Stochastic Simulation: Lecture 13

Christoph Reisinger

Oxford University Mathematical Institute

Modified from earlier slides by Prof. Mike Giles.

Objectives

The stochastic optimisation problem we consider is to determine $\theta \in \mathbb{R}^d$ that minimises

$$\mathbb{E}[f(\theta, X)].$$

In statistics and machine learning, this may correspond to maximising the log-likelihood given a large set of data:

$$-\text{log-likelihood} \;=\; \sum_{i=1}^{S} f_i(\theta) \;=\; \mathbb{E}[S\, f_I(\theta)]$$

where the expectation comes from taking a random index I, uniformly distributed over $\{1, 2, \ldots, S\}$.

Challenges

There are two main computational challenges:

1. the dimension d of θ is large: use gradient descent (as opposed to higher order methods, which require manipulations of the $(d \times d)$ Hessian) and adjoint differentiation (back propagation) to compute the gradient;

2. the number of samples S is large: use a (small) random sample of data to estimate gradient in each iteration.

Steepest descent

The classic steepest descent method for solving $\nabla f(\theta) = 0$ is based on a time-discretisation of

$$\dot{\theta} = -\nabla f(\theta)$$

which gives

$$\theta_{n+1} = \theta_n - \alpha \nabla f(\theta_n).$$

From this we get

$$\theta_{n+1} - \theta_n \approx (I - \alpha J)(\theta_n - \theta_{n-1})$$

where J is the Hessian at θ_{n-1}.

So it converges to the root θ^* from near θ^* if $\|I - \alpha J\| < 1$.

Robbins–Munro

Starting from

$$\theta_{n+1} = \theta_n - \alpha\, \mathbb{E}[\nabla f(\theta_n, X)]$$

the idea of Robbins & Munro was to replace the expectation by a single sample to give

$$\Theta_{n+1} = \Theta_n - \alpha_n\, \nabla f(\Theta_n, X_n)$$

with independent samples X_n. Here, we use capital Θ_n to indicate that it is random, and allow for varying step sizes α_n.

If we write $g(\theta) \equiv \mathbb{E}[\nabla f(\theta, X)]$ then we can write this as

$$\Theta_{n+1} = \Theta_n - \alpha_n\, g(\Theta_n) - \alpha_n\, (\nabla f(\Theta_n, X_n) - g(\Theta_n))$$

Robbins–Munro

Consider now the SDE

$$\mathrm{d}\widetilde{\Theta}_t = -g(\widetilde{\Theta}_t)\,\mathrm{d}t + \sigma(\widetilde{\Theta}_t)\,\mathrm{d}W_t$$

which has discretisation with timestep α_n

$$\widehat{\Theta}_{n+1} = \widehat{\Theta}_n - \alpha_n\, g(\widehat{\Theta}_n) + \sigma_n \sqrt{\alpha_n}\, Z_n$$

Equating this (approximately) to

$$\Theta_{n+1} = \Theta_n - \alpha_n\, g(\Theta_n) - \alpha_n\,(\nabla f(\Theta_n, X_n) - g(\Theta_n))$$

gives

$$\sigma_n^2 \;\approx\; \alpha_n\, \mathbb{V}[f(\Theta_n, X_n)]$$

Conclusion? For convergence we need $\displaystyle\sum_n^{\infty} \alpha_n \to \infty, \quad \alpha_n \to 0.$

Robbins–Munro

Usually, the second condition is tightened to $\sum_{n}^{\infty} \alpha_n^2 < \infty$.

A frequent choice is $\alpha_n = a/n$.

After running the iteration for N steps, the output of the Robbins–Munro algorithm is the final value θ_N.

Polyak and Ruppert independently improved this by using an average for the output

$$\overline{\Theta}_N \equiv N^{-1} \sum_{1}^{N} \Theta_n$$

– the averaging cancels out a lot of the noise in Θ_n

(Batch) Stochastic Gradient Descent

For a fixed sample of finite size S, we write

$$f(\theta) := \frac{1}{S} \sum_{i=1}^{S} f_i(\theta) \quad \to \quad \min_{\theta}.$$

- We also emphasise again that Θ_n are random;
- write $\mathcal{I}_n$ for the set of (randomly, uniformly) selected indices at iteration n;

Then the batch gradient iteration is

$$\Theta_{n+1} = \Theta_n - \alpha_n G_n, \qquad G_n = \frac{1}{|\mathcal{I}_n|} \sum_{i \in \mathcal{I}_n} \nabla f_i(\Theta_n).$$

Reduces the variance and provides scope for parallelisation or vectorisation.

