Fundamentals of Deep Learning of Representations

UPCOMING MIT PRESS BOOK
DRAFT CHAPTERS AVAILABLE
ON MY WEB PAGE.

Tel-Aviv University
Deep Learning Master Class

Yoshua Bengio

November 7, 2014, Tel-Aviv

Université de Montréal

LISA

Ultimate Goal

- **Understand the principles giving rise to intelligence**

Focus

- **Learning:** mathematical and computational principles allowing one to learn from examples in order to acquire knowledge

Breakthrough

- **Deep Learning: machine learning algorithms inspired by brains, based on learning multiple levels of representation / abstraction.**

Impact

Deep learning has revolutionized

- Speech recognition
- Object recognition

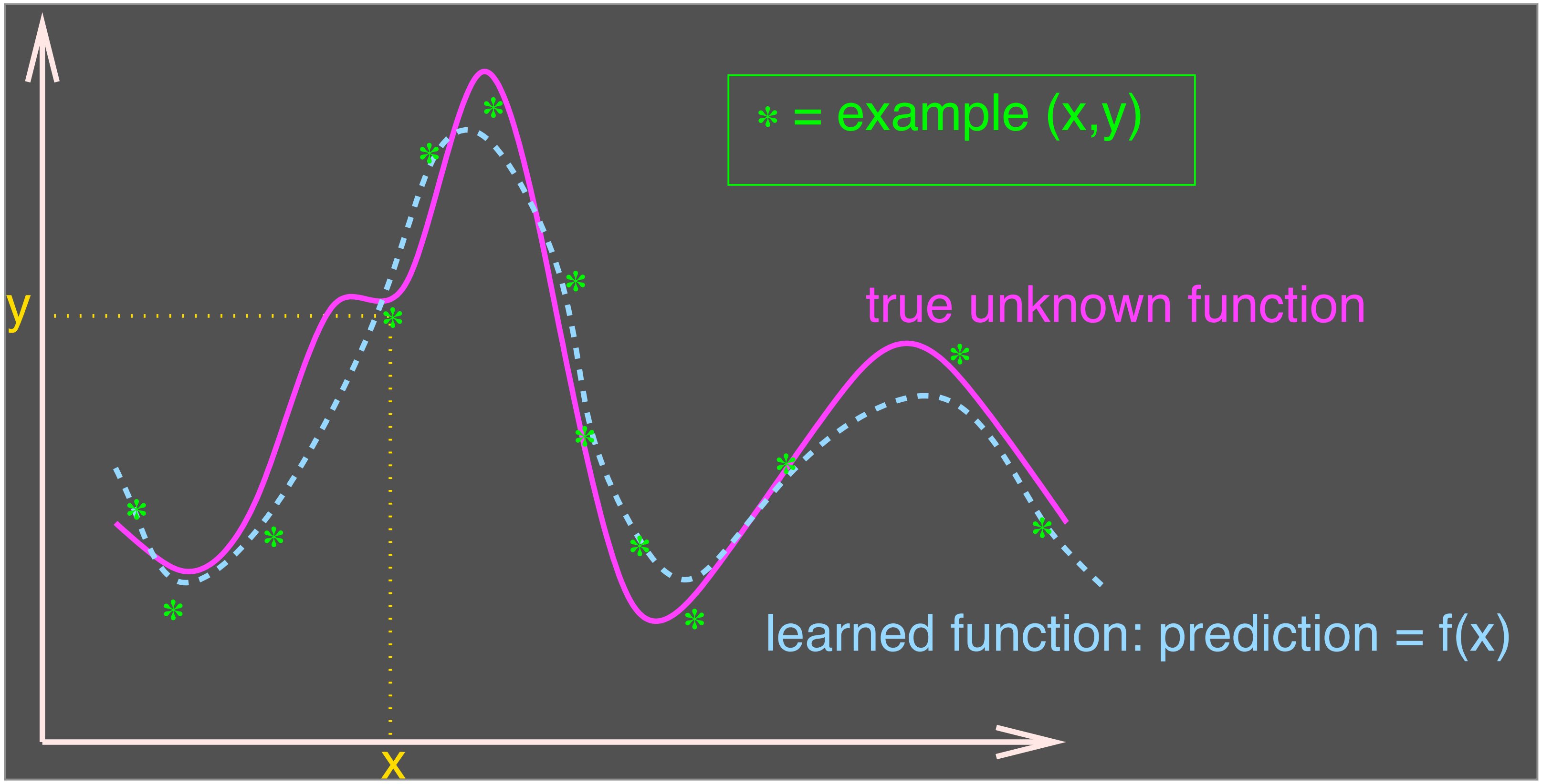
More coming, including other areas of computer vision, NLP, machine translation, dialogue, reinforcement learning…

Technical Goals Hierarchy

To reach AI:

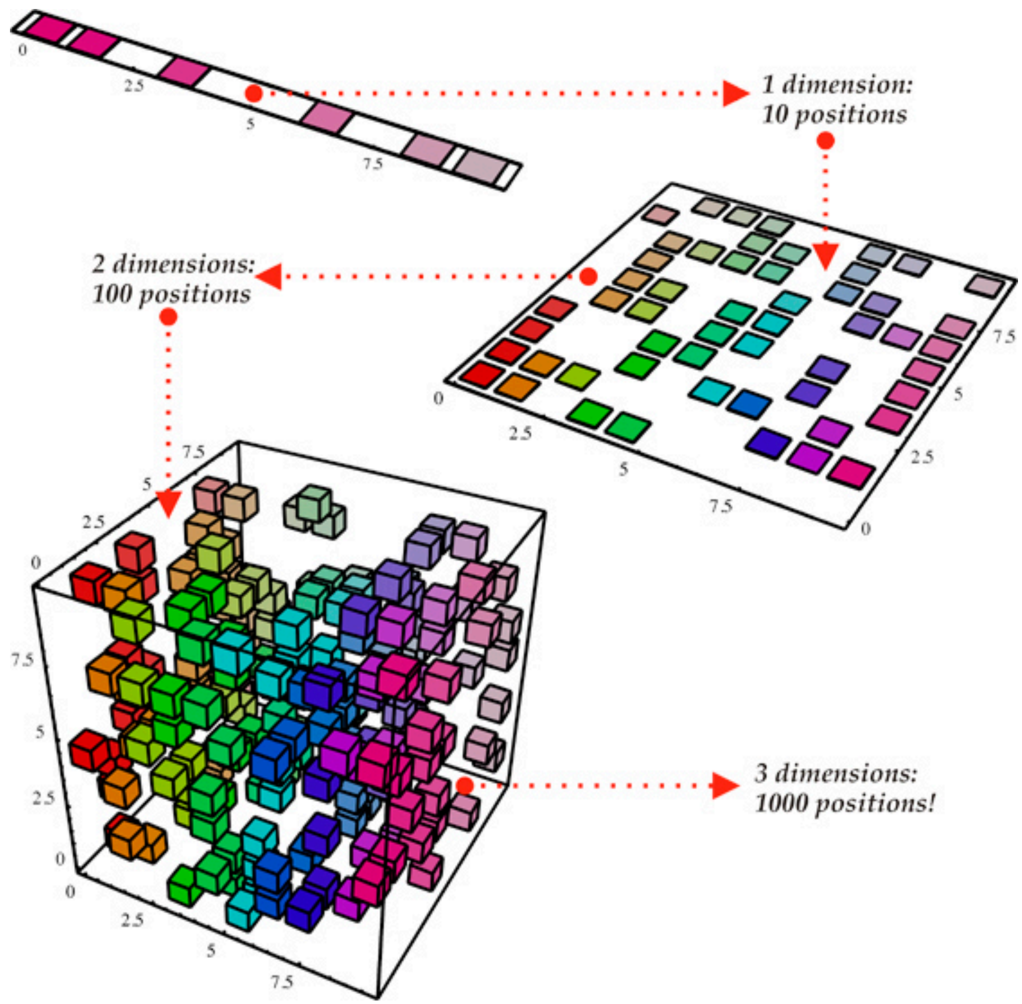
- Needs **knowledge**
- Needs **learning**
  (involves priors + ***optimization***/*search* ***+ efficient computation***)
- Needs **generalization**
  (guessing where probability mass concentrates)
- Needs ways to fight the curse of dimensionality
  (exponentially many configurations of the variables to consider)
- Needs disentangling the underlying explanatory factors
  (making sense of the data)

Easy Learning



ML 101. What We Are Fighting Against: The Curse of Dimensionality

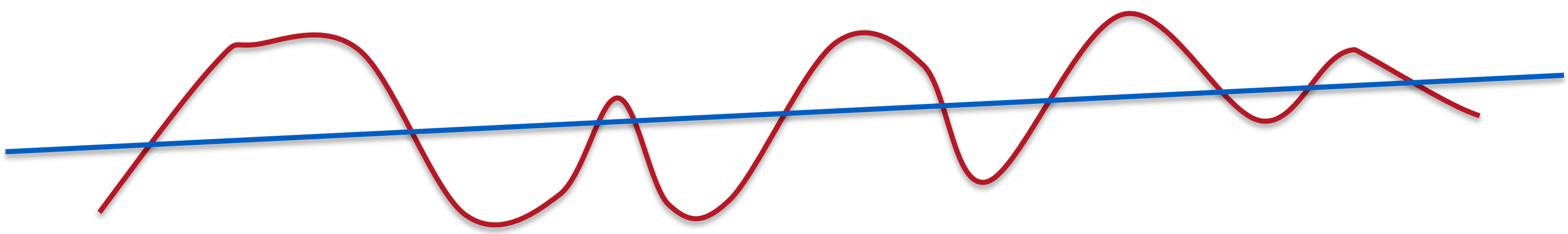
To generalize locally, need representative examples for all relevant variations!



Not Dimensionality so much as Number of Variations

(Bengio, Dellalleau & Le Roux 2007)

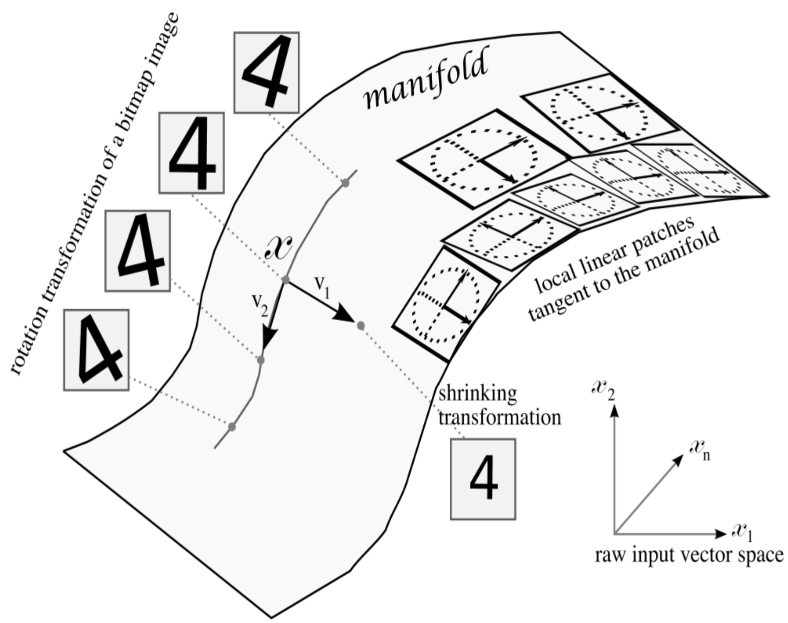
- Theorem: Gaussian kernel machines need at least *k* examples to learn a function that has *2k* zero-crossings along some line

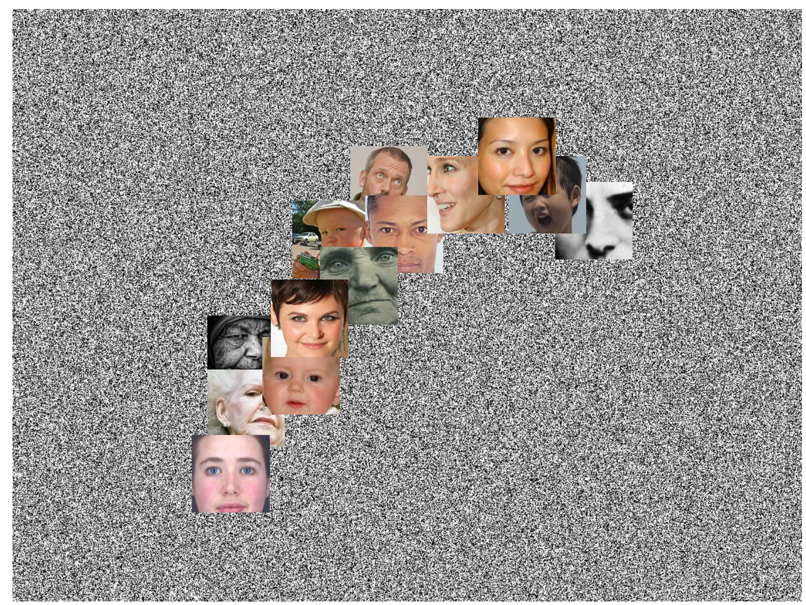


- Theorem: For a Gaussian kernel machine to learn some maximally varying functions over *d* inputs requires $O(2^d)$ examples

For AI Tasks: Manifold structure

- examples **concentrate** near a lower dimensional "manifold
- **Evidence: most input configurations are unlikely**





Representation Learning

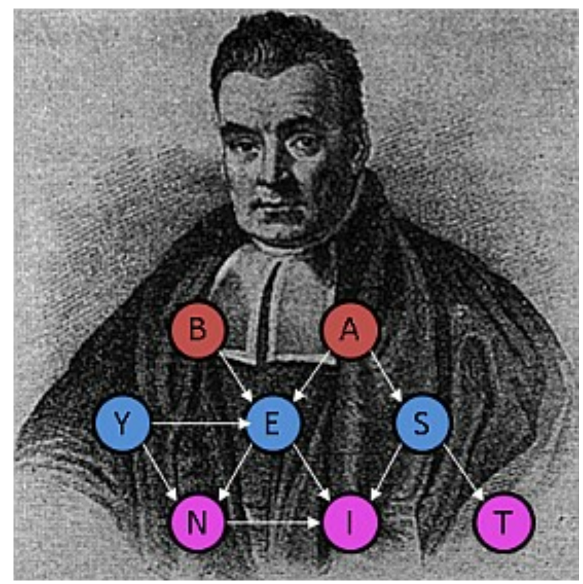
- Good **features** essential for successful ML: 90% of effort

