



UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MÉXICO

CENTRO UNIVERSITARIO UAEM VALLE DE MÉXICO

**Aprendizaje incremental para la tarea de reconocimiento
de dígitos con Redes Neuronales Artificiales**

PROTOCOLO DE TESIS

Que para obtener el Título de

INGENIERO EN SISTEMAS Y COMUNICACIONES

P r e s e n t a

**C. Fragoso García Sandra
C. González Hernández Luis Ángel**

Asesor: Dr. Víctor Manuel Landassuri Moreno

Atizapán de Zaragoza, Edo. de Méx. octubre 2022



Contents

1	Introducción	3
2	Planeamiento	4
3	Objetivos	5
3.1	Objetivos Particulares	5
4	Justificación	6
5	Delimitación	7
6	Consecuencias	7
7	Marco Teórico	8
7.1	Redes Neuronales Artificiales	8
7.1.1	Función de Activación	10
7.2	Redes Neuronales de Perceptrón Multicapa	12
7.3	Algoritmo Backpropagation	13
7.4	Redes Neuronales Convolucionales	14
7.5	Aprendizaje	16
7.5.1	Aprendizaje en humanos	16
7.5.2	Aprendizaje Humano	16
7.5.3	Aprendizaje con Compresión	16
7.5.4	Aprendizaje Activo	16
7.5.5	Aprendizaje Incremental	17
7.5.6	Algoritmos de Aprendizaje Incremental	17
8	Hipotesis	18
9	Metodología	18
10	Cronograma de Actividades	18
11	Organización del Capitulado	19

List of Figures

1	Red Neuronal Artificial Básica	8
2	Puertas Lógicas [12]	9
3	Puerta Lógica Xor [12]	10
4	Función Escalonada	10
5	Función Sigmoide	11
6	Función ReLU	11
7	Función Softmax	12
8	Perceptron Multicapa	13
9	Arquitectura de una RNC [14]	15
10	Métodos Pooling	15

Abstract

El aprendizaje incremental es una evolución para el área de la inteligencia artificial por el cual los algoritmos de Machine y Deep Learning pueden ser optimizados como es presentado en [2], en específico en las redes neuronales se utiliza esta metodología para que modelo de predicción sean más precisos, para esto se usan métodos como lo que es el backpropagation, el cual conbinado junto con las redes neuronales de perceptrón multiple ayudan a que los modelos sean muy específicos. En la presente investigación se maneja una optimización en la predicción con el uso del perceptron multiple.

1 Introducción

La inteligencia artificial (IA) es una área del conocimiento que se enfoca en poder hacer máquinas que tengan un comportamiento y razonamiento humano, para que en un momento, se pueda interactuar con ellas sin darse cuenta que se está interactuando con una máquina. Así mismo, también es posible pensar que mucho del desarrollo en el área de inteligencia artificial, es el poder tener mejores herramientas que ayuden a las actividades diarias.

En este sentido, un área de la IA es el llamado Aprendizaje Máquina, donde se estudian algoritmos que permiten aprender de forma automática una tarea. Así, una de las técnicas más conocidas en la actualidad, dentro del área de IA son las Redes Neuronales Artificiales (RNAs), siendo técnicas que realizan procesos matemáticos para poder aprenderse tareas a resolver. Algunas áreas en las que son útiles las RNAs son en el aprendizaje de tareas no lineales, como la predicción de la capacidad de la red 5G, basada en el tráfico diario de este [19] o clasificación, por ejemplo la clasificación de metales y rocas por medio de RNAs y lógica difusa [18].

Las RNAs están formadas por neuronas artificiales que simulan a las biológicas. Así los procesos químicos que suceden en el cerebro, se simulan computacionalmente a través de señales que viajen a través de las neuronas artificiales, de aquí en adelante simplemente se referirá a ellas como "neuronas". Las neuronas en una RNA cuentan con una estructura distribuida en paralelo, presentando una buena habilidad de aprendizaje [11].

Dentro del aprendizaje máquina, cuando una técnica, por ejemplo RNA, se enfoca en aprender una tarea, se le conoce como *algoritmo lineal sin memoria* siendo uno de los métodos más empleados desde el inicio de las RNAs [8]. Sin embargo, si es necesario incorporar nueva información del problema, es necesario volver a entrenar todo el modelo, considerando toda la información existente, esto es la anterior y la nueva que acaba de llegar, es ahí donde nace el concepto de Aprendizaje Incremental, siendo un área enfocada en poder incorporar información del problema en cuestión, sin tener que volver a re-entrenar todo el modelo.

Derivado del aprendizaje incremental se desprende el concepto de memoria dentro de la IA, analizando como un algoritmo de aprendizaje máquina puede olvidar la información que se usó en un entrenamiento previo, al entrenar con información más reciente. Si se hace la analogía con los humanos la memoria, es un factor importante para estudiar considerando la perdida de información aprendida, así este es un problema biológico, el cual tanto afecta a los humanos como a las máquinas. Por ello, se han elaborado distintos experimentos para

poder combatir esta problemática. Uno de estos es el caso de [2], donde se propone el manejo de RNAs con pesos dobles, donde la primer capa de pesos esta enfocada a comportarse como memoria a corto plazo, y la segunda como memoria a largo plazo. Los experimentos mostrados en [2] permiten notar un mejora en tareas de aprendizaje incremental, teniendo menos perdida de información en comparación de implementaciones anteriores como el algoritmo Learn++ [10, 5].

Así, el presente trabajo de investigación esta enfocado en poder explorar nuevas configuraciones de pesos duplicados para poder extender el trabajo previamente presentado en [2].

