

Data da publicação xxxx 00, 0000, data da versão atual xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.DOI

Geração de Curva de Colina para Turbina Hidrelétrica Usando Rede Neural Artificial

FELIPE L. LAVRADOR¹, CLARIMAR J. COELHO^{1,2} AND DIOGO F. COSTA SILVA³

¹School of Exact and Computer Sciences, Pontifical Catholic University of Goiás, Goiânia, GO 74605-220 Brazil

²Master's School of Production and Systems Engineering (MEPROS), Pontifical Catholic University of Goiás, Goiânia, GO 74605-220 Brazil

³Institute of Informatics, Federal University of Goiás, Goiânia, GO 74690-900 Brazil

Corresponding author: Felipe L. Lavrador (e-mail: feh.liborio@gmail.com).

Os autores agradecem à Energia Sustentável do Brasil pelo apoio na realização deste estudo “Projeto regulamentado pela ANEEL e desenvolvido no âmbito do Programa de P&D da Energia Sustentável do Brasil S.A. (PD-06631-0007/2018).”

• **RESUMO** Dentro de uma usina hidroelétrica, a curva de colina é um gráfico importante para realizar previsões de valores em circunstâncias específicas. Assim, é proposto três modelos de regressão sendo *Feedforward*, Regressão Linear e XGBoost. Os dados utilizados para treinar as redes, foram obtidos através de um modelo em escala reduzida, totalizam 1243 dados. Assim notando que dentre os três modelos, o modelo *Feedforward* demonstrou uma maior capacidade de generalização. Sendo assim uma melhor alternativa dentre as três redes propostas.

• **INDEX TERMS** *Feedforward*, Regressão Linear, XGBoost, curva de colina, Turbinas Hidrelétricas

I. INTRODUÇÃO

A curva de colina é uma ferramenta fundamental para o estudo de desempenho da turbina, gestão e controle de uma usina hidrelétrica. Dixon [1], em seu livro “Fluid Mechanics and Thermodynamics of Turbomachinery” diz que existe uma altura ideal para a maior eficiência de uma turbina hidrelétrica. A curva de colina é um diagrama que proporciona visualizar a relação entre múltiplos parâmetros, como a eficiência, altura de queda, vazão turbinada, potência gerada, ângulo da pá, e outros. A curva de colina é construída por medições feitas manualmente, e experimentalmente, em um modelo reduzido da turbina. Os dados obtidos ainda sofrem normalizações e interpolações para torná-los mais representativos possíveis. Caso a turbina sofra alguma alteração, os dados da curva apresentarão erros, e a construção de um novo modelo reduzido ou medições na planta real é extremamente oneroso.

Para a redução deste custo, as redes neurais artificiais (ANN's) podem ser utilizadas para a previsão dos parâmetros da turbina e geração de uma curva de colina baseada nos pontos de medição do modelo reduzido. Diversos trabalhos propõem o uso de ANN's para soluções em reconhecimento de padrões e previsões aplicados à problemas de geração de energia elétrica. “Shaw, et al.” [5] exemplifica e demonstra em “Hydropower Optimization Using Artificial Neural Network Surrogate Models of a High Fidelity Hydrodynamics

and Water Quality Model” um método baseado em redes neurais para determinar as operações ideais do reservatório multiuso, em Nashville cidade de Tennessee nos EUA, que segundo suas palavras “O modelo reproduziu com sucesso informações de reservatórios de alta fidelidade, permitindo aumentos de 6,8% e 6,6% no valor da produção de energia hidrelétrica em relação às operações reais para limites de oxigênio dissolvido”. “Old Hickory. Hammid, et al.” [6] e colaboradores apresentam um método baseado em ANN que, após vários testes, em uma pequena usina hidrelétrica no lago Himreen localizada em Diyala previu o desempenho da planta com um coeficiente de correlação, além de modelar o desempenho da planta. Já “Iraq. Bouzic e Jovanovic” [7] pela interpolação espacial de pontos medidos apresentam um método baseado em ANN para previsão das características de turbinas considerando um conjunto de dados do modelo reduzido da turbina.

Neste contexto, foi demonstrado por Galvão Filho [4] a possibilidade de gerar a curva de colina a partir de um modelo de regressão. Assim será proposto outros modelos de regressão com intuito de minimizar os custos computacionais para se gerar a própria curva de colina.