SGD basic properties

We give a few basic properties: Let $\mathcal{F}_n$ be generated by Θ_n, and $\mathbb{E}_n := \mathbb{E}[\cdot|\mathcal{F}_n]$. Assume for simplicity $|\mathcal{I}_n| = 1$.

Then:

- $\mathbb{E}_n[G_n] = \nabla f(\Theta_n)$;

(A1) if f_i have Lipschitz gradients ∇f_i with Lipschitz constant L,

$$\mathbb{E}_n f(\Theta_{n+1}) \leq f(\Theta_n) - \alpha_n \nabla f(\Theta_n)^\top \mathbb{E}_n G_n + \frac{L\alpha_n^2}{2} \mathbb{E}_n \|G_n\|^2;$$

(A2) if, moreover, $\mathbb{V}_k[G_k] := \mathbb{E}_k[G_k^2] - \mathbb{E}_k[G_k]^2 \leq M$ for some $M > 0$, then

$$\mathbb{E}_n f(\Theta_{n+1}) \leq f(\Theta_n) - \alpha_n(1 - \alpha_n L/2)\|\nabla f(\Theta_n)\|^2 + \frac{ML\alpha_n^2}{2}.$$

SGD convergence of gradient

Let $f \geq f_{\text{min}} > -\infty$, and let (A1) and (A2) be satisfied, $\alpha_n = \alpha$, $\Theta_0 = \theta_0$, $\alpha \leq 1/L$. Then

$$\min_{1 \leq j \leq n} \mathbb{E}[\|\nabla f(\Theta_j)\|^2] \leq \alpha L M + \frac{2(f(\theta_0) - f_{\text{min}})}{\alpha n}.$$

Remarks:

- Need $2(f(\theta_0) - f_{\text{min}})/(\epsilon\alpha)$ iterations to get $\mathbb{E}\|\nabla \ldots\|^2 \leq \epsilon + \alpha L M$.
- Under stronger assumptions, can drop 'min' above, and show convergence of $\mathbb{E}\|\nabla \ldots\|^2$ to norm below $\alpha L M$.

\+ guaranteed bound

\- no convergence due to noise

SGD convergence – convex case

Assume additionally that for all $\eta \in [0,1]$, θ_1, θ_2,

$$f(\eta\theta_1 + (1-\eta)\theta_2) \leq \eta f(\theta_1) + (1-\eta) f(\theta_2) - \frac{\gamma}{2}\eta(1-\eta)\|\theta_1 - \theta_2\|^2.$$

Then for the unique minimiser $\theta^\star$,

$$\mathbb{E}[f(\Theta_n)] - f(\theta^\star) - \frac{\alpha L M}{2\gamma} \leq (1-\alpha\gamma)^n \left(\mathbb{E}[f(\theta_0)] - f(\theta^\star) - \frac{\alpha L M}{2\gamma} \right).$$

+ faster (linear) decay to 'noise floor'
- no convergence due to noise

SGD – learning rate schedules

Under the assumptions from the convex case, let

$$\alpha_n = \frac{2}{2L + \gamma n}.$$

Then

$$\mathbb{E}[f(\Theta_n)] - f(\theta^\star) \leq \frac{\max\left(f(\theta_0) - f(\theta^\star), \frac{M}{\gamma}\right)}{1 + \frac{\gamma}{2L} n}.$$

+ Convergence to minimum.
- Loss of linear convergence, even in convex case.

