

Modelo de Hopfield de red neuronal

Álvaro Zorrilla Carriquí - Grupo A

Junio 2023



UNIVERSIDAD
DE GRANADA

Resumen

En este informe se exponen los resultados de aplicar el algoritmo de Metrópolis utilizado en el modelo de Ising para simular y estudiar el comportamiento emergente de una red neuronal de Hopfield para varios valores de temperatura T , usando uno y tres patrones no aleatorios. También se estudia el número de patrones recordados según el número de patrones aleatorios de referencia que tenga la red neuronal según un sesgo para los mismos.

1. Introducción teórica, objetivos y metodología

El modelo de Hopfield (o modelo de Ising para una red neuronal) es un modelo de red neuronal ideado para resolver problemas de reconocimiento de patrones. No obstante, Hopfield también aseguraba que su modelo es útil para resolver otros ciertos problemas de optimización, como el problema del viajante.

En el modelo de Hopfield se considera una matriz bidimensional de neuronas $(s_{ij})_{ij}$ de dimensión $N \times N$, en la que cada neurona solo puede estar en un estado 0 (inactiva) o 1 (activa).

Cada una de las neuronas están conectadas entre sí, salvo con ella misma. Esa conexión se representa por ω_{ijkl} , que representa el peso de la conexión entre las neuronas (i, j) y (k, l) . Evidentemente, esos pesos son simétricos entre neuronas.

La función de activación de una neurona es:

$$f(x) = \begin{cases} 0 & \text{si } x < \theta_{ij} \\ x & \text{si } x = \theta_{ij} \\ 1 & \text{si } x > \theta_{ij} \end{cases},$$

donde θ_{ij} es el umbral de disparo de la neurona (i, j) , que se define como $\theta_{ij} = \frac{1}{2} \sum_{k=1}^N \sum_{l=1}^N \omega_{ijkl}$.

En el modelo que usamos, se parten de un número P de patrones preestablecidos, denotados por $(\xi_{ij}^\mu)_{ij}$, $\mu = 1, \dots, P$, y que la red tratará de "recordar", notando por $a^\mu = \frac{1}{N^2} \sum_{k=1}^N \sum_{l=1}^N \xi_{kl}^\mu$ a la proporción de neuronas activas en cada patrón. Los pesos ω_{ijkl} vienen dados

$$\omega_{ijkl} = \begin{cases} \frac{1}{N^2} \sum_{\mu=1}^P (\xi_{ij}^\mu - a^\mu)(\xi_{kl}^\mu - a^\mu) & \text{si } (i, j) \neq (k, l) \\ 0 & \text{si } (i, j) = (k, l) \end{cases}.$$

Así, con esta configuración la red neuronal de Hopfield está diseñada para tender a un estado igual a uno de los patrones preestablecidos.

La conexión entre este algoritmo y el modelo de Ising para la transición ferromagnética-paramagnética en dos dimensiones, es, en primer lugar, porque existe un Hamiltoniano para el modelo de Hopfield dado por:

$$H = -\frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \sum_{k=1}^N \sum_{l=1}^N \omega_{ijkl} s_{ij} s_{kl} + \sum_{i=1}^N \sum_{j=1}^N \theta_{ij} s_{ij}.$$

Este Hamiltoniano es análogo a la energía del sistema porque siempre que escojamos las neuronas a actualizar su estado aleatoriamente, la red convergerá a un mínimo local del funcional H , que viene dado por los patrones preestablecidos. Pero éstos no son los únicos mínimos existentes.

El objetivo de este informe es el de mostrar la convergencia de la red neuronal en función de la "temperatura" (en analogía con el modelo de Ising) de la misma, del número de patrones iniciales considerados y del patrón inicial de la red.

