CS289ML: Notes on convergence of gradient descent

Raghu Meka

1 Gradient descent

In class we discussed the following notions:

- $f : \mathbb{R}^d \to \mathbb{R}$ is convex if for all $x, y \in \mathbb{R}^d$ and $0 \leq \lambda \leq 1$,

$$f(\lambda x + (1-\lambda)y) \leq \lambda f(x) + (1-\lambda) f(y).$$

- A smooth function $f : \mathbb{R}^d \to \mathbb{R}$ is L-Lipschitz if for all $x, y \in \mathbb{R}^d$,

$$\|\nabla f(x) - \nabla f(y)\|_2 \leq L\|x - y\|_2.$$

Examples x^2, e^x are all convex. So is the loss function from *least squares regression* LSR $f(x) = (1/n)\sum_{j=1}^{n}(\langle a_j, x\rangle - b_j)^2$. For what L is this loss Lipschitz? Let A be the $n \times d$ matrix whose rows are the vectors a_j and let b be the n-dimensional vector consisting of the b_j's. Then, we can also write $f(x) = (1/n)\|Ax - b\|^2$. So that $\nabla f(x) = (2/n)A^T(Ax - b)$. Therefore, for $x, y \in \mathbb{R}^d$,

$$\|\nabla f(x) - \nabla f(y)\| = (2/n)\|A^T(Ax - b) - A^T(Ay - b)\| = (2/n)\|A^T A(x - y)\| \leq (2/n)\|A^T A\|_2 \cdot \|x - y\|,$$

where for a matrix B, the *spectral norm*, $\|B\|_2$ is defined as $\|B\|_2 = \max_u \|Bu\|/\|u\|$. Thus, the loss function in LSR is $(2/n)\|A^T A\|_2$ Lipschitz.

In general, while convexity is a strong constraint in practice, Lipschitz-ness is more common. Nevertheless, Lipschitz convex functions are a rich class of functions that cover many common instances in optimization. We next analyze gradient descent for Lipschitz convex functions. Throughout this note, gradient descent (GD) will refer to the following algorithm:

- Choose $x_0 \in \mathbb{R}^d$ and step-size $t > 0$.
- For $i = 0, \ldots,$ define

$$x_{i+1} = x_i - t\nabla f(x_i).$$

1.1 Analysis of Gradient Descent

We first need some elementary properties of Lipschitz convex functions; proofs of the claims can be found at the end. The following claim captures the fact that the tangent plane of a convex function lies below the curve:

Claim 1.1. *For $f : \mathbb{R}^d \to \mathbb{R}$ convex, for all x, y,*

$$f(x) + \langle \nabla f(x), y - x\rangle \leq f(y).$$

The next claim complements the above property for Lipschitz convex functions.

Claim 1.2. *If $f : \mathbb{R}^d \to \mathbb{R}$ is L-Lipschitz convex, then for all x, y,*

$$f(y) \leq f(x) + \langle \nabla f(x), y - x \rangle + \frac{L}{2}\|y - x\|_2^2.$$

We also need the following property of convex functions.

Claim 1.3. *For any convex function $f : \mathbb{R}^d \to \mathbb{R}$, and $y_1, \ldots, y_k \in \mathbb{R}^d$,*

$$f\left(\frac{y_1 + \cdots + y_k}{k}\right) \leq \frac{f(y_1) + \cdots + f(y_k)}{k}.$$

We next prove that gradient descent converges to the global optimum for Lipschitz convex functions.

Theorem 1.4. *Let $f : \mathbb{R}^d \to \mathbb{R}$ be a L-Lipschitz convex function and $x^* = \arg\min_x f(x)$. Then, GD with step-size $t \leq 1/L$ satisfies the following:*

$$f(x_k) \leq f(x^*) + \frac{\|x_0 - x^*\|_2^2}{2tk}.$$

In particular, $\frac{L\|x_0 - x^*\|_2^2}{\varepsilon}$ iterations suffice to find an ε-approximate optimal value x (for $t = 1/L$).

