Deep Learning Theory

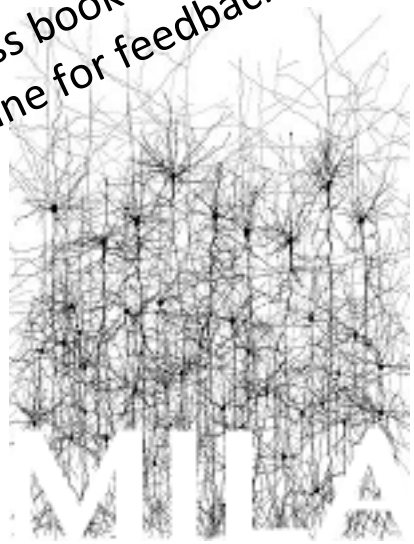
Yoshua Bengio

April 15, 2015

London & Paris ML Meetup

Université de Montréal

PLUG: **Deep Learning**, MIT Press book in preparation, draft chapters online for feedback



Breakthrough

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

Amazing improvements in error rate in object recognition, object detection, speech recognition, and more recently, some in machine translation

Ongoing Progress: Natural Language Understanding

- Recurrent nets generating credible sentences, even better if conditionally:
  - Machine translation
  - Image 2 text

Xu et al, to appear ICML'2015

A woman is throwing a frisbee in a park.

A dog is standing on a hardwood floor.

A stop sign is on a road with a mountain in the background.

A little girl sitting on a bed with a teddy bear.

A group of people sitting on a boat in the water.

A giraffe standing in a forest with trees in the background.

Why is Deep Learning Working so Well?

Machine Learning, AI & No Free Lunch

- Three key ingredients for ML towards AI

  1. Lots & lots of data

  2. Very flexible models

  3. Powerful priors that can defeat the curse of dimensionality

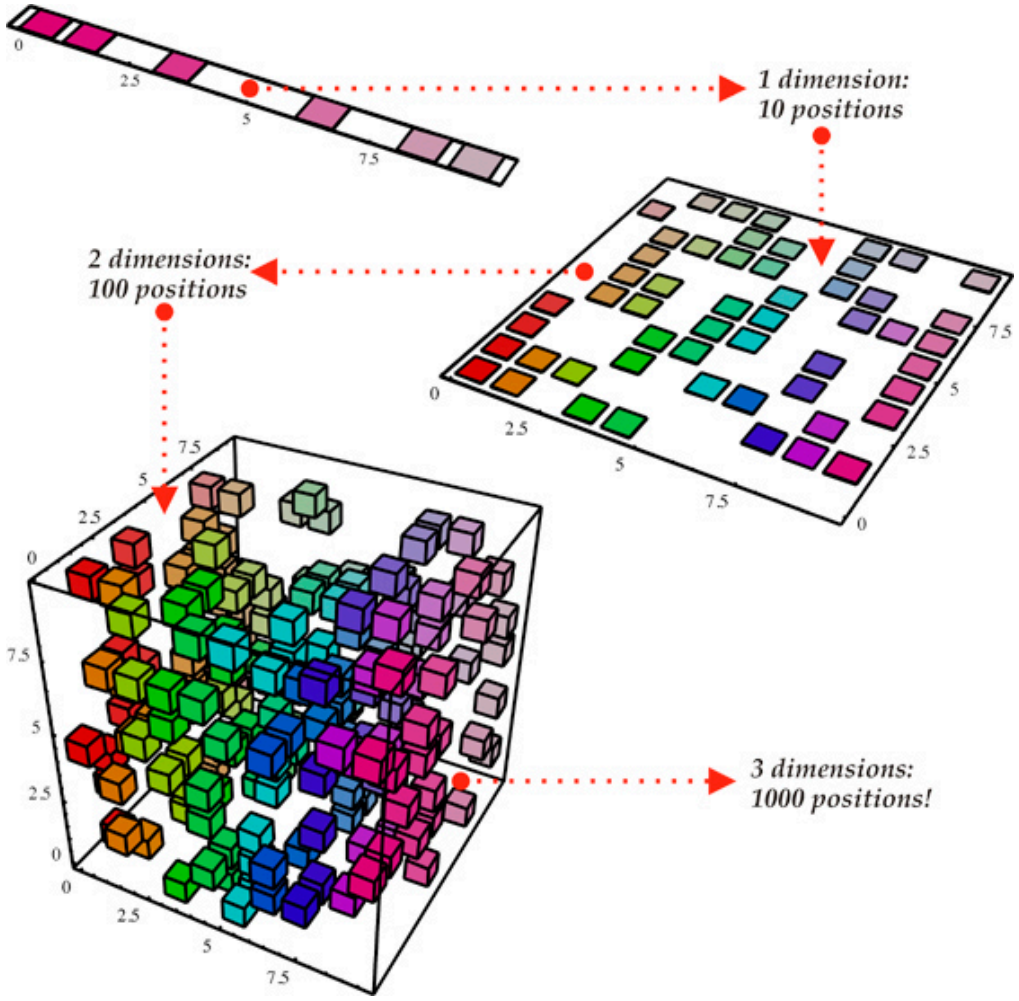
Ultimate Goals

- **AI**
- Needs **knowledge**
- Needs **learning**
  (involves priors + ***optimization****/search*)
- 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)

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

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

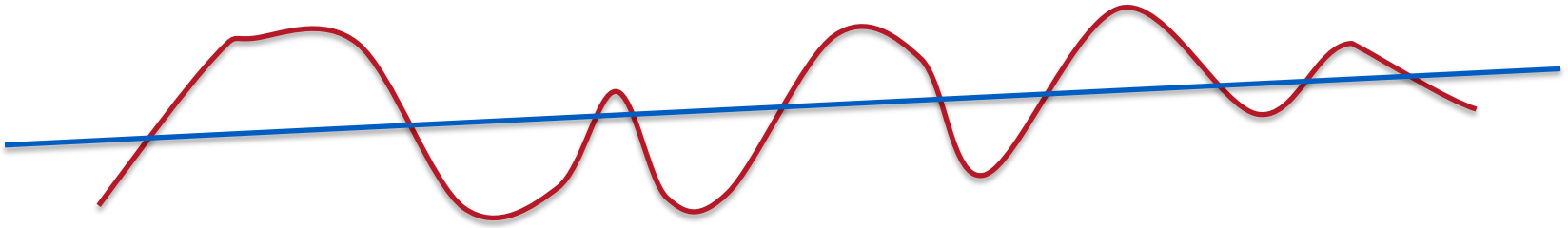
Classical solution: hope for a smooth enough target function, or make it smooth by handcrafting good features / kernel



Not Dimensionality so much as Number of Variations

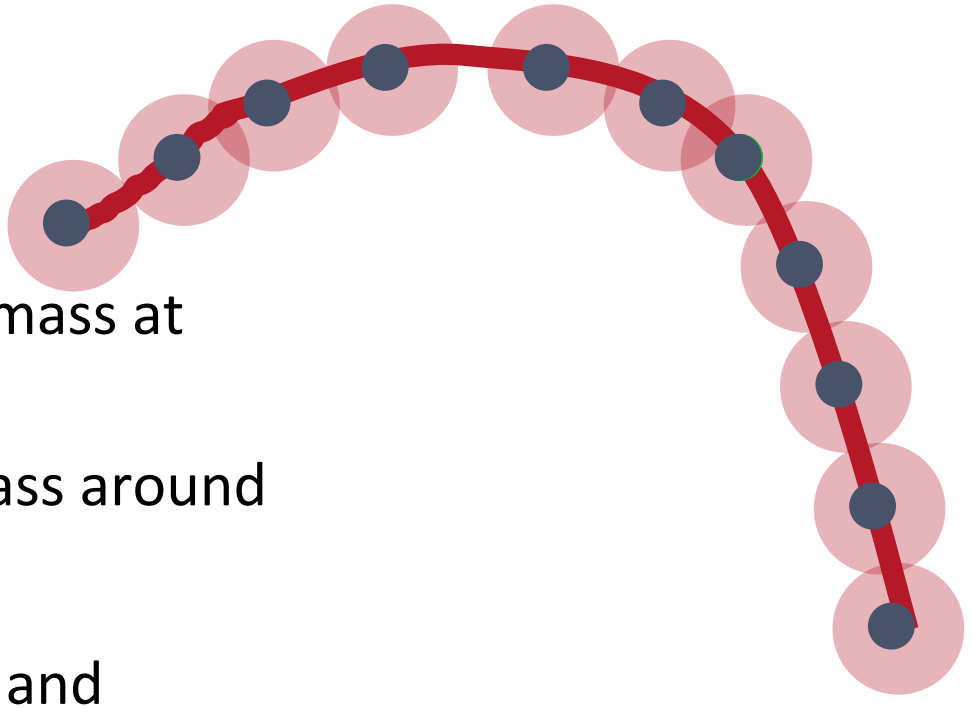
(Bengio, Dellalleau & Le Roux 2007)

- 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

Putting Probability Mass where Structure is Plausible



- Empirical distribution: mass at training examples
- Smoothness: spread mass around
- Insufficient
- Guess some 'structure' and generalize accordingly

Bypassing the curse of dimensionality

We need to build compositionality into our ML models

Just as human languages exploit compositionality to give representations and meanings to complex ideas

Exploiting compositionality gives an exponential gain in representational power

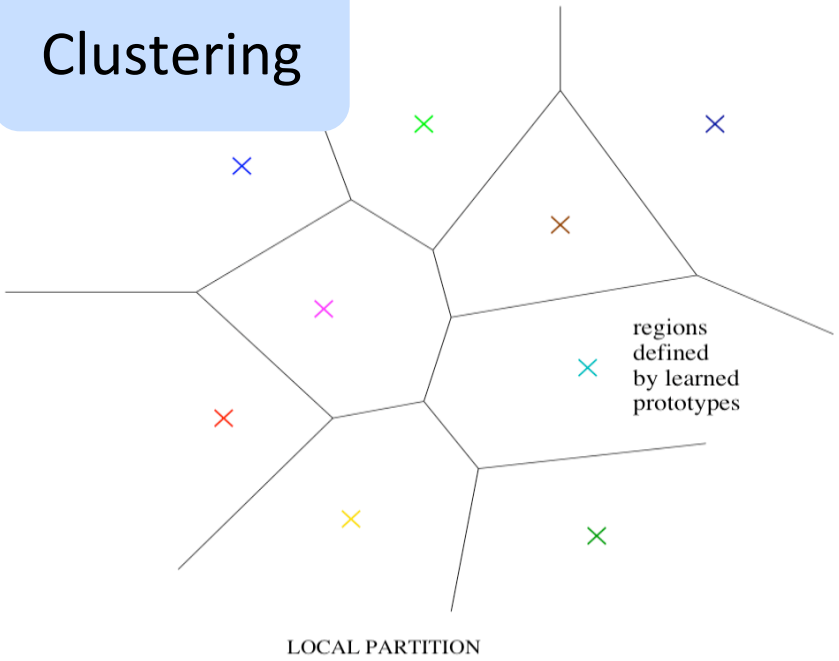
Distributed representations / embeddings: feature learning

