

Profesor responsable: Patricio García Báez

Fecha tope de corrección: 8 de noviembre

Objetivo: Dominio del SNNS y capacidad representación del conocimiento en las RN

Utilizando el neurosimulador SNNS se pretende diseñar y entrenar varias redes neuronales feed-forward capaces de aprender a **transformar valores** numéricos entre **diferentes tipos de codificaciones** y la codificación utilizada para los **visualizadores de siete segmentos**, tan comunes en dispositivos electrónicos. Para ello habrán de diseñarse los ficheros correspondientes de patrones de entrenamiento así como los que almacenen las diferentes redes en sí.

Utilizaremos los **números del 0 al 9**, los cuales los codificaremos de cuatro diferentes maneras:

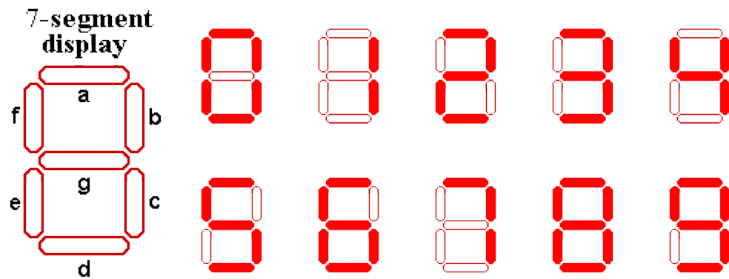


Figura 1: Codificación para visualizadores de 7 segmentos

1. Utilizando **una entrada o neurona de salida** asociada a **cada uno de estos números** (un total de **10 entradas o salidas**), en dicha codificación ha de estar activo un único elemento, que indica el número a representar.
2. En **modo binario**, esto es, mediante **cuatro elementos**, representando respectivamente las diferentes potencias de dos de la codificación binaria del número a representar. El elemento que este activo indica con un uno en la codificación binaria y el inactivo con un

cero.

3. En **modo decimal**, con **un solo elemento**, de forma que la magnitud de éste indique el valor que se quiere representar, una única unidad por tanto.
4. En el **formato utilizado por los visualizadores** de siete segmentos ([http://en.wikipedia.org/wiki/Seven-segment\\_display](http://en.wikipedia.org/wiki/Seven-segment_display)), representado por siete neuronas correspondientes a los valores (on/off = 1/0) de los segmentos a, b, c, d, e, f, g (ver figura 1).

Se pretende por tanto generar y entrenar **siete tipos de codificadores diferentes**:

1. Conversor de código 1 al 4
2. Conversor de código 4 al 1
3. Conversor de código 2 al 4
4. Conversor de código 4 al 2
5. Conversor de código 3 al 4
6. Conversor de código 4 al 3
7. Conversor de código 4 al 4 (con una o más neuronas ocultas)

Ha de tratarse de generar arquitecturas neuronales que utilicen el número mínimo posible de neuronas que resuelvan bien los problemas, y ratios de aprendizaje que la hagan converger en el número mínimo de ciclos posibles.

Con los resultados de los entrenamientos anteriores se confeccionará una **tabla de resultados** en la que se indique **tipo de patrón**, **mejor arquitectura de RN** encontrada, **mejor error** obtenido (tanto su valor **SSE** como en **porcentaje de errores**), valor del **ratio de aprendizaje óptimo** y **número de ciclos** empleado para el aprendizaje, así como las **observaciones** que se quieran hacer constar. Dicha tabla (*formulario\_practica1.odt*) será la que finalmente se entregue como justificación de la práctica.