2 Planeamiento

Las Redes Neuronales Artificiales tienen la habilidad de poderse aprender una tarea, y poder hacer o resolver tareas de predicción o clasificación básicamente. En este sentido, con algoritmos de aprendizaje como el Backpropagation, permite ajustar los pesos de una RNA para que esta pueda empezar a resolver una tarea particular. No obstante, muchas de las tareas que se resuelven en la vida diaria, van generando mas información con el tiempo, por ejemplo, el comportamiento de un una serie financiera, o bien la predicción del clima en una determinada región. Así, se tiene el aprendizaje incremental, siendo un método poco explorado, enfocado en poder aprender nueva información del problema, sin tener que volver a entrenar todo el modelo con la información anterior, y la nueva que acaba de llegar, esto es, en los modelos actuales de aprendizaje máquina, si se usa un conjunto de datos para entrenar un modelo en especifico, dicho modelo es funcional para dicho conjunto de datos y la información que ello representa. Sin embargo, si es necesario incorporar nueva información al modelo, es necesario recolectar dicha información nueva, agregarla a la que se tenia anteriormente, y volver a entrenar todo el modelo para que éste pueda incorporar la nueva información.

De esta forma, una red ya entrenada con un primer conjunto de datos d_1 se desea entrenar con un nuevo conjunto de datos del mismo problema d_2 , al entrenar la red con d_2 se perderá en conocimiento aprendido por d_1 . Si no se desea utilizar un modelo de aprendizaje incremental, se tendría que juntar el conjunto d_1 y d_2 en un solo conjunto y volver a entrenar la RAN para así poder incorporar el nuevo conocimiento (d_2) a la RNA. Si se desea utilizar el método de aprendizaje incremental, se puede entrenar en un primer moemnto a la red con el conjunto d_1 , posteriormente con d_2 teniendo poca perdida de información de d_1 . Si más adelante llega mas información del problema (d_3) que se desee incorporar a la base de conocimientos de la RNA, entonces solo habrá que entrenar la RNA con d_3 usando el modelo incremental para tener una pérdida mínima de información de d_1 y d_2 .

De esta forma, el presente trabajo tomará como base la investigación de [2], en donde utiliza una configuración de pesos dobles (a una RNA se duplican todos sus pesos), donde a una capa de pesos duplicados se enlaza con una tasa de aprendizaje alta, para simular un rápido aprendizaje, y por ende simular lo que sería memoria a corto plazo. En su contraparte, la segunda capa de pesos duplicados, se enlaza con una tasa de aprendizaje baja para aprender lentamente un problema, simulando la memoria a largo plazo. Es decir, al

momento de aprender una tarea nueva, una capa de pesos aprenderá muy rápido la nueva tarea (tasa alta de aprendizaje) y por ende olvidará más rápidamente los datos ya aprendidos anteriormente, por otro lado, la segunda capa de pesos duplicados, aprenderá muy lentamente los nuevos datos que llegan, y por ende olvidando poco la información que anteriormente se aprendió. Considerando ello y también que la RNA trabaja en conjunto con ambas capas de pesos duplicados, se pondera la salida de la RNA para que pueda contemplar la información nueva que se acaba de agregar a la RNA, en ambas capas, así como la información que se acaba de olvidar de ambas capas de pesos.

El problema principal del aprendizaje incremental mostrado en [2], es que entre más conjuntos de datos nuevos que lleguen, mas se olvidaran los primeros conjuntos que se aprendieron, lo cual no es tan útil si se contempla que en el futuro de una RNA podrán existir 10 o 20 etapas de entrenamiento incremental con nuevos conjuntos de datos que se vayan recolectando.

Por ello es indispensable poder explorar nuevas configuraciones de RNAs que permitan mejorar los métodos actuales para permitir una menor cantidad de olvido conforme llegue nueva información al modelo. Donde al igual que en trabajos anteriores, el presente trabajo se basará en conceptos de memoria a corto y largo plazo, y en lugar de hacer una copia de los pesos actuales y tener dos tasas de aprendizaje, una rápida para simular la memoria a corto plazo, y otra tasa de aprendizaje para simular la memoria a largo plazo, se explorará por hacer mas copias de los pesos, teniendo más tasas de aprendizaje que operen en cada una de dichas copias.

3 Objetivos

Diseñar una red neuronal artificial para aprendizaje incremental basada en el principio de la memoria a corto y largo plazo, buscando usar más de dos capas de pesos duplicados para el reconocimiento de dígitos, y con una menor pérdida de información de trabajos previos.

3.1 Objetivos Particulares

1. Implementar el algoritmo mostrado en [2] para el reconocimiento de dígitos con aprendizaje a corto y largo plazo con los parámetros que ahí se indican.
2. Obtener el conjunto de datos de Optical Digits, limpiar los datos y prepararlos según lo indicado con [2].
3. Separar el conjunto de entrenamiento y de prueba de acuerdo a lo que se explica en el artículo de Bullinaria y probar el primer código implementado en miras de comprobar los resultados previamente mostrados en [2].
4. Tomar como base el algoritmo implementado, y extenderlo para permitir mas de dos pesos duplicados, aplicando el conjunto de datos previamente mostrado.
5. Comparar ambas implementaciones en busca de una reducción significativa de las tasas de aprendizaje con respecto a trabajos previos en la literatura.

4 Justificación

Las redes neuronales permiten el aprendizaje automático y la resolución de distintos problemas, pero como se comentó anteriormente, las técnicas de aprendizaje máquina, tienen una deficiencia que es al momento de aumentar los nuevos bloques de datos que llegan para aprender, se obtiene un deterioro en el rendimiento de aprendizaje de información y olvido de la información anterior [2]. (REFIVEN COMO VOY PONEINDO LA REDACCIÓN DE LO QNTERIOR, Y CO NELLO TRATEN DE CAMBIAR LO QUE VIENE PARA QUE LE MOJOREN EL ESTILO, DE AQUÍ EN ADELANTE SOLO LES HARE OBSERVACIONES EN ESPERA QUE USTEDES MEJOREN EN GENERAR LA REDACCIÓN)

Los resultados que se han obtenido no funcionan a la perfección, la memoria a corto plazo olvida poco pero va olvidando, y lo ideal sería que no olvidara. Biológicamente los humanos pueden aprender nuevas tareas, o información nueva de un problema, y no olvida de forma significativa lo que anteriormente aprendió, no obstante eso no pasa actualmente con las RNA y en general con cualquier algoritmo de aprendizaje máquina. En otro sentido, los humanos ya tienen cierta configuración en el cerebro que les permite aprender como se hace actualmente, y se puede afirmar que por el momento no hay ningún procedimiento (quirúrgico o no) que permita modificar la estructura del cerebro para aprender mas y olvidar menos.

