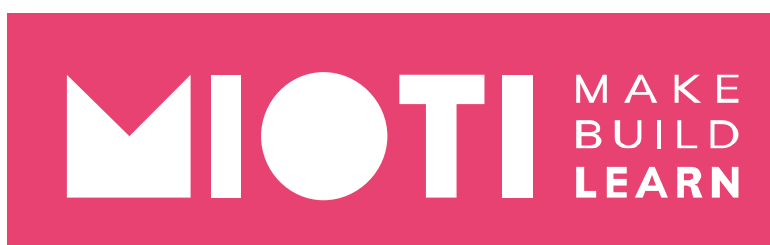




2021

# Deep Learning

RUBÉN ZAZO  
[rubenzc@faculty.mioti.es](mailto:rubenzc@faculty.mioti.es)







# Objetivos

- Introducción a las redes neuronales profundas
- Conocer los principales algoritmos:
  - Redes neuronales profundas (DNNs)
  - Redes neuronales convolucionales (CNNs)
  - Redes recurrentes (RNNs y LSTMs)
  - Reconocer en qué problemas pueden aplicarse y sus diferencias
- Aprender a entrenar, evaluar y aplicar redes neuronales profundas



# Sesiones

<b>Sesión 1</b>	Introducción a las redes neuronales profundas
<b>Sesión 2</b>	Redes neuronales en TensorFlow: DNNs – Fashion MNIST
<b>Sesión 3</b>	Redes neuronales profundas en TensorFlow: DNNs – Fashion MNIST
<b>Sesión 4</b>	Redes neuronales convolucionales en TensorFlow: CNNs – MNIST
<b>Sesión 5</b>	Redes neuronales convolucionales en TensorFlow: CNNs – CIFAR
<b>Sesión 6</b>	Redes neuronales recurrentes: RNNs y LSTMs
<b>Sesión 7</b>	Redes neuronales recurrentes en TensorFlow: LSTM – IMDB
<b>Sesión 8</b>	Conceptos avanzados de Deep Learning



# Challenges

- Tareas:
  - Implementar redes neuronales desde 0
  - Reconocer prendas de ropa a partir de imágenes en escala de grises
  - Reconocer objetos en imágenes
  - Reconocer secuencias – IMDB
- Redes:
  - DNNs, CNNs y LSTMs
- Herramientas (librerías):
  - TensorFlow y Keras



# Metodología

- MIOTI
  - Worksheets guiados
    - En casa
    - En clase
  - Challenge en clase
- A entregar como tarde el sábado de la semana cuando se haya presentado.



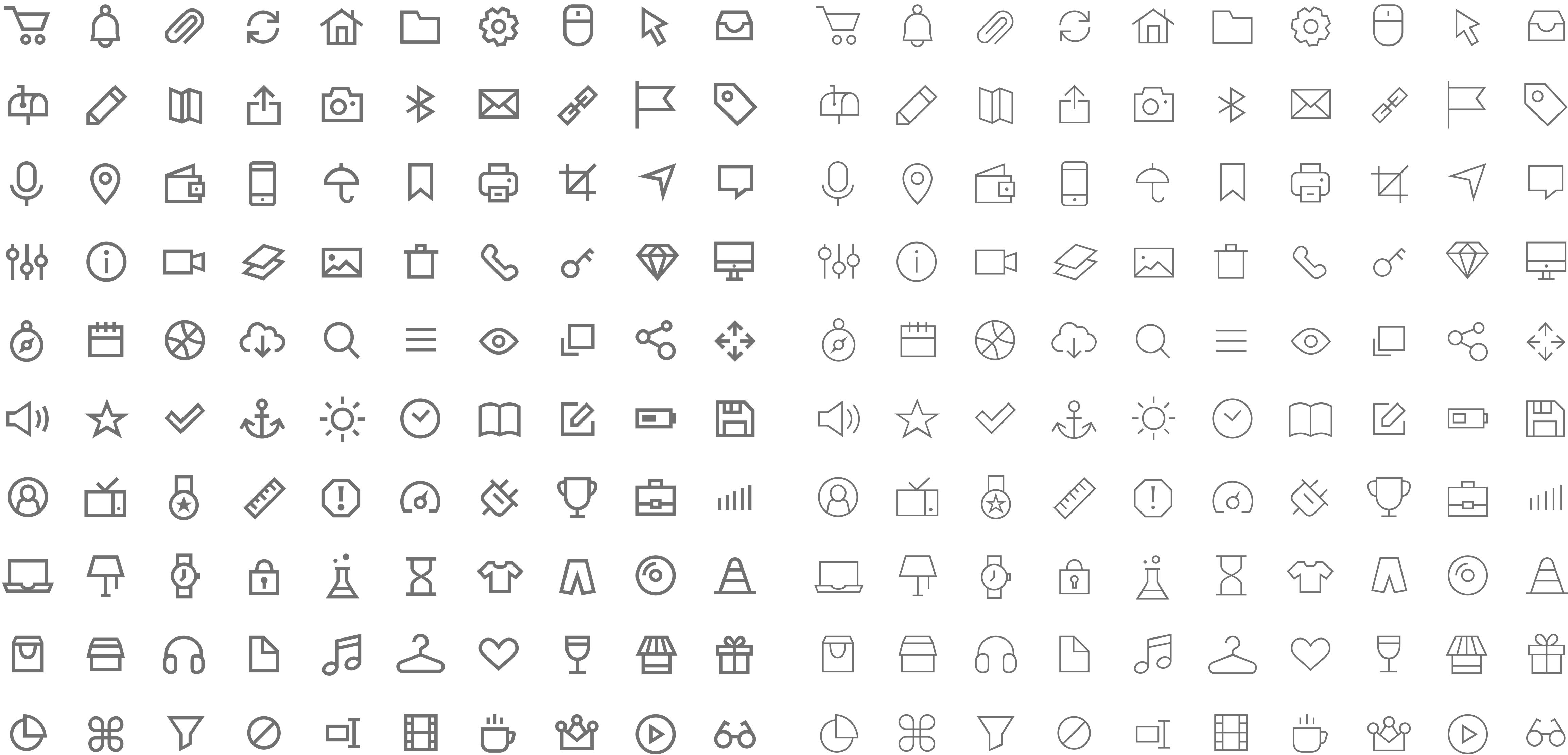
# Evaluación

- Valor ponderado:
  - 50% entrega de los challenges
  - 50% participación en clase

# Profesor: Rubén Zazo



- Background:
  - Ingeniero Superior en Telecomunicaciones (UAM)
  - **PhD**: Exploiting Temporal Context in Speech Technologies using LSTMs
- Estancias investigadoras
  - 2015. Google Inc: Speech Research Team.
    - Redes neuronales profundas en el procesamiento de voz
  - 2016. JHU & MIT
    - Estimación de edad a partir de la señal de voz utilizando redes neuronales recurrentes
  - 2017. Amazon: Alexa team
    - Mejora del asistente personal Alexa mediante la utilización de redes neuronales profundas





# Introducción

- Machine Learning



# Introducción

- Redes neuronales:
  - Inspiradas en el funcionamiento del cerebro: Un solo algoritmo capaz de aprender muchos problemas distintos
  - Están detrás de prácticamente todos los grandes avances en machine learning de la última década
  - Nacen en los años 60:
    - La publicación mas referenciada es del año 1969
    - El algoritmo back-propagation es del 86
- Redes neuronales **profundas**, ¿qué cambia?



# Redes Neuronales Profundas - Reconocimiento de Objetos



COCHES



NO COCHES

TEST: ¿QUÉ ES ESTO?





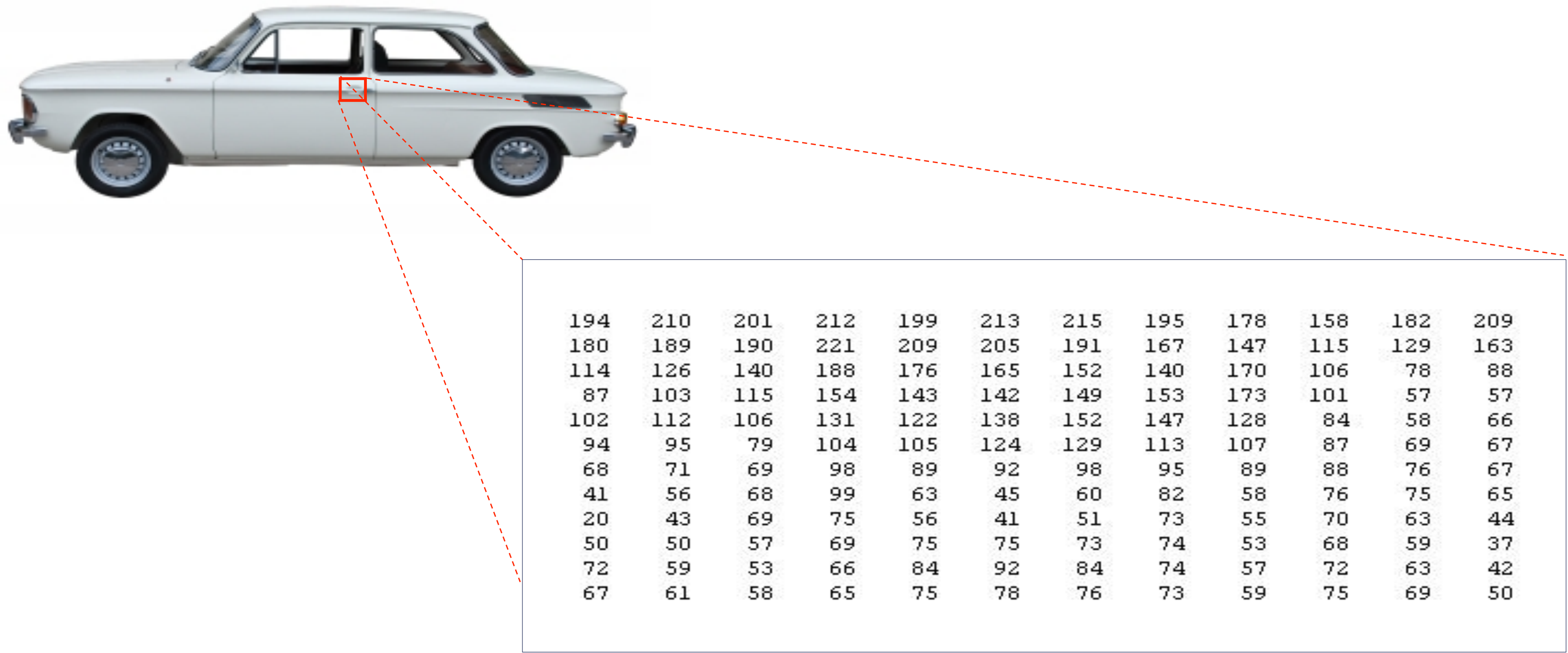
# Redes Neuronales Profundas - Reconocimiento de Objetos

