Redes Neuronales Artificiales (ANN)

Diego Armando Pérez Rosero ·, Santiago Pineda Quintero

Departamento de Ingeniería Eléctrica, Electrónica y Computación
Universidad Nacional de Colombia
Manizales, Colombia
Septiembre 2023

Contenido

Definiciones

- En aprendizaje de máquina (AM), un **modelo** es el elemento donde se almacena el aprendizaje.
- Los modelos de AM necesitan un conjunto de datos para poder aprender los patrones relevantes que ayuden a resolver una tarea. A esto se denomina la **etapa de entrenamiento**
- $f(\mathbf{X};\theta)$ representa un modelo; donde θ son los parámetros del modelo y representan lo que se aprendió.

Notación

UNIVERSIDAD NACIONAL DE COLOMBIA

$\mathbf{X} \in \mathbb{R}^{NxP}$, donde N y P son el número de muestras y atributos respectivamente.

$$\mathbf{X} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{np} \end{bmatrix}$$

$\mathbf{X_n} \in \mathbb{R}^{P}$ es una única muestra del conjunto de datos.

$$\mathbf{x}_n = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1p} \end{bmatrix}$$

el subíndice n se utiliza para referirse a una muestra del conjunto de datos.

Notación

UNIVERSIDAD NACIONAL DE COLOMBIA

$\mathbf{y} \in \Omega^N$ son el conjunto de etiquetas , y su dominio Ω depende de la tarea a resolver. Para tareas de regresión $\mathbf{y} \in \mathbb{R}^N$, en clasificación sería:

$\mathbf{Y} \in [0,1]^{N \times C}$, para el caso de que se prediga la probabilidad de cada clase.

$\mathbf{y} \in \{0, 1, \ldots, C\}^N$, para el caso de que se predigan directamente las clases.

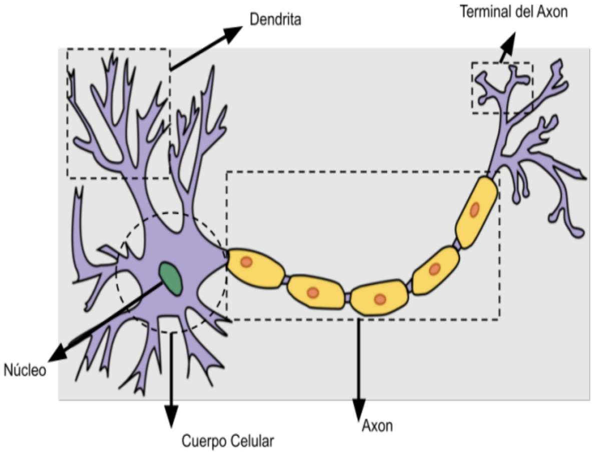
$$\mathbf{y} = \begin{bmatrix} 1 \\ 2 \\ \vdots \\ C \end{bmatrix}$$

$$\mathbf{Y} = \begin{bmatrix} P(0|\mathbf{x_1}) & P(1|\mathbf{x_1}) & \cdots & P(C|\mathbf{x_1}) \\ P(0|\mathbf{x_2}) & P(1|\mathbf{x_2}) & \cdots & P(C|\mathbf{x_2}) \\ \vdots & \vdots & \ddots & \vdots \\ P(0|\mathbf{x_n}) & P(1|\mathbf{x_n}) & \cdots & P(C|\mathbf{x_n}) \end{bmatrix}$$

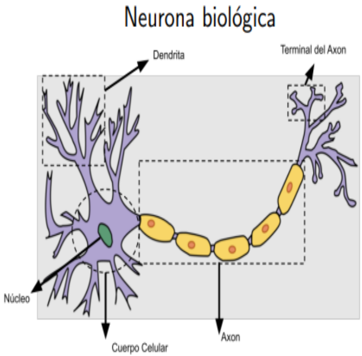
$\mathbf{y}_n \in [0,1]^C$ son las probabilidades de que *una* muestra pertenezca a cada clase.

