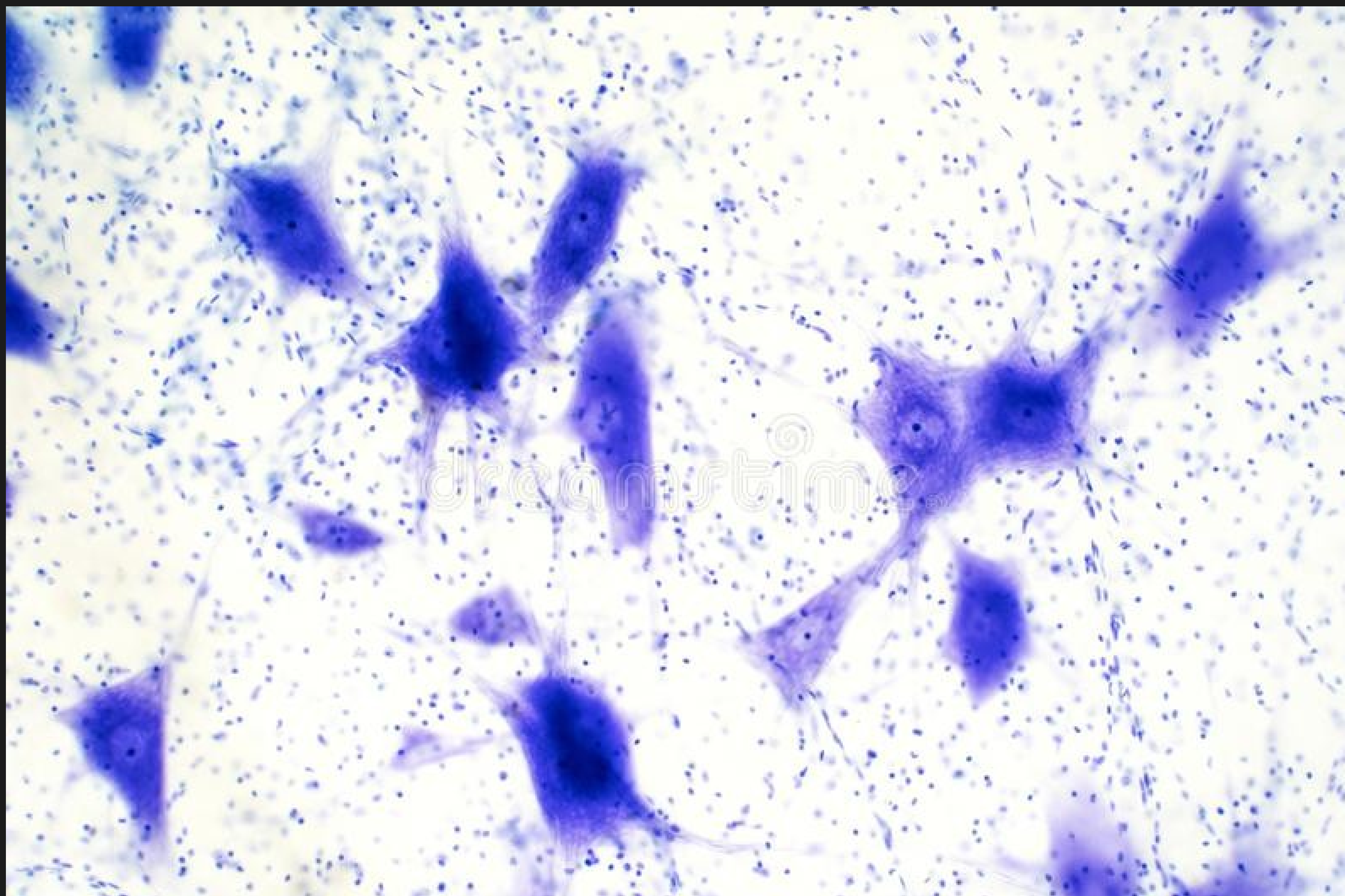
Deep Learning

Connectionism: AI by Mimicking the Human Brain



Perceptron (Rosenblatt 1958)

