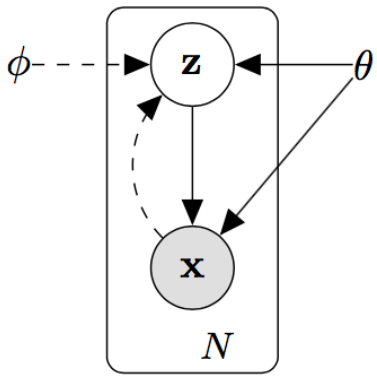
Deep Generative Models

Part I: introduction

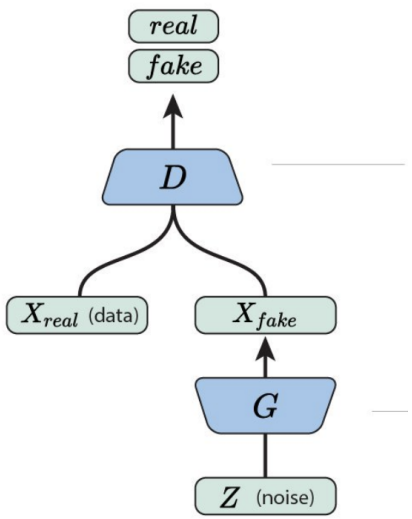
Welcome

Two of the most exciting areas of ML, combining forces.

- Probabilistic models $p_\theta(x) = \int p_\theta(x|z)p_\theta(z)dz$
  - interpretable structure, quantified uncertainty,...
- Deep neural networks $f_\theta(\cdot) = f_\theta^{(\ell)} \circ ... \circ f_\theta^{(0)}(\cdot)$
  - expressivity, simple learnability,...
- *Deep Generative Model* (DGM)
  - $p_\theta(x)$: latent $p_\theta(z)$, (noisy) map $f_\theta : \mathcal{Z} \to \mathcal{X}$.
- Central theme:



> Use probabilistic models where we have inductive bias;
> Use flexible function approximators where we do not.



- We will explore this theme with VAE, GAN, and more.

Criteria

Prerequisites

- Probabilistic machine learning $\supseteq$ Blei's PGM
- Deep learning $\supseteq$ Cunningham's AML
- Programming $\supseteq$ {python, tensorflow, pytorch, keras}

Attitude

- Excitement about this research area
- Desire to write a paper / work on a project in this domain
- Game to be lively and involved in seminar

Also

- Alignment with your research is terrific
- Diverse and nontraditional students welcome

Then again, maybe not

You aren't doing ML research already

- This course will be fast paced, Ph.D. level; prerequisites are real

You are just looking for course credit

- The format depends on the involvement and enthusiasm of a dedicated group

You are fully committed with an unrelated project

- This course will be a lot of work: reading, participation, lecturing, projects

Format

Tone

- fun and fast paced
- technical and creative
- positive and inclusive

Seminar

- Journal club-style: everybody reads, everybody participates
- RealDeal(TM): no phones, no email, no text, etc; tablets ok

Presentation

- You prepare and lead the seminar (slides, whiteboard, etc.)
- Pre-meetings with me to take me through it (Mondays 1000 ok?)

Project...

Project

Key points

- novel contribution: modeling, application, theory, code, etc.
- meaningful software implementation of a DGM
- project report with content and style of a top machine learning conference
  - intro, lit review, model, implementation, results, figures, discussion,...
- given: ideas, templates (paper, repo), support (Wednesdays 1600 ok?), etc.

Last time (STAT G8325, 2015-2016)

- ML modeling papers published: [Fagan+ UAI 2016, Bloem-Reddy+ ICML 2016, Fagan+ AISTATS 2017]
- Applications papers: [Elsayed+ Nature Communications 2016, Mena+ PLOS CB 2017]
- Code: new feature set added to `STAN`

In-bounds topics

- complicated probability distribution you want to learn
- intractable probabilistic inference where MCMC or similar might not do
- learn latent structure in high-dimensional, noisy data
- learn to sample from a complex and high-d $p_\theta(x)$
- applied DGM research (materials? drug discovery? quant trading?)
- questions?

Topics

Part	Week Counter	Content
I	1	Introduction and basics • Reading: **[KW13, GPAM+14]**; [RMW14]
II	$+1$	Structured VAE models • Reading: **[APB+15, JDW+16]**; [GAPC16, KSS15]
III	$+2-3$	Conditional and discrete DGMs • Reading: TBD
IV	$+2-3$	Disentangling, interpreting, and geometry of VAE latents • Reading: TBD
-	$+1$	(spring holiday)
V	$+2-3$	VAE next-level mechanics: inference, reparameterizations, lower bounds, etc. • Reading: TBD
VI	$+1$	VAE $\rightarrow$ GAN: learning likelihood free models • Reading: **[ML16]**; [TRB17, KHL16]
VI	$+1$	AR density estimation: NADE, MADE, NVP, norm flow, PixelRNN, WaveNet, etc. • Reading: TBD
VII	$+2-3$	GAN mechanics: basic issues, f, Wasserstein, OT, etc. • Reading: TBD
VIII	$+1-3$	GAN applications: superresolution, CycleGAN, steganography, etc. • Reading: TBD
IX	$+1-2$	GAN issues and emerging theory • Reading: TBD
X		Project final reports

Great, but I'm not registered...

Course enrollment will be prioritized roughly as:

1. Columbia Statistics PhD students
2. other Columbia students
3. Columbia auditors
4. non-Columbia enthusiasts
5. ...
6. faculty

Homework (Due Friday 25 January at 4:59pm)

```
https://goo.gl/forms/Y9A1DW8s81ZaNKs83
```

Despite (because of?) all of these particulars, I do hope you will join...

At last, content

Modeling

A (the?) central problem in statistics is fitting a *model* to *data*:

$$\mathcal{M} = \{p_\theta : \theta \in \Theta\} \quad , \quad X = \{x_1, ..., x_n\} \quad \rightarrow \quad \arg\max_\theta \log p_\theta(X)$$

Excellent strategy: partition randomness and structure...

- Generate *latent* $z_i \sim p_0(z)$, compute $x_i = f_\theta(z_i)$ with (noisy) f_θ
- Induces a (possibly) complex model/family of distributions $p_\theta(x)$
- This approach can take a few forms...

Prescribed probabilistic models:

- form $p_\theta(x)$ directly, proceed to inference, parameter estimation, regularization, etc.
- many simple models have this form...

Implicit Probability Models

Implicit probabilistic models:

- Specify a latent $p(z)$ followed by a procedure $f_\theta : \mathcal{Z} \to \mathcal{X}$
- Key point: in this setting, sampling data is almost always easy
- Sometimes the whole problem is easy: remember inversion sampling?

$$z \sim Unif(0,1) \quad x = F_\phi^{-1}(z) \quad \to \quad x \sim Exp(\phi)$$

F_ϕ is the cdf of the exponential distribution, $F_\phi(x) = 1 - \exp(-\phi x)$, with $F_\phi^{-1}(z) = -\phi \log(1-z)$

- More often f_θ induces an intractable log likelihood:

$$p_\theta(x) = \frac{\partial}{\partial x_1} \ldots \frac{\partial}{\partial x_d} \int_{\{f_\theta(z) \leq x\}} p_\theta(z) dz$$

1. Hard: tractable, specified conditional likelihood (bayesian/graphical models):

$$p_\theta(x) = \int p_\theta(x|z) p_\theta(z) dz$$

2. Harder: stochastic sampling procedures (diff eq, ecology, weather, finance, etc):

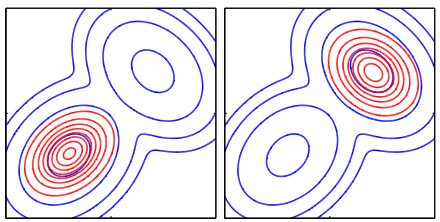
$$p_\theta(x|z) = ??$$

IPM with deep neural networks

With almost no exception, hereafter DGM == IPM. The main idea:

- Sample randomness from a particularly easy distribution $z \sim \mathcal{N}(0, I)$
- Use a deep neural network as the structure map f_θ
- Best of both worlds? ...flexible, expressive $p_\theta(x)$ that is easy to sample and learn

1. Tractable $p_\theta(x|z) \to$ EM, VI, ELBO,...



   - Today: the *variational autoencoder* [KW13]

2. Intractable $p_\theta(x|z) \to$ generative modeling...

$$z_i \sim \mathcal{N}(0, I) \quad \to \quad g_\theta(z_i) \quad \to$$

   - Today: the *generative adversarial network* [GPAM+14]

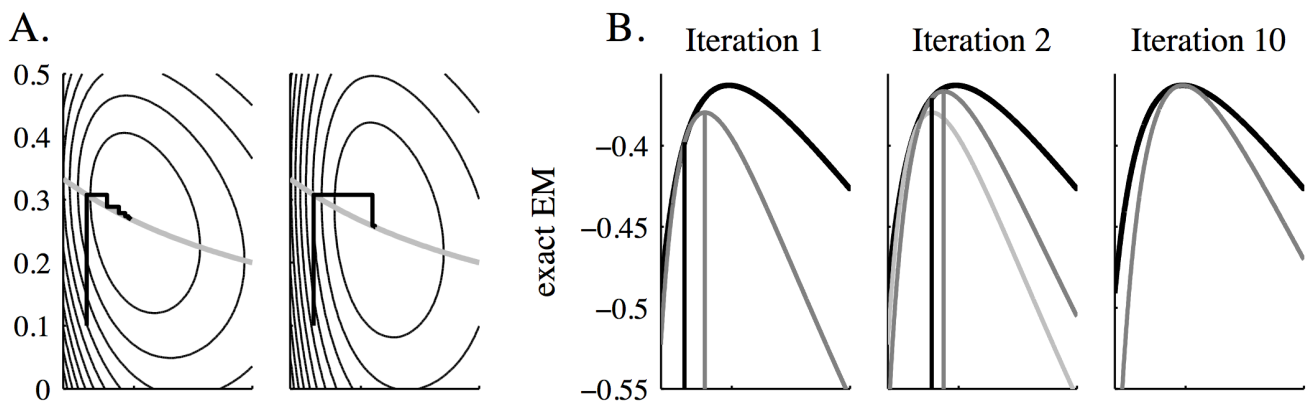
3. Much more to come on IPM approaches (see Part VI and [ML16]).

Tractable $p_\theta(x|z)$: EM

We wish to maximize $\log p_\theta(x) = \log \int p_\theta(x|z)p_\theta(z)dz$. EM:

$$\begin{aligned}
\log p_\theta(x) &= \log \int p_\theta(x|z)p_\theta(z)dz \\
&= \log \int p_\theta(x|z)p_\theta(z)\frac{q(z)}{q(z)}dz \\
&\geq \int q(z)\log\frac{p_\theta(x|z)p_\theta(z)}{q(z)} \qquad = \quad \mathcal{L}(q,\theta)
\end{aligned}$$

Key observation: if q is the posterior $p_\theta(z|x)$, then bound is achieved.



