Deep Learning

Davide Bacciu

Dipartimento di Informatica
Università di Pisa
bacciu@di.unipi.it

Machine Learning: Neural Networks and Advanced Models (AA2)

Today's Lecture

- An introduction to deep learning
  - What it is
  - Why it is so big now
- Deep learning architectures
  - Deep Belief Networks
  - Stacked Autoencoders
- Application examples

What is Deep Learning?

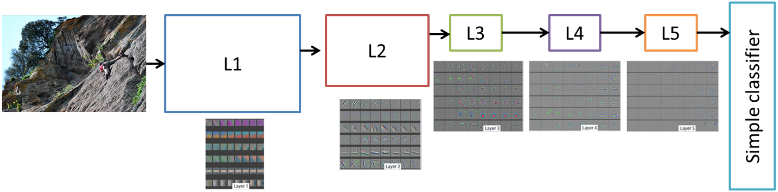
Machine learning algorithms inspired by brain organization, based on learning multiple levels of representation and abstraction

- Learning models with many layers trained layer-wise
- Build a hierarchical feature space through layering
- Reduce the need of supervised information
  - Unsupervised discovery of features in the internal layers
  - Final layer performs supervised step (any recall?)

Hierarchical Features

The traditional shallow way

The deep way



Why Deep Learning?

Google

DEEPMIND

DNNresearch

facebook

No, Seriously.. Why Deep Learning?

- Deep learning is THE hot topic now
- Revolutionized performance in
  - Speech recognition
  - Machine vision
- Now expanding to other topics
  - Natural language
  - Reinforcement learning
  - Robotics

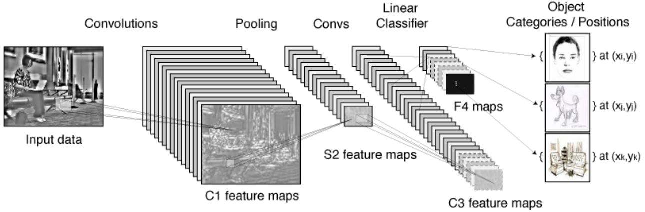
Performs best when input information has some form of structure, e.g. spatial, temporal, ...

The concept

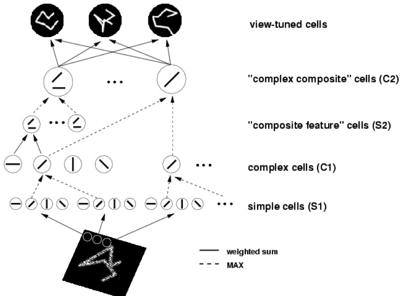
- Learning effective and efficient representation of complex data
- Learning hierarchical feature representation
- Learning distributed feature representation
- Exploit unlabeled and/or partially-labeled data

Foundations - Hierarchical Representation

Convolutional Neural Networks (CNNs)



Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, Gradient-based learning applied to document recognition, Proceedings of the IEEE 86(11): 2278-2324, 1998

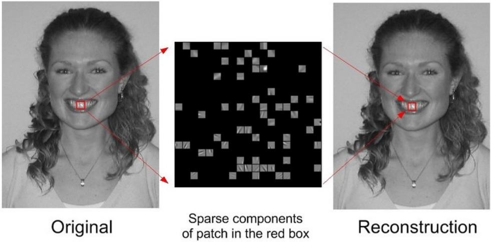


HMAX model - Inspired by the structure of the visual cortex in mammals

M. Riesenhuber, T. Poggio, Hierarchical Models of Object Recognition in Cortex. Nature Neuroscience 2: 1019-1025, 1999

Foundations - Distributed Representation

Originates from sparse coding theory in brain: sensory information in the brain is represented by a relatively small number of simultaneously active neurons out of a large population (B.A. Olshausen, D.J. Field, 1996)



The machine learning perspective: a single layer network learns better to generate a target output if the input has a sparse representation (Willshaw and Dayan, 1990)

How Deep is "Deep Learning"?

