DS 801: Advanced Optimization for Data Science **KAIST, Fall 2024**
Lecture #6: Proximal Gradient, Acceleration, and LASSO March 13, 2024
Lecturer: Dabeen Lee

1 Outline

In this lecture, we study

- proximal gradient descent,
- ISTA and FISTA for LASSO,
- accelerated proximal gradient descent

2 Proximal Gradient Descent and ISTA

We consider the following composite convex optimization problem.

$$\min_{x\in\mathbb{R}^d} \quad f(x) = g(x) + h(x)$$

where we assume that g is a smooth convex function and h is convex. For constrained smooth convex minimixation, we consider

$$\min_{x\in\mathcal{X}} \quad g(x) \quad = \quad \min_{x\in\mathbb{R}^d} \quad g(x) + I_{\mathcal{X}}(x)$$

where $I_{\mathcal{X}}(x)$ denotes the indicator function of the convex domain $\mathcal{X}$. For LASSO, we have

$$\min_{x\in\mathbb{R}^d} \quad \underbrace{\frac{1}{n}\|Ax-b\|_2^2}_{\text{smooth convex function } g(x)} \quad + \quad \underbrace{\lambda\|x\|_1}_{\text{convex function } h(x)} \quad .$$

For the constrained case, the associated prox operator is equivalent to the projection operator, i.e.,

$$\mathrm{prox}_{\eta I_{\mathcal{X}}(\cdot)}(x) = \operatorname*{argmin}_{u\in\mathbb{R}^d}\left\{\eta I_{\mathcal{X}}(u) + \frac{1}{2}\|x-u\|_2^2\right\} = \operatorname*{argmin}_{u\in\mathcal{X}}\left\{\frac{1}{2}\|x-u\|_2^2\right\} = \mathrm{proj}_{\mathcal{X}}(x).$$

For LASSO, we take $h(x) = \lambda\|x\|_1$ whose associated prox operator is given by

$$\mathrm{prox}_{\eta\lambda\|\cdot\|_1}(x) = \left(\underbrace{\max\left\{0, |x_i| - \eta\lambda\right\}}_{\text{shirinkage operator}} \cdot \mathrm{sign}(x_i)\right)_{i\in[d]}$$

The proximal gradient algorithm applies to this composite problem proceeds with the following update rule.

$$x_{t+1} = \mathrm{prox}_{\eta h}(x_t - \eta\nabla g(x_t)).$$

When f is smooth with parameter β, we set the step size $\eta = 1/\beta$ is in the smooth convex minimization setting. Proximal Gradient Descent applied to LASSO is referred to as Iterative Shrinkage-Thresholding Algorithm (ISTA).

Algorithm 1 Proximal Gradient Descent

```
Initialize $x_1 \in \mathbb{R}^d$.
for $t = 1, \ldots, T$ do
    Update $x_{t+1} = \mathrm{prox}_{\eta h}(x_t - (1/\beta)\nabla g(x_t))$ where $\beta$ is the smoothness parameter of $g$.
end for
Return $x_T$.
```

Theorem 6.1. *Let $f = g + h$ where g is a β-smooth convex function in the ℓ_2 norm and h is convex. Then x_{T+1} returned by Proximal Gradient Descent (Algorithm 1) satisfies*

$$f(x_{T+1}) - f(x^*) \leq \frac{\beta\|x_1 - x^*\|_2^2}{2}.$$

Furthermore, when g is strongly convex, we deduce the following convergence result.

Theorem 6.2. *Let $f = g + h$ where g is β-smooth and α-strongly convex in the ℓ_2 norm and h is convex. Then x_T returned by Proximal Gradient Descent (Algorithm 1) satisfies*

$$\|x_{T+1} - x^*\|_2^2 \leq \left(1 - \frac{\alpha}{\beta}\right)^T \|x_1 - x^*\|_2^2.$$

3 Nesterov's Acceleration and FISTA

We observed that proximal gradient descent achieves a convergence rate of $O(1/T)$, and therefore, ISTA solves LASSO with a convergence rate of $O(1/T)$. In fact, we may deduce a faster convergence rate based on Nesterov's acceleration. We mentioned that Nesterov's accelerated gradient descent guarantees a convergence rate of $O(1/T^2)$ for smooth convex minimization. We will show that an accelerated version of proximal gradient descent achives a rate of $O(1/T^2)$ for the composite convex minimization where g is smooth and convex.

