Optimisation Combinatoire et Convexe

First Order Methods: part I

Today

First Order Methods: Part One.

- Introduction
  - Exploiting structure
  - Classification
- Gradient/projection based methods
  - Acceleration
  - Optimal complexity, resisting oracles

Introduction

Interior Point Methods, Netwon.

- Even with efficient linear algebra, exploiting structure in the KKT system computing the Newton step, the cost of one iteration becomes prohibitive.
- The dependence on the precision target is logarithmic $O(\log(1/\epsilon))$: Newton's method produces high precision solutions, which is often unnecessary.
- Very good agreement between theoretical complexity bounds and empirical performance:
  - Two convergence phases for Newton's method (damped, quadratic).
  - Dimension independence: only precision improvement matters in Newton's iterations.
  - Very good dependence on precision target.
  - Affine invariance: immune to conditioning issues.

Unfortunately: does not scale forever. . .

Introduction

First order methods.

- Dependence on precision is polynomial $O(1/\epsilon^\alpha)$, not logarithmic $O(\log(1/\epsilon))$. This is OK in many applications (stats, etc).
- Run a much larger number of cheaper iterations. No Hessian means significantly lower memory and CPU costs per iteration.
- Lack of second order information means conditioning issues have much more impact on numerical performance.
- Much greater gap between theoretical complexity bounds and empirical performance.
- No unified analysis (self-concordance for IPM): large library of disparate methods.
- Algorithmic choices strictly constrained by **problem structure**.

Objective: classify these techniques, study their performance & complexity.

Introduction

First order methods. Algorithmic choices based on problem structure.

- Some optimization subproblems can be solved very efficiently (thresholding, binary search, SVD, etc).
- Classify algorithms according to these subproblems:
  - **Projection.** Project the current iterate on a simple convex set, according to a certain norm. Iterates are mostly based on projected gradient steps.
  - **Centering.** Solve a centering problem at each iteration and compute a subgradient at the center to localize the solution.
  - **Affine maximization.** Solve an affine maximization problem over the feasible set.
  - **Partial optimization.** Solve the minimization problem over a subset of the variables.
- Solving large-scale programs means solving a long sequence of these subproblems.

Gradient/projection methods

Gradient/projection methods: introduction

Solve

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

in $x \in \mathbb{R}^n$, with $C \subset \mathbb{R}^n$ convex.

Main assumptions in the subgradient/gradient methods that follow:

- The gradient $\nabla f(x)$ or a subgradient can be computed efficiently.
- If C is not $\mathbb{R}^n$, for any $y \in \mathbb{R}^n$, the following **subproblem can be solved efficiently**

  $$\begin{array}{ll} \text{minimize} & y^T x + d(x) \\ \text{subject to} & x \in C \end{array}$$

  in the variable $x \in \mathbb{R}^n$, where $d(x)$ is a **strongly convex** function. Typically, $d(x) = \|x\|_2^2$ and this is an Euclidean projection.

We will always assume that C is simple enough so that this projection step can be solved efficiently.

Subgradient. Definition.

- Suppose that f is a convex function with $\mathbf{dom} f = \mathbb{R}^n$, and that there is a vector $g \in \mathbb{R}^n$ such that:

$$f(y) \geq f(x) + g^T(y - x), \quad \text{for all } y \in \mathbb{R}^n$$

- The vector g is called a **subgradient** of f at x, we write $g \in \partial f$.
- Of course, if f is differentiable, the gradient of f at x satisfies this condition
- The subgradient defines a **supporting hyperplane** for f at the point x

Gradient methods

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

In theory...

- The theoretical convergence speed of gradient based methods is mostly controlled by the smoothness of the objective.

Convex objective $f(x)$	**Iterations...**
Nondifferentiable	$O(1/\epsilon^2)$
Differentiable	$O(1/\epsilon^2)$
Smooth (Lipschitz gradient)	$O(1/\sqrt{\epsilon})$
Strongly convex	$O(\log(1/\epsilon))$

- Obviously, the geometry of the (convex) feasible set also has an impact.

In practice...

- Compared to IPM, much larger gap between theoretical complexity guarantees and empirical performance.
- Conditioning, well-posedness, etc. also have a very strong impact.

Subgradient Methods

Subgradient method.

- **Algorithm.** At each iteration k, update the current point x_k according to:

$$x_{k+1} = x_k + \alpha_k g_k$$

  where g_k is a subgradient of f at x_k
- α_k is the step size sequence
- Similar to gradient descent but, not a descent method . . .
- Instead: use the best point and the minimum function value found so far

Subgradient methods

Step size strategies:

- Constant step size: $\alpha_k = h$ for all $k \geq 0$
- Constant step length: $\alpha_k / \|g_k\| = h$ for all $k \geq 0$
- Square summable but not summable:

$$\sum_{k=0}^{\infty} \alpha_k = \infty \quad \text{and} \quad \sum_{k=0}^{\infty} \alpha_k^2 < \infty$$

- Nonsummable diminishing:

$$\sum_{k=0}^{\infty} \alpha_k = \infty \quad \text{and} \quad \lim_{k \to \infty} \alpha_k = 0$$

Subgradient methods: convergence

Convergence proof. For standard gradient descent methods, convergence is based on the function value decreasing at each step. Here, the function value often increases, but the *Euclidean distance to the optimal set* converges.

