Convex Optimization: Modeling and Applications

Lieven Vandenberghe (University of California, Los Angeles)

Based on work with **Stephen Boyd** (Stanford University)

Machine Learning Summer School

Tübingen, Germany

August 25, 2007

Outline

- **Lecture 1**: introduction, convex sets and functions
- **Lecture 2**: standard problem classes, cone programming
- **Lecture 3**: robust optimization, polynomial optimization, 1-norm heuristics

Lecture 1

1. Introduction

- mathematical optimization
- least-squares and linear programming
- convex optimization
- brief history

Mathematical optimization

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

variable $x = (x_1, \ldots, x_n)$

- complexity varies widely, depending on properties of f_i, h_i
- general methods involve compromise (computation time, suboptimality)

Exceptions: certain problem classes can be solved efficiently and reliably

- least-squares, linear programming
- convex optimization problems

Least-squares

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

Solving least-squares problems

- analytical solution: $x^\star = (A^TA)^{-1}A^Tb$
- reliable and efficient algorithms and software
- computation time proportional to n^2p if A is $p \times n$; less if structured

Using least-squares

- least-squares problems are easy to recognize
- a few standard techniques increase flexibility ($e.g.$, including weights)

Linear programming

$$\begin{array}{ll} \text{minimize} & c_1x_1 + \cdots + c_nx_n \\ \text{subject to} & a_{i1}x_1 + \cdots + a_{in}x_n \le b_i, \quad i = 1, \ldots, m \end{array}$$

Solving linear programs

- no analytical formula for solution
- reliable and efficient algorithms and software
- computation time proportional to n^2m if $m \ge n$; less with structure

Using linear programming

- not as easy to recognize as least-squares problems
- a few standard tricks used to convert problems into linear programs ($e.g.$, problems involving 1- or ∞-norms, piecewise-linear functions)

Convex optimization problem

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq 0, \quad i = 1, \ldots, m \\ & Ax = b \end{array}$$

- objective and inequality constraint functions are convex:

$$f_i(\alpha x + \beta y) \leq \alpha f_i(x) + \beta f_i(y)$$

  if $\alpha + \beta = 1$, $\alpha \geq 0$, $\beta \geq 0$

- equality constraints are linear
- includes least-squares problems and linear programs as special cases

Solving convex optimization problems

- no analytical solution
- reliable and efficient algorithms
- computation time (roughly) proportional to $\max\{n^3, n^2m, F\}$, where F is cost of evaluating f_i's and their first and second derivatives
- several software packages (in Matlab, C, Python, ...)

Using convex optimization

- convex problems are often difficult to recognize
- many tricks for transforming problems into convex form
- surprisingly many problems can be solved via convex optimization

History

- 1940s: linear programming

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

- 1950s: quadratic programming
- 1960s: geometric programming
- 1990s: semidefinite programming, second-order cone programming, quadratically constrained quadratic programming, robust optimization, sum-of-squares programming, . . .

New applications since 1990

- linear matrix inequality techniques in control
- circuit design via geometric programming
- support vector machine learning via quadratic programming
- semidefinite pogramming relaxations in combinatorial optimization
- applications in structural optimization, statistics, machine learning, signal processing, communications, image processing, quantum information theory, finance, . . .

Interior-point methods

Linear programming

- 1984 (Karmarkar): first practical polynomial-time algorithm
- 1984-1990: efficient implementations for large-scale LPs

Nonlinear convex optimization

- around 1990 (Nesterov & Nemirovski): polynomial-time interior-point methods for nonlinear convex programming
- since 1990: extensions and high-quality software packages

2. Convex sets

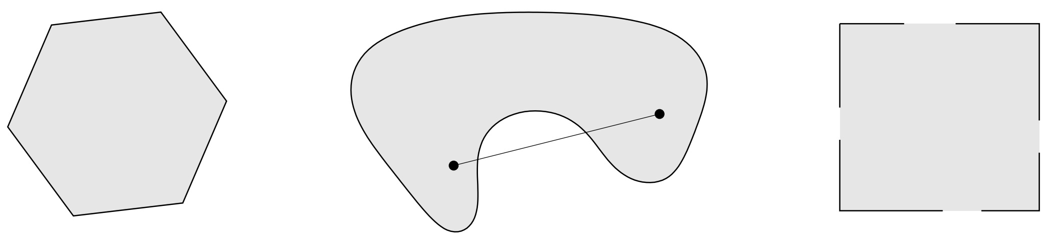
- definition
- some important examples
- operations that preserve convexity

Definition

Convex set: contains line segment between any two points in the set

$$x_1, x_2 \in C, \quad 0 \le \theta \le 1 \quad \Longrightarrow \quad \theta x_1 + (1-\theta) x_2 \in C$$

Examples (one convex, two nonconvex sets)



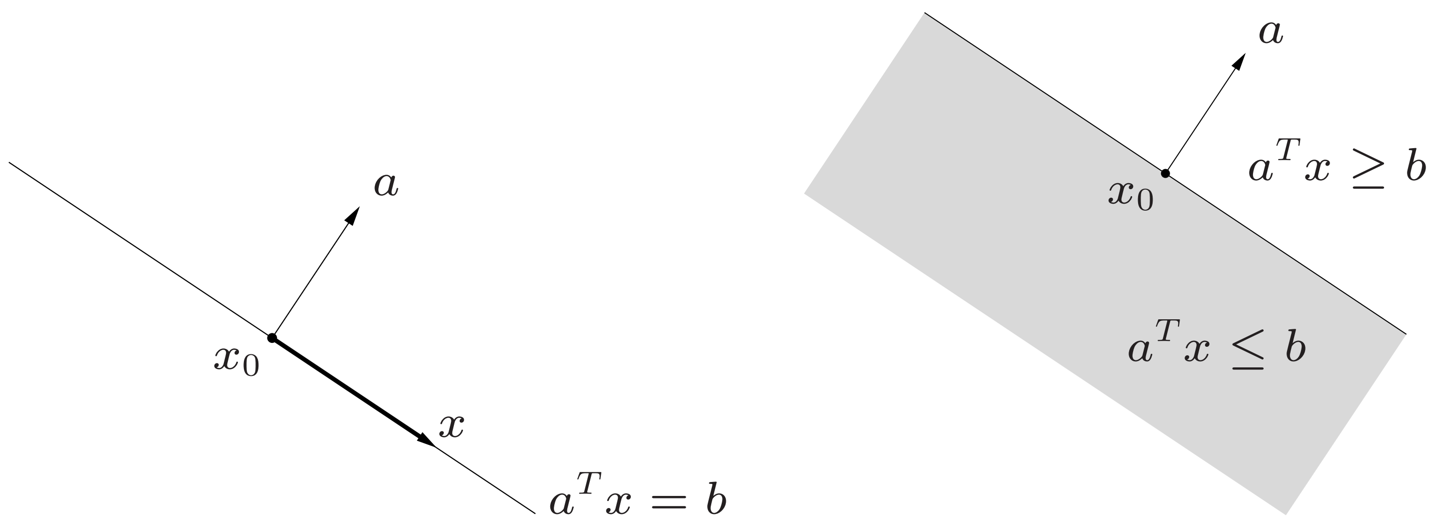
Convex cone:

$$x_1, x_2 \in C, \quad \theta_1, \theta_2 \ge 0 \quad \Longrightarrow \quad \theta_1 x_1 + \theta_2 x_2 \in C$$

Hyperplanes and halfspaces

Hyperplane: set of the form $\{x \mid a^T x = b\}$ with $a \neq 0$

Halfspace: set of the form $\{x \mid a^T x \leq b\}$ with $a \neq 0$



- a is the normal vector
- hyperplanes and halfspaces are convex

Euclidean balls and ellipsoids

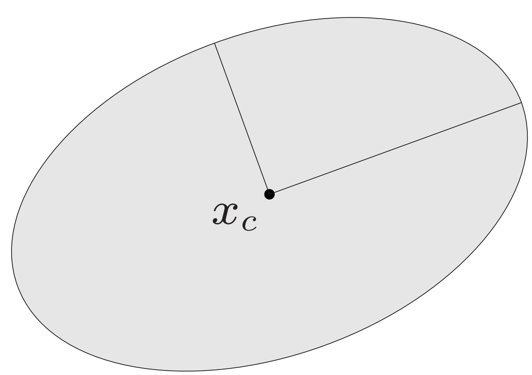
(Euclidean) ball with center x_c and radius r:

$$\{x_c + ru \mid \|u\|_2 \leq 1\}$$

Ellipsoid with center x_c:

$$\{x_c + Au \mid \|u\|_2 \leq 1\}$$

with A square and nonsingular



Norm balls and norm cones

Notation: $\|\cdot\|$ is general (unspecified) norm; $\|\cdot\|_{\text{symb}}$ is particular norm

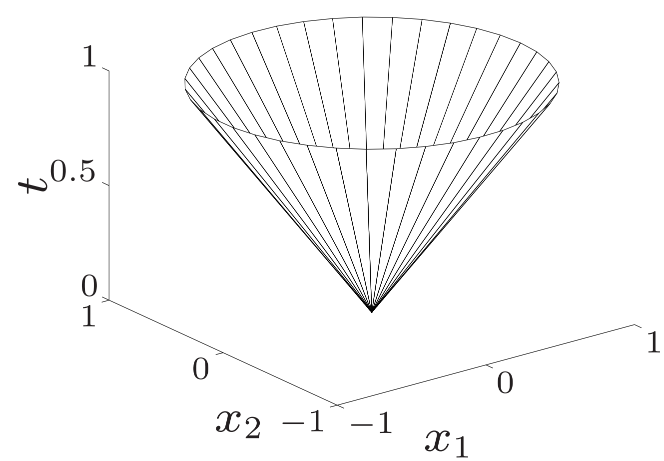
Norm ball with center x_c and radius r:

$$\{x \mid \|x - x_c\| \leq r\}$$

Norm cone:

$$\{(x,t) \mid \|x\| \leq t\}$$

Euclidean norm cone is called second-order cone



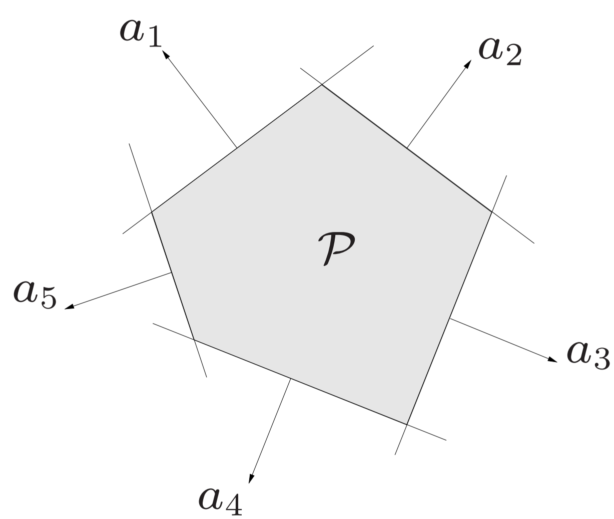
norm balls and cones are convex

Polyhedra

solution set of finitely many linear inequalities and equalities

$$Ax \preceq b, \qquad Cx = d$$

($A \in \mathbf{R}^{m \times n}$, $C \in \mathbf{R}^{p \times n}$, $\preceq$ is componentwise inequality)



polyhedron is intersection of finite number of halfspaces and hyperplanes

Positive semidefinite cone

Notation:

- $\mathbf{S}^n$ is set of symmetric $n \times n$ matrices
- $\mathbf{S}^n_+ = \{X \in \mathbf{S}^n \mid X \succeq 0\}$: positive semidefinite $n \times n$ matrices

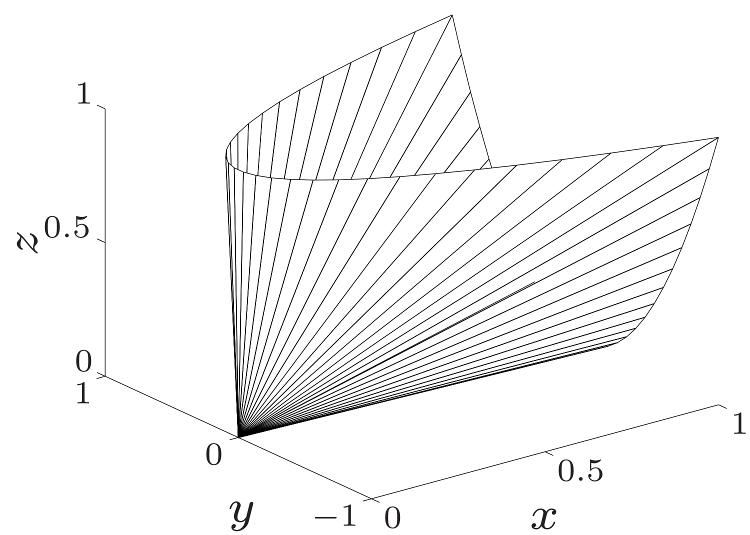
$$X \in \mathbf{S}^n_+ \quad \Longleftrightarrow \quad z^T X z \geq 0 \text{ for all } z$$

  $\mathbf{S}^n_+$ is a convex cone

- $\mathbf{S}^n_{++} = \{X \in \mathbf{S}^n \mid X \succ 0\}$: positive definite $n \times n$ matrices

