

**MBA  
USP  
ESALQ**

# Conceptos básicos para Introducción al Deep Learning

Prof. Dr. Jeronimo Marcondes

# Introducción

- Plan de ataque:
  - 1) Explicar el concepto
  - 2) Como fluyen los datos
  - 3) Como es el proceso de optimización
  - 4) Función costo y descenso del gradiente
  - 5) Parte práctica

# Introducción

- Problema de clasificación y regresión

$$y = f(X1, X2, \dots X3)$$

- Machine Learning es una tecnología donde las computadoras tienen la capacidad de aprender de acuerdo con las respuestas esperadas por medio de asociaciones de diferentes datos, los cuales pueden ser imágenes, números y todo lo que esa tecnología pueda identificar.

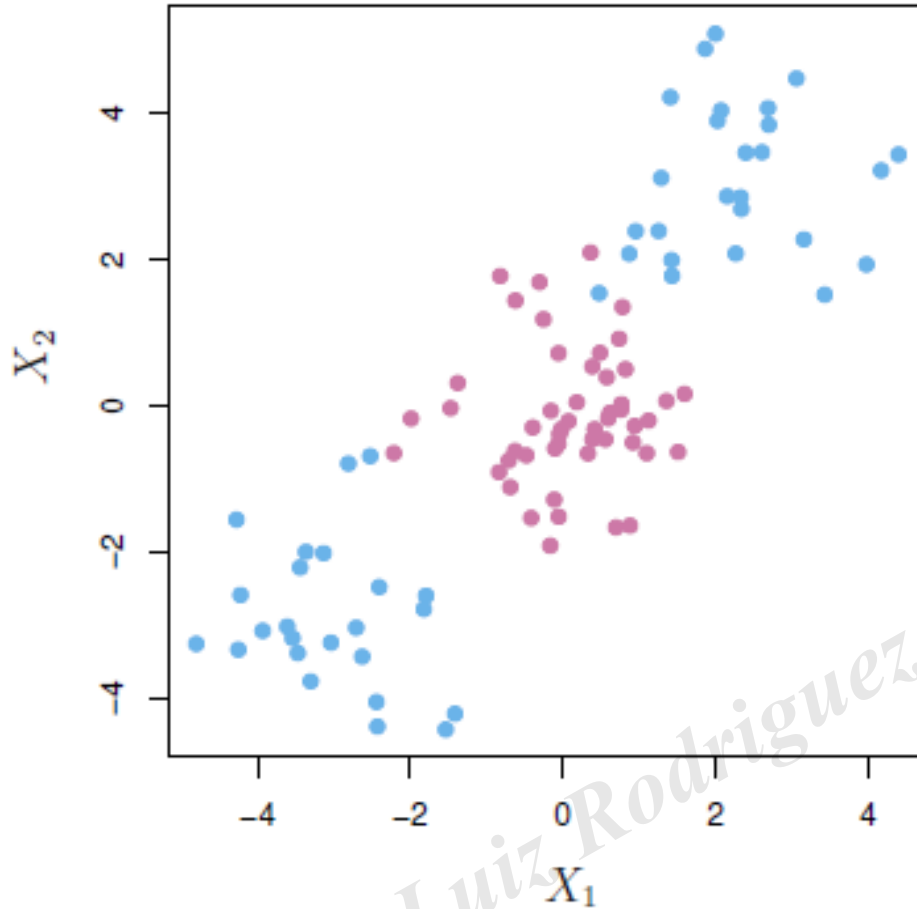
# Ejemplos

Es posible hacer modelos para:

clasificar clientes  
por probabilidad  
de no pagar;

prever el valor de  
una acción;

reducir un  
conjunto de  
datos.



