



## Previsão de óbitos por Covid-19 no Brasil utilizando Redes Neurais Artificiais

Lucas Duarte Soares (Unioeste)  
[lucas.2012@alunos.utfpr.edu.br](mailto:lucas.2012@alunos.utfpr.edu.br)

Peterson Diego Kunh (UTFPR)  
[petersondiego83@gmail.com](mailto:petersondiego83@gmail.com)

Jairo Marlon Corrêa (UTFPR)  
[jairomarlon@gmail.com](mailto:jairomarlon@gmail.com)

José Airton Azevedo dos Santos (UTFPR)  
[professorjairton@gmail.com](mailto:professorjairton@gmail.com)

*Neste trabalho realizou-se previsões da série histórica do número acumulado de óbitos por Covid-19 no Brasil. Para isso utilizou-se de modelos baseados em redes neurais artificiais, sendo eles: Multilayer Perceptron (MLP) e Elman. As redes foram programadas em python com auxílio da bibliotecas keras e suas previsões foram comparadas entre si por intermédio do erro médio percentual absoluto (MAPE), tanto para amostra de treino quanto para amostra de validação. Na amostra de treino o MAPE obtido com ambos os modelos foi de 3,33%. Enquanto para a amostra de validação o MAPE obtido com o modelo MLP foi de 5,23% e com o modelo Elman foi de 4,74%. O modelo Elman forneceu melhores previsões em relação ao modelo MLP. Todavia não foi possível a extrapolação da série devido à pequena quantidade de dados disponíveis até a finalização deste trabalho.*

*Palavras-chave: Multilayer Perceptron, Elman, Inteligência Artificial, Machining Learning.*

## 1. Introdução

Segundo o Ministério da Saúde (2020), o Brasil possui, até as 19:20 horas do dia 19/05/2020, 1.179 (mil cento e setenta e nove) óbitos confirmados por Covid-19. Somando esse número ao acumulado desde o primeiro caso ocorrido em dezessete de março de 2020, o montante resulta no total de 17.971 (dezessete mil novecentos e setenta e um) óbitos confirmados pelo vírus. De acordo com dados da Organização Mundial da Saúde (OMS), o Brasil atinge pela primeira vez a marca superior de mil mortes em 24 horas, com isso o país entra no grupo composto por Estados Unidos (2.612), França (1.417) e China (1.290) que possuem mais de mil óbitos diários. Com o aumento aparentemente exponencial do número de óbitos por Covid-19 no Brasil, o presente artigo tem como objetivo testar modelos preditivos que possam modelar a série histórica e, se possível, projetar valores futuros. A previsão será realizada com modelos de inteligência artificial baseados em Redes Neurais Artificiais (RNA), sendo elas: *multilayer perceptron* (MLP) e Elman. Com as previsões é possível projetar o número acumulado de óbitos para verificar se as medidas atuais de combate a propagação do vírus e/ou tratamentos estão sendo eficazes. Realizar previsões sobre o vírus da pandemia já é usado pelo meio científico, ver: Fanelli e Piazza (2020); Liu *et al.* (2020); Yang *et al.* (2020).

O presente trabalho está dividido em: referencial teórico, onde é apresentado os conceitos de RNAs; materiais e métodos, no qual é analisado a série temporal de óbitos do Covid-19 no Brasil e explicado como foram realizadas as previsões; resultados, onde é apresentado os valores projetados e seus respectivos erros; e conclusões.

## 2. Referencial teórico

Predizer dados futuros oriundos de informações já conhecidas é uma das várias aplicações de Redes Neurais Artificiais (MANDIC;CHAMBERS, 2001). Outra aplicação bastante recorrente é de classificação no qual um dado é processado pela rede e, então, rotulado em uma determinada classe já conhecida (HAGAN et al., 2006). Segundo Haykin (1999), Redes Neurais Artificiais são técnicas computacionais que criam modelos matemáticos complexos baseados na estrutura neural humana e adquirem conhecimento através da experiência. Quando informações conhecidas pela rede estão na forma de série histórica, pode-se dar início ao seu treinamento para obter um modelo matemático que descreve o processo estocástico das observações temporais. A rede treinada conseguirá prever com eficiência os valores futuros da série histórica, desde que bons parâmetros tenham sido encontrados durante o treinamento.

