Learning Representations for Unsupervised and Transfer Learning

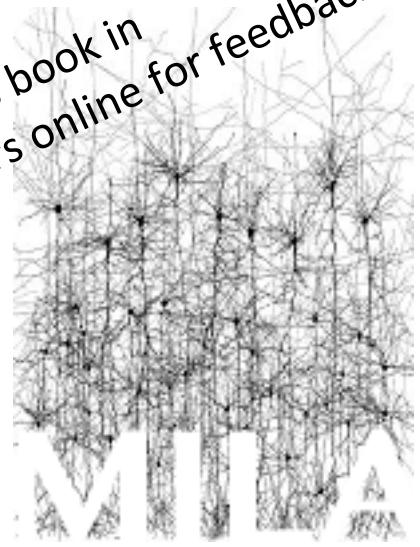
Yoshua Bengio

December 12, 2015

Transfer and Multi-Task Learning

NIPS'2015 Workshop

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



Why Unsupervised Representation Learning?

Representation & outputs = explanatory factors of X

- Recent progress mostly in supervised DL
- Real technical challenges for unsupervised DL
- If Y is a cause of X, then learning P(X) can help P(Y|X)

(Janzing et al ICML 2012)

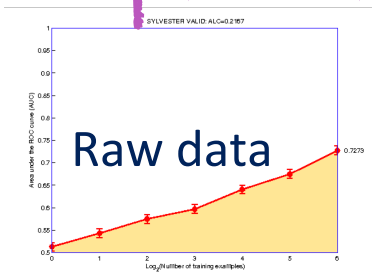
- Potential benefits:
  - **Exploit tons of unlabeled data**
  - Answer new questions about the variables observed
  - Regularizer – transfer learning – domain adaptation
  - Easier optimization (local training signal)
  - Structured outputs

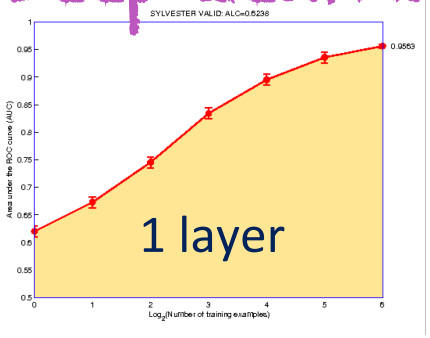
How do humans generalize from very few examples?

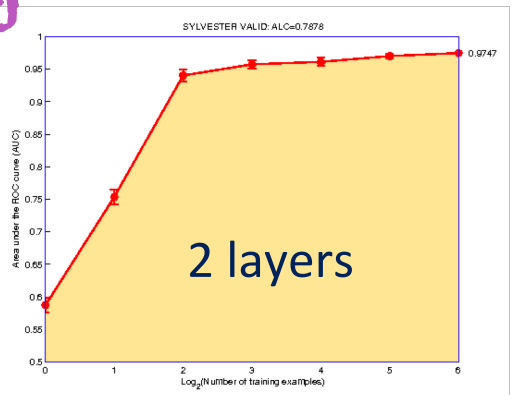
- They **transfer** knowledge from previous learning:
  - Representations
  - Explanatory factors
- Previous learning from: unlabeled data

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

Unsupervised and Transfer Learning Challenge + Transfer Learning Challenge: Won by Unsupervised Deep Learning

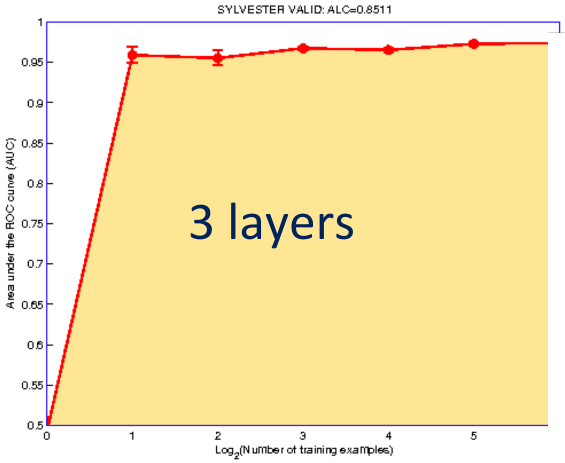


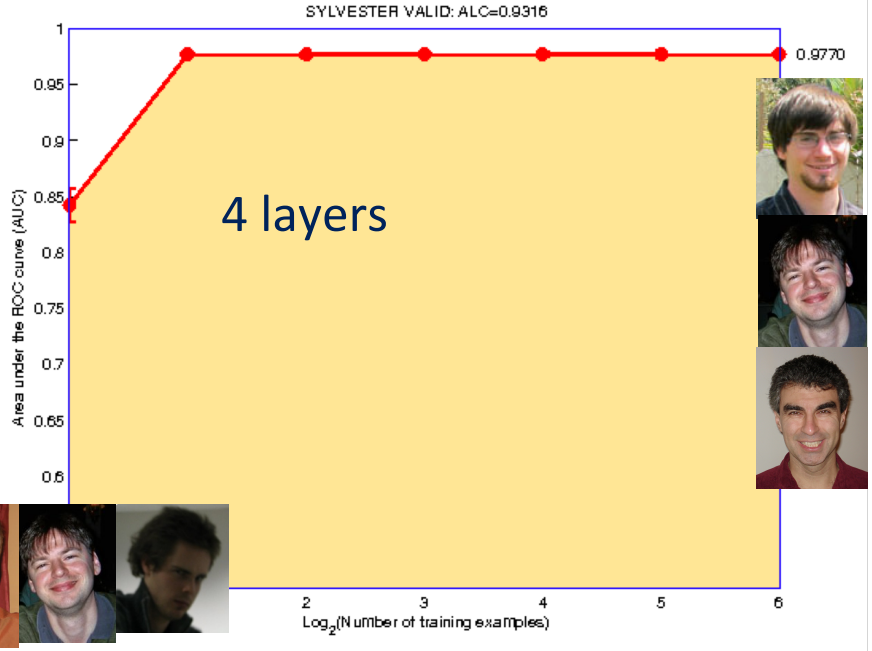




NIPS'2011 Transfer Learning Challenge Paper: ICML'2012

ICML'2011 workshop on Unsup. & Transfer Learning





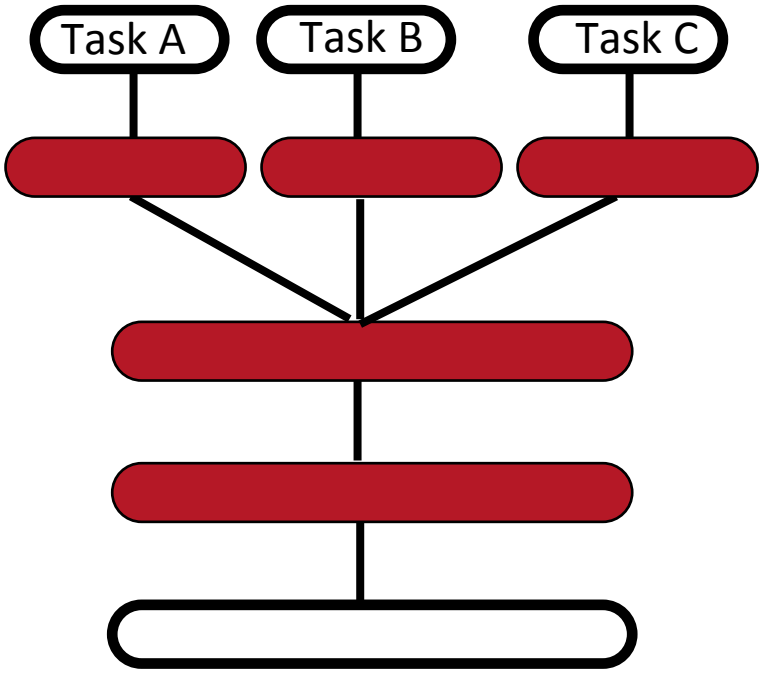
Multi-Task Learning

- Generalizing better to new tasks (tens of thousands!) is crucial to approach AI
- Example: speech recognition, sharing across multiple languages

