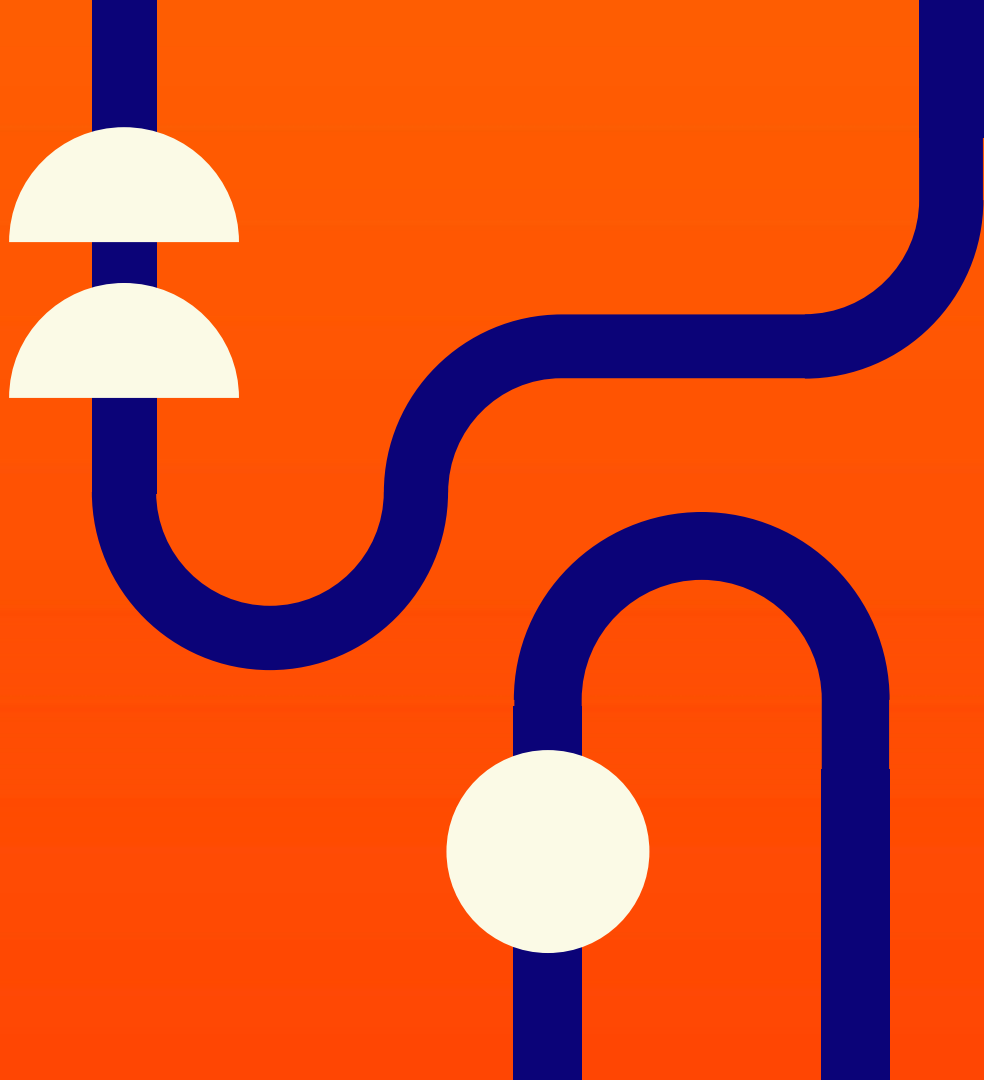




especialização

Tech Lead Data Science & IA

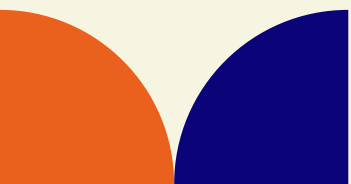
Gabriela Lindenberg
gcl2@cesar.school



Atividade Playground

As funções de ativação são essenciais em redes neurais — elas determinam como os neurônios “disparam” e ajudam a rede a aprender padrões complexos. Problema em espiral é altamente não linear, ou seja, os dados estão dispostos de forma "enrolada", e separar as classes exige que a rede neural aprenda padrões complexos e curvos. Temos 4 opções de ativação. Quanto mais complexo o data set, precisamos de ativações mais complexas que a Linear por exemplo. Para esse problema as ativações mais indicadas são ReLu (Rectified Linear Unit) e Tanh (Tangente Hiperbólica).

