

{desafío}
latam_

Perceptrón y Tensores _



Tensoros

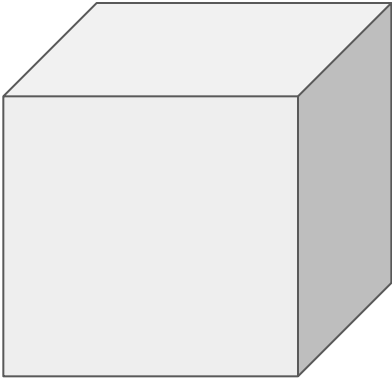
Implementaciones de Tensores

Definición: Arreglo multidimensional de datos, similar a una matriz, sólo que proyectada en dos o más dimensiones.

Son generalizaciones de estructuras de datos más simples:

- Un **escalar** es un tensor cero-dimensional.
- Un **vector** es un tensor uni-dimensional.
- Una **matriz** es un tensor bi-dimensional.

¿Qué son los tensores?

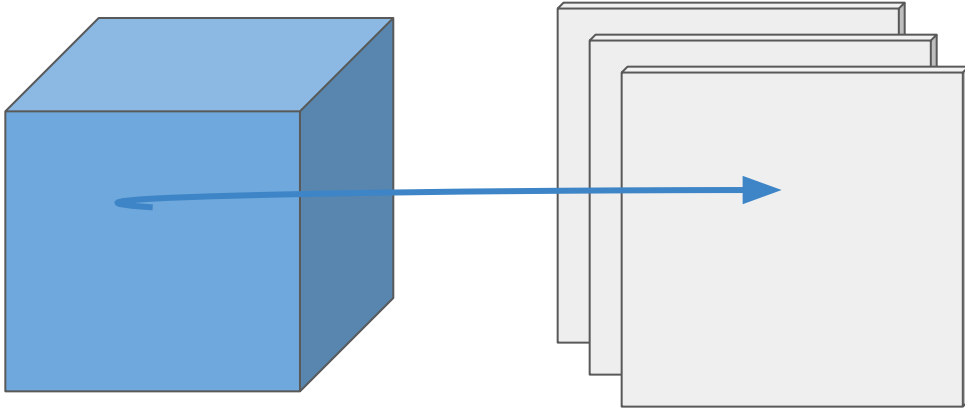


Tensor

$$\begin{bmatrix} \begin{bmatrix} 1 & 2 \\ 3 & 8 \end{bmatrix} & \begin{bmatrix} 1 & 6 \\ 6 & 12 \end{bmatrix} \end{bmatrix}$$

desafío
latam_

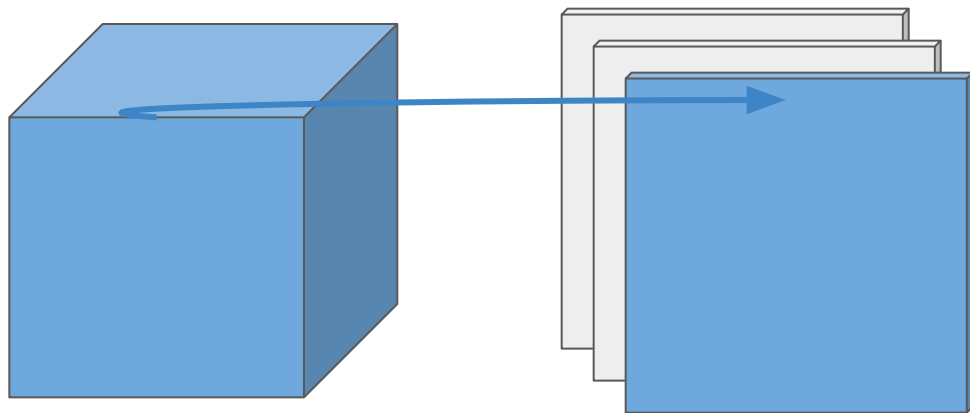
¿Qué son los tensores?



Tensor

$$\left[\begin{bmatrix} 1 & 2 \\ 3 & 8 \end{bmatrix} \quad \begin{bmatrix} 1 & 6 \\ 6 & 12 \end{bmatrix} \right]$$

¿Qué son los tensores?



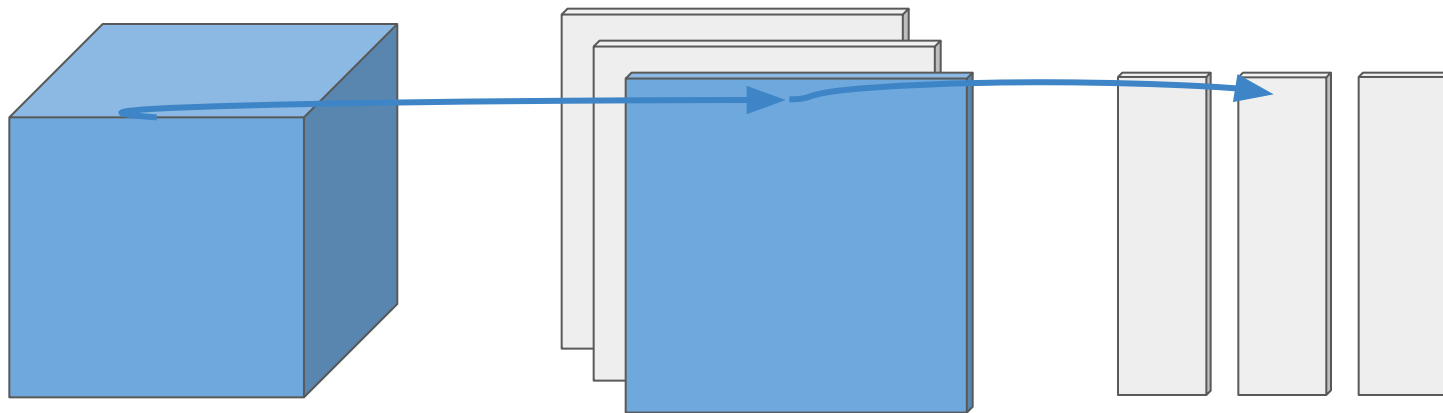
Tensor

$$\begin{bmatrix} \begin{bmatrix} 1 & 2 \\ 3 & 8 \end{bmatrix} & \begin{bmatrix} 1 & 6 \\ 6 & 12 \end{bmatrix} \end{bmatrix}$$

Matriz

$$\begin{bmatrix} 1 & 2 \\ 1 & 6 \end{bmatrix}$$

¿Qué son los tensores?



