

Combinando algoritmos bioinspirados.

Murielly Oliveira Nascimento - 11921BSI222
Matheus Costa Monteiro - 12111BSI1281
Thales Atheniel Farias de Godoi - 11511BSI232
Paulo Victor da Silva - 11521BSI219

20 de novembro, 2023

1 Introdução

O objetivo deste trabalho foi a combinação de duas técnicas — Rede Neural Perceptron e Algoritmos Genéticos — para a classificação de espécies da base de dados Iris. A primeira foi proposta por ([ROSENBLATT, 1958](#)) que idealizou um sistema eletrônico que aprenderia a reconhecer similaridades ou identidades entre padrões óticos, elétricos, ou sonoros de uma maneira que poderia ser aproximadamente análoga aos processos de percepção do cérebro. Tal sistema dependeria de princípios probabilísticos e ganharia confiança a partir de propriedades de medidas estatísticas obtidas de uma grande população de elementos. O sistema que seguisse esses princípios se chamaria *perceptron*. Em poucas palavras, o *perceptron* é uma rede neural com uma camada.

Os valores de entrada de um *perceptron* são coletivamente chamados de vetor de entrada. Analogamente, os valores de peso são chamados de vetor de pesos, os quais, inicialmente, são inicializados aleatoriamente; é durante o treinamento do *perceptron* que os pesos ideais são aprendidos pela rede neural ([GOODFELLOW; BENGIO; COURVILLE, 2016](#)).

O Algoritmo Genético (AG), por sua vez, foi desenvolvido por ([HOLLAND, 1975](#)) e seus alunos na Universidade de Michigan em Ann Arbo. São inspirados no princípio Darwiniano da evolução das espécies e na genética ([GOLDBERG, 1989](#)). São aplicados, principalmente, na busca de soluções ótimas para problemas combinatórios, cujas técnicas tradicionais são ineficientes. Um AG, é caracterizado, em sua maioria, pelos seguintes passos: dado uma população de indivíduos (i.e. um conjunto de soluções) com uma estrutura que represente seu cromossomo, pressões do ambiente desencadeiam a seleção natural em que indivíduos mais aptos são privilegiados (nesse caso, as soluções mais adequadas, medidas por um valor chamado usualmente de *fitness*) que constituirão uma nova população. O processo se repete até que um número de iterações seja atingido ou a solução mais qualificada, seja encontrada. O Algoritmo 1 ilustra esse processo.

Portanto, neste trabalho o Algoritmo Genético é usado para otimização dos pesos da rede Perceptron — os pesos, são, os cromossomos dos indivíduos da população do AG

Algoritmo 1 Pseudocódigo do Algoritmo Evolutivo.

Gerar população inicial.

Avaliar população segundo função objetivo.

Enquanto número de gerações não atingido.

 Selecionar indivíduos para reprodução.

 Realizar cruzamento entre os indivíduos.

 Realizar mutação em alguns indivíduos.

 Inserir filhos na população.

 Avaliar população segundo função objetivo.

— fazendo a etapa de treinamento de uma rede Perceptron, enquanto esta faz a avaliação (previsão) dos dados.

2 Implementação

Neste trabalho a implementação do algoritmo foi feita na linguagem *python* no sistema operacional Windows 11 de um *notebook* Acer i3-1005G1 com 8 GB de memória RAM. O indivíduo no Algoritmo Genético é definido como uma classe contendo os atributos cromossomo (vetor de pesos); *fitness* (erro quadrático médio); e viés.

A população de soluções é inicializada aleatoriamente, com o *fitness* calculado como a média da somatória dos erros quadráticos, os quais são calculados um a um como a diferença entre o valor previsto pela classe de teste e o valor obtido pelo *perceptron*. Em seguida, a seleção elitista é aplicada a população a uma taxa p , ordenando-a e mantendo os indivíduos mais adaptados nas posições iniciais do vetor, enquanto os demais sofrem a ação dos operadores de reprodução (recombinação e mutação).

