EECS 559 Optimization Methods for SIPML

Lecture 2 – Optimization Problem Overview

Instructor: Prof. Qing Qu (qingqu@umich.edu)

Lecture Agenda

- What is an Optimization Problem?
- Linear/Nonlinear Optimization Problems
- Convex/Nonconvex Optimization Problems
- Smooth/Nonsmooth Optimization Problems

What is an Optimization Problem?

A (mathematical) optimization problem is of the form:

$$\boxed{(P) \quad \min_{\boldsymbol{x}} \; f_0(\boldsymbol{x}), \quad \text{s.t.} \;\; f_i(\boldsymbol{x}) \;\le\; b_i, \; i = 1, \cdots, m}$$

- $\boldsymbol{x} \in \mathbb{R}^n$ is the **optimization variable**;
- $f_0(\cdot) : \mathbb{R}^n \mapsto \mathbb{R}$ is the **objective function**;
- The function $f_i(\cdot)$ is a **constraint**.

$f_0(\cdot)$ can be either discrete or continuous. In this course, we focus on continuous optimization.

What is an Optimization Problem?

A (mathematical) optimization problem is of the form:

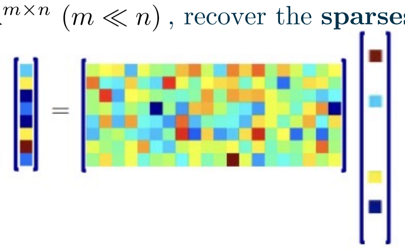
$$\boxed{(P) \quad \min_{\boldsymbol{x}} \; f_0(\boldsymbol{x}), \quad \text{s.t.} \;\; f_i(\boldsymbol{x}) \;\le\; b_i, \; i = 1, \cdots, m}$$

A vector $\boldsymbol{x}_\star$ is a **global solution** to (P), if

- $f_i(\boldsymbol{x}_\star) \le b_i, \quad i = 1, \cdots, m,$
- $f_0(\boldsymbol{x}_\star) \le f_0(\boldsymbol{x}), \quad \forall\, \boldsymbol{x}$ with $f_i(\boldsymbol{x}) \le b_i$

Example: Sparse Recovery

Problem: given measurement $\boldsymbol{y} \in \mathbb{R}^m$ and sensing matrix $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ $(m \ll n)$, recover the **sparsest** $\boldsymbol{x} \in \mathbb{R}^n$



$\boldsymbol{y}$ (observation) $=$ $\boldsymbol{A}$ (measurement matrix) $\cdot$ $\boldsymbol{x}$ (unknown signal)

Example: Sparse Recovery

Problem: given measurement $\boldsymbol{y} \in \mathbb{R}^m$ and sensing matrix $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ $(m \ll n)$, recover the **sparsest** $\boldsymbol{x} \in \mathbb{R}^n$

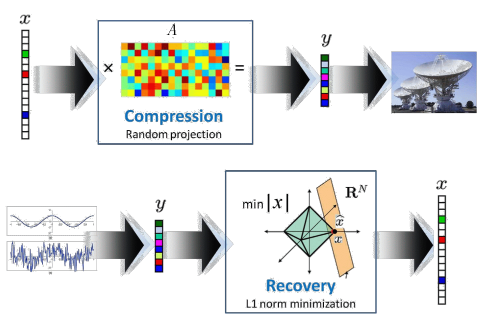
$$\min_{\boldsymbol{z}} \; \|\boldsymbol{z}\|_0, \quad \text{s.t.} \quad \boldsymbol{y} = \boldsymbol{A}\boldsymbol{z}.$$

- $f(\boldsymbol{z}) = \|\boldsymbol{z}\|_0$ is our **objective function**

$$\|\boldsymbol{z}\|_0 := \#\{\, i \mid z(i) \neq 0 \,\} = \sum_i \mathbb{1}_{z(i)\neq 0};$$

- $\boldsymbol{z} \in \mathbb{R}^n$ is the **optimization variable**
- $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{z}$ is the **constraint set**

Application I: Communication



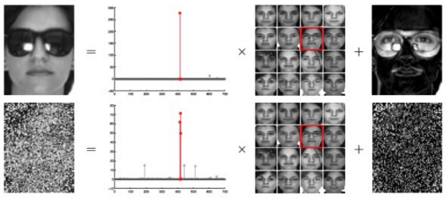
Application II: Image Denoising

$$\boldsymbol{y}_i = \boldsymbol{y}_{i\text{clean}} + \boldsymbol{z}_i = \underset{\text{patch dictionary}}{\boldsymbol{A}} \cdot \underset{\text{sparse coefficient}}{\boldsymbol{x}_i} + \boldsymbol{z}_i$$

noisy image — denoised image — dictionary for image patches

Sparse representation for color image restoration, Mairal, Elad, and Sapiro. TIP, 2008

