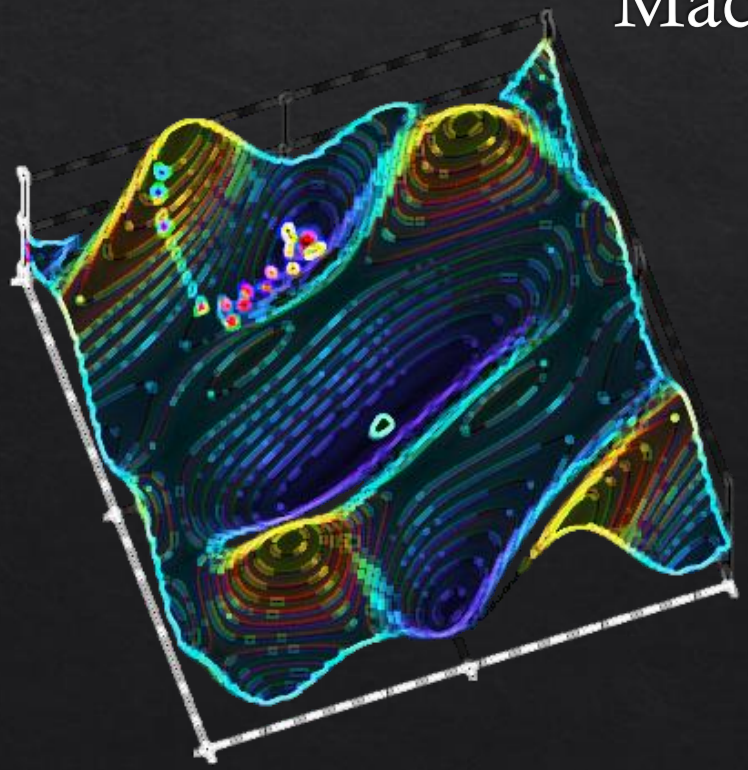
Introduction to Convex Optimization

Machine Learning Methods – Lecture 2



June 2021

fixelalgorithms.gitlab.io

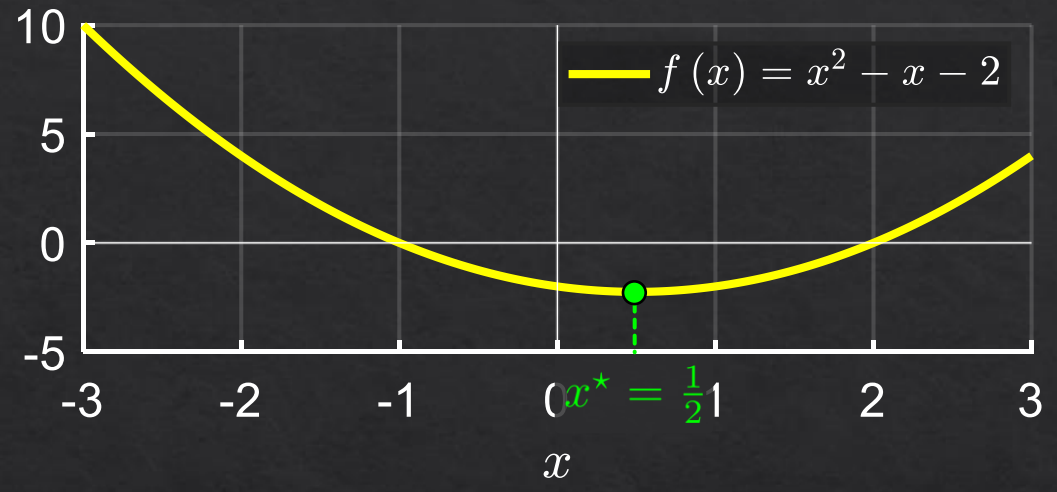
Or Yair

The arg min(·) Operator – 1D

- Consider the quadratic function $f : \mathbb{R} \to \mathbb{R}$:

$$f(x) = ax^2 + bx + c, \qquad a > 0$$

Find: $x^\star = \arg\min_{x \in \mathbb{R}} f(x)$

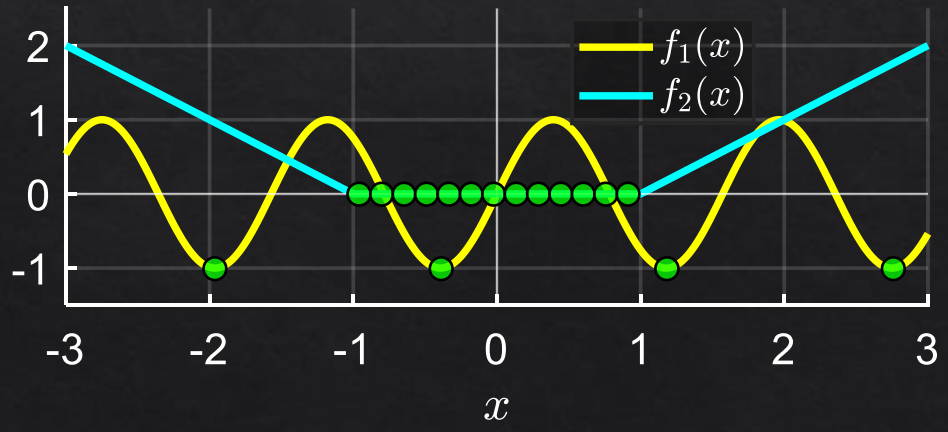


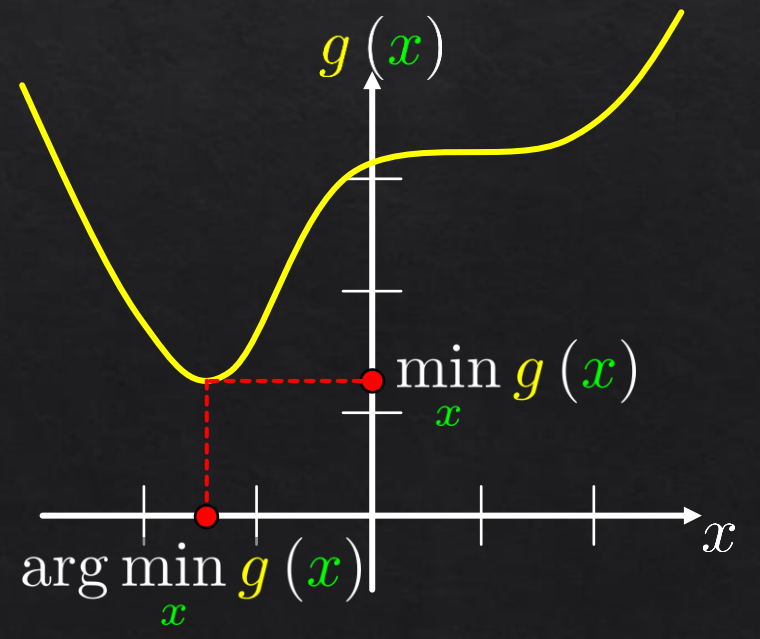
Solution

We can compare the derivative to zero:

$$f'(x^\star) = 0$$

$$2ax^\star + b = 0 \implies \boxed{x^\star = -\frac{b}{2a}}$$





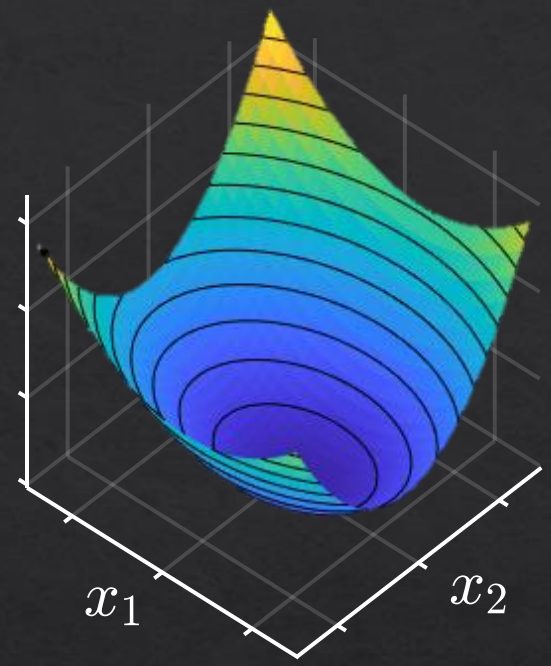
- $\arg\min(\cdot)$ is not always well-define (not unique).

The arg min(·) Operator – 2D

- Consider the quadratic function $f : \mathbb{R}^2 \to \mathbb{R}$:

$$f(x_1, x_2) = a_1 x_1^2 + a_2 x_2^2 + b_1 x_1 + b_2 x_2 + c, \quad a_1, a_2 > 0$$

Find: $x_1^\star, x_2^\star = \arg\min_{x_1, x_2} f(x_1, x_2)$



Solution:

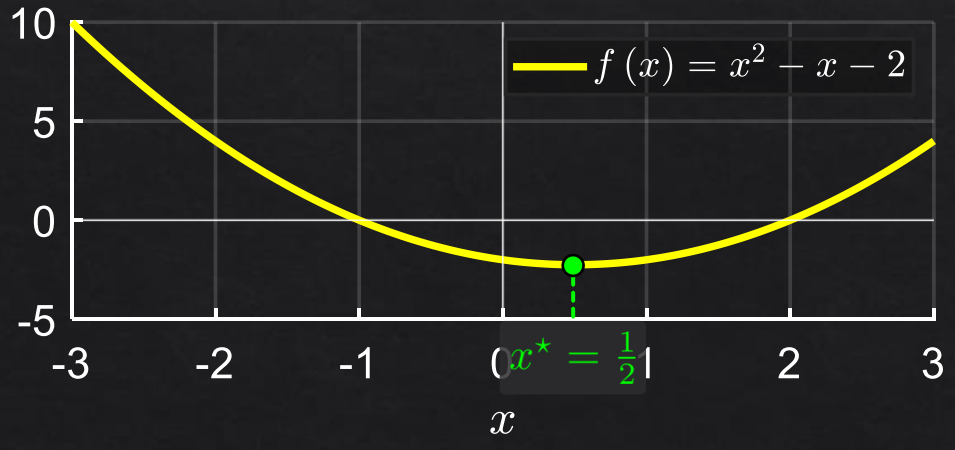
$$\begin{bmatrix} \frac{\partial f(x_1^\star, x_2^\star)}{\partial x_1} \\ \frac{\partial f(x_1^\star, x_2^\star)}{\partial x_2} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

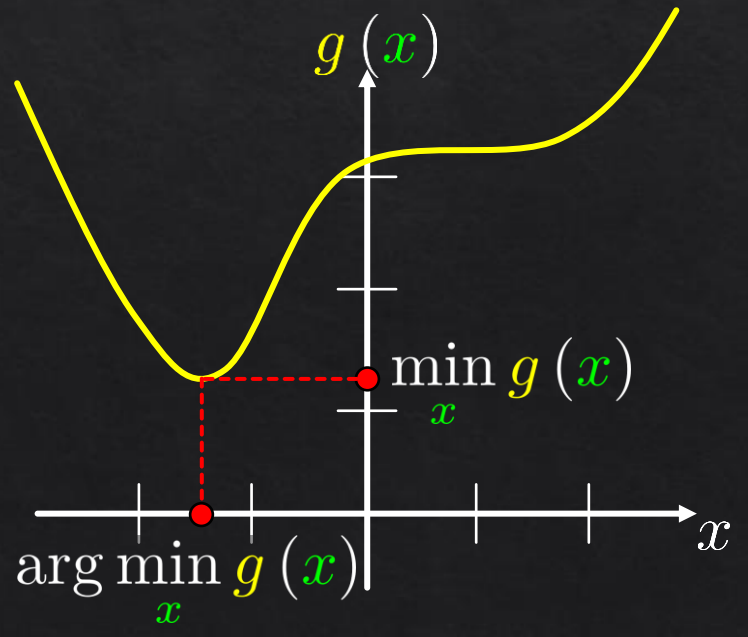
$$\nabla f(x_1, x_2) := \begin{bmatrix} \frac{\partial f(x_1, x_2)}{\partial x_1} \\ \frac{\partial f(x_1, x_2)}{\partial x_2} \end{bmatrix}$$

the gradient of f

$$\begin{bmatrix} 2a_1 x_1^\star + b_1 \\ 2a_2 x_2^\star + b_2 \end{bmatrix} = \mathbf{0}$$

$$\implies \begin{bmatrix} x_1^\star \\ x_2^\star \end{bmatrix} = \begin{bmatrix} -\frac{b_1}{2a_1} \\ -\frac{b_2}{2a_2} \end{bmatrix}$$

