Understanding Reparameterisation Gradients

Philip Schulz and Wilker Aziz

last modified: October 31, 2017

Abstract

This note explains in detail the reparametrisation trick presented in (Kingma and Welling, 2013; Rezende et al., 2014; Titsias and Lázaro-Gredilla, 2014). Our derivation mostly follows that of Titsias and Lázaro-Gredilla (2014). It also gives some advice how terms such as Jacobians should be distributed.

We assume a model of some data x whose log-likelihood is given as

$$\log p(x|\theta) = \int \log \underbrace{p(x, z|\theta)}_{g(z)} \, \mathrm{d}z \tag{1}$$

where x is any set of latent variables and θ are the model parameters. The joint likelihood, which we abbreviate as $g(z)$, can be arbitrarily complex; in particular, it can be given by a neural network. Since for complex models exact integration over the latent space is impossible, we employ variational inference for parameter optimisation. The variational parameters are called λ. The objective is

$$\arg\max_{\lambda} \mathbb{E}_{q(z;\lambda)}\left[\log \frac{g(z)}{q(z;\lambda)}\right] \ . \tag{2}$$

We further assume that exact integration is not possible even under the variational approximation (this is the case in non-conjugate models, such as neural networks). Instead we want to sample gradient estimates using Monte Carlo (MC) sampling. Unfortunately, the MC estimator is not differentiable, thus we derive what we call *reparameterised gradients* which apply to models where Z is a continuous random variable.

1 Location-scale q

In this section we focus on a restricted class of approximate posteriors for which Z can be represented by transforming samples from a standard distribution $\phi(\epsilon)$ using an affine transformation:

$$\epsilon = h(z;\lambda) = C^{-1}(z-\mu) \tag{3a}$$

$$z = h^{-1}(\epsilon;\lambda) = \mu + C\epsilon \ . \tag{3b}$$

Note that $\phi(\epsilon)$ does not depend on $\lambda = \{\mu, C\}$ which was absorbed in the affine transformation—this in fact restricts the class of approximations $q(z;\lambda)$ to location-scale distributions.[1]

For the sake of generality we take z and ϵ to be vector valued. Then we write $J_{h(z;\lambda)}$ to denote the Jacobian matrix of the transformation $h(z;\lambda)$, and $J_{h^{-1}(\epsilon;\lambda)}$ to denote the Jacobian matrix of the inverse transformation.[2] An important property, which we will use to derive

[1] The vector μ is called the *location* and C is a positive definite matrix called the *scale*.

[2] Recall that a Jacobian matrix $\mathbf{J} \triangleq J_{f(x)}$ of some vector value function $f(x)$ is such that $J_{i,j} = \frac{\partial}{\partial x_j} f_i(x)$.

reparameterised gradients, is that the inverse of a Jacobian matrix is related to the Jacobian matrix of the inverse function by $J_{f^{-1}} \circ f(x) = J^{-1}_{f(x)}$.[3]

For an invertible transformation of random variables, it holds that

$$q(z;\lambda) = \phi(h(z;\lambda))\left|\det J_{h(z;\lambda)}\right| \tag{4}$$

and therefore for the transformation in (3) we can write

$$q(z;\lambda) = \phi(C^{-1}(z-\mu))\left|\det C^{-1}\right| \tag{5}$$

and

$$\phi(\epsilon) = q(\mu + C\epsilon;\lambda)|\det C| \ . \tag{6}$$

In the following block of equations (7) we will re-express the expectation in Equation (2) in terms of the parameter-free standard density $\phi(\epsilon)$. The derivation relies on several identities, thus we will break it down into small steps. We start by a change of density

$$\int q(z;\lambda) \log \frac{g(z)}{q(z;\lambda)} \mathrm{d}z \tag{7a}$$

$$= \int \phi(\underbrace{h(z;\lambda)}_{\epsilon})\left|\det J_{h(z;\lambda)}\right| \log \frac{g(z)}{\phi(h(z;\lambda))\left|\det J_{h(z;\lambda)}\right|} \mathrm{d}z \tag{7b}$$

where we use the identity in (4) to introduce $\phi(\epsilon)$. Note, however, that the variable of integration is still z and therefore we have expressed every integrand—including $\phi(\epsilon)$—as a function of z. We now proceed to perform a change of variable

$$= \int \phi(\epsilon)\left|\det J_h \circ h^{-1}(\epsilon;\lambda)\right| \log \frac{g(h^{-1}(\epsilon;\lambda))}{\phi(\epsilon)|\det J_h \circ h^{-1}(\epsilon;\lambda)|} \underbrace{\left|\det J_{h^{-1}(\epsilon;\lambda)}\right| \mathrm{d}\epsilon}_{\mathrm{d}z} \tag{7c}$$

which calls for a change of infinitesimal volumes, i.e. $\mathrm{d}z = |\det J_{h^{-1}}(\epsilon;\lambda)|\mathrm{d}\epsilon$, and requires expressing every integrand as a function of ϵ rather than z. Note that, to express the Jacobian $J_{h(z;\lambda)}$ as a function of ϵ, we used function composition. At this point we can use the inverse function theorem

