Lecture 13: Lower bounds in convex optimization

Lecturer: Yanjun Han

May 10, 2021

Lecture plans for selected topics

- May 10: compression-based arguments in convex optimization
- May 12: privacy/communication constrained estimation
- May 17: scandiction problem (Tsachy)
- May 19: geometric/functional arguments in network information theory (Ayfer)
- May 24: min-max vs. max-min approaches
- May 26: adaptation lower bounds

Today's plan

Lower bounds for convex optimization without any noise

- linear convergence (compressing the gradient)
- dimension-independent convergence (zero-respecting algorithms)
- quadratic optimization (polynomial approximation)

Problem I: linear convergence under first-order oracle

- the learner is given a function class

$$\mathcal{F} = \{f : [-1,1]^d \to [-1,1], f \text{ convex and 1-Lipschitz}\}$$

- at each time $t = 1, \cdots, T$, learner queries x_t, and receives $(f(x_t), \nabla f(x_t))$ from the oracle
- target: learner aims to find $\widehat{x}$ to achieve a small suboptimality gap

$$L(f, \widehat{x}) = f(\widehat{x}) - \min_{x^\star} f(x^\star)$$

- equivalently, learner aims to find

$$\mathsf{Compl}(\varepsilon) \triangleq \min\{T : \exists\ \widehat{x}_T \text{ s.t. } \sup_f L(f, \widehat{x}_T) \le \varepsilon\}$$

Claim

$$\frac{d \log(1/\varepsilon)}{\log(d \log(1/\varepsilon))} \lesssim \mathsf{Compl}(\varepsilon) \lesssim d \log(1/\varepsilon)$$

(Upper bound achieved by ellipsoid method)

Warm-up case: $d = 1$

Claim

For $d = 1$, it holds that

$$\mathrm{Compl}(\varepsilon) \asymp \log(1/\varepsilon)$$

(Upper bound achieved by bisection method)

High-level idea:

- given queries $x_1, \cdots, x_T$, the learner couldn't distinguish between functions f_1 and f_2 if their values and gradients agree on all points
- tempted to apply two-point method to achieve full ambiguity
- problem: the queries $x_1, \cdots, x_T$ depend on the function
- solution: construct a tree of functions

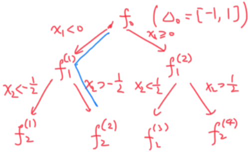
Tree construction

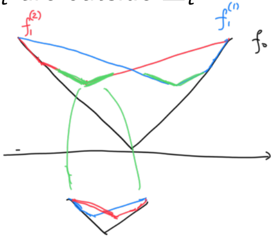
target: given any causal sequence $x_1, \cdots, x_T$, construct a sequence of functions $f_t \in \mathcal{F}$ such that

- f_t has an "active region" Δ_t such that

$$\min_{x \notin \Delta_t} f_t(x) - \min_x f_t(x) \geq 2^{-O(t)}$$

- the first t points $x_1, \cdots, x_t$ applied to f_t are outside Δ_t





General dimension: compression-based argument

- special case: argue that $\Omega(d/\log d)$ steps are needed to decrease the suboptimality gap by half
- compression-based idea: carefully find $\mathcal{F}_0 \subseteq \mathcal{F}$ such that restricting to $\mathcal{F}_0$, the oracle only provides a finite amount of information

$$\underbrace{\mathcal{O}(f,x)}_{\text{oracle info.}} = \mathcal{R}(\overbrace{\underbrace{\mathcal{I}(f,x)}_{\text{compressed info}}}^{\in\{I_1,\dots,I_K\}}, x), \qquad f \in \mathcal{F}_0, x \in [0,1]^d.$$

- separation condition: the ε-optimal solution of $f \in \mathcal{F}_0$ is pairwise disjoint

$\{x : f(x) - \min_{x^*} f(x^*) < \epsilon\}$.

