UVA CS 4501 - 001 / 6501 – 007 Introduction to Machine Learning and Data Mining

Lecture 20: Neural Network / Deep Learning

Yanjun Qi / Jane, , PhD

University of Virginia
Department of
Computer Science

This is just first part of the whole lecture.
Full lecture is in L21.

Where are we ? ➔ Five major sections of this course

- ❑ Regression (supervised)
- ❑ Classification (supervised)
- ❑ Unsupervised models
- ❑ Learning theory
- ❑ Graphical models

A study comparing Classifiers

An Empirical Comparison of Supervised Learning Algorithms

Rich Caruana CARUANA@CS.CORNELL.EDU
Alexandru Niculescu-Mizil ALEXN@CS.CORNELL.EDU
Department of Computer Science, Cornell University, Ithaca, NY 14853 USA

Abstract

A number of supervised learning methods have been introduced in the last decade. Unfortunately, the last comprehensive empirical evaluation of supervised learning was the Statlog Project in the early 90's. We present a large-scale empirical comparison between ten supervised learning methods: SVMs, neural nets, logistic regression, naive bayes, memory-based learning, random forests, decision trees, bagged trees, boosted trees, and boosted stumps. We also examine the effect that calibrating the models via Platt Scaling and Isotonic Regression has on their performance. An important aspect of our study is

This paper presents results of a large-scale empirical comparison of ten supervised learning algorithms using eight performance criteria. We evaluate the performance of SVMs, neural nets, logistic regression, naive bayes, memory-based learning, random forests, decision trees, bagged trees, boosted trees, and boosted stumps on eleven binary classification problems using a variety of performance metrics: accuracy, F-score, Lift, ROC Area, average precision, precision/recall break-even point, squared error, and cross-entropy. For each algorithm we examine common variations, and thoroughly explore the space of parameters. For example, we compare ten decision tree styles, neural nets of many sizes, SVMs with many kernels, etc.

Because some of the performance metrics we examine

Proceedings of the 23rd International Conference on Machine Learning (ICML `06).

Yanjun Qi / UVA CS 4501-01-6501-07

A study comparing Classifiers
➔ 11 binary classification problems / 8 metrics

Table 2. Normalized scores for each learning algorithm by metric (average over eleven problems)

MODEL	CAL	ACC	FSC	LFT	ROC	APR	BEP	RMS	MXE	MEAN	OPT-SEL
BST-DT	PLT	.843*	.779	**.939**	**.963**	**.938**	.929*	**.880**	**.896**	**.896**	**.917**
RF	PLT	.872*	.805	.934*	.957	.931	**.930**	.851	.858	.892	.898
BAG-DT	–	.846	.781	.938*	.962*	.937*	.918	.845	.872	.887*	.899
BST-DT	ISO	.826*	.860*	.929*	.952	.921	.925*	.854	.815	.885	.917*
RF	–	**.872**	.790	.934*	.957	.931	**.930**	.829	.830	.884	.890
BAG-DT	PLT	.841	.774	.938*	.962*	.937*	.918	.836	.852	.882	.895
RF	ISO	.861*	**.861**	.923	.946	.910	.925	.836	.776	.880	.895
BAG-DT	ISO	.826	.843*	.933*	.954	.921	.915	.832	.791	.877	.894
SVM	PLT	.824	.760	.895	.938	.898	.913	.831	.836	.862	.880
ANN	–	.803	.762	.910	.936	.892	.899	.811	.821	.854	.885
SVM	ISO	.813	.836*	.892	.925	.882	.911	.814	.744	.852	.882
ANN	PLT	.815	.748	.910	.936	.892	.899	.783	.785	.846	.875
ANN	ISO	.803	.836	.908	.924	.876	.891	.777	.718	.842	.884
BST-DT	–	.834*	.816	**.939**	**.963**	**.938**	.929*	.598	.605	.828	.851
KNN	PLT	.757	.707	.889	.918	.872	.872	.742	.764	.815	.837
KNN	–	.756	.728	.889	.918	.872	.872	.729	.718	.810	.830
KNN	ISO	.755	.758	.882	.907	.854	.869	.738	.706	.809	.844
BST-STMP	PLT	.724	.651	.876	.908	.853	.845	.716	.754	.791	.808
SVM	–	.817	.804	.895	.938	.899	.913	.514	.467	.781	.810
BST-STMP	ISO	.709	.744	.873	.899	.835	.840	.695	.646	.780	.810
BST-STMP	–	.741	.684	.876	.908	.853	.845	.394	.382	.710	.726
DT	ISO	.648	.654	.818	.838	.756	.778	.590	.589	.709	.774

