Toward Deep Statistical Speaker Representation

Dong Wang

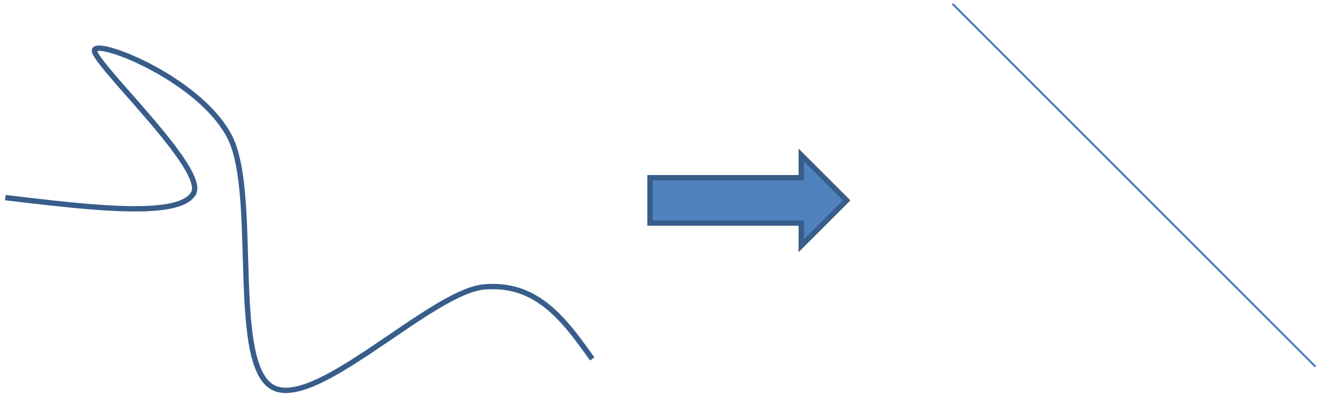
2018/12/26

Content

- Theory introduction
- Application to speaker recognition

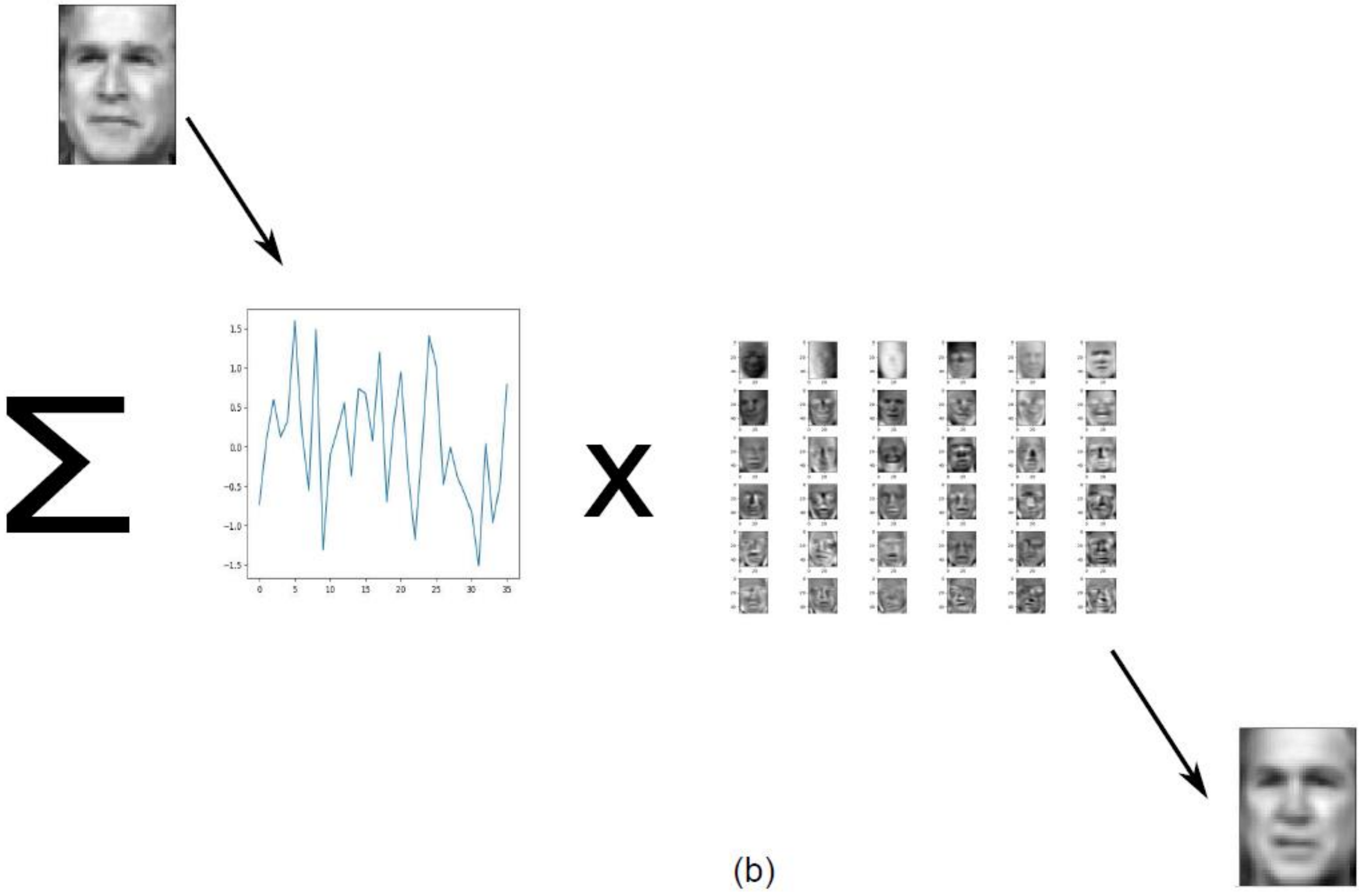
How we represent data?

- Data observed are always noisy
- We need a way to extract abstract representation
  - Representation is the first requirement
  - Second is invariance, which is task-dependent

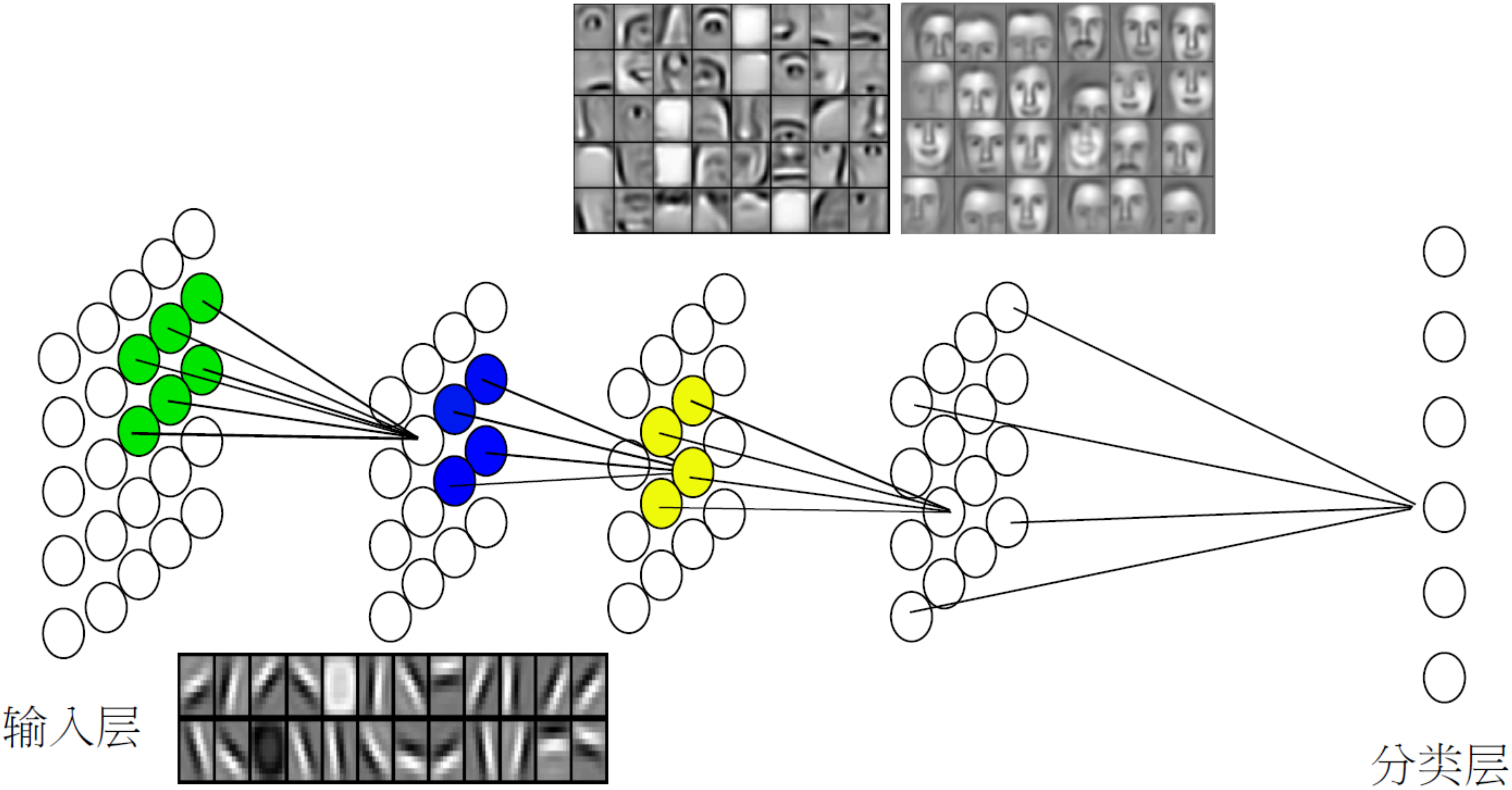


Methods known

- Heuristic approach (strong knowledge)
  - FFT, geometric features, histogram…
- Bayesian approach (medium knowledge)
  - PCA, FA, clustering, tSNE…
  - HMM, hierarchical Bayesian
- Neural model approach (weak knowledge)
  - Deep feature learning