Example:

$$\begin{bmatrix} x & y \\ y & z \end{bmatrix} \in \mathbf{S}^2_+$$



Operations that preserve convexity

practical methods for establishing convexity of a set C

1. apply definition

$$x_1, x_2 \in C, \quad 0 \leq \theta \leq 1 \quad \Longrightarrow \quad \theta x_1 + (1-\theta)x_2 \in C$$

2. show that C is obtained from simple convex sets (hyperplanes, halfspaces, norm balls, . . .) by operations that preserve convexity
   - intersection
   - affine functions
   - perspective function
   - linear-fractional functions

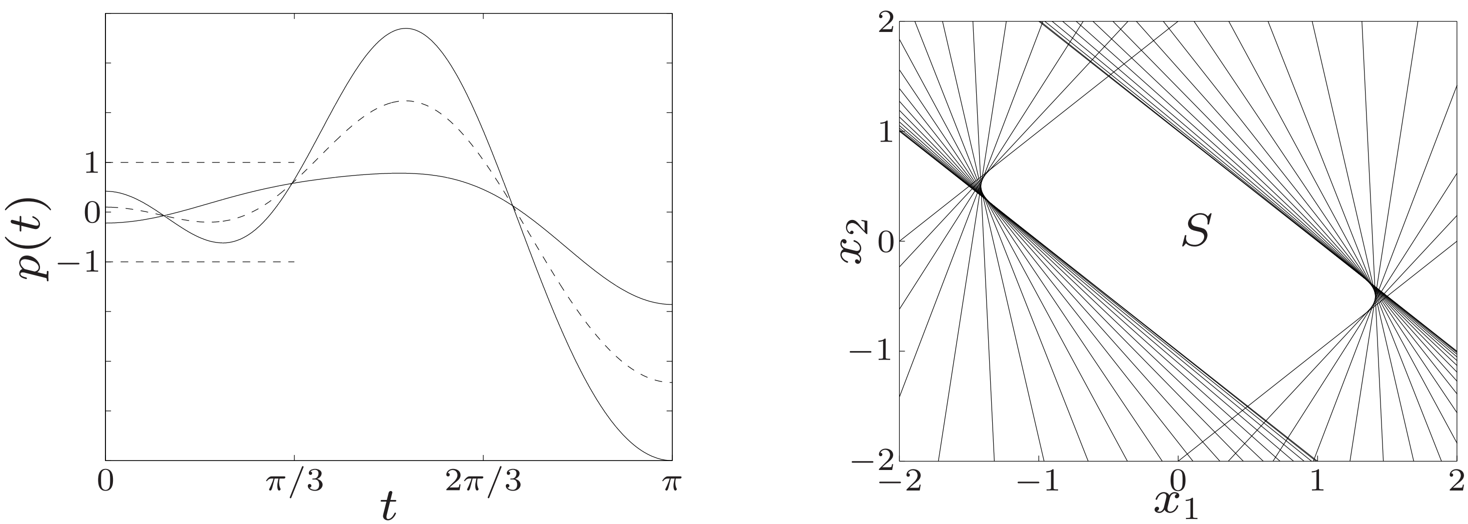
Intersection

the intersection of (any number of) convex sets is convex

Example:

$$S = \{x \in \mathbf{R}^m \mid |p(t)| \leq 1 \text{ for } |t| \leq \pi/3\}$$

where $p(t) = x_1 \cos t + x_2 \cos 2t + \cdots + x_m \cos mt$



Affine function

the image and inverse image of convex sets under an affine transformation

$$f(x) = \sum_i x_i a_i + b$$

are convex

Examples

- scaling, translation, projection
- solution set of linear matrix inequality ($A_i, B \in \mathbf{S}^p$)

$$\{x \mid x_1 A_1 + \cdots + x_n A_n \preceq B\}$$

- hyperbolic cone ($P \in \mathbf{S}^n_+$)

$$\{x \mid x^T P x \leq (c^T x)^2,\ c^T x \geq 0\}$$

Perspective and linear-fractional function

Perspective function $P : \mathbf{R}^{n+1} \to \mathbf{R}^n$:

$$P(x,t) = x/t, \qquad \mathbf{dom}\, P = \{(x,t) \mid t > 0\}$$

images and inverse images of convex sets under perspective are convex

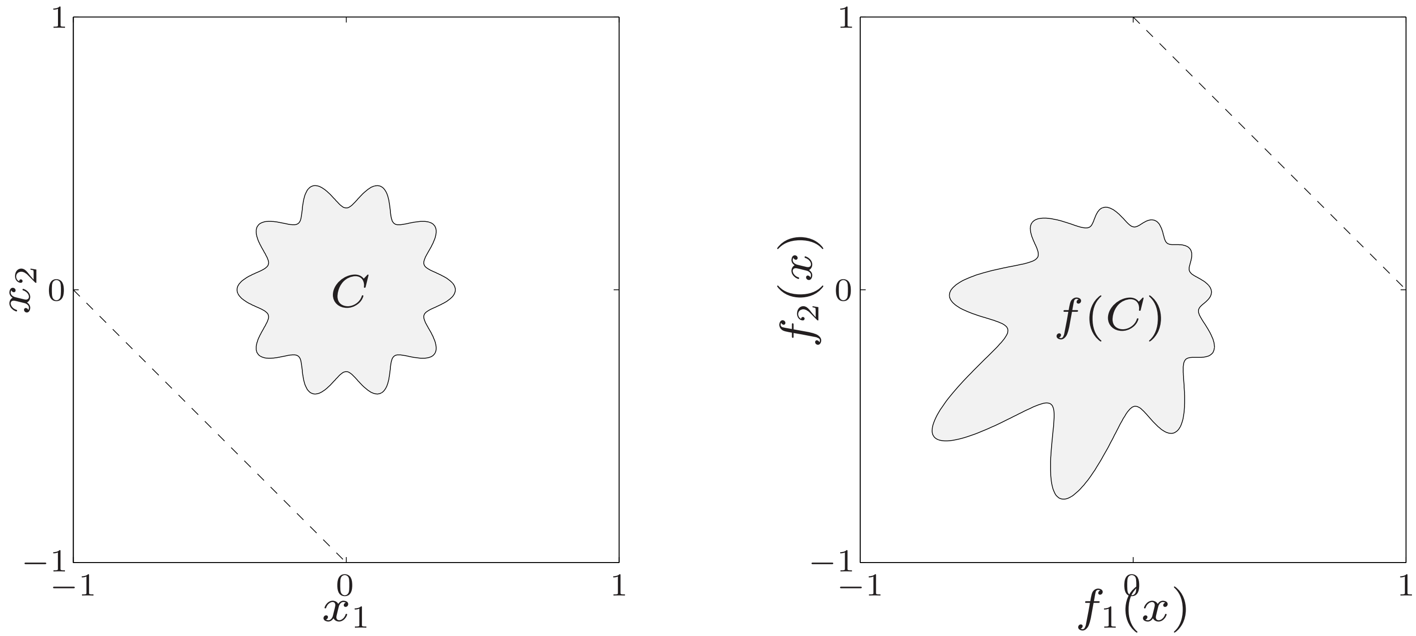
Linear-fractional function $f : \mathbf{R}^n \to \mathbf{R}^m$:

$$f(x) = \frac{Ax+b}{c^T x + d}, \qquad \mathbf{dom}\, f = \{x \mid c^T x + d > 0\}$$

images and inverse images of convex sets under linear-fractional functions are convex

Example

$$f(x) = \frac{1}{x_1 + x_2 + 1} x$$



3. Convex functions

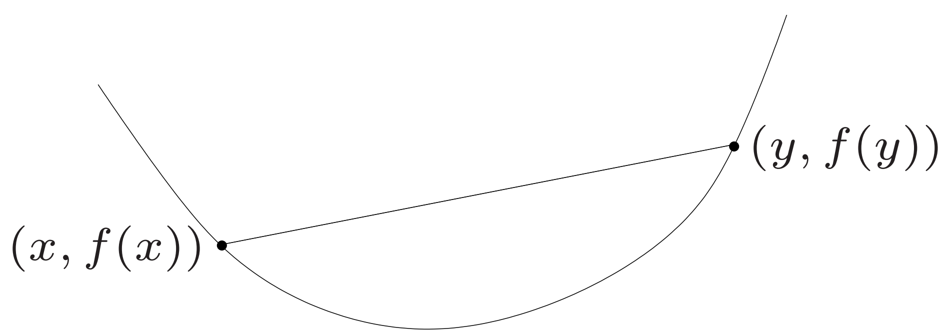
- basic properties and examples
- operations that preserve convexity

Definition

$f : \mathbf{R}^n \to \mathbf{R}$ is convex if its domain is a convex set and

$$f(\theta x + (1-\theta)y) \le \theta f(x) + (1-\theta)f(y)$$

for all $x, y \in \mathbf{dom}\, f$, $0 \le \theta \le 1$



f is concave if $-f$ is convex

Examples on R

convex:

- affine: $ax + b$ on **R**, for any $a, b \in$ **R**
- exponential: e^{ax}, for any $a \in$ **R**
- powers: x^α on $\mathbf{R}_{++}$, for $\alpha \geq 1$ or $\alpha \leq 0$
- powers of absolute value: $|x|^p$ on **R**, for $p \geq 1$
- negative entropy: $x \log x$ on $\mathbf{R}_{++}$

concave:

- affine: $ax + b$ on **R**, for any $a, b \in$ **R**
- powers: x^α on $\mathbf{R}_{++}$, for $0 \leq \alpha \leq 1$
- logarithm: $\log x$ on $\mathbf{R}_{++}$

Examples on $\mathbf{R}^n$ and $\mathbf{R}^{m \times n}$

Affine functions

$$f(x) = a^T x + b \quad \text{on } \mathbf{R}^n, \qquad f(X) = \mathbf{tr}(A^T X) + b \quad \text{on } \mathbf{R}^{m \times n}$$

Norms

- p-norms of vectors

$$\|x\|_p = (\sum_{i=1}^n |x_i|^p)^{1/p} \quad (p \geq 1), \qquad \|x\|_\infty = \max_k |x_k|$$

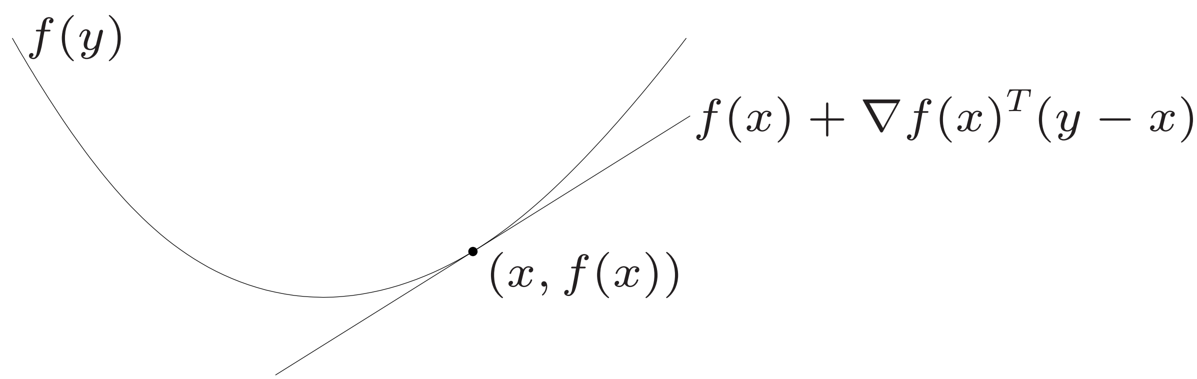
- spectral (maximum singular value) matrix norm

$$f(X) = \|X\|_2 = \sigma_{\max}(X) = (\lambda_{\max}(X^T X))^{1/2}$$

Differentiable convex functions

differentiable f is convex if and only if its domain is convex and

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



twice differentiable f is convex if and only if its domain is convex and

$$\nabla^2 f(x) \succeq 0 \quad \text{for all } x \in \mathbf{dom}\, f$$

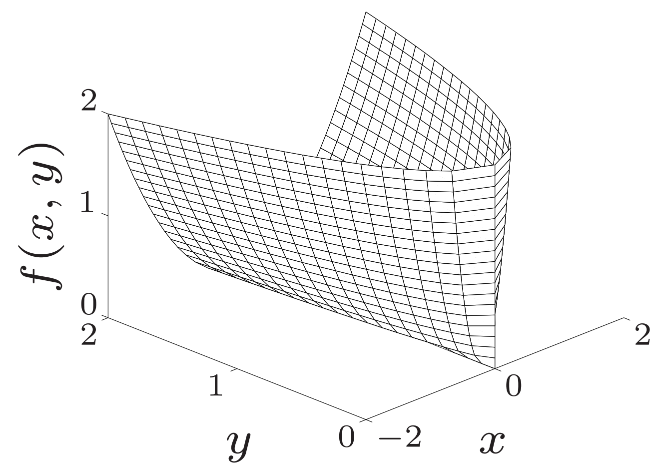
Examples

Quadratic function: $f(x) = (1/2)x^T Px + q^T x + r$ with $P \succeq 0$

Quadratic-over-linear: $f(x, y) = x^2/y$

$$\nabla^2 f(x, y) = \frac{2}{y^3} \begin{bmatrix} y \\ -x \end{bmatrix} \begin{bmatrix} y \\ -x \end{bmatrix}^T \succeq 0$$

convex for $y > 0$



Log-sum-exp: $f(x) = \log \sum_{k=1}^n \exp x_k$ is convex

Geometric mean: $f(x) = (\prod_{k=1}^n x_k)^{1/n}$ on $\mathbf{R}^n_{++}$ is concave

