Redes Neuronales

Las redes neuronales artificiales básicamente intentan simular los procesos de cómputo que se realizan en el cerebro.

El cerebro humano esta compuesto por células de diferentes tipos llamadas neuronas, en un número de 10^{11} (cien mil millones) aproximadamente.

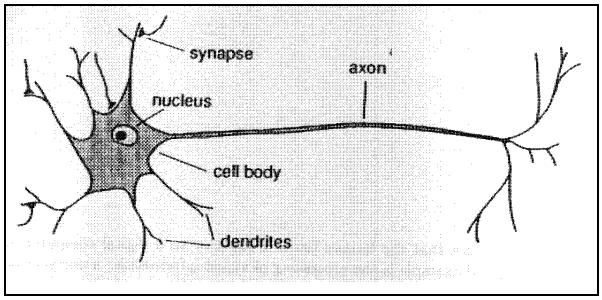

Figura 6.1 Una neurona natural

Dichas neuronas se componen, como se observa en la Figura 6.1 de:

- Un cuerpo celular o soma esférico, de 5 a 100 micras de diámetro, en el que se encuentra el núcleo de la célula.
- Varias dendritas, que son ramificaciones a través de las cuales las neuronas reciben estímulos electroquímicos procedentes de otras neuronas.
- Un axón, que es una ramificación a través de la cual la neurona envía estímulos a otras neuronas. Dicho axón puede dividirse a su vez en varios miles de ramas.

Cada una de las ramas en la que se divide un axón normalmente se unirá con una dendrita perteneciente a otra célula, a dicho enlace se le llama unión sináptica o sinapsis, la cual puede tener distintos grados de fuerza según la resistencia al paso de los estímulos o señales a través de ella sea mayor o menor.

El proceso que realiza básicamente una neurona es sumar las fuerzas de los estímulos que le entran a través de cada una de sus sinapsis y si dicha suma supera un cierto umbral, enviar un estímulo a través de su axón, se dice entonces que la neurona ha disparado.

Así pues, lo que determinará que un grupo de neuronas actúe de una manera determinada serán las entradas hacia esas neuronas, la fuerza de las conexiones sinápticas y la estructura que tengan dichas conexiones.

Neurona artificial

Una primera aproximación a una neurona podría ser el modelo de la Figura 6.2. En él existen unas entradas, x_1, x_2, ..., x_N a las que se les asocia un determinado peso, w_1, w_2,..., w_N, lo cual simularía las dendritas y su fuerza sináptica correspondiente.

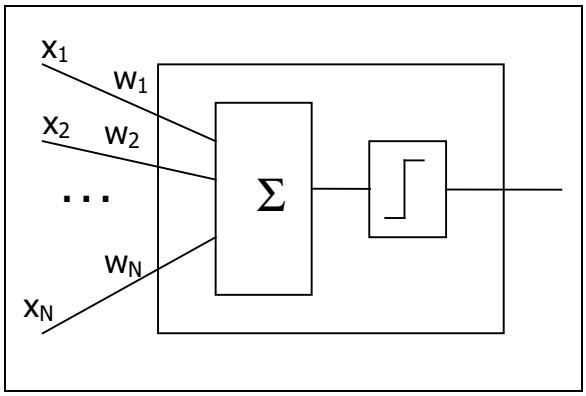

Figura 6.2 Neurona artificial

A continuación para simular el soma existe un bloque que se encargara de sumar el producto de las entradas por su peso correspondiente, y a su resultado se le aplicará una función que simule el umbral de la célula. Dicha función de umbral puede ser de diferentes tipos. La salida de la célula, que será el valor resultante, podrá ramificarse y conectarse a las entradas de otras neuronas.