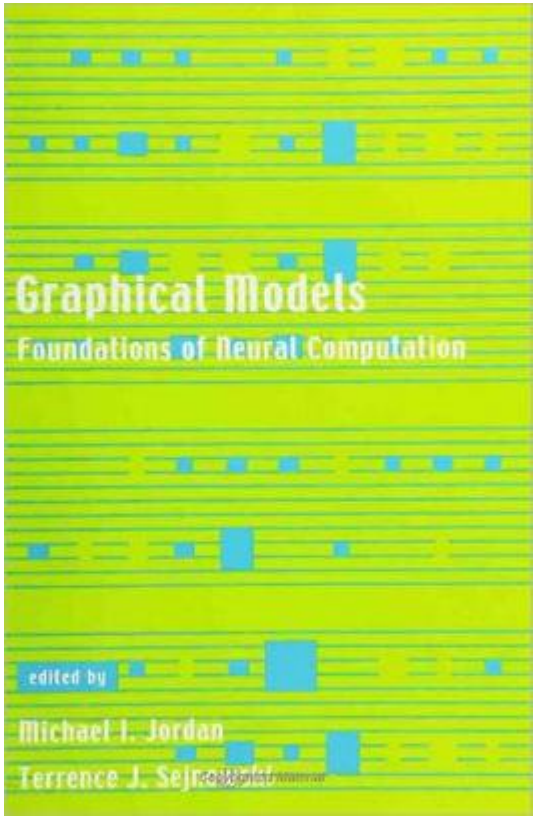
(b)

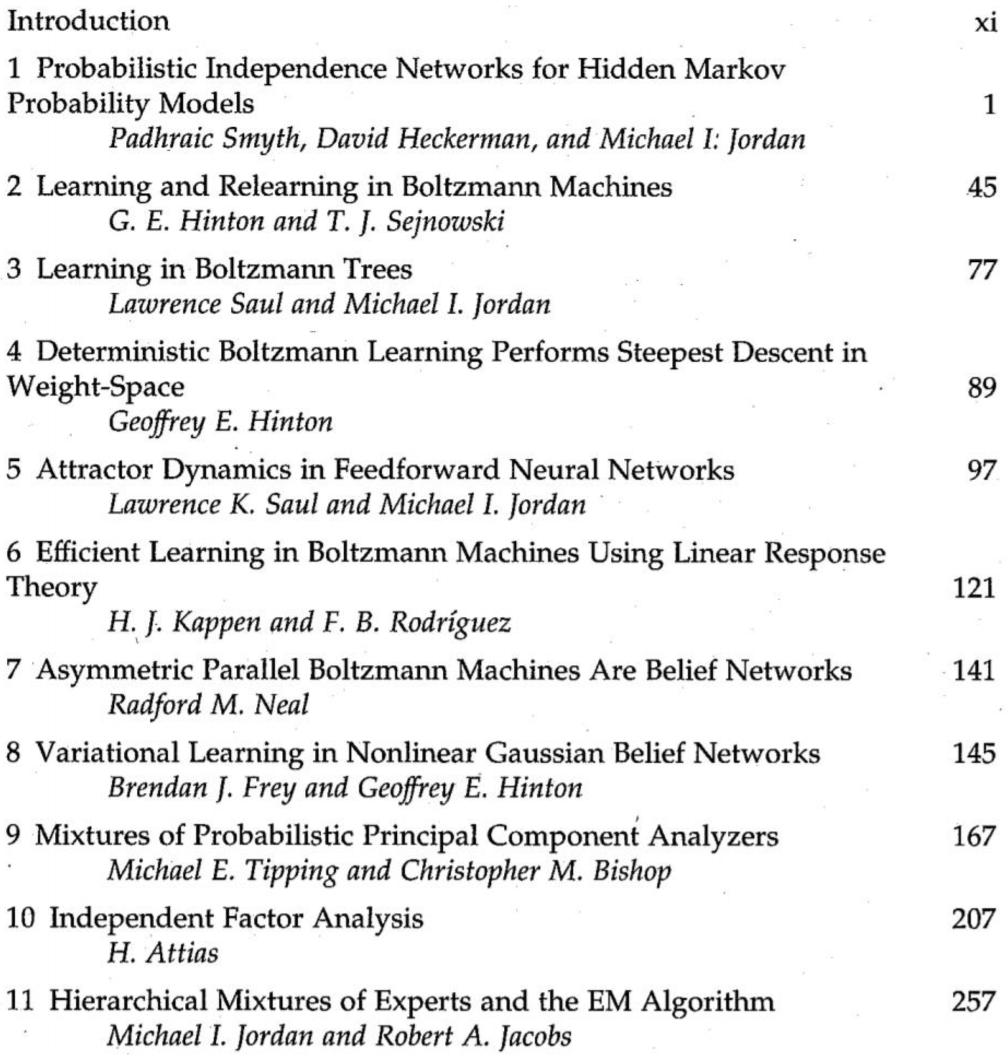

输入层
分类层

Both are with limitations

- Bayesian approach is basically shallow, otherwise the structure will be complex.
- Neural net approach is knowledge-blind, whose actions is not predictable.
- How they can be combined?

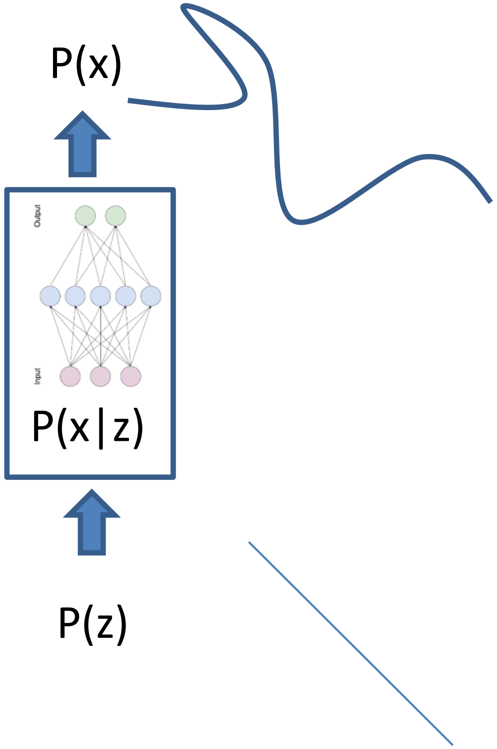
They are historically combined





A key idea for combination

- Some attempts: tandem, stochastic neural net.
- Modern: keep the Bayesian framework, but make the conditional probability complex using neural nets.
- Infer latent variables by neural nets.



P(x)

P(x|z)

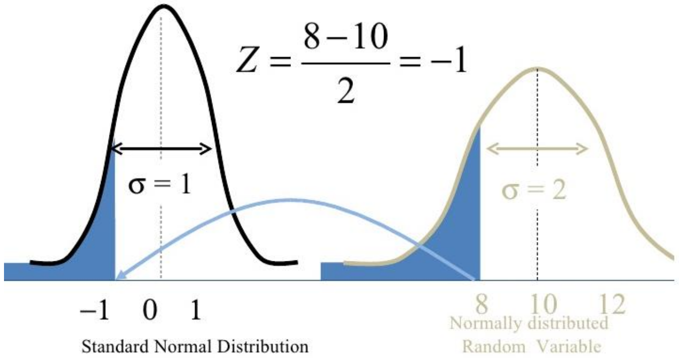
P(z)

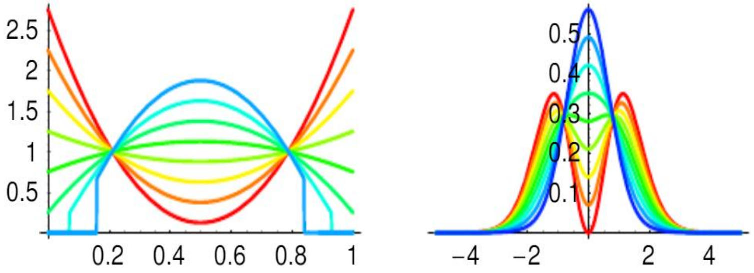
Goodness

- Very complex distributions can be generated from a simple distribution (Re-parameterization trick).

$$\varrho_Y(y) = \begin{cases} 0, & \text{if } y \notin f(\mathbb{R}) \\ \varrho_X(f^{-1}(y)) \cdot \left| \frac{df^{-1}(y)}{dy} \right|, & \text{if } y \in f(\mathbb{R}) \end{cases}$$

$$G_{R_{12}}(u) = \min[\max[u + \gamma u(1-u)\left(u - \frac{1}{2}\right), 0], 1].$$





William T. Shaw, Ian Buckley, The alchemy of probability distributions: beyond Gram-Charlier expansions, and a skew-kurtotic-normal distribution from a rank transmutation map,2009.

