Lecture 13
Neural Networks: Part I

GEOL 4397: Data analytics and machine learning for geoscientists

Jiajia Sun, Ph.D.

April. 9th, 2019

UNIVERSITY of HOUSTON
YOU ARE THE PRIDE
EARTH AND ATMOSPHERIC SCIENCES

Week	Date	Topics	Comments
1	01/15 Tues	Overview of syllabus Lecture: Introduction to Machine learning: applications	
	01/17 Thur	Lecture: Review of linear algebra	
2	01/22 Tues	Lab: Linear algebra in Python	Not graded
	01/24 Thur	Lecture: Introduction to optimization	
3	01/29 Tues	Lab: Gradient descent + Linear regression	Report due on 02/05 at 5:30 pm
	01/31 Thur	Lecture: Introduction to machine learning: concepts	
4	02/05 Tues	Lecture: Logistic regression	
	02/07 Thur	Lab: Logistic regression	Report due on 02/14 at 5:30 pm
5	02/12 Tues	Lecture: Support vector machine	
	02/14 Thur	Lab: Support vector machine	Report due on 02/21 at 5:30 pm
6	02/19 Tues	Lecture: Decision trees	
	02/21 Thur	Lab: Decision trees	Report due on 02/28 at 5:30 pm
7	02/26 Tues	Lecture: Random Forest	
	02/28 Thur	Lab: Random forest	Report due on 03/07 at 5:30 pm
8	03/05 Tues	Lecture: Ensemble learning	
	03/07 Thur	Lab: Ensemble learning	Reprot due on 03/19 at 5:30 pm
9	03/12 Tues	No class due to spring break	
	03/14 Thur	No class due to spring break	
10	03/19 Tues	Review & Recap	
	03/21 Thur	Exam	
11	03/26 Tues	Lecture: Clustering	
	03/28 Thur	Lab: Clustering	Report due on 04/04 at 5:30 pm
12	04/02 Tues	Lecture: Introduction to TensorFlow	
	04/04 Thur	Lab: TensorFlow	Not graded
13	04/09 Tues	Lecture: Introduction to neural networks 1	
	04/11 Thur	Lecture: Introduction to neural networks 2	
14	04/16 Tues	Lab: Deep learning	Report due on 04/23 at 5:30pm
	04/18 Thur	Lecture: Convolutional neural networks 1	
15	04/23 Tues	Guest lecture: Convolutional neural networks 2	
	04/25 Thur	Lab: CNN (optional)	Report due on 05/02 at 5:30 pm
16	04/30 Tues	final presentation??	
	05/02 Thur	final presentation??	
Note	28 class meetings		04/29 last day of class

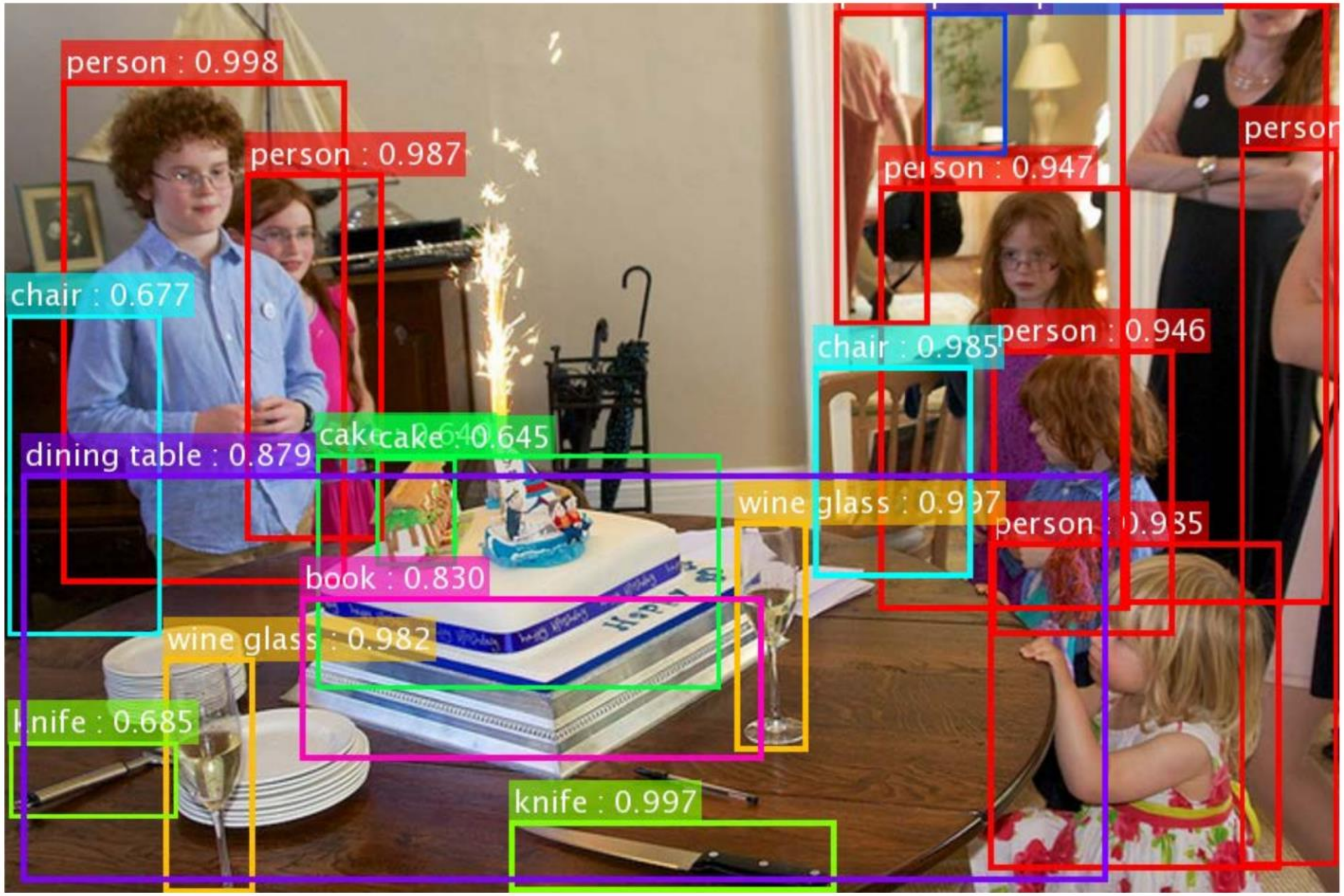
Outline

- What is neural networks?

- Forward propagation

- Backward propagation

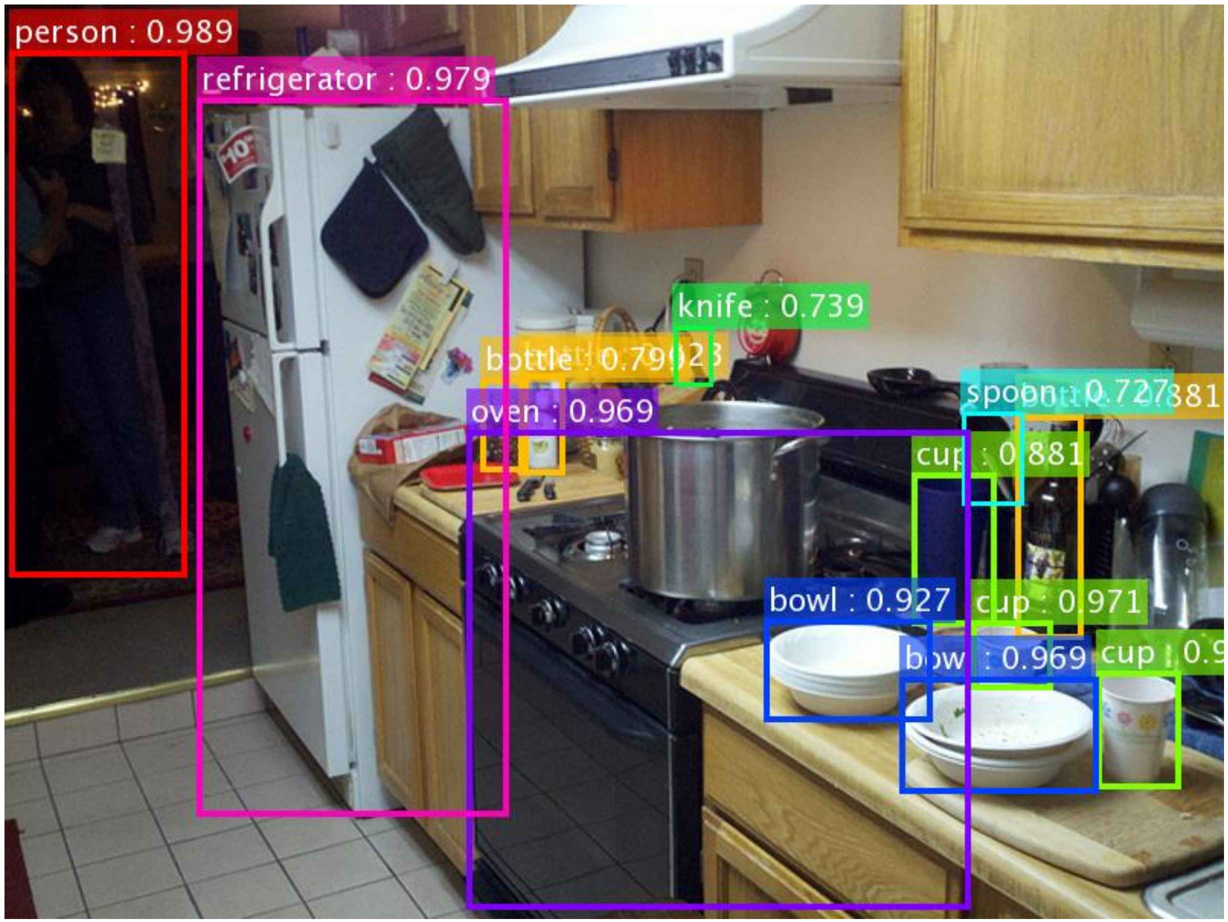
Object detection



ResNet applied to COCO dataset.

