Fundamentos de Redes Neuronales

Dioney Alberto Contreras Sanchez
Noviembre 2023

Agenda

Redes Neuronales

01.- Aprendizaje Automático

02.- Redes Neuronales

03.- Conceptos Importantes

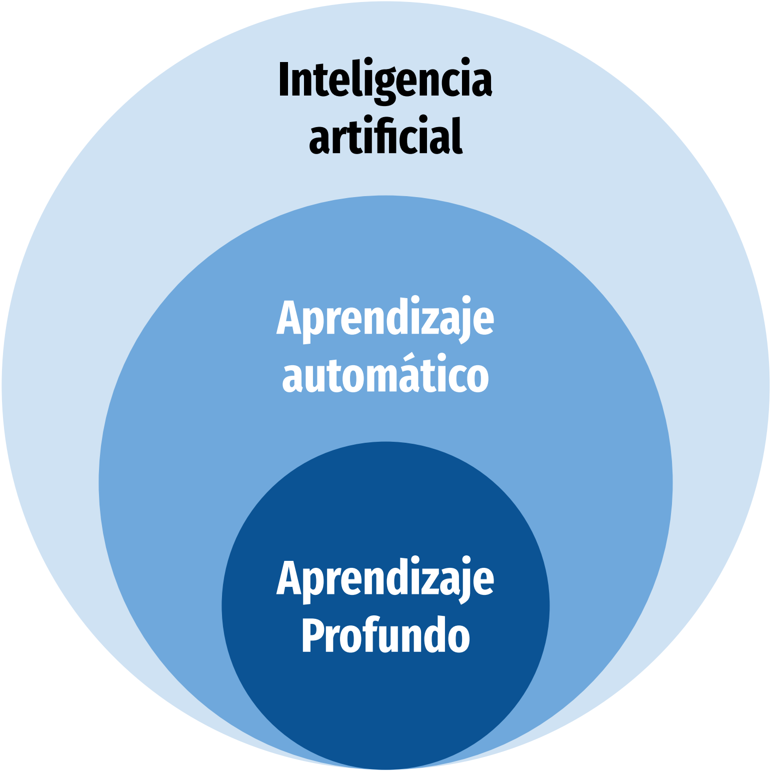
04.- Retropropagación

CENIC® Centro de Investigación Coppel

01. Aprendizaje automático

01. Aprendizaje automático

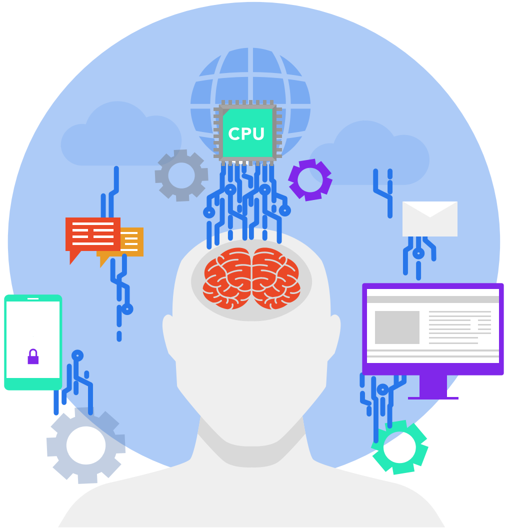
Inteligencia Artificial



CENIC® Centro de Investigación Coppel

01. Aprendizaje automático

Definición



Aprendizaje Automático

"Es un área de la inteligencia artificial que dota a los sistemas digitales a aprender de características y adquirir información relevante para el usuario"

Aprendizaje Profundo

"Es una rama del aprendizaje automático en donde se utilizan algoritmos de redes neuronales para resolver tareas computacionalmente complejas".

01. Aprendizaje automático

Aprendizaje Automático

Aprendizaje Automático

- **Supervisado**: Usan datos etiquetados para aprender de ellos.
- **No supervisado**: No utiliza datos etiquetados y aprende directamente de los datos.
- **Refuerzo**: Usa un sistema de recompensas para realizar aprendizaje

01. Aprendizaje automático

Aprendizaje Automático

Supervisado	No Supervisado
- Clasificación de imágenes	- Agrupamiento (Clustering)
- Predicción de Ventas	- Detección de patrones
- Detección de fraude	- Reducción de dimensionalidad
- Detección de Spam	- Segmentación

CENIC® Centro de Investigación Coppel

01. Aprendizaje automático

Regresión Lineal

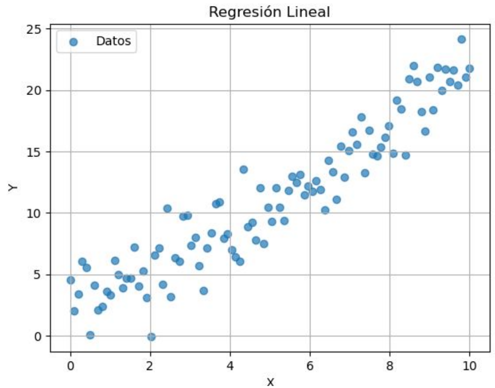
Es un método de modelado estadístico que busca determinar la relación entre una variable dependiente y las variables independientes.

