[Paper review 45]

Advances in Variational Inference

(Zhang, et.al , 2018)

[Contents]

1. Abstract

VI lets us approximate high-dim Bayesian posterior **with a simpler variational distn**

Start with standard MFVI,

then review recent advances :

- (1) scalable VI (including stochastic approx)
- (2) generic VI (extends the applicability of VI to a large class of models, i.e. non-conjugate models)

- (3) accurate VI (includes variational models beyond mean-field approx, or with atypical divergences)
- (4) amortized VI (implements inference over latent variables with inference networks)

2. Introduction

Variational Inference

- **"optimization based approach"**
- faster, but may suffer from oversimplified posterior approximation

In recent years, new interest in variational methods!

- (1) availability of large datasets $\rightarrow$ **scalable approaches**
- (2) classical VI is limited to conditionally conjugate exp fam model $\rightarrow$ **BBVI**
- (3) more accurate variational approx, such as **alternative divergence measures**
- (4) amortized inference employs **complex function** such as NN

  (ex. Bayesian deep learning architecture, such as VAE)

Summary : recent developments in **scalable, generic, accurate, amortized** VI

3. Variational Inference

(너무 많이 봐서 간단히 정리하고만 넘어감)

notation

- observation : $\boldsymbol{x}=\{x_1,x_2,\cdots,x_M\}$
- latent variables : $z=\{z_1,z_2,\cdots,z_N\}$
- variational params : $\boldsymbol{\lambda}=\{\lambda_1,\lambda_2,\cdots,\lambda_N\}$

  (variational distn : $q(z;\boldsymbol{\lambda})$$)

Variational Objective (ELBO)

$$\begin{aligned}\log p(\boldsymbol{x}) &= \log\int p(\boldsymbol{x},\boldsymbol{z})d\boldsymbol{z} = \log\int \frac{p(\boldsymbol{x},\boldsymbol{z})q(\boldsymbol{z};\boldsymbol{\lambda})}{q(\boldsymbol{z};\boldsymbol{\lambda})}d\boldsymbol{z} \\ &= \log\mathbb{E}_{q(\boldsymbol{z};\boldsymbol{\lambda})}\left[\frac{p(\boldsymbol{x},\boldsymbol{z})}{q(\boldsymbol{z};\boldsymbol{\lambda})}\right] \\ &\geq \mathbb{E}_{q(z;\boldsymbol{\lambda})}\left[\log\frac{p(\boldsymbol{x},\boldsymbol{z})}{q(\boldsymbol{z};\boldsymbol{\lambda})}\right]\equiv\mathscr{L}(\boldsymbol{\lambda})\end{aligned}$$

We can rewrite true log marginal probability of the data as sum of

- (1) ELBO
- (2) KL-div

$\log p(\boldsymbol{x}) = \mathscr{L}(\boldsymbol{\lambda}) + \mathrm{D}_{\mathrm{KL}}(q \| p).$

MFVI

Assumption : $q(z; \boldsymbol{\lambda}) = \prod_{i=1}^{N} q(z_i; \lambda_i)$

If we rewrite ELBO....

$$\begin{aligned} \mathscr{L} = & \int q(z_j) \mathbb{E}_{q(\boldsymbol{z}_{\neg j})} [\log p(z_j, \boldsymbol{x} \mid z_{\neg j})] \, dz_j \\ & - \int q(z_j) \log q(z_j) \, dz_j + c_j \end{aligned}.$$

Solution (optimize by minimizing **negative ELBO**)

$\log q^*(z_j) = \mathbb{E}_{q(z_{\neg j})} [\log p(z_j \mid z_{\neg j}, \boldsymbol{x})] + \text{ const.}$

$$\begin{aligned} q^*(z_j) & \propto \exp\left(\mathbb{E}_{q(\boldsymbol{z}_{\neg j})} [\log p(z_j \mid \boldsymbol{z}_{\neg j}, \boldsymbol{x})]\right) \\ & \propto \exp\left(\mathbb{E}_{q(\boldsymbol{z}_{\neg j})} [\log p(\boldsymbol{z}, \boldsymbol{x})]\right) \end{aligned}.$$

Beyond Vanilla VI

section 3) scale VI to large dataset

section 4) make VI both easier to use & more generic

section 5) non-MFVI

section 6) NN can be used to amortize the estimation of certain local latent variables
$\rightarrow$ bridges the gap between "Bayesian inference" & "modern representation learning"