raw input data → represented by chosen learned features → MACHINE LEARNING

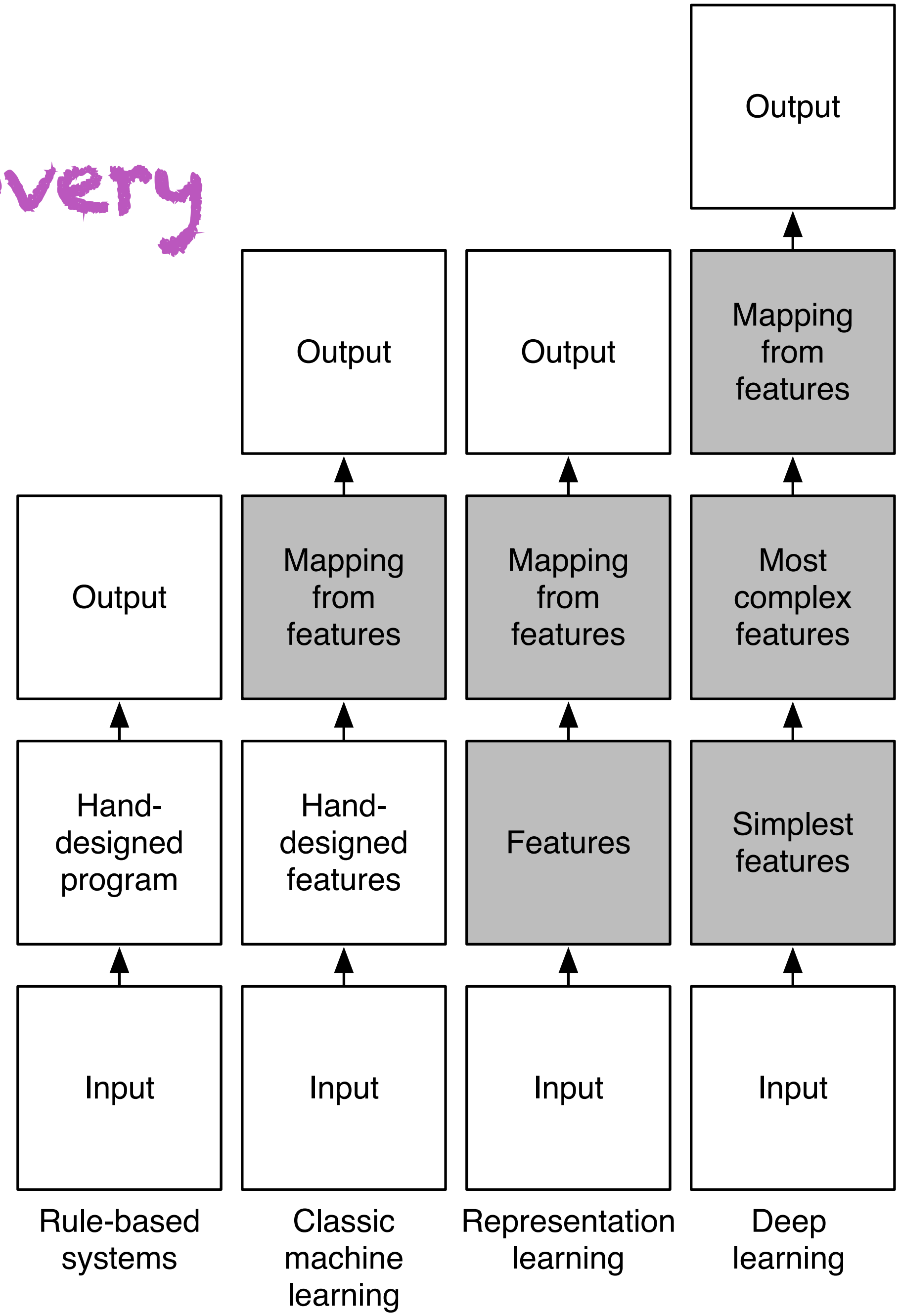
- Handcrafting features vs learning them

- Good representation?
- guesses

  the features / factors / causes



Automating Feature Discovery



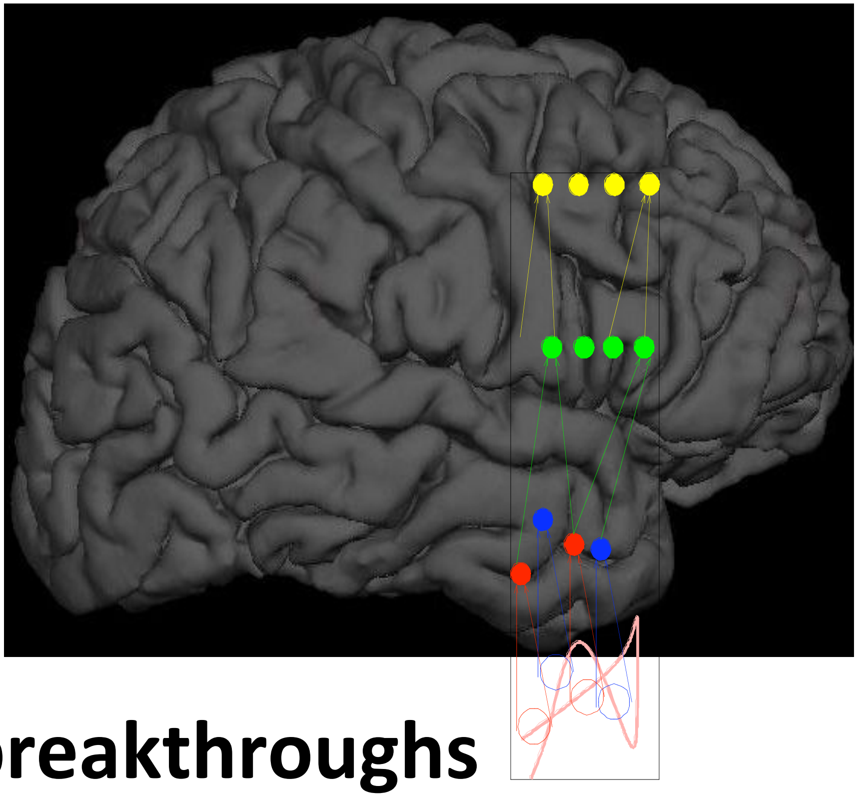
Learning multiple levels of representation

There is theoretical and empirical evidence in favor of multiple levels of representation

Exponential gain for some families of functions

Biologically inspired learning

- Brain has a deep architecture
- Cortex seems to have a generic learning algorithm
- **Humans first learn simpler concepts and compose them**



It works! Speech + vision breakthroughs

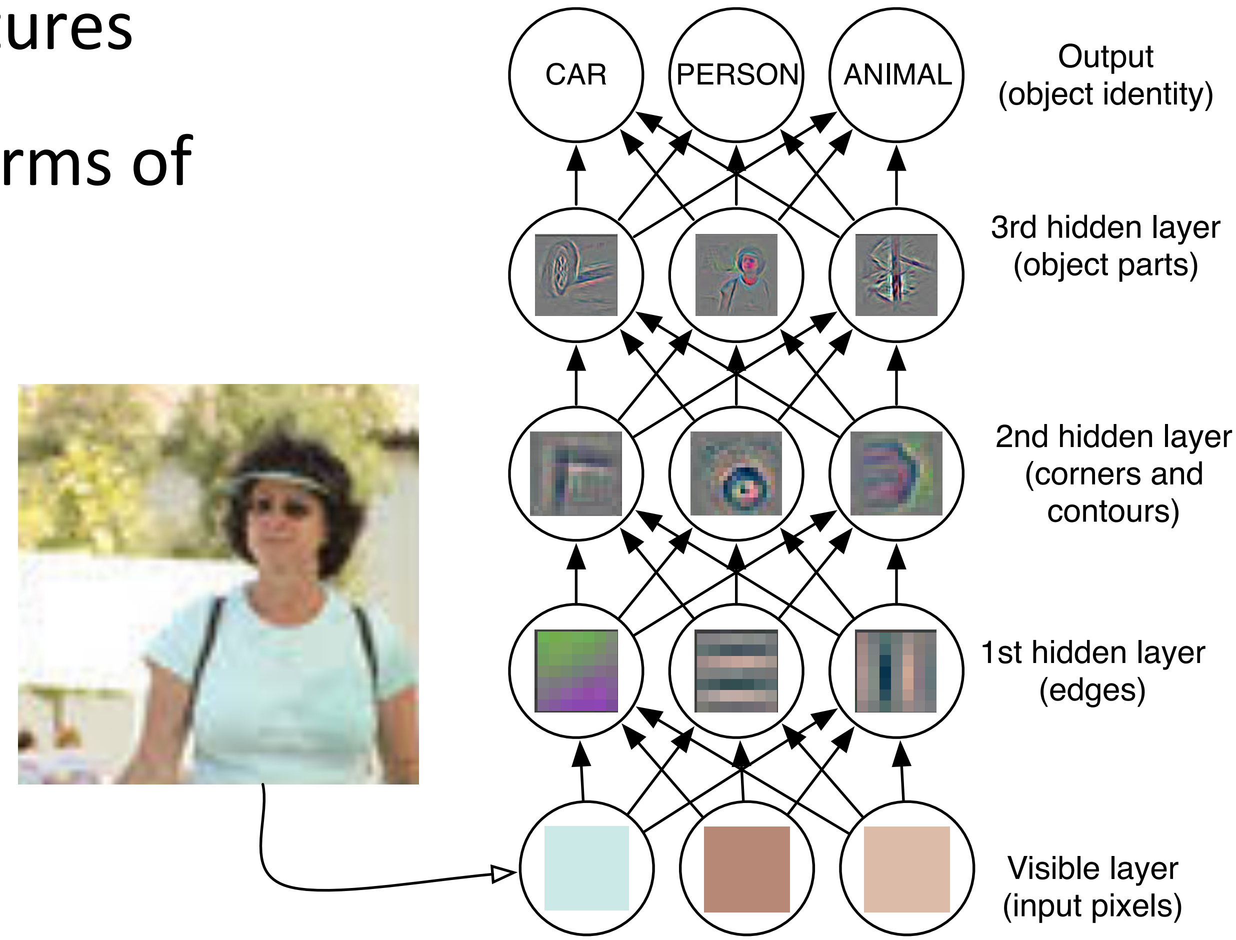
Composing Features on Features

Higher-level features

are defined in terms of

lower-level

features



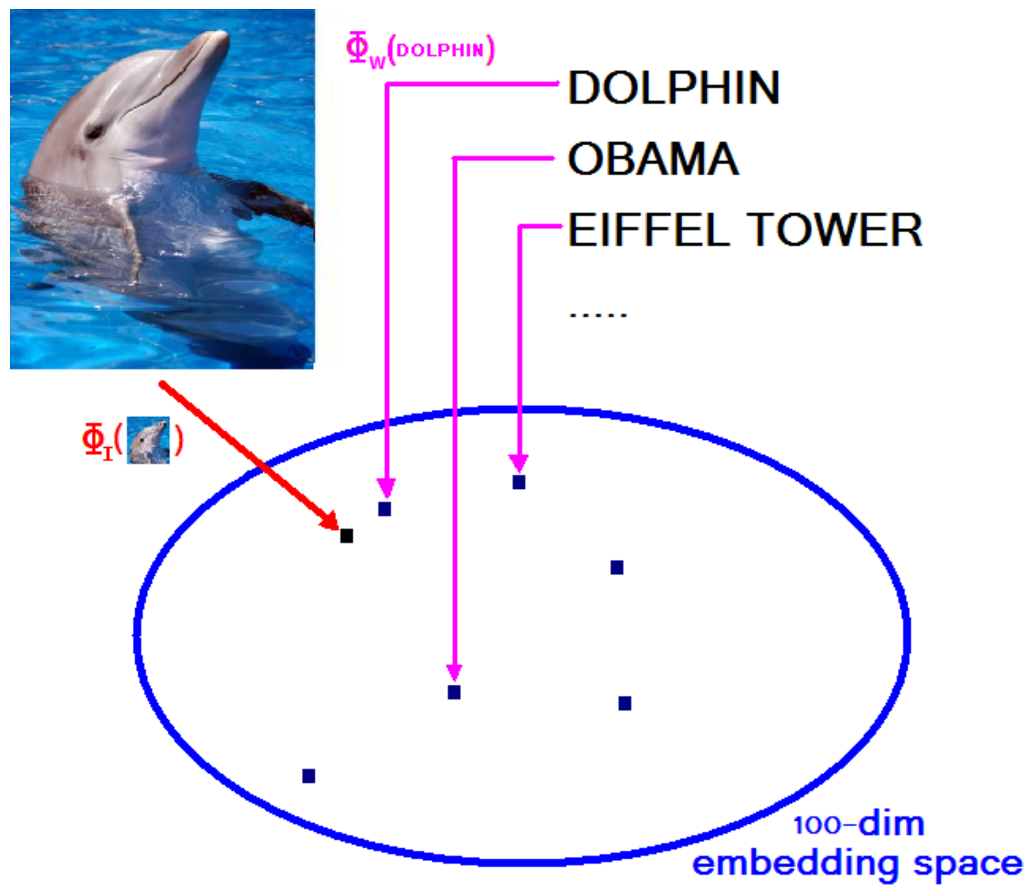
Google Image Search:
Different object types represented in the same space



Google:

S. Bengio, J. Weston & N. Usunier

(IJCAI 2011, NIPS'2010, JMLR 2010, MLJ 2010)

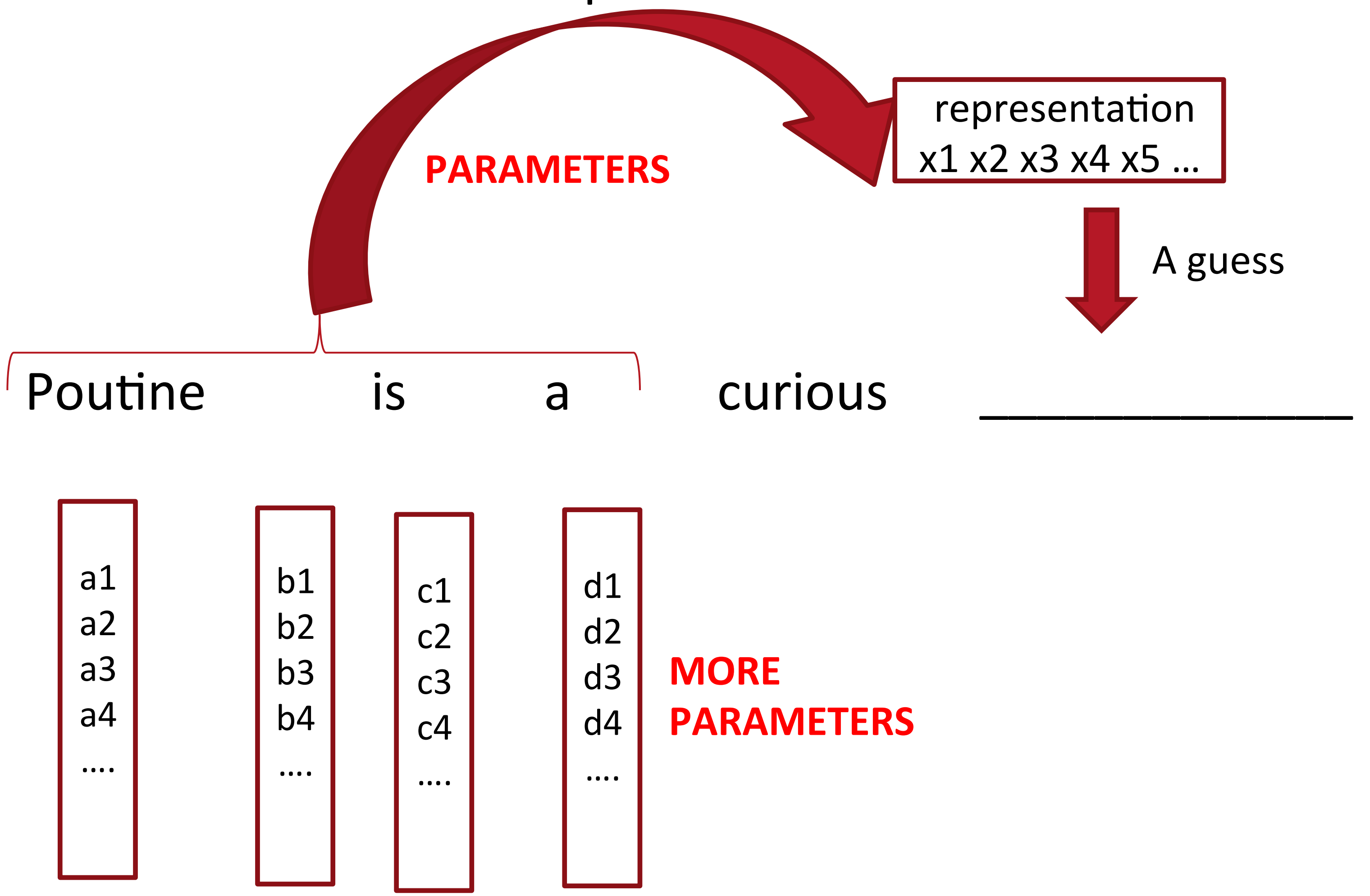


Learn $\Phi_I(\cdot)$ and $\Phi_W(\cdot)$ to optimize precision@k.

