ROB-GY 6323
reinforcement learning and optimal control for robotics

Lecture 2
Fundamentals of optimization
Linear Quadratic optimal control problems

Course material
All necessary material will be posted on Brightspace
Code will be posted on the Github site of the class

https://github.com/righetti/optlearningcontrol

Discussions/Forum with Slack

Contact
ludovic.righetti@nyu.edu
Office hours in person
Wednesday 3pm to 4pm
370 Jay street - room 801
(except next week)

Course Assistant
Armand Jordana
aj2988@nyu.edu
Office hours Monday 1pm to 2pm
Rogers Hall 515

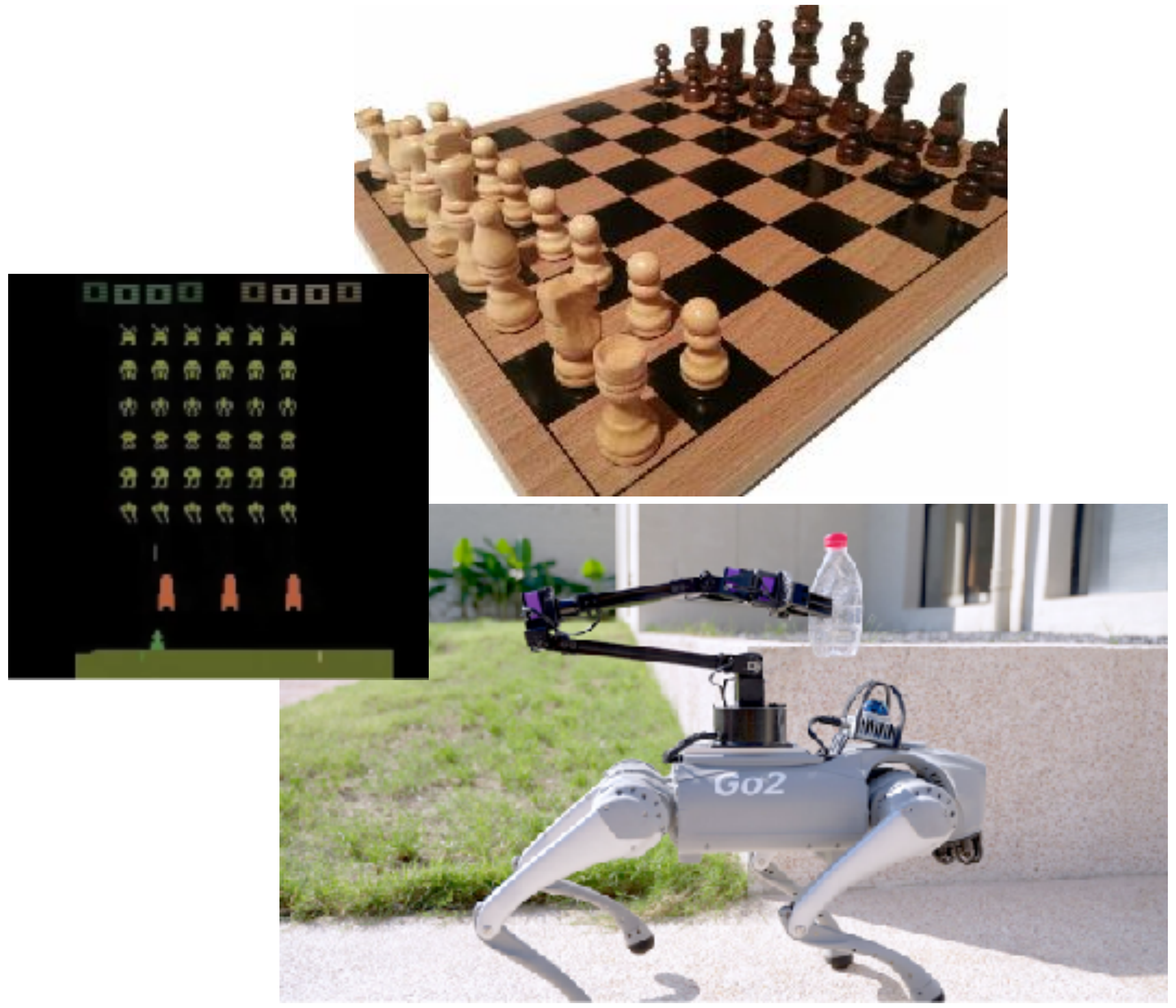
any other time by appointment only

A sequential decision making problem

A control decision

changes the “state” of

the dynamical system



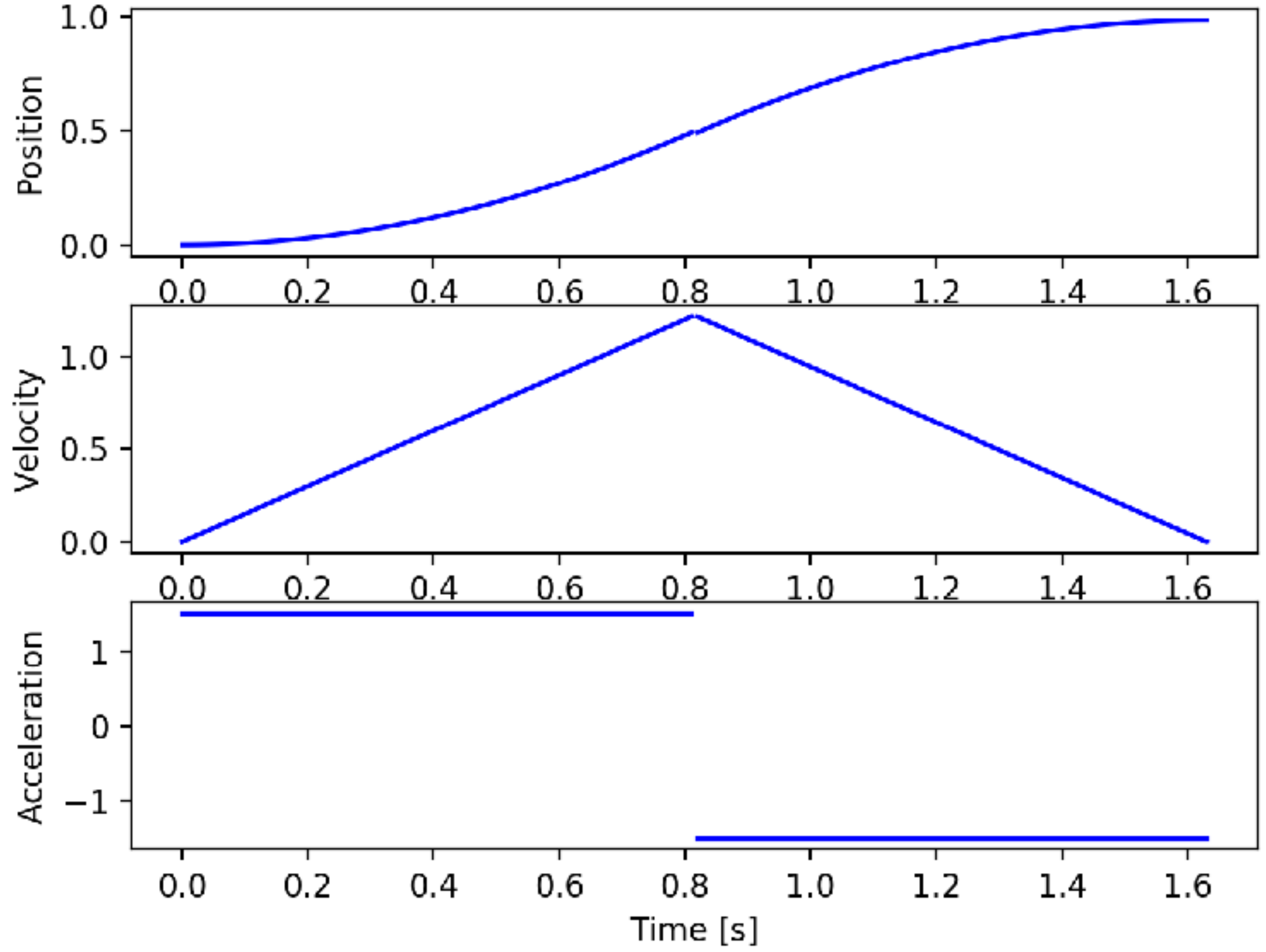
The problem: find the “**best sequence of actions**” to make the **dynamical system** behave as desired (e.g. win the game or do a backflip)

