Convex Optimization

Prof. Nati Srebro

Lecture 18:

Course Summary

About the Course

- Methods for solving convex optimization problems, based on oracle access, with guarantees based on their properties

- Understanding different optimization methods
  - Understanding their derivation
  - When are they appropriate
  - Guarantees (a few proofs, not a core component)

- Working and reasoning about opt problems
  - Standard forms: LP, QP, SDP, etc
  - Optimality conditions
  - Duality
  - Using Optimality Conditions and Duality to reason about x^*

Oracles

- **0^{th},1^{st},2^{nd}, etc order oracle for function $\boldsymbol{f}$:**
$$\boldsymbol{x} \mapsto \boldsymbol{f}(\boldsymbol{x}), \boldsymbol{\nabla f}(\boldsymbol{x}), \boldsymbol{\nabla^2 f}(\boldsymbol{x})$$
- Separation oracle for constraint $\mathcal{X}$
$$x \mapsto \begin{cases} x \in \mathcal{X} \\ g\ s.t.\, \mathcal{X} \subset \{x' | \langle g, x - x' \rangle < 0\} \end{cases}$$
- Projection oracle for constraint $\mathcal{X}$ (w.r.t. $\|\cdot\|_2$):
$$y \to \arg\min_{x \in \mathcal{X}} \|x - y\|_2^2$$
- Linear optimization oracle
$$v \to \arg\min_{x \in \mathcal{X}} \langle v, x \rangle$$
- Prox/projection oracle for function h w.r.t. $\|\cdot\|_2$:
$$y, \lambda \mapsto \arg\min_{x} h\,(x) + \lambda \|x - y\|_2^2$$
- Stochastic oracle:
$$x \mapsto g \quad s.t. \quad \mathbb{E}[g] = \nabla f(x)$$

Analysis Assumptions

- f is L-Lipschitz (w.r.t $\|\cdot\|$)
$$|f(x) - f(y)| \leq L\|x - y\|$$
- f is M-Smooth (w.r.t $\|\cdot\|$)
$$f(x + \Delta x) \leq f(x) + \langle \nabla f(x), \Delta x \rangle + \frac{M}{2}\|\Delta x\|^2$$
- f is λ-strongly convex (w.r.t $\|\cdot\|$)
$$f(x) + \langle \nabla f(x), \Delta x \rangle + \frac{\mu}{2}\|\Delta x\|^2 \leq f(x + \Delta x)$$
- f is self-concordant
$$\forall_x \forall_{\text{direction } v} |f_v'''(x)| \leq 2 f''(x)^{3/2}$$
- f is quadratic (i.e. $f''' = 0$)
- $\|x^*\|$ is small (or domain is bounded)
- Initial sub-optimality $\epsilon_0 = f\left(x^{(0)}\right) - p$ is bounded

Overall runtime

(runtime of each iteration) $\times$ **(number of required iterations)**

(oracle runtime) + **(other operations)**

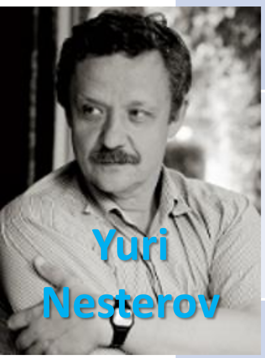
Contrast to Non-Oracle Methods

- Symbolically solving $\nabla f(x) = 0$
- Gaussian elimination
- Simplex method for LPs

Advantages of Oracle Approach

- Handle generic functions (not only of specific parametric form)
- Makes it clear what functions can be handled, what we need to be able to compute about them
- Complexity in terms of "oracle accesses"
- Can obtain lower bound on number of oracle accesses required

Unconstrained Optimization with 1^{st} Oder Oracle— Dimension Independent Guarantees