Following up on (Bengio et al NIPS'2000) Neural word embeddings - visualization

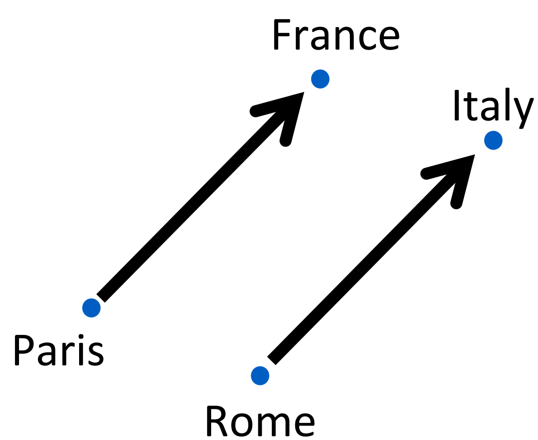
Neural Language Models

- Meanings and their combination all ‘learned’ together. Minimal structure imposed.



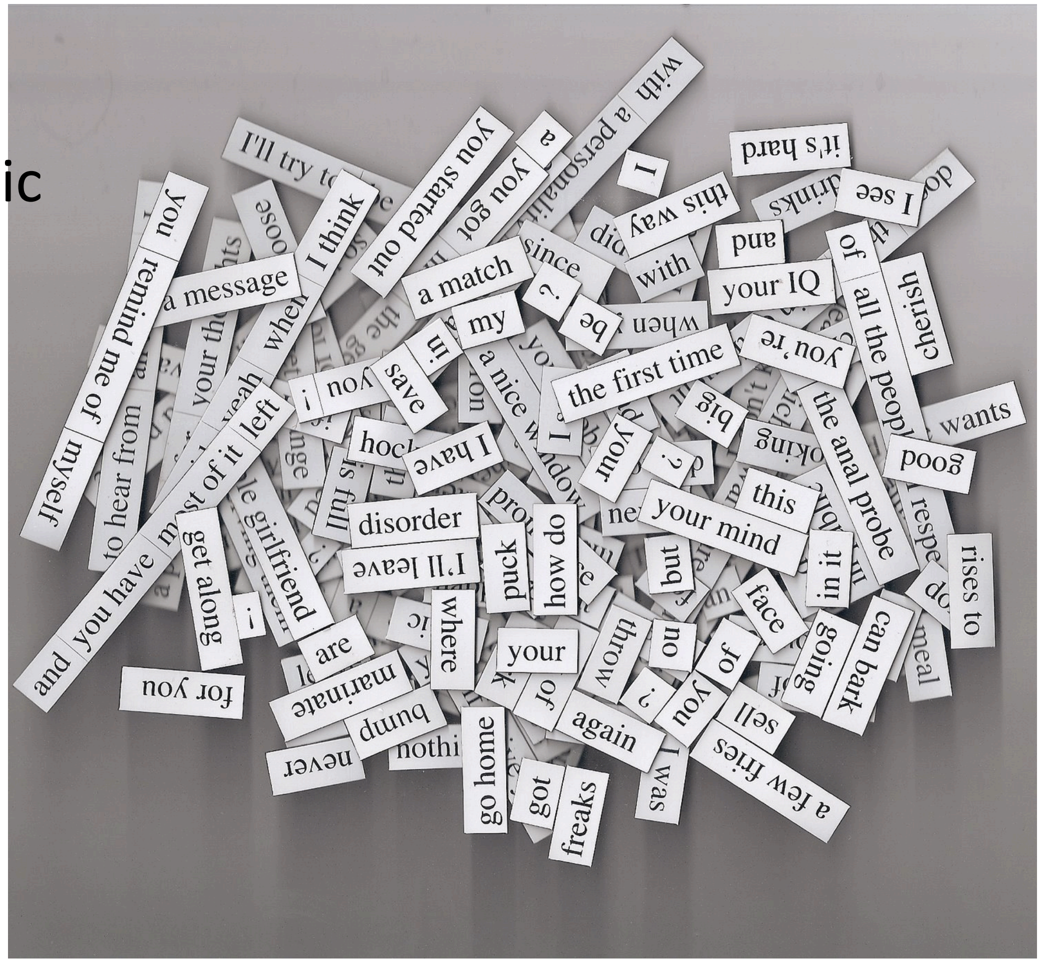
Analogical Representations for Free (Mikolov et al, ICLR 2013)

- Semantic relations appear as linear relationships in the space of learned representations
- King – Queen ≈ Man – Woman
- Paris – France + Italy ≈ Rome



The Next Challenge: Rich Semantic Representations for Word Sequences

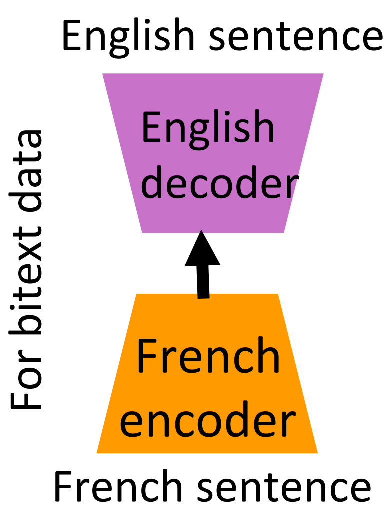
- Impressive progress in capturing word semantics
  Easier learning: non-parametric (table look-up)
- Optimization challenge for mapping sequences to rich & complete representations
- Good test case: machine translation

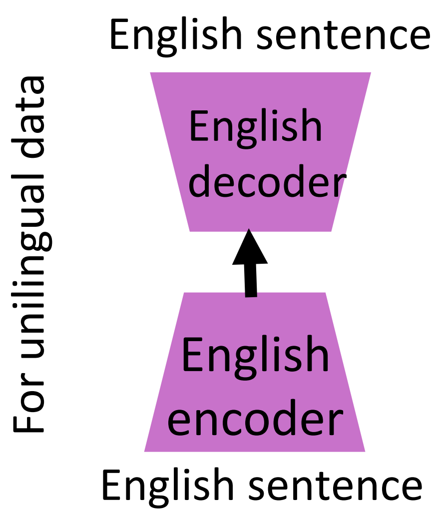


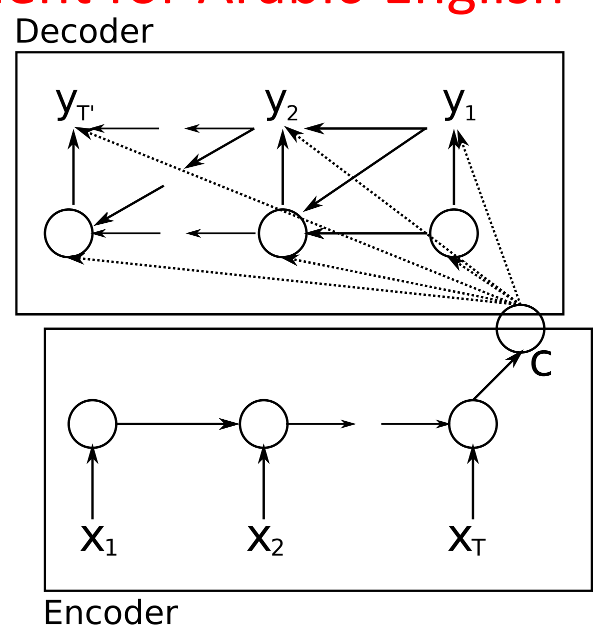
Breakthroughs in Machine Translation

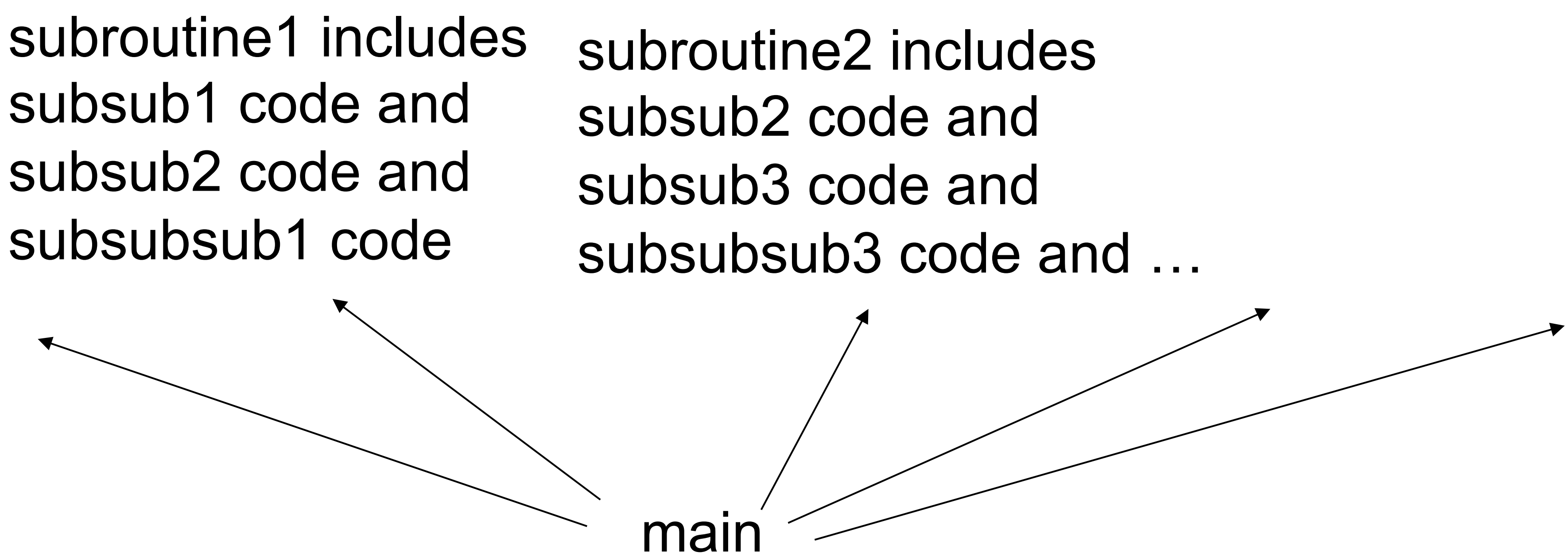
- (Cho et al, EMNLP 2014) Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation
- (Sutskever et al, NIPS 2014) Sequence to sequence learning with neural networks, 3 BLEU points improvement for English-French
- (Devlin et al, ACL 2014) Fast and Robust Neural Network Joint Models for Statistical Machine Translation