Application III: Robust Face Recognition



$$\underset{\text{observation}}{\boldsymbol{y}} \;=\; \underset{\text{clean data}}{\boldsymbol{B}\boldsymbol{x}} \;+\; \underset{\text{sparse error}}{\boldsymbol{e}}$$

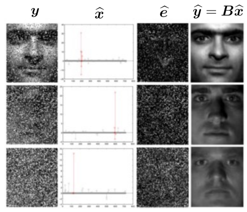
Concatenate gallery images of K subjects into a large "dictionary":

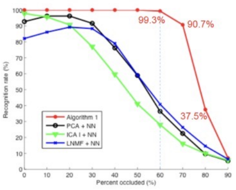
$$\boldsymbol{B} = \begin{bmatrix} \boldsymbol{B}_1 & \boldsymbol{B}_2 & \cdots & \boldsymbol{B}_K \end{bmatrix} \in \mathbb{R}^{m \times n}$$

Application III: Robust Face Recognition

Extended Yale B Database (38 subjects)

Training: subsets 1 and 2 (717 images)
Testing: subset 3 (453 images)





Find the sparsest solution $(\boldsymbol{x}, \boldsymbol{e})$ to the linear system

$$\boldsymbol{y} \;=\; \boldsymbol{B}\boldsymbol{x} + \boldsymbol{e} \;=\; \underbrace{\begin{bmatrix} \boldsymbol{B} & \boldsymbol{I} \end{bmatrix}}_{\boldsymbol{A}} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{e} \end{bmatrix}$$

Example: Sparse Recovery

Problem: given measurement $\boldsymbol{y} \in \mathbb{R}^m$ and sensing matrix $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ $(m \ll n)$, recover the **sparsest** $\boldsymbol{x} \in \mathbb{R}^n$

$$\min_{\boldsymbol{z}} \; \|\boldsymbol{z}\|_0 \,, \quad \text{s.t.} \quad \boldsymbol{y} \;=\; \boldsymbol{A}\boldsymbol{z}.$$

- $f(\boldsymbol{z}) = \|\boldsymbol{z}\|_0$ is our **objective function**

$$\|\boldsymbol{z}\|_0 \;:=\; \#\{\, i \mid z(i) \neq 0 \,\} \;=\; \sum_i \mathbb{1}_{z(i) \neq 0};$$

- $\boldsymbol{z} \in \mathbb{R}^n$ is the **optimization variable**
- $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{z}$ is the **constraint set**

Example: Sparse Recovery

- In general, optimizing $\|\boldsymbol{z}\|_0$ is NP-hard (computation growing **exponential** w.r.t. the data dimension)
- Nonetheless, when the matrix $\boldsymbol{A}$ satisfies certain benign conditions

$$e.g., \; (1-\delta)\,\|\boldsymbol{z}\|_2 \leq \|\boldsymbol{A}\boldsymbol{z}\|_2 \leq (1+\delta)\,\|\boldsymbol{z}\|_2 \,, \quad \forall \|\boldsymbol{z}\|_0 \leq k$$

It can be show that solving the *basis pursuit*

$$\boxed{(P1) \quad \min_{\boldsymbol{z}} \|\boldsymbol{z}\|_1 \,, \quad \text{s.t.} \quad \boldsymbol{y} \;=\; \boldsymbol{A}\boldsymbol{z},}$$

We obtain exactly the **same global solution** to that of (P0).

Here, $\|\boldsymbol{z}\|_1 = \sum_{k=1}^{m} |z_k|$ denotes the ℓ_1-norm of $\boldsymbol{z}$.

Basics: Vector Norms

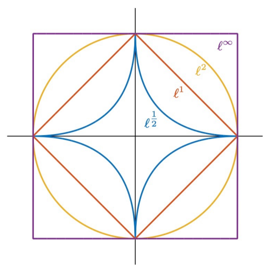
Definition. A vector norm is any real-valued function $\|\cdot\|$ that satisfies the following properties

- if $\boldsymbol{x} \neq 0$, then $\|\boldsymbol{x}\| > 0$
- for any $\alpha \in \mathbb{R}$, then $\|\alpha \boldsymbol{x}\| = |\alpha|\,\|\boldsymbol{x}\|$
- triangle inequality: $\|\boldsymbol{x} + \boldsymbol{y}\| \leq \|\boldsymbol{x}\| + \|\boldsymbol{y}\|$.

