

Aprendizaje Profundo

1 Introducción al aprendizaje profundo

Prof. Rodrigo López Farias

Querétaro

CONAHCYT-CentroGeo

Introducción al aprendizaje profundo

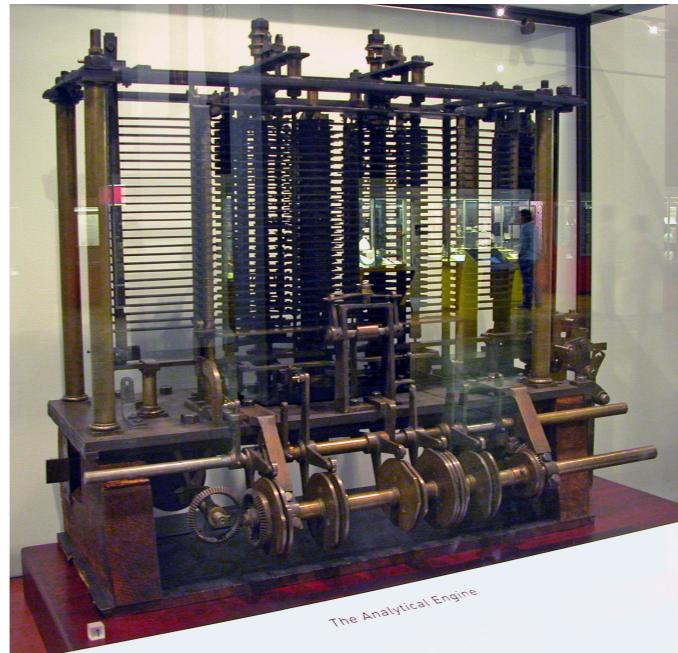
Desde la aparición de los primeros diseños de computadoras, la gente se preguntó si podían estas imitar la inteligencia.

En los primeros años la inteligencia artificial (IA), resolvió los primero problemas intelectualmente complicados para los humanos pero fáciles para las computadoras.

Se intentó crear inteligencia artificial con un enfoque de utilizar un conocimiento base “dura” con reglas explícitas de inferencia.

Se usó en ambientes controlados.

Incapaz de adquirir y usar nuevo conocimiento mas allá de la jugada del contrincante. El sistema no se automejora.



Máquina analítica de Charles Babbage, 1935.
Fuente: Wikipedia



Deep Blue venciendo a Gary Kasparov en 1997
En un juego de ajedrez.

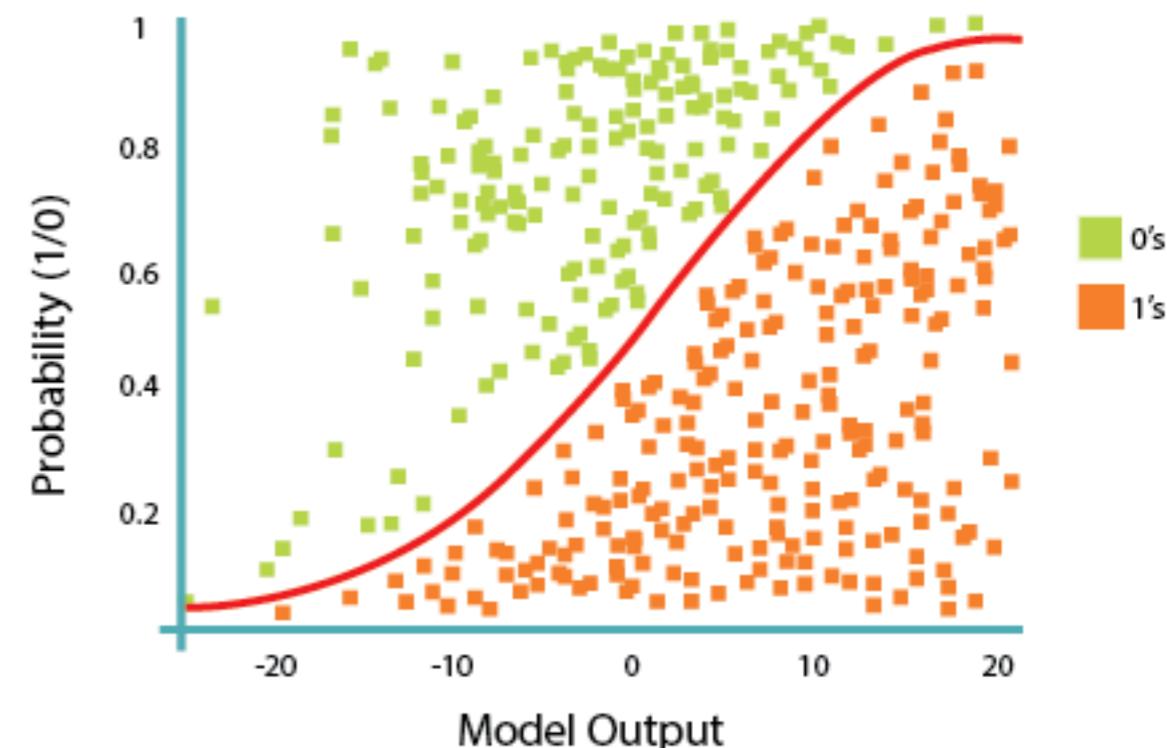
Introducción al aprendizaje profundo

La capacidad de auto aprender de las computadoras se le llama en inglés “Machine Learning”. en español es:

- *Aprendizaje Automático (AA),
- *aprendizaje de máquinas,
- *aprendizaje de computadoras

AA pudo abordar problemas de decisión aparentemente subjetivos

E.g., En 1990 recomendación por cesárea utilizando una regresión logística.



