

PREDIÇÃO DE DESLOCAMENTO DE BLOCOS DA BARRAGEM DE ITAIPU VIA MÉTODO HÍBRIDO WAVELET

Jairo Marlon Corrêa (UTFPR) jairocorrea@utfpr.edu.br

Anselmo Chaves Neto (UFPR) anselmo@ufpr.br

Luiz Albino Teixeira Júnior (UNILA) luiz.a.t.junior@gmail.com

Edgar Manoel Careño Franco (UNIOESTE) pgesde@unioeste.br

Resumo: Dada a relevância da predição dos dados temporais de deslocamento de blocos na barragem de Itaipu, localizada em Foz do Iguaçu, Paraná, Brasil, na execução de atividades preventivas e corretivas (Itaipu, 2015), este artigo propõe um método híbrido wavelet chamado de *ArimaX-Garch Wavelet Neural* (AXGWN) para tal fim. A fim de se produzir suas previsões, o predictor AXGWN considera informações sobre as estruturas de autodependência linear e não-linear (incluindo, a volatilidade) exibidas pelos dados. Para ilustrar o método proposto em um caso aplicado real, um estudo de caso utilizando uma série temporal diária da usina de Itaipu foi efetuado. Todas as análises numéricas mostram que a metodologia AXGWN alcançou maior acurácia preditiva que a tradicional abordagem Arima-Garch.

Palavras-chave: Decomposição Wavelet, ArimaX-Garch, Redes Neurais Artificiais.

PREDICTION OF DISPLACEMENT OF BLOCKS OF ITAIPU DAM VIA WAVELET HYBRID METHOD

Abstract: Given the relevance of the temporal data prediction of block displacements at Itaipu dam, located in Foz do Iguaçu, Paraná, Brazil, in the execution of preventive and corrective activities (Itaipu, 2015), this paper puts forward a hybrid wavelet method called *Neural ArimaX-Garch Wavelet* (NWAXG) for this purpose. In order to produce its forecasts, the NWAXG predictor accounts for information on linear and non-linear (including volatility) auto-dependence structures exhibited by the data. For illustrating the proposed method in a real applied case, a case study using a daily time series of Itaipu plant has been performed. All numerical analysis show that the methodology NWAXG achieved greater predictive accuracy than the traditional Arima-Garch approach.

Keywords: Wavelet Decomposition, ArimaX-Garch, Artificial Neural Networks.

1. INTRODUÇÃO

Desde os primórdios das civilizações, o uso da energia hidráulica teve importância devido à disponibilidade de recursos, facilidade de aproveitamento e, especialmente, seu caráter renovável. É sabido que a energia hidrelétrica contribuiu efetivamente para o desenvolvimento econômico de forma significativa como em atividades agrícolas, de serviços, comerciais e, também, no aumento do conforto das residências e aumento da qualidade de vida das pessoas. A energia hidráulica representa cerca de 20%, atualmente, de toda a eletricidade gerada no planeta; é utilizada como principal fonte na produção de energia elétrica em mais de 30 países, dentre os quais se destacam como maiores produtores mundiais: Canadá, China, Brasil, Estados Unidos e Rússia.

Um belo exemplo da obra prima da engenharia, a usina hidrelétrica de Itaipu, é resultado dos esforços e do comprometimento de dois países vizinhos, Brasil e Paraguai. A Itaipu binacional é um dos maiores projetos hidrelétricos do mundo, considerando-se a produção anual. Localiza-se no rio Paraná a 14 km à montante da ponte internacional que liga a cidade de Foz do Iguaçu, no Brasil, à Ciudad del Este no Paraguai. O projeto da barragem da Itaipu foi bastante criterioso tanto em termos de investigação da rocha de fundação quanto dos materiais utilizados na construção das barragens, seguindo um minucioso controle de qualidade, além disso foram instalados aproximadamente 2400 instrumentos, cujas leituras são feitas desde a época da construção, e um sistema de drenagem com mais de 5000 drenos instalados em suas estruturas e fundações, formando um banco de dados de mais de 30 anos.

As campanhas de leituras e medições tem uma periodicidade que varia de semanal até semestral, conforme o tipo de instrumento.

Alguns blocos são designados blocos chave, pois são dotados de maior quantidade de instrumentos devido a sua importância para a região monitorada e características predefinidas em projeto. Em 2005, como parte de modernização da usina, foi implantado um sistema de aquisição automática dos dados, chamado *Automatic Data Acquisition System* (ADAS) (Itaipu, 2015), onde 10% dos instrumentos mais importantes foram automatizados, passando a ter leituras a cada 30 minutos, além da leitura manual já existente.

A barragem principal da usina de Itaipu é de concreto, do tipo gravidade aliviada. É composta por 18 blocos, onde estão as tomadas d'água de 16 unidades geradoras (Itaipu, 2014). A água do reservatório exerce uma pressão na face montante da barragem e também exerce, nas partes inferiores dos blocos, uma subpressão ou empuxo hidrostático que cria um efeito contrário ao que a sua própria massa exerce sobre a fundação. Devido a essas duas forças, a barragem tem uma tendência a apresentar pequenos deslocamentos horizontais, principalmente no sentido montante-jusante. Em contrapartida, as forças estabilizantes ou a favor da segurança são o peso próprio da estrutura e o atrito entre a barragem e a rocha de fundação. De acordo com (Itaipu, 2009) foram instalados dois tipos de pêndulos em trechos da Usina Hidrelétrica de Itaipu: o pêndulo direto e o invertido. O pêndulo direto é responsável em monitorar os deslocamentos horizontais de pontos dos blocos instrumentados da barragem em determinadas cotas, em relação à fundação da estrutura. Já o pêndulo invertido monitora os deslocamentos da fundação da barragem em relação ao contato concreto/rocha ou a alguma cota determinada no bloco de concreto.

É fato que a barragem acumula um pequeno deslocamento ao longo de 32 anos desde o enchimento do reservatório, porém dentro dos valores previstos em projeto. A análise dos dados dos pêndulos mostra que o deslocamento descreve um movimento que gera uma curva cujo comportamento é cíclico, influenciado pela temperatura ambiente. É observável, por meio do gráfico em planta nos pêndulos, que alguns blocos já se deslocaram cerca de 13mm em relação à fundação considerando o período de tempo desde o enchimento do lago. Esses deslocamentos são considerados pequenos e estão dentro dos limites de segurança estabelecidos no projeto. Mesmo assim, a equipe técnica da Itaipu vem acompanhando periodicamente as leituras de todos os instrumentos e incentivado estudos para aplicação de ferramentas matemáticas para atuar preventivamente ante quaisquer riscos potenciais.

