Adaptive Learning Rate and Lower Bounds

1 Convex Optimization Problem

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

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

It was shown last time that for nonsmooth problems, learning rate should decay when t increases. Therefore it is useful to design methods that can tune learning rate automatically. We investigate such methods. We will then study lower bounds for convex optimization.

2 Automatic Estimation of Learning Rate

If we do not know the smoothness parameter L of $f(x)$, then we cannot set learning rate easily. In such case, we would like to estimate the best learning rate in gradient descent as well as in Nesterov's acceleration.

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 standard technique, which is used in the conjugate gradient method, is to do an exact line search that can minimize the objective function along a search direction. More generally, given the current point y and a search direction p, we want to find a learning rate α to minimize the objective

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

However, in practice, it is not practical to do an exact line search. Therefore we need to find other methods to set learning rate.

2.1 Armijo-Goldstein Step Size

One popular method to determine learning rate is to use the backtracking line search method in Algorithm 1 with a relatively large initial learning rate α_0. One may also employs a relatively small learning rate, and iteratively increase it by τ^{-1} until the condition $f(y + \alpha p) \leq f(y) + c\alpha \nabla f(y)^\top p$ is violated. The criterion used by the backtracking method, referred to as the *Armijo-Goldstein* condition [1], is illustrated in Figure 1.

If we take $p = -\nabla f(y)$ and $f(x)$ is L-smooth, then $c = 0.5$, then when $\alpha \leq 1/L$, we have

$$f(y + \alpha p) \leq f(y) + \alpha \nabla f(y)^\top p + \frac{L}{2} \alpha^2 \|p\|_2^2 \leq f(y) + \alpha(1 - 0.5\alpha L) \nabla f(y)^\top p.$$

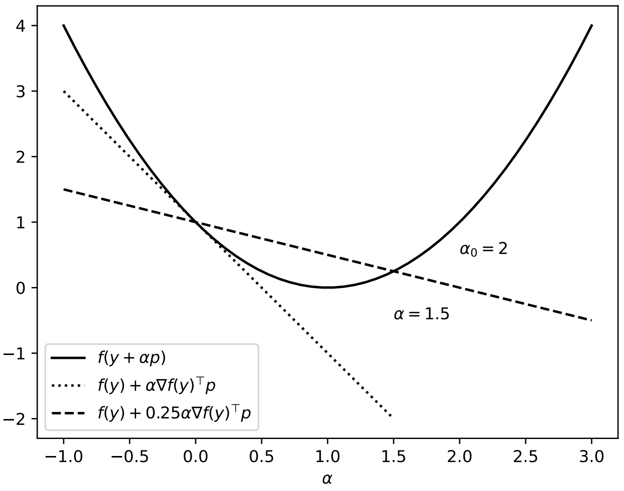


Figure 1: Illustration of Armijo-Goldstein condition

Therefore with $c = 0.5$, we have

$$f(y + \alpha p) \leq f(y) + c\alpha \nabla f(y)^\top p$$

as long as $\alpha \leq 1/L$. This allows us to search for the optimal learning rate around $1/L$.

Algorithm 1: Backtracking Line Search Method

```
Input: $f(x)$, $y$, $p$, $\alpha_0$, $\tau \in (0, 1)$, $c \in (0, 1)$ (default is $c = 0.5$)
Output: $\alpha$
Let $\alpha = \alpha_0$
while $f(y + \alpha p) > f(y) + c\alpha \nabla f(y)^\top p$ do
    $\alpha = \tau\alpha$
Return: $\alpha$
```

We may apply the backtracking line search method to determine the learning rate for gradient descent. There are various different implementations, and we describe one possible implementation in Algorithm 2. Moreover, similar idea can be applied to Nesterov's acceleration. We describe it in Algorithm 3.

Algorithm 2: Subgradient Descent with AG Learning Rate

```
Input: $f(x)$, $x_0$, $\eta_0$ , $\tau = 0.8$, $c = 0.5$
Output: $x_T$
for $t = 1, \dots, T$ 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: $x_T$
```

Algorithm 3: Adaptive Acceleration Method with AG Learning Rate

```
Input: $f(x)$, $x_0$, $\alpha_0$, $\tau = 0.8$, $c = 0.5$
Output: $x_T$
Let $x_{-1} = x_0$
Let $\gamma = 0$
Let $y_0 = x_0$
for $t = 1, \dots, T$ 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: $x_T$
```

2.2 Barzilai-Borwein Step Size

The backtracking method is popular, and simple to implement. However, it requires function value evaluation. In general, this can be avoided by only using gradient evaluations. For a smooth function $f(x)$, we have the following inequalities:

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

It follows that by summing the two inequalities that

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

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. For gradient descent method, we have Algorithm 4, which often works surprisingly well in practice. The original motivation of the BB criterion in [2] is to find the stepsize to match that of the Newton method. It is consistent with the derivation presented here.