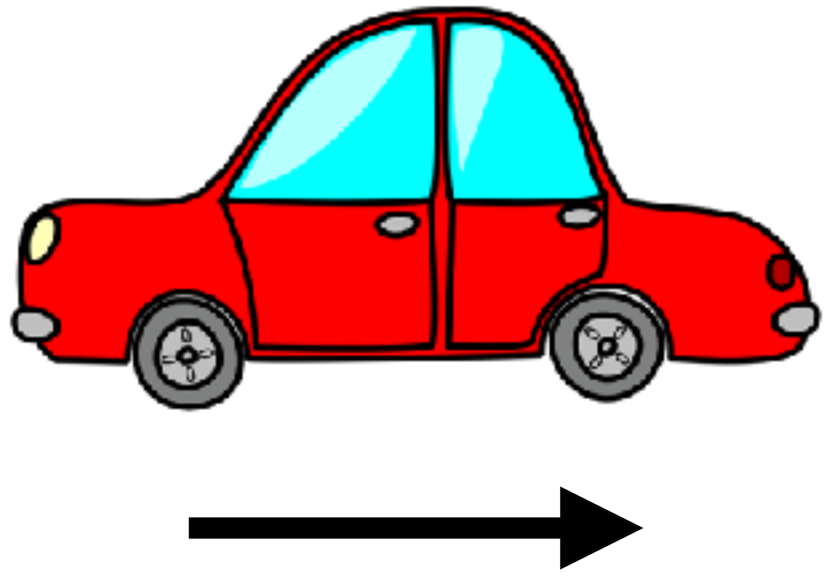
Best sequence of actions?

- How to accelerate a car to reach a goal in minimum time?
- How to accelerate a car to reach a goal in minimum acceleration?
- How to accelerate a car to minimize fuel consumption?
- How to accelerate a car to maximize passenger comfort?

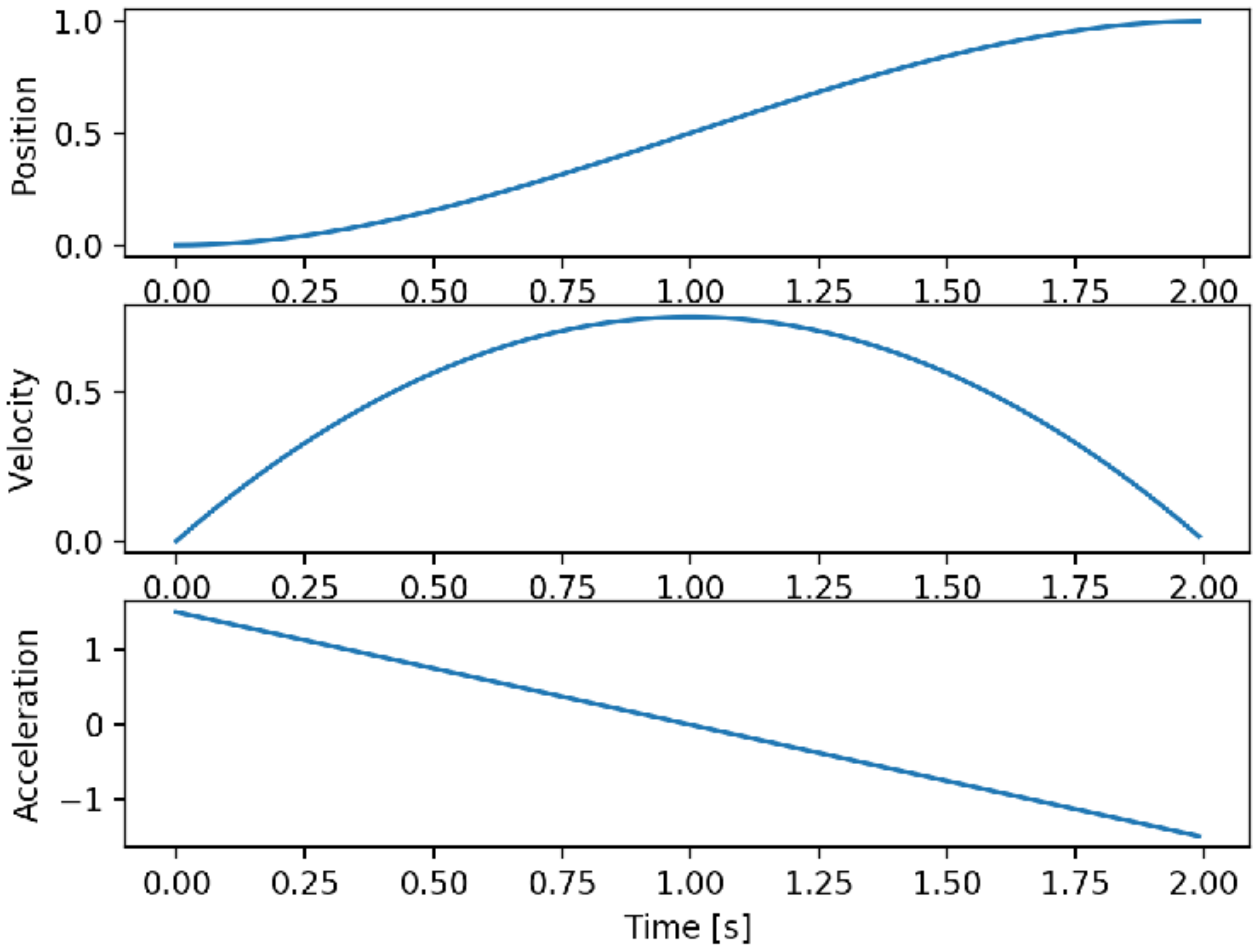
Different measures of “best” lead to very different answers

minimum time?





minimum acceleration?



