CLASE 9

MODELO DE HOPFIELD: Parte II

Repasemos lo definido hasta aquí

Tenemos N neuronas binarias $S_i(t)$

$$\begin{cases} S_i(t) = 1 & \text{disparo} \\ S_i(t) = -1 & \text{reposo} \end{cases}$$

con $i = 1, 2, \ldots, N$.

$$\bar{S}(t) = \left(S_1(t), S_2(t), \ldots, S_N(t) \right)$$

La dinámica autónoma es

$$S_i(t+\Delta t) = \text{signo}\left(h_i(t) \right)$$

con $$h_i(t) = \sum_{j \neq i} W_{ij} S_j(t)$$

Recordemos que

- los umbrales son todos nulos, $\mu_i = 0, \forall i$.

- No se permiten auto interacciones $W_{ii} = 0 \ \forall i$

Elegimos una dinámica secuencial, actualizando una neurona por vez, ya sea en forma ordenada, $i = 1, 2, 3, \ldots, N$, o en forma aleatoria. Además:

$$\Delta t = \frac{1}{N}$$

con lo cual $\Delta t = 1$ después de N intentos de actualización.

Digo "intento" pues puede suceder que la dinámica no cambie el estado de la neurona.

- El input es la condición inicial $\{S_i(0)\}$ o $\bar{S}(0)$, y el output es el atractor, si existe $\{\xi_i^\nu\}$ o $\vec{\xi}^\nu$

$$\bar{S}(0) \longrightarrow \vec{\xi}^\nu$$

Pero pueden ser otros atractores indeseados.

Vimos que con una memoria (una única configuración almacenada), la regla era:

$$\begin{cases} W_{ij} = \frac{1}{N} \xi_i^1 \xi_j^1 & \text{si } i \neq j \\ W_{ii} = 0 & \forall i \\ \mu_i = 0 & \forall i \end{cases}$$

Vimos que funcionaba bien.

MUCHOS PATRONES

Supongamos ahora que queremos almacenar $p > 1$ configuraciones. Entonces Hopfield propuso

$$W_{ij} = \frac{1}{N} \sum_{\mu=1}^{p} \xi_i^\mu \xi_j^\mu$$

$$W_{ii} = 0$$

$$\mu_i = 0$$

La regla sigue siendo local, pues depende de i y j y no de otras neuronas. Es una regla tipo "Hebb".

Además es una regla aditiva: si ya almacenamos p memorias y queremos almacenar una más, la P+1, entonces

$$W_{ij}' \longrightarrow W_{ij} + \frac{1}{N}\xi_i^{P+1}\xi_j^{P+1}$$

ANÁLISIS DE ESTABILIDAD

Supongamos que preparamos a la red inicialmente en una de las p memorias almacenadas, a la que denotamos por la letra griega ν.

$$\vec{S}(0) = \vec{\xi}^{\nu}$$

$$S_i(0+\Delta t) = \text{signo}\left(h_i(0)\right) \quad \begin{matrix} +1 \\ -1 \end{matrix}$$

$$= \text{signo}\left(\sum_{j\neq i} W_{ij} S_j(0)\right)$$

$$= \text{signo}\left(\sum_{j=1}^{N} W_{ij} S_j(0) - W_{ii} S_i(0)\right)$$

Sumé y resté el término

$$w_{ii} = \frac{1}{N}\sum_{\mu=1}^{P} \xi_i^\mu \xi_i^\mu = \frac{1}{N}\sum_{\mu=1}^{P} 1 = \frac{P}{N}$$

$$S_i(t+\Delta t) = \text{signo}\left(\underbrace{\frac{1}{N}\sum_{j=1}^{N}\sum_{\mu=1}^{P} \xi_i^\mu \xi_j^\mu}_{w_{ij}} \overbrace{\xi_j^\nu}^{S_j(0)} - \cancel{\frac{P}{N}\overbrace{\xi_i^\nu}^{S_i(0)}}\right)$$

Supongamos por ahora $P \ll N$, con lo cual $\frac{P}{N} \ll 1$. Por ahora lo despreciamos.