Algorithm 4: Subgradient Descent with BB Learning Rate

```
Input: $f(x)$, $x_0$, $\eta_0$ , $\tau = 0.8$, $c = 0.5$
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: $x_T$
```

3 Empirical Studies

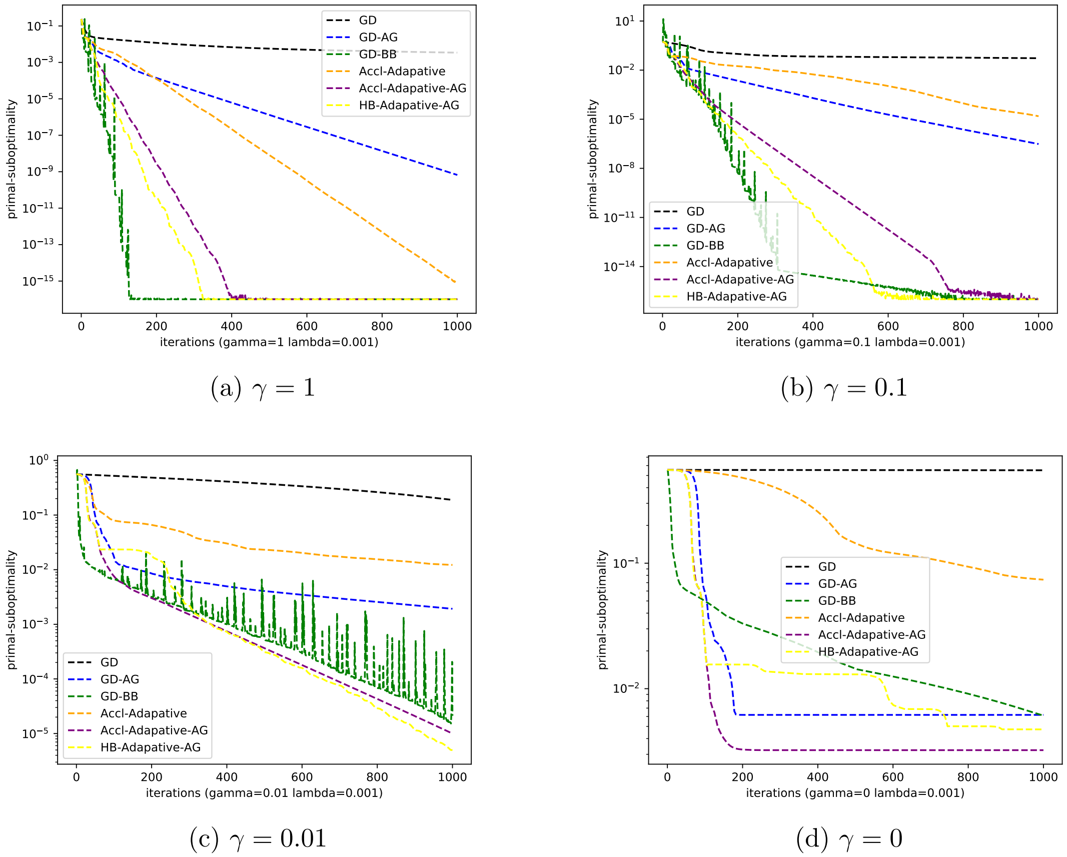
We study the same problems as those of last lecture. We use a smoothing of the hinge loss for SVM, where the hinge loss $(1-z)_+$ is replaced by

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

As we can see, even with a very small amount of smoothing or no smoothing, acceleration works well. However, the rate of convergence slows down with less smooth objective functions (that is, when γ is small). This is consistent with our theoretical results.

4 Lower Bounds

In order to show that these algorithms achieve best possible convergence, we will investigate lower bounds in this lecture. See [3].



(a) $\gamma = 1$

(b) $\gamma = 0.1$

(c) $\gamma = 0.01$

(d) $\gamma = 0$

Figure 2: Convergence Comparisons with Different Smoothing Parameter

In general a first order algorithm evaluates gradients at a sequence points $(x_0, x_1, \ldots, x_t)$, with subgradient $g_0, g_1, \ldots, g_t$, where $g_s \in \partial f(x_s)$. It employs this information to determine the next point x_{t+1} to evaluate the next subgradient.

If we take $x_0 = 0$, all algorithms described so far compute x_{t+1} as a linear combination of $\{g_s : s \leq t\}$.

Therefore we will investigate the best possible first order optimization algorithm that starts from $x_0 = 0$, and pick each point

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

The following bound shows that for smooth and strongly convex functions, the result for Nesterov's acceleration is best possible in the pessimistic case.

Theorem 1 *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.

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

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

Since the absolute values of eigenvalues of $A - 2I$ are no more than its infinity-norm, which is the maximum absolute row sum of the matrix, we know that the eigenvalues of $A - 2I$ are between $[-2, 2]$. Therefore the eigenvalues of A are in $[0, 4]$, which implies that the smoothness parameter of $f(x)$ is at most L, and the strong convexity parameter of $f(x)$ is at least λ.

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 $j \geq t + 1$. Therefore

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

On the other hand

$$\|x_* - x_0\|_2^2 = \sum_{j=1}^{d} x_{*,j}^2 = \gamma^2\frac{1-\gamma^{2d}}{1-\gamma^2}.$$

It follows that

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

This implies the result. ■

Similarly, it can be shown that

- There exists a convex L-smooth objective function such that first order methods can do no better than
$$\min_{s\leq t} f(x_s) - f(x_*) \geq= \Omega(L\|x_0 - x_*\|_2^2/t^2).$$

The lower bound for non-smooth optimization can be more subtle, because the technique of smoothing changes the objective function $f(x)$, which leads to a violation of (1). If we focus on methods that strictly follow (1), then the bounds for sub-gradient descent (without acceleration) are optimal. This also means that in practice, applying a small amount of smoothing can be important for nonsmooth optimization due to potentially faster convergence with acceleration.

References

[1] Larry Armijo. Minimization of functions having lipschitz continuous first partial derivatives. *Pacific J. Math*, 16(1):1–3, 1966.

[2] J. Barzilai and J.M. Borwein. Two-point step size gradient methods. *IMA Journal of Numerical Analysis*, 8(1):141–148, 1988.

[3] Yurii Nesterov. *Introductory Lectures on Convex Programming*. Springer, 2004.