Morettin y Singer – Introducción a la Ciencia de Datos

# AE Supervisado

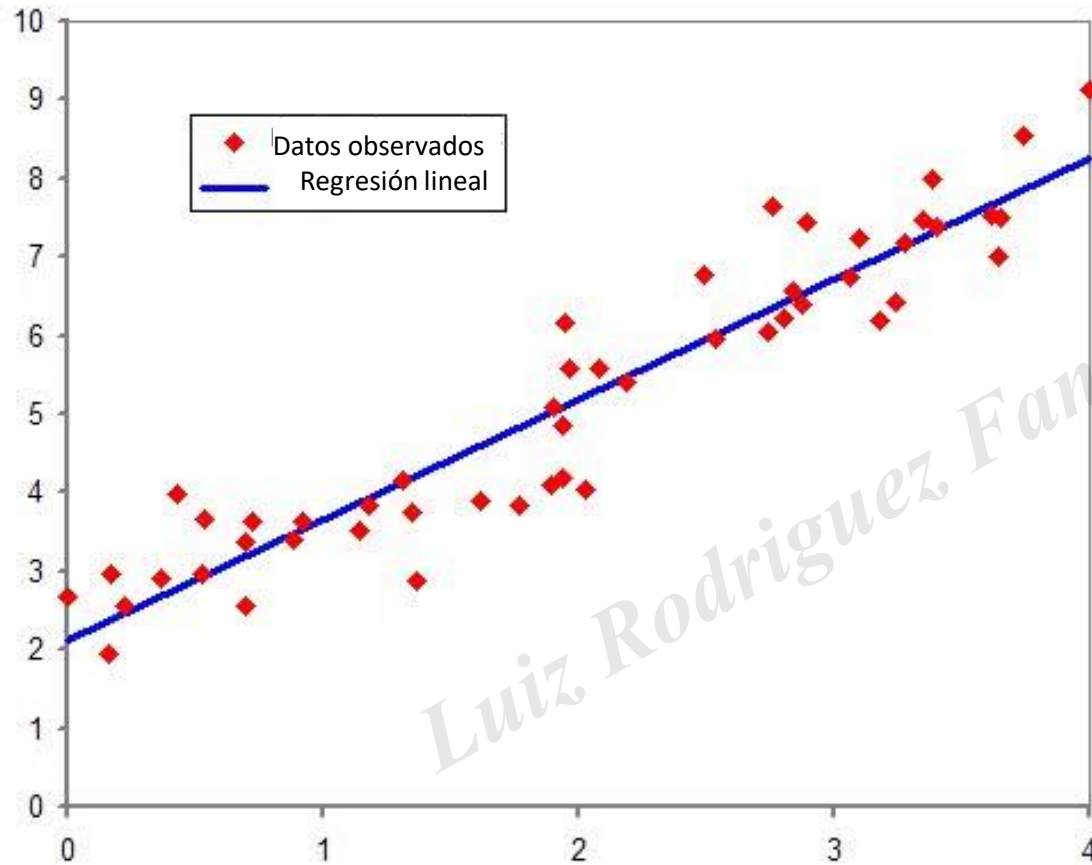
- Problema supervisado
- Regresión:

$$y = f(X) + e$$

$$y = a + bX + e$$

$$y = 0,2 + 0,1X$$

# Problema supervisado



Luiz Rodriguez Fantini 005.374.619-81

# AE No supervisado

- Tenemos apenas un conjunto de variables predictoras (inputs) y el objetivo es describir asociaciones y patrones entre esas variables. En ese caso, no hay una variable respuesta.
- Análisis de Cluster y Análisis de Componentes Principales.