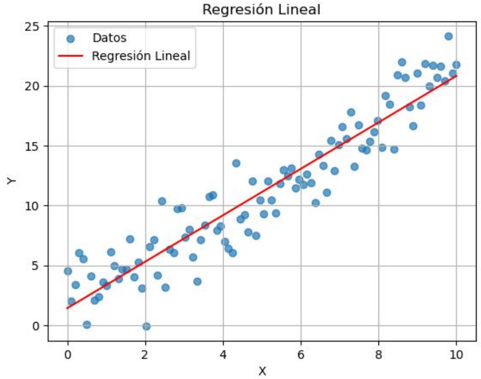
Se define como:

$$\hat{y} = b_0 + b_1 x_1 + \cdots + b_m x_i$$

donde:

$\hat{y}$: Es el valor de la estimación.
b_0 : Es el intercepto de la recta.
b_m: Es el coeficiente de pendiente de la recta.
x_i: Es la i-ésima variable independiente.





CENIC® Centro de Investigación Coppel

La regresión logística es un método estadístico que se usa para modelar la probabilidad de un evento binario, es decir, un evento que puede tener dos resultados posibles.

Se define como:

$$P(Y=1) = \frac{1}{1+e^{-\beta_0+\beta_1 x_1+\cdots+\beta_m x_i}}$$

donde:

$P(Y=1)$: Probabilidad de que la estimación sea de una clase.
b_0 : Es el intercepto de la recta.
b_m : Es el coeficiente de pendiente de la recta.
x_i : Es la i-ésima variable independiente.

02. Redes Neuronales

02. Redes Neuronales

Eventos relevantes

1943

Warren McCulloch y Walter Pitts desarrollan el primer modelo conceptual de las neuronas artificiales.

1957

Frank Rosenblatt desarrolla el perceptrón, un modelo de red neuronal de una sola capa.

1986

David Rumelhart, Geoffrey Hinton y Ronald Williams publican un artículo sobre la retropropagación.

1990

Desarrollo de las redes neuronales convolucionales y recurrentes para tareas más complejas

1998

Desarrollo de las redes LSTM, lo que ayuda para el problema del aprendizaje a largo plazo.

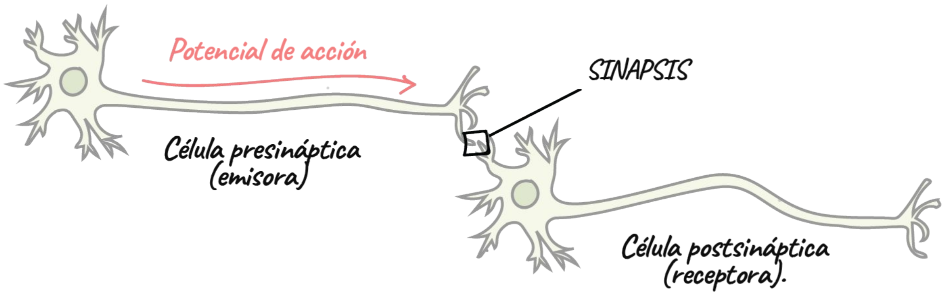
2020

Auge de los modelos basados en Transformers como GPT (Generative Pre Trained Transformers)

02. Redes Neuronales

Definición

Las redes neuronales son algoritmos inspirados en el funcionamiento de las neuronas biológicas.

