Vietnam National University - Ho Chi Minh

Optimization, Machine Learning and Kernel Methods.

Optimization II

Marco Cuturi - Princeton University

Outline of this module

- Start with convexity reminders (again...)
- Continue our review of optimization with Duality
- Introduce general convex programs
- Study practical implementations:
  - Gradient descent, Newton Methods
  - Equality constrained Newton Methods
  - Barrier methods.
- Many slides here have been given to me by **Stephen Boyd** (Stanford),
- Check his book (free on the web!) with Lieven Vandenberghe and the excellent videos of his course (youtube) if you want to dig deeper on this topic.

Reminders: Convex set

line segment between x_1 and x_2: all points

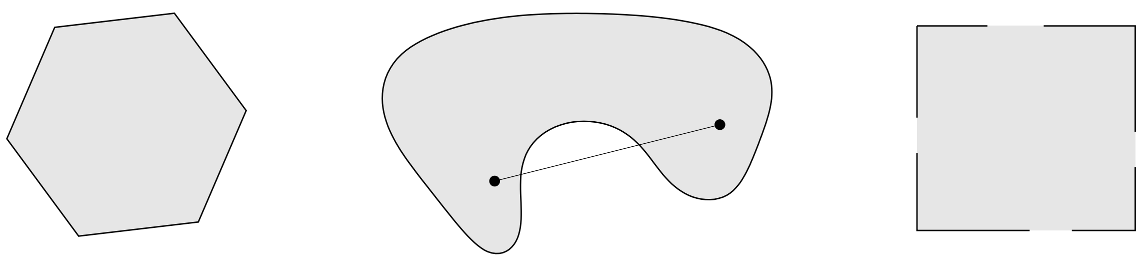
$$x = \lambda x_1 + (1 - \lambda) x_2$$

with $0 \leq \lambda \leq 1$

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

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

examples (one convex, two nonconvex sets)



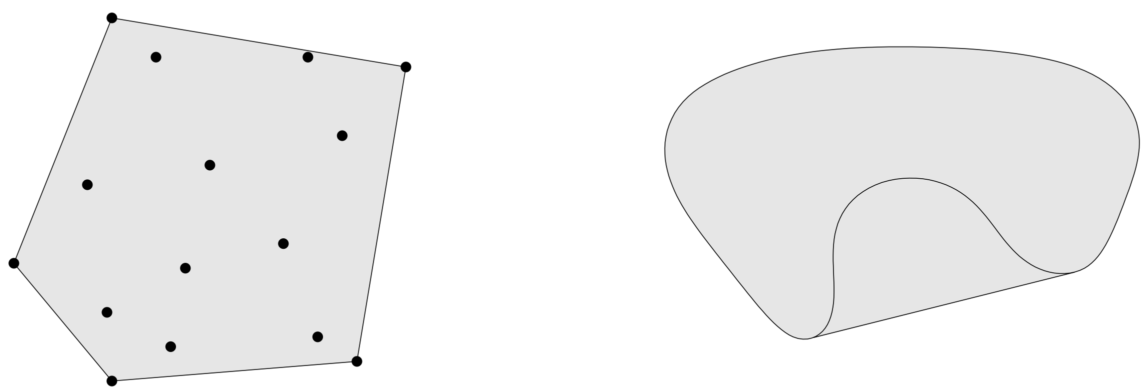
Convex combination and convex hull

convex combination of x_1,. . . , x_k: any point x of the form

$$x = \lambda_1 x_1 + \lambda_2 x_2 + \cdots + \lambda_k x_k$$

with $\lambda_1 + \cdots + \lambda_k = 1$, $\lambda_i \geq 0$

convex hull $\langle S \rangle$: set of all convex combinations of points in S

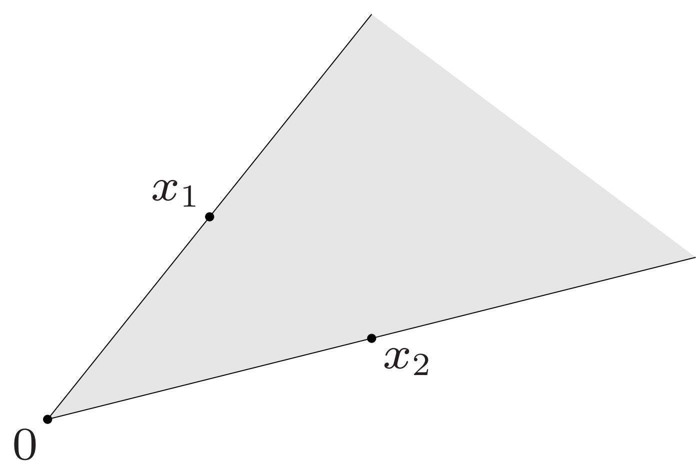


Convex cone

conic (nonnegative) combination of x_1 and x_2: any point of the form

$$x = \lambda_1 x_1 + \lambda_2 x_2$$

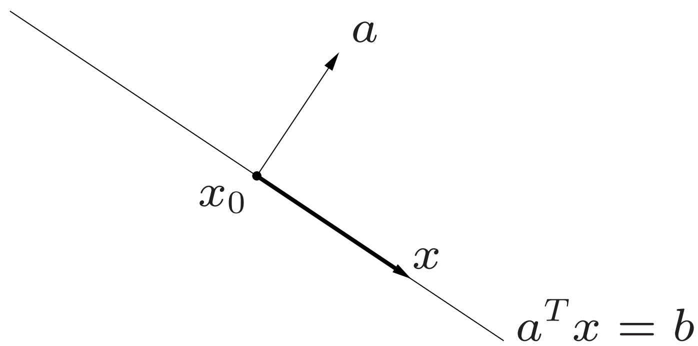
with $\lambda_1 \geq 0$, $\lambda_2 \geq 0$



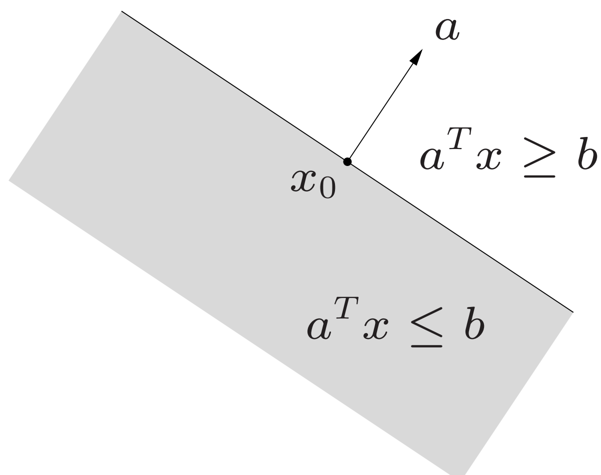
convex cone: set that contains all conic combinations of points in the set

Hyperplanes and halfspaces

hyperplane: set of the form $\{x \mid a^T x = b\}$ $(a \neq 0)$



halfspace: set of the form $\{x \mid a^T x \leq b\}$ $(a \neq 0)$



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

Euclidean balls and ellipsoids

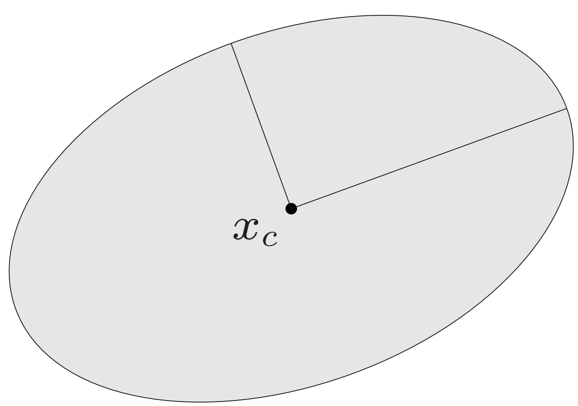
(Euclidean) ball with center x_c and radius r:

$$B(x_c, r) = \{x \mid \|x - x_c\|_2 \leq r\} = \{x_c + ru \mid \|u\|_2 \leq 1\}$$

ellipsoid: set of the form

$$\{x \mid (x - x_c)^T P^{-1}(x - x_c) \leq 1\}$$

with $P \in \mathbf{S}^n_{++}$ (*i.e.*, P symmetric positive definite)



other representation: $\{x_c + Au \mid \|u\|_2 \leq 1\}$ with A square and nonsingular

Norm balls and norm cones

norm: a function $\|\cdot\|$ that satisfies

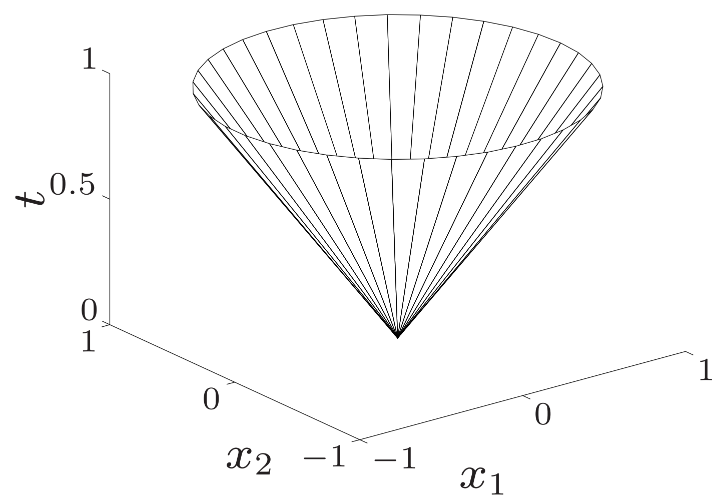
- $\|x\| \geq 0$; $\|x\| = 0$ if and only if $x = 0$
- $\|tx\| = |t| \, \|x\|$ for $t \in \mathbf{R}$
- $\|x + y\| \leq \|x\| + \|y\|$

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



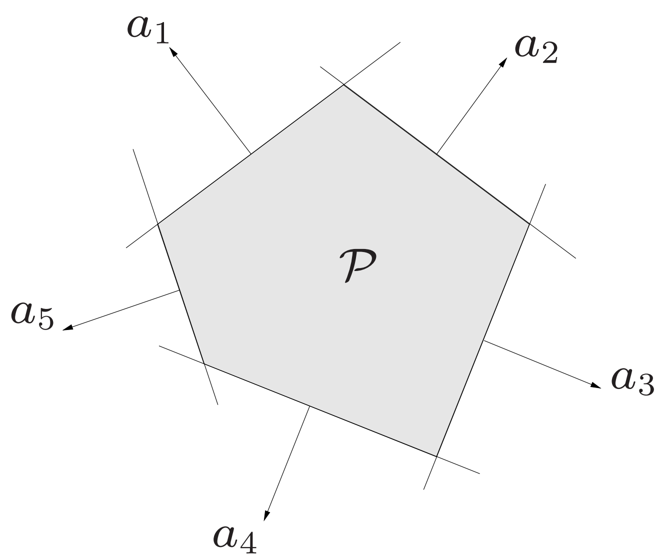
norm balls and cones are convex

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

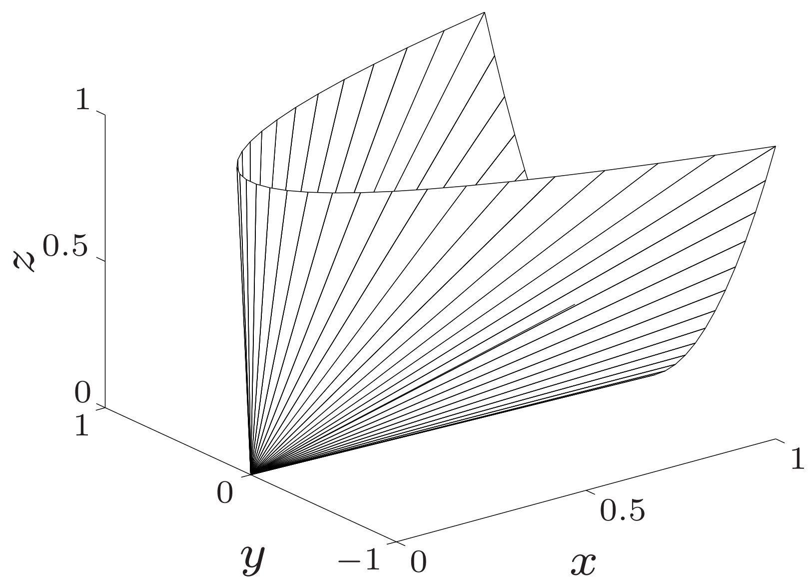
notation:

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



Duality

Duality

- **Duality theory**:
  - Keep this in mind: only a long list of **simple** inequalities. . . .
  - In the end: very powerful results at low technical/numerical cost.
  - A few important, intuitive theorems.

- **In a LP context**:
  - Dual problem provides a different **interpretation** on the same problem.
  - Essentially assigns cost ("displeasure" measure) to constraints.
  - Provides alternative algorithms (dual-simplex).
- **In a more general context**:
  - Very powerful tool to give approximate solutions to intractable problems.

Duality : the general case

Optimization problem

- Consider the following **mathematical program**:

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

  where $\mathbf{x} \in \mathcal{D} \subset \mathbf{R}^n$ with optimal value $p^\star$.

- **No particular assumptions** on $\mathcal{D}$ and the functions f and h (nothing about convexity, linearity, continuity, *etc.*)
- Very generic (includes linear programming and many other problems)

Lagrangian

We form the **Lagrangian** of this problem:

$$L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = f_0(\mathbf{x}) + \sum_{i=1}^{m} \boldsymbol{\lambda_i} f_i(\mathbf{x}) + \sum_{i=1}^{p} \boldsymbol{\mu_i} h_i(\mathbf{x}).$$

Variables $\lambda \in \mathbf{R}^m$ and $\mu \in \mathbf{R}^p$ are called **Lagrange multipliers**.

- The Lagrangian is a **penalized** version of the original objective
- The Lagrange multipliers λ_i, μ_i control the weight of the penalties.
- The Lagrangian is a smoothed version of the hard problem, we have turned $\mathbf{x} \in C$ into penalties that take into account the constraints that **define** C.

Lagrange dual function

- We originally have

$$L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = f_0(\mathbf{x}) + \sum_{i=1}^{m} \boldsymbol{\lambda_i} f_i(\mathbf{x}) + \sum_{i=1}^{p} \boldsymbol{\mu_i} h_i(\mathbf{x})$$

- The penalized problem is here:

$$\begin{aligned} g(\boldsymbol{\lambda}, \boldsymbol{\mu}) &= \inf_{\mathbf{x} \in \mathcal{D}} L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) \\ &= \inf_{\mathbf{x} \in \mathcal{D}} f_0(\mathbf{x}) + \sum_{i=1}^{m} \lambda_i f_i(\mathbf{x}) + \sum_{i=1}^{p} \mu_i h_i(\mathbf{x}) \end{aligned}$$

- The function $g(\lambda, \mu)$ is called the **Lagrange dual function**.
  - Easier to solve than the original one (the constraints are gone)
  - Can often be computed explicitly (more later)

Lower bound

- The function $g(\lambda, \mu)$ produces a lower bound on $p^\star$.
- **Lower bound property**: If $\lambda \geq 0$, then $g(\lambda, \mu) \leq p^\star$
- Why?
  - If $\tilde{\mathbf{x}}$ is feasible,
    - ▷ $f_i(\tilde{\mathbf{x}}) \leq 0$ and thus $\lambda_i f_i(\tilde{x}) \leq 0$
    - ▷ $h_i(\tilde{\mathbf{x}}) = 0$, and thus $\mu_i h_i(\tilde{x}) = 0$
  - thus by construction of L:

$$g(\lambda, \mu) = \inf_{\mathbf{x} \in \mathcal{D}} L(\mathbf{x}, \lambda, \mu) \leq L(\tilde{\mathbf{x}}, \lambda, \mu) \leq f_0(\tilde{\mathbf{x}})$$

  - This is true for any feasible $\tilde{\mathbf{x}}$, so it must be true for the optimal one, which means $g(\lambda, \mu) \leq f_0(\mathbf{x}^\star) = p^\star$.

Lower bound

- We have a **systematic** way of producing **lower bounds** on the optimal value $p^\star$ of the original problem:

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

  by computing the value for a given (λ, μ) couple where $\lambda \geq \mathbf{0}$.

- We can look for the best possible one. . .

Dual problem

- We can define the **Lagrange dual** problem:

$$\begin{array}{ll} \text{maximize} & g(\lambda, \mu) \\ \text{subject to} & \lambda \geq 0 \end{array}$$

in the variables $\lambda \in \mathbf{R}^m$ and $\mu \in \mathbf{R}^p$.

- Finds the best, that is **highest**, possible lower bound $g(\lambda, \mu)$ on the optimal value $p^\star$ of the original (now called **primal**) problem.
- We call its optimal value $d^\star$

Dual problem

- For each given $\mathbf{x}$, the function

$$L(\mathbf{x}, \lambda, \mu) = f_0(\mathbf{x}) + \sum_{i=1}^{m} \lambda_i f_i(\mathbf{x}) + \sum_{i=1}^{p} \mu_i h_i(\mathbf{x})$$

  is **linear** in the variables λ and μ.

- This means that the function

$$g(\lambda, \mu) = \inf_{\mathbf{x} \in \mathcal{D}} L(\mathbf{x}, \lambda, \mu)$$

  is a minimum of linear functions of (λ, μ), so it must be **concave** in (λ, μ)

- This means that the dual problem is always a **concave maximization** problem, whatever f, g, h's properties are.

Weak duality

We have shown the following property called **weak duality**:

$$d^\star \leq p^\star$$

i.e. the optimal value of the dual is always less than the optimal value of the primal problem.

- We haven't made any further assumptions on the problem
- Weak duality must **always hold**
- Produces lower bounds on the problem at low cost

What happens when $d^\star = p^\star$?. . .

Strong duality

When $d^\star = p^\star$ we have **strong duality**.

- Because $d^\star$ is a lower bound on the optimal value $p^\star$, if both are equal for some $(\mathbf{x}, \lambda, \mu)$, the current point must be optimal
- For most convex problems, we have strong duality
- The difference $p^\star - d^\star$ is called the **duality gap** and is a measure of how optimal the current solution $(\mathbf{x}, \lambda, \mu)$.

Slater's conditions

Example of sufficient conditions for **strong duality**:

- **Slater's conditions**. Consider the following problem:

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

  where all the $f_i(\mathbf{x})$ are **convex** and assume that:

$$\text{there exists } \mathbf{x} \in \mathcal{D} : \; f_i(\mathbf{x}) < 0, \; A\mathbf{x} = \mathbf{b}, \quad i = 1, \ldots, m$$

  in other words there is a **strictly feasible point**, then strong duality holds.

- Many other versions exist. . .
- Often easy to check.
- Let's see for linear programs.

Duality: the simple example of linear programming

Duality: linear programming

- Take a **linear program** in standard form:

$$\begin{array}{ll} \text{minimize} & \mathbf{c}^T\mathbf{x} \\ \text{subject to} & A\mathbf{x} = \mathbf{b} \\ & \mathbf{x} \geq 0 \,(\text{ which is equivalent to } -\mathbf{x} \leq 0) \end{array}$$

- We can form the **Lagrangian**:

$$L(\mathbf{x}, \lambda, \mu) = \mathbf{c}^T\mathbf{x} - \lambda^T\mathbf{x} + \mu^T(A\mathbf{x} - \mathbf{b})$$

- and the **Lagrange dual function**:

$$\begin{aligned} g(\lambda, \mu) \quad &= \inf_{\mathbf{x}} L(\mathbf{x}, \lambda, \mu) \\ &= \inf_{\mathbf{x}} \mathbf{c}^T\mathbf{x} - \lambda^T\mathbf{x} + \mu^T(A\mathbf{x} - b) \end{aligned}$$

Duality: linear programming

- For linear programs, the Lagrange dual function can be computed **explicitly**:

$$\begin{aligned} g(\lambda,\mu) &= \inf_{\mathbf{x}} \mathbf{c}^T\mathbf{x} \;\; - \lambda^T\mathbf{x} + \mu^T(A\mathbf{x} - b) \\ &= \inf_{\mathbf{x}} (c - \lambda + A^T\mu)^T\mathbf{x} - \mathbf{b}^T\mu \end{aligned}$$

- This is either $-\mathbf{b}^T\mu$ or $-\infty$, so we finally get:

$$g(\lambda,\mu) = \begin{cases} -\mathbf{b}^T\mu & \text{if } c - \lambda + A^T\mu = 0 \\ -\infty & \text{otherwise} \end{cases}$$

- If $g(\lambda,\mu) = -\infty$ we say that (λ,μ) are outside the domain of the dual.

Duality: linear programming

- With $g(\lambda, \mu)$ given by:

$$g(\lambda, \mu) = \begin{cases} -\mathbf{b}^T\mu & \text{if } c - \lambda + A^T\mu = 0 \\ -\infty & \text{otherwise} \end{cases}$$

- we can write the dual program as:

$$\begin{array}{ll} \text{maximize} & g(\lambda, \mu) \\ \text{subject to} & \lambda \geq 0 \end{array}$$

- which is again, writing the domain explicitly:

$$\begin{array}{ll} \text{maximize} & -\mathbf{b}^T\mu \\ \text{subject to} & c - \lambda + A^T\mu = 0 \\ & \lambda \geq 0 \end{array}$$

Duality: linear programming

- After simplification:

$$\begin{cases} c - \lambda + A^T\mu = 0 \\ \lambda \geq 0 \end{cases} \iff c + A^T\mu \geq 0$$

- we conclude that the dual of the linear program:

$$\begin{array}{ll} \text{minimize} & \mathbf{c}^T\mathbf{x} \\ \text{subject to} & A\mathbf{x} = \mathbf{b} \\ & \mathbf{x} \geq 0 \end{array} \qquad \textbf{(primal)}$$

- is given by:

$$\begin{array}{ll} \text{maximize} & -\mathbf{b}^T\mu \\ \text{subject to} & -A^T\mu \leq c \end{array} \qquad \textbf{(dual)}$$

- equivalently:

$$\begin{array}{ll} \text{maximize} & \mathbf{b}^T\mu \\ \text{subject to} & A^T\mu \leq c \end{array}$$

Dual Linear Program

Up to now, what have we introduced?

- A vector of parameters $\mu \in \mathbf{R}^m$, **one coordinate by constraint**.
- For **any** μ and any feasible $\mathbf{x}$ of the primal = a lower bound on the primal.
- For **some** μ the lower bound is $-\infty$, not useful.
- The **dual problem** computes the **biggest** lower bound.
- We discard values of μ which give $-\infty$ lower bounds.
- This the way **dual constraints** are defined.
- The **dual** is **another linear program** in dimensions $\mathbf{R}^{n\times m}$, that is
  - n constraints,
  - m variables.