There are many norms, for example

$$\|\boldsymbol{x}\|_p = \left(\sum_i |x_i|^p\right)^{1/p}, \text{ for any } p \geq 1.$$

Basics: Vector Norms



There are many norms, for example

- $\|\boldsymbol{x}\|_1 \;=\; \sum_i |x_i|$;
- $\|\boldsymbol{x}\|_2 \;=\; \sqrt{\sum_i x_i^2}$;
- $\|\boldsymbol{x}\|_\infty \;=\; \max_i |x_i|$.

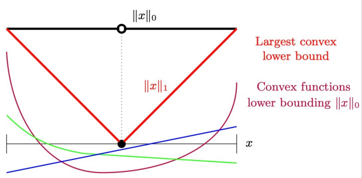
The following properties hold

- larger p results in smaller norm: $\|\boldsymbol{x}\|_\infty \leq \|\boldsymbol{x}\|_2 \leq \|\boldsymbol{x}\|_1$
- all norms are equivalent: $\|\boldsymbol{x}\|_\infty \leq \|\boldsymbol{x}\|_1 \leq n\,\|\boldsymbol{x}\|_\infty$
$\|\boldsymbol{x}\|_2 \leq \|\boldsymbol{x}\|_1 \leq \sqrt{n}\,\|\boldsymbol{x}\|_2$
$\|\boldsymbol{x}\|_\infty \leq \|\boldsymbol{x}\|_2 \leq \sqrt{n}\,\|\boldsymbol{x}\|_\infty$

Basics: ℓ_0-norm vs. ℓ_1-norm

$$(P0) \quad \min_{\boldsymbol{z}} \|\boldsymbol{z}\|_0 \,, \quad \text{s.t.} \quad \boldsymbol{y} \;=\; \boldsymbol{A}\boldsymbol{z}.$$

$$\boxed{(P1) \quad \min_{\boldsymbol{z}} \|\boldsymbol{z}\|_1 \,, \quad \text{s.t.} \quad \boldsymbol{y} \;=\; \boldsymbol{A}\boldsymbol{z},}$$



- $\|\cdot\|_0$ is **not** a norm, optimizing (P0) is NP-hard.
- $\|\cdot\|_1$ is a convex surrogate of $\|\cdot\|_0$, can be efficiently optimized.

Lecture Agenda

- What is an Optimization Problem?
- **Linear/Nonlinear Optimization Problems**
- Convex/Nonconvex Optimization Problems
- Smooth/Nonsmooth Optimization Problems

Linear vs Nonlinear Problems

A (mathematical) optimization problem is of the form:

$$\boxed{(P) \quad \min_{\boldsymbol{x}} \; f_0(\boldsymbol{x}), \;\; \text{s.t.} \;\; f_i(\boldsymbol{x}) \;\leq\; b_i, \; i = 1, \cdots, m}$$

- $\boldsymbol{x} \in \mathbb{R}^n$ is the **optimization variable**;
- $f_0(\cdot) : \mathbb{R}^n \mapsto \mathbb{R}$ is the **objective function**;
- The function $f_i(\cdot)$ is a **constraint**.

Linear vs Nonlinear Problems

- **Linear programming:** the objective and the constraints are *linear*. A function $f(\cdot)$ is linear *iff*

$$f(\lambda \boldsymbol{x} + (1-\lambda)\boldsymbol{y}) \;=\; \lambda f(\boldsymbol{x}) \;+\; (1-\lambda) f(\boldsymbol{y}),$$
$$\forall\, \boldsymbol{x},\, \boldsymbol{y} \in \mathbb{R}^n, \quad \forall\, \lambda \in \mathbb{R}.$$

- **Nonlinear programming:** either the objective or any of the constraints is **not** linear.

Linear Programming (LP)

$$\boxed{\min_{\boldsymbol{z}} \; \boldsymbol{c}^\top \boldsymbol{z}, \quad \text{s.t.} \;\; \boldsymbol{a}_i^\top \boldsymbol{z} \;\leq \boldsymbol{b}_i, \; i \;=\; 1, \cdots, m.}$$

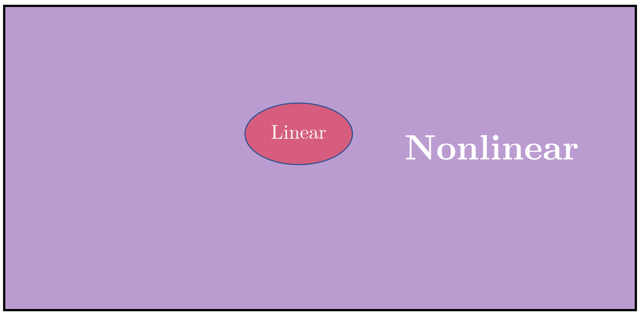
- No analytical solutions, many mature solvers:
  - Simplex method (Dantzig'47);
  - Interior point method (Karmakar'84);
- Many problems can be cast as LP.

