

Ejercicio 1:

Diseñe una neurona perceptrón con dos entradas binarias y una función de activación por umbral que calcule la función lógica de la coimplicación, es decir, que calcule $x_1 \leftrightarrow x_2$. Determine un conjunto de pesos adecuados para que la neurona calcule esta función (No es necesario aplicar ningún algoritmo concreto.)

Ejercicio 2:

Diseñe una neurona perceptrón con tres entradas binarias y una función de activación por umbral cuya salida de activa, si y sólo si dos o más de las entradas tienen un valor 0. Determine un conjunto de pesos adecuados para que la neurona calcule esta función (No es necesario aplicar ningún algoritmo concreto.)

Ejercicio 3:

Un banco quiere clasificar los clientes potenciales en fiables o no fiables. El banco tiene un dataset de clientes antiguos, con los siguientes atributos:

- Estado civil: {casado/a, soltero/a, divorciado/a}
- Género: {varón, mujer}
- Edad: {[18-30], [31-50], [51-65], [65+]}
- Ingresos: {[10K-25K], [26K-50K], [51K-65K], [66K-100K], [100K+]}

- a) Diseñe una red neuronal con neuronas perceptrón simples de activación umbral y de una única capa que se podría entrenar para predecir si un cliente es fiable o no.
- b) Inicializa los pesos de la red, inventa un caso de entrenamiento y reproduce cómo el algoritmo de aprendizaje realizaría modificaciones en la red con este ejemplo.

Ejercicio 4:

Una empresa de videojuegos está desarrollando un FPSG (first person shooting game). Para implementar los personajes artificiales del juego, el jefe de proyecto, ex estudiante del curso de IA en la URJC, ha pensado que podría ser interesante e innovador utilizar una red neuronal. Dicha red tendría que implementar el algoritmo de control de los personajes artificiales, usando los siguientes inputs:

- Salud: de 0 (débil) hasta 2 (fuerte)
- Tiene cuchillo: si o no
- Tiene arma: si o no
- Enemigos: número de enemigos en el campo visual

Las acciones que el personaje puede ejecutar son:

- Escapar
- Andar
- Atacar

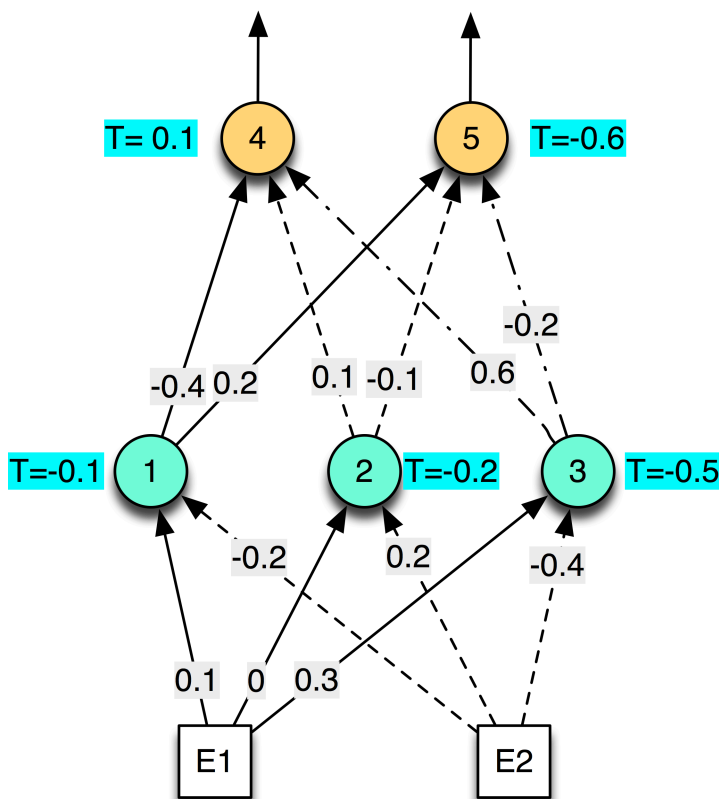
- Escondarse

- a) Diseña una red neuronal (de una única capa) que se podría entrenar para implementar el algoritmo de control de los personajes artificiales.
- b) Considera el siguiente vector de input:
 $x = [\text{Salud}=2, \text{Tiene cuchillo=no}, \text{Tiene arma=si}, \text{Enemigos}=2]$
 y la salida deseada:
 $y = \text{Atacar}.$

Inicializa los pesos y los sesgos de las neuronas de tu red de manera aleatoria en el rango $[-0.5, 0.5]$. Usando el elemento de entrenamiento definido arriba indica como cambiarían los pesos. Por simplicidad, asume que las neuronas tienen una función de activación por umbral y que se emplea $g'(x)=1$ como sustituto de la derivada de la función de activación.

Ejercicio 5:

Sea la red neuronal presentada en la figura. Supón que la función de activación de las neuronas 1,2,3,4 y 5 sea la función umbral que devuelve 1 si la suma pesada de las entradas es mayor que 0. Además, en la retropropagación de los errores emplea $g'(x)=1$ como sustituto de la derivada de la función de activación. T denota el sesgo de cada neurona.



Las entradas de una red neuronal no necesariamente tienen que tener valores binarios, también podrían ser de valores continuos en el intervalo $[0..1]$. Para neuronas con una función de activación por umbral, la salida, sin embargo, si será binaria.

Dado el siguiente elemento del conjunto de entrenamiento:

$$(x,y) = (x=[E1=0.6, E2=0.1], y=[4=0, 5=1])$$

donde el valor la entrada 1 es 0.6, el valor de la entrada 2 es 0.1, la salida de la neurona 4 es 0 y la salida de la neurona 5 es 1.

Sea 0.1 la constante de aprendizaje, aplica el algoritmo de retropropagación para actualizar los pesos de la red con este elemento del conjunto de entrenamiento e indicando como cambian los pesos.