Proof. First, by convexity of f, we have:

$$f(x_i) \leq f(x^*) + \langle \nabla f(x_i), x_i - x^* \rangle. \tag{1.1}$$

Further, as f is L-Lipschitz, by the previous lemma,

$$\begin{aligned} f(x_{i+1}) &\leq f(x_i) + \langle \nabla f(x_i), x_{i+1} - x_i \rangle + \frac{L}{2}\|x_{i+1} - x_i\|_2^2 \\ &= f(x_i) - t\|\nabla f(x_i)\|_2^2 + \frac{Lt^2}{2}\|\nabla f(x_i)\|_2^2 \\ &= f(x_i) - t(1 - Lt/2)\|\nabla f(x_i)\|_2^2 \\ &\leq f(x_i) - \frac{t}{2}\|\nabla f(x_i)\|_2^2, \end{aligned} \tag{1.2}$$

where the last inequality follows as $Lt \leq 1$. In particular, the above shows that GD is monotonic: the objective value is non-decreasing. Combining the above two equations and the fact that $\nabla f(x_i) = (1/t)(x_i - x_{i+1})$, we get

$$\begin{aligned} f(x_{i+1}) &\leq f(x^*) + \langle \nabla f(x_i), x_i - x^* \rangle - \frac{t}{2}\|\nabla f(x_i)\|_2^2 \\ &= f(x^*) + \frac{1}{t} \cdot \langle x_i - x_{i+1}, x_i - x^* \rangle - \frac{1}{2t}\|x_i - x_{i+1}\|^2 \\ &= f(x^*) + \frac{1}{2t}\|x_i - x^*\|_2^2 - \frac{1}{2t}\left(\|x_i - x^*\|_2^2 - 2\langle t\nabla f(x_i), x_i - x^* \rangle + \|t\nabla f(x_i)\|_2^2\right) \\ &\text{(we are basically ``completing'' a square for the last two terms)} \\ &= f(x^*) + \frac{1}{2t}\|x_i - x^*\|_2^2 - \frac{1}{2t}\|x_i - x^* - t\nabla f(x_i)\|_2^2 \\ &= f(x^*) + \frac{1}{2t}\left(\|x_i - x^*\|_2^2 - \|x_{i+1} - x^*\|_2^2\right). \end{aligned} \tag{1.3}$$

Summing the above equations for $i = 0, \ldots, k-1$, we get

$$\sum_{i=0}^{k-1} (f(x_{i+1}) - f(x^*)) \leq \frac{1}{2t} \left(\|x_0 - x^*\|_2^2 - \|x_k - x^*\|_2^2 \right) \leq \frac{\|x_0 - x^*\|_2^2}{2t}.$$

Finally, by Equation 1.2, $f(x_0), \ldots, f(x_k)$ is non-increasing. Therefore, $f(x_k) - f(x^*) \leq f(x_i) - f(x^*)$ for all $i < k$. Thus,

$$k \cdot (f(x_k) - f(x^*)) \leq \frac{\|x_0 - x^*\|_2^2}{2t}.$$

The theorem now follows. □

2 Stochastic gradient descent

We discussed several advantages of gradient descent. However, one disadvantage of GD is that sometimes it may be too expensive to compute the gradient of a function. Indeed, even for the special case of Least Squares Regression (LSR), the gradient depends on *all* the data points and thus requires time $O(nd)$ time to compute when there are n data points. In many situations, we cannot afford this. The basic idea of *stochastic gradient descent* (SGD) is to instead use an *estimator* for the gradient at each iteration. This results in significant speed-up of per-iteration cost and also does not hurt the number of iterations needed too much. SGD (as applied to ERM) also has several important advantages coming from statistical machine learning. We unfortunately will overlook these aspects completely.

In the following we describe SGD and analyze the rate of convergence for Lipschitz convex functions. As before, we have a L-Lipschitz convex function $f : \mathbb{R}^d \to \mathbb{R}$ that we are trying to minimize. The basic template for SGD is as follows:

1. Pick x_0 and set step-size $t > 0$.
2. For $i = 0, 1, \ldots,$:
   (a) Let $\boldsymbol{v}_i$ be a *random vector* such that $\mathbb{E}[\boldsymbol{v}_i] = \nabla f(x_i)$. That is, $\boldsymbol{v}_i$ is an *unbiased estimator* for $\nabla f(x_i)$. For simplicity, we will assume that $\boldsymbol{v}_i$ is independent of all previous random choices; technically, one only needs the conditional expectation to equal the gradient.
   (b) $x_{i+1} = x_i - t\boldsymbol{v}_i$.