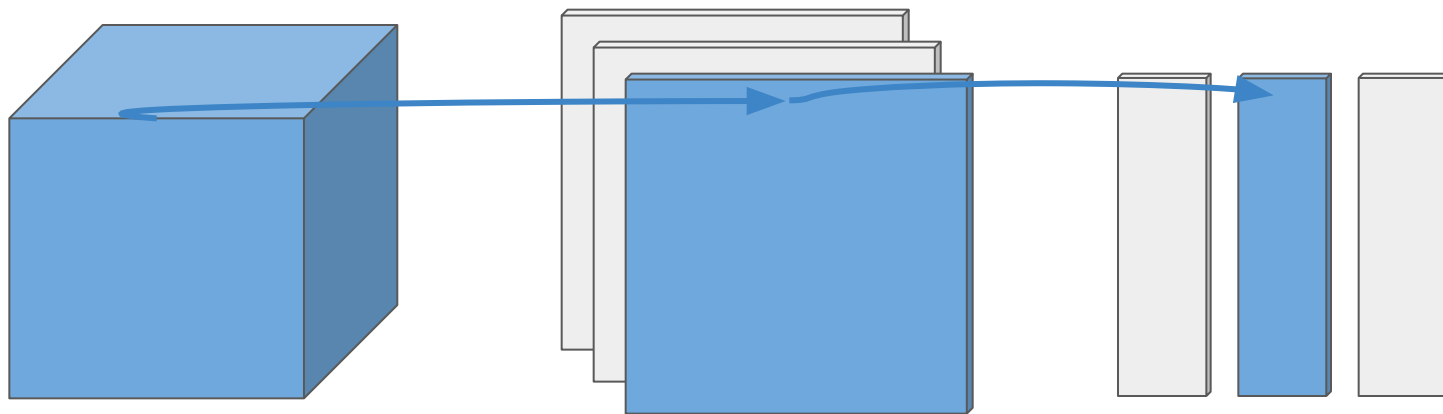
Tensor

$$\begin{bmatrix} \begin{bmatrix} 1 & 2 \\ 3 & 8 \end{bmatrix} & \begin{bmatrix} 1 & 6 \\ 6 & 12 \end{bmatrix} \end{bmatrix}$$

Matriz

$$\begin{bmatrix} 1 & 2 \\ 1 & 6 \end{bmatrix}$$

¿Qué son los tensores?



Tensor

$$\begin{bmatrix} \begin{bmatrix} 1 & 2 \\ 3 & 8 \end{bmatrix} & \begin{bmatrix} 1 & 6 \\ 6 & 12 \end{bmatrix} \end{bmatrix}$$

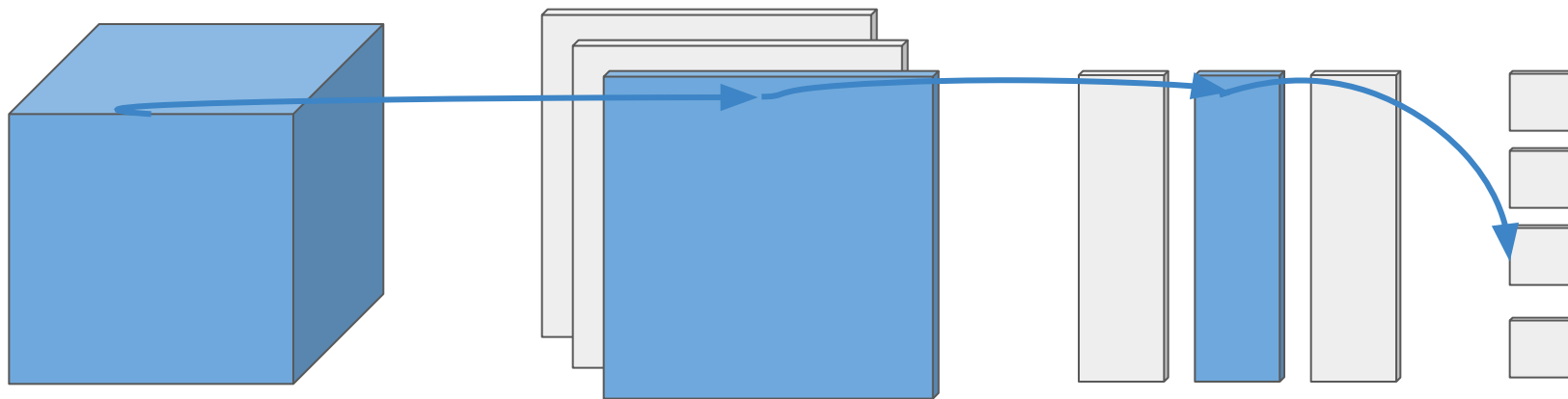
Matriz

$$\begin{bmatrix} 1 & 2 \\ 1 & 6 \end{bmatrix}$$

Vector

$$\begin{bmatrix} 1 \\ 2 \end{bmatrix}$$

¿Qué son los tensores?