Proposition

Subgradient method complexity. *Assuming* $\|g\|_2 \leq G$, *for all* $g \in \partial f$, *the subgradient method with step size* α_i *satisifes*

$$f_{\text{best}} - f^\star \leq \frac{\mathbf{dist}(x_1, x^*)^2 + G^2 \sum_{i=1}^k \alpha_i^2}{2 \sum_{i=1}^k \alpha_i}$$

Proof. 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 $\|x^{(k+1)} - x^\star\|_2^2 \geq 0$ we have

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

Combining this with

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

we have the inequality

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

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

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

Since $x^\star$ is any minimizer of f, we can state that

$$f_{\text{best}}^{(k)} - f^\star \leq \frac{\mathbf{dist}(x^{(1)}, X^\star)^2 + G^2 \sum_{i=1}^k \alpha_i^2}{2\sum_{i=1}^k \alpha_i}. \quad \blacksquare$$

Subgradient methods: convergence

Constant step size. If $\alpha_k = h$, we have

$$f_{\text{best}}^{(k)} - f^\star \leq \frac{\mathbf{dist}(x^{(1)}, X^\star)^2 + G^2 h^2 k}{2hk}.$$

To get an ϵ solution, we set $h = 2\epsilon/G^2$ and

$$\frac{\mathbf{dist}(x_1, X^\star)^2}{2hk} \leq \epsilon$$

hence the following bound on the number of iterations

$$k \geq \frac{\mathbf{dist}(x_1, X^\star)^2 G^2}{4\epsilon^2}.$$

Subgradient methods: convergence

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

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

which converges to zero as $k \to \infty$. In other words, the subgradient method converges (in the sense $f_{\text{best}}^{(k)} \to f^\star$).

Subgradient Methods

If the problem has constraints:

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

where $C \subset \mathbb{R}^n$ is a convex set

- Use the Euclidean projection $p_C(\cdot)$

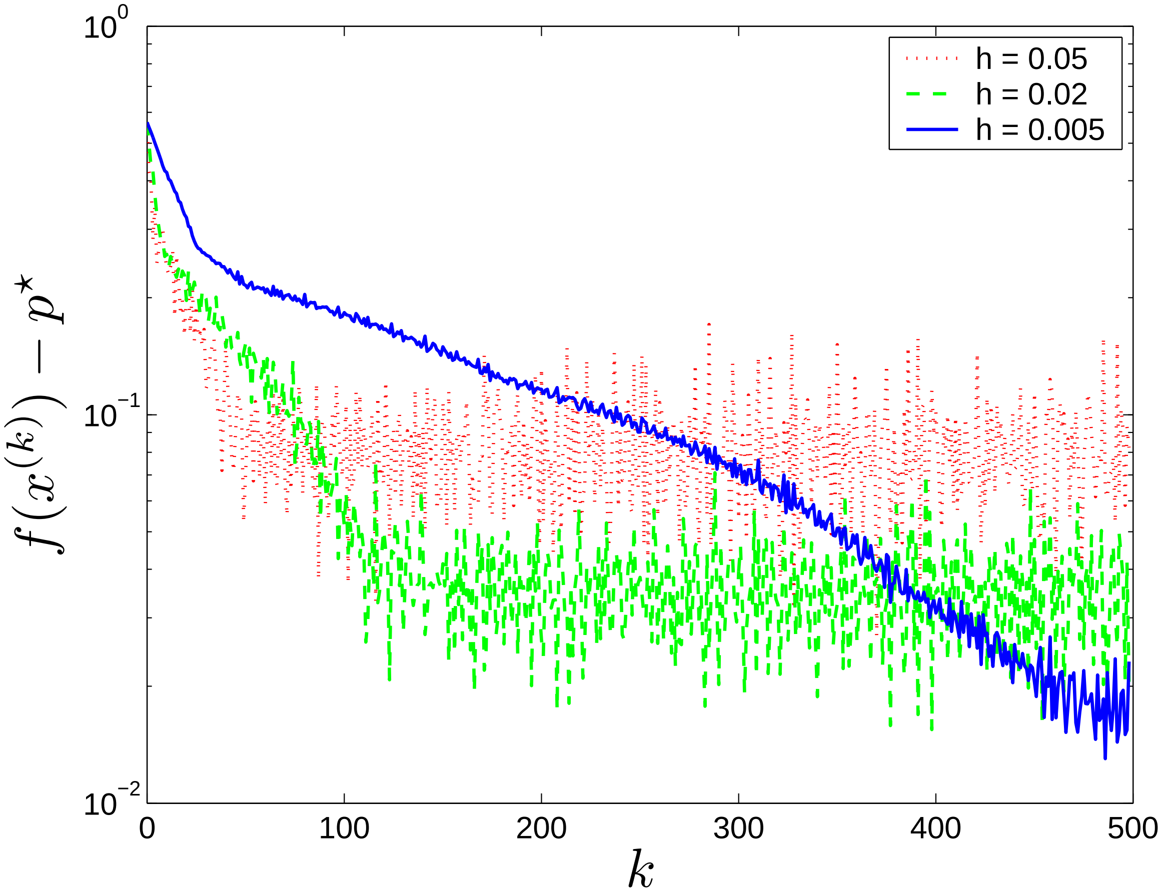
$$x_{k+1} = p_C(x_k + \alpha_k g_k)$$

- Similar complexity analysis
- Some numerical examples on piecewise linear minimization... Problem instance with $n = 10$ variables, $m = 100$ terms

"In theory, there is no difference between theory and practice. In practice, there is..."

