**10-725: Convex Optimization** **Spring 2023**

# Lecture 4: January 26

*Lecturer: Siva Balakrishnan*

Today we'll pick up our discussion of GD, and particularly consider GD in the smooth, strongly convex case. We will then discuss the sub-gradient method, and its analysis. We'll then very briefly discuss lower bounds – this discussion will motivate accelarated first-order methods which we will likely cover in the second part of the course, as well as projection/proximal methods which we cover next.

## 4.1 GD in the Smooth and Strongly Convex Case

Recall, that in our last lecture we studied GD for (nice) quadratics, and saw that it has a very fast rate of convergence. This is more generally true of GD applied to $\beta$-smooth, $\alpha$-strongly convex functions. As before we will denote the *condition number* by,

$$\kappa = \frac{\beta}{\alpha}.$$

**Theorem 4.1** *Let $x^*$ denote the minimizer of $f$, then after $k$ iterations the GD iterate $x^k$ satisfies,*

$$\|x^k - x^*\|_2^2 \leq \left(1 - \frac{1}{\kappa}\right)^k \|x^0 - x^*\|_2^2.$$

1. As a consequence of smoothness (and the fact that $\nabla f(x^*) = 0$ we know that,

$$f(x^k) - f(x^*) \leq \frac{\beta}{2}\|x^k - x^*\|^2 \leq \frac{\beta}{2}\left(1 - \frac{1}{\kappa}\right)^{2k} \|x^0 - x^*\|_2^2.$$

   As with quadratics, to reach a point with $f(x^k) - f(x^*) \leq \epsilon$, ignoring $\beta, \kappa$ dependent constants roughly $\log(1/\epsilon)$ iterations suffice. This (linear) convergence is much faster than GD under just smoothness and convexity (i.e. without strong convexity).

**Proof:** We'll follow exactly the same proof as we had in the smooth case, except replacing our application of convexity with one of strong-convexity. As before, we first observe that,

$$\begin{aligned}\|x^t - x^*\|_2^2 &= \|x^{t-1} - \frac{1}{\beta}\nabla f(x^{t-1}) - x^*\|_2^2 \\ &= \|x^{t-1} - x^*\|_2^2 - \frac{2}{\beta}\nabla f(x^{t-1})^T(x^{t-1} - x^*) + \frac{1}{\beta^2}\|\nabla f(x^{t-1})\|_2^2.\end{aligned}$$

Exactly as in our proof without strong-convexity we note that by our descent lemma,

$$\frac{1}{\beta^2}\|\nabla f(x^{t-1})\|_2^2 \leq \frac{2}{\beta}(f(x^{t-1}) - f(x^t)).$$

Now, by strong convexity we can do a bit better on the cross-term. We see that,

$$f(x^*) \geq f(x^{t-1}) + \nabla f(x^{t-1})^T(x^* - x^{t-1}) + \frac{\alpha}{2}\|x^* - x^{t-1}\|_2^2,$$

re-arranging we obtain that,

$$-\frac{2}{\beta}\nabla f(x^{t-1})^T(x^{t-1} - x^*) \leq \frac{2}{\beta}\left[f(x^*) - f(x^{t-1}) - \frac{\alpha}{2}\|x^* - x^{t-1}\|_2^2\right].$$

Putting these pieces together, we see that,

$$\begin{aligned}\|x^t - x^*\|_2^2 &\leq \|x^{t-1} - x^*\|_2^2 + \frac{2}{\beta}\left[f(x^*) - f(x^t) - \frac{\alpha}{2}\|x^* - x^{t-1}\|_2^2\right] \\ &\leq \left(1 - \frac{\alpha}{\beta}\right)\|x^{t-1} - x^*\|_2^2.\end{aligned}$$

which directly implies our desired theorem. ■

One can slightly improve the above result (by being a bit more careful with the negative term we dropped at the end), but it requires a bit of work and the improvement is not substantial. You can see Bubeck's book, particularly Lemma 3.11, if you want to see this (slight) improvement worked out.

We have by now developed some understanding of GD, and how well it solves optimization problems where the function is $\beta$-smooth over an unconstrained domain. Our next goal will be to try to understand (unconstrained) optimization in the non-smooth setting, i.e. we'll no longer assume our function is differentiable and won't be able to rely on gradients any longer.

## 4.2 Subgradients

We've defined subgradients before. We'll stick to a convex function $f$ (although one can define subgradients more generally). For any $x \in \text{dom}(f)$ we'll say $g_x \in \partial f(x)$ if for all $y \in \text{dom}(f)$,

$$f(y) \geq f(x) + g_x^T(y - x).$$

1. For a convex $f$ subgradients exist everywhere except in some pathological examples, on the boundary of the domain of $f$.

2. When unique the subgradient is equal to the gradient (and the function is differentiable).

3. The collection of vectors $g_x$ which satisfy the above inequality form the subdifferential $\partial f(x)$.

### 4.2.1 Examples

Here are a couple of useful examples:

1. We have discussed this example before: $f(x) = |x|$. Here if $x \neq 0$, then the function is differentiable and $g_x = \text{sign}(x)$. At 0 it is not differentiable, but it is easy to check that any $g \in [-1, 1]$ satisfies the above inequality, so the subdifferential $\partial f(0) = [-1, 1]$.

   To denote this more conveniently, we can define the sign function:

$$\text{sign}(x) = \begin{cases} +1, & x > 0 \\ [-1, +1], & x = 0 \\ -1, & x < 0. \end{cases}$$

   Then we have that, $\partial f(x) = \text{sign}(x)$.