A study comparing Classifiers → 11 binary classification problems

PROBLEM	#ATTR	TRAIN SIZE	TEST SIZE	%POZ
ADULT	14/104	5000	35222	25%
BACT	11/170	5000	34262	69%
COD	15/60	5000	14000	50%
CALHOUS	9	5000	14640	52%
COV_TYPE	54	5000	25000	36%
HS	200	5000	4366	24%
LETTER.P1	16	5000	14000	3%
LETTER.P2	16	5000	14000	53%
MEDIS	63	5000	8199	11%
MG	124	5000	12807	17%
SLAC	59	5000	25000	50%

Today

- Neural Network
  - MLP (Multilayer Perceptron Network)
  - Training
- Deep CNN, why Deep Learning ?

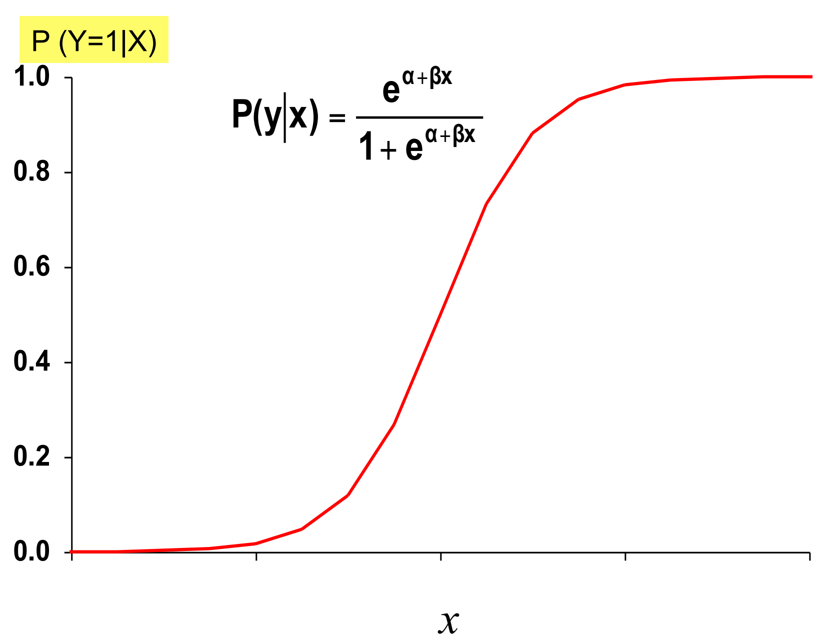
Logistic Regression

Task	classification
↓	↓
Representation	Log-odds(Y) = linear function of X's
↓	↓
Score Function	EPE, with conditional Log-likelihood
↓	↓
Search/Optimization	Iterative (Newton) method
↓	↓
Models, Parameters	Logistic weights

$$P(c=1|x) = \frac{1}{1+e^{-(\alpha+\beta x)}}$$

Using Logistic Function to Transfer

e.g. Probability of disease



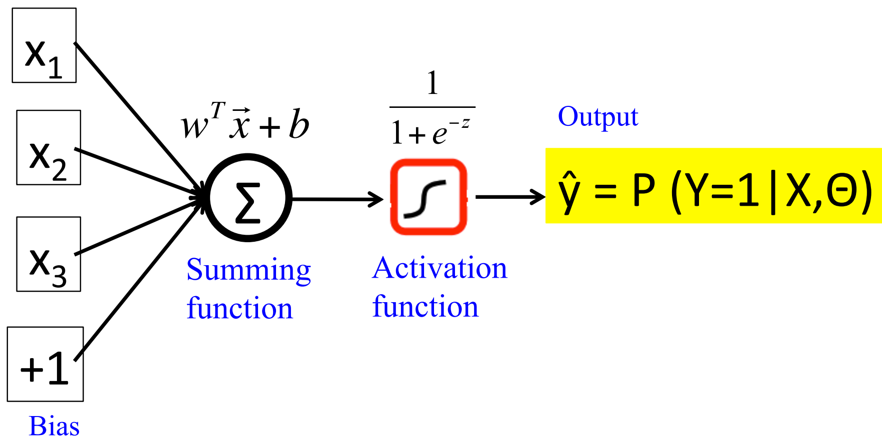
Logistic regression

Logistic regression could be illustrated as a module

On input x, it outputs ŷ:

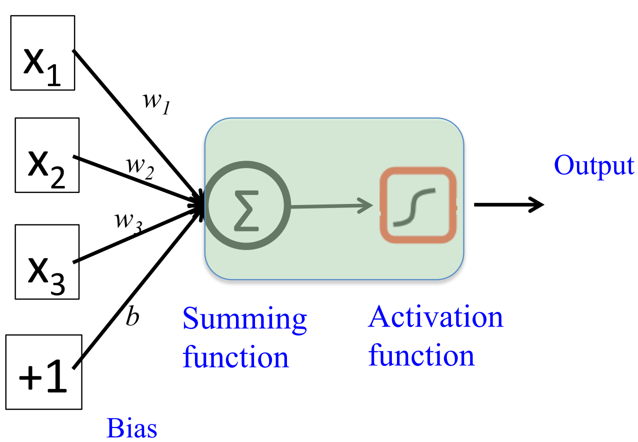
where

Draw a logistic regression unit as:



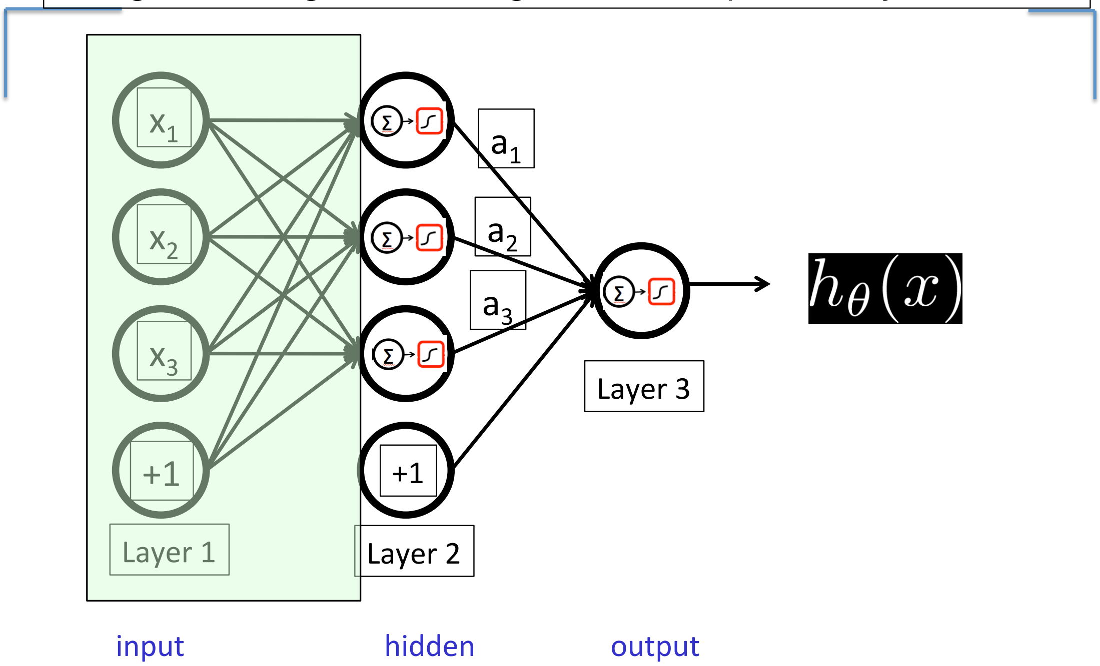
Multi-Layer Perceptron (MLP)

- 1 neuron, e.g.



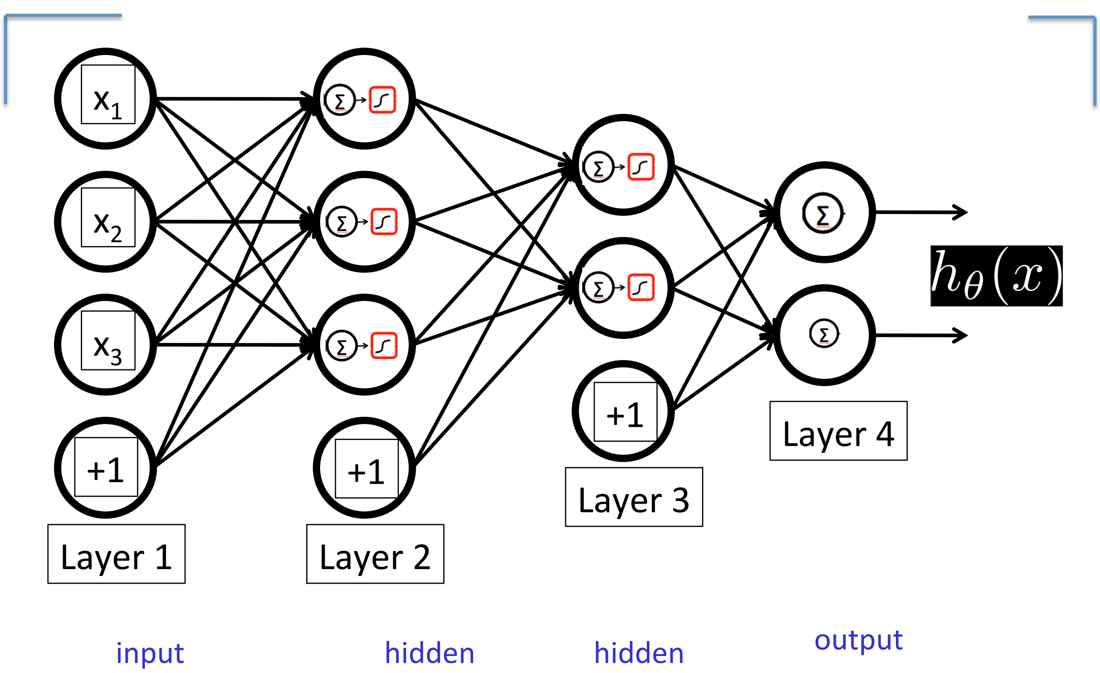
Multi-Layer Perceptron (MLP)