Para resolver o problema variei as ativações, o learning rate e adicionei mais neurônios nas duas camadas. Após observar que algumas linhas tinham menos fluxo, menor peso, testei remover o neurônio com menor fluxo para chegar na quantidade ideal que resolvesse o problema.



Atividade Playground

Teste 1:

Data set complexo, então optei pela ativação ReLu e learning rate de 0,1. Escolhi essa learning rate porque é uma taxa de aprendizagem rápido e estável e chega mais fácil em um bom resultado. O test loss ficou alto, as linhas dos pesos grossas (imagem 1), então testei adicionar mais neurônios (imagem 2). Ainda assim o resultado não chegou onde eu queria, apesar do test loss já ter alcançado o valor menor que 0,1.



Epoch
000,425

Learning rate
0.1

Activation
ReLU

Regularization
None

Regularization rate
0

Problem type
Classification

Imagen 1

DATA

Which dataset do you want to use?



Ratio of training to test data: 50%



Noise: 0



Batch size: 10



REGENERATE

FEATURES

Which properties do you want to feed in?

- X_1
- X_2
- X_1^2
- X_2^2
- $X_1 X_2$
- $\sin(X_1)$
- $\sin(X_2)$

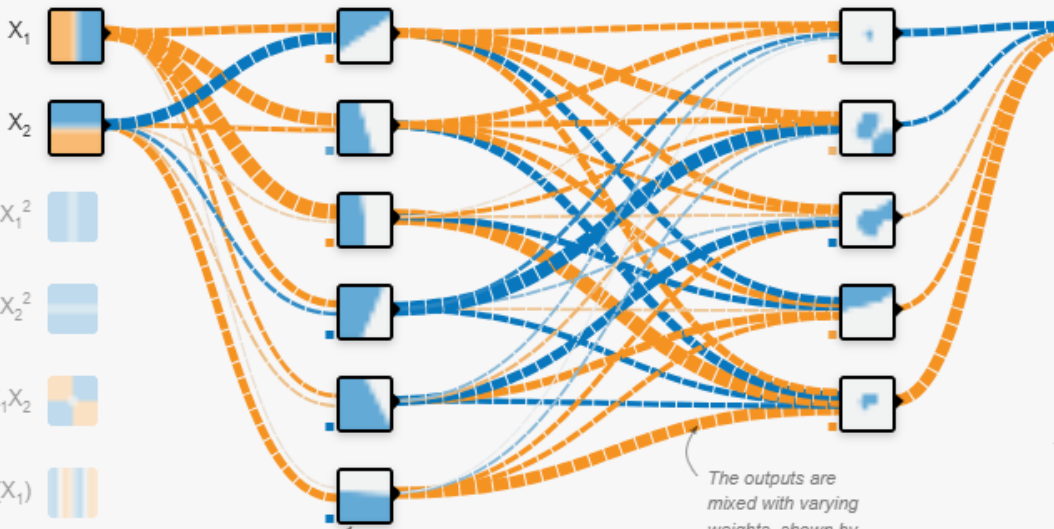
2 HIDDEN LAYERS

+ -

6 neurons

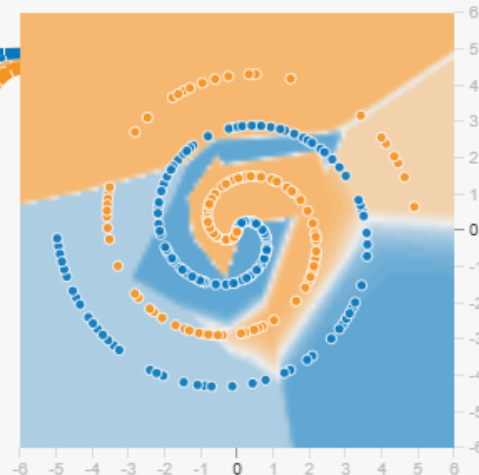
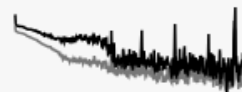
+ -

5 neurons

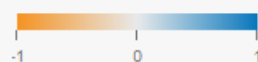


OUTPUT

Test loss 0.308
Training loss 0.349



Colors shows data, neuron and weight values.