$$S_i(t+\Delta t) = \text{signo}\left(\frac{1}{N}\sum_{j=1}^{N}\sum_{\mu\neq\nu}^{P} \xi_i^\mu \xi_j^\mu \xi_j^\nu + \frac{1}{N}\sum_{j=1}^{N} \xi_i^\nu \xi_j^\nu \xi_j^\nu\right)$$

$$= \text{signo}\left(\xi_i^\nu + \frac{1}{N}\sum_{j=1}^{N}\sum_{\mu\neq\nu} \xi_i^\mu \xi_j^\mu \xi_j^\nu\right)$$

$$= \text{signo}\left(\xi_i^\nu + \frac{(\xi_i^\nu)^2}{N}\sum_{j=1}^{N}\sum_{\mu\neq\nu} \xi_i^\mu \xi_j^\mu \xi_j^\nu\right)$$

$$= \text{signo}\left(\xi_i^\nu\left(1 + \frac{\xi_i^\nu}{N}\sum_{j=1}^{N}\sum_{\mu\neq\nu} \xi_i^\mu \xi_j^\mu \xi_j^\nu\right)\right)$$

$$S_i(t+\Delta t) = \text{signo}(\xi_i^\nu)\, \text{signo}\left(1 + \frac{\xi_i^\nu}{N} \sum_{j=1}^{N} \sum_{\mu\neq\nu} \xi_i^\mu \xi_j^\mu \xi_j^\nu\right)$$

Hasta ahora no hemos hecho ninguna suposición sobre la p configuraciones. A partir de ahora asumiremos que son aleatorias y que:

$$\xi_i^\mu = \begin{cases} +1 & \text{con probabilidad } \frac{1}{2} \\ -1 & \text{con probabilidad } \frac{1}{2} \end{cases}$$

Además por ser independientes tenemos que

$$P(\xi_1^1, \xi_2^1, \ldots \xi_N^1, \xi_1^2, \ldots\ldots, \xi_1^p, \ldots \xi_N^p) = \prod_{i=1}^{N} \prod_{\mu=1}^{p} P(\xi_i^\mu)$$

$$P(\xi_i^\mu = +1) = \frac{1}{2}$$

$$P(\xi_i^\mu = -1) = \frac{1}{2}$$

Nosotros queremos que $\vec{\xi}^{\nu}$ sea un atractor de la dinámica, o sea,

$$S(t+\Delta t) = \text{signo}(\xi_i^{\nu}) = \xi_i^{\nu}$$

para lo cual deseamos que

$$\text{signo}\left(1 + \xi_i^{\nu} \frac{1}{N} \sum_{j=1}^{N} \sum_{\mu \neq \nu} \xi_i^{\mu} \xi_j^{\mu} \xi_j^{\nu}\right) = 1$$

Para ello

$$\xi_i^{\nu} \frac{1}{N} \sum_{j=1}^{N} \sum_{\mu \neq \nu} \xi_i^{\mu} \xi_j^{\mu} \xi_j^{\nu}$$

debe ser mayor que -1.

Esta cantidad es una variable aleatoria, suma de $N \times (P-1)$ terminos que pueden valer $+1$ ó -1 con probabilidad $1/2$.

Definimos

$$C_i^\nu = - \xi_i^\nu \frac{1}{N} \sum_{j=1}^{N} \sum_{\mu \neq \nu} \xi_i^\mu \xi_j^\mu \xi_j^\nu$$

$\uparrow$

término crosstalk

$P(C_i^\nu) > 1$

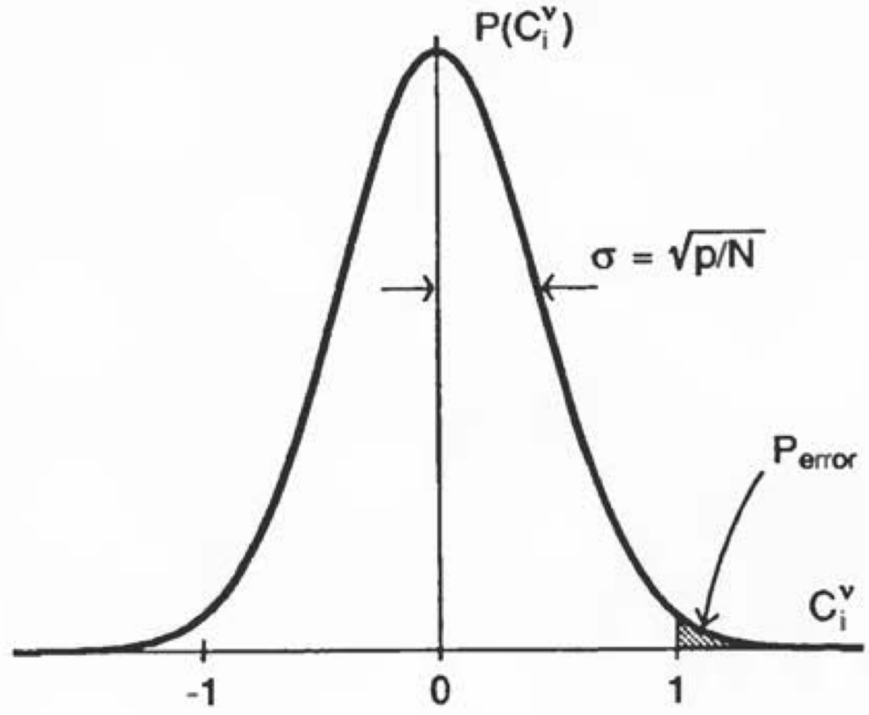


FIGURE 2.5 The distribution of values for the crosstalk C_i^ν given by (2.13). For p random patterns and N units this is a Gaussian with variance $\sigma^2 = p/N$. The shaded area is P_{error}, the probability of error per bit.

