

OPENCOURSEWARE
REDES DE NEURONAS ARTIFICIALES
Inés M. Galván – José M. Valls



Tema 6: Redes de Neuronas Recurrentes

En este tema se estudian las redes de neuronas recurrentes. Se presentan en primer lugar las características particulares de estas redes, para pasar a describir en detalle los tipos de redes recurrentes más conocidas.

En primer lugar, la red de Hopfield, que se caracteriza por estar formada por un conjunto de neuronas conectadas entre ellas y que se utiliza como memoria asociativa de patrones, en el sentido que es capaz de almacenar patrones para posteriormente ser recuperados.

A continuación se presentan dos tipos de redes, la red de Jordan y la red de Elman, que se refieren generalmente como redes parcialmente recurrentes, ya que solo unas pocas conexiones recurrentes son consideradas. Estas redes son muy similares al Perceptron Multicapa, pero poseen un conjunto de neuronas especiales, llamadas neuronas de contexto, que permiten el tratamiento de información temporal. Son redes fáciles de usar y su aprendizaje se lleva cabo con el algoritmo de retropropagación.

Posteriormente se estudian las redes totalmente recurrentes, para las cuales no hay ninguna restricción en la conectividad. Para el aprendizaje de estas redes no se puede utilizar el algoritmo de retropropagación, por lo que es necesario desarrollar otros algoritmos de aprendizaje. En este tema se estudian dos algoritmos de aprendizaje para las redes con conexiones recurrentes y con pesos asociados a estas conexiones.

Finalmente se presenta un tipo de red recurrente llamada Long Short-Term Memory (LSTM). Esta red presenta características particulares, que hace que su estudio se diferencie, pero no deja de ser una red recurrente para la cual se pueden aplicar los algoritmos de aprendizaje estudiados anteriormente.

A continuación se resume el contenido de las transparencias.

6.1 Introducción

Las redes de neuronas recurrentes se caracterizan porque las conexiones no solo se dirigen de la entrada a la salida, como las redes estudiadas hasta ahora, si no que pueden aparecer conexiones de una neurona a ella misma, de neuronas de una capa a neuronas de la capa anterior o entre neuronas de la misma capa. La inclusión de estas conexiones recurrentes permite que la activación de las neuronas en la red no solo dependa del patrón presentado en la entrada, sino también de valores anteriores de las activaciones y como consecuencia de patrones de entrada presentados anteriormente. En las redes como el PM o RBR, la salida depende solo de las entradas. En las redes recurrentes la salida depende también de las entradas en ese instante pero también de estados anteriores de la red. Esto hace que las redes

recurrentes puedan procesar **información temporal** y presenten un comportamiento dinámico, es decir para un mismo patrón de entrada, la salida de la red puede ser diferente. Generalmente son redes que tienen un mayor número de pesos ajustables, lo que hace que aumente su capacidad de representación, pero también se complica su aprendizaje.

Para el cálculo de la activación de una neurona recurrente es necesario incluir la variable tiempo (t) para tener en cuenta las activaciones de dónde vienen las conexiones recurrentes. Por lo demás, la manera de calcular la activación es igual que en redes como el PM (ver transparencia 5).

Las redes recurrentes se pueden utilizar en todas aquellas aplicaciones en las que los patrones no son independientes unos de otros, sino que existe una relación temporal entre ellos. Por ejemplo, en reconocimiento de caracteres, en las que un carácter aislado puede no ser reconocible, pero si en el contexto de una secuencia de caracteres o palabra; también en la modelización de sistemas dinámicos, como sistemas de control; y en la predicción de series temporales, donde los patrones de entrada siguen una secuencia temporal.

6.2 Red de Hopfield

La red de Hopfield fue propuesta por Hopfield en 1982. Se caracteriza por estar formada por n neuronas, en la que todas las neuronas están conectadas con todas las demás, salvo con ella misma (ver figura en transparencia 10). Estas conexiones llevan asociados pesos, resultando una matriz de pesos $W=(w_{ij})$ de orden $n \times n$ que se caracteriza por ser simétrica y con ceros en la diagonal.