Deep architecture: multiple levels of feature learning

Prior: compositionality is useful to describe the world around us efficiently

Non-distributed representations

Clustering



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

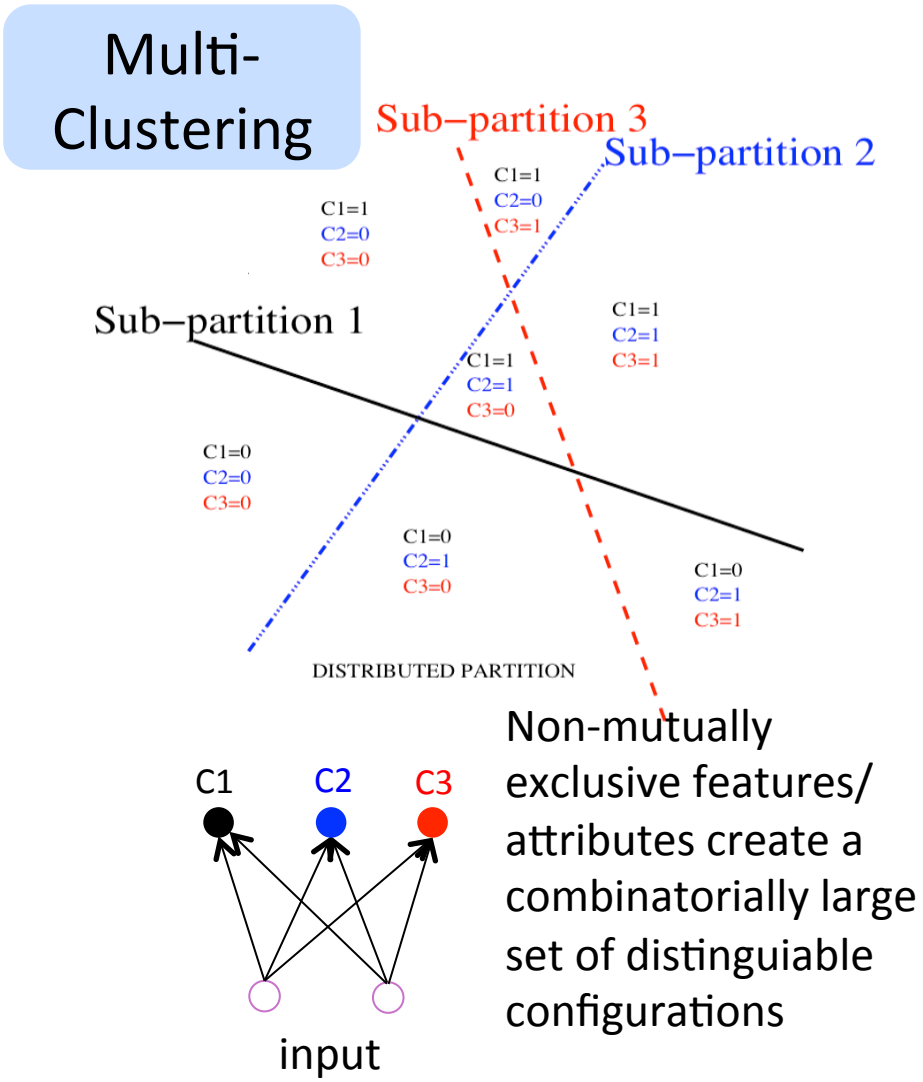
- Parameters for each distinguishable region
- **# of distinguishable regions is linear in # of parameters**

→ 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**



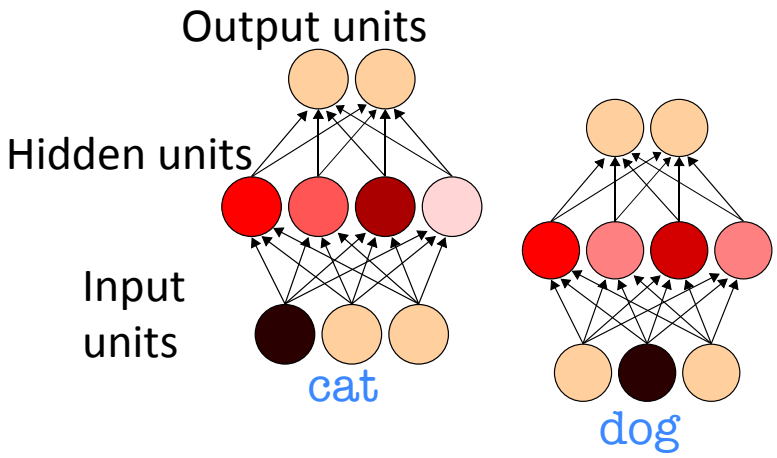
Classical Symbolic AI vs Representation Learning

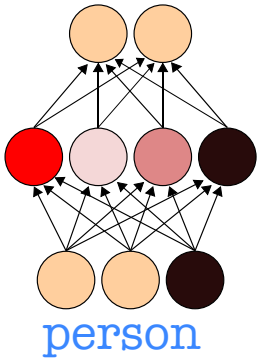
- Two symbols are equally far from each other
- Concepts are not represented by symbols in our brain, but by patterns of activation

  (Connectionism, 1980's)


Geoffrey Hinton

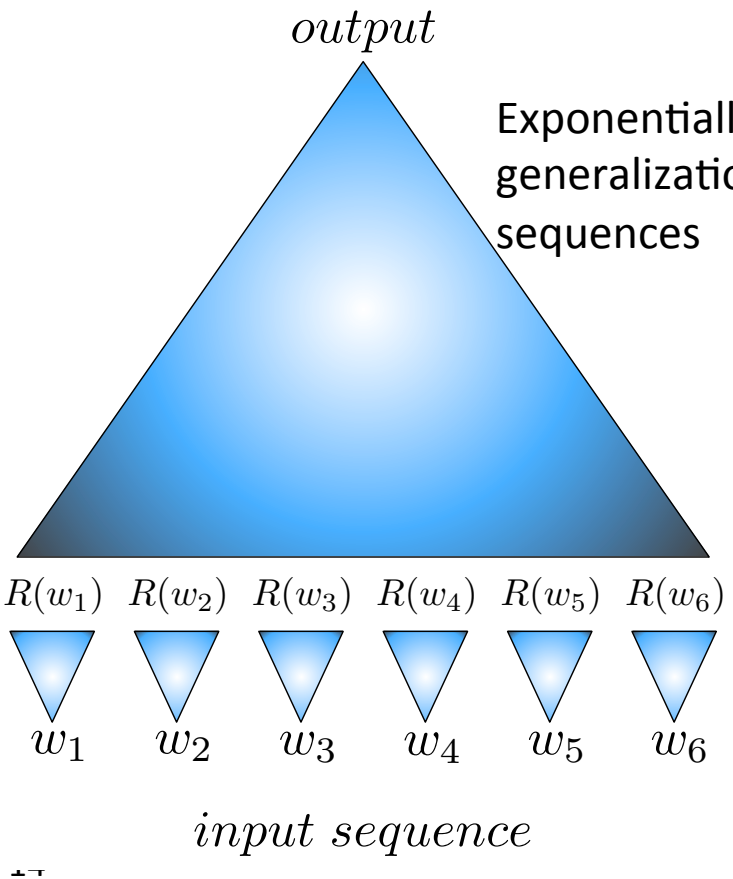

David Rumelhart



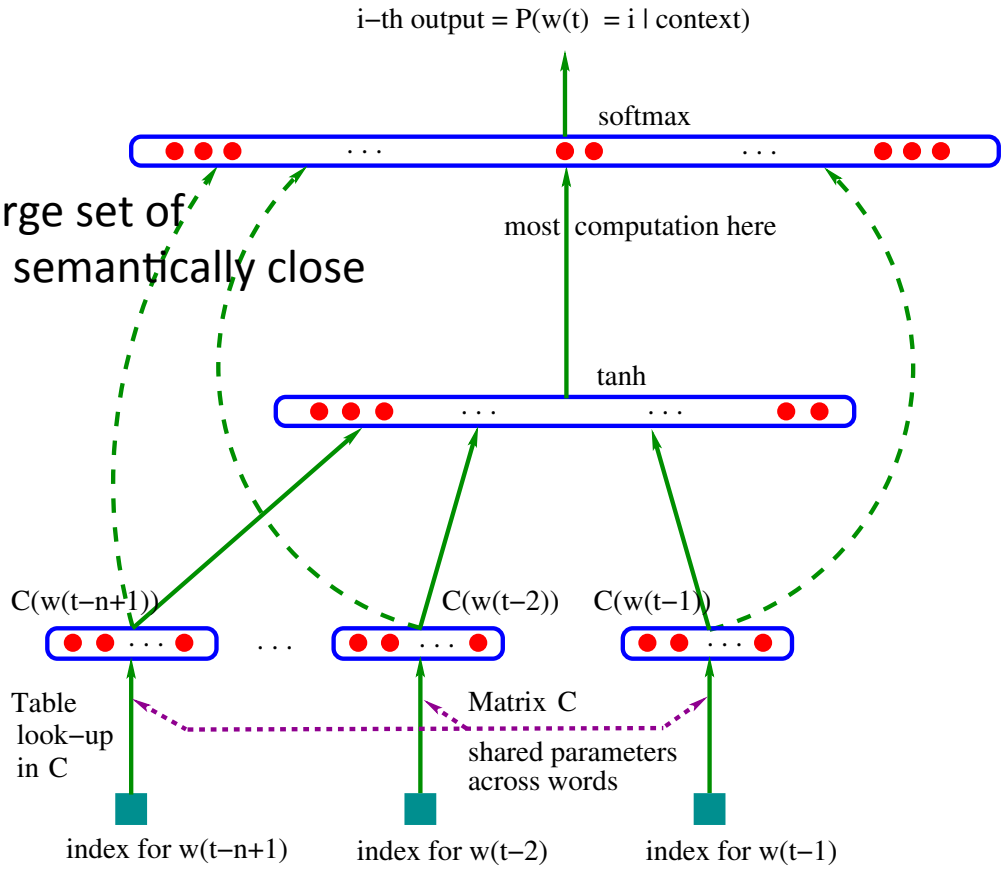


Neural Language Models: fighting one exponential by another one!

- (Bengio et al NIPS'2000)

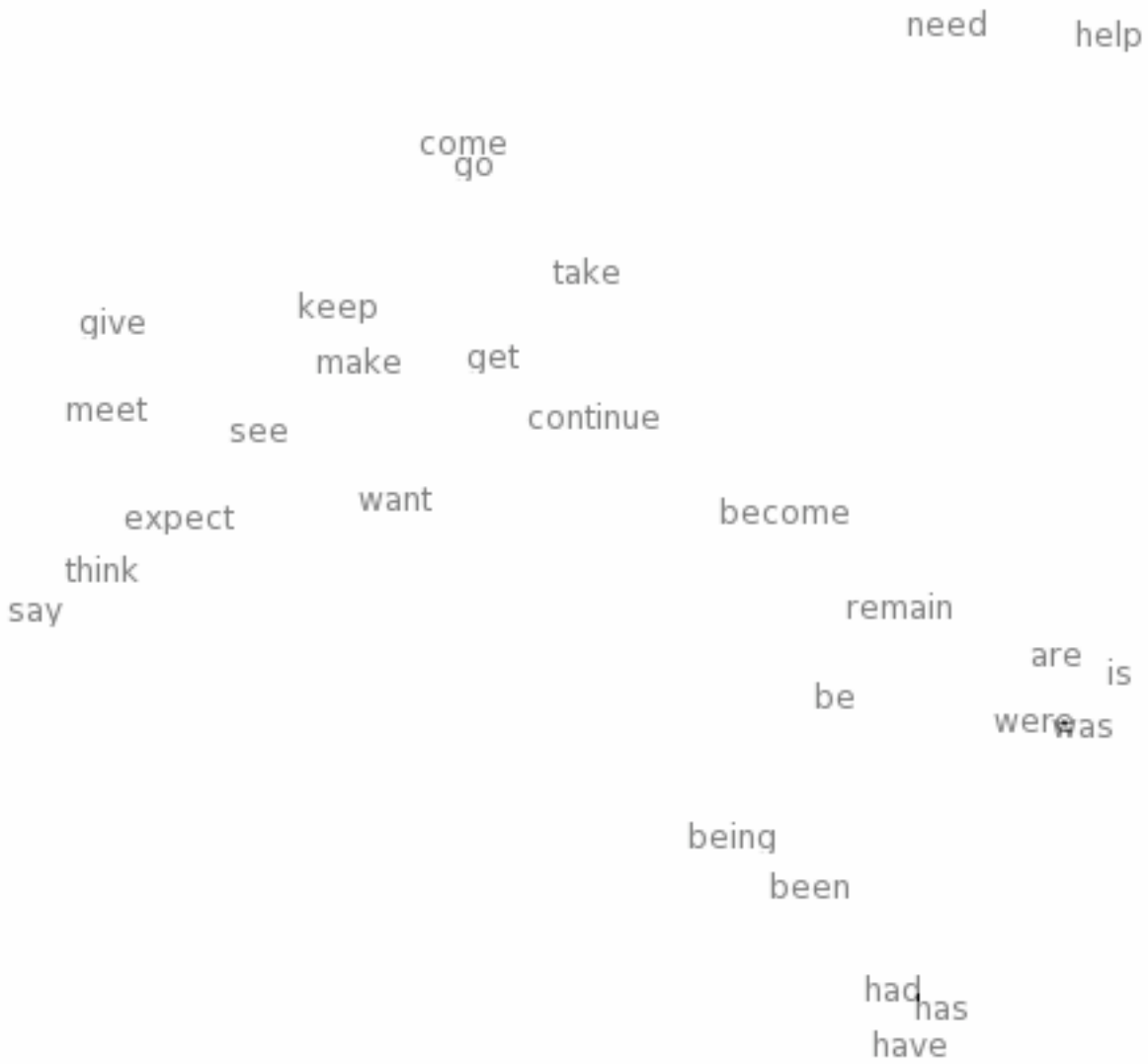


Exponentially large set of generalizations: semantically close sequences



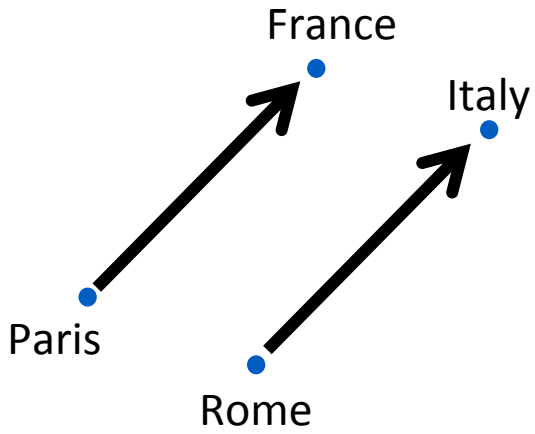
Exponentially large set of possible contexts