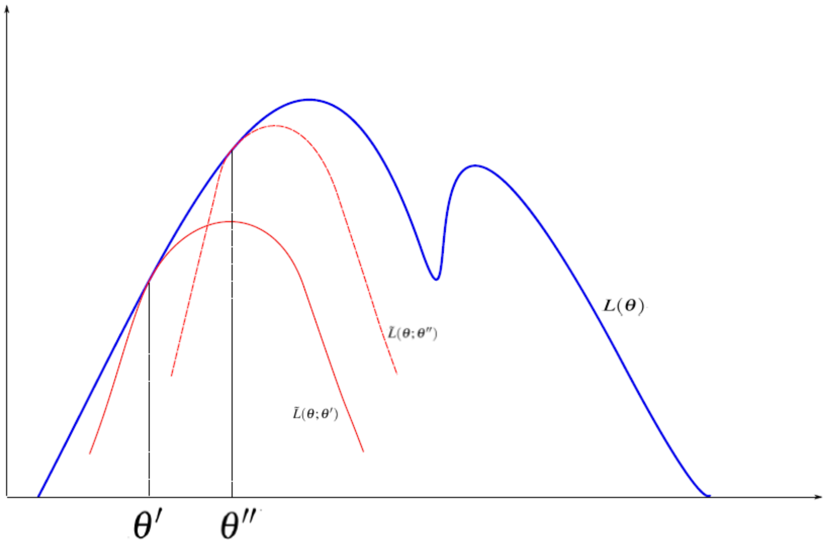
Goal and Difficulties

- Our goal:
  - Training: Estimate parameter for $p_{\boldsymbol{\theta}}(\mathbf{x})$
  - Inference: Estimate posterior $p_{\boldsymbol{\theta}^*}(\mathbf{x}|\mathbf{z})$
  - Difficulties
    - Computing p(x) and p(z|x) is hard.

Let’s back to EM

$$L(\boldsymbol{\theta}) = \sum^{N} \ln p(\mathbf{x}_n; \boldsymbol{\theta}).$$

$$\begin{aligned}
L(\boldsymbol{\theta}) &= \sum_n \sum_{\mathbf{z}} q(\mathbf{z}) \ln p(\mathbf{x}_n) \\
&= \sum_n \sum_{\mathbf{z}} q(\mathbf{z}) \ln \frac{p(\mathbf{x}_n, \mathbf{z})}{p(\mathbf{z}|\mathbf{x}_n)} \\
&= \sum_n \sum_{\mathbf{z}} q(\mathbf{z}) \ln \frac{p(\mathbf{x}_n, \mathbf{z})}{q(\mathbf{z})} \frac{q(\mathbf{z})}{p(\mathbf{z}|\mathbf{x}_n)} \\
&= \sum_n \sum_{\mathbf{z}} q(\mathbf{z}) \ln \frac{p(\mathbf{x}_n, \mathbf{z})}{q(\mathbf{z})} + \sum_n \sum_{\mathbf{z}} q(\mathbf{z}) \ln \frac{q(\mathbf{z})}{p(\mathbf{z}|\mathbf{x}_n)} \\
&\boxed{= \tilde{L}(\boldsymbol{\theta}) + \sum_n KL(q(\mathbf{z}) || p(\mathbf{z}|\mathbf{x}_n))}
\end{aligned}$$



$$\boxed{q(\mathbf{z}) = p(\mathbf{z}|\mathbf{x}; \boldsymbol{\theta}')}$$

$$\boxed{\tilde{L}(\boldsymbol{\theta}) = \sum_n \sum_{\mathbf{z}} q(\mathbf{z}) \ln \frac{p(\mathbf{x}_n, \mathbf{z})}{q(\mathbf{z})}.}$$

helmholtz energy Energy
KL(q||p)

Infer p(z|x,θ’)

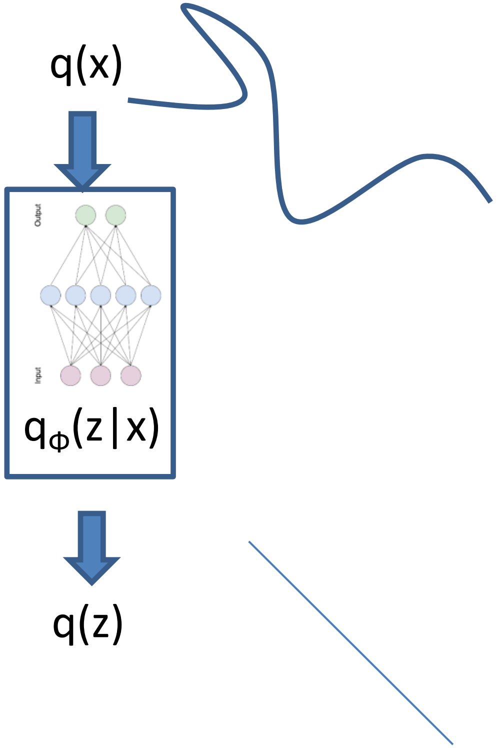
- How if we cannot compute posterior?
- Sampling
  - Very slow, as for each data point you need a MCMC
- Mean-field Variational Bayesian
  - P(x,z) difficult to marginalize

$$q(\mathbf{z}) = \prod_{i=1}^{M} q_i(z_i)$$

$$q_j^*(z_j) = \frac{exp(\mathbb{E}_{i\neq j}[\ln p(\mathbf{x},\mathbf{z})])}{\sum_{z_j} exp(\mathbb{E}_{i\neq j}[\ln p(\mathbf{x},\mathbf{z})])}$$

A flexible parametric variational bound

- $p_\theta(z|x) \approx q_\phi(z|x)$, $q_\phi(z|x)$ can be anything
- Seems neural net is a good selection



$q(x)$

$q_\phi(z|x)$

$q(z)$

But how to determine $q_{\phi}(\mathbf{z}|\mathbf{x})$?

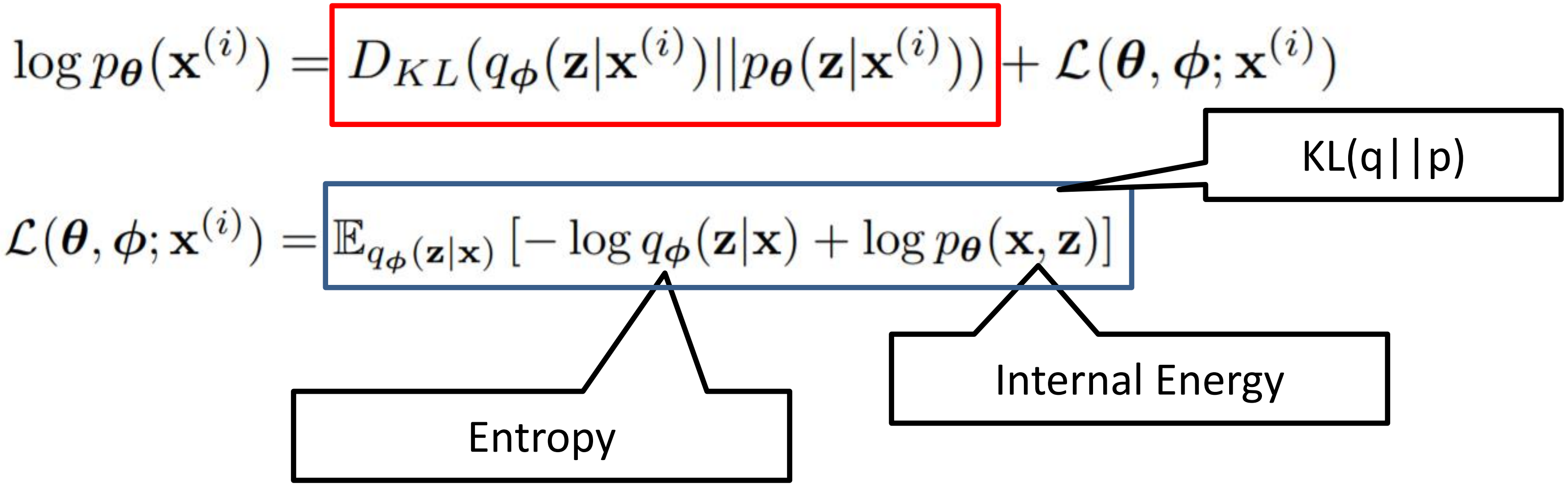
$$\log p_{\boldsymbol{\theta}}(\mathbf{x}^{(i)}) = \boxed{D_{KL}(q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}^{(i)})||p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x}^{(i)}))} + \mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}^{(i)})$$

KL(q||p)

$$\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}^{(i)}) = \boxed{\mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})}\left[\underbrace{-\log q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})}_{\text{Entropy}} + \underbrace{\log p_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{z})}_{\text{Internal Energy}}\right]}$$

- For a fixed θ, minimize $q_{\phi}(z)$ equals to maximizing bound L(θ,Φ) w.r.t. ϕ.
- The bound is a combination of internal energy plus an energy (**helmholtz**). Maximization of this function equal to minimize the internal energy with a maximum entropy regularization.
- In other words, we want z generated by q has the lowest energy, but we also hope the probability of z has maximum entropy, which leads to a stable thermodynamic system.
- Note that the best q can be derived from p in the case of Boltzmann distribution. However, since partial integration is intractable, this (as mean-field VB did) is not possible.

