* 1. Describe el algoritmo de retropropagación implementado:

El algoritmo de retropropagación implementado se trata de una serie de pasos que emula los pasos del implementado en teoría. Este consiste de la iteración multiple de tres fases diferentes: propagación hacia delante, retropropagación del error y ajuste de pesos.

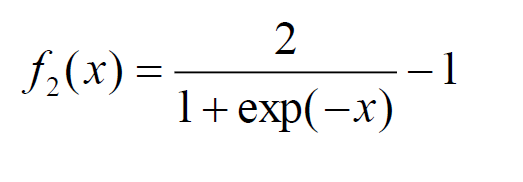
Propagación hacia delante

El algoritmo comienza inicializando la red, poniendo el valor de entrada de las diferentes neuronas y sesgos a valor 0, aunque en el sesgo la salida siempre valdrá 1 independientemente de la entrada. A continuación mientras la condición de parada no se cumpla se ejecutan iterativamente los siguientes pasos que pasare a explicar, correspondiéndose este conjunto de pasos a una época. La condición de parada se trata de que no haya habido ninguna actualización de pesos y que no se haya llegado al límite de épocas establecidas al inicializar la red multicapa, 100 por defecto si no se especifica parámetro.

Se itera sobre el conjunto de pares X e Y de entrenamiento a partir de los cuales primeramente se incializa con los valores de X la entrada de las neuronas de la capa de entrada, cada neurona con uno de los valores, al bias por otra parte no se le modifica la entrada.

Posteriormente se calcula la respuesta de la última capa oculta realizando el cálculo de la respuesta de las capa de entrada y de las diferentes capas ocultas, a partir de iterar desde la primera capa hasta la anterior a la de salida, disparando, inicializando y propagando para cada una de ellas.

En cada capa, cuando se dispara, se obtiene el valor de salida en función del valor de entrada de cada neurona haciendo uso de la función de activación de la sigmoide bipolar para este caso, la cual tiene la forma que se muestra más adelante. Este valor de salida se pasa como valor a cada una de las conexiones.



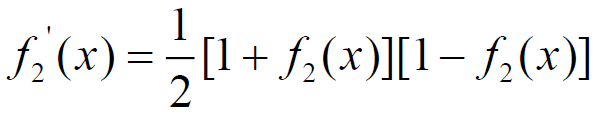
Una vez se ha disparado la capa se inicializa, estableciendo para las neuronas el valor de entrada a 0.

Posteriormente se propaga en la capa, a partir de lo cual el valor de entrada de la neurona se recalcula a partir de multiplicar el valor de la conexión por el peso de la conexión, haciendo el sumatorio para su conjunto de conexiones.

Una vez realizados los pasos anteriores iterando sobre las diferentes capas se obtiene la salida de la capa anterior a la capa de salida, a partir de lo cual se actualiza el valor de la capa de las conexiones de la capa de salida. En este punto, para la capa de salida solo se dispara, en vez de inicializar y propagar ya que para el ajuste de sus pesos hace falta y\_in, es decir su salida habiendo aplicado la función de activación.

Retropropagación del error:

Para cada neurona de salida se obtiene el valor de salida correcto a partir del vector con el conjunto de valores de Y, y a este valor se le resta el valor de salida obtenido para la neurona. Este resultado se multiplica por la derivada de la función de transferencia de la sigmoide bipolar, teniendo esta de entrada tiene el sumatorio de los pesos multiplicado por el valor de los enlaces, es decir f’(y\_ink). Por otra parte esta función se descompone de la siguiente forma, siendo x=y\_ink.



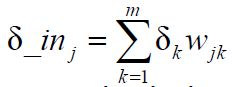
Debido a que f(x), es decir f(y\_ink) es yk, el valor de la derivada de la función de transferencia de la sigmoide con y\_ink de entrada es 1/2\*((1+yk)\*(1-yk)).

Debido a que en este caso la red neuronal se trata de un perceptrón multicapa, cada una de estas neuronas están conectadas con el conjunto de neuronas que componen la capa anterior. Por ello para cada neurona de salida se itera sobre el conjunto de neuronas de la capa anterior con el fin de calcular la corrección de peso. Esto se realiza a partir de multiplicar el error calculado anterior por el coeficiente de aprendizaje y por el valor de salida de la neurona por la que se itera neurona de la capa oculta. Esta fórmula de corrección de peso para cada conexión con neuronas de la capa anterior tiene la siguiente forma. Lo anterior se aplica indistintamente a neuronas y sesgos debido a que el sesgo como se explico al principio su salida es siempre 1, por lo que su fórmula se cumple. Esta también se muestra a continuación.

Estos cambios de pesos producidos por cada una de las neuronas de la capa de salida para cada conexión con las neuronas de la capa anterior se guardan en una matriz de tamaño(número neuronas capa de salida, numero neuronas capa anterior), esta matriz se llama “deltas\_w”. Por otra parte el error para cada neurona se guarda en otra matriz, de nombre “sigmas\_k”. Y estos cambios de peso en una matriz aun mayor que guardará el conjunto de cambios de pesos para todas las conexiones de todas las capas, llamada “cambios pesos”

Una vez realizados los pasos anteriores se va iterando capa por capa, desde la capa anterior a la de salida hasta la capa inicial, incluida. Para cada una de estas capas se recorren las neuronas una a una las cuales usando el error de cada neurona de la capa posterior el cual está guardado en la matriz mencionada calcula alfa\_inj a partir de multiplicar estas por los pesos de sus conexiones. Este cálculo se muestra en la siguiente fórmula.



Este valor calculado se multiplica por la derivada de la función de activación de la sigmoide binaria, explicada anteriormente, para calcular el error de cada una de las neuronas. Esta se descompone de la misma forma a como se explicó antes, pero en vez de para y\_ink a z\_inj. Esta formula se muestra adelante.



Con este error para cada neurona se itera para el conjunto de neuronas de la capa anterior, ya que cada neurona de la capa anterior esta conectado con cada neurona de la capa posterior que se itera, al ser un perceptrón multicapa. En estas iteraciones en cada neurona se multiplica su error por el valor de salida de la neurona de la capa anterior que se itera además de por el coeficiente de aprendizaje y se obtienen las correcciones de peso. Esta fórmula es la siguiente. El bias se trata de la misma forma que cualquier otra neurona ya que como se explico antes su salida es 1 independientemente de su valor de entrada aunque su fórmula se procede a mostrar también.

Ajuste de pesos:

Con la matriz que contiene los cambios de pesos de cada de cada neurona de cada capa, para el conjunto de conexiones con las neuronas de la capa anterior se itera y actualizan los pesos de estas conexiones.

Finalmente se comprueba la condición de parada y se inicializan las entradas de la última capa a 0