CSC412: Stochastic Variational Inference

David Duvenaud

Admin

- A3 will be released this week and will be shorter
- Motivation for REINFORCE
- Class projects

Class Project ideas

- **Develop a generative model for a new medium**.
  - Generate sound given video (hard to generate raw sound)
  - Automatic onomatopoeia: Generate text ‘ka-bloom-kshhhh’ given a sound of an explosion.
  - Generating text of a specific style. For instance, generating SMILES strings representing organic molecules
  - Emoji2Vec
  - Automatic data cleaning (flagging suspect entries)

Class Projects

- **Extend existing models, inference, or training**. For instance:

    - Extending variational autoencoders to have infinite capacity in some sense (combining Nonparametric Bayesian methods with variational autoencoders)

    - Explore the use of mixture distributions for approximating distributions

Class Projects

- **Review / comparison / tutorials:**
  - Approaches to generating images
  - Approaches to generating video
  - Approaches to handling discrete latent variables
  - Approaches to building invertible yet general transformations
  - Variants of the GAN training objective
  - Different types of recognition networks
- clearly articulate the differences between different approaches, and their strengths and weaknesses.
- Ideally, include experiments highlighting the different properties of each method on realistic problems.

Class Projects

- **Demos**
    - http://distill.pub/
    - https://chi-feng.github.io/mcmc-demo/
- E.g. javascript demo of variational inference
- Still needs report, but doesn't need to be novel