Example: Let us look at the example of LSR again. We have data points $a^1, \ldots, a^n \in \mathbb{R}^d$ with respective values $b_1, \ldots, b_n \in \mathbb{R}$ and our goal was to find x minimizing

$$f(x) = (1/n) \sum_{j=1}^{n} \left(\langle a^j, x \rangle - b_j \right)^2.$$

Then, $\nabla f(x) = (2/n) \sum_{j=1}^{n} \left(\langle a^j, x \rangle - b_j \right) \cdot a^j$. This takes $O(nd)$ time to compute. Now, define a random vector $\boldsymbol{v}(x)$ as follows: Pick a uniformly random $j \in [n]$ and set $\boldsymbol{v}(x) = 2 \left(\langle a^j, x \rangle - b_j \right) \cdot a^j$. Then, clearly, $\mathbb{E}[\boldsymbol{v}(x)] = \nabla f(x)$ and only takes $O(d)$ time to compute. In particular, SGD applied to LSR yields the following algorithm:

1. Pick x_0 and set step-size $t > 0$.
2. For $i = 0, 1, \ldots,$:
   (a) Pick a uniformly random index[1] $j \in [n]$ and set $\boldsymbol{v}_i = 2\left(\langle a^j, x\rangle - b_i\right) \cdot a^j$.
   (b) Set $x_{i+1} = x_i - t\boldsymbol{v}_i$.

The above algorithm can also be extended straightforwardly to ERM in general.

2.1 Analysis of SGD

We now bound the convergence rate for SGD. For a random vector $\boldsymbol{v}$, define the *variance* of the vector by $\mathsf{Var}(\boldsymbol{v}) = \mathbb{E}[\|\boldsymbol{v}\|_2^2] - \|\mathbb{E}[\boldsymbol{v}]\|_2^2$. To simplify the analysis of our algorithm[2], we actually look at a variant of SGD where the final output is the average of all the intermediate iterations; that is, after k iterations, we look at $\overline{x_k} = (1/k)(x_1 + \ldots + x_k)$. Alternately, the same guarantees hold for the point with the least objective value among the iterates: $x_k^* = \arg\min\{f(x_1), f(x_2), \ldots, f(x_k)\}$. Looking at the average has other theoretical advantages especially in settings where one cannot compute the objective value easily (cf. online learning).

Theorem 2.1. *Let $f : \mathbb{R}^d \to \mathbb{R}$ be a L-Lipschitz convex function and $x^* = \arg\min_x f(x)$. Consider an instance of SGD where the estimators $\boldsymbol{v}_i$ have bounded variance: for all $i \geq 0$, $\mathsf{Var}(\boldsymbol{v}_i) \leq \sigma^2$. Then, for any $k > 1$, SGD with step-size $t \leq 1/L$ satisifes*

$$\mathbb{E}[f(\overline{x_k})] \leq f(x^*) + \frac{\|x_0 - x^*\|_2^2}{2tk} + \frac{t\sigma^2}{2},$$

where $\overline{x_k} = (1/k)(x_1 + \ldots + x_k)$. In particular, for $k = (\sigma^2 + L\|x_0 - x^*\|_2^2)^2/\varepsilon^2$ iterations suffice to find a 2ε-approximate optimal value—in expectation—x by setting $t = 1/\sqrt{k}$.

Proof. The argument is very similar to that of the analysis of GD. By Lemma 1.2,

$$\begin{aligned} f(x_{i+1}) &\leq f(x_i) + \langle \nabla f(x_i), x_{i+1} - x_i\rangle + \frac{L}{2}\|x_{i+1} - x_i\|_2^2 \\ &= f(x_i) - t\langle \nabla f(x_i), \boldsymbol{v_i}\rangle + \frac{Lt^2}{2}\|\boldsymbol{v}_i\|_2^2. \end{aligned}$$

By taking expectation on both sides with respect to the choice of $\boldsymbol{v}_t$, we get

$$\begin{aligned} \mathbb{E}[f(x_{i+1})] &\leq f(x_i) - t\|\nabla f(x_i)\|_2^2 + \frac{Lt^2}{2}\left(\|\nabla f(x_i)\|_2^2 + \mathsf{Var}(\boldsymbol{v}_t)\right) \qquad (2.1) \\ &\leq f(x_i) - t(1 - Lt/2)\|\nabla f(x_i)\|_2^2 + \frac{Lt^2}{2}\sigma^2 \\ &\leq f(x_i) - \frac{t}{2}\|\nabla f(x_i)\|_2^2 + \frac{t}{2}\sigma^2, \end{aligned}$$

where the last inequality follows as $Lt \leq 1$. Note that, unlike GD, SGD need not be monotonic[3]. Combining the above two equations we get

$$\mathbb{E}[f(x_{i+1})] \leq f(x^*) + \langle \nabla f(x_i), x_i - x^*\rangle - \frac{t}{2}\|\nabla f(x_i)\|_2^2 + \frac{t}{2}\sigma^2.$$