II. CASO DE ESTUDO

Os dados foram cedidos pela Usina Hidrelétrica Jirau, instalada no rio Madeira, no estado de Rondônia - Brasil,

é composta por cinquenta unidades geradoras de bulbo horizontal tipo *Kaplan*. Dentre as 50, 28 estão localizadas na margem direita e 22 na margem esquerda. Assim, a partir das medições de um modelo em escala reduzida, foi retirado os dados que totalizavam 1243.

De acordo com Galvão Filho [4], a curva de colina é um gráfico que expressa a complexa dependência da eficiência com vários outros parâmetros. Podendo ter uma demonstração simplificada sendo:

$$\eta = F(h, \rho, \mu, \alpha, \beta) \quad (1)$$

A eficiência (η) é obtida pela relação entre a queda d'água (h), vazão (ρ) e os parâmetros de saída da turbina sendo, potência da turbina (μ), ângulo da porta *wicket* (α) e ângulo da pá do rotor (β).

III. LINHA DE BASE

Os três modelos selecionados para comparação e avaliação são Regressão Linear, Regressão XGBoost e *Feedforward Neural Network*. Utilizando de artifícios como *grid search* para a definição de hiperparâmetros [9] e *k-fold* para separação de variáveis entre treino e teste [10]. Por fim foi aplicado uma normalização na amplitude de [0,1].

A. REGRESSÃO LINEAR

Como dito por H. Roopa e T. Asha em [2], a regressão prevê o valor y com base em um conjunto de variáveis independentes de entrada. Como a regressão linear expresso pela equação:

$$y = b_0 + b_1x_0 + b_2x_2 + b_3x_3 + \dots + b_nx_n + \epsilon \quad (2)$$

Cada regressão consegue prever uma única variável, como o modelo necessita de quatro saídas, se aplica uma regressão por variável de saída, permitindo um modelo de rede predizer todos os resultados. Assim, ao se notar a equação, a mesma é ajustada de acordo com cada ponto, portanto não apresentando hiperparâmetros apenas a própria equação.

B. REGRESSÃO XGBOOST

Como explicado por [8], regressão XGBoost é um algoritmo de árvore com aumento de gradiente, permitindo através de uma árvore de decisões convergir até o melhor valor de uma dada entrada. Assim para tal modelo, utilizou-se os hiperparâmetros *learning-rate*, *n-estimators*, *max-depth*, *min-child-weight*, *seed*, *subsample*, *colsample-bytree*, *gamma*, *reg-alpha* e *reg-lambda* com os respectivos valores 0.116, 1000, 15, 0, 0, 0.8, 0.8, 0, 0 e 1.

C. FEEDFORWARD NEURAL NETWORK

Feedforward Neural Network (FFNN) são modelos inspirados no cérebro humano com capacidade de generalização [3]. Para tal problemática, foi estabelecido uma rede com dois *perceptrons* [11] de entrada sendo respectivamente h e ρ . Contendo duas camadas ocultas sendo a primeira com 100 *perceptrons* e a segunda com 80. Já a camada de saída

apresenta 4 *perceptrons* para determinar os valores de η , μ , α e β . Vale salientar que para as camadas de entrada e oculta, foi utilizado a tangente hiperbólica como função de ativação, enquanto a camada de saída usou sigmoide. Assim, os hiperparâmetros podem ser definidos, utilizando de valores como 0.01192 para *learning-rate* e 1000 para as épocas. Por fim, para melhorar a capacidade de generalização de tal algoritmo se aplica uma função de otimização chamada Adam [12], assim permitindo a rede se adequar de modo devido à problemática.

D. ERROS

Para avaliar o critério preditivo destes modelos, foi aplicado três critérios, sendo a raiz quadrada do erro-médio (RMSE)

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (\hat{y}_i - y_i)^2} \quad (3)$$

erro médio absoluto (MAE)

$$MAE = \frac{1}{N} \sum_{i=1}^N |\hat{y}_i - y_i| \quad (4)$$

e coeficiente de determinação (R^2)

$$R^2 = \frac{\sum_{i=1}^N (y_i - \hat{y}_i)}{\sum_{i=1}^N (y_i - \bar{y}_i)} \quad (5)$$

sendo N a quantidade de dados, \hat{y}_i o valor predito pela rede, y_i valor mensurado, \bar{y}_i a média dos valores mensurados.

