

## REDES BASEADAS EM PROJEÇÕES ALEATÓRIAS PARA PREDIÇÃO NÃO-LINEAR DE SÉRIES TEMPORAIS CAÓTICAS

JOSÉ MARIA P. DE MENEZES JÚNIOR\* GUILHERME A. BARRETO†

\*Universidade Federal do Piauí  
Teresina, Piauí, Brasil

†Universidade Federal do Ceará  
Fortaleza, Ceará, Brasil

Emails: [josemenezesjr@ufpi.edu.br](mailto:josemenezesjr@ufpi.edu.br), [guialenbar@gmail.com](mailto:guialenbar@gmail.com)

**Abstract**— In this paper we evaluate the performance of the ELM network based on NARX model in the task of prediction of Hénon, Mackey-Glass (MG) and chaotic laser series. An analysis of the results showed that the NARX-ELM network achieved the best results for the Hénon and MG series. The same was not true for the chaotic laser serie, whereas the ELM networks failed on making more accurate predictions for a horizon greater than the NARX network trained with the backpropagation algorithm.

**Keywords**— Recurrent neural networks, NARX network, Extreme Machine Learning, recursive prediction.

**Resumo**— Neste trabalho é avaliado o desempenho da rede ELM baseada no modelo NARX na tarefa de predição da série de Hénon, Mackey-Glass (MG) e laser caótico. Uma análise dos resultados obtidos mostrou que a rede NARX-ELM conseguiu o melhor resultados para as séries de Hénon e MG. O mesmo não aconteceu para a série do laser caótico, visto que as redes ELM não conseguiram fazer predições mais corretas por um horizonte maior do que a rede NARX treinada com *backpropagation*.

**Palavras-chave**— Redes neurais recorrentes, rede NARX, Máquina de Aprendizado Extremo, predição recursiva.

### 1 Introdução

Redes neurais artificiais (RNAs) têm sido utilizadas com sucesso em tarefas de predição e modelagem de séries temporais; ver, por exemplo, Palit and Popovic (2005), uma pesquisa de técnicas e aplicações em diversas áreas. Em particular, quando séries temporais são geradas por um sistema dinâmico não-linear, modelos de RNA geralmente tem melhor desempenho que as técnicas lineares tradicionais, tais como os modelos de Box-Jenkins. Neste caso, as habilidades de generalização e aproximação universal de funções de RNA justificam seu melhor desempenho preditivo.

Na tarefa de predição de séries temporais um-passo-adiante (UPA), modelos de RNA são usados para estimar somente o próximo valor de uma série temporal, sem realimentar o valor de saída predito para a entrada da rede na fase de validação. Para predição de horizonte mais amplo, comumente é utilizado um procedimento conhecido como predição  $H$ -passos-adiante (HPA), em que a saída do modelo deve ser realimentada para a entrada de forma recursiva até atingir o instante futuro desejado. A predição HPA é mais difícil de lidar do que a predição UPA e estas são tarefas complexas em que os modelos de RNA desempenham um importante papel, em particular, aqueles relacionados às arquiteturas neurais recorrentes (Principe et al., 2000).

RNAs recorrentes têm laços de realimentação local e/ou global em sua estrutura. Redes *feedforward* (i.e. sem laços de realimentação) podem ser adaptadas para processar séries temporais através de uma entrada com linha de atrasos com derivações (*tapped delay line*) e treinada pelo algoritmo *backpropagation*, elas podem também ser convertidas em arquiteturas recorrentes simples por realimentações de ativações

dos neurônios das camadas escondidas ou de saída, dando origem às redes de Elman e de Jordan, respectivamente.

Um novo paradigma de projeto de RNAs vem sendo proposto sob a alcunha de Projeções Aleatórias (Miche et al., 2010), onde os pesos dos neurônios da camada oculta são iniciados de forma aleatória e mantidos fixos, enquanto apenas os pesos da camada de saída são ajustáveis. Dentro do arcabouço geral das RNAs baseadas em projeções aleatórias destacam-se as arquiteturas no contexto da Computação de Reservatório (Miche et al., 2010) e a arquitetura ELM (*Extreme Learning Machine*) (Huang et al., 2011), sendo a primeira uma arquitetura recorrente e a segunda uma arquitetura *feedforward*.

Do exposto, este artigo tem o propósito de avaliar principalmente o desempenho da rede ELM com modificações comparando-os com outras arquiteturas neurais, na tarefa predição recursiva. O objetivo deste estudo é ressaltar as diferenças no projeto de tais arquiteturas neurais, a fim de oferecer subsídios ao usuário no momento da escolha da arquitetura mais adequada à tarefa de interesse. Para isto, são utilizadas as séries de Hénon, de MG e do laser caótico.