Para ello, utilizamos el algoritmo de Metropolis, según el cual la probabilidad de transición de la red de un estado s a otro estado s' viene dada por $\min\left(1, e^{-\frac{\Delta H}{T}}\right)$ (ver apéndice). Esta probabilidad se

compara con un número aleatorio entre 0 y 1, haciéndose el cambio o no según si ese valor es mayor o menor que la probabilidad anterior respectivamente.

Para cuantificar la convergencia de la red con los patrones, se define el solapamiento de la red en un estado s con cada uno de los patrones preestablecidos μ como:

$$m^\mu(s) = \frac{1}{N^2 a^\mu (1 - a^\mu)} \sum_{i=1}^N \sum_{j=1}^N (\xi_{ij}^\mu - a^\mu)(s_{ij} - a^\mu).$$

2. Resultados y comentarios.

Los *scripts*, los ficheros con los patrones y los resultados de la ejecución de los códigos empleados se encuentran en el repositorio “*leccion2-voluntario*” de mi perfil personal de GitHub <https://github.com/alzorrcarri>.

2.1. Un solo patrón preestablecido. Variación del patrón inicial y de la temperatura.

Se consideró como patrón preestablecido la imagen contenida en el archivo “*piplup.txt*” y el código empleado en el fichero “*script.cpp*”, contenidos ambos en el directorio “*apartado-1y2*” del repositorio referido.

Inicialmente, se estudió la evolución del solapamiento de la red en función del tiempo a una temperatura constante $T = 0,0001$ para dos patrones iniciales de la red como se muestra en la figura 1. La convergencia es rápida en ambos casos, dándose el caso de que el patrón aleatorio converge al negativo del patrón preestablecido, lo cual es posible ya que éste también es un mínimo del Hamiltoniano del sistema. A estos estados se los conoce como estados *espúreos*.

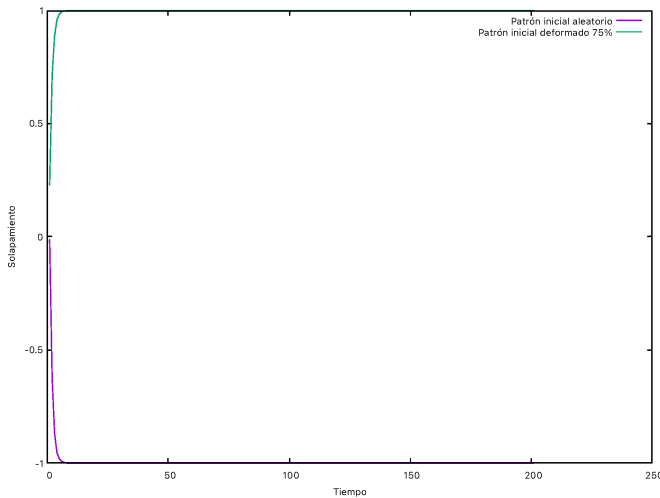


Figura 1: Evolución del solapamiento de la red con un único patrón preestablecido con el tiempo para un patrón inicial aleatorio y un patrón inicial igual al preestablecido deformado un 75 % para una temperatura igual a $T = 0,0001$.

Posteriormente, se estudió el comportamiento del solapamiento con el tiempo variando la temperatura de la red neuronal, considerando inicialmente un patrón aleatorio o una deformación del 50 % del preestablecido para la red neuronal. Los resultados se muestran en la figura 2. Para temperaturas menores o iguales a $T = 0,01$ la convergencia es prácticamente idéntica y rápida; mientras que para $T = 0,1$, la red neuronal queda en una configuración aleatoria con solapamiento variable y nulo en promedio en ambos casos (tanto para un patrón inicial aleatorio, como para uno deformado). Por tanto, para temperaturas $T > 0,01$, la red neuronal no convergerá al patrón y el solapamiento será cada vez más cercano a cero y variable con el tiempo hasta el caso de $T = 0,1$.

