PETUUM

Carnegie Mellon University
School of Computer Science

Probabilistic Graphical Models

Deep generative models: overview of the theoretical basis and connections

Eric Xing

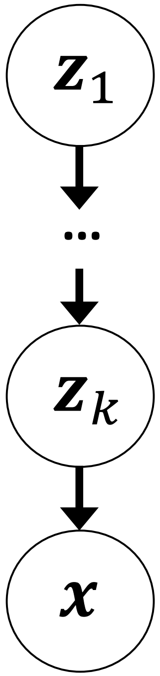
Lecture 12, February 25, 2020

Reading: see class homepage

Deep generative models

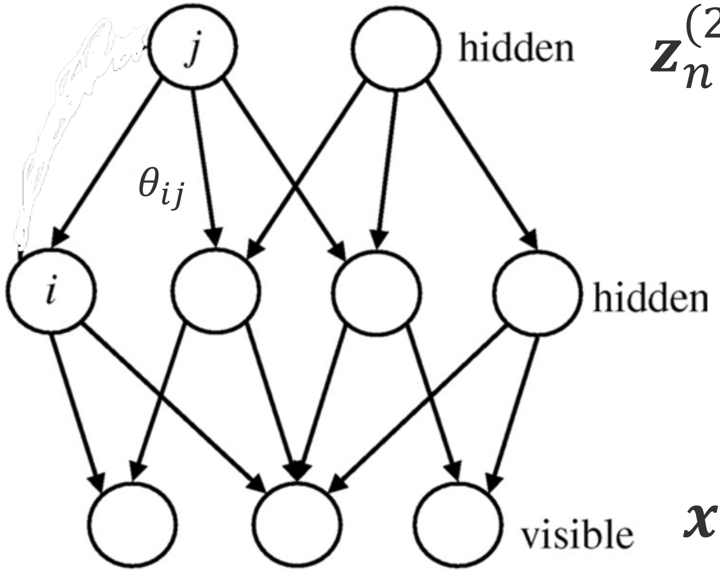
Deep generative models

- Define probabilistic distributions over a set of variables
- "Deep" means multiple layers of hidden variables!



Early forms of deep generative models

- Hierarchical Bayesian models
  - Sigmoid brief nets [Neal 1992]



$$\boldsymbol{z}_n^{(2)} = \{0,1\}^J$$

$$\boldsymbol{z}_n^{(1)} = \{0,1\}^I$$

$$\boldsymbol{x}_n = \{0,1\}^K$$

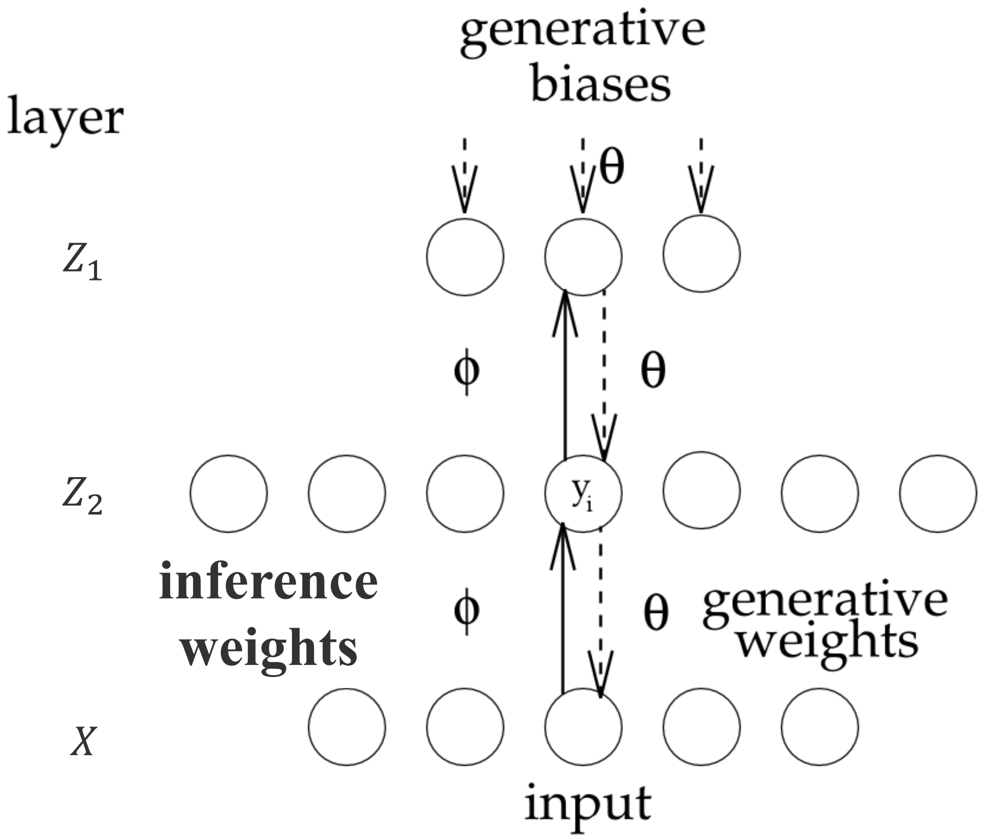
$$p\left(x_{kn} = 1 \middle| \boldsymbol{\theta}_k, \boldsymbol{z}_n^{(1)}\right) = \sigma\left(\boldsymbol{\theta}_k^T \boldsymbol{z}_n^{(1)}\right)$$
$$p\left(z_{in}^{(1)} = 1 \middle| \boldsymbol{\theta}_i, \boldsymbol{z}_n^{(2)}\right) = \sigma\left(\boldsymbol{\theta}_i^T \boldsymbol{z}_n^{(2)}\right)$$

Early forms of deep generative models

- Hierarchical Bayesian models
  - Sigmoid brief nets [Neal 1992]
- Neural network models
  - Helmholtz machines [Dayan et al.,1995]

    -- alternative inference/learning methods

    -- the process, not a global math expression, defines the model



[Dayan et al. 1995]

Early forms of deep generative models

- Hierarchical Bayesian models
  - Sigmoid brief nets [Neal 1992]

- Neural network models
  - Helmholtz machines [Dayan et al.,1995]
    -- alternative inference/learning methods
    -- the process, not a global math expression, defines the model
  - Predictability minimization [Schmidhuber 1995]
    -- alternative loss-functions
    -- again, no explicit global math expression. the training procedure defines it implicitly

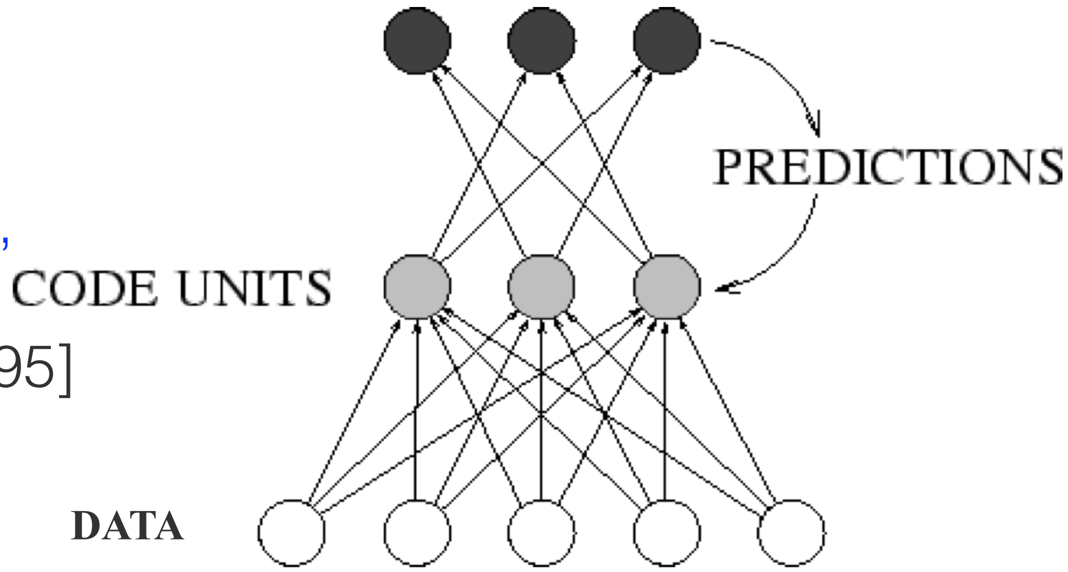


Figure courtesy: Schmidhuber 1996

The word "model" is here not very rigorous anymore!

Early forms of deep generative models

- Training of DGMs via an EM style framework
  - Sampling / data augmentation

$$\boldsymbol{z} = \{\boldsymbol{z}_1, \boldsymbol{z}_2\}$$
$$\boldsymbol{z}_1^{new} \sim p(\boldsymbol{z}_1 | \boldsymbol{z}_2, \boldsymbol{x})$$
$$\boldsymbol{z}_2^{new} \sim p(\boldsymbol{z}_2 | \boldsymbol{z}_1^{new}, \boldsymbol{x})$$

  - Variational inference

$$\log p(\boldsymbol{x}) \geq \mathrm{E}_{q_\phi(\boldsymbol{z}|\boldsymbol{x})}[\log p_\theta(\boldsymbol{x}, \boldsymbol{z})] - \mathrm{KL}(q_\phi(\boldsymbol{z}|\boldsymbol{x}) \,||\, p(\boldsymbol{z})) \coloneqq \mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}; \boldsymbol{x})$$

$$\max_{\boldsymbol{\theta}, \boldsymbol{\phi}} \mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}; \boldsymbol{x})$$

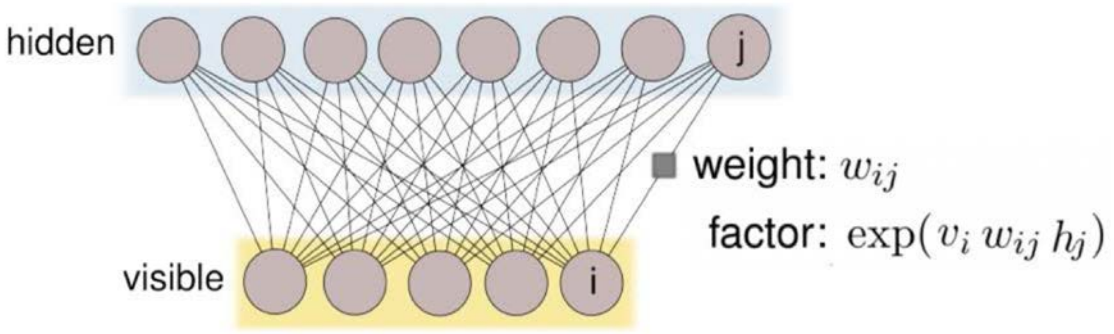
  - Wake sleep

Wake: $\min_\theta \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)]$

Sleep: $\min_\phi \mathbb{E}_{p_\theta(x|z)}[\log q_\phi(z|x)]$

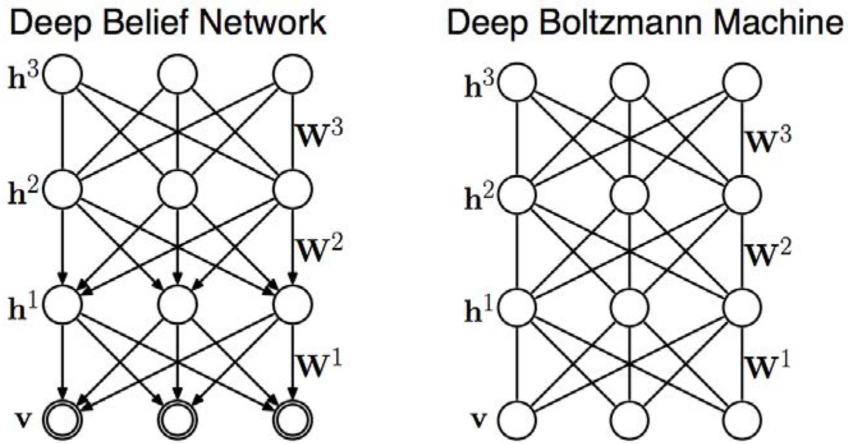
Resurgence of deep generative models

- Restricted Boltzmann machines (RBMs) [Smolensky, 1986]
  - Building blocks of deep probabilistic models



