

**Inteligência Artificial e Sistemas Cognitivos**

Mestrado em Engenharia Informática e Multimédia

**Relatório**

*Projeto*

Rodrigo Dias – 45881

# Objetivo 1

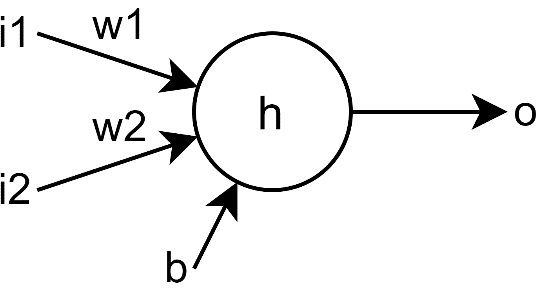
APRENDIZAGEM SUPERVISIONADA

## Realização de um protótipo com base numa plataforma de desenvolvimento de redes neuronais artificiais

### Aprendizagem da função lógica XOR

HIDDEN LAYER

A picture containing diagram

Description automatically generatedPara realizar a função lógica XOR, pensou-se em implementar uma rede neuronal constituída por 2 entradas e 1 saída. Isto é problemático porque tendo um só neurónio, não existe maneira de criar uma fronteira para distinguir as combinações do XOR pois como se pode ver na F, a expressão de saída do neurónio gera apenas uma reta. A F ilustra os valores possíveis do XOR consoante os inputs e observa-se que uma reta não consegue separar os valores lógicos de 1 (a verde) dos valores lógicos 0 (a vermelho).

Uma solução a este problema seria ter 2 retas de maneira que a área entre elas (AND) possua apenas um valor lógico e a área fora (OR) possua o outro. Para tal, é possível utilizar 2 neurónios na mesma camada escondida, entre as entradas e a saída.

#### Efeito da taxa de aprendizagem

#### Efeito da introdução de um termo de momento

#### Efeito da apresentação das amostras de treino com ordem fixa ou aleatória

#### Efeito de uma codificação binária ou polar

### Aprendizagem de padrões de imagem

Com o objetivo de concretizar uma rede neural para deteção de padrões, implementou-se um modelo com 16 entradas (para os inputs dos pixéis de padrões 4x4) e 4 saídas, já que se terá 4 padrões diferentes para detetar, ilustrados na F. Caso seja detetado o padrão A, então as saídas estarão a [1,0,0,0], caso seja detetado o B, as saídas serão [0,1,0,0] e caso não seja nenhum padrão as saídas serão [0,0,0,0].



## Aplicação de redes neuronais artificiais para resolução de um problema de escolha livre

A picture containing text, crossword puzzle

Description automatically generatedDecidiu-se fazer uma rede neural para resolver *Nonograms*, *puzzles* de lógica que consistem em preencher as células de uma matriz consoante os números indicados à esquerda das linhas e no topo das colunas, como ilustrado na F. No caso de uma linha ter os números “2 2”, indica que vão ser preenchidos dois pares de células da matriz separados por, pelo menos, uma célula em branco.

Para tal, o objetivo é colocar nas entradas da rede neural os números das linhas e colunas da matriz de forma a colocar na saída a matriz resolvida. Assumindo que se usará matrizes de 4x4, serão precisos 16 neurónios de saída. Para os neurónios de entrada, decidiu-se usar um para cada linha/coluna, caso contrário, o número de entradas vai depender do número de células preenchidas em cada linha/coluna. Imaginando que uma linha tem os números “2 1”, coloca-se no neurónio de entrada o valor 21, se tiver “1 1”, coloca-se o valor 11, se tiver “1”, coloca-se 1, etc. Dessa maneira, o número de neurónios de entrada serão 8, 4 para os números correspondentes às linhas e os outros 4 para os números das colunas.

Para treinar a rede neural, gerou-se primeiro um *array* de *arrays* únicos com 16 posições preenchidas com 1’s e 0’s aleatoriamente. Estes *arrays* de 16 posições são as matrizes quadradas transformadas de 4x4 para 1x16.