Operations that preserve convexity

practical methods for establishing convexity of a function

1. verify definition

2. for twice differentiable functions, show $\nabla^2 f(x) \succeq 0$

3. show that f is obtained from simple convex functions by operations that preserve convexity

   - nonnegative weighted sum
   - composition with affine function
   - pointwise maximum and supremum
   - composition
   - minimization
   - perspective

Positive weighted sum & composition with affine function

Nonnegative multiple: αf is convex if f is convex, $\alpha \geq 0$

Sum: $f_1 + f_2$ convex if f_1, f_2 convex (extends to infinite sums, integrals)

Composition with affine function: $f(Ax + b)$ is convex if f is convex

Examples

- log barrier for linear inequalities

$$f(x) = -\sum_{i=1}^{m} \log(b_i - a_i^T x)$$

- (any) norm of affine function: $f(x) = \|Ax + b\|$

Pointwise maximum

if f_1, . . . , f_m are convex, then

$$f(x) = \max\{f_1(x), \ldots, f_m(x)\}$$

is convex

Example: sum of r largest components of $x \in \mathbf{R}^n$

$$f(x) = x_{[1]} + x_{[2]} + \cdots + x_{[r]}$$

is convex ($x_{[i]}$ is ith largest component of x)

proof:

$$f(x) = \max\{x_{i_1} + x_{i_2} + \cdots + x_{i_r} \mid 1 \leq i_1 < i_2 < \cdots < i_r \leq n\}$$

Pointwise supremum

if $f(x, y)$ is convex in x for each $y \in \mathcal{A}$, then

$$g(x) = \sup_{y \in \mathcal{A}} f(x, y)$$

is convex

Example: maximum eigenvalue of symmetric matrix

$$\lambda_{\max}(X) = \sup_{\|y\|_2 = 1} y^T X y$$

Composition with scalar functions

composition of $g : \mathbf{R}^n \to \mathbf{R}$ and $h : \mathbf{R} \to \mathbf{R}$:

$$f(x) = h(g(x))$$

f is convex if

$$\begin{array}{l} g \text{ convex, } h \text{ convex and nondecreasing} \\ g \text{ concave, } h \text{ convex and nonincreasing} \end{array}$$

(if we assign $h(x) = \infty$ for $x \in \mathbf{dom}\, h$)

Examples

- $\exp g(x)$ is convex if g is convex
- $1/g(x)$ is convex if g is concave and positive

Vector composition

composition of $g : \mathbf{R}^n \to \mathbf{R}^k$ and $h : \mathbf{R}^k \to \mathbf{R}$:

$$f(x) = h(g(x)) = h(g_1(x), g_2(x), \ldots, g_k(x))$$

f is convex if

g_i convex, h convex and nondecreasing in each argument
g_i concave, h convex and nonincreasing in each argument

(if we assign $h(x) = \infty$ for $x \in \mathbf{dom}\, h$)

Examples

- $\sum_{i=1}^m \log g_i(x)$ is concave if g_i are concave and positive
- $\log \sum_{i=1}^m \exp g_i(x)$ is convex if g_i are convex

Minimization

if $f(x, y)$ is convex in (x, y) and C is a convex set, then

$$g(x) = \inf_{y \in C} f(x, y)$$

is convex

Example: distance to a convex set S:

$$\mathbf{dist}(x, S) = \inf_{y \in S} \|x - y\|$$

Perspective

the **perspective** of a function $f : \mathbf{R}^n \to \mathbf{R}$ is the function $g : \mathbf{R}^n \times \mathbf{R} \to \mathbf{R}$,

$$g(x,t) = tf(x/t)$$

g is convex if f is convex on $\mathbf{dom}\, g = \{(x,t) \mid x/t \in \mathbf{dom}\, f,\ t > 0\}$

Examples

- perspective of $f(x) = x^T x$ is quadratic-over-linear function

$$g(x,t) = \frac{x^T x}{t}$$

- perspective of negative logarithm $f(x) = -\log x$ is relative entropy

$$g(x,t) = t \log t - t \log x$$

Lecture 2

4. Convex optimization problems

- definition
- linear programming
- quadratic programming
- geometric programming
- modeling languages

Convex optimization problem

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq 0, \quad i = 1, \ldots, m \\ & Ax = b \end{array}$$

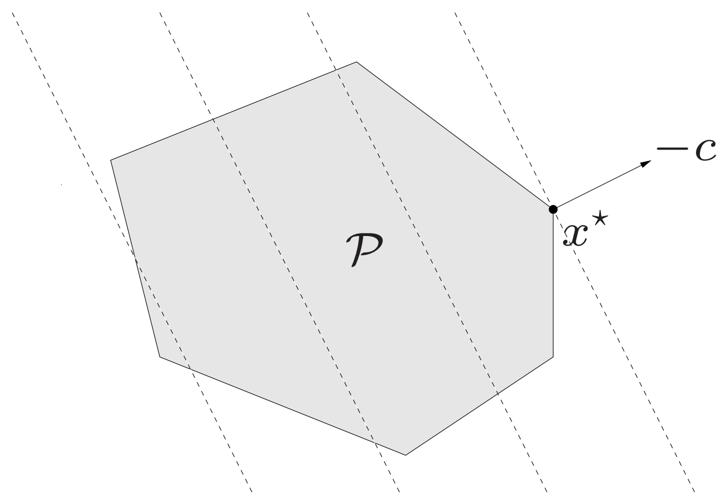
f_0, f_1, . . . , f_m are convex

- feasible set is convex
- locally optimal points are globally optimal
- tractable, both in theory and practice

Linear program (LP)

$$\begin{array}{ll} \text{minimize} & c^T x + d \\ \text{subject to} & Gx \preceq h \\ & Ax = b \end{array}$$

- convex problem with affine objective and constraint functions
- feasible set is a polyhedron



Piecewise-linear minimization

$$\text{minimize} \quad \max_{i=1,\ldots,m}(a_i^T x + b_i)$$

equivalent linear program

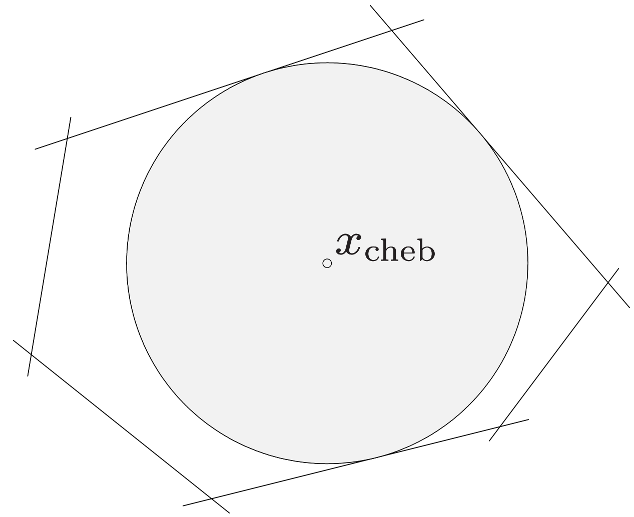
$$\begin{array}{ll} \text{minimize} & t \\ \text{subject to} & a_i^T x + b_i \leq t, \quad i = 1, \ldots, m \end{array}$$

variables x, $t \in \mathbf{R}$

Chebyshev center of a polyhedron

center of largest ball inscribed

$$\mathcal{P} = \{x \mid a_i^T x \leq b_i,\ i = 1, \ldots, m\}$$



- $a_i^T x \leq b_i$ if $\|x - x_c\|_2 \leq r$ if and only if

$$\sup\{a_i^T(x_c + u) \mid \|u\|_2 \leq r\} = a_i^T x_c + r\|a_i\|_2 \leq b_i$$

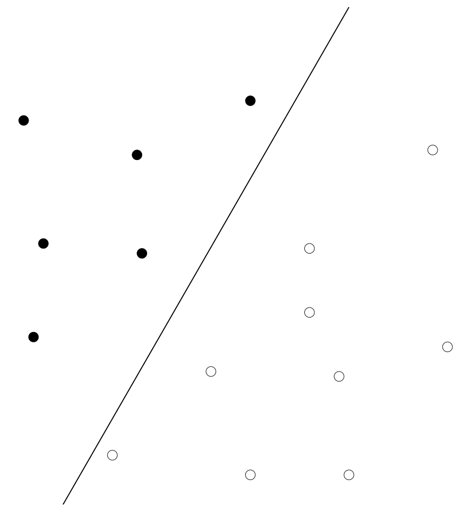
- hence, center x_c and radius r can be determined by solving the LP

$$\begin{array}{ll} \text{maximize} & r \\ \text{subject to} & a_i^T x_c + r\|a_i\|_2 \leq b_i, \quad i = 1, \ldots, m \end{array}$$

Linear discrimination

separate two sets of points $\{x_1, \ldots, x_N\}$, $\{y_1, \ldots, y_M\}$ by a hyperplane

$$a^T x_i + b > 0, \quad i = 1, \ldots, N$$
$$a^T y_i + b < 0 \quad i = 1, \ldots, M$$

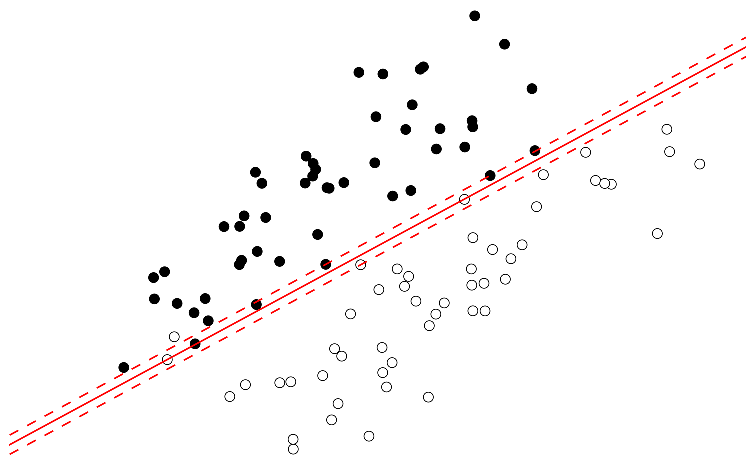


homogeneous in a, b, hence equivalent to the linear inequalities (in a, b)

$$a^T x_i + b \geq 1, \quad i = 1, \ldots, N, \qquad a^T y_i + b \leq -1, \quad i = 1, \ldots, M$$

Approximate linear separation of non-separable sets

$$\text{minimize} \quad \sum_{i=1}^{N} \max\{0, 1 - a^T x_i - b\} + \sum_{i=1}^{M} \max\{0, 1 + a^T y_i + b\}$$

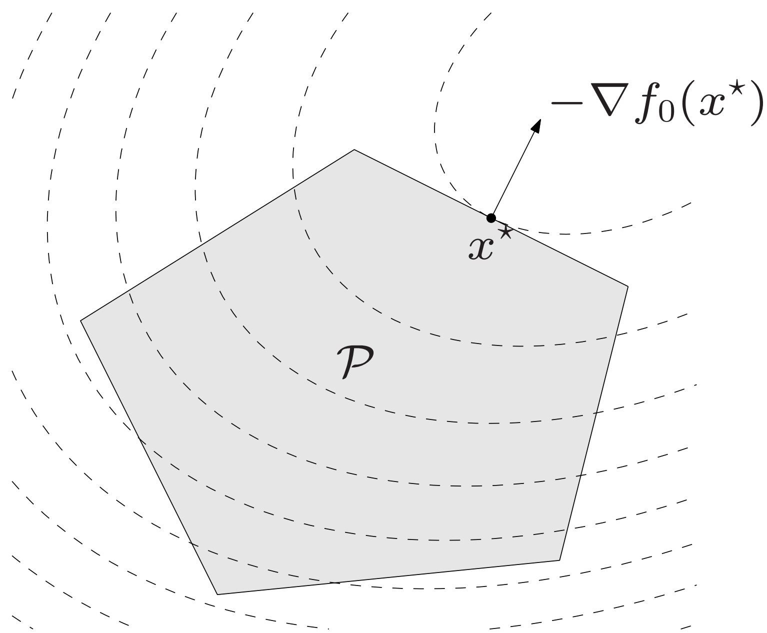


- a convex problem in a, b; equivalent to an LP
- can be interpreted as a heuristic for minimizing #misclassified points

Quadratic program (QP)

$$\begin{array}{ll} \text{minimize} & (1/2)x^T P x + q^T x + r \\ \text{subject to} & Gx \preceq h \\ & Ax = b \end{array}$$

- $P \in \mathbf{S}^n_+$, so objective is convex quadratic
- minimize a convex quadratic function over a polyhedron