From Primal to Dual for general LP's

- Some notations: for $A \in \mathbf{R}^{m \times n}$ we write
  - $\mathbf{a}_j$ for the n column vectors
  - $\mathbf{A}_i$ for the m row vectors of A.
- Following a similar reasoning we can flip from primal to dual changing
  - the constraints linear relationships A,
  - the constraints constants $\mathbf{b}$,
  - the constraints directions $(\leq, \geq, =)$
  - non-negativity conditions,
  - the objective

minimize	$\mathbf{c}^T\mathbf{x}$		maximize	$\mu^T\mathbf{b}$	
subject to	$\mathbf{A}_i^T\mathbf{x} \geq b_i,$	$i \in M_1$	subject to	$\mu_i \geq 0$	$i \in M_1$
	$\mathbf{A}_i^T\mathbf{x} \leq b_i,$	$i \in M_2$		$\mu_i \leq 0$	$i \in M_2$
	$\mathbf{A}_i^T\mathbf{x} = b_i,$	$i \in M_3$		μ_i free	$i \in M_3$
	$x_j \geq 0$	$j \in N_1$		$\mu^T\mathbf{a}_j \leq c_j$	$j \in N_1$
	$x_j \leq 0$	$j \in N_1$		$\mu^T\mathbf{a}_j \geq c_j$	$j \in N_2$
	x_j free	$j \in N_1$		$\mu^T\mathbf{a}_j = c_j$	$j \in N_3$

(1)

Dual Linear Program

- In summary, for any kind of constraint,

primal	minimize	maximize	**dual**
constraints	$\geq b_i$ $\leq b_i$ $= b_i$	≥ 0 ≤ 0 free	variables
variables	≥ 0 ≤ 0 free	$\leq c_j$ $\geq c_j$ $= c_j$	constraints

- For simple cases and in matrix form,

minimize $\mathbf{c}^T\mathbf{x}$ subject to $A\mathbf{x} = \mathbf{b}$ $\mathbf{x} \geq 0$	$\Rightarrow$	maximize $\mathbf{b}^T\mu$ subject to $A^T\mu \leq c$
minimize $\mathbf{c}^T\mathbf{x}$ subject to $A\mathbf{x} \geq \mathbf{b}$	$\Rightarrow$	maximize $\mathbf{b}^T\mu$ subject to $A^T\mu = c$ $\mu \geq 0$

Dual Linear Program: Equivalence Theorems

Theorem 1. *If we transform the dual problem into an equivalent minimization problem and the form its dual, we obtain a problem that is equivalent to the original problem*

- The **dual of the dual** of a given primal LP **is the primal LP** itself.
- Linear programs are **self-dual**.
- Not true in the general case: dual of the dual is called the **bi-dual**.
- The tables before can be used in both directions indifferently.

Dual Linear Program: Equivalence Theorems

Theorem 2. *If we transform a LP (1) into another LP (2) through any of the following operations:*

- *replace free variables with the difference of two nonnegative variables;*
- *replace inequality constraints by an equality constraint with a surplus/slack variable;*
- *remove redundant (colinear) rows of the constraint matrix for standard forms;*

then the duals of (1) and (2) are equivalent, i.e. they are either both infeasible or have the same optimal objective.

Duality for LP's : Weak Duality

We proved weak duality for general programs. Although LP's are a **particular case** the arguments are here explicit:

Theorem 3. *If* $\mathbf{x}$ *is a feasible solution to a primal LP and* μ *is a feasible solution to the dual problem then*

$$\mu^T \mathbf{b} \leq \mathbf{c}^T \mathbf{x}$$

- **Proof idea** check what is called the complementary slackness variables $\mu_i(\mathbf{A}_i^T \mathbf{x} - b_i)$ and $(c_j - \mu^T \mathbf{a}_j)\mathbf{x}_j$ and use the primal/dual relationships.

Weak Duality Proof

Proof. • Let $\mathbf{x} \in \mathbf{R}^n$ and $\mu \in \mathbf{R}^m$ and define

$$\begin{aligned} u_i &= \mu_i(\mathbf{A}_i^T\mathbf{x} - b_i) \quad & i = 1, .., m \\ v_j &= (c_j - \mu^T\mathbf{a}_j)\mathbf{x}_j \quad & j = 1, .., n \end{aligned}$$

- Suppose $\mathbf{x}$ and μ are primal and dual feasible for an LP involving A, $\mathbf{b}$ and $\mathbf{c}$.
- Check Equations 1. Whatever the constraints are,
  - μ_i and $(\mathbf{A}_i^T\mathbf{x} - b_i)$ have the same sign or their product is zero.
  - The same goes for $(c_j - \mu^T\mathbf{a}_j)$ and $\mathbf{x}_j$.
- Hence $u_i, v_j \geq 0$.
- Furthermore $\sum_i^m u_i = \mu^T(A\mathbf{x} - \mathbf{b})$ and $\sum_j^n v_j = (\mathbf{c}^T - \mu^T A)\mathbf{x}$
- Hence $0 \leq \sum_i^m u_i + \sum_j^n v_j = \mathbf{c}^T\mathbf{x} - \mu^T\mathbf{b}$

■

Weak Duality

- Not a very strong result at first look.
- Specially since we already discussed **strong duality**...

- Yet weak duality provides us with the two simple yet **important corollaries**.
- In the following we assume that the **primal** is a **minimization**.
- As usual, results can be easily proved the other way round.

Weak Duality Corollary 1

Corollary 1. • *If the objective in the primal can be arbitrarily small then the dual problem must be infeasible.*

• *If the objective in the primal can be arbitrarily big then the dual problem must be infeasible.*

Proof. • By weak duality, $\mu^T \mathbf{b} \leq \mathbf{c}^T \mathbf{x}$ for any two feasible points $\mathbf{x}, \mu$.

• If the objective for feasible $\mathbf{x}$ can be set arbitrarily low, then a feasible μ cannot exist.

• The same applies for a feasible $\mathbf{x}$ if the dual objective can be arbitrarily high.

■

Weak Duality Corollary 2

Corollary 2. *Let* $\mathbf{x}^\star$ *and* $\mu^\star$ *be two feasible solutions to the primal and dual respectively. Suppose that* $\mu^{\star T}\mathbf{b} = \mathbf{c}^T\mathbf{x}^\star$*. Then* $\mathbf{x}^\star$ *and* $\mu^\star$ *are optimal solutions for the primal and dual respectively.*

Proof. For every feasible point of the primal $\mathbf{y}$, $\mathbf{c}^T\mathbf{x}^\star = \mu^{\star T}\mathbf{b} \leq \mathbf{c}^T\mathbf{y}$ hence $\mathbf{x}^\star$ is optimal. Same thing for $\mu^\star$. ■

- Let's check whether strong duality holds or not for linear programs...

Strong Duality

- For linear programs, **strong duality is always ensured**.
- We use the **simplex**'s convergence to the optimal solution in this proof.
- We will cover a more geometric approach in the next lecture.

Theorem 4. *if an LP has an optima, so does its dual, and their* ***respective optimal objectives are equal****.*

- **Proof strategy**:
  - prove it first for a **standard form LP**, showing that the **reduced cost coefficient** can be used to define a **dual feasible solution**..
  - For a general LP, use Theorem 2

Strong Duality: Proof 1

Proof. • Consider the standard form

$$\begin{array}{ll} \text{minimize} & \mathbf{c}^T\mathbf{x} \\ \text{subject to} & A\mathbf{x} = \mathbf{b} \\ & \mathbf{x} \geq 0 \end{array}$$

- Let's use the simplex with the lexicographic rule for instance. Let $\mathbf{x}$ be the optimal solution with basis $\mathbf{I}$ and objective z.
- The reduced costs must be nonnegative (here we have a **min** problem) hence

$$\mathbf{c}^T - \mathbf{c}_{\mathbf{I}}^T B_{\mathbf{I}}^{-1} A \geq \mathbf{0}^T$$

- Let $\mu^T = \mathbf{c}_{\mathbf{I}}^T B_{\mathbf{I}}^{-1}$. Then $\mu^T A \leq \mathbf{c}^T$ coordinate wise.
- μ is a **feasible** solution to the dual problem.
- Furthermore $\mu^T\mathbf{b} = \mathbf{c}_{\mathbf{I}}^T B_{\mathbf{I}}^{-1}\mathbf{b} = \mathbf{c}_{\mathbf{I}}^T \mathbf{x}_{\mathbf{I}} = z$.
- μ is thus optimal w.r.t to the dual following the previous corollary.

Strong Duality: Proof 2

- Suppose now that we have a general LP (1).
- Through operations as described in Theorem 2 the program is changed into an equivalent standard program (2). They share the same optimal cost.
- The dual of program (D2) has the same optimal cost in turn.
- Both (D2) and (D1) have the same optimal cost by Theorem 2.
- Hence (1) and (D1) have the same optimal cost.

■

Complementary slackness

- Another important result that links both optima:

Theorem 5. *Let* $\mathbf{x}$ *and* μ *be feasible solutions to the primal and dual problems respectively. The vectors for* $\mathbf{x}$ *and* μ *are optimal solutions for the two respective problems if and only if*

$$\begin{aligned} u_i &= \mu_i(\mathbf{A}_i^T\mathbf{x} - b_i) &= \mathbf{0}, \quad i = 1, .., m; \\ v_j &= (c_j - \mu^T\mathbf{a}_j)\mathbf{x}_j &= \mathbf{0}, \quad j = 1, .., n. \end{aligned}$$

Proof. In the proof of the weak duality we showed that $u_i, v_j \geq 0$. Moreover

$$0 \leq \sum_i^m u_i + \sum_j^n v_j = \mathbf{c}^T\mathbf{x} - \mu^T\mathbf{b}.$$

Hence, $\mathbf{x}, \mu$ optimal $\Leftrightarrow u_i = v_j = 0$ through strong duality ($\Rightarrow$) and the second corollary of weak duality ($\Leftarrow$). ■

Examples for LP's

Duality

- A simple example with the following linear program:

$$\begin{array}{ll} \text{minimize} & 3x_1 + x_2 \\ \text{subject to} & x_2 - 2x_1 = 1 \\ & x_1, x_2 \geq 0 \end{array}$$

- Two inequality constraints, one equality constraint. The Lagrangian is written:

$$L(x, \lambda, \mu) = 3x_1 + x_2 - \lambda_1 x_1 - \lambda_2 x_2 + \mu(1 - x_2 + 2x_1)$$

in the (dual variables) $\lambda_1, \lambda_2 \geq 0$ and μ (free).

Duality

$$\begin{aligned} g(\lambda, \mu) &= \inf_{\mathbf{x}} L(\mathbf{x}, \lambda, \mu) \\ &= \inf_{\mathbf{x}} 3x_1 + x_2 - \lambda_1 x_1 - \lambda_2 x_2 + \mu(1 - x_2 + 2x_1) \\ &= \inf_{\mathbf{x}} (3 - \lambda_1 + 2\mu)x_1 + (1 - \lambda_2 - \mu)x_2 + \mu \end{aligned}$$

- We minimize a linear function of $x_1, \; x_2$, only two possibilities:

$$g(\lambda, \mu) = \begin{cases} \mu & \text{if } 3 - \lambda_1 + 2\mu = 1 - \lambda_2 - \mu = 0 \\ -\infty & \text{otherwise} \end{cases}$$

- The dual problem is finally:

$$\begin{array}{ll} \text{maximize} & \mu \\ \text{subject to} & 3 - \lambda_1 + 2\mu = 0 \\ & 1 - \lambda_2 - \mu = 0, \lambda \geq 0 \end{array}$$

LP's, Duality and Arbitrage

Duality and Arbitrage

- We propose in this an economic interpretation of duality
- Due to Arrow, Debreu, in the 50's. . .
- Used **every day** on financial markets (sometimes unknowingly)
- Simple LP duality result, but underpins most of modern finance theory. . .

One period model

- As in the previous section, basic discrete, **one period** model on a single asset.
- Its price **today** is q_1. Its (random) price **time T ahead** is x.
- Assume x can only take any of the following values

$$x \in \{x_1, \ldots, x_n\}$$

at a **maturity date** T, and that we have an estimate of their probabilities,

$$\{p_1, \cdots, p_n\}.$$

- We have **discretized** the space of possibilities.
- We can only trade **today** and at **maturity**
- There is a **cash** security worth $\$1$ today, that pays $\$1$ at maturity
- near-zero interest rates. sounds familiar?

One period model

- There are also $m-1$ other securities with payoffs at maturity given by

$$h_k(x_i) \quad \text{if } x = x_i \text{ at time } T$$

  for $k = 2, \ldots, m-1$.

- The payoffs are **arbitrary** functions of the n possible values of the asset at time T.
- We could have $h_k(x) = x^2$. Or that for $i \le j$, $h_k(x_i) = 0$, $i > j$, $h_k(x_i) = 1$.
- We denote by q_k the price **today** of security k with payoff $h_k(x)$.

All these securities are tradeable, can we use them to get information on the price of **another security** with payoff $h_0(x)$?

Static Arbitrage

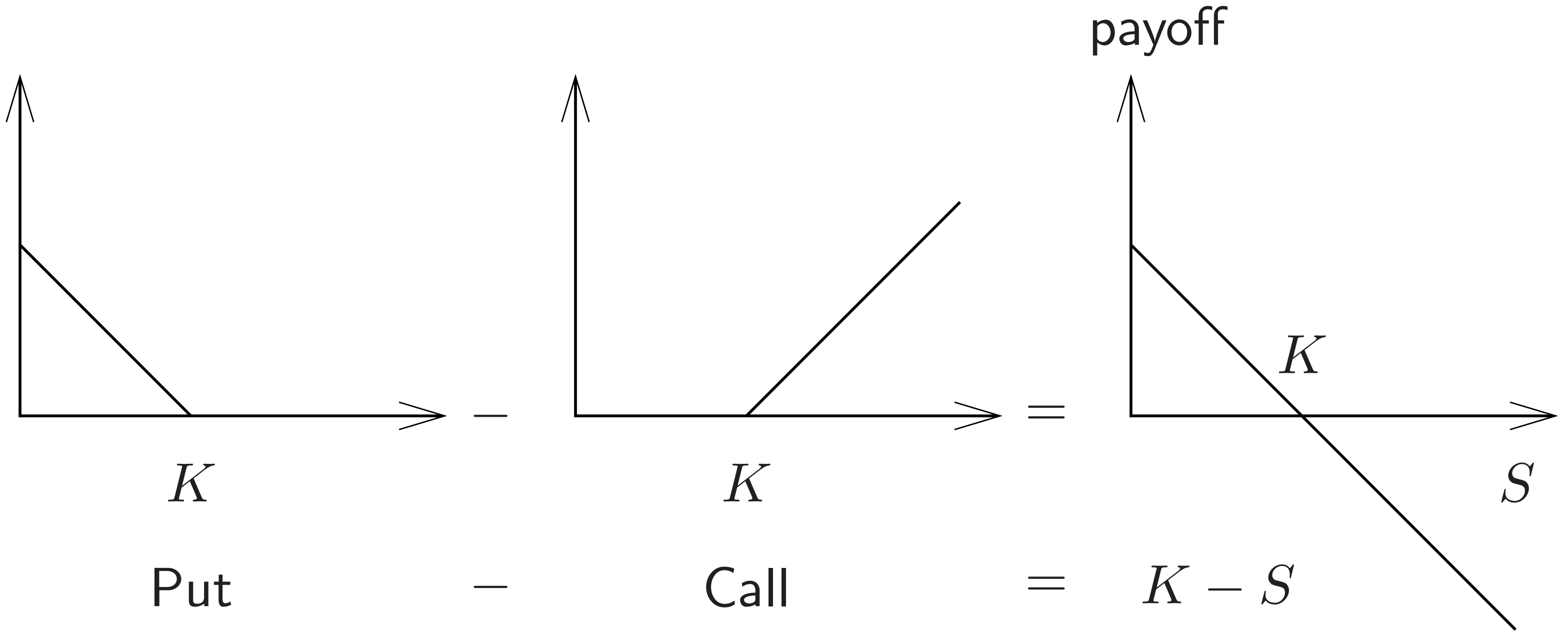
Remember:

- We can only trade today and at maturity.
- We can only trade in securities which are priced by the market.

We want to exclude **arbitrage strategies**

- If the payoff of a portfolio A is always larger than that of a portfolio B then Price(A) $\geq$ Price(B).
- The price of the sum of two products is equal to the sum of the prices.

Simplest Example: Put Call Parity



Put $-$ Call $= K - S$

Price bounds

Suppose that we form a portfolio of cash, stocks and securities $h_k(x)$ with coefficients λ_k:

λ_0	in cash
λ_1	in stock
λ_k	in security $h_k(x)$

- All portfolios that satisfy

$$\lambda_0 + \lambda_1 x_i + \sum_{k=2}^{m} \lambda_k h_k(x_i) \geq h_0(x_i) \quad \text{i=1,. . . ,n}$$

  must be **more expensive** than the security $h_0(x)$

- All portfolios that satisfy the **opposite** inequality must be **cheaper**
- For portfolios that satisfy neither of these, **nothing** can be said. . .
- We are just comparing portfolios dominated for **all** outcomes of x.

Price bounds

- For each of these portfolios, we get an upper/lower bound on the price today of the security $h_0(x)$.
- We can look for optimal bounds. . .
- We can solve:

$$\begin{array}{ll} \text{minimize} & \lambda_0 + \lambda_1 q_1 + \sum_{k=1}^{m} \lambda_k q_k \\ \text{subject to} & \lambda_0 + \lambda_1 x_i + \sum_{k=2}^{m} \lambda_k h_k(x_i) \geq h_0(x_i), \quad i = 1, \ldots, n \end{array}$$

  - Linear program in the variable $\lambda \in \mathbf{R}^{(m+1)}$
  - Produces an optimal upper bound on the price today of the security $h_0(x)$

Linear Programming Duality

- The original linear program looks like:

$$\begin{array}{ll} \text{minimize} & c^T\lambda \\ \text{subject to} & A\lambda \geq b \end{array}$$

which is a linear program in the variable $\lambda \in \mathbf{R}^m$.

- We can form the Lagrangian

$$L(\lambda, p) = c^T\lambda + y^T(b - A\lambda)$$

in the variables $\lambda \in \mathbf{R}^m$ and $y \in \mathbf{R}^n$, with $y \succeq 0$.

Linear Programming Duality

- We then minimize in λ to get the dual function

$$g(y) = \inf_{\lambda} c^T \lambda + y^T (b - A\lambda)$$

for $y \succeq 0$, which is again

$$g(y) = \inf_{\lambda} y^T b + \lambda^T (c - A^T y)$$

and we get:

$$g(y) = \begin{cases} y^T b & \text{if } c - A^T y = 0 \\ -\infty & \text{if not.} \end{cases}$$

Linear Programming Duality

- With

$$g(y) = \begin{cases} y^T b & \text{if } c - A^T y = 0 \\ -\infty & \text{if not.} \end{cases}$$

- we get the **dual linear program** as:

$$\begin{array}{ll} \text{maximize} & b^T y \\ \text{subject to} & A^T y = c \\ & y \geq 0 \end{array}$$

which is also a linear program in $x \in \mathbf{R}^n$.

LP duality: summary

- The primal LP is the original linear program looks like:

$$\begin{array}{ll} \text{minimize} & c^T\lambda \\ \text{subject to} & A\lambda \geq b \end{array}$$

- its **dual** is then given by:

$$\begin{array}{ll} \text{maximize} & b^T y \\ \text{subject to} & A^T y = c \\ & y \geq 0 \end{array}$$

Strong duality: both optimal values are **equal**

LP duality & arbitrage

- Let's look at what this produces for the portfolio problem. . .
  - The **primal** problem in the variable $\lambda \in \mathbf{R}^m$ is given by:

$$\begin{array}{rll} p^{\max} := & \text{min.} & \lambda_0 + \lambda_1 q_1 + \sum_{k=2}^m \lambda_k q_k \\ & \text{s.t.} & \lambda_0 + \lambda_1 x_i + \sum_{k=2}^m \lambda_k h_k(x_i) \geq h_0(x_i), \quad i = 1, \ldots, n \end{array}$$

  - The **dual** in the variable $y \in \mathbf{R}^n$ is then

$$\begin{array}{rll} p^{\max} := & \text{max.} & \sum_{i=1}^n y_i h_0(x_i) \\ & \text{s.t.} & \sum_{i=1}^n y_i h_k(x_i) = q_k, \quad k = 2, \ldots, m \\ & & \sum_{i=1}^n y_i x_i = q_1 \\ & & \sum_{i=1}^n y_i = 1 \\ & & y \geq 0 \end{array}$$

LP duality & arbitrage

- The last two constraints $\{\sum_{i=1}^{n} y_i = 1,\ y \geq 0\}$ mean that y is a **probability measure**.
- We can rewrite the previous program as:

$$\begin{array}{lll} p^{\max} := & \text{max.} & \mathbf{E}_y[h_0(x)] \\ \\ & \text{s.t.} & \mathbf{E}_y[h_k(x)] = q_k, \quad k = 2, \ldots, m \\ & & \mathbf{E}_y[x] = q_1 \\ & & y \text{ is a probability} \end{array}$$

- We can compute $p^{\min}$ by minimizing instead.

LP duality & arbitrage

- What does this mean?
- There are three ranges of prices for the security with payoff $h_0(x)$:
  - Prices above $p^{\max}$: these are **not viable**, you can get a cheaper portfolio with a payoff that always dominates $h_0(x)$.
  - Prices in $[p^{\min}, p^{\max}]$: prices are **viable**, $i.e.$ compatible with the absence of arbitrage.
  - Prices below $p^{\min}$: these are **not viable**, you can get a portfolio that is more expensive than $h_0(x)$ with a payoff that is always dominated by $h_0(x)$.