Inspiradas en el cerebro humano, se han creado estructuras artificiales (llamadas redes neuronales) formadas por un conjunto (grande) de unidades elementales (a las que se puede llamar neuronas artificiales o formales), conectadas masivamente que, sin ser muy complejas, son capaces de exhibir ciertos comportamientos que muestran sorprendentes analogías con el comportamiento del cerebro humano. algunos de estos comportamientos son los siguientes:

- Capacidad de aprendizaje. Las redes neuronales artificiales son capaces de modificar su comportamiento en respuesta a las entradas que se le suministran. Presentándole una serie de entradas y, eventualmente, las salidas deseadas, la red es capaz de modificar su estructura para producir respuestas consistentes. La forma particular en que la red modifica su estructura en función de los datos recibidos se conoce con el nombre de regla de aprendizaje.
- Capacidad de generalización. Una vez "enseñada" la red muestra un comportamiento en cierta medida "insensible" a pequeñas variaciones en las entradas; esto es, es capaz de reconocer patrones de entrada aunque se hayan introducido en ellos pequeñas variaciones respecto a los patrones originales que se le enseñaron. Esta posibilidad de reconocer a pesar de la presencia de ruidos o distorsiones es vital si se desea interaccionar con el mundo real.
- Capacidad de abstracción. Algunas redes neuronales son capaces de abstraer características del conjunto de entradas que se les muestra. Por ejemplo, mostrándole una serie de versiones distorsionadas de la letra A, tras una fase de aprendizaje, la red es capaz de producir una A "perfecta", que no le ha sido mostrada nunca.
- Memoria asociativa. Tanto la capacidad de generalización como la de abstracción están intimamente ligadas con el concepto de memoria asociativa. Las redes neuronales son capaces de almacenar patrones (recuerdos) de forma que para recuperarlos basta con mostrar a la red una información parcial del recuerdo almacenado. Esta capacidad es también característica de la mente humana, donde la recuperación de recuerdos funciona por asociación, y no a través de la asignación de un lugar fijo de almacenamiento con una dirección asignada, como es el caso de los ordenadores.

Aplicaciones

Algunas de las aplicaciones más usuales que tienen las redes neuronales artificiales hoy en día son:

- Resolución de problemas de optimización combinatoria, como pueden ser el TSP (viajante de comercio) o la bipartición de grafos.
- Reconstrucción y tratamiento de señales. P. ej. aclarado de imágenes, filtros de señales, etc.
- Compresión de datos.
- Resolución de problemas de percepción, como reconocimiento de imágenes o del habla, vocalización de textos, etc.

La neurona de McCulloch y Pitts

El modelo anterior de neurona puede simplificarse en forma de máquinas simples con sólo dos posibles estados, que se interconectan entre sí. A partir de dichas máquinas podemos construir el equivalente a cualquier otra máquina de estado finito. El objetivo que se consigue es construir máquinas de todas clases partiendo de estos elementos simples.

Utilizaremos elementos desarrollados por McCulloch y Pitts en 1943 que modelizan aspectos del funcionamiento del cerebro. Esto no quiere decir que sean un modelo exacto, en realidad las neuronas reales son mucho más complejas. Fueron diseñados para analizar las situaciones lógicas que se producen en cualquier proceso discreto.

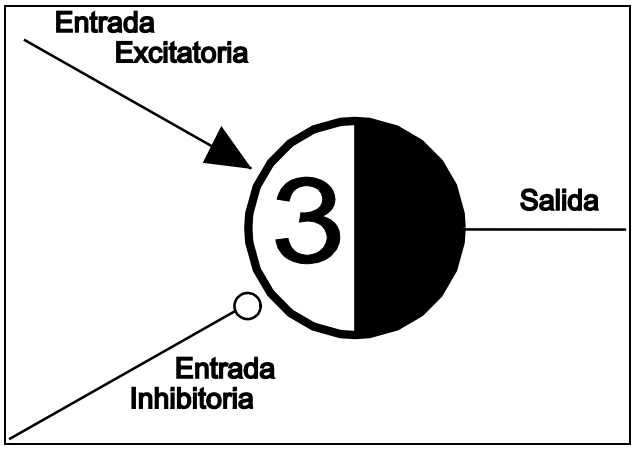

Figura 6.3 Modelo de neurona de McCulloch-Pitts.

