Hierarchical Variational Models

Rajesh Ranganath, Dustin Tran, David M. Blei

December 11, 2015

Review: Variational Inference

Goal: Fit a distribution to the posterior with optimization

Model:

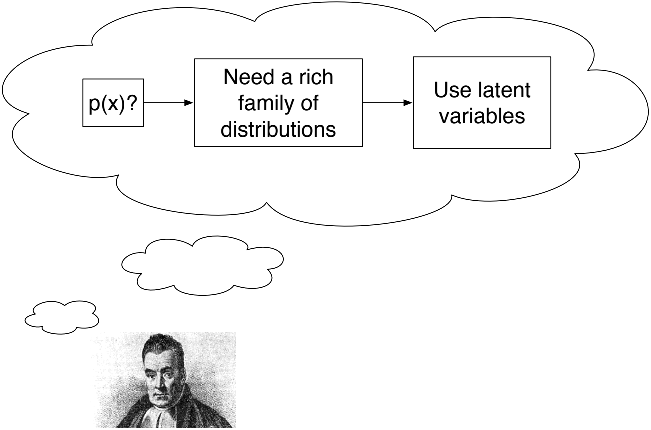
- Model: $p(x, z)$
- Latent Variables: z
- Data: x

Variational Inference:

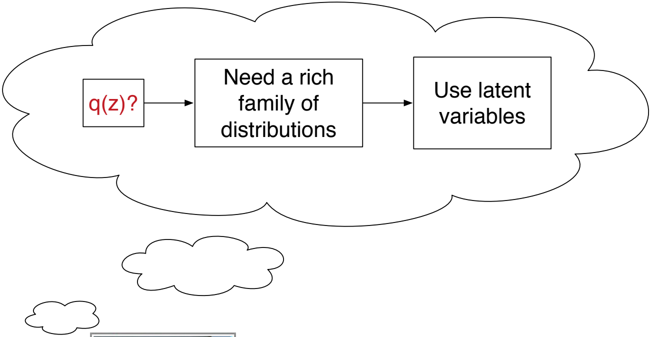
- Approximating Family: $q(z; \lambda)$
- Minimize $KL(q||p(z \mid x))$ or maximize ELBO:

$$\mathcal{L}(\lambda) = \mathbb{E}_q[\log p(x, z) - \log q(z; \lambda)]$$

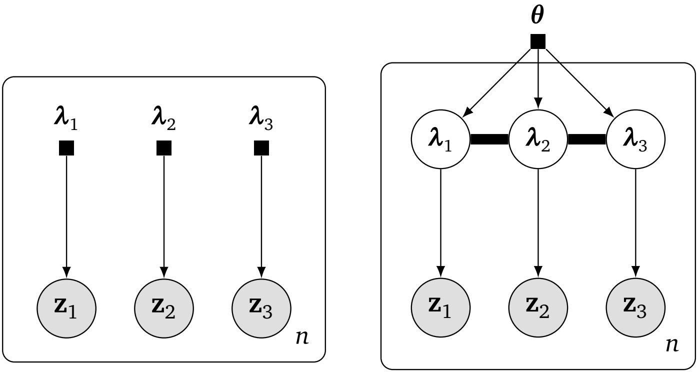
Models


p(x)?
Need a rich family of distributions
Use latent variables

Variational Models



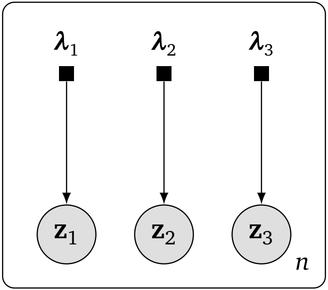
- Variational approximations by using priors on tractable families
- We focus on the mean-field



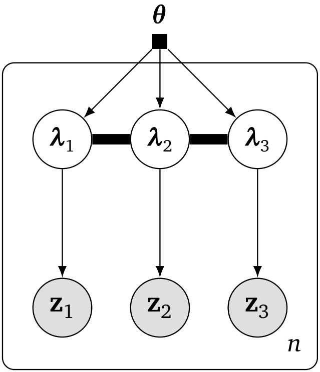
(a) MEAN-FIELD MODEL **(b)** HIERARCHICAL MODEL