	$\mu \preccurlyeq \nabla^2 \preccurlyeq M$	$\nabla^2 \preccurlyeq M$	$\|\nabla\| \le L$	$\|\nabla\| \le L$, $\mu \preccurlyeq \nabla^2$
GD	$\frac{M}{\mu} \log 1/\epsilon$	$\frac{M\|x^*\|^2}{\epsilon}$	$\frac{L^2\|x^*\|^2}{\epsilon^2}$	$\frac{L^2}{\mu\epsilon}$
A-GD	$\sqrt{M/\mu} \log 1/\epsilon$	$\sqrt{\frac{M\|x^*\|^2}{\epsilon}}$		
Lower bound	$\Omega\left(\sqrt{M/\mu} \log 1/\epsilon\right)$	$\Omega\left(\sqrt{\frac{M\|x^*\|^2}{\epsilon}}\right)$	$\Omega\left(\frac{L^2\|x^*\|^2}{\epsilon^2}\right)$	$\Omega\left(\frac{L^2}{\mu\epsilon}\right)$

Theorem: for any method that uses only 1^{st} order oracle access, there exists a L-Lipschitz function with $\|x^*\| \le 1$ s.t. at least $\Omega\left(L^2/\epsilon^2\right)$ oracle accesses are required for the method to find an ϵ-suboptimal solution

Unconstrained Optimization with 1st Oder Oracle— Low Dimensional Guarantees

	#oracle calls	Runtime per iter	Overall runtime
Center of Mass	$n \log \frac{1}{\epsilon}$	*	*
Ellipsoid	$n^2 \log \frac{1}{\epsilon}$	n^2	$n^4 \log \frac{1}{\epsilon}$
Vaidya [1989]	$\tilde{O}\left(n \log \frac{1}{\epsilon}\right)$	$O(n^3)$	$\tilde{O}\left(n^4 \log \frac{1}{\epsilon}\right)$
Vaidya++ [Lee Sidford Wang 2015]	$\tilde{O}\left(n \log \frac{1}{\epsilon}\right)$	$\tilde{O}(n^2)$	$\tilde{O}\left(n^3 \log \frac{1}{\epsilon}\right)$
Lower Bound	$\Omega\left(n \log \frac{1}{\epsilon}\right)$		

Overall runtime

(runtime of each iteration) $\times$ (number of required iterations)

Unconstrained Optimization-Practical Methods

	Memory	Per iter	#iter $\mu \preccurlyeq \nabla^2 \preccurlyeq M$	#iter quadratic
GD	$\boldsymbol{O(n)}$	$\boldsymbol{\nabla f + O(n)}$	$\boldsymbol{\kappa \log 1/\epsilon}$	$\boldsymbol{\kappa \log 1/\epsilon}$
A-GD	$O(n)$	$\nabla f + O(n)$	$\sqrt{\boldsymbol{\kappa}}\, \boldsymbol{\log 1/\epsilon}$	$\sqrt{\kappa} \log 1/\epsilon$
Momentum	$O(n)$	$\nabla f + O(n)$		
Conj-GD	$O(n)$	$\nabla f + O(n)$		$\sqrt{\kappa} \log 1/\epsilon$
L-BFGS	$O(rn)$	$\nabla f + O(rn)$		
BFGS	$O(n^2)$	$\nabla f + O(n^2)$		$\sqrt{\kappa} \log 1/\epsilon$
Newton	$\boldsymbol{O(n^2)}$	$\boldsymbol{\nabla^2 f + O(n^3)}$	$\boldsymbol{\frac{f(x^{(0)} + p^*)}{\gamma} + \log \log 1/\epsilon}$	**1**

Constrained Optimization

$$\min_{x \in \mathcal{X}} f(x)$$

- Projected Gradient Descent
  - Oracle: $y \mapsto \arg\min_{x \in \mathcal{X}} \|x - y\|_2^2$
  - $O(n)$ + projection per iteration
  - Similar iteration complexity to Grad Descent:
    poly dependence on $\boldsymbol{\epsilon}$ or on $\boldsymbol{\kappa}$
- Conditional Gradient Descent:
  - Oracle: $v \to \arg\min_{x \in \mathcal{X}} \langle v, x \rangle$
  - $O(n)$ + linear optimization per iteration
  - $O(^M/_\epsilon)$ iterations, but no acceleration, no $O(\kappa \log {}^1/_\epsilon)$
- Ellipsoid Method
  - Separation oracle (can handle much more generic $\mathcal{X}$)
  - $O(n^2)$ + separation oracle per itration
  - $O(n^2 \log {}^1/_\epsilon)$ iterations ➔ total runtime $O(n^4 \log {}^1/_\epsilon)$