Os indivíduos pai e mãe de um novo indivíduo são selecionados pelo torneio — em que são realizados torneios entre N indivíduos aleatórios e o vencedor é aquele com melhor *fitness* —, a recombinação uniforme — a qual calcula a probabilidade de cada gene ser trocado entre os pais — é aplicada ao pai e mãe. O filho resultante dessa operação sofre a ação da mutação 1-ponto — que modifica aleatoriamente um gene de um cromossomo. Por fim, o *fitness* do novo indivíduo é calculado usando a etapa de treinamento da rede *perceptron*. Este processo é repetido um certo número de gerações. O vetor peso resultante é usado na classificação dos dados pela rede *perceptron*.

3 Experimentos

Neste trabalho, a base de dados Iris é usada para os testes. A mesma contém informações sobre três variedades da planta iris: *Setosa*, *Versicolour* e *Virginica*. As características que as separam são: comprimento da sépala, largura da sépala, comprimento da pétala e largura da pétala.

Na Tabela 1 são descritos os testes para a classificação de 100 instâncias da base de dados Iris, com as espécies *Setosa* e *Versicolour*. O ponto de partida é: número de iterações ou *epochs* (300), taxa de aprendizado ou *learning rate* (1%) e conjunto de teste (30%), o

que significa que 70% dos dados foram usados para treinar o *perceptron*. Nos demais testes os parâmetros são alterados com o intuito de aumentar a acurácia — proximidade entre o valor obtido experimentalmente e o valor verdadeiro na medição de uma grandeza física (OXFORD..., 2023) — do algoritmo. A acurácia obtida e o *fitness* do melhor indivíduo são descritos.

Observa-se que a configuração, número de iterações ou *epochs* (50), taxa de aprendizado ou *learning rate* (1%) e conjunto de teste (30%) apresentou a melhor acurácia (1.0). Embora o número de *epochs* 300 apresente um *fitness* melhor (0.18), a acurácia ideal (1.0) é obtida com um número menor de gerações (50), sendo, portanto, mais eficiente.

Tabela 1 – População (60); número de gerações (100); mutação dos vizinhos (15%); elitismo (20%); torneio (3).

Medidas de Avaliação.				
Iterações	Taxa de Aprendizado	Conjunto de Teste	<i>Fitness</i>	Acurácia.
50	1%	30%	0.019	1.0
100	1%	30%	0.019	1.0
300	1%	30%	0.018	1.0

Na Tabela 2 são avaliados os testes com o número de gerações menores (60) do Algoritmo Genético. Considerando que o número de *epochs* (50) apresentou bons resultados nos testes anteriores, este parâmetro é mantido. Observa-se que o aumento da taxa de aprendizado não afetou os resultados do algoritmo.

Tabela 2 – População (60); número de gerações (60); mutação dos vizinhos (15%); elitismo (20%); torneio (3).

Medidas de Avaliação.				
Iterações	Taxa de Aprendizado	Conjunto de Teste	<i>Fitness</i>	Acurácia.
50	1%	30%	0.019	1.0
50	2%	30%	0.019	1.0
50	3%	30%	0.019	1.0