No obstante, computacionalmente nada puede impedir que se experimente con más configuraciones y llegar al punto en donde toda la información que ingrese a un modelo (por ejemplo RNAs) se acumule, y si no hay problema de almacenamiento que se siga acumulando y que no olvide, esto podría ser bueno en diversas situaciones.

Desde el punto de vista computacional, si llega nueva información y no se ocupa aprendizaje incremental, ello implicará volver a entrenar todo el sistema con la información anterior y la actual (por ejemplo d_1 y d_2) y considerando que una de las desventajas que tienen la RNAs es que el entrenamiento es un cuello de botella, siendo este donde se lleva la mayor parte de cómputo y por consiguiente de energía. Ello implica que volver a entrenar con toda la información acumulada, gastará más energía y tiempo, que si solo se entrena con la nueva información que llega al modelo. En su contraparte, existe una gran variedad de herramientas las cuales permiten codificar una red neuronal artificial con librerías ya preexistentes, por el momento se expondrán solo 2 empresas, siendo estas las más importantes: Microsoft y Google. La primera cuenta con la plataforma de Azure que renta una maquina virtual donde se puede programar en Python. Por el contrario, Google cuenta con Google Colab que igual brinda una máquina virtual para realizar experimentos de Maching Learning, la única diferencia contra Azure es que dicha herramienta es gratuita, una similitud que tienen es que en las dos se puede programar en el mismo lenguaje.

Como se observa, ambas herramientas permiten la programación en Pyhton y esto se debe a que dicho lenguaje es una herramienta de software libre que no requiere licencia, es relativamente fácil poder depurar un código y permite acelerar más el desarrollo de aplicaciones, a diferencia de otros lenguajes más estructurados como C o Java, además tiene más librerías para el desarrollo de Maching Learning, por ejemplo, TensorFlow, Numpi, entre otras.

TensorFlow es una librería de Python que permite construir y entrenar redes neuronales para detectar patrones y razonamientos usados por los humanos, en la presente investigación se usará dado a que favorece la creación de una RNA, permite la elaboración de cualquier tipo de algoritmo de Machine Learning, cabe mencionar que también se puede usar para Deep Learning, facilita la adquisición de datos modelos de capacitación, predicciones y refinamiento de resultados, está disponible para el uso en computadores personales, pero es recomendado usarlo en su propio editor en la nube que es Colab.

Keras (MISMO COMENTARIO QUE LES PONGO DE TENSORFLOW) es un framework de alto nivel para el aprendizaje, escrito en Python y capaz de correr sobre los frameworks TensorFlow. Fue desarrollado con el objetivo de facilitar un proceso de experimentación rápida. Diseñado para construir por bloques la arquitectura de cada red neuronal, incluyendo redes convolucionales y modelos recurrentes, que son las que permiten, junto a los bloques "más tradicionales", entrenar aprendizaje profundo. Dicha biblioteca esta desarrollada para proyectos de Aprendizaje Profundo, además se puede ejecutar en TensorFlow y por lo consiguiente se puede correr en Colab.

5 Delimitación

En la siguiente investigación solamente se van a utilizar redes neuronales de perceptrón multicapa, cabe mencionar que este no es el único tipo de red, porque también se tiene lo que son Redes Neuronales Monocapa, Redes Neuronales Perceptrón Multicapa (RNPM), Redes Neuronales Convolucionales (RNC), Redes Neuronales Recurrentes (RNR), Redes de Base Radial (RBR) [16]. Pero para este experimento se van a utilizar RNAs que es el algoritmo que por el momento beneficiará a la presente investigación, cabe mencionar que no se usarán algoritmos genéticos, ya que si se implementan, se estará optimizando y el objetivo principal es utilizar el aprendizaje incremental para que acepte más datos de entrenamiento.

La inteligencia artificial se divide en dos, el Aprendizaje Maquina (Maching Learnin) y el Aprendizaje Profundo (Deep Learning. El Maching Learning son algoritmos que permiten a una computadora realizar procesos sin la intervención del ser humano y el Deep Learning es una parte del aprendizaje maquina que usa algoritmos de alto nivel, los cuales pretenden imitar una red neuronal humana. En el Maching Learning también se emulan las RNAs, a excepción de que en este tipo de inteligencia artificial no usa tantas capas ocultas, como el caso del aprendizaje profundo.

6 Consecuencias

Si el experimento funciona a la perfección ocurrirá lo siguiente:

1. Habrá menos olvido.
2. El aprendizaje tomará menos tiempo.

Como se menciona antes, se van a poder ingresar más datos sin tener que volver a codificar el modelo.

Los modelos de predicción van a ser más precisos, porque se podrán ingresar datos mensualmente o hasta semanalmente, esto provocará que el proyecto esté trabajando con datos actuales.

Pueden funcionar para proyectos y predicciones tan simples, por ejemplo, la preidcción climatológica, predicciones de la bolsa de valores, predicción del Bitcoin, entre otras.

7 Marco Teórico

7.1 Redes Neuronales Artificiales

Las redes neuronales artificiales (RNA) son procesos los cuales contienen simples unidades de procesamiento.

Como se puede observar, al mencionar RNA lo primero que se viene a la mente es el procesamiento biológico por el que transita el cerebro humano. El método principal de las redes neuronales es sacarle el máximo poder a los algoritmos de aprendizaje maquina, ya que las redes neuronales tienen un antecedente biológico.