Hierarchical Variational Models

- Mean-field distribution: $q(z;\lambda) = \prod_{i=1}^{d} q(z_i;\lambda_i)$
- Hierarchical variational approximation
  $q(z;\theta) = \int \prod_{i=1}^{d} q(z_i \mid \lambda_i) q(\lambda;\theta) d\lambda$



(a) MEAN-FIELD MODEL



(b) HIERARCHICAL MODEL

Example HVM Priors

- Multivariate Normal: $q(\lambda) = \mathrm{Normal}(\mu, \Sigma)$
- Normalizing Flow:

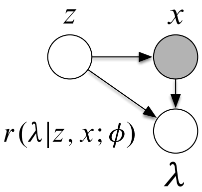
$$q_0 \sim F$$

$$\log q(\boldsymbol{\lambda}) = \log q(\boldsymbol{\lambda}_0) - \sum_{k=1}^{K} \log \left(\left| \det(\frac{\partial f_k}{\partial z_k}) \right| \right)$$

- The number of free moments equals the number of parameters in the hyperprior

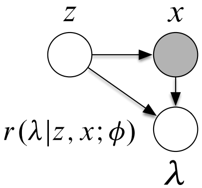
How to find a good HVM?

- Entropy is intractable
- Approximate by expanding the model and doing VI



How to find a good HVM?

- Entropy is intractable
- Approximate by expanding the model and doing VI



- $\widetilde{\mathcal{L}}(\theta, \phi) = \mathbb{E}_q[\log p(x, z) + \log r(\lambda \mid x, z; \phi) - \log q(z, \lambda; \theta)]$
- Looser than VB in the marginal model

Stochastic Gradient of HVM

- $\nabla_\lambda \mathcal{L} = \mathbb{E}_q[\nabla_\lambda \log q(z;\lambda)(\log p(x,z) - \log q(z;\lambda))]$
- Variance of Monte Carlo estimates scales with learning signal

Stochastic Gradient of HVM

- $\nabla_\lambda \mathcal{L} = \mathbb{E}_q[\nabla_\lambda \log q(z;\lambda)(\log p(x,z) - \log q(z;\lambda))]$
- Variance of Monte Carlo estimates scales with learning signal
- Mean-field gradient:
  $\nabla_{\lambda_i} \mathcal{L} = E_{q_{(i)}}[\nabla_{\lambda_i} \log q(z_i;\lambda_i)(\log p_i(x, z_{(i)}) - \log q(z_i;\lambda_i))]$

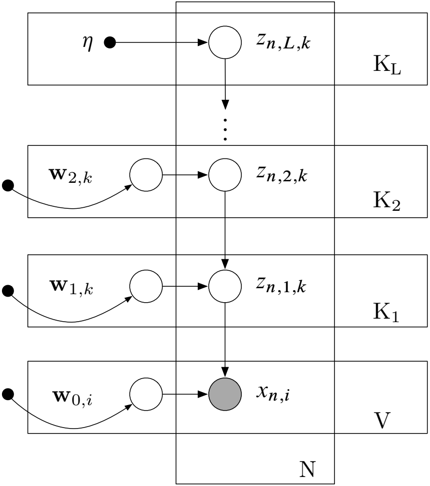
Stochastic Gradient of HVM

- $\nabla_\lambda \mathcal{L} = \mathbb{E}_q[\nabla_\lambda \log q(z;\lambda)(\log p(x,z) - \log q(z;\lambda))]$
- Variance of Monte Carlo estimates scales with learning signal
- Mean-field gradient:
  $\nabla_{\lambda_i} \mathcal{L} = E_{q_{(i)}}[\nabla_{\lambda_i} \log q(z_i;\lambda_i)(\log p_i(x, z_{(i)}) - \log q(z_i;\lambda_i))]$
- Gradient of HVM is

