Deep Generative Models

DLSS 2015

Deep Learning Summer School
Montreal, Canada

Yoshua Bengio

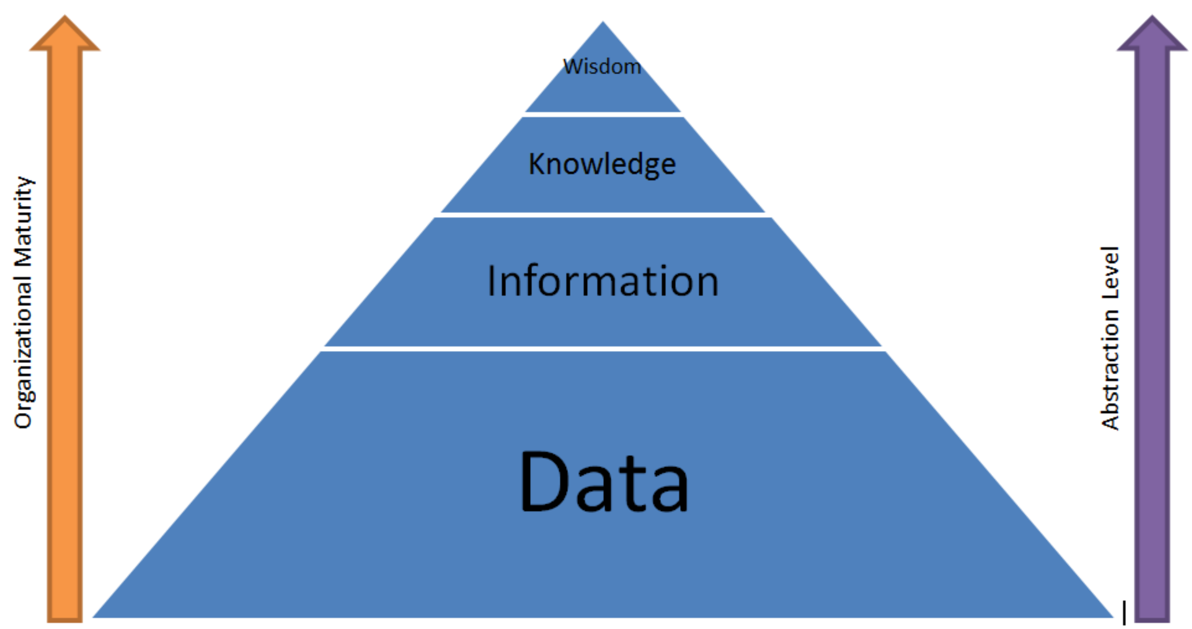
August 12, 2015

Learning « How the world ticks »

- So long as our machine learning models « cheat » by relying only on surface statistical regularities, they remain vulnerable to out-of-distribution examples
- Humans generalize better than other animals by implicitly having a more accurate internal model of the underlying causal relationships
- This allows one to predict future situations (e.g., the effect of planned actions) that are far from anything seen before, an essential component of reasoning, intelligence and science

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



Invariance and Disentangling

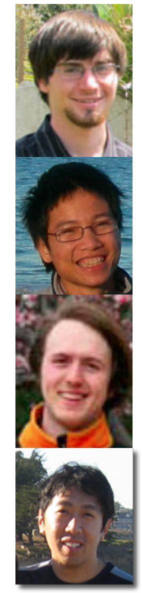
- Invariant features

- 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?

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.**

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

- Alternatives?

Log-Likelihood Gradient in Undirected Graphical Models (e.g. Boltzmann Machine)

$$P(x) = \frac{1}{Z} \sum_h e^{-\text{Energy}(x,h)} = \frac{1}{Z} e^{-\text{FreeEnergy}(x)}$$

- Gradient has two components:

"positive phase" "negative phase"

$$\frac{\partial \log P(x)}{\partial \theta} = \boxed{-\frac{\partial \text{FreeEnergy}(x)}{\partial \theta}} + \boxed{\sum_{\tilde{x}} P(\tilde{x}) \frac{\partial \text{FreeEnergy}(\tilde{x})}{\partial \theta}}$$

$$= \boxed{-\sum_h P(h|x) \frac{\partial \text{Energy}(x,h)}{\partial \theta}} + \boxed{\sum_{\tilde{x},\tilde{h}} P(\tilde{x},\tilde{h}) \frac{\partial \text{Energy}(\tilde{x},\tilde{h})}{\partial \theta}}$$

- Difficult part: sampling from P(*x*) or P(*x*,*h*), typically with a Markov chain

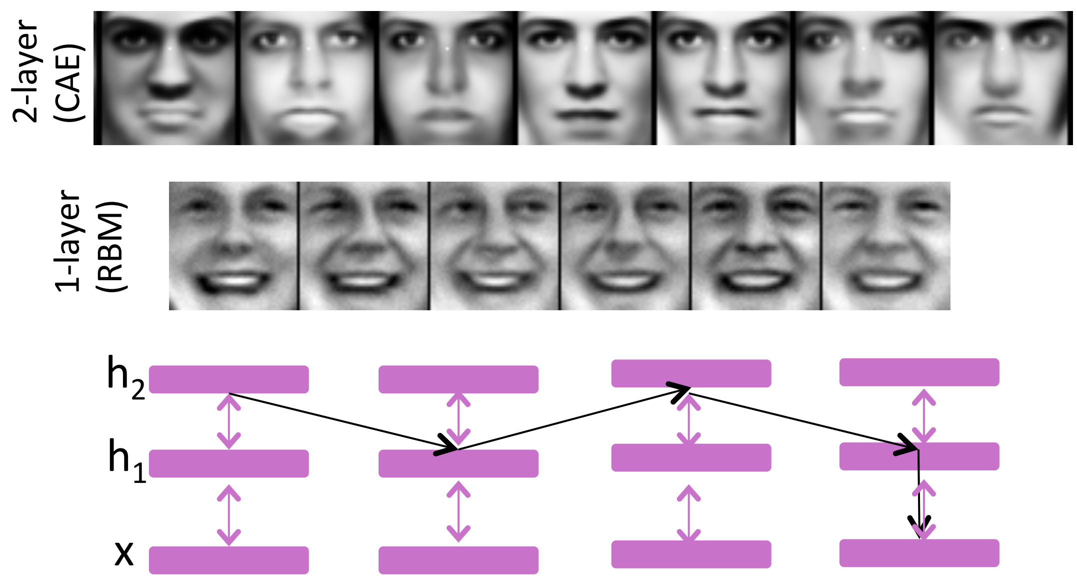
Issues with Maximum Likelihood for Boltzmann Machines

- Sampling from an MCMC of the model is required in the inner loop of training (for each example)
- As the model gets sharper, mixing between well-separated modes stalls, yielding a poor estimate of the gradient

Poor Mixing: Depth to the Rescue

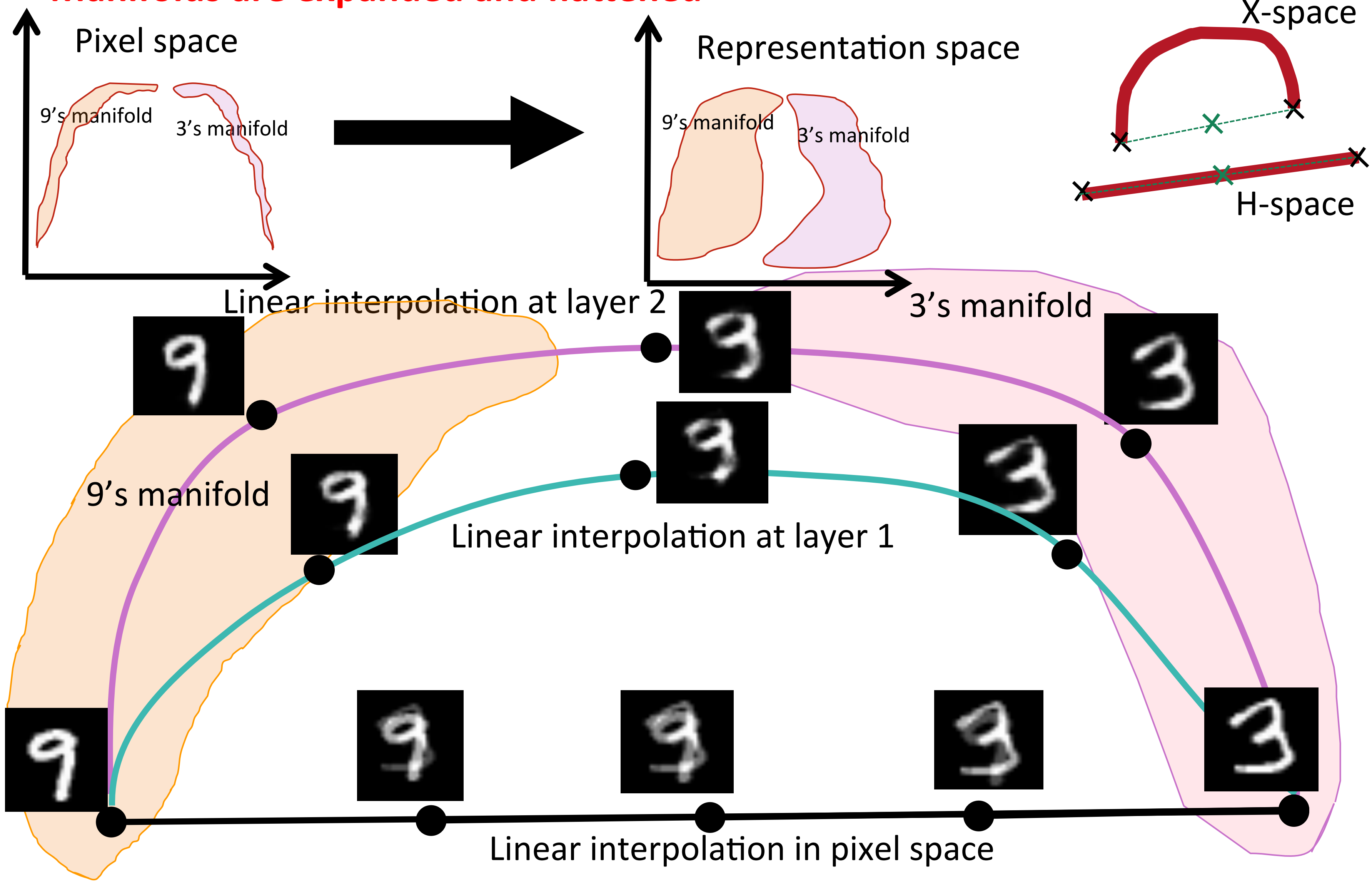
(Bengio et al ICML 2013)

- Sampling from DBNs and stacked Contractive Auto-Encoders:
  1. MCMC sampling from top layer model
  2. Propagate top-level representations to input-level repr.
- Deeper nets visit more modes (classes) faster

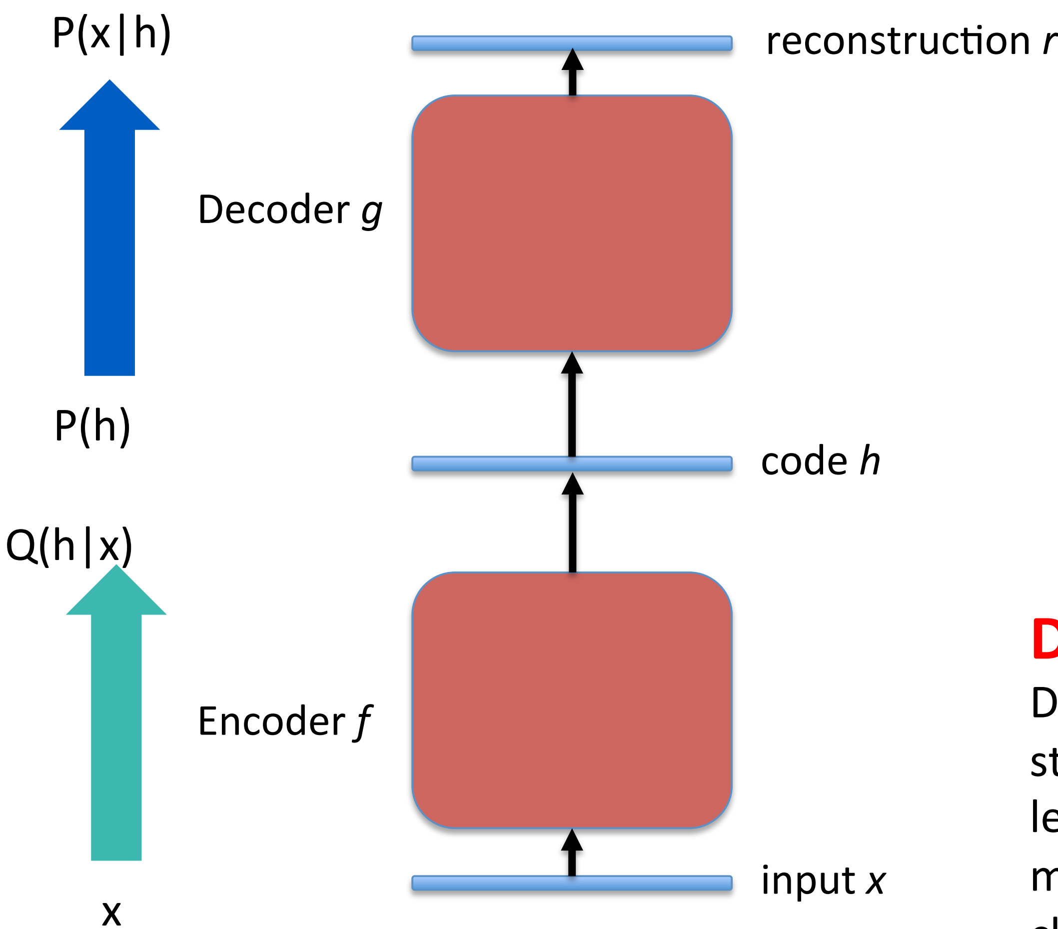


Space-Filling in Representation-Space

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



Auto-Encoders



Probabilistic criterion:

Reconstruction log-likelihood =

$- \log P(x \mid h)$

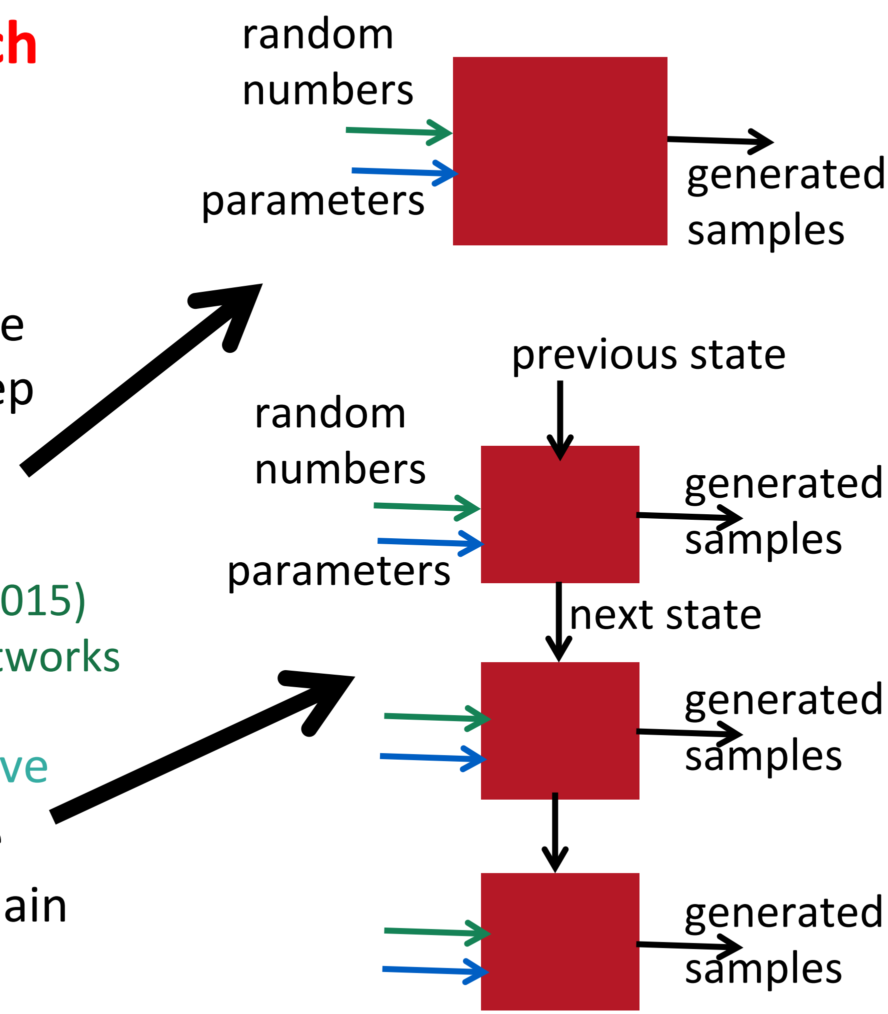
Denoising auto-encoder:
During training, input is corrupted stochastically, and auto-encoder must learn to guess the distribution of the missing information (reconstruct the clean original input)

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.
    - Also: (Li, Swersky & Zemel arXiv 2015) Generative moment matching networks

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



Score Matching (Hyvarinen 2005)

- Score of model *p*: d log p(x)/dx   does not contain partition fn Z
- Matching score of *p* to target score:

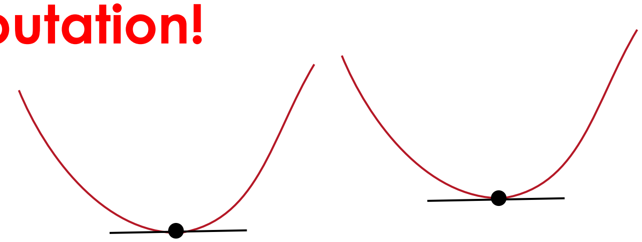
?

$$\mathbb{E}_{q(\mathbf{x})}\left[\frac{1}{2}\left\|\frac{\partial \log p(\mathbf{x})}{\partial \mathbf{x}} - \frac{\partial \log q(\mathbf{x})}{\partial \mathbf{x}}\right\|^2\right]$$

- Hyvarinen shows it equals

$$\mathbb{E}_{q(\mathbf{x})}\left[\frac{1}{2}\left\|\frac{\partial \log p(\mathbf{x})}{\partial \mathbf{x}}\right\|^2 + \sum_i \frac{\partial^2 \log p(\mathbf{x})}{\partial \mathbf{x_i^2}}\right] + const$$

- and proposes to minimize corresponding empirical mean
- Shown to be asymptotically unbiased to estimate parameters
- **Requires O(#parameters x #inputs) computation!**



Denoising Auto-Encoder

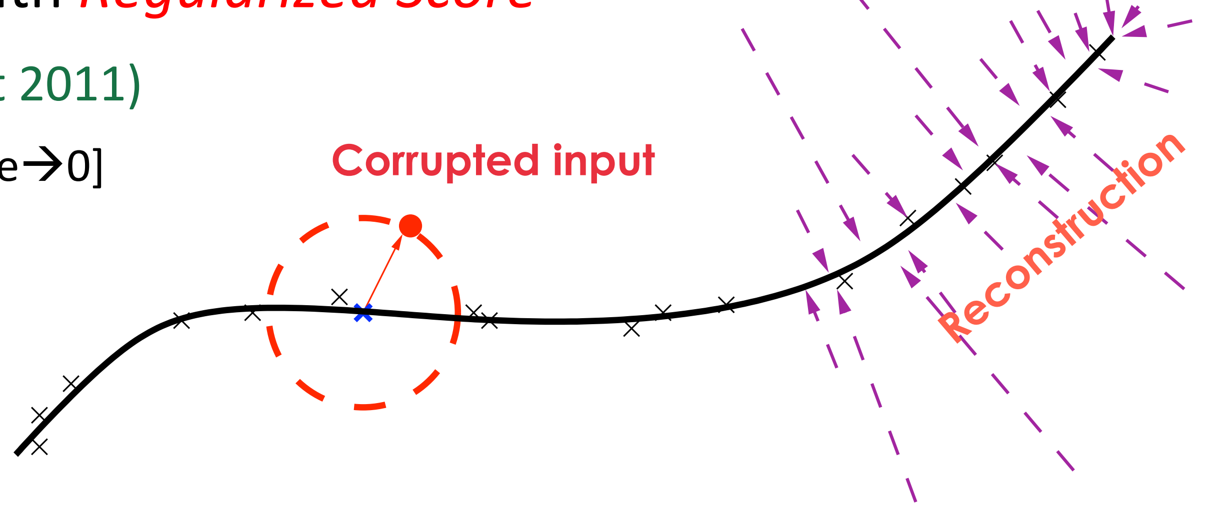
- Learns a vector field pointing towards higher probability direction (Alain & Bengio 2013)

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

- 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"



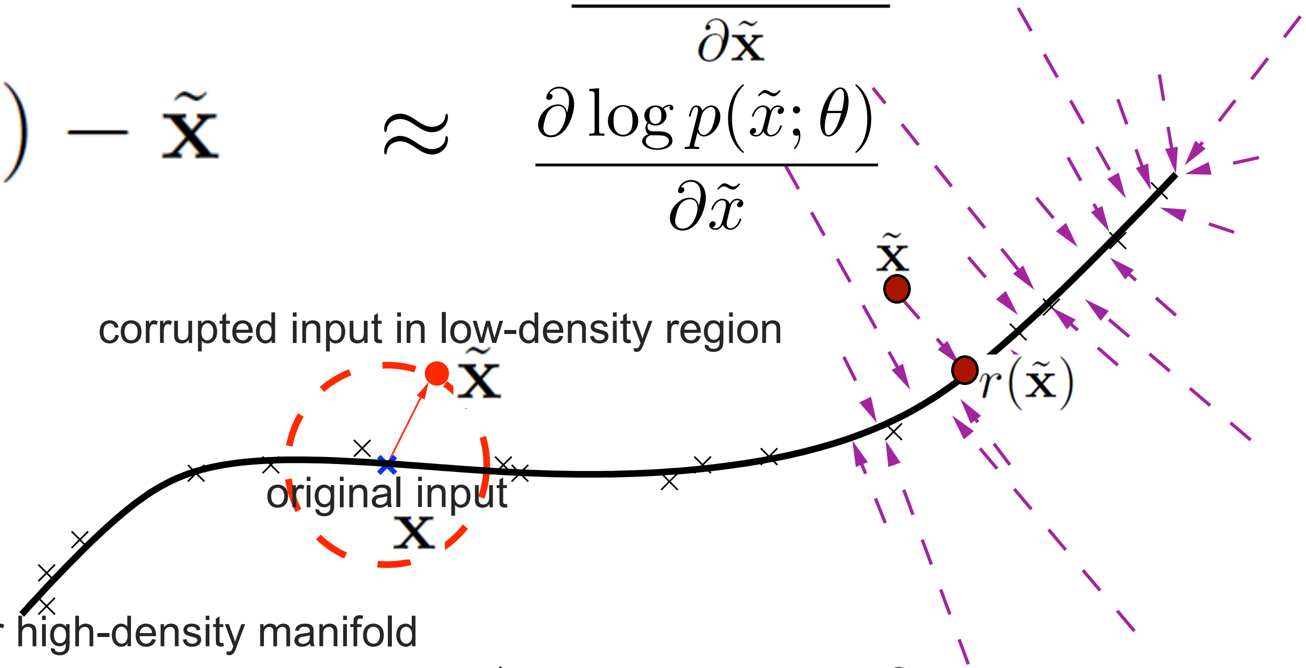
Denoising Auto-Encoders doing Score Matching on Gaussian RBMs

(Vincent 2011)

- clean input - corrupted input = direction of increasing log-likelihood

$$\mathbf{x} - \tilde{\mathbf{x}} \approx \frac{\partial \log q_\sigma(\tilde{\mathbf{x}}|\mathbf{x})}{\partial \tilde{\mathbf{x}}}$$

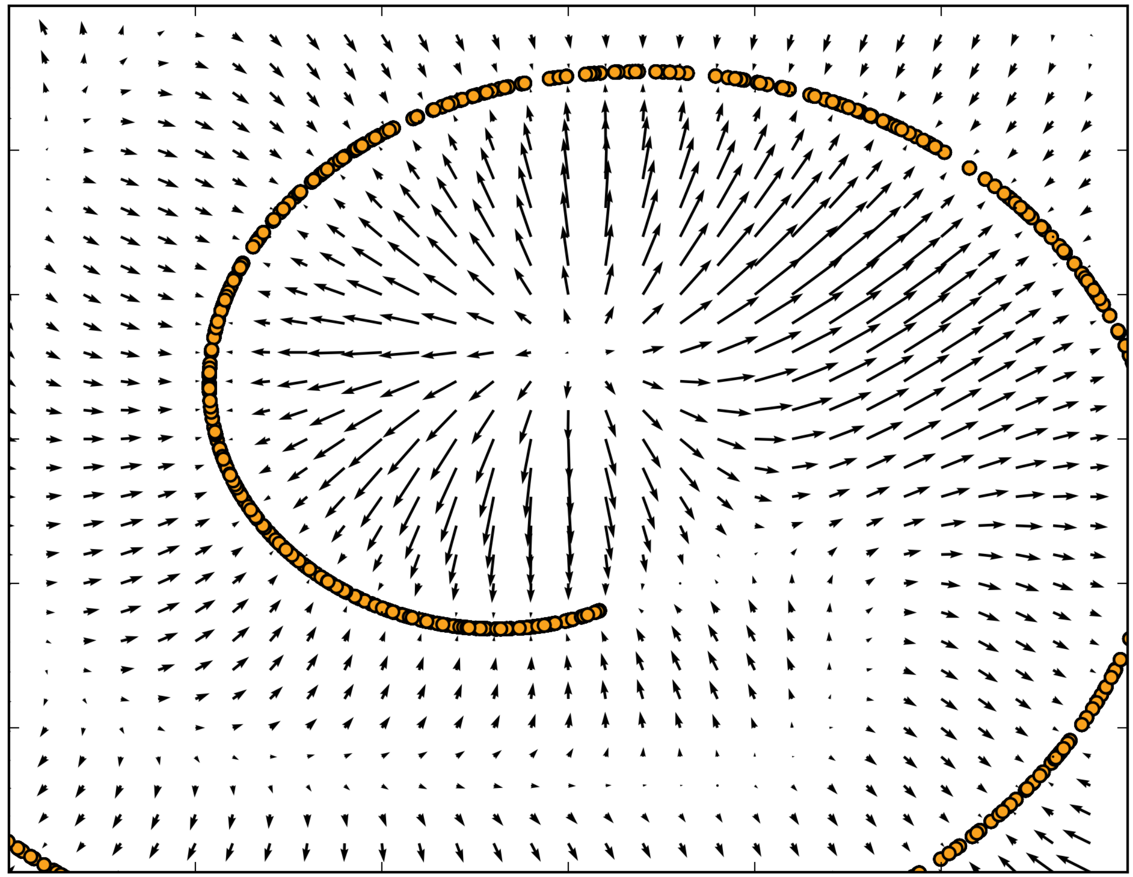
- $$r(\tilde{\mathbf{x}}) - \tilde{\mathbf{x}} \approx \frac{\partial \log p(\tilde{x};\theta)}{\partial \tilde{x}}$$



