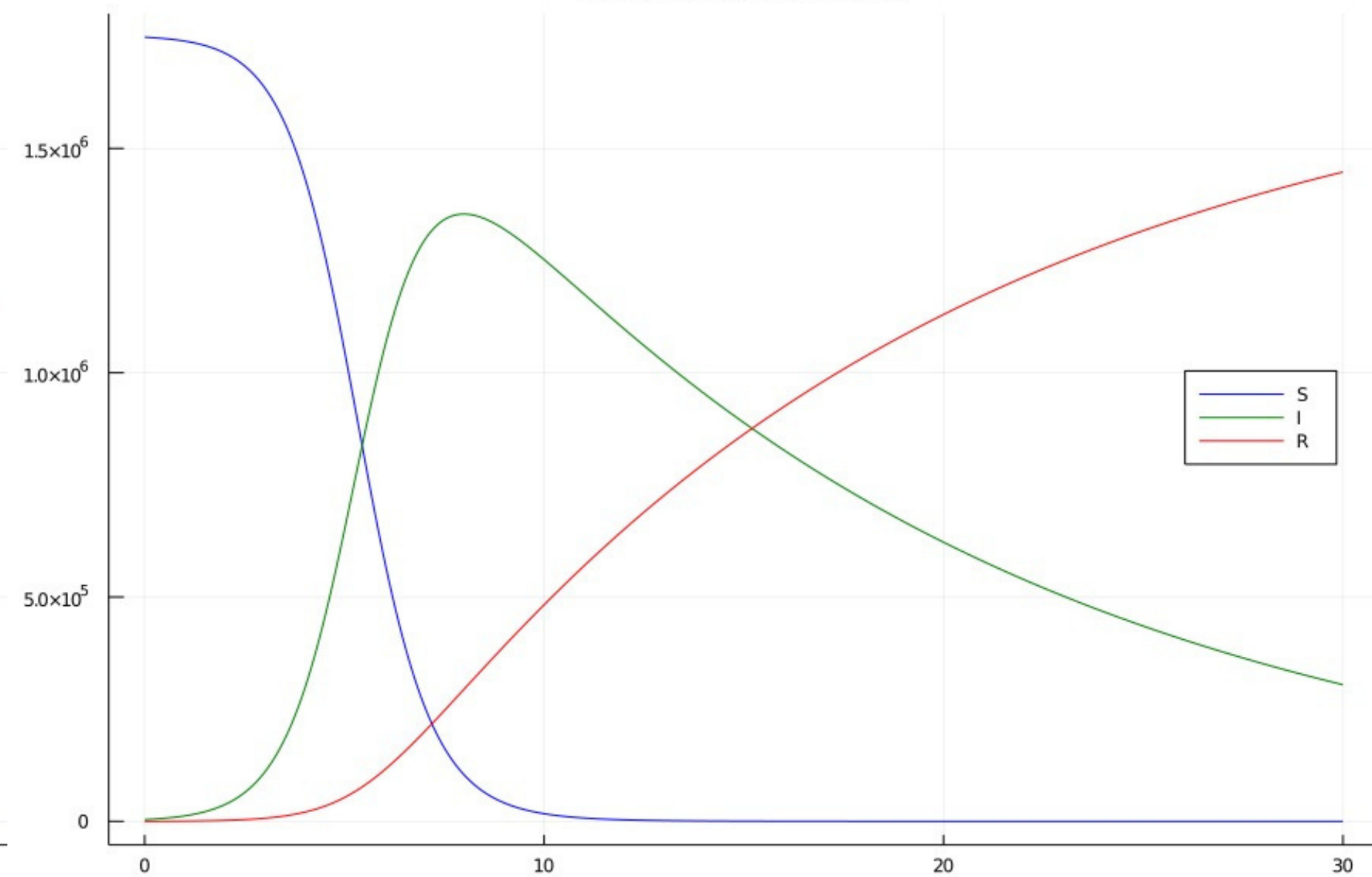
Modelo SIR Novembro - Ideal



Modelo SIR Novembro - Ideal



Modelo SIR Novembro



# Taxas

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

$$R_0 = R_e * s$$

