Optimization Algorithms

Zhouwang Yang

University of Science and Technology of China

2019-02-25

Outline

Description

- The course is devoted to the mathematical fundamentals of optimization and the practical algorithms of optimization.
- The course covers the topics of nonlinear continuous optimization, sparse optimization, stochastic optimization, combinatorial optimization, and global optimization.

Objectives

Objectives of the course are

- to develop an understanding of the fundamentals of optimization;
- to learn how to analyze the widely used algorithms for optimization;
- to become familiar with the implementation of optimization algorithms.

Prerequisites

- Knowledge of Linear Algebra, Real Analysis, and Mathematics of Operations Research will be important.
- Simultaneously, the ability to write computer programs of algorithms is also required.

Topics Covered

- Unconstrained Optimization
- Constrained Optimization
- Convex Optimization
- Sparse Optimization
- Stochastic Optimization
- Combinatorial Optimization
- Global Optimization

Textbook and References

1. R. Fletcher. Practical Methods of Optimization (2nd Edition), John Wiley & Sons, 1987.
2. J. Nocedal and S. J. Wright. Numerical Optimization (2nd Edition), Springer, 2006.
3. S. Boyd and L. Vandenberghe. Convex Optimization, Cambridge University Press, 2004.
4. M. Elad. Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing. Springer, 2010.
5. A. Shapiro, D. Dentcheva and A. Ruszczynski. Lectures on Stochastic Programming: Modeling and Theory, SIAM, Philadelphia, 2009.
6. T. Weise. Global Optimization Algorithms - Theory and Application, 2009. [http://www.it-weise.de/]

Grading

(1) Homework (20%)

(2) Project (30%)

(3) Final Exam (50%)

Outline

1. Convex Optimization
   - Convex Set and Convex Function
   - Convex Optimization and Algorithms

2. Sparse Optimization
   - Compressed Sensing
   - Sparse Modeling
   - Sparse Optimization Algorithms

About convex optimization

Convex optimization is a subfield of mathematical optimization that studies the problem of minimizing convex functions over convex sets. Whereas many classes of convex optimization problems admit polynomial-time algorithms, mathematical optimization is in general NP-hard.

We introduce the main definitions and results of convex optimization needed for the analysis of algorithms presented in the section.

Outline

Affine set

Definition (affine set)

A set $C \subseteq \mathbb{R}^n$ is *affine* if $\forall x_1, x_2 \in C$ and $\theta \in \mathbb{R}$, we have

$$\theta x_1 + (1-\theta)x_2 \in C$$

i.e., if it contains the line through any two distinct points in it.

It can be generalized to more than two points: If C is an affine set, $x_1, \ldots, x_k \in C$ and $\theta_1 + \ldots + \theta_k = 1$, then $\theta_1 x_1 + \ldots + \theta_k x_k \in C$.

We refer to a point of the form $\theta_1 x_1 + \ldots + \theta_k x_k$ where $\theta_1 + \ldots + \theta_k = 1$, as an *affine combination* of the points $x_1, \ldots, x_k$.

Affine set

If C is an affine set and $x_0 \in C$, then the set

$$V = C - x_0 = \{x - x_0 | x \in C\}$$

is a (linear) subspace. We can express C as

$$C = V + x_0 = \{v + x_0 | v \in V\}.$$

The *dimesion* of an affine set C is the dimesion of the subspace $V = C - x_0$.

Example (Solution set of linear equations)

For $A \in \mathbb{R}^{m \times n}, b \in \mathbb{R}^m$, the set

$$C = \{x | Ax = b\}$$

is affine. Let $V = \{v | Av = 0\}$ be a subspace, then $C = V + b$.

Affine set

Definition (affine hull)

The set of all affine combinations of points in some set $C \subseteq \mathbb{R}^n$ is called the *affine hull* of C, denoted $\mathbf{aff}C$:

$$\mathbf{aff}C = \{\theta_1 x_1 + \ldots + \theta_k x_k | x_1, \ldots, x_k \in C, \theta_1 + \ldots + \theta_k = 1\}.$$

The affine hull is the smallest affine set that contains C.

Convex set

Definition (convex set)

A set C is *convex* if $\forall x_1, x_2 \in C$ and $0 \leqslant \theta \leqslant 1$, we have

$$\theta x_1 + (1-\theta)x_2 \in C$$

i.e., if it contains the line segment between any two points in it.

Generalization to more than two points: for any $k \geqslant 1$, $x_1, \ldots, x_k \in C$ and $\theta_1 + \ldots + \theta_k = 1$ where $\theta_i \geqslant 0$, $i = 1, \ldots, k$, we have

$$\theta_1 x_1 + \ldots + \theta_k x_k \in C.$$

The form $\theta_1 x_1 + \ldots + \theta_k x_k$ is called the *convex combination* of the points $x_1, \ldots, x_k$, where $\theta_1, \ldots, \theta_k \geqslant 0$ and $\sum_{i=1}^{k} \theta_i = 1$.

Convex hull

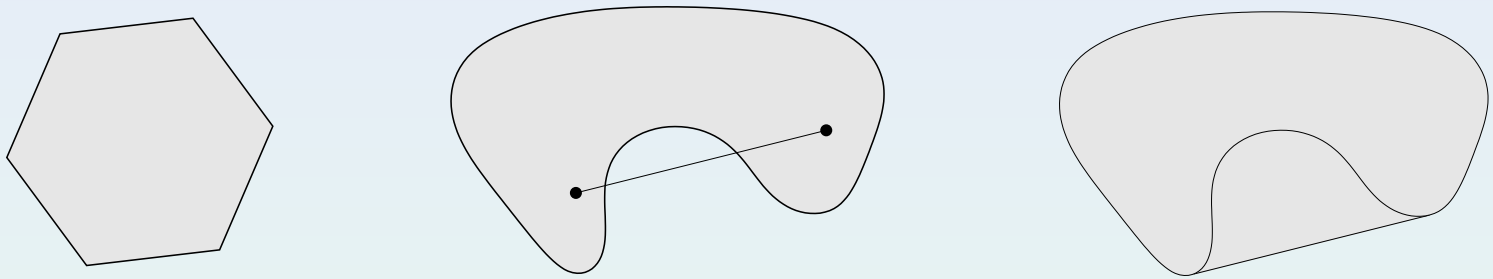
Definition (convex hull)

The *convex hull* of a set C, denoted $\mathbf{conv}C$, is the set of all convex combinations of points in C:

$$\mathbf{conv}C = \{\theta_1 x_1 + \ldots + \theta_k x_k | x_i \in C, \theta_i \geqslant 0, i = 1, \ldots, k, \theta_1 + \ldots + \theta_k = 1\}.$$

The convex hull is the smallest convex set that contains C.

Convex set and convex hull



(a) (b) (c)

Figure: **(a)** A convex set (polyhydron). **(b)** A non-convex set. **(c)** The convex hull of (b).

Cone

Definition (cone)

A set C is called a *cone*, if $\forall x \in C$ and $\theta \geqslant 0$ we have θx in C.
A set C is a convex cone if it's convex and a cone, i.e., $\forall x_1, x_2 \in C$ and $\theta_1, \theta_2 \geqslant 0$, we have

$$\theta_1 x_1 + \theta_2 x_2 \in C.$$

A point of the form $\theta_1 x_1 + \ldots + \theta_k x_k$ with $\theta_1, \ldots, \theta_k \geqslant 0$ is called a *conic combination* of $x_1, \ldots, x_k$.

Definition (conic hull)

The conic hull of a set C is the set of all conic combinations of points in C, i.e.,

$$\{\theta_1 x_1 + \ldots + \theta_k x_k | x_i \in C, \theta_i \geqslant 0, i = 1, \ldots, k\}.$$

Conic hull

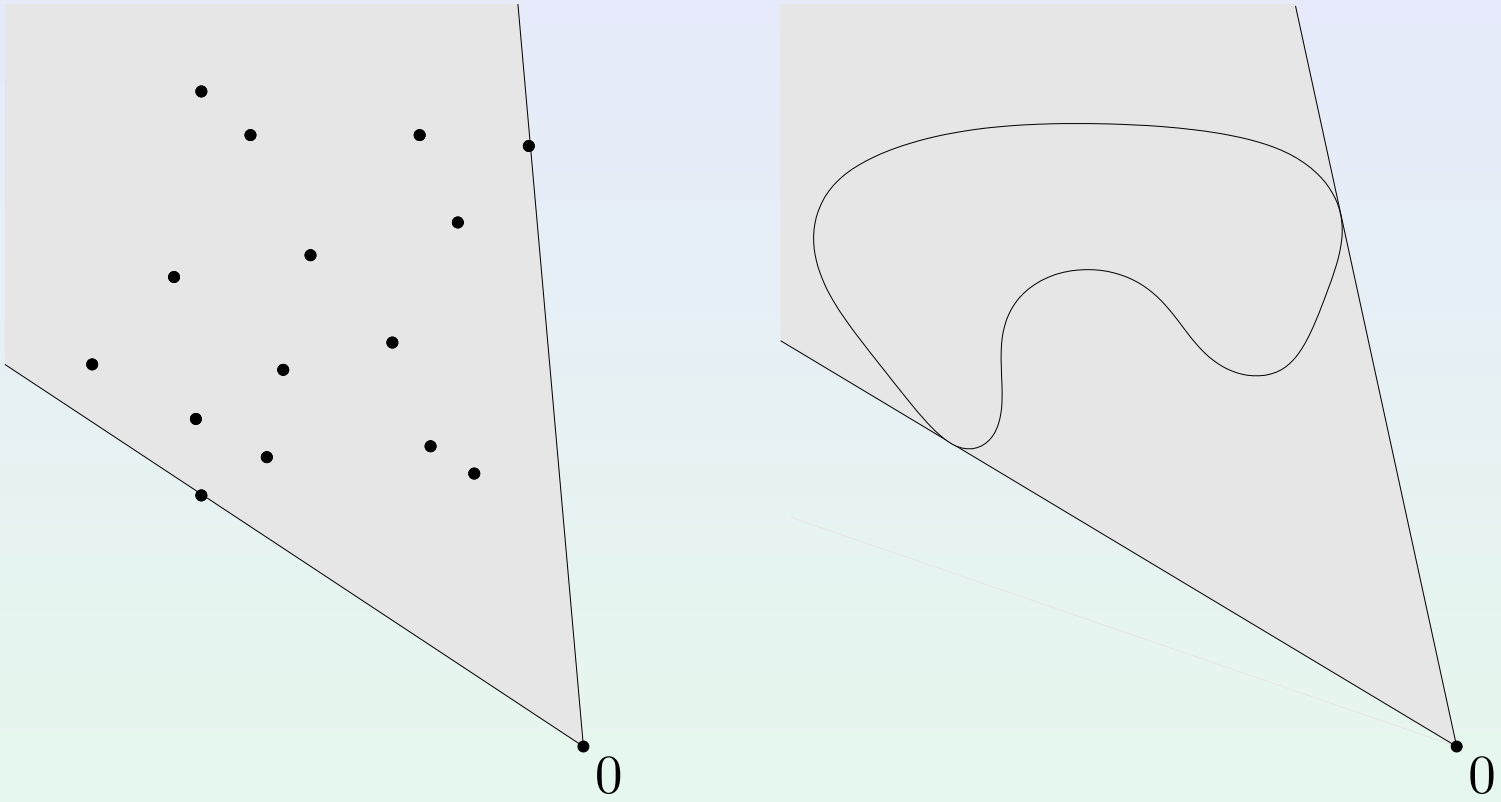


Figure: **Left.** The shaded set is the conic hull of a set of fifteen points (not including the origin). **Right.** The shaded set is the conic hull of the non-convex kidney-shaped set that is surrounded by a curve.

Some important convex examples

- *Hyperplane:* A hyperplane is a set of the form

$$\{x|a^\top x = b\}.$$

  It's also affine.

- *Halfspace:* A (closed) halfspace is a set of the form

$$\{x|a^\top x \leqslant b\}.$$

  A hyperplane divides $\mathbb{R}^n$ into two halfspaces.

Some important convex examples

- *Polyhedra:* A polyhedron is defined as the solution set of a finite number of linear equalities and inequalities:

$$\mathcal{P} = \{x | a_j^\top x \leq b_j, j = 1, \ldots, m, c_k^T x = d_k, k = 1, \ldots, p\}.$$

- *Ball:* A (Euclidean) ball in $\mathbb{R}^n$ has the form

$$B(x_c, r) = \{x | \|x - x_c\|_2 \leqslant r\}$$

where $r > 0$ and $\|u\|_2 = (u^\top u)^{1/2}$ denotes the Euclidean norm.

Some important convex examples

- *Norm balls* and *norm cones:*
  Suppose $\|\cdot\|$ is any norm on $\mathbb{R}^n$, a norm ball of radius r and center x_c is given by

$$\{x | \|x - x_c\| \leqslant r\}.$$

  The norm cone associated with the norm $\|\cdot\|$ is the set

$$C = \{(x, t) | \|x\| \leqslant t\} \subseteq \mathbb{R}^{n+1}.$$

  It's a convex cone.

Some important convex examples

- *The positive semidefinite cone:*
  The set of symmetric $n \times n$ matrices $\mathbf{S}^n$:

$$\mathbf{S}^n = \{X \in \mathbb{R}^{n \times n} | X = X^\top\},$$

  the set of symmetric positive semidefinite matrices $\mathbf{S}^n_+$:

$$\mathbf{S}^n_+ = \{X \in \mathbf{S}^n | X \succeq 0\},$$

  and the set of symmetric positive definite matrices $\mathbf{S}^n_{++}$:

$$\mathbf{S}^n_{++} = \{X \in \mathbf{S}^n | X \succ 0\}$$

  are all convex.

Proper cones and generalized inequalities

A cone $K \subseteq \mathbb{R}^n$ is called a *proper cone* if it satisfies the following:

- K is convex.
- K is closed.
- K is solid, which means it has nonempty interior.
- K is pointed, which means that it contains no line, i.e.,

$$x \in K \text{ and } -x \in K \;\Rightarrow\; x = 0.$$

A proper cone K can be used to define a *generalized inequality*:

$$x \preceq_K y \iff y - x \in K,$$

which is a partial ordering on $\mathbb{R}^n$. Similarly, we define an associated strict partial ordering by

$$x \prec_K y \iff y - x \in \mathbf{int} K$$

Properties of generalized inequalities

- If $x \preceq_K y$ and $u \preceq_K v$, then $x + u \preceq_K y + v$.
- If $x \preceq_K y$ and $y \preceq_K z$ then $x \preceq_K z$.
- If $x \preceq_K y$ and $\alpha \geqslant 0$ then $\alpha x \preceq_K \alpha y$.
- $x \preceq_K x$.
- If $x \preceq_K y$ and $y \preceq_K x$ then $x = y$.
- If $x_i \preceq_K y_i$ for $i = 1, 2, \ldots$, $x_i \to x$ and $y_i \to y$ as $i \to \infty$, then $x \preceq_K y$.

Minimum and minimal elements

- $x \in S$ is the *minimum* element of S (with respect to the generalized inequality $\preceq_K$) if for every $y \in S$ we have $x \preceq_K y$, i.e.,

$$S \subseteq x + K,$$

  where $x + K = \{x + y | y \in K\}$.

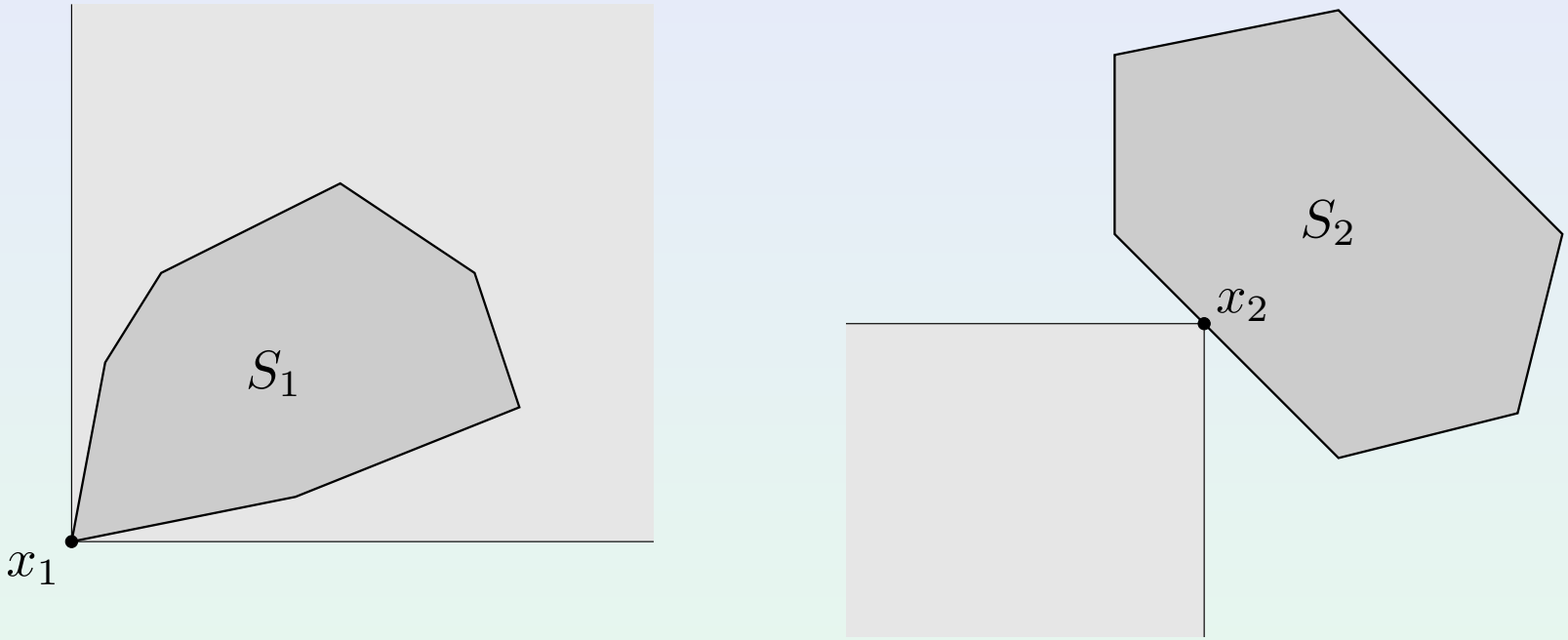
- $x \in S$ is a *minimal* element of S (with respect to the generalized inequality $\preceq_K$) if $y \in S, y \preceq_K x$ only if $y = x$, i.e.,

$$(x - K) \cap S = \{x\},$$

  where $x - K = \{x - y | y \in K\}$.

- Maximum element and maximal element are defined in a similar way.

Minimum and minimal elements



(a) (b)

Figure: Let $K = \{(u, v)|u, v \geqslant 0\}$. **(a)** x_1 is the minimum element of S_1. **(b)** x_2 is a minimal element of S_2.