Neural word embeddings - visualization Directions = Learned Attributes



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



Summary of New Theoretical Results

- Expressiveness of deep networks with piecewise linear activation functions: exponential advantage for depth

  (Montufar et al NIPS 2014)

- Theoretical and empirical evidence against bad local minima

  (Dauphin et al NIPS 2014)

- Manifold & probabilistic interpretations of auto-encoders
  - Estimating the gradient of the energy function *(Alain & Bengio ICLR 2013)*
  - Sampling via Markov chain *(Bengio et al NIPS 2013)*
  - Variational auto-encoder breakthrough *(Gregor et al arXiv 2015)*

The Depth Prior can be Exponentially Advantageous

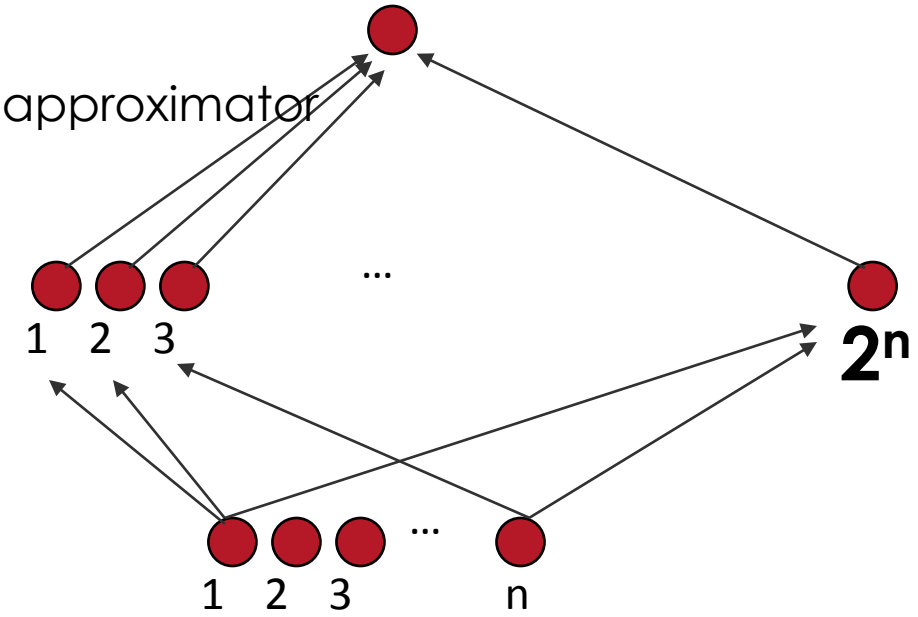
Theoretical arguments:

2 layers of { Logic gates / Formal neurons / RBF units } = universal approximator

RBMs & auto-encoders = universal approximator

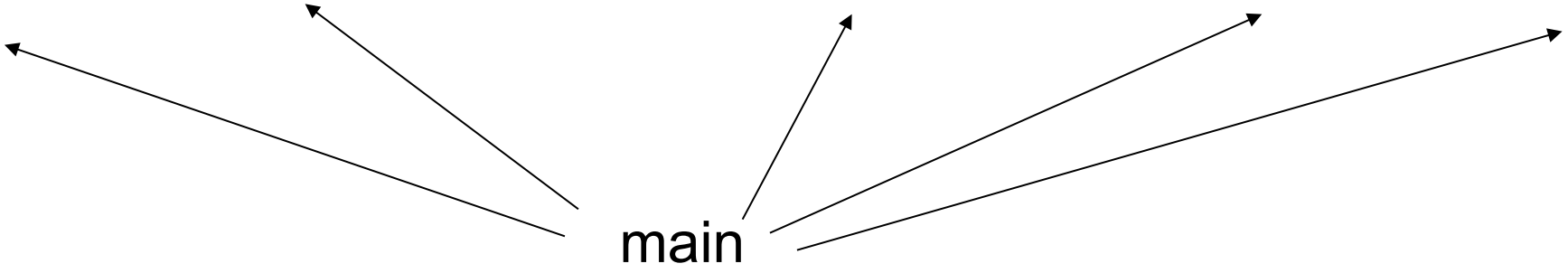
Theorems on advantage of depth:
(Hastad et al 86 & 91, Bengio et al 2007, Bengio & Delalleau 2011, Braverman 2011, Pascanu et al 2014, Montufar et al **NIPS 2014**)