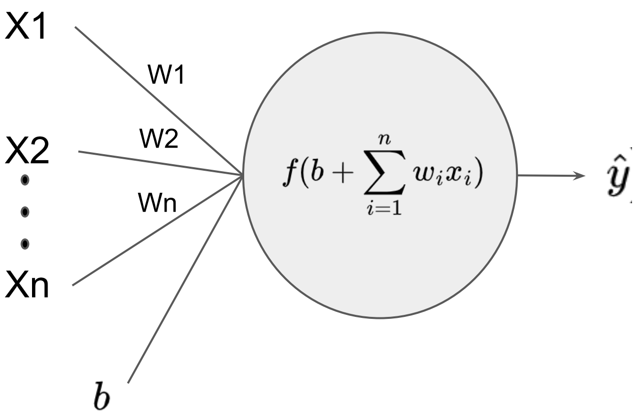


Representación del traspaso de información Neuronal

02. Redes Neuronales

Estructura Básica de una red Neuronal

La neurona artificial



Es la unidad básica de procesamiento de las redes neuronales.
Está compuesto por pesos w_i datos de entrada x_i suma ponderada $\sum_{i=1}^{n} w_i x_i$ y una función de activación f que generan la salida $\hat{y}$

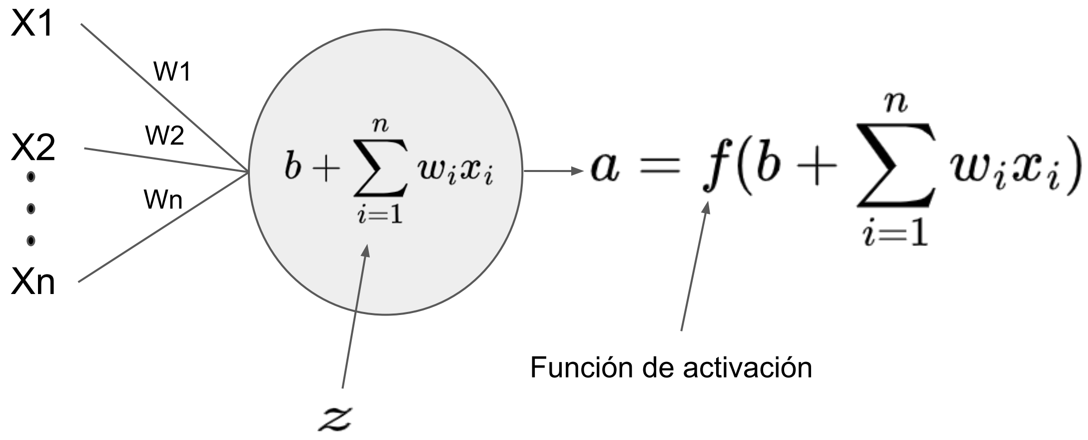
Donde:

- x_i Es la i-ésima variable predictora.
- w_i Pesos sinápticos
- n Es el número de variables independientes
- b Es el sesgo
- f Función de activación
- $\hat{y}$ Salida de la neurona

02. Redes Neuronales

Estructura Básica de una red Neuronal

La neurona artificial

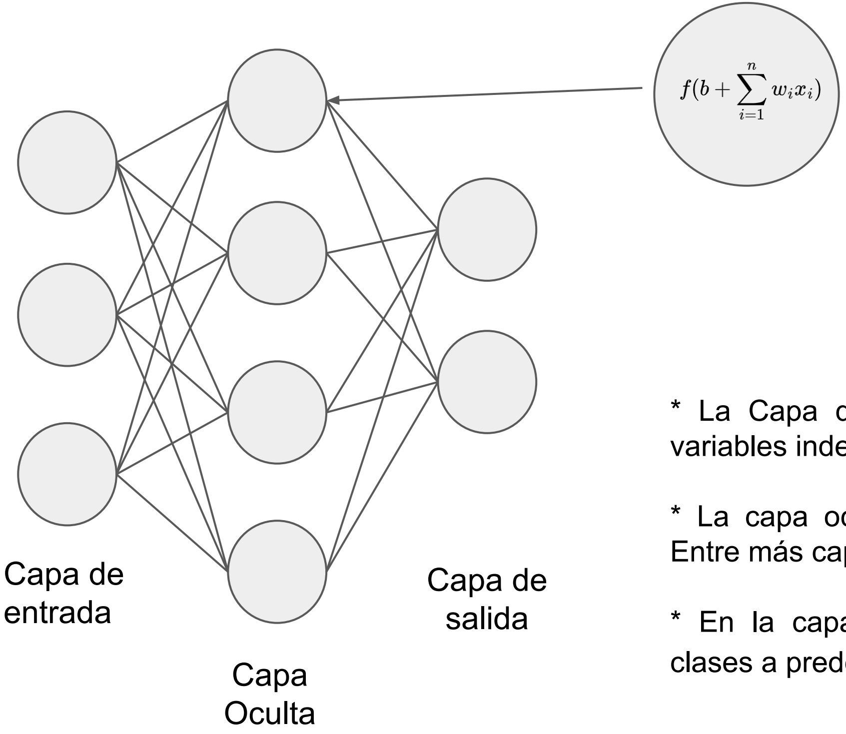


Es la unidad básica de procesamiento de las redes neuronales.
Está compuesto por pesos w_i datos de entrada x_i suma ponderada $\sum_{i=1}^{n} w_i x_i$ y una función de activación f que generan la salida $\hat{y}$

CENIC® Centro de Investigación Coppel

02. Redes Neuronales

Perceptrón Simple



Es la estructura más básica de una red neuronal.

