

Aprendizaje Incremental para la Tarea de Reconocimiento de Dígitos con Redes Neuronales Artificiales

by

Fragoso García Sandra
González Hernández Luis Ángel

CUUAEM VM
UAEM Diciembre 2022

Resumen

El aprendizaje incremental es un área de la Inteligencia Artificial la cual permite agregar nuevo conocimiento a un modelo (e.g. Redes Neuronales Artificiales) sin la necesidad de entrenar el modelo con toda la información histórica de la tarea en cuestión. Neuronales Artificiales enfocada en la clasificación de dígitos escritos a mano usando el algoritmo de entrenamiento de backpropagation, con redes Multi capa Perceptron y duplicación de pesos múltiples simulando memoria a corto y largo plazo para mejorar los resultados presentados en

Dedicatoria

Agradecimientos

Sus agradecimientos

Índice general

Índice de figuras

Índice de tablas

kdfdfsjb gdfsjb gdfjksh of Acronyms

Capítulo 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 dado, 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 [?] o clasificación, por ejemplo la clasificación de metales y rocas por medio de RNAs y lógica difusa [?].

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 [?].

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 [?]. 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 pérdida 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 [?], el cual 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 [?] permiten notar un mejora en tareas de aprendizaje incremental, teniendo menos pérdida de información en comparación de implementaciones anteriores como el algoritmo Learn++ [?, ?].

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

[?].

Capítulo 2

Marco Teórico

2.1. Redes Neuronales Artificiales

Las redes neuronales artificiales (RNA) son modelos computacionales de la Inteligencia Artificial los cuales contienen simples unidades de procesamiento llamadas neuronas. Ellas se inspiran en el cerebro humano, tomando como base la conectividad entre neuronas y el aprendizaje que pueden tener. Un perceptron o neurona (artificial) solamente resuelve problemas lineales y tiene la siguiente forma:

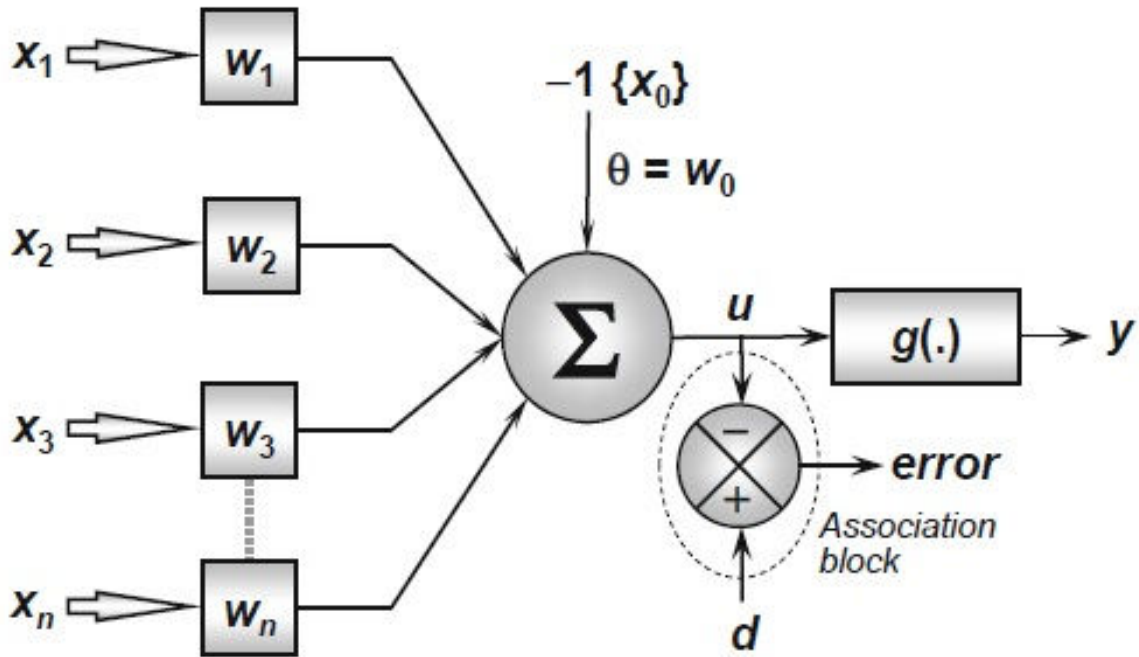


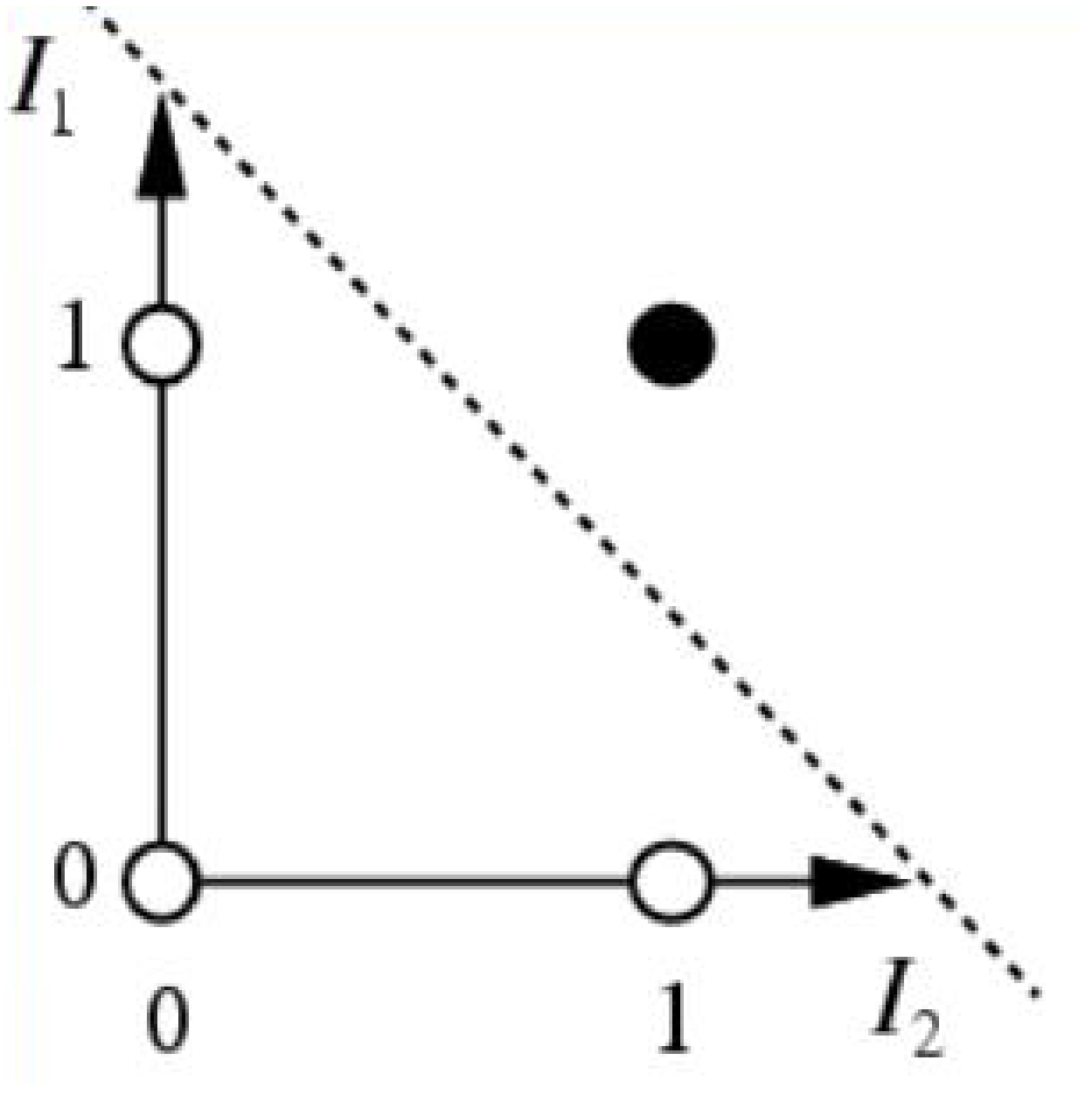
Figura 2.1: Red Neuronal Artificial Básica

Donde Σ es la representación matemática de la neurona., x_1, x_2, \dots, x_n son las variables de entrada a la red. w_1, w_2, \dots, w_n son los pesos con los cuales se van a ponderar las entradas, es decir multiplicar cuando la información entra en la neurona. Posterior a multiplicar el peso por la entrada correspondiente, se suman todos esos valores $w_1x_1 + w_2x_2 + w_3x_3$

Al revisar esta formula, se puede observar que se parece a la operación de una regresión la cual es: $y = w_0 + w_ix_i$, de esta forma, internamente la neurona realiza una regresión lineal. En su contraparte, el parámetro que permite a la neurona trazar una recta cruzando el eje y en el plano cartesiano (eje de las ordenadas), a ello se conoce como sesgo (del inglés *bias*), este valor se agrega a la conexión, el cual usualmente se le da un valor de 1.