[Turner and Sahani 2011]

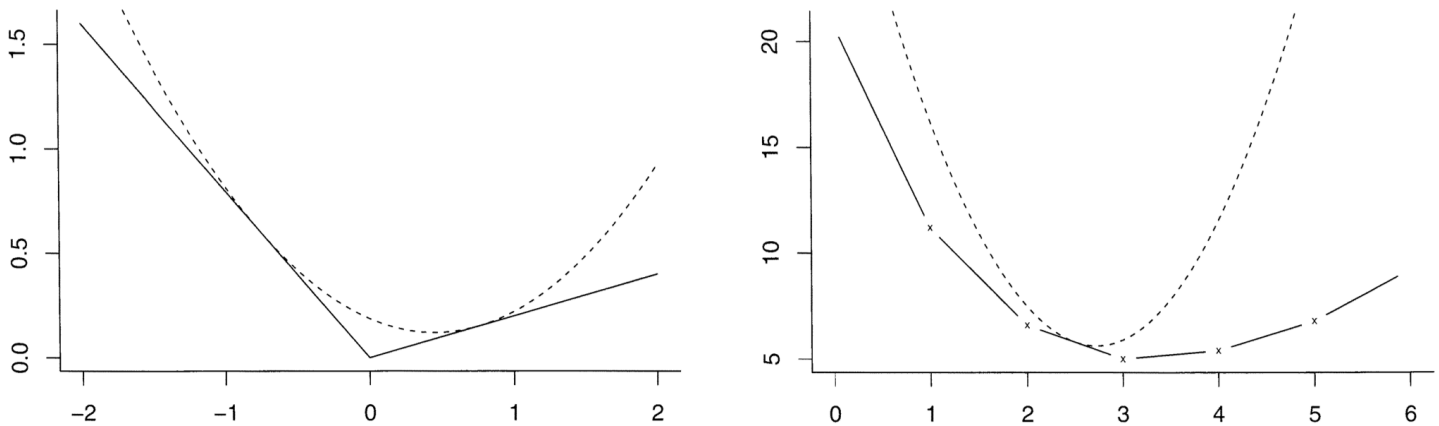
NB: A is $\mathcal{L}(\theta, q)$ (ignore right panel); B is EM as a function of θ (M steps).

Tractable $p_\theta(x|z)$: (sidebar) MM algorithms

EM is a special case of a *minorize-maximize* algorithm (MM):

- Bound $g(\theta)$ has to be tight. Or at least gap independent of θ.
- Specifically two conditions (here majorize-minimize):

$$\begin{aligned} g\left(\theta|\theta^m\right) &\geq f(\theta) \quad \forall\theta \\ g\left(\theta^m|\theta^m\right) &= f(\theta^m) \end{aligned}$$



[Hunter and Lange 2004]

NB: here $f(\theta) = \log p_\theta(x)$, aka the ELBO is tight when q is posterior.

Tractable $p_\theta(x|z)$: variational posterior

Suppose we can't calculate the posterior; then *for a fixed* θ we can still:

$$\arg\min_\phi KL\left(q_\phi(z)||p_\theta(z|x)\right) \quad = \quad \arg\max_\phi \mathcal{L}(\phi, \theta)$$

Notice

- We call this maximizing the ELBO with a model $\mathcal{Q} = \{q_\phi(z) : \phi \in \Phi\}$
- It's VI, and is also an ok thing to do.
- Lots of textbook work here, but...

  [Murphy 2014, Mackay 2003, Bishop 2006, Wainright and Jordan 2008, Jordan et al 1999, etc.]

> *There is as much "art" as there is "science" in our current understanding of how variational methods can be applied to probabilistic inference.*
> – [Jordan, Ghahramani, Jaakkola, Saul 1999]

Warning: we will get confused about our goal – $\arg\max \log p_\theta(x)$ or $\arg\min KL(q||p)$...

Tractable $p_\theta(x|z)$: vEM (these days $\approx$ VI)

- Why not do both – variational EM (replace E step with VI):

$$\underset{\text{vE-step}}{\arg\min_\phi KL\left(q_\phi(z)||p_\theta(z|x)\right)} \quad = \quad \underset{\text{M-step}}{\arg\max_\phi \mathcal{L}(\phi,\theta)}$$

$$\arg\max_\theta \mathcal{L}(\phi,\theta)$$

- An old and well practiced idea [Jordan et al 1999]
- Extremely common now [Kingma and Welling 2013, ...]
- Mechanics:

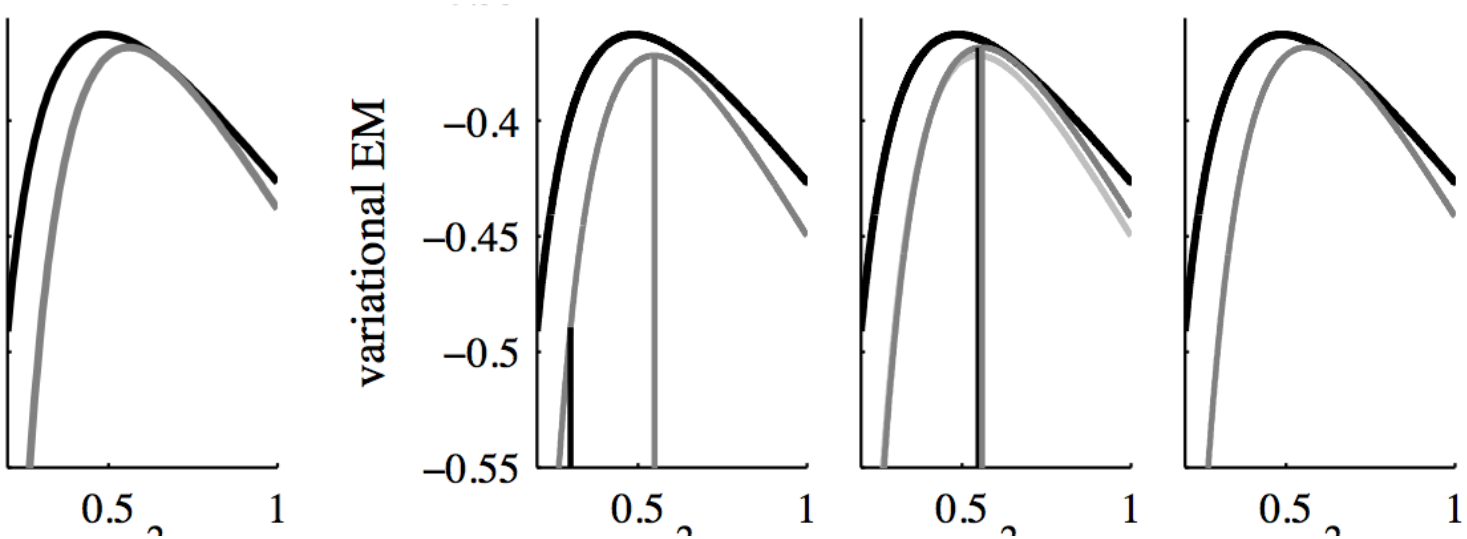
$$\begin{aligned} KL(q_\phi(z)|p_\theta(z|x)) &= E_{q_\phi}\left(\log\frac{q_\phi(z)}{p_\theta(z|x)}\right) \\ &= E\left(\log q_\phi(z)\right) - E\left(\log p_\theta(z|x)\right) \\ &= E\left(\log q_\phi(z)\right) - E\left(\log p_\theta(z,x)\right) + \log p_\theta(x) \\ &\propto E_{q_\phi}\left(\log q_\phi(z)\right) - E_{q_\phi}\left(\log p_\theta(z,x)\right) \\ &= \mathcal{L}(\phi,\theta) \end{aligned}$$

- Is this ok?

Tractable $p_\theta(x|z)$: (sidebar) ELBO gap

Notice

- ...we are violating the second condition of an MM algorithm
- ...we are changing the optimization landscape each iteration
- A key result from [Turner and Sahani 2011]



Reminder

- not an issue with VI
- not an issue with EM.
- Rather the (careless?) combination of the two.
- Why is EM ok in the first place (uneven: posterior + global MLE...)?
- Is this a bigger or smaller issue in neural network land?

Tractable $p_\theta(x|z)$: (sidebar) ELBO gap

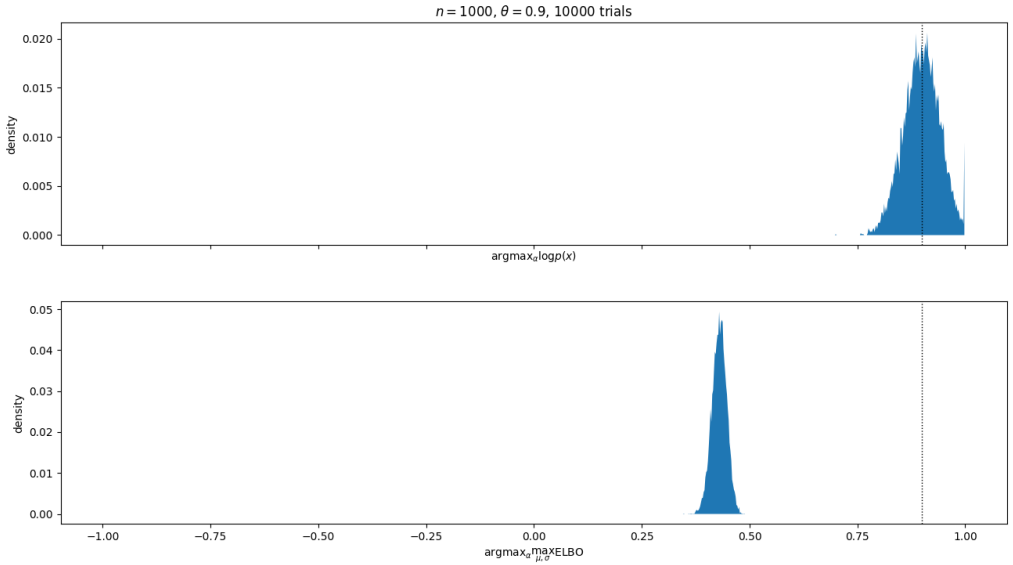
Simple closed-form example that will allow us to explore this pathology:

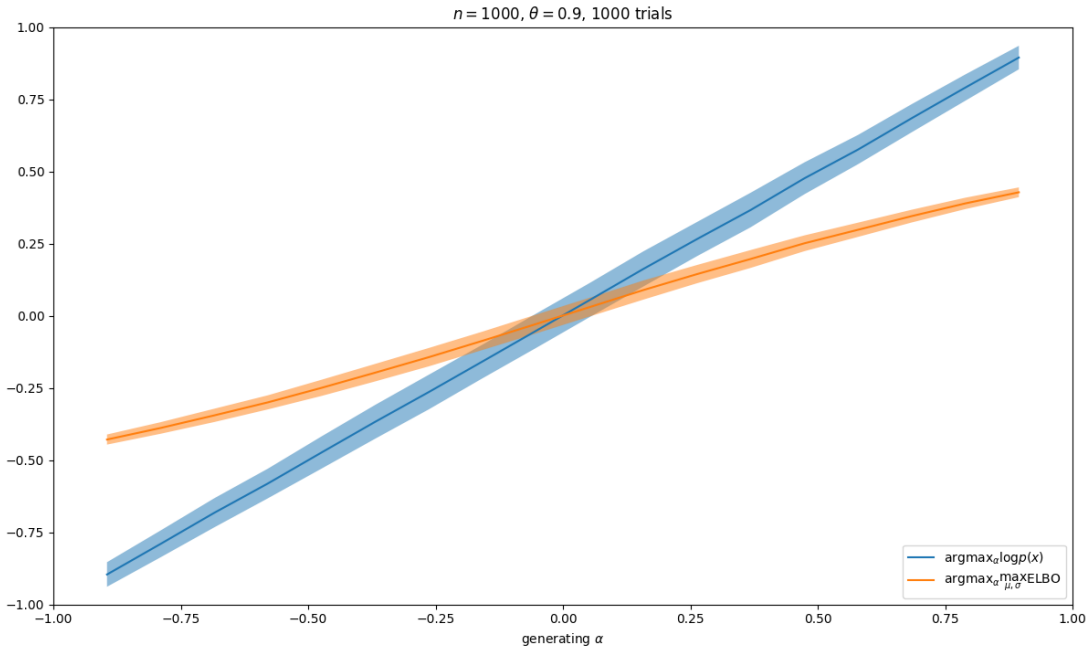
$$\begin{aligned} p_\alpha(z) &= \mathcal{N}\left(0, \begin{bmatrix} 1 & \alpha \\ \alpha & 1 \end{bmatrix}\right) \\ p(x_i|z_i) &= \mathcal{N}\left(z_i, \begin{bmatrix} \beta & 0 \\ 0 & \beta \end{bmatrix}\right) \\ q_{\mu,\sigma}(z) &= \prod_{d=1}^{2} \mathcal{N}\left(\mu^d, \sigma^d\right) \end{aligned}$$

Note:

- Evidence $p_\theta(x)$ and posterior $p_\theta(z|x)$ are closed-form
- $\theta = \alpha$ for simplicity
- $\phi = \{\mu^1, \mu^2, \sigma^1, \sigma^2\}$
- Variational parameters $\arg\max_\phi \mathcal{L}(\phi, \theta)$ also closed form...

