Universidade Federal do Rio de Janeiro
Instituto de Matemática
Departamento de Matemática Aplicada

Relatório Técnico INCTMat - Modelagem de Dados Epidemiológicos por Equações Diferenciais Ordinárias Universais

Luan Lima Freitas
Aluno do Bacharelado em Matemática Aplicada - UFRJ
Bolsista INCTMat (10/2021 - 02/2022)

Orientador
Prof. Dr. Ricardo Rosa
Departamento de Matemática Aplicada - IM/UFRJ

1 Introdução

Na vigência da bolsa de iniciação científica INCTMat, o aluno bolsista desenvolveu modelos e previsões para os dados da pandemia de COVID-19 na cidade do Rio de Janeiro. Os principais modelos trabalhados consistiram de *equações diferenciais ordinárias universais* (UODEs) [1]. As UODEs se encontram no escopo do *aprendizado científico de máquina* (SciML), trazendo consigo a proposta de mesclar técnicas clássicas de modelagem matemática com recursos de aprendizado de máquina. Toda a parte computacional foi implementada no ecossistema de SciML da linguagem Julia. Este trabalho dá sequência às realizações dos estudantes Gil Miranda [2] e Beatriz Farah [3] sob a orientação do professor Ricardo Rosa.

2 Equações Diferenciais Ordinárias Universais

Avanços recentes na área de aprendizado de máquina viabilizaram a utilização de técnicas de *deep learning* para a modelagem de fenômenos dos quais dispõe-se de grandes quantidades de dados. Uma vantagem desta abordagem é o aprendizado automático do conjunto completo de interações não-lineares, as quais podem ser tão complexas quanto numerosas. Por outro lado, esta estratégia é impraticável para a solução de problemas nos quais a disponibilidade de dados é exígua.

No extremo oposto do espectro encontram-se os modelos de equações diferenciais conhecidos como *mecanicistas*, na medida em que correspondem a uma simplificação de fenômenos naturais (ou sociais) por meio de um conjunto de leis explícitas, as quais representam a ação de um mecanismo e exprimem um conhecimento consolidado pela literatura científica.

No intuito de conjugar as virtudes e mitigar as limitações de ambos os extremos surgem diversos modelos e métodos híbridos, dentre os quais estão as *equações diferenciais ordinárias universais* (UODEs), objeto do presente trabalho. Uma UODE é uma equação diferencial ordinária definida por

$$\boldsymbol{u}'(t) = f(\boldsymbol{u}, t, U_{\boldsymbol{\theta}}(\boldsymbol{u}, t)),$$

onde $U_{\boldsymbol{\theta}}$ é um aproximador universal, i.e., uma função capaz de aproximar qualquer função suficientemente regular. No nosso caso, o aproximador universal será dado por uma rede neural de vetor de pesos $\boldsymbol{\theta}$ que denotaremos por $NN_{\boldsymbol{\theta}}$. A ideia por trás de uma UODE consiste em inscrever em uma estrutura *a priori* legada pela experiência científica um ou mais termos capazes de adquirir dos dados relações não-lineares potencialmente intrincadas e obscuras. Uma classe particular de UODE é formada pelas *equações diferenciais ordinárias neurais* (NODEs) [4]. Uma NODE é uma UODE dada por

$$\boldsymbol{u}'(t) = NN_{\boldsymbol{\theta}}(\boldsymbol{u}, t).$$

3 Dados e Modelos

Os dados disponibilizados na internet pela Secretaria Municipal de Saúde da Prefeitura do Rio de Janeiro foram manipulados para a obtenção das curvas de casos ativos, recuperados e mortos durante a pandemia de COVID-19 na cidade (cf. **Seção 5**). Em seguida, foi selecionado um período correspondente a uma "onda" da pandemia para ser modelado por meio das técnicas mencionadas acima, a saber, o período de 18/03/2020 a 30/06/2020 (cf. **Figura 1**).