But how to determine $q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})$?



$$\log p_{\boldsymbol{\theta}}(\mathbf{x}^{(i)}) = \boxed{D_{KL}(q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}^{(i)})||p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x}^{(i)}))} + \mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}^{(i)})$$

KL(q||p)

$$\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}^{(i)}) = \boxed{\mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})}\left[-\log q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}) + \log p_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{z})\right]}$$

Internal Energy

Entropy

- What we can do is simply a gradient approach.
- Fox a fixed θ, compute gradient of L(θ,Φ) w.r.t. Φ.
- Note the gradient variable is on the distribution of expectation. A little tricky but still possible.
- However, it requires samples from $q_\Phi(z|x)$, which assumed to be hard.
- Re-parameterization: using a simple distribution p(ε) to produce a complex distribution $q_\Phi(z|x)$.

$$\widetilde{\mathbf{z}} = g_{\boldsymbol{\phi}}(\boldsymbol{\epsilon}, \mathbf{x}) \quad \text{with} \quad \boldsymbol{\epsilon} \sim p(\boldsymbol{\epsilon})$$

$$\widetilde{\mathcal{L}}^{A}(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}^{(i)}) = \frac{1}{L}\sum_{l=1}^{L} \log p_{\boldsymbol{\theta}}(\mathbf{x}^{(i)}, \mathbf{z}^{(i,l)}) - \log q_{\boldsymbol{\phi}}(\mathbf{z}^{(i,l)}|\mathbf{x}^{(i)})$$

$$\text{where} \quad \mathbf{z}^{(i,l)} = g_{\boldsymbol{\phi}}(\boldsymbol{\epsilon}^{(i,l)}, \mathbf{x}^{(i)}) \quad \text{and} \quad \boldsymbol{\epsilon}^{(l)} \sim p(\boldsymbol{\epsilon})$$

- The gradient with respect to Φ then can be computed.
- If NN is used, BP is possible

Kingma et al., Auto-Encoding Variational Bayes

More inspiring formulation

$$\log p_{\boldsymbol{\theta}}(\mathbf{x}^{(i)}) = D_{KL}(q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}^{(i)})||p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x}^{(i)})) + \mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}^{(i)})$$

$$\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}^{(i)}) = \mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})}\left[-\log q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}) + \log p_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{z})\right]$$

$$\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}^{(i)}) = \boxed{-D_{KL}(q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}^{(i)})||p_{\boldsymbol{\theta}}(\mathbf{z}))} + \mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}^{(i)})}\left[\log p_{\boldsymbol{\theta}}(\mathbf{x}^{(i)}|\mathbf{z})\right]$$

$$\widetilde{\mathcal{L}}^{B}(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}^{(i)}) = -D_{KL}(q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}^{(i)})||p_{\boldsymbol{\theta}}(\mathbf{z})) + \frac{1}{L}\sum_{l=1}^{L}(\log p_{\boldsymbol{\theta}}(\mathbf{x}^{(i)}|\mathbf{z}^{(i,l)}))$$

$$\text{where} \quad \mathbf{z}^{(i,l)} = g_{\boldsymbol{\phi}}(\boldsymbol{\epsilon}^{(i,l)}, \mathbf{x}^{(i)}) \quad \text{and} \quad \boldsymbol{\epsilon}^{(l)} \sim p(\boldsymbol{\epsilon})$$

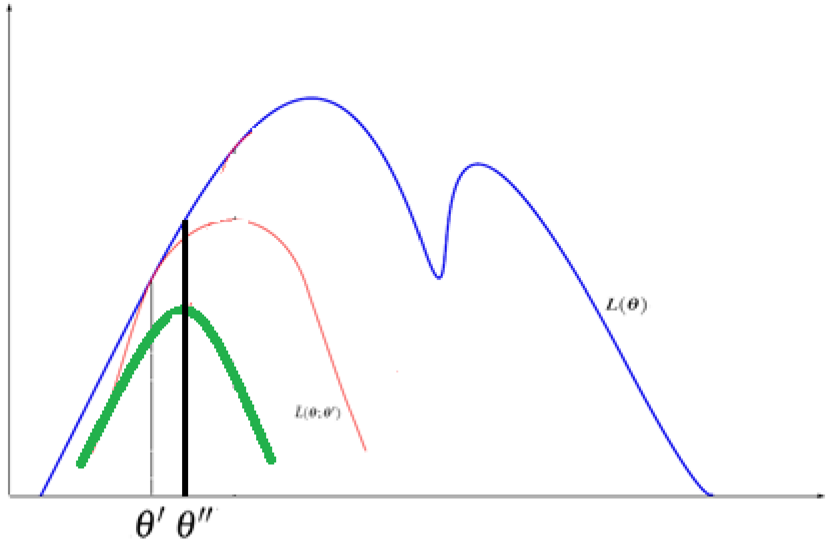
- Reconstruction error constrained by a KL to the prior.
- Will encourage a posterior approach to a wanted form, e.g., Gaussian
- We will back to this point later.

A quick summary

- To have a good representation, we need a complex generation model.
- With the complex generation, both training and inference is complex.
- If posterior is computable, EM can be used to train the model.
- If posterior is hard, we need an estimation.
- VB does not work since the generation model cannot be integrated.
- So we use a parametric model to approximate the posterior, and optimize the parameter using gradient approach with a fixed θ.
- A sampling approach based on re-parameterzation trick is used to compute the gradient.
- Now the posterior is updated with the fixed θ, and so the E step is completed (fortunately).

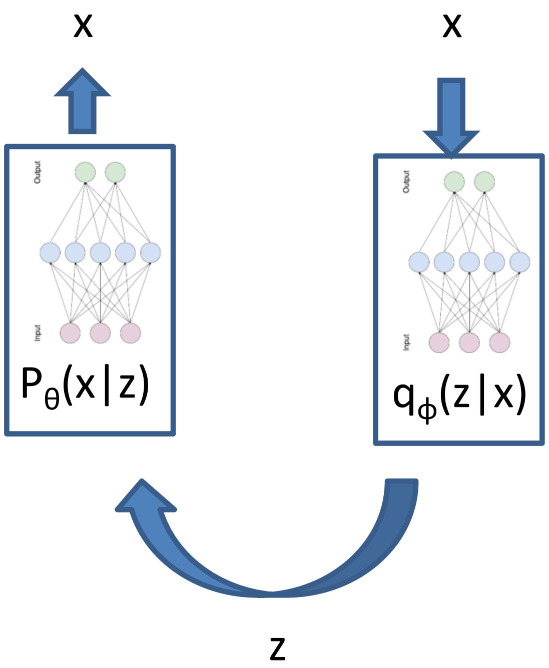
Come back to EM…

- We come back to the M step of the EM, update θ, with Φ fixed (essentially the gradient is computed simultaneously).
- An imperfect posterior this time.



$p(z|x) \approx q_{\Phi}(z|x)$,

The full process is a loop



x

x

$P_\theta(x|z)$

$q_\phi(z|x)$

z

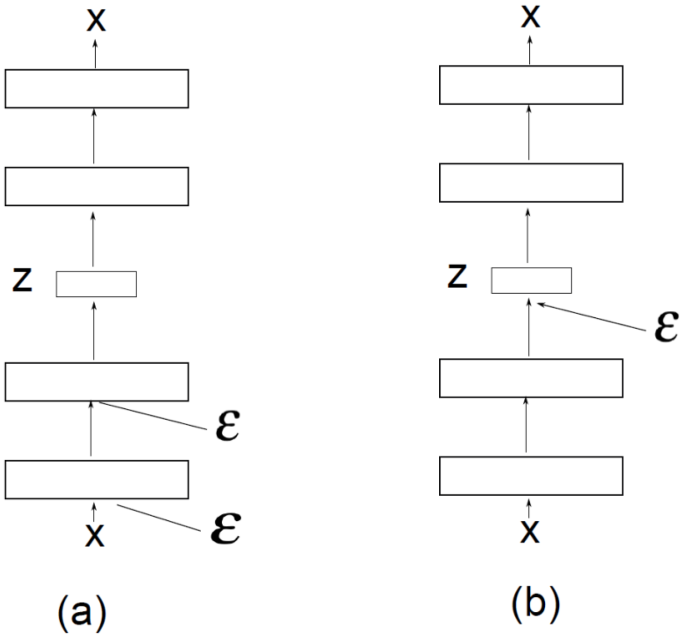
$$\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}^{(i)}) = \mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})}\left[-\log q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}) + \log p_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{z})\right]$$

1. A single objective function KL(q||p), targeting for a good $p_\theta(x)$
2. E and M are different part of the objective

Some questions

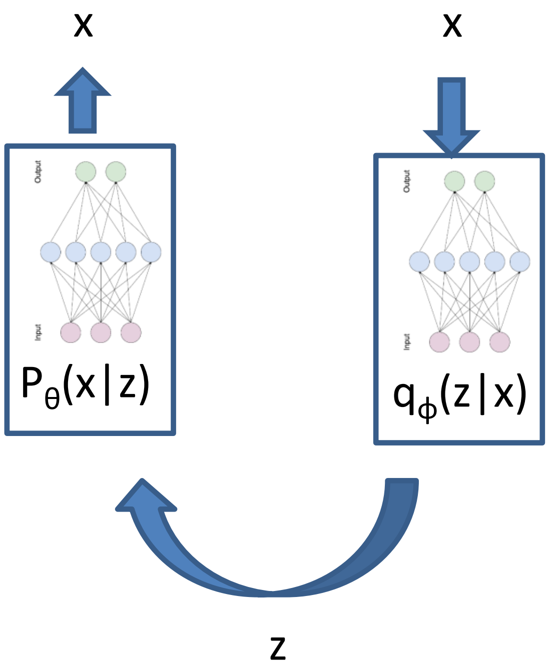
- Will the found p_θ an optimal generation model?
  - NO. The bound is not tangent to the true objective.
- Will $q_\Phi$be optimal with the selected p_θ?
  - Yes. No way to improve the bound w.r.t $q_{\Phi,}$ meaning that noway to reduce the KL to p_θ.