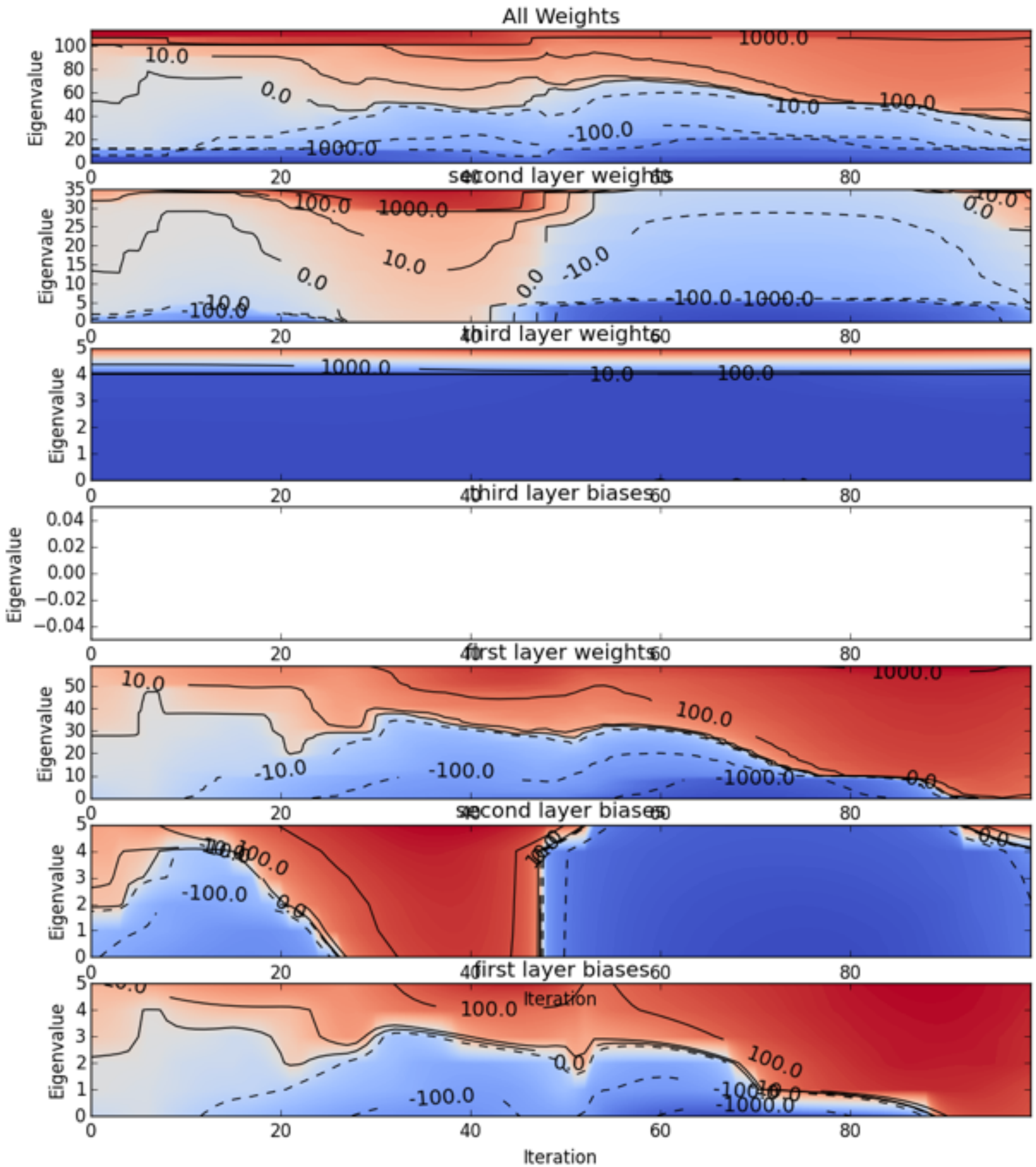
Hard Class Projects

- Graph-valued latent variable models (attend, infer, repeat) https://arxiv.org/abs/1603.08575

- Eigenscapes - characterize the loss surface of neural nets / GANs through the eigenspectrum of the Hessian of the loss

- HMC Recognition networks with accept/reject (http://jmlr.org/proceedings/papers/v37/salimans15.pdf)

- Simultaneous localization and mapping (SLAM) from scratch


All Weights
second layer weights
third layer weights
third layer biases
first layer weights
second layer biases
first layer biases
Eigenvalue
Iteration

- Many situations where we want to estimate or sample from an unnormalized distribution
- MCMC is always available, but
  - Guarantees are only asymptotic
  - Hard to know how well it’s doing
  - Hard to tune hyperparameters
  - Gradient-free MCMC hard to get to work in high dimensions

Gradient-based MCMC

- Gradient-based MCMC (Hamiltonian Monte Carlo, Langevin dynamics) scale to high dimension.
- These look like SGD with noise, or SGD with momentum with noise.
- Fairly effective (see Stan)
- But we have better optimizers (e.g. Adam, Quasi-Newton methods) that we don’t know how to use.

Variational Inference

- Directly optimize the parameters of an approximate distribution $q(z|x)$ to match $p(z|x)$
- Main technical difficulty:
  - Need to measure difference between $q(z|x)$ and $p(z|x)$ (and its gradient) using only cheap operations.
- By assumption, we can't sample from $p(z|x)$ or evaluate its density. We can:
  - Evaluate density $p(x, z)$ aka unnormalized $p(z|x)$
  - Sample from $q(z|x)$ and evaluate its density

Which divergence to use?

Name	$D_f(P\|Q)$
Total variation	$\frac{1}{2}\int \lvert p(x) - q(x)\rvert \, \mathrm{d}x$
Kullback-Leibler	$\int p(x) \log \frac{p(x)}{q(x)} \, \mathrm{d}x$
Reverse Kullback-Leibler	$\int q(x) \log \frac{q(x)}{p(x)} \, \mathrm{d}x$
Pearson χ^2	$\int \frac{(q(x)-p(x))^2}{p(x)} \, \mathrm{d}x$
Neyman χ^2	$\int \frac{(p(x)-q(x))^2}{q(x)} \, \mathrm{d}x$
Squared Hellinger	$\int \left(\sqrt{p(x)} - \sqrt{q(x)}\right)^2 \, \mathrm{d}x$
Jeffrey	$\int (p(x) - q(x)) \log \left(\frac{p(x)}{q(x)}\right) \, \mathrm{d}x$
Jensen-Shannon	$\frac{1}{2}\int p(x) \log \frac{2p(x)}{p(x)+q(x)} + q(x) \log \frac{2q(x)}{p(x)+q(x)} \, \mathrm{d}x$
Jensen-Shannon-weighted	$\int p(x)\pi \log \frac{p(x)}{\pi p(x)+(1-\pi)q(x)} + (1-\pi)q(x) \log \frac{q(x)}{\pi p(x)+(1-\pi)q(x)} \, \mathrm{d}x$
GAN	$\int p(x) \log \frac{2p(x)}{p(x)+q(x)} + q(x) \log \frac{2q(x)}{p(x)+q(x)} \, \mathrm{d}x - \log(4)$
α-divergence ($\alpha \notin \{0, 1\}$)	$\frac{1}{\alpha(\alpha-1)} \int \left(p(x) \left[\left(\frac{q(x)}{p(x)}\right)^\alpha - 1\right] - \alpha(q(x) - p(x))\right) \mathrm{d}x$

