ADAM (Adaptive Moments) I

- The update rule (Kingma and Ba, 2015)

$$
\begin{aligned}
\boldsymbol{g} &\leftarrow \frac{\boldsymbol{\theta}}{C} + \frac{1}{|S|}\nabla_{\boldsymbol{\theta}} \sum_{i:i\in S} \xi(\boldsymbol{\theta}; \boldsymbol{y}^i, Z^{1,i}) \\
\boldsymbol{s} &\leftarrow \rho_1 \boldsymbol{s} + (1-\rho_1)\boldsymbol{g} \\
\boldsymbol{r} &\leftarrow \rho_2 \boldsymbol{r} + (1-\rho_2)\boldsymbol{g} \odot \boldsymbol{g} \\
\hat{\boldsymbol{s}} &\leftarrow \frac{\boldsymbol{s}}{1-\rho_1^t} \\
\hat{\boldsymbol{r}} &\leftarrow \frac{\boldsymbol{r}}{1-\rho_2^t} \\
\boldsymbol{\theta} &\leftarrow \boldsymbol{\theta} - \frac{\epsilon}{\sqrt{\hat{\boldsymbol{r}}} + \delta} \odot \hat{\boldsymbol{s}}
\end{aligned}
$$

ADAM (Adaptive Moments) II

- t is the current iteration index
- Roughly speaking, ADAM is the combination of
  - Momentum
  - RMSprop
- From Goodfellow et al. (2016),

$$\frac{\epsilon}{\sqrt{\hat{\boldsymbol{r}}} + \delta} \odot \hat{\boldsymbol{s}}$$

(i.e., the use of momentum combined with rescaling) "does not have a clear theoretical motivation"

ADAM (Adaptive Moments) III

- The two steps

$$\hat{\boldsymbol{s}} \leftarrow \frac{\boldsymbol{s}}{1-\rho_1^t}$$
$$\hat{\boldsymbol{r}} \leftarrow \frac{\boldsymbol{r}}{1-\rho_2^t}$$

  are called "bias correction"
- Why "bias correction"?

ADAM (Adaptive Moments) IV

- They hope that

$$E[\boldsymbol{s}_t] = E[\boldsymbol{g}_t]$$

and

$$E[\boldsymbol{r}_t] = E[\boldsymbol{g}_t \odot \boldsymbol{g}_t],$$

where t is the iteration index

ADAM (Adaptive Moments) V

- For $\boldsymbol{s}_t$, we have

$$\begin{aligned} \boldsymbol{s}_t &= \rho_1 \boldsymbol{s}_{t-1} + (1-\rho_1)\boldsymbol{g}_t \\ &= \rho_1(\rho_1 \boldsymbol{s}_{t-2} + (1-\rho_1)\boldsymbol{g}_{t-1}) + (1-\rho_1)\boldsymbol{g}_t \\ &= (1-\rho_1)\sum_{i=1}^{t} \rho_1^{t-i} \boldsymbol{g}_i \end{aligned}$$

We assume that $\boldsymbol{s}$ is initialized as 0

ADAM (Adaptive Moments) VI

- Then

$$
\begin{aligned}
E[\boldsymbol{s}_t] &= E[(1-\rho_1)\sum_{i=1}^{t} \rho_1^{t-i} \boldsymbol{g}_i] \\
&= E[\boldsymbol{g}_t](1-\rho_1)\sum_{i=1}^{t} \rho_1^{t-i}
\end{aligned}
$$

- Note that we assume

$$E[\boldsymbol{g}_i], \forall i \geq 1$$

are the same

ADAM (Adaptive Moments) VII

- Next,

$$
\begin{aligned}
&(1-\rho_1)\sum_{i=1}^{t}\rho_1^{t-i} \\
=&(1-\rho_1)(1+\cdots+\rho_1^{t-1}) \\
=&1-\rho_1^t
\end{aligned}
$$

