

Aplicação das Redes Neurais MLP, ELM e ESN para a Estimação da Curva de Potência de Aerogeradores

Denilson Alves de Sousa

Programa de Pós-Graduação em Engenharia Elétrica, Curso de
Engenharia Elétrica
Universidade Federal do Piauí-UFPI
Teresina, Piauí.
eng.denilalvessousa@gmail.com

Luís G. M. Souza

Programa de Pós-Graduação em Engenharia Elétrica, Curso de
Engenharia Elétrica
Universidade Federal do Piauí-UFPI
Teresina, Piauí.
lmosouza@ufpi.edu.br

Abstract— Wind turbine power forecasting is essential for the efficient and safe operation of wind power generation systems. A fundamental aspect of this process is the accurate determination of the power curve, which relates wind speed to the energy generated. This curve directly influences production estimation and energy planning. Models based on artificial intelligence, such as MLP (Multilayer Perceptron), ELM (Extreme Learning Machine) and ESN (Echo State Network), have proven effective in capturing the complex and non-linear relationships between the variables involved. Therefore, choosing a good method to forecast the power curve is not only a technical issue, but also a strategic one. A well-adjusted model allows for the reduction of uncertainties, the optimization of wind energy integration into the electricity grid and the contribution to more effective operational decisions.

Resumo: A previsão da potência de aerogeradores é essencial para a operação eficiente e segura dos sistemas de geração eólica. Um aspecto fundamental nesse processo é a determinação precisa da curva de potência, que relaciona a velocidade do vento à energia gerada. Essa curva influencia diretamente a estimativa de produção e o planejamento energético. Modelos baseados em inteligência artificial, como MLP (Multilayer Perceptron), ELM (Extreme Learning Machine) e ESN (Echo State Network), têm se mostrado eficazes em capturar as complexas e não lineares relações entre as variáveis envolvidas. Dessa forma, a escolha de um bom método para prever a curva de potência não é apenas uma questão técnica, mas estratégica. Um modelo bem ajustado permite reduzir incertezas, otimizar a integração da energia eólica na matriz elétrica e contribuir para decisões operacionais mais eficazes.

Keywords— Artificial Neural Networks; Power Curve Estimation; Wind Turbines.

Palavras-chaves: Redes Neurais Artificiais; Estimação da Curva de Potência;

1. INTRODUÇÃO

O uso de fontes renováveis de energia, especialmente a eólica, tem crescido substancialmente nas últimas décadas. A eficiência de parques eólicos depende, entre outros fatores, da capacidade de prever com precisão a energia gerada por cada aerogerador. Essa previsão de energia gerada por aerogeradores é essencial para o planejamento energético eficiente. Tradicionalmente, utiliza-se a curva de potência baseada na relação entre velocidade do vento e potência gerada. Entretanto, métodos tradicionais não consideram as dependências temporais e sazonais dos dados. Este trabalho explora o uso de redes neurais para capturar essas dinâmicas temporais e comparar os resultados com o modelo MLP, ELM e ESN.

A escolha das arquiteturas MLP, ELM e ESN baseia-se em suas características complementares frente à tarefa de modelagem de sistemas não lineares e dependentes do tempo. A MLP é uma arquitetura consagrada e amplamente utilizada em problemas de regressão, mas com custo computacional elevado. A ELM, por sua vez, é uma evolução estrutural da MLP, com treinamento extremamente rápido e boa capacidade de generalização. A ESN, com arquitetura recorrente e estrutura de reservatório, é especialmente indicada para capturar dependências temporais — como aquelas presentes na variação do vento. Essa diversidade de abordagens permite analisar não apenas o desempenho em termos de erro, mas também aspectos de robustez, escalabilidade e aplicabilidade em tempo real.

A curva de potência relaciona velocidade do vento à energia gerada, sendo um parâmetro essencial para o monitoramento e controle de parques eólicos. Modelos paramétricos (como funções logísticas de múltiplos parâmetros) e modelos de aprendizado não paramétrico (como MLPs, SVR, GPs) são amplamente utilizados. Entretanto, muitos desses modelos não lidam bem com ruídos e ignoram dados não rotulados. Nesse contexto, algoritmos baseados em ELMs com aprimoramentos robustos têm se destacado por sua simplicidade, velocidade e capacidade de generalização.