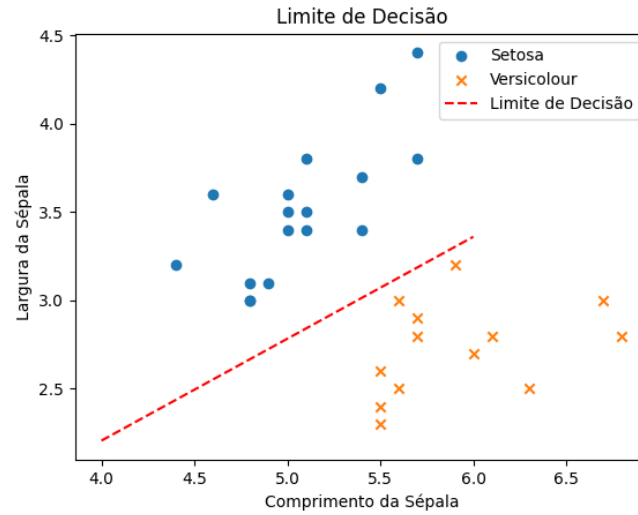
Na Tabela 3 são avaliados os testes com o número de indivíduos na população menor (30) do Algoritmo Genético. O tamanho do conjunto de teste foi alterado enquanto os demais parâmetros do *perceptron* permaneceram os mesmos. Observou-se que quanto maior o conjunto de teste maior o erro quadrático médio, levando a menor acurácia. Com um conjunto de teste 70% a acurácia obtida é 0.98.

Tabela 3 – População (30); número de gerações (60); mutação dos vizinhos (15%); elitismo (20%); torneio (3).

Medidas de Avaliação.				
Iterações	Taxa de Aprendizado	Conjunto de Teste	<i>Fitness</i>	Acurácia.
50	1%	30%	0.019	1.0
50	1%	50%	0.021	0.98
50	1%	70%	0.036	0.98

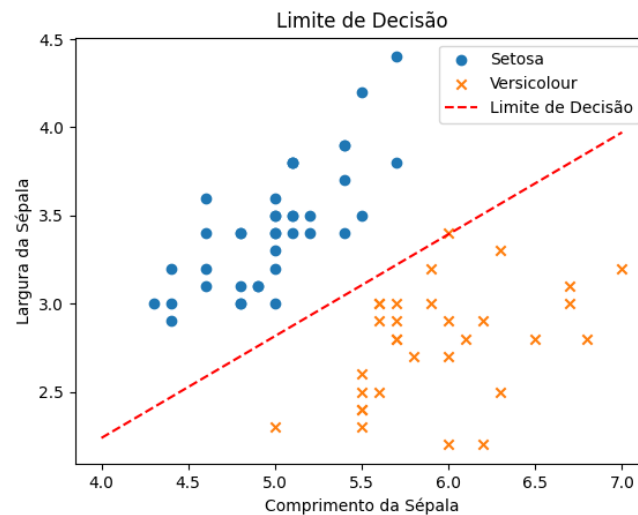
A Figura 1 ilustra o Limite de Decisão com a configuração da Tabela 3 com um conjunto de teste 30%. Nela a linha que desenha o limite de decisão ficou incompleta, isso se deve ao conjunto de teste menor.

Figura 1 – Limite de Decisão.



A Figura 2 ilustra o Limite de Decisão com a configuração da Tabela 3 com um conjunto de teste 70%. Nela a linha que desenha o limite de decisão ficou completa, isso se deve ao conjunto de teste maior, permitindo a classificação de um número maior de instâncias da Base Iris.

Figura 2 – Limite de Decisão.



Conclui-se, a partir dos resultados, que o uso do AG como abordagem na etapa de treinamento da Rede *Perceptron* é um método eficiente, pelo qual foi possível obter a acurácia 1.0 na maioria dos testes.

Referências

GOLDBERG, D. E. *Genetic algorithms in search, optimization, and machine learning*. New York, NY: Addison-Wesley, 1989. Citado na página 1.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. *Deep Learning*. [S.l.]: MIT Press, 2016. <<http://www.deeplearningbook.org>>. Citado na página 1.

HOLLAND, J. H. *Adaptation in Natural and Artificial Systems*. MA, USA: MIT Press, 1975. Citado na página 1.

OXFORD Dicionário Português. 2023. <<https://languages.oup.com/google-dictionary-pt/>>. Accessed on 19/10/2023. Citado na página 3.

ROSENBLATT, F. *The Perceptron*. Artigo — Cornell Aeronautical Laboratory, 1958. Citado na página 1.