

Tom

Will Tebbutt, wct23, Darwin College

Paweł Budzianowski, pfb30, Clare Hall

1. Introduction

2. Stochastic Variational Inference

The aim of Variational Inference is to provide a deterministic approximation to an intractable posterior distribution by finding parameters ϕ such that $D_{KL}(q_\phi(\theta) \parallel p(\theta|D))$ is minimised. This is achieved by noting that

$$\begin{aligned} D_{KL}(q_\phi(\theta) \parallel p(\theta|D)) &= \log p(D) + \mathbb{E}_{q_\phi(\theta)} [\log q_\phi(\theta) - \log p(\theta, D)] \\ &=: \log p(X) - \mathcal{L}(\phi; D). \end{aligned} \quad (1)$$

Noting that $\log p(D)$ is constant w.r.t. ϕ , we can now minimise the KL-divergence by maximising the evidence lower bound (ELBO) \mathcal{L} (that this is indeed a lower bound follows from the non-negativity of the KL-divergence). Aside from some notable exceptions (eg. [Titsias, 2009]) this quantity is not tractably point-wise evaluable. However, if $q_\phi(\theta)$ and $\log p(\theta, D)$ are point-wise evaluable, it can be approximated using Monte Carlo as

$$\mathcal{L}(\phi; D) \approx \frac{1}{L} \sum_{l=1}^L \log p(\theta_l, D) - \log q_\phi(\theta_l), \quad \theta_l \sim q_\phi(\theta) \quad (2)$$

This stochastic approximation to the ELBO is not differentiable w.r.t. ϕ as the distribution from which each θ_l is sampled itself depends upon ϕ , meaning that the gradient of the log likelihood cannot be exploited to perform inference. One of the primary contributions of the paper being reviewed is to provide a differentiable estimator for \mathcal{L} that allows gradient information to be exploited. In particular it notes that if there exists a tractable reparameterisation of the random variable $\theta \sim q_\phi(\theta)$ such that

$$\tilde{\theta} = g_\phi(\epsilon), \quad \epsilon \sim p(\epsilon), \quad (3)$$

then we can approximate the gradient of the ELBO as

$$\mathcal{L}(\phi; D) = \mathbb{E}_{p(\epsilon)} [\log p(\theta, X) - q_\phi(\theta)] \approx \frac{1}{L} \sum_{l=1}^L \log p(\theta_l, X) - \log q_\phi(\theta_l) =: \tilde{\mathcal{L}}^1(\phi; X), \quad (4)$$

where $\theta_l = g_\phi(\epsilon_l)$ and $\epsilon_l \sim p(\epsilon)$. Thus the dependence of the sampled parameters θ on ϕ has been removed, yielding a differentiable estimator provided that both q_ϕ and $\log p(\theta, D)$ are themselves differentiable. Approximate inference can now be performed by computing the gradient of $\tilde{\mathcal{L}}^1$ w.r.t. ϕ either by hand or using one's favourite reverse-mode automatic differentiation package (eg. Autograd [Maclaurin et al.,]) and performing gradient-based stochastic optimisation to maximise the elbo using, for example, AdaGrad [?].

The authors also point out that one can re-express the elbo in the following manner

$$\mathcal{L}(\phi; D) = \mathbb{E}_{q_\phi(\theta)} [\log p(D|\theta)] - D_{KL}(q_\phi(\theta) \parallel p(\theta)). \quad (5)$$

This is useful as the KL-divergence between the variational approximation $q_\phi(\theta)$ and the prior over the parameters θ has a tractable closed-form expression in a number of useful cases. This leads to a second estimator for the elbo:

$$\tilde{\mathcal{L}}^2(\phi; D) := \frac{1}{L} \sum_{l=1}^L \log p(D|\theta_l) - D_{KL}(q_\phi(\theta) \parallel p(\theta)). \quad (6)$$

It seems probable that this estimator will in general have lower variance than $\tilde{\mathcal{L}}^1$.

