

Lec 10: Unconstrained Optimization I

15-369/769: Numerical Methods (Foundations, ML & Visual Computing)

Instructor: Minchen Li

Table of Content

- Motivation
- Optimality
- One-Dimensional Strategies

Table of Content

- Motivation
- Optimality
- One-Dimensional Strategies

Motivation

Variational Approach to Numerical Algorithms

- Strategy: formulate problems as minimizing (or maximizing) an *objective function* (energy) $E(\mathbf{x})$, possibly with constraints.
- Unknown variable $\mathbf{x}$ may be a vector or reshaped into a matrix.

Problem	§	Objective	Constraints
Least-squares	4.1.2	$E(\mathbf{x}) = \ A\mathbf{x} - \mathbf{b}\ _2^2$	None
Project $\mathbf{b}$ onto $\mathbf{a}$	5.4.1	$E(c) = \ c\mathbf{a} - \mathbf{b}\ _2^2$	None
Eigenvectors of symmetric A	6.1	$E(\mathbf{x}) = \mathbf{x}^\top A\mathbf{x}$	$\ \mathbf{x}\ _2 = 1$
Pseudoinverse	7.2.1	$E(\mathbf{x}) = \ \mathbf{x}\ _2^2$	$A^\top A\mathbf{x} = A^\top \mathbf{b}$
Principal component analysis	7.2.5	$E(C) = \ X - CC^\top X\ _{\text{Fro}}$	$C^\top C = I_{d \times d}$
Broyden step	8.2.2	$E(J_k) = \ J_k - J_{k-1}\ _{\text{Fro}}^2$	$J_k \cdot \Delta\mathbf{x}_k = \Delta f_k$

Motivation

Unconstrained Optimization Overview

- Focus of this lecture: **unconstrained optimization problems**.
- Goal: minimize or maximize $f : \mathbb{R}^n \rightarrow \mathbb{R}$ without additional constraints.
- The objective function f can be nonlinear, which may require nonlinear optimization methods.
- In practice, such problems are common across statistics, machine learning, and applied mathematics.

Motivation

Example: Nonlinear Least-Squares

- Given pairs (x_i, y_i) with $f(x_i) \approx y_i$.
- Goal: find best-fitting function parameters.
- Suppose $f(x) = ce^{ax}$, with $c, a \in \mathbb{R}$.
- Parameters a, c chosen to minimize error:

$$E(a, c) = \sum_i (y_i - ce^{ax_i})^2$$

- Not quadratic in $a \Rightarrow$ linear least-squares inapplicable.
- Requires nonlinear optimization techniques.

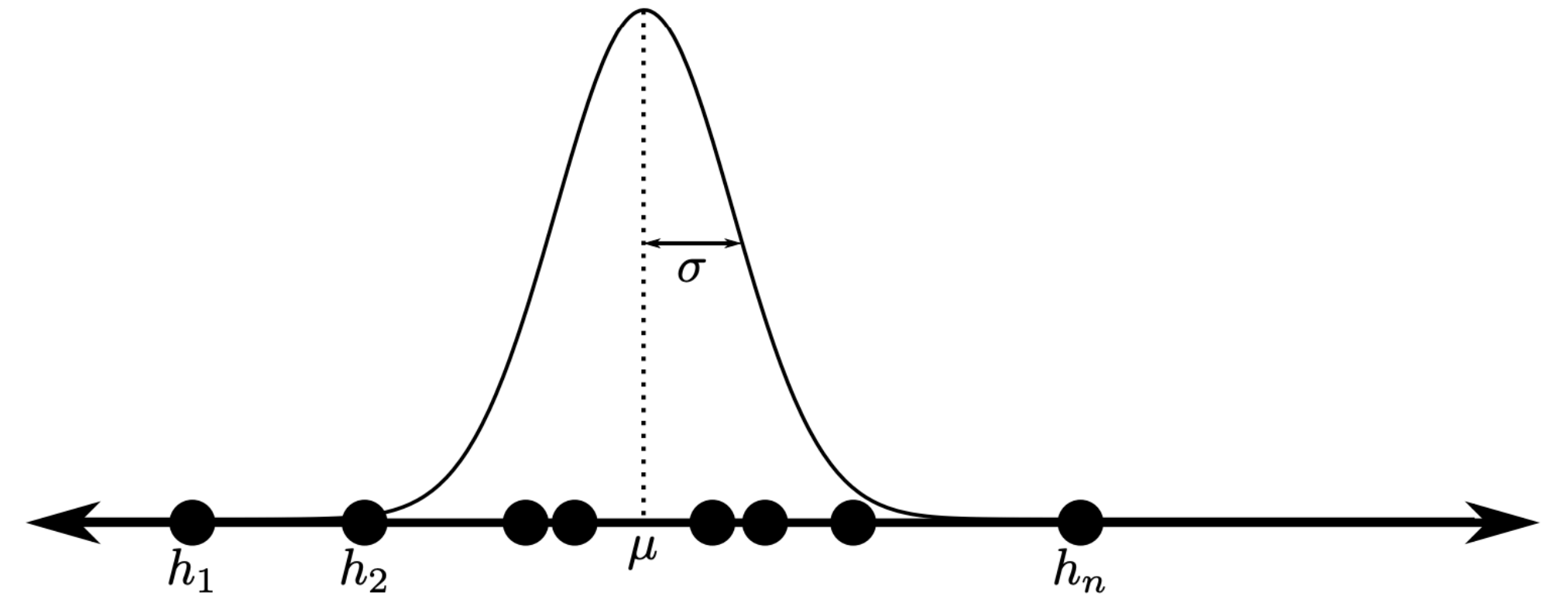
Motivation

Example: Maximum Likelihood Estimation

- Common in machine learning: estimate parameters of a distribution from data.
- Example: student heights $\{h_i\}$ modeled by a *normal distribution*:

$$g(h; \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(h-\mu)^2/2\sigma^2}$$

- Parameters:
 - μ : mean (center of distribution)
 - σ : standard deviation (spread)



- Likelihood of observing heights $\{h_i\}$ under independence assumption:

$$P(\{h_1, \dots, h_n\}; \mu, \sigma) = \prod_i g(h_i; \mu, \sigma).$$

Motivation

Example: Maximum Likelihood Estimation (cont.)

- **Maximum-likelihood estimate:** choose μ, σ to maximize P .
- Equivalent to maximizing the *log-likelihood*:

$$\ell(\mu, \sigma) = \log P(\{h_1, \dots, h_n\}; \mu, \sigma).$$

- Log-likelihood has same maxima but is easier numerically:
 - Transforms products into sums.
 - Improves numerical stability.
 - Simplifies derivatives for optimization.
- Bayesian statistics: add a *prior* on (μ, σ) for Bayesian inference.