- First try to emulate the functionality of a neuron

$(x_1, x_2, x_3 \ldots)^T$

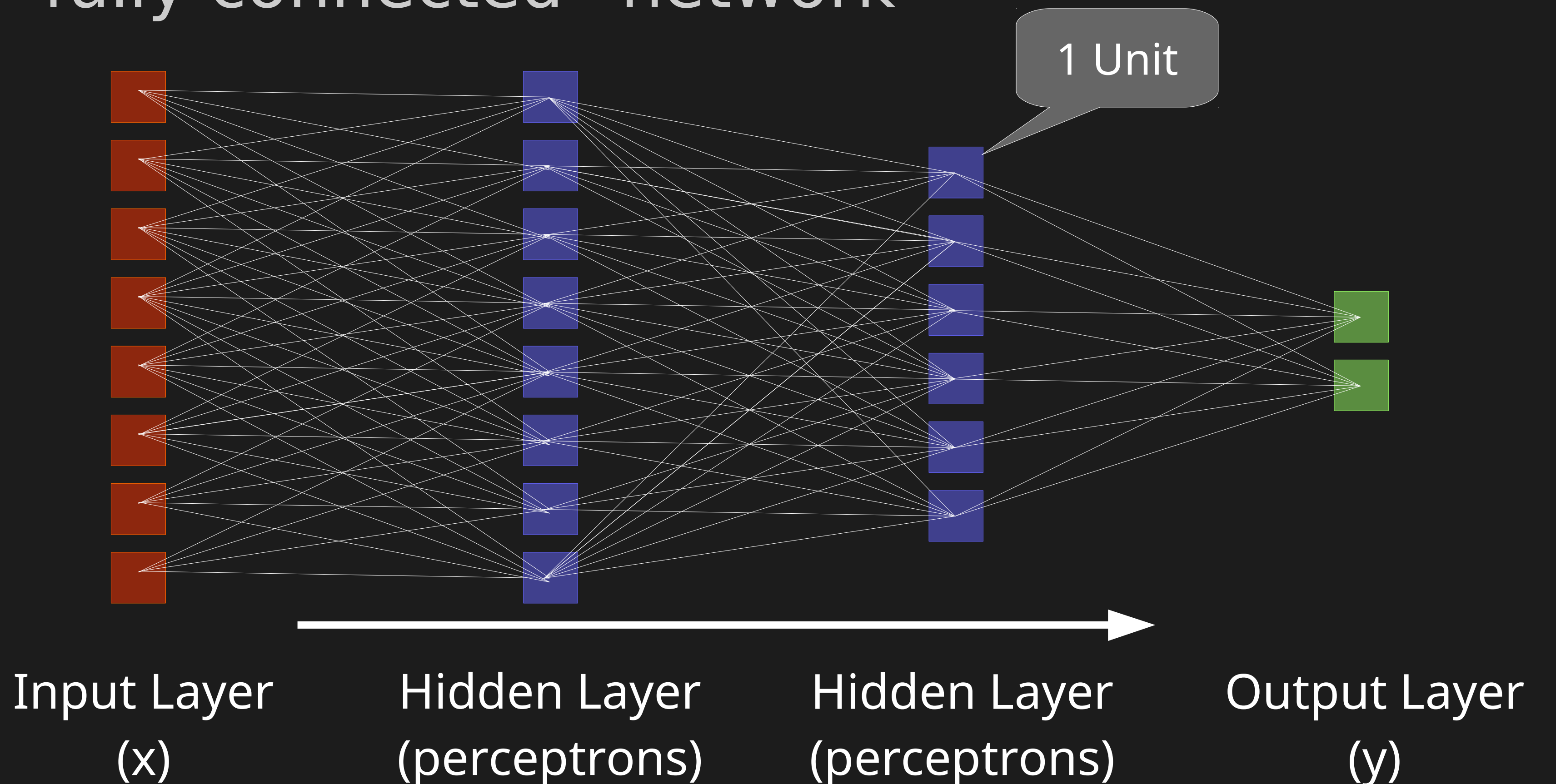
$$if\ \vec{w}\,\vec{x} + b > 0:\ \ return\ 1$$
$$if\ \vec{w}\,\vec{x} + b \leq 0:\ \ return\ 0$$