Homework: show that the problem

$$\min_{\boldsymbol{z}} \|\boldsymbol{z}\|_1, \quad \text{s.t.} \quad \boldsymbol{y} \;=\; \boldsymbol{A}\boldsymbol{z},$$

can be cast as LP. (Hint: rewrite $\boldsymbol{z} = \boldsymbol{z}_+ - \boldsymbol{z}_-$).

Linear vs Nonlinear Problems



Lecture Agenda

- What is an Optimization Problem?
- Linear/Nonlinear Optimization Problems
- **Convex/Nonconvex Optimization Problems**
- Smooth/Nonsmooth Optimization Problems

Convex vs Nonconvex Problems

A (mathematical) optimization problem is of the form:

$$\boxed{(P) \quad \min_{\boldsymbol{x}} \; f_0(\boldsymbol{x}), \;\; \text{s.t.} \;\; f_i(\boldsymbol{x}) \;\leq\; b_i, \; i = 1, \cdots, m}$$

- $\boldsymbol{x} \in \mathbb{R}^n$ is the **optimization variable**;
- $f_0(\cdot) : \mathbb{R}^n \mapsto \mathbb{R}$ is the **objective function**;
- The function $f_i(\cdot)$ is a **constraint**.

Convex vs Nonconvex Problems

- **Convex programming:** the objective $f_0(\boldsymbol{x})$ and the constraint sets $f_i(\boldsymbol{x})$ are *all* **convex**.
- **Nonconvex programming:** either the objective or any of the constraints is **not** convex.

Basics: Convex Function and Convex Set

- **Convex function.** A function $f : \mathbb{R}^n \mapsto \mathbb{R}$ is convex *iff*

$$\boxed{f(\lambda \boldsymbol{x} + (1-\lambda)\boldsymbol{y}) \;\leq\; \lambda f(\boldsymbol{x}) \;+\; (1-\lambda) f(\boldsymbol{y})}$$

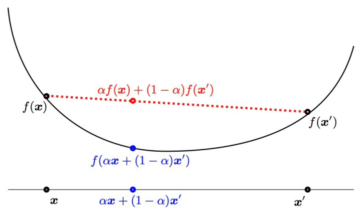
  for all $\boldsymbol{x},\, \boldsymbol{y} \in \mathbb{R}^n$ and $\lambda \in (0,1)$.

- **Convex set.** Let $\mathcal{C}_i = \{\boldsymbol{x} \in \mathbb{R}^n \mid f_i(\boldsymbol{x}) - b_i \leq 0\}$.

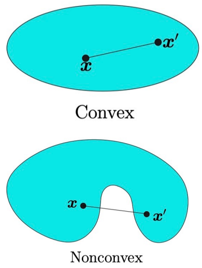
  We say the set $\mathcal{C}_i \subseteq \mathbb{R}^n$ is convex *iff*

$$\forall \boldsymbol{x}, \boldsymbol{x}' \in \mathcal{C}_i, \quad \alpha \in [0,1], \quad \alpha \boldsymbol{x} + (1-\alpha)\boldsymbol{x}' \in \mathcal{C}_i.$$

Convexity is More General than Linearity



Convex function



Convex sets

Why Convexity?

The **most important** property of a convex function:

***All* Local Minima are Global Minima**

(Homework): Every ***local minimizer*** of a convex function over a convex set is also a ***global minimizer*** of the function over the convex set.

Basics: Local Minima vs. Global Minima

Let the function $f : \mathcal{C} \mapsto \mathbb{R}$, where $\mathcal{C}$ is the constrain set.

- **Local Minima**. We say $\boldsymbol{x}_\star$ is a *local minimizer* of f over $\mathcal{C}$, if there exists a scalar $\varepsilon > 0$, such that
$$f(\boldsymbol{x}_\star) \;\le\; f(\boldsymbol{x}), \quad \forall \boldsymbol{x} \in \mathcal{B}(\boldsymbol{x}_\star, \varepsilon) := \{\boldsymbol{z} \in \mathcal{C} \mid \|\boldsymbol{z} - \boldsymbol{x}_\star\| \le \varepsilon\}.$$
- **Global Minima.** We say $\boldsymbol{x}_\star$ is a *global minimizer* of f over $\mathcal{C}$,
$$f(\boldsymbol{x}_\star) \;\le\; f(\boldsymbol{x}), \quad \forall \boldsymbol{x} \in \mathcal{C}.$$

Properties of Convex Functions

How to check whether a function is convex?