Source: He et al., Deep residual learning for image recognition, CVPR, 2016

Object detection

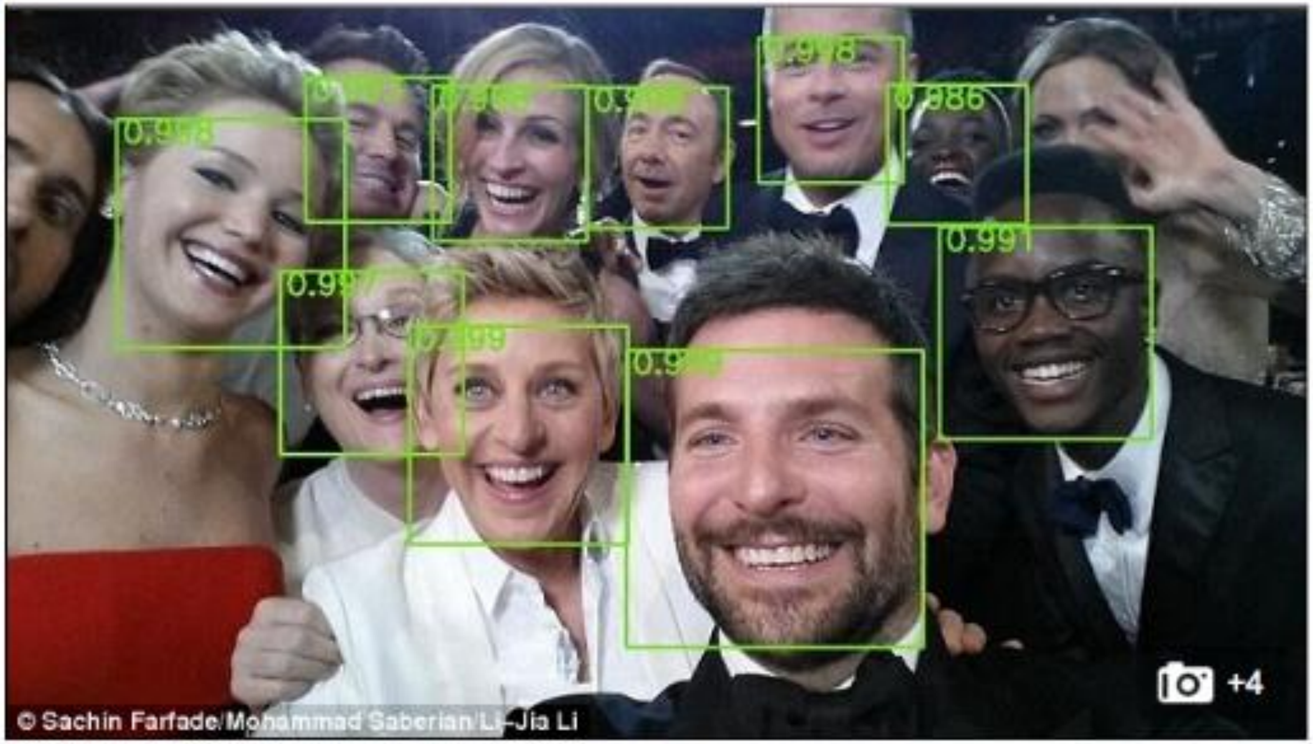


Source: He et al., Deep residual learning for image recognition, CVPR, 2016

Face detection

Convolution Neural Net (CNN)

Yahoo + Stanford example – find a face in a pic, even upside down

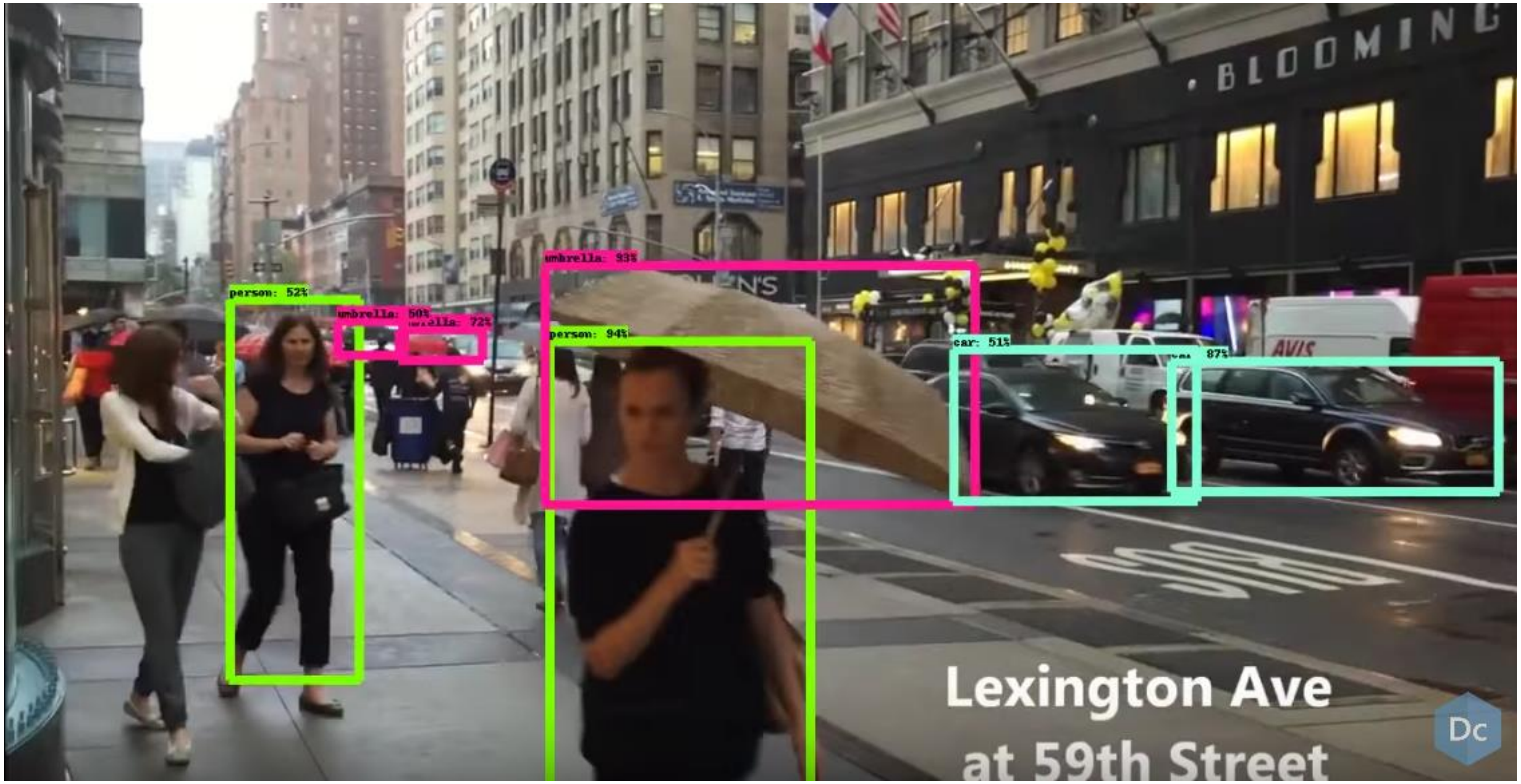


The Deep Dense Face Detector algorithm was built by Yahoo Labs in California and Stanford University. The researchers used a form of machine learning known as a deep convolutional neural network to train a computer to spot facial features (pictured) in a database of images

http://www.da

At the moment, the so-called Deep Dense Face Detector doesn't recognise who the individual faces belong to, just that there is a face.

