

# An Approach Using Multiple MLP Neural Networks for Predicting the Brazilian Stock Market

Paulo Tasinaffo<sup>1</sup> and Luiz Dias

<sup>1</sup> Instituto Tecnológico de Aeronáutica (ITA), São José dos Campos/SP, Brazil; tasinaffo@ita.br

**Abstract:** The Brazilian stock market undergoes many fluctuations over time. This makes him highly unpredictable. To make matters worse, the 2020 pandemic and financial speculation made the Brazilian stock market even more unpredictable. Therefore, in this article, an approach based on Artificial Intelligence is proposed to carry out this prediction. For this purpose, multiple MLP neural networks will be coupled, using supervised learning with input/output training patterns through the NARMAX (Nonlinear Auto Regressive Moving Average with eXogenous input) method. A case study, using the actions of the three main companies in Brazil (PETROBRAS, EMBRAER and Vale do Rio Doce) are considered for the validation of the presented methodology. A numerical and computational comparison between the proposed Multiple method and the method using only one neural network is also presented.

**Keywords:** Universal Numerical Integrator (UNI); Nonlinear Auto Regressive Moving Average with eXogenous input (NARMAX); neural differential equations; Euler-Type Universal Numerical Integrator (E-TUNI); Runge-Kutta Neural Network (RKNN); Adams-Bashforth Neural Network (ABNN)

## 1. Introduction

As redes neurais se tornaram muito populares nas últimas décadas, pois são modelos matemáticos complexos e eficientes, que tentam imitar, no computador, o comportamento do cérebro humano. Existem basicamente dois tipos de redes neurais artificiais: redes neurais rasas e redes neurais profundas. As redes neurais rasas foram as maiores contribuições, desta área do conhecimento, durante a última metade do século XX. Por outro lado, as redes neurais profundas são as contribuições mais atuais, sobre o tema, neste século XXI.

Assim sendo, neste artigo é apresentado o modelo NARMAX utilizando Múltiplas Redes Neurais Artificiais (MRNA) com arquitetura Multi-Layer Perceptron (MLP) para a previsão de séries temporais do mercado brasileiro de ações. A abordagem de treinamento empregada é a aprendizagem supervisionada utilizando padrões de treinamento de entrada/saída. Sendo assim, serão realizados vários experimento computacionais, para verificar se a abordagem proposta, utilizando múltiplas redes neurais, é mais eficiente ou não do que a abordagem utilizando somente uma rede neural artificial.

Este trabalho está dividido em cinco seções. Na Seção 2, é realizada uma revisão bibliográfica das principais referências relacionadas ao tema proposto. Na Seção 3, é desenvolvido o modelo matemático detalhado do algoritmo neural múltiplo proposto. Ainda na Seção 3, uma descrição detalhada do funcionamento da bolsa de valores do mercado brasileiro é também apresentada. Na Seção 4, experimentos computacionais, baseados em dados do mundo real, comparam o desempenho do modelo proposto com aquele obtido com apenas uma rede neural. Por fim, a Seção 5 apresenta as principais conclusões do trabalho proposto.

## 2. Related Works

Em [1] e [2], é demonstrado, formalmente, que as redes neurais Multi-Layer Perceptrons (MLP) com uma camada interna são aproximadores universais de funções. Isto

**Citation:** Tasinaffo, P. M.; Dias, L. A. V. An Approach Using Multiple MLP Neural Networks for Predicting the Brazilian Stock Market. *Journal Not Specified* **2023**, *1*, 0. <https://doi.org/>

Received:

Revised:

Accepted:

Published:

**Copyright:** © 2023 by the authors. Submitted to *Journal Not Specified* for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

significa que elas podem, pelo menos teoricamente, resolver qualquer problema importante em Inteligência Artificial (e.g., previsão de séries temporais, neurocontrol, diagnóstico médico, classificação de padrões, processamento de imagens, entre outros). Em [3], um resumo das principais arquiteturas neurais rasas do século XX são discutidas em certa profundidade.

Entretanto, entre todas as possíveis aplicações de redes neurais artificiais, neste artigo, é considerado somente o problema de previsão de séries temporais. Nesta caso, a rede neural artificial pode ser vista como um modelo empírico de equações diferenciais autônomas e não-lineares [4]. Ainda em [4], é realizada uma classificação dos sistemas dinâmicos tratáveis, através de redes neurais artificiais, que utilizam a abordagem de aprendizagem supervisionada através de padrões de treinamento de entrada/saída. Segundo estes autores, são de três tipos as metodologias para modelagem empírica de sistemas dinâmicos não-lineares, a saber: (i) método NARMAX (Auto Regressive Moving Average with eXogenous input) [5] and [6], (ii) metodologia das derivadas médias [7,8], and [9], and (iii) metodologia das derivadas instantâneas [10–13], and [14]. Uma visão geral, da utilização de redes neurais artificiais e lógica fuzzy, na modelagem de sistemas dinâmicos e posterior aplicação em controle, pode também ver encontrado em [15].

