Proceeding Series of the Brazilian Society of Computational and Applied Mathematics, v. 12, n. 1, 2026.

Trabalho apresentado no XLIV CNMAC, Fundação Getulio Vargas - Rio de Janeiro - RJ, 2025.

Proceeding Series of the Brazilian Society of Computational and Applied Mathematics

Aplicação de PINN em Redes Neurais para Treinamento do Modelo SIR na Covid-19

Douglas S. de Albuquerque[1], Renato S. Silva[2], Regina C. C. de Almeida[3]
LNCC, Petrópolis, RJ

Resumo. Extrair padrões dos dados é crucial na modelagem computacional, especialmente em epidemiologia, onde a qualidade da informação é essencial. Redes neurais – convencionais ou informadas por física – são métodos poderosos para essa aproximação. Neste trabalho, utilizamos modelos epidemiológicos, com dados sintéticos e reais, para testar estes métodos, constatando a superioridade das redes neurais informadas por física.

Palavras-chave. Redes Neurais, MLP, PINN, Dados Epidemiológicos, Modelo SIR, Covid-19

1 Introdução

A compreensão de fenômenos físicos e a capacidade de replicar seu comportamento são fundamentais para embasar decisões estratégicas e orientar políticas públicas eficazes. Nesse contexto, destaca-se a epidemiologia, área dedicada à modelagem matemática da transmissão de doenças contagiosas, cujos resultados dependem da qualidade e interpretação de dados epidemiológicos. As informações extraídas dessas observações permitem inferir padrões de propagação de patógenos, sendo essenciais para obter uma compreensão aprofundada sobre a dinâmica das enfermidades [2].

Diante deste desafio, as redes neurais surgem como modelos computacionais capazes de assimilar padrões complexos nos dados e reproduzi-los precisamente [6]. Contudo, um avanço significativo ocorre com o surgimento das redes neurais informadas por física [9]. As redes neurais convencionais e as informadas por física têm sido amplamente utilizadas para resolver problemas epidemiológicos [1, 5, 10].

Neste trabalho, os dois tipos de redes neurais serão aplicados a dados epidemiológicos regidos pelo Modelo SIR [3], então, a abordagem com melhor desempenho foi aplicada a dados reais, da doença Covid-19, para validação. O trabalho está estruturado da seguinte forma: a Seção 2 apresenta o Modelo SIR; a Seção 3 descreve o primeiro tipo de rede neural utilizada neste trabalho; a Seção 4 introduz a rede neural informada por física; a Seção 5 detalha os testes computacionais, divididos entre dados sintéticos e reais; e, por fim, a Seção 6 apresenta as considerações finais.

2 Modelo SIR

Os modelos epidemiológicos compartimentais baseiam-se na divisão da população em compartimentos distintos, fornecendo uma estrutura para analisar a propagação de doenças infecciosas e sua evolução ao longo do tempo [2], destacando-se o Modelo SIR, desenvolvido por Kermack e McKendrick [3]. Neste modelo se assume a hipótese que o indivíduo adquire alguma enfermidade

[1]dougalbu@posgrad.lncc.br
[2]rssr@lncc.br
[3]rcca@lncc.br

e permanece infectado até desenvolver imunidade permanente. Os indivíduos são dispostos de acordo com os estados da doença, sendo também denominados compartimentos. Convém ressaltar que esses indivíduos integram a mesma população e não estão fisicamente isolados, o que propicia a interação entre eles e possibilita a transição entre os compartimentos. Assim, os estados são classificados como: Suscetíveis ($\boldsymbol{S}$), indivíduos saudáveis passíveis de infecção por contato; Infectados ($\boldsymbol{I}$), portadores ativos capazes de transmitir a doença; e Recuperados ($\boldsymbol{R}$), aqueles que adquiriram imunidade. É importante destacar que as subpopulações são funções temporais cuja soma equivale à população total N, que é constante para qualquer tempo (t) [2]. Ou seja:

$$S(t) + I(t) + R(t) = N. \tag{1}$$

O Modelo SIR é definido por um Sistema de Equações Diferenciais Ordinárias, descrito pela equação (2) [2, 3]. Onde β representa a taxa de contato, medindo a transição diária de indivíduos suscetíveis para infectados pelo contato com uma pessoa contaminada pela doença. Por sua vez, γ é a taxa de recuperação, definida como o inverso do tempo médio de recuperação, em dias, que indica o período em que o indivíduo permanece infectado.