Diante disso, diversos autores afirmam "Os maiores desastres de barragens de gravidade ocorreram por resistência inadequada ao movimento horizontal, isto é, ruptura por cisalhamento na fundação". Uma das grandes forças atuantes na barragem é a subpressão, a qual ocorre no sentido ascendente e que reduz o peso efetivo da estrutura. Tais efeitos podem gerar pequenos deslizamentos existentes no maciço rochoso.

Neste estudo, avalia-se a implementação de um modelo de previsão do deslocamento de um bloco da barragem principal da Usina Hidrelétrica de Itaipu por meio do qual seja possível realizar a análise do comportamento da estrutura em torno deste bloco. Tais informações permitirão medidas corretivas em planos de execução e/ou tomada de decisões já adotadas por engenheiros e técnicos especialistas. Diante disso, também será possível avaliar os demais blocos da usina com a aplicação do método proposto na previsão das demais séries de tempo, uma vez que os resultados poderão operar em benefício da análise do desempenho estrutural da barragem da Usina Hidrelétrica de Itaipu. Assim sendo, neste artigo, é proposta uma metodologia preditiva não-linear híbrida chamada de *ArimaX-Garch Wavelet Neural* (AXGWN) cujo objetivo é gerar previsões para uma série temporal de deslocamento de um bloco da Usina Hidrelétrica de Itaipu, localizada em Foz do Iguaçu, Paraná, Brasil.

Assim, além da introdução o artigo está dividido em quatro outras seções. Na Seção 2, é feita uma revisão sobre a Análise Wavelet e modelos Arima-Garch. Na Seção 3, é detalhada a metodologia proposta (AXGWN). Na Seção 4, são apresentados os principais resultados obtidos a partir da modelagem via modelos Arima-Garch e AXGWN de uma série temporal de deslocamento de barragens da Usina de Itaipu. E, por fim, na Seção 5, o artigo é concluído.

2. REVISÃO DA LITERATURA

2.1 Decomposição Wavelet

Seja y_t ($t = 1, \dots, T$) uma série temporal de tamanho igual a T , onde $T > 1$. Segundo Teixeira Júnior et al. (2015), uma *decomposição wavelet* de cada estado y_t é dada, genericamente, em (1)

$$y_t = y_{t_{A_{m_0}}} + \sum_{m=m_0}^{\infty} y_{t_{D_m}} \quad (1)$$

Onde:

(i) $y_{t_{A_{m_0}}} := \sum_{n \in \mathbb{Z}} a_{m_0,n} \phi_{m_0,n}(t)$ e $y_{t_{D_m}} := \sum_{n \in \mathbb{Z}} d_{m,n} \omega_{m,n}(t)$ são, respectivamente, as *componentes wavelet* (CW) de aproximação de nível m_0 e de detalhe de nível m do estado y_t ;

(ii) $a_{m_0,n}$ e $d_{m,n}$ são, respectivamente, os coeficientes wavelet de aproximação (de níveis m_0 e n) e de detalhe (de níveis m e n), os quais são, respectivamente, definidos pelo produto interno usual entre y_t ($t = 1, \dots, T$) e $\phi_{m_0,n}(\cdot)$ ($t = 1, \dots, T$) - isto é, $a_{m_0,n} := \sum_{t=1}^T f(t) \phi_{m_0,n}(t)$ - e entre y_t ($t = 1, \dots, T$) e $\omega_{m,n}(t)$ ($t = 1, \dots, T$) - isto é, $d_{m,n} := \sum_{t=1}^T f(t) \omega_{m,n}(t)$ -, onde $m_0, m, n \in \mathbb{Z}$; e

(iii) $\phi_{m_0,n}(t) := 2^{\frac{m_0}{2}} \phi(2^{m_0}(t) - n)$ consiste em uma *função escala* gerada por superposições de uma função escala original $\phi(\cdot)$, onde $n, m_0 \in \mathbb{Z}$; e $\omega_{m,n}(\cdot) := 2^{\frac{m}{2}} \omega(2^m(\cdot) - n)$, onde $m, n \in \mathbb{Z}$, consiste em uma *função wavelet* gerada por superposições de uma função wavelet original $\omega(\cdot)$ (Mallat, 2009; Teixeira Júnior et al., 2015).

Em termos práticos, para decompor séries temporais, é usual se utilizar uma decomposição wavelet de nível r , a qual consiste, basicamente, na decomposição wavelet descrita em (1), com um truncamento na r -ésima componente de detalhe - onde é usual que m_0 é assumido ser igual a r (Teixeira Jr et al., 2012). Com efeito, em uma decomposição wavelet de nível 2 (isto é, $r = 2$), onde o truncamento, em (1), ocorre na segunda componente de detalhe, adota-se $m_0 = 2$, de modo que o estado y_t é expandido tal como: $y_t = y_{t_{A_2}} + y_{t_{D_2}} + y_{t_{D_3}} + \dots + \varepsilon_t$, em que ε_t é o erro de aproximação em t .

2.2 Redes Neurais Artificiais

De acordo com Haykin (2001), as *Redes Neurais Artificiais* (RNA) são sistemas paralelos compostos por unidades de processamento simples (chamados de neurônios artificiais) e dispostas em uma ou mais camadas interligadas por um grande número de conexões (chamadas de sinapses). Geralmente, uma RNA é unidirecional (feed-forward) e com pesos sinápticos para ponderar as entradas recebidas por cada neurônio artificial. Na Figura 1, tem-se uma ilustração da arquitetura de uma *RNA feed-forward multicamadas perceptron* (RNA-MLP) com três camadas: camada de entrada (input), camada escondida (ou intermediária, ou oculta) e camada de saída (output).