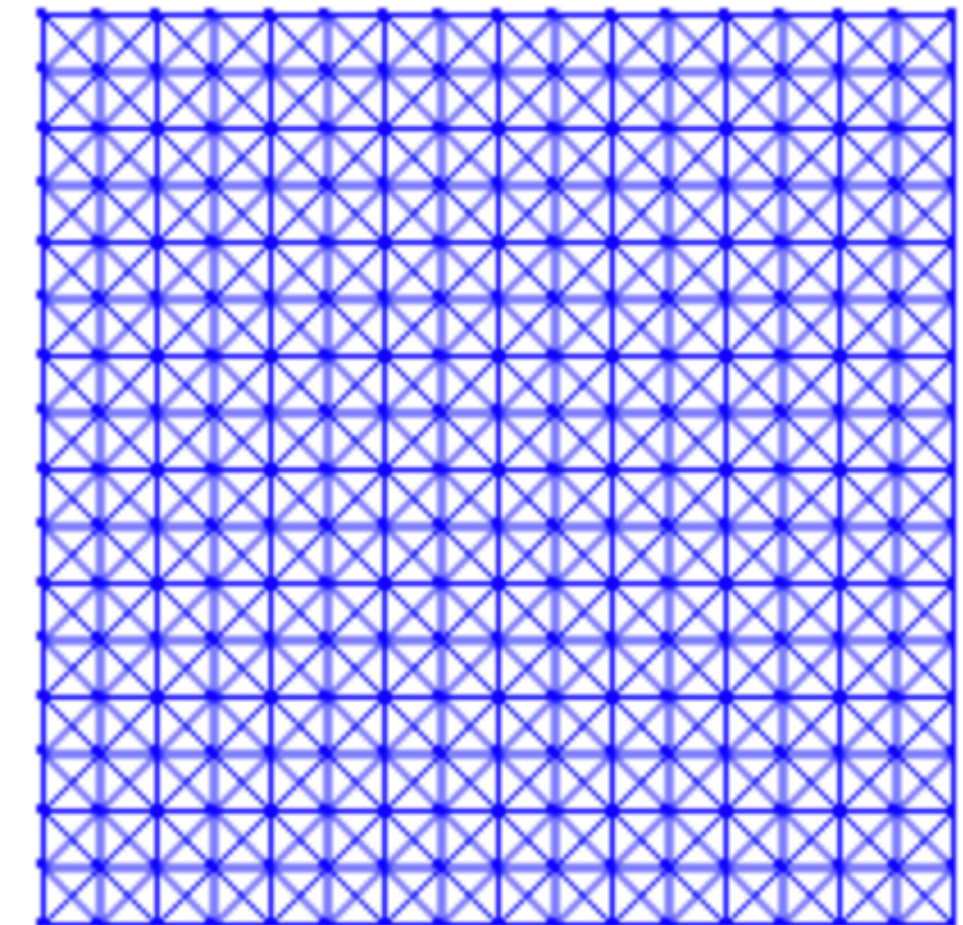
Motivation

Example: Simulating Elastic Objects

- We discretize a square object by uniformly sampling a grid of N points.
- Internal structure is modeled using springs that connect nearby points.
- Each spring e follows a squared version of Hooke's law:

$$P_e(x) = \frac{1}{2} l^2 k \left(\frac{\|\mathbf{x}_1 - \mathbf{x}_2\|^2}{l^2} - 1 \right)^2$$

- $\mathbf{x}_1, \mathbf{x}_2$: endpoints of spring e .
- l : rest length, k : stiffness parameter.
- $x \in \mathbb{R}^{2N}$: stacked vector of all point positions.
- Using squared lengths avoids square roots $\Rightarrow$ polynomial gradient and Hessian (efficient).



Motivation

Example: Simulating Elastic Objects (cont.)

- The total spring energy is the sum over all springs: $P(x) = \sum_e P_e(x)$.
- Boundary conditions: top-left and top-right corners fixed with stiff “anchor” springs:

$$A_i(x) = \frac{1}{2} k_A \|\mathbf{x}_i - \bar{\mathbf{x}}_i\|^2, \quad A_j(x) = \frac{1}{2} k_A \|\mathbf{x}_j - \bar{\mathbf{x}}_j\|^2$$

Here k_A is a large stiffness parameter.

- The static equilibrium satisfies force balance among: internal spring forces, anchor spring forces, and gravitational forces:

$$-\nabla P(x) - \nabla A_i(x) - \nabla A_j(x) + Mg = 0,$$

where $M \in \mathbb{R}^{2N \times 2N}$: diagonal mass matrix, and $g \in \mathbb{R}^{2N}$: gravitational acceleration.

- In variational form, the solution is also a stationary point of the potential energy

$$P(x) + A_i(x) + A_j(x) - x^T Mg$$

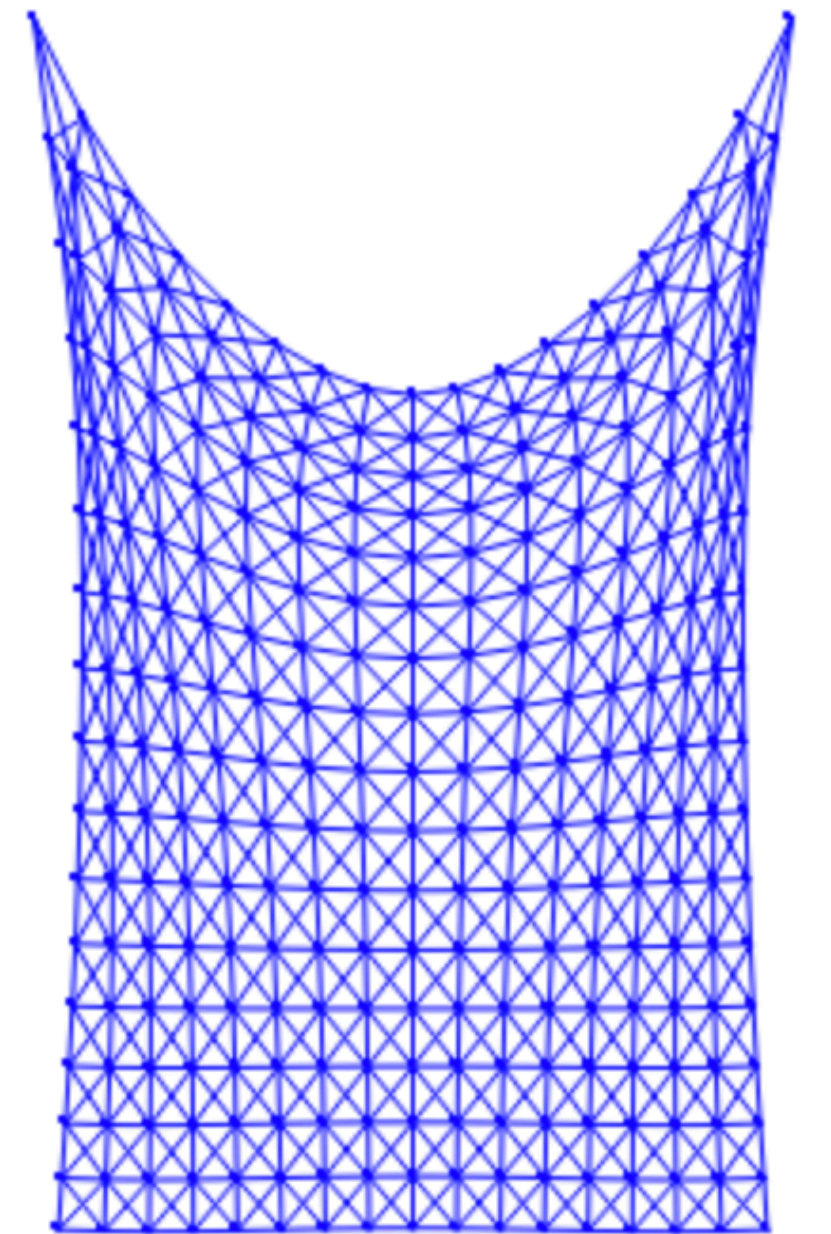


Table of Content

- Motivation
- Optimality
- One-Dimensional Strategies

Optimality

Global vs. Local Minima

- Given $f : \mathbb{R}^n \rightarrow \mathbb{R}$, we study minima of $f(\mathbf{x})$.