1/0

- Parameters: weights (w), bias (b)
- Threshold activation
- Binar/real-valued input, binary output

Multilayer Perceptron

- A single perceptron is not very useful, but you can combine many of them in layers in a "fully connected" network

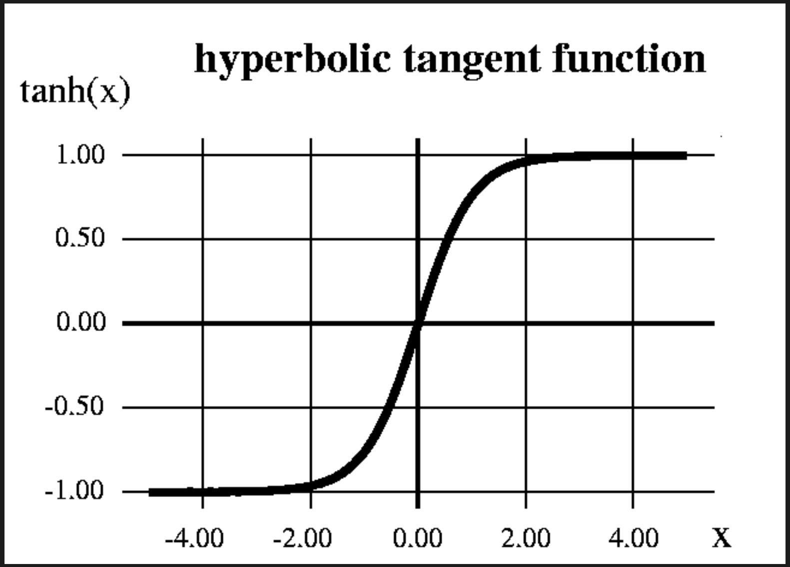


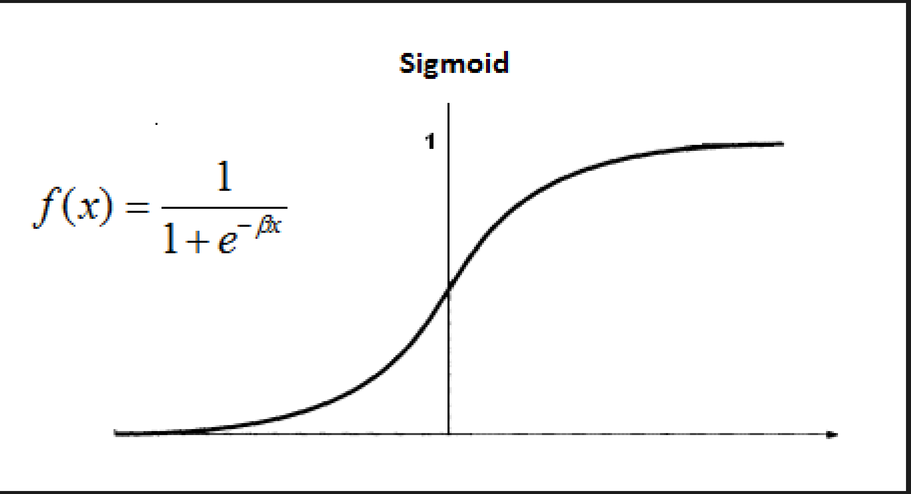
Multilayer Perceptron

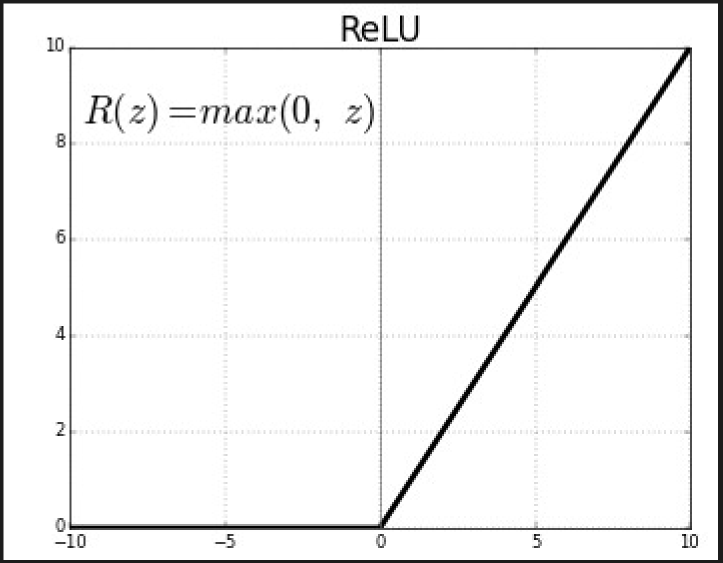
- First “artifical neural network”, popular in the 1980s
- Somewhat limited in complexity by threshold activation (see next slide)
- Successful in classification and regression; it was shown that a MLP can approximate any mathematical function (Cybenko’s theorem)