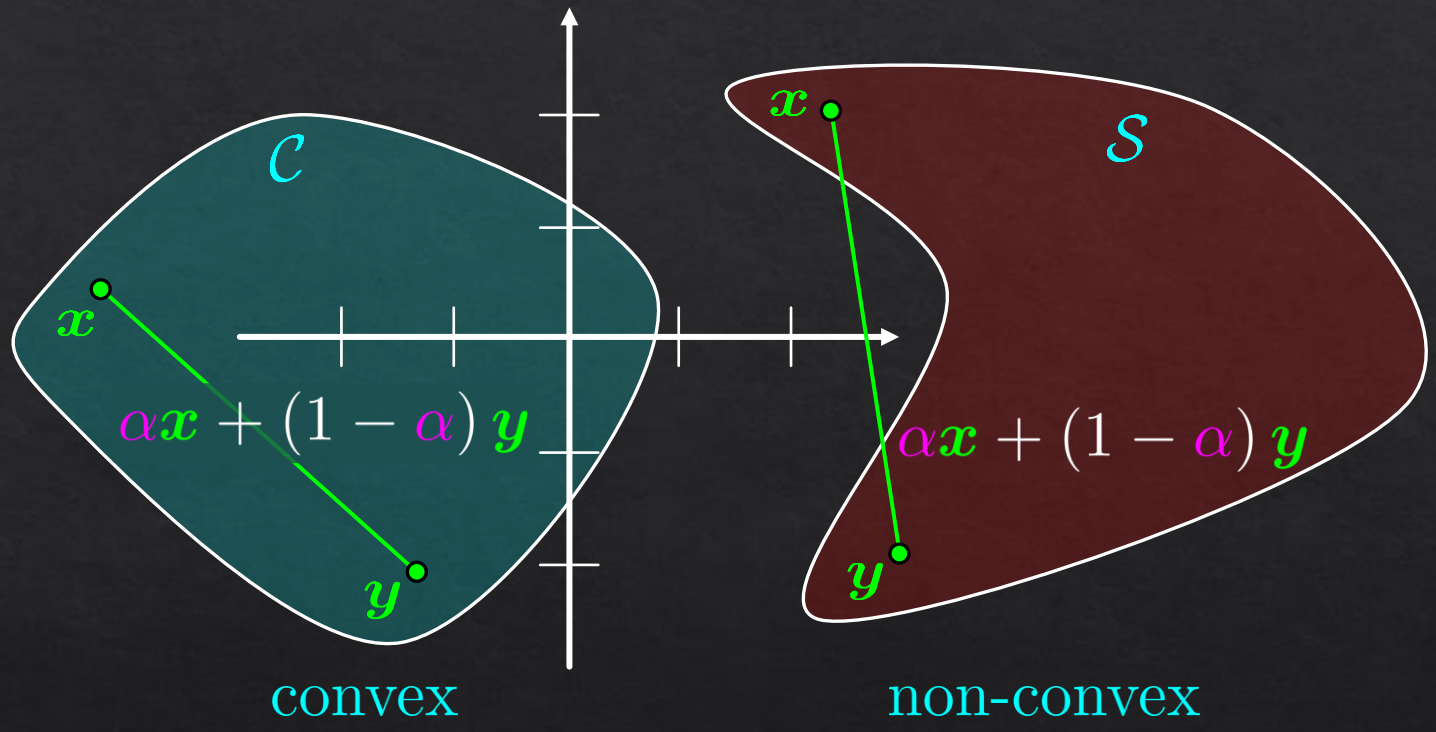




Convex Set

- A set $\mathcal{C} \subseteq \mathbb{R}^d$ is convex if, for all $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{C}$:

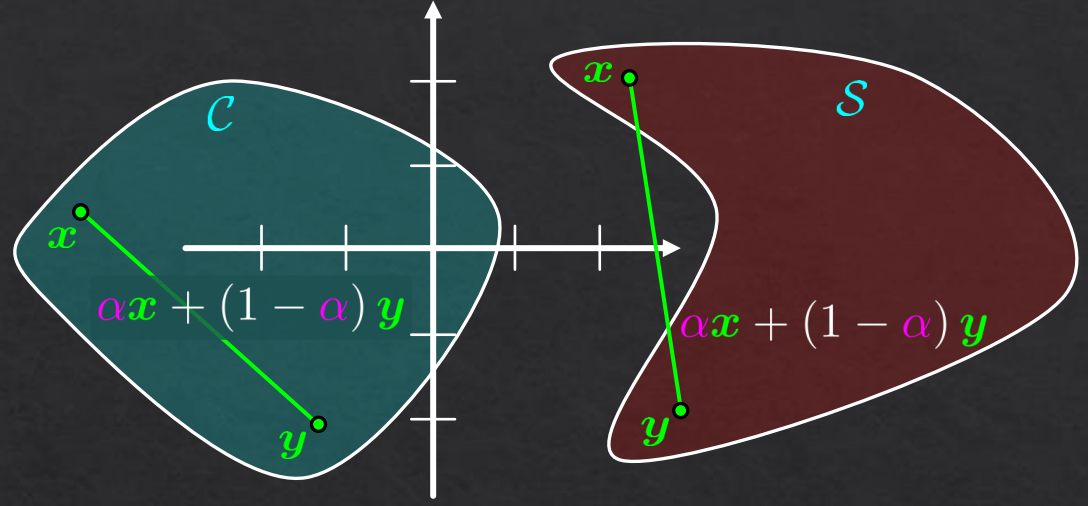
$$\alpha \boldsymbol{x} + (1 - \alpha)\, \boldsymbol{y} \in \mathcal{C}, \quad \forall \alpha \in [0, 1]$$



Convex Set – Properties

- Consider $\mathcal{X} = \{\boldsymbol{x}_i \in \mathcal{C}\}_{i=1}^{N}$ where $\mathcal{C}$ is convex.
- Any convex combination of $\mathcal{X}$ is also in $\mathcal{C}$.

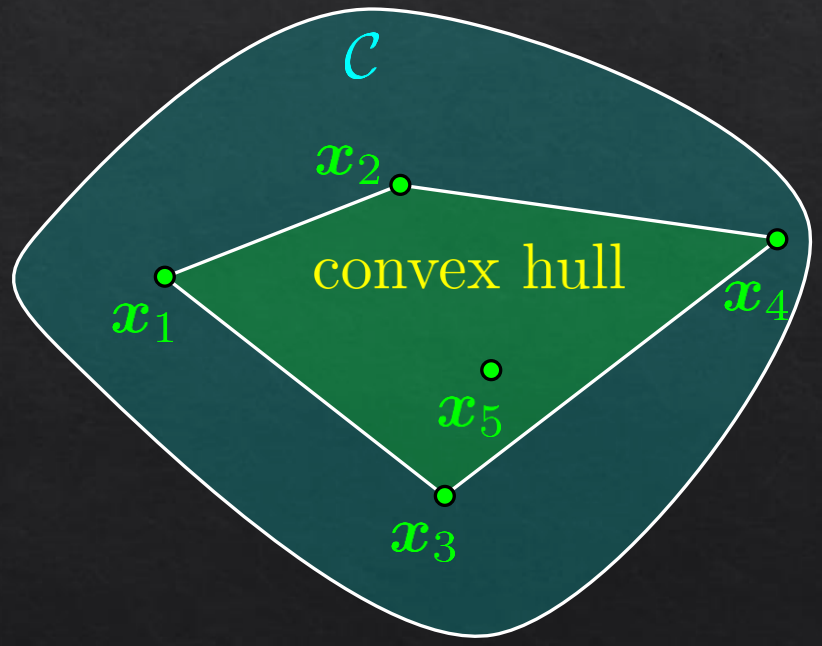
$$\sum_{i=1}^{N} \alpha_i \boldsymbol{x}_i \in \mathcal{C}, \quad \sum_{i=1}^{N} \alpha_i = 1, \quad \alpha_i \geq 0$$

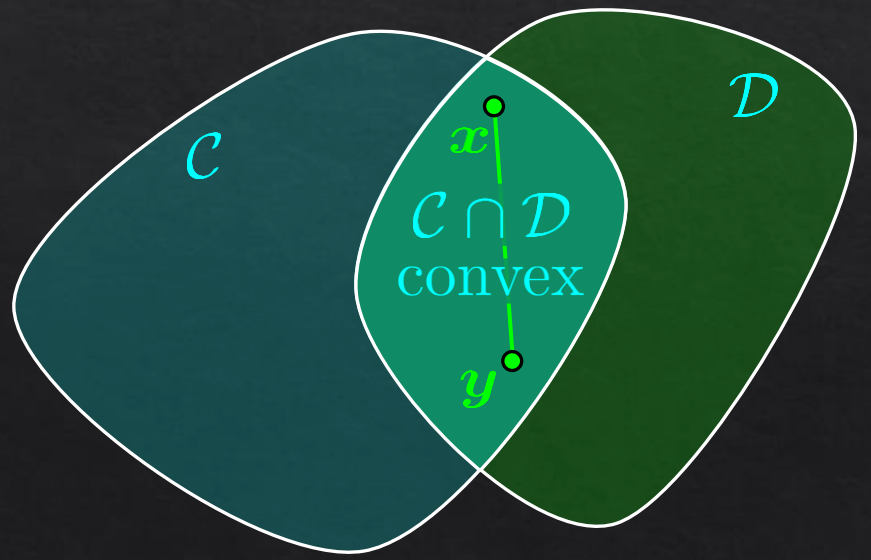


convex

non-convex

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

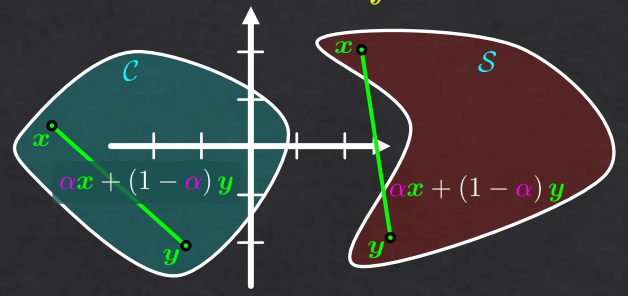




- If $\mathcal{C}$ and $\mathcal{D}$ are convex, so is $\mathcal{C} \cap \mathcal{D}$.

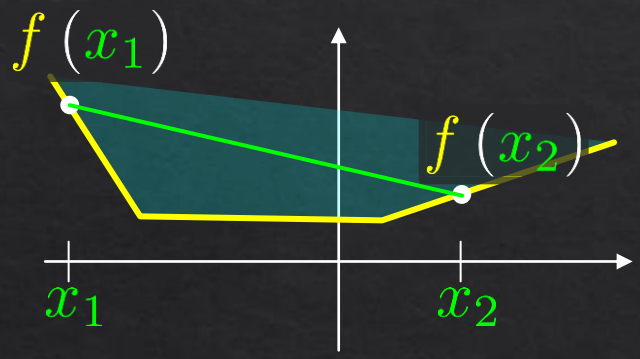
Convex Functions

(∗) f is concave if $-f$ is convex

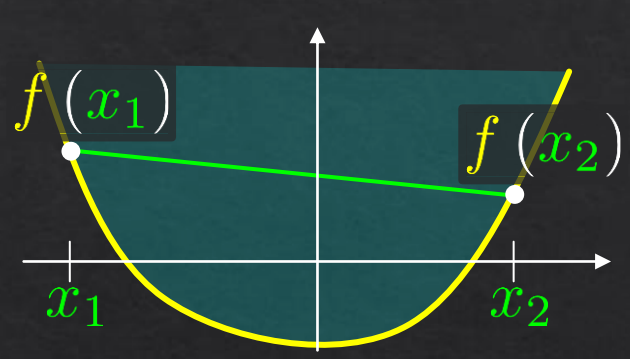


- Consider $f : \mathcal{C} \to \mathbb{R}$ where $\mathcal{C}$ is convex.
- f is called convex if, for all $\boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathcal{C}$:

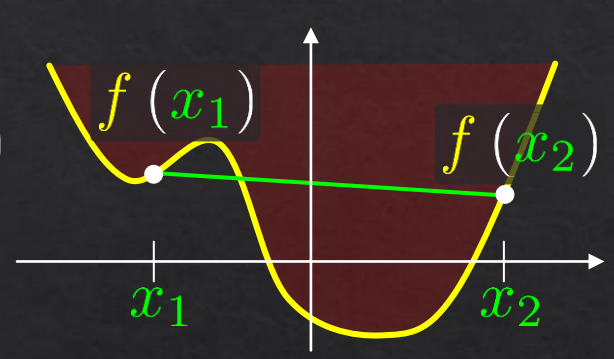
$$f\left(\alpha \boldsymbol{x}_1 + (1-\alpha)\, \boldsymbol{x}_2\right) \leq \alpha f\left(\boldsymbol{x}_1\right) + (1-\alpha)\, f\left(\boldsymbol{x}_2\right), \quad \forall \alpha \in [0,1]$$