O restante do artigo está organizado da seguinte forma. A Seção 2 traz uma apresentação da rede NARX aplicada à predição de séries temporais univariadas. As redes baseadas em projeções aleatórias e suas modificações são descritas na Seção 3. Por fim, na Seção 4 são discutidos os resultados da predição recursiva obtidos pelos vários paradigmas de redes neurais recorrentes, utilizando as séries de Hénon, MG e laser caótico como *benchmarking*.

## 2 A Rede NARX

Uma importante classe de sistemas não-lineares de tempo discreto é matematicamente representada pelo modelo NARX (*Nonlinear AutoRegressive model with eXogenous inputs*)

$$y(n) = f[y(n-1), \dots, y(n-d_y); u(n), u(n-1), \dots, u(n-d_u+1)], \quad (1)$$

em que  $u(n) \in \mathbb{R}$  e  $y(n) \in \mathbb{R}$  representam, respectivamente, a entrada e a saída do modelo no instante  $n$ , enquanto  $d_u > 0$  e  $d_y > 0$ ,  $d_u \leq d_y$ , são as ordens dos regressores de entrada e memória de saída.

Assume-se que  $f(\cdot)$  é um mapeamento não-linear desconhecido. Quando esta função é aproximada por uma rede MLP, a topologia resultante é chamada de *rede recorrente NARX* (Narendra and Parthasarathy, 1990) (Figura 1). Quando a ordem da memória da saída da rede NARX é feita  $d_y = 0$ , reduz-se a rede NARX a uma rede MLP padrão.

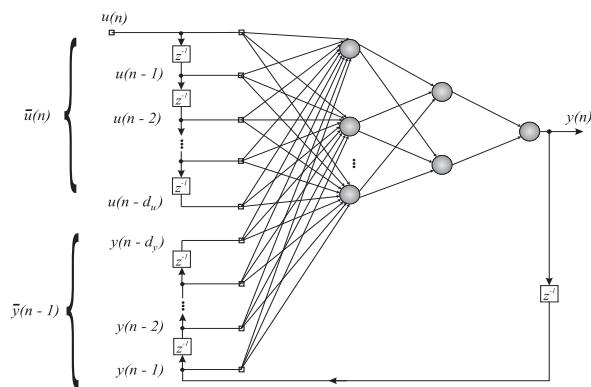


Figura 1: Rede NARX com  $d_u$  entradas e  $d_y$  atrasos da saída.

Dada uma série temporal  $\{x(n)\}_{n=1}^N$ , define-se o regressor do sinal de entrada da rede NARX, denotado por  $\mathbf{U}(n)$ . Assim, tem-se que

$$\mathbf{U}(n) = [u(n), u(n-1), \dots, u(n-d_u+1)], \quad (2)$$

$$= [x(n), x(n-\tau), \dots, x(n-(d_E-1)\tau)],$$

em que se percebe que o regressor  $\mathbf{U}(n) \in \mathbb{R}^{d_u}$  é composto de  $d_E$  (dimensão de imersão) valores observados da série temporal, amostrados a cada  $\tau$  (atraso de imersão) unidades de tempo, seguindo o teorema de Takens (Takens, 1981).

Para levar em conta a informação dinâmica fornecida pelo laço de realimentação, a rede NARX pode ser treinada e usada nos modos paralelo e série-paralelo. No modo de treinamento paralelo (ou recorrente), o regressor do sinal de saída, representado por  $\mathbf{Y}(n)$ , é definido como

$$\mathbf{Y}(n-1) = [y(n-1), y(n-2), \dots, y(n-d_y)], \quad (3)$$

em que se percebe que o regressor de saída  $\mathbf{Y}(n) \in \mathbb{R}^{d_y}$  comprehende  $d_y$  saídas prévias da rede neural.

Vale lembrar que, para uma rede treinada adequadamente, a saída da rede no instante  $n$  é uma estimativa do valor futuro da série, ou seja,  $y(n) = \hat{x}(n + 1)$ . Assim, a Equação (3) pode ser escrita também da seguinte forma:

$$\mathbf{Y}(n-1) = [\widehat{x}(n), \widehat{x}(n-1), \dots, \widehat{x}(n-d_y+1)]. \quad (4)$$

Já, no modo série-paralelo (ou não-recorrente), o regressor de saída é definido como segue

$$\mathbf{Y}(n-1) = [x(n), x(n-1), \dots, x(n-d_y+1)], \quad (5)$$

sendo, portanto, construído com amostras reais da série temporal de interesse. Qualquer que seja o modo de uso da rede NARX, ambas as arquiteturas implementam o seguinte mapeamento

$$y(n) = \hat{x}(n+1) = \hat{f}[\mathbf{Y}(n-1), \mathbf{U}(n)], \quad (6)$$

em que a função não-linear  $\hat{f}(\cdot)$  pode ser realizada prontamente pela rede MLP padrão e treinada pelo algoritmo *backpropagation* convencional.