- 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

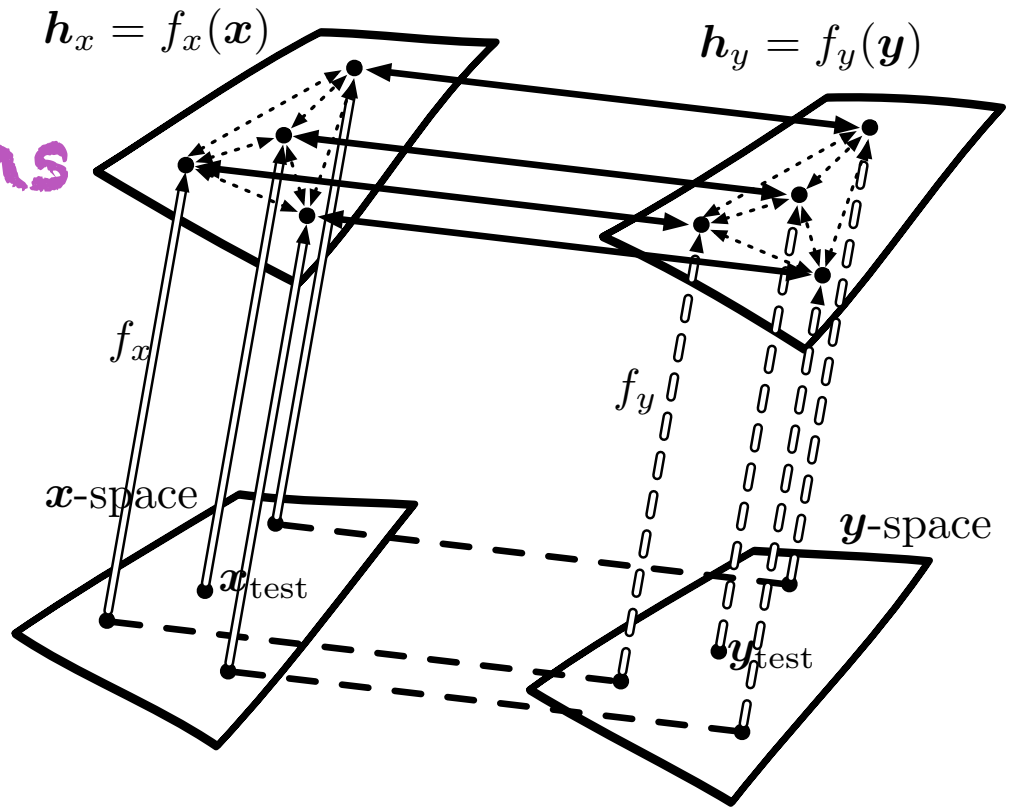
Prior: shared underlying explanatory factors between tasks

Maps Between Representations

x and *y* represent different modalities, e.g., image, text, sound…

Can provide 0-shot generalization to new categories (values of *y*)

(Larochelle et al AAAI 2008)

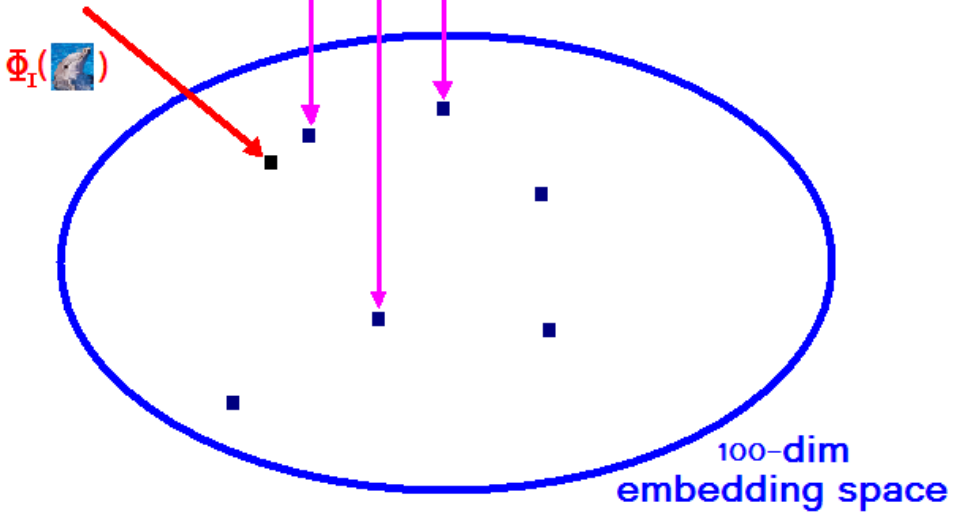


Google Image Search Joint Embedding: different object types represented in same space

Google:

S. Bengio, J. Weston & N. Usunier

(IJCAI 2011, NIPS'2010, JMLR 2010, ML J 2010)

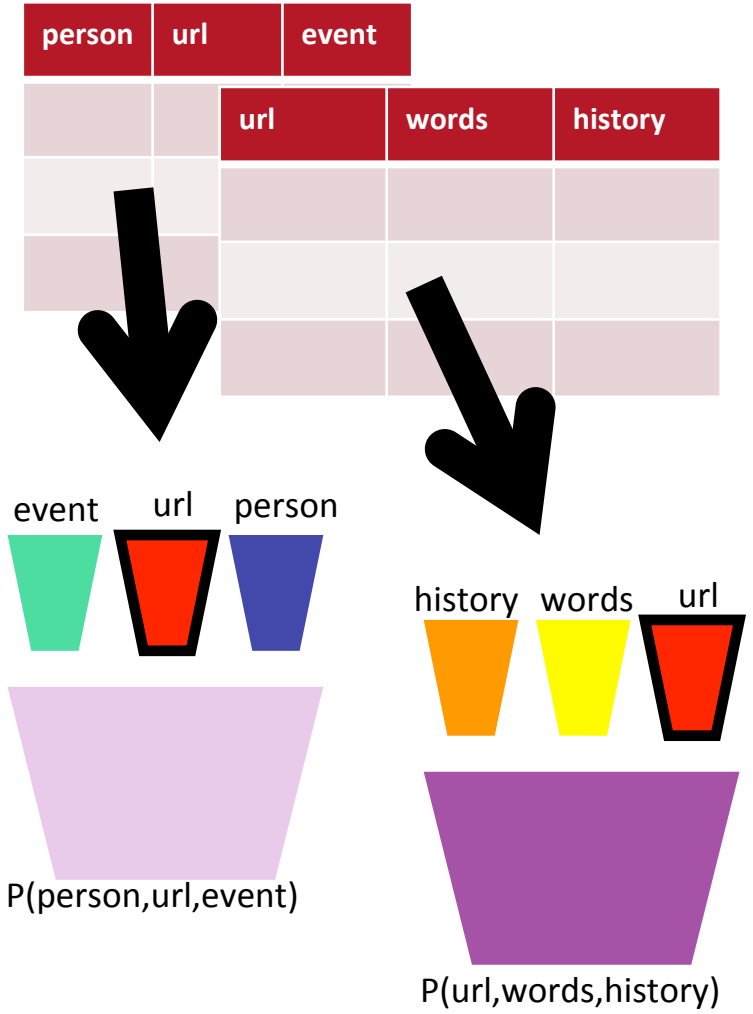


WSABIE objective function:

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

Combining Multiple Sources of Evidence with Shared Representations

- Traditional ML: data = matrix
- Relational learning: multiple sources, different tuples of variables
- Share representations of same types across data sources
- Shared learned representations help propagate information among data sources: e.g., WordNet, XWN, Wikipedia, **FreeBase**, ImageNet…
  (Bordes et al AISTATS 2012, ML J. 2013)
- **FACTS = DATA**
- **Deduction = Generalization**



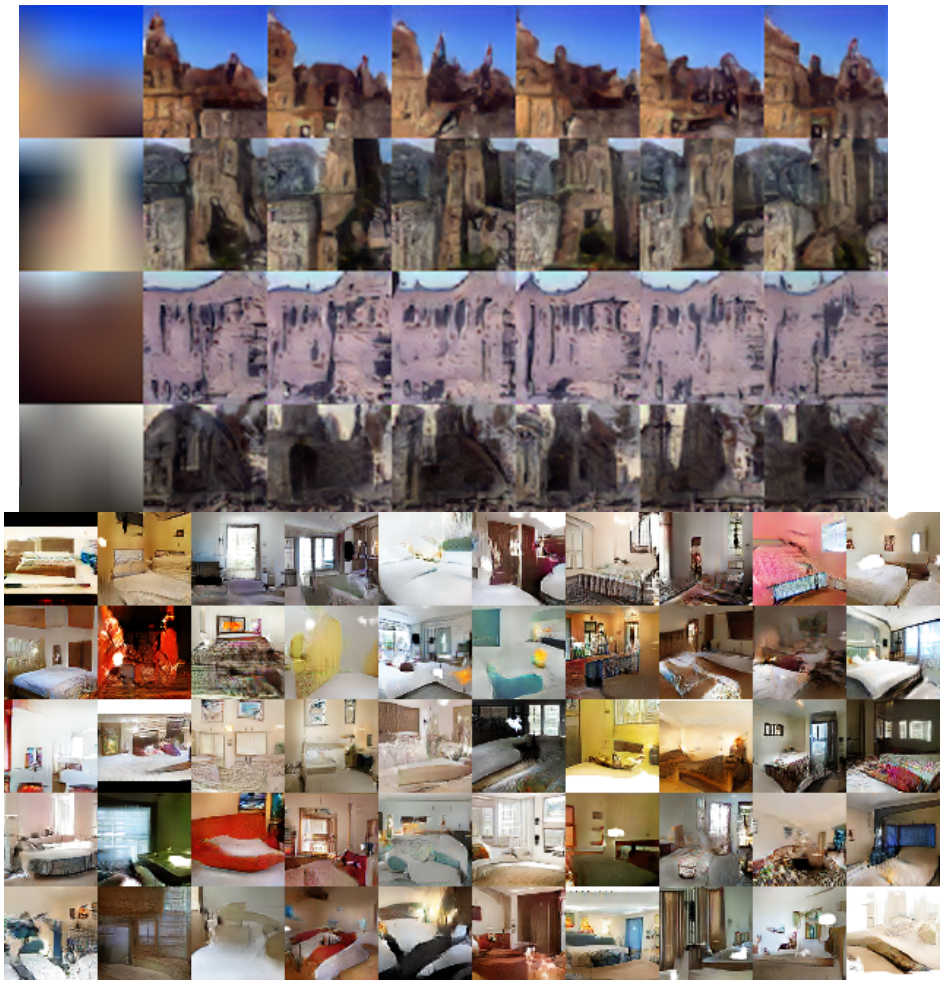
Deep Generative Learning: Hot Frontier

- Many very different approaches being explored
- Exciting area of research

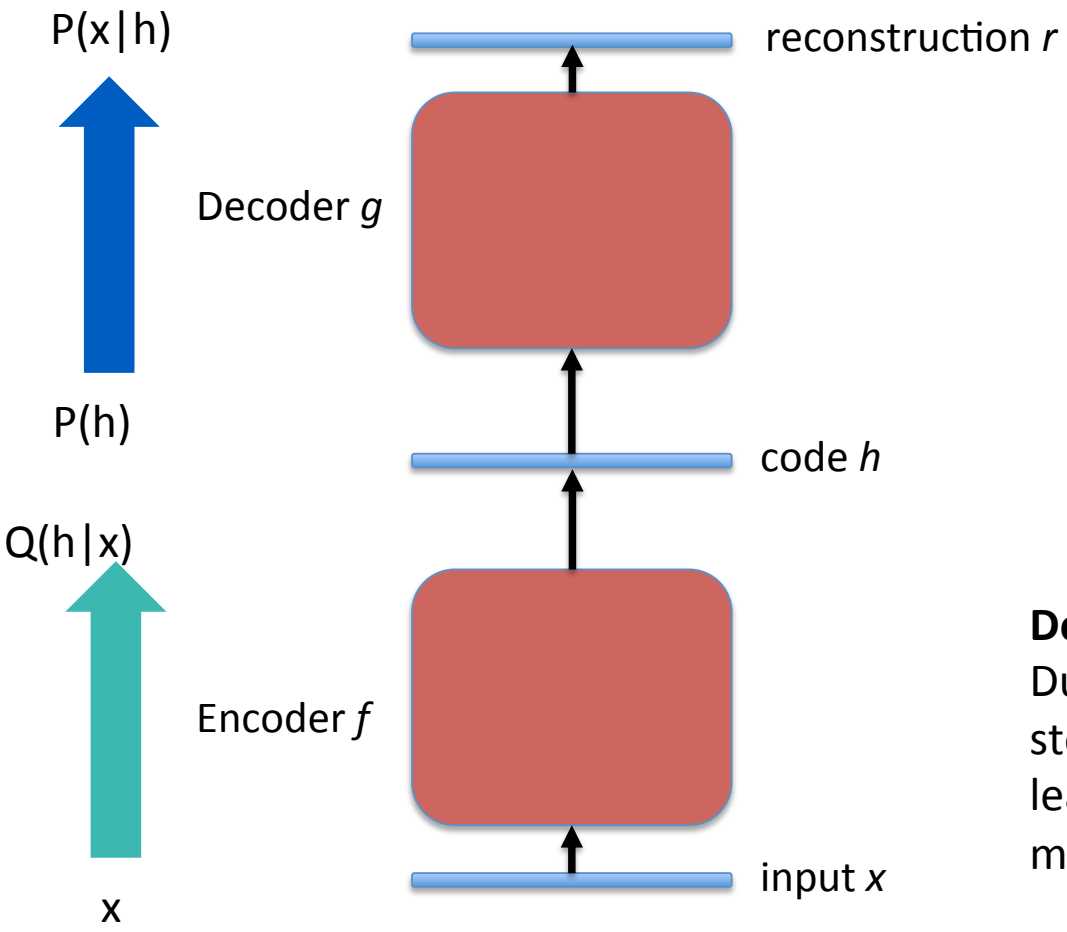
VAE-DRAW

Time ⟶

GAN



Auto-Encoders



Probabilistic criterion:

Reconstruction log-likelihood =

- log P(x | h)

Denoising auto-encoder:
During training, input is corrupted stochastically, and auto-encoder must learn to guess the distribution of the missing information.

