1 Introduction

The *subgradient method* is a very simple algorithm for minimizing a nondifferentiable convex function. The method looks very much like the ordinary gradient method for differentiable functions, but with several notable exceptions:

- The subgradient method applies directly to *nondifferentiable* f.
- The step lengths are not chosen via a line search, as in the ordinary gradient method. In the most common cases, the step lengths are fixed ahead of time.
- Unlike the ordinary gradient method, the subgradient method is *not* a descent method; the function value can (and often does) increase.

The subgradient method is readily extended to handle problems with constraints.

Subgradient methods can be *much* slower than interior-point methods (or Newton's method in the unconstrained case). In particular, they are first-order methods; their performance depends very much on the problem scaling and conditioning. (In contrast, Newton and interior-point methods are second-order methods, not affected by problem scaling.)

However, subgradient methods do have some advantages over interior-point and Newton methods. They can be immediately applied to a far wider variety of problems than interior-point or Newton methods. The memory requirement of subgradient methods can be much smaller than an interior-point or Newton method, which means it can be used for extremely large problems for which interior-point or Newton methods cannot be used. Moreover, by combining the subgradient method with primal or dual decomposition techniques, it is sometimes possible to develop a simple distributed algorithm for a problem. In any case, subgradient methods are well worth knowing about.

The subgradient method was originally developed by Shor and others in the Soviet Union in the 1960s and 1970s. A basic reference on subgradient methods is his book [Sho85]; a very clear discussion can be found in chapter 5 of Polyak's book [Pol87]. Bertsekas [Ber99] is another good reference on the subgradient method, in particular, on how to combine it with primal and dual decomposition. Other book treatments of the topic are in Ruszczynski [Rus06, §7.1], Nesterov [Nes04, Chap. 3], Akgul [Akg84], Yudin and Nemirovski [NY83], Censor and Zenios [CZ97], and Shor [Sho98, Chap. 2]. Some interesting recent research papers on subgradient methods are [NB01] and [Nes05].

2 Basic subgradient method

2.1 Negative subgradient update

We start with the unconstrained case, where the goal is to minimize $f : \mathbf{R}^n \to \mathbf{R}$, which is convex and has domain $\mathbf{R}^n$ (for now). To do this, the subgradient method uses the simple iteration

$$x^{(k+1)} = x^{(k)} - \alpha_k g^{(k)}.$$

Here $x^{(k)}$ is the kth iterate, $g^{(k)}$ is *any* subgradient of f at $x^{(k)}$, and $\alpha_k > 0$ is the kth step size. Thus, at each iteration of the subgradient method, we take a step in the direction of a negative subgradient.

Recall that a subgradient of f at x is any vector g that satisfies the inequality $f(y) \geq f(x) + g^T(y - x)$ for all y. When f is differentiable, the only possible choice for $g^{(k)}$ is $\nabla f(x^{(k)})$, and the subgradient method then reduces to the gradient method (except, as we'll see below, for the choice of step size). The set of subgradients of f at x is the subdifferential of f at x, denoted $\partial f(x)$. So the condition that $g^{(k)}$ be a subgradient of f at $x^{(k)}$ can be written $g^{(k)} \in \partial f(x^{(x)})$.

It can happen that $-g^{(k)}$ is not a descent direction for f at $x^{(k)}$, *i.e.*, $f'(x; -g^{(k)}) > 0$. In such cases we always have $f(x^{(k+1)}) > f(x^{(k)})$. Even when $-g^{(k)}$ is a descent direction at $x^{(k)}$, the step size can be such $f(x^{(k+1)}) > f(x^{(k)})$. In other words, an iteration of the subgradient method can increase the objective function.