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

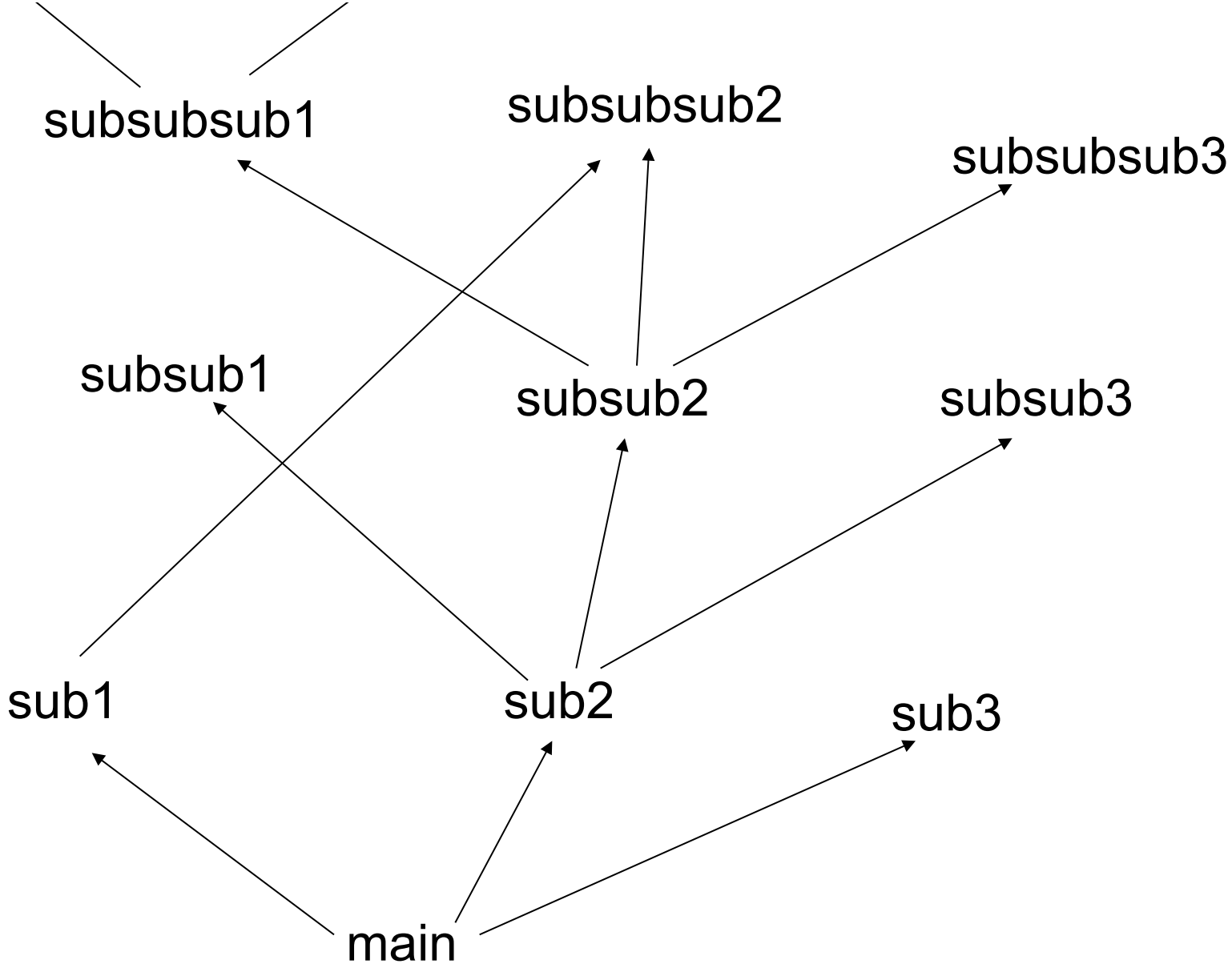


subroutine1 includes
subsub1 code and
subsub2 code and
subsubsub1 code

subroutine2 includes
subsub2 code and
subsub3 code and
subsubsub3 code and …



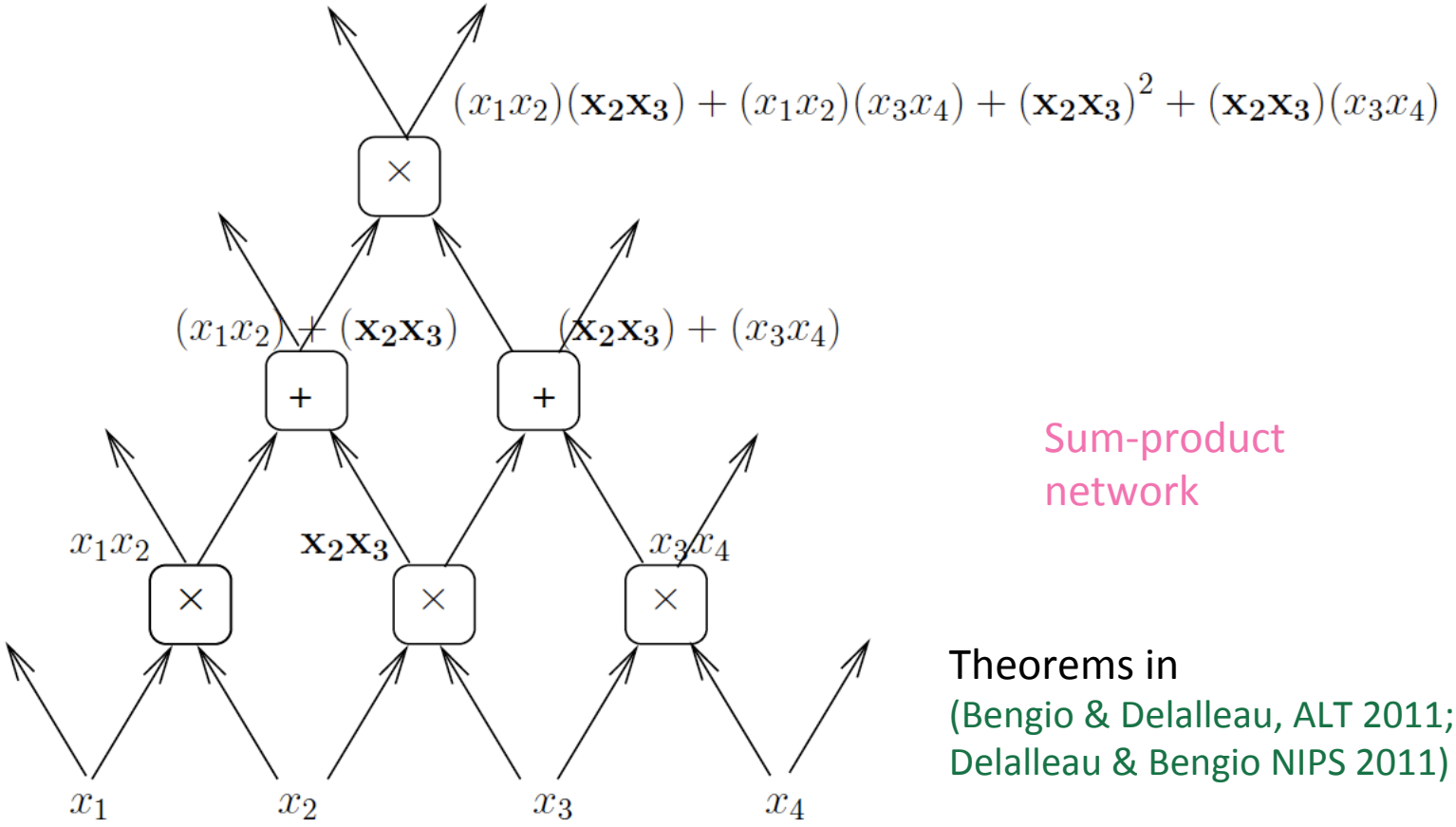
“Shallow” computer program



“Deep” computer program

Sharing Components in a Deep Architecture

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



Sum-product network

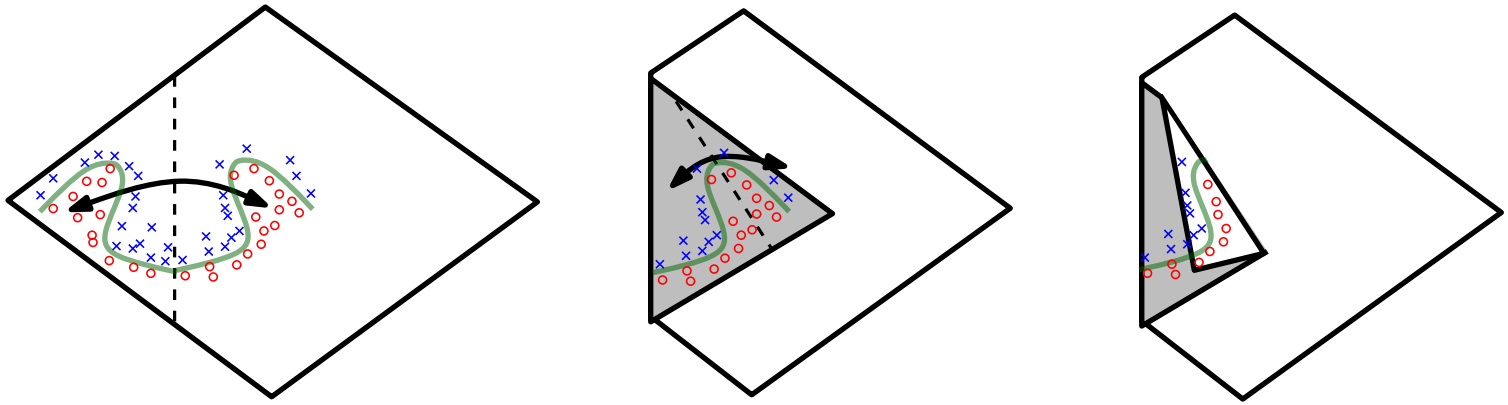
Theorems in
(Bengio & Delalleau, ALT 2011;
Delalleau & Bengio NIPS 2011)

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

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

(Montufar, Pascanu, Cho & Bengio; NIPS 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:

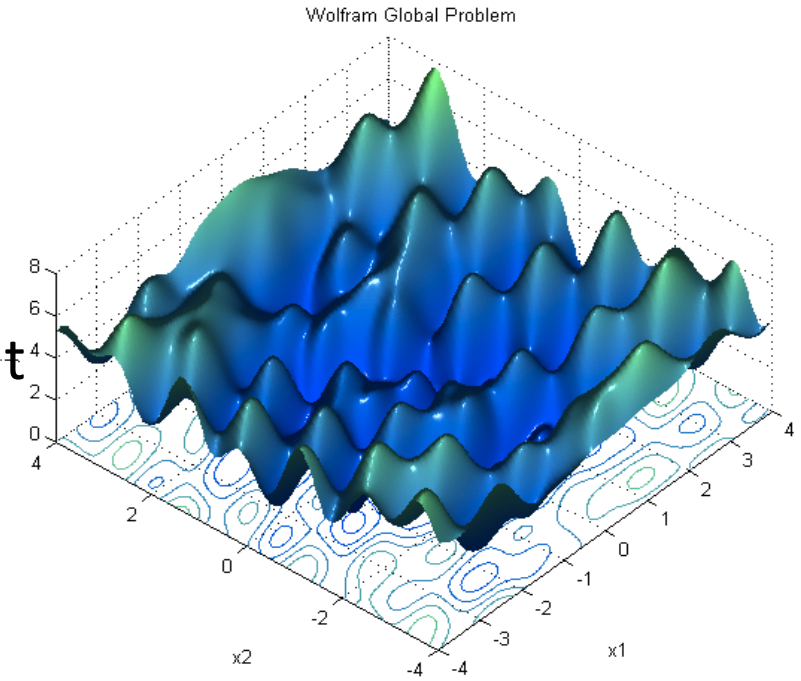


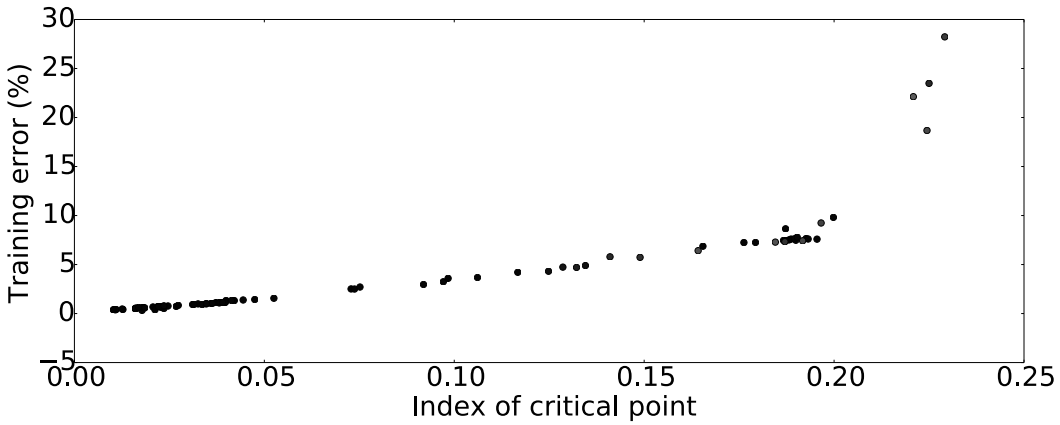
A Myth is Being Debunked: Local Minima in Neural Nets → Convexity is not needed

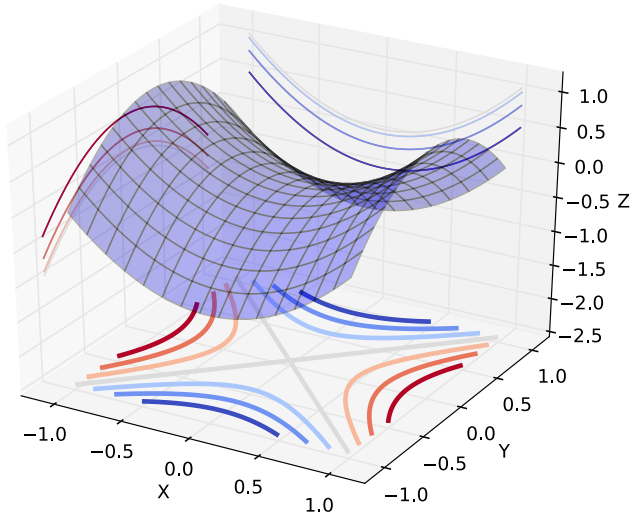
- (Pascanu, Dauphin, Ganguli, Bengio, arXiv May 2014): *On the saddle point problem for non-convex optimization*
- (Dauphin, Pascanu, Gulcehre, Cho, Ganguli, Bengio, NIPS' 2014): *Identifying and attacking the saddle point problem in high-dimensional non-convex optimization*
- (Choromanska, Henaff, Mathieu, Ben Arous & LeCun 2014): *The Loss Surface of Multilayer Nets*

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)

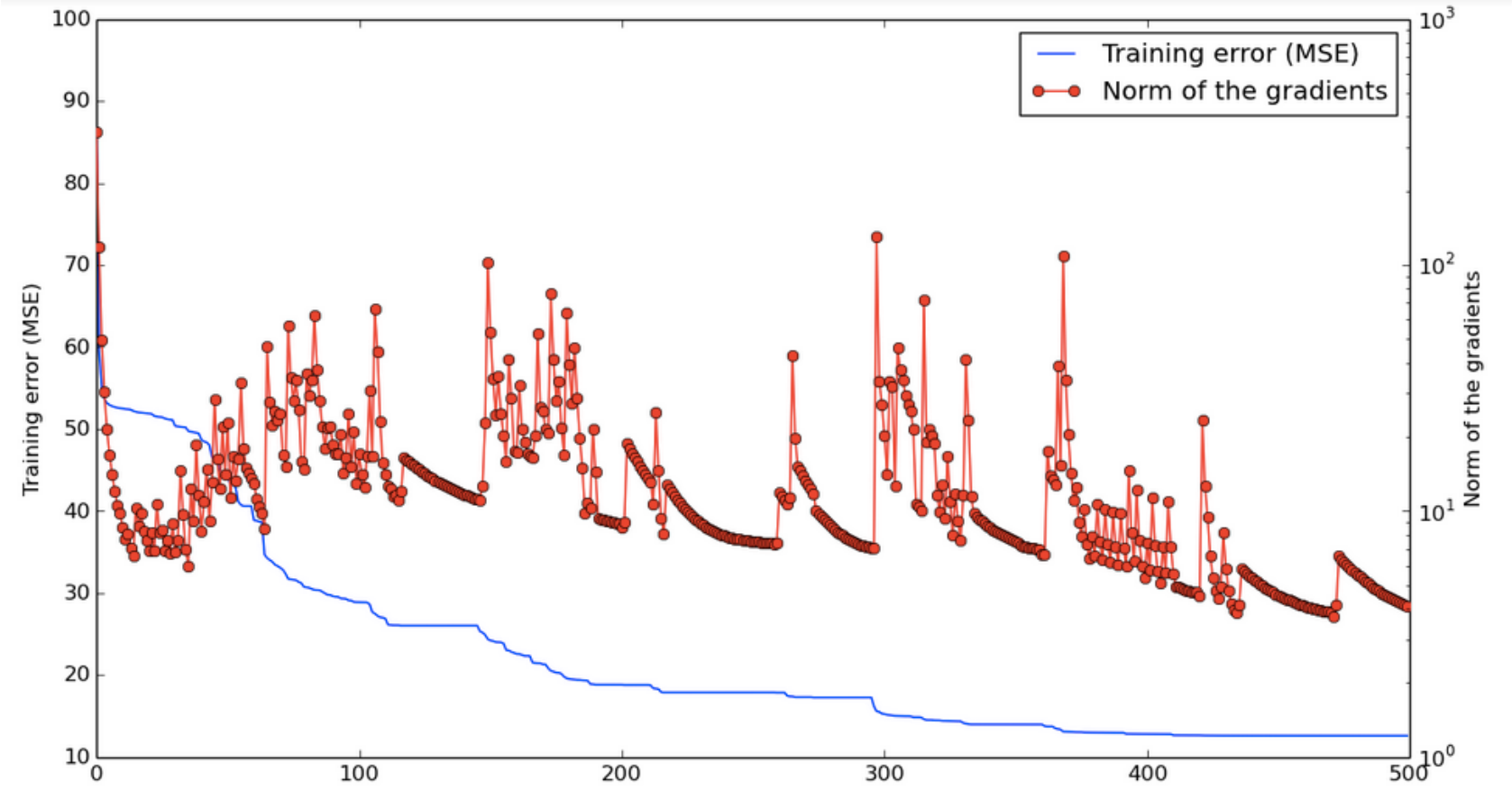






Saddle Points During Training

- Oscillating between two behaviors:
  - Slowly approaching a saddle point
  - Escaping it