$$\begin{aligned} \frac{dS}{dt} &= -\beta SI; \\ \frac{dI}{dt} &= \beta SI - \gamma I; \\ \frac{dR}{dt} &= \gamma I. \end{aligned} \tag{2}$$

3 Rede Multi Layer Perceptron

As Redes Neurais Artificiais são modelos de aprendizado de máquina capazes de aprender e replicar padrões complexos a partir de observações de dados [6]. De modo geral, esses modelos realizam o mapeamento do conjunto de entrada $\boldsymbol{X}$ para uma saída $\boldsymbol{y}$ através de um processo de treinamento. Este processo consiste na busca do conjunto ótimo de parâmetros, denominados pesos e viés (*bias*), que irá reduzir o erro entre a saída prevista e o valor verdadeiro minimizando uma Função de Perda (*Loss Function*) [6]. Neste contexto, uma métrica comumente empregada para quantificar essa discrepância é o erro médio quadrado (*mean squared error*, MSE), definida como:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2. \tag{3}$$

Tal que, $\hat{y}$ é a saída prevista, y é o valor verdadeiro e n é o número de observações.

Uma rede neural é definida por sua arquitetura, que irá estabelecer o número de camadas e nós, ou neurônios, que a compõe. Inicialmente, os dados são recebidos pela camada de entrada, onde são distribuídos e processados pelos nós, encaminhando a informação para a camada seguinte, denominada camada oculta. Este processo é reproduzido até alcançar a camada de saída, que consolida o treinamento, fornecendo a resposta final do modelo. Esse fluxo de dados, denominado *feedforward*, ocorre de maneira unidirecional, garantindo que a informação transite da camada de entrada até a saída sem retroalimentação. Este tipo de rede, com mais de uma camada oculta, é denominado *Multi Layer Perceptron* (MLP) [6].

No decorrer do treinamento, aplica-se uma função de ativação ϕ em cada camada, permitindo que a rede aprenda relações complexas nos dados e introduza não linearidade nas operações [6]. Sejam os pesos definidos como a matriz $\boldsymbol{W} \in \mathbb{R}^{\text{nós} \times \text{tam.entrada}}$ e os *bias* como o vetor $\boldsymbol{b} \in \mathbb{R}^{\text{nós}}$.

DOI: 10.5540/03.2026.012.01.0254

É possível descrever a operação do *feedforward* em uma rede com k camadas como:

$$\boldsymbol{y} = \phi_k\Big(\boldsymbol{W}_k\phi_{k-1}\Big(\cdots\phi_3\Big(\boldsymbol{W}_2\phi_1\Big(\boldsymbol{W}_1\boldsymbol{x}+\boldsymbol{b}_1\Big)+\boldsymbol{b}_2\Big)\cdots+\boldsymbol{b}_{k-1}\Big)+\boldsymbol{b}_k\Big). \tag{4}$$

Então, o algoritmo *Backpropagation* é aplicado na equação (4) para ajustar os parâmetros da rede minimizando a *loss function* $\boldsymbol{L}$. Este algoritmo calcula, de forma automatizada, o gradiente de $\boldsymbol{L}$ em relação aos parâmetros por meio da regra da cadeia. Os parâmetros são atualizados se movendo na direção oposta aos gradientes, com base em uma taxa de aprendizado η, conforme descrito na equação abaixo [6]:

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) - \eta\frac{\partial \boldsymbol{L}}{\partial \omega_{i,j}}. \tag{5}$$

Onde, $\omega_{i,j}(t)$ representa o valor de peso atual e $\frac{\partial \boldsymbol{L}}{\partial \omega_{i,j}}$ o gradiente da função de perda em relação ao peso $\omega_{i,j}(t)$.

