

# Optimization

Ngoc-Hoang Luong

University of Information Technology (UIT)  
Vietnam National University - Ho Chi Minh City (VNU-HCM)

Math for CS, Fall 2021

The contents of this document are taken mainly from the follow sources:

- Kevin P. Murphy. Probabilistic Machine Learning: An Introduction. <sup>1</sup>

---

<sup>1</sup><https://probml.github.io/pml-book/book1.html>

# Table of Contents

- 1 Introduction
- 2 Matrix calculus
- 3 Positive definite matrices
- 4 Optimality conditions
- 5 Constrained vs unconstrained optimization
- 6 Convex vs nonconvex optimization
- 7 Smooth vs nonsmooth optimization
- 8 First-order methods

# Table of Contents

- 1 Introduction
- 2 Matrix calculus
- 3 Positive definite matrices
- 4 Optimality conditions
- 5 Constrained vs unconstrained optimization
- 6 Convex vs nonconvex optimization
- 7 Smooth vs nonsmooth optimization
- 8 First-order methods

# Introduction

- The core problem in ML is parameter estimation (model fitting).

# Introduction

- The core problem in ML is parameter estimation (model fitting).
- We need to solve an **optimization problem**: i.e., trying to find the values for a set of variables  $\theta \in \Theta$  that minimize a scalar-valued **loss function** or **cost function**:  $\mathcal{L} : \Theta \rightarrow \mathbb{R}$

$$\theta = \underset{\theta}{\operatorname{argmin}} \mathcal{L}^*(\theta) \quad (1)$$

# Introduction

- The core problem in ML is parameter estimation (model fitting).
- We need to solve an **optimization problem**: i.e., trying to find the values for a set of variables  $\theta \in \Theta$  that minimize a scalar-valued **loss function** or **cost function**:  $\mathcal{L} : \Theta \rightarrow \mathbb{R}$

$$\theta = \underset{\theta}{\operatorname{argmin}} \mathcal{L}^*(\theta) \quad (1)$$

- The **parameter space** is given by  $\Theta \in \mathbb{R}^D$ , where  $D$  is the number of variables being optimized.

# Introduction

- The core problem in ML is parameter estimation (model fitting).
- We need to solve an **optimization problem**: i.e., trying to find the values for a set of variables  $\theta \in \Theta$  that minimize a scalar-valued **loss function** or **cost function**:  $\mathcal{L} : \Theta \rightarrow \mathbb{R}$

$$\theta = \underset{\theta}{\operatorname{argmin}} \mathcal{L}^*(\theta) \quad (1)$$

- The **parameter space** is given by  $\Theta \in \mathbb{R}^D$ , where  $D$  is the number of variables being optimized.
- We focus on **continuous optimization**.



# Introduction

- The core problem in ML is parameter estimation (model fitting).
- We need to solve an **optimization problem**: i.e., trying to find the values for a set of variables  $\theta \in \Theta$  that minimize a scalar-valued **loss function** or **cost function**:  $\mathcal{L} : \Theta \rightarrow \mathbb{R}$

$$\theta = \underset{\theta}{\operatorname{argmin}} \mathcal{L}^*(\theta) \quad (1)$$

- The **parameter space** is given by  $\Theta \in \mathbb{R}^D$ , where  $D$  is the number of variables being optimized.
- We focus on **continuous optimization**.
- To maximize a **score function** or **reward function**  $R(\theta)$ , we can minimize  $\mathcal{L}(\theta) = -R(\theta)$ .

# Introduction

- The core problem in ML is parameter estimation (model fitting).
- We need to solve an **optimization problem**: i.e., trying to find the values for a set of variables  $\theta \in \Theta$  that minimize a scalar-valued **loss function** or **cost function**:  $\mathcal{L} : \Theta \rightarrow \mathbb{R}$

$$\theta = \underset{\theta}{\operatorname{argmin}} \mathcal{L}^*(\theta) \quad (1)$$

- The **parameter space** is given by  $\Theta \in \mathbb{R}^D$ , where  $D$  is the number of variables being optimized.
- We focus on **continuous optimization**.
- To maximize a **score function** or **reward function**  $R(\theta)$ , we can minimize  $\mathcal{L}(\theta) = -R(\theta)$ .
- The term **objective function** refers to a function we want to maximize or minimize.

# Introduction

- The core problem in ML is parameter estimation (model fitting).
- We need to solve an **optimization problem**: i.e., trying to find the values for a set of variables  $\theta \in \Theta$  that minimize a scalar-valued **loss function** or **cost function**:  $\mathcal{L} : \Theta \rightarrow \mathbb{R}$

$$\theta = \underset{\theta}{\operatorname{argmin}} \mathcal{L}^*(\theta) \quad (1)$$

- The **parameter space** is given by  $\Theta \in \mathbb{R}^D$ , where  $D$  is the number of variables being optimized.
- We focus on **continuous optimization**.
- To maximize a **score function** or **reward function**  $R(\theta)$ , we can minimize  $\mathcal{L}(\theta) = -R(\theta)$ .
- The term **objective function** refers to a function we want to maximize or minimize.
- An algorithm to find an optimum of an objective function is a **solver**.

# Local versus global optimization

- A point that satisfies Equation 1 is called a **global optimum**. Finding such a point is called **global optimization**.

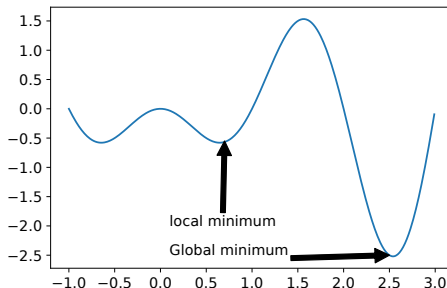
# Local versus global optimization

- A point that satisfies Equation 1 is called a **global optimum**. Finding such a point is called **global optimization**.
- In general, finding global optima is computationally **intractable**. We will try to find a **local optimum**.

# Local versus global optimization

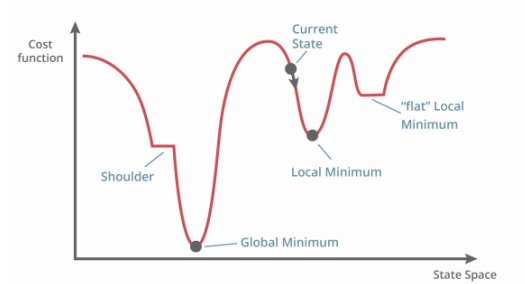
- A point that satisfies Equation 1 is called a **global optimum**. Finding such a point is called **global optimization**.
- In general, finding global optima is computationally **intractable**. We will try to find a **local optimum**.
- For continuous problem, a local optimum is a point  $\theta^*$  which has lower (or equal) cost than “nearby” points.

$$\exists \delta > 0, \forall \theta \in \Theta, \quad \text{s.t.} \quad \|\theta - \theta^*\| < \delta, \quad \mathcal{L}(\theta^*) \leq \mathcal{L}(\theta) \quad (2)$$



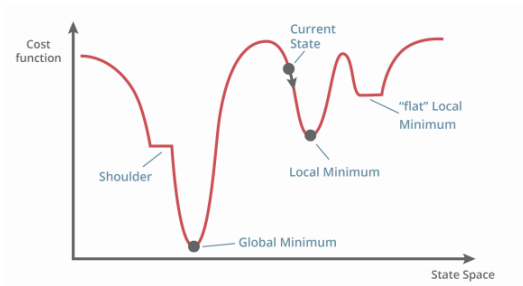
# Local versus global optimization

- A local minimum could be surrounded by other local minima with the same objective value; this is known as a **flat local minimum**.



# Local versus global optimization

- A local minimum could be surrounded by other local minima with the same objective value; this is known as a **flat local minimum**.



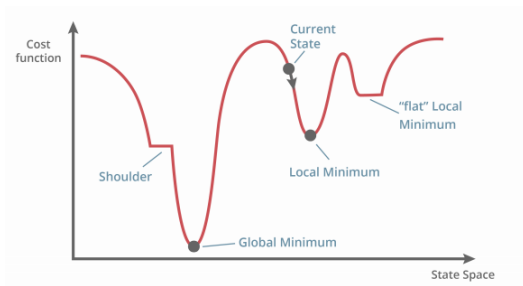
- A point is said to be a **strict local minimum** if its cost is strictly lower than those of neighboring points.

$$\exists \delta > 0, \forall \theta \in \Theta, \theta \neq \theta^* : \|\theta - \theta^*\| < \delta, \mathcal{L}(\theta^*) < \mathcal{L}(\theta) \quad (3)$$



# Local versus global optimization

- A local minimum could be surrounded by other local minima with the same objective value; this is known as a **flat local minimum**.



- A point is said to be a **strict local minimum** if its cost is strictly lower than those of neighboring points.

$$\exists \delta > 0, \forall \theta \in \Theta, \theta \neq \theta^* : \|\theta - \theta^*\| < \delta, \mathcal{L}(\theta^*) < \mathcal{L}(\theta) \quad (3)$$

- We can define a (strict) **local maximum** analogously.

# Table of Contents

- 1 Introduction
- 2 Matrix calculus**
- 3 Positive definite matrices
- 4 Optimality conditions
- 5 Constrained vs unconstrained optimization
- 6 Convex vs nonconvex optimization
- 7 Smooth vs nonsmooth optimization
- 8 First-order methods

# Derivatives

- The topic of **calculus** concerns computing “**rates of change**” of functions as we vary their inputs.

# Derivatives

- The topic of **calculus** concerns computing “**rates of change**” of functions as we vary their inputs.
- Consider a scalar-argument function  $f : \mathbb{R} \rightarrow \mathbb{R}$ . Its **derivative** at a point  $a$  is the quantity

$$f'(x) \triangleq \lim_{h \rightarrow 0} \frac{f(x+h) - f(x)}{h}$$

assuming the limit exists.

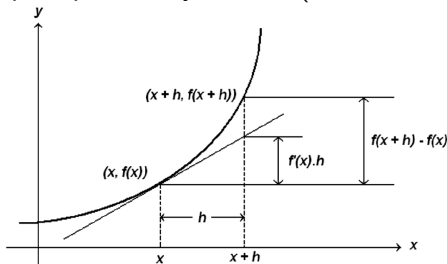
# Derivatives

- The topic of **calculus** concerns computing “**rates of change**” of functions as we vary their inputs.
- Consider a scalar-argument function  $f : \mathbb{R} \rightarrow \mathbb{R}$ . Its **derivative** at a point  $a$  is the quantity

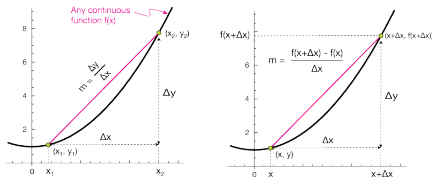
$$f'(x) \triangleq \lim_{h \rightarrow 0} \frac{f(x+h) - f(x)}{h}$$

assuming the limit exists.

- This measures how quickly the output changes when we move a small distance in the input space away from  $x$  (i.e., the “rate of change”).



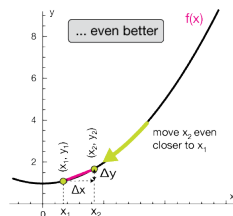
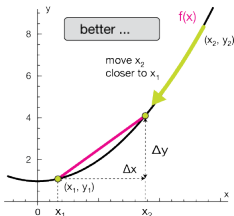
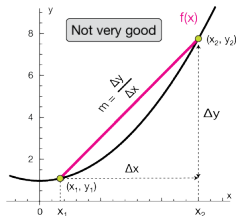
# Derivatives



- $f'(x)$  can be seen as the slope of the tangent line at  $f(x)$

$$f(x + \Delta x) \approx f(x) + f'(x)\Delta x$$

for small  $\Delta x$ .



- We can compute a **finite difference** approximation to the derivative by using a finite step size  $h$

$$\begin{aligned} f'(x) &= \lim_{h \rightarrow 0} \underbrace{\frac{f(x+h) - f(x)}{h}}_{\text{forward difference}} \\ &= \lim_{h \rightarrow 0} \underbrace{\frac{f(x+h/2) - f(x-h/2)}{h}}_{\text{central difference}} \\ &= \lim_{h \rightarrow 0} \underbrace{\frac{f(x) - f(x-h)}{h}}_{\text{backward difference}} \end{aligned}$$

- The smaller the step size  $h$ , the better the estimate.

# Derivatives

- We can think of **differentiation** as an operator that maps functions to functions,  $D(f) = f'$



# Derivatives

- We can think of **differentiation** as an operator that maps functions to functions,  $D(f) = f'$
- $f'(x)$  computes the derivative at  $x$  (assuming the derivative exists at that point).

# Derivatives

- We can think of **differentiation** as an operator that maps functions to functions,  $D(f) = f'$
- $f'(x)$  computes the derivative at  $x$  (assuming the derivative exists at that point).
- The prime symbol  $f'$  to denote derivative is **Lagrange notation**.

# Derivatives

- We can think of **differentiation** as an operator that maps functions to functions,  $D(f) = f'$
- $f'(x)$  computes the derivative at  $x$  (assuming the derivative exists at that point).
- The prime symbol  $f'$  to denote derivative is **Lagrange notation**.
- The second derivative function, which measures how quickly the gradient is changing, is denoted by  $f''$ .