Epoch
000,545

Learning rate
0.1

Activation
ReLU

Regularization
None

Regularization rate
0

Problem type
Classification

Imagen 2

DATA

Which dataset do you want to use?



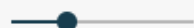
Ratio of training to test data: 50%



Noise: 0



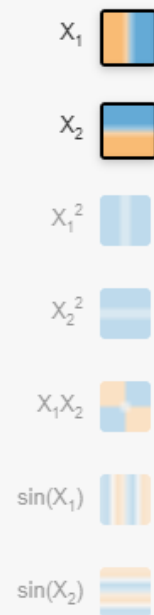
Batch size: 10



REGENERATE

FEATURES

Which properties do you want to feed in?



+ - 2 HIDDEN LAYERS

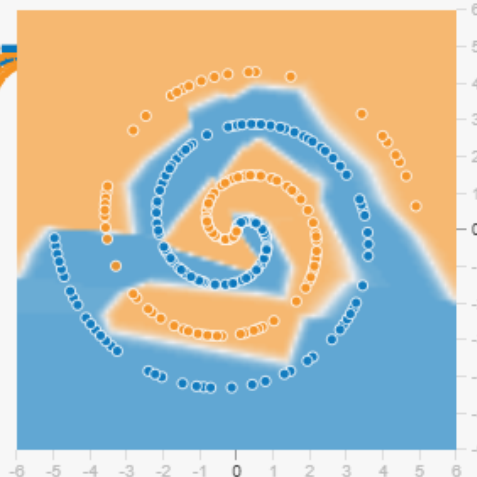
+ -
8 neurons

+ -
8 neurons

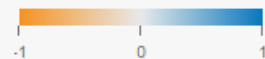
OUTPUT

Test loss 0.089

Training loss 0.022



Colors shows data, neuron and weight values.



Atividade Playground

Teste 2:

Altere a ativação de ReLu para Tanh e mantive a learning rate de 0,1 e a quantidade máxima de 16 neurônios, mas ainda não cheguei no resultado desejado (imagem 3); executei 3 vezes mais épocas no segundo teste. Então retomei a ativação para ReLu e alterei o learning rate, obtendo um resultado mais satisfatório (imagem 4) com test loss 0,089.



Epoch
001,211

Learning rate
0.1

Activation
Tanh

Regularization
None

Regularization rate
0

Problem type
Classification

Imagen 3

DATA

Which dataset do you want to use?



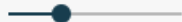
Ratio of training to test data: 50%



Noise: 0



Batch size: 10



REGENERATE

FEATURES

Which properties do you want to feed in?

- X_1
- X_2
- X_1^2
- X_2^2
- $X_1 X_2$
- $\sin(X_1)$
- $\sin(X_2)$

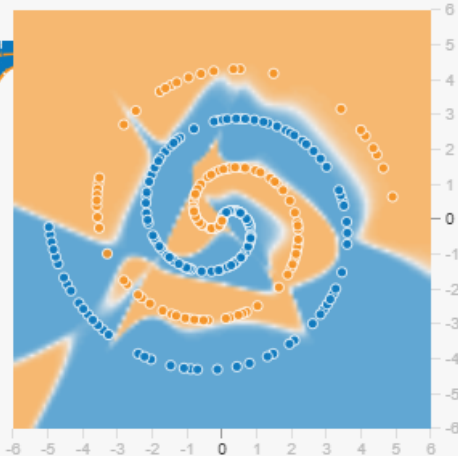
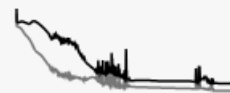
2 HIDDEN LAYERS

+ -
8 neurons

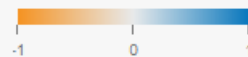
+ -
8 neurons

OUTPUT

Test loss 0.062
Training loss 0.009



Colors shows data, neuron and weight values.





