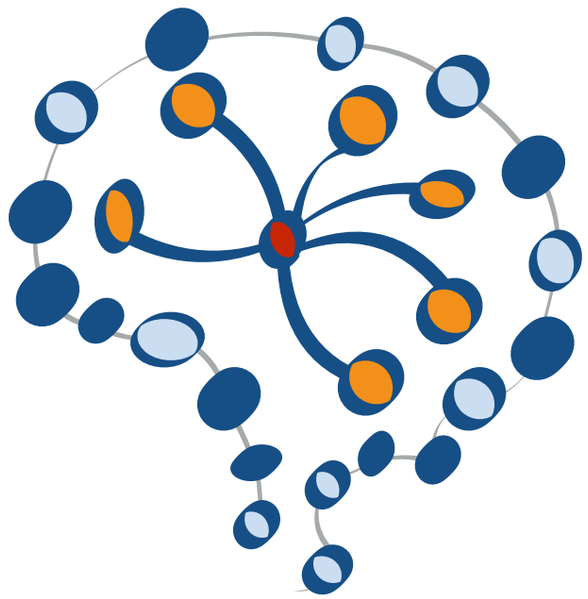
STAT 453: Introduction to Deep Learning and Generative Models

Sebastian Raschka

http://stat.wisc.edu/~sraschka/teaching



Lecture 02

A Brief Summary of the History of Neural Networks and Deep Learning

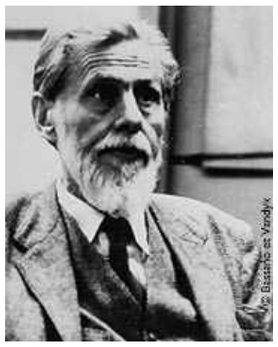
Lecture Topics

1. **Artificial neurons**
2. Multilayer neural networks
3. Deep learning
4. The DL hardware & software landscape
5. Current research trends

Neural Networks and Deep Learning -- A Timeline

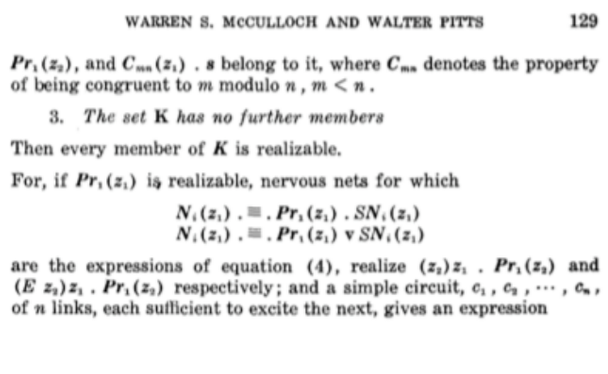
McCulloch & Pitt's neuron model (1943)

McCulloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, *5*(4), 115-133.



Warren McCulloch

Walter Pitts



WARREN S. McCULLOCH AND WALTER PITTS 129

$Pr_1(z_2)$, and $C_{mn}(z_1)$. s belong to it, where C_{mn} denotes the property of being congruent to m modulo n, $m < n$.

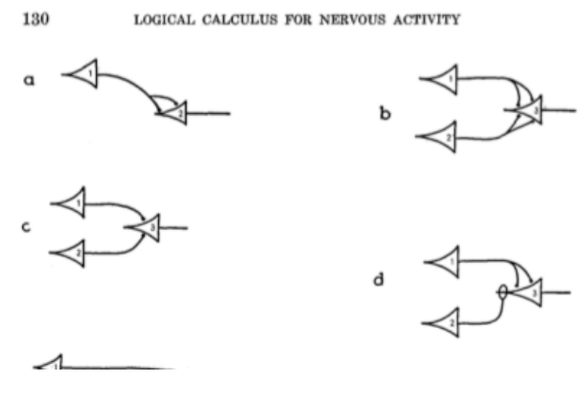
3. *The set* K *has no further members*

Then every member of K is realizable.

For, if $Pr_1(z_1)$ is realizable, nervous nets for which

$$N_1(z_1) \cdot \equiv \cdot Pr_1(z_1) \cdot SN_1(z_1)$$
$$N_1(z_1) \cdot \equiv \cdot Pr_1(z_1) \vee SN_1(z_1)$$

are the expressions of equation (4), realize $(z_2)z_1 \cdot Pr_1(z_2)$ and $(E\ z_2)z_1 \cdot Pr_1(z_2)$ respectively; and a simple circuit, $c_1, c_2, \cdots, c_n$, of n links, each sufficient to excite the next, gives an expression



130 LOGICAL CALCULUS FOR NERVOUS ACTIVITY

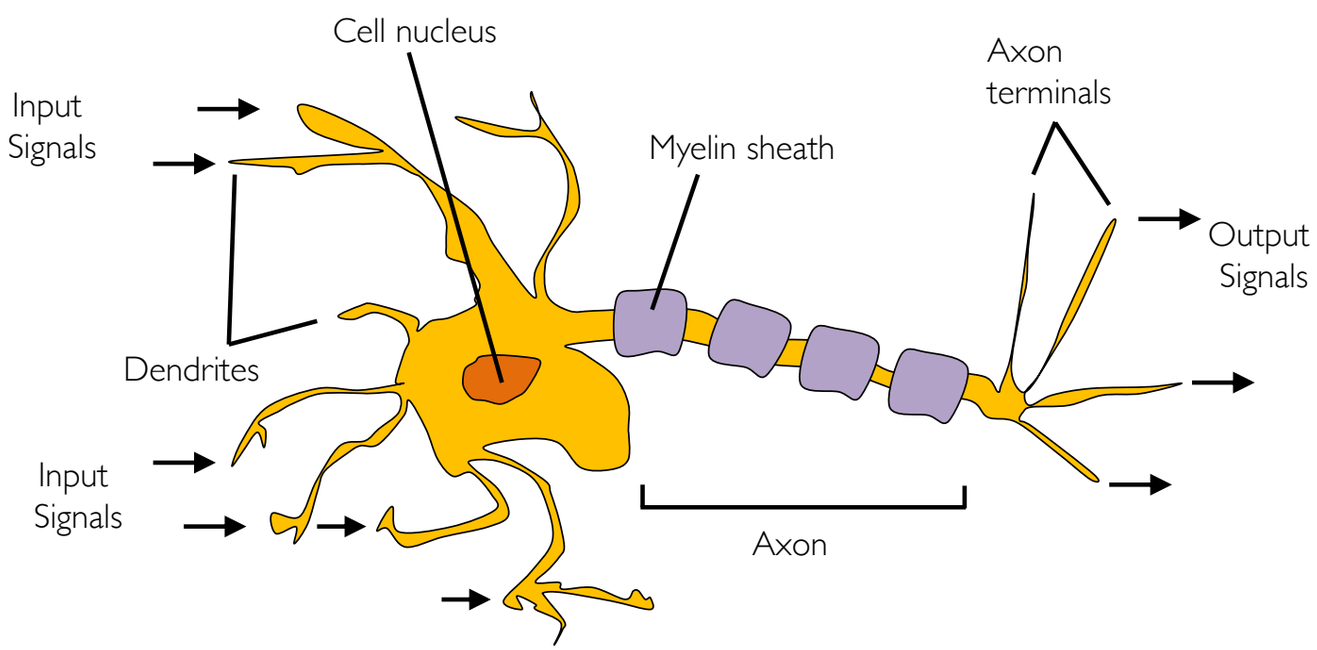
a

b

c

d

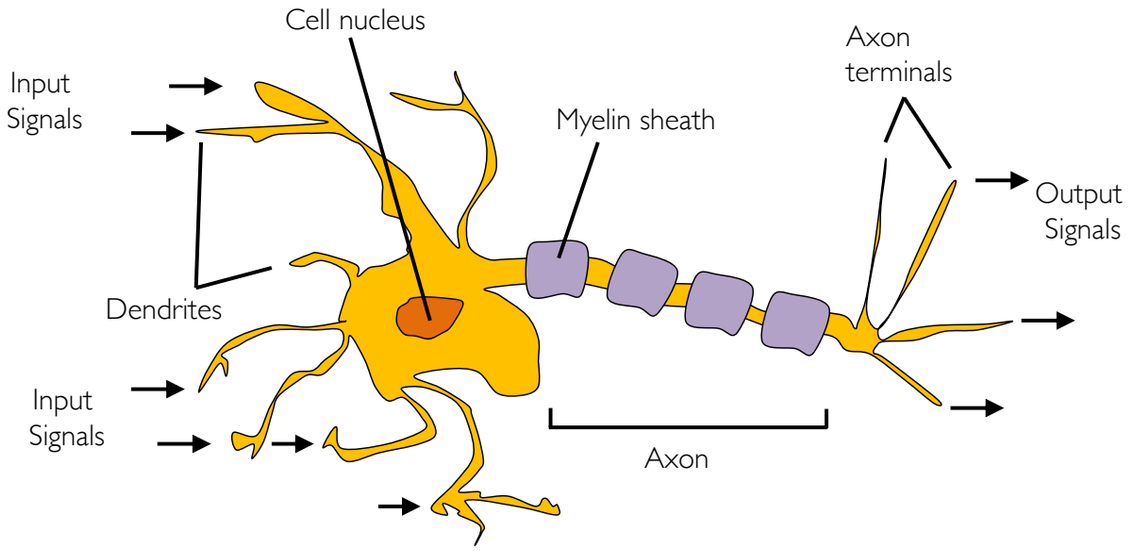
Image Source: https://www.i-programmer.info/babbages-bag/325-mcculloch-pitts-neural-networks.html

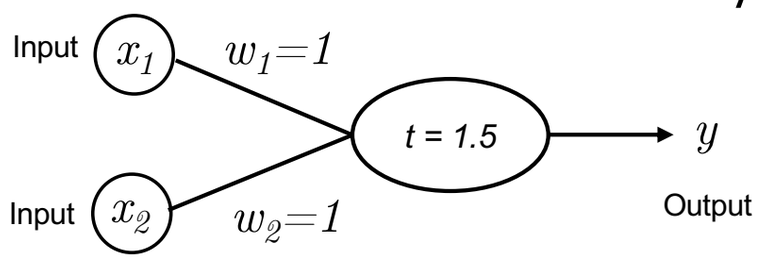


Mathematical formulation of a biological neuron,
could solve AND, OR, NOT problems

McCulloch & Pitt's neuron model (1943)

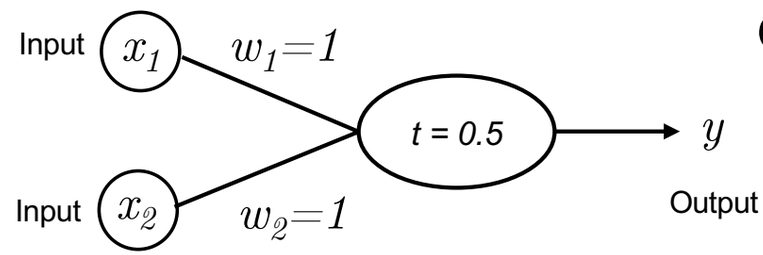
Mathematical formulation of a biological neuron,
could solve AND, OR, NOT problems





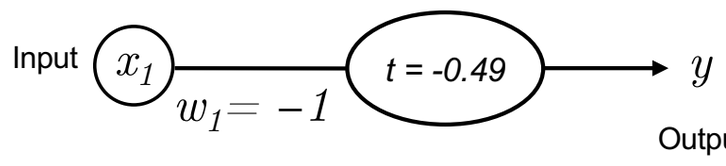
AND

x_1	x_2	y
0	0	0
0	1	0
1	0	0
1	1	1



OR

x_1	x_2	y
0	0	0
0	1	1
1	0	1
1	1	1



NOT

x_1	y
0	1
1	0

Neural Networks and Deep Learning -- A Timeline

Frank Rosenblatt's Perceptron (1957)

A learning algorithm for the neuron model

Rosenblatt, F. (1957). *The perceptron, a perceiving and recognizing automaton Project Para*. Cornell Aeronautical Laboratory.