Nosso objetivo é comparar a precisão das previsões fornecidas por essas arquiteturas, analisando sua capacidade de generalização e robustez diante de diferentes conjuntos de dados. Os resultados obtidos podem contribuir para aprimorar a modelagem do desempenho de aerogeradores e auxiliar na otimização da geração eólica.

2. ESTIMAÇÃO DA CURVA DE POTÊNCIA

2.1. Curva de Potência do Aerogerador

A curva de potência de um aerogerador é uma ferramenta fundamental para avaliar seu desempenho, estimar a produção de energia e planejar operações de manutenção. Essa curva descreve a relação entre a velocidade do vento e a potência gerada, sendo tradicionalmente obtida a partir de medições empíricas ou modelos físicos (IEC 61400-12-1, 2017). No entanto, devido à variabilidade das condições ambientais e operacionais, abordagens baseadas em aprendizado de máquina têm se mostrado alternativas promissoras para a estimação mais precisa da curva de potência (Zhang et al., 2018).

A curva de potência representa a relação entre a velocidade do vento (v) e a potência gerada (P), podendo ser expressa de forma ideal por:

$$P = 0.5 \times \rho \times A \times C_p \times v^3 \quad (1)$$

onde:

ρ = densidade do ar,

A = área varrida das pás,

C_p = coeficiente de performance,

v = velocidade do vento.

A curva de potência de um aerogerador apresenta três regiões operacionais distintas em função da velocidade do vento. Na primeira região, denominada região de corte inferior, a velocidade do vento está abaixo da velocidade mínima necessária para iniciar a conversão de energia, conhecida como velocidade de partida ou velocidade de corte inferior. Nessa condição, o torque aerodinâmico gerado não é suficiente para vencer as perdas mecânicas e elétricas do sistema, resultando em potência de saída nula.

À medida que a velocidade do vento aumenta e ultrapassa esse valor mínimo, o aerogerador entra na segunda região, compreendida entre a velocidade de partida e a velocidade nominal. Nesta faixa, o controle do sistema é orientado para maximizar a extração de energia do vento, por meio de estratégias como o controle de pitch (ajuste do ângulo das pás) ou a variação da razão de velocidade entre o rotor e o gerador. A potência gerada nessa fase cresce rapidamente, geralmente de forma cúbica em função da velocidade do vento, até atingir a potência nominal do equipamento.

A terceira região, denominada região de potência nominal ou de saturação, ocorre quando a velocidade do vento está entre a velocidade nominal e a velocidade de corte superior. Nesse intervalo, a potência de saída do aerogerador é mantida constante, limitada à sua capacidade máxima, com o objetivo de evitar sobrecarga e preservar a integridade estrutural do sistema. O controle do ângulo das pás torna-se fundamental para manter essa limitação. Caso a velocidade do vento ultrapasse a velocidade de corte superior, o aerogerador é automaticamente desligado (*shutdown*) como medida de proteção contra esforços mecânicos excessivos, vibrações e riscos de falha estrutural.

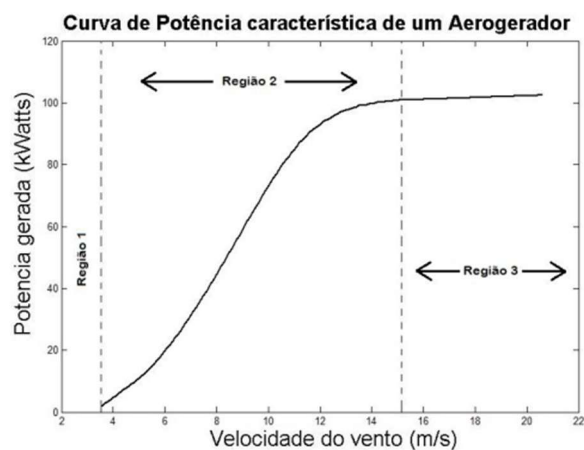


Fig. 1: Curva característica de um aerogerador.

A curva de potência dos aerogeradores é um dos parâmetros mais relevantes tanto para fabricantes quanto para comerciantes, pois ela representa a relação entre a velocidade do vento incidente e a potência elétrica entregue na saída do gerador, sob condições padronizadas. Esse perfil de desempenho é essencial para avaliar a eficiência do equipamento, prever a produção anual de energia (AEP – *Annual Energy Production*) e estabelecer critérios de viabilidade econômica para projetos eólicos.