4. Scalable VI

SVI (Stochastic Variational Inference)

- **"use SGD"** to scale VI to large dataset

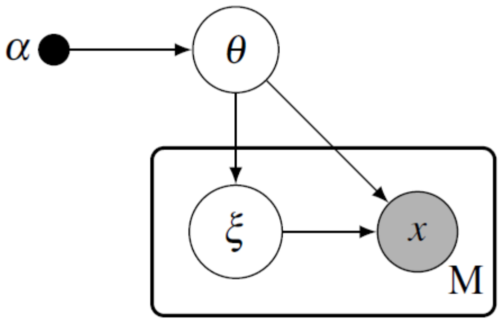

Fig. 1. A graphical model of the observations $\boldsymbol{x}$ that depend on underlying local hidden factors $\boldsymbol{\xi}$ and global parameters θ. We use $\boldsymbol{z} = \{\theta, \boldsymbol{\xi}\}$ to represent all latent variables. M is the number of the data points. N is the number of the latent variables.

notation

- latent variable : $z = \{\theta, \xi\}$
  - local : $\xi = \{\xi_1, \ldots, \xi_M\}$
  - global : θ
- variational parameter : $\boldsymbol{\lambda} = \{\gamma, \boldsymbol{\phi}\}$
  - $\boldsymbol{\phi}$: corresponds to "local" latent variable
  - γ : corresponds to "global" latent variable
- hyperparameters : α
- mini-batch size : S

4-1. SVI (Stochastic Variational Inference)

Variational distn :

$q(\boldsymbol{\xi}, \boldsymbol{\theta}) = q(\boldsymbol{\theta} \mid \gamma) \prod_{i=1}^{M} q(\xi_i \mid \phi_i)$

ELBO :

$$\mathscr{L} = \mathbb{E}_q[\log p(\theta \mid \alpha) - \log q(\theta \mid \gamma)] + \sum_{i=1}^{M} \mathbb{E}_q\left[\log p(\xi_i \mid \theta) + \log p(x_i \mid \xi_i, \theta) - \log q(\xi_i \mid \phi_i)\right].$$

ELBO above can be optimized by CAVI, GD...

$\rightarrow$ but both CAVI & GD is not scalable

(every iteration / gradient step scales with M, therefore expensive for large dataset)

THUS, use SVI (STOCHASTIC VARIATIONAL INFERENCE)

SVI (Stochastic Variational Inference)

- every iteration, select mini-batchs of size S to obtain stochastic estimate of ELBO

- **stochastic estimate of ELBO** :

  $\hat{\mathscr{L}} = \mathbb{E}_q[\log p(\theta \mid \alpha) - \log q(\theta \mid \gamma)] +$

  $\frac{M}{S} \sum_{s=1}^{S} \mathbb{E}_q \left[\log p\left(\xi_{i_s} \mid \theta\right) + \log p\left(x_{i_s} \mid \xi_{i_s}, \theta\right) - \log q\left(\xi_{i_s} \mid \phi_{i_s}\right)\right]$

  - there is a stochastic part in the **second term**
  - this is a noisy estimator of the direction of steepest ascent of the true ELBO
- using natural gradients (instead of standard gradients) in SVI simplifies the variational updates for models in the conditionally conjugate exponential family!
- when $S = M$: same as traditional batch VI

  (computational savings when $S << M$)
- learning rate ρ_t : should decrease with iteration

  (Robbins-Monro conditions : $\sum_{t=1}^{\infty} \rho_t = \infty$ and $\sum_{t=1}^{\infty} \rho_t^2 < \infty$)

SVI is referred to as **ONLINE VI**

- SVI = Online VI, when the **volume of data M is known!**
- in streaming applications, the mini-batches arrive sequentially from a data source,

  but the SVI updatse are the same

  (However, when M is unknown, it is unclear how to set the scale param M/S

4-2. Tricks of the Trade for SVI

convergence speed of SGD depends on **"variance of the gradient estimates"**

(a) Adaptive Learning Rate & Mini-batch size

(due to LLN) mini-batch size $\uparrow \rightarrow$ stochastic gradient noise $\downarrow$ (allowing larger learning rates)

[method 1] learning rate adaptation

- empirical gradient variance can guide the adaptation of the learning rate

  (inversely proportional to gradient noise)
- γ : global variational param

  γ^* : optimal global variational param

  Σ : covariance matrix of the variational parameter in this mini-batch

  optimal learning rate : $\rho_t^* = \frac{(\gamma_t^* - \gamma_t)^T (\gamma_t^* - \gamma_t)}{(\gamma_t^* - \gamma_t)^T (\gamma_t^* - \gamma_t) + \operatorname{tr}(\Sigma)}$.