Minimum and minimal elements

If x is the minimum element of S, then x must be a minimal element of S (with respect to the generalized inequality $\preceq_K$).

Brief proof: Suppose $S \subseteq x + K$, and $z \in (x - K) \cap S$, *i.e.*, $\exists y \in K$ such that $z = x - y$. By $z \in S \subseteq x + K$, there exists $w \in K$ such that $z = x + w$. Then we have $w = -y$, which leads to $-w = y \in K$ and $w \in K$. Since K is a proper cone, $w = 0$ and $z = x$.

But the reverse proposition doesn't hold.

Simple example: Let $K = \{(u, v) | u, v \geqslant 0\}$ and $L = \{(x, y) | x = -y\}$. Then every point of L is a minimal element, but none of them is the minimum element of L.

Definition

Definition (convex function)

A function $f : \mathbb{R}^n \to \mathbb{R}$ is *convex* if $\mathbf{dom} f$ is a convex set and if $\forall x, y \in \mathbf{dom} f$ and θ with $0 \leqslant \theta \leqslant 1$, we have

$$f(\theta x + (1-\theta)y) \leqslant \theta f(x) + (1-\theta) f(y). \tag{1}$$

A function is *strictly convex* if strict inequality holds in (1) whenever $x \neq y$ and $0 < \theta < 1$.

We say f is *concave* if $-f$ is convex, and *strictly concave* if $-f$ is strictly convex.

Definition

Geometrically, Eq.(1) means that the line segment between $(x, f(x))$ and $(y, f(y))$ lies above the graph of f (as shown in Fig.4).

Figure: Graph of a convex function.

First-order conditions

Suppose f is differentiable, i.e., its gradient ∇f exists at each point in $\mathbf{dom} f$.

Function f is convex if and only if $\mathbf{dom} f$ is convex and for $\forall x, y \in \mathbf{dom} f$, the following holds:

$$f(y) \geqslant f(x) + \nabla f(x)^{\top}(y - x).$$

Remark. As a simple result, if $\nabla f(x^*) = 0$, then for all $y \in \mathbf{dom} f$, $f(y) \geqslant f(x^*)$, *i.e.*, x^* is a global minimizer of the function f.

First-order conditions

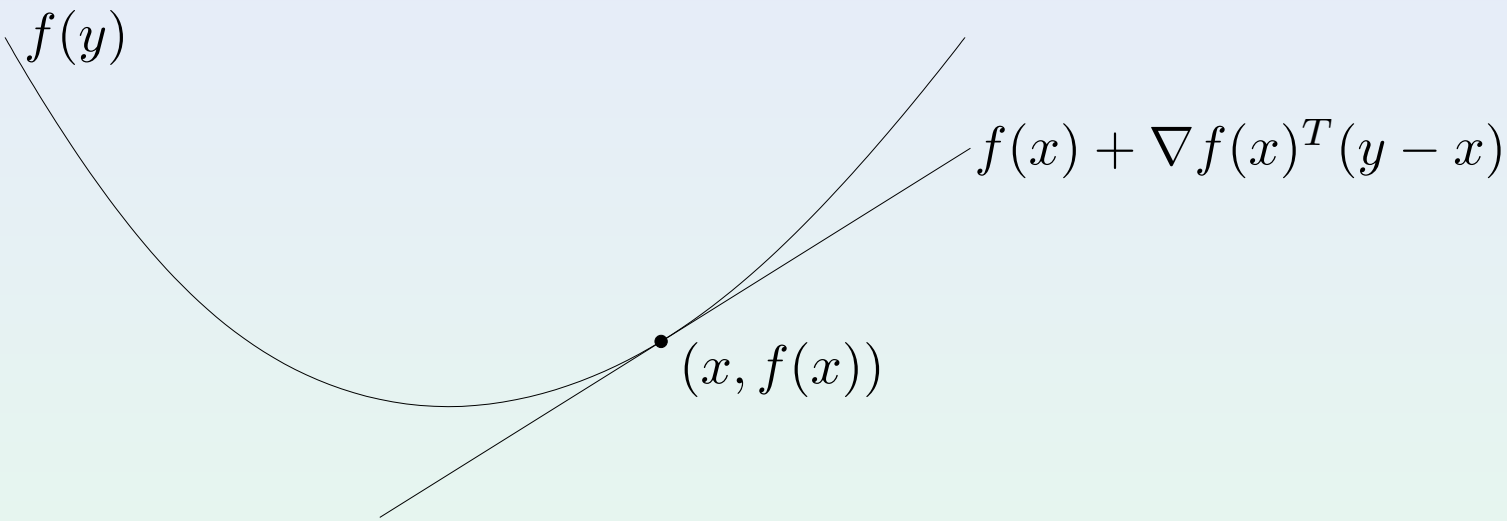


Figure: The tangent to a convex function.

First-order conditions

Function f is strictly convex if and only if $\mathbf{dom} f$ is convex and for $\forall x, y \in \mathbf{dom} f, x \neq y$, we have

$$f(y) > f(x) + \nabla f(x)^\top (y - x).$$

Correspondingly, f is concave if and only if $\mathbf{dom} f$ is convex and for $\forall x, y \in \mathbf{dom} f$, we have

$$f(y) \leqslant f(x) + \nabla f(x)^\top (y - x).$$

Second-order conditions

Assume that f is twice differentiable.

Function f is convex if and only if $\mathbf{dom} f$ is convex and for $\forall x \in \mathbf{dom} f$,

$$\nabla^2 f(x) \succeq 0.$$

Similarly, f is concave if and only if $\mathbf{dom} f$ is convex and $\nabla^2 f(x) \preceq 0$ for $\forall x \in \mathbf{dom} f$.

Second-order conditions

Strict convexity can be partially characterized by second-order conditions.

If $\nabla^2 f(x) \succ 0$ for $\forall x \in \mathbf{dom} f$, then f is strictly convex.

However, the converse is not true. For example, $f : \mathbb{R} \to \mathbb{R}$ given by $f(x) = x^4$ is strictly convex but has zero second derivative at $x = 0$.

Examples

- *Exponential:*
  e^{ax} is convex on $\mathbb{R}$, for any $a \in \mathbb{R}$.
- *Powers:*
  x^a is convex on $\mathbb{R}_{++}$ when $a \geqslant 1$ or $a \leqslant 0$, and concave for $0 \leqslant a \leqslant 1$.
- *Powers of absolute value:*
  $|x|^p$, for $p \geqslant 1$, is convex on $\mathbb{R}$.
- *Logarithm:*
  $\log x$ is concave on $\mathbb{R}_{++}$.
- *Negative entropy:*
  $x \log x$ is convex on $\mathbb{R}_+$, where $0 \log 0$ defined to be 0.

Examples

- *Norms:*
  Every norm on $\mathbb{R}^n$ is convex.

- *Max function:*
  $f(x) = \max\{x_1, \ldots, x_n\}$ is convex on $\mathbb{R}^n$.

- *Log-sum-exp:*
  Then function $f(x) = \log(e^{x_1} + \ldots + e^{x_n})$ is convex on $\mathbb{R}^n$. This function can be interpreted as a differentiable approximation of the max function, since for all x,

$$\max\{x_1, \ldots, x_n\} \leqslant f(x) \leqslant \max\{x_1, \ldots, x_n\} + \log n.$$

- *Geometric mean:*
  $f(x) = (\prod_{i=1}^n x_i)^{1/n}$ is concave on $\mathbf{dom} f = \mathbb{R}^n_{++}$.

- *Log-determinant:*
  $f(X) = \log\det X$ is concave on $\mathbf{dom} f = S^n_{++}$

Jensen's inequality

The inequality (1), *i.e.*, $f(\theta x + (1-\theta)y) \leqslant \theta f(x) + (1-\theta)f(y)$, is sometimes called *Jensen's inequality.*

It is easily extended to convex combinations of more than two points:

> If f is convex, $x_1, \ldots, x_k \in \mathbf{dom} f$, and $\theta_1, \ldots, \theta_k \geqslant 0$ with $\theta_1 + \ldots + \theta_k = 1$, then
>
> $$f(\theta_1 x_1 + \ldots + \theta_k x_k) \leqslant \theta_1 f(x_1) + \ldots + \theta_k f(x_k).$$

Operations that preserve convexity

- *Nonnegative weighted sums:*

  If $f_1, \ldots, f_m$ are convex and $w_1, \ldots, w_m \geqslant 0$, then

  $$f = w_1 f_1 + \ldots + w_m f_m$$

  is convex.

- These properties extend to infinite sums and integrals:

  If $f(x, y)$ is convex in x for each $y \in \mathcal{A}$, and $w(y) \geqslant 0$ for each $y \in \mathcal{A}$, then the function

  $$g(x) = \int_{\mathcal{A}} w(y) f(x, y) dy$$

  is convex in x (provided the integral exists).

Operations that preserve convexity

- *Composition with an affine mapping:*

  Suppose $f : \mathbb{R}^n \to \mathbb{R}$, $A \in \mathbb{R}^{n\times m}$, and $b \in \mathbb{R}^n$. Define $g : \mathbb{R}^m \to \mathbb{R}$ by

  $$g(x) = f(Ax + b),$$

  with $\mathbf{dom} g = \{x | Ax + b \in \mathbf{dom} f\}$. Then if f is convex, so is g; if f is concave, so is g.

Operations that preserve convexity

- *Pointwise maximum:*

  If f_1 and f_2 are convex functions, then

  $$f(x) = \max\{f_1(x), f_2(x)\},$$

  with $\mathbf{dom} f = \mathbf{dom} f_1 \cap \mathbf{dom} f_2$, is also convex.

- Extension to the pointwise supremum:

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

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

  is convex in x, where

  $$\mathbf{dom} g = \{x | (x, y) \in \mathbf{dom} f \text{ for all } y \in \mathcal{A}, \sup_{y \in \mathcal{A}} f(x, y) < \infty\}.$$

Functions closed to convex functions

- *Quasi-convex function*: A function $f : \mathbb{R}^n \to \mathbb{R}$ such that its domain and all its sublevel sets

$$S_\alpha = \{x \in \mathbf{dom} f \,|\, f(x) \leqslant \alpha\}, \quad \alpha \in \mathbb{R}$$

  are convex.

- *Log-concave function*: A function $f : \mathbb{R}^n \to \mathbb{R}$ such that $f(x) > 0, \forall x \in \mathbf{dom} f$ and $\log f$ is concave.

Outline

Basic terminology

$$\begin{array}{ll} \min & f_0(x) \\ \text{s.t.} & f_i(x) \leqslant 0, \quad i = 1, \ldots, m \\ & h_j(x) = 0, \quad j = 1, \ldots, p \end{array} \tag{2}$$

$x \in \mathbb{R}^n$	the *optimization variable*
$f_0 : \mathbb{R}^n \to \mathbb{R}$	the *objective function* or *cost function*
$f_i(x) \leqslant 0$	the *inequality constraints*
$f_i : \mathbb{R}^n \to \mathbb{R}$	the *inequality constraint functions*
$h_j(x) = 0$	the *equality constraints*
$h_j : \mathbb{R}^n \to \mathbb{R}$	the *equality constraint functions*

If there are no constraints (*i.e.*, $m = p = 0$) we say the problem is unconstrained.

Basic terminology

- The *domain* of the optimization problem (2) is given as

$$\mathcal{D} = \bigcap_{i=0}^{m} \mathbf{dom} f_i \cap \bigcap_{j=1}^{p} \mathbf{dom} h_j.$$

- A point $x \in \mathcal{D}$ is *feasible* if $f_i(x) \leqslant 0, i = 1, \ldots, m$, and $h_j(x) = 0, j = 1, \ldots, p$.

- The problem (2) is said to be feasible if there exists at least one feasible point, and *infeasible* otherwise.

Basic terminology

The *optimal value* v^* of the problem (2) is defined as

$$v^* = \inf\{f_0(x) | f_i(x) \leqslant 0, i = 1, \ldots, m, h_j(x) = 0, j = 1, \ldots, p\}$$

If the problem is infeasible, we have $v^* = \infty$.

- We say x^* is an *optimal point*, or solves the problem (2), if x^* is feasible and $f_0(x^*) = v^*$.
- We say a feasible points x is *locally optimal* if there is a constant $\delta > 0$ such that

$$f_0(x) = \inf\{f_0(z) | f_i(z) \leqslant 0, i = 1, \ldots, m, \\ h_j(z) = 0, j = 1, \ldots, p, \|z - x\|_2 \leqslant \delta\}.$$

Convex optimization

A *convex optimization problem* is one of the form

$$\begin{array}{ll} \min & f_0(x) \\ \text{s.t.} & f_i(x) \leqslant 0, \quad i = 1, \ldots, m \\ & a_j^\top x = b_j, \quad j = 1, \ldots, p \end{array} \tag{3}$$

where $f_0, \ldots, f_m$ are convex functions.

Any locally optimal point of a convex optimization problem is also globally optimal.

An optimality criterion for differentiable f_0

Suppose that the objective f_0 in a convex optimization problem is differentiable. Let X denote the feasible set, *i.e.*,

$$X = \{x | f_i(X) \leqslant 0, i = 1, \ldots, m, h_j(x) = 0, j = 1, \ldots, p\}.$$

Then x is optimal if and only if $x \in X$ and

$$\nabla f_0(x)^\top (y - x) \geqslant 0, \ y \in X. \tag{4}$$

An optimality criterion for differentiable f_0

For an unconstrained problem, the condition (4) reduces to

$$\nabla f_0(x) = 0 \tag{5}$$

for x to be optimal.

An optimality criterion for differentiable f_0

For a convex problem with equality constraints only, *i.e.*,

$$\begin{aligned} \min \ & f_0(x) \\ \text{s.t.} \ & Ax = b \end{aligned}$$

We assume that the feasible set is nonempty. The optimality condition can be expressed as:

$$\nabla f_0(x)^\top u \geqslant 0 \text{ for all } u \in \mathcal{N}(A).$$

In other words,

$$\nabla f_0(x) \perp \mathcal{N}(A).$$

Linear optimization problems

A general *linear program* (LP) has the form

$$\begin{aligned} \min \quad & c^\top x + d \\ \text{s.t.} \quad & Gx \leq h \\ & Ax = b \end{aligned} \tag{6}$$

where $G \in \mathbb{R}^{m \times n}$ and $A \in \mathbb{R}^{p \times n}$. It is common to omit the constant d in the objective function.

Quadratic optimization problems

A convex optimization problem is called *quadratic program* (QP) if it has the form

$$\begin{array}{ll} \min & \frac{1}{2}x^\top Px + q^\top x + r \\ \text{s.t.} & Gx \leq h \\ & Ax = b \end{array} \tag{7}$$

where $P \in \mathbf{S}_+^n$, $G \in \mathbb{R}^{m\times n}$, and $A \in \mathbb{R}^{p\times n}$.

QPs include LPs as a special case by taking $P = 0$.

Quadratic optimization problems

If the objective in (3) as well as the inequality constraint functions are (convex) quadratic, as in

$$\begin{array}{ll} \text{min} & \frac{1}{2}x^\top P_0 x + q_0^\top x + r_0 \\ \text{s.t.} & \frac{1}{2}x^\top P_i x + q_i^\top x + r_i \leqslant 0, \quad i = 1, \ldots, m \\ & Ax = b \end{array} \tag{8}$$

where $P_i \in \mathbf{S}_+^n, i = 0, 1, \ldots, m$, the problem is called a *quadratically constrained quadratic program* (QCQP).

QCQPs include QPs as a special case by taking $P_i = 0$ for $i = 1, \ldots, m$.

Second-order cone programming

A problem that is closely related to quadratic programming is the *second-order cone program* (SOCP):

$$\begin{aligned} \min \quad & f^\top x \\ \text{s.t.} \quad & \|A_i x + b_i\|_2 \leqslant c_i^\top x + d_i, \quad i = 1, \ldots, m \\ & Fx = g \end{aligned} \tag{9}$$

where $x \in \mathbb{R}^n$ is the optimization variable, $A_i \in \mathbb{R}^{n_i \times n}$, and $F \in \mathbb{R}^{p \times n}$.

When $c_i = 0, i = 1, \ldots, m$, the SOCP is equivalent to a QCQP. However, second-order cone programs are more general than QCQPs (and of course, LPs).

Transform a QCQP into an SOCP

For a QCQP problem (8), let y be an auxiliary variable with constraint:

$$\frac{1}{2}x^\top P_0 x + q_0^\top x + r_0 \leqslant y,$$

then (8) becomes

$$\begin{aligned} \min \ & y \\ \text{s.t.} \ & \frac{1}{2}x^\top P_i x + q_i^\top x + r_i \leqslant 0, \quad i = 1, \ldots, m \\ & \frac{1}{2}x^\top P_0 x + q_0^\top x + r_0 - y \leqslant 0 \\ & Ax = b \end{aligned}$$

whose objective is linear. To transform it into an SOCP, we need only translate quadratic constraints into second-order conic ones.

Transform a QCQP into an SOCP

For a quadratic constraint

$$\frac{1}{2}x^T Px + q^\top x + r \leqslant 0$$

with $P \in \mathbf{S}_+^n$, let $A_1 \in \mathbf{S}_+^n$ be the square root of P, *i.e.*, $A_1 A_1 = P$. Let

$$A = \begin{bmatrix} A_1 \\ q^\top \end{bmatrix}, \quad b = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ r + \frac{1}{2} \end{bmatrix} \in \mathbb{R}^{n+1},$$

then the constraint is equivalent to

$$\|Ax + b\|_2 \leqslant -q^\top x - r + \frac{1}{2}.$$

The Lagrangian

Consider an optimization problem in the standard form (2):

$$\begin{array}{ll} \min & f_0(x) \\ \text{s.t.} & f_i(x) \leqslant 0, \quad i = 1, \ldots, m \\ & h_j(x) = 0, \quad j = 1, \ldots, p. \end{array} \tag{10}$$

We assume its domain $\mathcal{D} = \bigcap_{i=0}^{m} \mathbf{dom} f_i \cap \bigcap_{j=1}^{p} \mathbf{dom} h_j$ is nonempty, and denote the optimal value of (10) by v^*, but do not assume the problem (10) is convex.

The Lagrangian

The basic idea of Lagrangian duality is to take the constraints in (10) into account by augmenting the objective function with a weighted sum of the constraint functions.

The Lagrangian

We define the *Lagrangian* $L : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^p \to \mathbb{R}$ associated with the problem (10) as

$$L(x, \lambda, \nu) = f_0(x) + \sum_{i=1}^{m} \lambda_i f_i(x) + \sum_{j=1}^{p} \nu_j h_j(x)$$

with $\mathbf{dom} L = \mathcal{D} \times \mathbb{R}^m \times \mathbb{R}^p$.