3. MODELOS APLICADOS

3.1. Perceptron de Múltiplas Camadas (MLP)

A rede MLP (*Multilayer Perceptron*) é caracterizada por ter pelo menos uma camada neural intermediária e uma camada neural de saída. Assim, apresentando pelo menos duas camadas de neurônios, uma intermediária localizada entre a entrada e a camada de saída e, uma camada de saída. É utilizada em aplicações nas mais diferentes áreas do conhecimento e é considerada uma das arquiteturas mais flexíveis quanto a sua aplicabilidade.

A modificação dos coeficientes e dos elementos não lineares (que podem empregar funções sigmóides ou lineares) é realizada por meio de aprendizado supervisionado, ou seja, para cada instância dos dados de entrada, é conhecida a resposta esperada.

O aprendizado da MLP ocorre através de um algoritmo conhecido como retropropagação do erro (*backpropagation*), combinado com um método de otimização (como o gradiente descendente). Esse processo ajusta iterativamente os pesos da rede para minimizar a diferença entre a saída da rede e a saída desejada (o erro). Para isso, o erro é propagado de trás para frente, a partir da camada de saída em direção às camadas anteriores, atualizando os pesos com base na contribuição de cada neurônio ao erro total.

O processo de treinamento das redes MLP, neste trabalho, foi utilizado o algoritmo *backpropagation*, também é conhecido como regra delta generalizada (SILVA; SPATTI; FLAUZINO, 2010).

Na primeira etapa de treinamento da rede MLP, denominada de propagação adiante, os sinais de entrada $[X_1, X_2, \dots, X_p]$ de uma amostra do conjunto de dados de treinamento são inseridos na entrada da rede e propagados até atingir a camada oculta da rede, onde é calculado o vetor das

saídas dos neurônios da camada oculta (Equação 2) e o vetor das saídas dos neurônios da camada de saída (Equação 3). Em seguida é calculado o erro de estimação, definido pela Equação 4.

$$y_1(n) = f(\sum_{i=1}^p W_{ji}^{(1)} \cdot X_i(n)) \quad (2)$$

onde $y_1(n)$ é o vetor de saídas dos neurônios da camada oculta para o n -ésimo padrão de entrada da rede, $W_{ji}^{(1)}$ é o peso da conexão sináptica do j -ésimo neurônio da camada oculta com o i -ésimo sinal de entrada da rede e $f(\cdot)$ é uma função de ativação do tipo sigmoidal.

$$y_2(n) = g(\sum_{j=1}^q y_1(n) \cdot W_{1j}^{(2)}) \quad (3)$$

onde $y_2(n)$ é o vetor de saídas dos neurônios da camada oculta para o n -ésimo padrão de entrada da rede, $W_{1j}^{(2)}$ é o peso da conexão sináptica do j -ésimo neurônio da camada oculta com o neurônio de saída da rede e $g(\cdot)$ é uma função de ativação do tipo linear.

$$E(n) = y_2(n) - d(n) \quad (4)$$

onde $E(n)$ é o erro de estimação e $d(n)$ é a saída desejada para o n -ésimo padrão de entrada da rede.

Após realizar a etapa de propagação adiante, na etapa de propagação reversa, são ajustados os pesos das conexões sinápticas da camada de saída e camada oculta. Inicialmente, são ajustados os pesos das conexões sinápticas localizadas entre a camada de saída e camada oculta, conforme Equação 5.

$$W_{1j}^{(2)}(t+1) = W_{1j}^{(2)}(t) + \eta \cdot \delta_1^{(2)} \cdot y_1(n) \quad (5)$$

onde: $\delta_1^{(2)}$, calculado pela Equação 6, é o gradiente local em relação ao neurônio da camada de saída, t corresponde a época atual de treinamento e η é definido como a taxa de aprendizagem do algoritmo *backpropagation*.

$$\delta_1^{(2)} = -E(n) \cdot g'(y_2(n)). \quad (6)$$

Por fim, na etapa de propagação reversa, é feito o ajuste dos pesos das conexões sinápticas localizadas entre a camada oculta e a camada de entrada, conforme Equação 7.

