Deep Learning

Caio Corro – Michele Sebag
CNRS – INRIA – LIMSI – LRI

Orsay – Oct. 2019

Credit for slides: Yoshua Bengio, Yann Le Cun, Nando de Freitas, Christian Perone, Honglak Lee, Stéphane Canu

Types of Machine Learning problems

WORLD – DATA – USER

Observations	+ **Target**	+ Rewards
Understand **Code**	**Predict** **Classification/Regression**	Decide Action Policy/Strategy
Unsupervised **LEARNING**	**Supervised** **LEARNING**	Reinforcement LEARNING

News

Good News: Neural Nets can be used for all three goals:

- Unsupervised learning — change of representation
- Supervised learning — achieves prediction
- Reinforcement learning — yields the state-action value

Bad News

- not so easy to learn — **non convex** optimization
- not so easy to understand — black-box model
- its extensions (to complex/higher order logic domains) require *finesse*

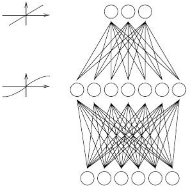
Resources: https://tao.lri.fr/ (Activities; Courses; Module Deep Learning)

Position

Generalities

Convolutional NN

Neural Nets



(C) David McKay - Cambridge Univ. Press

Properties

- Good: Multi-layer perceptrons are universal approximators
  For every decent function f ($= f^2$ has a finite integral on every compact of $\mathbb{R}^d$)
  for every $\epsilon > 0$,
  there exists some MLP/RBF g such that $||f - g|| < \epsilon$.

- Bad
  - Not a constructive proof (the solution exists, so what ?)
  - Everything is possible $\rightarrow$ no guarantee (overfitting).

- Very bad
  - A non convex **hard** optimization problem
  - Lots of local minima
  - Low reproducibility of the results (tricks; computational cost)

History

1943 A neuron as a computable function $y = f(\mathbf{x})$ — Pitts, McCullough

Intelligence → Reasoning → Boolean functions

1960 Connexionism + learning algorithms — Rosenblatt

1969 AI Winter — Minsky-Papert

1989 Back-propagation — Amari, Rumelhart & McClelland, LeCun

1995 Winter again — Vapnik

2006 Deep Learning — Bengio, Hinton

It was hard to come back — Le Cun 2007

The NIPS community has suffered of an acute convexivitis epidemic

- ML applications seem to have trouble moving beyond logistic regression, SVMs, and exponential-family graphical models.
- For a new ML model, convexity is viewed as a virtue
- Convexity is sometimes a virtue
- But it is often a limitation

- ML theory has essentially never moved beyond convex models
  - the same way control theory has not really moved beyond linear systems

- Often, the price we pay for insisting on convexity is an unbearable increase in the size of the model, or the scaling properties of the optimization algorithm [O(n^2), O(n^3)...]

Here dragons

Model selection

- Selecting number of neurons, NN architecture **More $\Rightarrow$? Better**
- Which learning criterion, how to find enough examples

Algorithmic choices

a difficult optimization problem

- Enforce stability through relaxation

$$\mathbf{W}_{neo} \leftarrow (1-\alpha)\mathbf{W}_{old} + \alpha\mathbf{W}_{neo}$$

- Decrease the learning rate α with time
- Stopping criterion ?

Tricks

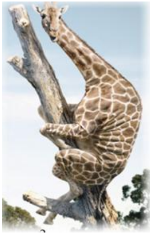
- Normalize data
- Initialize **W** small ! See Glorot initialization.

Position

Generalities

Convolutional NN

Toward deeper representations



Invariances matter

- The label of an image is invariant through small translation, homothety, rotation...
- Invariance of labels $\rightarrow$ Invariance of model

$$y(x) = y(\sigma(x)) \rightarrow h(x) = h(\sigma(x))$$

Enforcing invariances

- by augmenting the training set:

$$\mathcal{E} = \{(x_i, y_i)\} \bigcup \{(\sigma(x_i), y_i)\}$$

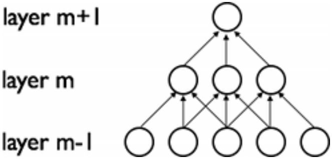
- by structuring the hypothesis space

Convolutional networks

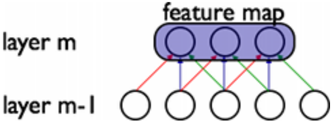
Hubel & Wiesel 1968

Visual cortex of the cat

- cells arranged in such a way that
- ... each cell observes a fraction of the visual field — receptive field
- ... their union covers the whole field



