Previsão Inteligente de Séries Temporais Baseada em Redes *Neuro-Fuzzy* *Takagi-Sugeno* Evolutivas Distribuídas

Selmo Eduardo Rodrigues Júnior*
Ginalber Luiz de Oliveira Serra**

* Instituto Federal de Educação, Ciência e Tecnologia, Imperatriz, MA
(e-mail: selmo.junior@ifma.edu.br).

** Instituto Federal de Educação, Ciência e Tecnologia, São Luís, MA
(e-mail: ginalber@ifma.edu.br)

Abstract: This paper proposes an evolving methodology based on Neuro-Fuzzy Takagi-Sugeno networks (NF-TS) distributed for time series forecasting. This proposed approach extracts the unobservable components from time series. Then, the forecast is performed separately for each component, and there is an NF-TS associated with each of these patterns, a fact that highlights the distributed characteristics of proposed approach. The NF-TS are still evolving, i.e., using the data of components to adapt and adjust their structure, where the amount of fuzzy rules of that network can increase or be reduced according to the behavior of components. The method used to extract the components is based on Singular Spectral Analysis technique. The proposed methodology was evaluated and compared with other similar techniques applied for series forecasting, presenting promising results.

Resumo: Esse trabalho propõe uma metodologia evolutiva baseada em redes *Neuro-Fuzzy Takagi-Sugeno* (NF-TS) distribuídas para a previsão de séries temporais. Essa abordagem proposta extrai as componentes não-observáveis a partir de uma série temporal. Em seguida, a previsão de cada componente separadamente é realizada, sendo que há uma NF-TS associada a cada um desses padrões, fato que destaca a característica distribuída da proposta. As NF-TS são ainda evolutivas, ou seja, usam os dados dessas componentes para adaptar e ajustar sua estrutura, sendo que a quantidade de regras *fuzzy* dessa rede pode aumentar ou ser reduzida conforme o comportamento das componentes. O método utilizado para extrair as componentes é baseado na técnica de Análise Espectral Singular. A metodologia proposta foi avaliada e comparada com outras técnicas semelhantes aplicadas para a previsão de séries, apresentando resultados promissores.

Keywords: Time Series Forecasting; Evolving Systems, Neuro-Fuzzy Takagi-Sugeno; Distributed; Unobservable Components.

Palavras-chaves: Previsão de Séries Temporais; Sistemas Evolutivos; *Neuro-Fuzzy Takagi-Sugeno*; Distribuída; Componentes Não-Observáveis.

1. INTRODUÇÃO

Devido sobretudo ao avanço tecnológico, a quantidade de dados disponíveis de diversas fontes e naturezas vem crescendo exponencialmente. Uma categoria específica de dados são aqueles que estão organizados de forma sequencial no tempo, cuja denominação é série temporal. Um dos objetivos mais importantes ao se trabalhar com séries temporais é a previsão, que consiste em projetar o comportamento futuro dos dados com base em informações atuais e históricas (Sagheer and Kotb, 2019). Os resultados de previsão são importantes para que especialistas possam fazer análises para dar suporte ao processo de tomada de decisão em diversas aplicações.

Há muitas pesquisas que envolvem aplicações de redes neurais para previsão de séries temporais, conforme apresentado em Tealab (2018). Em Wang et al. (2016), os autores desenvolveram uma arquitetura que combina redes neurais recorrentes para previsão de índices de mercado financeiro. Já em Wang et al. (2013), um modelo híbrido que busca a integração das vantagens do ARIMA e das redes neurais na modelagem dos comportamentos lineares e não-lineares de séries temporais é apresentado com bons resultados. Além dessas propostas, um modelo que considera ao mesmo tempo uma rede neural para estruturas de séries temporais lineares e não-lineares é desenvolvido em Yolcu et al. (2013). Essa rede é treinada usando o algoritmo de otimização de partículas (PSO). Os autores mostram que o modelo proposto permite obter melhores resultados em comparação com alguns modelos tradicionais.

* Programa de Pós-Graduação em Engenharia de Eletricidade/UFMA.

Figura 1. Estrutura da Metodologia Proposta Baseada em Redes *Neuro-Fuzzy Takagi-Sugeno* Evolutivas Distribuídas.