Tensor

$$\begin{bmatrix} \begin{bmatrix} 1 & 2 \\ 3 & 8 \end{bmatrix} & \begin{bmatrix} 1 & 6 \\ 6 & 12 \end{bmatrix} \end{bmatrix}$$

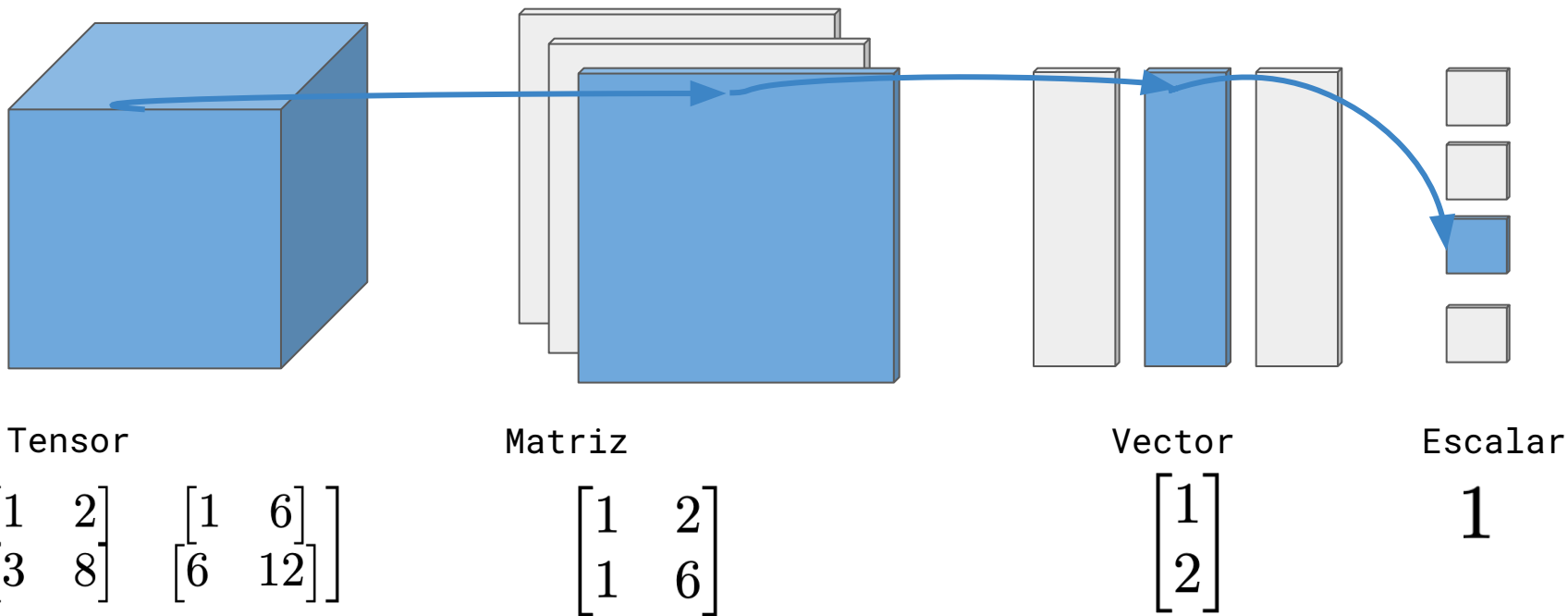
Matriz

$$\begin{bmatrix} 1 & 2 \\ 1 & 6 \end{bmatrix}$$

Vector

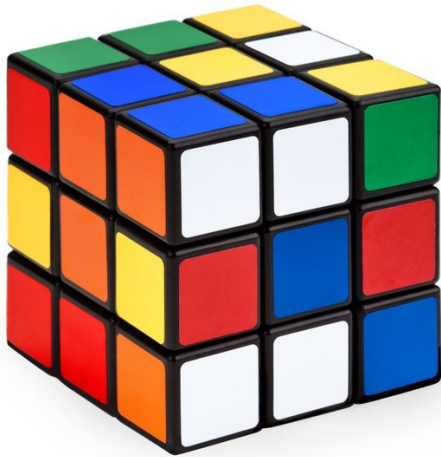
$$\begin{bmatrix} 1 \\ 2 \end{bmatrix}$$

¿Qué son los tensores?



Características de los Tensores

Forma: Informa la cantidad de información alojada en cada una de las dimensiones del tensor.



Forma del Tensor: (9, 9, 9)

Características de los Tensores

Forma: Informa la cantidad de información alojada en cada una de las dimensiones del tensor.

Rango: Define la cantidad de dimensiones en un tensor



Forma del Tensor: (9, 9, 9)

Rango: 3

Características de los Tensores

Forma: Informa la cantidad de información alojada en cada una de las dimensiones del tensor.

Rango: Define la cantidad de dimensiones en un tensor

Tipo de dato: Un tensor debe estar asociado a un tipo de dato específico.



Forma del Tensor: (9, 9, 9)

Rango: 3

Tipo de Datos: Colores

Implementaciones de Tensores

Ya tenemos conocimiento sobre cómo operar con matrices simples.

Muchas de las implementaciones modernas de Machine Learning hacen uso de estructuras de datos más complejas que se pueden capturar bien en tensores:

- **Estructuras de datos 3-dimensionales:** Archivos de audio, imágenes, etc...
- **Estructuras de datos 4-dimensionales:** Vídeos, reconocimiento facial online.

Operaciones a Nivel de Tensores

- Como toda estructura de datos, debe tener una serie de operaciones elementales.
- Si tenemos un vector de N-dimensiones con un tipo de dato específico con una función a nivel tensorial, su resultado debe estar definido a nivel del tipo de dato.
- Operaciones a diferenciar:
 - **Operaciones de elemento-a-elemento** (*element-wise*): Opera sobre cada uno de los elementos alojados en el tensor.
 - **Operaciones a nivel matricial** (*matrix/tensor-wise*): Opera sobre todos los elementos alojados en el tensor para realizar una salida.

Backend asociado a las Redes Neuronales

Tensorflow

- Framework creado por Google para el trabajo optimizado con tensores dentro de la computación numérica.
- Es uno de los principales frameworks de implementación de redes neuronales, junto con Pytorch.
- Provee de API's estables en Python y C.
- **Paradigma:** Definimos una estructura neuronal en base a un grafo, el cuál es posteriormente compilado, y ejecutado en C++.