- Conventional neural networks
  - 1-layer - Linear Classifiers: logistic regression, naive Bayes
  - 2-layers - Universal approximators: MLP, RBFnet, nonlinear SVM
- 3-layers or more
  - MLPs, RNN, decision trees
  - Deep learning
- It does not suffice to stack layers to do deep learning
  - Need to have the right computational elements
  - Weighted sum, product, max operators, single neuron, kernels, ...

Deep Networks

- Deep architecture with multiples layers focused on learning sparse encoding of input data
- Unsupervised training between layers to decompose the problem into distributed sub problems with increasing levels of abstraction
- Deep networks type
  - Deep Belief Networks
  - Stacked Autoencoders
  - Hybrid

Training Deep Networks

Problem with conventional backpropagation training

- Strongly relies on labeled training data
- Learning does not scale well to multiple hidden layers (BPTT)

Greedy layer-wise training

- Pre-training - unsupervised (internal) layer by layer training
- Read-out - supervised training of last layer
- Fine-tuning - supervised adjustment of all weights

Key advantages

- Give full learning focus to each layer
- Exploit unlabeled data
- Use supervised training only for fine tuning with weights hopefully close to global maxima

Deep Belief Networks

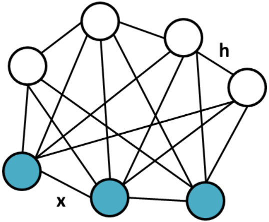
Intuition - Discover hidden (latent) features $\mathbf{h}$ which represent well input-data $\mathbf{x}$ (observable)

- Directed latent variable models?
  - Marginally independent causes $\mathbf{h}$ can become dependent given evidence $\mathbf{x}$
  - Deep features posterior $P(\mathbf{h}|\mathbf{x})$ can be intractable
- Undirected models?
  - Energy-based models (Boltzmann machines)

$$P(\mathbf{x}, \mathbf{h}) = \frac{1}{Z} \exp\left(E(\mathbf{x}, \mathbf{h})\right)$$

  - Problem is intractability of Z factor

Boltzmann Machines



A variant of the Ising model

- Visible RV $x \in \{0, 1\}$
- Latent RV $h \in \{0, 1\}$

- A linear energy function

$$E(\mathbf{x}, \mathbf{h}) = -\frac{1}{2}\mathbf{x}^T\mathbf{U}\mathbf{x} - \frac{1}{2}\mathbf{h}^T\mathbf{V}\mathbf{h} - \mathbf{x}^T\mathbf{W}\mathbf{h} - \mathbf{b}^T\mathbf{x} - \mathbf{d}^T\mathbf{h}$$

- Model parameters $\theta = \{\mathbf{U}, \mathbf{v}, \mathbf{W}, \mathbf{b}, \mathbf{d}\}$ encode the interactions between the variables (observable and not)
- Posterior inference intractable due to the (exponential) marginalization term

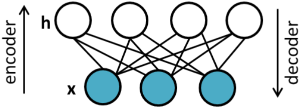
Boltzmann machines are a type of Recurrent Neural Network

Restricted Boltzmann Machines (RBM)

A special Boltzmann machine

- Bipartite graph
- Connections only between hidden and observable nodes

$$E(\mathbf{x}, \mathbf{h}) = -\mathbf{x}^T \mathbf{W} \mathbf{h} - \mathbf{b}^T \mathbf{x} - \mathbf{d}^T \mathbf{h}$$

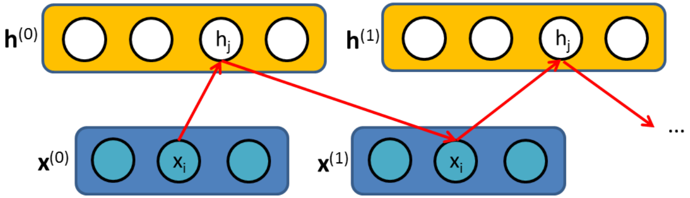


Used to implement layers of a Deep Network

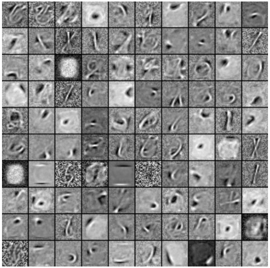
- Observable nodes are inputs
- Hidden nodes are a latent feature representation of the input
- Tractable inference due to graph bipartition which factorizes posterior distribution