Denominaremos Red Neuronal a todo sistema formado por combinación de dichas unidades elementales. Cada neurona se representa como se muestra en la Figura 6.3. Tiene una sola salida, que una vez abandonado el cuerpo de la neurona se puede ramificar, y dichas ramas pueden servir de entrada a otras neuronas (o a ella misma en forma de realimentación). Cada neurona puede tener cualquier número de terminales de entrada, y dentro de dichos terminales se distinguen dos tipos:

- Las entradas excitatorias.
- Las entradas inhibitorias.

representados como muestra la figura.

Cada neurona es capaz de variar su estado interno y el instante de tiempo en que es posible dicho cambio está sincronizado entre todas las neuronas de la red. Los dos estados en que puede permanecer una neurona son encendido y apagado. También se pueden denominar neurona activada y desactivada, disparada o inactiva según que el estado en que esté sea encendido o apagado.

Para cada estado hay una determinada señal de salida. Se puede suponer que dicha señal será un pulso cuando el estado es encendido y una ausencia de pulso cuando sea apagado. Los estados cambiarán en función de los pulsos recibidos en las entradas, y el tipo de entradas de que se trate. Existe un número representado dentro de la célula, al que denominamos umbral, que también determinará las propiedades de cambio de estado. Las reglas que determinan el cambio de estado desde el instante de tiempo t al t+1 son las que se muestran en la Figura 6.4.

```
Si alguna de las entradas inhibitorias está activa, entonces
        La célula estará apagada en el instante t+1.
si no
        Sea N el número de entradas excitatorias activas
        Si N >= umbral entonces
                La célula estará encendida en el instante t+1.
        si no
                La célula estará apagada en el instante t+1.
```

Figura 6.4 Las reglas que rigen el cambio de estado

Nótese que el nuevo estado no depende directamente de su estado anterior, lo cual ya marca diferencias con la neuronas reales, ya que éstas últimas una vez que se activan, no pueden volver a

hacerlo hasta pasado un intervalo de tiempo llamado período refractario. Nótese también que la inhibición es absoluta: una sola entrada inhibitoria activa puede bloquear la respuesta de múltiples entradas excitatorias.

Existen otros modelos en que se produce activación de la neurona si la diferencia entre entradas excitatorias activas e inactivas es mayor o igual al umbral. Esto es equivalente a suponer que las entradas inhibitorias aumentan el umbral.

Algunas redes neuronales sencillas

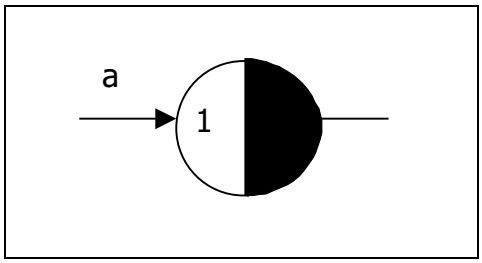

Figura 6.5 Una red que produce un retardo unitario

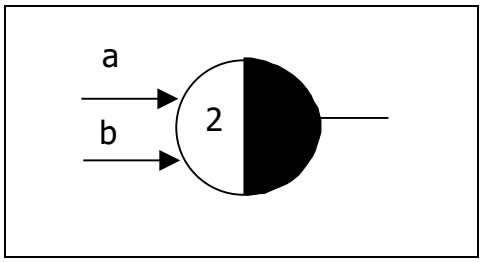

Figura 6.6 La función "and"

La Figura 6.5 representa una red neuronal (formada por una sola neurona) que produce un retardo unitario en la señal que recibe como entrada: si la entrada está activa en un instante *t*, la salida lo estará en el instante *t+1*.

La neurona de la Figura 6.6 realiza la función "y" lógica: la salida sólo disparará cuando exactamente las dos entradas *a* y *b* hayan estado activas simultáneamente en el instante de tiempo inmediatamente anterior.

Como vemos, utilizando redes simples podemos construir funciones lógicas que a veces pueden llegar a ser complejas. Consideremos un ejemplo más: la neurona de la Figura 6.7 realiza la función lógica (a AND b) AND (NOT c) AND (NOT d) puesto que para que la neurona dispare es necesario que las entradas *a* y *b* estén activas y que no lo esté ni *c* ni *d*.