Uma rede neural artificial é composta por um conjunto de neurônios artificiais que baseiam seu comportamento em um neurônio biológico (HAYKIN, 1999). De acordo com Guyton e Hall (2017), os neurônios se comunicam através da sinapse, que é a região onde ocorre contato físico entre eles. Esse processo é responsável pela transmissão de impulsos nervosos de um neurônio para outro. Segundo Fausett (1994), os impulsos recebidos por um neurônio em um instante de tempo são processados e quando atingem um limiar de ação (função de ativação) são disparados, produzindo uma substância neurotransmissora que flui do corpo celular para o axônio.

O neurotransmissor, conforme Guyton e Hall (2017), pode diminuir ou aumentar a polaridade da membrana pós-sináptica, inibindo ou excitando a geração dos pulsos elétricos no neurônio. Este processo depende de vários fatores físicos, como a geometria da sinapse e o tipo de neurotransmissor.

Antes de um neurônio transmitir informações para outros neurônios, ocorre uma análise prévia realizada pela função de ativação com o objetivo de definir se essa informação será transferida ou não. Além disso, a maioria das funções de ativação são necessárias para dar capacidade representativa de problemas do mundo real às redes neurais artificiais, por elas introduzirem uma componente de não linearidade. Entretanto surge uma grande dificuldade que, de acordo com Haykin (2000), a superfície da função objetivo da rede neural deixa de ser convexa, tornando sua otimização mais complexa. Exemplificando melhor, o método dos mínimos quadrados utilizado na regressão linear não é eficaz pela dificuldade em igualar as derivadas parciais da função objetivo em zero. Logo é necessário utilizar algoritmos eficientes para treinar os parâmetros da rede neural para gerar resultados satisfatórios.

Um conglomerado de neurônios transferindo ou retraindo sinais com auxílio de funções de ativação é considerado uma rede neural artificial. De acordo com Fausett (1994), uma rede pode ser dividida em 3 tipos de camadas, sendo elas: camada de entrada; camada intermediária e camada de saída. A primeira camada da rede neural artificial é definida como a camada de entrada, nela os dados brutos são apresentados à rede dando início o treinamento ou previsão. Já nas camadas intermediárias, também conhecidas como camadas ocultas, os dados são processados através das conexões ponderadas. Por último a camada de saída, onde o resultado é obtido procedente das informações inseridas e processadas pela rede.

Segundo Hagan et al. (2006), as camadas de entrada e saída podem conter diversos neurônios, porém elas são camadas únicas, ou seja, existem apenas uma de cada tipo-uma de entrada e

uma de saída. Diferentemente destas camadas, as do tipo intermediárias podem conter várias camadas, além de vários neurônios em cada uma delas.

Uma rede neural necessita ser treinada com grandes quantidades de informações para gerar resultados satisfatórios. O treinamento é feito através de um processo iterativo de ajustes aplicados aos seus pesos. Existem vários tipos de algoritmos de aprendizagem específicos para determinados modelos de redes neurais, nos quais diferem entre si principalmente pela maneira como os pesos são alterados a cada iteração do treinamento (Galushkin, 2007).

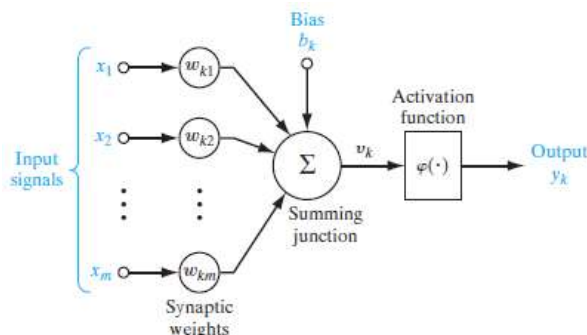
Os dois tipos de aprendizado mais utilizados pelas redes neurais são: Aprendizado supervisionado e aprendizado não supervisionado. O primeiro é empregado quando um agente externo indica à rede a saída (valor real) desejada para o padrão de entrada. Já o segundo, aprendizado não supervisionado, é utilizado quando esse agente externo não é requisitado para desempenhar sua função de comparar as saídas da rede com os valores desejáveis (Haykin, 1999).