A decomposição de séries temporais em componentes mais simples vem sendo considerada em muitas pesquisas, obtendo promissores resultados. No trabalho de Joo and Kim (2015), os autores desenvolveram uma abordagem que decompõe a série original em componentes de tendência e outras variações, usando um modelo baseado em filtragem de *Wavelets* para realizar a previsão separadamente de cada componente. Outro trabalho que segue essa linha está em Nguyen and Novak (2019). Nesse caso, a série temporal é decomposta em componentes cíclicas, de tendência, sazonais e uma componente irregular, sendo que o resultado de previsão é a combinação da previsão individual de cada componente. As componentes de tendência, cíclicas e sazonais são representadas por modelos baseados em lógica *fuzzy*, enquanto que a componente irregular usa-se um modelo tradicional *Box-Jenkins*. Em situações em que os dados são gerados a grande velocidade, são complexos e não-lineares, o paradigma dos sistemas em evolutivos pode ser aplicado. Neste paradigma, os modelos desenvolvidos podem modificar sua estrutura, ou seja, são capazes de adaptar sua arquitetura interna de acordo com o comportamento dos dados. Os modelos evolutivos são aplicados em situações em que modelos puramente adaptativos não são suficientes para representar a dinâmica dos dados (Angelov et al., 2018; Rodrigues Júnior and Serra, 2017).

À vista desse cenário apresentado, este artigo propõe uma metodologia baseada em um conjunto de redes *Neuro-Fuzzy Takagi-Sugeno* (NF-TS) evolutivas para previsão distribuída de uma série temporal, considerando as componentes não-observáveis extraídas por um método baseado em Análise Singular Espectral. Cada componente não-observável extraída possui uma NF-TS associada para re-

alizar sua previsão individual. Em seguida, as previsões de todas as componentes são agrupadas para resultar na previsão da série. A contribuição principal desse trabalho está na característica distribuída da metodologia para previsão de uma série temporal, considerando NF-TS evolutivas para cada componente não-observável extraída.

2. METODOLOGIA PARA PREVISÃO DISTRIBUÍDA DE SÉRIES TEMPORAIS

A metodologia proposta neste trabalho está representada pela Figura 1. O primeiro passo dessa abordagem é extrair padrões ou componentes não-observáveis da série temporal. Para cada padrão extraído, há um modelo associado, baseado em uma estrutura *Neuro-Fuzzy Takagi-Sugeno* (NF-TS), que tem o papel de prever observações para essa componente específica. Sendo assim, diz-se que estas NF-TS se caracterizam como distribuídas. As NF-TS são evolutivas, ou seja, adicionam ou removem regras *fuzzy* visando adaptar-se de forma *online* aos dados que chegam a todo instante. Por fim, tendo em mãos as previsões de cada componente efetuadas pelas NF-TS, estes resultados são agrupados para obter as previsões da série temporal original.

2.1 Extração das Componentes Não-Observáveis

As componentes não-observáveis são padrões ocultos nos dados das séries temporais. Esses padrões são mais simples e possibilitam bons resultados para previsão (Abdollahzade et al., 2015; Nguyen and Novak, 2019). Vários métodos podem extrair essas componentes, como pode

ser encontrado nos trabalhos de L. Wu and Yang (2016), Bergmeir et al. (2016) e Abdollahzade et al. (2015). Nesse artigo, utiliza-se uma versão recursiva do método de Análise Singular Espectral, inspirado nos trabalhos de Abdollahzade et al. (2015) e Rodrigues Júnior and Serra (2017). O procedimento para extração das componentes não-observáveis é descrito nas subseções a seguir.

Condições Iniciais Inicialmente, três parâmetros devem ser definidos: w , que é uma janela de dados da série temporal para efetuar o procedimento das condições iniciais; l , que indica a dimensão da matriz de covariância do método; e n , que determina a quantidade de componentes não-observáveis extraídas da série temporal. Seja y_k uma observação da série temporal no instante k . Forma-se uma janela inicial $\{y_k | k = 1, 2, \dots, w\}$, contendo as primeiras w observações da série. A partir dessa janela, a matriz trajetória M , que representa esses dados da série temporal em formato matricial, pode ser construída conforme a seguir:

$$M = \begin{bmatrix} y_1 & y_2 & \cdots & y_{\zeta_w} \\ y_2 & y_3 & \cdots & y_{\zeta_w+1} \\ \vdots & \vdots & \ddots & \vdots \\ y_l & y_{l+1} & \cdots & y_w \end{bmatrix}, \quad M \in \mathbb{R}^{l \times \zeta_w}, \quad (1)$$

sendo que $\zeta_w = w - l + 1$ é um índice para observações da série temporal. A partir da matriz trajetória, as condições iniciais desse método finalizam com a determinação da matriz de covariância inicial S_w das w primeiras observações da série temporal, como segue:

$$S_w = MM^T, \quad S_w \in \mathbb{R}^{l \times l}. \quad (2)$$

Atualização Recursiva da Matriz de Covariância Com a obtenção da matriz de covariância inicial S_w , esta passa a ser atualizada recursivamente para os instantes $k = w + 1, w + 2, \dots$. Para cada nova observação da série temporal y_k , o parâmetro ζ_k deve ser incrementado em uma unidade. Após incrementar esse parâmetro, forma-se o seguinte vetor de atualização q_k :

$$q_k = [y_{\zeta_k} \ y_{\zeta_k+1} \ \cdots \ y_k]^T, \quad q_k \in \mathbb{R}^{l \times 1}, \quad (3)$$

onde y_{ζ_k} até y_k corresponde a uma janela de dados com tamanho fixo que se desloca ao longo do tempo, contendo as últimas l observações da série inclusive a atual. Usando o vetor q_k , é possível atualizar recursivamente a matriz de covariância de acordo com:

$$S_k = \frac{k-1}{k} S_{k-1} + \frac{1}{k} q_k q_k^T, \quad S_k \in \mathbb{R}^{l \times l}. \quad (4)$$

Nota-se que S_k representa toda a informação histórica da série temporal em uma matriz de dimensão fixa.

Decomposição em Valores Singulares O próximo passo baseia-se na decomposição em valores singulares da matriz S_k , para $k = w + 1, w + 2, \dots$. Para cada k , são calculados l autovalores e autovetores. Os autovalores e seus autovetores ortonormais associados são representados respectivamente por:

$$\begin{aligned} \rho_k^1 \geq \rho_k^2 \geq \cdots \geq \rho_k^l \geq 0 \quad (\text{autovalores}), \\ p_k^1, p_k^2, \dots, p_k^l \in \mathbb{R}^{l \times 1} \quad (\text{autovetores}). \end{aligned} \quad (5)$$

À vista disso, a decomposição da observação da série temporal y_k é definida por:

$$y_k = \theta_k^1 + \theta_k^2 + \cdots + \theta_k^d, \quad \text{para} \quad \theta_k^i = \bar{p}_k^i q_k^T p_k^i, \quad i = 1, \dots, d, \quad (6)$$

onde d é a quantidade de autovalores não-nulos e $\bar{p}_k^i$ é o último elemento do autovetor p_k^i (índice l do autovetor).

Agrupamento O agrupamento consiste em distribuir o conjunto de índices da Eq. 6 $\{1, \dots, d\}$ em n subconjuntos disjuntos de índices denotados por $I_1, I_2, \dots, I_n$, lembrando que n foi definido nas condições iniciais. Seja $I = i_h, \dots, i_j$ um desses subconjuntos genéricos de índices. Dessa forma, o novo termo associado nesse caso é $\theta_k^I = \theta_k^{i_h} + \cdots + \theta_k^{i_j}$. Sintetizando, faz-se esse procedimento para os n subconjuntos definidos. Com isso, é possível reescrever a Eq. 6 da seguinte forma:

$$y_k = \theta_k^{I_1} + \theta_k^{I_2} + \cdots + \theta_k^{I_n}. \quad (7)$$

Componentes Não-Observáveis Por fim, cada termo da Eq. 7 é associado a uma componente não-observável, ou seja:

$$c_k^j = \theta_k^{I_j}, \quad \text{para } j = 1, \dots, n, \quad (8)$$

onde c_k^j é a j -ésima componente no instante k . Pelas equações 7 e 8, é possível inferir que:

$$y_k = c_k^1 + c_k^2 + \cdots + c_k^n, \quad (9)$$

isto é, a soma dessas componentes é exatamente igual a observação no mesmo instante da série temporal.