Introducción al aprendizaje profundo

El truco para que funcionen estos modelos reside en **una buena representación de los datos** que incluye una **selección de características** cuidadosa por un experto.

Una mala selección o representación de datos, genera malos algoritmos predictivos.

Una solución es utilizar AA para aprender la mejor representación de la información. A esto se le llama aprendizaje de representación (**Representation Learning**).

El ejemplo representativo de Aprendizaje de representación, es el Autoencoder.

Introducción al aprendizaje profundo

El verdadero reto de la inteligencia artificial es resolver tareas que son “fáciles” o naturales de ejecutar para los humanos pero difíciles de describir formalmente, como el reconocimiento de palabras, imágenes, sentimientos, entendimiento de patrones etc...

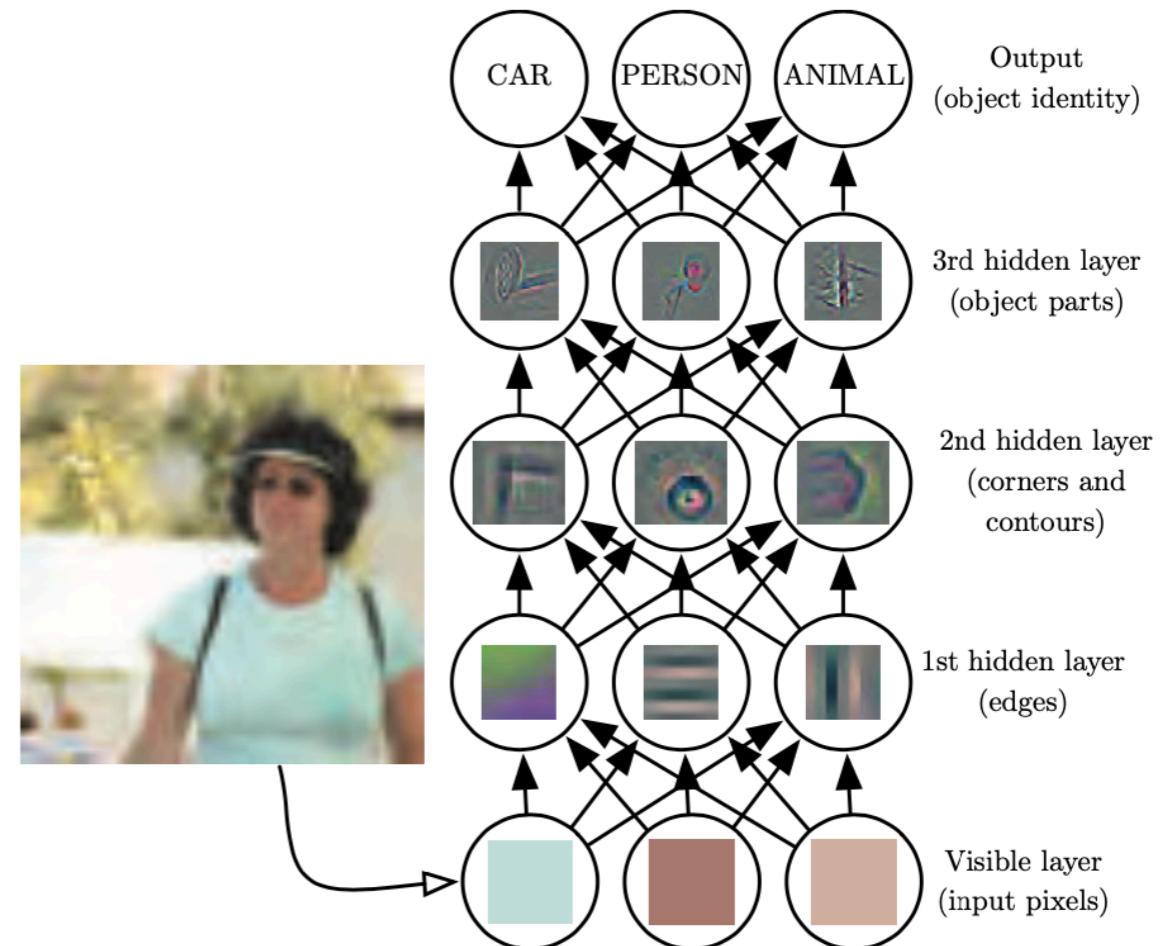
Ejemplos de **problemas o tareas que los humanos ejecutan de manera natural** pero son difíciles para las computadoras.

- Entender el lenguaje natural.
- Inferir el sentimiento de las personas.
- Traducción Automática
- Clasificación de Imágenes u otro tipo de objetos
- Reconocimiento de objetos.
- **Toma de decisiones**
- Qué otros problemas los humanos pueden resolver fácilmente?

Introducción al aprendizaje profundo

El Aprendizaje Automático es un conjunto de algoritmos enfocadas a resolver estos problemas **humanamente intuitivos** permitiendo a las computadoras imitando el aprendizaje, la experiencia y entendimiento el mundo como una **Jerarquía de Conceptos**.

Permite liberar al experto humano de describir y definir formalmente (o de manera directa) la información exacta que las computadoras necesitan para aprender.



La jerarquía de conceptos permite a la computadora aprender conceptos complicados construyéndolos considerando aquellos más simples.