Constrained Optimization—Functional Constraints

$$\min_{x \in \mathbb{R}^n} \quad f_0(x) \qquad s.t. \quad f_i(x) \leq 0, \quad Ax = b$$

$$\min_{x \in \mathbb{R}^n} \quad f_0(x) \qquad s.t. \quad f_i(x) \preccurlyeq 0, \quad Ax = b$$

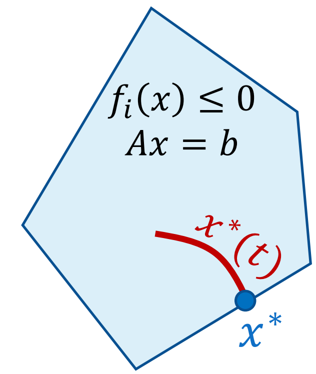
$$\min_{x \in \mathbb{R}^n} \quad f_0(x) \qquad s.t. \quad -f_i(x) \in K_i, \quad Ax = b$$

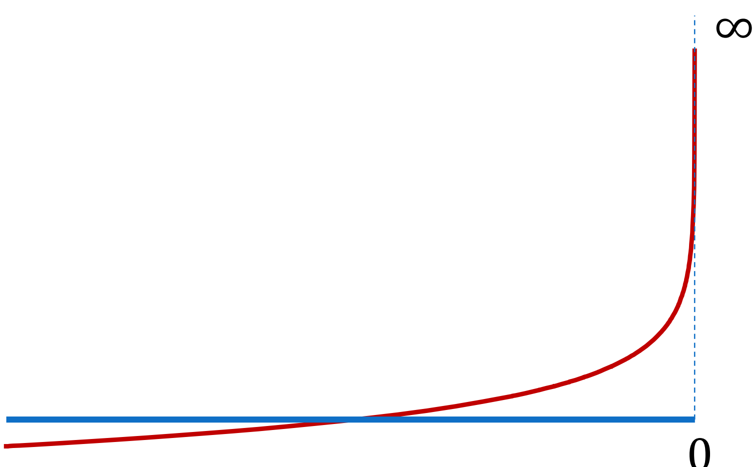
- Interior Point Methods
  - Only 1st and 2nd oracle access to f_0, f_i
  - Analysis when f_0 self concordant and f_i linear or quadratic
  - Overall #Newton Iterations: $O(\sqrt{m}\,(\log {}^{1}/_{\epsilon} + \log\log {}^{1}/_{\delta}))$
  - Overall runtime: $\approx O\left(\sqrt{m}\left((n+p)^3 + m\ \nabla^2 \text{ evals}\right)\log {}^{1}/_{\epsilon}\right)$
- Can also handle matrix inequalities (SDPs)
- Other cones: need self concordant (log-like) barrier
- "Standard Method" for LPs, QPs, SDPs, etc

Barrier Methods

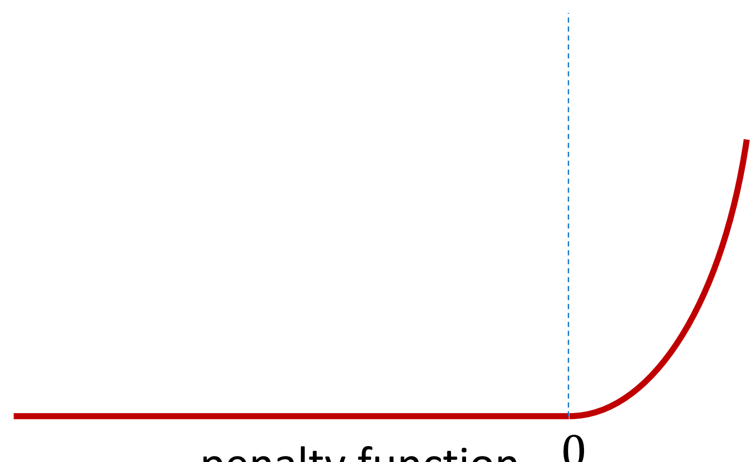
$$\boxed{\begin{array}{ll} \min\limits_{x \in \mathbb{R}^n} & f_0(x) + \sum_{i=1}^{m} I\big(f_i(x)\big) \\ s.t. & Ax = b \end{array}}$$

