

# Density Ratio Estimation in Variational Bayesian Machine Learning

Alexander Lam

Department of Mathematics and Statistics  
UNSW

Statistics Honours, 2018

## 1 Background Information

- Neural Networks
- (Amortized) Variational Inference
- Density Ratio Estimation

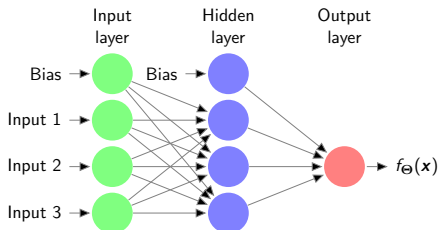
## 2 Undertrained Estimator Experiment

## 3 Autoencoder Experiment

# Neural Networks

## Overall Structure

- Objective is to approximate a function  $f^*$  using mapping with parameters  $\Theta$ :  $\mathbf{f}_{\Theta}(\mathbf{x})$ .
- Universal Approximation Theorem states a neural network can **approximate (almost) any function** if it is complex enough.
- Each node output is a transformed, weighted sum of previous node outputs.



- Weights trained such that (ideally convex) loss function is minimized  
e.g. Mean Squared Error:  $\min_{\Theta} \frac{1}{2} \|\mathbf{y} - \mathbf{f}_{\Theta}(\mathbf{x})\|_2^2$ .
- Back-propagation finds partial derivatives of loss function with respect to weights.
- **Gradient descent** uses these partial derivatives to optimize network.

# (Amortized) Variational Inference

## Bayesian Inference

- Fundamental problem in Bayesian computation is to **estimate posterior densities**  $p(z|x)$ :

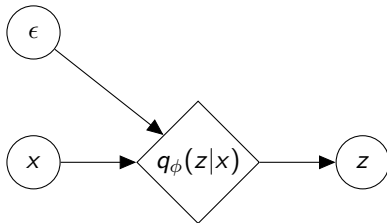
$$p(z|x) \propto \underbrace{p(z)}_{\text{Prior}} \underbrace{p(x|z)}_{\text{Likelihood}} .$$

- Typical MCMC methods are slow with large datasets or high dimensional data.
- Variational Inference is a solution.

# (Amortized) Variational Inference

## Introduction

- Amortized variational inference approximates  $p(z|x)$  with a different distribution  $q_\phi(z|x)$ .
- $q_\phi(z|x)$  is a **neural network** with parameters  $\phi$  that takes in data  $x$  and random noise  $\epsilon \sim \mathcal{N}(0, I_{n \times n})$  and outputs samples  $z \sim q_\phi(z|x)$ .



# (Amortized) Variational Inference

## Network Training

- Train network by minimizing the (expected) reverse KL divergence between the two distributions:

$$\mathbb{E}_{q^*(x)}[KL(q(z|x)||p(z|x))] = \mathbb{E}_{q^*(x)q(z|x)} \left[ \log \left( \frac{q(z|x)}{p(z|x)} \right) \right]$$

.

- This is the same as solving the following minimization problem:

$$\min_{\phi} \underbrace{-\mathbb{E}_{q_{\phi}(z|x)q^*(x)}[\log p(x|z)]}_{\text{Likelihood}} + \underbrace{\mathbb{E}_{q^*(x)}[KL(q_{\phi}(z|x)||p(z))]}_{\text{Log Density Ratio}}.$$

- We call this  $NELBO(q)$  as it is the **n**egative of **e**vidence **l**ower **b**ound.

# (Amortized) Variational Inference

## Problems with Implicit Distributions

Consider our log density ratio term

$$KL(q_\phi(z|x)||p(z)) = \mathbb{E}_{q_\phi(z|x)} \left[ \frac{q_\phi(z|x)}{p(z)} \right].$$

- $q_\phi(z|x)$  is an **implicit** distribution.
- Use density ratio estimation to evaluate  $\frac{q_\phi(z|x)}{p(z)}$  in  $KL(q_\phi(z|x)||p(z))$ .
- Density ratio estimation only requires **samples** from the distributions.



# Density Ratio Estimation

## Class Probability Estimation

We want to estimate  $\frac{q(u)}{p(u)}$ .

- 1 Define discriminator network with parameters  $\alpha$  that finds probability that a sample  $u$  came from  $q(u)$ .