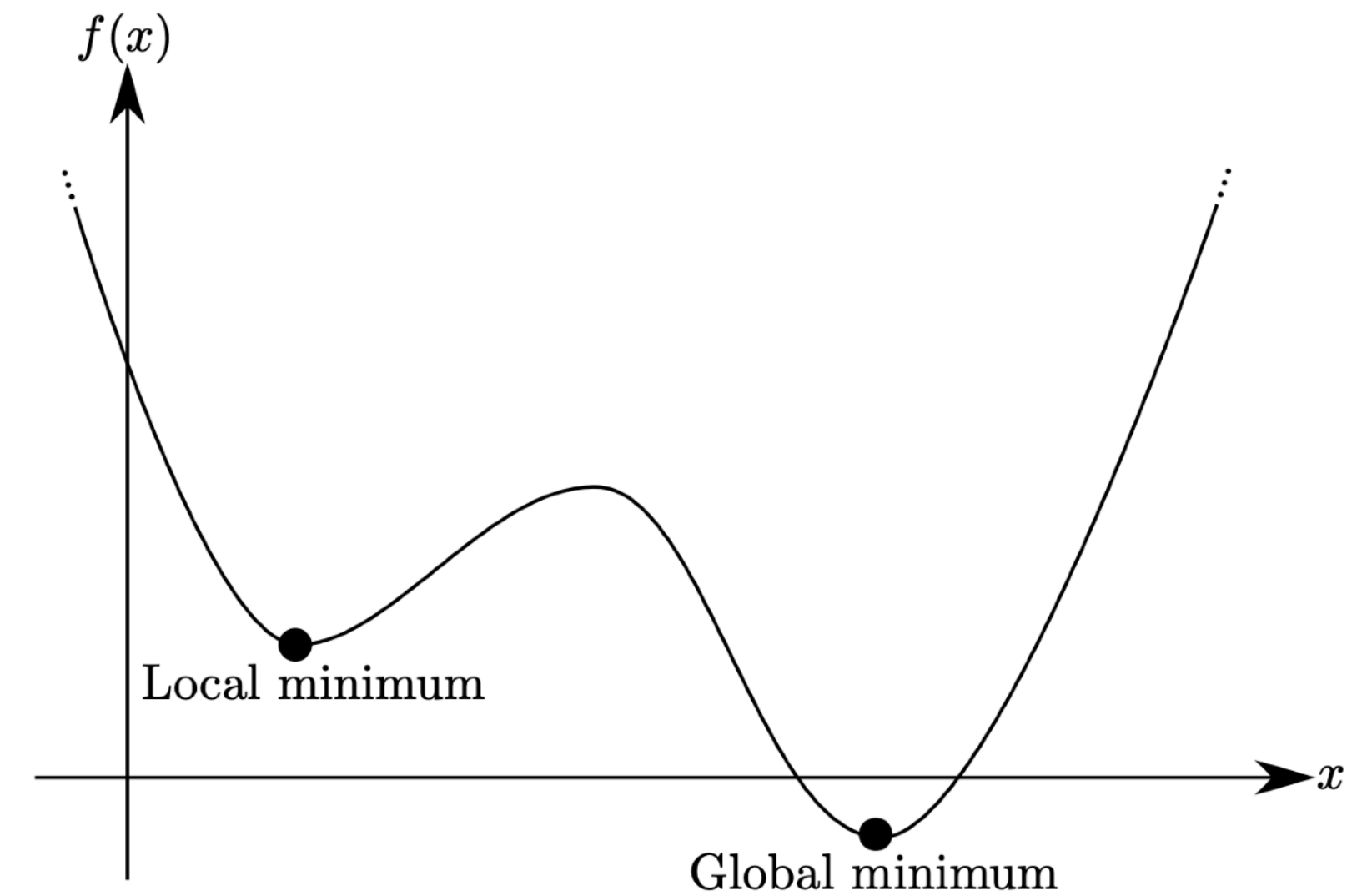
- **Global minimum** (Def. 9.1): $\mathbf{x}^*$ is global if

$$f(\mathbf{x}^*) \leq f(\mathbf{x}) \quad \forall \mathbf{x} \in \mathbb{R}^n.$$

- **Local minimum** (Def. 9.2): $\mathbf{x}^*$ is local if $\exists \varepsilon > 0$ such that

$$f(\mathbf{x}^*) \leq f(\mathbf{x}) \quad \forall \mathbf{x} \text{ with } \|\mathbf{x} - \mathbf{x}^*\|_2 < \varepsilon.$$

- Global search can be intractable; many algorithms aim for local minima while using heuristics to explore the landscape.



Optimality

Differential Optimality: First-Order Condition

- Assume f is differentiable. Linearize near $\mathbf{x}_0$:

$$f(\mathbf{x}) \approx f(\mathbf{x}_0) + \nabla f(\mathbf{x}_0)^\top (\mathbf{x} - \mathbf{x}_0).$$

- Cauchy–Schwarz: $\nabla f(\mathbf{x}_0)^\top (\mathbf{x} - \mathbf{x}_0) \leq \|\nabla f(\mathbf{x}_0)\|_2 \|\mathbf{x} - \mathbf{x}_0\|_2$, with equality when $\mathbf{x} - \mathbf{x}_0 = \alpha \nabla f(\mathbf{x}_0)$, $\alpha \geq 0$.
- Along $\mathbf{x} = \mathbf{x}_0 + \alpha \nabla f(\mathbf{x}_0)$:

$$f(\mathbf{x}_0 + \alpha \nabla f(\mathbf{x}_0)) \approx f(\mathbf{x}_0) + \alpha \|\nabla f(\mathbf{x}_0)\|_2^2.$$

- Hence, if $\mathbf{x}_0$ were a local minimum and $\|\nabla f(\mathbf{x}_0)\|_2 > 0$, moving along $-\nabla f(\mathbf{x}_0)$ would further decrease f (contradiction).
- **Necessary condition:** any local minimizer $\mathbf{x}^*$ satisfies $\nabla f(\mathbf{x}^*) = \mathbf{0}$.