De forma resumida, a rede NARX é definida de modo que seu regressor de entrada contém amostras da variável medida  $x(n)$  separados por  $\tau > 0$  intervalos de tempo uns com os outros, enquanto que o regressor de saída contém valores reais ou estimados da mesma variável, mas amostrados em intervalos de tempo consecutivos.

Um importante destaque a respeito da aproximação proposta é relacionada com a estabilidade da aprendizagem da rede NARX. Narendra and Parthasarathy (1990) mostram que o modo de treinamento paralelo é mais instável do que o modo série-paralelo. Isto se deve basicamente à propagação do erro de previsão gerada pelas estimativas realimentadas para o regressor de saída. Desta forma, neste artigo utiliza-se a rede NARX somente no modo de treinamento série-paralelo.

### 3 A Rede ELM

Os algoritmos de treinamento das RNAs tradicionais (recorrentes ou *feedforward*) são caracterizados pelo excesso de parâmetros ajustáveis, pela baixa velocidade de convergência e por problemas causados pela sensibilidade aos pesos iniciais. Já as RNAs baseadas em projeções aleatórias possuem poucos parâmetros a serem ajustados e uma estrutura fácil de ser construída (Huang et al., 2011). Mesmo com um projeto simplificado, as RNAs baseadas em projeções aleatórias normalmente possuem desempenho equivalente ao de outras RNAs convencionais, tais como as redes MLP, RBF e ELMAN, despertando, por isto, o interesse da comunidade científica.

Dentro do arcabouço geral das redes baseadas em projeções aleatórias, a rede ELM, utiliza uma simples camada de uma rede neural *feedforward*, sem realimentação. Os pesos da camada oculta são iniciados aleatoriamente e não precisam ser atualizados, ficando

a cargo apenas do cálculo dos pesos da camada de saída.

A Figura 2 ilustra a rede ELM com a entrada  $\mathbf{x}(n)$  definida de acordo com o teorema de Takens, com a matriz  $\mathbf{W}$  representando os pesos da camada não-linear que permanecem fixos durante o treinamento e o vetor  $\mathbf{m}$  representando os pesos do único neurônio de saída.

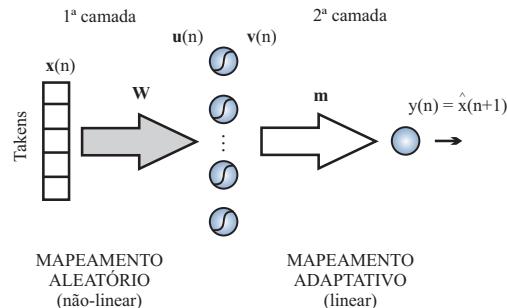


Figura 2: Ilustração da arquitetura da rede ELM aplicada em predição de séries temporais.

Na rede ELM, o aprendizado é executado em três passos distintos, que são comentados a seguir.

### (i) Inicialização Aleatória dos Pesos da Camada Oculta

O funcionamento da rede ELM envolve o cálculo das ativações e saídas de todos os neurônios da camada oculta e do neurônio da camada de saída, uma vez que os pesos  $w_{ij}$ ,  $i = 1, \dots, q$  e  $j = 0, \dots, d_E$ , tenham sido inicializados com valores aleatórios.

Para isto, precisa-se definir uma matriz de pesos  $\mathbf{W}$ , com  $q$  linhas e  $d_E + 1$  colunas, em que a  $i$ -ésima linha da matriz  $\mathbf{W}$  é composta pelo vetor de pesos do  $i$ -ésimo neurônio oculto.

### (ii) Acúmulo das Saídas dos Neurônios Ocultos

Este passo corresponde a etapa de treinamento da rede, no qual se obtém as ativações dos neurônios ocultos e suas respectivas saídas. Este passo destina-se à obtenção de uma matriz formada a partir das saídas dos neurônios da camada oculta, calculada conforme a apresentação de cada vetor de entrada à rede neural.