Activation Functions

- More complexity using other activation functions:



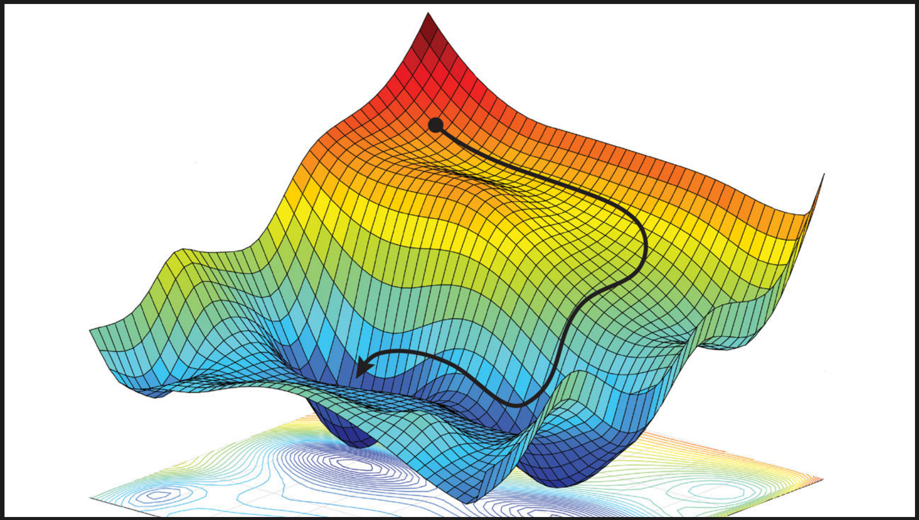




Training: Stochastic Gradient Descent

- Follow the gradient to the minimum loss:

$$\vec{p}_{i+1} = \vec{p}_i - \alpha \nabla_p L(y, y')$$



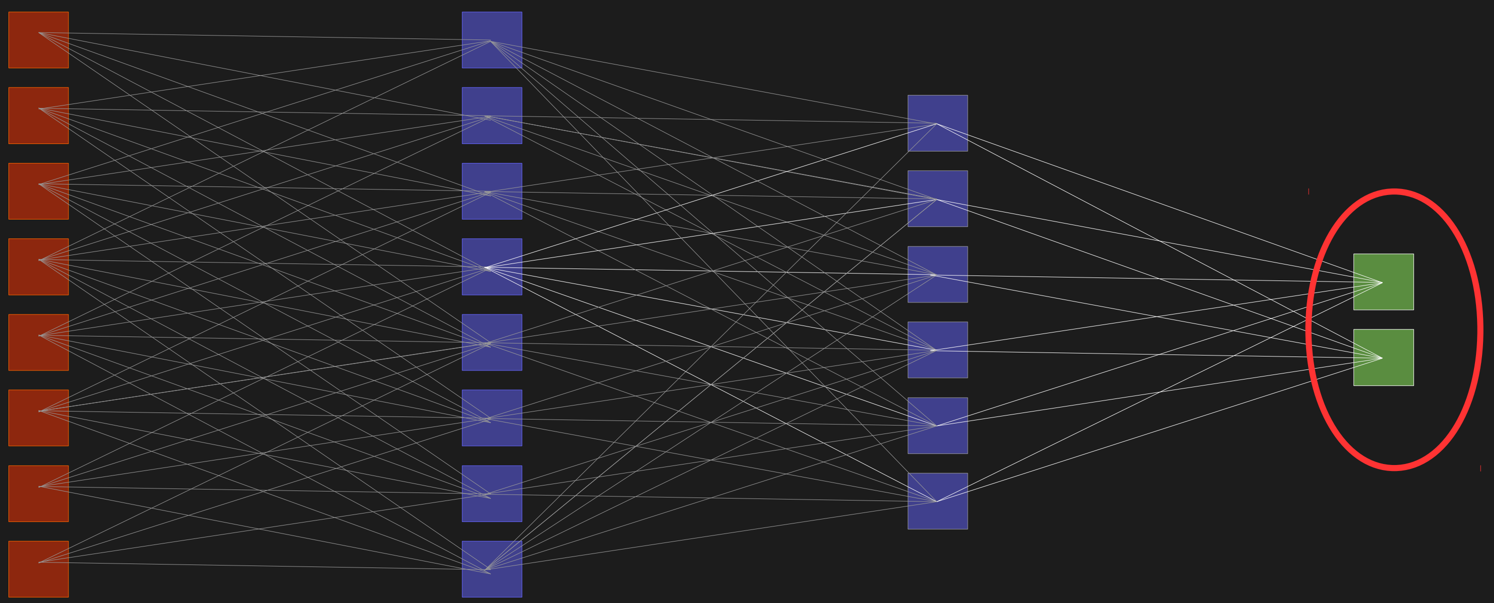
- Stochastic: consider only part of the data sample at a time
- Biggest problem: calculate gradients in weight space