Tractable $p_\theta(x|z)$: (sidebar) ELBO gap





Tractable $p_\theta(x|z)$: (sidebar) ELBO gap

We dwell on this for a few reasons:

- vEM (VI) is a biased estimator of θ
- ELBOs are lower bounds and thus can not be compared across models
- Other issues with VAE will arise later (Parts IV-V), but this is different
- Remember why we even do EM, VI, etc (literature is confused on this point)

Tractable $p_\theta(x|z)$: variational autoencoder

Armed with vEM, we:

- choose a generative model $\mathcal{M} = \{p_\theta(x|z)p_\theta(z) : \theta \in \Theta\}$
- choose a variational model $\mathcal{Q} = \{q_\phi(z|x) : \phi \in \Phi\}$
- maximize:

$$\mathcal{L}(\phi, \theta) = -E_{q_\phi}\left(\log q_\phi(z|x)\right) + E_{q_\phi}\left(\log p_\theta(z, x)\right)$$

View this setup as *dimension reduction*:

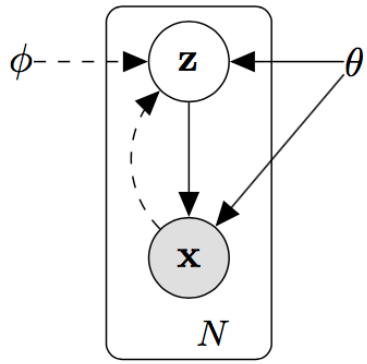
- $p_\theta(x|z)$ is a *probabilistic decoder*, converting latent code z to observed data x
- $q_\phi(z|x)$ is a *probabilistic encoder*, converting observed data x to latent code z

We need only choose our model particulars...

Tractable $p_\theta(x|z)$: variational autoencoder

ELBO:

$$\mathcal{L}(\phi, \theta) = -E_{q_\phi}\left(\log q_\phi(z|x)\right) + E_{q_\phi}\left(\log p_\theta(z, x)\right)$$



- Suppose $z \in \mathbb{R}^d$ and $x \in \mathcal{X}$

DGM: use deep networks as flexible, expressive function families:

$$q_\phi(z|x) = \mathcal{N}\left(\mu_\phi(x), \sigma^2_\phi(x)\right)$$

- Here both μ_ϕ and σ_ϕ are MLPs that map $\mathcal{X} \to \mathbb{R}^d$
- Perhaps easier to view this from the perspective of a noise variable ϵ:

$$\epsilon \sim \mathcal{N}(0, I_d) \quad \text{and} \quad z = \mu_\phi(x) + \sigma_\phi(x) \odot \epsilon \quad \to \quad z|x \sim \mathcal{N}\left(\mu_\phi(x), \sigma^2_\phi(x)\right)$$

- This *reparameterization trick* makes it simple to sample from $q_\phi(z|x)$ (Part V)
- Note: this gaussian is one basic choice, but many others are used (Parts II-V)
- Note: full VAE also has DGM for p_θ (but needn't)

Tractable $p_\theta(x|z)$: stochastic optimization

We still have the issue of calculating (and differentiating!) these expectations:

$$\arg\max_{\phi,\theta} \mathcal{L}(\phi,\theta) = \arg\max_{\phi,\theta} -E_{q_\phi}\left(\log q_\phi(z|x)\right) + E_{q_\phi}\left(\log p_\theta(z,x)\right)$$

Turn to stochastic optimization and mini-batch gradient descent:

- Draw a noise minibatch $\epsilon_1, ..., \epsilon_L$ iid from $\mathcal{N}(0, I)$ (often $L = 1$)
- Draw a data minibatch $x_1, ..., x_M$ from the dataset (often $M = 100$ or similar)
- Compute $z_{m\ell} = \mu_\phi(x_m) + \sigma_\phi(x_m) \odot \epsilon_\ell$ (...sidesteps explicit diff of $E_{q_\phi}(\cdot)$!)
- Optimize with SGD (Adam, etc.) as usual with stochastic gradients:

$$\nabla_{\phi,\theta}\mathcal{L}(\phi,\theta) \approx \frac{N}{ML}\sum_{\ell=1}^{L}\sum_{m=1}^{M} -\nabla_\phi \log q_\phi(z_{m\ell}|x_m) + \nabla_{\phi,\theta}\log p_\theta(z_{m\ell}, x_m)$$

- note that in some cases entropy (first) term will have closed form E_{q_ϕ}...

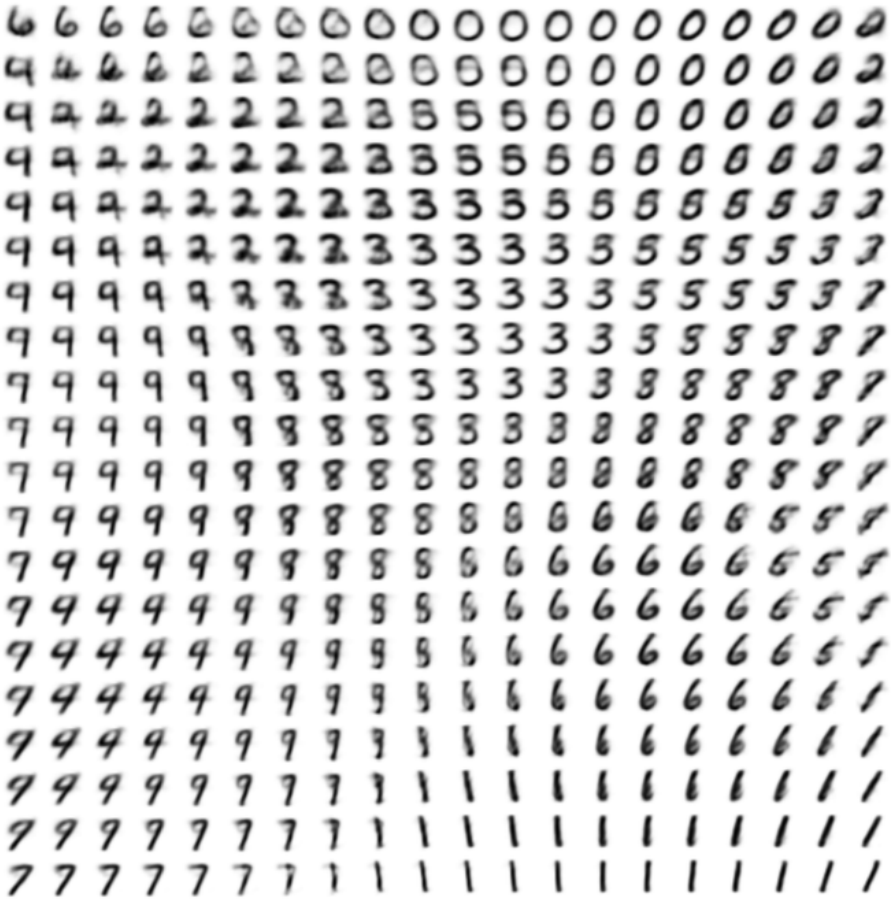
Optimizing this objective:

- Learns a generative distribution $p_\theta(x)$
- Learns an *amortized* approximation $q_\phi(z|x)$ that can be queried for any x
- We call this general approach variational autoencoding (VAE) [KW13]

Tractable $p_\theta(x|z)$: VAE in action

Learn the autoencoder and then:

- Choose a point z_i in latent space (not drawing from the posterior!)
- Decode this point with $x_i \sim p_\theta(x_i|z_i)$:

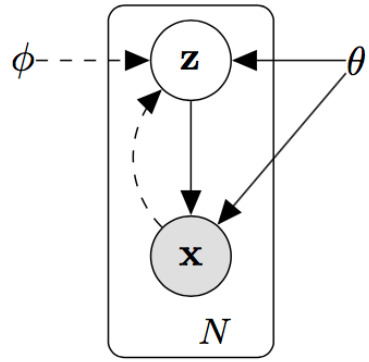


Learns a manifold of simple images and how to generate...

From VAE to GAN

Variational autoencoders:

- ...are usually thought of as doing inference (but...)
- ...are seen as dimension reduction / latent modeling
- can generate, but that is not their specific design...



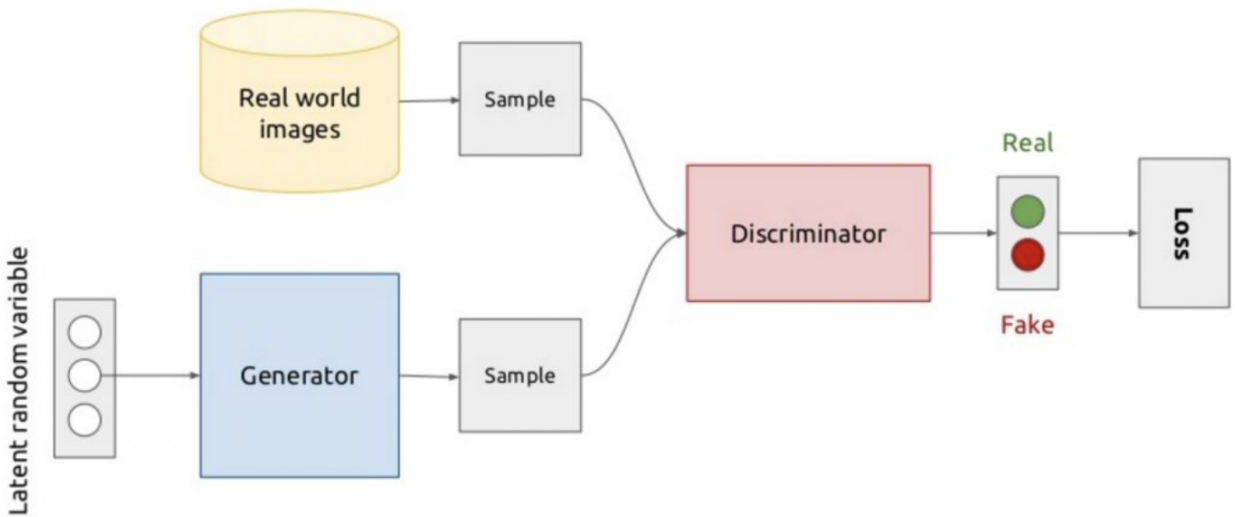
A useful minimax analogy for directly solving the data generation problem:

https://www.secretservice.gov/data/KnowYourMoney.pdf

Generative adversarial networks

From a deep learning perspective:

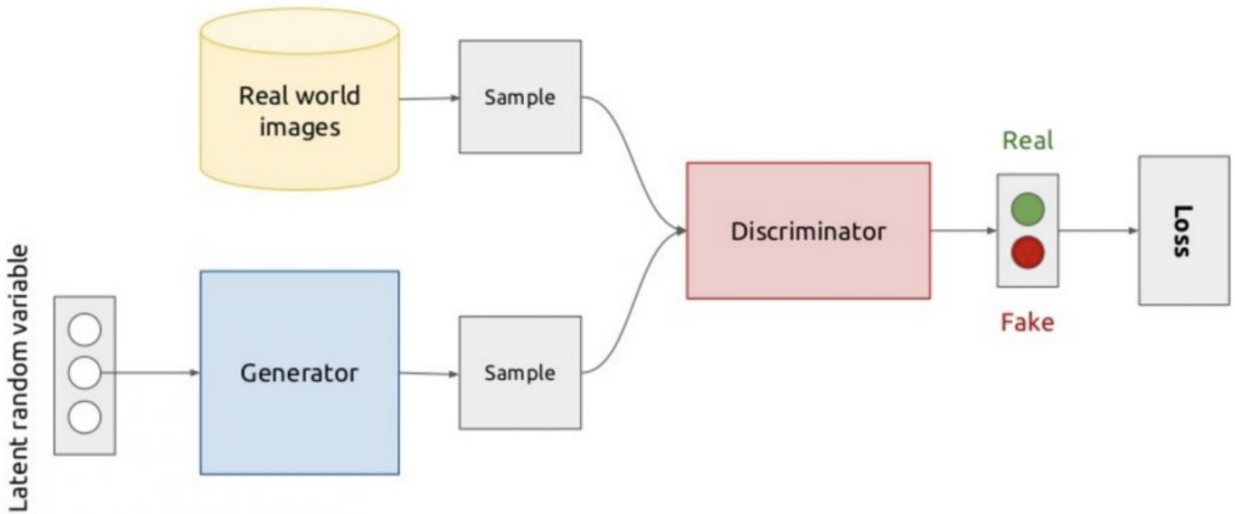
- True data samples $x_i^D \sim p_{data}(x)$... (minibatch) draws from the training set
- The latent code $z_i \sim \mathcal{N}(0, I)$
- The *generator* neural network $x_i^G = G_{\theta_G}(z_i)$
- The *discriminator* neural network $D_{\theta_D}(x_i) \to [0, 1]$



[image from http://cognitivechaos.com/understanding-generative-adversarial-networks/]

- The discriminator classifies fake vs real images
- The generator adapts to fool the discriminator
- This two-player game is repeated...