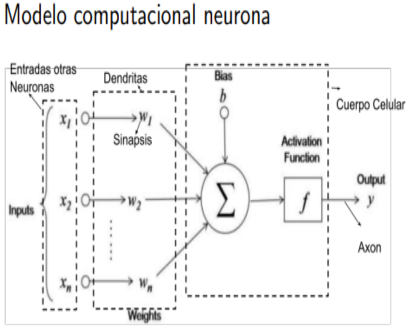
$$\mathbf{y}_n = \begin{bmatrix} P(0|\mathbf{x_n}) & P(1|\mathbf{x_n}) & \cdots & P(C|\mathbf{x_n}) \end{bmatrix}$$

Neurona Biológica



Aproximación matemática de una Neurona





Comparación de Modelos

Neurona biológica

1. # de dendritas.
2. Sinapsis.
3. Cuerpo de la neurona.
4. Axón.

Neurona artificial

1. # de características P que tiene cada muestra.

$$\mathbf{x}_n = \begin{bmatrix} x_1 & x_1 & \cdots & x_p \end{bmatrix}^\top$$

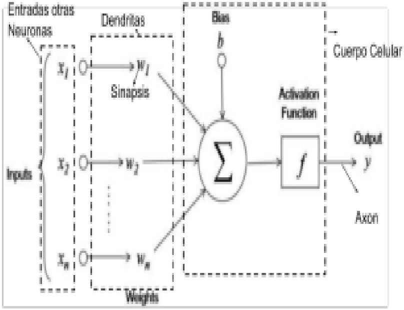
2. Los pesos w_i.

$$\mathbf{w} = \begin{bmatrix} w_1 & w_1 & \cdots & w_p \end{bmatrix}^\top$$

3. El sumatorio $\sum$ y la *función de activación* $f(.)$.
4. Salida $\hat{y}$.

Aproximación Matemática Neurona

Modelo computacional neurona



$$\hat{y} = f(w_1x_1 + w_2x_2 + \ldots + w_px_p + b)$$

$$\hat{y} = f(\sum_{i=1}^{p} w_ix_i + b)$$

$$\hat{y} = f(\langle \mathbf{w}, \mathbf{x} \rangle + b)$$

donde $\mathbf{x}, \mathbf{w} \in \mathbb{R}^p$, $b \in \mathbb{R}$ y el dominio de y depende la tarea tarea a resolver, como se vio anteriormente.

¿Qué importancia tienen w en el modelo computacional de una neurona?

El conocimiento que se va adquiriendo, se almacena en las conexiones neuronales (sinapsis). Entre más fuerte sea la conexión, mayor importancia tiene el conocimiento adquirido.

Por lo tanto, el vector **w** representa el conocimiento adquirido y sus elementos w_i *la importancia que tiene cada característica para el modelo.*

Generalizando el modelo de una neurona artificial

El modelo que vimos anteriormente nos arroja la salida $\hat{y}$ para una única muestra $\mathbf{x}_n$. Se puede generar la salida para un conjunto de muestras $\mathbf{X}$ como:

$$\hat{\mathbf{y}} = f(\tilde{\mathbf{X}}\mathbf{w})$$

donde $\tilde{\mathbf{X}} = [\mathbf{X}, \mathbf{1}]$ es $\mathbf{X}$ más un vector columna de unos y $\hat{\mathbf{y}} \in \Omega^N$ depende de la tarea a resolver, al igual que para el caso de una muestra.

Nota: El símbolo $\wedge$ se utiliza para denotar las salidas generadas por un modelo.

Capa Densa

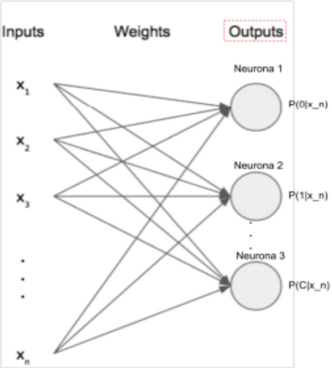
Adicionalmente, el modelo de neurona artificial descrito genera una única salida $\hat{y}$, que para el caso de una tarea de regresión o clasificación bi-clase seria suficiente. No obstante, en el caso de que se tuviera una tarea de clasificación multi-clase donde se prediga la probabilidad de que una muestra pertenezca a cada clase, se necesitaría que el modelo arroje a la salida un vector de probabilidades con número de elementos igual al número de clases como se menciono anteriormente.

$$\mathbf{y}_n = \begin{bmatrix} P(0|\mathbf{x_n}) & P(1|\mathbf{x_n}) & \cdots & P(C|\mathbf{x_n}) \end{bmatrix}$$

Para lograr este objetivo, se introduce una pila de neuronas conocida como **Capa Densa**, donde cada una de estas se encargara de predecir la probabilidad de una clase.