Backpropagation (Rumelhart et al. 1986)

- Make use of chain rule of calculus:

$$f(x) = y, g(y) = z; \quad \frac{dg}{dx} = \frac{dg}{dy}\frac{dy}{dx}$$

- Account for all possible paths: summation



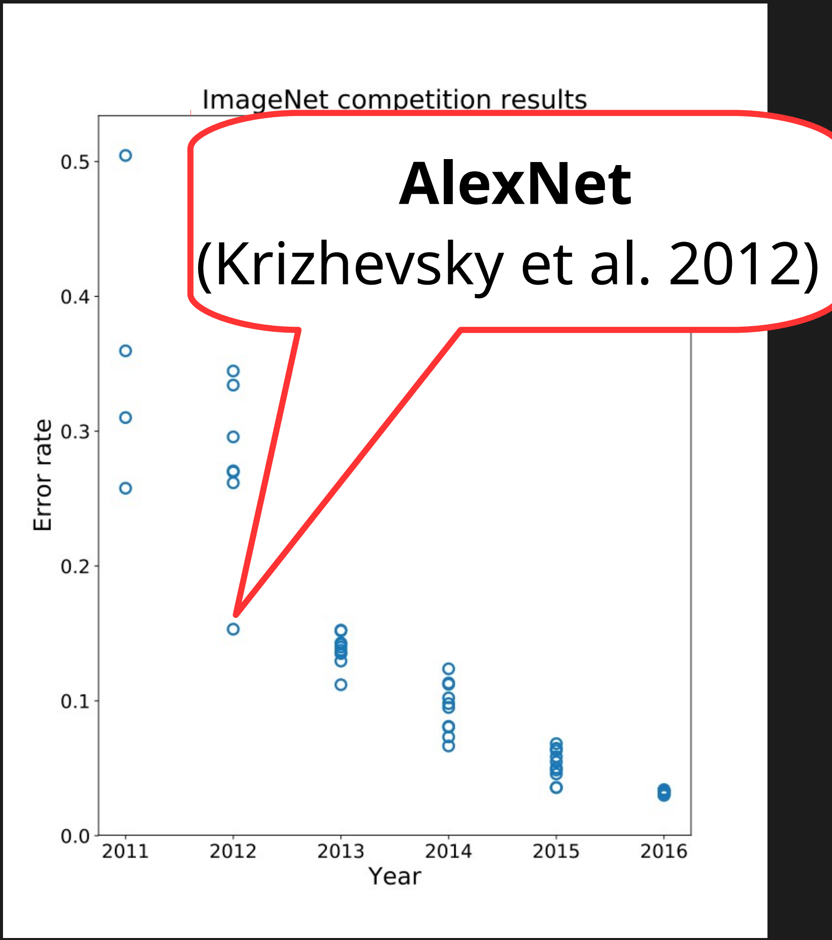
The Age of Shallow Learning (1980s – early 2000s)

