

Atividade Playground – Problema da Espiral

Disciplina: Ciência de Dados

Data: 17/08/2025

Aluno: Douglas F. Azevedo

Professor: Vitor Casadei

Introdução

Para esta atividade, realizei experimentos no TensorFlow Playground com o objetivo de resolver o problema da espiral usando apenas duas entradas (x_1 e x_2), conforme as orientações da disciplina. O desafio principal foi alcançar resultados satisfatórios ($\text{test loss} < 0.1$) utilizando o mínimo possível de recursos computacionais, especialmente diminuindo a quantidade de neurônios por camada.

Estratégia e Achados Principais

Diversifiquei combinações dos parâmetros, buscando soluções eficientes tentando não usar o máximo de neurônios permitidos pela ferramenta. Nas próximas páginas apresento os principais resultados obtidos com suas respectivas configurações e print das soluções.

Solução 1

- **Learning rate:** 0.1
- **Função de ativação:** ReLU
- **Arquitetura:** 2 camadas ocultas com 8 neurônios cada
- **Test loss:** 0.031
- **Training loss:** 0.011

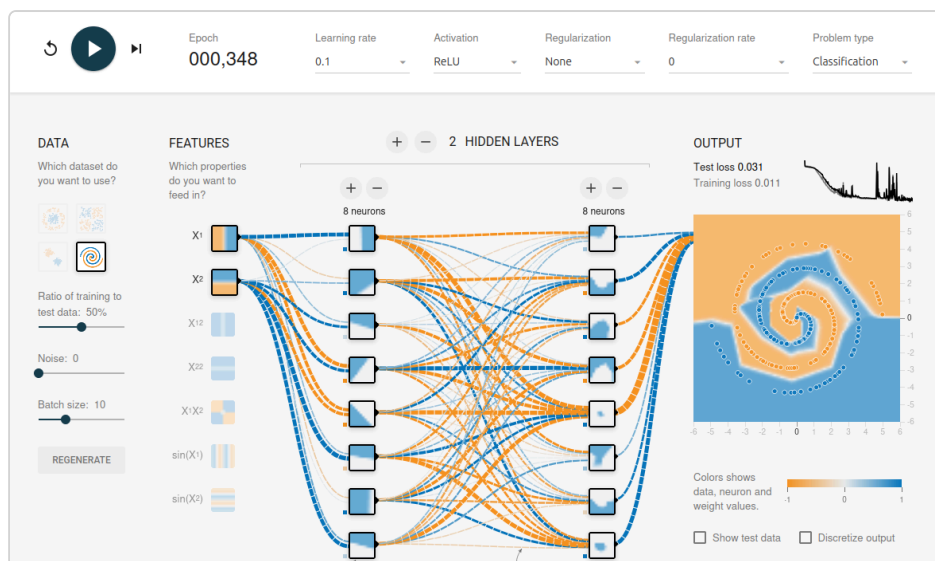


Figura 1: rede com 2 camadas de 8 neurônios e ativação ReLU

Solução 2

- **Learning rate:** 0.1
- **Função de ativação:** Tanh
- **Arquitetura:** 2 camadas ocultas com 8 neurônios cada
- **Test loss:** 0.007
- **Training loss:** 0.004