**TÚ VES ESTO**



# Redes Neuronales Profundas - Reconocimiento de Objetos

TÚ VES ESTO



PERO LA CÁMARA VE ESTO

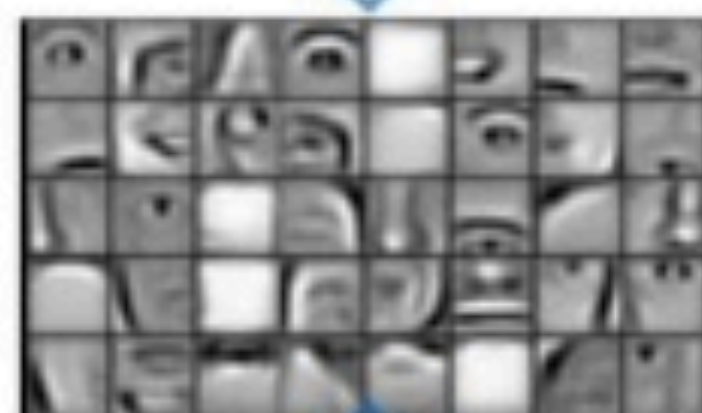
# Redes Neuronales Profundas - Reconocimiento de Objetos

- Reconocimiento de cara:

Feature representation



3rd layer  
"Objects"



2nd layer  
"Object parts"



1st layer  
"Edges"



Pixels

- A la entrada de la red tenemos píxeles
- La primera capa extrae aristas o vértices
- La segunda capa identifica partes de objetos
- La profundidad depende del problema



# Redes Neuronales Profundas

- Redes neuronales profundas: ¿Por qué?

## Scientists See Promise in Deep-Learning Programs

From Page A1

seeing in the accuracy of these systems is very rare indeed."

Artificial intelligence researchers are acutely aware of the dangers of being overly optimistic. Their field has long been plagued by outbreaks of misplaced enthusiasm followed by equally striking declines.

In the 1960s, some computer scientists believed that a workable artificial intelligence system was just 10 years away. In the 1980s, a wave of commercial start-ups collapsed, leading to what some people called the "A.I. winter."

But recent achievements have impressed a wide spectrum of computer experts. In October, for example, a team of graduate students studying with the University of Toronto computer scientist Geoffrey E. Hinton won the top prize in a contest sponsored by Merck to design software to help find molecules that might lead to new drugs.

From a data set describing the chemical structure of 15 different molecules, the program devised a

that organizes data science competitions, including the Merck contest.

Advances in pattern recognition hold implications not just for drug development but for an array of applications, including marketing and law enforcement. With greater accuracy, for example, marketers can comb large databases of consumer behavior to get more precise information on buying habits. And improvements in facial recognition are likely to make surveillance technology cheaper and more commonplace.

Artificial neural networks, an idea going back to the 1950s, seek to mimic the way the brain absorbs information and learns from it. In recent decades, Dr. Hinton, 64 (a great-great-grandson of the 19th-century mathematician George Boole, whose work in logic is the foundation for modern digital computers), has pioneered powerful new techniques for helping the artificial networks recognize patterns.

Modern artificial neural networks are composed of an array of software components divided



KEITH FRANKLIN

A student team led by the computer scientist Geoffrey E. Hinton used deep-learning technology to design software.

An advance in a technology that can best human brains.

lish on a large screen above his head.

Then, in a demonstration that led to stunned applause, he paused after each sentence and the words were translated into Mandarin Chinese characters, accompanied by a simulation of his own voice in that language, which Dr. Rashid has never spoken.

The feat was made possible, in part, by deep-learning techniques that have spurred improvements in the accuracy of speech recognition.

Dr. Rashid, who oversees Microsoft's worldwide research organization, acknowledged that while his company's new speech recognition software made 30 percent fewer errors than previous models, it was "still far from perfect."

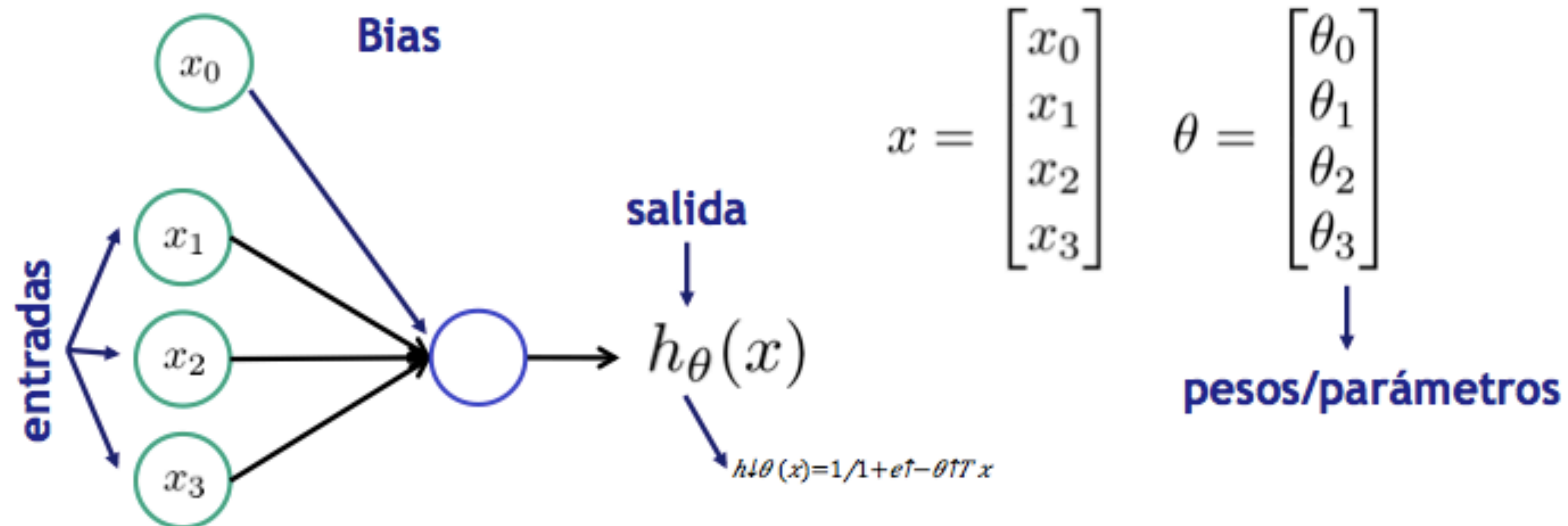
"Rather than having one word in four or five incorrect, now the error rate is one word in seven or eight," he wrote on Microsoft's Web site. Still, he added that this was "the most dramatic change in accuracy" since 1979, "and as we add more data to the training

