Comp6211e: Optimization for Machine Learning

Tong Zhang

Lecture 10: Adaptive Learning Rate and Lower Bounds

Convex Optimization

In this lecture, we consider the general unconstrained convex optimization problem:

$$\min_{x \in \mathbb{R}^d} f(x).$$

In first order methods, we can set learning rate as $1/L$, where L is the smoothness parameter.
However, if we do not know the smoothness parameter L of $f(x)$, then what to do?

Line Search for First Order Methods

In general first order methods, we are given a tentative solution y, and a search direction p.

We want to find a learning rate α so that the algorithm can converge fast.
A simple criterion is exact line search:

$$\min_{\alpha} f(y + \alpha p).$$

Inexact Line Search: Backtracking

Algorithm 1: Backtracking Line Search Method

Input: $f(x)$, y, p, α_0, $\tau \in (0, 1)$, $c \in (0, 1)$ (default is $c = 0.5$)

Output: α

Let $\alpha = \alpha_0$

while $f(y + \alpha p) > f(y) + c\alpha \nabla f(y)^\top p$ **do**

  $\alpha = \tau\alpha$

Return: α

Armijo-Goldstein condition

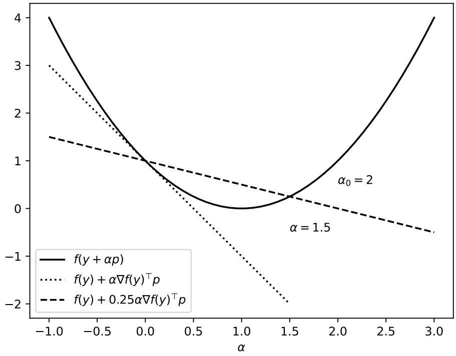


Figure: Illustration of Armijo-Goldstein condition

GD-AG

Algorithm 2: Subgradient Descent with AG Learning Rate

Input: $f(x)$, x_0, η_0 , $\tau = 0.8$, $c = 0.5$
Output: x_T

for $t = 1, \ldots, T$ **do**
    Let $x_t = x_{t-1} - \eta_{t-1} g_t$, where $g_t \in \partial f(x_{t-1})$ is a subgradient
    Let $\tilde{\eta} = (f(x_{t-1}) - f(x_t))/\|g_t\|_2^2$
    Let $\eta_t = \eta_{t-1}$
    while $\tilde{\eta} \leq c\eta_t$ *and* $\tilde{\eta} \geq 10^{-4}\alpha_0$ **do**
        Let $\eta_t = \tau\eta_t$
        Let $x_t = x_{t-1} - \eta_t g_t$
        Let $\tilde{\eta} = (f(x_{t-1}) - f(x_t))/\|g_t\|_2^2$
    if $\tilde{\eta} \geq \tau^{-0.5} c\eta_t$ **then**
        Let $\eta_t = \tau^{-0.5}\eta_t$

Return: x_T

AGD-AG

Algorithm 3: Adaptive Acceleration Method with AG Learning Rate

Input: $f(x)$, x_0, α_0, $\tau = 0.8$, $c = 0.5$
Output: x_T

Let $x_{-1} = x_0$
Let $\gamma = 0$
Let $y_0 = x_0$
for $t = 1, \ldots, T$ **do**
  Let $\beta = \min(1, \exp(\gamma))$
  Let $y_t = x_{t-1} + \beta(x_{t-1} - x_{t-2})$
  Let $x_t = y_t - \alpha_{t-1} \nabla f(y_t)$
  Let $\alpha_t = \alpha_{t-1}$
  Let $\tilde{\eta} = (f(x_t) - f(y_t))/\|\nabla f(y_t)\|_2^2$
  while $\tilde{\eta} \leq c\alpha_t$ *and* $\tilde{\eta} \geq 10^{-4}\alpha_0$ **do**
    Let $\alpha_t = \tau \alpha_t$
    Let $x_t = y_t - \alpha_t \nabla f(y_t)$
    Let $\tilde{\eta} = (f(y_t) - f(x_t))/\|\nabla f(y_t)\|_2^2$
  if $\tilde{\eta} \geq \tau^{-1} c\alpha_t$ **then**
    Let $\alpha_t = \tau^{-0.5} \alpha_t$
  Let $\gamma = 0.8\gamma + 0.2 \ln(\|\nabla f(y_t)\|_2^2 / \|\nabla f(y_{t-1})\|_2^2)$

Return: x_T

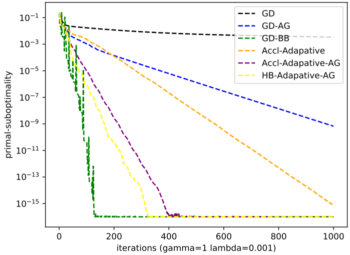
Empirical Study

We study the effect of smoothing for gradient descent and accelerated gradient methods for SVM. This is the same experiments as those in the last lecture.