$$W_{1j}^{(1)}(t+1) = W_{1j}^{(1)}(t) + \eta \cdot \delta_1^{(1)} \cdot X_i \quad (7)$$

onde: $\delta_1^{(1)}$, calculado pela Equação 8, é o gradiente local em relação ao j -ésimo neurônio da camada oculta.

$$\delta_1^{(1)} = \delta_1^{(2)} \cdot W_{1j}^{(2)} \cdot f'(y_1) \quad (8)$$

É fundamental destacar que as etapas de propagação direta (*forward*) e retropropagação (*backward*) são realizadas para todos os n padrões de entrada até que algum critério de parada da rede seja alcançado — seja ele o número máximo de épocas de treinamento ou um erro mínimo previamente definido.

Em síntese, essas etapas têm como objetivo ajustar, a cada iteração do algoritmo de treinamento, os pesos sinápticos e os limiares de ativação dos neurônios, de forma que o erro total na camada de saída seja progressivamente reduzido.

3.2. Máquina de Aprendizagem Extremo (ELM)

A Máquina de Aprendizagem Extrema (ELM – *Extreme Learning Machine*) é uma rede neural artificial do tipo *feedforward* (sem realimentação), composta por uma única camada oculta de neurônios e uma camada de saída. Proposta por Huang et al. (2004), a ELM destaca-se por oferecer um desempenho comparável ao das redes MLP (*Multilayer Perceptron*), com a vantagem significativa de apresentar uma fase de treinamento muito mais rápida.

A rede ELM foi desenvolvida a partir da arquitetura MLP, com a principal diferença de que os pesos das conexões entre a camada de entrada e a camada oculta são definidos aleatoriamente e permanecem fixos, ou seja, não são ajustados durante o processo de aprendizagem. Os neurônios da camada oculta utilizam funções de ativação que devem ser completamente diferenciáveis em todo o domínio, sendo comum o uso de funções não lineares, como a logística ou a tangente hiperbólica. No processo de treinamento da ELM, apenas os pesos entre a camada oculta e a camada de saída são efetivamente ajustados.

A aprendizagem da rede ELM é realizada em três etapas: inicialização aleatória dos pesos das conexões sinápticas da rede, cálculo das saídas dos neurônios da camada escondida e cálculo da matriz de pesos sinápticos da camada de saída.

A matriz de pesos das conexões sinápticas entre a camada de camada e a camada oculta W_{ji} tem dimensão $(p+1) \times q$, sendo $p+1$ o número de linha e q o número de colunas, definida pela Equação 9.

$$W_{ji} = \begin{bmatrix} w_{11} & w_{12} & \dots & w_{1q} \\ w_{21} & w_{22} & \dots & w_{2q} \\ \vdots & \vdots & \ddots & \vdots \\ w_{p1} & w_{p2} & \dots & w_{pq} \\ b_1 & b_2 & \dots & b_q \end{bmatrix}_{(p+1) \times q} \quad (9)$$

onde $[b_1, b_2, \dots, b_q]$ é o vetor de bias, sendo a matriz de pesos W_{ji} inicializada com valores aleatórios segundo uma distribuição normal, matematicamente, definida por:

$$W_{ji} \sim N(\mu, \sigma^2), \quad (10)$$

sendo $N(\mu, \sigma^2)$ um número aleatório distribuído uniformemente no intervalo $[-1,1]$, caracterizado por apresentar média $\mu = 0$ e variância σ^2 .

No cálculo das saídas dos neurônios da camada escondida da ELM, inicialmente, é necessário que os sinais de entrada X (contendo p elementos) sejam propagados à camada intermediária para que as ativações dos neurônios dessa camada sejam calculadas (Equação 11)

$$u_j(n) = \sum_{i=1}^p W_{ji} \cdot X_i(n) - \theta_j(n) \quad (11)$$

Onde W_{ji} é o peso da conexão sináptica que interliga a i -ésima unidade de entrada ao j -ésimo neurônio oculto e $\theta_j(n)$ é o limiar de ativação do j -ésimo neurônio da camada oculta da rede ELM. Em seguida, são calculadas as saídas dos neurônios da camada escondida, conforme a Equação 12.

$$S_j(n) = f(\sum_{i=1}^p W_{ji} \cdot X_i(n) - \theta_j(n)) \quad (12)$$

onde $f(\cdot)$ é uma função de ativação não linear, geralmente sigmoidal, que é aplicada a todos os neurônios da camada oculta.