Linear program with random cost

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Gx \preceq h \end{array}$$

- c is random vector with mean $\bar{c}$ and covariance Σ
- hence, $c^T x$ is random variable with mean $\bar{c}^T x$ and variance $x^T \Sigma x$

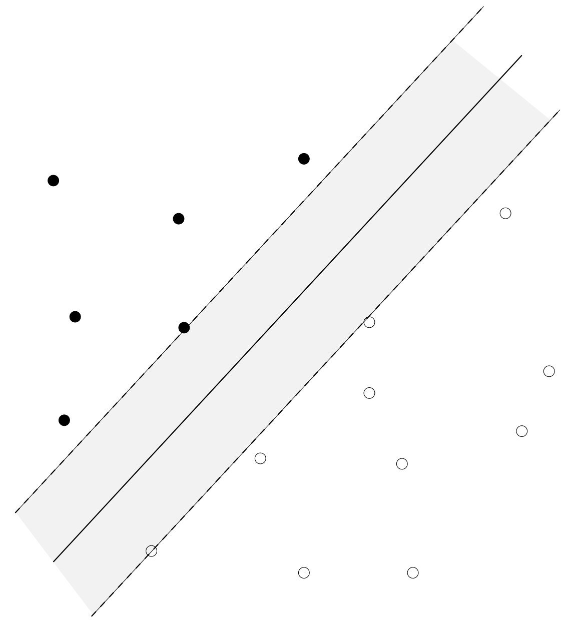
Trade-off between expected cost and variance (risk):

$$\begin{array}{ll} \text{minimize} & \mathbf{E}\, c^T x + \gamma\, \mathbf{var}(c^T x) = \bar{c}^T x + \gamma x^T \Sigma x \\ \text{subject to} & Gx \preceq h \end{array}$$

$\gamma > 0$ is risk aversion parameter

Robust linear discrimination

$$\begin{aligned}\mathcal{H}_1 &= \{z \mid a^T z + b = 1\}\\ \mathcal{H}_2 &= \{z \mid a^T z + b = -1\}\end{aligned}$$



distance between hyperplanes is $2/\|a\|_2$

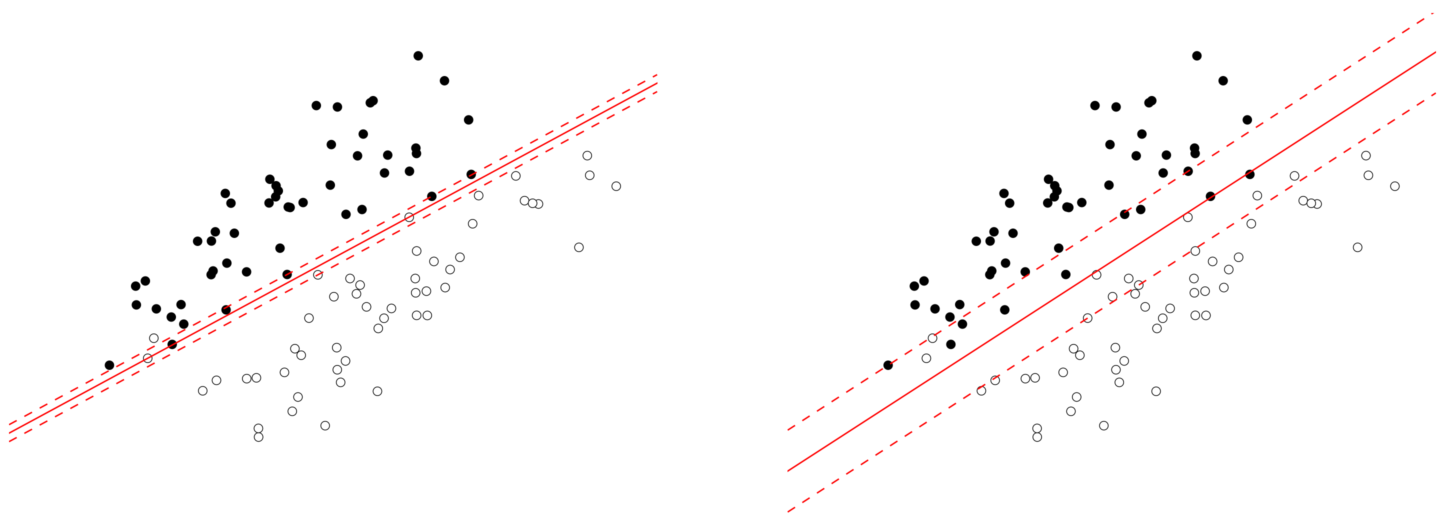
to separate two sets of points by maximum margin,

$$\begin{array}{ll}\text{minimize} & \|a\|_2^2 = a^T a\\ \text{subject to} & a^T x_i + b \geq 1, \quad i = 1, \ldots, N\\ & a^T y_i + b \leq -1, \quad i = 1, \ldots, M\end{array}$$

a quadratic program in a, b

Support vector classifier

$$\text{min.} \quad \gamma\|a\|_2^2 + \sum_{i=1}^{N} \max\{0, 1 - a^T x_i - b\} + \sum_{i=1}^{M} \max\{0, 1 + a^T y_i + b\}$$



$\gamma = 0$ $\gamma = 10$

equivalent to a QP

Geometric programming

Posynomial function:

$$f(x) = \sum_{k=1}^{K} c_k x_1^{a_{1k}} x_2^{a_{2k}} \cdots x_n^{a_{nk}}, \qquad \mathbf{dom}\, f = \mathbf{R}^n_{++}$$

with $c_k > 0$

Geometric program (GP)

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

with f_i posynomial

Geometric program in convex form

change variables to

$$y_i = \log x_i,$$

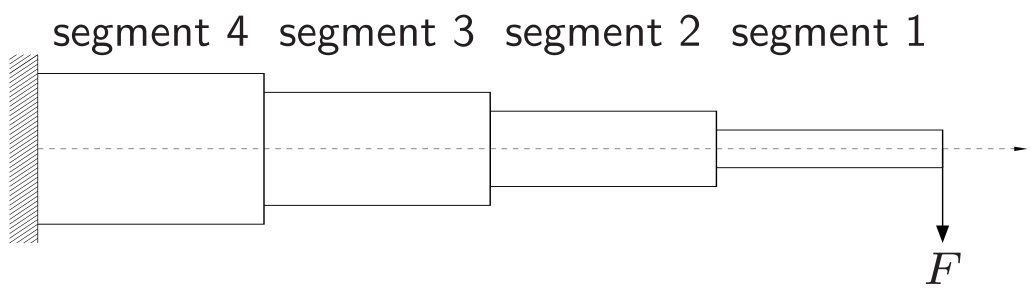
and take logarithm of cost, constraints

Geometric program in convex form:

$$\begin{array}{ll} \text{minimize} & \log\left(\sum_{k=1}^{K} \exp(a_{0k}^T y + b_{0k})\right) \\ \text{subject to} & \log\left(\sum_{k=1}^{K} \exp(a_{ik}^T y + b_{ik})\right) \leq 0, \quad i = 1, \ldots, m \end{array}$$

$b_{ik} = \log c_{ik}$

Design of cantilever beam



- N segments with unit lengths, rectangular cross-sections of size $w_i \times h_i$
- given vertical force F applied at the right end

design problem

minimize	total weight
subject to	upper & lower bounds on w_i, h_i
	upper bound & lower bounds on aspect ratios h_i/w_i
	upper bound on stress in each segment
	upper bound on vertical deflection at the end of the beam

variables: w_i, h_i for $i = 1, \ldots, N$

Objective and constraint functions

- total weight $w_1 h_1 + \cdots + w_N h_N$ is posynomial
- aspect ratio h_i/w_i and inverse aspect ratio w_i/h_i are posynomials
- maximum stress in segment i is given by $6iF/(w_i h_i^2)$, a posynomial
- the vertical deflection y_i and slope v_i of central axis at the right end of segment i are defined recursively as

$$\begin{aligned} v_i &= 12(i-1/2)\frac{F}{E w_i h_i^3} + v_{i+1} \\ y_i &= 6(i-1/3)\frac{F}{E w_i h_i^3} + v_{i+1} + y_{i+1} \end{aligned}$$

for $i = N, N-1, \ldots, 1$, with $v_{N+1} = y_{N+1} = 0$ (E is Young's modulus)

v_i and y_i are posynomial functions of w, h

Modeling software

Modeling packages for convex optimization (Matlab)

- Yalmip
- CVX

assist in formulating convex problems by automating two tasks:

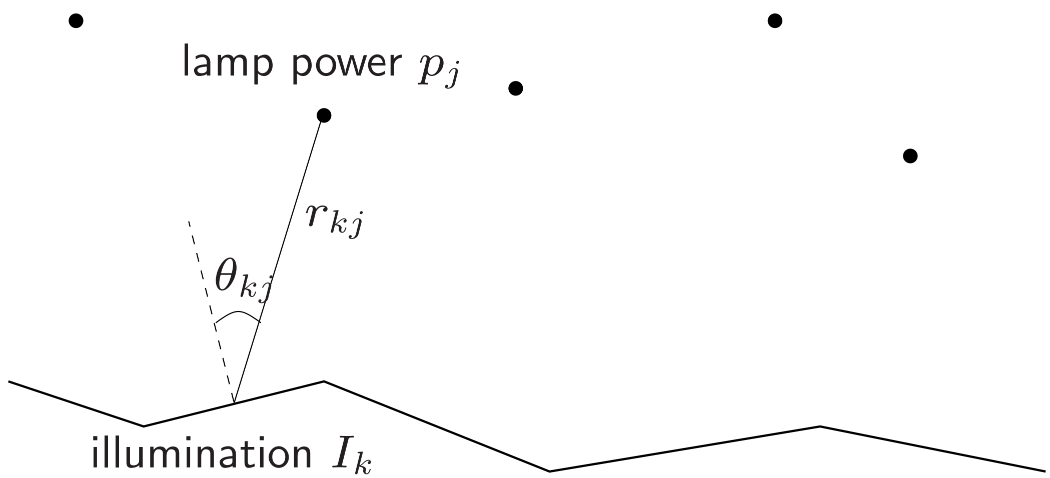
- verifying convexity, using convex calculus rules
- transforming problem in input format required by standard solvers

Related packages

- general purpose optimization modeling: AMPL, GAMS
- convex piecewise-linear optimization: CVXOPT (Python)
- generalized geometric programming: ggplab (Matlab)

Example

m lamps illuminating n (small, flat) patches



intensity I_k at patch k depends linearly on lamp powers p_j: $I_k = a_k^T p$

Problem: achieve desired illumination $I_{\text{des}} = 1$ with bounded lamp powers

$$\begin{array}{ll} \text{minimize} & \max_{k=1,\ldots,n} |\log(a_k^T p)| \\ \text{subject to} & 0 \leq p_j \leq 1, \quad j = 1, \ldots, m \end{array}$$

Convex formulation

$$\begin{array}{ll} \text{minimize} & \max_{k=1,\ldots,n} h(a_k^T p) \\ \text{subject to} & 0 \leq p_j \leq 1, \quad j = 1, \ldots, m \end{array}$$

with $h(u) = \max\{u, 1/u\}$ (maximum of convex functions, hence convex)

CVX code

```
cvx_begin
    variable x(m)
    minimize( max( [ A*x;  inv_pos(A*x) ] ) )
    subject to
        x >= 0
        x <= 1
cvx_end
```

5. Cone programming

- generalized inequality constraints
- second-order cone programming
- semidefinite programming

Definition

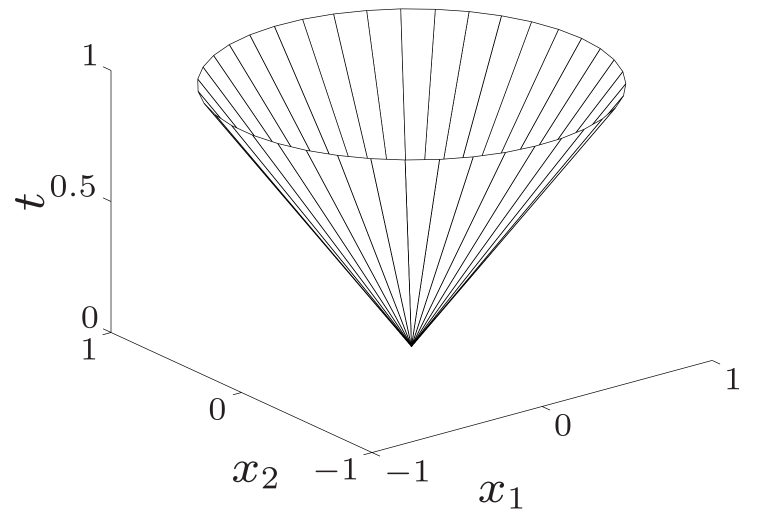
$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Gx \preceq_K h \\ & Ax = b \end{array}$$

- $y \preceq_K z$ means $z - y \in K$, where K is a proper convex cone
- extends linear programming ($K = \mathbf{R}_+^m$) to nonpolyhedral cones
- (duality) theory and algorithms very similar to linear programming