... Gracias a tener más datos y capacidad de cómputo

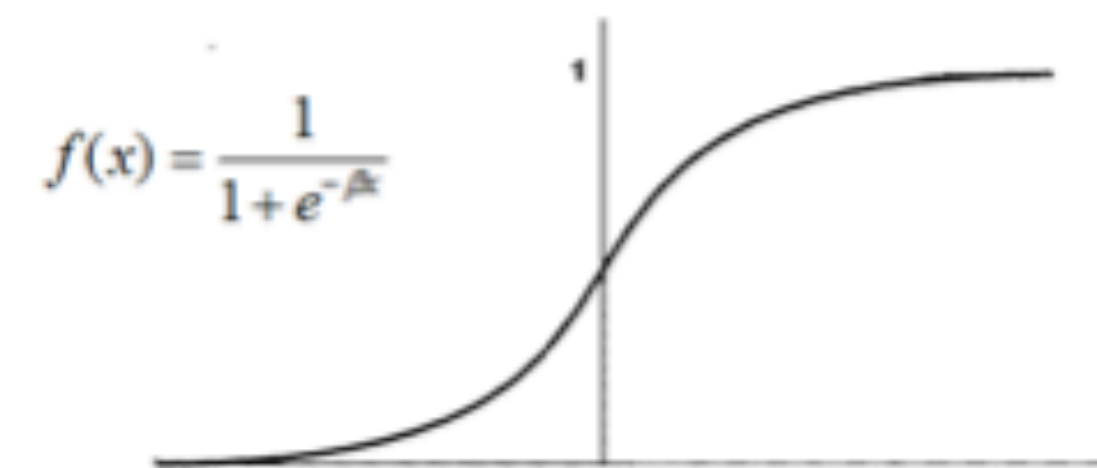
# The New York Times

Nov 23, 2012

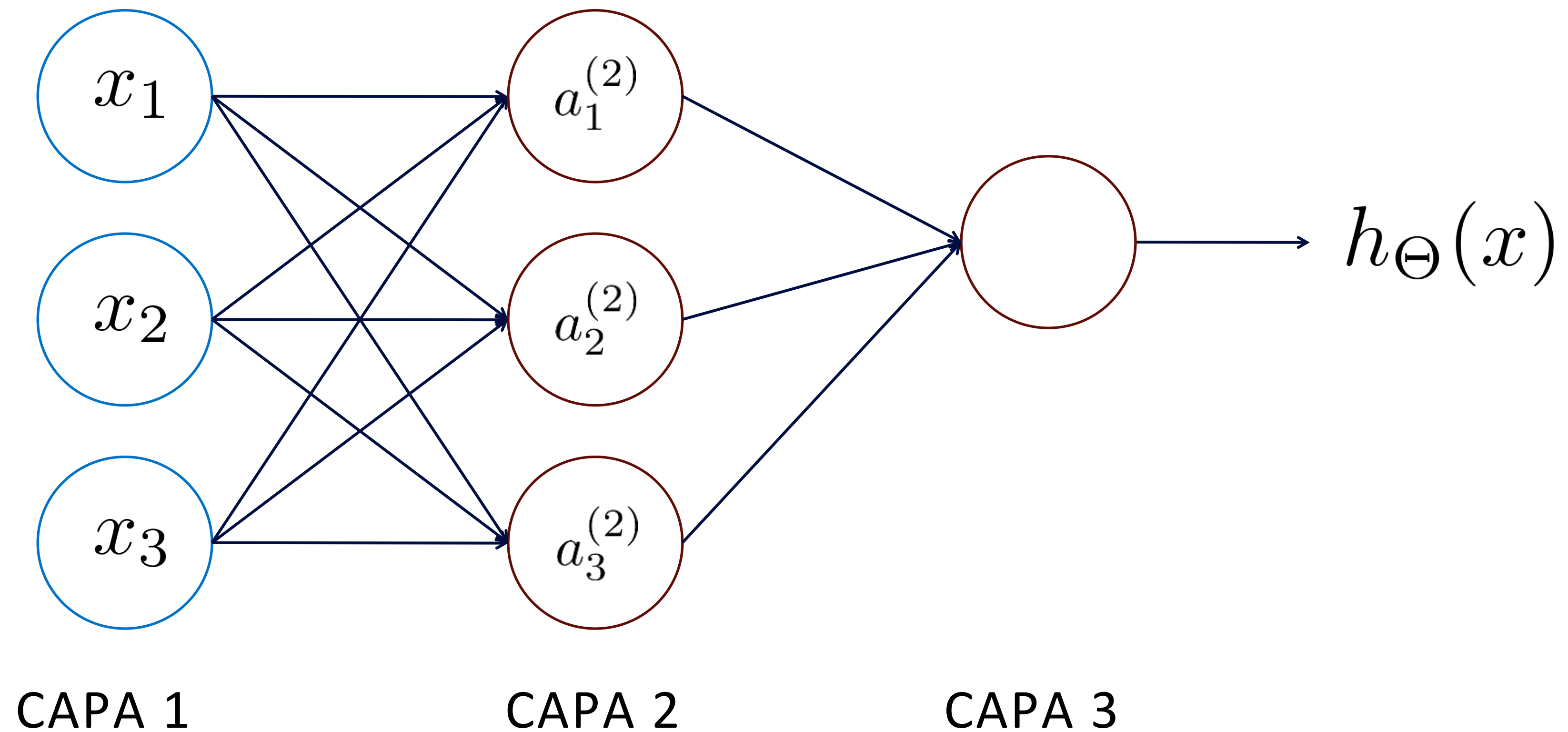
# Redes Neuronales Profundas – la neurona



Función de activación sigmoide (logística)