Para finalizar a etapa de aprendizagem da rede ELM é necessário calcular matriz de pesos sinápticos da camada de saída. Por se tratar de um aprendizado do tipo supervisionado, para cada vetor de entrada $X(n), n = 1, \dots, T$, tem-se uma saída desejada $y(n)$ correspondente. Assim, a matriz de saídas desejadas tem dimensão dada por T linhas e 1 coluna (apenas uma saída)

$$y(n) = \beta^T S(n) \quad (12)$$

onde β corresponde a matriz de pesos sinápticos do neurônio da camada de saída. Sendo β calculada por meio do método dos mínimos quadrados, também conhecido como método da pseudo-inversa (Equação 13)

$$\beta = [SS^T]^{-1} S^T y \quad (13)$$

3.3. Echo State Network

Semelhante à ELM, a Rede de Estados de Eco (ESN – Echo State Network) possui apenas uma camada oculta, cujos pesos das conexões entre a entrada e essa camada permanecem fixos durante todo o processo de aprendizagem. Assim como na ELM, a ESN é considerada uma rede de projeções aleatórias, exigindo, em geral, um número elevado de neurônios na camada oculta. Isso se deve à necessidade de compensar a aleatoriedade dos pesos fixos, que são definidos na inicialização e não sofrem ajustes durante o treinamento (JAEGER, 2001).

A ESN é uma rede neural com arquitetura recorrente, na qual as saídas dos neurônios podem ser realimentadas como entradas para outros neurônios da rede. Sua principal característica é a presença de um componente interno chamado reservatório, uma estrutura dinâmica composta por neurônios com conexões esparsas e pesos inicializados aleatoriamente. As funções de ativação utilizadas nos neurônios do reservatório são, em geral, as mesmas empregadas em redes neurais convencionais, como as funções sigmoidais.

A estrutura de uma rede ESN aplicada a identificação de sistemas, com uma camada de entrada contendo p unidades de entrada, uma camada intermediária chamada de reservatório de dinâmicas com q neurônios e uma camada de saída com um neurônio. Considerando um vetor de sinais de entradas representado por $X(n) = [X_1(n), \dots, X_p(n)]^T \in \mathbb{R}^p$, os pesos das conexões entre os sinais de entrada e os neurônios do reservatório serão definidos por W_{in} com dimensão $q \times p$.

Os neurônios pertencentes ao reservatório estão conectados. Essas ligações sinápticas serão denotadas pela matriz de pesos W com dimensão $q \times q$, de maneira a permitir ligações entre todos os neurônios do reservatório (o que nem sempre ocorre). As ativações dos neurônios do reservatório são denotadas por $r(n) = [r_1(n), \dots, r_N(n)]^T \in \mathbb{R}^q$, sendo q a quantidade de neurônios do reservatório. A atualização das ativações dos neurônios do reservatório é feita pela Equação 14.

$$r(n) = f(W_{in} \cdot X(n) + W(n-1) \cdot r(n-1) + W_{back} y(n-1)) \quad (14)$$

onde $f(\cdot)$ representa a função de ativação dos neurônios do reservatório, normalmente do tipo sigmoidal.

A matriz de pesos das conexões sinápticas entre o neurônio da camada de saída e os neurônios do reservatório são denotadas por W_{black} , com dimensão $q \times 1$. A saída do neurônio da camada de saída da rede é expressa por $\hat{y}(n) = [\hat{y}_1(n), \dots, \hat{y}_N(n)]^T \in \mathbb{R}$, sendo determinada conforme a Equação 15.

$$\hat{y}(n) = g \cdot (W_{out}(n)) \quad (15)$$

onde $g(\cdot)$ corresponde à função de ativação dos neurônios da camada de saída (normalmente sendo do tipo linear) e $z(n) = [X(n); r(n); \hat{y}_N(n-1)] \in \mathbb{R}^{p+q+1}$ representa a matriz resultante da concatenação da matriz de entradas atuais, da matriz de ativações atuais dos neurônios do reservatório e a matriz saída do neurônio da camada de saída no instante anterior. Já a matriz W_{out} representa os pesos sinápticos da camada de saída com dimensão dada por $1 \times (p+q+1)$.