O vetor de entrada é definido como:

$$\mathbf{x}(n) = [x(n) \ x(n-\tau) \ \dots \ x(n-(d_E-1)\tau)]^T, \quad (7)$$

em que  $\mathbf{x}(n)$  segue o teorema de Takens e é um vetor que contém  $d_E$  elementos da série, espaçados um do outro de  $\tau$  unidades de tempo.

Assim, após a apresentação de um vetor de entrada  $\mathbf{x}$ , na iteração  $n$ , as saídas da camada oculta são calculadas como:

$$\begin{aligned} v_i(n) &= \phi[u_i(n)], \\ &= \phi \left[ \sum_{j=1}^{d_E} w_{ij} x_j(n) - \theta_i(n) \right], \quad i = 1, \dots, q, \end{aligned} \quad (8)$$

em que  $\phi(\cdot)$  é uma não-linearidade do tipo sigmoidal,  $q$  indica o número de neurônios da camada oculta e  $\theta_i(n)$  é o limiar do  $i$ -ésimo neurônio da camada oculta.

Vectorialmente, a Equação (8) pode ser escrita como

$$\mathbf{v}(n) = \phi(\mathbf{u}(n)) = \phi(\mathbf{W}\mathbf{x}(n)). \quad (9)$$

Para cada vetor de entrada  $\mathbf{x}(n)$ ,  $n = 1, \dots, N$ , tem-se um vetor  $\mathbf{v}(n)$  correspondente, que deve ser organizado (disposto) como uma coluna de uma matriz  $\mathbf{V}$ . Esta matriz terá  $q$  linhas por  $N$  colunas:

$$\mathbf{V} = [\mathbf{v}(1) \ | \ \mathbf{v}(2) | \ \dots \ | \ \mathbf{v}(N)]_{q \times N}. \quad (10)$$

A matriz  $\mathbf{V}$  será usada no Passo 3 para calcular os valores dos pesos dos neurônios de saída da rede ELM.

### (iii) Cálculo dos Pesos dos Neurônios de Saída

Sabe-se que para cada vetor de entrada  $\mathbf{x}(n)$ ,  $n = 1, \dots, N$ , tem-se um escalar de saída desejado  $d(n)$  correspondente. Se estes  $N$  escalares forem organizado ao longo das linhas de um vetor  $\mathbf{d}$ , então este vetor possui dimensão  $N$ :

$$\mathbf{d} = [d(1) \ d(2) \ \dots \ d(N)]^T. \quad (11)$$

Pode-se entender o cálculo dos pesos da camada de saída como o cálculo dos parâmetros de um mapeamento linear entre a camada oculta e a camada de saída. O papel do vetor de “entrada” para a camada de saída na iteração  $n$  é desempenhado pelo vetor  $\mathbf{v}(n)$  enquanto a “saída” é representada pelo escalar  $d(n)$ . Assim, busca-se determinar o vetor  $\mathbf{m}$  que melhor representa a transformação:

$$d(n) = \mathbf{m}^T \mathbf{v}(n). \quad (12)$$

Para isto, pode-se usar o método dos mínimos quadrados, também conhecido como método da pseudoinversa. Assim, usando a matriz  $\mathbf{V}$  e o vetor  $\mathbf{d}$ , o vetor de pesos  $\mathbf{m}$  é calculado por meio da seguinte expressão:

$$\mathbf{m} = [\mathbf{V}\mathbf{V}^T]^{-1} \mathbf{V}\mathbf{d}, \quad (13)$$

em que  $\mathbf{m}$  é o vetor de pesos dos neurônios da camada de saída e possui dimensão  $q$ .

Singh and Balasundaram (2007) utilizaram a rede ELM para predição de séries temporais caóticas. Alguns parâmetros foram otimizados em termos do erro médio quadrático, tais como o número de neurônios escondidos e a ordem da memória de entrada. Estes autores utilizaram as séries de MG, do laser caótico e de batimentos cardíacos. Sovilj et al. (2010) também utilizaram a rede ELM em tarefas de predições de longo prazo, estudando soluções para processamento da entrada e propondo métodos automáticos para determinação da dimensão da projeção aleatória.

### 3.1 Extensões da Rede ELM para Predição de Séries Temporais

As extensões da rede ELM, propostas a seguir, envolvem as ideias já estabelecidas para redes *feedforward*, como a rede NARX, discutida na Seção 2 e a rede de Elman, introduzida por Elman (1990).

As arquiteturas sugeridas utilizam os mesmos três passos enumerados para o treinamento da rede ELM original. A única diferença é a forma com que a entrada da rede ELM é construída. O acúmulo das saídas dos neurônios ocultos e o cálculo dos pesos dos neurônios de saída seguem as mesmas considerações matemáticas definidas para a rede ELM original, descritas nas Equações (10) e (13), respectivamente.