Second-order cone programming

Second-order cone

$$C_{m+1} = \{(x,t) \in \mathbf{R}^m \times \mathbf{R} \mid \|x\| \le t\}$$



Second-order cone program

$$\begin{array}{ll} \text{minimize} & f^T x \\ \text{subject to} & \|A_i x + b_i\|_2 \le c_i^T x + d_i, \quad i = 1, \ldots, m \\ & Fx = g \end{array}$$

inequality constraints require $(A_i x + b_i, c_i^T x + d_i) \in C_{m_i+1}$

Stochastic linear programming

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

- assume a_i is Gaussian with mean $\bar{a}_i$, covariance Σ_i
- LP with chance constraints

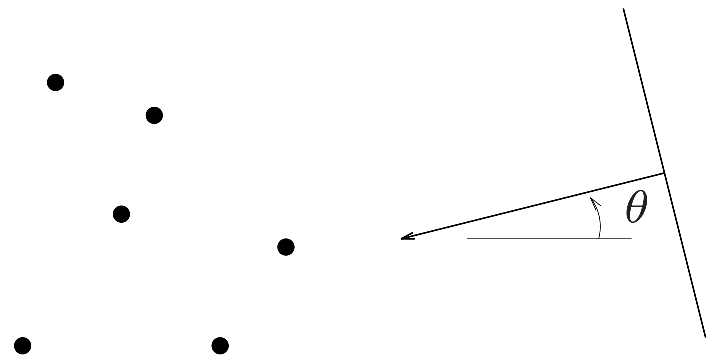
$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & \mathbf{prob}(a_i^T x \leq b_i) \geq \eta, \quad i = 1, \ldots, m, \end{array}$$

- for $\eta \geq 1/2$, equivalent to the SOCP

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & \bar{a}_i^T x + \Phi^{-1}(\eta) \|\Sigma_i^{1/2} x\|_2 \leq b_i, \quad i = 1, \ldots, m \end{array}$$

where $\Phi(x) = (1/\sqrt{2\pi}) \int_{-\infty}^{x} e^{-t^2/2}\, dt$ is CDF of $\mathcal{N}(0,1)$

Antenna array antenna beamforming



- n omnidirectional antenna elements at positions $(x_i, y_i) \in \mathbf{R}^2$
- unit plane wave ($\lambda = 1$) incident from angle θ
- (demodulated) signal in ith element is $e^{j(x_i \cos\theta + y_i \sin\theta - \omega t)}$ ($j = \sqrt{-1}$)
- combine signals using complex weights w_i to get array output

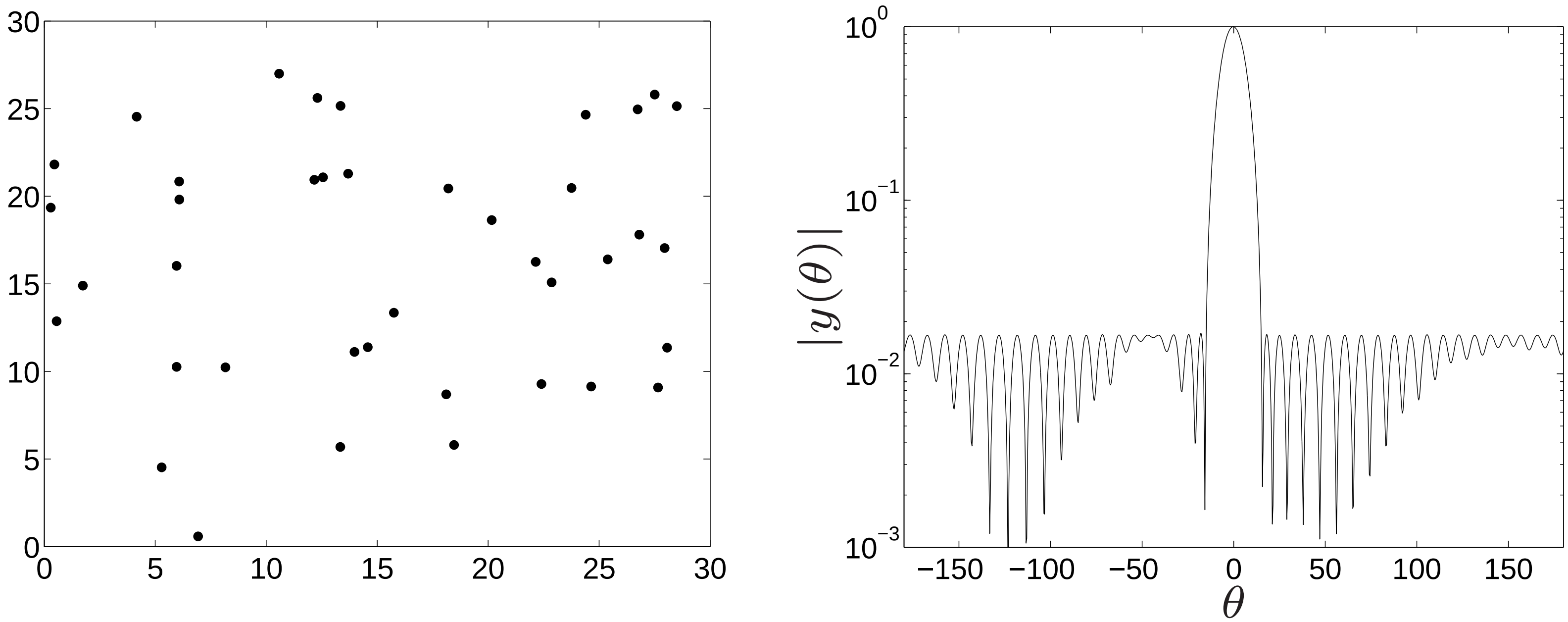
$$y(\theta) = \sum_{i=1}^{n} w_i e^{j(x_i \cos\theta + y_i \sin\theta)}$$

Sidelobe level minimization

$$\begin{array}{ll} \text{minimize} & \sup_{|\theta|>\Delta} |y(\theta)| \\ \text{subject to} & y(\theta_{\text{tar}}) = 1 \end{array}$$

θ_{tar} is target direction; 2Δ is beamwidth; $|y|$ is modulus of complex y

Example (30 elements randomly placed in a plane)



SOCP formulation

$$\begin{array}{ll} \text{minimize} & t \\ \text{subject to} & |y(\theta_k)| \leq t, \quad k = 1, \ldots, N \\ & y(\theta_{\text{tar}}) = 1 \end{array}$$

- $\theta_1, \ldots, \theta_N$ are discretized angles outside beam
- $y(\theta)$ is a complex linear function of w,

$$|y(\theta)| = \left\| \begin{bmatrix} \Re\, y(\theta) \\ \Im\, y(\theta) \end{bmatrix} \right\|_2$$

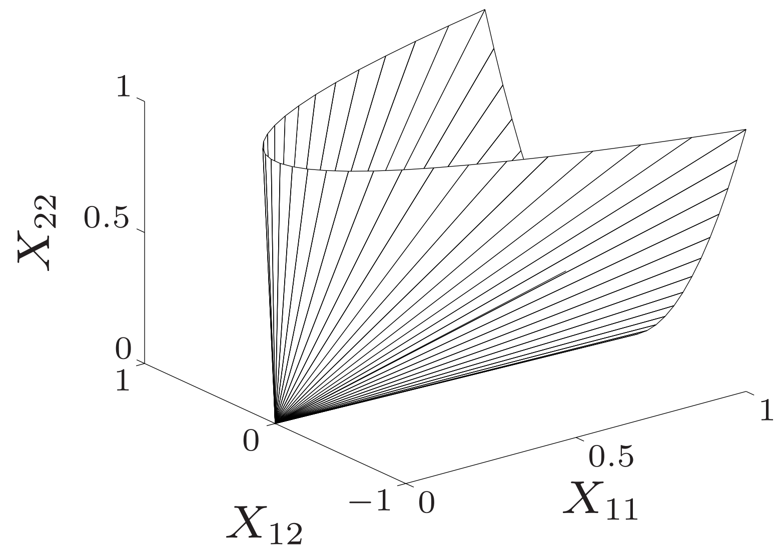
CVX code

```
cvx_begin
    variable w(n) complex
    minimize( max(abs(A*w) )
    subject to Atar*w == 1
cvx_end
```

Semidefinite programming

Positive semidefinite cone

$$\mathbf{S}_+^m = \{X \in \mathbf{S}^m \mid X \succeq 0\}$$



Semidefinite programming

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & x_1 A_1 + \cdots + x_n A_n \preceq B \end{array}$$

constraint requires $B - x_1 A_1 - \cdots - x_n A_n \in \mathbf{S}_+^m$

SOCP as SDP

SOCP

$$\begin{array}{ll} \text{minimize} & f^T x \\ \text{subject to} & \|A_i x + b_i\|_2 \leq c_i^T x + d_i, \quad i = 1, \ldots, m \end{array}$$

SDP

$$\begin{array}{ll} \text{minimize} & f^T x \\ \text{subject to} & \left[\begin{array}{cc} (c_i^T x + d_i) I & A_i x + b_i \\ (A_i x + b_i)^T & c_i^T x + d_i \end{array} \right] \succeq 0, \quad i = 1, \ldots, m \end{array}$$

Eigenvalue minimization

$$\text{minimize} \quad \lambda_{\max}(A(x))$$

where $A(x) = A_0 + x_1 A_1 + \cdots + x_n A_n$ (with given $A_i \in \mathbf{S}^k$)

equivalent SDP

$$\begin{array}{ll} \text{minimize} & t \\ \text{subject to} & A(x) \preceq tI \end{array}$$

- variables $x \in \mathbf{R}^n$, $t \in \mathbf{R}$
- follows from

$$\lambda_{\max}(A) \leq t \quad \Longleftrightarrow \quad A \preceq tI$$

Matrix norm minimization

$$\text{minimize} \quad \|A(x)\|_2 = \left(\lambda_{\max}(A(x)^T A(x))\right)^{1/2}$$

where $A(x) = A_0 + x_1 A_1 + \cdots + x_n A_n$ (with given $A_i \in \mathbf{R}^{p \times q}$)

equivalent SDP

$$\begin{array}{ll} \text{minimize} & t \\ \text{subject to} & \left[\begin{array}{cc} tI & A(x) \\ A(x)^T & tI \end{array}\right] \succeq 0 \end{array}$$

- variables $x \in \mathbf{R}^n$, $t \in \mathbf{R}$
- constraint follows from

$$\begin{array}{rcl} \|A\|_2 \leq t & \Longleftrightarrow & A^T A \preceq t^2 I, \quad t \geq 0 \\ & \Longleftrightarrow & \left[\begin{array}{cc} tI & A \\ A^T & tI \end{array}\right] \succeq 0 \end{array}$$

Semidefinite relaxations

Boolean least-squares

$$\begin{array}{ll} \text{minimize} & \|Ax - b\|_2^2 \\ \text{subject to} & x_i^2 = 1, \quad i = 1, \ldots, n. \end{array}$$

- a basic problem in digital communciations
- non-convex, very hard to solve exactly

Equivalent formulation

$$\begin{array}{ll} \text{minimize} & \mathbf{tr}(A^T A Z) - 2b^T A z + b^T b \\ \text{subject to} & Z_{ii} = 1, \quad i = 1, \ldots, n \\ & Z = zz^T \end{array}$$

follows from $\|Az - b\|_2^2 = \mathbf{tr}(A^T A Z) - 2b^T A z + b^T b$ if $Z = zz^T$

Semidefinite relaxation

replace constraint $Z = zz^T$ with $Z \succeq zz^T$

$$\begin{array}{ll} \text{minimize} & \mathbf{tr}(A^TAZ) - 2b^TAz + b^Tb \\ \text{subject to} & Z_{ii} = 1, \quad i = 1, \ldots, n \\ & \begin{bmatrix} Z & z \\ z^T & 1 \end{bmatrix} \succeq 0 \end{array}$$