- Thus

$$E[\boldsymbol{s}_t] = E[\boldsymbol{g}_t](1-\rho_1^t)$$

and they do

$$\hat{\boldsymbol{s}} \leftarrow \frac{\boldsymbol{s}}{1-\rho_1^t}$$

ADAM (Adaptive Moments) VIII

- The above derivation on bias correction partially follows from https://towardsdatascience.com/adam-latest-trends-in-deep-learning-optimiz
- The situation for $E[\boldsymbol{g}_t \odot \boldsymbol{g}_t]$ is similar
- How about ADAM's practical performance?
- From Goodfellow et al. (2016), "generally regarded as being fairly robust to the choice of hyperparmeters, though the learning rate may need to be changed from the default"

ADAM (Adaptive Moments) IX

- However, from the web page we referred to for deriving the bias correction, "The original paper ... showing huge performance gains in terms of speed of training. However, after a while people started noticing, that in some cases Adam actually finds worse solution than stochastic gradient"
- One example of showing the above is Wilson et al. (2017)

Choosing Stochastic Gradient Algorithms

- From Goodfellow et al. (2016), "there is currently no consensus"
- Further, "the choice ... seemed to depend on the user's familiarity with the algorithm"

Why Stochastic Gradient Widely Used? I

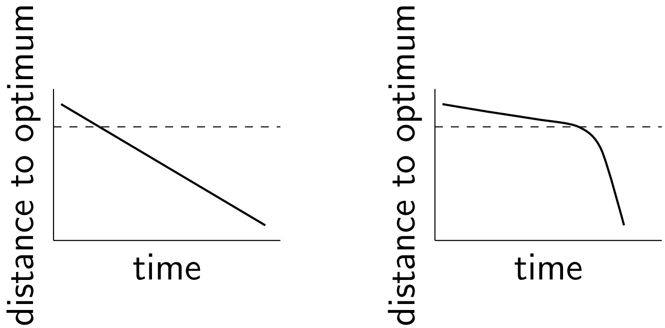
- In machine learning fast final convergence may not be important
  - An optimal solution $\boldsymbol{\theta}^*$ may not lead to the best model
  - Further, we don't need a point close to $\boldsymbol{\theta}^*$. In prediction we find

$$\arg\max_k z_k^{L+1}(\boldsymbol{\theta})$$

  A not-so-accurate $\boldsymbol{\theta}$ may be good enough

An illustration

Why Stochastic Gradient Widely Used? II



Slow final convergence Fast final convergence

Why Stochastic Gradient Widely Used? III

- The special property of data classification is essential

$$E(\nabla_{\boldsymbol{\theta}} \xi(\boldsymbol{z}^{L+1}; \boldsymbol{x}, \boldsymbol{y})) = \frac{1}{l} \nabla_{\boldsymbol{\theta}} \sum_{i=1}^{l} \xi(\boldsymbol{z}^{L+1,i}(\boldsymbol{\theta}); \boldsymbol{x}^i, \boldsymbol{y}^i)$$

- We can cheaply get a good approximation of the gradient
- Indeed stochastic gradient is less used outside machine learning

Why Stochastic Gradient Widely Used? IV

- Easy implementation. It's simpler than methods using, for example, second derivative

  Now for complicated networks, (subsampled) gradient is calculated by automatic differentiation
- We will explain more about this
- Non-convexity plays a role
  - For convex, other methods may possess advantages to more efficiently find the global minimum
  - But for non-convex, efficiency to reach a stationary point is less useful

Why Stochastic Gradient Widely Used? V

  - A global minimum usually gives a good model (as loss is minimized), but for a stationary point we are less sure
- Some variants of SG have been proposed to improve the robustness or the convergence
- All these explain why SG is popular for deep learning

References I

I. J. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. The MIT Press, 2016.

D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In *Proceedings of International Conference on Learning Representations (ICLR)*, 2015.

A. C. Wilson, R. Roelofs, M. Stern, N. Srebro, and B. Recht. The marginal value of adaptive gradient methods in machine learning. In *Advances in Neural Information Processing Systems*, pages 4148–4158, 2017.