Sendo assim, este artigo pretende realizar a simulação do mercado brasileiro de ações, através do método NARMAX, utilizando múltiplas redes neurais artificiais com arquitetura MLP. O algoritmo de treinamento que será utilizado, nos exemplos apresentados aqui, será o algoritmo Levenberg Marquardt [16] de 1994 e que está disponível no Toolbox de Redes Neurais Artificiais (ANN) do Matlab.

### 3. Mathematical Development

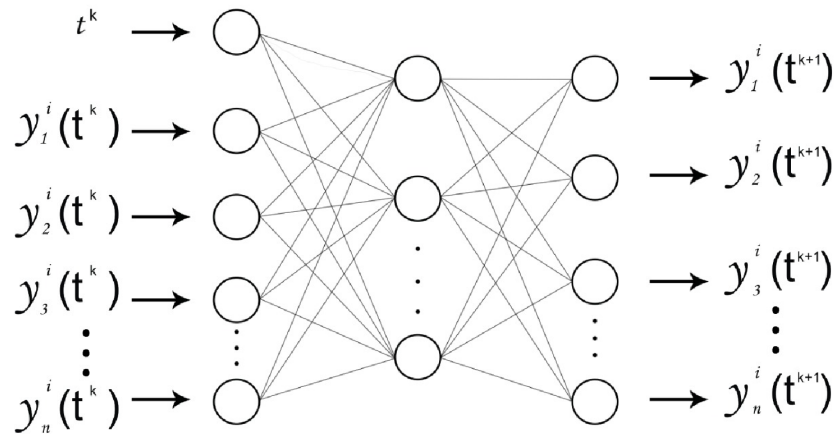
A Seção 3 vai aqui.

Aqui vem o texto 3: [1],[2], [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15] e [16].

O desenvolvimento matemático vai aqui.

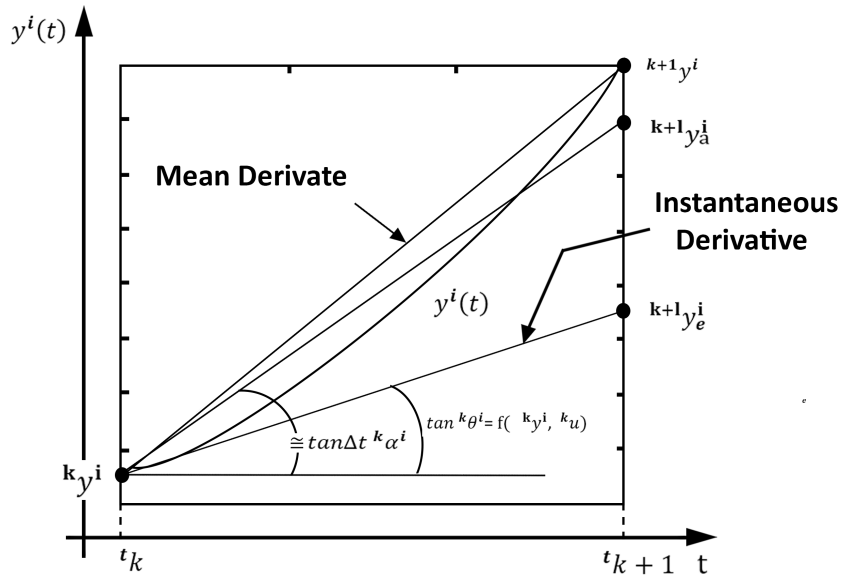
#### 3.1. Relationship Between the Universal Numerical Integrator (UNI) and the NARMAX Model

A Seção 3.1 vai aqui.