Optimality

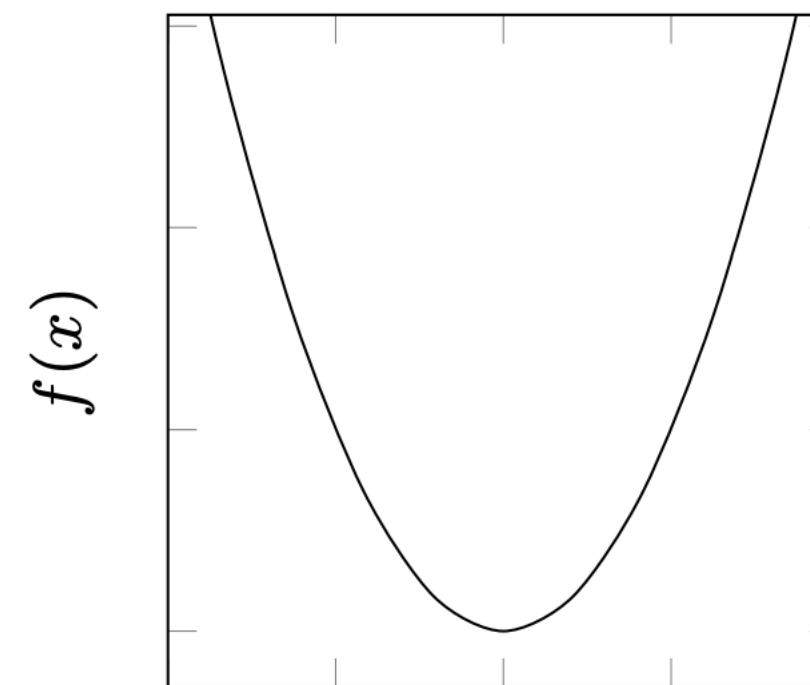
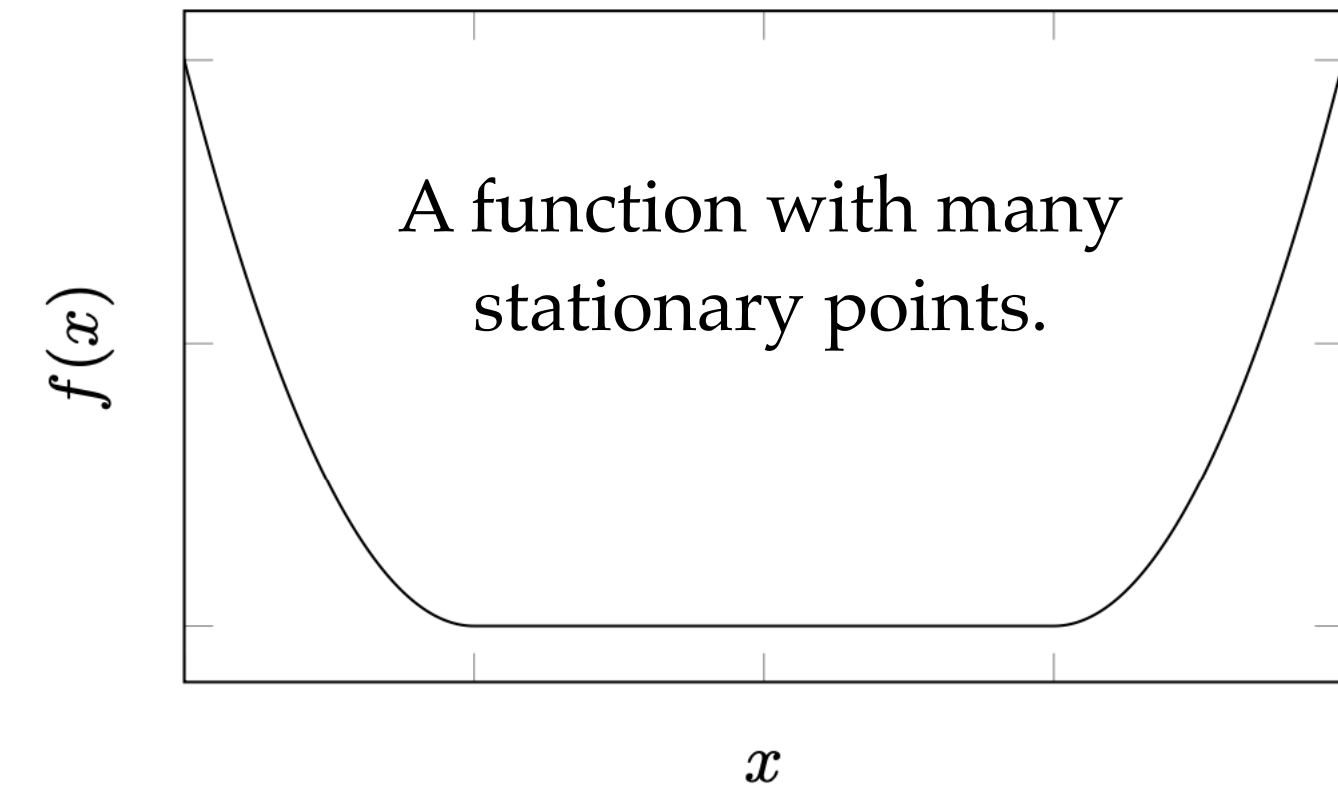
Stationary Points and Second-Order Test

- **Stationary point** (Def. 9.3): $\nabla f(\mathbf{x}) = \mathbf{0}$.
- If $f \in C^2$, Hessian $H_f(\mathbf{x}) = \left[\frac{\partial^2 f}{\partial x_i \partial x_j} \right]_{i,j}$ refines the test via quadratic model:

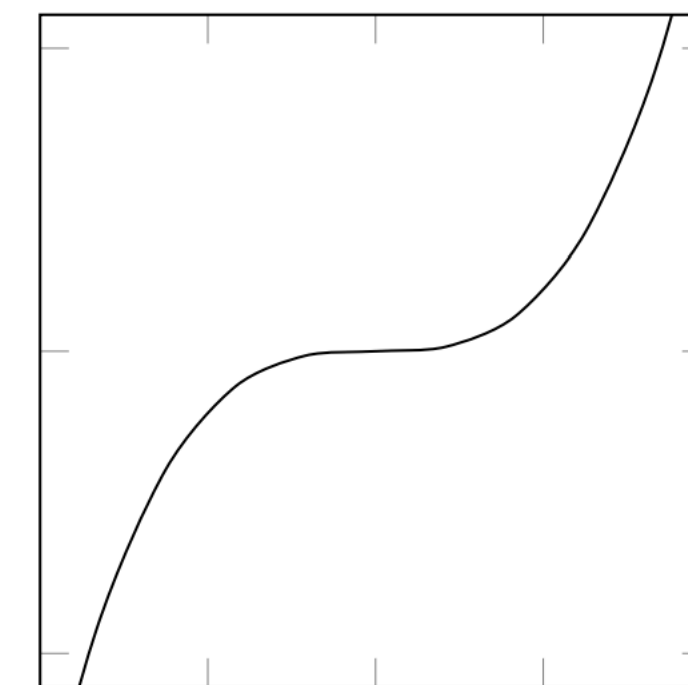
- At a stationary point $\mathbf{x}^*$: $f(\mathbf{x}) \approx f(\mathbf{x}^*) + \frac{1}{2}(\mathbf{x} - \mathbf{x}^*)^\top H_f(\mathbf{x}^*)(\mathbf{x} - \mathbf{x}^*)$.

- **Classification:**

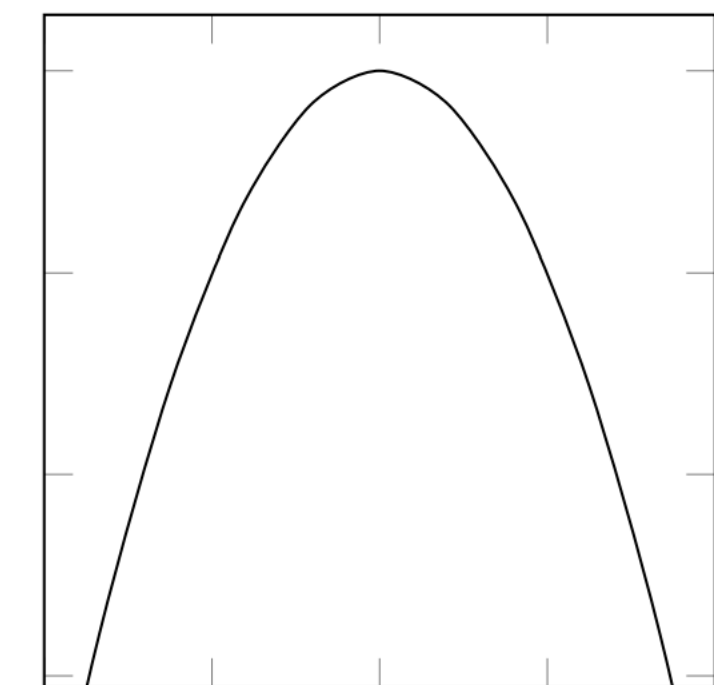
- $H_f(\mathbf{x}^*) \succ 0$: local minimum.
- $H_f(\mathbf{x}^*) \prec 0$: local maximum.
- $H_f(\mathbf{x}^*)$ indefinite: saddle.
- $H_f(\mathbf{x}^*)$ noninvertible (semidefinite): degenerate behaviors possible.