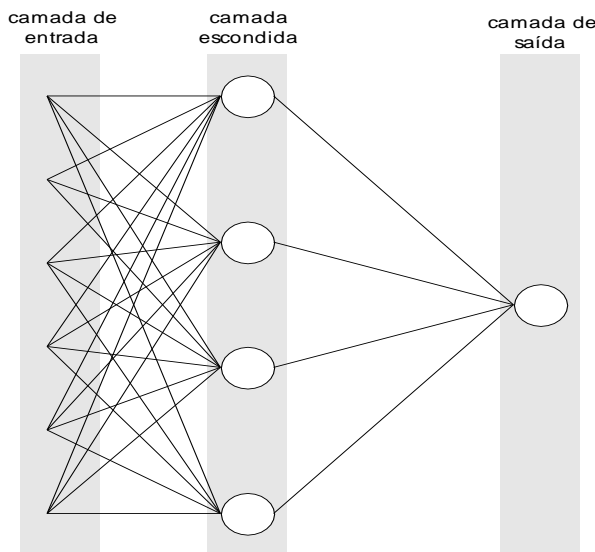


Figura 1 - RNA-MLP com 3 camadas

A primeira camada da RNA-MLP consiste na a camada de entrada, sendo a única exposta aos padrões (ou dados, ou sinais) de entrada. Ela transmite os valores dos padrões de entrada para os neurônios da camada intermediária para que estes façam transformações não-lineares e transmitam os resultados destas para a camada de saída (última camada da RNA-MLP).

O processo de treinamento de uma RNA-MLP consiste no ajuste numérico ótimo dos pesos sinápticos (que realizam as ponderações dos sinais recebidos por cada neurônios artificial). Ao longo do treinamento, vários números de padrões de entrada e de neurônios na camada escondida são testados empiricamente, de forma a minimizar a soma dos erros quadráticos entre a resposta produzida pela RNA-MLP e os valores desejados (que são os estados da série temporal disponível, denotada por y_t ($t = 1, \dots, T$)). Tais valores produzidos pela RNA-MLP, no treinamento, consistem nas previsões *in-sample* para y_t ($t = 1, \dots, T$), denotadas por \hat{y}_t ($t = 1, \dots, T$).

Na Figura 2, tem-se uma ilustração de um conjunto de treinamento para um processo autoregressivo não-linear de ordem 4. Note que a construção dos padrões de treinamento da RNA-MLP consiste em mover as janelas de entrada e saída ao longo de toda série temporal, de modo que cada par entrada/saída funciona como um padrão de treinamento e deve ser apresentado repetidas vezes até que o algoritmo de aprendizado convirja (Haykin, 2001).

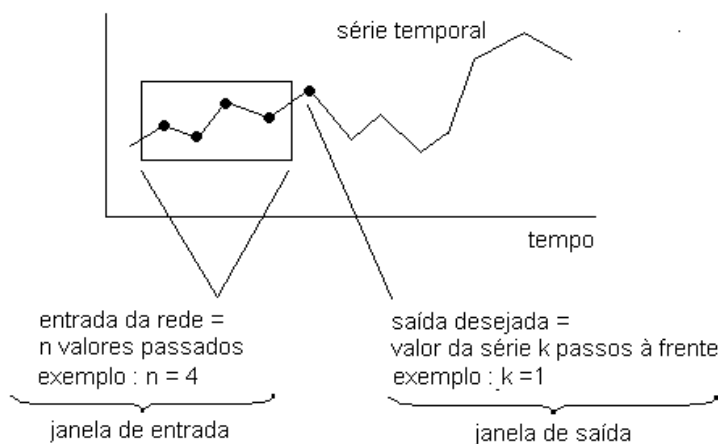


Figura 2 – Treinamento de um processo autoregressivo de ordem 4.

Com efeito, em um processo autoregressivo não-linear de ordem L , a camada de entrada é formada pelos L valores defasados do estado que se deseja prever. Ou seja, para prever o estado y_t , a sua previsão \hat{y}_t é tal que $\hat{y}_t := f(y_{t-1}, y_{t-2}, \dots, y_{t-L})$, onde $f(\cdot)$ consiste na RNA-MLP (que, na verdade, é um modelo não-linear, pelo fato de as transformações realizadas nos neurônios na camada intermediária serem não-lineares). Para mais detalhes, veja Zhang (2003) e Campos (2010).

2.3 Modelo ArimaX-Garch

Seja y_t ($t = 1, \dots, T$) uma série temporal que exhibe estrutura de autodependência linear. Baseado em Box & Tiao (1975), cada estado y_t pode ser representado por um modelo ArimaX (p, d, q), conforme em (2).

$$\nabla^d y_t = \phi_0 + \sum_{j=1}^p \phi_j y_{t-j} + \sum_{i=1}^q \theta_i e_{t-i} + \sum_{i=1}^{r+1} \sum_{l_i=0}^{L_i} \alpha_{il_i} x_{i,t-l_i} + e_t \quad (2)$$

onde: B é o operador de atraso, definido por $B^k y_t := y_{t-k}$, onde k pertence a \mathbb{Z} ; $\nabla^d := (1 - B)^d$ é o operador diferença, onde d é a ordem da diferenciação; $(\phi_i)_{i=0}^p$ e $(\theta_j)_{j=1}^q$ são listas ordenadas de parâmetros complexos, com $\phi_p \neq 0$ e $\theta_q \neq 0$, e α_{il_i} é o parâmetro complexo associado à componente exógena $x_{i,t-l_i}$; e_t é uma inovação no tempo t que consiste em uma realização de uma variável aleatória ε_t , onde ε_t ($t = 1, \dots, T$) é um processo estacionário em média zero e covariância constante; e p e q são, respectivamente, as ordens da parte autoregressiva $\sum_{j=1}^p \phi_j y_{t-j}$ (notação: AR(p)) e da parte de médias móveis $\sum_{i=1}^q \theta_i e_{t-i}$ (notação: MA(q)); e L_i é a ordem máxima do lag em $\sum_{l_i=0}^{L_i} \alpha_{il_i} x_{i,t-l_i}$ da variável exógena $x_{i,t}$, onde $i = 1, \dots, (r + 1)$.

Note que o termo do erro e_t , em t , é a realização de uma variável randômica ε_t , com média zero e variância constante, isto é, $\sigma_t^2 = \sigma^2$. Em alguns casos a heteroscedasticidade pode depender dos valores observados do passado da série temporal ou de valores passados da sua variância, de tal modo que um modelo AR, bem como suas extensões, pode ser adaptado para projetar a variação temporal. Desta forma, incondicionalmente, a variância é constante, mas condicionada a esses valores passados, poderá mudar temporalmente. Um modelo vastamente utilizado em sua modelagem é o chamado *Heteroscedasticidade Condicional Autoregressiva* (GARCH). Segundo Bollerslev (1990), o modelo geral GARCH (P, Q) para a variância condicional σ_t^2 de inovações ε_t é dado em (3).

$$\sigma_t^2 = \gamma_0 + \sum_{j=1}^P \gamma_j \sigma_{t-j}^2 + \sum_{i=1}^Q \delta_i e_t^2, \quad (3)$$

Em (3), as seguintes restrições tem de ser atendidas: $\sum_{j=1}^P \gamma_j + \sum_{i=1}^Q \delta_i < 1$; $\gamma_0 > 0$; e $\gamma_j \geq 0$ ($j=1, \dots, P$) e $\delta_i \geq 0$ ($i=1, \dots, Q$) (Hamilton, 1994).