It is AE with randomness in code



1. More complex distribution x’|x=g(x)+ε v.s. x’|x=g(x,ε)
2. Code constrained by prior, simple representation
3. Easy to train

A deep thinking



x

x

$P_\theta(x|z)$

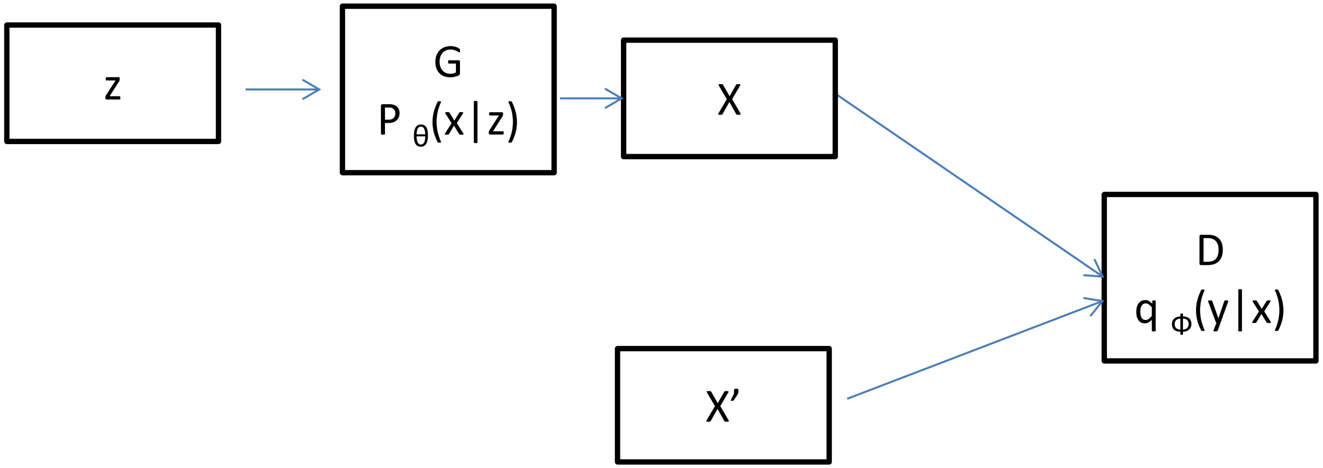
$q_\phi(z|x)$

z

- Essentially, it generates samples from a reverse process (posterior) to match the forward process.
- The forward process is from latent to visible.
- Essentially make the two process consistent, as they should.

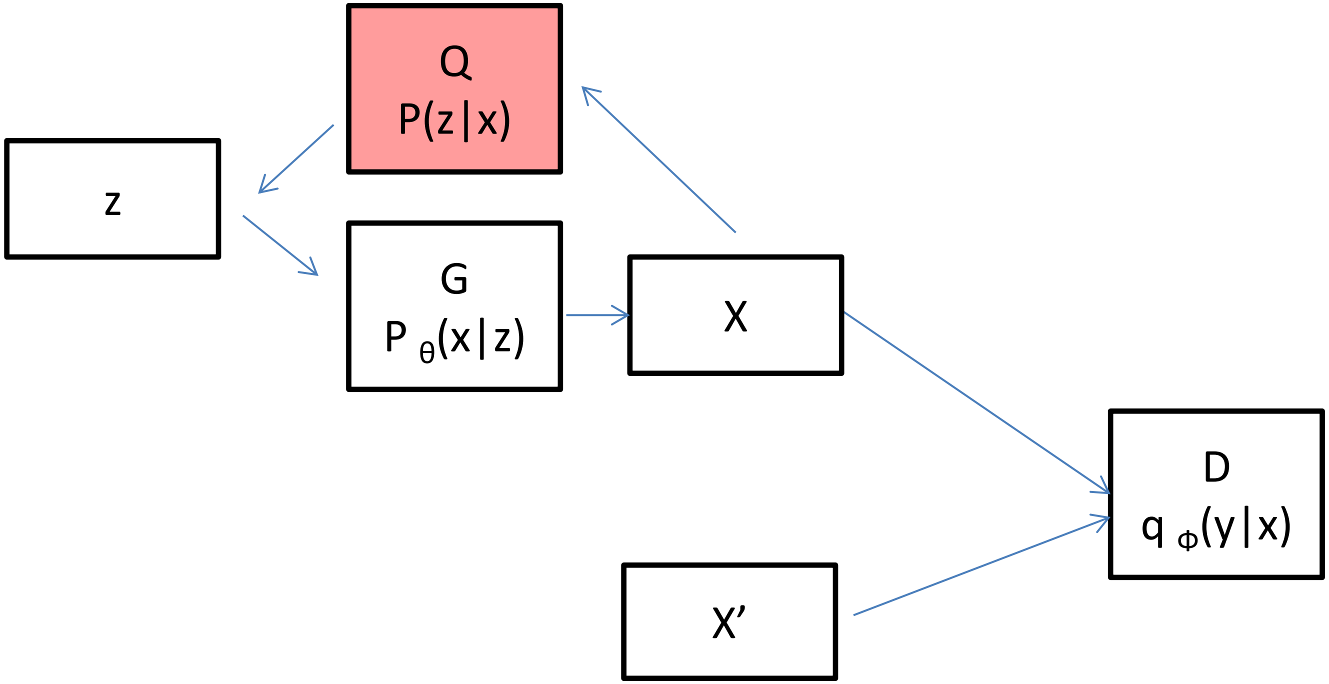
Now look at GAN

- $L(\theta, \Phi)=E_{p\ (x;\ \theta)}\log\boxed{(q^{r}(}y|x; \Phi))$
- A reverse KL: generation process as the reverse process.
- No regularization term.
- Objective does not match the generation process (so adversarial).
- Produce sharp generation compared with VAE, due to the asymmetry of KL.



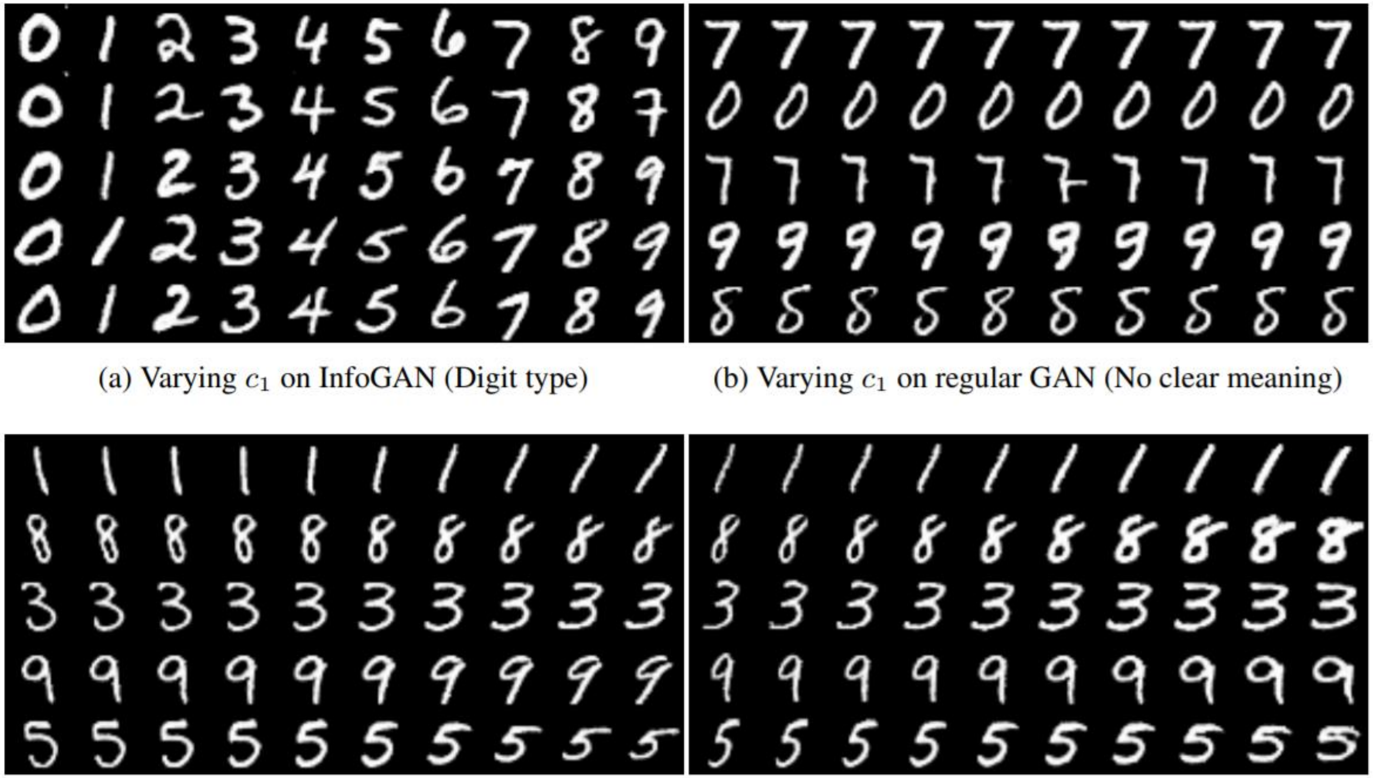
InfoGan: GAN with x-z pair

$$\min_{G,Q}\max_{D} V_{\mathrm{InfoGAN}}(D,G,Q) = V(D,G) - \lambda L_I(G,Q)$$



Chen et al. InfoGAN: Interpretable Representation Learning by Information Maximizing Generative Adversarial Nets

InfoGan



(a) Varying c_1 on InfoGAN (Digit type)