[method 2] mini-batch size adaptation

- keep learning rate fixed

(b) Variance Reduction

[method 1] Control Variates

- same expectation, lower variance
- used commonly in MC simulation & stochastic optimization
- **SVRG (Stochastic Variance Reduced Gradient)**
  - construct control variate (take advantage of previous gradients from all data point),
  - standard) $\gamma_{t+1} = \gamma_t - \rho_t \left(\nabla \mathscr{L}\left(\gamma_t\right)\right)$

    SVRG) $\gamma_{t+1} = \gamma_t - \rho_t \left(\nabla \hat{\mathscr{L}}\left(\gamma_t\right) - \nabla \hat{\mathscr{L}}(\tilde{\gamma}) + \tilde{\mu}\right)$
  - $\hat{\mathscr{L}}$: estimated objective (= negative ELBO),based on current mini-batch

    $\tilde{\gamma}$: snapshot of γ after every m iterations

    $\tilde{\mu}$: batch gradient computed over all the datapoints ($\tilde{\mu} = \nabla \mathscr{L}(\tilde{\gamma})$)
  - $E[-\nabla \mathscr{L}(\tilde{\gamma}) + \tilde{\mu}] = 0$ (thus can be used as control variates!)
  - convergence rate :

    standard) $\mathscr{O}(1/\sqrt{T})$

    SVRG) $\mathscr{O}(1/T)$

[method 2] Non-uniform Sampling

Instead of subsampling with **equal** prob, use **non-uniform** sampling when selecting mini-batches

(for lower variance!)

but not always practical

[method 3] Other methods

- Rao Blackwellization
- average expected sufficient statistics over sliding window of mini-batches

4-3. Collapse, Sparse, Distributed VI

Instead of using stochastic optimization for faster convergence,

present methods that **leverage the structure of certain models** for faster convergence

Collapsed VI (CVI)

"Integrate out certain model params"

$\rightarrow$ due to reduced number of params to be estimated, it becomes FASTER & ELBO tighter

(but constrained to conjugate exp fams.... :( ))

CVI for topic models : ex) collapse topic proportions or topic assignments

Computational benefit of CVI depends on the "statistic of collapsed variables"

Collapsing latent variables can make other inference tractable

- ex) topic models : collapse discrete variables

  $\rightarrow$ only infer the continuous ones , thus allowing using **inference network**

Shortcomings

- 1) Mathematical challenges
- 2) Marginalizing variables can introduce **additional dependencies** between variables

Sparse VI

introduce **additional low-rank approx** , enabling scalable inference

can be interpreted as "modeling choice", or "inference scheme"

Often encountered in **GP** literature

- computational cost of GP : $O(M^3)$ where M = number of data

  (by inversion of kernel matrix K_{MM} , which hinders the application of GPS to big data)

Sparse inference in GP

- introduce T **inducing points**

  (= pseudo-inputs that reflect original data)
- since $T << M$, yield more sparse representation
- only $O(MT^2)$ is needed :)
- further) collapse distn of inducing points & extends to a stochastic version $\rightarrow O(T^3)$
- makes Deep GPs tractable!

Parallel and Distributed VI

can also be adjusted to distributed computing

required in large scale scenarios

5. Generic VI : Beyond the Conjugate Exponential Family

Making VI more generic!

- applicable to **broader class** of models

- eliminate the need for **model-specific** calculations

Key : ***"Stochastic gradient estimators"*** of ELBO that can be **computed for a broader class** of model

(1) **Laplace Approximation**

(2) **BBVI** that rely on REINFORCE(or score function gradient)

(3) **BBVI** that uses reparameterization gradients

(4) Other approaches for **non-conjuagte VI**

5-1. Laplace's method & limitations

Laplaces' approximation

- an **alternative to non-conjugate inference**
- approximate the posterior by **Gaussian**
- step 1) seek the MAP (mean of Gaussian)

  step 2) compute the inverse of Hessian (cov of Gaussian)
- needs to be twice-differentiable
- (by Bayesian CLT) posterior approaches Gaussian asymptotically, in the limit of large data

Shortcomings

- 1) being purely local & depend only on the curvature of the posterior around the optimum
- 2) does not apply to discrete variables
- 3) Hessian can be costly in high dimensions

  $\rightarrow$ makes intractable with large number of datasets

5-2. REINFORCE gradients

(classical VI) ELBO is derived analytically

(BBVI) propose a generic inference algorithm

- ONLY the generative process of data has to be specified
- model can be anything!