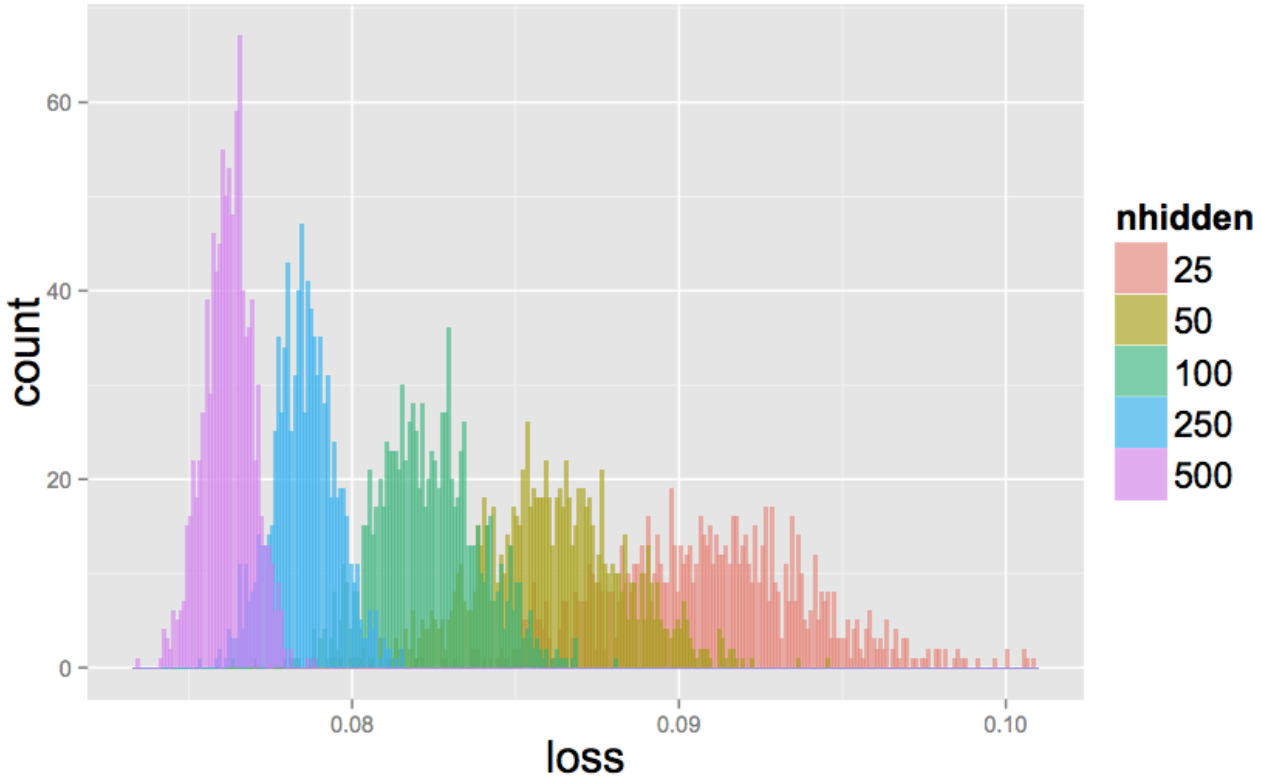


Low Index Critical Points

Choromanska et al & LeCun 2014, 'The Loss Surface of Multilayer Nets'

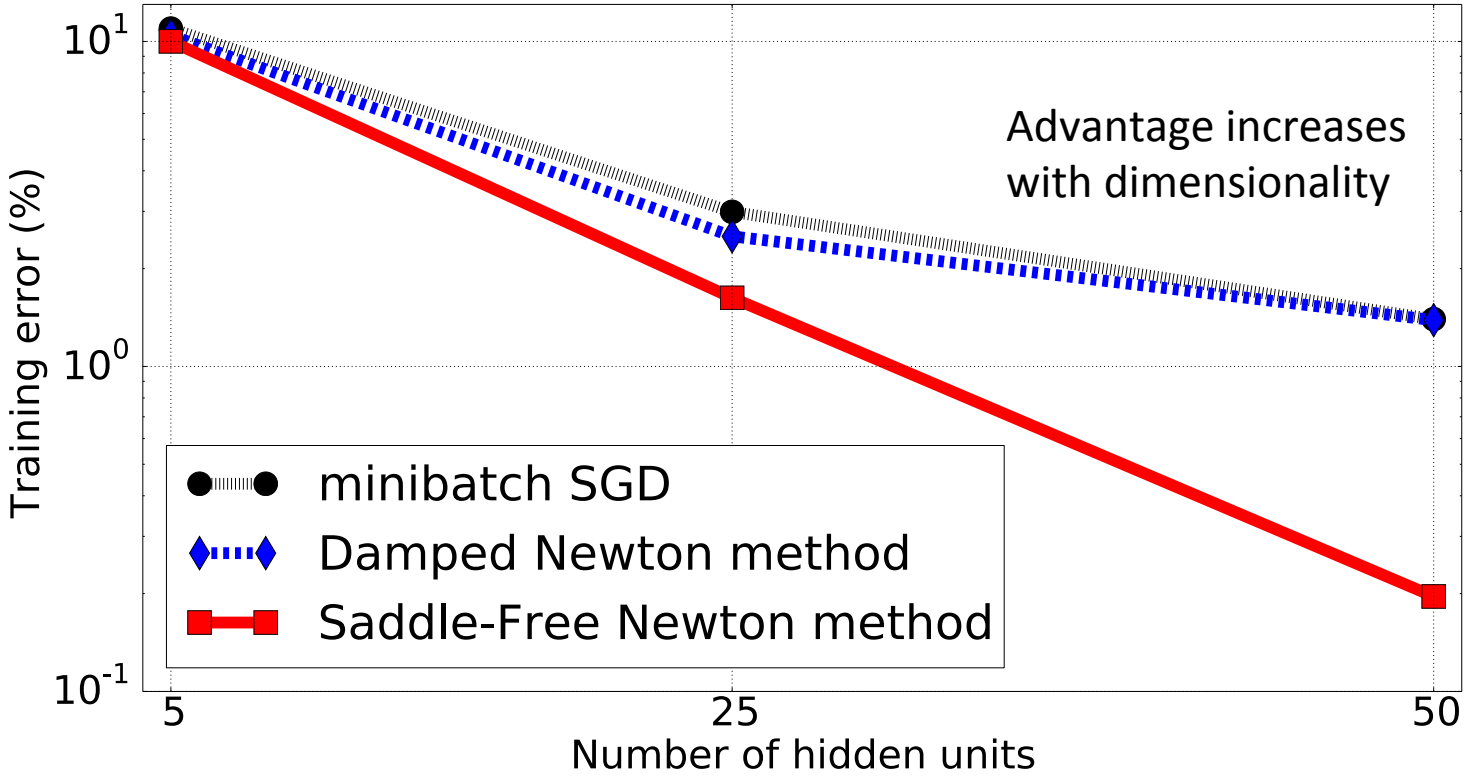
Shows that deep rectifier nets are analogous to spherical spin-glass models

The low-index critical points of large models concentrate in a band just above the global minimum



Saddle-Free Optimization (Pascanu, Dauphin, Ganguli, Bengio 2014)

- Saddle points are ATTRACTIVE for Newton's method
- Replace eigenvalues λ of Hessian by |λ|
- Justified as a particular trust region method



How do humans generalize from very few examples?

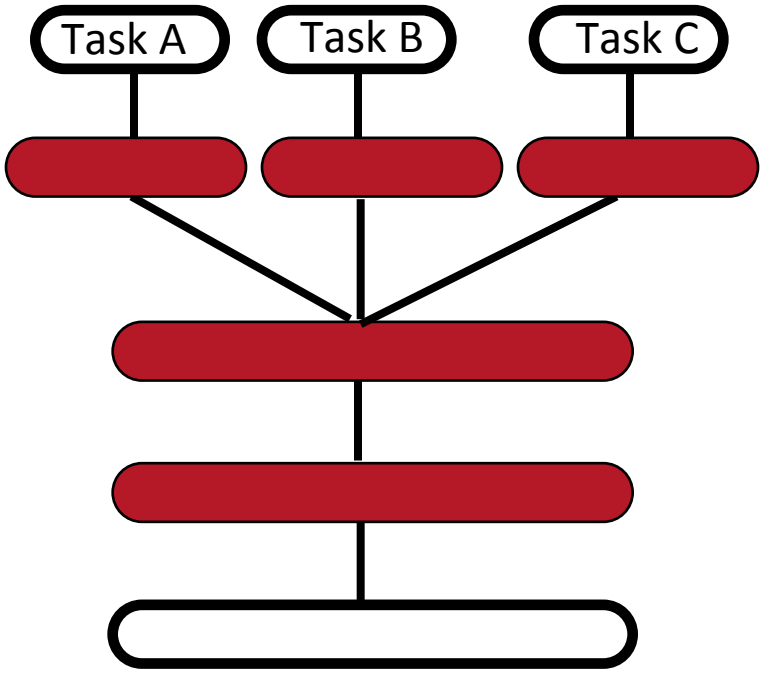
- They **transfer** knowledge from previous learning:
  - Representations
  - Explanatory factors
- Previous learning from: unlabeled data

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

Multi-Task Learning

- Generalizing better to new tasks (tens of thousands!) is crucial to approach AI
- Deep architectures learn good intermediate representations that can be shared across tasks

  (Collobert & Weston ICML 2008, Bengio et al AISTATS 2011)
- Good representations that disentangle underlying factors of variation make sense for many tasks because **each task concerns a subset of the factors**

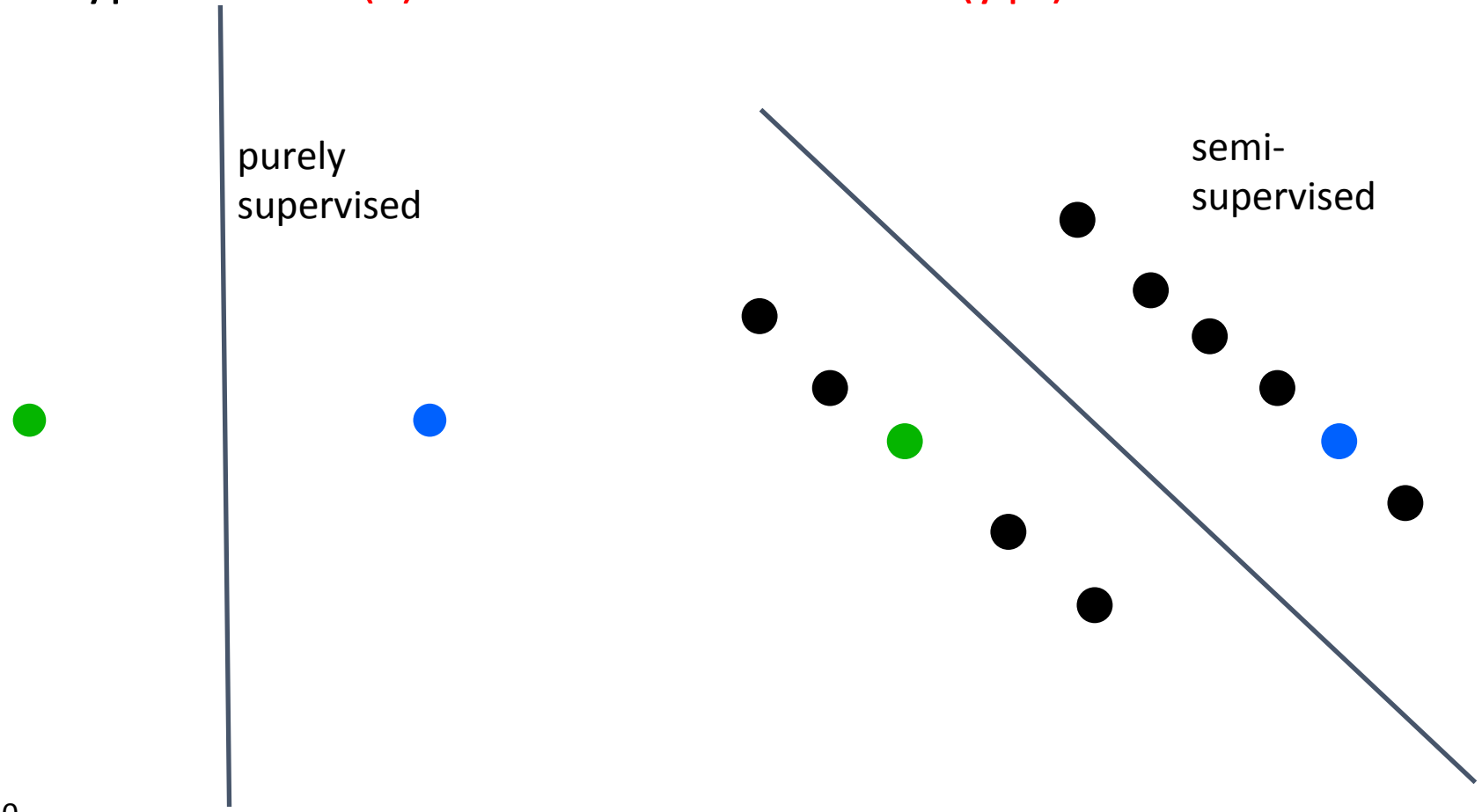


E.g. dictionary, with intermediate concepts re-used across many definitions

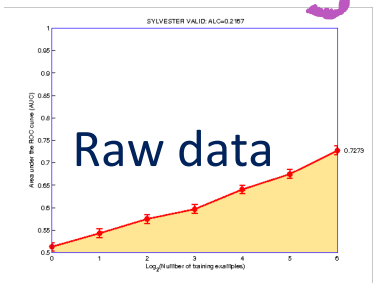
Prior: shared underlying explanatory factors between tasks