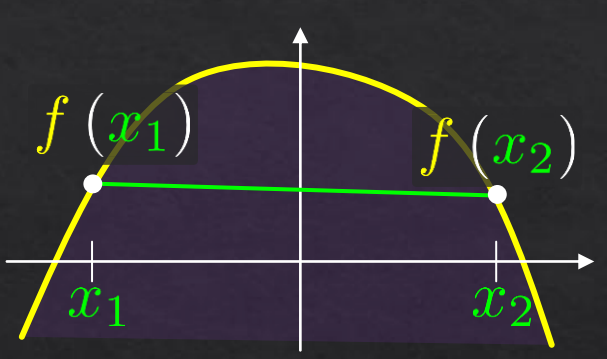
convex



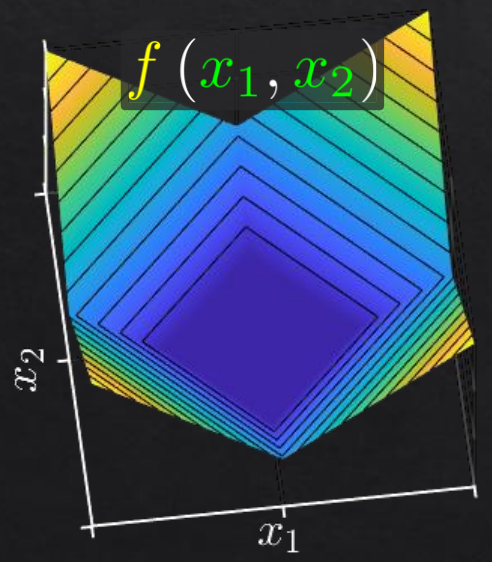
strictly convex

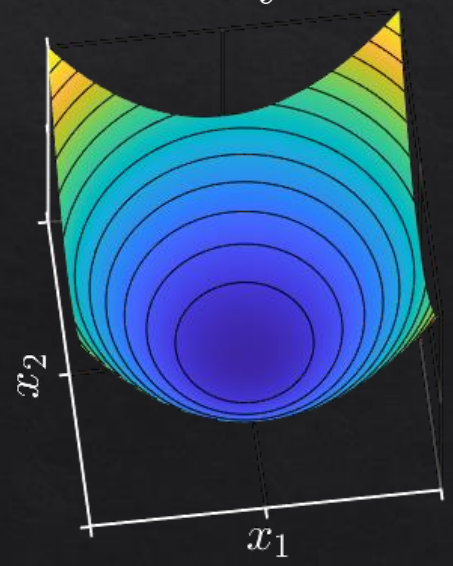


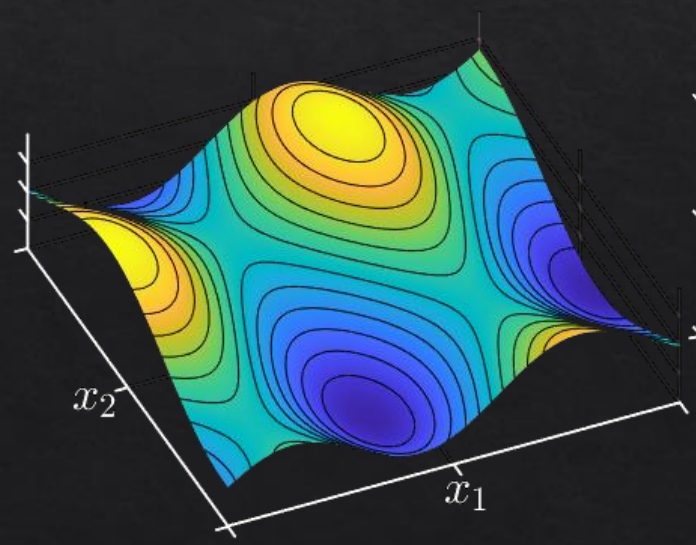
non-convex

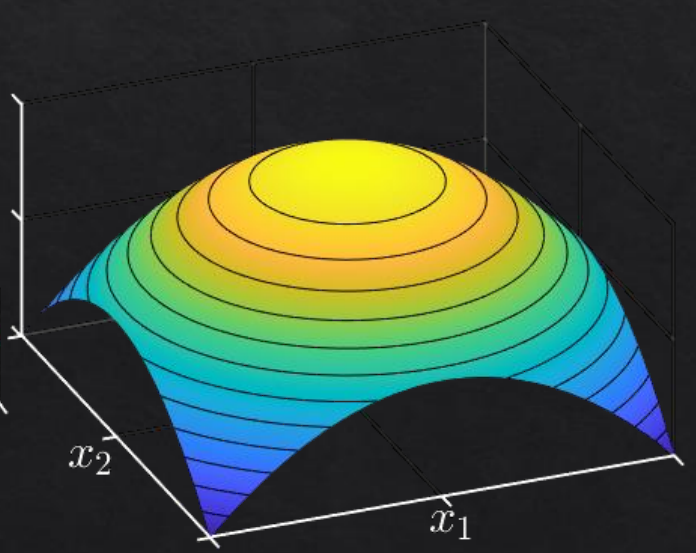


concave*

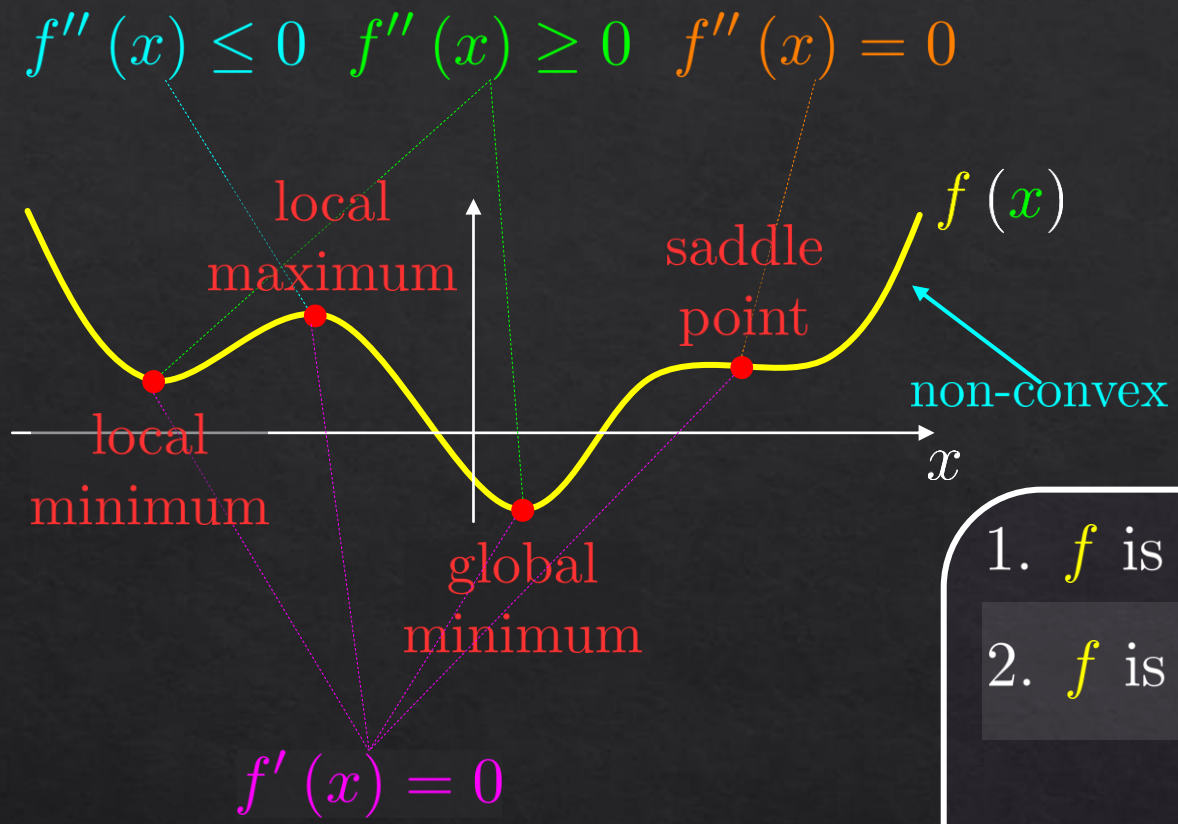


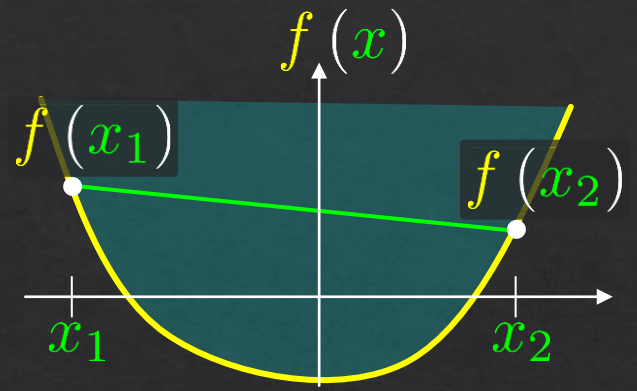






Convex Functions – Properties

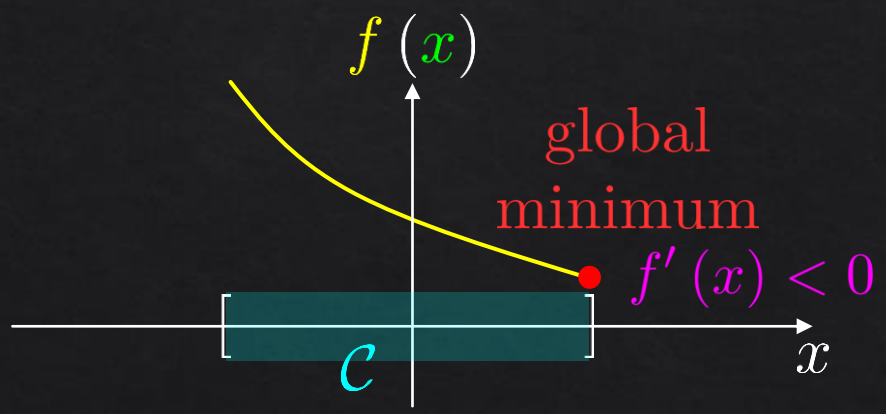




strictly convex

$$f(\alpha \boldsymbol{x}_1 + (1-\alpha)\boldsymbol{x}_2) \leq \alpha f(\boldsymbol{x}_1) + (1-\alpha) f(\boldsymbol{x}_2)$$

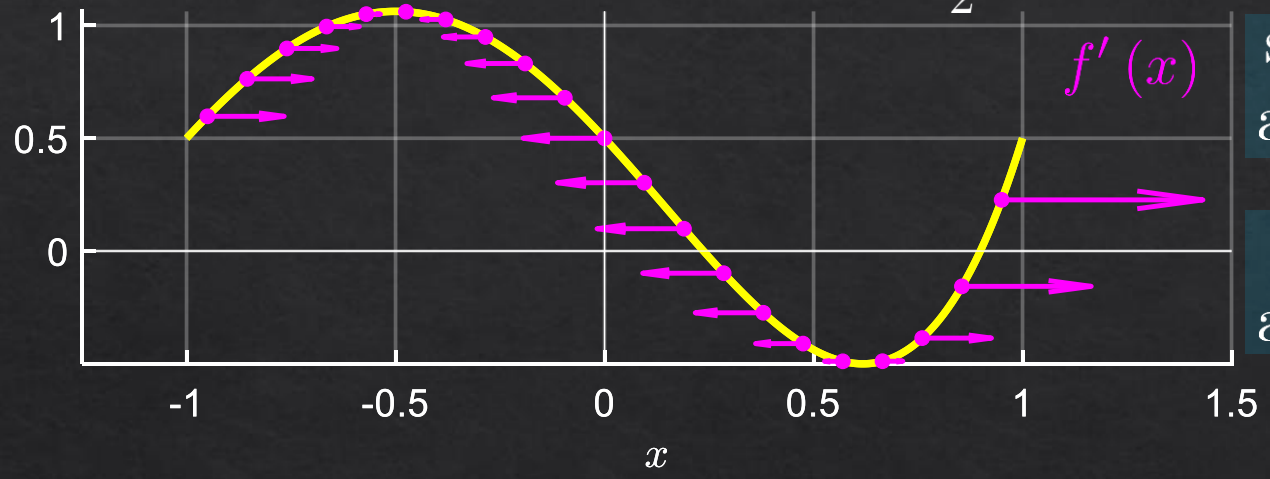
1. f is convex $\iff$ $\forall x : f''(x) \geq 0$ $\quad (\nabla^2 f(\boldsymbol{x}) \succeq 0)$
2. f is convex $\implies$ local minimum $=$ global minimum
3. f is strictly convex $\implies$ the (global) minimum (if exist) is unique.
4. If f is convex, differentiable and $\boldsymbol{x}^\star \notin \partial\mathcal{C}$, then:
   $\boldsymbol{x}^\star$ is a (global) minimum $\iff$ $\nabla f(\boldsymbol{x}^\star) = \mathbf{0}$
   (f has no maxima)



Derivative (1D Gradient)

- Consider the following function $f : \mathbb{R} \to \mathbb{R}$:

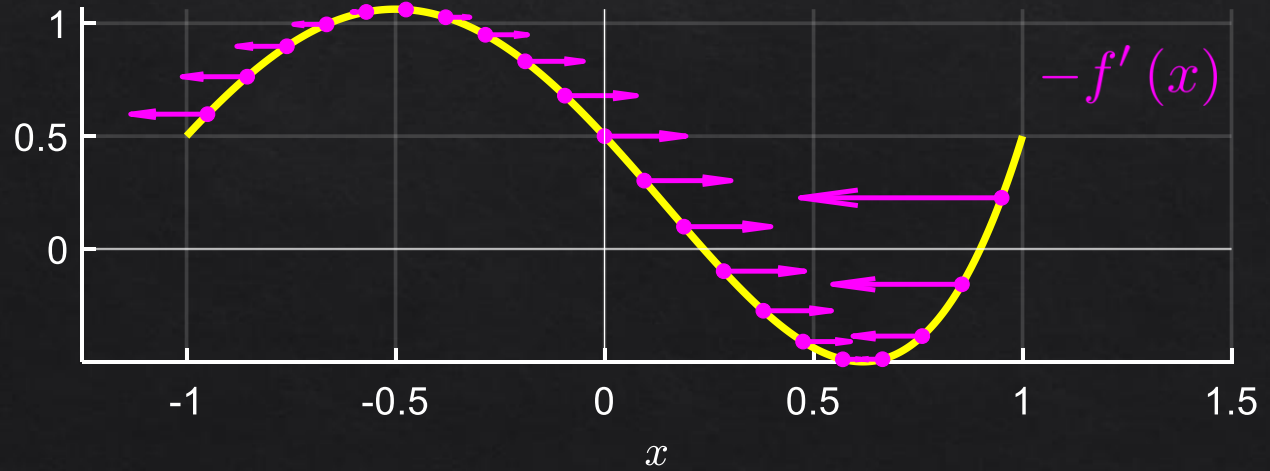
$$f(x) = x^4 + 2x^3 - x^2 - 2x + \frac{1}{2}$$



short arrow $\implies f'(x) \approx 0 \implies$ f is flat (around x)

long arrow $\implies |f'(x)|$ is large $\implies$ steep slope (around x)