Nota: En una capa densa, todas las neuronas están conectadas a todas las entradas.

Capa Densa



Por lo tanto, la salida de una capa densa se puede generar como:

$$\hat{\mathbf{Y}} = f(\tilde{\mathbf{X}}\mathbf{W}^T)$$

donde $\mathbf{W} \in \mathbb{R}^{Q\times(P+1)}$, es la matriz que alberga los pesos de cada una de las neuronas de la capa densa, $\mathbf{b} \in \mathbb{R}^Q$ es un vector que alberga el bias de cada neurona y Q es el número de neuronas en la capa densa, que en este caso sería $Q = C$.

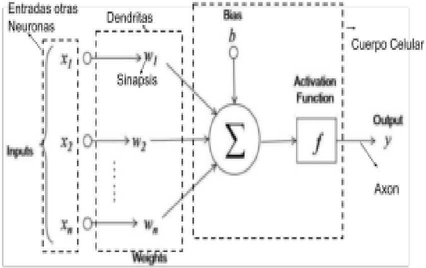
Capa Densa

UNIVERSIDAD NACIONAL DE COLOMBIA

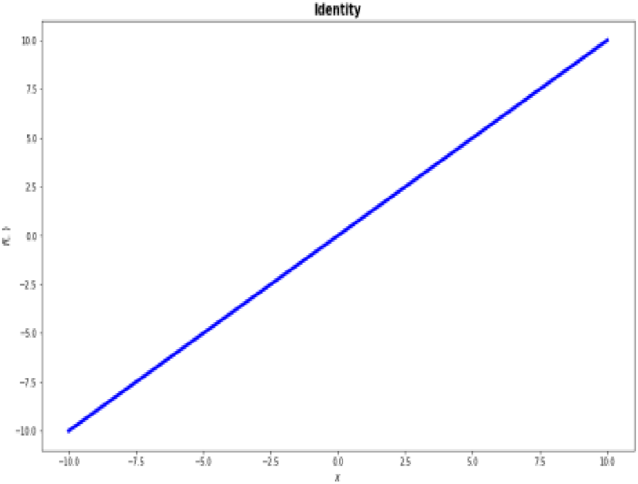
$$
\begin{aligned}
\hat{\mathbf{Y}} &= f(\tilde{\mathbf{X}}\mathbf{W}^T) \\
&= f\left(\begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1p} & 1 \\ x_{21} & x_{22} & \cdots & x_{2p} & 1 \\ \vdots & \vdots & \ddots & \vdots & \\ x_{n1} & x_{n2} & \cdots & x_{np} & 1 \end{bmatrix} \begin{bmatrix} w_{11} & w_{21} & \cdots & w_{q1} \\ w_{12} & x_{22} & \cdots & w_{q2} \\ \vdots & \vdots & \ddots & \vdots \\ w_{1p} & w_{2p} & \cdots & w_{qp} \\ w_{1(p+1)} & w_{2(p+1)} & \cdots & w_{q(p+1)} \end{bmatrix}\right) \\
&= \begin{bmatrix} P(0|\mathbf{x_1}) & P(1|\mathbf{x_1}) & \cdots & P(C|\mathbf{x_1}) \\ P(0|\mathbf{x_2}) & P(1|\mathbf{x_2}) & \cdots & P(C|\mathbf{x_2}) \\ \vdots & \vdots & \ddots & \vdots \\ P(0|\mathbf{x_n}) & P(1|\mathbf{x_n}) & \cdots & P(C|\mathbf{x_n}) \end{bmatrix}
\end{aligned}
$$

Funciones de Activación

El cuerpo de la neurona, es la parte encargada de realizar el proceso sobre las señales de entrada para generar la salida. La neurona artificial imita este elemento a través de las funciones de activación $f(.)$ en conjunto con el sumatorio $\sum$. Por lo tanto, es de vital importancia conocer los diferentes tipos de funciones de activación que se tienen, para de acuerdo a la tarea a resolver se escoja la más adecuada.