Etapa de aprendizagem da ESN consiste em ajustar a matriz de pesos das conexões sinápticas dos neurônios da camada de saída (W_{out}) da rede. A rede ESN implementada neste trabalho possui apenas 1 (um) neurônio na camada de saída. Portanto, semelhante ao que é feito com a camada de saída da rede ELM, na rede ESN, a matriz de pesos W_{out} pode ser calculada pelo método de estimação dos mínimos quadrados, também conhecido como método da pseudo-inversa, conforme mostrado na Equação 16.

$$W_{out} = [MM^T]^{-1} M^T D \quad (16)$$

onde M é uma matriz auxiliar definida pela Equação 4.16, composta pela concatenação das matrizes de estados r , X e y , sem o período transitório da rede. A variável D é uma matriz auxiliar que armazena as saídas desejadas da rede, com exclusão do período transitório da rede ESN.

$$M = [r, X, y] \quad (17)$$

4. RESULTADOS E DISCUSSÃO

O banco de dados utilizado disponibiliza informações sobre velocidade do vento e potência gerada por seis turbinas eólicas, sendo quatro localizadas em parques eólicos *offshore* (instalados no mar) e duas em parques *onshore* (instalados em terra), identificadas de WT1 a WT6. Neste estudo, no entanto, são utilizados exclusivamente os dados da turbina WT1.

Os dados da turbina WT1 utilizados neste estudo referem-se à velocidade do vento, coletada por sensores instalados em uma torre meteorológica, e à potência, medida diretamente na própria turbina. Cada torre meteorológica está associada a duas turbinas eólicas, ou seja, cada medição de velocidade do vento é vinculada às potências geradas por essas duas turbinas correspondentes.

Antes do treinamento das redes, os dados passaram por um processo de normalização Min-Max, com o objetivo de reduzir disparidades nas escalas entre a velocidade do vento e a potência gerada. Foram ainda removidos registros

inconsistentes e *outliers* principalmente relacionados a falhas de medição. Além disso, optou-se por manter o intervalo de 10 minutos entre os registros, considerado suficiente para preservar as variações características da geração eólica, sem induzir ruído temporal excessivo.

Para a turbina WT1, o conjunto de dados é composto por $N = 47.542$ registros. De forma geral, os dados coletados em parques eólicos são agrupados em blocos de 10 minutos, intervalo considerado adequado devido à relativa estabilidade da velocidade do vento nesse período. Como a geração de potência pelos aerogeradores depende diretamente da velocidade do vento, ambas as variáveis são registradas com a mesma frequência.

Os modelos desenvolvidos neste estudo foram implementados por meio de programação no software Matlab®. Para uma análise comparativa e verificação, usaremos a raiz do erro quadrático médio, mais conhecido RMSE que é uma métrica de avaliação amplamente utilizada e reconhecida na comunidade de *machine learning* para medir o desempenho de modelos de regressão.

Para calcular o RMSE, basta tirar a raiz quadrada da média dos erros quadrados, conforme a fórmula abaixo:

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^n (y_i - p_i)^2} \quad (18)$$

Onde:

- n é o número de amostras
- y_i é o valor observado para a amostra i
- p_i é o valor previsto pelo modelo para a amostra i

O RMSE pode ser interpretado como o desvio médio que as previsões têm do alvo e quanto menor o RMSE, melhor o modelo está se saindo. Essas métricas permitirão uma avaliação mais ampla e comparativa dos modelos, sobretudo em cenários reais.

Os melhores resultados simulados para os modelos abordados neste trabalho são apresentados na Tabela 1.

Tabela 1. Resultados da métricas de avaliação.

REDE NEURAL	MÉTRICAS DE AVALIAÇÃO		
	RMSE	DESVIO PADRAO	TEMPO PROCESSAMENTO
MLP	0,0979	0,014	135,7001
ELM	0,0884	0,0015	0,7191
ESN	0,1291	0,0102	1,777

Na Tabela 1 foram observados resultados que demonstram viabilidade entre modelagens e qual da modelagem apresenta uma maior assertividade para utilização da construção da curva de potência de um aerogerador com a utilização da rede neural ELM possuindo um menor RSME, no desvio padrão pois consegue atingir bons resultados com muitas repetições e menor custo computacional pelo tempo de processamento ser considerado baixo.

