



Comisión Nacional
de Energía Atómica



Trabajo Práctico N° 2: Introducción a las Redes Neuronales

Aprendizaje Profundo y Redes Neuronales Artificiales

20 de septiembre de 2020

Alumnos :

Tomás LIENDRO tomas.liendro@ib.edu.ar

Docentes :

Ariel CURIALE
Germán MATO
Lucca DELLAZOPPA

San Carlos de Bariloche, Argentina

Índice

1. Ejercicio 1

En este trabajo se abordaron los conceptos de *Computational Graphs* y *Back Propagation* para entrenar una red neuronal densa. Un **grafo computacional** es una manera de representar una función matemática en el lenguaje de la teoría de grafos. En el grafo, cada nodo representa un valor de una variable o una función que combina valores. Los valores que ingresan o salen de los nodos se denominan **Tensores**.

Utilizando el concepto de **grafos computacionales** se busca implementar las funciones **Sigmoide**, **Tanh**, **ELU** y **Leaky Relu**, tomando como entrada: $x = [2,8, -1,8]$, $W = [1,45, -0,35]$, $b = -4$. En los siguientes grafos, los valores en color negro indican los resultados del proceso hacia adelante (*forward*) mientras que en rojo se colocan los resultados del *Backpropagation* recordando que el valor del gradiente a la salida del grafo es 1.

1.1. Sigmoide

$$f(x) = \frac{1}{1 + e^{-x}}; \quad f'(x) = \sigma(x)(1 - \sigma(x)) \quad (1.1)$$