- From Nowozin et al, 2016, f-GANs

Why we like KL(q||p)

- Can get unbiased estimate using only samples from q(z|x) and evaluations of q(z|x) and p(z,x)
- Minimizing this maximizes a lower bound on marginal likelihood (good for model comparison)
- An aside: Upper bounds on marginal likelihood are hard in general
- Can use simple Monte Carlo (hence <u>stochastic</u>)

Algorithm:

- 1. Sample z from q(z|x, phi)
- 2. Return log p(z, x | theta) - log q(z | x, phi)

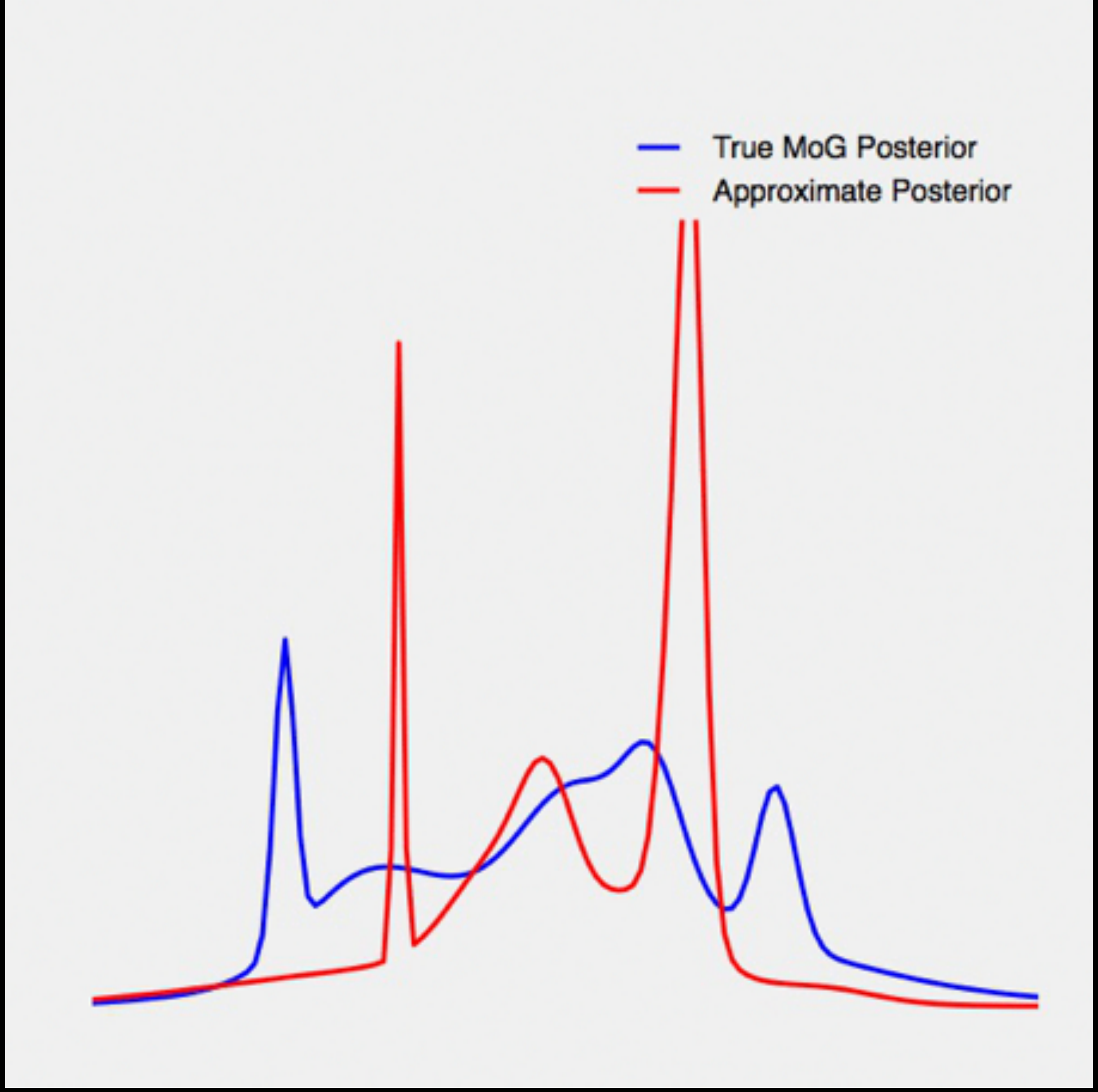
- That's it! Can optimize theta and phi using automatic differentiation if z is continuous, and dependence on phi is exposed

