

# DEEP LEARNING: VARIANTES Y APLICACIONES

[1.- ¿Qué es “Deep learning”?](#)

[2.- Algoritmos de “Deep Learning”](#)

[2.1.- Red neuronal profunda \(Deep neural network\)](#)

[2.1.1.- Auto-codificadores](#)

[2.1.2.- Auto-codificadores dispersos](#)

[2.1.3.- Red profunda mediante auto-codificadores](#)

[2.1.4.- Máquinas de Boltzmann restringidas \(Restricted Boltzmann Machines\)](#)

[2.1.5.- Red de creencia profunda \(Deep belief network\)](#)

[3.- Conclusiones](#)

[4.- Bibliografía](#)

**Autores:**  
**Gacel Ivorra Rodríguez**  
**Ismael Piñeiro Ramos**

## 1.- ¿Qué es “Deep learning”?

El *deep learning*, conocido en español como aprendizaje profundo, es una rama del aprendizaje automático (*machine learning*). Se basa en una abstracción de alto nivel de los datos utilizando algoritmos de aprendizaje automático, principalmente utilizando arquitecturas complejas de datos con el apoyo de transformaciones no lineales múltiples.

Hoy en día, el término como tal se encuentra de moda, ya que los avances dados en el campo de la inteligencia artificial, robótica y otros, se apoyan en el aprendizaje automático, que esté a su vez utiliza en gran medida las herramientas proporcionadas por el aprendizaje profundo.

El científico jefe de Baidu, profesor de Stanford University y ex-empleado de Google, Andrew NG, describe el aprendizaje profundo de la siguiente manera:

*“Es una tecnología de aprendizaje que simula el cerebro humano de manera libre. El cerebro humano trabaja debido a que una gran cantidad de neuronas hablan entre sí. Pues el deep learning es eso, una simulación por computador, de miles de millones de neuronas hablando entre sí.”*

Para hacernos una idea de forma sencilla de la diferencia que aporta el Deep Learning al Machine Learning convencional podemos decir que Deep Learning es un conjunto de técnicas muy potentes que permiten crear nuevos tipos de redes neuronales más avanzadas.

Las redes entrenadas con ayuda de algoritmos de aprendizaje profundo, no solo superan en precisión a los mejores métodos alternativos, sino también en una serie de tareas en las que mostraron nociones de comprensión del sentido de la información suministrada (por ejemplo, a la hora de reconocer imágenes, en el análisis de información de texto, etcétera).

## 2.- Algoritmos de “Deep Learning”

### 2.1.- Red neuronal profunda (*Deep neural network*)

Una red neuronal profunda, no es más que una red que cuenta con diferentes capas ocultas entre las entradas y las salidas de la misma. De esta manera se consigue modelar relaciones no lineales complejas.

Gracias a las capas ocultas añadidas se permite la composición de características de las capas inferiores, por lo que podemos modelar datos complejos con relaciones no lineales con un menor número de neuronas y en menor tiempo que una red neuronal convencional.

El algoritmo de entrenamiento más sencillo y extendido de las redes neuronales profundas es el *backpropagation* estándar y la actualización de pesos se realizan mediante el gradiente por descenso estocástico.

Este tipo de red es ampliamente usado hoy en día, aunque hay que tener cuidado ya que si no se trabaja bien con los datos de entrada se puede obtener un gran problema de sobreentrenamiento y mucho tiempo de computación.

Los principales usos de este tipo de algoritmo de aprendizaje profundo son:

- Reconocimiento de imágenes
- Reconocimiento de voz
- Reconocimiento de patrones
- Reconocimiento de lenguajes (analizadores sintácticos)

Con las técnicas tradicionales de uso de redes neuronales, al entrenar redes muy profundas existe un problema porque durante la propagación del error hacia atrás (*backpropagation*) éste se va diluyendo y las primeras capas casi no se entrenan, sin embargo, haciendo uso de la técnica de Deep Learning llamada apilación de auto-codificadores, podemos obtener muy buenos resultados con este tipo de redes.

### 2.1.1.- Auto-codificadores

Los auto-codificadores son otra de las herramientas utilizadas habitualmente para implementar Deep Learning un tipo de redes neuronales de aprendizaje no supervisado que aplica el algoritmo de *backpropagation* para ser entrenada.

Normalmente se componen de 3 capas:

- **Capa de entrada:** Número de neuronas  $\rightarrow n$
- **Capa oculta:** Número de neuronas  $\rightarrow m < n$
- **Capa de salida:** Número de neuronas  $\rightarrow l = n$

Estas redes neuronales tratan de obtener la función más aproximada a  $h(x) = x$ , es decir, intenta producir a la salida exactamente la misma información que recibe a la entrada.

La clave para que esto no sea un proceso trivial es el número de neuronas de la capa oculta.