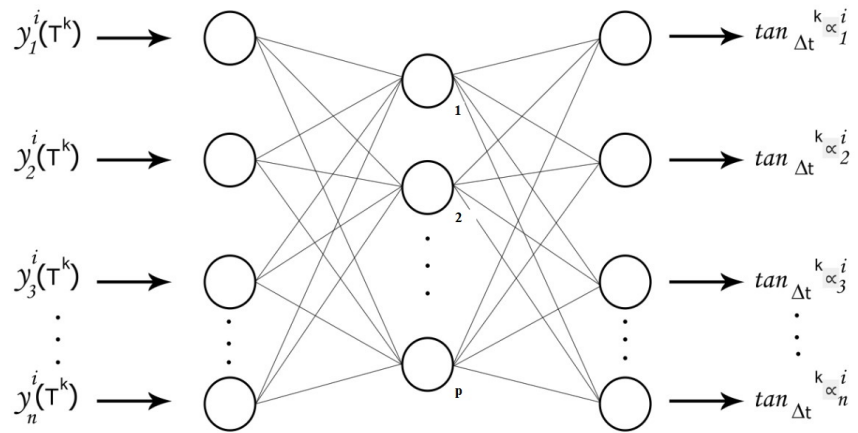


**Figure 1.** Esquema básico da metodologia NARMAX projetada numa arquitetura neural feed-forward (Source: see [5,6]).

$$\begin{cases} \dot{y}_1 = f_1(t, y_1, y_2, \dots, y_n), & y_1(a) = \eta_1 \\ \dot{y}_2 = f_2(t, y_1, y_2, \dots, y_n), & y_2(a) = \eta_2 \\ \vdots & \vdots \\ \dot{y}_n = f_n(t, y_1, y_2, \dots, y_n), & y_n(a) = \eta_n \end{cases} \quad (1)$$



**Figure 2.** Difference between mean derivative and instantaneous derivative functions (Source: see [7,8]).



**Figure 3.** A feed-forward neural network project with the concept of mean derivative functions.

Assim, um modelo NARMAX (sem entrada com ruído) do tipo Single-Input e Single-Output (SISO) é dado por [5,6]:

$$\hat{y}(k+1) = H[\varphi(k)] =$$

$$\hat{f}_{NN}[y(k) \cdots y(k-n_y) \quad u(k) \cdots u(k-n_u)] \quad (2)$$

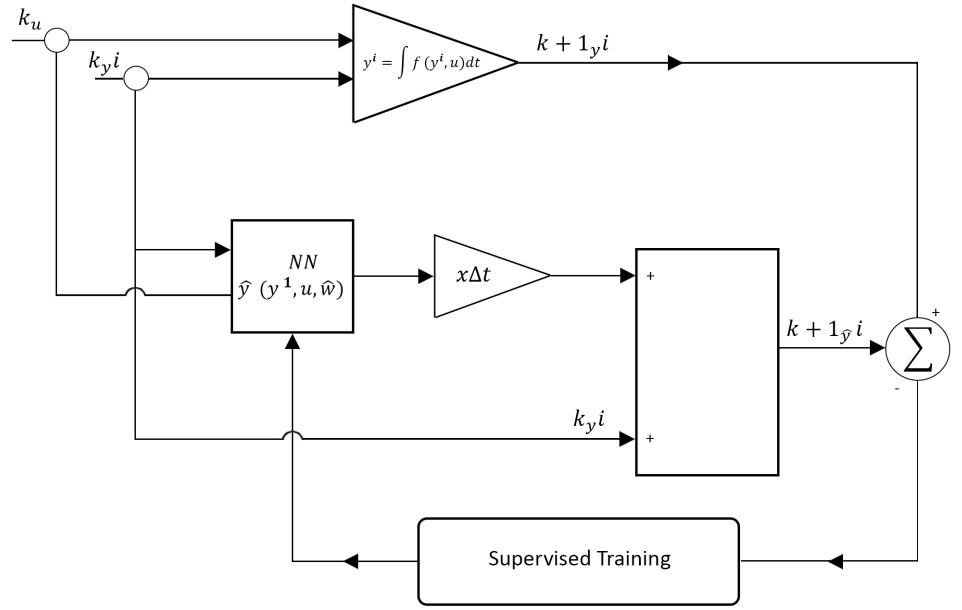
Ademais, o modelo NARMAX também pode ser facilmente estendido para o caso Multiple-Input e Multiple-Output (MIMO). Por exemplo, se  $\vec{y}(k-i) = [y_1(k-i) \ y_2(k-i) \cdots y_n(k-i)]^T$  e  $\vec{u}(k-j) = [u_1(k-j) \ u_2(k-j) \cdots u_n(k-j)]^T$  então, ter-se-ia o seguinte:

$$[\hat{\vec{y}}(k+1) \ \hat{\vec{y}}(k+2) \cdots \hat{\vec{y}}(k+n_{y_{out}})] =$$