[1] In practice, one would order the data randomly and process them sequentially after that.
[2] In practice, one would try various tricks to pick the right one.
[3] However, it is *almost* monotonic if σ is small

We now back-substitute $\boldsymbol{v}_i$ into the equation by using $\mathbb{E}[\boldsymbol{v}_i] = \nabla f(x_i)$ and $\|\nabla f(x_i)\|_2^2 = \mathbb{E}[\|\boldsymbol{v}_i\|_2^2] - Var(\boldsymbol{v}_i) \leq \mathbb{E}[\|\boldsymbol{v}_i\|_2^2] - \sigma^2$:

$$\begin{aligned}\mathbb{E}[f(x_{i+1})] &\leq f(x^*) + \langle \mathbb{E}[\boldsymbol{v}_i], x_i - x^* \rangle - \frac{t}{2}\,\mathbb{E}[\|\boldsymbol{v}_i\|_2^2] + t\sigma^2 \\ &= f(x^*) + \mathbb{E}\left[\langle \boldsymbol{v}_i, x_i - x^* \rangle - \frac{t}{2}\|\boldsymbol{v}_i\|_2^2\right] + t\sigma^2.\end{aligned}$$

We now repeat the calculations as in the analysis of GD (Equation 1.3) by completing the square for the middle two terms to get:

$$\begin{aligned}\mathbb{E}[f(x_{i+1})] &\leq f(x^*) + \mathbb{E}\left[\frac{1}{2t}\left(\|x_i - x^*\|_2^2 - \|x_i - x^* - t\boldsymbol{v}_i\|_2^2\right)\right] + t\sigma^2 \\ &= f(x^*) + \mathbb{E}\left[\frac{1}{2t}\left(\|x_i - x^*\|_2^2 - \|x_{i+1} - x^*\|_2^2\right)\right] + t\sigma^2.\end{aligned}$$

The above is analogous to Equation 1.3 but for an additional $t\sigma^2$ term (and taking expectation). Summing the above equations for $i = 0, \ldots, k-1$, we get

$$\sum_{i=0}^{k-1} \left(\mathbb{E}[f(x_{i+1})] - f(x^*)\right) \leq \frac{1}{2t}\left(\|x_0 - x^*\|_2^2 - \mathbb{E}\left[\|x_k - x^*\|_2^2\right]\right) + kt\sigma^2 \leq \frac{\|x_0 - x^*\|_2^2}{2t} + kt\sigma^2.$$

Finally, by Claim 1.3,

$$k \cdot f(\overline{x_k}) = k \cdot f\left(\frac{x_1 + \ldots + x_k}{k}\right) \leq f(x_1) + \cdots + f(x_k).$$

Thus,

$$\sum_{i=0}^{k-1} \left(\mathbb{E}[f(x_{i+1})] - f(x^*)\right) = \mathbb{E}\left[f(x_1) + \cdots + f(x_k)\right] - kf(x^*) \geq k\,\mathbb{E}\left[f(\overline{x_k})\right] - kf(x^*).$$

Combining the above equations we get

$$\mathbb{E}[f(\overline{x_k})] \leq f(x^*) + \frac{\|x_0 - x^*\|_2^2}{2tk} + t\sigma^2.$$

The main statement of the theorem now follows. The in particular part follows by substituting the specific values of k and t. □

3 Remarks about GD and SGD

- SGD is widely used in many large-scale machine learning systems. It is simple, efficient, can be run in parallel, and is ideal for ERM.
- In practice, one uses various heuristics and tricks to implement GD or SGD for choosing the step-size (such as line-search) and stopping criterion. Choosing the right parameters is a bit of black-art and you can find some advice here.

- If you ignore the dependence on L, and $\|x_0 - x^*\|_2$, GD takes $O(1/\varepsilon)$ iterations and SGD takes $O(1/\varepsilon^2)$ iterations to get ε-error. Both these bounds are tight for the specific algorithms.
- Remarkably, there are various accelerated gradient descent algorithms which only need $O(1/\sqrt{\varepsilon})$ iterations. This can be quite significant when you can compute the gradient fast: the accelerated methods—such as the celebrated Nesterov's AGD—only need two gradient evaluations.
- Unfortunately, the acceleration does not help in the stochastic setting. When using estimators with variance σ^2, the best error one can get after k iterations is $O(L/k^2 + \sigma/\sqrt{k})$.
- Even more remarkably, one can show that if only given access to gradient computations, it is not possible to do better than $\Omega(\sqrt{L/\varepsilon})$ iterations. A good resource for such advanced material is Nesterov's textbook *Introductory lectures on convex optimization*.