The main idea behind Nesterov's acceleration is to use "momentum", so the algorithm is often called gradient descent with momentum. Recall that proximal gradient descent for minimizing $g + h$ where g is β-smooth and convex and h is convex follows the update rule of

$$x_{t+1} = \mathrm{prox}_{h/\beta}\left(x_t - \frac{1}{\beta}\nabla g(x_t)\right)$$

from a given point x_t. The idea of momentum is to incorporate the direction $x_t - x_{t-1}$ that we took when moving from x_{t-1} to x_t to obtain the next iterate x_{t+1}. Then x_{t+1} is determined by not only the previous iterate x_t but also x_{t-1} which is the one before x_t. Figure 6.1 illustrates how the idea of momentum applies. Instead of applying the gradient descent update to x_t, we move a bit further from x_t along the momentum direction that we took from x_{t-1} to x_t. Let $\gamma_t > 0$ be a weight, and

$$y_t = x_t + \gamma_t(x_t - x_{t-1}).$$

Then we apply the primal gradient descent update on y_t to obtain the next point x_{t+1}, as follows.

$$x_{t+1} = \mathrm{prox}_{h/\beta}\left(y_t - \frac{1}{\beta}\nabla g(y_t)\right).$$

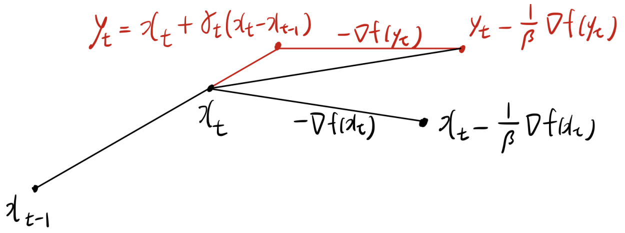
$y_t = x_t + \gamma_t(x_t - x_{t-1})$
$-\nabla f(y_t)$
$y_t - \frac{1}{\beta}\nabla f(y_t)$
x_t
$-\nabla f(x_t)$
$x_t - \frac{1}{\beta}\nabla f(x_t)$
x_{t-1}

Figure 6.1: Illustration of Gradient Descent Update with Momentum

Algorithm 2 Accelerated Proximal Gradient Descent

```
Initialize $x_1 \in \mathbb{R}^d$.
Set $x_0 = x_1$.
for $t = 1, \dots, T$ do
    $y_t = x_t + \gamma_t(x_t - x_{t-1})$ for some $\gamma_t > 0$.
    $x_{t+1} = \mathrm{prox}_{h/\beta}\left(y_t - \frac{1}{\beta}\nabla g(y_t)\right)$.
end for
Return $x_{T+1}$.
```

Algorithm 2 summarizes the accelerated version of proximal gradient descent that we just explained.

To provide a convergence result of the accelerated proximal gradient descent method, we need the following lemma.

Lemma 6.3. *Let $u, v \in \mathbb{R}^d$. Then for all $z \in \mathbb{R}^d$,*

$$\frac{1}{\eta}(\mathrm{prox}_{\eta h}(x) - x)^\top (z - \mathrm{prox}_{\eta h}(x)) + h(z) \geq h(\mathrm{prox}_{\eta h}(x)).$$

Proof. Note that

$$\mathrm{prox}_{\eta h}(x) = \operatorname*{argmin}_{z \in \mathbb{R}^d} \left\{ h(z) + \frac{1}{2\eta}\|x - z\|_2^2 \right\}.$$

By the optimality condition, it follows that for any $z \in \mathbb{R}^d$ and $g \in \partial h(\mathrm{prox}_{\eta h}(x))$,

$$\left(g + \frac{1}{\eta}\left(\mathrm{prox}_{\eta h}(x) - x\right)\right)^\top (z - \mathrm{prox}_{\eta h}(x)) \geq 0.$$

This implies that

$$\frac{1}{\eta}(\mathrm{prox}_{\eta h}(x) - x)^\top (z - \mathrm{prox}_{\eta h}(x)) + g^\top (z - \mathrm{prox}_{\eta h}(x)) \geq 0.$$

Here, since h is convex, we have

$$h(z) \geq h(\mathrm{prox}_{\eta h}(x)) + g^\top (z - \mathrm{prox}_{\eta h}(x)).$$

Adding the two inequalities, we prove the desired bound of this lemma. □

Theorem 6.4. *Let $f = g + h$ where g is a β-smooth convex function in the ℓ_2 norm and h is convex. We set η and γ_t as*

$$\eta = \frac{1}{\beta}, \quad \gamma_t = \frac{t-2}{t+1}.$$

Then x_{T+1} returned by Accelerated Proximal Gradient Descent (Algorithm 2) satisfies

$$f(x_{T+1}) - f(x^*) \leq \frac{2\beta}{(T+1)^2}\|x_1 - x^*\|_2^2$$

where x^* is an optimal solution to $\min_{x\in\mathbb{R}^d} f(x)$.

