

Aluno: Frederico Dias Castro - 12021BSI246

Aluno: Victor Silva Santos - 11811BSI218

Aluno: Matheus Lopes Ciccotti - 11811BSI275

Aluno: Gabriel Aparecido Oliveira Nunes – 11811BSI274

Computação Bioinspirada – Terceira Atividade Prática

Na atividade houve a troca do modelo de aprendizagem de rede neural (Segunda Atividade Prática) para o modelo de algoritmo genético e por consequência todo o código teve de ser reescrito para atender a demanda da Terceira Atividade Prática.

Na reescrita do código foram feitas as seguintes mudanças:

- Retirada do modelo de aprendizagem de rede neural.
- Implementação da estrutura genética contendo uma população com N indivíduos.
- Implementação do cálculo de fitness para a população de pesos do algoritmo genético utilizando a precisão como métrica.
- Implementação de algoritmo de seleção de indivíduos pais através do algoritmo de seleção por torneio.
- Implementação de algoritmo de cruzamento entre indivíduos escolhidos por meio de seleção por torneio.
- Implementação de algoritmo de mutação dos indivíduos e seus pesos.
- Implementação de algoritmo para retirar da população indivíduos com fitness baixo e trocá-los por indivíduos com fitness alto, estes últimos gerados do cruzamento de pais selecionados através da seleção por torneio.
- Reaproveitamento de parte da implementação do Perceptron da Segunda Atividade Prática para prever o atributo classe das flores da base Iris.Data

Para a execução e testes da atividade realizamos cinco execuções com os seguintes parâmetros:

Tamanho da População	Quantidade de Gerações	Taxa de mutação	Taxa de Erro 3 Classes
10	100	0.1	20.71
30	100	0.1	0.83
50	100	0.1	0.59
10	100	0.2	22
10	100	0.4	13.42
30	10	0.2	8.33
30	100	0.2	7.16
30	1000	0.4	5.83
50	100	0.2	6.20
50	1000	0.2	1
50	1000	0.4	<0,5

Comparando com o aprendizado realizado pela rede neural da Segunda Atividade Prática percebemos que a rede neural consegue trabalhar melhor com a base Iris.data dando resultados mais assertivos com menos variação conforme tabela abaixo:

Conjunto de Treinamento	Taxa de treinamento	Época	Taxa de erro 3 classes
10	0.1	100	2,42
30	0.1	100	0,5
50	0.1	100	0,4
10	0.2	100	1,0
10	0.2	100	0
10	0.4	100	0,71
30	0.2	10	1
30	0.2	100	0,83
30	0.4	1000	0

Conjunto de Treinamento	Taxa de treinamento	Época	Taxa de erro 3 classes
10	0.1	100	2,42
30	0.1	100	0,5
50	0.2	100	0
50	0.2	1000	0
50	0.4	1000	0,2

Conclui-se que, para valores menores de entrada nos dois algoritmos o Perceptron treinado com rede neural apresenta maior precisão do que o algoritmo genético e à medida que os valores de entrada são maiores os algoritmos começam a ter precisão aproximada com pouca diferença entre eles. Outro ponto importante é que a medida que aumentam a quantidade dos parâmetros o AG parece ficar mais lento em relação a rede neural. Para classificação da base Iris.data o Perceptron se mostrou melhor na precisão e no tempo de execução.