Goal

×

Structure of an optimal control problem

$$\boxed{\min_{u_0, u_1, \cdots, u_N} \sum_{i=0}^{N} g_i(x_i, u_i)}$$

Find actions that optimize a performance cost

Subject to:

$$\boxed{\begin{aligned} & x_{n+1} = f(x_n, u_n) \\ & h_n(x_n, u_n) \leq 0 \end{aligned}}$$

to control a dynamical system (maybe with constraints)

This is an optimization problem

Basics of optimization

Off-the-shelf optimization algorithms

SciPy

Fundamental algorithms for scientific computing in Python

GET STARTED

```
scipy.optimize.minimize(fun, x0, args=(), method=None, jac=None, hess=None,
hessp=None, bounds=None, constraints=(), tol=None, callback=None, options=None)
```

Optimization (`scipy.optimize`)

- Unconstrained minimization of multivariate scalar functions (`minimize`)
  - Nelder-Mead Simplex algorithm (`method='Nelder-Mead'`)
  - Broyden-Fletcher-Goldfarb-Shanno algorithm (`method='BFGS'`)
    - Avoiding Redundant Calculation
  - Newton-Conjugate-Gradient algorithm (`method='Newton-CG'`)
    - Full Hessian example:
    - Hessian product example:
  - Trust-Region Newton-Conjugate-Gradient Algorithm (`method='trust-ncg'`)
    - Full Hessian example:
    - Hessian product example:
  - Trust-Region Truncated Generalized Lanczos / Conjugate Gradient Algorithm (`method='trust-krylov'`)
    - Full Hessian example:
    - Hessian product example:
  - Trust-Region Nearly Exact Algorithm (`method='trust-exact'`)
- Constrained minimization of multivariate scalar functions (`minimize`)
  - Trust-Region Constrained Algorithm (`method='trust-constr'`)
    - Defining Bounds Constraints:
    - Defining Linear Constraints:
    - Defining Nonlinear Constraints:
    - Solving the Optimization Problem:
  - Sequential Least SQuares Programming (SLSQP) Algorithm (`method='SLSQP'`)
- Global optimization
- Least-squares minimization (`least_squares`)
  - Example of solving a fitting problem
  - Further examples
- Univariate function minimizers (`minimize_scalar`)
  - Unconstrained minimization (`method='brent'`)
  - Bounded minimization (`method='bounded'`)
- Custom minimizers
- Root finding
  - Scalar functions
  - Fixed-point solving
  - Sets of equations
  - Root finding for large problems
  - Still too slow? Preconditioning.
- Linear programming (`linprog`)
  - Linear programming example
- Assignment problems
  - Linear sum assignment problem example
- Mixed integer linear programming
  - Knapsack problem example

Minimizing a function

$$\min_x f(x)$$

subject to

$$g(x) = 0$$

$$h(x) \leq 0$$

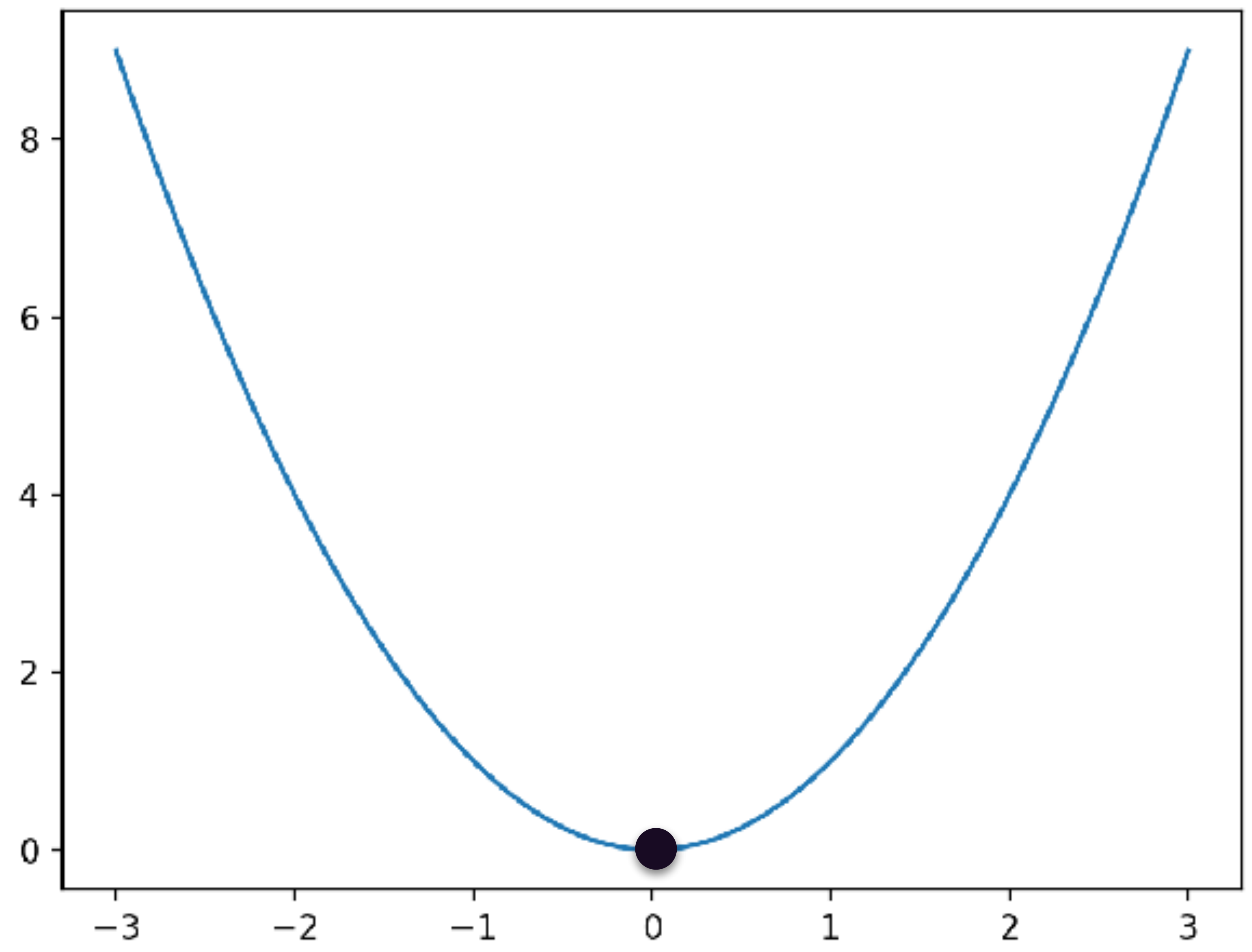
$x \in \mathbb{R}^n$ is a vector of variables