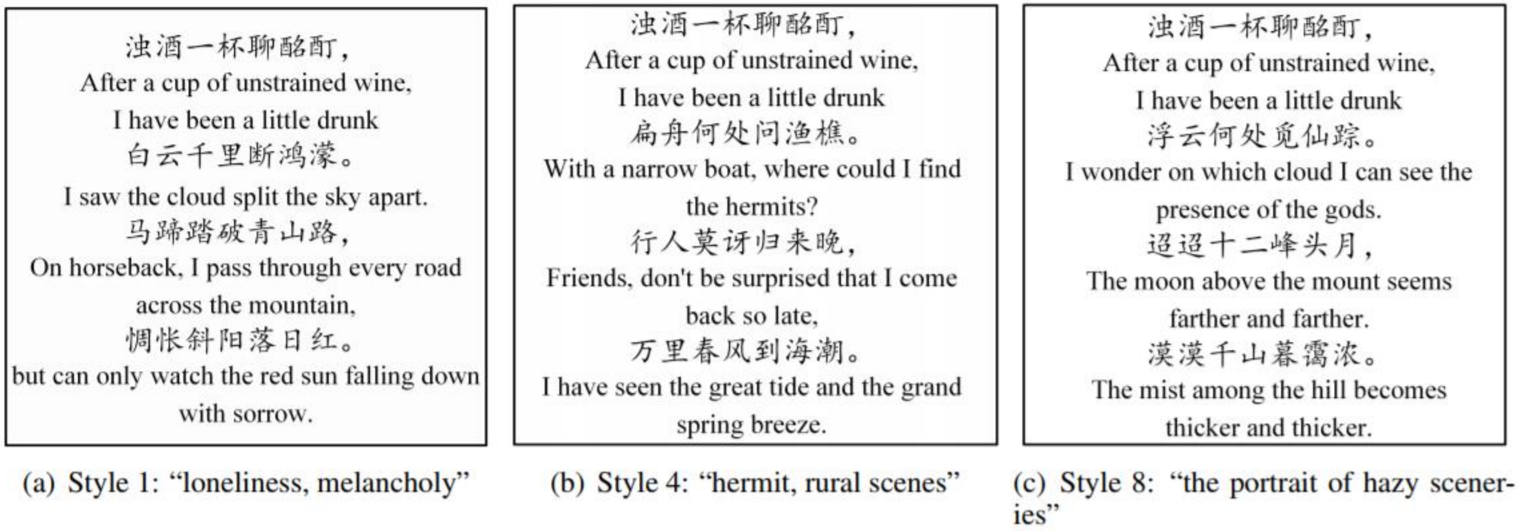
(b) Varying c_1 on regular GAN (No clear meaning)

(c) Varying c_2 from -2 to 2 on InfoGAN (Rotation)

(d) Varying c_3 from -2 to 2 on InfoGAN (Width)

Chen et al. InfoGAN: Interpretable Representation Learning by Information Maximizing Generative Adversarial Nets

Poem generation by InfoGan



浊酒一杯聊酩酊，
After a cup of unstrained wine,
I have been a little drunk
白云千里断鸿濛。
I saw the cloud split the sky apart.
马蹄踏破青山路，
On horseback, I pass through every road
across the mountain,
惆怅斜阳落日红。
but can only watch the red sun falling down
with sorrow.

(a) Style 1: “loneliness, melancholy”

浊酒一杯聊酩酊，
After a cup of unstrained wine,
I have been a little drunk
扁舟何处问渔樵。
With a narrow boat, where could I find
the hermits?
行人莫讶归来晚，
Friends, don't be surprised that I come
back so late,
万里春风到海潮。
I have seen the great tide and the grand
spring breeze.

(b) Style 4: “hermit, rural scenes”

浊酒一杯聊酩酊，
After a cup of unstrained wine,
I have been a little drunk
浮云何处觅仙踪。
I wonder on which cloud I can see the
presence of the gods.
迢迢十二峰头月，
The moon above the mount seems
farther and farther.
漠漠千山暮霭浓。
The mist among the hill becomes
thicker and thicker.

(c) Style 8: “the portrait of hazy sceneries”

- Yang et al., Stylistic Chinese Poetry Generation via Unsupervised Style Disentanglement

Cycle GAN

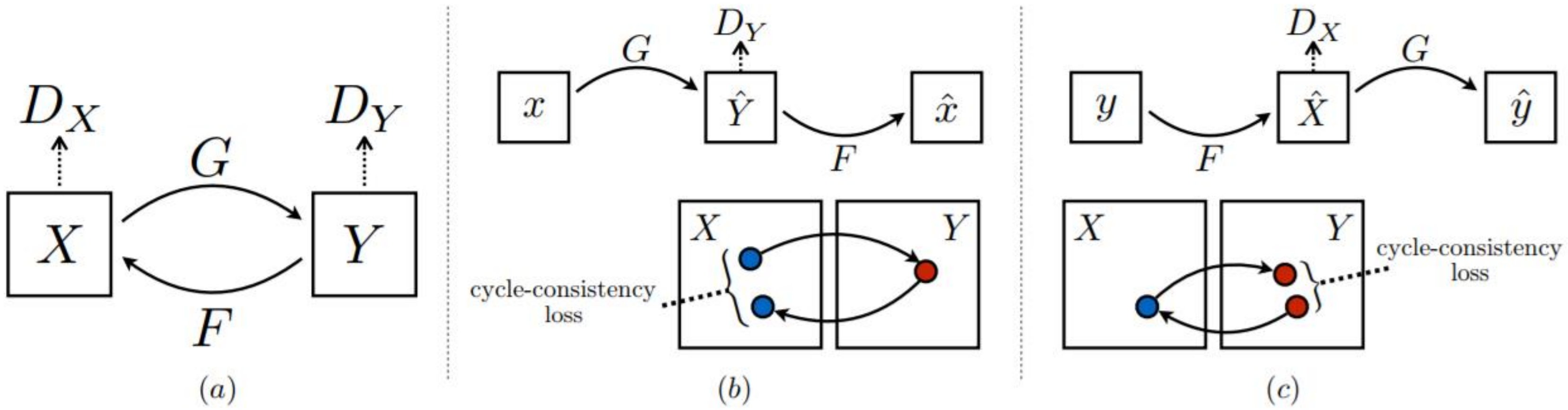
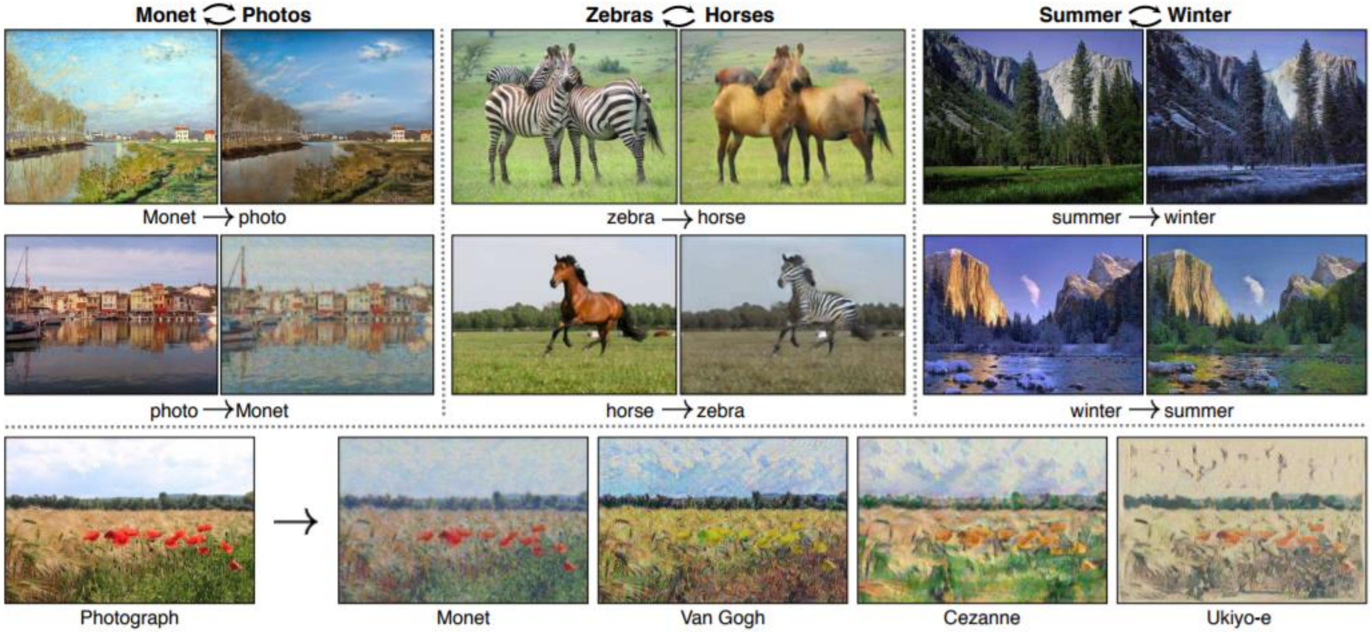


Figure 3: (a) Our model contains two mapping functions $G : X \rightarrow Y$ and $F : Y \rightarrow X$, and associated adversarial discriminators D_Y and D_X. D_Y encourages G to translate X into outputs indistinguishable from domain Y, and vice versa for D_X, F, and X. To further regularize the mappings, we introduce two "cycle consistency losses" that capture the intuition that if we translate from one domain to the other and back again we should arrive where we started: (b) forward cycle-consistency loss: $x \rightarrow G(x) \rightarrow F(G(x)) \approx x$, and (c) backward cycle-consistency loss: $y \rightarrow F(y) \rightarrow G(F(y)) \approx y$