Resurgence of deep generative models

- Restricted Boltzmann machines (RBMs) [Smolensky, 1986]
  - Building blocks of deep probabilistic models
- Deep belief networks (DBNs) [Hinton et al., 2006]
  - Hybrid graphical model
  - Inference in DBNs is problematic due to explaining away
- Deep Boltzmann Machines (DBMs) [Salakhutdinov & Hinton, 2009]
  - Undirected model

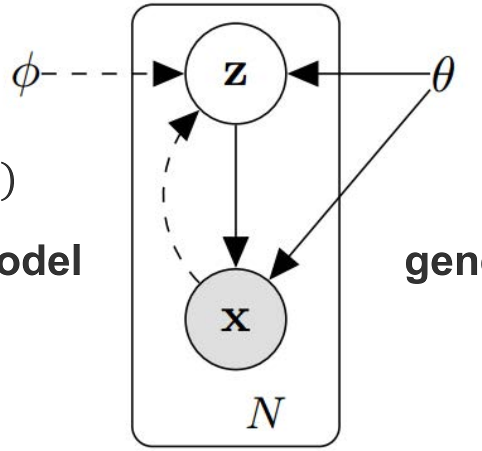


Resurgence of deep generative models

- Variational autoencoders (VAEs) [Kingma & Welling, 2014]
  / Neural Variational Inference and Learning (NVIL) [Mnih & Gregor, 2014]

$q_\phi(\boldsymbol{z}|\boldsymbol{x})$

inference model



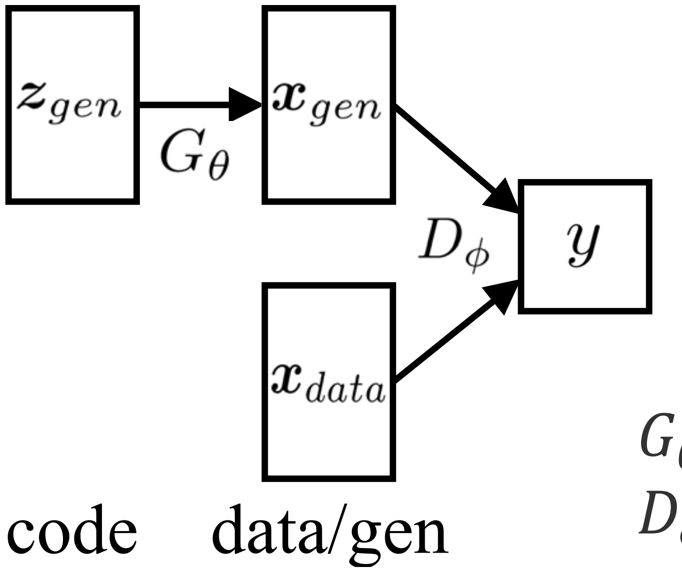
$p_\theta(\boldsymbol{x}|\boldsymbol{z})$

generative model

Figure courtesy: Kingma & Welling, 2014

Resurgence of deep generative models

- Variational autoencoders (VAEs) [Kingma & Welling, 2014]
  / Neural Variational Inference and Learning (NVIL) [Mnih & Gregor, 2014]
- Generative adversarial networks (GANs) [Goodfellow et al,. 2014]



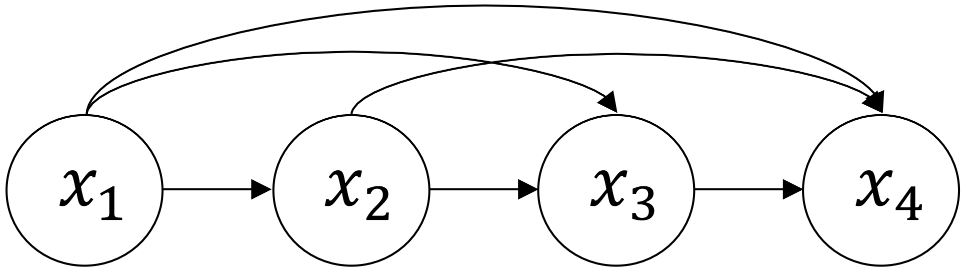
G_θ: **generative model ?**
D_ϕ: **discriminator**

Resurgence of deep generative models

- Variational autoencoders (VAEs) [Kingma & Welling, 2014]
  / Neural Variational Inference and Learning (NVIL) [Mnih & Gregor, 2014]
- Generative adversarial networks (GANs) [Goodfellow et al,. 2014]
- Generative moment matching networks (GMMNs) [Li et al., 2015; Dziugaite et al., 2015]

Resurgence of deep generative models

- Variational autoencoders (VAEs) [Kingma & Welling, 2014]
  / Neural Variational Inference and Learning (NVIL) [Mnih & Gregor, 2014]
- Generative adversarial networks (GANs) [Goodfellow et al,. 2014]
- Generative moment matching networks (GMMNs) [Li et al., 2015; Dziugaite et al., 2015]
- Autoregressive neural networks



Outline

- Theoretical Basis of deep generative models
  - Wake sleep algorithm
  - Variational autoencoders
  - Generative adversarial networks
- A unified view of deep generative models
  - new formulations of deep generative models
  - Symmetric modeling of latent and visible variables

Synonyms in the literature

- Posterior Distribution -> Inference model
  - Variational approximation
  - Recognition model
  - Inference network (if parameterized as neural networks)
  - Recognition network (if parameterized as neural networks)
  - (Probabilistic) encoder
- "The Model" (prior + conditional, or joint) -> Generative model
  - The (data) likelihood model
  - Generative network (if parameterized as neural networks)
  - Generator
  - (Probabilistic) decoder

Recap: Variational Inference

- Consider a generative model $p_\theta(\boldsymbol{x}|\boldsymbol{z})$, and prior $p(\boldsymbol{z})$
  - Joint distribution: $p_\theta(\boldsymbol{x}, \boldsymbol{z}) = p_\theta(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z})$
- Assume variational distribution $q_\phi(\boldsymbol{z}|\boldsymbol{x})$
- Objective: Maximize lower bound for log likelihood

$$\begin{aligned}
&\log p(\boldsymbol{x}) \\
&= KL\left(q_\phi(\boldsymbol{z}|\boldsymbol{x}) \,||\, p_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x})\right) + \int_{\boldsymbol{z}} q_\phi(\boldsymbol{z}|\boldsymbol{x}) \log \frac{p_\theta(\boldsymbol{x}, \boldsymbol{z})}{q_\phi(\boldsymbol{z}|\boldsymbol{x})} \\
&\geq \int_{\boldsymbol{z}} q_\phi(\boldsymbol{z}|\boldsymbol{x}) \log \frac{p_\theta(\boldsymbol{x}, \boldsymbol{z})}{q_\phi(\boldsymbol{z}|\boldsymbol{x})} \\
&\coloneqq \mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}; \boldsymbol{x})
\end{aligned}$$

  - Equivalently, minimize free energy

$$F(\boldsymbol{\theta}, \phi; \boldsymbol{x}) = -\log p(\boldsymbol{x}) + KL(q_\phi(\boldsymbol{z}|\boldsymbol{x}) \,||\, p_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x}))$$

Recap: Variational Inference

Maximize the variational lower bound:

$$\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}; \boldsymbol{x}) = \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)] + KL(q_\phi(z|x)||p(z))$$

$$= \log p(\boldsymbol{x}) - KL(q_\phi(\boldsymbol{z}|\boldsymbol{x}) \,||\, p_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x}))$$

- E-step: maximize $\mathcal{L}$ wrt. $\boldsymbol{\phi}$, with $\boldsymbol{\theta}$ fixed

$$\max_\phi \mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}; \boldsymbol{x})$$

  - If closed form solutions exist:

$$q_\phi^*(z|x) \propto \exp[\log p_\theta(x, z)]$$

- M-step: maximize $\mathcal{L}$ wrt. $\boldsymbol{\theta}$, with $\boldsymbol{\phi}$ fixed

$$\max_\theta \mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}; \boldsymbol{x})$$

Wake Sleep Algorithm [Hinton et al., Science 1995]

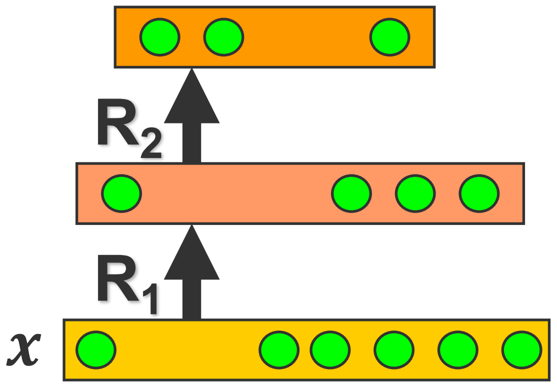
- Train a separate inference model along with the generative model
  - Generally applicable to a wide range of generative models, e.g., Helmholtz machines
- Consider a generative model $p_\theta(\boldsymbol{x}|\boldsymbol{z})$ and prior $p(\boldsymbol{z})$
  - Joint distribution $p_\theta(\boldsymbol{x}, \boldsymbol{z}) = p_\theta(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z})$
  - E.g., multi-layer brief nets
- Inference model $q_\phi(\boldsymbol{z}|\boldsymbol{x})$
- Maximize data log-likelihood with two steps of loss relaxation:
  - Maximize the variational lower bound of log-likelihood, or equivalently, minimize the free energy

$$F(\boldsymbol{\theta}, \boldsymbol{\phi}; \boldsymbol{x}) = -\log p(\boldsymbol{x}) + KL(q_\phi(\boldsymbol{z}|\boldsymbol{x}) \,||\, p_\theta(\boldsymbol{z}|\boldsymbol{x}))$$