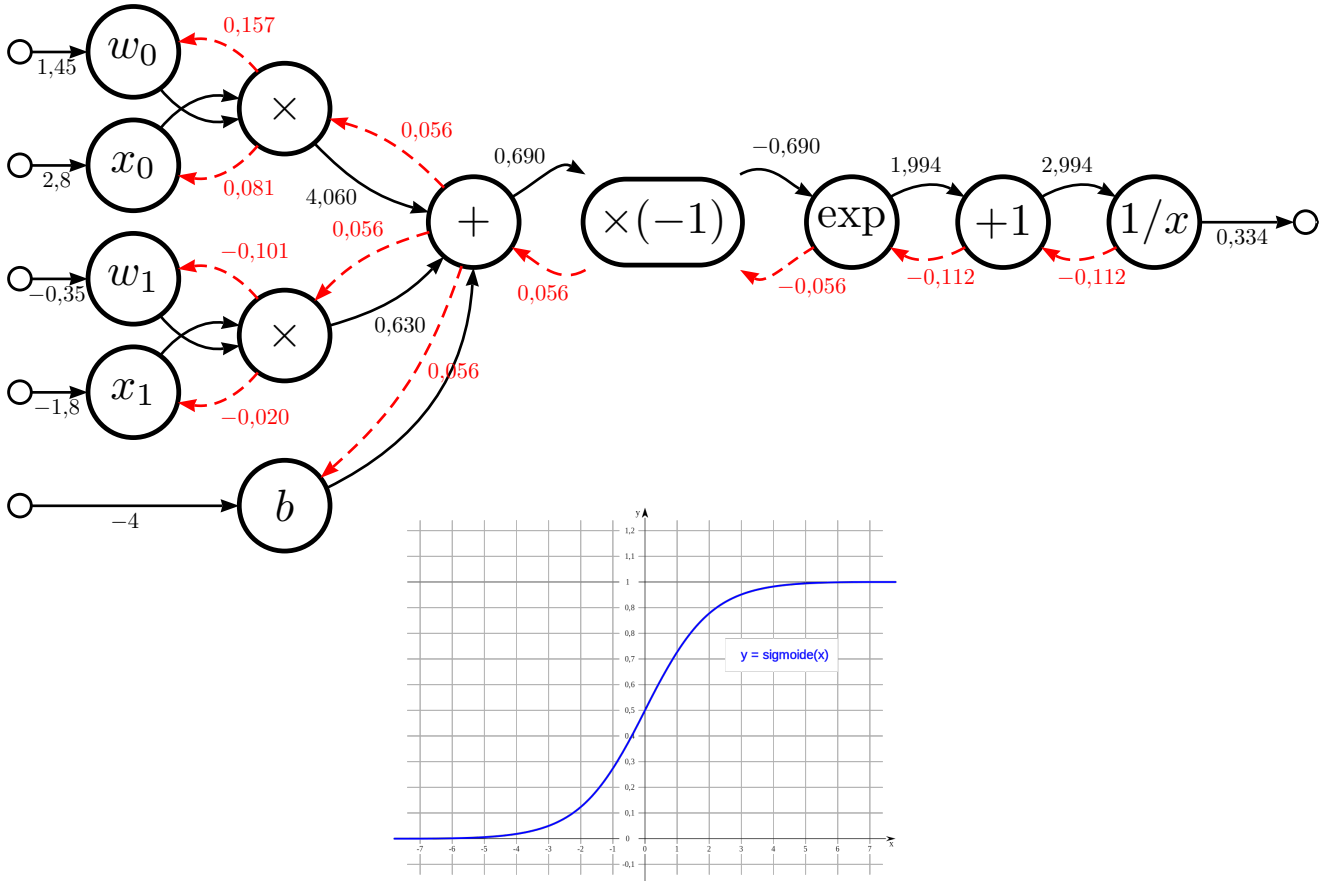


Figura 1: Función Sigmoide ($\sigma(x)$)

1.2. Tanh

$$f(x) = \tanh(x) = \frac{e^{-x} - 1}{e^{-x} + 1}; \quad f'(x) = 1 - \tanh^2(x) \quad (1.2)$$

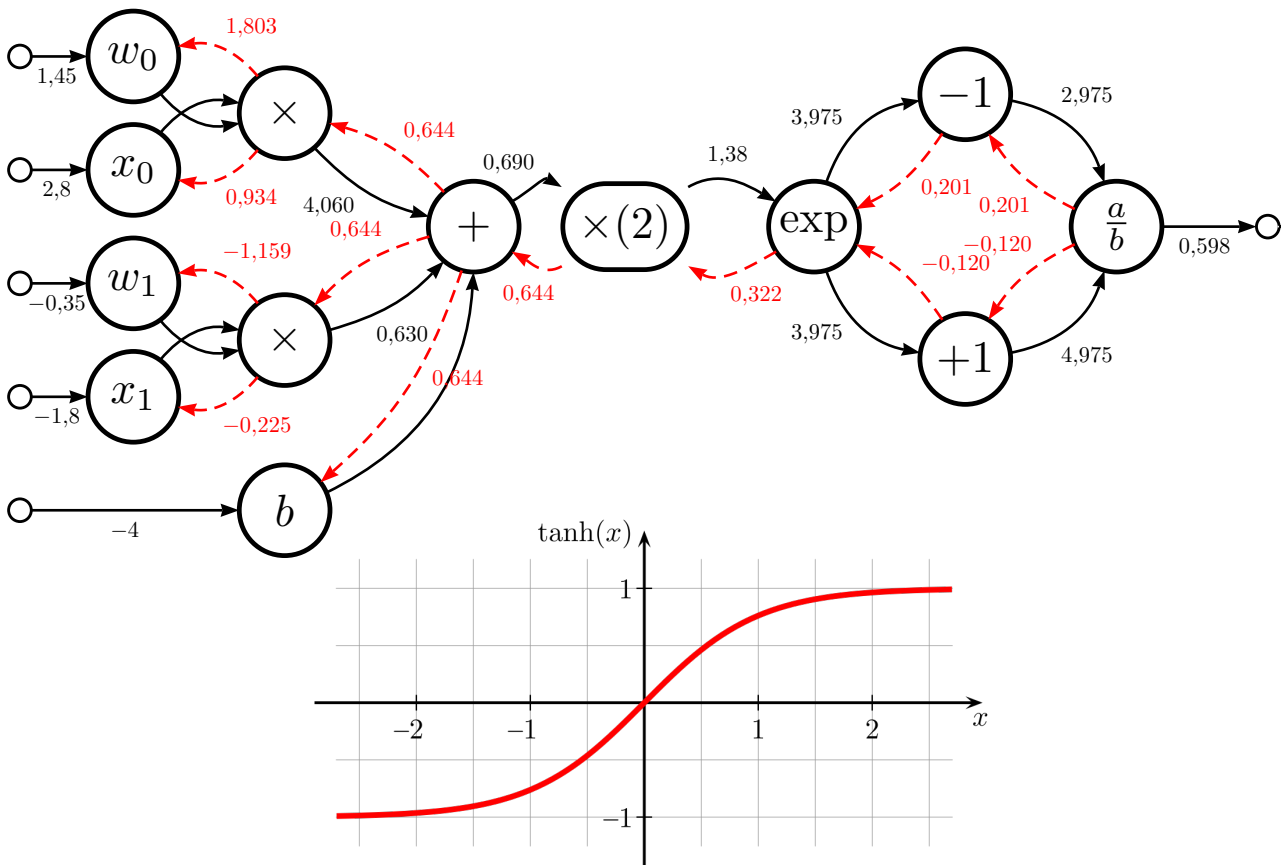
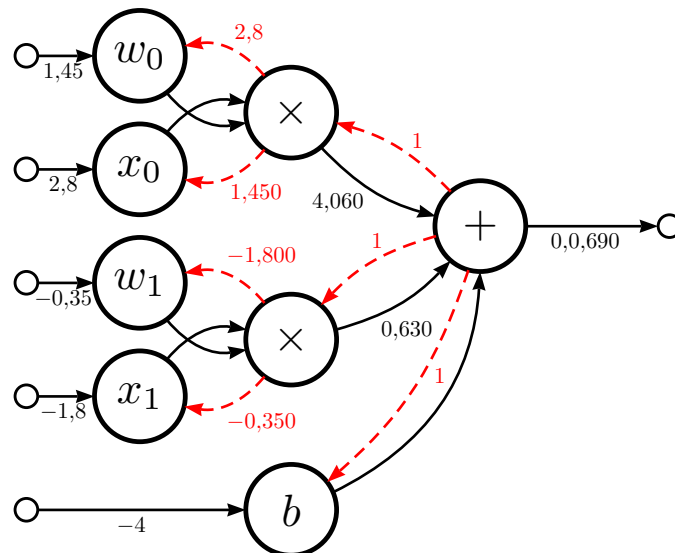