- Refer to λ_i as the *Lagrange multiplier* associated with the ith inequality constraint $f_i(x) \leqslant 0$.
- Refer to ν_j as the Lagrange multiplier associated with the jth equality constraint $h_j(x) = 0$.
- The vectors λ and ν are called the *dual variables* or *Lagrange multiplier vectors* associated with the problem (10).

The Lagrange dual function

We define the *Lagrange dual function* (or just *dual function*) $g : \mathbb{R}^m \times \mathbb{R}^p \to \mathbb{R}$ as

$$g(\lambda, \nu) = \inf_{x \in \mathcal{D}} L(x, \lambda, \nu) = \inf_{x \in \mathcal{D}} \left(f_0(x) + \sum_{i=1}^{m} \lambda_i f_i(x) + \sum_{j=1}^{p} \nu_j h_j(x) \right).$$

Since the dual function is the pointwise infimum of a family of affine functions of (λ, ν), it is concave, even when the problem (10) is not convex.

Lower bounds on optimal value

For any $\lambda \geq 0$ and any ν we have

$$g(\lambda, \nu) \leqslant v^*. \tag{11}$$

Proof.

Suppose $\tilde{x}$ is a feasible point for (10), then we have

$$\sum_{i=1}^{m} \lambda_i f_i(\tilde{x}) + \sum_{j=1}^{p} \nu_j h_j(\tilde{x}) \leqslant 0.$$

Hence

$$g(\lambda, \nu) = \inf_{x \in \mathcal{D}} L(x, \lambda, \nu) \leqslant L(\tilde{x}, \lambda, \nu) \leqslant f_0(\tilde{x}).$$

Since $g(\lambda, \nu) \leqslant f_0(\tilde{X})$ holds for every feasible point $\tilde{x}$, the inequality (11) follows. □

Lower bounds on optimal value

The dual function gives a nontrivial lower bound on v^* only when $\lambda \geq 0$ and $(\lambda, \nu) \in \mathbf{dom} g$, *i.e.*, $g(\lambda, \nu) > -\infty$.

We refer to a pair (λ, ν) with $\lambda \geq 0$ and $(\lambda, \nu) \in \mathbf{dom} g$ as *dual feasible*.

Linear approximation interpretation

Let $I_- : \mathbb{R} \to \mathbb{R} \cup \{\infty\}$ and $I_0 : \mathbb{R} \to \mathbb{R} \cup \{\infty\}$ to be the indicator function for the nonpositive reals and $\{0\}$ respectively:

$$I_-(u) = \begin{cases} 0 & u \leqslant 0 \\ \infty & u > 0 \end{cases}, \quad I_0(u) = \begin{cases} 0 & u = 0 \\ \infty & u \neq 0 \end{cases}.$$

Then the original problem (10) can be rewritten as an unconstrained problem:

$$\min \quad f_0(x) + \sum_{i=1}^{m} I_-(f_i(x)) + \sum_{j=1}^{p} I_0(h_j(x)). \tag{12}$$

Linear approximation interpretation

We replace the function $I_-(u)$ with the linear function $\lambda_i u$, where $\lambda_i \geqslant 0$, and the function $I_0(u)$ with $\nu_j u$. The objective becomes the Lagrangian function, i.e.,

$$\text{min} \quad L(x, \lambda, \nu) = f_0(x) + \sum_{i=1}^{m} \lambda_i f_i(x) + \sum_{j=1}^{p} \nu_j h_j(x).$$

In this formulation, we use a linear or "soft" displeasure function in place of I_- and I_0.

Linear function is an *underestimator* of the indicator function. Since $\lambda_i u \leqslant I_-(u)$ and $\nu_j u \leqslant I_0(u)$ for all u, we see immediately that the dual function yields a lower bound on the optimal value of the original problem.

The Lagrange dual problem

To attain the *best* lower bound that can be obtained from the Lagrange dual function leads to the optimization problem

$$\begin{aligned} \max \quad & g(\lambda, \nu) \\ \text{s.t.} \quad & \lambda \geq 0 \end{aligned} \tag{13}$$

This problem is called the *Lagrange dual problem* associated with the problem (10). Correspondingly, the problem (10) is called the *primal problem*.

The Lagrange dual problem

The term *dual feasible*, to describe a pair (λ, ν) with $\lambda \geq 0$ and $g(\lambda, \nu) > -\infty$, now makes sense.

We refer to (λ^*, ν^*) as *dual optimal* or *optimal Lagrange multipliers* if they are optimal for the Lagrange dual problem (13).

The Lagrange dual problem (13) is a convex optimization problem no matter the primal problem is convex or not, since the objective to be maximized is concave and the constraint is convex.

Weak duality

For the optimal value of the Lagrange dual problem g^*, we have

$$g^* \leqslant v^*. \tag{14}$$

This property is called *weak duality*.

$v^* - g^*$ is the *optimal duality gap* of the original problem.

Strong duality and Slater's constraint qualification

If the equality

$$g^* = v^* \tag{15}$$

holds, then we say that *stong duality* holds.

- Strong duality does not, in general, hold.
- For a convex primal problem, there are many additional conditions on the primal problem, under which strong duality holds.

Strong duality and Slater's constraint qualification

One simple condition is *Slater's condition*:

> There exists an $x \in \mathbf{relint}\mathcal{D}$ such that
>
> $$f_i(x) < 0, \quad i = 1, \ldots, m, \qquad Ax = b, \tag{16}$$
>
> where $\mathbf{relint}\mathcal{D} = \{x \in \mathcal{D} | B(x, r) \cap \mathbf{aff}\mathcal{D} \subseteq \mathcal{D} \text{ for some } r > 0\}$. Such a point is called *strictly feasible*.

Slater's theorem states that strong duality holds if Slater's condition holds (and the problem is convex).

Optimality conditions

Dual feasible points allow us to bound how suboptimal a given feasible point is, without knowing the exact value of v^*.

If x is primal feasible and (λ, ν) is dual feasible, then

$$f_0(x) - v^* \leqslant f_0(x) - g(\lambda, \nu)$$

and

$$v^* \in [g(\lambda, \nu), f_0(x)], \quad g^* \in [g(\lambda, \nu), f_0(x)].$$

It leads to

$$g(\lambda, \nu) = f_0(x) \Longrightarrow v^* = f_0(x) = g(\lambda, \nu) = g^*.$$

We refer to $f_0(x) - g(\lambda, \nu)$ as the *duality gap* associated with the primal feasible point x and dual feasible point (λ, ν).

Complementary slackness

Suppose that the primal and dual optimal values are attained and equal, let x^* be a primal optimal and (λ^*, ν^*) be a dual optimal points, then

$$\begin{aligned} f_0(x^*) &= g(\lambda^*, \nu^*) \\ &= \inf_x \left(f_0(x) + \sum_{i=1}^m \lambda_i^* f_i(x) + \sum_{j=1}^p \nu_j^* h_j(x) \right) \\ &\leqslant f_0(x^*) + \sum_{i=1}^m \lambda_i^* f_i(x^*) + \sum_{j=1}^p \nu_j^* h_j(x^*) \\ &\leqslant f_0(x^*) \end{aligned}$$

Complementary slackness

By $\lambda_i^* \geqslant 0, f_i(x^*) \leqslant 0, i = 1, \ldots, m$, we have

$$\lambda_i^* f_i(x^*) = 0, \quad i = 1, \ldots, m. \tag{17}$$

This condition is known as *complementary slackness*.

We can express it as

$$\lambda_i^* > 0 \Longrightarrow f_i(x^*) = 0,$$
$$f_i(x^*) < 0 \Longrightarrow \lambda_i^* = 0.$$

KKT optimality conditions

We now assume that the functions $f_0, \ldots, f_m, h_1, \ldots, h_p$ are differentiable. As above, let x^* and (λ^*, ν^*) be any primal and dual optimal points with zero duality gap.

Since x^* minimizes $L(x, \lambda^*, \nu^*)$ over x, it follows

$$\nabla f_0(x^*) + \sum_{i=1}^{m} \lambda_i^* \nabla f_i(x^*) + \sum_{j=1}^{p} \nu_j^* \nabla h_j(x^*) = 0.$$

KKT optimality conditions

Together with constraints and complementary slackness, we have

$$\begin{cases} f_i(x^*) \leqslant 0, \quad i = 1, \ldots, m \\ h_j(x^*) = 0, \quad j = 1, \ldots, p \\ \lambda_i^* \geqslant 0, \quad i = 1, \ldots, m \\ \lambda_i^* f_i(x^*) = 0, \quad i = 1, \ldots, m \\ \nabla f_0(x^*) + \sum_{i=1}^m \lambda_i^* \nabla f_i(x^*) + \sum_{j=1}^p \nu_j^* \nabla h_j(x^*) = 0 \end{cases} \tag{18}$$

which are called the *Karush-Kuhn-Tucker* (KKT) conditions.

KKT optimality conditions

For *any* optimization problem with differentiable objective and constraint functions for which strong duality obtains, any pair of primal and dual optimal points must satisfy the KKT conditions.

When the primal problem is convex, the KKT conditions are also sufficient for the points to be primal and dual optimal.

About optimization algorithm

There is *no* analytical formula for the solution of convex optimization problems, not to mention general nonlinear optimization problems.

Thus we describe numerical methods for solving convex optimization problems in the section.

Recall: descent methods

To solve an unconstrained optimization problem

$$\min f(x)$$

where $f(x)$ is differentiable and convex, we usually employ descent methods.

Recall: descent methods

Given a starting point $x^{(0)}$, a descent method produces a sequence $x^{(k)}, k = 1, \ldots,$ where

$$x^{(k+1)} = x^{(k)} + \alpha_k \delta_x^{(k)}, \quad f(x^{(k+1)}) < f(x^{(k)}). \tag{19}$$

We usually drop the superscripts and use the notation $x := x + \alpha\delta_x$ to focus on one iteration of an algorithm. $\alpha > 0$ is called step size and δ_x called search direction. Different methods differ from choices of α or/and δ_x.

Recall: gradient descent and Newton's method

Given a descent direction δ_x, we usually use line search to determine step size α.

Different search directions:

- Negative gradient:
$$\delta_x = -\nabla f(x).$$
- Normalized steepest descent direction (with respect to the norm $\|\cdot\|$):
$$\delta_{x_{\text{nsd}}} = \arg\min\{\nabla f(x)^\top v \mid \|v\| = 1\}.$$
- Newton step:
$$\delta_{x_{\text{nt}}} = -\nabla^2 f(x)^{-1} \nabla f(x).$$

Equality constrained minimization problems

A convex optimization problem with equality constraints has the form

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & Ax = b, \end{aligned} \tag{20}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is convex and twice continuously differentiable, and $A \in \mathbb{R}^{p \times n}$ with $\mathbf{rank} A = p < n$. We assume that an optimal solution x^* exists and $v^* = f(x^*)$.

KKT condition

Recall the KKT conditions for (20): a point $x^* \in \mathbf{dom} f$ is optimal if and only if there is a multiplier $\nu^* \in \mathbb{R}^p$ such that

$$Ax^* = b, \quad \nabla f(x^*) + A^\top \nu^* = 0. \tag{21}$$

The first set of equations, $Ax^* = b$, are called the *primal feasibility equations*.
The second set of equations, $\nabla f(x^*) + A^\top \nu^* = 0$, are called the *dual feasibility equations*.

Newton's method with equality constraints

Newton's method with equality constraints is almost the same as Newton's method without constraints, except for two differences:

- The initial point must be feasible (*i.e.*, $x \in \mathbf{dom} f$ and $Ax = b$).
- The definition of Newton step $\delta_{x_{\mathrm{nt}}}$ is modified to take the equality constraints into account.

The Newton step

To derive the Newton step $\delta_{x_{\text{nt}}}$ for problem (20) at the feasible point x, we replace the objective with its second-order Taylor approximation near x

$$\begin{array}{ll} \text{min} & \hat{f}(x+s) = f(x) + \nabla f(x)^\top s + \frac{1}{2} s^\top \nabla^2 f(x) s \\ \text{s.t.} & A(x+s) = b \end{array} \tag{22}$$

with variable s. Suppose $\delta_{x_{\text{nt}}}$ is optimal for (22). By KKT conditions, there exists associated optimal dual variable $w \in \mathbb{R}^p$ such that

$$\begin{bmatrix} \nabla^2 f(x) & A^\top \\ A & 0 \end{bmatrix} \begin{bmatrix} \delta_{x_{\text{nt}}} \\ w \end{bmatrix} = \begin{bmatrix} -\nabla f(x) \\ 0 \end{bmatrix}. \tag{23}$$

The Newton step

We can also derive the Newton Step $\delta_{x_{\text{nt}}}$ by simply replacing x^* and ν^* in the KKT conditions for problem (20):

$$Ax^* = b, \quad \nabla f(x^*) + A^\top \nu^* = 0$$

with $x + \delta_{x_{\text{nt}}}$ and w, respectively, and replace the gradient term in the second equation by its linearized approximation near x, to obtain the equations

$$A(x + \delta_{x_{\text{nt}}}) = b,$$
$$\nabla f(x + \delta_{x_{\text{nt}}}) + A^\top w \approx \nabla f(x) + \nabla^2 f(x)\delta_{x_{\text{nt}}} + A^\top w = 0.$$

The Newton step

Using $Ax = b$, these become

$$A\delta_{x_{\text{nt}}} = 0, \quad \nabla^2 f(x)\delta_{x_{\text{nt}}} + A^\top w = -\nabla f(x),$$

which are precisely the equations (23).

The Newton decrement

The Newton decrement is defined as

$$\kappa(x) = (\delta_{x_{\mathrm{nt}}}^{\top} \nabla^2 f(x) \delta_{x_{\mathrm{nt}}})^{1/2}.$$

Since

$$\frac{\mathrm{d}}{\mathrm{d}\alpha} f(x + \alpha \delta_{x_{\mathrm{nt}}}) \Big|_{\alpha=0} = \nabla f(x)^{\top} \delta_{x_{\mathrm{nt}}} = -\kappa(x)^2,$$

the algorithm should terminate when $\kappa(x)$ is small.

Newton's method with equality constraints

Algorithm. *Newton's method for equality constrained minimization.*

given starting point $x \in \mathbf{dom} f$ with $Ax = b$, tolerance $\epsilon > 0$.

repeat

1. Compute the Newton step and decrement $\delta_{x_{\text{nt}}}, \kappa(x)$.
2. *Stopping criterion.* **quit** if $\kappa^2/2 \leqslant \epsilon$.
3. *Line search* Choose step size α by backtracking line search.
4. *update.* $x := x + \alpha\delta_{x_{\text{nt}}}$.

Infeasible start Newton method

Newton's method described above is a feasible descent method. Now we describe a generalization of Newton's method that works with initial points and iterates that are *not* feasible.

Newton step at infeasible points

Let x denote the current point, which we do not assume to be feasible, but we do assume satisfies $x \in \mathbf{dom} f$.

Our goal is to find a step δ_x so that $x + \delta_x$ satisfies the optimality conditions (21), *i.e.*, $x + \delta_x \approx x^*$.

Newton step at infeasible points

Similarly, we substitute $x + \delta_x$ for x^* and μ for ν^* in

$$Ax^* = b, \quad \nabla f(x^*) + A^\top \nu^* = 0$$

and use the first-order approximation for the gradient to obtain

$$A(x + \delta_x) = b,$$

$$\nabla f(x + \delta_x) + A^\top \mu \approx \nabla f(x) + \nabla^2 f(x)\delta_x + A^\top \mu = 0.$$

This is a set of linear equations for δ_x and μ,

$$\begin{bmatrix} \nabla^2 f(x) & A^\top \\ A & 0 \end{bmatrix} \begin{bmatrix} \delta_x \\ \mu \end{bmatrix} = - \begin{bmatrix} \nabla f(x) \\ Ax - b \end{bmatrix}. \tag{24}$$

Interpretation as primal-dual Newton step

We express the optimality conditions (21) as $r(x^*, \nu^*) = 0$, where $r : \mathbb{R}^n \times \mathbb{R}^p \mapsto \mathbb{R}^n \times \mathbb{R}^p$ is defined as

$$r(x, \nu) = (r_{\text{dual}}(x, \nu), r_{\text{pri}}(x, \nu)).$$

Here

$$r_{\text{dual}}(x, \nu) = \nabla f(x) + A^\top \nu, \quad r_{\text{pri}}(x, \nu) = Ax - b$$

are the *dual residual* and *primal residual*, respectively.

Interpretation as primal-dual newton step

The first-order Taylor approximation of r, near our current point $y = (x, \nu)$, is

$$r(y + \delta_y) \approx \hat{r}(y + \delta_y) = r(y) + J[r(y)]\delta_y,$$

where $J[r(y)] \in \mathbb{R}^{(n+p)\times(n+p)}$ is the derivative (Jacobian) of r, evaluated at y.

Interpretation as primal-dual Newton step

We define $\delta_{y_{\text{pd}}}$ as the primal-dual Newton step for which $\hat{r}(y + \delta_y) = 0$, *i.e.*,

$$J[r(y)]\delta_{y_{\text{pd}}} = -r(y). \tag{25}$$

Note that $\delta_{y_{\text{pd}}} = (\delta_{x_{\text{pd}}}, \delta_{\nu_{\text{pd}}})$ gives both a primal and a dual step.

Interpretation as primal-dual Newton step

Equations (25) can be expressed as

$$\begin{bmatrix} \nabla^2 f(x) & A^\top \\ A & 0 \end{bmatrix} \begin{bmatrix} \delta_{x_{\text{pd}}} \\ \delta_{\nu_{\text{pd}}} \end{bmatrix} = -\begin{bmatrix} r_{\text{dual}} \\ r_{\text{pri}} \end{bmatrix} = -\begin{bmatrix} \nabla f(x) + A^\top \nu \\ Ax - b \end{bmatrix}. \quad (26)$$

Writing $\nu + \delta_{\nu_{\text{pd}}}$ as μ, we find it coincide with (24)

$$\begin{bmatrix} \nabla^2 f(x) & A^\top \\ A & 0 \end{bmatrix} \begin{bmatrix} \delta_x \\ \mu \end{bmatrix} = -\begin{bmatrix} \nabla f(x) \\ Ax - b \end{bmatrix}.$$

Residual norm reduction property

The Newton direction at an infeasible point is not necessarily a descent direction for f.

The primal-dual interpretation, however, shows that the norm of the residual decreases in the Newton direction. By calculation we have

$$\frac{\mathrm{d}}{\mathrm{d}\alpha}\|r(y+\alpha\delta_{y_{\mathrm{pd}}})\|_2\Big|_{\alpha=0} = -\|r(y)\|_2.$$

This allows us to use $\|r\|_2$ to measure the progress of the infeasible start Newton method.