  - Minimize a different objective (reversed KLD) wrt ϕ to ease the optimization
    - Disconnect to the original variational lower bound loss

$$F'(\boldsymbol{\theta}, \boldsymbol{\phi}; \boldsymbol{x}) = -\log p(\boldsymbol{x}) + KL(p_\theta(\boldsymbol{z}|\boldsymbol{x}) \,||\, q_\phi(\boldsymbol{z}|\boldsymbol{x}))$$

Wake Sleep Algorithm



- Free energy:

$$F(\boldsymbol{\theta}, \boldsymbol{\phi}; \boldsymbol{x}) = -\log p(\boldsymbol{x}) + KL(q_{\boldsymbol{\phi}}(\mathbf{z}|\boldsymbol{x}) \,||\, p_{\boldsymbol{\theta}}(\mathbf{z}|\boldsymbol{x}))$$

- Minimize the free energy wrt. $\boldsymbol{\theta}$ of p_θ → *wake* phase

$$\max_{\boldsymbol{\theta}} \mathrm{E}_{q_\phi(\mathbf{z}|\boldsymbol{x})} [\log p_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{z})]$$

  - Get samples from $q_\phi(z|x)$ through inference on hidden variables
  - Use the samples as targets for updating the generative model $p_\theta(\boldsymbol{z}|\boldsymbol{x})$
  - Correspond to the variational M step

[Figure courtesy: Maei's slides]

Wake Sleep Algorithm

- Free energy:

$$F(\boldsymbol{\theta}, \boldsymbol{\phi}; \boldsymbol{x}) = -\log p(\boldsymbol{x}) + KL(q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}) \,||\, p_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x}))$$

- Minimize the free energy wrt. $\boldsymbol{\phi}$ of $q_{\phi}(\boldsymbol{z}|\boldsymbol{x})$
  - Correspond to the variational E step $\max_{\boldsymbol{\phi}} \mathrm{E}_{q_{\phi}(\boldsymbol{z}|\boldsymbol{x})} [\log p_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{z})]$
  - Difficulties:
    - Optimal $q_{\boldsymbol{\phi}}^{*}(\boldsymbol{z}|\boldsymbol{x}) = \dfrac{p_{\boldsymbol{\theta}}(\boldsymbol{z}, \boldsymbol{x})}{\int p_{\boldsymbol{\theta}}(\boldsymbol{z}, \boldsymbol{x})\, d\boldsymbol{z}}$ intractable
    - High variance of direct gradient estimate $\nabla_{\phi} F(\theta, \phi; x) = \cdots + \nabla_{\phi} \mathbb{E}_{q_{\phi}(z|x)}[\log p_{\theta}(z, x)] + \cdots$
      - Gradient estimate with the log-derivative trick:

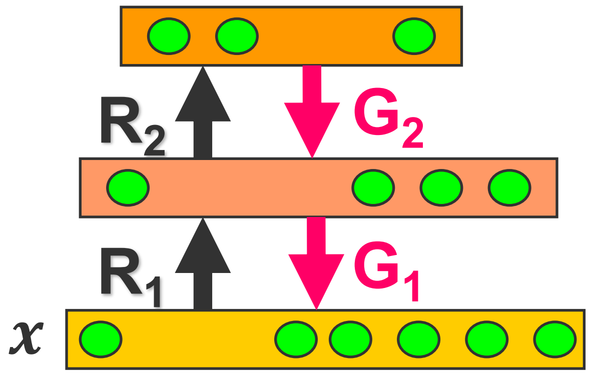
$$\nabla_{\phi} \mathbb{E}_{q_{\phi}}[\log p_{\theta}] = \int \nabla_{\phi} q_{\phi} \log p_{\theta} = \int q_{\phi} \log p_{\theta} \, \nabla_{\phi} \log q_{\phi} = \mathbb{E}_{q_{\phi}}[\log p_{\theta} \, \nabla_{\phi} \log q_{\phi}]$$

        - Monte Carlo estimation:

$$\nabla_{\phi} \mathbb{E}_{q_{\phi}}[\log p_{\theta}] \approx \mathbb{E}_{z_i \sim q_{\phi}}[\log p_{\theta}(x, z_i) \, \nabla_{\phi} q_{\phi}(z_i|x)]$$

        - The scale factor $\log p_{\theta}(x, z_i)$ can have arbitrary large magnitude

Wake Sleep Algorithm



- Free energy:

$$F(\boldsymbol{\theta}, \boldsymbol{\phi}; \boldsymbol{x}) = -\log p(\boldsymbol{x}) + KL(q_{\boldsymbol{\phi}}(\mathbf{z}|\boldsymbol{x}) \,||\, p_{\boldsymbol{\theta}}(\mathbf{z}|\boldsymbol{x}))$$

- WS works around the difficulties with the sleep phase approximation
- Minimize the following objective → *sleep* phase

$$F'(\boldsymbol{\theta}, \boldsymbol{\phi}; \boldsymbol{x}) = -\log p(\boldsymbol{x}) + KL(p_{\theta}(\mathbf{z}|\boldsymbol{x}) \,||\, q_{\phi}(\mathbf{z}|\boldsymbol{x}))$$

$$\max_{\boldsymbol{\phi}} \mathrm{E}_{p_{\theta}(\mathbf{z},\boldsymbol{x})} \left[\log q_{\phi}(\mathbf{z}|\boldsymbol{x})\right]$$

$$\cancel{\max_{\boldsymbol{\phi}} \mathrm{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\boldsymbol{x})} [\log p_{\boldsymbol{\theta}}(\boldsymbol{x}, \mathbf{z})]}$$

  - "Dreaming" up samples from $p_{\theta}(\boldsymbol{x}|\mathbf{z})$ through top-down pass
  - Use the samples as targets for updating the inference model
- (Recent approaches other than sleep phase are developed to reduce the variance of gradient estimate: slides later)

[Figure courtesy: Maei's slides]

Wake Sleep Algorithm

Wake sleep

- Parametrized inference model $q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})$
- Wake phase:
  - minimize $KL(q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}) \,||\, p_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x}))$ wrt. θ
  - $\mathrm{E}_{q_{\phi}(\boldsymbol{z}|\boldsymbol{x})}\left[\nabla_{\theta} \log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})\right]$
- Sleep phase:
  - minimize $KL(p_{\theta}(\boldsymbol{z}|\boldsymbol{x}) \,||\, q_{\phi}(\boldsymbol{z}|\boldsymbol{x}))$ wrt. ϕ
  - $\mathrm{E}_{p_{\theta}(\boldsymbol{z},\boldsymbol{x})}\left[\nabla_{\phi} \log q_{\phi}(\boldsymbol{z},\boldsymbol{x})\right]$
  - low variance
  - Learning with generated samples of $\boldsymbol{x}$
- Two objective, not guaranteed to converge

Variational EM

- Variational distribution $q_{\phi}(\boldsymbol{z}|\boldsymbol{x})$
- Variational M step:
  - minimize $KL(q_{\phi}(\boldsymbol{z}|\boldsymbol{x}) \,||\, p_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x}))$ wrt. θ
  - $\mathrm{E}_{q_{\phi}(\boldsymbol{z}|\boldsymbol{x})}\left[\nabla_{\theta} \log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})\right]$
- Variational E step:
  - minimize $KL(q_{\phi}(\boldsymbol{z}|\boldsymbol{x}) \,||\, p_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x}))$ wrt. ϕ
  - $q_{\phi}^{*} \propto \exp[\log p_{\theta}]$ if with closed-form
  - $\nabla_{\phi} \mathbb{E}_{q_{\phi}}[\log p_{\theta}(z, x)]$
    - need variance-reduce in practice
  - Learning with real data $\boldsymbol{x}$
- Single objective, guaranteed to converge

Variational Autoencoders (VAEs)

- [Kingma & Welling, 2014]
- Use variational inference with an inference model
  - Enjoy similar applicability with wake-sleep algorithm
- Generative model $p_\theta(\boldsymbol{x}|\boldsymbol{z})$, and prior $p(\boldsymbol{z})$
  - Joint distribution $p_\theta(\boldsymbol{x}, \boldsymbol{z}) = p_\theta(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z})$
- Inference model $q_\phi(\boldsymbol{z}|\boldsymbol{x})$

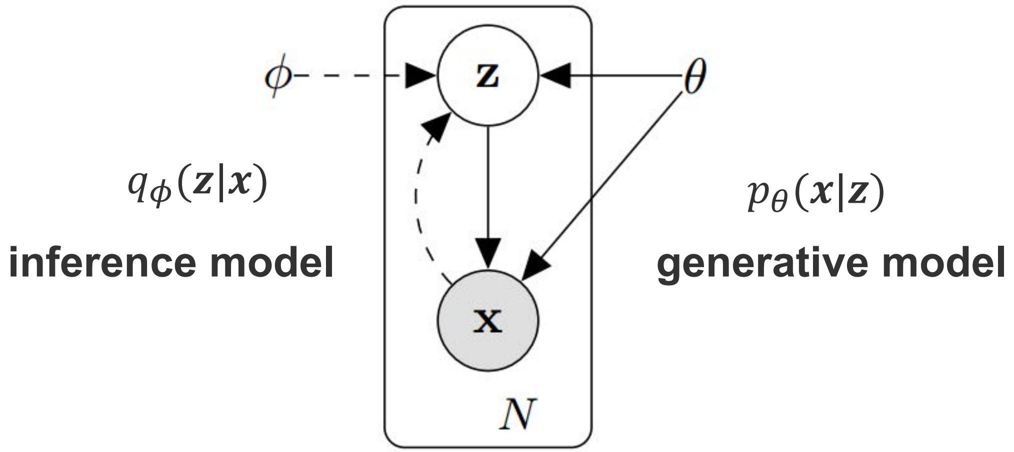


Figure courtesy: Kingma & Welling, 2014

Variational Autoencoders (VAEs)

- Variational lower bound

$$\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}; \boldsymbol{x}) = \mathrm{E}_{q_\phi(\boldsymbol{z}|\boldsymbol{x})}[\log p_\theta(\boldsymbol{x}, \boldsymbol{z})] - \mathrm{KL}(q_\phi(\boldsymbol{z}|\boldsymbol{x}) \,||\, p(\boldsymbol{z}))$$

- Optimize $\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}; \boldsymbol{x})$ wrt. θ of $p_\theta(\boldsymbol{x}|\boldsymbol{z})$
  - The same with the wake phase
- Optimize $\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}; \boldsymbol{x})$ wrt. ϕ of $q_\phi(\boldsymbol{z}|\boldsymbol{x})$

$$\nabla_\phi \mathcal{L}(\theta, \phi; x) = \cdots + \nabla_\phi \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)] + \cdots$$

  - Use *reparameterization trick* to reduce variance
  - Alternatives: use control variates as in reinforcement learning [Mnih & Gregor, 2014; Paisley et al., 2012]