Una red de una sola neurona solamente resuelve problemas lineales y tiene la siguiente forma:

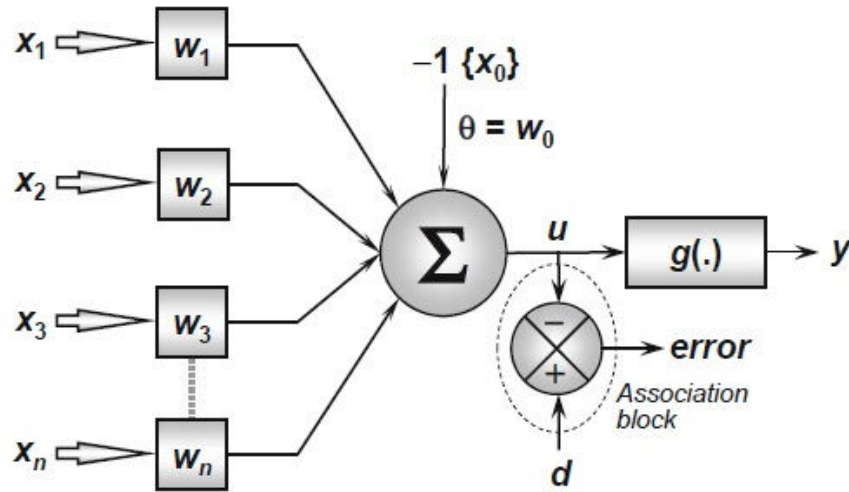


Figure 1: Red Neuronal Artificial Básica

Donde $\{\Sigma$ es la representación matemática de la neurona.

Los siguientes valores $\{x_1, x_2, \dots, x_n$ señalando a $\{w_1, w_2, \dots, w_n$ son los datos de entrada que se le dan a la red. Cuando la información entra en la neurona aquí se procesan los datos, la operación realizada es una sumatoria ponderada de ellos, dicha ponderación es asignada a ella como vemos en la siguiente

ecuación (7.1) Entonces la formula de la sumatoria quedaría de la siguiente manera: $\{w_1x_1 + w_2x_2 + w_3x_3$

Al verificar bien esta formula, se puede observar que se parece a la operación de una regresión la cual es: $\{y = w_0 + w_x$. Internamente la neurona realiza una regresión lineal, el parametro que permite a la neurona moverse verticalmente en la recta se conoce como sesgo, este valor se agrega a la conexión, el cual usualmente se le da un valor de 1.

Agregando este nuevo valor a la formula, queda de la siguiente manera:

$\{w_1 + w_2 + w_3 + b$ donde b es el sesgo.

Un inconveniente del uso de una sola neurona para experimentos es que solo va a resolver ejercicios parecidos a la puerta lógica AND u OR .

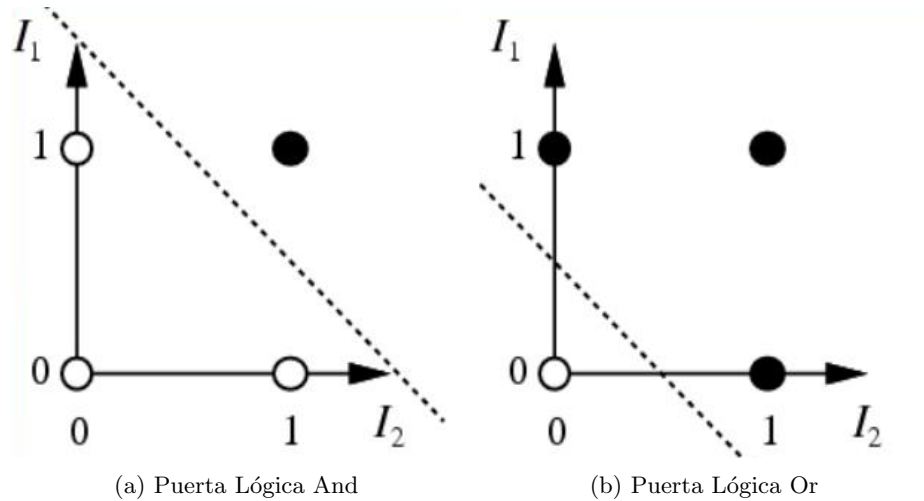


Figure 2: Puertas Lógicas [12]

Pero problemas de tipo XOR no puede, ya que como se nota, una sola neurona sirve para clasificar de un solo lado, así que no puede clasificar ejercicios como los de la imagen (3)

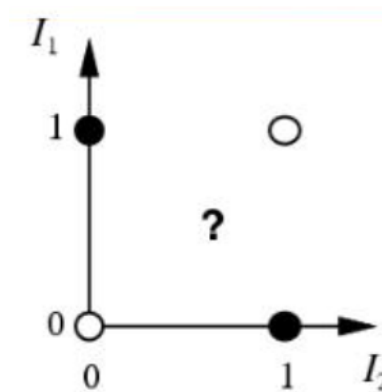


Figure 3: Puerta Lógica Xor [12]

Para solucionarlo se usan dos o más neuronas, además de la función de activación.

7.1.1 Función de Activación

Dicho método se utiliza cuando el modelo de RNA contiene dos o más neuronas. Esta función lo que provoca es dar al modelo una salida no lineal, para eso la formula (7.1) es distorciana para quedar de la siguiente manera: $\left\{ f(w_1x_1 + w_2x_2 + w_3x_3) \right\}$ En otras palabras esto es la suma de varias regresiones lineales, lo cual provoca que se obtenga un resultado no linel. Al hablar de funciones de activación se deben de comentar las más comunes, como lo es la función escalonada.

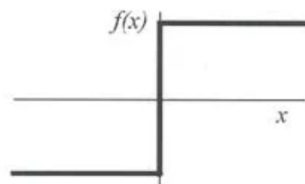


Figure 4: Función Escalonada

Dicha función es representada con:

$$f(x) = \begin{cases} 0 & : x < 0 \\ 1 & : x \geq 0 \end{cases}$$

La función sigmoidal, es una de las más comunes, su forma es:

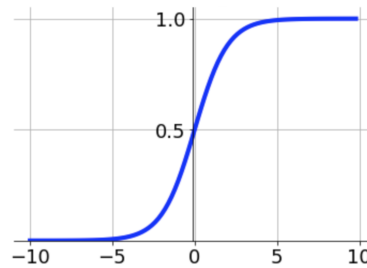


Figure 5: Función Sigmoide

La cual es representada por la siguiente formula:

$$f(x) = \sigma(x) = \left\{ \frac{1}{1 + e^{-x}} \right.$$

Aparte de agregar deformaciones en la linea. nos ayuda en cuestiones probabilísticas ya que se representa del rango de 0 a 1.