Por ejemplo, si tenemos una imagen de 10x10 píxeles (100 píxeles en total), deberíamos tener una red neuronal con 100 neuronas de entrada y 100 de salida.

Imaginemos que en la capa oculta colocamos un total de tan solo 50 neuronas.

Si pensamos en esto, dado que exigimos a la red que produzca a la salida el mismo resultado que recibe a la entrada, se entiende que cuando se complete el entrenamiento contendrá en su capa oculta una versión codificada de una imagen de 100 píxeles en tan solo 50, además de tener disponible la función codificadora y la función decodificadora.

Precisamente por eso, este tipo de redes se denominan auto-codificadores, son capaces de descubrir por sí mismos (sin supervisión) una forma alternativa de codificar la información en su capa oculta, que podrá utilizarse después en procesos de predicción.

### 2.1.2.- Auto-codificadores dispersos

El ejemplo de red que comprime información es interesante, pero no es el uso más habitual de estas redes en *Deep Learning*. ¿Qué pasaría si la capa oculta, en lugar de tener menos neuronas, tuviese más que las capas de entrada y salida?

Los auto-codificadores dispersos son un tipo de auto-codificador cuya cantidad de neuronas en la capa oculta es bastante superior a la dimensión de la entrada, pero estas tienen una activación dispersa. La activación dispersa es cuando la cantidad de neuronas inactivas en la capa oculta supera significativamente la cantidad de activas.

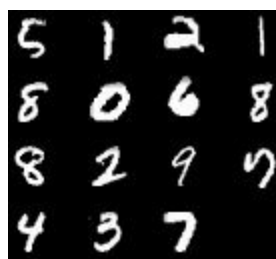
En una red con la arquitectura mencionada, existe el riesgo de que la red no aprenda nada: que se limite a copiar la entrada a un subconjunto de las neuronas en la capa oculta (dejando las neuronas sobrantes sin usar), y luego las vuelva a copiar a la salida.

Una técnica sencilla para conseguir una activación dispersa de las neuronas de la capa oculta trata de, al componer los ejemplos para el entrenamiento, en lugar de indicar el mismo valor para las entradas y las salidas, introducir un poco de **ruido en el vector de entrada**, y dejar el de salida sin ruido. De esta forma, **la red está forzada a generalizar**, porque tendrá varios ejemplos ligeramente diferentes de entrada que producen la misma salida.

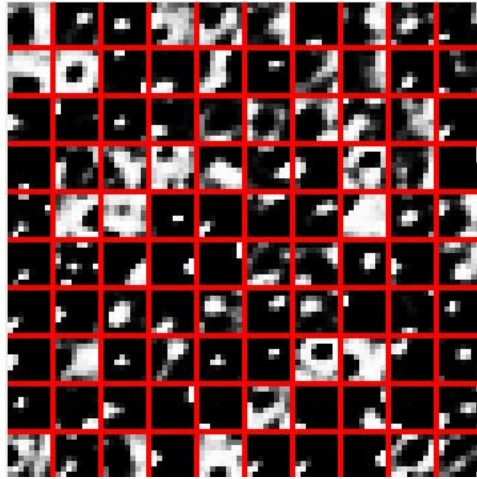
La representación intermedia en **la capa oculta tendrá que enfocarse en las características comunes de todas las versiones de la misma imagen con diferentes ruidos**.

Para entender mejor el resultado obtenido con el entrenamiento y las ventajas que se obtienen, vamos a ver un ejemplo:

Supongamos que tenemos imágenes de números manuscritos tal que así:



Cogemos parches de 8x8 de todas estas imágenes y las introducimos como entradas para el entrenamiento de una red neuronal que contenga 100 neuronas en la capa intermedia. Cuando tengamos la red entrenada, si dibujamos el parche que activa completamente cada una de las 100 neuronas de la capa oculta, obtenemos 100 patrones como estos:



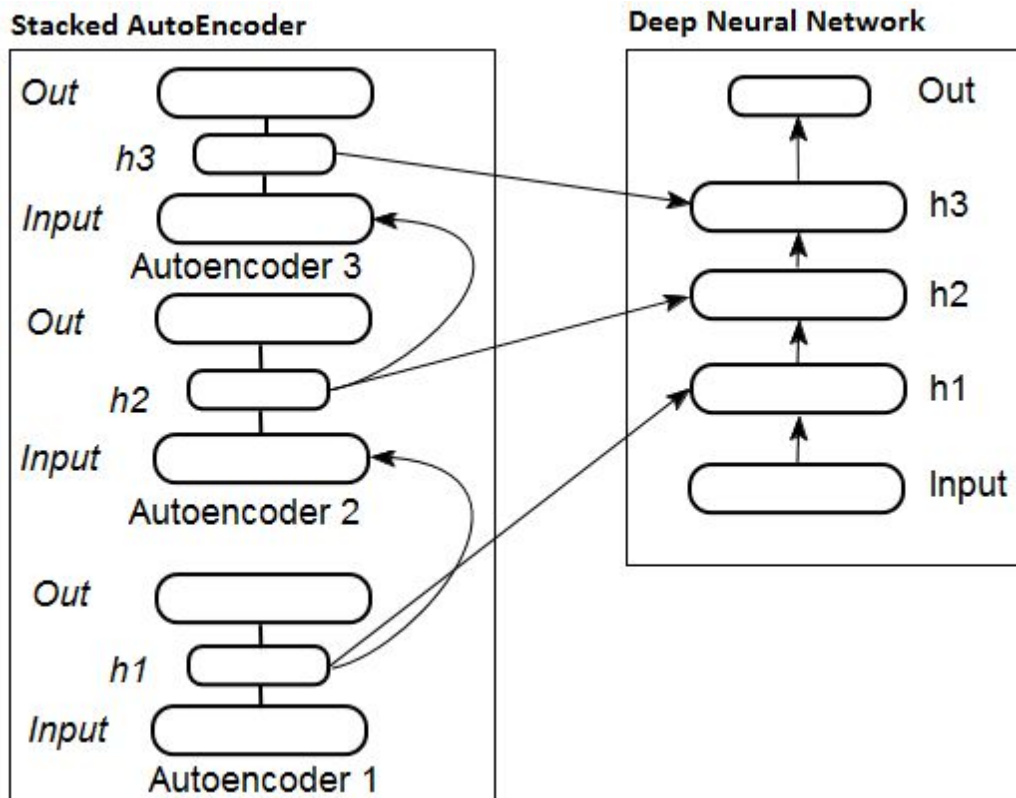
Como se puede apreciar, los diferentes patrones representan características fundamentales de los números tales como curvas, rectas, círculos, etc.

En este punto, ya solo nos interesa la primera mitad de la red (que realiza la codificación) que nos servirá como capa de entrada para otra red neuronal.

### 2.1.3.- Red profunda mediante auto-codificadores

Un solo auto-codificador puede encontrar características fundamentales en la información de entrada, las características más primitivas y simples que se pueden extraer de esa información, como rectas y curvas en el caso de las imágenes. Sin embargo, si queremos que nuestras máquinas detecten conceptos más complejos como rostros, nos hace falta más potencia.

Fijémonos en la operación que realiza un auto-codificador en su capa oculta. A partir de información cruda sin significado (por ejemplo, píxeles de imágenes), es capaz de etiquetar características algo más complejas (por ejemplo, formas simples presentes en cualquier imagen como líneas y curvas). Entonces la pregunta es, ¿qué pasa si al resultado codificado, en esa capa oculta, le aplicamos otro auto-codificador? Si lo hacemos bien, encontrará características más complejas todavía (como círculos, arcos, ángulos rectos, etc). Si continuamos haciendo esto varias veces, tendremos una jerarquía de características cada vez más complejas, junto con una pila de codificadores. Siguiendo el ejemplo de las imágenes, dada una profundidad suficiente e imágenes de ejemplo suficientes, conseguiremos alguna neurona que se active cuando la imagen tenga un rostro, y sin necesidad de que ningún supervisor le explique a la red cómo es un rostro.



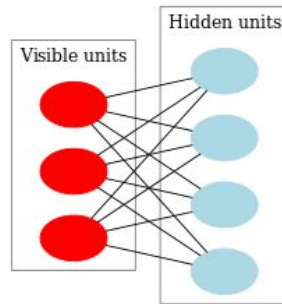
Red neuronal profunda construida con auto-codificadores

La idea de Deep Learning mediante auto-codificadores apilados es precisamente esa, usar varios codificadores, y entrenarlos uno a uno, usando cada codificador entrenado para entrenar el siguiente. Podríamos llamarlo un algoritmo voraz, y éste es realmente el gran avance del Deep Learning.

#### 2.1.4.- Máquinas de Boltzmann restringidas (Restricted Boltzmann Machines)

Una máquina de Boltzmann restringida o RBM es una red neuronal artificial que utiliza un método estocástico generativo para aprender una distribución de probabilidad sobre un conjunto de datos de entrada.

Este tipo de redes surgieron debido al uso de redes de conexión de retorno, las cuales son muy difíciles de entrenar. A causa de esto se empezaron a utilizar modelos recurrentes más limitados para adoptar algoritmos de aprendizaje más sencillos. Uno de estos modelos fueron las redes de Hopfield, a continuación aparecieron las máquinas de Boltzmann simples que eran de naturaleza estocástica y tenían las neuronas divididas en dos grupos, visibles y ocultas.