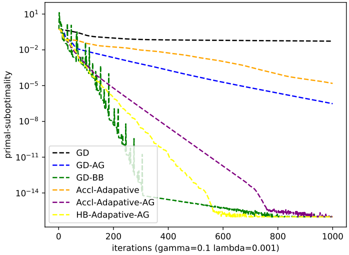
We use a smoothing of the hinge loss for SVM, where the hinge loss $(1 - z)_+$ is replaced by

$$\phi_\gamma(z) = \max_z \left[(1 - z)_+ + \frac{1}{2\gamma}(x - z)^2 \right].$$

Empirical Results

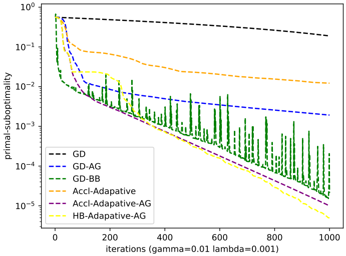


(a) $\gamma = 1$

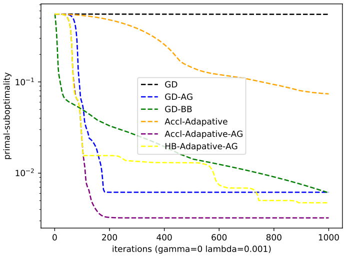


(b) $\gamma = 0.1$

Empirical Results



(a) $\gamma = 0.01$



(b) $\gamma = 0$

Barzilai-Borwein Step Size

Determine step size α along the line $y + \alpha p$.
For a smooth function $f(x)$:

$$(\nabla f(y + \alpha p) - \nabla f(y))^\top (\alpha p) \leq L \|\alpha p\|_2^2.$$

This implies that we can set

$$\frac{1}{L} \leq \frac{\|\alpha p\|_2^2}{(\nabla f(y + \alpha p) - \nabla f(y))^\top (\alpha p)}.$$

The largest learning rate is to set it equal to the right hand side, using estimate from previous iterations.

GD-BB

Algorithm 3: Subgradient Descent with BB Learning Rate

Input: $f(x)$, x_0, η_0 , $\tau = 0.8$, $c = 0.5$
Output: x_T

Let $g_0 \in \partial f(x_0)$ be a subgradient
for $t = 1, \ldots, T$ **do**
  Let $x_t = x_{t-1} - \eta_{t-1} g_t$
  Let $g_{t+1} \in \partial f(x_t)$ be a subgradient
  Let $\eta_t = \|x_t - x_{t-1}\|_2^2 / ((x_t - x_{t-1})^\top (g_{t+1} - g_t))$

Return: x_T

Lower Bounds

In general a first order algorithm evaluates gradients at a sequence points $(x_0, x_1, \ldots, x_t)$, with subgradient

$$g_0, g_1, \cdots, g_t,$$

where

$$g_s \in \partial f(x_s).$$

Therefore all first order optimization algorithms that start from $x_0 = 0$ satisfy

$$x_t \in \mathrm{span}\{g_s : s < t\}. \tag{1}$$

Strongly Convex Functions

Theorem

Given $L > \lambda > 0$ and $d \geq 2t \geq 2$. There exists an L-smooth and λ-strongly convex function $f(x)$, such that first order optimization algorithms can only produce solutions achieving convergence no better than:

$$f(x_t) - f(x_*) \geq \frac{\lambda}{2}\gamma^{2t}\frac{1}{1+\gamma^d}\|x_* - x_0\|_2^2,$$

where $\kappa = L/\lambda$, $\gamma = (\sqrt{\kappa} - 1)/(\sqrt{\kappa} + 1)$, and x_* is the optimal solution.

The theorem is meaningful when d is large.

Proof

For any $t \geq 1$ and $d \geq 2t$, we consider a d dimensional quadratic optimization problem, where

$$f(x) = \frac{L-\lambda}{4}\left(\frac{1}{2}x^\top Ax - e_1^\top x\right) + \frac{\lambda}{2}\|x\|_2^2.$$

Here e_1 denotes the vector of zeros, except the first coordinate being one. The matrix A is defined as

$$A = \begin{bmatrix} 2 & -1 & 0 & 0 & \ddots \\ -1 & 2 & -1 & 0 & \ddots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \ddots & 0 & -1 & 2 & -1 \\ \ddots & 0 & 0 & -1 & 2 - \frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1} \end{bmatrix}.$$

Proof

The optimal solution x_* of the problem is

$$[A + 4/(\kappa - 1)I]x_* = e_1.$$

It can be checked that $x_* = [x_{*,1}, \ldots, x_{*,d}]$ with $x_{*,j} = \gamma^j$ for $j = 1, \ldots, d$. Let $x_0 = 0$, and let $x_t = [x_{t,1}, \ldots, x_{t,d}]$. Since it is in the subspace spanned by $\{A^s e_1 : 0 \leq s < t\}$, we have $x_{t,j} = 0$ when $j \geq t + 1$.

$$\|x_* - x_t\|_2^2 \geq \gamma^{2(t+1)} \frac{1 - \gamma^{2(d-t)}}{1 - \gamma^2}.$$

and

$$\|x_* - x_t\|_2^2 \geq \gamma^{2t} \frac{1 - \gamma^{2(d-t)}}{1 - \gamma^{2d}} \|x_* - x_0\|_2^2.$$

Other Lower Bounds

Similarly, it can be shown that

- There exists a convex L-smooth objective function such that first order methods can do no better than

$$\min_{s \le t} f(x_s) - f(x_*) \geq = \Omega(L\|x_0 - x_*\|_2^2/t^2).$$

Summary

Automatic tuning learning rate is possible in practice

- Backtracking line search is a practical method
- BB method has different motivation, and works well.

Lower Bounds

- For Smooth problems: upper bounds of Nesterov's method are optimal in the high dimensional case.
- For Nonsmooth problems: without smoothing, subgradient methods are optimal.