

Relatório — Backpropagation do Zero (XOR e Display de 7 Segmentos)

João Vitor Gonçalves Del Cantoni Baldo
PUC Minas

27 de outubro de 2025

Resumo

Resumo. Foi implementado do zero uma rede neural Perceptron com backpropagation para dois problemas: (i) XOR; (ii) reconhecimento de dígitos (0–9) em display de 7 segmentos. Foi avaliado o desempenho por MSE e acurácia, e a robustez com ruído de flip de segmento (probabilidade p). Foram comparadas duas codificações de saída (4 bits vs. one-hot de 10 neurônios) e ativações (sigmoide vs. ReLU na oculta). Os resultados indicaram 100% de acerto sem ruído em todos os cenários e melhor robustez para one-hot + sigmoide (0,688 em $p=0,10$ e 0,496 em $p=0,20$).

1 Introdução

A lista de exercício 8 tem como objetivo implementar do zero, sem utilizar bibliotecas como Keras, PyTorch ou Scikit-learn, o algoritmo de backpropagation e treinar nos conjuntos XOR e 7 segmentos e apresentar um relatório com equações de ajuste dos pesos.

2 Dados e Métodos

2.1 Arquiteturas

XOR: 2–2–1 (sigmoide na oculta e na saída).

7 segmentos: entrada com 7 bits; camada oculta com 5 neurônios; saídas em dois formatos:

- **bits4:** 4 neurônios (código binário do dígito);
- **onehot10:** 10 neurônios (vetor one-hot).

Ativações na camadas ocultas comparadas: sigmoide e ReLU.

2.2 Custo e Backpropagation

Custo quadrático médio (MSE): $E = \frac{1}{2} \sum_k (y_k - t_k)^2$.

Para sigmoide + MSE, os deltas são:

$$\begin{aligned}\delta_k &= (y_k - t_k) y_k (1 - y_k) \quad (\text{camada de saída}) \\ \delta_j &= \left(\sum_k w_{jk} \delta_k \right) h_j (1 - h_j) \quad (\text{camada oculta})\end{aligned}$$

Gradientes: $\frac{\partial E}{\partial w_{j,i}^{(l)}} = \delta_j^{(l)} a_i^{(l-1)}$ e $\frac{\partial E}{\partial b_j^{(l)}} = \delta_j^{(l)}$.
 Atualização: $w \leftarrow w - \eta \nabla E$, $b \leftarrow b - \eta \nabla E$.

2.3 Ruído, treino e avaliação

Ruído por flip independente de segmentos com probabilidade p .

Treino com $p=0,05$ (data augmentation). Teste/robustez em $p \in \{0,00, 0,05, 0,10, 0,20\}$.

Hiperparâmetros (exemplo): XOR=8000 épocas, 7-seg=10000 épocas, $n=0.5$, semente fixa.

3 Resultados

3.1 XOR

MSE final: 0,083656. Saídas:

```
[0, 0] -> 0.0234 (bin 0)
[0, 1] -> 0.6620 (bin 1)
[1, 0] -> 0.6620 (bin 1)
[1, 1] -> 0.6635 (bin 1)
```

Histórico final do MSE: ... 0.083666, 0.083656.

3.2 Display de 7 segmentos (0–9)

Acurácia sem ruído: **100,0%**. Matriz de confusão (limpa): diagonal perfeita (10×10).

Comparação de robustez (acurácia média)

Saída	Oculto	$p=0,00$	$p=0,05$	$p=0,10$	$p=0,20$
bits4	sigmoide	1.000	0.844	0.616	0.404
bits4	ReLU	1.000	0.846	0.640	0.418
onehot10	sigmoide	1.000	0.858	0.688	0.496
onehot10	ReLU	1.000	0.858	0.636	0.468

Tabela 1: Acurácia com ruído de teste (flip de segmento) para diferentes arquiteturas.

MSEs (exemplos por cenário): bits4/sigmoide: 0,005400; bits4/ReLU: 0,001916; onehot10/sigmoide: 0,004306; onehot10/ReLU: 0,052713.

4 Discussão

Sem nenhum ruído, todos os modelos alcançaram 100% de acerto, já com ruído, a codificação one-hot com sigmoide na oculta apresentou maior robustez, principalmente para $p \geq 0,10$. Em bits4, a ReLU reduziu o MSE mas não superou a robustez de one-hot + sigmoide.

Observação: para a variante onehot10, é comum empregar softmax + entropia cruzada, o que tende a melhorar estabilidade e separação entre classes. Foi mantido sigmoide + MSE para consistência com a implementação base.

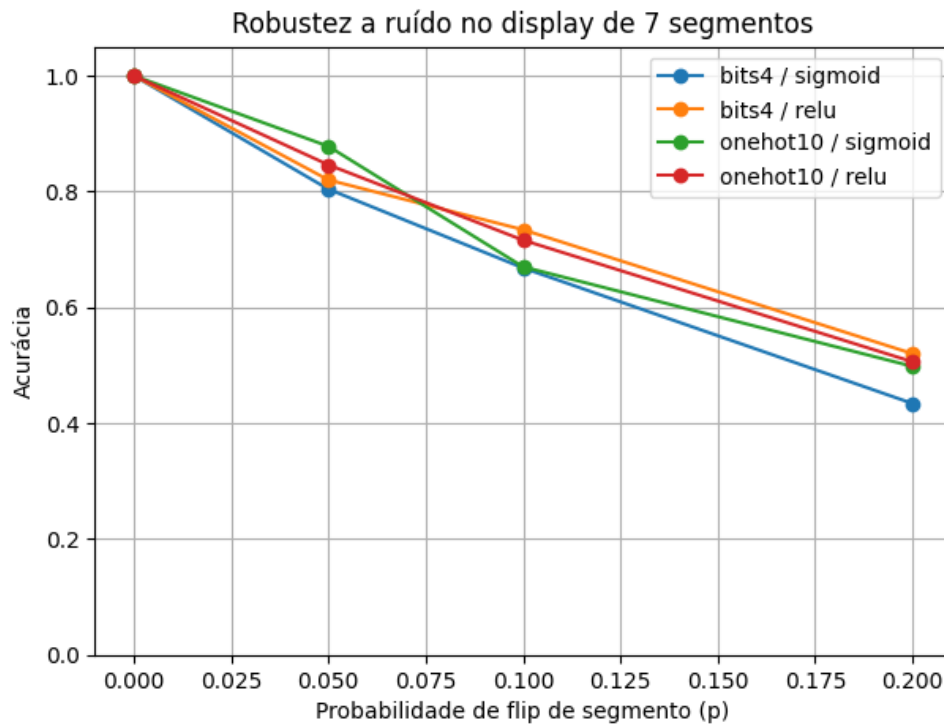


Figura 1: Robustez (Acurácia \times probabilidade de flip de segmento).

5 Conclusões

(1) A MLP do zero resolve o XOR e reconhece 0–9 com 100% sem ruído. (2) Para robustez a flip de segmentos, onehot10 + sigmoide obteve as melhores médias. (3) Próximos passos: softmax + cross-entropy na saída one-hot, varredura de hiperparâmetros e noise schedules no treino.

6 Artefatos

relatorio_resultados.txt (fórmulas e logs), robustez.csv, robustez_resumo_do_usuario.csv, robustez_plot.png, código: backprop_do_zero_xor_e_7_segmentos_codigo_base.py.

7 Código

O código está disponível no GitHub: <https://github.com/JoaoVitorB1/IAexs/tree/main/Lista%208>