- Feature selection + shallow learning more successful because:
  - Limited computational power
  - Limited data
- MLPs were superseded by much simpler Support Vector Machines and other models
- Mostly theoretical progress in Neural Networks

The Age of Deep Learning (2010+)

- No feature selection + deep learning:
  - Computation is cheap (GPUs)
  - Data is plenty (internet)

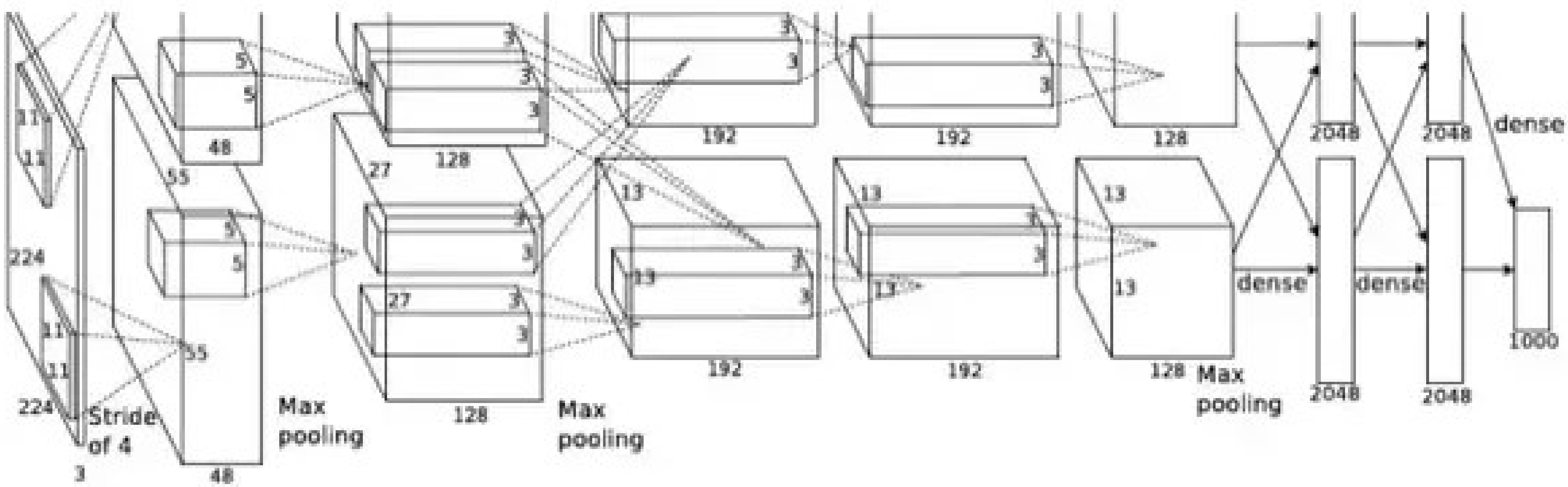
- ImageNet object recognition challenge:
  14M images
  20k different classes



