Sharpness, Restart, Acceleration

Vincent Roulet, Alexandre d'Aspremont

OSL, Les Houches, April 13, 2017

Motivation

- **Goal :**

$$\text{minimize} \quad f(x), \quad f : \mathbb{R}^n \to \mathbb{R} \text{ cvx}$$

- Some algorithms use past information to build next iterate
  - Accelerated Gradient Method
  - Universal Fast Gradient Method
  - Quasi-Newton methods
  - ...
- **Idea :** Refresh algorithms when past information is "no longer relevant"
- Doesn't make any sense for gradient descent with line search for example

How to characterize past information ?

- Take an algorithm $\mathcal{A}$ that outputs points $x = \mathcal{A}(x_0, \theta, t)$, where
  - x_0 is the initial point,
  - θ are parameters of the algorithm
  - t is the number of iterations.
- Look at the convergence rate

$$f(x) - f^* \leq \frac{cd(x_0, X^*)^q}{t^p}$$

  where
  - $d(x_0, X^*)$ is the Euclidean distance from x_0 to the set of minimizers X^*
  - c, p, q are constants depending on the problem
- Bound increases with $d(x_0, \mathcal{X}^*)$, intuition :

x_0 close to $X^* \rightarrow$ good initialization so fast convergence

- Exploit information on $d(x_0, X^*)$?

Plan

Sharpness

Definition

A function f satisfies the sharpness property on a set K if there exists $r \geq 1$, $\mu > 0$, s.t.

$$\mu d(x, X^*)^r \leq f(x) - f^*, \quad \text{for every } x \in K \qquad \text{(Sharp)}$$

Examples

- Strongly convex function ($r = 2$)
- Gradient dominated functions ($r = 2$)
- Matrix game problems like $\min_x \max_y x^T Ay$ ($r = 1$)
- Real analytic functions (r unknown)
- Subanalytic functions (r unknown)

Sharpness for real analytic function

For f real analytic, $x \in \mathbb{R}$ and $x^* \in X^*$,

$$f(x) - f^* = \sum_{k=q}^{\infty} \frac{f^{(k)}(x^*)}{k!}(x - x^*)^k$$

where $q \geq 0$ is the smallest coefficient for which $f^{(q)}(x^*) \neq 0$.
There is an interval V around x^* s.t.

$$\frac{1}{2}\frac{f^{(q)}(x^*)}{q!}|x - x^*|^q \leq f(x) - f^*$$

Setting $x^* = \Pi_{X^*}(x)$ this yields (Sharp) on V with q and $\frac{1}{2}\frac{f^{(q)}(x^*)}{q!}$.

Sharpness for subanalytic functions

Łojasevicz inequality

- Sharpness property is known to be satisfied for real analytic functions as the Łojasevicz inequality [Łojasevicz 1963]
- Generalized recently to broad class of non-smooth convex functions called subanalytic [Bolte et al 2007].
- Subanalytic functions are functions whose epigraph can be expressed as a semi-analytic manifold.
- Proofs rely on topological arguments so (r, μ) are mostly unknown.

Smoothness

Definition

A function f satisfies the smoothness property on a set J if there exists $s \in [1,2]$, $L > 0$ s.t.

$$\|\nabla f(x) - \nabla f(y)\|_2 \leq L\|x-y\|_2^{s-1}, \quad \text{for every } x, y \in J \quad \text{(Smooth)}$$

Examples

- Non-smooth ($s = 1$)
- Smooth ($s = 2$)
- Hölder smooth ($s \in (1,2)$)

Sharpness and smoothness

If f satisfies (Smooth), for every $x \in \mathbb{R}^n$ and $y = \Pi_{X^*}(x)$,

$$f(x) \leq f(y) + \nabla f(y)^T(x-y) + \frac{L}{s}\|x-y\|_2^s = f^* + \frac{L}{s}d(x,X^*)^s$$

Combined with (Sharp), $\mu d(x,X^*)^r \leq f(x) - f^*$, this yields

$$0 < \frac{s\mu}{L} \leq d(x,X^*)^{s-r}$$

Taking $x \to X^*$, necessarily

$$\boldsymbol{s \leq r}$$

Moreover if $\boldsymbol{s} < \boldsymbol{r}$, last inequality can **only be valid on a bounded set**, either smoothness or sharpness or both are not valid in the whole space.

Condition numbers

We denote

$$\tau = 1 - \frac{s}{r}$$

a condition number on the ratio of powers, s.t.

$$0 \leq \tau < 1$$

and

$$\kappa = L^{\frac{2}{s}} / \mu^{\frac{2}{r}}$$

a generalized condition number.