x (local minimum)



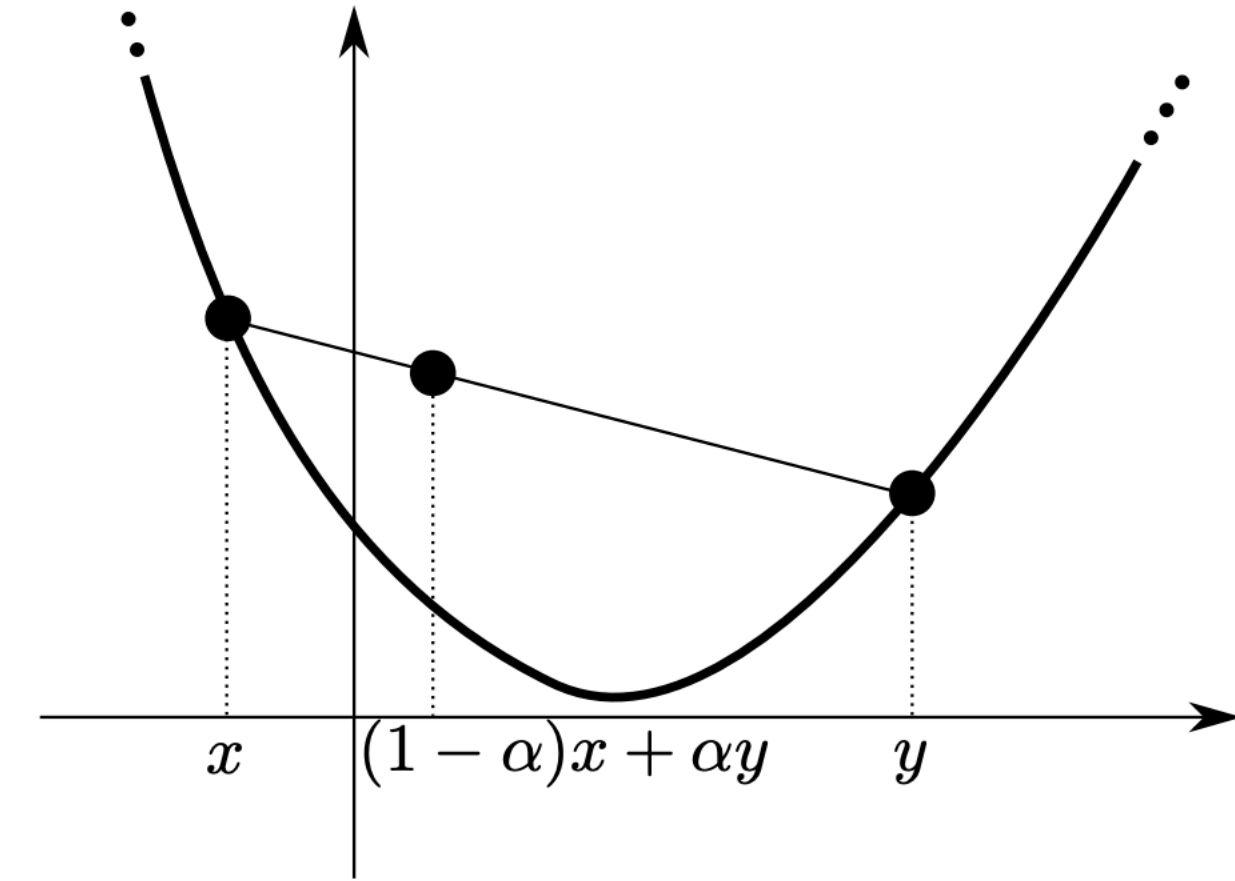
x (saddle point)



x (local maximum)

Optimality

Convexity and Global Optimality



- **Convex function** (Def. 9.4): $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is convex if

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

- **Key fact (Prop. 9.1):** every local minimum of a convex f is a *global* minimum.

- Sketch proof:

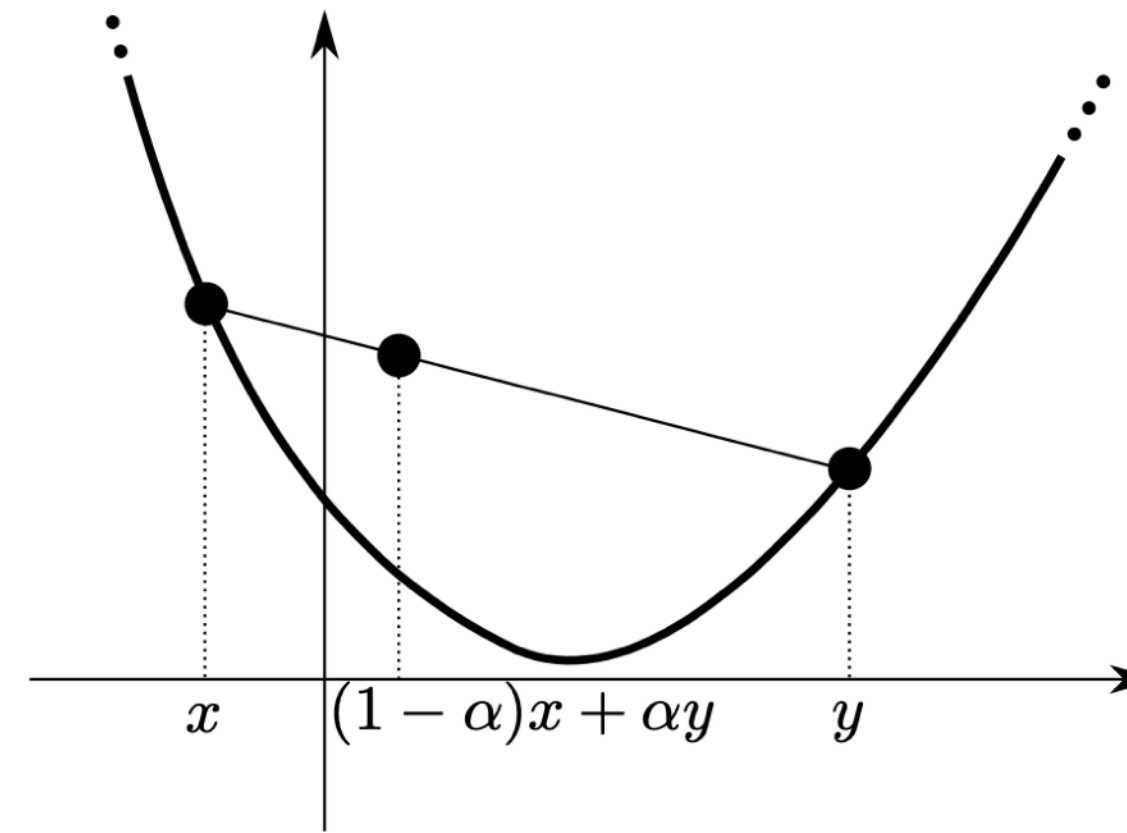
If $\mathbf{x}$ is local min and $\exists \mathbf{x}^* : f(\mathbf{x}^*) < f(\mathbf{x})$,

then for small $\alpha \in (0, 1)$, $f(\mathbf{x}) \leq f(\mathbf{x} + \alpha(\mathbf{x}^* - \mathbf{x})) \leq (1 - \alpha)f(\mathbf{x}) + \alpha f(\mathbf{x}^*) < f(\mathbf{x})$ — contradiction.