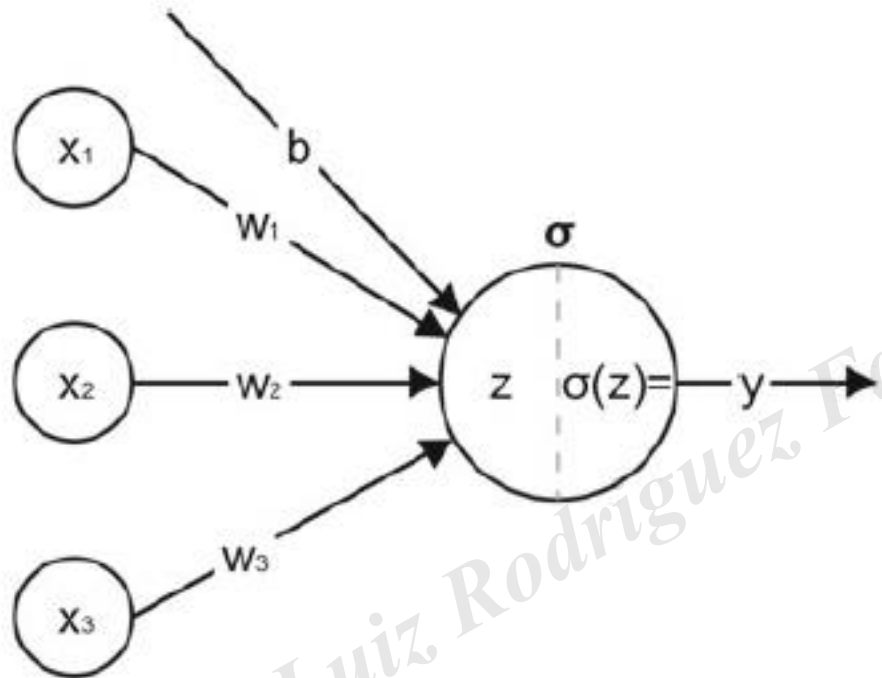
# clasificación

- Respuesta del modelo es una variable cualitativa.
- Ejemplo del riesgo de default.

$$\textit{Default} = f(\textit{Riesgo del Individuo})$$

Luiz Rodrigues Fantini 005.374.619-81

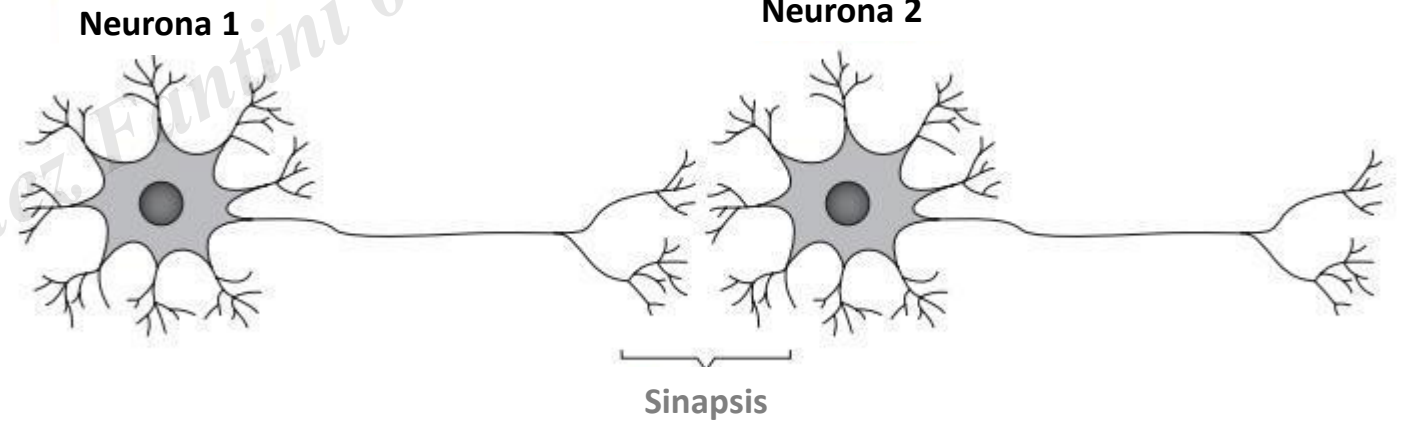
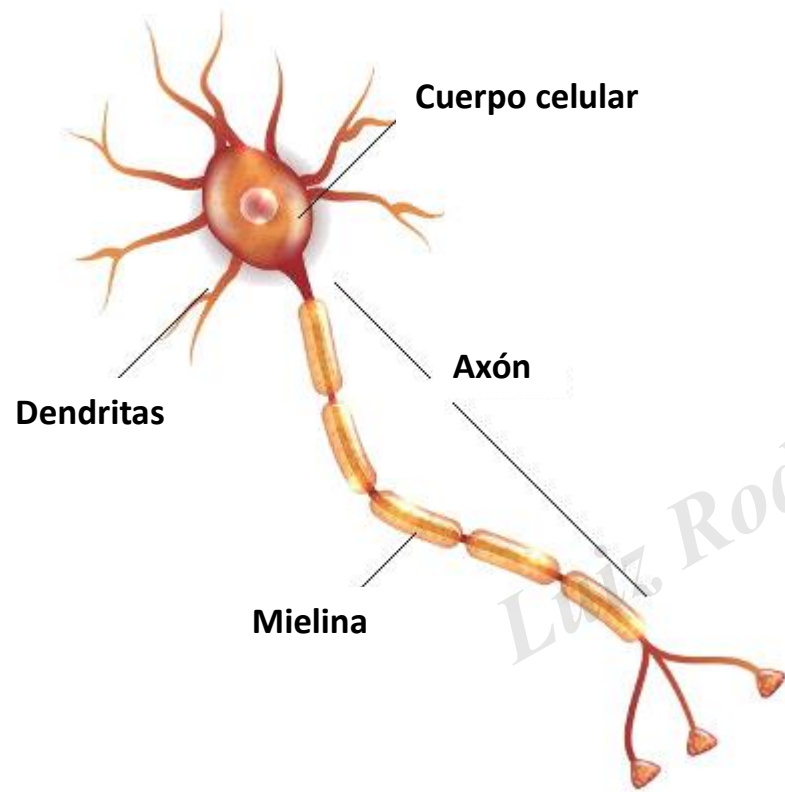
# Regresión como ANN