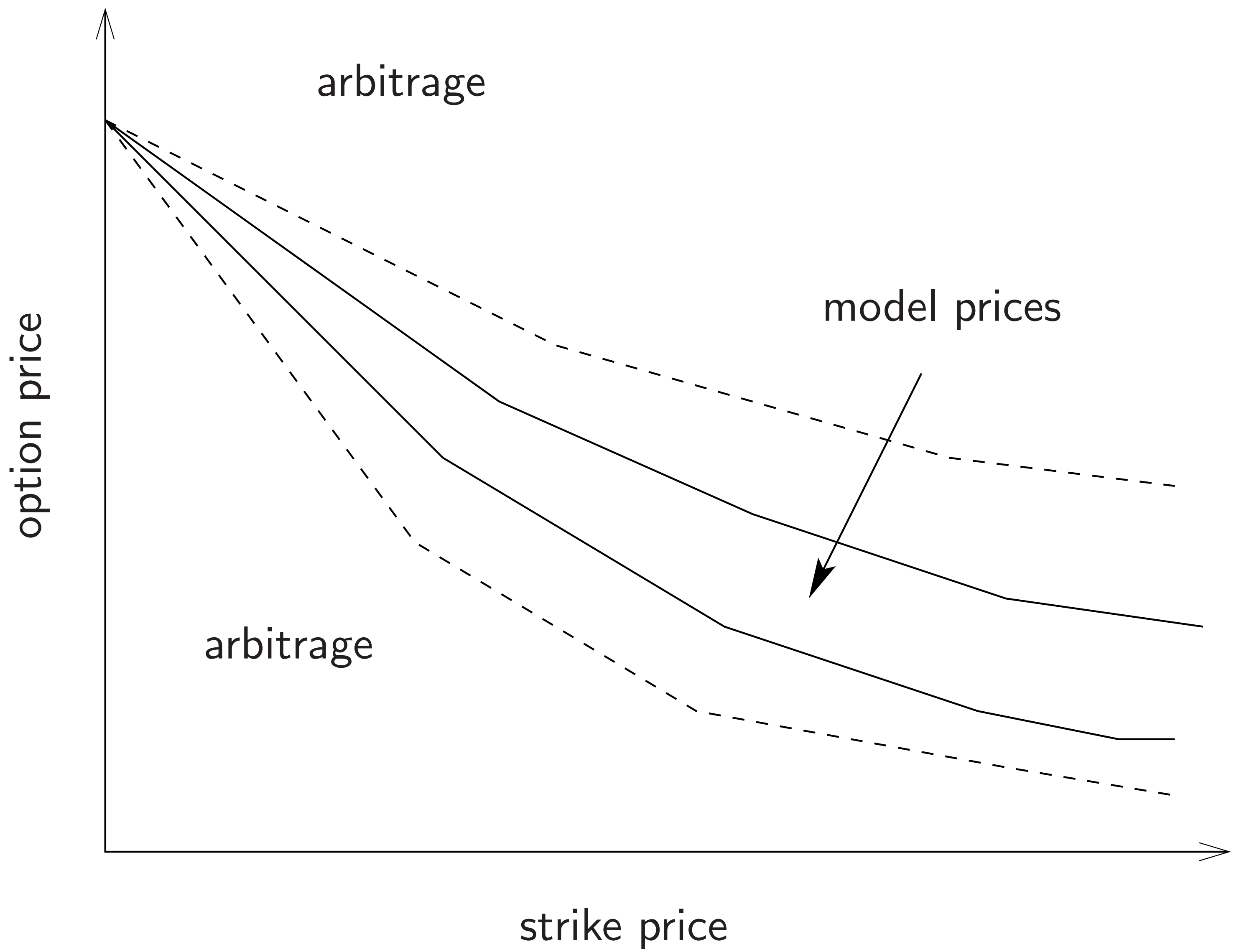
Price bounds

- Example:
  - Suppose the product in the objective is a call option:

$$h_0(x) = (x - K)^+$$

    where K is called the strike price.
  - Suppose also that we know the prices of some other instruments
  - We get upper and lower price bounds on the price of this call for each strike K
- On a graphic. . .

Price Bounds



LP duality & arbitrage

- What if there is no solution y and the linear program is infeasible?
  - Then the original data set q must contain an arbitrage.
  - Start with one product, stock and cash. . . and test.
  - Increase the number of products. . .

LP duality & arbitrage

Fundamental theorem of asset pricing

Theorem 6. *In the one period model, there is no arbitrage between the prices* $\{q_0, \ldots, q_m\}$ *of securities with payoffs at maturity* $\{h_0(x), \ldots, h_m(x)\}$

$$\Updownarrow$$

There exists a probability y *(with* $\sum_{i=1}^{n} y_i = 1$ *and* $y \geq 0$*) such that*

$$q_k = \mathbf{E}_y[h_k(x)], \quad k = 0, \ldots, m$$

LP duality & arbitrage

- Because prices are computed using **expectations under** y (and not expected utility/certain equivalent), we call the probability y **risk-neutral**.
- In particular, it satisfies $q_1 = \mathbf{E}_y[x]$
- If there are *constant* interest rates, simply use **discounted** values for **prices at maturity**. . .
- This probability y has **nothing to do** with the observed distribution of the asset x or its past distribution! (Very common mistake)

LP duality & arbitrage

- Because one can trade
  - the asset
  - derivative products based on the asset

  to form portfolios to hedge/replicate other products, it is possible to evaluate these products using expected value under an **appropriate choice** of probability.

- Again, the risk-neutral probability y is a **tool** **inferred from market prices**,

- it has nothing to do with the statistical properties of the underlying asset x.

- Linear programming duality is interpreted as a duality between **portfolios on assets** problems and **probabilities** (models)

LP duality & arbitrage

In the previous result:

- Set of possible **probabilistic models** = **probability simplex**:
  $p_i \geq 0, \;\; \sum_i p_i = 1$
- Expected value, hence price is linear in the probability p_i

$$\mathbf{E}[h(x)] = \sum_i p_i h(x_i)$$

- A price constraint is just a linear equality constraint on the probabilities:

$$\sum_i p_i h(x_i) = b_i$$

- Simple family of distributions.

Moment constraints

Choices for asset pricing formulas that depend on the prices directly:. . .

- Use indicator function as payoff:

$$h(x) = 1_{\{x \geq K\}}$$

to produce the constraint:

$$\sum_i p_i \; 1_{\{x_i \geq K\}} = P(X \geq K) = b$$

- Also, quadratic variation:

$$h(x) = x^2$$

Corresponds to:

$$\sum_i p_i \; x_i^2 = \mathbf{E}[x_i^2] = b$$

Moment constraints

Higher order formulations? Variance?

- We can't incorporate a variance swap
- A constraint of the form

$$\textbf{Variance}(x) = q_V$$

  why?
- Becomes $\sum_i p_i x_i^2 - (\sum_i p_i x_i)^2 = q_V \Rightarrow$ quadratic constraints in p_i.
- Would however works if we also fix the expected value:

$$\mathbf{E}[x] = b$$

  Corresponds to a **forward** price (EV of the asset):

$$\sum_i p_i\ x_i = q_F \quad \text{and} \quad \textbf{Variance}(x) = \sum_i p_i\ x_i^2 - q_F^2 = q_V$$

- We came back to a simple **linear constraint**

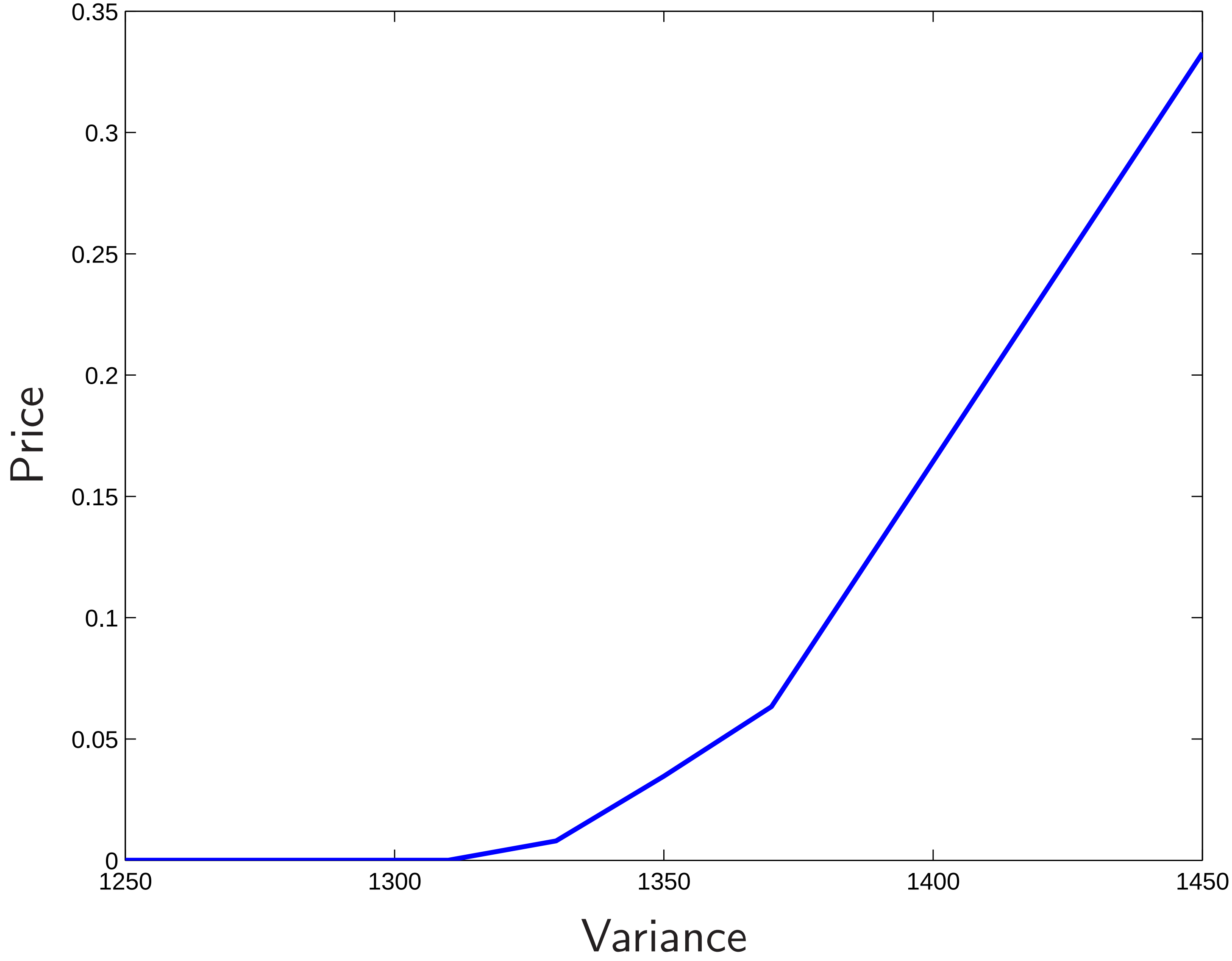
Option price vs. variance

- Fix the forward price (expected value of the asset), **move the variance**. . .
- We study the price of a **call option** h_0.

$$
\begin{array}{ll}
\text{maximize} & \sum_i p_i \, h_0(x_i) \\
\text{subject to} & \sum_i p_i \, x_i = S_0 \\
& \sum_i p_i \, x_i^2 = b^2 \\
& 0 \leq p_i \leq 1,
\end{array}
$$

- Look at the price as a function of b^2. . .

Option price vs. variance



Option pricing & LP: example

Option pricing

Option pricing example. . .

- Study the price **CutCall** option, with payoff:

$$h_0(X) = (X - K)^+ 1_{\{X \leq L\}}$$

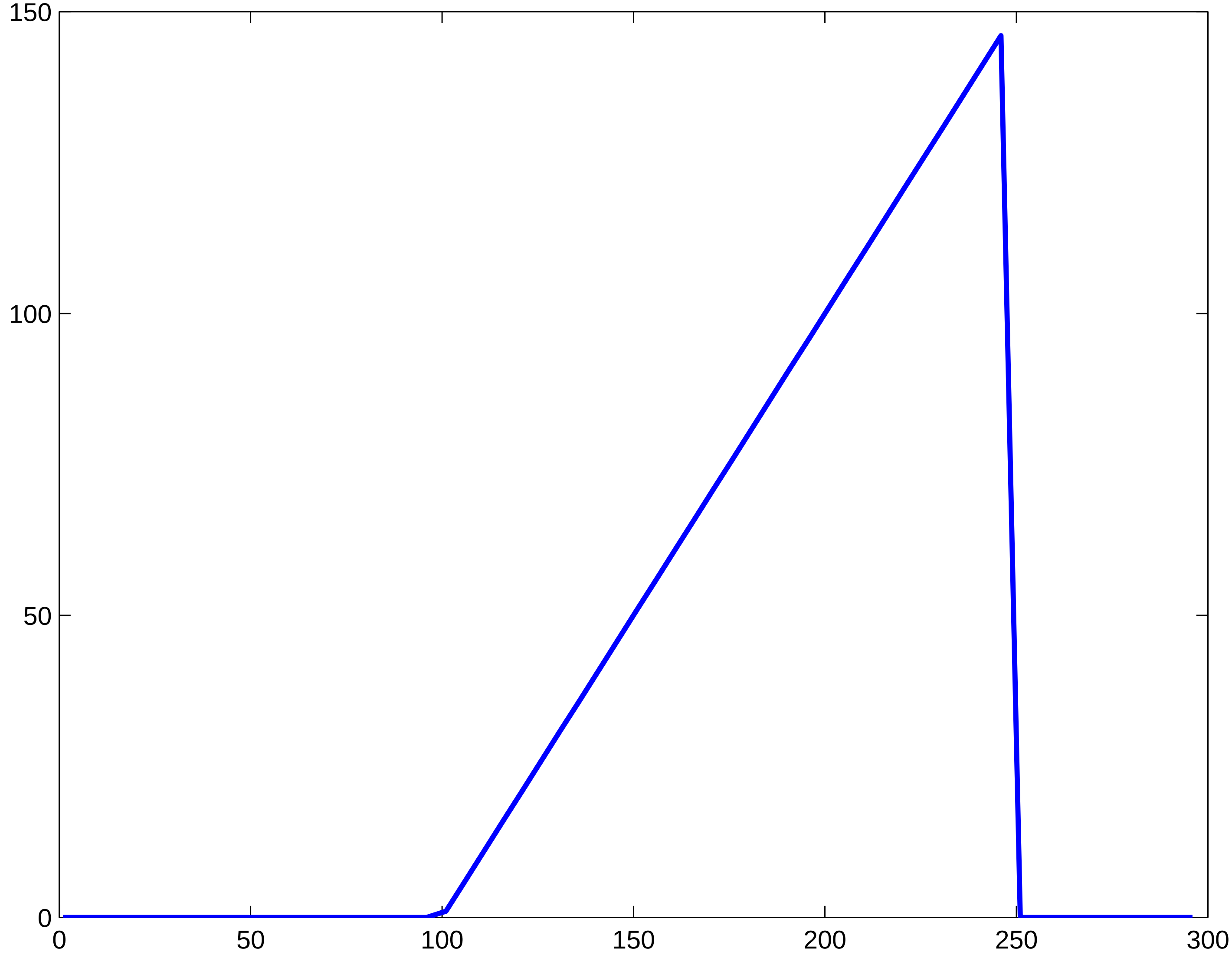
- Similar to knock-out option but only **check at maturity**. **No knock-out** during its life, **european** kind of knock-out.
- Get some market prices q_k for **regular** calls:

$$h_k(X) = (X - K_k)^+$$

- Solve for the maximum CutCall price:

$$\begin{array}{ll} \text{maximize} & \sum_i p_i h_0(x_i) \\ \text{subject to} & \sum_i p_i h_k(x_i) = q_k \\ & \sum_i p_i = 1 \\ & p_i \geq 0 \end{array}$$

Payoff



Option pricing

Solve

$$\begin{array}{ll} \text{maximize} & \sum_i p_i h_0(x_i) \\ \text{subject to} & \sum_i p_i h_k(x_i) = q_k \\ & \sum_i p_i = 1 \\ & p_i \geq 0 \end{array}$$

with

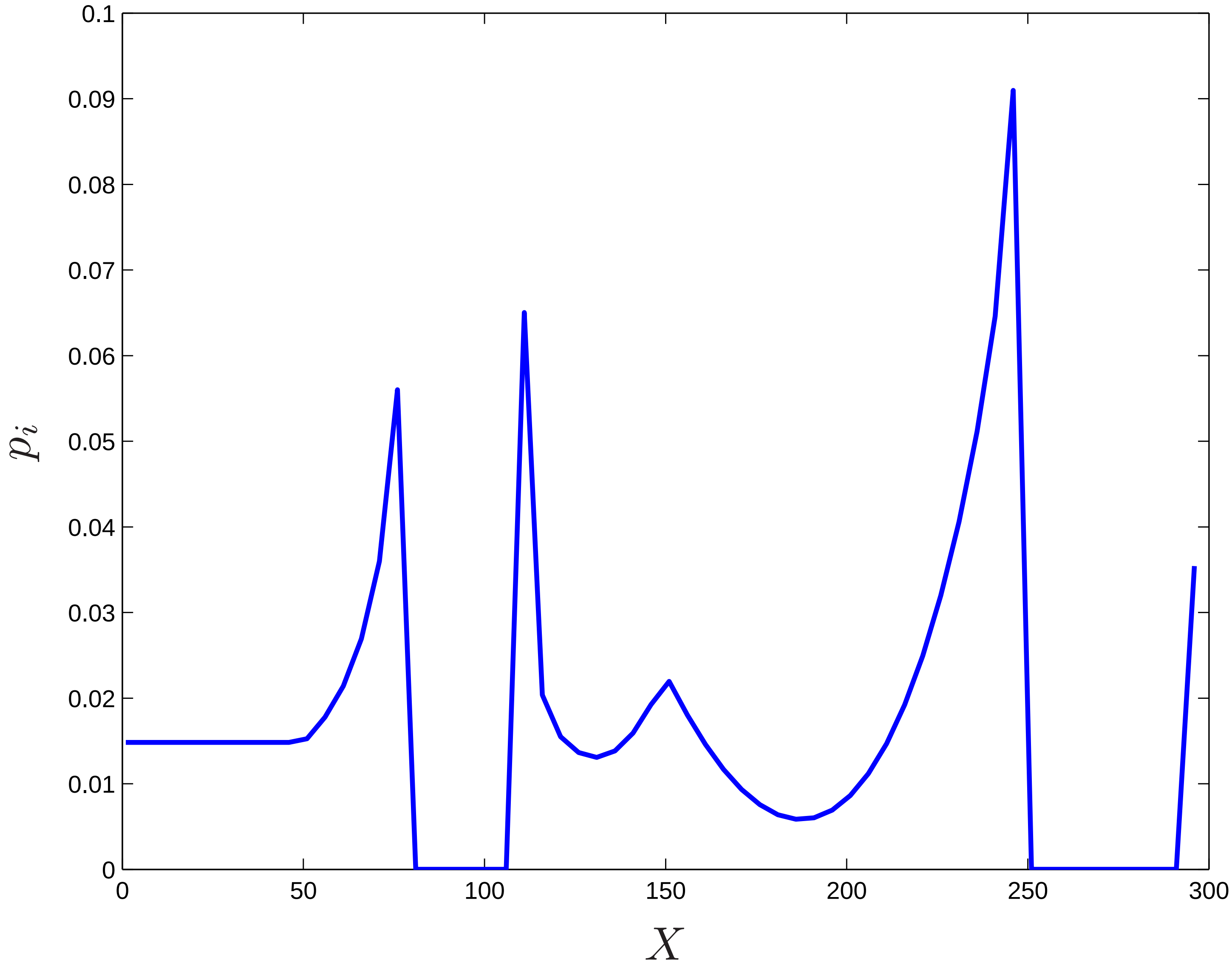
$$K = \{50, 80, 110, 120, 150, 280\}$$

and vector of prices for the 6 options.

$$q = (102.9167, 79.5667, 59.2167, 53.1000, 36.7500, 0.5667)$$

- Prices were computed above using the **uniform** distribution on $[0, 300]$
- **Result**: maximum price for the CutCall is **59**
- Next slide: risk neutral distribution for that maximal price.

Corresponding Risk-Neutral Probability



Option pricing

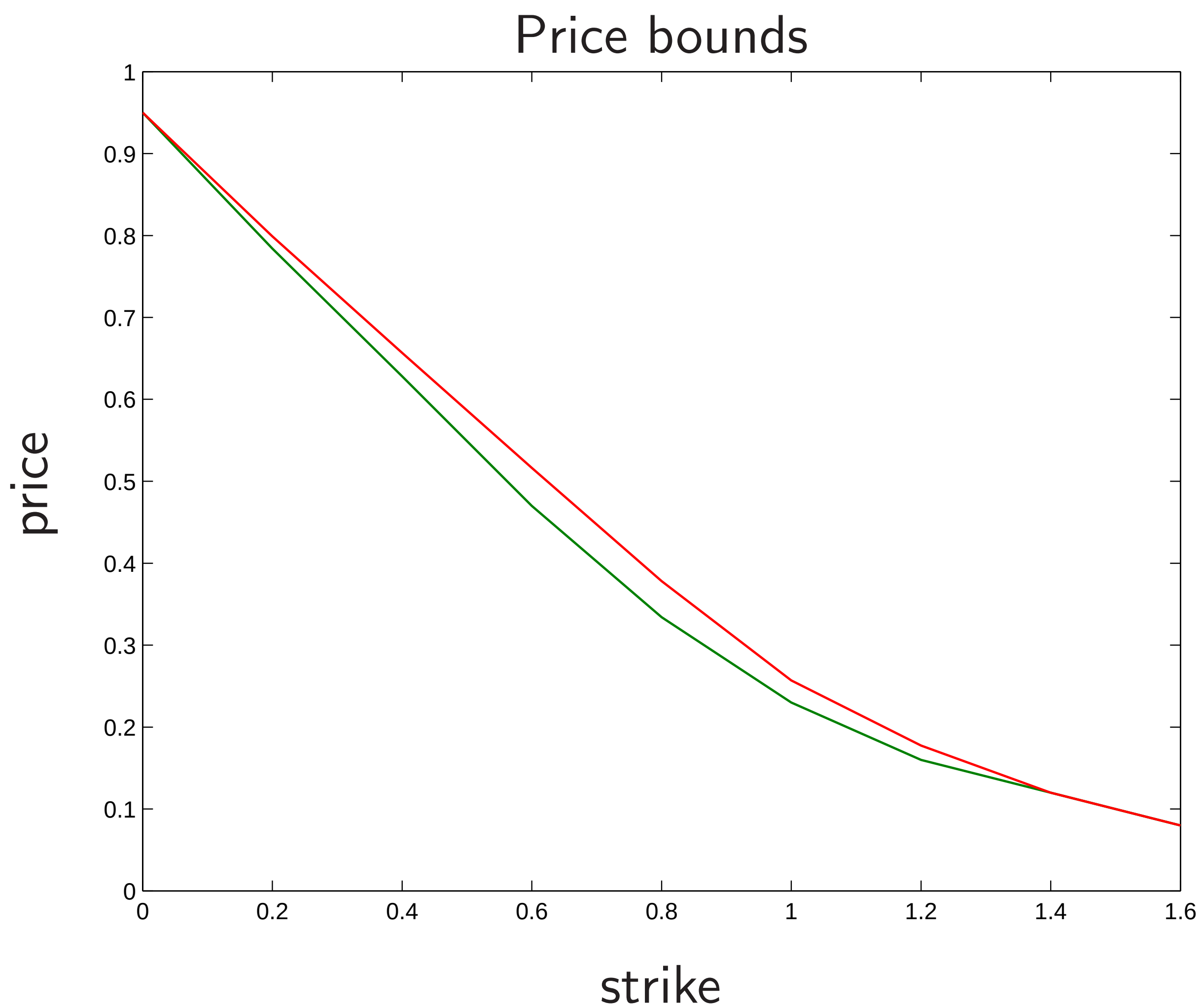
- Problem in dimension 2, price a **basket options** with payoff

$$(x_1 + x_2 - K)_+$$

- The input data set is composed of the asset prices together with the following call prices:

$$(.2x_1 + x_2 - .1)_+,\ (.5x_1 + .8x_2 - .8)_+,$$
$$(.5x_1 + .3x_2 - .4)_+,\ (x_1 + .3x_2 - .5)_+,$$
$$(x_1 + .5x_2 - .5)_+,\ (x_1 + .4x_2 - 1)_+,$$
$$(x_1 + .6x_2 - 1.2)_+.$$