Real time object detection



Video online: https://www.youtube.com/watch?v=_zZe27JYi8Y

Real time object detection

YOLO V2 achieves better results at very high FPS

Video online: https://www.youtube.com/watch?v=VOC3huqHrss&list=RDVOC3huqHrss

Go game

Image credit: Nature

Image credit: theverge.com

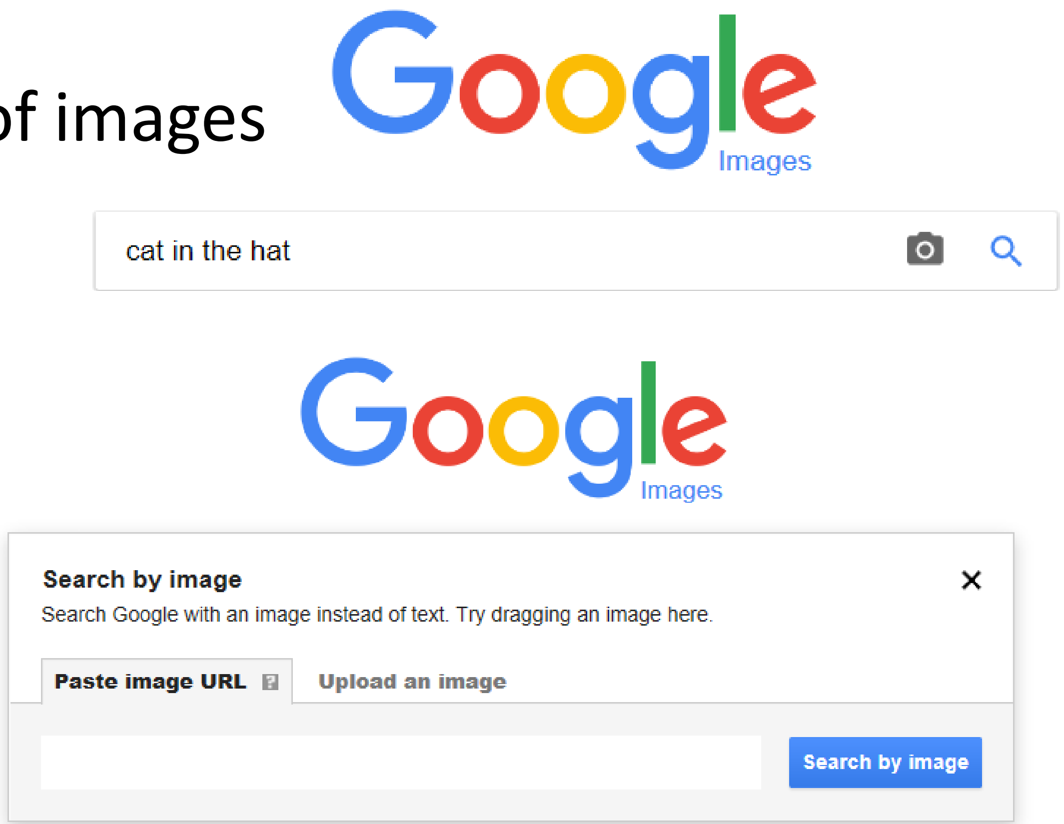
Deep learning applications

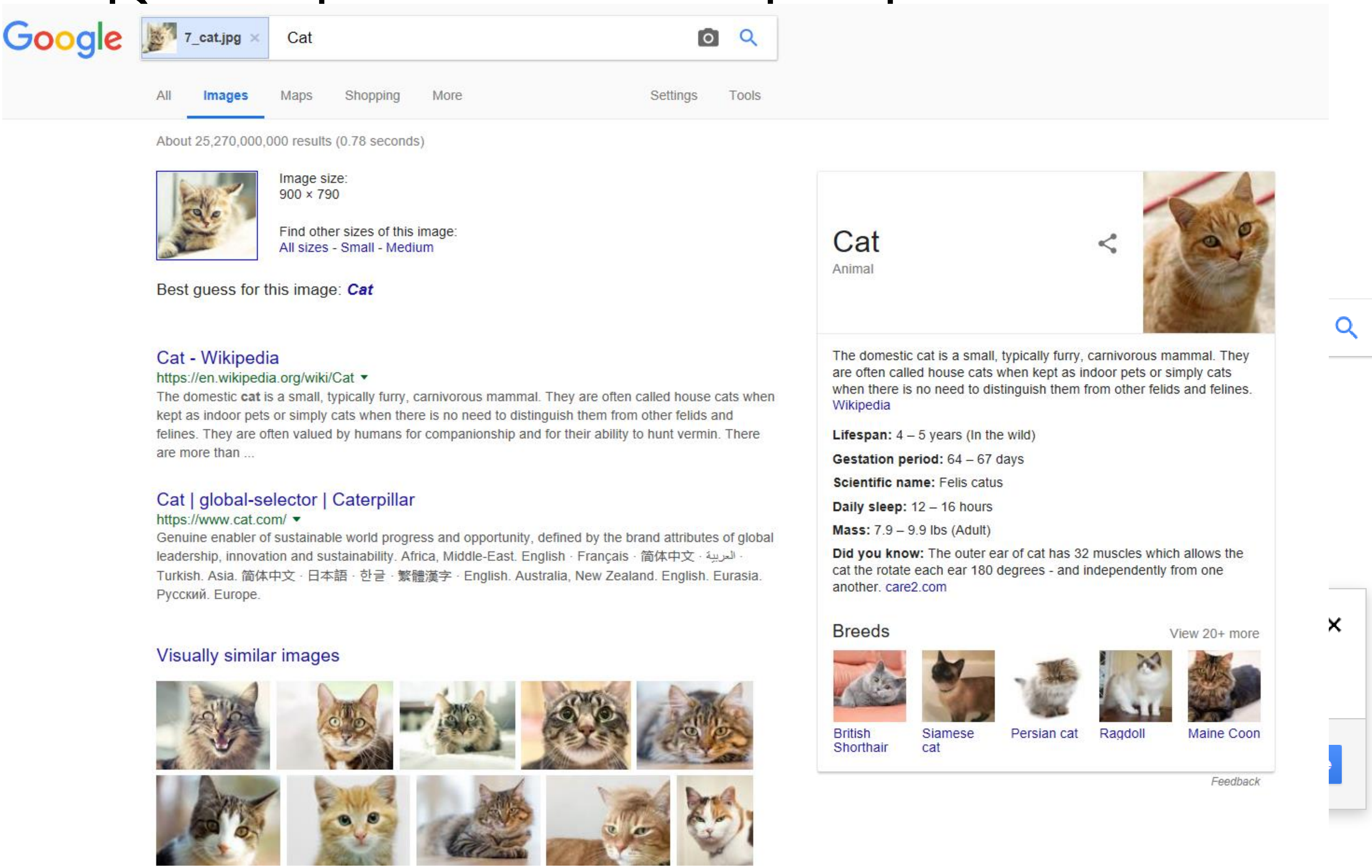
- Google images
  - Classifying billions of images

Deep learning applications

Deep learning applications

- Google images
- Classifying billions of images




7_cat.jpg
Cat
All
Images
Maps
Shopping
More
Settings
Tools
About 25,270,000,000 results (0.78 seconds)
Image size:
900 × 790
Find other sizes of this image:
All sizes - Small - Medium
Best guess for this image: Cat
Cat - Wikipedia
https://en.wikipedia.org/wiki/Cat
The domestic cat is a small, typically furry, carnivorous mammal. They are often called house cats when kept as indoor pets or simply cats when there is no need to distinguish them from other felids and felines. They are often valued by humans for companionship and for their ability to hunt vermin. There are more than ...
Cat | global-selector | Caterpillar
https://www.cat.com/
Genuine enabler of sustainable world progress and opportunity, defined by the brand attributes of global leadership, innovation and sustainability. Africa, Middle-East. English · Français · 简体中文 · العربية · Turkish. Asia. 简体中文 · 日本語 · 한글 · 繁體漢字 · English. Australia, New Zealand. English. Eurasia. Русский. Europe.
Visually similar images
Cat
Animal
The domestic cat is a small, typically furry, carnivorous mammal. They are often called house cats when kept as indoor pets or simply cats when there is no need to distinguish them from other felids and felines. Wikipedia
Lifespan: 4 – 5 years (In the wild)
Gestation period: 64 – 67 days
Scientific name: Felis catus
Daily sleep: 12 – 16 hours
Mass: 7.9 – 9.9 lbs (Adult)
Did you know: The outer ear of cat has 32 muscles which allows the cat the rotate each ear 180 degrees - and independently from one another. care2.com
Breeds
View 20+ more
British Shorthair
Siamese cat
Persian cat
Ragdoll
Maine Coon
Feedback

Deep learning applications

- Google images
  - Classifying billions of images

- Apple’s Siri
  - Powering speech recognition

- Recommending the best videos to watch (e.g., YouTube)
- Beat world champion at the game of GO

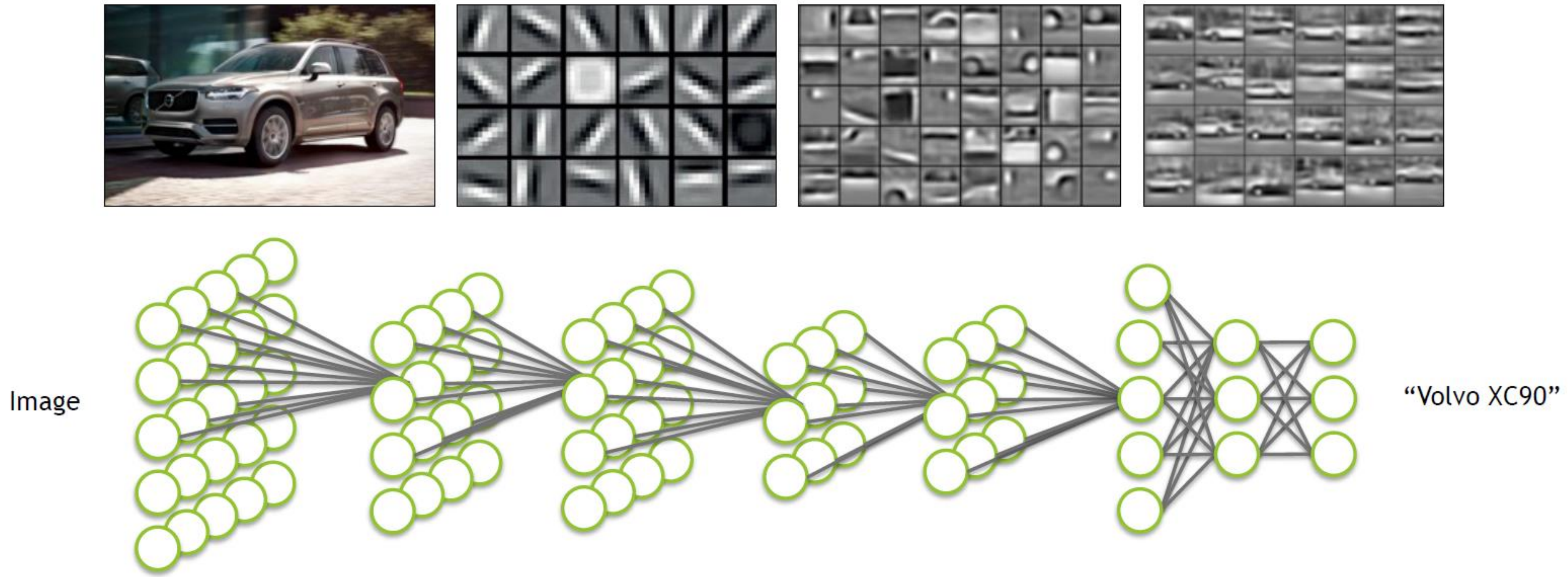
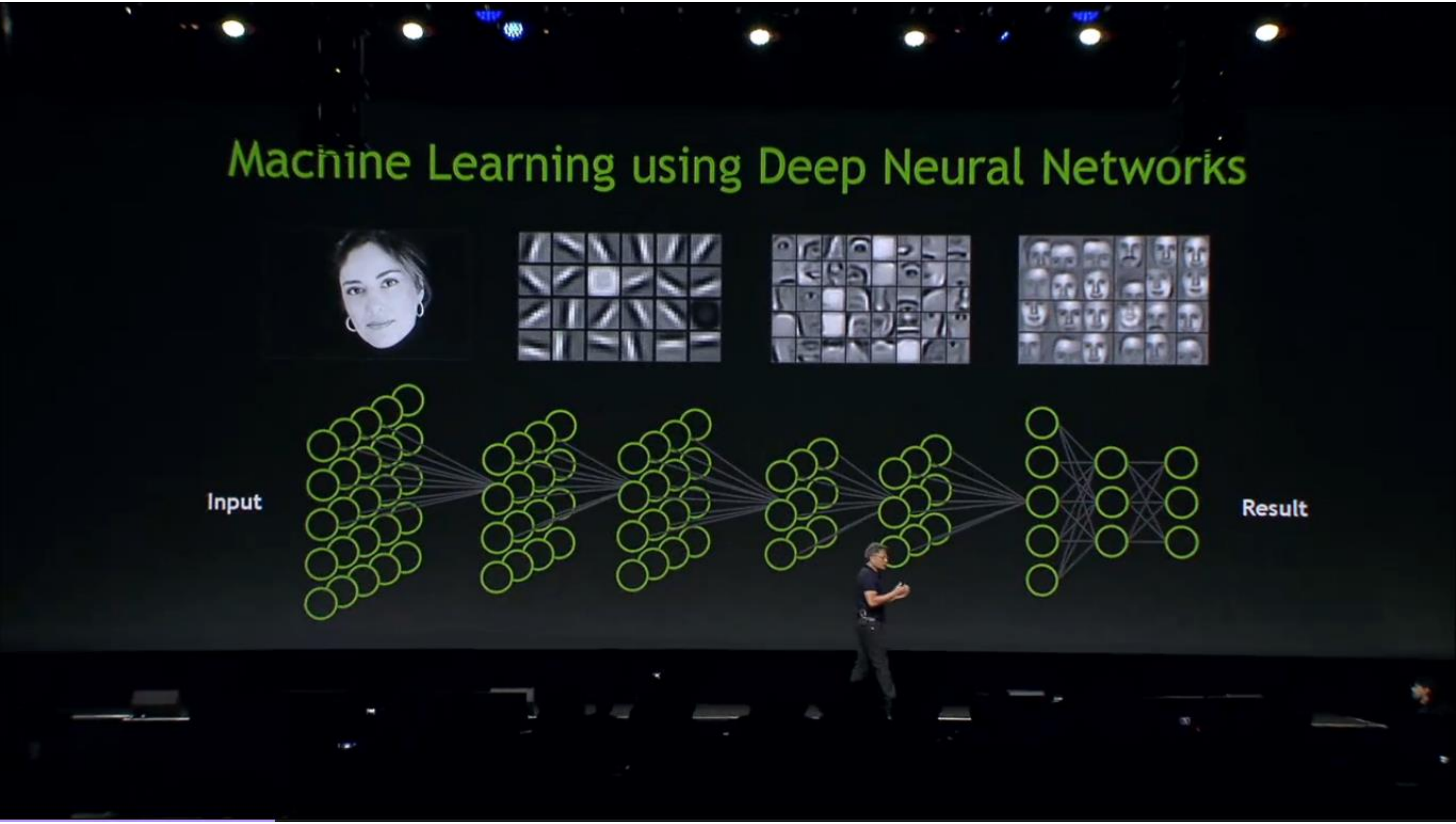