# Redes Neurales Artificiales

- Deep learning – concepto
- En esa clase comenzaremos con la red neural “rasa”.
- Las contribuciones pioneras para el área de Redes Neurales (RN) fueron las de McCulloch y Pitts (1943), que introdujeron la idea de RN como máquinas computacionales y de Hebb (1949), por postular la primera regla para aprendizaje organizado.

# Redes Neuronales Artificiales

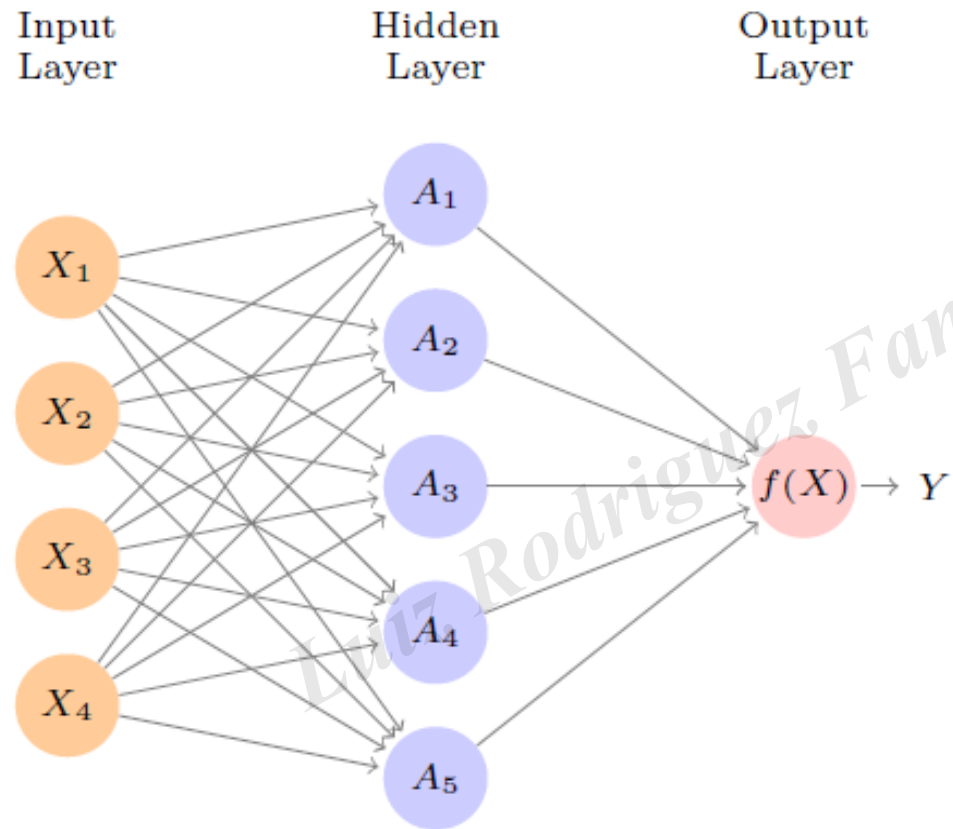


# Redes Neurales Artificiales

- La idea del perceptrón de Rosenblatt.
- Aprende respuestas binarias
- Pesos entrenados para producir vector objetivo

Shiffman (2012) demuestra el Perceptrón como un modelo computacional de una simple neurona, el cual consiste en poseer diversas entradas de datos, un núcleo de procesamiento y una salida, el cual posibilita sólo la salida de valores lógicos.

# Redes Neurales Artificiales



# Redes Neurales Artificiales

- Red neural del tipo feedforward.
- Neuronas no se interconectan.
- Capas de entrada no puede ser menor que la cantidad de variables que estamos utilizando para explicar Y.

