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

Lecture 13: February 28th

Lecturer: Siva Balakrishnan

In today's lecture (the final one from me (!!)) we'll discuss a few of the main ideas related to some non-convex problems where we know how to guarantee (sometimes local) convergence to a global optimum. These settings are still pretty far from things like optimization in deep learning, but they're already fairly interesting settings.

At a very high-level one should view both the results that we discuss as types of stability/robustness statements for GD, i.e. GD doesn't quite require convexity to be effective.

13.1 The Polyak-Lojasiewicz (PL) Condition

Suppose we have a β-smooth function f, that we're interested in minimizing (unconstrained). We'll assume that a minimizer x^* exists (need not be unique). This function need not be convex. Suppose it instead satisfies the so-called μ-PL inequality, i.e. for some $\mu > 0$,

$$\frac{1}{2}\|\nabla f(x)\|_2^2 \geq \mu(f(x) - f(x^*)).$$

The way to interpret this condition is that it is some weakening of strong convexity. Recall, that when we discussed the linear convergence of GD we highlighted the key property that strong convexity gave us, if we're far away from optimal then strong convexity will ensure that the gradient is large. The PL condition is a more direct statement of that desirable property (but highlights the crucial fact that this is all we need for linear rates, i.e. we do not need convexity itself).

The paper of Karimi-Nutini-Schmidt which re-popularized this condition and highlighted its usefulness also gives examples of some non-convex functions which satisfy the condition. Here is a lemma relating the PL condition to strong convexity:

Lemma 13.1 *An α-strongly convex function also satisfies the α-PL inequality.*

Proof: We know by strong convexity that for any y,

$$f(y) \geq f(x) + \nabla f(x)^T(y - x) + \frac{\alpha}{2}\|x - y\|_2^2.$$

We can now minimize both sides with respect to y to see that,

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

which is precisely the α-PL inequality. ■

The reason why the PL condition is useful is that it is sufficient (together with smoothness) to ensure linear convergence of GD. Here is a theorem:

Theorem 13.2 *Suppose f is β-smooth, and μ-PL, then GD iterates with $\eta = 1/\beta$ converge linearly, i.e.*

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

Proof: The proof is in some sense a bit simpler than the proof of convergence with strong convexity (since we've already distilled its essence into the PL inequality). Recall our main descent lemma (go back to Lecture 3 if you want to refresh your memory) which holds under smoothness for our choice of step-size,

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

Re-arranging we get,

$$f(x^{t+1}) - f(x^*) \leq \left(1 - \frac{\mu}{\beta}\right)(f(x^t) - f(x^*)).$$

This yields the claimed result. ■

1. This result is at the heart of many global convergence results in non-convex optimization (for instance, the analysis of GD for a randomly initialized neural network in the so-called NTK regime, or the analysis of the policy gradient method for LQR). The main convenience is that convexity is some type of global property of a function that can be hard to verify in some examples, whereas the PL condition is often easier to verify. In some sense it is a type of 2-point property, i.e. we only need to verify some condition on the function at the current iterate, relative to the optimal point, in order to ensure that we make progress in the current iteration.

2. In our analysis of GD under strong convexity and smoothness we showed linear convergence of the iterates (not just of their associated function values). It turns out that something similar is true for PL functions (one needs to be a bit careful since x^* is not unique, so the distance to the set of the solutions decreases linearly). This proof is a bit involved, but one can show that PL functions exhibit something called quadratic growth (again the Karimi-Nutini-Schmidt paper is a great resource), i.e. letting x^* denote the closest optimal solution to x we have,

   $$\frac{\mu}{2}\|x - x^*\|_2^2 \leq f(x) - f(x^*).$$

   This in turn allows us show linear convergence of the iterates.

13.2 Local convergence to a Global Optimum of GD

We won't have time to belabour this point, but it's also worth noticing, that our proofs for convergence of GD under strong-convexity and smoothness, or under PL and smoothness don't require these conditions to hold globally.

In many cases, one can simply argue that the conditions hold locally around some optimal point x^*, i.e. in some ball of radius r around x^* (say). Our guarantees ensure that if we satisfy these conditions just within the ball and initialize within the ball, the iterates will (or can be shown in most cases) to stay within this ball, and converge linearly to the optimal solution. This is easier to explain with a picture.

The key takeaway is, if you can show some nice properties (things like smoothness, strong-convexity/PL) hold locally around x^* in some region, then you can usually make some type of statement about the local convergence to x^* (i.e. provided you initialize your algorithm smartly it will converge to the global optimum x^*). We'll revisit this at the end of lecture when we talk about the EM algorithm briefly.

13.3 The Stability of GD

Suppose we're interested in optimizing a function which is α-strongly convex, and β-smooth, but we can't compute exact gradients of the function in question. Suppose instead that we have access to an oracle – like a first-order oracle – but one that doesn't give us an exact gradient but rather gives us a vector g_x at any query point which satisfies:

$$\|g_x - \nabla f(x)\|_2 \leq \mu_1 \|x - x^*\|_2 + \mu_2,$$

where $\mu_1, \mu_2 \geq 0$, and $\mu_1 < \alpha$. We'll see in a moment some examples where this will be a very natural oracle model. For now, notice that it allows your gradients to be fairly inaccurate when x is far from x^* (since the term on the RHS will be large).

Historically, such conditions were studied to prove guarantees for what are sometimes called inexact gradient methods. Conditions of this form also arise often in the analysis of SGD-type methods (since those can also be viewed as returning inexact gradients).