Option pricing



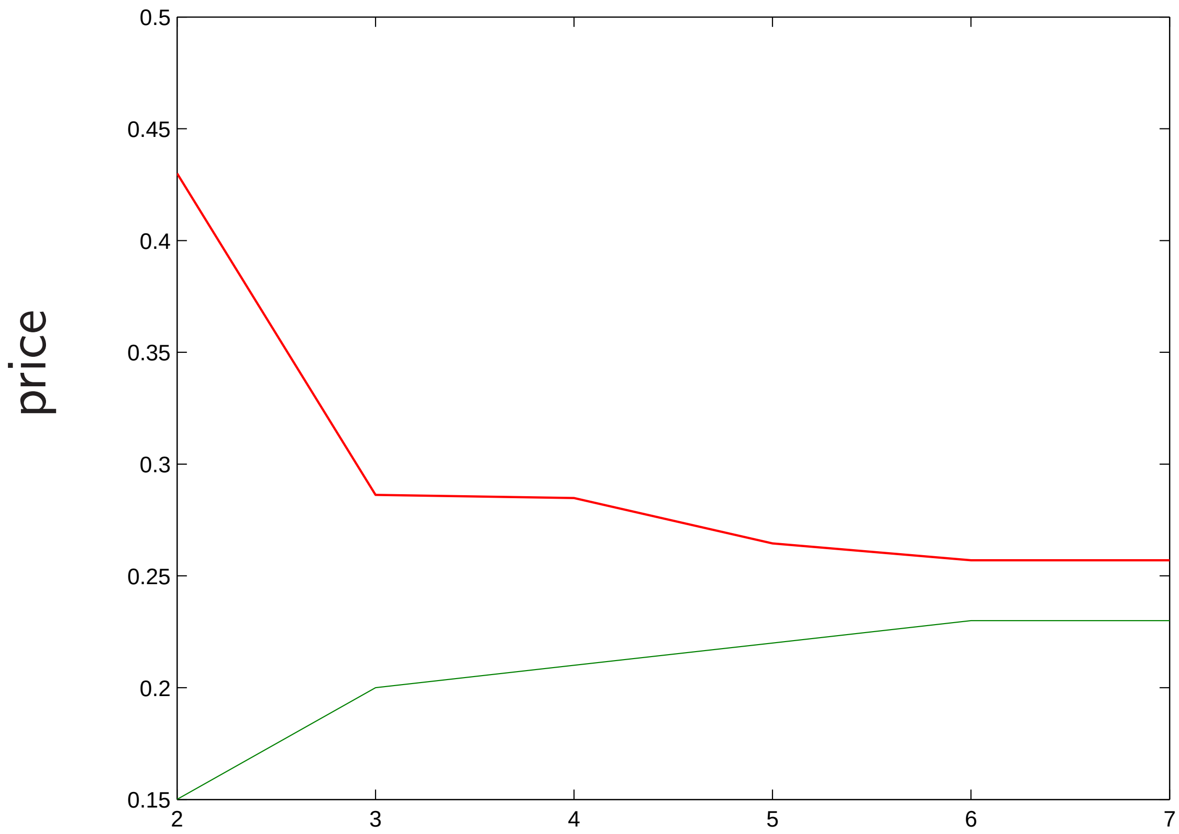
Option pricing

Run another test:

- Look at how these bounds evolve as more and more instruments are incorporated into the data set.
- Fix $K = 1$, we compute the bounds using only the k first instruments in the data set, for $k = 2, \ldots, 7$.
- Plot the **upper** and **lower** bounds
- Also plot one of the solutions

Conclusion: **more market values** $\Rightarrow$ **tighter bounds**

Option pricing



Option pricing

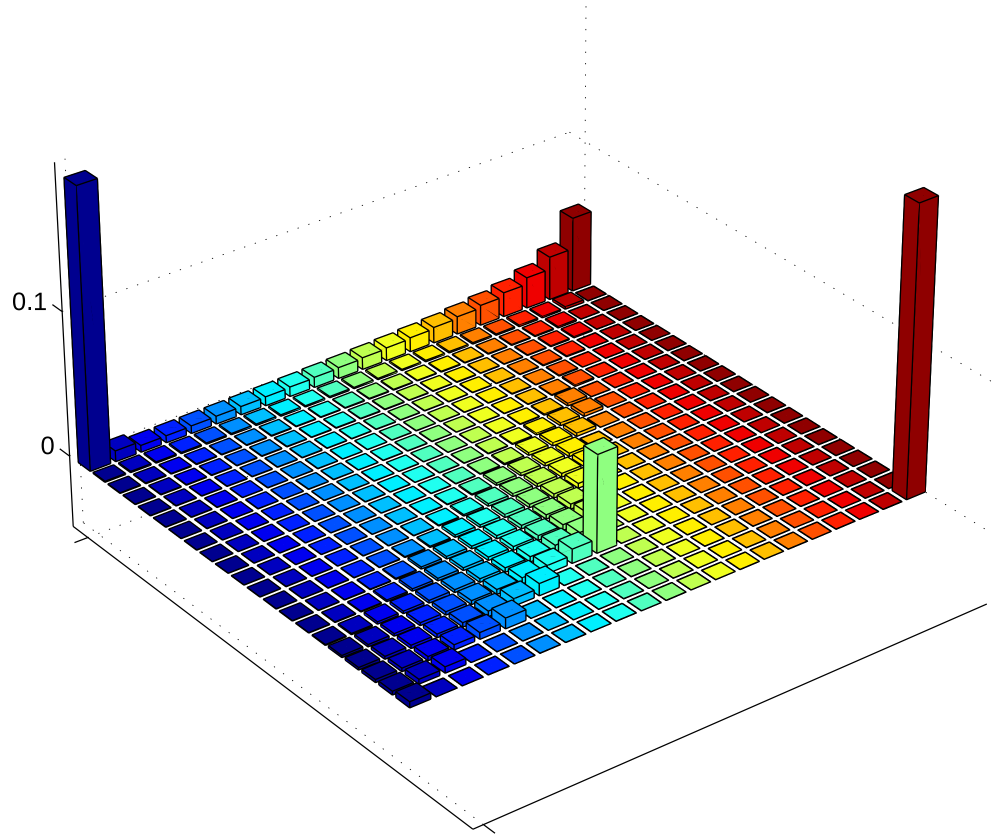


Figure 1: Example of discrete distribution minimizing the price of $(x_1 + x_2 - K)_+$.

Caveats

Size!

- Grows **exponentially** in k^n with the number of points
- Only works with **discrete** and **bounded** models

Everything comes at a price. . .

Duality in a more general setting

Example: Two-way partitioning

$$\begin{array}{ll} \text{minimize} & x^T W x \\ \text{subject to} & x_i^2 = 1, \quad i = 1, \ldots, n \end{array}$$

- a nonconvex problem; feasible set contains 2^n discrete points
- interpretation: partition $\{1, \ldots, n\}$ in two sets; W_{ij} is cost of assigning i, j to the same set; $-W_{ij}$ is cost of assigning to different sets

dual function

$$\begin{aligned} g(\nu) = \inf_x (x^T W x + \sum_i \nu_i (x_i^2 - 1)) &= \inf_x x^T (W + \mathbf{diag}(\nu)) x - \mathbf{1}^T \nu \\ &= \begin{cases} -\mathbf{1}^T \nu & W + \mathbf{diag}(\nu) \succeq 0 \\ -\infty & \text{otherwise} \end{cases} \end{aligned}$$

lower bound property: $p^\star \geq -\mathbf{1}^T \nu$ if $W + \mathbf{diag}(\nu) \succeq 0$

example: $\nu = -\lambda_{\min}(W)\mathbf{1}$ gives bound $p^\star \geq n\lambda_{\min}(W)$

Lagrange dual and conjugate function

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & Ax \preceq b, \quad Cx = d \end{array}$$

dual function

$$\begin{aligned} g(\lambda, \nu) &= \inf_{x \in \mathbf{dom}\, f_0} \left(f_0(x) + (A^T\lambda + C^T\nu)^T x - b^T\lambda - d^T\nu \right) \\ &= -f_0^*(-A^T\lambda - C^T\nu) - b^T\lambda - d^T\nu \end{aligned}$$

- f_0^* is the **convex conjugate** of f_0: $f^*(y) = \sup_{x \in \mathbf{dom}\, f}(y^Tx - f(x))$
- simplifies derivation of dual if conjugate of f_0 is known

example: entropy maximization

$$f_0(x) = \sum_{i=1}^n x_i \log x_i, \qquad f_0^*(y) = \sum_{i=1}^n e^{y_i - 1}$$

Quadratic program

primal problem (assume $P \in \mathbf{S}^n_{++}$)

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

dual function

$$g(\lambda) = \inf_x \left(x^T P x + \lambda^T (Ax - b) \right) = -\frac{1}{4} \lambda^T A P^{-1} A^T \lambda - b^T \lambda$$

dual problem

$$\begin{array}{ll} \text{maximize} & -(1/4)\lambda^T A P^{-1} A^T \lambda - b^T \lambda \\ \text{subject to} & \lambda \succeq 0 \end{array}$$

- from Slater's condition: $p^\star = d^\star$ if $A\tilde{x} \prec b$ for some $\tilde{x}$
- in fact, $p^\star = d^\star$ always

A nonconvex problem with strong duality

$$\begin{array}{ll} \text{minimize} & x^T A x + 2b^T x \\ \text{subject to} & x^T x \leq 1 \end{array}$$

nonconvex if $A \not\succeq 0$

dual function: $g(\lambda) = \inf_x (x^T(A + \lambda I)x + 2b^T x - \lambda)$

- unbounded below if $A + \lambda I \not\succeq 0$ or if $A + \lambda I \succeq 0$ and $b \notin \mathcal{R}(A + \lambda I)$
- minimized by $x = -(A + \lambda I)^\dagger b$ otherwise: $g(\lambda) = -b^T(A + \lambda I)^\dagger b - \lambda$

dual problem:

$$\begin{array}{ll} \text{maximize} & -b^T(A + \lambda I)^\dagger b - \lambda \\ \text{subject to} & A + \lambda I \succeq 0 \\ & b \in \mathcal{R}(A + \lambda I) \end{array}$$

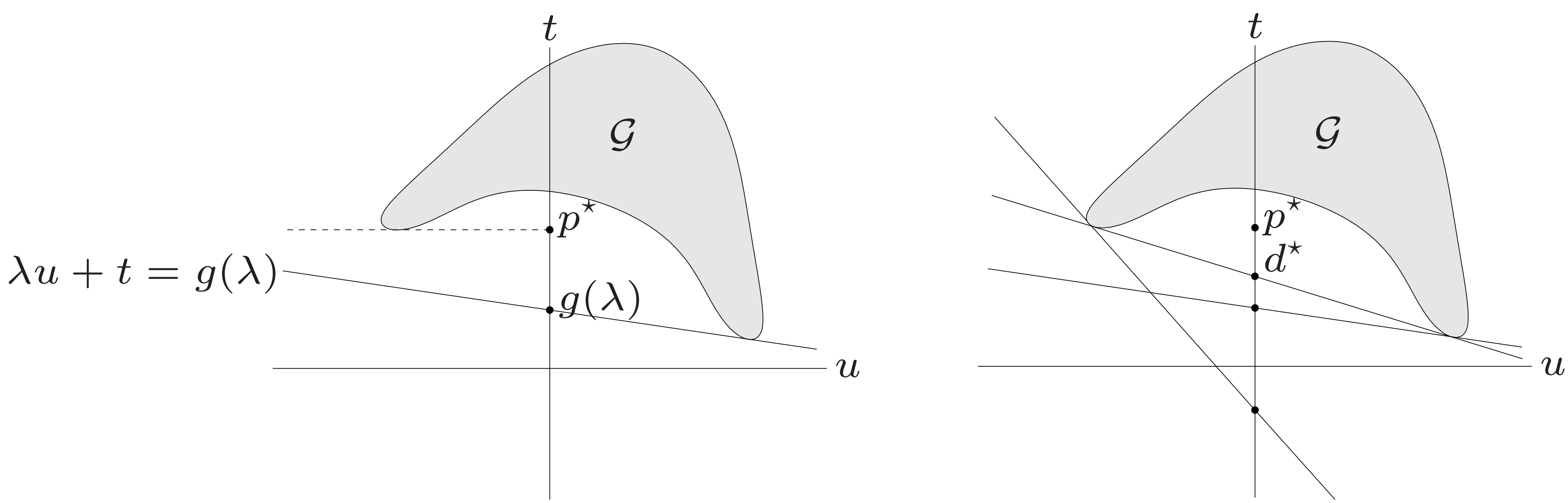
strong duality although primal problem is not convex (not easy to show)

Geometric interpretation

for simplicity, consider problem with one constraint $f_1(x) \leq 0$

interpretation of dual function:

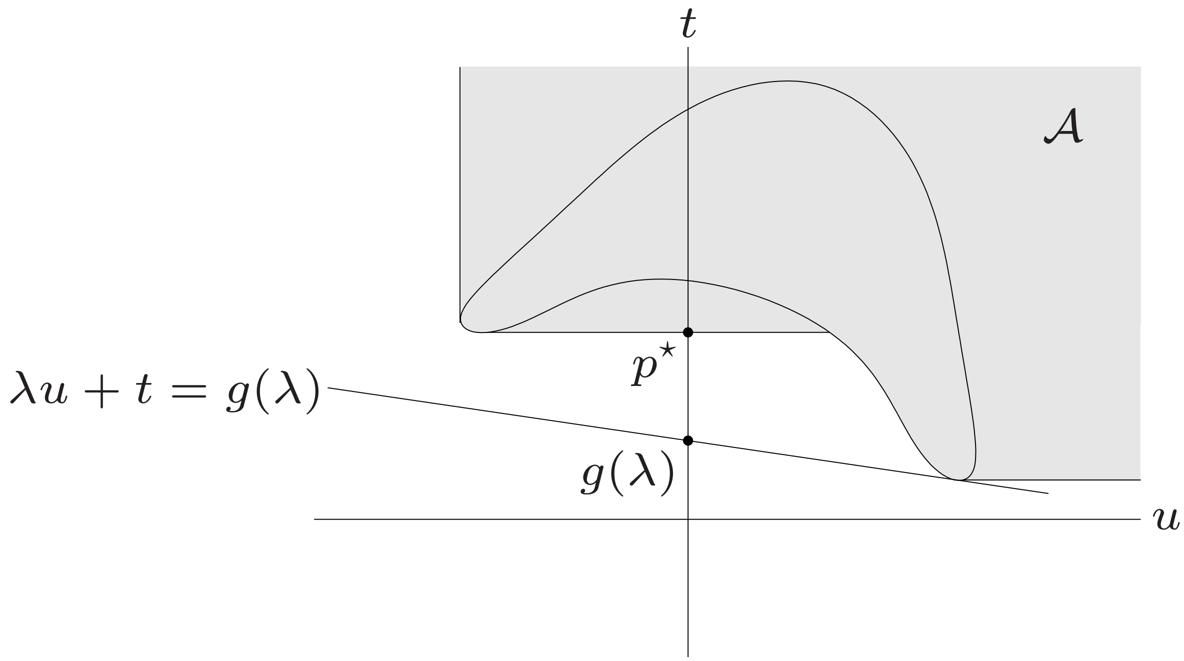
$$g(\lambda) = \inf_{(u,t)\in\mathcal{G}} (t + \lambda u), \qquad \text{where} \quad \mathcal{G} = \{(f_1(x), f_0(x)) \mid x \in \mathcal{D}\}$$



- $\lambda u + t = g(\lambda)$ is (non-vertical) supporting hyperplane to $\mathcal{G}$
- hyperplane intersects t-axis at $t = g(\lambda)$

epigraph variation: same interpretation if $\mathcal{G}$ is replaced with

$$\mathcal{A} = \{(u,t) \mid f_1(x) \le u, f_0(x) \le t \text{ for some } x \in \mathcal{D}\}$$



strong duality

- holds if there is a non-vertical supporting hyperplane to $\mathcal{A}$ at $(0, p^\star)$
- for convex problem, $\mathcal{A}$ is convex, hence has supp. hyperplane at $(0, p^\star)$
- Slater's condition: if there exist $(\tilde{u}, \tilde{t}) \in \mathcal{A}$ with $\tilde{u} < 0$, then supporting hyperplanes at $(0, p^\star)$ must be non-vertical

Complementary slackness

assume strong duality holds, $x^\star$ is primal optimal, $(\lambda^\star, \nu^\star)$ is dual optimal

$$
\begin{aligned}
f_0(x^\star) = g(\lambda^\star, \nu^\star) &= \inf_x \left(f_0(x) + \sum_{i=1}^m \lambda_i^\star f_i(x) + \sum_{i=1}^p \nu_i^\star h_i(x) \right) \\
&\leq f_0(x^\star) + \sum_{i=1}^m \lambda_i^\star f_i(x^\star) + \sum_{i=1}^p \nu_i^\star h_i(x^\star) \\
&\leq f_0(x^\star)
\end{aligned}
$$

hence, the two inequalities hold with equality

- $x^\star$ minimizes $L(x, \lambda^\star, \nu^\star)$
- $\lambda_i^\star f_i(x^\star) = 0$ for $i = 1, \ldots, m$ (known as complementary slackness):

$$
\lambda_i^\star > 0 \Longrightarrow f_i(x^\star) = 0, \qquad f_i(x^\star) < 0 \Longrightarrow \lambda_i^\star = 0
$$

Karush-Kuhn-Tucker (KKT) conditions

the following four conditions are called KKT conditions (for a problem with **differentiable** f_i, h_i):

1. primal constraints: $f_i(x) \leq 0$, $i = 1, \ldots, m$, $h_i(x) = 0$, $i = 1, \ldots, p$
2. dual constraints: $\lambda \succeq 0$
3. complementary slackness: $\lambda_i f_i(x) = 0$, $i = 1, \ldots, m$
4. gradient of Lagrangian with respect to x vanishes:

$$\nabla f_0(x) + \sum_{i=1}^{m} \lambda_i \nabla f_i(x) + \sum_{i=1}^{p} \nu_i \nabla h_i(x) = 0$$

if strong duality holds and x, λ, ν are optimal, then they must satisfy the KKT conditions

KKT conditions for convex problem

if $\tilde{x}$, $\tilde{\lambda}$, $\tilde{\nu}$ satisfy KKT for a convex problem, then they are optimal:

- from complementary slackness: $f_0(\tilde{x}) = L(\tilde{x}, \tilde{\lambda}, \tilde{\nu})$
- from 4th condition (and convexity): $g(\tilde{\lambda}, \tilde{\nu}) = L(\tilde{x}, \tilde{\lambda}, \tilde{\nu})$

hence, $f_0(\tilde{x}) = g(\tilde{\lambda}, \tilde{\nu})$

if **Slater's condition** is satisfied:

x is optimal if and only if there exist λ, ν that satisfy KKT conditions

- recall that Slater implies strong duality, and dual optimum is attained
- generalizes optimality condition $\nabla f_0(x) = 0$ for unconstrained problem

example: water-filling (assume $\alpha_i > 0$)

$$\begin{array}{ll} \text{minimize} & -\sum_{i=1}^n \log(x_i + \alpha_i) \\ \text{subject to} & x \succeq 0, \quad \mathbf{1}^T x = 1 \end{array}$$

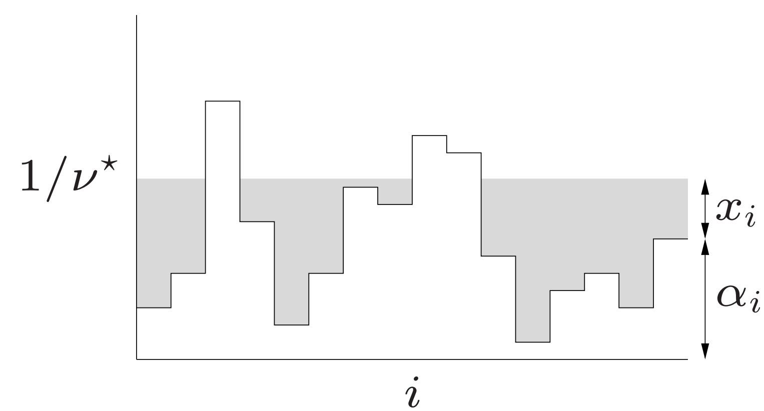
x is optimal iff $x \succeq 0$, $\mathbf{1}^T x = 1$, and there exist $\lambda \in \mathbf{R}^n$, $\nu \in \mathbf{R}$ such that

$$\lambda \succeq 0, \qquad \lambda_i x_i = 0, \qquad \frac{1}{x_i + \alpha_i} + \lambda_i = \nu$$

- if $\nu < 1/\alpha_i$: $\lambda_i = 0$ and $x_i = 1/\nu - \alpha_i$
- if $\nu \geq 1/\alpha_i$: $\lambda_i = \nu - 1/\alpha_i$ and $x_i = 0$
- determine ν from $\mathbf{1}^T x = \sum_{i=1}^n \max\{0, 1/\nu - \alpha_i\} = 1$

interpretation

- n patches; level of patch i is at height α_i
- flood area with unit amount of water
- resulting level is $1/\nu^\star$



Unconstrained Convex Optimization Algorithms

- terminology and assumptions
- gradient descent method
- steepest descent method
- Newton's method
- self-concordant functions
- implementation

Unconstrained minimization

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

- f convex, twice continuously differentiable (hence $\mathbf{dom}\, f$ open)
- we assume optimal value $p^\star = \inf_x f(x)$ is attained (and finite)

unconstrained minimization methods

- produce sequence of points $x^{(k)} \in \mathbf{dom}\, f$, $k = 0, 1, \ldots$ with

$$f(x^{(k)}) \to p^\star$$

- can be interpreted as iterative methods for solving optimality condition

$$\nabla f(x^\star) = 0$$

Initial point and sublevel set

algorithms in this chapter require a starting point $x^{(0)}$ such that

- $x^{(0)} \in \mathbf{dom}\, f$
- sublevel set $S = \{x \mid f(x) \leq f(x^{(0)})\}$ is closed

2nd condition is hard to verify, except when *all* sublevel sets are closed:

- equivalent to condition that $\mathbf{epi}\, f$ is closed
- true if $\mathbf{dom}\, f = \mathbf{R}^n$
- true if $f(x) \to \infty$ as $x \to \mathbf{d}\,\mathbf{dom}\, f$

examples of differentiable functions with closed sublevel sets:

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

Strong convexity and implications

f is strongly convex on S if there exists an $m > 0$ such that

$$\nabla^2 f(x) \succeq mI \qquad \text{for all } x \in S$$

implications

- for $x, y \in S$,

$$f(y) \geq f(x) + \nabla f(x)^T (y - x) + \frac{m}{2} \|x - y\|_2^2$$

  hence, S is bounded

- $p^\star > -\infty$, and for $x \in S$,

$$f(x) - p^\star \leq \frac{1}{2m} \|\nabla f(x)\|_2^2$$

  useful as stopping criterion (if you know m)

Descent methods

$$x^{(k+1)} = x^{(k)} + t^{(k)}\Delta x^{(k)} \quad \text{with } f(x^{(k+1)}) < f(x^{(k)})$$

- other notations: $x^+ = x + t\Delta x$, $x := x + t\Delta x$
- Δx is the *step*, or *search direction*; t is the *step size*, or *step length*
- from convexity, $f(x^+) < f(x)$ implies $\nabla f(x)^T \Delta x < 0$
  ($i.e.$, Δx is a *descent direction*)

General descent method.

given a starting point $x \in \mathbf{dom}\, f$.
repeat
1. Determine a descent direction Δx.
2. *Line search.* Choose a step size $t > 0$.
3. *Update.* $x := x + t\Delta x$.

until stopping criterion is satisfied.