Luiz Roberto Fantini 005.374.619-81

# Cómo funciona

- Cada Neurona recibe datos de entrada o de la capa anterior.
- Diferente del caso de regresión — sólo tenía una capa de entrada y salida.
- Cada Neurona al enviar información para la próxima multiplica la información por **pesos**.



# Matemática

- Dado el valor de n entradas:

$$X_1 = [X_{11}, X_{12} \dots X_{1n}]$$

¡Esas son nuestras variables explicativas! Por ejemplo, el riesgo de default del cliente:

$$X = [\textit{¿ Pagó en el pasado? , Renta Actual, etc}]$$

# Matemática

- Dado el valor de salidas para  $t$  clientes:

$$Y = [Y_1, Y_2, \dots, Y_t]$$

Esas son nuestras variables explicadas. O sea, si un determinado cliente efectivamente pagó.

Luiz Rodriguez Fantini 005.374.619-81

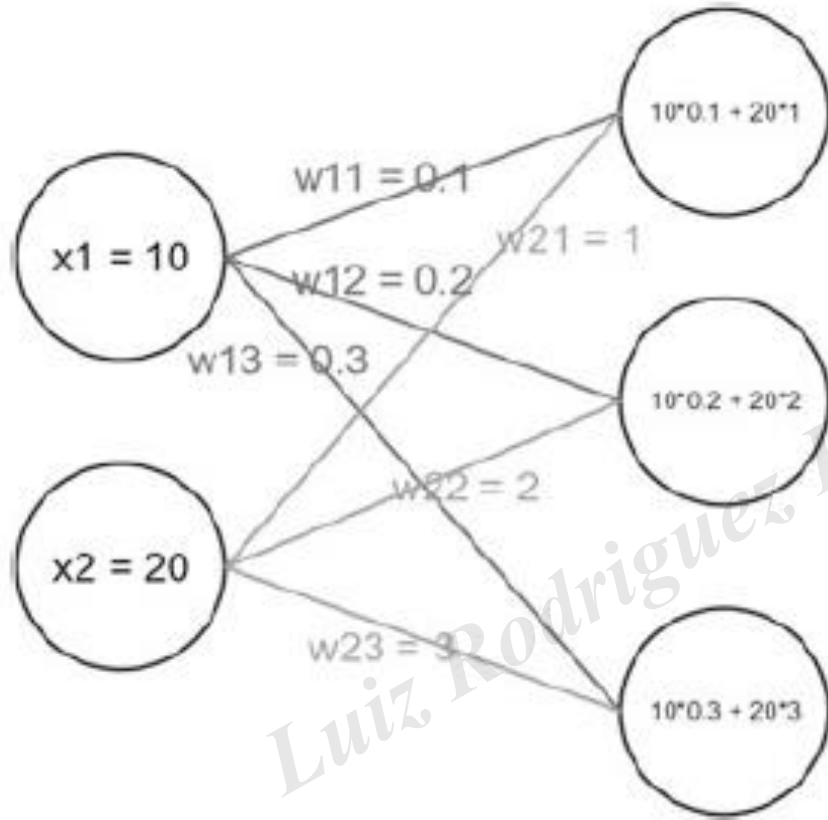
# Matemática

- Dado el valor de pesos para  $h$  neuronas de la próxima capa:

$$w = [w_{11}, w_{12}, \dots, w_{nh}]$$

Esos son los valores que serán multiplicados por cada variable de entrada hasta la próxima capa oculta.

# Visualización



# Matemática

- Para cada una de las neuronas:

$$z = b + \sum_i w_i X_i$$

Lo que es equivalente, en el caso de la primera neurona, a:

$$z_1 = b + w_{11}X_1 + w_{21}X_2$$

# Función de activación

- Supongamos que estamos en problema de clasificación de default con resultados posibles 1 o 0.
- 1 va a dar morosidad y 0 no.
- Nuestra variable explicada es dicotómica y se trata de un problema de clasificación.

# Función de activación

La función de activación es aquella que procesa la señal generada por la combinación lineal de las entradas y de los pesos de las sinapsis, para generar la señal de salida de la neurona.

O sea, es la que hace el “procesamiento” de la información.

Podemos tener respuestas lineales y no lineales.