La función de Unidad Rectificada Lineal o ReLU, la cual es una función lineal que cuando es positiva es igual a 1 y cuando es negativa es constante a 0, su forma es:

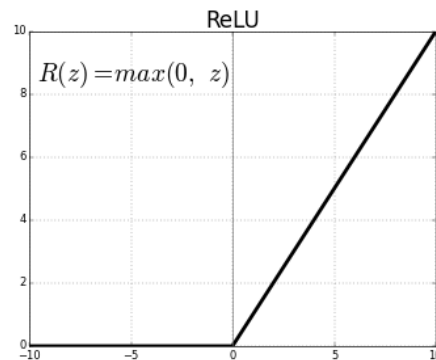


Figure 6: Función ReLU

se representa con la siguiente formula [6]:

$$f(x) = \begin{cases} 0 & : x < 0 \\ x & : x \geq 0 \end{cases}$$

La función Softmax transforma las salidas a una representación en forma de probabilidades, de tal manera que el sumatorio de todas las probabilidades de las salidas de 1, su gráfica es:

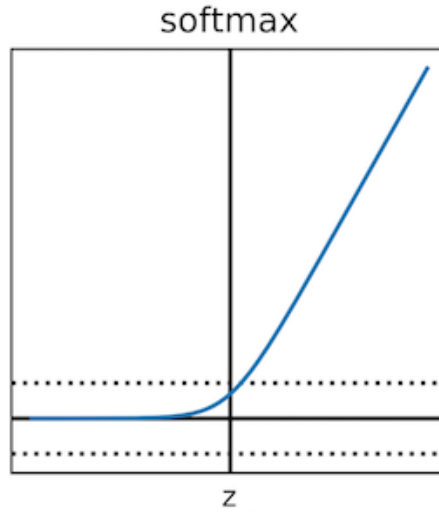


Figure 7: Función Softmax

su representación matemática es [3]:

$$f(z)_j = \left\{ \frac{e^{z_j}}{\sum_{k=1}^K e^{z_k}} \right.$$

Las redes neuronales presentan demasiadas utilidades las cuales ayudan a resolver problemas como los siguientes [11]: no linealidad, mapeo entrada-salida, aprendizaje robusto a errores en los datos de entrenamiento, entre otros.

Existen varios tipos de Redes Neuronales tales como: Redes Neuronales de Perceptrón Multicapa, Redes Neuronales Convolucionales, entre otras.

7.2 Redes Neuronales de Perceptrón Multicapa

Este tipo de redes entran en el paradigma de Aprendizaje Profundo, ya que como se explicó en la sección anterior (5) aquí la emulación de las neuronas humanas son más exactas. Además como se explica en esta sección (7.1) no es muy recomendado trabajar con una sola neurona por los problemas de tipo XOR, que por lo general es el que se presenta frecuentemente en las problemáticas.

Dichas neuronas se divide en tres capas, capa inicial, oculta y final, en las capas ocultas podemos tener más de una fila de neuronas, estas son las encargadas de realizar las operaciones para eliminar la linealidad de los proyectos.

En la sección vista anteriormte (7.1.1) la linealidad se elimina con las funciones de activación, dicha función es manipulable si se modifican los parametros de la red, esto permitira que se elabore un plano tridimensional, con este plano tridimensional se puede encontrar la solución al experimento planteado.

Como se puede observar en la siguiente imagen (6) esta red está construida por cuatro capas, la cual de ellas dos son ocultas, dentro de esas capas es donde se realiza la ejecución de las funciones de activación.

Cada neurona de color azul va a generar una función (puede ser sigmoideal, escalonada, entre otras), y cuando se llegue a la capa de salida, cada función se

va a sumar, de esta manera se obtendra una función no lineal que de resolución al proyecto.

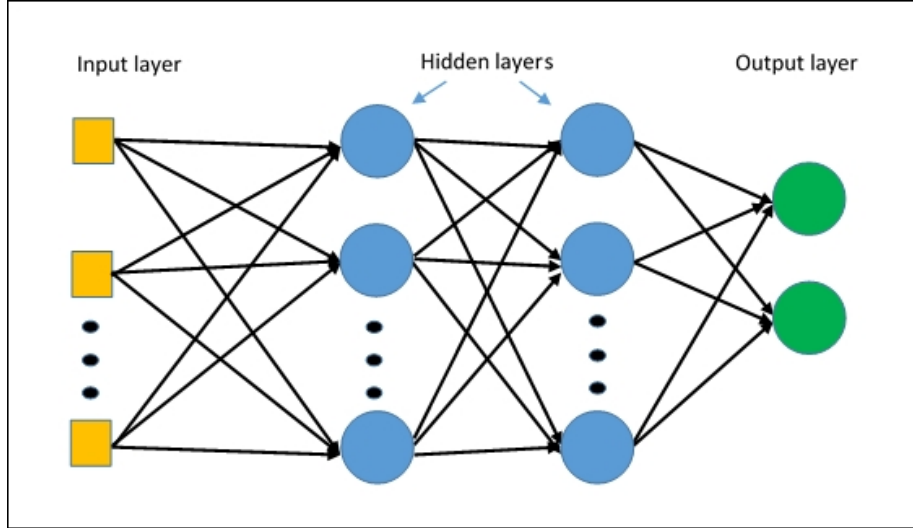


Figure 8: Perceptron Multicapa

7.3 Algoritmo Backpropagation

El backpropagation es el algoritmo donde se explica ¿Cómo aprende la red? Dicho algoritmo fue presentado en 1986, luego a resolver el problema del perceptrón ya que tenía muchas limitaciones, lo que provocó que varios experimentos de inteligencia artificial se pararan.

Este algoritmo permite que la red aprenda por sí sola, en 1986 se presentó [17] donde muestran un nuevo algoritmo el cual permite que una red neuronal pueda auto-ajustar todos sus parámetros para aprender una representación interna de la información que se está procesando.