Um modelo composto por (2) e (3) é chamado modelo ArimaX-Garch. Assim, para se obter o melhor modelo ArimaX-Garch, três passos básicos devem ser executados: no passo 1, testar os valores plausíveis para os parâmetros p, d, q, I, L_i , em (2), e os parâmetros P e Q , em (3), baseado na meio da análise do gráfica das funções de autocorrelação simples e parcial dos erros de previsão padronizados ordinários, z_t ($t = 1, \dots, T$), e quadráticos, z_t^2 ($t = 1, \dots, T$) (Bollerslev, 1990; Engle, 1982); no passo 2, definir o método de estimação dos parâmetros do modelo ArimaX-Garch (o método mais comum é o de Máxima Verossimilhança (Hamilton, 1994)); e, no passo 3, analisar os testes de diagnósticos para validar (ou não) o modelo ArimaX-Garch estimado.

3. METODOLOGIA PROPOSTA

Seja y_t ($t = 1, \dots, T$) uma série temporal com estrutura de autodependência linear e não-linear a ser projetada. A representação por meio de fluxograma do método AXGWN é dado na Figura 3.

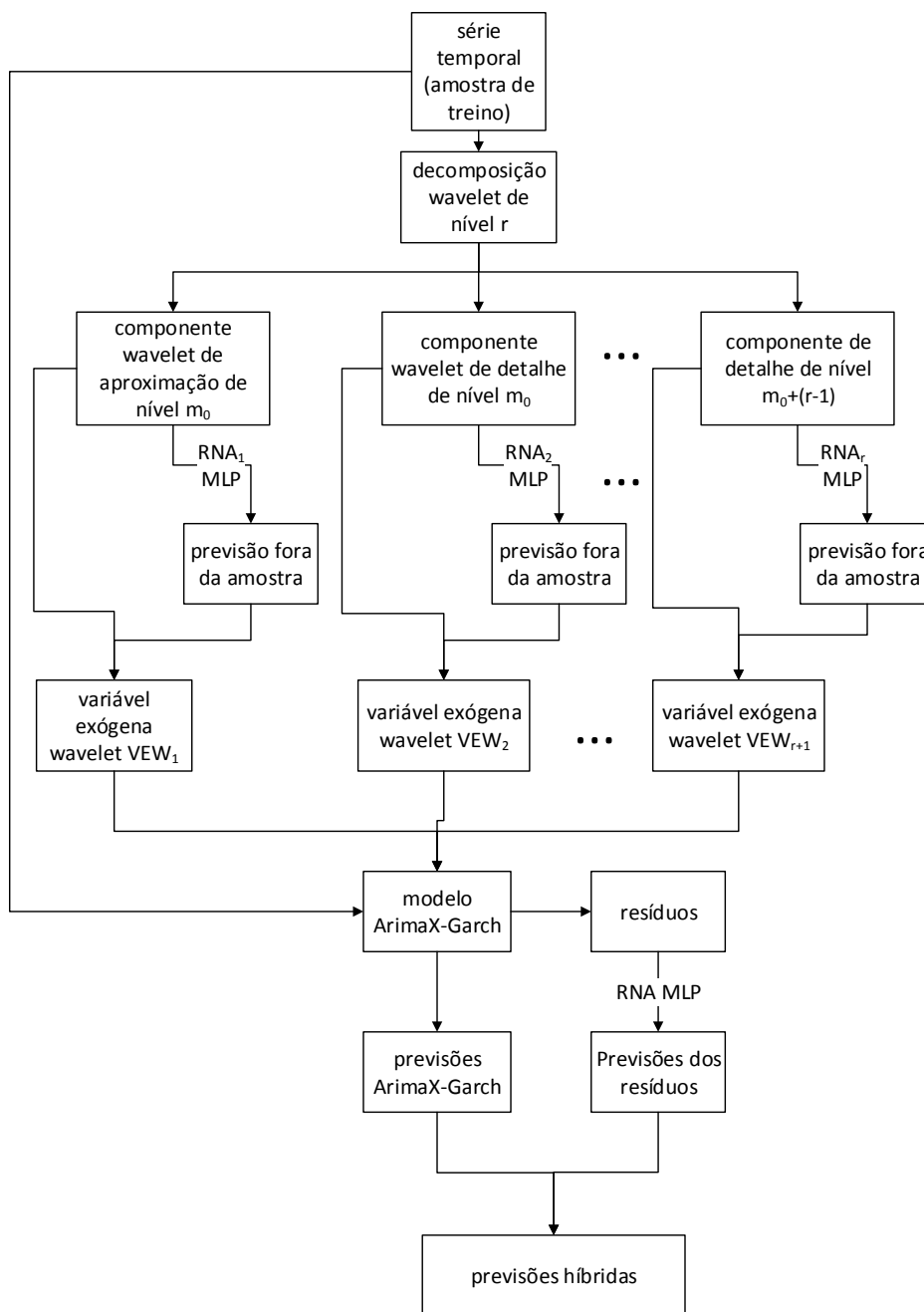


Figura 3 - Fluxograma do método AXGWN.

Mais detalhadamente, o método *ArimaX-Garch Wavelet Neural* (AXGWN) consiste na execução dos seis passos a seguir. (1) realiza-se uma decomposição wavelet de nível r de y_t ($t = 1, \dots, T$), produzindo $r+1$, representadas pela lista $\left((x_{1,t})^T_{t=1}, \dots, (x_{(r+1),t})^T_{t=1} \right)$; (2) cada CW produzida no passo (1) é modelada, individualmente, por meio de uma RNA-MLP, a fim de se gerar previsões fora da amostra; (3) As CWs do passo 1 são completadas pelas respectivas previsões geradas no passo 2, obtendo-se as Variáveis Exógenas Wavelet (VEWs), dotadas de informações da frequência espectral (Mallat, 2009) associadas a y_t ($t =$

1, ..., T)); (4) é feita a projeção da série temporal subjacente através de um modelo ArimaX-Garch utilizando as VEWs produzidas no passo (3); (5) os erros de previsão da amostra de treino do modelo ArimaX-Garch são projetados através uma RNA-MPL; e (6) as previsões oriundas dos passos 4 e 5 são somadas, para cada instante t , gerando as previsões híbridas de y_t ($t = 1, \dots, T$).

