# Mathematics for Machine Learning

## — Continuous Optimization: Preliminary Convex Optimization

Joseph Chuang-Chieh Lin

Department of Computer Science & Information Engineering,
Tamkang University

Fall 2023

# Credits for the resource

- The slides are based on the textbooks:
  - *Marc Peter Deisenroth, A. Aldo Faisal, and Cheng Soon Ong: Mathematics for Machine Learning. Cambridge University Press. 2020.*
  - *Howard Anton, Chris Rorres, Anton Kaul: Elementary Linear Algebra. Wiley. 2019.*
- We could partially refer to the monograph:
  *Francesco Orabona: A Modern Introduction to Online Learning. https://arxiv.org/abs/1912.13213*

# Outline

# Outline

# Our Focus & Motivation

Convex Optimization.

- A class of optimization problems where we can guarantee global optimality.

  $f(\cdot)$ is a convex function.

  The constraints $g(\cdot)$ and $h(\cdot)$ form convex sets.

# Convex Sets & Functions

## Convex set

A set $\mathcal{C}$ is convex if for any $\mathbf{x}, \mathbf{y} \in \mathcal{C}$, we have

$$\forall \alpha \in [0,1], \alpha \mathbf{x} + (1-\alpha)\mathbf{y} \in \mathcal{C}.$$

## Convex function

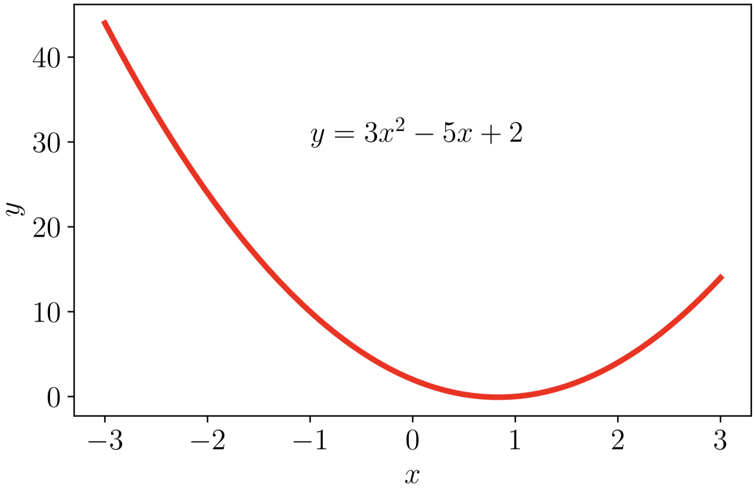
A function $f : \mathcal{C} \subseteq \mathbb{R}^D \mapsto \mathbb{R}$ is convex if for any $\mathbf{x}, \mathbf{y} \in \mathcal{C}$,

$$\forall \alpha \in [0,1], f((1-\alpha)\mathbf{x} + \alpha \mathbf{y}) \leq (1-\alpha) f(\mathbf{x}) + \alpha f(\mathbf{y}).$$

Equivalently, if $f$ is differentiable (i.e., $\nabla f(\mathbf{x})$ exists for all $\mathbf{x} \in \mathcal{C}$), then $f$ is convex if and only if for all $\mathbf{x}, \mathbf{y} \in \mathcal{C}$,

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \nabla_{\mathbf{x}} f(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}).$$

# An Example of Convex Functions



# Remark

- If $f(\mathbf{x})$ is twice differentiable (i.e., the Hessian exists for all $\mathbf{x} \in \mathcal{C}$), then

$$f(\mathbf{x}) \text{ is convex } \iff \nabla_{\mathbf{x}}^2 f(\mathbf{x}) \text{ is positive semidefinite.}$$

# Example

**Example**

Show that $f(x) = x \lg x$ is convex for $x > 0$.

# Example

## Example

Show that $f(x) = x \lg x$ is convex for $x > 0$.

- Note: $\lg x := \log_2 x$ and $\ln x := \log_e x$.

# Example

## Example

Show that $f(x) = x \lg x$ is convex for $x > 0$.

- Note: $\lg x := \log_2 x$ and $\ln x := \log_e x$.
- Compute $\nabla_x f(x)$.