4.1. Perceptron de Múltiplas Camadas (MLP)

A rede neural utilizada foi uma perceptron de múltiplas camadas (MLP) com uma camada de entrada, uma camada escondida e uma camada de saída. Logo conseguimos construir a figura 2.

No entanto, seu treinamento pode ser computacionalmente custoso, especialmente quando há múltiplas camadas e muitos neurônios. De acordo com os resultados obtidos, o MLP apresentou um RMSE de 0,0979, com um tempo de execução de aproximadamente 135 segundos. Esses valores indicam que, embora o modelo possa atingir uma boa precisão, seu custo computacional é significativamente maior que os demais, o que pode ser um limitador em aplicações em tempo real ou com restrições de hardware.

Durante os testes iniciais com a MLP, observou-se uma tendência ao sobreajuste (overfitting), principalmente em redes com muitas camadas ocultas e épocas de treinamento elevadas. Para contornar essa limitação, foi utilizada validação cruzada (5-fold) e monitoramento do erro de validação para aplicação de early stopping. A taxa de aprendizado também foi ajustada dinamicamente, evitando oscilações no processo de convergência. Nas arquiteturas ELM e ESN, o risco de overfitting é reduzido devido à simplicidade do modelo ou à aleatoriedade dos pesos fixos.

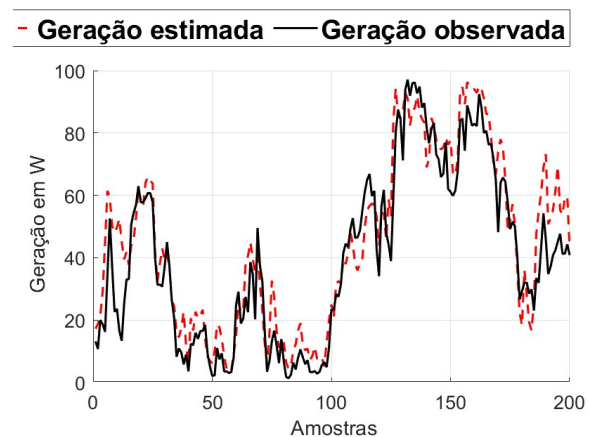


Fig. 2: Curva modelo MLP.

4.2. Máquina de Aprendizado Extremo (ELM)

o ELM é um modelo de rede neural de camada única com treinamento extremamente rápido. Sua principal característica é a atribuição aleatória dos pesos da camada oculta, enquanto os pesos da camada de saída são obtidos por meio de uma solução analítica. Essa simplicidade estrutural permite ao ELM alcançar tempos de treinamento na ordem de milissegundos a segundos, sem comprometer drasticamente a precisão. Na análise realizada, o ELM obteve o menor RMSE (0,0884) e o menor tempo de execução (0,719 segundos) entre os três modelos considerados. Esses resultados confirmam a capacidade do ELM de fornecer previsões eficientes e precisas em contextos de predição ultra-curta ou de resposta rápida.

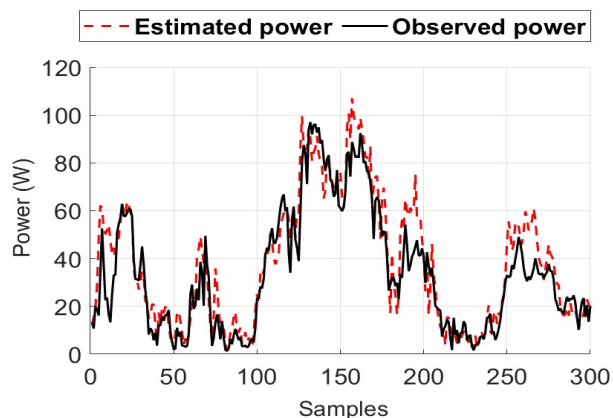


Fig. 2: Curva modelo ELM.

4.3. O modelo ESN

A ESN obteve um RMSE de 0,1291 e um tempo de execução de aproximadamente 1,82 segundos. Embora o tempo de execução seja relativamente baixo, a precisão foi inferior à dos modelos MLP e ELM, sugerindo a necessidade de técnicas de pré-processamento.