4 Redes Neurais Informadas por Física

O treinamento convencional de redes neurais é essencialmente baseado em dados e ignora conhecimentos adquiridos *a priori*, ou seja, não considera as leis físicas que governam o sistema. Neste contexto, foram elaboradas as Redes Neurais Informadas por Física (*Physics-Informed Neural Network*, PINNs) [9], que incorporam ao treinamento informações provenientes de modelos matemáticos que representam a física do problema. Assim, a PINN adiciona o resíduo de equações diferenciais à *loss function*, buscando minimizar a soma entre o erro proveniente dos dados e o resíduo das equações (equação (10)). Essa integração do conhecimento físico atua como um agente regularizador, restringindo o espaço de busca das soluções admissíveis e garantindo uma boa capacidade de generalização, mesmo quando há poucos dados disponíveis.

Neste trabalho, será empregado o Modelo SIR (equação (2)). Desse modo, adotando MSE como métrica (equação (3)), o resíduo é calculado da seguinte forma:

$$MSE_S = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{dS(t_i)}{dt_i} + \beta\, S(t_i)I(t_i)\right)^2; \tag{6}$$

$$MSE_I = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{dI(t_i)}{dt_i} - \beta\, S(t_i)I(t_i) + \gamma\, I(t_i)\right)^2; \tag{7}$$

$$MSE_R = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{dR(t_i)}{dt_i} - \gamma\, I(t_i)\right)^2. \tag{8}$$

$$\tag{9}$$

Assim, a *loss function* da PINN incorporada com o Modelo SIR é definida por:

$$\boldsymbol{L} = MSE_{Dados} + MSE_{SIR}. \tag{10}$$

Onde, o termo MSE_{Dados} corresponde à equação (3), e o termo MSE_{SIR} trata-se da soma das equações (6), (7) e (8). Com efeito, para assegurar que as restrições físicas sejam cumpridas em todo o domínio, pontos de colocação são definidos ao longo do intervalo de tempo. Esses pontos permitem avaliar os resíduos do modelo de forma contínua, garantindo que as equações diferenciais sejam satisfeitas em todas as regiões do conjunto de treinamento [9].

DOI: 10.5540/03.2026.012.01.0254

5 Testes Computacionais

Para avaliar o desempenho das redes, foram gerados dados sintéticos, livres de ruído, a partir do Modelo SIR (equação (2)), simulando 51 dias de uma epidemia genérica, em um intervalo que destaca as principais dinâmicas das curvas. O conjunto de entrada corresponde às observações diárias das três subpopulações deste modelo. Para otimizar o treinamento, os dados foram normalizados no intervalo $[0, 1]$. Os parâmetros adotados para a simulação foram $\beta = 0,3$ (taxa de contato) e $\gamma = 0,1$ (taxa de recuperação). Durante o treinamento, os dados foram divididos em 80% para o conjunto de treino e 20% para teste. Assim, a parcela destinada ao treino foi empregada para ajustar os parâmetros da rede, enquanto a porção reservada para teste serviu para avaliar o processo.

Neste trabalho, utilizou-se a mesma arquitetura para a MLP e a PINN, permitindo a comparação direta da eficácia de ambas. As redes foram configuradas com 64 nós em 3 camadas ocultas, 1 nó de entrada (tempo t) e 3 nós de saída (Modelo SIR). Em cada camada oculta, aplicou-se a função de ativação $ReLU$, enquanto a camada de saída utilizou a função *Sigmoid* para manter os valores no intervalo $[0, 1]$ [6]. Foram definidas 10000 épocas, equivalentes a iterações típicas dos métodos iterativos clássicos. Para evitar o superajuste na MLP e assegurar uma boa generalização, implementou-se uma técnica de parada antecipada, interrompendo o treinamento se, após 50 épocas, a variação na *loss function* for inferior a 1×10^{-7}. Essa técnica foi omitida na PINN para avaliar o efeito regularizador das equações físicas incorporadas. O otimizador *Adam* [4], da biblioteca *PyTorch*, foi utilizado com taxa de aprendizado definida empiricamente como $\eta = 2,5 \times 10^{-2}$ para a PINN e $\eta = 5 \times 10^{-4}$ para a MLP, ajuste necessário para garantir a convergência nas duas redes. A *loss function* adotada foi a MSE (equação (3)). Destaca-se que diversas configurações de arquiteturas foram testadas, sendo esta a escolhida pela sua boa performance.