Reparametrized gradient

- Optimize $\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}; \boldsymbol{x})$ wrt. ϕ of $q_\phi(\boldsymbol{z}|\boldsymbol{x})$
  - Recap: gradient estimate with log-derivative trick:

$$\nabla_\phi \mathbb{E}_{q_\phi}[\log p_\theta(\boldsymbol{x}, \boldsymbol{z})] = \mathbb{E}_{q_\phi}[\log p_\theta(\boldsymbol{x}, \boldsymbol{z})\ \nabla_\phi \log q_\phi]$$

    - High variance: $\nabla_\phi \mathbb{E}_{q_\phi}[\log p_\theta] \approx \mathbb{E}_{z_i \sim q_\phi}[\log p_\theta(x, z_i)\ \nabla_\phi q_\phi(z_i|x)]$
      - The scale factor $\log p_\theta(x, z_i)$ can have arbitrary large magnitude
  - gradient estimate with *reparameterization trick*

$$\boldsymbol{z} \sim q_\phi(\boldsymbol{z}|\boldsymbol{x}) \quad \Leftrightarrow \quad \boldsymbol{z} = \mathrm{g}_\phi(\boldsymbol{\epsilon}, \boldsymbol{x}), \qquad \boldsymbol{\epsilon} \sim p(\boldsymbol{\epsilon})$$

$$\nabla_\phi \mathrm{E}_{q_\phi(\boldsymbol{z}|\boldsymbol{x})}[\log p_\theta(\boldsymbol{x}, \boldsymbol{z})] = \mathrm{E}_{\epsilon \sim p(\epsilon)}\left[\nabla_\phi \log p_\theta\left(\boldsymbol{x}, \boldsymbol{z}_\phi(\boldsymbol{\epsilon})\right)\right]$$

    - (Empirically) lower variance of the gradient estimate
    - E.g., $\boldsymbol{z} \sim N\left(\boldsymbol{\mu}(\boldsymbol{x}), \boldsymbol{L}(\boldsymbol{x})\boldsymbol{L}(\boldsymbol{x})^T\right) \quad \Leftrightarrow \quad \boldsymbol{\epsilon} \sim N(0,1),\ \boldsymbol{z} = \boldsymbol{\mu}(\boldsymbol{x}) + \boldsymbol{L}(\boldsymbol{x})\boldsymbol{\epsilon}$

VAEs: algorithm

Algorithm 1 Minibatch version of the Auto-Encoding VB (AEVB) algorithm. Either of the two SGVB estimators in section 2.3 can be used. We use settings $M = 100$ and $L = 1$ in experiments.

$\boldsymbol{\theta}, \boldsymbol{\phi} \leftarrow$ Initialize parameters
repeat
  $\mathbf{X}^M \leftarrow$ Random minibatch of M datapoints (drawn from full dataset)
  $\boldsymbol{\epsilon} \leftarrow$ Random samples from noise distribution $p(\boldsymbol{\epsilon})$
  $\mathbf{g} \leftarrow \nabla_{\boldsymbol{\theta},\boldsymbol{\phi}} \widetilde{\mathcal{L}}^M(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{X}^M, \boldsymbol{\epsilon})$ (Gradients of minibatch estimator (8))
  $\boldsymbol{\theta}, \boldsymbol{\phi} \leftarrow$ Update parameters using gradients $\mathbf{g}$ (e.g. SGD or Adagrad [DHS10])
until convergence of parameters $(\boldsymbol{\theta}, \boldsymbol{\phi})$
return $\boldsymbol{\theta}, \boldsymbol{\phi}$

[Kingma & Welling, 2014]

VAEs: example results

- VAEs tend to generate **blurred** images due to the mode covering behavior (more later)

Celebrity faces [Radford 2015]

- Latent code interpolation and sentences generation from VAEs [Bowman et al., 2015].

" i want to talk to you . "
"i want to be with you . "
"i do n't want to be with you . "
i do n't want to be with you .
she did n't want to be with him .

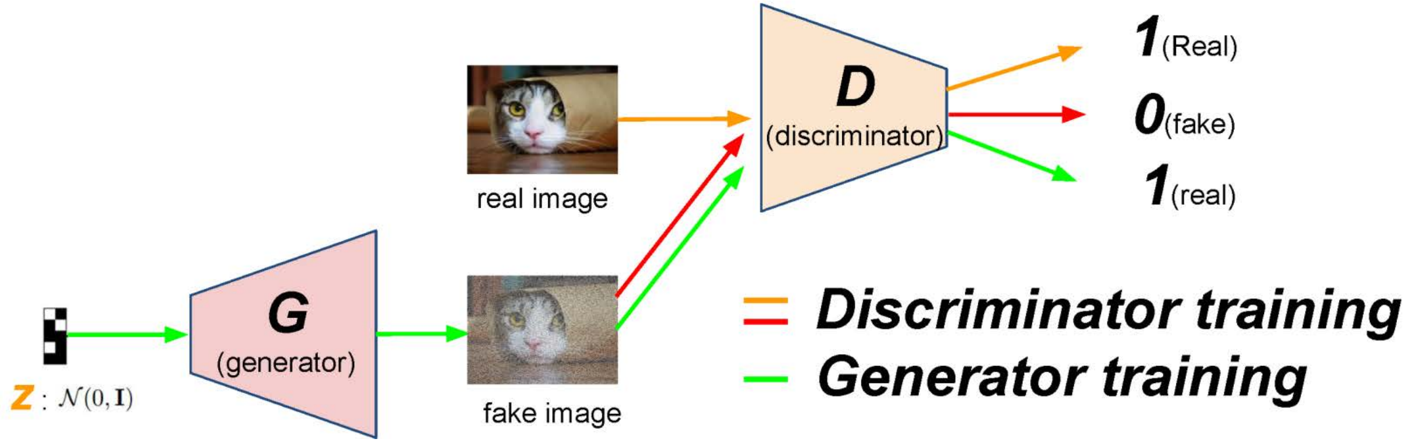
Generative Adversarial Nets (GANs)

- [Goodfellow et al., 2014]
- Generative model $\boldsymbol{x} = G_\theta(\boldsymbol{z}), \ \ \boldsymbol{z} \sim p(\boldsymbol{z})$
  - Map noise variable $\boldsymbol{z}$ to data space $\boldsymbol{x}$
  - Define an implicit distribution over $\boldsymbol{x}$: $p_{g_\theta}(\boldsymbol{x})$
    - a stochastic process to simulate data $\boldsymbol{x}$
    - Intractable to evaluate likelihood
- Discriminator $D_\phi(\boldsymbol{x})$
  - Output the probability that $\boldsymbol{x}$ came from the data rather than the generator
- No explicit inference model
- No obvious connection to previous models with inference networks like VAEs
  - We will build formal connections between GANs and VAEs later

Generative Adversarial Nets (GANs)

- Learning
  - A minimax game between the generator and the discriminator
  - Train D to maximize the probability of assigning the correct label to both training examples and generated samples
  - Train G to fool the discriminator

$$\max_D \mathcal{L}_D = \mathbb{E}_{\boldsymbol{x}\sim p_{data}(\boldsymbol{x})}\left[\log D(\boldsymbol{x})\right] + \mathbb{E}_{\boldsymbol{x}\sim G(\boldsymbol{z}),\boldsymbol{z}\sim p(\boldsymbol{z})}\left[\log(1-D(\boldsymbol{x}))\right]$$
$$\min_G \mathcal{L}_G = \mathbb{E}_{\boldsymbol{x}\sim G(\boldsymbol{z}),\boldsymbol{z}\sim p(\boldsymbol{z})}\left[\log(1-D(\boldsymbol{x}))\right].$$

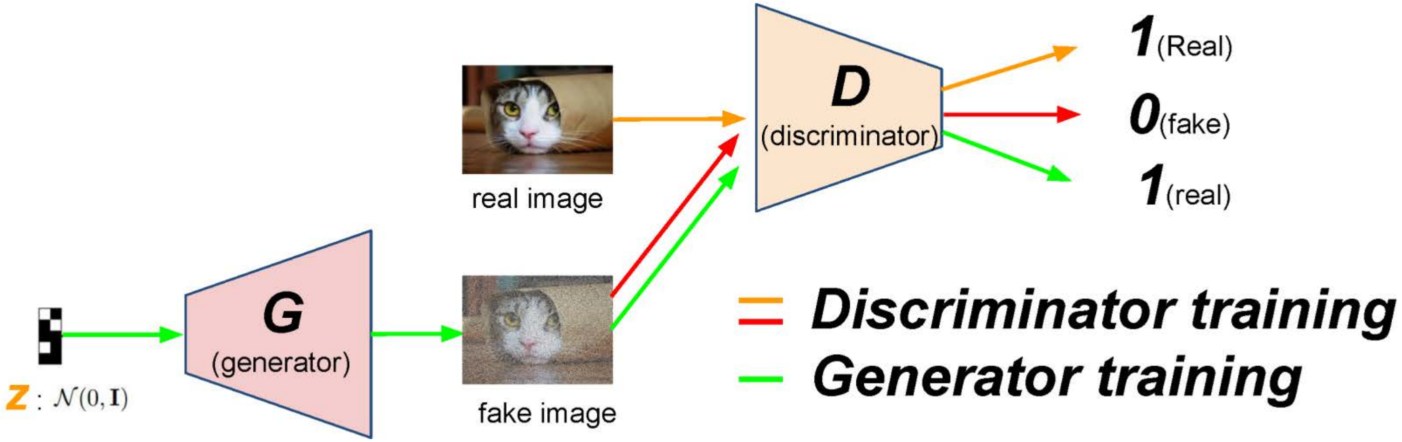


[Figure courtesy: Kim's slides]

Generative Adversarial Nets (GANs)

- Learning
  - Train G to fool the discriminator
    - The original loss suffers from vanishing gradients when D is too strong
    - Instead use the following in practice

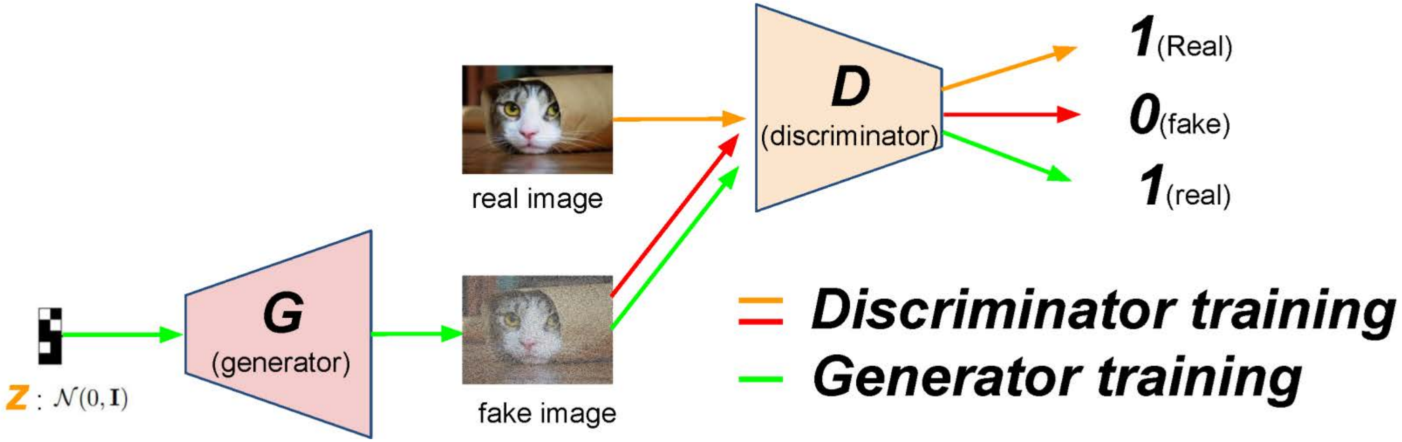
$$\max_G \mathcal{L}_G = \mathbb{E}_{\boldsymbol{x}\sim G(\boldsymbol{z}),\boldsymbol{z}\sim p(\boldsymbol{z})}\left[\log D(\boldsymbol{x})\right]$$