Figura 2: Función Tangente Hiperbólica ($\tanh(x)$)

1.3. ELU

$$f(a, b) = \begin{cases} x & x \geq 0 \\ \alpha(\exp(x) - 1) & x < 0 \end{cases}$$



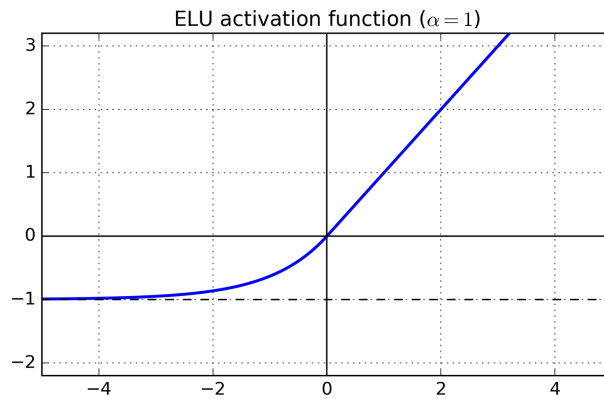
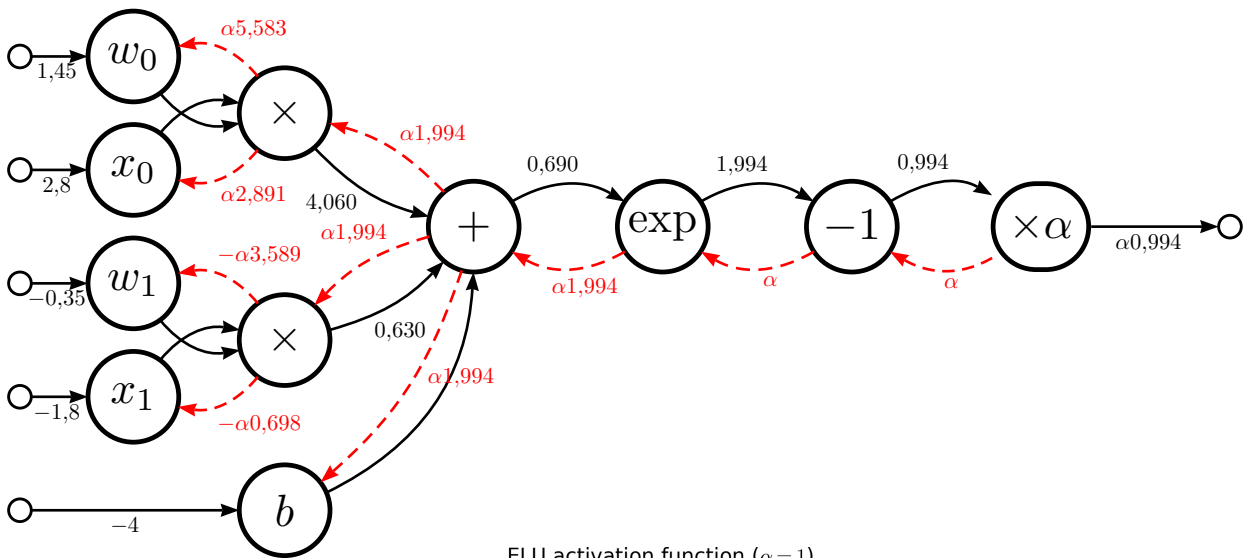
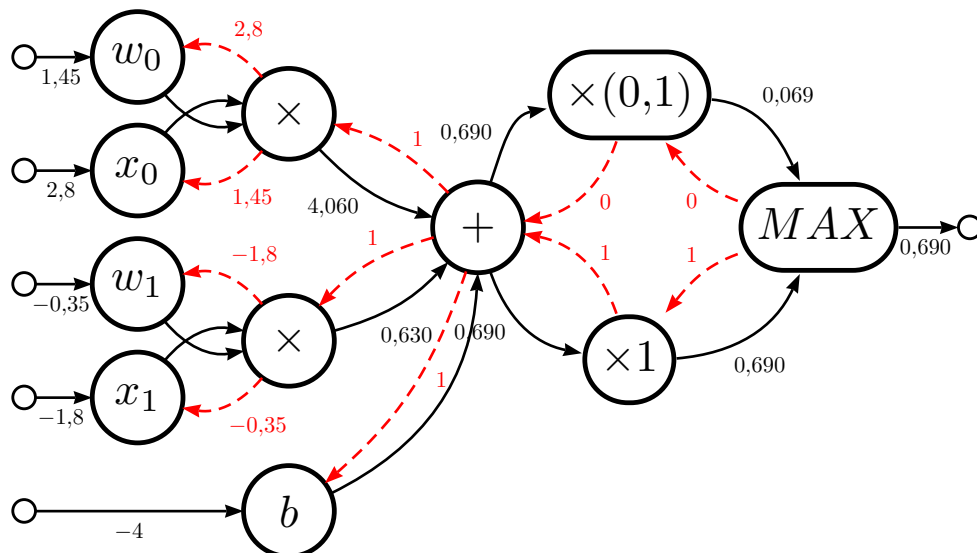