Agregando este nuevo valor a la fórmula, queda de la siguiente manera: $y = \Sigma w_ix_i + w_0b$, donde b es el sesgo.

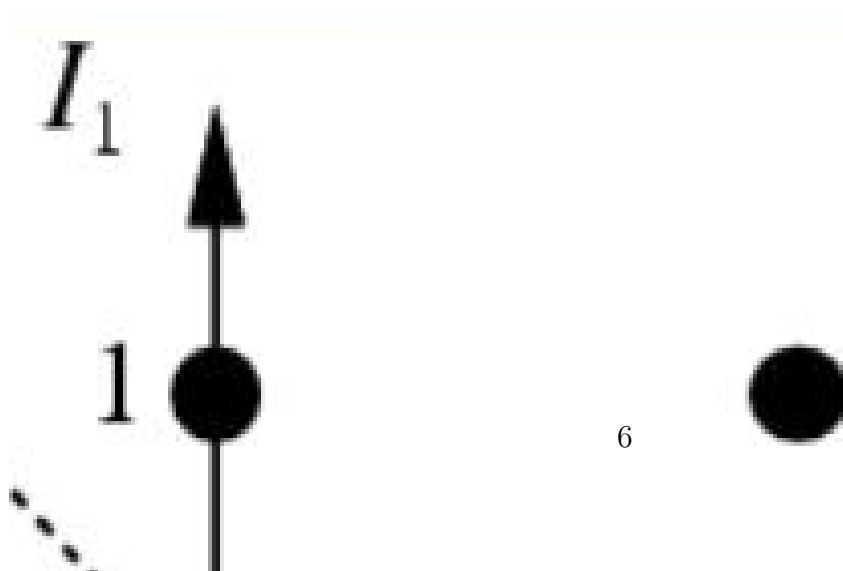
Un inconveniente del uso de una sola neurona para experimentos es que solo va a resolver ejercicios parecidos a la puerta lógica AND u OR



[H]0.49

Figura 2.2: Puerta Lógica And

[H]0.49



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 Figura fig:fig3.

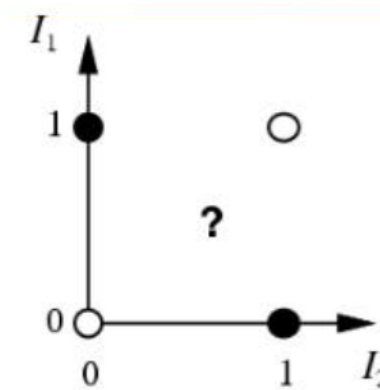


Figura 2.5: Puerta Lógica Xor

Para solucionarlo se usan dos o más neuronas, además de la función de activación, que es la que permite pasar la información de una neurona a otra, en un rango especificado, y la cual se describirá en la siguiente sección.

2.1.1. 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 segunda fórmula presentada es distorcionada para quedar de la siguiente manera: $f(w_1x_1 + w_2x_2 + w_3x_3 + b_0)$ para el caso de 3 entradas.

Al hablar de funciones de activación se deben de comentar las más comunes, como lo es la función escalonada.

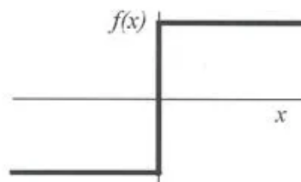


Figura 2.6: Función Escalonada

Dicha función es representada con:

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

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

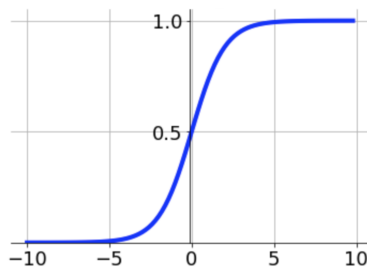


Figura 2.7: Función Sigmoide

La cual es representada por la siguiente formula:

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

Aparte de realizar dicha transformación, las funciones de transferencia ayudan en cuestiones probabilísticas ya que se representa del rango de 0 a 1. Aunado a que por ejemplo, la función sigmoideal es diferenciable, lo que permite al algoritmo de backpropagation poder llevar a cabo el entrenamiento.