$f(x) : \mathbb{R}^n \leftarrow \mathbb{R}$ is the objective function
a scalar function we want to minimize or maximize
we will assume it is at least continuously differentiable

$g(x) = 0$ and $h(x) \leq 0$ are equality and inequality constraints
a set of equations that the variables x must satisfy

Example (unconstrained minimization)

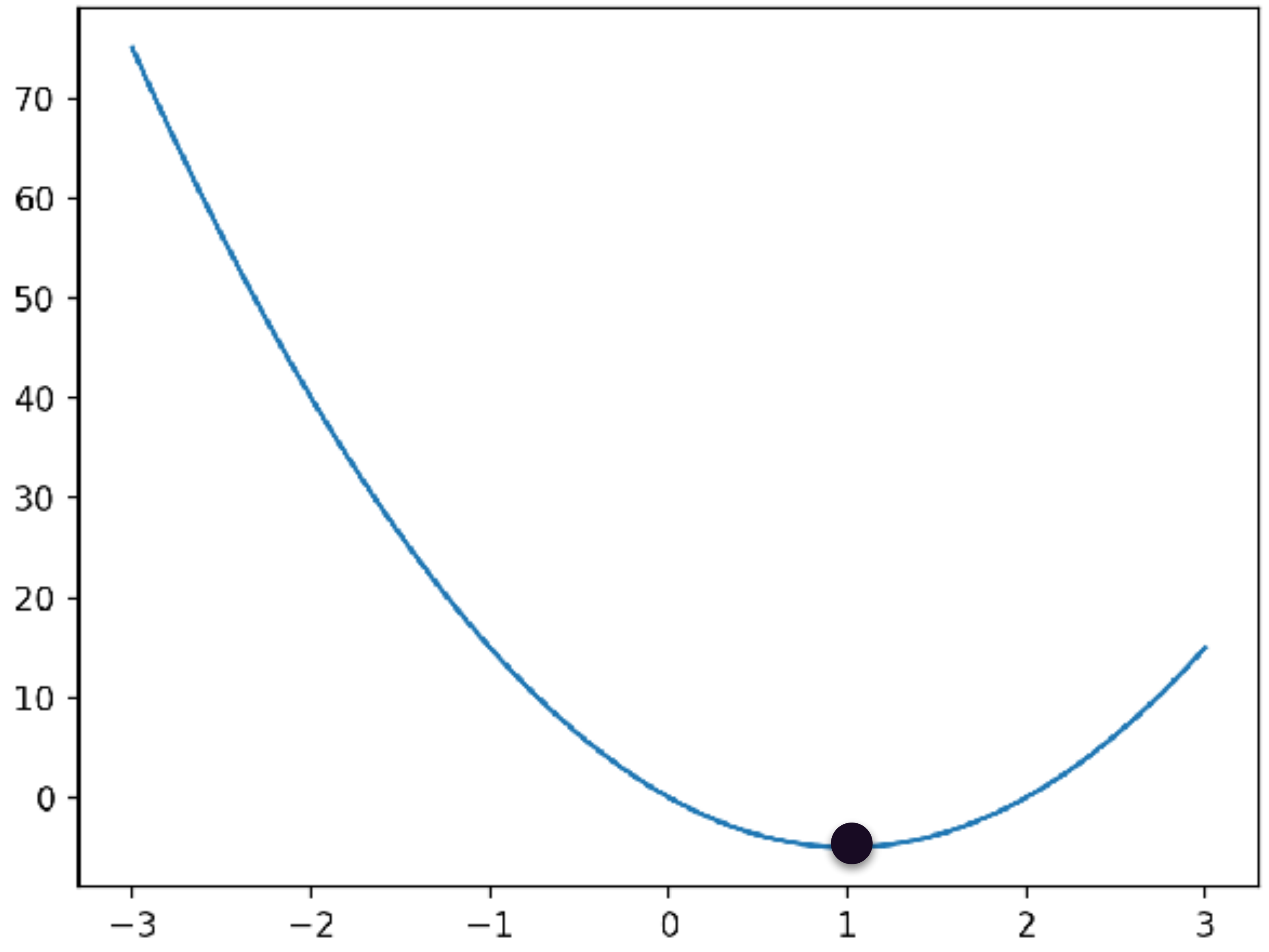
$$\min_x x^2$$



$f(x) = x^2$, its minimum is 0, which is reached when $x^* = 0$

Example (unconstrained minimization)