2. A slight generalization of this is: $f(x) = \|x\|_1$ where $x \in \mathbb{R}^d$. In this case, we just obtain (applying the same logic as above, elementwise) that $\partial f(x) = \text{sign}(x)$, where now we apply the sign function elementwise.

3. A more interesting example is when we consider $f(x) = \|x\|_2$. When $x \neq 0$ we can find the gradient directly and see that $\nabla f(x) = x/\|x\|_2$.

   The function is not differentiable at $x = 0$, so we need to check which vectors $g_0$ satisfy the condition that,

$$\|y\| \geq g_0^T y,$$

   for every $y \in \mathbb{R}^d$. As a consequence of the Cauchy-Schwarz inequality, any $g_0$ with $\|g_0\|_2 \leq 1$ satisfies this condition, and therefore is in the subdifferential at 0.

4. An even more interesting example is to consider the function $f(x) = \mathbb{I}_C(x)$ the indicator function for a convex set. Now it turns out that for any $x \in C$,

$$\partial f(x) = N_C(x),$$

   i.e. the subdifferential of the indicator function is the same as the normal cone.

To see this, fix a point $x \in C$ and observe that if $g_x \in \partial f(x)$ then we must have that,

$$f(y) \geq f(x) + g_x^T(y - x) = g_x^T(y - x).$$

Now, there are two possibilities: if $y \notin C$ then the above condition is trivially satisfied (since the LHS is $\infty$), so the only interesting possibility is when $y \in C$. The vector $g_x$ must thus satisfy,

$$g_x^T(y - x) \leq 0, \quad \text{for all } y \in C,$$

which is the same as requiring that $g_x \in N_C(x)$. Conversely, any vector in the normal cone is a valid subgradient via similar reasoning.

### 4.2.2 Some basic subgradient calculus

In many ways subgradients behave just like gradients provided you interpret "set valued" operations correctly. Here are a couple of facts (you will explore a couple more on your HW):

1. Scaling: $\partial(af) = a \times \partial f(x)$, when $a > 0$.
2. Sums: $\partial(f_1 + \ldots + f_m) = \partial f_1 + \ldots + \partial f_m$.

### 4.2.3 Using Subgradients to Derive Constrained, Non-Smooth Optimality Conditions

We have already seen one way to derive the optimality conditions for non-smooth constrained optimization. Now we will see a different way, reducing it to an unconstrained problem where we know the optimality conditions.

Observe that,

$$\min_{x \in C} f(x)$$

is equivalent to the unconstrained problem:

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

So $x^*$ is optimal for this program iff $0 \in \partial(f(x^*) + \mathbb{I}_C(x^*))$, i.e. $0 \in \partial f(x^*) + \partial I_C(x^*) = \partial f(x^*) + N_C(x^*)$, which is exactly the condition we derived before.

### 4.2.4 LASSO Optimality Conditions

Now, we'll briefly look at the optimality conditions for the LASSO program, and try to see another example where optimality conditions are a useful lens for understanding an optimization problem.

The LASSO program is simply least squares with an $\ell_1$ penalty, i.e. we solve:

$$x^* = \arg\min_{x \in \mathbb{R}^d} \frac{1}{2}\|b - Ax\|_2^2 + \lambda\|x\|_1.$$

Now, by our optimality conditions we know that any solution must have zero subgradient, i.e. it must be the case that,

$$0 \in -A^T(b - Ax^*) + \lambda \text{sign}(x^*).$$

Coordinatewise this gives us the condition that,

$$A_j^T(b - Ax^*) \in \lambda \text{sign}(x_j^*).$$

One intuitive fact that one can glean from this is that at optimum if $x_j^* = 0$ then we know that, $|A_j^T(b - Ax^*)| \leq \lambda$, i.e. roughly the 0s of the LASSO solution will correspond to variables $A_j$ which have a small correlation with the residuals $(b - Ax^*)$. You can with a bit more effort glean many other nice factoids about the LASSO solution from these conditions – but unfortunately (unlike ridge, or ordinary least squares) one cannot solve them in closed form to solve the LASSO.

If you take 36-709 with me then you'll likely see a nice proof that the LASSO selects the right variables (under some set of assumptions), and the starting point of that proof is just the above optimality conditions.

### 4.2.5 Soft Thresholding

A closely related optimization problem to the LASSO, but one where we can in fact find the optimal solution in closed form using the optimality conditions is the following program:

$$x^* = \arg\min_{x} \frac{1}{2}\|y - x\|_2^2 + \lambda\|x\|_1.$$

Now, the optimality conditions tell us that $x^*$ is optimal iff

$$0 \in -(y - x^*) + \lambda \text{sign}(x^*),$$

or equivalently,

$$(y - x^*) \in \lambda \text{sign}(x^*).$$

Lets define the soft-thresholding operation for a scalar $y_i$, and $\lambda > 0$,

$$S_\lambda(y_i) = \begin{cases} y_i - \lambda & \text{if } y_i > \lambda \\ 0 & \text{if } -\lambda \leq y_i \leq \lambda \\ y_i + \lambda & \text{if } y_i < -\lambda. \end{cases}$$

Now, if you stared at the above optimality conditions you would see that,

$$x^* = S_\lambda(y),$$

where we apply soft-thresholding element-wise satisfies the optimality conditions (i.e. is an optimal solution). It's not difficult to convince yourself that since the objective is strictly convex it is also the unique optimal solution.

This idea that we can efficiently, in closed-form, solve this simplified optimization problem will be useful to us when we discuss proximal methods.