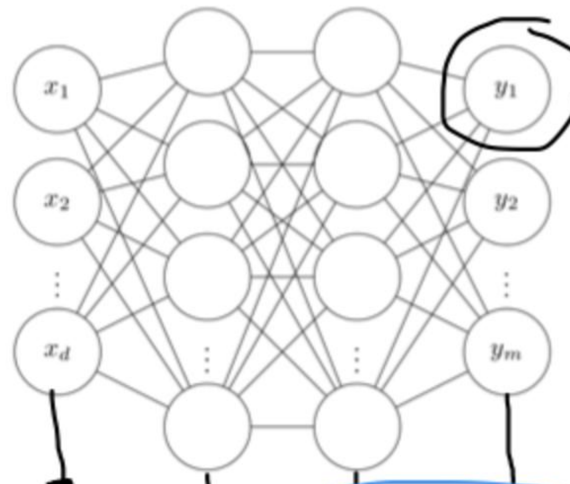
29/09/21

Resumen del lunes

Redes neuronales unidireccionales

4

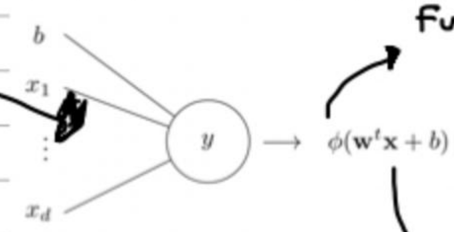


rasgos

1° capa 2° capa Capa de salida

f :

1 Neurona / Unidad



Función de

- Perceptrón Rosenblatt

$$\phi(z) = \begin{cases} +1 & \\ -1 & \text{e.o.c.} \end{cases}$$

2

- Adaline

$$\phi(z) = z$$

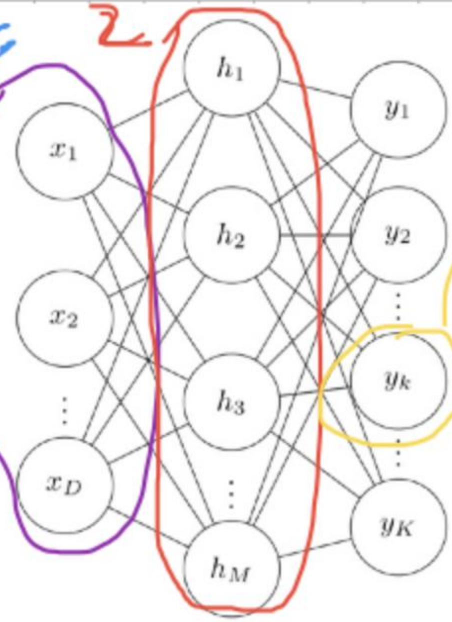
3 Arquitectura

- Topología de la red
  - # { • Capas colección de neuronas por niveles
  - • Neuronas por capa
- Funciones de activación

6 Analogía

5

De la neurona al perceptrón multicapa unidireccional completamente conectado



$$y_k = \phi\left(\sum_{j=1}^{M} w_{kj}^{(2)} \phi\left(\sum_{i=1}^{D} w_{ji}^{(1)} x_i + b^{(1)}\right) + b^{(2)}\right)$$

$$\{y_k\}_{k=1}^{K}$$

Definición. Perceptrón multi-capa unidireccional completamente conectado

Definición 2.2.1. *Una* ***red neuronal unidireccional completamente conectada*** *esta descrita por su arquitectura a* $\cancel{= \langle N, \phi \rangle,}$ *donde* $N \in \mathbb{N}^{L+1}$ *representa a la cantidad de nodos en cada capa,* $L \in \mathbb{N}$ *representa al número de capas y* $\phi : \mathbb{R} \to \mathbb{R}$ *se refiere a la función de activación. Sean* $n_0, \cancel{n_L \text{ y } n_\ell \in \mathbb{N}}$ *el número de neuronas en la capa de entrada, salida y* ℓ*-ésima, respectivamente. Sea* $f_a(\cdot|\theta) : \mathbb{R}^{n_0} \to \mathbb{R}^{n_L}$ *la función correspondiente a la red neuronal, que para todo valor* $\mathbf{x} \in \mathbb{R}^{n_0}$ *y parámetros*