Epoch
001,324

Learning rate

0.001

Activation

ReLU

Regularization

None

Regularization rate

0

Problem type

Classification

Imagen 4

DATA

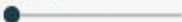
Which dataset do you want to use?



Ratio of training to test data: 50%



Noise: 0



Batch size: 10



REGENERATE

FEATURES

Which properties do you want to feed in?



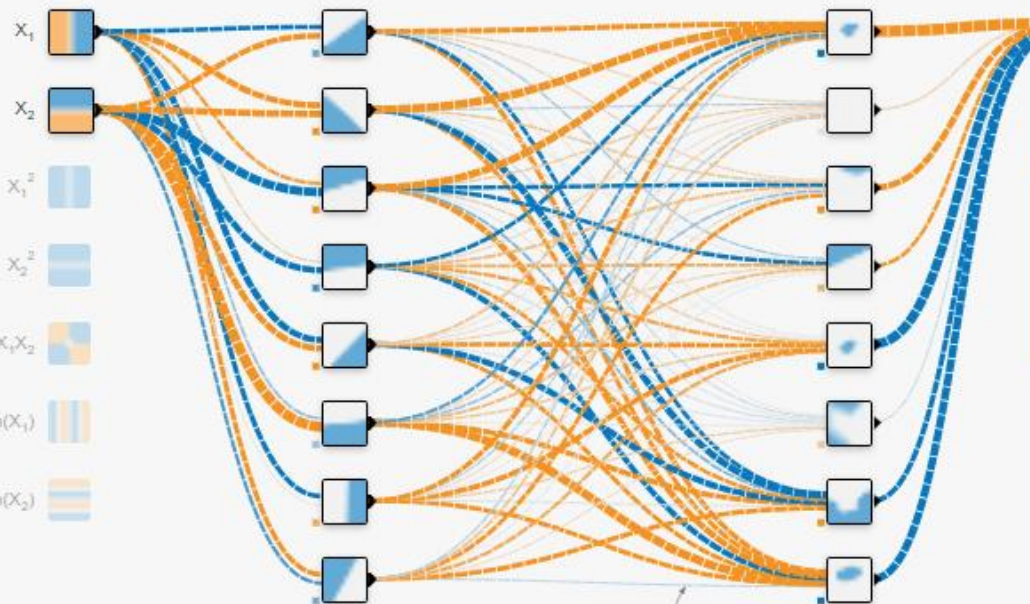
+ - 2 HIDDEN LAYERS

+ -

8 neurons

+ -

8 neurons



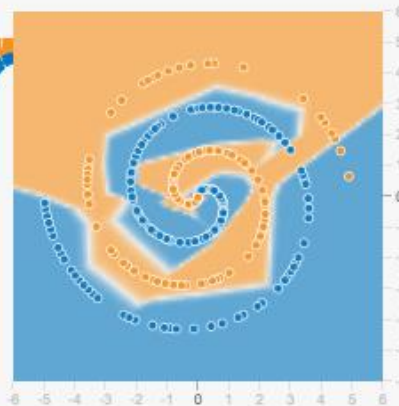
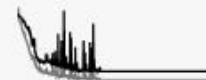
This is the output from one neuron. Hover to see it

The outputs are mixed with varying weights, shown by the thickness of the lines

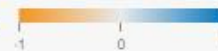
OUTPUT

Test loss 0.089

Training loss 0.025



Colors shows data, neuron and weight values.