Antes de iniciar o treinamento da rede neural para um conjunto de dados é de extrema importância definir seus parâmetros para obter um bom desempenho, tanto de formas computacionais quanto de resultados. Estes parâmetros são conhecidos como hiperparâmetros, sendo eles: número de camadas ocultas; números de neurônios em cada camada; funções de ativação; otimizadores e inicialização dos pesos e bias (Fausett, 1994; Haykin, 1999). Os otimizadores são algoritmos utilizados para alterar os pesos da rede a cada iteração com o objetivo de diminuir a função objetivo durante o treinamento.

Além de definir os parâmetros da rede, outro fator necessário é analisar os dados que serão inseridos nela. Caso os dados estejam correlacionados de alguma forma, como em uma série temporal, uma rede neural recorrente é mais indicada, pois, de acordo com Graves (2012), a ideia de seu surgimento foi para fazer uso de informações sequenciais. Segundo Mandic e Chambers (2001), redes neurais recorrentes (RNN) são um tipo de rede neural no qual a saída da iteração do passo anterior é utilizada como entrada da iteração atual. Enquanto em uma rede neural clássica, de acordo com Fausett (1994), todas as entradas e saídas são independentes uma da outra, ou seja, não é necessário que os dados sejam alimentados na rede em uma sequência predeterminada.

A representação gráfica de um neurônio de uma rede neural pode ser vista na Figura 1 onde  $x$  é a entrada,  $w$  é o peso da entrada,  $b$  é o Bias,  $\phi$  a função de ativação e  $y$  a saída.

Figura 1 - Representação gráfica de um neurônio artificial



Fonte - Haykin (1999)

A Figura 1 pode ser representada pelas Equações 1, 2, 3 e 4 (Haykin, 1999). As quatro equações foram utilizadas para desenvolver o código. Segundo Haykin (1999), o bias é utilizado para aplicar um efeito de transformação afim na combinação linear das entradas e pesos da rede no modelo da Figura 1.

$$u_k = \sum_{j=1}^m w_{kj} x_j \quad (1)$$

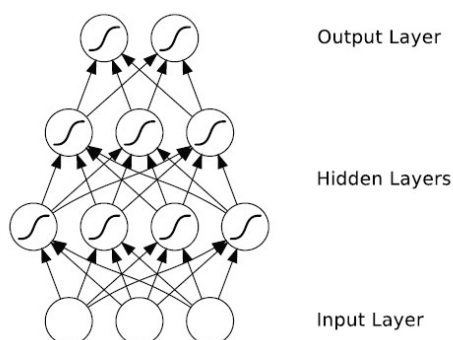
$$y_k = \varphi(u_k + b_k) \quad (2)$$

$$v_k = u_k + b_k \quad (3)$$

$$y_k = \varphi(v_k) \quad (4)$$

A Figura 2 mostra a representação gráfica de uma rede neural *multi layer perceptron* (MLP) com duas camadas intermediárias. Todos os neurônios dessa rede são ativados pela função *sigmoid*, com exceção da camada de entrada. Esta é uma rede *feedforward* pelo fato de todas as saídas dos neurônios serem no mesmo sentido, de baixo para cima.

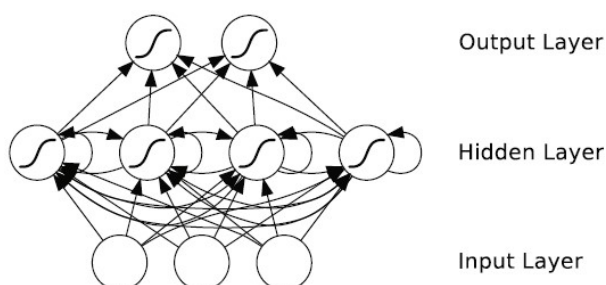
Figura 2 – rede MLP



Fonte - Graves (2012)

A Figura 3 mostra a representação gráfica de uma rede neural recorrente simples com uma camada intermediária. Todos os neurônios dessa rede são ativados pela função *sigmoid*, com exceção da camada de entrada. Esta é uma rede recorrente pelo fato de todas as saídas dos neurônios da camada intermediária não seguirem um único sentido.

