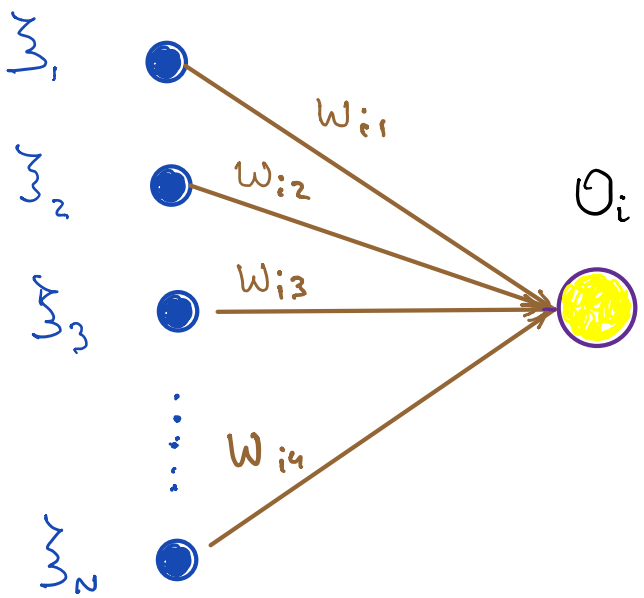
PERCEPTRON SIMPLE CON UNIDADES LINEALES

Volvamos a analizar un perceptron de una única capa reparado en M perceptrones simples. La diferencia con el caso anterior, donde teníamos neuronas binarias de salida, es que ahora la neurona de salida es lineal



N neuronas de entrada

1 neurona i de salida

Los elementos del vector $\vec{\xi} = (\xi_1, \xi_2, \ldots, \xi_N)$ son números reales. La salida es

$$O_i = h_i = \sum_k w_{ik} \xi_k$$

Notem que la función de activación es

$$g(h) = h$$

Suposem novament que tenim P entrades $\mu = 1, 2, \dots, P$ etiquetades, o sia, per les quals coneixem la sortida

Si entra ξ^μ ha de sortir ζ_i^μ.

Separem la que surt, que és O_i^μ

$$O_i^\mu = h_i = \sum_k w_{ik} \xi_k^\mu$$

i la que ha de sortir, ζ_i^μ.

Nota: Els valors desitjats podrien ser discrets
DISCUTIR

Este problema té una solució explícita, anomenada PSEUDO-INVERSA. En este cas calculem la matriu

$$Q_{\mu\nu} = \frac{1}{N} \sum_k \xi_k^\mu \xi_k^\nu$$

que defineix la superposició entre els elements μ i ν del conjunt d'entrenament.

Si $\mathbf{Q}$ es invertible, podemos definir los acoplamientos como

$$W_{ik} = \frac{1}{N} \sum_{\mu} \sum_{\nu} \xi_i^{\mu} (\mathbf{Q}^{-1})_{\mu\nu} \xi_k^{\nu}$$

Para esto los p vectores $\vec{\xi}^{\mu}$ deben ser linealmente independientes

Si no lo son, no existe solución. Esta es una condición que aparece en todas las redes neuronales lineales. Esto las hace poco interesantes.

Para que sean linealmente independientes debe ser p menor o igual a N.

$$P \leq N$$

El método de descenso por el gradiente

Supongamos que mostramos a la red los p inputs y para cada uno de ellos calculamos

$$E(\vec{w}) = \frac{1}{2} \sum_{\mu} \sum_{i} (\xi_i^{\mu} - O_i^{\mu})^2$$

Noten que si $O_i^{\mu} = \xi_i^{\mu}$ (si la red resuelve bien

un ejemplo del conjunto de entrenamiento, ese ejemplo no contribuye a la suma, pues

$$\zeta_i^\mu = O_i^\mu$$

resultado deseado

resultado real.

Esta función $E(\bar{w})$ se llama función coste. Es función de los N elementos de $\bar{w}$ y no es función de los ζ_i^μ, que serán considerados parámetros dados (fijos). Es un concepto que nos remite a la idea de función energía de los modelos de Hopfield, pero no es exactamente lo mismo. DISCUTIR

Por construcción $E(\bar{w}) \geq 0$ (es suma de NP términos mayores o iguales a cero.

Si $E(\bar{w}) = 0$, entonces la red resolvió exactamente el problema, pues significa que $\zeta_i^\mu = O_i^\mu$ para todo $\mu = 1, 2, \ldots, P$.