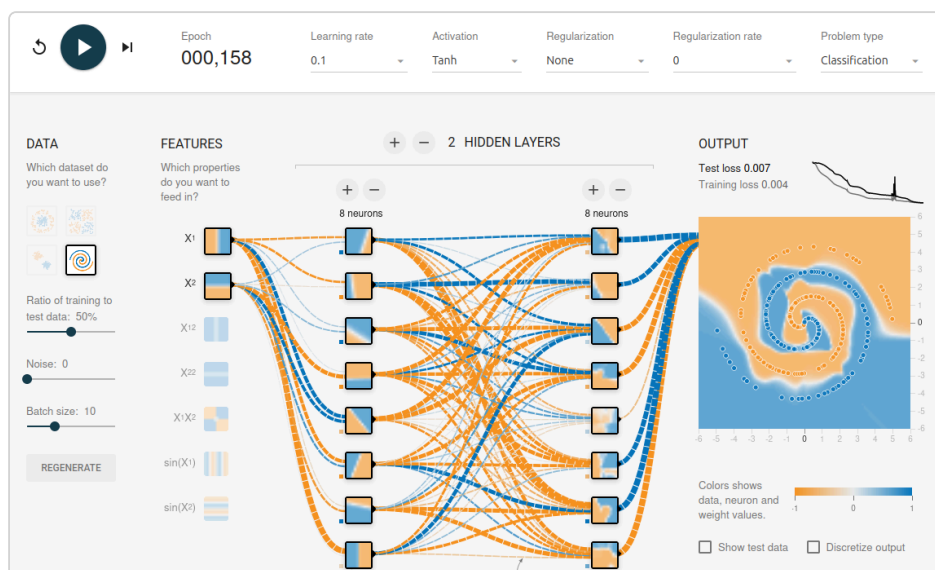


Figura 2: rede com 2 camadas de 8 neurônios e ativação Tanh

Solução 3

- **Learning rate:** 0.1
- **Função de ativação:** Sigmoid
- **Arquitetura:** 2 camadas ocultas com 8 neurônios cada
- **Test loss:** 0.031
- **Training loss:** 0.001

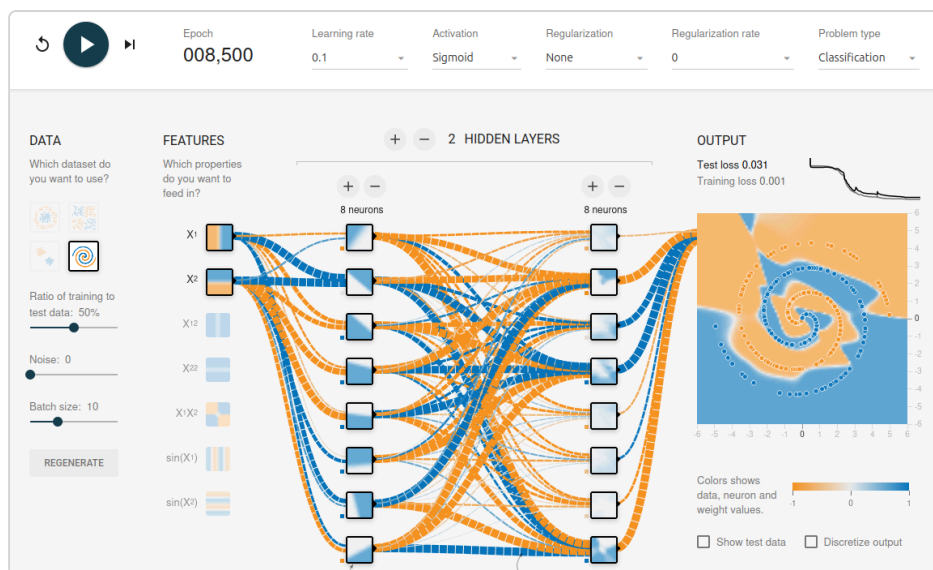


Figura 3: rede com 2 camadas de 8 neurônios e ativação Sigmoid

Solução 4

- **Learning rate:** 0.1
- **Função de ativação:** ReLU
- **Arquitetura:** 2 camadas ocultas, 8 e 6 neurônios
- **Test loss:** 0.007
- **Training loss:** 0.005