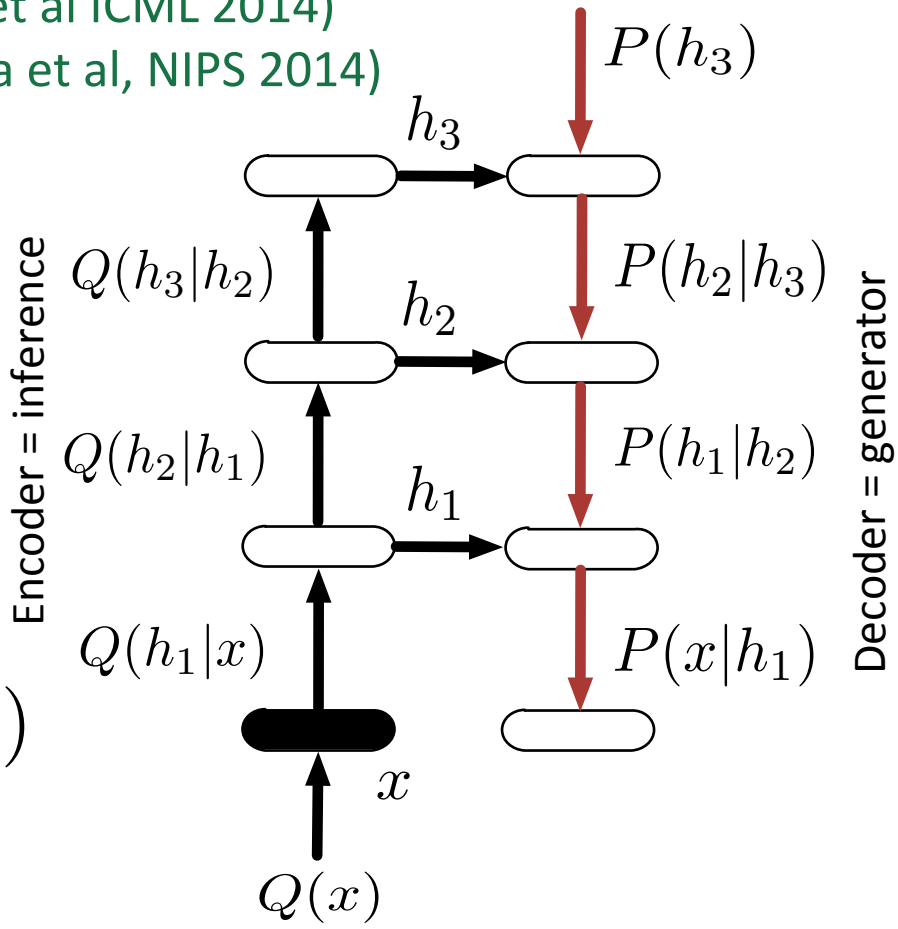
Variational Auto-Encoders (VAEs)

(Kingma & Welling 2013, ICLR 2014)
(Gregor et al ICML 2014; Rezende et al ICML 2014)
(Mnih & Gregor ICML 2014; Kingma et al, NIPS 2014)

- Parametric approximate inference
- Successor of Helmholtz machine (Hinton et al '95)
- Maximize variational lower bound on log-likelihood:

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

where $Q(x)$ = data distr.



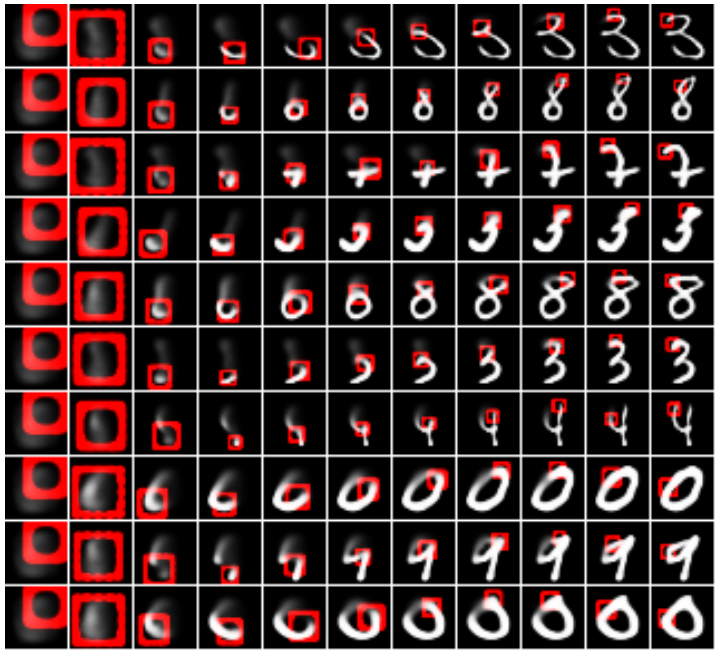
or equivalently

$$\max \sum_x Q(h|x) \log \frac{P(x,h)}{Q(h|x)} = \max \sum_x Q(h|x) \log P(x|h) + KL(Q(h|x)||P(h))$$

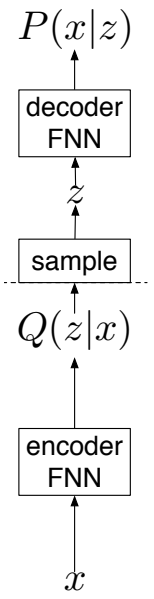
DRAW: Sequential Variational Auto-Encoder with Attention

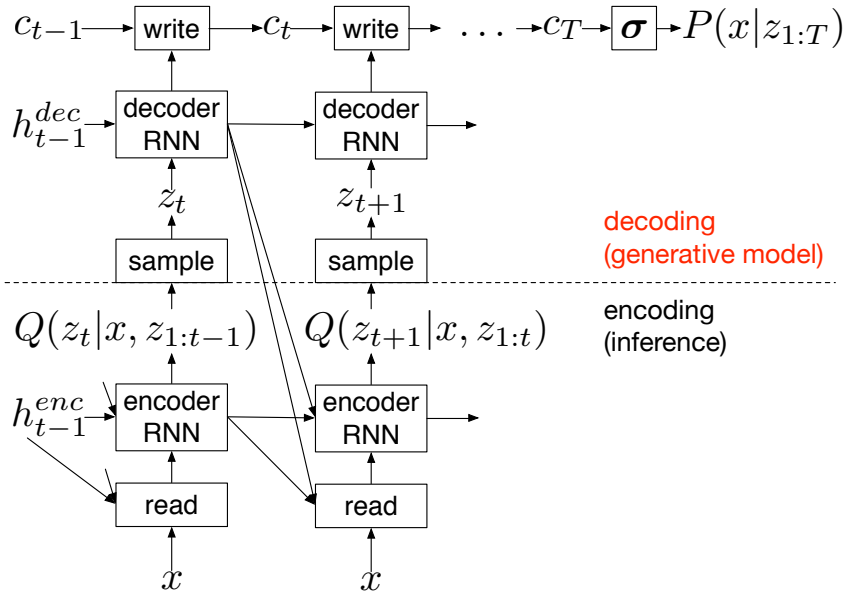
(Gregor et al of Google DeepMind, arXiv 1502.04623, 2015)

- 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.