[Figure courtesy: Kim's slides]

Generative Adversarial Nets (GANs)

- Learning
  - Aim to achieve equilibrium of the game
  - Optimal state:
    - $p_g(\boldsymbol{x}) = p_{data}(x)$
    - $D(\boldsymbol{x}) = \frac{p_{data}(x)}{p_{data}(x)+p_g(x)} = \frac{1}{2}$



[Figure courtesy: Kim's slides]

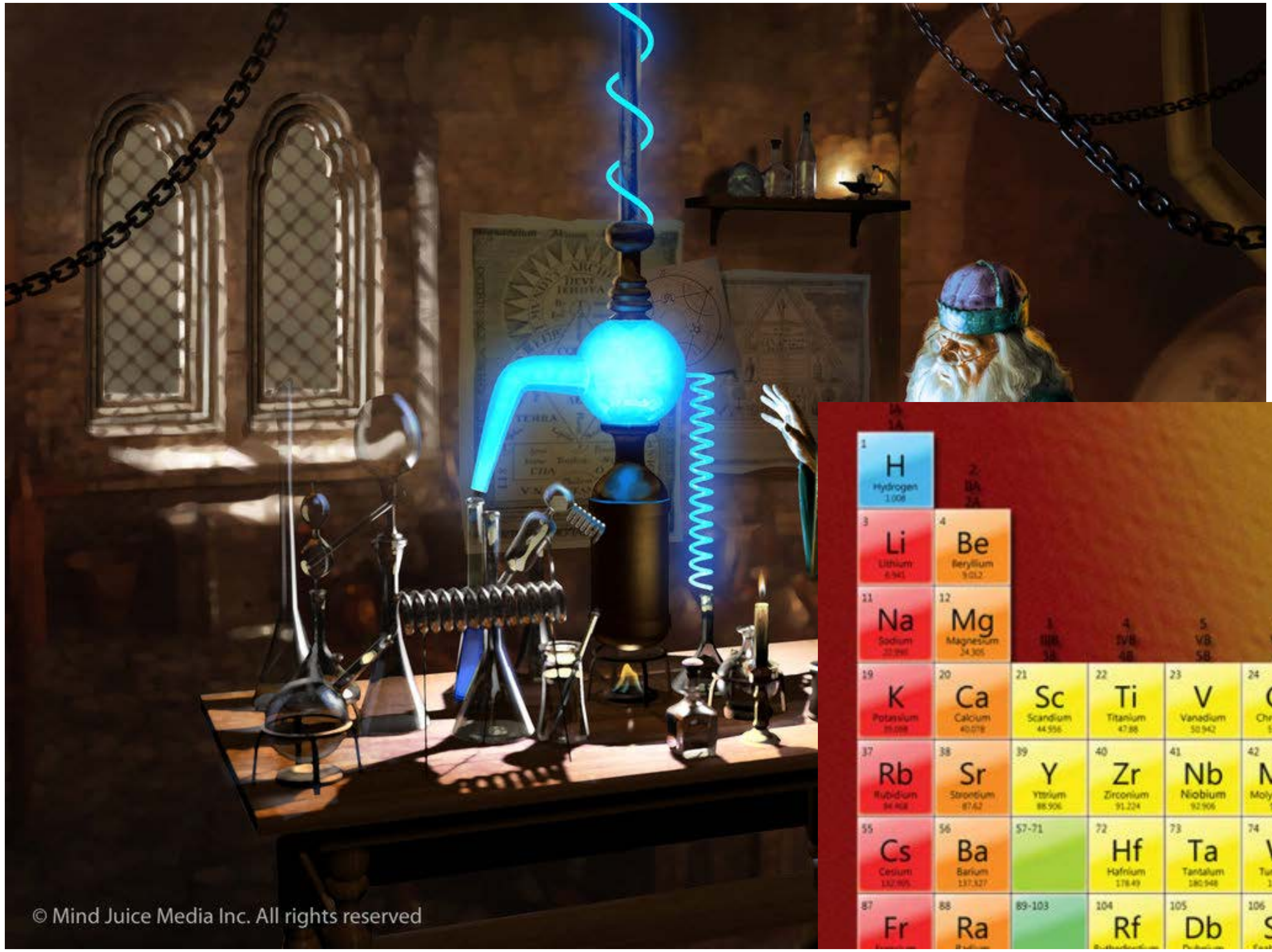
GANs: example results

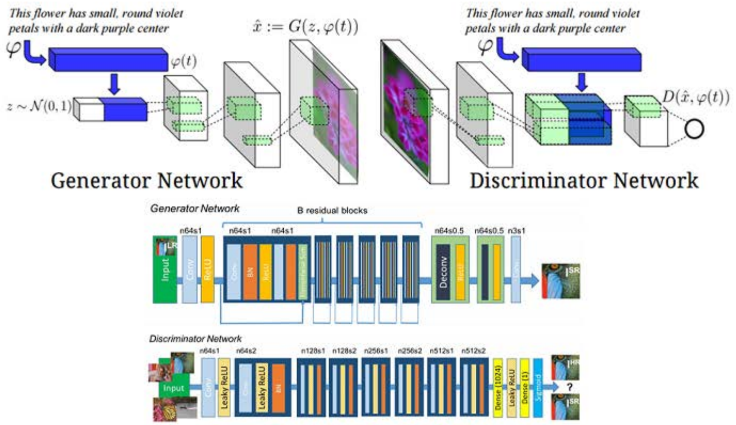
Generated bedrooms [Radford et al., 2016]

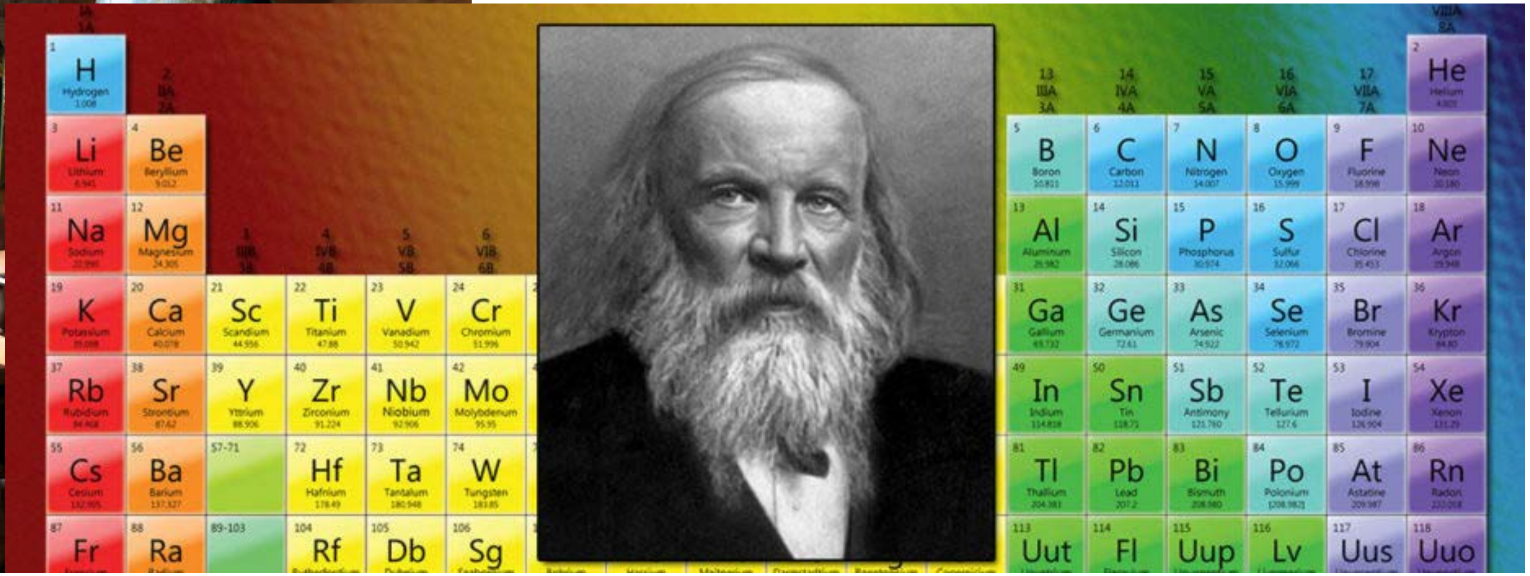
The Zoo of DGMs

- Variational autoencoders (VAEs) [Kingma & Welling, 2014]
  - Adversarial autoencoder [Makhzani et al., 2015]
  - Importance weighted autoencoder [Burda et al., 2015]
  - Implicit variational autoencoder [Mescheder., 2017]
- Generative adversarial networks (GANs) [Goodfellos et al., 2014]
  - InfoGAN [Chen et al., 2016]
  - CycleGAN [Zhu et al., 2017]
  - Wasserstein GAN [Arjovsky et al., 2017]
- Autoregressive neural networks
  - PixelRNN / PixelCNN [Oord et al., 2016]
  - RNN (e.g., for language modeling)
- Generative moment matching networks (GMMNs) [Li et al., 2015; Dziugaite et al., 2015]
- Retricted Boltzmann Machines (RBMs) [Smolensky, 1986]

Alchemy Vs Modern Chemistry







Outline

- Overview of advances in deep generative models
- Theoretical backgrounds of deep generative models
  - Wake sleep algorithm
  - Variational autoencoders
  - Generative adversarial networks
- A unified view of deep generative models
  - new formulations of deep generative models
  - Symmetric modeling of latent and visible variables

Z Hu, Z YANG, R Salakhutdinov, E Xing,
“On Unifying Deep Generative Models”, arxiv 1706.00550

A unified view of deep generative models

- Literatures have viewed these DGM approaches as distinct model training paradigms
  - GANs: achieve an equilibrium between generator and discriminator
  - VAEs: maximize lower bound of the data likelihood