Por otro lado, cuanto menor es $E(\bar{w})$, menos sera el error o el costo, pues más se parecerán los $\hat{y}_i$ a los O_i.

Nota: Dada una función

$$f: \mathbb{R}^n \to \mathbb{R} \quad \text{(varias variables)}$$

su gradiente en un punto $\bar{x} \in D(f)$ (dominio) se define como un campo vectorial, o sea, un vector en $\mathbb{R}^n$ asignado a cada punto del dominio el cual pertenece a $\mathbb{R}^n$. Se lo denota ∇f y se define como:

$$\nabla f(\bar{x}) = \left(\frac{\partial f(\bar{x})}{\partial x_1}, \frac{\partial f(\bar{x})}{\partial x_2}, \dots, \frac{\partial f(\bar{x})}{\partial x_n} \right)$$

O sea, a cada vector $\bar{x}$ asignamos un vector ∇f (esto se llama campo vectorial)

Generaliza el concepto de derivada en el caso de funciones de una variable $f: \mathbb{R} \to \mathbb{R}$ y geométricamente se puede ver que ∇f apunta en la dirección en la cual f

varía más rápidamente en el punto $\bar{x}$.

Volvamos a nuestro problema: el perceptron simple lineal.

$E(\bar{w}) = E(w_1, w_2, \ldots, w_N)$ es una función

$$E: \mathbb{R}^n \longrightarrow \mathbb{R} \quad \text{(campo escalar)}$$

Como es bien conocido podemos definir su gradiente en el vector $\bar{w}$

$$\nabla E = \left(\frac{\partial E}{\partial w_1}, \frac{\partial E}{\partial w_2}, \ldots, \frac{\partial E}{\partial w_N} \right)$$

Este vector ∇E nos indica hacia donde crece más rapido E al variar $\bar{w}$. Entonces

$$-\nabla E$$

indica la dirección hacia donde decrece más rápido $E(w)$. Como buscamos un $\bar{w}^{optimo}$ para el cual $E(\bar{w}^{optimo}) = 0$ construimos la siguiente regla de aprendizaje

$$w_{ik}^{\text{nuevo}} = w_{ik}^{\text{viejo}} + \Delta w_{ik}$$

con $\Delta w_{ik} = -\eta \frac{\partial E}{\partial w_{ik}}$ $\quad \vec{w}^{\text{nuevo}} = \vec{w}^{\text{viejo}} + \vec{\Delta w}$

O sea, Δw_{ik} apunta hacia la dirección de más rápido descenso de $E(\vec{w})$.

$$\vec{\Delta w} = -\eta \nabla E(\vec{w})$$

Hagamos algunos cálculos

$$\frac{\partial E(\vec{w})}{\partial w_{ik}} = \frac{\partial}{\partial w_{ik}} \left(\frac{1}{2} \sum_{\nu} \sum_{j} \left(\zeta_j^\nu - O_j^\nu \right)^2 \right)$$

$$= \frac{\partial}{\partial w_{ik}} \left(\frac{1}{2} \sum_{\nu} \sum_{j} \left(\zeta_j^\nu - \sum_{\ell} w_{j\ell} \xi_\ell^\nu \right)^2 \right)$$

$$= -\frac{1}{2} \sum_{\nu} \left(\zeta_i^\mu - O_i^\mu \right) \xi_k^\mu \quad \text{batch}$$

Entonces

$$\Delta w_{ik} = \eta \left(\zeta_i^\mu - O_i^\mu \right) \xi_k^\mu = \eta \, \delta_i^\mu \, \xi_k^\mu \quad \underline{\text{online}}$$

con $\delta_i^\mu = \left(\zeta_i^\mu - O_i^\mu \right)$

Esta regla se conoce como

- Regla Delta (Delta rule)
- Regla ADALINE
- Regla Widrow-Hoff
- Regla LMS (Least mean square)

