Comp6211e: Optimization for Machine Learning

Tong Zhang

Lecture 21: Stochastic Adaptive Learning Rate and Acceleration Methods

Stochastic Optimization in Machine Learning

In machine learning, we observe training data (x_i, y_i) for $i = 1, \ldots, n$, and would like to learn a model parameter w of the form

$$\min_{w \in C} \left[\frac{1}{n} \sum_{i=1}^{n} f_i(w) + g(w) \right].$$

More generally, we can write this optimization problem as:

$$\min_{w \in C} \phi(w), \quad \phi(w) = f(w) + g(w), \qquad f(w) = \mathbf{E}_{\xi \sim D} f(\xi, w), \tag{1}$$

where ξ is a random variable, drawn from a distribution D.

Gradient versus Stochastic Gradient

In gradient based methods, we use gradient

$$\nabla f(w) = \mathbf{E}_\xi \nabla f(\xi, w).$$

In SGD, we replace the full gradient with stochastic gradient

$$\nabla_w f(\xi, w^{(t-1)}),$$

or minibatch stochastic gradient:

$$\nabla f_B(w) = \frac{1}{|B|} \sum_{\xi \in B} \nabla_w f(\xi, w),$$

Adaptive Learning Rate

Consider a minibatch B, and let

$$f_B(w) = \frac{1}{|B|} \sum_{\xi \in B} f(\xi, w),$$

and let

$$V = \mathbf{E}_\xi \|\nabla_w f(\xi, w) - \nabla_w f(w)\|_2^2.$$

Now consider the SGD update rule

$$w' = w - \eta \nabla f_B(w),$$

How to determine the best learning rate η?

One-step Objective Reduction

$$
\begin{aligned}
\mathbf{E}_B f(w') =& \mathbf{E}_B f(w - \eta \nabla f_B(w)) \\
\leq& \mathbf{E}_B \left[f(w) - \eta \nabla f(w)^\top \nabla f_B(w) + \frac{\eta^2 L}{2} \mathbf{E}_B \|\nabla f_B(w)\|_2^2 \right] \\
=& f(w) - (\eta - 0.5\eta^2 L)\|\nabla f(w)\|_2^2 + \frac{\eta^2 L}{2m} V.
\end{aligned}
$$

If we choose

$$
\eta \leq \frac{\|\nabla f(w)\|_2^2}{L(\|\nabla f(w)\|_2^2 + V/m)},
$$

then we have convergence rate of

$$
\mathbf{E}_B f(w') = f(w) - 0.5\eta \|\nabla f(w)\|_2^2. \tag{2}
$$

Interpretation

If

$$V \le (n-m)\,\|\nabla f(w)\|_2^2, \tag{3}$$

then (2) holds as long as

$$\eta \le m/(nL).$$

Since each minibatch SGD requires m gradient computations, per sample error reduction as in (2) becomes

$$\frac{\eta}{m}\|\nabla f(w)\|_2^2 \ge \frac{\|\nabla f(w)\|_2^2}{nL},$$

which is the per sample function value reduction as GD with step size $\eta = 1/L$:

$$\frac{\eta}{n}\|\nabla f(w)\|_2^2 = \frac{\|\nabla f(w)\|_2^2}{nL}.$$

Generalized Armijo-Goldstein Condition

Proposition (Stochastic AG Criterion)

Consider the SGD update rule:

$$w' = w - \eta \nabla f_B(w).$$

Let

$$V_{B',B} = \left[f_{B'}(w - \eta \nabla f_B(w)) - \eta \nabla f_{B'}(w)^\top \nabla f_B(w) - f_{B'}(w) \right].$$

If for some $c < 1$,

$$\mathbf{E}_{B',B}\, V_{B',B} \leq c\eta \|\nabla f(w)\|_2^2,$$

then

$$\mathbf{E}_B f(w') \leq f(w) - (1 - c)\eta \|\nabla f(w)\|_2^2.$$

Proximal Case

When $g(w) \neq 0$, we may generalize the Proposition 1 by using gradient mapping as follows.

