

Universidade Federal Rural de Pernambuco

Unidade Acadêmica de Garanhuns

Bacharelado em Ciência da Computação

Reconhecimento de Padrões II 2016.2

Atividade 1

Prof. Tiago de Carvalho

Aluno

João Antônio Chagas Nunes

## 1. Introdução

O objetivo dessa atividade foi implementar um *MultiLayer Perceptron* usando o algoritmo de atualização de pesos *Backpropagation*. Para testar a *MLP* criada utilizamos as bases *Iris* e *Wine*.

## 2. Metodologia

As bases foram divididas em 10 folders. Na base *Iris* em cada folder foram separados 5 exemplos de cada classe para o teste, o restante ficou para o treino. Já na base *Wine* cada folder ficou com apenas 4 exemplos de cada classe para o teste, o restante foi usado para o treino.

Na *MLP* criada é possível alterar parâmetros como, *taxa de aprendizagem*, *número de camadas*, *número de neurônios em cada camada* (cada camada pode ter um número diferente de neurônios) e até *número de ciclos* usados para o treinamento. Sendo assim, tivemos que achar empiricamente qual combinação de parâmetros se encaixava melhor em cada base.

## 3. Resultados

Os resultados foram bastantes satisfatórios na base *Iris*, talvez por conter poucas features. Por outro lado não conseguimos obter resultados satisfatórios na base *Wine*.

### 10-Fold Cross-Validation: MLP

Taxas de acerto por fold:

|                         |            |
|-------------------------|------------|
| - Testando fold (r=1):  | 1          |
| - Testando fold (r=2):  | 1          |
| - Testando fold (r=3):  | 1          |
| - Testando fold (r=4):  | 0.93333333 |
| - Testando fold (r=5):  | 0.86666667 |
| - Testando fold (r=6):  | 1          |
| - Testando fold (r=7):  | 0.86666667 |
| - Testando fold (r=8):  | 1          |
| - Testando fold (r=9):  | 1          |
| - Testando fold (r=10): | 1          |

|                    |       |
|--------------------|-------|
| - Média de acerto: | 97%   |
| - Desvio padrão:   | 5.67% |

*Imagem 1 – resultado obtido na base Iris.*

Na *Imagem 1* temos o resultado obtido pela *MLP* treinada para a classe *Iris*. Nessa *MLP* tínhamos 3 camadas cada uma com 3 neurônios, usamos 1000 ciclos de treinamento com uma taxa de aprendizagem de 0,1.

#### 10-Fold Cross-Validation: MLP

Taxas de acerto por fold:

|                         |          |
|-------------------------|----------|
| - Testando fold (r=1):  | 0.583333 |
| - Testando fold (r=2):  | 0.41666  |
| - Testando fold (r=3):  | 0.66666  |
| - Testando fold (r=4):  | 0.66666  |
| - Testando fold (r=5):  | 0.583333 |
| - Testando fold (r=6):  | 0.5      |
| - Testando fold (r=7):  | 0.5      |
| - Testando fold (r=8):  | 0.583333 |
| - Testando fold (r=9):  | 0.666666 |
| - Testando fold (r=10): | 0.5      |
| - Média de acerto:      | 56.67%   |
| - Desvio padrão:        | 8.61%    |

*Imagem 2 – resultado obtido na base Wine*

Na *Imagem2* temos o resultado obtido pela *MLP* treinada para a classe *Wine*. Nessa *MLP* tínhamos 3 camadas, a primeira camada tinha 18 neurônios, a segunda tinha 10 e a terceira apenas 3. Usamos 10000 ciclos de treinamento com uma taxa de aprendizagem de 0.001.

## 4. Conclusão

Conclui-se que o poder da *MLP* é bem superior ao do *perceptron*, porém sua complexidade é bem maior também. Sendo assim, nem sempre é fácil achar os parâmetros certos para determinadas bases. Como vimos nessa atividade, apesar de todo o esforço dedicado não conseguimos achar os parâmetros certos para obter uma boa taxa de acerto na base *Wine*.