Time ⟶



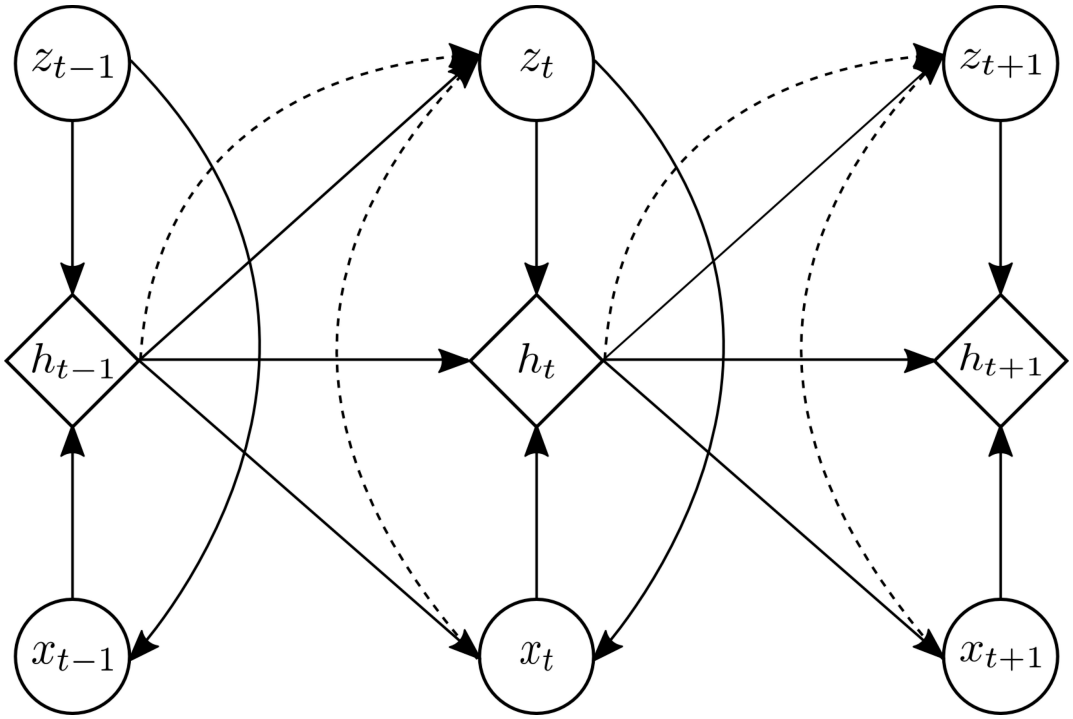


DRAW Samples of SVHN Images: generated samples vs training nearest neighbor

Nearest training example for last column of samples

Variational Generative RNNs

- *(Chung et al, NIPS'2015)*
- Regular RNNs have noise injected only in input space
- VRNNs also allow noise (latent variable) injected in top hidden layer; more « high-level » variability



Other Descendants of the Helmholtz Machine

- Reweighted Wake-Sleep *(Bornschein & Bengio ICLR 2015)*

$$p(\mathbf{x}) = \sum_{\mathbf{h}} q(\mathbf{h} \mid \mathbf{x}) \frac{p(\mathbf{x}, \mathbf{h})}{q(\mathbf{h} \mid \mathbf{x})} = \mathop{\mathbb{E}}_{\mathbf{h} \sim q(\mathbf{h} \mid \mathbf{x})} \left[\frac{p(\mathbf{x}, \mathbf{h})}{q(\mathbf{h} \mid \mathbf{x})} \right]$$

$$\simeq \frac{1}{K} \sum_{\substack{k=1 \\ \mathbf{h}^{(k)} \sim q(\mathbf{h} \mid \mathbf{x})}}^{K} \frac{p(\mathbf{x}, \mathbf{h}^{(k)})}{q(\mathbf{h}^{(k)} \mid \mathbf{x})}$$

Uses importance sampling approximations

- = Wake-Sleep with K=1, systematically works better with larger K, comparable to VAE performance

Results on binarized MNIST

Method	NLL bound	NLL est.
RWS (SBN/SBN 10-100-200-300-400)		85.48
RWS (NADE/NADE 250)		85.23
RWS (AR-SBN/SBN 500)†		84.18
NADE (500 units, [1])		88.35
EoNADE (2hl, 128 orderings, [2])		85.10
DARN (500 units, [3])		84.13
RBM (500 units, CD3, [4])	105.5	
RBM (500 units, CD25, [4])	86.34	
DBN (500-2000, [5])	86.22	84.55

Other Descendants of the Helmholtz Machine

- Training Bidirectional Helmholtz Machines, *(Bornschein et al. arXiv:1506.03877)*
- **Both encoder and decoder paths participate in the energy fn**

$$p^*(\mathbf{x}, \mathbf{h}_1, \mathbf{h}_2) = \frac{1}{Z}\sqrt{p(\mathbf{x}, \mathbf{h}_1, \mathbf{h}_2)\; q(\mathbf{x}, \mathbf{h}_1, \mathbf{h}_2)}$$

$$q(\mathbf{x}) = p^*(\mathbf{x}) = \sum_{\mathbf{h}_1, \mathbf{h}_2} p^*(\mathbf{x}, \mathbf{h}_1, \mathbf{h}_2)$$

$$= \frac{\sqrt{q(\mathbf{x})}}{Z} \sum_{\mathbf{h}_1, \mathbf{h}_2} \sqrt{p(\mathbf{x}, \mathbf{h}_1, \mathbf{h}_2)\; q(\mathbf{h}_1|\mathbf{x}) q(\mathbf{h}_2|\mathbf{h}_1)}$$

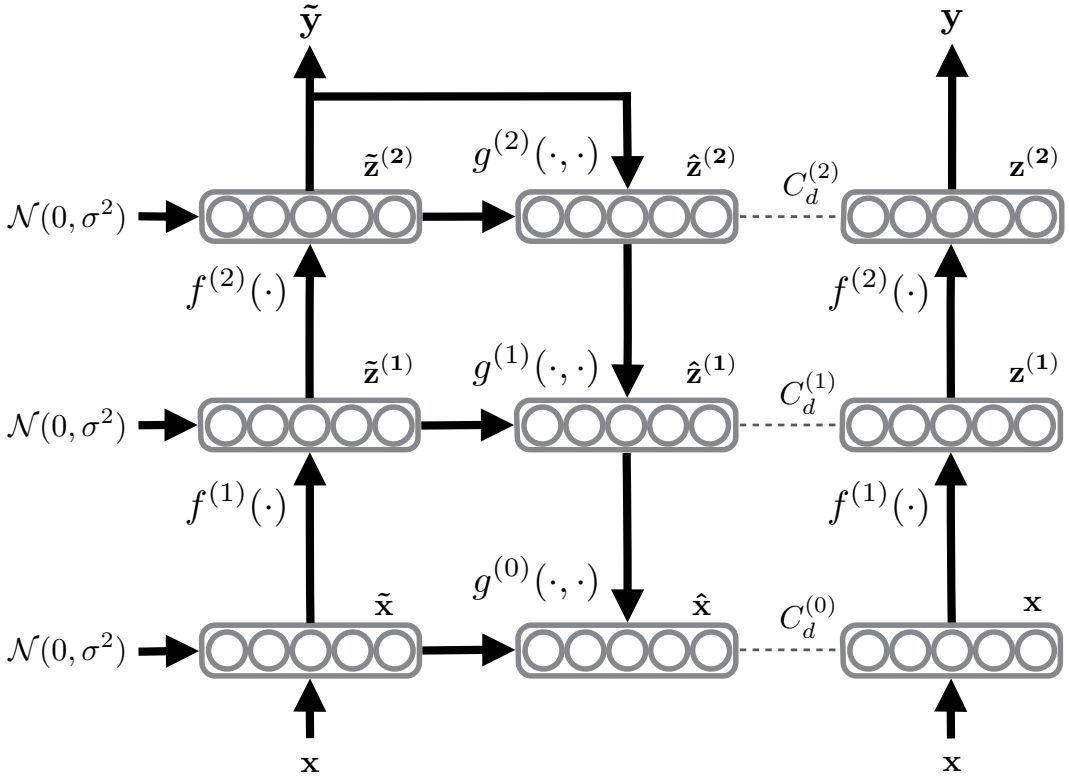
$$= \left(\frac{1}{Z} \sum_{\mathbf{h}_1, \mathbf{h}_2} \sqrt{p(x, \mathbf{h}_1, \mathbf{h}_2)\; q(\mathbf{h}_1|\mathbf{x}) q(\mathbf{h}_2|\mathbf{h}_1)} \right)^2$$

A lower bound can be maximized

Encouraging News: Semisupervised Learning with Ladder Network

(Rasmus et al, NIPS'2015)

- Jointly trained stack of denoising auto-encoders with gated lateral connections and semi-supervised objective