Fonte: Observatório COVID-19 BR

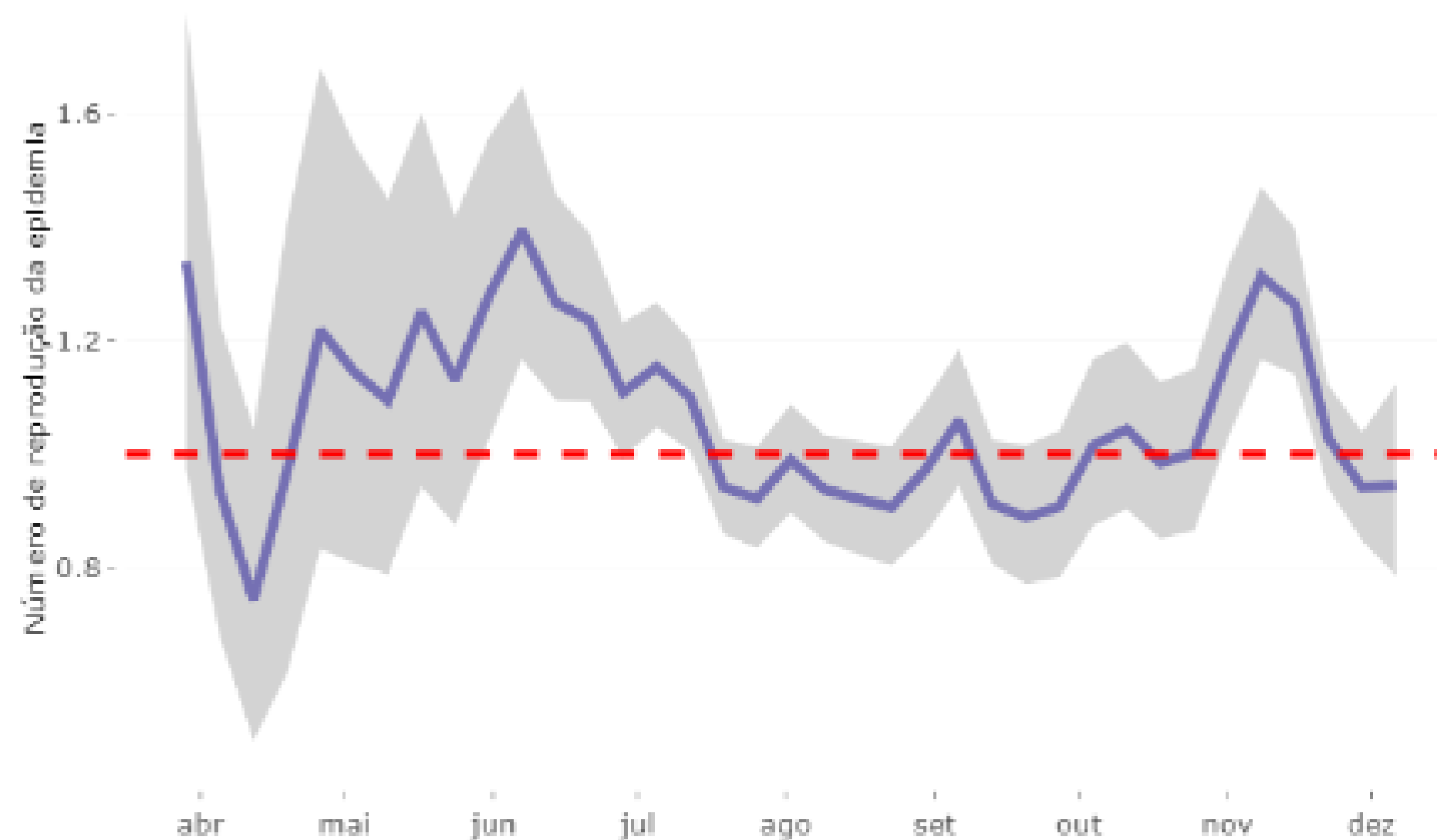
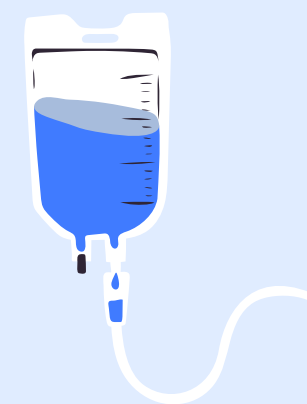
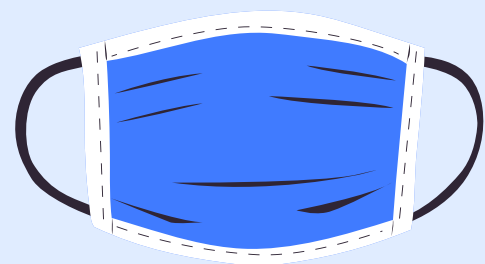
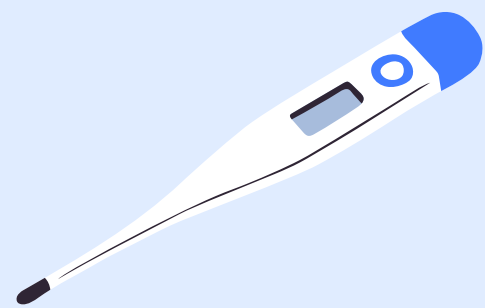


