Tecnológico de Monterrey

Redes Neuronales Recursivas

Aprendizaje Automático

Víctor de la Cueva

vcueva@itesm.mx

Redes Neuronales

EL MODELO DE HOPFIELD

John Joseph Hopfield (1933-)

- Es uno de los principales responsables del resurgimiento y avance de las RN.
- Profesor de Biología Molecular en la Universidad de Princeton.
- Ganador del IEEE Computational Intelligence Society 2009 Frank Rosenblatt Award.
- Su trabajo combina neurobiología, física e ingeniería eléctrica.
- Construyó un modelo de red [Hopfield, 1982].

El modelo

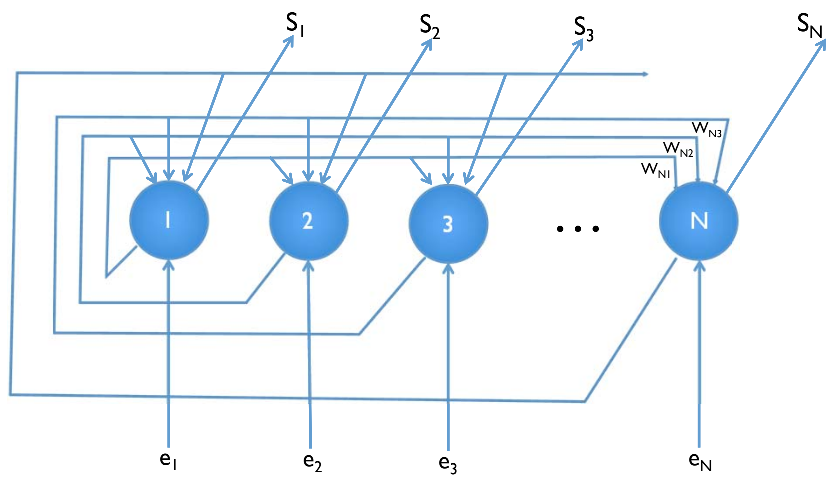
- Tiene el número de simplificaciones suficiente como para poder extraer analíticamente información sobre las características relevantes del sistema, conservando las ideas fundamentales de las redes construidas en el pasado y presentando una serie de funciones básicas de sistemas neuronales reales.
- Supo establecer un paralelismo entre su modelo y ciertos sistemas exitosamente estudiados en física estadística, lo cual permitió aplicar todo un conjunto de técnicas bien conocidas en este campo y, con ello, producir un avance en la comprensión del funcionamiento de las RN.
- Con esta aportación redescubrió el mundo casi olvidado de las <u>redes autoasociativas</u>, caracterizadas por una nueva arquitectura y un nuevo funcionamiento, a las que añadió un nuevo tipo de reglas de aprendizaje.

Arquitectura

- Consiste en una red monocapa con N neuronas cuyos valores de salida son binarios o bipolares.
- En la versión original del modelo (Discrete Hopfield o DH), las funciones de activación de las neuronas eran de tipo escalón.
- Posteriormente [1984] desarrolló una versión continua (CH), con entradas y salidas analógicas y funciones de activación sigmoidales.

- Cada neurona de la red se encuentra conectada a todas las demás (conexiones laterales), pero no consigo misma (no existen conexiones autorrecurrentes).
- Los pesos asociados a las conexiones de pares de neuronas son simétricos, es decir, el peso de la conexión de una neurona i con otra neurona j es de igual valor que el de la conexión inversa ($w_{ij}=w_{ji}$).

Red de Hopfield



Función de activación DH

- La DH fue ideada para trabajar con valores binarios -1 y +1 (aunque mediante un ajuste en los pesos pueden usarse en su lugar los valores de 1 y 0).
- Entonces, la función de activación de cada neurona i de la red es de tipo escalón:

$$f(x) = \begin{cases} +1 & x > \theta_i \\ -1 & x < \theta_i \\ x & x = \theta_i \end{cases}$$

- θ_i es el umbral de disparo de la neurona i.
- En el modelo discreto suele adoptarse un valor proporcional a la suma de los pesos de las conexiones de la neurona con el resto:

$$\theta_i = \mathrm{k} \sum_{j=1}^{N} w_{ji}$$

- Si se trabaja con valore -1 y +1, suele usarse $\theta_i = 0$. Si los valores son 0 y 1, se toma k=1/2.

Función de activación CH

- En ambos casos la función de activación es de tipo sigmoidal.
- Si se trabaja con valores reales [-1,+1], la función que se utiliza es la tangente hiperbólica:

$$f(x-\theta_i) = tgh\big(\alpha(x-\theta_i)\big) = \frac{e^{\alpha(x-\theta_i)} - e^{-\alpha(x-\theta_i)}}{e^{\alpha(x-\theta_i)} + e^{-\alpha(x-\theta_i)}}$$

- Si se trabaja con valores reales [0,1], se utiliza la sigmoidal:

$$f(x-\theta_i) = \frac{1}{1+e^{-\alpha(x-\theta_i)}}$$

- En ambos casos α es un parámetro que determina la pendiente de la función sigmoidal.