# Derivatives

- We can think of **differentiation** as an operator that maps functions to functions,  $D(f) = f'$
- $f'(x)$  computes the derivative at  $x$  (assuming the derivative exists at that point).
- The prime symbol  $f'$  to denote derivative is **Lagrange notation**.
- The second derivative function, which measures how quickly the gradient is changing, is denoted by  $f''$ .
- The  $n$ 'th derivative function is denote  $f^{(n)}$ .

# Derivatives

- We can think of **differentiation** as an operator that maps functions to functions,  $D(f) = f'$
- $f'(x)$  computes the derivative at  $x$  (assuming the derivative exists at that point).
- The prime symbol  $f'$  to denote derivative is **Lagrange notation**.
- The second derivative function, which measures how quickly the gradient is changing, is denoted by  $f''$ .
- The  $n$ 'th derivative function is denote  $f^{(n)}$ .
- We can use **Leibniz notation**, if we denote the function by  $y = f(x)$ , and its derivative by  $\frac{dy}{dx}$  or  $\frac{d}{dx} f(x)$ .

# Derivatives

- We can think of **differentiation** as an operator that maps functions to functions,  $D(f) = f'$
- $f'(x)$  computes the derivative at  $x$  (assuming the derivative exists at that point).
- The prime symbol  $f'$  to denote derivative is **Lagrange notation**.
- The second derivative function, which measures how quickly the gradient is changing, is denoted by  $f''$ .
- The  $n$ 'th derivative function is denote  $f^{(n)}$ .
- We can use **Leibniz notation**, if we denote the function by  $y = f(x)$ , and its derivative by  $\frac{dy}{dx}$  or  $\frac{d}{dx} f(x)$ .
- To denote the evaluation of the derivative at a point  $a$ , we write

$$\left. \frac{df}{dx} \right|_{x=a}.$$

# Gradients

- We extend the notion of derivatives to handle vector-argument functions,  $f : \mathbb{R}^n \rightarrow \mathbb{R}$ , by defining the **partial derivative** of  $f$  with respect to  $x_i$  to be

$$\frac{\partial f}{\partial x_i} = \lim_{h \rightarrow 0} \frac{f(\mathbf{x} + h\mathbf{e}_i) - f(\mathbf{x})}{h}$$

where  $\mathbf{e}_i$  is the  $i$ 'th unit vector,  $\mathbf{e}_i = (0, \dots, 1, \dots, 0)$  with the  $i$ 'th element = 1 and all the other elements are 0.

# Gradients

- We extend the notion of derivatives to handle vector-argument functions,  $f : \mathbb{R}^n \rightarrow \mathbb{R}$ , by defining the **partial derivative** of  $f$  with respect to  $x_i$  to be

$$\frac{\partial f}{\partial x_i} = \lim_{h \rightarrow 0} \frac{f(\mathbf{x} + h\mathbf{e}_i) - f(\mathbf{x})}{h}$$

where  $\mathbf{e}_i$  is the  $i$ 'th unit vector,  $\mathbf{e}_i = (0, \dots, 1, \dots, 0)$  with the  $i$ 'th element = 1 and all the other elements are 0.

- The **gradient** of  $f$  at a point  $\mathbf{x}$  is the vector of its partial derivatives

$$\mathbf{g} = \frac{\partial f}{\partial \mathbf{x}} = \nabla f = \begin{pmatrix} \frac{\partial f}{\partial x_1} \\ \vdots \\ \frac{\partial f}{\partial x_n} \end{pmatrix} = \frac{\partial f}{\partial x_1} \mathbf{e}_1 + \dots + \frac{\partial f}{\partial x_n} \mathbf{e}_n$$



# Gradients

- We extend the notion of derivatives to handle vector-argument functions,  $f : \mathbb{R}^n \rightarrow \mathbb{R}$ , by defining the **partial derivative** of  $f$  with respect to  $x_i$  to be

$$\frac{\partial f}{\partial x_i} = \lim_{h \rightarrow 0} \frac{f(\mathbf{x} + h\mathbf{e}_i) - f(\mathbf{x})}{h}$$

where  $\mathbf{e}_i$  is the  $i$ 'th unit vector,  $\mathbf{e}_i = (0, \dots, 1, \dots, 0)$  with the  $i$ 'th element = 1 and all the other elements are 0.

- The **gradient** of  $f$  at a point  $\mathbf{x}$  is the vector of its partial derivatives

$$\mathbf{g} = \frac{\partial f}{\partial \mathbf{x}} = \nabla f = \begin{pmatrix} \frac{\partial f}{\partial x_1} \\ \vdots \\ \frac{\partial f}{\partial x_n} \end{pmatrix} = \frac{\partial f}{\partial x_1} \mathbf{e}_1 + \dots + \frac{\partial f}{\partial x_n} \mathbf{e}_n$$

- To emphasize the point at which the gradient is evaluated, we write

$$\mathbf{g}(\mathbf{x}^*) \triangleq \left. \frac{\partial f}{\partial \mathbf{x}} \right|_{\mathbf{x}^*}$$

- Example:

$$f(x_1, x_2) = x_1^2 + x_1x_2 + 3x_2^2$$
$$\nabla f(x_1, x_2) = \begin{pmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \end{pmatrix} = \begin{pmatrix} 2x_1 + x_2 \\ x_1 + 6x_2 \end{pmatrix}$$

- The nabla operator  $\nabla$  maps a function  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  to another function  $\mathbf{g} : \mathbb{R}^n \rightarrow \mathbb{R}^n$ .
- Since  $\mathbf{g}()$  is a vector-valued function, it is known as a vector field.

# Directional derivative

- The **directional derivative** measures how much the function  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  changes along a direction  $\mathbf{v}$  in space.

$$D_{\mathbf{v}}f(\mathbf{x}) = \lim_{h \rightarrow 0} \frac{f(\mathbf{x} + h\mathbf{v}) - f(\mathbf{x})}{h}$$

- We can approximate this numerically using 2 function calls to  $f$ , regardless of  $n$ .
- By contrast, a numerical approximation to the standard gradient vector takes  $n + 1$  calls (or  $2n$  if using central differences).
- The directional derivative along  $\mathbf{v}$  is the scalar product of the gradient  $\mathbf{g}$  and the vector  $\mathbf{v}$ :

$$D_{\mathbf{v}}f(\mathbf{x}) = \nabla f(\mathbf{x}) \cdot \mathbf{v}$$

# Directional derivative

**Example:** Let  $f(x, y) = x^2y$ . Find the derivative of  $f$  in the direction  $(1,2)$  at the point  $(3,2)$ .

- The gradient  $\nabla f(x, y)$  is:

$$\nabla f(x, y) = \begin{pmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{pmatrix} = \begin{pmatrix} 2xy \\ x^2 \end{pmatrix}$$

$$\nabla f(3, 2) = \begin{pmatrix} 12 \\ 9 \end{pmatrix} = 12 \begin{pmatrix} 1 \\ 0 \end{pmatrix} + 9 \begin{pmatrix} 0 \\ 1 \end{pmatrix} = 12\mathbf{e}_1 + 9\mathbf{e}_2$$

- Let  $\mathbf{u} = u_1\mathbf{e}_1 + u_2\mathbf{e}_2$  be a unit vector. The derivative of  $f$  in the direction of  $\mathbf{u}$  at  $(3,2)$  is:

$$\begin{aligned} D_{\mathbf{u}}f(3, 2) &= \nabla f(3, 2) \cdot \mathbf{u} \\ &= (12\mathbf{e}_1 + 9\mathbf{e}_2) \cdot (u_1\mathbf{e}_1 + u_2\mathbf{e}_2) \\ &= 12u_1 + 9u_2 \end{aligned}$$

# Directional derivative

## Example (cont.)

- The unit vector in the direction of vector  $(1,2)$  is:

$$\mathbf{u} = \frac{(1,2)}{\|(1,2)\|} = \frac{(1,2)}{\sqrt{1^2+2^2}} = \frac{(1,2)}{\sqrt{5}} = (1/\sqrt{5}, 2/\sqrt{5})$$

- The directional derivative at  $(3,2)$  in the direction of  $(1,2)$  is:

$$\begin{aligned} D_{\mathbf{u}}f(3,2) &= 12u_1 + 9u_2 \\ &= \frac{12}{\sqrt{5}} + \frac{18}{\sqrt{5}} = \frac{30}{\sqrt{5}} \end{aligned}$$

- We normalize vector  $(1,2)$  so that the directional derivative is independent of its magnitude and depending only on its direction.

**Example 2:** Let  $f(x, y) = x^2y$ . Find the derivative of  $f$  in the direction of  $(2,1)$  at the point  $(3,2)$ .

# Directional derivative

**Example 2:** Let  $f(x, y) = x^2y$ . Find the derivative of  $f$  in the direction of  $(2,1)$  at the point  $(3,2)$ .

- The unit vector in the direction of  $(2,1)$  is:

$$\mathbf{u} = \frac{(2, 1)}{\sqrt{5}} = (2/\sqrt{5}, 1/\sqrt{5})$$

- The directional derivative of  $f$  at  $(3,2)$  in the direction of  $(2,1)$  is:

$$\begin{aligned} D_{\mathbf{u}}f(3, 2) &= 12u_1 + 9u_2 \\ &= \frac{24}{\sqrt{5}} + \frac{9}{\sqrt{5}} = \frac{33}{\sqrt{5}} \end{aligned}$$

## Questions:

- At a point  $\mathbf{a}$ , in which direction  $\mathbf{u}$  is the directional derivative  $D_{\mathbf{u}}f(\mathbf{a})$  maximal?
- What is the directional derivative in that direction  $D_{\mathbf{u}}f(\mathbf{a}) = ?$



# Directional derivative

## Questions:

- At a point  $\mathbf{a}$ , in which direction  $\mathbf{u}$  is the directional derivative  $D_{\mathbf{u}}f(\mathbf{a})$  maximal?
- What is the directional derivative in that direction  $D_{\mathbf{u}}f(\mathbf{a}) = ?$

The relationship between the **gradient** and the **directional derivative**:

$$\begin{aligned} D_{\mathbf{u}}f(\mathbf{a}) &= \nabla f(\mathbf{a}) \cdot \mathbf{u} \\ &= \|\nabla f(\mathbf{a})\| \|\mathbf{u}\| \cos \theta \quad [\theta \text{ is the angle between } \mathbf{u} \text{ and the gradient.}] \\ &= \|\nabla f(\mathbf{a})\| \cos \theta \quad [\mathbf{u} \text{ is a unit vector.}] \end{aligned}$$

# Directional derivative

## Questions:

- At a point  $\mathbf{a}$ , in which direction  $\mathbf{u}$  is the directional derivative  $D_{\mathbf{u}}f(\mathbf{a})$  maximal?
- What is the directional derivative in that direction  $D_{\mathbf{u}}f(\mathbf{a}) = ?$

The relationship between the **gradient** and the **directional derivative**:

$$\begin{aligned} D_{\mathbf{u}}f(\mathbf{a}) &= \nabla f(\mathbf{a}) \cdot \mathbf{u} \\ &= \|\nabla f(\mathbf{a})\| \|\mathbf{u}\| \cos \theta \quad [\theta \text{ is the angle between } \mathbf{u} \text{ and the gradient.}] \\ &= \|\nabla f(\mathbf{a})\| \cos \theta \quad [\mathbf{u} \text{ is a unit vector.}] \end{aligned}$$

The maximal value of  $D_{\mathbf{u}}f(\mathbf{a})$  occurs when  $\mathbf{u}$  and  $\nabla f(\mathbf{a})$  point in the same direction (i.e.,  $\theta = 0$ ).

# Directional derivative

$$\begin{aligned} D_{\mathbf{u}}f(\mathbf{a}) &= \nabla f(\mathbf{a}) \cdot \mathbf{u} \\ &= \|\nabla f(\mathbf{a})\| \|\mathbf{u}\| \cos \theta \quad [\theta \text{ is the angle between } \mathbf{u} \text{ and the gradient.}] \\ &= \|\nabla f(\mathbf{a})\| \cos \theta \quad [\mathbf{u} \text{ is a unit vector.}] \end{aligned}$$

- When  $\theta = 0$ , the directional derivative  $D_{\mathbf{u}}f(\mathbf{a}) = \|\nabla f(\mathbf{a})\|$ .
- When  $\theta = \pi$ , the directional derivative  $D_{\mathbf{u}}f(\mathbf{a}) = -\|\nabla f(\mathbf{a})\|$ .
- For what value of  $\theta$  is  $D_{\mathbf{u}}f(\mathbf{a}) = 0$ ?

- Consider a function that maps a vector to another vector,  $\mathbf{f} : \mathbb{R}^n \rightarrow \mathbb{R}^m$ . The **Jacobian matrix** of this function is an  $m \times n$  matrix of partial derivatives:

$$\mathbf{J}_{\mathbf{f}}(\mathbf{x}) = \frac{\partial \mathbf{f}}{\partial \mathbf{x}^T} \triangleq \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1} & \cdots & \frac{\partial f_m}{\partial x_n} \end{pmatrix} = \begin{pmatrix} \nabla f_1(\mathbf{x})^T \\ \vdots \\ \nabla f_m(\mathbf{x})^T \end{pmatrix}$$

- We layout the results in the same orientation as the output  $\mathbf{f}$ . This is called the numerator layout of the Jacobian formulation.