Infeasible start Newton method

Algorithm. *Infeasible start Newton method.*

given starting point $x \in \mathbf{dom} f$ with $Ax = b$, tolerance $\epsilon > 0$, $\tau \in (0, 1/2), \gamma \in (0, 1)$.

repeat

1. Compute primal and dual Newton steps $\delta_{x_{\mathrm{nt}}}, \delta_{\nu_{\mathrm{nt}}}$.
2. *Backtracking line search on* $\|r\|_2$.
   $\alpha := 1$.
   while $\|r(x + \alpha\delta_{x_{\mathrm{nt}}}, \nu + \alpha\delta_{\nu_{\mathrm{nt}}})\|_2 > (1 - \tau\alpha)\|r(x, \nu)\|_2$, $\alpha := \gamma\alpha$.
3. *Update.* $x := x + \alpha\delta_{x_{\mathrm{nt}}}, \nu := \nu + \alpha\delta_{\nu_{\mathrm{nt}}}$.

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

Inequality constrained minimization problems

The convex optimization problems that include inequality constraints:

$$\begin{array}{ll} \min & f_0(x) \\ \text{s.t.} & f_i(x) \leqslant 0, \quad i = 1, \ldots, m \\ & Ax = b \end{array} \tag{27}$$

where $f_0, \ldots, f_m : \mathbb{R}^n \to \mathbb{R}$ are convex and twice continuously differentiable, and $A \in \mathbb{R}^{p \times n}$ with $\mathbf{rank} A = p < n$.

We assume that an optimal x^* exists and denote the optimal value $f_0(x^*)$ as v^*.

Assumptions

We also assume that the problem is strictly feasible, *i.e.*, $\exists x \in \mathcal{D}$ satisfying $Ax = b$ and $f_i(x) < 0$ for $i = 1, \ldots, m$.

This means that Slater's constraint qualification holds, and therefore strong duality holds, so there exists dual optimal $\lambda^* \in \mathbb{R}^m, \nu^* \in \mathbb{R}^p$, which together with x^* satisfy the KKT conditions:

$$\begin{aligned} Ax^* = b, \quad f_i(x^*) &\leqslant 0, \quad i = 1, \ldots, m \\ \lambda^* &\geqslant 0 \\ \nabla f_0(x^*) + \textstyle\sum_{i=1}^m \lambda_i^* \nabla f_i(x^*) + A^\top \nu^* &= 0 \\ \lambda_i^* f_i(x^*) &= 0, \quad i = 1, \ldots, m. \end{aligned} \tag{28}$$

About interior-point method

Interior-point methods solve the problem (27) by applying Newton's method to a sequence of equality constrained problems, or to a sequence of modified versions of the KKT conditions.

We will introduce two particular interior-point algorithms:

- *The barrier method*
- *The primal-dual interior-point method*

Logarithmic barrier function

Rewrite the problem (27) and make the inequality constraints implicit in the objective:

$$\begin{aligned} \min \quad & f_0(x) + \sum_{i=1}^{m} I_-(f_i(x)) \\ \text{s.t.} \quad & Ax = b, \end{aligned} \tag{29}$$

where

$$I_-(u) = \begin{cases} 0 & u \leqslant 0 \\ \infty & u > 0. \end{cases}$$

Logarithmic barrier function

The basic idea of the barrier method is to approximate the indicator function I_- by the function

$$\hat{I}_-(u) = -(1/t)\log(-u), \quad \mathbf{dom}\hat{I}_- = -\mathbb{R}_{++}$$

where t is a parameter that sets the accuracy of the approximation.

Obviously, $\hat{I}_-$ is convex, nondecreasing and differentiable.

Logarithmic barrier function

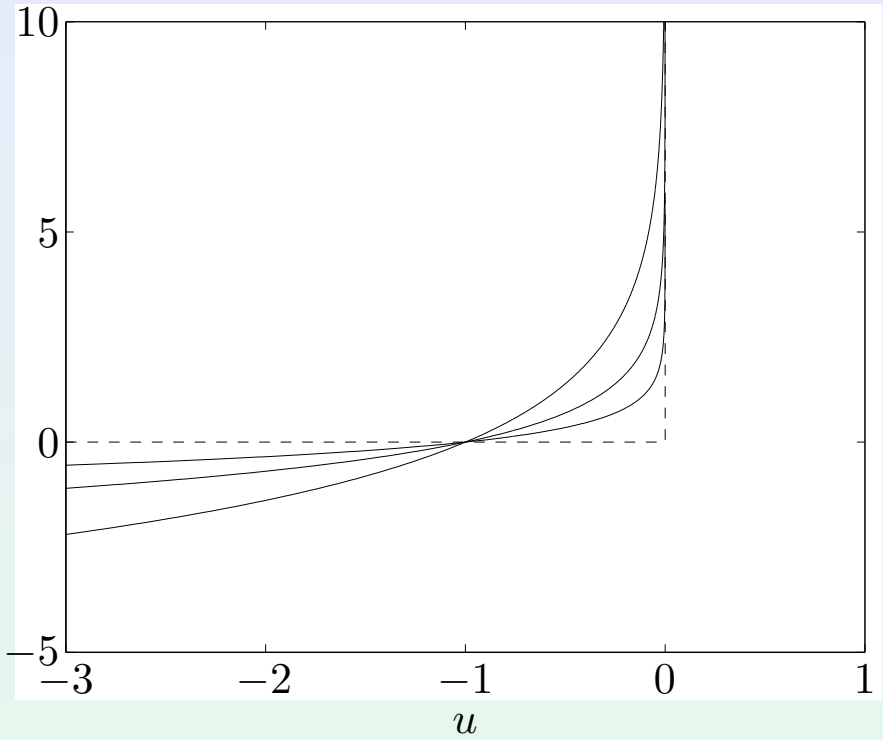


Figure: The dashed lines show the function $I_-(u)$, and the solid curves show $\hat{I}_-(u) = -(1/t)\log(-u)$, for $t = 0.5, 1, 2$. The curve for $t = 2$ gives the best approximation.

Logarithmic barrier function

Substituting $\hat{I}_-$ for I_- in (29) gives the approximation

$$\begin{array}{ll} \min & f_0(x) + \sum_{i=1}^{m} -(1/t)\log(-f_i(x)) \\ \text{s.t.} & Ax = b. \end{array} \tag{30}$$

The function

$$\phi(x) = -\sum_{i=1}^{m} \log(-f_i(x)), \tag{31}$$

is called the *logarithmic barrier* for the problem (27). Its domain is the set of points that satisfy the inequality constraints of (27) strictly:

$$\mathbf{dom}\phi = \{x \in \mathbb{R}^n | f_i(x) < 0, i = 1, \ldots, m\}.$$

Logarithmic barrier function

The gradient and Hessian of ϕ are given by

$$\nabla\phi(x) = \sum_{i=1}^{m} \frac{1}{-f_i(x)} \nabla f_i(x),$$

$$\nabla^2\phi(x) = \sum_{i=1}^{m} \frac{1}{f_i(x)^2} \nabla f_i(x) \nabla f_i(x)^\top + \sum_{i=1}^{m} \frac{1}{-f_i(x)} \nabla^2 f_i(x).$$

Central path

We multiply the objective of (30) by t, and consider the equivalent problem

$$\begin{array}{ll} \min & tf_0(x) + \phi(x) \\ \text{s.t.} & Ax = b. \end{array} \tag{32}$$

We assume problem (32) can be solved via *Newton's method*, and, that it has a unique solution for each $t > 0$.

Central path

For $t > 0$ we define $x^*(t) = \arg\min_x \{tf_0(x) + \phi(x) \text{ s.t. } Ax = b\}$ as the solution of (32).

The *central path* associated with problem (27) is defined as the set of points $\{x^*(t) \mid t > 0\}$, which we call the *central points*.

Central path

Points on the central path are characterized by the following necessary and sufficient conditions: $x^*(t)$ is strictly feasible, *i.e.*, satisfies

$$Ax^*(t) = b, \quad f_i(x^*(t)) < 0, \ i = 1, \ldots, m$$

and $\exists \hat{\nu} \in \mathbb{R}^p$ such that

$$\begin{aligned} 0 &= t\nabla f_0(x^*(t)) + \nabla \phi(x^*(t)) + A^\top \hat{\nu} \\ &= t\nabla f_0(x^*(t)) + \sum_{i=1}^{m} \frac{1}{-f_i(x^*(t))} \nabla f_i(x^*(t)) + A^\top \hat{\nu} \end{aligned} \tag{33}$$

holds.

Dual points from central path

Every central point yields a dual feasible point.

Define

$$\lambda_i^*(t) = -\frac{1}{tf_i(x^*(t))},\ i = 1, \ldots, m, \quad \nu^*(t) = \frac{\hat{\nu}}{t}. \tag{34}$$

Because $f_i(x^*(t)) < 0, i = 1, \ldots, m$, it's clear that $\lambda^*(t) > 0$.

Dual points from central path

By expressing (33) as

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

we see that $x^*(t)$ minimizes the Lagrangian

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

for $\lambda = \lambda^*(t)$ and $\nu = \nu^*(t)$. Thus $(\lambda^*(t), \nu^*(t))$ is a dual feasible pair.

Dual points from central path

Therefore the dual function $g(\lambda^*(t), \nu^*(t)) = \min_x L(x, \lambda^*(t), \nu^*(t))$ is finite and

$$\begin{aligned} g(\lambda^*(t), \nu^*(t)) &= f_0(x^*(t)) + \sum_{i=1}^{m} \lambda_i^*(t) f_i(x^*(t)) + \nu^*(t)^\top (Ax^*(t) - b) \\ &= f_0(x^*(t)) - m/t. \end{aligned}$$

- As an important consequence, we have

$$f_0(x^*(t)) - v^* \leqslant m/t.$$

- This confirms that $x^*(t)$ converge to an optimal point as $t \to \infty$.

Interpretation via KKT conditions

Since we have assumed that $x^*(t)$ is the unique solution to problem (32) for each $t > 0$, a point is equal to $x^*(t)$ if and only if $\exists \lambda, \nu$ such that

$$\begin{aligned} Ax = b, \quad f_i(x) &\leqslant 0, \qquad i = 1, \ldots, m \\ \lambda &\geqslant 0 \\ \nabla f_0(x) + \sum_{i=1}^m \lambda_i \nabla f_i(x) + A^\top \nu &= 0 \\ -\lambda_i f_i(x) &= 1/t, \quad i = 1, \ldots, m. \end{aligned} \tag{35}$$

The only difference between (35) and the KKT condition (28) is that the complementarity condition $-\lambda_i f_i(x) = 0$ is replaced by the condition $-\lambda_i f_i(x) = 1/t$.

In particular, for large t, $x^*(t)$ and $\lambda^*(t), \nu^*(t)$ *'almost' satisfy* the KKT optimality conditions for the problem (27).

The barrier method

Algorithm. *Barrier method*

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

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

An execution of step 1 is called an *outer iteration*. We assume that Newton's method is used in step 1, and we refer to the Newton iterations or steps executed during the centering step as *inner iterations*.

The barrier method

- Computing $x^*(t)$ exactly is not necessary.
- The choice of the parameter γ involves a trade-off:
  If γ is small (*i.e.*, near 1) then centering step will be easy since the previous iterate x is a very good starting point but of course there will be a large number of outer iterations.
  On the other hand, a large γ resulting in fewer outer iterations but more inner iterations.
- Choice of $t^{(0)}$:
  If $t^{(0)}$ is chosen too large, the first outer iteration will require too many iterations.
  If $t^{(0)}$ is chosen too small, the algorithm will require extra outer iterations.

Newton step for modified KKT equations

In the step 1 of the barrier method, the Newton step $\delta_{x_{\text{nt}}}$ and associated dual variable are given by the linear equations

$$\begin{bmatrix} t\nabla^2 f_0(x) + \nabla^2 \phi(x) & A^\top \\ A & 0 \end{bmatrix} \begin{bmatrix} \delta_{x_{\text{nt}}} \\ \nu_{\text{nt}} \end{bmatrix} = - \begin{bmatrix} t\nabla f_0(x) + \nabla \phi(x) \\ 0 \end{bmatrix}. \quad (36)$$

These Newton steps for the centering problem can be interpreted as Newton steps for directly solving the modified KKT equations

$$\begin{aligned} \nabla f_0(x) + \textstyle\sum_{i=1}^m \lambda_i \nabla f_i(x) + A^\top \nu &= 0 \\ -\lambda_i f_i(x) &= 1/t, \quad i = 1, \ldots, m \\ Ax &= b. \end{aligned} \quad (37)$$

Newton step for modified KKT equations

Let $\lambda_i = -1/(tf_i(x))$. This transforms (37) into

$$\nabla f_0(x) + \sum_{i=1}^{m} \frac{1}{-tf_i(x)} \nabla f_i(x) + A^\top \nu = 0, \quad Ax = b. \tag{38}$$

For small δ_x,

$$\begin{aligned}
&\nabla f_0(x+\delta_x) + \sum_{i=1}^{m} \frac{1}{-tf_i(x+\delta_x)} \nabla f_i(x+\delta_x) \\
\approx &\nabla f_0(x) + \sum_{i=1}^{m} \frac{1}{-tf_i(x)} \nabla f_i(x) + \nabla^2 f_0(x)\delta_x + \sum_{i=1}^{m} \frac{1}{-tf_i(x)} \nabla^2 f_i(x)\delta_x \\
&+ \sum_{i=1}^{m} \frac{1}{tf_i(x)^2} \nabla f_i(x) \nabla f_i(x)^\top \delta_x.
\end{aligned}$$

Newton step for modified KKT equations

Let

$$H = \nabla^2 f_0(x) + \sum_{i=1}^{m} \frac{1}{-tf_i(x)} \nabla^2 f_i(x) + \sum_{i=1}^{m} \frac{1}{tf_i(x)^2} \nabla f_i(x) \nabla f_i(x)^\top$$

$$g = \nabla f_0(x) + \sum_{i=1}^{m} \frac{1}{-tf_i(x)} \nabla f_i(x).$$

Observe that

$$H = \nabla f_0(x) + (1/t)\nabla^2 \phi(x), \quad g = \nabla f_0(x) + (1/t)\nabla \phi(x).$$

The Newton step for (38) is

$$H\delta_x + A^\top \nu = -g, \quad A\delta_x = 0.$$

Comparing this with (36) shows that

$$\delta_x = \delta_{x_{\text{nt}}}, \quad \nu = \frac{\nu_{\text{nt}}}{t}.$$

Feasibility and phase I method

- The barrier method requires a strictly feasible starting point $x^{(0)}$.
- When such a point is not known, the barrier method is preceded by a preliminary stage, called *phase I*, in which a strictly feasible point is computed and used as the starting point for the barrier method.

Basic phase I method

To find a strictly feasible solution of inequalities and equalities

$$f_i(x) < 0, \quad i = 1, \ldots, m, \quad Ax = b, \tag{39}$$

we form and solve the following optimization problem

$$\begin{array}{ll} \min & s \\ \text{s.t.} & f_i(x) \leqslant s, \quad i = 1, \ldots, m \\ & Ax = b \end{array} \tag{40}$$

in the variable $x \in \mathbb{R}^n, s \in \mathbb{R}$. It's always strictly feasible, and called the *phase I optimization problem* associated with the inequality and equality system (39).

Basic phase I method

Let $\bar{v}^*$ be the optimal value of (40).

- If $\bar{v}^* < 0$, then (39) has a strictly feasible solution. In fact, we can terminate solving the problem (40) when $s < 0$.
- If $\bar{v}^* > 0$, then (39) is infeasible. In fact, we can terminate when a central point give a positive lower bound of $\bar{v}^* > 0$.
- If $\bar{v}^* = 0$ and the minimum is attained at x^* and $s^* = 0$, then the set of inequalities is feasible but note strictly feasible. If $\bar{v}^* = 0$ and the minimum is not attained, then the inequalities are infeasible.

Primal-dual search direction

The modified KKT conditions (37) can be expressed as $r_t(x, \lambda, \nu) = 0$, where

$$r_t(x, \lambda, \nu) = \begin{bmatrix} \nabla f_0(x) + J[f(x)]^\top \lambda + A^\top \nu \\ -\mathbf{diag}(\lambda) f(x) - (1/t)\mathbf{1} \\ Ax - b \end{bmatrix}, \tag{41}$$

and $t > 0$. Here $f : \mathbb{R}^n \to \mathbb{R}^m$ and $J[f]$ are given by

$$f(x) = \begin{bmatrix} f_1(x) \\ \vdots \\ f_m(x) \end{bmatrix}, \quad J[f(x)] = \begin{bmatrix} \nabla f_1(x)^\top \\ \vdots \\ \nabla f_m(x)^\top \end{bmatrix}.$$

Primal-dual search direction

If x, λ, ν satisfy $r_t(x, \lambda, \nu) = 0$ (and $f_i(x) < 0$), then $x = x^*(t)$, $\lambda = \lambda^*(t)$ and $\nu = \nu^*(t)$.

- The first block component of r_t,

$$r_{\text{dual}} = \nabla f_0(x) + Df(x)^\top \lambda + A^\top \nu$$

  is called the *dual residual*.
- The last block component, $r_{\text{pri}} = Ax - b$, is called the *primal residual.*
- The middle block

$$r_{\text{cent}} = -\mathbf{diag}(\lambda) f(x) - (1/t)\mathbf{1},$$

  is the *centrality residual*, *i.e.*, the residual for the modified complementarity condition.

Primal-dual search direction

Let $y = (x, \lambda, \nu)$ denote the current point and $\delta_y = (\delta_x, \delta_\lambda, \delta_\nu)$ denote the Newton step for solving the equation $r_t(x, \lambda, \nu) = 0$, for fixed t where $f(x) < 0, \lambda > 0$.

The Newton step is characterized by

$$r_t(y + \delta_y) \approx r_t(y) + J[r_t(y)]\delta_y = 0.$$

Primal-dual search direction

In terms of x, λ, ν, we have

$$\begin{bmatrix} \nabla^2 f_0(x) + \sum_{i=1}^m \lambda_i \nabla^2 f_i(x) & J[f(x)]^\top & A^\top \\ -\mathbf{diag}(\lambda) J[f(x)] & -\mathbf{diag}(f(x)) & 0 \\ A & 0 & 0 \end{bmatrix} \begin{bmatrix} \delta_x \\ \delta_\lambda \\ \delta_\nu \end{bmatrix} = - \begin{bmatrix} r_{\text{dual}} \\ r_{\text{cent}} \\ r_{\text{pri}} \end{bmatrix} \tag{42}$$

The *primal-dual search direction* $\delta_{y_{\text{pd}}} = (\delta_{x_{\text{pd}}}, \delta_{\lambda_{\text{pd}}}, \delta_{\nu_{\text{pd}}})$ is defined as the solution of (42).

The surrogate duality gap

In the primal-dual interior-point method the iterates $x^{(k)}, \lambda^{(k)}$ and $\nu^{(k)}$ are not necessarily feasible. We cannot easily evaluate a duality gap as we do in the barrier method.