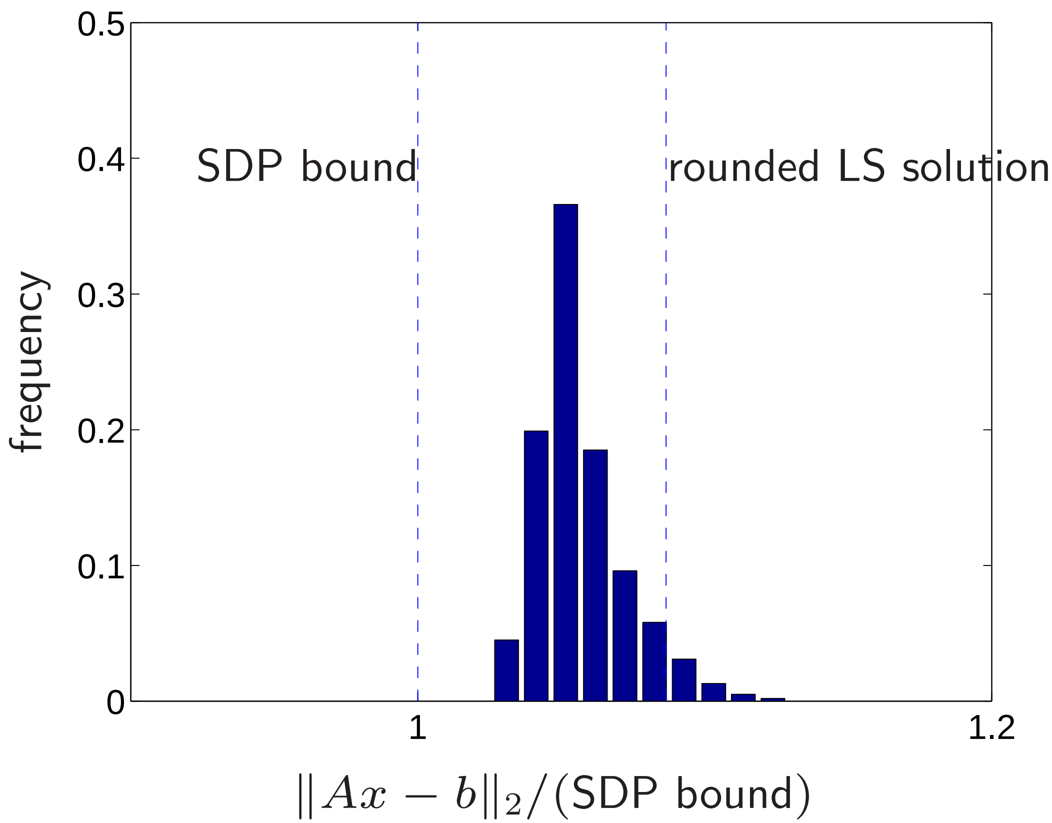
- an SDP with variables Z, z
- optimal value is a lower bound for Boolean LS optimal value
- rounding Z, z gives suboptimal solution for Boolean LS

Randomized rounding

- generate vector from $\mathcal{N}(z, Z - zz^T)$
- round components to ± 1

Example

- (randomly chosen) parameters $A \in \mathbf{R}^{150\times 100}$, $b \in \mathbf{R}^{150}$
- $x \in \mathbf{R}^{100}$, so feasible set has $2^{100} \approx 10^{30}$ points



distribution of randomized solutions based on SDP solution

Chebyshev inequalities

Classical inequality: for random scalar X with $\mathbf{E}\,X = 0$, $\mathbf{E}\,X^2 = \sigma^2$

$$\mathbf{prob}(|X| < 1) \geq 1 - \sigma^2$$

Generalized inequality: for random vector X with $\mathbf{E}\,X = a$, $\mathbf{E}\,XX^T = S$

$$\mathbf{prob}(X^T A_i X + 2b_i^T X + c_i < 0,\ i = 1, \ldots, m\} \geq f$$

where f is the optimal value of the SDP (with variables P, q, r, τ_i)

$$\begin{array}{ll} \text{maximize} & 1 - \mathbf{tr}(SP) - 2a^T q - r \\ \text{subject to} & \left[\begin{array}{cc} P & q \\ q^T & r-1 \end{array}\right] \succeq \tau_i \left[\begin{array}{cc} A_i & b_i \\ b_i^T & c_i \end{array}\right], \quad \tau_i \geq 0 \quad i = 1, \ldots, m \\ & \left[\begin{array}{cc} P & q \\ q^T & r \end{array}\right] \succeq 0 \end{array}$$

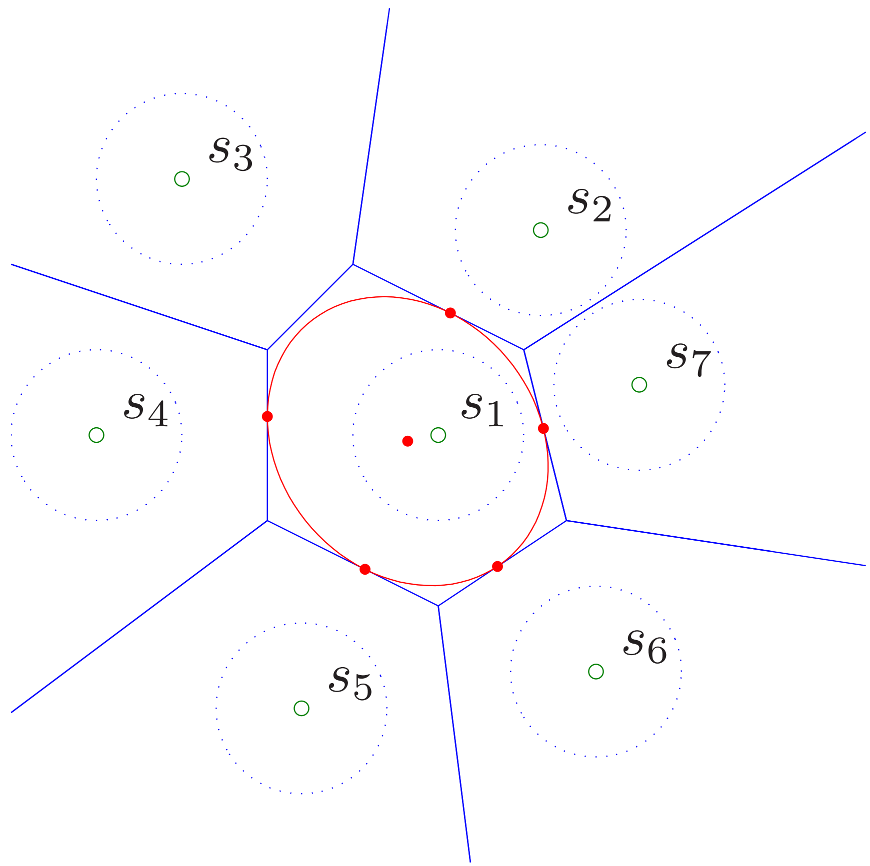
Detection example

$$x = s + v$$

- $x \in \mathbf{R}^n$: received signal
- s: transmitted signal $s \in \{s_1, s_2, \ldots, s_N\}$ (one of N possible symbols)
- v: noise with $\mathbf{E}\, v = 0$, $\mathbf{E}\, vv^T = \sigma^2 I$

Detection problem: given observed value of x, estimate s

Example $(N = 7)$: bound on probability of correct detection of s_1 is 0.205



dots: distribution with probability of correct detection 0.205

Summary: old and new view of convex optimization

Traditional: special case of nonlinear programming with interesting theory

New: extension of LP, as tractable but substantially more general

Reflected in cone programming notation:

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax \preceq b \end{array}$$

'$\preceq$' is inequality with respect to non-polyhedral convex cone

Lecture 3

6. Robust optimization

- optimization with uncertain data
- robust linear programming
- robust estimation

Robust convex optimization

$$
\begin{array}{ll}
\text{minimize} & \sup_{\theta \in \mathcal{A}_0} f_0(x, \theta) \\
\text{subject to} & \sup_{\theta \in \mathcal{A}_i} f_i(x, \theta) \leq 0, \quad i = 1, \ldots, m
\end{array}
$$

- f_i convex in x for fixed θ
- θ is an unknown parameter
- tractability depends on choice of $\mathcal{A}_i$

El Ghaoui & Lebret (1997), Ben-Tal & Nemirovski (1998, 2000), Bertsimas & Sim, . . .

Robust linear programming

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

with

$$\mathcal{A}_i = \{\bar{a}_i + P_i u \mid \|u\|_2 \leq 1\} \qquad (\bar{a}_i \in \mathbf{R}^n, \quad P_i \in \mathbf{R}^{n \times n})$$

Equivalent SOCP

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & \bar{a}_i^T x + \|P_i^T x\|_2 \leq b_i, \quad i = 1, \ldots, m \end{array}$$

(follows from $\sup_{\|u\|_2 \leq 1} (\bar{a}_i + P_i u)^T x = \bar{a}_i^T x + \|P_i^T x\|_2$)

Robust least-squares

$$\text{minimize} \quad \sup_{\|u\|_2 \leq 1} \|(A_0 + u_1 A_1 + \cdots + u_p A_p)x - b\|_2$$

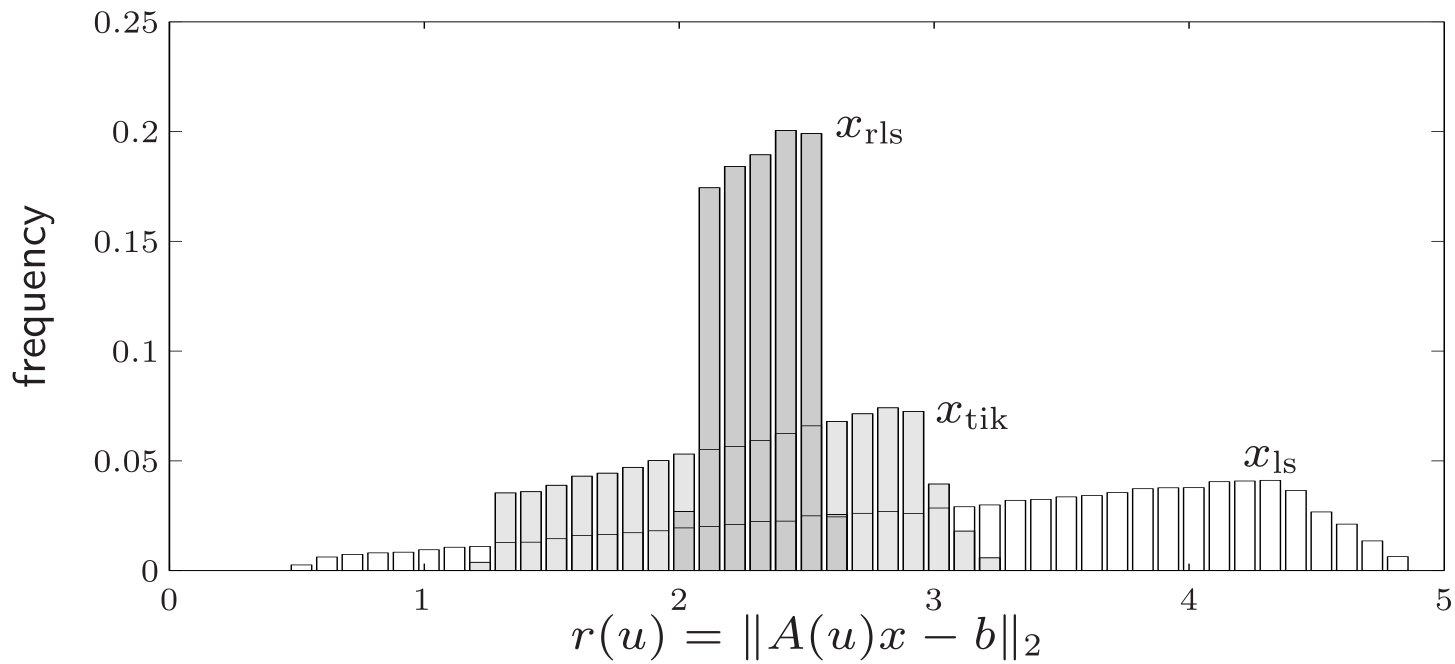
can be solved as a semidefinite program

$$\begin{array}{ll} \text{minimize} & t + \lambda \\ \text{subject to} & \left[\begin{array}{ccc} I & P(x) & q(x) \\ P(x)^T & \lambda I & 0 \\ q(x)^T & 0 & t \end{array}\right] \succeq 0 \end{array}$$

where

$$P(x) = \left[\begin{array}{cccc} A_1 x & A_2 x & \cdots & A_p x \end{array}\right], \qquad q(x) = A_0 x - b$$

Example ($p = 2$)



- x_{ls} minimizes $\|A_0x - b\|_2^2$ (nominal solution)
- x_{tik} minimizes $\|A_0x - b\|_2^2 + \|x\|_2^2$ (Tikhonov regularized solution)
- x_{rls} minimizes $\sup_{\|u\|_2} \|A(u)x - b\|_2$ (robust solution)

7. Polynomial optimization

- sums of squares
- semidefinite relaxations

Sums of squares and semidefinite programming

Sum of squares

$$x^T p(t) = \sum_{k=1}^{s} (y_k^T q(t))^2 = q(t)^T \left(\sum_{k=1}^{s} y_k y_k^T \right) q(t)$$

- p, q: basis functions (polynomials, trigonometric polynomials, . . .)
- independent variable t can be one- or multidimensional

Equivalent SDP formulation

$x^T p(t)$ is a sum of squares if and only if

$$x^T p(t) = q(t)^T X q(t), \qquad X \succeq 0$$

an SDP constraint in x and X

Applications

$$x^T p(t) = q(t)^T X q(t), \qquad X \succeq 0$$

- A *sufficient* condition for nonnegativity of $x^T p(t)$, useful in nonconvex polynomial optimization in several variables

  (Parrilo, Lasserre, . . .)

- In some nontrivial cases, *necessary and sufficient*.