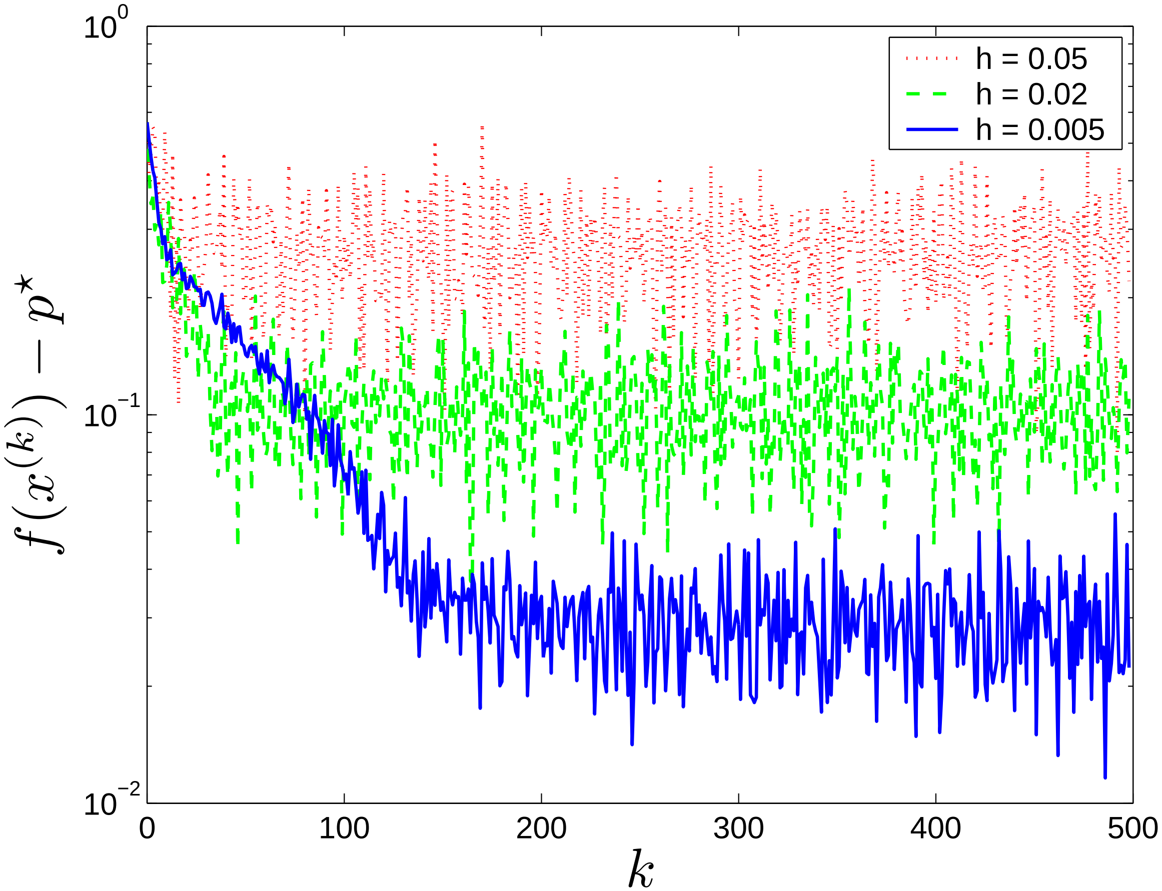
Subgradient Methods: Numerical Examples

Constant step length, $h = 0.05,\ 0.02, 0.005$



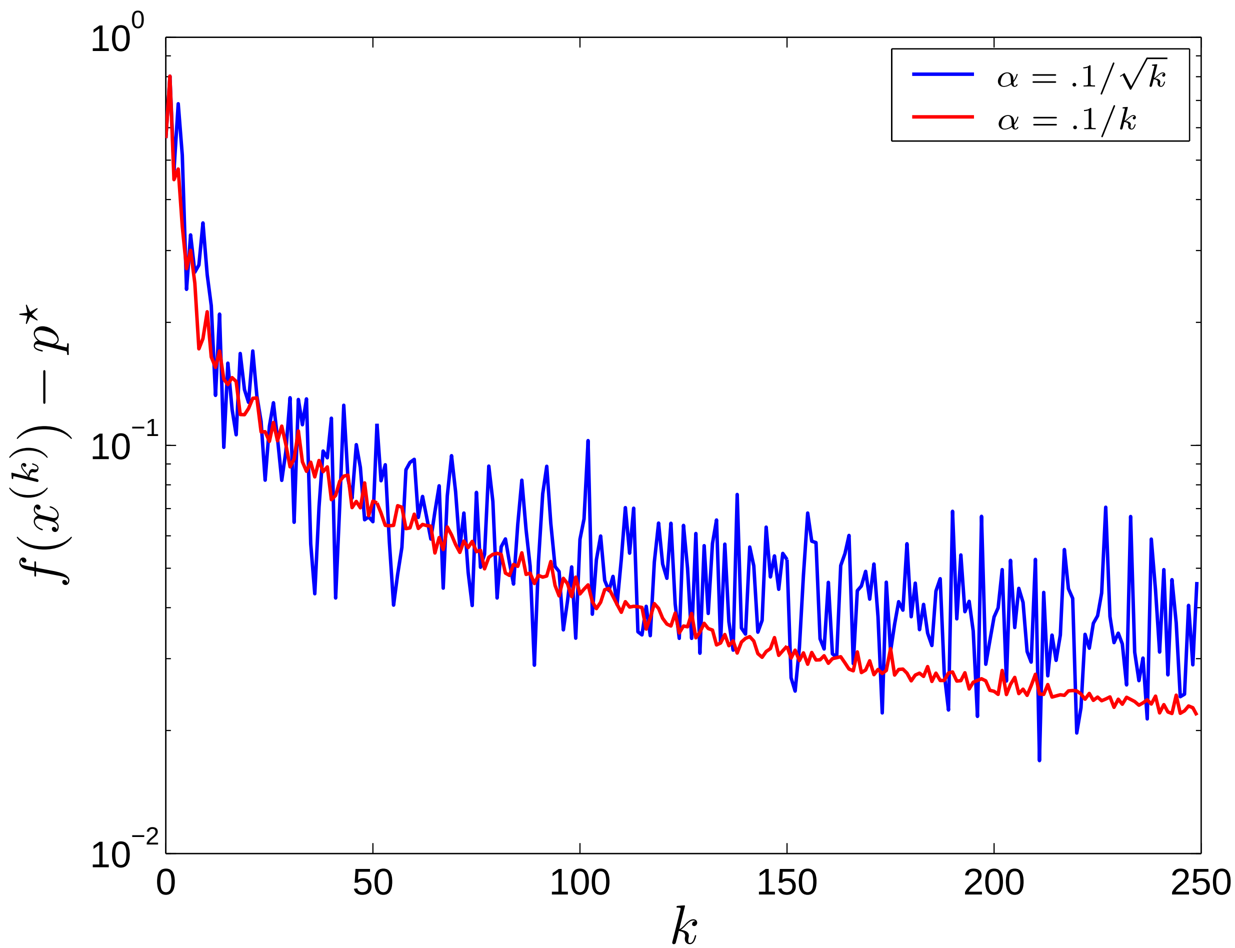
Subgradient Methods: Numerical Examples

Constant step size $h = 0.05,\ 0.02,\ 0.005$



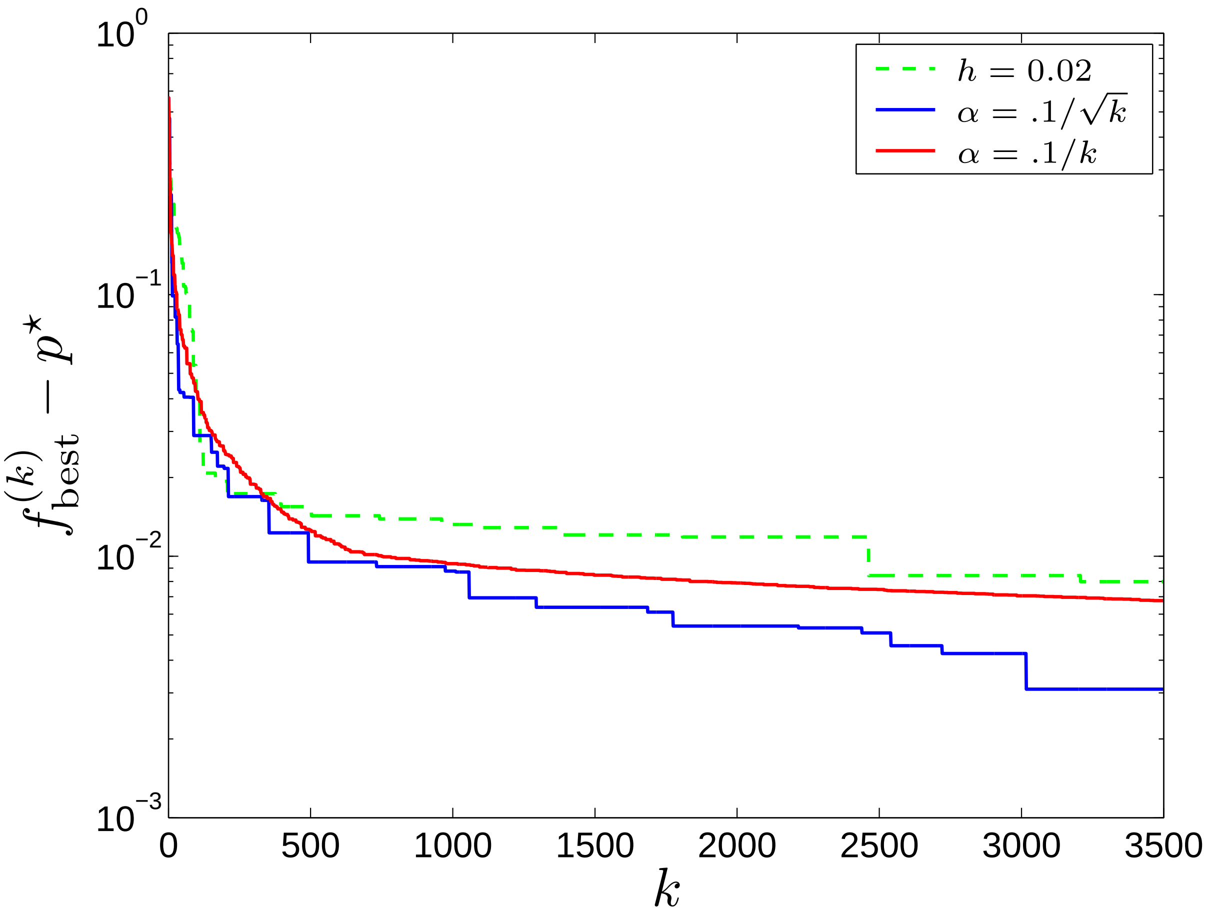
Subgradient Methods: Numerical Examples

Diminishing step rule $\alpha = 0.1/\sqrt{k}$ and square summable step size rule $\alpha = 0.1/k$.



Subgradient Methods: Numerical Examples

Constant step length $h = 0.02$, diminishing step size rule $\alpha = 0.1/\sqrt{k}$, and square summable step rule $\alpha = 0.1/k$



Accelerated Gradient Methods

Accelerated Gradient Methods

Solve

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

in $x \in \mathbb{R}^n$, with $C \subset \mathbb{R}^n$ convex.

- Additional smoothness assumption: the **gradient is Lipschitz** continuous

$$\|\nabla f(x) - \nabla f(y)\|_* \leq L\|x - y\| \quad \text{for all } x, y \in C$$