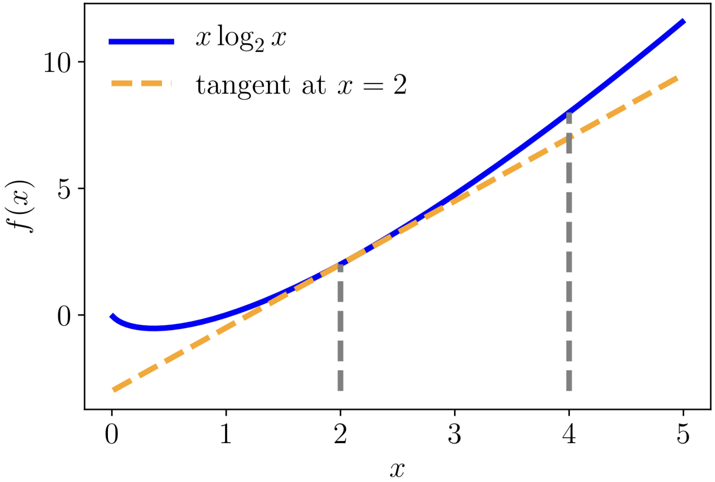
# Example

## Example

Show that $f(x) = x \lg x$ is convex for $x > 0$.

- Note: $\lg x := \log_2 x$ and $\ln x := \log_e x$.
- Compute $\nabla_x f(x)$.
- Say given $x = 2, y = 4$, compute $f(x) + \nabla_x f(x)^\top (y - x)$.

# Example



# Example (Theorem)

**Theorem**

Given a nonnegative real $\alpha \geq 0$ and two convex functions $f_1$ and $f_2$, then $\alpha \cdot f_1 + (1 - \alpha) f_2$ is still convex.

# Example (Theorem)

**Theorem**

Given a nonnegative real $\alpha \geq 0$ and two convex functions $f_1$ and $f_2$, then $\alpha \cdot f_1 + (1-\alpha) f_2$ is still convex.

- By definition,

$$\begin{aligned} f_1(\alpha \mathbf{x} + (1-\alpha)\mathbf{y}) &\leq \alpha f_1(\mathbf{x}) + (1-\alpha) f_1(\mathbf{y}) \\ f_2(\alpha \mathbf{x} + (1-\alpha)\mathbf{y}) &\leq \alpha f_2(\mathbf{x}) + (1-\alpha) f_2(\mathbf{y}). \end{aligned}$$

- Summing up:

$$\begin{aligned} & f_1(\alpha \mathbf{x} + (1-\alpha)\mathbf{y}) + f_2(\alpha \mathbf{x} + (1-\alpha)\mathbf{y}) \\ & \leq \alpha f_1(\mathbf{x}) + (1-\alpha) f_1(\mathbf{y}) + \alpha f_2(\mathbf{x}) + (1-\alpha) f_2(\mathbf{y}) \\ & \alpha(f_1(\mathbf{x}) + f_2(\mathbf{x})) + (1-\alpha)(f_1(\mathbf{y}) + f_2(\mathbf{y})). \end{aligned}$$

# Outline

# Linear Programming

- Consider the special case that all the preceding functions are linear.

$$\begin{aligned} \min_{\mathbf{x} \in \mathbb{R}^d} \quad & \mathbf{c}^\top \mathbf{x} \\ \text{subject to} \quad & \boldsymbol{A}\mathbf{x} \leq \mathbf{b}. \end{aligned}$$

where $\boldsymbol{A} \in \mathbb{R}^{m \times d}$ and $\mathbf{b} \in \mathbb{R}^m$.

# Linear Programming + Lagrangian (1/2)

- The Lagrangian:

$$\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}) = \mathbf{c}^\top \mathbf{x} + \boldsymbol{\lambda}^\top (\boldsymbol{A}\mathbf{x} - \mathbf{b})$$

where $\boldsymbol{\lambda} \in \mathbb{R}^m$ is the vector of non-negative Lagrange multipliers.

# Linear Programming + Lagrangian (1/2)

- The Lagrangian:

$$\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}) = \mathbf{c}^\top \mathbf{x} + \boldsymbol{\lambda}^\top (\boldsymbol{A}\mathbf{x} - \mathbf{b})$$

where $\boldsymbol{\lambda} \in \mathbb{R}^m$ is the vector of non-negative Lagrange multipliers.

- Rearranging the terms:

$$\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}) = (\mathbf{c} + \boldsymbol{A}^\top \boldsymbol{\lambda})^\top \mathbf{x} - \boldsymbol{\lambda}^\top \mathbf{b}.$$

# Linear Programming + Lagrangian (1/2)

- The Lagrangian:

$$\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}) = \mathbf{c}^\top \mathbf{x} + \boldsymbol{\lambda}^\top (\boldsymbol{A}\mathbf{x} - \mathbf{b})$$

where $\boldsymbol{\lambda} \in \mathbb{R}^m$ is the vector of non-negative Lagrange multipliers.