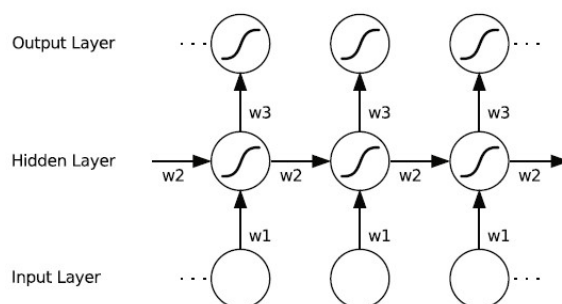
Figura 3 - rede neural recorrente



Fonte - Graves (2012)

A Figura 3 é muito carregada e de difícil compreensão. Logo, para efetuar uma análise minuciosa da rede para entender o seu funcionamento e, dessa forma, realizar sua representação matemática, é necessário realizar um processo chamado desdobramento. A Figura 4 mostra esse processo.

Figura 4 – Desdobramento da Rede recorrente



Fonte - Graves (2012)

A Figura 4 mostra a rede neural recorrente da Figura 3 desdobrada em uma rede completa. Desdobrar uma RNN significa desenrolar ela em sua sequência completa. Por exemplo, se a sequência que estamos tentando prever é uma palavra de 3 letras, a rede neural vai ser desdobrada em 3 camadas de rede neural, uma cada para cada letra.

Se o valor da entrada  $i$  no instante de tempo  $t$ ,  $a$  e  $b$  respectivamente as entradas  $j$  no instante  $t$  e a ativação do neurônio  $j$  no instante  $t$ . A Equação 5 mostra o comportamento do neurônio oculto (GRAVES, 2012).

$$a_h^t = \sum_{i=1}^I w_{ih} x_i^t + \sum_{h'=1}^H w_{h'h} b_{h'}^{t-1} \quad (5)$$

A Equação 6 mostra uma função de ativação não linear e diferenciável teta aplicado na Equação 1 da mesma maneira que seria para uma rede neural *multilayer perceptron* (GRAVES, 2012)

$$b_h^t = \theta_h(a_h^t) \quad (6)$$

A sequência completa das ativações ocultas pode ser calculada começando o  $t=1$  e de maneira recursiva aplicando as Equações 1 e 2, incrementando  $t$  a cada iteração. Nota-se que é necessário um valor de  $b$  no instante  $t=0$  para iniciar o algoritmo.

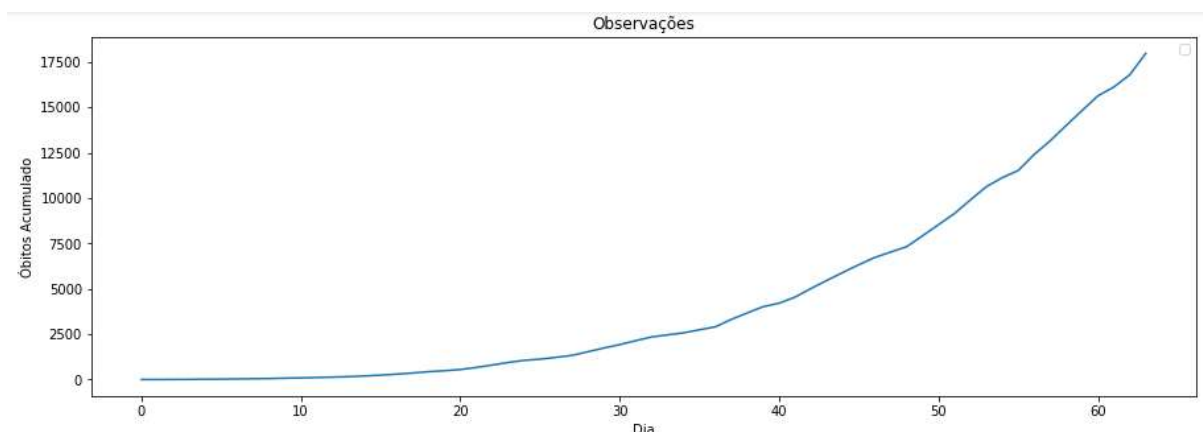
A Equação 7 mostra que é possível as entradas da rede neural recorrente para a camada de saída serem calculadas no mesmo instante tempo que as camadas ocultas de ativação.