Main idea of BBVI :

Represent the gradient as an expectation & use MC techniques to estimate this expectation

- can obtain an UNBIASED gradient estimator by SAMPLING from the variational distribution, WITHOUT the need of calculating ELBO anlaytically
- full gradient)

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

- stochastic gradient)

  $\nabla_\lambda \hat{\mathscr{L}}_s = \frac{1}{K} \sum_{k=1}^{K} \nabla_{\boldsymbol{\lambda}} \log q\left(z_k \mid \boldsymbol{\lambda}\right)\left(\log p\left(\boldsymbol{x}, z_k\right) - \log q\left(z_k \mid \boldsymbol{\lambda}\right)\right)$

  where $z_k \sim q(z \mid \lambda)$

  - $\nabla_\lambda \log q\left(z_k \mid \boldsymbol{\lambda}\right)$: called **score function**

    (key part of REINFORCE algorithm)

Variance reduction

- Rao-Blackwellization
- Control variates

Variance Reduction for BBVI

SVI vs BBVI

- SVI) noise resulted from subsampling from a FINITE set of datapont
- BBVI) noise originates from r.v with possibly INFINITE support

  $\rightarrow$ SVRG is not applicable

  (full gradient is not a sum over FINITELY many terms)

  thus, BBVI invloves **different set of control variates**

Score Function control variate

- most important control variate in BBVI
- subtract MC expectation of the score function from the gradient estimator

  $\nabla_{\boldsymbol{\lambda}} \mathscr{L}_{\text{control}} = \nabla_{\boldsymbol{\lambda}} \hat{\mathscr{L}} - \frac{w}{K} \sum_{k=1}^{K} \nabla_{\boldsymbol{\lambda}} \log q\left(z_k \mid \boldsymbol{\lambda}\right).$

  - $\nabla_{\boldsymbol{\lambda}} \log q(z_k \mid \boldsymbol{\lambda})$: expectation is zero (under variational distn)
  - $\frac{w}{K} \sum_{k=1}^{K}$: w is selected s.t. it minimizes the variance of the gradient

Original BBVI paper introduces both

- 1) Rao-Blackwellization
- 2) control-variates

$\rightarrow$ good choice depends on the model!

Different approach ex)

- **overdispersed importance sampling**

  from proposal distn that place high mass on the tails $\rightarrow$ variance of gradient is reduced!

5-3. Reparameterization Gradient VI

Reparameterization Gradients

Reparam trick

- by MC samples!
- gives low-variance stochastic gradients

  & do not need to compute analytic expectation
- distn $q(z;\lambda)$ can be expressed as a transformation of r.v $\epsilon \sim r(\epsilon)$
- ex) $z \sim \mathscr{N}\left(z; \mu, \sigma^2\right)$, $z = \mu + \sigma\varepsilon$, where $\varepsilon \sim \mathscr{N}(\varepsilon; 0, 1)$

Allows to compute any expectation over z as an expectation over ε

build a **STOCHASTIC GRADIENT ESTIMATOR** of the ELBO!

$$\nabla_{\boldsymbol{\lambda}} \hat{\mathscr{L}}_{rep} = \frac{1}{K}\sum_{k=1}^{K} \nabla_{\boldsymbol{\lambda}} \left(\log p\left(x_i, g\left(\boldsymbol{\varepsilon}_k, \boldsymbol{\lambda}\right)\right) - \log q\left(g\left(\varepsilon_k, \boldsymbol{\lambda}\right) \mid \boldsymbol{\lambda}\right)\right), \varepsilon_k \sim r(\varepsilon) .$$

Variance of this estimator is often lower than that of score function!

Etc

- reparam gradients are also key to VAE
- (discrete distn version) **Gumbel-Max trick**
  - replace argmax operation with a softmax operator
  - temperature parameter controls the degree to which the softmax can approx the categorical distn

5-4. Other Generalizations

Approaches that consider VI in non-conjugate models, but do not follow BBVI principle

Examples

- Taylor approximations
- lower-bounding the ELBO
- using some form of MC estimators....