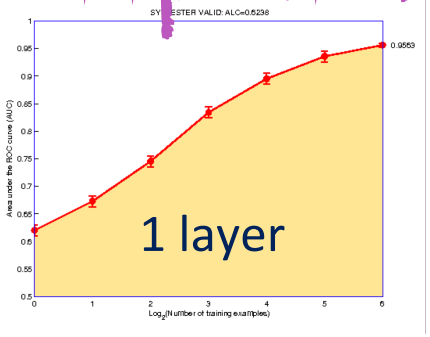
Sharing Statistical Strength by Semi-Supervised Learning

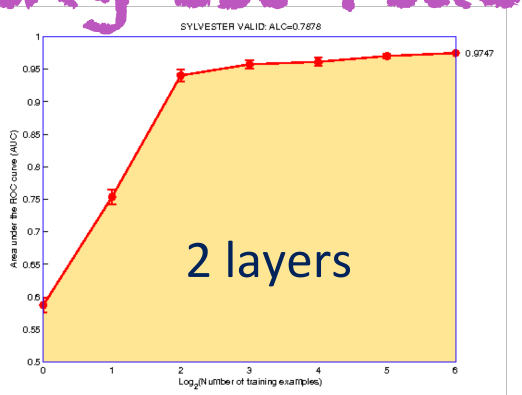
- Hypothesis: P(x) shares structure with P(y|x)



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

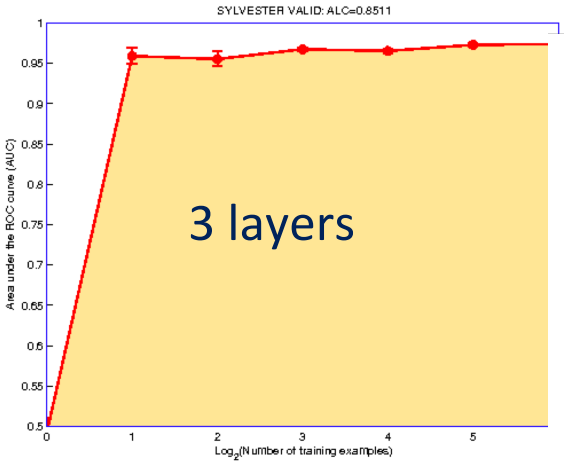


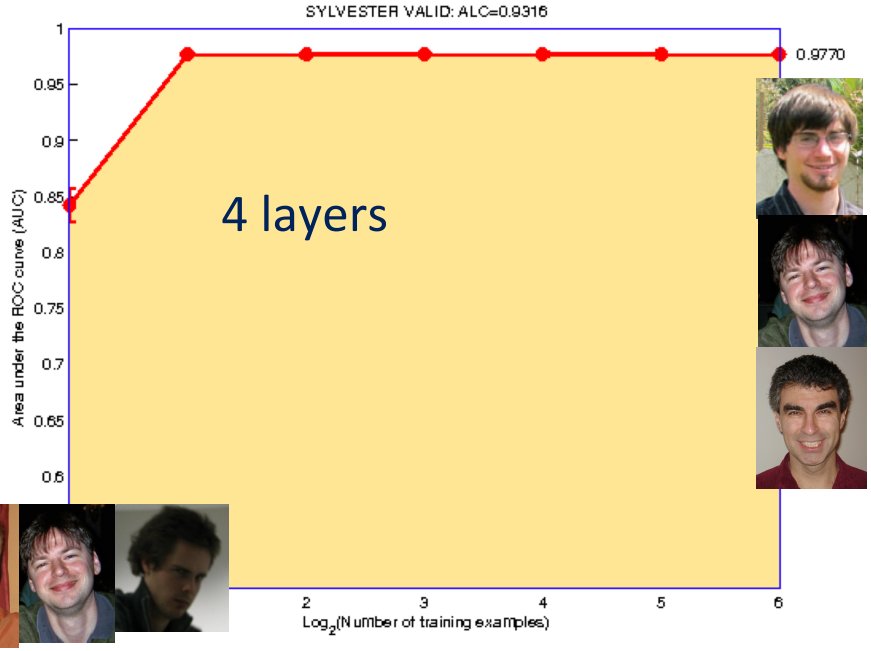




NIPS'2011 Transfer Learning Challenge Paper: ICML'2012

ICML'2011 workshop on Unsup. & Transfer Learning





The Next Challenge: Unsupervised Learning

- Recent progress mostly in supervised DL
- Real technical 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

Why Latent Factors & 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.

Invariance and Disentangling

- Invariant features

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

  avoid the curse of dimensionality

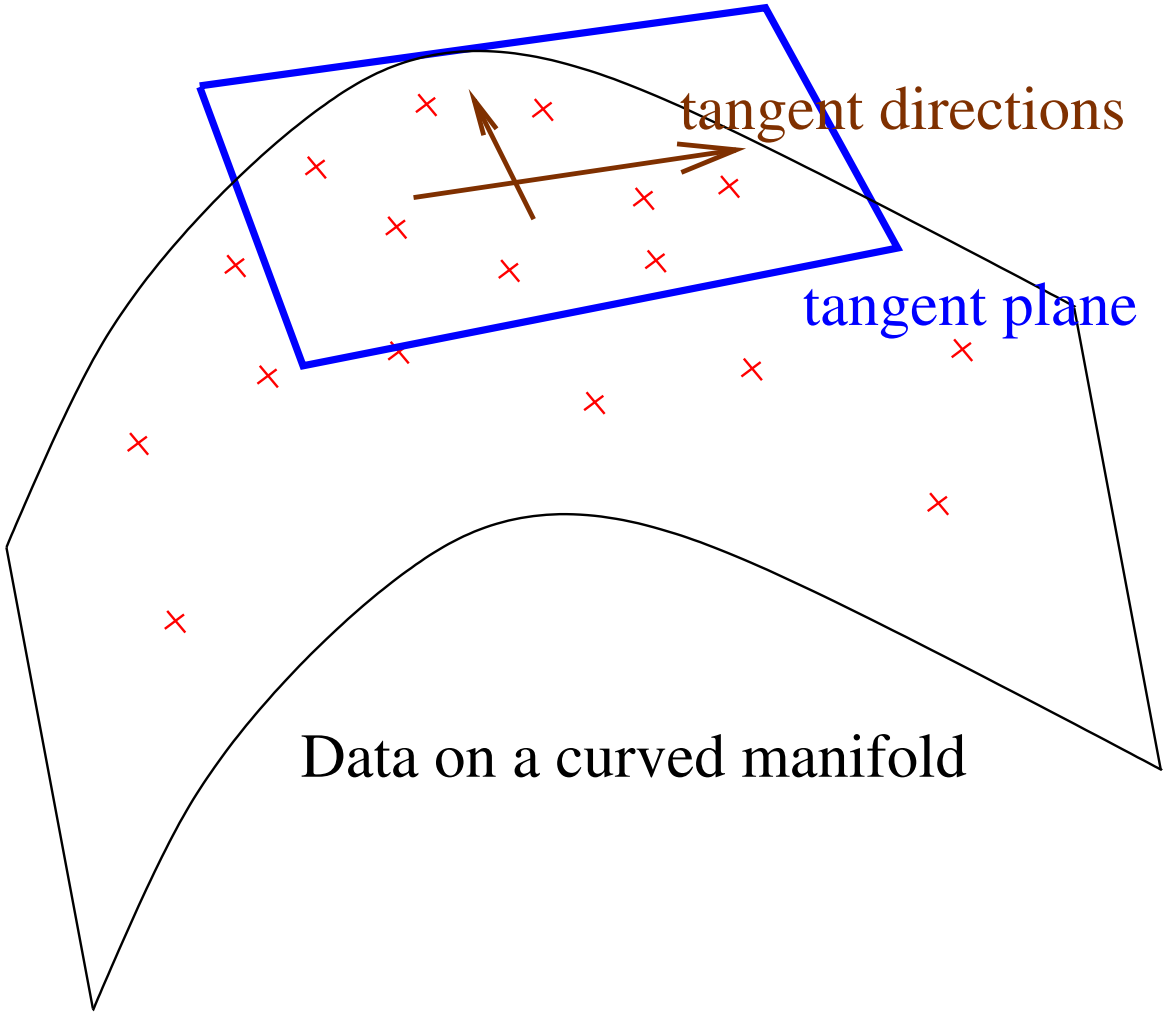
Emergence of Disentangling

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

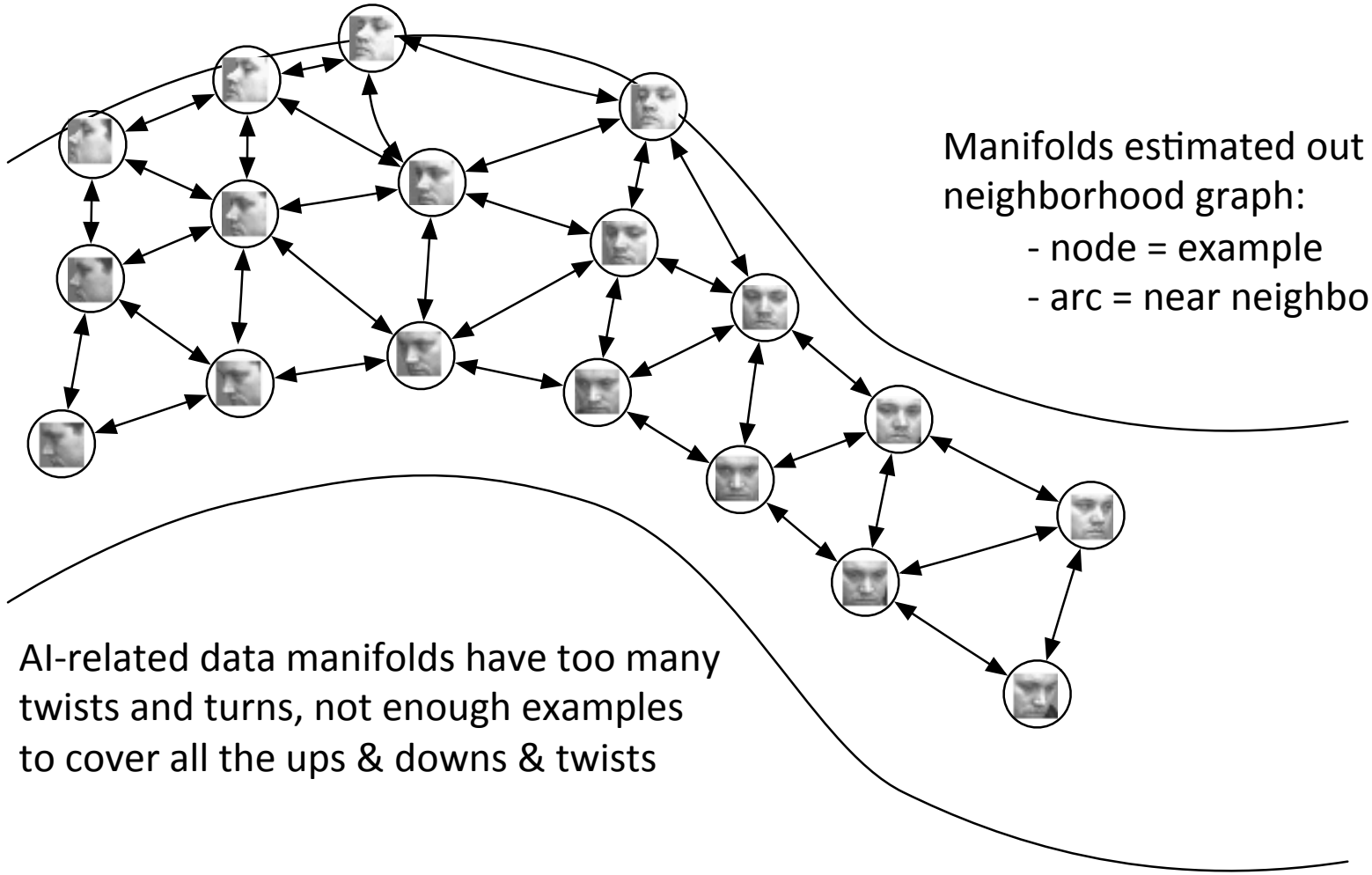
- (Glorot et al. 2011): sparse rectified denoising auto-encoders trained on bags of words for sentiment analysis
  - different features specialize on different aspects (domain, sentiment)

WHY?