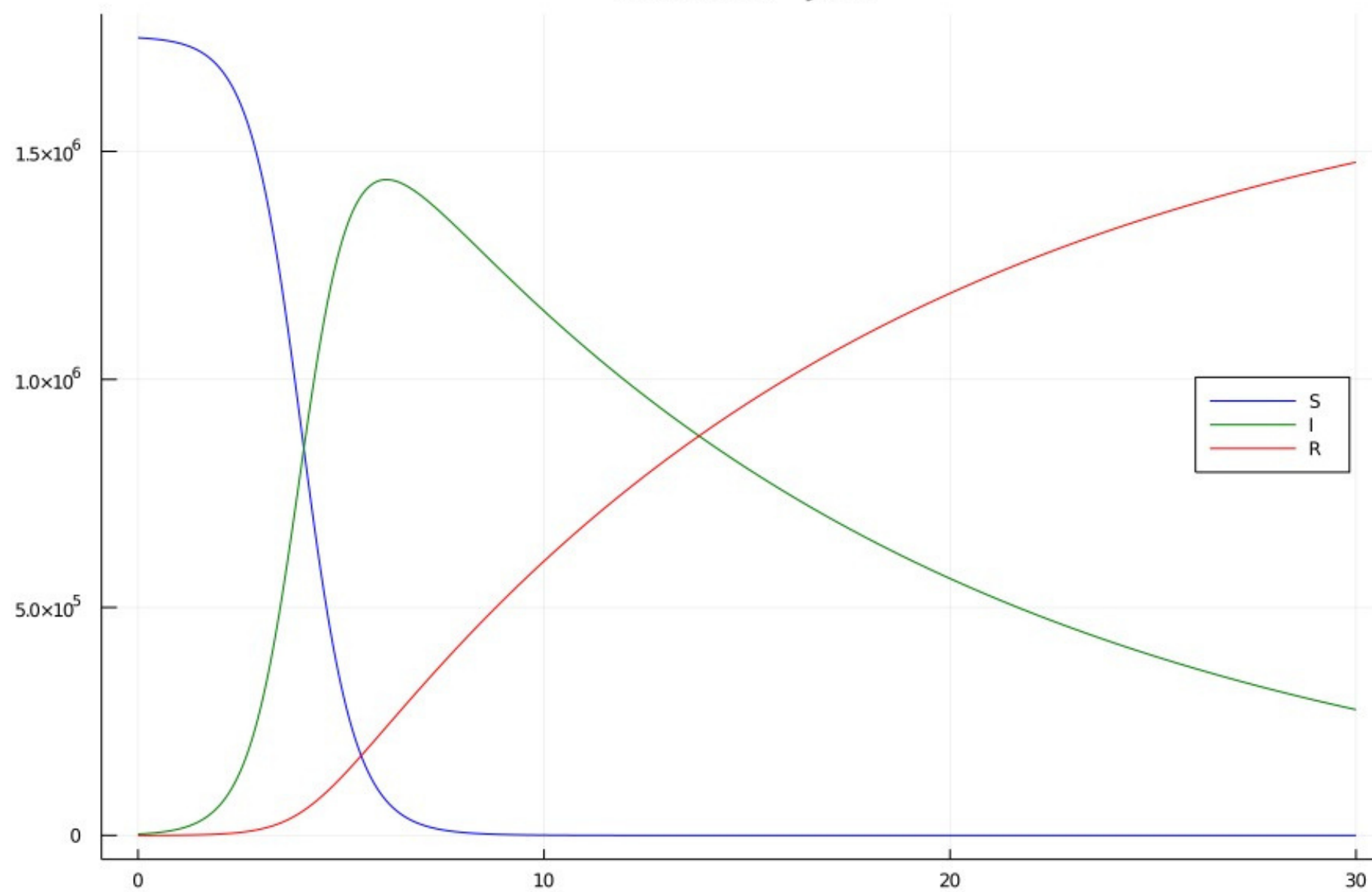
Figura 9.1: Gráfico da taxa de transmissão em Curitiba

