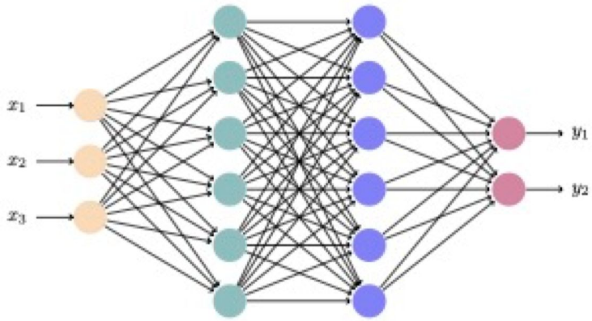
Fundamentos Matemáticos del Aprendizaje Profundo

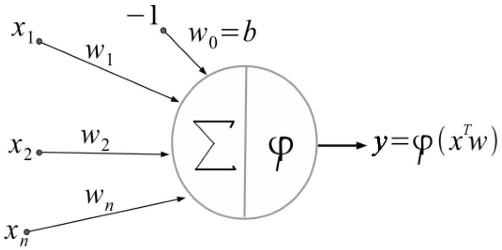
1er cuat. 2025
Clase 6

Repaso

Una red neuronal está compuesta de **neuronas**



DEFINICIÓN 5.1.1. Una neurona abstracta es un cuádruple $(\mathbf{x}, \mathbf{w}, \varphi, y)$, donde $\mathbf{x} = (x_0, \ldots, x_n)$ es el vector de entrada, $\mathbf{w} = (w_0, \ldots, w_n)$ es el vector de pesos, con $x_0 = -1$ y $w_0 = b$, el sesgo, y φ es la función de activación que define la función de salida $y = \varphi(\mathbf{w} \cdot \mathbf{x}) = \varphi(\sum_{i=0}^{n} w_i x_i)$.



Repaso (2)

Perceptrón: $$y = \varphi(\mathbf{w} \cdot \mathbf{x}) = \begin{cases} 1, & \text{si } \sum_{i=1}^{n} w_i x_i < b \\ 0, & \text{si } \sum_{i=1}^{n} w_i x_i \geq b. \end{cases}$$

Implementan AND y OR, dividen clusters “linealmente separables”

Sigmoide: $$y = \sigma(\mathbf{w} \cdot \mathbf{x} - b) = \frac{1}{1 + e^{-\mathbf{w} \cdot \mathbf{x} + b}}$$

Implementan “regresión logística” (Ej: probabilidad de default).

Neurona sigmoide como clasificador binario

Tenemos dos grupos de puntos en el plano:

$$\mathcal{G}_1 = \{\text{puntos negros}\}$$

$$\mathcal{G}_2 = \{\text{puntos blancos}\}$$

Definimos la función objetivo:

$$z(\mathbf{x}) = \begin{cases} 1, & \text{si } \mathbf{x} \in \mathcal{G}_1 \\ -1, & \text{si } \mathbf{x} \in \mathcal{G}_2. \end{cases}$$

Asumamos que los grupos son separables linealmente por la recta

$$w_1 x_1 + w_2 x_2 - b = 0$$

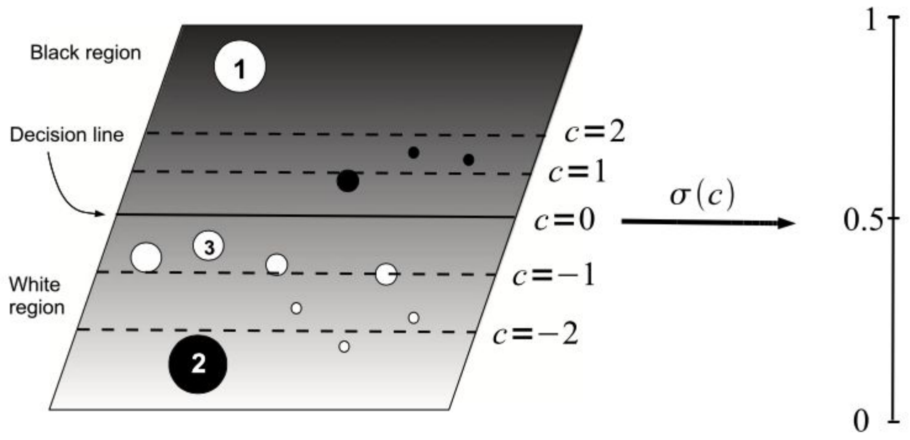
Objetivo: Ajustar los parámetros para que la recta represente la mejor partición de los datos

Línea de decisión

$$w_1x_1 + w_2x_2 - b = 0$$

Partimos el plano en rectas paralelas a la línea de decisión

$$\{w_1x_1 + w_2x_2 - b = c \colon c \in \mathbb{R}\}$$



La información que provee cada punto: $H(\mathbf{x}) = -\ln\sigma(z(\mathbf{x})(\mathbf{w}\cdot\mathbf{x} - b))$