- For a function  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  that is twice differentiable, the **Hessian matrix** is the (symmetric)  $n \times n$  matrix of second partial derivatives

$$\mathbf{H}_f = \frac{\partial^2 f}{\partial \mathbf{x}^2} = \nabla^2 f = \begin{pmatrix} \frac{\partial^2 f}{\partial x_1^2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{pmatrix}$$

- The Hessian is the Jacobian of the gradient.

# Hessian

**Example:** Find the Hessian of  $f(x, y) = x^2y + y^2x$  at the point  $(1,1)$ .

# Hessian

**Example:** Find the Hessian of  $f(x, y) = x^2y + y^2x$  at the point  $(1, 1)$ .

- First, compute the gradient (i.e., first-order partial derivatives):

$$\nabla f(x, y) = \begin{pmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{pmatrix} = \begin{pmatrix} 2xy + y^2 \\ x^2 + 2yx \end{pmatrix}$$

# Hessian

**Example:** Find the Hessian of  $f(x, y) = x^2y + y^2x$  at the point  $(1,1)$ .

- First, compute the gradient (i.e., first-order partial derivatives):

$$\nabla f(x, y) = \begin{pmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{pmatrix} = \begin{pmatrix} 2xy + y^2 \\ x^2 + 2yx \end{pmatrix}$$

- Second, compute the Hessian (i.e., second-order partial derivatives):

$$\mathbf{H}_f(x, y) = \begin{pmatrix} \frac{\partial^2 f}{\partial x^2} & \frac{\partial^2 f}{\partial x \partial y} \\ \frac{\partial^2 f}{\partial y \partial x} & \frac{\partial^2 f}{\partial y^2} \end{pmatrix} = \begin{pmatrix} 2y & 2x + 2y \\ 2x + 2y & 2x \end{pmatrix}$$



**Example:** Find the Hessian of  $f(x, y) = x^2y + y^2x$  at the point  $(1,1)$ .

- First, compute the gradient (i.e., first-order partial derivatives):

$$\nabla f(x, y) = \begin{pmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{pmatrix} = \begin{pmatrix} 2xy + y^2 \\ x^2 + 2yx \end{pmatrix}$$

- Second, compute the Hessian (i.e., second-order partial derivatives):

$$\mathbf{H}_f(x, y) = \begin{pmatrix} \frac{\partial^2 f}{\partial x^2} & \frac{\partial^2 f}{\partial x \partial y} \\ \frac{\partial^2 f}{\partial y \partial x} & \frac{\partial^2 f}{\partial y^2} \end{pmatrix} = \begin{pmatrix} 2y & 2x + 2y \\ 2x + 2y & 2x \end{pmatrix}$$

- Finally, evaluate the Hessian matrix at the point  $(1,1)$ :

$$\mathbf{H}_f(1, 1) = \begin{pmatrix} 2 & 4 \\ 4 & 2 \end{pmatrix}$$

# Geometric meaning

- If we follow the direction  $\mathbf{d}$  from  $\mathbf{x}$ , we can define a uni-dimensional function  $g(\alpha)$ :

$$g(\alpha) = f(\mathbf{x} + \alpha\mathbf{d})$$

$$g'(\alpha) = \mathbf{d}^\top \nabla f(\mathbf{x} + \alpha\mathbf{d})$$

$$g''(\alpha) = \mathbf{d}^\top \nabla^2 f(\mathbf{x} + \alpha\mathbf{d}) \mathbf{d}$$

- Interpretation

$$g'(0) = \mathbf{d}^\top \nabla f(\mathbf{x}) \quad [\text{directional derivative}]$$

$$g''(0) = \mathbf{d}^\top \nabla^2 f(\mathbf{x}) \mathbf{d} \quad [\text{directional curvature}]$$

- If  $g''(0)$  is non-negative with a certain  $\mathbf{d}$ :  $f$  is convex in direction  $\mathbf{d}$ .
- If  $g''(0)$  is non-negative for all  $\mathbf{d}$ :  $\nabla^2 f(\mathbf{x})$  is positive semidefinite  $\rightarrow f$  is convex at  $\mathbf{x}$ .

# Table of Contents

- 1 Introduction
- 2 Matrix calculus
- 3 Positive definite matrices**
- 4 Optimality conditions
- 5 Constrained vs unconstrained optimization
- 6 Convex vs nonconvex optimization
- 7 Smooth vs nonsmooth optimization
- 8 First-order methods

We say that a symmetric  $n \times n$  matrix  $A$  is:

- positive semidefinite ( $A \succeq 0$ ) if  $\mathbf{x}^\top A \mathbf{x} \geq 0$  for all  $\mathbf{x}$ ,

We say that a symmetric  $n \times n$  matrix  $A$  is:

- positive semidefinite ( $A \succeq 0$ ) if  $\mathbf{x}^\top A \mathbf{x} \geq 0$  for all  $\mathbf{x}$ ,
- positive definite ( $A \succ 0$ ) if  $\mathbf{x}^\top A \mathbf{x} > 0$  for all  $\mathbf{x} \neq \mathbf{0}$ ,

# Definitions

We say that a symmetric  $n \times n$  matrix  $A$  is:

- positive semidefinite ( $A \succeq 0$ ) if  $\mathbf{x}^\top A \mathbf{x} \geq 0$  for all  $\mathbf{x}$ ,
- positive definite ( $A \succ 0$ ) if  $\mathbf{x}^\top A \mathbf{x} > 0$  for all  $\mathbf{x} \neq \mathbf{0}$ ,
- negative semidefinite ( $A \preceq 0$ ) if  $\mathbf{x}^\top A \mathbf{x} \leq 0$  for all  $\mathbf{x}$ ,

We say that a symmetric  $n \times n$  matrix  $A$  is:

- positive semidefinite ( $A \succeq 0$ ) if  $\mathbf{x}^\top A \mathbf{x} \geq 0$  for all  $\mathbf{x}$ ,
- positive definite ( $A \succ 0$ ) if  $\mathbf{x}^\top A \mathbf{x} > 0$  for all  $\mathbf{x} \neq \mathbf{0}$ ,
- negative semidefinite ( $A \preceq 0$ ) if  $\mathbf{x}^\top A \mathbf{x} \leq 0$  for all  $\mathbf{x}$ ,
- negative definite ( $A \prec 0$ ) if  $\mathbf{x}^\top A \mathbf{x} < 0$  for all  $\mathbf{x} \neq \mathbf{0}$ ,

We say that a symmetric  $n \times n$  matrix  $A$  is:

- positive semidefinite ( $A \succeq 0$ ) if  $\mathbf{x}^\top A \mathbf{x} \geq 0$  for all  $\mathbf{x}$ ,
- positive definite ( $A \succ 0$ ) if  $\mathbf{x}^\top A \mathbf{x} > 0$  for all  $\mathbf{x} \neq \mathbf{0}$ ,
- negative semidefinite ( $A \preceq 0$ ) if  $\mathbf{x}^\top A \mathbf{x} \leq 0$  for all  $\mathbf{x}$ ,
- negative definite ( $A \prec 0$ ) if  $\mathbf{x}^\top A \mathbf{x} < 0$  for all  $\mathbf{x} \neq \mathbf{0}$ ,
- indefinite if none of the above apply.



We say that a symmetric  $n \times n$  matrix  $A$  is:

- positive semidefinite ( $A \succeq 0$ ) if  $\mathbf{x}^\top A \mathbf{x} \geq 0$  for all  $\mathbf{x}$ ,
- positive definite ( $A \succ 0$ ) if  $\mathbf{x}^\top A \mathbf{x} > 0$  for all  $\mathbf{x} \neq \mathbf{0}$ ,
- negative semidefinite ( $A \preceq 0$ ) if  $\mathbf{x}^\top A \mathbf{x} \leq 0$  for all  $\mathbf{x}$ ,
- negative definite ( $A \prec 0$ ) if  $\mathbf{x}^\top A \mathbf{x} < 0$  for all  $\mathbf{x} \neq \mathbf{0}$ ,
- indefinite if none of the above apply.

We say that a symmetric  $n \times n$  matrix  $A$  is:

- positive semidefinite ( $A \succeq 0$ ) if  $\mathbf{x}^\top A \mathbf{x} \geq 0$  for all  $\mathbf{x}$ ,
- positive definite ( $A \succ 0$ ) if  $\mathbf{x}^\top A \mathbf{x} > 0$  for all  $\mathbf{x} \neq \mathbf{0}$ ,
- negative semidefinite ( $A \preceq 0$ ) if  $\mathbf{x}^\top A \mathbf{x} \leq 0$  for all  $\mathbf{x}$ ,
- negative definite ( $A \prec 0$ ) if  $\mathbf{x}^\top A \mathbf{x} < 0$  for all  $\mathbf{x} \neq \mathbf{0}$ ,
- indefinite if none of the above apply.
- The expression  $\mathbf{x}^\top A \mathbf{x}$  is a function of  $\mathbf{x}$  called the quadratic form associated to  $A$ . (It's made up of terms like  $x_i^2$  and  $x_i x_j$ .)
- We make these definitions for a symmetric matrix  $A$ , i.e.,  $A^\top = A$ .
- Hessian matrices are symmetric.

# Diagonal matrices

For a diagonal matrix

$$D = \begin{bmatrix} d_1 & 0 & \dots & 0 \\ 0 & d_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \vdots & d_n \end{bmatrix}$$

the quadratic form

$$\mathbf{x}^T D \mathbf{x} = \begin{bmatrix} x_1 & x_2 & \dots & x_n \end{bmatrix} \begin{bmatrix} d_1 & 0 & \dots & 0 \\ 0 & d_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \vdots & d_n \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

is just  $d_1 x_1^2 + d_2 x_2^2 + \dots + d_n x_n^2$ .

# Diagonal matrices

- If  $d_1, \dots, d_n$  are all nonnegative, then  $d_1x_1^2 + d_2x_2^2 + \dots + d_nx_n^2$  must be nonnegative for any  $x$ , so  $D \succeq 0$ :  $D$  is positive semidefinite.

# Diagonal matrices

- If  $d_1, \dots, d_n$  are all nonnegative, then  $d_1x_1^2 + d_2x_2^2 + \dots + d_nx_n^2$  must be nonnegative for any  $x$ , so  $D \succeq 0$ :  $D$  is positive semidefinite.
- If  $d_1, \dots, d_n$  are all positive, then  $d_1x_1^2 + d_2x_2^2 + \dots + d_nx_n^2$  can only be 0 if  $x = 0$ , so  $D \succ 0$ :  $D$  is positive definite.

# Diagonal matrices

- If  $d_1, \dots, d_n$  are all nonnegative, then  $d_1x_1^2 + d_2x_2^2 + \dots + d_nx_n^2$  must be nonnegative for any  $\mathbf{x}$ , so  $D \succeq 0$ :  $D$  is positive semidefinite.
- If  $d_1, \dots, d_n$  are all positive, then  $d_1x_1^2 + d_2x_2^2 + \dots + d_nx_n^2$  can only be 0 if  $\mathbf{x} = \mathbf{0}$ , so  $D \succ 0$ :  $D$  is positive definite.
- If  $d_1, \dots, d_n \neq 0$ , then  $D \preceq 0$ , and if  $d_1, \dots, d_n < 0$ , then  $D \prec 0$ .

# Diagonal matrices

- If  $d_1, \dots, d_n$  are all nonnegative, then  $d_1x_1^2 + d_2x_2^2 + \dots + d_nx_n^2$  must be nonnegative for any  $\mathbf{x}$ , so  $D \succeq 0$ :  $D$  is positive semidefinite.
- If  $d_1, \dots, d_n$  are all positive, then  $d_1x_1^2 + d_2x_2^2 + \dots + d_nx_n^2$  can only be 0 if  $\mathbf{x} = \mathbf{0}$ , so  $D \succ 0$ :  $D$  is positive definite.
- If  $d_1, \dots, d_n \neq 0$ , then  $D \preceq 0$ , and if  $d_1, \dots, d_n < 0$ , then  $D \prec 0$ .
- $D$  is indefinite if the signs of  $d_1, \dots, d_n$  are mixed.

# Diagonal matrices

- If  $d_1, \dots, d_n$  are all nonnegative, then  $d_1x_1^2 + d_2x_2^2 + \dots + d_nx_n^2$  must be nonnegative for any  $\mathbf{x}$ , so  $D \succeq 0$ :  $D$  is positive semidefinite.
- If  $d_1, \dots, d_n$  are all positive, then  $d_1x_1^2 + d_2x_2^2 + \dots + d_nx_n^2$  can only be 0 if  $\mathbf{x} = \mathbf{0}$ , so  $D \succ 0$ :  $D$  is positive definite.
- If  $d_1, \dots, d_n \neq 0$ , then  $D \preceq 0$ , and if  $d_1, \dots, d_n < 0$ , then  $D \prec 0$ .
- $D$  is indefinite if the signs of  $d_1, \dots, d_n$  are mixed.