$$\theta = \left(W^{[\ell]}, \mathbf{b}^{[\ell]}\right)_{\ell=1}^{L}, \in \bigtimes_{\ell=1}^{L} \left(\mathbb{R}^{n_\ell \times n_{\ell-1}} \times \mathbb{R}^{n_\ell}\right) \cong \mathbb{R}^{P(N)}$$

se expresa de la siguiente forma:

$$f_a(\boldsymbol{x}|\theta) := \phi(T_L(\phi(T_{L-1}(\cdots \phi(T_1(\mathbf{x}))\cdots)))),$$

donde T_ℓ *representa la transformación aplicada al valor de salida* $\mathbf{z} \in \mathbb{R}^{n_{\ell-1}}$ *de la capa anterior y esta dada por la siguiente expresión:*

$$T_\ell(\mathbf{z}) = W^{[\ell]}\mathbf{z} + \mathbf{b}^{[\ell]} \qquad W^{[\ell]} \in \mathbb{R}^{n_\ell \times n_{\ell-1}}, \mathbf{b}^{[\ell]} \in \mathbb{R}^{n_\ell} \qquad \forall \ell \in \{1, 2, \cdots, L\}.$$

La función de activación $\phi : \mathbb{R} \to \mathbb{R}$ *se aplica a cada componente de las salidas de cada una de las capas.*

Aclaraciones

- $N = (n_0, n_1, \dots n_\ell \dots n_L)$
- $a = \langle N, \{\phi^{[\ell]}\}_{\ell=1}^{L} \rangle$

Cardinalidad de parámetros

$$P(N) = \sum_{\ell=1}^{L} n_\ell \cdot n_{\ell-1} + n_\ell$$

Propagación hacia adelante (Forward propagation) -forma vectorial-

Para $\mathbf{x} \in \mathbb{R}^{n_0}$ el cálculo de las salidas $\mathbf{h}^{[\ell]}$ de cada capa ℓ de la red neuronal se puede calcular de la siguiente forma:

1

$$\mathbf{z}^{[1]} = W^{[1]}\mathbf{x} + \mathbf{b}^{[1]}$$
$$\mathbf{h}^{[1]} = \phi^{[1]}\left(\mathbf{z}^{[1]}\right)$$

2

$$\mathbf{z}^{[2]} = W^{[1]}\mathbf{h}^{[1]} + \mathbf{b}^{[2]}$$
$$\mathbf{h}^{[2]} = \phi^{[2]}\left(\mathbf{z}^{[2]}\right)$$
$$\vdots$$
$$\mathbf{z}^{[L]} = W^{[L]}\mathbf{h}^{[L-1]} + \mathbf{b}^{[L]}$$
$$\mathbf{y} = \phi^{[L]}\left(\mathbf{z}^{[L]}\right)$$