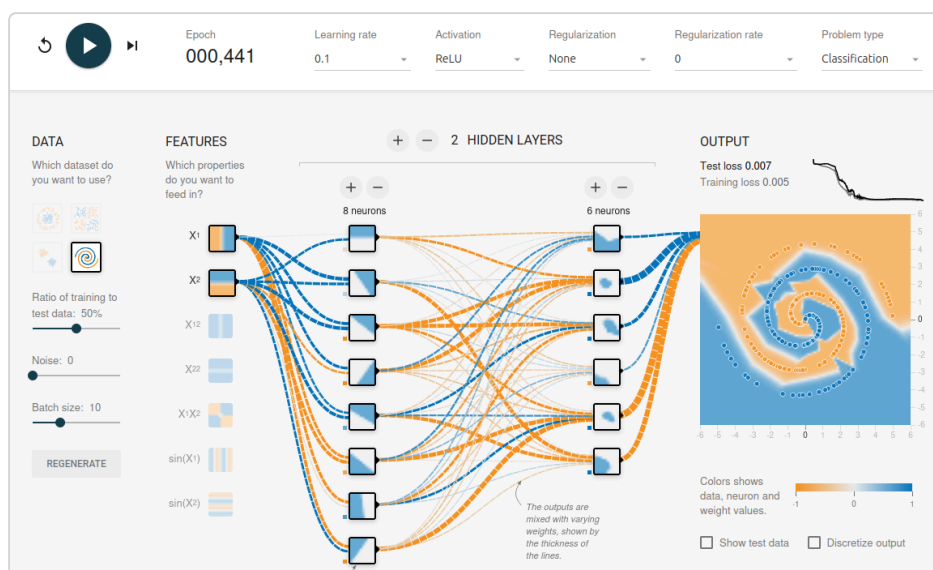


Figura 4: rede com 2 camadas de 8 e 6 neurônios, ativação ReLU

Solução 5

- **Learning rate:** 0.1
- **Função de ativação:** ReLU
- **Arquitetura:** 2 camadas ocultas, 8 e 5 neurônios
- **Test loss:** 0.058
- **Training loss:** 0.056

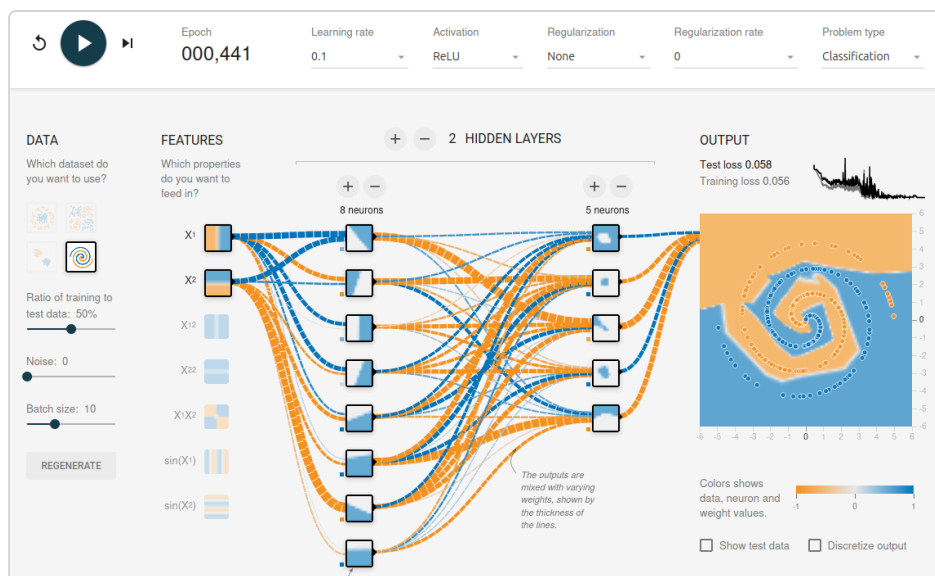


Figura 5: rede com 2 camadas de 8 e 5 neurônios, ativação ReLU

Solução 6

- **Learning rate:** 0.1
- **Função de ativação:** Tanh
- **Arquitetura:** 2 camadas ocultas, 8 e 4 neurônios
- **Test loss:** 0.072
- **Training loss:** 0.012