☐ Show test data

☐ Discretize output

Atividade Playground

Teste 3:

Após realizar uma breve pesquisa no Chatgpt repeti a configuração que ele sugeriu, mas modifiquei a ativação para ReLu; learning rate 0,01, regularization L2 e regularization rate de 0,003 e **obtive um resultado satisfatório (imagem 5)**. Testei com a ativação Tanh, que foi a sugerida pelo Chatgpt, mantendo as outras configurações iguais e observei que o modelo também chegou em um resultado satisfatório (imagem 6), porém precisou de muito mais épocas e ainda ainda **o resultado melhor foi obtivo pela ReLu**.

Atividade Playground

CONCLUSÃO:

A ReLU foi mais eficiente porque:

- Não sofre com saturação como a Tanh.
- Conseguiu aproveitar melhor os 8 neurônios por camada para formar fronteiras precisas.
- Regularização controlou o risco de overfitting, mantendo o modelo limpo.
- O learning rate baixo deu estabilidade no ajuste.
- **Melhor configuração -> IMAGEM 5; test loss 0,048**



Epoch
006,697

Learning rate
0.01

Activation
ReLU

Regularization
L2

Regularization rate
0.003

Problem type
Classification

Imagen 5

DATA

Which dataset do you want to use?



Ratio of training to test data: 70%



Noise: 0



Batch size: 10



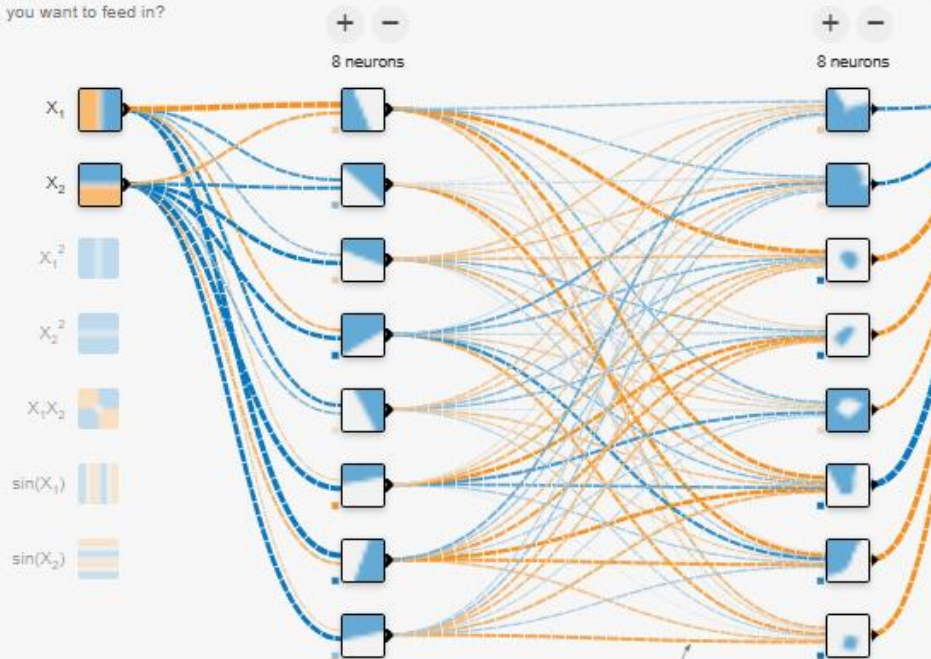
REGENERATE

FEATURES

Which properties do you want to feed in?



2 HIDDEN LAYERS

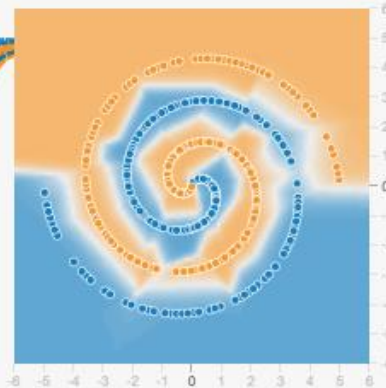


This is the output from one neuron. Hover to see it larger.

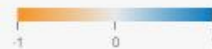
The outputs are mixed with varying weights, shown by the thickness of the lines.

OUTPUT

Test loss 0.048
Training loss 0.022



Colors shows data, neuron and weight values.



☐ Show test data

☐ Discretize output

Imagen 6

