

ANÁLISE DE MÉTODO DE TREINAMENTO DE REDES NEURAIS DO TIPO ELM PARA IMPLEMENTAÇÃO EM HARDWARE.

Luiz Henrique Barra Tavares¹ (IC), Gabriel A. Fanelli de Souza (PQ)¹

¹Universidade Federal de Itajubá

Palavras-chave: Redes neurais. Extreme Learning Machine.

Introdução

Uma Rede Neural Artificial (RNA) tem como objetivo imitar a maneira como o cérebro humano processa informações, uma vez que o cérebro é capaz de realizar tarefas altamente complexas, como o reconhecimento de padrões visuais. Enquanto um computador convencional pode requerer mais tempo para concluir tarefas de igual ou menor complexidade, as RNAs não apenas buscam essa eficiência, mas também têm o potencial de aprender e adaptar-se ao longo do tempo, tornando-as ainda mais poderosas na resolução de problemas complexos [1]. A presente pesquisa demonstra o estudo da implementação em hardware de uma rede neural do tipo Extreme Learning Machine (ELM). Este estudo se apoia nos resultados obtidos na pesquisa realizada anteriormente [2]. Foi conduzido um estudo dos métodos de regressão linear para treinamento da rede, uma vez determinado o método a ser utilizado, o estudo voltou-se para as adequações necessárias para a descrição em hardware.

Metodologia

Tomando como base as respostas dos neurônios e os modelos de superfície obtidos na pesquisa anterior, o estudo teve como foco obter o modelo matemático que melhor caracteriza o método empregado no aprendizado da rede neural. Com o modelo em nossa posse, avançamos em direção a correção de potenciais deficiências técnicas do mesmo, visando aprimorar a abrangência da rede neural, ou seja, aumentar a compatibilidade da rede com diferentes respostas de neurônios e superfícies alvo, dentre as correções implementadas, está uma ferramenta matemática conhecida como regularização de Tikhonov. A solução que pode ser obtida utilizando o método dos mínimos quadrados é:

$$w = (XX^T)^{-1} X^T y$$

Utilizando agora a técnica de regressão, obtermos:

$$w = (XX^T + \lambda I)^{-1} X^T y$$

Em que o termo I é a matriz identidade e o termo λ é o parâmetro de regularização:

O parâmetro de regularização não possui um valor pré definido, ele precisa ser determinado caso a caso. No caso estudado, ele foi definido seguindo dois passos: primeiramente foi calculado para cada superfície estudada o valor do parâmetro que causava o menor impacto no erro, em seguida a média aritmética simples dos valores obtidos foi calculada e este valor utilizado como parâmetro.

O próximo passo foi estudar o impacto das características físicas do circuito a ser utilizado na implementação em hardware, nesse ponto, optou-se por realizar uma maior filtragem nos dados utilizados, obtendo assim uma precisão muito próxima à obtida anteriormente, porém, permitindo uma simplificação nos recursos utilizados pelo circuito. O modelo matemático devidamente polido foi, então, testado nos modelos obtidos na pesquisa anterior e também no modelo obtido em [3]. As Superfícies obtidas por meio do treinamento, bem como as superfícies de erro absoluto e o RMSE (Raiz do erro médio quadrático) dos modelos foram comparadas aos modelos anteriores.

Resultados e discussão

A seguir são apresentadas algumas SCs (Superfícies de controle) obtidas por meio do treinamento direto e por meio do treinamento desenvolvido:

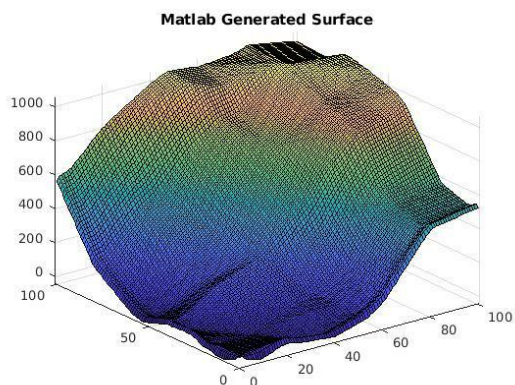


Figura 1: SC gerada para [3].

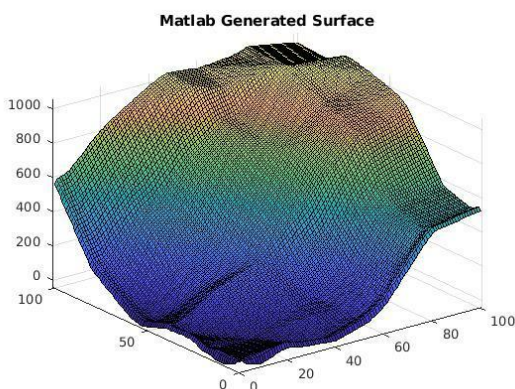


Figura 2: SC gerada pelo treinamento da rede neural.