4. RESULTADOS NUMÉRICOS

Nesta seção, os principais resultados numéricos de uma aplicação do modelo Arima-Garch tradicional (benchmark) e do método AXGWN (proposto) a uma série temporal de aferições diárias do deslocamento de um bloco da barragem da usina de Itaipu são mostrados. Assim, na Seção 4.1, a série temporal subjacente é descrita, a fim de se justificar a utilização dos métodos preditivos, além de apresentar os testes residuais utilizados na etapa de modelagem dos dados. Nas Seções 4.2 e 4.3 são tratadas, respectivamente, informações acerca do modelo Arima-Garch e do método AXGWN estimados. Na seção 4.4, são comparados os resultados numéricos obtidos.

4.1 Série temporal de deslocamento

Uma série temporal de deslocamento físico de um bloco da barragem que abastece a usina hidrelétrica de Itaipu é modelada devido às suas propriedades estatísticas e à sua relevância. Na figura 4, o seu gráfico mostra o comportamento temporal das 2.554 observações, em intervalos diários, os quais foram coletados do sistema automatizado ADAS.

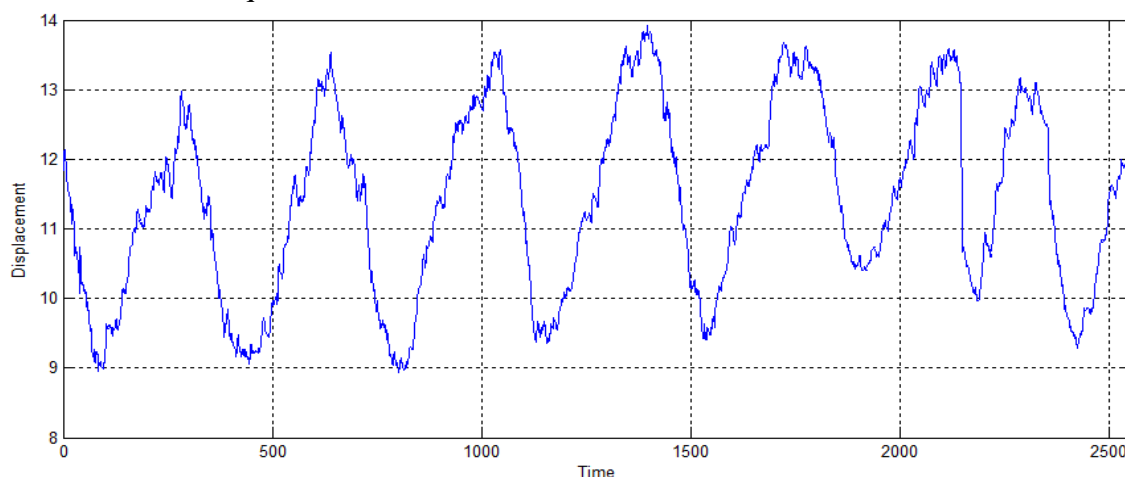


Figura 4 - Série temporal diária de deslocamento físico de um bloco da barragem da usina de Itaipu.

Os referidos dados temporais compreendem o período de 28 de outubro de 2005 a 24 de outubro de 2012, totalizando 2554 pontos. Destes, as 2506 primeiras observações foram utilizadas para obtenção dos parâmetros ótimos dos dois métodos preditivos envolvidos (em outras palavras, tais dados compõem a amostra de treino); enquanto que as 48 observações subsequentes formam a amostra de teste.

Considerando o teste de raiz unitária de Augmented *Dickey-Fuller* (ADF) (Hamilton, 1994), não se rejeitou a hipótese nula de não estacionariedade de segunda ordem da série temporal em questão, ao nível de 1% de significância. Por outro lado, ao considerar a série transformada com a aplicação do logaritmo, seguida da diferenciação, rejeitou-se fortemente a hipótese nula, ao nível de 1% de significância. As funções de autocorrelação simples e parcial (Box et al., 2008), referidas aqui como FAC e FACP, respectivamente, de sua versão transformada, bem como de sua versão quadrática exibiu valores de autocorrelações simples e parciais significativas, ao nível de 1%, cuja volatilidade é capturada com o uso do modelo

Arima-Garch. Uma vez que não havia variáveis exógenas adequadas disponíveis para serem usadas pelo modelo ArimaX-Garch, foi adotado o modelo Arima-Garch como benchmark.

É importante salientar ainda que foram produzidas, pelos modelos Arima-Garch (benchmark) e AXGWN (o qual é proposto aqui) somente previsões 1 passo à frente, para um horizonte de previsão igual a 48 passos à frente. Para os diagnósticos dos resíduos, na etapa de modelagem, foram utilizados os seguintes critérios: teste de Ljung-Box (para verificação de existência de autocorrelação significativa até o lag 36); estatística de Durbin-Watson (para analisar a evidência empírica de autocorrelação de lag 1); análises gráficas da FAC e da FACP (para visualizar, respectivamente, a presença de autodependência linear simples e parcial significativas, até o lag 36); teste ARCH (para verificação de evidência de heteroscedasticidade condicional significativa, até o lag 36) (Hamilton, 1994). Todos os modelos Arima-Garch, bem como as suas análises residuais, foram executadas no software EViews 8 (www.eviews.com/).

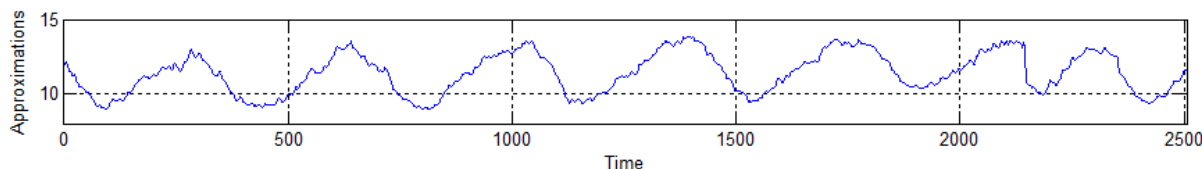
4.2 Modelo Arima-Garch tradicional

O modelo Arima-Garch com maior aderência à série temporal de deslocamento considerada foi o Arima-Garch (3,2,4) (2,1), com transformação logarítmica seguida por dupla diferenciação. Todas as estimativas de máxima verossimilhança seus parâmetros, tanto da equação da média condicional e quanto da variância condicional, foram estatisticamente significativas, ao nível de 1% e satisfaziam às condições de estacionariedade e invertibilidade. Em todos os testes residuais, aceitou-se a hipótese de inexistência de autocorrelação nos resíduos padronizados ordinários e quadráticos, a 1% de significância - o que significa que o modelo Arima-Garch capturou uma estrutura de autodependência plausível. Em particular, a estatística Durbin-Watson, calculada a partir dos resíduos ordinários padronizados, obteve valor igual a 2,092367, que sugere a inexistência de autocorrelação de lag 1 (por ser muito próxima ao valor referência 2).