- 2 Train discriminator with Bernoulli loss:  
$$\min_{\alpha} -\mathbb{E}_{q(u)}[\log D_{\alpha}(u)] - \mathbb{E}_{p(u)}[\log(1 - D_{\alpha}(u))].$$

- 3 Optimal discriminator is  $D_{\alpha}^*(u) = \frac{q(u)}{q(u)+p(u)}$ .

$$\frac{q(u)}{p(u)} = \frac{D_{\alpha}^*(u)}{1 - D_{\alpha}^*(u)}$$

# Density Ratio Estimation

## Divergence Minimisation

### Theorem

*If  $f$  is a convex function with derivative  $f'$  and convex conjugate  $f^*$ , and  $\mathcal{R}$  is a class of functions with codomains equal to the domain of  $f'$ , then we have the lower bound for the  $f$ -divergence between distributions  $p(u)$  and  $q(u)$ :*

$$D_f[p(u)||q(u)] \geq \sup_{r \in \mathcal{R}} \{ \mathbb{E}_{q(u)}[f'(r(u))] - \mathbb{E}_{p(u)}[f^*(f'(r(u)))] \},$$

*with equality when  $r(u) = q(u)/p(u)$ .*

For the reverse KL divergence,  $f(u) = u \log u$  so letting  $r_\alpha(u) \simeq \frac{q(u)}{p(u)}$  we have:

$$KL[q(u)||p(u)] \geq \sup_{\alpha} \{ \mathbb{E}_{q(u)}[1 + \log r_\alpha(u)] - \mathbb{E}_{p(u)}[r_\alpha(u)] \}$$

# Density Ratio Estimation

## Algorithm Generalisation

- Actually, upper bound  $f$ -divergence of  $\underbrace{2JS(p(u)||q(u)) - \log 4}_{\text{GAN Divergence}}$  and

$D(u) = \frac{r(u)}{r(u)+1}$  leads to class probability estimation loss function.

- Choose either reverse KL or GAN  **$f$ -divergence bound** and **estimator parametrisation**:
  - Class Probability Estimator  $D_\alpha(u) \simeq \frac{q(u)}{q(u)+p(u)}$
  - Direct Ratio Estimator  $r_\alpha(u) \simeq \frac{q(u)}{p(u)}$
  - Direct Log Ratio Estimator  $T_\alpha(u) \simeq \log \frac{q(u)}{p(u)}$ .

To train a variational posterior network:

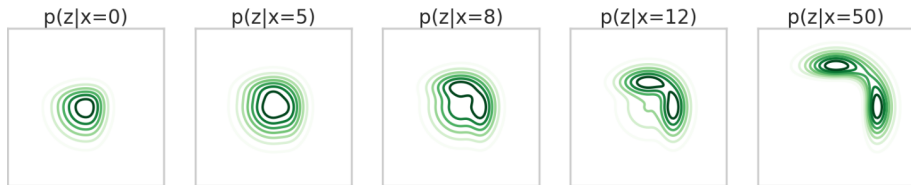
- 1 Train estimator network until convergence.
- 2 Use estimator network to calculate intractable term in  $NELBO$ .
- 3 Take one optimisation step of posterior network.
- 4 Repeat until posterior convergence.

# Undertrained Estimator Experiment

## Experiment Outline

$$p(z_1, z_2) \sim \mathcal{N}(0, \sigma^2 I_{2 \times 2})$$

$$p(x|z) \sim \text{Exp}(3 + \max(0, z_1)^3 + \max(0, z_2)^3)$$



- Posterior is flexible and bimodal.
- Use Gaussian KDE to find 'true' KL divergence for  $q_\phi(z|x = 0, 5, 8, 12, 50)$ .

# Undertrained Estimator Experiment

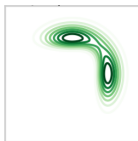
## Experiment Outline

- In a previous experiment we found that all three estimator parametrisations lead to similar results when optimised effectively.
- What if they are poorly optimised?
- Training parameters:
  - High posterior training rate.
  - Low estimator training rate.
  - Low estimator to posterior iteration ratio (11:1).

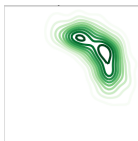
# Undertrained Estimator Experiment

## Results

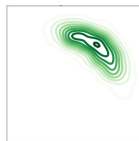
Algorithm	Mean KL Divergence	Standard Deviation
Reverse KL - $D_\alpha(z, x)$	<b>1.3786</b>	<b>0.0286</b>
Reverse KL - $r_\alpha(z, x)$	1.3934	0.0410
Reverse KL - $T_\alpha(z, x)$	1.4133	0.0597
GAN - $D_\alpha(z, x)$	1.4017	<b>0.0286</b>
GAN - $r_\alpha(z, x)$	1.4086	0.0555
GAN - $T_\alpha(z, x)$	1.4214	0.0518