Instead, we define the *surrogate duality gap*, for any x that satisfies $f(x) < 0$ and $\lambda \geqslant 0$, as

$$\hat{\eta}(x, \lambda) = -f(x)^{\top} \lambda.$$

The surrogate gap $\hat{\eta}$ would be the duality gap, if x were primal feasible and λ, ν were dual feasible. Note that the value of the parameter t corresponding to the surrogate duality gap $\hat{\eta}$ is $m/\hat{\eta}$.

Primal-dual interior-point method

Algorithm. *Primal-dual interior-point method.*

given x that satisfies
$f_1(x) < 0, \ldots, f_m(x) < 0, \lambda > 0, \gamma > 1, \epsilon_{\text{feas}} > 0, \epsilon > 0$.
repeat

1. *Determine t.* Set $t := \gamma m/\hat{\eta}$.
2. Compute primal-dual search direction $\delta_{y_{\text{pd}}}$.
3. *Line search and update.*
   Determine step length $\alpha > 0$ and set $y := y + \alpha\delta_{y_{\text{pd}}}$.

until $\|r_{\text{pri}}\|_2 \leqslant \epsilon_{\text{feas}}$, $\|r_{\text{dual}}\|_2 \leqslant \epsilon_{\text{feas}}$, and $\hat{\eta} \leqslant \epsilon$.

Line search in primal-dual interior-point method

The line search in step 3 is a standard backtracking line search.

For a step size α, let

$$y^+ = \begin{bmatrix} x^+ \\ \lambda^+ \\ \nu^+ \end{bmatrix} = \begin{bmatrix} x \\ \lambda \\ \nu \end{bmatrix} + \alpha \begin{bmatrix} \delta_{x_{\text{pd}}} \\ \delta_{\lambda_{\text{pd}}} \\ \delta_{\nu_{\text{pd}}} \end{bmatrix}$$

Let

$$\alpha^{\max} = \sup\{\alpha \in [0,1] | \lambda + \alpha\delta_\lambda \geqslant 0\} = \min\{1, \min\{-\lambda_i/\delta_{\lambda_i} | \delta_{\lambda_i} < 0\}\}$$

to be the largest positive step length the gives $\lambda^+ \geqslant 0$.

Line search in primal-dual interior-point method

We start backtracking with $\alpha = 0.99\alpha^{\max}$, and multiply α by $\beta \in (0, 1)$ until we have $f(x^+) < 0$. We continue multiplying α by β until we have

$$\|r_t(x^+, \lambda^+, \nu^+)\|_2 \leqslant (1 - \tau\alpha)\|r_t(x, \lambda, \nu)\|_2.$$

Here τ is typically chosen in the range 0.01 to 0.1.

Exercises

Ex 1. Let $C \subseteq \mathbb{R}^n$ be the solution set of a quadratic inequality,

$$C = \{x \in \mathbb{R}^n | x^\top A x + b^\top x + c \leqslant 0\},$$

with $A \in \mathbf{S}^n$, $b \in \mathbb{R}^n$, and $c \in \mathbb{R}$.

(a) Show that C is convex if $A \succeq 0$.

(b) Show that the intersection of C and the hyperplane defined by $g^\top x + h = 0$ (where $g \neq 0$) is convex if $A + \lambda g g^\top \succeq 0$ for some $\lambda \in \mathbb{R}$.

Ex 2. Let $\lambda_1(X) \geqslant \lambda_2(X) \geqslant \ldots \geqslant \lambda_n(X)$ denote the eigenvalues of a matrix $X \in \mathbf{S}^n$. Prove that the maximum eigenvalue $\lambda_1(X)$ is convex. Moreover, Show that $\sum_{i=1}^k \lambda_i(X)$ is convex on $\mathbf{S}^n$. *Hint.* Use the variational characterization

$$\sum_{i=1}^k \lambda_i(X) = \sup\{\mathbf{tr}(V^\top X V) | V \in \mathbb{R}^{n\times k}, V^\top V = \mathrm{I}\}.$$

Exercises

Ex 3. Find the dual function of the LP

$$\begin{array}{ll} \min & c^T x \\ \text{s.t.} & Gx \preceq h \\ & Ax = b. \end{array}$$

Give the dual problem, and make the implicit equality constraints explicit.

Ex 4. Consider the equality constrained least-squares problem

$$\begin{array}{ll} \min & \|Ax - b\|_2^2 \\ \text{s.t.} & Gx = h \end{array}$$

where $A \in \mathbb{R}^{m \times n}$ with $\mathbf{rank} A = n$, and $G \in \mathbb{R}^{p \times n}$ with $\mathbf{rank} G = p$. Give the KKT conditions, and derive expressions for the primal solution x^* and the dual solution ν^*.

Exercises

Ex 5. Suppose $Q \succeq 0$. The problem

$$\begin{array}{ll} \min & f(x) + (Ax - b)^\top Q(Ax - b) \\ \text{s.t.} & Ax = b \end{array}$$

is equivalent to the original equality constrained optimization problem (20). What is the Newton step for this problem? Is it the same as that for the original problem?

Exercises

Ex 6. Suppose we use the infeasible start Newton method to minimize $f(x)$ subject to $a_i^\top x = b_i$, $i = 1, \ldots, p$.

(a) Suppose the initial point $x^{(0)}$ satisfies the linear equality $a_i^\top x^{(0)} = b_i$. Show that the linear equality will remain satisfied for future iterates, *i.e.*, $a_i^\top x^{(k)} = b_i$ for all k.

(b) Suppose that one of the equality constraints becomes satisfied at iteration k, *i.e.*, we have $a_i^\top x^{(k-1)} \neq b_i$, $a_i^\top x^{(k)} = b_i$. Show that at iteration k, *all* the equality constraints are satisfied.

Exercises

Ex 7. Suppose we add the constraint $x^\top x \leqslant R^2$ to the problem (27):

$$\begin{array}{ll} \min & f_0(x) \\ \text{s.t.} & f_i(x) \leq 0, \quad i = 1, \ldots, m \\ & Ax = b \\ & x^T x \leq R^2 \end{array}$$

Let $\tilde{\phi}$ denote the logarithmic barrier function for this modified problem. Find $a > 0$ for which $\nabla^2(tf_0(x) + \phi(x)) \succeq a\mathrm{I}$ holds, for all feasible x.

Exercises

Ex 8. Consider the problem (27), with central path $x^*(t)$ for $t > 0$, defined as the solution of (32).
For $u > p^*$, let $z^*(u)$ denote the solution of

$$\begin{array}{ll} \min & -\log(u - f_0(x)) - \sum_{i=1}^m \log(-f_i(x)) \\ \text{s.t.} & Ax = b \end{array}$$

Show that the curve define by $z^*(u)$, for $u > p^*$, is the central path. (In other words, for each $u > p^*$, there is a $t > 0$ for which $x^*(t) = z^*(u)$, and conversely, for each $t > 0$, there is a $u > p^*$ for which $z^*(u) = x^*(t)$).

Outline

Sparse Optimization

Many problems of recent interest in statistics and related areas can be posed in the framework of sparse optimization. Due to the explosion in size and complexity of modern data analysis (BigData), it is increasingly important to be able to solve problems with a very large number of features, training examples, or both.

Outline

Big data era

Electronic commerce data

Social network data

Financial data

Multimedia data

Bioinformatics data

Geometric data

...

High-dimensional data and sparsity

Techniques:

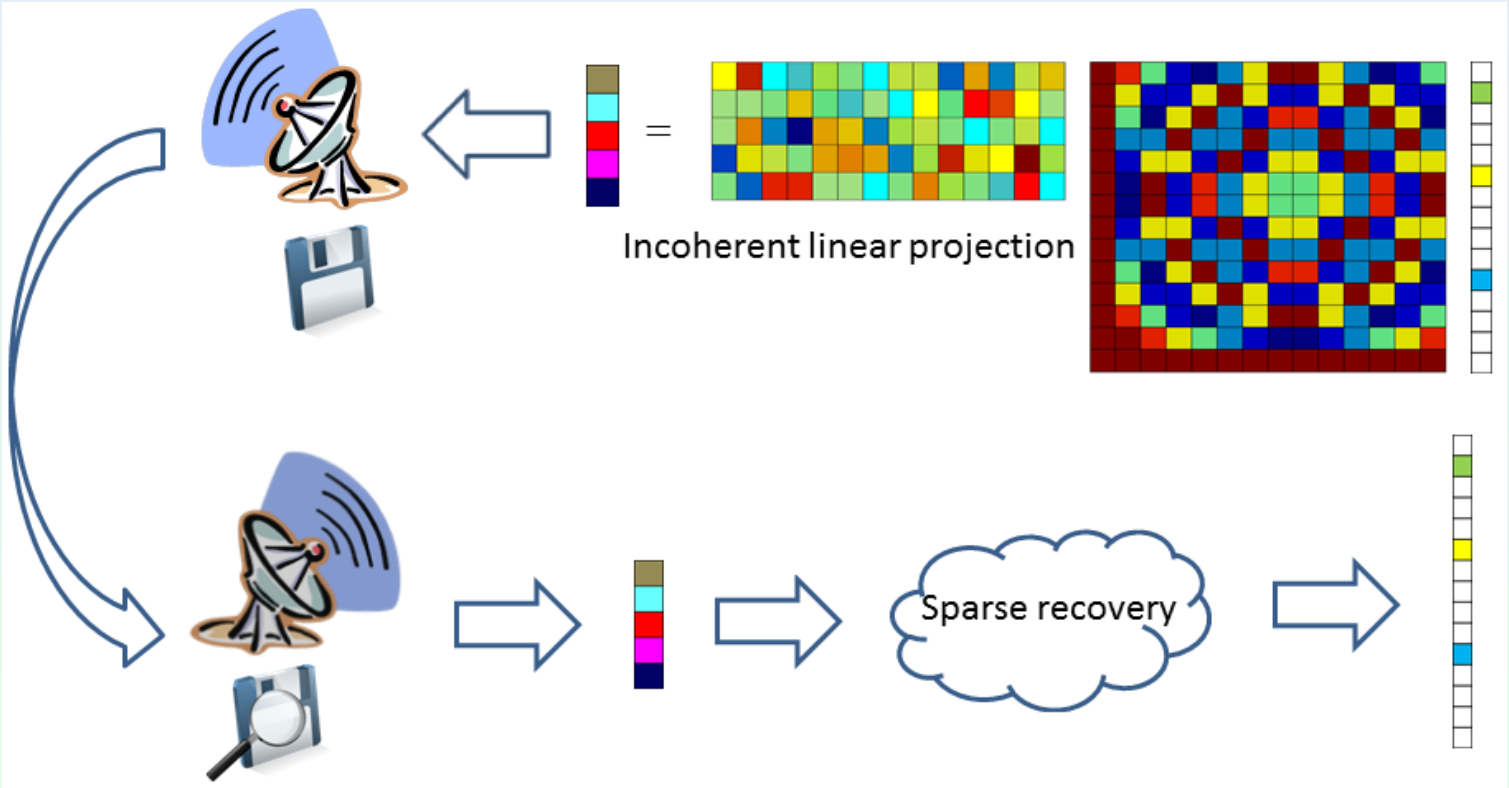
- Statistics (Bayesian/Lasso)
- Priors and Transforms
- Sparse and Redundant Representations
- Low Rank Representations
- Optimization (OMP/BP)
- ...

Compressed Sensing

In recent years, Compressed Sensing (CS) has attracted considerable attention in areas of applied mathematics, computer science, and signal processing [Candes and Tao 2005; Donoho 2006; Bruckstein et al. 2009].

Compressed Sensing

The central insight of CS is that many signals are sparse, i.e., represented using only a few non-zero coefficients in a suitable basis or dictionary and such signals can be recovered from very few measurements (undersampled data) by an optimization algorithm.



The Sparsest Solution of $A\mathbf{x} = \mathbf{b}$

$$(P_0) \quad \min_{\mathbf{x}} \|\mathbf{x}\|_0 \quad \text{s.t.} \quad A\mathbf{x} = \mathbf{b}. \tag{43}$$

For the underdetermined linear system of equations $A\mathbf{x} = \mathbf{b}$ (a full-rank matrix $A \in \mathbb{R}^{m\times n}$ with $m << n$), the following questions are posed:

Q1: When can uniqueness of the sparsest solution be claimed?

Q2: Can a candidate solution be tested to verify its (global) optimality?

Q3: Can the solution be reliably and efficiently found in practice?

Q4: What performance guarantees can be given for various approximate and practical solvers?

Uniqueness via the Spark

- **Definition 01:** The spark of a given matrix A is the smallest number of columns from A that are linearly dependent.

- **Theorem 02:** If a system of linear equations $A\mathbf{x} = \mathbf{b}$ has a solution $\mathbf{x}$ obeying $\|\mathbf{x}\|_0 < spark(A)/2$, this solution is necessarily the sparsest possible.

Uniqueness via the Mutual Coherence

- **Definition 03:** The mutual coherence of a given matrix A is the largest absolute normalized inner product between different columns from A. Denoting the k-th column in A by $\mathbf{a}_k$, the mutual coherence is given by

$$\mu(A) = \max_{1 \leq i \neq j \leq n} \frac{|\mathbf{a}_i^T \mathbf{a}_j|}{\|\mathbf{a}_i\|_2 \|\mathbf{a}_j\|_2}.$$

- **Lemma 04:** For any matrix $A \in \mathbb{R}^{m \times n}$, the following relationship holds:

$$spark(A) \geqslant 1 + \frac{1}{\mu(A)}.$$

- **Theorem 05:** If a system of linear equations $A\mathbf{x} = \mathbf{b}$ has a solution $\mathbf{x}$ obeying $\|\mathbf{x}\|_0 < (1 + 1/\mu(A))/2$, this solution is necessarily the sparsest possible.

Pursuit Algorithms

Greedy strategies are usually adopted in solving the problem (P_0).

The following algorithm is known in the literature of signal processing by the name *Orthogonal Matching Pursuit* (OMP).

Task: Approximate the solution of (P_0): $\min_{\mathbf{x}} \|\mathbf{x}\|_0$ subject to $\mathbf{A}\mathbf{x} = \mathbf{b}$.

Parameters: We are given the matrix $\mathbf{A}$, the vector $\mathbf{b}$, and the error threshold ϵ_0.

Initialization: Initialize $k = 0$, and set

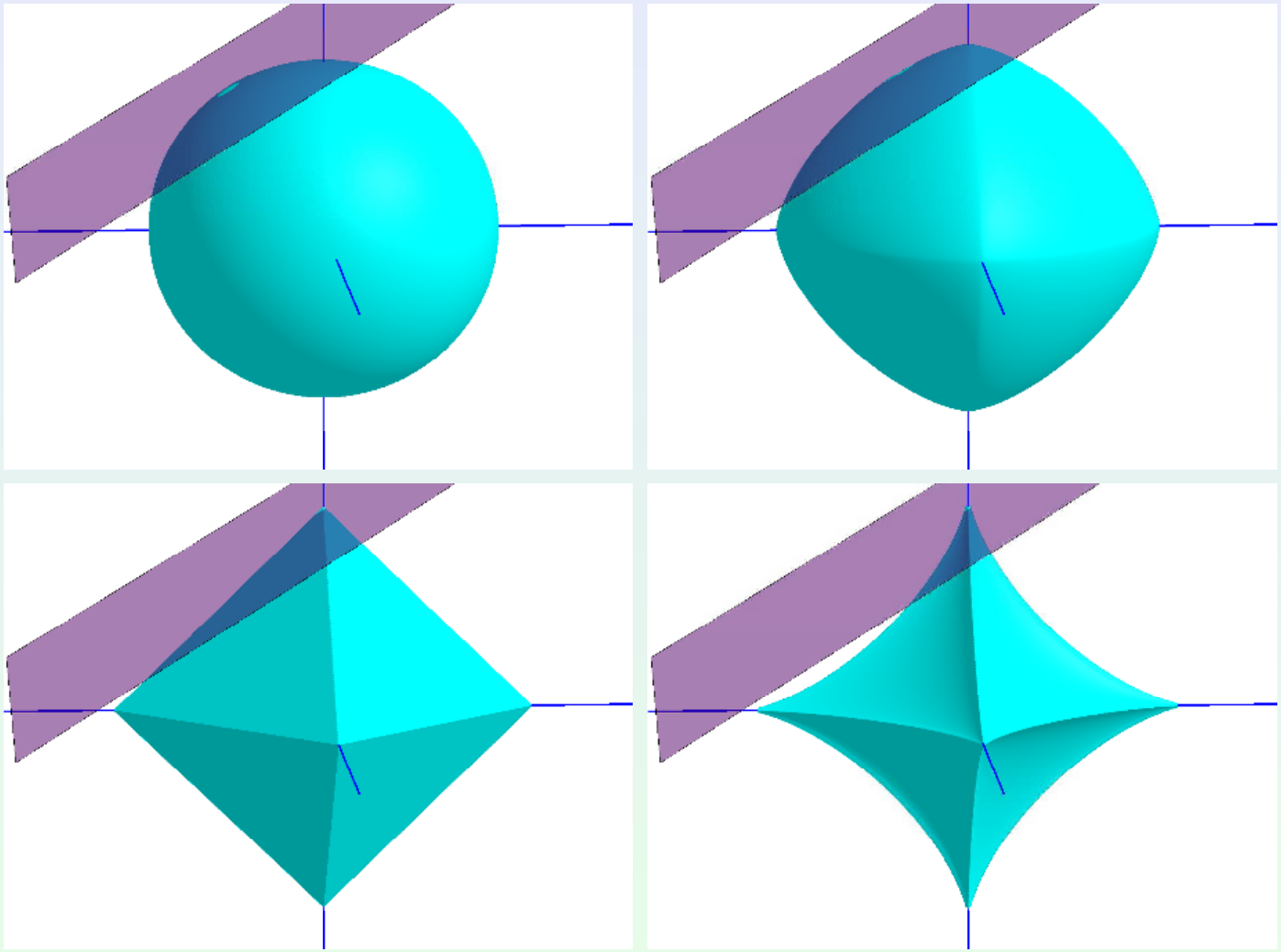
- The initial solution $\mathbf{x}^0 = 0$.
- The initial residual $\mathbf{r}^0 = \mathbf{b} - \mathbf{A}\mathbf{x}^0 = \mathbf{b}$.
- The initial solution support $\mathcal{S}^0 = Support\{\mathbf{x}^0\} = \emptyset$.