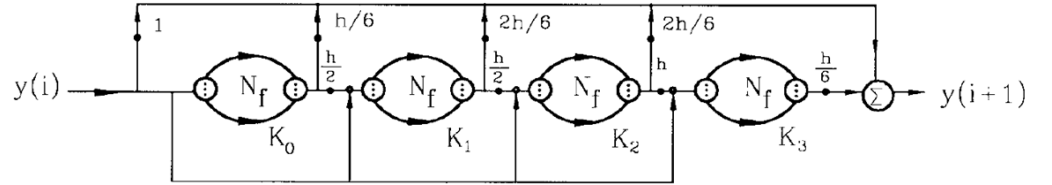
$$\hat{f}_{NN}[\vec{y}(k) \ \vec{y}(k-1) \cdots \vec{y}(k-n_{y_{in}}) \ \vec{u}(k) \ \vec{u}(k-1) \cdots \vec{u}(k-n_u)]^T \quad (3)$$

Existem também a metodologia das derivadas médias, que é uma alternativa ao modelo NARMAX [7–9]. Este tipo de integrador acopla uma rede neural feed-forward ao integrador de Euler de primeira ordem e como esquematizado pelas equações (4) e (5).

$$^{k+1}y^i = \tan_{\Delta t}^k \alpha^i \cdot \Delta t + ^k y^i \quad (4)$$



**Figure 4.** Esquema básico de uma rede feed-forward projetada internamente no integrador Runge-Kutta 4-5 (Source: see [11]).



**Figure 5.** Esquema básico de uma rede feed-forward projetada internamente no integrador Adams-Bashforth 4 (Source: see [12]).

onde  $^{k+1}y^i = [^{k+1}y_1^i \ ^{k+2}y_2^i \ \dots \ ^{k+1}y_n^i]^T$ ,  $\tan_{\Delta t}^{k+1}\alpha^i = [\tan_{\Delta t}^{k+1}\alpha_1^i \ \tan_{\Delta t}^{k+1}\alpha_2^i \ \dots \ \tan_{\Delta t}^{k+1}\alpha_n^i]^T$  and  $^k y^i = [^k y_1^i \ ^k y_2^i \ \dots \ ^k y_n^i]^T$ .

$$\tan_{\Delta t}^k \alpha_j^i = \frac{^{k+1}y_j^i - ^k y_j^i}{\Delta t} \quad (5)$$

A única Rede Neural Runge-Kutta (RKNN) que está disponível na literatura é a de ordem 4-5 [11]:

$$y_{n+1} = y_n + \frac{h}{6} \cdot (k_1 + 2 \cdot k_2 + 2 \cdot k_3 + k_4) \quad (6)$$

where,

$$k_1 = N_f(y_n; w)$$

$$k_2 = N_f(y_n + \frac{h}{2} \cdot k_1; w)$$

$$k_3 = N_f(y_n + \frac{h}{2} \cdot k_2; w)$$

$$k_4 = N_f(y_n + h, y_n + h \cdot k_3; w)$$

$$N_f(\cdot, \cdot; w) \cong \dot{y} = f(y(t))$$

Alternativamente, existe também a Rede Neural Adams-Bashforth (ABNN) que também está disponível na literatura e é dada por [12]:

$$y(t_{n+1}) = y(t_n) + \frac{h}{24} \cdot (55 \cdot f_n - 59 \cdot f_{n-1} + 37 \cdot f_{n-2} - 9 \cdot f_{n-3}) \quad (7)$$

### 3.2. Detailed Description of the Brazilian Financial Stock Market

Aqui vai a Seção 3.2.

### 3.3. Methodology Using Only One MLP Neural Network

Aqui vai a Seção 3.3.

### 3.4. The Proposed Method Using Multiple Neural Networks with MLP Architecture

Aqui vai a Seção 3.4.

## 4. Results and Analysis

Aqui vai a Seção de Resultados e Experimentos.

### 4.1. Simple Method

Aqui vai a Seção 4.1.

### 4.2. Compound Method

Aqui vai a Seção 4.2.

### 4.3. Numerical and Computational Comparisons Between the Two Proposed Methodologies

Aqui vai a Seção 4.3.

## 5. Conclusion

Aqui vai a Conclusão.

**Author Contributions:** Methodology, software, and writing was made by P. T.; supervision and project administration, L. D. All authors have read and agree to the published version of the manuscript.