  where $\|\cdot\|$ is a norm.

- We will also study the case where the function is **strongly convex**, i.e. there exists $\mu > 0$

$$f(y) \geq f(x) + (y - x)^T \nabla f(x) + \frac{\mu}{2}\|y - x\|^2 \quad \text{for all } x, y \in C$$

  where $\|\cdot\|$ is a norm. But acceleration works even when $\sigma = 0$.

Accelerated Gradient Methods

The fact that the gradient $\nabla f(x)$ is Lipschitz continuous

$$\|\nabla f(x) - \nabla f(y)\|_* \leq L\|x - y\| \quad \text{for all } x, y \in C$$

has important algorithmic consequences:

- For any $x, y \in \mathbb{R}^n$,

$$f(y) \leq f(x) + \nabla f(x)^T(y - x) + \frac{L}{2}\|y - x\|^2$$

  and we get a **quadratic upper bound** on the function $f(x)$.
- This means in particular that if $y = x - \frac{1}{L}\nabla f(x)$, then

$$f(y) \leq f(x) - \frac{1}{2L}\|\nabla f(x)\|^2$$

  and we get a guaranteed **decrease in the function value** at each gradient step.

Accelerated Gradient Methods

Suppose we seek to solve

$$\min f(x)$$

over $x \in \mathbb{R}^n$, assuming $\nabla f(x)$ is Lipschitz continuous with constant L.

Consider the following method (due to Adrien Taylor), based on [Nesterov, 1983].

For $k = 1, \ldots, k^{max}$ **iterate**

1. Set $y_{k+1} = (1 - \tau_k) y_k + \tau_k z_k - \alpha_k \nabla f(y_k)$.
2. Set $z_{k+1} = z_k - \gamma_k \nabla f(y_{+1}k)$.

---

where the parameters are set using the value of a time varying sequence A_k

$$\tau_k = \frac{A_{k+1} - A_k}{A_{k+1}},\ \alpha_k = \frac{A_k}{L A_{k+1}},\ \gamma_k = \frac{A_{k+1} - A_k}{L}.$$

Theorem

Convergence. *Let f be L-smooth and convex. For all values $A_k \geq 0$ the iterates satisfy*

$$A_{k+1}(f(y_{k+1}) - f(x_\star)) + \frac{L}{2}\|z_{k+1} - x_\star\|^2 \leq A_k(f(y_k) - f(x_\star)) + \frac{L}{2}\|z_k - x_\star\|^2,$$

if A_k is monotonically increasing and $A_{k+1} - (A_k - A_{k+1})^2 \geq 0$.