Best paper award, 6 BLEU points improvement for Arabic-English

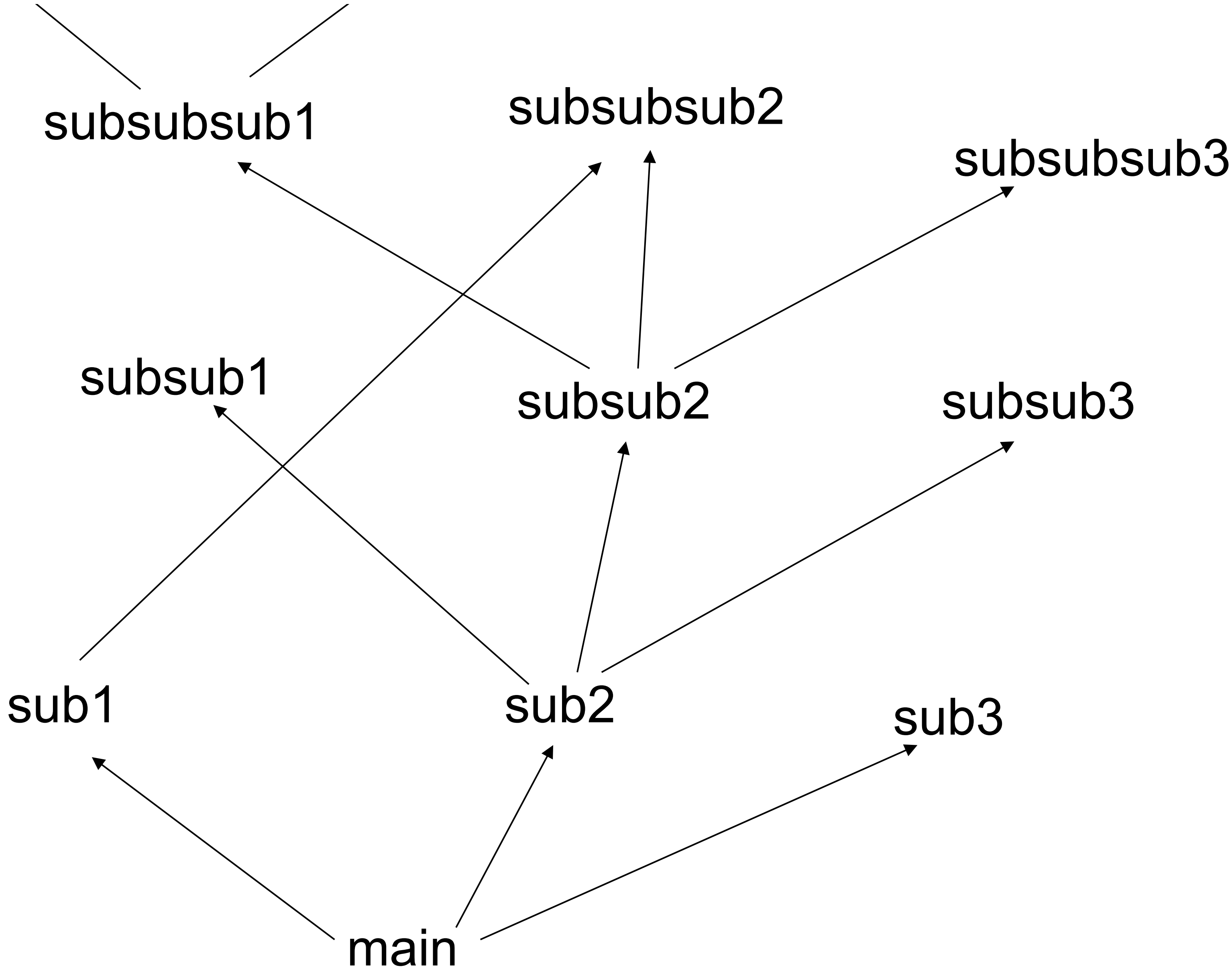








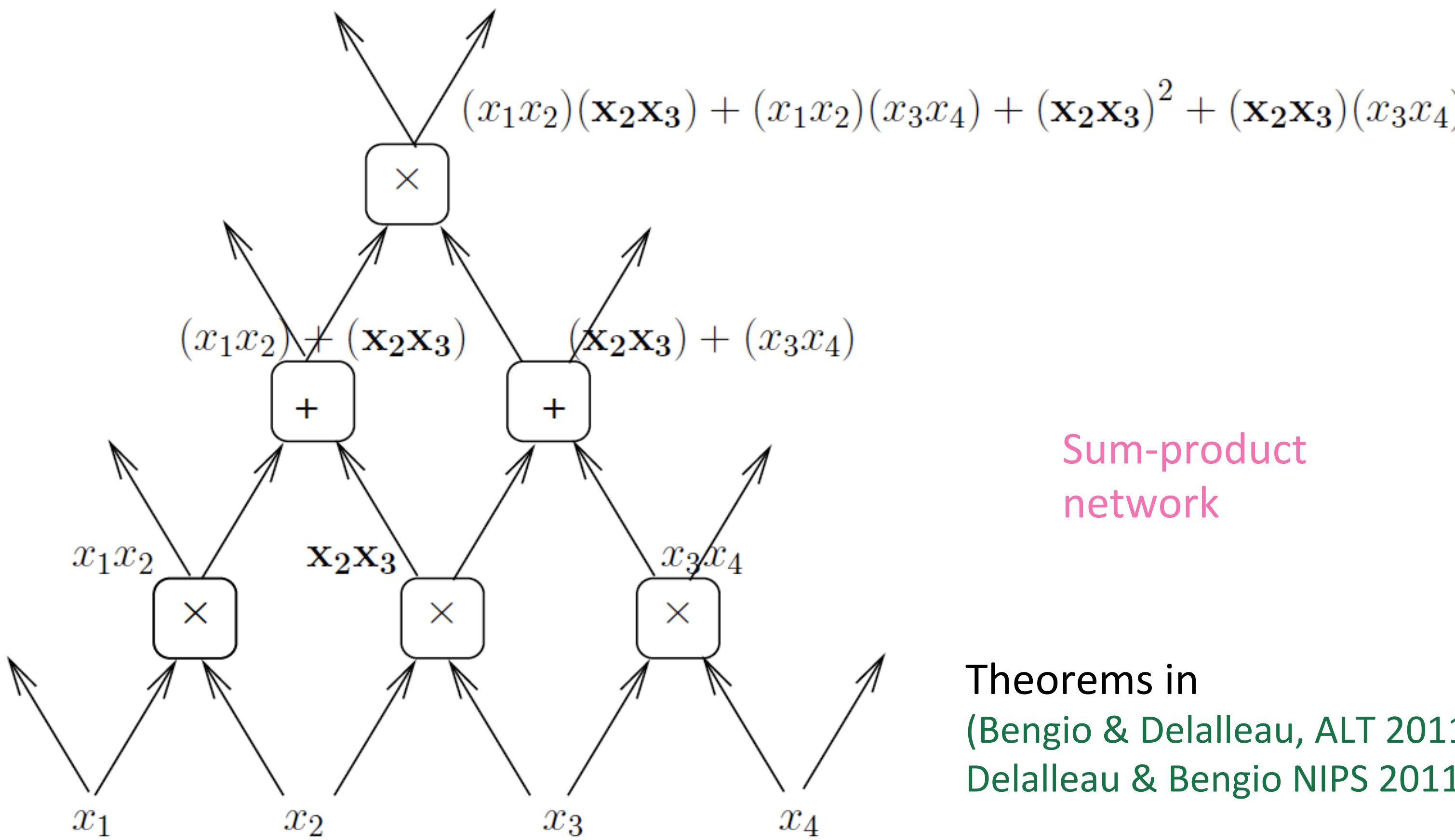
“Shallow” computer program



“Deep” computer program

Sharing Components in a Deep Architecture

Polynomial expressed with shared components: advantage of depth may grow exponentially



Sum-product network

Theorems in
(Bengio & Delalleau, ALT 2011; Delalleau & Bengio NIPS 2011)

Deep Architectures are More Expressive

Theoretical arguments:

2 layers of
- Logic gates
- Formal neurons
- RBF units

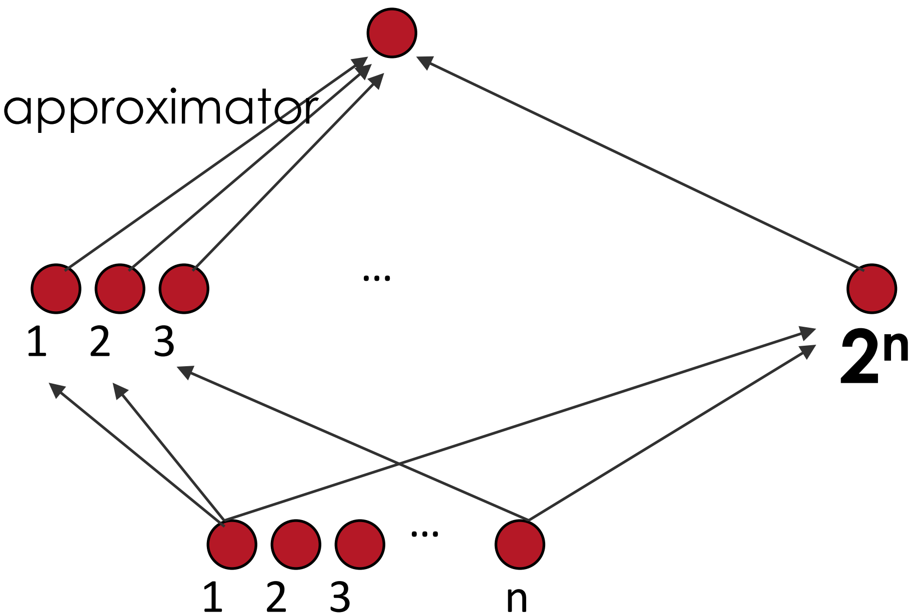
= universal approximator

RBMs & auto-encoders = universal approximator

Theorems on advantage of depth:

(Hastad et al 86 & 91, Bengio et al 2007, Bengio & Delalleau 2011, Braverman 2011, Pascanu et al 2014)

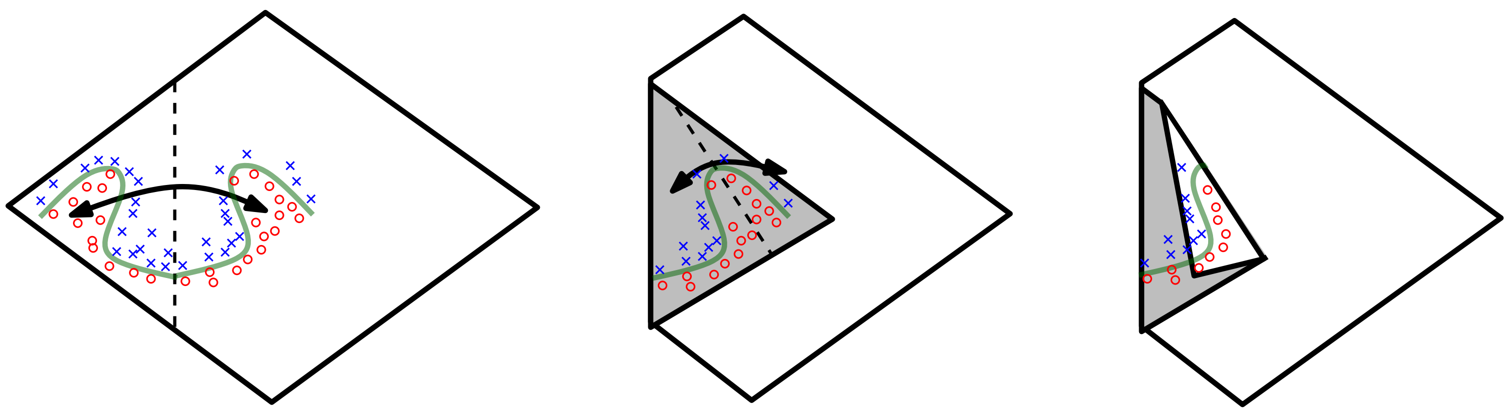
Some functions compactly represented with k layers may require exponential size with 2 layers



New theoretical result: Expressiveness of deep nets with piecewise-linear activation fns

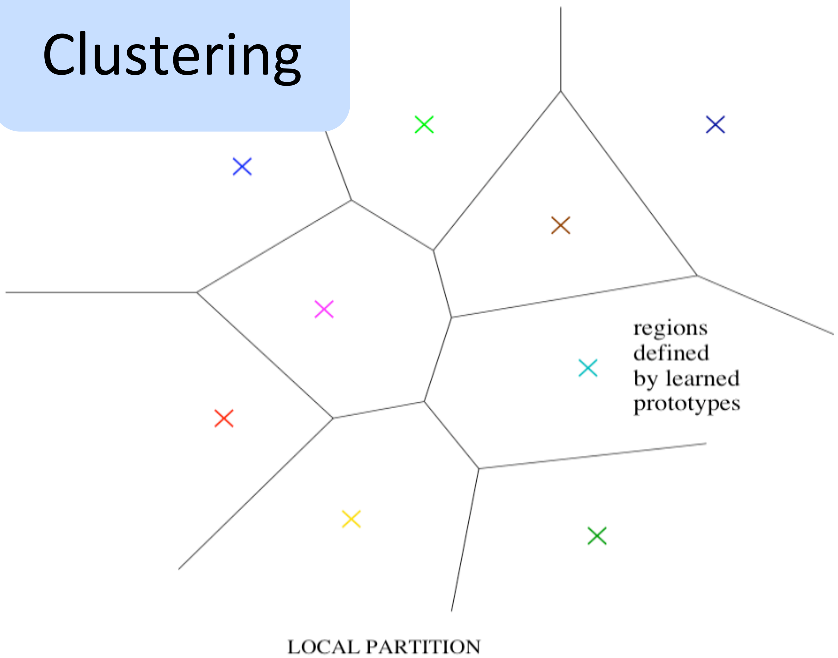
(Pascanu, Montufar, Cho & Bengio; ICLR 2014)

Deeper nets with rectifier/maxout units are exponentially more expressive than shallow ones (1 hidden layer) because they can split the input space in many more (not-independent) linear regions, with constraints, e.g., with abs units, each unit creates mirror responses, folding the input space:



Non-distributed representations

Clustering



- Clustering, Nearest-Neighbors, RBF SVMs, local non-parametric density estimation & prediction, decision trees, etc.

- Parameters for each distinguishable region
- **# of distinguishable regions is linear in # of parameters**

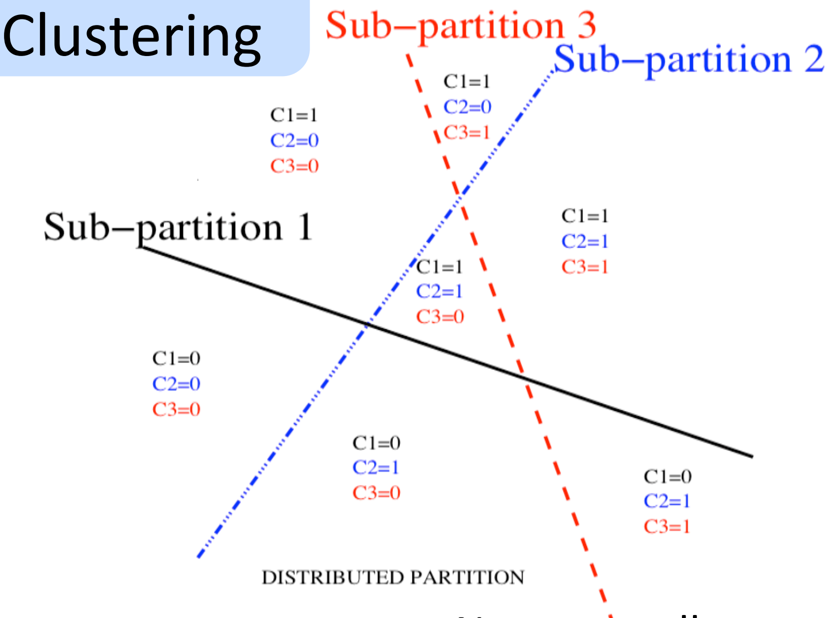
→ No non-trivial generalization to regions without examples

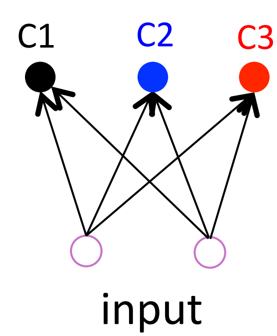
The need for distributed representations

- Factor models, PCA, RBMs, Neural Nets, Sparse Coding, Deep Learning, etc.

- Each parameter influences many regions, not just local neighbors
- **# of distinguishable regions grows almost exponentially with # of parameters**
- **GENERALIZE NON-LOCALLY TO NEVER-SEEN REGIONS**

Multi-Clustering

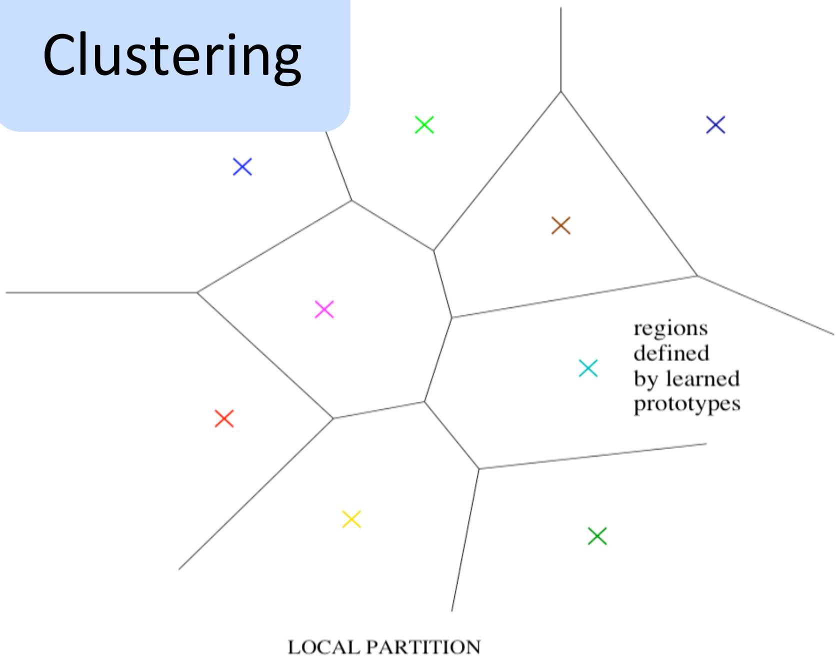




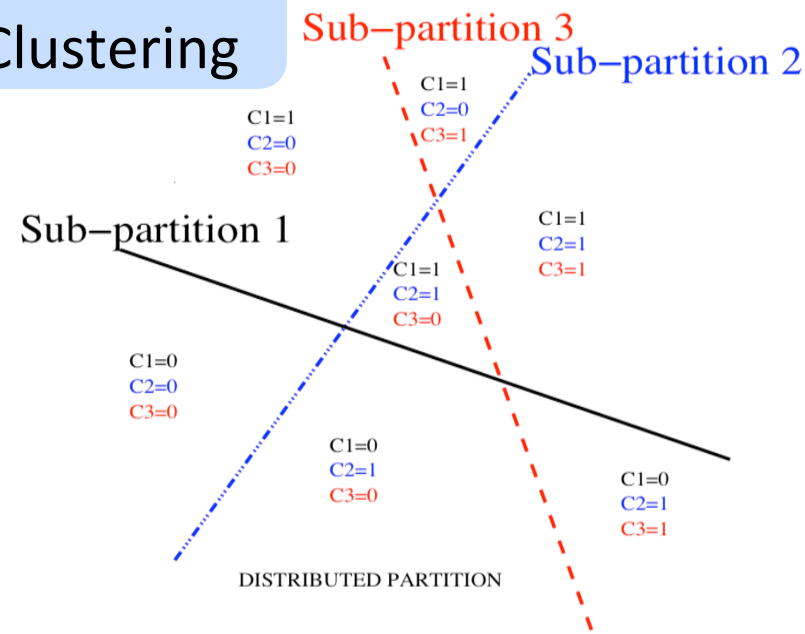
Non-mutually exclusive features/ attributes create a combinatorially large set of distinguiable configurations

The need for distributed representations

Clustering



Multi-Clustering



Learning a **set of features** that are not mutually exclusive can be exponentially more statistically efficient than having nearest-neighbor-like or clustering-like models

Major Breakthrough in 2006

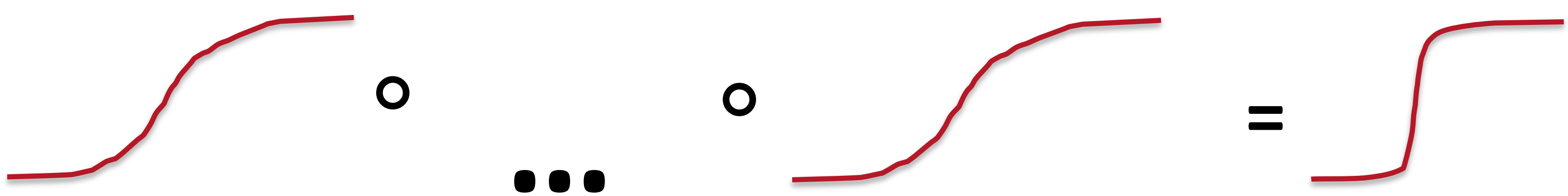
- Ability to train deep architectures by using layer-wise unsupervised learning, whereas previous purely supervised attempts had failed

- Unsupervised feature learners:
    - RBMs
    - Auto-encoder variants
    - Sparse coding variants

(Bengio & LeCun 2007), Scaling Learning Algorithms towards AI

Issues with Back-Prop

- In very deep nets or recurrent nets with many steps, non-linearities compose and yield sharp non-linearity → gradients vanish or explode
- Training deeper nets: harder optimization
- In the extreme of non-linearity: discrete functions, can't use back-prop
- Not biologically plausible?

