Logistic Regression and Variational Inference

Iain Murray

Variational methods used to be complicated. After writing down the standard KL-divergence objective (reviewed below), researchers would try to derive clever fixed-point update equations to optimize it. For some models, even some simple ones like logistic regression, this strategy didn't work out. Special-case variational objectives would be crafted for particular models. As a result, the text-book treatments of the applications of variational methods are fairly complicated and beyond what's required for this course.

Fortunately the stochastic variational methods developed in the last few years are simpler to understand, more general, and scale to enormous datasets. We'll give you a flavour of these.

We review notation (Section 1), but you should understand the longer previous notes on Bayesian logistic regression and Gaussian approximations for the big picture. Straightforward variational inference is possible, but computational expensive (Section 2). Using stochastic gradient descent (Section 3) can make the method scale to huge datasets, and can be easily generalized to more complicated models (like neural networks).

1 Bayesian logistic regression posterior

I'll use the version of the model with labels $y \in \{-1, +1\}$, the probabilities for both settings of y at input location x given the weights can then be written compactly as:

$$P(y \,|\, \mathbf{x}, \mathbf{w}) = \frac{1}{1 + \exp(-y\mathbf{w}^\top \mathbf{x})} = \sigma(y\mathbf{w}^\top \mathbf{x}). \tag{1}$$

The posterior probability can be evaluated up to a constant for any particular weight vector $\mathbf{w}$, given training data $\mathcal{D} = \{\mathbf{x}^{(n)}, y^{(n)}\}_{n=1}^{N}$:

$$P(\mathbf{w} \,|\, \mathcal{D}) \propto p(\mathbf{w})\, p(\mathcal{D} \,|\, \mathbf{w}) = p(\mathbf{w}) \prod_{n=1}^{N} \sigma(y^{(n)} \mathbf{w}^\top \mathbf{x}^{(n)}). \tag{2}$$

Our goal is to approximate this distribution with a Gaussian. We could do that with the Laplace approximation. In this note we'll optimize a variational objective instead.

2 Variational inference

We've motivated in lectures that we may want to form our Gaussian approximation $q(\mu, \Sigma) = \mathcal{N}(\mathbf{w};\, \mu, \Sigma)$ to the posterior $p(\mathbf{w} \,|\, \mathcal{D})$ by minimizing the KL-divergence:

$$D_{\mathrm{KL}}(q(\mathbf{w};\, \mu, \Sigma) \,||\, p(\mathbf{w} \,|\, \mathcal{D}) = E_{\mathcal{N}(\mathbf{w};\, \mu, \Sigma)}\left[\log \frac{\mathcal{N}(\mathbf{w};\, \mu, \Sigma)}{p(\mathbf{w} \,|\, \mathcal{D})}\right], \tag{3}$$

We fit the variational parameters μ and Σ, *not* the original parameters $\mathbf{w}$. Although there is an interpretation that μ is an estimate of $\mathbf{w}$, while Σ indicates a credible range of where the parameters could plausibly be around this estimate.

As we can only evaluate the posterior up to a constant, we write:

$$D_{\mathrm{KL}}(q||p) = \underbrace{E_{\mathcal{N}(\mathbf{w};\, \mu, \Sigma)}\left[\log \frac{\mathcal{N}(\mathbf{w};\, \mu, \Sigma)}{p(\mathbf{w})\, p(\mathcal{D} \,|\, \mathbf{w})}\right]}_{\mathcal{J}} + \log p(\mathcal{D}), \tag{4}$$

and minimize $\mathcal{J}$. Because $D_{\mathrm{KL}}(q||p) \geq 0$, we obtain a lower-bound to the model likelihood[1], $\log p(\mathcal{D}) \geq -\mathcal{J}$.