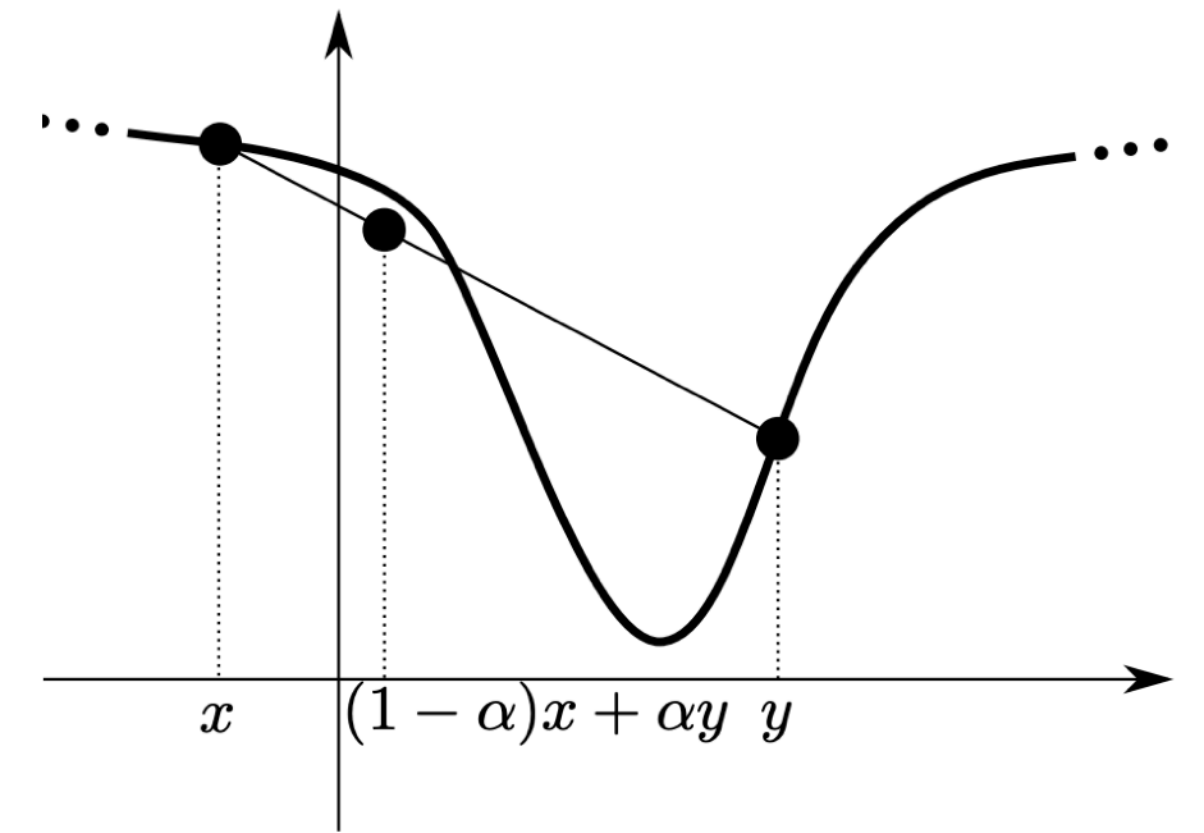
- Sufficient test (when $f \in C^2$): $H_f(\mathbf{x}) \succcurlyeq 0$ for all $\mathbf{x}$ (and $\succ 0$ for strict convexity).

Optimality

Quasi-Convexity



(a) Convex



(b) Quasiconvex

- **Quasi-convex** (a relaxation of convexity):

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

- Although the “bowl” shape may be absent, *local minimizers are global minimizers* for quasi-convex functions as well.
- Useful when f is not differentiable yet has favorable level-set geometry.

Table of Content

- Motivation
- Optimality
- One-Dimensional Strategies

One-Dimensional Strategies

Newton's Method

- Begin with optimization for functions $f : \mathbb{R} \rightarrow \mathbb{R}$ (single variable).
- Then expand to general functions $f : \mathbb{R}^n \rightarrow \mathbb{R}$.
- Goal: Find stationary points $\mathbf{x}^*$ such that $\nabla f(\mathbf{x}^*) = 0$.
- Assume we can post-process to classify maxima, minima, saddle points.
- Focus: solving for stationary points.

One-Dimensional Strategies

Newton's Method (cont.)

$$f(x) \approx f(x_k) + f'(x_k)(x - x_k) + \frac{1}{2}f''(x_k)(x - x_k)^2$$

- Approximation is parabolic, with vertex at $x_k - f'(x_k) / f''(x_k)$.
- Iterative update:

$$x_{k+1} = x_k - \frac{f'(x_k)}{f''(x_k)}.$$

- Quadratic convergence if x_0 is close to $\mathbf{x}^*$.
- Closely tied to Newton's method for root-finding of $f'(x)$.

One-Dimensional Strategies

Beyond Newton: Secant and Parabolic Interpolation

- Secant method can eliminate need for f'' , but requires f' .
- Successive Parabolic Interpolation:
 - Uses parabolas through last three iterates.
 - Relies on $f(x_k)$, $f(x_{k-1})$, $f(x_{k-2})$.
 - Can converge superlinearly.

