Notations and Conventions

X	Set
$\inf X \subseteq \mathbb{R}$	Infimum over the set X
$\mathbb{R}^{m\times n}$	set of all $m \times n$ real-valued matrices
$\mathbb{C}^{m\times n}$	set of all $m \times n$ complex-valued matrices
x_i	ith entry of column vector $\boldsymbol{x}$
a_{ij}	(i,j)th entry of matrix $\boldsymbol{A}$
$\boldsymbol{a}_i$	ith column of matrix $\boldsymbol{A}$
$\boldsymbol{a}_i^{\mathrm{T}}$	ith row of matrix $\boldsymbol{A}$
$\mathbb{S}^n$	set of all $n \times n$ real symmetric matrices, i.e., $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ and $a_{ij} = a_{ji}$ for all i,j
$\mathbb{H}^n$	set of all $n \times n$ complex Hermitian matrices, i.e., $\boldsymbol{A} \in \mathbb{C}^{n\times n}$ and $\bar{a}_{ij} = a_{ji}$ for all i,j
$\boldsymbol{A}^{\mathrm{T}}$	transpose of $\boldsymbol{A}$, i.e, $\boldsymbol{B} = \boldsymbol{A}^{\mathrm{T}}$ means $b_{ji} = a_{ij}$ for all i,j
$\boldsymbol{A}^{\mathrm{H}}$	Hermitian transpose of $\boldsymbol{A}$, i.e, $\boldsymbol{B} = \boldsymbol{A}^{\mathrm{H}}$ means $b_{ji} = \bar{a}_{ij}$ for all i,j
$\mathrm{trace}(\boldsymbol{A})$	sum of diagonal entries of square matrix $\boldsymbol{A}$
$\mathbf{1}$	A vector with all 1 entries
$\mathbf{0}$	either a vector of all zeros, or a matrix of all zeros
$\boldsymbol{e}_i$	a unit vector with the nonzero element at the ith entry
$\mathcal{C}(\boldsymbol{A})$	the column space of $\boldsymbol{A}$
$\mathcal{R}(\boldsymbol{A})$	the row space of $\boldsymbol{A}$
$\mathcal{N}(\boldsymbol{A})$	the null space of $\boldsymbol{A}$
$\mathrm{Proj}_{\mathcal{M}}(\boldsymbol{A})$	the projection of $\boldsymbol{A}$ onto the set $\mathcal{M}$

Chapter 1

Week1

1.1. Monday

1.1.1. Introduction to Optimizaiton

The usual optimization formulation is given by:

$$\begin{aligned} &\min f(\boldsymbol{x}), && \text{where } f:\mathbb{R}^n \mapsto \mathbb{R} \\ &\text{such that} && \boldsymbol{x} \in X \subseteq \mathbb{R}^n \end{aligned}$$

One example of the set X is given by:

$$X = \left\{ \boldsymbol{x} \in \mathbb{R}^n \;\middle|\; \begin{array}{l} C_i(\boldsymbol{x}) = \boldsymbol{0}, i = 1,2,\dots,m \le n \\ h_i(\boldsymbol{x}) \ge \boldsymbol{0}, i = 1,2,\dots,p \end{array} \right\}$$

Linear programming can be easily solved, but Integer linear programming is much harder. The equivalent LP formulation is given by:

$$\begin{aligned} \min \quad & \boldsymbol{c}^{\mathrm{T}}\boldsymbol{x} \\ \text{s.t.} \quad & \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b} \\ & \boldsymbol{c} \le \boldsymbol{B}\boldsymbol{x} \le \boldsymbol{c}' \end{aligned}$$

1.2. Wednesday

1.2.1. Reviewing for Linear Algebra

Questions:

- What is the necessary and sufficient condition for the linear system $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ to have a solution $\boldsymbol{x}$?

  Answer: $\boldsymbol{b} \in \mathcal{C}(\boldsymbol{A})$.

- For $\boldsymbol{A} \in \mathbb{S}^n$, what is the necessary and sufficient condition for $\boldsymbol{A} \succeq 0$?

  Answer: $\boldsymbol{x}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{x} = 0$ for $\forall x \in \mathbb{R}^n$; or $\lambda_i(\boldsymbol{A}) \geq 0$ for all i.

1.2.2. Reviewing for Calculus

For function $f : \mathbb{R}^n \mapsto \mathbb{R}$:

- We use notation $f \in \mathcal{C}^n$ to denote f is **continuously differentiable to nth order**. This course will basically deal with such functions.

- We use notation $\nabla f(x)$ to denote the **Gradient** of f at x; and $\nabla^2 f(x)$ denotes the second order derivative of f at x. Note that $\nabla^2 f(x) \in \mathbb{S}^n$ for $f \in \mathcal{C}^1$.

