Computação Bioinspirada

Projeto 3 - 23/11/2023

Lucas Cerutti Sergio - 11611BSI252
Rick Gomes Ricarte - 11921BSI200

Combinando algoritmos bioinspirados

Parâmetros e resultados

Após substituir a fase de treinamento do Perceptron pelo algoritmo genético, foi possível validar a eficácia dos AGs na evolução dos pesos do perceptron para a tarefa de classificação de dados.

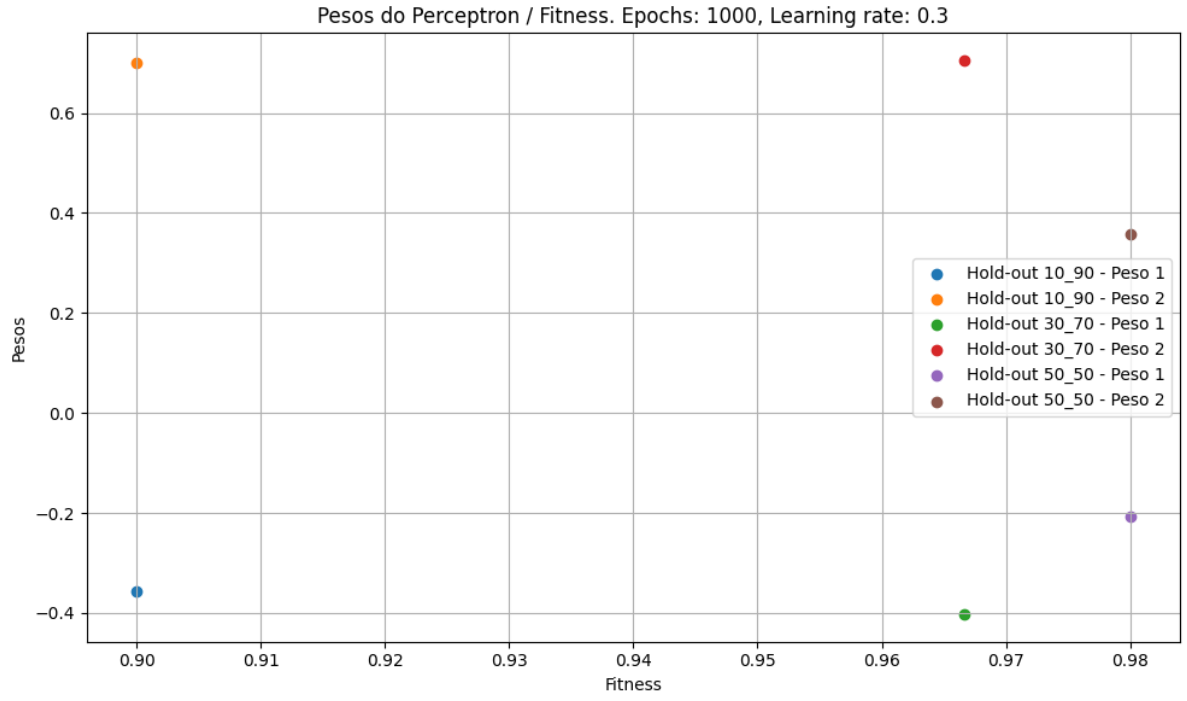
Resultados da execução

```
Epochs: 10, Learning rate: 0.1, Generation qtd: 100
Hold-out 10%: Train fitness: 0.9000, Test accuracy: 0.8778, Weights: [-0.31477875  0.634111  ]
Hold-out 30%: Train fitness: 0.9667, Test accuracy: 1.0000, Weights: [-0.10288631  0.17921065]
Hold-out 50%: Train fitness: 0.9800, Test accuracy: 0.9800, Weights: [-0.41034833  0.67037414]

Epochs: 10, Learning rate: 0.2, Generation qtd: 100
Hold-out 10%: Train fitness: 0.9000, Test accuracy: 0.9667, Weights: [-0.41303321  0.76699338]
Hold-out 30%: Train fitness: 0.9667, Test accuracy: 0.9857, Weights: [-0.23891945  0.39530932]
Hold-out 50%: Train fitness: 0.7200, Test accuracy: 0.9400, Weights: [-0.14475075  0.29721523]

Epochs: 10, Learning rate: 0.3, Generation qtd: 100
Hold-out 10%: Train fitness: 0.9000, Test accuracy: 1.0000, Weights: [-0.53987848  0.92745766]
Hold-out 30%: Train fitness: 0.9667, Test accuracy: 1.0000, Weights: [-0.26105911  0.45226841]
Hold-out 50%: Train fitness: 0.9800, Test accuracy: 0.9800, Weights: [-0.34661704  0.57227464]

Epochs: 100, Learning rate: 0.1, Generation qtd: 100
Hold-out 10%: Train fitness: 0.8000, Test accuracy: 0.9222, Weights: [-0.33900082  0.6543433 ]
Hold-out 30%: Train fitness: 0.9667, Test accuracy: 0.9857, Weights: [-0.39202222  0.64912395]
Hold-out 50%: Train fitness: 0.8600, Test accuracy: 0.9800, Weights: [-0.26689022  0.51246653]

Epochs: 100, Learning rate: 0.2, Generation qtd: 100
Hold-out 10%: Train fitness: 0.9000, Test accuracy: 0.8778, Weights: [-0.31419866  0.63577728]
Hold-out 30%: Train fitness: 0.9667, Test accuracy: 1.0000, Weights: [-0.08353525  0.14376877]