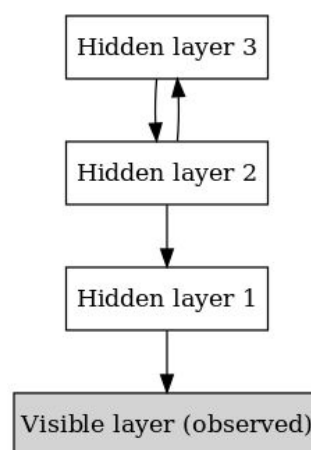
Una máquina de Boltzmann restringida se caracteriza por la ausencia de conexiones entre las neuronas de una capa, al contrario que en la simple que si dispone de conexiones entre las neuronas de la misma capa.

En las RBM tenemos una serie de estados visibles (neuronas visibles) y otros ocultos (neuronas ocultas), se utilizan estas capas ocultas para realizar una probabilidad respecto a los estados visibles para obtener una conclusión. Después del entrenamiento utilizando la teoría de Bayes podemos obtener conclusiones de manera inversa, es decir, obtener conclusiones de los estados visibles mediante los ocultos.

El algoritmo de aprendizaje utilizado para entrenar este tipo de redes es la desviación comparativa o divergencia contrastiva (CD-k), el cual destaca por su sencillez. Únicamente trabajamos con las esperanzas, siendo estas sustituidas por valores totalmente determinados mediante el muestreo de Gibbs.

### 2.1.5.- Red de creencia profunda (Deep belief network)

Una red de creencia profunda es una alternativa a las redes neuronales profundas mencionadas anteriormente. Se basa en un modelo de grafos generadores, compuesto por múltiples capas ocultas, las cuales están conectadas con otras capas pero no están conectadas con unidades dentro de las mismas capas.



El entrenamiento del algoritmo se realiza de manera no supervisada, de manera que pueda aprender de manera probabilística a reconstruir las entradas. Cada capa actúa como identificador de una única característica de dicha entrada. Después del entrenamiento no supervisado se suele realizar un entrenamiento supervisado para mejorar los resultados de la clasificación.

Este tipo de red puede verse como una red neuronal simple no supervisada, donde cada capa oculta de un nivel dado sirve como capa visible al siguiente nivel, de esta manera conseguimos un entrenamiento más eficiente y con menos posibilidades de sobreentrenamiento.

Como se ha dicho anteriormente, es una alternativa a las redes neuronales profundas, por los que los usos son similares a estas.

### **3.- Conclusiones**

Con los conocimientos básicos que tenemos sobre machine learning nos parece un campo muy interesante, así pues ocurre lo mismo con el deep learning. Si se presta atención, se puede observar como el deep learning está hoy en día presente en muchos sitios, y es algo que sigue a la alza. Muchas empresas o instituciones buscan profesionales en este campo, como no puede ser de otra manera debido a la época de la información en la que vivimos, donde aplicando las técnicas adecuadas sobre un conjunto de datos nos pueden dar respuesta a muchas inquietudes, desde la obtención de preferencias de un individuo en cuanto a publicidad hasta detección precoz de enfermedades o crímenes, pasando por el campo de la robótica y otros muchos.



## 4.- Bibliografía

- [https://en.wikipedia.org/wiki/Deep\\_learning](https://en.wikipedia.org/wiki/Deep_learning) [última rev. 18 marzo 2016]
- <https://rubenlopezg.wordpress.com/2014/05/07/que-es-y-como-funciona-deep-learning/> [última rev. 18 marzo 2016]
- <http://www.sorayapaniagua.com/2015/10/22/inteligencia-artificial-de-machine-learning-al-inquietante-deep-learning/> [última rev. 18 marzo 2016]
- <http://recode.net/2015/07/15/deep-learning-ai-is-taking-over-tech-what-is-it/> [última rev. 18 marzo 2016]
- [https://en.wikipedia.org/wiki/Deep\\_belief\\_network](https://en.wikipedia.org/wiki/Deep_belief_network) [última rev. 20 marzo 2016]
- <https://rubenlopezg.wordpress.com/2014/05/07/que-es-y-como-funciona-deep-learning/> [última rev. 19 marzo 2016]
- <http://ufldl.stanford.edu/tutorial/unsupervised/Autoencoders/> [última rev. 19 marzo 2016]
- <http://neuralnetworksanddeeplearning.com/> [última rev. 21 marzo 2016]
- <http://deeplearning.net/tutorial/dA.html#daa> [última rev. 21 marzo 2016]
- <http://deeplearning.net/tutorial/SdA.html#sda> [última rev. 20 marzo 2016]
- [https://www.mql5.com/es/articles/1103#ch\\_1](https://www.mql5.com/es/articles/1103#ch_1) [última rev. 20 marzo 2016]