IV. RESULTADOS

Com as métricas de erro estabelecido é possível determinar para cada rede, os erros avaliando entre elas qual apresentou um menor erro e uma melhor capacidade de generalização. Assim para os modelos propostos, se obteve os seguinte resultados:

A. REGRESSÃO LINEAR

TABLE 1. Erros

Erros treino	η	μ	α	β
RMSE	0.0949	0.0505	0.0429	0.0166
MAE	0.0716	0.0377	0.0354	0.0135
R^2	0.3139	0.9457	0.9656	0.9969
Erros teste	η	μ	α	β
RMSE	0.0979	0.0489	0.0427	0.0172
MAE	0.0692	0.0375	0.0339	0.0134
R^2	0.2453	0.9470	0.9655	0.9966

Erros demonstrados na figura 1 e 2, para treino e teste respectivamente.

B. REGRESSÃO XGBOOST

Erros demonstrados na figura 3 e 4, para treino e teste respectivamente.

TABLE 2. Erros

Erros treino	η	μ	α	β
RMSE	0.0087	0.0076	0.0032	0.0015
MAE	0.0025	0.0023	0.0014	0.0008
R^2	0.9966	0.9988	0.9998	0.9999
Erros teste	η	μ	α	β
RMSE	0.0438	0.0506	0.0221	0.0079
MAE	0.0246	0.0324	0.0154	0.0060
R^2	0.9177	0.9416	0.9905	0.9993

C. FEEDFORWARD NEURAL NETWORK

TABLE 3. Erros

Erros treino	η	μ	α	β
RMSE	0.0438	0.0506	0.0221	0.0079
MAE	0.0246	0.0324	0.0154	0.0060
R^2	0.9177	0.9416	0.9905	0.9993
Erros teste	η	μ	α	β
RMSE	0.0074	0.0061	0.0075	0.0072
MAE	0.0055	0.0046	0.0055	0.0052
R^2	0.9978	0.9992	0.9989	0.9994

Erros demonstrados na figura 5 e 6, para treino e teste respectivamente.

V. DISCUSSÃO

Todos os modelos de modelos apresentados, se mostraram promissores, contudo o modelo *Feedforward* teve uma capacidade de generalização superior ao modelo XGBoost, contudo o XGBoost ainda se mostrou superior ao modelo de regressão linear. Vale ressaltar também que o modelo de regressão linear apresentou um erro alto em eficiência, demonstrando uma incapacidade de generalizar significativamente os dados, devido à maneira de dispersão dos mesmos.

VI. CONCLUSÃO

Como citado anteriormente, o modelo *Feedforward* dentre todos os modelos propostos, apresentou uma melhor capacidade de generalização, gerando a curva de colina demon-

strada na imagem 6. Outro modelo que se mostra bastante promissor em questão de geração de dados, é o modelo XGBoost, visto que todos seus erros foram próximos ao modelo *Feedforward*.

REFERENCES

- [1] Dixon, Sydney Lawrence, and Cesare Hall. Fluid mechanics and thermodynamics of turbomachinery. Butterworth-Heinemann, 2013.
- [2] H. Roopa and T. Asha, "A linear model based on principal component analysis for disease prediction," IEEE Access, vol. 7, pp. 105314-105318, 2019.
- [3] M. Hassoun, "Fundamentals of Artificial Neural Networks", A Bradford Book, 2003.
- [4] Galvão Filho, Arlindo R., et al. "Generation of Two Turbine Hill Chart Using Artificial Neural Networks." 2020 IEEE 10th International Conference on Intelligent Systems (IS). IEEE, 2020.
- [5] A. R. Shaw and H. S. Sawyer and E. J. LeBoeuf and B. Hadjerioua, "Hydropower Optimization Using Artificial Neural Network Surrogate Models of a High-Fidelity Hydrodynamics and Water Quality Model", Water Resources Research, vol. 53, nr. 11, pp. 9444-9461, 2017.
- [6] A. T. Hammid and M. H. B. Sulaiman and A. N. Abdalla, "Prediction of small hydropower plant power production in Himreen Lake dam (HLD) using artificial neural network", Alexandria Eng. J., <http://dx.doi.org/10.1016/j.aej.2016.120.011>, 2018.
- [7] I. Bozic and R. Jovanović "Prediction of Double-Regulated Hydraulic Turbine On-Cam Energy Characteristics by Artificial Neural Networks Approach", FME Transactions, vol. 44, pp. 125-132, DOI: 10.5937/fmet1602125B, 2016.
- [8] Nguyen, Lien Thi Kim, et al. "Using XGBoost and skip-gram model to predict online review popularity." SAGE Open 10.4 (2020): 2158244020983316.
- [9] Alibrahim, Hussain, and Simone A. Ludwig. "Hyperparameter optimization: Comparing genetic algorithm against grid search and bayesian optimization." 2021 IEEE Congress on Evolutionary Computation (CEC). IEEE, 2021.
- [10] Anguita, Davide, et al. "The 'K' in K-fold cross validation." 20th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN). i6doc. com publ, 2012.
- [11] Minsky, Marvin, and Seymour A. Papert. Perceptrons: An introduction to computational geometry. MIT press, 2017.
- [12] Zhang, Zijun. "Improved adam optimizer for deep neural networks." 2018 IEEE/ACM 26th International Symposium on Quality of Service (IWQoS). IEEE, 2018.