- We use notation $\mathbb{S}^n$ to denote the set of all symmetric $n \times n$ matrices, i.e.,

  $$\mathbb{S}^n = \{\boldsymbol{X} \in \mathbb{R}^{n\times n} \mid \boldsymbol{X}^{\mathrm{T}} = \boldsymbol{X}\}$$

  Moreover, $\mathbb{S}^n_+$ denotes the set of all symmetric $n \times n$ matrices with all eigenvalues non-negative:

  $$\mathbb{S}^n_+ = \{\boldsymbol{X} \in \mathbb{R}^{n\times n} \mid \boldsymbol{X}^{\mathrm{T}} = \boldsymbol{X} \succeq 0\}$$

1.2.3. Introduction to Optimization

The usual optimization formulation is given by:

$$\begin{aligned} &\min f(\boldsymbol{x}), \quad \text{where } f:\mathbb{R}^n \mapsto \mathbb{R} \\ &\text{such that} \quad \boldsymbol{x} \in X \subseteq \mathbb{R}^n \end{aligned}$$

- The simplest case for the constraint is $X = \mathbb{R}^n$, which leads to **unconstrainted** optimization problem.
- Or $X = P$ is a **polyhedron**, i.e., the boundaries for the region are all lines.

Definition 1.1 [Constraint Regions] In space $\mathbb{R}^n$,

- the hyper-plane is defined as:
$$\left\{\boldsymbol{x} \middle| \boldsymbol{a}^{\mathrm{T}}\boldsymbol{x} = \boldsymbol{\beta}\right\}$$
with constants $\boldsymbol{a} \in \mathbb{R}^n$ and $\boldsymbol{\beta} \in \mathbb{R}$
- the half-space is defined as
$$\left\{\boldsymbol{x} \middle| \boldsymbol{a}^{\mathrm{T}}\boldsymbol{x} \le \boldsymbol{\beta}\right\}$$
- the polyhedron is defined as the **intersection** of a **finite** number of hyperplanes or half-spaces

■

Next, we give the definition for the basic optimization problem:

Definition 1.2 [Linear Programming] The Linear Programming is given by:

$$\begin{aligned} &\min \quad \boldsymbol{c}^{\mathrm{T}}\boldsymbol{x}, \\ &\text{such that} \quad \boldsymbol{x} \in P(\text{polyhedron}) \end{aligned}$$

Or it can be reformulated as:

$$\begin{aligned} \min \quad & \boldsymbol{c}^{\mathrm{T}}\boldsymbol{x}, \\ \text{such that} \quad & \boldsymbol{A}_I\boldsymbol{x} \le \boldsymbol{b}_I \\ & \boldsymbol{A}_E\boldsymbol{x} = \boldsymbol{b}_E \in \mathbb{R}^m, \quad m < n. \end{aligned}$$

Definition 1.3 [Optimality] $\boldsymbol{x}^*$ is said to be :

- the **local minimum** of $f(\boldsymbol{x})$ if there exists small ϵ such that

$$f(\boldsymbol{x}^*) \le f(\boldsymbol{x}), \quad \forall \boldsymbol{x} \in \mathcal{B}(\boldsymbol{x}^*, \epsilon) \bigcap X := \{\boldsymbol{x} \mid \|\boldsymbol{x} - \boldsymbol{x}^*\| \le \epsilon\} \bigcap X$$

- the **global minimum** if

$$f(\boldsymbol{x}^*) \le f(\boldsymbol{x}), \quad \forall \boldsymbol{x} \in X$$

R Unless specified, when we want to minimize a non-convex function, it usually means we only find its **local minimum**. This is because usually the local minimum is good enough.

The optimization task is essentially find $\boldsymbol{x}^*$ such that

$$\boldsymbol{x}^* = \arg\min_{\boldsymbol{x} \in X} f(\boldsymbol{x}) \in \mathbb{R}^n.$$

philosophy (optimization sufficient and necessity). philosophy of relaxation (convex nulls)

The Optimality conditions are the **most important** theoretical tools for optimization.

Theorem 1.1 — **Optimality condition.** The optimality condition contains

1. Necessary Condition (exclude non-optimal points):

$$n = 1 \text{ special case: } \begin{cases} \text{1st order: } f'(x) = 0 \\ \text{2rd order: } f''(x) \geq 0 \end{cases} \implies \begin{cases} \text{1st order: } \nabla f(x) = 0 \\ \text{2rd order: } \nabla f^2(x) \succeq 0 \end{cases}$$

2. Sufficient Condition (may identify optimal solutions)

$$n = 1 \text{ special case: } \begin{cases} \text{1st order: } f'(x) = 0 \\ \text{2rd order: } f''(x) > 0 \end{cases} \implies \begin{cases} \text{1st order: } \nabla f(x) = 0 \\ \text{2rd order: } \nabla f^2(x) \succ 0 \end{cases}$$

Proof. The $n = 1$ special case can imply the general case for optimality condition. For multivariate f, we set $\boldsymbol{x} = \boldsymbol{x}^* + td$ with t to be the stepsize and d to be the direction. For fixed t and d, we define $h(t) = f(\boldsymbol{x}) = f(\boldsymbol{x}^* + td)$. It follows that

$$h'(t) = \nabla^{\mathrm{T}} f(\boldsymbol{x}^* + td)d$$

We find $h'(0) = \nabla^{\mathrm{T}} f(\boldsymbol{x}^*)d$ for $\forall$ d, which implies $\nabla f(\boldsymbol{x}^*) = 0$. ■

Note that there is a gap between necessary and sufficient conditions, which puts us in an embarrassing position. However, the convex condition can save us:

Theorem 1.2 If f is convex in $\mathcal{C}^1$, then $\nabla f(\boldsymbol{x}) = 0$ is the **necessary** and **sufficient** condition.