Line search types

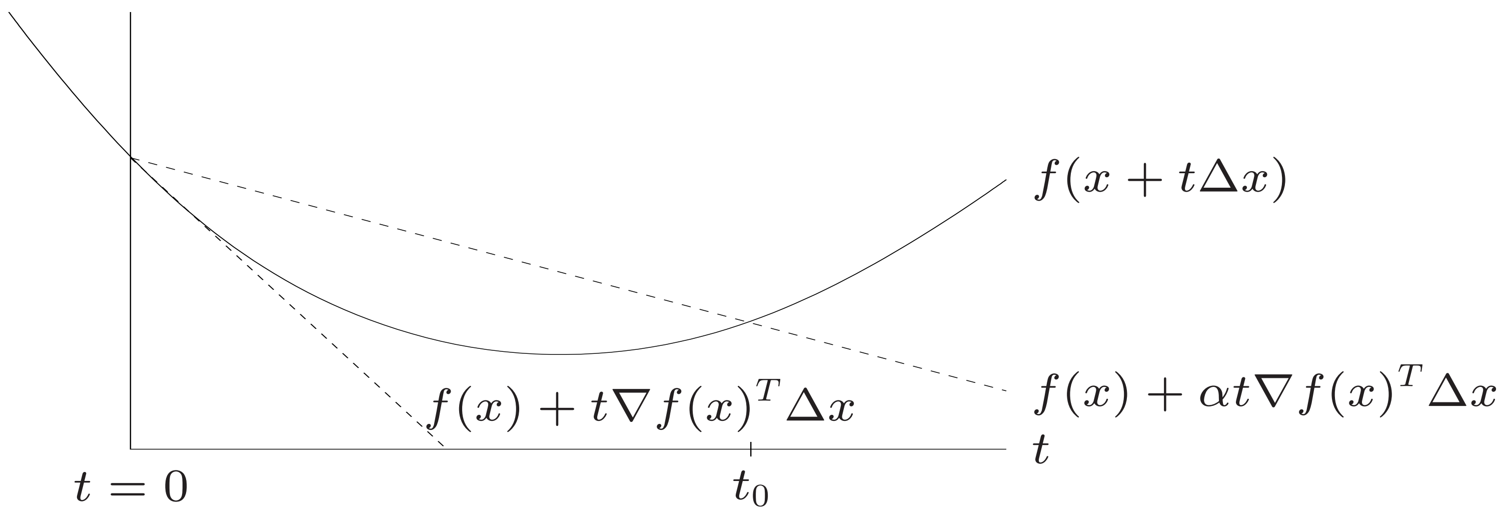
exact line search: $t = \operatorname{argmin}_{t>0} f(x + t\Delta x)$

backtracking line search (with parameters $\alpha \in (0, 1/2)$, $\beta \in (0, 1)$)

- starting at $t = 1$, repeat $t := \beta t$ until

$$f(x + t\Delta x) < f(x) + \alpha t \nabla f(x)^T \Delta x$$

- graphical interpretation: backtrack until $t \leq t_0$



Gradient descent method

general descent method with $\Delta x = -\nabla f(x)$

> **given** a starting point $x \in \mathbf{dom}\, f$.
> **repeat**
> 1. $\Delta x := -\nabla f(x)$.
> 2. *Line search.* Choose step size t via exact or backtracking line search.
> 3. *Update.* $x := x + t\Delta x$.
>
> **until** stopping criterion is satisfied.

- stopping criterion usually of the form $\|\nabla f(x)\|_2 \leq \epsilon$
- convergence result: for strongly convex f,

$$f(x^{(k)}) - p^\star \leq c^k (f(x^{(0)}) - p^\star)$$

  $c \in (0, 1)$ depends on m, $x^{(0)}$, line search type

- very simple, but often very slow; rarely used in practice

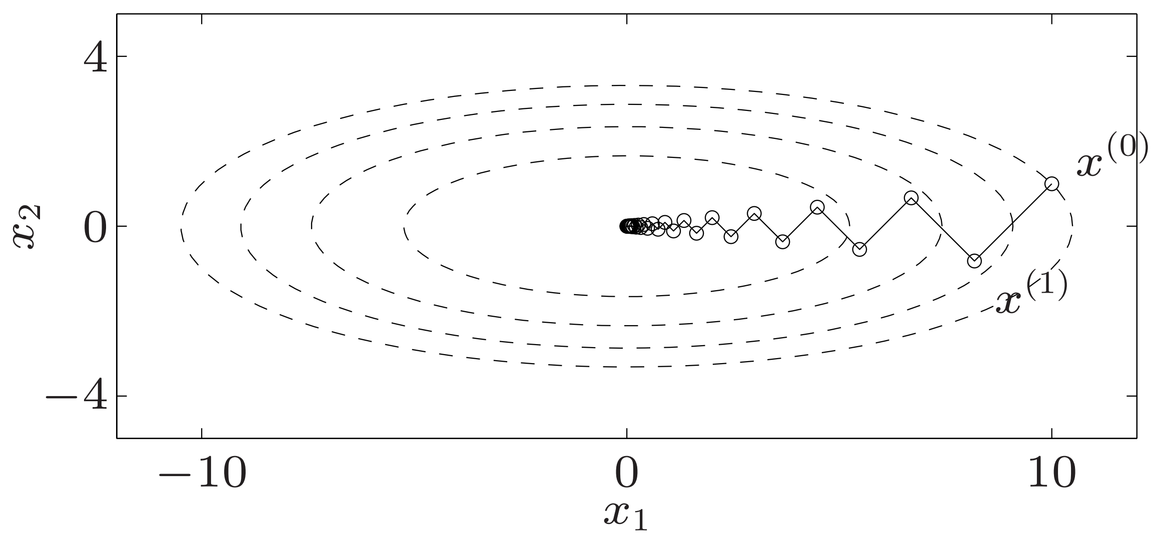
quadratic problem in $\mathbf{R}^2$

$$f(x) = (1/2)(x_1^2 + \gamma x_2^2) \qquad\qquad (\gamma > 0)$$

with exact line search, starting at $x^{(0)} = (\gamma, 1)$:

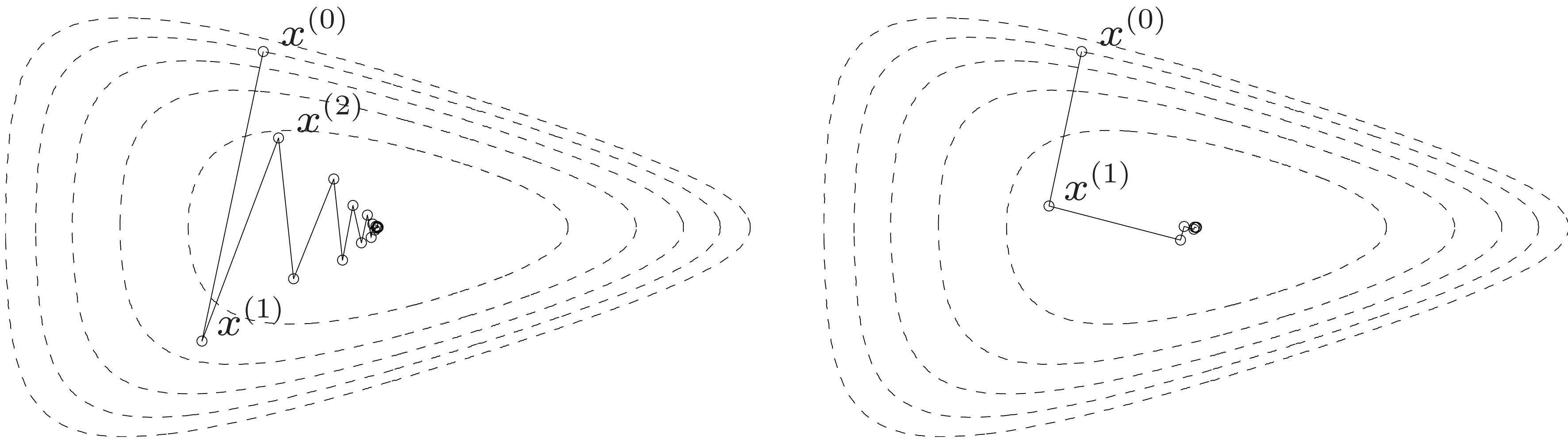
$$x_1^{(k)} = \gamma \left(\frac{\gamma - 1}{\gamma + 1}\right)^k, \qquad x_2^{(k)} = \left(-\frac{\gamma - 1}{\gamma + 1}\right)^k$$

- very slow if $\gamma \gg 1$ or $\gamma \ll 1$
- example for $\gamma = 10$:



nonquadratic example

$$f(x_1, x_2) = e^{x_1+3x_2-0.1} + e^{x_1-3x_2-0.1} + e^{-x_1-0.1}$$

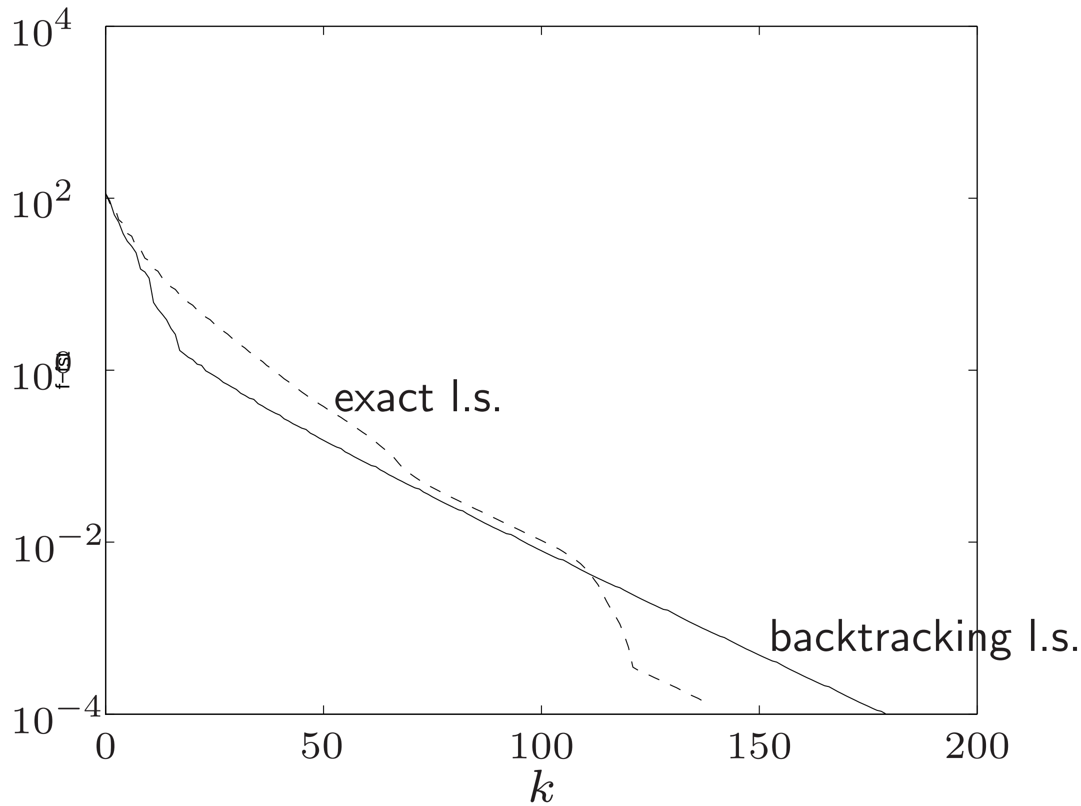


backtracking line search

exact line search

a problem in $\mathbf{R}^{100}$

$$f(x) = c^T x - \sum_{i=1}^{500} \log(b_i - a_i^T x)$$



'linear' convergence, *i.e.*, a straight line on a semilog plot

Steepest descent method

normalized steepest descent direction (at x, for norm $\|\cdot\|$):

$$\Delta x_{\mathrm{nsd}} = \mathrm{argmin}\{\nabla f(x)^T v \mid \|v\| = 1\}$$

interpretation: for small v, $f(x+v) \approx f(x) + \nabla f(x)^T v$;
direction Δx_{nsd} is unit-norm step with most negative directional derivative

(unnormalized) steepest descent direction

$$\Delta x_{\mathrm{sd}} = \|\nabla f(x)\|_* \Delta x_{\mathrm{nsd}}$$

satisfies $\nabla f(x)^T \Delta_{\mathrm{sd}} = -\|\nabla f(x)\|_*^2$

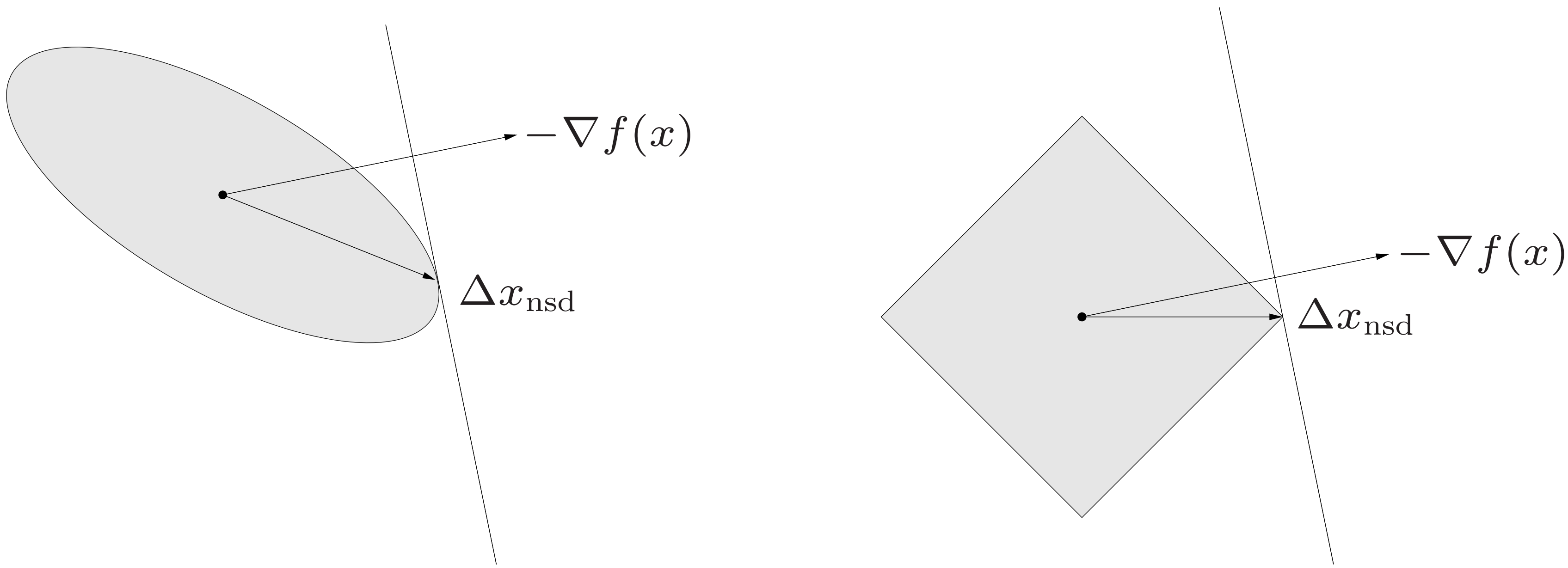
steepest descent method

- general descent method with $\Delta x = \Delta x_{\mathrm{sd}}$
- convergence properties similar to gradient descent

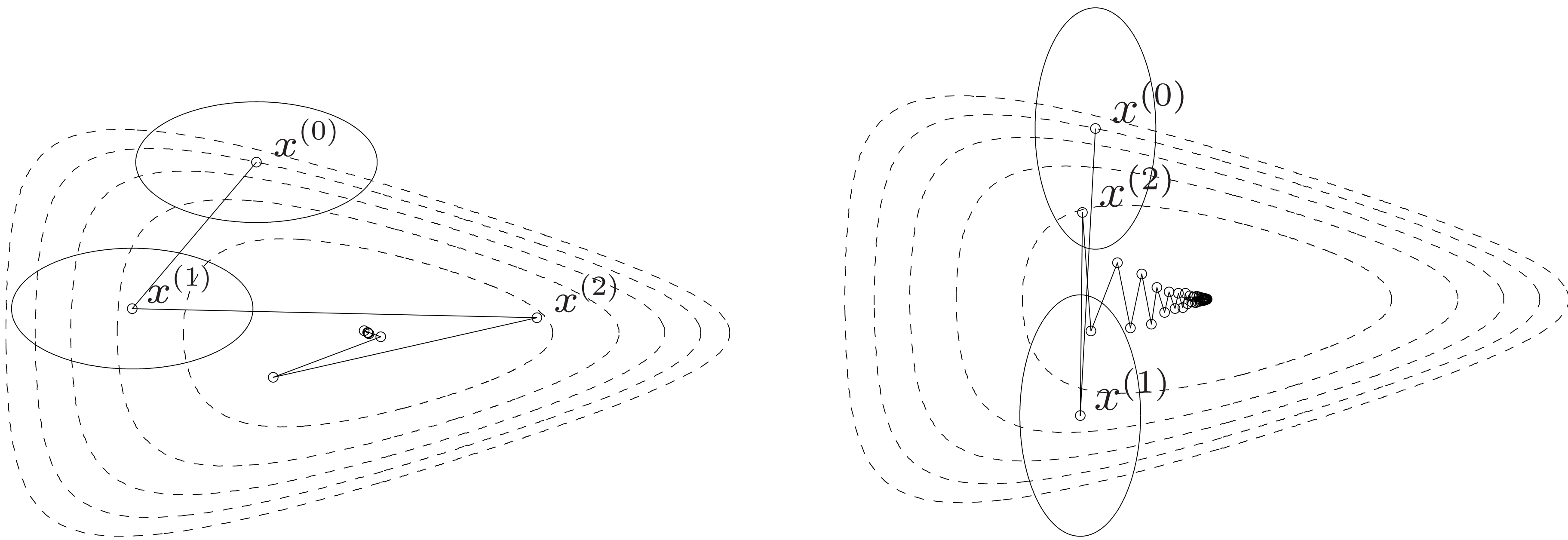
examples

- Euclidean norm: $\Delta x_{\rm sd} = -\nabla f(x)$
- quadratic norm $\|x\|_P = (x^T P x)^{1/2}$ ($P \in \mathbf{S}^n_{++}$): $\Delta x_{\rm sd} = -P^{-1}\nabla f(x)$
- ℓ_1-norm: $\Delta x_{\rm sd} = -(\partial f(x)/\partial x_i)e_i$, where $|\partial f(x)/\partial x_i| = \|\nabla f(x)\|_\infty$

unit balls and normalized steepest descent directions for a quadratic norm and the ℓ_1-norm:



choice of norm for steepest descent



- steepest descent with backtracking line search for two quadratic norms
- ellipses show $\{x \mid \|x - x^{(k)}\|_P = 1\}$
- equivalent interpretation of steepest descent with quadratic norm $\|\cdot\|_P$: gradient descent after change of variables $\bar{x} = P^{1/2}x$

shows choice of P has strong effect on speed of convergence

Newton step

$$\Delta x_{\rm nt} = -\nabla^2 f(x)^{-1} \nabla f(x)$$

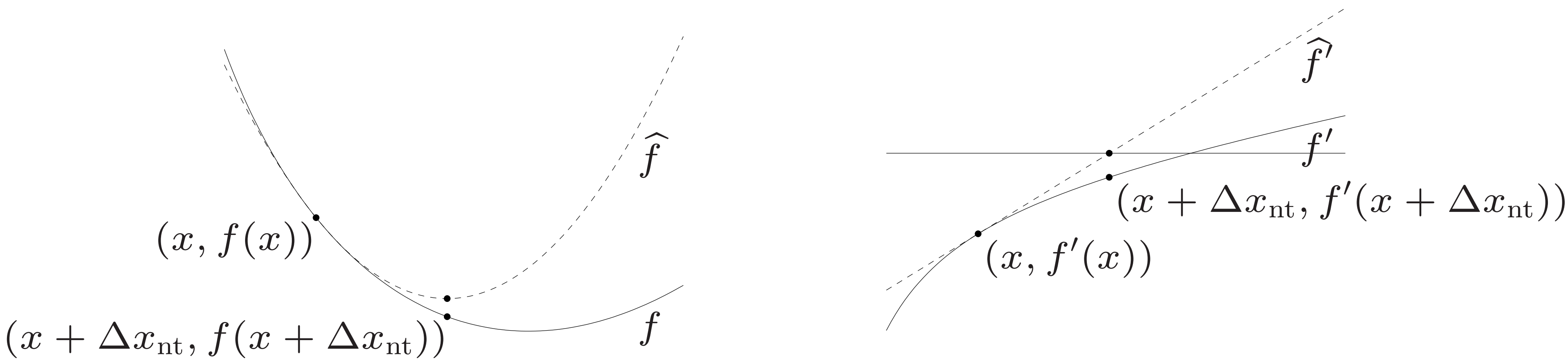
interpretations

- $x + \Delta x_{\rm nt}$ minimizes second order approximation

$$\widehat{f}(x+v) = f(x) + \nabla f(x)^T v + \frac{1}{2} v^T \nabla^2 f(x) v$$

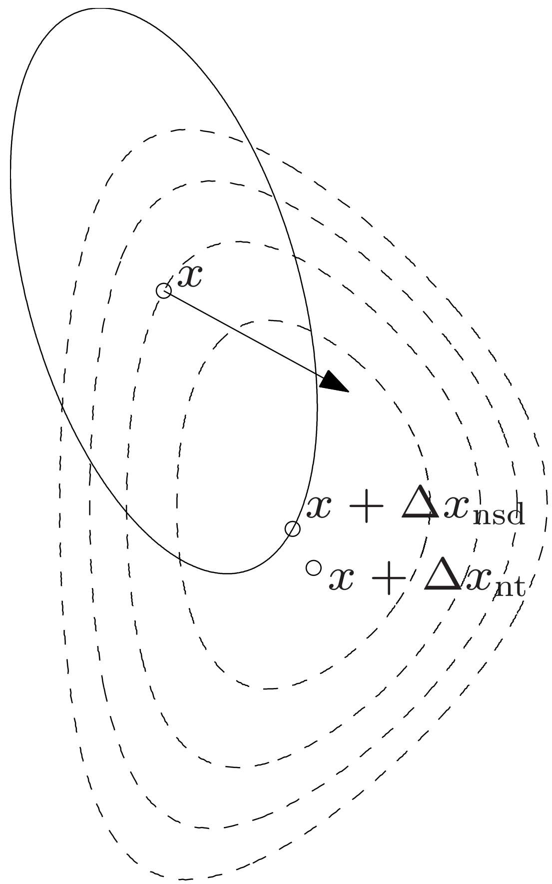
- $x + \Delta x_{\rm nt}$ solves linearized optimality condition

$$\nabla f(x+v) \approx \nabla \widehat{f}(x+v) = \nabla f(x) + \nabla^2 f(x) v = 0$$