$$a_k^t = \sum_{h=1}^H w_{hk} b_h^t \quad (7)$$

### 3. Materiais e métodos

A série temporal do número de óbitos confirmados pelo Ministério da Saúde no Brasil, utilizada neste trabalho, envolve 65 observações diárias realizadas entre o período de 17 de março de 2020 até 19 de maio do mesmo ano. A representação gráfica da série histórica pode ser vista na Figura 5, no qual o eixo das ordenadas é o valor acumulado do número de óbitos ao longo do tempo e o eixo das abscissas é cada observação temporal dado em formato diário. A planilha com os valores da série temporal pode ser encontrada no site do governo federal, com o seguinte endereço eletrônico: <https://covid.saude.gov.br/>.

Figura 5 – Série temporal do Covid-19 Brasil

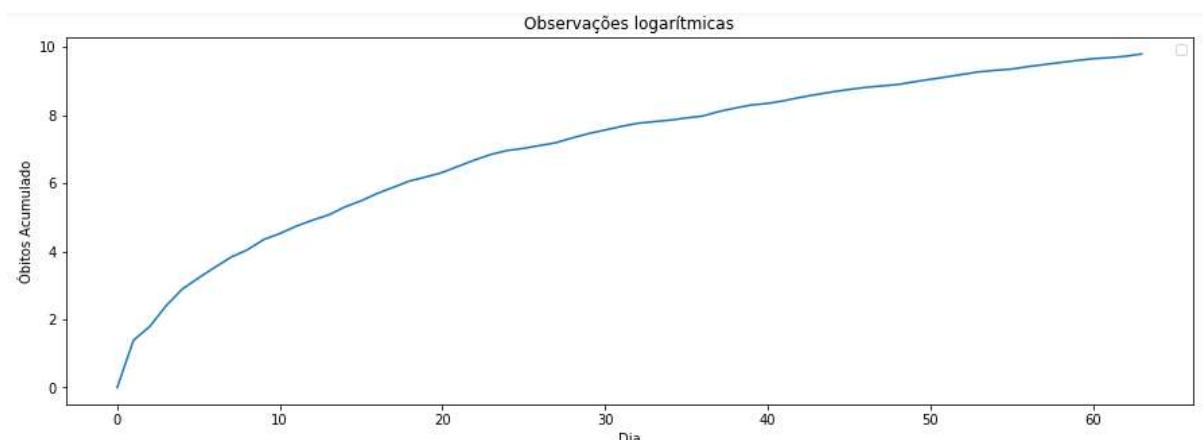


Fonte – Autores (2020)

Analisando a Figura 5 é verificado que a curva apresenta um comportamento exponencial, portanto uma transformação logarítmica é aplicada na série temporal visando facilitar o treinamento das redes neurais e, conseqüentemente, melhores previsões pelos modelos. As previsões fornecidas pelas RNAs estarão na escala logarítmica, assim sendo é necessário aplicar sua transformada inversa, ou seja, a função exponencial. A Figura 6 apresenta a série temporal logarítmica.



Figura 6 – Série temporal logarítmica do Covid-19 Brasil



Fonte – Autores (2020)