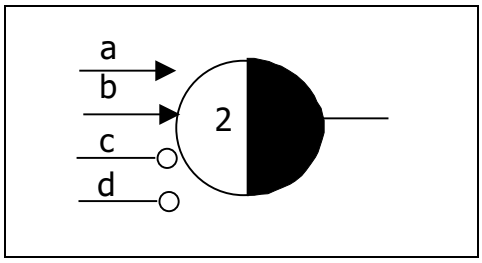

Figura 6.7 Una función lógica

Memoria

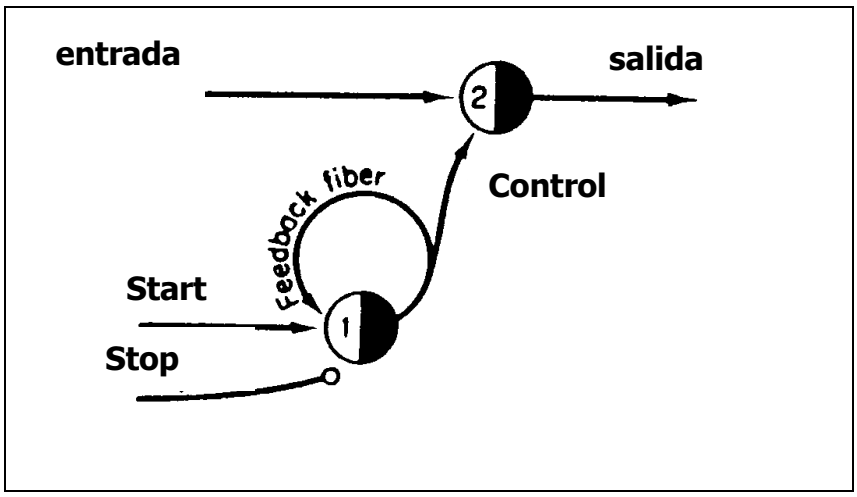

Figura 6.8 Control de flujo con memoria

La red que se presenta en la Figura 6.8 controla el flujo de pulsos desde su entrada hacia salida mediante la neurona de la parte inferior, que está realimentada. Esa neurona de la parte inferior tiene dos entradas *Start* y *Stop* que sirven para iniciar y detener el flujo de la entrada a la salida. Una vez que la entrada *Start* ha sido disparada una vez, la neurona permanece siempre disparando gracias a la fibra realimentada (*Feedback fiber*) hasta que eventualmente llegue un pulso inhibitorio por la entrada *Stop*. Supuesto que la señal de comienzo ha sido disparada y que por tanto la neurona inferior está disparando constantemente, la señal de salida coincidirá con la entrada con un desfase de una unidad de tiempo.

En general, cualquier clase de memoria en una red dependerá de la existencia de bucles (realimentaciones). Si no los hay, ante la ausencia de estimulación externa las células pronto llegarían a un estado de inactividad, con el que no se puede representar lo que ha ocurrido en el pasado. Por tanto el estado interno de una red sin ningún ciclo vendrá determinado por los estímulos ocurridos en una porción limitada del pasado inmediato, dicho límite se corresponderá con la longitud del camino más largo existente en su interior. Como en una red con realimentaciones la longitud de camino existente en el interior no está limitada, tampoco lo está la cantidad de tiempo durante la cual la red mantiene la información. Aquí hay otra diferencia clara entre el funcionamiento de nuestro modelo neuronal y el cerebro: no toda la información que almacena el cerebro está almacenada de este modo. No se conoce hasta qué punto la existencia de realimentaciones neuronales tiene importancia en la memoria humana.

El escalador binario

Se denomina escalador binario a un dispositivo que produce un pulso por cada dos pulsos que hayan entrado. Una red que implementa este dispositivo es la de la Figura 6.9. El comportamiento es el siguiente: la neurona A funciona como un retardo unitario, disparando un instante después de que por su entrada haya llegado un pulso. Una vez que esto ocurra, la neurona A seguirá disparando hasta que sea inhibida por la salida de toda la red (que para la neurona A es una entrada inhibitoria). Una vez que a la neurona A haya llegado un primer pulso, si llega otro pulso más, la neurona B disparará (han llegado dos pulsos por tanto, y la red dispara un pulso) e inicializará de nuevo a la neurona A para volver a comenzar el proceso.