Image source: *"Unsupervised Learning of Hierarchical Representations with Convolutional Deep Belief Networks" ICML 2009 & Comm. ACM 2011. Honglak Lee, Roger Grosse, Rajesh Ranganath, and Andrew Ng.*


Machine Learning using Deep Neural Networks
Input
Result

History of neural networks

- First introduced back in 1943 by Warren McCulloch and Walter Pitts

Aurelien Geron, 2017, Hands-on Machine Learning with Scikit-Learn & TensorFlow, pp 254

History of neural networks

- First introduced back in 1943 by Warren McCulloch and Walter Pitts
- In 1960s, realized that ANNs was quite limited

Aurelien Geron, 2017, Hands-on Machine Learning with Scikit-Learn & TensorFlow, pp 254

History of neural networks

- First introduced back in 1943 by Warren McCulloch and Walter Pitts
- In 1960s, realized that ANNs was quite limited
- Early 1980s, a revival of interest in ANNs
  - new network architectures were invented
  - better training techniques developed

Aurelien Geron, 2017, Hands-on Machine Learning with Scikit-Learn & TensorFlow, pp 254

History of neural networks

- First introduced back in 1943 by Warren McCulloch and Walter Pitts
- In 1960s, realized that ANNs was quite limited
- Early 1980s, a revival of interest in ANNs
  - new network architectures were invented
  - better training techniques developed
- 1990s, alternative ML algorithms such as SVM were favored
  - They seemed to offer better results and stronger theoretical foundations

Aurelien Geron, 2017, Hands-on Machine Learning with Scikit-Learn & TensorFlow, pp 254

History of neural networks

- First introduced back in 1943 by Warren McCulloch and Walter Pitts
- In 1960s, realized that ANNs was quite limited
- Early 1980s, a revival of interest in ANNs
  - new network architectures were invented
  - better training techniques developed
- 1990s, alternative ML algorithms such as SVM were favored
  - They seemed to offer better results and stronger theoretical foundations
- Around 2010, another wave of interest

Aurelien Geron, 2017, Hands-on Machine Learning with Scikit-Learn & TensorFlow, pp 254

2012

- Breakthrough in solving the ImageNet competition by AlexNet (achieved a top-5 error of 15.3%, more than 10.8% lower than that of the runner up, thanks to GPU and very deep neural networks).
- Often considered as the beginning of the deep learning revolution of the 2010s.
- Google brain (a deep learning team at Google): in 2012, trained a neural network to recognize cat images based on 10 million images from YouTube (used 16,000 processors)
- New York Times, National Public Radio and SmartPlanet covered the story.

Why is machine learning taking off?

- Huge amount of labeled data available
    - Internet, social media, mobile devices, etc…

Aurelien Geron, 2017, Hands-on Machine Learning with Scikit-Learn & TensorFlow, pp 254

Why is machine learning taking off?

- Huge amount of labeled data available
  - Internet, social media, mobile devices, etc…
- Tremendous increase in computing power
  - Thanks to gaming industry for the use of GPUs

Aurelien Geron, 2017, Hands-on Machine Learning with Scikit-Learn & TensorFlow, pp 254

Why is machine learning taking off?

- Huge amount of labeled data available
  - Internet, social media, mobile devices, etc…
- Tremendous increase in computing power
  - Thanks to gaming industry for the use of GPUs
- Training algorithms improved
  - Xavier and He initialization
  - Batch normalization
  - Slightly different from the one used in 1990s, but having a huge positive impact

Aurelien Geron, 2017, Hands-on Machine Learning with Scikit-Learn & TensorFlow, pp 254

Why is machine learning taking off?

- Huge amount of labeled data available
  - Internet, social media, mobile devices, etc…
- Tremendous increase in computing power
  - Thanks to gaming industry for the use of GPUs
- Training algorithms improved
  - Xavier and He initialization
  - Batch normalization
  - Slightly different from the one used in 1990s, but having a huge positive impact
- Better understanding of the ANNs
  - Theoretical proof that existence of local minima is not a problem
- A healthy circle of funding and progress
  - Amazing products regularly make headline news
  - Draws more attention and funding to ANN R&D
  - Resulting in more progress and better products

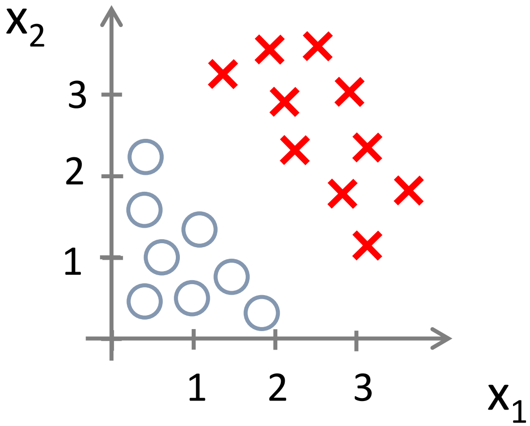
Aurelien Geron, 2017, Hands-on Machine Learning with Scikit-Learn & TensorFlow, pp 254

Neural networks

- Feedforward Neural networks
  - Most classical
- Convolutional Neural Networks
  - Extremely powerful at computer vision tasks
- Recurrent Neural Networks
  - Analyze time series and make predictions
  - E.g., analyze stock prices, and tell you when to buy or sell
  - E.g., natural language processing (NLP)

Understanding logistic regression

Decision Boundary



$$h_w(x) = g(w_0 + w_1 x_1 + w_2 x_2)$$

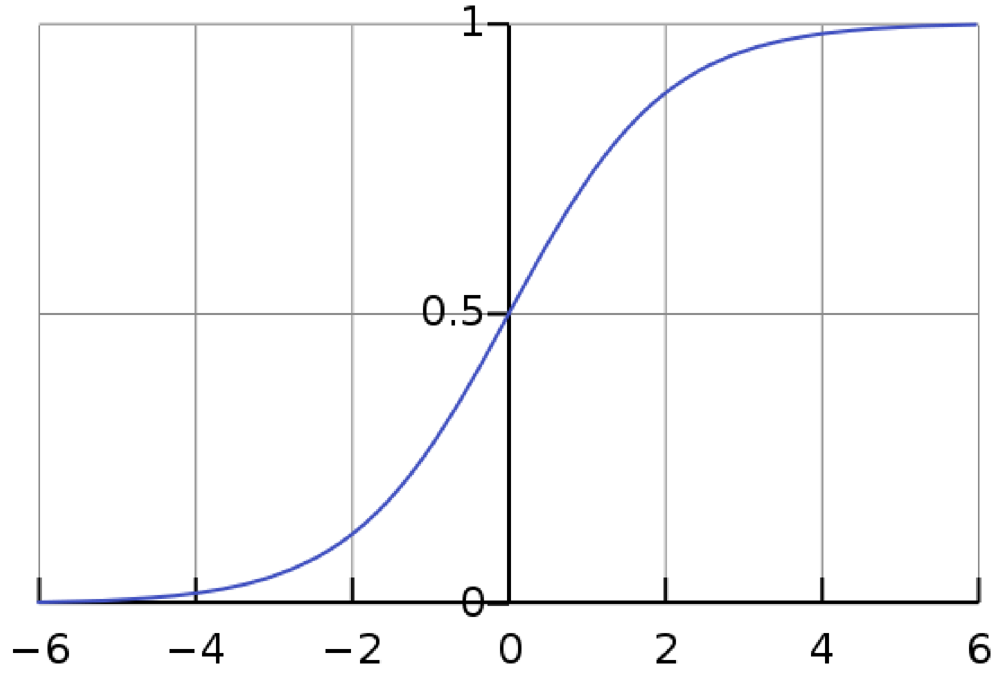
Predict "$y = 1$" if $-3 + x_1 + x_2 \geq 0$

