Variational Inference - Lecture 3

Guido Sanguinetti

Institute for Adaptive and Neural Computation
School of Informatics
University of Edinburgh
gsanguin@inf.ed.ac.uk

April 1, 2019

Today's lecture

1. Black-box variational inference
2. The reparametrisation trick and variational autoencoders
3. Further developments and concluding remarks

Parametric variational inference revisited

- Variational Inference maximises the Evidence Lower BOund (ELBO)

$$\mathcal{L} = \int d\theta q(\theta) \log \frac{p(\mathbf{x}, \theta)}{q(\theta)}$$

  with respect to the variational distribution $q(\theta)$
- If $q(\theta) = q_\lambda(\theta)$ is in a parametric family indexed by $\lambda \in \mathbb{R}^n$, then this is a finite dimensional optimisation problem
- **PROBLEM:** analytical expressions for the gradients contingent on being able to perform expectations analytically

Monte-Carlo estimation

- The ELBO can be rewritten as

$$\mathcal{L} = E_{q_\lambda}\left[\log p(\mathbf{x}, \theta)\right] - H[q_\lambda]$$

  i.e. as an expectation of the joint under the approximating distribution
- If we can easily sample from q_λ, then we can obtained an unbiased estimate of $\mathcal{L}$ by Monte-Carlo
- More importantly, the gradient w.r.t. λ

$$\nabla_\lambda \mathcal{L} = E_{q_\lambda}\left[\nabla_\lambda (\log q_\lambda(\theta)) p(\mathbf{x}, \theta)\right] - \nabla_\lambda H[q_\lambda] \qquad (1)$$

  so it is still an expectation (prove on board)
- A Monte-Carlo estimate of the gradient (1) is a *stochastic* gradient

Mini-batch up-scaling

- In the common case where the data are i.i.d. conditioned on the latent variables, the likelihood is a product

$$p(\mathbf{x}|\theta) = \prod_{i=1}^{N} p(x_i|\theta)$$

- The gradient estimator in (1) contains the logarithm of the likelihood therefore becomes

$$\nabla_\lambda \mathcal{L} = E_{q_\lambda}\left[\sum_i \nabla_\lambda(\log q_\lambda(\theta))p(x_i|\theta)p(\theta)\right] - \nabla_\lambda H[q_\lambda] \quad (2)$$

- Randomly subsampling the data (mini-batch) yields an unbiased estimate of the gradient (**??**)

Black-Box variational inference (Ranganath et al 2013)

- Framework for parametric variational inference by stochastic gradient descent
- Assumptions: q distribution is easy to sample from, likelihood $p(x|\theta)$ can be computed easily and observaitons are iid
- Procedure: sample mini-batch, Monte-Carlo estimate gradient from finet sample from q_λ, take noisy gradient step
- In the paper, additional tricks to reduce variance of the estimator via Rao-Blackwellisation or control variables (now not used)

Autoencoders

- "Old" neural networks way of performing unsupervised learning
- Data are regressed on themselves via nonlinear maps (*encoder* and *decoder*) going through a low-dimensional bottleneck
- If the neural networks are linear (a.k.a. matrices), then the autoencoder is PCA
- Nonlinear neural networks can in principle capture non-trivia structure in data

Autoencoders

- "Old" neural networks way of performing unsupervised learning
- Data are regressed on themselves via nonlinear maps (*encoder* and *decoder*) going through a low-dimensional bottleneck
- If the neural networks are linear (a.k.a. matrices), then the autoencoder is PCA
- Nonlinear neural networks can in principle capture non-trivia structure in data
- **BIG CAVEAT**: PCA solution is unique modulo rotation. Nonlinear autoencoders have multiple local optima; looking at structures may or may not make sense.

Variational autoencoders (Kingma and Welling 2014)

- Originally formulated as free-form variational inference for a general dimensionality reduction model $p(\mathbf{x}|\mathbf{z})$
- In practice, for continuous $\mathbf{x}$ the assumption is that $p(\mathbf{x}|\mathbf{z}) = \mathcal{N}(\mu_\theta(\mathbf{z}), \sigma_\theta(\mathbf{z}))$ where $\mu_\theta(\mathbf{z}), \quad \sigma_\theta(\mathbf{z})$ are complex nonlinear functions (neural networks) parametrised by (weights) θ (switched notation to paper)
- The posterior $p(\mathbf{z}|\mathbf{x})$ is approximated variationally as a Gaussian $q_\phi(\mathbf{z})$ using BBVI
- Innovation (and link to autoencoders): the parameters of the approximating distribution are themselves neural networks dependent on the data $\mathbf{x}$

Variational autoencoders (Kingma and Welling 2014)

- Originally formulated as free-form variational inference for a general dimensionality reduction model $p(\mathbf{x}|\mathbf{z})$
- In practice, for continuous $\mathbf{x}$ the assumption is that $p(\mathbf{x}|\mathbf{z}) = \mathcal{N}(\mu_\theta(\mathbf{z}), \sigma_\theta(\mathbf{z}))$ where $\mu_\theta(\mathbf{z}), \quad \sigma_\theta(\mathbf{z})$ are complex nonlinear functions (neural networks) parametrised by (weights) θ (switched notation to paper)
- The posterior $p(\mathbf{z}|\mathbf{x})$ is approximated variationally as a Gaussian $q_\phi(\mathbf{z})$ using BBVI
- Innovation (and link to autoencoders): the parameters of the approximating distribution are themselves neural networks dependent on the data $\mathbf{x}$
- Discuss various interpretations of this

Stein variational gradient (Liu and Wang 2016)

- New idea for *exact* gradient-based inference using functional optimisation
- The idea is to deform a base distribution using a bijective transformation $\mathbf{T}(\mathbf{z})$ that will map a base distribution $q(\mathbf{z})$ into the desired posterior $p(\mathbf{z}|\mathbf{x})$
- The optimal (infinitesimal) transformation is given by the so-called Stein gradient, which is the gradient of the KL divergence $KL[q\|p]$ evaluated at q
- Amazingly, this gradient transformation can be computed *analytically* when restricting to a RKHS

Stein variational gradient (Liu and Wang 2016)

- New idea for *exact* gradient-based inference using functional optimisation
- The idea is to deform a base distribution using a bijective transformation $\mathbf{T}(\mathbf{z})$ that will map a base distribution $q(\mathbf{z})$ into the desired posterior $p(\mathbf{z}|\mathbf{x})$
- The optimal (infinitesimal) transformation is given by the so-called Stein gradient, which is the gradient of the KL divergence $KL[q\|p]$ evaluated at q
- Amazingly, this gradient transformation can be computed *analytically* when restricting to a RKHS
- Algorithm: sample a set of particles from a starting distribution q_0 (e.g. Gaussian) and iteratively transform them using the Stein gradients.

Deterministic variational inference (Wu et al 2019)

- How do we compute the gradient in BBVI or VAEs?
- The sample-based approximation needs to be propagated through layers to obtain an estimate of the likelihood $\rightarrow$ potentially problems due to sampling in regions of low posterior mass/ strong non-linearities
- Wu et al propose to propagate *moments* of the q distribution through the layers, instead of particles
- Some Central Limit Theorem justification, empirical validation rather impressive

Final considerations

- VI is a very powerful approximation to perform Bayesian inference
- In general, quality of the approximation can be poor when the true distribution is far from the approximator
- In neural network context, ELBO provides a differentiable objective for complex unsupervised learning models
- NOT a credible approach to quantify uncertainty
- Many interesting novel approaches in recent years