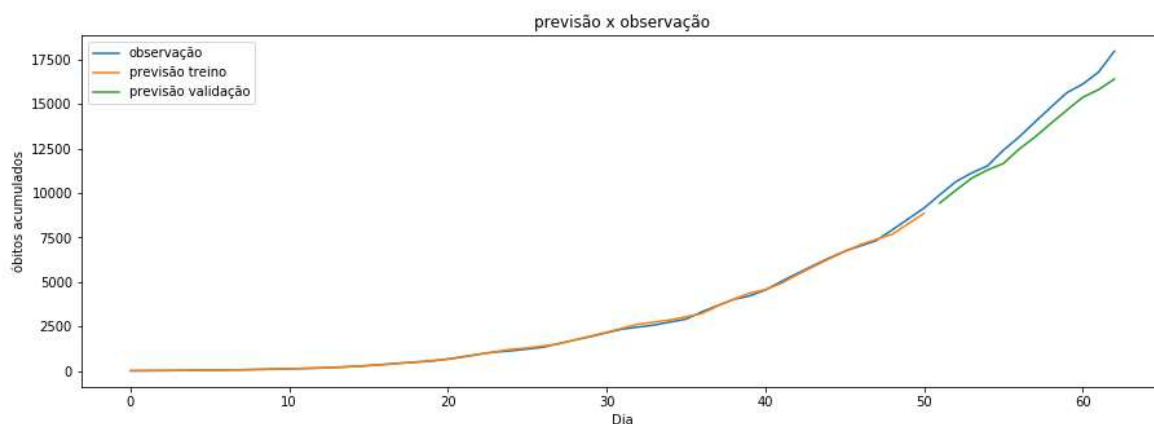
A série temporal do número de óbitos por Covid-19 no Brasil, formada por 65 observações, foi dividida em duas amostras. Com as primeiras 52 observações formou-se a amostra de treinamento para estimar os parâmetros dos modelos de previsão, realizada na modalidade um passo à frente. A amostra de validação foi formada pelas últimas 13 observações e utilizada para averiguar a eficiência preditiva dos modelos propostos e compará-los.

Os modelos neurais de previsão (MLP e Elman) utilizados no presente trabalho foram programados em *python* por meio da biblioteca Keras. Ambas as redes possuem um (1) neurônio na camada de entrada, cem (100) neurônios com função de ativação tangente hiperbólica na camada intermediária, e um (1) neurônio com função de ativação reLu na camada de saída. O treinamento das redes ocorreu por cinco mil (5.000) épocas e o algoritmo de aprendizado utilizado foi o gradiente descendente.

#### 4. Resultados

A primeira rede treinada foi a *multilayer perceptron* e o seu resultado pode ser verificado na Figura 7. O gráfico contém três curvas: a primeira (em azul) são os dados reais; a segunda (em laranja) são as previsões na amostra treino e a terceira (em verde) são as previsões na amostra de validação. A rede MLP conseguiu captar o processo estocástico gerador da série temporal de maneira considerável para amostra treino (erro MAPE 3,33%), já para a amostra de validação o resultado teve um decréscimo (erro MAPE 5,23 %).

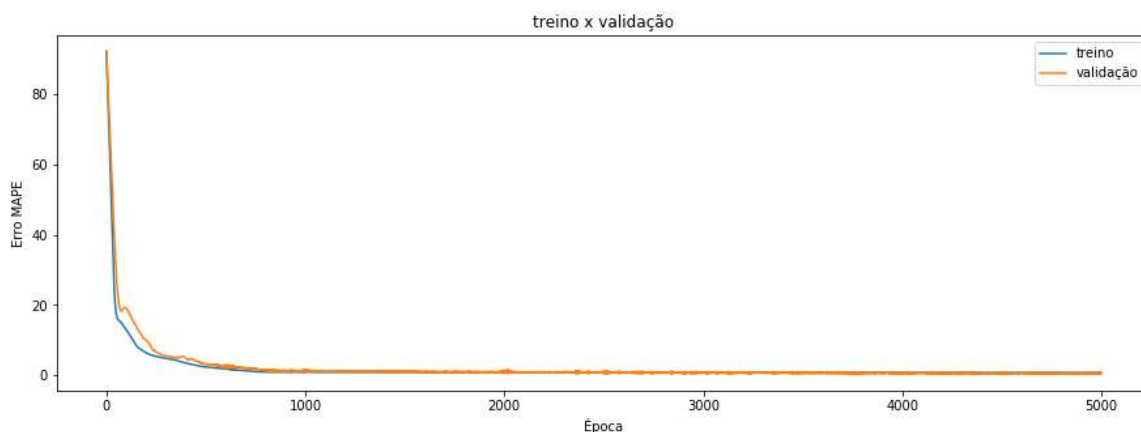
Figura 7- Previsões rede MLP



Fonte – Autores (2020)

A Figura 8 mostra o comportamento dos erros MAPE da amostra de treino e amostra de validação em função do número de épocas. Verifica-se que ambos os erros das previsões decaem exponencialmente tendendo à zero. Os erros convergem após mil (1.000) épocas, resultando em um gasto de tempo e computacional até o final da compilação do código.