Proof. Note that Algorithm 2 is equivalent to

$$\begin{aligned} y_t &= (1-\lambda_t)x_t + \lambda_t v_t \\ x_{t+1} &= \mathrm{prox}_{h/\beta}\left(y_t - \frac{1}{\beta}\nabla g(y_t)\right) \\ v_{t+1} &= x_t + \frac{1}{\lambda_t}(x_{t+1} - x_t) \end{aligned}$$

where

$$\lambda_t = \frac{2}{t+1}.$$

This is because $y_t = x_t + \lambda_t(v_t - x_t)$ and

$$\lambda_t(v_t - x_t) = \lambda_t\left(\left(\frac{1}{\lambda_{t-1}} - 1\right)x_t + \left(1 - \frac{1}{\lambda_{t-1}}\right)x_{t-1}\right) = \frac{\lambda_t(1-\lambda_{t-1})}{\lambda_{t-1}}(x_t - x_{t-1}) = \gamma_t(x_t - x_{t-1}).$$

Moreover, we have $\lambda_1 = 1$, and for $t \geq 2$,

$$\frac{1-\lambda_t}{\lambda_t^2} \leq \frac{1}{\lambda_{t-1}^2}.$$

First, as g is β-smooth,

$$g(x_{t+1}) \leq g(y_t) + \nabla g(y_t)^\top(x_{t+1} - y_t) + \frac{\beta}{2}\|x_{t+1} - y_t\|_2^2.$$

Next, Lemma 6.3 implies that for any $z \in \mathbb{R}^d$,

$$\begin{aligned} h(x_{t+1}) &\leq h(z) + \beta\left(x_{t+1} - y_t + \frac{1}{\beta}\nabla g(y_t)\right)^\top (z - x_{t+1}) \\ &= h(z) + \nabla g(y_t)^\top(z - x_{t+1}) + \beta\left(x_{t+1} - y_t\right)^\top(z - x_{t+1}). \end{aligned}$$

Adding these two inequalities, we deduce that

$$\begin{aligned} f(x_{t+1}) &\leq h(z) + g(y_t) + \nabla g(y_t)^\top(z - y_t) + \beta\left(x_{t+1} - y_t\right)^\top(z - x_{t+1}) + \frac{\beta}{2}\|x_{t+1} - y_t\|_2^2 \\ &\leq f(z) + \beta\left(x_{t+1} - y_t\right)^\top(z - x_{t+1}) + \frac{\beta}{2}\|x_{t+1} - y_t\|_2^2 \end{aligned}$$

where the second inequality follows from convexity of g. By setting $z = x^*$ and $z = x_t$, we have

$$\begin{aligned} f(x_{t+1}) - f(x^*) &\leq \beta\left(x_{t+1} - y_t\right)^\top(x^* - x_{t+1}) + \frac{\beta}{2}\|x_{t+1} - y_t\|_2^2 \\ f(x_{t+1}) - f(x_t) &\leq \beta\left(x_{t+1} - y_t\right)^\top(x_t - x_{t+1}) + \frac{\beta}{2}\|x_{t+1} - y_t\|_2^2. \end{aligned}$$

Summing up the first inequality multiplied by λ_t and the second one multiplied by $(1-\lambda_t)$, we get

$$\begin{aligned}
&f(x_{t+1}) - f(x^*) - (1-\lambda_t)(f(x_t) - f(x^*))\\
&\leq \beta\left(x_{t+1} - y_t\right)^\top \left(\lambda_t x^* + (1-\lambda_t)x_t - x_{t+1}\right) + \frac{\beta}{2}\|x_{t+1} - y_t\|_2^2\\
&= \frac{\beta}{2}(x_{t+1} - y_t)^\top (2\lambda_t x^* + 2(1-\lambda_t)x_t - x_{t+1} - y_t)\\
&= \frac{\beta}{2}\|y_t - (1-\lambda_t)x_t - \lambda_t x^*\|_2^2 - \frac{\beta}{2}\|x_{t+1} - (1-\lambda_t)x_t - \lambda_t x^*\|_2^2\\
&= \frac{\beta\lambda_t^2}{2}\|v_t - x^*\|_2^2 - \frac{\beta\lambda_t^2}{2}\|v_{t+1} - x^*\|_2^2.
\end{aligned}$$