- Log barriers $I_t(u) = -\frac{1}{t}\log(-u)$:
  - Analysis and Guarantees
  - “Central Path”
  - Relaxation of KKT conditions





barrier function



penalty function

Optimality Condition (assuming Strong Duality)

x^* optimal for (P)

λ^*, ν^* optimal for (D)

<u>KKT Conditions</u>

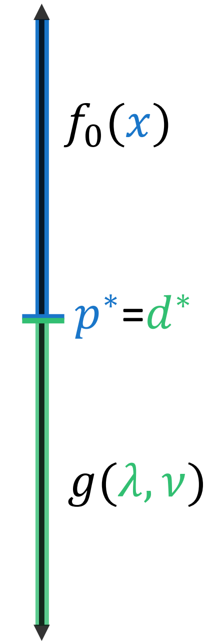
$f_i(x^*) \leq 0 \; \forall_{i=1\ldots m}$

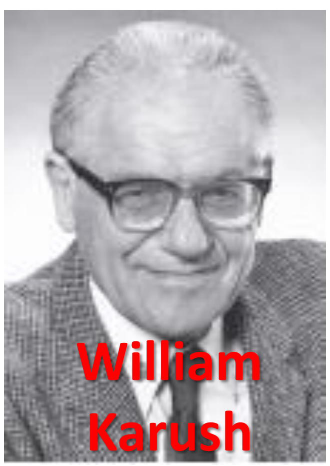
$h_j(x^*) = 0 \; \forall_{j=1\ldots p}$

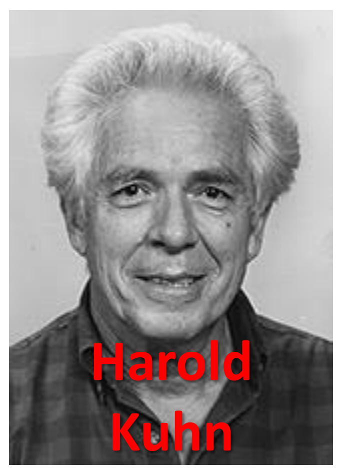
$\lambda_i^* \geq 0 \; \forall_{i=1..m}$

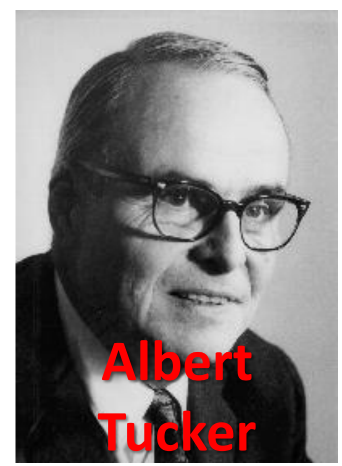
$\nabla_x L\big(x^*, (\lambda^*, \nu^*)\big) = 0$

$\lambda_i^* f_i(x^*) = 0 \; \forall_{i=1..m}$









Optimality Condition for Problem with Log Barrier

x^* optimal for (P_t)

ν^* optimal for (D_t)