Semi-supervised objective:

$$-\log P(\tilde{\mathbf{y}} = t(n) \mid \mathbf{x}) + \sum_{l=1}^{L} \lambda_l \left\| \mathbf{z}^{(l)} - \hat{\mathbf{z}}_{\mathrm{BN}}^{(l)} \right\|^2$$

They also use Batch Normalization

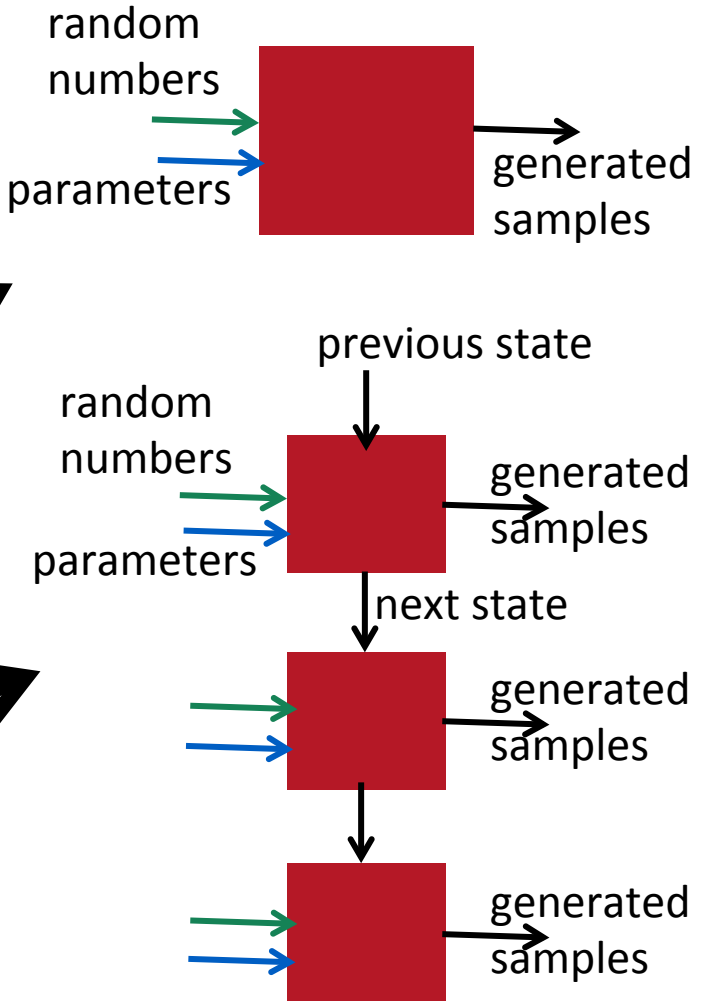
New records:
1% error with 100 labeled examples
.6% with 60000

Bypassing Normalization Constants with Generative Black Boxes

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

- (Goodfellow et al, NIPS 2014, GAN = 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; Sohl-Dickstein et al ICML 2015), learning the transition operator of a Markov chain that generates the data.

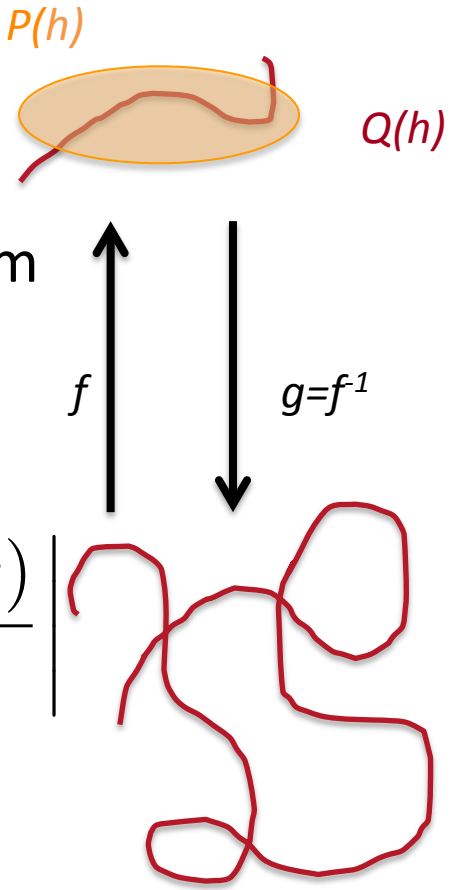


NICE

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**

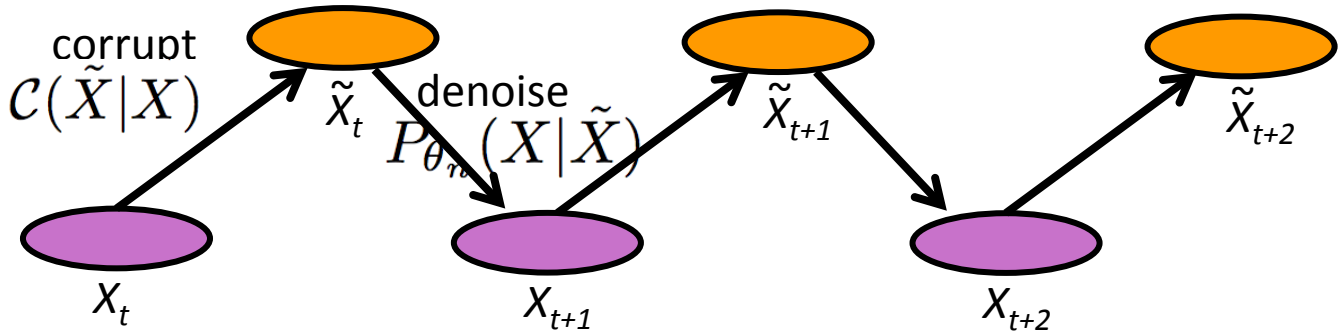


$$\log p_X(x) = \log p_H(f(x)) + \log \left| \det \frac{\partial f(x)}{\partial x} \right|$$