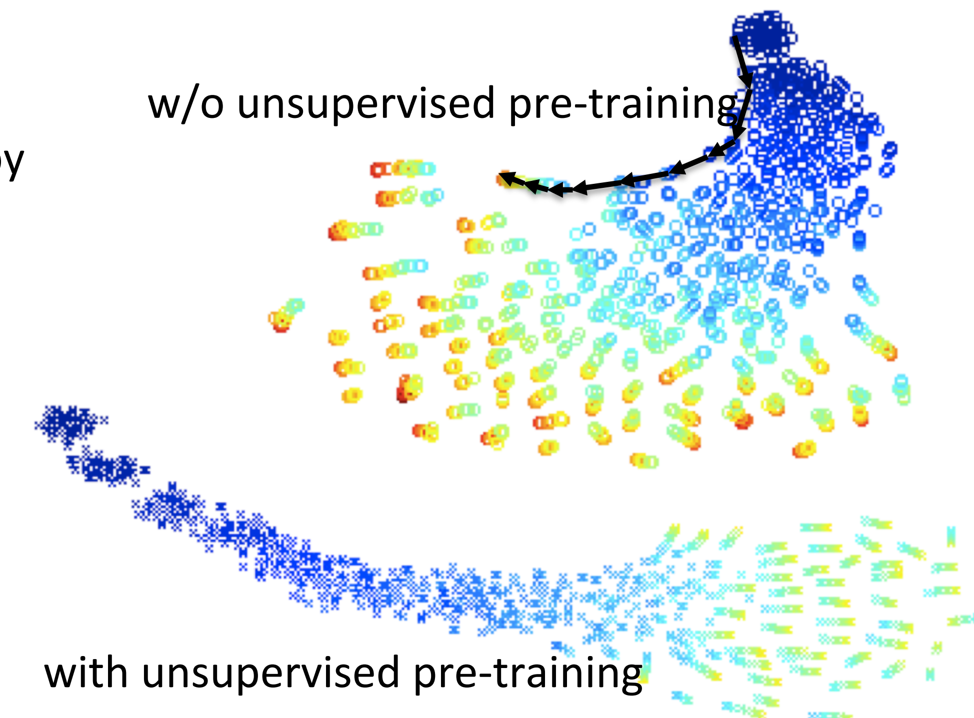


Effect of Initial Conditions in Deep Nets

- *(Erhan et al 2009, JMLR)*
- Supervised deep net with vs w/o unsupervised pre-training → very different minima

Neural net trajectories in function space, visualized by t-SNE

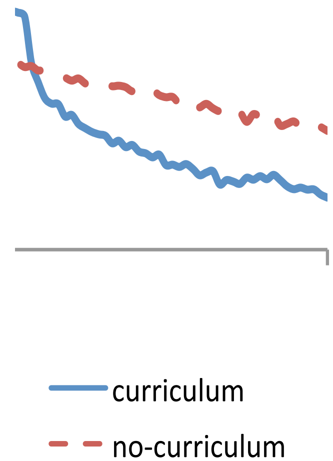
No two training trajectories end up in the same place → huge number of effective local minima



Order & Selection of Examples Matters

(Bengio, Louradour, Collobert & Weston, ICML'2009)

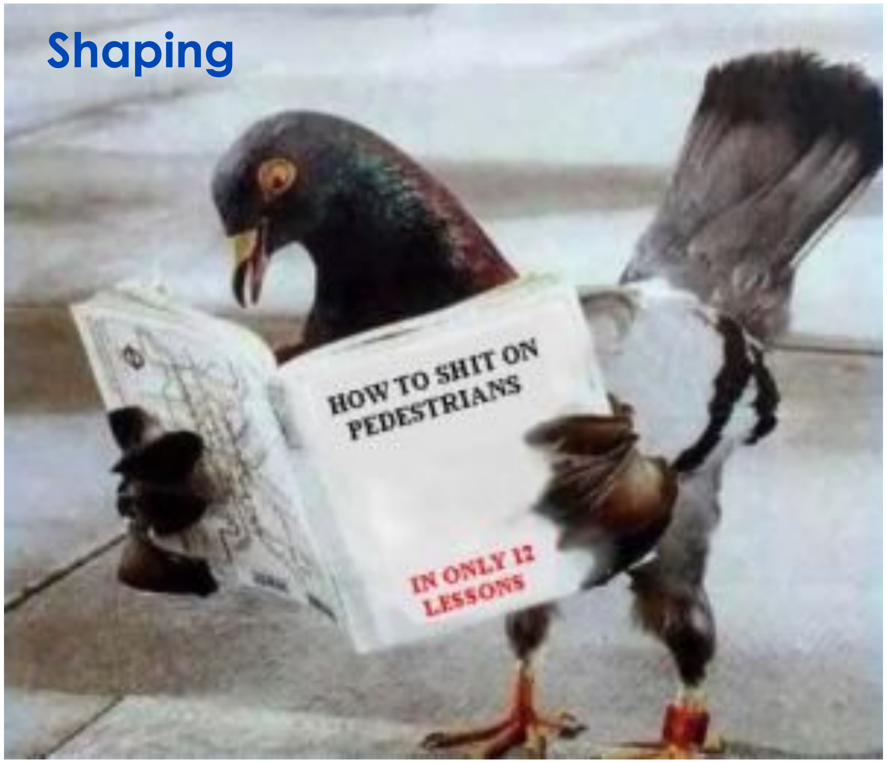
- Curriculum learning
- (Bengio et al 2009, Krueger & Dayan 2009)
- **Start with easier examples**



- Faster convergence to a better local minimum in deep architectures

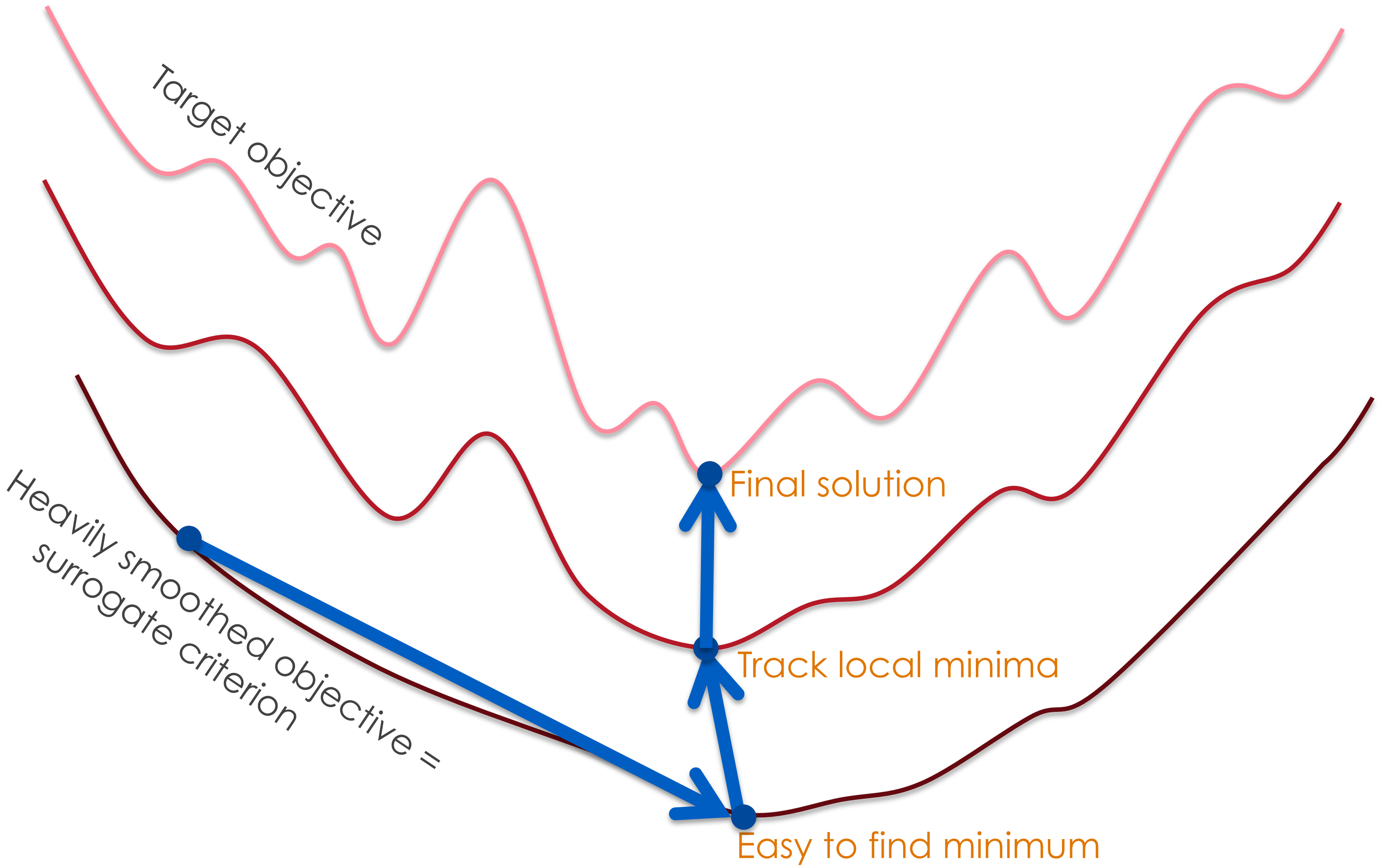
Curriculum Learning

Guided learning helps training humans and animals



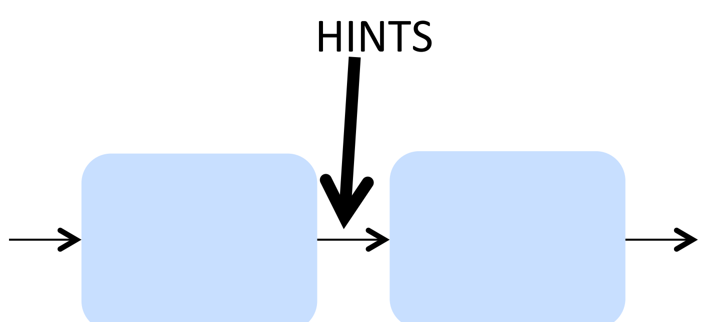
Start from simpler examples / easier tasks (Piaget 1952, Skinner 1958)

Continuation Methods



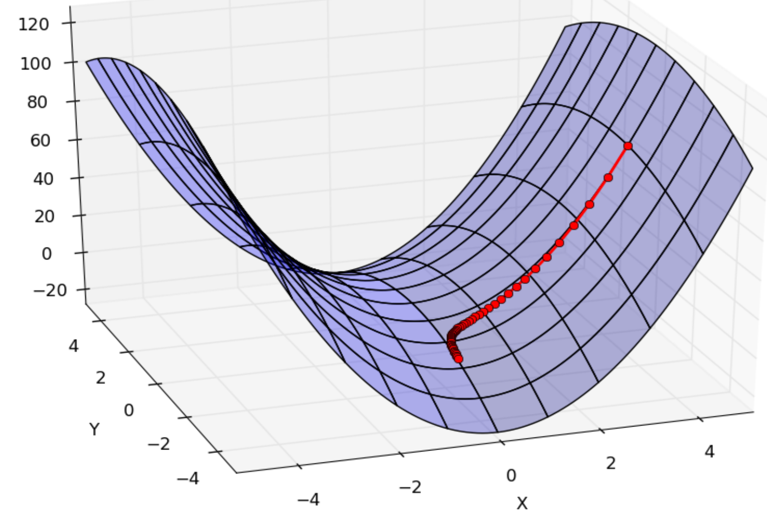
Guided Training, Intermediate Concepts

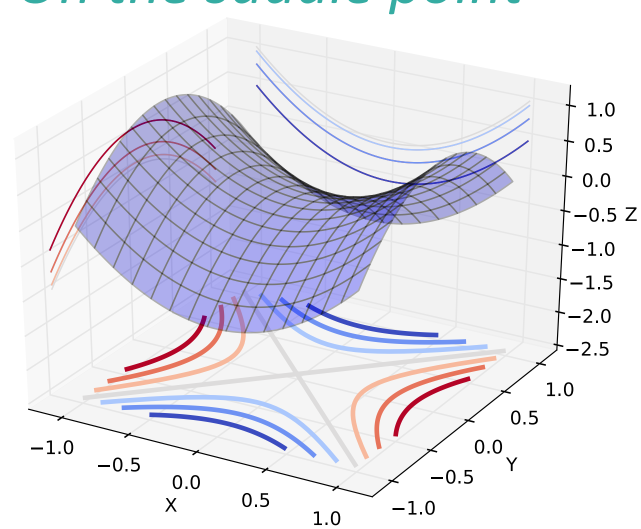
- In (Gulcehre & Bengio ICLR'2013) we set up a task that seems almost impossible to learn by shallow nets, deep nets, SVMs, trees, boosting etc
- Breaking the problem in two sub-problems and pre-training each module separately, then fine-tuning, nails it
- *Need prior knowledge to decompose the task*
- Guided pre-training allows to find much better solutions, escape effective local minima



Saddle Points, not Local Minima

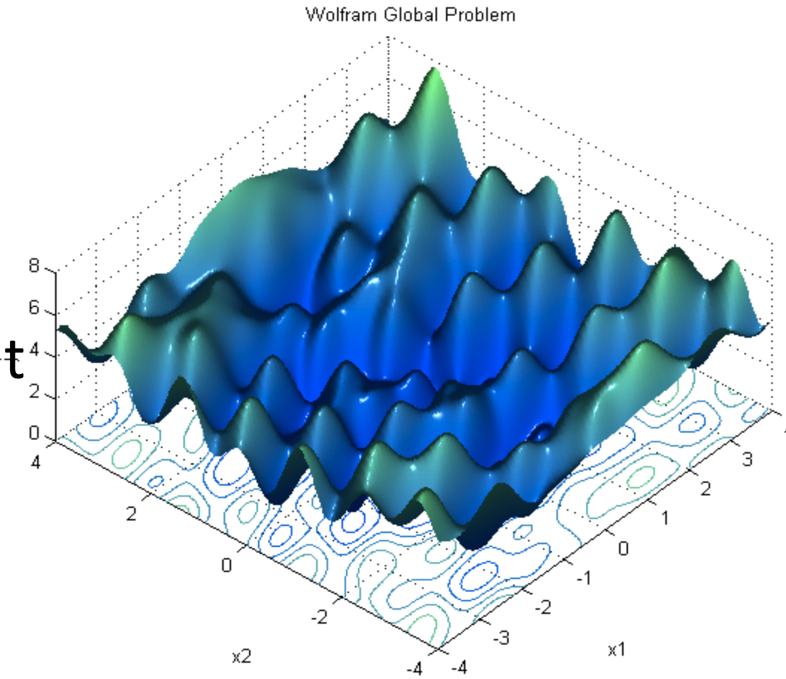
- Traditional thinking is that major obstacle for training deep nets is local minima
- Theoretical and empirical evidence suggest instead that saddle points are exponentially more prevalent critical points, and local minima tend to be of cost near that of global minimum
- (Pascanu, Dauphin, Ganguli, Bengio 2014): *On the saddle point problem for non-convex optimization.*