$\Rightarrow$

$\Leftarrow$

KKT Conditions

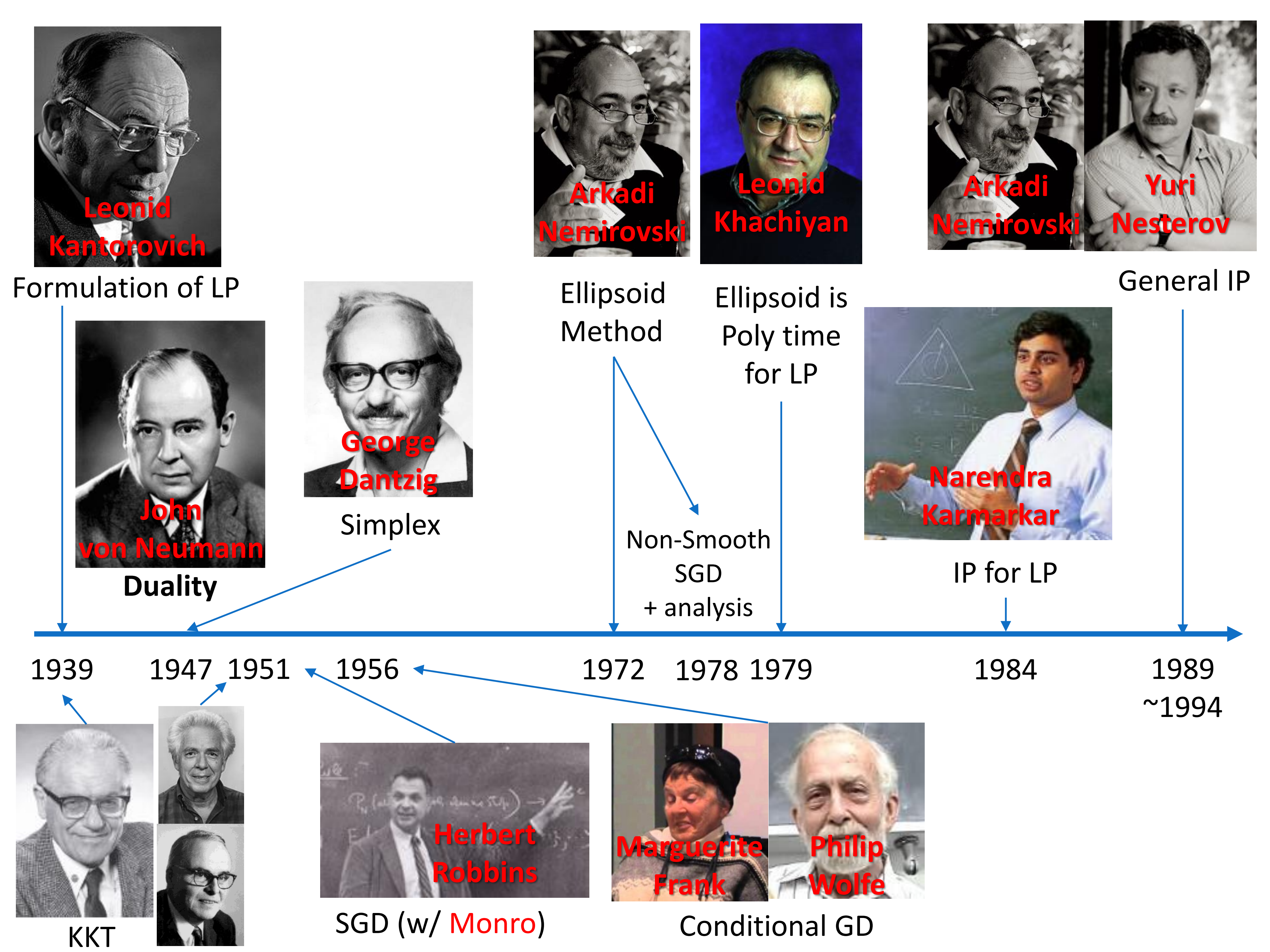
$f_i(x^*) \leq 0 \; \forall_{i=1\ldots m}$

$h_j(x^*) = 0 \; \forall_{j=1\ldots p}$

$\lambda_i^* \geq 0 \; \forall_{i=1..m}$

$\nabla_x L\left(x^*, (\lambda^*, \nu^*)\right) = 0$

$\lambda_i^* f_i(x^*) = \frac{1}{t} \; \forall_{i=1..m}$



Leonid Kantorovich
Formulation of LP
John von Neumann
Duality
George Dantzig
Simplex
Arkadi Nemirovski
Ellipsoid Method
Leonid Khachiyan
Ellipsoid is Poly time for LP
Non-Smooth SGD + analysis
Arkadi Nemirovski
Yuri Nesterov
General IP
Narendra Karmarkar
IP for LP
1939
1947
1951
1956
1972
1978
1979
1984
1989 ~1994
KKT
Herbert Robbins
SGD (w/ Monro)
Marguerite Frank
Philip Wolfe
Conditional GD