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:

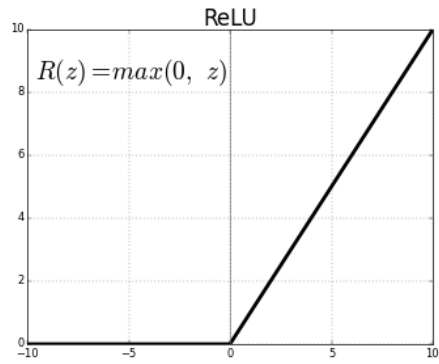


Figura 2.8: Función ReLU

Se representa con la siguiente formula [?]:

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

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:

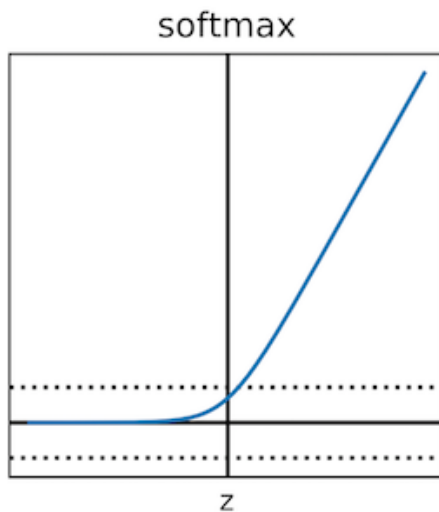


Figura 2.9: Función Softmax

Su representación matemática es [?]:

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

Las redes neuronales presentan demasiadas utilidades las cuales ayudan a resolver problemas como los siguientes [?]: 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, las cuales se describirán brevemente más adelante.

2.2. Redes Neuronales de Perceptrón Multicapa

Las Redes Neuronales de Perceptrón Multicapa se pueden dividir en dos capas (las de entrada y salida), pero también en tres o más capas (la de entrada, una o más capas ocultas y la de salida). En las capas ocultas se pueden tener más de una fila de neuronas, las cuales son las encargadas de realizar las operaciones para eliminar la linealidad de los datos. También, como se comentó anteriormente, Sec. sec: activation la linealidad de los datos se elimina con las funciones de activación, las cuales modifican los parámetros de la red, permitiendo que se elabore un plano tridimensional, con el cual se puede encontrar la solución al problema planteado.

Además como se explicó anteriormente, no es muy recomendado trabajar con una sola neurona por los problemas presentados al resolver, tareas como el XOR donde se requieren de dos líneas rectas para clasificar el problema correctamente.

Como se puede observar en la figura fig:fig8, para este caso se cuenta con una MLP que consta de 4 capas, 1 de entrada, 2 ocultas y 1 de salida. En las capas ocultas y de salida se lleva a cabo el procesamiento de las funciones de activación, donde cabe notar que no es así para la primera capa de entrada, donde únicamente sirve para representar las entradas al modelo, i.e. ahí no hay funciones de transferencia.

Así, las neuronas de color azul cuentan con 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 obtendrá una función no lineal que de resolución a la tarea a resolver.