So far Stochastic Variational Inference has been discussed only in a general parametric setting. The paper's other primary contribution is to use a differentiable recognition network to learn to parameterise the posterior

distribution over latent variables z_i local to each observation x_i in a parametric model. In particular, they assume that given some global parameters θ , $\mathbf{z}_i \sim p_\theta(\mathbf{z})$ and $\mathbf{x}_i \sim p_\theta(\mathbf{x}_i | \mathbf{z}_i)$. In the general case the posterior distribution over each z_i will be intractable. Furthermore, the number of latent variables \mathbf{z}_i increases as the number of observations increases, meaning that under the framework discussed above we would have to optimise the variational objective with respect to each of them independently. This is potentially computationally intensive and quite wasteful as it completely disregards the any information about the posterior distribution over the z_i provided by the similarities between inputs locations $\mathbf{x}_{\neq i}$ and corresponding posteriors $\mathbf{z}_{\neq i}$. To rectify this the recognition model $q_\phi(\mathbf{z} | \mathbf{x})$ is introduced.

Given the recognition model and a point estimate for θ , the ELBO becomes

$$\begin{aligned} \mathcal{L}(\theta, \phi; D) &= \mathbb{E}_{q_\phi(\mathbf{z}_1), \dots, q_\phi(\mathbf{z}_N)} \left[\log \prod_{i=1}^N p_\theta(\mathbf{x}_i, \mathbf{z}_i) - \log \prod_{i=1}^N q_\phi(\mathbf{z}_i) \right] \\ &= \sum_{i=1}^N \mathbb{E}_{q_\phi(\mathbf{z}_i)} [\log p_\theta(\mathbf{x}_i, \mathbf{z}_i) - \log q_\phi(\mathbf{z}_i)] \end{aligned} \quad (7)$$

For this ELBO we can derive a similar result to $\tilde{\mathcal{L}}^1$, where we do not assume a closed-form solution for the KL divergence between distributions and include mini-batching to obtain an estimator for the ELBO for a mini-batch of observations

$$\tilde{\mathcal{L}}^A(\theta, \phi; D) \approx \frac{N}{LM} \sum_{i=1}^M \sum_{l=1}^L \log p_\theta(\mathbf{x}_i, \mathbf{z}_{i,l}) - \log q_\phi(\mathbf{z}_{i,l}), \quad \mathbf{z}_{i,l} = g_\phi(\mathbf{x}_i, \epsilon_{i,l}), \quad \epsilon_{i,l} \sim p(\epsilon) \quad (8)$$

where the M observations in the mini-batch are drawn uniformly from the data set comprised of N observations and for each observation we draw L samples from the approximate posterior $q_\phi(\mathbf{z}_i | \mathbf{x}_i)$. Similarly, if $q_\phi(\mathbf{z} | \mathbf{x})$ and $p_\theta(\mathbf{z})$ are such that the KL-divergence between them has a tractable closed-form solution then we can use an approximate bound which we could reasonably expect to have a lower variance:

$$\begin{aligned} \mathcal{L}(\theta, \phi; D) &= \mathbb{E}_{q_\phi(\mathbf{z}_1), \dots, q_\phi(\mathbf{z}_N)} \left[\log \prod_{i=1}^N p_\theta(\mathbf{x}_i | \mathbf{z}_i) \right] - D_{KL} \left(\prod_{i=1}^N q_\phi(\mathbf{z}_i | \mathbf{x}_i) \parallel \prod_{i=1}^N p_\theta(\mathbf{z}_i) \right) \\ &= \sum_{i=1}^N \mathbb{E}_{q_\phi(\mathbf{z}_i)} [\log p_\theta(\mathbf{x}_i | \mathbf{z}_i)] - D_{KL}(q_\phi(\mathbf{z}_i | \mathbf{x}_i) \parallel p_\theta(\mathbf{z}_i)) \\ &\approx \frac{N}{M} \sum_{i=1}^M \left[\frac{1}{L} \sum_{l=1}^L \log p_\theta(\mathbf{x}_i | \mathbf{z}_{i,l}) - D_{KL}(q_\phi(\mathbf{z}_i | \mathbf{x}_i) \parallel p_\theta(\mathbf{z}_i)) \right] =: \tilde{\mathcal{L}}^B(\theta, \phi; D), \end{aligned} \quad (9)$$

where $\mathbf{z}_{i,l} = g_\phi(\mathbf{x}_i, \epsilon_{i,l})$ and $\epsilon_{i,l} \sim p(\epsilon)$.