- Let's study a new formulation for DGMs
  - Connects GANs, VAEs, and other variants, under a unified view
  - Links them back to inference and learning of Graphical Models, and the wake-sleep heuristic that approximates this
  - Provides a tool to analyze many GAN-/VAE-based algorithms
  - Encourages mutual exchange of ideas from each individual class of models

Generative Adversarial Nets (GANs):

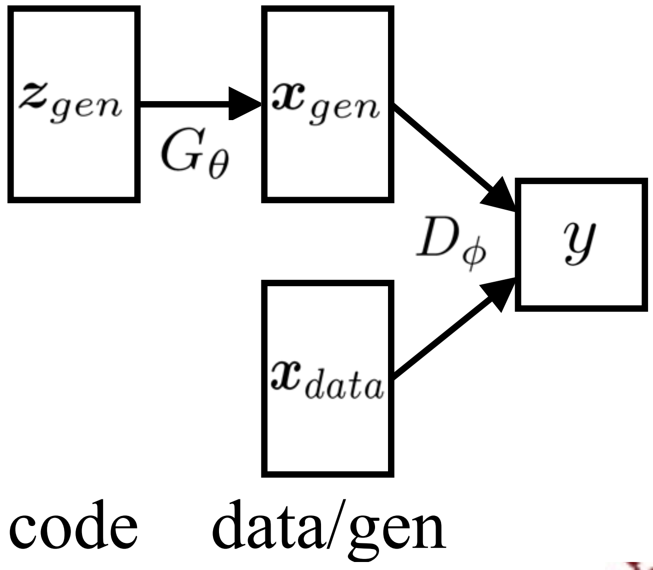
- Implicit distribution over $\boldsymbol{x} \sim p_\theta(\boldsymbol{x}|y)$

$$p_\theta(\boldsymbol{x}|y) = \begin{cases} p_{g_\theta}(\boldsymbol{x}) & y = 0 \\ p_{data}(\boldsymbol{x}) & y = 1. \end{cases}$$

(distribution of generated images)

(distribution of real images)

- $\boldsymbol{x} \sim p_{g_\theta}(\boldsymbol{x}) \Leftrightarrow \boldsymbol{x} = G_\theta(\boldsymbol{z}),\ \boldsymbol{z} \sim p(\boldsymbol{z}|y=0)$
- $\boldsymbol{x} \sim p_{data}(\boldsymbol{x})$
  - the code space of $\boldsymbol{z}$ is degenerated
  - sample directly from data



A new formulation

- Rewrite GAN objectives in the "variational-EM" format
- Recap: conventional formulation:

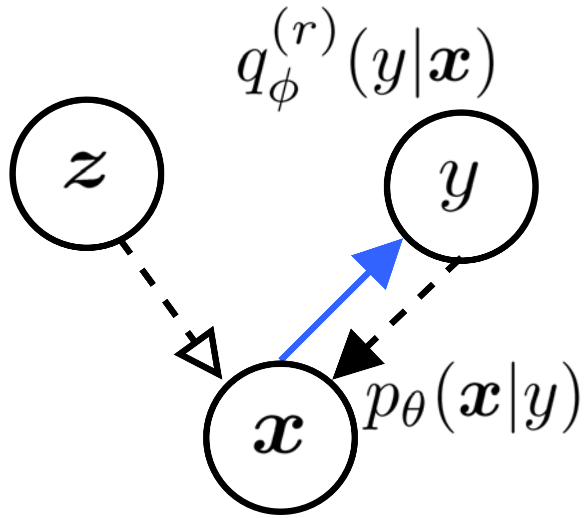
$$\begin{aligned}
\max_{\boldsymbol{\phi}} \mathcal{L}_\phi &= \mathbb{E}_{\boldsymbol{x}=G_\theta(\boldsymbol{z}),\boldsymbol{z}\sim p(\boldsymbol{z}|y=0)}\left[\log(1-D_\phi(\boldsymbol{x}))\right] + \mathbb{E}_{\boldsymbol{x}\sim p_{data}(\boldsymbol{x})}\left[\log D_\phi(\boldsymbol{x})\right] \\
\max_{\boldsymbol{\theta}} \mathcal{L}_\theta &= \mathbb{E}_{\boldsymbol{x}=G_\theta(\boldsymbol{z}),\boldsymbol{z}\sim p(\boldsymbol{z}|y=0)}\left[\log D_\phi(\boldsymbol{x})\right] + \mathbb{E}_{\boldsymbol{x}\sim p_{data}(\boldsymbol{x})}\left[\log(1-D_\phi(\boldsymbol{x}))\right] \\
&= \mathbb{E}_{\boldsymbol{x}=G_\theta(\boldsymbol{z}),\boldsymbol{z}\sim p(\boldsymbol{z}|y=0)}\left[\log D_\phi(\boldsymbol{x})\right]
\end{aligned}$$

- Rewrite in the new form
  - Implicit distribution over $\boldsymbol{x} \sim p_\theta(\boldsymbol{x}|y)$

    $$\boldsymbol{x} = G_\theta(\boldsymbol{z}), \;\; \boldsymbol{z} \sim p(\boldsymbol{z}|y)$$

  - Discriminator distribution $q_\phi(y|\boldsymbol{x})$

    $$q_\phi^r(y|\boldsymbol{x}) = q_\phi(1-y|\boldsymbol{x}) \quad \text{(reverse)}$$



$$\begin{aligned}
\max_{\boldsymbol{\phi}} \mathcal{L}_\phi &= \mathbb{E}_{p_\theta(\boldsymbol{x}|y)p(y)}\left[\log q_\phi(y|\boldsymbol{x})\right] \\
\max_{\boldsymbol{\theta}} \mathcal{L}_\theta &= \mathbb{E}_{p_\theta(\boldsymbol{x}|y)p(y)}\left[\log q_\phi^r(y|\boldsymbol{x})\right]
\end{aligned}$$

GANs vs. Variational EM

Variational EM

- Objectives

$$\max_{\phi}\mathcal{L}_{\boldsymbol{\phi},\boldsymbol{\theta}} = \mathbb{E}_{q_{\phi}(z|x)}[\log p_{\theta}(x|z)] + KL\left(q_{\phi}(z|x)||p(z)\right)$$
$$\max_{\theta}\mathcal{L}_{\boldsymbol{\phi},\boldsymbol{\theta}} = \mathbb{E}_{q_{\phi}(z|x)}[\log p_{\theta}(x|z)] + KL\left(q_{\phi}(z|x)||p(z)\right)$$

  - Single objective for both θ and ϕ
  - Extra prior regularization by $p(z)$
- The reconstruction term: maximize the conditional log-likelihood of x with the generative distribution $p_{\theta}(x|z)$ conditioning on the latent code z inferred by $q_{\phi}(z|x)$
- $p_{\theta}(x|z)$ is the generative model
- $q_{\phi}(z|x)$ is the inference model

GAN

- Objectives

$$\max_{\boldsymbol{\phi}} \mathcal{L}_{\phi} = \mathbb{E}_{p_{\theta}(\boldsymbol{x}|y)p(y)}\left[\log q_{\phi}(y|\boldsymbol{x})\right]$$
$$\max_{\boldsymbol{\theta}} \mathcal{L}_{\theta} = \mathbb{E}_{p_{\theta}(\boldsymbol{x}|y)p(y)}\left[\log q^{r}_{\phi}(y|\boldsymbol{x})\right]$$

  - Two objectives
  - Have global optimal state in the game theoretic view
- The objectives: maximize the conditional log-likelihood of y (or $1-y$) with the distribution $q_{\phi}(y|x)$ conditioning on data/generation x inferred by $p_{\theta}(x|y)$
- Interpret $q_{\phi}(y|x)$ as the generative model
- Interpret $p_{\theta}(x|y)$ as the inference model

GANs vs. Variational EM

- Interpret x as latent variables
- Interpret generation of x as performing inference over latent

In VEM, we minimize the following:

$$F(\boldsymbol{\theta}, \boldsymbol{\phi}; \boldsymbol{x}) = -\log p(\boldsymbol{x}) + KL\left(q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}) \,||\, p_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x})\right)$$

→ KL (inference model | posterior)

Variational EM

- Objectives

$$\max_{\phi}\mathcal{L}_{\boldsymbol{\phi},\boldsymbol{\theta}} = \mathbb{E}_{q_{\boldsymbol{\phi}}(z|x)}[\log p_{\theta}(x|z)] + KL\left(q_{\boldsymbol{\phi}}(z|x)||p(z)\right)$$

$$\max_{\theta}\mathcal{L}_{\boldsymbol{\phi},\boldsymbol{\theta}} = \mathbb{E}_{q_{\boldsymbol{\phi}}(z|x)}[\log p_{\theta}(x|z)] + KL\left(q_{\boldsymbol{\phi}}(z|x)||p(z)\right)$$

  - Single objective for both θ and ϕ
  - Extra prior regularization by $p(z)$
- The reconstruction term: maximize the conditional log-likelihood of x with the generative distribution $p_\theta(x|z)$ conditioning on the latent code z inferred by $q_\phi(z|x)$

- $p_\theta(x|z)$ is the generative model
- $q_\phi(z|x)$ is the inference model

GAN

- Objectives

$$\max_{\boldsymbol{\phi}} \mathcal{L}_{\phi} = \mathbb{E}_{p_\theta(\boldsymbol{x}|y)p(y)}\left[\log q_\phi(y|\boldsymbol{x})\right]$$

$$\max_{\boldsymbol{\theta}} \mathcal{L}_{\theta} = \mathbb{E}_{p_\theta(\boldsymbol{x}|y)p(y)}\left[\log q_\phi^r(y|\boldsymbol{x})\right]$$

  - Two objectives
  - Have global optimal state in the game theoretic view
- The objectives: maximize the conditional log-likelihood of y (or $1-y$) with the distribution $q_\phi(y|x)$ conditioning on data/generation x inferred by $p_\theta(x|y)$

- Interpret $q_\phi(y|x)$ as the generative model
- Interpret $p_\theta(x|y)$ as the inference model

GANs: minimizing KLD

- As in Variational EM, we can further rewrite in the form of minimizing KLD to reveal more insights into the optimization problem
- For each optimization step of $p_\theta(\boldsymbol{x}|y)$ at point $(\theta = \theta_0, \phi = \phi_0)$, let
  - $p(y)$: uniform prior distribution
  - $p_{\theta=\theta_0}(\boldsymbol{x}) = \mathrm{E}_{p(y)}\left[p_{\theta=\theta_0}(\boldsymbol{x}|y)\right]$
  - $q^r(\boldsymbol{x}|y) \propto q^r_{\phi=\phi_0}(y|\boldsymbol{x}) p_{\theta=\theta_0}(\boldsymbol{x})$
- *Lemma 1*: The updates of $\boldsymbol{\theta}$ at $\boldsymbol{\theta}_0$ have

$$\nabla_\theta \left[-\mathbb{E}_{p_\theta(\boldsymbol{x}|y)p(y)} \left[\log q^r_{\phi=\phi_0}(y|\boldsymbol{x})\right] \right] \Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}_0} =$$

$$\nabla_\theta \left[\mathbb{E}_{p(y)} \left[KL\left(p_\theta(\boldsymbol{x}|y) \| q^r(\boldsymbol{x}|y)\right)\right] - JSD\left(p_\theta(\boldsymbol{x}|y=0) \| p_\theta(\boldsymbol{x}|y=1)\right) \right] \Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}_0}$$