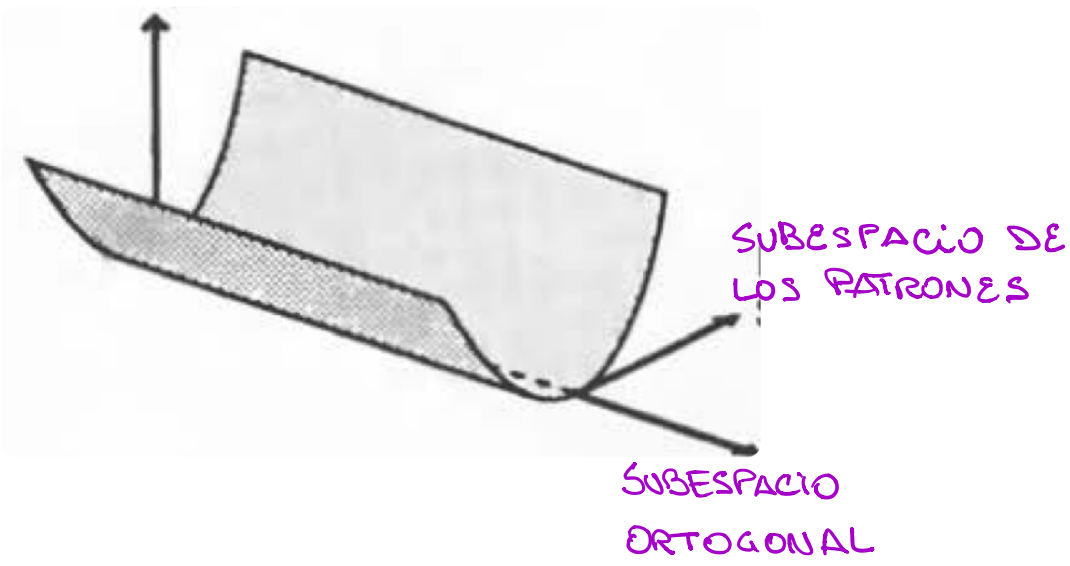
ADALINE viene de ADAptative Linear Neuron y fue definida por primera vez en 1960 por Bernard Widrow y su estudiante de doctorado Ted Hoff en Standford University.

El parámetro η se llama "razón de aprendizaje" y ya había aparecido en el perceptrón binario. Veremos que es "FUNDAMENTAL" y que nos acompañará siempre. Digamos por ahora que debe ser "pequeño" sin abundar mucho más.

Noten que es una regla hebbiana, derivada de un gradiente.

En el caso binario el equivalente a δ_i^H era ± 2 o 0. Ahora toma un continuo de valores.

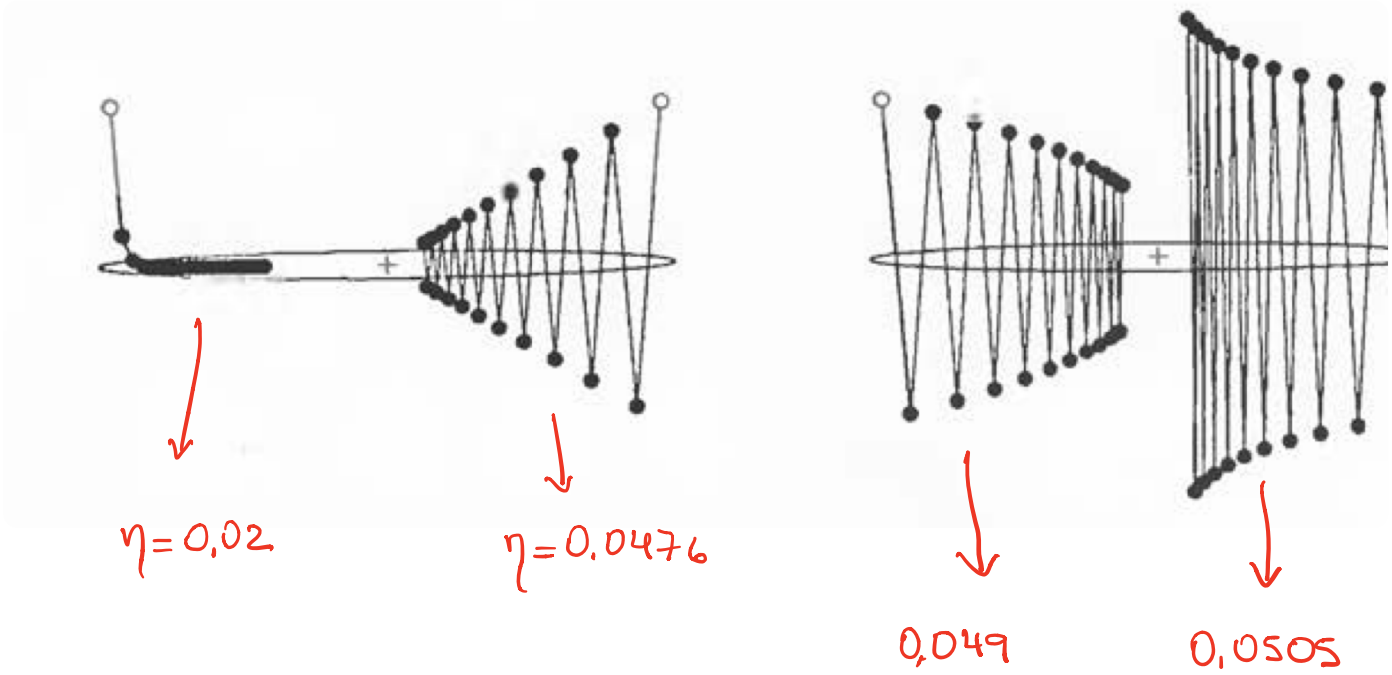
E es una función de $\bar{w}$ y tiene un mínimo $E=0$.
Supongamos que existe solución, entonces $p \leq N$.
Luego los p vectores definen un subespacio.
En el subespacio ortogonal al subespacio expandido la función debe valer también cero, pues no depende de esas dimensiones. El siguiente es un esquema de la forma de $E(\bar{w})$ en $\mathbb{R}^N$.