Foi tomado como base dos experimentos o modelo compartimental SIRD. O modelo SIRD apresenta quatro compartimentos (S = "suscetíveis", I = "infectados", R = "recuperados" e D = "decessos") e a

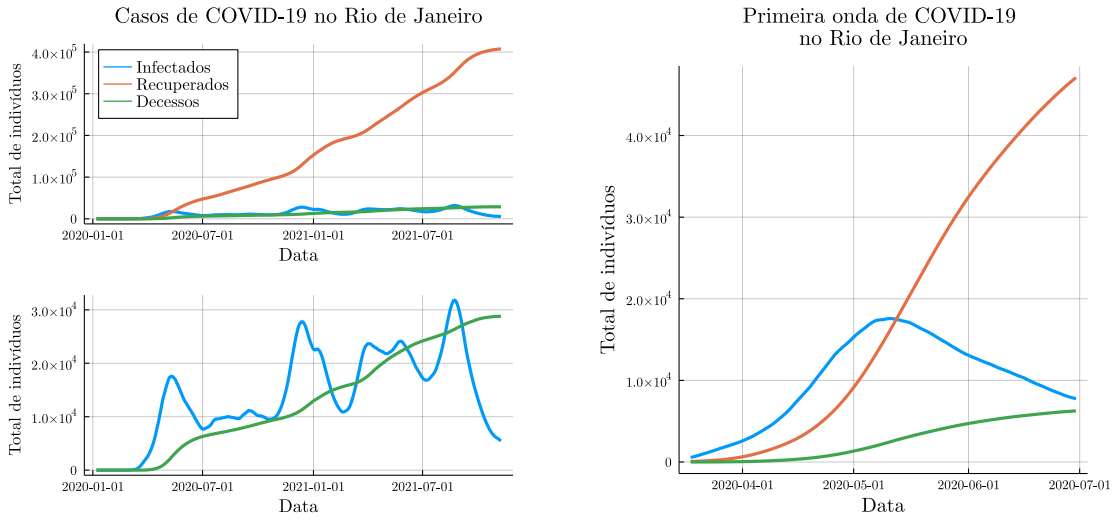

Figura 1: Média móvel dos casos de COVID-19 na cidade do Rio de Janeiro.

evolução do número de casos em cada compartimento é dada pelo sistema de equações

$$\begin{cases} \dot{S} = -\beta \frac{I}{N} S \\ \dot{I} = \beta \frac{I}{N} S - (\gamma_R + \gamma_D) I \\ \dot{R} = \gamma_R I \\ \dot{D} = \gamma_D I \end{cases}$$

onde $N = ||\boldsymbol{u}||_1$ e $\{\beta, \gamma_R, \gamma_D\} \subset [0, +\infty)$.

Tipicamente, a dinâmica mais árdua de se modelar em uma pandemia é a conversão dos indivíduos suscetíveis em infectados mediante a interação entre os dois grupos, no nosso caso representada pelo *termo de infecção* $\beta \frac{I}{N} S$. As demais dinâmicas podem ser muito satisfatoriamente modeladas por termos lineares, havendo inclusive em alguns casos a possibilidade de estimar seus respectivos parâmetros por meio de investigações empíricas suplementares. Desta feita, propomos substituir inteira ou parcialmente o termo de infecção do modelo por redes neurais, dando assim gênese a UODEs. No nosso estudo foram contempladas duas possibilidades:

- Substituir o termo de infecção por $NN_{\boldsymbol{\theta}}(S/N, I)$. A UODE assim definida foi nomeada de SIRD UODE βSI, onde a terminação βSI denota o termo substituído por uma rede neural.
- Substituir o termo de infecção por $NN_{\boldsymbol{\theta}}(\boldsymbol{u}) \frac{I}{N} S$. A UODE assim definida foi nomeada de SIRD UODE β, por motivo análogo ao especificado acima.

Os parâmetros dos modelos ora definidos foram ajustados aos dados coletados (cf. **Seção 6**), valendo-se o bolsista do ecossistema desenvolvido na linguagem de programação Julia sob o nome SciML, e em particular do pacote DiffEqFlux.jl [5]. O ajuste foi realizado utilizando-se os dados dos primeiros 20, 30, 40 e 55 dias e os modelos assim obtidos foram extrapolados para o período completo e comparados aos dados reais com a finalidade de avaliar sua capacidade de previsão (cf. **Figuras 2 e 3**).