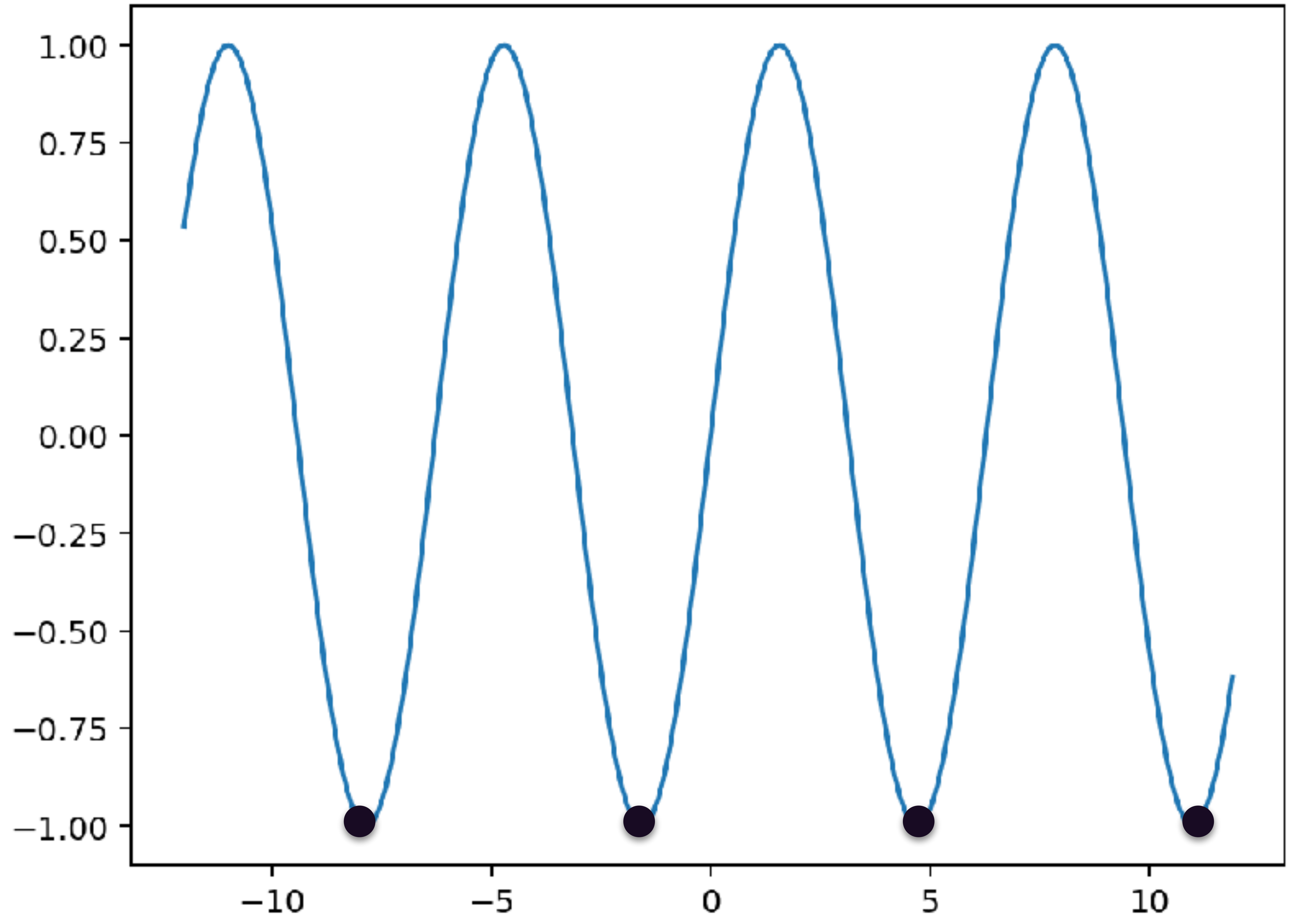
$$\min_x 5x^2 - 10x$$



$f(x) = 5x^2 - 10x$, its minimum is -5, which is reached when $x^* = 1$

Example (unconstrained minimization)

$$\min_x \sin(x)$$

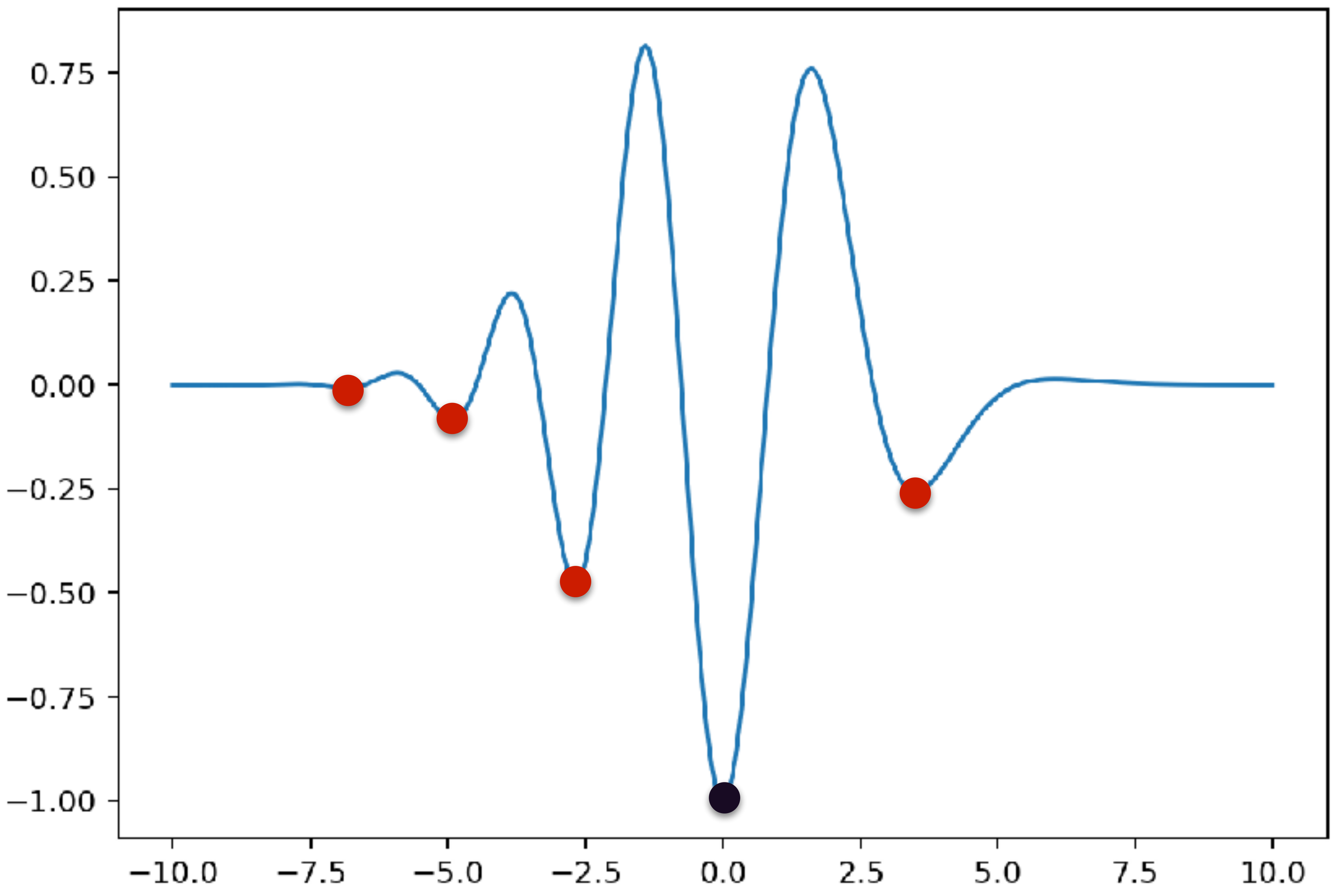


The minimum is -1

it is reached when $x^* = -\frac{\pi}{2} + k2\pi \quad \forall k \in \mathbb{N}$

Example (unconstrained minimization)