Generative adversarial networks



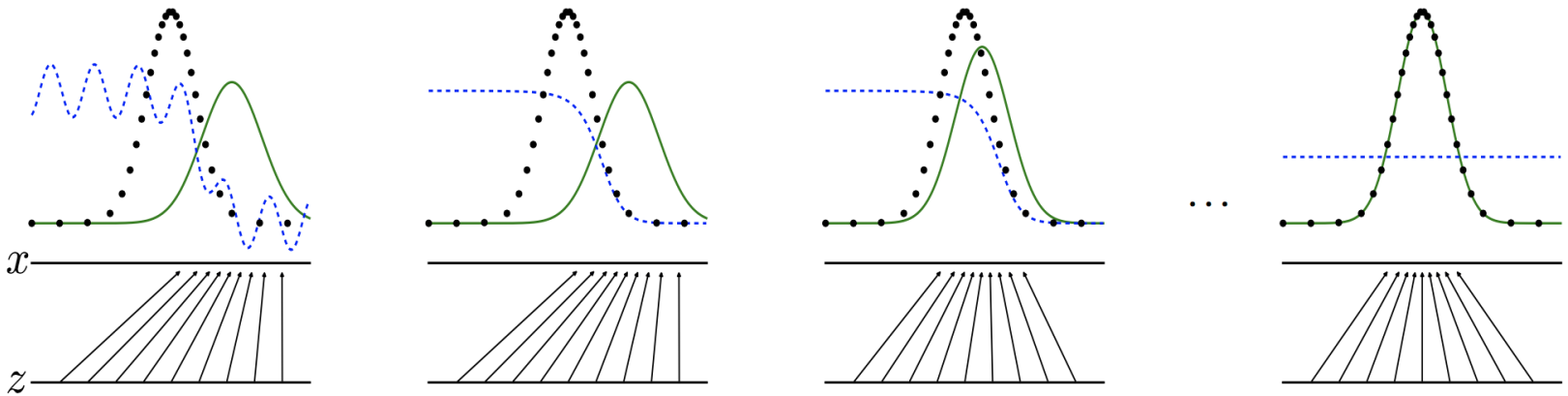
Specify the following objective:

$$\min_{\theta_G} \max_{\theta_D} \left[E_{x \sim p_{data}} \left(\log D_{\theta_D}(x) \right) + E_{z \sim p(z)} \left(\log \left(1 - D_{\theta_D} \left(G_{\theta_G}(z) \right) \right) \right) \right]$$

- $D_{\theta_d}(x_i)$ gives the probability ($\in [0, 1]$) that x_i^D is genuine (from data distribution)
- $1 - D_{\theta_D} \left(G_{\theta_G}(z_i) \right)$ gives the probability that x_i^G is *fake*
- $\min_{\theta_G}$ attempts to minimize the probability of being caught as a fake
- $\max_{\theta_D}$ attempts to maximize discriminability (reals $\uparrow$, fakes $\downarrow$)...

Generative adversarial networks

$$\min_{\theta_G} \max_{\theta_D} \left[E_{x \sim p_{data}} \left(\log D_{\theta_D}(x) \right) + E_{z \sim p(z)} \left(\log \left(1 - D_{\theta_D} \left(G_{\theta_G}(z) \right) \right) \right) \right]$$



Here:

- discriminator $D(x)$ (blue); generative $p_G(x)$ (green); true $p_{data}(x)$ (black)
- if arbitrarily expressive (2nd panel), $\max_D$ optimizes to $D_{\theta_D}(x) = \frac{p_{data}(x)}{p_{data}(x)+p_G(x)}$
- If everything works, eventually p_G is indistinguishable from p_{data}...

Note:

- do not take this objective/optima as absolute truth (Parts VI-IX)
- theory is starting to appear (Part IX)
- a taste: *mode collapse* and learning/generating the training set

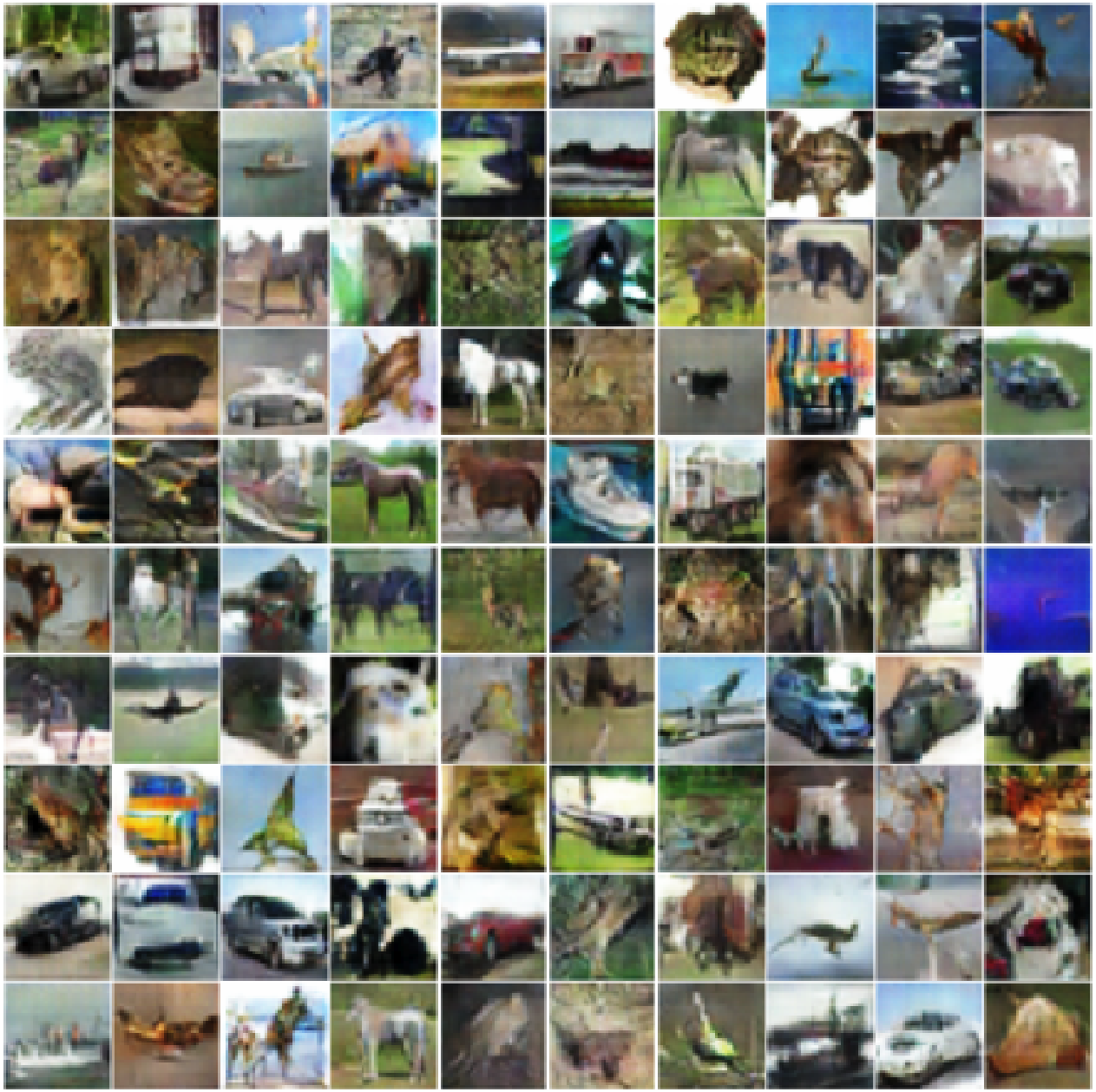
GAN in action

(right column images are nearest neighbor training points)

GAN in action

Is this good? What could we do with it if it were? (Part VIII)

GAN in action



Part II: adding (dynamical) structure

Welcome back

Reminder of our central theme:

> Use probabilistic models where we have inductive bias;
> Use flexible function approximators where we do not.

Time to add structure to DGM

- today: dynamical latent structure
- next time: discrete latent structure
- and more to come

But first... (with thanks for filling out the google form!)

- All set (mod admin): hml2134, tb2595, ta2507, sh3453, eg2912, ct2747, kv2294, hs2703, srb2201, jl4303, fg2425, gl2480, am4589, yt2541, yw2539, dz2336, wz2335
- Please talk to me after class: zw2501, lw2827, ekb2154, mn2822, ad3395, tg2632, cjc2197, az2522, bp2576
- Otherwise, thanks for your interest...

Criteria

Prerequisites

- Probabilistic machine learning $\supseteq$ Blei's PGM
- Deep learning $\supseteq$ Cunningham's AML
- Programming $\supseteq$ {python, tensorflow, pytorch, keras}

Attitude

- Excitement about this research area
- Desire to write a paper / work on a project in this domain
- Game to be lively and involved in seminar

Also

- Alignment with your research is terrific
- Diverse and nontraditional students welcome

Then again, maybe not

You aren't doing ML research already

- This course will be fast paced, Ph.D. level; prerequisites are real

You are just looking for course credit

- The format depends on the involvement and enthusiasm of a dedicated group

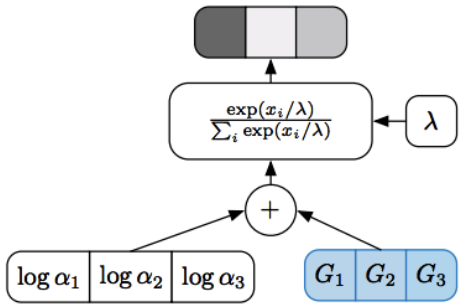
You are fully committed with an unrelated project

- This course will be a lot of work: reading, participation, lecturing, projects

Next Time

The next important area: adding discrete structure to DGM

- reparameterization trick does not work out of the box
- Gumbel-softmax, concrete, etc.
- neat and timely bag of tricks



Papers:

- The Concrete Distribution: A Continuous Relaxation of Discrete Random Variables