- Denoising error = $\|(r(\tilde{\mathbf{x}}) - \cancel{\tilde{\mathbf{x}}}) - (\mathbf{x} - \cancel{\tilde{\mathbf{x}}})\|^2 = \|r(\tilde{\mathbf{x}}) - \mathbf{x}\|^2$

Learning a Vector Field that Estimates a Gradient Field (Alain & Bengio ICLR 2013)

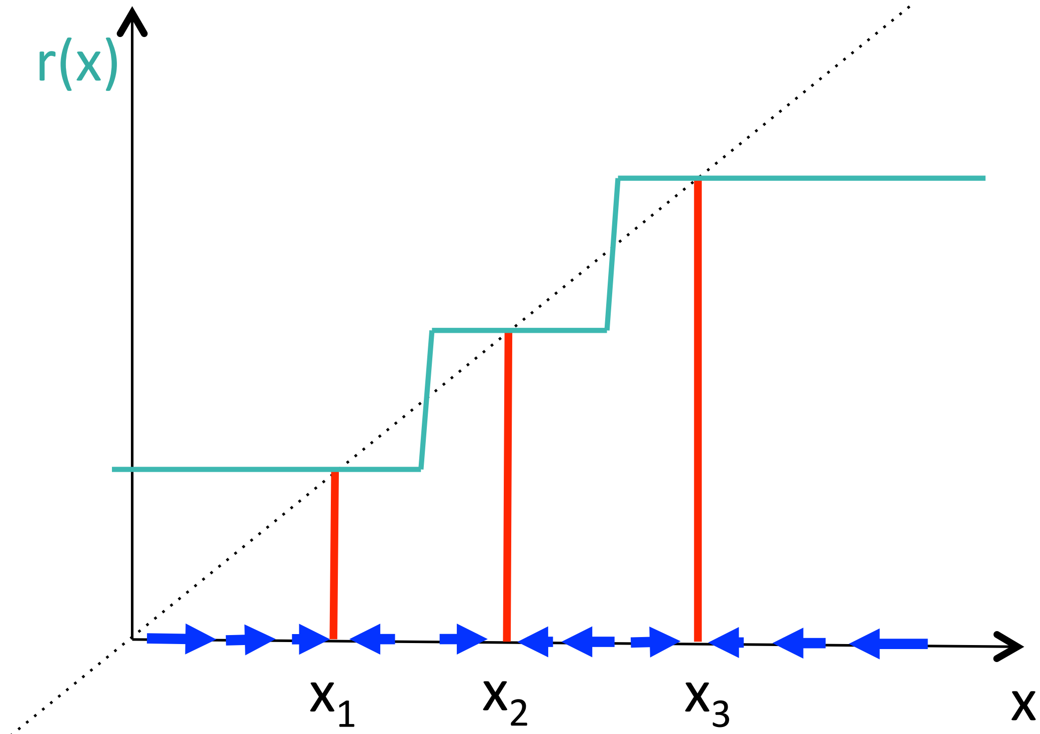
- **Reconstruction(x)-x estimates dlogp(x)/dx**
- A regularized form of *score matching* (Vincent 2011)
- Generalized to arbitrary corruption, r-v type & reconstruction log-lik. Bengio et al NIPS'2013



Continuous x, Gaussian noise, squared error

Preference for Locally Constant Features

- Denoising or contractive auto-encoder on 1-D input:



$$E[||r(x+\sigma z)-x||^2] \approx E[||r(x)-x||^2] + \sigma^2 ||\frac{\partial r(x)}{\partial x}||_F^2$$

Denoising Score Matching

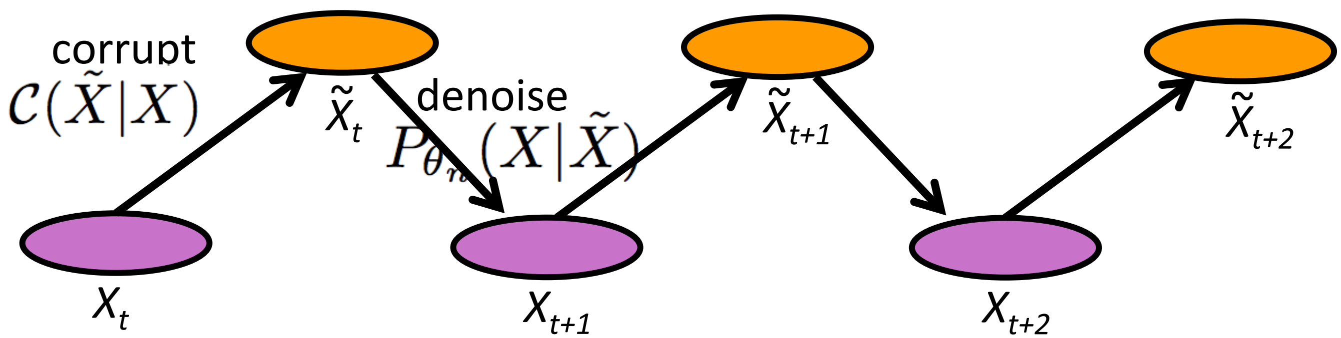
- An alternative to maximum likelihood for continuous random variables
- Asymptotically consistent estimator (as noises level decreases and # examples increases)
- Reconstruction: $$r(x) = x - \sigma^2 \frac{\partial Energy(x)}{\partial x}$$
- Denoising training objective, with N(0,1) noise *z*:

$$E_{x,z}[||r(x + \sigma z) - x||^2]$$

→ No partition function gradient!

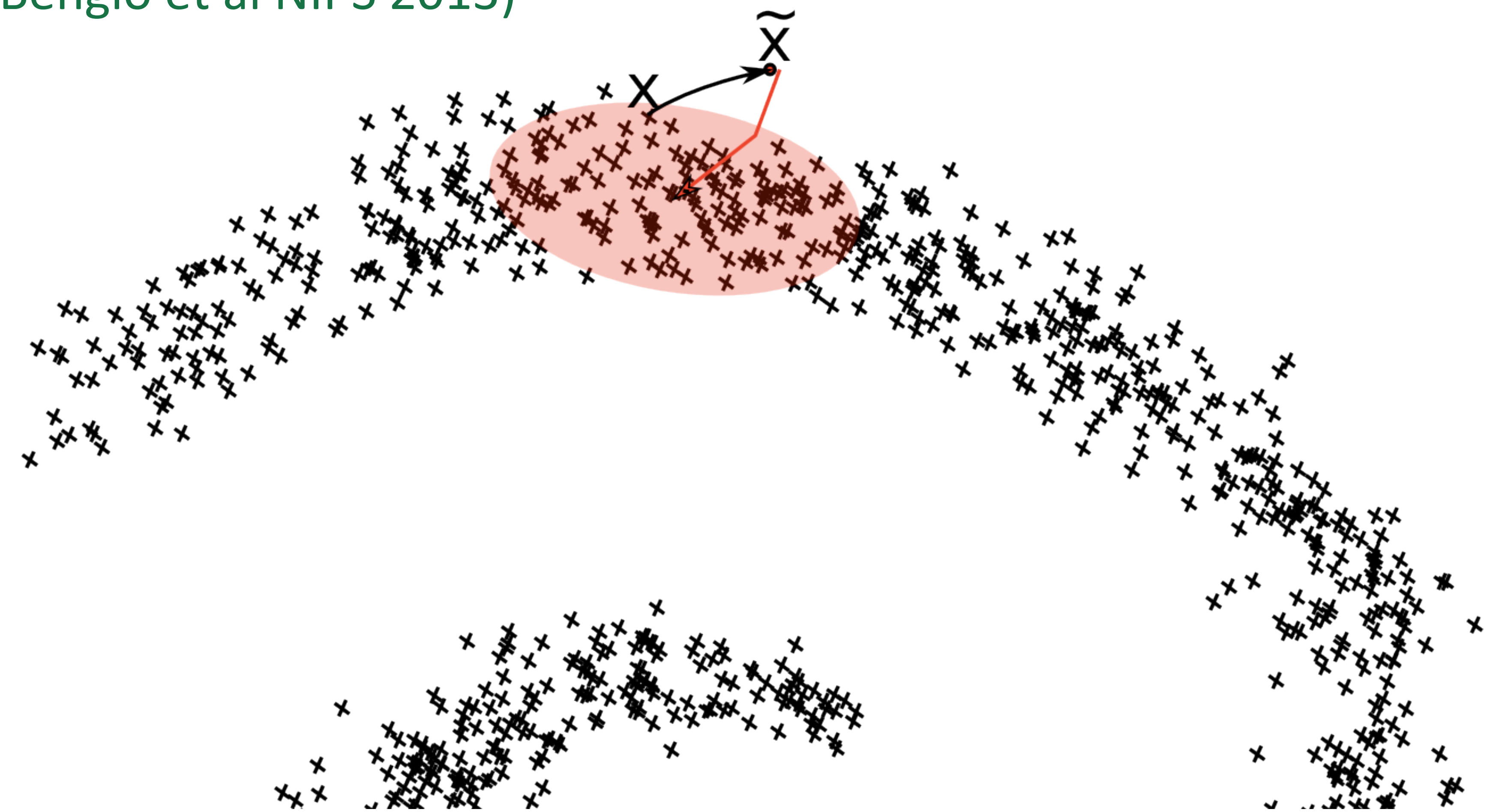
Denoising Auto-Encoder Markov Chain

Each Markov chain step = corrupt / encode / decode / sample



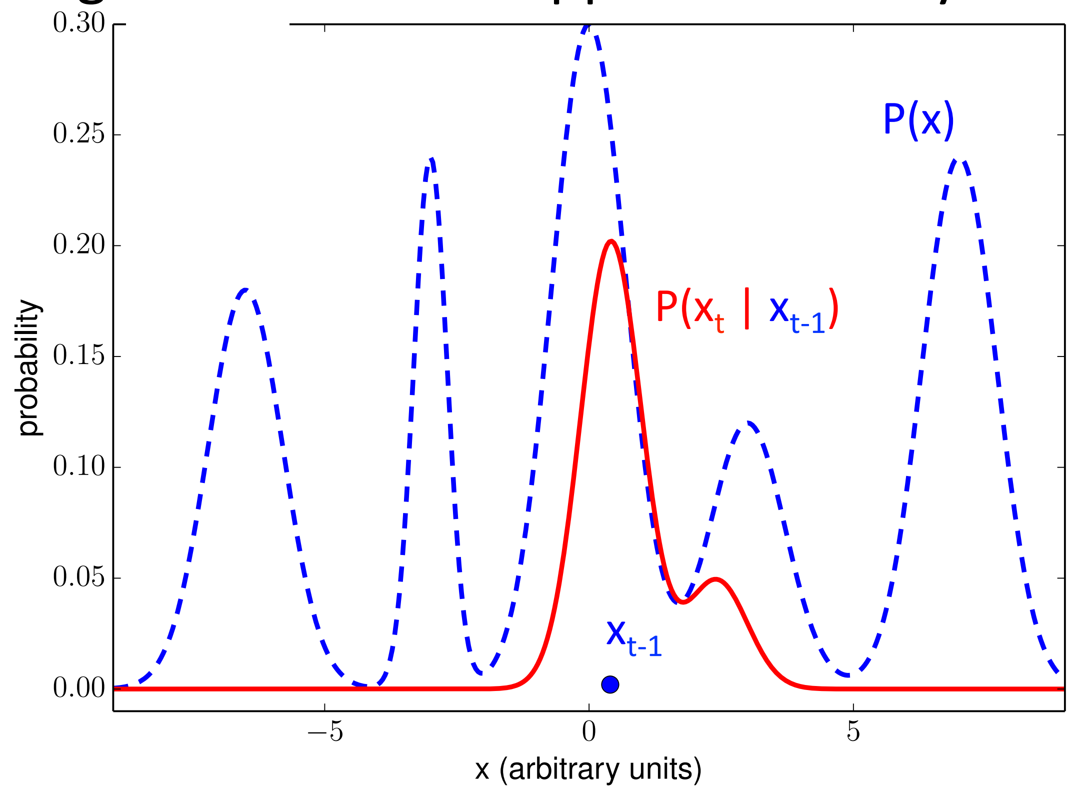
Denoising Auto-Encoders Learn a Markov Chain Transition Distribution

(Bengio et al NIPS 2013)



Many Modes Challenge: Instead of learning P(x) directly, learn Markov chain operator $P(x_t \mid x_{t-1})$

- P(x) may have many modes, making the normalization constant intractable, and MCMC approximations poor
- Partition fn of $P(x_t \mid x_{t-1})$ much simpler because most of the time a local move, might even be well approximated by unimodal



Consistency Results (Bengio et al 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.