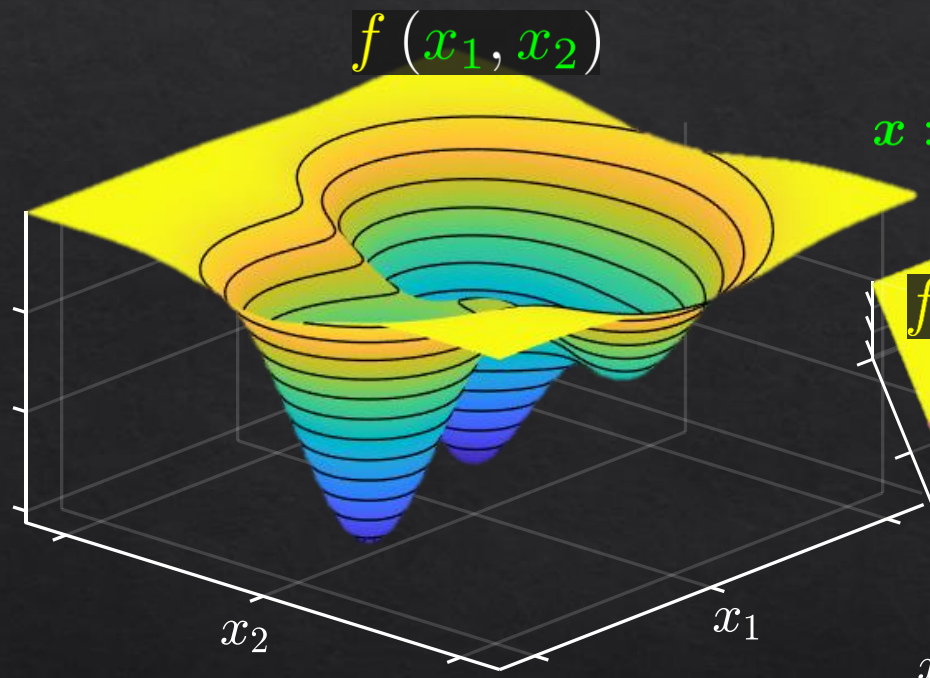
- The arrows indicate the size of the derivative $f'(x)$ (including sign).
- Note that $-f'(x)$ points in a descent direction:



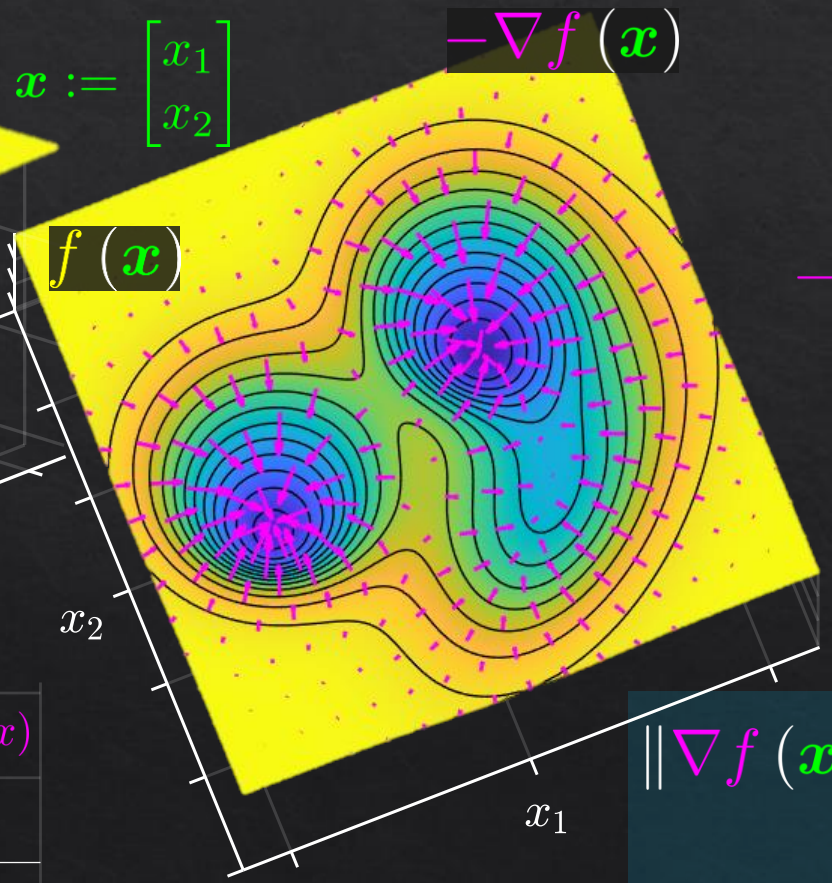
The Gradient

- Let $f : \mathbb{R}^d \to \mathbb{R}$.
- The gradient $\nabla f(\boldsymbol{x})$ is a vector of all partial derivatives:

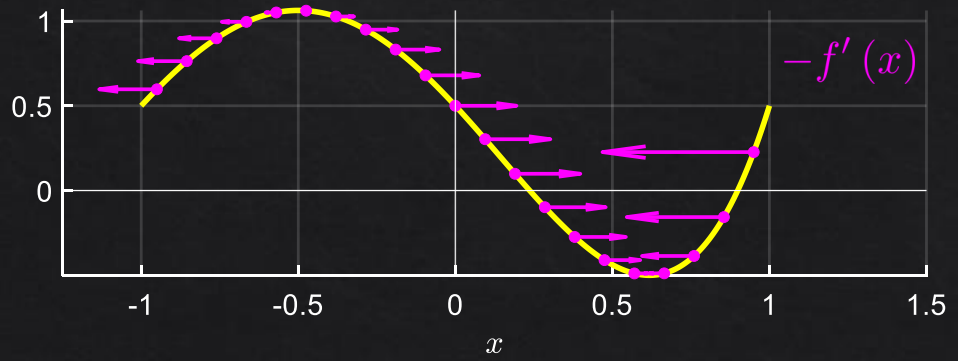
$$\nabla f(\boldsymbol{x}) := \begin{bmatrix} \frac{\partial f(\boldsymbol{x})}{\partial x_1} \\ \frac{\partial f(\boldsymbol{x})}{\partial x_2} \\ \vdots \\ \frac{\partial f(\boldsymbol{x})}{\partial x_d} \end{bmatrix} \in \mathbb{R}^d$$



$$\boldsymbol{x} := \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$



$-\nabla f(\boldsymbol{x})$ points in a descent direction



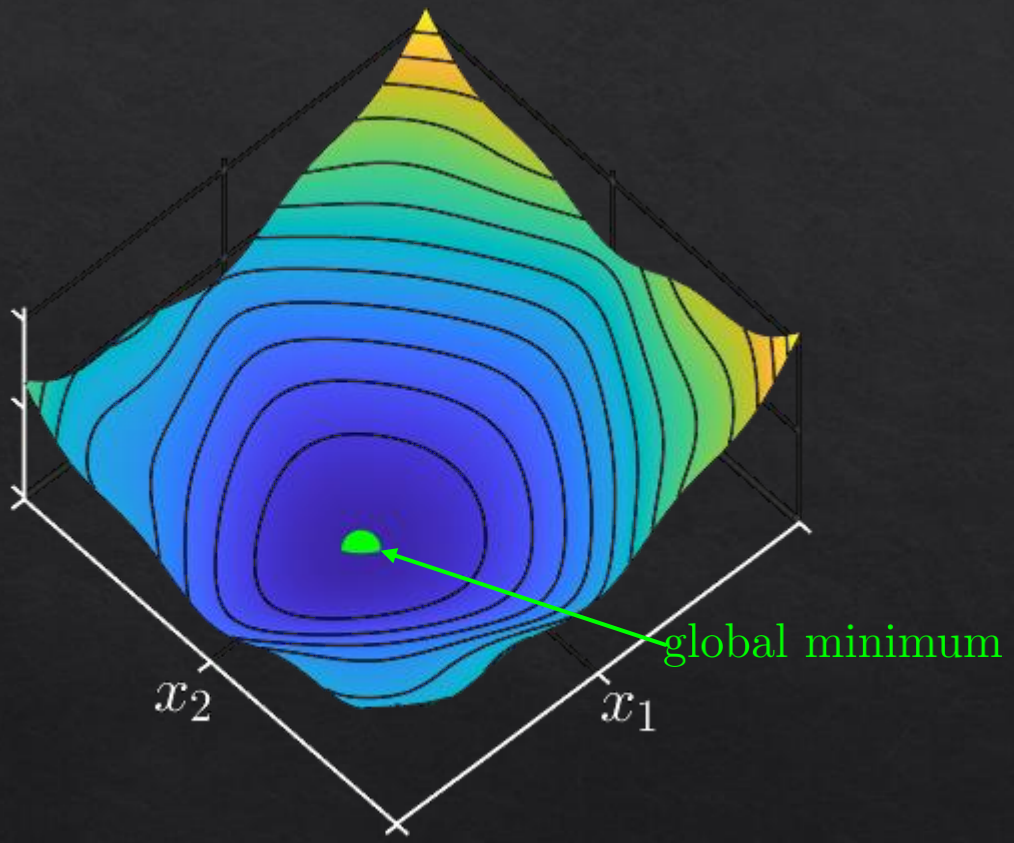
$\|\nabla f(\boldsymbol{x})\| \approx 0 \implies f$ is flat (around $\boldsymbol{x}$)

$\|\nabla f(\boldsymbol{x})\|$ is large $\implies$ steep slope (around $\boldsymbol{x}$)

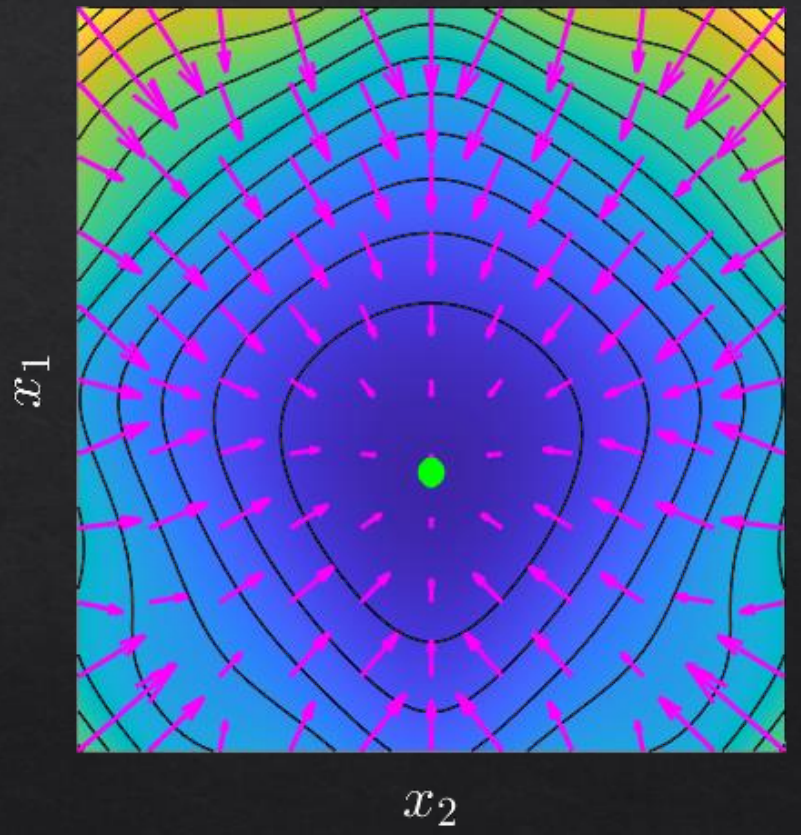
Gradient (Steepest) Descent

- Consider the following function:

$$f(x_1, x_2) = x_1^2 + x_2^2 + \sin^2(x_1 x_2) + x_1$$



$-\nabla f(\boldsymbol{x})$



- $-\nabla f$ points in the direction of (locally) maximum decrease in f.
- We can reach the global minimum by following the (minus) gradient.

$(*)$ – Note that f is not convex.

Gradient (Steepest) Descent – Algorithm

- **Input:** A function $f : \mathbb{R}^d \to \mathbb{R}$.
- **Output:** A local minimum $\boldsymbol{x}^\star \in \mathbb{R}^d$.

Step I:

Set an initial point $\boldsymbol{x}_0 \in \mathbb{R}^d$.