Inspired by

McCulloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, *5*(4), 115-133.

Hebb, D. O. (1949). The organization of behavior. A neuropsychological theory.

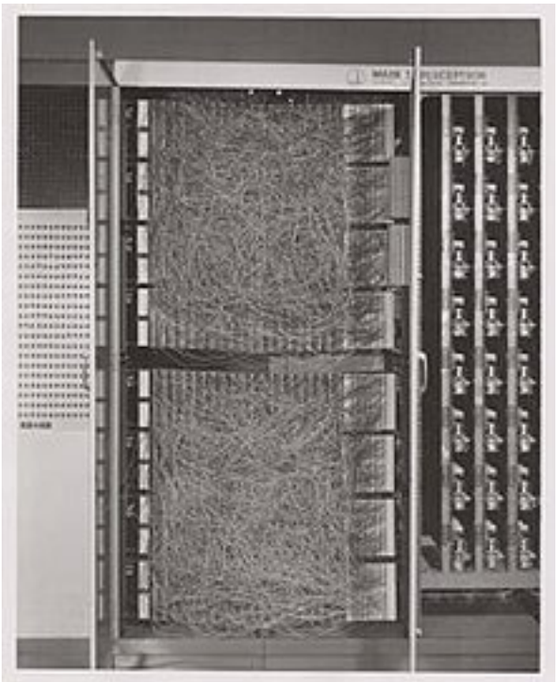
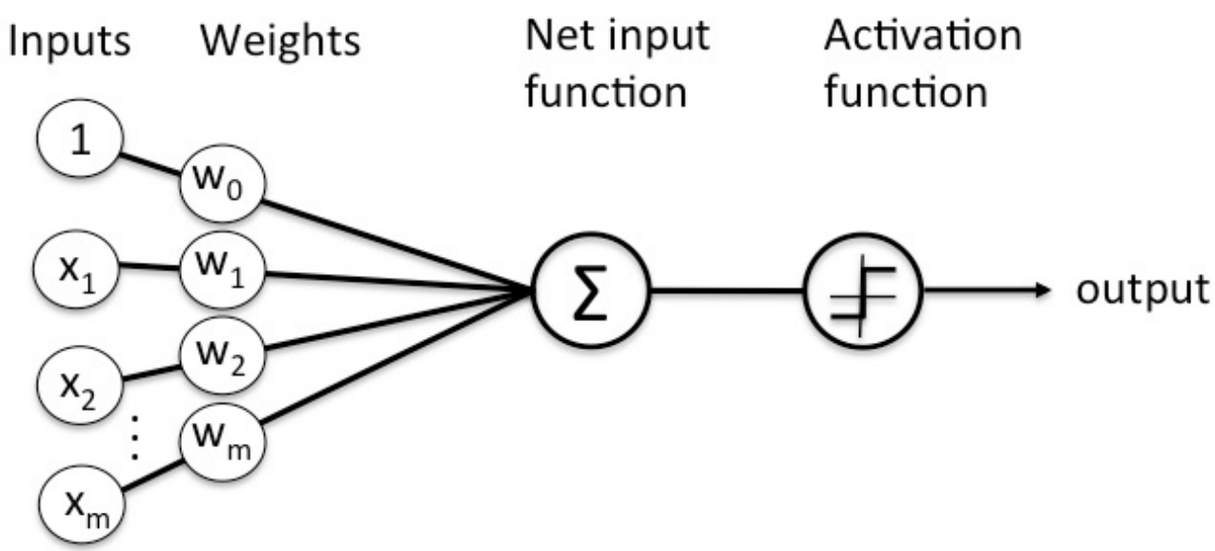


Image source: https://en.wikipedia.org/wiki/Perceptron#/media/File:Mark_I_perceptron.jpeg



Neural Networks and Deep Learning -- A Timeline

Widrow and Hoff's ADALINE (1960)

A nicely differentiable neuron model

Widrow, B., & Hoff, M. E. (1960). *Adaptive switching circuits* (No. TR-1553-1). Stanford Univ Ca Stanford Electronics Labs.

Widrow, B. (1960). *Adaptive" adaline" Neuron Using Chemical" memistors."*.

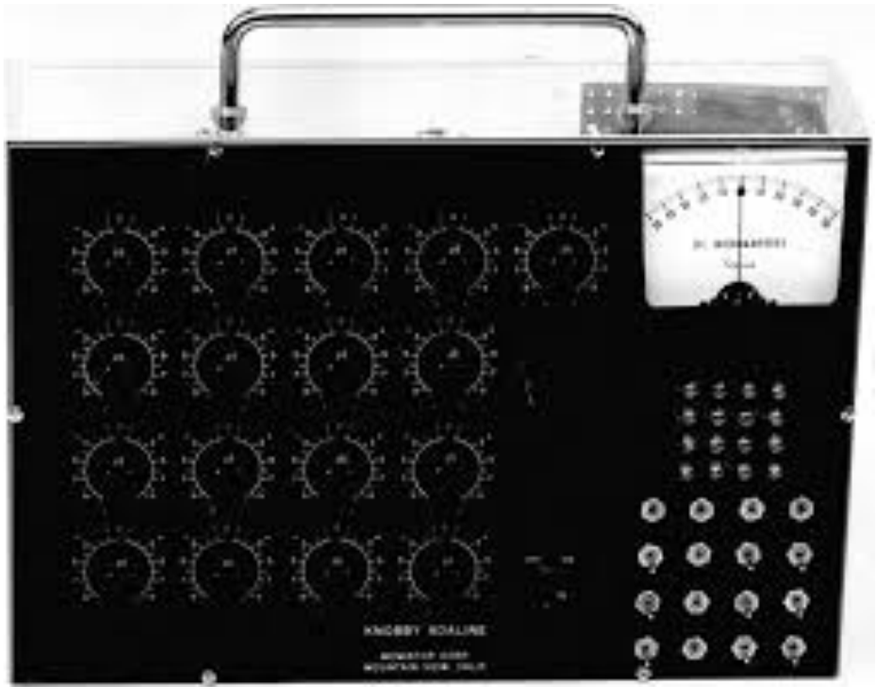
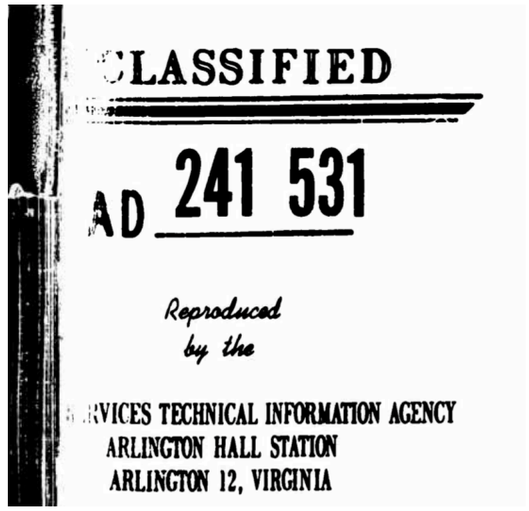


Image source: https://www.researchgate.net/profile/Alexander_Magoun2/publication/265789430/figure/fig2/AS:392335251787780@1470551421849/ADALINE-An-adaptive-linear-neuron-Manually-adapted-synapses-Designed-and-built-by-Ted.png


CLASSIFIED

AD 241 531

Reproduced by the

SERVICES TECHNICAL INFORMATION AGENCY
ARLINGTON HALL STATION
ARLINGTON 12, VIRGINIA

THIS REPORT HAS BEEN DELIMITED AND CLEARED FOR PUBLIC RELEASE UNDER DOD DIRECTIVE 5200.20 AND NO RESTRICTIONS ARE IMPOSED UPON ITS USE AND DISCLOSURE.

DISTRIBUTION STATEMENT A

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

Widrow and Hoff's ADALINE (1960)

A nicely differentiable neuron model

A "single layer neural network"

Widrow, B., & Hoff, M. E. (1960). *Adaptive switching circuits* (No. TR-1553-1). Stanford Univ Ca Stanford Electronics Labs.

Widrow, B. (1960). *Adaptive" adaline" Neuron Using Chemical" memistors."*.