$$\begin{aligned}
\mathrm{prox}_{\eta g}(w) =& \arg\min_z \left[\frac{1}{2\eta} \|z - w\|_2^2 + g(z) \right] \\
D_\eta \phi(w) =& \frac{1}{\eta}(w - \mathrm{prox}_{\eta g}(w - \eta \nabla f(w))) \\
D_{\eta,B} \phi(w) =& \frac{1}{\eta}(w - \mathrm{prox}_{\eta g}(w - \eta \nabla f_B(w))).
\end{aligned}$$

With this modification, we obtain the adaptive learning rate method for stochastic gradient descent.

Stochastic Gradient with AG Learning Rate

Algorithm 1: Stochastic Proximal Gradient Descent with Adaptive Learning Rate

Input: $f(\cdot)$, $g(\cdot)$, w_0, η_0, p (default is $\lceil n/m \rceil$)
Output: $w^{(T)}$

```
Let η_1 = η_0
Let q = 0
Let V = 0
Let Randomly select a minibatch B_0 independent samples from D
Let g̃ = D_{η_0, B_0} φ(w_0)
for t = 1, 2, ..., T do
    Randomly select a minibatch B_t of m independent samples from D
    Let w^(t) = prox_{η_t g}(w^(t-1) − η_t ∇f_{B_t}(w^(t-1)))
    Let V = V + [f_{B_t}(w^(t) − η_t g̃) − f_{B_t}(w^(t)) − η_t ∇f_{B_t}(w^(t))^⊤ g̃]
    Let g̃ = (w^(t-1) − w^(t)) / η_t
    Let η_{t+1} = η_t
    Let q = q + 1
    if q >= p then
        Let ρ = min(0.5, max(2, q c η_t ||D_{η_t} φ(w_t)||_2^2 / V))
        Let η_{t+1} = η_t ρ
        Let q = 0
        Let V = 0
        Let g̃ = D_{η_{t+1}, B_t} φ(w_t)
```

Return: $w^{(T)}$

Stochastic Accelerated Gradient

Algorithm 2: Stochastic Accelerated Proximal Gradient Descent

Input: $f(\cdot)$, $g(\cdot)$, $\{\eta_t, \beta_t\}$
Output: $w^{(T)}$

for $t = 1, 2, \ldots, T$ **do**
  Let $z^{(t)} = w^{(t-1)} + \beta_t(w^{(t-1)} - w^{(t-2)})$
  Randomly select a minibatch B_t of m independent samples from D
  Let $w^{(t)} = \mathrm{prox}_{\eta_t g}(z^{(t)} - \eta_t \nabla f_{B_t}(z^{(t)}))$

Return: $w^{(T)}$

Acceleration with AG Learning Rate

Algorithm 3: Stochastic Accelerated Proximal Gradient with Adaptive Learning Rate

Input: $f(\cdot)$, $g(\cdot)$, w_0, η_0, β, p (default is $\lceil n/m \rceil$)
Output: $w^{(T)}$

Let $\eta_1 = \eta_0$
Let $q = 0$
Let $V = 0$
Let Randomly select a minibatch B_0 independent samples from D
Let $\tilde{g} = D_{\eta_0, B_0} \phi(w_0)$
for $t = 1, 2, \ldots, T$ **do**
    Let $z^{(t)} = w^{(t-1)} + \beta(w^{(t-1)} - w^{(t-2)})$
    Randomly select a minibatch B_t of m independent samples from D
    Let $w^{(t)} = \mathrm{prox}_{\eta_t g}(z^{(t)} - \eta_t \nabla f_{B_t}(z^{(t)}))$
    Let $V = V + [f_{B_t}(z^{(t)} - \eta_t \tilde{g}) - f_{B_t}(z^{(t)}) - \eta_t \nabla f_{B_t}(z^{(t)})^\top \tilde{g}]$
    Let $\tilde{g} = (z^{(t)} - w^{(t)})/\eta_t$
    Let $\eta_{t+1} = \eta_t$
    Let $q = q + 1$
    if $q >= p$ **then**
        Let $\rho = \max(0.5, \min(2, qc(1-\beta)\eta_t \|D_{\eta_t}\phi(w_t)\|_2^2 / V))$
        Let $\eta_{t+1} = \eta_t \rho$
        Let $q = 0$
        Let $V = 0$
        Let $\tilde{g} = D_{\eta_{t+1}, B_t} \phi(w_t)$