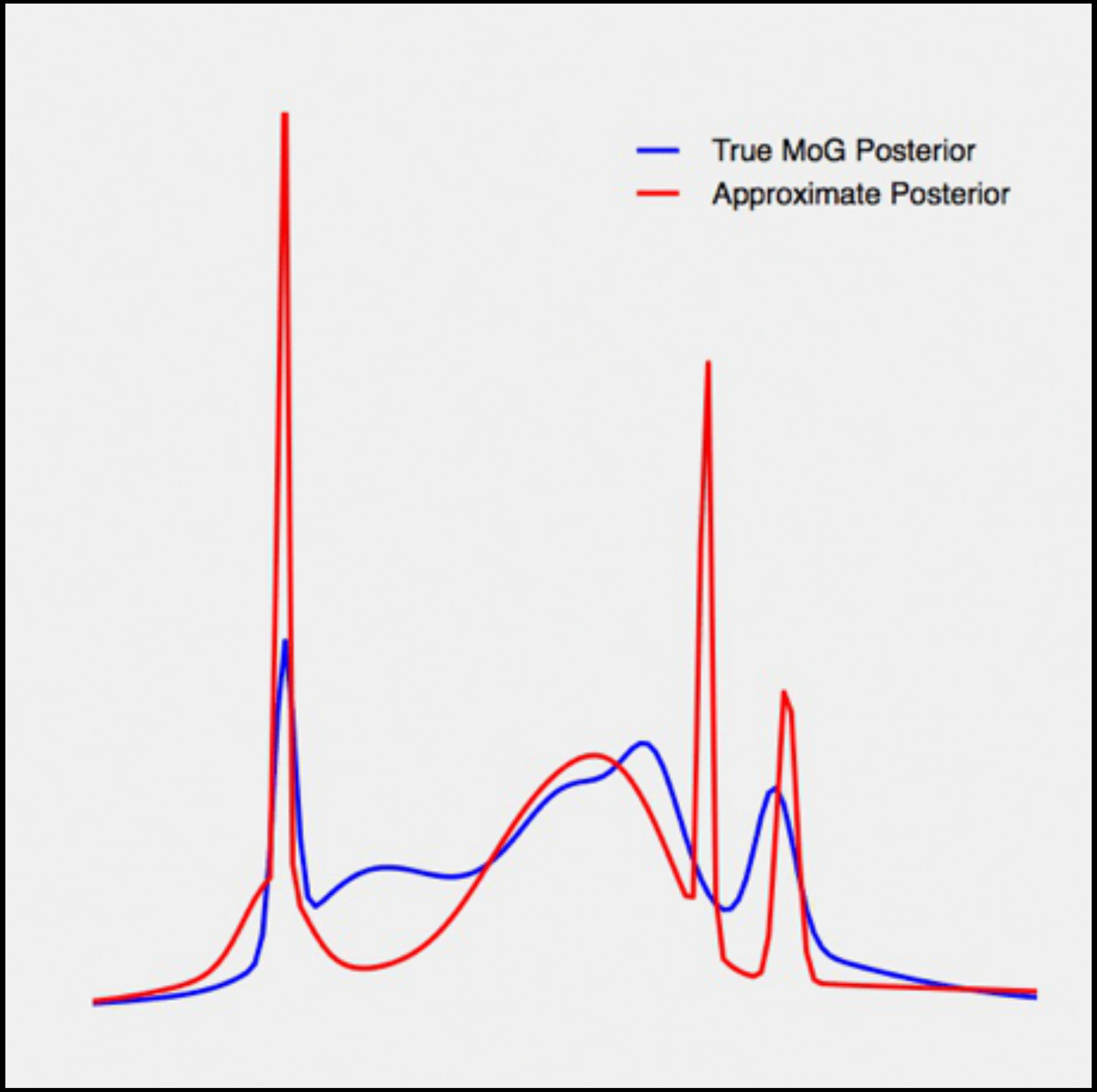
Three forms of bound

$$
\begin{aligned}
\mathcal{L}(\phi) &= \mathbb{E}_{\mathbf{z}\sim q}[\log p(x|z) + \log p(z) - \log q_\phi(z|x)] && (1) \\
&= \mathbb{E}_{\mathbf{z}\sim q}[\log p(x|z) + \log p(z))] + \mathbb{H}[q_\phi] && (2) \\
&= \mathbb{E}_{\mathbf{z}\sim q}[\log p(x|z)] - KL(q_\phi(z|x)||p(z)) && (3)
\end{aligned}
$$

- Each has its pros and cons
- First is most general
- First one can have zero variance

Hot off the press





- Roeder, Wu, Duvenaud

ADVI in 5 Lines of Python

Ryan Adams @ryan_p_adams · 7 Nov 2015

@DavidDuvenaud

```
def elbo(p, lp, D, N):
 v=exp(p[D:])
 s=randn(N,D)*sqrt(v)+p[:D]
 return mvn.entropy(0, diag(v))+mean(lp(s))
gf = grad(elbo)
```

1 7 22

What are we optimizing?

- Variational parameters phi specify q(z|x, phi)
- Simplest example: q(z|x, phi) =

Simple but not obvious

- It took a long time get here!
  - Reparameterization trick vs REINFORCE
  - Automatic differentiation vs local updates
  - Simple Gaussian vs optimal variational family

Other Formerly Promising Directions

- Expectation Propagation (reverse KL with local updates)
- local biased gradient estimation,
- Laplace approximation, UKF, etc

Code examples

- Show ADVI code
- Show Bayesian neural net example

Pros and Cons vs HMC

- Both are applicable out-of-the-box for almost any continuous latent variable model
- HMC is asymptotically exact, which makes statisticians happy (but it shouldn't…)
- ADVI with simple form (Gaussian) underestimates posterior variance
  - So does un-mixed HMC
  - Can make approximate posterior more and more complex (show mixture example)
- Biggest pro in my opinion - can measure inference progress, and use fancy optimization methods (e.g. Adam)

Recent Extensions

- Importance-Weighted Autoencoders (IWAE) Burda, Grosse, Salakhudtinov
- Mixture distributions in posterior
- GAN-style ideas to avoid evaluating $q(z|x)$
- Normalizing flows: Produce arbitrarily-complicated $q(z|x)$
- Incorporate HMC or local optimization to define $q(z|x)$

Next part: Variational Autoencoders

- Haven’t talked about learning theta
- Haven’t talked about having a latent variable per-datapoint