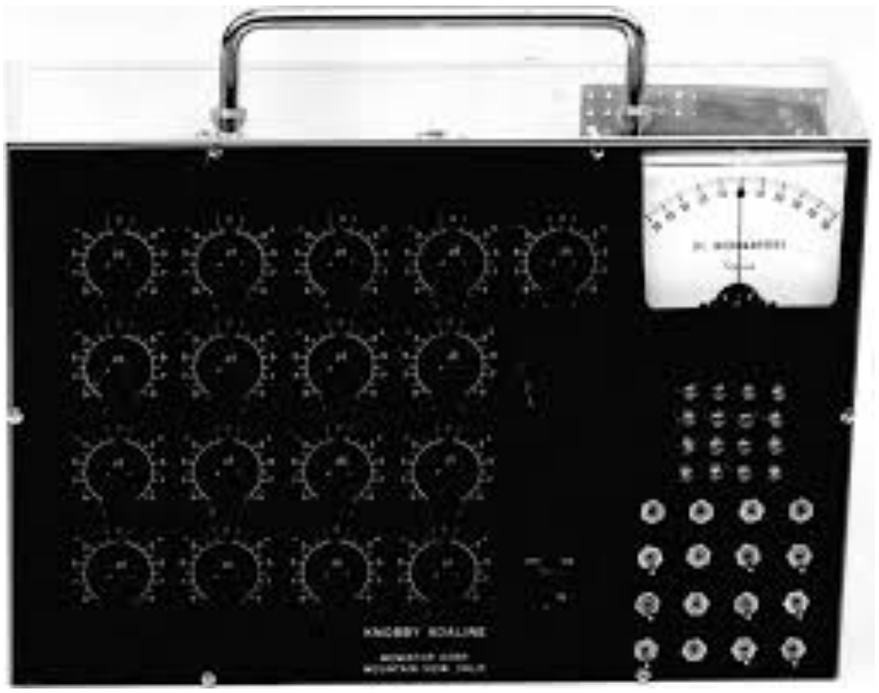
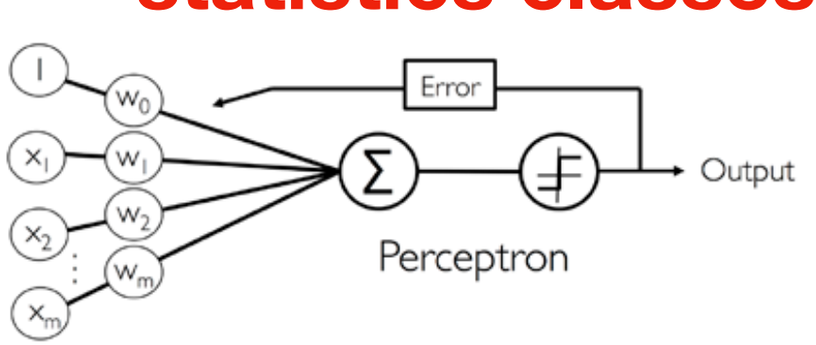
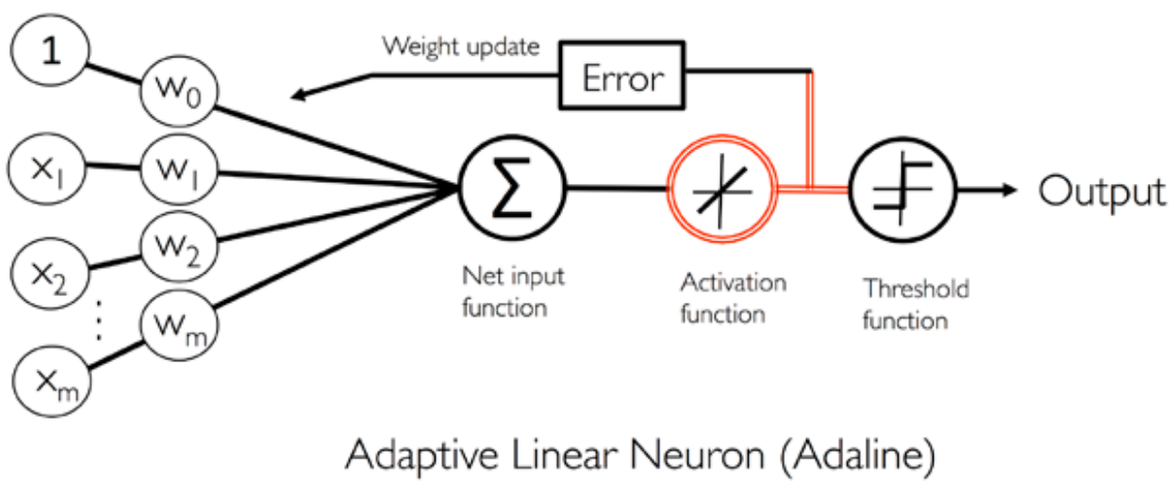


Image source: https://www.researchgate.net/profile/Alexander_Magoun2/publication/265789430/figure/fig2/AS:392335251787780@1470551421849/ADALINE-An-adaptive-linear-neuron-Manually-adapted-synapses-Designed-and-built-by-Ted.png

Does the figure below remind you of something you know from other statistics classes?





Source: Raschka and Mirjalily (2019). *Python Machine Learning, 3rd Edition*

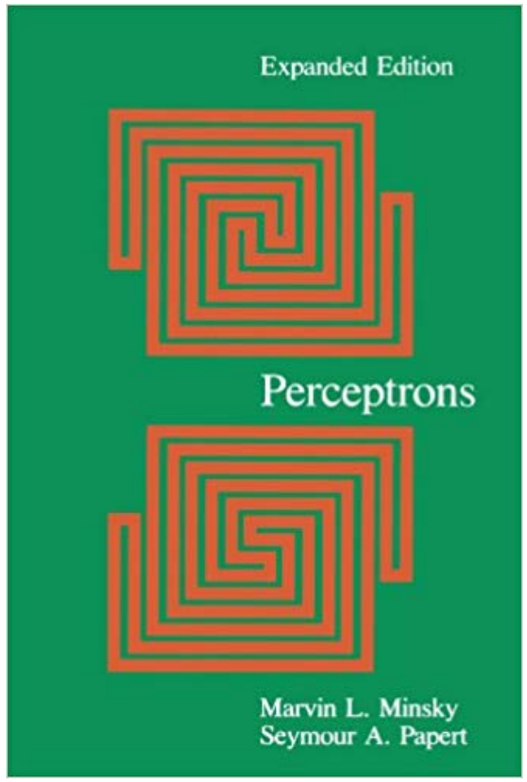
Neural Networks and Deep Learning -- A Timeline

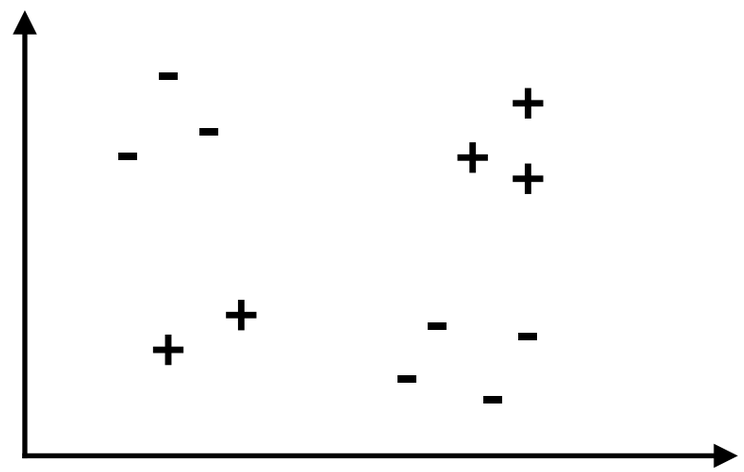
Minsky and Pappert (1969): "Perceptrons" book

=> Problem: Perceptrons (and ADALINEs) could not solve XOR problems!

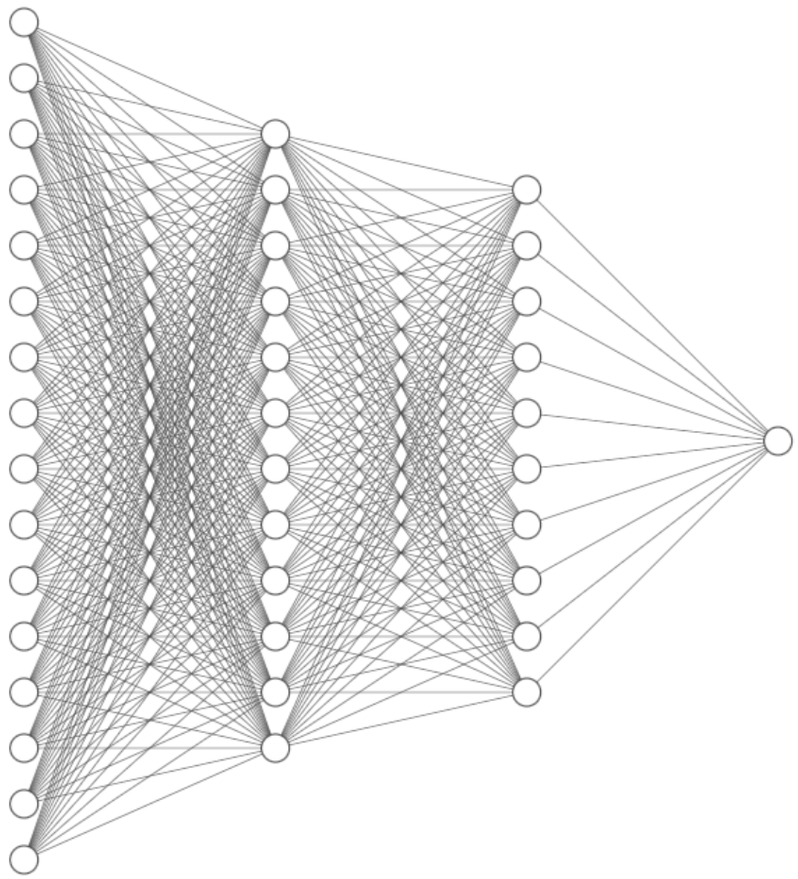
=> Neurons, a dead end? Start of the first "AI Winter"

Minsky, M. L., & Papert, S. A. (1969). Perceptrons: an introduction to computational geometry. *MA: MIT Press, Cambridge*.