# Diagonal matrices

- If  $d_1, \dots, d_n$  are all nonnegative, then  $d_1x_1^2 + d_2x_2^2 + \dots + d_nx_n^2$  must be nonnegative for any  $\mathbf{x}$ , so  $D \succeq 0$ :  $D$  is positive semidefinite.
- If  $d_1, \dots, d_n$  are all positive, then  $d_1x_1^2 + d_2x_2^2 + \dots + d_nx_n^2$  can only be 0 if  $\mathbf{x} = \mathbf{0}$ , so  $D \succ 0$ :  $D$  is positive definite.
- If  $d_1, \dots, d_n \neq 0$ , then  $D \preceq 0$ , and if  $d_1, \dots, d_n < 0$ , then  $D \prec 0$ .
- $D$  is indefinite if the signs of  $d_1, \dots, d_n$  are mixed.

**Example:** Consider the function  $f(x, y) = x^2 + 2y^2$ .

- The gradient  $\nabla f(x, y) = (2x, 4y)$ .

# Diagonal matrices

- If  $d_1, \dots, d_n$  are all nonnegative, then  $d_1x_1^2 + d_2x_2^2 + \dots + d_nx_n^2$  must be nonnegative for any  $\mathbf{x}$ , so  $D \succeq 0$ :  $D$  is positive semidefinite.
- If  $d_1, \dots, d_n$  are all positive, then  $d_1x_1^2 + d_2x_2^2 + \dots + d_nx_n^2$  can only be 0 if  $\mathbf{x} = \mathbf{0}$ , so  $D \succ 0$ :  $D$  is positive definite.
- If  $d_1, \dots, d_n \neq 0$ , then  $D \preceq 0$ , and if  $d_1, \dots, d_n < 0$ , then  $D \prec 0$ .
- $D$  is indefinite if the signs of  $d_1, \dots, d_n$  are mixed.

**Example:** Consider the function  $f(x, y) = x^2 + 2y^2$ .

- The gradient  $\nabla f(x, y) = (2x, 4y)$ .
- The Hessian matrix of  $f$  is:

$$\mathbf{H}_f(x, y) = \begin{bmatrix} 2 & 0 \\ 0 & 4 \end{bmatrix}$$

# Diagonal matrices

- If  $d_1, \dots, d_n$  are all nonnegative, then  $d_1x_1^2 + d_2x_2^2 + \dots + d_nx_n^2$  must be nonnegative for any  $\mathbf{x}$ , so  $D \succeq 0$ :  $D$  is positive semidefinite.
- If  $d_1, \dots, d_n$  are all positive, then  $d_1x_1^2 + d_2x_2^2 + \dots + d_nx_n^2$  can only be 0 if  $\mathbf{x} = \mathbf{0}$ , so  $D \succ 0$ :  $D$  is positive definite.
- If  $d_1, \dots, d_n \neq 0$ , then  $D \preceq 0$ , and if  $d_1, \dots, d_n < 0$ , then  $D \prec 0$ .
- $D$  is indefinite if the signs of  $d_1, \dots, d_n$  are mixed.

**Example:** Consider the function  $f(x, y) = x^2 + 2y^2$ .

- The gradient  $\nabla f(x, y) = (2x, 4y)$ .
- The Hessian matrix of  $f$  is:

$$\mathbf{H}_f(x, y) = \begin{bmatrix} 2 & 0 \\ 0 & 4 \end{bmatrix}$$

- For an arbitrary  $\mathbf{x} \in \mathbb{R}^2$ , we have

$$\mathbf{x}^\top \begin{bmatrix} 2 & 0 \\ 0 & 4 \end{bmatrix} \mathbf{x} = 2x_1^2 + 4x_2^2 > 0 \text{ for all } \mathbf{x} \neq \mathbf{0}.$$

# Diagonal matrices

- If  $d_1, \dots, d_n$  are all nonnegative, then  $d_1x_1^2 + d_2x_2^2 + \dots + d_nx_n^2$  must be nonnegative for any  $\mathbf{x}$ , so  $D \succeq 0$ :  $D$  is positive semidefinite.
- If  $d_1, \dots, d_n$  are all positive, then  $d_1x_1^2 + d_2x_2^2 + \dots + d_nx_n^2$  can only be 0 if  $\mathbf{x} = \mathbf{0}$ , so  $D \succ 0$ :  $D$  is positive definite.
- If  $d_1, \dots, d_n \neq 0$ , then  $D \preceq 0$ , and if  $d_1, \dots, d_n < 0$ , then  $D \prec 0$ .
- $D$  is indefinite if the signs of  $d_1, \dots, d_n$  are mixed.

**Example:** Consider the function  $f(x, y) = x^2 + 2y^2$ .

- The gradient  $\nabla f(x, y) = (2x, 4y)$ .
- The Hessian matrix of  $f$  is:

$$\mathbf{H}_f(x, y) = \begin{bmatrix} 2 & 0 \\ 0 & 4 \end{bmatrix}$$

- For an arbitrary  $\mathbf{x} \in \mathbb{R}^2$ , we have

$$\mathbf{x}^\top \begin{bmatrix} 2 & 0 \\ 0 & 4 \end{bmatrix} \mathbf{x} = 2x_1^2 + 4x_2^2 > 0 \text{ for all } \mathbf{x} \neq \mathbf{0}.$$

- So,  $\mathbf{H}_f(x, y) \succ 0$  for all  $(x, y) \in \mathbb{R}^2$ .  $\mathbf{H}_f(x, y)$  is positive definite.

# Positive definiteness and eigenvalues

- For an  $n \times n$  matrix  $A$ , if a nonzero vector  $\mathbf{x} \in \mathbb{R}^n$  satisfies

$$A\mathbf{x} = \lambda\mathbf{x}$$

for some scalar  $\lambda \in \mathbb{R}$ , we call  $\lambda$  an eigenvalue of  $A$  and  $\mathbf{x}$  its associated eigenvector.

- If  $A$  is an  $n \times n$  symmetric matrix, then it can be factored as

$$A = Q^T \Lambda Q = Q^T \begin{bmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \vdots & \lambda_n \end{bmatrix} Q$$

where  $\lambda_1, \dots, \lambda_n$  are the eigenvalues of  $A$  and the columns of  $Q$  are the corresponding eigenvectors.

# Positive definiteness and eigenvalues

- Apply to the quadratic form  $\mathbf{x}^\top A \mathbf{x}$ , we get

$$\mathbf{x}^\top A \mathbf{x} = \mathbf{x}^\top Q^\top \Lambda Q \mathbf{x} = (Q \mathbf{x})^\top \Lambda (Q \mathbf{x})$$

# Positive definiteness and eigenvalues

- Apply to the quadratic form  $\mathbf{x}^\top A \mathbf{x}$ , we get

$$\mathbf{x}^\top A \mathbf{x} = \mathbf{x}^\top Q^\top \Lambda Q \mathbf{x} = (Q \mathbf{x})^\top \Lambda (Q \mathbf{x})$$

- If we substitute  $\mathbf{y} = Q \mathbf{x}$  (converting to a different basis), the quadratic form becomes diagonal:

$$\mathbf{x}^\top A \mathbf{x} = \mathbf{y}^\top \Lambda \mathbf{y} = \lambda_1 y_1^2 + \lambda_2 y_2^2 + \dots + \lambda_n y_n^2$$

# Positive definiteness and eigenvalues

- Apply to the quadratic form  $\mathbf{x}^\top A \mathbf{x}$ , we get

$$\mathbf{x}^\top A \mathbf{x} = \mathbf{x}^\top Q^\top \Lambda Q \mathbf{x} = (Q \mathbf{x})^\top \Lambda (Q \mathbf{x})$$

- If we substitute  $\mathbf{y} = Q \mathbf{x}$  (converting to a different basis), the quadratic form becomes diagonal:

$$\mathbf{x}^\top A \mathbf{x} = \mathbf{y}^\top \Lambda \mathbf{y} = \lambda_1 y_1^2 + \lambda_2 y_2^2 + \dots + \lambda_n y_n^2$$



# Positive definiteness and eigenvalues

- Apply to the quadratic form  $\mathbf{x}^\top A \mathbf{x}$ , we get

$$\mathbf{x}^\top A \mathbf{x} = \mathbf{x}^\top Q^\top \Lambda Q \mathbf{x} = (Q \mathbf{x})^\top \Lambda (Q \mathbf{x})$$

- If we substitute  $\mathbf{y} = Q \mathbf{x}$  (converting to a different basis), the quadratic form becomes diagonal:

$$\mathbf{x}^\top A \mathbf{x} = \mathbf{y}^\top \Lambda \mathbf{y} = \lambda_1 y_1^2 + \lambda_2 y_2^2 + \dots + \lambda_n y_n^2$$

We can classify the matrix  $A$  by looking at the eigenvalues of  $A$ .

- $A \succeq 0$  if  $\lambda_1, \lambda_2, \dots, \lambda_n \geq 0$

# Positive definiteness and eigenvalues

- Apply to the quadratic form  $\mathbf{x}^\top A \mathbf{x}$ , we get

$$\mathbf{x}^\top A \mathbf{x} = \mathbf{x}^\top Q^\top \Lambda Q \mathbf{x} = (Q \mathbf{x})^\top \Lambda (Q \mathbf{x})$$

- If we substitute  $\mathbf{y} = Q \mathbf{x}$  (converting to a different basis), the quadratic form becomes diagonal:

$$\mathbf{x}^\top A \mathbf{x} = \mathbf{y}^\top \Lambda \mathbf{y} = \lambda_1 y_1^2 + \lambda_2 y_2^2 + \dots + \lambda_n y_n^2$$

We can classify the matrix  $A$  by looking at the eigenvalues of  $A$ .

- $A \succeq 0$  if  $\lambda_1, \lambda_2, \dots, \lambda_n \geq 0$
- $A \succ 0$  if  $\lambda_1, \lambda_2, \dots, \lambda_n > 0$

# Positive definiteness and eigenvalues

- Apply to the quadratic form  $\mathbf{x}^\top A \mathbf{x}$ , we get

$$\mathbf{x}^\top A \mathbf{x} = \mathbf{x}^\top Q^\top \Lambda Q \mathbf{x} = (Q \mathbf{x})^\top \Lambda (Q \mathbf{x})$$

- If we substitute  $\mathbf{y} = Q \mathbf{x}$  (converting to a different basis), the quadratic form becomes diagonal:

$$\mathbf{x}^\top A \mathbf{x} = \mathbf{y}^\top \Lambda \mathbf{y} = \lambda_1 y_1^2 + \lambda_2 y_2^2 + \dots + \lambda_n y_n^2$$

We can classify the matrix  $A$  by looking at the eigenvalues of  $A$ .

- $A \succeq 0$  if  $\lambda_1, \lambda_2, \dots, \lambda_n \geq 0$
- $A \succ 0$  if  $\lambda_1, \lambda_2, \dots, \lambda_n > 0$
- $A \preceq 0$  if  $\lambda_1, \lambda_2, \dots, \lambda_n \leq 0$

# Positive definiteness and eigenvalues

- Apply to the quadratic form  $\mathbf{x}^\top A \mathbf{x}$ , we get

$$\mathbf{x}^\top A \mathbf{x} = \mathbf{x}^\top Q^\top \Lambda Q \mathbf{x} = (Q \mathbf{x})^\top \Lambda (Q \mathbf{x})$$

- If we substitute  $\mathbf{y} = Q \mathbf{x}$  (converting to a different basis), the quadratic form becomes diagonal:

$$\mathbf{x}^\top A \mathbf{x} = \mathbf{y}^\top \Lambda \mathbf{y} = \lambda_1 y_1^2 + \lambda_2 y_2^2 + \dots + \lambda_n y_n^2$$

We can classify the matrix  $A$  by looking at the eigenvalues of  $A$ .

- $A \succeq 0$  if  $\lambda_1, \lambda_2, \dots, \lambda_n \geq 0$
- $A \succ 0$  if  $\lambda_1, \lambda_2, \dots, \lambda_n > 0$
- $A \preceq 0$  if  $\lambda_1, \lambda_2, \dots, \lambda_n \leq 0$
- $A \prec 0$  if  $\lambda_1, \lambda_2, \dots, \lambda_n < 0$

# Positive definiteness and eigenvalues

- Apply to the quadratic form  $\mathbf{x}^\top A \mathbf{x}$ , we get

$$\mathbf{x}^\top A \mathbf{x} = \mathbf{x}^\top Q^\top \Lambda Q \mathbf{x} = (Q \mathbf{x})^\top \Lambda (Q \mathbf{x})$$

- If we substitute  $\mathbf{y} = Q \mathbf{x}$  (converting to a different basis), the quadratic form becomes diagonal:

$$\mathbf{x}^\top A \mathbf{x} = \mathbf{y}^\top \Lambda \mathbf{y} = \lambda_1 y_1^2 + \lambda_2 y_2^2 + \dots + \lambda_n y_n^2$$

We can classify the matrix  $A$  by looking at the eigenvalues of  $A$ .