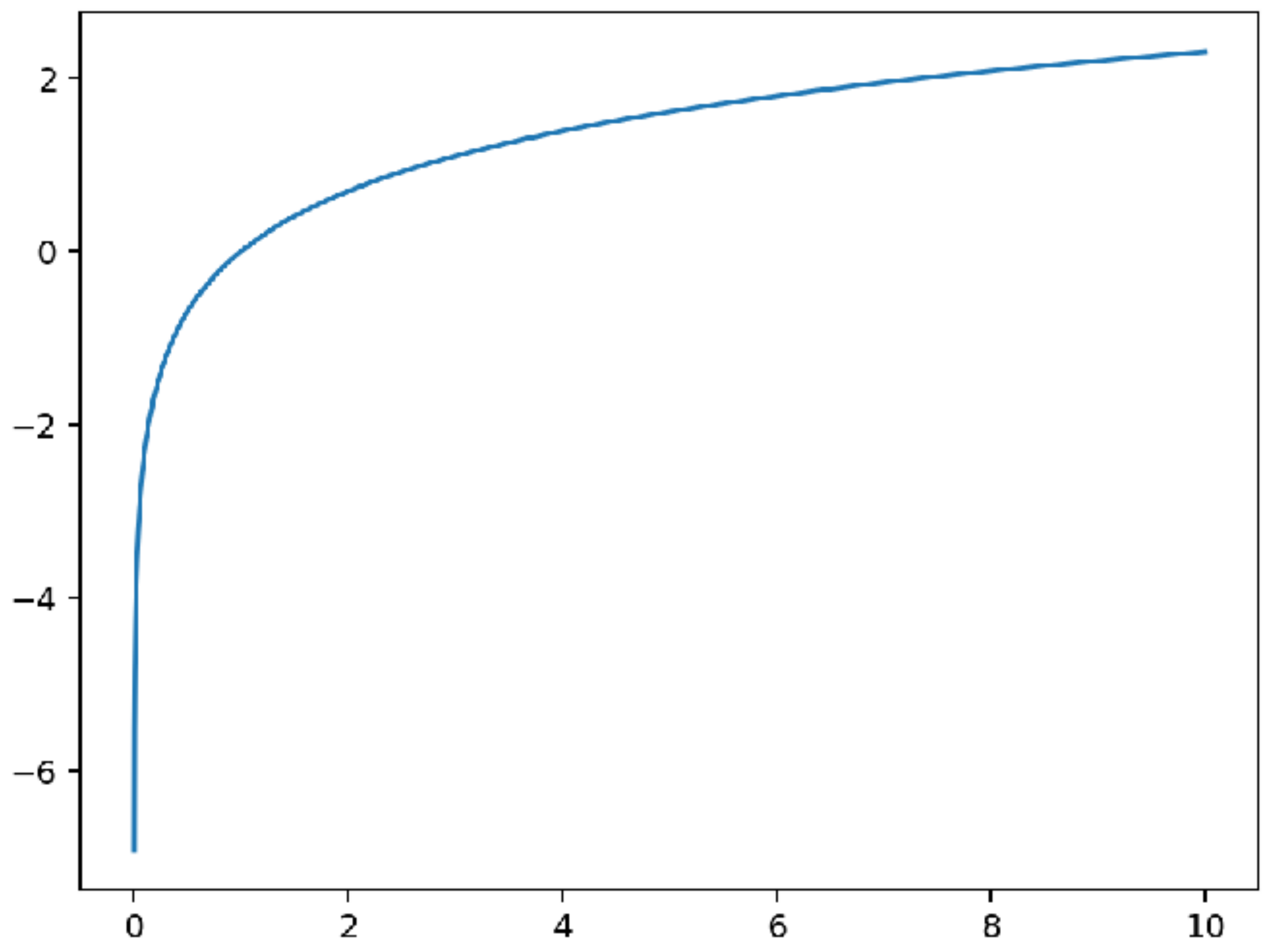
$$\min_x -\cos(2x - 0.1x^2)\mathrm{e}^{-0.1x^2}$$



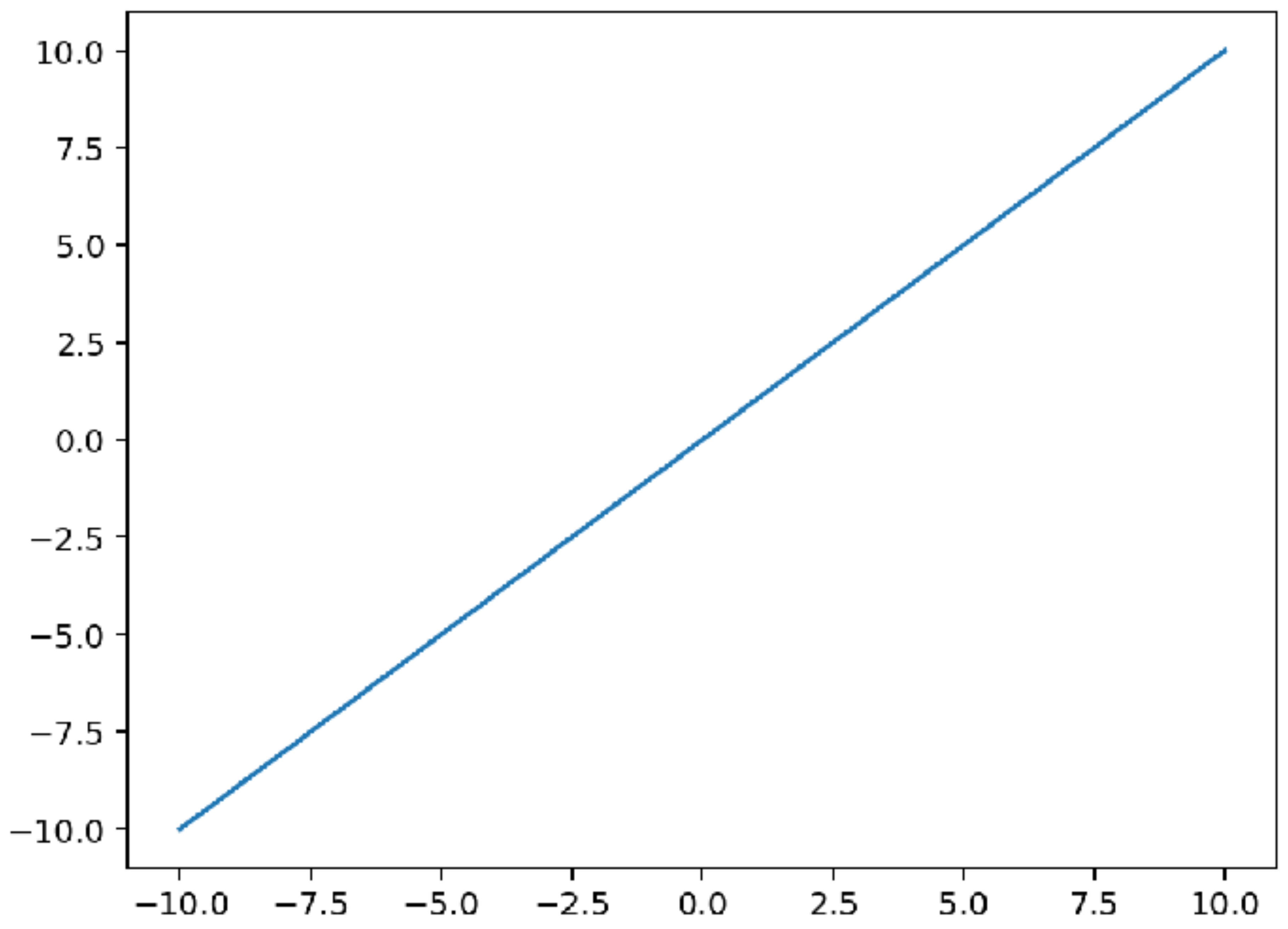
The (global) minimum is -1 and it is reached at $x = 0$
There are several other "local" minimum