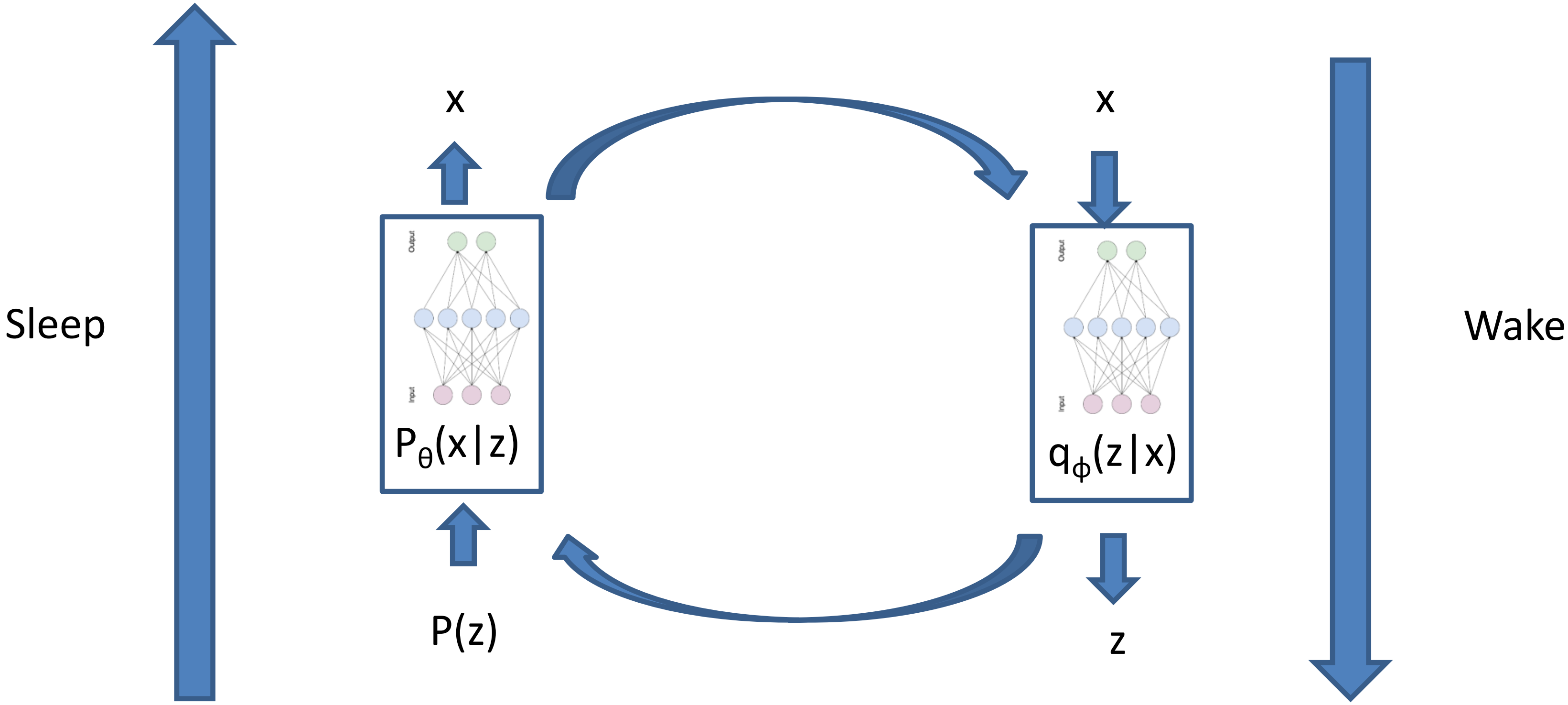
Zhu et al., Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks

Cycle GAN



Zhu et al., Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks

Wake-sleep: A generic framework



- Using a reverse process to generate samples that will match the forward process
- Inference and generation are essentially paired and coupled
- But can be optimized in either side of KL, leading to different properties.

GE Hinton, P Dayan, BJ Frey, RM Neal, The "wake-sleep" algorithm for unsupervised neural networks, Science 1995.

We now understand…

- Complex inference is possible, with the help of deep learning.
- The complex inference should be coupled with generation, thanks to the Bayesian rule.
- Re-parameterization and KL regularization help us infer simple representations.

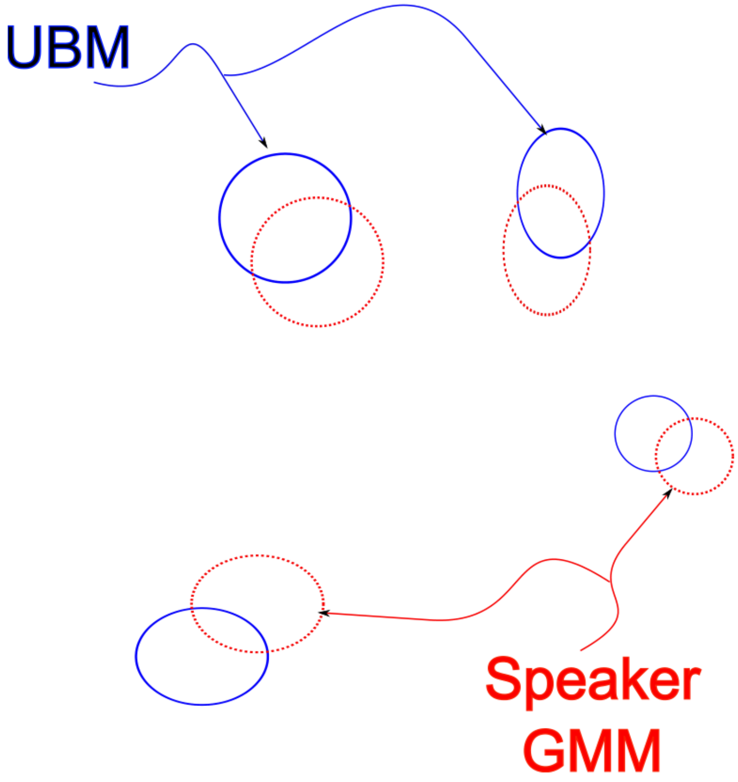
Things that are under going

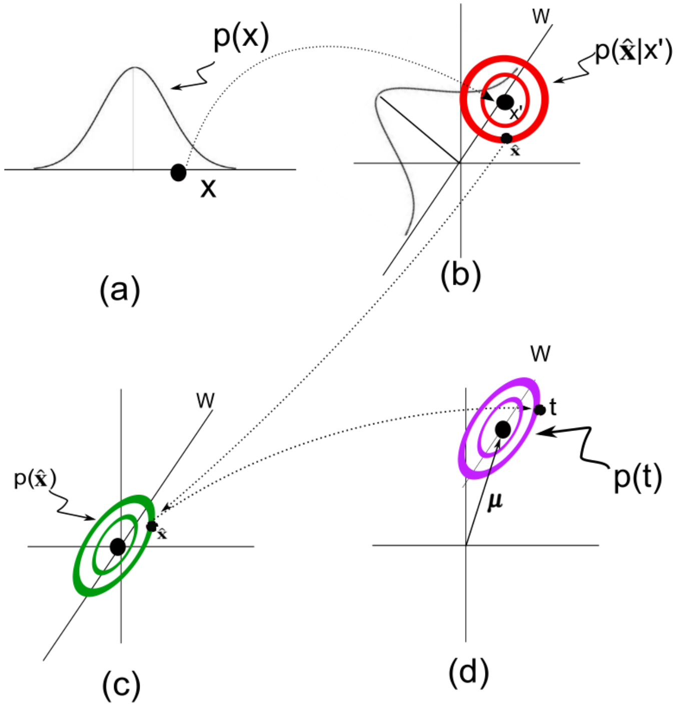
- Flexible optimization variable
  - The generator can be either the probability or the target of the expectation
  - The optimization can be on either the probability or the function of the target
  - Formally can be in any part of the KL.
- Not coupled pair
  - The p and q may be not so tightly coupled
  - Mostly not KL
- Multiple couples
  - Info GAN

Content

- Theory introduction
- Application to speaker recognition

Bayesian approach

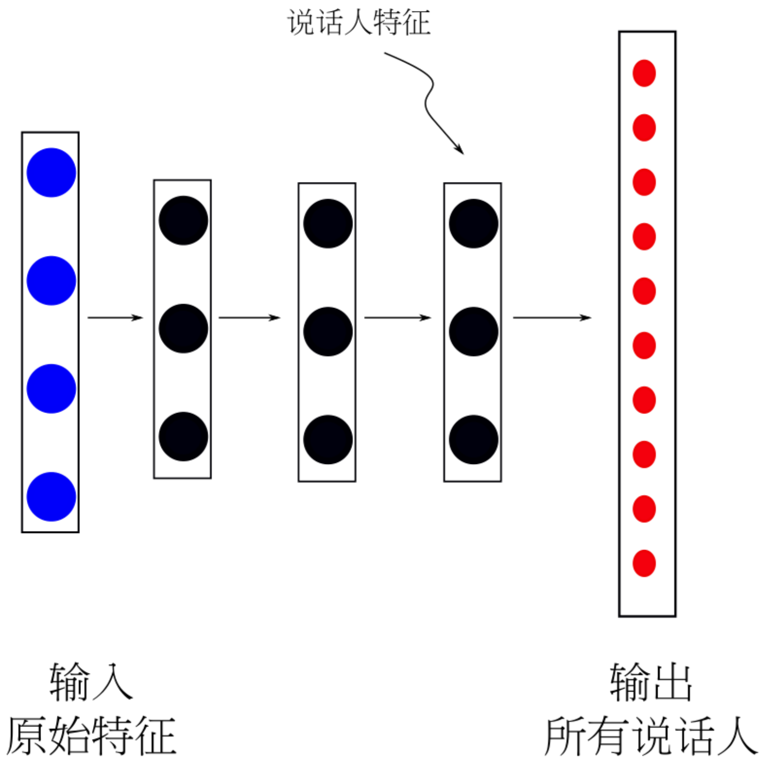




Basically generative and shallow model, but with mixture. It is a cuo-qiong-fat model.