Factorized prior

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

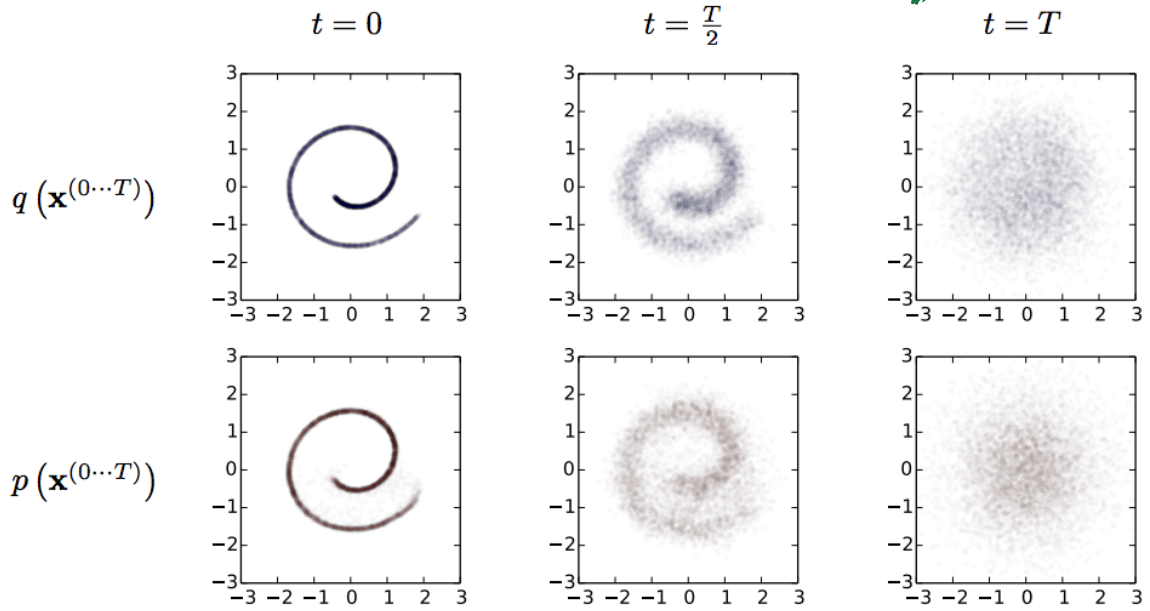
Denoising Auto-Encoder Markov Chain

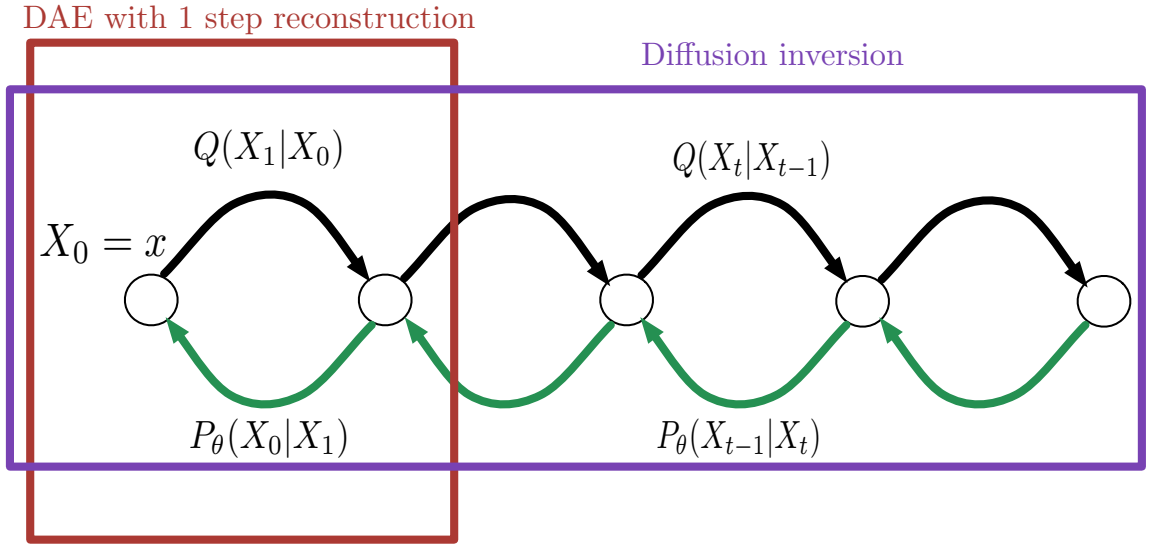


- NIPS'2013: Denoising AE are consistent estimators of the data-generating distribution through their Markov chain, so long as they consistently estimate the conditional denoising distribution and the Markov chain converges.

Denoising Auto-Encoder vs Diffusion Inverter (Sohl-Dickstein et al ICML 2015)

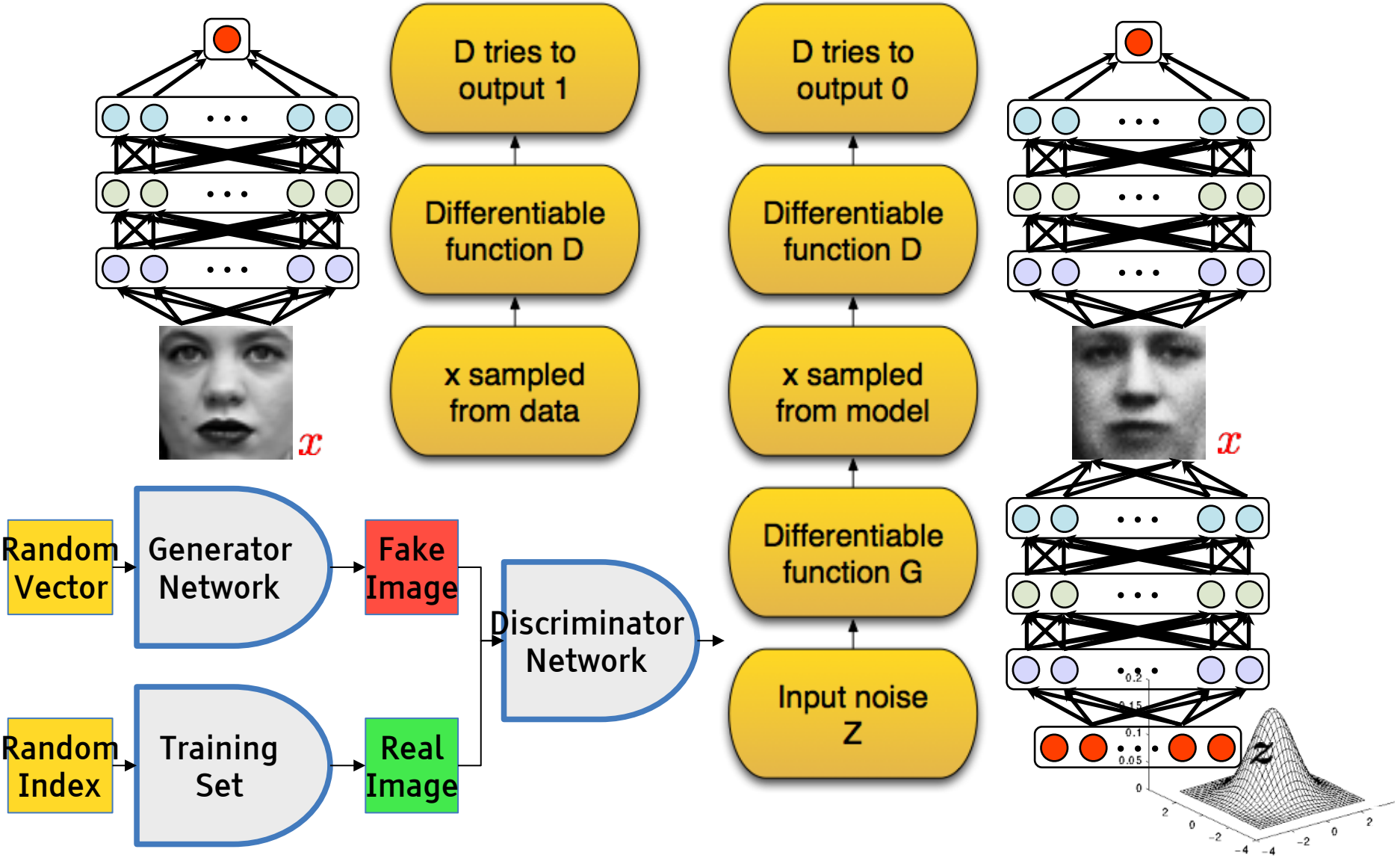
- DAE: after 1 step of diffusion (adding noise, Q), try to reconstruct the clean original (with P).
- Diffusion inverter: after each step of diffusion, try to stochastically undo the effect of diffusion.





