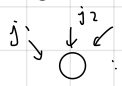


Redes neuronales artificiales

Las redes neuronales reales tienen dinámica muy compleja. Hay una línea que se trata en tomar la mayor simplificación posible y ver si se puede implementar alguna tarea computacional interesante.

El estado de la neurona a tiempo t viene dado por: $n_i(t) \in \{0, 1\}$ (está o no generando un potencial de acción)

Consideremos que i recibe inputs de neuronas j_1, j_2, \dots



$$\Rightarrow \text{input} = \sum_j w_{ij} n_j(t)$$

eficacia
sináptica
↓
aumento del
potencial
post sináptico

$$n_i(t + \Delta t) = \Theta \left(\sum_j w_{ij} n_j(t) - \theta_i \right) \rightarrow \text{si el input a tiempo } t \text{ pasa umbral, se tiene un spike a tiempo } t + \Delta t$$

Heaviside umbral

↓
no es lo que pasa realmente, por ejemplo rítmico que en HH se tiene un periodo refractario después de un spike → este no se considera aquí

Podemos tomar un cambio de variables $s_i = 2n_i - 1 \Rightarrow n_i = \frac{s_i + 1}{2}$

$$\begin{aligned} \Rightarrow s_i(t + \Delta t) &= 2 \Theta \left(\sum_j w_{ij} \left(\frac{s_j(t) + 1}{2} \right) - \theta_i \right) - 1 \\ &= 2 \Theta \left(\underbrace{\sum_j \frac{w_{ij}}{2} s_j(t)}_{w'_{ij} \text{ pesos nuevos}} + \underbrace{\frac{w_{ij}}{2} - \theta_i}_{-\theta'_i \text{ umbral nuevo}} \right) - 1 \end{aligned}$$

$$\begin{aligned} s_i(t + \Delta t) &= 2 \Theta \left(\sum_j w'_{ij} s_j(t) - \theta'_i \right) - 1 \\ &= \text{sgn} \left(\sum_j w'_{ij} s_j(t) - \theta'_i \right) \end{aligned}$$

$\Rightarrow 2 \Theta - 1 = \begin{cases} 1 & \text{arg} > 0 \\ -1 & \text{arg} < 0 \end{cases}$

Representación continua

Contornemos la cantidad de spikes en una ventana temporal

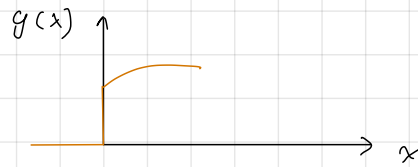
$0 \leq f_i(\tau) \leq \infty$: tasa de disparos promedio en una ventana temporal

↓
depende de en donde está la ventana
input umbral

$$f_i(\tau + \Delta\tau) = g\left(\sum_j w_{ij} f_j(\tau) - \theta_i\right)$$

↓
relación entre input y tasa de disparos
es una función continua.

una arquitectura
puede tener cuantas
de los dos
representaciones



distintos
modelos

arquitectura de la red: como están conectados los neuronas unas con otras. no solo cual está conectado con cual, sino que también la fuerza de la conexión

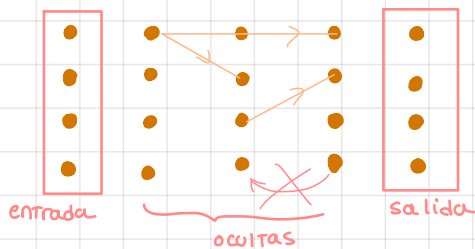
Tenemos dos tipos:

arquitectura feed forward

: w_{ij} está ordenado de manera que hay un flujo ordenado de información

↓
naturalmente aparecen capas, las conexiones van en un sentido

(cada pto es una neurona, por ej, el nivel visual está dispuesto en forma de capas)



cada neurona tiene un índice (índice de capa)

¿qué tipo de cosas siguen esta arquitectura?