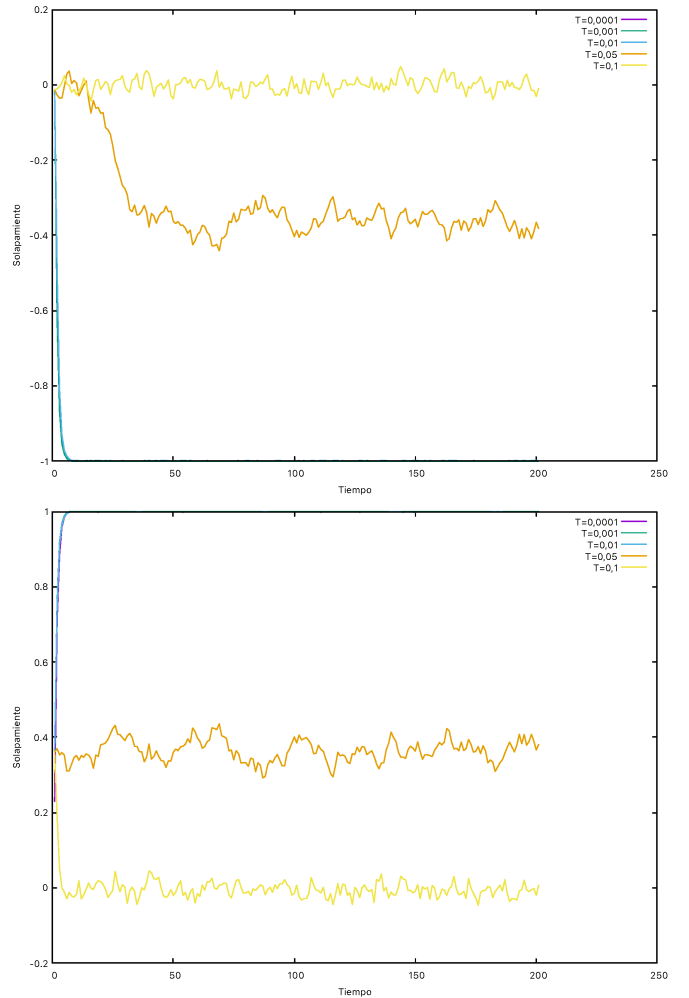


Figura 2: Evolución del solapamiento de la red con un único patrón preestablecido con el tiempo para un patrón inicial aleatorio (arriba) y un patrón inicial igual al preestablecido deformado un 50 % (abajo) para cinco valores de temperatura $T = 0,0001$, $T = 0,001$, $T = 0,01$, $T = 0,05$ y $T = 0,1$.

2.2. Tres patrones preestablecidos. Variación del patrón inicial y de la temperatura.

Para estudiar el comportamiento de la red neuronal con varios patrones consideramos, por ejemplo, tres patrones iguales a los números uno, dos y tres, contenidos en los archivos “1.txt”, “2.txt” y “3.txt”, respectivamente claro está, del directorio “apartado-3”.

Primero, considerando un patrón inicial aleatorio, observamos el comportamiento del solapamiento de la red neuronal con los tres patrones para dos temperaturas distintas. En la figura 3 se observa que la red converge al patrón del número 2 de entre los tres posibles para temperatura igual a $T = 0,0001$; pero, para $T = 0,05$, el solapamiento no converge, como era de esperar según lo mencionado en la sección anterior. El comportamiento del solapamiento de la red con el resto de patrones es similar al expuesto para el segundo patrón, salvo que para estos el solapamiento es menor.

