Adam: A Method for Stochastic Optimization

Jun Tao

Penn State University

Dec. 2018

Motivation: SGD

Stochastic gradient descent (SGD) randomly selects a subset of the data to calculate the gradient. It is efficient comparing to naive gradient descent. However, it performs frequent updates with a high variance that causes the convergence path to fluctuate heavily.

The toy example on the right side shows how SGD behaves in simple linear regression, where the optimal function is quadratic. Within 600 iterations, SGD progresses slowly in the neighborhood of the optimum.

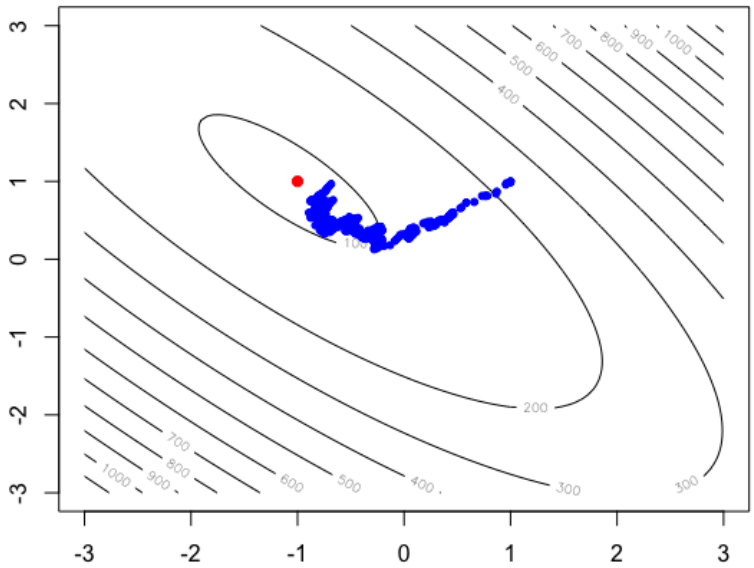

Figure 1: Toy example of SGD

Momentum Method

The momentum method (Polyak, 1964), or classical momentum (CM), is a technique for accelerating SGD that accumulates a velocity vector in directions of gradient across iterations. Polyak showed that CM can considerably accelerate convergence to a local minimum.

$$v_{t+1} = \beta v_t + (1-\beta)\nabla_\theta f(\theta_t);$$
$$\theta_{t+1} = \theta_t - \alpha v_{t+1},$$

where α is the learning rate and β is the momentum coefficient. They are both hyperparameters.

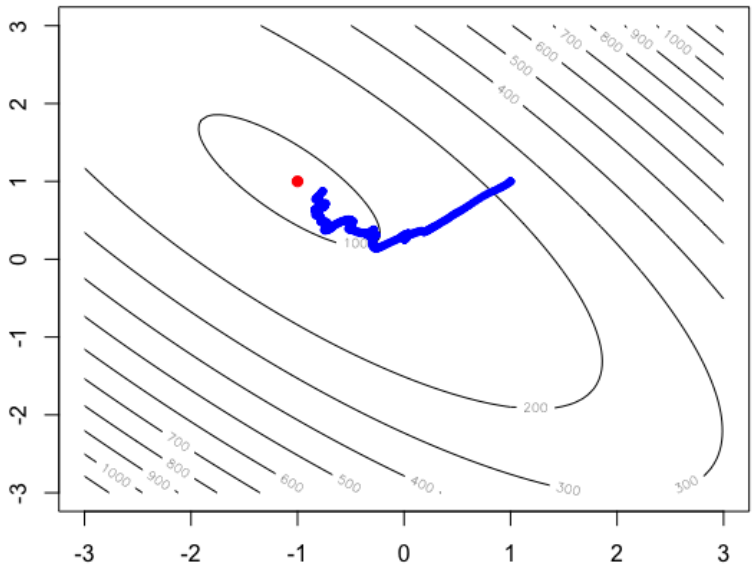

Figure 2: Toy example of CM

RMSProp

Root Mean Square Propogation (RMSprop) is an unpublished, adaptive learning rate method proposed by Geoff Hinton. It has been successfully employed in several practical applications but there's no theoretical guarantee of global convergence.

RMSprop divides the learning rate by an exponentially decaying average of squared gradients.

$$S_{t+1} = \beta S_t + (1-\beta)(\nabla_\theta f(\theta_t)).^2;$$
$$\theta_{t+1} = \theta_t - \alpha \nabla_\theta f(\theta_t) / \sqrt{\epsilon + S_{t+1}},$$

where α is the learning rate and β is weight hyperparameter. ϵ is a small quantity to make sure the denominator is nonzero.

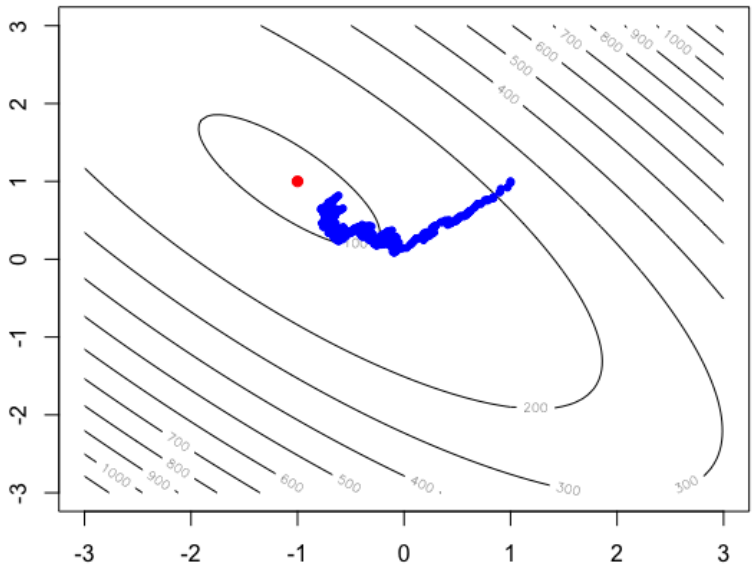

Figure 3: Toy example of RMSprop

Adam

Adam is a combination of CM and RMSprop with additional de-biasing process. The name Adam is derived from adaptive moment estimation.