String a lot of logistic units together. Example: A 3 layer network:



Multi-Layer Perceptron (MLP)

Example: 4 layer network with 2 output units:

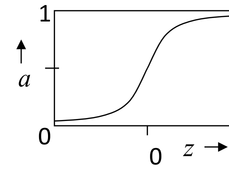


Transfer / Activation functions

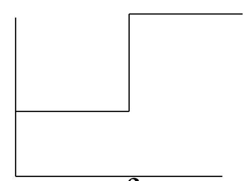
- Common ones include:
  - Threshold f(v) = 1 if v > c, else -1
  - Sigmoid (s shape func)
    - E.g. logistic func: $f(v) = 1/(1 + e^{-v})$, Range [0, 1]
    - E.g. hyperbolic tanh
  - Tanh $f(v) = (e^{v} - e^{-v})/(e^{v} + e^{-v})$, Range [-1,1]

- Desirable properties:
  - Monotonic, Nonlinear, Bounded
  - Easily calculated derivative

sigmoid func



$$a = \frac{1}{1 + e^{-z}}$$



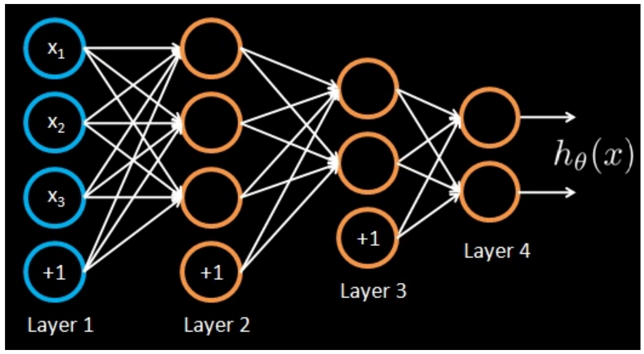
step func

Today

- Neural Network
  - MLP (Multilayer Perceptron Network)
  - Training of MLP

- Deep CNN, why Deep Learning ?

Training a neural network



Given training set (x_1, y_1), (x_2, y_2), (x_3, y_3), ….

Adjust parameters θ (for every node) to make: $h_\theta(x_i) \approx y_i$

(Use gradient descent. "Backpropagation" algorithm. Susceptible to local optima.)

- **Backpropagation**
  - Using backward recurrence to jointly optimize all parameters
  - Requires all activation functions to be differentiable
  - Enables flexible design in deep model architecture
  - Gradient descent is used to (locally) minimize objective:

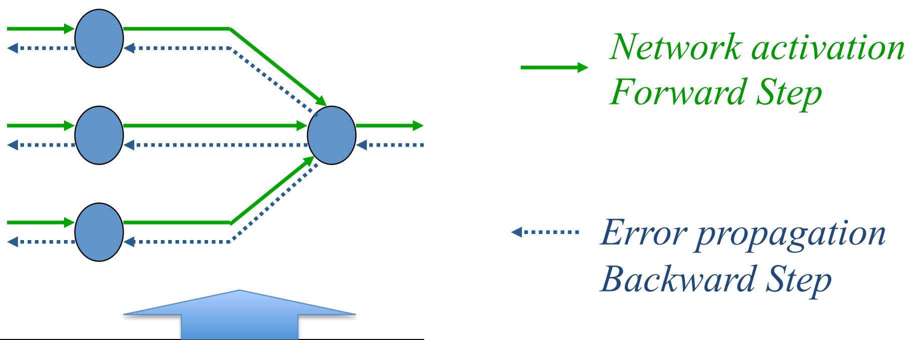
$$W^{k+1} = W^{k} - \eta \frac{\partial L}{\partial W^{k}}$$