2.2 Treinamento das Redes Neuro-Fuzzy Distribuídas

Para cada componente extraída da série, há um modelo NF-TS para realizar a previsão separadamente. Essas redes são geradas no instante $k = w + 1$ (após as condições iniciais do bloco de extração das componentes) e depois passam a ser atualizadas de forma evolutiva.

A fim de realizar a etapa de aprendizagem das n NF-TS, essa pesquisa apresenta uma abordagem baseada no método ALMMo-1 (*Autonomous Learning of Multi-Model Systems Version 1*), proposto em Angelov et al. (2018). O método ALMMo-1 se caracteriza por ser recursivo e ser totalmente orientado a dados. A estrutura de aprendizagem é baseada em regiões no espaço de dados denominadas *clouds*, que se assemelham com os *clusters*. Entretanto, os *clouds* são não-paramétricos, livres de restrições externas e não possuem uma forma bem definida, tendo o papel de representar diretamente as propriedades do conjunto local das amostras de dados observadas (Angelov et al., 2018). Cada NF-TS irá possuir seus próprios *clouds*, sendo que é possível associar a cada *cloud* uma regra *fuzzy*. Para cada instante k e para cada NF-TS individual, é necessário formar uma *data stream* que contém dados das componentes extraídas em instantes anteriores, conforme a seguir:

$$d_k^j = [c_{k-1}^j \ c_{k-2}^j \ \cdots \ c_{k-s}^j], \quad d_k^j \in \mathbb{R}^{1 \times r}, \quad j = 1, \dots, n, \quad (10)$$

onde d_k^j representa o *data stream* da componente j e os dados usados pela NF-TS j ($j = 1, \dots, n$), s representa um instante específico da série temporal que tenha boa autocorrelação com o instante atual e r corresponde a quantidade de regressores do vetor, inclusive a observação da componente denotada por c_{k-s}^j . Portanto, os *clouds* são gerados no espaço de dados com r dimensões, sendo que cada NF-TS terá um espaço de dados distinto. Ressalta-se que o valor de s pode ser determinado por uma análise de autocorrelação entre os dados usados nas condições iniciais do método de extração.

Atualização da Estrutura NF-TS Considerando as n NF-TS individualmente, para cada novo $\mathbf{d}_k^j$ que é formulado, deve-se atualizar os seguintes parâmetros como segue:

$$\begin{aligned}\mu_k^j &= \frac{k-1}{k} \mu_{k-1}^j + \frac{1}{k} \mathbf{d}_k^j, \quad \mu_k^j \in \mathbb{R}^{1 \times r}, \\ \beta_k^j &= \frac{k-1}{k} \beta_{k-1}^j + \frac{1}{k} \|\mathbf{d}_k^j\|_2, \quad j = 1, \dots, n,\end{aligned}\quad (11)$$

sendo que μ_k^j representa a média global de $\mathbf{d}_k^j$ e β_k^j é o produto escalar médio. Em seguida, é necessário calcular as densidades dos *data streams* $\mathbf{d}_k^j$ e de todos os centros dos *clouds*, considerando individualmente as NF-TS. A densidade é uma medida que diz respeito à distribuição espacial de uma amostra de dados específica em relação a todas as outras amostras de dados (Angelov et al., 2018). A densidade D de uma amostra genérica $\mathbf{z}$ considerando uma distância euclidiana é:

$$D(\mathbf{z}) = \frac{1}{1 + \frac{\|\mathbf{z} - \mu_k^j\|^2}{\beta_k^j - \|\mu_k^j\|^2}}. \quad (12)$$

Por meio da medida de densidade, é possível verificar se $\mathbf{d}_k^j$ pode ser considerado um centro de um novo *cloud*. A condição 1 a seguir analisa se um novo *cloud* ou regra *fuzzy* deve ser criada:

$$\text{Cond. 1: SE } D(\mathbf{d}_k^j) > \max_{i=1}^{R^j} D(\mathbf{f}_i^j) \text{ OU } D(\mathbf{d}_k^j) < \min_{i=1}^{R^j} D(\mathbf{f}_i^j) \quad (13)$$