Introducción al aprendizaje profundo

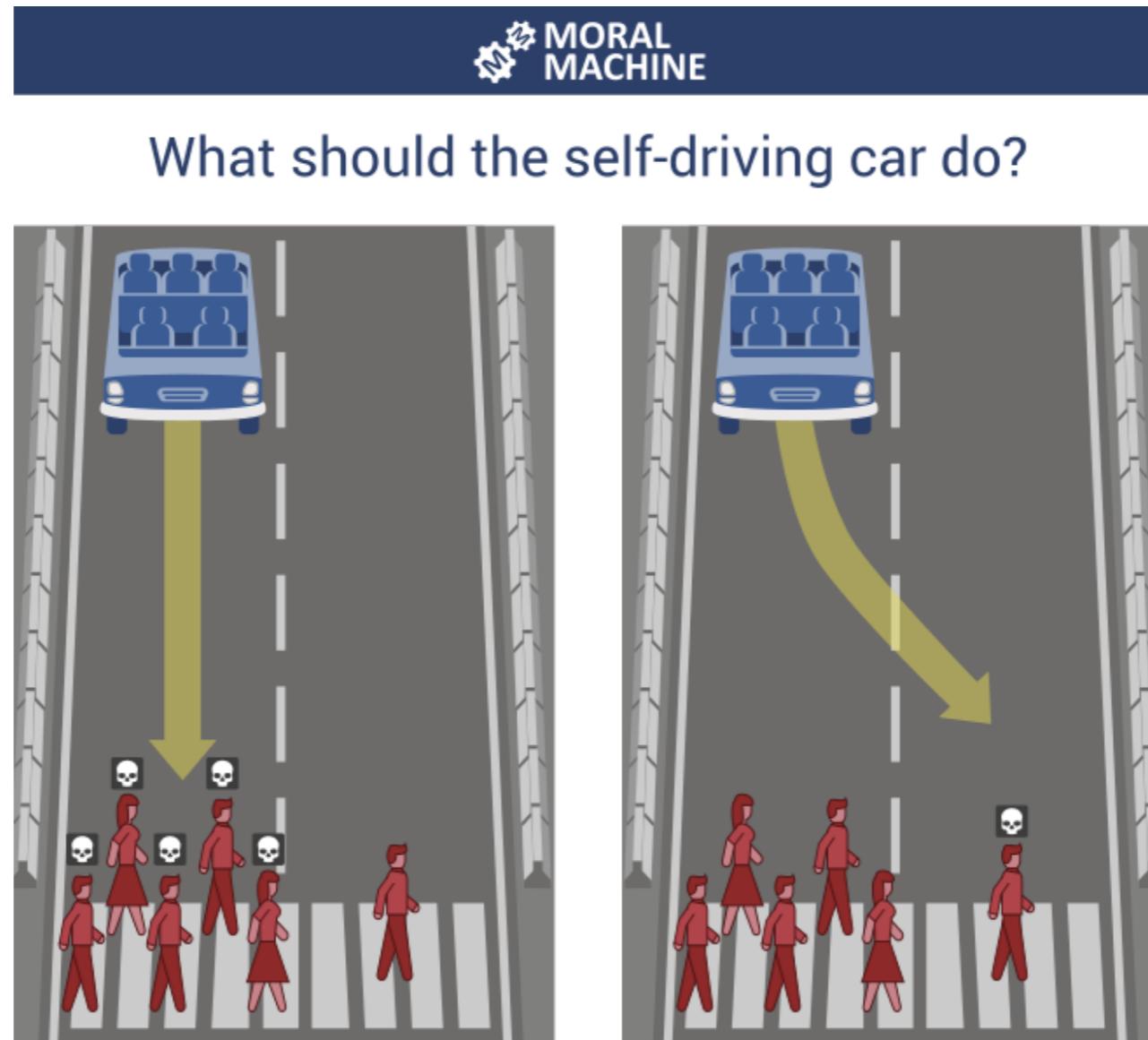
Para que las computadoras puedan comportarse de manera humanamente Inteligente, se deben crear algoritmos que imiten la manera en que los humanos adquieren y perciben esta gran cantidad de información sobre el mundo real para adaptar su comportamiento.

Toda esta información que percibimos y procesamos es, abstracta, subjetiva, no estructurada e informal, muy complicada de articular formalmente.

Hay otros problemas que llevan al límite la subjetividad y son problemas de decisión difíciles para las computadoras y para los humanos. La cuestión es si los humanos queremos delegar a una computadora la toma de decisiones éticas.

Introducción al aprendizaje profundo

- Decisiones Éticas



Ejemplo de un problema abordada por la Máquina Moral <https://www.technologyreview.com/2018/10/24/139313/a-global-ethics-study-aims-to-help-ai-solve-the-self-driving-trolley-problem/>

Redes neuronales como modelo de implementación de aprendizaje Profundo

Las 3 Olas del Aprendizaje Profundo

El Aprendizaje profundo se refiere al estudio de modelos con estructura jerárquica (multinivel) orientados a resolver problemas de aprendizaje automático independientemente si son inspirados en neurociencia o no.

El desarrollo histórico de lo que conocemos como Aprendizaje Profundo esta dado por 3 Olas.