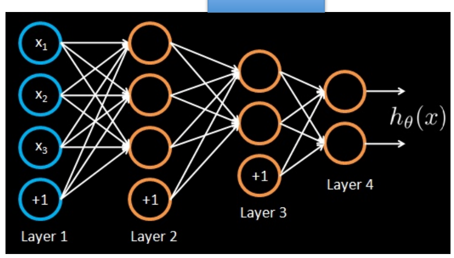
Y. LeCun et al. 1998. Efficient BackProp.

Olivier Bousquet and Ulrike von Luxburg. 2004. Stochastic Learning.

Backpropagation

- Back-propagation training algorithm



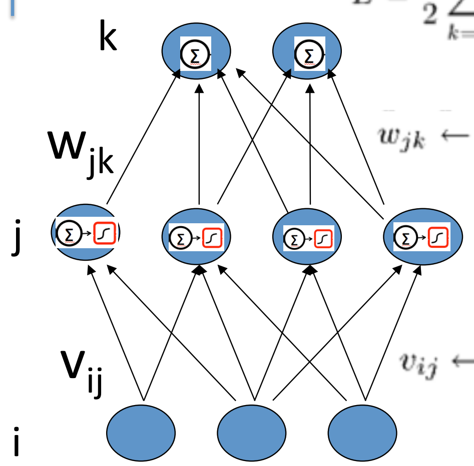


•**Stochastic Gradient Descent (SGD)** (first-order iterative optimization)

- •an **online learning** method
- •Approximates "true" gradient with a gradient at one data point
- •Attractive because of low computation requirement
- •Rivals **batch learning** methods on large datasets

When for Regression

$$E = \frac{1}{2}\sum_{k=1}^{N}(t_k - y_k)^2$$

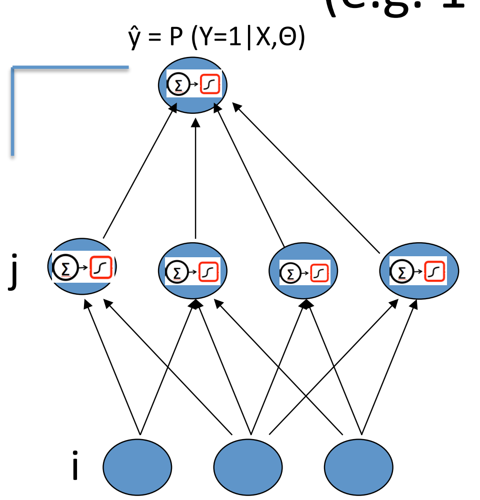


$$w_{jk} \leftarrow w_{jk} - \eta \frac{\partial E}{\partial w_{jk}}$$

$$v_{ij} \leftarrow v_{ij} - \eta \frac{\partial E}{\partial v_{ij}}$$

- Back Propagation adjusts the weights of the NN in order to minimize the network total mean squared error.

When for classification (e.g. 1 neuron for binary output)



When multi-class output, last layer is softmax output layer ➔ multinomial logistic regression unit

For Bernoulli distribution,

$$p(y=1|x)^y(1-p)^{1-y}$$

$$Loss(\theta) = -\sum_{i=1}^{N}\{\log \Pr(Y = y_i | X = x_i)\} = -\sum_{i=1}^{N} y_i \log(\hat{y}_i) + (1-y_i)\log(1-\hat{y}_i)$$

Cross-entropy loss function, OR deviance

Today

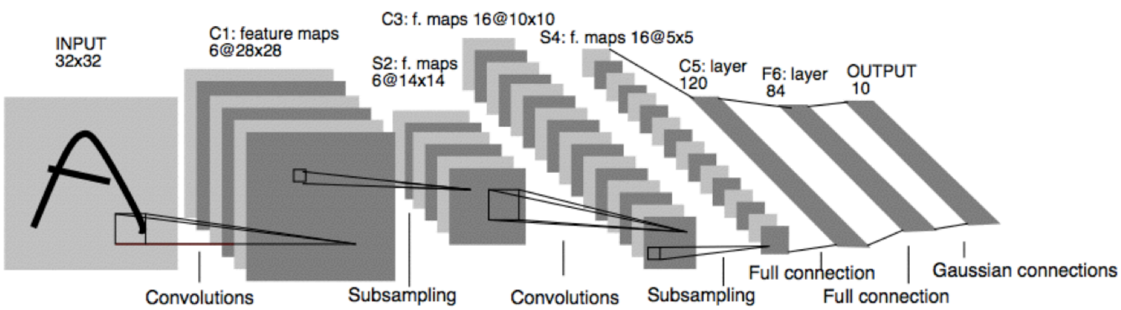
- Neural Network
  - MLP (Multilayer Perceptron Network)
  - Training of MLP
- Deep Learning
  - History
  - Application