# Función de activación

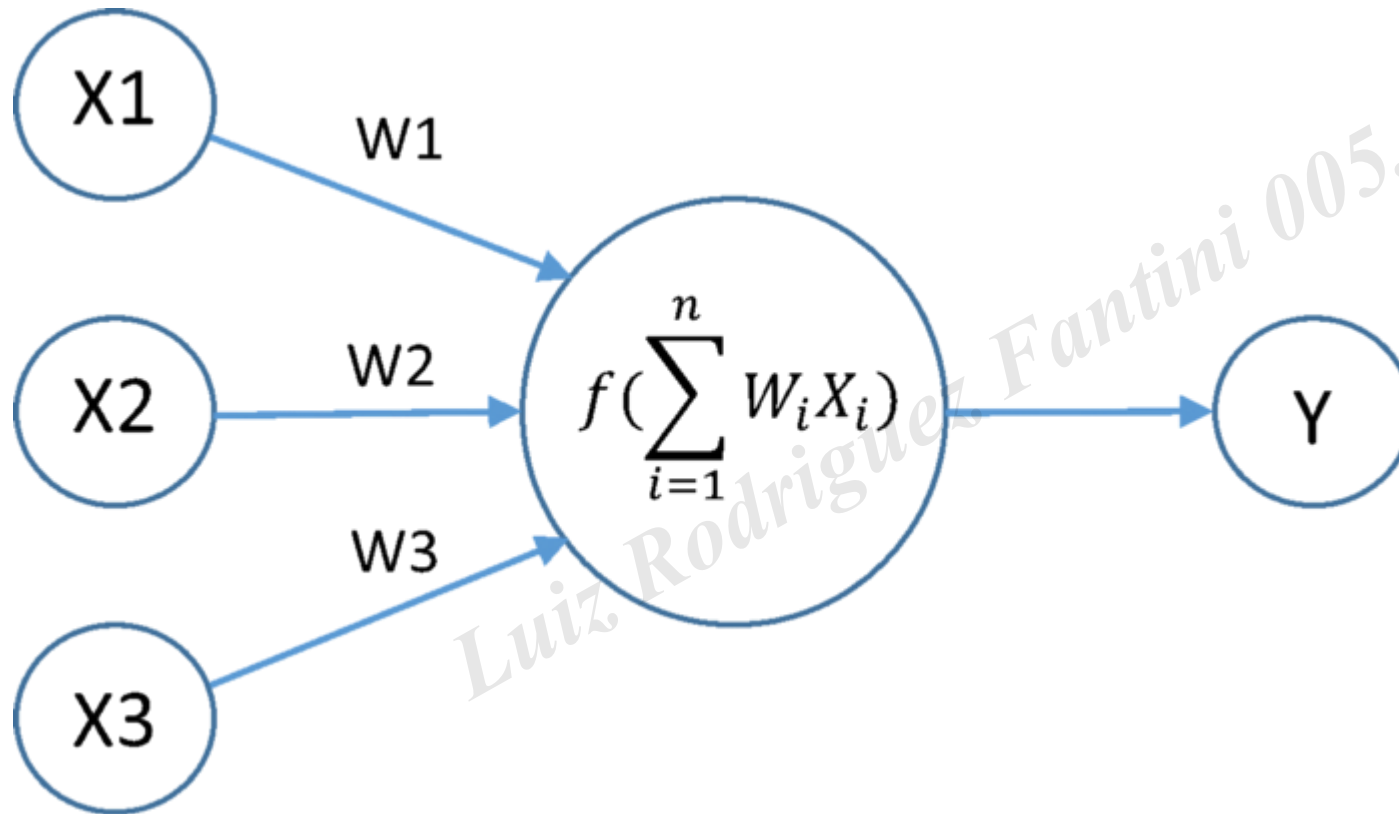
- Para decidir el resultado recibido de la neurona anterior tenemos (ReLU):

$$f(z) = \begin{cases} 1 & \text{si } z \geq 0 \\ 0 & \text{caso contrario} \end{cases}$$

¿Cuál es la diferencia para una regresión?



# Función de activación



# Función de activación

$$z_1 = w_{11}X_1 + w_{21}X_2 = 1.2$$

$$\text{Relu}(z_1) = 1$$

Comparando con la tabla:

$$y - \text{Relu}(z_1)$$

¿Qué hacer si salió mal??

# Función de activación

- Otro ejemplo:
- Función Sigmoides:

$$g(z) = \frac{e^z}{e^z + 1}$$

¿Y si fuera una función lineal?