El estado de una red de Hopfield en un instante de tiempo $t+1$ se define como un vector formado por los estados en ese instante de todas las neuronas, $S(t+1)=(s_1(t+1), \dots, s_n(t+1))$, donde $s_i(t+1)$ es 1 ó -1 dependiendo del signo del valor que le llega a la neurona (transparencia 11):

Esta red se suele utilizar como memoria asociativa, en el sentido que está diseñada para almacenar un conjunto de patrones, para después recuperarlos a partir de información incompleta e incluso a partir de patrones con ruido. Por tanto, el **aprendizaje de la red o fase de almacenamiento** consiste en determinar los pesos de la red de modo que almacene un conjunto de patrones dados, utilizando la expresión dada en la transparencia 15, que básicamente multiplica las diferentes coordenadas de los patrones de entrada. Al calcular los pesos utilizando esta expresión, lo que se está haciendo es minimizar la función energía dada en la transparencia 19.

Una vez calculados los pesos, en la **fase de recuperación**, se puede presentar un patrón desconocido (diferente de los patrones almacenados) y obtener el patrón más parecido del conjunto de patrones almacenados. Para ello, basta inicializar el estado cada neurona con el patrón dado, y hacerla evolucionar hasta alcanzar un punto estable ($S(t+1)=S(t)$). Dicho estado estable representará el patrón recuperado.

6.3 Redes parcialmente recurrentes

Las redes parcialmente recurrentes son redes del tipo PM (capa de entrada, capa oculta y capa de salida) con unas cuantas conexiones recurrentes. En la capa de entrada se encuentran las **neuronas de contexto** que reciben las conexiones recurrentes, actuando como una memoria de la red que almacena el estado o activación de las neuronas de una cierta capa en el instante

anterior. En la figura de la transparencia 22 se muestra la estructura de una red parcialmente recurrente.

Existen dos tipos de redes parcialmente recurrentes. La red de Jordan (ver figura de la transparencia 24) y la red de Elman (ver figura de la transparencia 28). Ambas tienen la misma estructura, con la diferencia de las neuronas de contexto en la red de Jordan reciben una copia de las neuronas de salida, mientras que en la red de Elman reciben la activación de las neuronas ocultas. Las neuronas de contexto en la red de Jordan poseen también conexiones a ellas mismas, a través de un parámetro fijo μ (valor entre 0 y 1) que actúa como inercia para la memoria de la red. De este modo, las activaciones de las neuronas de contexto vienen dadas por:

$$\text{Red de Jordan: } c_i(t) = \mu c_i(t-1) + y_i(t-1)$$

$$\text{Red de Elman: } c_i(t) = a_i(t-1)$$

En la red de Jordan habrá tantas neuronas de contexto como salidas tenga la red y en la red de Elman, tantas como neuronas ocultas. En ambos casos, la salida de la red va a depender de las entradas en un instante de tiempo, pero también del estado de la red en instantes anteriores. La red de Elman solo tiene en cuenta el estado anterior, y en la red de Jordan se memorizan todos los instantes anteriores, controlado por el parámetro μ .

Concatenando el vector de entrada con la activación de las neuronas de contexto $(x_1(t), \dots, x_n(t), c_1(t), \dots, c_m(t))$, ambas redes pueden verse como un PM. Por tanto, a la hora de realizar el aprendizaje puede utilizarse el algoritmo de retropropagación o regla delta generalizada, ya que las conexiones recurrentes de estas redes no llevan asociado un peso. El aprendizaje, al igual que para el PM, se realiza para minimizar el error medido en la salida. Los pasos para el funcionamiento y aprendizaje de estas redes se muestra en la transparencia 30.

6.4 Redes totalmente recurrentes

Dentro del grupo de redes totalmente recurrentes se encuentra cualquier red en la que no exista restricción en la conectividad entre las neuronas. Así por ejemplo, una única neurona con conexión recurrente a ella misma se puede considerar una red totalmente recurrente. El aprendizaje de estas redes no es tarea sencilla. El algoritmo de retropropagación utilizado para el PM y las redes de Jordan y Elman no puede utilizarse para una red recurrente, aunque el objetivo siga siendo minimizar el error en la salida. Esto es debido (revísese la deducción de la regla delta generalizada, transparencias de la 19 hasta la 32 del Tema 3) a que a la hora de obtener la derivada de la activación de la neurona con respecto al peso, hay que tener presente que esa activación depende de la activación anterior, que a su vez también depende del peso. Por tanto, las expresiones que se obtienen al calcular las derivadas no son las mismas que si se trata de una red con conexiones hacia adelante.