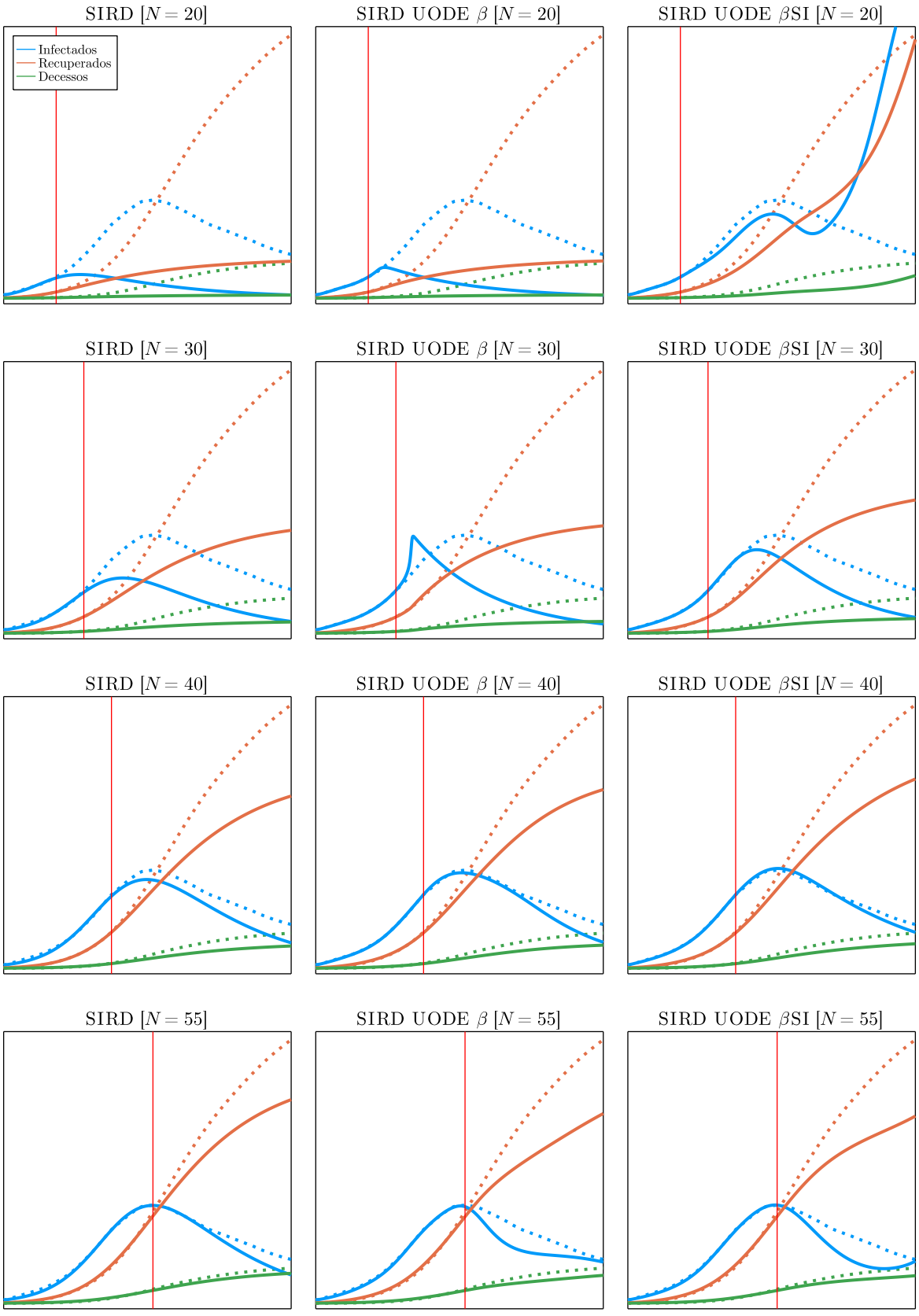

Figura 2: Ajuste dos modelos propostos aos dados da primeira onda de COVID-19 no Rio de Janeiro para 20, 30, 40 e 55 dias de treino (de cima para baixo). As curvas pontilhadas representam os dados e as curvas sólidas representam os modelos ajustados. As linhas vermelhas representam o último dia de treino.