Return: $w^{(T)}$

Momentum (Heavy Ball) Method

Algorithm 4: Stochastic Heavy-Ball Gradient Descent

Input: $f(\cdot)$, $g(\cdot)$, $\{\eta_t, \beta_t\}$
Output: $w^{(T)}$

for $t = 1, 2, \ldots, T$ **do**
  Let $z^{(t)} = w^{(t-1)} + \beta_t(w^{(t-1)} - w^{(t-2)})$
  Randomly select a minibatch B_t of m independent samples from D
  Let $w^{(t)} = \mathrm{prox}_{\eta_t g}(z^{(t)} - \eta_t \nabla f_{B_t}(w^{(t-1)}))$

Return: $w^{(T)}$

Stochastic Accelerated RDA

Algorithm 5: Stochastic Accelerated Regularized Dual Averaging

Input: $f(\cdot)$, $g(\cdot)$, $w^{(0)}$, $\tilde{\eta}_0 \le \tilde{\eta}_1, \ldots$, $\theta_t)$
$h(w)$ (default is $h(w) = 0.5\|w\|_2^2$)

Output: $w^{(T)}$

Let $\tilde{\alpha}_0 \in \partial h(w^{(0)})$
Let $v^{(0)} = w^{(0)}$
for $t = 1, 2, \ldots, T$ **do**
    Let $u^{(t)} = (1 - \theta_t) w^{(t-1)} + \theta_t v^{(t-1)}$
    Randomly select a minibatch B_t of m independent samples from D
    Let $\tilde{\alpha}_t = (1 - \theta_t)\tilde{\alpha}_{t-1} - \theta_t \nabla f_B(u^{(t)})$
    Let $v^{(t)} = \arg\min_w \left[-\tilde{\alpha}_t^\top w + \tilde{\eta}_t^{-1} h(w) + g(w) \right]$
    Let $w^{(t)} = (1 - \theta_t) w^{(t-1)} + \theta_t v^{(t)}$

Return: $w^{(T)}$

Stochastic Accelerated Linearized ADMM

Algorithm 6: Stochastic Accelerated Linearized ADMM

Input: $\phi(\cdot)$, A, B, c, $\{\eta_t\}$, β, ρ, α_0, w_0, z_0
Output: w_T, z_T, α_T

Let $\bar{w}_0 = x_0$
Let $\bar{z}_0 = z_0$
for $t = 1, 2, \ldots, T$ **do**
  Let $\tilde{z}_t = \bar{z}_{t-1} - \eta_t B^\top[\alpha_{t-1} + \rho(A\bar{w}_{t-1} + Bz_{t-1} - c)]$
  Let $z_t = \arg\min_z [0.5\|z - \tilde{z}_t\|_2^2 + \eta_t g(z)]$
  Let $w_t = \bar{w}_{t-1} - \eta_t \nabla f_B(\bar{w}_{t-1}) - \eta_t A^\top[\alpha_{t-1} + \rho(A\bar{w}_{t-1} + Bz_t - c)]$
  Let $\alpha_t = \alpha_{t-1} + \rho(1 - \beta)[Aw_t + Bz_t - c]$
  Let $\bar{z}_t = z_t + \beta(z_t - z_{t-1})$
  Let $\bar{w}_t = w_t + \beta(w_t - w_{t-1})$

Return: w_T, z_T, α_T

Empirical Studies

We study the smoothed hinge loss function $\phi_\gamma(z)$ with $\gamma = 1$, and solves the following $L_1 - L_2$ regularization problem:

$$\min_w \left[\underbrace{\frac{1}{n} \sum_{i=1}^{n} \phi_\gamma(w^\top x_i y_i)}_{f(w)} + \underbrace{\frac{\lambda}{2} \|w\|_2^2 + \mu \|w\|_1}_{g(w)} \right].$$

We compare different algorithms with constant learning rate.