Training RBM

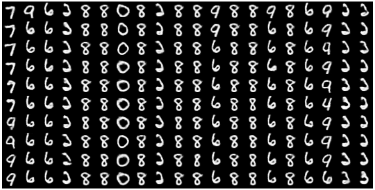
- Gradient ascent of the RBM log-likelihood (Computationally intensive)
- Gibbs sampling: Iterative sample **x** then **h** (Slow convergence)
- Contrastive-Divergence
  - A form of alternating Gibbs sampling, iterated few times
  - Feed training input to observable nodes and update all hidden units posterior (encoding)
  - Update all visible units to get a reconstruction from hidden units (decoding); update again all hidden units



RBM - Character Recognition Example



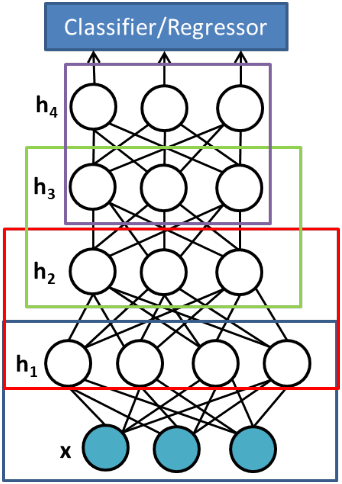
Learned latent features (filters)



RBM reconstructed inputs using Gibbs sampling

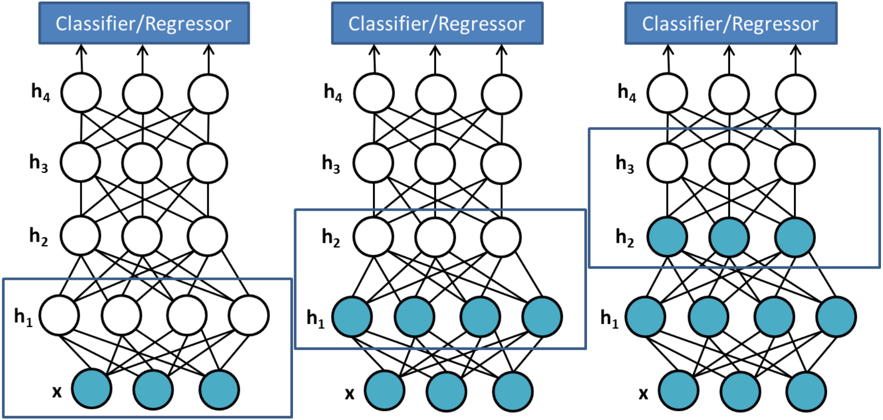
Deep Belief Network - Architecture

A network of stacked RBM plus a supervised read-out layer



Deep Belief Network - Training

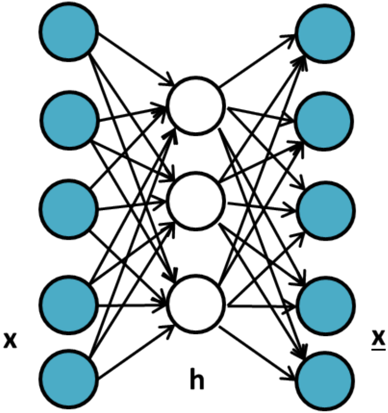
Layer-wise training of the RBM (e.g. Contrastive Divergence)



Train the read-out layer (independent learning model,additional RBM layer,...)

Autoencoder Networks

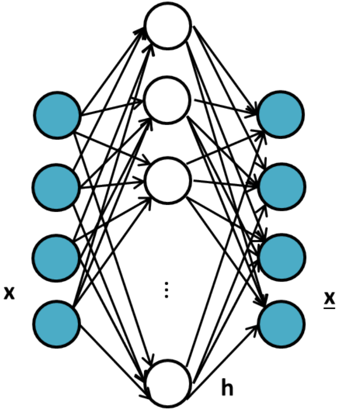
Neural networks for feature discovery and data compression