Tenemos entonces una estructura tipo canaleta.

Esto nos dice que hay muchas soluciones, que viven en el fondo de la canaleta. En el subespacio expandido por los patrones de entrenamiento la regla ADALINE decrece la función costo. Por eso el valor de η se vuelve crucial.

No vamos a demostrar la convergencia



Noten cuan sensible es el método al valor de η.

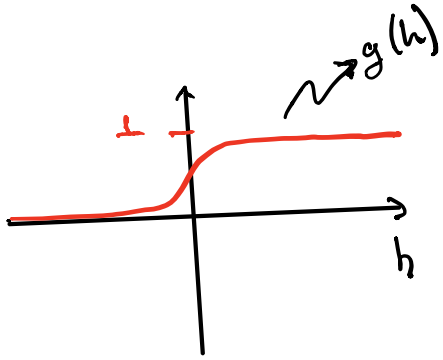
PERCEPTRON SIMPLE CON UNIDADES NO LINEALES

Ahora haremos la misma cuenta en el caso en el cual la neurona de salida es no lineal

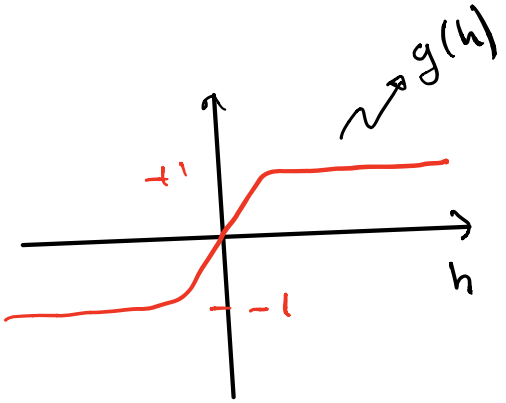
$$O_i = g(h_i)$$

Dos casos típicos son

$$g(h) = \frac{1}{1+e^{-\beta h 2}}$$



$$g(h) = \tanh(\beta h)$$



Ahora la cuenta es muy parecida.

$$E(\vec{w}) = \frac{1}{2} \sum_{\mu i} (\zeta_i^\mu - O_i^\mu)^2$$

$$E(\vec{w}) = \frac{1}{2} \sum_{\mu i} \left[\zeta_i^\mu - g\left(\sum_k w_{ik} \xi_k^\mu \right) \right]^2$$

$$\frac{\partial E}{\partial w_{ik}} = - \sum \left[\zeta_i^\mu - g(h_i^\mu) \right] g'(h_i^\mu) \, \xi_k^\mu$$

Definimos $\quad \delta_i^\mu = (\zeta_i^\mu - O_i^\mu) \, g'(h_i^\mu)$

de forma que

$$\begin{aligned} w_{ik}^{\text{nuevo}} &= w_{ik}^{\text{viejo}} + \Delta w_{ik} \\ &= w_{ik}^{\text{viejo}} - \eta \frac{\partial E}{\partial w_{ik}} \\ &= w_{ik}^{\text{viejo}} + \eta \, (\zeta_i^\mu - O_i^\mu) \, g'(h_i^\mu) \, \xi_i^\mu \end{aligned}$$

$$\boxed{w_{ik}^{\text{nuevo}} = w_{ik}^{\text{viejo}} + \eta \, \delta_i^\mu \, \xi_i^\mu}$$

Un problema es la necesidad de evaluar $g(h_i^\mu)$ que a diferencia del caso lineal tiene un alto costo en tiempo de cálculo. Además hay que evaluar $g'(h_i^\mu)$.