- Layer m: detection of local patterns (same weights)

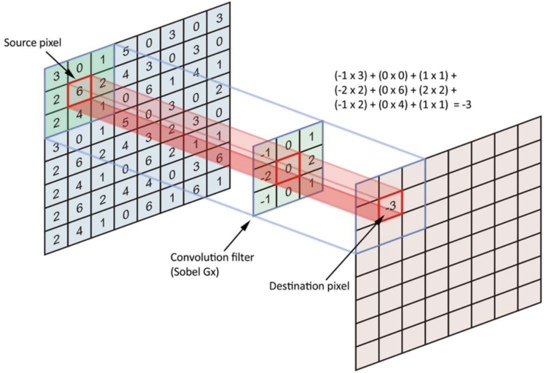


- Layer $m+1$: non linear aggregation of output of layer m

Ingredients of convolutional networks

1. Local receptive fields

(aka kernel or filter)



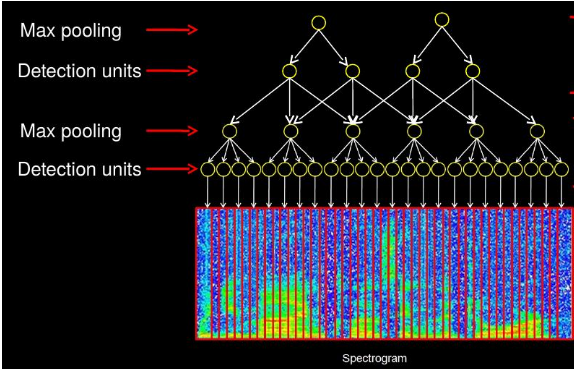
2. Sharing weights

through adapting the gradient-based update: the update is averaged over all occurrences of the weight.

Reduces the number of parameters by several orders of magnitude

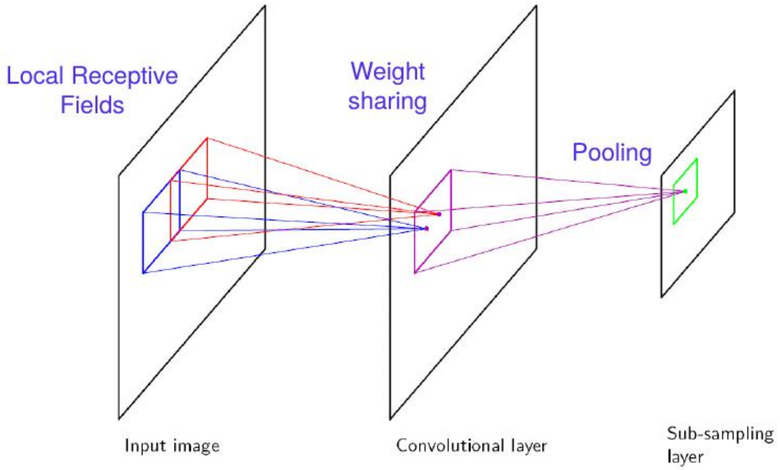
Ingredients of convolutional networks, 2

3. Pooling: reduction and invariance



- Overlapping / non-overlapping regions
- Return the max / the sum of the feature map over the region
- Larger receptive fields (see more of input)