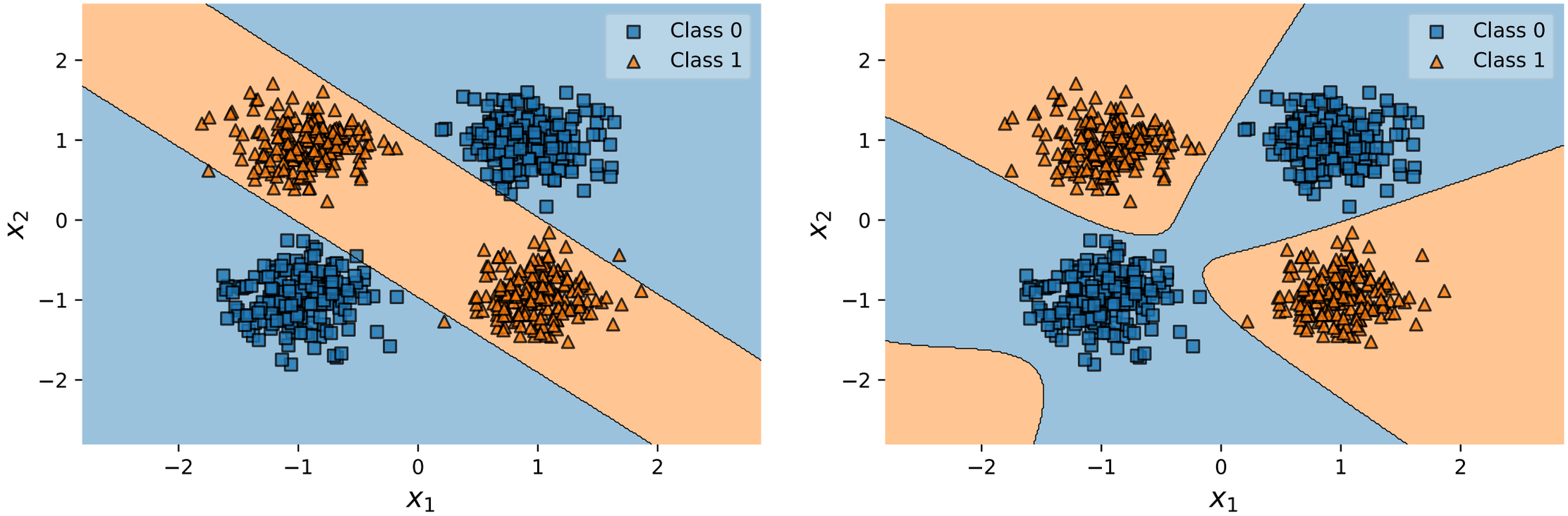


Training neural networks with multiple layers

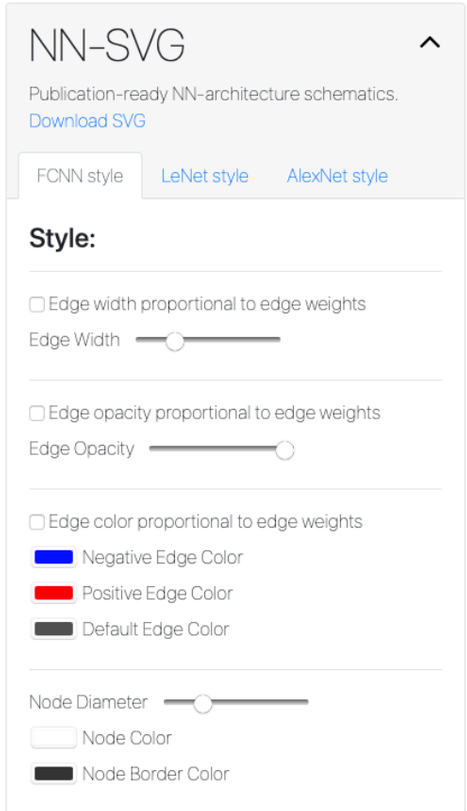


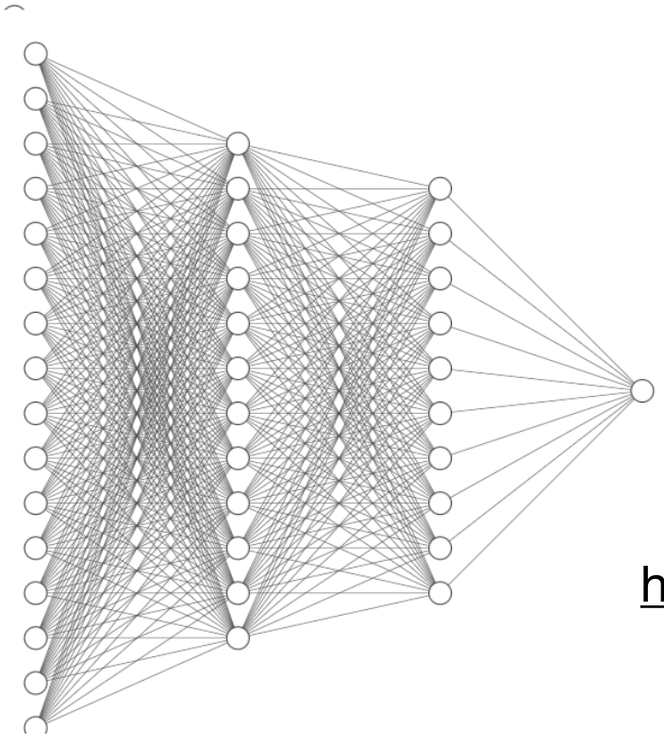
1. Artificial neurons
2. **Multilayer neural networks**
3. Deep learning
4. The DL hardware & software landscape
5. Current research trends

Multilayer Neural Networks Can Solve XOR Problems



Decision boundaries of two different multilayer perceptrons on simulated data solving the XOR problem

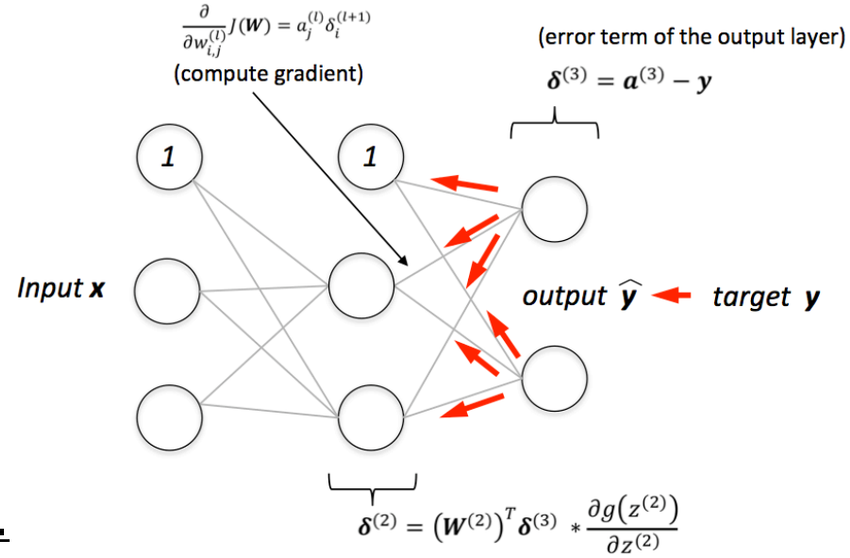




https://alexlenail.me/NN-SVG/index.html

Multilayer Neural Networks -- A Timeline

- Solution to the XOR problem: hidden layers and non-linear activation functions
- New problem: Hard to train
- Solution: Backpropagation



Note that backpropagation has been independently formulated many times ...

http://people.idsia.ch/~juergen/who-invented-backpropagation.

Rumelhart and Hinton (1986) formulated it independently and then showed that it really works (and formed the basis of all consequent neural network and DL progress):

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, *323*(6088), 533.

Also, it was later shown that "neural nets" are universal function approximators

Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural networks*, *2*(5), 359-366.

Neural Networks and Deep Learning -- A Timeline

Rumelhart and Hinton (1986) formulated it independently and then showed that it really works (and formed the basis of all consequent neural network and DL progress):

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, *323*(6088), 533.

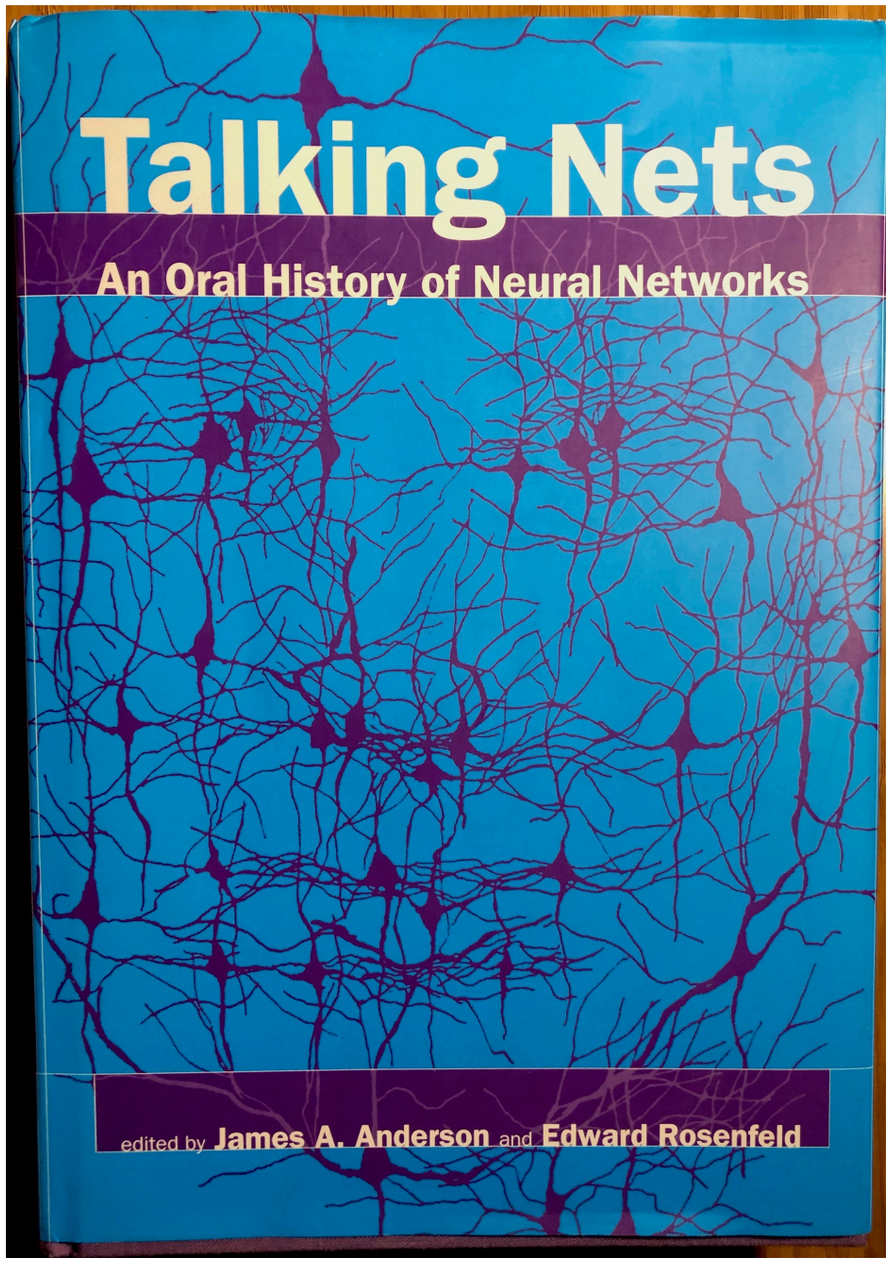
In late 1985, I actually had a deal with Dave Rumelhart that I would write a short paper about backpropagation, which was his idea, and he would write a short paper about autoencoders, which was my idea. It was always better to have someone who didn't come up with the idea write the paper because he could say more clearly what was important.

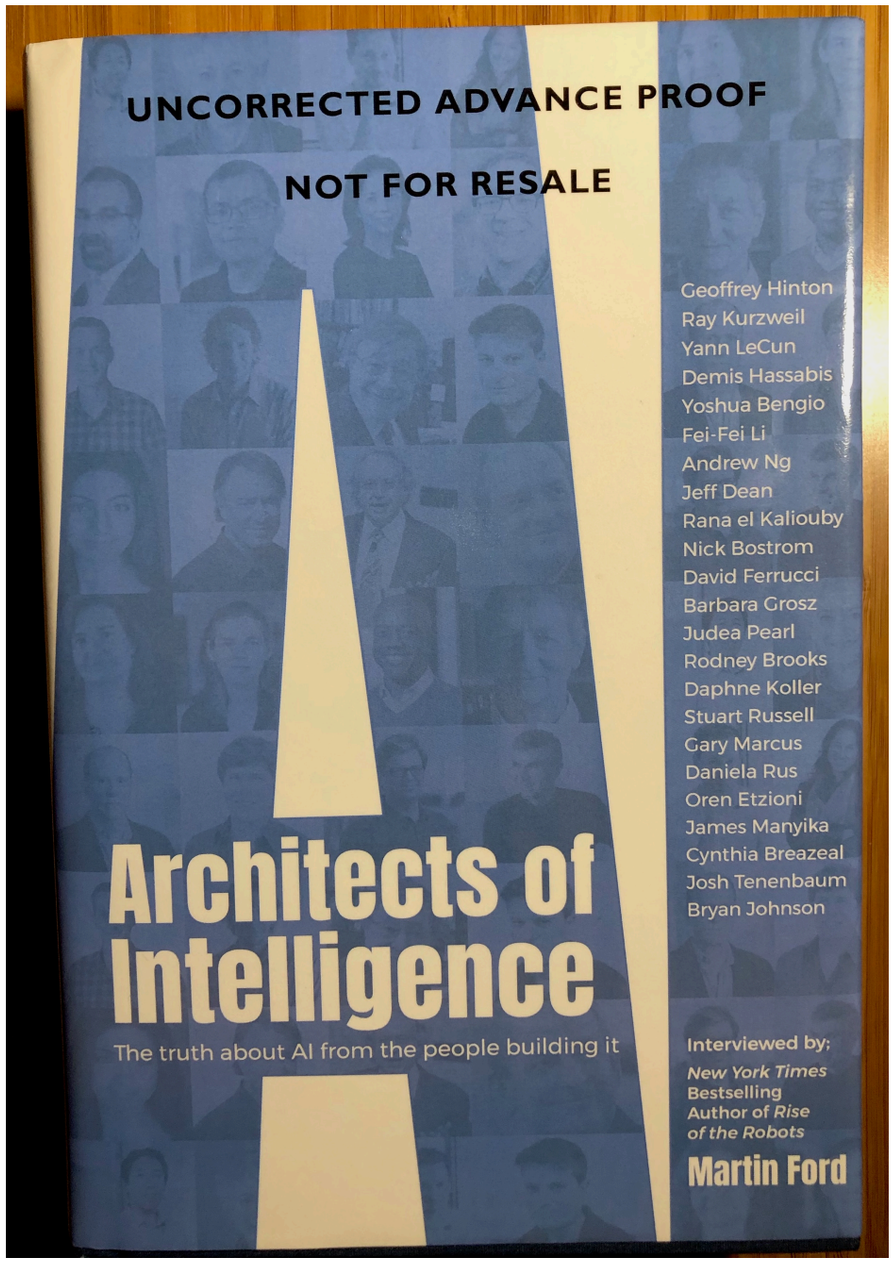
So I wrote the short paper about backpropagation, which was the *Nature* paper that came out in 1986, but Dave still hasn't written the short paper about autoencoders. I'm still waiting.

