

Comparação entre classificação usando Perceptrons e MLP

¹Hugo Fernandes

¹Depto de Computação e Tecnologia - Universidade Federal do Rio Grande do Norte (UFRN)
Caixa Postal: 59.300-000 - Caicó - RN - Brazil

¹ hugo.medeiros.fernandes@gmail.com

Abstract. This report seek to compare two classification techniques, in a database of binary vowel characteristics.

Resumo. Este relatório busca comparar duas técnicas de classificação, numa base de dados de características binárias de vogais.

Keywords: Perceptrons, MLP

1. Base de Dados

Os conjuntos de dados que podem ser encontrados nos arquivos, 'vogais1.csv', 'vogais2.csv' e 'vogais3.csv', consiste em valores que representam características das 5 vogais do alfabeto (A, E, I, O, U). Cada um dos arquivos possuem 1000 linhas, onde cada linha representa um exemplo de vogal que segue a seguinte estrutura:

A,0,1,0,1,0,1,1,1,1,0,1,1,0,1

Sendo primeiro valor, a vogal referência, já os demais valores, consiste numa matriz binária 5x3, que representa as característica dessa vogal referência.

Foi desenvolvido um script, que pode ser encontrado no arquivo 'script_dataset.py', que gera os 3 conjuntos de dados, com parâmetros ajustados para cada exemplo.

O primeiro conjunto de dados ('vogais1.csv'), possui 1000 exemplos de vogais, geradas nas mesmas proporções, ou seja, 'A' 20%, 'E' 20%, 'I' 20%, 'O' 20%, 'U' 20%, sendo que a cada vogal inserida, ela tem uma taxa de 10% de ocorrer um ruído, em algum dos bits

que determinam suas características. Esse ruído, trata da inversão de algum dos bits da matriz 5x3. A escolha de qual bit será invertido, é feita de forma aleatória.

No segundo conjunto de dados ('vogais2.csv'), a única mudança que se tem para o primeiro, é a quantidade de bits invertidos, agora sendo 2 bits invertidos por ruído.

Por fim, no terceiro conjunto de dados ('vogais3.csv'), como no segundo, a única mudança é que agora serão 3 bits invertidos por ruído.

2. As Estratégias

As duas técnicas usadas executam as mesmas bases de dados, e retornam uma porcentagem de acerto na tarefa de classificação, essa porcentagem dar-se o nome de Acurácia, e é uma relação entre o resultado retornado pela estratégia de classificação e seu valor real esperado. Para se alcançar uma acurácia média do resultado dos algoritmos, cada caso foi executado por 10 vezes.

Outra característica das duas estratégias, é que ambas passam por uma etapa de treinamento supervisionado, antes de executar a classificação. Para essa etapa, foi utilizado 75% das bases de dados, e os 25% restantes para os testes. Os dados que serão usados para treinamento e testes, são coletados de forma aleatória, mas sempre respeitando as proporções acima citadas.

Tratando agora das estratégias separadamente, as duas se utilizam do conceito de perceptrons, sendo que o algoritmo ('vogais_perceptron.py'), foi construído apenas com 5 perceptrons, para assim classificar as 5 vogais do problema. Já o algoritmo ('vogais_mlp.py'), é implementado através da biblioteca “Sklearn”, e trabalha com o conceito de múltiplas camadas de perceptrons. Uma particularidade deste último, é que se pode configurar sua arquitetura de múltiplas camadas, sendo que por possuir essa característica, foi preferido executá-lo algumas vezes a mais, para se coletar sua acurácia com arquiteturas diferentes. As

arquiteturas aplicadas foram 5 camadas, 10 camadas e 100 camadas, para cada uma das bases de dados.

3. Resultados

Os resultados podem ser vistos na tabela 1.

	vogais1.csv (1 bit de ruído)	vogais2.csv (2 bits de ruído)	vogais3.csv (3 bits de ruído)
vogais_perceptron.py	0.9416	0.8456000000000001	0.718
vogais_mlp.py (5 camadas)	0.9827999999999999	0.8172000000000003	0.7656
vogais_mlp.py (10 camadas)	0.9992000000000001	0.9655999999999999	0.8916000000000002
vogais_mlp.py (100 camadas)	1.0	0.9992000000000001	0.9964000000000001

Tabela 1. Resultado da execução dos algoritmos

4. Considerações Finais

Três conclusões podem ser verificadas nesse experimento, a primeira é que como previsto, a acurácia média tendeu a cair quando o ruído nas bases de dados foi aumentando. A segunda conclusão, ficou em relação a vantagem na maioria dos casos do MLP sobre os 5 perceptrons, para esse problema em específico. E finalmente a terceira conclusão, está na leve melhora da acurácia do MLP, ao se aumentar o número de camadas, visto que, para um bit de ruído, o algoritmo conseguiu classificar 100% dos casos de testes, feito não alcançado em nenhum outro cenário.