El escalador binario que hemos presentado en la Figura 6.9 no funcionará correctamente si los pulsos de entrada llegan en instantes muy próximos, como por ejemplo 1111. Para comprender este hecho, estudiemos el diagrama de transición de estados de esta red neuronal. Cada neurona tiene 2 estados (activa o inactiva), con lo cual en la red habrá un total de 4 estados posibles (debido a que la red consta de 2 neuronas). Cada estado estará caracterizado por qué neurona está activa; así los cuatro estados de la red son: oo (ni A ni B están activas), Ao (A activa, B inactiva), oB (B activa, A inactiva) y AB (A y B activas). Las dos posibles entradas a la red se representan por {0, 1}. En el

diagrama de transición que se muestra en la Figura 6.10 se observa que en el estado AB, tanto si la entrada es un 0 ó un 1, se transita al estado oo, habiendo activado la salida de la red en esta transición. Dicho de otro modo, en la transición del estado AB al estado oo, que es cuando la red dispara su salida, se ignora la entrada de la red.

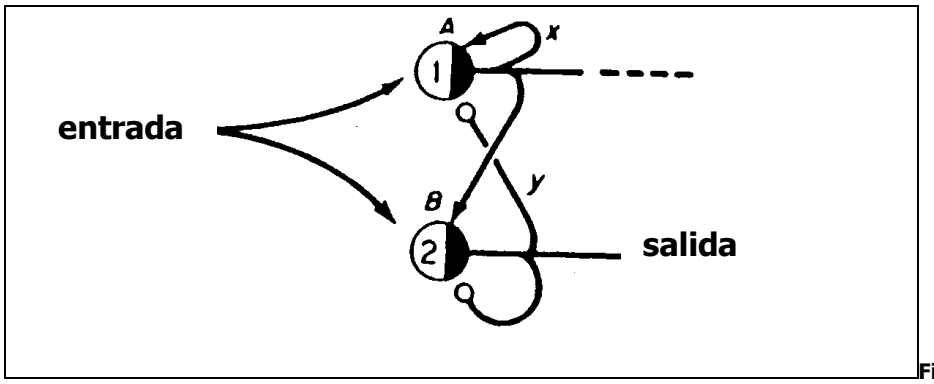

Figura 6.9 Un escalador binario defectuoso

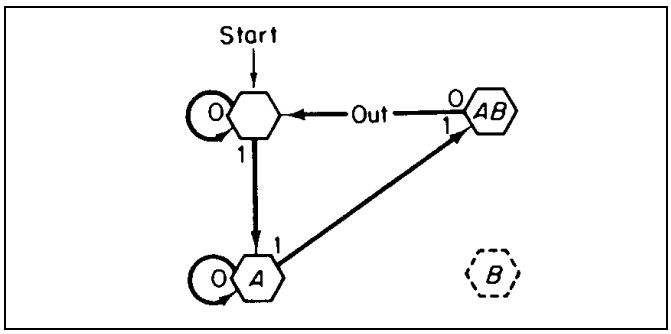

Figura 6.10 Diagrama de transición de estados del escalador binario defectuoso.

Para solventar dicho problema se podría añadir una neurona C que reconozca esta situación y la solucione; así obtendríamos la red que se muestra en la Figura 6.11. En este diseño interesa observar que la salida de la neurona A estará siempre inactiva después de haber recibido un número par de pulsos, sin embargo (con frecuencia) permanecerá disparando de forma regular cuando a la entrada de la red haya llegado un número impar de pulsos. En realidad deberíamos decir que después de un número impar de pulsos, o bien A o bien C permanecen activas de forma regular y estable.

Este comportamiento se puede observar en el diagrama de transición de estados que se muestra en la Figura 6.12 en el que los estados de la parte superior corresonden a estados en los que la red ha recibido un número par de pulsos en su entrada, mientras que los de la parte inferior corresponden a un número impar de pulsos en la entrada.