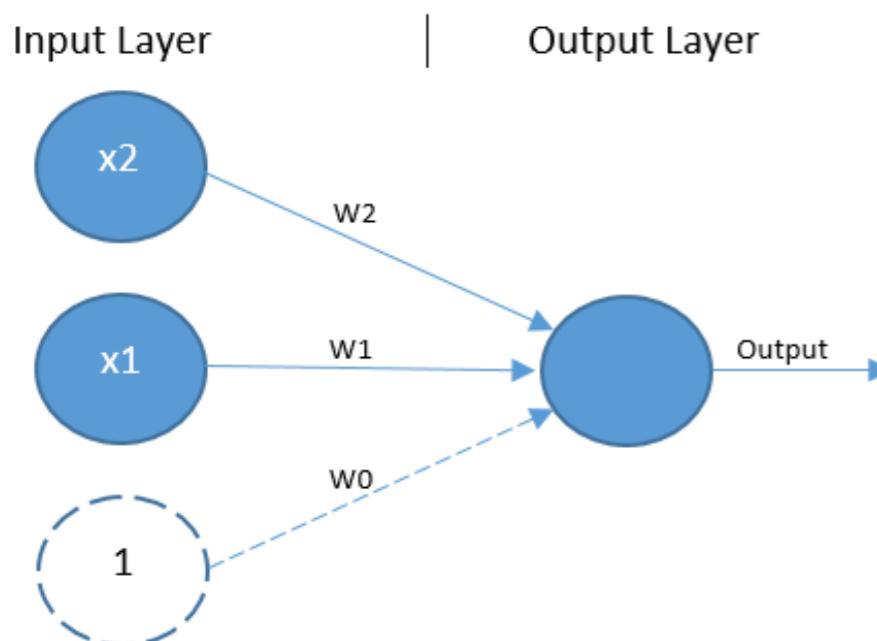
1. Ola 1, 1940-1960s: Cibernética.
2. Ola 2, 1980s-1990s: Conexiónismo.
3. Ola 3, 2006: Aprendizaje Profundo.

Redes neuronales como modelo de implementación de aprendizaje Profundo

Primera Ola

Cibernética (1940-1960, Rosenblatt 1958). Impulsado por la teoría de aprendizaje Biológico.

Nace el Modelo perceptrón, primer modelo de una sola neurona **capaz de aprender de los datos.**



El algoritmo de aprendizaje del perceptrón consiste en:

1. Definir el conjunto de entrenamiento e identificar las variables dependientes (etiquetas) e independientes. ($y=f(x)$)
2. Inicializar el valor de un conjunto de pesos (parámetros) \mathbf{w} .
3. Por cada muestra en el conjunto de entrenamiento clasificada por el modelo, evaluar si la salida del perceptrón coincide con la clase real .
 1. Si la salida es incorrecta se reajustan los pesos \mathbf{w} , si no, se dejan como están.

Redes neuronales como modelo de implementación de aprendizaje Profundo

TABLE 1.1 Summary of the Perceptron Convergence Algorithm

Variables and Parameters:

$$\mathbf{x}(n) = (m + 1)\text{-by-1 input vector} \\ = [+1, x_1(n), x_2(n), \dots, x_m(n)]^T$$

$$\mathbf{w}(n) = (m + 1)\text{-by-1 weight vector} \\ = [b, w_1(n), w_2(n), \dots, w_m(n)]^T$$

b = bias

$y(n)$ = actual response (quantized)

$d(n)$ = desired response

η = learning-rate parameter, a positive constant less than unity

1. *Initialization.* Set $\mathbf{w}(0) = \mathbf{0}$. Then perform the following computations for time-step $n = 1, 2, \dots$
2. *Activation.* At time-step n , activate the perceptron by applying continuous-valued input vector $\mathbf{x}(n)$ and desired response $d(n)$.
3. *Computation of Actual Response.* Compute the actual response of the perceptron as

$$y(n) = \text{sgn}[\mathbf{w}^T(n)\mathbf{x}(n)]$$

where $\text{sgn}(\cdot)$ is the signum function.

4. *Adaptation of Weight Vector.* Update the weight vector of the perceptron to obtain

$$\mathbf{w}(n + 1) = \mathbf{w}(n) + \eta[d(n) - y(n)]\mathbf{x}(n)$$

where

$$d(n) = \begin{cases} +1 & \text{if } \mathbf{x}(n) \text{ belongs to class } \ell_1 \\ -1 & \text{if } \mathbf{x}(n) \text{ belongs to class } \ell_2 \end{cases}$$