Classification models since late 80's

- Neural networks
- Boosting
- Support Vector Machine
- Maximum Entropy
- Random Forest
- ……

Deep Learning in the 90's

- Yann LeCun invented Convolutional Networks
- First NN successfully trained with many layers

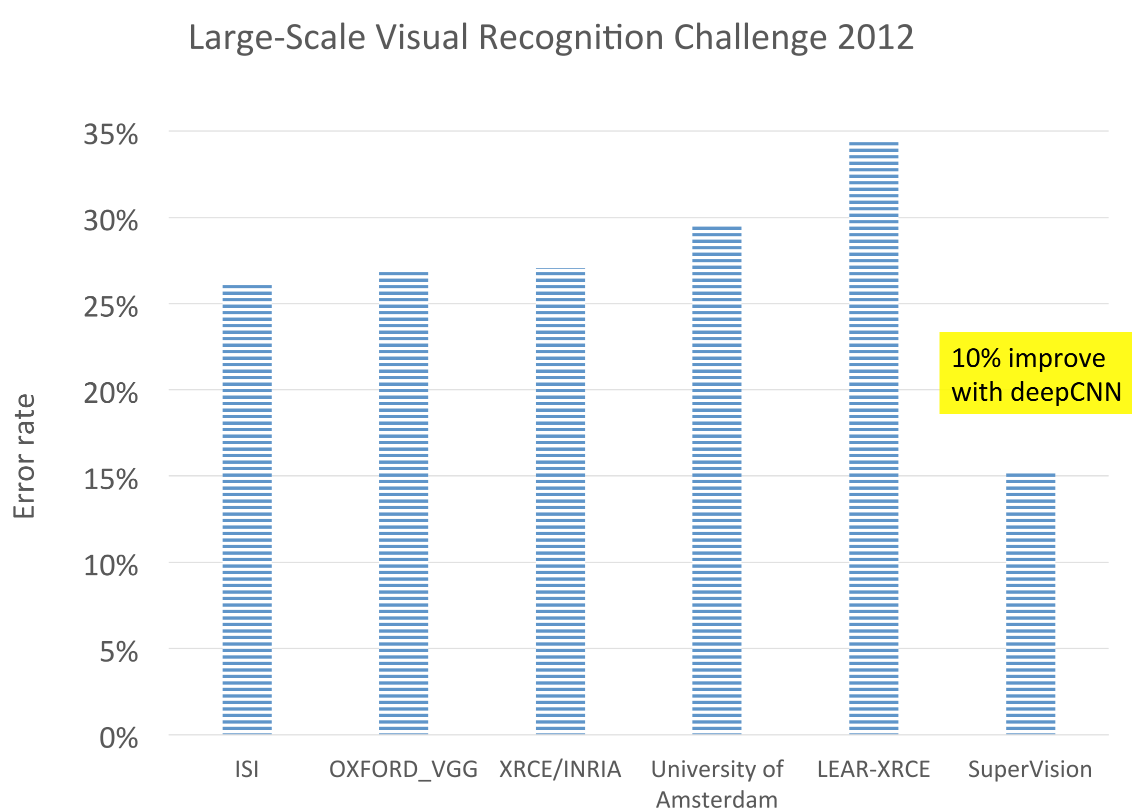