Main Iteration: Increment k by 1 and perform the following steps:

- **Sweep:** Compute the errors $\epsilon(j) = \min_{z_j} \|\mathbf{a}_j z_j - \mathbf{r}^{k-1}\|_2^2$ for all j using the optimal choice $z_j^* = \mathbf{a}_j^T \mathbf{r}^{k-1} / \|\mathbf{a}_j\|_2^2$.
- **Update Support:** Find a minimizer j_0 of $\epsilon(j)$: $\forall\ j \notin \mathcal{S}^{k-1},\ \epsilon(j_0) \leq \epsilon(j)$, and update $\mathcal{S}^k = \mathcal{S}^{k-1} \cup \{j_0\}$.
- **Update Provisional Solution:** Compute $\mathbf{x}^k$, the minimizer of $\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2$ subject to $Support\{\mathbf{x}\} = \mathcal{S}^k$.
- **Update Residual:** Compute $\mathbf{r}^k = \mathbf{b} - \mathbf{A}\mathbf{x}^k$.
- **Stopping Rule:** If $\|\mathbf{r}^k\|_2 < \epsilon_0$, stop. Otherwise, apply another iteration.

Output: The proposed solution is $\mathbf{x}^k$ obtained after k iterations.

Geometry of ℓ_p-Norm



Pursuit Algorithms

Convex relaxation technique is a second way to render (P_0) more tractable.

Convexifying with the ℓ_1 norm, we come to the new optimization problem

$$(P_1) \quad \min_{\mathbf{x}} \|W\mathbf{x}\|_1 \quad \text{s.t.} \quad A\mathbf{x} = \mathbf{b} \tag{44}$$

where W is a diagonal positive-definite matrix that introduces the precompensating weights.

It was named *Basis Pursuit* (BP) when all the columns of A are normalized (and thus $W = I$).

Pursuit Algorithms

- **Theorem 06:** For a system of linear equations $A\mathbf{x} = \mathbf{b}$, if a solution $\mathbf{x}$ exists obeying $\|\mathbf{x}\|_0 < (1 + 1/\mu(A))/2$, then an OMP algorithm run with threshold parameter $\epsilon_0 = 0$ is guaranteed to find it exactly.

- **Theorem 07:** For a system of linear equations $A\mathbf{x} = \mathbf{b}$, if a solution $\mathbf{x}$ exists obeying $\|\mathbf{x}\|_0 < (1 + 1/\mu(A))/2$, that solution is both the unique solution of (P_1) and the unique solution of (P_0).

From Exact to Approximate Solutions

An error-tolerant version of (P_0) is defined by

$$(P_0^\epsilon) \quad \min_{\mathbf{x}} \|\mathbf{x}\|_0 \quad \text{s.t.} \quad \|\mathbf{b} - A\mathbf{x}\| \leqslant \epsilon. \tag{45}$$

- **Theorem 08:** Consider the instance of problem (P_0^ϵ) defined by the triplet $(A; \mathbf{b}; \epsilon)$. Suppose that a sparse vector $\mathbf{x}_0$ satisfies the sparsity constraint $\|\mathbf{x}_0\|_0 < (1 + 1/\mu(A))/2$, and gives a representation of $\mathbf{b}$ to within error tolerance ϵ (i.e., $\|\mathbf{b} - A\mathbf{x}_0\| \leqslant \epsilon$). Every solution $\mathbf{x}_0^\epsilon$ of (P_0^ϵ) must obey

$$\|\mathbf{x}_0^\epsilon - \mathbf{x}_0\|_2^2 \leqslant \frac{4\epsilon^2}{1 - \mu(A)(2\|\mathbf{x}_0\|_0 - 1)}.$$

From Exact to Approximate Solutions

An error-tolerant version of (P_1) is defined by

$$(P_1^\epsilon) \quad \min_{\mathbf{x}} \|\mathbf{x}\|_1 \quad \text{s.t.} \quad \|\mathbf{b} - A\mathbf{x}\| \leqslant \epsilon. \tag{46}$$

- **Theorem 09:** Consider the instance of problem (P_1^ϵ) defined by the triplet $(A; \mathbf{b}; \epsilon)$. Suppose that a sparse vector $\mathbf{x}_0$ is a feasible solution to (P_1^ϵ) satisfying the sparsity constraint $\|\mathbf{x}_0\|_0 < (1 + 1/\mu(A))/4$. The solution $\mathbf{x}_1^\epsilon$ of (P_1^ϵ) must obey

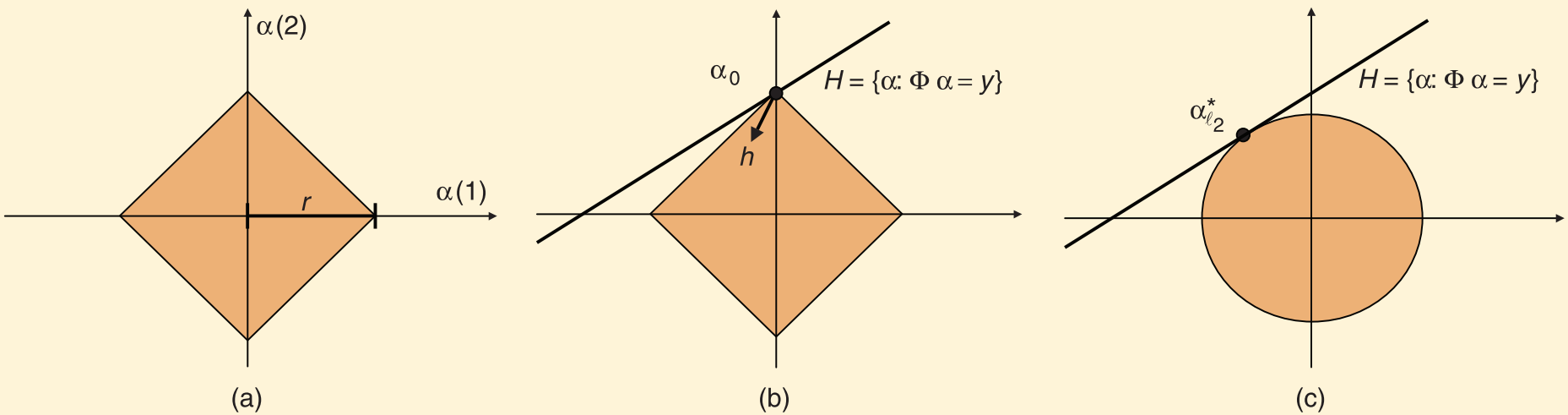
$$\|\mathbf{x}_1^\epsilon - \mathbf{x}_0\|_2^2 \leqslant \frac{4\epsilon^2}{1 - \mu(A)(4\|\mathbf{x}_0\|_0 - 1)}.$$

Restricted Isometry Property

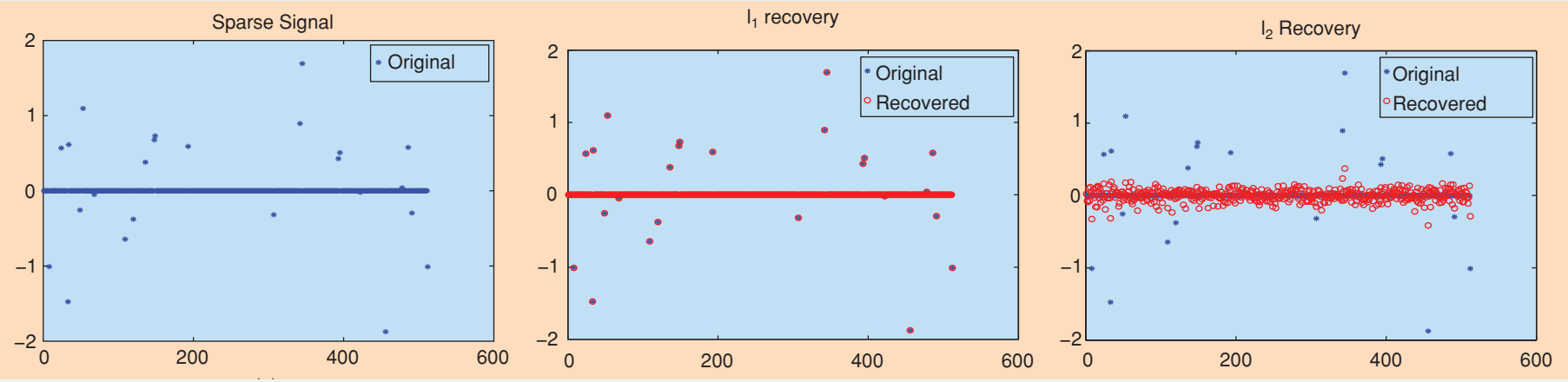
- **Definition 10:** A matrix $A \in \mathbb{R}^{m\times n}$ is said to have the restricted isometry property $RIP(\delta; s)$ if each submatrix A_s formed by combining at most s columns of A has its nonzero singular values bounded above by $1+\delta$ and below by $1-\delta$.

- **Theorem 11:** Candès and Tao have shown that $A \in RIP(\sqrt{2}-1; 2s)$ implies that (P_1) and (P_0) have identical solutions on all s-sparse vectors and, moreover, that (P_1^{ϵ}) stably approximates the sparsest near-solution of $\mathbf{b} = A\mathbf{x} + \mathbf{v}$ with a reasonable stability coefficient.

ℓ_1 and ℓ_2 Recovery



ℓ_1 and ℓ_2 Recovery



Compressed Sensing

Designing measurement/sensing matrices with favorable properties and constructing suitable transforms/dictionaries are the important research topics in Compressed Sensing.

Outline

Sparse Modeling

All the previous theorems have shown us that the problem of finding a sparse solution to an under-determined linear system (or approximation of it) can be given a meaningful definition and can also be computationally tractable.

We now turn to discuss the applicability of these ideas to signal, image, and geometric processing, i.e., sparsity-seeking representations.

Priors and transforms for signals

The Bayesian framework imposes a Probability-Density-Function (PDF) on the signals – a prior distribution $P(\mathbf{y})$.

Priors are extensively used in signal processing, serving in inverse problems, compression, anomaly detection, and more.

Priors and transforms for signals

Consider the denoising problem: a given image $\mathbf{b}$ is known to be a noisy version of a clean image $\mathbf{y}$, contaminated by an additive perturbation vector $\mathbf{v}$, known to have a finite energy $\|\mathbf{v}\|_2 \leqslant \epsilon$, i.e., $\mathbf{b} = \mathbf{y} + \mathbf{v}$.

The optimization problem

$$\max_{\mathbf{y}} P(\mathbf{y}) \quad \text{s.t.} \quad \|\mathbf{y} - \mathbf{b}\|_2 \leqslant \epsilon$$

leads to the most probable image $\hat{\mathbf{y}}$ that is an effective estimate of $\mathbf{y}$.

This way the prior is exploited for solving the noise cleaning problem. The above formulation of the denoising problem is in fact that Maximum-A-posteriori-Probability (MAP) estimator.

Priors and transforms for signals

Much effort has been allocated in the signal and image processing communities for forming priors as closed-form expressions.

One very common way to construct $P(\mathbf{y})$ is to guess its structure based on intuitive expectations from the data content. For example, the Gibbs distribution $P(\mathbf{y}) = Const \cdot exp\{-\lambda \|L\mathbf{y}\|_2^2\}$ uses a Laplacian matrix to give an evaluation of the probability of the image $\mathbf{y}$.

In such a prior, smoothness, measured by the Laplacian operator, is used for judging the probability of the signal.

Priors and transforms for signals

This prior is well-known and extensively used in signal processing, and is known to be related to both Tikhonov regularization and Wiener filtering.

The prior leads to an optimization problem of the form

$$\min \|L\mathbf{y}\|_2^2 \quad \text{s.t.} \quad \|\mathbf{y} - \mathbf{b}\|_2 \leqslant \epsilon$$

which can be converted to

$$\min \lambda\|L\mathbf{y}\|_2^2 + \|\mathbf{y} - \mathbf{b}\|_2^2$$

where we have replaced the constraint by an equivalent penalty.

The closed-form solution is easily obtained as

$$\hat{\mathbf{y}} = (I + \lambda L^T L)^{-1}\mathbf{b}.$$

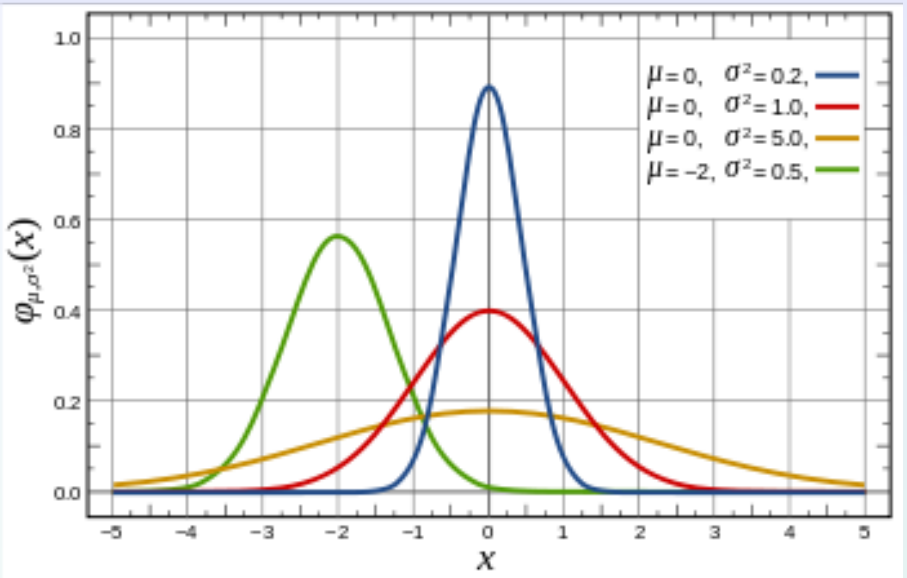
Priors and transforms for signals

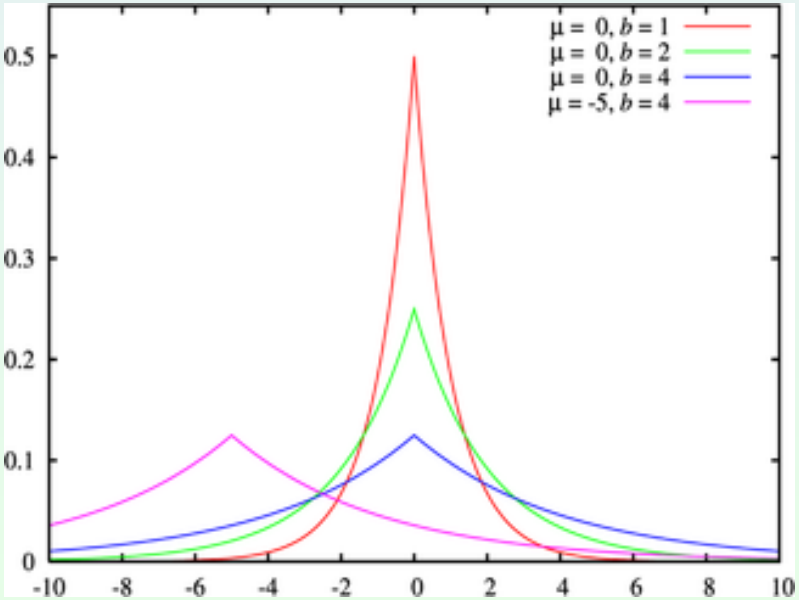
The above specific prior stressing smoothness is known to cause blurring of the image when used in various restoration tasks. The remedy for this problem was found to be the replacement of the ℓ_2-norm by a more robust measure, such as an ℓ_1-norm, that allows heavy tails for the distribution of the values of $L\mathbf{y}$.

A prior of the form $P(\mathbf{y}) = Const \cdot exp\{-\lambda \|L\mathbf{y}\|_1\}$ is far more versatile and thus became popular in recent years.

Similar to this option is the Total-Variation (TV) prior $P(\mathbf{y}) = Const \cdot exp\{-\lambda \|\mathbf{y}\|_{TV}\}$ [Rudin, Osher, and Fatemi, 1993] that also promotes smoothness, but differently, by replacing the Laplacian with gradient norms.

Priors and transforms for signals





Priors and transforms for signals

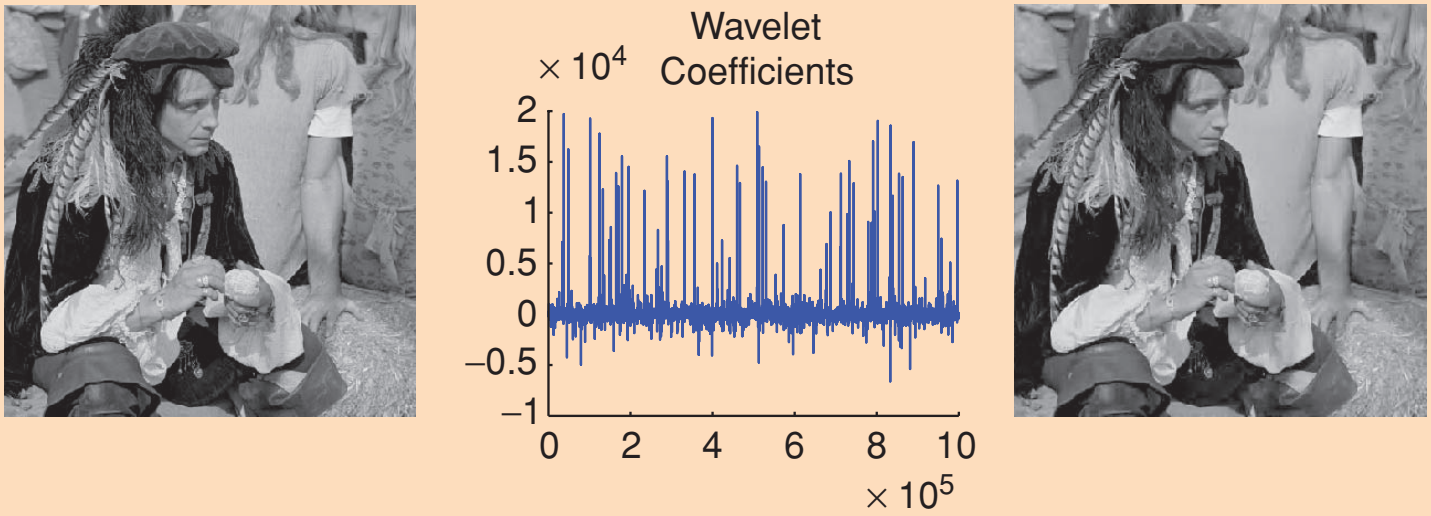
A different property that can be used for constructing a prior is assuming a structure on the signals transform-coefficients.

One such example is the JPEG compression algorithm, which relies on the fact that 2D-DCT coefficients of small image patches tend to behave in a predicted way (being concentrated around the origin).

Another well-known example refers to the wavelet transform of signals and images, where the coefficients are expected to be sparse, most of them tending to zero while few remain active.

Priors and transforms for signals

For a signal $\mathbf{y}$, the wavelet transform is given by $T\mathbf{y}$ where the matrix T is a specially designed orthogonal matrix that contains in its rows spatial derivatives of varying scale, thereby providing what is known as multi-scale analysis of the signal.