Noten que si $C_i^\nu > 1$ entonces

$$\text{signo}\left(1 - C_i^\nu\right) = -1$$

y de aquí

$$S_i(t+\Delta t) = -\xi_i^{\nu}$$

O sea, ese crosstalk término desestabiliza el patrón ν.

$$P_{error} = P_{rob}\left(C_i^{\nu} > 1\right).$$

C_i^{ν} tiene una distribución binomial y si $N*(P-1) \gtrsim 30$ podemos aproximarlo por una gaussiana, con la misma media que la binomial (cero) y con una varianza

$$\sigma^2 = \frac{P-1}{N} \simeq \frac{P}{N}$$

pues habíamos asumido $P \ll N$.

$$P(C_i^\nu) = \frac{1}{2\pi\sqrt{\frac{P}{N}}} e^{-\frac{(C_i^\nu)^2}{(2P/N)}}$$

¿Cuál es la probabilidad de que la neurona i se desestabilice si $\vec{S}(0) = \vec{\xi}^\nu$?

$$P_{error} = P(C_i^\nu > 1)$$

$$= \int_1^\infty P(x)\, dx$$

$$= \frac{1}{2\pi\sigma} \int_1^\infty e^{-\frac{x^2}{2\sigma^2}} dx$$

$$= \frac{1}{2}\left[1 - \operatorname{erf}\left(\frac{1}{\sqrt{2\sigma^2}}\right)\right]$$

$$\boxed{P_{error} = \frac{1}{2}\left[1 - \operatorname{erf}\left(\sqrt{\frac{N}{2p}}\right)\right]}$$

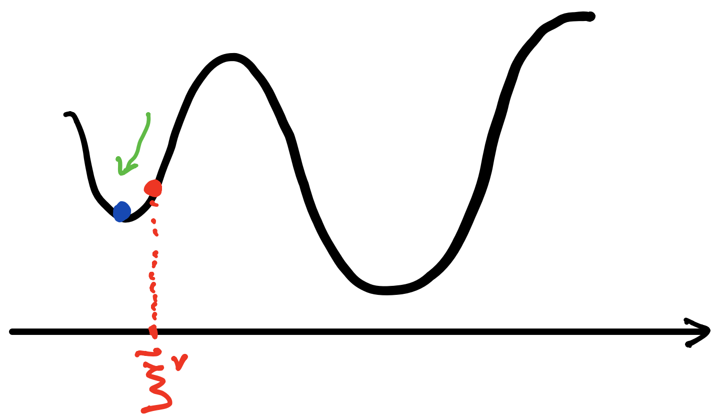
La función erf (error) se define como

$$\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2}\, dt.$$

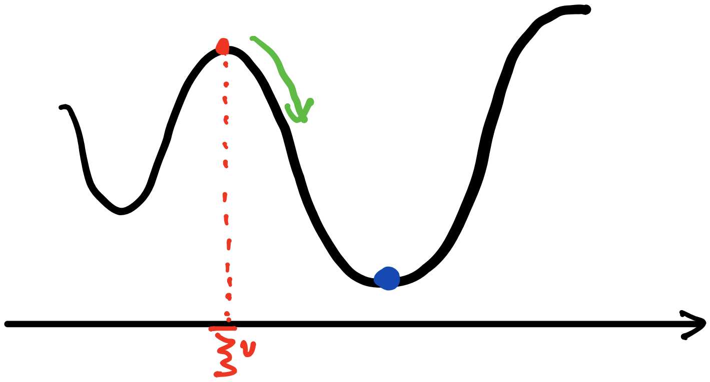
Podemos hacer una tabla que nos dice cual es el máximo p que podemos almacenar para una cierta tolerancia P_{error}.

P_{error}	p_{max}/N
0.001	0.105
0.0036	0.138
0.01	0.185
0.05	0.37
0.1	0.61

Pero esto solo habla de la estabilidad inicial. Si $P/N > 0$, nos dice que una fracción de las neuronas se apartarán de la memoria almacenada. Pero no sabemos si se "detiene" en una configuración cercana a $\vec{\xi}^{\nu}$ o si se va a una configuración alejada y descorrelacionada de $\vec{\xi}^{\nu}$.