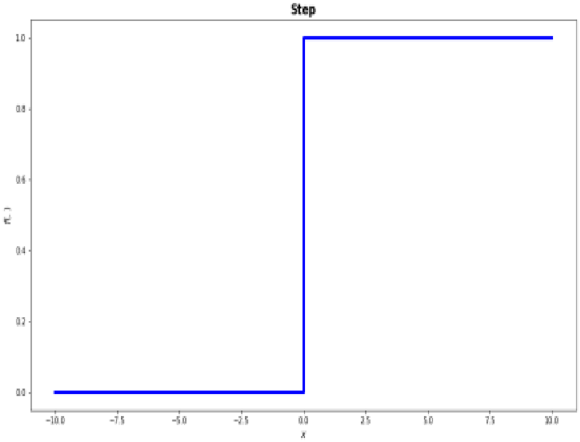


Funciones de Activación



$$f : \mathbb{R} \longrightarrow \mathbb{R}$$
$$f(x) = x$$

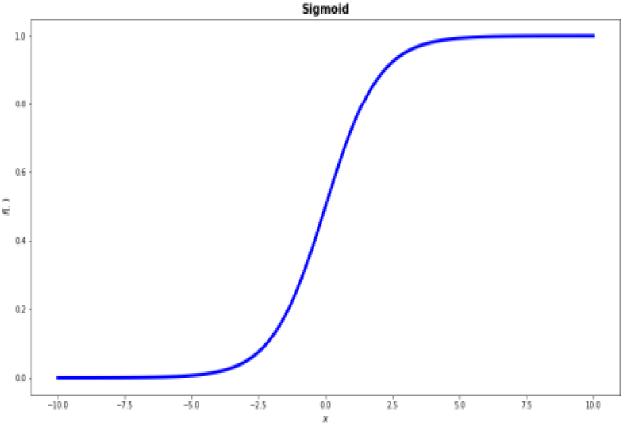
Funciones de Activación



$$f : \mathbb{R} \longrightarrow \{0,1\}$$

$$f(x) = \begin{cases} 0 & \text{if } x < 0 \\ x & \text{if } x \geq 0 \end{cases}$$

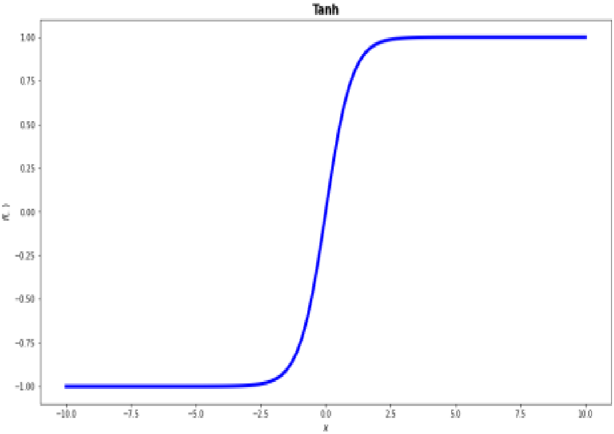
Funciones de Activación



$f : \mathbb{R} \longrightarrow (0,1)$

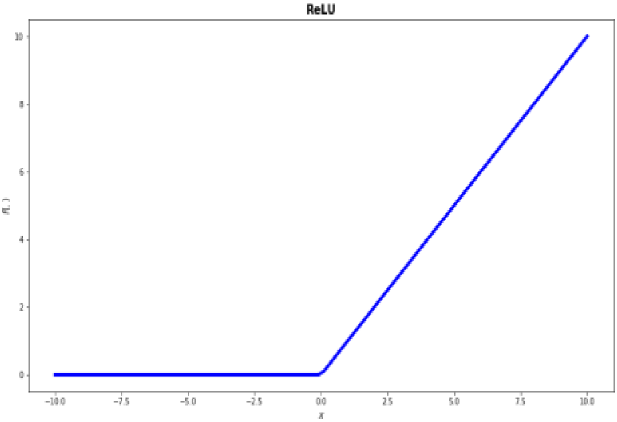
$f(x) = \frac{1}{1+e^{-x}}$

Funciones de Activación



$$f:\mathbb{R}\longrightarrow(-1,1)$$
$$f(x)=\frac{e^{x}-e^{-x}}{e^{x}+e^{-x}}$$

Funciones de Activación



$$f : \mathbb{R} \longrightarrow [0, \infty)$$
$$f(x) = \max(0, x)$$

Recapitulando

UNIVERSIDAD NACIONAL DE COLOMBIA

- El modelo es el elemento que se encarga de almacenar el aprendizaje.
- Los parámetros del modelo, ($\mathbf{w}$ en el caso de la neurona artificial y $\mathbf{W}$ en el caso de la capa densa) representan el aprendizaje.
- Las métricas o medidas de desempeño, nos permiten evaluar si el aprendizaje adquirido es acorde o no a la tarea que se desea resolver.