E.g. linear hidden units with MSE perform PCA

Sparse Autoencoders

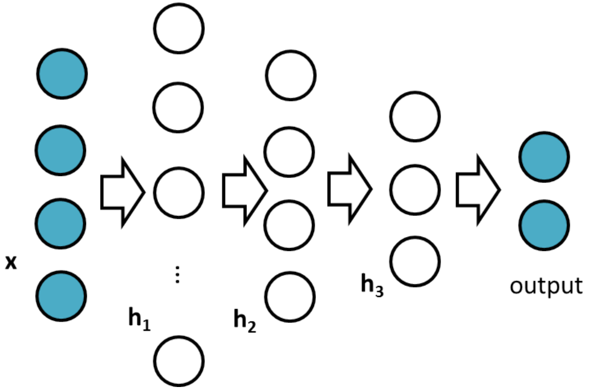
Autoencoders using more features with a significant number being 0 when encoding an input



Using regularization approaches to enforce sparsity

Stacked Autoencoders

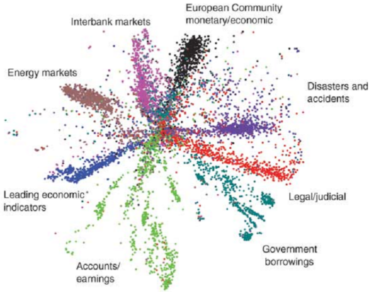
Stack autoencoders with level-wise training as with RBM



Train each autoencoder in isolation and drop the decoding layer when training is completed

Document Encoding

Finding sparse features for $> 800K$ newswire stories

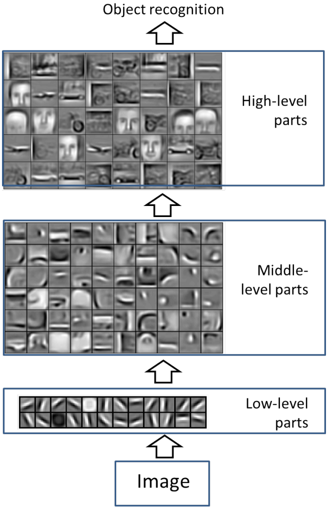


RBM document encoding

Latent semantic analysis encoding

Hinton, Salakhutdinov, Reducing the dimensionality of data with neural networks, Science, 2006

Convolutional DBN for Hierarchical Object Recognition

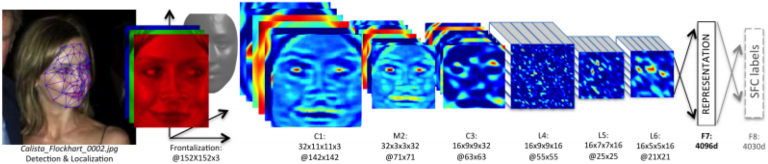


Obtaining a hierarchical representation of object parts through a deep convolutional network

Lee et al, Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations,

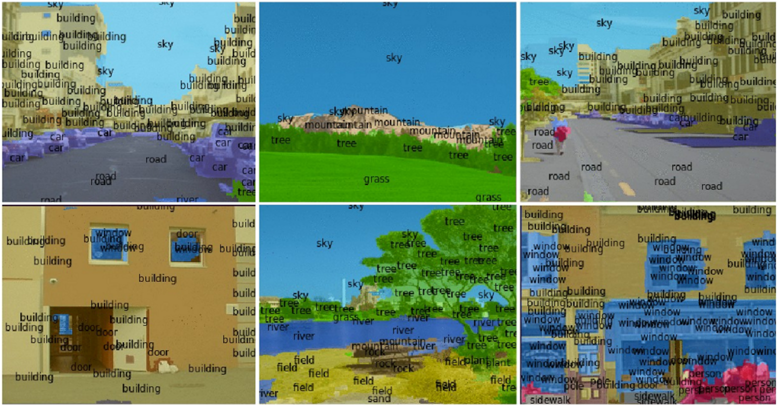
Deep Faces

DeepFace recognition accuracy is $\approx$ 97.35% (23% better than previous results)



Taigman et al, Deepface: Closing the gap to human-level performance in face verification, CVPR 2014

