

Aprendizagem Automática Avançada

Trabalho Laboratorial N° 2

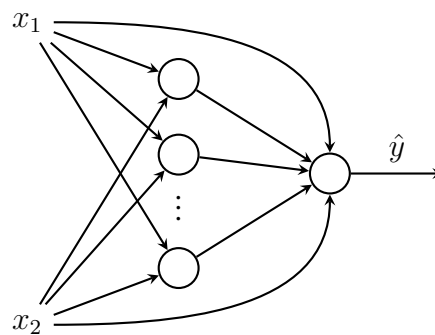
Redes MLP

2º Semestre de 2021/2022

Objetivos do trabalho:

Implementar, treinar e testar redes neurais do tipo Perceptrão Multi-Camada (MLP). O relatório e código implementados deve ser realizado num Jupyter Notebook com o nome `A#####A#####_T2.ipynb` (A##### corresponde ao número de aluno).

1. Sem recorrer à biblioteca TensorFlow/Keras, implemente a seguinte rede MLP de três camadas, para a tarefa de classificação binária de dados bi-dimensionais (problema XOR):



Tenha em conta os seguintes pontos:

- No diagrama acima, cada círculo representa um perceptrão, e como tal realiza um somatório das entradas multiplicadas pelos respetivos pesos mais um termo de polarização, seguido de uma função de ativação (ver slide 4/21 de “An Introduction to Artificial Neural Networks”).
 - Use a tangente hiperbólica para as funções de ativação desta rede.
 - O número de unidades escondidas deverá ser um parâmetro que pode ser alterado no código.
 - Use os dados do ficheiro `pickle xorData.p`. Este ficheiro contém um dicionário com 1000 pontos bi-dimensionais relativos ao problema XOR (chave `data`) e as respetivas classes (chave `target` - array de 0s e 1s).
- (a) Implemente um algoritmo de treino da rede. Neste deverá ser possível escolher o valor do passo de adaptação, η , o termo de momento, α , bem como o número de iterações de adaptação.
 - (b) Treine uma rede com uma unidade escondida. Execute várias vezes o treino de modo a escolher os valores que achar apropriados para η , α e para o número de iterações. Faça um gráfico da evolução do erro ao longo do treino com os parâmetros escolhidos. Visualize o conjunto de treino e os erros obtidos neste conjunto. Faça

um gráfico da função de saída da rede, usando o comando `plot_wireframe`, para uma grelha de 50×50 compreendida no intervalo $[-1, 2]$.

- (c) Repita a alínea 1.b) para uma rede com 10 unidades escondidas.
- (d) Repita a alínea 1.b) para uma rede com 50 unidades escondidas.

2. Pretende-se implementar, treinar e testar várias redes MLP com a base de dados CIFAR-10.

- (a) Treine e avalie uma rede MLP com 10 camadas escondidas e 100 unidade por camada. Use a função `ReLU` para as ativações das camadas escondidas, a função `softmax` para a camada de saída, e a otimização `Nadam`.
- (b) Treine uma rede MLP à sua escolha tendo como objetivo chegar a uma probabilidade de acerto acima dos 50% no conjunto de teste. Tenha em conta os seguintes pontos:
 - Pode pré-processar os dados (ex.: *standard scalar*, PCA)
 - Use um conjunto de validação para visualizar a evolução no treino.
 - Use *callbacks* para implementar *early stopping*.
 - Experimente usar diferentes parâmetros/métodos no treino, como camadas de *dropout*, diferentes funções de ativação, etc.

Elaboração do Relatório: Deverá ter em conta vários aspetos inerentes a um projeto de aprendizagem supervisionada tais como as metodologias de treino/teste usadas, métricas de avaliação usadas, etc. Compare os resultados da alínea 2. com os obtidos no 1º trabalho.

O Jupyter Notebook deverá estar devidamente comentado de modo a se perceber claramente as várias etapas do trabalho desenvolvido e os resultados obtidos.