Comp6211e: Optimization for Machine Learning

Tong Zhang

Lecture 19: Stochastic Gradient Descent

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 Descent versus Stochastic Gradient Descent

In proximal gradient descent, we have

$$w^{(t)} = \mathrm{prox}_{\eta_t g}\left(w^{(t-1)} - \eta_t \nabla f(w^{(t-1)})\right),$$

where

$$\mathrm{prox}_{\eta g}(w) = \arg\min_z \left[\frac{1}{2\eta}\|z - w\|_2^2 + g(z)\right].$$

In SGD, we replace the full gradient

$$\nabla f(w^{(t-1)})$$

by stochastic gradient

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

with random $\xi \sim D$.

Stochastic Gradient Descent

Algorithm 1: Proximal Stochastic Gradient Descent (Proximal SGD)

Input: $\phi(\cdot)$, learning rates $\{\eta_t\}$, $w^{(0)}$
Output: $w^{(T)}$

for $t = 1, 2, \ldots, T$ **do**
  Randomly pick $\xi \sim D$
  Let $w^{(t)} = \mathrm{prox}_{\eta_t g}(w^{(t-1)} - \eta_t \nabla_w f(\xi, w^{(t-1)}))$

Return: $w^{(T)}$

Convergence

Theorem

Consider proximal SGD. If $f(w)$ is convex, and for all ξ and $w \in C$:

$$\|\nabla_w f(\xi, w)\|_2 \leq G,$$

and $g(w)$ is convex. We have for all $w \in C$:

$$\sum_{t=1}^{T} \eta_t \mathbf{E}\left[\phi(w^{(t)}) - \phi(w)\right] \leq 2G^2 \sum_{t=1}^{T} \eta_t^2 + \frac{1}{2}\|w - w^{(0)}\|_2^2.$$

Convergence

Theorem

Consider proximal SGD. If $f(w)$ is λ strongly convex, and $g(w)$ is λ' strongly convex. Let $\eta_t = t^{-1}/(\lambda + \lambda')$. We have for all $w \in C$:

$$\frac{1}{T}\sum_{t=1}^{T} \mathbf{E}\, \phi(w^{(t)}) \leq \phi(w) + 2G^2 \frac{\ln(T+1)}{(\lambda+\lambda')T} + \frac{\lambda'}{2T}\|w - w^{(0)}\|_2^2.$$

MiniBatch SGD

In practice, we need to work with a minibatch B of m training samples per iteration. If we let

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

then the minibatch SGD algorithm is presented in Algorithm 2. In this case, the gradient is

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

which requires m gradient evaluations, and it is an unbiased gradient estimator:

$$\mathbf{E}_B \nabla f_B(w) = \nabla f(w).$$

Minibatch SGD Algorithm

Algorithm 2: Proximal Minibatch Stochastic Gradient Descent (Proximal Minibatch SGD)

Input: $\phi(\cdot)$, learning rates $\{\eta_t\}$, $w^{(0)}$
Output: $w^{(T)}$

for $t = 1, 2, \ldots, T$ **do**
  Randomly pick a minibatch $B \sim D$ of size $|B| = m$
  Let $w^{(t)} = \mathrm{prox}_{\eta_t g}(w^{(t-1)} - \eta_t \nabla f_B(, w^{(t-1)}))$

Return: $w^{(T)}$

Example

Consider the regression problem:

$$\mathbf{E}_\xi f(\xi, w), \qquad f(\xi, w) = \frac{1}{2}(\nu(w, x) - y)^2 + \frac{\lambda}{2}\|w\|_2^2.$$

In this example, $g(w) = 0$ and $\mathrm{prox}_{\eta g}(w) = w$. Therefore we have

$$\begin{aligned} w^{(t)} =& w^{(t-1)} - \eta_t \left[\frac{1}{m} \sum_{\xi \in B} (\nu(w^{(t-1)}, x) - y) \nabla_w \nu(w^{(t-1)}, x) + \lambda w^{(t-1)} \right] \\ =& (1 - \eta_t \lambda) w^{(t-1)} - \frac{\eta_t}{m} \sum_{\xi \in B} (\nu(w^{(t-1)}, x) - y) \nabla_w \nu(w^{(t-1)}, x). \end{aligned}$$

Example

Consider the regression problem:

$$\mathbf{E}_\xi f(\xi, w) + g(w), \qquad g(w) = \frac{\lambda}{2}\|w\|_2^2 \qquad \text{subject to} \;\; w \in C.$$

We assume that $g(w)$ and C are convex. The proximal gradient becomes

$$w^{(t)} = \mathrm{proj}_C \left((1 - \tilde{\eta}_t \lambda) w^{(t-1)} - \tilde{\eta}_t \frac{1}{m} \sum_{\xi \in B} \nabla_w f(\xi, w^{(t-1)}) \right),$$

where $\tilde{\eta}_t = \eta_t / (1 + \eta_t \lambda)$.