From Interior Point Methods Back to First Order Methods

- Interior Point Methods (and before them Ellipsoid):
  - $O\left(poly(n)\log\frac{1}{\epsilon}\right)$
  - LP in time polynomial in size of input (number of bits in coefficients)
- But in large scale applications, $O(n^{3.5})$ too high
- First order (and stochastic) methods:
  - Better dependence on n
  - $poly\left(\frac{1}{\epsilon}\right)$ (or $poly(\kappa)$)

Overall runtime

(runtime of each iteration) × **(number of required iterations)**

(oracle runtime) + **(other operations**)

Example: ℓ_1 regression

$$f(x) = \sum_{i=1}^{m} |\langle a_i, x\rangle - b_i|$$

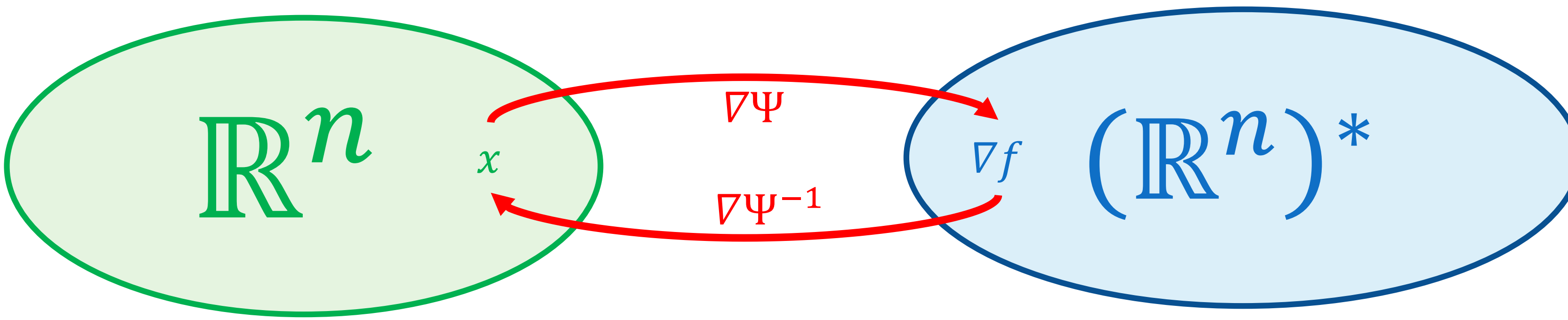
- Option 1: Cast as constrained optimization

$$\min_{x\in\mathbb{R}^n, t\in\mathbb{R}^m} \sum_i t_i \quad s.t. \quad -t_i \le \langle a_i, x\rangle - b_i \le t$$

  - Runtime: $O(\sqrt{m}(n+m)^3 \log {}^1/_\epsilon)$
- Options 2: Gradient Descent
  - $O\left(\frac{\|a\|^2\|x^*\|^2}{\epsilon^2}\right)$ iterations
  - $O(nm)$ per iteration ➔ overall $O\left(nm\frac{1}{\epsilon^2}\right)$
- Option 3: Stochastic Gradient Descent
  - $O\left(\frac{\|a\|^2\|x^*\|^2}{\epsilon^2}\right)$ iterations
  - $O(n)$ per iteration ➔ overall $O\left(n\frac{1}{\epsilon^2}\right)$

Remember!

- Gradient is in dual space---don't take mapping for granted!