Proof. Perform a weighted sum of the following inequalities:

- smoothness and convexity between $x_\star$ and y_{k+1} with weight $\lambda_1 = A_{k+1} - A_k$

$$f(x_\star) \geq f(y_{k+1}) + \langle \nabla f(y_{k+1}); x_\star - y_{k+1}\rangle + \tfrac{1}{2L}\|\nabla f(y_{k+1})\|^2,$$

- smoothness and convexity between y_k and y_{k+1} with weight $\lambda_2 = A_k$

$$f(y_k) \geq f(y_{k+1}) + \langle \nabla f(y_{k+1}); y_k - y_{k+1}\rangle + \tfrac{1}{2L}\|\nabla f(y_{k+1}) - \nabla f(y_k)\|^2.$$

The weighted sum can be written as

$$
\begin{aligned}
0 \geq & \lambda_1[f(y_{k+1}) - f(x_\star) + \langle \nabla f(y_{k+1}); x_\star - y_{k+1}\rangle + \tfrac{1}{2L}\|\nabla f(y_{k+1})\|^2] \\
& + \lambda_2[f(y_{k+1}) - f(y_k) + \langle \nabla f(y_{k+1}); y_k - y_{k+1}\rangle + \tfrac{1}{2L}\|\nabla f(y_{k+1}) - \nabla f(y_k)\|^2],
\end{aligned}
$$

which is equivalently formulated as

$$
\begin{aligned}
& A_{k+1}(f(y_{k+1}) - f(x_\star)) + \frac{L}{2}\|z_{k+1} - x_\star\|^2 \\
\leq\ & A_k(f(y_k) - f(x_\star)) + \frac{L}{2}\|z_k - x_\star\|^2 - \frac{A_k}{2L}\|\nabla f(y_k)\|^2 \\
& - \frac{A_{k+1} - (A_{k+1} - A_k)^2}{2L}\|\nabla f(y_{k+1})\|^2.
\end{aligned}
$$

Therefore, we reach the desired statement as soon as we can remove the last two terms. This means $A_k \geq 0$ and $A_{k+1} - (A_{k+1} - A_k)^2 \geq 0$ (both verified by assumptions). The choice $A_{k+1} = A_k + \frac{1+\sqrt{4A_k+1}}{2}$ allows satisfying $A_{k+1} - (A_{k+1} - A_k)^2 = 0$ with the largest possible value of A_{k+1}. ■

Accelerated Gradient Methods

We get the following result, with a **convergence rate of $O(1/k^2)$.**

Theorem

Complexity. *After k iterations, we obtain points y_k and z_k satisfying*

$$f(y_k) - f(x_\star) \leq \frac{L\|z_0 - x_\star\|^2}{k^2}.$$

Proof. We can pick $A_k = k^2/2$ which satisfies $A_{k+1} - (A_k - A_{k+1})^2 \geq 0$ and, together with the previous theorem, yields the bound above. $\blacksquare$

Accelerated Gradient Methods

The **choice of norm** has a significant impact on complexity. Consider

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

- **Euclidean.** Pick $d(x) = \|x\|_2^2/2$, strongly convex with $\sigma = 1$ w.r.t. the Euclidean norm
$$f(x_k) - f^* \leq \frac{2L_2\|x^*\|_2^2}{(k+1)^2}$$
where L_2 is such that $\|\nabla f(x) - \nabla f(y)\|_2 \leq L_2\|x - y\|_2$, for all $x, y \in C$.

- **Entropy.** Pick $d(x) = \sum_{i=1}^n x_i \log x_i$, strongly convex with $\sigma = 1$ w.r.t. the $\|.\|_1$ norm
$$f(x_k) - f^* \leq \frac{2L_\infty d(x^*)}{(k+1)^2}$$
where L_e is such that $\|\nabla f(x) - \nabla f(y)\|_\infty \leq L_\infty\|x - y\|_1$, for all $x, y \in C$.

Because $\|\cdot\|_\infty \leq \|\cdot\|_2 \leq \|\cdot\|_1$, we always have $L_\infty \leq L_2$.

Accelerated Gradient Methods: optimality

Accelerated gradient methods. Can we do better than $O(1/\sqrt{\epsilon})$?

Problem class. $f(x)$ has a Lipschitz continuous gradient with constant L. At each iteration, we get a **black-box gradient oracle**, and we look for a solution satisfying $f(x) - f^* \leq \epsilon$

If we know nothing about $f(x)$ except its gradient at certain points and its gradient Lipschitz constant L.

- We need at least $O(\|x_0 - x^*\|_2\sqrt{L/\epsilon})$ iterations.
- We can construct an explicit quadratic function reaching this bounds, which is hard for all schemes.

Accelerated Gradient Methods: optimality

Definition

Iterative method. *We will assume that an iterative method generates a sequence of points y_k such that*

$$y_k \in \mathcal{L}_k \triangleq y_0 + \textbf{span}\{\nabla f(y_0), \nabla f(y_1), \ldots, \nabla f(y_{k-1})\}$$

This can be relaxed, but simplifies analysis and covers most classical algorithms.

Accelerated Gradient Methods: optimality

Proof structure.

- Design a set of (quadratic) functions $f_n(x)$ whose gradients at sparse points have only one more nonzero coefficient.
- Without loss of generality, we can always start at $y_0 = 0$.
- Starting at $y_0 = 0$, any iterate y_k will have at most cardinality k, **whatever the algorithm**.
- These iterates poorly approximate the optimum, which has cardinality n.

Accelerated Gradient Methods: optimality

We write $S_{k,n} \triangleq \{x \in \mathbb{R}^n : x_i = 0,\ i = k+1, \ldots, n\}$.