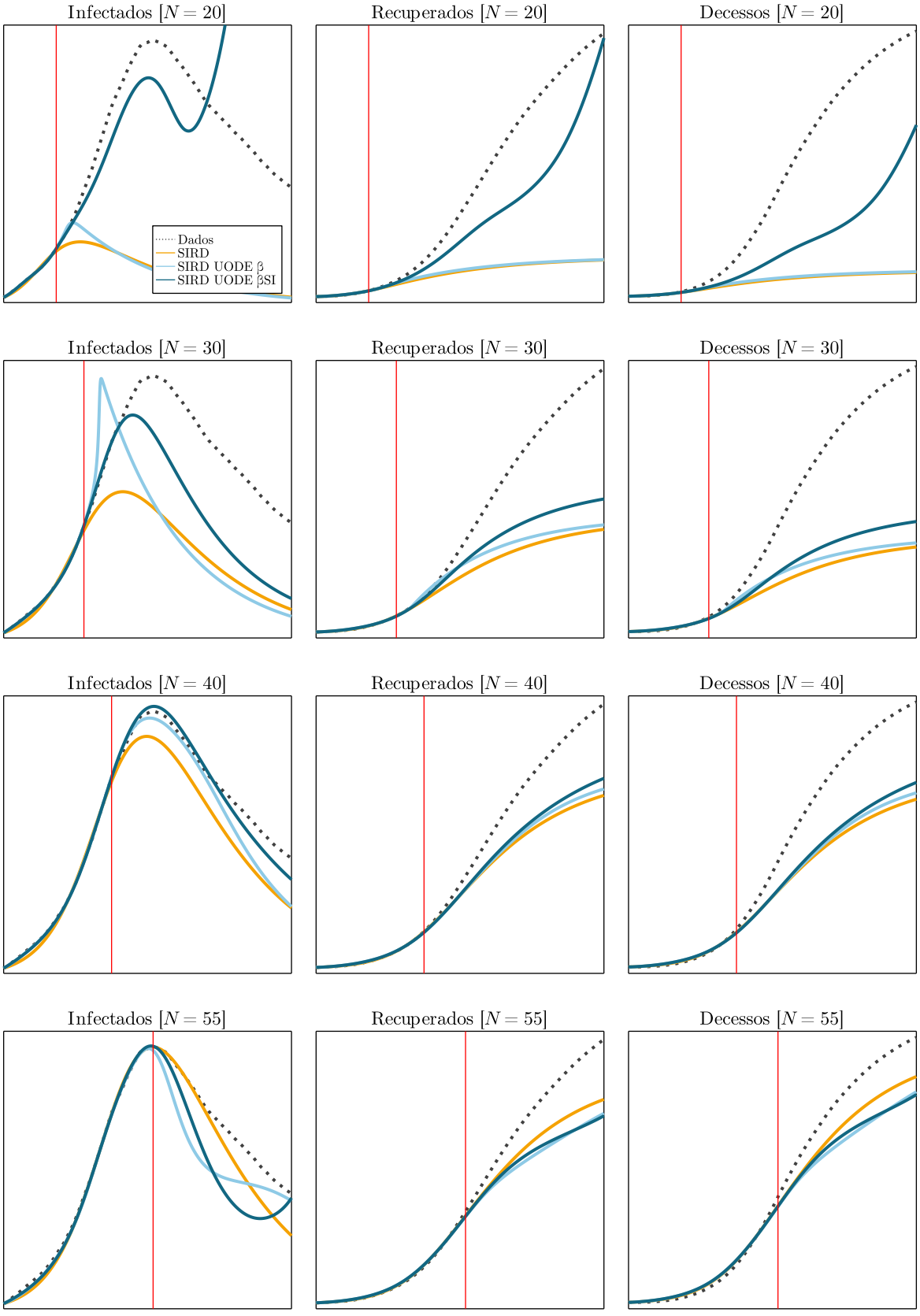

Figura 3: Comparação por compartimento do ajuste dos modelos propostos aos dados da primeira onda de COVID-19 no Rio de Janeiro para 20, 30, 40 e 55 dias de treino (de cima para baixo). As linhas vermelhas representam o último dia de treino.

4 Análise dos Resultados

Os modelos ajustados foram avaliados segundo a quantidade de *dias de boa previsão* após o último dia de treino. Esta quantidade é dada pelo número de dias consecutivos a partir do primeiro dia de previsão para os quais o erro relativo da previsão é inferior a uma certa tolerância dada. Os dias de boa previsão foram calculados para cada compartimento de cada modelo para diversos valores da tolerância (cf. **Figura 4**).