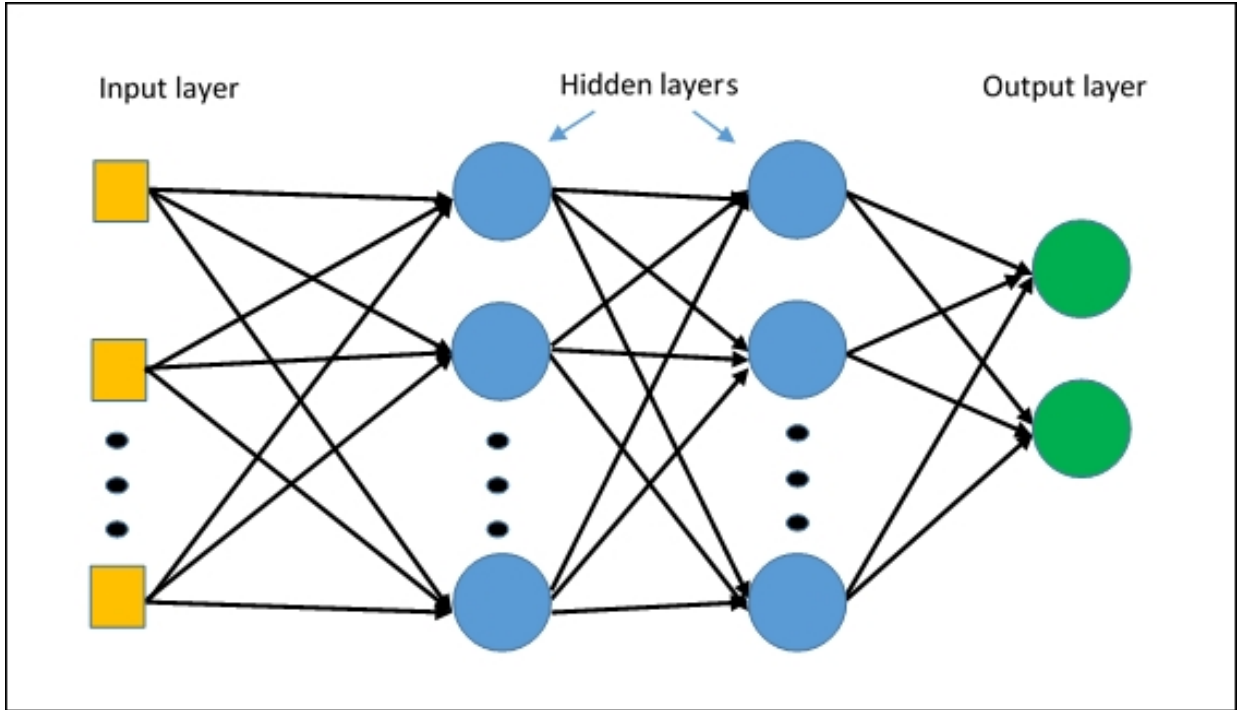


Figura 2.10: Perceptron Multicapa

2.3. Algoritmo Backpropagation

El backpropagation es un algoritmo de aprendizaje que permite que una red neuronal pueda auto-ajustar todos sus parámetros para aprender una representación interna de la información que se está procesando. Llegó a dar solución a la limitante del perceptron, este resuelve los problemas lineales, la cual es que no se puede extender a redes más complejas, es decir, a problemas no lineales.

Usando este algoritmo se podrán obtener las derivadas parciales del gradiente y del peso, las cuales sirven para la optimización de la red neuronal. Pero también se deben de calcular las derivadas del sesgo, donde se encuentra el número de capa donde esta la falla.

El uso de estas derivadas parciales permite encontrar el error, en otras palabras, lo que realiza dicho algoritmo es terminar un proceso y si se encuentra un error, este va a regresar hasta la neurona donde se encuentra este, pero va regresando desde la capa de salida hacia la primer capa oculta, 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.

Con lo anterior expuesto, se puede decir que esta metodología es muy útil en el uso de redes neuronales, es por eso que se usa en la investigación [?], 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.

2.4. Redes Neuronales Convolucionales

Las redes neuronales convolucionales (RNC) son las que facilitan el reconocimiento de imágenes, el trabajo de reconocimiento de imágenes se puede elaborar con RNAs, pero esta tiene demasiadas desventajas, como lo que es pérdida 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 tendrá un vector de 10000 pixeles, esto solo será en imágenes a blanco y negro, si la imagen es del mismo tamaño pero a color se obtendrá 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 Figura fig:fig9:

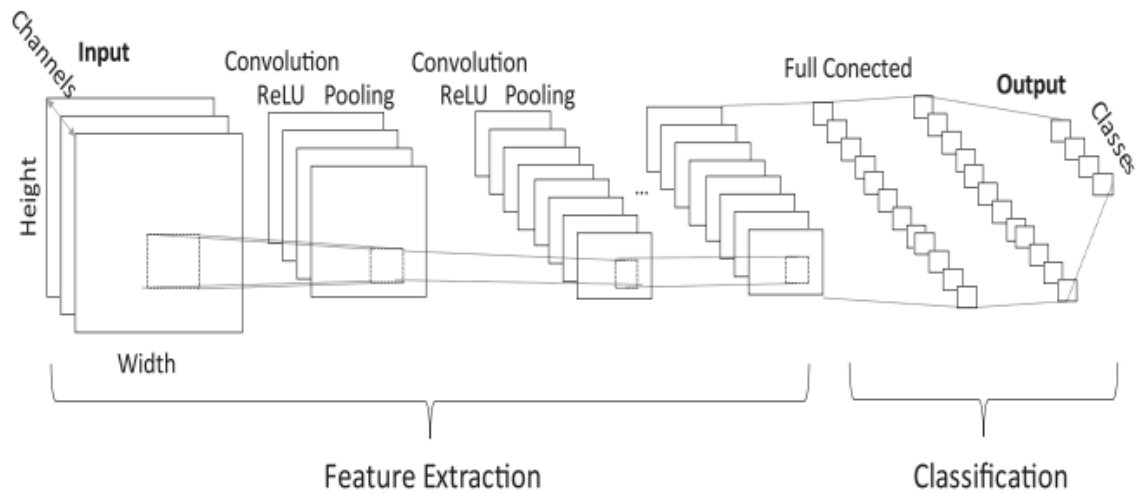


Figura 2.11: Arquitectura de una RNC

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 permitirá 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 Figura fig:f3, 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 Figura fig:f4 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 subsec: relu.

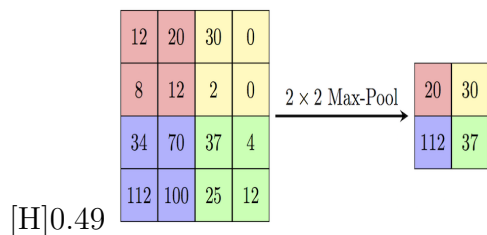


Figura 2.12: Max Pooling

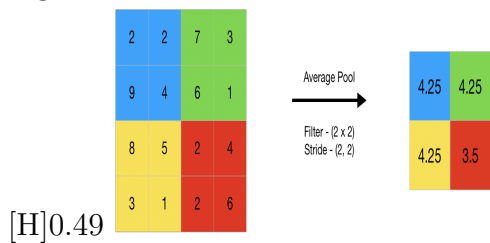


Figura 2.13: Average Pooling

Figura 2.14: Métodos Pooling

La segunda etapa es la de Full Conected, 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 softmax subsec: softmax, la cual se maneja con probabilidad, esto permitirá que se obtenga una buena probabilidad de lo que se puede obtener por la capa de entrada [?, ?].

Este tipo de redes es una parte del Deep Learning, y no se entrará tanto en detalle ya que el presente trabajo de investigación, se enfoca en las redes multicapa.

2.5. Aprendizaje

2.5.1. 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 [?]. Pero cabe mencionar que esto no es lo único que causa la perdida de memoria, ya que existe la déficit de memoria.

2.5.2. 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 [?]: 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.

2.5.2.1. 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 y durabilidad del aprendizaje. Al conocer esto se puede decir con seguridad que para cualquier tipo de aprendizaje la comprensión es una parte primordial [?].

2.5.2.2. 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 estimulo 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 [?].

2.5.3. 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 aumentando 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* [?], 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 [?].

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 [?].

2.5.3.1. Algoritmos de Aprendizaje Incremental

El algoritmo de aprendizaje incremental puede definirse como aquel que cumple los siguientes criterios: 1) Ser capaz de aprender y actualizarse con cada nuevo dato etiquetado

o no etiquetado. 2) Conservar los conocimientos adquiridos previamente. 3) No debe requerir el acceso a los datos originales. 4) Generar una nueva clase o cluster cuando sea necesario. Dividir o fusionar los clusters cuando sea necesario. 5). Ser de naturaleza dinámica con el entorno cambiante [?].

