



Exercícios - Perceptron Simples

As questões 1 a 3 necessitam do script python da nossa primeira aula prática. Se você perdeu, você pode acessá-la em:

https://www.youtube.com/watch?v=Zflb__rg2As

- 1) Diminua o número de épocas para 63000 e execute o script. O que você observou?
- 2) Refaça a primeira questão, porém aumentando eta para 0.6. Execute novamente o script e responda o que você observou.
- 3) Modifique agora a função de ativação pela função tangente hiperbólica ($\text{np.tanh}(V)$) e observe. O que aconteceu agora? Aproveite e tente outras funções de ativação como:

`np.sign(V)`
`np.heaviside(V, 1)`

4) A Tabela abaixo apresenta os dados referentes a um problema de classificação conhecido como *The Iris Data Set* (<https://archive.ics.uci.edu/ml/datasets/iris>); um conjunto de dados reais e experimentais a respeito da classificação de flores da espécie *Iris sp.* Todas os atributos estão em centímetros.

Comprimento da Sépala	Largura da sépala	Comprimento da Pétala	Largura da Pétala	Classe
5,1	3,5	1,4	0,2	1
4,9	3,0	1,4	0,2	1
6,1	2,9	4,7	1,4	2
6,4	3,1	5,5	1,8	3
...

- a) Quantos atributos esse problema apresenta?
- b) A superfície de decisão de um classificador como o Perceptron será de que tipo (reta, plano ou hiperplano)? Justifique.
- c) Supondo que você vá montar uma pequena rede neural Perceptron de camada simples, qual a condição para que o classificador desse tipo seja possível?



Canal Inteligência Artificial na Prática

- d) Como seria a arquitetura para a rede perceptron do item c (número de entradas, com ou sem bias, número de neurônios)?
- 5) Os Perceptrons podem aprender a reproduzir várias funções lógicas. Demonstre a implementação das funções lógicas binárias AND (E), OR (OU) e NOT (NÃO).
- 6) Os Perceptrons não conseguem aprender a reproduzir uma função lógica específica: A função OU EXCLUSIVO (XOR). Explique o motivo.
- 7) Considere duas Classes C_1 e C_2 , unidimensionais, com distribuição Gaussiana e variâncias iguais e unitárias ($\sigma = 1$). Seus valores médios são:

$$\begin{aligned}\mu_1 &= -10 \\ \mu_2 &= +10\end{aligned}$$

As duas classes são, essencialmente, linearmente separáveis. Implemente uma rede neural Perceptron de camada simples, em Python, que separe as duas classes.

Dica: Para gerar os pontos da sua base de dados, use a função gaussiana:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

Use cerca de 50 pontos gerados aleatoriamente para cada classe.

AND			OR			XOR			NOT	
Entradas		Saída	Entradas		Saída	Entradas		Saída	Entrada	Saída
X1	X2	Y	X1	X2	Y	X1	X2	Y	X1	Y
0	0	0	0	0	0	0	0	0	0	1
0	1	0	0	1	1	0	1	1	1	0
1	0	0	1	0	1	1	0	1		
1	1	1	1	1	1	1	1	0		