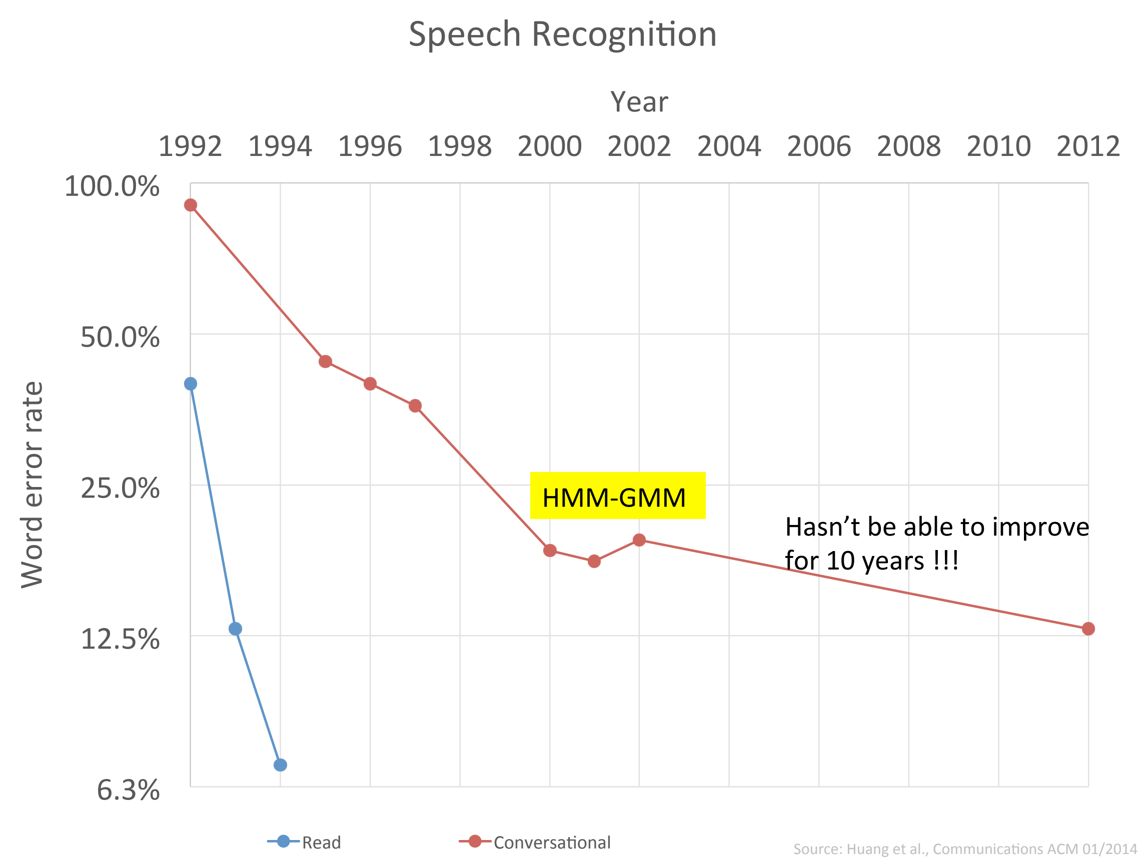
Since 2000-2011

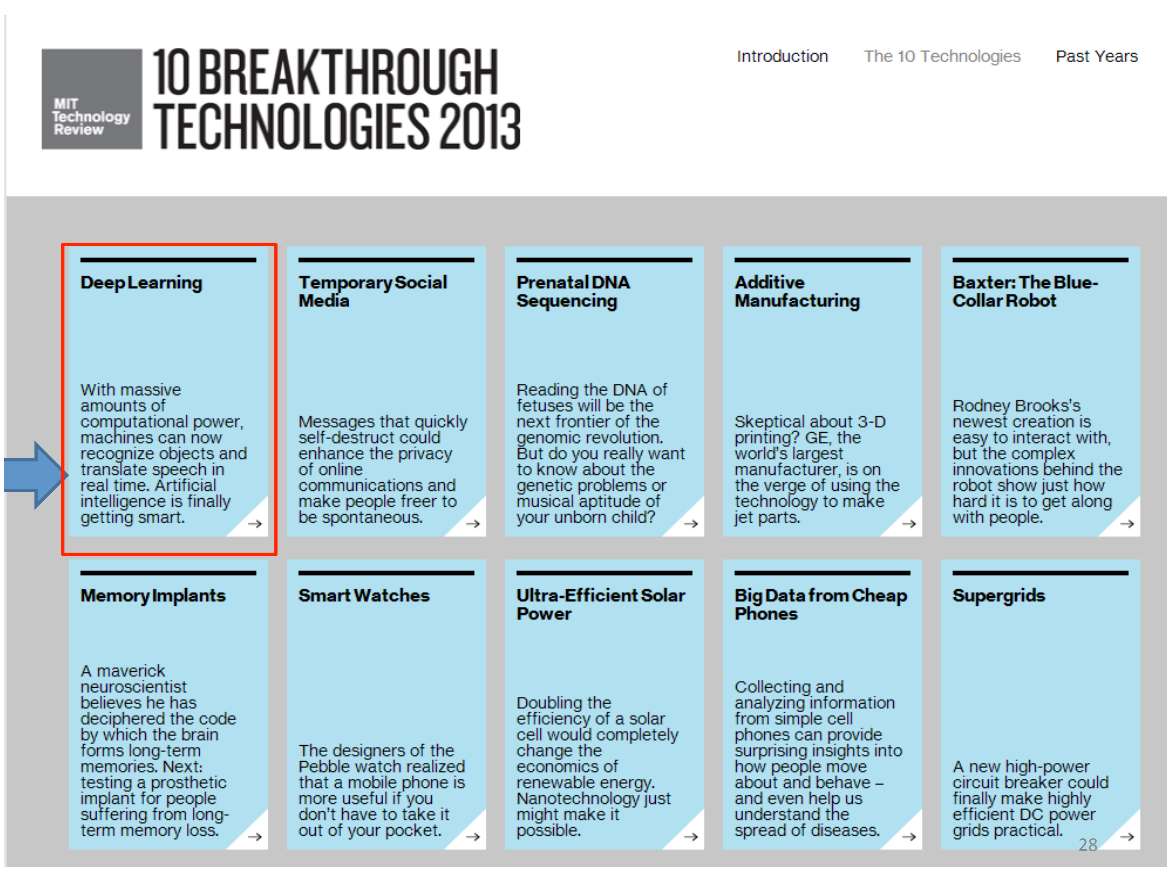
- Learning with Structures !
  - Kernel learning
  - Transfer Learning
  - Semi-supervised
  - Manifold Learning
  - Sparse Learning
  - Structured input-output learning …