Usando este algoritmo se podrá obtener las derivadas parciales del gradiente y del peso, las cuales sirven para la optimización de la red neuronal, dichas derivadas son:

$$\frac{\partial C}{\partial W^L}$$

Pero no se debe olvidar que también se debe calcular las derivadas del sesgo, la derivada anterior queda de la siguiente forma:

$$\frac{\partial C}{\partial b^L}$$

donde L pertenece al número de capa donde se encuentra.

Lo que permite a este algoritmo encontrar el error de la derivada es la *chain rule*, la cual en resumen nos da la siguiente ecuación :

$$C(a(Z^L))$$

Donde $\{Z^L$ representa el resultado de la suma ponderada, y la forma explicita de Z es:

$$Z^L = W^L X + b^L$$

, a es la función de activación y C es la función de Coste.

Al aplicar la *Chain Rule* se obtiene que las derivadas parciales a obtener son:

$$\frac{\partial C}{\partial w^L} = \frac{\partial C}{\partial a^L} \cdot \frac{\partial a^L}{\partial z^L} \cdot \frac{\partial z^L}{\partial w^L}$$

$$\frac{\partial C}{\partial b^L} = \frac{\partial C}{\partial a^L} \cdot \frac{\partial a^L}{\partial z^L} \cdot \frac{\partial z^L}{\partial b^L}$$

Como se ve el uso de todas estas derivadas parciales permiten encontrar el error. En otras palabras lo que realiza dicho algoritmo es terminar un proceso, si se encuentra un error, esta va a regresar hasta la neurona que da error, pero regresa cambiando el valor de w , este proceso se va a repetir hasta encontrar el error perfecto, el cual es donde el error disminuye a lo más bajo y el resultado de la red es lo más acertado.

Como se observa esta metodología es muy útil en el uso de redes neuronales, es por eso que se usa en la siguiente investigación [2] es usada para obtener un buen resultado en el aprendizaje incremental, como es explicado ahí es usado para que las redes obtengan una buena topología con buena actualización de pesos.

7.4 Redes Neuronales Convolucionales

Las redes neuronales convolucionales (RNC) son las que facilitan el reconocimiento de imagenes, el trabajo de reconocimiento de imagenes se puede elaborar con RNAs, pero esta tiene demasiadas desventajas, como lo que es perdida de datos, porque cuando se ingresa una imagen a la capa de entrada, esta se debe de convertir en vectores, si la imagen es de 100 pixeles por 100 pixeles se tendra un vector de 10000 pixeles, esto solo sera en imagenes a blanco y negro, si la imagen es del mismo tamaño pero a color se obtendra un vector de 30000 pixeles, ya que se usan los filtros RGB (Rojo, Verde, Azul).

Estas redes, presentan una solución para la problemática mencionada anteriormente, se basan en las conexiones que existen en el cerebro junto a la de los ojos.

Sus capas se distribuyen como se muestra imagen (7):

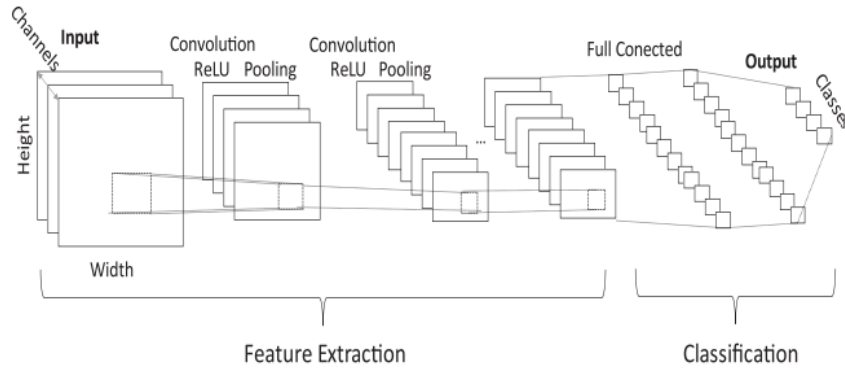


Figure 9: Arquitectura de una RNC [14]

La capa de entrada contiene una matriz o tensor de matriz (más de una matriz), las capas ocultas con dos etapas y la capa de salida, donde se obtendrá el resultado de la predicción.

La primer etapa es de extracción de características donde las capas están unidas a capas de *pooling*, estas capas son donde la imagen se divide en distintos kernels (filtros) el cual permitiera hacer submatrices a las imágenes y así poder recrearla en otra matriz, estos filtros deben ser menor de la imagen, se pueden tener más de un kernel para analizar bien la imagen, existen dos métodos de *pooling*: el *Max Pooling* y el *Average Pooling*, el *Max Pooling* extrae el valor máximo de las submatrices para crear otra matriz, como se puede observar en la imagen (10a), mientras que el *Average Pooling* el cual obtiene el promedio de las submatrices y con estos promedios genera otra matriz, como se puede ver en la imagen (10b) para un mejor resultado se tiene una capa de extracción o de convolución con otra capa de *pooling*, se pueden tener tantas capas de convolución como las de *pooling* para tener una mejor abstracción de conocimientos, en este tipo de conexiones se usan las funciones de activación conocidas como Relu (7.1.1).

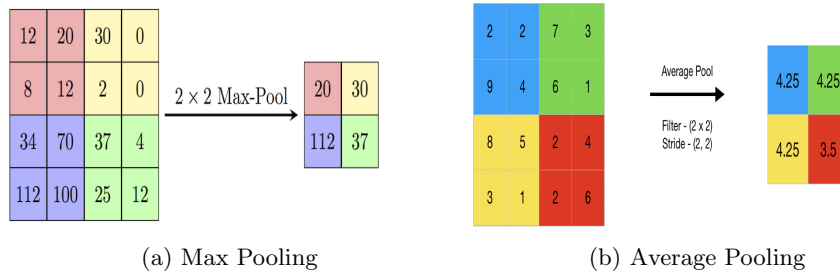


Figure 10: Métodos Pooling

La segunda etapa es la de Full Connected, que equivale a tener una red normal o en otras palabras tener una RNA, aquí es donde el modelo va a predecir que es lo que se presenta, puede decir si en la imagen hay un perro, gato o algún mueble, para eso la función de activación que se usa es la softmax (7.1.1), la cual se maneja con probabilidad, esto permitiera que se obtenga una buena

probabilidad de lo que se puede obtener por la capa de entrada [4, 14].