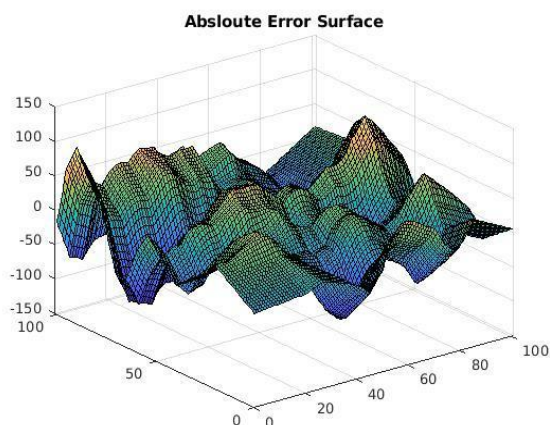


Figura 3: Superfície de erro absoluto.

A Figura 1 exibe a SC obtida para o artigo [3], em seguida, a Figura 2 exibe a SC obtida com o desenvolvimento do treinamento da rede neural. Posteriormente, ao analisar a superfície de erro absoluto, representada na Figura 3, é factível examinar as discrepâncias em cada ponto entre as superfícies de controle (SCs) produzidas nas Figuras 1

e 2. Isso resulta em um valor de Raiz do Erro Médio Quadrático (RMSE) igual a 3,34% do total.

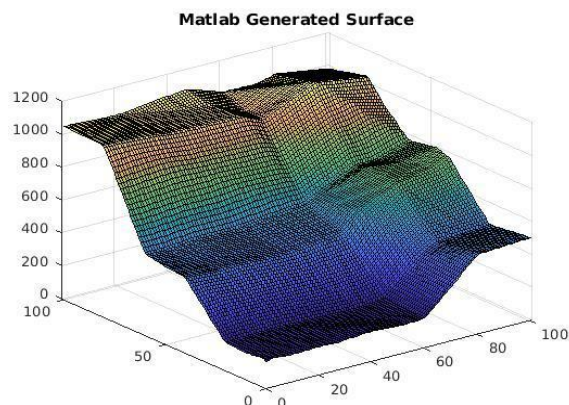


Figura 4: SC gerada para [4].

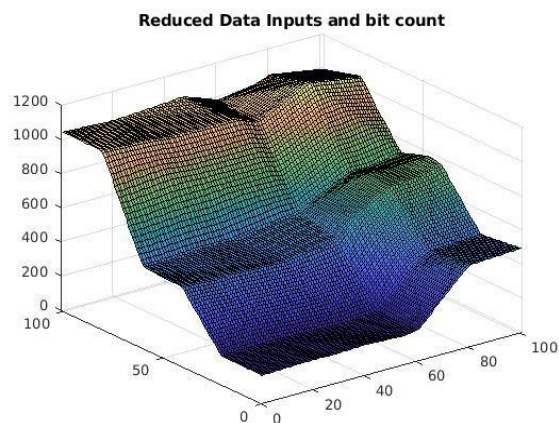


Figura 5: SC gerada pelo treinamento da rede neural.

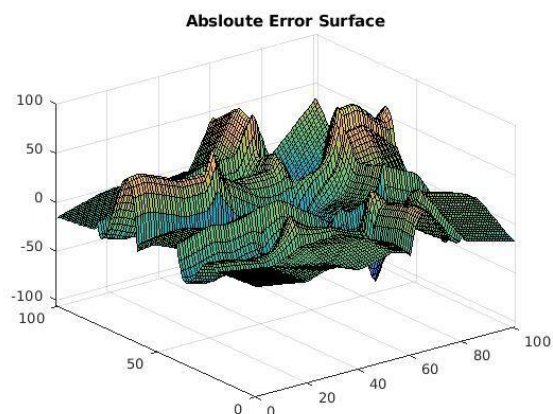


Figura 6: Superfície de erro absoluto.

A Figura 4 apresenta a SC obtida para o artigo

mencionado [4]. Em seguida, na Figura 5, podemos observar a SC resultante após o desenvolvimento do treinamento da rede neural. Após analisar a superfície de erro absoluto, representada na Figura 6, é notável o erro em cada ponto entre as SCs geradas nas Figuras 4 e 5, resultando em um RMSE de 2,27% do valor total.

Tabela 1: Comparação entre RMSE obtidos anteriormente dos artigos estudados, e pelo modelo de aprendizado da rede neural.