- $A \succeq 0$  if  $\lambda_1, \lambda_2, \dots, \lambda_n \geq 0$
- $A \succ 0$  if  $\lambda_1, \lambda_2, \dots, \lambda_n > 0$
- $A \preceq 0$  if  $\lambda_1, \lambda_2, \dots, \lambda_n \leq 0$
- $A \prec 0$  if  $\lambda_1, \lambda_2, \dots, \lambda_n < 0$
- $A$  is indefinite if it has both positive and negative eigenvalues.

# Table of Contents

- 1 Introduction
- 2 Matrix calculus
- 3 Positive definite matrices
- 4 Optimality conditions**
- 5 Constrained vs unconstrained optimization
- 6 Convex vs nonconvex optimization
- 7 Smooth vs nonsmooth optimization
- 8 First-order methods

# Optimality conditions for local vs global optima

- For continuous, twice differentiable functions, we can characterize the **points** which correspond to **local optima**.

# Optimality conditions for local vs global optima

- For continuous, twice differentiable functions, we can characterize the **points** which correspond to **local optima**.
- Let  $\mathbf{g}(\boldsymbol{\theta}) = \nabla \mathcal{L}(\boldsymbol{\theta})$  be the **gradient** vector, and  $\mathbf{H}(\boldsymbol{\theta}) = \nabla^2 \mathcal{L}(\boldsymbol{\theta})$  be the **Hessian** matrix.



# Optimality conditions for local vs global optima

- For continuous, twice differentiable functions, we can characterize the **points** which correspond to **local optima**.
- Let  $\mathbf{g}(\boldsymbol{\theta}) = \nabla \mathcal{L}(\boldsymbol{\theta})$  be the **gradient** vector, and  $\mathbf{H}(\boldsymbol{\theta}) = \nabla^2 \mathcal{L}(\boldsymbol{\theta})$  be the **Hessian** matrix.
- Consider a point  $\boldsymbol{\theta}^* \in \mathbb{R}^D$ , and let  $\mathbf{g}^* = \mathbf{g}(\boldsymbol{\theta})|_{\boldsymbol{\theta}^*}$  be the gradient at that point, and  $\mathbf{H}^* = \mathbf{H}(\boldsymbol{\theta})|_{\boldsymbol{\theta}^*}$  be the corresponding Hessian.

# Optimality conditions for local vs global optima

- For continuous, twice differentiable functions, we can characterize the **points** which correspond to **local optima**.
- Let  $\mathbf{g}(\boldsymbol{\theta}) = \nabla \mathcal{L}(\boldsymbol{\theta})$  be the **gradient** vector, and  $\mathbf{H}(\boldsymbol{\theta}) = \nabla^2 \mathcal{L}(\boldsymbol{\theta})$  be the **Hessian** matrix.
- Consider a point  $\boldsymbol{\theta}^* \in \mathbb{R}^D$ , and let  $\mathbf{g}^* = \mathbf{g}(\boldsymbol{\theta})|_{\boldsymbol{\theta}^*}$  be the gradient at that point, and  $\mathbf{H}^* = \mathbf{H}(\boldsymbol{\theta})|_{\boldsymbol{\theta}^*}$  be the corresponding Hessian.
- **Necessary conditions:** If  $\boldsymbol{\theta}^*$  is a local minimum, then we must have  $\mathbf{g}^* = \mathbf{0}$  (i.e.,  $\boldsymbol{\theta}^*$  must be a **stationary point**), and  $\mathbf{H}^*$  must be positive semi-definite.

# Optimality conditions for local vs global optima

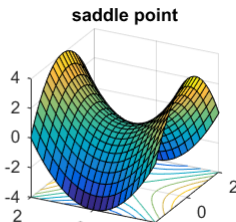
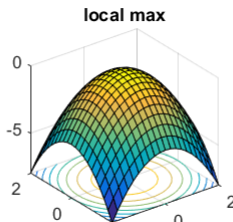
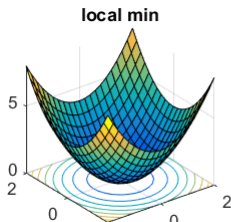
- For continuous, twice differentiable functions, we can characterize the **points** which correspond to **local optima**.
- Let  $\mathbf{g}(\boldsymbol{\theta}) = \nabla \mathcal{L}(\boldsymbol{\theta})$  be the **gradient** vector, and  $\mathbf{H}(\boldsymbol{\theta}) = \nabla^2 \mathcal{L}(\boldsymbol{\theta})$  be the **Hessian** matrix.
- Consider a point  $\boldsymbol{\theta}^* \in \mathbb{R}^D$ , and let  $\mathbf{g}^* = \mathbf{g}(\boldsymbol{\theta})|_{\boldsymbol{\theta}^*}$  be the gradient at that point, and  $\mathbf{H}^* = \mathbf{H}(\boldsymbol{\theta})|_{\boldsymbol{\theta}^*}$  be the corresponding Hessian.
- **Necessary conditions:** If  $\boldsymbol{\theta}^*$  is a local minimum, then we must have  $\mathbf{g}^* = \mathbf{0}$  (i.e.,  $\boldsymbol{\theta}^*$  must be a **stationary point**), and  $\mathbf{H}^*$  must be positive semi-definite.
- **Sufficient conditions:** If  $\mathbf{g}^* = \mathbf{0}$  and  $\mathbf{H}^*$  is positive definite, then  $\boldsymbol{\theta}^*$  is a local optimum.

# Optimality conditions for local vs global optima

- ① **Necessary conditions:** If  $\theta^*$  is a local minimum, then we must have  $g^* = 0$  (i.e.,  $\theta^*$  must be a **stationary point**), and  $H^*$  must be positive semi-definite.
- Suppose we were at a point  $\theta^*$  at which the gradient is non-zero.
  - At such a point, we could decrease the function by following the negative gradient a small distance, so this would not be optimal.
  - So the gradient must be zero.

# Optimality conditions for local vs global optima

- ① **Necessary conditions:** If  $\theta^*$  is a local minimum, then we must have  $g^* = 0$  (i.e.,  $\theta^*$  must be a **stationary point**), and  $H^*$  must be positive semi-definite.
  - Suppose we were at a point  $\theta^*$  at which the gradient is non-zero.
  - At such a point, we could decrease the function by following the negative gradient a small distance, so this would not be optimal.
  - So the gradient must be zero.
- ② **Sufficient conditions:** If  $g^* = 0$  and  $H^*$  is positive definite, then  $\theta^*$  is a local optimum.
  - Why a zero gradient is not sufficient?
  - The stationary point could be a local minimum, local maximum, or **saddle point**.



- We classify a stationary point of a function  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  as a **global minimizer** if the Hessian matrix of  $f$  is positive semidefinite **everywhere**,
- and as a **global maximizer** if the Hessian matrix is negative semidefinite everywhere.
- If the Hessian matrix is positive definite, or negative definite, the minimizer and maximizer (respectively) is strict.

# Example

Let  $f(x_1, x_2) = (x_1^2 + x_2^2 - 1)^2 + (x_2^2 - 1)^2$ .

- The gradient is  $\nabla f(\mathbf{x}) = 4 \begin{pmatrix} (x_1^2 + x_2^2 - 1)x_1 \\ (x_1^2 + x_2^2 - 1)x_2 + (x_2^2 - 1)x_2 \end{pmatrix}$

# Example

Let  $f(x_1, x_2) = (x_1^2 + x_2^2 - 1)^2 + (x_2^2 - 1)^2$ .

- The gradient is  $\nabla f(\mathbf{x}) = 4 \begin{pmatrix} (x_1^2 + x_2^2 - 1)x_1 \\ (x_1^2 + x_2^2 - 1)x_2 + (x_2^2 - 1)x_2 \end{pmatrix}$
- The stationary points are  $(0,0)$ ,  $(1,0)$ ,  $(-1,0)$ ,  $(0,1)$ ,  $(0,-1)$ .



# Example

Let  $f(x_1, x_2) = (x_1^2 + x_2^2 - 1)^2 + (x_2^2 - 1)^2$ .

- The gradient is  $\nabla f(\mathbf{x}) = 4 \begin{pmatrix} (x_1^2 + x_2^2 - 1)x_1 \\ (x_1^2 + x_2^2 - 1)x_2 + (x_2^2 - 1)x_2 \end{pmatrix}$
- The stationary points are  $(0,0)$ ,  $(1,0)$ ,  $(-1,0)$ ,  $(0,1)$ ,  $(0,-1)$ .
- The Hessian is  $\nabla^2 f(\mathbf{x}) = 4 \begin{pmatrix} 3x_1^2 + x_2^2 - 1 & 2x_1x_2 \\ 2x_1x_2 & x_1^2 + 6x_2^2 - 2 \end{pmatrix}$

# Example

Let  $f(x_1, x_2) = (x_1^2 + x_2^2 - 1)^2 + (x_2^2 - 1)^2$ .

- The gradient is  $\nabla f(\mathbf{x}) = 4 \begin{pmatrix} (x_1^2 + x_2^2 - 1)x_1 \\ (x_1^2 + x_2^2 - 1)x_2 + (x_2^2 - 1)x_2 \end{pmatrix}$
- The stationary points are  $(0,0)$ ,  $(1,0)$ ,  $(-1,0)$ ,  $(0,1)$ ,  $(0,-1)$ .
- The Hessian is  $\nabla^2 f(\mathbf{x}) = 4 \begin{pmatrix} 3x_1^2 + x_2^2 - 1 & 2x_1x_2 \\ 2x_1x_2 & x_1^2 + 6x_2^2 - 2 \end{pmatrix}$
- Since  $\nabla^2 f(0,0) = 4 \begin{pmatrix} -1 & 0 \\ 0 & -2 \end{pmatrix} \prec 0$ , it follows that  $(0,0)$  is a **strict local maximum** point.

# Example

Let  $f(x_1, x_2) = (x_1^2 + x_2^2 - 1)^2 + (x_2^2 - 1)^2$ .

- The gradient is  $\nabla f(\mathbf{x}) = 4 \begin{pmatrix} (x_1^2 + x_2^2 - 1)x_1 \\ (x_1^2 + x_2^2 - 1)x_2 + (x_2^2 - 1)x_2 \end{pmatrix}$
- The stationary points are  $(0,0)$ ,  $(1,0)$ ,  $(-1,0)$ ,  $(0,1)$ ,  $(0,-1)$ .
- The Hessian is  $\nabla^2 f(\mathbf{x}) = 4 \begin{pmatrix} 3x_1^2 + x_2^2 - 1 & 2x_1x_2 \\ 2x_1x_2 & x_1^2 + 6x_2^2 - 2 \end{pmatrix}$
- Since  $\nabla^2 f(0,0) = 4 \begin{pmatrix} -1 & 0 \\ 0 & -2 \end{pmatrix} \prec 0$ , it follows that  $(0,0)$  is a **strict local maximum** point.
- By the fact that  $f(x_1, 0) = (x_1^2 - 1)^2 + 1 \rightarrow \infty$  as  $x_1 \rightarrow \infty$ , the function is not bounded above, and thus  $(0,0)$  is not a global maximum point.

# Example

- $\nabla^2 f(1, 0) = \nabla^2 f(-1, 0) = 4 \begin{pmatrix} 2 & 0 \\ 0 & -1 \end{pmatrix}$ , which is an indefinite matrix.

# Example

- $\nabla^2 f(1, 0) = \nabla^2 f(-1, 0) = 4 \begin{pmatrix} 2 & 0 \\ 0 & -1 \end{pmatrix}$ , which is an indefinite matrix.
- Hence  $(1, 0)$  and  $(-1, 0)$  are **saddle points**.

# Example

- $\nabla^2 f(1, 0) = \nabla^2 f(-1, 0) = 4 \begin{pmatrix} 2 & 0 \\ 0 & -1 \end{pmatrix}$ , which is an indefinite matrix.
- Hence  $(1, 0)$  and  $(-1, 0)$  are **saddle points**.
- $\nabla^2 f(0, 1) = \nabla^2 f(0, -1) = 4 \begin{pmatrix} 0 & 0 \\ 0 & 4 \end{pmatrix}$ , which is positive semidefinite.

# Example

- $\nabla^2 f(1, 0) = \nabla^2 f(-1, 0) = 4 \begin{pmatrix} 2 & 0 \\ 0 & -1 \end{pmatrix}$ , which is an indefinite matrix.
- Hence  $(1,0)$  and  $(-1,0)$  are **saddle points**.
- $\nabla^2 f(0, 1) = \nabla^2 f(0, -1) = 4 \begin{pmatrix} 0 & 0 \\ 0 & 4 \end{pmatrix}$ , which is positive semidefinite.
- The fact that the Hessian matrices of  $f$  at  $(0,1)$  and  $(0,-1)$  are positive semidefinite is **not enough** to conclude that these are local minimum points; they **might be** saddle points.