Since the subgradient method is not a descent method, it is common to keep track of the best point found so far, *i.e.*, the one with smallest function value. At each step, we set

$$f_{\text{best}}^{(k)} = \min\{f_{\text{best}}^{(k-1)}, f(x^{(k)})\},$$

and set $i_{\text{best}}^{(k)} = k$ if $f(x^{(k)}) = f_{\text{best}}^{(k)}$, *i.e.*, if $x^{(k)}$ is the best point found so far. (In a descent method there is no need to do this, since the current point is always the best one so far.) Then we have

$$f_{\text{best}}^{(k)} = \min\{f(x^{(1)}), \ldots, f(x^{(k)})\},$$

i.e., the best objective value found in k iterations. Since $f_{\text{best}}^{(k)}$ is decreasing, it has a limit (which can be $-\infty$).

2.2 Step size rules

In the subgradient method the step size selection is very different from the standard gradient method. Many different types of step size rules are used. We'll start with five basic step size rules.

- *Constant step size.* $\alpha_k = \alpha$ is a positive constant, independent of k.
- *Constant step length.* $\alpha_k = \gamma / \|g^{(k)}\|_2$, where $\gamma > 0$. This means that $\|x^{(k+1)} - x^{(k)}\|_2 = \gamma$.
- *Square summable but not summable.* The step sizes satisfy

$$\alpha_k \geq 0, \qquad \sum_{k=1}^{\infty} \alpha_k^2 < \infty, \qquad \sum_{k=1}^{\infty} \alpha_k = \infty.$$

  One typical example is $\alpha_k = a/(b + k)$, where $a > 0$ and $b \geq 0$.

- *Nonsummable diminishing.* The step sizes satisfy

$$\alpha_k \geq 0, \qquad \lim_{k\to\infty} \alpha_k = 0, \qquad \sum_{k=1}^{\infty} \alpha_k = \infty.$$

  Step sizes that satisfy this condition are called *diminishing step size rules*. A typical example is $\alpha_k = a/\sqrt{k}$, where $a > 0$.

- *Nonsummable diminishing step lengths.* The step sizes are chosen as $\alpha_k = \gamma_k/\|g^{(k)}\|_2$, where

$$\gamma_k \geq 0, \qquad \lim_{k\to\infty} \gamma_k = 0, \qquad \sum_{k=1}^{\infty} \gamma_k = \infty.$$

There are still other choices, and many variations on these choices. In §4.1 we will encounter another step size rule that requires knowledge of the optimal value $f^\star$.

The most interesting feature of these choices is that *they are determined before the algorithm is run; they do not depend on any data computed during the algorithm.* This is very different from the step size rules found in standard descent methods, which very much depend on the current point and search direction.

2.3 Convergence results

There are many results on convergence of the subgradient method. For constant step size and constant step length, the subgradient algorithm is guaranteed to converge to within some range of the optimal value, *i.e.*, we have

$$\lim_{k\to\infty} f_{\text{best}}^{(k)} - f^\star < \epsilon,$$

where $f^\star$ denotes the optimal value of the problem, *i.e.*, $f^\star = \inf_x f(x)$. (This implies that the subgradient method finds an ϵ-suboptimal point within a finite number of steps.) The number ϵ is a function of the step size parameter h, and decreases with it.

For the diminishing step size and step length rules (and therefore also the square summable but not summable step size rule), the algorithm is guaranteed to converge to the optimal value, *i.e.*, we have $\lim_{k\to\infty} f(x^{(k)}) = f^\star$. It's remarkable that such a simple algorithm can be used to minimize any convex function for which you can compute a subgradient at each point. We'll also see that the convergence proof is also simple.

When the function f is differentiable, we can say a bit more about the convergence. In this case, the subgradient method with constant step size yields convergence to the optimal value, provided the parameter α is small enough.

3 Convergence proof

3.1 Assumptions

Here we give a proof of some typical convergence results for the subgradient method. We assume that there is a minimizer of f, say $x^\star$. We also make one other assumption on f:

We will assume that the norm of the subgradients is bounded, *i.e.*, there is a G such that $\|g^{(k)}\|_2 \leq G$ for all k. This will be the case if, for example, f satisfies the Lipschitz condition

$$|f(u) - f(v)| \leq G\|u - v\|_2,$$

for all u, v, because then $\|g\|_2 \leq G$ for any $g \in \partial f(x)$, and any x. In fact, some versions of the subgradient method (*e.g.*, diminishing nonsummable step lengths) work when this assumption doesn't hold; see [Sho85] or [Pol87].