  Maddison, Mnih, Teh ICLR 2017; https://arxiv.org/abs/1611.00712

- Categorical Reparameterization with Gumbel-Softmax

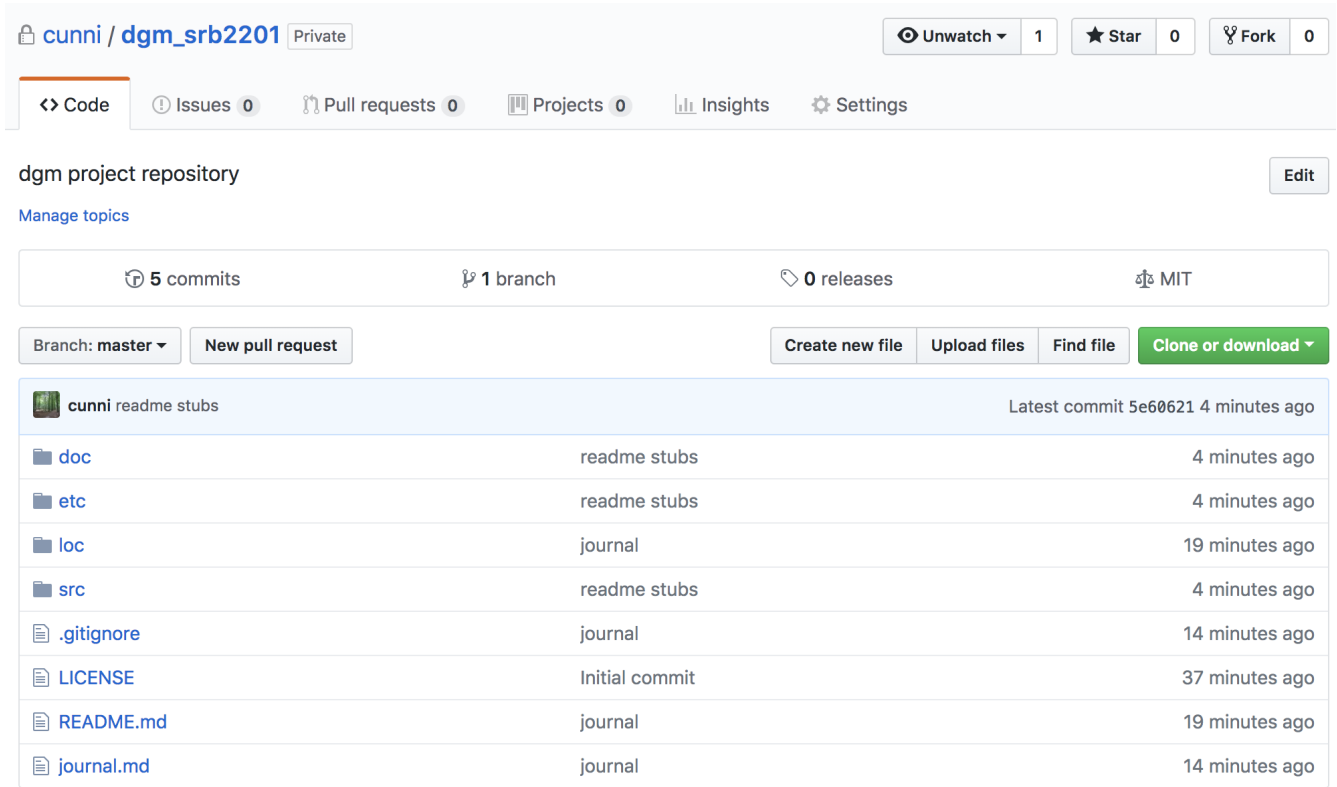
  Jang, Gu, Poole ICLR 2017; https://arxiv.org/abs/1611.01144

- optional: `https://arxiv.org/abs/1706.04161` , `https://arxiv.org/abs/1611.04051`

Presentation schedule:

- time to volunteer
- earliest is easiest...

Projects

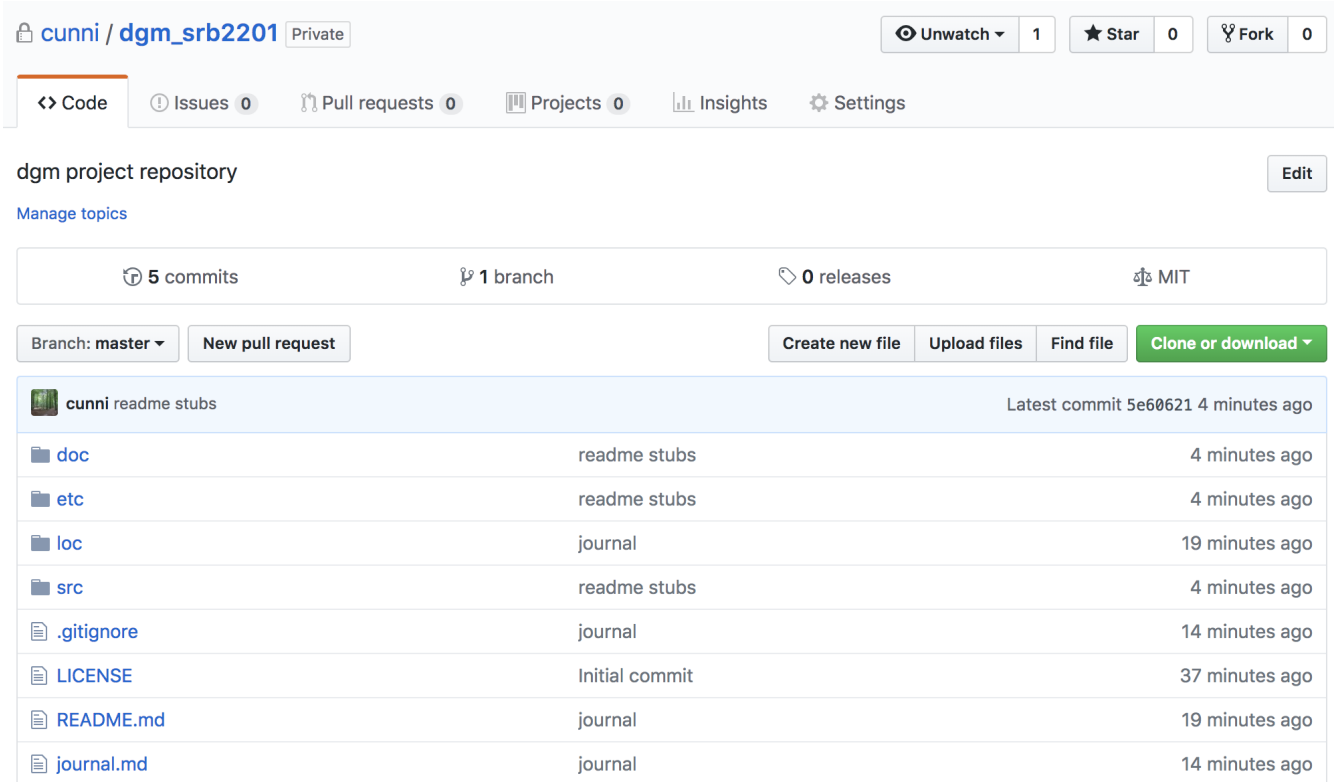


Your project repo (commit/push early and often)

- update `README.md` with project abstract (at first), and later with tutorial example
- update `journal.md` with a diary of your progress: dated entries with your activities, thoughts, findings, scientific notes. This is a lab notebook.
- update `journal.md` at least once a week

Projects

I need your github username (...uni/username sheet)



This week's assignment: project vision (in `README.md`)

- What area of DGM will your project approach?
- Assuming maximum success, what do you hope to find?
- ...initializer of an abstract

Projects

This week's assignment: project vision (in `readme.md`)

- What area of DGM will your project approach?
- Assuming maximum success, what do you hope to find?
- ...initializer of an abstract

Other items on their way...

- piazza form
- course website
- compute resources (?)

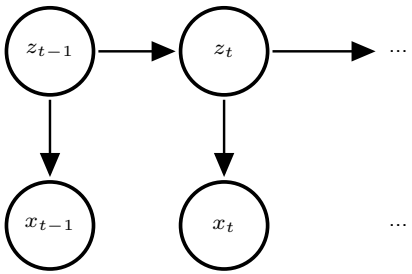
Planning purposes

- Who submitted to ICML?
- Who is going to AISTATS? COSYNE?
- Any other planned absenses? (Feb 20)

At last, content

Modeling time series data

One canonical model in time series analysis is the *linear dynamical system*:



$$\begin{aligned} p_\theta\left(z_t|z_{t-1}\right) &= \mathcal{N}\left(Az_{t-1}, Q\right) \\ p_\theta\left(x_t|z_t\right) &= \mathcal{N}\left(Cz_t, V\right) \end{aligned}$$

- an HMM in continuous latent space
- maintains joint gaussianity
- inference has linear runtime (Kalman filter, message passing, sum product, etc.)

[Kalman 1960]

This inference operation is likely convoluted or distant in your mind. Let's consider:

$$p_\theta(\mathbf{z}|\mathbf{x}) \propto \prod_{t=1}^{T} p_\theta(x_t|z_t)p_\theta(z_t|z_{t-1}) = \mathcal{N}(\mu, \Sigma)$$

$$\mathbf{z} = [z_1, ..., z_T]^\top \in \mathbb{R}^{dT} \qquad \mathbf{x} = [x_1, ..., x_T]^\top \in \mathbb{R}^{pT}$$

Reminder: exponential family representation

Considering this problem in its natural form clarifies all:

$$\mathcal{N}(\mu, \Sigma) \propto \exp\left\{-\frac{1}{2}(\mathbf{z}-\mu)^\top \Sigma^{-1}(\mathbf{z}-\mu)\right\} \propto \exp\left\{\begin{bmatrix}\Sigma^{-1}\mu \\ -\frac{1}{2}\Sigma^{-1}\end{bmatrix}^\top \begin{bmatrix} z \\ zz^\top \end{bmatrix}\right\} \triangleq \exp\left\{\begin{bmatrix} h \\ J \end{bmatrix}^\top \begin{bmatrix} z \\ zz^\top \end{bmatrix}\right\}$$

Now the LDS model:

$$\begin{aligned}
p_\theta(\mathbf{z}|\mathbf{x}) &\propto \prod_{t=1}^{T} p_\theta(x_t|z_t)p_\theta(z_t|z_{t-1}) \\
&= \prod_{t=1}^{T} \exp\left\{-\frac{1}{2}(x_t - Cz_t)^\top V^{-1}(x_t - Cz_t) - \frac{1}{2}(z_t - Az_{t-1})^\top Q^{-1}(z_t - Az_{t-1})\right\} \\
&\propto \prod_{t=1}^{T} \exp\left\{x_t^\top \left(V^{-1}C\right) z_t + z_t^\top \left(Q^{-1}A\right) z_{t-1} - \frac{1}{2} z_t^\top \left(Q^{-1} + A^\top Q^{-1} A + C^\top V^{-1} C\right) z_t\right\}
\end{aligned}$$

(with some laziness around $t = 0$)

So it is immediate that the natural parameter $\mathbf{h} = [h_1, ..., h_T]^\top$ has form:

$$h_t = C^\top V^{-1} x_t$$

Foreshadowing: this term *conditions on evidence in a conjugate form*.

Reminder: exponential family representation

...and the natural parameter J:

$$p_\theta(\mathbf{z}|\mathbf{x}) \propto \prod_{t=1}^{T} \exp\left\{x_t^\top \left(V^{-1}C\right) z_t + z_t^\top \left(Q^{-1}A\right) z_{t-1} - \frac{1}{2} z_t^\top \left(C^\top V^{-1} C + Q^{-1} + A^\top Q^{-1} A\right) z_t\right\}$$

$$-\frac{1}{2}J = \begin{bmatrix} D_0 & B_0^\top & & \\ B_0 & D_1 & B_1^\top & \\ & \ddots & \ddots & B_{T-1}^\top \\ & & B_{T-1} & D_T \end{bmatrix} \quad \text{where} \quad \begin{bmatrix} D_t = Q^{-1} + A^\top Q^{-1} A + C^\top V^{-1} C \\ B_t = -2Q^{-1}A \end{bmatrix}$$

So what?

