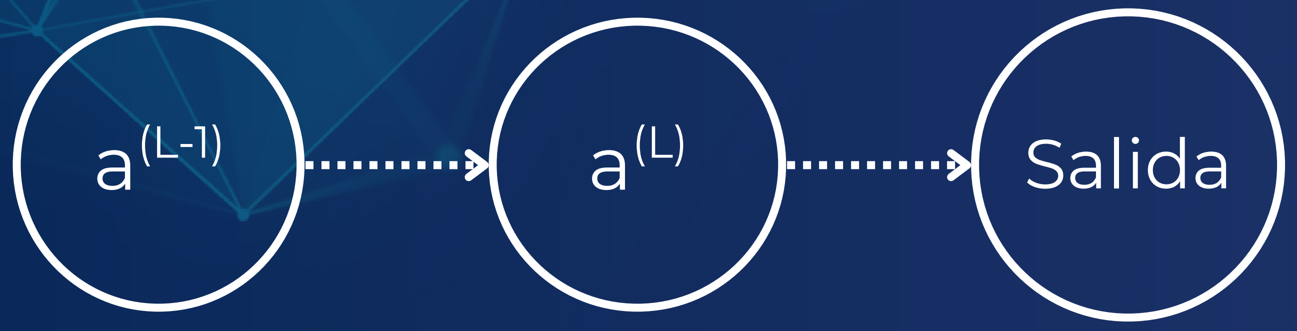
Backpropagation

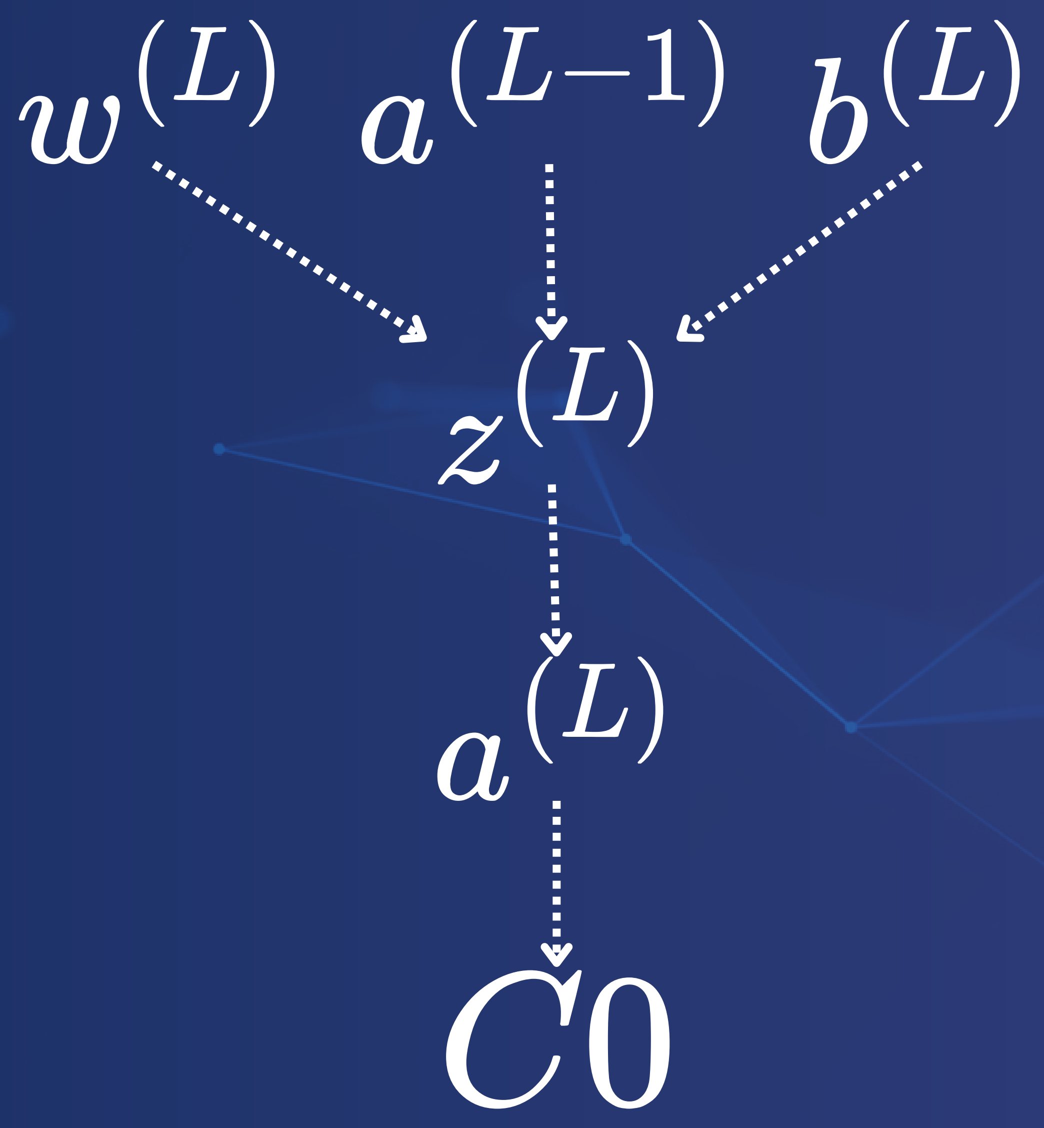
JUAN SEBASTIAN RODRIGUEZ CHIQUIZA
SERGIO ALEJANDRO REITA SERRANO
FABIAN HUMBERTO CHAPARRO AGUILERA
DAVID SANTIAGO CRUZ HERNANDEZ