We'll also assume that a number R is known that satisfies $R \geq \|x^{(1)} - x^\star\|_2$. We can interpret R as an upper bound on $\mathbf{dist}(x^{(1)}, X^\star)$, the distance of the initial point to the optimal set.

3.2 Some basic inequalities

For the standard gradient descent method, the convergence proof is based on the function value decreasing at each step. In the subgradient method, the key quantity is not the function value (which often increases); it is the *Euclidean distance to the optimal set*.

Recall that $x^\star$ is a point that minimizes f, *i.e.*, it is an arbitrary optimal point. We have

$$\begin{aligned}
\|x^{(k+1)} - x^\star\|_2^2 &= \|x^{(k)} - \alpha_k g^{(k)} - x^\star\|_2^2 \\
&= \|x^{(k)} - x^\star\|_2^2 - 2\alpha_k g^{(k)T}(x^{(k)} - x^\star) + \alpha_k^2\|g^{(k)}\|_2^2 \\
&\leq \|x^{(k)} - x^\star\|_2^2 - 2\alpha_k(f(x^{(k)}) - f^\star) + \alpha_k^2\|g^{(k)}\|_2^2,
\end{aligned}$$

where $f^\star = f(x^\star)$. The last line follows from the definition of subgradient, which gives

$$f(x^\star) \geq f(x^{(k)}) + g^{(k)T}(x^\star - x^{(k)}).$$

Applying the inequality above recursively, we have

$$\|x^{(k+1)} - x^\star\|_2^2 \leq \|x^{(1)} - x^\star\|_2^2 - 2\sum_{i=1}^k \alpha_i(f(x^{(i)}) - f^\star) + \sum_{i=1}^k \alpha_i^2\|g^{(i)}\|_2^2.$$

Using $\|x^{(k+1)} - x^\star\|_2^2 \geq 0$ and $\|x^{(1)} - x^\star\|_2 \leq R$ we have

$$2\sum_{i=1}^k \alpha_i(f(x^{(i)}) - f^\star) \leq R^2 + \sum_{i=1}^k \alpha_i^2\|g^{(i)}\|_2^2. \tag{1}$$

Combining this with

$$\sum_{i=1}^k \alpha_i(f(x^{(i)}) - f^\star) \geq \left(\sum_{i=1}^k \alpha_i\right) \min_{i=1,\ldots,k}(f(x^{(i)}) - f^\star) = \left(\sum_{i=1}^k \alpha_i\right)(f_{\text{best}}^{(k)} - f^\star),$$

we have the inequality

$$f_{\text{best}}^{(k)} - f^\star = \min_{i=1,\ldots,k} f(x^{(i)}) - f^\star \leq \frac{R^2 + \sum_{i=1}^k \alpha_i^2\|g^{(i)}\|_2^2}{2\sum_{i=1}^k \alpha_i}. \tag{2}$$

Finally, using the assumption $\|g^{(k)}\|_2 \leq G$, we obtain the basic inequality

$$f_{\text{best}}^{(k)} - f^\star \leq \frac{R^2 + G^2 \sum_{i=1}^k \alpha_i^2}{2 \sum_{i=1}^k \alpha_i}. \tag{3}$$

From this inequality we can read off various convergence results.

Constant step size. When $\alpha_k = \alpha$, we have

$$f_{\text{best}}^{(k)} - f^\star \leq \frac{R^2 + G^2 \alpha^2 k}{2\alpha k}.$$

The righthand side converges to $G^2\alpha/2$ as $k \to \infty$. Thus, for the subgradient method with fixed step size α, $f_{\text{best}}^{(k)}$ converges to within $G^2\alpha/2$ of optimal. We also find that $f(x^{(k)}) - f^\star \leq G^2\alpha$ within at most $R^2/(G^2\alpha^2)$ steps.