Approximations based on...

- inner optimization routines : prohibitively slow
- additional lower bounds with closed form updates : computationally efficient

6. Accurate VI : Beyond KL and MFVI

(until now, have dealt with MFVI & KL-div as a measure of distance)

Recent developments go beyond this!

- goal of avoiding poor local optima
- increase the accuracy of VI!

ex) Normalizing Flow, Inference Networks (will be dealt in next section)

(1) origins of MFVI & limitations (skip)

(2) Alternative Divergence measures

(3) Structured VI

6-1. Origins and Limitations of MFVI

- skip

6-2. VI with Alternative Divergences

KL-divergence

- computationally convenient method to measure the distancce
- analytically tractable expectations for certain models!
- problems)
  - underestimating posterior variances
  - unable to break symmetry when multiple modes are close

$\rightarrow$ other divergence measures?

(ex. EP (Expectation Propagation) : use alternative divergence measures)

introduce relevant divergence measure & show how to use in VI

- KL divergence $\subset \alpha$ divergence $\subset f$ divergence
- they all can be written in the form of **Stein discrepancey**

α divergence

both KL divergence & Hellinger distance is a special case of α divergence

(Renyi's formulation)

$D_{\alpha}^{R}(p \| q)=\frac{1}{\alpha-1} \log \int p(x)^{\alpha} q(x)^{1-\alpha} d x$, where $\alpha>0, \alpha \neq 1.$

- For $\alpha \rightarrow 1$, same as standard VI (involving the KL divergence)
- implies a bound on the marginal likelihood

$$\mathcal{L}_\alpha = \log p(\boldsymbol{x}) - D_\alpha^R(q(\boldsymbol{z}) \| p(\boldsymbol{z} \mid \boldsymbol{x})) \\ = \frac{1}{\alpha - 1} \log \mathbb{E}_q \left[\left(\frac{p(\boldsymbol{z}, \boldsymbol{x})}{q(\boldsymbol{z})} \right)^{1-\alpha} \right].$$

- negative value of α = UPPER bound

  (it is not a divergence in this case)

f- Divergence & Generalized VI

α divergence is a subset of f divergence

$D_f(p \| q) = \int q(x) f\left(\frac{p(x)}{q(x)}\right) dx$

Stein Discrepancy and VI

introduce **(1) Stein Discrepancy** & **(2) two VI methods** that use this :

- (a) Stein Variational Gradient Descent (SVGD)
- (b) operator VI

  both share the same objective
  but differ in optimization method

(1) Stein Discrepancy

$D_{\text{stein}}(p, q) = \sup_{f \in \mathscr{F}} \left| \mathbb{E}_{q(z)}[f(z)] - \mathbb{E}_{p(z \mid x)}[f(z)] \right|^2$

- where $\mathscr{F}$ indicates a set of smooth, real-valued functions
- second term $\mathbb{E}_{p(z \mid x)}[f(z)]$: intractable $\rightarrow$ can be only used in VI if this is zero(0) for arbitrary ϕ

$f(z) = \mathscr{A}_p \phi(z)$. where $z \sim p(z)$

- operator $\mathscr{A}$, which makes second term ($\mathbb{E}_{p(z \mid x)}[f(z)]$) zero!

  That is, $\mathscr{A}_p \phi(\boldsymbol{z}) = \phi(\boldsymbol{z}) \nabla_{\boldsymbol{z}} \log p(\boldsymbol{z}, \boldsymbol{x}) + \nabla_{\boldsymbol{z}} \phi(\boldsymbol{z})$

(2) SVGD & Operator VI

both SVGD and Operator VI share the same objective above!

Difference : optimization of the variational objective

- SVGD) kernelized Stein discrepancy
- Operator VI) minmax (GAN-style) formulation

6-3. Structured VI

MFVI assumes fully-factorized
$\rightarrow$ limited accuracy (especially when latent variables are hihgly CORRELATED)

Structured VI

- not fully factorized
- contain dependencies between latent variables
- more expressive
- higher computational cost

  makes harder to estimate the gradient of ELBO

Allowing structured variational distribution is a modeling choice! depends on model

- ex) Structured VI for LDA : maintaining a global structure is vital
- ex) Structured VI for Beta Bernoulli Process : maintaining a local structure is vital

ex) Hierarchical VI & copula VI

(a) Hierarchical VI

HVM(Hierarchical variational models)

- BBVI framework for Structured variational distributions, which applies to broad class of models
- step 1) start with a mean-field variational distribution, $\prod_i q\left(z_i ; \lambda_i\right)$

  step 2) instead of estimating variational param λ,

  place a prior $q(\boldsymbol{\lambda}; \boldsymbol{\theta})$ & marginalize them out!