- **By definition:** A function $f(\cdot)$ is convex *iff*
$$f(\lambda \boldsymbol{x} + (1-\lambda)\boldsymbol{y}) \;\le\; \lambda f(\boldsymbol{x}) \;+\; (1-\lambda) f(\boldsymbol{y})$$
for all $\boldsymbol{x},\ \boldsymbol{y} \in \mathrm{dom} f$ and $\lambda \in (0,1)$.
- **First-order condition:** if f is first-order continuously differentiable (i.e., $f \in \mathcal{C}^1$)
$$f(\boldsymbol{y}) \;\ge\; f(\boldsymbol{x}) \;+\; \langle \nabla f(\boldsymbol{x}), \boldsymbol{y} - \boldsymbol{x} \rangle, \quad \forall \boldsymbol{x},\ \boldsymbol{y} \in \mathrm{dom} f.$$

Gradient and Hessian

- If $f : \mathbb{R}^n \mapsto \mathbb{R}$ is differentiable at $\boldsymbol{x}$, the gradient of f at $\boldsymbol{x}$ is
$$\nabla f(\boldsymbol{x}) \;:=\; \begin{bmatrix} \frac{\partial f(\boldsymbol{x})}{\partial x_1} \\ \vdots \\ \frac{\partial f(\boldsymbol{x})}{\partial x_n} \end{bmatrix}$$

Properties of Convex Functions

How to check whether a function is convex?

- **By definition:** A function f is convex *iff*
$$f(\lambda \boldsymbol{x} + (1-\lambda)\boldsymbol{y}) \;\le\; \lambda f(\boldsymbol{x}) \;+\; (1-\lambda) f(\boldsymbol{y})$$
for all $\boldsymbol{x},\ \boldsymbol{y} \in \mathrm{dom} f$ and $\lambda \in (0,1)$.
- **First-order condition:** if $f \in \mathcal{C}^1$
$$f(\boldsymbol{y}) \;\ge\; f(\boldsymbol{x}) \;+\; \langle \nabla f(\boldsymbol{x}), \boldsymbol{y} - \boldsymbol{x} \rangle, \quad \forall \boldsymbol{x},\ \boldsymbol{y} \in \mathrm{dom} f.$$
- **Second-order condition:** if $f \in \mathcal{C}^2$
$$\nabla^2 f(\boldsymbol{x}) \;\succeq\; \boldsymbol{0}, \quad \forall\, \boldsymbol{x} \in \mathrm{dom} f.$$

Gradient and Hessian

- If $f : \mathbb{R}^n \mapsto \mathbb{R}$ is differentiable at $\boldsymbol{x}$, the gradient of f at $\boldsymbol{x}$ is
$$\nabla f(\boldsymbol{x}) \;:=\; \begin{bmatrix} \frac{\partial f(\boldsymbol{x})}{\partial x_1} \\ \vdots \\ \frac{\partial f(\boldsymbol{x})}{\partial x_n} \end{bmatrix}$$
- If $f : \mathbb{R}^n \mapsto \mathbb{R}$ is twice-differentiable at $\boldsymbol{x}$, the Hessian of f at $\boldsymbol{x}$ is
$$\nabla^2 f(\boldsymbol{x}) \;:=\; \begin{bmatrix} \frac{\partial^2 f(\boldsymbol{x})}{\partial x_1^2} & \cdots & \frac{\partial^2 f(\boldsymbol{x})}{\partial x_n \partial x_1} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 f(\boldsymbol{x})}{\partial x_1 \partial x_n} & \cdots & \frac{\partial^2 f(\boldsymbol{x})}{\partial x_n^2} \end{bmatrix}$$

Properties of Convex Functions

How to check whether a function is convex?

- **By definition:** A function f is convex *iff*
$$f(\lambda \boldsymbol{x} + (1-\lambda)\boldsymbol{y}) \;\le\; \lambda f(\boldsymbol{x}) \;+\; (1-\lambda) f(\boldsymbol{y})$$
for all $\boldsymbol{x},\ \boldsymbol{y} \in \mathrm{dom} f$ and $\lambda \in (0,1)$.
- **First-order condition:** if $f \in \mathcal{C}^1$
$$f(\boldsymbol{y}) \;\ge\; f(\boldsymbol{x}) \;+\; \langle \nabla f(\boldsymbol{x}), \boldsymbol{y} - \boldsymbol{x} \rangle, \quad \forall \boldsymbol{x},\ \boldsymbol{y} \in \mathrm{dom} f.$$
- **Second-order condition:** if $f \in \mathcal{C}^2$
$$\nabla^2 f(\boldsymbol{x}) \;\succeq\; \boldsymbol{0}, \quad \forall\, \boldsymbol{x} \in \mathrm{dom} f.$$

If the inequality is *strict*, then it is called a *strict convex function*.