3. The Variational Autoencoder

The Variational Autoencoder exploits the methods described in the previous section to define a probabilistic model whose elbo is highly reminiscent of the objective optimised in a traditional autoencoder. In particular we define a generative model where we assume that the i^{th} observation was generated by first sampling a latent variable $z_i \sim N(0, I)$ and that the each observation vector real-valued observation $x_i \sim N(\mu_\theta(z_i), \sigma_\theta^2(z_i))$ where

$$\mu_\theta(z_i) = h_i W_\mu^{(dec)} + b_\mu^{(dec)}, \quad (10)$$

$$\log \sigma_\theta^2(z_i) = h_i W_\sigma^{(dec)} + b_\sigma^{(dec)}, \quad (11)$$

$h_i \in \mathbb{R}^{1 \times D_h}$ is the output at the final hidden layer of the “decoder MLP”, $W_\mu^{(dec)}, W_\sigma^{(dec)} \in \mathbb{R}^{D_h \times D_z}$ are matrices mapping from the D_h hidden units to the D_z dimensional latent space. Similarly, $b_\mu^{(dec)}, b_\sigma^{(dec)} \in \mathbb{R}^{1 \times D_z}$ are row vector biases. Note that the variances are parameterised implicitly through their logs to ensure that they are correctly valued only on the positive reals.

If the output vectors are binary valued then $x_i \sim \text{Bernoulli}(f_\theta(z_i))$ where again given h_i , the output at the final hidden layer of the decoder MLP for latent-space value z_i

$$f_\theta(z_i) = \left[1 + \exp \left(-h_i W^{(dec)} - b^{(dec)} \right) \right]^{-1}, \quad (12)$$

where $W^{(dec)} \in \mathbb{R}^{D_h \times D_z}$ and $b^{(dec)} \in \mathbb{R}^{1 \times D_z}$.

The recognition model is given by

$$q_\phi(z|x) = \mathcal{N}(z | \mu_\phi(x_i), \sigma_\phi^2(x_i)), \quad (13)$$

where the distributional parameters $\mu_\phi(x_i)$ and $\sigma_\phi^2(x_i)$ are again given by an MLP, which will be referred to as the “encoding MLP”, whose input is x_i .

In this case, there is a tractable closed form solution for the KL-divergence between the variational posterior $q_\phi(z|x)$ and the prior $p_\theta(z)$ meaning that we can approximate the ELBO using equation 9. Additionally we have that

$$g_\phi(x, \epsilon) = \mu_\phi(x_i) + \sigma_\phi(x_i) \odot \epsilon, \quad \epsilon \sim \mathcal{N}(0, I). \quad (14)$$

The algorithm to perform inference in the Variational Autoencoder is provided in algorithm 1.

Algorithm 1 Procedure by which inference is performed in the Variational Autoencoder.

```

 $\theta, \phi \leftarrow$  Initialisation.
while not converged in  $\theta, \phi$  do
    Pick subset of size  $x_{1:M}$  from the full dataset uniformly at random.
    Compute  $\mu_{1:M}, \log \sigma_{1:M}^2$  using the encoding MLP.
    For all  $i \in \{1, \dots, M\}$ , sample  $\epsilon_i \sim \mathcal{N}(0, I)$ .
    For all  $i \in \{1, \dots, M\}$ ,  $z_i \leftarrow \mu_i + \sigma_i \odot \epsilon_i$ .
    For all  $i \in \{1, \dots, M\}$ , compute  $\log p(x_i | z_i)$  using the decoder MLP and likelihood function.
    Compute  $g \leftarrow \nabla_{\theta, \phi} \mathcal{L}^B(\theta, \phi; x_{1:M}, \epsilon_{1:M})$ 
    Update  $\theta, \phi$  using noisy gradient estimate  $g$ .
end while

```

4. Reconstruction

On frey faces: used epochs 100 for latent space 2, 550 for 10 and 20

Tried mean and sampling for VAE, mean worked better

5. Full variational bayes

Tried FVB on random initialisation, however performed really badly: continuous: Lower bound: -53582.4758831, time: 0.483903589 Lower bound on validation set: -245029.915134 discrete: Lower bound: -57463.0461588, time: 4.11004209518 Lower bound on validation set: -5434312.97123

6. Conclusion

References

- [Maclaurin et al.,] Maclaurin, D., Duvenaud, D., and Adams, R. P. Autograd: Effortless gradients in numpy.
- [Titsias, 2009] Titsias, M. K. (2009). Variational learning of inducing variables in sparse gaussian processes. In *International Conference on Artificial Intelligence and Statistics*, pages 567–574.