Función de costo

$$E(\mathbf{w}, b) = \sum_{i=1}^{n} H(\mathbf{x}_i) = -\sum_{i=1}^{n} \ln\sigma(z_i(\mathbf{w}\cdot\mathbf{x}_i - b))$$

Minimización del costo

$$(\mathbf{w}^*, b^*) = \arg\min_{(\mathbf{w},b)} E(\mathbf{w}, b) = \arg\max_{(\mathbf{w},b)} \sum_{i=1}^{n} \ln \sigma(z_i(\mathbf{w} \cdot \mathbf{x}_i - b))$$

… asumiendo independencia …

$$= \arg\max_{(\mathbf{w},b)} \mathbb{P}_{\mathbf{w},b}(z_1, \ldots, z_n | \mathbf{x}_1, \ldots, \mathbf{x}_n)$$

$\mathbb{P}_{\mathbf{w},b}(z|\mathbf{x})$ probabilidad de ser de tipo z dado que las coordenadas son **x**

Descenso de gradiente

$$E(\mathbf{w}, b) = \sum_{i=1}^{n} H(\mathbf{x}_i) = -\sum_{i=1}^{n} \ln \sigma(z_i(\mathbf{w} \cdot \mathbf{x}_i - b)) \qquad \nabla E = (\nabla_{\mathbf{w}} E, \partial_b E)$$

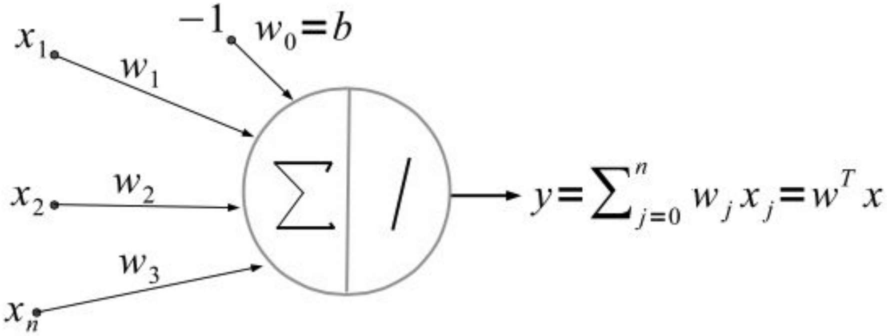
$$\nabla_{\mathbf{w}} E = -\sum_{i=1}^{n} \frac{z_i \mathbf{x}_i}{1 + e^{-(z_i(\mathbf{w} \cdot \mathbf{x}_i - b))}}$$

$$\partial_b E = \sum_{i=1}^{n} \frac{z_i}{1 + e^{-(z_i(\mathbf{w} \cdot \mathbf{x}_i - b))}}.$$

$$\mathbf{w}_{j+1} = \mathbf{w}_j - \eta \nabla_{\mathbf{w}} E(\mathbf{w}_j, b_j)$$
$$b_{j+1} = b_j - \eta \partial_b E(\mathbf{w}_j, b_j),$$

$\eta > 0$ tasa de aprendizaje

Neurona lineal



Asumamos que las entradas son aleatorias

$$\mathbf{X} = (X_0, X_1, \ldots, X_n), \ \mathbf{w} = (w_0, w_1, \ldots, w_n), \qquad X_0 = -1, \ w_0 = b$$

La salida es una v.a. unidimensional $Y = \sum_{j=0}^{n} w_j X_j = \mathbf{w} \cdot \mathbf{X}$

El costo es el error cuadrático medio: $\mathbf{w}^* = \arg\min_{\mathbf{w}} \mathbb{E}[(Z - Y)^2]$

Cálculo de $\mathbf{w}^*$

Para este caso sencillo es posible calcular el valor óptimo de los pesos exactamente (al menos en papel y lápiz)

$$\begin{aligned}\mathbb{E}[(Z-Y)^2] &= \mathbb{E}[Z^2 - 2ZY + Y^2] = \mathbb{E}[Z^2 - 2Z\mathbf{w}\cdot\mathbf{X} + (\mathbf{w}\cdot\mathbf{X})^2] \\ &= \mathbb{E}[Z^2] - 2\mathbb{E}[Z\mathbf{X}]\cdot\mathbf{w} + \mathbb{E}[(\mathbf{w}\mathbf{X}^t)(\mathbf{X}\mathbf{w}^t)] \\ &= \mathbb{E}[Z^2] - 2\mathbb{E}[Z\mathbf{X}]\cdot\mathbf{w} + \mathbf{w}\mathbb{E}[\mathbf{X}^t\mathbf{X}]\mathbf{w}^t \\ &= c - 2\mathbf{b}\cdot\mathbf{w} + \mathbf{w}A\mathbf{w}^t,\end{aligned}$$