Plan

General strategy

- Take an algorithm $\mathcal{A}$ that outputs points $x = \mathcal{A}(x_0, \theta, t)$, where
  - x_0 is the initial point,
  - θ are parameters of the algorithm
  - t is the number of iterations
- Look at the convergence rate if f satisfies (Sharp)

$$f(x) - f^* \leq \frac{c d(x_0, X^*)^q}{t^p} \leq \frac{c'(f(x_0) - f^*)^{q/r}}{t^p}$$

- Given $\gamma \geq 0$, compute analytically t s.t.

$$f(x) - f^* \leq e^{-\gamma}(f(x_0) - f^*)$$

- Iterate and compute total complexity

General formulation

Given an algorithm $\mathcal{A}$ that outputs points $x = \mathcal{A}(x_0, \theta, t)$

Scheduled restart schemes :

Inputs: x_0, sequence θ_k, sequence t_k
for $k = 1 \ldots R$ **do**
$x_k = \mathcal{A}(x_{k-1}, \theta_k, t_k)$
end for
Output: $\hat{x} = x_R$

General analysis

Lemma

Given $\gamma \geq 0$, suppose setting

$$t_k = Ce^{\alpha k}, \qquad \text{with } C > 0,\ \alpha \geq 0,$$

ensures

$$f(x_k) - f^* \leq Me^{-\gamma k}, \qquad \text{with } M > 0.$$

Writing $N = \sum_{k=1}^{R} t_k$ the total number of iterations, we get

$$f(\hat{x}) - f^* \leq M \exp(-\gamma C^{-1} N), \qquad \text{when } \alpha = 0,$$

$$f(\hat{x}) - f^* \leq \frac{M}{(\alpha e^{-\alpha} C^{-1} N + 1)^{\frac{\gamma}{\alpha}}}, \quad \text{when } \alpha > 0.$$

Smooth convex problems

- If f is cvx and smooth $(s = 2, L)$, an optimal algorithm is the Accelerated Gradient $\mathcal{A}cc$.
- Given x_0, it outputs after t iterations, a point $x = \mathcal{A}cc(x_0, t)$, s.t.

$$f(x) - f^* \leq \frac{cL}{t^2} d(x_0, X^*)^2,$$

  where c is a universal constant.
- Assume that f satisfies (Sharp) with (r, μ) on a set K

$$\mu d(x, X^*)^r \leq f(x) - f^*, \quad \text{for every } x \in K$$

- Assume we are given $x_0 \in \mathbb{R}^n$, s.t. $\{x,\ f(x) \leq f(x_0)\} \subset K$.

Optimal scheme

Proposition 1st part

Assume f cvx, smooth $(s = 2, L)$ and sharp (r, μ) on a set K.
Run scheduled restarts of $\mathcal{A}cc$ with

$$t_k = C_{\tau,\kappa} e^{\tau k}$$
$$C_{\tau,\kappa} = e^{1-\tau}(c\kappa)^{\frac{1}{2}}(f(x_0) - f^*)^{-\frac{\tau}{2}}$$

Then for every outer iteration $k \geq 0$,

$$f(x_k) - f^* \leq e^{-2k}(f(x_0) - f^*).$$

Optimal scheme

Proposition

Denote N the total number of iterations at the output $\hat{x}$, then, when $\tau = 0$,

$$f(\hat{x}) - f^* \leq \exp\left(-2e^{-1}(c\kappa)^{-\frac{1}{2}}N\right)(f(x_0) - f^*) = O\left(\exp(-\kappa^{-\frac{1}{2}}N)\right),$$

while, when $\tau > 0$,

$$f(\hat{x}) - f^* \leq \frac{f(x_0) - f^*}{\left(\tau e^{-1}(f(x_0) - f^*)^{\frac{\tau}{2}}(c\kappa)^{-\frac{1}{2}}N + 1\right)^{\frac{2}{\tau}}} = O\left(\kappa^{\frac{1}{\tau}}N^{-\frac{2}{\tau}}\right),$$

Note : Optimal for this class of problems [*Optimal methods of smooth convex optimization*, A. Nemirovski, Y. Nesterov 1985]

Adaptive scheme

- In practice (r, μ) are unknown
- Given a fixed total number of iterations N, run following schemes

$$\mathcal{S}_{i,j} : \text{Scheduled restart with } t_k = C_i e^{\tau_j k}, \text{ where } C_i = 2^i \text{ and } \tau_j = 2^-$$

with $i \in [1, \ldots, \lfloor \log_2 N \rfloor]$, $j \in [0, \ldots, \lceil \log_2 N \rceil]$

- Optimal bounds up to constant factor 4
- Has a complexity $\log_2(N)^2$ higher than running N iterations in the optimal scheme
- **Adaptive** algorithm

Non-smooth or Hölder smooth convex problems

- If f is cvx, satisfies (Smooth) with (s, L) on a set J, i.e.

$$\|\nabla f(x) - \nabla f(y)\|_2 \leq L\|x - y\|_2^{s-1}, \quad \text{for every } x, y \in J, \tag{Smooth}$$

  an optimal algorithm is the Fast Universal Gradient method $\mathcal{U}$ by Nesterov, 2015.