For logistic regression, we can substitute in the model prior and likelihood (2),

$$\mathcal{J} = E_{\mathcal{N}(\mathbf{w};\,\mu,\Sigma)}\left[\log \mathcal{N}(\mathbf{w};\, \mu, \Sigma) - \log p(\mathbf{w}) - \sum_{n=1}^{N} \log \sigma\big(y^{(n)}\mathbf{w}^\top \mathbf{x}^{(n)}\big)\right]. \tag{5}$$

We could evaluate this expression numerically. The first two expectations can be computed analytically, and the remaining N terms in the sum can each be reduced to a one-dimensional integral (see note on Bayesian predictions). We could similarly evaluate the derivatives wrt μ and Σ, and fit the variational parameters with a gradient-based optimizer. However, in the inner-loop each function evaluation would require N numerical integrations, or further approximation.

3 Stochastic variational inference

We can avoid numerical integration by a simple Monte Carlo estimate of the expectation:

$$\mathcal{J} \approx \frac{1}{S}\sum_{s}\left[\log \mathcal{N}(\mathbf{w}^{(s)};\, \mu, \Sigma) - \log p(\mathbf{w}^{(s)}) - \sum_{n=1}^{N} \log \sigma\big(y^{(n)}{\mathbf{w}^{(s)}}^\top \mathbf{x}^{(n)}\big)\right], \quad \mathbf{w}^{(s)} \sim \mathcal{N}(\mu, \Sigma). \tag{6}$$

A really cheap estimate would use only one sample. We could further form an unbiased estimate of the sum over data points by randomly picking one term:

$$\mathcal{J} \approx \log \mathcal{N}(\mathbf{w}^{(s)};\, \mu, \Sigma) - \log p(\mathbf{w}^{(s)}) - N \log \sigma\big(y^{(n)}{\mathbf{w}^{(s)}}^\top \mathbf{x}^{(n)}\big), \tag{7}$$
$$n \sim \mathrm{Uniform}[1, \ldots, N], \;\; \mathbf{w}^{(s)} \sim \mathcal{N}(\mu, \Sigma).$$

The goal is to move the variational parameters μ and Σ, which express which weights are plausible, so that cost $\mathcal{J}$ gets smaller (on average). If the variational parameters change, the samples $\mathbf{w}^{(s)}$ we'd draw change, which complicates our reasoning.

We can remove the variational parameters from the random draws by writing down how a Gaussian random generator works: $\mathbf{w}^{(s)} = (\mu + \Sigma^{1/2}\nu)$, where $\nu \sim \mathcal{N}(0, I)$. Here $\Sigma^{1/2}$ is a matrix square root, such as the Cholesky decomposition. If we are fitting a diagonal covariance, $\Sigma^{1/2}$ is simply a diagonal matrix of standard deviations. Then our cost function becomes:

$$\mathcal{J} \approx \log \mathcal{N}((\mu + \Sigma^{1/2}\nu);\, \mu, \Sigma) - \log p(\mathbf{w}\!=\!(\mu + \Sigma^{1/2}\nu)) - N \log \sigma\big(y^{(n)}(\mu + \Sigma^{1/2}\nu)^\top \mathbf{x}^{(n)}\big),$$
$$n \sim \mathrm{Uniform}[1, \ldots, N], \;\; \nu \sim \mathcal{N}(0, I). \tag{8}$$

where all the dependence on variational parameters μ and Σ are now in the estimator itself.

We can minimize the cost function $\mathcal{J}$ by stochastic gradient descent. We draw a datapoint n at random, some Gaussian white noise ν, and evaluate (8) and its derivatives wrt μ and Σ. We then make a small change to μ and Σ to improve the current estimate of $\mathcal{J}$ and repeat.

I have omitted some details. Σ needs to be positive semi-definite, so we need an optimizer for this case, or to transform Σ so that it is unconstrained. Also the scale of the gradients wrt μ grows very large when Σ is small. We need an optimizer that can deal with that, or we can scale the gradients wrt μ by multiplying by Σ.

While some of the fine details are slightly complicated, none of them depend on the fact we were considering logistic regression. We could easily replace the logistic function $\sigma(\cdot)$ with another model likelihood, such as from a neural network. As long as we can differentiate the log-likelihood, we can apply stochastic variational inference.

[1] Non-examinable: Some texts call $\log p(\mathcal{D})$ the evidence, and $-\mathcal{J}$ the Evidence Lower BOund (ELBO).