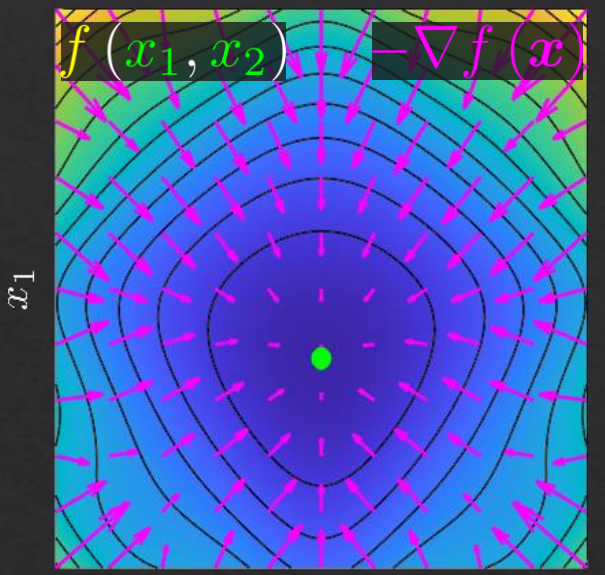
Step II:

1. **for** $k = 0, 1, 2, \ldots$ Update:

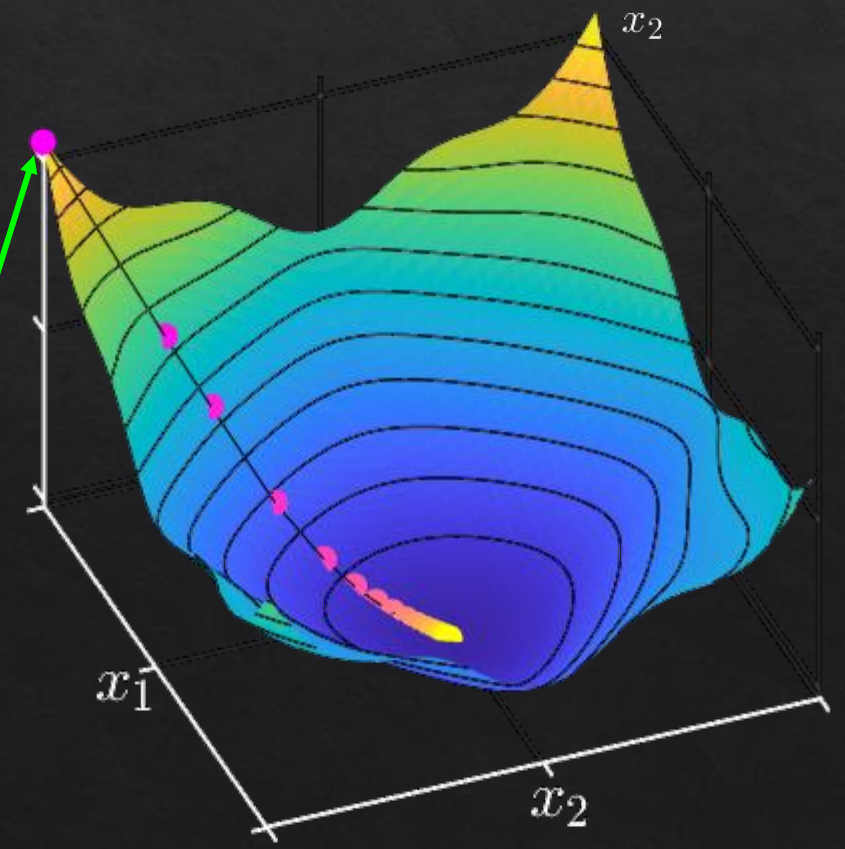
$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \mu \nabla f(\boldsymbol{x}_k)$$

2. **return** $\boldsymbol{x}^\star = \boldsymbol{x}_k$.

- $\mu \in \mathbb{R}$ is the step size (a.k.a. the learning rate).



$$\boldsymbol{x}_0 = \begin{bmatrix} 2 \\ 2 \end{bmatrix}$$



Gradient Descent – Example

Gradient Descent

Step I:

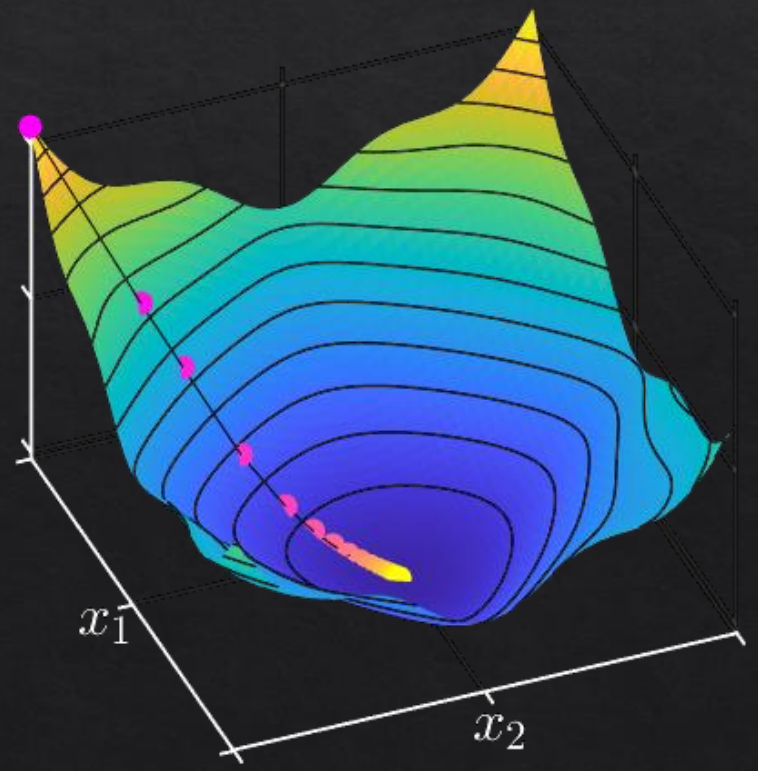
- Set an initial point $\boldsymbol{x}_0 \in \mathbb{R}^d$.

Step II:

1. **for** $k = 0, 1, 2, \ldots$ Update:

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \mu \nabla f(\boldsymbol{x}_k)$$

2. **return** $\boldsymbol{x}^\star = \boldsymbol{x}_k$.



Matrix Inversion Example

I – Original Image

$$I = \begin{bmatrix} | & & | \\ i_1 & \cdots & i_{256} \\ | & & | \end{bmatrix},$$

column stack

$$x := I^{\text{cs}} = \begin{bmatrix} i_1 \\ \vdots \\ i_{256} \end{bmatrix} \in \mathbb{R}^{256^2}$$

$$256^2 = 65,536$$

I_b – Blurred Image

$$y := I_b^{\text{cs}} \in \mathbb{R}^{256^2}$$

$$y = Hx$$

$$H \in \mathbb{R}^{65,536 \times 65,536}$$

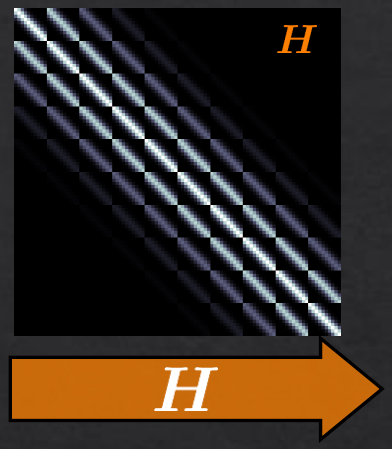
blurring matrix

$$\begin{bmatrix} | \\ y \\ | \end{bmatrix} = \begin{bmatrix} & H & \\ & & \end{bmatrix} \begin{bmatrix} | \\ x \\ | \end{bmatrix}$$

Matrix Inversion Example

$\boldsymbol{H} \in \mathbb{R}^{65,536 \times 65,536}$

I – Original Image



$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{x}$

I_b – Blurred Image

- Given $\boldsymbol{y}$ and $\boldsymbol{H}$, find $\boldsymbol{x}$.
- The solution is: $\boldsymbol{x} = \boldsymbol{H}^{-1}\boldsymbol{y}$
- Since $\boldsymbol{H} \in \mathbb{R}^{65,536 \times 65,536}$, standard computers cannot compute $\boldsymbol{H}^{-1}$.
- Using gradient descent we can solve:

$$\boldsymbol{x}^\star = \arg\min_{\tilde{\boldsymbol{x}}} \|\boldsymbol{y} - \boldsymbol{H}\tilde{\boldsymbol{x}}\|_2^2 \quad \Longrightarrow \quad \boldsymbol{x}^\star = \boldsymbol{x}$$

Matrix Inversion Example

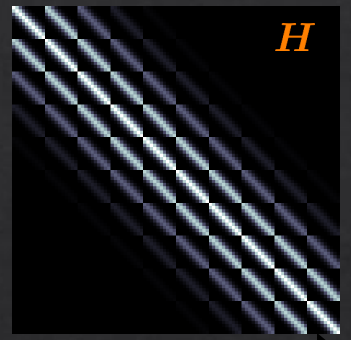
$$x^\star = \arg\min_{\tilde{x}} \underbrace{\|y - H\tilde{x}\|_2^2}_{:= f(\tilde{x})}$$

- The gradient is given by:

$$\left.\begin{aligned} f(x) &= (y - hx)^2 \\ f'(x) &= -2h(y - hx) \end{aligned}\right\} \text{1D}$$

$$\nabla f(x) = -2H^T(y - Hx)$$

I – Original Image



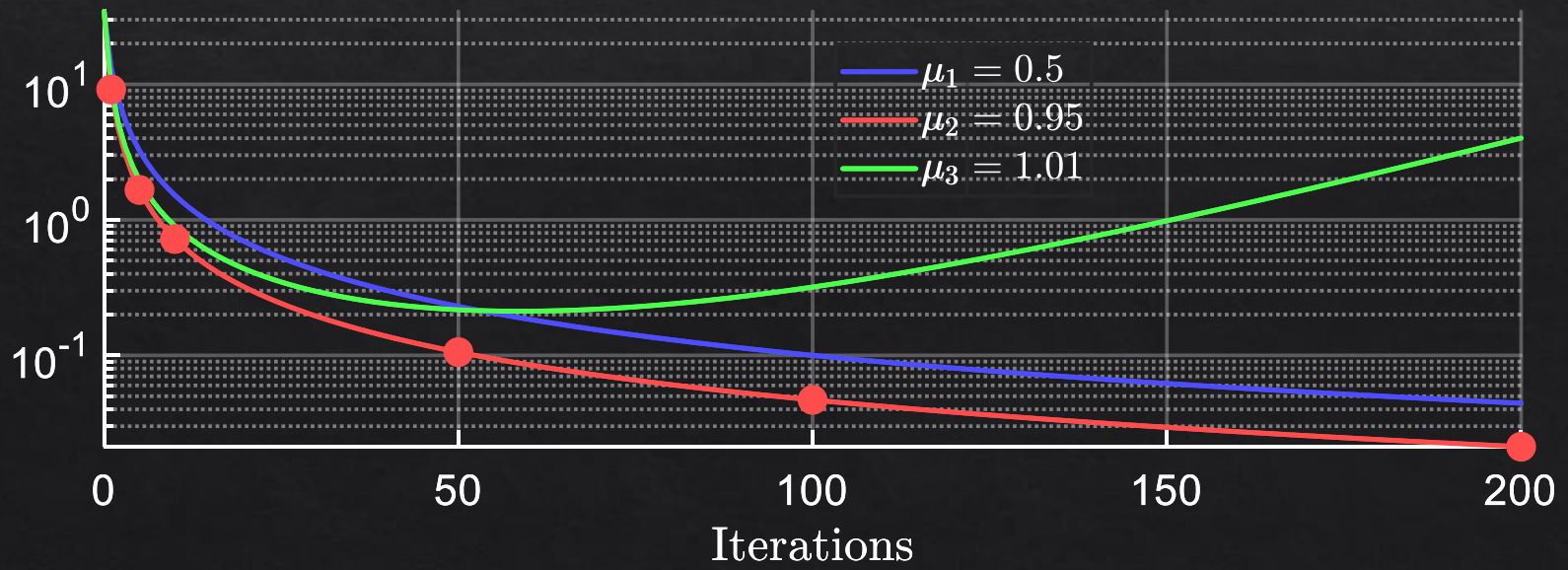
H

$y = Hx$

I_b – Blurred Image

$$x_{k+1} = x_k - \mu\nabla f(x_k)$$

$$f(x) = \|y - Hx\|_2^2$$



Can you order μ_1, μ_2, μ_3 from small to large?

$\mu_1 < \mu_2 < \mu_3$

Matrix Inversion Example

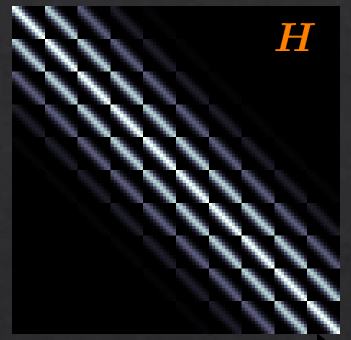
$$x^{\star} = \arg\min_{\tilde{x}} \underbrace{\|y - H\tilde{x}\|_2^2}_{:= f(\tilde{x})}$$