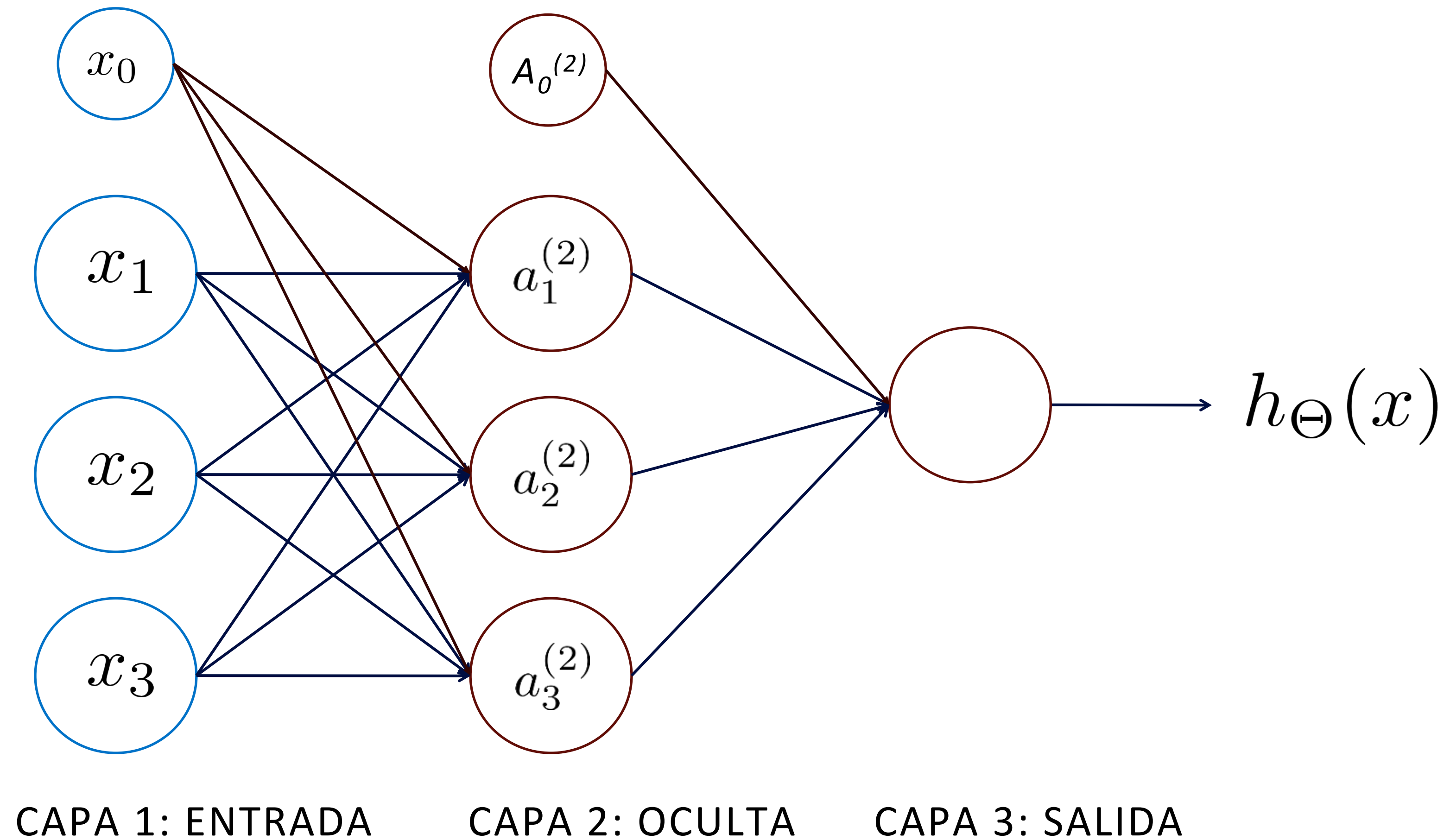


# Redes Neuronales Profundas

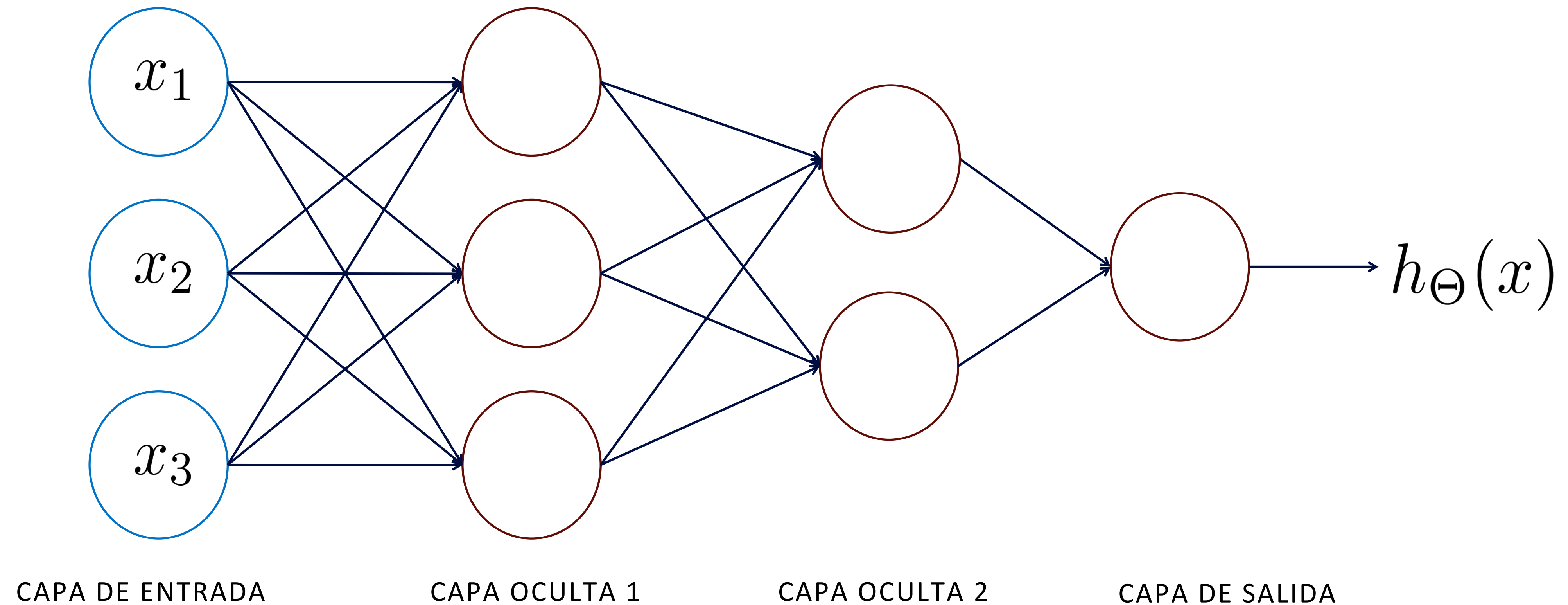




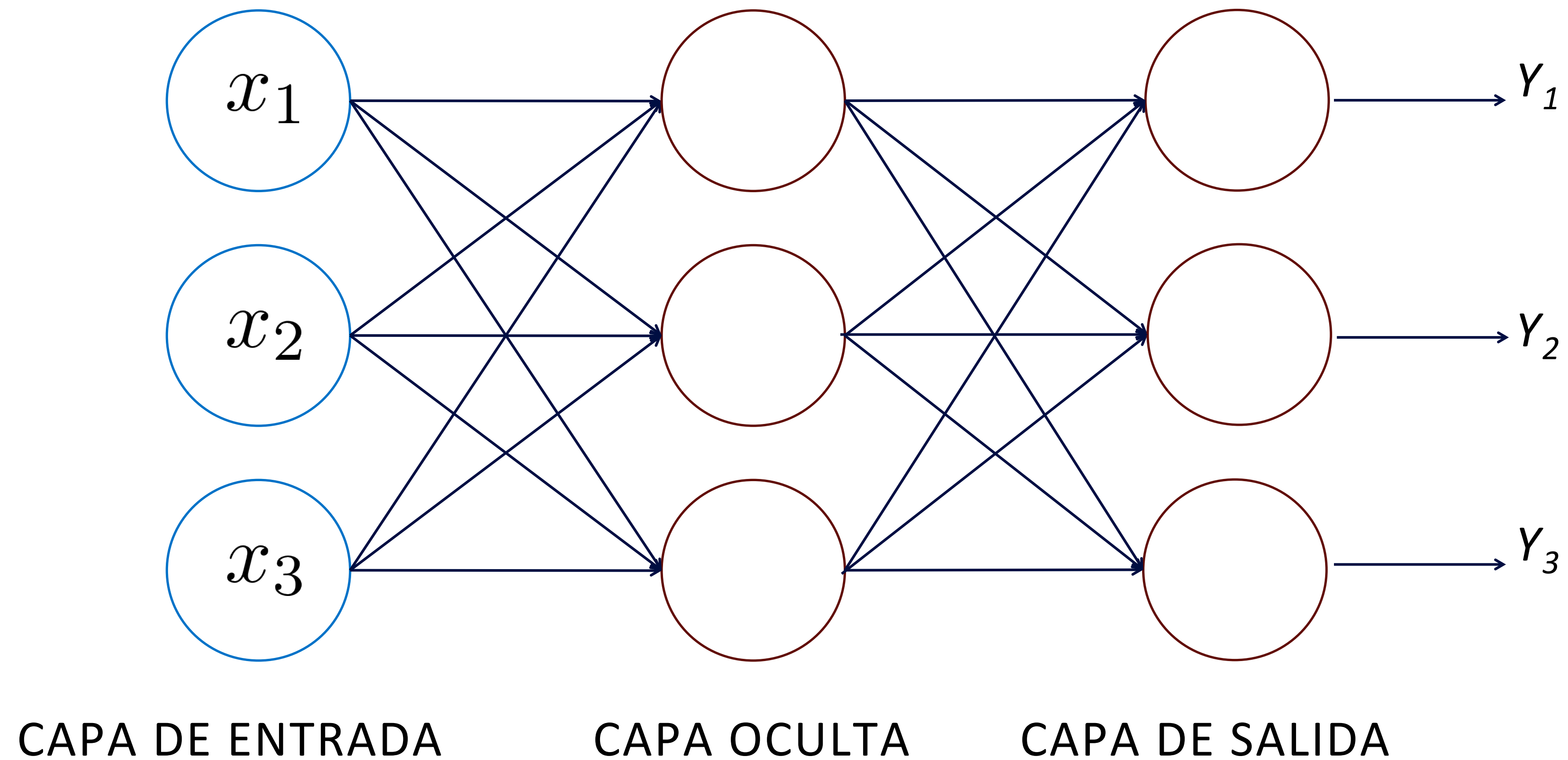
# Redes Neuronales Profundas



# Redes Neuronales Profundas



# Redes Neuronales Profundas





# Redes Neuronales Profundas – Clasificación Multiclase



PEATÓN



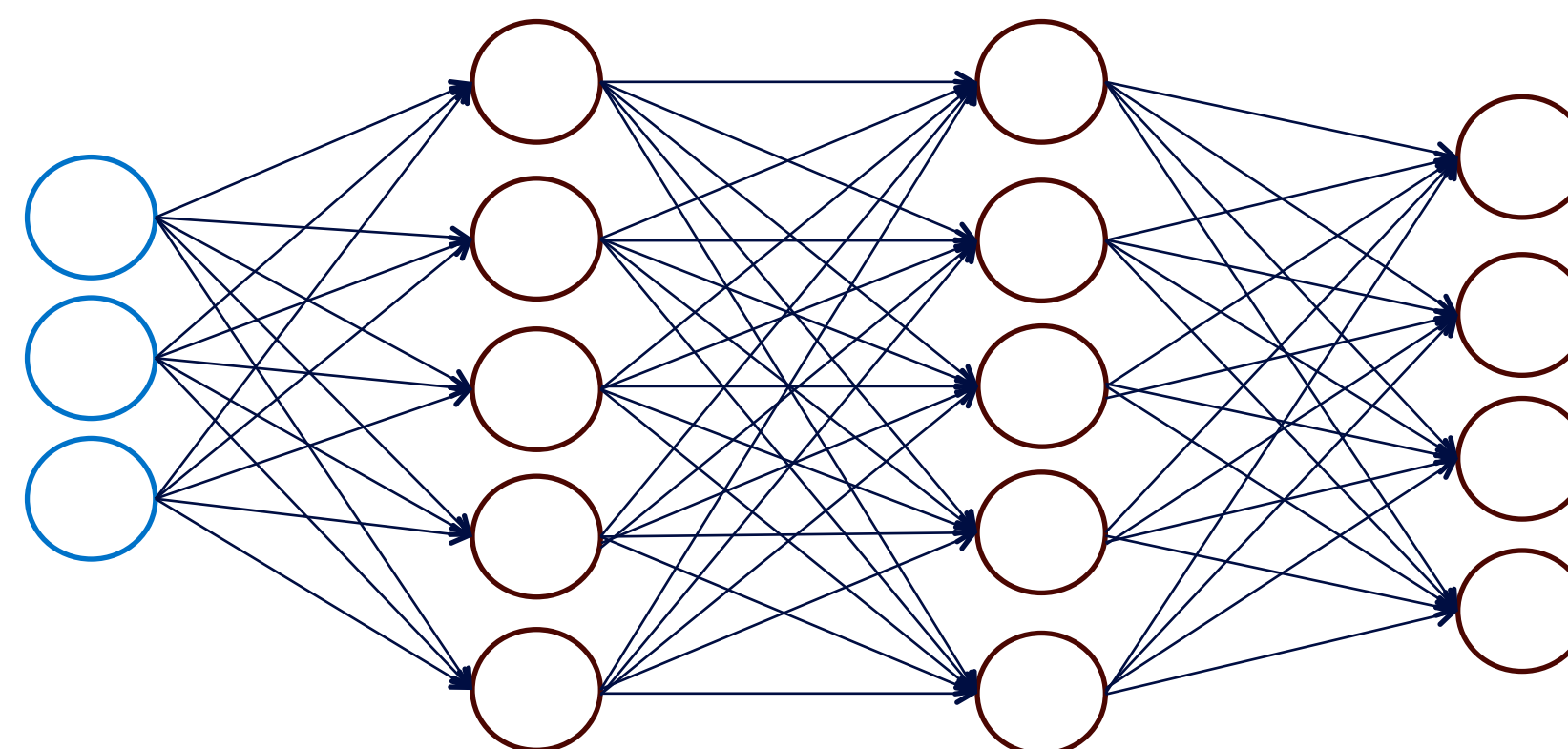
COCHE



MOTO



CAMIÓN



$$h_{\Theta}(x) \in \mathbb{R}^4$$

$$h_{\Theta}(x) \approx \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

PEATÓN

$$h_{\Theta}(x) \approx \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}$$

COCHE

$$h_{\Theta}(x) \approx \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

MOTO

ETC...