Consta de Una capa de entrada, capa Oculta y una capa de salida,

* La Capa de entrada tiene tantas neuronas como variables independientes tiene el conjunto de datos.

* La capa oculta control la complejidad del modelo. Entre más capas ocultas más complejo será el modelo.

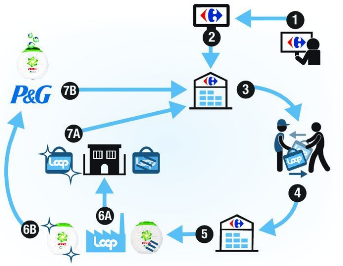
* En la capa de salida hay tantas neuronas como clases a predecir.

02. Redes Neuronales

Aplicaciones de las redes neuronales

[1] Medicina

[2] Detección de fraudes



[3] Cadena de suministros

[1]. Faust, O. *et al.* (2018) 'Deep learning for healthcare applications based on Physiological Signals: A Review', *Computer Methods and Programs in Biomedicine*, 161, pp. 1–13. doi:10.1016/j.cmpb.2018.04.005.

[2] Motie, S. and Raahemi, B. (2023) 'Financial fraud detection using graph neural networks: A systematic review', *Expert Systems with Applications*, p. 122156. doi:10.1016/j.eswa.2023.122156.

[3] Han, C. and Zhang, Q. (2020) 'Optimization of supply chain efficiency management based on machine learning and Neural Network', *Neural Computing and Applications*, 33(5), pp. 1419–1433. doi:10.1007/s00521-020-05023-1.

03. Conceptos Importantes

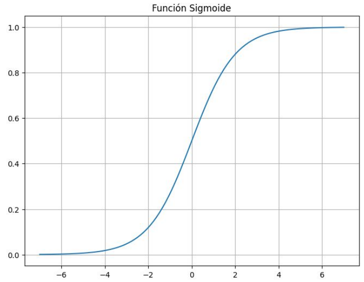
04. Conceptos adicionales

Funciones de activación

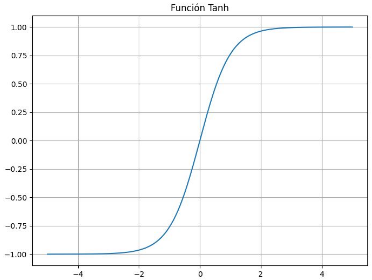
Se utilizan para dotar de no linealidad a la salida de las neuronas y hacer el modelo más complejo

Hay distintas funciones de activación que son usadas dependiendo la finalidad del modelo. Entre ellas está:

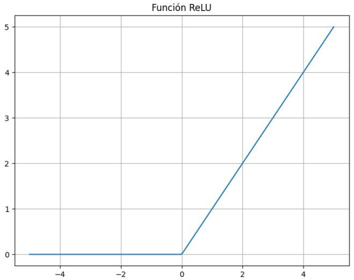
$$sigmoid = \frac{1}{1 - e^{-x}}$$



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



$$ReLU = \max(0, x)$$

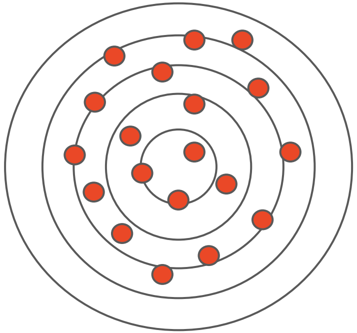


CENIC® Centro de Investigación Coppel

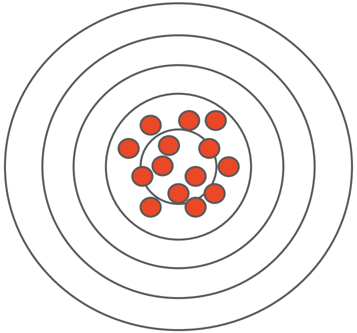
04. Conceptos adicionales

Bias-Variance Tradeoff

Es un concepto importante en el aprendizaje automático. Se refiere a la importancia de mantener un equilibrio entre el sesgo (exactitud del modelo) y la varianza (capacidad del modelo para generalizar)