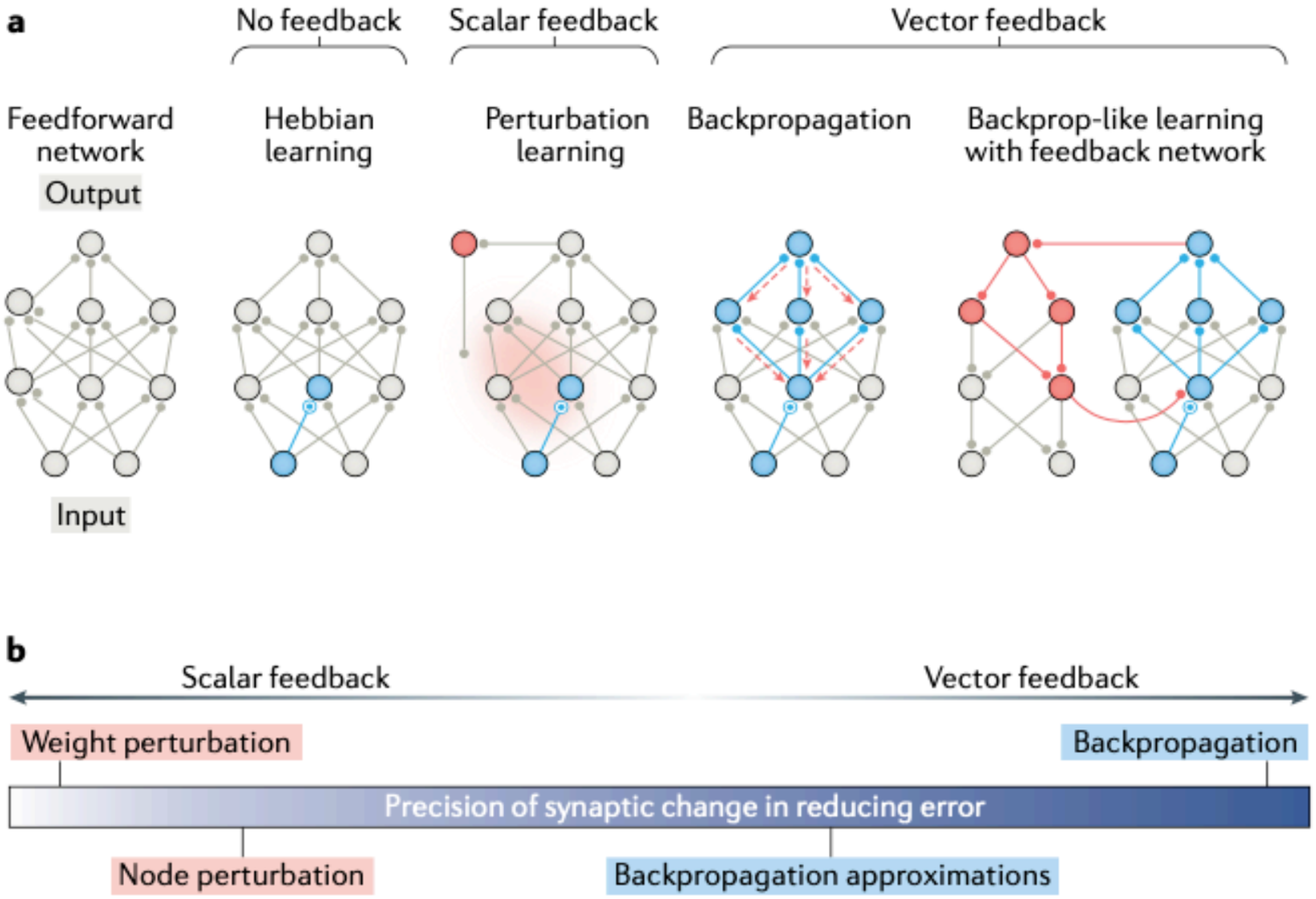
What he did do was tell Dave Zipser about the idea of autoencoders and

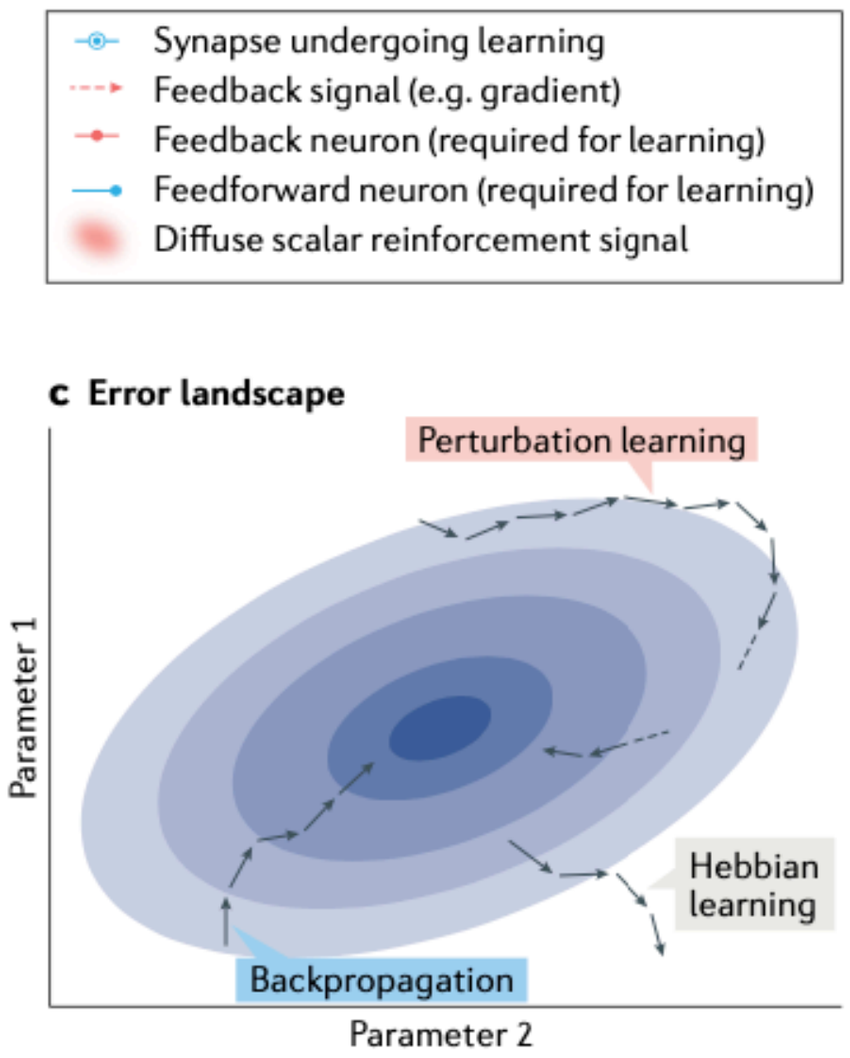
– Geoffrey Hinton in Talking Nets - An Oral History of Neural Networks, pg. 380

Interview with Experts: Suggestions for Pleasure Reading



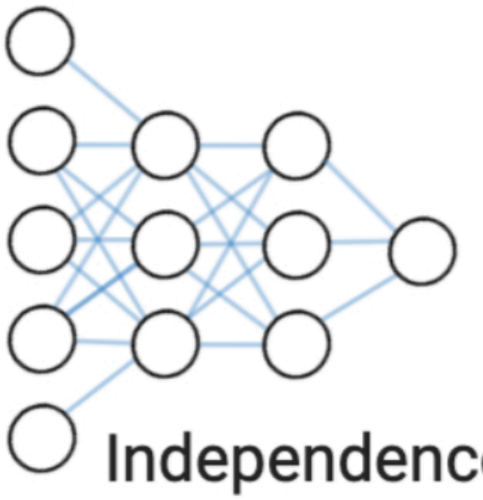






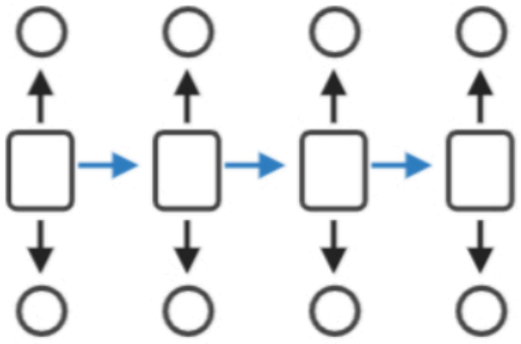
Lillicrap, T. P., Santoro, A., Marris, L., Akerman, C. J., & Hinton, G. (2020). Backpropagation and the brain. *Nature Reviews Neuroscience*, *21*(6), 335-346.

Relational Inductive Biases

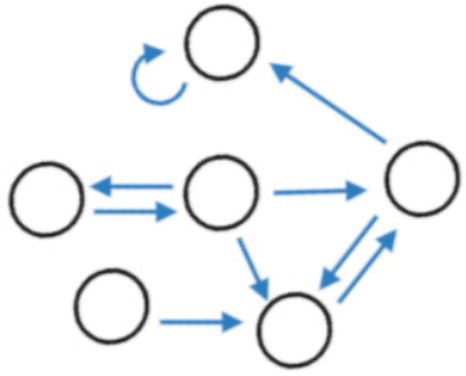


Independence

Locality



Sequentiality



Specified

Source: https://sgfin.github.io/2020/06/22/Induction-Intro/

- "inductive biases stipulate the properties that we believe our model should have in order to generalize to future data; they thus encode our key assumptions about the problem itself."