Example (unconstrained minimization)

$$\min_x \log(x)$$



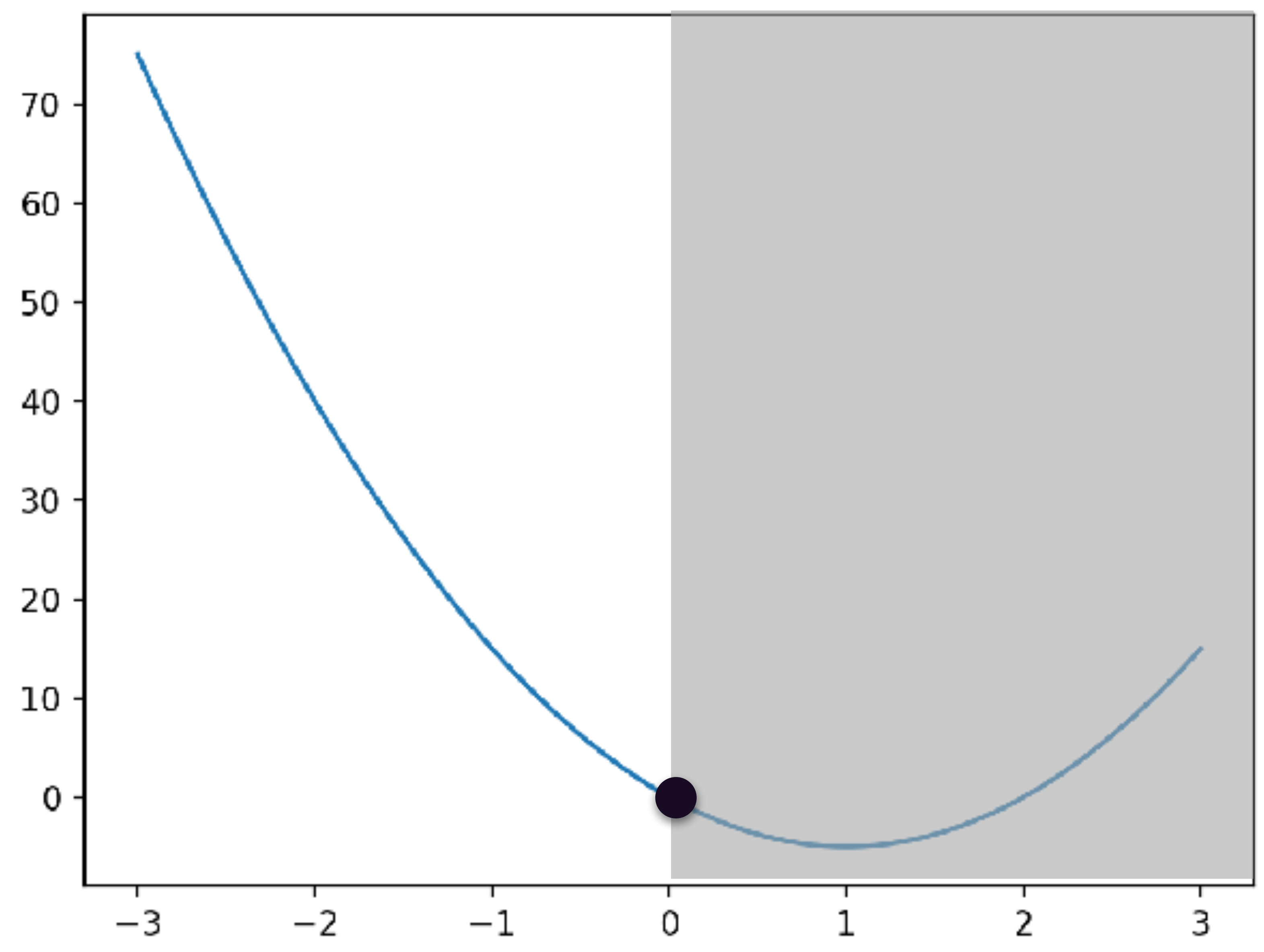
$$\min_x x$$



Constrained optimization

$$\min_x 5x^2 - 10x$$

subject to $\quad x \leq 0$

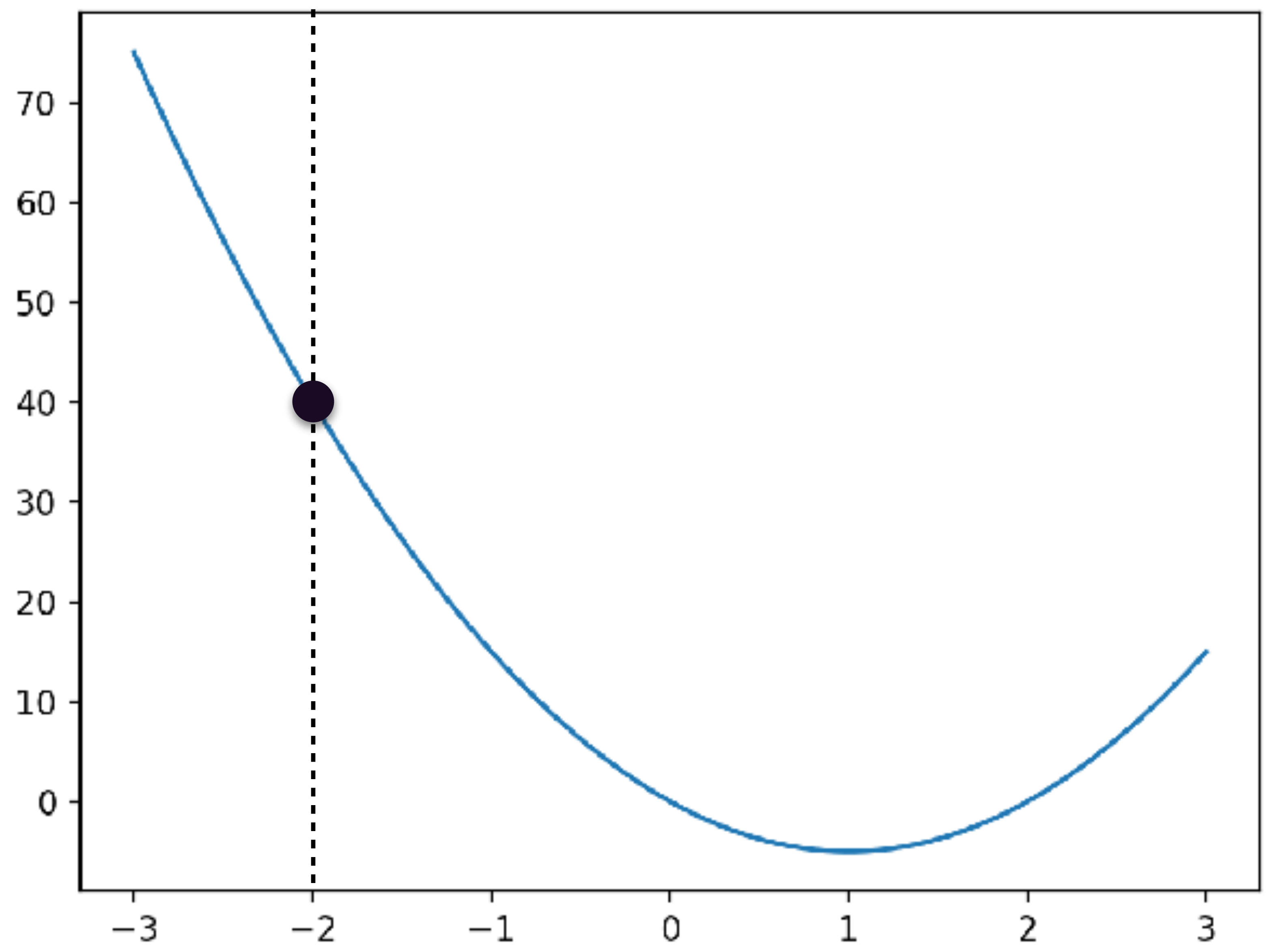


The minimum now 0 and it is reached at $x^* = 0$