  - KL: KL divergence
  - JSD: Jensen-shannon divergence

GANs: minimizing KLD

- *Lemma 1*: The updates of $\boldsymbol{\theta}$ at $\boldsymbol{\theta}_0$ have

$$\nabla_\theta \Big[- \mathbb{E}_{p_\theta(\boldsymbol{x}|y)p(y)} \left[\log q^r_{\phi=\phi_0}(y|\boldsymbol{x}) \right] \Big] \Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}_0} =$$

$$\nabla_\theta \Big[\mathbb{E}_{p(y)} \left[\mathrm{KL}\left(p_\theta(\boldsymbol{x}|y) \| q^r(\boldsymbol{x}|y) \right) \right] - \mathrm{JSD}\left(p_\theta(\boldsymbol{x}|y=0) \| p_\theta(\boldsymbol{x}|y=1) \right) \Big] \Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}_0}$$

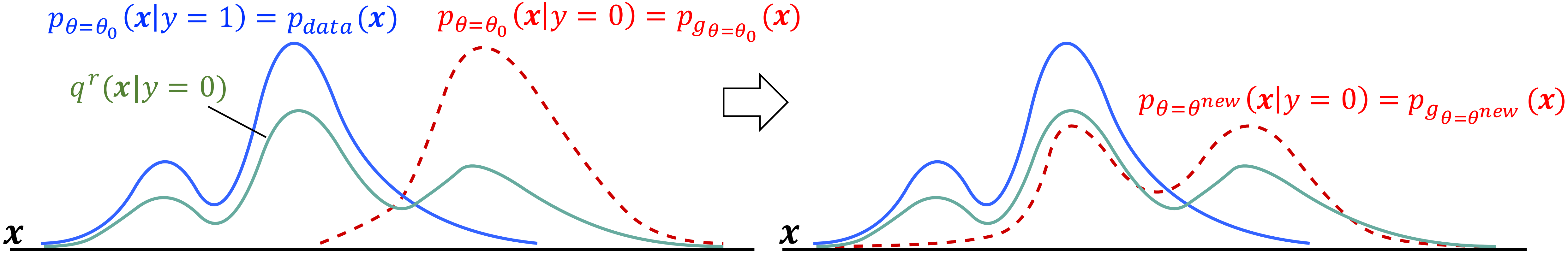
- Connection to variational inference
  - See $\boldsymbol{x}$ as latent variables, y as visible
  - $p_{\theta=\theta_0}(\boldsymbol{x})$: prior distribution
  - $q^r(\boldsymbol{x}|y) \propto q^r_{\phi=\phi_0}(y|\boldsymbol{x}) p_{\theta=\theta_0}(\boldsymbol{x})$: posterior distribution
  - $p_\theta(\boldsymbol{x}|y)$: variational distribution
    - Amortized inference: updates model parameter $\boldsymbol{\theta}$
- Suggests relations to VAEs, as we will explore shortly

In VEM, we minimize the following:

$$F(\boldsymbol{\theta}, \boldsymbol{\phi}; \boldsymbol{x}) = -\log p(\boldsymbol{x}) + KL\left(q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}) \,||\, p_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x}) \right)$$

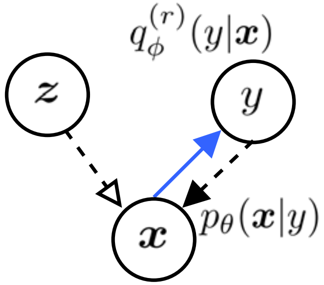
→ KL (inference model | posterior)

GANs: minimizing KLD

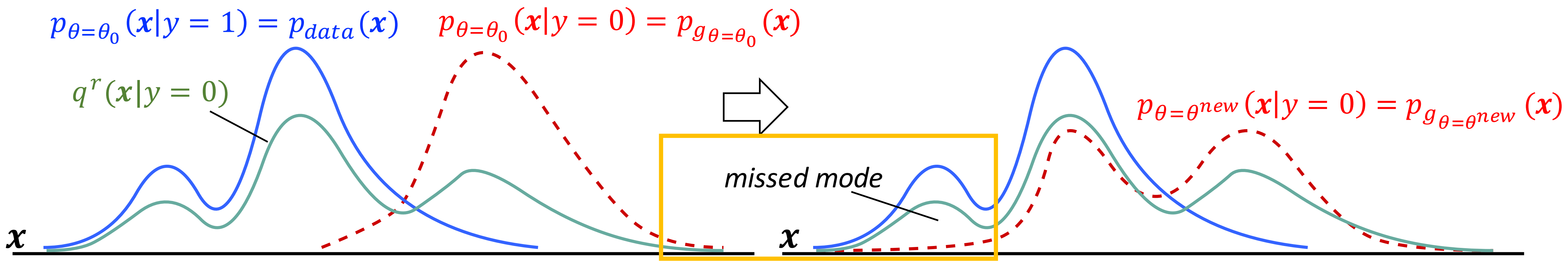


- Minimizing the KLD drives $p_{g_\theta}(\boldsymbol{x})$ to $p_{data}(\boldsymbol{x})$
  - By definition: $p_{\theta=\theta_0}(\boldsymbol{x}) = \mathrm{E}_{p(y)}[p_{\theta=\theta_0}(\boldsymbol{x}|y)] = \left(p_{g_{\theta=\theta_0}}(\boldsymbol{x}) + p_{data}(\boldsymbol{x})\right) / 2$
  - $\mathrm{KL}\big(p_\theta(x|y=1)||q^r(x|y=1)\big) = \mathrm{KL}\big(p_{data}(x)||q^r(x|y=1)\big)$: constant, no free parameters
  - $\mathrm{KL}\big(p_\theta(x|y=0)||q^r(x|y=0)\big) = \mathrm{KL}\left(p_{g_\theta}(x)||q^r(x|y=0)\right)$: parameter θ to optimize
    - $q^r(\boldsymbol{x}|y=0) \propto q^r_{\phi=\phi_0}(y=0|\boldsymbol{x})p_{\theta=\theta_0}(\boldsymbol{x})$
      - seen as a mixture of $p_{g_{\theta=\theta_0}}(\boldsymbol{x})$ and $p_{data}(\boldsymbol{x})$
      - mixing weights induced from $q^r_{\phi=\phi_0}(y=0|\boldsymbol{x})$
    - Drives $p_{g_\theta}(\boldsymbol{x}|y)$ to mixture of $p_{g_{\theta=\theta_0}}(\boldsymbol{x})$ and $p_{data}(\boldsymbol{x})$

      $\Rightarrow$ Drives $p_{g_\theta}(\boldsymbol{x})$ to $p_{data}(\boldsymbol{x})$



GANs: minimizing KLD



- Missing mode phenomena of GANs
  - Asymmetry of KLD
    - Concentrates $p_\theta(\boldsymbol{x}|y=0)$ to large modes of $q^r(\boldsymbol{x}|y)$

      $\Rightarrow p_{g_\theta}(\boldsymbol{x})$ misses modes of $p_{data}(\boldsymbol{x})$
  - Symmetry of JSD
    - Does not affect the behavior of mode missing

$$\mathrm{KL}\left(p_{g_\theta}(x) || q^r(x|y=0)\right)$$
$$= \int p_{g_\theta}(x) \log \frac{p_{g_\theta}(x)}{q^r(x|y=0)} dx$$

- **Large positive contribution to the KLD in the regions of x space where $q^r(x|y=0)$ is small, unless $p_{g_\theta}(x)$ is also small**
- **$\Rightarrow p_{g_\theta}(x)$ tends to avoid regions where $q^r(x|y=0)$ is small**

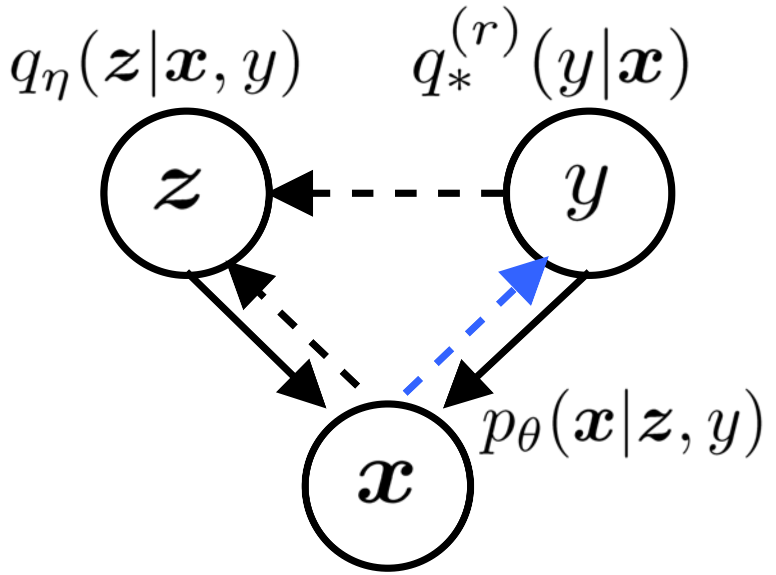
Recap: conventional formulation of VAEs

- Objective:

$$\max_{\boldsymbol{\theta},\boldsymbol{\eta}} \mathcal{L}_{\theta,\eta}^{\text{vae}} = \mathbb{E}_{p_{data}(\boldsymbol{x})}\left[\mathbb{E}_{\tilde{q}_\eta(\boldsymbol{z}|\boldsymbol{x})}\left[\log \tilde{p}_\theta(\boldsymbol{x}|\boldsymbol{z})\right] - \mathrm{KL}(\tilde{q}_\eta(\boldsymbol{z}|\boldsymbol{x}) \| \tilde{p}(\boldsymbol{z}))\right]$$

  - $\tilde{p}(\boldsymbol{z})$: prior over $\boldsymbol{z}$
  - $\tilde{p}_\theta(\boldsymbol{x}|\boldsymbol{z})$: generative model
  - $\tilde{q}_\eta(\boldsymbol{z}|\boldsymbol{x})$: inference model
  - Only uses real examples from $p_{data}(\boldsymbol{x})$, lacks adversarial mechanism
- To align with GANs, let's introduce the real/fake indicator y and adversarial discriminator