- *block tridiagonal precision matrix*: $z_t z_{t-\tau}^\top$ terms $= 0 \; \forall \tau > 1$.
- Recover $\mu = -\frac{1}{2}J^{-1}h$ and $\Sigma = -\frac{1}{2}J^{-1}$
- Reparameterize $z^\ell = -\frac{1}{2}J^{-1}h + \left(-\frac{1}{2}J^{-1}\right)^{\frac{1}{2}} \epsilon^\ell$

Kalman filter/smoother

Still, so what...

$$-\frac{1}{2}J = \begin{bmatrix} D_0 & B_0^\top & & & \\ B_0 & D_1 & B_1^\top & & \\ & \ddots & \ddots & B_{T-1}^\top & \\ & & B_{T-1} & D_T \end{bmatrix} = \begin{bmatrix} L_0 & 0 & & & \\ C_0 & L_1 & 0 & & \\ & \ddots & \ddots & 0 & \\ & & C_{T-1} & L_T \end{bmatrix} \begin{bmatrix} L_0 & C_0^\top & & & \\ 0 & L_1 & C_1^\top & & \\ & \ddots & \ddots & C_{T-1}^\top & \\ & & 0 & L_T \end{bmatrix} \triangleq RR^\top$$

Iterative solve:

$$L_0 L_0^\top = D_0 \;\;,\;\; C_0 L_0^\top = B_0 \;\;,\;\; C_0 C_0^\top + L_1 L_1^\top = D_1 \;\;,\;\; \ldots \;\;,\;\; C_{T-1} C_{T-1}^\top + L_T L_T^\top = D_T$$

Fast Cholesky factorization (aka fwd/bk substitution, message passing, KF/KS, etc.)

`https://software.intel.com/en-us/node/531896`

And finally we recover either the reparameterization or the closed form posterior:

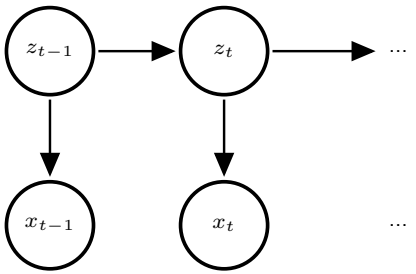
$$z^\ell = R^{-1} R^{-\top} h + R^{-\top} \epsilon^\ell$$

$$p_\theta(\mathbf{z}|\mathbf{x}) = \mathcal{N}\left(R^{-1} R^{-\top} h, R^{-1} R^{-\top}\right)$$

- All key operations are $\mathcal{O}(T)$ $\rightarrow$ for today let's call that a *fast PGMInference*

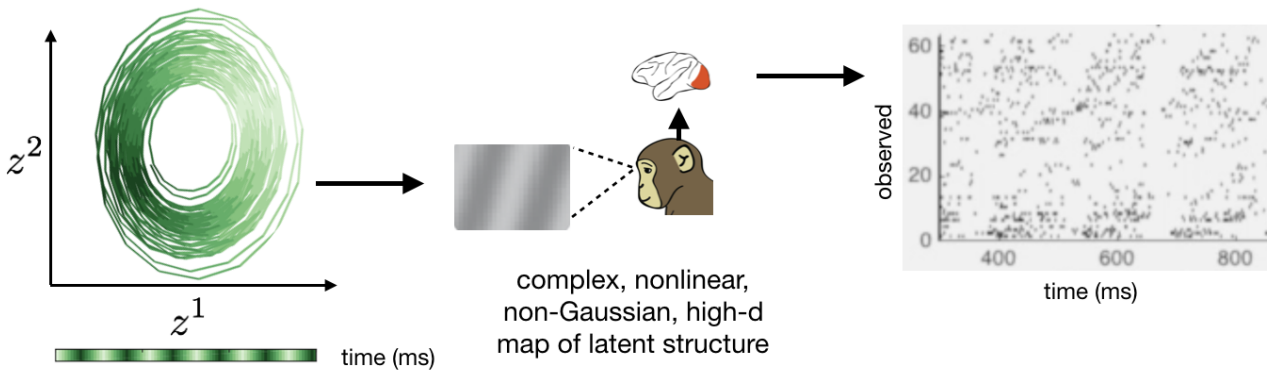
VAE with dynamical structure

Now suppose we have an intractable (but explicit) likelihood:



$$p_\theta(z_t|z_{t-1}) = \mathcal{N}(Az_{t-1}, Q)$$
$$p_\theta(x_t|z_t) = Poisson(f_\theta(z_t))$$

For example:



[Gao et al (2016) NIPS]

Historically:

- apply SMC/particle filtering to this model (slow)
- design a bespoke VI method

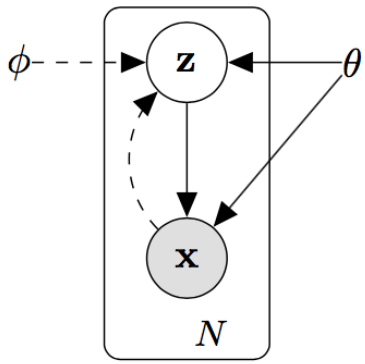
eg [Khan et al (2013) NIPS]

Today (and now, generally): use a VAE

VAE with dynamical structure

Again, the ELBO (a few extra views here):

$$\begin{aligned}\mathcal{L}(\phi,\theta) &= E_{q_\phi}\left(\log p_\theta(z,x)\right) - E_{q_\phi}\left(\log q_\phi(z|x)\right) \\ &= E_{q_\phi}\left(\log p_\theta(z,x)\right) + \mathbb{H}\left(q_\phi(z|x)\right) \\ &= E_{q_\phi}\left(\log p_\theta(x|z)\right) - KL\left(q_\phi(z|x)||p_\theta(z)\right)\end{aligned}$$



- Now suppose $\mathbf{z} \in \mathbb{R}^{dT}$ and $\mathbf{x} \in \mathbb{N}_+^{pT}$

As before, let us choose the default:

$$q_\phi(\mathbf{z}|\mathbf{x}) = \mathcal{N}\left(\mu_\phi(\mathbf{x}), \sigma_\phi^2(\mathbf{x})\right)$$

- now μ_ϕ is an MLP that maps $\mathcal{X} \to \mathbb{R}^d$
- but σ_ϕ is an MLP that maps $\mathcal{X} \to \mathbb{R}^{dT \times dT}$ $\to$ problems...
- we have lost linear in time scaling, making this model intractable in practice

VAE with dynamical structure

LDS, but with our new nonconjugate likelihood $p_\theta(x_t|z_t)$:

$$\begin{aligned} p_\theta(\mathbf{z}|\mathbf{x}) &\propto \prod_{t=1}^{T} p_\theta(x_t|z_t)p_\theta(z_t|z_{t-1}) \\ &\propto \prod_{t=1}^{T} \exp\left\{x_t^\top \left(V^{-1}C\right) z_t + z_t^\top \left(Q^{-1}A\right) z_{t-1} - \frac{1}{2} z_t^\top \left(Q^{-1} + A^\top Q^{-1} A + C^\top V^{-1} C\right) z_t\right\} \\ &\propto \prod_{t=1}^{T} \exp\left\{g_\theta(x_t, z_t) + z_t^\top \left(Q^{-1}A\right) z_{t-1} - \frac{1}{2} z_t^\top \left(Q^{-1} + A^\top Q^{-1} A + \text{???}\right) z_t\right\} \end{aligned}$$

Key idea: stipulate a variational family with LDS form:

$$q_\phi(\mathbf{z}) = \mathcal{N}(\mu, \Sigma) \propto \exp\left\{\begin{bmatrix} h_\phi(\mathbf{x}) \\ J_\phi(\mathbf{x}) \end{bmatrix}^\top \begin{bmatrix} \mathbf{z} \\ \mathbf{z}\mathbf{z}^\top \end{bmatrix}\right\}$$

Specifically, build an MLP into the block (banded) lower diagonal factor R:

$$-\frac{1}{2} J_\phi(\mathbf{x}) = \begin{bmatrix} L_\phi(x_1) & 0 & & \\ C_\phi(x_1, x_2) & L_\phi(x_2) & 0 & \\ & \ddots & \ddots & 0 \\ & & C_\phi(x_{T-1}, x_T) & L_\phi(x_T) \end{bmatrix} \begin{bmatrix} L_\phi & C_\phi^\top & & \\ 0 & L_\phi & C_\phi^\top & \\ & \ddots & \ddots & C_\phi^\top \\ & & 0 & L_\phi \end{bmatrix} \triangleq R_\phi(\mathbf{x}) R_\phi(\mathbf{x})^\top$$

Fast (approx) *PGMInference* results! [Archer et al (2016) ICLR]

$$q_\phi(\mathbf{z}|\mathbf{x}) = \mathcal{N}\left(R_\phi(\mathbf{x})^{-1} R_\phi(\mathbf{x})^{-\top} h_\phi(\mathbf{x}), R_\phi(\mathbf{x})^{-1} R_\phi(\mathbf{x})^{-\top}\right)$$

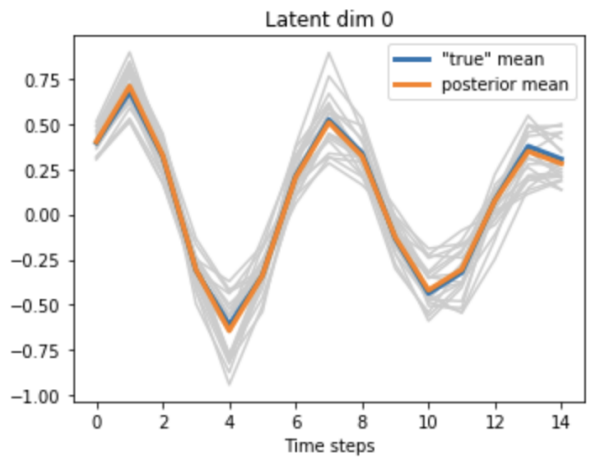
VAE with dynamical structure

Explicit coded example: `https://github.com/themattinthehatt/netlds`

```
200     def _build_posterior_mean(self):
201
202         ia = tf.reduce_sum(
203             tf.multiply(self.c_psi_inv,
204                         tf.expand_dims(self.m_psi, axis=2)),
205             axis=3)
206
207         # ia now S x T x dim_latent
208
209         # get posterior means for each element in batch
210         def scan_chol_inv(_, inputs):
211             """inputs refer to L/U matrices, outputs to means"""
212             [chol_decomp_Sinv_0, chol_decomp_Sinv_1, ia] = inputs
213             # mult by R
214             ib = blk_chol_inv(
215                 chol_decomp_Sinv_0, chol_decomp_Sinv_1, ia,
216                 lower=True, transpose=False)
217             post_z_means = blk_chol_inv(
218                 chol_decomp_Sinv_0, chol_decomp_Sinv_1, ib,
219                 lower=False, transpose=True)
220
221             return post_z_means
222
223         self.post_z_means = tf.scan(
224             fn=scan_chol_inv,
225             elems=[self.chol_decomp_Sinv[0], self.chol_decomp_Sinv[1], ia],
226             initializer=ia[0])  # throwaway to get scan to behave
227
```

In [10]:

```
# plot latents
plt.figure(figsize=(12, 4))
num_rows = 1
num_cols = dim_latent
num_samples = 20  # number of samples from fi
for l in range(dim_latent):
    plt.subplot(num_rows, num_cols, l + 1)
    for s in range(num_samples):
        plt.plot(posterior_means_rot[s, :, l]
    plt.plot(z_avg[:, l], linewidth=3, label=
    plt.plot(z_pred[:, l], linewidth=3, label
    plt.legend()
    plt.title('Latent dim %1i' % l)
    plt.xlabel('Time steps')
plt.show()
```



Notes

- I strongly recommend playing with this notebook and code (esp. `inference.py`)
- Thanks to Matt Whiteway for putting together the notebook for us

VAE WITH DYNAMICAL STRUCTURE

Is this exactly what I read in [Archer et al (2016) ICLR]?