This slide is taken from Andrew Ng's ML class on coursera

Logistic function

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

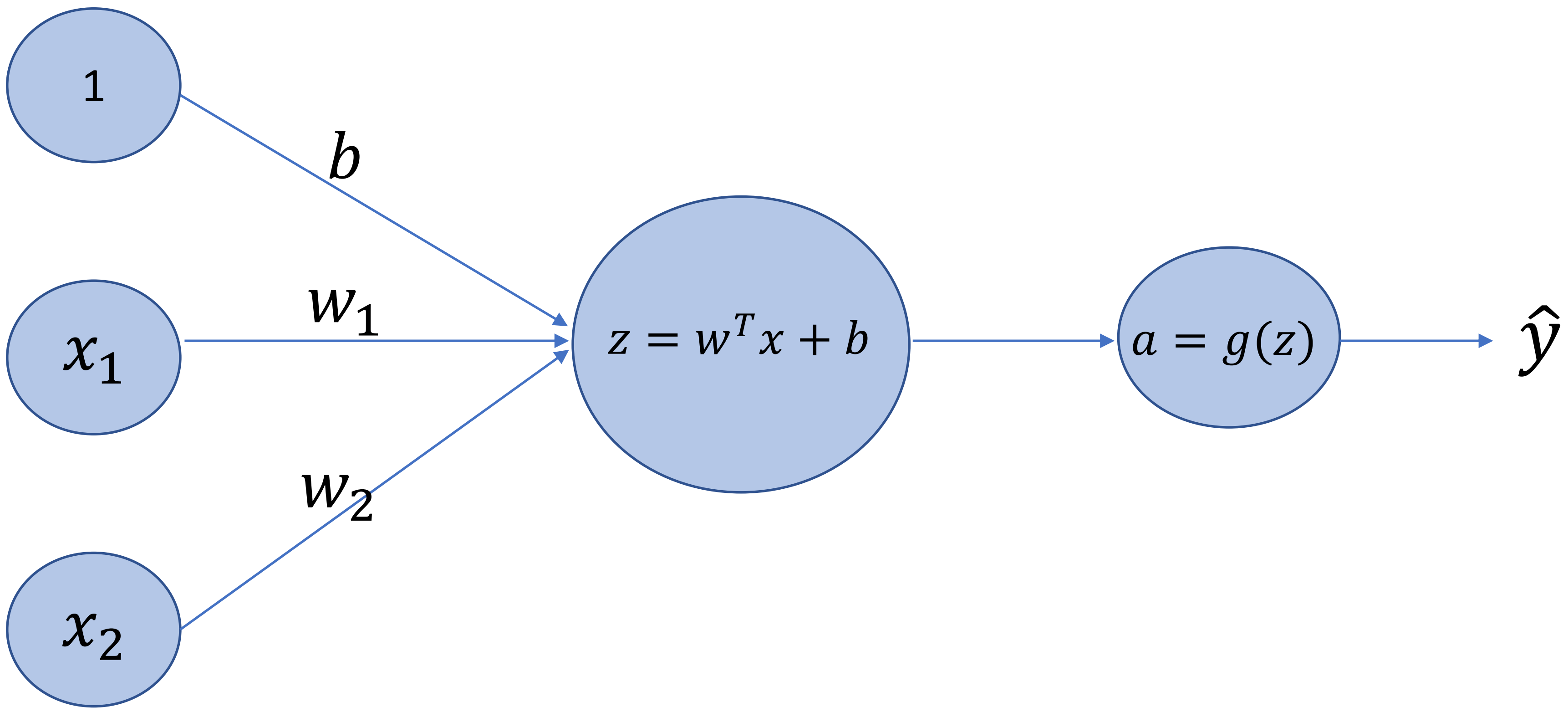
Also called sigmoid function



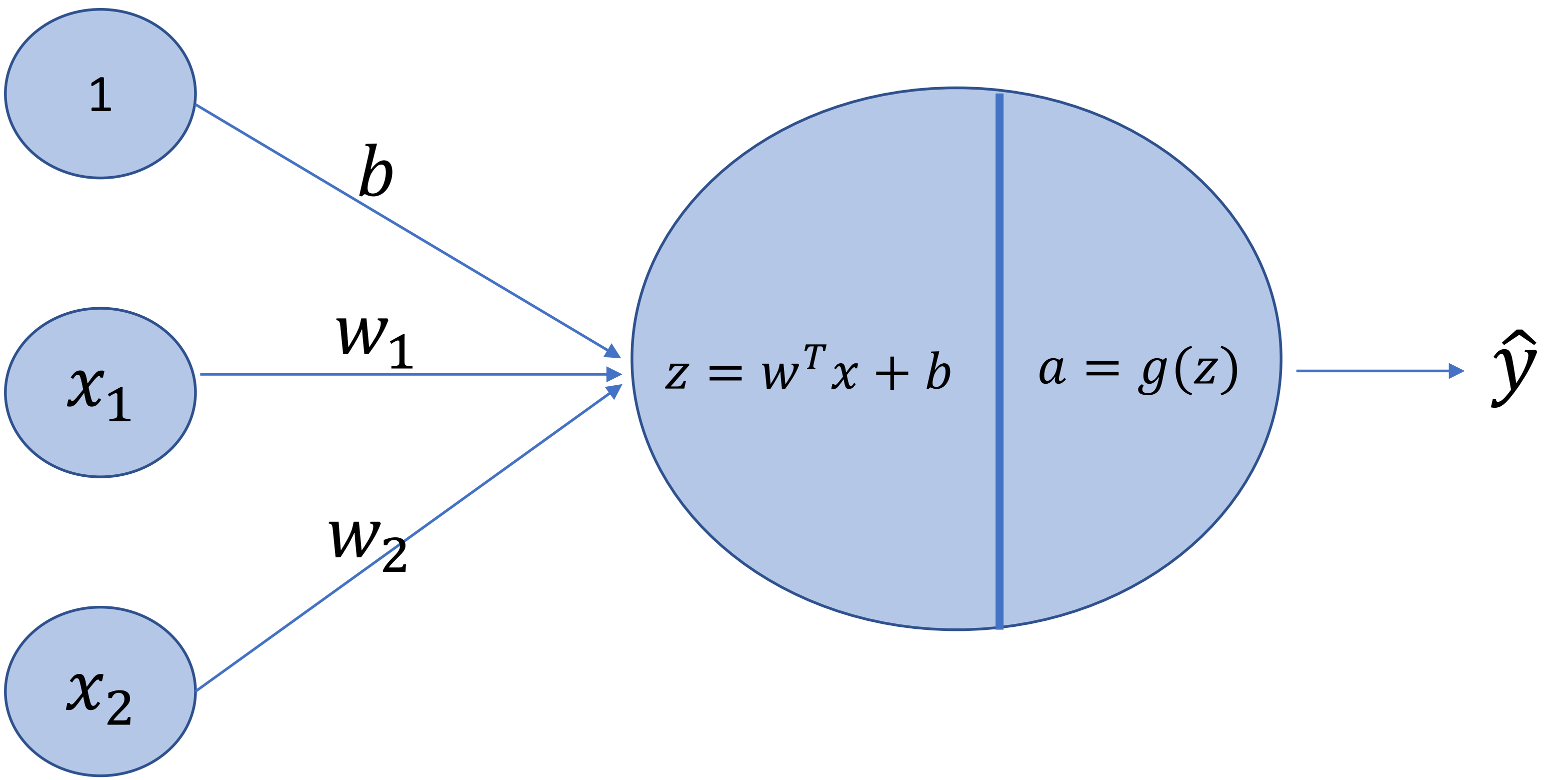
- Map any real number in $[-\infty, +\infty]$ to a real number within [0, 1]