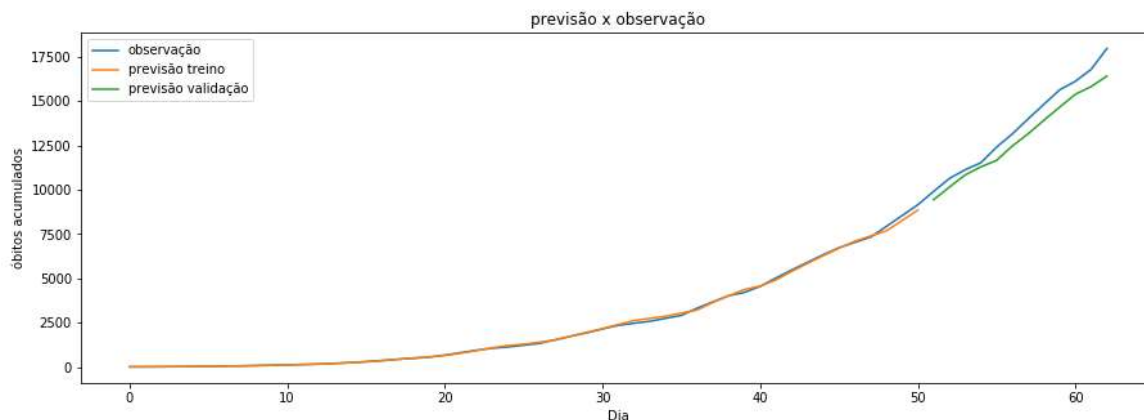
Figura 8 – Convergência do erro rede MLP



Fonte – Autores (2020)

A Segunda rede treinada foi a Elman e o seu resultado pode ser verificado na Figura 9. O gráfico contém três curvas: a primeira (em azul) são os dados reais; a segunda (em laranja) são as previsões na amostra treino e a terceira (em verde) são as previsões na amostra de validação. A rede Elman se comportou de forma similar à rede MLP. Na amostra treino da rede Elman o erro MAPE foi de 3,33%, enquanto para a amostra de validação o erro MAPE foi de 4,74 %

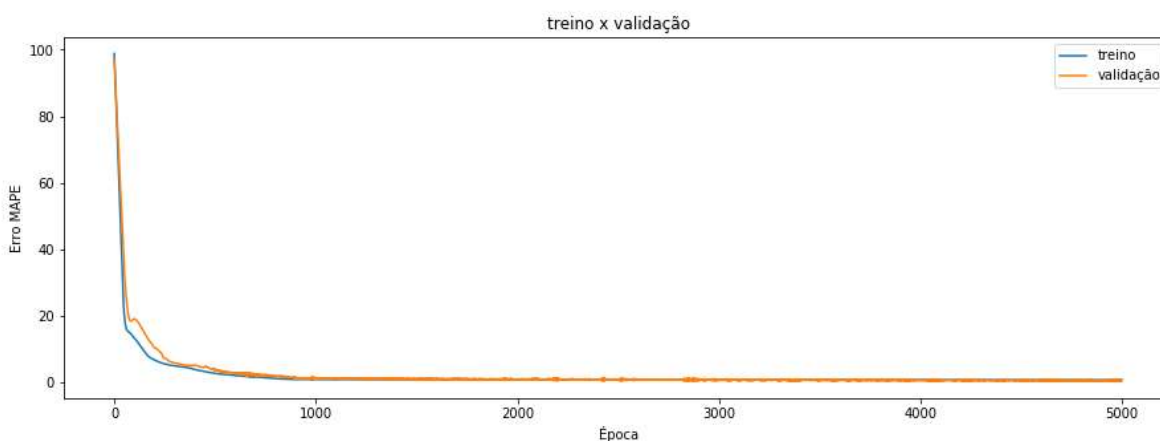
Figura 9 – Previsões rede Elman



Fonte – Autores (2020)

O comportamento dos erros Mape das amostras de treino e validação da rede Elman foi similar à rede MLP. A Figura 10 apresenta esse cenário.