The Origins of Deep Learning

Representation learning with advanced architectures with many layers & algorithmic improvements for better computational efficiency and convergence

1. Artificial neurons
2. Multilayer neural networks
3. **Deep learning**
4. The DL hardware & software landscape
5. Current research trends

Convolutional Neural Networks

Shortly after followed a breakthrough in image recognition using some clever enhancements to

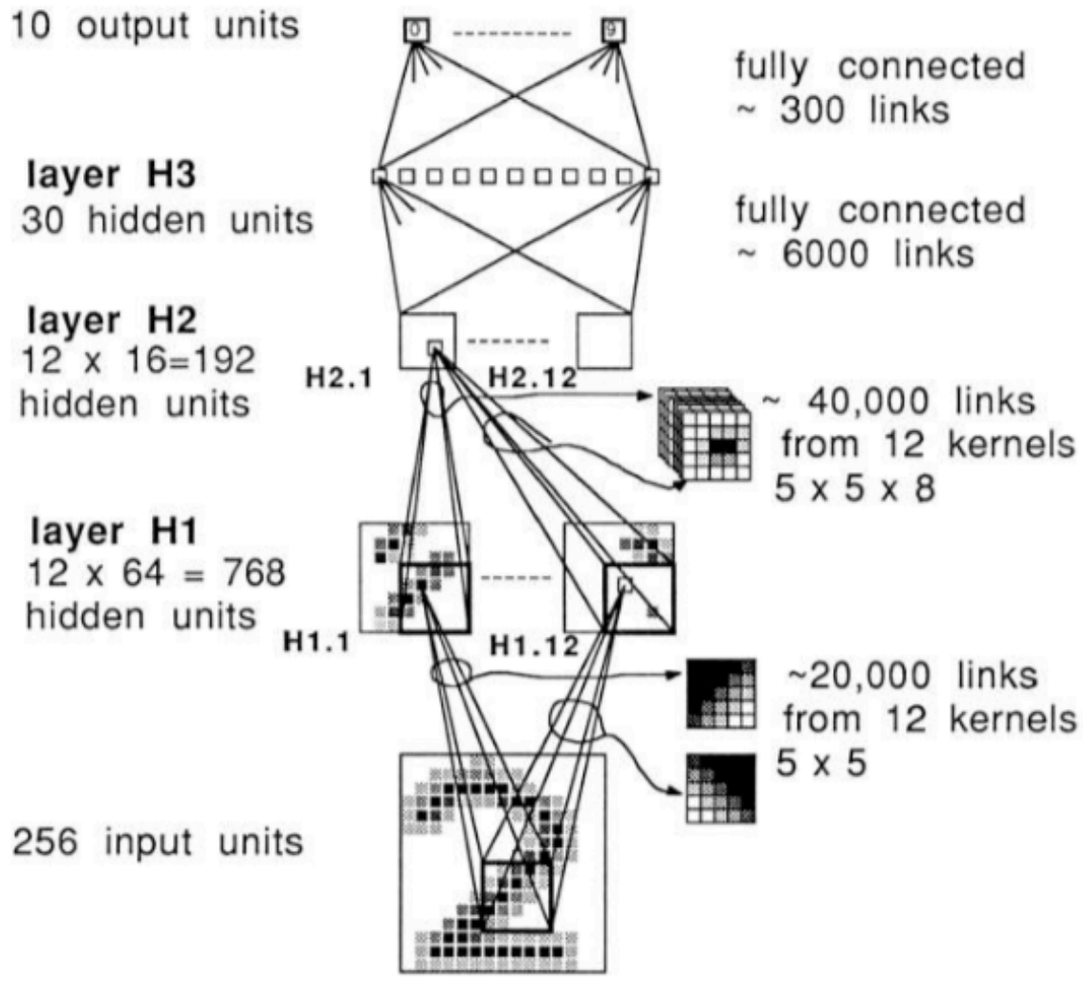
a) make training more efficient
b) extract local features (and better capture feature dependency)

by

- Weight sharing
- Pooling

LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., & Jackel, L. D. (1989). Backpropagation applied to handwritten zip code recognition. *Neural computation*, *1*(4), 541-551.

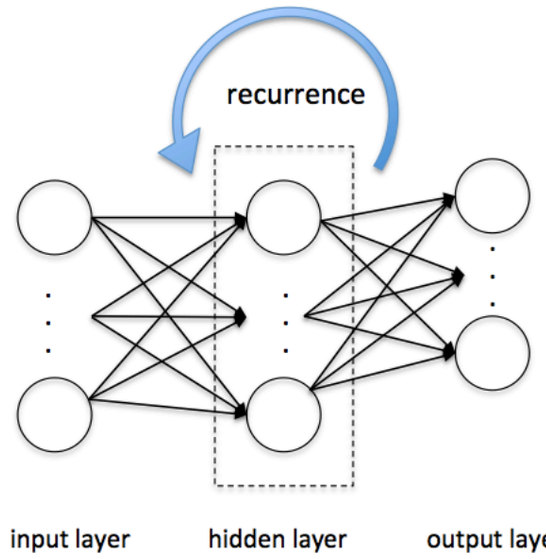
(the origin of Convolutional Neural Networks and MNIST)



Recurrent Neural Networks

Recurrent Neural Networks and Backpropagation through time

Some time created in the 1980's based on Rumelhart's work



New problems: vanishing and exploding gradients!

Schmidhuber, Jürgen (1993). *Habilitation thesis: System modeling and optimization*. Page 150 ff demonstrates credit assignment across the equivalent of 1,200 layers in an unfolded RNN.

Solution: LSTMs (still popular and commonly used)

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, *9*(8), 1735-1780.

About the term "Deep Learning" ...

Representation learning is a set of methods that allows a machine to be fed with raw data and to automatically discover the representations needed for detection or classification. Deep learning methods are representation-learning methods with multiple levels of representation [...]

-- LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. Nature, 521(7553), 436.

Neural Networks and Deep Learning -- A Timeline

- 2nd "AI Winter" in the late 1990's and 2000's
- Probably due to popularity of
  Support Vector Machines and Random Forests
- Also, neural networks were still expensive to train, until GPUs came into play

Oh, K. S., & Jung, K. (2004). GPU implementation of neural networks. *Pattern Recognition*, *37*(6), 1311-1314.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems* (pp. 1097-1105).

Image source: https://www.amax.com/blog/?p=907

Specifications	Intel® Core™ i7-6900K Processor Extreme Ed.	NVIDIA GeForce® GTX™ 1080 Ti
Base Clock Frequency	3.2 GHz	< 1.5 GHz
Cores	8	3584
Memory Bandwidth	64 GB/s	484 GB/s
Floating-Point Calculations	409 GFLOPS	11300 GFLOPS
Cost	~ $1000.00	~ $700.00

(data from ~2017)

When did Deep Learning Become Really Popular?

That was the view of people in computer vision until 2012. Most people in computer vision thought this stuff was crazy, even though Yann LeCun sometimes got systems working better than the best computer vision systems, they still thought this stuff was crazy, it wasn't the right way to do vision. They even rejected papers by Yann, even though they worked better than the best computer vision systems on particular problems, because the referees thought it was the wrong way to do things. That's a lovely example of scientists saying, "We've already decided what the answer has to look like, and anything that doesn't look like the answer we believe in is of no interest."

In the end, science won out, and two of my students won a big public competition, and they won it dramatically. They got almost half the error rate of the best computer vision systems, and they were using mainly techniques developed in Yann LeCun's lab but mixed in with a few of our own techniques as well.

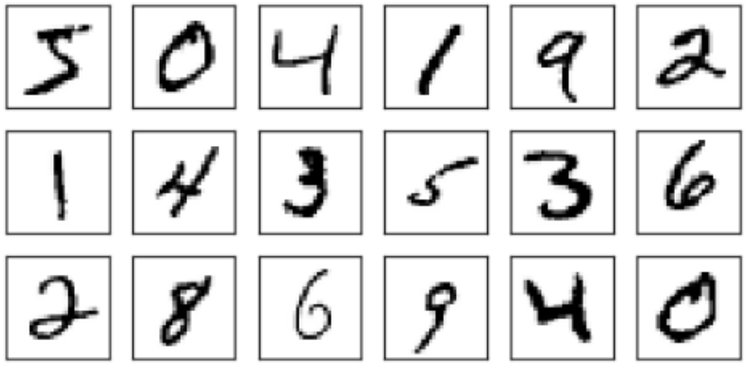
MARTIN FORD: This was the ImageNet competition?

GEOFFREY HINTON: Yes, and what happened then was what should happen in science. One method that people used to think of as complete nonsense had now worked much better than the method they believed in, and within two years, they all switched. So, for things like object classification, nobody would dream of trying to do it without using a neural network now.

(Excerpt from "Architects of Intelligence")

AlexNet achieved 15.4% error on top-5 in 2012
2nd best was not even close: 26.2% (nowadays ~3% error on ImageNet)

Evolution of Benchmark Datasets for Computer Vision



MNIST (1998)

- 60,000 examples, 10 classes
- features: 28x28x1
- http://yann.lecun.com/exdb/mnist/

CIFAR-10/CIFAR-100 (2009)

- 60,000 examples, 10 or 100 classes
- features: 32x32x3,
- https://www.cs.toronto.edu/~kriz/cifar.html

ImageNet (~2010)

- ~14 million images
- features: full resolution
- http://www.image-net.org

Neural Networks and Deep Learning -- A Timeline

- Many enhancements where developed to make neural networks perform better and solve new problems

Rectified Linear Units

Nair, V., & Hinton, G. E. (2010). Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)* (pp. 807-814).

BatchNorm

Ioffe, S., & Szegedy, C. (2015, June). Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In *International Conference on Machine Learning* (pp. 448-456).

Dropout

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, *15*(1), 1929-1958.

GANs

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... & Bengio, Y. (2014). Generative adversarial nets. In *Advances in neural information processing systems* (pp. 2672-2680).