5 Tratamento dos Dados

Os dados retirados do endereço eletrônico [6] no dia 08/11/2021 consistem na listagem completa dos casos individuais registrados na cidade do Rio de Janeiro. Dentre os diversos atributos de cada caso foram de utilidade para nós a data do início dos sintomas, a evolução ("ativo", "recuperado" ou "óbito") e a data de evolução. Com esses dados em mãos, bastaria percorrer a lista caso a caso, incrementando as curvas pertinentes nos intervalos adequados. Foram necessárias, porém, algumas etapas de pré-processamento dos dados, uma vez que os mesmos se encontravam frequentemente e em larga escala incompletos ou em contradição com o bom-senso mais generoso. Não há motivos para discorrer sobre todos os pormenores desta empreitada, como, por exemplo, o fato de que a data de início dos sintomas mais antiga consta do dia 23/03/1945, havendo, aliás, muitos outros registros anteriores aos primeiros casos reportados na capital da província chinesa de Hubei. A seguir, nos limitaremos a expôr duas ocasiões onde a intervenção do bolsista foi realizada de maneira sistemática e teve maior impacto sobre as figuras finais obtidas.

A primeira anomalia relevante observada nos dados foi a existência de centenas de casos ativos cujas datas de início dos sintomas estendem-se ao longo de todo o período da pandemia (cf. **Figura 5**). É evidente que tal quadro é incompatível com a realidade biológica da COVID-19. Por conseguinte, a classe de tais casos foi alterada para recuperados, partindo-se da suposição de que, de maneira geral, os óbitos devem ter sido registrados com maior zelo. Deve-se reparar, ademais, na explosão de casos ativos no final do perído apreciado. Estes, por sua vez, foram considerados como verdadeiros ativos e suas classes não foram alteradas.

A irregularidade remanescente diz respeito à impossibilidade de se obter de maneira direta e confiável o tempo de infecção por COVID-19 de uma grande quantidade de casos recuperados e de óbitos. Com efeito, cerca de 93% dos recuperados não possuem registro de suas respectivas datas de evolução. Além disso, para uma parcela significativa dos óbitos, as datas de evolução coicidem com as datas de início dos sintomas, o que não é menos do que duvidoso. Para corrigir tais problemas sorteou-se, para cada caso incompleta ou incorretamente registrado, uma quantidade de dias de infecção do vetor contendo as quantidades de dias de infecção de todos os casos completa e corretamente registrados da classe correspondente (cf. **Figura 6**).

6 Estratégia de Otimização

Para ajustar o modelo SIRD aos dados minimizou-se a soma dos quadrados dos resíduos (RSS) do modelo pelo método de Nelder-Mead disponível no pacote Optim.jl. Foi feita uma reparametrização do modelo:

$$(\beta, \gamma_R, \gamma_D) = \left(\hat{\beta}^2, \hat{\gamma}_R^2, \hat{\gamma}_D^2\right),$$

para a qual temos $\left\{\hat{\beta}, \hat{\gamma}_R, \hat{\gamma}_D\right\} \subset \mathbb{R}$, e então foram ajustados os parâmetros $\hat{\beta}$, $\hat{\gamma_R}$ e $\hat{\gamma_D}$ e a condição inicial $S(1)$.

A rede neural utilizada foi implementada em Julia por meio do código:

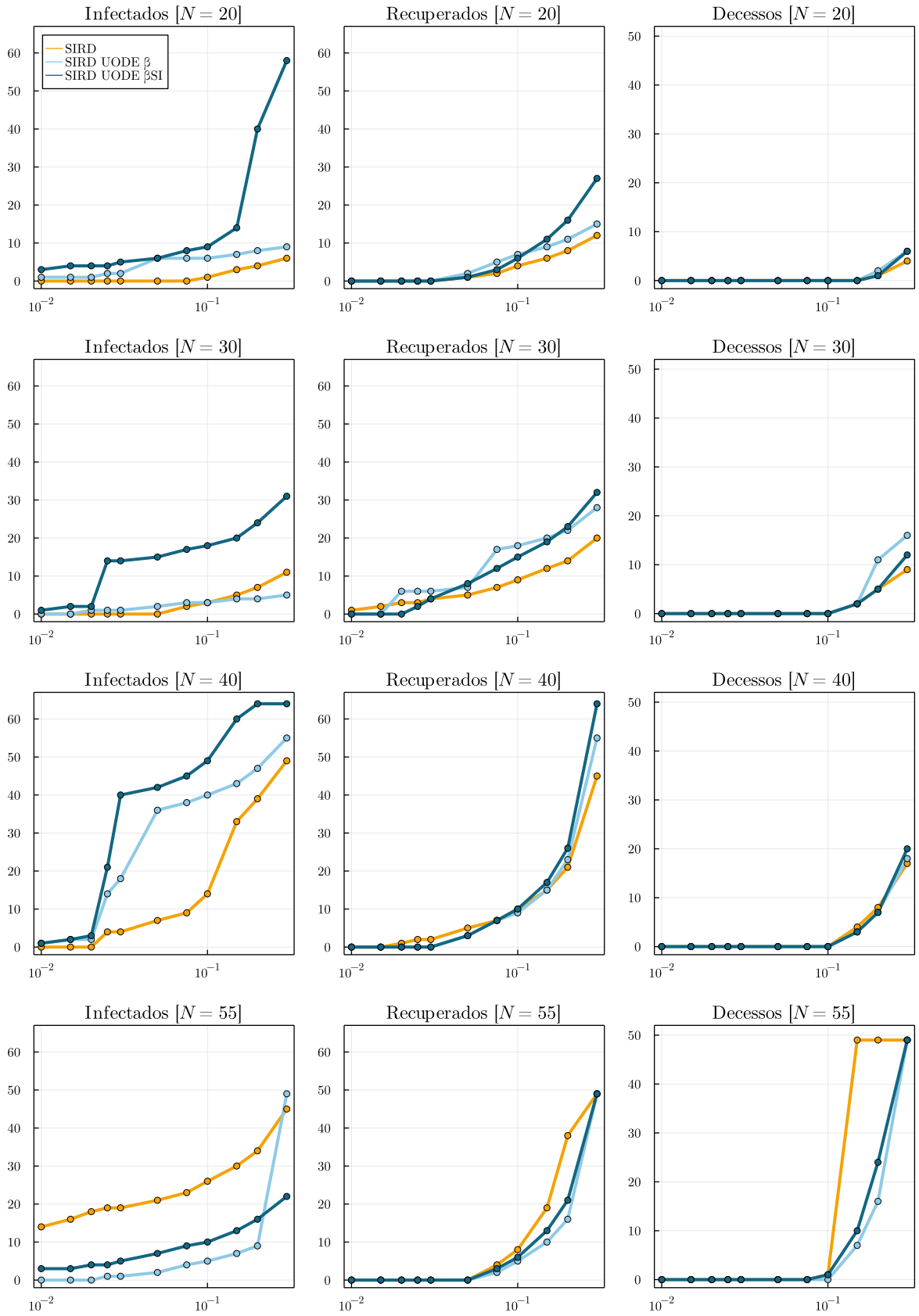

Figura 4: Dias de boa previsão por tolerância para 20, 30, 40 e 55 dias de treino (de cima para baixo).

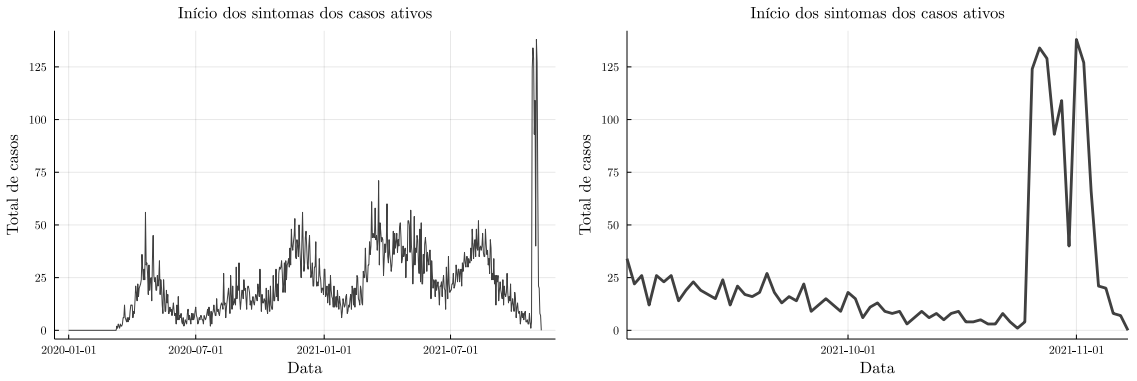

Figura 5: Início dos sintomas dos casos ativos ao longo da pandemia.