Compression lemma

Assume that $\mathcal{I}$ only takes K possible values, then

$$\mathrm{Compl}(\varepsilon) \geq \frac{\log |\mathcal{F}_0|}{\log K}$$

The construction for $\varepsilon = 1$

- one possible construction: for $v \in \{\pm 1\}^d$, define

$$f_v(x) = \max_{i \in [d]} v_i x_i$$

$\min_x f_v(x) = -1$

$x_v^* = (-v_1, -v_2, \cdots, -v_d)$

then $\nabla f_v(x) \in \{\pm e_1, \pm e_2, \cdots, \pm e_d\}$, and $f_v(x) = x^\top \nabla f_v(x)$

- therefore, $\mathcal{I}(f, x) \in \{\pm e_1, \cdots, \pm e_d\}$, and $K = 2d$
- separation condition: the sets $\{x : f_v(x) < 0\}$ are pairwise disjoint
- applying the compression lemma:

$$\mathrm{Compl}(1) \geq \frac{d}{\log_2(2d)}$$

The construction for general $\varepsilon > 0$

- fix some $k \in \mathbb{N}$, let $\mathcal{F}_k$ be the family of 2^k functions on the k-th level of the tree in the 1-dimensional example
- construction of $\mathcal{F}_{k,d}$ in d-dimensions:

$$f_{i_1,\cdots,i_d}(x) = \max\{f_{i_1}(x_1), \cdots, f_{i_d}(x_d)\}, \qquad i_1, \cdots, i_d \in [2^k]$$

- claim:

$$|\mathcal{F}_{k,d}| = 2^{kd}, \quad K = 2kd, \quad \varepsilon = 2^{-O(k)}$$

- applying the compression lemma:

$$\mathrm{Compl}(\varepsilon) \gtrsim \frac{d \log(1/\varepsilon)}{\log(d \log(1/\varepsilon))}$$

Problem II: dimension-independent convergence

- same setting, but now with no assumption on d
- also the domain of f becomes the unit ℓ_2 ball

Claim

$\min\{d \log \frac{1}{\varepsilon}, \frac{1}{\varepsilon^2}\}$

$$\mathrm{Compl}(\varepsilon, \mathcal{F}) \asymp \begin{cases} 1/\varepsilon^2 & \text{if } \mathcal{F} = \text{convex Lipschitz} \\ 1/\sqrt{\varepsilon} & \text{if } \mathcal{F} = \text{convex smooth} \\ \sqrt{Q} \log(\mu/\varepsilon) & \text{if } \mathcal{F} = \mu\text{-s.c. and } (\mu Q)\text{-smooth} \end{cases}$$

$\nabla^2 f \preceq I$

$\nabla^2 f \succeq \mu I$

(Upper bound achieved by either GD or accelerated GD)

Zero-respecting algorithm

- previous idea: restricting the value of gradient
- current idea: restricting the family of algorithms
- zero-respecting algorithm: $x_1 = 0$, and the i-th coordinates of $\nabla f(x_s)$ are identically zero for all $s < t$, then $x_{t,i} = 0$

Lemma

If a deterministic algorithm achieves $\mathrm{Compl}(\varepsilon, \mathcal{F})$ regardless of the dimension d, then an appropriate zero-respecting algorithm achieves it too.

Proof of the sufficiency of zero-respecting algorithms

A more technical lemma

For every deterministic algorithm $\mathcal{A}$ and $T > 0$, there exists a zero-respecting algorithm $\mathcal{A}_{\mathrm{NR}}$ such that, for every function f, there exists a matrix $U \in \mathbb{R}^{(d+T)\times d}$ with $U^\top U = I_d$ with the following property: if $(x_1, \cdots, x_T)$ is the trajectory of applying $\mathcal{A}_{\mathrm{NR}}$ to f, and $(y_1, \cdots, y_T)$ is the trajectory of applying $\mathcal{A}$ to $f_U(y) \triangleq f(U^\top y)$, then

$$x_t = U^\top y_t, \qquad \forall t \in [T].$$

$\mathcal{A} \downarrow f_U(y) = f(U^\top y)$, $y \in \mathbb{R}^{d+T}$, $y_1, \cdots, y_T$

$\mathcal{A}_{ZR} \downarrow f(x)$, $x \in \mathbb{R}^d$, $x_1, \cdots, x_T$

$x_t = U^\top y_t$

proof of the previous lemma:

- suppose that $\mathcal{A}$ has a worst-case complexity at most T
- construct $\mathcal{A}_{\mathrm{NR}}$ based on $\mathcal{A}$ and T
- for every f, the complexity of $\mathcal{A}_{\mathrm{NR}}$ does not exceed that of $\mathcal{A}$ on f_U, which is T provided that $\mathcal{F}$ is orthogonally invariant

($f_U \in \mathcal{F}$ $\forall$ orthogonal U)