$$= \int \phi(\epsilon)\left|\det J^{-1}_{h^{-1}(\epsilon;\lambda)}\right| \log \frac{g(h^{-1}(\epsilon;\lambda))}{\phi(\epsilon)\left|\det J^{-1}_{h^{-1}(\epsilon;\lambda)}\right|} \left|\det J_{h^{-1}(\epsilon;\lambda)}\right| \mathrm{d}\epsilon \tag{7d}$$

to rewrite both Jacobian terms of the kind $J_h \circ h^{-1}(\epsilon;\lambda)$ as inverse Jacobians. This is convenient because the determinant of invertible matrices is such that $\det A^{-1} = \frac{1}{\det A}$ which we can use to re-arrange the inverse Jacobian terms

$$= \int \phi(\epsilon) \frac{1}{|\det J_{h^{-1}}(\epsilon;\lambda)|} \log \frac{g(h^{-1}(\epsilon;\lambda))|\det J_{h^{-1}}(\epsilon;\lambda)|}{\phi(\epsilon)} |\det J_{h^{-1}}(\epsilon;\lambda)|\mathrm{d}\epsilon \tag{7e}$$

revealing that some of them can be cancelled. We are now left with a simpler expectation wrt $\phi(\epsilon)$

$$= \int \phi(\epsilon) \log \frac{g(h^{-1}(\epsilon;\lambda))|\det J_{h^{-1}}(\epsilon;\lambda)|}{\phi(\epsilon)} \mathrm{d}\epsilon \tag{7f}$$

and we can proceed to solve the Jacobian of the affine transformation

$$= \int \phi(\epsilon) \log \left(g(h^{-1}(\epsilon;\lambda)) \left|\det \underbrace{J_{h^{-1}}(\epsilon;\lambda)}_{C}\right| \right) \mathrm{d}\epsilon \underbrace{- \int \phi(\epsilon) \log \phi(\epsilon) \mathrm{d}\epsilon}_{\mathbb{H}[\phi(\epsilon)]} \tag{7g}$$

and to separate out the expected log-denominator (an entropy term). Finally, recall that $\phi(\epsilon)$ does not depend on C and therefore the log-determinant is constant with respect to the standard distribution and can be pushed outside the expectation.

$$= \mathbb{E}_{\phi(\epsilon)}[\log g(h^{-1}(\epsilon;\lambda))] + \log|C| + \mathbb{H}[\phi(\epsilon)] \tag{7h}$$

[3] The notation $J_{f^{-1}} \circ f(x)$ denotes function composition, that is, $J_{f^{-1}(y=f(x))}$ or equivalently $J_{f^{-1}(y)}\big|_{y=f(x)}$.

Note that every expectation in (7h) is taken with respect to $q(\epsilon)$ which does not depend on λ, thus the gradient of (2) wrt λ can be re-expressed as shown in Equation (8).

$$\begin{aligned}
\nabla_\lambda \mathbb{E}_{q(z;\lambda)}\left[\log \frac{g(z)}{q(z;\lambda)}\right] &= \nabla_\lambda \left(\mathbb{E}_{\phi(\epsilon)}[\log g(h^{-1}(\epsilon;\lambda))] + \log|C| + \mathbb{H}[\phi(\epsilon)]\right) && \text{(8a)} \\
&= \mathbb{E}_{\phi(\epsilon)}[\nabla_\lambda \log g(h^{-1}(\epsilon;\lambda))] + \nabla_\lambda \log|C| + \nabla_\lambda \mathbb{H}[\phi(\epsilon)] && \text{(8b)} \\
&= \mathbb{E}_{\phi(\epsilon)}\underbrace{[\nabla_{h^{-1}} \log g(h^{-1}(\epsilon;\lambda))\nabla_\lambda h^{-1}(\epsilon;\lambda)]}_{\text{chain rule}} + \nabla_\lambda \log|C| && \text{(8c)}
\end{aligned}$$

Importantly, note that the first term can be estimated via MC, and that is exactly what automatic differentiation/backprop computes for a given sample, while the second term can be found analytically.

Noteworthy Points

- The cancellation of the absolute value of the Jacobian determinant and
- We can usually rewrite $g(z) = p(x|z,\theta)p(z|\theta)$. This enables us to split up the objective function as

$$\mathbb{E}_{q(z;\lambda)}\left[\log \frac{p(x|z,\theta)p(z|\theta)}{q(z;\lambda)}\right] = \mathbb{E}_{q(z;\lambda)}\left[\log p(x|z,\theta)\right] - \mathrm{KL}\left(q(z;\lambda) \,||\, p(z|\theta)\right) \tag{9}$$

In case we can compute the KL term analytically, we do not need to included $\left|\det J_{h^{-1}(\epsilon;\lambda)}\right|$ in the objective.

References

Diederik P. Kingma and Max Welling. Auto-Encoding Variational Bayes. 2013. URL http://arxiv.org/abs/1312.6114.

Danilo J. Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 1278–1286, 2014. URL http://jmlr.org/proceedings/papers/v32/rezende14.pdf.

Michalis Titsias and Miguel Lázaro-Gredilla. Doubly stochastic variational bayes for non-conjugate inference. In Tony Jebara and Eric P. Xing, editors, *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 1971–1979, 2014. URL http://jmlr.org/proceedings/papers/v32/titsias14.pdf.