**Funding:** The authors thank the Brazilian Aeronautics Institute of Technology (Instituto Tecnológico de Aeronáutica - ITA); the Casimiro Montenegro Filho Foundation (Fundação Casimiro Montenegro Filho - FCMF); and the Brazilian Enterprise Ecosystema Negócios Digitais Ltda for their support and infrastructure, which motivate the challenges and innovations of this research project.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Acknowledgments:** I would like to thank Professors and great friends Atair Rios Neto and Adilson Marques da Cunha for their valuable tips for improving this article. Finally, I would also like to thank the valuable improvement tips given by the good reviewers of this journal. The authors of this article would also like to thank God for making all of this possible.

**Conflicts of Interest:** The authors declare no conflict of interest.

## Abbreviations

The following abbreviations are used in this manuscript:

ABNN	Adams-Bashforth Neural Network
E-TUNI	Euler-Type Universal Numerical Integrator
NARMAX	Nonlinear Auto Regressive Moving Average with eXogenous input
MLP	Multi-Layer Perceptron
PCNN	Predictive-Corrector Neural Network
RBF	Radial Basis Function
RKNN	Runge-Kutta Neural Network
SVM	Support Vector Machine
UNI	Universal Numerical Integrator

## References

1. Cybenko, G. *Continuous Valued Networks with Two Hidden Layers Are Sufficient*. University of Illinois at Urbana-Champaign: Center for Supercomputing Research and Development, 1988.
2. Hornik, K.; Stinchcombe, M.; White, H. Multilayer feedforward networks are universal approximators. *Neural Networks* **1989**, *2*(5), 359–366.
3. Haykin, S. *Neural Networks: A Comprehensive Foundation*. Publisher: Prentice-Hall, Inc., New Jersey, USA, 1999.
4. Tasinaffo, P. M.; Gonçalves, G. S.; Cunha, A. M.; Dias, L. A. V. An introduction to universal numerical integrators. *Int. J. Innov. Comput. Inf. Control* **2019**, *15*(1), 383–406.
5. Billings, S. A.; Chen, S.; Koreberg, M. J. Identification of MIMO non-linear systems using forward-regression orthogonal estimator. *Int. J. Control* **1989**, *49*(6), 2157–2189.
6. Chen, S. and Billings; S. A. Neural networks for nonlinear dynamic system modelling and identification. *Int. J. Control* **1992**, *56*(2), 319–346.
7. Tasinaffo, P. M. Estruturas de Integração Neural Feedforward Testadas em Problemas de Controle Preditivo. Doctoral Thesis, INPE-10475-TDI/945, São José dos Campos/SP, Brazil, 2003.
8. Tasinaffo, P. M.; Rios Neto, A. Mean derivatives based neural Euler integrator for nonlinear dynamic systems modeling. *Learning and Nonlinear Models* **2005**, *3*(2), 98–109.
9. de Figueiredo, M. O.; Tasinaffo, P. M.; Dias, L. A. V. Modeling autonomous nonlinear dynamic systems using mean derivatives, fuzzy logic and genetic algorithms. *Int. J. Innov. Comput. Inf. Control* **2016**, *12*(5), 1721–1743.
10. Vidyasagar, M. *Nonlinear Systems Analysis*. Publisher: Prentice-Hall, Inc., Electrical Engineering Series, New Jersey, USA, 1978.
11. Wang, Y.-J.; Lin, C.-T. Runge-Kutta neural network for identification of dynamical systems in high accuracy. *IEEE Transactions on Neural Networks* **1998**, *9*(2), 294–307.
12. Tasinaffo, P. M.; Rios Neto, A. Adams-Bashforth neural networks applied in a predictive control structure with only one horizon. *Int. J. Innov. Comput. Inf. Control* **2019**, *15*(2), 445–464.
13. Chen, R. T. Q.; Rubanova, Y.; Bettencourt, J.; Duveand, D. Neural ordinary differential equations. In Proceedings of the 32nd Conference on Neural Information Processing Systems (NeurIPS), Montréal, Canada, 2018, 1–19.
14. Uçak, K. A Runge-Kutta MLP neural network based control method for nonlinear MIMO systems. In Proceedings of the 6th International Conference on Electrical and Electronics Engineering (ICEEE), Istanbul, Turkey, 2019, 186–192.
15. Spooner, J. T.; Maggiore, M.; Ordóñez, R.; Passino, K. M. *Stable Adaptive Control and Estimation for Nonlinear Systems Neural and Fuzzy Approximator Techniques*. Publisher: Wiley-Interscience, New York, USA, 2002.
16. Hagan, M. T.; Menhaj, M. B Training feedforward networks with the Marquardt algorithm. *IEEE Transactions on Neural Networks* **1994**, *5*(6), 989–993.

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.