Image source: https://en.wikipedia.org/wiki/Logistic_function

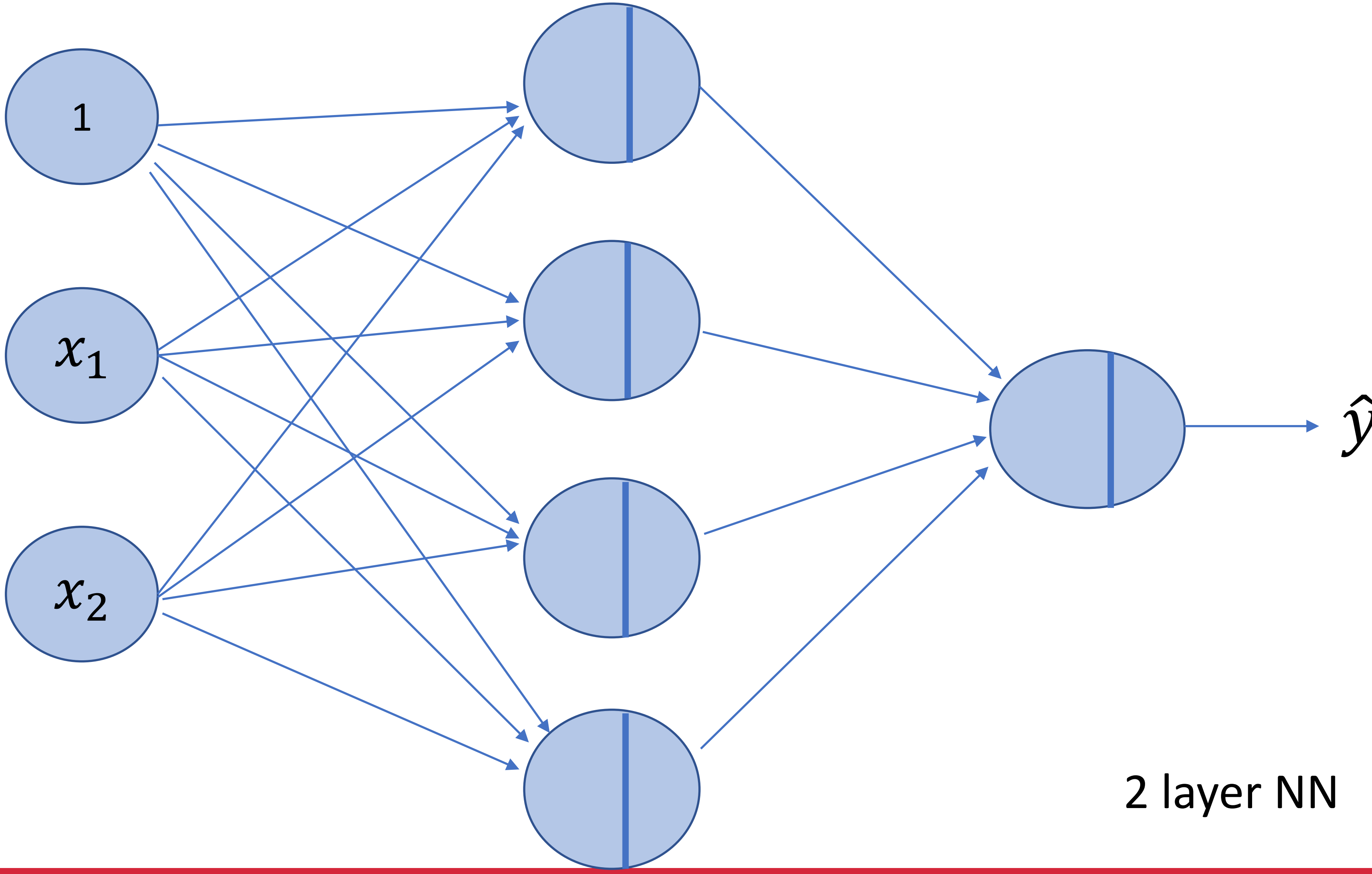
Logistic regression as a neural network



Logistic regression as a neural network

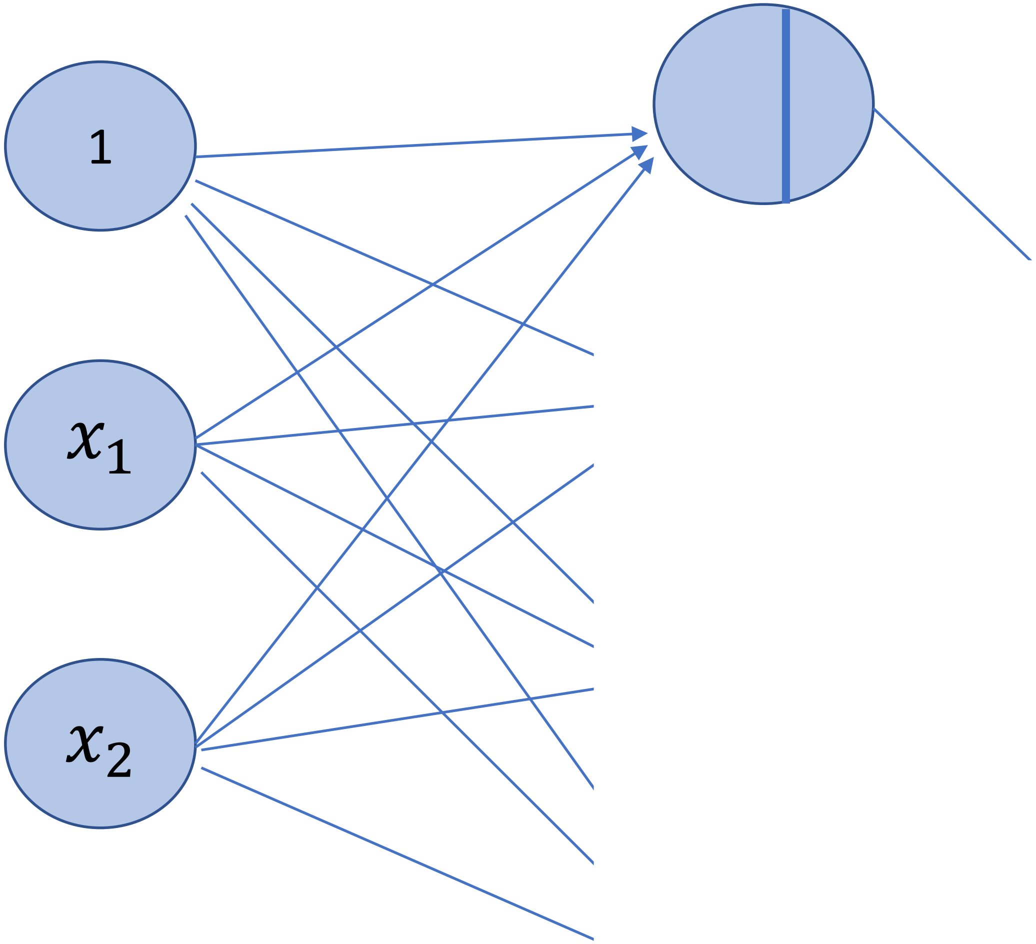


A more general neural network

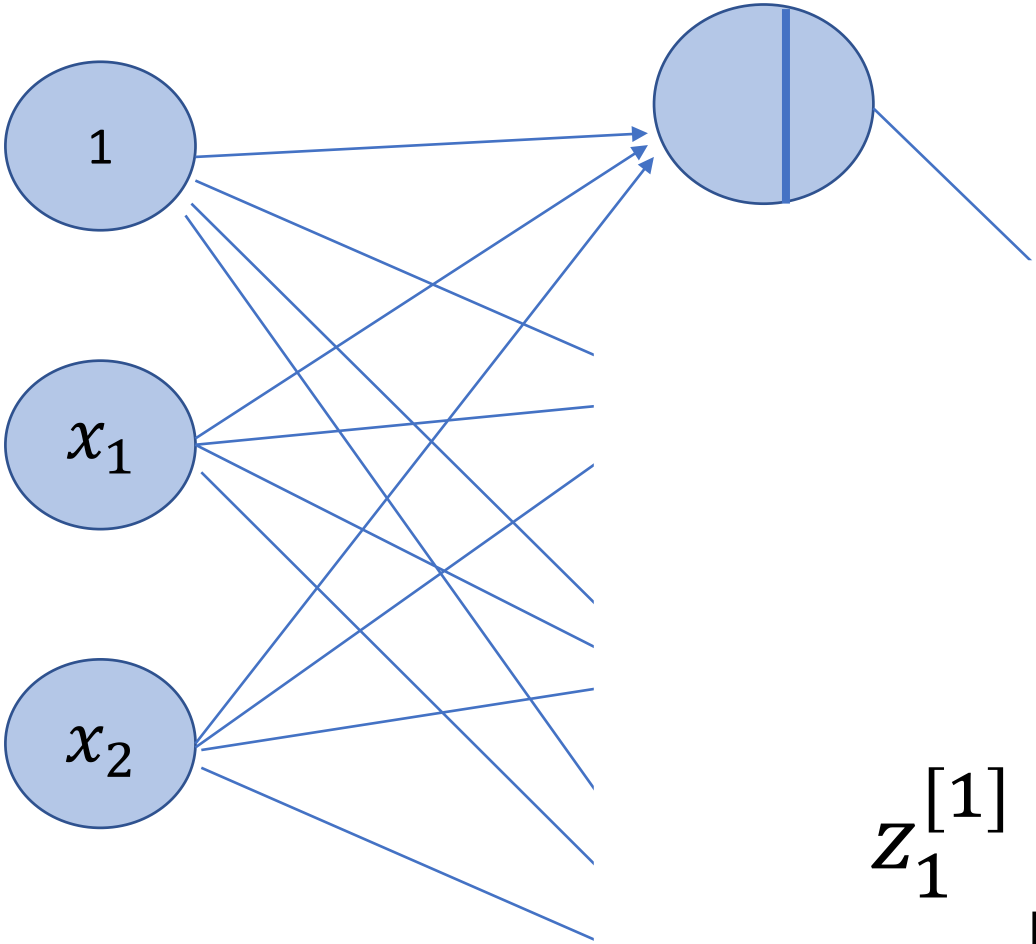


2 layer NN

A more general neural network

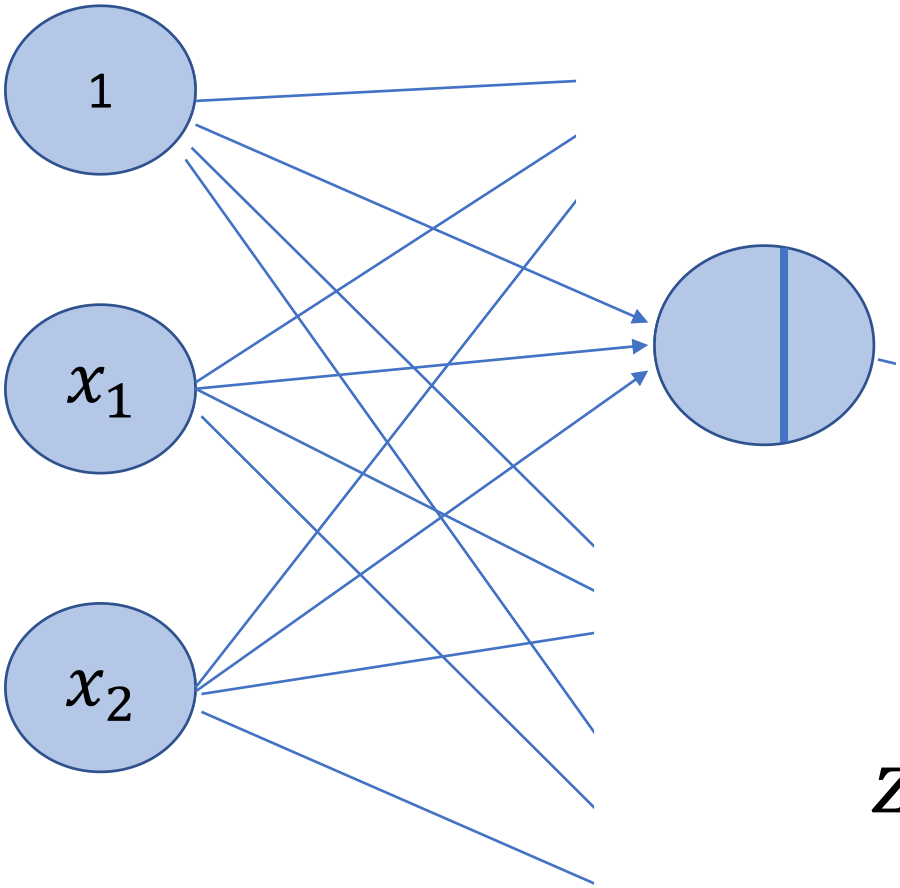


A more general neural network

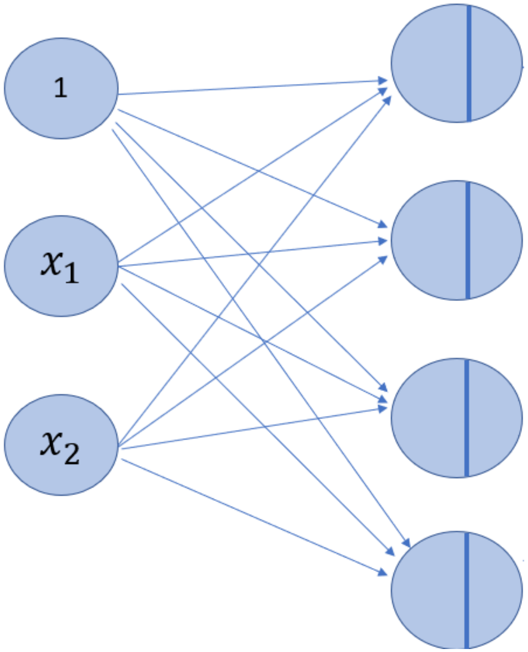


$$z_1^{[1]} = w_1^{[1]}x + b_1^{[1]}$$
$$a_1^{[1]} = g(z_1^{[1]})$$

A more general neural network



$$z_2^{[1]} = w_2^{[1]}x + b_2^{[1]}$$
$$a_2^{[1]} = g(z_2^{[1]})$$



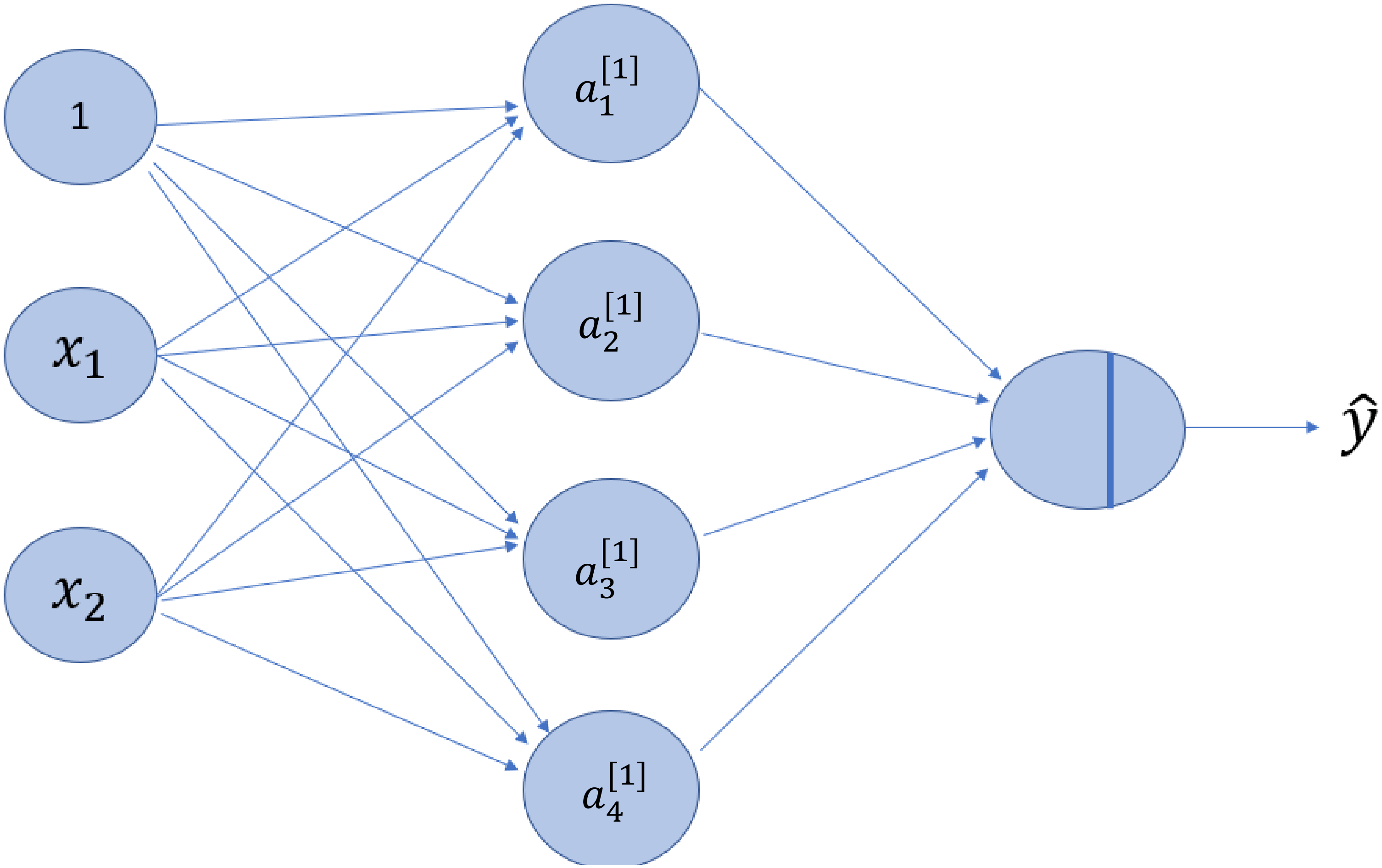
$$z_1^{[1]} = w_1^{[1]}x + b_1^{[1]}, a_1^{[1]} = g(z_1^{[1]})$$