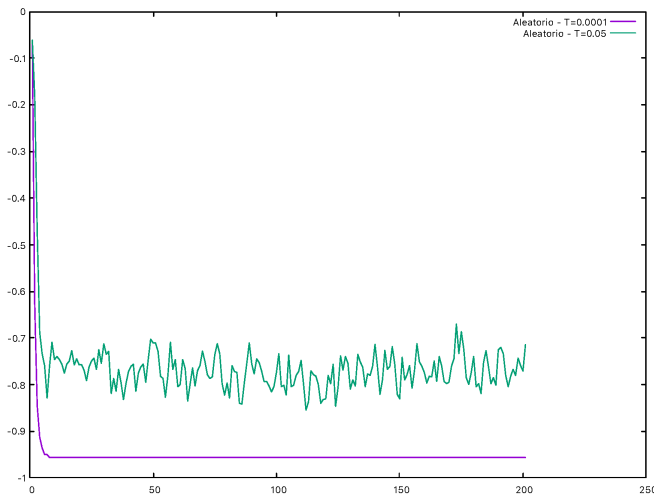


Figura 3: Evolución del solapamiento de la red neuronal con el patrón preestablecido 2 considerando inicialmente un patrón inicial aleatorio para dos temperaturas $T = 0,0001$ y $T = 0,05$.

Consideramos después como patrón inicial uno de los patrones preestablecidos deformados. En la figura 4 se muestra el solapamiento para la red neuronal con los patrones 1 y 2 tomando como patrón inicial una deformación del 50 % del patrón 1. El comportamiento para ambas temperaturas es el esperado; pero es curioso que, a pesar de partir del patrón 1 deformado, la red neuronal no converge a ese, sino al patrón 2, como puede observarse en la misma.

De hecho, al tomar como patrón inicial el patrón 3 deformado, para $T = 0,0001$, la red neuronal converge a una mezcla de los patrones 2 y 3, mostrando que los mínimos del Hamiltoniano del modelo de Hopfield pueden ser mezclas de patrones (ver figura 5).

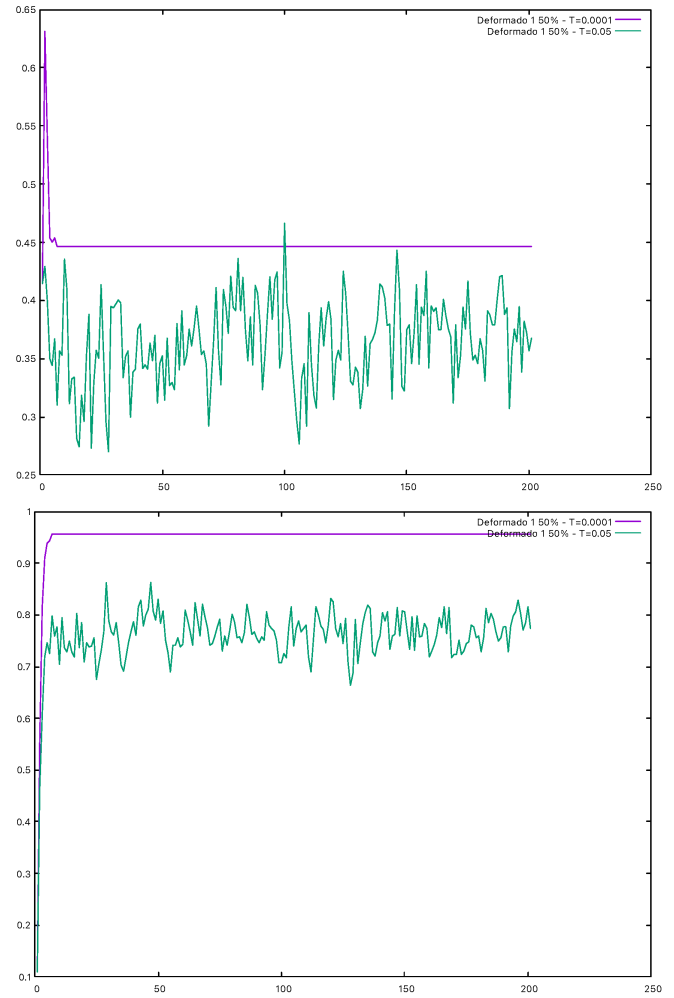


Figura 4: Evolución del solapamiento de la red neuronal con el patrón preestablecido 1 considerando inicialmente el patrón 1 (arriba) y el patrón 2 (abajo) deformados un 50 % para dos temperaturas $T = 0,0001$ y $T = 0,05$.