onde $D(\mathbf{d}_k^j)$ e $D(\mathbf{f}_i^j)$ correspondem as densidades (Eq. 12) do *data stream* e do centro do *cloud* i ($\mathbf{f}_i^j$), respectivamente. A quantidade de *clouds* ou regras *fuzzy* da rede NF-TS j é denotada por R^j . Caso a Cond. 1 seja verdadeira, um novo *cloud* ou regra *fuzzy* é criado. Entretanto, precisa-se verificar se há sobreposição excessiva entre os *clouds* no espaço quando se adiciona este novo. Para evitar essa redundância e sobreposição, a seguinte condição é avaliada:

$$\text{Cond. 2: SE } \Delta_i^j \geq 1/(1 + \phi^2) \quad (14)$$

onde ϕ é um parâmetro universal para todas as NF-TS definido pelo especialista e Δ_i^j é a densidade calculada por regra *fuzzy* e por NF-TS usando a seguinte equação:

$$\Delta_i^j = \frac{1}{1 + \frac{(S_i^j)^2 \|\mathbf{d}_k^j - \mathbf{f}_i^j\|^2}{(S_i^j + 1)(S_i^j \chi_i^j + \|\mathbf{d}_k^j\|^2) - \|\mathbf{d}_k^j + S_i^j \mathbf{f}_i^j\|^2}}, \quad i = 1, \dots, R^j, \quad (15)$$

sendo que S_i^j é o suporte ou quantidade pontos associados ao *cloud* i , χ_i^j é o produto escalar médio dos pontos associados ao *cloud* i e $\mathbf{f}_i^j$ representa o centro do *cloud* i da NF-TS j . Sendo assim, caso a Cond. 2 (Eq. 14) seja verdadeira, não há necessidade de criar um novo *cloud* e apenas atualiza-se os parâmetros do *cloud* mais próximo representado pelo índice i^* conforme a seguir:

$$\mathbf{f}_{i^*}^j = \frac{\mathbf{f}_{i^*}^j + \mathbf{d}_k^j}{2}, \quad \chi_{i^*}^j = \frac{\chi_{i^*}^j + \|\mathbf{d}_k^j\|^2}{2}, \quad S_{i^*}^j = S_{i^*}^j + 1. \quad (16)$$

Caso contrário, cria-se um novo *cloud* ($R^j = R^j + 1$), cujo centro é $\mathbf{f}^j = \mathbf{d}_k^j$.

Se a Cond.1 na Eq. 13 não for satisfeita, significa que o ponto $\mathbf{d}_k^j$ deve ser associado ao *cloud* mais próximo. A atualização dos parâmetros do *cloud* nesse caso é feita por:

$$\begin{aligned}S_{i^*}^j &= S_{i^*}^j + 1, \quad \mathbf{f}_{i^*}^j = \frac{S_{i^*}^j - 1}{S_{i^*}^j} \mathbf{f}_{i^*}^j + \frac{1}{S_{i^*}^j} \mathbf{d}_k^j, \\ \chi_{i^*}^j &= \frac{S_{i^*}^j - 1}{S_{i^*}^j} \chi_{i^*}^j + \frac{1}{S_{i^*}^j} \|\mathbf{d}_k^j\|^2.\end{aligned}\quad (17)$$

A contribuição $\lambda_k^{j,i}$ de cada regra *fuzzy* i na saída global da NF-TS j , também conhecida como grau de ativação normalizado, apresenta valor entre 0 e 1 e pode ser calculada como:

$$\lambda_k^{j,i} = \frac{\Delta_i^j}{R^j} \quad (18)$$

onde o denominador é a soma das densidades de todos as regras *fuzzy* da NF-TS j . Algumas regras *fuzzy* podem perder utilidade ao longo do tempo, contribuindo muito pouco para a saída. Para simplificar o modelo, é interessante remover esses *clouds*. Logo, uma medida de utilidade desses *clouds* pode ser calculada para cada instante k como segue:

$$\eta_i^j = \frac{1}{k - T_i^j} \Lambda_i^j \text{ para } k \neq T_i^j, \quad \eta_i^j = 1 \text{ para } k = T_i^j, \quad (19)$$

onde η_i^j é a utilidade do *cloud* i da NF-TS j , T_i^j é o instante k que o *cloud* i foi gerado e Λ_i^j traz a soma de todos os graus de ativação $\lambda_k^{j,i}$ dessa regra *fuzzy* ao longo de todos os instantes k desde sua criação. A condição para remoção de regras é:

$$\text{Cond. 3: SE } \eta_i^j < \eta_0, \quad (20)$$

sendo que η_0 é um parâmetro definido pelo especialista. Os *clouds* avaliados que satisfazem a Cond. 3 acima são removidos, simplificando a estrutura de cada NF-TS.