$$z_2^{[1]} = w_2^{[1]}x + b_2^{[1]}, a_2^{[1]} = g(z_2^{[1]})$$

$$z_3^{[1]} = w_3^{[1]}x + b_3^{[1]}, a_3^{[1]} = g(z_3^{[1]})$$

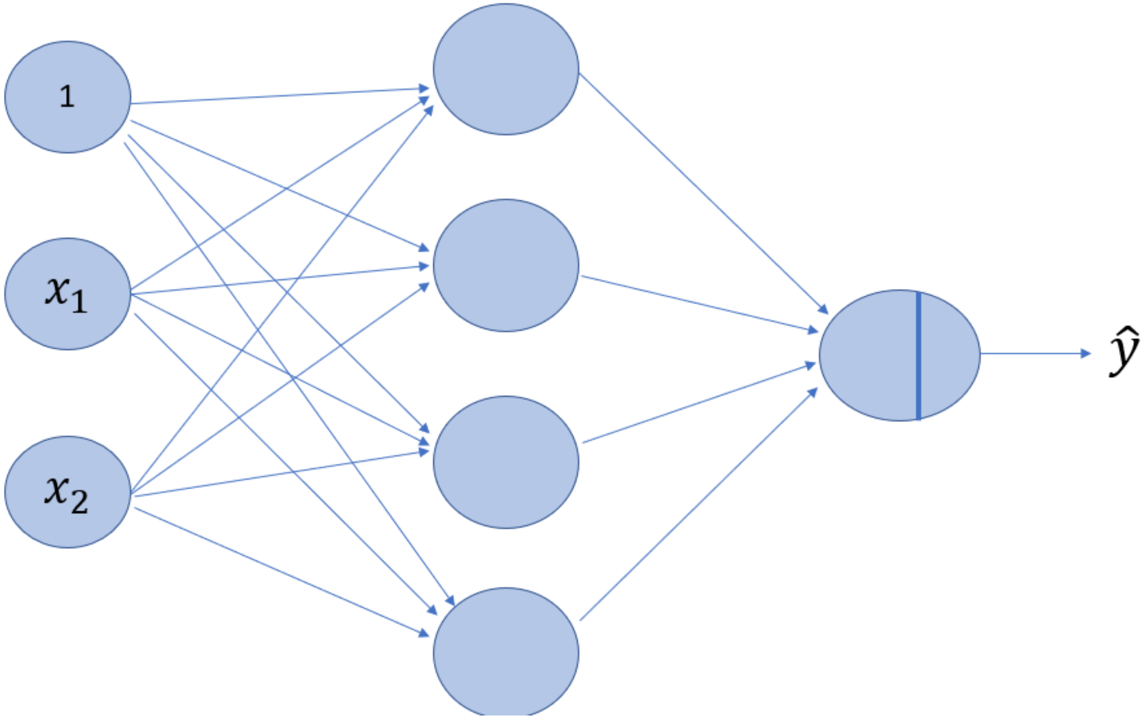
$$z_4^{[1]} = w_4^{[1]}x + b_4^{[1]}, a_4^{[1]} = g(z_4^{[1]})$$

$$z^{[1]} = w^{[1]}x + b^{[1]}, a^{[1]} = g(z^{[1]})$$



$$z^{[2]} = w^{[2]}a^{[1]} + b^{[2]}, a^{[2]} = g(z^{[2]})$$

Forward propagation



$$z^{[1]} = w^{[1]}x + b^{[1]}$$
$$a^{[1]} = g(z^{[1]})$$

$$z^{[2]} = w^{[2]}a^{[1]} + b^{[2]}$$
$$a^{[2]} = g(z^{[2]})$$

$$L = \frac{1}{2}||a^{[2]} - y||^2$$

Training

- Minimizing cost function with respect to all the weights:

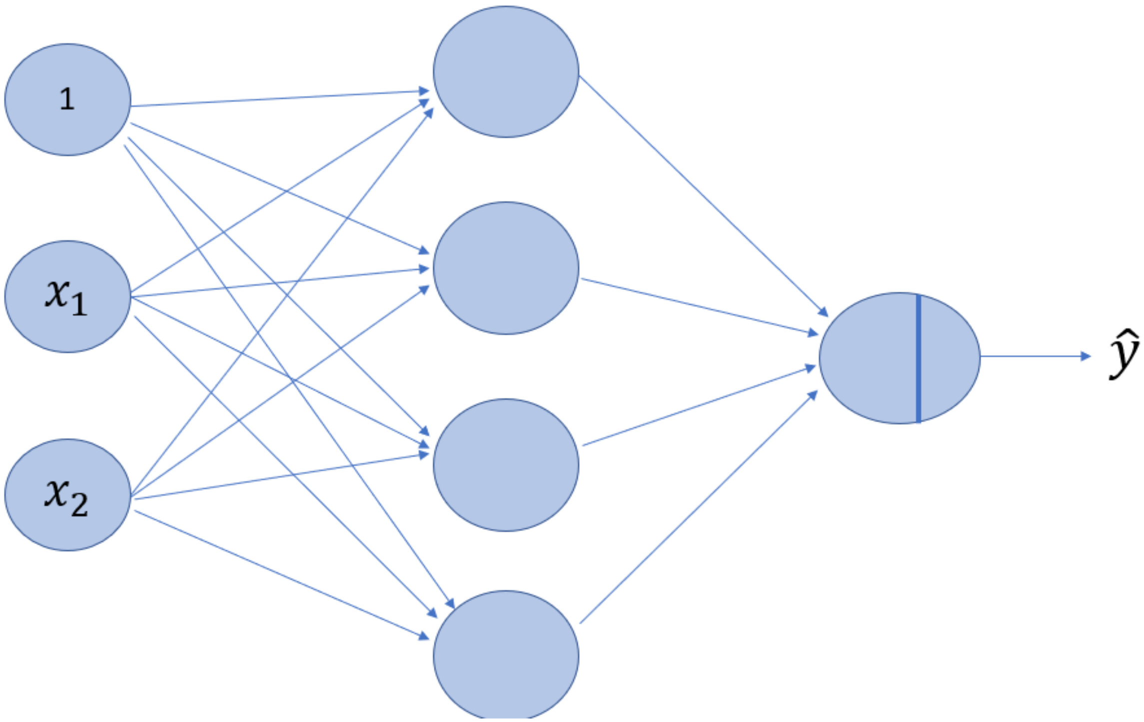
$$L\left(w^{[1]}, b^{[1]}, w^{[2]}, b^{[2]}\right) = \frac{1}{2}||a^{[2]} - y||^2$$

Gradient descent

- Initialize $w^{[1]}, b^{[1]}, w^{[2]}, b^{[2]}$
- While (not converge):
- compute $\frac{\partial L}{\partial w^{[1]}}, \frac{\partial L}{\partial b^{[1]}}, \frac{\partial L}{\partial w^{[2]}}, \frac{\partial L}{\partial b^{[2]}}$
- $w^{[1]} = w^{[1]}$-$\alpha \frac{\partial L}{\partial w^{[1]}}$
- $b^{[1]} = b^{[1]}$-$\alpha \frac{\partial L}{\partial b^{[1]}}$
- $w^{[2]} = w^{[2]}$-$\alpha \frac{\partial L}{\partial w^{[2]}}$
- $b^{[2]} = b^{[2]}$-$\alpha \frac{\partial L}{\partial b^{[2]}}$
- end

Forward propagation

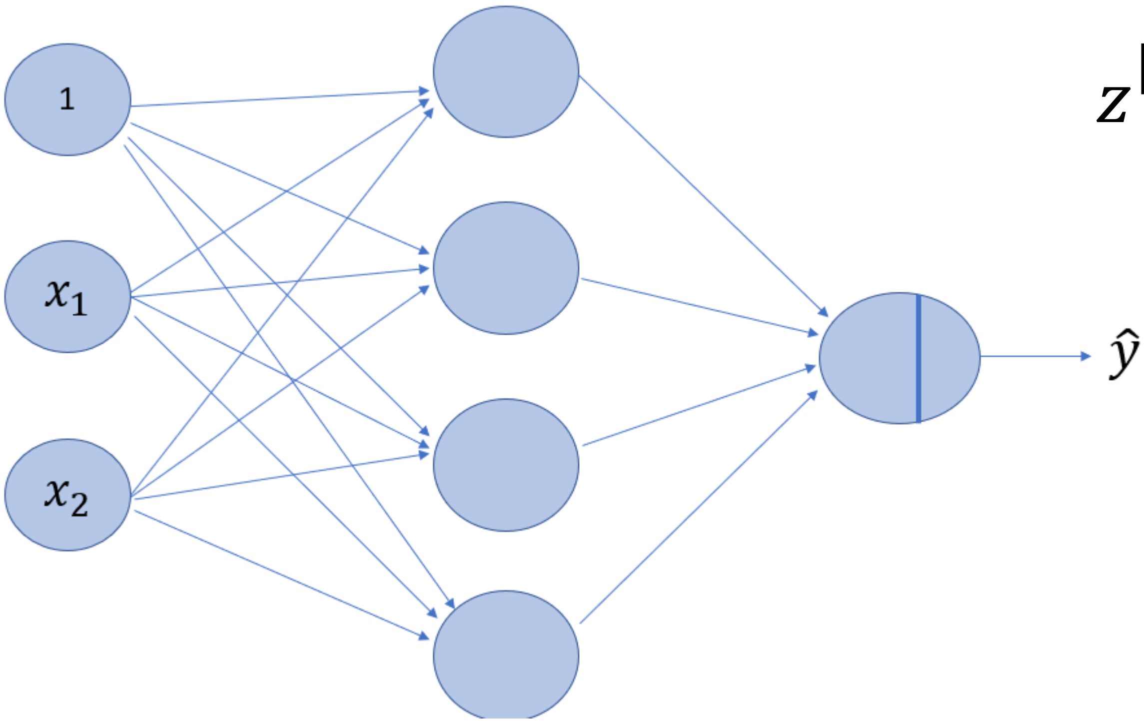
$$z^{[1]} = w^{[1]}x + b^{[1]}$$
$$a^{[1]} = g(z^{[1]})$$