Alta Varianza - Bajo Sesgo



Alto Sesgo - Baja Varianza

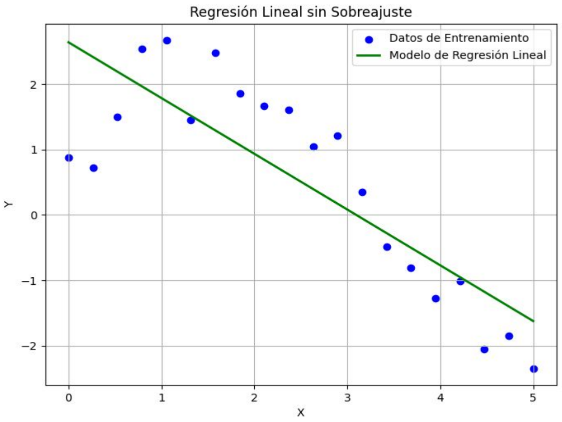
Sesgo: Es el ajuste que se le otorga al modelo hacia los datos al momento de realizar el entrenamiento del mismo.

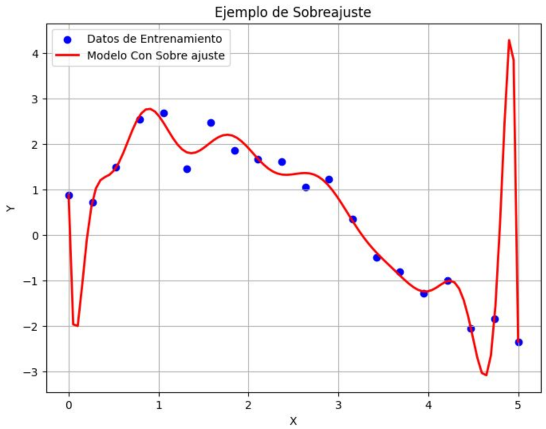
Varianza: Es la sensibilidad del modelo hacia nuevos datos de entrada. También se puede traducir como la capacidad del modelo para generalizar ante observaciones.

04. Conceptos adicionales

Sobreajuste

El sobre ajuste se da cuando un modelo se ajusta demasiado bien a un conjunto de datos, de tal manera que al momento de realizar predicciones sobre datos que no conoce, puede realizar errores.





Características de un modelo con sobre ajuste

- Bajo Sesgo
- Alta Varianza

CENIC® Centro de Investigación Coppel

04. Conceptos adicionales

Regularización

Es un concepto utilizado en aprendizaje automático para tratar el problema del sobreajuste en los modelos

Regularización L1 (LASSO)

agrega la suma de los valores absolutos de los coeficientes a la función de costo.

$$g(w) = \lambda \sum_{i=1}^{n} |w_i|$$

$$J_{L1}(w) = J(w) + \lambda \sum_{i=1}^{n} |w_i|$$

Regularización L2 (RIDGE)

Agrega la suma de los cuadrados de los valores de los coeficientes a la función de costo.

$$g(w) = \lambda \sum_{i=1}^{n} w_i^2$$

$$J_{L2}(w) = J(w) + \lambda \sum_{i=1}^{n} w_i^2$$

04. Conceptos adicionales

Evaluación de modelos

Matriz de Confusión

Es una tabla que nos ayuda a cuantificar el nivel de error de las predicciones de un modelo de aprendizaje supervisado.

	Real Negativo	Real Positivo
Predicho Negativo	TN	FP
Predicho Positivo	FN	TP

Accuracy

$$\frac{TP + TN}{TP + TN + FP + FN}$$

Precision

$$\frac{TP}{TP + FP}$$

Recall

$$\frac{TP}{TP + FN}$$

F1-Score

$$\frac{2 \cdot Precision \cdot Recall}{Precision + Recall}$$

CENIC® Centro de Investigación Coppel

Matriz de Confusión

Es una tabla que nos ayuda a cuantificar el nivel de error de las predicciones de un modelo de aprendizaje supervisado.

	Real Negativo	Real Positivo
Predicho Negativo	TN	FP
Predicho Positivo	FN	TP

Accuracy

$$\frac{TP + TN}{TP + TN + FP + FN}$$

Precision

$$\frac{TP}{TP + FP}$$

Recall

$$\frac{TP}{TP + FN}$$

F1-Score

$$\frac{2 \cdot Precision \cdot Recall}{Precision + Recall}$$

Matriz de Confusión

Es una tabla que nos ayuda a cuantificar el nivel de error de las predicciones de un modelo de aprendizaje supervisado.