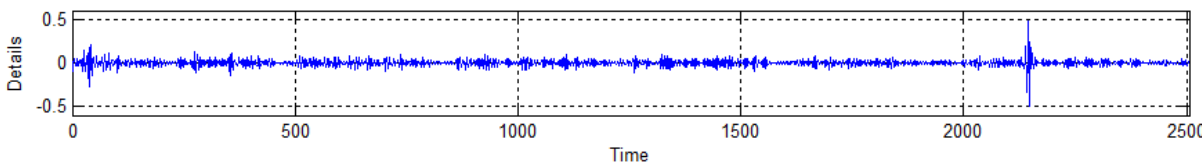
Salienta-se ainda que, dentre todos os modelos Arima-Garch plausíveis identificados, foram adotadas, como critério de seleção, as seguintes estatísticas de aderência, na amostra de treinamento: *Absolute Percentage Error* (APE), *Mean Absolute Percentage Error* (MAPE), *Mean Absolute Error* (MAE) e *explanation coefficient* (R^2) (Hamilton, 1994).

4.3 Método AXGWN

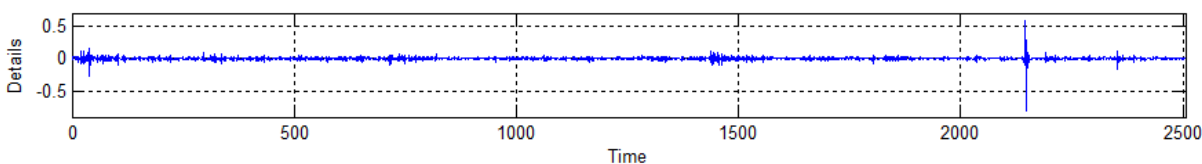
Para a implementação do método AXGWN, foi utilizado o software MATLAB versão 2013a (www.mathworks.com/) e o software EViews 8 (www.eviews.com/). Primeiramente, foi realizada, no MATLAB, uma decomposição wavelet de nível 2 da amostra de treino da série temporal y_t ($t = 1, \dots, 2506$) de deslocamento em questão, de maneira a gerar: uma CW de aproximação de nível 2, a qual é denotada por $y_{A_2,t}$ ($t = 1, \dots, 2506$); duas CWs de detalhes de níveis 2 e 3, as quais são denotadas, respectivamente, por $y_{D_2,t}$ ($t = 1, \dots, 2506$) e $y_{D_3,t}$ ($t = 1, \dots, 2506$). Os seus gráficos podem ser visualizados na Figura 5. Para tal, foram utilizadas as funções originais escala $\phi(\cdot)$ e wavelet $\omega(\cdot)$ da família de Daubechies com momento nulo igual a 40 (Mallat, 2008). O critério de escolha da família wavelet foi baseado na minimização de $\sum_{t=1}^{2506} \varepsilon_t^2$, onde: $y_t = y_{A_2,t} + y_{D_2,t} + y_{D_3,t} + \varepsilon_t$, para todo $t=1, \dots, 2506$. Foram testadas as famílias de Daubechies de momento nulo igual a 1 até 45 (ou seja, 45 diferentes famílias).



(a) Componente wavelet de aproximação de nível 2, $y_{A_2,t}$ ($t = 1, \dots, 2506$).



(b) Componente wavelet de detalhes de nível 2, $y_{D_2,t}$ ($t = 1, \dots, 2506$).



(c) Componente wavelet de detalhes de nível 3, $y_{D_3,t}$ ($t = 1, \dots, 2506$).

Figura 5 - Componentes wavelet da série de deslocamento de um bloco de barragem.

No passo 2 do método AXGWN, cada uma das três componentes wavelet $y_{A_2,t}$, $y_{D_2,t}$ e $y_{D_3,t}$ ($t = 1, \dots, 2506$) foi modelada usando uma RNA-MLP (descrita na seção 2.3), a fim de se produzir as seguintes seqüências de previsões, 48 passos fora da amostra (isto é, $h=48$): $\hat{y}_{A_2,t}$, $\hat{y}_{D_2,t}$ e $\hat{y}_{D_3,t}$ ($t = 2507, \dots, 2554$). A Tabela 1 apresenta as configurações ótimas das RNA-MLP's de cada uma das três CWs.

Tabela 1 – Configurações das RNA-MLPs para as componentes wavelet.

| <i>Parâmetros da arquitetura neural</i> | <i>Componente de aproximação nível 2</i> | <i>Componente de detalhe nível 2</i> | <i>Componente de detalhe nível 3</i> |
|---|--|--------------------------------------|--------------------------------------|
| Transformação | premnmx | premnmx | premnmx |
| Tamanho de janela de entrada | 5 | 2 | 2 |
| Número de camadas escondidas | 1 | 1 | 1 |
| Número de neurônio na camada escondida | 3 | 3 | 5 |
| Função de ativação da camada oculta | Tangente hiperbólica | Tangente hiperbólica | Tangente hiperbólica |
| Camada de saída | Função de ativação linear | Função de ativação linear | Função de ativação linear |
| Algoritmo de treinamento | Levenberg-Marquardt (Millar, 2011) | Levenberg-Marquardt (Millar, 2011) | Levenberg-Marquardt (Millar, 2011) |