Keras

- Una de las principales desventajas de TensorFlow es la curva de aprendizaje.
- Es necesario entender los conceptos de sesiones, grafos, entre otros.
- Keras se posiciona como una capa intermedia entre TensorFlow (y otros frameworks de redes neuronales), teniendo como máxima ser amigable con el usuario.
- Es modular, permitiendo generar prototipos en poco tiempo.

Terminología asociada a las Redes Neuronales

Terminología

- **Neuronas:** Elemento base de una red. Recibe un impulso y devuelve un output procesado por una función de activación
- **Capas:** Un ensamble de neuronas se conoce como capas. Éstas permiten definir cierta secuencialidad en el flujo.
- **Función de activación:** Permiten traducir el output de una neurona/capa.
- **Modelos:** Las capas y neuronas se ensamblan dentro de modelos, que definen el flujo de conexión.
- **Funciones de pérdida/optimización:** Definimos el proceso de optimización a nivel de modelo.

Perceptrones

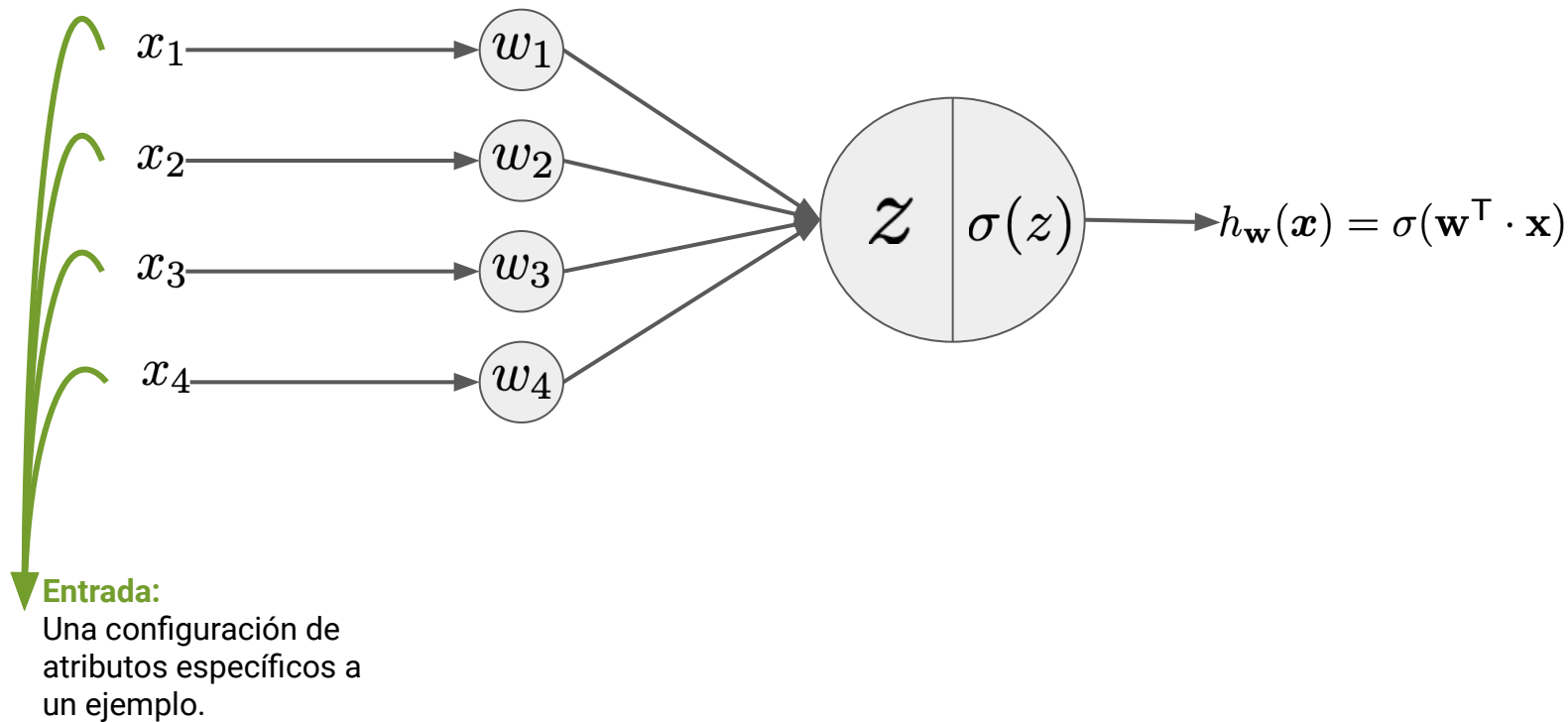
¿Qué es un perceptrón?

Orígenes: Neurona Artificial de McCulloch y Pitts (1943). Un conjunto de señales binarias que generan un output de salida binario.