Lemma

Worst function in class. *[Nesterov, 2003, §2.1.2] Define*

$$f_k(x) \triangleq \frac{L}{8}\left(x_1^2 + \sum_{i=1}^{k-1}(x_i - x_{i+1})^2 + x_k^2 - 2x_1\right)$$

then for any sequence $y_i \in \mathbb{R}^n$, $i = 0, \ldots, p$, *such that*

$$y_k \in \mathcal{L}_k \triangleq y_0 + \textit{span}\,\{\nabla f_p(y_0), \nabla f_p(y_1), \ldots, \nabla f_p(y_{k-1})\}$$

we have $y_k \in S_{k,n}$.

Proof. We can write

$$\begin{array}{rcl} 0 & \leq & \frac{L}{4}\left(s_1^2+\sum_{i=1}^{k-1}(s_i-s_{i+1})^2+s_k^2\right) \\ & \leq & s^T\nabla^2 f(x)s \\ & \leq & \frac{L}{4}\left(s_1^2+\sum_{i=1}^{k-1}2(s_i^2+s_{i+1}^2)+s_k^2\right) \leq L\sum_{i=1}^{n}s_i^2 \end{array}$$

which means $0 \preceq \nabla^2 f_k(x) \preceq L\,\mathbf{I}_n$, hence $\nabla f_k(x)$ **is Lipschitz continuous** with constant L, because $\nabla^2 f_k(x) = \frac{L}{4}A_k$ with

$$A_k = \left(\begin{array}{cc} B & 0 \\ 0 & 0 \end{array}\right) \quad \text{where} \quad B_k = \left(\begin{array}{cccc} 2 & -1 & \cdots & 0 \\ -1 & 2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & -1 \\ 0 & \cdots & -1 & 2 \end{array}\right)$$

where A_k is block tridiagonal with an upper left block of dimension $B_k \in \mathbf{S}_k$. By induction now, $\nabla f_p(x_0) = (L/4)e_1 \in S_{1,n}$ and assuming $y \in S_{k,n}$, then $\nabla f_p(y) = (L/2)(A_k y - e_1) \in S_{k+1,n}$ **because A_k is tridiagonal**. ∎

Accelerated Gradient Methods: optimality

Theorem

Worst-case complexity. *For any $1 \leq k \leq (n-1)/2$, there exists a function $f(X)$ with $\nabla f(x)$ L-Lipschitz continuous, such that for any iterative method (cf. above) we have*

$$f(y_k) - f^* \geq \frac{3L\|y_0 - y^*\|^2}{32(k+1)^2}$$

and

$$\|y_k - y^*\|^2 \geq \frac{1}{8}\|y_0 - y^*\|^2.$$

Proof. Without loss of generality, we can assume that $y_0 = 0$, otherwise we simply shift the function without changing its nature. We will apply an iterative method to the function $f(x) \triangleq f_{2k+1}(x)$. Let us first note that the minimizer of $f(x)$, solving

$$\nabla f_k(x) = A_k x - e_1 = 0$$

is given by

$$y^* = \begin{cases} 1 - \frac{i}{2k+1}, & i = 1, \ldots, 2k+1, \\ 0 & i = k+1, \ldots, n. \end{cases}$$

and

$$f^*_{2k+1} = \frac{L}{8}\left(\frac{1}{2k+2} - 1\right). \tag{3}$$

and

$$\|y^*\|^2 = \sum_{i=1}^{2k+1}\left(1 - \frac{i}{2k+1}\right)^2 \leq \frac{1}{3}(2k+2) \tag{4}$$

using

$$\sum_{i=1}^{k} i = \frac{k(k+1)}{2} \quad \text{and} \quad \sum_{i=1}^{k} i^2 = \frac{k(k+1)(2k+1)}{6} \leq \frac{(k+1)^3}{3}$$

From the form of $f_p(x)$ we have $f_p(x) = f_k(x)$ **whenever** $x \in S_{k,n}$ **and** $p \geq k$, hence in particular,

$$f(y_k) \triangleq f_{2k+1}(y_k) = f_k(y_k) \geq f^* = \frac{L}{8}\left(\frac{1}{k+1} - 1\right),$$

in view of (3) and (4), with $y_0 = 0$, $f^* \triangleq f^*_{2k+1}$ we get

$$\frac{f(y_k) - f^*}{\|y_0 - y^*\|^2} \geq \frac{\frac{L}{8}(-1 + \frac{1}{k+1} + 1 - \frac{1}{2k+2})}{(2k+2)/3} = \frac{3L}{32(k+1)^2}$$

which is the first inequality. Since $y_k \in S_{k,n}$ we have

$$\|y_k - y^*\|^2 \geq \sum_{i=k+1}^{2k+1} (\bar{y}^*_{2k+1,i})^2 = \sum_{i=k+1}^{2k+1} \left(1 - \frac{i}{2k+2}\right)^2$$

hence, with $y_0 = 0$ and using again (4)

$$\begin{aligned}
\|y_k - y^*\|^2 &\geq \frac{2k^2 + 7k + 6}{24k + 1} \\
&\geq \frac{2k^2 + 7k + 6}{16(k+1)^2} \|y_0 - \bar{y}^*_{2k+1}\|^2 \\
&\geq \frac{1}{8} \|y_0 - \bar{y}^*_{2k+1}\|^2
\end{aligned}$$

because

$$\frac{2k^2 + 7k + 6}{16(k+1)^2} \geq \frac{1}{8} \|y_0 - \bar{y}^*\|^2$$

for all $k \geq 0$ and $y^* \triangleq \bar{y}^*_{2k+1}$. ■

Gradient/projection methods for stochastic problems

Stochastic Optimization

Solve

$$\begin{array}{ll} \text{minimize} & \phi(x) \triangleq \mathbf{E}[f(x,\xi)] \\ \text{subject to} & x \in C, \end{array}$$

in $x \in \mathbb{R}^n$, where C is a simple convex set. The key difference here is that the function we are minimizing is **stochastic.**