# Example

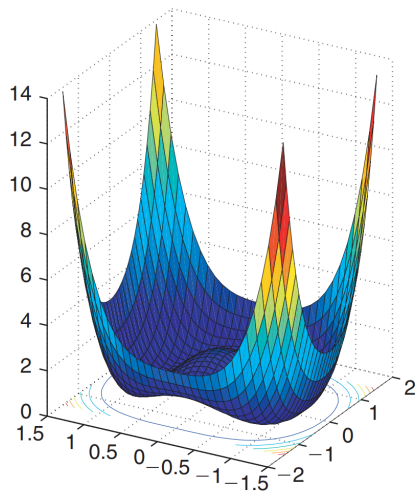
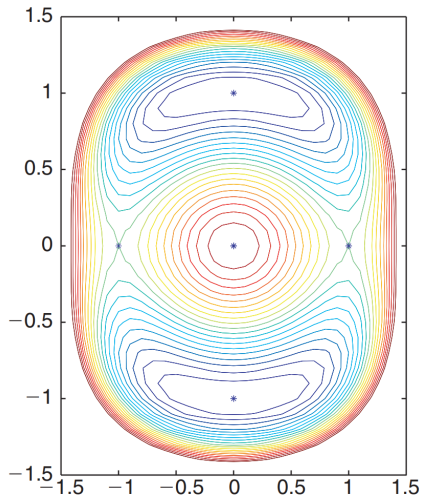
- $\nabla^2 f(1, 0) = \nabla^2 f(-1, 0) = 4 \begin{pmatrix} 2 & 0 \\ 0 & -1 \end{pmatrix}$ , which is an indefinite matrix.
- Hence  $(1, 0)$  and  $(-1, 0)$  are **saddle points**.
- $\nabla^2 f(0, 1) = \nabla^2 f(0, -1) = 4 \begin{pmatrix} 0 & 0 \\ 0 & 4 \end{pmatrix}$ , which is positive semidefinite.
- The fact that the Hessian matrices of  $f$  at  $(0, 1)$  and  $(0, -1)$  are positive semidefinite is **not enough** to conclude that these are local minimum points; they **might be** saddle points.
- However, in this case, since  $f(0, 1) = f(0, -1) = 0$  and the function is lower bounded by zero,  $(0, 1)$  and  $(0, -1)$  are **global minimum points**.



# Example

- $\nabla^2 f(1, 0) = \nabla^2 f(-1, 0) = 4 \begin{pmatrix} 2 & 0 \\ 0 & -1 \end{pmatrix}$ , which is an indefinite matrix.
- Hence  $(1, 0)$  and  $(-1, 0)$  are **saddle points**.
- $\nabla^2 f(0, 1) = \nabla^2 f(0, -1) = 4 \begin{pmatrix} 0 & 0 \\ 0 & 4 \end{pmatrix}$ , which is positive semidefinite.
- The fact that the Hessian matrices of  $f$  at  $(0, 1)$  and  $(0, -1)$  are positive semidefinite is **not enough** to conclude that these are local minimum points; they **might be** saddle points.
- However, in this case, since  $f(0, 1) = f(0, -1) = 0$  and the function is lower bounded by zero,  $(0, 1)$  and  $(0, -1)$  are **global minimum points**.
- Because there are two global minimum points, they are **nonstrict** global minima, but they are **strict** local minimum points, since each has a neighborhood in which it is the unique minimizer.

# Example



# Table of Contents

- 1 Introduction
- 2 Matrix calculus
- 3 Positive definite matrices
- 4 Optimality conditions
- 5 Constrained vs unconstrained optimization**
- 6 Convex vs nonconvex optimization
- 7 Smooth vs nonsmooth optimization
- 8 First-order methods

# Constrained vs unconstrained optimization

- In **unconstrained optimization**, we find any value in the parameter space  $\Theta$  that minimizes the loss.

# Constrained vs unconstrained optimization

- In **unconstrained optimization**, we find any value in the parameter space  $\Theta$  that minimizes the loss.
- We can also have a set of **constraints**  $\mathcal{C}$  on the allowable values.

# Constrained vs unconstrained optimization

- In **unconstrained optimization**, we find any value in the parameter space  $\Theta$  that minimizes the loss.
- We can also have a set of **constraints**  $\mathcal{C}$  on the allowable values.
- We partition the set of constraints  $\mathcal{C}$  into:
  - **Inequality constraints**:  $g_j(\boldsymbol{\theta}) \leq 0$  for  $j \in \mathcal{I}$ .
  - **Equality constraints**:  $h_k(\boldsymbol{\theta}) = 0$  for  $k \in \mathcal{E}$ .

# Constrained vs unconstrained optimization

- In **unconstrained optimization**, we find any value in the parameter space  $\Theta$  that minimizes the loss.
- We can also have a set of **constraints**  $\mathcal{C}$  on the allowable values.
- We partition the set of constraints  $\mathcal{C}$  into:
  - **Inequality constraints**:  $g_j(\boldsymbol{\theta}) \leq 0$  for  $j \in \mathcal{I}$ .
  - **Equality constraints**:  $h_k(\boldsymbol{\theta}) = 0$  for  $k \in \mathcal{E}$ .
- The **feasible set** is the subset of the parameter space that satisfies the constraints:

$$\mathcal{C} = \{\boldsymbol{\theta} : g_j(\boldsymbol{\theta}) \leq 0 : j \in \mathcal{I}, h_k(\boldsymbol{\theta}) = 0 : k \in \mathcal{E}\} \subseteq \mathbb{R}^D$$

# Constrained vs unconstrained optimization

- In **unconstrained optimization**, we find any value in the parameter space  $\Theta$  that minimizes the loss.
- We can also have a set of **constraints**  $\mathcal{C}$  on the allowable values.
- We partition the set of constraints  $\mathcal{C}$  into:
  - **Inequality constraints**:  $g_j(\boldsymbol{\theta}) \leq 0$  for  $j \in \mathcal{I}$ .
  - **Equality constraints**:  $h_k(\boldsymbol{\theta}) = 0$  for  $k \in \mathcal{E}$ .
- The **feasible set** is the subset of the parameter space that satisfies the constraints:

$$\mathcal{C} = \{\boldsymbol{\theta} : g_j(\boldsymbol{\theta}) \leq 0 : j \in \mathcal{I}, h_k(\boldsymbol{\theta}) = 0 : k \in \mathcal{E}\} \subseteq \mathbb{R}^D$$

- Our **constrained optimization** problem is

$$\hat{\boldsymbol{\theta}}^* = \underset{\boldsymbol{\theta} \in \mathcal{C}}{\operatorname{argmin}} \mathcal{L}(\boldsymbol{\theta})$$



# Constrained vs unconstrained optimization

- In **unconstrained optimization**, we find any value in the parameter space  $\Theta$  that minimizes the loss.
- We can also have a set of **constraints**  $\mathcal{C}$  on the allowable values.
- We partition the set of constraints  $\mathcal{C}$  into:
  - **Inequality constraints**:  $g_j(\theta) \leq 0$  for  $j \in \mathcal{I}$ .
  - **Equality constraints**:  $h_k(\theta) = 0$  for  $k \in \mathcal{E}$ .
- The **feasible set** is the subset of the parameter space that satisfies the constraints:

$$\mathcal{C} = \{\theta : g_j(\theta) \leq 0 : j \in \mathcal{I}, h_k(\theta) = 0 : k \in \mathcal{E}\} \subseteq \mathbb{R}^D$$

- Our **constrained optimization** problem is

$$\hat{\theta}^* = \underset{\theta \in \mathcal{C}}{\operatorname{argmin}} \mathcal{L}(\theta)$$

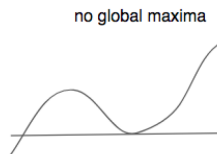
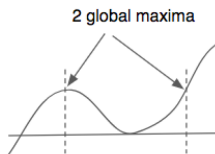
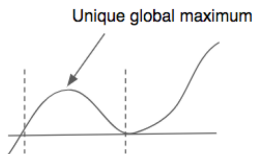
- If  $\mathcal{C} = \mathbb{R}^D$ , it is called **unconstrained optimization**.

# Constrained vs unconstrained optimization

- Constraints can change the number of optima of a function.

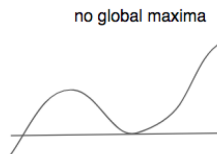
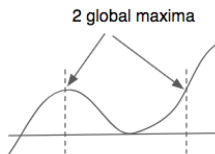
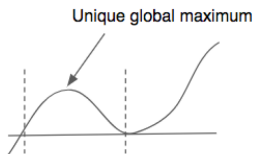
# Constrained vs unconstrained optimization

- Constraints can change the number of optima of a function.
- A function that was unbounded (no well-defined global maximum or minimum) can acquire multiple maxima or minima when we add constraints.



# Constrained vs unconstrained optimization

- Constraints can change the number of optima of a function.
- A function that was unbounded (no well-defined global maximum or minimum) can acquire multiple maxima or minima when we add constraints.



- The task of finding any point (regardless of its cost) in the feasible set is called **feasibility problem**.

# Table of Contents

- 1 Introduction
- 2 Matrix calculus
- 3 Positive definite matrices
- 4 Optimality conditions
- 5 Constrained vs unconstrained optimization
- 6 Convex vs nonconvex optimization**
- 7 Smooth vs nonsmooth optimization
- 8 First-order methods

# Convex sets

- In **convex optimization**, the objective is a convex function defined over a convex set.
- In such problems, every local minimum is also a global minimum.

# Convex sets

- In **convex optimization**, the objective is a convex function defined over a convex set.
- In such problems, every local minimum is also a global minimum.
- Many models are designed so that their training objectives are convex.

# Convex sets

- In **convex optimization**, the objective is a convex function defined over a convex set.
- In such problems, every local minimum is also a global minimum.
- Many models are designed so that their training objectives are convex.
- We say  $\mathcal{S}$  is a **convex set** if, for any  $x, x' \in \mathcal{S}$ , we have

$$\lambda x + (1 - \lambda)x' \in \mathcal{S}, \forall \lambda \in [0, 1]$$

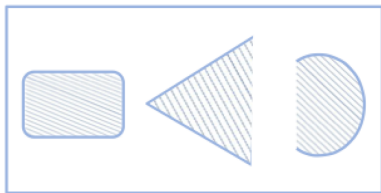


# Convex sets

- In **convex optimization**, the objective is a convex function defined over a convex set.
- In such problems, every local minimum is also a global minimum.
- Many models are designed so that their training objectives are convex.
- We say  $\mathcal{S}$  is a **convex set** if, for any  $x, x' \in \mathcal{S}$ , we have

$$\lambda x + (1 - \lambda)x' \in \mathcal{S}, \forall \lambda \in [0, 1]$$

- If we draw a line from  $x$  to  $x'$ , all points on the line lie inside the set.



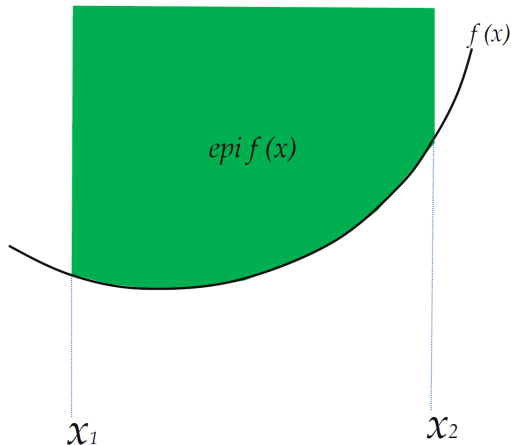
Convex



Not Convex

# Convex functions

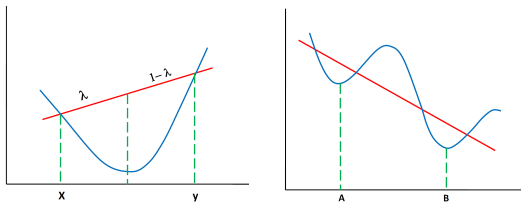
- $f$  is a **convex function** if its **epigraph** (the set of points above the function) defines a convex set.



# Convex functions

- $f(x)$  is called a convex function if it is defined on a convex set, and if, for any  $x, y \in \mathcal{S}$ , and for any  $0 \leq \lambda \leq 1$ , we have:

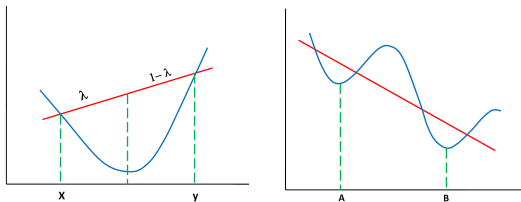
$$f(\lambda x + (1 - \lambda)y) \leq \lambda f(x) + (1 - \lambda)f(y)$$



# Convex functions

- $f(x)$  is called a convex function if it is defined on a convex set, and if, for any  $x, y \in \mathcal{S}$ , and for any  $0 \leq \lambda \leq 1$ , we have:

$$f(\lambda x + (1 - \lambda)y) \leq \lambda f(x) + (1 - \lambda)f(y)$$

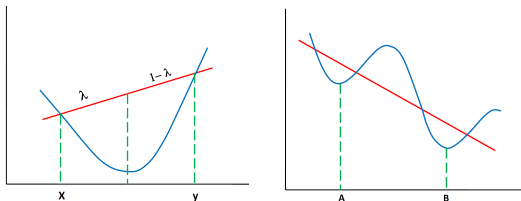


- A function is **strictly convex** if the inequality is strict.