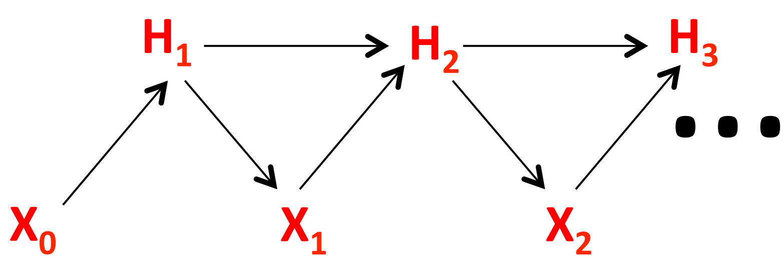
$$\text{Making } P_{\theta_n}(X|\tilde{X}) \text{ match } \mathcal{P}(X|\tilde{X}) \text{ makes } \pi_n(X) \text{ match } \mathcal{P}(X)$$

denoising distr. truth stationary distr. truth

Generative Stochastic Networks

- Generalizes the denoising auto-encoder training scheme
  - Introduce latent variables in the Markov chain (over X,H)
  - Instead of a fixed corruption process, have a deterministic function with parameters θ_1 and a noise source Z as input

$$H_{t+1} = f_{\theta_1}(X_t, Z_t, H_t)$$



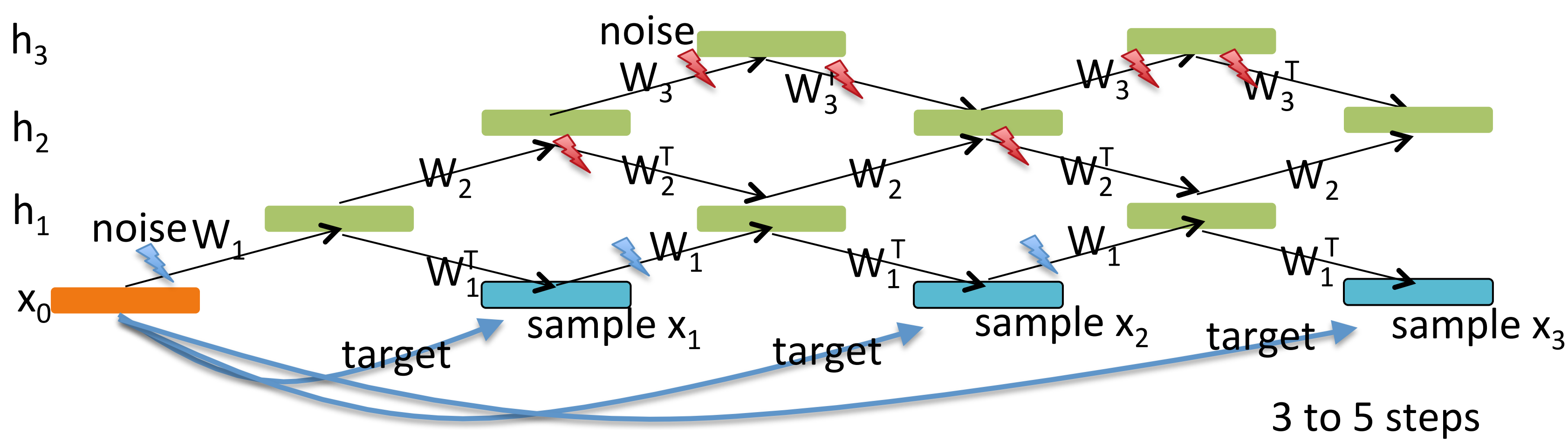
$$\begin{aligned} H_{t+1} &\sim P_{\theta_1}(H \mid H_t, X_t) \\ X_{t+1} &\sim P_{\theta_2}(X \mid H_{t+1}) \end{aligned}$$

  - DAE special case of GSN, both generate a Markov chain whose stationary distribution is a consistent estimator of the data generating distribution *(Bengio et al, NIPS'2013; ICML'2014)*

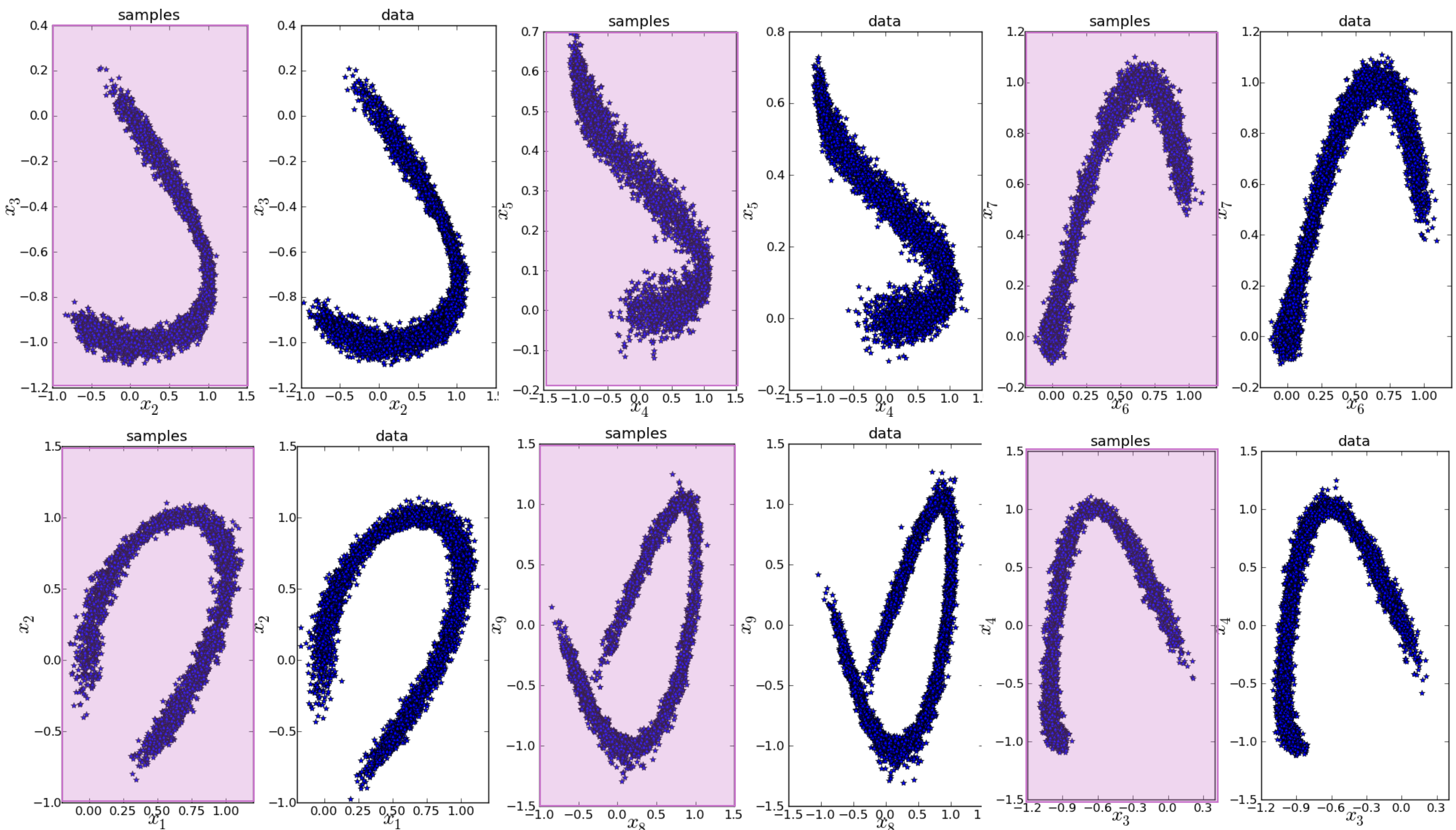
Generative Stochastic Networks (GSN)

(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:

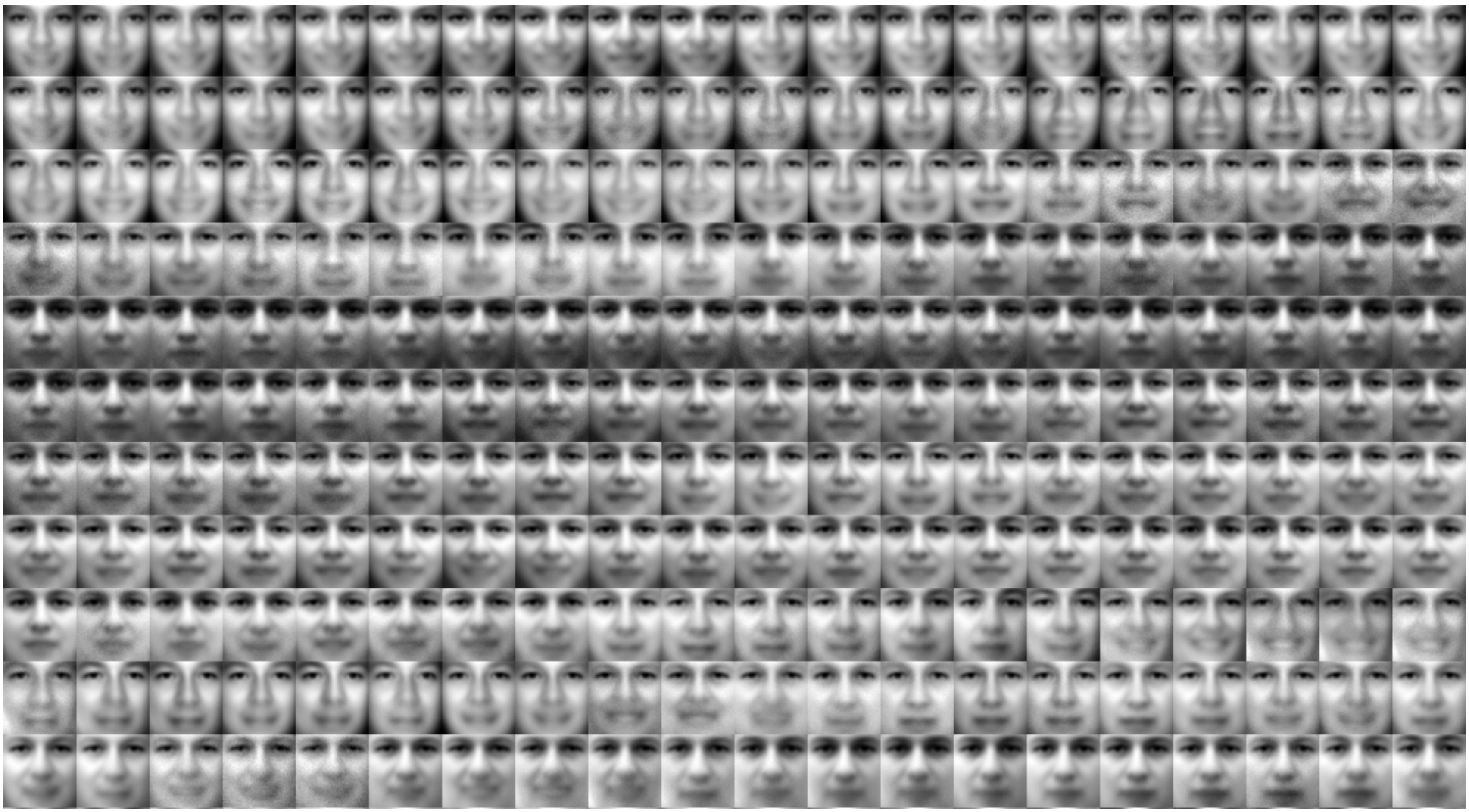


GSN Experiments: validating the theorem in a continuous non-parametric setting



Not Just MNIST: experiments on TFD

- 3 hidden layer model, consecutive samples:

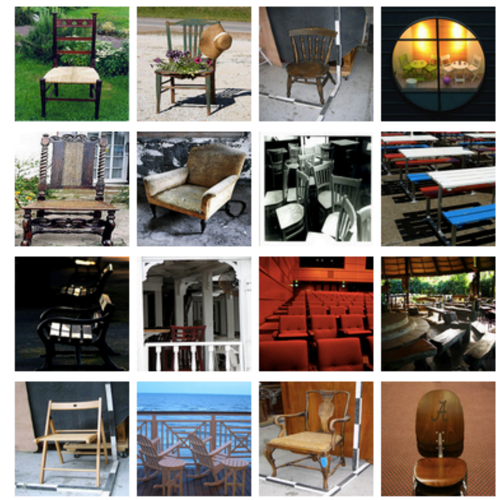


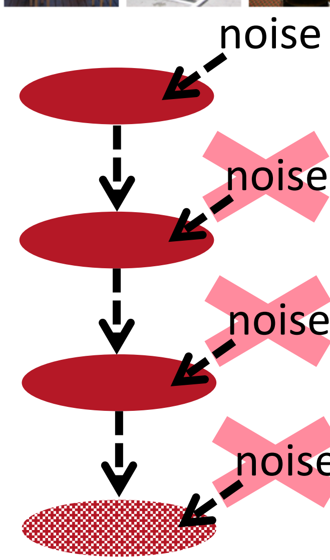
GSNs/DAEs can model complex distributions and missing modalities, but like DBNs and DBMs they add a lot of unnecessary noise in lower levels

- Injecting iid noise in lower levels: ugly white noise showing up in generated images, unless the lower layers are deterministic (poor mixing)

  ~~DBN~~ ~~DBM~~

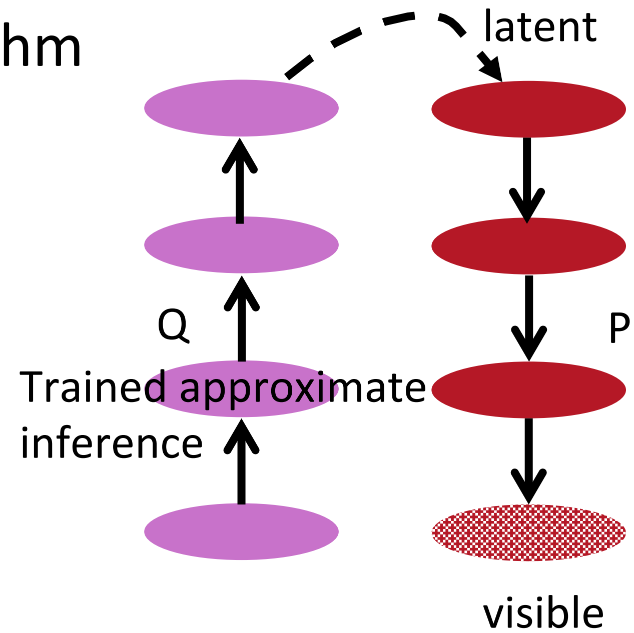
- Most factors of interest have highly non-linear relationship to pixel-level → must be generated at top-level and then transformed deterministically to pixel level: otherwise → blurred P(x)