→ Salida

en general,

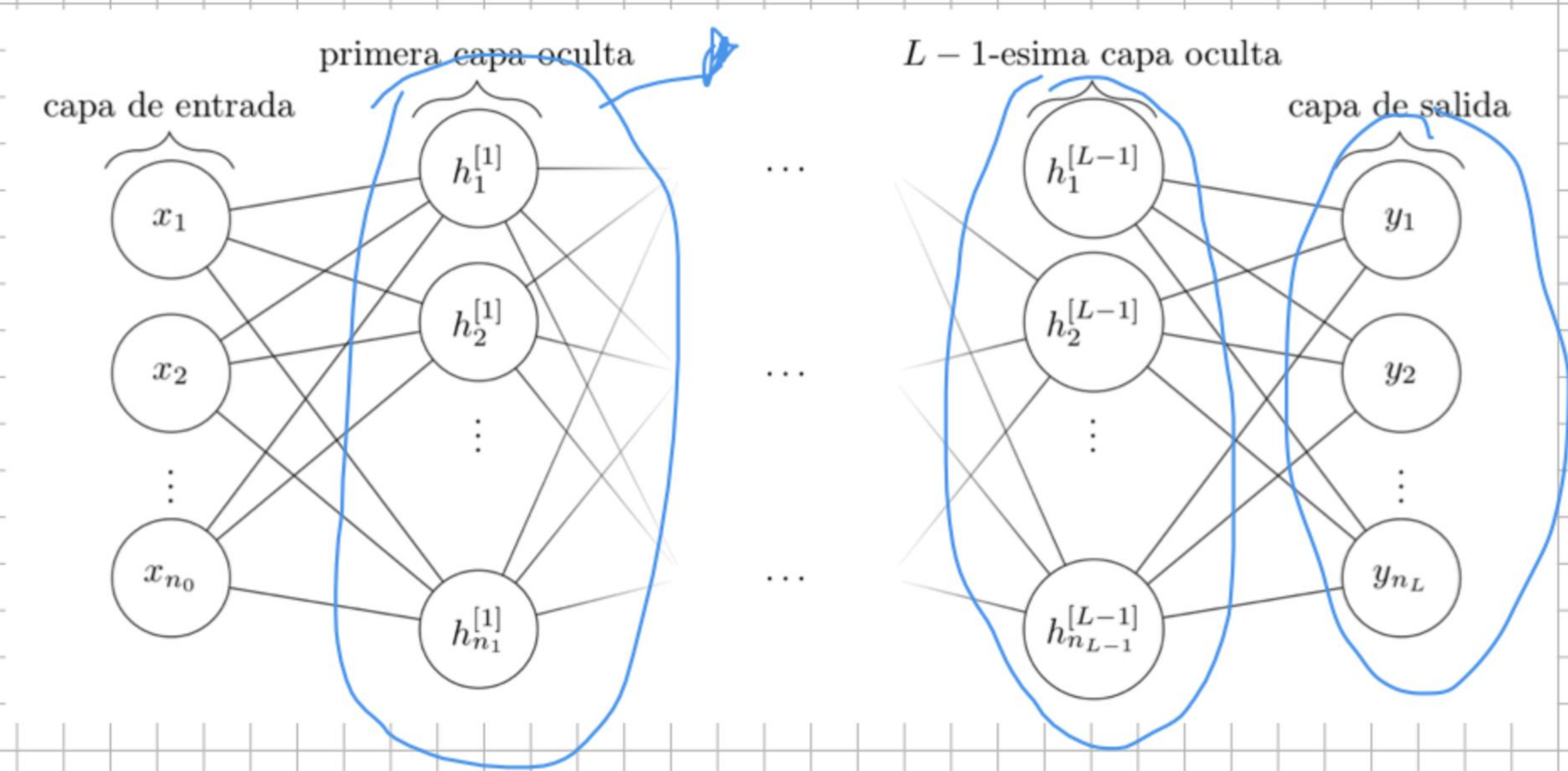
$$\left\{\begin{aligned} \mathbf{z}^{[\ell]} &= W^{[\ell]}\mathbf{h}^{[\ell-1]} + \mathbf{b}^{[\ell]} \\ \mathbf{h}^{[\ell]} &= \phi^{[\ell]}\left(\mathbf{z}^{[\ell]}\right) \end{aligned}\right. \quad \ell \in \{1, \cdots, L\},$$

donde $\mathbf{x} = \mathbf{h}^{[0]}$ y $f_a(\mathbf{x}|\theta) = \mathbf{h}^{[L]}$.

$$\underbrace{\begin{bmatrix} z_1^{[\ell]} \\ z_2^{[\ell]} \\ \vdots \\ z_{n_\ell}^{[\ell]} \end{bmatrix}}_{(n_\ell)} = \underbrace{\begin{bmatrix} w_{1,1}^{[\ell]} & w_{1,2}^{[\ell]} & \cdots & w_{1,n_{\ell-1}}^{[\ell]} \\ w_{2,1}^{[\ell]} & w_{2,2}^{[\ell]} & \cdots & w_{2,n_{\ell-1}}^{[\ell]} \\ \vdots & \vdots & & \vdots \\ w_{n_\ell,1}^{[\ell]} & w_{n_\ell,2}^{[\ell]} & \cdots & w_{n_\ell,n_{\ell-1}}^{[\ell]} \end{bmatrix}}_{\in \mathbb{R}^{n_\ell \times n_{\ell-1}}} \underbrace{\begin{bmatrix} h_1^{[\ell-1]} \\ \vdots \\ h_1^{n_{\ell-1}} \end{bmatrix}}_{(n_{\ell-1})} + \underbrace{\begin{bmatrix} b_1^{[\ell]} \\ b_2^{[\ell]} \\ \vdots \\ b_{n_\ell}^{[\ell]} \end{bmatrix}}_{(n_\ell)}$$