- **Batch method.** A simple option is to approximate the problem by

$$\begin{array}{ll} \text{minimize} & \sum_{i=1}^m f(x,\xi_m) \\ \text{subject to} & x \in C, \end{array}$$

  where ξ_i are sampled from the distribution of ξ.
- Sampling is costly, the full batch is heavy, we can do better. . .

Stochastic Optimization

Assume we have an unbiased estimate $g(x, \xi)$ of the subgradient of $\phi(x)$, i.e.

- $\mathbf{E}[g(x, \xi)|x] = g(x) \in \partial\phi(x)$
- In particular
$$\phi(y) \geq \phi(x) + g(x)^T(y - x)$$

Stochastic Optimization

Let $p_C(\cdot)$ be the Euclidean projection operator on C.

Algorithm (Robust stochastic averaging)

- Choose $x_0 \in C$ and a step sequence $\gamma_j > 0$.
- **For** $k = 1, \ldots, k^{max}$ **iterate**
  1. Compute a subgradient
  $$g \in \partial f(x_k, \xi_k)$$
  2. Update the current point
  $$x_{k+1} = p_C(x_k - h_k g)$$
  3. Compute
  $$\bar{x} = \frac{\sum_{k=0}^{N-1} h_k x_k}{\sum_{k=0}^{N-1} h_k}$$

Convergence proof.

Theorem

Complexity. *Suppose* $\|x^\star - x_0\| \leq R$ *for some* $x_0 \in C$, *and* $\mathbf{E}[\|g\|_2^2] \leq L^2$, *then*

$$\mathbf{E}[f(\bar{x})] - \min_{x \in C} \mathbf{E}[f(x,\xi)] \leq \frac{R^2 + L^2 \sum_{k=0}^{N-1} h_k^2}{2 \sum_{k=0}^{N-1} h_k}$$

Proof. Let x^* be an optimal solution and define $r_k = \|x^* - x_k\|$. Since x_{k+1} is the projection of $x_k - h_k g_k$ over C, it satisfies

$$\begin{array}{rcl} r_{k+1}^2 & \leq & \|x_k - h_k g_k - x^*\|^2 \\ & = & r_k^2 - 2h_k \langle g_k, x_k - x^* \rangle + h_k^2 \|g_k\|^2 \end{array}$$

because x_{k+1} must be closer to $x^* \in C$ than $x_k - h_k g_k$.

Taking expectations, we get, by convexity and because ξ_k and x_k are independent.

$$\begin{aligned}
\mathbf{E}[r_{k+1}^2] &\leq \mathbf{E}[r_k^2] - 2h_k\,\mathbf{E}[\langle g_k, x_k - x^*\rangle] + h_k^2\,\mathbf{E}[\|g_k\|^2] \\
&\leq \mathbf{E}[r_k^2] - 2h_k\,\mathbf{E}[\langle \mathbf{E}[g_k|x_k], x_k - x^*\rangle] + h_k^2 L^2 \\
&\leq \mathbf{E}[r_k^2] - 2h_k(\mathbf{E}[\phi(x_k)] - \phi(x^*)) + h_k^2 L^2
\end{aligned}$$

Summing all these inequalities and using the convexity of $\phi(\cdot)$, we finally get

$$\begin{aligned}
r_0^2 + L^2 \sum_{k=0}^{N-1} h_k^2 &\leq \sum_{k=0}^{N-1} h_k(\mathbf{E}[\phi(x_k)] - \phi(x^*)) \\
&\leq 2\left(\sum_{k=0}^{N-1} h_k\right)(\mathbf{E}[\phi(\bar{x})] - \phi(x^*))
\end{aligned}$$

hence the desired result. ■

Stochastic Optimization

Complexity.

- If we set $h_k = R/(L\sqrt{N})$, we have

$$\mathbf{E}[f(\bar{x}) - f^*] \leq \frac{LR}{\sqrt{N}}$$

- Furthermore, if we assume

$$\mathbf{E}\left[\exp\left(\frac{\|g\|_2^2}{L^2}\right)\right] \leq e, \quad \text{for all } g \in \partial f(x_k, \xi) \text{ and } x \in C$$

we get

$$\mathbf{Prob}\left[\phi(\tilde{x}_k) - \phi^* \geq \frac{LR}{\sqrt{N}}(12 + 2t)\right] \leq 2\exp(-t).$$

*

References

Y. Nesterov. A method of solving a convex programming problem with convergence rate $O(1/k^2)$. *Soviet Mathematics Doklady*, 27(2): 372–376, 1983.

Y. Nesterov. *Introductory Lectures on Convex Optimization*. Springer, 2003.