Em particular para a PINN, explorou-se a sua capacidade de identificar os parâmetros desconhecidos do modelo [9]. Para tanto, os parâmetros do Modelo SIR foram incorporados como variáveis treináveis na rede, ajustadas durante a otimização e inicializadas como $\beta_0 = \gamma_0 = 0,01$. Além disso, assumiu-se uma distribuição uniforme para os dados de entrada, permitindo definir 200 pontos de colocação igualmente espaçados e contidos no interior do conjunto de treinamento. Destaca-se que, a *loss function* desta rede está definida na equação (10).

Para avaliar a performance do modelo, os dados originais foram comparados com as saídas da rede por meio de métricas que quantificam o erro. A Raiz do Desvio Quadrático Médio (*Root Mean Squared Deviation*, RMSD), o Erro Médio Absoluto (*Mean Absolute Error*, MAE) e o Erro Percentual Absoluto Médio (*Mean Absolute Percentage Error*, MAPE) [7].

5.1 Dados Sintéticos

A análise do desempenho ao longo das épocas revelou que a MLP convergiu após 792 épocas, com a *loss function* estabilizando nas fases finais, indicando saturação no aprendizado. Em contrapartida, a PINN manteve um decaimento consistente da perda, sugerindo convergência mais robusta. Ambas as componentes da *loss function* (dados e resíduo) evoluíram em sincronia, denotando exercer influência equivalente no treinamento da PINN, embora o resíduo tenha apresentado um comportamento mais estável. Os resultados da Tabela 1a evidenciam o desempenho superior da PINN em todas as métricas de avaliação, enquanto a Tabela 1b confirma a precisão na calibração dos parâmetros, pois assegurou que convergissem para valores muito próximos dos verdadeiros.

DOI: 10.5540/03.2026.012.01.0254

Tabela 1: Comparação de métricas de desempenho e parâmetros calibrados.

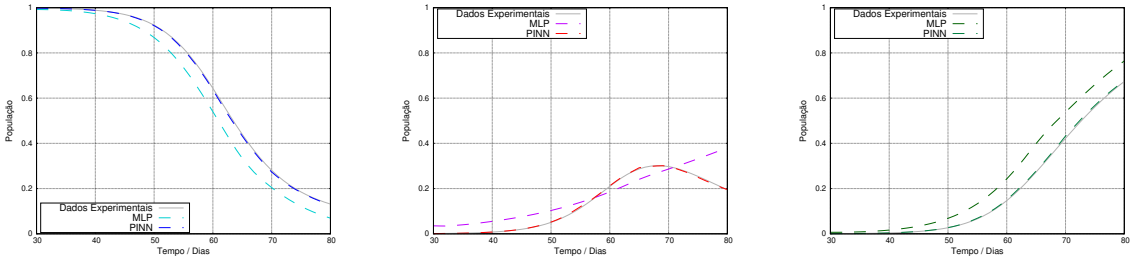
	RMSD	MAE	MAPE
PINN	0,0045	0,0034	0,02%
MLP	0,0678	0,0581	1,48%

(a) Métricas de desempenho.

	Calibração	Verdadeiros
β	0,3012	0,30
γ	0,0999	0,10

(b) Parâmetros calibrados.

A Figura 1 compara as previsões da MLP e PINN para as subpopulações do SIR. Ambas foram aplicadas ao conjunto de entrada, revelando a superioridade da PINN. Embora ajustes na arquitetura possam melhorar o desempenho da MLP, o estudo manteve ambas as redes o mais similares possíveis para melhor análise do impacto dos conhecimentos físicos incorporada na PINN. Testes com configurações alternativas são propostos para trabalhos futuros.