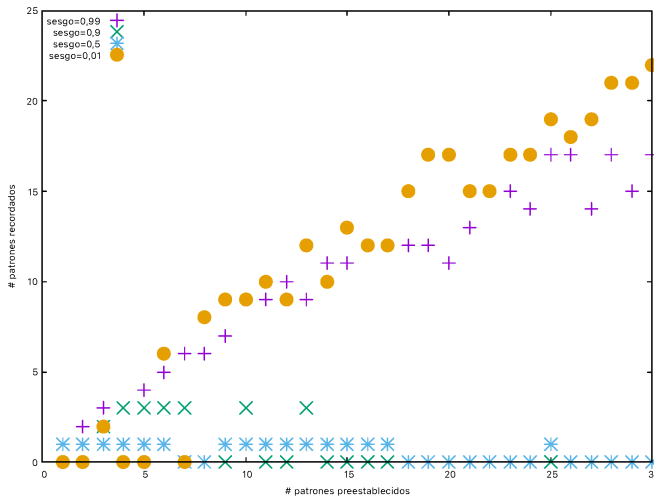


Figura 5: Estado final de la red neuronal al final de la ejecución del código para una temperatura $T = 0,0001$ y una configuración inicial del patrón 3 deformado un 50 %. Secuencia tomada del vídeo “aiv.mp4”.

3. Memoria en función del número de patrones.

Finalmente, el objetivo de esta sección es el de estudiar el decaimiento de la memoria de la red neuronal en función del número de patrones almacenados para una temperatura de la red de 10^{-4} . Para ello, tanto el estado inicial de la red como los patrones empleados se generan aleatoriamente, todos con dimensión 20×20 . El script creado para realizar este estudio se encuentra en el archivo "*script.cpp*", donde el número máximo de patrones empleados es 30. Adicionalmente, se introdujo un sesgo $s \in (0,1)$ en la generación aleatoria de los patrones para estudiar cómo el sesgo influenciaba la memoria de la red. Este sesgo hacía que la probabilidad de generar 1 o 0 en los patrones aleatorios no fuera $1/2$.

Consideramos que una red neuronal recuerda un patrón cuando el solapamiento es mayor a 0,75. De este modo, la memoria de la red es el número de patrones que la red recuerda. En la figura 6, se representa la evolución de la memoria en función del número de patrones considerados. El caso en el que los patrones se generan de forma puramente aleatoria es cuando el sesgo es igual a 0,5.



un único patrón al tomar pocos patrones, y no recordaba ninguno cuando el número de los mismos aumentaba. La generación de patrones aleatorios en los que se favorecía mucho un valor de los dos posibles, hacía que la memoria tomara valores más altos como cabría esperar, ya que al ser los patrones más parecidos entre sí, el solapamiento con ellos era mayor, y mayor la memoria.

Referencias

- [1] *Voluntario 2: Modelo de Hopfield de red neuronal.* Enunciado de clase del ejercicio. Física Computacional, Grado en Física. Universidad de Granada.
- [2] Daniel J. Amit, Hanoah Cxutfreund, and H. Sompolinsky, *Information storage in neural networks with low levels of activity.* Physical Review A, 35 5 (1987).

Apéndice de cálculo

En este apéndice mostramos una expresión reducida para el cambio de energía del sistema ΔH que se usará en los *scripts* de esta práctica. Supongamos que la neurona que se propone para el cambio ocupa la posición (p, q) de la matriz:

$$\Delta H = H' - H$$

$$\begin{aligned} H &= -\frac{1}{2} \left[\sum_{i=1}^N \sum_{j=1}^N s_{ij} \left(\sum_{k=1, k \neq p}^N \sum_{l=1, l \neq q}^N \omega_{ijkl} s_{kl} + \omega_{ijpq} s_{pq} \right) \right] + \sum_{i=1, i \neq p}^N \sum_{j=1, j \neq q}^N \theta_{ij} s_{ij} + \theta_{pq} s_{pq} \\ &= -\frac{1}{2} \left[\sum_{k=1, k \neq p}^N \sum_{l=1, l \neq q}^N s_{kl} \left(\sum_{i=1, i \neq p}^N \sum_{j=1, j \neq q}^N \omega_{ijkl} s_{ij} + \omega_{pqkl} s_{pq} \right) + \sum_{i=1, i \neq p}^N \sum_{j=1, j \neq q}^N \omega_{ijpq} s_{ij} s_{pq} + \omega_{pqpq} s_{pq} s_{pq} \right] \\ &\quad + \sum_{i=1, i \neq p}^N \sum_{j=1, j \neq q}^N \theta_{ij} s_{ij} + \theta_{pq} s_{pq} \\ &= -\frac{1}{2} \left[\sum_{i=1, i \neq p}^N \sum_{j=1, j \neq q}^N \sum_{k=1, k \neq p}^N \sum_{l=1, l \neq q}^N \omega_{ijkl} s_{ij} s_{kl} + s_{pq} \left(\sum_{k=1, k \neq p}^N \sum_{l=1, l \neq q}^N \omega_{pqkl} s_{kl} + \sum_{i=1, i \neq p}^N \sum_{j=1, j \neq q}^N \omega_{ijpq} s_{ij} \right) + \omega_{pqpq} s_{pq}^2 \right] \\ &\quad + \sum_{i=1, i \neq p}^N \sum_{j=1, j \neq q}^N \theta_{ij} s_{ij} + \theta_{pq} s_{pq}. \end{aligned}$$

Análogamente para H' , tenemos:

$$\begin{aligned} H' &= -\frac{1}{2} \left[\sum_{i=1, i \neq p}^N \sum_{j=1, j \neq q}^N \sum_{k=1, k \neq p}^N \sum_{l=1, l \neq q}^N \omega_{ijkl} s_{ij} s_{kl} + s'_{pq} \left(\sum_{k=1, k \neq p}^N \sum_{l=1, l \neq q}^N \omega_{pqkl} s_{kl} + \sum_{i=1, i \neq p}^N \sum_{j=1, j \neq q}^N \omega_{ijpq} s_{ij} \right) + \omega_{pqpq} (s'_{pq})^2 \right] \\ &\quad + \sum_{i=1, i \neq p}^N \sum_{j=1, j \neq q}^N \theta_{ij} s_{ij} + \theta_{pq} s'_{pq}. \end{aligned}$$

Restando ambas expresiones y simplificando se obtiene:

$$\begin{aligned} \Delta H &= -\frac{1}{2} \left[(s'_{pq} - s_{pq}) \left(\sum_{k=1, k \neq p}^N \sum_{l=1, l \neq q}^N \omega_{pqkl} s_{kl} + \sum_{i=1, i \neq p}^N \sum_{j=1, j \neq q}^N \omega_{ijpq} s_{ij} \right) + ((s'_{pq})^2 - s_{pq}^2) \omega_{pqpq} \right] + (s'_{pq} - s_{pq}) \theta_{pq} = \\ &= -\frac{1}{2} \left[2 (s'_{pq} - s_{pq}) \sum_{i=1, i \neq p}^N \sum_{j=1, j \neq q}^N \omega_{ijpq} s_{ij} + (s'_{pq} - s_{pq}) (s'_{pq} + s_{pq}) \omega_{pqpq} \right] + (s'_{pq} - s_{pq}) \theta_{pq}. \end{aligned}$$

Finalmente, usando que $s' = 1 - s$, se tiene:

$$\Delta H = (1 - 2s_{pq}) \left(\theta_{pq} - \frac{\omega_{pqpq}}{2} - \sum_{i=1, i \neq p}^N \sum_{j=1, j \neq q}^N \omega_{ijpq} s_{ij} \right) \quad (1)$$