Figura 10 - Convergência do erro rede Elman



Fonte – Autores (2020)

As previsões na amostra treino dos dois modelos apresentados são, aparentemente, iguais. Existe a possibilidade desse fenômeno ser resultado da quantidade de dados inseridos na rede para treinamento e/ou a magnitude deles serem bem discrepantes. Por ser uma série exponencial com poucos dados, o modelo recorrente preditivo (Elman) não foi capaz, supostamente, de captar a correlação temporal dos dados, por isso seu erro teve convergência idêntica da rede

MLP. A Tabela 1 mostra a comparação dos erros Mape das redes MLP e Elman tanto para amostra treino, quanto para amostra de validação.

Tabela 1 – Erros Mape

Arquitetura	Erro Mape treino	Erro Mape validação
MLP	3,3 %	5,23%
Elman	3,3 %	4,74%

Fonte – Autores (2020)

Analisando a Tabela 1, verifica-se que os erros foram idênticos na amostra treino para os dois modelos neurais de previsão, porém na amostra de validação o modelo de arquitetura Elman teve um desempenho melhor que o MLP.

## 5. Conclusão

Neste trabalho foi proposto dois modelos de previsão baseados em redes neurais artificiais (MLP e Elman) para prever o número de óbitos causados por Covid-19 no Brasil. Ambos os modelos tiveram o erro Mape baixo para dados conhecidos. Todavia para realizar projeções fora das amostras de treino e de validação, ou seja, extrapolação dos dados, não foi possível devido a quantidade escassa de dados.

O ponto a ser melhorado nos modelos de previsão propostos está na inserção de uma quantidade substancialmente maior de dados para alimentar as redes, o que não foi possível pelo curto tempo de espaço entre o primeiro óbito por Covid-19 e a confecção deste artigo. Portanto para trabalhos futuros fica como sugestão um maior conjunto de dados, bem como utilização de outras arquiteturas para realizar as previsões, por exemplo, *long short-term memory* e *gated recurrent unit*.

## REFERÊNCIAS

FANELLI, Duccio; PIAZZA, Francesco. **Analysis and forecast of COVID-19 spreading in China, Italy and France**. *Chaos, Solitons & Fractals*, v. 134, maio 2020.

FAUSETT, L.: **Fundamentals of Neural Networks: Architectures, Algorithms, and Applications**. New Jersey. Prentice Hall, 1 ed., 1994.

GALUSHKIN, A.: **Neural Networks Theory**. New York. Springer. 1 ed., 2007.

GRAVES, A.: **Supervised Sequence Labelling with Recurrent Neural Network**. New York. Springer. 1 ed., 2012.

GUYTON, A. C.; Hall, J. E.: **Tratado de Fisiologia Médica**. Rio de Janeiro. Elsevier. 13 ed., 2017.

HAGAN, M. T.; DEMUTH, H. B.; Beale, M. H.: **Neural Networks Design**. Martin Hagan. 2 ed., 2006.

HAYKIN, S.: **Neural networks and learning Machines**. In **Institute of Physics Conference Series**. New Jersey Pearson, 2 ed., Vol. 127, 1999.

LIU, Zhihua. MAGAL, Pierre. SEYDI, Ousmane. WEBB, Glenn. **Predicting the cumulative number of cases for the COVID-19 epidemic in China from early data**. arXiv preprint arXiv:2002.12298.

MANDIC, D. P., CHAMBERS, J. A.: **Recurrent Neural Networks for Prediction: Learning Algorithms, Architectures and Stability**. Wiley-Blackwell, New Jersey, 1 ed., Vol. 4, 2001.

MINISTÉRIO DA SAÚDE. **Corona vírus Brasil**: Disponível em: < <https://covid.saude.gov.br/>> Acesso em: 19 maio. 2020.

YANG, Zifeng; ZENG, Zhiqi; WANG, Ke; *et al.* **Modified SEIR and AI prediction of the epidemics trend of COVID-19 in China under public health interventions**. J Thorac Dis. 2020 Mar; 12(3): 165–174.  
doi:10.21037/jtd.2020.02.64.