Convolutional networks, summary

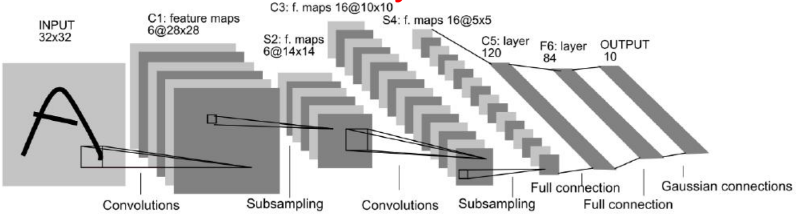


LeCun 1998

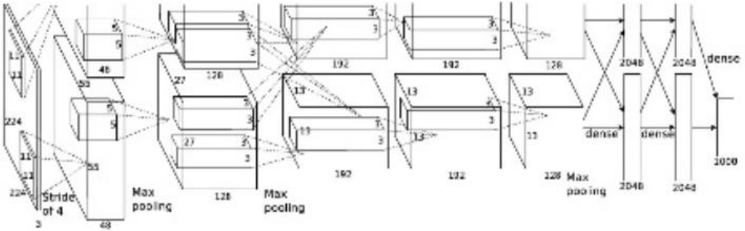
Properties

- Invariance to small transformations (over the region)
- Reducing the number of weights

Convolutional networks, summary



LeCun 1998



Kryzhevsky et al. 2012

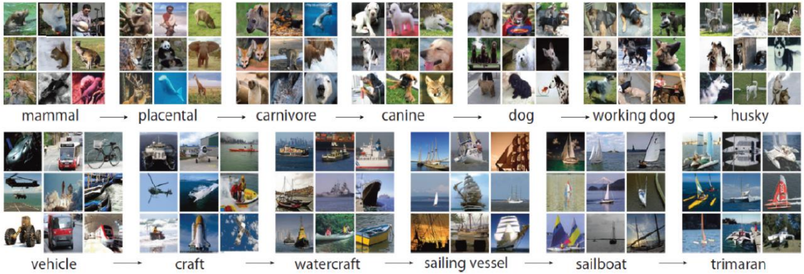
Properties

- Invariance to small transformations (over the region)
- Reducing the number of weights
- Usually many convolutional layers

ImageNet

Deng et al. 12

15 million labeled high-resolution images; 22,000 classes.



Large-Scale Visual Recognition Challenge

- 1000 categories.
- 1.2 million training images,
- 50,000 validation images,
- 150,000 testing images.

A leap in the state of the art

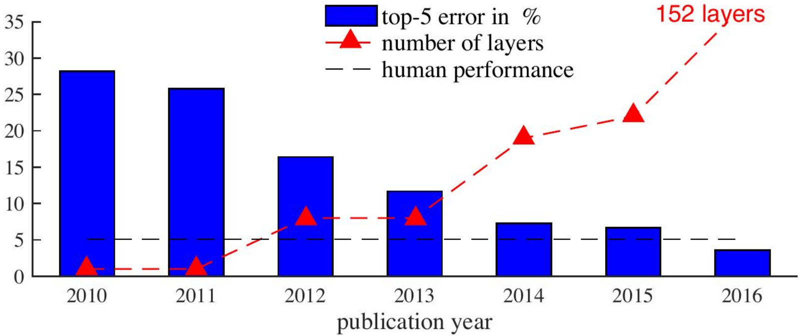
2012 Teams	%error	2013 Teams	%error	2014 Teams	%error
Supervision (Toronto)	15.3	Clarifai (NYU spinoff)	11.7	GoogLeNet	6.6
ISI (Tokyo)	26.1	NUS (singapore)	12.9	VGG (Oxford)	7.3
VGG (Oxford)	26.9	Zeiler-Fergus (NYU)	13.5	MSRA	8.0
XRCE/INRIA	27.0	A. Howard	13.5	A. Howard	8.1
UvA (Amsterdam)	29.6	OverFeat (NYU)	14.1	DeeperVision	9.5
INRIA/LEAR	33.4	UvA (Amsterdam)	14.2	NUS-BST	9.7
		Adobe	15.2	TTIC-ECP	10.2
		VGG (Oxford)	15.2	XYZ	11.2
		VGG (Oxford)	23.0	UvA	12.1

shallow approaches

deep learning

Y. LeCun StatLearn tutorial

Super-human performances



2012 Alex Net

2013 ZFNet

2014 VGG

2015 GoogLeNet / Inception

2016 Residual Network

karpathy.github.io/2014/09/02/what-i-learned-from-competing-against-a-convnet-on-imagenet/