Estimação dos Parâmetros dos Consequentes Nesta fase, os parâmetros dos consequentes das NF-TS são atualizados. A estrutura de uma regra *fuzzy* é representada para cada NF-TS $j = 1, \dots, n$ por:

$$\begin{aligned}\text{SE } (c_{k-1}^j \sim f_1^{j,1}) \text{ E } (c_{k-2}^j \sim f_1^{j,2}) \dots \text{ E } (c_{k-s}^j \sim f_1^{j,r}) \\ \text{ENTÃO } \hat{c}_k^{j,i} = \sum_{q=1}^{r-1} a_q^{j,i} c_{k-q}^j + a_r^{j,i} c_{k-s}^j + a_{r+1}^{j,i}\end{aligned}\quad (21)$$

onde $f_1^{j,p1}$ ($q = 1, \dots, r$) corresponde ao elemento na posição q do vetor $\mathbf{f}_1^j$, ou seja, o centro do *cloud* i da NF-TS j ; e os elementos $a_q^{j,i}$ são parâmetros dos consequentes que devem ser estimados. Por exemplo, uma regra *fuzzy* com $r = 3$ e $s = 10$ é representada por:

$$\begin{aligned}\text{SE } (c_{k-1}^j \sim f_1^{j,1}) \text{ E } (c_{k-2}^j \sim f_1^{j,2}) \text{ E } (c_{k-10}^j \sim f_1^{j,3}) \\ \text{ENTÃO } \hat{c}_k^{j,i} = a_1^{j,i} c_{k-1}^j + a_2^{j,i} c_{k-2}^j + a_3^{j,i} c_{k-10}^j + a_4^{j,i}\end{aligned}\quad (22)$$

Para estimar os parâmetros, usa-se o método dos Mínimos Quadrados Recursivos Ponderados *Fuzzy* Local (Angelov et al., 2018) para cada regra *fuzzy*, considerando o grau de ativação normalizado $\lambda_k^{j,i}$ como fator de ponderação conforme a seguir:

$$G_k^{j,i} = \frac{P_{k-1}^{j,i} v^T}{v \lambda_k^{j,i} P_{k-1}^{j,i} v^T + 1}, \quad G_k^{j,i} \in \mathbb{R}^{(r+1) \times 1}$$

$$a_k^{j,i} = a_{k-1}^{j,i} + \lambda_k^{j,i} G_k^{j,i} (c_k^j - v a_{k-1}^{j,i}), \quad a_k^{j,i} \in \mathbb{R}^{(r+1) \times 1}$$

$$P_k^{j,i} = P_{k-1}^{j,i} - \lambda_k^{j,i} G_k^{j,i} v P_{k-1}^{j,i} v^T, \quad P_k^{j,i} \in \mathbb{R}^{(r+1) \times (r+1)}$$

para $v = [d_k^j \ 1] \in \mathbb{R}^{1 \times (r+1)}$.

(23)

Saídas das NF-TS Distribuídas A partir dessas saídas dos submodelos locais representados pelos *clouds* ou regras *fuzzy*, é possível obter a previsão da componente $\hat{c}_k^j$ como segue:

$$\hat{c}_k^j = \sum_{i=1}^{R^j} \lambda_k^{j,i} \hat{c}_k^{j,i} \quad (24)$$

Somando os resultados das j NF-TS (Eq. 24), obtém-se a observação para a série temporal original:

$$\hat{y}_k = \sum_{j=1}^n \hat{c}_k^j. \quad (25)$$

Após o treinamento, as NF-TS realizam a previsão de acordo com o horizonte de previsão h desejado. Quando se realiza a previsão, não há dados disponíveis da série temporal, portanto, não é possível extrair novas amostras das componentes não-observáveis. Como dados das componentes são fundamentais na formulação das regras *fuzzy* para a previsão livre, os próprios resultados das previsões das componentes são realimentados para as respectivas NF-TS, a fim de que a inferência *fuzzy* seja executada. Na previsão, as redes NF-TS não evoluem, ou seja, mantem as estruturas fixas após o treinamento.