# Cómo funciona

- Peso para multiplicar la información depende del par origen\destino.
- Resultado es insertado dentro de una función de activación.
- Al depender del resultado, tal como en la neurona, la señal es disparada para la próxima neurona.

# Cómo funciona

- ¿Cómo determinar los pesos?
- Iniciado de forma randómica o con alguna regla preespecificada.
- Caso de la estimación de valores de las compras.

# Explicación

- Usted sabe la cantidad que compró.
- Usted no sabe el precio por kg.
- Usted sabe el total que gastó.

$$gasto = ppk_{uva}c_{uva} + ppk_{manzana}c_{manzana}$$

# Explicación

- Supongamos que gastó 10.
- Recibió 1 kg de cada.
- Usted había estimado que el ppk de la uva es de R\$ 4.00 y el de la manzana es de R\$ R\$ 5.00. Usted estima antes de saber el valor total.
- Usted se ha equivocado por  $10 - 9 = \text{R\$}1.00$ . ¡Error residual!

# Explicación

- Supongamos que gastó 10.
- Recibió 1 kg de cada.
- Usted había estimado que el ppk de la uva es de R\$ 4.00 y el de la manzana es de R\$ R\$ 5.00. Usted estima antes de saber el valor total.
- Usted se ha equivocado por  $10 - 9 = \text{R\$}1.00$ . ¡Error residual!



# Ajuste de los Pesos

- *error residual<sub>j</sub> = valor real<sub>j</sub> – valor estimado<sub>j</sub>*
- $\Delta w_j = \eta \nabla E$

Luiz Rodriguez Fantini 005.374.619-81

# Función Costo

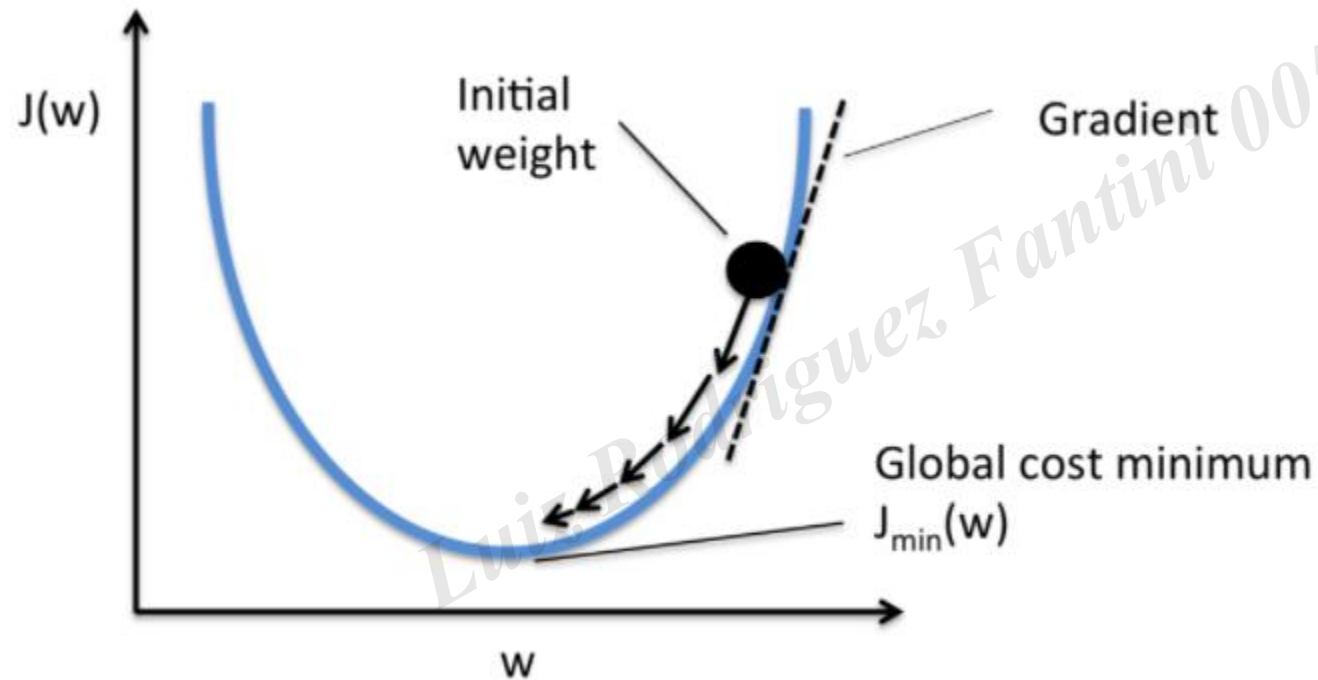
- Función que mide lo que estamos acertando.
- Por ejemplo: error cuadrático medio.