$(n_\ell \times n_{\ell-1})$

$$\underbrace{\begin{bmatrix} h_1^{[\ell]} \\ h_2^{[\ell]} \\ \vdots \\ h_{n_\ell}^{[\ell]} \end{bmatrix}} = \phi^{[\ell]} \begin{bmatrix} z_1^{[\ell]} \\ z_2^{[\ell]} \\ \vdots \\ z_{n_\ell}^{[\ell]} \end{bmatrix} = \begin{bmatrix} \phi^{[\ell]}\left(z_1^{[\ell]}\right) \\ \phi^{[\ell]}\left(z_2^{[\ell]}\right) \\ \vdots \\ \phi^{[\ell]}\left(z_{n_\ell}^{[\ell]}\right) \end{bmatrix}$$



primera capa oculta
L − 1-esima capa oculta
capa de entrada
capa de salida
x_1
x_2
x_{n_0}
$h_1^{[1]}$
$h_2^{[1]}$
$h_{n_1}^{[1]}$
$h_1^{[L-1]}$
$h_2^{[L-1]}$
$h_{n_{L-1}}^{[L-1]}$
y_1
y_2
y_{n_L}

Comentario sobre el teorema de aproximación universal

2.1 Universality

One of the most famous results in neural network theory is that, under minor conditions on the activation function, the set of networks is very expressive, meaning that every continuous function on a compact set can be arbitrarily well approximated by a MLP. This theorem was first shown by Hornik [13] and Cybenko [7].

To talk about approximation, we first need to define a topology on a space of functions of interest. We define, for $K \subset \mathbb{R}^d$

$$C(K) := \{f : K \to \mathbb{R} : f \text{ continuous}\}$$

and we equip $C(K)$ with the uniform norm

$$\|f\|_\infty := \sup_{x \in K} |f(x)|.$$

If K is a compact space, then the representation theorem of Riesz [28, Theorem 6.19] tells us that the topological dual space of $C(K)$ is the space

$$\mathcal{M} := \{\mu : \mu \text{ is a signed Borel measure on } K\}.$$

Having fixed the topology on $C(K)$, we can define the concept of *universality* next.

Definition 2.2. *Let $\varrho : \mathbb{R} \to \mathbb{R}$ be continuous, $d, L \in \mathbb{N}$ and $K \subset \mathbb{R}^d$ be compact. Denote by* $\mathrm{MLP}(\varrho, d, L)$ *the set of all MLPs with d-dimensional input, L layers, $N_L = 1$, and activation function ϱ.*

We say that $\mathrm{MLP}(\varrho, d, L)$ *is universal, if* $\mathrm{MLP}(\varrho, d, L)$ *is dense in* $C(K)$.

Example 2.1 demonstrates that $\mathrm{MLP}(\varrho, d, L)$ is *not* universal for every activation function.

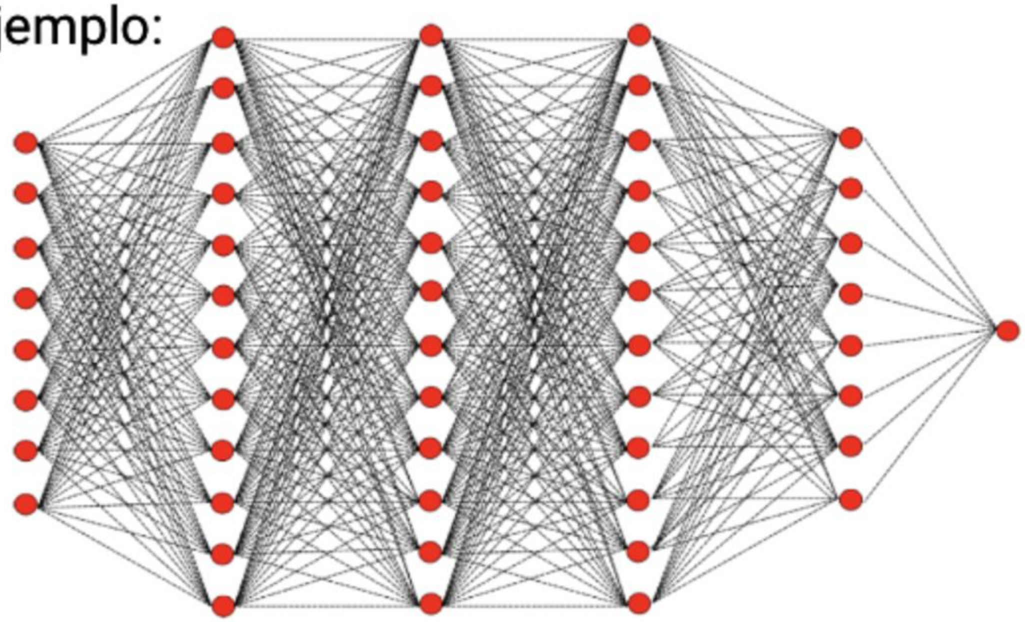
Definition 2.3. *Let $d \in \mathbb{N}$, $K \subset \mathbb{R}^d$, compact. A continuous function $f : \mathbb{R} \to \mathbb{R}$ is called* discriminatory *if the only measure $\mu \in \mathcal{M}$ such that*