	Real Negativo	Real Positivo
Predicho Negativo	9000	20
Predicho Positivo	30	50

Accuracy $\frac{TP+TN}{TP+TN+FP+FN} = \frac{9000+50}{50+9000+20+30} = \boxed{0.99}$

Precision $\frac{TP}{TP+FP} = \frac{50}{50+20} = \boxed{0.71}$

Recall $\frac{TP}{TP+FN} = \frac{50}{50+30} = \boxed{0.62}$

F1-Score $\frac{2 \cdot Precision \cdot Recall}{Precision + Recall} = \frac{2 \cdot 0.71 \cdot 0.62}{0.71+0.62} = \boxed{0.66}$

04. Retropropagación

05. Back Propagation

Descenso por gradiente

Es un algoritmo de optimización usado para encontrar el mínimo local de una función, la cual pueda ser diferenciable, es decir, que se pueda encontrar su derivada, por ejemplo una función de costo.

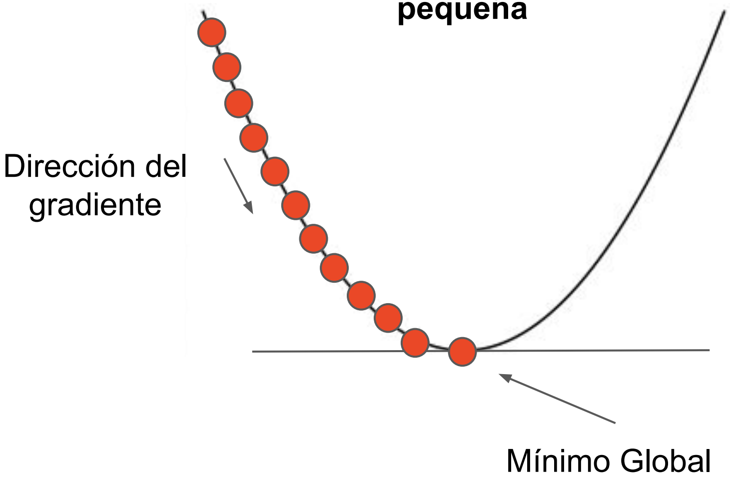
Este algoritmo puede definirse mediante la siguiente ecuación:

$$x_{n+1} = x_n - \alpha \nabla f(x_n)$$

Donde:

x_{n+1}	Nueva posición del gradiente
x_n	Valor del gradiente en la posición actual
α	Tasa de aprendizaje
$\nabla f(x_n)$	Derivada de la función en la posición actual

Tasa de aprendizaje pequeña



05. Back Propagation

Descenso por gradiente

Es un algoritmo de optimización usado para encontrar el mínimo local de una función, la cual pueda ser diferenciable, es decir, que se pueda encontrar su derivada, por ejemplo una función de costo.

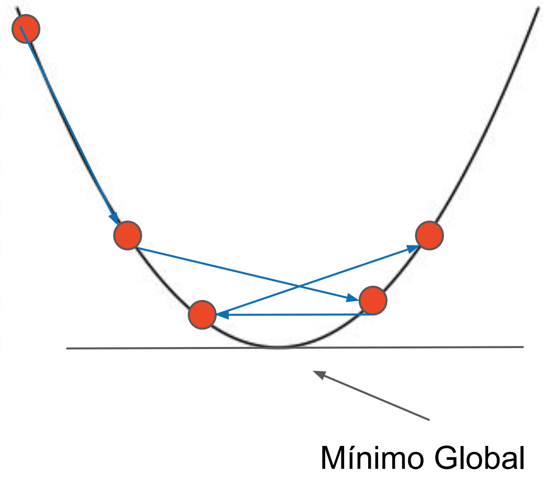
Este algoritmo puede definirse mediante la siguiente ecuación:

$$x_{n+1} = x_n - \alpha \nabla f(x_n)$$

Donde:

x_{n+1}	Nueva posición del gradiente
x_n	Valor del gradiente en la posición actual
α	Tasa de aprendizaje
$\nabla f(x_n)$	Derivada de la función en la posición actual

Tasa de aprendizaje grande



05. Back Propagation

Función de Costo

La función de costo ayuda a evaluar que tan bien se está ajustando el modelo con respecto al conjunto de entrenamiento,

A medida de que el modelo va ajustando los hiper parámetros, la función de costo debe ir disminuyendo hasta un mínimo global.