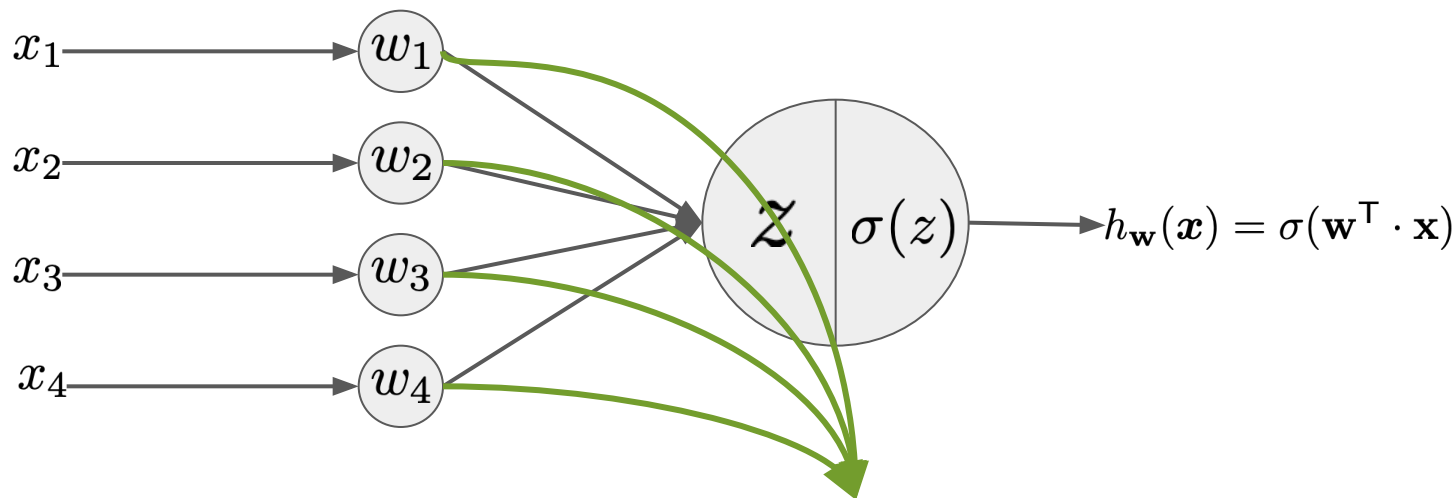
Perceptrón: Desarrollado por Rosenblatt (1957) Flexibiliza el supuesto de las señales binarias a un proceso de suma ponderada entre partes y un output de salida que evalúa el signo de la suma.

Si bien prometía bastante en cuanto a la capacidad de generar cálculos automáticos facilitando la toma de decisiones binarias, no presentaba soluciones para casos binarios no-lineales.

Anatomía de un Perceptrón



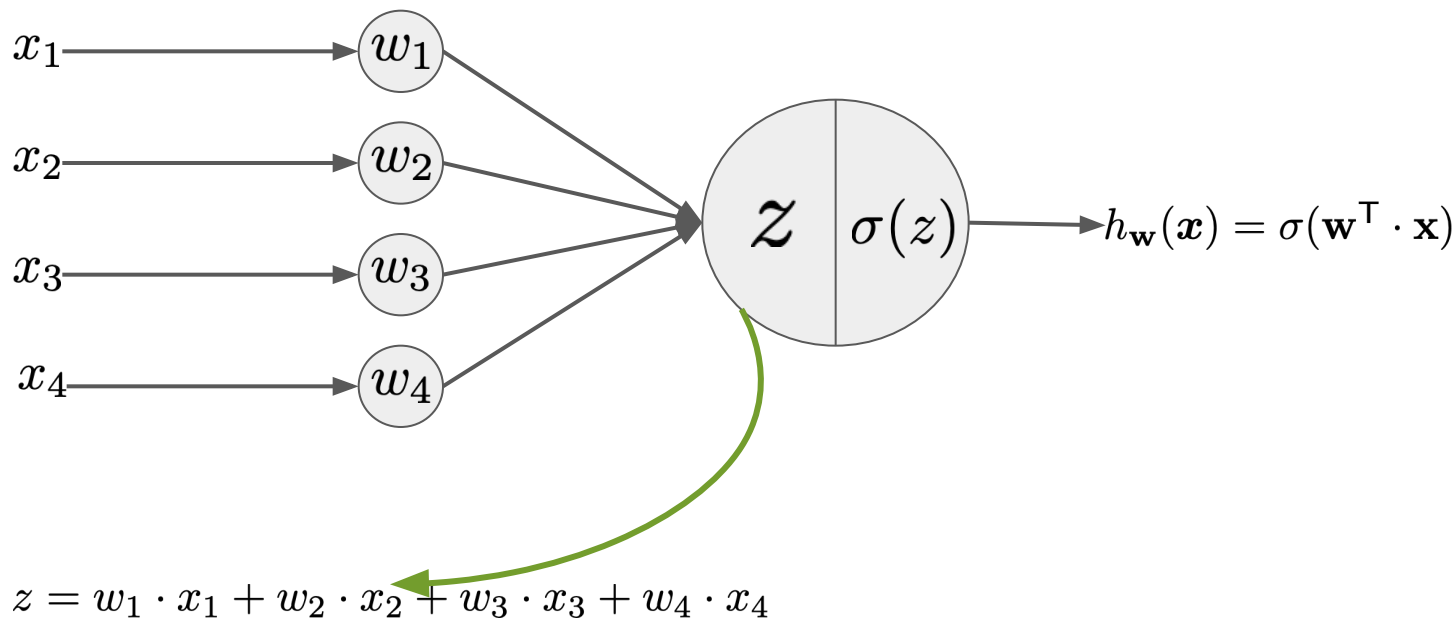
Anatomía de un Perceptrón



Pesos:

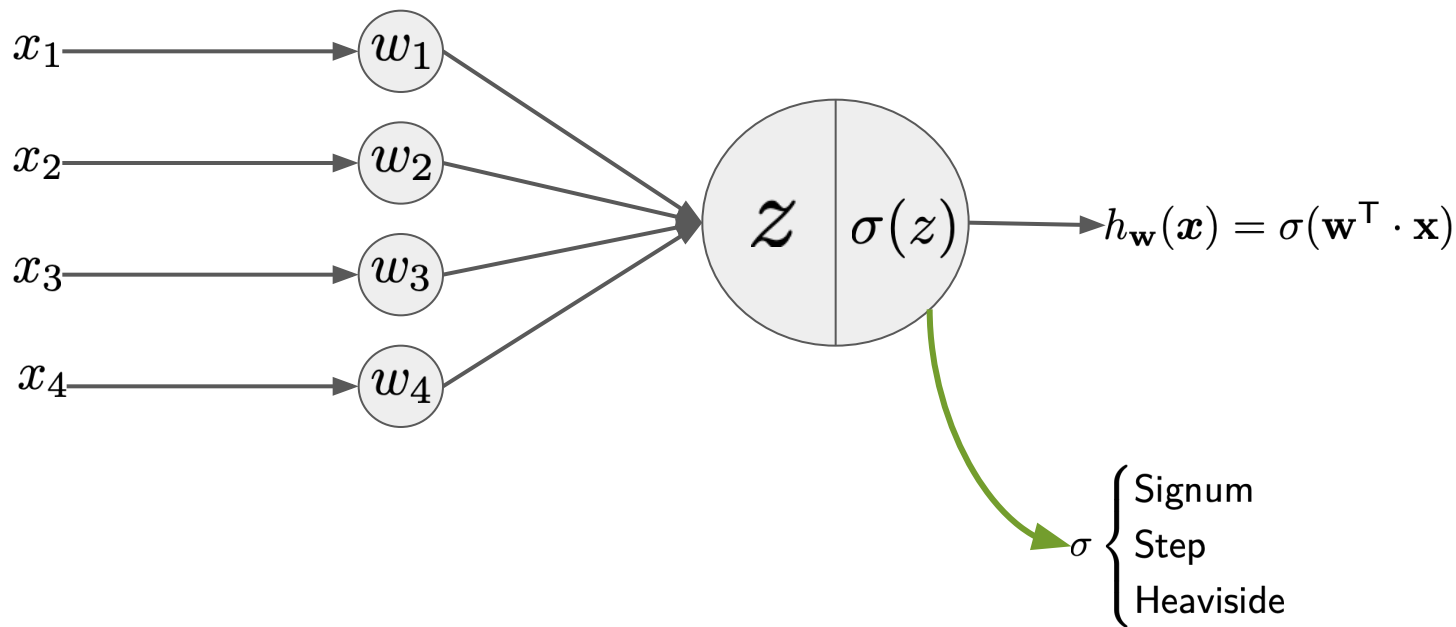
Se aprenden en base a SGD, y se refuerzan en base a las observaciones incorrectamente clasificadas.

Anatomía de un Perceptrón



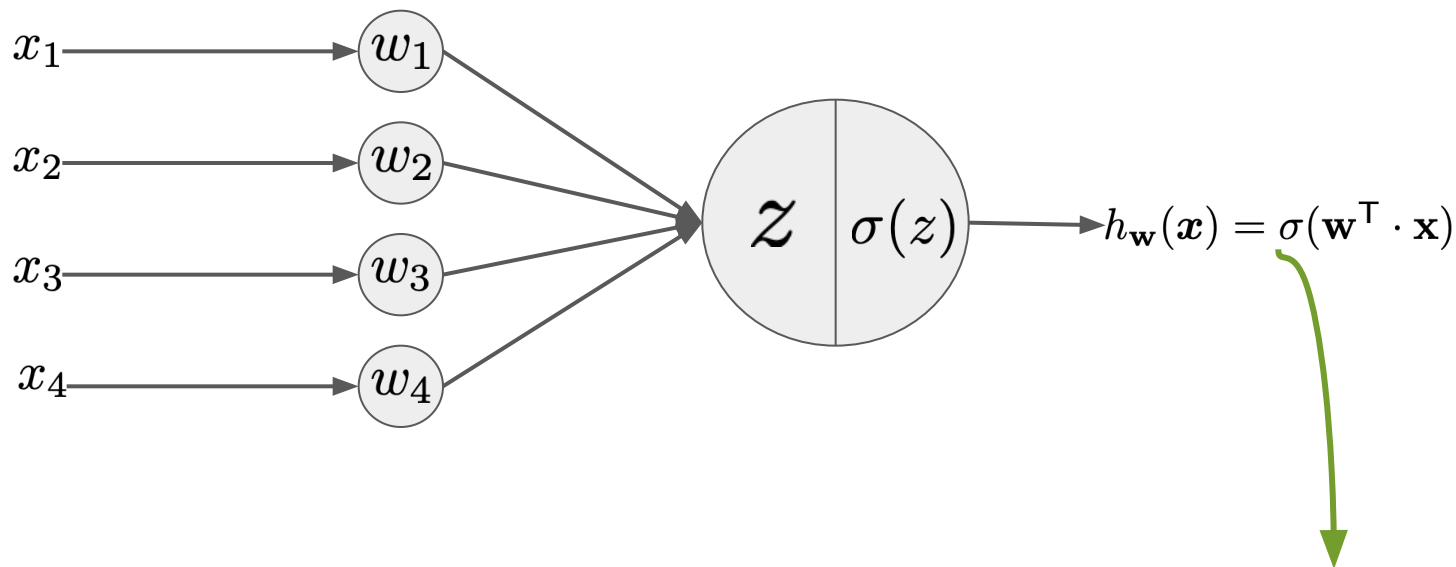
Paso Lineal: Generamos la suma ponderada entre un input x y su peso asociado w .

Anatomía de un Perceptrón



Paso Umbral: Evaluamos el valor de la suma ponderada mediante una de las siguientes funciones.

Anatomía de un Perceptrón



Salida: Procesamos la suma ponderada con la función de activación, dando paso a una clase binaria