Constant step length. With $\alpha_k = \gamma/\|g^{(k)}\|_2$, the inequality (2) becomes

$$f_{\text{best}}^{(k)} - f^\star \leq \frac{R^2 + \gamma^2 k}{2 \sum_{i=1}^k \alpha_i} \leq \frac{R^2 + \gamma^2 k}{2\gamma k/G},$$

using $\alpha_i \geq \gamma/G$. The righthand side converges to $G\gamma/2$ as $k \to \infty$, so in this case the subgradient method converges to within $G\gamma/2$ of optimal.

Square summable but not summable. Now suppose

$$\|\alpha\|_2^2 = \sum_{k=1}^\infty \alpha_k^2 < \infty, \qquad \sum_{k=1}^\infty \alpha_k = \infty.$$

Then we have

$$f_{\text{best}}^{(k)} - f^\star \leq \frac{R^2 + G^2\|\alpha\|_2^2}{2 \sum_{i=1}^k \alpha_i},$$

which converges to zero as $k \to \infty$, since the numerator converges to $R^2 + G^2\|\alpha\|_2^2$, and the denominator grows without bound. Thus, the subgradient method converges (in the sense $f_{\text{best}}^{(k)} \to f^\star$).

Diminishing step size rule. If the sequence α_k converges to zero and is nonsummable, then the righthand side of the inequality (3) converges to zero, which implies the subgradient method converges. To show this, let $\epsilon > 0$. Then there exists an integer N_1 such that $\alpha_i \leq \epsilon/G^2$ for all $i > N_1$. There also exists an integer N_2 such that

$$\sum_{i=1}^{N_2} \alpha_i \geq \frac{1}{\epsilon}\left(R^2 + G^2 \sum_{i=1}^{N_1} \alpha_i^2\right),$$

since $\sum_{i=1}^{\infty} \alpha_i = \infty$. Let $N = \max\{N_1, N_2\}$. Then for $k > N$, we have

$$\begin{aligned}
\frac{R^2 + G^2 \sum_{i=1}^k \alpha_i^2}{2 \sum_{i=1}^k \alpha_i} &\leq \frac{R^2 + G^2 \sum_{i=1}^{N_1} \alpha_i^2}{2 \sum_{i=1}^k \alpha_i} + \frac{G^2 \sum_{i=N_1+1}^k \alpha_i^2}{2 \sum_{i=1}^{N_1} \alpha_i + 2 \sum_{i=N_1+1}^k \alpha_i} \\
&\leq \frac{R^2 + G^2 \sum_{i=1}^{N_1} \alpha_i^2}{(2/\epsilon)\left(R^2 + G^2 \sum_{i=1}^{N_1} \alpha_i^2\right)} + \frac{G^2 \sum_{i=N_1+1}^k (\epsilon \alpha_i / G^2)}{2 \sum_{i=N_1+1}^k \alpha_i} \\
&= \frac{\epsilon}{2} + \frac{\epsilon}{2} = \epsilon.
\end{aligned}$$

Nonsummable diminishing step lengths. Finally, suppose that $\alpha_k = \gamma_k / \|g^{(k)}\|_2$, with γ_k nonsummable and converging to zero. The inequality (2) becomes

$$f_{\text{best}}^{(k)} - f^\star \leq \frac{R^2 + \sum_{i=1}^k \gamma_k^2}{2 \sum_{i=1}^k \alpha_i} \leq \frac{R^2 + \sum_{i=1}^k \gamma_k^2}{(2/G) \sum_{i=1}^k \gamma_i},$$

which converges to zero as $k \to 0$.