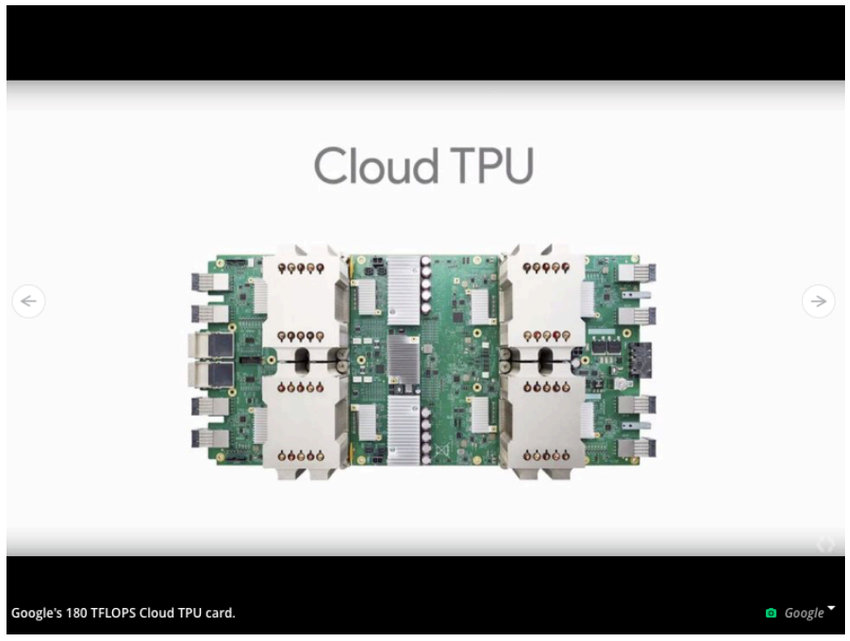
(All covered in this course)

& many more

The Deep Learning Hardware and Software Landscape

1. Artificial neurons
2. Multilayer neural networks
3. Deep learning
4. **The DL hardware & software landscape**
5. Current research trends

Developing Specialized Hardware

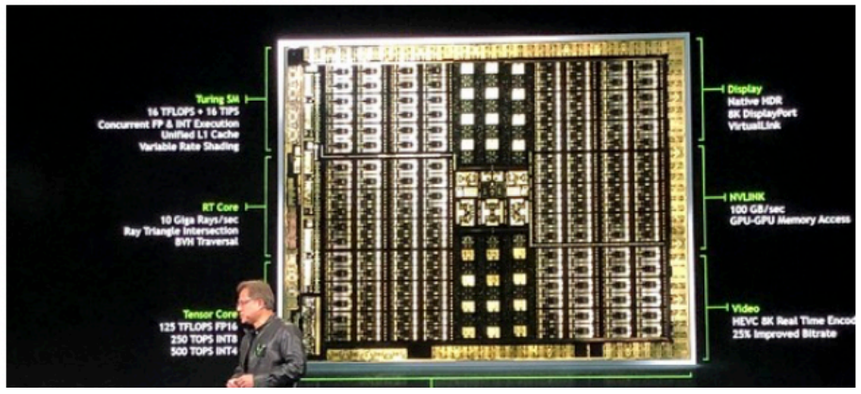


https://arstechnica.com/gadgets/2018/07/the-ai-revolution-has-spawned-a-new-chips-arms-race/

Opinion: New Nvidia chip extends the company's lead in graphics, artificial intelligence

By Ryan Shrout
Published: Aug 14, 2018 2:35 p.m. ET

The only question that remains: How big is Nvidia's advantage over its rivals?



https://www.marketwatch.com/story/new-nvidia-chip-extends-the-companys-lead-in-graphics-artificial-intelligence-2018-08-14

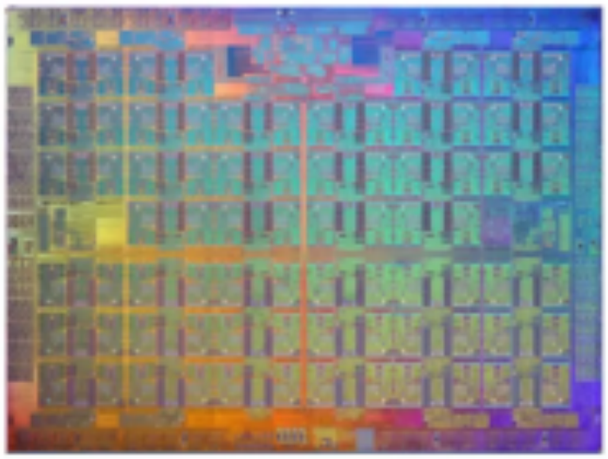
https://developer.arm.com/products/processors/machine-learning/arm-ml-processor

TECHNOLOGY NEWS NOVEMBER 28, 2018 / 2:59 PM / 2 MONTHS AGO

Amazon launches machine learning chip, taking on Nvidia, Intel

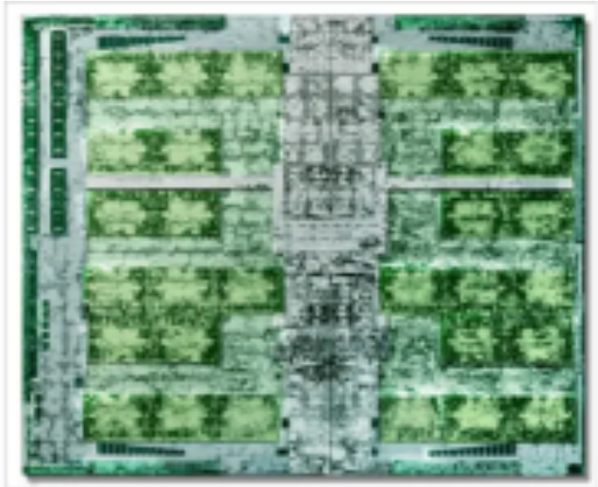
https://www.reuters.com/article/us-amazon-com-nvidia/amazon-launches-machine-learning-chip-taking-on-nvidia-intel-idUSKCN1NX2PY

Image credit: Graphcore & https://towardsdatascience.com/predictions-and-hopes-for-graph-ml-in-2021-6af2121c3e3d



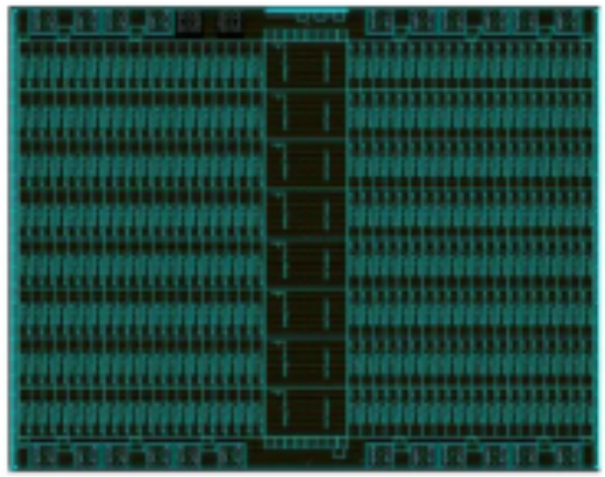
CPU

Scalar



GPU

Vector



IPU

Graph

Intelligence Processing Unit (IPU)

At the time of PyTorch's first beta release:

- Theano and TensorFlow were the premiere low-level libraries, working with a model that had the user define a computational graph and then execute it.
- Lasagne and Keras were high-level wrappers around Theano, with Keras wrapping TensorFlow and CNTK as well.
- Caffe, Chainer, DyNet, Torch (the Lua-based precursor to PyTorch), MXNet, CNTK, DL4J, and others filled various niches in the ecosystem.

In the roughly two years that followed, the landscape changed drastically. The community largely consolidated behind either PyTorch or TensorFlow, with the adoption of other libraries dwindling, except for those filling specific niches. In a nutshell:

- Theano, one of the first deep learning frameworks, has ceased active development.
- TensorFlow:
  - Consumed Keras entirely, promoting it to a first-class API
  - Provided an immediate-execution "eager mode" that is somewhat similar to how PyTorch approaches computation
  - Released TF 2.0 with eager mode by default
- JAX, a library by Google that was developed independently from TensorFlow, has started gaining traction as a NumPy equivalent with GPU, autograd and JIT capabilities.
- PyTorch:
  - Consumed Caffe2 for its backend
  - Replaced most of the low-level code reused from the Lua-based Torch project
  - Added support for ONNX, a vendor-neutral model description and exchange format
  - Added a delayed-execution "graph mode" runtime called *TorchScript*
  - Released version 1.0
  - Replaced CNTK and Chainer as the framework of choice by their respective corporate sponsors

From Stevens et al.'s
"Deep Learning with PyTorch"

Current Trends in Deep Learning

1. Artificial neurons
2. Multilayer neural networks
3. Deep learning
4. The DL hardware & software landscape
5. **Current research trends**

Self-supervised learning

Self-supervised learning and computer vision by Jeremy Howard; https://www.fast.ai/2020/01/13/self_supervised/

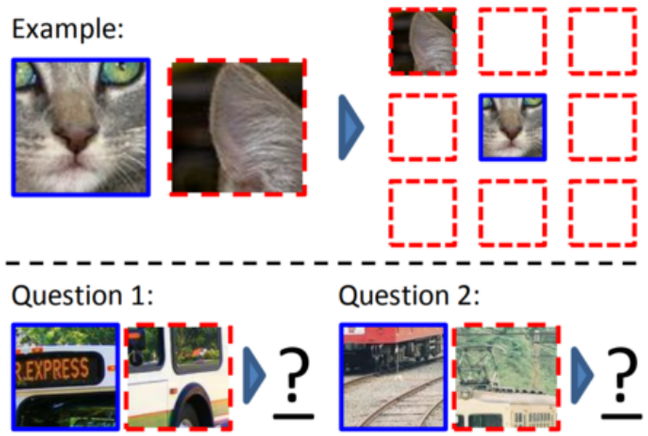
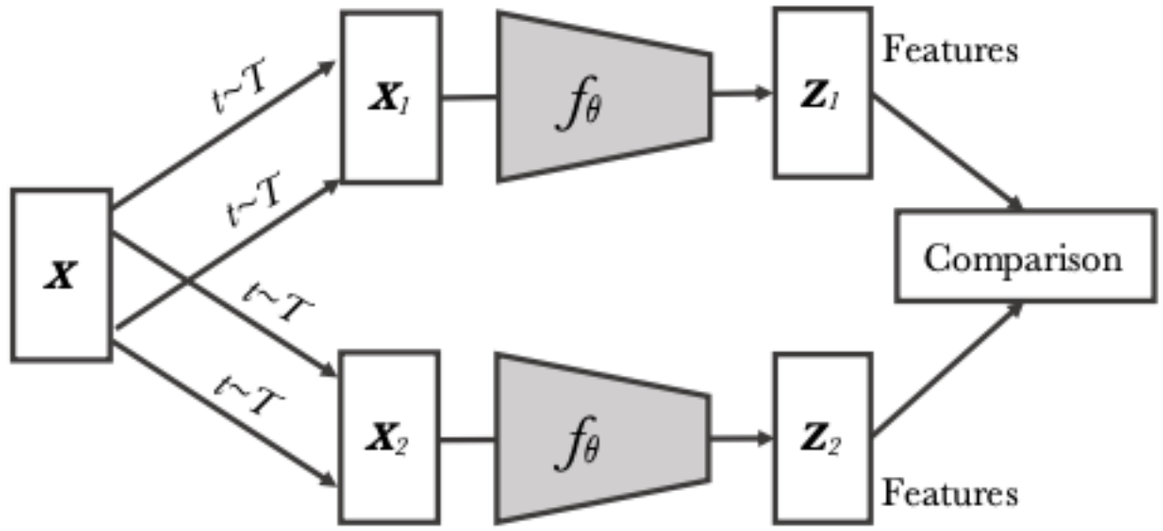


Figure 1. Our task for learning patch representations involves randomly sampling a patch (blue) and then one of eight possible neighbors (red). Can you guess the spatial configuration for the two pairs of patches? Note that the task is much easier once you have recognized the object!