Convergence

Theorem

Consider minibatch SGD. If $f(w)$ is convex and L smooth, $g(w)$ is convex. Let

$$V = \sup_{w \in C} \mathbf{E}_{\xi \sim D} \|\nabla f(\xi, w) - \nabla f(w)\|_2^2.$$

If we choose $\eta_t < 1/L$ for all t, then for all $w \in C$:

$$\sum_{t=1}^{T} \eta_t \mathbf{E} \left[\phi(w^{(t)}) - \phi(w)\right] \leq \sum_{t=1}^{T} \frac{\eta_t^2 V}{2(1 - \eta_t L) m} + \frac{1}{2}\|w - w^{(0)}\|_2^2.$$

Interpretation

If we take $\eta_t = \eta\sqrt{m/T}$, then

$$\frac{1}{T}\sum_{t=1}^{T} \mathbf{E}\left[\phi(w^{(t)}) - \phi(w)\right] \leq \frac{\eta\ V}{2(\sqrt{mT} - \eta m L)} + \frac{1}{2\eta\sqrt{mT}}\|w - w^{(0)}\|_2^2.$$

With the learning rate increased by a factor of $\sqrt{m}$. The number of samples needed to achieve accuracy ϵ is:

$$mT = O(V^2/\epsilon^2).$$

Convergence

Theorem

Consider minibatch SGD. If $f(w)$ is λ strongly convex and L smooth, $g(w)$ is λ' strongly convex. Let $\eta_t = 1/(2L + 0.5(t-1)(\lambda+\lambda'))$, and

$$V = \sup_{w \in C} V(w).$$

We have

$$\begin{aligned}
&\sum_{t=1}^{T}(2L - \lambda + 0.5(t-1)(\lambda+\lambda'))\mathbf{E}\left[\phi(w^{(t)}) - \phi(w)\right] \\
\leq & \frac{2TV}{m} + L(2L+\lambda')\|w - w^{(0)}\|_2^2.
\end{aligned}$$

Analysis: Key Idea

Consider a minibatch B, and define for $\eta > 0$,

$$Q_{\eta,B}(w; w') = f(w') + \nabla f_B(w')^\top (w - w') + \frac{1}{2\eta}\|w - w'\|_2^2 + g(w).$$

SGD optimizes the above objective at each step.
We have

Proposition

Assume that $f(w)$ is L-smooth in C. If we pick $\eta < 1/L$, then given any w', we have

$$\phi(w) \leq Q_{\eta,B}(w; w') + \frac{\eta}{2(1 - \eta L)}\|\nabla f_B(w') - \nabla f(w')\|_2^2.$$

Similar to the deterministic case, but with an extra variance term.

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

Comparisons (smooth and strongly convex)

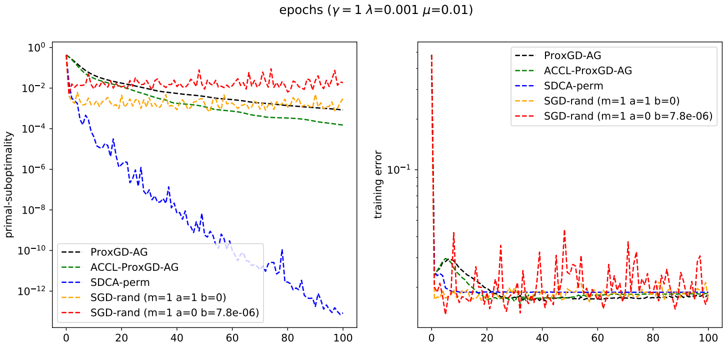


Figure: Comparisons of SGD to minibatch SGD with different learning rates

$$\eta_t = \eta/(1 + a\sqrt{t} + bt).$$

Comparisons ($m = 1$ versus $m > 1$)

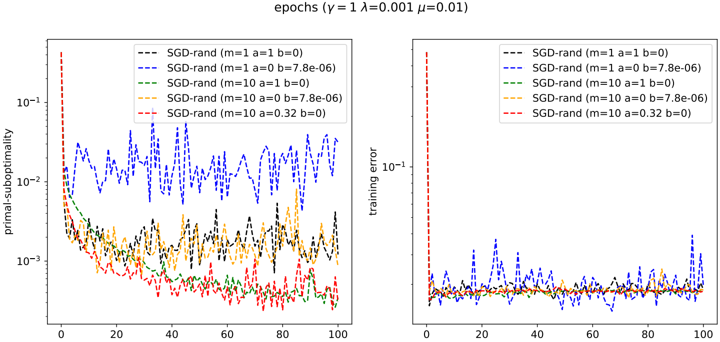
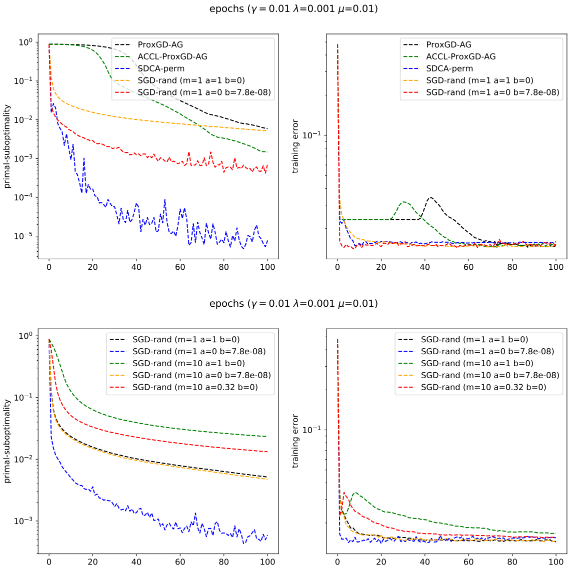


Figure: Comparisons of Proximal Gradient, SDCA and primal CD, SPDC

$$\eta_t = \eta/(1 + a\sqrt{t} + bt).$$

Comparisons (near non-smooth and strongly convex)



Summary

Finite Sum Optimization

- stochastic optimization

Stochastic gradient descent

- unbiased estimate of the full gradient
- less computation per iteration

Convergence

- can be obtained for different cases.
- different learning rate schedule, which may depend on the minibatch size