Cosine polynomials and spectral mask constraints

$$f(\omega) = x_0 + x_1 \cos\omega + \cdots + x_{2n} \cos 2n\omega \geq 0$$

Sum of squares theorem: $f(\omega) \geq 0$ for $\alpha \leq \omega \leq \beta$ if and only if

$$f(\omega) = g_1(\omega)^2 + s(\omega) g_2(\omega)^2$$

- g_1, g_2: cosine polynomials of degree n and $n-1$
- $s(\omega) = (\cos\omega - \cos\beta)(\cos\alpha - \cos\omega)$ is a given weight function

Equivalent SDP formulation: $f(\omega) \geq 0$ for $\alpha \leq \omega \leq \beta$ if and only if

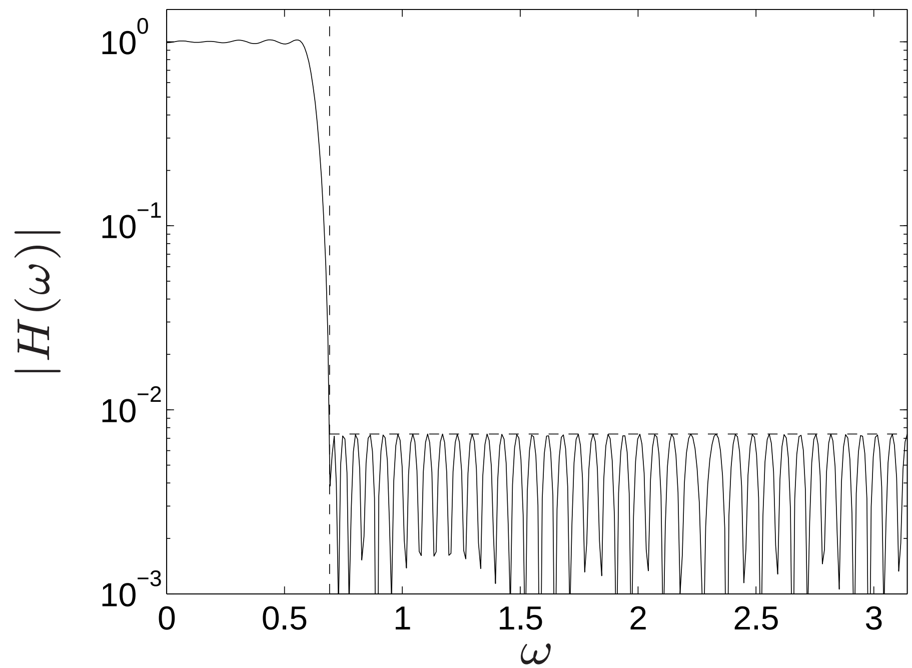
$$x^T p(\omega) = q_1(\omega)^T X_1 q_1(\omega) + s(\omega) q_2(\omega)^T X_2 q_2(\omega), \quad X_1 \succeq 0, \quad X_2 \succeq 0$$

p, q_1, q_2: basis vectors $(1, \cos\omega, \cos(2\omega), \ldots)$ up to order $2n$, n, $n-1$

Example: Linear-phase Nyquist filter

$$\text{minimize} \quad \sup_{\omega \geq \omega_s} |h_0 + h_1 \cos \omega + \cdots + h_{2n} \cos 2n\omega|$$

with $h_0 = 1/M$, $h_{kM} = 0$ for positive integer k



(Example with $n = 25$, $M = 5$, $\omega_s = 0.69$)

SDP formulation

$$\begin{array}{ll} \text{minimize} & t \\ \text{subject to} & -t \leq H(\omega) \leq t, \quad \omega_{\rm s} \leq \omega \leq \pi \end{array}$$

Equivalent SDP

$$\begin{array}{ll} \text{minimize} & t \\ \text{subject to} & t - H(\omega) = q_1(\omega)^T X_1 q_1(\omega) + s(\omega) q_2(\omega)^T X_2 q_2(\omega) \\ & t + H(\omega) = q_1(\omega)^T X_3 q_1(\omega) + s(\omega) q_2(\omega)^T X_3 q_2(\omega) \\ & X_1 \succeq 0, \quad X_2 \succeq 0, \quad X_3 \succeq 0, \quad X_4 \succeq 0 \end{array}$$

Variables t, h_i $(i \neq kM)$, 4 matrices X_i of size roughly n

8. 1-Norm heuristics

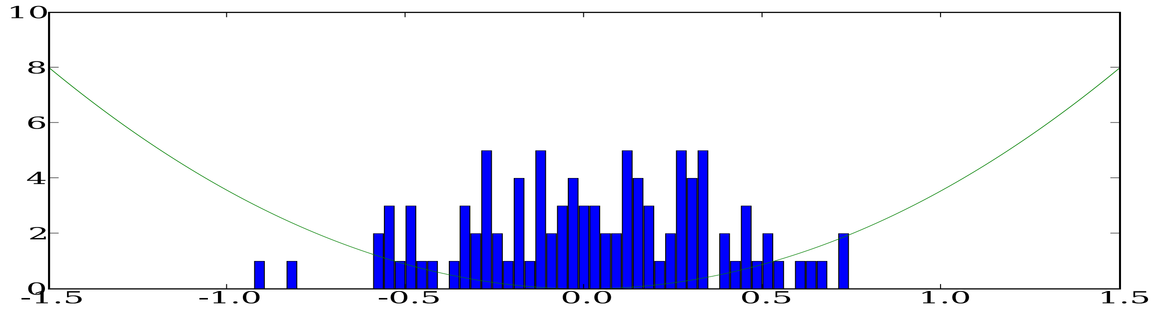
- robust regression
- regressor selection
- total variation

Norm approximation

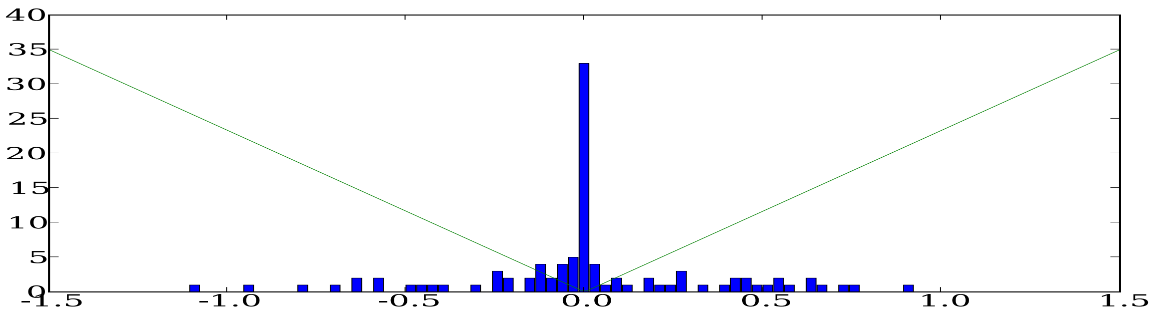
minimize $\|Ax - b\|_2$ minimize $\|Ax - b\|_1$

example (A is 100×30): histograms of residuals

2-norm



1-norm

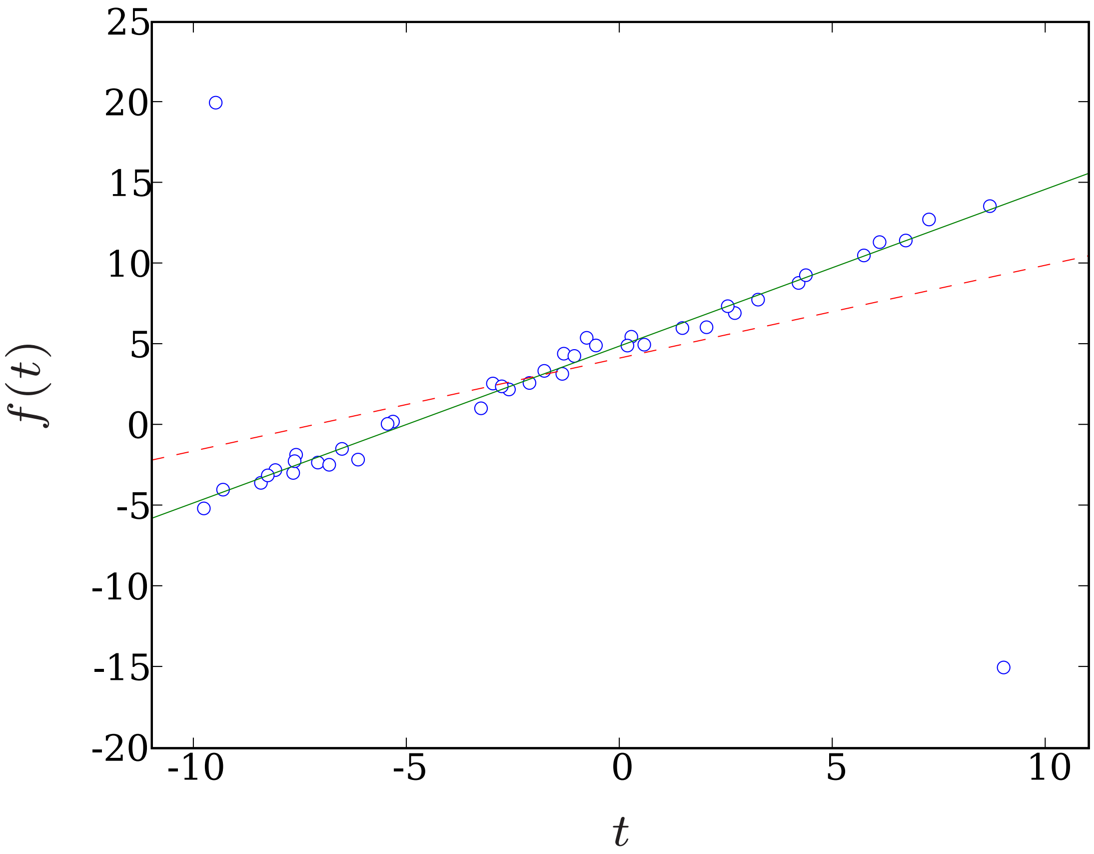


note large number of zero residuals in 1-norm solution

Applications

- find solution that satisfies many of a set of inconsistent equations
- robust regression
- total variation signal reconstruction
- sparse regressor selection

Robust regression



- 42 points t_i, y_i (circles), including two outliers
- function $f(t) = \alpha + \beta t$ fitted using 2-norm (dashed) and 1-norm

Signal reconstruction

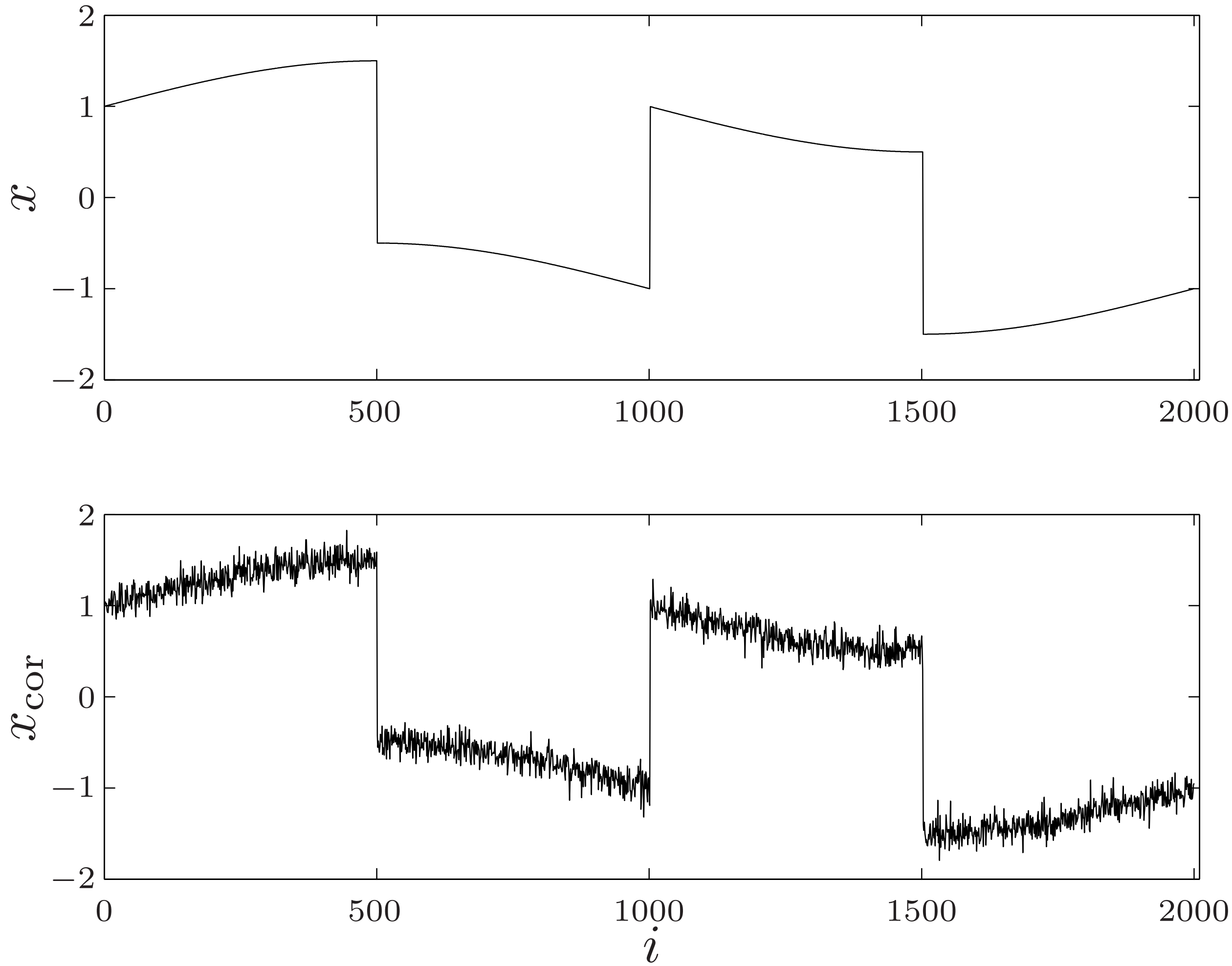
$$\text{minimize} \quad \|\hat{x} - x_{\text{cor}}\|_2 + \gamma\phi(\hat{x})$$