Escenario inicial

Se empieza con una red neuronal lo mas sencilla posible para poder analizar la diferencia que causa en el resultado final un cambio en las variables de entrada de una neurona



cada neurona tiene una salida, $a^{(L)}$ que depende de la salida bruta por una funcion $f(z^{(L)}) = a^{(L)}$, y la salidad bruta depende del peso del peso, la salida anterior y un sesgo $z^{(L)} = w^{(L)}a^{(L-1)} + b^{(L)}$



Regla de la cadena

a partir de estos hecho se puede determinar el cambio de la salida con respecto a una de las variable de peso o sesgo, que sera igual al producto de los cambios de cada funcion dependiente de un resultado anterior, lo que se conoce como regla de la cadena

$$\frac{\delta C0}{\delta w^{(L)}} = \frac{\delta z^{(L)}}{\delta w^{(L)}} \frac{\delta a^{(L)}}{\delta z^{(L)}} \frac{\delta C0}{\delta a^{(L)}}$$

Representacion de cada derivada

De la regla de la cadena podemos ver diferentes derivadas, las cuales **representan lo que provocaria un pequeño cambio en algunas de sus partes**, y como este pequeño cambio **afectaria a las demas**, dicho de otra forma:

La regla de la cadena nos dice **como cada cambio en una parte afectaria a otra y cada una las otras partes que deriven de esta**. Por lo que cada parte nos indica la relación entre estas partes:

$$\frac{\delta z^{(L)}}{\delta w^{(L)}} \quad \frac{\delta a^{(L)}}{\delta z^{(L)}} \quad \frac{\delta C0}{\delta a^{(L)}}$$

Un cambio en **w(L)** provocaria un cambio en **z(L)**
Un cambio en **z(L)** provocaria un cambio en **a(L)**
Un cambio en **a(L)** provocaria un cambio en **C0**