(a) Suscetíveis (b) Infectados (c) Recuperados

Figura 1: Inferência de cada Rede Neural. Fonte: Elaborada pelo autor.

5.2 Dados Reais

Nesta etapa, a PINN foi aplicada a um conjunto de dados referentes aos casos diários de infectados por COVID-19 na cidade de Recife, de domínio público e obtidos em [11]. Os dados exibiam um nível considerável de flutuação, indicando um ruído elevado. Para mitigar esse problema, foi aplicada uma *spline* cúbica para suavização [8]. Além disso, devido à insuficiência de dados para um treinamento eficaz, esse método foi usado para interpolar a curva suavizada e expandir o domínio. Se as observações do conjunto original possuíam um tamanho de passo $\Delta = 1$, este passou a ser $\Delta = 0,25$. A Figura 2 compara os dados originais com as curvas tratadas. Note-se que, a curva obtida difere da curva padrão prevista pelo Modelo SIR. Como o modelo que melhor reproduz a realidade é desconhecido, optou-se por incorporar o conhecimento físico por meio do Modelo SIR, escolha devida a sua robustez teórica e simplicidade interpretativa.

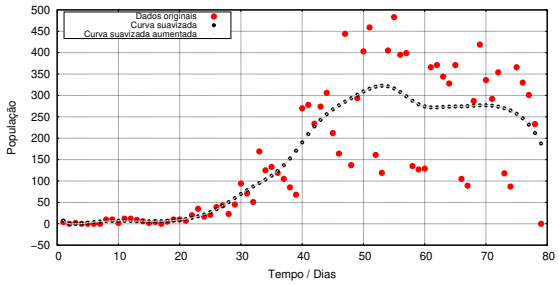


Figura 2: Dados experimentais de Recife. Fonte: Elaborada pelo autor.

As estratégias de preparação dos dados e a arquitetura dos testes anteriores foram mantidas,

com a única alteração sendo a adição de uma camada oculta. Vale destacar que os dados foram normalizados para o treinamento. Embora os parâmetros verdadeiros do Modelo SIR no cenário atual sejam desconhecidos, a PINN foi utilizada para identificá-los, partindo da mesma inicialização empregada anteriormente. Considerando que informações sobre as populações de suscetíveis e recuperados não estão disponíveis, apenas dados de infectados foram utilizados na minimização da *loss function*. Para compensar essa limitação, adicionou-se um termo de penalização correspondente à lei de conservação da população total, MSE_N, (equação (1)), garantindo que a saída da rede respeite essa restrição física. Desse modo, a *loss function* é composta pela soma de três termos: MSE_{Dados}, MSE_I (equação (7)) e MSE_N.

O desempenho, avaliado pelas métricas na Tabela 2, foi satisfatório, porém inferior ao obtido com dados sintéticos, devido a complexidade inerente aos dados reais, como o ruído. Tal fator não estava presente no cenário controlado. Quanto à calibração dos parâmetros, β e γ apresentaram uma trajetória ruidosa mas estável, apontando uma convergência adequada, obtendo $\beta = 0,2104$ e $\gamma = 0,0089$. A Figura 3 ilustra a inferência da PINN sobre os dados de treinamento, evidenciando o êxito da rede treinada. De fato, A PINN capturou os padrões dos dados e reproduziu com precisão a dinâmica da curva, exceto por uma diferença na magnitude de infectados.

Tabela 2: Comparação das métricas de desempenho.

	RMSD	MAE	MAPE
Dados de Recife	0,0217	0,0146	0,58%

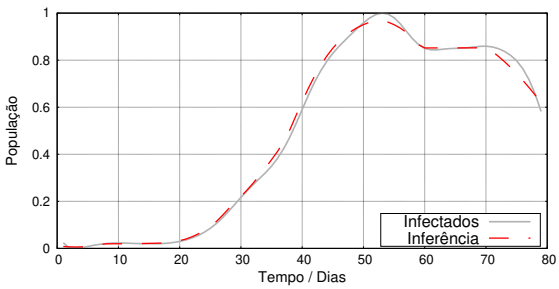


Figura 3: Inferência da PINN. Fonte: Elaborada pelo autor.