4.1 DIAGONAL AND BLOCK OFF-DIAGONAL PARAMETERIZATION

We can naturally parameterize $\mu_\phi(\mathbf{x})$ and $\Sigma_\phi(\mathbf{x})$ of eq. 10 using 3 neural networks. We use one neural network to represent a map $\mathbf{x}_t \to \mu_t$,

$$\mu_t = \mathrm{NN}_{\phi_\mu}(\mathbf{x}_t), \tag{13}$$

where μ_t is a $n \times 1$ segment of μ, and $\mu = (\mu_1, \mu_2, \ldots, \mu_T)$. We can parameterize the block tri-diagonal covariance $\Sigma_\phi(\mathbf{x})^{-1}$,

$$\Sigma_\phi(\mathbf{x})^{-1} = \begin{bmatrix} D_0 & B_0^{\mathrm{T}} & & \\ B_0 & D_1 & B_1^{\mathrm{T}} & \\ & \ddots & \ddots & B_{T-1}^{\mathrm{T}} \\ & & B_{T-1} & D_T \end{bmatrix}, \tag{14}$$

by parmeterizing each of the blocks separately:

$$D_t = \mathrm{NN}_{\phi_D}(\mathbf{x}_t) \tag{15}$$
$$B_t = \mathrm{NN}_{\phi_B}(\mathbf{x}_t, \mathbf{x}_{t-1}). \tag{16}$$

- ambiguity: is μ a direct map (and thus not really a LDS posterior $q_\phi(\mathbf{z}|\mathbf{x})$?
- potential PSD violations in Σ; avoided by directly parameterizing L, C, as we did.

VAE WITH DYNAMICAL STRUCTURE

Is this exactly what I read in [Archer et al (2016) ICLR]?

4.2 PRODUCT-OF-GAUSSIANS APPROXIMATE POSTERIOR

We can also define the approximate posterior through a product of Gaussian factors, $q(\mathbf{z}|\mathbf{x}) \propto r_1(\mathbf{z}|\mathbf{x})r_0(\mathbf{z})$, where:

$$r_0(\mathbf{z}) := \mathcal{N}(\mathbf{z}|0, \mathbf{D}) \quad (17)$$

$$r_1(\mathbf{z}|\mathbf{x}) := \mathcal{N}(\mathbf{z}|\mathbf{M}_\phi(\mathbf{x}), \mathbf{C}_\phi(\mathbf{x})), \quad (18)$$

$\mathbf{D}$ and $\mathbf{C}$ are $nT \times nT$ matrices and $\mathbf{M}$ is a nT-dimensional vector. In this set-up, we can view r_0 as a prior. In terms of eq. 10, the final posterior is then given by:

$$\Sigma_\phi(\mathbf{x}) = \left(\mathbf{D}^{-1} + \mathbf{C}_\phi^{-1}(\mathbf{x})\right)^{-1} \quad (19)$$

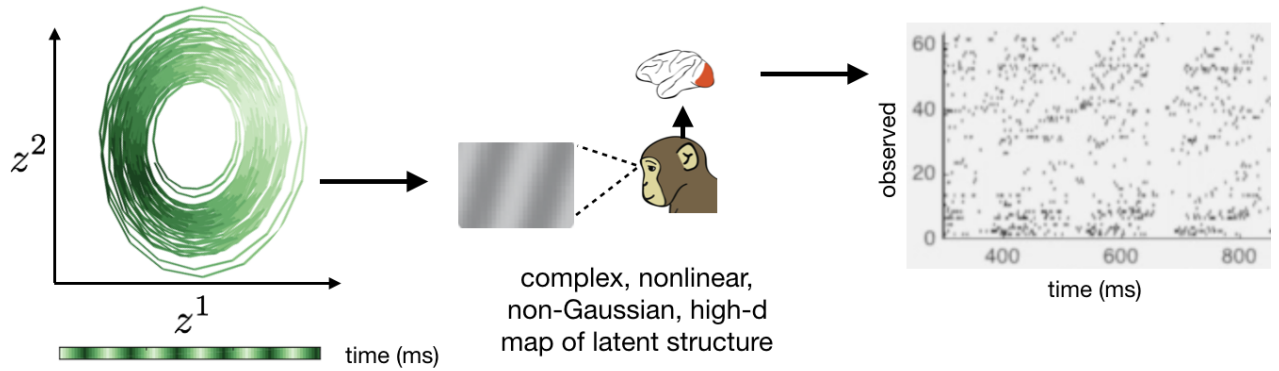
$$\mu_\phi(\mathbf{x}) = \Sigma_\phi(\mathbf{x})\mathbf{C}_\phi^{-1}(\mathbf{x})\mathbf{M}_\phi(\mathbf{x}). \quad (20)$$

In order to be a parameterization of the smoothing posterior, eq. 10, $\mathbf{D}^{-1}$ and $\mathbf{C}^{-1}$ must be block tri-diagonal. The multiplicative interaction between the posterior mean and covariance leads to different performance from the parameterization described in Section 4.1. Further, we can choose $\mathbf{D}$ to initialize the means with a given degree of smoothness, which is not possible in the formulation of Section 4.1. In the experiments, we refer to this parameterization as VILDSmult; in Appendix A we describe the specific parameteriation we used for $\mathbf{C}^{-1}$ and $\mathbf{D}^{-1}$.

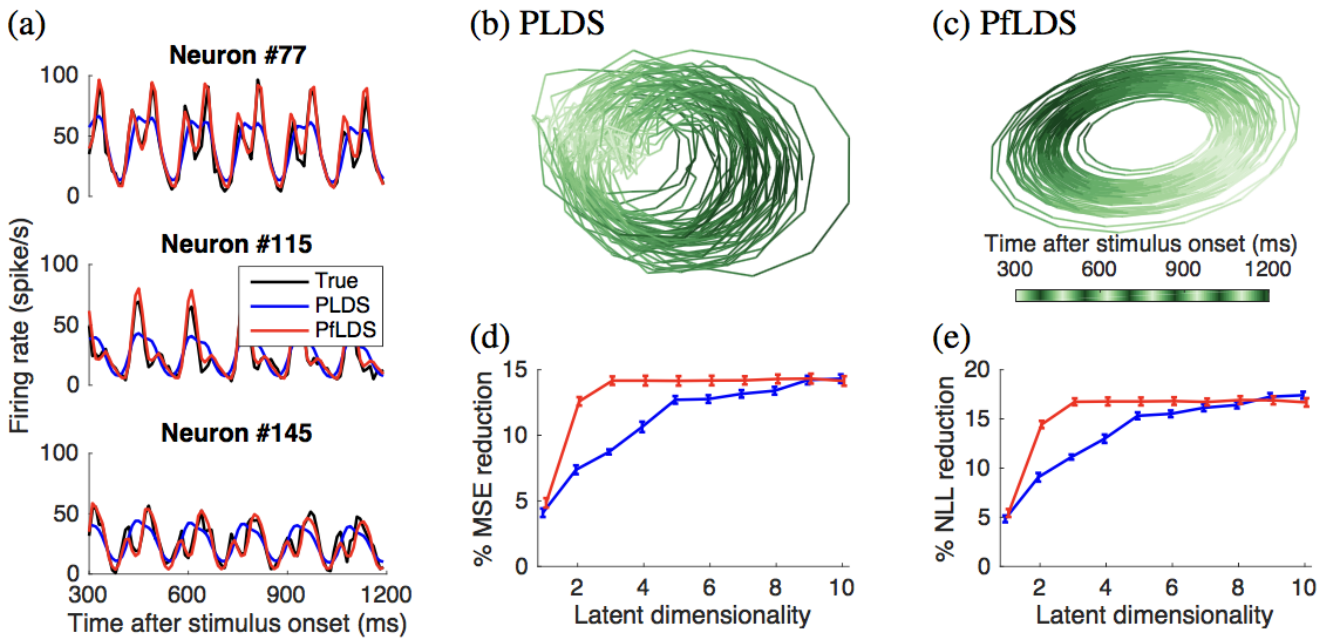
- cf. the usual exponential family $\mu = -\frac{1}{2}J^{-1}h$ and $\Sigma = -\frac{1}{2}J^{-1}$
- strict LDS q_ϕ is not fundamentally "correct" $\rightarrow$ tweaks are at our discretion.
- these days: what we derived == [Archer et al (2016) ICLR], [Gao et al (2016) NIPS]

VAE with dynamical structure

Reminder: example problem setup



Results:



[Gao et al (2016) NIPS]

Have we done something more general here?

Stepping back:

- We had a nasty nonconjugate model, and a vanilla VAE would not scale
- Special structure allowed fast PGMInference, if only we had conjugacy...
- We shoehorned our model by *learning to condition on evidence in conjugate form*

...encoder network into the natural parameters $[J_\phi(x), h_\phi(x)] \triangleq r_\phi(x)$

Is this a more general concept? Apparently lots of models have special PGMInference:

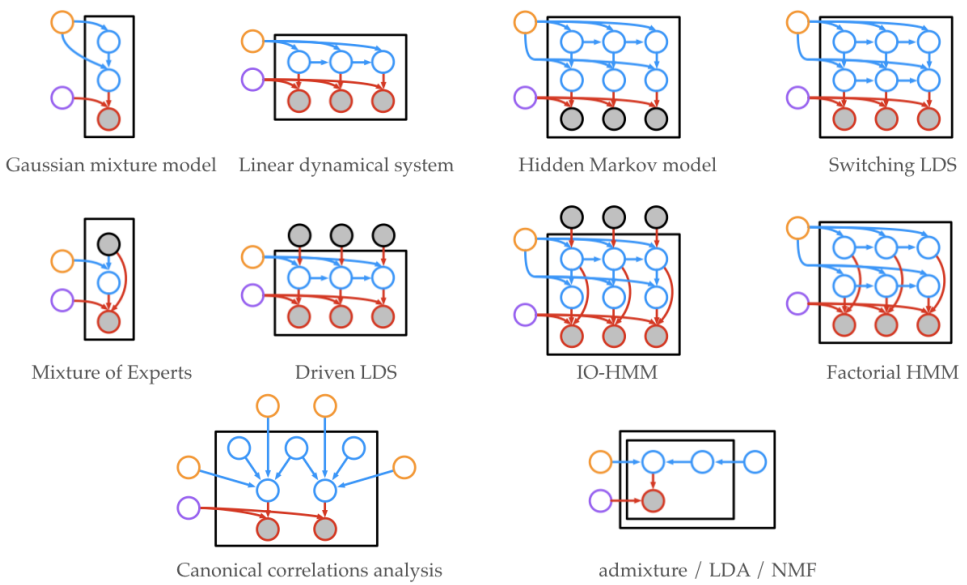


image credit: Ryan Adams

We now generalize in this way (and add variational distributions over parameters)

[Johnson et al (2016) NIPS]

Sidebar: how to read and present papers

First inspection suggests [Archer et al (2015) ICLR, Johnson et al (2016) NIPS]:

- are quite similar in spirit/motivation
- are quite different in execution
- Is that true?

Reality:

- the core approach is nearly (exactly?) the same
- the latter generalizes the former in a great PGM way
- the latter generalizes the former in a nice "fully Bayesian" way

  $p(\theta)$... recall ELBO gap!
- $p(\theta)$ complicates the structure, making connections less clear

Question for discussion:

- is [Archer et al (2016) ICLR] *exactly* a special case of [Johnson et al (2016) NIPS]?
- let's step back to build up from where we are...

Generalizing VAE with (dynamical) structure

ELBO:

$$\begin{aligned}\mathcal{L}(\phi,\theta) &= E_{q_\phi}\left(\log p_\theta(z,x)\right) - E_{q_\phi}\left(\log q_\phi(z|x)\right) \\ &= E_{q_\phi}\left(\log p_\theta(z,x)\right) + \mathbb{H}\left(q_\phi(z|x)\right) \\ &= E_{q_\phi}\left(\log p_\theta(z|x)\right) - KL\left(q_\phi(z|x)||p_\theta(z)\right)\end{aligned}$$

- choose $q_\phi(z|x) \propto \exp\left\{r_\phi(x)^\top t_z(z)\right\}$
- conjugate, (by definition) a fast *PGMInference* $\rightarrow$ easy sampling, local KL
- we just did this: $[J_\phi(x), h_\phi(x)] \triangleq r_\phi(x)$

Apparently we now have everything we need (eqn at bottom of p6, absent $p(\theta), p(\gamma)$):

$$\mathcal{L}_{SVAE}(\phi,\theta) = E_{q_\phi}\left(\log p_\theta(x|z)\right) - KL\left(q_\phi(z|x)||p_\theta(z)\right)$$

> *"By optimizing over ϕ we are effectively learning how to condition on observations so as to best approximate the posterior while maintaining conjugacy structure."*

Technical note: $r_\phi(x)$ may be the full posterior natural parameters, or just those associated with the likelihood

Generalizing VAE with (dynamical) structure

Now let's go (more) fully Bayesian:

$$\begin{aligned}\mathcal{L}_{SVAE}(\eta_\theta, \eta_\gamma, \eta_z) &= E_{q(\theta)q(\gamma)q(z)}\left(\log \frac{p(\theta)p(\gamma)p(z|\theta)p(x|z,\gamma)}{q(\theta)q(\gamma)q(z)}\right) \\ &= E_{q(\gamma)q(z)}\left(\log p(x|z,\gamma)\right) - KL\left(q(\theta)q(z)||p(\theta,z)\right) - KL\left(q(\gamma)||p(\gamma)\right)\end{aligned}$$

This should mitigate the ELBO gap issue (I hope?)