Neural model

- Discriminative, deep feature learning.
- Gao-fu-Thin model



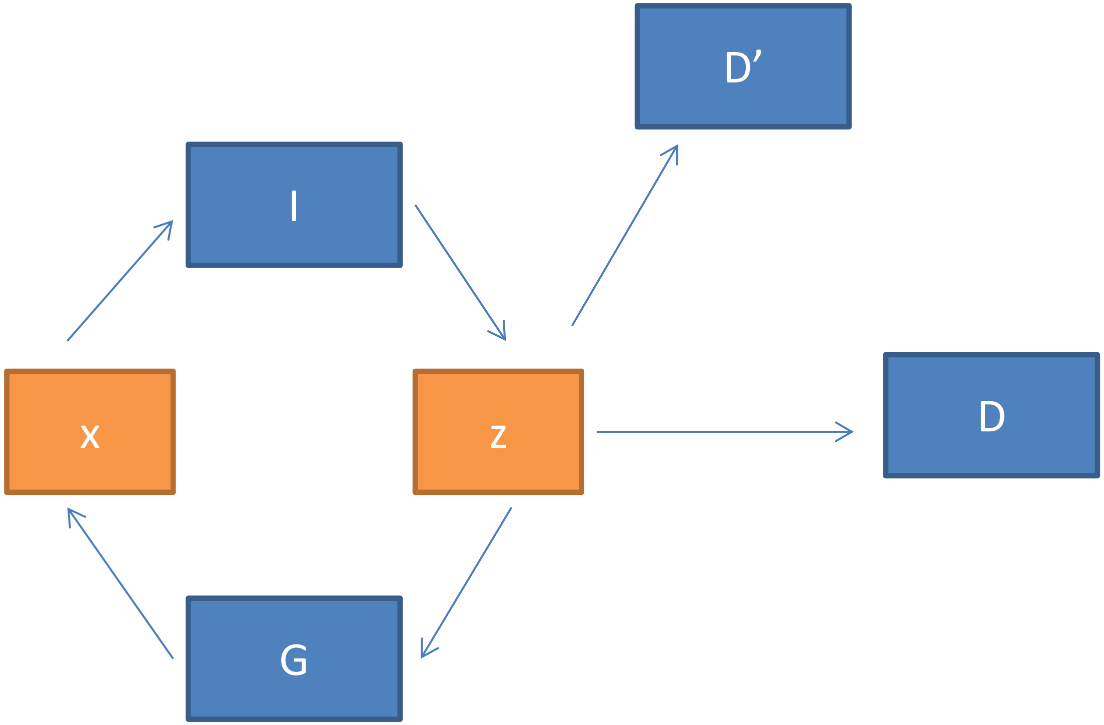
Relative problems

- Bayesian model
  - Description power limited by structure
  - Fragile in domain change
- Neural model
  - Unconstrained distribution (risky and limited)
  - Long-term dependency (do we need?)
    - A problem of text dependent

Possible solution

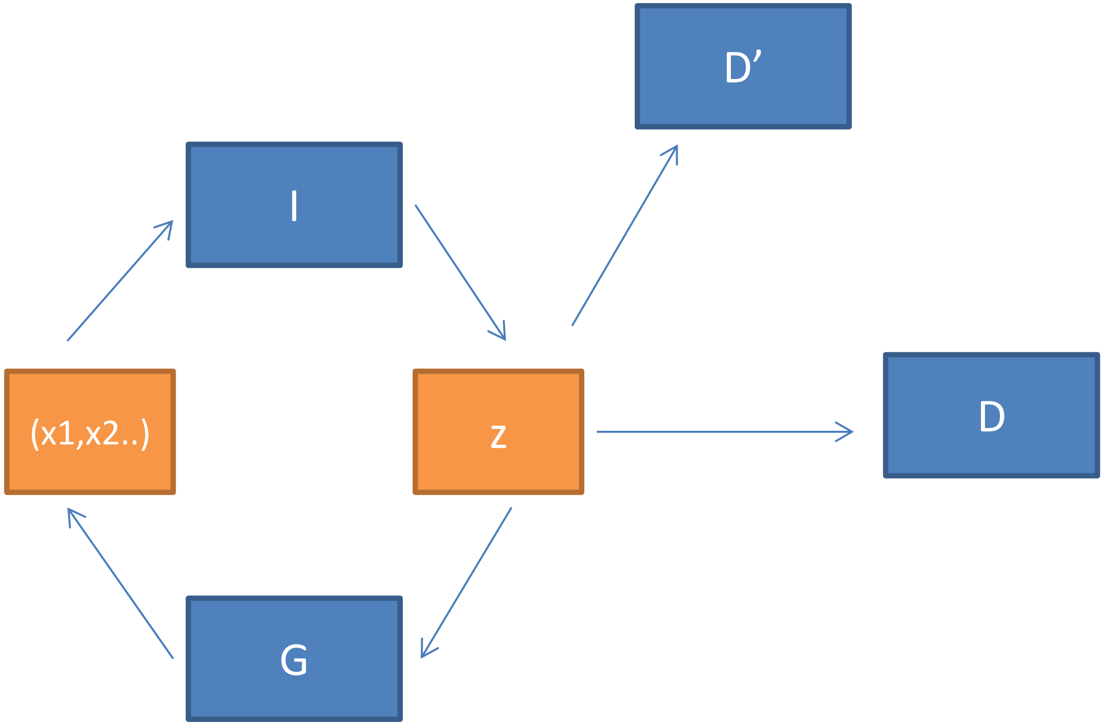
- Neural net + description
  - Multitask training
- Neural net + constrained distribution
  - Constrained training (LLT 18)
- We hope to put them in a single objective, as infoGan.

V-feature Architecture



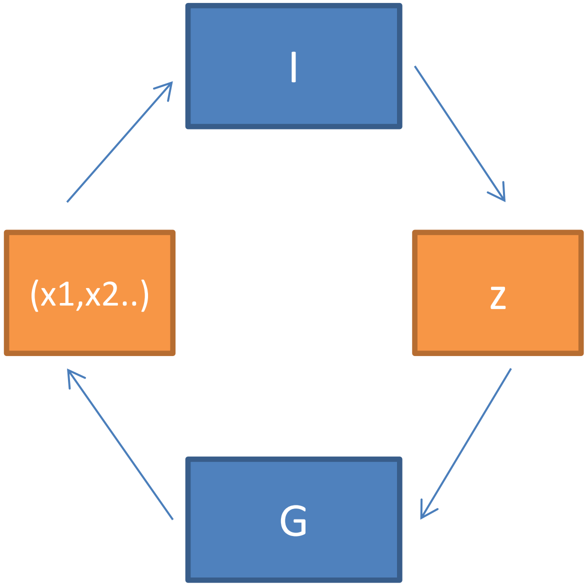
1: Using VAE (I,G) objective to form deep descriptive representations
2. Using global discriminative target (D) to improve discriminative power
3. Using paired discriminative target (D’) to reduce within-speaker variation
4. Using phonetic information as auxiliary information (input? Randomness choice?)

Utterance/speaker level v vector

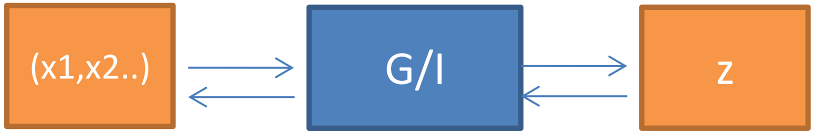


1. Derive z from feature sequence {x_i}; z then sample to produce {x_i}.
2. Similar to i-vector, however its never linear Gaussian, but complex distribution; interestingly, z is inheriently Gaussian.

Compare to i-vector



Deep, non-linear net



Linear, shallow net

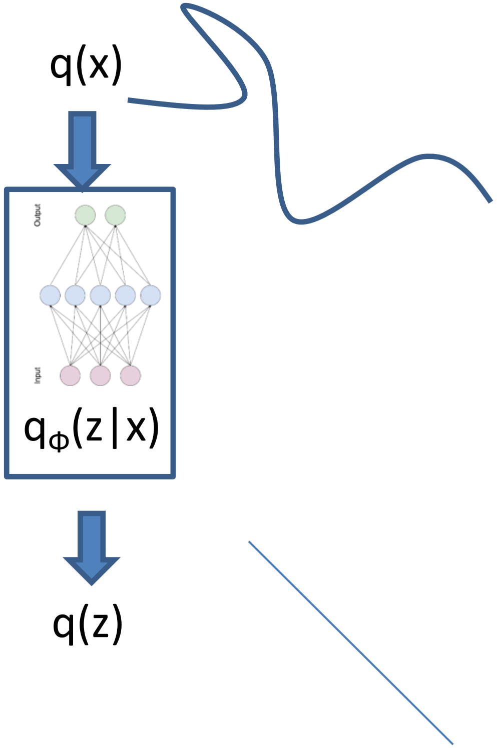
A way to transfer i-vector to deep i-vectors.

Something concerned

- Disentanglement of factors in z: shall they independent? Shall they be grouped? How to regularize?
- Discriminative power involved in z, by introducing pair-wised regularization: How about a smooth hinge loss with a pre-defined threshold?
- Is this a way for other speech-related tasks? Can they be trained jointly? A better way for speech factorization? Shall be multiple ε in the factorization?

Conclusions

- It’s likely that we are approaching a reasonable path towards using both neural and Bayesian methods. That helps us to infer simple code by designing complex generation model.
- Fortunately, a practical training/inference approach is ready, by something like wake-sleep. This is analog to the fact that you need listen before speaking.
- We have a good reason to move towards this direction, at least keep this in mind when facing a new task.



q(x)

$q_\Phi(z|x)$

q(z)