Manifold Learning = Representation Learning



Non-Parametric Manifold Learning: hopeless without powerful enough priors

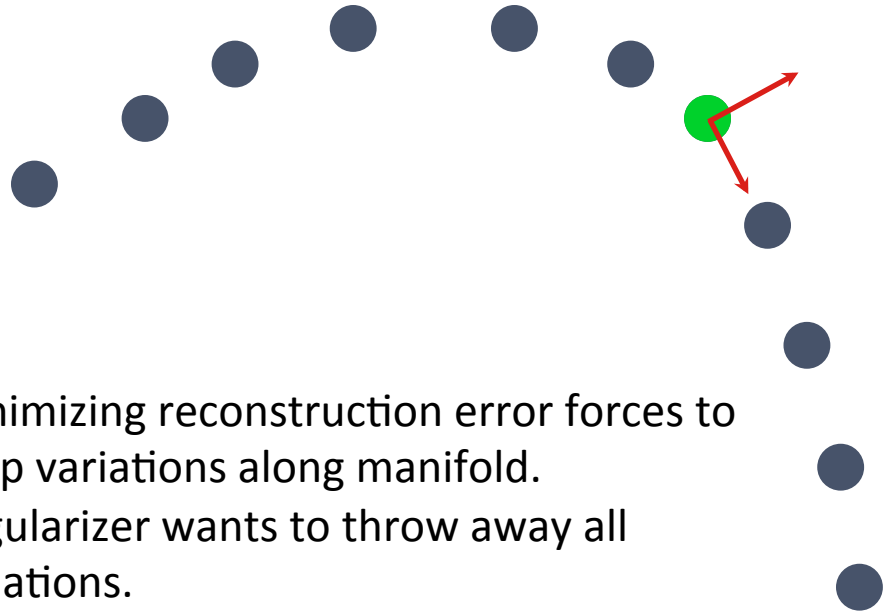


Manifolds estimated out of the neighborhood graph:

- node = example
- arc = near neighbor

AI-related data manifolds have too many twists and turns, not enough examples to cover all the ups & downs & twists

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.

Denoising Auto-Encoder

- Learns a vector field pointing towards higher probability direction (Alain & Bengio 2013)

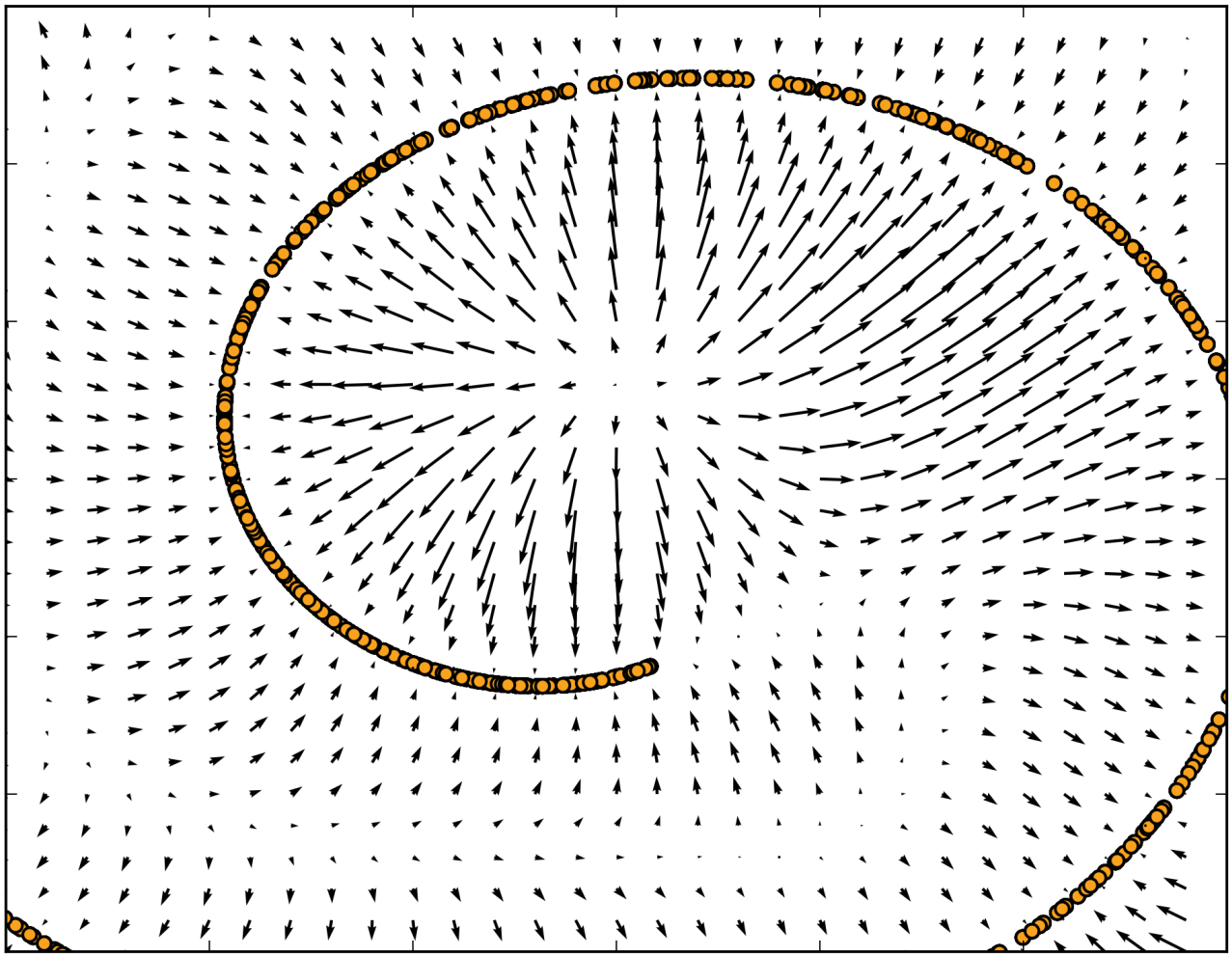
$$\text{reconstruction}(x) - x \quad \rightarrow \quad \sigma^2 \frac{\partial \log p(x)}{\partial x}$$

- Some DAEs correspond to a kind of Gaussian RBM with *regularized* Score Matching (Vincent 2011)

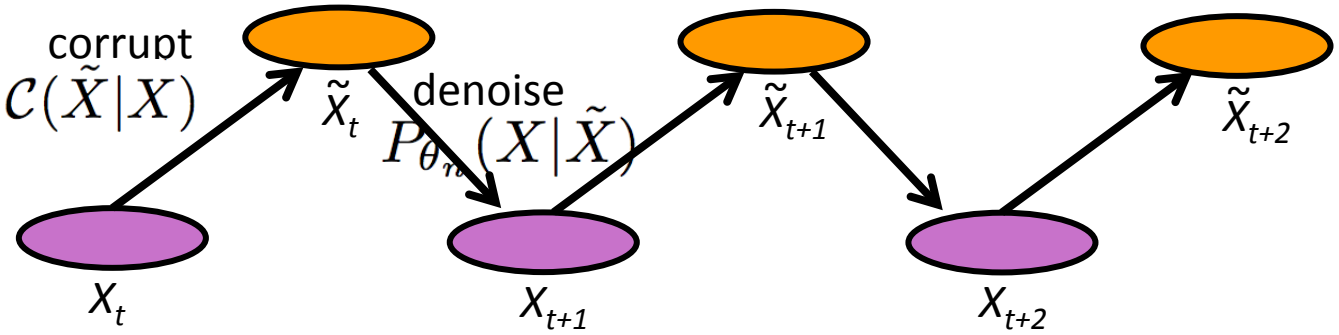
  [equivalent when noise→0]

prior: examples concentrate near a lower dimensional "manifold"

Regularized Auto-Encoders Learn a Vector Field that Estimates a Gradient Field (Alain & Bengio ICLR 2013)

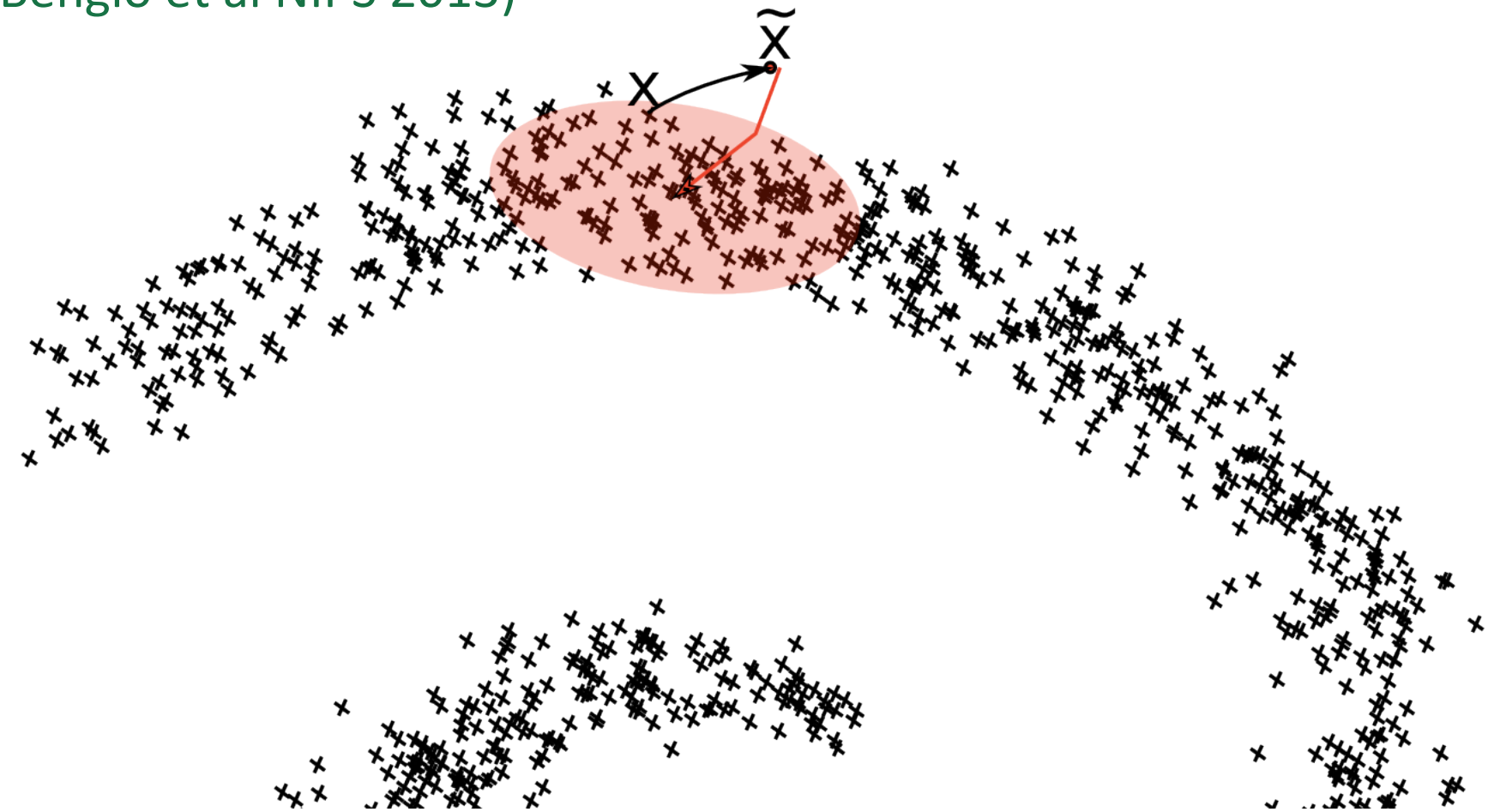


Denoising Auto-Encoder Markov Chain



Denoising Auto-Encoders Learn a Markov Chain Transition Distribution

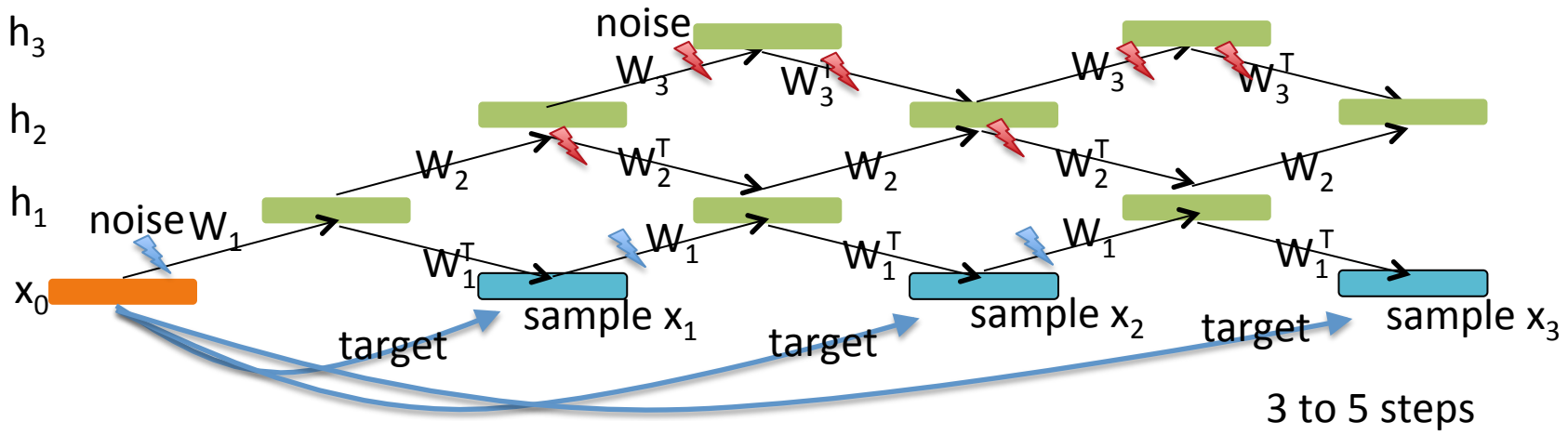
(Bengio et al NIPS 2013)