Therefore, the prior in this case becomes
$P(\mathbf{y}) = Const \cdot exp\{-\lambda \| T\mathbf{y} \|_p^p\}$ with $p \leqslant 1$ to promote sparsity.

Priors and transforms for signals

A rich family of signal priors assign likelihood for an image based on the behavior of its transform coefficients $T\mathbf{y}$. In the signal and image processing literature, such priors were postulated in conjunction with a variety of transforms, such as

- the Discrete-Fourier-Transform (DFT)
- the Discrete-Cosine-Transform (DCT)
- the Hadamard-Transform (HT)
- the Principal-Component-Analysis (PCA)

Bayesian perspective

- Using Bayes' rule, the posterior probability of $\mathbf{y}$ given the measurements is formulated by

$$P(\mathbf{y} \mid \mathbf{z}) = \frac{P(\mathbf{z} \mid \mathbf{y})P(\mathbf{y})}{P(\mathbf{z})}.$$

- Considering the fact that the denominator $P(\mathbf{z})$ is not a function of the unknown $\mathbf{y}$, and as such it can be disregarded, the MAP estimation amounts to

$$\hat{\mathbf{y}}_{\text{MAP}} = \arg\max_{\mathbf{y}} P(\mathbf{y} \mid \mathbf{z}) = \arg\max_{\mathbf{y}} P(\mathbf{z} \mid \mathbf{y})P(\mathbf{y}).$$

- The probability $P(\mathbf{z} \mid \mathbf{y})$ is known as the likelihood function, and the probability $P(\mathbf{y})$ is the known/unknown's prior.

The Sparse-Land model

The linear system $D\mathbf{x} = \mathbf{y}$ can be interpreted as a way of constructing signals $\mathbf{y}$. Every column in D is a possible signal in $\mathbb{R}^n$ – we refer to these m columns as atomic signals, and the matrix D displays a dictionary of atoms.

The multiplication of D by a sparse vector x with $\|\mathbf{x}\|_0 = k_0 \ll n$ produces a linear combination of k_0 atoms with varying portions, generating the signal $\mathbf{y}$. The vector $\mathbf{x}$ that generates $\mathbf{y}$ will be called its representation.

The Sparse-Land model

The Sparse-Land model $\mathcal{M}(D, k_0, \alpha, \epsilon)$:

$$\mathbf{y} = D\mathbf{x} + \mathbf{v}$$

- Consider all the possible sparse representation vectors with cardinality $\|\mathbf{x}\|_0 = k_0 \ll n$, and assume that this set of $C_m^{k_0}$ possible cardinalities are drawn with uniform probability.
- Assume further that the non-zero entries in $\mathbf{x}$ are drawn from the zero-mean Gaussian distribution $Const \cdot exp\{-\alpha x_i^2\}$.
- Postulate that the observations are contaminated by a random perturbation (noise) vector $\mathbf{v} \in \mathbb{R}^n$ with bounded power $\|\mathbf{v}\|_2 \leqslant \epsilon$.

The quest for a dictionary

In the quest for the proper dictionary to use in applications, one line of work considers choosing pre-constructed dictionaries, such as undecimated wavelets, steerable wavelets, contourlets, curvelets, and more.

Some of these proposed dictionaries (which are often referred to also as transforms) are accompanied by a detailed theoretical analysis establishing the sparsity of the representation coefficients for such simplified content of signals.

The quest for a dictionary

While pre-constructed or adapted dictionaries typically lead to fast transforms, they are typically limited in their ability to sparsify the signals they are designed to handle. Furthermore, most of those dictionaries are restricted to signals/images of a certain type, and cannot be used for a new and arbitrary family of signals of interest.

This leads us to yet another approach for obtaining dictionaries that overcomes these limitations – by adopting a learning point-of-view.

As opposed to the pre-constructed and adapted dictionaries, the learning method is able to adapt to any family of signals that complies with the Sparse-Land model.

Dictionary learning

Assume that a training database $\{\mathbf{y}_j\}_{j=1}^N$ is given, and thought to have been generated by some fixed but unknown model $\mathcal{M}(D, k_0, \alpha, \epsilon)$.

- Control the deviation:

$$\min_{D,\{\mathbf{x}_j\}_{j=1}^N} \sum_{j=1}^N \|\mathbf{x}_j\|_0 \quad \text{s.t.} \quad \|\mathbf{y}_j - D\mathbf{x}_j\|_2 \leqslant \epsilon,\ j = 1, \cdots, N$$

- Control the sparsity:

$$\min_{D,\{\mathbf{x}_j\}_{j=1}^N} \sum_{j=1}^N \|\mathbf{y}_j - D\mathbf{x}_j\|_2^2 \quad \text{s.t.} \quad \|\mathbf{x}_j\|_0 \leqslant k_0,\ j = 1, \cdots, N$$

Analysis versus Synthesis

- Synthesis based modeling

$$(P_s) \quad \hat{\mathbf{y}}_s = D \cdot \arg\min_{\mathbf{x}} \|\mathbf{x}\|_p \quad \text{s.t.} \quad \|\mathbf{z} - D\mathbf{x}\| \leqslant \epsilon. \tag{47}$$

- Analysis based modeling

$$(P_a) \quad \hat{\mathbf{y}}_a = \arg\min_{\mathbf{y}} \|T\mathbf{y}\|_p \quad \text{s.t.} \quad \|\mathbf{z} - \mathbf{y}\| \leqslant \epsilon. \tag{48}$$

Dictionary learning algorithms

The optimization problem for dictionary learning (sparse representations and coding):

$$\min_{D,X} \|Y - DX\|_{Frob} \quad \text{s.t.} \quad \|\mathbf{x}_j\|_0 \leqslant k_0,\ j = 1, \cdots, N \tag{49}$$

where

$$Y = (\mathbf{y}_1, \cdots, \mathbf{y}_N) \in \mathbb{R}^{n \times N},$$

$$D = (\mathbf{d}_1, \cdots, \mathbf{d}_m) \in \mathbb{R}^{n \times m},$$

$$X = (\mathbf{x}_1, \cdots, \mathbf{x}_N) \in \mathbb{R}^{m \times N}.$$

Dictionary learning algorithms

There are two training mechanisms, the first named Method of Optimal Directions (MOD) by Engan et al., and the second named K-SVD, by Aharon et al..

- MOD
- K-SVD
- .....

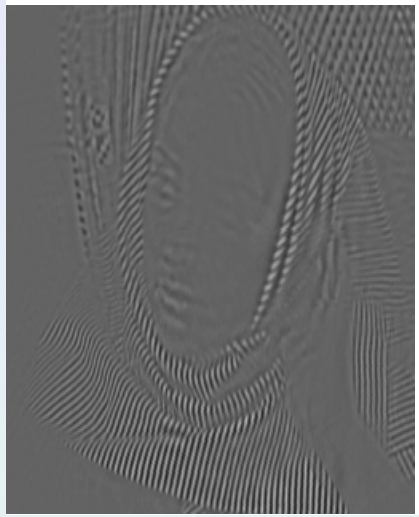
Applications

- Image deblurring
- Facial image compression
- Image denoising
- Image inpainting
- .....

Applications

Original image

Cartoon part



Texture part

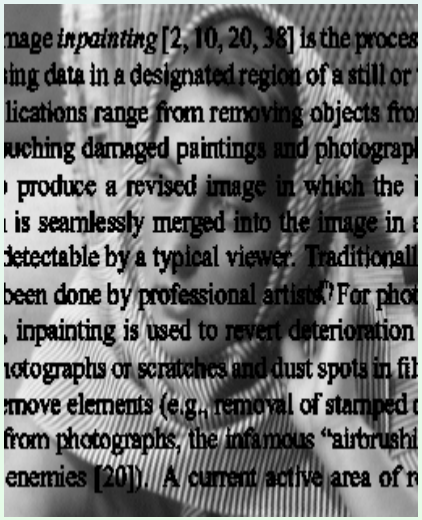
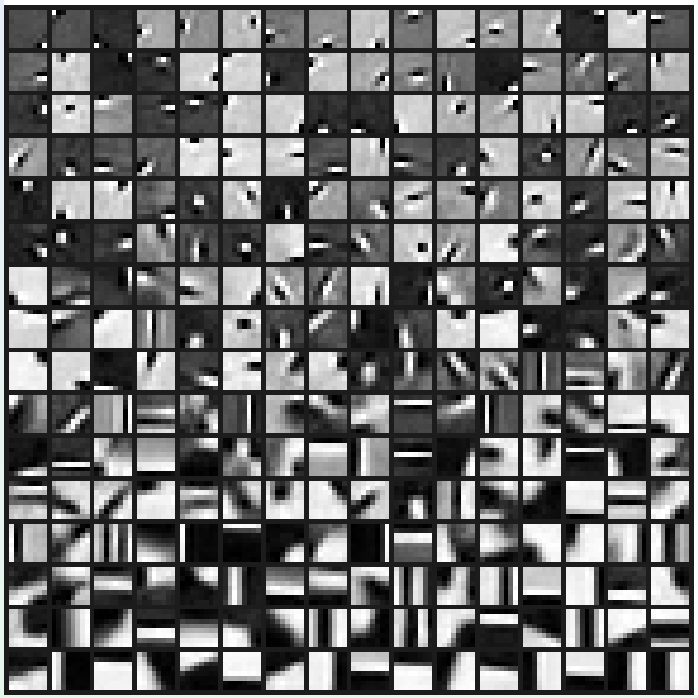
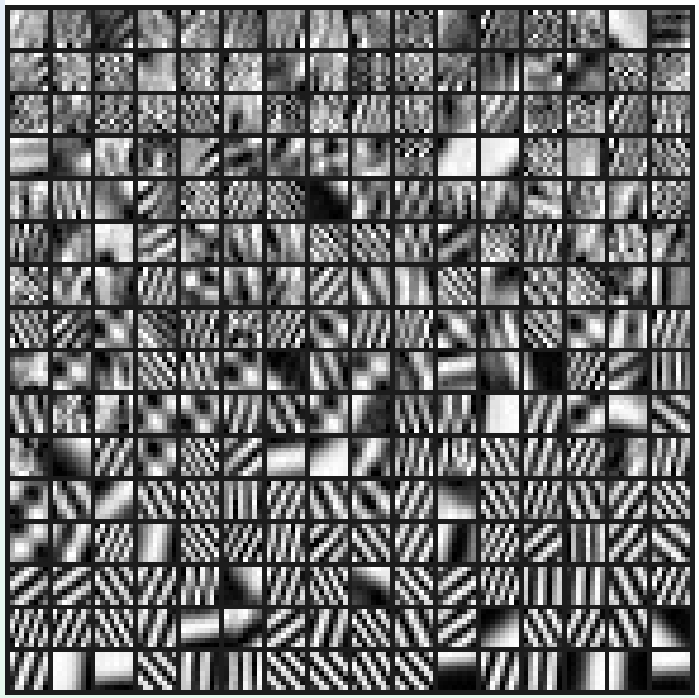


Image with missing data

Inpainting result

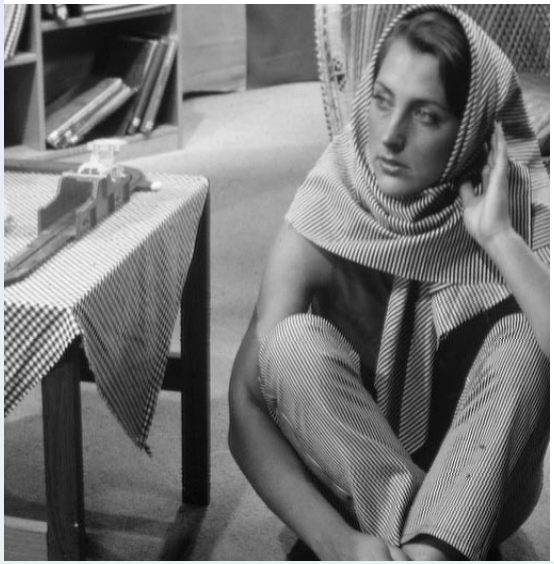
Applications



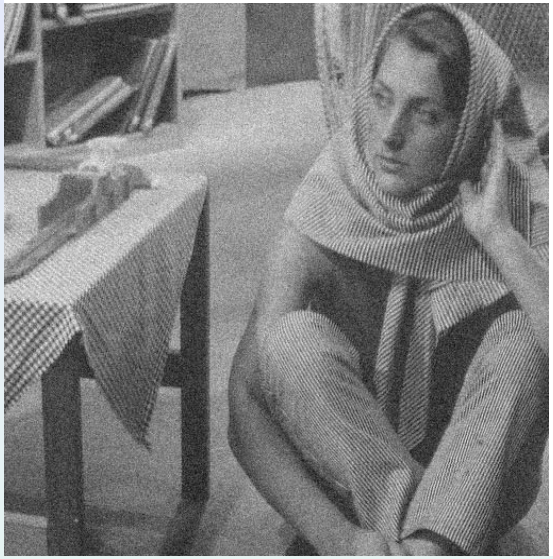


Applications

Original Image



Noisy Image (22.1307 dB, σ=20)



Denoised Image Using
Global Trained Dictionary (28.8528 dB)

Denoised Image Using
Adaptive Dictionary (30.8295 dB)

References

1. Emmanuel J. Candès, Terence Tao. Decoding by linear programming. Information Theory, IEEE Transactions on 51.12 (2005): 4203-4215.
2. Emmanuel J. Candès, Justin Romberg, Terence Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. Information Theory, IEEE Transactions on 52.2 (2006): 489-509.
3. Emmanuel J. Candès, Michael B. Wakin. An introduction to compressive sampling. Signal Processing Magazine, IEEE 25.2 (2008): 21-30.
4. Alfred M. Bruckstein, David L. Donoho, Michael Elad. From sparse solutions of systems of equations to sparse modeling of signals and images. SIAM review 51.1 (2009): 34-81.
5. Michael Elad. Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing. Springer, 2010.

Outline

0-norm optimization

$$(P_0) \quad \min_{\mathbf{x}} \|\mathbf{x}\|_0 \quad \text{s.t.} \quad A\mathbf{x} = \mathbf{b}. \tag{50}$$

$$(P_0^\epsilon) \quad \min_{\mathbf{x}} \|\mathbf{x}\|_0 \quad \text{s.t.} \quad \|\mathbf{b} - A\mathbf{x}\| \leqslant \epsilon. \tag{51}$$

Greedy algorithms

Greedy strategies are usually adopted in solving the 0-norm problems. The following algorithm is known in the literature of signal processing by the name *Orthogonal Matching Pursuit* (OMP).

Task: Approximate the solution of (P_0): $\min_{\mathbf{x}} \|\mathbf{x}\|_0$ subject to $\mathbf{A}\mathbf{x} = \mathbf{b}$.

Parameters: We are given the matrix $\mathbf{A}$, the vector $\mathbf{b}$, and the error threshold ϵ_0.

Initialization: Initialize $k = 0$, and set

- The initial solution $\mathbf{x}^0 = 0$.
- The initial residual $\mathbf{r}^0 = \mathbf{b} - \mathbf{A}\mathbf{x}^0 = \mathbf{b}$.
- The initial solution support $\mathcal{S}^0 = Support\{\mathbf{x}^0\} = \emptyset$.

Main Iteration: Increment k by 1 and perform the following steps:

- **Sweep:** Compute the errors $\epsilon(j) = \min_{z_j} \|\mathbf{a}_j z_j - \mathbf{r}^{k-1}\|_2^2$ for all j using the optimal choice $z_j^* = \mathbf{a}_j^T \mathbf{r}^{k-1} / \|\mathbf{a}_j\|_2^2$.
- **Update Support:** Find a minimizer j_0 of $\epsilon(j)$: $\forall\ j \notin \mathcal{S}^{k-1},\ \epsilon(j_0) \leq \epsilon(j)$, and update $\mathcal{S}^k = \mathcal{S}^{k-1} \cup \{j_0\}$.
- **Update Provisional Solution:** Compute $\mathbf{x}^k$, the minimizer of $\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2$ subject to $Support\{\mathbf{x}\} = \mathcal{S}^k$.
- **Update Residual:** Compute $\mathbf{r}^k = \mathbf{b} - \mathbf{A}\mathbf{x}^k$.
- **Stopping Rule:** If $\|\mathbf{r}^k\|_2 < \epsilon_0$, stop. Otherwise, apply another iteration.

Output: The proposed solution is $\mathbf{x}^k$ obtained after k iterations.

Dictionary learning

The optimization model of dictionary learning for sparse and redundant representations:

$$\min_{D,X} \|Y - DX\|_{Frob} \quad \text{s.t.} \quad \|\mathbf{x}_j\|_0 \leqslant k_0,\ j = 1, \cdots, N \tag{52}$$

where

$$Y = (\mathbf{y}_1, \cdots, \mathbf{y}_N) \in \mathbb{R}^{n \times N},$$

$$D = (\mathbf{d}_1, \cdots, \mathbf{d}_m) \in \mathbb{R}^{n \times m},$$

$$X = (\mathbf{x}_1, \cdots, \mathbf{x}_N) \in \mathbb{R}^{m \times N}.$$

Dictionary learning

There are two training mechanisms, the first named Method of Optimal Directions (MOD) by Engan et al., and the second named K-SVD, by Aharon et al..

- MOD
- K-SVD
- .....

Convex relaxation

Convex relaxation technique is a way to render 0-norm more tractable.

Convexifying with the ℓ_1 norm, we come to the new optimization problem

$$(P_1) \quad \min_{\mathbf{x}} \|W\mathbf{x}\|_1 \quad \text{s.t.} \quad A\mathbf{x} = \mathbf{b} \tag{53}$$

where W is a diagonal positive-definite matrix that introduces the precompensating weights.