$$\int_K f(ax - b)\,d\mu(x) = 0, \quad \text{for all } a \in \mathbb{R}^d, b \in \mathbb{R}$$

is $\mu = 0$.

Theorem 2.4 (Universal approximation theorem [7]). *Let $d \in \mathbb{N}$, $K \subset \mathbb{R}^d$ compact, and $\varrho : \mathbb{R} \to \mathbb{R}$ be discriminatory. Then* $\mathrm{MLP}(\varrho, d, 2)$ *is universal.*

Ejemplo:



L=

N=

a=

P(N)=

Propagación hacia adelante - forma matricial -

1

$$X = \begin{bmatrix} — & \mathbf{x}^{(1)} & — \\ — & \mathbf{x}^{(2)} & — \\ \vdots & \vdots & \vdots \\ — & \mathbf{x}^{(k-1)} & — \\ — & \mathbf{x}^{(k)} & — \end{bmatrix} \in \mathbb{R}^{k \times n_0}$$

Obs

Batch vs. mini-batch

k → # datos

k → fracción

3 Broadcasting

$$\underbrace{x}_{\in\mathbb{R}^{m\times n}} \circ \underbrace{y}_{\in\mathbb{R}^{1\times n}} = \underbrace{z}$$

$$(m, n) \circ (1, n) \to (m, n)$$

2

$$Z^{[1]} = XW^{[1]\top} + \mathbf{b}^{[1]\top}$$

$$H^{[1]} = \phi^{[1]}\left(Z^{[1]}\right)$$

$$Z^{[2]} = H^{[1]}W^{[2]\top} + \mathbf{b}^{[2]\top}$$

$$H^{[2]} = \phi^{[2]}\left(Z^{[2]}\right)$$

$$\vdots$$

$$Z^{[L]} = H^{[L-1]}W^{[L]\top} + \mathbf{b}^{[L]\top}$$

$$Y = \phi^{[L]}\left(Z^{[L]}\right)$$

$$Z^{[\ell]} = H^{[\ell-1]}W^{[\ell]\top} + \mathbf{b}^{[\ell]\top} \quad \ell \in \{1, \cdots, L\}$$

$$H^{[\ell]} = \phi^{[\ell]}\left(Z^{[\ell]}\right)$$

donde $X = H^{[0]}$ y $f_a(X|\theta) = H^{[L]}$

Entrenamiento y optimización de redes neuronales

1. Planteamiento del problema
2. Descenso de gradiente (gradient descent)
3. Stochastic vs. batch vs. mini-batch gradient descent
4. Retropropagación de errores (backpropagation)

1.

$$\min(\max) J(\theta)$$

función objetivo

- MSE
- BCE

$$J(\theta; f_a(X|\theta))$$

etiquetas

2.

$$\theta_{t+1} = \theta_t - \eta \nabla_{\theta_t} J(\theta)$$

tasa de aprendizaje

3.

$$\theta_{t+1} = \theta_t - \eta \nabla_\theta J(\theta_t; X, Y)$$

$$\theta_{t+1} = \theta_t - \eta \nabla_\theta J\left(\theta_t; x^{(i)}, y^{(i)}\right)$$

$$\theta_{t+1} = \theta_t - \eta \nabla_\theta J\left(\theta_t; x^{(i:i+k)}, y^{(i:i+k)}\right)$$