Generative Stochastic Networks (GSN)

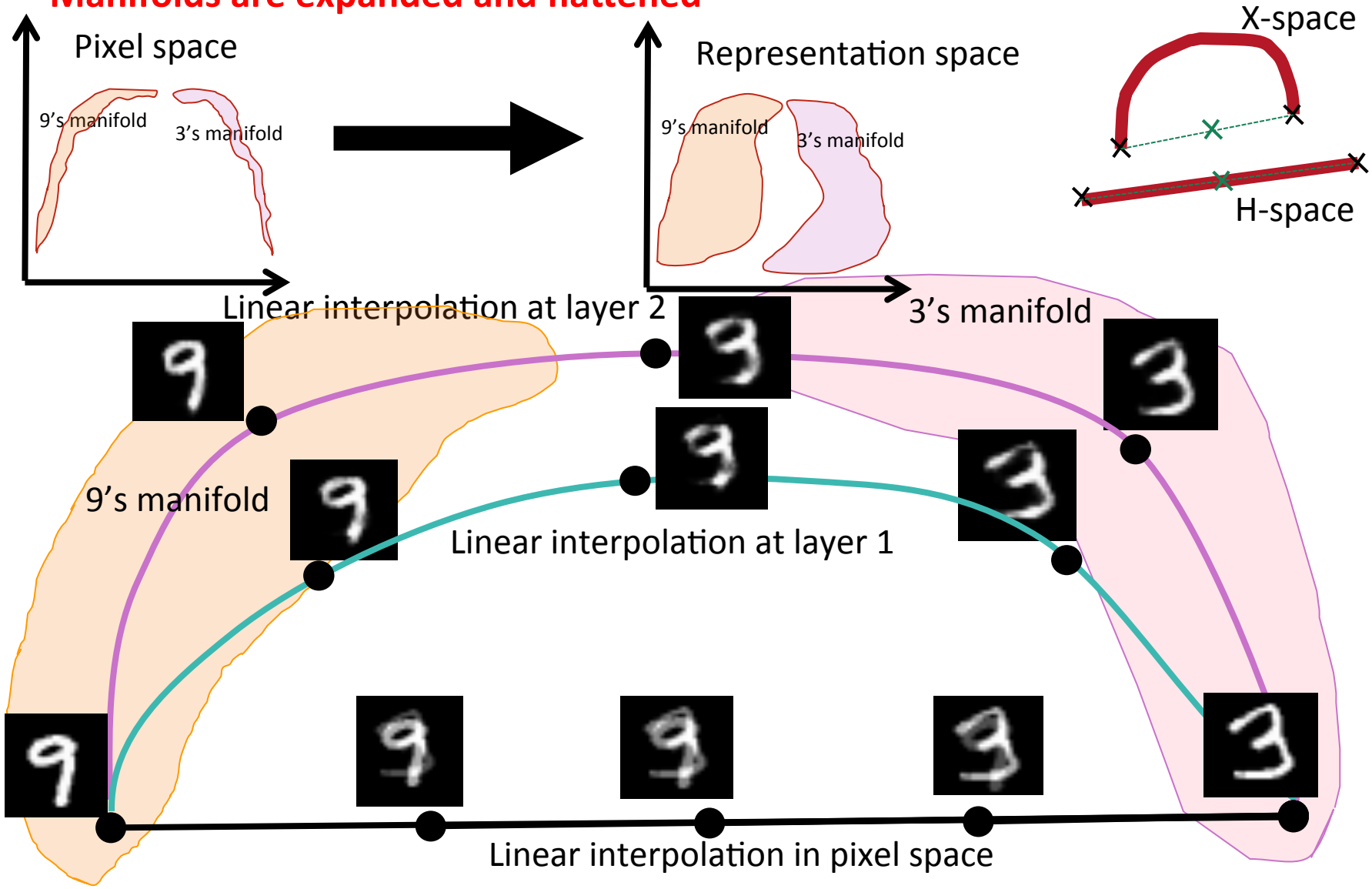
(Bengio et al ICML 2014, Alain et al arXiv 2015)

- **Recurrent parametrized stochastic computational graph that defines a transition operator for a Markov chain whose asymptotic distribution is implicitly estimated by the model**
- Noise injected in input and hidden layers
- Trained to max. reconstruction prob. of example at each step
- **Example** structure inspired from the DBM Gibbs chain:



Space-Filling in Representation-Space

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

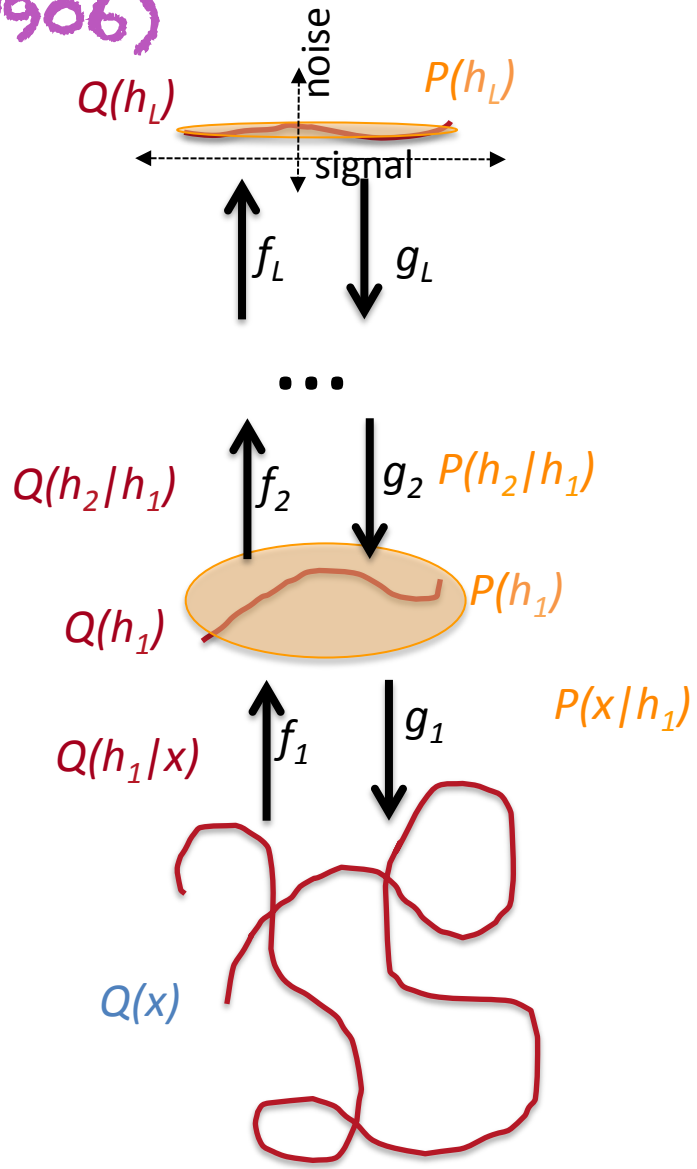


Extracting Structure By Gradual Disentangling and Manifold Unfolding (Bengio 2014, arXiv 1407.7906)

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.

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

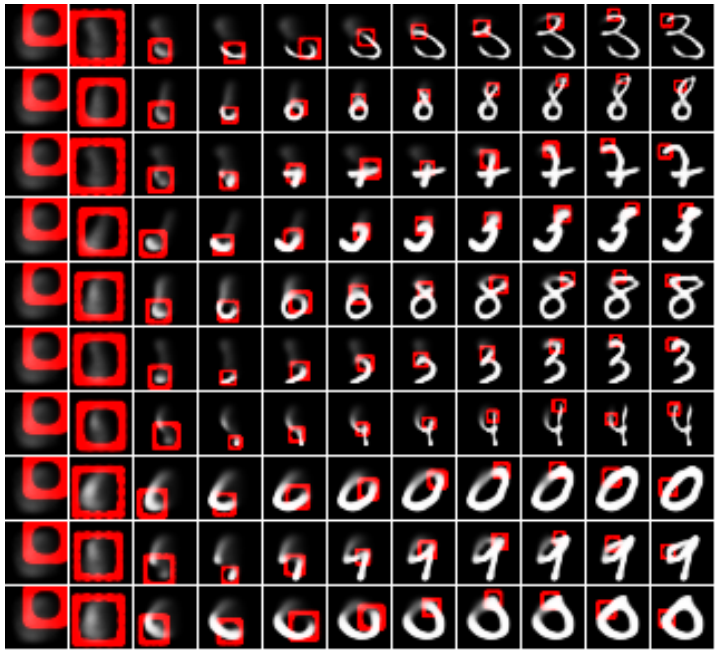
for each intermediate level h

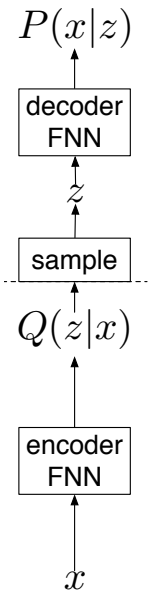


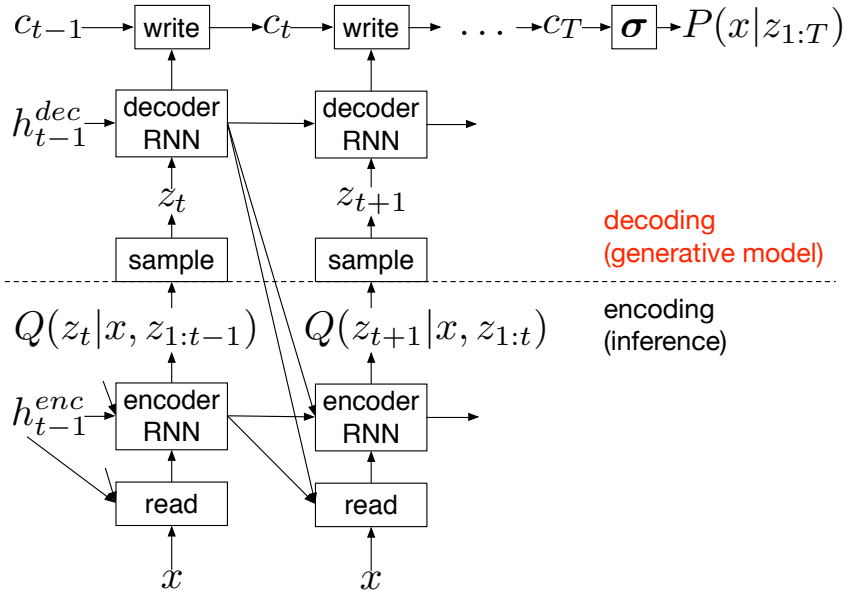
DRAW: the latest variant of Variational Auto-Encoder

(Gregor et al of Google DeepMind, arXiv 1502.04623, 2015)

- Even for a static input, the encoder and decoder are now **recurrent nets**, which gradually add elements to the answer, and use an attention mechanism to choose where to do so.







DRAW Samples of SVHN Images: the drawing process

DRAW Samples of SVHN Images: generated samples vs training nearest neighbor

Nearest training example for last column of samples

Conclusions

- **Distributed representations**:
  - prior that can buy exponential gain in generalization
- **Deep composition of non-linearities**:
  - prior that can buy exponential gain in generalization
- Both yield **non-local generalization**
- Strong evidence that **local minima are not an issue, saddle points**
- **Auto-encoders capture the data generating distribution**
  - Gradient of the energy
  - Markov chain generating an estimator of the dgd
  - Can be generalized to deep generative models

MILA: Montreal Institute for Learning Algorithms