Existen dos algoritmos para el entrenamiento de las redes totalmente recurrentes:

- **Retropropagación a través del tiempo** (Backpropagation through time). Este algoritmo consiste básicamente en desarrollar la red recurrente en un intervalo de tiempo, resultando entonces un PM (ver figura transparencia 35). Ante esta situación, es posible aplicar el algoritmo de retropropagación. Por ejemplo, para la red que se muestra en la transparencia 35, si el intervalo de tiempo es $[n,2]$, resulta una red con 2 entradas, 1 capa oculta con dos neuronas y 2 salidas. Bastaría entonces aplicar el algoritmo de retropropagación, teniendo en cuenta que los

pesos coinciden para todas las capas de la red multicapa desarrollada, por lo que al calcular el gradiente con respecto a un peso hay que hacerlo individualmente para cada capa y posteriormente sumarlos.

- **Aprendizaje en tiempo real** (Real time recurrent learning). Este algoritmo consiste en calcular las derivadas del error con respecto a los pesos, teniendo en cuenta que la activación de una neurona también puede depender de activaciones anteriores, que a su vez dependen también del peso (ver ejemplo transparencia 32). En las transparencias 39-42 se deduce la ley de aprendizaje para una red recurrente general con M neuronas de entradas y N neuronas más (de las cuales algunas pueden ser de salida) y donde todas las neuronas están conectadas con todas las demás. En la ley de aprendizaje que se obtiene (transparencia 42) interviene el término (p_{kl}^i) que es la salida de un sistema de ecuaciones diferenciales. Este término contempla la derivada de las activaciones anteriores de la red respecto al peso.

Ambos algoritmos poseen ventajas e inconvenientes. El primero es relativamente sencillo, pero si el intervalo de tiempo es amplio, puede ocurrir el problema de “desvanecimiento del gradiente” debido a que el número de neuronas ocultas aumenta al aumentar el ancho del intervalo. Además, es necesario definir el intervalo de tiempo más adecuado para cada problema. El segundo suele ser bastante eficiente, si bien las ecuaciones que se obtienen son complejas. Además, tiene una alta dependencia de los pesos iniciales de la red, así como los valores iniciales para resolver las ecuaciones $((p_{kl}^i(0))$.

Ante las dificultades encontradas en los algoritmos de aprendizaje para redes recurrentes, han surgido nuevas ideas y alternativas para el aprendizaje de estas redes. Entre ellas, se destaca aquí el uso de técnicas de computación evolutiva para a determinación de los pesos y más recientemente Deep Learning. Ambas aproximaciones también se pueden utilizar (y de hecho se han utilizado) para entrenar otras redes, como el PM o RBR.

Finalmente, en este tema se estudia una arquitectura particular de red recurrente, **Long short-term memory (LSTM)** propuesta por Sepp Hochreiter y Jürgen Schmidhuber en 1997. Se caracteriza porque está compuesta por unidades en la capa oculta, llamadas “memory blocks” (ver figura en la transparencia 43). Cada bloque (LSTM) puede contener varias células y cada célula sigue el esquema que se muestra en la figura de la transparencia 44.

Cada célula o neurona tiene asociada tres puertas, conocidas como puertas de entrada, olvido y salida, que se encargan de habilitar o deshabilitar entradas (o salidas) a (de) la célula. Las puertas tienen el objetivo de permitir o no que la célula utilice información almacenada a lo largo del tiempo. Así, si la puerta de entrada está cerrada (activación cercana a cero), la activación de la célula no dependerá de nueva información que llegue a la red a través de las entradas o de otras células. De manera similar, la activación de la célula está disponible para el resto de las células de la red dependiendo de si la puerta de salida está abierta o cerrada, y dependerá de su activación en instantes de tiempo anteriores dependiendo de la puerta de olvido.

Las puertas actúan como neuronas que reciben a través de los pesos (W^{tipo}) las activaciones de otras células o de entradas externas, dependiendo del tipo de puerta. Por tanto, la activación de las puertas se calculan como es habitual en las redes de neuronas: función de activación aplicada al sumatorio del producto de las activaciones que le llegan por sus pesos. Ver expresión $g_i^{tipo}(t)$ en transparencia 47.

La salida de la célula viene dada por el estado s_i de la célula y la activación de la puerta de salida (transparencia 45). Y finalmente el estado de la célula depende de la entrada total a la célula net_i , del estado anterior y de las puertas de entrada y olvido, respectivamente (transparencia 46).

En la red LSTM aparecen pesos asignados a las puertas (W^{tip0}) y pesos asignados a las células (W^{cell}). El aprendizaje de la red consiste en determinar ambos conjuntos de pesos. Estas redes se han entrenado con el algoritmo de retropropagación a través del tiempo y más recientemente con algoritmos evolutivos, dando lugar al sistema conocido como EVOLINO, o también con Deep Learning.