Sin embargo,

¿Cómo se le especifica al modelo que es lo que debe de aprender? y al mismo tiempo, ¿Cómo se supervisa el proceso de aprendizaje?
Respuesta: A través de las *funciones de costo*.

Dependiendo del tipo de aprendizaje y tarea que se desee desempeñar existen diferentes tipos de funciones de costo. En este curso, nos enfocaremos en **tareas de clasificación para modelos de aprendizaje supervisados**.

Definiciones

UNIVERSIDAD NACIONAL DE COLOMBIA

Función de perdida

Permite evaluar el proceso de aprendizaje en una sola muestra $\mathbf{x}_n$ del conjunto de entrenamiento.

Función de costo

Permite evaluar el proceso de aprendizaje en todo el conjunto de entrenamiento $\mathbf{X}$.

La función de perdida mas comúnmente utilizada para tareas de clasificación es la **Entropía Cruzada**.

Entropia Cruzada

UNIVERSIDAD NACIONAL DE COLOMBIA

Función de perdida

Permite evaluar el proceso de aprendizaje en una sola muestra $\mathbf{x}_n$ del conjunto de entrenamiento.

Función de costo

Permite evaluar el proceso de aprendizaje en todo el conjunto de entrenamiento $\mathbf{X}$.

La función de perdida mas comúnmente utilizada para tareas de clasificación es la **Entropía Cruzada**.

Tarea: Consultar a cerca de la función de costo entropia cruzada categorica

Ejemplo 1

Ejemplo 1 :

$\mathbf{y}$	1	0	0
$\hat{\mathbf{y}}$	0,98	0,01	0,01

$\mathbf{y}$	1	0	0
$\log(\hat{\mathbf{y}})$	$-0{,}02$	$-4{,}61$	$-4{,}61$

$-\mathbf{y} \odot \log(\hat{\mathbf{y}})$	0,02	0	0

$H(\mathbf{y}, \hat{\mathbf{y}}) = 0{,}02$;

Ejemplo 2

Ejemplo 1 :

$\mathbf{y}$	1	0	0
$\hat{\mathbf{y}}$	0,98	0,01	0,01

$\mathbf{y}$	1	0	0
$\log(\hat{\mathbf{y}})$	$-0{,}02$	$-4{,}61$	$-4{,}61$

$-\mathbf{y} \odot \log(\hat{\mathbf{y}})$	0,02	0	0

$H(\mathbf{y}, \hat{\mathbf{y}}) = 0{,}02$; Un valor cercano a 0 significa que el modelo desempeño bien la tarea en la muestra.

Ejemplo 2 :

$\mathbf{y}$	0	0	1
$\hat{\mathbf{y}}$	0,3	0,2	0,5

$\mathbf{y}$	0	0	1
$\log(\hat{\mathbf{y}})$	$-1{,}2$	$-1{,}61$	$-0{,}69$

$-\mathbf{y} \odot \log(\hat{\mathbf{y}})$	0	0	0,69

$H(\mathbf{y}, \hat{\mathbf{y}}) = 0{,}69$; Un valor lejano a 0 significa que el modelo no desempeño bien la tarea en la muestra y debe mejorar.

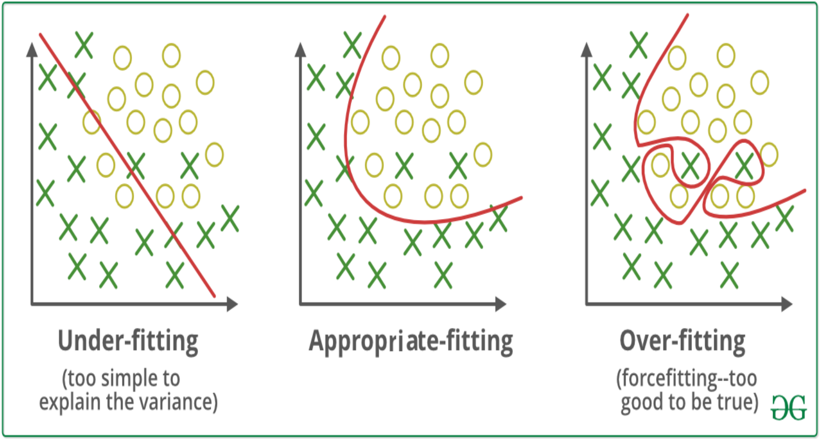
Función de Costo

Evalúa el proceso de aprendizaje sobre el conjunto de muestras $\mathbf{X}$ como el promedio de las perdidas en las muestras.