- The gradient is given by:

$$\left.\begin{aligned} f(x) &= (y - hx)^2 \\ f'(x) &= -2h(y - hx) \end{aligned}\right\} \text{1D}$$

$$\nabla f(x) = -2H^T(y - Hx)$$

I – Original Image



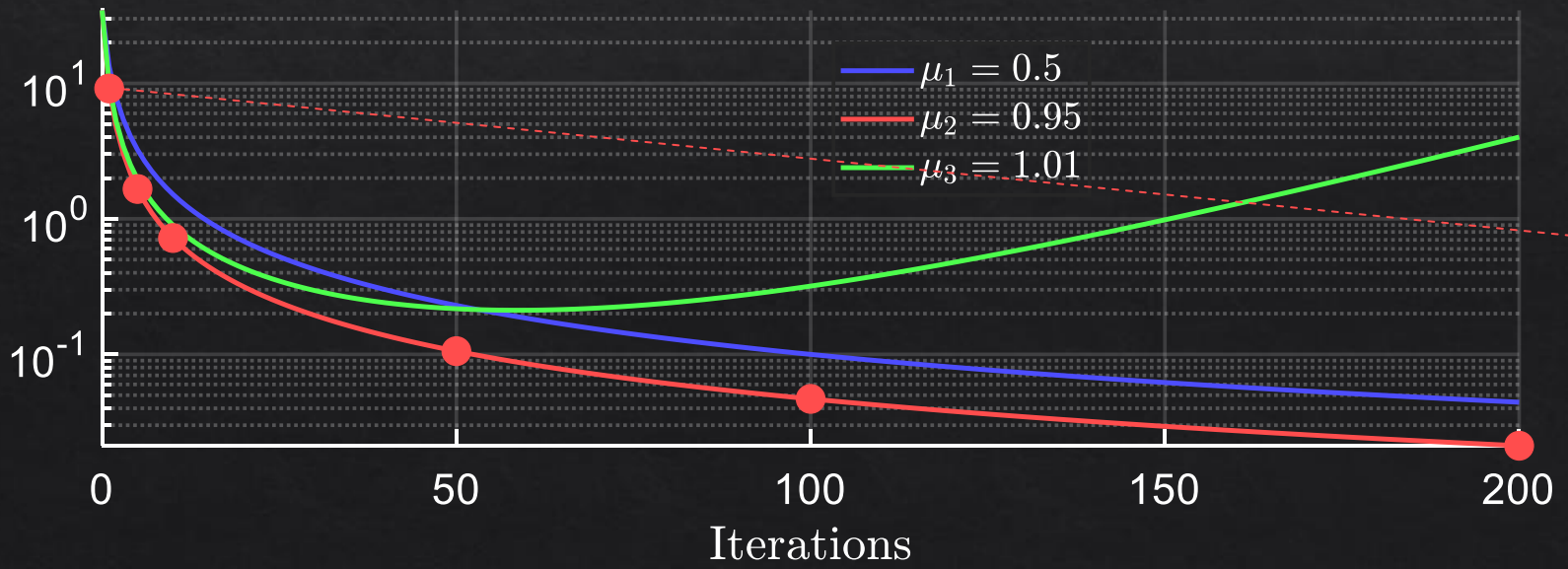
H

$y = Hx$

I_b – Blurred Image

$$x_{k+1} = x_k - \mu\nabla f(x_k)$$

$$f(x) = \|y - Hx\|_2^2$$



Matrix Inversion Example

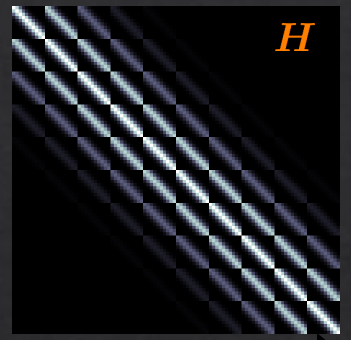
$$x^\star = \arg\min_{\tilde{x}} \underbrace{\|y - H\tilde{x}\|_2^2}_{:= f(\tilde{x})}$$

- The gradient is given by:

$$\left.\begin{aligned} f(x) &= (y - hx)^2 \\ f'(x) &= -2h(y - hx) \end{aligned}\right\} \text{1D}$$

$$\nabla f(x) = -2H^T(y - Hx)$$

I – Original Image



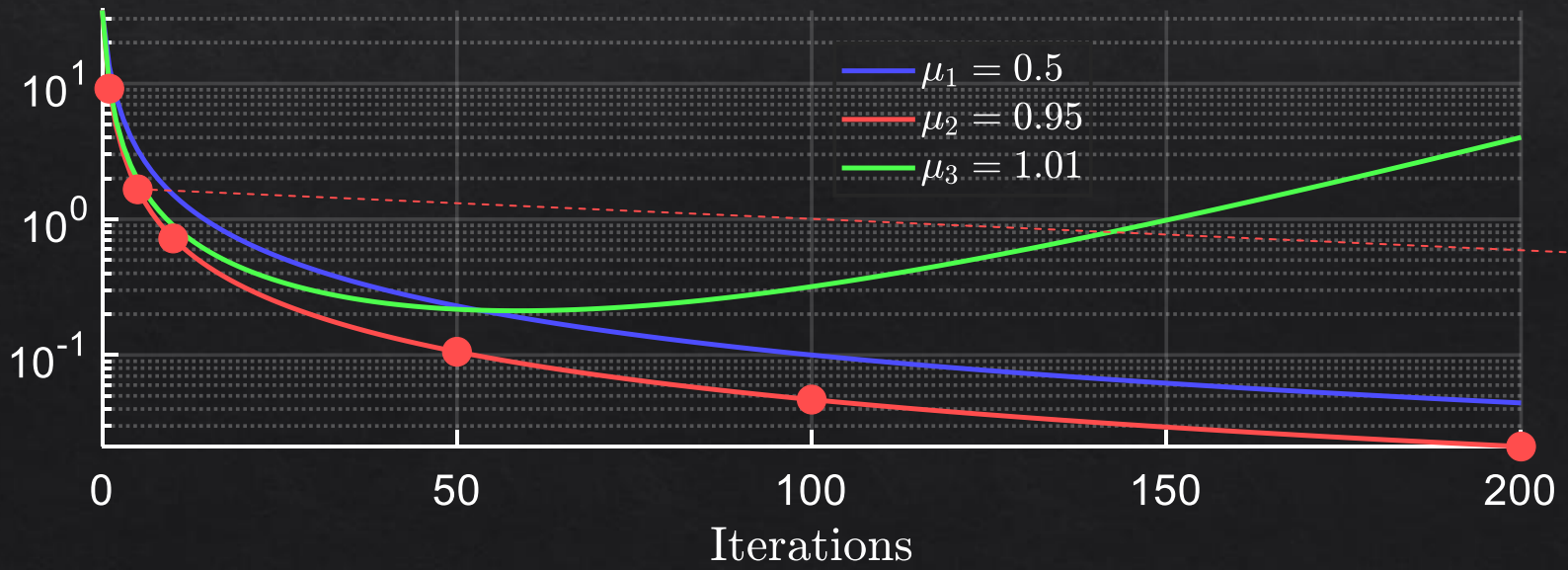
H

$y = Hx$

I_b – Blurred Image

$$x_{k+1} = x_k - \mu\nabla f(x_k)$$

$$f(x) = \|y - Hx\|_2^2$$



Matrix Inversion Example

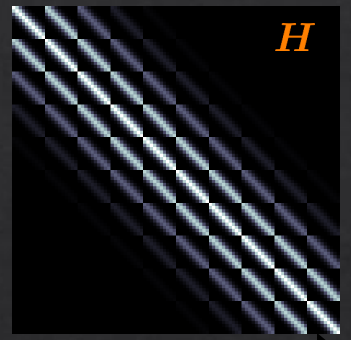
$$x^{\star} = \arg\min_{\tilde{x}} \underbrace{\|y - H\tilde{x}\|_2^2}_{:= f(\tilde{x})}$$

- The gradient is given by:

$$\left.\begin{aligned} f(x) &= (y - hx)^2 \\ f'(x) &= -2h(y - hx) \end{aligned}\right\} \text{1D}$$

$$\nabla f(x) = -2H^T(y - Hx)$$

I – Original Image



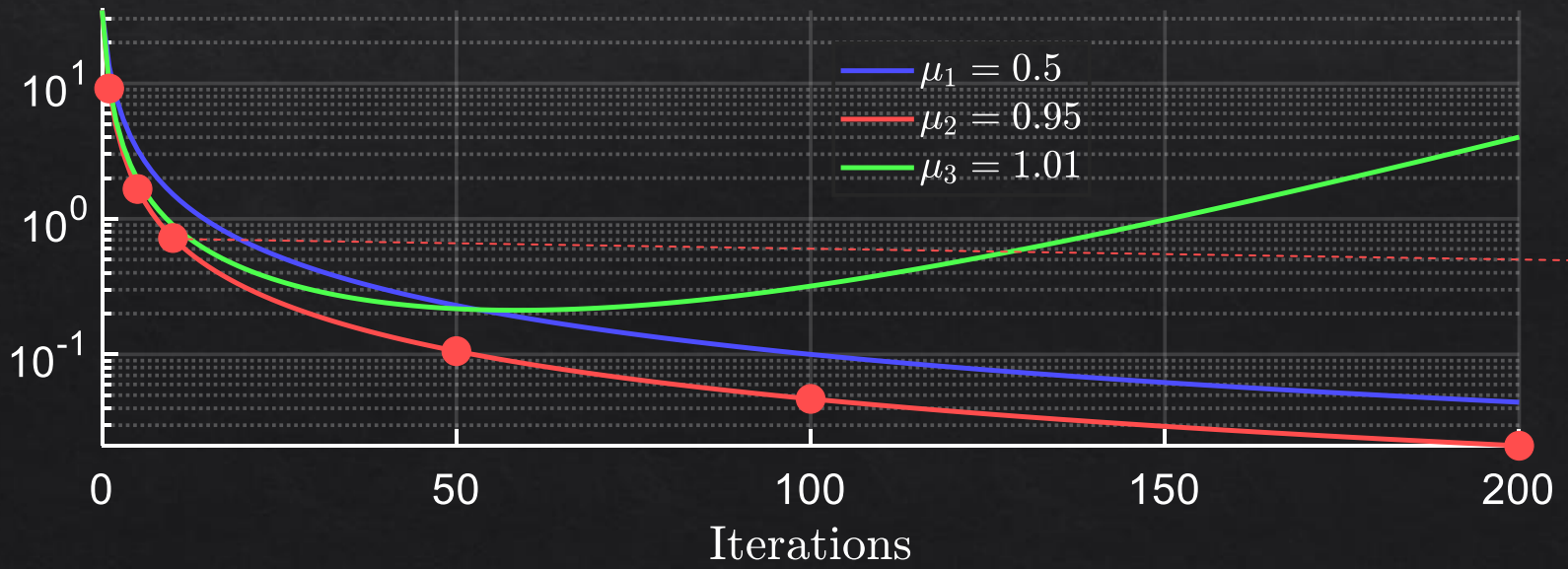
H

$y = Hx$

I_b – Blurred Image

$$x_{k+1} = x_k - \mu\nabla f(x_k)$$

$$f(x) = \|y - Hx\|_2^2$$



Matrix Inversion Example

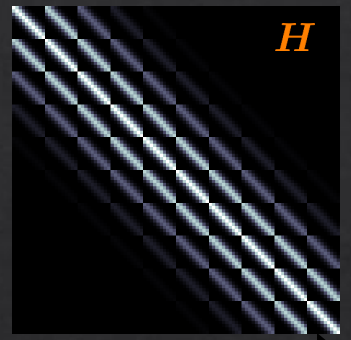
$$x^{\star} = \arg\min_{\tilde{x}} \underbrace{\|y - H\tilde{x}\|_2^2}_{:= f(\tilde{x})}$$

- The gradient is given by:

$$\left.\begin{aligned} f(x) &= (y - hx)^2 \\ f'(x) &= -2h(y - hx) \end{aligned}\right\} \text{1D}$$

$$\nabla f(x) = -2H^T(y - Hx)$$

I – Original Image



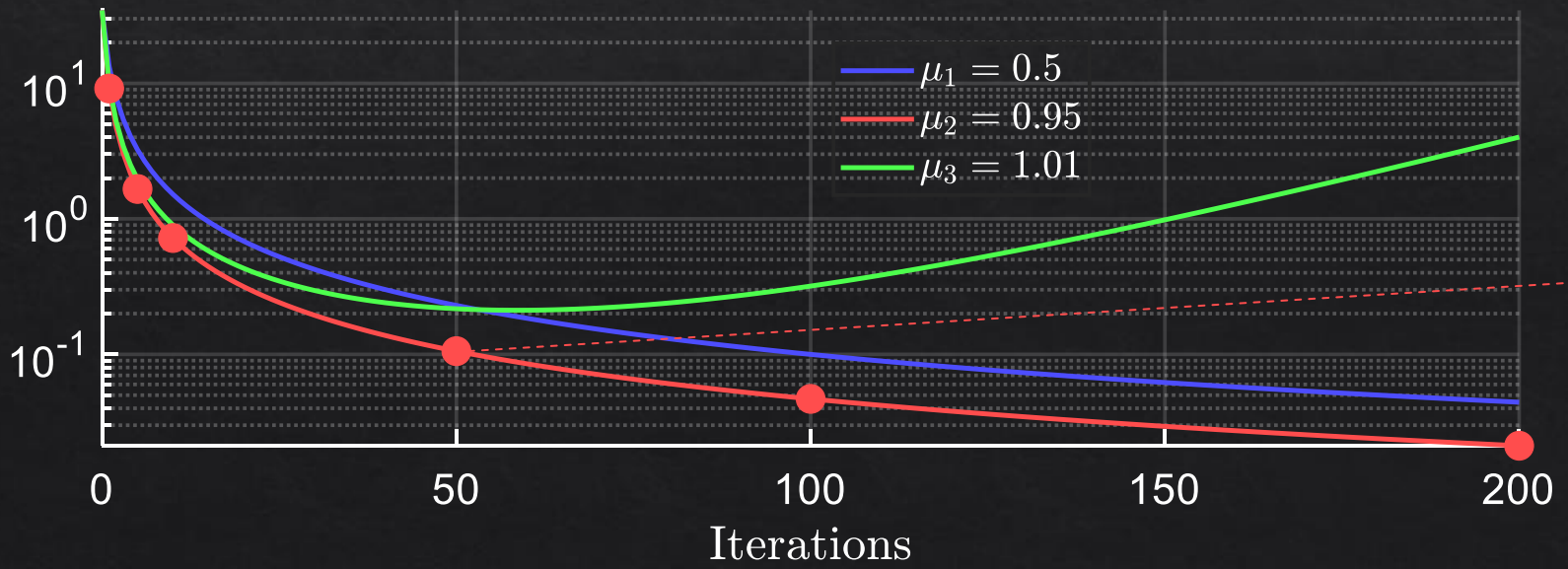
H

$y = Hx$

I_b – Blurred Image

$$x_{k+1} = x_k - \mu\nabla f(x_k)$$

$$f(x) = \|y - Hx\|_2^2$$



Matrix Inversion Example

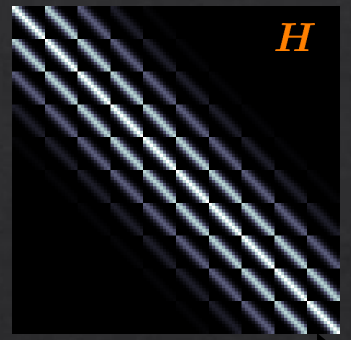
$$x^{\star} = \arg\min_{\tilde{x}} \underbrace{\|y - H\tilde{x}\|_2^2}_{:= f(\tilde{x})}$$