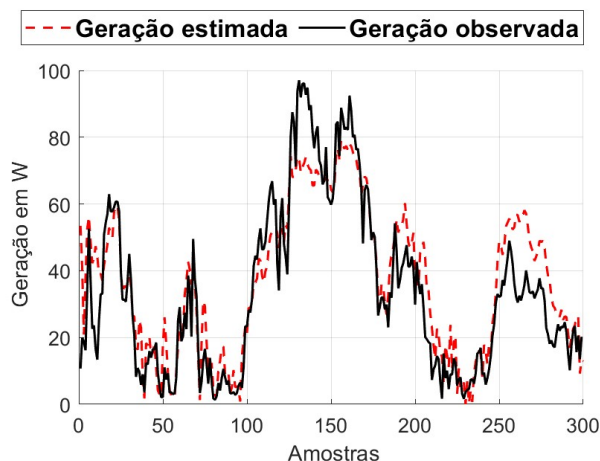


Fig. 4: Curva modelo ESN.

Apesar da boa performance geral dos modelos, é importante considerar sua robustez frente a dados ruidosos. A ELM apresentou maior estabilidade mesmo com pequenas perturbações nos dados de entrada, o que pode estar relacionado à diversidade criada pela aleatoriedade dos pesos iniciais. Por outro lado, a ESN demonstrou sensibilidade à configuração do reservatório, exigindo um ajuste cuidadoso de sua conectividade e do parâmetro de spectral radius para evitar instabilidades na dinâmica interna da rede.

Com base nos resultados alcançados neste trabalho, tornou-se claro que os modelos fundamentados em redes neurais artificiais apresentam um elevado potencial para estimar a curva de potência de turbinas eólicas. Entre as diferentes arquiteturas avaliadas, a Máquina de Aprendizagem Extremo (ELM) se destacou, apresentando o melhor desempenho em precisão (menor RMSE) e rapidez no processamento, emergindo como uma opção extremamente eficiente para situações que requerem respostas rápidas e baixo custo de computação. Apesar de também ter mostrado boa precisão, o modelo MLP enfrenta dificuldades devido ao seu longo tempo de execução, o que restringe sua aplicação em contextos com limitações operacionais. Por outro lado, a ESN teve um desempenho inferior em comparação com os outros modelos, o que sugere a necessidade de ajustes e pré-processamentos extras.

Dessa forma, pode-se afirmar que a seleção correta da arquitetura de rede neural é crucial para a eficácia na modelagem da curva de potência de turbinas eólicas. A implementação de modelos como o ELM pode ser extremamente benéfica para aprimorar o planejamento energético e facilitar uma integração mais eficiente da energia eólica na rede elétrica. Pesquisas futuras podem investigar a fusão de modelos, utilizar conjuntos de dados mais abrangentes e aplicar técnicas de regularização e ajuste de hiperparâmetros para aumentar ainda mais a precisão das previsões em diferentes ambientes operacionais.

AGRADECIMENTOS

Agradeço primeiramente a Deus, por conceder sabedoria, força e perseverança durante toda a jornada e ao programa de pós-graduação em Engenharia Elétrica da Universidade Federal do Piauí pelo apoio acadêmico e estrutural.

REFERÊNCIAS

- [1] Haykin, S. (1999). *Neural Networks: A Comprehensive Foundation* (2nd ed.). Prentice Hall. — Fundamentos do MLP e redes neurais em geral
- [2] [Soman, S. S. et al., 'A review of wind power and wind speed forecasting methods with different time horizons', IEEE PES ISGT Asia, 2010.
- [3] Haykin, S., 'Neural Networks and Learning Machines', 3ª Ed., Pearson Education, 2009.
- [4] Camargo, J. et al., 'Artificial Neural Networks in Wind Power Forecasting: A Review', Renewable and Sustainable Energy Reviews, 2020.
- [5] Silva, Ronnyel Carlos Cunha. Otimização de hiperparâmetros de modelos neurais NARX usando algoritmos PSO e GA para
- [6] Y. Shi et al., "Novel L1-Regularized Extreme Learning Machine and Application," IEEE TII, 2021.
- [7] D. Wu et al., "A Robust Learning Method for Wind Turbine Power Curve Prediction," Applied Energy, 2021.
- [8] T. Zhang et al., "A Hybrid Approach for Estimating Wind Turbine Power Curve using MLP and SVR," MDPI Energies, 2019.
- [9] G. Huang et al., "Extreme Learning Machine: Theory and Applications," Neurocomputing, 2006.