3. RESULTADOS EXPERIMENTAIS

Nesta seção, a metodologia proposta é avaliada usando a previsão livre ou múltiplos passos a frente, comparando seus resultados com a pesquisa de Nguyen and Novak (2019), que também usam um procedimento semelhante, realizando a previsão de componentes separadamente a *priori* e, em seguida, agrupando os resultados individuais para prever a série temporal original. À vista disso, as séries temporais reais consideradas nesses experimentos são (Nguyen and Novak, 2019):

- (1) TS1: Quantidade mensal de pessoas desempregadas na Austrália (Set. 1978 – Ago. 1995);
- (2) TS2: Despesa trimestral bruta de capital fixo - construção não-residencial privada, Austrália: em milhões de dólares. 3º Trimestre de 1960 a 2º Trimestre de 1991;
- (3) TS3: Vazão média mensal em um rio, Jan. 1953 – Dez. 1981;
- (4) TS4: Produção trimestral de tijolos de barro: milhões de unidades. Q4 1956–Q3 1994).

A Tabela 1 a seguir exibe os parâmetros necessários para realização dos experimentos de cada série temporal usando a abordagem proposta por esse trabalho. Destaca-se que: a janela inicial foi considerada $w = s$; foram extraídas $n = 3$ componentes não-observáveis de cada série; h é o horizonte

Figura 2. Previsão das Séries Temporais Aplicando as NF-TS Distribuídas.

de previsão; r corresponde a quantidade de regressores no *data stream*; ϕ é o parâmetro da Eq. 14; e η_0 é o parâmetro da Eq. 20.

Tabela 1. Parâmetros dos Experimentos para Cada Série Temporal.

Série Temporal	w	n	s	h	r	ϕ	η_0
TS1	10	3	10	20	4	0.1	0.25
TS2	16	3	16	12	7	0.1	0.33
TS3	23	3	23	18	5	0.1	0.36
TS4	6	3	6	16	3	0.1	0.32

A Figura 2 mostra os resultados da previsão em vermelho para as quatro séries temporais citadas, aplicando a metodologia proposta nesse trabalho baseada em NF-TS evolutivas e distribuídas. Por sua vez, a Figura 3 traz com mais detalhes a série temporal TS4, exibindo o resultado de previsão das 3 NF-TS distribuídas. Cada NF-TS tenta prever uma componente não-observável em separado evoluindo sua estrutura, ou seja, adicionando ou removendo regras *fuzzy* conforme pode-se visualizar.

Por fim, a Tabela 2 exibe os resultados comparativos da metodologia proposta com 3 abordagens apresentadas no trabalho de Nguyen and Novak (2019). Percebe-se que a estratégia apresentada obteve os melhores resultados segundo os critérios RMSE e MAPE durante o horizonte de previsão h .

4. CONCLUSÃO

Neste trabalho, foi apresentada uma metodologia evolutiva baseada em redes *Neuro-Fuzzy Takagi-Sugeno* (NF-TS) distribuídas. A originalidade desse trabalho consiste na utilização de várias NF-TS evolutivas para prever dados de cada componente não-observável extraída da série temporal original. Portanto, cada componente possui uma NF-TS evolutiva associada. Por fim, agrupa-se a previsão de cada componente para obter a previsão da série temporal.

Este trabalho destaca a ideia em dividir um problema complexo em vários subproblemas mais simples. Os resul-

Tabela 2. Resultados Comparativos entre a Metodologia Proposta e Métodos Apresentados em Nguyen and Novak (2019) de acordo com os Critérios RMSE e MAPE.

Série Temporal	NF-TS Distribuídas		Abordagem Fuzzy		STL (<i>Seasonal Trend decomposition based on Loess</i>)		ARIMA	
	RMSE	MAPE	RMSE	MAPE	RMSE	MAPE	RMSE	MAPE
TS1	30092.26	0.02	47807	0.05	58662	0.07	89893	0.09
TS2	142.64	0.04	274.01	0.08	274.39	0.08	337.07	0.11
TS3	9.97	0.12	10.29	0.23	16.42	0.41	15.81	0.38
TS4	22.87	0.01	23.35	0.04	29.46	0.05	81.85	0.18

(a) Previsão da Componente 1 (NF-TS 1).