Si pasa esto tenemos suerte.



Si pasa esto estamos perdidos.

Veremos que si $P < 0{,}138 \times N$, los atractores son muy parecidos a las memorias almacenadas, pero que para $P > 0{,}138 \times N$ se produce una avalancha que aleja al atractor de ξ^{ν}.

UN ENFOQUE ALTERNATIVO

Si deseamos que no haya errores, la cuenta anterior requiere que N crezca de forma tal que $\frac{P}{N} \longrightarrow 0$.

Miremos ahora el caso en que requerimos que para toda la configuración almacenada se cumpla que la probabilidad de recuperarla bien sea mayor a $0{,}99$.

$$(1 - P_{error})^{N} > \underline{\underline{0{,}99}}$$

$$(1 - P_{error})^N = \sum_{n=0}^{N} \binom{N}{n} 1^{N-n} P_{error}^{n}$$

$$\approx 1 - \binom{N}{1} P_{error}^{1} = 1 - N P_{error}$$

Con esto

$$(1 - P_{error})^N > 0,99 \Rightarrow 1 - N P_{error} > 0,99$$

$$(1 - 0,99) > N \cdot P_{error}$$

$$P_{error} < \frac{0,01}{N}$$

Esto nos acota P_{error}, de donde podemos ver la cota P_{MAX}.

Miremos P_{error}.

$$P_{error} = \frac{1}{2} \left[1 - \operatorname{erf}\left(\sqrt{\frac{N}{2P}} \right) \right]$$

Si $\quad x \to \infty \quad 1 - \text{erf}(x) \to \dfrac{e^{-x^2}}{x\sqrt{\pi}}$

$$\ln(P_{error}) = \ln\left[\frac{1}{2}\left[1 - \text{erf}\left(\sqrt{\frac{N}{2P}}\right)\right]\right]$$

$$\approx \ln\left[\frac{1}{2}\,\frac{e^{-\frac{N}{2P}}}{\sqrt{\frac{N}{2P}}\cdot\sqrt{\pi}}\right]$$

$$\approx \ln\left(\frac{1}{2}\right) + \ln\left(e^{-\frac{N}{2P}}\right) + \ln\left(\frac{1}{\sqrt{\frac{N}{2P}}}\right) + \ln\left(\frac{1}{\sqrt{\pi}}\right)$$

$$\approx -\ln(2) - \frac{N}{2P} - \frac{1}{2}\ln\left(\frac{N}{2P}\right) - \frac{1}{2}\ln(\pi)$$

donde hemos usado

$$\ln(a \cdot b) = \ln(a) + \ln(b)$$

$$\ln(a^z) = z\ln(a)$$

En particular

$$\ln\left(\frac{1}{a}\right) = -\ln(a)$$

Volvemos a la expresión

$$P_{error} < 0{,}01 \cdot N$$

$$\ln(P_{error}) < \ln(0.01) - \ln(N)$$

$$-\ln(2) - \frac{N}{2P} - \frac{1}{2}\ln(\pi) - \frac{1}{2}\ln\left(\frac{N}{2P}\right) < \ln(0{,}01) - \ln(N)$$

Si $N \longrightarrow \infty$ solo sobreviven pocos términos

$$\frac{N}{2P} > \ln(N)$$

Esto nos da una cota, con $N \gg 1$

$$P_{MAX} = \frac{N}{2\ln(N)} \quad \leftarrow \text{OJO}$$

Si $N = 10000$, $\ln(N) \approx 6{,}9$, y

$$P_{MAX} = 74$$

Si $N = 10^{11}$ $\quad P_{MAX} \sim 2 \times 10^{9}$

Ahora podemos pedir más. Pedimos que la red ande bien para los p patrones almacenados

$$(1 - P_{error})^{Np} > 0{,}99$$

$$1 - Np\, P_{error} > 0{,}99$$

$$P_{error} < \frac{0{,}01}{Np}$$

Haciendo la misma cuenta

$$P_{MAX} = \frac{N}{4 \ln(N)}$$

Nota Si tomamos los valores de las memorias ξ_i^μ como igualmente probables e independientes

$$\frac{1}{N} \sum_i \xi_i^\mu \xi_i^\nu = O\left(\frac{1}{\sqrt{N}}\right) \quad \text{si } \mu \neq \nu$$

Pero los podemos elegir también ortogonales

$$\sum_i \xi_i^\mu \xi_i^\nu = \delta_{\mu\nu}$$

En este caso se puede ver que no hay "ruido"

y $P_{MAX} = N$