VAEs: new formulation



- Assume a *perfect* discriminator $q_*(y|\boldsymbol{x})$
  - $q_*(y = 1|\boldsymbol{x}) = 1$ if $\boldsymbol{x}$ is real examples
  - $q_*(y = 0|\boldsymbol{x}) = 1$ if $\boldsymbol{x}$ is generated samples
  - $q_*^r(y|\boldsymbol{x}) := q_*(1 - y|\boldsymbol{x})$
- Generative distribution

$$p_\theta(\boldsymbol{x}|\boldsymbol{z}, y) = \begin{cases} p_\theta(\boldsymbol{x}|\boldsymbol{z}) & y = 0 \\ p_{data}(\boldsymbol{x}) & y = 1. \end{cases}$$

- Let $p_\theta(\boldsymbol{z}, y|\boldsymbol{x}) \propto p_\theta(\boldsymbol{x}|\boldsymbol{z}, y)p(\boldsymbol{z}|y)p(y)$
- *Lemma 2*

$$\begin{aligned}\mathcal{L}_{\theta,\eta}^{vae} &= 2 \cdot \mathbb{E}_{p_{\theta_0}(\boldsymbol{x})}\left[\mathbb{E}_{q_\eta(\boldsymbol{z}|\boldsymbol{x},y)q_*^r(y|\boldsymbol{x})}\left[\log p_\theta(\boldsymbol{x}|\boldsymbol{z}, y)\right] - KL(q_\eta(\boldsymbol{z}|\boldsymbol{x}, y)q_*^r(y|\boldsymbol{x})\|p(\boldsymbol{z}|y)p(y))\right] \\ &= 2 \cdot \mathbb{E}_{p_{\theta_0}(\boldsymbol{x})}\left[-KL\left(q_\eta(\boldsymbol{z}|\boldsymbol{x}, y)q_*^r(y|\boldsymbol{x})\|p_\theta(\boldsymbol{z}, y|\boldsymbol{x})\right)\right].\end{aligned}$$

GANs vs VAEs side by side

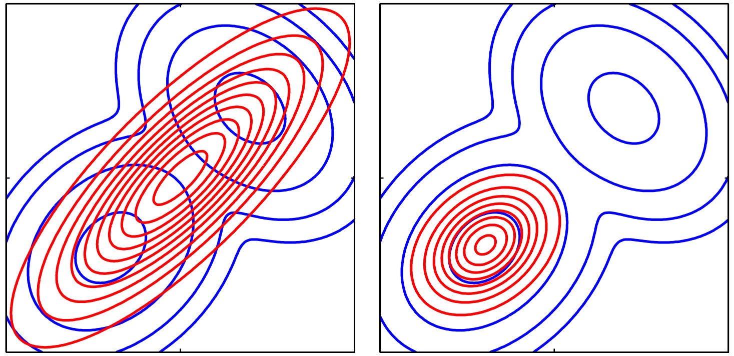
$p_\theta(\boldsymbol{z}, y|\boldsymbol{x}) \propto p_\theta(\boldsymbol{x}|\boldsymbol{z}, y)p(\boldsymbol{z}|y)p(y)$

	GANs (InfoGAN)	VAEs
Generative distribution	$p_\theta(\boldsymbol{x}\|y) = \begin{cases} p_{g_\theta}(\boldsymbol{x}) & y=0 \\ p_{data}(\boldsymbol{x}) & y=1. \end{cases}$	$p_\theta(\boldsymbol{x}\|\boldsymbol{z}, y) = \begin{cases} p_\theta(\boldsymbol{x}\|\boldsymbol{z}) & y=0 \\ p_{data}(\boldsymbol{x}) & y=1. \end{cases}$
Discriminator distribution	$q_\phi(y\|\boldsymbol{x})$	$q_*(y\|\boldsymbol{x})$, perfect, degenerated
$\boldsymbol{z}$-inference model	$q_\eta(\boldsymbol{z}\|\boldsymbol{x}, y)$ of InfoGAN	$q_\eta(\boldsymbol{z}\|\boldsymbol{x}, y)$
KLD to minimize	$\min_\theta \mathrm{KL}\left(p_\theta(\boldsymbol{x}\|y) \,\|\|\, q^r(\boldsymbol{x}\|\boldsymbol{z}, y)\right)$ $\sim \min_\theta \mathrm{KL}(P_\theta \,\|\|\, Q)$	$\min_\theta \mathrm{KL}\left(q_\eta(\boldsymbol{z}\|\boldsymbol{x}, y)q_*^r(y\|\boldsymbol{x}) \,\|\|\, p_\theta(\boldsymbol{z}, y\|\boldsymbol{x})\right)$ $\sim \min_\theta \mathrm{KL}(Q \,\|\|\, P_\theta)$

GANs vs VAEs side by side

	GANs (InfoGAN)	VAEs
KLD to minimize	$\min_\theta \mathrm{KL}\left(p_\theta(\boldsymbol{x}\|y) \,\|\|\, q^r(\boldsymbol{x}\|\boldsymbol{z}, y)\right)$ $\sim \min_\theta \mathrm{KL}(P_\theta \,\|\|\, Q)$	$\min_\theta \mathrm{KL}(q_\eta(\boldsymbol{z}\|\boldsymbol{x}, y) q_*^r(y\|\boldsymbol{x}) \,\|\|\, p_\theta(\boldsymbol{z}, y\|\boldsymbol{x}))$ $\sim \min_\theta \mathrm{KL}(Q \,\|\|\, P_\theta)$

- Asymmetry of KLDs inspires combination of GANs and VAEs
  - GANs: $\min_\theta \mathrm{KL}(P_\theta || Q)$ tends to missing mode
  - VAEs: $\min_\theta \mathrm{KL}(Q || P_\theta)$ tends to cover regions with small values of p_{data}



Mode covering Mode missing

[Figure courtesy: PRML]

Link back to wake sleep algorithm

- Denote
  - Latent variables $\boldsymbol{h}$
  - Parameters $\boldsymbol{\lambda}$
- Recap: wake sleep algorithm

$$\begin{aligned} \text{Wake}: &\quad \max_{\boldsymbol{\theta}} \mathbb{E}_{q_\lambda(\boldsymbol{h}|\boldsymbol{x})p_{data}(\boldsymbol{x})}\left[\log p_\theta(\boldsymbol{x}|\boldsymbol{h})\right] \\ \text{Sleep}: &\quad \max_{\boldsymbol{\lambda}} \mathbb{E}_{p_\theta(\boldsymbol{x}|\boldsymbol{h})p(\boldsymbol{h})}\left[\log q_\lambda(\boldsymbol{h}|\boldsymbol{x})\right] \end{aligned}$$

VAEs vs. Wake-sleep

- Wake sleep algorithm

$$\text{Wake}: \quad \max_{\boldsymbol{\theta}} \mathbb{E}_{q_\lambda(\boldsymbol{h}|\boldsymbol{x})p_{data}(\boldsymbol{x})} [\log p_\theta(\boldsymbol{x}|\boldsymbol{h})]$$

$$\text{Sleep}: \quad \max_{\boldsymbol{\lambda}} \mathbb{E}_{p_\theta(\boldsymbol{x}|\boldsymbol{h})p(\boldsymbol{h})} [\log q_\lambda(\boldsymbol{h}|\boldsymbol{x})]$$

- Let $\boldsymbol{h}$ be $\boldsymbol{z}$, and $\boldsymbol{\lambda}$ be $\boldsymbol{\eta}$

$\Rightarrow \max_{\boldsymbol{\theta}} \mathbb{E}_{q_\eta(\boldsymbol{z}|\boldsymbol{x})p_{data}(\boldsymbol{x})} [\log p_\theta(\boldsymbol{x}|\boldsymbol{z})]$, recovers VAE objective of optimizing $\boldsymbol{\theta}$

- VAEs extend wake phase by also learning the inference model (η)

$$\max_{\boldsymbol{\theta},\boldsymbol{\eta}} \mathcal{L}^{\text{vae}}_{\theta,\eta} = \mathbb{E}_{q_\eta(\boldsymbol{z}|\boldsymbol{x})p_{data}(\boldsymbol{x})} [\log p_\theta(\boldsymbol{x}|\boldsymbol{z})] - \mathbb{E}_{p_{data}(\boldsymbol{x})} [\mathrm{KL}(q_\eta(\boldsymbol{z}|\boldsymbol{x}) \| p(\boldsymbol{z}))]$$

  - Minimize the KLD in the original variational free energy wrt. $\boldsymbol{\eta}$
  - Stick to minimizing the wake-phase KLD wrt. both $\boldsymbol{\theta}$ and $\boldsymbol{\eta}$
  - Do not involve sleep-phase objective
  - Recall: sleep phase minimizes the *reverse* KLD in the variational free energy

GANs vs. Wake-sleep

- Wake sleep algorithm

$$\text{Wake}: \quad \max_{\boldsymbol{\theta}} \mathbb{E}_{q_\lambda(\boldsymbol{h}|\boldsymbol{x})p_{data}(\boldsymbol{x})} \left[\log p_\theta(\boldsymbol{x}|\boldsymbol{h})\right]$$

$$\text{Sleep}: \quad \max_{\boldsymbol{\lambda}} \mathbb{E}_{p_\theta(\boldsymbol{x}|\boldsymbol{h})p(\boldsymbol{h})} \left[\log q_\lambda(\boldsymbol{h}|\boldsymbol{x})\right]$$

- Let $\boldsymbol{h}$ be y, and $\boldsymbol{\lambda}$ be $\boldsymbol{\phi}$

$\Rightarrow \quad \max_{\boldsymbol{\phi}} \mathbb{E}_{p_\theta(\boldsymbol{x}|y)p(y)} \left[\log q_\phi(y|\boldsymbol{x})\right]$, recovers GAN objective of optimizing $\boldsymbol{\phi}$

- GANs extend sleep phase by also learning the generative model ($\boldsymbol{\theta}$)
  - Directly extending sleep phase: $\max_{\boldsymbol{\phi}} \mathcal{L}_\phi = \mathbb{E}_{p_\theta(\boldsymbol{x}|y)p(y)} \left[\log q_\phi(y|\boldsymbol{x})\right]$
  - GANs: $\max_{\boldsymbol{\theta}} \mathcal{L}_\theta = \mathbb{E}_{p_\theta(\boldsymbol{x}|y)p(y)} \left[\log q^r_\phi(y|\boldsymbol{x})\right]$
  - The only difference is replacing $q_{\boldsymbol{\phi}}$ with $q^r_{\boldsymbol{\phi}}$
  - This is where adversarial mechanism come about !
  - GANs stick to minimizing the sleep-phase KLD
  - Do not involve wake-phase objective

Conclusions

Z Hu, Z YANG, R Salakhutdinov, E Xing,
"On Unifying Deep Generative Models", arxiv 1706.00550

- Deep generative models research have a long history
  - Deep belief nets / Helmholtz machines / Predictability Minimization / …

- Unification of deep generative models
  - GANs and VAEs are essentially minimizing KLD in opposite directions
    - Extends two phases of classic wake sleep algorithm, respectively
  - A general formulation framework useful for
    - Analyzing broad class of existing DGM and variants: ADA/InfoGAN/Joint-models/…
    - Inspiring new models and algorithms by borrowing ideas across research fields