No passo 3, as três componentes wavelet do passo 1 são completadas pelas 48 previsões geradas no passo 2, gerando as três componentes wavelet completadas (denotadas por $\tilde{y}_{A_2,t,C}$, $\tilde{y}_{D_2,t,C}$ e $\tilde{y}_{D_3,t,C}$ ou, equivalentemente, por x_1 , x_2 e x_3). Estas são interpretadas como se fossem, de fato, variáveis exógenas convencionais. No passo 4, por sua vez, tais variáveis exógenas artificiais wavelet são utilizadas em um modelo Arima-Garch. A sua componente ArimaX é dada por:

$$\hat{y}_t = 1,38298232989 x_{1,t-1} - 0,383871938428 x_{1,t-2} + 1,22288709773 x_{2,t-1} - 0,488659896985 x_{3,t-2} - 0,235082399151 y_{t-1} + 0,235082399151 y_{t-3} + 0,721450081095 y_{t-4} - 0,23553545046 y_{t-5} + 0,0617467629966 \hat{e}_{t-2} - 0,489351187041 \hat{e}_{t-3} - 0,475632809678 \hat{e}_{t-4} + 0,363586761893 \hat{e}_{t-5} + 2,56991622814 \hat{\sigma}_t^2. \tag{4}$$

Note, em (4), que todas as três variáveis exógenas artificiais wavelet obtidas no passo 2 foram utilizadas, porém na sua forma defasada. Além disso, a previsão da variância $\hat{\sigma}_t^2$, gerada no modelo (5), fornece um efeito *redes neurais artificiais* para a construção da previsão da média condicional \hat{y}_t .

Por sua vez, a sua componente Garch é dada pelo modelo Garch (1,1), com distribuição GED (Hamilton, 1994) para os resíduos padronizados, a qual é dada em (5).

$$\hat{\sigma}_t^2 = 0.0005577966714 + 0.03470971923 (\hat{e}_{t-1})^2 + 0.8038587221 \hat{\sigma}_{t-1}^2. \tag{5}$$

O método da Máxima Verossimilhança (veja Millar, 2011) foi usado para se obter as estimativas do modelo Arima-Garch, em (4) e (5). Todas as estimativas foram estatisticamente significantes, ao nível de 1%, e satisfaziam às condições de estacionariedade e invertibilidade.

Foram analisadas as funções FAC e a FACP (até o lag 36) dos resíduos padronizados ordinários e quadráticos. Em todos os *lags* até 36, os valores de autocorrelação simples e parcial permaneceram dentro dos intervalos de confiança, com 99% de credibilidade, o que sugere autocorrelações não diferentes de zero, a 1% de significância. Baseado na estatística de Ljung-Box sobre os resíduos padronizados ordinários, na amostra de treino, verificou-se que não havia evidência de autodependência linear e quadrática significativas, ao nível de 1%, até a defasagem 36. O teste Arch confirmou inexistência de evidência empírica, ao nível de 1% de significância, de heteroscedasticidade nos resíduos padronizados ordinários de treino, até o lag 36. Por fim, a estatística Durbin-Watson foi 2.026328 sugere, empiricamente, a inexistência de autocorrelação de lag 1, nos resíduos padronizados, na amostra de treino.

No passo 5, os erros de previsão de treinamento do modelo ArimaX-Garch com *Variáveis Aleatórias Wavelet* (VAWs) foram modelados por uma RNA-MLP, a fim de se identificar uma estrutura não-linear plausível presente nos resíduos - conforme sugere Zhang (2003). Para obter a RNA-MLP, utilizou-se o software MATLAB. As configurações topológicas da RNA-MLP ótima foram: normalização premmx; tamanho de janela igual a 1; 1 camada escondida constituída de 8 neurônios com função de ativação tangente hiperbólica; camada de saída com função de ativação linear; e algoritmo de treinamento de Levenberg-Marquardt (Millar, 2011; Haykin, 2001).

4.4 Comparativos de Desempenho

A Tabela 2 fornece os valores obtidos pelas estatísticas de aderência MAPE e MAE, nas amostras de treinamento e de teste.

Tabela 2 - Performance de previsões dos Arima-Garch e AXGWN.

| MÉTODO | MAPE | | MAE | |
|-------------|-------------------|-----------------|-------------------|-----------------|
| | Dentro da Amostra | Fora da Amostra | Dentro da Amostra | Fora da Amostra |
| Arima-Garch | 0,719% | 0,506% | 0,0813 | 0,0602 |
| AXGWN | 0,406% | 0,295% | 0,0457 | 0,0350 |

Note, na Tabela 2, que o método AXGWN obteve melhor desempenho preditivo em relação ao modelo Arima-Garch, tanto dentro da amostra (amostra de treino) e quanto fora da amostra (amostra de teste). Mais precisamente, tem-se que o método AXGWN produziu um ganho preditivo relativo sobre o modelo Arima-Garch tradicional, quanto à estatística MAPE, igual a 43,53 %, na amostra de treino, e igual a 41,70%, na amostra de teste; enquanto que, em relação à estatística MAE, foi de 43,79%, na amostra de treino, e de 41,87%, na amostra de teste. É desejável que as previsões sejam as mais acuradas possíveis para que os processos decisórios operacionais e financeiros, que por sinal são grandes, sejam executados no momento correto. Salienta-se que a execução de ações operacionais de reparos na barragem ou de investimentos desnecessários ou em períodos inoportunos, pode acarretar, dentre outras coisas, em altas perdas financeiras.

A Figura 6 mostra o gráfico dos *Erros Percentuais Absolutos* (APEs) calculados para as 48 previsões fora da amostra, para ambos os modelos. Note que o modelo AXGWN mostra que os APEs foram baixos do que o modelo Arima-Garch na maioria dos 48 pontos; além de apresentar maior estabilidade (isto é, menores “picos”) na flutuação dos erros. Por exemplo, no nível de 1%, apenas um valor de APE do AXGWN o viola, enquanto que ocorrem 3 violações para o modelo Arima-Garch. Considerando o nível de 0,5%, verificam-se 21 violações para o modelo Arima-Garch e 9, para o AXGWN. Por fim, o APE máximo do modelo Arima-Garch foi aproximadamente igual a 2,25%, enquanto que o APE máximo do AXGWN foi pouco superior a 1% (o que significa um ganho de acurácia preditiva, em termos relativos, em torno de 50%).

Em termos do coeficiente R^2 (que mede a variabilidade explicada pelos modelos, na amostra de treinamento), o modelo Arima-Garch obteve $R^2 = 0,361378$ e o AXGWN, de $R^2 = 0,997249$.