Representacion de cada derivada

$$\frac{\delta z^{(L)}}{\delta w^{(L)}} = a^{(L-1)}$$

Representa un Ratio o Razón , que indica como afecta **un cambio sobre w(L) a z(L)**, lo cual es simplemente **a(L-1) osea el output de la anterior neurona en si** pues **z(l)** es la expresion para todo lo que compone el **output** de la anterior neurona, **cambiar el peso cambiaria entonces el output de la anterior neurona que es el input de mi neurona actual.**

$$\frac{\delta a^{(L)}}{\delta z^{(L)}} = \sigma'\left(\delta z^{(L)}\right)$$

Indica como afecta **un cambio sobre z(L) en a(L)**, lo cual es la **derivada de la funcion de tal neurona (o) que tiene en cuenta a z(l)**, es decir **el cambio del output de la anterior neurona (input actual) cambia el resultado de la funcion de la neurona actual.**

$$\frac{\delta C0}{\delta a^{(L)}} = 2\left(\delta a^{(L)} - y\right)$$

Indica como afecta **un cambio sobre a(L) en el costo C0**, lo cual es **2 veces la diferencia entre ese cambio en a(L) y el output esperado,** ya que mi neurona actual es la ultima de la red(o aquella que estoy evaluando), representa el como **el cambio del output final de la red(o de la red hasta esta neurona) cambia el coste de entranamiento.**

Representacion de cada derivada

Al final la representacion de la regla de la cadena explicitamente nos dice que **un cambio del peso w(L) provocaria un cambio del Coste C0 que provacaria a su vez todos los demas cambios**:

$$\frac{\delta C0}{\delta w^{(L)}} = a^{(L-1)}\sigma'\left(z^{(L)}\right)2\left(a^{(L)} - y\right)$$

Todo esto para el cambio de un solo peso para una sola neurona osea para un solo input de esta neurona, por lo que el coste total de entramiento es:

$$\frac{\delta C}{\delta w^{(L)}} = \frac{1}{n}\sum_{k=0}^{n-1}\frac{\delta Ck}{\delta w^{(L)}}$$

El promedio de todas las iteraciones o variaciones del entrenamiento que ocurren al cambiar el peso

Función de Costo (Entrenamiento)

La función de costo mide qué tan lejos está la predicción de la red del valor real.

Para un solo ejemplo de entrenamiento:

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

donde:

$a^{(L)}$: Activación de salida de la red (predicción).

y : Valor esperado (etiqueta real).

Para todo el conjunto de entrenamiento (promedio sobre todos los ejemplos):

$$C = \frac{1}{n}\sum_{i=1}^{n}\left(a^{(L)} - y\right)^2$$

donde:

n : número total de ejemplos de entrenamiento.

Sensibilidad del Costo/Peso

Se desea saber cómo cambiar un **Peso** afecta al costo. Este hecho brinda la dirección de descenso para optimizar los parámetros.

Tasa de cambio del peso : $$\frac{\partial C}{\partial w^{(L)}}$$

Usando la **regla de la cadena** : $$\frac{\partial C}{\partial w^{(L)}} = \frac{\partial C}{\partial a^{(L)}} \cdot \frac{\partial a^{(L)}}{\partial z^{(L)}} \cdot \frac{\partial z^{(L)}}{\partial w^{(L)}}$$

Se calculan las **derivadas parciales** :

$$\frac{\partial C}{\partial a^{(L)}} = \left[\left(a^{(L)} - y\right)^2\right]' = 2\left(a^{(L)} - y\right)$$

$$\frac{\partial a^{(L)}}{\partial z^{(L)}} = \sigma\prime\left(z^{(L)}\right)$$

$$\frac{\partial z^{(L)}}{\partial w^{(L)}} = \left[w^{(L)} \cdot a^{(L-1)} + b^{(L)}\right]' = a^{(L-1)}$$

Sensibilidad del Costo/Peso

Fórmula Final para **Sensibilidad** respecto a **Peso**:

$$\frac{\partial C}{\partial w^{(L)}} = 2(a^{(L)} - y) \cdot \sigma\prime(z^{(L)}) \cdot a^{(L-1)}$$

Sensibilidad del Costo/Bias

Se desea saber cómo cambiar el **Bias** afecta al costo.