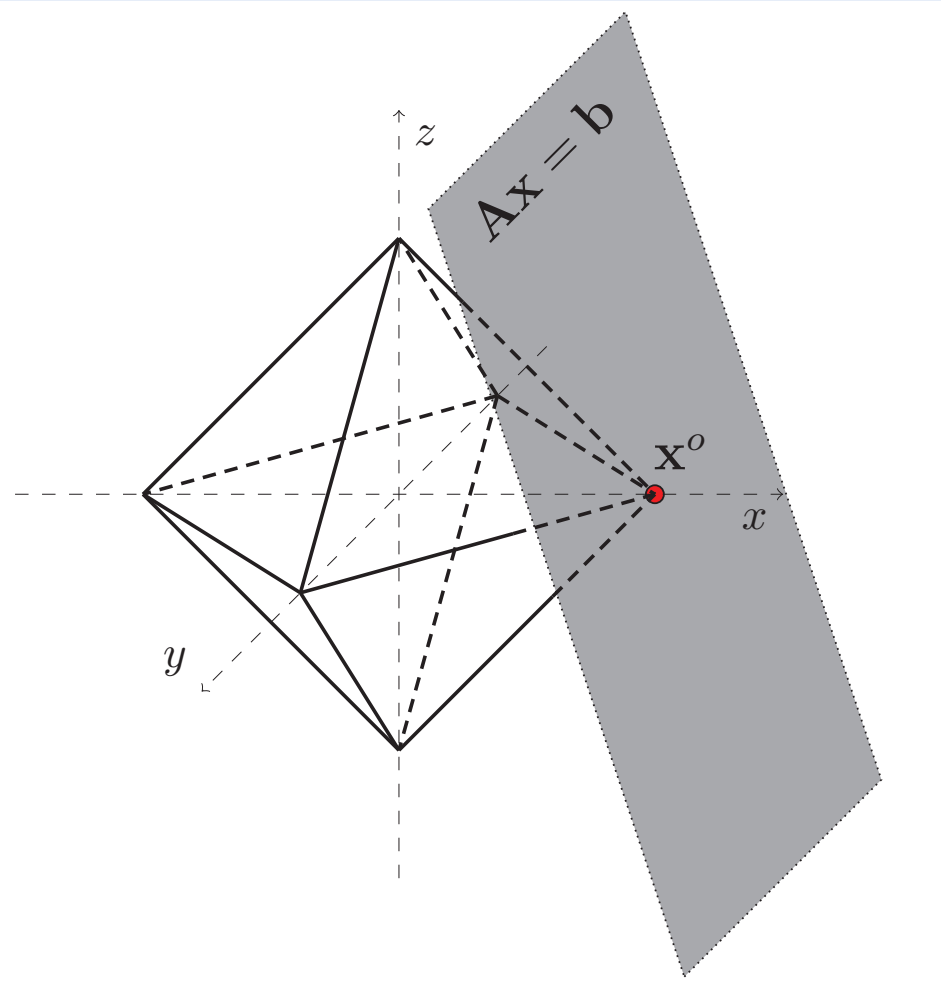
An error-tolerant version of (P_1) is defined by

$$(P_1^{\epsilon}) \quad \min_{\mathbf{x}} \|W\mathbf{x}\|_1 \quad \text{s.t.} \quad \|\mathbf{b} - A\mathbf{x}\| \leqslant \epsilon. \tag{54}$$

It was named *Basis Pursuit* (BP) when all the columns of A are normalized (and thus $W = I$).

Basic pursuit

$$(BP) \quad \min_{\mathbf{x}} \|\mathbf{x}\|_1 \quad \text{s.t.} \quad A\mathbf{x} = \mathbf{b}.$$



BP denoising and LASSO

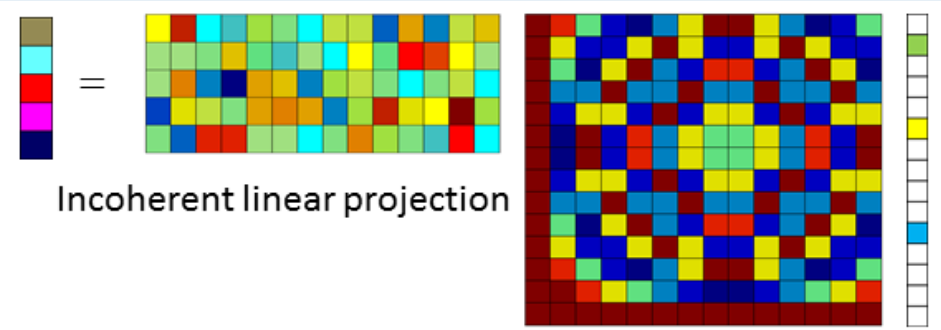
$$
\begin{aligned}
(BP_\tau) \quad & \min_{\mathbf{x}} \|A\mathbf{x}-\mathbf{b}\|_2^2 \quad \text{s.t.} \quad \|\mathbf{x}\|_1 \leqslant \tau, \\
(BP_\mu) \quad & \min_{\mathbf{x}} \|\mathbf{x}\|_1 + \tfrac{\mu}{2}\|A\mathbf{x}-\mathbf{b}\|_2^2, \\
(BP_\delta) \quad & \min_{\mathbf{x}} \|\mathbf{x}\|_1 \quad \text{s.t.} \quad \|A\mathbf{x}-\mathbf{b}\|_2 \leqslant \delta.
\end{aligned}
$$

Questions:

- Are they equivalent? and in what sense?
- How to choose parameters?

Sparse under basis Ψ

$$\min_{\mathbf{s}} \{\|\mathbf{s}\|_1 : A\Psi\mathbf{s} = \mathbf{b}\}$$



If Ψ is orthogonal, the problem is equivalent to

$$\min_{\mathbf{x}} \{\|\Psi^*\mathbf{x}\|_1 : A\mathbf{x} = \mathbf{b}\}.$$

Sparse after transform $\mathcal{L}$

$$\min_{\mathbf{x}}\{\|\mathcal{L}\mathbf{x}\|_1 : A\mathbf{x} = \mathbf{b}\}$$

Examples of $\mathcal{L}$:

- DCT, wavelets, curvelets, ridgelets, ...
- tight frames, Gabor, ...
- total (generalized) variation

Ref: E. J. Cands, Y. Eldar, D. Needell and P. Randall. Compressed sensing with coherent and redundant dictionaries. Applied and Computational Harmonic Analysis, 31(1): 59-73.

Joint/group sparsity

Decompose $\{1, 2, \cdots, n\} = \mathcal{G}_1 \bigcup \mathcal{G}_2 \bigcup \cdots \bigcup \mathcal{G}_S$, and $\mathcal{G}_i \bigcap \mathcal{G}_j = \emptyset, i \neq j$.

Joint/group sparse recovery model:

$$\min_{\mathbf{x}} \{ \|\mathbf{x}\|_{\mathcal{G},2,1} : A\mathbf{x} = \mathbf{b} \}$$

where

$$\|\mathbf{x}\|_{\mathcal{G},2,1} = \sum_{s=1}^{S} w_s \|\mathbf{x}_{\mathcal{G}_s}\|_2.$$

Side constraints

- Nonnegativity: $\mathbf{x} \geq 0$
- Box constraints: $\mathbf{lb} \leq \mathbf{x} \leq \mathbf{ub}$
- Linear inequalities: $Q\mathbf{x} \leq \mathbf{c}$

They generate "corners" and can be very effective in practice.

Shrinkage

- Shrinkage is popular in sparse optimization algorithms
- In optimization, non-smooth functions like ℓ_1 has difficulty using general smooth optimization methods.
- But, ℓ_1 is component-wise separable, so it does get along well with separable (smooth or non-smooth) functions.
- For example,

$$\min_{\mathbf{x}} \|\mathbf{x}\|_1 + \frac{1}{2\tau}\|\mathbf{x} - \mathbf{z}\|_2^2$$

is equivalent to solving $\min_{x_i} |x_i| + \frac{1}{2\tau}|x_i - z_i|^2$ over each i.

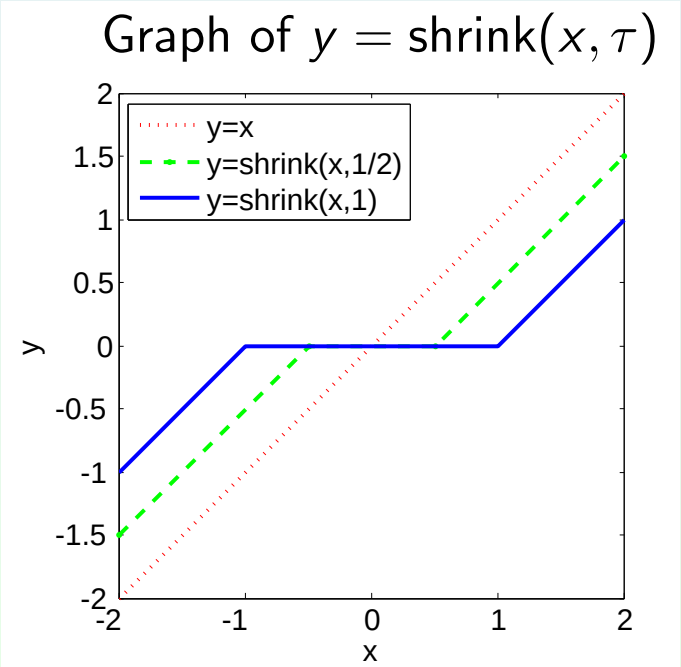
Soft-thresholding shrinkage

The problem is separable and has an explicit solution

$$(\text{shrink}(\mathbf{z},\tau))_i = \begin{cases} z_i - \tau & z_i > \tau, \\ 0 & -\tau \le z_i \le \tau, \\ z_i + \tau & z_i < -\tau. \end{cases}$$

The shrinkage operator can be written in Matlab code as:

y = max(abs(**x**)-tau,0).*sign(**x**)



Soft-thresholding shrinkage

- The following problem is called Moreau-Yosida regularization

$$\min_{\mathbf{x}} r(\mathbf{x}) + \frac{1}{2\tau}\|\mathbf{x}-\mathbf{z}\|_2^2.$$

- For example $r(\mathbf{x}) = \|x\|_2$, the solution to

$$\min_{\mathbf{x}} \|\mathbf{x}\|_2 + \frac{1}{2\tau}\|\mathbf{x}-\mathbf{z}\|_2^2$$

  is, if we treat $0/0 = 0$,

$$\mathbf{x}_{opt} = \max\{\|\mathbf{z}\|_2 - \tau, 0\} \cdot (\mathbf{z}/\|\mathbf{z}\|_2).$$

- Used in joint/group-sparse recovery algorithms.

Soft-thresholding shrinkage

- Consider the following nuclear norm optimization

$$\min_{\mathbf{X}} \|\mathbf{X}\|_* + \frac{1}{2\tau}\|\mathbf{X} - \mathbf{Z}\|_F^2.$$

  Let $\mathbf{Z} = \mathbf{U}\Sigma\mathbf{V}^T$ be the singular value decomposition of $\mathbf{Z}$.

- Let $\hat{\Sigma}$ be the diagonal matrix with diagonal entries

$$\mathrm{diag}(\hat{\Sigma}) = \mathrm{shrink}(\mathrm{diag}(\Sigma), \tau)),$$

  then

$$\mathbf{X}_{opt} = \mathbf{U}\hat{\Sigma}\mathbf{V}^T.$$

- In general, matrix problems with only unitary-invariant functions (e.g., $\|\cdot\|_*$, $\|\cdot\|_F$, spectral norm, trace) and constraints (e.g., positive or negative semi-definiteness) typically reduce to vector problems regarding singular values.

Prox-linear algorithm

Consider the general form

$$\min_{\mathbf{x}} r(\mathbf{x}) + f(\mathbf{x}).$$

where r is the regularization function and f is the data fidelity function.

The prox-linear algorithm is:

$$\mathbf{x}^{k+1} = \arg\min_{\mathbf{x}} r(\mathbf{x}) + f(\mathbf{x}^k) + < \nabla f(\mathbf{x}^k), \mathbf{x} - \mathbf{x}^k > + \frac{1}{2\delta_k}\|\mathbf{x} - \mathbf{x}^k\|_2^2.$$

The last term keeps $\mathbf{x}^{k+1}$ close to $\mathbf{x}^k$, and the parameter δ_k determines the step size. It is equivalent to

$$\mathbf{x}^{k+1} = \arg\min_{\mathbf{x}} r(\mathbf{x}) + \frac{1}{2\delta_k}\|\mathbf{x} - (\mathbf{x}^k - \delta_k \nabla f(\mathbf{x}^k))\|_2^2.$$

Alternating direction method of multipliers (ADMM)

The Alternating Direction Method of Multipliers (ADMM) was developed in the 1970s, with roots in the 1950s, and is equivalent or closely related to many other algorithms, such as dual decomposition, the method of multipliers, Douglas-Rachford splitting, Spingarns method of partial inverses, Dykstras alternating projections, Bregman iterative algorithms for 1-norm problems, proximal methods, and others.

ADMM

The ADMM can be applied to a wide variety of statistical and machine learning problems of recent interest, including the lasso, sparse logistic regression, basis pursuit, covariance selection, support vector machines, and many others.

ADMM

$$\min_{\mathbf{X}\in C^{n\times T}} \mu\|\mathbf{X}\|_p + \|\mathbf{AX}-\mathbf{B}\|_q \tag{55}$$

Let $p := \{2,1\}, q := \{1,1\}$ which denote joint convex norm, we have

$$\min_{\mathbf{X}\in C^{n\times T}} \mu\|\mathbf{X}\|_{2,1} + \|\mathbf{AX}-\mathbf{B}\|_{1,1}$$

where $\|\mathbf{X}\|_{2,1} = \sum_{i=1}^{n}\sqrt{\sum_{j=1}^{T} x_{ij}^2}$, $\|\mathbf{X}\|_{1,1} = \sum_{i=1}^{n}\sum_{j=1}^{T}|x_{ij}|$.

For example $T = 1$,

$$\min_{\mathbf{x}\in C^{n}} \mu\|\mathbf{x}\|_p + \|\mathbf{Ax}-\mathbf{b}\|_q.$$

ADMM

$$\min_{\mathbf{X}\in C^{n\times T}} \mu\|\mathbf{X}\|_p + \|\mathbf{AX}-\mathbf{B}\|_q \tag{55}$$

Let $p := \{2,1\}, q := \{1,1\}$ which denote joint convex norm, we have

$$\min_{\mathbf{X}\in C^{n\times T}} \mu\|\mathbf{X}\|_{2,1} + \|\mathbf{AX}-\mathbf{B}\|_{1,1}$$

where $\|\mathbf{X}\|_{2,1} = \sum_{i=1}^{n}\sqrt{\sum_{j=1}^{T} x_{ij}^2}$, $\|\mathbf{X}\|_{1,1} = \sum_{i=1}^{n}\ \sum_{j=1}^{T}|x_{ij}|$.

For example $T = 1$,

$$\min_{\mathbf{x}\in C^{n}} \mu\|\mathbf{x}\|_p + \|\mathbf{Ax}-\mathbf{b}\|_q.$$

$$\min_{\mathbf{X}\in C^{n\times T}} \mu\|\mathbf{X}\|_p + \|\mathbf{AX}-\mathbf{B}\|_q \tag{55}$$

Let $p := \{2,1\}, q := \{1,1\}$ which denote joint convex norm, we have

$$\min_{\mathbf{X}\in C^{n\times T}} \mu\|\mathbf{X}\|_{2,1} + \|\mathbf{AX}-\mathbf{B}\|_{1,1}$$

where $\|\mathbf{X}\|_{2,1} = \sum_{i=1}^{n}\sqrt{\sum_{j=1}^{T} x_{ij}^2}$, $\|\mathbf{X}\|_{1,1} = \sum_{i=1}^{n}\ \sum_{j=1}^{T}|x_{ij}|$.

For example $T = 1$,

$$\min_{\mathbf{x}\in C^{n}} \mu\|\mathbf{x}\|_p + \|\mathbf{Ax}-\mathbf{b}\|_q.$$

ADMM

$$\begin{aligned} \min\ & \mu\|\mathbf{z}\|_p + \|\mathbf{y}\|_q \\ \text{s.t.}\ & \mathbf{x} - \mathbf{z} = \mathbf{0} \\ & \mathbf{A}\mathbf{x} - \mathbf{y} = \mathbf{b} \end{aligned} \tag{56}$$

$$\begin{aligned} L(\mathbf{x}, \mathbf{y}, \mathbf{z}, \lambda_y, \lambda_z, \rho) =& \mu\|\mathbf{z}\|_p + \|\mathbf{y}\|_q + \mathbf{Re}(\lambda_z^T(\mathbf{x} - \mathbf{z}) + \lambda_y^T(\mathbf{A}\mathbf{x} - \mathbf{y} - \mathbf{b})) \\ & + \frac{\rho}{2}(\|\mathbf{x} - \mathbf{z}\|_2^2 + \|\mathbf{A}\mathbf{x} - \mathbf{y} - \mathbf{b}\|_2^2) \end{aligned} \tag{57}$$

where $\lambda_y \in C^n, \lambda_z \in C^m$ are the Lagrangian multipliers and $\rho > 0$ is a penalty parameter.

ADMM

$$\begin{aligned} \min\ & \mu\|\mathbf{z}\|_p + \|\mathbf{y}\|_q \\ \text{s.t.}\ & \mathbf{x} - \mathbf{z} = \mathbf{0} \\ & \mathbf{A}\mathbf{x} - \mathbf{y} = \mathbf{b} \end{aligned} \tag{56}$$

$$\begin{aligned} L(\mathbf{x}, \mathbf{y}, \mathbf{z}, \lambda_y, \lambda_z, \rho) =& \mu\|\mathbf{z}\|_p + \|\mathbf{y}\|_q + \mathbf{Re}(\lambda_z^T(\mathbf{x} - \mathbf{z}) + \lambda_y^T(\mathbf{A}\mathbf{x} - \mathbf{y} - \mathbf{b})) \\ & + \frac{\rho}{2}(\|\mathbf{x} - \mathbf{z}\|_2^2 + \|\mathbf{A}\mathbf{x} - \mathbf{y} - \mathbf{b}\|_2^2) \end{aligned} \tag{57}$$

where $\lambda_y \in C^n, \lambda_z \in C^m$ are the Lagrangian multipliers and $\rho > 0$ is a penalty parameter.

ADMM

$$\begin{cases} \mathbf{x}^{k+1} := \arg\min \frac{1}{2}(\|\mathbf{x} - \mathbf{z}^k + \mathbf{u}_z^k\|_2^2 + \|\mathbf{A}\mathbf{x} - \mathbf{y}^k - \mathbf{b} + \mathbf{u}_y^k\|_2^2) \\ \mathbf{y}^{k+1} := \arg\min \|\mathbf{y}\|_q + \frac{\rho}{2}\|\mathbf{y} - (\mathbf{A}\mathbf{x}^{k+1} - \mathbf{b}) - \mathbf{u}_y^k\|_2^2 \\ \mathbf{z}^{k+1} := \arg\min \mu\|\mathbf{z}\|_p + \frac{\rho}{2}\|\mathbf{z} - \mathbf{x}^{k+1} - \mathbf{u}_z^k\|_2^2 \end{cases} \tag{58}$$

After solving three subproblems, we update the Lagrangian multipliers as follows:

$$\begin{cases} \mathbf{u}_z^{k+1} = \mathbf{u}_z^k + \gamma(\mathbf{x}^{k+1} - \mathbf{z}^{k+1}) \\ \mathbf{u}_y^{k+1} = \mathbf{u}_y^k + \gamma(\mathbf{A}\mathbf{x}^{k+1} - \mathbf{y}^{k+1} - \mathbf{b}) \end{cases} \tag{59}$$

where $\mathbf{u}_y = \frac{1}{\rho}\lambda_y, \mathbf{u}_z = \frac{1}{\rho}\lambda_z, \gamma > 0$ is the step size.

References I

Thanks for your attention!