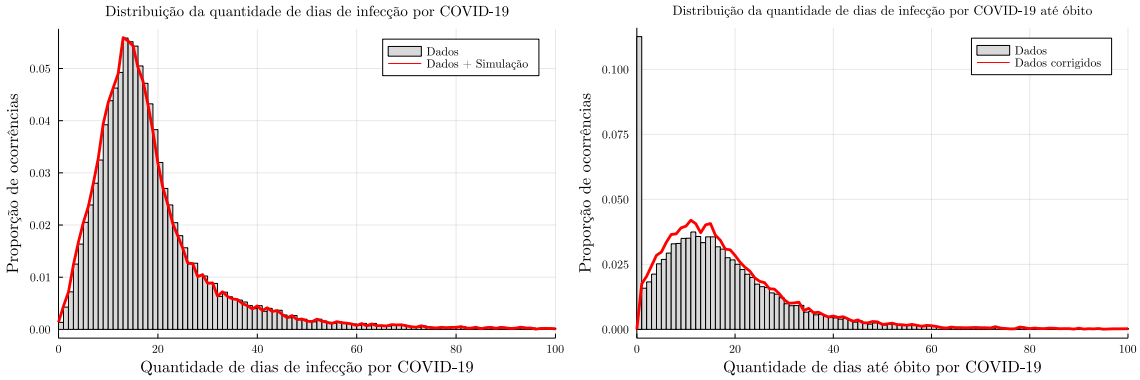

Figura 6: Distribuição dos tempos de infecção por COVID-19.

```
NN = FastChain(FastDense(n,16,tanh), FastDense(16,16,tanh), FastDense(16,1),
                (x, p) -> x.^2)
```

onde `n = 2` ou `4`. O ajuste das UODEs foi feito em diversas etapas. Primeiramente, otimizaram-se os parâmetros e a condição inicial $S(1)$ do modelo SIRD subjacente da forma descrita acima. A experiência então demonstrou ser vantajoso reduzir a escala do problema. Isso foi alcançado dividindo todas entradas dos dados de treino pela de maior valor entre as mesmas. A etapa seguinte pode ser considerada o *pulo do gato* do nosso algoritmo, tendo um impacto decisivo no modelo final obtido e consistindo em *pré-ajustar* a rede neural ao termo de infecção ou ao parâmetro β, a depender de cada UODE, os quais a mesma tem como propósito substituir. A escolha de função objetivo para a realização desta tarefa foi novamente a RSS e o vetor paramétrico $\boldsymbol{\theta}$ foi otimizado através do algoritmo de Broyden–Fletcher–Goldfarb–Shanno (BFGS, cf. **Seção 7**) [7, 8] implementado no pacote DiffEqFlux.jl. Enfim, foi realizada uma rodada de ajuste simultâneo dos parâmetros γ_R, γ_D e $\boldsymbol{\theta}$, minimizando-se a RSS do modelo UODE pelo algoritmo BFGS, restando apenas multiplicar os valores finais obtidos para cada compartimento pelo mesmo fator de escala utilizado anteriormente para recuperar a escala original do problema.

7 BFGS

No que se segue, faremos uma breve exposição do algoritmo BFGS, que deverá ser tomada como não mais que uma insinuação em linhas gerais do sentido do mesmo. O algoritmo BFGS é um método iterativo desenvolvido para resolver o problema padrão de minimizar $f(\boldsymbol{x})$ para $\boldsymbol{x} \in \mathbb{R}^d$ qualquer. Partindo de uma

estimativa $\boldsymbol{x}_k$ de $\boldsymbol{x}$, a próxima direção de busca $\boldsymbol{p}_k$ é dada pela solução da equação

$$B_k \boldsymbol{p}_k = -\nabla f(\boldsymbol{x}_k),$$

onde B_k é uma aproximação da matriz Hessiana obtida iterativamente ao longo do algoritmo. Teremos então