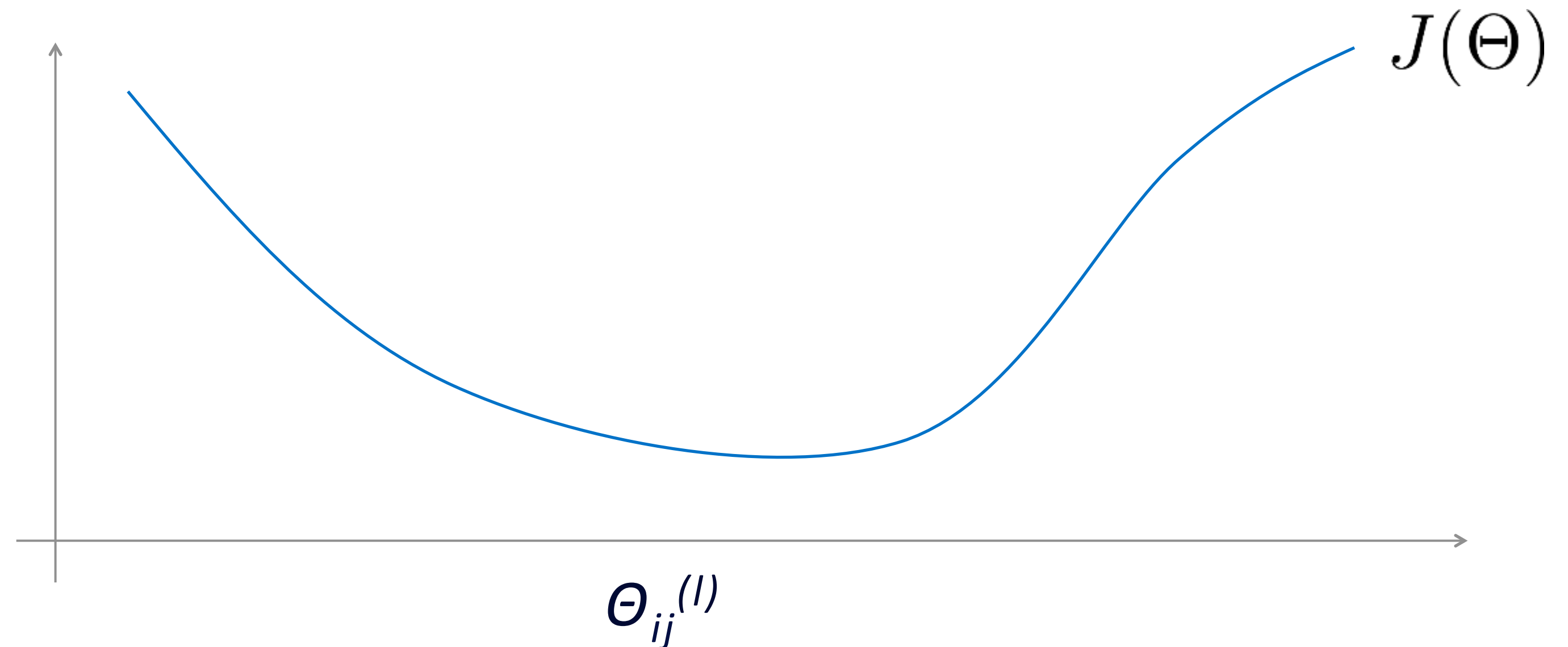
# Redes Neuronales Profundas – Función de Coste

$J(\Theta)$  = Función de coste, mide el rendimiento actual de la red neuronal

Objetivo:  $\min_{\Theta} J(\Theta)$

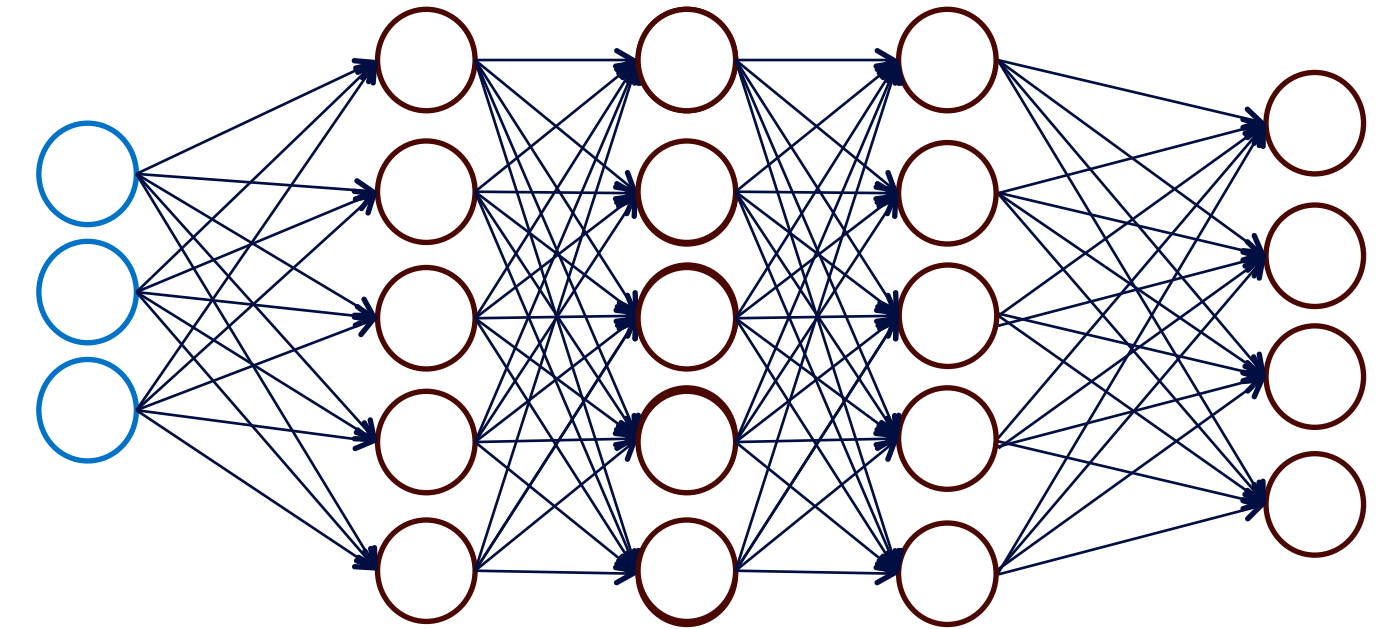
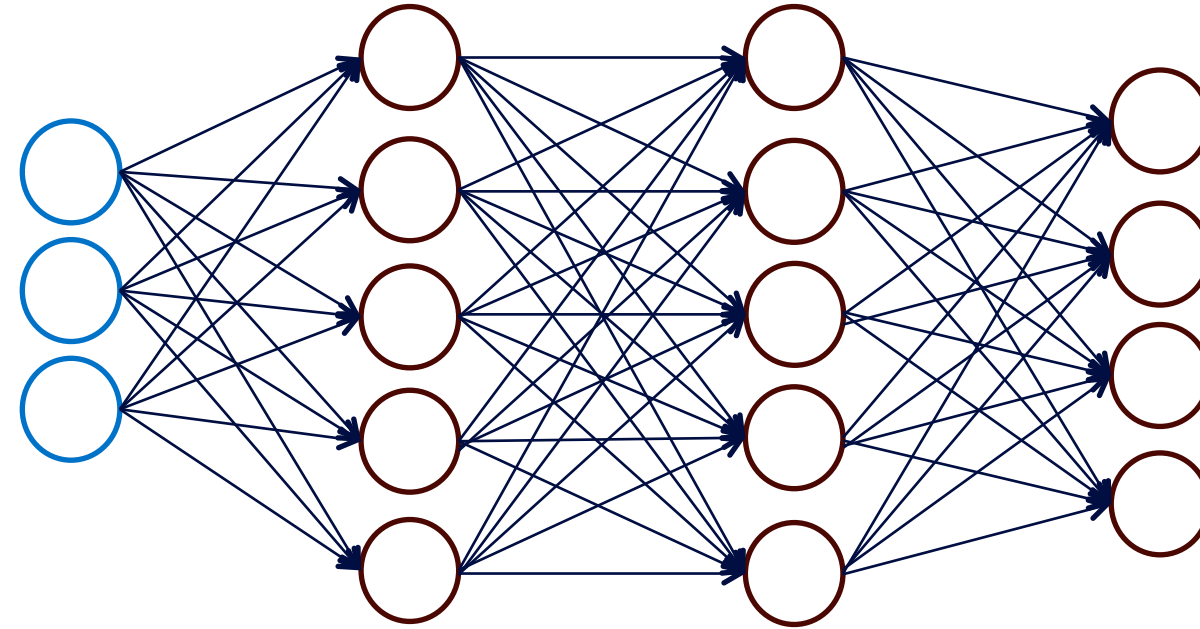
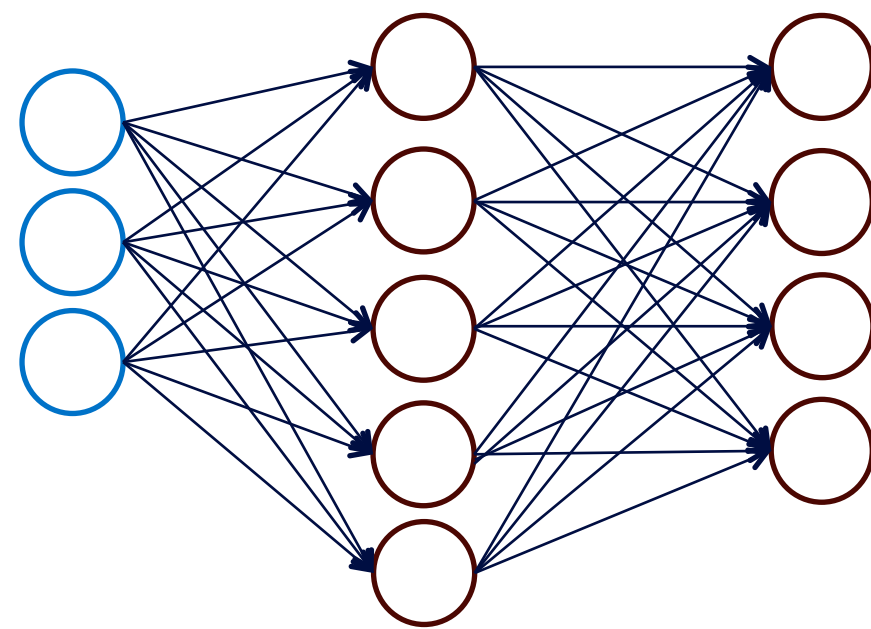
Se debe calcular:

- 1.  $J(\Theta)$
- 2.  $\frac{\partial}{\partial \Theta_{ij}^{(l)}} J(\Theta)$
- 3. Back-propagation



# Redes Neuronales Profundas – Receta

- Escoge la arquitectura (esquema de conectividad entre neuronas)

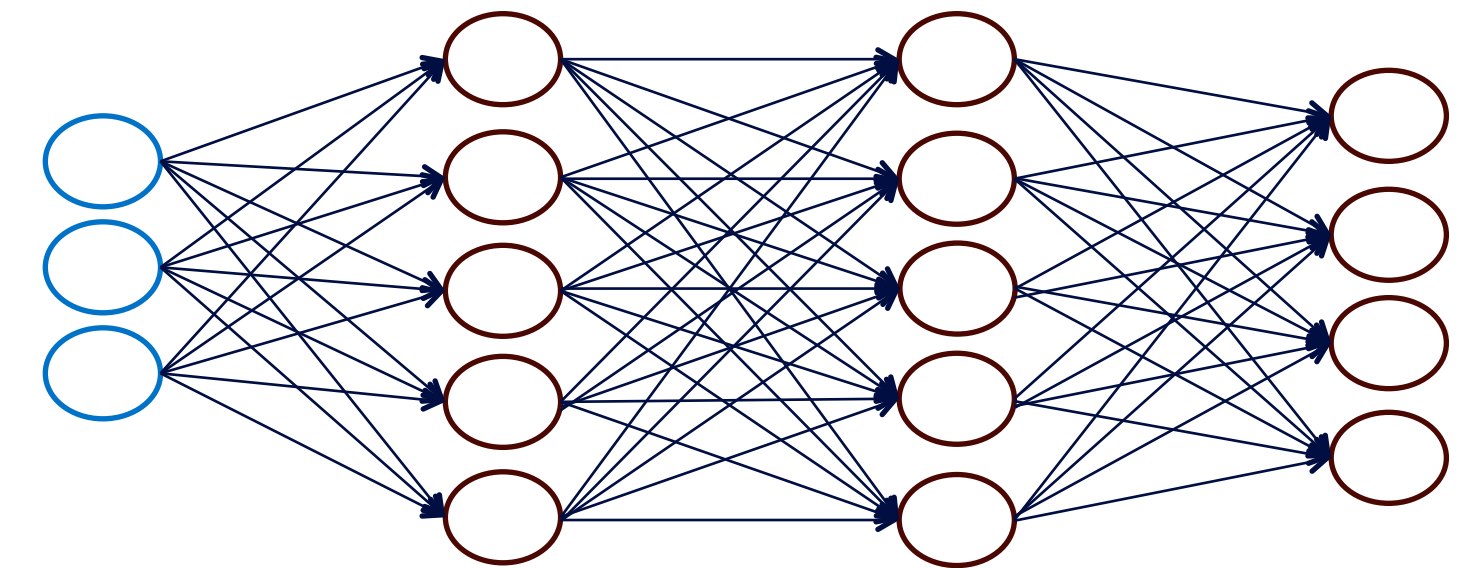


- Número de unidades de entrada: dimensión de las características  $x^{(i)}$
- Número de unidades de salida: número de clases
- Esquema típico: el número de capas ocultas dependerá del problema que estemos tratando de resolver, a más complejo es el problema más capas ocultas necesitaremos (y más datos)



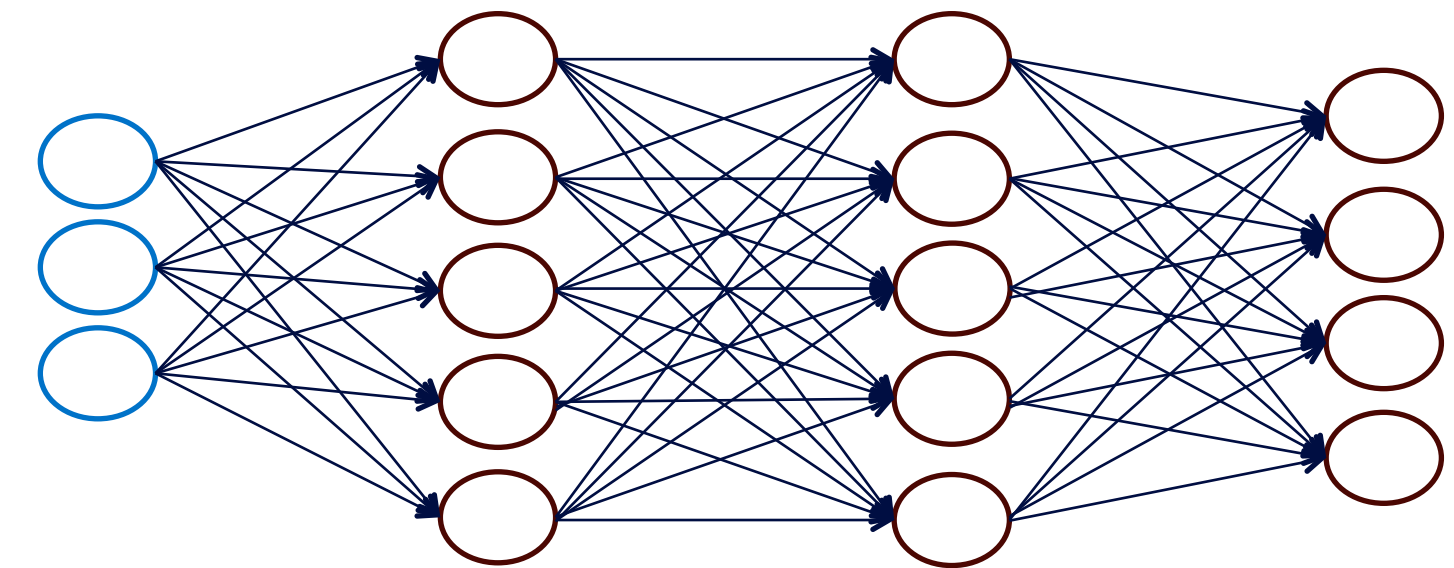
# Redes Neuronales Profundas – Receta

- 1. Inicialice los pesos de forma aleatoria



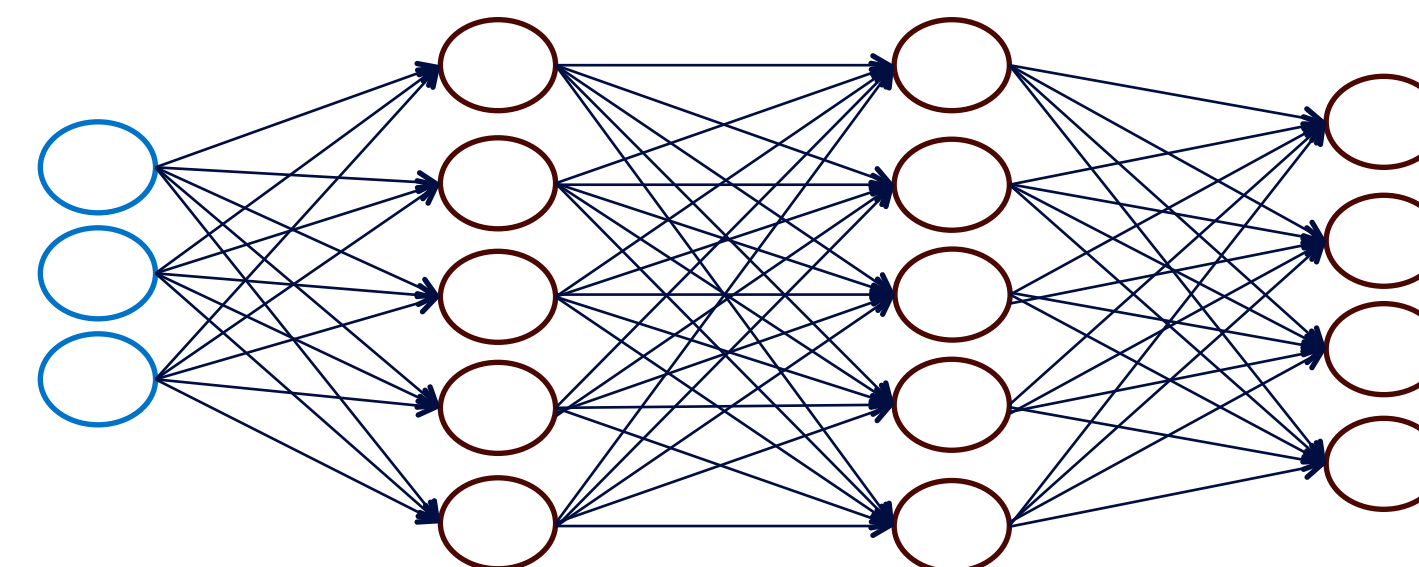
# Redes Neuronales Profundas – Receta

- 1. Inicialice los pesos de forma aleatoria
- 2. Realice la propagación forward para obtener  $h_{\Theta}(x^{(i)})$  para cualquier  $x^{(i)}$