- Δx_{nt} is steepest descent direction at x in local Hessian norm

$$\|u\|_{\nabla^2 f(x)} = \left(u^T \nabla^2 f(x) u\right)^{1/2}$$



dashed lines are contour lines of f; ellipse is $\{x + v \mid v^T \nabla^2 f(x) v = 1\}$

arrow shows $-\nabla f(x)$

Newton decrement

$$\lambda(x) = \left(\nabla f(x)^T \nabla^2 f(x)^{-1} \nabla f(x)\right)^{1/2}$$

a measure of the proximity of x to $x^\star$

properties

- gives an estimate of $f(x) - p^\star$, using quadratic approximation $\widehat{f}$:

$$f(x) - \inf_y \widehat{f}(y) = \frac{1}{2}\lambda(x)^2$$

- equal to the norm of the Newton step in the quadratic Hessian norm

$$\lambda(x) = \left(\Delta x_{\mathrm{nt}} \nabla^2 f(x) \Delta x_{\mathrm{nt}}\right)^{1/2}$$

- directional derivative in the Newton direction: $\nabla f(x)^T \Delta x_{\mathrm{nt}} = -\lambda(x)^2$
- affine invariant (unlike $\|\nabla f(x)\|_2$)

Newton's method

given a starting point $x \in \mathbf{dom}\, f$, tolerance $\epsilon > 0$.
repeat

1. *Compute the Newton step and decrement.*
   $\Delta x_{\mathrm{nt}} := -\nabla^2 f(x)^{-1}\nabla f(x); \quad \lambda^2 := \nabla f(x)^T \nabla^2 f(x)^{-1} \nabla f(x).$
2. *Stopping criterion.* **quit** if $\lambda^2/2 \leq \epsilon$.
3. *Line search.* Choose step size t by backtracking line search.
4. *Update.* $x := x + t\Delta x_{\mathrm{nt}}$.

affine invariant, $i.e.$, independent of linear changes of coordinates:

Newton iterates for $\tilde{f}(y) = f(Ty)$ with starting point $y^{(0)} = T^{-1}x^{(0)}$ are

$$y^{(k)} = T^{-1}x^{(k)}$$

Classical convergence analysis

assumptions

- f strongly convex on S with constant m
- $\nabla^2 f$ is Lipschitz continuous on S, with constant $L > 0$:

$$\|\nabla^2 f(x) - \nabla^2 f(y)\|_2 \leq L\|x - y\|_2$$

  (L measures how well f can be approximated by a quadratic function)

outline: there exist constants $\eta \in (0, m^2/L)$, $\gamma > 0$ such that

- if $\|\nabla f(x)\|_2 \geq \eta$, then $f(x^{(k+1)}) - f(x^{(k)}) \leq -\gamma$
- if $\|\nabla f(x)\|_2 < \eta$, then

$$\frac{L}{2m^2}\|\nabla f(x^{(k+1)})\|_2 \leq \left(\frac{L}{2m^2}\|\nabla f(x^{(k)})\|_2\right)^2$$

damped Newton phase ($\|\nabla f(x)\|_2 \geq \eta$)

- most iterations require backtracking steps
- function value decreases by at least γ
- if $p^\star > -\infty$, this phase ends after at most $(f(x^{(0)}) - p^\star)/\gamma$ iterations

quadratically convergent phase ($\|\nabla f(x)\|_2 < \eta$)

- all iterations use step size $t = 1$
- $\|\nabla f(x)\|_2$ converges to zero quadratically: if $\|\nabla f(x^{(k)})\|_2 < \eta$, then

$$\frac{L}{2m^2}\|\nabla f(x^l)\|_2 \leq \left(\frac{L}{2m^2}\|\nabla f(x^k)\|_2\right)^{2^{l-k}} \leq \left(\frac{1}{2}\right)^{2^{l-k}}, \qquad l \geq k$$

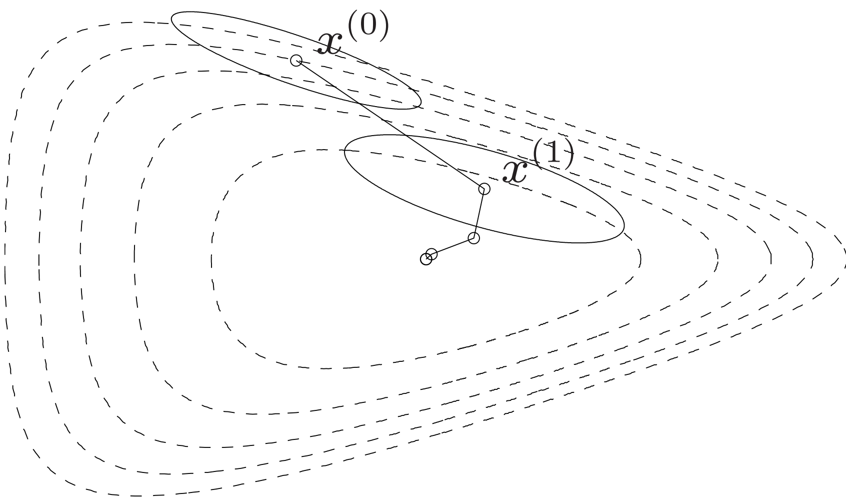
conclusion: number of iterations until $f(x) - p^\star \leq \epsilon$ is bounded above by

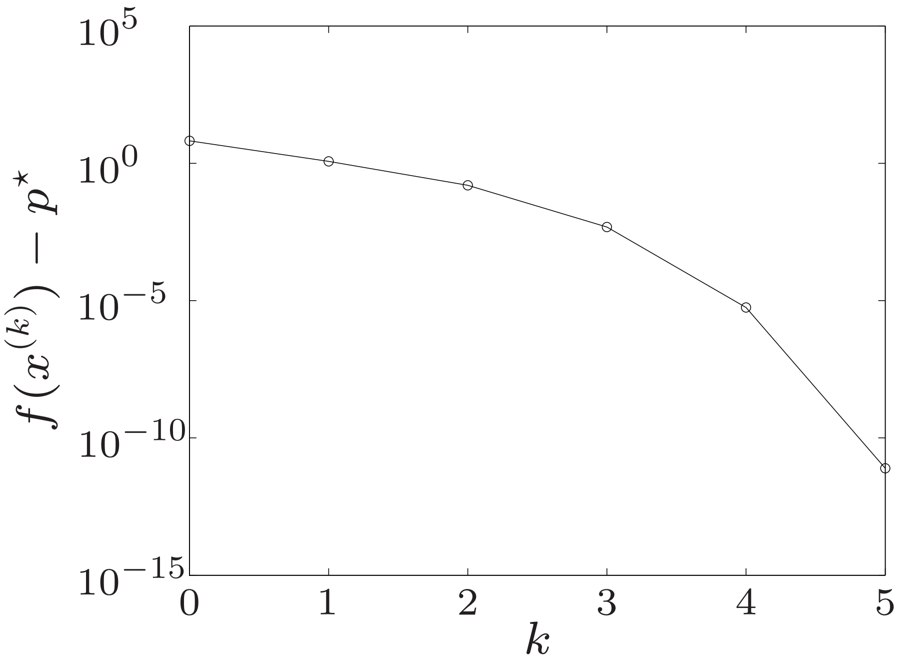
$$\frac{f(x^{(0)}) - p^\star}{\gamma} + \log_2 \log_2(\epsilon_0/\epsilon)$$

- γ, ϵ_0 are constants that depend on m, L, $x^{(0)}$
- second term is small (of the order of 6) and almost constant for practical purposes
- in practice, constants m, L (hence γ, ϵ_0) are usually unknown
- provides qualitative insight in convergence properties ($i.e.$, explains two algorithm phases)

Examples

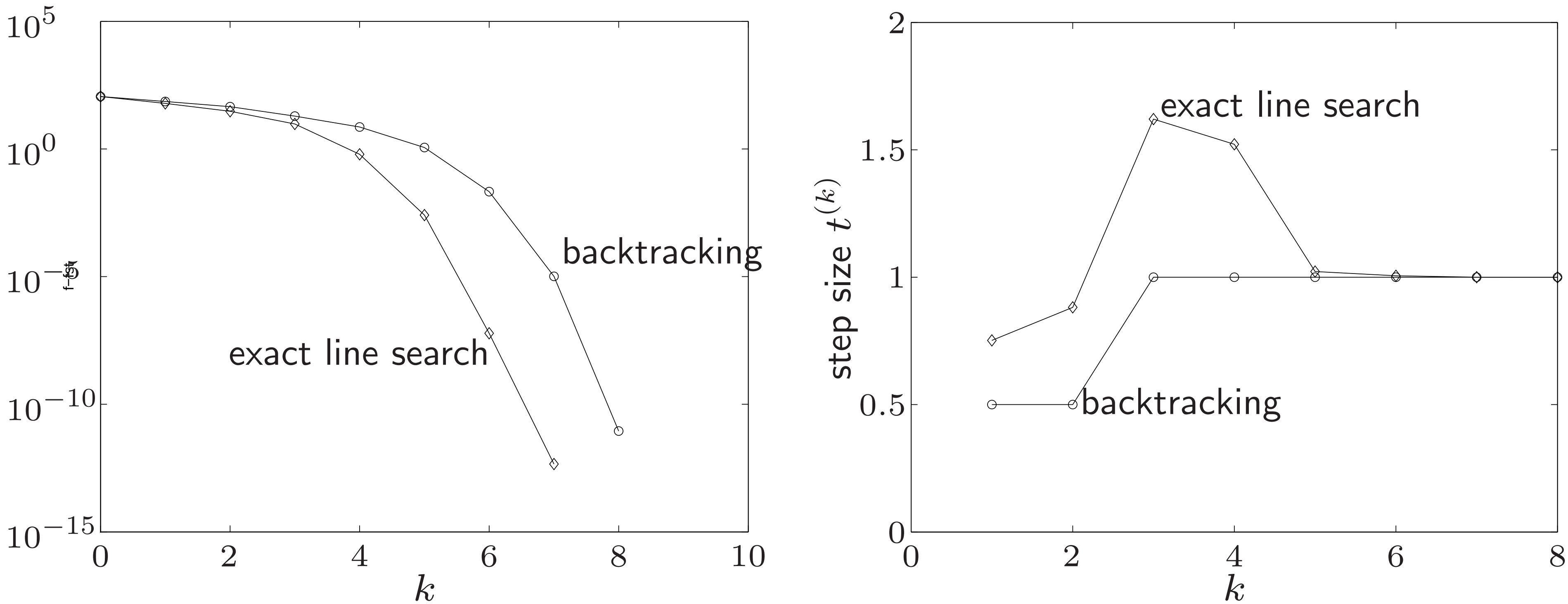
example in $\mathbf{R}^2$ (page 102)





- backtracking parameters $\alpha = 0.1$, $\beta = 0.7$
- converges in only 5 steps
- quadratic local convergence

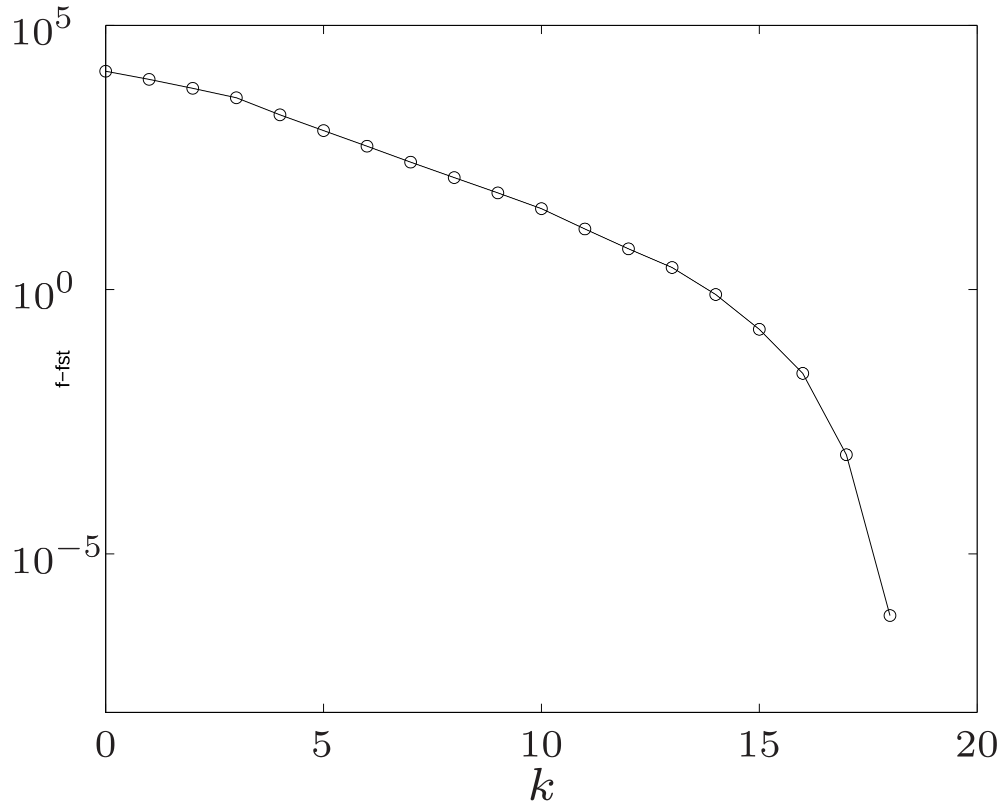
example in $\mathbf{R}^{100}$ (page 103)



- backtracking parameters $\alpha = 0.01$, $\beta = 0.5$
- backtracking line search almost as fast as exact l.s. (and much simpler)
- clearly shows two phases in algorithm

example in $\mathbf{R}^{10000}$ (with sparse a_i)

$$f(x) = -\sum_{i=1}^{10000} \log(1 - x_i^2) - \sum_{i=1}^{100000} \log(b_i - a_i^T x)$$



- backtracking parameters $\alpha = 0.01$, $\beta = 0.5$.
- performance similar as for small examples

A few words on Self-concordance

shortcomings of classical convergence analysis

- depends on unknown constants (m, L, . . .)
- bound is not affinely invariant, although Newton's method is

convergence analysis via self-concordance (Nesterov and Nemirovski)

- does not depend on any unknown constants
- gives affine-invariant bound
- applies to special class of convex functions ('self-concordant' functions)
- developed to analyze polynomial-time interior-point methods for convex optimization
- Please check Boyd & Vandenberghe book for a review!

Implementation

main effort in each iteration: evaluate derivatives and solve Newton system

$$H\Delta x = g$$

where $H = \nabla^2 f(x)$, $g = -\nabla f(x)$

via Cholesky factorization

$$H = LL^T, \qquad \Delta x_{\text{nt}} = L^{-T}L^{-1}g, \qquad \lambda(x) = \|L^{-1}g\|_2$$

- cost $(1/3)n^3$ flops for unstructured system
- cost $\ll (1/3)n^3$ if H sparse, banded

example of dense Newton system with structure

$$f(x) = \sum_{i=1}^{n} \psi_i(x_i) + \psi_0(Ax+b), \qquad H = D + A^T H_0 A$$

- assume $A \in \mathbf{R}^{p\times n}$, dense, with $p \ll n$
- D diagonal with diagonal elements $\psi_i''(x_i)$; $H_0 = \nabla^2 \psi_0(Ax+b)$

method 1: form H, solve via dense Cholesky factorization: (cost $(1/3)n^3$)

method 2: factor $H_0 = L_0 L_0^T$; write Newton system as

$$D\Delta x + A^T L_0 w = -g, \qquad L_0^T A \Delta x - w = 0$$

eliminate Δx from first equation; compute w and Δx from

$$(I + L_0^T A D^{-1} A^T L_0) w = -L_0^T A D^{-1} g, \qquad D\Delta x = -g - A^T L_0 w$$

cost: $2p^2 n$ (dominated by computation of $L_0^T A D^{-1} A L_0$)

Convex Optimization Algorithms With Equality Constraints

- equality constrained minimization
- Newton's method with equality constraints
- infeasible start Newton method
- implementation

Equality constrained minimization

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & Ax = b \end{array}$$

- f convex, twice continuously differentiable
- $A \in \mathbf{R}^{p\times n}$ with $\textbf{Rank}\, A = p$
- we assume $p^\star$ is finite and attained

optimality conditions: $x^\star$ is optimal iff there exists a $\nu^\star$ such that

$$\nabla f(x^\star) + A^T\nu^\star = 0, \qquad Ax^\star = b$$

equality constrained quadratic minimization (with $P \in \mathbf{S}_+^n$)

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

optimality condition:

$$\left[\begin{array}{cc} P & A^T \\ A & 0 \end{array}\right] \left[\begin{array}{c} x^\star \\ \nu^\star \end{array}\right] = \left[\begin{array}{c} -q \\ b \end{array}\right]$$

- coefficient matrix is called KKT matrix
- KKT matrix is nonsingular if and only if

$$Ax = 0, \quad x \neq 0 \qquad \Longrightarrow \qquad x^T P x > 0$$

- equivalent condition for nonsingularity: $P + A^T A \succ 0$

Newton step

Newton step of f at feasible x is given by (1st block) of solution of

$$\begin{bmatrix} \nabla^2 f(x) & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} \Delta x_{\mathrm{nt}} \\ w \end{bmatrix} = \begin{bmatrix} -\nabla f(x) \\ 0 \end{bmatrix}$$

interpretations

- Δx_{nt} solves second order approximation (with variable v)

$$\begin{array}{ll} \text{minimize} & \widehat{f}(x+v) = f(x) + \nabla f(x)^T v + (1/2) v^T \nabla^2 f(x) v \\ \text{subject to} & A(x+v) = b \end{array}$$

- equations follow from linearizing optimality conditions

$$\nabla f(x + \Delta x_{\mathrm{nt}}) + A^T w = 0, \qquad A(x + \Delta x_{\mathrm{nt}}) = b$$

Newton decrement

$$\lambda(x) = \left(\Delta x_{\rm nt}^T \nabla^2 f(x) \Delta x_{\rm nt}\right)^{1/2} = \left(-\nabla f(x)^T \Delta x_{\rm nt}\right)^{1/2}$$

properties

- gives an estimate of $f(x) - p^\star$ using quadratic approximation $\widehat{f}$:

$$f(x) - \inf_{Ay=b} \widehat{f}(y) = \frac{1}{2}\lambda(x)^2$$

- directional derivative in Newton direction:

$$\left.\frac{d}{dt} f(x + t\Delta x_{\rm nt})\right|_{t=0} = -\lambda(x)^2$$

- in general, $\lambda(x) \neq \left(\nabla f(x)^T \nabla^2 f(x)^{-1} \nabla f(x)\right)^{1/2}$

Newton's method with equality constraints

given starting point $x \in \mathbf{dom}\, f$ with $Ax = b$, tolerance $\epsilon > 0$.
repeat
1. Compute the Newton step and decrement $\Delta x_{\rm nt}$, $\lambda(x)$.
2. *Stopping criterion.* **quit** if $\lambda^2/2 \leq \epsilon$.
3. *Line search.* Choose step size t by backtracking line search.
4. *Update.* $x := x + t\Delta x_{\rm nt}$.

- a feasible descent method: $x^{(k)}$ feasible and $f(x^{(k+1)}) < f(x^{(k)})$
- affine invariant

Newton step at infeasible points

extends to infeasible x (*i.e.*, $Ax \neq b$)

linearizing optimality conditions at infeasible x (with $x \in \mathbf{dom}\, f$) gives

$$\begin{bmatrix} \nabla^2 f(x) & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} \Delta x_{\mathrm{nt}} \\ w \end{bmatrix} = - \begin{bmatrix} \nabla f(x) \\ Ax - b \end{bmatrix} \qquad (1)$$

primal-dual interpretation

- write optimality condition as $r(y) = 0$, where

$$y = (x, \nu), \qquad r(y) = (\nabla f(x) + A^T \nu, Ax - b)$$

- linearizing $r(y) = 0$ gives $r(y + \Delta y) \approx r(y) + Dr(y)\Delta y = 0$:

$$\begin{bmatrix} \nabla^2 f(x) & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} \Delta x_{\mathrm{nt}} \\ \Delta \nu_{\mathrm{nt}} \end{bmatrix} = - \begin{bmatrix} \nabla f(x) + A^T \nu \\ Ax - b \end{bmatrix}$$

  same as (1) with $w = \nu + \Delta \nu_{\mathrm{nt}}$

Infeasible start Newton method

given starting point $x \in \textbf{dom}\, f$, ν, tolerance $\epsilon > 0$, $\alpha \in (0, 1/2)$, $\beta \in (0, 1)$.
repeat
1. Compute primal and dual Newton steps $\Delta x_{\rm nt}$, $\Delta \nu_{\rm nt}$.
2. *Backtracking line search on* $\|r\|_2$.
   $t := 1$.
   while $\|r(x + t\Delta x_{\rm nt}, \nu + t\Delta \nu_{\rm nt})\|_2 > (1 - \alpha t)\|r(x, \nu)\|_2$, $\quad t := \beta t$.
3. *Update.* $x := x + t\Delta x_{\rm nt}$, $\nu := \nu + t\Delta \nu_{\rm nt}$.

until $Ax = b$ and $\|r(x, \nu)\|_2 \leq \epsilon$.