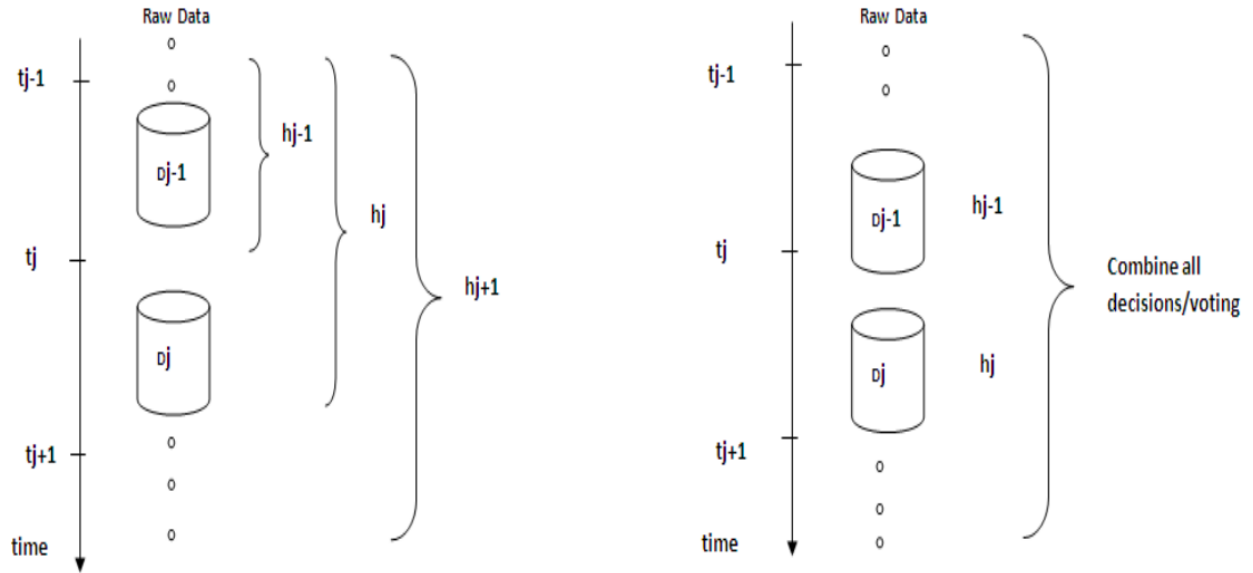


Figura 2.15: Dos enfoques tradicionales del aprendizaje incremental.

1 Metodología de acumulación de datos. 2 Metodología de aprendizaje por conjuntos.

Como se observa en la Figura 11, en el primer método, cuando se recibe una nueva porción de datos D_j , se descarta h_{j-1} y se desarrolla una nueva hipótesis h_j , basada en todos los datos disponibles acumulados hasta el momento. Y en el segundo método, cuando se recibe una nueva porción de datos D_j , se desarrolla una única hipótesis nueva o un conjunto de hipótesis nuevas basadas en los nuevos datos. Por último, se puede utilizar un mecanismo de votación para combinar todas las decisiones de las diferentes hipótesis y obtener la predicción final.

Por ejemplo, si se deja que D_{j-1} represente la porción de datos recibida entre el tiempo t_{j-1} y t_j , y que la hipótesis h_{j-1} se desarrolle sobre D_{j-1} .

El sistema aprenderá información de forma adaptativa cuando se reciba una nueva porción de datos D_j . En el método de aprendizaje por conjuntos, se desarrolla una nueva hipótesis h_j o un conjunto de hipótesis $H: h_1, h_2, \dots, h_M$, basadas en los nuevos datos. A continuación, se utiliza el mecanismo de votación para combinar todas las decisiones de las diferentes hipótesis y llegar a la predicción final. La mayor ventaja de este enfoque es que no se requiere almacenar los datos vistos anteriormente, el conocimiento se ha almacenado en la serie de hipótesis desarrolladas a lo largo de la vida de aprendizaje.

Conocimiento en el momento t :

D_t es un trozo de datos con n instancias ($i=1, \dots, n$)

(x_i, y_i) es una instancia en el espacio de características m -dimensional X

$Y_i \in Y = 1, \dots, K$ clases

Función de distribución D_f

Una hipótesis h_t , desarrollada por los datos basados en D_t con P_t

La nueva entrada estará disponible en el momento $(t+1)$

Algoritmo de aprendizaje:

1. Encontrar la relación entre D_t y D_{t+1}
2. Actualizar la función de distribución inicial D_{t+1}
3. Aplicar la hipótesis h_t a D_{t+1} y calcular el pseudoerror de h_t
4. Refinar la función de distribución para D_{t+1}
5. Se desarrolla una hipótesis por los datos basados en D_{t+1} con P_{t+1}

6. Repetir el procedimiento cuando se reciba la siguiente porción del nuevo conjunto de datos.

Resultado: La hipótesis final.

Un algoritmo de aprendizaje es incremental si, para cualquier muestra de entrenamiento dada: e_1, \dots, e_s produce una secuencia de hipótesis h_0, h_1, \dots, h_n tal que

$$h_{i+1}$$

depende solo de

$$h_i$$

y del ejemplo actual e " [?], 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 [?].