Convex Optimization

Lecture 1

Today

- Convex optimization: introduction
- Course organization and other gory details...

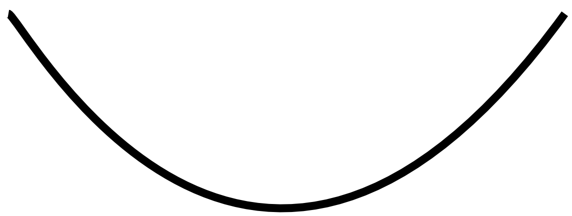
Convex Optimization

Convex Optimization

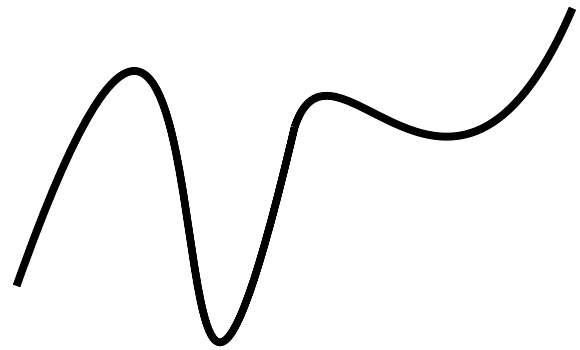
- How do we identify **easy** and **hard** problems?
- **Convexity**: why is it so important?
- Modeling: how do we recognize easy problems in real **applications**?
- Algorithms: how do we solve these problems **in practice**?

Introduction

Convexity.



Convex



Not convex

Key message from **complexity theory:** as the problem dimension gets large

- all **convex** problems are easy,
- most nonconvex problems are hard.

Introduction

Convex problem.

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

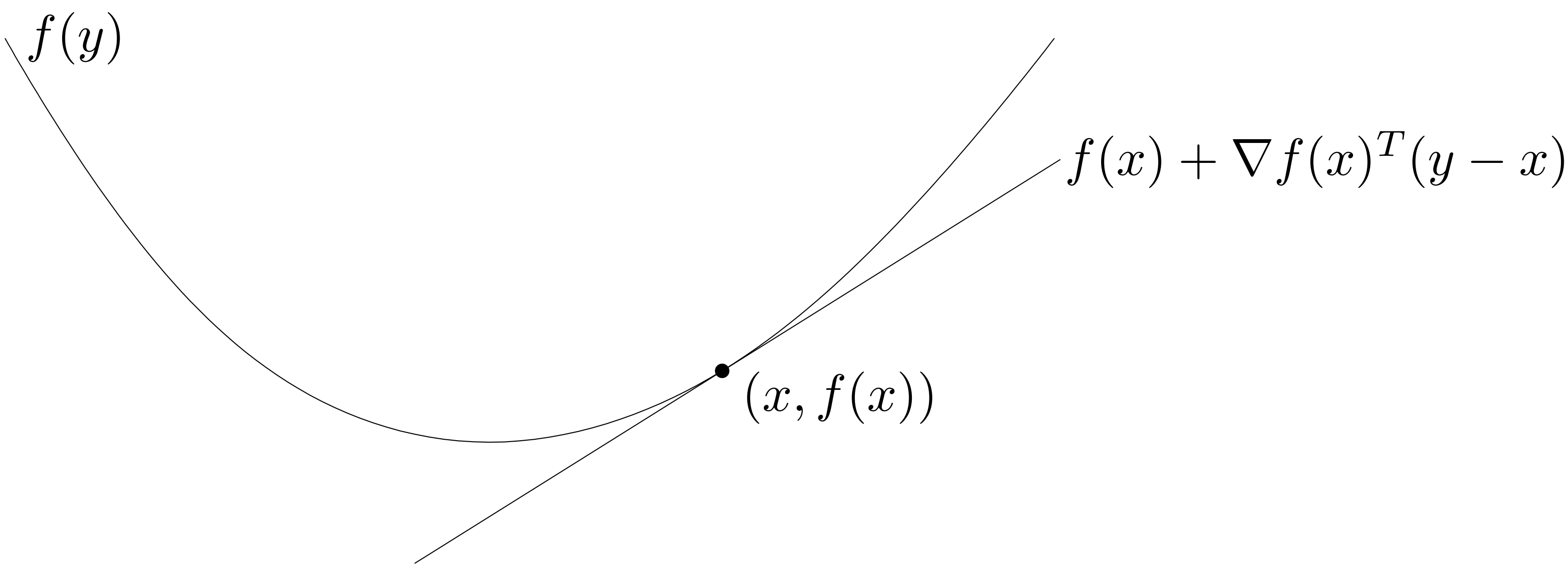
$f_0, f_1, \ldots, f_m$ are convex functions, the equality constraints are all affine.

- Strong assumption, yet **surprisingly expressive.**
- Good convex approximations of nonconvex problems.