3.3 A bound on the suboptimality bound

It's interesting to ask the question, what sequence of step sizes minimizes the righthand side of (3)? In other words, how do we choose positive $\alpha_1, \ldots, \alpha_k$ so that

$$\frac{R^2 + G^2 \sum_{i=1}^k \alpha_i^2}{2 \sum_{i=1}^k \alpha_i}$$

(which is an upper bound on $f_{\text{best}}^{(k)} - f^\star$) is minimized? This is a convex and symmetric function of $\alpha_1, \ldots, \alpha_k$, so we conclude the optimal occurs when all α_i are equal (to, say, α). This reduces our suboptimality bound to

$$\frac{R^2 + G^2 k \alpha^2}{2k\alpha}$$

which is minimized by $\alpha = (R/G)/\sqrt{k}$.

In other words, the choice of $\alpha_1, \ldots, \alpha_k$ that minimizes the suboptimality bound (3) is given by

$$\alpha_i = (R/G)/\sqrt{k}, \quad i = 1, \ldots, k.$$

This choice of constant step size yields the suboptimality bound

$$f_{\text{best}}^{(k)} - f^\star \leq RG/\sqrt{k}.$$

Put another way, we can say that for *any* choice of step sizes, the suboptimality bound (3) must be at least as large as $RG/\sqrt{k}$. If we use (3) as our stopping criterion, then the number of steps to achieve a guaranteed accuracy of ϵ will be at least $(RG/\epsilon)^2$, no matter what step sizes we use. (It will be this number if we use the step size $\alpha_k = (R/G)/\sqrt{k}$).

Note that RG has a simple interpertation as an initial bound on $f(x^{(1)}) - f^\star$, based on $\|x^{(1)} - x^\star\|_2 \leq R$ and the Lipschitz constant G for f. Thus $(RG)/\epsilon$ is the ratio of initial uncertainty in $f^\star$ to final uncertainty in $f^\star$. If we square this number, we get the minimum number of steps it will take to achieve this reduction in uncertainty. This tells us that the subgradient method is going to be very slow, if we use (3) as our stopping criterion. To reduce the initial uncertainty by a factor of 1000, say, it will require at least 10^6 iterations.

4 Projected subgradient method

One extension of the subgradient method is the *projected subgradient method*, which solves the constrained convex optimization problem

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & x \in \mathcal{C}, \end{array}$$

where $\mathcal{C}$ is a convex set. The projected subgradient method is given by

$$x^{(k+1)} = P\left(x^{(k)} - \alpha_k g^{(k)}\right),$$

where P is (Euclidean) projection on $\mathcal{C}$, and $g^{(k)}$ is any subgradient of f at $x^{(k)}$. The step size rules described before can be used here, with similar convergence results. Note that $x^{(k)} \in \mathcal{C}$, *i.e.*, $x^{(k)}$ is feasible.

The convergence proofs for the subgradient method are readily extended to handle the projected subgradient method. Let $z^{(k+1)} = x^{(k)} - \alpha_k g^{(k)}$, *i.e.*, a standard subgradient update, before the projection back onto $\mathcal{C}$. As in the subgradient method, we have

$$\begin{aligned} \|z^{(k+1)} - x^\star\|_2^2 &= \|x^{(k)} - \alpha_k g^{(k)} - x^\star\|_2^2 \\ &= \|x^{(k)} - x^\star\|_2^2 - 2\alpha_k g^{(k)T}(x^{(k)} - x^\star) + \alpha_k^2 \|g^{(k)}\|_2^2 \\ &\leq \|x^{(k)} - x^\star\|_2^2 - 2\alpha_k (f(x^{(k)}) - f^\star) + \alpha_k^2 \|g^{(k)}\|_2^2. \end{aligned}$$

Now we observe that

$$\|x^{(k+1)} - x^\star\|_2 = \|P(z^{(k+1)}) - x^\star\|_2 \leq \|z^{(k+1)} - x^\star\|_2,$$

i.e., when we project a point onto $\mathcal{C}$, we move closer to every point in $\mathcal{C}$, and in particular, any optimal point. Combining this with the inequality above we get

$$\|x^{(k+1)} - x^\star\|_2^2 \leq \|x^{(k)} - x^\star\|_2^2 - 2\alpha_k (f(x^{(k)}) - f^\star) + \alpha_k^2 \|g^{(k)}\|_2^2,$$

and the proof proceeds exactly as in the ordinary subgradient method.