- reconocimiento de objetos

aspecto semántico (*)

: reconocer que un ente es de un tipo a pesar de su variabilidad

aspecto espacial (*)

en que lugar está el objeto

(*) tienen potenciales en paralelo pero en regiones diferentes del cerebro

- sistema auditivo

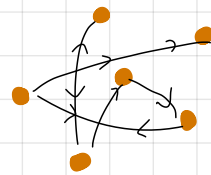
llega una onda de sonido

↓
puede descomponer superposiciones de componentes (por ej, dos personas hablando)

Para cada configuración de la capa de entrada, se tiene una diferente salida

arquitectura recurrente

↓
hay conexiones entre todas las neuronas



Sistema dinámico en el cual el input externo define la condición inicial. El sistema evoluciona y eventualmente llega a un estado final
↓ resultado de cálculo

No tiene una salida como tal, se miden los valores de las neuronas cuando se desee

Modelo de memoria asociativa

Preguntas

Redes Feed-Forward : ¿que modelos son posibles de implementar con una red feed-forward? **COMPLEJIDAD**

¿cuántos casos diferentes se pueden resolver con una red? **CAPACIDAD**

¿Cómo encontrar los parámetros de la red que resuelven un problema?

APRENDIZAJE
machine learning (aprendizaje automático)

objetivo final

¿Se puede aprender a partir de ejemplos?
GENERALIZACIÓN

ejemplo :

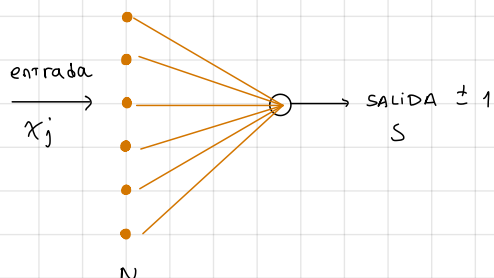
recibe imágenes de perros y gatos
(cada neurona recibe la intensidad de un pixel)

sale en 0 si se tiene un gato y en 1 si se tiene un perro

→ se dan 100 imágenes de perros y gatos

↓
después se testea con imágenes nuevas y se ve si da el resultado correcto

Feed-forward : caso sencillo no trivial - Perceptron



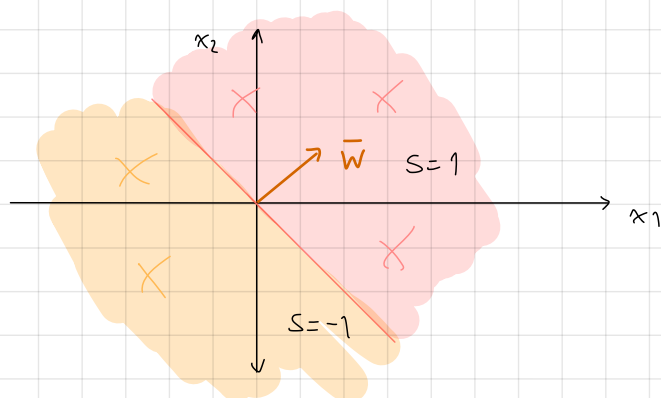
$$s = \text{sgm} \left(\sum_{i=1}^N w_i x_i - u \right)$$

↓
da 1 o -1
según el signo de ()

$$= \text{sgm} (\bar{w} \cdot \bar{x} - u)$$

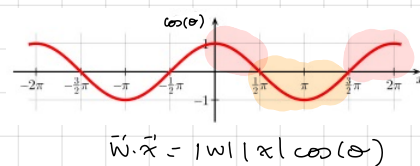
interpretación geométrica

entendidos (en n dimensional, lo supongamos 2D)