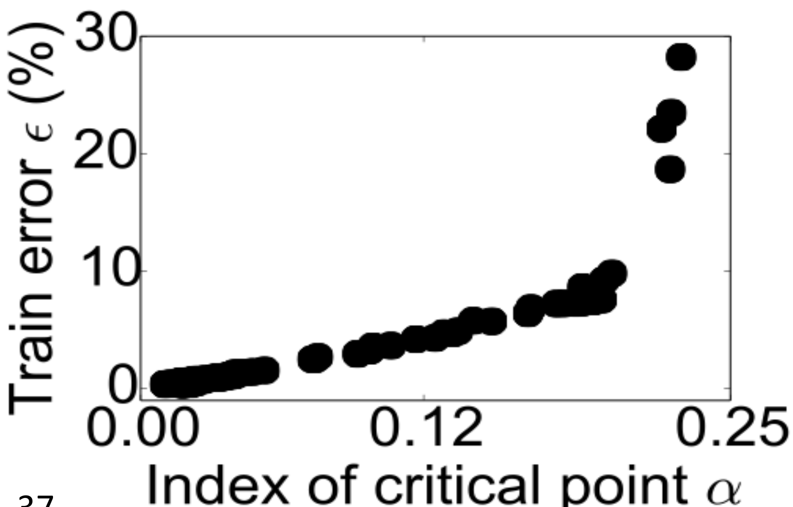
Saddle Points

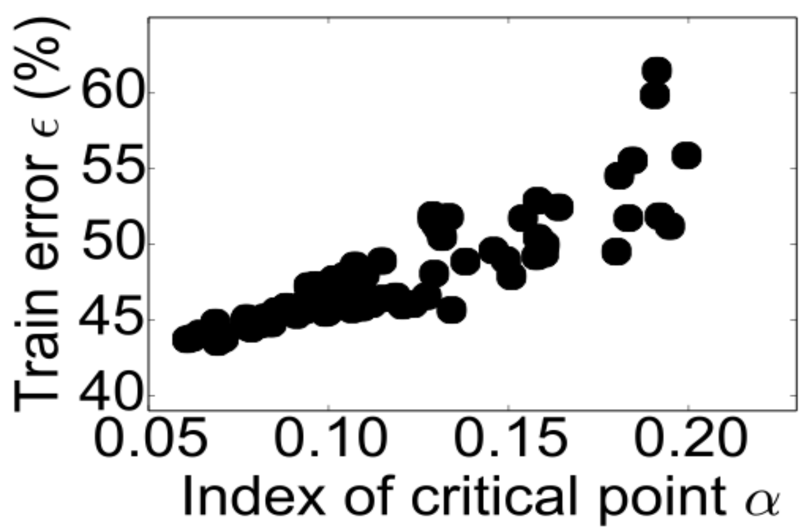
- Local minima dominate in low-D, but saddle points dominate in high-D
- Most local minima are close to the bottom (global minimum error)



MNIST

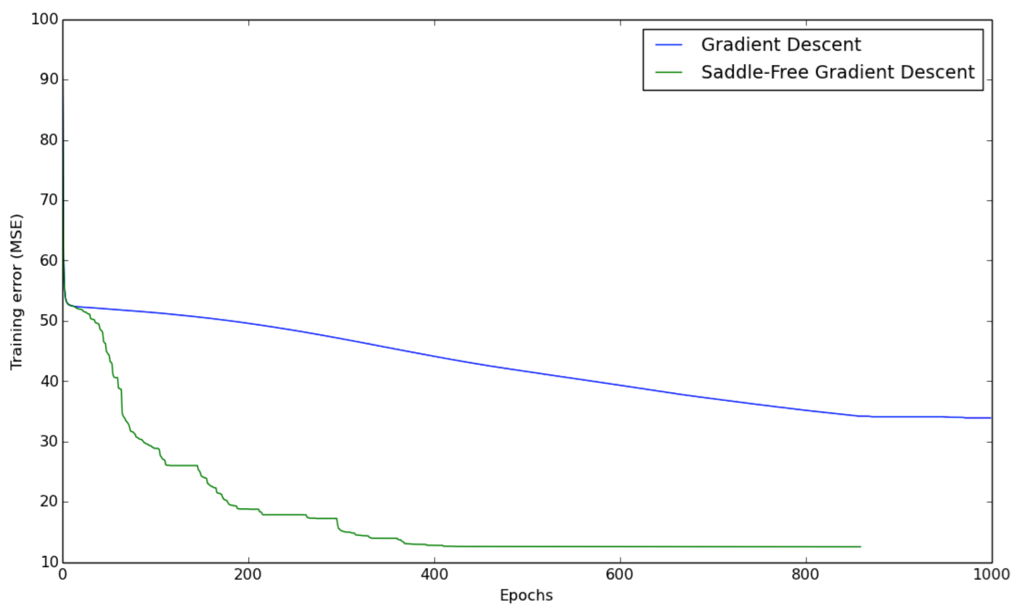
CIFAR





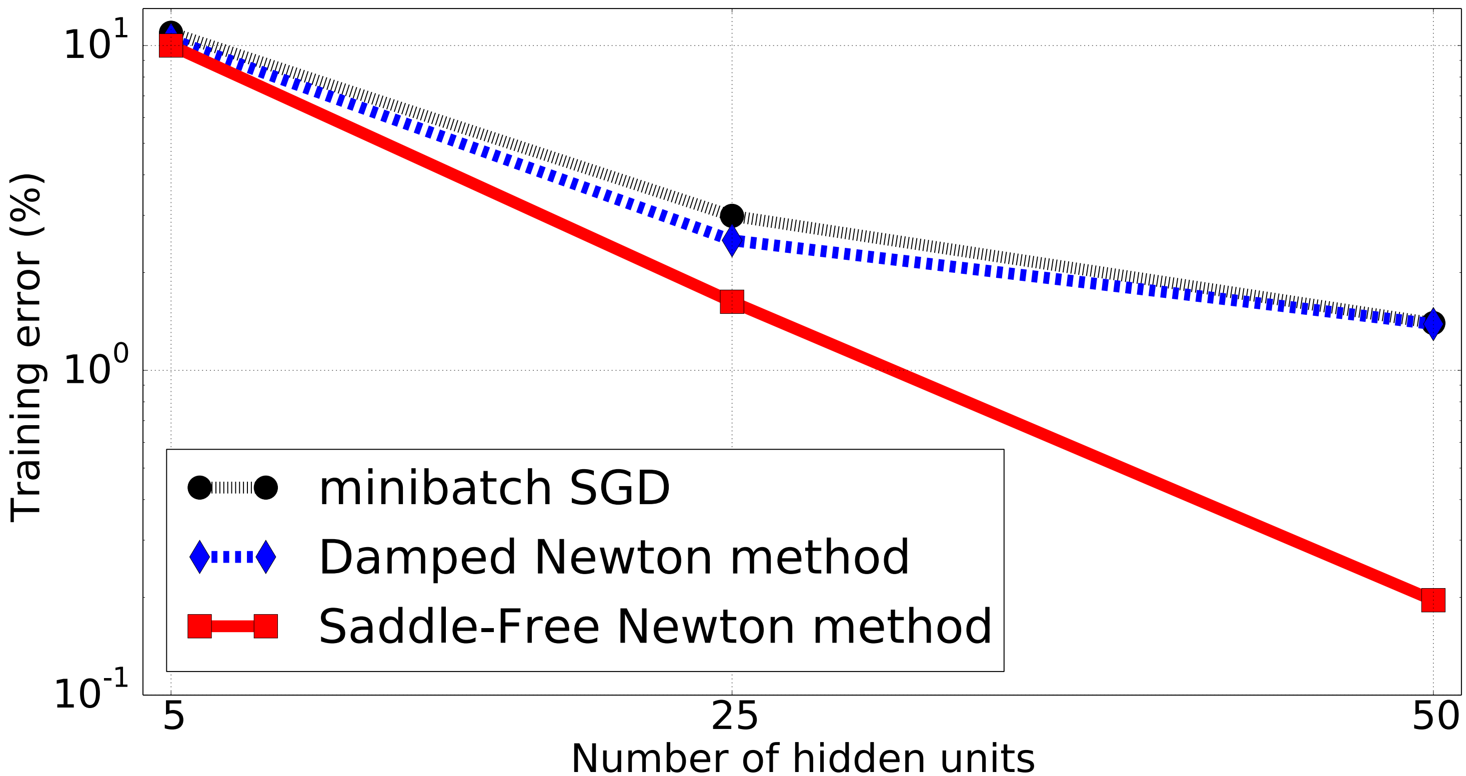
It is possible to escape saddle points!

- NIPS'2014 paper, Dauphin et al.
- More work is ongoing to make it online
- Challenge: track the most negative eigenvector, which is easy in batch mode with power method, if we also track most positive, via $v \leftarrow (H - \lambda I)v$



Saddle-Free Optimization
(Dauphin et al NIPS'2014)

- Replace eigenvalues λ of Hessian by |λ|



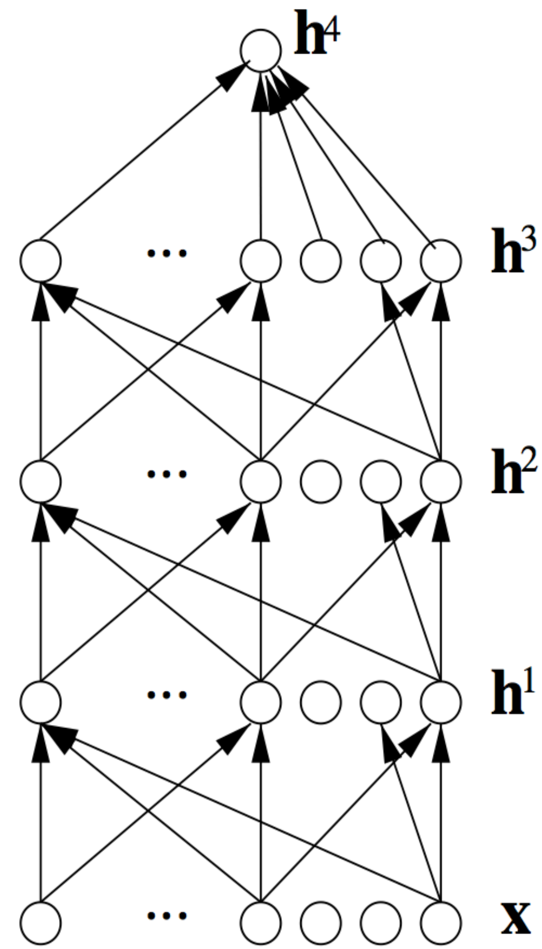
Deep Supervised Neural Nets

- Now can train them even without unsupervised pre-training:

  better initialization and non-linearities (rectifiers, maxout), generalize well with large labeled sets and regularizers (dropout)

- **Unsupervised pre-training:**

  rare classes, transfer, smaller labeled sets, or as extra regularizer.



Why Unsupervised Learning?

- Recent progress mostly in supervised DL
- ∃ real challenges for unsupervised DL
- Potential benefits:
  - Exploit tons of unlabeled data
  - Answer new questions about the variables observed
  - Regularizer – transfer learning – domain adaptation
  - Easier optimization (local training signal)
  - Structured outputs

Invariance and Disentangling

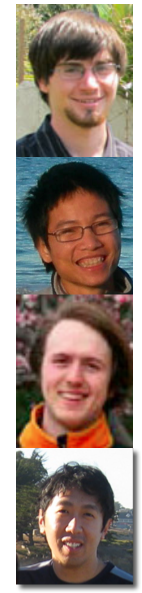
- Invariant features

- Which invariances?
- Alternative: learning to disentangle factors
- Good disentangling →

  avoid the curse of dimensionality

Emergence of Disentangling

- (Goodfellow et al. 2009): sparse auto-encoders trained on images
  - some higher-level features more invariant to geometric factors of variation



- (Glorot et al. 2011): sparse rectified denoising auto-encoders trained on bags of words for sentiment analysis
  - different features specialize on different aspects (domain, sentiment)

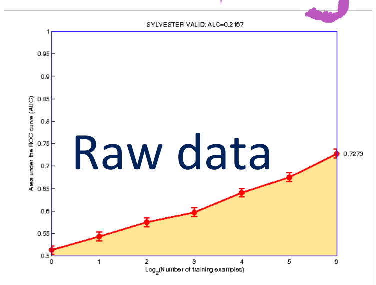
WHY?

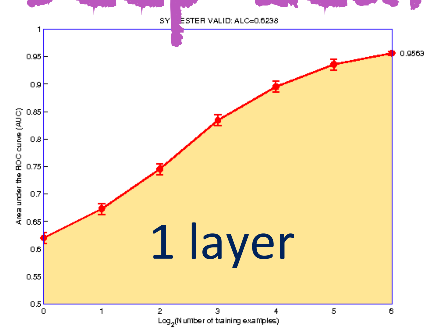
How do humans generalize from very few examples?

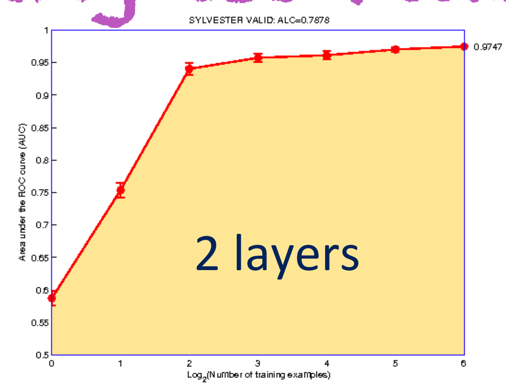
- They **transfer** knowledge from previous learning:
  - Representations
  - Explanatory factors
- Previous learning from: unlabeled data

  + labels for other tasks
- **Prior: shared underlying explanatory factors, in particular between P(x) and P(Y|x)**

Unsupervised and Transfer Learning Challenge + Transfer Learning Challenge: Deep Learning 1st Place