Error cuadrático Medio (MSE) $$\frac{1}{m}\sum_{i=1}^{m}(y_i - \hat{y})^2$$

Entropía Cruzada $$H(y,\hat{y}) = -\sum_{i} y_i \log(\hat{y}_i)$$

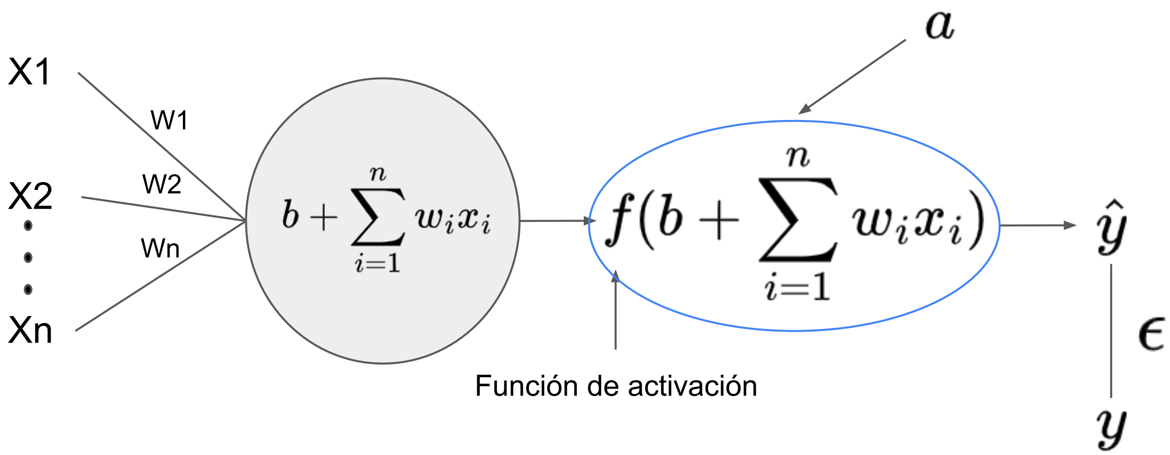
Entropía Cruzada Binaria $$H(y,\hat{y}) = -\left(y \log(\hat{y}) + (1-y)\log(1-\hat{y})\right)$$

CENIC® Centro de Investigación Coppel

05. Back Propagation

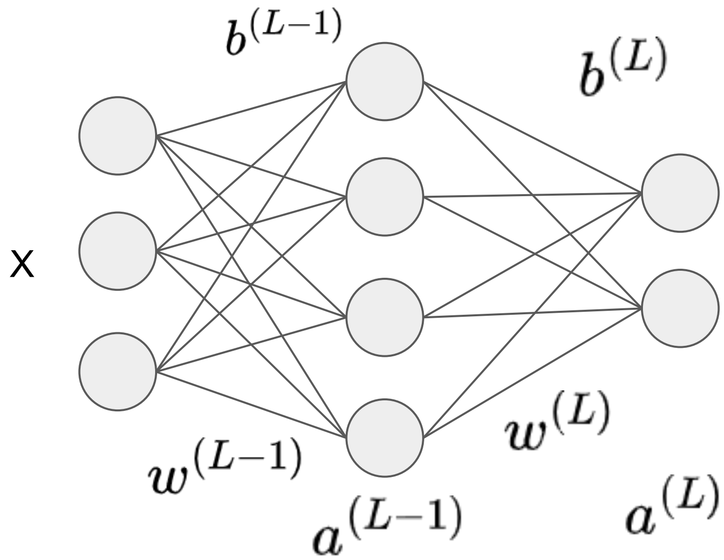
Forward Propagation

La propagación hacia adelante es un proceso que realiza las operaciones lineales y no lineales dentro de las neuronas para encontrar un



05. Back Propagation

Derivación



En **Back Propagation** se calculan los errores de la salida de algoritmo, con la finalidad de generar la actualización de los hiper parámetros que minimicen dicho error.

Algoritmo:

Para n hasta N:

1.- Inicializar los hiper parámetros
2.- Realizar Forward propagation
3.- Calcular la derivada parcial de las funciones de activación con respecto al los **pesos** y al **sesgo**.
4.- Actualizar los pesos y sesgo.