True



Reverse KL

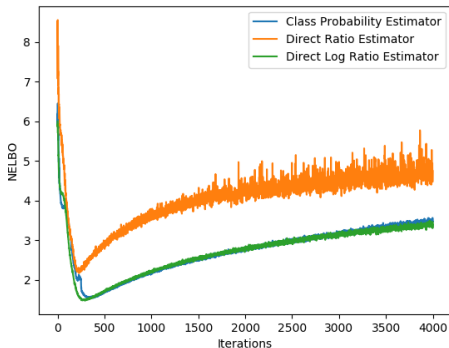


GAN

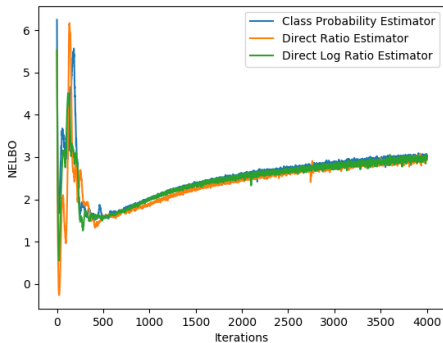
- Reverse KL divergence better than GAN divergence.
- $D_\alpha(z, x) < r_\alpha(z, x) < T_\alpha(z, x)$

# Undertrained Estimator Experiment

## NELBO Plots



(a) GAN Divergence



(b) Reverse KL Divergence

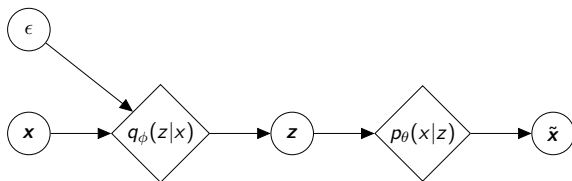
- Unclear why direct ratio estimator has unusual NELBO plot: posterior convergence was not affected.
- Reverse KL Divergence has initial instability.



# Autoencoder Experiment

## Autoencoders

- Likelihood  $p_{\theta}(x|z)$  is now a neural network.
- Posterior  $q_{\phi}(z|x)$  represents data  $x$  as lower dimensional latent  $z$ .
- Likelihood  $p_{\theta}(x|z)$  reconstructs data  $\tilde{x}$  from  $z$ .
- Generate new data  $\tilde{x}$  using  $z$  from  $p(z)$ .



$$\min_{\theta, \phi} \underbrace{-\mathbb{E}_{q_{\phi}(z|x)q^*(x)}[\log p_{\theta}(x|z)]}_{\text{Likelihood}} + \underbrace{\mathbb{E}_{q^*(x)}[KL(q_{\phi}(z|x)||p(z))]}_{\text{Density Ratio}}$$

# Autoencoder Experiment

## Experiment Outline



- MNIST dataset -  $28 \times 28$  grey-scale images of handwritten digits
- Again use undertrained estimator.
- Use reconstruction error  $\|x - \tilde{x}\|^2$  as metric.
- Perform experiment with low dimensional latent space (2 dimensions) and high dimensional latent space (20 dimensions).

# Generation Experiment

Results - high dimensional latent space

Algorithm	Mean Reconstruction Error	Standard Deviation
Reverse KL - $D_{\alpha}(z, x)$	0.0647	0.0019
GAN - $D_{\alpha}(z, x)$	<b>0.0444</b>	<b>0.0017</b>



Reverse KL

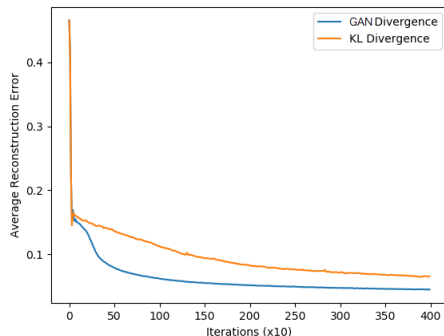


GAN

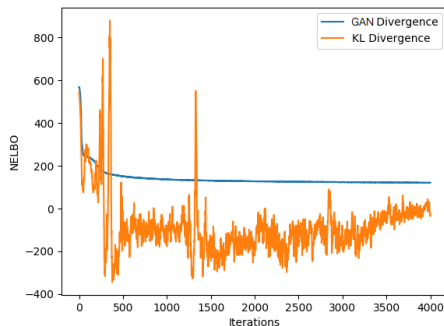
- Direct ratio and direct log ratio estimators attempted to store numbers exceeding float64(max).
- Exponential of  $T_{\alpha}(z, x)$  taken in loss function.
- $D_{\alpha}(z, x)$  ranges in  $(0, 1)$ .

# Autoencoder Experiment

Results - high dimensional latent space



(a) Reconstruction Error



(b) NELBO

- As before, GAN divergence is more stable.
- Recall reverse KL divergence is initially unstable but stabilizes later.
- In this case it fails to stabilise by the end of the program runtime.

- The class probability estimator  $D_\alpha(u) \simeq \frac{q(u)}{q(u)+p(u)}$  is the ‘best’ parametrisation as it can store the **highest density ratios**.
- Reverse KL divergence upper bound may be **unstable** but leads to **faster convergence** when stable.
- Outlook
  - Still unclear exactly why reverse KL divergence is more unstable but more accurate when stable.
  - Several more  $f$ -divergences exist which have unknown stability when undertrained.