Ancestral Sampling with Learned Approximate Inference: Replace Intractable P(h|x) by Learned Q(h|x)

- Helmholtz machine & Wake-Sleep algorithm
  - (Hinton, Dayan, Frey, Neal, 1995;
    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)

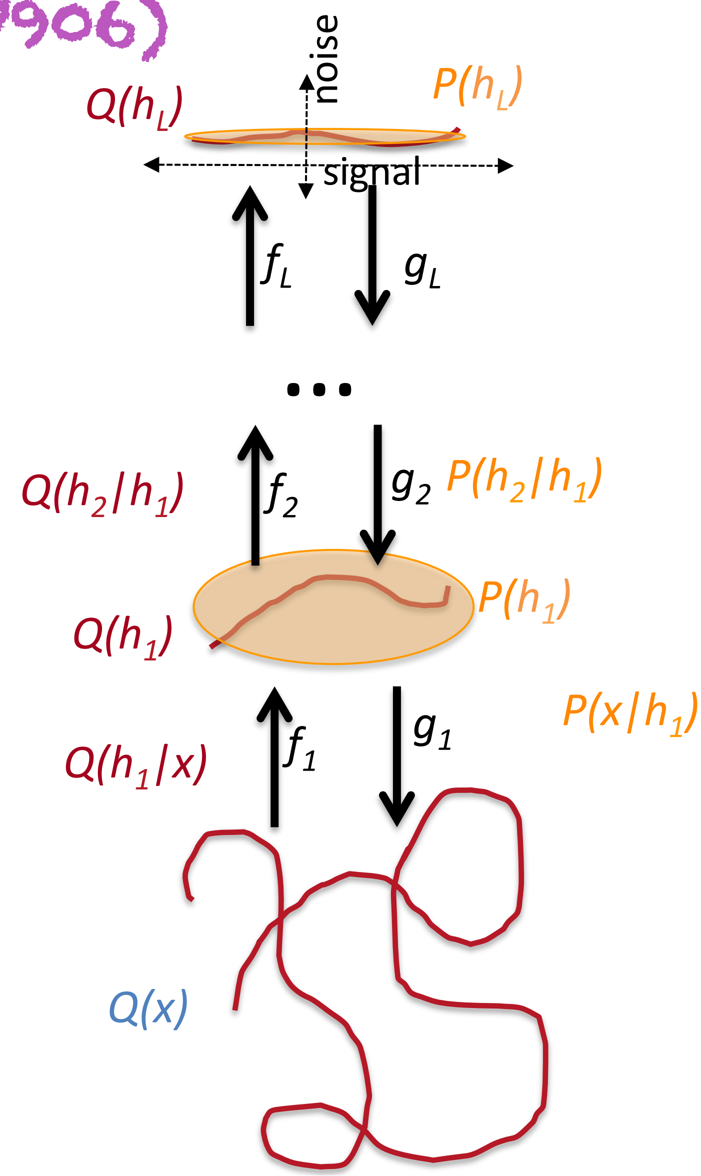


- Reweighted Wake-Sleep (Bornschein & Bengio 2014, ICML2015)
- Target Propagation (Bengio 2014)
- Deep Directed Generative Auto-Encoders (Ozair & Bengio 2014)
- NICE (Dinh et al 2014)

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.

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

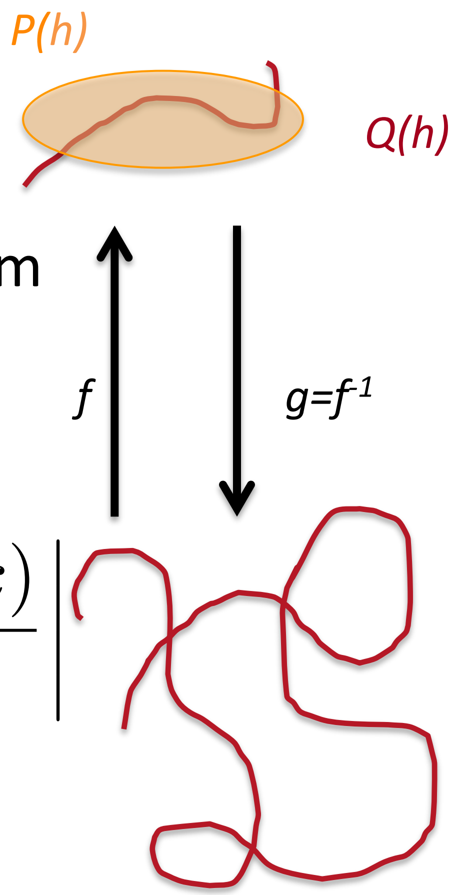


NICE
Nonlinear Independent Component Estimation

(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**



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

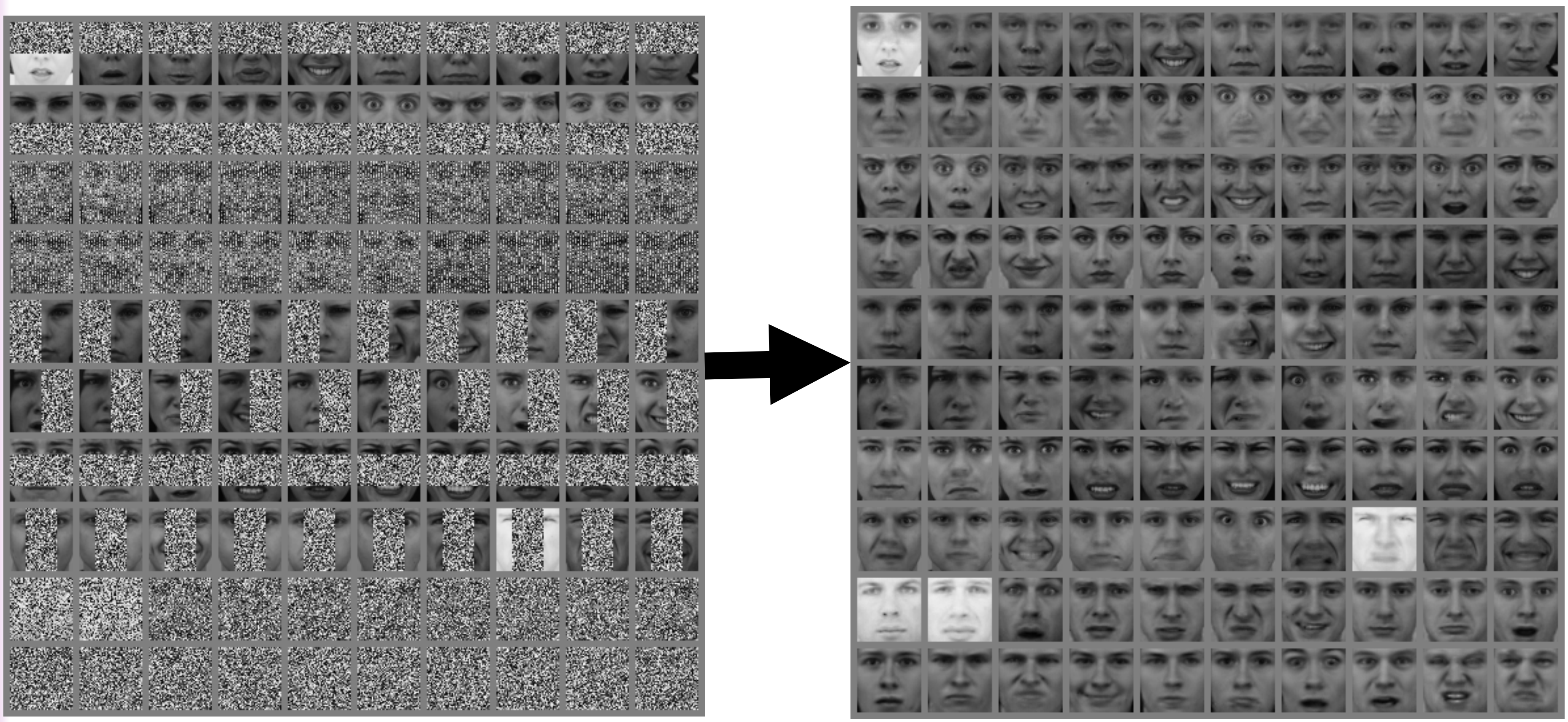
Factorized prior

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

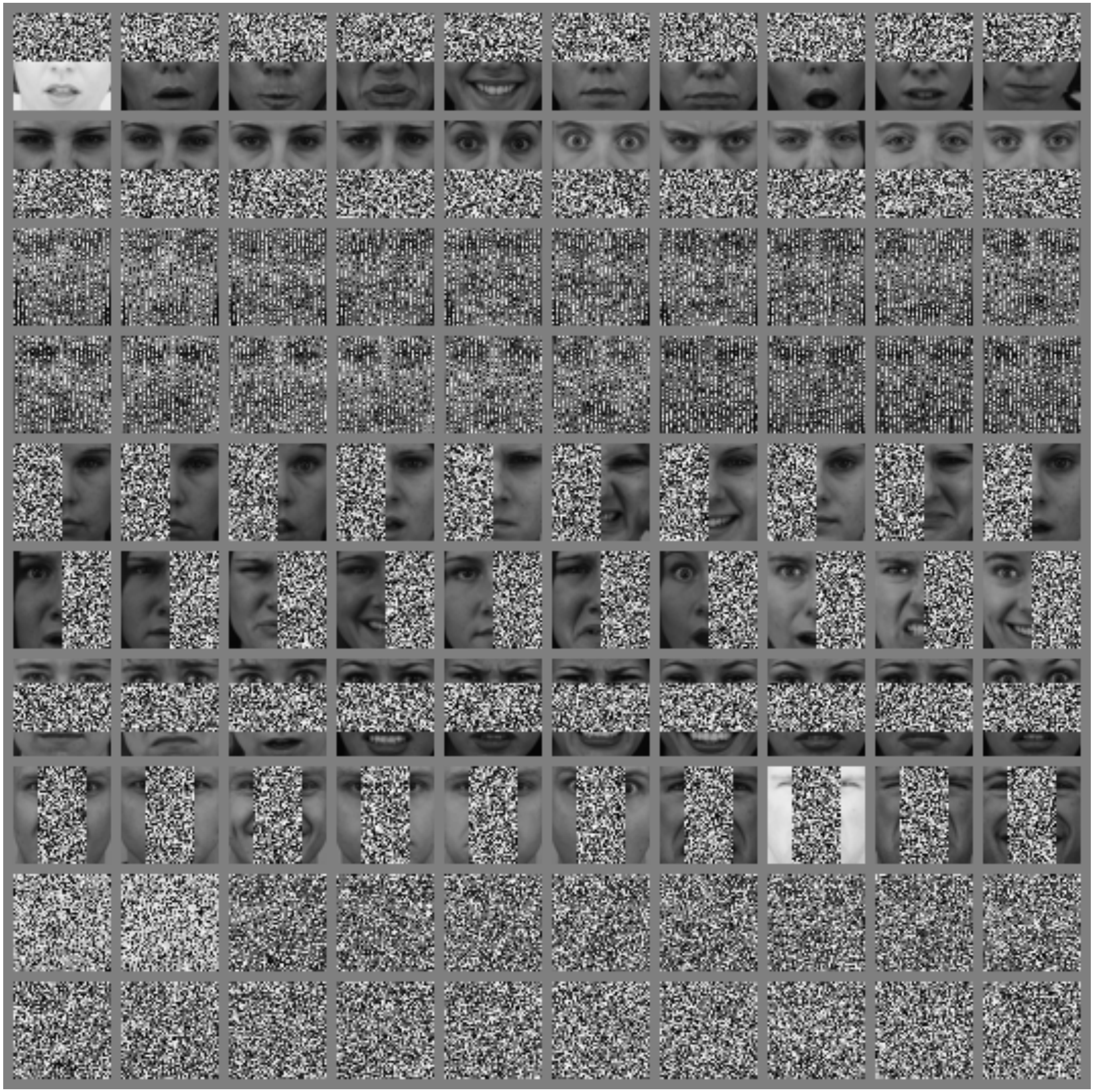
NICE Samples
(not convolutional)

NICE Inpainting

- Gradient ascent on the likelihood, over missing inputs



NICE Inpaiting Movies (not conv.)