AlexNet

AlexNet

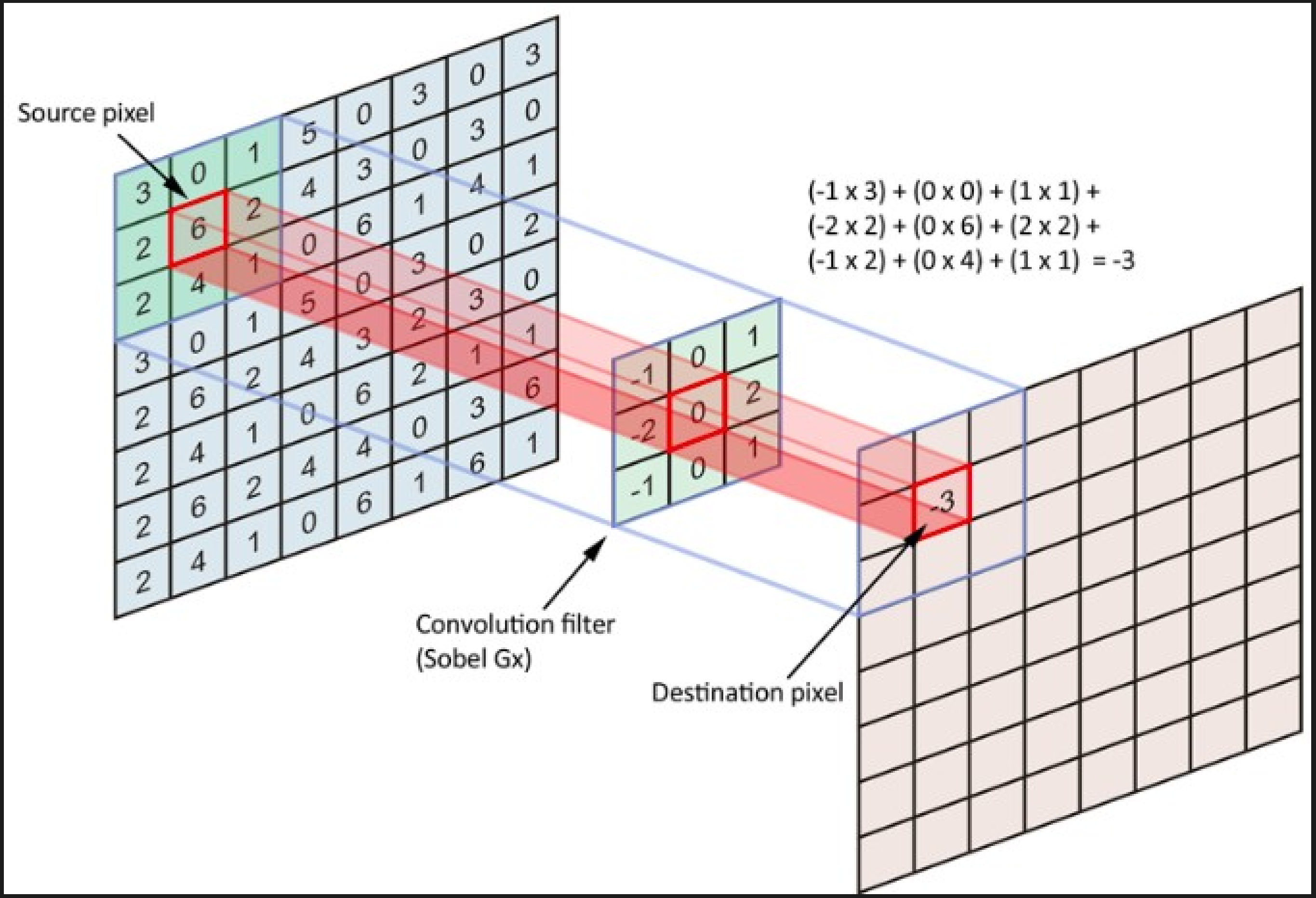
- Similar framework to LeCun'98 but:
  - Bigger model (7 hidden layers, 650,000 units, 60,000,000 params)
  - More data (10^6 vs. 10^3 images)
  - GPU implementation (50x speedup over CPU)
    - Trained on two GPUs for a week