(b) Evolução da quantidade de Clouds da NF-TS 1.

(c) Previsão da Componente 2 (NF-TS 2).

(d) Evolução da quantidade de Clouds da NF-TS 2.

(e) Previsão da Componente 3 (NF-TS 3).

(f) Evolução da quantidade de Clouds da NF-TS 3.

Figura 3. Previsão das Componentes Não-Observáveis e Evolução da Quantidade de Regras Fuzzy ou Clouds para a Série Temporal TS4.

tados das NF-TS distribuídas mostraram índices de erro inferiores em comparação com métodos presentes na pesquisa de Nguyen and Novak (2019), destacando o potencial da metodologia proposta. Para trabalhos futuros, definir um critério para ajustar os parâmetros da metodologia proposta é interessante. A aplicação em mais estudos de casos é importante para delimitar e identificar para que tipos de séries temporais essa abordagem apresenta bons resultados. Finalmente, o desenvolvimento de uma versão para séries temporais multivariáveis também é um objetivo futuro dessa pesquisa.

AGRADECIMENTOS

Os autores agradecem ao IFMA/Campus Imperatriz, a PPGEE/UFMA e ao IFMA/Campus Monte Castelo por todo apoio ao desenvolvimento dessa pesquisa.

REFERÊNCIAS

- Abdollahzade, M., Miranian, A., Hassani, H., and Iranmanesh, H. (2015). A new hybrid enhanced local linear neuro-fuzzy model based on the optimized singular spectrum analysis and its application for nonlinear and chaotic time series forecasting. *Information Sciences*, 361, 107–125. doi:10.1016/j.ins.2014.09.002.
- Angelov, P., Gu, X., and Príncipe, J. (2018). Autonomous learning multi-model systems from data streams. *IEEE Transactions on Fuzzy Systems*, 26, 2213–2224. doi:10.1109/TFUZZ.2017.2769039.
- Bergmeir, C., Hyndman, R., and Bentez, J. (2016). Bagging exponential smoothing methods using stl decomposition and box-cox transformation. *International Journal of Forecasting*, 32, 303–312. doi:10.1016/j.ijforecast.2015.07.002.
- Joo, T. and Kim, S. (2015). Time series forecasting based on wavelet filtering. *Expert Systems With Applications*, 42, 3868–3874. doi:10.1016/j.eswa.2015.01.026.
- L. Wu, S.L. and Yang, Y. (2016). Grey double exponential smoothing model and its application on pig price forecasting in china. *Applied Soft Computing*, 391, 117–123. doi:10.1016/j.asoc.2015.09.054.
- Nguyen, L. and Novak, V. (2019). Forecasting seasonal time series based on fuzzy techniques. *Fuzzy Sets and Systems*, 361, 114–129. doi:10.1016/j.fss.2018.09.010.
- Rodrigues Júnior, S. and Serra, G. (2017). A novel intelligent approach for state space evolving forecasting of seasonal time series. *Engineering Applications of Artificial Intelligence*, 64, 272–285. doi:10.1016/j.engappai.2017.06.016.
- Sagheer, A. and Kotb, M. (2019). Time series forecasting of petroleum production using deep LSTM recurrent networks. *Neurocomputing*, 323, 203–213.
- Tealab, A. (2018). Time series forecasting using artificial neural networks methodologies: A systematic review. *Future Computing and Informatics Journal*, 3, 334–340.
- Wang, J., Wang, J., Fang, W., and Niu, H. (2016). Financial time series prediction using elman recurrent random neural networks. *Computational Intelligence and Neuroscience*, 2016. doi:10.1155/2016/4742515.
- Wang, L., Zou, H., Su, J., Li, L., and Chaudhry, S. (2013). An arima-ann hybrid model for time series forecasting. *Systems Research and Behavioral Science*, 30, 244–259. doi:10.1002/sres.2179.
- Yolcu, U., Egrioglu, E., and Aladag, C. (2013). A new linear & nonlinear artificial neural network model for time series forecasting. *Decision Support Systems*, 54, 1340–1347. doi:10.1016/j.dss.2012.12.006.