"Winter of Neural Networks" Since 90's !

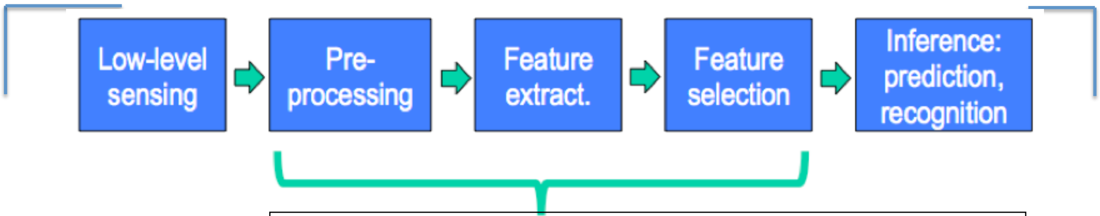
- Non-convex
- Need a lot of tricks to play with
- Hard to perform theoretical analysis




Speech Recognition
Year
1992 1994 1996 1998 2000 2002 2004 2006 2008 2010 2012
100.0%
50.0%
25.0%
12.5%
6.3%
Word error rate
HMM-GMM
Hasn't be able to improve for 10 years !!!
Read
Conversational
Source: Huang et al., Communications ACM 01/2014


MIT Technology Review
10 BREAKTHROUGH TECHNOLOGIES 2013
Introduction
The 10 Technologies
Past Years
Deep Learning
With massive amounts of computational power, machines can now recognize objects and translate speech in real time. Artificial intelligence is finally getting smart.
Temporary Social Media
Messages that quickly self-destruct could enhance the privacy of online communications and make people freer to be spontaneous.
Prenatal DNA Sequencing
Reading the DNA of fetuses will be the next frontier of the genomic revolution. But do you really want to know about the genetic problems or musical aptitude of your unborn child?
Additive Manufacturing
Skeptical about 3-D printing? GE, the world's largest manufacturer, is on the verge of using the technology to make jet parts.
Baxter: The Blue-Collar Robot
Rodney Brooks's newest creation is easy to interact with, but the complex innovations behind the robot show just how hard it is to get along with people.
Memory Implants
A maverick neuroscientist believes he has deciphered the code by which the brain forms long-term memories. Next: testing a prosthetic implant for people suffering from long-term memory loss.
Smart Watches
The designers of the Pebble watch realized that a mobile phone is more useful if you don't have to take it out of your pocket.
Ultra-Efficient Solar Power
Doubling the efficiency of a solar cell would completely change the economics of renewable energy. Nanotechnology just might make it possible.
Big Data from Cheap Phones
Collecting and analyzing information from simple cell phones can provide surprising insights into how people move about and behave – and even help us understand the spread of diseases.
Supergrids
A new high-power circuit breaker could finally make highly efficient DC power grids practical.

Deep Learning Way:
Learning features / Representation from data



Feature Engineering
- ✓ Most critical for accuracy
- ✓ Account for most of the computation for testing
- ✓ Most time-consuming in development cycle
- ✓ Often hand-craft and task dependent in practice

Feature Learning
- ✓ Easily adaptable to new similar tasks
- ✓ Layerwise representation
- ✓ Layer-by-layer unsupervised training
- ✓ Layer-by-layer supervised training

Today

- Neural Network
  - MLP (Multilayer Perceptron Network)
  - Training of MLP
- Deep Learning
  - History
  - Applications