NICE: Perfect Auto-Encoders

- Compose a series of stages that have determinant 1 or a diagonal Jacobian
- Such that each stage is trivially invertible
- And composing them allows arbitrary capacity

Encoding stage (permute *x*):

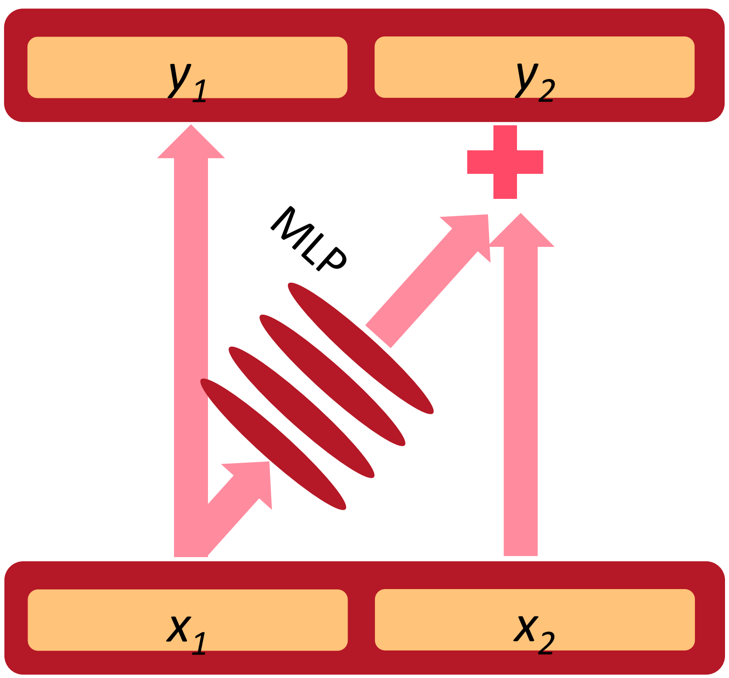
$$y_1 = x_1$$
$$y_2 = x_2 + \mathrm{MLP}(x_1)$$

Decoding stage:

$$x_1 = y_1$$
$$x_2 = y_2 - \mathrm{MLP}(x_1)$$

Determinant of Jacobian = 1

$$\begin{pmatrix} I & 0 \\ MLP'(x_1) & I \end{pmatrix}$$

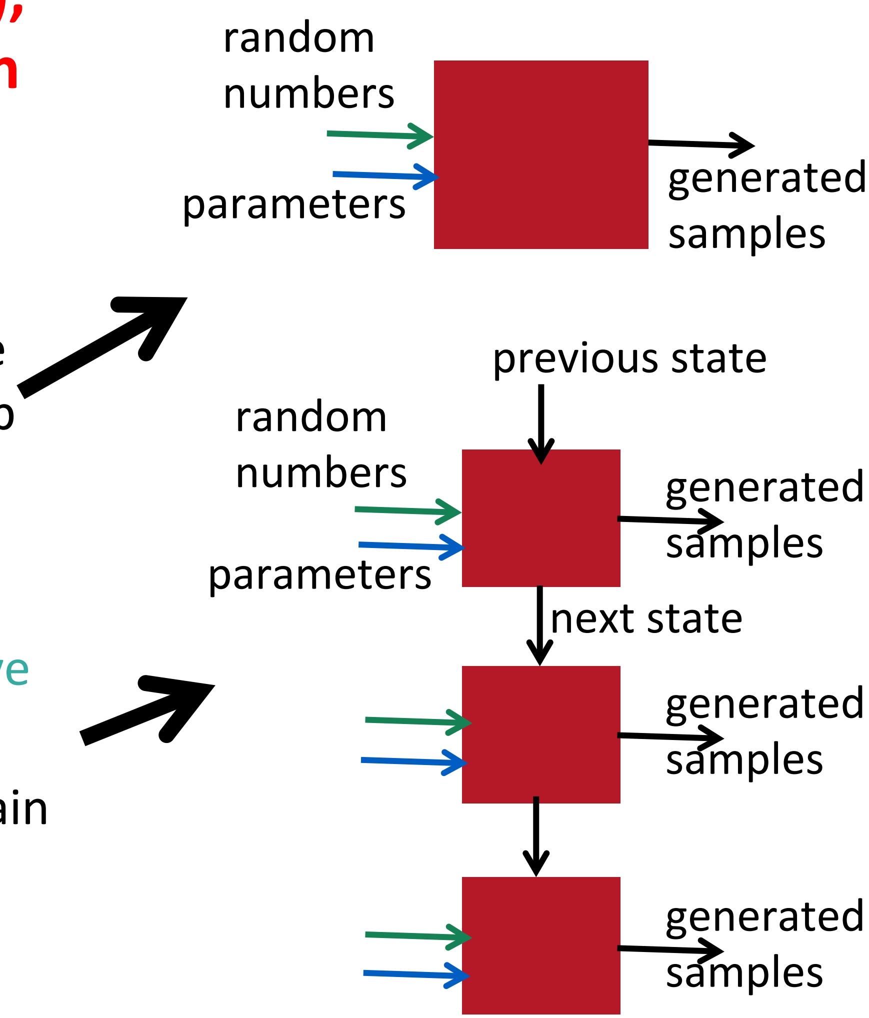


Bypassing Normalization Constants with Generative Black Boxes

- **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.



Generative adversarial networks

- Don't write a formula for $p(x)$, just learn to sample directly.
- No Markov Chain
- No variational bound
- How? By playing a game.

slide adapted from Ian Goodfellow

Adversarial nets framework

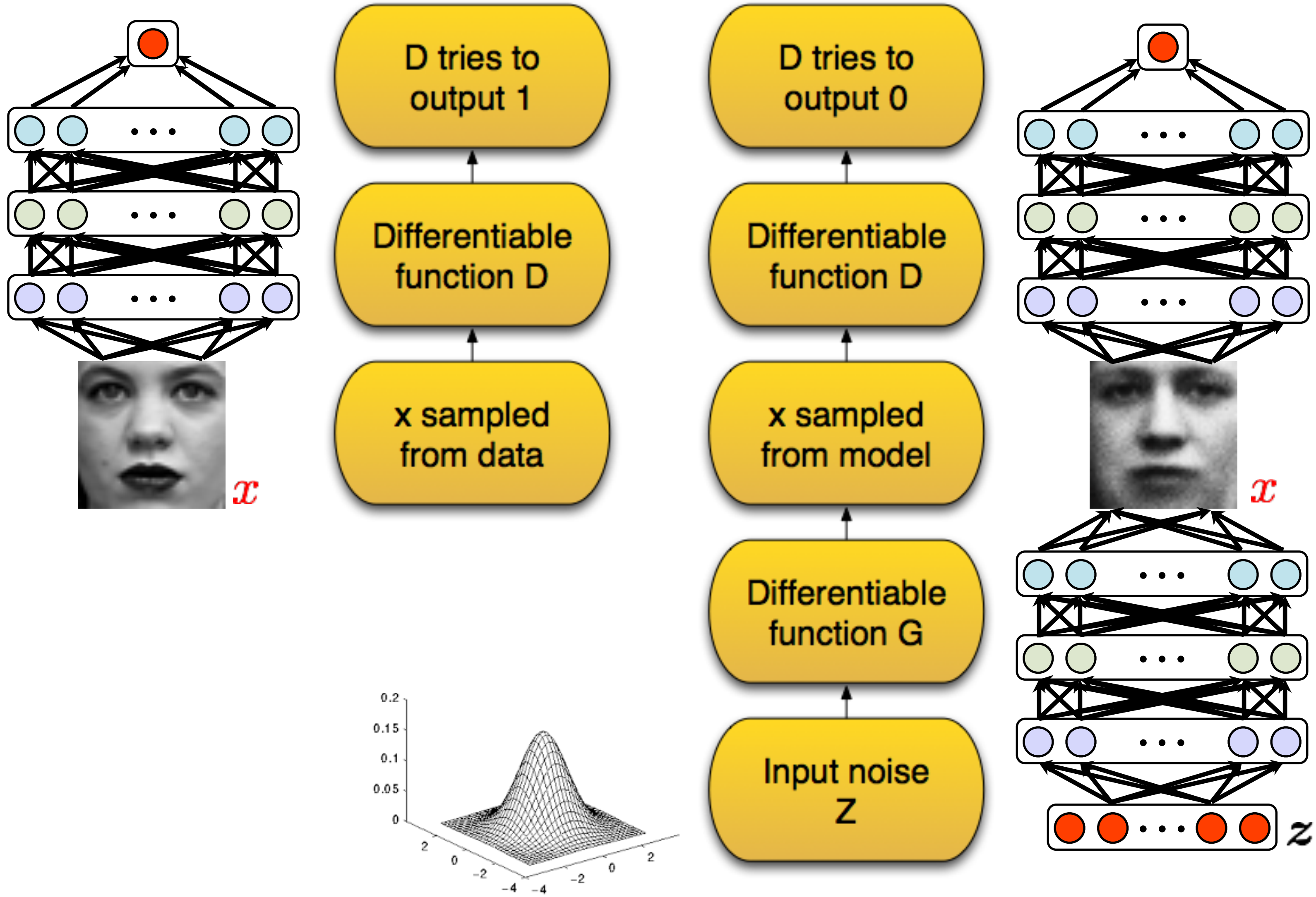
- A game between two players:
    1. Discriminator D
    2. Generator G

- D tries to discriminate between:
    - A sample from the data distribution.
    - And a sample from the generator G.

- G tries to "trick" D by generating samples that are hard for D to distinguish from data.

slide adapted from Ian Goodfellow

Adversarial nets framework



slide adapted from Ian Goodfellow

Zero-sum game

- Minimax value function:

$$\min_G \max_D V(D, G) = \mathbb{E}_{\boldsymbol{x} \sim p_{\text{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z} \sim p_{\boldsymbol{z}}(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z})))]$$

Discriminator pushes up

Discriminator's ability to recognize data as being real

Discriminator's ability to recognize generator samples as being fake

Generator pushes down

slide adapted from Ian Goodfellow

Police (Discriminator) vs Counterfeiter (Generator)

- Optimal discriminator:

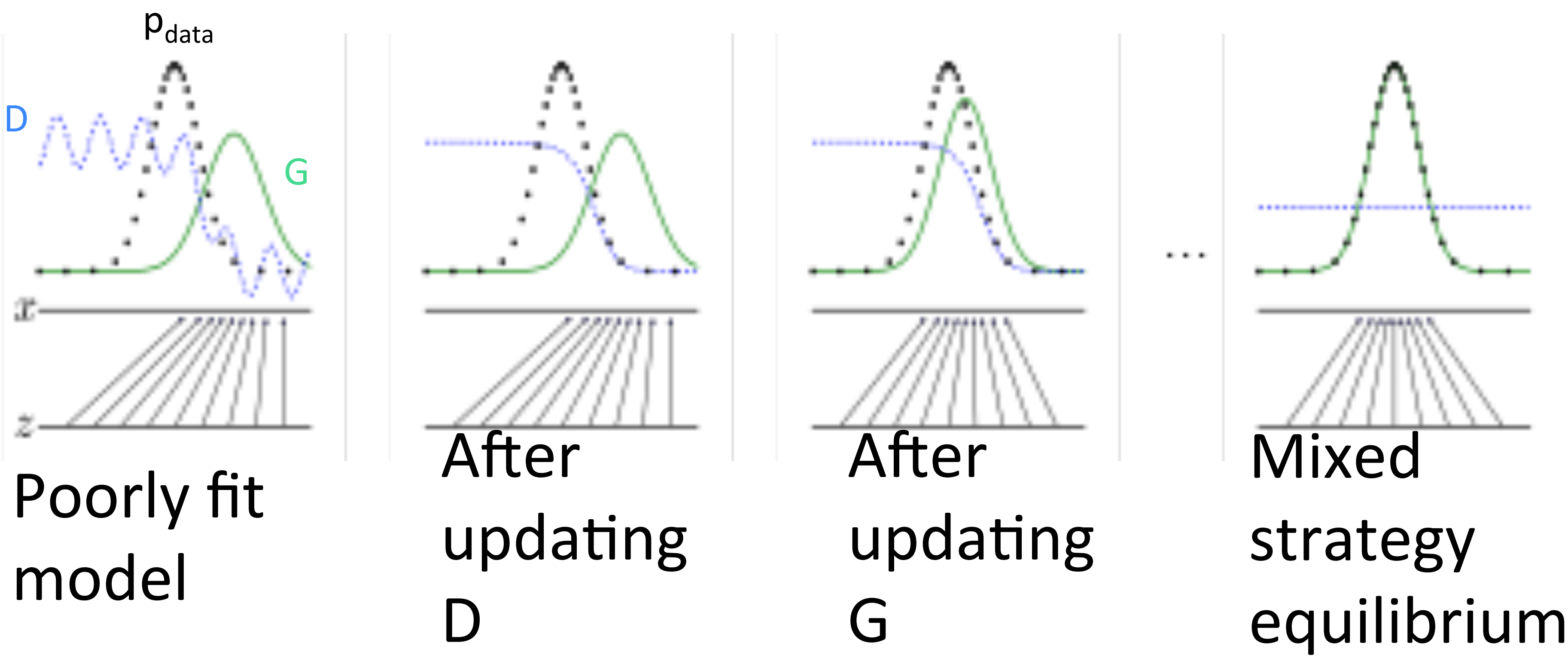
$$D(x) = \frac{p_{\text{data}}(x)}{p_{\text{data}}(x) + p_{\text{model}}(x)}$$

- Zero-sum game between discriminator D and generator G:

$$\min_G \max_D V(D,G) = \mathbb{E}_{\boldsymbol{x} \sim p_{\text{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z} \sim p_{\boldsymbol{z}}(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z})))]$$

- With non-parametric D and G and infinite data, recovers the data-generating distribution

Learning process

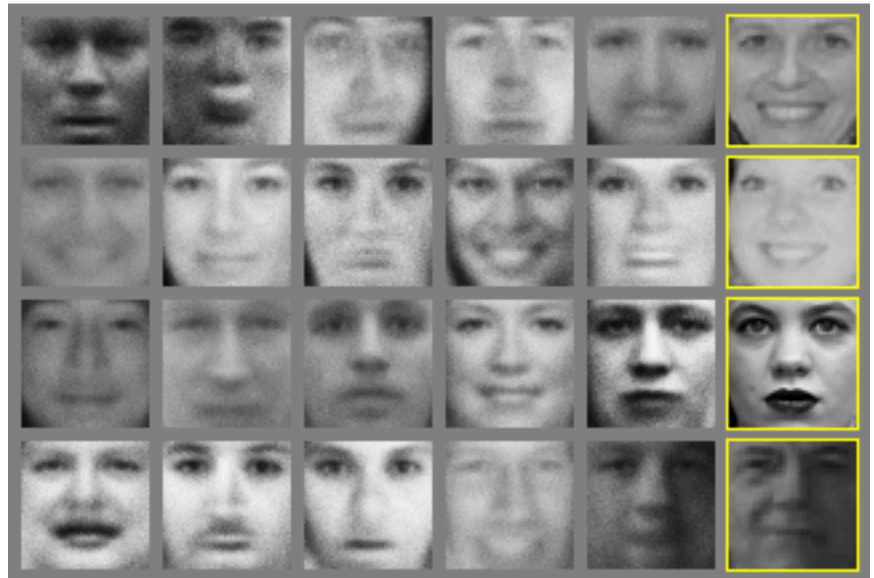


slide adapted from Ian Goodfellow

Generated Samples (see Ian's movies)