6 Considerações Finais

Neste trabalho foram comparadas duas abordagens de redes neurais aplicadas a dados epidemiológicos: a MLP, baseada exclusivamente em dados, e a PINN, que incorpora algum conhecimento físico adquirido *a priori*. Os resultados demonstraram que essa integração conferiu à PINN uma vantagem significativa, especialmente em cenários desafiadores com dados reais, ruidosos e escassos. Essa abordagem é especialmente relevante em situações de dados limitados, como no início de uma nova epidemia. Além disso, sua habilidade de inferir parâmetros desconhecidos amplia a compreensão da dinâmica de doenças, possibilitando a simulação de cenários diversos, ação fundamental para a formulação de políticas públicas.

Agradecimentos

Esse trabalho foi financiado em parte pela Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES). R.C.A. agradece ao CNPq, processo 306588/2022-6.

Referências

[1] S. Han, L. Stelz, H. Stoecker, Wang. L. e K. Zhou. "Approaching epidemiological dynamics of COVID-19 with physics-informed neural networks". Em: **Journal of the Franklin Institute** 361.6 (2024), p. 106671. DOI: https://doi.org/10.1016/j.jfranklin.2024.106671.

[2] M. J. Keeling e P. Rohani. **Modeling infectious diseases in humans and animals**. New Jersey: Princeton University Press, 2011. ISBN: 978-0-691-11617-4.

[3] W. O. Kermack e A. G. McKendrick. "A Contribution to the Mathematical Theory of Epidemics". Em: **Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character** 115.772 (1927), pp. 700–721. DOI: 10.1098/rspa.1927.0118.

[4] D. P. Kingma e J. Ba. "Adam: A Method for Stochastic Optimization". Em: **arXiv preprint arXiv:1412.6980** (2014). URL: https://arxiv.org/abs/1412.6980.

[5] J. Long, A. Q. M. Khaliq e K. M. Furati. "Identification and prediction of time-varying parameters of COVID-19 model: a data-driven deep learning approach". Em: **International Journal of Computer Mathematics** 98.8 (2021), pp. 1617–1632. DOI: 10.1080/00207160.2021.1929942.

[6] D. J. C. MacKay. **Information Theory, Inference, and Learning Algorithms**. Cambridge: Cambridge University Press, 2003. ISBN: 0521642981 9780521642989.

[7] S. Makridakis, S. Wheelwright e R. J. Hyndman. **Forecasting: Methods and Applications, 3rd Ed**. USA: John Wiley Sons, 1997. ISBN: 0-471-53233-9.

[8] W. H. Press, S. A. Teukolsky, W. T. Vetterling e B. P. Flannery. **Numerical Recipes in C: The Art of Scientific Computing**. 2nd. USA: Cambridge University Press, 1992. ISBN: 0521437148.

[9] M. Raissi, P. Perdikaris e G.E. Karniadakis. "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations". Em: **Journal of Computational Physics** 378 (2019), pp. 686–707. DOI: https://doi.org/10.1016/j.jcp.2018.10.045.

[10] M. Raissi, N. Ramezani e P. Seshaiyer. "On Parameter Estimation Approaches for Predicting Disease Transmission Through Optimization, Deep Learning and Statistical Inference Methods". Em: **Letters in Biomathematics** (2019). DOI: https://doi.org/10.30707/LiB6.2Raissi.

[11] G. L. Vasconcelos, G. C. Duarte-Filho, A. A. Brum, R. Ospina, F. A. G. Almeida e A. M. S. Macêdo. "Análise de curvas epidêmicas da Covid-19 via modelos generalizados de crescimento: Estudo de caso para as cidades de Recife e Teresina". Em: **SciELO Preprints** (2020). DOI: 10.1590/SciELOPreprints.690.