Funcionamiento

- Se trata de una red autoasociativa.
- Varios patrones diferentes pueden ser almacenadas en la red, como si se tratara de una memoria, durante la fase de aprendizaje (memoria autoasociativa).
- Posteriormente, si se presenta a la entrada alguno de los patrones almacenados, la red evoluciona hasta estabilizarse, ofreciendo entonces en la salida la información almacenada que coincida con la presentada.
- Si no coincide por estar distorsionada o incompleta, la red evoluciona generando como salida la más parecida.
- La información que recibe esta red debe haber sido codificada previamente y representada en forma de vector (como una configuración binaria si es DH, y como conjunto de valores reales si es CH), con tantas componentes como neuronas (N) tenga la red.

Pasos del Funcionamiento

1. En el instante inicial (t=0) se aplica la información de entrada (valores $e_1, e_2, \ldots, e_N$).
$$s_i(t=0) = e_i \qquad 1 \leq i \leq N$$
Inicialmente, la salida de las neuronas coincide con la información aplicada a la entrada.
2. La red realiza iteraciones hasta alcanzar la convergencia, es decir, hasta que:
$$s_i(t+1) = s_i(t).$$
Si f es la función de transferencia:
$$s_i(t+1) = f\left(\sum_{j=1}^{N} w_{ij} s_j(t) - \theta_i\right) \qquad 1 \leq i \leq N$$
Si se usa DH (-1 y +1). Sea B = $\sum_{j=1}^{N} w_{ij} s_j(t)$:
$$s_i(t) = \begin{cases} +1 & si\ B > \theta_i \\ s_i(t) & si\ B = \theta_i \\ -1 & si\ B < \theta_i \end{cases}$$
Si se usa CH, la función de activación es sigmoidal.

Aprendizaje

- Tiene un mecanismo de aprendizaje OFF-LINE, es decir, existe una etapa de aprendizaje y una de producción.
- Esta red utiliza un aprendizaje no supervisado de tipo hebbiano.
- El peso de una conexión entre la neurona i y la j se obtiene mediante el producto de los componentes i-ésimo y j-ésimo del vector que representa la información o patrón que debe almacenar.
- Si el número de patrones a aprender es M, el valor definitivo de cada uno de los pesos se obtiene mediante la suma de los M productos.

Obtención de los pesos

- En el caso de DH con valores -1 y +1, el algoritmo de aprendizaje se expresa como:

$$w_{ij} = \begin{cases} \sum_{k=1}^{M} e_i^{(k)} e_j^{(k)} & 1 \leq i,j \leq N; i \neq j \\ 0 & 1 \leq i,j \leq N; i = j \end{cases}$$

- Donde:
  - w_{ij}: peso asociado a la conexión entre la neurona j y la i.
  - $e_i^{(k)}$:Valor de la componente te i-ésima del vector correspondiente a la información k-ésima que debe aprender la red.
  - N: número de neuronas de la red.
  - M: Número de patrones que debe aprender la red.

- Si la red trabaja con valores 0 y 1:

$$w_{ij} = \begin{cases} \sum_{k=1}^{M} \left(2e_i^{(k)} - 1\right)\left(2e_j^{(k)} - 1\right) & 1 \leq i,j \leq N: i \neq j \\ 0 & 1 \leq i,j \leq N: i = j \end{cases}$$

- El algoritmo de aprendizaje también suele expresarse utilizando una notación matricial W, de NXN, donde w_{ii}=0.
- Si los vectores de entrenamiento están dados por:

$$E_i = \left[e_1^{(i)}, e_2^{(i)}, \dots e_N^{(i)}\right]$$

- Entonces W es: $W = \sum_{k=1}^{M}\left[E_k^T E_k - I\right]$ donde E_k^T es la transpuesta de la matriz E_k, e I es la matriz identidad de NXN que anula los pesos de las conexiones autorrecurrentes (w_{ii}).

Limitantes

- El número de patrones que puede ser aprendido está severamente limitado:

$$Capacidad = \begin{cases} 0.138N & recuperación\ suficientemente\ buena \\ \dfrac{N}{4lnN} & para\ una\ recuperación\ perfecta \end{cases}$$

- Si los patrones no son suficientemente diferentes no siempre se garantiza que la red realice una asociación correcta.

Algunas Aplicaciones

- Reconocimiento de imágenes.
- Resolución de problemas de optimización.
  - TSP
  - Bipartición de grafos
  - Emparejamiento ponderado
- Diseño de circuitos conversores analógicos-digitales

Referencias

- J.A. Freeman and D.M. Skapura. Neural Networks: Algorithms, Applications, and Programming Techniques. Addison-Wesley, USA (1991).
- J.R. Hilera y V.J. Martínez. Redes Neuronales Artificiales: Fundamentos, modelos y aplicaciones. Adison-Wesley Iberoamericana, USA (1995).
- J.A. Anderson. Redes Neuronales. Alfaomega, México (2007).