Here is a theorem:

Theorem 13.3 *Suppose f is α-strongly convex and β-smooth, and we run an inexact gradient descent algorithm, following g_{x^t} instead of the real gradient direction, with the step-size $\eta = 2/(\alpha + \beta)$ then our algorithm has linear convergence upto a ball of radius roughly μ_2. Concretely, denoting $0 < \kappa = 2(\alpha - \mu_1)/(\alpha + \beta) < 1$, we have that,*

$$\|x^k - x^*\|_2 \leq (1 - \kappa)^k \|x^0 - x^*\| + \frac{\mu_2}{1 - \kappa}.$$

1. This result is pretty magical, and highlights one of the key stability properties of GD. It shows that if we're trying to optimize a (very) nice function f, then we can follow fairly inaccurate "estimates" of the gradient and still have linear convergence. Of course, as we get closer to the optimum we need our gradient estimates to be more accurate but even if they're not we'll simply bounce around in a ball around the optimal point. This should remind you of what happened in our analysis of SGD.

2. This result is often useful in statistical optimization settings. For instance, me (and some friends) have a paper on robust estimation, where the basic idea is simply that we can solve some robust estimation problems, as long as we can robustly estimate the gradient of our loss/likelihood. This latter problem turns out to be much simpler (robustly estimating the gradient of some loss is simpler than robustly estimating some general parameter vector). We'll of course have some errors in estimating the gradient (we're doing it from samples, and in our case from corrupted samples) but the result above gives us some useful guarantees.

Proof: The proof is very similar to the proof of convergence of GD under smoothness and strong convexity, but now we track the extra errors we get by following g_{x^t} instead of the gradient direction. In our earlier proof we had shown the following basic fact, which yielded linear convergence:

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

We will use a slight improvement of this result (you can find this version in Bubeck's book – Theorem 3.12) which will improve slightly some dependencies:

$$\|x^t - \frac{2}{\alpha+\beta}\nabla f(x^t) - x^*\| \leq \left(\frac{\beta-\alpha}{\beta+\alpha}\right) \|x^t - x^*\|.$$

Now, from this we get for our inexact algorithm:

$$\begin{aligned}
\|x^{t+1} - x^*\| = \|x^t - \frac{2}{\alpha+\beta}g_{x^t} - x^*\| &\leq \|x^t - \frac{2}{\alpha+\beta}\nabla f(x^t) - x^*\| + \frac{2}{\alpha+\beta}\|g_{x^t} - \nabla f(x^t)\| \\
&\leq \left(\frac{\beta-\alpha}{\beta+\alpha}\right) \|x^t - x^*\| + \frac{2\mu_1}{\alpha+\beta}\|x - x^*\|_2 + \mu_2 \\
&\leq \left(1 - \frac{2\alpha - 2\mu_1}{\alpha+\beta}\right) \|x^t - x^*\| + \mu_2.
\end{aligned}$$

Unrolling this recursion, and upper bounding the geometric series for the μ_2 terms, we obtain the final result. ∎

13.4 Local Convergence of the EM Algorithm

This will be mostly a very high-level description of some ideas from another paper I (and some other friends) wrote many years ago. At some point, many years ago, we were interested in understanding the convergence of methods like the EM (Expectation-Maximimization) algorithm. At this point in time one of the most interesting developments was that of so-called "spectral methods" – basically these were method-of-moments based methods for estimating latent variable models that had strong guarantees (i.e. you could prove they worked in some settings). However, empirically they weren't great and most papers which proposed these methods would follow a two-stage procedure, where they first ran the spectral method, and then used that to initialize EM (often giving much better results). I wanted to try to study this further.

The EM algorithm itself requires some notation to describe but here is the rough idea (I've mentioned this briefly before in the context of minorize-maximize algorithms). We'd like to fit a statistical model, but we have some unobserved (latent) variables. At each step, using our current guess for the parameters θ^t we'll impute/guess the values of the latent variables, and then try to maximize the likelihood (this latter problem is usually concave, because we've filled in all the missing data).

Slightly more concretely, at each step we form a function $Q(\theta|\theta^t)$ (a function of θ formed using our current guess θ^t), and then we compute:

$$\theta^{t+1} = \arg\max_{\theta} Q(\theta|\theta^t).$$

The first step is usually called the E-step, and the second is called the M-step.

It seems a little difficult to analyze this algorithm (usually the likelihood we're trying to optimize is non-concave). We gave some local convergence guarantees for EM, roughly by viewing it as an inexact gradient algorithm. The way to think about this is to first suppose we're only doing gradient steps (instead of a full maximization) in the M-step, i.e. we iterate the following step:

$$\theta^{t+1} = \theta^t - \eta \nabla Q(\theta|\theta^t).$$

This seems equally challenging to analyze. The insight however, is that there is some oracle algorithm that we'd like to mimic. The oracle would be the algorithm which did the E-step with the true parameters (we don't know the true parameters so we can't do this), i.e. the oracle iteration is:

$$\widetilde{\theta}^{t+1} = \widetilde{\theta}^t - \eta \nabla Q(\theta|\theta^*).$$

This iteration (where we fill latent variables in using the true parameters) is easily seen to converge to θ^*. Now, the basic insight is that, if we can argue that our likelihood is nice (strongly-convex and smooth), and further show that,

$$\|\nabla Q(\theta|\theta^t) - \nabla Q(\theta|\theta^*)\|_2 \leq \mu_1 \|\theta - \theta^*\|_2 + \mu_2,$$

for some appropriate μ_1, μ_2 then we can argue that our (first-order approximation to) EM converges to something useful (just following the ideas of the previous section).

It turns out that after some pages of algebra, for some nice statistical models (mixture of Gaussians etc.) we can actually check that this condition holds locally (i.e. in some ball around the truth). This in turn motivates the two-stage procedure we described in the beginning (where we use something like the method of moments to get a reasonable initialization and then use EM after that).