# Conclusões

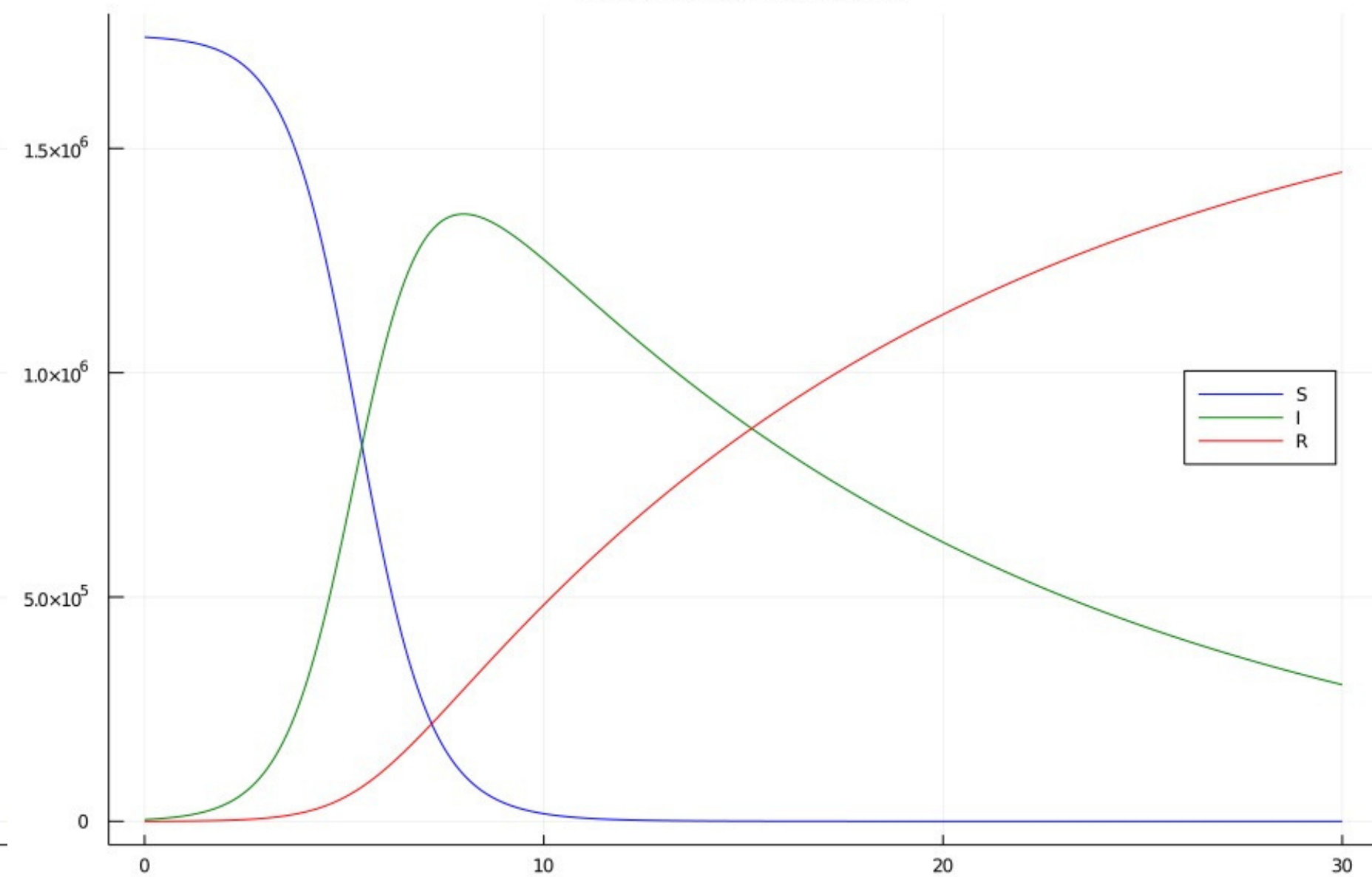
- Comparação com  $\beta = 0,5$
- Há necessidade de restrições
- Comparação entre bandeiras