SGD – impact of batch size

In practice, choose $|\mathcal{I}_n| = m > 0$.
To simplify the analysis, we choose instead

$$G_n = \frac{1}{m} \sum_{i=1}^{m} \nabla f_{I_n^i}(X_n),$$

where I_n^i are drawn i.i.d. from $\{1, 2, \ldots, S\}$, i.e. with replacement.
Then if M is the bound for the single sample variance,

$$\mathbb{V}[G_n] \leq \frac{M}{m},$$

and we get

$$\mathbb{E}[f(\Theta_n)] - f(\theta^\star) - \frac{\alpha L M}{2\gamma m} \leq (1 - \alpha\gamma)^n \left(\mathbb{E}[f(\theta_0)] - f(\theta^\star) - \frac{\alpha L M}{2\gamma m} \right).$$

SGD – using control variates

Again, I_n^j are drawn i.i.d. from $\{1, 2, \ldots, S\}$, $j = 0, \ldots, m-1$.

Now consider $\Theta_n^0 = \Theta_n$ and then, for $j = 0, \ldots, m-1$:

$$\begin{aligned} G_n^j &= \nabla f(\Theta_n) + \nabla f_{I_n^j}(\Theta_n^j) - \nabla f_{I_n^j}(\Theta_n), \\ \Theta_n^{j+1} &= \Theta_n^j - \alpha_n^j G_n^j. \end{aligned}$$

Then set

1. $\Theta_{n+1} = \Theta_n^m$; or
2. $\Theta_{n+1} = \frac{1}{m} \sum_{j=1}^m \Theta_n^j$; or
3. $\Theta_{n+1} = \Theta_n^{J_n}$, where J_n is a uniform, independent sample of $\{1, \ldots, m\}$.

This is referred to as stochastic variance reduced gradient (SVRG).

\+ Faster convergence due to reduced variance.

\- Needs periodic evaluation of full gradient.

Convergence of SVRG

In addition to (A1) and (A2), assume all f_i convex. Moreover,

$$4\alpha L < 1, \qquad 1 < m\alpha\gamma(1 - 4\alpha L).$$

Then, for option 3. from the previous slide,

$$\mathbb{E}[f(\Theta_n)] - f(\theta^\star) \leq \rho^n \left(f(\theta_0) - f(\theta^\star)\right),$$

where

$$\rho = \frac{1 + 2m\alpha\gamma L}{m\alpha\gamma(1 - 2\alpha L)} < 1.$$

R. Johnson and T. Zhang, Accelerating stochastic gradient descent using predictive variance reduction, in NIPS 26, 2013, pp. 315?323.

SAGA

Inspired by stochastic average gradient descent (SAG) and SVRG, SAGA avoids evaluation of the full gradient after the first iteration.

Let N_n^j be the latest time prior to n that the gradient of f_j was computed.

- $G_0 = \nabla f(\Theta_0)$; $N_0^j = 0$
- For random uniform I_n, let

$$\begin{aligned} G_n &= \nabla f_{I_n}(\Theta_n) + \frac{1}{S}\sum_{j=1}^{S} \nabla f_j(\Theta_{N_n^j}) - \nabla f_{I_n}(\Theta_{N_n^{I_n}}), \\ \Theta_{n+1} &= \Theta_n - \alpha_n G_n. \end{aligned}$$

- Then set $N_{n+1}^{I_n} = n$ and $N_{n+1}^j = N_n^j$ for $j \neq I_n$.

A. Defazio, F. Bach, and S. Lacoste-Julien, SAGA: A fast incremental gradient methodwith support for non-strongly convex composite objectives, in NIPS 27, 2014, pp. 1646–1654

Final words

- Stochastic gradient descent is good for fitting high-dimensional parametric models for large sample sizes.
- Convergence requires a suitable learning rate schedule.
- Careful choice of mini-batch sizes and variance reduction can help.
- Challenges in practice include non-convexity and lack of a priori knowledge of constants in assumptions.

Key references

H. Robbins, S. Monro. "A Stochastic Approximation Method". The Annals of Mathematical Statistics. 22(3):400, 1951

B.T. Polyak, A.B. Juditsky. "Acceleration of Stochastic Approximation by Averaging". SIAM Journal on Control and Optimization. 30(4):838, 1992

D.P. Bertsekas, Nonlinear Programming, Athena Scientific, Belmont, Massachusetts, 1995.

S.J. Wright. "Optimization Algorithms for Data Analysis". http://www.optimization-online.org/DB FILE/2016/12/5748.pdf

L. Bottou, F.E. Curtis, J. Nocedal, "Optimization Methods for Large-Scale Machine Learning", SIAM Review, 60(2), pp 223–311, 2018.