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \gamma \boldsymbol{p}_k,$$

onde γ é dado pela solução do problema de busca linear

$$\gamma = \underset{\gamma > 0}{\operatorname{argmin}} \ f(\boldsymbol{x}_k + \gamma \boldsymbol{p}_k)$$

ou algum substituto satisfatório cabível. Em verdade, o passo característico do algoritmo BFGS consiste na maneira como as aproximações para a matriz Hessiana são obtidas:

$$B_{k+1} = B_k + \frac{\boldsymbol{y}_k \boldsymbol{y}_k^T}{\boldsymbol{y}_k^T \boldsymbol{s}_k} - \frac{B_k \boldsymbol{s}_k \boldsymbol{s}_k^T B_k^T}{\boldsymbol{s}_k^T B_k \boldsymbol{s}_k},$$

onde

$$\begin{aligned} \boldsymbol{y}_k &= \nabla f(\boldsymbol{x}_{k+1}) - \nabla f(\boldsymbol{x}_k) \\ \boldsymbol{s}_k &= \boldsymbol{x}_{k+1} - \boldsymbol{x}k \end{aligned}$$

A discussão das fórmulas indicadas acima vai além do escopo deste relatório. Cabe pontuar, enfim, que a inversa de B_k pode ser acessada eficientemente através da expressão

$$B_k^{-1} = B_{k-1}^{-1} + \frac{(\boldsymbol{s}_k^T \boldsymbol{y}_k + \boldsymbol{y}_k^T B_{k-1}^{-1} \boldsymbol{y}_k)(\boldsymbol{s}_k \boldsymbol{s}_k^T)}{(\boldsymbol{s}_k^T \boldsymbol{y}_k)^2} - \frac{B_{k-1}^{-1} \boldsymbol{y}_k \boldsymbol{s}_k^T + \boldsymbol{y}_k \boldsymbol{s}_k^T B_{k-1}^{-1}}{\boldsymbol{s}_k^Y \boldsymbol{y}_k}.$$

8 Difusão e Divulgação

Não houve tempo hábil para a inscrição do projeto na 11ª Semana de Integração Acadêmica (SIAc) da UFRJ, realizada entre os dias 14 e 18 de Fevereiro de 2022. Aguardamos ansiosamente pela próxima edição deste evento para comunicar os resultados da nossa pesquisa à comunidade acadêmica.

Todo o código desenvolvido para este projeto está disponível na forma de um repositório público do GitHub [9].

9 Referências

[1] RACKAUCKAS, Chris *et al.* **Universal Differential Equations for Scientific Machine Learning**. arXiv:2001.04385 [cs.LG], 2020.

[2] MIRANDA, Gil. **Modelos Dinâmicos Híbridos em Problemas Científicos com Aprendizado de Máquina**. Relatório INCTMat, 2021. Comunicação pessoal.

[3] FARAH, Beatriz. **Equações Diferenciais Universais e Aprendizado de Maquina**. Relatório PIBIC, 2021. Comunicação pessoal.

[4] CHEN, Ricky *et al.* **Neural Ordinary Differential Equations**. arXiv:1806.07366v5 [cs.LG], 2019.

[5] RACKAUCKAS, Chris *et al.* **DiffEqFlux.jl — A Julia Library for Neural Differential Equations**. arXiv:1902.02376v1 [cs.LG], 2019.

[6] RIO DE JANEIRO, Secretaria Municipal de Saúde do. **CEP dos casos confirmados de COVID-19 no município do Rio de Janeiro**. Rio de Janeiro: Prefeitura do Rio de Janeiro, 2020. Disponível em: `https://www.data.rio/datasets/PCRJ::cep-dos-casos-confirmados-de-covid-19-no-município-do-rio-de-janeiro-1/about`. Acesso em: 8 nov. 2021.

[7] FLETCHER, R. **Practical Methods of Optimization**. 2ª edição. West Sussex: John Wiley & Sons, 1987.

[8] KOCHENDERFER, Mykel; WHEELER, Tim. **Algorithms for Optimization**. Massachusetts: The MIT Press, 2019.

[9] FREITAS, Luan. **UDE-COVID-RJ**. Rio de Janeiro: 2022. Disponível em: `https://github.com/Luan-Lima-Freitas/UDE-COVID-RJ`. Acesso em: 29 mar. 2022.