Some things we didn't cover

- Technology for unconstrained optimization
  - Different quasi-Newton and conj gradient methods
  - Different line searches
- Technology for Interior Point Primal-Dual methods
- Numerical Issues
- Recent advances in first order and stochastic methods
  - Mirror Descent, Different Geometries, Adaptive Geometries
  - Decomposable functions, partial linearization and acceleration
  - 0^{th} order optimization (using only function values, no derivatives)
- Faster methods for problems with specific forms or properties
  - Message passing for solving LPs
  - Flows and network problems
- Distributed Optimization

Beyond Convex Optimization

Non-Convex Optimization a.k.a. “Non-Linear Programming”

- Many of the ideas and methods, especially for unconstrained optimization, carry over
- Ensuring global optimality is hard, and often impossible
- Some theory for convergence to critical point or local minimum (even this is much harder then for convex)

Current Trends

- Small scale problems:
  - Super-efficient and reliable solvers for use in real-time
- Very large scale problems:
  - Stochastic first order methods
  - Linear time, one-pass or few-pass
- Relationship to Machine Learning

Other Courses

Spring 2018:

- Online Optimization and Decision Making under Uncertainty (Varun Gupta, Booth)

Fall 2018:

- Computational and Statistical Learning Theory (Srebro, TTIC)
- Matrix Computation (Lek-Heng Lim, Statistics)

Winter 2018:

- Stochastic Optimization (John Birge, Booth)

About the Course

- Methods for solving convex optimization problems, based on oracle access, with guarantees based on their properties

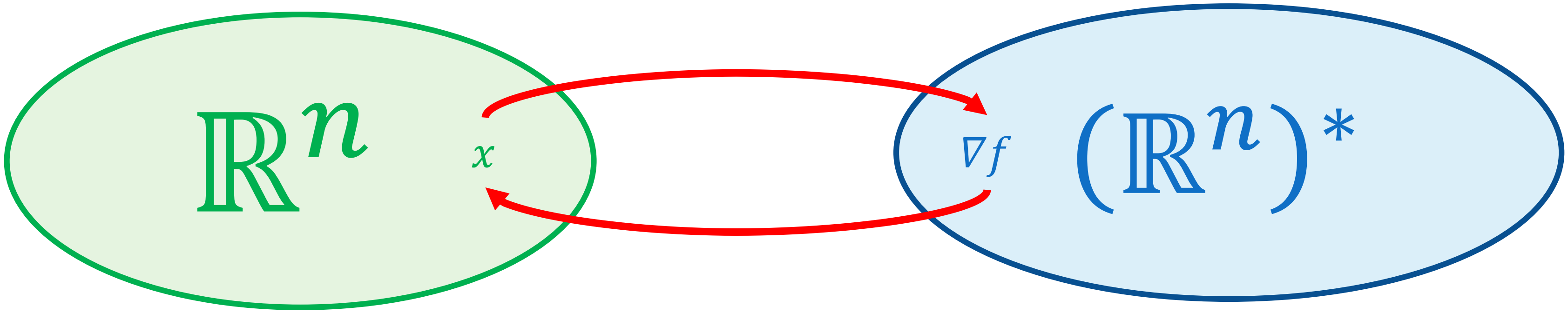
- Understanding different optimization methods
  - Understanding their derivation
  - When are they appropriate
  - Guarantees (a few proofs, not a core component)

- Working and reasoning about opt problems
  - Standard forms: LP, QP, SDP, etc
  - Optimality conditions
  - Duality
  - Using Optimality Conditions and Duality to reason about x^*

Final

- Wednesday 9:30am
- Straight-forward questions
- Allowed: anything hand-written by you (not photo-copied)

- In depth: Up to and including Interior Point Methods
    - Unconstrained Optimization
    - Duality
    - Optimality Conditions
    - Interior Point Methods
    - Phase I Methods and Feasibility Problems
- Superficially (a few multiple choice or similar)
    - Other first order and stochastic methods
    - Lower Bounds
    - Simplex, Center of Mass / Ellipsoid
- Not covered: Monday's lecture (prox methods, mirror descent)



x^* optimal for (P)

λ^*, ν^* optimal for (D)

$\rightleftarrows$

$$f_i(x^*) \leq 0 \ \forall_{i=1\ldots m}$$

$$h_j(x^*) = 0 \ \forall_{j=1\ldots p}$$

$$\lambda_i^* \geq 0 \ \forall_{i=1..m}$$

$$\nabla_x L\big(x^*, (\lambda^*, \nu^*)\big) = 0$$

$$\lambda_i^* f_i(x^*) = 0 \ \forall_{i=1..m}$$