Proof of technical lemma

$\mathcal{A}$ $\quad$ $\mathcal{A}_{ZR}$

$f_U(y) = f(U^\top y)$ $\quad$ $f(x), x \in \mathbb{R}^d$

$y \in \mathbb{R}^{d+T}$

- base step: given y_1, choose $x_1 = 0$ and any U such that $U^\top y_1 = 0$
- induction step:
  - fix any U with certain conditions, let y_t be the t-th iterate of $\mathcal{A}$ on f_U
  - simply set $x_t = U^\top y_t$
- feasibility of x_t:
  - the t-iterate y_t of $\mathcal{A}$ is determined by $\{f(U^\top y_s), U \cdot \nabla f(U^\top y_s)\}_{s<t}$
  - note that $U^\top y_s = x_s$, and $\mathcal{A}_{\mathrm{NR}}$ knows $\{f(x_s), \nabla f(x_s)\}_{s<t}$
- zero-respecting condition:
  - let $S_t \subseteq [d]$ be the union of the support of $\{\nabla f(x_s)\}_{s<t}$
  - choose $\langle u_j, y_t \rangle = 0$ for all $j \notin S_t$
  - a causal way to do so: for $j \in S_t \backslash S_{t-1}$, let u_j be any vector such that $\langle u_j, y_s \rangle = 0$ for all $s < t$
  - feasibility: y_t depends only on U through $(u_j)_{j \in S_t}$
  - existence: each u_j is orthogonal to at most $d - 1 + T$ vectors

Case I: convex Lipschitz function

- construction of the function:

$$f(x) = \max_{i \in [d]} (x_i - \varepsilon_i)$$

$x_1 = (0, 0, \cdots, 0)$
$\nabla f(x_1) = (1, 0, 0, \cdots, 0)$
$x_2 = (?, 0, 0, \cdots, 0)$
$\nabla f(x_2) = e_1$ or e_2
$x_3 = (?, ?, 0, \cdots, 0)$

  where $0 < \varepsilon_1 < \varepsilon_2 < \cdots < \varepsilon_d$ are very close to zero
- zero-respecting algorithm: must have $x_t = (\overbrace{?, ?, \cdots, ?}^{t-1 \text{ entries}}, 0, \cdots, 0)$
- for $d = T + 1$, we have $f(\widehat{x}) \triangleq f(x_{T+1}) \geq -\varepsilon_{T+1} \to 0$
- on the other hand, $\min_{\|x\|_2 \leq 1} f(x) \approx -1/\sqrt{d}$
- conclusion: suboptimality gap after T steps at least $\Omega(1/\sqrt{T})$

Case II: convex smooth function

- "worst function in the world":

$\nabla^2 f \preceq I$

$$f(x) = \frac{1}{8}\left[\left(x_1 - \frac{1}{\sqrt{d}}\right)^2 + (x_1 - x_2)^2 + \cdots + (x_{d-1} - x_d)^2 + x_d^2\right]$$

- $\min_{\|x\|_2 \le 1} f(x) = \min_{x \in \mathbb{R}^d} f(x) \sim 1/(8d^2)$
- again, last $d - t + 1$ entries of x_t must be zero for zero-respecting algorithms
- $\min_{x_{T+1} = \cdots = x_d = 0} f(x) \sim 1/(8dT)$
- conclusion: $d = 2T$ gives the suboptimality gap at least $\Omega(1/T^2)$

Case III: strongly-convex smooth function

- slight modification of "worst function in the world":

$$f(x) = \frac{(Q-1)\mu}{8}\left[(x_1-\delta)^2+(x_1-x_2)^2+\cdots+x_d^2\right]+\frac{\mu}{2}\|x\|_2^2$$

- minimizer of f: $x_k^\star = \delta q^k$ with

$$q = \frac{\sqrt{Q}-1}{\sqrt{Q}+1}$$

- choice of δ: $\|x^\star\|_2 \le 1 \Longrightarrow \delta \asymp Q^{-1/4}$
- minimum of $f \sim c\mu(1-q^{2d})$
- minimum of f with last $d-T$ entries being zero $\asymp c\mu(1-q^{2T})$
- conclusion: $d = T+1$ gives the suboptimality gap $\Omega(\mu e^{-O(T/\sqrt{Q})})$