Así como las redes convolucionales, también existen las redes profundas (deep learning), sin embargo, como se comenta en esta sección (5) el presente trabajo solo se basará en redes del tipo multicapa perceptron usando el algoritmo de backpropagation, donde de ser demostrados los principios descritos, se podrá aplicar a cualquier otro tipo de algoritmo de aprendizaje.

7.5 Aprendizaje

7.5.1 Aprendizaje en humanos

El humano tiene una forma de aprendizaje muy particular, la cual se basa del estudio, donde lee, escribe y practica acerca de su tema de interés, pero dicho aprendizaje se puede ir olvidando, esta es una acción muy común que a cualquier persona le sucede. Existen estudios donde se comenta que existen tres motivos del porque se olvidan las cosas, proviene parte de la regularización de las emociones, el como se adquirieron los conocimientos, y porque el olvido es un proceso por el cual el ser humano transita a lo largo de su vida [13]. Pero cabe mencionar que esto no es lo único que causa la pérdida de memoria, ya que existe la déficit de memoria.

7.5.2 Aprendizaje Humano

Al momento de hablar del aprendizaje humano, se debe de hablar de la ciencia cognitiva, que es quien se encarga de descubrir esta incógnita, esta ciencia lo estudia de un modo multidisciplinario, el cual abarca las áreas de [1]: antropología, lingüística, filosofía, sicología del desarrollo, ciencia de la computación, neurociencia. Con el método de esta ciencia se pueden descubrir dos tipos de aprendizaje que son: el aprendizaje con comprensión y el aprendizaje Activo.

7.5.3 Aprendizaje con Comprensión

La comprensión es una actividad la cual se ha generado al momento de realizar cualquier tipo de lectura.

Teniendo un enfocamiento en el ámbito estudiantil, ya que es donde más se maneja esta táctica, esto es una practica algo compleja, sistemática y organizada, pues da el significado de la literatura, gracias a esto se puede obtener el contexto de la literatura. (Insisto que lo anterior esta mal dicho, pero no se me ocurre de que manera expresarlo)

Al conocer esto se puede decir con seguridad que para cualquier tipo de aprendizaje la comprensión es una parte primordial [15]. (Tambien siento que se debe de agregar un concepto en si o como dice Landassuri mas explicado)

7.5.4 Aprendizaje Activo

El aprendizaje de la forma en la que se conoce no es del todo efectiva, ya que el sistema educativo no se basa en el principio de *belongingness*, el cual esta asociado al estímulo con su respuesta, y esto es lo más importante para que el ser humano pueda aprender cualquier cosa.

Este tipo de aprendizaje se basa en la recepción de conocimientos y la práctica

donde se ponen en marcha los conocimientos adquiridos.

Otro concepto importante aquí es la tautología doble (*selbsttätiges Lernen*), que en palabras informales es convertirse en autodidacta, se puede observar que esto pertenece a dicho aprendizaje, porque usa el principio mencionado anteriormente [9].

7.5.5 Aprendizaje Incremental

Con el pasar de los años la tecnología a evolucionado, eso quiere decir que el Aprendizaje Automático se ha actualizado y que la cantidad de datos va aumentado con más frecuencia.

Se puede verificar como *"Una tarea de aprendizaje es incremental si los ejemplos de entrenamiento usados para resolverla están disponibles en horas extras, generalmente uno a la vez"* [8], si los resultados no se necesitan de manera urgente, este tipo de trabajos serán resueltos por algoritmos de aprendizaje no incremental.

Una área donde esto es de mucha utilidad es la *Robótica* porque este necesita estar en constante entrenamiento [8].

Dicha forma de aprender fue inspirada en la forma en que el humano aprende y esta más rápida, fue por esto que fue adoptada por el aprendizaje máquina.

Con el paso del tiempo se ha convertido en un paradigma del aprendizaje automático, aquí el aprendizaje toma el lugar de nuevos ejemplos para juntarlos y conforme van aprendiendo estos toman el lugar de los ejemplos ya aprendidos [11].

7.5.6 Algoritmos de Aprendizaje Incremental

"Un algoritmo de aprendizaje es incremental si, para cualquier muestra de entrenamiento dada:

$$e_1, \dots, e_s \tag{1}$$

produce un secuencia de hipótesis

$$h_0, h_1, \dots, h_n \tag{2}$$

tal que h_{i+1} depende solo de h_i (Tengo duda con la palabra hola si lo tradujiste fijate bien que hayas copiado bien, luego los simbolos no los copia como es) y del ejemplo actual e " [8], como se observa, estos son algoritmos que permiten a la inteligencia artificial poder realizar actividades de predicción de una manera más eficaz.

Un ejemplo del uso de esta rama es el proyecto *COBWEB*, donde se trata de categorizar el número de Clúster y la pertenencia de dichas categorías por medio de una métrica probabilística global, esto lo realiza por medio de que se agrega una nueva categoría, este proceso lo que realizará es actualizar todas las probabilísticas con los nuevos datos recabados [7].

AQUÍ FALTA DESCRIBIR A DETALLE CÓMO SE LLEVA ACABO EL APRENDIZAJE INCREMENTAL, SE PUEDEN BASAR EN EL TRABAJO DE BULLINARIA Y LEANR ++, EXPLIQUENLO A DETALLE PARA QUE SEA SU ESTADO DEL ARTE

8 Hipotesis

Al tener más de dos capas de pesos duplicados con sus respectivas tasas de aprendizaje, permite tener un menor olvido de la información previa aprendida, en un modelo de aprendizaje incremental con redes neuronales artificiales del tipo MLP, al usar el algoritmo de Backpropagation.