GAN: Generative Adversarial Networks

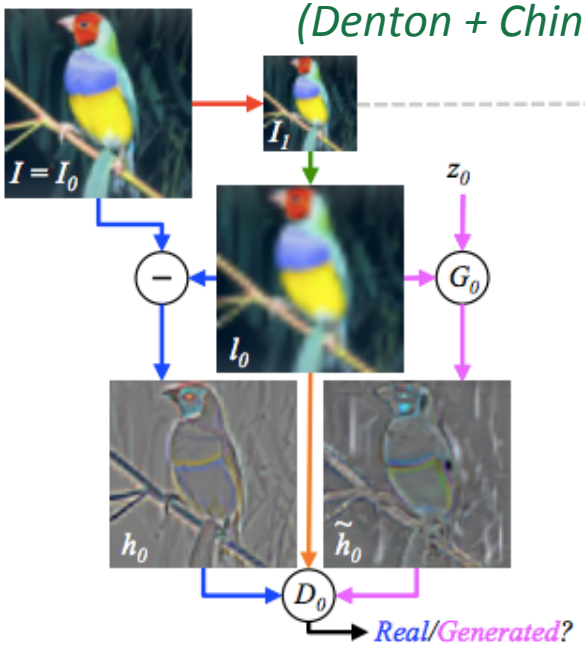
Goodfellow et al NIPS 2014

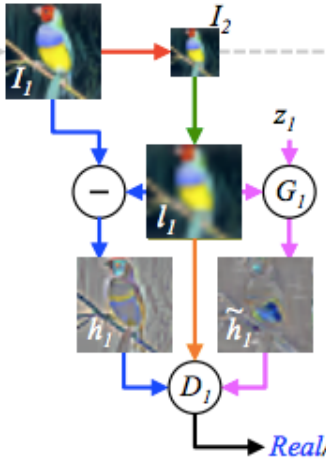


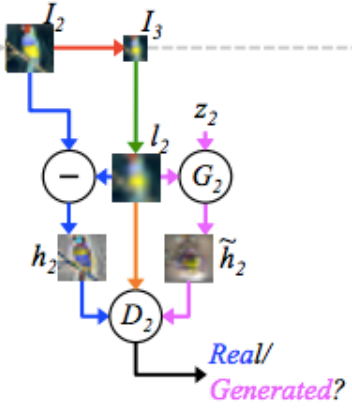
LAPGAN: Laplacian Pyramid of Generative Adversarial Networks

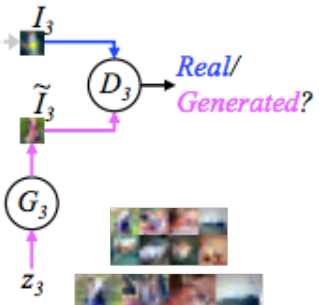
(Denton + Chintala, et al 2015)

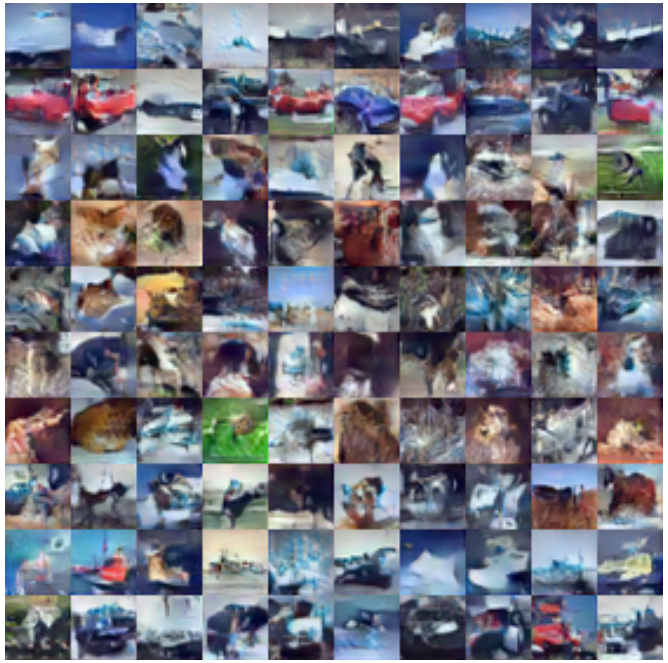
http://soumith.ch/eyescream/

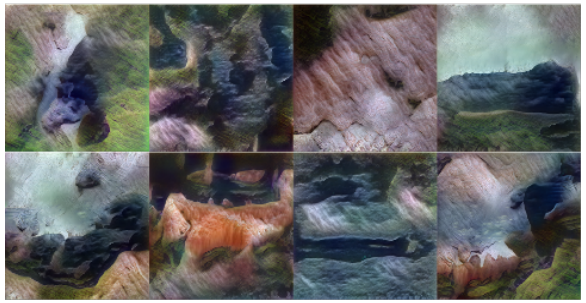








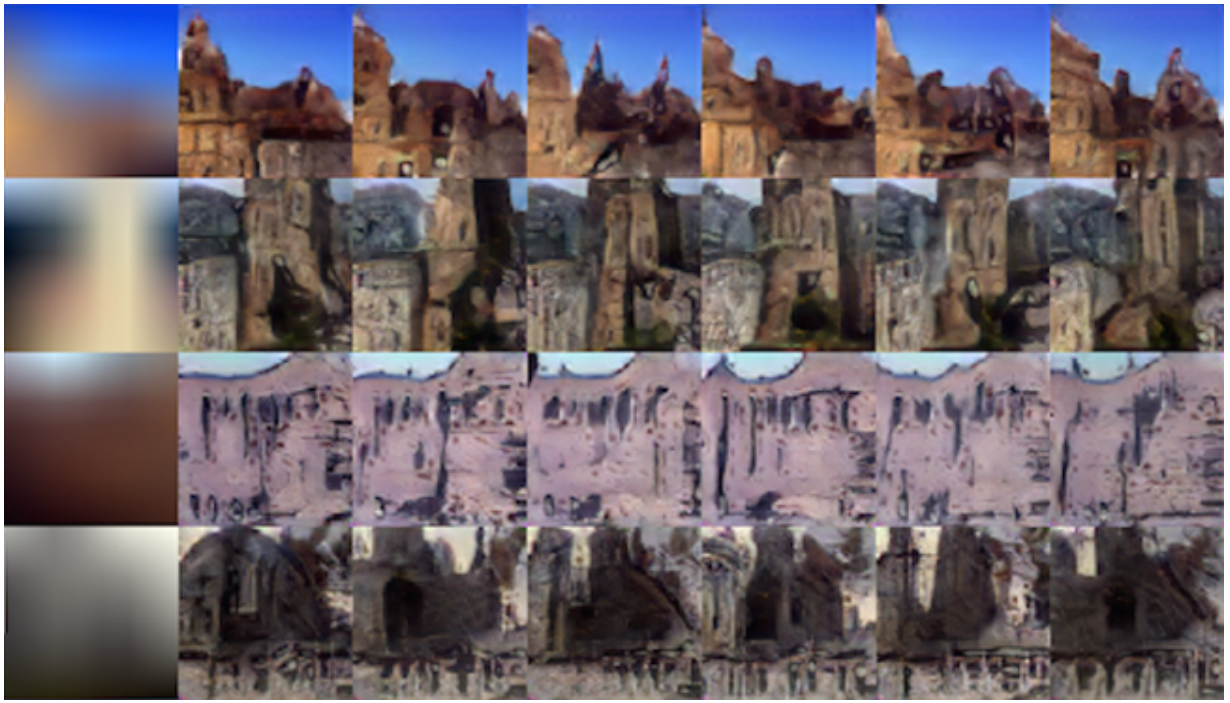




LAPGAN: Visual Turing Test

(Denton + Chintala, et al 2015)

- 40% of samples mistaken *by humans* for real photos



- Sharper images than max. lik. proxys (which min. KL(data|model)):
- GAN objective = compromise between KL(data|model) and KL(model|data)

Convolutional GANs

(Radford et al, arXiv 1511.06343)

Strided convolutions, batch normalization, only convolutional layers, ReLU and leaky ReLU

Conclusions

- Although unsupervised learning is not yet in industrial applications, it is a key for future large scale ones, to build machines that incorporate knowledge from large quantities of unlabeled data.

- The field is vibrant with many different approaches, based on many alternative learning principles.

- It may even help us connect deep learning with brain learning…

MILA: Montreal Institute for Learning Algorithms