- Given ϵ, x_0, it outputs, after t iterations, a point $x = \mathcal{U}(x_0, \epsilon, t)$ s.t.

$$f(x) - f^* \leq \frac{\epsilon}{2} + \frac{cL^{\frac{2}{s}} d(x_0, X^*)^2}{\epsilon^{\frac{2}{s}} t^{\frac{2\rho}{s}}} \frac{\epsilon}{2}$$

  where

$$\rho = \frac{3s - 2}{2}$$

  is the optimal rate for this class of functions.

Hölder smooth convex problems strategy

- Assume that we have access to $\epsilon_0 \geq f(x_0) - f^*$ for a given $x_0 \in \mathbb{R}^n$
- Given $\gamma \geq 0$ run scheduled restarts with sequence of target accuracies

$$\epsilon_k = e^{-\gamma k}\epsilon_0$$

- Choose t_k to ensure

$$f(x_k) - f^* \leq \epsilon_k$$

Optimal scheme

Proposition 1st part

Assume f cvx, Hölder smooth (s, L) and sharp (r, μ) on a set K.
Run scheduled restarts of $\mathcal{U}$ with

$$\epsilon_k = e^{-\rho k}\epsilon_0 \qquad t_k = C_{\tau,\kappa,\rho}e^{\tau k}$$

$$C_{\tau,\kappa,\rho} = e^{1-\tau}(c\kappa)^{\frac{s}{3s-2}}\epsilon_0^{\frac{\tau}{\rho}}$$

Then for every outer iteration $k \geq 0$,

$$f(x_k) - f^* \leq e^{-\rho k}\epsilon_0.$$

Optimal scheme

Proposition 2nd part

Denote N the total number of iterations at the output $\hat{x}$, then, when $\tau = 0$,

$$f(\hat{x}) - f^* \leq \exp\left(-\rho e^{-1}(c\kappa)^{-\frac{s}{2\rho}} N\right)\epsilon_0 = O\left(\exp(-\kappa^{-\frac{s}{2\rho}} N)\right),$$

while, when $\tau > 0$,

$$f(\hat{x}) - f^* \leq \frac{\epsilon_0}{\left(\tau e^{-1}(c\kappa)^{-\frac{s}{2\rho}} \epsilon_0^{\frac{\tau}{\rho}} N + 1\right)^{\frac{\rho}{\tau}}} = O\left(\kappa^{\frac{s}{2\tau}} N^{-\frac{\rho}{\tau}}\right),$$

Note : Optimal for this class of problems [*Optimal methods of smooth convex optimization*, A. Nemirovski, Y. Nesterov 1985]

General convex problems

- 3 parameters for the schedule γ, C, α
- Grid search inefficient if r or s unknown
- Otherwise grid search on C works
- Can be used for
  $\rightarrow$ non-smooth ($s = 1$), gradient dominated functions ($r = 2$)
  $\rightarrow$ non-smooth ($s = 1$), sharp functions ($r = 1$)

Plan

Strategy

- Assume f^* known (e.g. zero sum-game matrix problem, projection a convex set...)
- Given an accuracy ϵ, denote t_ϵ the number of iterations to observe that $x = \mathcal{U}(x_0, \epsilon, t_\epsilon)$ satisfies

$$f(x) - f^* \leq \epsilon$$

$\rightarrow$ Stop when target accuracy reached
$\rightarrow$ Restart with a reduced target accuracy

Formulation

Given the Fast Universal Gradient method $\mathcal{U}$ that outputs $x = \mathcal{U}(x_0, \epsilon, t)$

Restarts with termination criterion :

Inputs: x_0, γ, f^*
$\epsilon_0 = f(x_0) - f^*$
for $k = 1 \dots R$ **do**
  $\epsilon_k = e^{-\gamma}\epsilon_{k-1}$
  $x_k = \mathcal{U}(x_{k-1}, \epsilon_k, t_{\epsilon_k})$
end for
Output: $\hat{x} = x_R$

Restarts with termination criterion

Assume f cvx, Hölder smooth (s, L) and sharp (r, μ) on a set K.
Run restarts with termination criterion with $\gamma = \rho$.
Denote N the total number of iterations at the output $\hat{x}$, then, when $\tau = 0$,

$$f(\hat{x}) - f^* \leq \exp\left(-\rho e^{-1}(c\kappa)^{-\frac{s}{2\rho}} N\right)\epsilon_0 = O\left(\exp(-\kappa^{-\frac{s}{2\rho}} N)\right),$$

while, when $\tau > 0$,

$$f(\hat{x}) - f^* \leq \frac{\epsilon_0}{\left(\tau e^{-1}(c\kappa)^{-\frac{s}{2\rho}}\epsilon_0^{\frac{\tau}{\rho}} N + 1\right)^{\frac{\rho}{\tau}}} = O\left(\kappa^{\frac{s}{2\tau}} N^{-\frac{\rho}{\tau}}\right),$$

Note : Restarts robust to the choice of γ.
Taking $\gamma = 1$ is optimal up to a small constant factor.