  $q(z; \boldsymbol{\theta})=\int\left(\prod_i q\left(z_i ; \lambda_i\right)\right) q(\boldsymbol{\lambda} ; \boldsymbol{\theta}) d \boldsymbol{\lambda}$
- $q(z; \boldsymbol{\theta})$ captures dependencies (through marginalization as above !)
- resulting ELBO can be made tractable, by
  - further **lower-bounding the resulting entropy** &
  - sampling from the hierarchical model
- this approach is used in development of **Variational Gaussian Process (VGP)**

Variational Gaussian Process (VGP)

- applies GP to generative variational estimates

  (thus form a Bayesian non-parametric prior)
- able to approximate diverse posterior distn

(b) copula VI

instead of fully-factorized variational distn, use form as below :

$q(z)=\left(\prod_{i} q\left(z_{i} ; \lambda_{i}\right)\right) c\left(Q\left(z_{1}\right), \ldots, Q\left(z_{N}\right)\right).$

- c : copula distn

  (= joint distn over marginal cumulative distn functions $Q\left(z_{1}\right), \ldots, Q\left(z_{N}\right)$)

VI for time series

ex) Hidden Markov Models (HMM), Dynamic Topic Models (DTM)

have strong **dependencies** between time steps

Thus, typically employs a STRUCTURED variational distn

(capture dependencies between time points, while remaining fully-factorized in the remaining variables)

6-4. Other Non-standard VI methods

Methods that improve accuracy of VI, but

- not categorized as alternative measures
- or structured models

(a) VI with Mixture distn

Very flexible! (+ but also computationally difficult)

To fit a mixture models, can use **auxiliary bounds**, **fixed point update**, etc....

ex) **Boosting VI, Variational boosting**

- refine the approximate posterior ITERATIVELY by adding one component at time

  (while keeping previously fitted components fixed)

(b) VI by Stochastic Gradient Descent

SGD on NLL can be seen as an **IMPLICIT VI** algorithm!

consider SGD with

- 1) constant learning rates, **constant SGD**
- 2) **early stopping**

Constant SGD

- can be viewed as a Markov chain that converges to stationary distn

- variance of stationary distn is controlled by the learning rate

Early stopping

- interpret SGD as non-parametric
- track entropy changes based on estimates of "Hessian"

(c) Robustness to Outliers & Local Optima

ELBO : non-convex $\rightarrow$ VI benefits from advanced optimization algorithms

- ex) **Variational tempering**

7. Amortized VI and Deep Learning

(Previous : not-amortized)

- x_i is governed by its latent variable z_i, with variational parameter ξ_i

Amortized Variational inference

- use powerful predictor to predict the optimal z_i, **based on the features of** x_i
  (i.e. $z_i = f(x_i)$)
- local variational params (ξ_i) are replaced by a function of the data, whose params are shared across all the data points! (called **"inference is amortized"**)