- Rearranging the terms:

$$\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}) = (\mathbf{c} + \boldsymbol{A}^\top \boldsymbol{\lambda})^\top \mathbf{x} - \boldsymbol{\lambda}^\top \mathbf{b}.$$

- Taking the derivate w.r.t. $\mathbf{x}$:

$$\mathbf{c} + \boldsymbol{A}^\top \boldsymbol{\lambda} = \mathbf{0}.$$

# Linear Programming + Lagrangian (1/2)

- The Lagrangian:

$$\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}) = \mathbf{c}^\top \mathbf{x} + \boldsymbol{\lambda}^\top (\boldsymbol{A}\mathbf{x} - \mathbf{b})$$

  where $\boldsymbol{\lambda} \in \mathbb{R}^m$ is the vector of non-negative Lagrange multipliers.

- Rearranging the terms:

$$\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}) = (\mathbf{c} + \boldsymbol{A}^\top \boldsymbol{\lambda})^\top \mathbf{x} - \boldsymbol{\lambda}^\top \mathbf{b}.$$

- Taking the derivate w.r.t. $\mathbf{x}$:

$$\mathbf{c} + \boldsymbol{A}^\top \boldsymbol{\lambda} = \mathbf{0}.$$

- Thus, the dual Lagrangian is $\mathcal{D}(\boldsymbol{\lambda}) = -\lambda^\top \mathbf{b}$.

# Linear Programming + Lagrangian (2/2)

- Recall that we would like to maximize $\mathcal{D}(\boldsymbol{\lambda})$ and the constraint that $\boldsymbol{\lambda} \geq \mathbf{0}$.
- The dual optimization problem is

$$\begin{aligned} \max_{\boldsymbol{\lambda} \in \mathbb{R}^m} \quad & -\mathbf{b}^\top \boldsymbol{\lambda} \\ \text{subject to} \quad & \mathbf{c} + \boldsymbol{A}^\top \boldsymbol{\lambda} = \mathbf{0} \\ & \boldsymbol{\lambda} \geq \mathbf{0} \end{aligned}$$

which is also a linear program but with $m$ variables.

# Linear Programming + Lagrangian (2/2)

- Recall that we would like to maximize $\mathcal{D}(\boldsymbol{\lambda})$ and the constraint that $\boldsymbol{\lambda} \geq \mathbf{0}$.
- The dual optimization problem is

$$\begin{aligned} \max_{\boldsymbol{\lambda} \in \mathbb{R}^m} \quad & -\mathbf{b}^\top \boldsymbol{\lambda} \\ \text{subject to} \quad & \mathbf{c} + \boldsymbol{A}^\top \boldsymbol{\lambda} = \mathbf{0} \\ & \boldsymbol{\lambda} \geq \mathbf{0} \end{aligned}$$

which is also a linear program but with $m$ variables.

⋆ Solve the primal or the dual program depending on whether $m$ (i.e., # constraints) or $d$ (i.e., # variables) is larger.

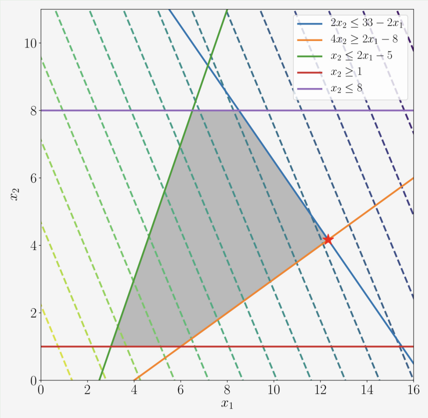
## Example

Consider the linear program

$$\min_{\mathbf{x}\in\mathbb{R}^2} -\begin{bmatrix} 5 \\ 3 \end{bmatrix}^\top \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

subject to

$$\begin{bmatrix} 2 & 2 \\ 2 & -4 \\ -2 & 1 \\ 0 & -1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \le \begin{bmatrix} 33 \\ 8 \\ 5 \\ -1 \\ 8 \end{bmatrix}.$$



# Outline

# Quadratic Programming

Consider the case of a convex quadratic objective function, where the constraints are affine:

$$\begin{aligned} \min_{\mathbf{x} \in \mathbb{R}^d} \quad & \frac{1}{2}\mathbf{x}^\top \boldsymbol{Q}\mathbf{x} + \mathbf{c}^\top \mathbf{x} \\ \text{subject to} \quad & \boldsymbol{A}\mathbf{x} \leq \mathbf{b}, \end{aligned}$$

where

- $\boldsymbol{A} \in \mathbb{R}^{m \times d}, \mathbf{b} \in \mathbb{R}^m$ and $\mathbf{c} \in \mathbb{R}^d$.
- $\boldsymbol{Q} \in \mathbb{R}^{d \times d}$: a positive definite matrix.