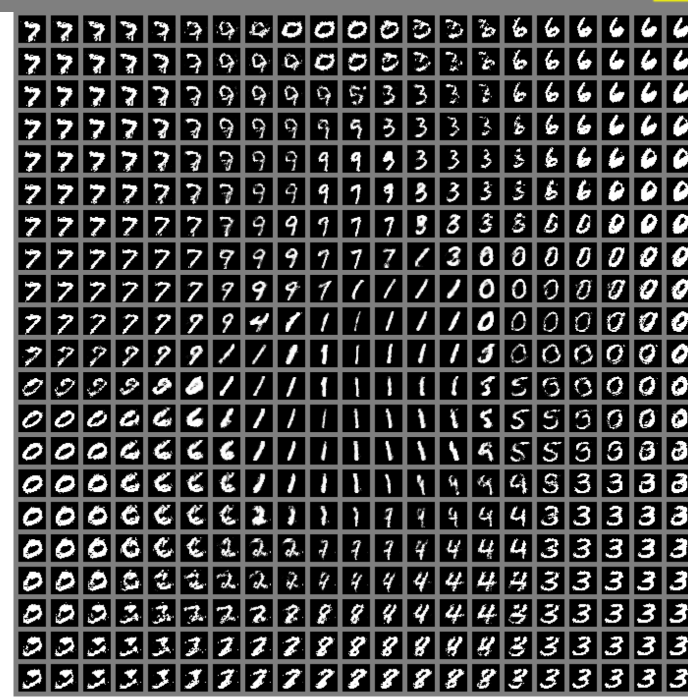
cifar

Nearest neighbor in training set



TFD

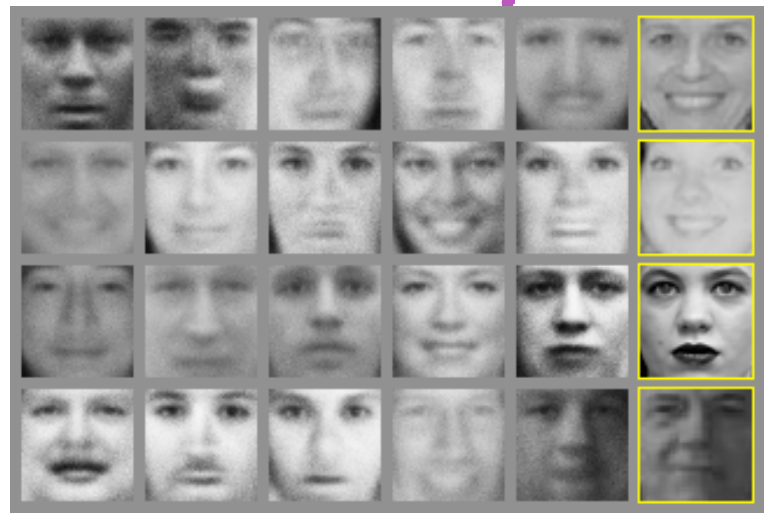
SVHN



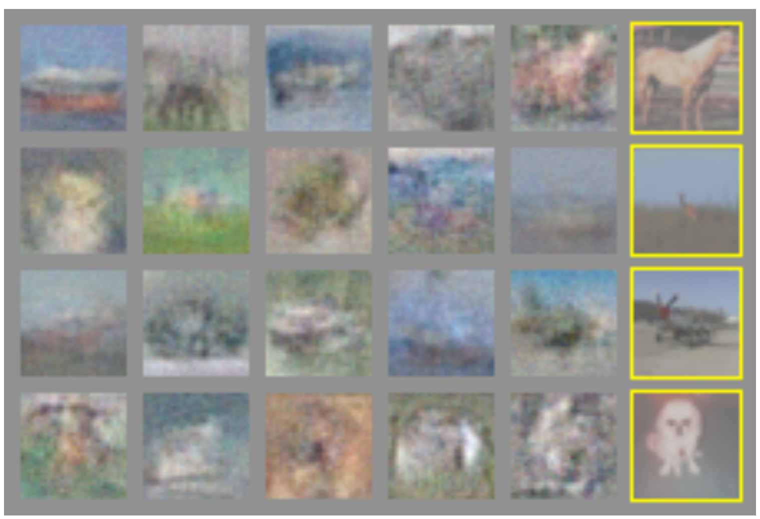
2-D manifold, MNIST

Visualization of model samples

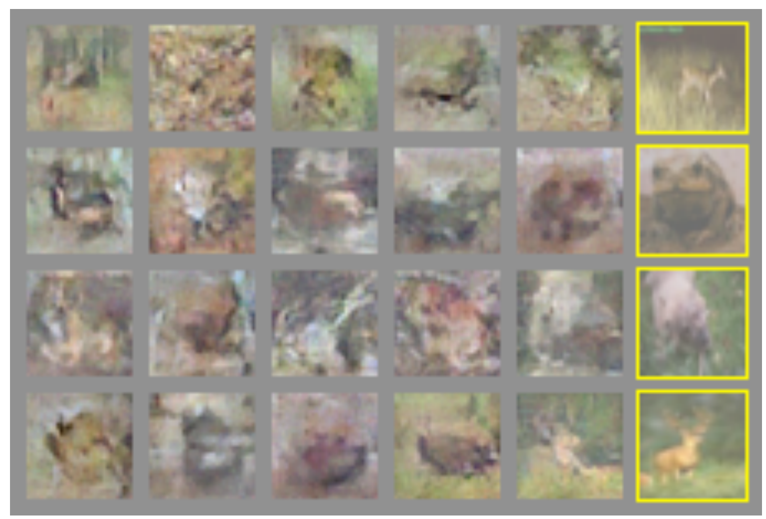
MNIST



TFD

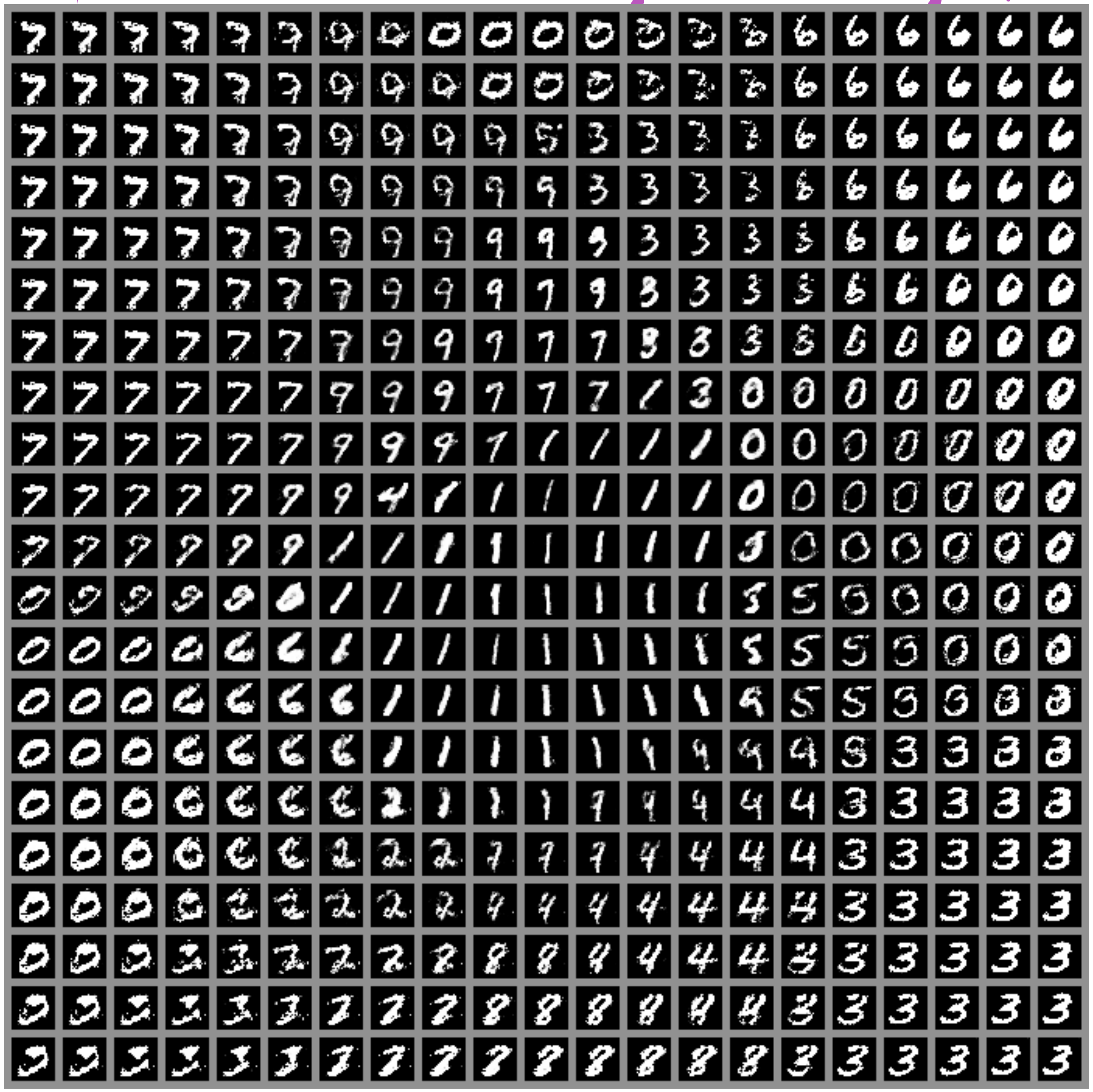


CIFAR-10 (fully connected)

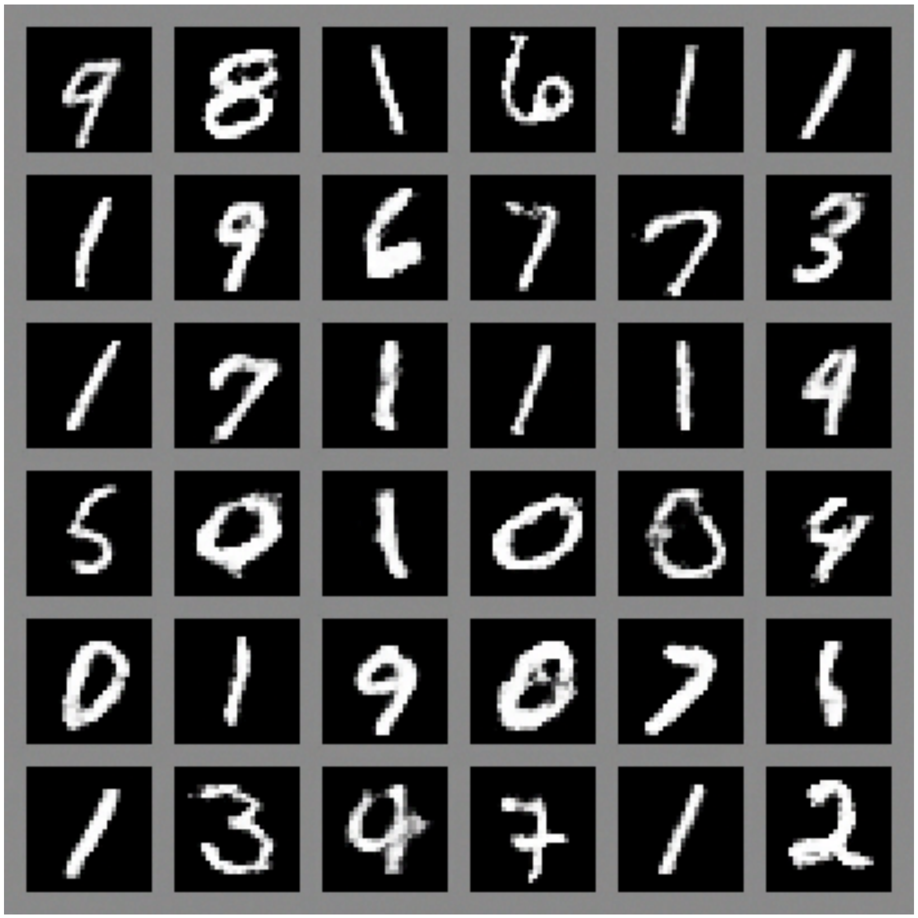


CIFAR-10 (convolutional)