$$C(\mathbf{Y}, \hat{\mathbf{Y}}) = \frac{1}{N} \sum_{n=1}^{N} H(\mathbf{y}_n, \hat{\mathbf{y}}_n)$$

El objetivo, es tratar de que el valor de la función de costo sea el **mínimo** posible. De esta manera, aseguramos que el modelo va a desempeñar bien la tarea en el conjunto de entrenamiento. Sin embargo, es importante aclarar que puede ocurrir sobre-ajuste y en lugar del modelo aprender memoriza los datos. Por tal motivo, es importante el conjunto de prueba, que permite validar si efectivamente el modelo aprendió los patrones (reglas) relevantes asociados a la tarea de interés. Las funciones de costo por si solas no aseguran que el modelo aprenda, como se menciono anteriormente puede memorizar.

Underfitting y Overfitting



Función de Costo en Términos de W

UNIVERSIDAD NACIONAL DE COLOMBIA

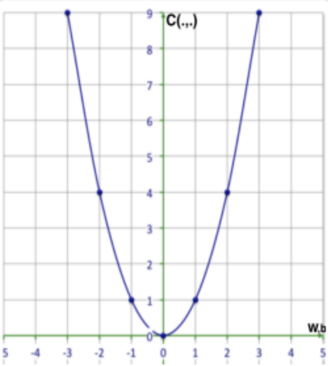
Si escribimos $\hat{\mathbf{y}}_n$ en términos de la salida de una capa densa como $\hat{\mathbf{y}}_n = f(\tilde{\mathbf{x}}_n \mathbf{W}^T)$, la función de costo ahora nos quedaría:

$$C(\mathbf{X}, \mathbf{Y}; \mathbf{W}, \mathbf{b}) = \frac{1}{N} \sum_{n=1}^{N} H(\mathbf{y}_n, f(\tilde{\mathbf{x}}_n \mathbf{W}^T))$$

Ya que $\mathbf{X}$ y $\mathbf{Y}$ están dados, la función de costo nos quedaría en términos de $\mathbf{W}$.

Adquisición de Aprendizaje en una Capa Densa

¿Cómo se calcula **W** para una capa densa?
Respuesta: Minimizando la función de costo



Adquisición de Aprendizaje en una Capa Densa

Si hacemos $\theta = \{\mathbf{W}\}$, se tendría el siguiente problema de optimización.

$$\theta^* = \arg\min_{\theta} C(\mathbf{Y}, \hat{\mathbf{Y}})$$

Donde encontrar el mínimo valor de θ, corresponde a la adquisición de aprendizaje por parte de la capa densa para resolver la tarea de interés.

¿Cómo se minimiza la función de costo?

Respuesta: Al igual que se hacía en calculo diferencial cuando se quería minimizar una función: (i) Calculando el gradiente (∇) (ii) igualando a cero y (iii) resolviendo la ecuación resultante.

Ejemplo1

Dada una capa densa con función de activación *lineal*, se requiere resolver una tarea de regresión utilizando la función de perdida de *mínimos cuadrados*.

- Escriba el modelo matemático.
- Escriba la función de costo.
- Calcule el ∇ de la función de costo con respecto a W.
- Encuentre los valores de W que igualen el gradiente a 0 (minimice la función de costo).

Ejemplo2

Dada una capa densa con función de activación *sigmoide*, se requiere resolver una tarea de clasificación utilizando la función de perdida de *entropía cruzada*.

- Escriba el modelo matemático.
- Escriba la función de costo.
- Calcule el ∇ de la función de costo con respecto a W.
- Encuentre los valores de W que igualen el gradiente a 0 (minimice la función de costo).