$c = \mathbb{E}[Z^2]$ segundo momento del objetivo

$\mathbf{b} = \mathbb{E}[Z\mathbf{X}]$ correlación entrada-salida

Cálculo de $\mathbf{w}^*$

$$A = \mathbb{E}[\mathbf{X}^t\mathbf{X}] = \begin{pmatrix} \mathbb{E}[X_0X_0] & \mathbb{E}[X_0X_1] & \cdots & \mathbb{E}[X_0X_n] \\ \mathbb{E}[X_1X_0] & \mathbb{E}[X_1X_1] & \cdots & \mathbb{E}[X_1X_n] \\ \vdots & \vdots & \ddots & \vdots \\ \mathbb{E}[X_nX_0] & \mathbb{E}[X_nX_1] & \cdots & \mathbb{E}[X_nX_n] \end{pmatrix}$$

auto-correlación de entradas

Asumimos *entradas coherentes* (i.e. *A* es inversible, *A>0*)

Obtenemos entonces $\mathbf{w}^* = A^{-1}\mathbf{b}$

Problema: Cálculo de A^{-1}

Descenso de gradiente

Costo $$\mathbb{E}[(Z-Y)^2] = \xi(\mathbf{w}) = c - 2\mathbf{b}\cdot\mathbf{w} + \mathbf{w}A\mathbf{w}^t$$

Gradiente $$\nabla\xi(\mathbf{w}) = 2A\mathbf{w} - 2\mathbf{b}$$

Descenso $$\mathbf{w}_{k+1} = \mathbf{w}_k - \eta\nabla\xi(\mathbf{w}_k)$$

$$= \underbrace{(\mathbb{I}_n - 2\eta A)}_{M}\mathbf{w}_k + 2\eta\mathbf{b}$$

¿Este método es convergente?

Convergencia del método

$$\mathbf{w}_{k+1} = M\mathbf{w}_k + 2\eta\mathbf{b} \qquad \text{con} \qquad M = \mathbb{I}_n - 2\eta A$$

Iterando $\mathbf{w}_k = M^k\mathbf{w}_0 + 2\eta(\mathbb{I}_n + M + \cdots + M^{k-1})\mathbf{b}$

Ahora $\underbrace{(\mathbb{I}_n - M)}_{2\eta A}(\mathbb{I}_n + M + \cdots + M^{k-1}) = (\mathbb{I}_n - M^k)$

de donde obtenemos $\mathbf{w}_k = M^k\mathbf{w}_0 + (\mathbb{I}_n - M^k)A^{-1}\mathbf{b}$

Si asumimos $M^k \to \mathbb{O}_n \ (k \to \infty)$ entonces $\mathbf{w}_k \to A^{-1}\mathbf{b} = \mathbf{w}^* \ (k \to \infty)$

Convergencia del método (2)

Debemos verificar que $M^k \to \mathbb{O}_n \ (k \to \infty)$

equivalentemente $|\lambda_i| < 1$ para todo autovalor de $M = \mathbb{I}_n - 2\eta A$

Observación:

$$\alpha_i = \frac{1 - \lambda_i}{2\eta} \Leftrightarrow \lambda_i = 1 - 2\eta\alpha_i$$

$$\lambda_i \text{ autovalor de } M \Leftrightarrow \alpha_i \text{ autovalor de } A$$

Luego $A > 0 \Rightarrow \alpha_i > 0 \Rightarrow \lambda_i < 1$ y $0 < \eta < \dfrac{1}{2 \max_{1 \leq i \leq n} \alpha_i} \Rightarrow \lambda_i > 0$

Neurona de entrada continua

Entrada: $x \in [a, b]$

Pesos: $w(dx)$ (medida sobre el intervalo)

Salida: $y = \sigma \left(\int_a^b x \, dw(x) \right)$ σ función de activación

Ej: Si *X* v.a. y *w* es la medida de distribución de *X*, entonces $y = \sigma(\mathbb{E}[X])$

Neurona con medida discreta

$$\mu(A) = \sum_{x_j \in E} w_j \delta_{x_j}(A), \qquad E = \{x_1, \ldots, x_n\}$$

$$y = \sigma\left(\int_a^b x\, d\mu(x)\right) = \sigma\left(\sum_{j=1}^{n} w_j x_j\right)$$

Neurona clásica. Se ajusta la función de error para aprender. P.ej.

$$F(w) = \frac{1}{2}\left(\sigma\left(\sum_{j=1}^{n} w_j x_j\right) - z\right)^2$$

Neurona con medida continua

$$d\mu(x) = p(x)dx \qquad y = \sigma\left(\int_0^1 x\, d\mu(x)\right) = \sigma\left(\int_0^1 xp(x)\, dx\right)$$

$p(x)$ función de peso

Aprendizaje: Buscamos minimizar el funcional de error. P.ej.

$$F(p) = \frac{1}{2}\left(\sigma\left(\int_0^1 xp(x)\, dx\right) - z\right)^2$$