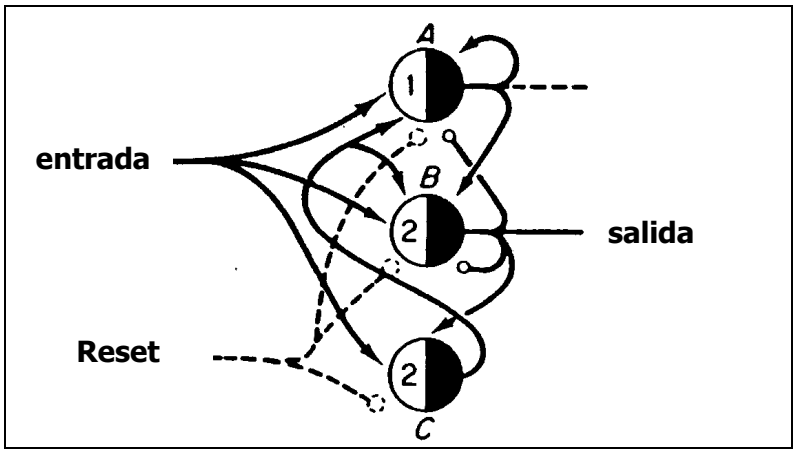

Figura 6.11 Escalador binario correcto

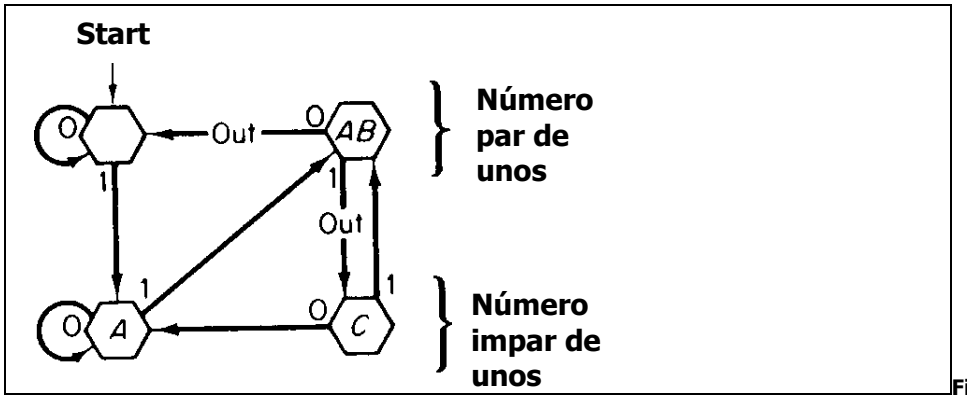

Figura 6.12 Diagrama de transición de estados del escalador binario corregido

Práctica 6.1: Simulador de redes neuronales formales

La práctica consistirá en realizar un simulador de redes neuronales que utilice el modelo de neurona de McCulloch y Pitts estudiado en clase. El simulador debe cargar desde un fichero la estructura de la red a simular, y desde otro las entradas que tendrá dicha red a lo largo del tiempo, y debe emitir por pantalla (y en otro fichero) una tabla con el valor de las entradas y estado de las neuronas conforme evoluciona la red.

Todos los ficheros que se utilizarán serán ficheros de texto.

Fichero de definición de la red neuronal.

Los ficheros que contendrán las estructuras de las redes neuronales tendrán la extensión .RED y la siguiente estructura:

- Línea 1: Número de neuronas+entradas (la suma del número de neuronas y el número de entradas) que posea la red.
- Definición de las entradas.
  A partir de la segunda línea del fichero (incluída la 2ª), habrá tantas líneas como entradas tenga la red, todas ellas conteniendo un -1 (habrá tantas lineas conteniendo el valor –1 como entradas tenga la red). Las entradas se numerarán desde 1 en adelante (la primera entrada es la 1, la 2ª la 2, etc. aunque estos números no aparecen en el fichero que describe la red).
- Definición de neuronas.
  Comenzará en la línea siguiente a la última entrada. Se utilizará una línea por cada neurona de la red. A cada neurona se le asociará un número (aunque estos números no