Figura 3: Función de activación ELU

1.4. Leaky Relu

$$f(x) = \text{LeakyRelu}(x) = \max(0, 1x, x) \quad (1.3)$$



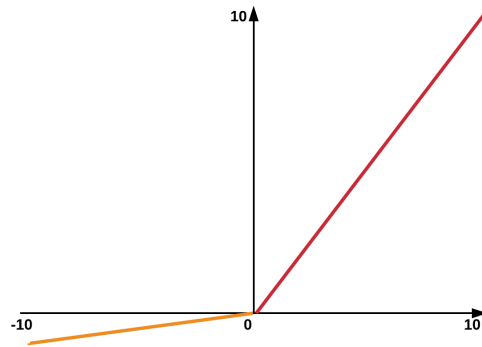


Figura 4: Función Leaky Relu

La elección de la función de activación para una capa de una red neuronal depende de los valores de salida que se requieren y de las propiedades de la función de activación utilizada. Por ejemplo, si se trabajan con probabilidades, podría pensarse en usar la función de activación **Sigmoide** dado que esta función tiene la salida acotada entre 0 y 1, si se trabaja con valores positivos se pensaría en usar la función **Relu** que anula los valores negativos y para valores generales, se pensaría en usar la función de activación **Lineal**.

En esquemas con más de una capa, uno preferiría usar la función **Leaky Relu** en lugar de la **Relu**, la cual permite ser más permisivo al no anular completamente ciertas neuronas y por el mismo motivo, se prefiere usar la función **tanh** en lugar de la **Sigmoide**.

Referencias

- [1] Julien Jacques, Linear Regression in High Dimension and/or for Correlated Inputs. Article in EAS Publications Series 5 January 2015. <https://www.researchgate.net/publication/276249836>
- [2] Ariel Curiale, German Matos, *Notas de la cátedra: Aprendizaje Profundo y Redes Neuronales Artificiales*. <https://classroom.google.com/u/2/c/MTQxNDcxMTM5NjEx>
- [3] Fei-Fei Li, *Convolutional Neural Networks for Visual Recognition*. <http://cs231n.stanford.edu/2017/>
- [4] Códigos del TP1: https://github.com/TomasLiendro/Deep_Learning/blob/master/TP1_Introduction%20to%20ML/