- not a descent method: $f(x^{(k+1)}) > f(x^{(k)})$ is possible
- directional derivative of $\|r(y)\|_2^2$ in direction $\Delta y = (\Delta x_{\rm nt}, \Delta \nu_{\rm nt})$ is

$$\frac{d}{dt} \|r(y + \Delta y)\|_2 \bigg|_{t=0} = -\|r(y)\|_2$$

Solving KKT systems

$$\begin{bmatrix} H & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} v \\ w \end{bmatrix} = - \begin{bmatrix} g \\ h \end{bmatrix}$$

solution methods

- LDL^T factorization
- elimination (if H nonsingular)

$$AH^{-1}A^Tw = h - AH^{-1}g, \qquad Hv = -(g + A^Tw)$$

- elimination with singular H: write as

$$\begin{bmatrix} H + A^TQA & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} v \\ w \end{bmatrix} = - \begin{bmatrix} g + A^TQh \\ h \end{bmatrix}$$

with $Q \succeq 0$ for which $H + A^TQA \succ 0$, and apply elimination

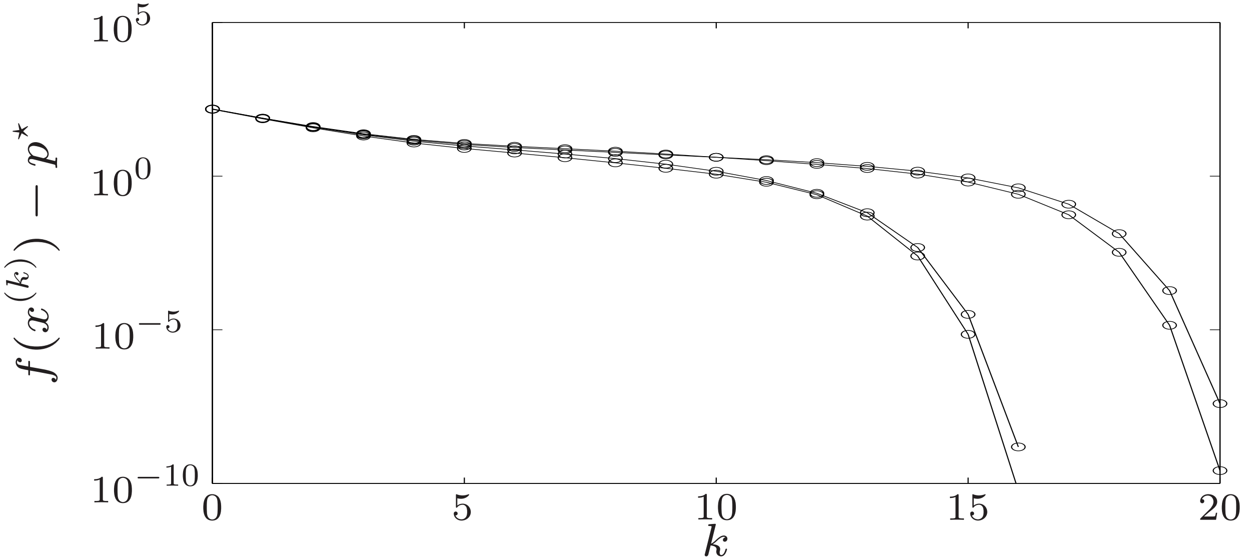
Equality constrained analytic centering

primal problem: minimize $-\sum_{i=1}^{n} \log x_i$ subject to $Ax = b$

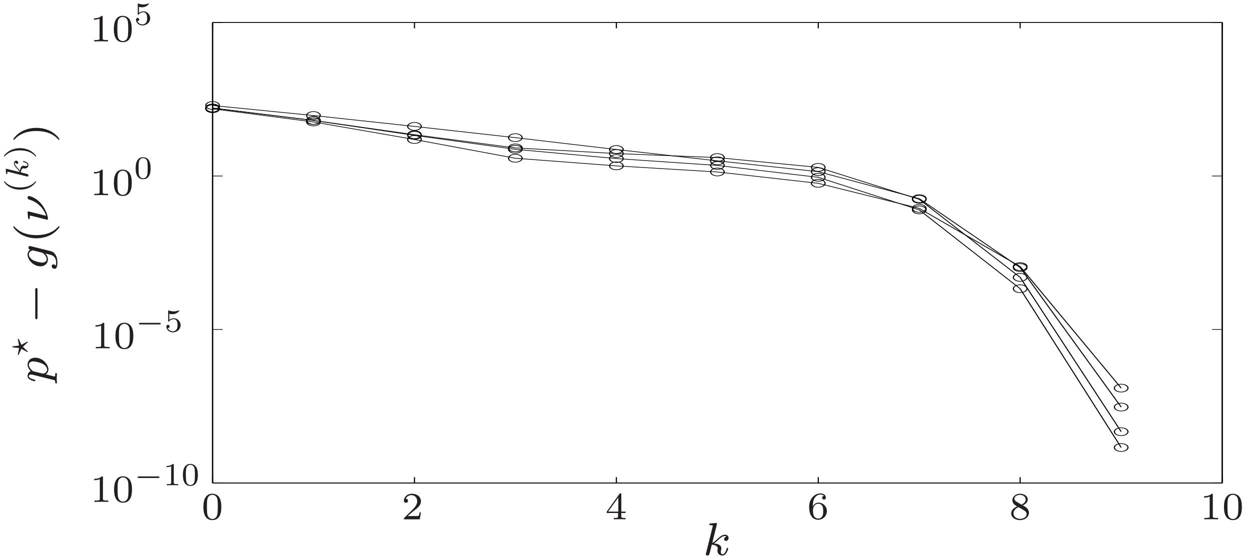
dual problem: maximize $-b^T\nu + \sum_{i=1}^{n} \log(A^T\nu)_i + n$

three methods for an example with $A \in \mathbf{R}^{100\times 500}$, different starting points

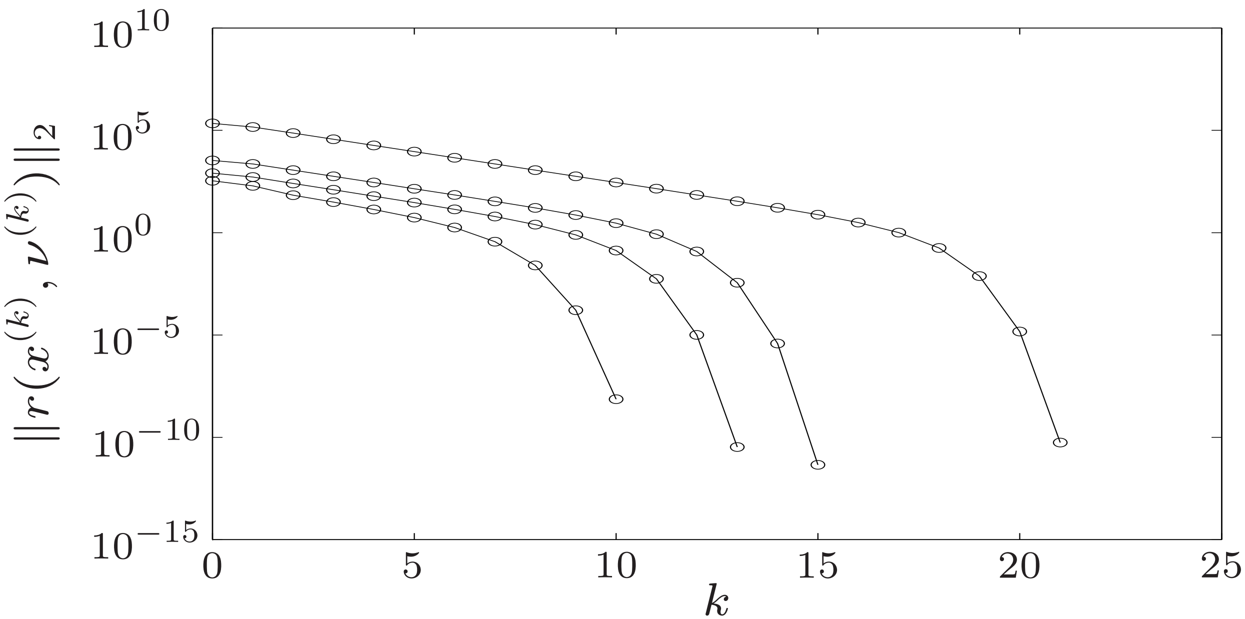
1. Newton method with equality constraints (requires $x^{(0)} \succ 0$, $Ax^{(0)} = b$)



2. Newton method applied to dual problem (requires $A^T\nu^{(0)} \succ 0$)



3. infeasible start Newton method (requires $x^{(0)} \succ 0$)



complexity per iteration of three methods is identical

1. use block elimination to solve KKT system

$$\begin{bmatrix} \mathbf{diag}(x)^{-2} & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} \Delta x \\ w \end{bmatrix} = \begin{bmatrix} \mathbf{diag}(x)^{-1}\mathbf{1} \\ 0 \end{bmatrix}$$

   reduces to solving $A\,\mathbf{diag}(x)^2 A^T w = b$

2. solve Newton system $A\,\mathbf{diag}(A^T\nu)^{-2}A^T\Delta\nu = -b + A\,\mathbf{diag}(A^T\nu)^{-1}\mathbf{1}$

3. use block elimination to solve KKT system

$$\begin{bmatrix} \mathbf{diag}(x)^{-2} & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta\nu \end{bmatrix} = \begin{bmatrix} \mathbf{diag}(x)^{-1}\mathbf{1} \\ Ax - b \end{bmatrix}$$

   reduces to solving $A\,\mathbf{diag}(x)^2 A^T w = 2Ax - b$

conclusion: in each case, solve $ADA^T w = h$ with D positive diagonal

Network flow optimization

$$\begin{array}{ll} \text{minimize} & \sum_{i=1}^n \phi_i(x_i) \\ \text{subject to} & Ax = b \end{array}$$

- directed graph with n arcs, $p+1$ nodes
- x_i: flow through arc i; ϕ_i: cost flow function for arc i (with $\phi_i''(x) > 0$)
- node-incidence matrix $\tilde{A} \in \mathbf{R}^{(p+1)\times n}$ defined as

$$\tilde{A}_{ij} = \begin{cases} 1 & \text{arc } j \text{ leaves node } i \\ -1 & \text{arc } j \text{ enters node } i \\ 0 & \text{otherwise} \end{cases}$$

- reduced node-incidence matrix $A \in \mathbf{R}^{p\times n}$ is $\tilde{A}$ with last row removed
- $b \in \mathbf{R}^p$ is (reduced) source vector
- $\mathbf{Rank}\, A = p$ if graph is connected

KKT system

$$\begin{bmatrix} H & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} v \\ w \end{bmatrix} = - \begin{bmatrix} g \\ h \end{bmatrix}$$

- $H = \mathbf{diag}(\phi_1''(x_1), \ldots, \phi_n''(x_n))$, positive diagonal
- solve via elimination:

$$AH^{-1}A^T w = h - AH^{-1}g, \qquad Hv = -(g + A^T w)$$

sparsity pattern of coefficient matrix is given by graph connectivity

$$\begin{aligned} (AH^{-1}A^T)_{ij} \neq 0 \quad &\Longleftrightarrow \quad (AA^T)_{ij} \neq 0 \\ &\Longleftrightarrow \quad \text{nodes } i \text{ and } j \text{ are connected by an arc} \end{aligned}$$

The real deal: General Convex Problems

- inequality constrained minimization
- logarithmic barrier function and central path
- barrier method
- feasibility and phase I methods
- complexity analysis via self-concordance
- generalized inequalities

Inequality constrained minimization

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

- f_i convex, twice continuously differentiable
- $A \in \mathbf{R}^{p \times n}$ with $\mathbf{Rank}\, A = p$
- we assume $p^\star$ is finite and attained
- we assume problem is strictly feasible: there exists $\tilde{x}$ with

$$\tilde{x} \in \mathbf{dom}\, f_0, \qquad f_i(\tilde{x}) < 0, \quad i = 1, \ldots, m, \qquad A\tilde{x} = b$$

hence, strong duality holds and dual optimum is attained

Examples

- LP, QP, QCQP, GP
- entropy maximization with linear inequality constraints

$$\begin{array}{ll} \text{minimize} & \sum_{i=1}^n x_i \log x_i \\ \text{subject to} & Fx \preceq g \\ & Ax = b \end{array}$$

  with $\mathbf{dom}\, f_0 = \mathbf{R}^n_{++}$

- differentiability may require reformulating the problem, *e.g.*, piecewise-linear minimization or ℓ_∞-norm approximation via LP

Logarithmic barrier

reformulation of (1) via indicator function:

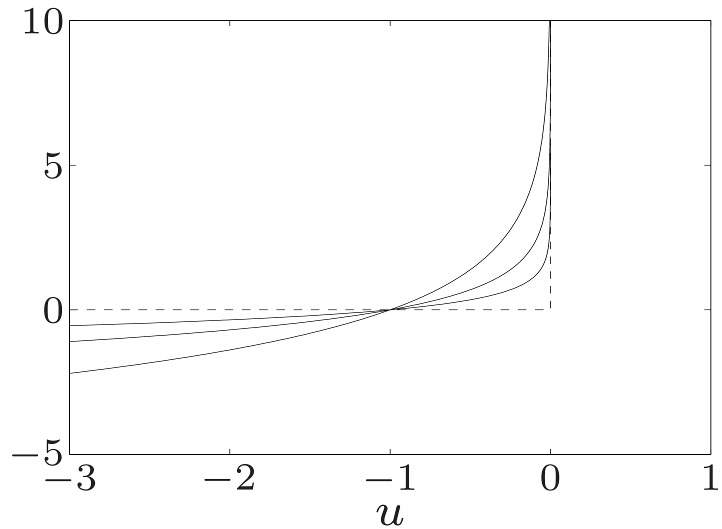
$$\begin{array}{ll} \text{minimize} & f_0(x) + \sum_{i=1}^m I_-(f_i(x)) \\ \text{subject to} & Ax = b \end{array}$$

where $I_-(u) = 0$ if $u \leq 0$, $I_-(u) = \infty$ otherwise (indicator function of $\mathbf{R}_-$)

approximation via logarithmic barrier

$$\begin{array}{ll} \text{minimize} & f_0(x) - (1/t)\sum_{i=1}^m \log(-f_i(x)) \\ \text{subject to} & Ax = b \end{array}$$

- an equality constrained problem
- for $t > 0$, $-(1/t)\log(-u)$ is a smooth approximation of I_-
- approximation improves as $t \to \infty$



logarithmic barrier function

$$\phi(x) = -\sum_{i=1}^{m} \log(-f_i(x)), \quad \mathbf{dom}\, \phi = \{x \mid f_1(x) < 0, \ldots, f_m(x) < 0\}$$

- convex (follows from composition rules)
- twice continuously differentiable, with derivatives

$$\begin{aligned} \nabla \phi(x) &= \sum_{i=1}^{m} \frac{1}{-f_i(x)} \nabla f_i(x) \\ \nabla^2 \phi(x) &= \sum_{i=1}^{m} \frac{1}{f_i(x)^2} \nabla f_i(x) \nabla f_i(x)^T + \sum_{i=1}^{m} \frac{1}{-f_i(x)} \nabla^2 f_i(x) \end{aligned}$$

Central path

- for $t > 0$, define $x^\star(t)$ as the solution of

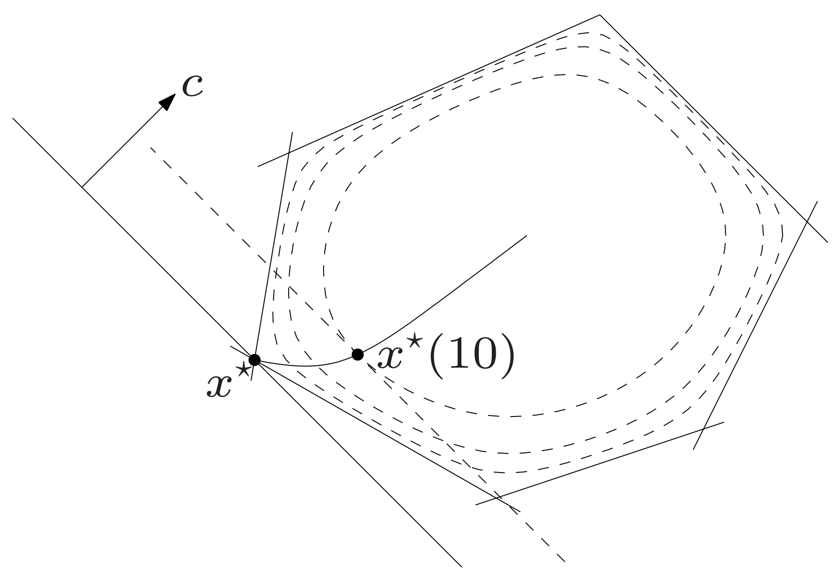
$$\begin{array}{ll} \text{minimize} & tf_0(x) + \phi(x) \\ \text{subject to} & Ax = b \end{array}$$

  (for now, assume $x^\star(t)$ exists and is unique for each $t > 0$)

- central path is $\{x^\star(t) \mid t > 0\}$

example: central path for an LP

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & a_i^T x \le b_i, \quad i = 1, \ldots, 6 \end{array}$$



hyperplane $c^T x = c^T x^\star(t)$ is tangent to level curve of ϕ through $x^\star(t)$

Dual points on central path

$x = x^\star(t)$ if there exists a w such that

$$t\nabla f_0(x) + \sum_{i=1}^m \frac{1}{-f_i(x)} \nabla f_i(x) + A^T w = 0, \qquad Ax = b$$

- therefore, $x^\star(t)$ minimizes the Lagrangian

$$L(x, \lambda^\star(t), \nu^\star(t)) = f_0(x) + \sum_{i=1}^m \lambda_i^\star(t) f_i(x) + \nu^\star(t)^T (Ax - b)$$

  where we define $\lambda_i^\star(t) = 1/(-t f_i(x^\star(t))$ and $\nu^\star(t) = w/t$

- this confirms the intuitive idea that $f_0(x^\star(t)) \to p^\star$ if $t \to \infty$:

$$\begin{aligned} p^\star &\geq g(\lambda^\star(t), \nu^\star(t)) \\ &= L(x^\star(t), \lambda^\star(t), \nu^\star(t)) \\ &= f_0(x^\star(t)) - m/t \end{aligned}$$

Interpretation via KKT conditions

$x = x^\star(t)$, $\lambda = \lambda^\star(t)$, $\nu = \nu^\star(t)$ satisfy

1. primal constraints: $f_i(x) \leq 0$, $i = 1, \ldots, m$, $Ax = b$
2. dual constraints: $\lambda \succeq 0$
3. approximate complementary slackness: $-\lambda_i f_i(x) = 1/t$, $i = 1, \ldots, m$
4. gradient of Lagrangian with respect to x vanishes:

$$\nabla f_0(x) + \sum_{i=1}^{m} \lambda_i \nabla f_i(x) + A^T \nu = 0$$

difference with KKT is that condition 3 replaces $\lambda_i f_i(x) = 0$

Force field interpretation

centering problem (for problem with no equality constraints)

$$\text{minimize} \quad tf_0(x) - \textstyle\sum_{i=1}^m \log(-f_i(x))$$

force field interpretation

- $tf_0(x)$ is potential of force field $F_0(x) = -t\nabla f_0(x)$
- $-\log(-f_i(x))$ is potential of force field $F_i(x) = (1/f_i(x))\nabla f_i(x)$

the forces balance at $x^\star(t)$:

$$F_0(x^\star(t)) + \sum_{i=1}^m F_i(x^\star(t)) = 0$$

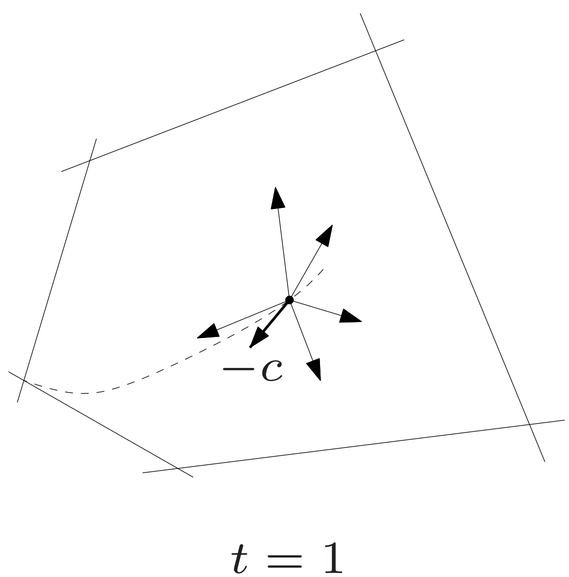
example

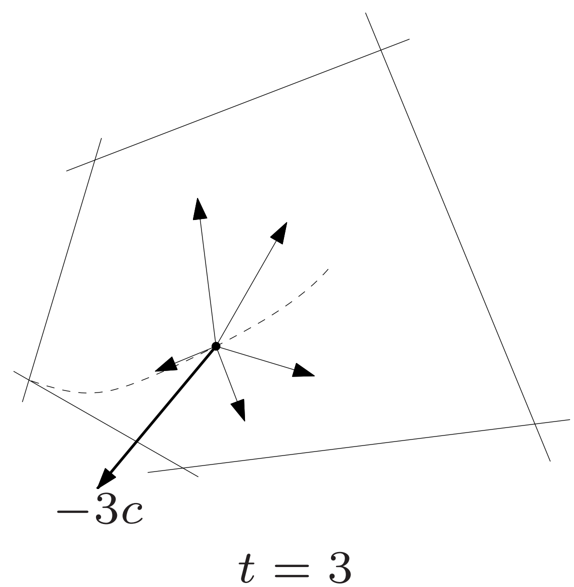
$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & a_i^T x \leq b_i, \quad i = 1, \ldots, m \end{array}$$

- objective force field is constant: $F_0(x) = -tc$
- constraint force field decays as inverse distance to constraint hyperplane:

$$F_i(x) = \frac{-a_i}{b_i - a_i^T x}, \qquad \|F_i(x)\|_2 = \frac{1}{\mathbf{dist}(x, \mathcal{H}_i)}$$

where $\mathcal{H}_i = \{x \mid a_i^T x = b_i\}$





Barrier method

given strictly feasible x, $t := t^{(0)} > 0$, $\mu > 1$, tolerance $\epsilon > 0$.

repeat

1. *Centering step.* Compute $x^\star(t)$ by minimizing $tf_0 + \phi$, subject to $Ax = b$.
2. *Update.* $x := x^\star(t)$.
3. *Stopping criterion.* **quit** if $m/t < \epsilon$.
4. *Increase* t*.* $t := \mu t$.