...

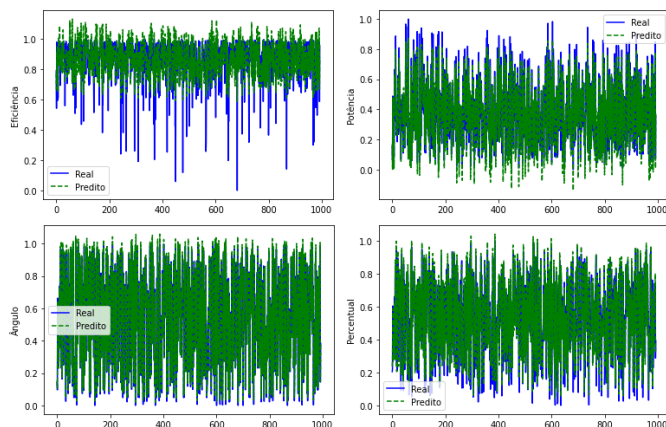


FIGURE 1. Amostragem dados treino

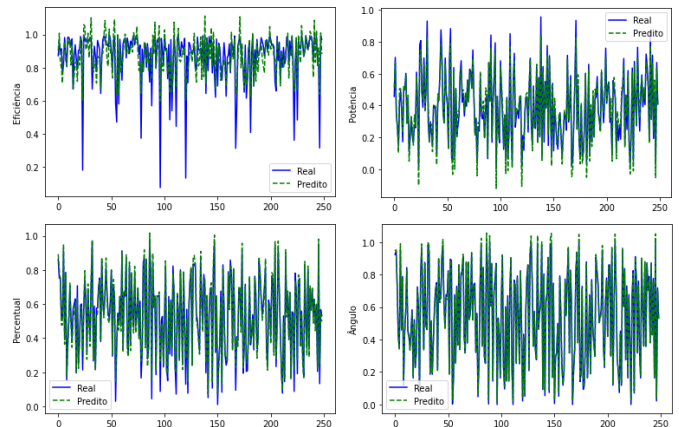


FIGURE 2. Amostragem dados teste

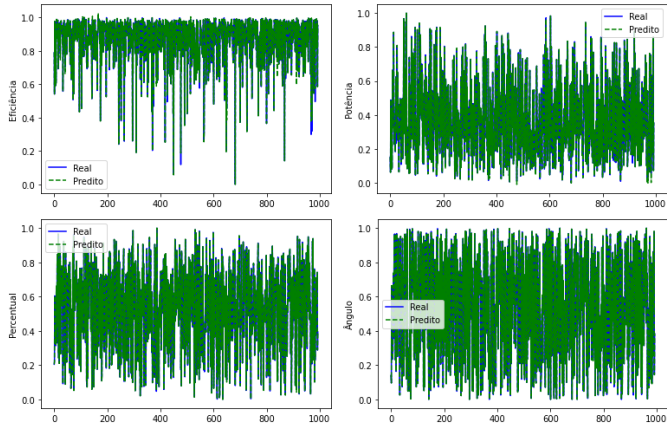


FIGURE 3. Amostragem dados treino

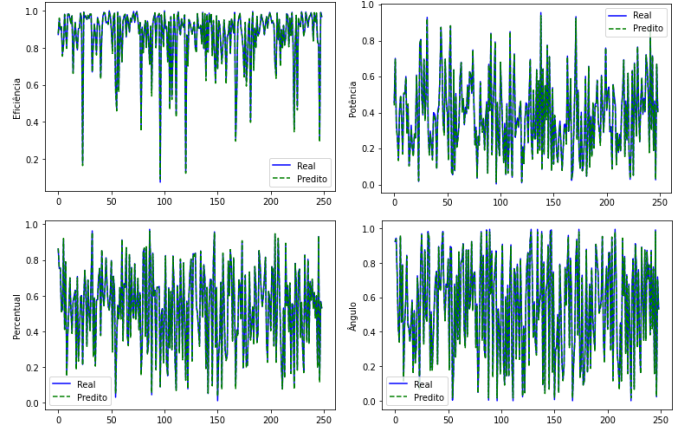