5 Projected subgradient for dual problem

One famous application of the projected subgradient method is to the dual problem. We start with the (convex) primal problem

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq 0, \quad i = 1, \ldots, m \end{array}$$

We'll assume, for simplicity, that for each $\lambda \succeq 0$, the Lagrangian

$$L(x, \lambda) = f_0(x) + \sum_{i=1}^{m} \lambda_i f_i(x)$$

has a unique minimizer over x, which we denote $x^*(\lambda)$. The dual function is then

$$g(\lambda) = \inf_x L(x, \lambda) = f_0(x^*(\lambda)) + \sum_{i=1}^{m} \lambda_i f_i(x^*(\lambda))$$

(for $\lambda \succeq 0$). The dual problem is

$$\begin{array}{ll} \text{maximize} & g(\lambda) \\ \text{subject to} & \lambda \succeq 0. \end{array}$$

We'll assume that Slater's condition holds (again, for simplicity), so we can solve the primal problem by finding an optimal point $\lambda^\star$ of the dual, and then taking $x^\star = x^*(\lambda^\star)$. (For a discussion of solving the primal problem via the dual, see [BV04, §5.5.5].)

We will solve the dual problem using the projected subgradient method,

$$\lambda^{(k+1)} = \left(\lambda^{(k)} - \alpha_k h\right)_+, \qquad h \in \partial(-g)(\lambda^{(k)}).$$

Let's now work out a subgradient of the negative dual function. Since $-g$ is a supremum of a family of affine functions of λ, indexed by x, we can find a subgradient by finding one of these functions that achieves the supremum. But there is just one, and it is

$$-f_0(x^*(\lambda)) - \sum_{i=1}^{m} \lambda_i f_i(x^*(\lambda)),$$

which has gradient (with respect to λ)

$$h = -(f_1(x^*(\lambda)), \ldots, f_m(x^*(\lambda)) \in \partial(-g)(\lambda).$$

(Our assumptions imply that $-g$ has only one element in its subdifferential, which means g is differentiable. Differentiability means that a small enough constant step size will yield convergence. In any case, the projected subgradient method can be used in cases where the dual is nondifferentiable.)

The projected subgradient method for the dual has the form

$$x^{(k)} = x^*(\lambda^{(k)}), \qquad \lambda_i^{(k+1)} = \left(\lambda_i^{(k)} + \alpha_k f_i(x^{(k)})\right)_+ \tag{9}$$

In this algorithm, the primal iterates $x^{(k)}$ are not feasible, but become feasible only in the limit. (Sometimes we can find a method for constructing a feasible, suboptimal $\tilde{x}^{(k)}$ from $x^{(k)}$.) The dual function values $g(\lambda^{(k)})$, as well as the primal function values $f_0(x^{(k)})$, converge to $f^\star = f_0(x^\star)$.

We can give a simple interpretation of the algorithm (9). We interpret λ_i as the price for a 'resource' with usage measured by $f_i(x)$. When we calculate $x^*(\lambda)$, we are finding the x that minimizes the total cost, *i.e.*, the objective plus the total bill (or revenue) for the resources used. The goal is to adjust the prices so that the resource usage is within budget (*i.e.*, $f_i(x) \leq 0$). At each step, we increase the price λ_i if resource i is over-utilized (*i.e.*, $f_i(x) > 0$), and we decrease the price λ_i if resource i is under-utilized (*i.e.*, $f_i(x) < 0$). But we never let prices get negative (which would encourage, rather than discourage, resource usage).

In general, there is no reason to solve the dual instead of the primal. But for specific problems there can be an advantage. We will see later that the projected subgradient dual algorithm (9) is, in some cases, a *decentralized* algorithm.