A. Krizhevsky, I. Sutskever, and G. Hinton,
ImageNet Classification with Deep Convolutional Neural Networks, NIPS 2012

Convolutional Neural Networks

- Convolutional layers



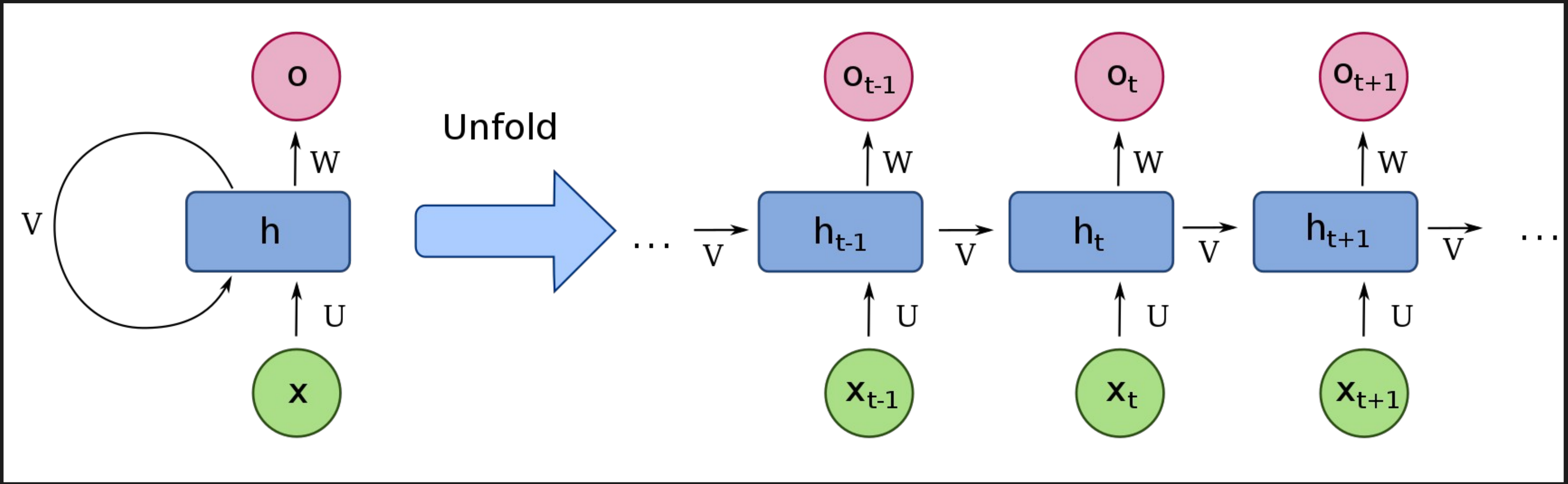
- Require equidistant data: images!

Convolutional Neural Network

- New regularization tools:
  - Max pooling: only keep max value from input
  - Dropout: skip random units
- Convolution provides translational invariance
- Example: MNIST handwritten digits

Recurrent Neural Network

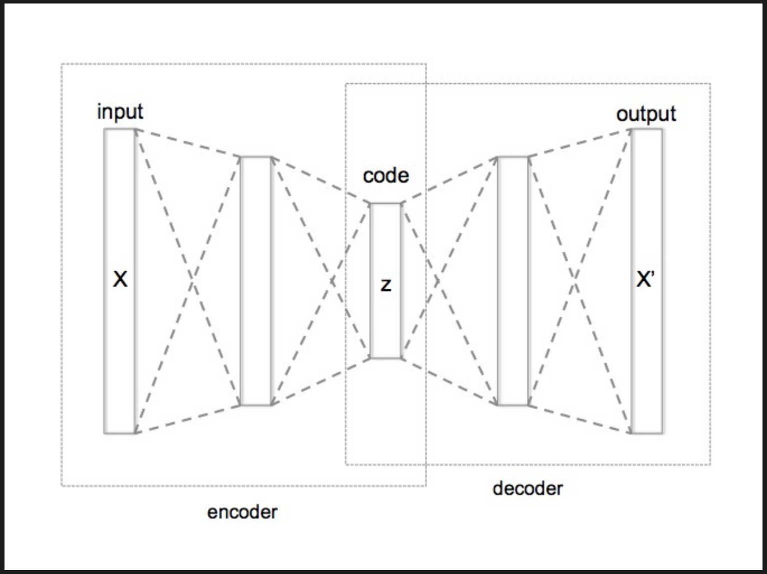
- Serial input data; predict next input value (e.g., language or time-series data)



- Example: DeepL translator

Decoder-Encoder Networks (Autoencoders)

- Transform input data into code space and then try to replicate the original data as good as possible



- Idea: learn efficient representation of input data

Generative Adversarial Networks

- Learn input data distribution to generate new data
- Example: celebrity images
- Example: NVIDIA GauGAN
- Example: Deep Fakes

Other Cool Stuff

- MuseNet
- Neural Transfer
- Adversarial Attacks