### 3.1.1 Rede NARX-ELM

A camada de entrada da rede NARX-ELM passa a ser dividida em duas partes (Figura 3). A primeira parte, corresponde ao vetor de entrada, definido conforme a Equação (7). A segunda parte é formada pelo regressor de saída da rede NARX no modo-SP:

$$\begin{aligned} \mathbf{y}_{sp}(n-1) &= [x(n) \ x(n-1) \ \cdots \ x(n-d_y+1)]^T \\ &= [x_1^c(n) \ x_2^c(n) \ \cdots \ x_{d_y}^c(n)]^T. \end{aligned} \quad (14)$$

Note que o regressor de saída para o modo-SP mostrado na Equação (14) contém  $d_y$  valores passados da série temporal observada. Desta forma, a rede ELM com esta estrutura não possui recorrência na fase de treinamento, pois as duas partes que formam a unidade de entrada da rede são oriundas da própria série observada.

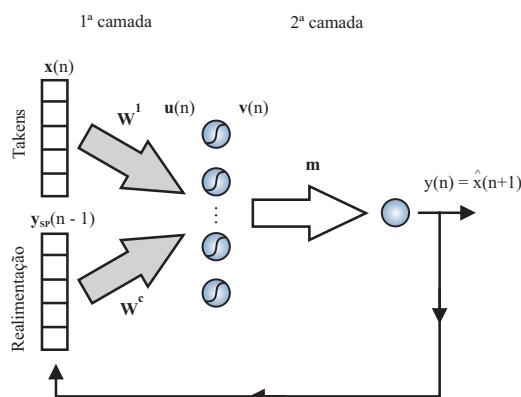


Figura 3: Rede NARX-ELM aplicada em predição de séries temporais.

### 3.1.2 Rede ELMAN-ELM

A rede ELMAN-ELM é baseada na arquitetura recorrente de Elman, sendo obtida a partir da rede MLP através da redefinição da camada de entrada da rede (Figura 4). A primeira parte corresponde ao vetor de entrada definido conforme a Equação (7). A segunda parte da entrada da rede de Elman contém as unidades de contexto, cujos valores são obtidos a partir da realimentação das saídas dos neurônios ocultos no instante  $n-1$ , ou seja

$$\begin{aligned} \mathbf{x}^c(n) &= [x_1^c(n) \ x_2^c(n) \ \cdots \ x_q^c(n)]^T \in \mathbb{R}^q \quad (15) \\ &= [v_1(n-1) \ v_2(n-1) \ \cdots \ v_q(n-1)]^T, \end{aligned}$$

em que  $\mathbf{x}^c(n) \in \mathbb{R}^q$  é chamada de vetor de contexto.

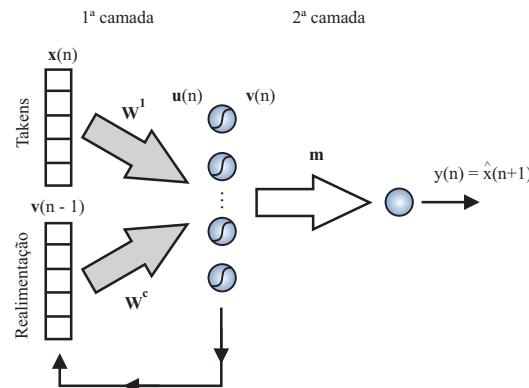


Figura 4: Rede ELMAN-ELM aplicada em predição de séries temporais.

### 3.1.3 Rede Híbrida ELMAN/NARX-ELM

Esta arquitetura recorrente é baseada nas propostas da rede NARX-ELM e da rede ELMAN-ELM ao mesmo tempo. A camada de entrada desta rede passa a ser dividida em três partes: a primeira parte corresponde ao vetor definido conforme a Equação 7, a segunda parte corresponde as unidades obtidas a partir da rede de Elman e a terceira parte é definida pelo regressor de saída da rede NARX.

Esta nova abordagem tem por objetivo aliar a capacidade de modelar dependências temporais de curto prazo da rede de Elman, com a capacidade de modelar dependências de longo prazo da rede NARX. Com isto, tenta-se verificar se esta combinação de informações pode trazer algum ganho na tarefa de predição múltiplos-passos-adiante.

## 4 Simulações e Resultados

As séries caóticas utilizadas para avaliação das RNAs são as séries de Hénon, MG e por fim do laser caótico. É observado o desempenho da rede ELM padrão, sem realimentações, como também o desempenho das variantes NARX-ELM, ELMAN-ELM e ELMAN/NARX-ELM. Esses resultados são comparados com o desempenho da rede NARX treinada com *backpropagation*.