# Redes Neuronales Profundas – Receta

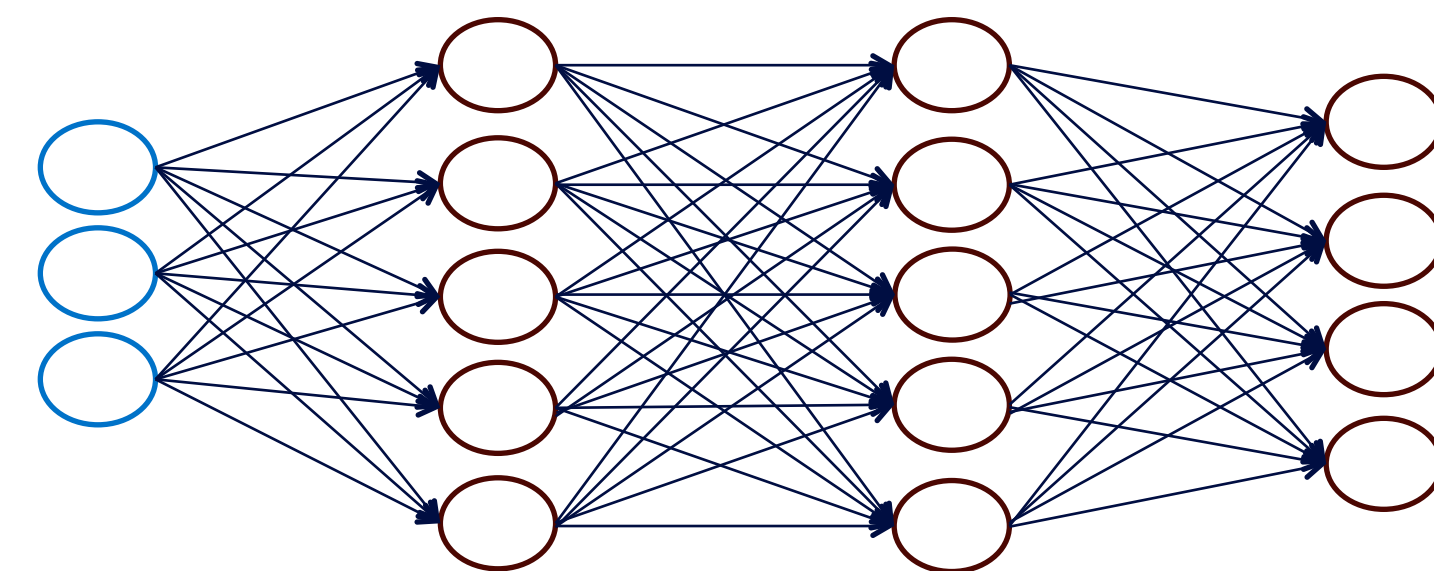
- 1. Inicialice los pesos de forma aleatoria
- 2. Realice la propagación forward para obtener  $h_{\Theta}(x^{(i)})$  para cualquier  $x^{(i)}$
- 3. Calcule la función de coste  $J(\Theta)$





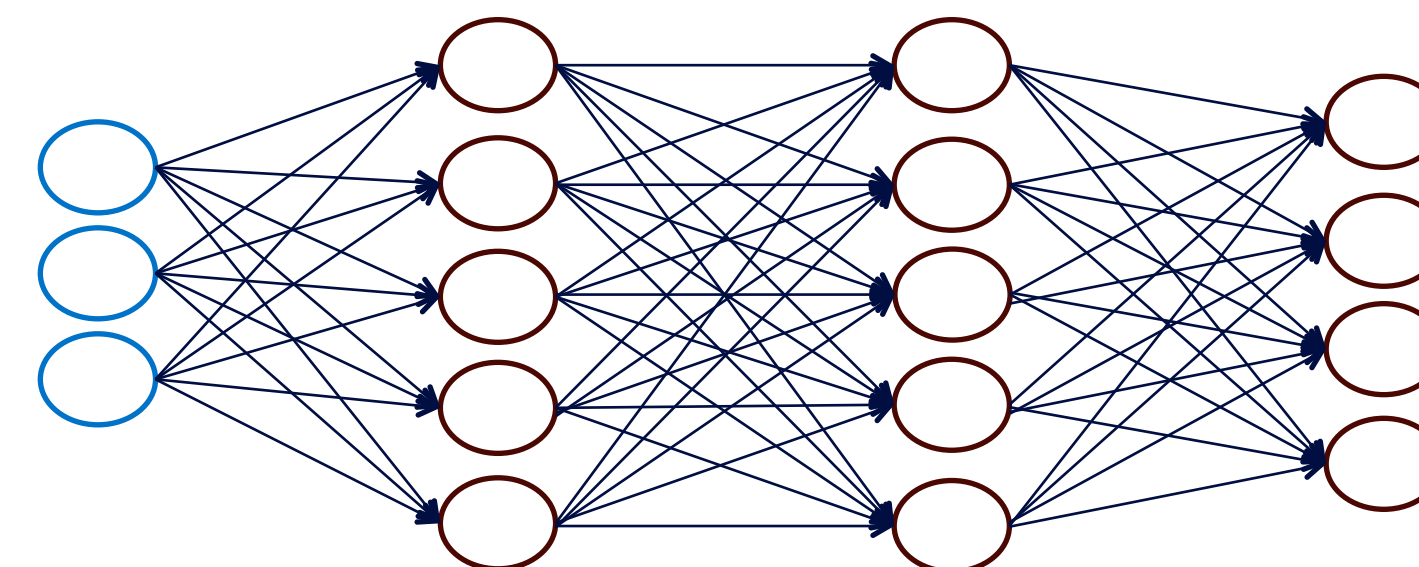
# Redes Neuronales Profundas – Receta

- 1. Inicialice los pesos de forma aleatoria
- 2. Realice la propagación forward para obtener  $h_{\Theta}(x^{(i)})$  para cualquier  $x^{(i)}$
- 3. Calcule la función de coste  $J(\Theta)$
- 4. Realice la propagación backward, calcular las derivadas parciales  $\frac{\partial}{\partial \Theta_{jk}^{(l)}} J(\Theta)$



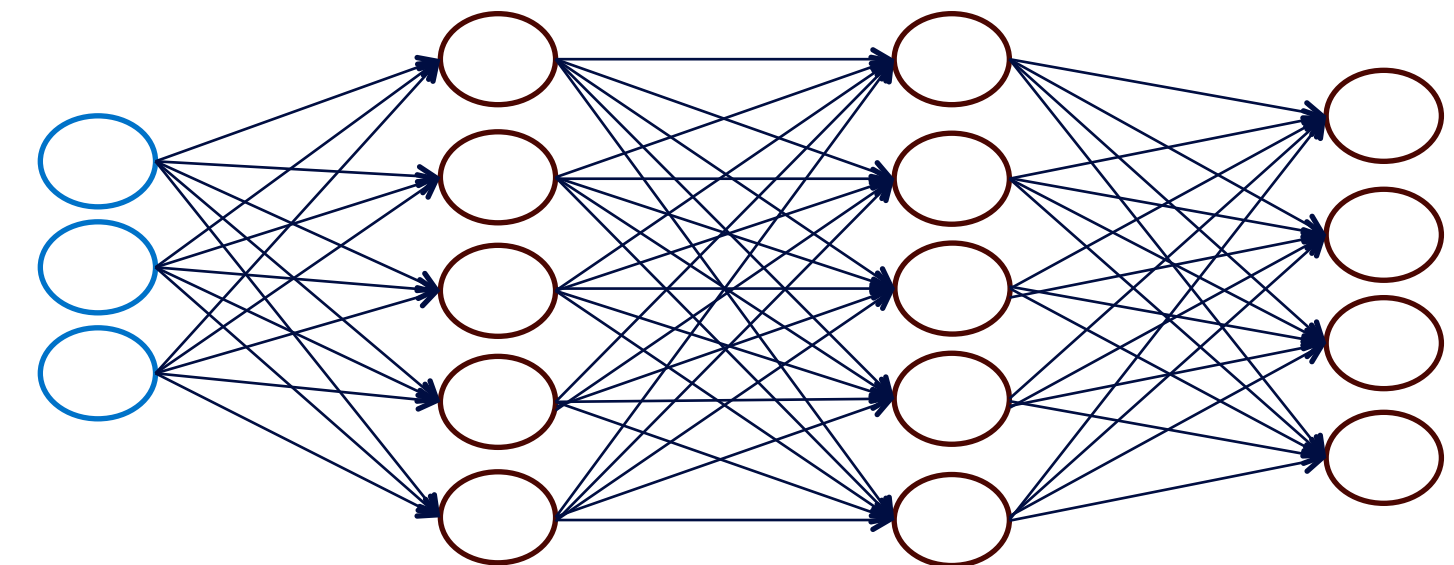
# Redes Neuronales Profundas – Receta

- 1. Inicialice los pesos de forma aleatoria
- 2. Realice la propagación forward para obtener  $h_{\Theta}(x^{(i)})$  para cualquier  $x^{(i)}$
- 3. Calcule la función de coste  $J(\Theta)$
- 4. Realice la propagación backward, calcular las derivadas parciales  $\frac{\partial}{\partial \Theta_{jk}^{(l)}} J(\Theta)$
- 5. Actualice los pesos acorde al learning rate