$$\begin{aligned}
\nabla_\theta \widetilde{\mathcal{L}}(\theta,\phi) &= \mathbb{E}_{s(\epsilon)}[\nabla_\theta \lambda(\epsilon) \nabla_\lambda \mathcal{L}_{\mathrm{MF}}(\lambda)] \\
&\quad + \mathbb{E}_{s(\epsilon)}[\nabla_\theta \lambda(\epsilon) \nabla_\lambda [\log r(\lambda \mid z;\phi) - \log q(\lambda;\theta)]] \\
&\quad + \mathbb{E}_{s(\epsilon)}[\nabla_\theta \lambda(\epsilon) \mathbb{E}_{q(\mathbf{z} \mid \lambda)}[\nabla_\lambda \log q(z;\lambda) \log r(\lambda \mid z;\phi)]].
\end{aligned}$$

If r factorizes in z, we maintain computational efficiency

Deep Exponential Families



$$p(z_{n,L,k}) = \text{EXPFAM}_L(z_{n,L,k}, \eta)$$

$$p(z_{2,k} \mid \mathbf{z}_3, \mathbf{w}_{2,k}) = \text{EXPFAM}_2(z_{2,k}, g_2(\mathbf{z}_3^\top \mathbf{w}_{2,k}))$$

$$p(z_{1,k} \mid \mathbf{z}_2, \mathbf{w}_{1,k}) = \text{EXPFAM}_1(z_{1,k}, g_1(\mathbf{z}_2^\top \mathbf{w}_{1,k}))$$

$$p(x_{n,i} \mid \mathbf{z}_{n,1}, \mathbf{w}_{0,i}) = \text{Poisson}(\mathbf{z}_{n,1}^\top \mathbf{w}_{0,i})$$

Results on Deep Exponential Families

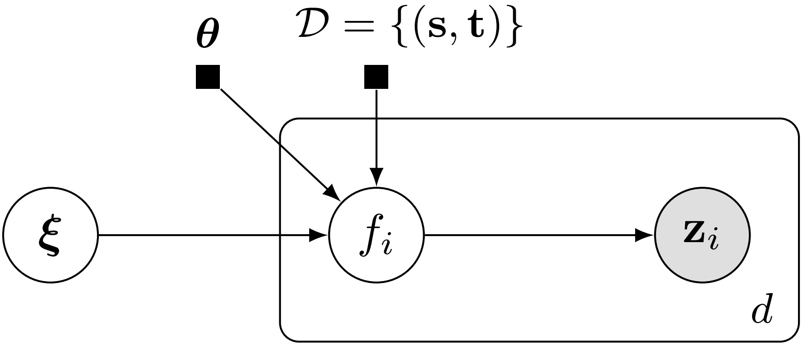
Results on DEF with Poisson latent layers

	Model	HVM	Mean-Field
NYT	100	**3570**	**3570**
	100-30	**3460**	3660
	100-30-15	**3480**	3550
Science	100	**3360**	3377
	100-30	**3080**	3240
	100-30-15	**3110**	3190

Held out Perplexity; Similar results on sigmoid belief networks

HVM with Gaussian Processes

We can build variational models with Gaussian processes.



$\xi \sim \mathrm{Normal}(0, I), \; f_i \sim \mathrm{GP}(0, K)|\mathcal{D}_i$

Results on Variational Autoencoders

Model	$-\log p(\mathbf{x})$	$\leq$
DLGM + VAE [Burda et al., 2015]		86.76
DLGM + HVI (8 leapfrog steps) [Salimans et al., 2015]	85.51	88.30
DLGM + NF ($k = 80$) [Rezende + Mohamed, 2015]		85.10
EoNADE-5 2hl (128 orderings) [Raiko et al., 2015]	84.68	
DBN 2hl [Murray + Salakhutdinov, 2009]	84.55	
DARN 1hl [Gregor et al., 2014]	84.13	
Convolutional VAE + HVI [Salimans et al., 2015]	81.94	83.49
DLGM 2hl + IWAE ($k = 50$) [Burda et al., 2015]		82.90
DRAW [Gregor et al. 2015]		80.97
DLGM 1hl + VGP		83.64
DLGM 2hl + VGP		81.90
DRAW + VGP		**80.11**

We also find richer latent representations than the VAE or IWAE.

Thanks Again