Learned 2-D manifold of MNIST



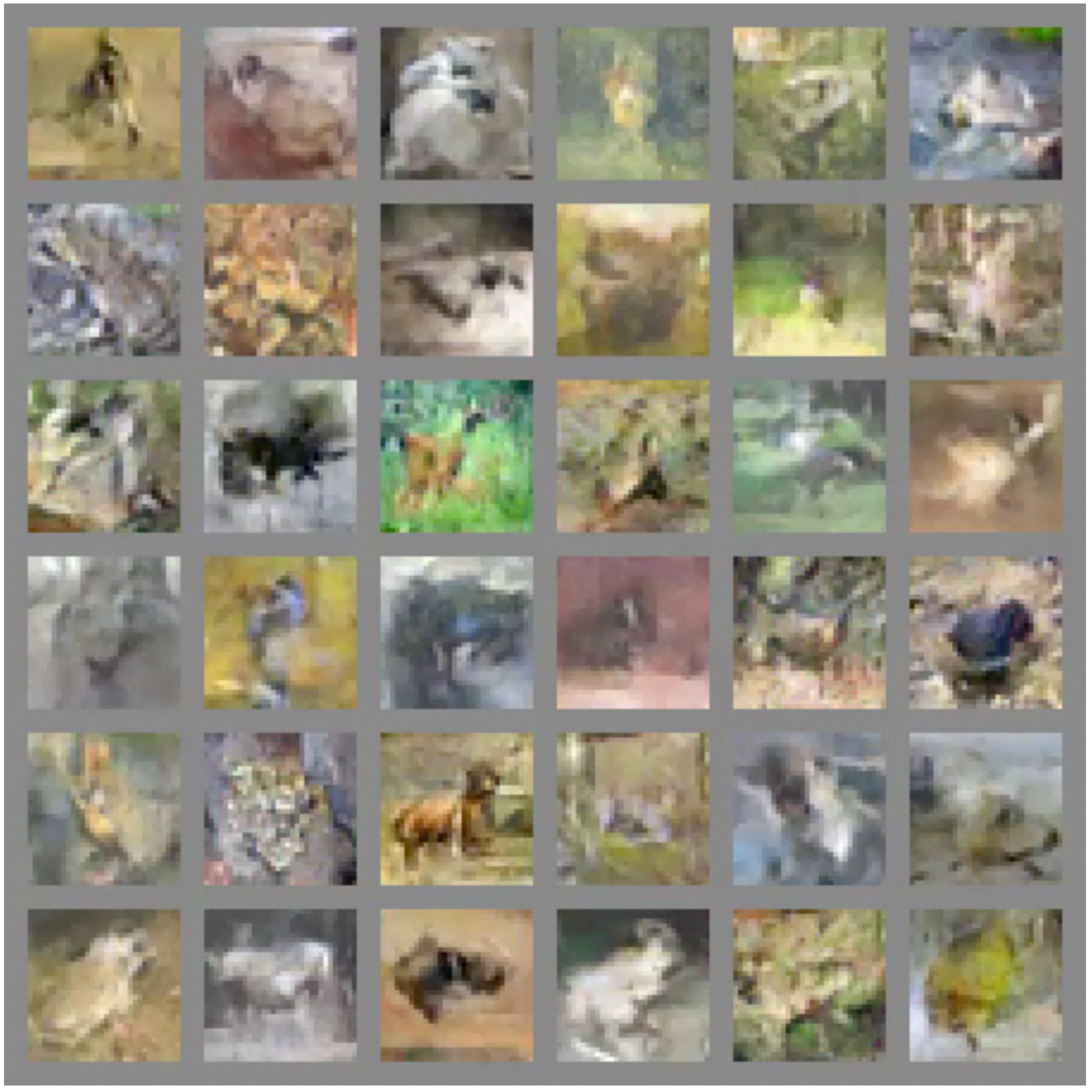
Visualization of model trajectories



MNIST digit dataset

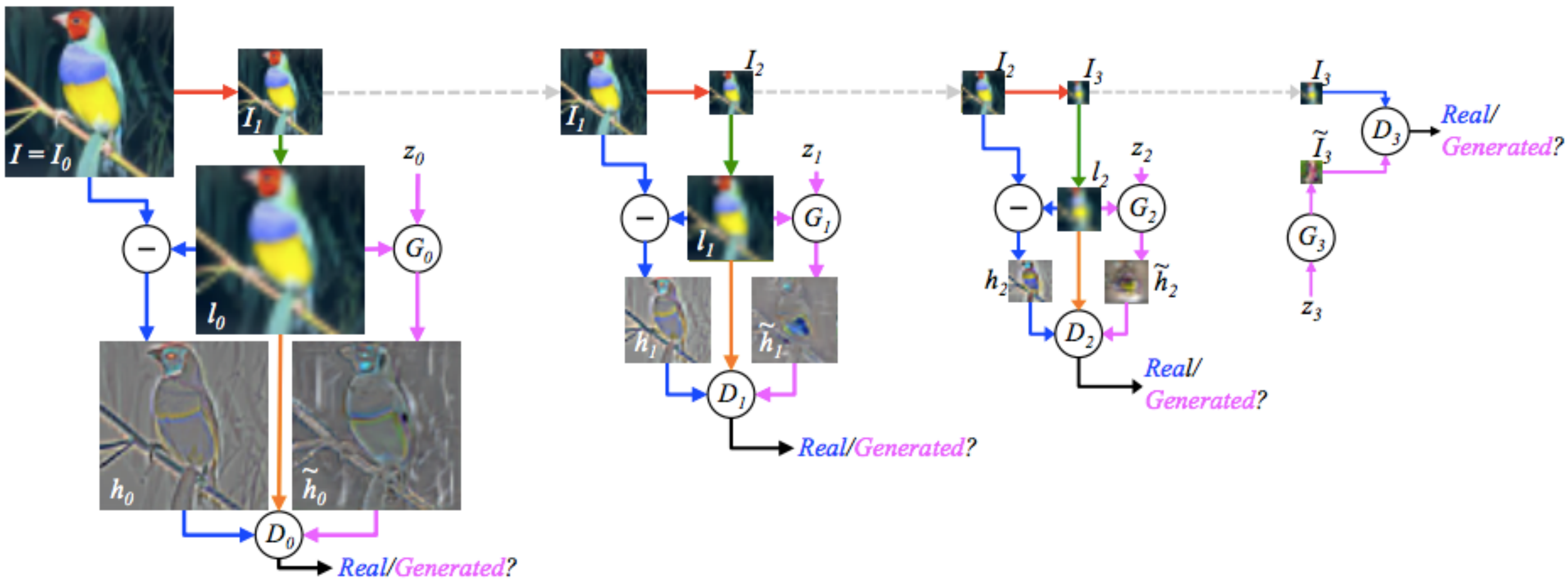
Toronto Face Dataset (TFD)

Visualization of model trajectories



CIFAR-10
(convolutional)

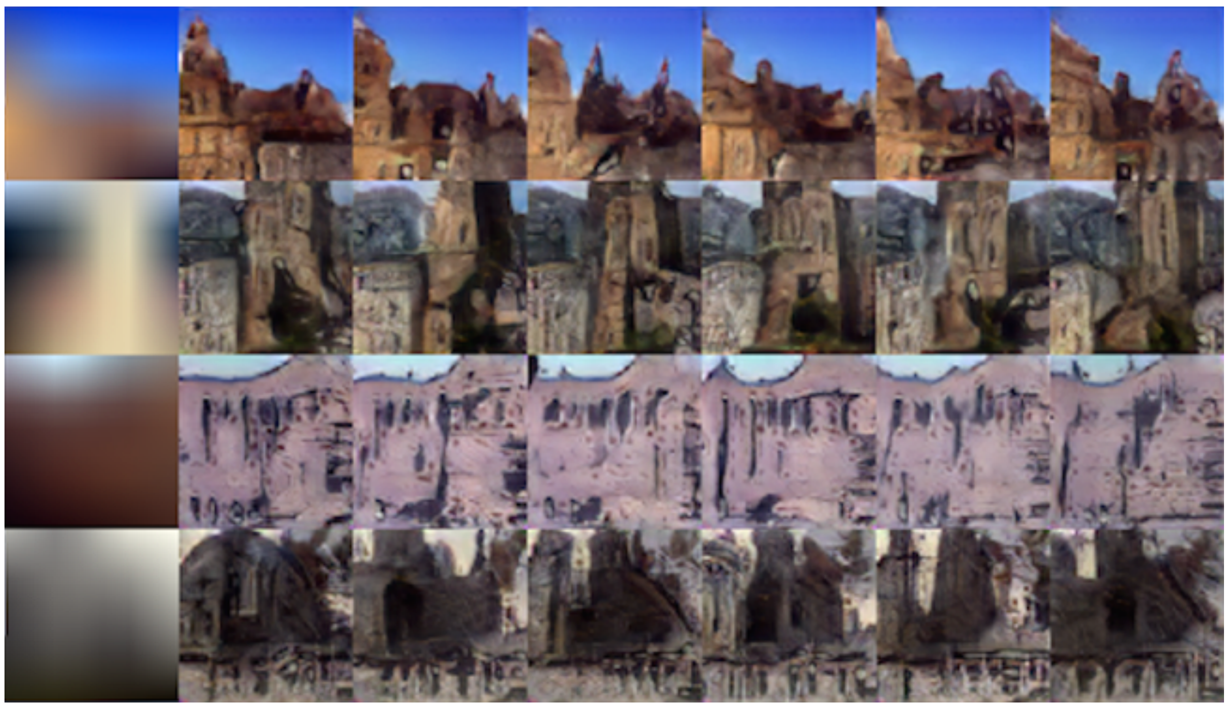
Laplacian Pyramid of Conditional GANs



(Denton + Chintala, et al arXiv 1506.05751, 2015)

LAPGAN results

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

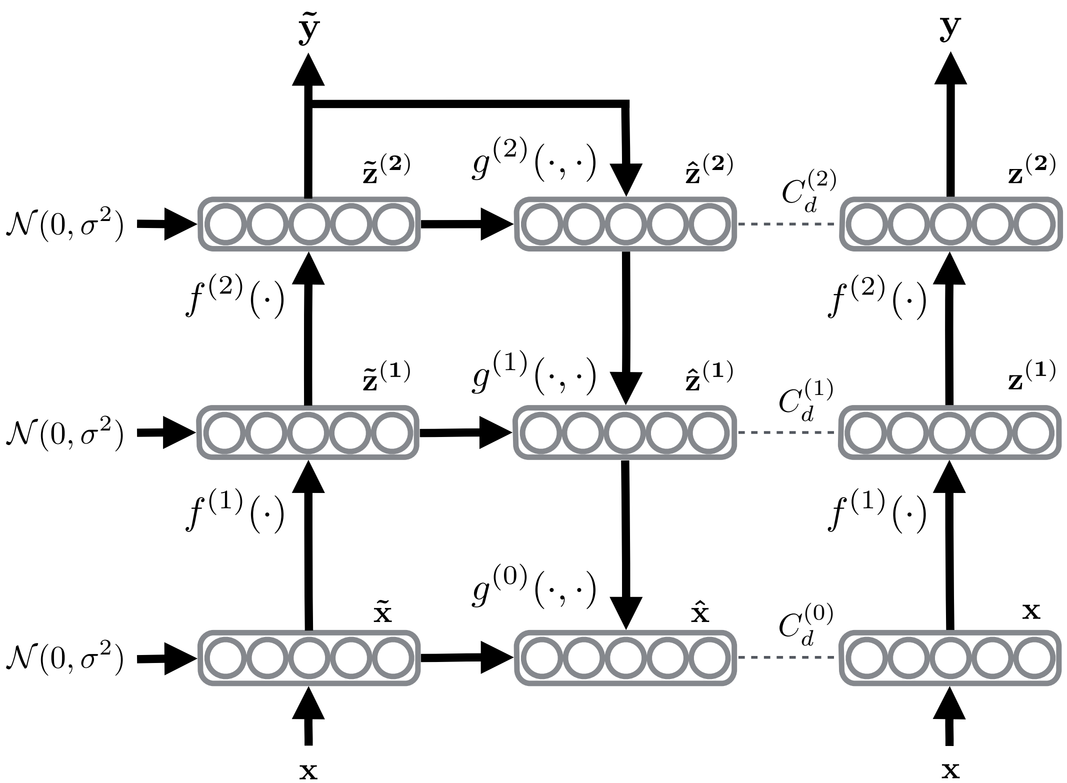


(Denton + Chintala, et al 2015)

Other Encouraging News: Semisupervised Learning with Ladder Network

(Rasmus et al, arXiv 1507.0267)

- 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

Outstanding Results

(Rasmus et al, arXiv 1507.0267)

- Permutation invariant MNIST

Test error % with # of used labels	100	1000	All
Semi-sup. Embedding (Weston *et al.*, 2012)	16.86	5.73	1.5
Transductive SVM (from Weston *et al.*, 2012)	16.81	5.38	1.40*
MTC (Rifai *et al.*, 2011b)	12.03	3.64	0.81
Pseudo-label (Lee, 2013)	10.49	3.46	
AtlasRBF (Pitelis *et al.*, 2014)	8.10 ($\pm$ 0.95)	3.68 ($\pm$ 0.12)	1.31
DGN (Kingma *et al.*, 2014)	3.33 ($\pm$ 0.14)	2.40 ($\pm$ 0.02)	0.96
DBM, Dropout (Srivastava *et al.*, 2014)			0.79
Adversarial (Goodfellow *et al.*, 2015)			0.78
Virtual Adversarial (Miyato *et al.*, 2015)	2.66	1.50	0.64 ($\pm$ 0.03)
Baseline: MLP, BN, Gaussian noise	21.74 ($\pm$ 1.77)	5.70 ($\pm$ 0.20)	0.80 ($\pm$ 0.03)
Γ-model (Ladder with only top-level cost)	4.34 ($\pm$ 2.31)	1.71 ($\pm$ 0.07)	0.79 ($\pm$ 0.05)
Ladder, only bottom-level cost	1.38 ($\pm$0.49)	1.07 ($\pm$ 0.06)	**0.61** ($\pm$ 0.05)
Ladder, full	**1.13** ($\pm$ 0.04)	**1.00** ($\pm$ 0.06)	

- The paper also shows improvement with a convolutional version, on CIFAR-10

Conclusions

- Likelihood is generally intractable
- Many criteria have been proposed as alternatives to maximum likelihood
- Denoising auto-encoders optimize a denoising score matching criterion and are generative models
- Variational auto-encoders justify noise injection in the middle of the auto-encoder
- Generative adversarial nets optimize a kind of Turing test and are currently the basis of the best generative model of images