Answer key: Q1: Bottom right Q2: Top center

Lee, H. Y., Huang, J. B., Singh, M., & Yang, M. H. (2017). Unsupervised representation learning by sorting sequences. In *Proceedings of the IEEE International Conference on Computer Vision* (pp. 667-676).



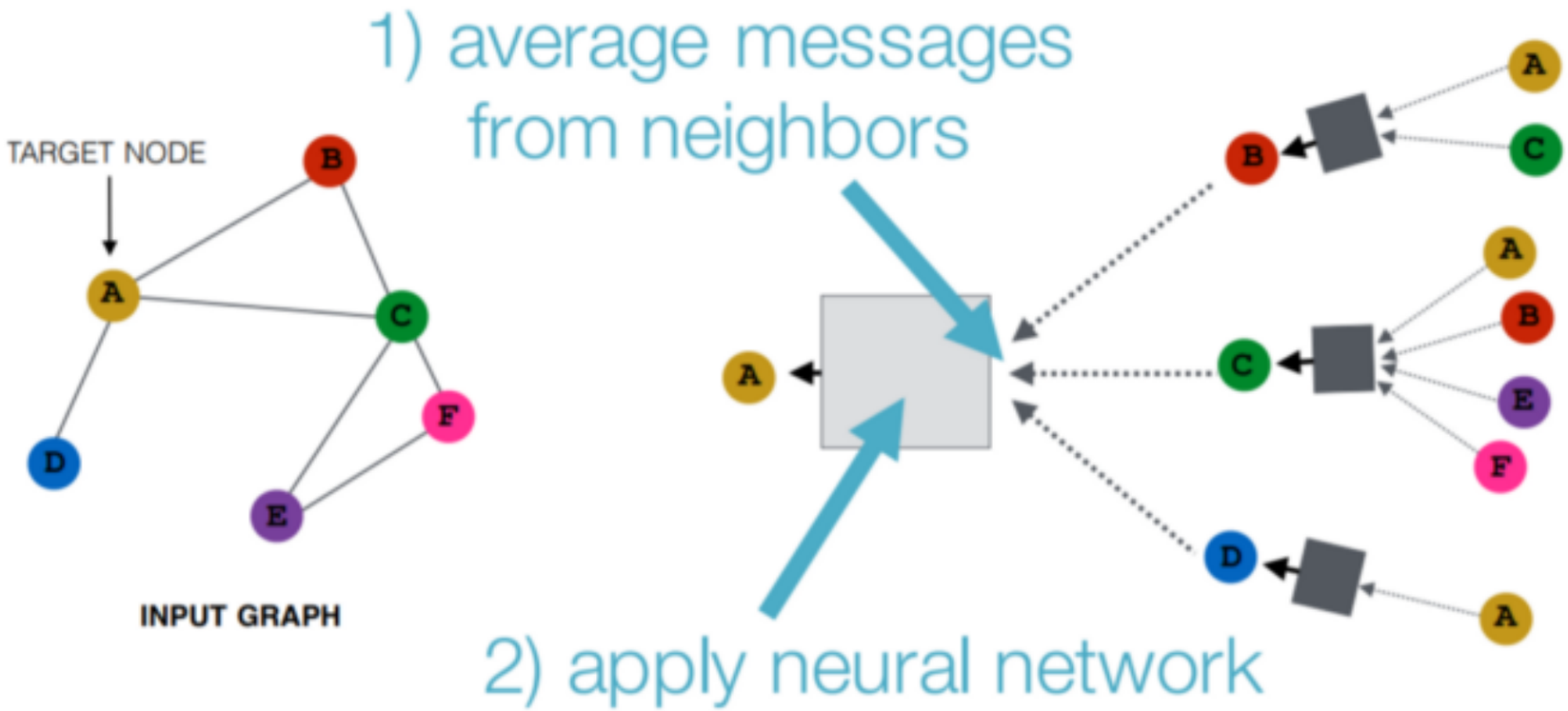
contrastive learning gained popularity in self-supervised representation learning

Caron, M., Misra, I., Mairal, J., Goyal, P., Bojanowski, P., & Joulin, A. (2020). Unsupervised learning of visual features by contrasting cluster assignments. *arXiv preprint arXiv:2006.09882*.

Instance discrimination compares features from different transformations of the same images to each other

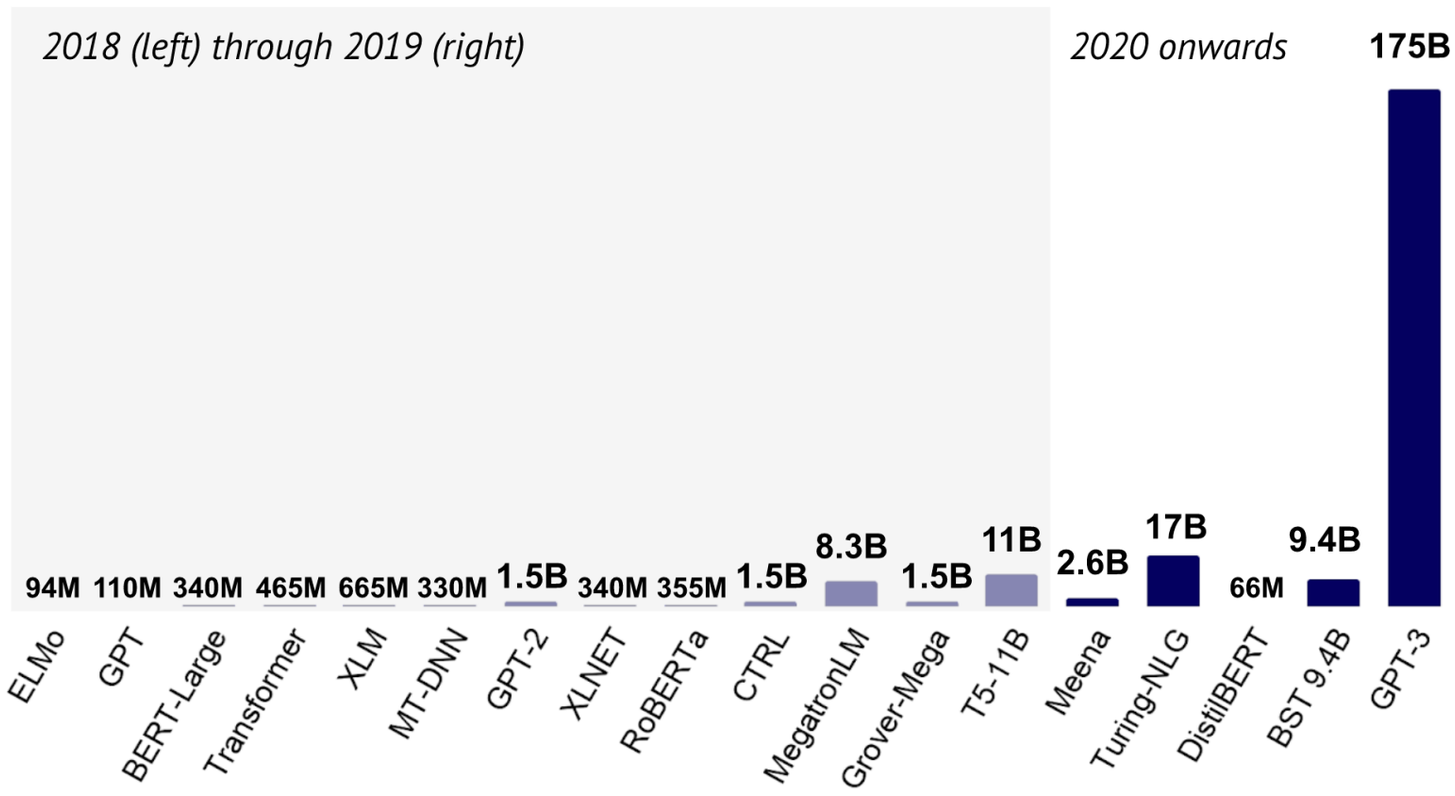
Graph neural networks

(A gentle introduction to graph neural networks: https://heartbeat.fritz.ai/introduction-to-graph-neural-networks-c5a9f4aa9e99)



Large-scale language models

https://ruder.io/research-highlights-2020/



Model sizes of language models from 2018–2020 (Credit: State of AI Report 2020)

arXiv.org > cs > arXiv:2101.01169

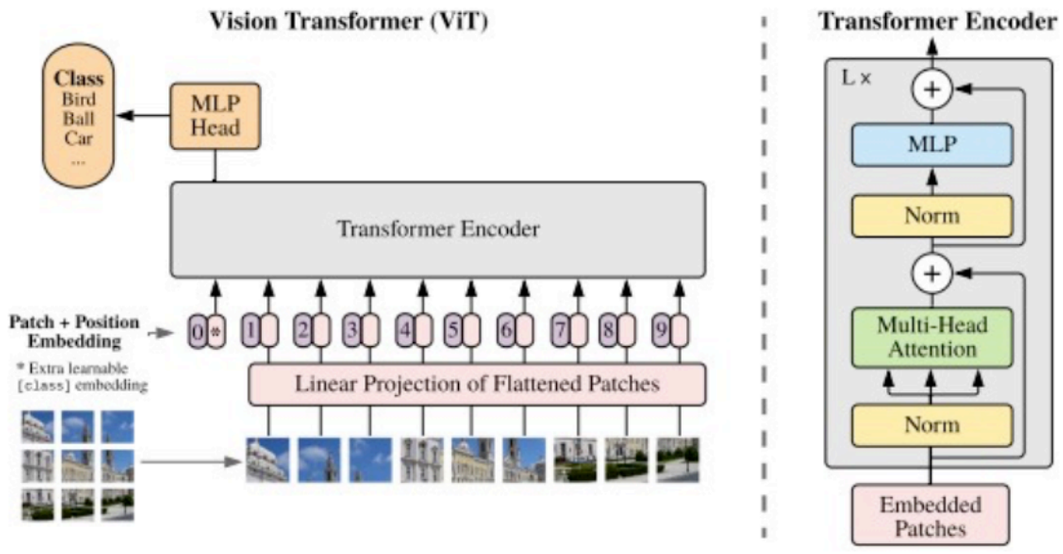
Computer Science > Computer Vision and Pattern Recognition

[Submitted on 4 Jan 2021]

Transformers in Vision: A Survey

Salman Khan, Muzammal Naseer, Munawar Hayat, Syed Waqas Zamir, Fahad Shahbaz Khan, Mubarak Shah

https://arxiv.org/abs/2101.01169



ig. 7. An overview of vision transformer. The architecture resembles ansformers used in NLP domain and the image patches are simply fed the model after flattening. After training, the feature obtained from the rst position is used for classification. Image obtained from [9].

"Transformer is data-hungry in nature *e.g.*, a large- scale dataset like ImageNet [14 million images] is not enough to train vision transformer from scratch so [10] proposes to ..."

Next Lecture:

The Perceptron

Important: Homework for next lecture (ungraded)

As preparation for next lecture

Scientific Computing in Python: Introduction to NumPy and Matplotlib

-- Including Video Tutorials

https://sebastianraschka.com/blog/2020/numpy-intro.html