- $x \in \mathbf{R}^n$ is unknown signal
- $x_{\text{cor}} = x + v$ is (known) corrupted version of x, with additive noise v
- variable $\hat{x}$ (reconstructed signal) is estimate of x
- $\phi : \mathbf{R}^n \to \mathbf{R}$ is regularization function or smoothing objective

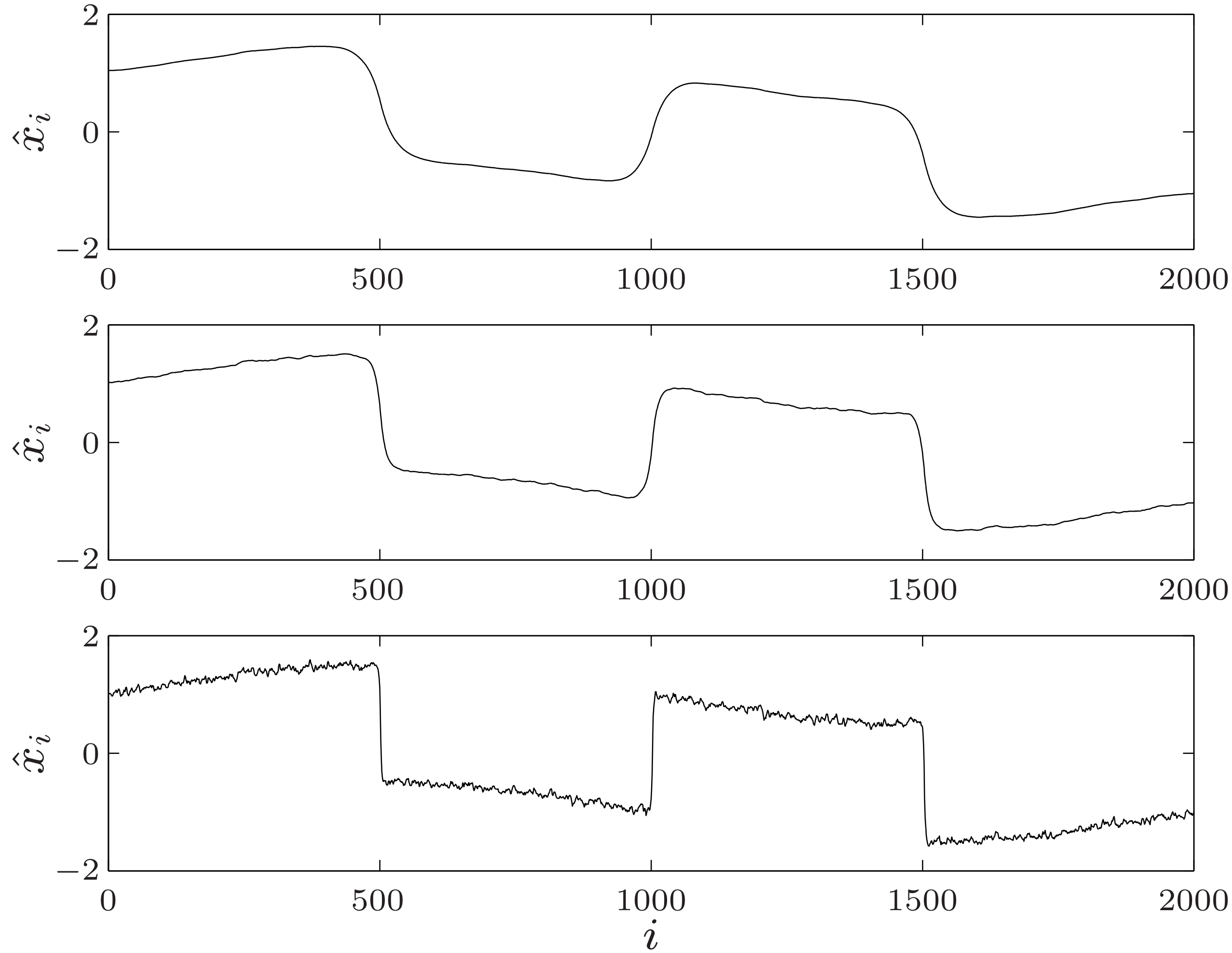
examples: quadratic smoothing, total variation smoothing:

$$\phi_{\text{quad}}(\hat{x}) = \sum_{i=1}^{n-1} (\hat{x}_{i+1} - \hat{x}_i)^2, \qquad \phi_{\text{tv}}(\hat{x}) = \sum_{i=1}^{n-1} |\hat{x}_{i+1} - \hat{x}_i|$$

Example: signal $x \in \mathbf{R}^{2000}$ and corrupted signal $x_{\rm cor} \in \mathbf{R}^{2000}$

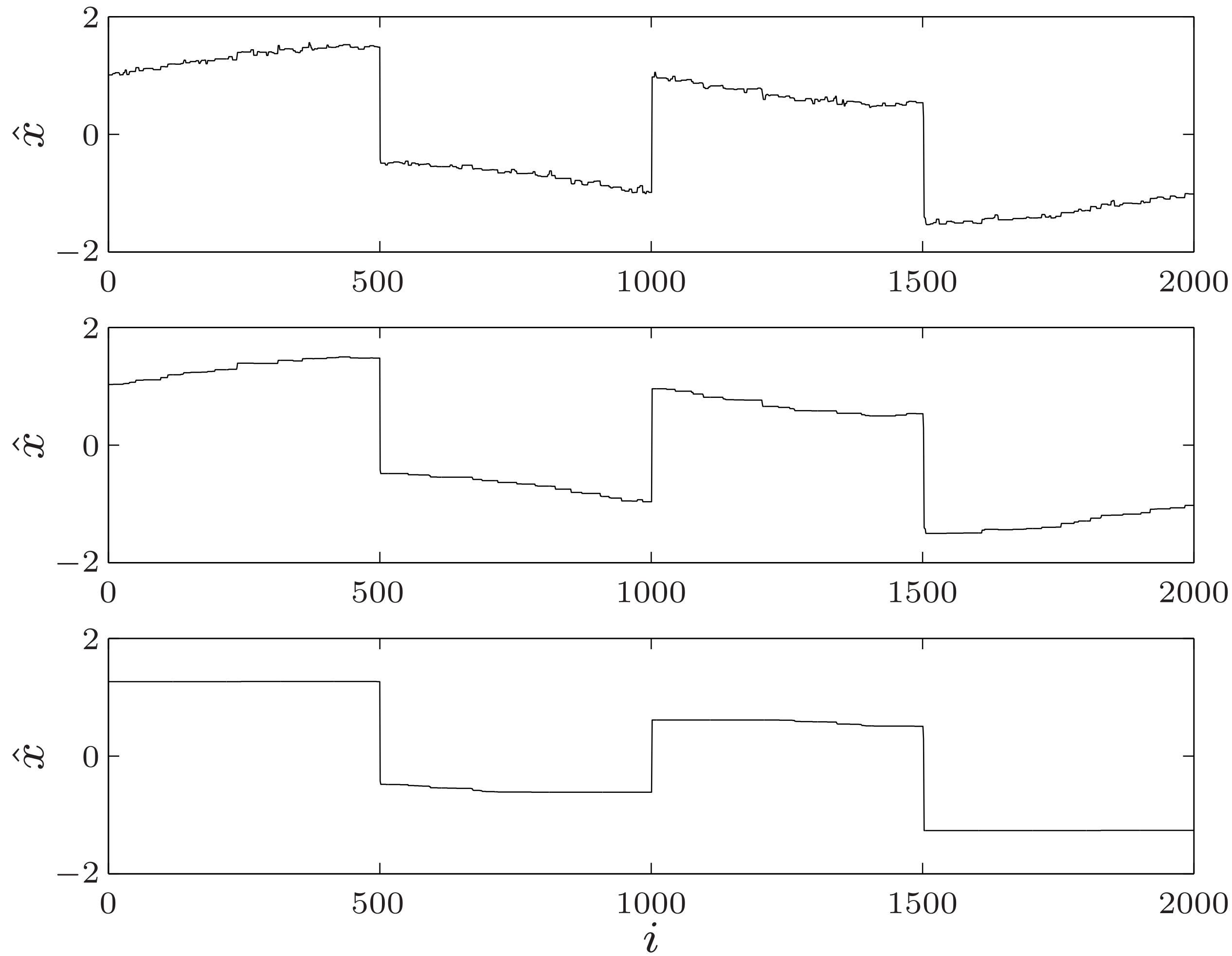


Quadratic smoothing (for three values of γ)



quadratic smoothing smooths out noise **and** sharp transitions in signal

Total variation reconstruction (for three values of γ)



total variation smoothing preserves sharp transitions in signal

Cardinality problems

ℓ_1-norm $\|x\|_1$ as convex approximation of the ℓ_0-'norm' $\mathbf{card}(x)$

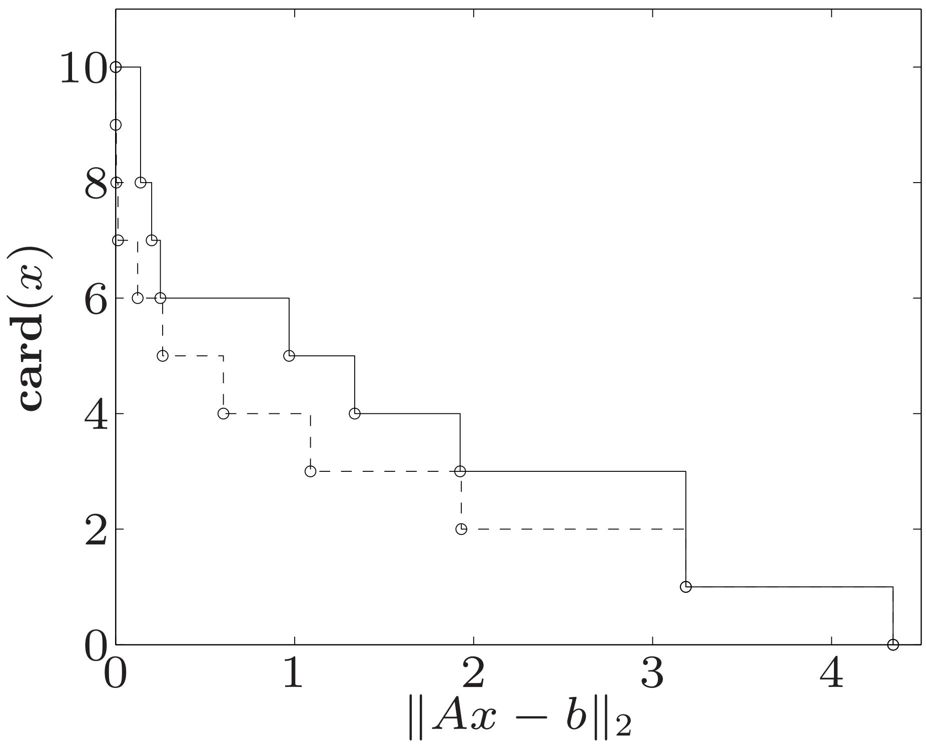
- Sparse regressor selection (LASSO, Tibshirani & Hastie)

$$\text{minimize} \quad \|Ax - b\|_2 + \rho\|x\|_1$$

- Sparse signal representation (basis pursuit, sparse compression) (Donoho, Candès, Tao, Romberg, . . .)

$$\begin{array}{ll} \text{minimize} & \|x\|_1 \\ \text{subject to} & Ax = b \end{array}$$

Sparse regressor selection



(10×20-matrix A)

dashed: optimal trade-off. solid: sparse solution obtained from

$$\text{minimize} \quad \|Ax - b\|_2 + \rho\|x\|_1$$

9. Conclusion

- topics we did not cover
- summary
- references

Topics not covered

Duality

- basis of optimality conditions, sensitivity analysis, provable lower bounds in convex problems
- fundamental for primal-dual algorithms

Algorithms

- interior-point algorithms motivated much of the recent work in convex optimization
- with the right software, suitable for embedded applications
- new (non-interior-point) algorithms for very large scale applications

Summary: Advances in convex optimization

Fundamental theory

new problem classes, robust optimization, convex relaxations, . . .

Applications

new applications in different fields; surprisingly many discovered recently

Algorithms and software

- high-quality general-purpose implementations of interior-point methods
- software packages for convex modeling

Sources for these lectures

- Boyd & Vandenberghe, *Convex Optimization*

  `www.stanford.edu/~boyd/cvxbook`

- Course material for EE364, EE364B (Stanford), EE236B (UCLA)

  `www.stanford.edu/class/ee364`
  `www.stanford.edu/class/ee364b`
  `www.ee.ucla.edu/ee236b`