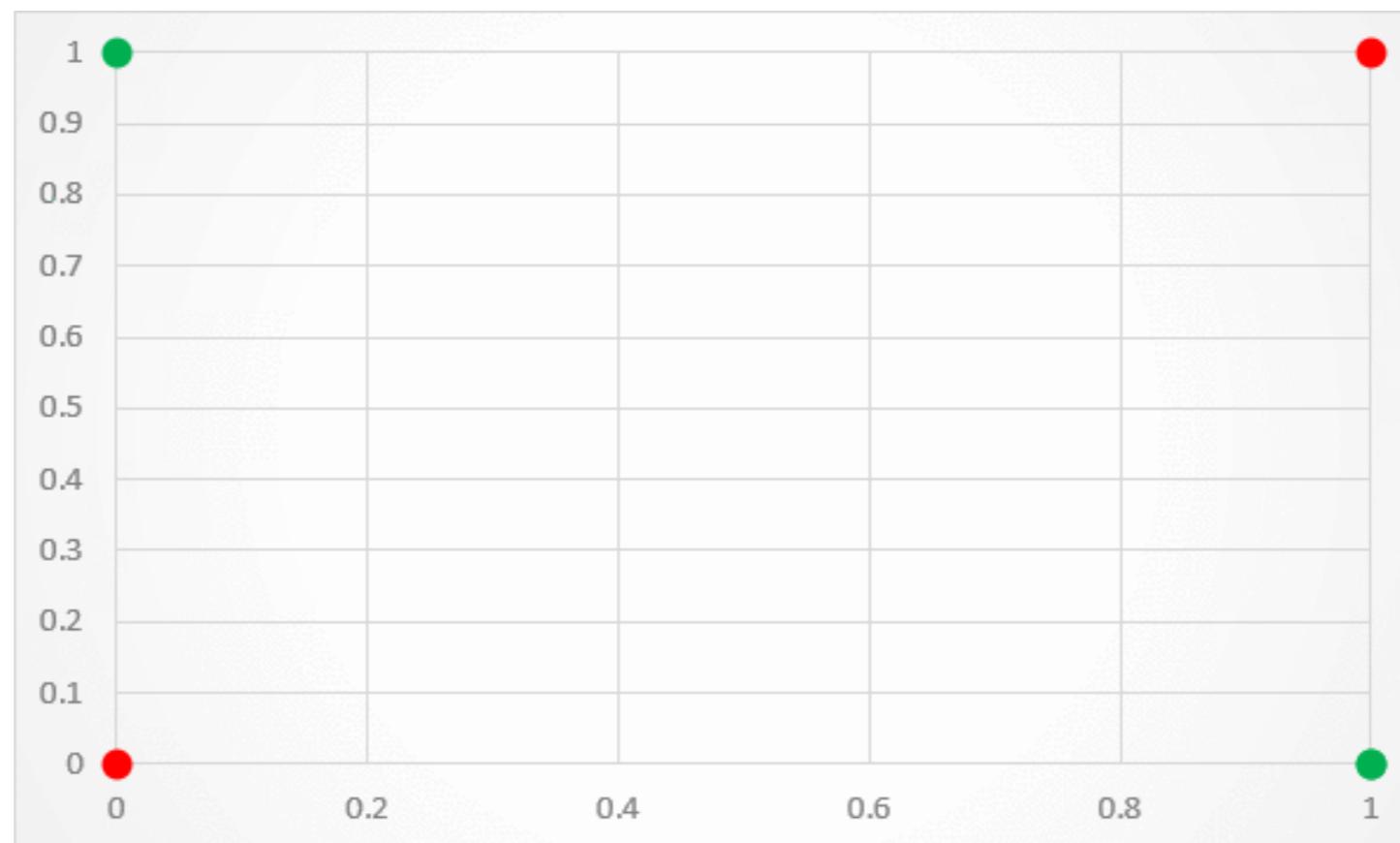
5. *Continuation.* Increment time step n by one and go back to step 2.

[Haykin: “Neural Networks and Learning Machines”, 3rd edition]

Redes neuronales como modelo de implementación de aprendizaje Profundo

(Minsky and Papert, 1969) Estos modelos lineales tienen limitaciones.

La limitación que desanimó el estudio de estos modelos es la incapacidad de resolver la clasificación de clases definidas por del operador binario lógico XOR, las cuales son no linealmente separables.



Redes neuronales como modelo de implementación de aprendizaje Profundo

Tabla de verdad de la operación lógica XOR

```
a XOR b  
=====0 xor 0 = 0  
0 xor 1 = 1  
1 xor 0 = 1  
1 xor 1 = 0
```

El problema del operador XOR es que es un operador compuesto dada por la tabla de verdad:

$a \text{ XOR } b = (a \text{ AND NOT } b) \text{ or } (b \text{ AND NOT } a)$

Primero tenemos que resolver $(a \text{ AND NOT } b)$

$(a \text{ AND NOT } b)$

a	b	$a \text{ AND NOT } b$
0	0	0
0	1	0
1	1	0
1	0	1

Después tenemos que resolver $(b \text{ AND NOT } a)$

a	b	$b \text{ AND NOT } a$
0	0	0
0	1	1
1	1	0
1	0	0

$(a \text{ or } b)$

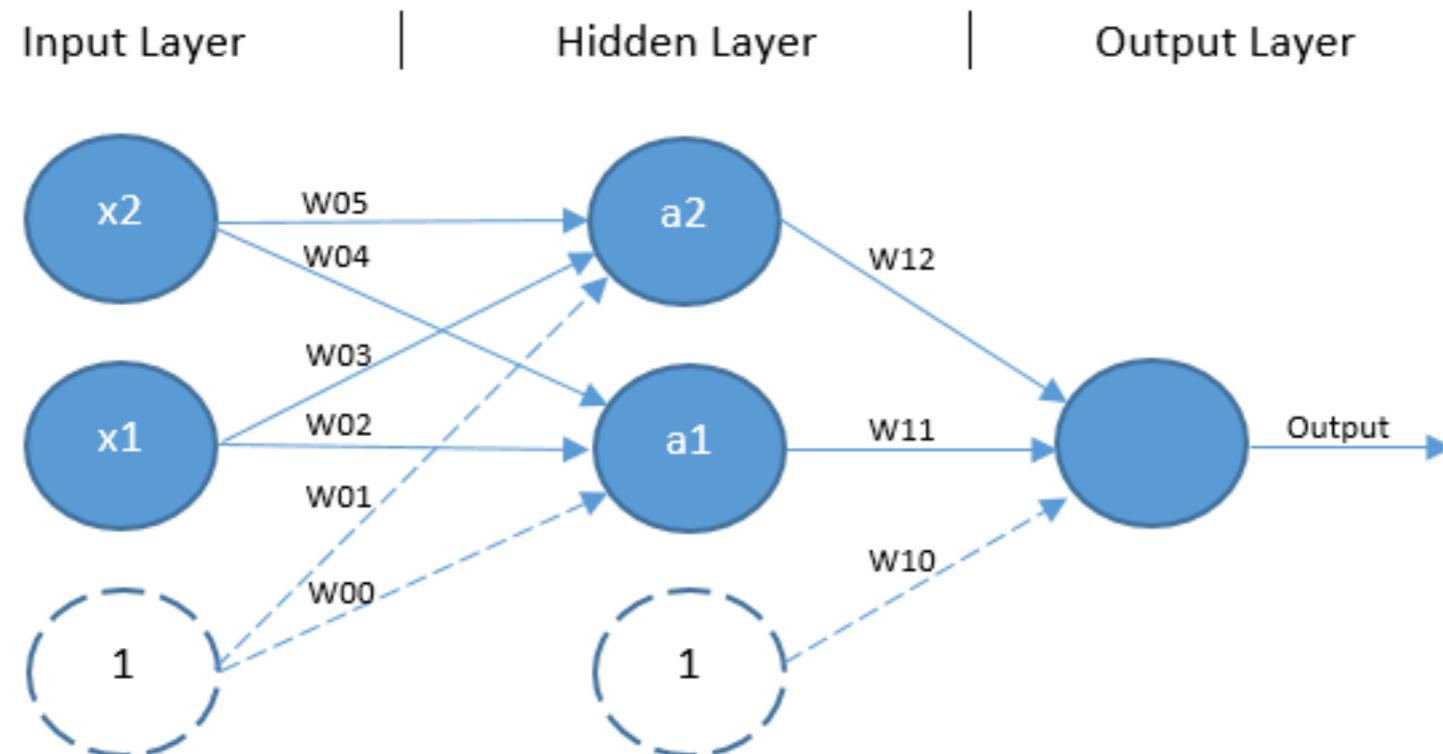
$a \mid b \mid a \text{ or } b$

a	b	$a \text{ or } b$
0	0	0
0	1	1
1	0	1
1	0	0

Redes neuronales como modelo de implementación de aprendizaje Profundo

Segunda Ola

Conexiónismo: (1980-1995, Rumelhart) Impulsado por una red neuronal con una o dos capas ocultas con funciones de activación no lineales que resuelve el problema del XOR.



Redes neuronales como modelo de implementación de aprendizaje Profundo

la idea principal es que las unidades de procesamiento simples (neuronas) interactuando entre ellas, genera un comportamiento complejo e inteligente. Se tiene éxito con funciones de activación sigmoidales.

En esta ola se introduce el concepto de **Representación Distribuida** (Hinton et al., 1986) y el **algoritmo de Propagación Hacia atrás**.

Se implementa el **algoritmo de descenso de gradiente** para el entrenamiento de estas redes. Este algoritmo depende de la **derivada** para decidir “hacia dónde mover” los pesos w.

Su éxito se debe al algoritmo de propagación hacia atrás resuelve el **cálculo automático de la derivada parcial** para en un tipo de redes neuronales llamadas de propagación hacia adelante.

Redes neuronales como modelo de implementación de aprendizaje Profundo

Problemas:

Este esquema no escala con el tamaño de las redes neuronales de propagación hacia adelante y gran cantidad de datos.

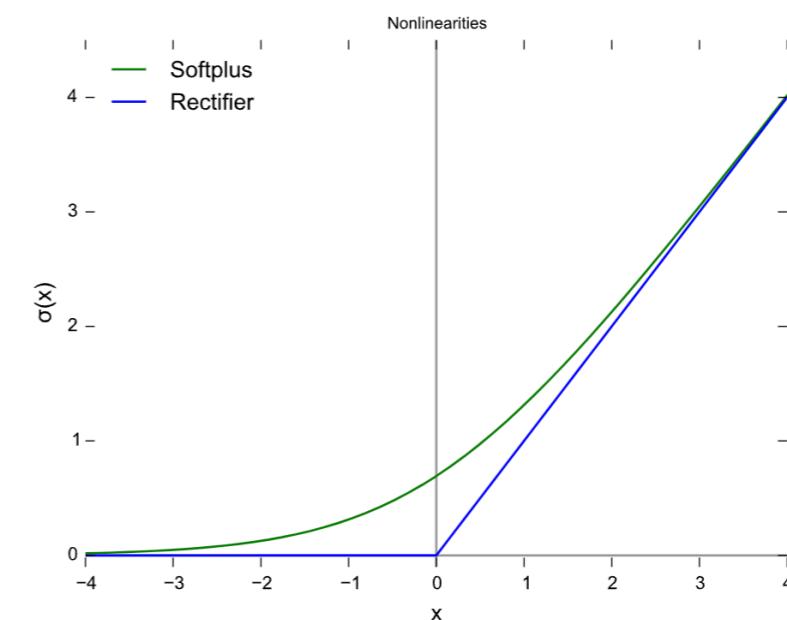
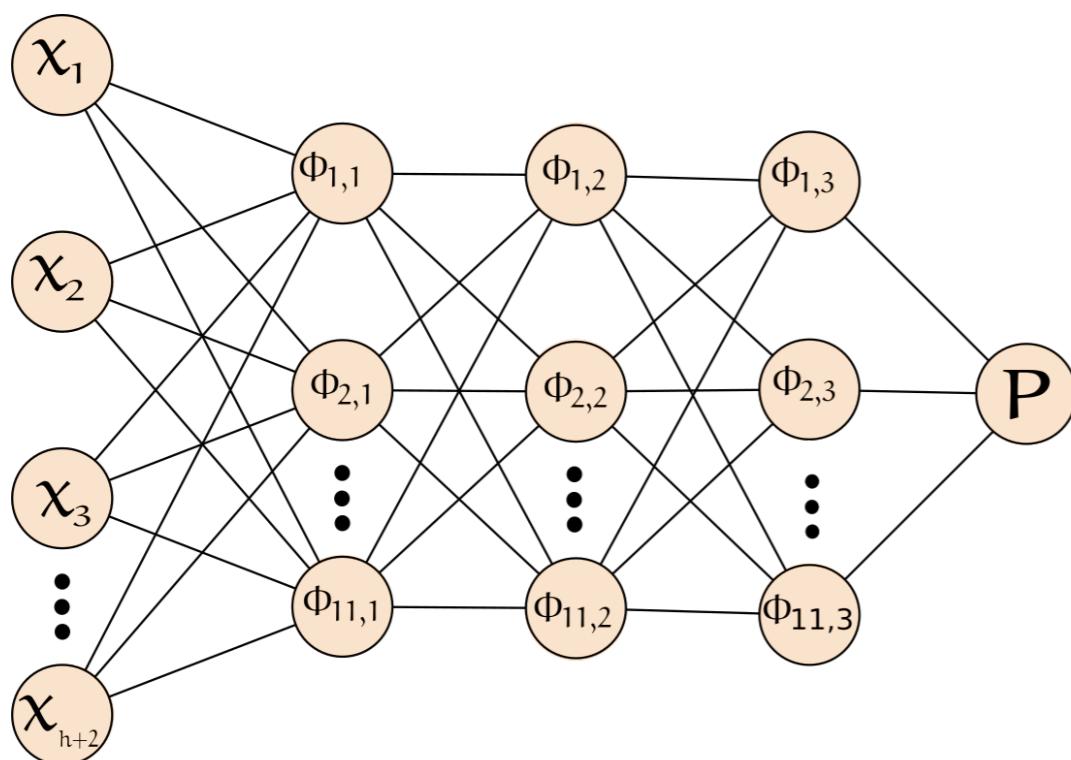
La segunda ola termina con Hochreiter (1991) y Bengio et al. (1994). Ellos Identificaron otro problema con el aprendizaje profundo que es el modelado de secuencias muy largas.