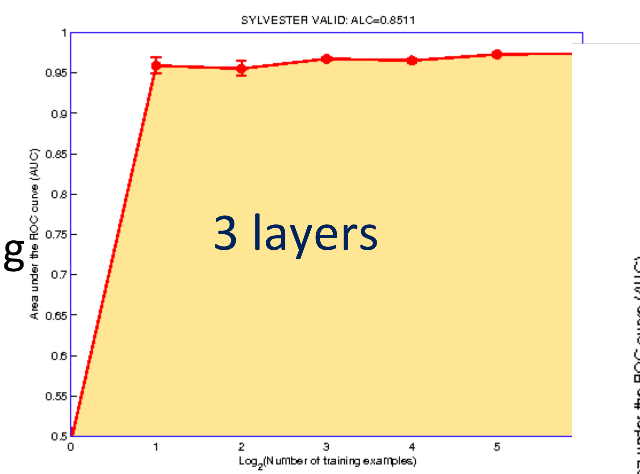


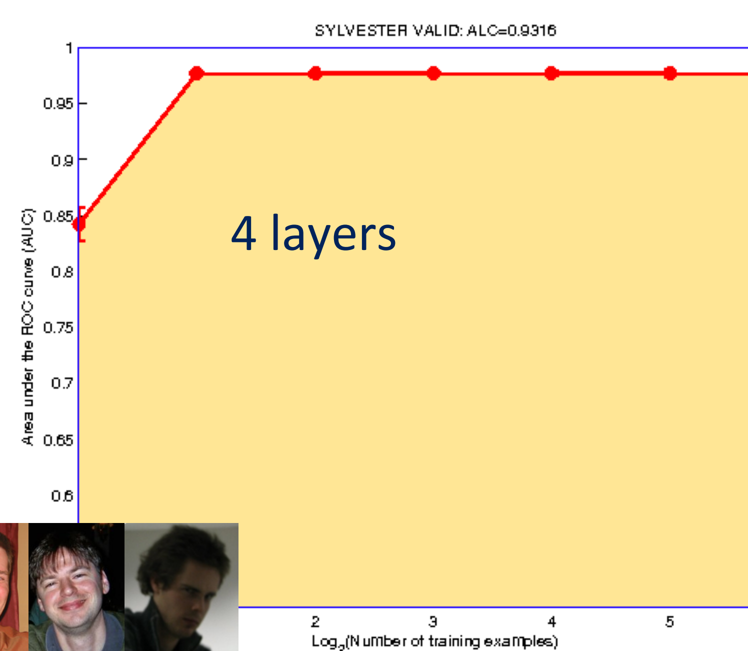


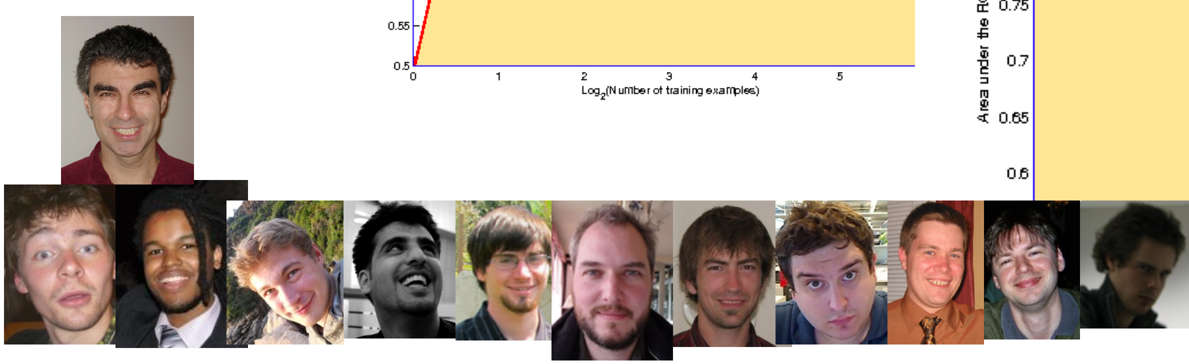


NIPS'2011 Transfer Learning Challenge
Paper: ICML'2012

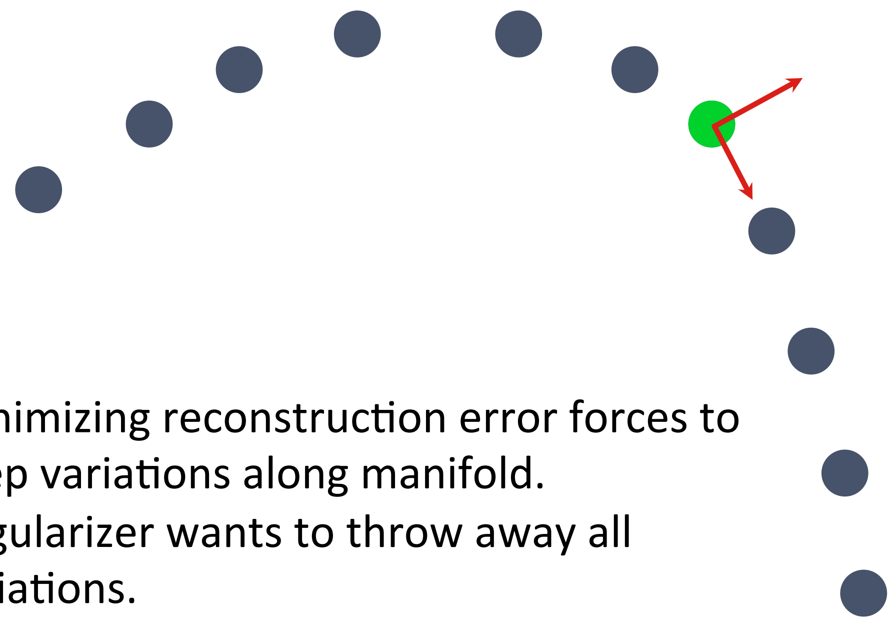
ICML'2011 workshop on Unsup. & Transfer Learning







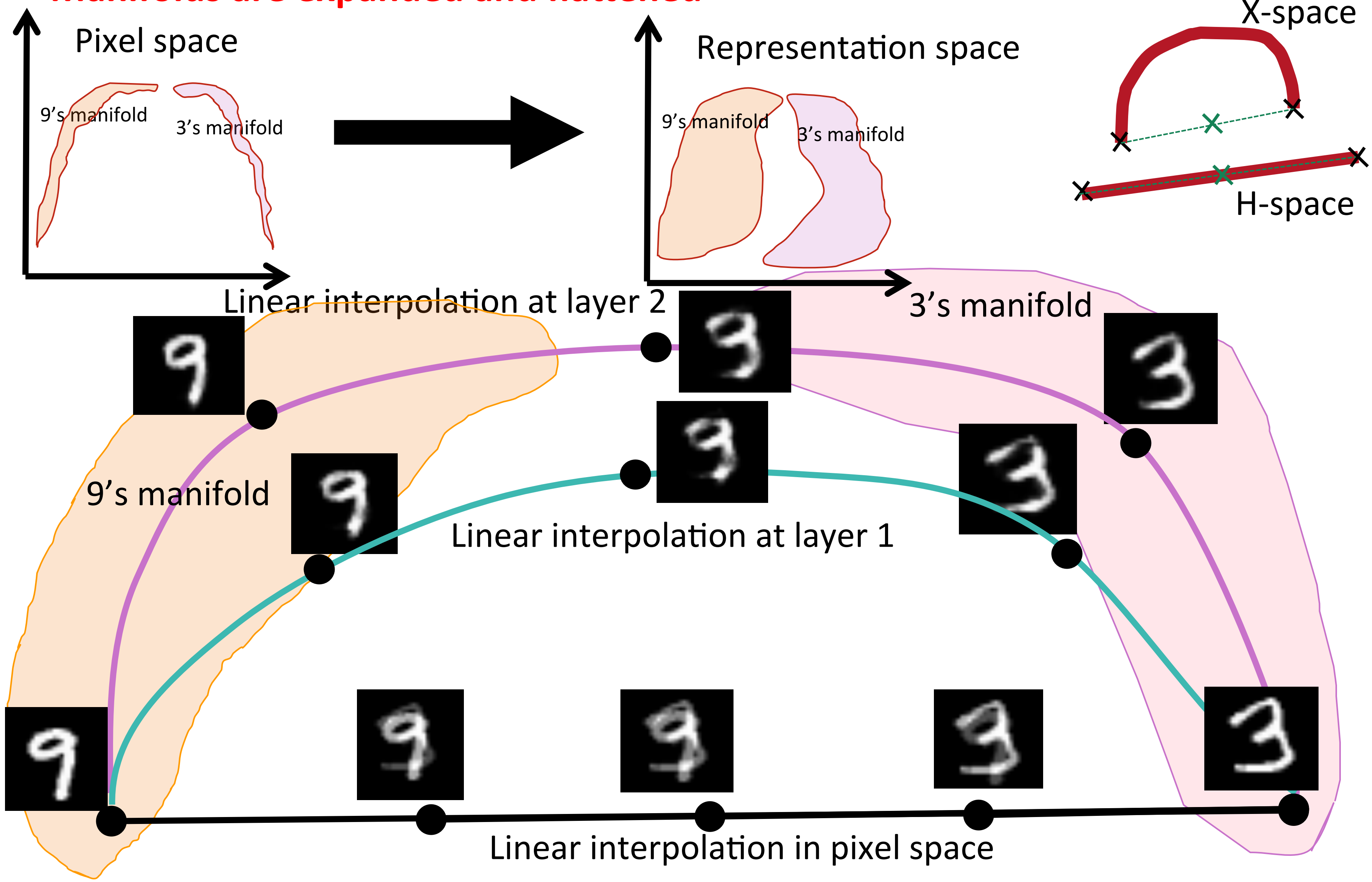
Auto-Encoders Learn Salient Variations, like a non-linear PCA



- Minimizing reconstruction error forces to keep variations along manifold.
- Regularizer wants to throw away all variations.
- With both: keep ONLY sensitivity to variations ON the manifold.

Space-Filling in Representation-Space

- **Deeper representations ➔ abstractions ➔ disentangling**
- **Manifolds are expanded and flattened**



Why Unsupervised Representation Learning? Because of Causality.

- If Ys of interest are among the causal factors of X, then

$$P(Y|X) = \frac{P(X|Y)P(Y)}{P(X)}$$

is tied to P(X) and P(X|Y), and P(X) is defined in terms of P(X|Y), i.e.

- The best possible model of X (unsupervised learning) MUST involve Y as a latent factor, implicitly or explicitly.
- Representation learning SEEKS the latent variables H that explain the variations of X, making it likely to also uncover Y.
- We need 3 pieces:
  - latent variable model P(H),
  - generative decoder P(X|H), and
  - approximate inference encoder Q(H|X).

Challenges with Graphical Models with Latent Variables

- Latent variables help to avoid the curse of dimensionality
- But they come with intractabilities due to sums over an exponentially large number of terms (marginalization):
  - Exact inference (P(h|x)) is typically intractable
  - With undirected models, the normalization constant and its gradient are intractable

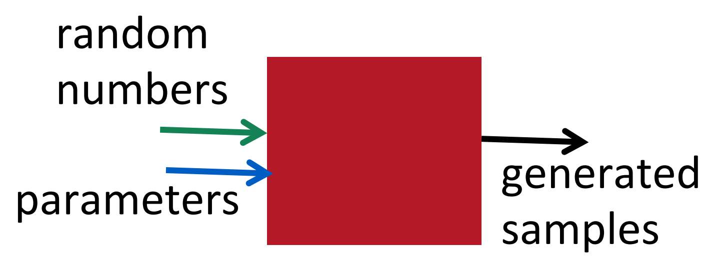
Issues with Boltzmann Machines

- Sampling from the MCMC of the model is required in the inner loop of training
- As the model gets sharper, mixing between well-separated modes stalls

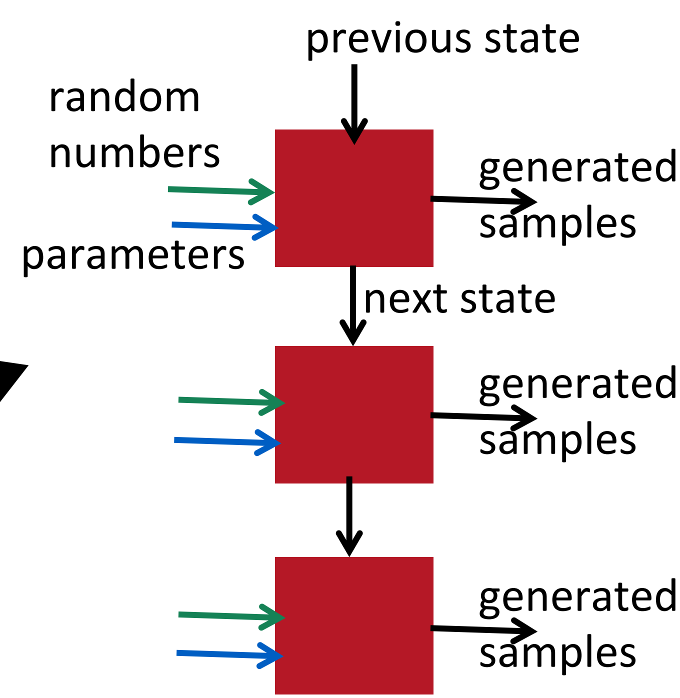
Bypassing Normalization Constants with Generative Black Boxes

- **Instead of parametrizing p(x), parametrize a machine which generates samples**

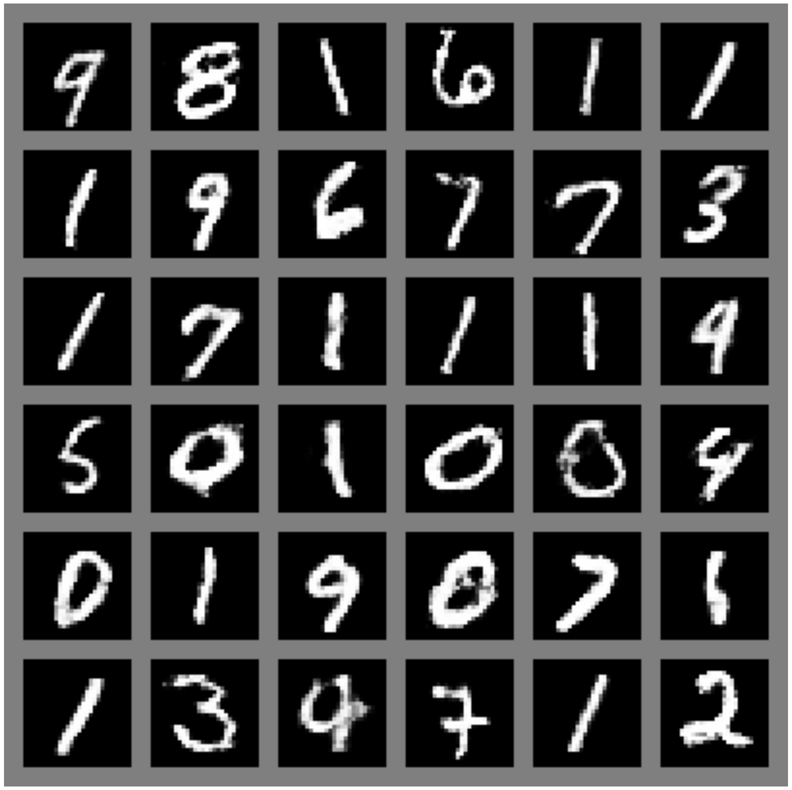
- (Goodfellow et al, NIPS 2014, Generative adversarial nets) for the case of ancestral sampling in a deep generative net. Variational auto-encoders are closely related.

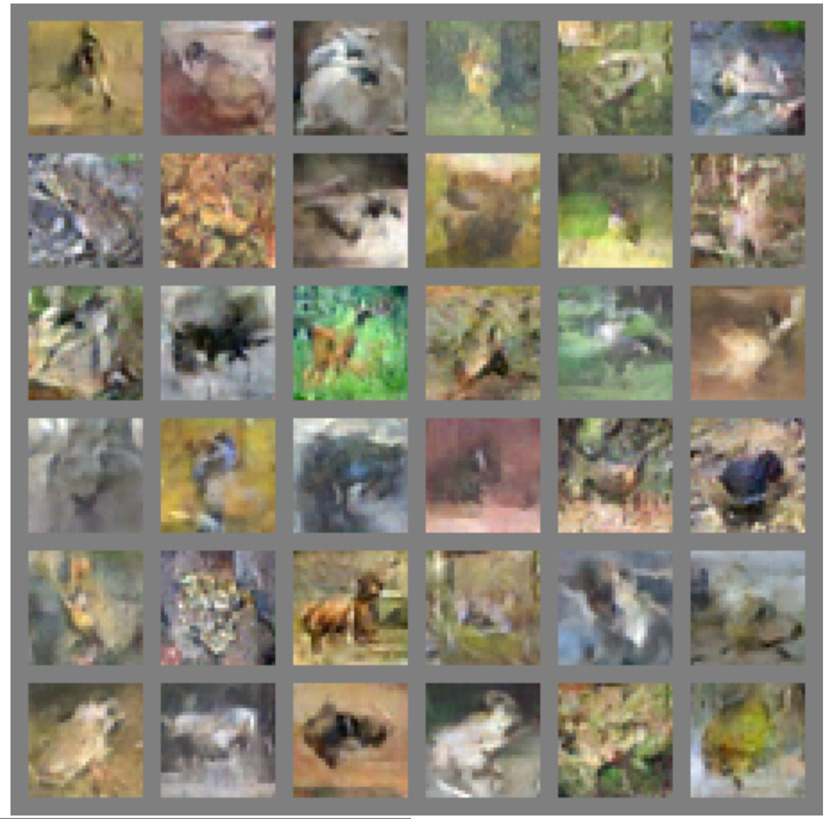


- (Bengio et al, ICML 2014, Generative Stochastic Networks), learning the transition operator of a Markov chain that generates the data.