# Convex functions

- $f(x)$  is called a convex function if it is defined on a convex set, and if, for any  $x, y \in \mathcal{S}$ , and for any  $0 \leq \lambda \leq 1$ , we have:

$$f(\lambda x + (1 - \lambda)y) \leq \lambda f(x) + (1 - \lambda)f(y)$$

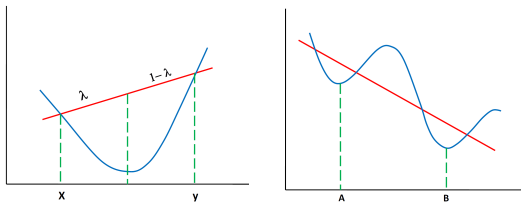


- A function is **strictly convex** if the inequality is strict.
- A function is **concave** if  $-f(x)$  is convex.

# Convex functions

- $f(x)$  is called a convex function if it is defined on a convex set, and if, for any  $x, y \in \mathcal{S}$ , and for any  $0 \leq \lambda \leq 1$ , we have:

$$f(\lambda x + (1 - \lambda)y) \leq \lambda f(x) + (1 - \lambda)f(y)$$

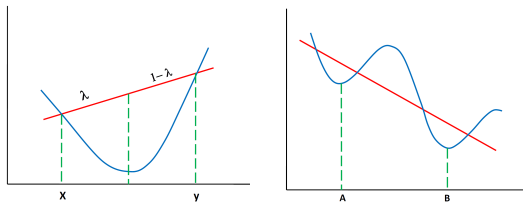


- A function is **strictly convex** if the inequality is strict.
- A function is **concave** if  $-f(x)$  is convex.
- A function can be neither convex nor concave.

# Convex functions

- $f(x)$  is called a convex function if it is defined on a convex set, and if, for any  $x, y \in \mathcal{S}$ , and for any  $0 \leq \lambda \leq 1$ , we have:

$$f(\lambda x + (1 - \lambda)y) \leq \lambda f(x) + (1 - \lambda)f(y)$$



- A function is **strictly convex** if the inequality is strict.
- A function is **concave** if  $-f(x)$  is convex.
- A function can be neither convex nor concave.
- Some examples of 1d convex functions:  $x^2$ ,  $e^{ax}$ ,  $-\log(x)$ ,  $x^a (a > 1, x > 0)$ ,  $|x|^a (a \geq 1)$ ,  $x \log x (x > 0)$ .

## Theorem

*Suppose  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  is twice differentiable over its domain. Then  $f$  is convex iff  $\mathbf{H} = \nabla^2 f(\mathbf{x})$  is positive semi-definite for all  $\mathbf{x} \in \text{dom}(f)$ . Furthermore,  $f$  is strictly convex if  $\mathbf{H}$  is positive definite.*



## Theorem

*Suppose  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  is twice differentiable over its domain. Then  $f$  is convex iff  $\mathbf{H} = \nabla^2 f(\mathbf{x})$  is positive semi-definite for all  $\mathbf{x} \in \text{dom}(f)$ . Furthermore,  $f$  is strictly convex if  $\mathbf{H}$  is positive definite.*

- For example, consider the quadratic form

$$f(\mathbf{x}) = \mathbf{x}^\top \mathbf{A} \mathbf{x}$$

## Theorem

*Suppose  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  is twice differentiable over its domain. Then  $f$  is convex iff  $\mathbf{H} = \nabla^2 f(\mathbf{x})$  is positive semi-definite for all  $\mathbf{x} \in \text{dom}(f)$ . Furthermore,  $f$  is strictly convex if  $\mathbf{H}$  is positive definite.*

- For example, consider the quadratic form

$$f(\mathbf{x}) = \mathbf{x}^\top \mathbf{A} \mathbf{x}$$

- This is convex if  $\mathbf{A}$  is positive semi-definite.

## Theorem

*Suppose  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  is twice differentiable over its domain. Then  $f$  is convex iff  $\mathbf{H} = \nabla^2 f(\mathbf{x})$  is positive semi-definite for all  $\mathbf{x} \in \text{dom}(f)$ . Furthermore,  $f$  is strictly convex if  $\mathbf{H}$  is positive definite.*

- For example, consider the quadratic form

$$f(\mathbf{x}) = \mathbf{x}^\top \mathbf{A} \mathbf{x}$$

- This is convex if  $\mathbf{A}$  is positive semi-definite.
- This is strictly convex if  $\mathbf{A}$  is positive definite.
- It is neither convex nor concave if

## Theorem

*Suppose  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  is twice differentiable over its domain. Then  $f$  is convex iff  $\mathbf{H} = \nabla^2 f(\mathbf{x})$  is positive semi-definite for all  $\mathbf{x} \in \text{dom}(f)$ . Furthermore,  $f$  is strictly convex if  $\mathbf{H}$  is positive definite.*

- For example, consider the quadratic form

$$f(\mathbf{x}) = \mathbf{x}^\top \mathbf{A} \mathbf{x}$$

- This is convex if  $\mathbf{A}$  is positive semi-definite.
- This is strictly convex if  $\mathbf{A}$  is positive definite.
- It is neither convex nor concave if  $\mathbf{A}$  has eigenvalues of mixed sign.

## Theorem

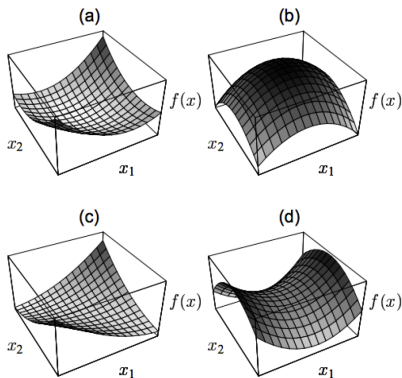
*Suppose  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  is twice differentiable over its domain. Then  $f$  is convex iff  $\mathbf{H} = \nabla^2 f(\mathbf{x})$  is positive semi-definite for all  $\mathbf{x} \in \text{dom}(f)$ . Furthermore,  $f$  is strictly convex if  $\mathbf{H}$  is positive definite.*

- For example, consider the quadratic form

$$f(\mathbf{x}) = \mathbf{x}^\top \mathbf{A} \mathbf{x}$$

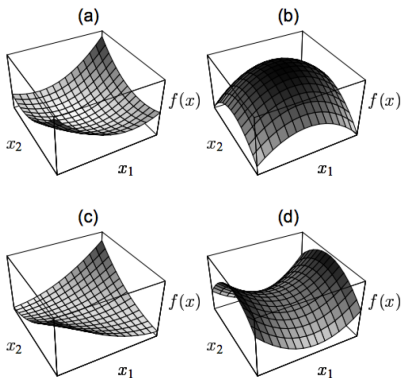
- This is convex if  $\mathbf{A}$  is positive semi-definite.
- This is strictly convex if  $\mathbf{A}$  is positive definite.
- It is neither convex nor concave if  $\mathbf{A}$  has eigenvalues of mixed sign.
- Intuitively, a convex function is shaped like a bowl.

# Convex functions



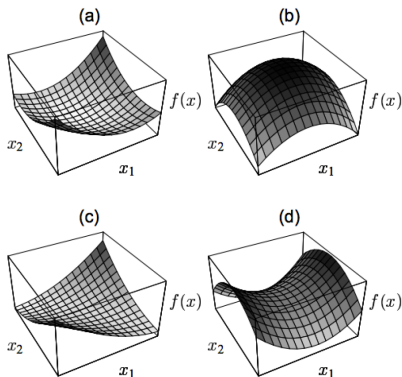
- The quadratic form  $f(x) = x^T A x$  in  $2d$ .

# Convex functions



- The quadratic form  $f(x) = x^T A x$  in  $2d$ .
- (a)  $A$  is positive definite, so  $f$  is strictly convex.

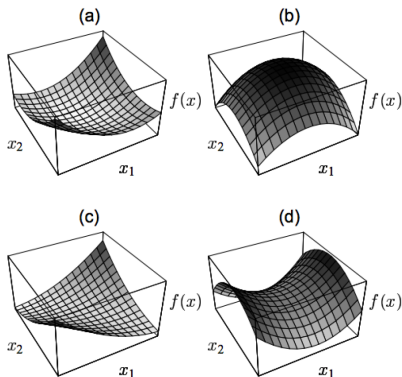
# Convex functions



- The quadratic form  $f(x) = x^T A x$  in  $2d$ .
- (a)  $A$  is positive definite, so  $f$  is strictly convex.
- (b)  $A$  is negative definite, so  $f$  is strictly concave.

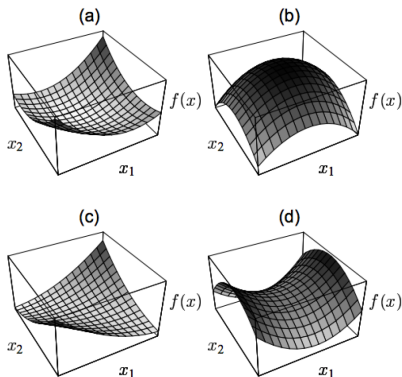


# Convex functions



- The quadratic form  $f(x) = x^T A x$  in  $2d$ .
- (a)  $A$  is positive definite, so  $f$  is strictly convex.
- (b)  $A$  is negative definite, so  $f$  is strictly concave.
- (c)  $A$  is positive semi-definite, but singular, so  $f$  is convex.

# Convex functions

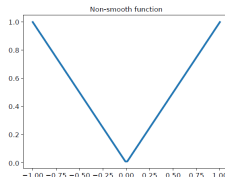
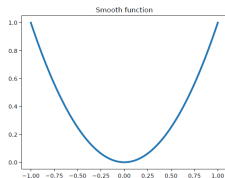


- The quadratic form  $f(x) = x^T A x$  in  $2d$ .
- (a)  $A$  is positive definite, so  $f$  is strictly convex.
- (b)  $A$  is negative definite, so  $f$  is strictly concave.
- (c)  $A$  is positive semi-definite, but singular, so  $f$  is convex.
- (d)  $A$  is indefinite, so  $f$  is neither convex nor concave.

# Table of Contents

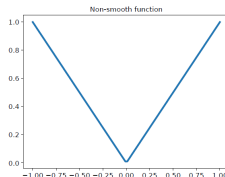
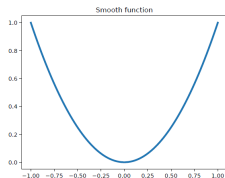
- 1 Introduction
- 2 Matrix calculus
- 3 Positive definite matrices
- 4 Optimality conditions
- 5 Constrained vs unconstrained optimization
- 6 Convex vs nonconvex optimization
- 7 Smooth vs nonsmooth optimization**
- 8 First-order methods

# Smooth vs nonsmooth optimization



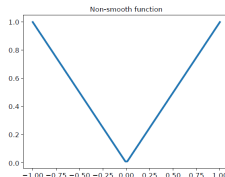
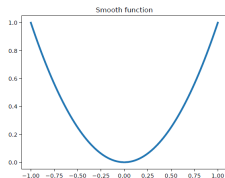
- In **smooth optimization**, the objective and constraints are continuously differentiable functions.

# Smooth vs nonsmooth optimization



- In **smooth optimization**, the objective and constraints are continuously differentiable functions.
- In **nonsmooth optimization**, there are some points where the gradient of the objective or the constraints is not well-defined.

# Smooth vs nonsmooth optimization

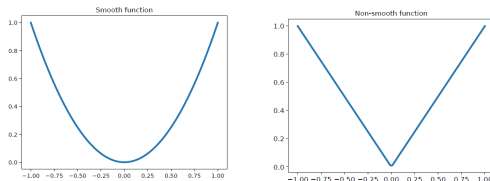


- In **smooth optimization**, the objective and constraints are continuously differentiable functions.
- In **nonsmooth optimization**, there are some points where the gradient of the objective or the constraints is not well-defined.
- In some problems, we partition the objective into a part that contains smooth terms, and a part that contains the nonsmooth terms:

$$\mathcal{L}(\theta) = \mathcal{L}_s\theta + \mathcal{L}_r(\theta)$$

where  $\mathcal{L}_s$  is smooth (differentiable), and  $\mathcal{L}_r$  is nonsmooth (“rough”).

# Smooth vs nonsmooth optimization



- In **smooth optimization**, the objective and constraints are continuously differentiable functions.
- In **nonsmooth optimization**, there are some points where the gradient of the objective or the constraints is not well-defined.
- In some problems, we partition the objective into a part that contains smooth terms, and a part that contains the nonsmooth terms:

$$\mathcal{L}(\theta) = \mathcal{L}_s\theta + \mathcal{L}_r(\theta)$$

where  $\mathcal{L}_s$  is smooth (differentiable), and  $\mathcal{L}_r$  is nonsmooth (“rough”).

- In ML,  $\mathcal{L}_s$  is the train loss, and  $\mathcal{L}_r$  is a regularizer, like  $\ell_1$  norm of  $\theta$ .

# Smooth vs nonsmooth optimization

- For smooth functions, we can quantify the degree of smoothness using the **Lipschitz constant**.