Estableciendo algunos criterios se tiene que:

$$C_0 = (a^{(L)} - y)^2$$

$$a^{(L)} = f(z^L)$$

$$z^{(L)} = b^{(L)} + w^{(L)} a^{(L-1)}$$

Donde:

C_0	Función de costo
$a^{(L)}$	Activación de la capa actual
$z^{(L)}$	Ponderación de parámetros
$b^{(L)}$	Sesgo de la capa actual
$w^{(L)}$	Pesos de la capa actual

Estableciendo algunos criterios se tiene que:

$$C_0 = (a^{(L)} - y)^2$$

$$a^{(L)} = f(z^L)$$

$$z^{(L)} = b^{(L)} + w^{(L)} a^{(L-1)}$$

La idea es encontrar la razón de cambio del costo con respecto a

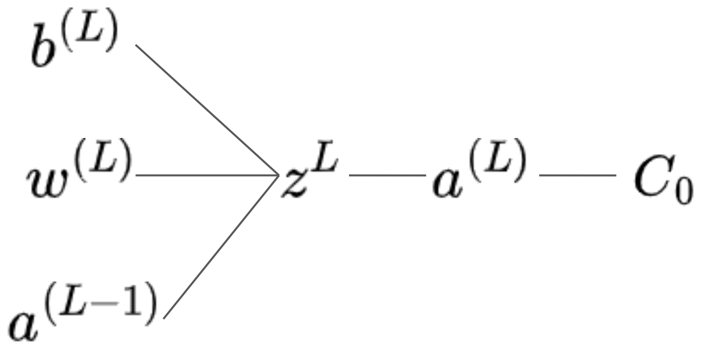
$$w^{(L)} \quad b^{(L)}$$

es decir:

$$\frac{\partial C_0}{\partial w^L}, \frac{\partial C_0}{\partial b^L}$$

05. Back Propagation

Conceptualización



$$\frac{\partial C_0}{\partial w^{(L)}} = \frac{\partial z^{(L)}}{\partial w^{(L)}} \frac{\partial a^{(L)}}{\partial z^{(L)}} \frac{\partial C_0^{(L)}}{\partial a^{(L)}}$$

$$\frac{\partial C_0}{\partial b^{(L)}} = \frac{\partial z^{(L)}}{\partial b^{(L)}} \frac{\partial a^{(L)}}{\partial z^{(L)}} \frac{\partial C_0^{(L)}}{\partial a^{(L)}}$$

Se debe calcular la derivada parcial para cada momento del cálculo del costo con respecto a

$$w^{(L)} \quad b^{(L)}$$

05. Back Propagation

Conceptualización

Encontrando las derivadas parciales en función de los pesos obtenemos que:

$$\frac{\partial z^{(L)}}{\partial w^{(L)}} = \partial(b^{(L)} + w^{(L)} a^{(L-1)}) = a^{(L-1)}$$

$$\frac{\partial a^{(L)}}{\partial z^{(L)}} = \partial(f(z^{(L)}) = f'(z^{(L)})$$

$$\frac{\partial C_0^{(L)}}{\partial a^{(L)}} = \partial((a^{(L)} - y)^2) = 2(a^{(L)} - y)$$

CENIC® Centro de Investigación Coppel

05. Back Propagation

Conceptualización

Por otro lado, para el sesgo obtenemos:

$$\frac{\partial z^{(L)}}{\partial b^{(L)}} = \partial(b^{(L)} + w^{(L)} a^{(L-1)}) = 1$$

$$\frac{\partial a^{(L)}}{\partial z^{(L)}} = \partial(f(z^{(L)}) = f'(z^{(L)})$$

$$\frac{\partial C_0^{(L)}}{\partial a^{(L)}} = \partial((a^{(L)} - y)^2) = 2(a^{(L)} - y)$$

CENIC® Centro de Investigación Coppel

05. Back Propagation

Conceptualización

Finalmente, obtenemos los siguientes resultados:

$$\frac{\partial C_0}{\partial w^{(L)}} = \frac{\partial z^{(L)}}{\partial w^{(L)}} \frac{\partial a^{(L)}}{\partial z^{(L)}} \frac{\partial C_0^{(L)}}{\partial a^{(L)}} = \boxed{a^{(L-1)} f'(z^L)\ 2(a^L - y)}$$

$$\frac{\partial C_0}{\partial b^{(L)}} = \frac{\partial z^{(L)}}{\partial b^{(L)}} \frac{\partial a^{(L)}}{\partial z^{(L)}} \frac{\partial C_0^{(L)}}{\partial a^{(L)}} = \boxed{1 \quad f'(z^L)\ 2(a^L - y)}$$

Muchas gracias

dioney.contreras@coppel.com