Hold-out 50%: Train fitness: 0.9800, Test accuracy: 1.0000, Weights: [-0.11757368  0.20737909]

Epochs: 100, Learning rate: 0.3, Generation qtd: 100
Hold-out 10%: Train fitness: 0.9000, Test accuracy: 1.0000, Weights: [-0.58684306  0.97954379]
Hold-out 30%: Train fitness: 0.9667, Test accuracy: 1.0000, Weights: [-0.32387506  0.56294615]
Hold-out 50%: Train fitness: 0.9800, Test accuracy: 1.0000, Weights: [-0.44311622  0.75992523]

Epochs: 1000, Learning rate: 0.1, Generation qtd: 100
Hold-out 10%: Train fitness: 0.9000, Test accuracy: 0.9889, Weights: [-0.36967188  0.65372454]
Hold-out 30%: Train fitness: 0.9667, Test accuracy: 1.0000, Weights: [-0.53943955  0.93478947]
Hold-out 50%: Train fitness: 0.9600, Test accuracy: 0.9800, Weights: [-0.38558004  0.62291653]

Epochs: 1000, Learning rate: 0.2, Generation qtd: 100
Hold-out 10%: Train fitness: 0.9000, Test accuracy: 0.9889, Weights: [-0.34285737  0.56345202]
Hold-out 30%: Train fitness: 0.9667, Test accuracy: 1.0000, Weights: [-0.09128222  0.15592329]
Hold-out 50%: Train fitness: 0.9800, Test accuracy: 0.9800, Weights: [-0.12090407  0.20069435]

Epochs: 1000, Learning rate: 0.3, Generation qtd: 100
Hold-out 10%: Train fitness: 0.9000, Test accuracy: 0.8889, Weights: [-0.35559532  0.7010237 ]
Hold-out 30%: Train fitness: 0.9667, Test accuracy: 1.0000, Weights: [-0.40214899  0.70465479]
Hold-out 50%: Train fitness: 0.9800, Test accuracy: 1.0000, Weights: [-0.20681979  0.35801381]
```

Após executar o algoritmo, é possível ver que em todas as combinações de quantidade de epochs e learning rate (com exceção de 10 epochs e LR = 0.2) o pior fitness se encontra no Hold-out 10-90.

Em contrapartida, os outros hold-outs (30-70 e 50-50) dividem os melhores valores de fitness ao longo das combinações, além das maiores acurácias.

Pesos / Fitness por Hold-out



Em algumas combinações os pesos que obtiveram altos valores fitness são muito distantes um do outro. Já em outras combinações, seus valores são mais próximos.