- terminates with $f_0(x) - p^\star \leq \epsilon$ (stopping criterion follows from $f_0(x^\star(t)) - p^\star \leq m/t$)
- centering usually done using Newton's method, starting at current x
- choice of μ involves a trade-off: large μ means fewer outer iterations, more inner (Newton) iterations; typical values: $\mu = 10$–20
- several heuristics for choice of $t^{(0)}$

Convergence analysis

number of outer (centering) iterations: exactly

$$\left\lceil \frac{\log(m/(\epsilon t^{(0)}))}{\log \mu} \right\rceil$$

plus the initial centering step (to compute $x^\star(t^{(0)})$)

centering problem

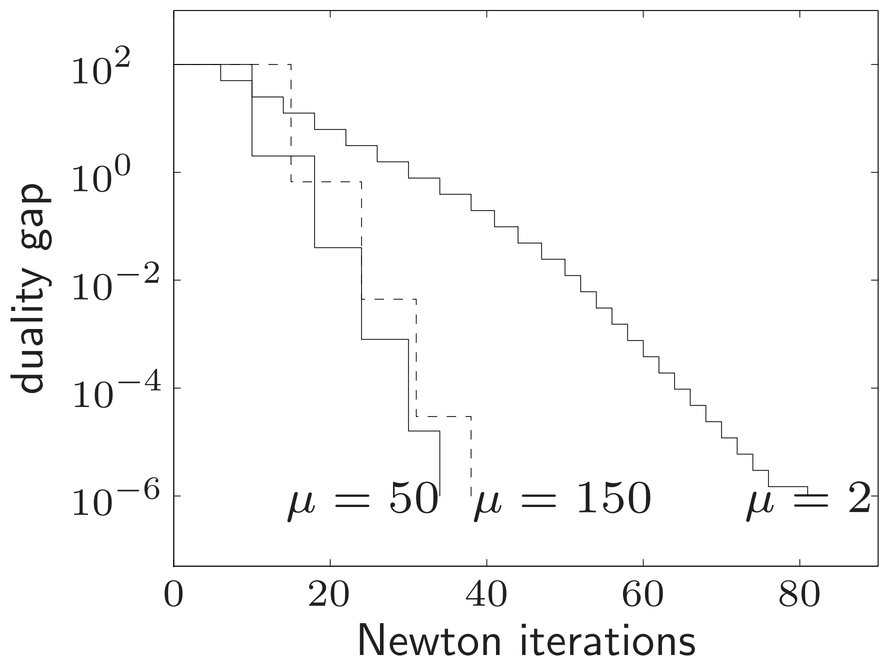
$$\text{minimize} \quad t f_0(x) + \phi(x)$$

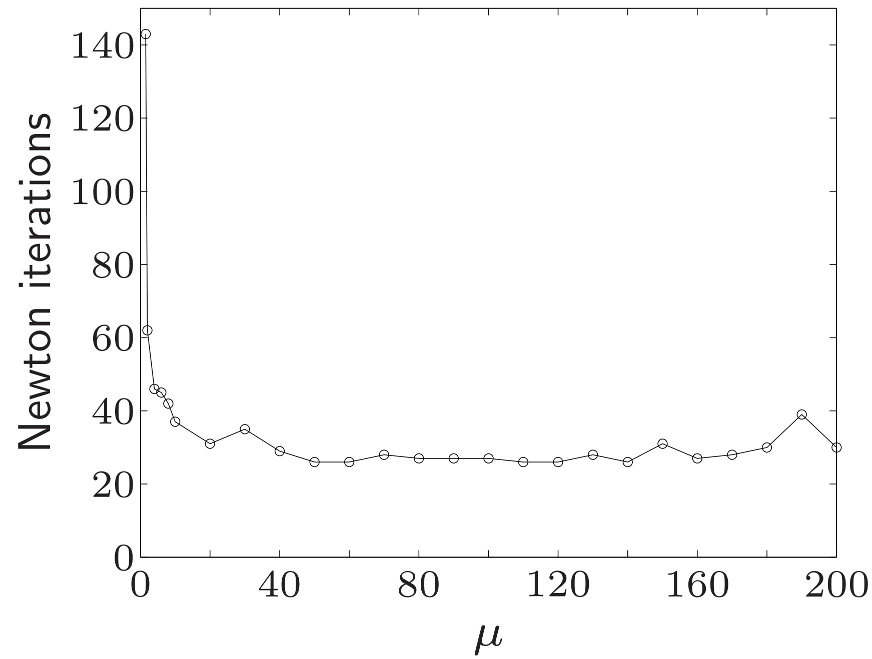
see convergence analysis of Newton's method

- $t f_0 + \phi$ must have closed sublevel sets for $t \geq t^{(0)}$
- classical analysis requires strong convexity, Lipschitz condition
- analysis via self-concordance requires self-concordance of $t f_0 + \phi$

Examples

inequality form LP ($m = 100$ inequalities, $n = 50$ variables)

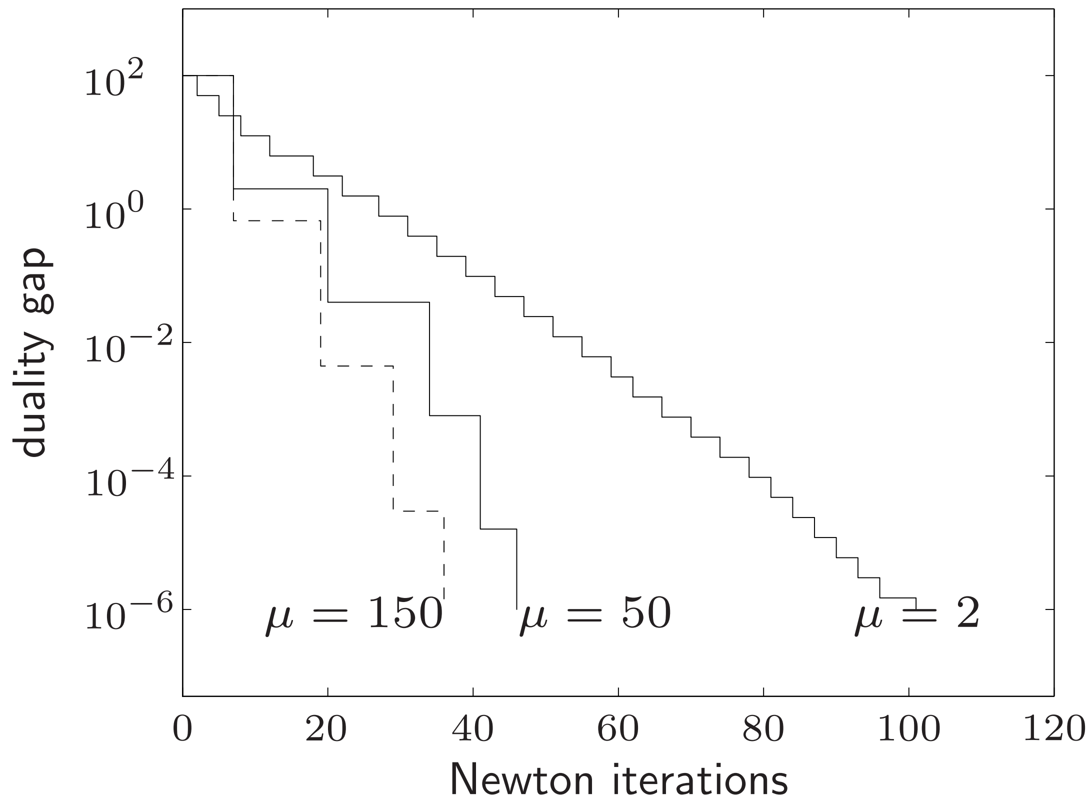




- starts with x on central path ($t^{(0)} = 1$, duality gap 100)
- terminates when $t = 10^8$ (gap 10^{-6})
- centering uses Newton's method with backtracking
- total number of Newton iterations not very sensitive for $\mu \geq 10$

geometric program ($m = 100$ inequalities and $n = 50$ variables)

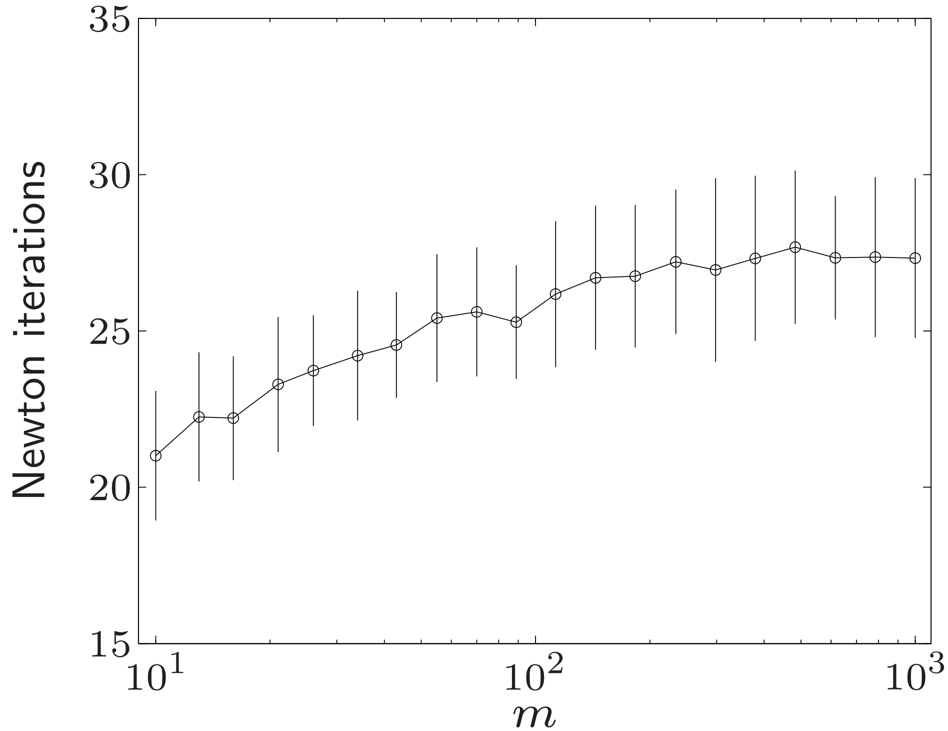
$$\begin{array}{ll} \text{minimize} & \log\left(\sum_{k=1}^{5} \exp(a_{0k}^T x + b_{0k})\right) \\ \text{subject to} & \log\left(\sum_{k=1}^{5} \exp(a_{ik}^T x + b_{ik})\right) \leq 0, \quad i = 1, \ldots, m \end{array}$$



family of standard LPs ($A \in \mathbf{R}^{m \times 2m}$)

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b, \quad x \succeq 0 \end{array}$$

$m = 10, \ldots, 1000$; for each m, solve 100 randomly generated instances



number of iterations grows very slowly as m ranges over a $100:1$ ratio

Feasibility and phase I methods

feasibility problem: find x such that

$$f_i(x) \leq 0, \quad i = 1, \ldots, m, \qquad Ax = b \tag{2}$$

phase I: computes strictly feasible starting point for barrier method

basic phase I method

$$\begin{array}{ll} \text{minimize (over } x, \, s) & s \\ \text{subject to} & f_i(x) \leq s, \quad i = 1, \ldots, m \\ & Ax = b \end{array} \tag{3}$$

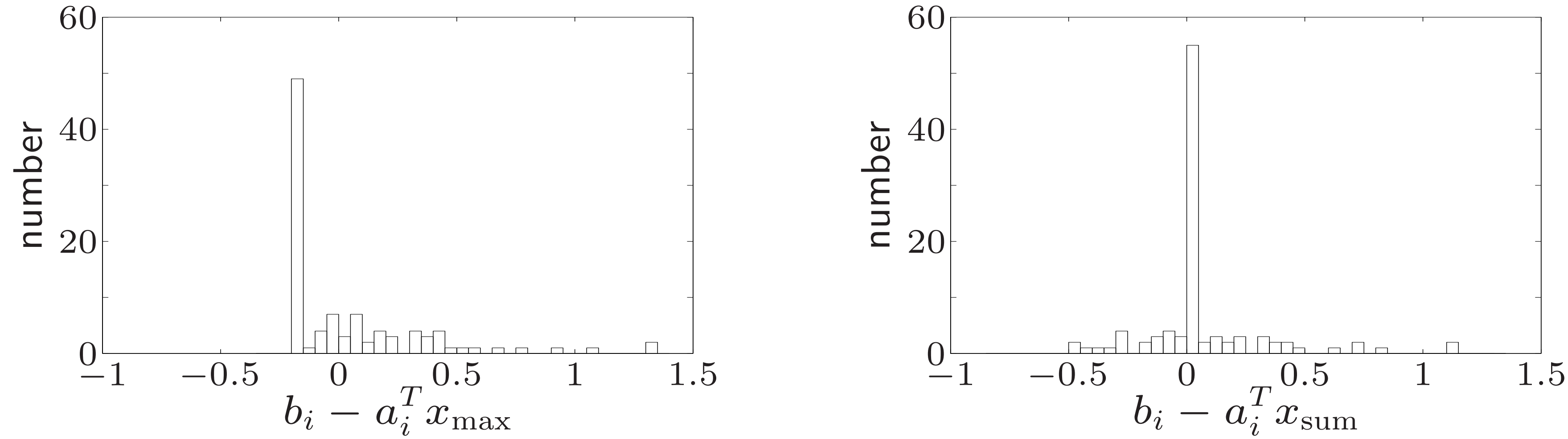
- if x, s feasible, with $s < 0$, then x is strictly feasible for (2)
- if optimal value $\bar{p}^\star$ of (3) is positive, then problem (2) is infeasible
- if $\bar{p}^\star = 0$ and attained, then problem (2) is feasible (but not strictly); if $\bar{p}^\star = 0$ and not attained, then problem (2) is infeasible

sum of infeasibilities phase I method

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

for infeasible problems, produces a solution that satisfies many more inequalities than basic phase I method

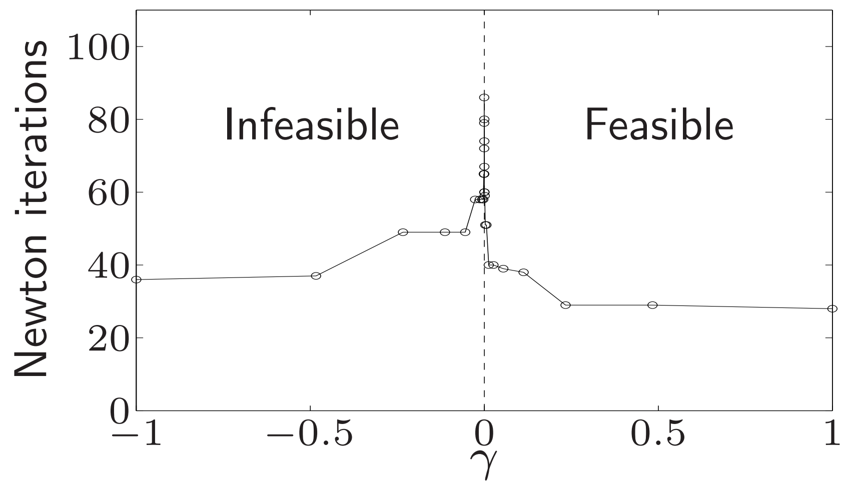
example (infeasible set of 100 linear inequalities in 50 variables)

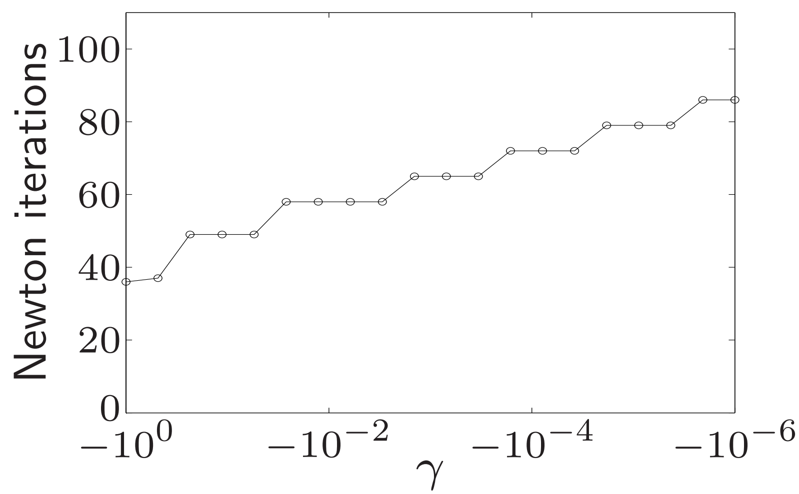


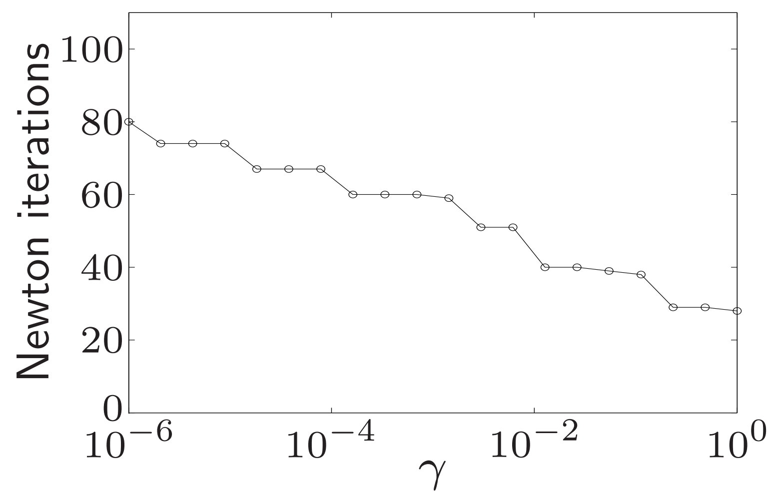
left: basic phase I solution; satisfies 39 inequalities
right: sum of infeasibilities phase I solution; satisfies 79 solutions

example: family of linear inequalities $Ax \preceq b + \gamma\Delta b$

- data chosen to be strictly feasible for $\gamma > 0$, infeasible for $\gamma \le 0$
- use basic phase I, terminate when $s < 0$ or dual objective is positive







number of iterations roughly proportional to $\log(1/|\gamma|)$

Complexity analysis via self-concordance

same assumptions as on page 135, plus:

- sublevel sets (of f_0, on the feasible set) are bounded
- $tf_0 + \phi$ is self-concordant with closed sublevel sets

second condition

- holds for LP, QP, QCQP
- may require reformulating the problem, *e.g.*,

$$\begin{array}{ll} \text{minimize} & \sum_{i=1}^n x_i \log x_i \\ \text{subject to} & Fx \preceq g \end{array} \quad \longrightarrow \quad \begin{array}{ll} \text{minimize} & \sum_{i=1}^n x_i \log x_i \\ \text{subject to} & Fx \preceq g, \quad x \succeq 0 \end{array}$$

- needed for complexity analysis; barrier method works even when self-concordance assumption does not apply

Newton iterations per centering step: from self-concordance theory

$$\#\text{Newton iterations} \leq \frac{\mu t f_0(x) + \phi(x) - \mu t f_0(x^+) - \phi(x^+)}{\gamma} + c$$

- bound on effort of computing $x^+ = x^\star(\mu t)$ starting at $x = x^\star(t)$
- γ, c are constants (depend only on Newton algorithm parameters)
- from duality (with $\lambda = \lambda^\star(t)$, $\nu = \nu^\star(t)$):

$$\begin{aligned}
&\mu t f_0(x) + \phi(x) - \mu t f_0(x^+) - \phi(x^+) \\
&\quad = \mu t f_0(x) - \mu t f_0(x^+) + \sum_{i=1}^m \log(-\mu t \lambda_i f_i(x^+)) - m \log \mu \\
&\quad \leq \mu t f_0(x) - \mu t f_0(x^+) - \mu t \sum_{i=1}^m \lambda_i f_i(x^+) - m - m \log \mu \\
&\quad \leq \mu t f_0(x) - \mu t g(\lambda, \nu) - m - m \log \mu \\
&\quad = m(\mu - 1 - \log \mu)
\end{aligned}$$

total number of Newton iterations (excluding first centering step)

$$\#\text{Newton iterations} \leq N = \left\lceil \frac{\log(m/(t^{(0)}\epsilon))}{\log \mu} \right\rceil \left(\frac{m(\mu - 1 - \log \mu)}{\gamma} + c \right)$$

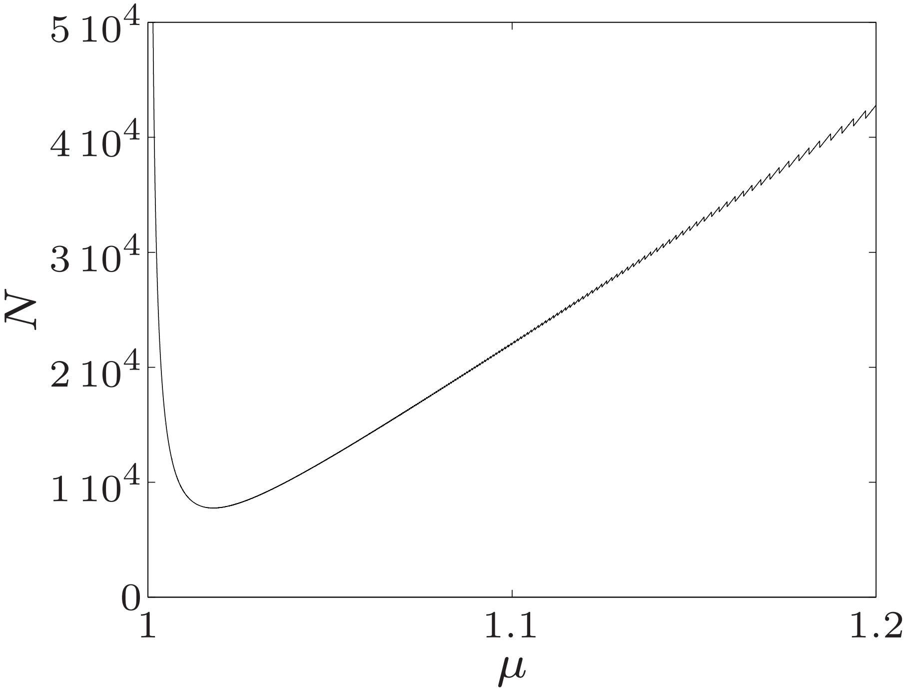


figure shows N for typical values of γ, c,

$$m = 100, \qquad \frac{m}{t^{(0)}\epsilon} = 10^5$$

- confirms trade-off in choice of μ
- in practice, #iterations is in the tens; not very sensitive for $\mu \geq 10$

polynomial-time complexity of barrier method

- for $\mu = 1 + 1/\sqrt{m}$:

$$N = O\left(\sqrt{m}\log\left(\frac{m/t^{(0)}}{\epsilon}\right)\right)$$

- number of Newton iterations for fixed gap reduction is $O(\sqrt{m})$
- multiply with cost of one Newton iteration (a polynomial function of problem dimensions), to get bound on number of flops

this choice of μ optimizes worst-case complexity; in practice we choose μ fixed ($\mu = 10, \ldots, 20$)

Barrier method

given strictly feasible x, $t := t^{(0)} > 0$, $\mu > 1$, tolerance $\epsilon > 0$.

repeat

1. *Centering step.* Compute $x^\star(t)$ by minimizing $tf_0 + \phi$, subject to $Ax = b$.
2. *Update.* $x := x^\star(t)$.
3. *Stopping criterion.* **quit** if $(\sum_i \theta_i)/t < \epsilon$.
4. *Increase* t. $t := \mu t$.

- only difference is duality gap m/t on central path is replaced by $\sum_i \theta_i/t$
- number of outer iterations:

$$\left\lceil \frac{\log((\sum_i \theta_i)/(\epsilon t^{(0)}))}{\log \mu} \right\rceil$$