Por razões de precisão estatística, cada rodada de treino/teste de cada modelo é repetido  $K = 20$  vezes. Quantitativamente, para cada  $k$ -th rodada de treino/teste, os modelos são avaliados em termos do NMSE.

É utilizada uma heurística, através de um algoritmo de busca exaustiva, para encontrar os melhores parâmetros de cada modelo. São otimizados os valores da dimensão de imersão, atraso de imersão, variância nos pesos na inicialização aleatória, número de neurônios em cada camada escondida e por fim a dimensão do contexto da rede NARX, caso se esteja usando o modelo NARX. A busca se inicia com todos os valores de parâmetros aleatórios e o processo se estende otimizando uma faixa de um determinado parâmetro. O

melhor parâmetro encontrado é aquele dado pelo menor valor da mediana do NMSE, dentro da faixa de busca. Este parâmetro encontrado é guardado e utilizado para a otimização do próximo parâmetro. Depois que todos os parâmetros já tenham sido otimizados, o processo se reinicia com os melhores parâmetros de um ciclo já encontrados. A heurística continua até que um requisito de parada seja encontrado.

#### 4.1 Resultados para as Séries de Hénon e de MG

Para a série de Hénon são gerados 200 valores, sendo 50 utilizados para testes e o restante para treino. Para a série de MG são utilizadas 500 amostras, sendo que, 200 são utilizados para treino e o restante para teste. Nas Figuras 5 e 6 encontram-se, por meio de *box-plots*, os resultados comparativos das quatro variações da rede ELM para as séries de Hénon e MG, respectivamente. Pode-se destacar o bom desempenho da rede NARX-ELM, que obteve o menor NMSE comparado com as outras redes avaliadas na tarefa de predição múltiplos-passos-adiante.

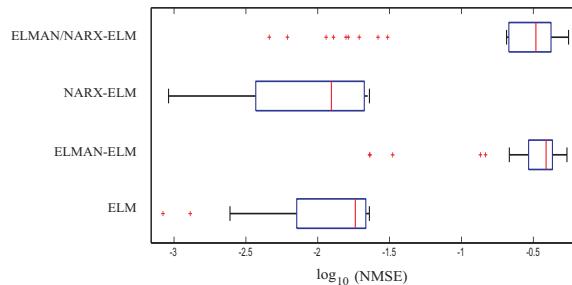


Figura 5: Valores do NMSE das variantes da rede ELM (teste recursivo,  $H = 50$ ) para série de Hénon.

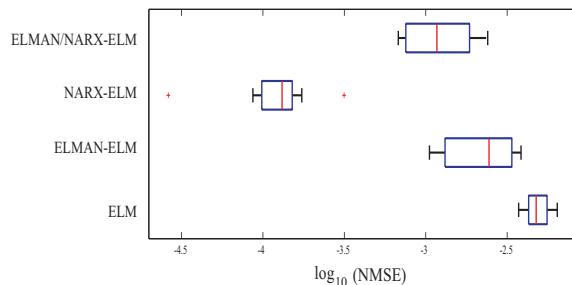


Figura 6: Valores do NMSE das variantes da rede ELM (teste recursivo,  $H = 30$ ) para a série de MG.

Verificou-se que a rede NARX-ELM, além de possuir o menor NMSE na predição múltiplos-passos-adiante, necessitou de um número menor de neurônios na camada oculta. É importante destacar que este parâmetro é o mais importante da rede ELM, pois é diretamente ligado ao tamanho da rede e desta forma influencia o tempo de treinamento.

Nas Figuras 7 e 8, observa-se os resultados da predição múltiplos-passos-adiante da série Hénon e da série de MG, respectivamente, utilizando a rede NARX-ELM. No resultado da série de Hénon, verifica-se

que os valores da predição conseguem acompanhar os valores reais da série em quase toda totalidade do horizonte utilizado. Já para o resultado da série de MG verifica-se que os valores da predição conseguem acompanhar os valores reais da série até em torno do horizonte  $H = 120$ .

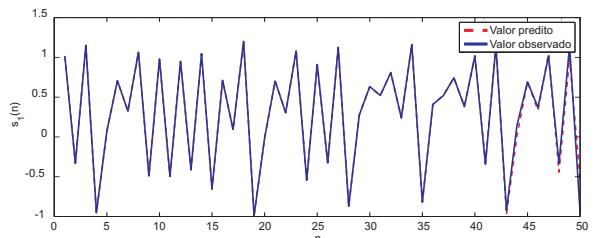


Figura 7: Predição da série Hénon com a rede NARX-ELM, teste recursivo,  $H = 50$ .