Plan

General setting

- Extension to

$$\text{minimize } f(x) = \phi(x) + g(x)$$

where

  - ϕ satisfies (Smooth) w.r.t a generic norm $\|.\|$.

$$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|^{s-1}, \quad \text{for every } x, y \in J, \tag{Smooth}$$

  - we have access to a prox function h 1-strongly convex w.r.t. $\|.\|$ defining a Bregman divergence

$$D_h(z; x) = h(z) - h(x) - \nabla h(x)^T(z - x)$$

  - g is simple in the sense that we can easily solve

$$\min_z y^T z + g(z) + \lambda D_h(z; x)$$

- Covers a whole class f of problems such as sparse or constrained.
- Need an appropriate notion of sharpness w.r.t $\|.\|$.

Relative sharpness

Definition

A convex function f is called relatively sharp with respect to a strictly convex function h on a set $K \subset \mathrm{dom}(f)$ if there exists $r \geq 1$, $\mu > 0$ such that

$$2\mu D_h(x; X^*)^{\frac{r}{2}} \leq f(x) - f^* \quad \text{for any } x \in K \quad \text{(Relative Sharpness)}$$

where $D_h(x; X^*) = \min_{x^* \in X^*} D_h(x; x^*)$ and D_h is the Bregman divergence associated to h.

Plan

Numerical Experiments

- Classification problems on UCI *Sonar* data set with various losses.
- Check convergence of best method found by grid search **Adap**
- Compare against
  - Gradient descent **Grad**
  - Accelerated gradient descent **Acc**
  - Restarts enforcing monotonicity **Mono**,
    i.e., when $f(x_{k+1}) \leq f(x_k)$ in the inner iterations.

Least Squares and Logistic

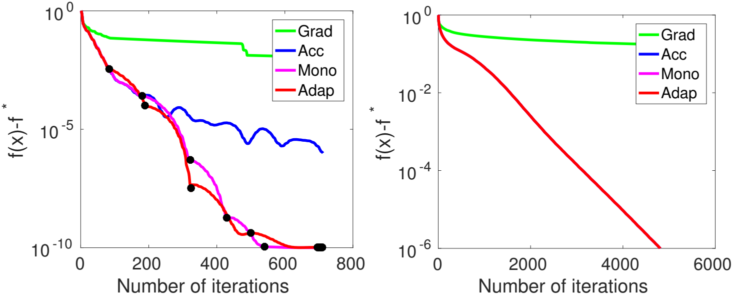

Figure: Least squares loss (left) and Logistic loss (right). Large dots represent restart iterations

Lasso and Dual SVM

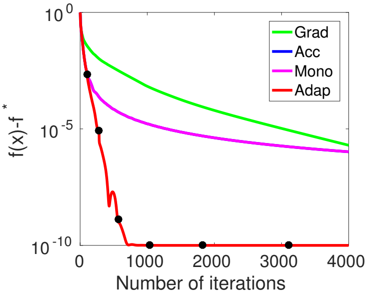

Figure: Lasso (left) and dual SVM (right) problems.
Large dots represent restart iterations

Plan

Contributions

- Open the black box model by adding a generic assumption on the behavior of the function around minimizers
- Convergence analysis of restart schemes
- Optimal schemes for smooth, Hölder smooth, non-smooth convex optimization
- Adaptive scheme for smooth convex optimization

Future work

Sharpness analysis

- Sharpness reads

$$\mu d(x, X^*)^r \leq f(x) - f^*, \quad \text{for every } x \in K$$

- μ depends generally on K, thorough analysis in
  From error bounds to the complexity of first-order descent methods for convex functions, J. Bote et al, 201
- Local adaptivity of restart schemes ?
- If f^* known, restart with termination criterion is adaptive.
  $\rightarrow$ Approximate f^* ?

Practical algorithm

- Grid search shows robustness but not very practical
- Restarting from a combination of points, see
  Restarting accelerated gradient methods with a rough strong convexity estimate, O. Fercoq, Z. Qu, 2016

Thanks !

Questions ?