Problemas al minimizar la función de costo

La mayoría de las veces, encontrar los valores de W que igualen el gradiente a 0 es difícil, ya que en muchos casos no se tiene una solución cerrada o el problema esta mal condicionado.
Por ende, es importante tener otras estrategias que permitan encontrar los parámetros W óptimos que minimicen la función de costo.

Solución: Gradiente Descendiente.

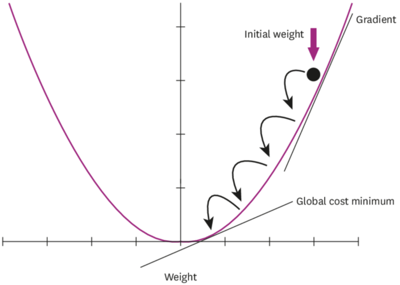
Problemas al minimizar la función de costo

La mayoría de las veces, encontrar los valores de W que igualen el gradiente a 0 es difícil, ya que en muchos casos no se tiene una solución cerrada o el problema esta mal condicionado.
Por ende, es importante tener otras estrategias que permitan encontrar los parámetros W óptimos que minimicen la función de costo.

Solución: Gradiente Descendiente.

Gradiente Descendiente

La técnica de gradiente descendiente, es un algoritmo iterativo que permite minimizar o maximizar una función, utilizando la información de máximo o mínimo cambio que arroja el gradiente. Al ser un algoritmo iterativo, requiere de una secuencia de intentos hasta que este converja.



Vista Geométrica de una capa Densa

La capa densa es una pila de neuronas, donde el funcionamiento de cada una de estas se resume en 2 operaciones fundamentales:

- Una combinación lineal (suma ponderada y la adición de una constante).
- Una función de activación.

$$\hat{y} = f(\langle w, x \rangle + b)$$

La combinación lineal que realiza cada neurona sobre sus entradas, se ve desde el punto de vista geométrico como un hiper-plano con vector normal w.

Vista Geométrica de una capa Densa

La capa densa es una pila de neuronas, donde el funcionamiento de cada una de estas se resume en 2 operaciones fundamentales:

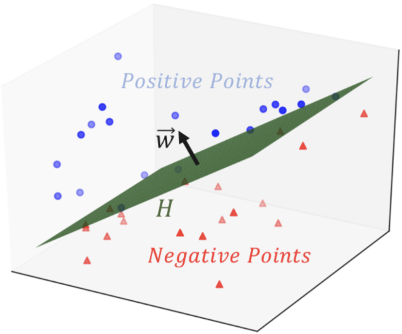
- Una combinación lineal (suma ponderada y la adición de una constante).
- Una función de activación.

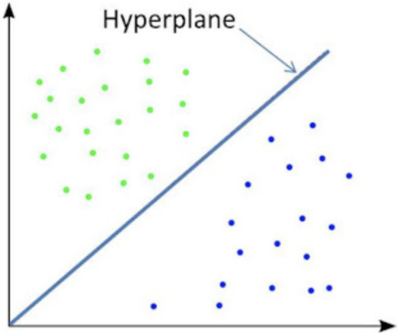
$$\hat{y} = f(\langle w, x \rangle + b)$$

La combinación lineal que realiza cada neurona sobre sus entradas, se ve desde el punto de vista geométrico como un hiper-plano con vector normal w.

Vista Geométrica de una capa Densa

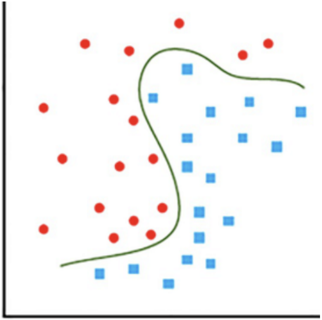
UNIVERSIDAD NACIONAL DE COLOMBIA

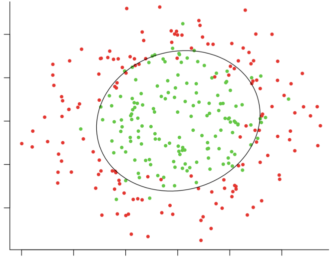




Fronteras de decisión no-lineales

Para el caso donde se requiera resolver un problema de clasificación con una frontera de decisión lineal este modelo es adecuado. Sin embargo, para problemas con fronteras de decisión no-lineal este modelo no es suficiente.





Redes Neuronales Artificiales (ANN)

¿Qué se puede hacer en este caso?.