Redes neuronales como modelo de implementación de aprendizaje Profundo

Tercera Ola

Fué impulsado al encontrar la solución del problema de entrenar redes neuronales multicapa con muchos pesos y muchos datos. La arquitectura de las redes neuronales ahora pueden crecer y aprender de datos mas complejos.

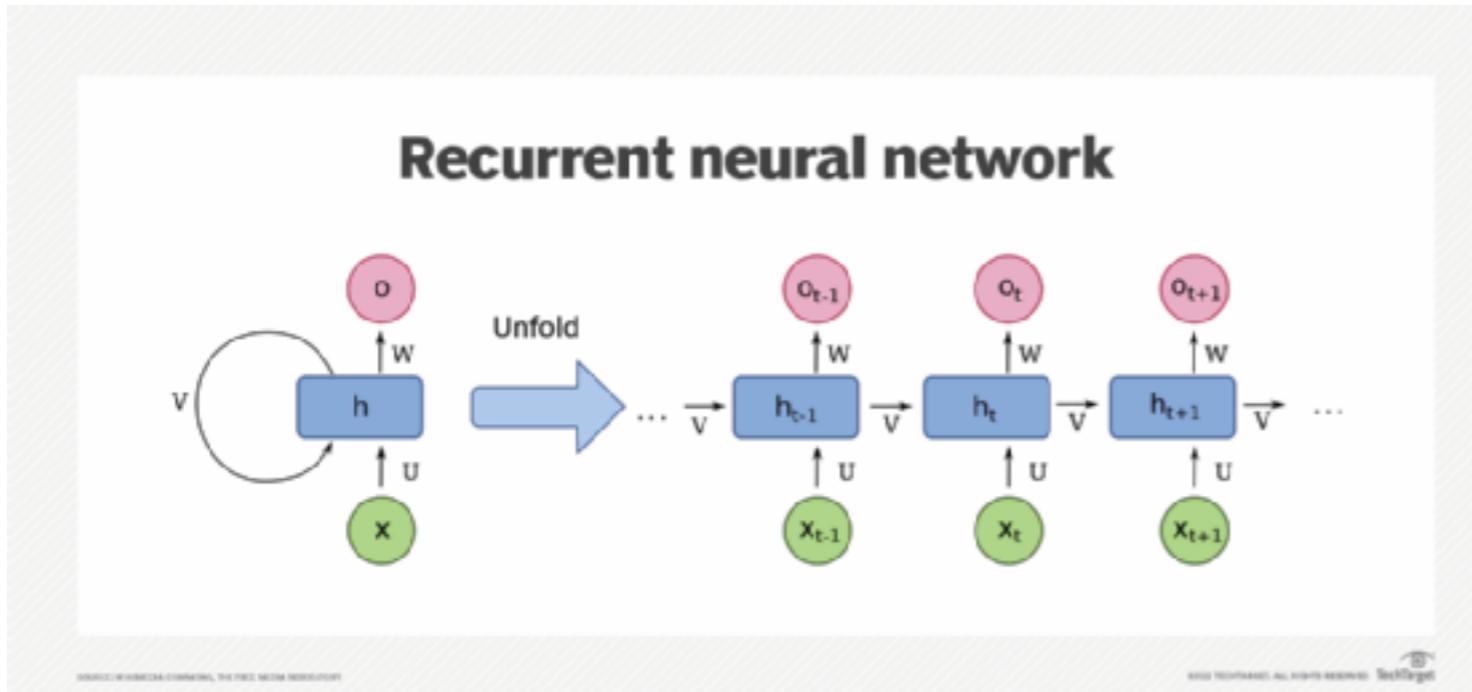
El algoritmo de descenso de gradiente estocástico logró entrenar redes neuronales mas grandes junto con funciones de activación relee



Redes neuronales como modelo de implementación de aprendizaje Profundo

Hochreiter (1991) y Bengio et al. (1994), Identificaron otro problema con el aprendizaje profundo en redes neuronales recurrentes.