- The gradient is given by:

$$\left.\begin{aligned} f(x) &= (y - hx)^2 \\ f'(x) &= -2h(y - hx) \end{aligned}\right\} \text{1D}$$

$$\nabla f(x) = -2H^T(y - Hx)$$

I – Original Image



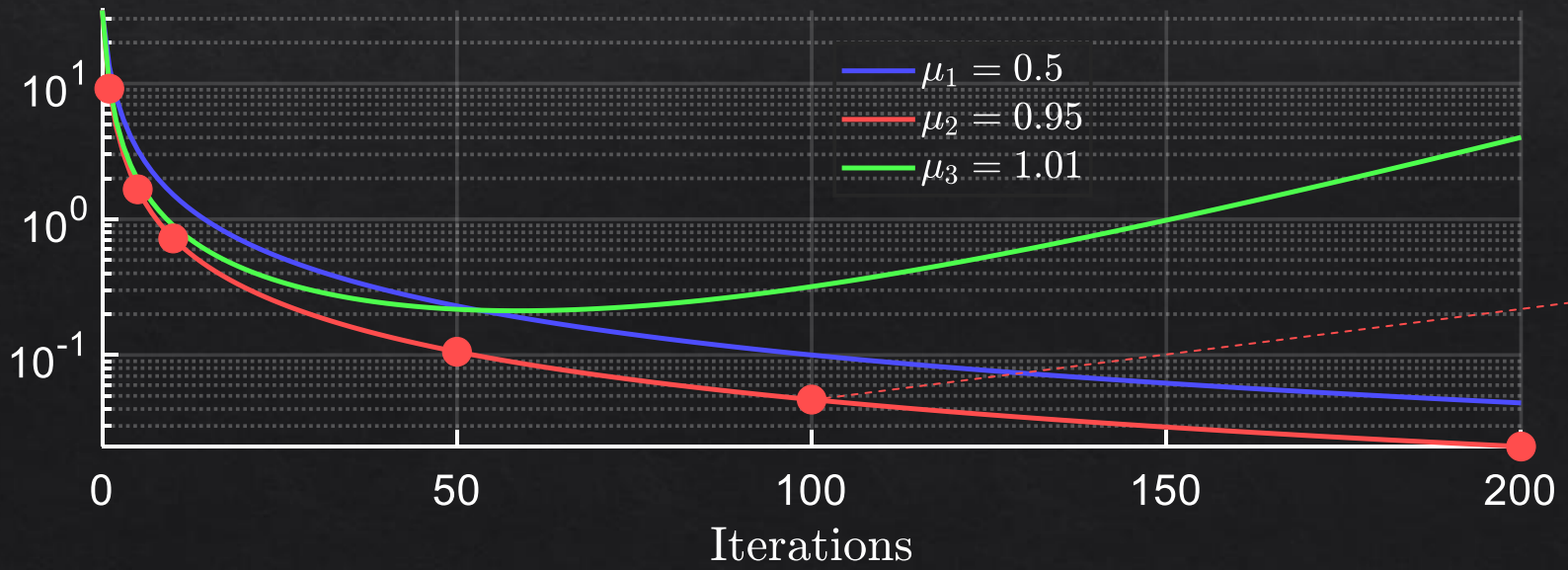
H

$y = Hx$

I_b – Blurred Image

$$x_{k+1} = x_k - \mu\nabla f(x_k)$$

$$f(x) = \|y - Hx\|_2^2$$



Matrix Inversion Example

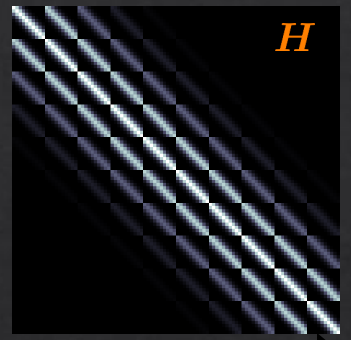
$$x^{\star} = \arg\min_{\tilde{x}} \underbrace{\|y - H\tilde{x}\|_2^2}_{:= f(\tilde{x})}$$

- The gradient is given by:

$$\left.\begin{aligned} f(x) &= (y - hx)^2 \\ f'(x) &= -2h(y - hx) \end{aligned}\right\} \text{1D}$$

$$\nabla f(x) = -2H^T(y - Hx)$$

I – Original Image



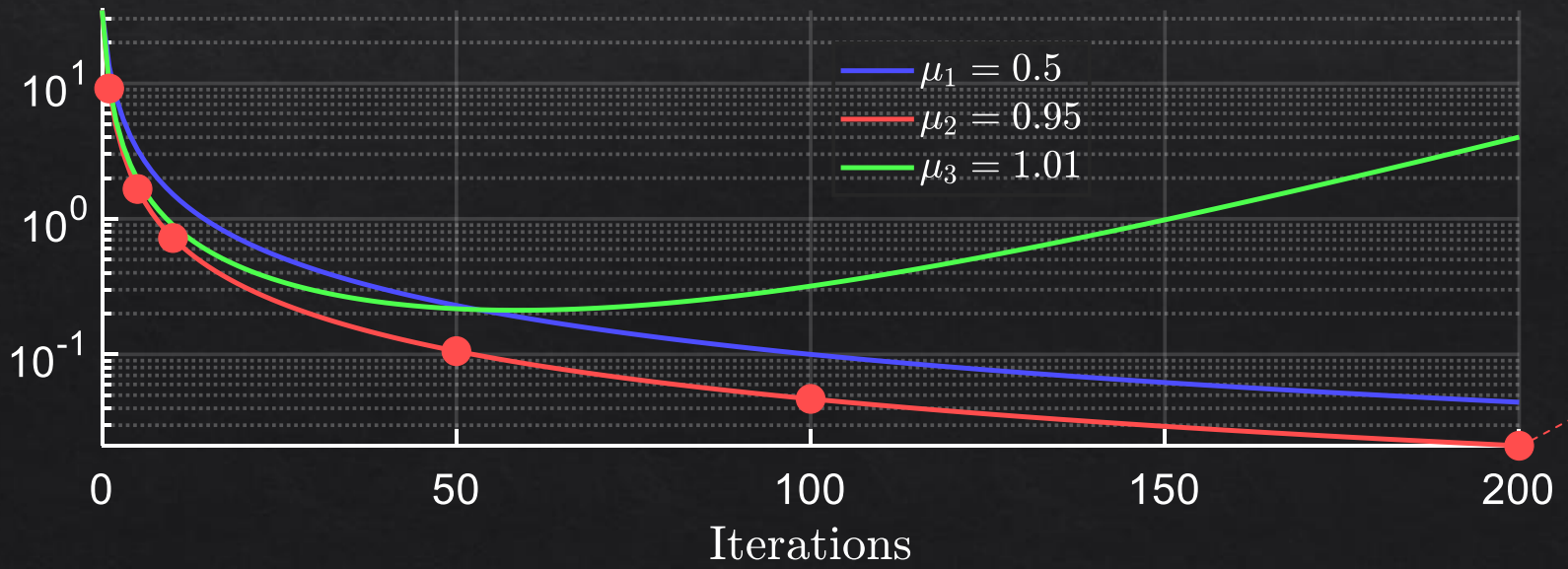
H

$y = Hx$

I_b – Blurred Image

$$x_{k+1} = x_k - \mu\nabla f(x_k)$$

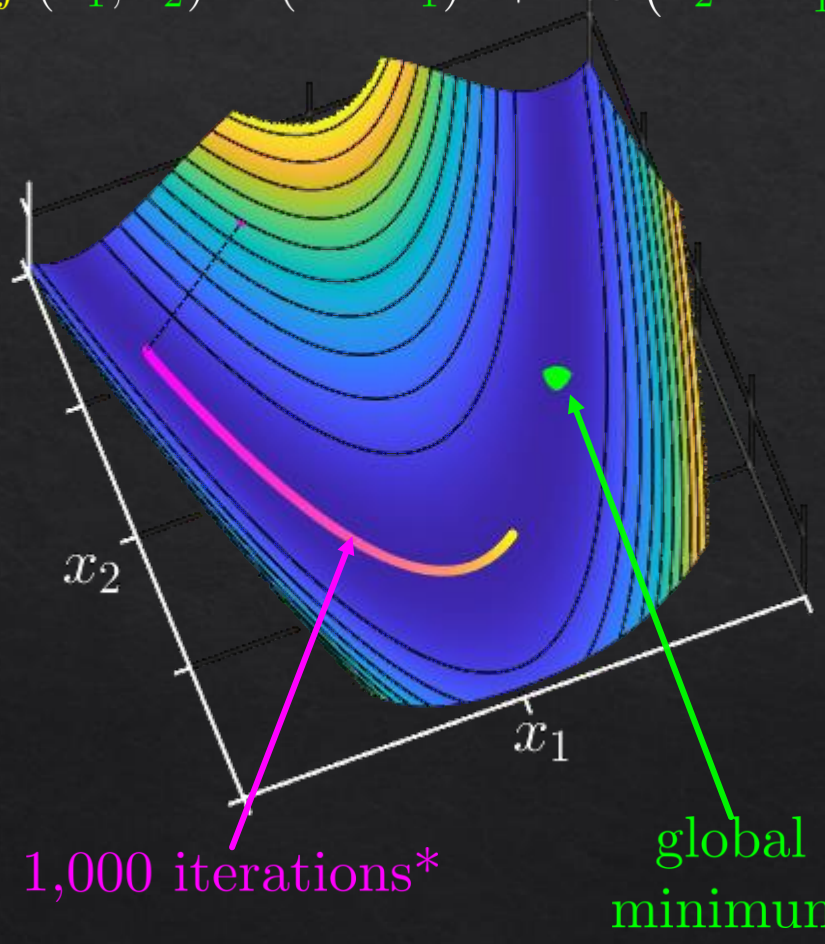
$$f(x) = \|y - Hx\|_2^2$$



Non-convex Functions

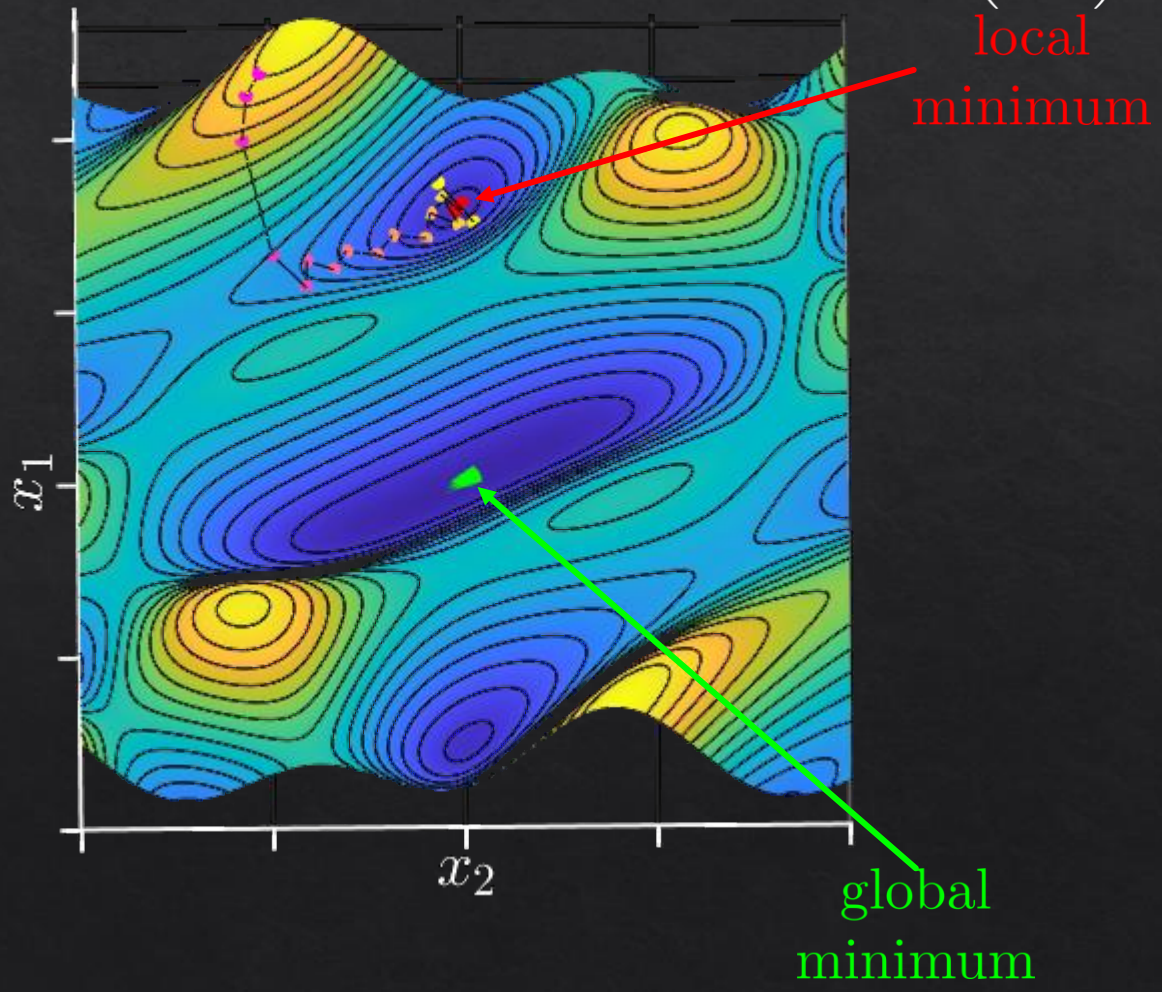
- Slow convergence – Rosenbrock function:

$$f(x_1, x_2) = (1 - x_1)^2 + 100\left(x_2 - x_1^2\right)^2$$



- Local minimum:

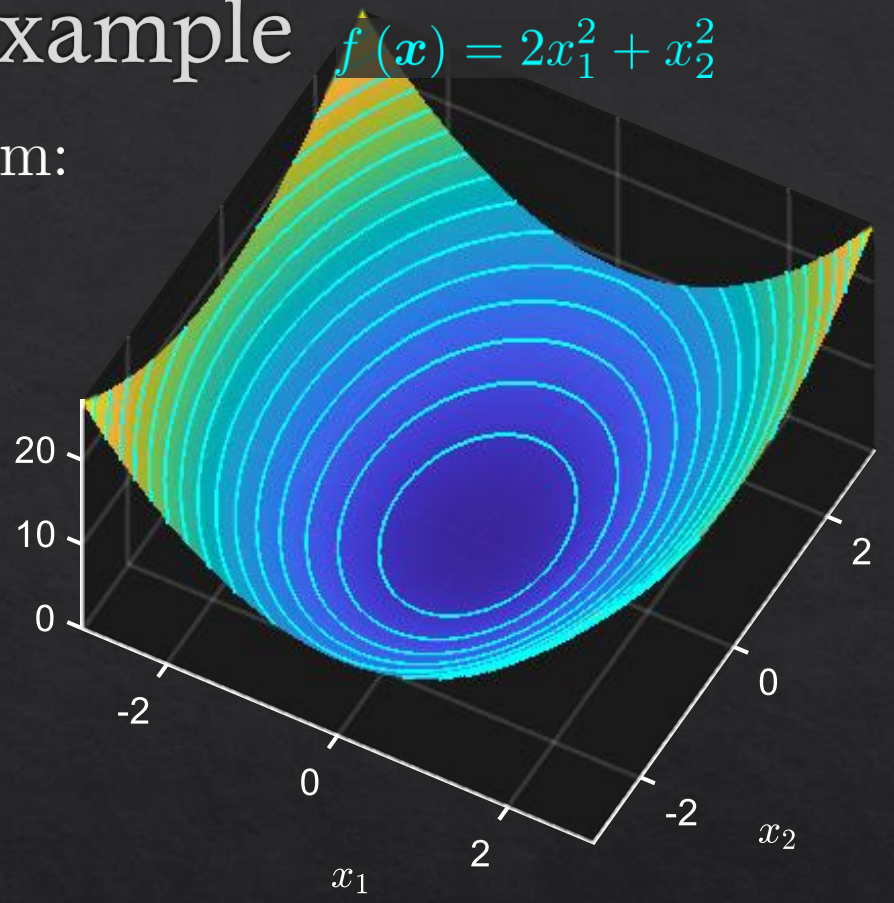
$$f(x_1, x_2) = \sin^2(x_1 x_2) + \sin^2(2x_1 + x_2) + \left(\frac{1}{5}x_1\right)^2$$



(∗) – with the optimal step size.

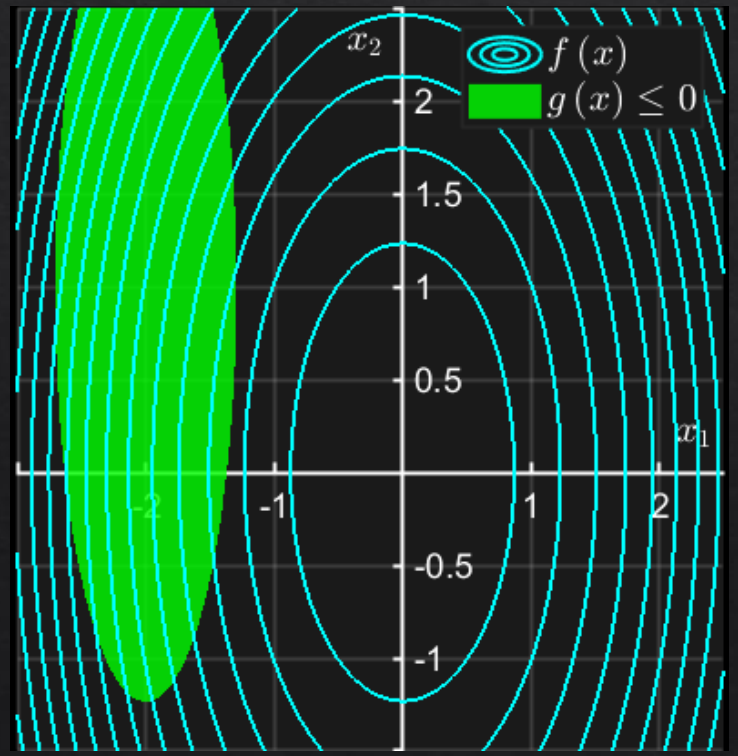
Constrained Optimization – A Simple Example

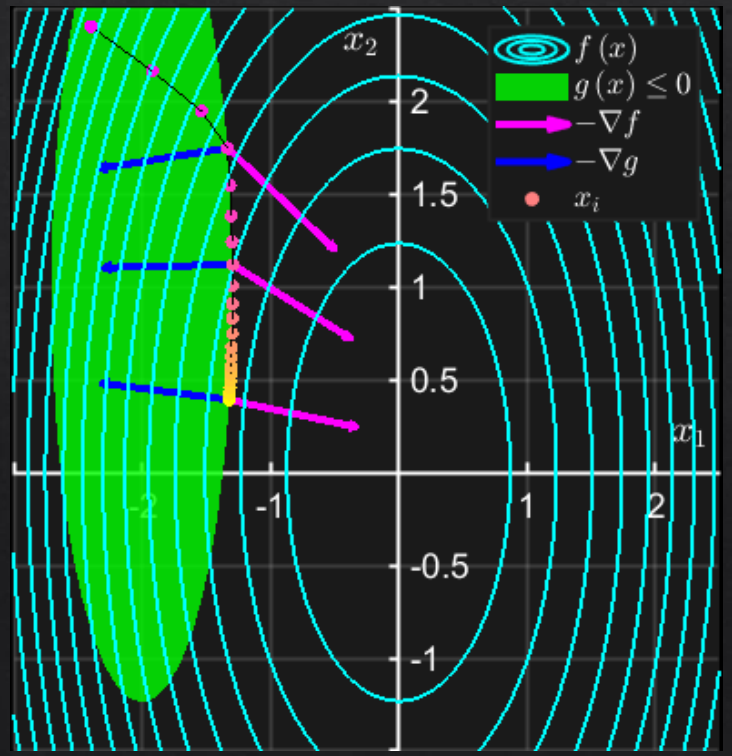
$f(\boldsymbol{x}) = 2x_1^2 + x_2^2$



- Consider the following constrained optimization problem:

$$\begin{cases} \arg\min\limits_{\boldsymbol{x}\in\mathbb{R}^2} f(\boldsymbol{x}) \\ \text{subject to} \\ g(\boldsymbol{x}) \leq 0 \end{cases} = \begin{cases} \arg\min\limits_{\boldsymbol{x}\in\mathbb{R}^2} 2x_1^2 + x_2^2 \\ \text{subject to} \\ 10(x_1+2)^2 + (x_2-1)^2 - 1 \leq 0 \end{cases}$$





we stop when

$$\nabla f(\boldsymbol{x}) = -\lambda \nabla g(\boldsymbol{x})$$

for some $\lambda \geq 0$

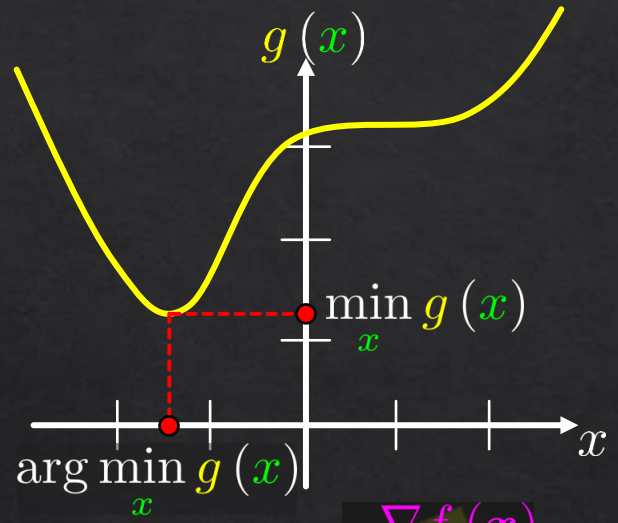
There are several algorithms for such problems (such as the augmented Lagrangian methods).

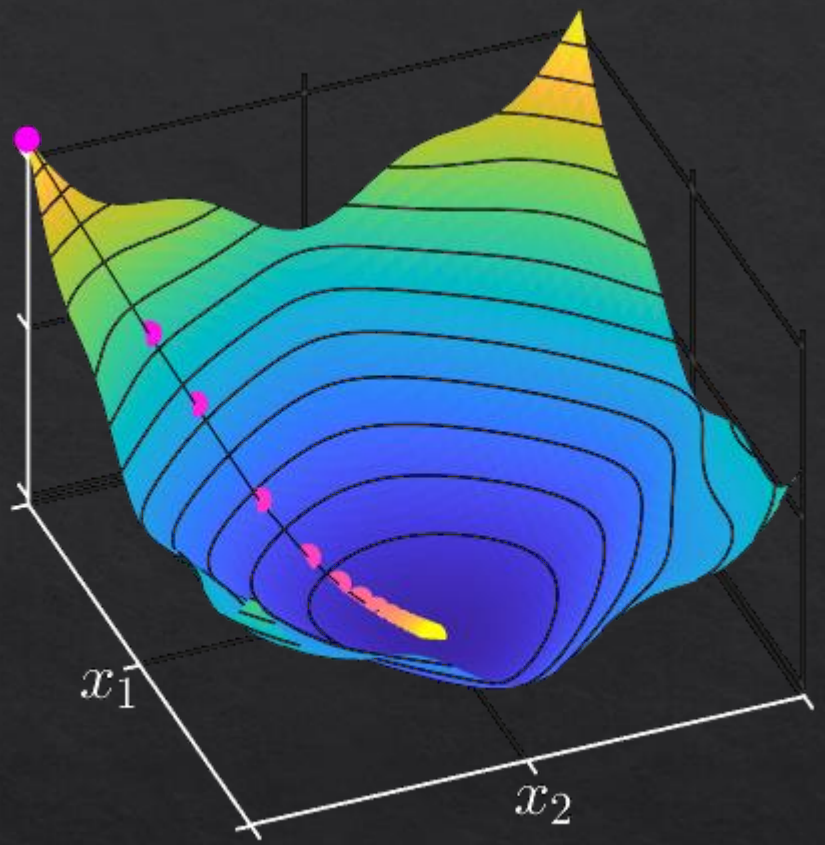
Other Important Topics

- Hessian and condition number
- Adaptive step size (and momentum)
- Line search
- Conjugate gradient
- Stochastic gradient descent
- Newton method (and quassi Newton)
- minimax problems
- The Lagrangian
- Dual problem

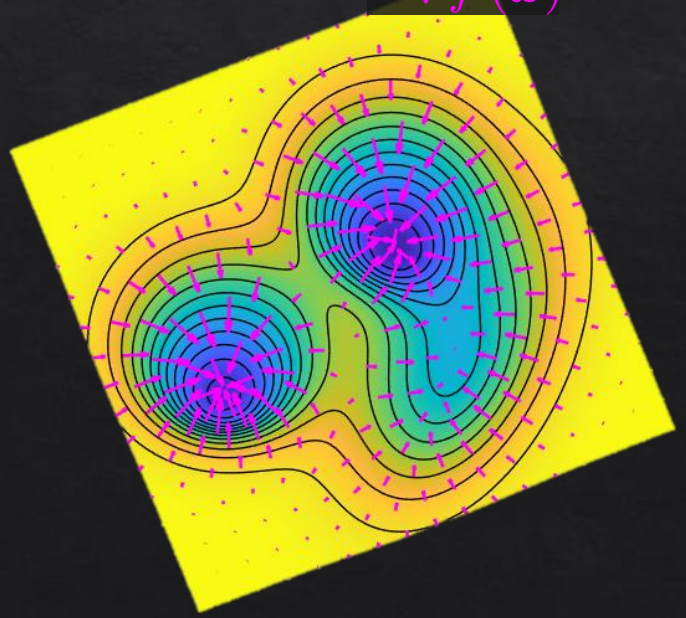
Questions

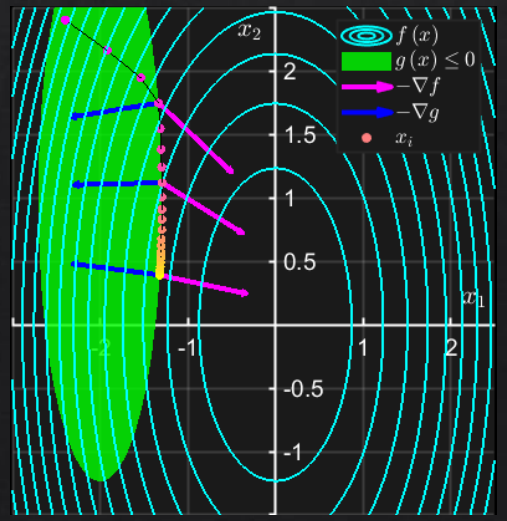
$$x^\star = \arg\min_{x \in \mathbb{R}} f(x)$$





$-\nabla f(\boldsymbol{x})$





fixel your pixels

fixelalgorithms.gitlab.io