Proceso de decisión de un Perceptrón

1

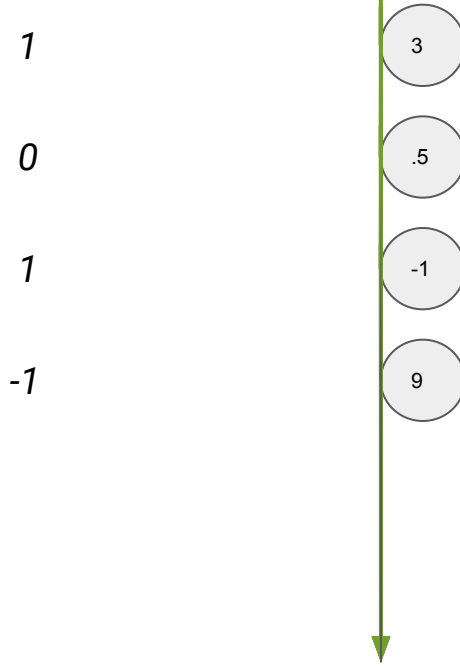
0

1

-1

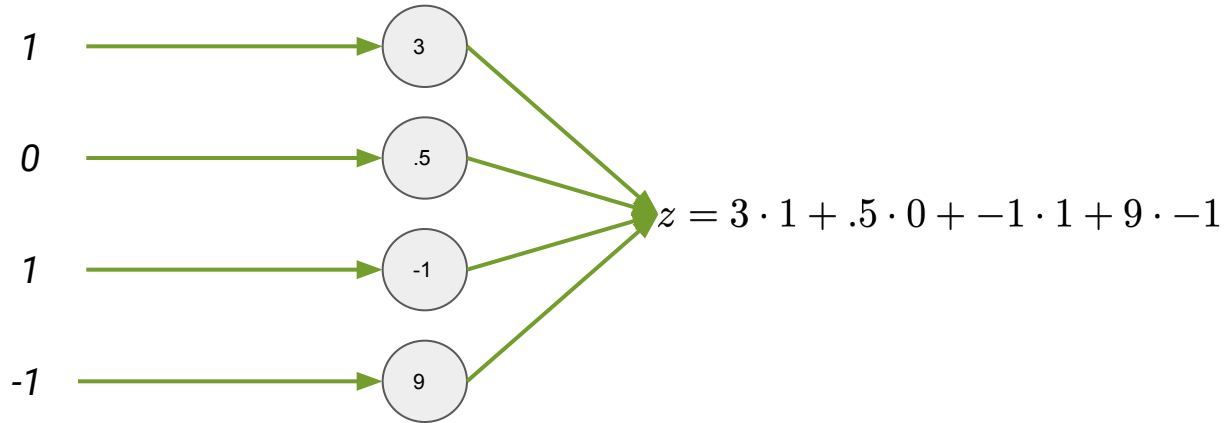
Ingreso del ejemplo: Preservamos los valores de cada atributo en el ejemplo.

Proceso de decisión de un Perceptrón



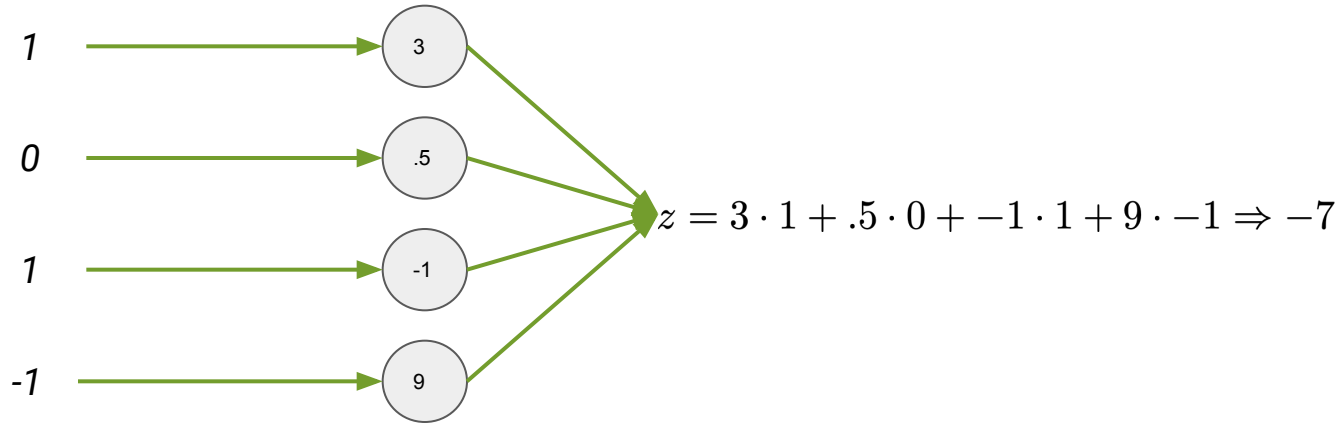
Asignación de pesos: Dado un conjunto de pesos, los asociamos a cada atributo en el ejemplo.

Proceso de decisión de un Perceptrón



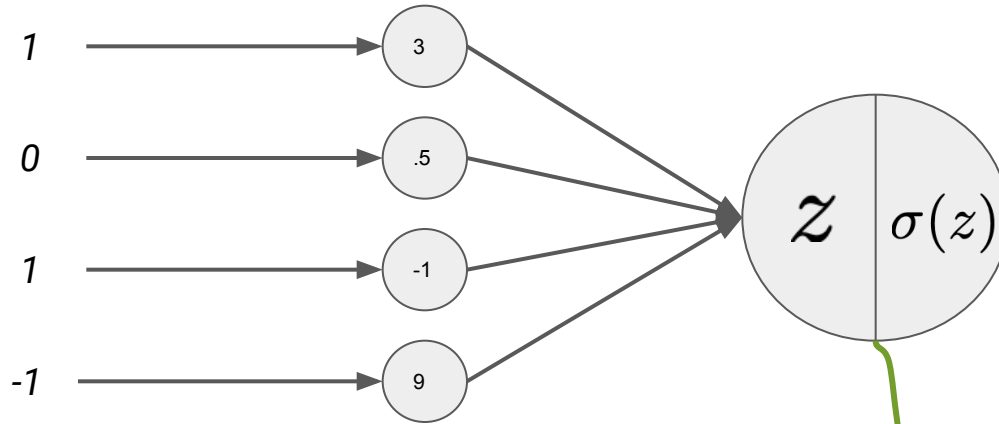
Suma ponderada de los atributos: Ajustamos cada atributo por los pesos y posteriormente sumamos

Proceso de decisión de un Perceptrón



Suma ponderada de los atributos: Ajustamos cada atributo por los pesos y posteriormente sumamos