aparecerán en el fichero). El número de la primera neurona de la red que se numere será el siguiente al de la última entrada.
Cada línea contendrá en primer lugar un entero con el umbral de dicha neurona, y a continuación una lista con los números asociados a las entradas que posee dicha neurona. El valor absoluto de dichos números indicará de dónde proviene la entrada (si de otra neurona -en este caso el número corresponderá con un número de neurona- o de una entrada externa -en este caso el número será el correspondiente a una de las entradas-) y el signo indicará si es de tipo inhibitorio o excitatorio según sea negativo o positivo respectivamente.

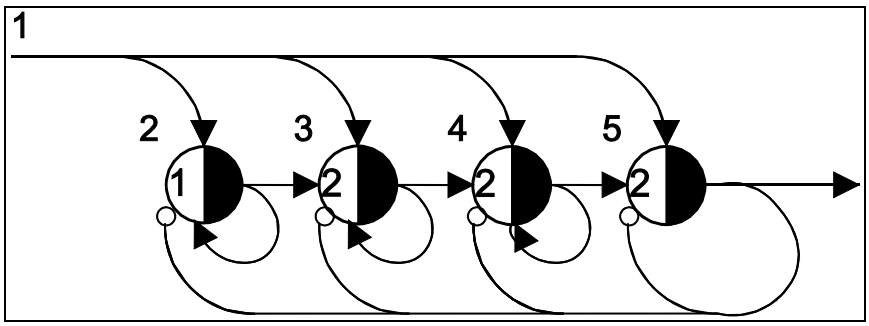

Figura 6.13 Ejemplo de red neuronal. Los números 2, 3, 4 y 5 son los números de las neuronas, comenzando a partir del de la última entrada (la 1).

Para la red de la Figura 6.13, el fichero que describe su estructura es el que se muestra en la Figura 6.14.

```
5
-1
112-5
2132-5
2143-5
214-5
```

Figura 6.14 Fichero de descripción de la red de la Figura 6.13.

Veamos la explicación de la información contenida en el fichero de la Figura 6.14 (para entenderla, ver también la Figura 6.13):

- Línea 1: El 5 corresponde a una red de 4 neuronas con una única entrada (4+1).
- Línea 2: El -1 indica una entrada. La designaremos por entrada 1. Como no hay más valores -1, ello indica que la red tiene una única entrada.
- Línea 3: Esta línea describe una de las neuronas de la red, la número 2 (el número es el siguiente a la última entrada, aunque este número 2 no aparece): tiene un umbral de 1 (primer número en la línea 3), y tres entradas procedentes de la entrada (número 1), de sí misma (número 2), y de la neurona número 5 (número -5). De estas entradas sólo la última es inhibitoria (por ser negativa).
- Las líneas 4, 5 y 6 describen el resto de neuronas de la red.

Fichero de definición de las entradas a la red.

El fichero de entradas tendrá la extensión .IN y estará formado por líneas que contienen cada una tantos valores (0 ó 1) como líneas de entrada posea la red. Dichos valores se corresponderán con las entradas que ha de tener la red en los sucesivos intervalos de tiempo.

Un posible ejemplo de 10 entradas para la red anterior sería el que se muestra en la Figura 6.15. Este fichero indica que en el instante t_0 la entrada vale 1, en t_1 vale 0 y así sucesivamente.

```
1
0
0
1
1
0
```

```
1
0
0
0
```

Figura 6.15 Fichero de entradas a la red.

Tabla de salida.

La tabla que se debe imprimir por pantalla ha de contener el mismo número de líneas que en el fichero de entradas, y cada línea contendrá:

- Número del instante de tiempo que se representa (serán números naturales comenzando con el 0).
- Valor de cada una de las entradas en dicho instante.
- Estado de activación (1 o bien 0) en dicho instante de cada una de las neuronas que posea la red.

```
Entrada     Neuronas
t |   1   2   3   4   5
-------------------------
0|    1   0   0   0   0
1|    0   1   0   0   0
2|    0   1   0   0   0
3|    1   1   0   0   0
4|    1   1   1   0   0
5|    0   1   1   1   0
6|    1   1   1   1   0
7|    0   1   1   1   1
8|    0   0   0   0   0
9|    0   0   0   0   0
```

Figura 6.16 Tabla de salidas de la red.