Scene Understanding



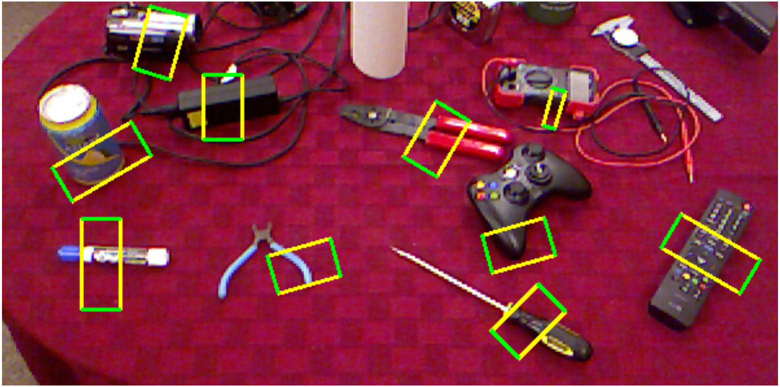
Farabet et al, Learning Hierarchical Features for Scene Labeling, TPAMI 2013

Want to try something out?

- The Caffe project at Berkeley:
  `http://demo.caffe.berkeleyvision.org/`
- The Zeiler-Fergus image classifier at NYU:
  `http://horatio.cs.nyu.edu/`

Deep Learning and Robotics

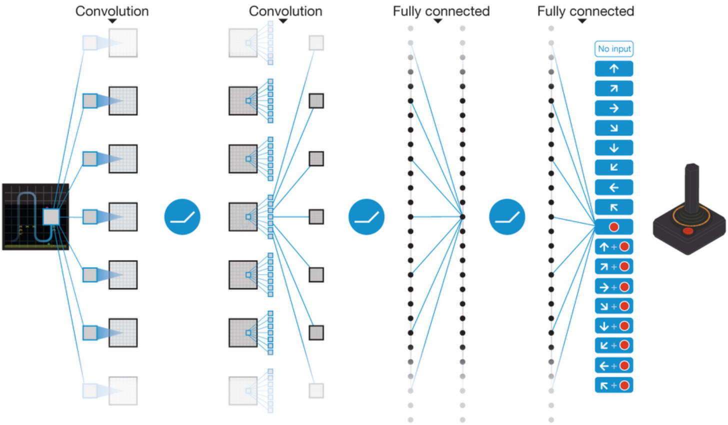
Learning where to grasp objects with robotic manipulators



Lenz et al, Deep Learning for Detecting Robotic Grasps, IJRR 2014

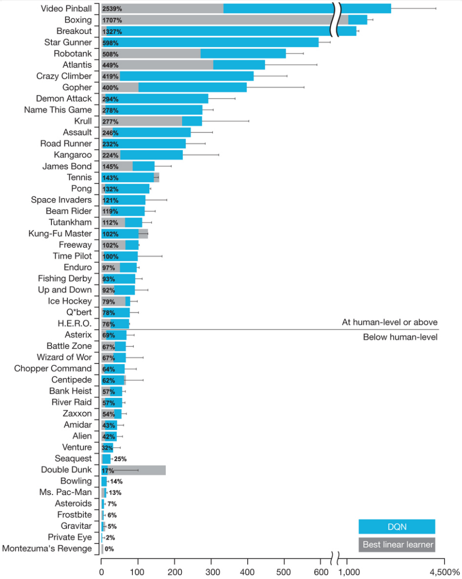
Learning to Play Atari (I)

Using Deep Learning with Reinforcement Learning



Mnih et al, Human-level control through deep reinforcement learning, Nature 2015

Learning to Play Atari (II)



Take Home Messages

- Deep learning is about learning features of complex data
  - Probabilistic interpretation: features $\equiv$ latent factors (RBM)
  - Neural interpretation: features $\equiv$ hidden neurons (Neural autoencoders)
  - Bridging neural and probabilistic world
- Stacking and level-wise training
  - Let the deep network discover the features and then place your preferred learning model to perform your task
- Breakthrough performance in several learning tasks/application areas
  - Complex spatio/temporal structured data
- Do we really know what is going on with the encoding? How many layers?