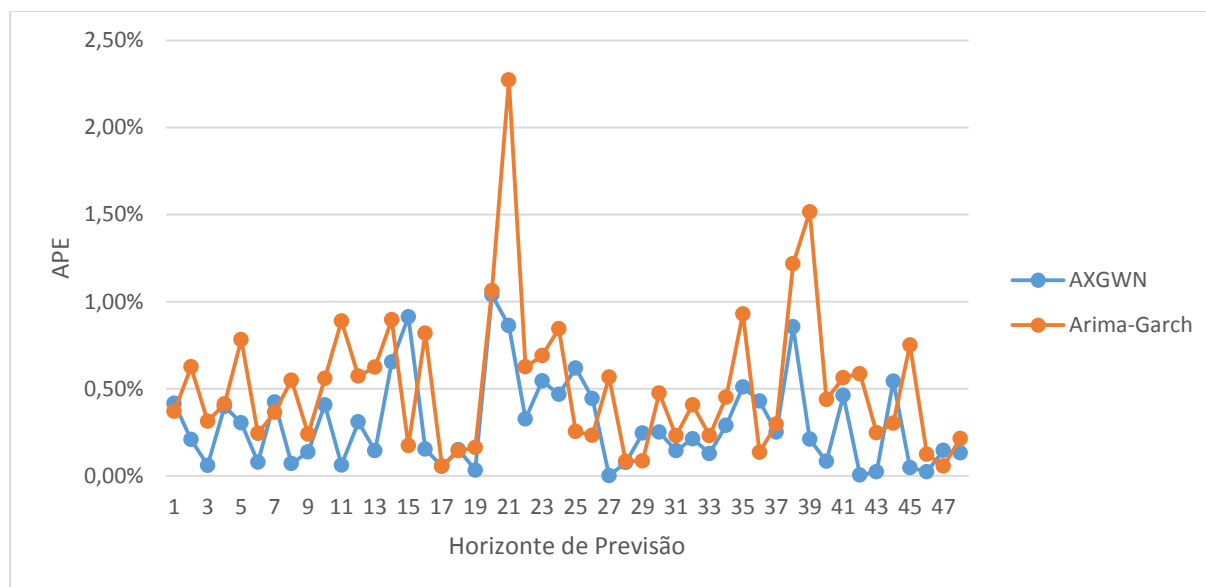


Figura 6 – Comparação da evolução temporal do APE, na amostra de teste, dos métodos Arima-Garch e AXGWN.

5. CONCLUSÕES

Neste artigo, é proposto um novo método preditivo híbrido, designado por *ArimaX-Garch Wavelet Neural*, para a previsão de séries temporais de deslocamento de blocos de barragem que utiliza variáveis exógenas wavelet (obtida a partir da decomposição wavelet da série

subjacente), como se fossem, de fato, variáveis exógenas. Este método foi utilizado para a modelagem da série temporal diária, oriunda das aferições do pêndulo em um dos blocos da barragem principal da Usina de Itaipu, que possibilitou inserir informações (decodificadas no domínio do tempo) da frequência espectral (Souza et al., 2015) presente nos dados na modelagem para a construção das previsões. De fato, as previsões agregaram informações provenientes das estruturas lineares (por meio do modelo Arima), não-lineares (via modelo Garch e RNA-MLP) e da frequência espectral (por meio da utilização das CWs como variáveis exógenas) exibidas pela série temporal subjacente. Com efeito, o método proposto foi mais preciso que o tradicional modelo Arima-Garch.

A de utilização das componentes wavelet se tornou viável uma vez que as de detalhe são sempre um processo estacionário de segunda ordem (o que é requerido para as variáveis exógenas da modelagem ArimaX). Além disso, a de aproximação pode ser modelada por um Arima-Garch sempre que a série de tempo original também for capaz de sê-lo. Ainda, as componentes wavelet, bem como as suas versões defasadas, tendem a apresentar autocorrelação com a série temporal original, uma vez que são obtidas a partir desta.

Por fim, salienta-se também que o método proposto pode ser operacionalizado de uma forma relativamente simples, com a utilização dos software mencionados no texto.

REFERÊNCIAS

BOLLERSLEV, T. *Generalized Autoregressive Conditional Heteroscedasticity. Journal of Econometrics*, v. 31, p. 307–327, 1990.

BOX, G. E. P.; JENKINS, G. M.; REINSEL, G. C. *Time Series Analysis: Forecasting and Control. 4th ed. ed. Hoboken, NJ: Wiley, 2008.*

BOX, G. E. P.; TIAO, G. C. *Intervention Analysis with Applications to Economic and Environmental problems. Journal of American Statistical Association*, v. 70, n. 349, p. 70–79, 1975.

CAMPOS, L. C. D. *Modelo Estocástico Periódico baseado em Redes Neurais.*, 2010. Tese de Doutorado. Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio).

ENGLE, R. F. *Autoregressive Conditional Heteroscedasticity with Estimates of the Variance of United Kingdom Inflation. Econometrica*, v. vol. 50, n. No. 4, p. 987–1007, 1982.

HAMILTON, J. *Time Series Analysis. Princeton University Press, 1994.*

HAYKIN, S. *Redes Neurais: Princípios e Prática. 2º ed. Porto Alegre, Bookman, 2001.*

KIRCHGÄSSNER, G.; WOLTERS, J. *Introduction to Modern Time Series Analysis. Springer- Verlag Berlin Heidelberg, 2007.*

MALLAT, S. *A Wavelet Tour of Signal Processing: The Sparse Way. Third Edit ed. 2008.*

MILLAR, R. B. *Maximum Likelihood Estimation and Inference: With Examples in R, SAS and ADMB. Wiley, 2011.*

SOUZA, A. C. C. DE; RODRIGUES, A. L.; PACHECO, D. A. DE J. *Métodos quantitativos de previsão de demanda aplicados em orçamento de despesas. Revista ESPACIOS*, v. 36, n. 10, 2015.

TEIXEIRA JÚNIOR, L. A.; MENEZES, M. L. DE; CASSIANO, K. M.; PESSANHA, J. F. M.; SOUZA, R. C. *Modelagem Box e Jenkins, decomposição wavelet e programação matemática na geração de previsões. XV Simpósio de Pesquisa Operacional e Logística da Marinha*, 2012.

TEIXEIRA JÚNIOR, L. A.; SOUZA, R. M.; MENEZES, L. M.; et al. *Artificial Neural Network and Wavelet decomposition in the Forecast of Global Horizontal Solar Radiation. Sobrapo*, v. 35, n. 1, p. 1–16, 2015.

ZHANG, G. P. *Time series forecasting using a hybrid ARIMA and neural network model. Neurocomputing*, v. 50, p. 159–175, 2003.