  $d$ variables and $m$ linear constraints.

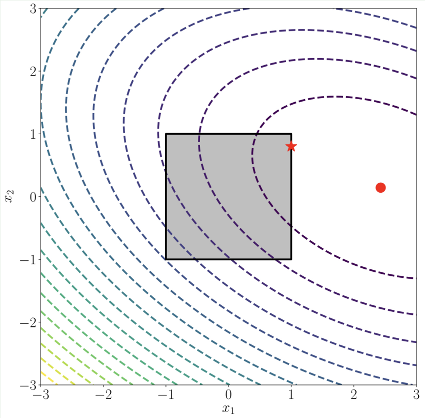
## Example

Consider the quadratic program

$$\min_{\mathbf{x}\in\mathbb{R}^2} \frac{1}{2}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}^\top \begin{bmatrix} 2 & 1 \\ 1 & 4 \end{bmatrix}^\top \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 5 \\ 3 \end{bmatrix}^\top \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

subject to

$$\begin{bmatrix} 1 & 0 \\ -1 & 0 \\ 0 & 1 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \leq \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}.$$



# Quadratic Programming (1/3)

Consider the case of a convex quadratic objective function, where the constraints are affine:

$$\begin{aligned} \min_{\mathbf{x} \in \mathbb{R}^d} \quad & \frac{1}{2}\mathbf{x}^\top \boldsymbol{Q}\mathbf{x} + \mathbf{c}^\top \mathbf{x} \\ \text{subject to} \quad & \boldsymbol{A}\mathbf{x} \leq \mathbf{b}, \end{aligned}$$

# Quadratic Programming (1/3)

Consider the case of a convex quadratic objective function, where the constraints are affine:

$$\begin{aligned} \min_{\mathbf{x}\in\mathbb{R}^d} \quad & \frac{1}{2}\mathbf{x}^\top \boldsymbol{Q}\mathbf{x} + \mathbf{c}^\top \mathbf{x} \\ \text{subject to} \quad & \boldsymbol{A}\mathbf{x} \le \mathbf{b}, \end{aligned}$$

The Lagrangian is given by

$$\begin{aligned} \mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}) &= \frac{1}{2}\mathbf{x}^\top \boldsymbol{Q}\mathbf{x} + \mathbf{c}^\top \mathbf{x} + \boldsymbol{\lambda}^\top (\boldsymbol{A}\mathbf{x} - \mathbf{b}) \\ &= \frac{1}{2}\mathbf{x}^\top \boldsymbol{Q}\mathbf{x} + (\mathbf{c} + \boldsymbol{A}^\top \boldsymbol{\lambda})^\top \mathbf{x} - \boldsymbol{\lambda}^\top \mathbf{b}. \end{aligned}$$

# Quadratic Programming (2/3)

$$\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}) = \frac{1}{2}\mathbf{x}^\top \boldsymbol{Q}\mathbf{x} + (\mathbf{c} + \boldsymbol{A}^\top \boldsymbol{\lambda})^\top \mathbf{x} - \boldsymbol{\lambda}^\top \mathbf{b}.$$

# Quadratic Programming (2/3)

$$\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}) = \frac{1}{2}\mathbf{x}^\top \boldsymbol{Q}\mathbf{x} + (\mathbf{c} + \boldsymbol{A}^\top \boldsymbol{\lambda})^\top \mathbf{x} - \boldsymbol{\lambda}^\top \mathbf{b}.$$

Taking the derivative w.r.t. $\mathbf{x}$ and setting it to zero:

# Quadratic Programming (2/3)

$$\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}) = \frac{1}{2}\mathbf{x}^\top \boldsymbol{Q}\mathbf{x} + (\mathbf{c} + \boldsymbol{A}^\top \boldsymbol{\lambda})^\top \mathbf{x} - \boldsymbol{\lambda}^\top \mathbf{b}.$$

Taking the derivative w.r.t. $\mathbf{x}$ and setting it to zero:

$$\boldsymbol{Q}\mathbf{x} + (\mathbf{c} + \boldsymbol{A}^\top \boldsymbol{\lambda}) = \mathbf{0}.$$

# Quadratic Programming (2/3)

$$\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}) = \frac{1}{2}\mathbf{x}^\top \boldsymbol{Q}\mathbf{x} + (\mathbf{c} + \boldsymbol{A}^\top \boldsymbol{\lambda})^\top \mathbf{x} - \boldsymbol{\lambda}^\top \mathbf{b}.$$