# Smooth vs nonsmooth optimization

- For smooth functions, we can quantify the degree of smoothness using the **Lipschitz constant**.
- In the  $1d$  case, this is defined as any constant  $L \geq 0$  such that, for all real  $x_1$  and  $x_2$ , we have:

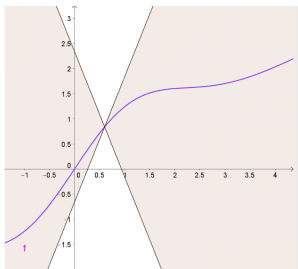
$$|f(x_1) - f(x_2)| \leq L|x_1 - x_2|$$

# Smooth vs nonsmooth optimization

- For smooth functions, we can quantify the degree of smoothness using the **Lipschitz constant**.
- In the  $1d$  case, this is defined as any constant  $L \geq 0$  such that, for all real  $x_1$  and  $x_2$ , we have:

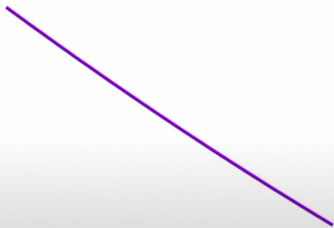
$$|f(x_1) - f(x_2)| \leq L|x_1 - x_2|$$

- Given a constant  $L$ , the function output cannot change by more than  $L$  if we change the function input by 1 unit.

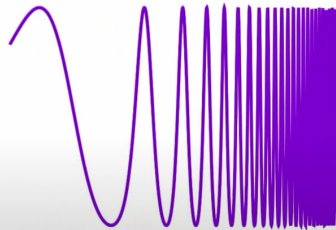


# Smooth vs nonsmooth optimization

$$f(x) = \frac{x^2}{100} - 2x + 1$$



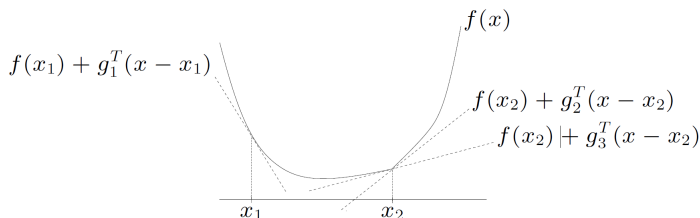
$$f(x) = \exp(\cos(e^{-x}))$$



# Subgradients

- We generalize the notion of a derivative to work with functions which have local discontinuities.
- For a convex function  $f : \mathbb{R}^n \rightarrow \mathbb{R}$ , we say  $\mathbf{g} \in \mathbb{R}^n$  is a **subgradient** of  $f$  at  $\mathbf{x} \in \text{dom}(f)$  if for all vector  $\mathbf{z} \in \text{dom}(f)$ ,

$$f(\mathbf{z}) \geq f(\mathbf{x}) + \mathbf{g}^T(\mathbf{z} - \mathbf{x})$$



- At  $x_1$ ,  $f$  is differentiable, and  $g_1$  is the unique subgradient at  $x_1$
- At  $x_2$ ,  $f$  is not differentiable, and there are many subgradients at  $x_2$ .

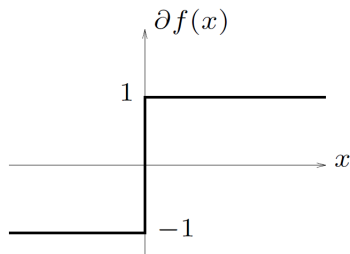
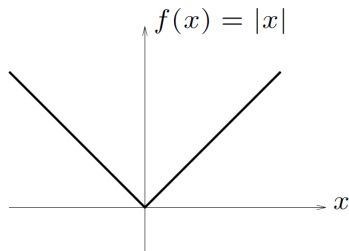
# Subgradients

- A function  $f$  is called subdifferentiable at  $x$ , if there is at least one subgradient at  $x$ .
- The set of such subgradients is called **subdifferential** of  $f$  at  $x$ , denoted as  $\partial f(x)$
- For example, consider  $f(x) = |x|$ . Its subdifferential is given by

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

where  $[-1, 1]$  here means any value between -1 and 1 (inclusive).

# Subgradients



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

where  $[-1, 1]$  here means any value between  $-1$  and  $1$  (inclusive).

# Table of Contents

- 1 Introduction
- 2 Matrix calculus
- 3 Positive definite matrices
- 4 Optimality conditions
- 5 Constrained vs unconstrained optimization
- 6 Convex vs nonconvex optimization
- 7 Smooth vs nonsmooth optimization
- 8 First-order methods**

# First-order methods

- We consider **iterative** optimization methods that leverage **first order** derivatives of the objective function.



# First-order methods

- We consider **iterative** optimization methods that leverage **first order** derivatives of the objective function.
- They compute which directions point “downhill”, but ignore curvature information.

# First-order methods

- We consider **iterative** optimization methods that leverage **first order** derivatives of the objective function.
- They compute which directions point “downhill”, but ignore curvature information.
- All these algorithms require the user specify a starting point  $\theta_0$ .

# First-order methods

- We consider **iterative** optimization methods that leverage **first order** derivatives of the objective function.
- They compute which directions point “downhill”, but ignore curvature information.
- All these algorithms require the user specify a starting point  $\theta_0$ .
- At each iteration  $t$ , an update is performed

$$\theta_{t+1} = \theta_t + \rho_t d_t$$

where  $\rho_t$  is the **step size** or **learning rate**, and  $d_t$  is a **descent direction**, e.g, the negative of the **gradient** given by  $g_t = \nabla_{\theta} \mathcal{L}(\theta)|_{\theta_t}$ .

# First-order methods

- We consider **iterative** optimization methods that leverage **first order** derivatives of the objective function.
- They compute which directions point “downhill”, but ignore curvature information.
- All these algorithms require the user specify a starting point  $\theta_0$ .
- At each iteration  $t$ , an update is performed

$$\theta_{t+1} = \theta_t + \rho_t d_t$$

where  $\rho_t$  is the **step size** or **learning rate**, and  $d_t$  is a **descent direction**, e.g, the negative of the **gradient** given by  $g_t = \nabla_{\theta} \mathcal{L}(\theta)|_{\theta_t}$ .

- The update steps are continued until a **stationary point** is reached, where the gradient is zero.

# Descent direction

- A direction  $d$  is a **descent direction** if there is a small enough (but nonzero) amount  $\rho$  that we can move in direction  $d$  and be guaranteed to decrease the function value.

# Descent direction

- A direction  $\mathbf{d}$  is a **descent direction** if there is a small enough (but nonzero) amount  $\rho$  that we can move in direction  $\mathbf{d}$  and be guaranteed to decrease the function value.
- We require there exists an  $\rho_{max} > 0$  such that

$$\mathcal{L}(\boldsymbol{\theta} + \rho \mathbf{d}) < \mathcal{L}(\boldsymbol{\theta})$$

for all  $0 < \rho < \rho_{max}$ .

# Descent direction

- A direction  $\mathbf{d}$  is a **descent direction** if there is a small enough (but nonzero) amount  $\rho$  that we can move in direction  $\mathbf{d}$  and be guaranteed to decrease the function value.
- We require there exists an  $\rho_{max} > 0$  such that

$$\mathcal{L}(\boldsymbol{\theta} + \rho \mathbf{d}) < \mathcal{L}(\boldsymbol{\theta})$$

for all  $0 < \rho < \rho_{max}$ .

- The gradient at the current iterate,

$$\mathbf{g}_t \triangleq \nabla \mathcal{L}(\boldsymbol{\theta})|_{\boldsymbol{\theta}_t} = \nabla \mathcal{L}(\boldsymbol{\theta}_t) = \mathbf{g}(\boldsymbol{\theta}_t)$$

points in the direction of maximal increase in  $f$ , so the negative gradient is a descent direction.

# Descent direction

- Any direction  $\mathbf{d}$  is also a descent direction if the angle  $\theta$  between  $\mathbf{d}$  and  $-\mathbf{g}_t$  is less than 90 degrees and satisfies

$$\mathbf{d}^T \mathbf{g}_t = \|\mathbf{d}\| \|\mathbf{g}_t\| \cos(\theta) < 0$$



# Descent direction

- Any direction  $\mathbf{d}$  is also a descent direction if the angle  $\theta$  between  $\mathbf{d}$  and  $-\mathbf{g}_t$  is less than 90 degrees and satisfies

$$\mathbf{d}^T \mathbf{g}_t = \|\mathbf{d}\| \|\mathbf{g}_t\| \cos(\theta) < 0$$

- The best choice would be to pick  $\mathbf{d}_t = -\mathbf{g}_t$ .

# Descent direction

- Any direction  $\mathbf{d}$  is also a descent direction if the angle  $\theta$  between  $\mathbf{d}$  and  $-\mathbf{g}_t$  is less than 90 degrees and satisfies

$$\mathbf{d}^T \mathbf{g}_t = \|\mathbf{d}\| \|\mathbf{g}_t\| \cos(\theta) < 0$$

- The best choice would be to pick  $\mathbf{d}_t = -\mathbf{g}_t$ .
- This is the direction of **steepest descent**.

# Step size (learning rate)

- The sequence of step sizes  $\{\rho_t\}$  is called the **learning rate schedule**.

# Step size (learning rate)

- The sequence of step sizes  $\{\rho_t\}$  is called the **learning rate schedule**.
- The simplest method is to use constant step size,  $\rho_t = \rho$ .

# Step size (learning rate)

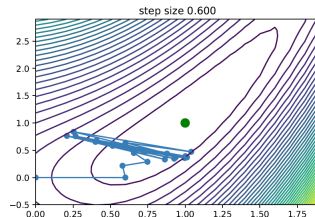
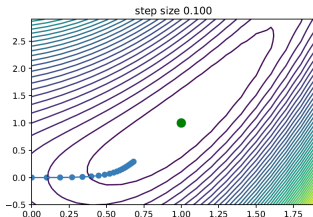
- The sequence of step sizes  $\{\rho_t\}$  is called the **learning rate schedule**.
- The simplest method is to use constant step size,  $\rho_t = \rho$ .
- However, if it is too large, the method may fail to converge. If it is too small, the function will converge but very slowly.

# Step size (learning rate)

- The sequence of step sizes  $\{\rho_t\}$  is called the **learning rate schedule**.
- The simplest method is to use constant step size,  $\rho_t = \rho$ .
- However, if it is too large, the method may fail to converge. If it is too small, the function will converge but very slowly.
- Example:

$$\mathcal{L}(\theta) = 0.5(\theta_1^2 - \theta_2)^2 + 0.5(\theta_1 - 1)^2$$

- Pick our descent direction  $\mathbf{d}_t = -\mathbf{g}_t$ . Consider  $\rho_t = 0.1$  vs  $\rho_t = 0.6$ :



- The **optimal step size** can be found by finding the value that maximally decreases the objective along the chosen direction by solving the 1d minimization problem

$$\rho_t = \operatorname{argmin}_{\rho > 0} \phi_t(\rho) = \operatorname{argmin}_{\rho > 0} \mathcal{L}(\boldsymbol{\theta}_t + \rho \mathbf{d}_t)$$

- This is **line search**: we are searching along the line defined by  $\mathbf{d}_t$ .
- $\phi_t(\rho) = \mathcal{L}(\boldsymbol{\theta}_t + \rho \mathbf{d}_t)$  is a convex function of an affine function of  $\rho$ , for fixed  $\boldsymbol{\theta}_t$  and  $\mathbf{d}$ .
- If the loss is convex, this subproblem is also convex.

# Line search

- Example, consider the quadratic loss

$$\mathcal{L}(\boldsymbol{\theta}) = \frac{1}{2} \boldsymbol{\theta}^\top \mathbf{A} \boldsymbol{\theta} + \mathbf{b}^\top \boldsymbol{\theta} + c$$

- Compute the derivatives of  $\phi(\rho) = \mathcal{L}(\boldsymbol{\theta} + \rho \mathbf{d})$  gives

$$\begin{aligned} \frac{d\phi(\rho)}{d\rho} &= \frac{d}{d\rho} \left[ \frac{1}{2} (\boldsymbol{\theta} + \rho \mathbf{d})^\top \mathbf{A} (\boldsymbol{\theta} + \rho \mathbf{d}) + \mathbf{b}^\top (\boldsymbol{\theta} + \rho \mathbf{d}) + c \right] \\ &= \mathbf{d}^\top \mathbf{A} (\boldsymbol{\theta} + \rho \mathbf{d}) + \mathbf{d}^\top \mathbf{b} \\ &= \mathbf{d}^\top (\mathbf{A} \boldsymbol{\theta} + \mathbf{b}) + \rho \mathbf{d}^\top \mathbf{A} \mathbf{d} \end{aligned}$$

- Solving for  $\frac{d\phi(\rho)}{d\rho} = 0$  gives

$$\rho = - \frac{\mathbf{d}^\top (\mathbf{A} \boldsymbol{\theta} + \mathbf{b})}{\mathbf{d}^\top \mathbf{A} \mathbf{d}}$$

- This is **exact line search**. There are several methods, such as **Armijo backtracking method**, that try to ensure reduction in the objective function without spending too much time trying to solve this subproblem precisely.