Example: Convex Problems

The *basis pursuit* problem for the sparse recovery

$$\boxed{(P1) \quad \min_{\boldsymbol{z}} \|\boldsymbol{z}\|_1\,, \quad \text{s.t.} \quad \boldsymbol{y} \;=\; \boldsymbol{A}\boldsymbol{z},}$$

is a **convex** optimization problem.

- The loss function $\|\boldsymbol{z}\|_1$ is convex;
- The constraint $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{z}$ is linear and hence convex.

Properties of Convex Functions

Common operations preserve convexity (**Homework**):

- **Nonnegative weighted sum:**

  If $f_1, \cdot, f_n$ are convex and $\alpha_1, \cdots, \alpha_n > 0$, then we have $f(\boldsymbol{x}) = \sum_{i=1}^n \alpha_i f_i(\boldsymbol{x})$ is a convex function.

- **Pointwise maximum:**

  If $f_1, \cdot, f_n$ are convex, then $f(\boldsymbol{x}) = \max_{1\leq i\leq n} \{f_i(\boldsymbol{x})\}$ is a convex function.

Properties of Convex Functions

Common operations preserve convexity (**Homework**):

- **Composition with an affine mapping:**

  Suppose $g : \mathbb{R}^n \mapsto \mathbb{R}$ is convex, and $\boldsymbol{A} \in \mathbb{R}^{n\times m}$ and $\boldsymbol{b} \in \mathbb{R}^n$, then $f = g(\boldsymbol{A}\boldsymbol{x} + \boldsymbol{b})$ is convex.

- **Restriction to a line:**

  Suppose $g : \mathbb{R}^n \mapsto \mathbb{R}$ is convex, and fix some $\boldsymbol{x},\ \boldsymbol{y} \in \mathbb{R}^n$ then $f : \mathbb{R} \mapsto \mathbb{R}$ with $f(\alpha) = g(\boldsymbol{x} + \alpha\boldsymbol{y})$ is convex.

Properties of Convex Functions

- **Strong convexity:**

Definition. A function f is μ-*strongly convex* with a constant $\mu > 0$, *iff*

$$g(\boldsymbol{x}) = f(\boldsymbol{x}) - \tfrac{\mu}{2}\|\boldsymbol{x}\|^2$$

is convex, $\forall\, \boldsymbol{x} \in \operatorname{dom} f$.

Moreover, if $f \in \mathcal{C}^1$, $\forall\, \boldsymbol{x},\ \boldsymbol{y} \in \operatorname{dom} f$:

$$f(\boldsymbol{y}) \;\geq\; f(\boldsymbol{x}) + \langle \nabla f(\boldsymbol{x}), \boldsymbol{y} - \boldsymbol{x}\rangle + \frac{\mu}{2}\|\boldsymbol{y} - \boldsymbol{x}\|^2$$

Example: Convex Problem

The *lasso* problem

$$\boxed{\min_{\boldsymbol{x}} \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2 + \lambda \cdot \|\boldsymbol{x}\|_1}$$

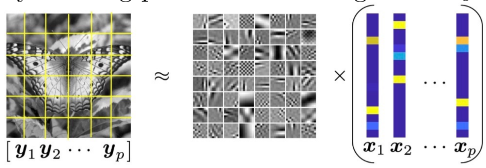
is convex, but not strongly convex;

$$\boxed{\min_{\boldsymbol{x}} \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2 + \frac{\mu}{2}\cdot\|\boldsymbol{x}\|_2^2 + \lambda \cdot \|\boldsymbol{x}\|_1}$$

is strongly convex.