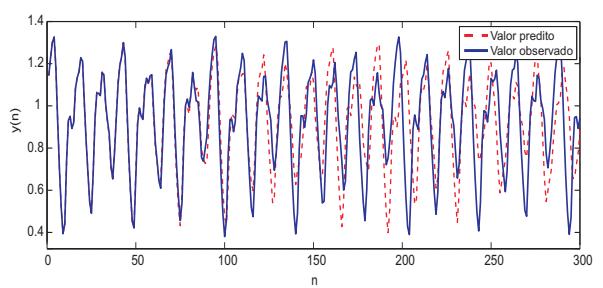


Figura 8: Predição para a série de MG com a rede NARX-ELM, teste recursivo,  $H = 300$ .

#### 4.2 Resultados para a Série do Laser Caótico

No próximo teste, avaliam-se as redes ELM para a série do laser caótico. Esta série temporal possui 1100 amostras, sendo que as 1000 primeiras são destinadas para o treino e as 100 últimas para o teste. A Figura 9 apresenta os resultados produzidos pelas diversas redes analisadas. A rede NARX-ELM e NARX/Elman-ELM alcançaram os melhores resultados, dentre as redes analisadas.

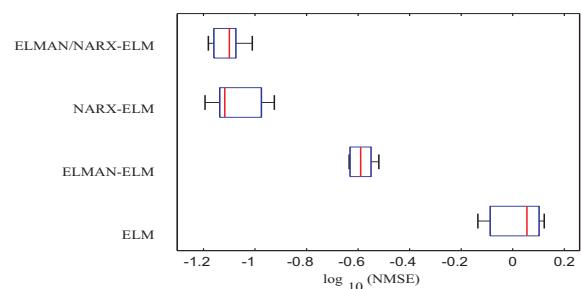


Figura 9: Valores do NMSE das redes ELM (teste recursivo,  $H = 50$ ) para a série do laser caótico.

Na Figura 10 observa-se a predição recursiva da série do laser caótico utilizando a rede NARX-ELM. Verifica-se que os valores da predição conseguem acompanhar os valores observados da série somente

até em torno do horizonte  $H = 50$ . Diferentemente, a rede NARX treinada com *backpropagation* é capaz de reproduzir (modelar) as dinâmicas do laser caótico para horizontes mais longos, como pode ser visto na Figura 11.

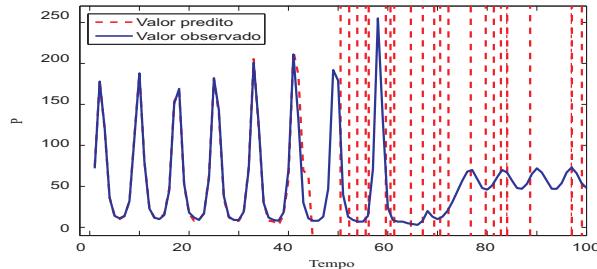


Figura 10: Predição recursiva da série do laser caótico com a rede NARX-ELM

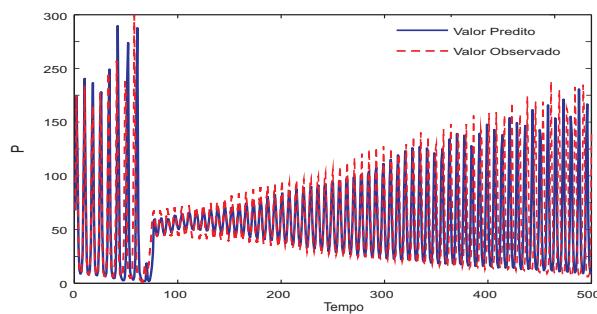


Figura 11: Predição recursiva da a série do laser caótico utilizando a rede NARX.

Por fim, a título de comparação, na Tabela 1 são mostrados os resultados dos desempenhos da rede NARX treinada com *backpropagation* e da rede NARX-ELM para três diferentes horizontes de predição. São mostrados os valores médios do erro e as respectivas variâncias (entre parênteses).

Tabela 1: Análise comparativa.