# Redes Neuronales Profundas – Receta

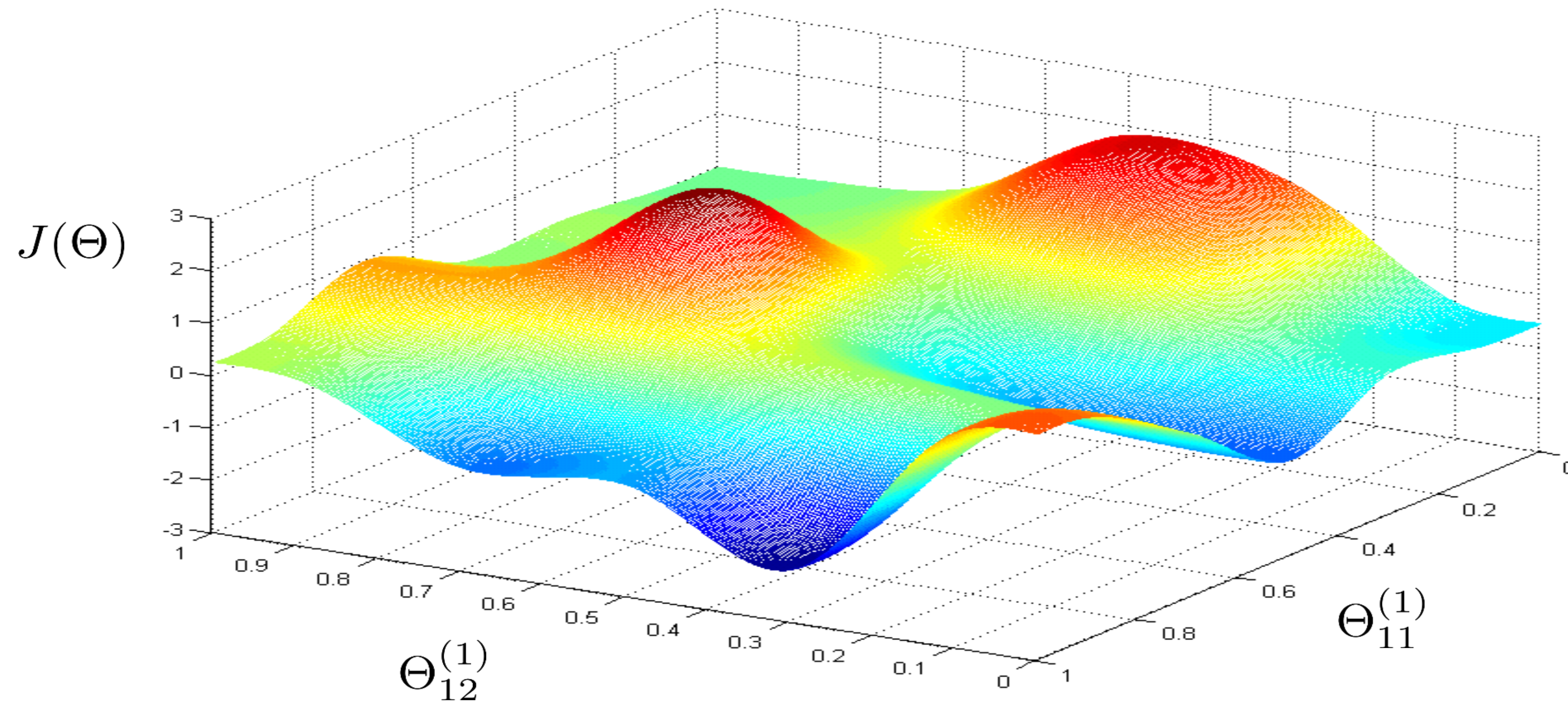
- 1. Inicialice los pesos de forma aleatoria
- 2. Realice la propagación forward para obtener  $h_{\Theta}(x^{(i)})$  para cualquier  $x^{(i)}$
- 3. Calcule la función de coste  $J(\Theta)$
- 4. Realice la propagación backward, calcular las derivadas parciales  $\frac{\partial}{\partial \Theta_{jk}^{(l)}} J(\Theta)$
- 5. Actualice los pesos acorde al learning rate
- 6. Repita los pasos 2 a 6 hasta alcanzar el criterio de parada





# Redes Neuronales Profundas – Receta

Parametros iniciales (inicialización aleatoria)

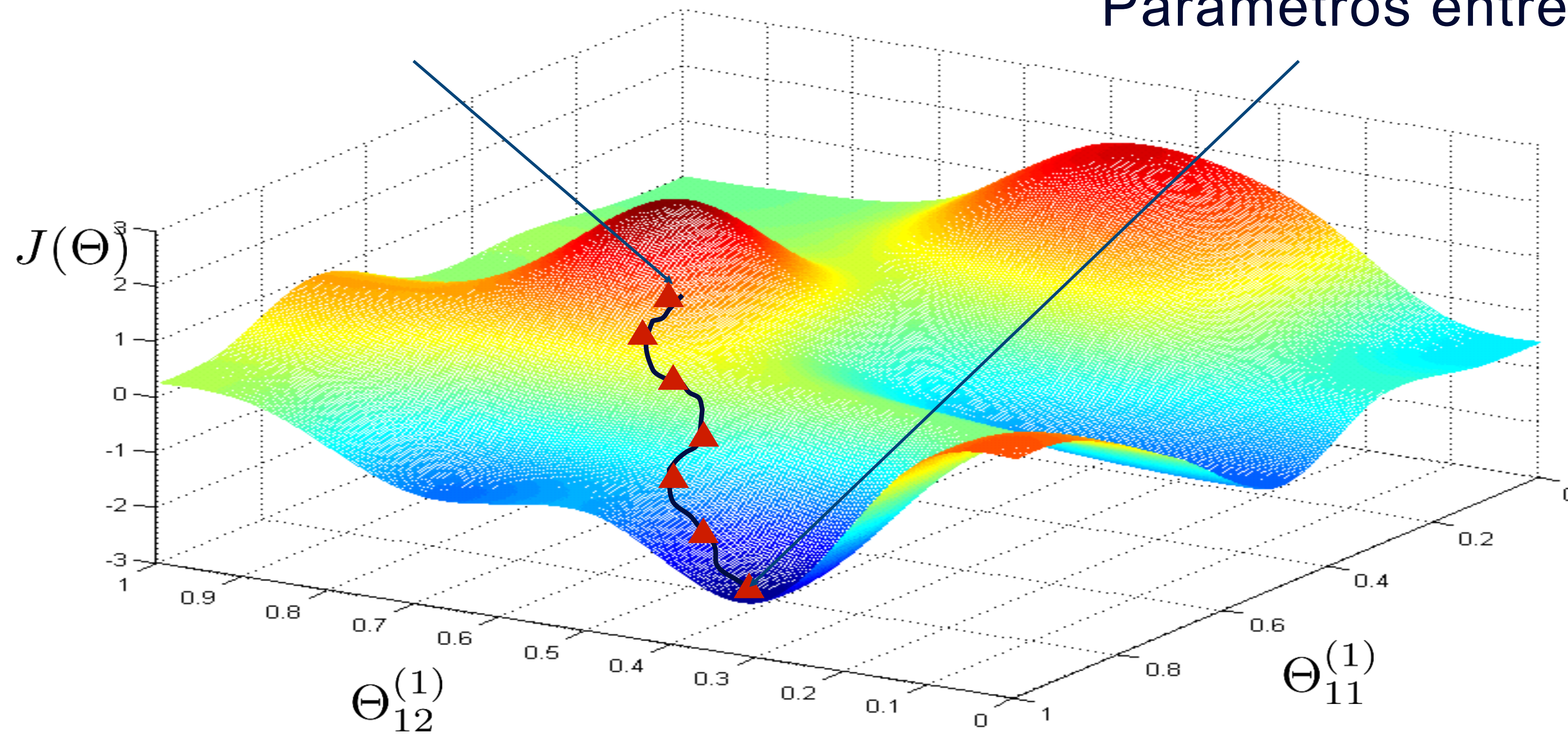




# Redes Neuronales Profundas – Receta

Parametros iniciales (aleatorios)

Parametros entrenados





# Redes Neuronales Profundas – Receta

- Limitaciones de back-propagation:
  - Sobreentrenamiento
    - Seleccionar bien el modelo
    - Datos
- Mínimo local





# Redes Neuronales Profundas – ¿Para qué sirven?

- Agente Racional
  - Reconocimiento de Imágenes
  - Reconocimiento de Sonidos
  - Reconocimiento de Patrones
  - Aproximación de Funciones
  - Control Inteligente
  - Generación música
  - ...
- Pero entonces... para qué no sirven???



# Redes Neuronales Profundas – ¿Para qué no sirven?

- Cuando los datos de entrenamiento no permiten generalizar:
  - Técnicas para evitar el overfitting
  - Pero... se necesitan datos similares a los de evaluación
  - No son capaces de “inventar”
- Se requieren soluciones interpretables:
  - En general, es muy difícil saber qué ha aprendido la red y por qué
- Entre otras cosas...



# Redes Neuronales Profundas – Falsos Tópicos

- Son modelos del cerebro
- Cuanto más grande es el modelo, mejor
- Requieren muchos datos
- Son cajas negras
- Son difíciles de implementar





# Tensorflow y Keras

- **TensorFlow:** Una herramienta software de código abierto desarrollada por Google pensada para enfrentarse a cualquier problema de Inteligencia Artificial, con mucho éxito en el campo de las redes neuronales.
- **Keras:** API de alto nivel, trabaja sobre TensorFlow y otras librerías.
- Actualización Oct 2019:
  - TensorFlow 2.0: Se unifican TensorFlow y Keras!!



# Worksheet

- Worksheet Sesión 1
  - Preparación del entorno, instalación
  - Introducción a TensorFlow
  - MachineLearning in Tensorflow