Las redes neuronales recurrentes son un tipo de redes que se caracterizan por retroalimentarse de su propia salida. Son utilizadas para modelar por ejemplo, secuencia de palabras, series de tiempo.



Redes neuronales como modelo de implementación de aprendizaje Profundo

Las redes neuronales recurrentes en teoría deben ser buenas para aprender a recordar información a largo plazo, pero la persistencia no regulada, para recordar u olvidar información según se vaya adquiriendo, no ayuda en la práctica.

Este problema repercute en aplicaciones como:

1. Procesamiento de lenguaje natural
2. Series de tiempo (Sistemas dinámicos variantes en el tiempo)*
3. Clasificación dependiente del tiempo.
4. Reconocimiento de imágenes.

Para esto Hochreiter y Schmidhuber (1997) propusieron the long short-term memory (LSTM) para resolver estos problemas. LSTM es una red neuronal recurrente que incorpora una operación para definir el qué tanto importa recordar la información reciente y a largo plazo para hacer una Clasificación o regresión.

Por ejemplo,

LSTM es capaz de completar una frase coherentemente considerando el contexto como:

Hola soy Esteban, estudio Geoinformática en centrogeo.... y me gusta_____

- a) programar
- b) cabalgar
- c) estudiar
- d) caminar

Redes neuronales como modelo de implementación de aprendizaje Profundo

Motivación gráfica para el estudio de aprendizaje profundo

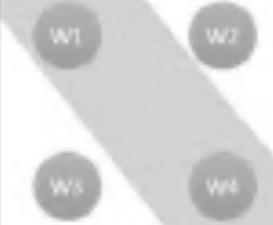
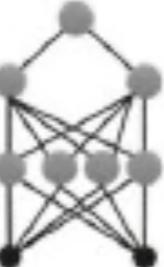
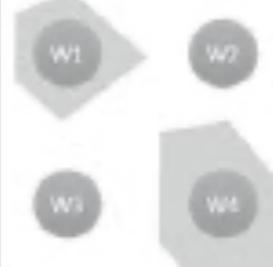
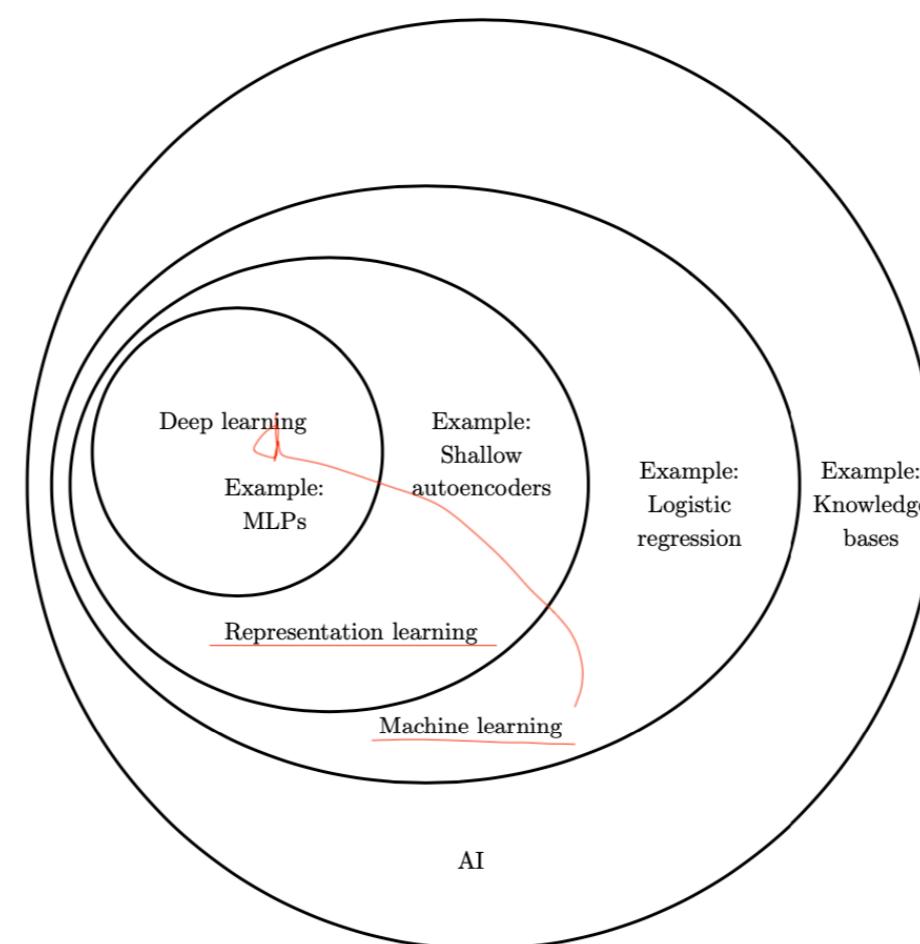
Estructura de la red	Tipo de región de decisión	Solución OR Exclusivo	clases con regiones geométricas	Formas de superficie de decisión mas generales
	Hiperplano simple			
	Regiones convexas abiertas o cerradas			
	Arbitrario (complejidad limitada por el numero de nodos)			

Diagrama de Venn de las subdisciplinas de la IA

Deep Learning es una subsdisciplina de la IA!

Deep Learning es un tipo de aprendizaje automático que, además de encontrar una función de mapeo para asociar un conjunto de características x con una salida datos y , internamente logran aprender como representar esos datos para mejorar el desempeño del modelo.



Técnicas modernas y la solución de problemas de AP, se nutren de muchas otras áreas, como:

- Álgebra Lineal.
- Probabilidad.
- Teoría de la información.
- Optimización Numérica.