Taking the derivative w.r.t. $\mathbf{x}$ and setting it to zero:

$$\boldsymbol{Q}\mathbf{x} + (\mathbf{c} + \boldsymbol{A}^\top \boldsymbol{\lambda}) = \mathbf{0}.$$

Note that $\boldsymbol{Q}$ is invertible

# Quadratic Programming (2/3)

$$\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}) = \frac{1}{2}\mathbf{x}^\top \boldsymbol{Q}\mathbf{x} + (\mathbf{c} + \boldsymbol{A}^\top \boldsymbol{\lambda})^\top \mathbf{x} - \boldsymbol{\lambda}^\top \mathbf{b}.$$

Taking the derivative w.r.t. $\mathbf{x}$ and setting it to zero:

$$\boldsymbol{Q}\mathbf{x} + (\mathbf{c} + \boldsymbol{A}^\top \boldsymbol{\lambda}) = \mathbf{0}.$$

Note that $\boldsymbol{Q}$ is invertible ($\because$ positive definite $\Rightarrow$ nonzero eigenvalues $\Rightarrow$ nonzero determinant), then

$$\mathbf{x} = -\boldsymbol{Q}^{-1}(\mathbf{c} + \boldsymbol{A}^\top \boldsymbol{\lambda}).$$

(Thanks to Yo-Cheng Chang)

# Quadratic Programming (2/3)

$$\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}) = \frac{1}{2}\mathbf{x}^\top \boldsymbol{Q}\mathbf{x} + (\mathbf{c} + \boldsymbol{A}^\top \boldsymbol{\lambda})^\top \mathbf{x} - \boldsymbol{\lambda}^\top \mathbf{b}.$$

Taking the derivative w.r.t. $\mathbf{x}$ and setting it to zero:

$$\boldsymbol{Q}\mathbf{x} + (\mathbf{c} + \boldsymbol{A}^\top \boldsymbol{\lambda}) = \mathbf{0}.$$

Note that $\boldsymbol{Q}$ is invertible ($\because$ positive definite $\Rightarrow$ nonzero eigenvalues $\Rightarrow$ nonzero determinant), then

$$\mathbf{x} = -\boldsymbol{Q}^{-1}(\mathbf{c} + \boldsymbol{A}^\top \boldsymbol{\lambda}).$$

(Thanks to Yo-Cheng Chang)

Substituting it back to $\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda})$, we get the dual Lagrangian

# Quadratic Programming (2/3)

$$\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}) = \frac{1}{2}\mathbf{x}^\top \boldsymbol{Q}\mathbf{x} + (\mathbf{c} + \boldsymbol{A}^\top \boldsymbol{\lambda})^\top \mathbf{x} - \boldsymbol{\lambda}^\top \mathbf{b}.$$

Taking the derivative w.r.t. $\mathbf{x}$ and setting it to zero:

$$\boldsymbol{Q}\mathbf{x} + (\mathbf{c} + \boldsymbol{A}^\top \boldsymbol{\lambda}) = \mathbf{0}.$$

Note that $\boldsymbol{Q}$ is invertible ($\because$ positive definite $\Rightarrow$ nonzero eigenvalues $\Rightarrow$ nonzero determinant), then

$$\mathbf{x} = -\boldsymbol{Q}^{-1}(\mathbf{c} + \boldsymbol{A}^\top \boldsymbol{\lambda}).$$

(Thanks to Yo-Cheng Chang)

Substituting it back to $\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda})$, we get the dual Lagrangian

$$\mathcal{D}(\boldsymbol{\lambda}) = -\frac{1}{2}(\mathbf{c} + \boldsymbol{A}^\top \boldsymbol{\lambda})^\top \boldsymbol{Q}^{-1}(\mathbf{c} + \boldsymbol{A}^\top \boldsymbol{\lambda}) - \boldsymbol{\lambda}^\top \mathbf{b}.$$

# Quadratic Programming (3/3)

Therefore, the dual optimization problem is given by

$$\begin{aligned} \max_{\boldsymbol{\lambda} \in \mathbb{R}^m} \quad & -\frac{1}{2}(\mathbf{c} + \boldsymbol{A}^\top \boldsymbol{\lambda})^\top \boldsymbol{Q}^{-1}(\mathbf{c} + \boldsymbol{A}^\top \boldsymbol{\lambda}) - \boldsymbol{\lambda}^\top \mathbf{b} \\ \text{subject to} \quad & \boldsymbol{\lambda} \geq \mathbf{0} \end{aligned}$$

- **Heads up:** Application in Support Vector Machine (SVM).

# Discussions