This implies that

$$\begin{aligned}
\frac{1}{\lambda_t^2}(f(x_{t+1}) - f(x^*)) + \frac{\beta}{2}\|v_{t+1} - x^*\|_2^2 &\leq \frac{1-\lambda_t}{\lambda_t^2}(f(x_t) - f(x^*)) + \frac{\beta}{2}\|v_t - x^*\|_2^2\\
&\leq \frac{1}{\lambda_{t-1}^2}(f(x_t) - f(x^*)) + \frac{\beta}{2}\|v_t - x^*\|_2^2\\
&\qquad\vdots\\
&\leq \frac{1}{\lambda_1^2}(f(x_2) - f(x^*)) + \frac{\beta}{2}\|v_2 - x^*\|_2^2\\
&\leq \frac{1-\lambda_1}{\lambda_1^2}(f(x_1) - f(x^*)) + \frac{\beta}{2}\|v_1 - x^*\|_2^2\\
&= \frac{\beta}{2}\|v_1 - x^*\|_2^2\\
&= \frac{\beta}{2}\|x_1 - x^*\|_2^2.
\end{aligned}$$

Therefore, it follows that

$$f(x_{T+1}) - f(x^*) \leq \frac{\beta\lambda_T^2}{2}\|x_1 - x^*\|_2^2 = \frac{2\beta}{(T+1)^2}\|x_1 - x^*\|_2^2,$$

as required. □

Hence, the convergence rate is $O(1/T^2)$, which matches the oracle lower bound. The number of required iterations to bound the error by ϵ is $O(1/\sqrt{\epsilon})$.

FISTA stands for Fast ISTA, that is an accelerated version of ISTA. Basically, FISTA is the accelerated proximal gradient descent method applied to LASSO. ISTA requires $O(1/\epsilon)$ iterations, while FISTA needs $O(1/\sqrt{\epsilon})$ iterations to converge to an ϵ-approximate solution.

We generated a random instance with 300 feature variables and 100 data samples. The figure compares the subgradient method, ISTA, and FISTA for the random LASSO instance. We can see that FISTA has the fastest rate of convergence while ISTA is also faster than the subgradient method.

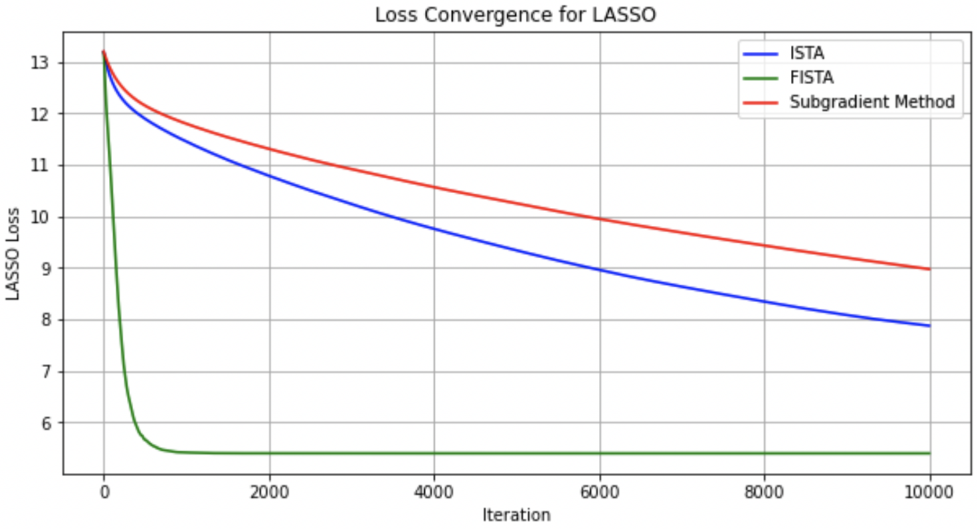

Figure 6.2: Comparing the subgradient method, ISTA, and FISTA for LASSO