Modelo SIR - Julho

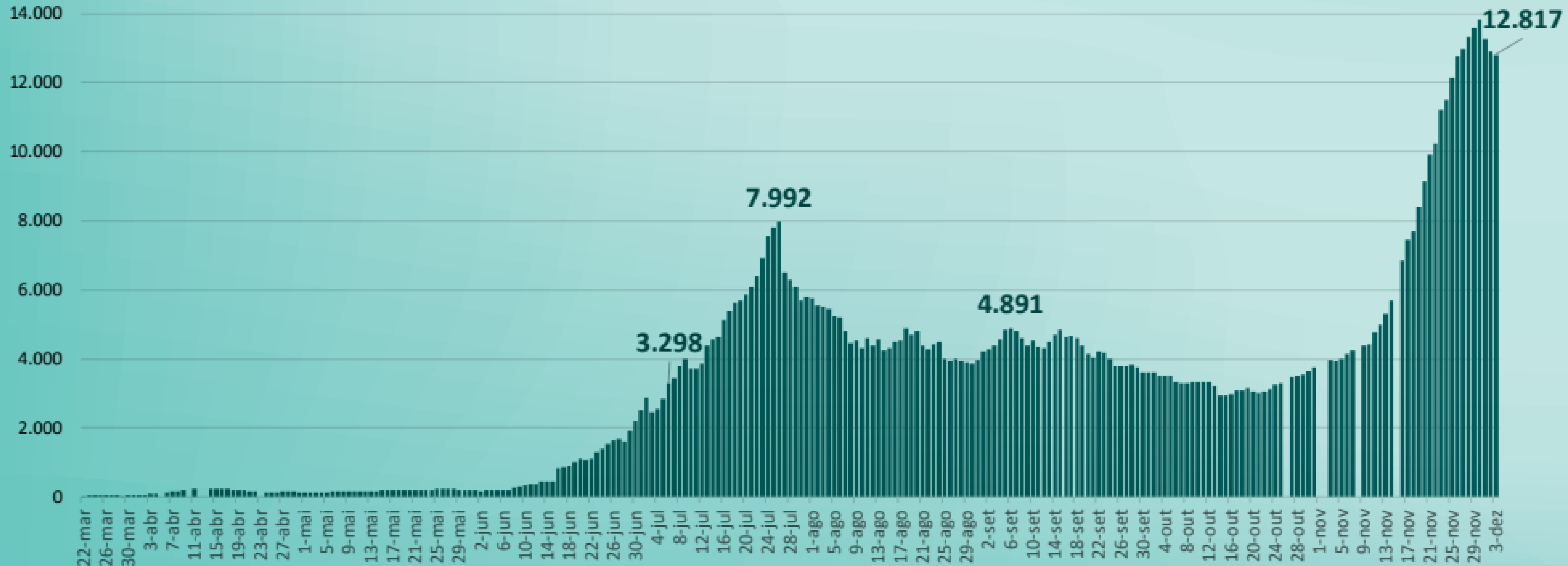


Modelo SIR Novembro





# Casos ativos diários de COVID-19 em Curitiba (PR), no período de 11/03 até 03/12/2020.



Fontes: SMS/CE/ Monitoramento COVID-19. Dados até 03/12/2020.