$$E = \frac{1}{2} \sum_{\text{entrenamiento}} (t^n - y^n)^2$$

# Función Costo

	Valor estimado	Valor real	Error	Cuadrado del error
precio 1	2	2	0	0
precio 2	3	5	-2	4
precio 3	2	4	-2	4
precio 4	5	1	4	16
suma				24
				6

# ¿Cómo es realizado el ajuste?



Fuente: Deep Learning book

# ¿Cómo es realizado el ajuste?

- Estime un valor inicial para el peso
- Calcule los valores de salida para esos pesos
- Utilice la fórmula de ajuste para obtener los nuevos pesos:

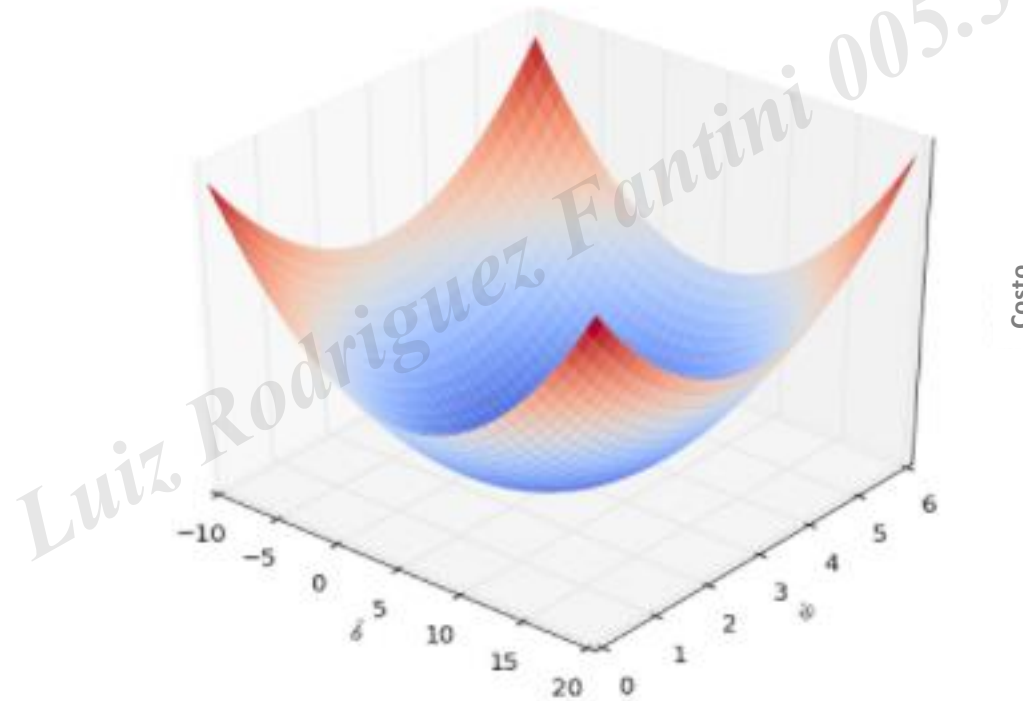
$$\text{nuevo peso} = \text{peso anterior} - \eta \nabla E$$

# Descenso del gradiente

- Proceso utilizado para encontrar el mínimo de la función costo con base en variación en los pesos.
- A través del proceso de backpropagation los errores obtenidos después de los cálculos de los pesos son retroalimentados en la red hasta que optimizamos el resultado.

# Descenso del gradiente

- Matemáticamente: ¿cuál es la variación en los pesos que reduce más mi diferencia entre el valor estimado y el valor real?



Fuente: <https://matheusfacure.github.io/2017/02/20/MQO-Gradiente-Descendente/>

# Descenso del gradiente

- Uso del cálculo diferencial:

El gradiente va a darnos las variaciones en la función pérdida para pequeñas alteraciones en los parámetros:

$$\nabla(L) = \left[ \frac{\partial L}{\partial w}, \frac{\partial L}{\partial b} \right]$$



# Parte Práctica

*Luiz Rodriguez Fantini 005374.619-81*