Para los dos ficheros anteriores, la tabla de salida es la que se muestra en la Figura 6.16

Estructura del programa:

Presentaremos en la Figura 6.17 las especificaciones de un posible diseño del programa que se pide realizar.

En este diseño se supone que toda dendrita es la salida de una determinada neurona. Normalmente esto es siempre cierto salvo para las dendritas que son las entradas a la red. En nuestro diseño suponemos que las entradas a la red proceden de unas neuronas ficticias; es por ello que en el fichero de especificación de la red se incluye el número de neuronas + entradas: ese será el número toal de neuronas (ficticias y reales) de nuestro sistema.

La estructura del fichero de definición de red está pensada para tener en cuenta este tipo de diseño.

En la Figura 6.17 observamos que cada neurona está caracterizada por un umbral, dos estados (el que tiene en un instante dado y el que pasará a tener en el instante siguiente en función de sus entradas) y punteros a las dendritas excitatorias e inhibitorias que entran en la neurona.

El conjunto de toda la red se representa por un puntero (*red*) a las neuronas. Una vez que se aloje memoria dinámica para toda la red, será conveniente tratar este identificador como un array, y referirse a la neurona i-ésima como *red[i]*.

```
enum conj_est {INACTIVA = 0, ACTIVA = 1};/* Estados de neuronas */

typedef struct nodo_d {/* Dendrita */
  unsigned neu;/* Neurona/entrada de la que viene la dendrita */
  struct nodo_d *sig;/* Siguiente dendrita */
} dendrita;

typedef struct {/* Representación de una neurona */
  enum conj_est estado[2];/* Estados actual y nuevo de la neurona */
  unsigned short umbral;/* Umbral de la neurona */
  dendrita *inh, *exi;        /* Dendritas inh. y excit. */
} neurona;

/************** variables globales ******************************/

unsigned total_ne;/* Número total de neuronas+entradas */
neurona *red;/* Red neuronal */
unsigned e_actual = 0,
e_nuevo = 1,/* Estados actual y nuevo de las neuronas  */
/* neuronas. Después de evolucionar, los   */
/* estados se intercambian: e_actual pasa  */
/* a ser e_nuevo y e_nuevo pasa a e_actual */
instante;/* Instantes de tiempo */

/***************** prototipos de funciones **********************/

void err_formato(void);
        Notifica un error de formato
void lee_red(char *nom);
        Lee el fichero con las características de la red y lo carga en
        memoria dinámica, inicializando sus estados de activación a nulo
void lee_entradas(FILE *fd);
        Lee una fila de las entradas desde el fichero de entradas y las
        carga en el estado de las neuronas correspondientes a entradas
void cambiar_red(void);
        Cambia los estados de todas las neuronas de la red de acuerdo a
        las reglas descritas por McCulloch y Pitts
void imprimir_cabecera(void);
        Imprime la cabecera de la tabla de estados de las neuronas
void imprimir_red(void);
        Imprime la tabla de estados de las neuronas para un instante
        dado
void itera_red(char *nom);
        Función que se encarga de llamar a las otras iterativamente, con
        lo que se irá construyendo la tabla de estados
```

Figura 6.17 Estructuras básicas del programa.

Notas finales:

Los nombres de los ficheros de entrada (el de definición de la red y el de entradas) deben leerse como parámetros a través de la línea de comandos del programa. Si no se encuentran dichos nombres en la línea de comandos, el programa deberá explicar el tipo de entrada que requiere.

En el servidor ftp del CSI se encuentran los programas

ftp://ftp.csi.ull.es/pub/asignas/AUTOMALF/p06_RN/simularn

ftp://ftp.csi.ull.es/pub/asignas/AUTOMALF/p06_RN/simularn.exe

que tienen un comportamiento similar al que ha de tener el programa que se pide diseñar en esta práctica.