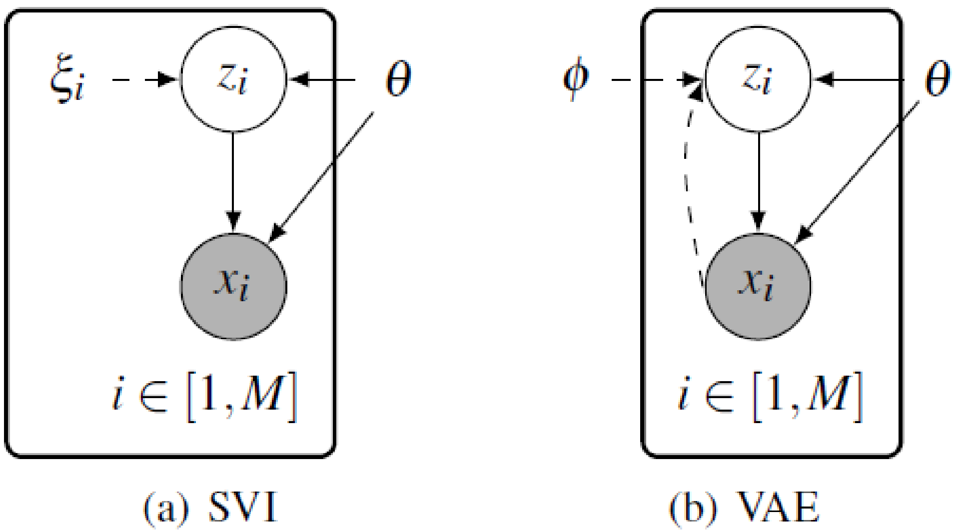

Fig. 2. The graphical representation of stochastic variational inference (a) and the variational autoencoder (b). Dashed lines indicate variational approximations.

7-1. Amortized VI

"Amortized Inference" : utilizing inferences from past computations

"Amortized Inference in VI" : inference over **local variables**

- instead of approximating separate variables,
- assumes that local variational parameters can be **predicted by a function of the data**
- DNN used in this context is called **INFERENCE network**

Amortized VI with inference networks

= (1) probabilistic modeling + (2) representational power of DL

DGPs (Deep Gaussian Processes)

- apply amortized inference
- inference is intractable! solution?
  - (method 1) apply MFVI with inducing points
  - (method 2) propose to estimate these latent variables as a functions of inference networks

    (allowing to scale to bigger dataset)

7-2. Variational Auto Encoder (VAE)

Amortized VI has become poupluar tool for inference in DLGM

$\rightarrow$ leads to VAEs

(a) Generative Model

introduce class of DLGMs

Generative Process

- draw latent variable $z: p(z)=\mathcal{N}(0, \mathbb{I})$
- more generally, can use prior that depends on θ : $p_\theta(\boldsymbol{x} \mid \boldsymbol{z})=\prod_{i=1}^N \mathcal{N}\left(x_i ; \mu\left(z_i\right), \sigma^2\left(z_i\right) \mathbb{I}\right)$
  - likelihood depends on z through two non-linear functions $\mu(\cdot)$ and $\sigma(\cdot)$ (typically NN)
  - θ entails the parameters of the networks $\mu(\cdot)$ and $\sigma(\cdot)$

DLGMs are very FLEXIBLE density estimators!

Modified version

- for binary data, Gaussian likelihood can be replaced by Bernoulli likelihood

(b) VAE

VAEs refer to DLGMs which are trained using **inference networks**

Architecture

- Encoder = RECOGNITION network / INFERENCE network
- Decoder = GENERATIVE nework

Amortized mean-field variational distn

- To approximate posterior, VAE employ **amortized mean-field variational distn**

  $q_\phi(z \mid x) = \prod_{i=1}^{N} q_\phi\left(z_i \mid x_i\right)$

- Typically chosen as

  $q_\phi\left(z_i \mid x_i\right) = \mathscr{N}\left(z_i \mid \mu\left(x_i\right), \sigma^2\left(x_i\right)\mathbb{I}\right)$

- similar to generative model, employs non-linear mappings $\mu(x_i)$ and $\sigma(x_i)$ (ex. NN)

During optimization, **both INFERENCE & GENERATIVE networks** are trained **jointly** to maximize **ELBO**

use **Reparameterization Trick**

- $z_{(i,l)} = \mu\left(x_i\right) + \sigma\left(x_i\right) * \varepsilon_{(i,l)}$
- ELBO :

  $$\begin{aligned} \hat{\mathscr{L}}\left(\theta, \phi, x_i\right) = & -D_{KL}\left(q_\phi\left(z_i \mid x_i\right) \| p_\theta\left(z_i\right)\right) \\ & + \frac{1}{L} \sum_{l=1}^{L} \log p_\theta\left(x_i \mid \mu\left(x_i\right) + \sigma\left(x_i\right) * \varepsilon_{(i,l)}\right) \end{aligned}$$

- differentiate w.r.t θ and ϕ
- also implies that the gradient variance is bounded by a constant

(c) A probabilistic Encoder-Decoder Persepectiove

Auto Encoder

- DNN that are trained to reconstruct their inputs
- bottleneck forces network to learn a **compact** representation of the data

Variational Auto Encoder

- probabilistic model
- hidden variable of VAE can be thought of as **intermediate representations** of the data in the bottle neck of an auto encoder
- during training, **inject noise** into the intermediate layer

  KL-divergence term makes posterior close to the prior

  $\rightarrow$ **regularizing effect**

- When noise is reduced to zero, VAE = AE

7-3. Advancements in VAEs

Lots of extensions have been proposed

Summarize as below : extensions that modify the..