Approximating $p(x|z,\gamma) \approx \frac{1}{Z}\exp\left\{r_\phi(x)^\top t_z(x)\right\}$:

- above is still mean field in the product space $\Theta \times \Gamma \times \Phi$
- Great, but to exploit PGMInference object, we do this with a fixed $q(\theta) \to \eta_\theta$
- Then can evaluate and take gradients as before...

Generalizing VAE with (dynamical) structure

$$\mathcal{L}_{SVAE}(\eta_\theta, \eta_\gamma, \eta_z) \quad = \quad E_{q(\gamma)q(z)}\left(\log p(x|z,\gamma)\right) - KL\left(q(\theta)q(z)||p(\theta,z)\right) - KL\left(q(\gamma)||p(\gamma)\right)$$

Algorithm 1 Estimate SVAE lower bound and its gradients

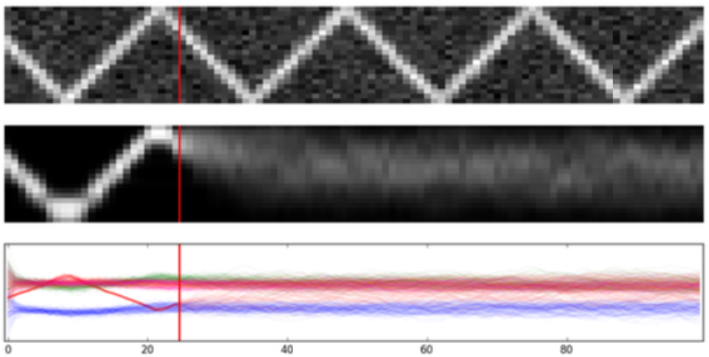
Input: Variational parameters $(\eta_\theta, \eta_\gamma, \phi)$, data sample y
  function SVAEGradients(η_θ, η_γ, ϕ, y)
    $\psi \leftarrow r(y_n; \phi)$ ▷ Get evidence potentials
    $(\hat{x},\ \bar{t}_x,\ \mathrm{KL}^{\mathrm{local}}) \leftarrow$ PGMInference(η_θ, ψ) ▷ Combine evidence with prior
    $\hat{\gamma} \sim q(\gamma)$ ▷ Sample observation parameters
    $\mathcal{L} \leftarrow N \log p(y \mid \hat{x}, \hat{\gamma}) - N\,\mathrm{KL}^{\mathrm{local}} - \mathrm{KL}(q(\theta)q(\gamma)\|p(\theta)p(\gamma))$ ▷ Estimate variational bound
    $\widetilde{\nabla}_{\eta_\theta}\mathcal{L} \leftarrow \eta_\theta^0 - \eta_\theta + N(\bar{t}_x, 1) + N(\nabla_{\eta_x} \log p(y \mid \hat{x}, \hat{\gamma}), 0)$ ▷ Compute natural gradient
    return lower bound $\mathcal{L}$, natural gradient $\widetilde{\nabla}_{\eta_\theta}\mathcal{L}$, gradients $\nabla_{\eta_\gamma,\phi}\mathcal{L}$
  function PGMInference(η_θ, ψ)
    $q^*(x) \leftarrow$ OptimizeLocalFactors(η_θ, ψ) ▷ Fast message-passing inference
    return sample $\hat{x} \sim q^*(x)$, statistics $\mathbb{E}_{q^*(x)} t_x(x)$, divergence $\mathbb{E}_{q(\theta)} \mathrm{KL}(q^*(x)\|p(x \mid \theta))$

- notation change $x \to z$, $y \to x$ (inevitable...)
- don't get distracted by natural gradients (or, do you want to?)
- first two lines in language of [Archer et al (2015) ICLR]:
  1. compute $[J_\phi(\mathbf{x}), h_\phi(\mathbf{x})]$
  2. sample $z^\ell = R^{-1}R^{-\top}h + R^{-\top}\epsilon^\ell$ (or get local KL, etc...)

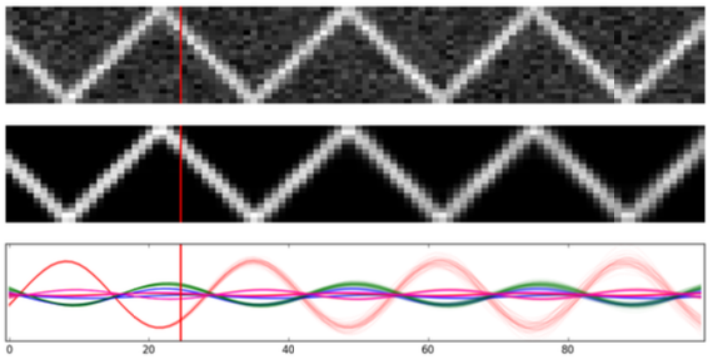
Summary: this is a nontrivial generalization, along two axes.

Some results

Apply this to the same LDS problem:



(a) Predictions after 200 training steps.



(b) Predictions after 1100 training steps.

Now a switching LDS with depth video (of mice):

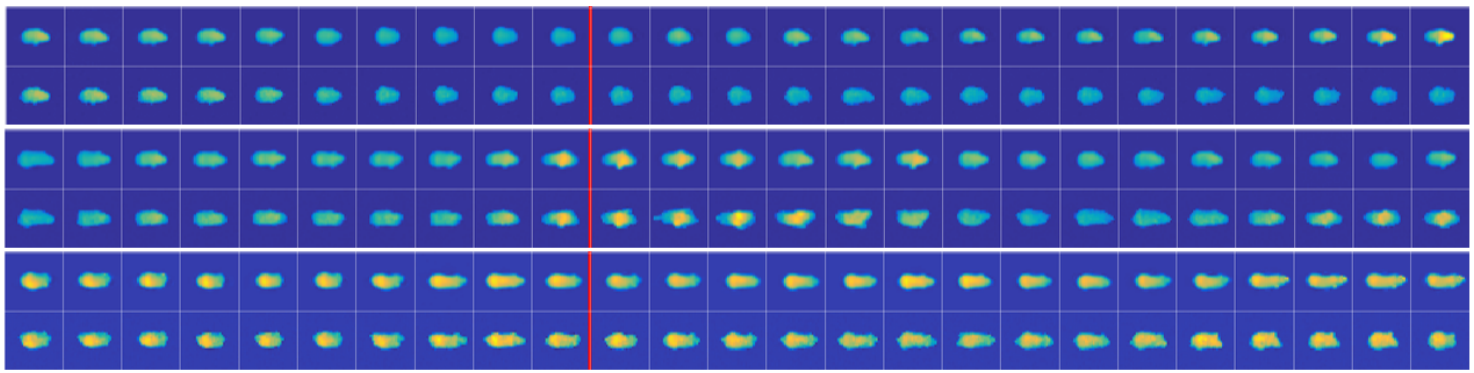
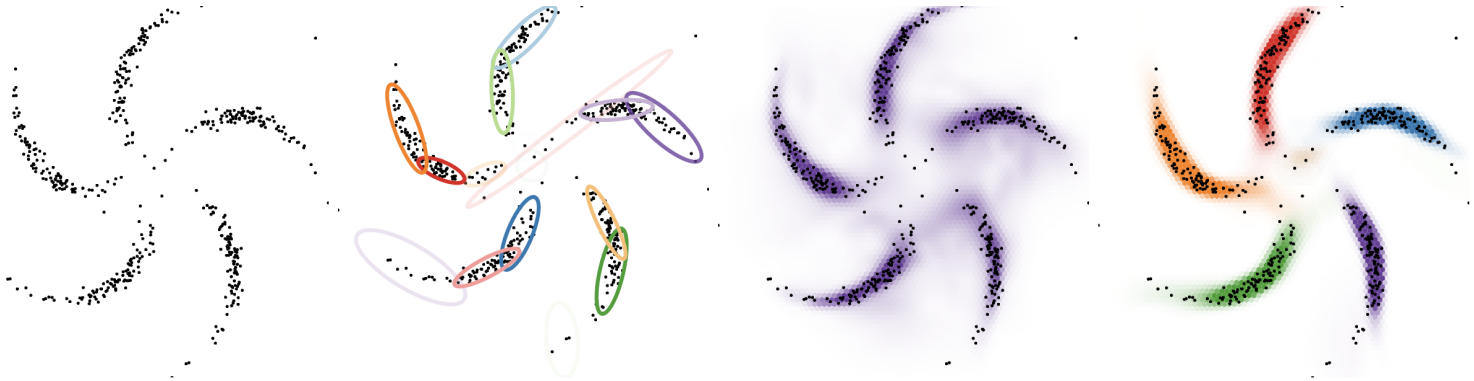


Figure 6: Predictions from an LDS SVAE fit to depth video. In each panel, the top is a sampled prediction and the bottom is real data. The model is conditioned on observations to the left of the line.

see also [Wiltschko et al (2015) Neuron]

Some results

Should now be easy (conceptually) to extend to other PGMInference models:



(a) Data (b) GMM (c) Density net (VAE) (d) GMM SVAE

Parting thoughts:

- application to neural data analysis: [Gao et al (2016) NIPS]
- nonlinear / RNN dynamics: [Krishnan et al (2016) arXiv]
- switching (and recurrent) LDS: [Johnson et al (2016) NIPS, Linderman et al (2017) AISTATS]
- spatially dependent linear dynamics: [Hernandez et al (2017) NIPS TSW]
- ...

Next: let's add different sorts of structure...

Part III: adding (discrete) structure

References

References

[APB+15] Evan Archer, Il Memming Park, Lars Buesing, John Cunningham, and Liam Paninski.
Black box variational inference for state space models.
11 2015.

[GAPC16] Yuanjun Gao, Evan W Archer, Liam Paninski, and John P Cunningham.
Linear dynamical neural population models through nonlinear embeddings.
In *Advances in neural information processing systems*, pages 163–171, 2016.

[GPAM+14] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio.
Generative adversarial nets.
In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[JDW+16] Matthew Johnson, David K Duvenaud, Alex Wiltschko, Ryan P Adams, and Sandeep R Datta.
Composing graphical models with neural networks for structured representations and fast inference.
In *Advances in neural information processing systems*, pages 2946–2954, 2016.

[KHL16] Matt J Kusner and José Miguel Hernández-Lobato.
Gans for sequences of discrete elements with the gumbel-softmax distribution.
arXiv preprint arXiv:1611.04051, 2016.

[KSS15] Rahul G Krishnan, Uri Shalit, and David Sontag.
Deep kalman filters.
arXiv preprint arXiv:1511.05121, 2015.

[KW13] Diederik P Kingma and Max Welling.
Auto-encoding variational bayes.
arXiv preprint arXiv:1312.6114, 2013.

[ML16] Shakir Mohamed and Balaji Lakshminarayanan.
Learning in implicit generative models.
arXiv preprint arXiv:1610.03483, 2016.

[RMW14] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra.
Stochastic backpropagation and approximate inference in deep generative models.
arXiv preprint arXiv:1401.4082, 2014.

[TRB17] Dustin Tran, Rajesh Ranganath, and David Blei.
Hierarchical implicit models and likelihood-free variational inference.
In *Advances in Neural Information Processing Systems*, pages 5523–5533, 2017.