FIGURE 6. Amostragem dados teste

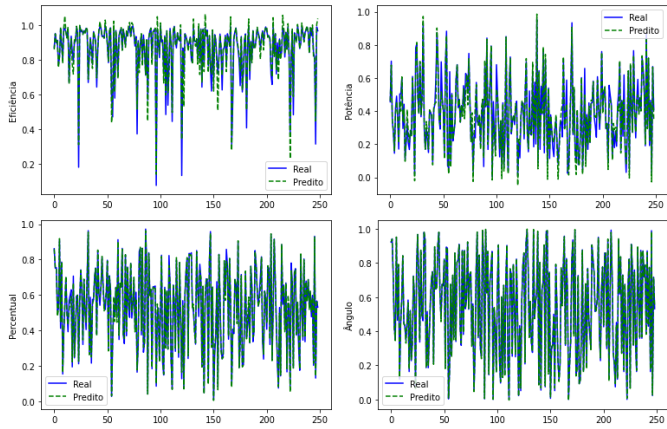


FIGURE 4. Amostragem dados teste

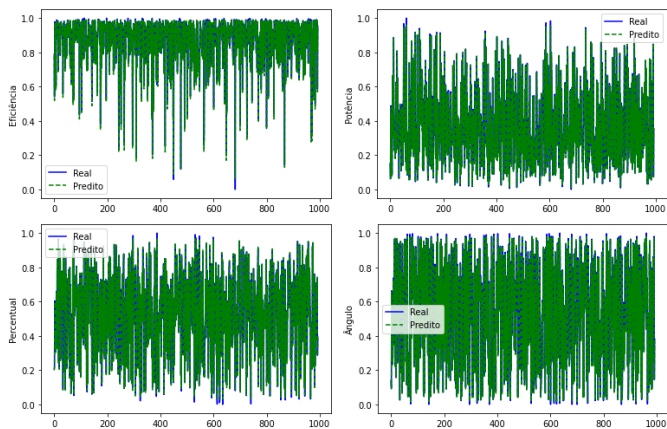


FIGURE 5. Amostragem dados treino

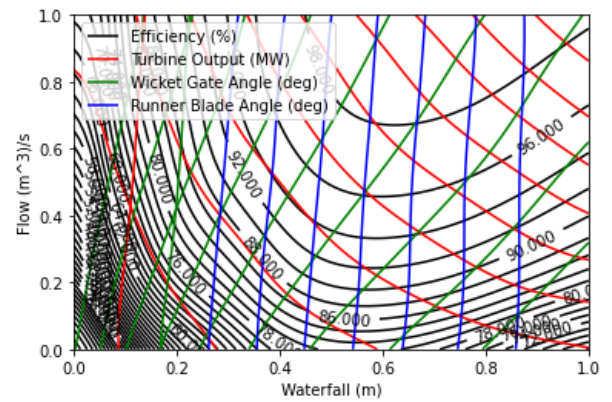
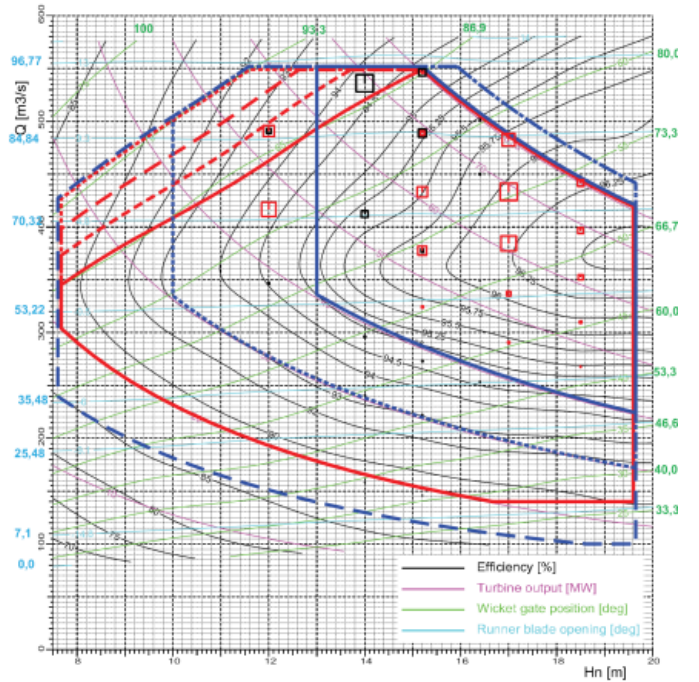


FIGURE 7. Amostragem dados teste