One-Dimensional Strategies

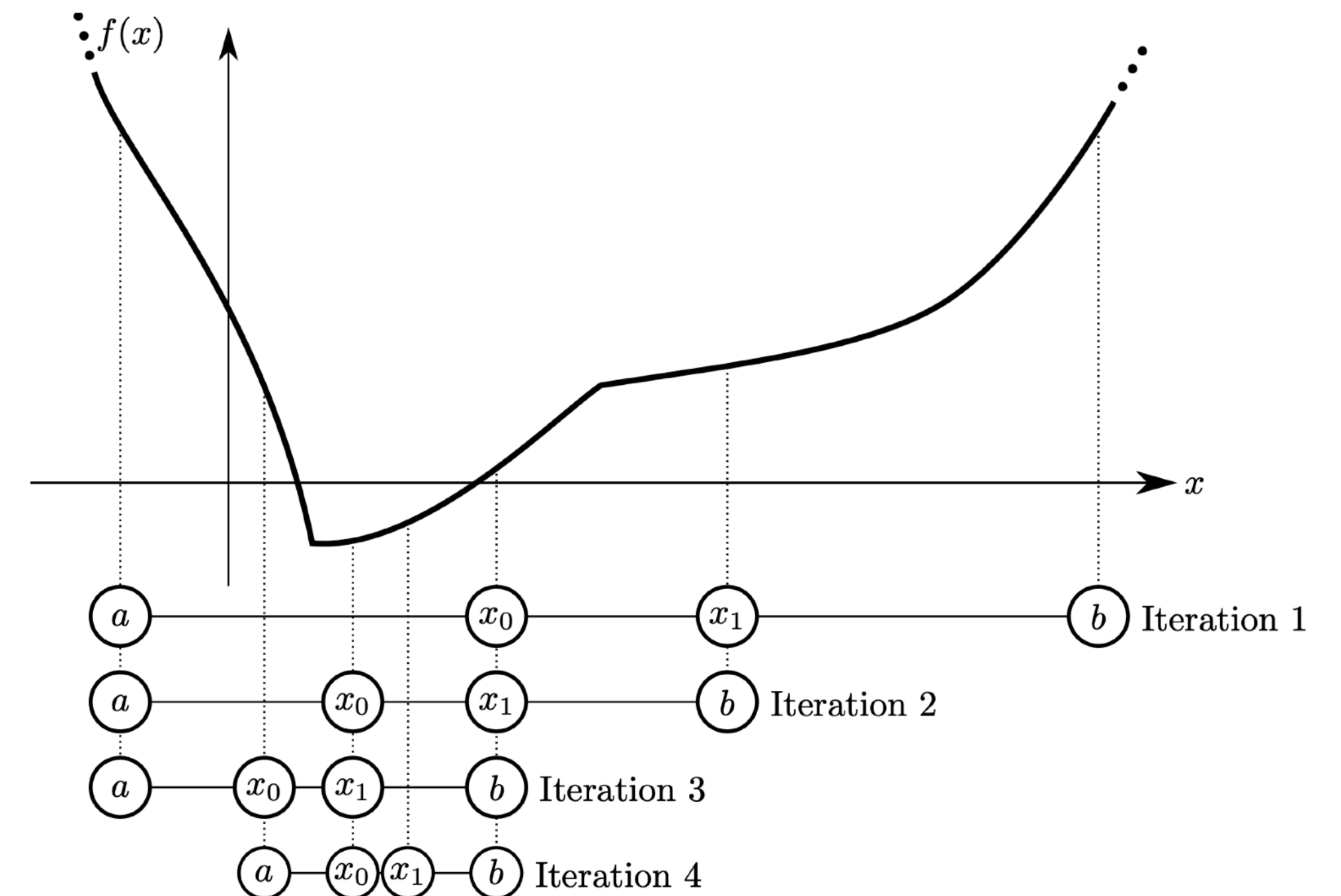
Golden Section Search

- Adapts bisection for optimization.
- Requires weaker assumption: **unimodularity**.
- **Definition 9.5 (Unimodular)** $f : [a, b] \rightarrow \mathbb{R}$ is unimodular if $\exists x^* \in [a, b]$ such that:
 - f is decreasing on $[a, x^*]$,
 - f is increasing on $[x^*, b]$.
- Ensures a single minimum exists (no local minima).

One-Dimensional Strategies

Golden Section Search (cont.)

- Begin with interval $[a, b]$, remove pieces iteratively.
- In each iteration, pick 2 points $x_0 < x_1$.
- Rules:
 - If $f(x_0) \geq f(x_1)$, discard $[a, x_0]$.
 - If $f(x_1) \geq f(x_0)$, discard $[x_1, b]$.
- Strategy reduces interval by about $1/3$ each step.



One-Dimensional Strategies

Golden Ratio Construction

- Make symmetric choice $x_0 = \alpha$, $x_1 = 1 - \alpha$.
- To reuse function evaluations:

$$(1 - \alpha)^2 = \alpha \quad \Rightarrow \quad \alpha = \frac{1}{2}(3 - \sqrt{5}), \quad 1 - \alpha = \frac{1}{2}(\sqrt{5} - 1).$$

- $1 - \alpha = \tau$: the golden ratio.
- Each iteration discards fraction α of the interval.

One-Dimensional Strategies

Golden Section Search Algorithm

```
function GOLDEN-SECTION-SEARCH( $f(x), a, b$ )  
   $\tau \leftarrow \frac{1}{2}(\sqrt{5} - 1)$   
   $x_0 \leftarrow a + (1 - \tau)(b - a)$            ▷ Initial division of interval  $a < x_0 < x_1 < b$   
   $x_1 \leftarrow a + \tau(b - a)$   
   $f_0 \leftarrow f(x_0)$                        ▷ Function values at  $x_0$  and  $x_1$   
   $f_1 \leftarrow f(x_1)$   
  for  $k \leftarrow 1, 2, 3, \dots$   
    if  $|b - a| < \varepsilon$  then                 ▷ Golden section search converged  
      return  $x^* = \frac{1}{2}(a + b)$   
    else if  $f_0 \geq f_1$  then                 ▷ Remove the interval  $[a, x_0]$   
       $a \leftarrow x_0$                        ▷ Move left side  
       $x_0 \leftarrow x_1$                      ▷ Reuse previous iteration  
       $f_0 \leftarrow f_1$   
       $x_1 \leftarrow a + \tau(b - a)$            ▷ Generate new sample  
       $f_1 \leftarrow f(x_1)$   
    else if  $f_1 > f_0$  then                 ▷ Remove the interval  $[x_1, b]$   
       $b \leftarrow x_1$                        ▷ Move right side  
       $x_1 \leftarrow x_0$                      ▷ Reuse previous iteration  
       $f_1 \leftarrow f_0$   
       $x_0 \leftarrow a + (1 - \tau)(b - a)$      ▷ Generate new sample  
       $f_0 \leftarrow f(x_0)$ 
```

One-Dimensional Strategies

Golden Section Search Remarks

- Converges unconditionally and linearly.
- Requires only function evaluations (no derivatives).
- Applicable only if f is unimodular on $[a, b]$.
- If f is not globally unimodular, need to bracket a local minimum first.

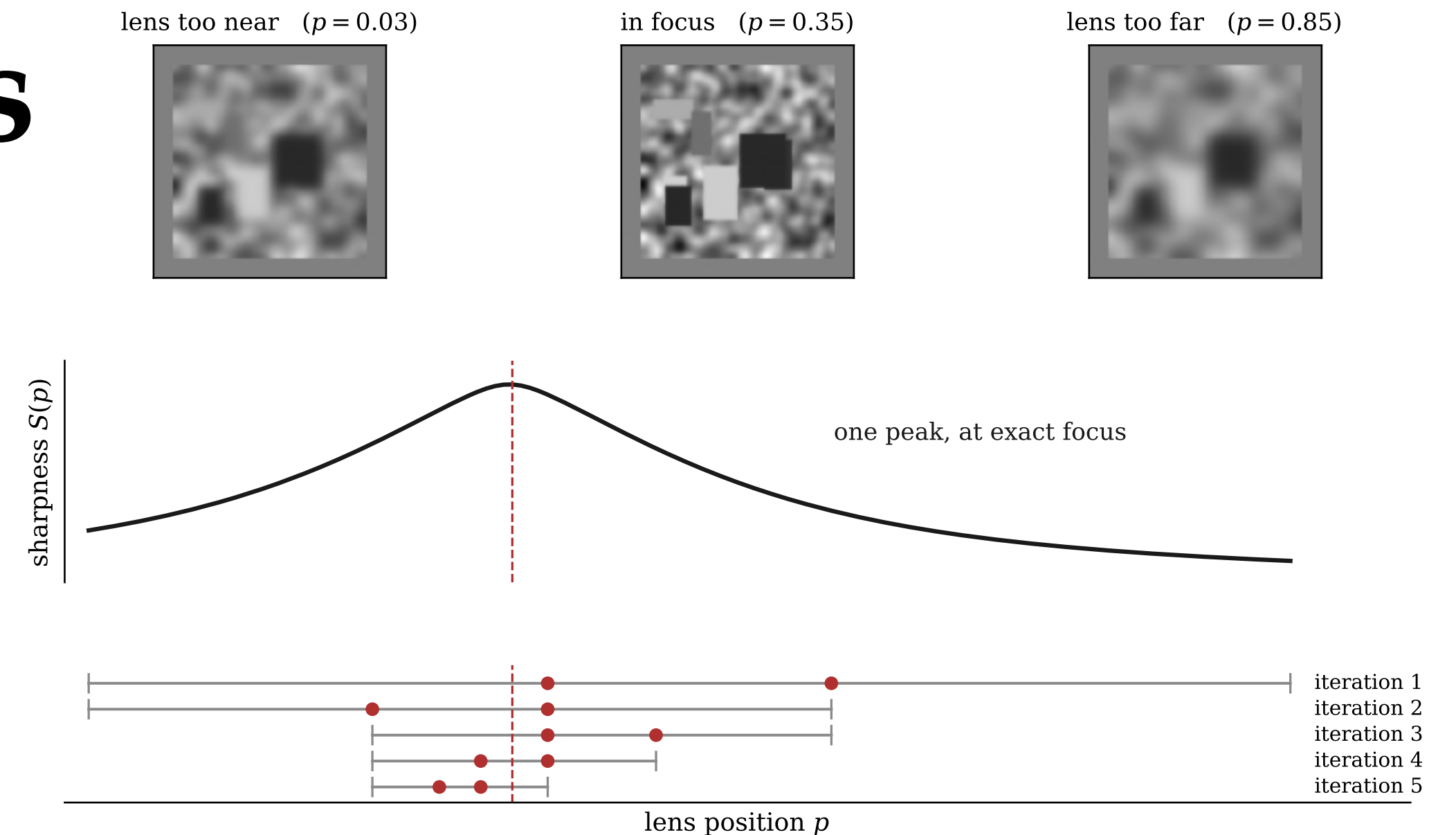
One-Dimensional Strategies

Application: Camera Autofocus

- A camera must place one lens so the picture is sharp. One variable: lens position p .
- Blur destroys fine detail, so measure sharpness by squared image gradients over the focus region:

$$S(p) = \sum_{\text{pixels}} \|\nabla I_p\|^2, \quad \max_p S(p)$$

- No derivative exists: S is only known by physically moving the lens and taking a frame.
- Each evaluation is expensive and visible to the user — so the search must finish in a handful of steps.
- This is exactly golden section's setting: bracketed, derivative-free, unimodal, few evaluations.



One-Dimensional Strategies

When the Unimodal Assumption Fails

- Golden section is guaranteed **only** if S is unimodal on $[a, b]$ — one hill, no others.
- Two subjects at different depths give two peaks. The method still converges, just to the wrong one.
- Real cameras hit this constantly:
 - point light sources produce false "pseudo-hill" peaks;
 - low-contrast scenes flatten the hill until the peak is undetectable.
- This is the hunting you see when your phone racks focus back and forth.
- Standard remedy is the one from your Remarks slide: **bracket a single peak first**, then refine.

two subjects at different depths

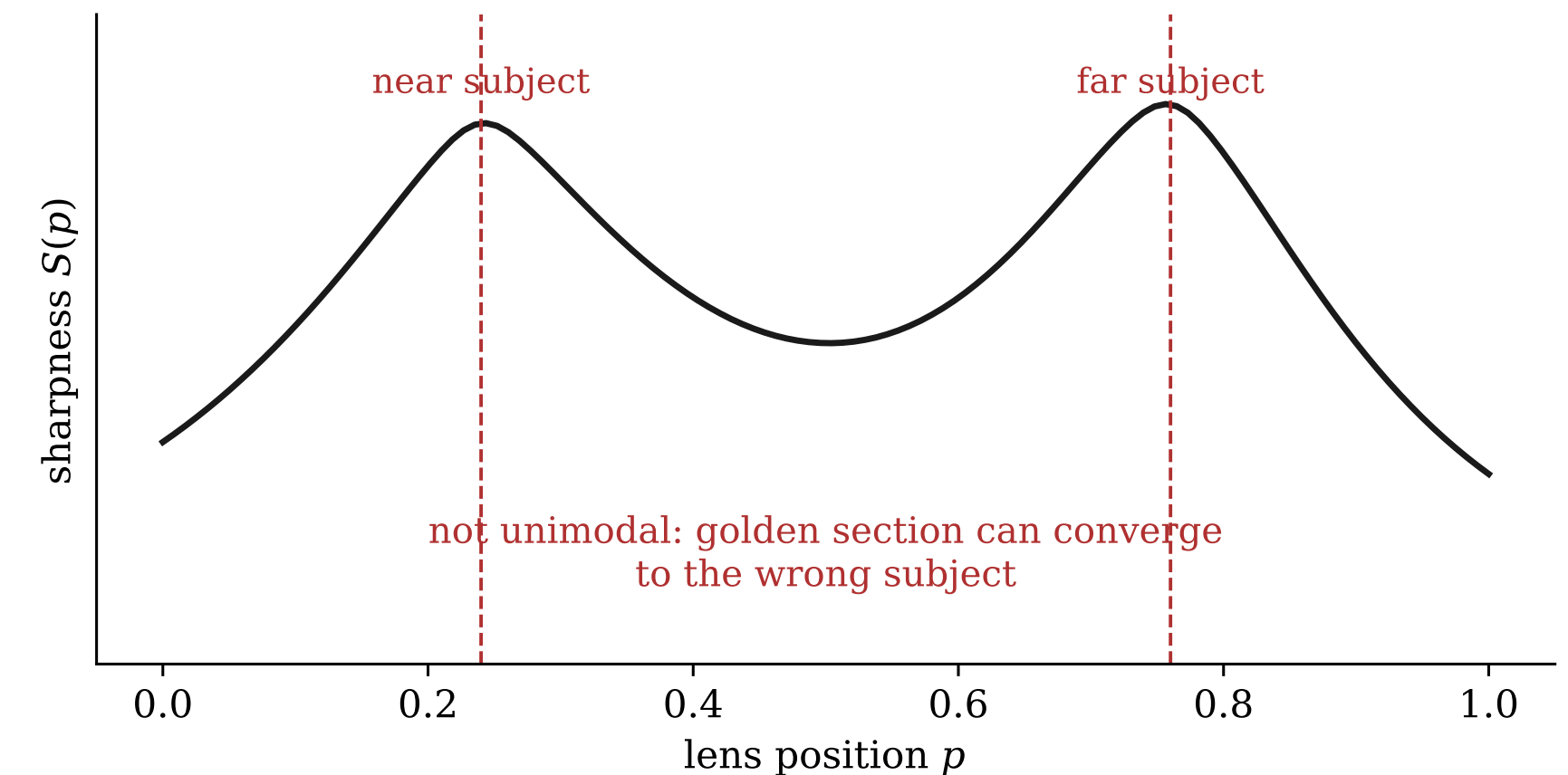
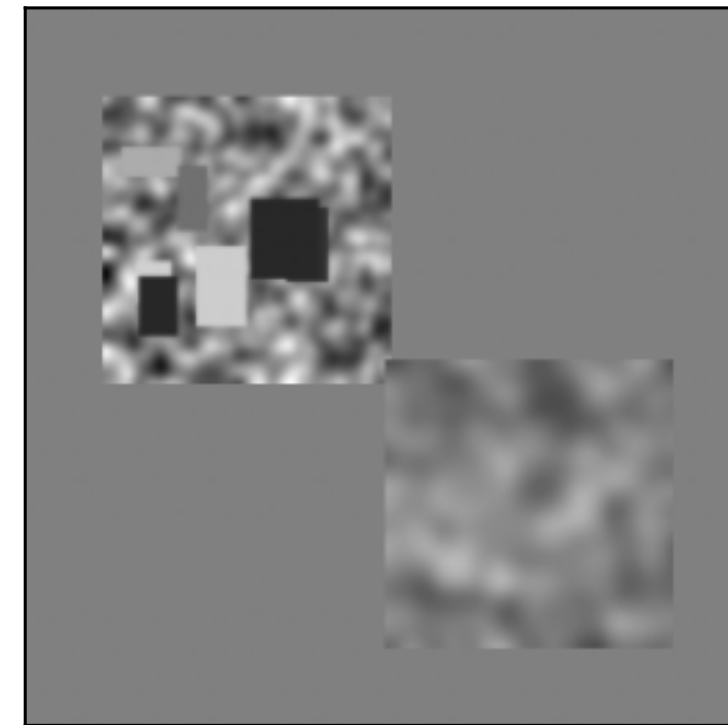


Table of Content

- Motivation
- Optimality
- One-Dimensional Strategies