Problem III: unconstrained quadratic optimization

- setting: solve the equation $Ax = b$ or minimize the quadratic form $f_{A,b}(x) = \frac{1}{2}x^\top Ax - b^\top x$
- assumptions:
  - $A \in \mathbb{R}^{d\times d}$ is PSD with eigenvalues supported on $\Sigma \subseteq [0, \infty)$
  - the solution (minimizer) $x^\star$ satisfies $\|A^{-\tau}x^\star\|_2 \leq 1$
  - error measurement: $L((A, b), \widehat{x}) = \|A^\omega(x^\star - \widehat{x})\|_2$
  - first-order oracle: learner observes the residual $Ax_t - b$
- typical examples:
  - $\tau = 0$ vs. $\tau = -1$: ℓ_2 ball assumption for $x^\star$ or $Ax^\star$
  - $\omega = 0$: estimation error $\|x^\star - \widehat{x}\|_2$
  - $\omega = 1$: residual of the equation $\|A\widehat{x} - b\|_2$
  - $\omega = 1/2$: residual of the minimization $\sqrt{(x^\star - \widehat{x})^\top A(x^\star - \widehat{x})}$

Results

Claim

$$\mathrm{Compl}_{d,\Sigma,\tau,\omega}(\varepsilon) \asymp \min\{d, N^\star(\varepsilon, \Sigma, \omega+\tau)\}$$

where $N^\star(\varepsilon, \Sigma, t)$ is the smallest N such that

$$\inf_{P \in \mathrm{poly}_N} \sup_{x \in \Sigma} x^t |1 - xP(x)| \leq \varepsilon.$$

(Upper bound achieved by Chebyshev's method or conjugate gradient)

typical examples:

- $\Sigma = [\mu, \mu Q]$: $N^\star(\varepsilon, \Sigma, \omega+\tau) \asymp \sqrt{Q}\log(\mu^{\omega+\tau}/\varepsilon)$
- $\Sigma = [0, 1]$: $N^\star(\varepsilon, \Sigma, \omega+\tau) \asymp (1/\varepsilon)^{1/(2(\omega+\tau))}$

$\omega = \frac{1}{2}$, $\tau = 0$, $\varepsilon \to \varepsilon^2$

The linear-span condition

- a new condition on the optimization algorithm:

$$x_t - x_1 \in \mathrm{span}\{\nabla f(x_1), \cdots, \nabla f(x_{t-1})\}$$

Lemma

For quadratic optimization with a given dimension, the performance of any algorithm with $T+1$ iterations could be achieved via an algorithm satisfying the linear-span condition with $2T+1$ iterations.

- lemma implies that $\widehat{x}$ lies in the Krylov space

$$\mathrm{span}\{b, Ab, A^2 b, \cdots, A^{2T} b\}$$

Polynomial approximation

- final output $\widehat{x} = p(A) \cdot b$, where p is a polynomial with degree $n = 2\tau$
- write $A = \sum_{i=1}^{d} \lambda_i e_i e_i^\top$ and $A^{-\tau} x^\star = \sum_{i=1}^{d} u_i e_i$ with $\|u\|_2 \leq 1$, then

$$x^\star = \sum u_i \lambda_i^\tau e_i, \quad b = Ax^\star = \sum u_i \lambda_i^{\tau+1} e_i,$$

$$A^\omega(\widehat{x} - x^\star) = \sum u_i \lambda_i^{\tau+\omega}(\lambda_i p(\lambda_i) - 1) e_i$$

- consequently,

$$\sup_A \|A^\omega(\widehat{x} - x^\star)\|_2^2 = \sup_{\|u\|_2^2 \leq 1, \lambda_i \in \Sigma} \sum u_i^2 \cdot \lambda_i^{2(\tau+\omega)}(\lambda_i p(\lambda_i) - 1)^2$$

$$= \sup_{\lambda \in \Sigma} \lambda^{2(\tau+\omega)}(\lambda p(\lambda) - 1)^2 \stackrel{\triangle}{=} f(\lambda)$$

References

- Arkadi S. Nemirovsky and David B. Yudin, "Problem Complexity and Method Efficiency in Optimization." J. Wiley & Son, 1983.
- Arkadi Nemirovsky. "Information-based complexity of convex programming." Lecture Notes (1995).
- Yurii Nesterov. "Introductory Lectures on Convex Optimization." Kluwer Academic Publishers, Cambridge, 2004.
- Yair Carmon, John C. Duchi, Oliver Hinder, and Aaron Sidford. "Lower bounds for finding stationary points: I." Mathematical Programming (2019): 1-50.
- Kasper Green Larsen and Jelani Nelson. "Optimality of the Johnson-Lindenstrauss lemma." IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS). IEEE, 2017.

Next lecture: communication/privacy constrained estimation