$$z^{[2]} = w^{[2]}a^{[1]} + b^{[2]}$$
$$a^{[2]} = g(z^{[2]})$$

$$L = \frac{1}{2}||a^{[2]} - y||^2$$

A different representation



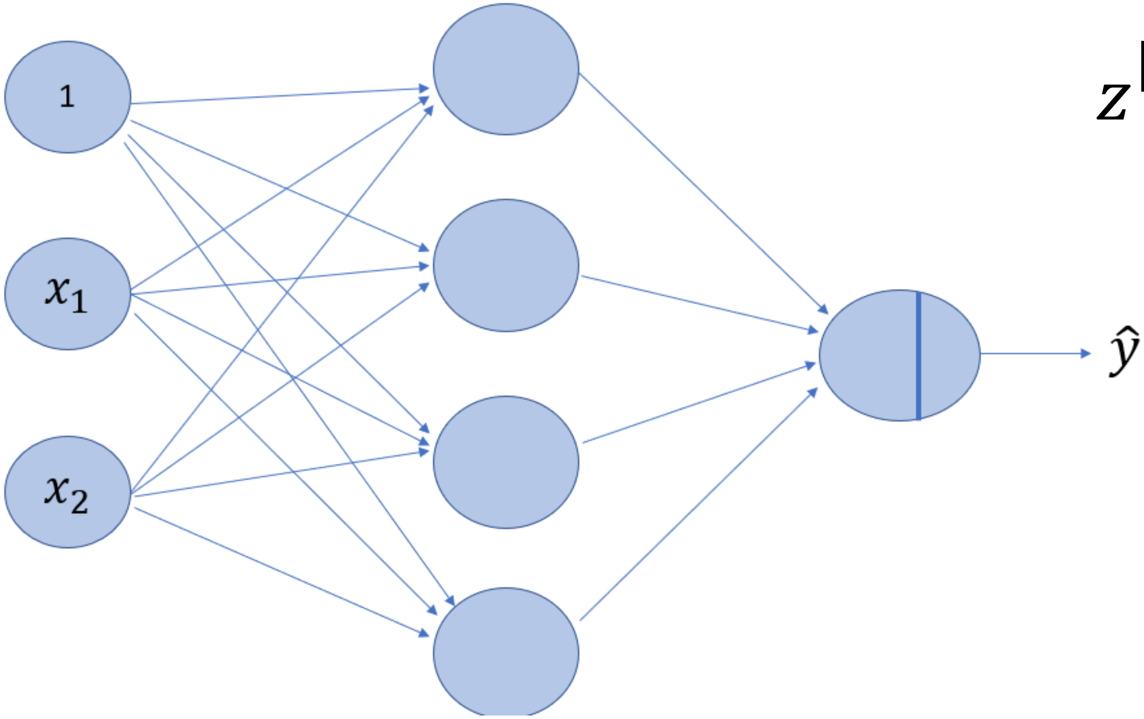
$$z^{[1]} = f_1(w^{[1]}, b^{[1]}, x)$$
$$a^{[1]} = g_1(z^{[1]})$$

$$z^{[2]} = f_2(w^{[2]}, b^{[2]}, a^{[1]})$$
$$a^{[2]} = g_2(z^{[2]})$$
$$L = l(a^{[2]})$$

How to calculate these gradients?

- Back propagation!

Back propagation intuition



$$z^{[1]} = f_1(w^{[1]}, b^{[1]}, x)$$
$$a^{[1]} = g_1(z^{[1]})$$

$$z^{[2]} = f_2(w^{[2]}, b^{[2]}, a^{[1]})$$
$$a^{[2]} = g_2(z^{[2]})$$
$$L = l(a^{[2]})$$

Back propagation intuition

$$z^{[1]} = f_1(w^{[1]}, b^{[1]}, x)$$
$$a^{[1]} = g_1(z^{[1]})$$

$$z^{[2]} = f_2(w^{[2]}, b^{[2]}, a^{[1]})$$
$$a^{[2]} = g_2(z^{[2]})$$
$$L = l(a^{[2]})$$

$$L = l\left(g_2(z^{[2]})\right)$$
$$= l\left(g_2(f_2(w^{[2]}, b^{[2]}, a^{[1]}))\right)$$
$$= l\left(g_2\left(f_2(w^{[2]}, b^{[2]}, g_1(z^{[1]}))\right)\right)$$
$$= l\left(g_2(f_2(w^{[2]}, b^{[2]}, g_1(f_1(w^{[1]}, b^{[1]}, x))))\right)$$

Compute gradients

$$L = l\left(g_2\left(f_2(w^{[2]}, b^{[2]}, g_1(f_1(w^{[1]}, b^{[1]})))\right)\right)$$

How to compute $\frac{\partial L}{\partial w^{[2]}}$?

$$z^{[1]} = f_1(w^{[1]}, b^{[1]})$$
$$a^{[1]} = g_1(z^{[1]})$$

Chain rule!

$$z^{[2]} = f_2(w^{[2]}, b^{[2]}, a^{[1]})$$
$$a^{[2]} = g_2(z^{[2]})$$

$$L = l(a^{[2]})$$

Compute gradients

$$L = l\left(g_2(f_2(w^{[2]}, b^{[2]}, g_1(f_1(w^{[1]}, b^{[1]})\,))\,)\right)$$

How to compute $\frac{\partial L}{\partial w^{[2]}}$?

Chain rule!

Compute gradients

$$L = l\left(g_2(f_2(w^{[2]}, b^{[2]}, g_1(f_1(w^{[1]}, b^{[1]})\,))\,)\right)$$

How to compute $\frac{\partial L}{\partial w^{[2]}}$?

Chain rule!

$$\frac{\partial L}{\partial w^{[2]}} = \frac{\partial L}{\partial g_2}\frac{\partial g_2}{\partial f_2}\frac{\partial f_2}{\partial w^{[2]}}$$

Compute gradients

$$L = l\left(g_2(f_2(w^{[2]}, b^{[2]}, g_1(f_1(w^{[1]}, b^{[1]})))\right)$$

How to compute $\frac{\partial L}{\partial w^{[2]}}$?

Chain rule!

$$\frac{\partial L}{\partial w^{[2]}} = \frac{\partial L}{\partial g_2} \frac{\partial g_2}{\partial f_2} \frac{\partial f_2}{\partial w^{[2]}}$$

$$z^{[1]} = w^{[1]}x + b^{[1]}$$

$$a^{[1]} = g_1\,(z^{[1]})$$

$$z^{[2]} = w^{[2]}a^{[1]} + b^{[2]}$$

$$a^{[2]} = g_2(z^{[2]})$$

$$L = \frac{1}{2}||a^{[2]} - y||^2$$

Similarly, for the other three

$$L = l\left(g_2(f_2(w^{[2]}, b^{[2]}, g_1(f_1(w^{[1]}, b^{[1]})\,))\,)\right)$$

Back Propagation

$$\frac{\partial L}{\partial w^{[1]}} = \frac{\partial L}{\partial g_2}\frac{\partial g_2}{\partial f_2}\frac{\partial f_2}{\partial g_1}\frac{\partial g_1}{\partial f_1}\frac{\partial f_1}{\partial w^{[1]}}$$

$$x,\ w^{[1]},\ b^{[1]} \rightarrow z^{[1]} = f_1(w^{[1]}, b^{[1]}) \rightarrow a^{[1]} = g_1(z^{[1]}) \rightarrow z^{[2]} = f_2(w^{[2]}, a^{[1]}, b^{[2]}) \rightarrow a^{[2]} = g_2(z^{[2]}) \rightarrow L(a^{[2]})$$

2018 Turing Award winners

From left to right: Yann LeCun | Photo: Facebook; Geoffrey Hinton | Photo: Google; Yoshua Bengio | Photo: Botler AI

Godfathers of AI

https://www.theverge.com/2019/3/27/18280665/ai-godfathers-turing-award-2018-yoshua-bengio-geoffrey-hinton-yann-lecun

https://amturing.acm.org/

Geoffrey Hinton

Backpropagation: In a 1986 paper, "Learning Internal Representations by Error Propagation," co-authored with Dav Rumelhart and Ronald Williams, Hinton demonstrated that th backpropagation algorithm allowed neural nets to discover the own internal representations of data, making it possible to use neural nets to solve problems that had previously been though beyond their reach. The backpropagation algorithm is standar most neural networks today.

Boltzmann Machines: In 1983, with Terrence Sejnowski, H invented Boltzmann Machines, one of the first neural networ capable of learning internal representations in neurons that w not part of the input or output.

Improvements to convolutional neural networks: In 2 with his students, Alex Krizhevsky and Ilya Sutskever, Hinton improved convolutional neural networks using rectified linear neurons and dropout regularization. In the prominent ImageN competition, Hinton and his students almost halved the error for object recognition and reshaped the computer vision field.

Yann LeCun

Convolutional neural networks: In the 1980s, LeCun developed convolutional neural networks, a foundational principle in the field, which, among other advantages, have been essential in making deep learning more efficient. In the late 1980s, while working at the University of Toronto and Bell Labs, LeCun was the first to train a convolutional neural network system on images of handwritten digits. Today, convolutional neural networks are an industry standard in computer vision, as well as in speech recognition, speech synthesis, image synthesis, and natural language processing. They are used in a wide variety of applications, including autonomous driving, medical image analysis, voice-activated assistants, and information filtering.

Improving backpropagation algorithms: LeCun proposed an early version of the backpropagation algorithm (backprop), and gave a clean derivation of it based on variational principles. His work to speed up backpropagation algorithms included describing two simple methods to accelerate learning time.

Broadening the vision of neural networks: LeCun is also credited with developing a broader vision for neural networks as a computational model for a wide range of tasks, introducing in early work a number of concepts now fundamental in AI. For example, in the context of recognizing images, he studied how hierarchical feature representation can be learned in neural networks—a concept that is now routinely used in many recognition tasks. Together with Léon Bottou, he proposed the idea, used in every modern deep learning software, that learning systems can be built as complex networks of modules where backpropagation is performed through automatic differentiation. They also proposed deep learning architectures that can manipulate structured data, such as graphs.

https://amturing.acm.org/