Referência	RMSE (%)	RMSE obtido pela RN (%)
[3]	3,09%	3,34%
[4]	2,12%	2,27%
[5]	4,61%	4,81%
[6]	3,30%	3,76%
[7]	3,26%	3,44%
[8]	3,92%	4,05%

A Tabela 1 apresenta um resumo dos RMSEs obtidos nos modelos estudados e os compara com os RMSEs obtidos com a RN treinada utilizando o método desenvolvido na presente pesquisa.

A figura abaixo demonstra a relação entre o parâmetro de regularização inserido na rede e a variação do RMSE obtido.

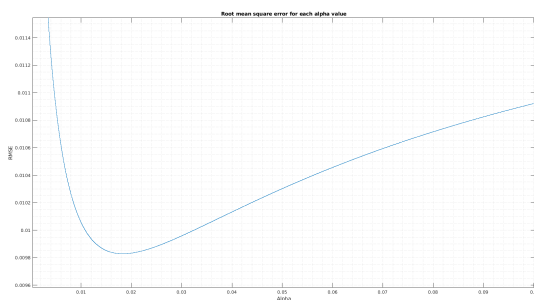


Figura 7: Ajuste do parâmetro de regularização da SCs obtido em [7].

O parâmetro foi otimizado de acordo com o parâmetro médio entre as Superfícies de Controle estudadas.

Conclusões

Com base nos resultados obtidos, posso afirmar que, o modelo de treinamento desenvolvido neste projeto, o qual utiliza como base o método dos mínimos quadrados e a regularização de Tikhonov, consegue aproximar todas as superfícies de controle que no âmbito desta pesquisa, foram investigados, revelando um incremento de, no máximo 0,46%, além disso, o modelo ainda aumenta a estabilidade da rede, melhorando a capacidade da mesma de se adequar a diferentes respostas dos neurônios, algo que, no caso de estudo dessa pesquisa, se torna uma vantagem.

Sendo assim, é esperado que o modelo desenvolvido por esta pesquisa, quando implementado em FPGA e/ou um chip dedicado consiga, realizar o treinamento da rede de neurônios baseados em espelho de corrente.

Agradecimento

Expresso minha gratidão ao CNPq pelo valioso apoio financeiro concedido a este estudo, cuja realização teria sido inviável sem o seu suporte. Além disso, gostaria de estender meus agradecimentos à UNIFEI - Universidade Federal de Itajubá, pela valiosa contribuição com o ambiente propício à pesquisa.

Referências

[1] HUANG, Guang-Bin; ZHU, Qin-Yu; SIEW, Chee-Kheong. Extreme learning machine: a new learning scheme of feedforward neural networks. In: 2004 IEEE international joint conference on neural networks (IEEE Cat. No. 04CH37541). Ieee, 2004. p. 985-990.

[2] FERREIRA, Phillipe de Carvalho Tavares Rezende, Análise da Viabilidade para Desenvolvimento de Arquitetura Neuromórfica de Baixa Potência Visando a Implementação de um Controlador MPPT para Célula Solar Fotovoltaica em um Sistema de Energy Harvesting. 2022. P.0-31.

[3] FANELLI DE SOUZA, Gabriel & SANTOS, Rodrigo & FARIA, Lester. (2019). Low-Power Current-Mode Interval Type-2 Fuzzy Inference Engine Circuit. IEEE Transactions on Circuits and Systems I: Regular Papers. PP. 1-12. 10.1109/TCSI.2019.2899505.

[4] PRAUZEK, Michal et al. Fuzzy energy management of autonomous weather station. IFAC-PapersOnLine, v. 48, n. 4, p. 226-229, 2015.

- [5] BENDIB, B. et al. Advanced Fuzzy MPPT Controller for a stand-alone PV system. *Energy Procedia*, v. 50, p. 383-392, 2014.
- [6] GUENOUNOU, Ouahib; DAHHOU, Boutaib; CHABOUR, Ferhat. Adaptive Fuzzy controllerbased MPPT for photovoltaic systems. *Energy Conversion and Management*, v. 78, p. 843-850, 2014.
- [7] ZAINURI, MAA Mohd et al. Adaptive P&O-Fuzzy control MPPT for PV boost dc-dc converter. In: 2012 IEEE International Conference on Power and Energy (PECon). IEEE, 2012. p. 524-529.
- [8] ALTIN, Necmi. Single phase grid interactive PV system with MPPT capability based on type-2 Fuzzy logic systems. In: 2012 International Conference on Renewable Energy Research and Applications (ICRERA). IEEE, 2012. p. 1-6.