Introduction

First-order condition. Differentiable f with convex domain is convex iff

$$f(y) \geq f(x) + \nabla f(x)^T (y - x) \quad \text{for all } x, y \in \mathbf{dom}\, f$$



First-order approximation of f is global underestimator

Least squares (LS)

$$\text{minimize} \quad \|Ax - b\|_2^2$$

$A \in \mathbf{R}^{m \times n}$, $b \in \mathbf{R}^m$ are parameters; $x \in \mathbf{R}^n$ is variable

- Complete theory (existence & uniqueness, sensitivity analysis . . .)
- Several algorithms compute (global) solution reliably
- We can solve dense problems with $n = 1000$ vbles, $m = 10000$ terms
- By exploiting structure (e.g., sparsity) can solve **far larger** problems

. . . LS is a (widely used) **technology**

Linear program (LP)

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & a_i^T x \leq b_i, \quad i = 1, \ldots, m \end{array}$$

$c, \; a_i \in \mathbf{R}^n$ are parameters; $x \in \mathbf{R}^n$ is variable

- Nearly complete theory
  (existence & uniqueness, sensitivity analysis . . .)
- Several algorithms compute (global) solution reliably
- Can solve dense problems with $n = 1000$ vbles, $m = 10000$ constraints
- By exploiting structure (e.g., sparsity) can solve **far larger** problems

. . . LP is a (widely used) **technology**

Quadratic program (QP)

$$\begin{array}{ll} \text{minimize} & \|Fx - g\|_2^2 \\ \text{subject to} & a_i^T x \leq b_i, \quad i = 1, \ldots, m \end{array}$$

- Combination of LS & LP
- Same story . . . QP is a technology
- Reliability: Programmed on chips to solve **real-time** problems
- Classic application: **portfolio optimization**

The bad news

- LS, LP, and QP are **exceptions**
- Most optimization problems, even some very simple looking ones, are **intractable**
- The objective of this class is to show you how to recognize the nice ones. . .
- Many, many applications across all fields. . .

Polynomial minimization

$$\text{minimize} \quad p(x)$$

p is polynomial of degree d; $x \in \mathbf{R}^n$ is variable

- Except for special cases (e.g., $d = 2$) this is a **very difficult problem**
- Even sparse problems with size $n = 20$, $d = 10$ are essentially intractable
- All algorithms known to solve this problem require effort exponential in n

What makes a problem easy or hard?

Classical view:

- **linear** is easy
- **nonlinear** is hard(er)

What makes a problem easy or hard?

Emerging (and correct) view:

. . . the great watershed in optimization isn't between linearity and nonlinearity, but convexity and nonconvexity.

— R. Rockafellar, SIAM Review 1993

Convex Optimization

A brief history...

- The field is about 50 years old.
- Starts with the work of Von Neumann, Kuhn and Tucker, etc
- Explodes in the 60's with the advent of "relatively" cheap and efficient computers...
- Key to all this: fast linear algebra
- Some of the theory developed before computers even existed...

Linear programming & the simplex method

Linear Programming, history:

- First solution by Dantzig in the late 40's. Famous story. . .
- At the time, programs were solved by hand, the algorithm reflects this.
- In 1972, Klee and Minty show that the simplex has an exponential worst case complexity
- Low complexity of linear programming proved (in theory) by Nemirovski, Yudin and Khachiyan in the USSR in 1976.
- First efficient algorithm with provably low complexity discovered by Karmarkar at Bell Labs in 1984.

Linear programming & the simplex method

Also in 1948. . .

Linear programming & the simplex method

- First serious LP solved: 9 variables and 77 constraints.
- It took **120 man-days** to solve it. . .
- Computing power: A few air force soldiers stuck in a room for a few days.

- Sixty years later, the same (mostly) algorithm is used to solve problems with millions of variables.

Optimization

Always the same process: starting from a particular application. . .

- **Modeling**: model your problem as a member of a particular class of problems that can be solved efficiently (a linear program for example).
- **Solving**: feed this problem to your favorite solver. If that's not possible, write an algorithm to solve it.

Course Organization

Course Plan

- Convex analysis & modeling
- Duality
- Applications
- Algorithms: interior point methods, first order methods.

Website

Course website with lecture notes, homework, etc.

http://www.di.ens.fr/~aspremon/ENSAE.html

- TDs
- Final exam: TBD

Short blurb

- Contact info on `http://www.di.ens.fr/~aspremon`
- Email: `aspremon@ens.fr`
- Dual PhDs: Ecole Polytechnique & Stanford University
- Interests: Optimization, machine learning, statistics & finance.

References

- All lecture notes will be posted online
- Textbook: **Convex Optimization** by Lieven Vandenberghe and Stephen Boyd, available online at:

  http://www.stanford.edu/~boyd/cvxbook/

- See also Ben-Tal and Nemirovski (2001), "Lectures On Modern Convex Optimization: Analysis, Algorithms, And Engineering Applications", SIAM.

  http://www2.isye.gatech.edu/~nemirovs/

- Nesterov (2003), "Introductory Lectures on Convex Optimization", Springer.
- Nesterov and Nemirovskii (1994), "Interior Point Polynomial Algorithms in Convex Programming", SIAM.

*

References

Ben-Tal, A. & Nemirovski, A. (2001), *Lectures on modern convex optimization : analysis, algorithms, and engineering applications*, MPS-SIAM series on optimization, Society for Industrial and Applied Mathematics : Mathematical Programming Society, Philadelphia, PA.

Karmarkar, N. K. (1984), 'A new polynomial-time algorithm for linear programming', *Combinatorica* **4**, 373–395.

Khachiyan, L. G. (1979), 'A polynomial algorithm in linear programming (in Russian)', *Doklady Akademiia Nauk SSSR* **224**, 1093–1096.

Nemirovskii, A. & Yudin, D. (1979), 'Problem complexity and method efficiency in optimization', *Nauka (published in English by John Wiley, Chichester, 1983)* .

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

Nesterov, Y. & Nemirovskii, A. (1994), *Interior-point polynomial algorithms in convex programming*, Society for Industrial and Applied Mathematics, Philadelphia.