Comparisons (strongly convex)

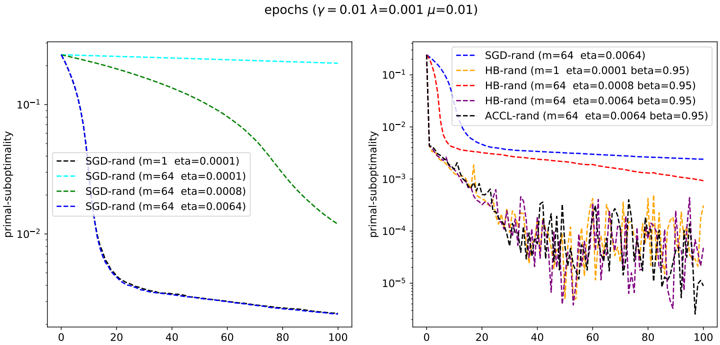


Figure: Comparisons of stochastic algorithms

Comparisons (strongly convex)

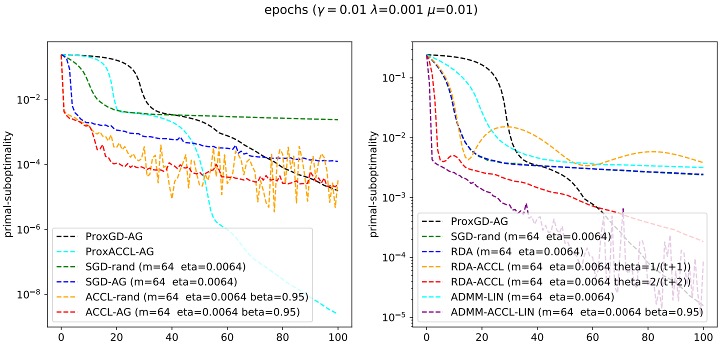
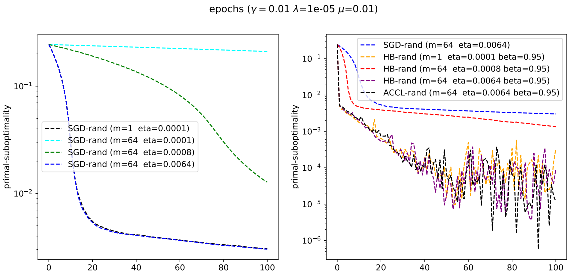
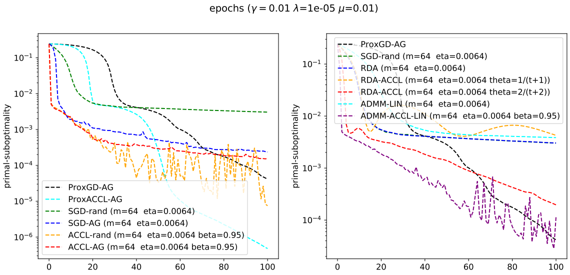


Figure: Comparisons of stochastic algorithms

Comparisons (near non-strongly convex)





Summary

SGD versus GD

- SGD converges faster when gradient is relatively large

$$\|D_\eta \phi(w)\|_2^2 \geq O(m/n) \times \text{minibatchvariance}.$$

- Eventually GD converges faster

Tuning of Learning Rate

- Can generalize AG criterion to the stochastic setting
- Requires estimating the variance on the fly.

Acceleration

- improve over non-accelerated in the beginning
- eventually variance dominate the convergence rate
- has an effect of using a larger learning rate