- 1) variational approximations q_ϕ
- 2) model p_θ
- 3) dying units problem

(a) Flexible Variational Distributions q_θ

q_θ can be explicit distn (ex. Gaussian, discrete distn...)

More flexible distn can be made by **transforming a simple parametric distn**

- **Implicit distributions**
- **Normalizing Flow (NF)**
- **Importance Weighted VAE (IWAE)**

Implicit distributions

- can be used, since closed-form density is not required

  (only need them to be able to sample from!)
- reparameterization gradients can still be computed
- VI requires the computation of **log density ratio** (= $\log p(z) - \log q_\phi(z \mid x)$)

  $\rightarrow$ can use GAN style discriminator T, that discriminate prior & variational distn

  $T(\boldsymbol{x}, \boldsymbol{z}) = \log q_\phi(\boldsymbol{z} \mid \boldsymbol{x}) - \log p(\boldsymbol{z})$

Normalizing Flow (NF)

- transform simple approximate posterior $q(z)$ into more expressive distn
- transform it using an **invertible** function

  $z \sim q(z)$, $z' = f(z)$

  $q\left(z'\right) = q(z)\left|\frac{\partial f^{-1}}{\partial z'}\right| = q(z)\left|\frac{\partial f}{\partial z'}\right|^{-1}$
- necessary that we compute the determinant!
- choose transformation function f such that $\left|\frac{\partial f}{\partial z'}\right|$ is easily computable!
- variants :
  - Linear time transformations, Langevin and Hamlitonian flow
  - IAF (Inverse Autoregressive Flow)
  - Autoregressive Flow

Implicit distribution & Normalizing Flow

- both share common idea of using **transformations**, to transform simple into complicated!
- difference) NF : density of $q(z)$ can be estimated due to **invertible transformation** function

Importance Weighted VAE (IWAE)

- originally proposed to tighten the ELBO
- reinterpreted to **sample from a more flexible distn**
- require L samples from approximate posteriors, weighted by the ratio

  $\hat{w}_l = \frac{w_l}{\sum_{l=1}^{L} w_l}, \text{ where } w_l = \frac{p_\theta(x_i, z_{(i,l)})}{q_\phi(z_{(i,l)} \mid x_i)}$
- bigger L, tighter ELBO
- same as VAE, but sample from a **more expressive distn**

  (which converges pointwise to the true posterior as $L \rightarrow \infty$)
- introduce a **biased** esitmator

  (better variance-bias trade-offs can be taken)

(b) Modeling Choices of p_θ

improving the prior in VAE can lead to more interpretable fits & better model performance!

- ex) **Structured Prior** for VAE
  - overcome the intractability by learning variational params with a recognition model

Other approaches tackle the assumption "likelihood factorizes over dimensions"

- ex) Deep Recurrent Attentive Writer (relies on a recurrent structure)
- ex) PixelVAE (dependencies between pixels, using conditional model below)

  $p_\theta(x_i \mid z_i) = \prod_j p_\theta\left(x_i^j \mid x_i^1, \ldots x_i^{j-1}, z_i\right).$

(c) Dying units problem

= Learning a good low-dim representation fails!

2 main effects are responsible!

- 1) TOO powerful decoder
- 2) KL-divergence term

TOO powerful decoder

- so strong that some dimensions of z are ignored

  (might model $p_\theta(\mathbf{x} \mid \mathbf{z})$ independently of $\mathbf{z}$)
- in this case, "true posterior = prior", thus **variational distn tries to match prior**

- solve) **Lossy VAE**
  - by conditioning the decoding distn for each output dimension on partial input information
  - force the distn to encode global info in the latent variables

KL-divergence term

- ELBO can be rewritten as **sum of 2 KL-div**

  $\hat{\mathscr{L}}\left(\theta, \phi, x_i\right)=-D_{KL}\left(q_\phi\left(z \mid x_i\right) \| p_\theta(z)\right)-D_{KL}\left(p\left(x_i\right) \| p_\theta\left(x_i \mid z\right)\right)+C$
- if the model is expressive enough $\rightarrow$ second term = 0

  then, will try to satisfy only the first term

  $\rightarrow$ thus, inference model places its probability mass to match the prior