9 Metodología

El primer paso a realizar en esta investigación es recrear el código mostrado en [2], donde se describe la implementación de una red neuronal multicapa usando el algoritmo de entrenamiento backpropagation en el lenguaje de programación Python. Para esto se utilizará el conjunto de datos de Optical Digits, en donde se tendrá que preprocesar los datos, para eliminar registros inválidos.

Posteriormente se implementará una nueva versión del código en donde se experimentará con mas de dos capas de pesos duplicados para mejorar la tasa de olvido de información al momento de usar el aprendizaje incremental. Para ello se explorará incrementando gradualmente el nueron de capas (esto no lo entendi,neuronas de capas?)de pesos duplicados, hasta llegar al punto en que mas capas no generen un decremento de las tasas de olvido/error.

(REVISEN LA REDACCIÓN Y MEJORENLA DE FAVOR) Cuando los dos proyectos se tengan, se realizará una comparación, donde se verá cual de estos dos experimentos es más eficaz en proyectos de la vida real.

10 Cronograma de Actividades

FASES	PRIMER SEMESTRE						SEGUNDO SEMESTRE					
	MESES											
	1	2	3	4	5	6	1	2	3	4	5	6
Delimitación del tema												
Preprocesamiento de datos												
Implementación del algoritmo												
Extensión del algoritmo												
Comparación de ambas implementaciones												
Preparación del trabajo para congresos												
Escritura de tesis												
Pre-examen												
Atender observaciones												
Examen profesional												

11 Organización del Capitulo

En el capítulo 2 se verá lo que es el aprendizaje humano y el aprendizaje incremental con sus algoritmos, se describirán las redes neuronales artificiales.

En el capítulo 3 se implementará el artículo de John A. Bullinaria, como funciona, resultado que da al pasar los datos que dice para comprobar que funciona como menciona en su artículo. En el capítulo 4 se explicará como se hará la modificación a su algoritmo, cuantas capas se van a poner, como se van a repartir las tasas de aprendizaje.

Posteriormente en el capítulo 5 se mostrará una comparación de los resultados de ambos trabajos. En el capítulo 6 se verán las conclusiones y trabajo futuro.

References

- [1] J Bransford, A Brown, and R Cocking. “Cómo aprende la gente: cerebro, mente, experiencia, y escuela.” In: *Revista del Instituto de Matemática y Física* (2000), pp. 44–64.
- [2] John A Bullinaria. “Evolved Dual Weight Neural Architectures to Facilitate Incremental Learning.” In: *IJCCI*. 2009, pp. 427–434.
- [3] Calvo Diego. *Función de activación – Redes neuronales*. Dec. 2018. URL: <https://www.diegocalvo.es/funcion-de-activacion-redes-neuronales/>.
- [4] Jaime Durán Suárez. “Redes neuronales convolucionales en R: Reconocimiento de caracteres escritos a mano.” In: (2017).
- [5] Ryan Elwell and Robi Polikar. “Incremental Learning of Concept Drift in Nonstationary Environments.” In: *IEEE Transactions on Neural Networks* 22.10 (2011), pp. 1517–1531. DOI: 10.1109/TNN.2011.2160459.
- [6] Sarahí Silva Estefanía Freire. *Redes neuronales - Bootcamp AI*. Dec. 2021. URL: <https://bootcampai.medium.com/redes-neuronales-13349dd1a5bb>.
- [7] Douglas H Fisher. “Knowledge acquisition via incremental conceptual clustering.” In: *Machine learning* 2.2 (1987), pp. 139–172.
- [8] Christophe G. Giraud-Carrier. “A Note on the Utility of Incremental Learning.” In: *AI Commun.* 13 (2000), pp. 215–224.
- [9] Günter L. Huber. “Aprendizaje activo y metodologías educativas.” In: (2008). URL: http://www.revistaeducacion.mec.es/re2008/re2008_04.pdf.
- [10] Ai-Jun Li. “An improved algorithm for incremental learning learn++.” In: *2008 International Conference on Wavelet Analysis and Pattern Recognition*. Vol. 1. IEEE. 2008, pp. 310–315.
- [11] Yuan Liu. “Yuan Liu INCREMENTAL LEARNING IN DEEP NEURAL NETWORKS.” In: 2015.
- [12] Ayanna McMahon. *Redes Neuronales Artificiales*. Dec. 2014. URL: <https://www.slideserve.com/ayanna-mcmahon/redes-neuronales-artificiales>.
- [13] Simon Nørby. “Why Forget? On the Adaptive Value of Memory Loss.” In: *Perspectives on Psychological Science* 10 (2015), pp. 551–578.

- [14] Flor G. Ortiz-Gomez et al. “Convolutional Neural Networks for Flexible Payload Management in VHTS Systems.” In: *IEEE Systems Journal* 15.3 (2021), pp. 4675–4686. DOI: 10.1109/JSYST.2020.3020038.
- [15] Karel Pérez Ariza and José Emilio Hernández Sánchez. “Aprendizaje y comprensión. Una mirada desde las humanidades.” In: *Humanidades Médicas* 14.3 (2014), pp. 699–709.
- [16] Paloma Royo. *Qué son las redes neuronales y cuál es su aplicación en el marketing*. 2021. URL: <https://artyco.com/que-son-las-redes-neuronales-y-cual-es-su-aplicacion-en-el-marketing/>.
- [17] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. “Learning representations by back-propagating errors.” In: *nature* 323.6088 (1986), pp. 533–536.
- [18] Luis A Cruz Salazar, David J Muñoz Aldana, and Juan A Contreras Montes. “Implementación de redes neuronales y lógica difusa para la clasificación de patrones obtenidos por un Sónar.” In: *2013 II International Congress of Engineering Mechatronics and Automation (CIIMA)*. IEEE. 2013, pp. 1–6.
- [19] Beibei Zhao et al. “Prediction method of 5G high-load cellular based on BP neural network.” In: *2022 8th International Conference on Mechatronics and Robotics Engineering (ICMRE)*. IEEE. 2022, pp. 148–151.