Adversarial Nets movies

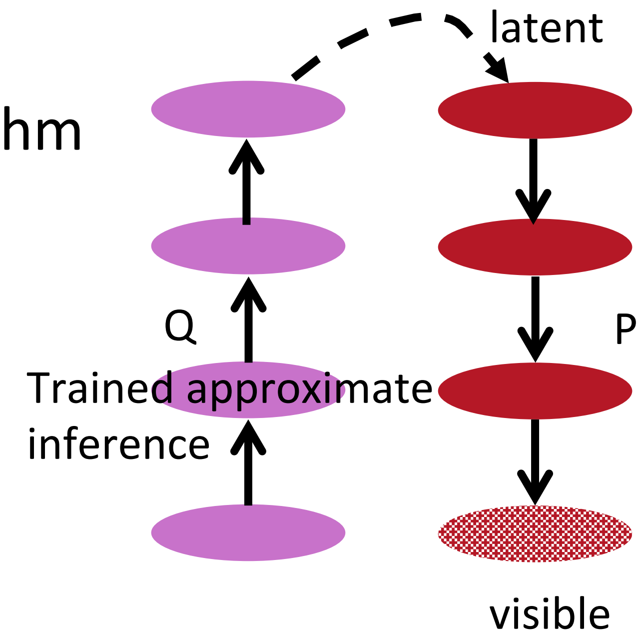




Each movie = linear interpolation between 2 random samples in representation-space

Ancestral Sampling with Learned Approximate Inference

- Helmholtz machine & Wake-Sleep algorithm
  - (Dayan, Hinton, Neal, Zemel 1995)
- Variational Auto-Encoders
  - (Kingma & Welling 2013, ICLR 2014)
  - (Gregor et al ICML 2014)
  - (Rezende et al ICML 2014)
  - (Mnih & Gregor ICML 2014)
- Reweighted Wake-Sleep (Bornschein & Bengio 2014)
- Target Propagation (Bengio 2014)
- Deep Directed Generative Auto-Encoders (Ozair & Bengio 2014)
- NICE (Dinh et al 2014)

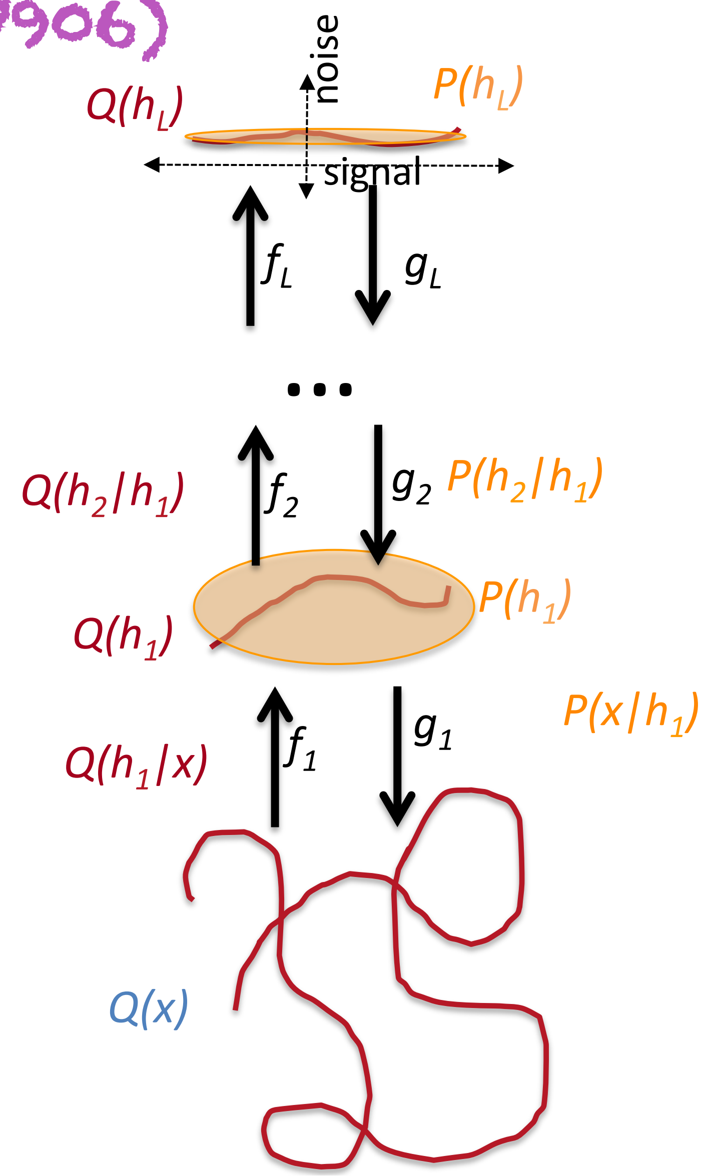


Extracting Structure By Gradual Disentangling and Manifold Unfolding (Bengio 2014, arXiv 1407.7906)

Each level transforms the data into a representation in which it is easier to model, unfolding it more, contracting the noise dimensions and mapping the signal dimensions to a factorized (uniform-like) distribution.

$$\min KL(Q(x,h)||P(x,h))$$

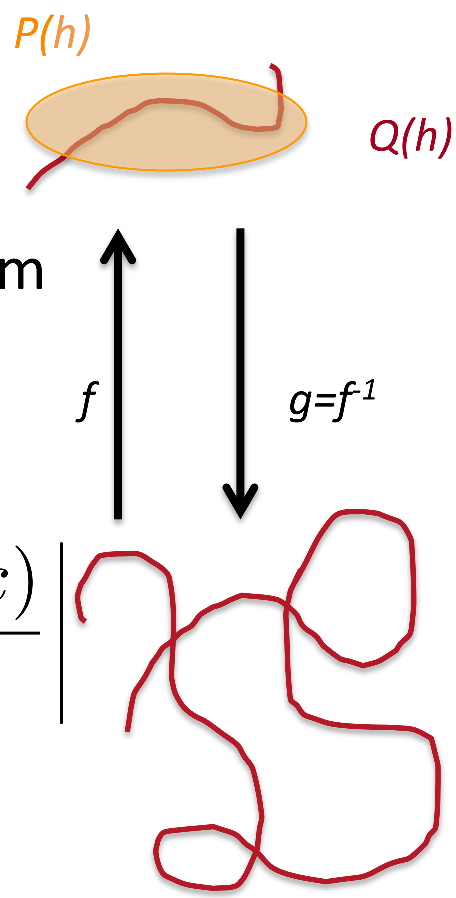
for each intermediate level h



NICE:
Nonlinear Independent Component Estimation

(Dinh, Krueger & Bengio 2014, arxiv 1410.8516)

- Perfect auto-encoder $g=f^{-1}$
- No need for reconstruction error
- Deterministic encoder, no need for entropy term
- But need to correct for density scaling
- **Exact tractable likelihood**

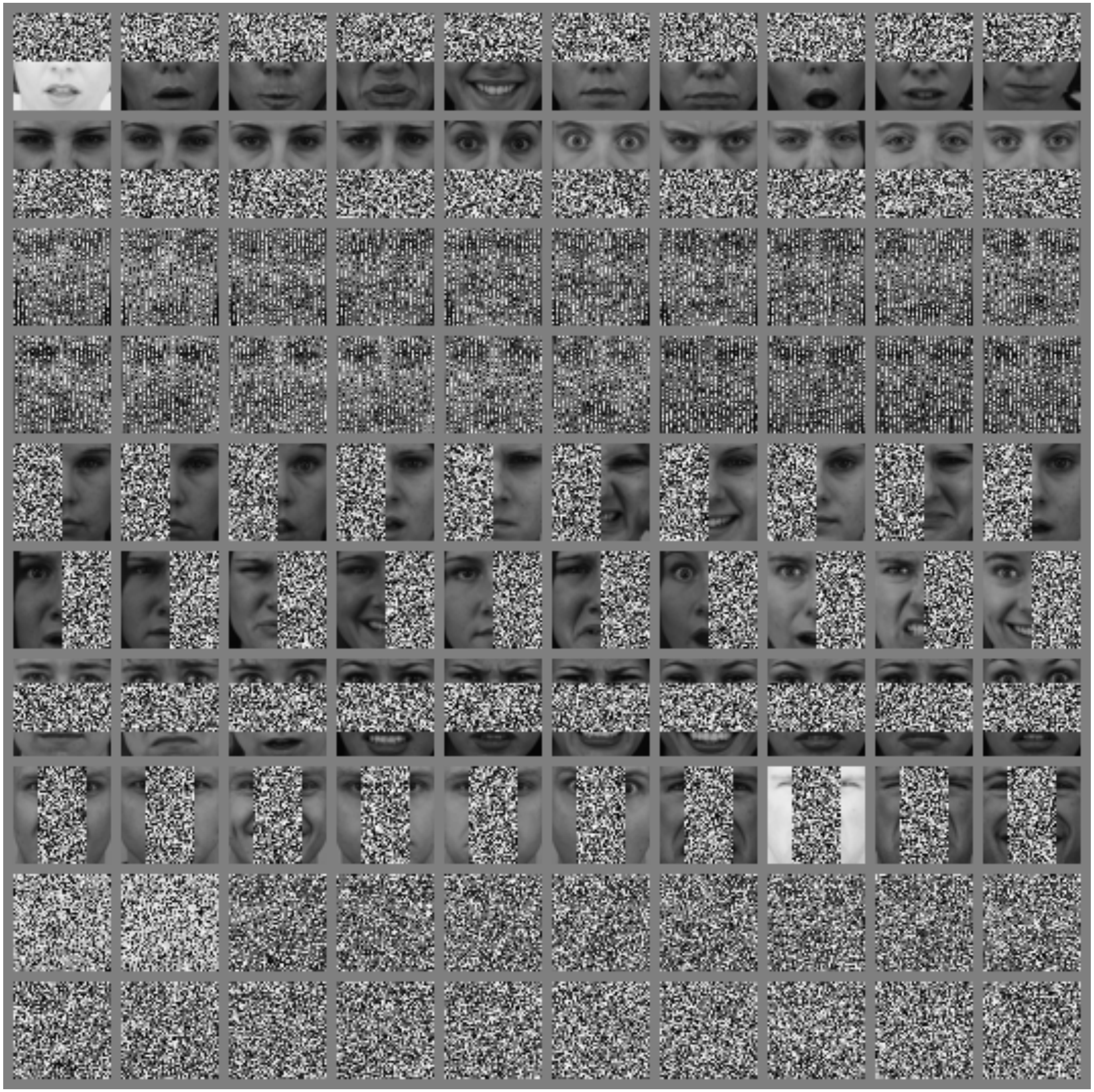


$$\log p_X(x) = \log p_H(f(x)) + \log \left| \det \frac{\partial f(x)}{\partial x} \right|$$

factorized prior

$$P_H(h) = \prod_i P_{H_i}(h_i)$$

NICE Inpaiting Movies (not conv.)



Unfolding AND Disentangling

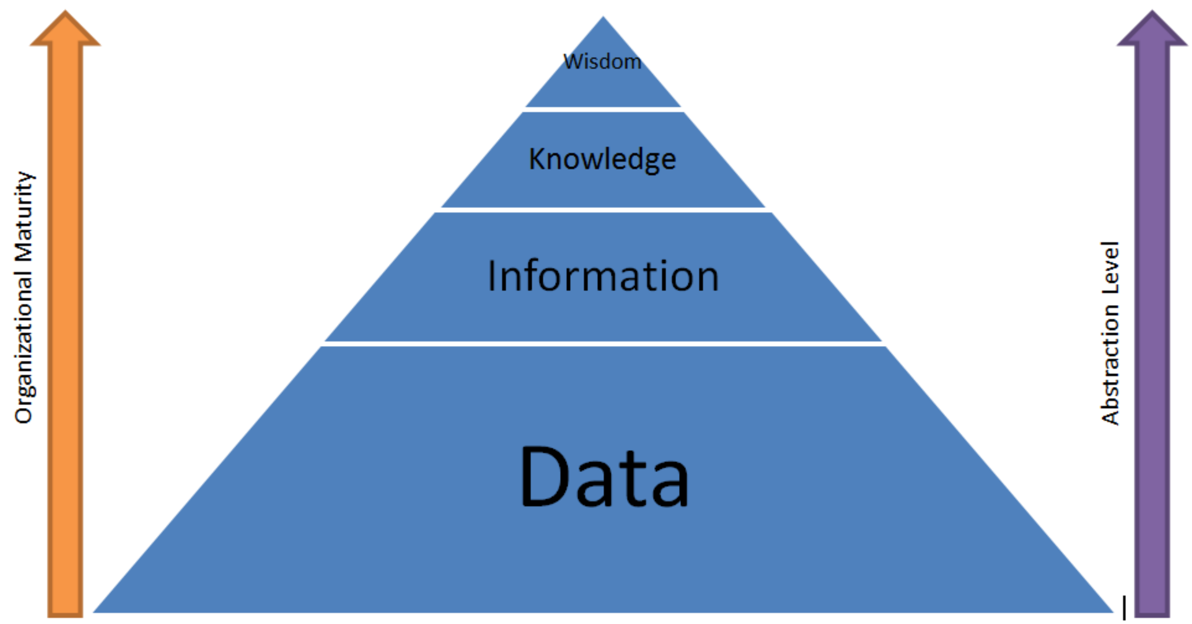
- The previous criteria may allow us to unfold and flatten the data manifold
- What about disentangling the underflying factors of variation?
- Is it enough to assume they are marginally independent?
- They are not conditionally independent…
- There may be intrinsinc ambiguities what makes the disentangling job impossible → need more prior knowledge.

Broad Priors as Hints to Disentangle the Factors of Variation

- *Multiple factors*: distributed representations
- Multiple levels of abstraction: *depth*
- *Semi-supervised* learning: Y is one of the factors explaining X
- *Multi-task* learning: different tasks share some factors
- *Manifold* hypothesis: probability mass concentration
- Natural *clustering*: class = manifold, well-separated manifolds
- Temporal and spatial *coherence*
- *Sparsity*: most factors irrelevant for particular X
- *Simplicity* of factor dependencies (in the right representation)

Learning Multiple Levels of Abstraction

- The big payoff of deep learning is to allow learning higher levels of abstraction
- Higher-level abstractions disentangle the factors of variation, which allows much easier generalization and transfer



Conclusions

- Deep Learning has become a crucial machine learning tool:
  - Int. Conf. on Learning Representation 2013 & 2014 a huge success!

  Conference & workshop tracks, open to new ideas ☺
- Industrial applications (Google, IBM, Microsoft, Baidu, Facebook, Samsung, Yahoo, Intel, Apple, Nuance, BBN, …)
- Potential for more breakthroughs and approaching the "understanding" part of AI by
  - Scaling computation
  - Numerical optimization (better training much deeper nets, RNNs)
  - Bypass intractable marginalizations and exploit broad priors and layer-wise training signals to learn more disentangled abstractions for unsupervised & structured output learning

LISA team: **Merci! Questions?**