

Exercício 2

Matheus Arataque Uema - 10276949

SCC5809 - Redes Neurais e Aprendizado Profundo

1 Exercício I

Para implementação de um XOR, uma rede neural simples possui dificuldades em resolver devido a dificuldade de se traçar uma reta que consiga separar bem as saídas de 1 e 0 decorrente das entradas. Por contada disso, utiliza-se um *Multi-Layer Perceptron*, onde adição de uma nova camada à rede neural possibilita que o algoritmo delimite melhor a região de verdadeiro e falso das saídas.

Para implementá-la, desenvolveu-se um código em *Python*¹, com o auxílio da biblioteca *NumPy*² para computação científica. Desenvolveu-se uma classe para o MLP que utiliza como função de ativação a sigmoid, e para o *backward* utiliza-se a derivada da sigmoide para ajuste dos pesos.

$$\sigma = \frac{1}{1 + e^x}$$

$$\sigma' = \sigma(1 - \sigma)$$

Utilizando como entrada a matriz abaixo, foram registradas duas saídas. Uma antes da aplicação e treinamento do modelo, e outra após. Para o modelo, foram utilizadas 10000 iterações e uma taxa de aprendizado de 0.5, o que gerou a Figura 1 com um erro convergindo para 0.1%.

$$\text{Entradas} = \begin{bmatrix} x_1 & x_2 \\ 0 & 0 \\ 0 & 1 \\ 1 & 0 \\ 1 & 1 \end{bmatrix}$$
$$\text{Saída1} = \begin{bmatrix} y_1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \quad \text{Saída2} = \begin{bmatrix} y_2 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}$$

¹Python.org Disponível em: <https://www.python.org/>. Acesso em: 12 de set. de 2024.

²NumPy Disponível em: <https://numpy.org/>. Acesso em: 12 de set. de 2024.

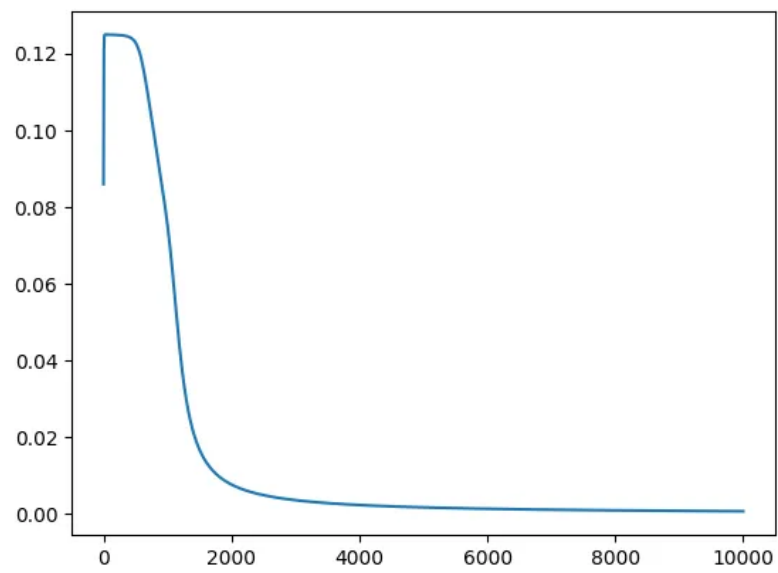


Figure 1: Gráfico do Erro por Iteração para o Xor

2 Exercício II

Para o *AutoEncoder*, foi utilizada a biblioteca *Torch* para criação das camadas do neurônio. Foi usada a ReLU para o *Encoder* e a Sigmoid para o *Decoder*. A partir do modelo criado e com 30000 iterações com a taxa de aprendizado de 0.5, foram obtidas as Figuras 2 e 3, com erros de 0.04% para matriz identidade 8x8 e 0.02% para 11x11.

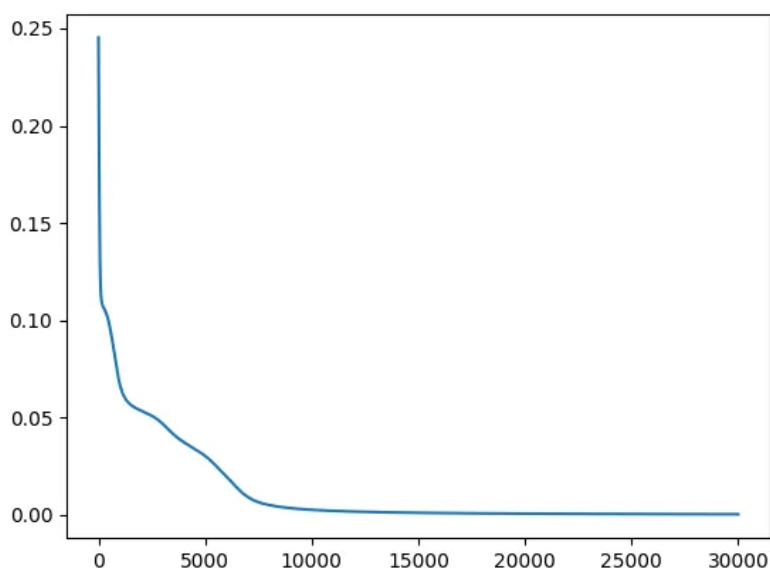


Figure 2: Gráfico do Erro por Iteração para o Autoencoder para Matriz Identidade 8x8

A implementação inteira do código, assim como a base de teste criada, pode ser visualizada no repositório criado no *GitHub*³, onde também contém instruções para a execução do código.

³Repositório do Código Disponível em: <https://github.com/MatheusUema/neural-networks-study>. Acesso em: 12 de set. de 2024.

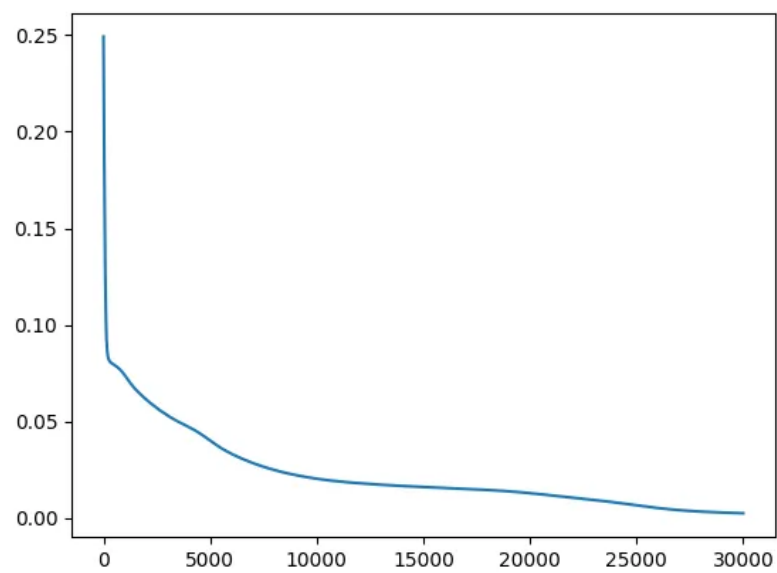


Figure 3: Gráfico do Erro por Iteração para o Autoencoder para Matri Identidade 11x11