Tasa de cambio del bias: $$\frac{\partial C}{\partial b^{(L)}}$$

Usando la **regla de la cadena** : $$\frac{\partial C}{\partial b^{(L)}} = \frac{\partial C}{\partial a^{(L)}} \cdot \frac{\partial a^{(L)}}{\partial z^{(L)}} \cdot \frac{\partial z^{(L)}}{\partial b^{(L)}}$$

Se calculan las **derivadas parciales** de cada componente:

$$\frac{\partial C}{\partial a^{(L)}} = \left[\left(a^{(L)} - y\right)^2\right]' = 2\left(a^{(L)} - y\right)$$

$$\frac{\partial a^{(L)}}{\partial z^{(L)}} = \sigma\prime\left(z^{(L)}\right)$$

$$\frac{\partial z^{(L)}}{\partial b^{(L)}} = \left[w^{(L)} \cdot a^{(L-1)} + b^{(L)}\right]' = 1$$

Sensibilidad del Costo/Bias

Fórmula Final para **Sensibilidad** respecto a **Bias**:

$$\frac{\partial C}{\partial w^{(L)}} = 2(a^{(L)} - y) \cdot \sigma\prime(z^{(L)}) \cdot 1$$

Backward Step: Sensibilidad a la Capa Anterior

Ahora se desea observar como afecta la activación de la capa **anterior** al costo

$$\frac{\partial C}{\partial a^{(L-1)}} = \frac{\partial C}{\partial z^{(L)}} \cdot \frac{\partial z^{(L)}}{\partial a^{(L-1)}}$$

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

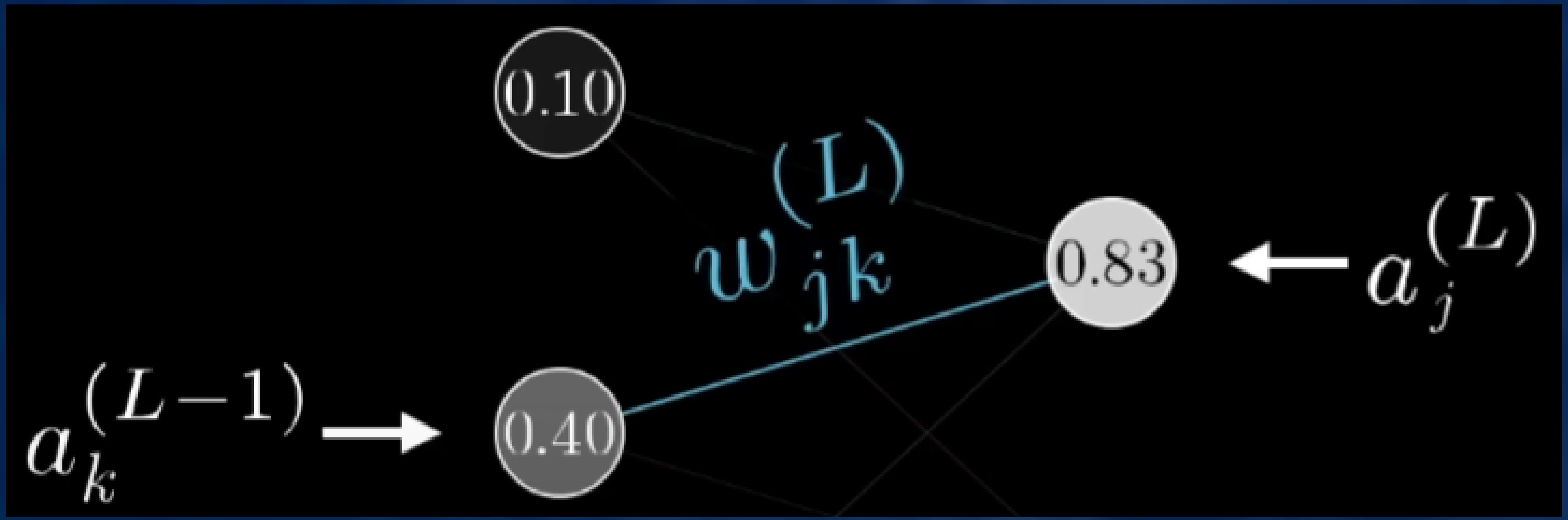
$$\frac{\partial C}{\partial z^{(L)}} = \frac{\partial C}{\partial a^{(L)}} \cdot \frac{\partial a^{(L)}}{\partial z^{(L)}} = 2(a^{(L)} - y) \cdot \sigma\prime(z^{(L)})$$