# Referências

PROTOCOLO DE RESPONSABILIDADE SANITÁRIA E SOCIAL. Curitiba contra o Coronavírus, Prefeitura de Curitiba, Curitiba, 29/11/2020. Disponível em: < <https://coronavirus.curitiba.pr.gov.br/numerosCovid> >. Acesso em: 29 de novembro de 2020 e datas anteriores.

MARTINS, Camila Marinelli et al. Modelo Preditivo da ocorrência de Covid-19 em município de médio porte no Brasil (Ponta Grossa - Paraná). Texto Contexto-Enfermagem, v. 29, 2020. Disponível em: < [https://www.scielo.br/scielo.php?pid=S0104-07072020000100204script=sci\\_arttexttln=pt](https://www.scielo.br/scielo.php?pid=S0104-07072020000100204script=sci_arttexttln=pt) >. Acesso em: 25 de novembro de 2020.

A Evolução epidêmica do COVID-19 - Modelo SIR. Disponível em:< <https://wp.ufpel.edu.br/fentransporte/2020/04/09/a-evolucao-epidmica-do-covid-19-modelo-sir/> >. Acesso em: 25 de novembro de 2020.

Modelos matemáticos serão estratégicos no combate ao coronavírus. Publicado em 11 de Junho de 2020. Disponível em: < <https://www.sbmt.org.br/portal/mathematical-models-will-be-strategic-in-combating-coronavirus/> >. Acesso em: 26 de novembro de 2020.

# Referências

GOMES, Sebastião C. P. et al. MODELAGEM DINÂMICA DA COVID-19 COM APLICAÇÃO A ALGUMAS CIDADES BRASILEIRAS. Universidade Federal do Rio Grande, Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Sul. Rio Grande do Sul. Disponível em  $\langle \text{https} : // \text{imef.furg.br/images/stories/documentos/2020/artigo2.pdf} \rangle$ . Acesso em: 26 de novembro de 2020.

R efetivo em Curitiba. Disponível em  $\langle \text{https} : // \text{covid19br.github.io/municipios.html?aba} = \text{aba3uf} = \text{PRmun} = \text{Curitiba} = \text{dia} \rangle$ . Acesso em: 10 de dezembro de 2020.

MONICH, J. W. Modelagem da Dinâmica Epidemiológica da Dengue. 2018. 50 f. Trabalho de Conclusão de Curso, Instituto de Matemática e Estatística, Universidade de São Paulo, São Paulo, 2018. Disponível em:  $\langle \text{https} : // \text{maratona.ime.usp.br/map/tcc/2018/JeniferMonichV1.pdf} \rangle$ .  $\langle \text{https} : // \text{www2.unifap.br/matematicaead/files/2016/03/Enviar} - \text{Ellison} - \text{TCC.pdf} \rangle$

$\langle \text{https} : // \text{www.ime.unicamp.br/valle/Teaching/2015/MS211/Aula14.pdf} \rangle$

$\langle \text{https} : // \text{planetcalc.com/5992/?language}_{\text{select}} = \text{ptthanks} = 1 \rangle$

SIQUEIRA, Abel. Modelo de Equações Diferenciais SIR para o COVID-19. Partes 1 e 2. Disponíveis em:  $\langle \text{https} : // \text{www.twitch.tv/videos/577731792} \rangle$ ,  $\langle \text{https} : // \text{www.twitch.tv/videos/577731791} \rangle$ ; Acesso em: 04 de dezembro de 2020.

# Obrigado pela atenção

Se cuidem, fiquem em casa,  
higienizem as mãos e usem máscara