Example: Nonconvex Problems

For dictionary learning problem of learning both $\boldsymbol{A}_0$ and $\boldsymbol{X}_0$



$$\boldsymbol{Y} \quad\approx\quad \boldsymbol{A}_0 \quad\times\quad \boldsymbol{X}_0$$

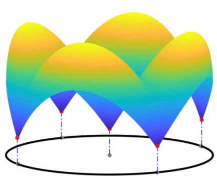
The following optimization formulation is **nonconvex**.

$$\boxed{\min_{\boldsymbol{A},\,\boldsymbol{X}} f(\boldsymbol{A}, \boldsymbol{X}) = \frac{1}{2}\|\boldsymbol{Y} - \boldsymbol{A}\boldsymbol{X}\|_F^2 + \lambda\|\boldsymbol{X}\|_1\,, \quad \text{s.t.} \quad \|\boldsymbol{a}_i\|_2 = 1,}$$

Example: Nonconvex Problems

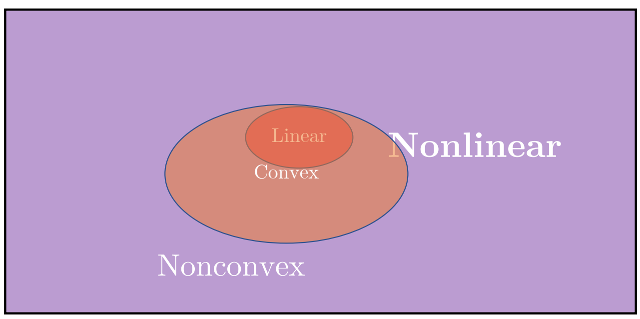
$$\boxed{\min_{\boldsymbol{A},\,\boldsymbol{X}} f(\boldsymbol{A}, \boldsymbol{X}) = \frac{1}{2}\|\boldsymbol{Y} - \boldsymbol{A}\boldsymbol{X}\|_F^2 + \lambda\|\boldsymbol{X}\|_1\,, \quad \text{s.t.} \quad \|\boldsymbol{a}_i\|_2 = 1,}$$

$$\boldsymbol{A}\cdot\boldsymbol{X} = (\boldsymbol{A}\boldsymbol{\Pi})\cdot(\boldsymbol{\Pi}^\top\boldsymbol{X})$$



Discrete symmetry

Convex vs Nonconvex Problems



Lecture Agenda

- What is an Optimization Problem?
- Linear/Nonlinear Optimization Problems
- Convex/Nonconvex Optimization Problems
- **Smooth/Nonsmooth Optimization Problems**

Smooth vs Nonsmooth Problems

- **Smooth problems.** A problem is smooth if both the objective function and the constraints are **differentiable**.
- **Nonsmooth problems.** A problem is nonsmooth if either the objective function or the constraint is **non-differentiable.**

Example: Smooth Problems

The ridge regression problem

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} f(\boldsymbol{x}) = \frac{1}{2}\|\boldsymbol{y}-\boldsymbol{A}\boldsymbol{x}\|_2^2 + \lambda\|\boldsymbol{x}\|_2^2,$$

- The function $f(\boldsymbol{x})$ is *continuously differentiable* everywhere;

Example: Smooth Problems

The ridge regression problem

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} f(\boldsymbol{x}) = \frac{1}{2}\|\boldsymbol{y}-\boldsymbol{A}\boldsymbol{x}\|_2^2 + \lambda\|\boldsymbol{x}\|_2^2,$$

- The function $f(\boldsymbol{x})$ is *continuously differentiable* everywhere;
- optimize the problem via *gradient descent*

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} - \underbrace{\tau}_{\text{stepsize}} \cdot \underbrace{\nabla f(\boldsymbol{x}^{(k)})}_{\text{gradient}},$$

$$\nabla f(\boldsymbol{x}) = \left(\boldsymbol{A}^\top\boldsymbol{A} + 2\lambda\boldsymbol{I}\right)\boldsymbol{x} - \boldsymbol{A}^\top\boldsymbol{y}.$$

Example: Nonsmooth Problems

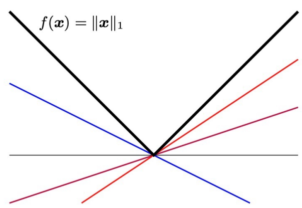
The *lasso* problem:

$$\min_{\boldsymbol{x}} g(\boldsymbol{x}) = \frac{1}{2}\|\boldsymbol{y}-\boldsymbol{A}\boldsymbol{x}\|_2^2 + \lambda\|\boldsymbol{x}\|_1,$$

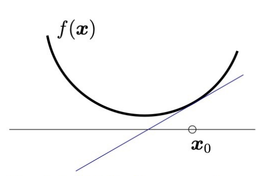
- The function $g(\boldsymbol{x})$ is *nonsmooth,* because $\|\boldsymbol{x}\|_1$ is non-differentiable at $\boldsymbol{0}$;
- The gradient is *not* well-defined everywhere.