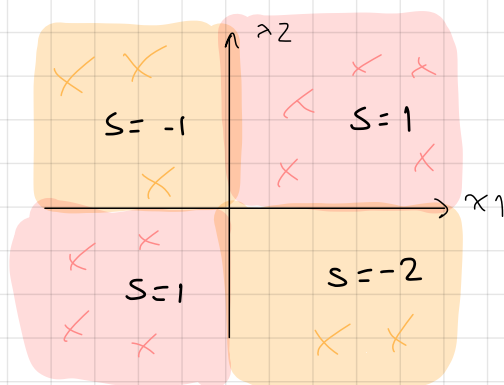


$$\dim(\bar{w}) = \dim(\bar{x})$$

supongamos $y = 0$



no se pueden encontrar \bar{w} y \bar{x} tales que:



no es linealmente separable

Un perceptron solo puede resolver problemas linealmente separables

veamos cual es el algoritmo del perceptron

j : índice de neurona

x_j^u $j = 1, \dots, N$

$u = 1, \dots, P$

$S^u = \pm 1$

$u = 1, \dots, P$

s : salida deseada

¿quién encuentra?

$$W_j / S^u = \text{sgm} \left(\sum_{j=1}^N w_j x_j^u \right)$$

$$= \text{sg} (\bar{w} \cdot \bar{x}^u)$$

↓
de
cruces
"imágenes"
a
analizar

- 1) Primero elegin w_{ij} aleatorios y pequeños
Tomo en x^u (u fijo)

2) Calcular solidos como

$$O^u = \text{sgm} \left(\sum_{j=1}^n w_j x_j^u \right) \text{ para un } u$$

↓
solido
debe
(no se
si es
la decodir)

3) Si $O^u = S^u$ tomar otro u y volver a (2)

4) Si $O^u \neq S^u \Rightarrow$ tomar $\Delta w_j = \underbrace{\eta}_{\substack{\text{para el} \\ \text{aprendizaje} \\ (0,1)}} (S^u - O^u) \underbrace{x_j^u}_{\text{entrada}}$

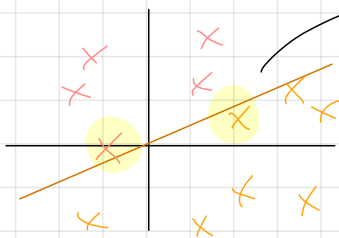
si pose por los p
potencia con el
resultado correcto
↓
termino

vale 20-2

modifica w_j en
direccion de la
entrada.

5) Volver a (2)

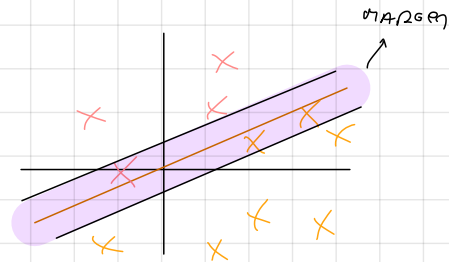
El algoritmo no es sensible a puntos que están muy cerca del hiperplano



no es sensible a pequeñas fluctuaciones

\Rightarrow Queremos tener un margen de seguridad:

$$\sum_j w_j x_j^u = \vec{w} \cdot \vec{x}^u \begin{matrix} > \\ < \end{matrix} N \underbrace{K}_{\substack{\text{factor} \\ \text{de} \\ \text{seguridad}}}$$



$$\sum_j w_j x_j^u \stackrel{(*)}{S^u} = \vec{w} \cdot \vec{x}^u S^u > NK \rightarrow \text{fuera del margen de seguridad} \rightarrow \text{error}$$

algoritmo

1) elegir w_j inicial aleatorio

2) Calcular $O^u = \text{sgm}(\vec{w} \cdot \vec{x}^u)$ para un u

pendiente de seg. O^u

$$3) \Delta w_j = \eta \left(\underbrace{NK}_{\text{factor de seguridad}} - \sum_j w_j x_j^u s_j \right) S^u x_j^u$$

si $(*) < NK \Rightarrow$ estoy dentro del margen de seguridad \Rightarrow puede estar en el lado correcto

$$\Rightarrow \vec{w} \rightarrow \vec{w} + \Delta \vec{w}$$

si estamos en $(*) > NK \rightarrow$ no estoy en el margen $\Rightarrow (\Delta w = 0)$ por la heaviside

4) Si paso por todos los u sin modificar \bar{w} , termine

Si este problema tiene solución \rightarrow converge en t finito