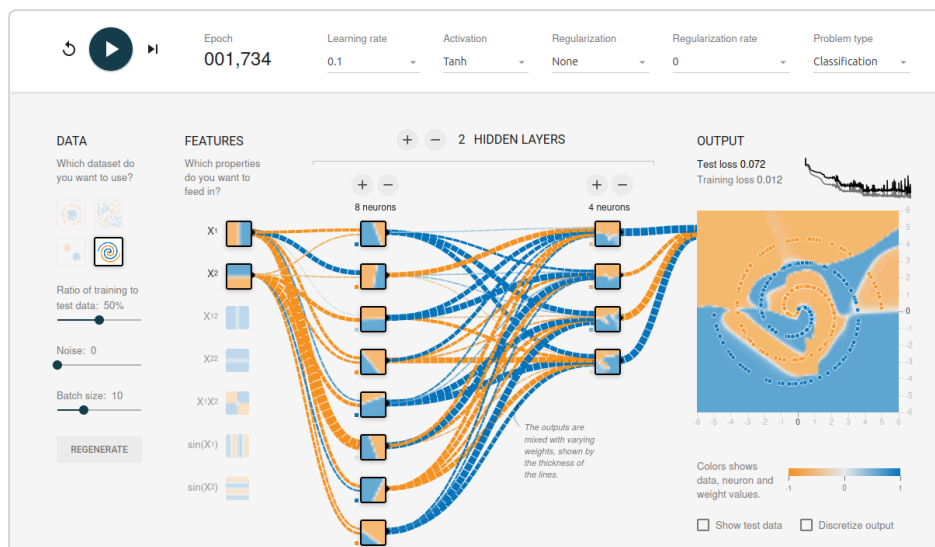


Figura 6: rede com 2 camadas de 8 e 4 neurônios, ativação Tanh

Análise e Considerações

- Aumentar o número de neurônios por camada revelou-se fundamental para que a rede consiga captar a complexidade da espiral.
- A função Tanh apresentou o melhor desempenho geral (menor test loss), seguido da ReLU próximo, proporcionando também convergência rápida.
- Reduzir neurônios na segunda camada mostrou que a rede ainda consegue resolver o problema, embora com uma ligeira queda de performance.
- A profundidade da rede (mais de duas camadas) não trouxe ganhos relevantes, reafirmando que a largura (neurônios por camada) é o aspecto mais crítico para este problema específico.
- A taxa de aprendizado 0.1 foi ideal e estável para as configurações testadas. Nas variações que tentei desta taxa, não obtive bons resultados.
- Os prints apresentados ilustram apenas as soluções que efetivamente resolveram o problema (test loss < 0.1) dentre diversas outras combinações testadas sem sucesso.

Comparativo entre os melhores achados encontrados

Solução	Learning Rate	Ativação	Camadas Ocultas	Test Loss	Training Loss
1	0.1	ReLU	2 camadas (8 e 8 neurônios)	0.031	0.011
2	0.1	Tanh	2 camadas (8 e 8 neurônios)	0.007	0.004
3	0.1	Sigmoid	2 camadas (8 e 8 neurônios)	0.031	0.001
4	0.1	ReLU	2 camadas (8 e 6 neurônios)	0.007	0.005
5	0.1	ReLU	2 camadas (8 e 5 neurônios)	0.058	0.056
6	0.1	Tanh	2 camadas (8 e 4 neurônios)	0.072	0.012

Conclusão

Neste estudo, foram identificadas configurações que resolvem o problema da espiral, priorizando o uso reduzido de neurônios e buscando equilíbrio entre simplicidade e eficiência computacional.

Diversas variações foram testadas, como número de camadas ocultas, taxas de aprendizado e funções de ativação; contudo, as combinações diversas não foram listadas no comparativo por apresentaram falhas na solução.

A configuração mais enxuta — com duas camadas ocultas de 8 e 4 neurônios — também resolveu o problema, com test loss de 0.072. Isso reforça a eficiência da função de ativação tangente hiperbólica (Tanh) para arquiteturas compactas. Essas análises mostram que otimizar a rede com menos neurônios pode garantir soluções robustas, valorizando o custo computacional sem perder qualidade nos resultados.