Example: $f(x) = |x|,\ x\in\mathbb{R}$

$$\partial f(x) = \begin{cases} \{1\} & x>0, \\ [-1,1] & x=0, \\ \{-1\} & x<0. \end{cases}$$

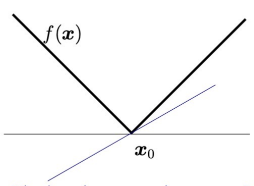


Basics: Subgradient & Subdifferential



$f(\boldsymbol{x}_0) + \langle\nabla f(\boldsymbol{x}_0), \boldsymbol{x}-\boldsymbol{x}_0\rangle$

differentiable



$f(\boldsymbol{x}_0) + \langle\boldsymbol{g}, \boldsymbol{x}-\boldsymbol{x}_0\rangle, \quad \boldsymbol{g}\in\partial f(\boldsymbol{x}_0)$

nondifferentiable

Basics: Subgradient & Subdifferential

- **Subgradient.** Let $f:\mathbb{R}^n\mapsto\mathbb{R}$ be *convex. A subgradient* of f at $\boldsymbol{x}_0$ is any $\boldsymbol{u}$ satisfying

$$\boxed{f(\boldsymbol{x}) \geq f(\boldsymbol{x}_0) + \langle\boldsymbol{u}, \boldsymbol{x}-\boldsymbol{x}_0\rangle, \quad \forall\,\boldsymbol{x}.}$$

- **Subdifferential.** It is the *set of all subgradients* of f at $\boldsymbol{x}_0$

$$\partial f(\boldsymbol{x}_0) := \{\boldsymbol{u} \mid f(\boldsymbol{x}) \geq f(\boldsymbol{x}_0) + \langle\boldsymbol{u}, \boldsymbol{x}-\boldsymbol{x}_0\rangle, \forall\boldsymbol{x}\in\mathbb{R}^n\}.$$

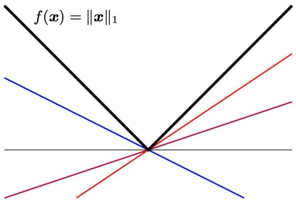
If f is differentiable at $\boldsymbol{x}_0$, then $\partial f(\boldsymbol{x}) = \{\nabla f(\boldsymbol{x}_0)\}$.

Example: $f(\boldsymbol{x}) = \|\boldsymbol{x}\|_1, \; \boldsymbol{x} \in \mathbb{R}^n$

$$\partial f(\boldsymbol{x}) \;=\; \mathcal{J}_1 \times \cdots \times \mathcal{J}_n,$$

where for $k = 1, 2, \cdots, n$,

$$\mathcal{J}_k \;=\; \begin{cases} \{1\} & x_k > 0, \\ [-1, 1] & x_k = 0, \\ \{-1\} & x_k < 0. \end{cases}$$



Example: Nonsmooth Problems

The *lasso* problem:

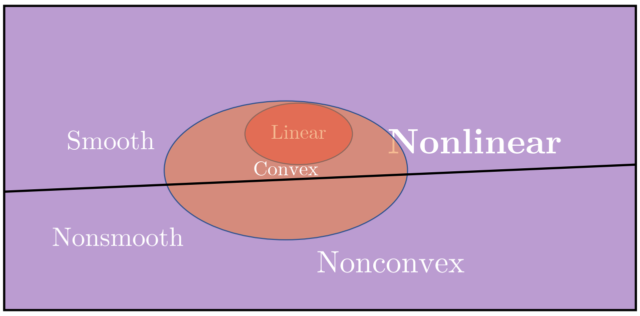
$$\min_{\boldsymbol{x}} \;\; g(\boldsymbol{x}) \;=\; \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2 + \lambda \|\boldsymbol{x}\|_1,$$

- optimize the problem via *subgradient method*

$$\boldsymbol{x}^{(k+1)} \;=\; \boldsymbol{x}^{(k)} - \underbrace{\tau}_{\text{stepsize}} \cdot \underbrace{\partial f(\boldsymbol{x}^{(k)})}_{\text{subgradient}},$$

$$\partial f(\boldsymbol{x}) \;=\; \boldsymbol{A}^\top (\boldsymbol{A}\boldsymbol{x} - \boldsymbol{y}) + \lambda \cdot \partial \|\boldsymbol{x}\|_1 .$$

Relationships of All Problems



This Course

We **mainly** focus on **efficient, global** methods for **nonlinear optimization** problems. In particular, we study

- **nonsmooth convex** problems,
- **smooth nonconvex** problems.

Additionally, we will touch base on basis of

- stochastic large-scale optimization for deep learning,
- constraint optimization methods,
- local methods for nonsmooth nonconvex problems.

Reference & Readings

- *High-Dimensional Data Analysis with Low-Dimensional Models: Principles, Computation, and Applications.* John Wright, Yi Ma. (**Chapter 1 & Appendix B**)