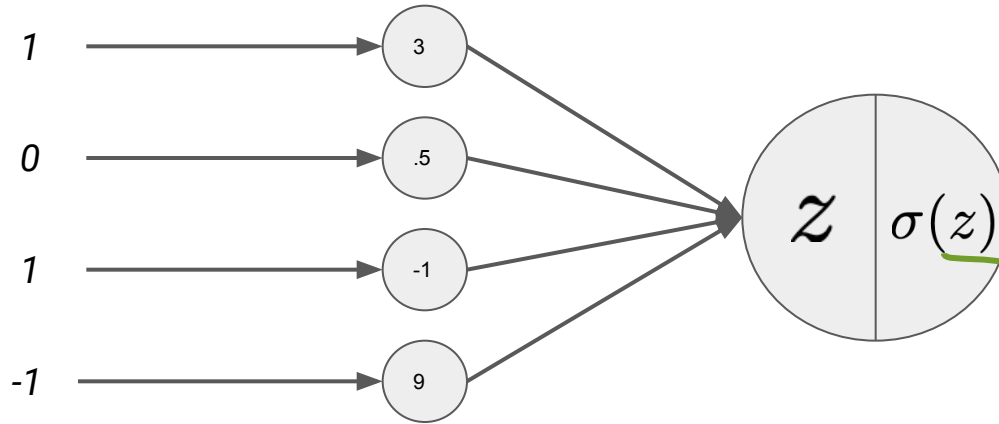
Proceso de decisión de un Perceptrón



$$z = 3 \cdot 1 + .5 \cdot 0 + -1 \cdot 1 + 9 \cdot -1 \Rightarrow -7$$

Paso a la función de evaluación: Evaluamos el signo/magnitud de la suma ponderada

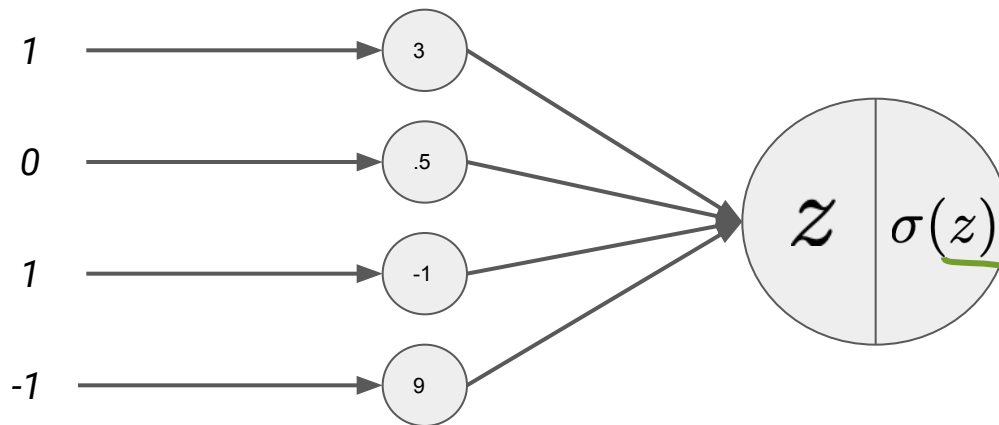
Proceso de decisión de un Perceptrón



Paso a la función de evaluación: Evaluamos el signo/magnitud de la suma ponderada.

$$\text{Signum}(-7) = \begin{cases} 1 & \text{si su signo es positivo} \\ 0 & \text{si su magnitud es cero} \\ -1 & \text{si su signo es negativo} \end{cases}$$

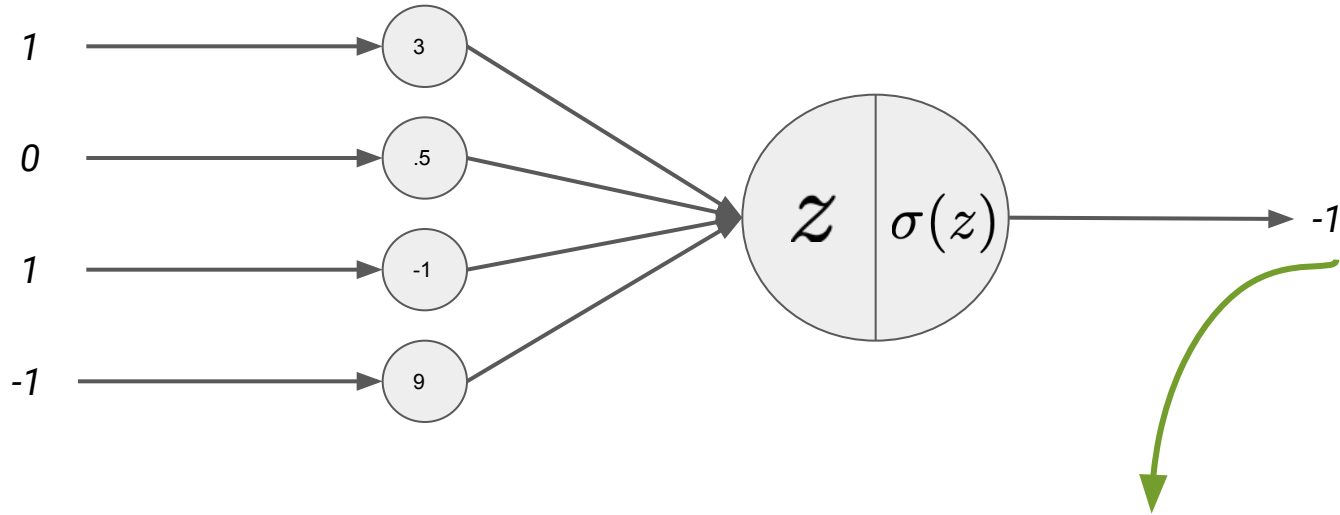
Proceso de decisión de un Perceptrón



Paso a la función de evaluación: Evaluamos el signo/magnitud de la suma ponderada.

$$\text{Signum}(-7) = \begin{cases} 1 & \text{si su signo es positivo} \\ 0 & \text{si su magnitud es cero} \\ -1 & \text{si su signo es negativo} \end{cases}$$

Proceso de decisión de un Perceptrón



Salida: Asignamos la clase para el ejemplo evaluado.

{desafío}
latam_

*Academia de
talentos digitales*

www.desafiolatam.com