Si $g(h) = \tanh(\beta h)$, entonces $g'(h) = \beta(1-g^2)$

Si $g(h) = \dfrac{1}{1+e^{-2\beta h}}$, entonces $g'(h) = 2\beta g(1-g)$

El hecho de que g' sea función de g facilita los cosas.

Veremos más adelante que estas expresiones para g y g' son responsables de la dificultad de construir redes de muchas capas (o sea, redes profundas).

Es importante saber que las neuronas no lineales de salida se usan incluso para problemas binarios. DISCUTIR. Esto permite pensar en términos de descenso por el gradiente.

La cantidad de problemas que se pueden resolver con redes de una capa es pequeña.

Las condiciones para la existencia de solución son los mismos. Esto es fácil de verificar si se reemplaza la salida deseada ξ_i^μ por $g^{-1}(\xi_i^\mu)$. Lo que cambia es el "paisaje" de la función costo $E(\bar{w})$. Pueden existir mínimos locales.

OTRAS FUNCIONES DE COSTO

$$E(\bar{w}) = \sum_i \sum_\mu \left[\frac{1}{2}(1+\xi_i^\mu) \log\left(\frac{1+\xi_i^\mu}{1+O_i^\mu}\right) + \frac{1}{2}(1-\xi_i^\mu) \log\left(\frac{1-\xi_i^\mu}{1-O_i^\mu}\right) \right]$$

Sara Solla, 1988 ENTROPÍA RELATIVA

Si $\xi_i^\mu = O_i^\mu$ para todo μ y todo i, $E(\bar{w})$ es cero.

Por otro lado, con los $g(h)$ usados $(-1 < g(h) < +1)$ es una función definida positiva.

Ejercicio: mostrar que $\Delta w_{ik} = \eta \delta_i^\mu \xi_k^\mu$ pero con

$$\delta_i^\mu = \beta(\xi_i^\mu - O_i^\mu)$$

Esto nos ahorra tener que calcular g'

NEURONAS ESTOCÁSTICAS

Ahora que tenemos una función $E(\bar{w})$ que minimizar la cual tiene un mínimo global en $E(\bar{w})=0$ pero puede tener muchos mínimos locales, podemos introducir ruidos aleatorios en la salida, como hicimos con la red de Hopfield.

Para ello calculamos $h_i^\mu = \sum_k w_{ik} \xi_i^k$, y con h_i^μ definimos una probabilidad

$$\text{Prob}\left(S_i^\mu = +1\right) = \frac{1}{1 + e^{-2\beta h_i^\mu}}$$

$$\text{Prob}\left(S_i^\mu = -1\right) = 1 - \text{Prob}\left(S_i^\mu = +1\right) = \frac{1}{1 + e^{2\beta h_i^\mu}}$$

Nuevamente tenemos que

$$\langle S_i^\mu \rangle = (+1)\cdot \text{Prob}\left(S_i^\mu = +1\right) + (-1)\,\text{Prob}\left(S_i^\mu = -1\right)$$

$$= \frac{1}{1 + e^{-2\beta h_i^\mu}} - \frac{1}{1 + e^{2\beta h_i^\mu}}$$

$$= \frac{1}{e^{-\beta h_i^\mu}\left(e^{\beta h_i^\mu} + e^{-\beta h_i^\mu}\right)} - \frac{1}{e^{\beta h_i^\mu}\left(e^{-\beta h_i^\mu} + e^{\beta h_i^\mu}\right)}$$

$$= \frac{\left(e^{\beta h_i^\mu} - e^{-\beta h_i^\mu}\right)}{\left(e^{\beta h_i^\mu} + e^{-\beta h_i^\mu}\right)}$$

$$= \tanh\left(\beta h_i^\mu\right)$$

o

$$\boxed{\langle S_i^\mu \rangle = \tanh\left(\beta \sum_k w_{ik} \xi_k^\mu\right)}$$

Podemos calcular $\langle S_i^\mu \rangle$ en una simulación y el valor estimado lo podemos usar en la actualización de los pesos sinápticos

$$\Delta w_{ik} = \eta \delta_i^\mu \xi_k^\mu$$

$$\delta_i^\mu = \left(\xi_i^\mu - \langle S_i^\mu \rangle\right)$$

Se puede mostrar (está en el libro) que esta regla decrece el promedio de la función error

$$E = \frac{1}{2} \sum_i \sum_\mu \left(\xi_i^\mu - S_i^\mu\right)^2$$