The Hessian of the Log-Posterior for Latent-Gaussian Models

Alex Stringer

2019/10/01

Consider a latent Gaussian model of the form

$$\begin{aligned} Y|W,\theta &\sim \pi(Y|W,\theta) \\ Y &\in \mathbb{R}^N,\ W \in \mathbb{R}^M,\ \theta \in \mathbb{R}^K \\ W &= \begin{pmatrix} U \\ \beta \end{pmatrix} \\ U &\in \mathbb{R}^{M_U},\ \beta \in \mathbb{R}^{M_\beta};\ M_U \ll M_\beta \\ \eta(W) &= AU + X\beta = \begin{pmatrix} A & X \end{pmatrix} W \\ W|\theta &\sim \text{Normal}\left[0, Q(\theta)^{-1}\right] \\ \theta &\sim \pi(\theta) \end{aligned} \tag{1}$$

where A and $Q(\theta)$ are highly sparse matrices. The log-posterior $\pi(W|Y,\theta)$ has the following form:

$$\log \pi(W|Y,\theta) = -\frac{1}{2}W^T Q(\theta) W + \ell(W;Y) \tag{2}$$

where $\ell(W;Y) = \log \pi(Y|W,\theta)$ is a log-likelihood which depends on W only through $\eta(W)$, $\ell(W;Y) \equiv \ell(\eta(W);Y)$. The gradient and Hessian of $\log \pi(W|Y,\theta)$ are required for optimiza-

tion and computation of posterior summaries.

The sample size N and parameter space dimension M can get very large in practice, and storage cost and computation time are a major concern when fitting these models to modern data sets. The sparsity of the gradient and Hessian are a very important practical concern. The gradient $\nabla_W \log \pi(W|Y,\theta) \in \mathbb{R}^M$ and Hessian $\nabla\nabla_{WW^T} \log \pi(W|Y,\theta) \in \mathbb{R}^{M\times M}$ have the following form:

$$\begin{aligned} \nabla_W \log \pi(W|Y,\theta) &= -Q(\theta)W + \nabla_W \ell(W;Y) \\ \nabla\nabla_{WW^T} \log \pi(W|Y,\theta) &= -Q(\theta) + \nabla\nabla_{WW^T} \ell(W;Y) \end{aligned} \tag{3}$$

The log-likelihood $\nabla_W \ell(W;Y)$ is a function of W only through $\eta(W)$, so computation of its gradient and Hessian invoke the chain rule:

$$\begin{aligned} \nabla_W \ell(W;Y) &= \frac{\partial \ell(W;Y)}{\partial W} \\ &= \frac{\partial \eta}{\partial W} \cdot \frac{\partial \ell(W;Y)}{\partial \eta} \\ &= \begin{pmatrix} A^T \\ X^T \end{pmatrix} \cdot \frac{\partial \ell(W;Y)}{\partial \eta} \end{aligned} \tag{4}$$

The $M \times N$ jacobian $\partial\eta/\partial W$ is very sparse, because A is sparse and it is much larger than

X. The $N \times 1$ gradient $\partial \ell(W;Y)/\partial \eta$ is dense. The Hessian is computed in a similar fashion:

$$\begin{aligned}
\nabla\nabla_{WW^T}\ell(W;Y) &= \frac{\partial^2 \ell(W;Y)}{\partial W \partial W^T} \\
&= \frac{\partial}{\partial W^T} \cdot \left[\frac{\partial \eta}{\partial W} \cdot \frac{\partial \ell(W;Y)}{\partial \eta} \right] \\
&= \begin{pmatrix} A^T \\ X^T \end{pmatrix} \cdot \frac{\partial^2 \ell(W;Y)}{\partial \eta \partial \eta^T} \cdot \begin{pmatrix} A & X \end{pmatrix} \\
&= \begin{pmatrix} A^T \frac{\partial^2 \ell(W;Y)}{\partial \eta \partial \eta^T} A & A^T \frac{\partial^2 \ell(W;Y)}{\partial \eta \partial \eta^T} X \\ X^T \frac{\partial^2 \ell(W;Y)}{\partial \eta \partial \eta^T} A & X^T \frac{\partial^2 \ell(W;Y)}{\partial \eta \partial \eta^T} X \end{pmatrix} \\
&\equiv \begin{pmatrix} A^T C A & A^T C X \\ X^T C A & X^T C X \end{pmatrix}
\end{aligned} \tag{5}$$

The $N \times N$ matrix

$$C = \frac{\partial^2 \ell(W;Y)}{\partial \eta \partial \eta^T} \tag{6}$$

is the Hessian of the log-likelihood with respect to the linear predictor η, and is the **nonzero part** of the "c" matrix used in INLA and the "C" matrix used in the case crossover paper. I will denote it by C from now on. In most models it is highly sparse, and hence (5) is highly sparse.

In certain models, such as the partial likelihood for Cox's proportional hazards regression, C is dense. Its dimension is equal to the sample size. This means that storing and computing this matrix becomes prohibitively expensive for moderate sized samples.

One strategy for overcoming this challenge is to note that the elements of C, $C_{ij} = \frac{\partial^2 \ell(W;Y)}{\partial \eta_i \partial \eta_j}$, only enter (5) through sums of a specific form. Let a_i and x_i be the i^{th} columns of A and X. Then:

$$\left[A^T \frac{\partial^2 \ell(W;Y)}{\partial \eta \partial \eta^T} A \right]_{ij} = a_i^T C a_j = \sum_{k=1}^{N} \sum_{l=1}^{N} A_{ik} A_{jl} C_{kl} \tag{7}$$

and

$$\left[X^T \frac{\partial^2 \ell(W;Y)}{\partial \eta \partial \eta^T} X\right]_{ij} = x_i^T C x_j = \sum_{k=1}^{N} \sum_{l=1}^{N} X_{ik} X_{jl} C_{kl} \tag{8}$$

The sum in (7) has a small number of non-zero terms, and it is known which ones they are, due to the sparsity of A. Sums involving the crossproduct terms with A and X are also sparse (though less so) because of this.

The sum in (8) has all terms non-zero if C is dense. To overcome this, note that it has the form of a scaled average:

$$\sum_{k=1}^{N} \sum_{l=1}^{N} X_{ik} X_{jl} C_{kl} = N^2 \times \left[\frac{1}{N^2} \sum_{k=1}^{N} \sum_{l=1}^{N} X_{ik} X_{jl} C_{kl}\right] \tag{9}$$

Hence it can be *estimated*:

$$\sum_{k=1}^{N} \sum_{l=1}^{N} X_{ik} X_{jl} C_{kl} \approx N^2 \times \left[\frac{1}{|S|^2} \sum_{k,l \in S} X_{ik} X_{jl} C_{kl}\right] \tag{10}$$

for some sub-sample S with $|S| \ll N$. This is the idea behind *stochastic gradient descent*, and is commonly used in machine learning applications to perform large-scale optimizations where the objective function takes the form of a sum over a massive number of datapoints. It works because due to the weak law of large numbers, (9) and (10) are close with high probability.