3.1 Subgradient Methods

In several applications, the cost function $f : \mathbb{R}^d \to \mathbb{R}$ though convex could be non-differentiable. Such situations can sometimes be handled by *subgradient descent*. Recall one of the nice properties of the gradient: if f is convex and is differentiable at a point x, then for all y,

$$f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle.$$

The above inequality in fact serves as one of the principles behind gradient descent and motivates the definition of *subgradient*.

Definition 3.1. *For a function $f : \mathbb{R}^d \to \mathbb{R}$, and a point $x \in \mathbb{R}^d$, a vector $v \in \mathbb{R}^d$ is a* subgradient *of f at x if for all $y \in \mathbb{R}^d$,*

$$f(y) \geq f(x) + \langle v, y - x \rangle.$$

In words, the hyperplane at x with normal v lies below f. Note that from the definition, there can be multiple subgradients for a function at a point. However, *if* is differentiable at x, then $\nabla f(x)$ is the only subgradient of f at x. For example, consider $f : \mathbb{R} \to \mathbb{R}$ defined by $f(x) = \max(x, 0)$ [4]. Then, for all $x > 0$, the subgradient of f at x is 1 and for $x < 0$, the subgradient is 0. How about $x = 0$? Any number between $[0, 1]$ is a subgradient.

Subgradient descent is given by an update similar to gradient descent: $x_{i+1} = x_i - tv$, where v is any subgradient of f at x_i. One can show that for some suitable notion of Lipschitz-ness, subgradient descent also converges to the global minimum of convex Lipschitz functions. We will unfortunately not cover this.

[4] This is the *RELU* function that is used often in neural networks.

4 Missing proofs

Proof of Claim 1.2. By the fundamental theorem of calculus[5],

$$\begin{aligned}
f(y) &= f(x) + \int_0^1 \langle \nabla f(x + \tau(y-x)), y - x\rangle d\tau \\
&= f(x) + \langle \nabla f(x), y - x\rangle + \int_0^1 \langle \nabla f(x + \tau(y-x)) - \nabla f(x), y - x\rangle d\tau \\
&\le f(x) + \langle \nabla f(x), y - x\rangle + \int_0^1 \|\nabla f(x + \tau(y-x)) - \nabla f(x)\|_2 \|y - x\|_2 d\tau \\
&\text{(as for all vectors } u, v, \ |\langle u, v\rangle| \le \|u\|_2 \cdot \|v\|_2). \\
&\le f(x) + \langle \nabla f(x), y - x\rangle + \int_0^1 L\|\tau(y-x)\|_2 \|y - x\|_2 d\tau \\
&\text{(as } f \text{ is } L\text{-Lipschitz)} \\
&= f(x) + \langle \nabla f(x), y - x\rangle + \frac{L}{2}\|y - x\|_2^2.
\end{aligned}$$

□

Proof of Claim 1.3. The proof is by induction. For $k = 2$, the claim follows by convexity. Suppose it is true for $k-1$. Let $y = (y_1 + \ldots + y_{k-1})/(k-1)$. Then,

$$\begin{aligned}
f\left(\frac{y_1 + \cdots + y_k}{k}\right) &= f\left(\frac{k-1}{k} y + \frac{y_k}{k}\right) \\
&\le \frac{k-1}{k} \cdot f(y) + \frac{1}{k} f(y_k) \\
&\text{(by convexity applied with } \lambda = (k-1)/k) \\
&\le \left(\frac{k-1}{k}\right) \cdot \frac{f(y_1) + \cdots + f(y_{k-1})}{k-1} + \left(\frac{1}{k}\right) f(y_k) \\
&\text{(induction hypothesis applied to } y_1, \ldots, y_{k-1}) \\
&= \frac{f(y_1) + \cdots + f(y_k)}{k}.
\end{aligned}$$

□

[5] Basically, you define a function $h : \mathbb{R} \to \mathbb{R}$ by $h(\tau) = f(x + \tau(y - x))$ and apply the fundamental theorem of calculus to h: $h(1) = h(0) + \int_0^1 h'(\tau) d\tau$.