	NARX	NARX-ELM
Série de Hénon $H = 10$	$8,1 \times 10^{-4}$ ( $2,3 \times 10^{-7}$ )	$1,74 \times 10^{-14}$ ( $4,73 \times 10^{-28}$ )
Série de Mackey-Glass $H = 30$	$6,42 \times 10^{-4}$ ( $2,88 \times 10^{-7}$ )	$1,31 \times 10^{-4}$ ( $5,63 \times 10^{-9}$ )
Série do Laser Caótico $H = 50$	$9,76 \times 10^{-4}$ ( $1,31 \times 10^{-4}$ )	$7,6 \times 10^{-2}$ ( $4,97 \times 10^{-4}$ )

Uma análise dos resultados obtidos mostrou que a rede NARX-ELM conseguiu o melhor resultados para as séries de Hénon e de MG. O mesmo não aconteceu para a série do laser caótico, visto que as redes ELM não conseguiram fazer predições mais corretas por um horizonte maior do que a rede NARX treinada com *backpropagation*.

## 5 Conclusões

Neste trabalho são avaliados os desempenhos das arquiteturas de RNAs baseadas em projeções aleatórias (rede ELM) e suas modificações na predição  $H$ -passos-adiante (recursiva) da série de Hénon, de MG e do laser caótico. Os resultados obtidos foram construídos a partir de uma heurística que possibilita encontrar os melhores parâmetros para cada modelo.

O destaque das arquiteturas baseadas no modelo NARX na predição recursiva pode ser justificado pela formulação dos regressores. O regressor de entrada supre informações de médio a longo prazo sobre o comportamento dinâmico da série temporal, uma vez que o atraso  $\tau$  é geralmente maior do que a unidade, enquanto o regressor de saída, supre informações de curto prazo acerca da série temporal.

Os baixos desempenhos das redes baseadas em projeções aleatórias obtidos para a série do laser caótico, estão em consonância com indícios semelhantes reportados por Sheng et al. (2012) e Wallace et al. (2013) para modelagem e predição de séries temporais. Nestes trabalhos tem sido observado o baixo desempenho das redes de projeções aleatórias para as séries reais, séries ruidosas ou com memória longa.

Em trabalhos futuros pretende-se melhorar o desempenho da rede ELM, procurando melhorar o desempenho na predição da série do laser caótico, em séries reais ou séries ruidosas.

## Referências

- Elman, J. (1990). Finding structure in time, *Cognitive Science* **14**: 179–211. DOI: [10.1207/s15516709cog1402\\_1](https://doi.org/10.1207/s15516709cog1402_1)
- Huang, G.-B., Wang, D. H. and Lan, Y. (2011). Extreme learning machines: A survey, *International Journal of Machine Learning and Cybernetics* **2**(2): 107–122. DOI: [10.1007/s13042-011-0019-y](https://doi.org/10.1007/s13042-011-0019-y)
- Miche, Y., Schrauwen, B. and Lendasse, A. (2010). Machine learning techniques based on random projections, in M. Verleysen (ed.), *ESANN2010: 18th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, Bruges, Belgium, pp. 295–302.
- Narendra, K. S. and Parthasarathy, K. (1990). Identification and control of dynamical systems using neural networks, *IEEE Transactions on Neural Networks* **1**(1): 4–27. DOI: [10.1109/72.80202](https://doi.org/10.1109/72.80202)
- Palit, A. K. and Popovic, D. (2005). *Computational Intelligence in Time Series Forecasting*, 1st edn, Springer Verlag.
- Principe, J. C., Euliano, N. R. and Lefebvre, W. C. (2000). *Neural Adaptive Systems: Fundamentals Through Simulations*, John Wiley and Sons.
- Sheng, C., Zhao, J., Liu, Y. and Wang, W. (2012). Prediction for noisy nonlinear time series by echo state network based on dual estimation, *Neurocomputing* **82**: 186–195. DOI: [10.1016/j.neucom.2011.11.021](https://doi.org/10.1016/j.neucom.2011.11.021)
- Singh, R. and Balasundaram, S. (2007). Application of extreme learning machine method for time series analysis, *International Journal of Computer Systems Science and Engineering* **2**(4): 256–262.
- Sovilj, D., Sorjamaa, A., Yu, Q., Miche, Y. and Séverin, E. (2010). OPELM and OPKNN in long-term prediction of time series using projected input data, *Neurocomputing* **73**(10–12): 148–156.
- Takens, F. (1981). Detecting strange attractors in turbulence, in D. A. Rand and L.-S. Young (eds), *Dynamical Systems and Turbulence*, Vol. 898 of *Lecture Notes in Mathematics*, Springer, pp. 366–381.
- Wallace, E., Maei, H. R. and Latham, P. E. (2013). Randomly connected networks have short temporal memory, *Neural Computing* **25**(6): 1408–1439. DOI: [10.1162/NECO\\_a\\_00449](https://doi.org/10.1162/NECO_a_00449)