Apilar varias capas densas, cada una de estas con funciones de activación no lineal.

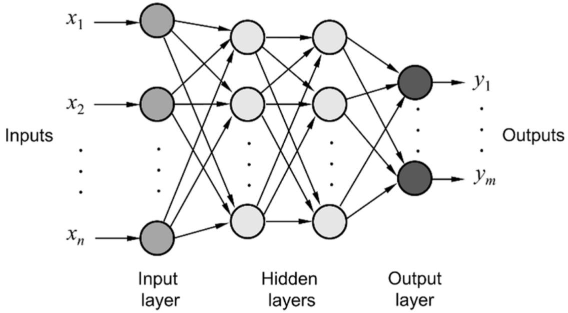
A este secuencia de capas densas se les conoce como **Redes Neuronales Artificiales**. Las capas densas que se encuentren ubicadas entre la capa de entrada y capa de salida, se les conoce como *Capas ocultas* y su trabajo consistirá en mapear los datos de un espacio a otro, buscando que estos sean linealmente separables para que la capa de salida puedo operar sobre ellos.

Redes Neuronales Artificiales (ANN)

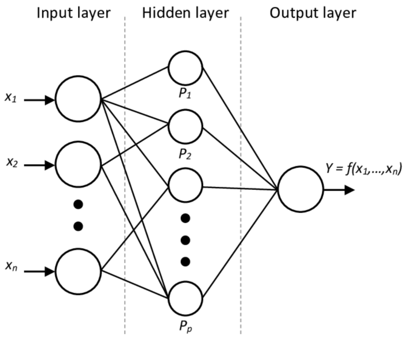
¿Qué se puede hacer en este caso?.
Apilar varias capas densas, cada una de estas con funciones de activación no lineal.

A este secuencia de capas densas se les conoce como **Redes Neuronales Artificiales**. Las capas densas que se encuentren ubicadas entre la capa de entrada y capa de salida, se les conoce como *Capas ocultas* y su trabajo consistirá en mapear los datos de un espacio a otro, buscando que estos sean linealmente separables para que la capa de salida puedo operar sobre ellos.

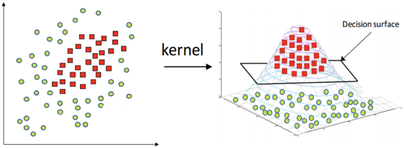
Redes Neuronales Artificiales (ANN)

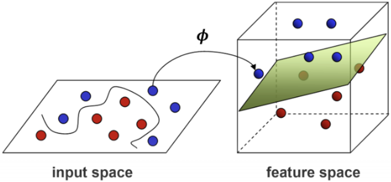


Vista Geométrica de una capa Oculta



Vista Geométrica de una capa Oculta





Modelo de una ANN

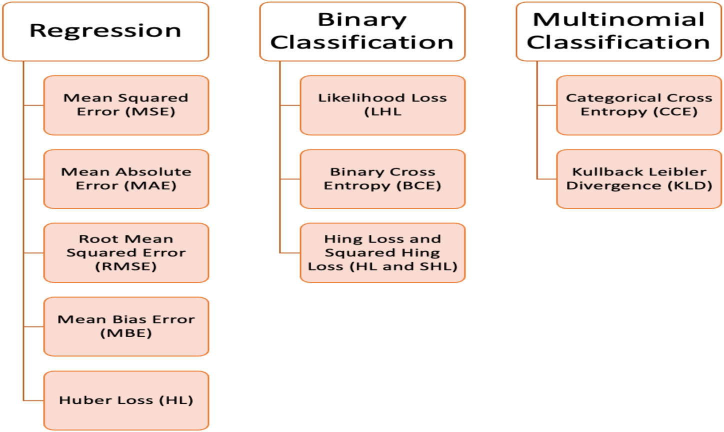
Teniendo en cuenta lo anterior, el modelo de una red neuronal se define como:

$$\hat{Y} = (f_L \circ f_{L-1} \circ \cdots \circ f_1)(\tilde{X})$$

donde L es el numero de capas ocultas, $f = f(\tilde{X}W^T)$ es el modelo de una capa densa, $\tilde{X}$ es el conjunto de datos extendido con el vector de unos y Y es el conjunto de etiquetas.

Funciones de costo

UNIVERSIDAD NACIONAL DE COLOMBIA



Thanks!