Algorithm 1 Adam

1: Set proper initial values, $\alpha = 0.001$, $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\epsilon = 10^{-8}$.
2: Require $f(\theta)$ and θ_0, the stochastic objective function and the starting point.
3: Initialize variables: $m_0 \leftarrow 0$, $v_0 \leftarrow 0$ and $t \leftarrow 0$.
4: **while** θ_t does not converge **do**
5: [Step 1] $t \leftarrow t + 1$;
6: [Step 2] $g_t \leftarrow \nabla_\theta f_t(\theta_{t-1})$;
7: [Step 3] $m_t \leftarrow \beta_1 \cdot m_{t-1} + (1 - \beta_1) \cdot g_t$;
8: [Step 4] $v_t \leftarrow \beta_2 \cdot v_{t-1} + (1 - \beta_2) \cdot g_t^2$;
9: [Step 5] $\alpha_t \leftarrow \alpha \cdot \sqrt{1 - \beta_2^t}/(1 - \beta_1^t)$;
10: [Step 6] $\theta_t \leftarrow \theta_{t-1} - \alpha_t \cdot m_t/(\sqrt{v_t} + \epsilon\sqrt{1 - \beta_2^t})$;
11: **end while**; **return** θ_t

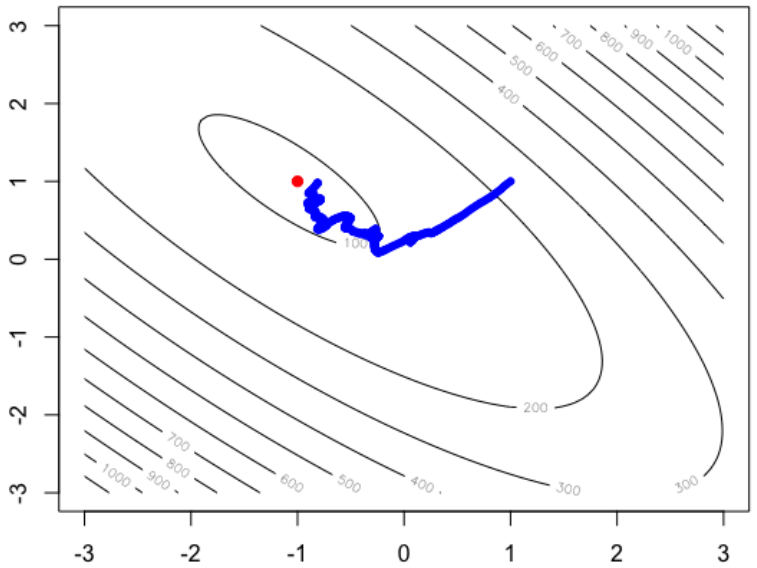

Figure 4: Toy example of Adam

Example: Logistic Regression

- In logistic regression setting, the goal is to maximize loglikelihood function $\ell(\boldsymbol{\beta}) = \sum_i \ell_i(\boldsymbol{\beta}) = \sum_i [y_i \eta_i - \log(1 + \exp(\eta_i))]$.
- Newton-Raphson method requires the inverse of observed Fisher information for $\boldsymbol{\beta}$, which can be difficult to obtain when p is large.
- For simulation, choose $p = 19$, $x_i \sim N(0, .7I_p + .3\mathbf{1}_p\mathbf{1}_p^T)$, $\beta \sim U(-1, 1)^{p+1}$. Training data size $= 1000$ and test size $= 250$.

	SGD	CM	Adam	N-R
time (s)	0.23	0.35	0.08	0.01
training error (%)	15.83	15.83	22.71	16.04
test error (%)	30.83	30.83	27.50	30.83
$\left\|\hat{\beta} - \beta\right\|$	0.66	0.77	2.11	0.72

Example: Logistic Regression

- For simulation, choose $p = 99$, $x_i \sim N(0, .7I_p + .3\mathbf{1}_p\mathbf{1}_p^T)$, $\beta \sim U(-1, 1)^{p+1}$. Training data size = 1000 and test size = 250.

	SGD	CM	Adam	N-R
time (s)	1.26	1.27	0.57	0.15
training error (%)	1.04	1.25	4.17	0*
test error (%)	13.33	12.50	11.70	16.70
$\left\|\hat{\beta} - \beta\right\|$	3.18	3.26	3.04	151.50

*: I used "glm" function in R. Actually N-R fails to work in this case. The warning message said: the algorithm does not converge.

Real Data

- MNIST is database of handwritten digits. It is a subset of a larger set available from NIST. The digits have been size-normalized and centered in a fixed-size image.
- I applied Adam to distinguishing 0 from 6.

	SGD	CM	Adam	N-R
time (s)	1.58	1.05	0.30	203.90
training error (%)	2.34	2.31	3.25	NA*
test error (%)	2.37	2.79	3.25	NA

*: For MNIST data, identification of 0 and 6 has $n = 11841$ and $p = 28^2 = 784$. "glm" function can not return a proper answer.

Potential Problem of Adam

- Adam may miss the global optimum.
- Possible solution (Keskar, 2017): first use Adam to locate the rough area of the optimum, and then switch to SGD.