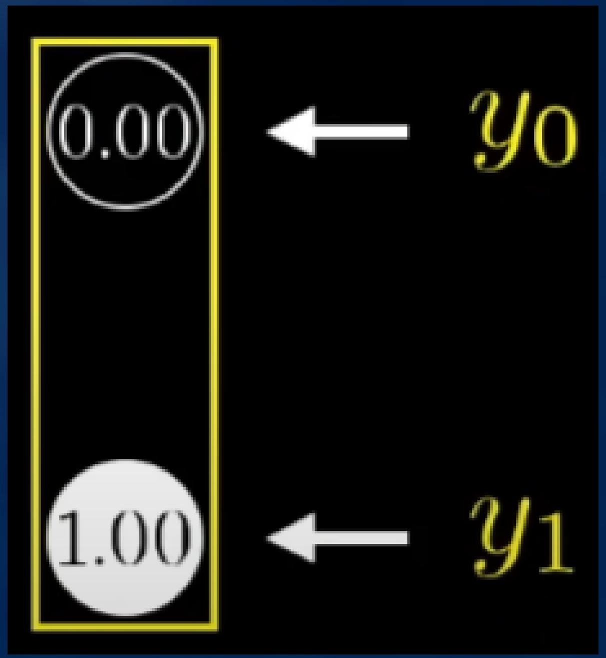
Por lo tanto:

$$\frac{\partial C}{\partial a^{(L-1)}} = 2(a^{(L)} - y) \cdot \sigma\prime(z^{(L)}) \cdot w^{(L)}$$

Capas con neuronas adicionales

Se dejan indicadas las posiciones para cada neurona, usando índices como
0, 1,.., k, k-1, ..., kn





El principal cambio viene de tener que tener en cuenta las posiciones de las neuronas

Capas con **neuronas adicionales**

Formula para **calcular costos**:

$$C_0 = \sum_{j=0}^{n_{L-1}} \left(a_j^{(L)} - y_j\right)^2$$

Es planteada de la misma manera que lo mostrado anteriormente, con el cambio de enfoque en que hay que ver la posición de la neurona, pero tiene misma implementación

Capas con
neuronas adicionales

Formula para **calcular sensibilidad** para un **peso especifico**:

$$\frac{\partial C_0}{\partial w_{jk}^{(L)}} = \frac{\partial z_j^{(L)}}{\partial w_{jk}^{(L)}} \cdot \frac{\partial a_j^{(L)}}{\partial z_j^{(L)}} \cdot \frac{C_0}{\partial a_j^{(L)}}$$

También pasa por la misma situación de la demostración anterior, solo teniendo en cuenta sus posiciones como una matriz

Capas con
neuronas adicionales

Formula para **calcular sensibilidad** respecto a una **capa anterior**:

$$\frac{\partial C_0}{\partial a_k^{(L-1)}} = \sum_{j=0}^{n_{L-1}} \frac{\partial z_j^{(L)}}{\partial a_k^{(L-1)}} \cdot \frac{\partial a_j^{(L)}}{\partial z_j^{(L)}} \cdot \frac{C_0}{\partial a_j^{(L)}}$$

Aquí, se presenta la necesidad de también sumar los pesos de las neuronas de la capa actual, ya que su valor es dependiente de los valores de la capa anterior

Gracias

Computación Visual

Grupo 7

Universidad Nacional de Colombia