(compared to a minimum of -5 without constraints, reached at $x = 1$)

Constrained optimization

$$\min_{x} 5x^2 - 10x$$

$$2x + 4 = 0$$



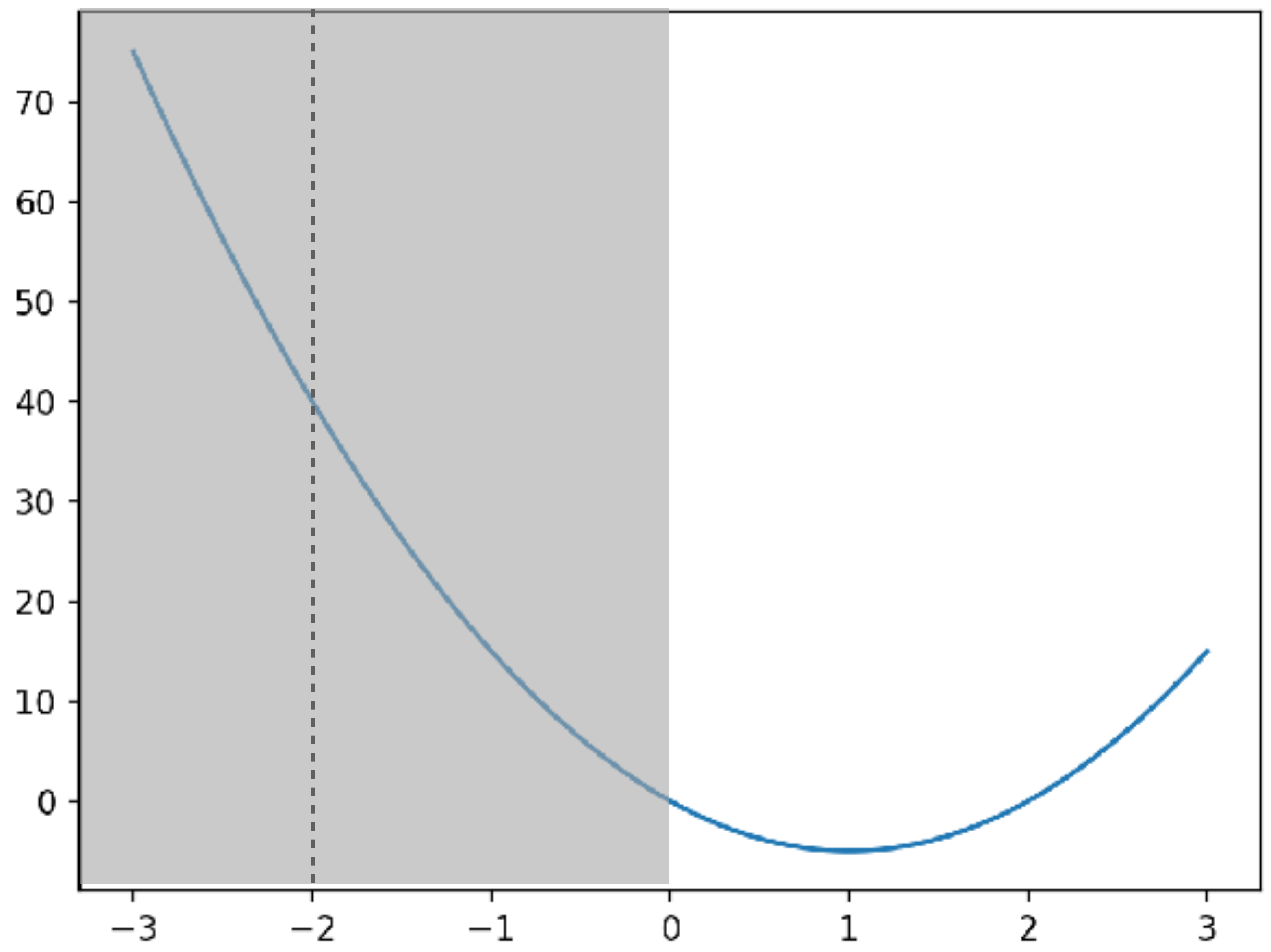
The minimum now 40 and it is reached at $x^* = -2$

Constrained optimization

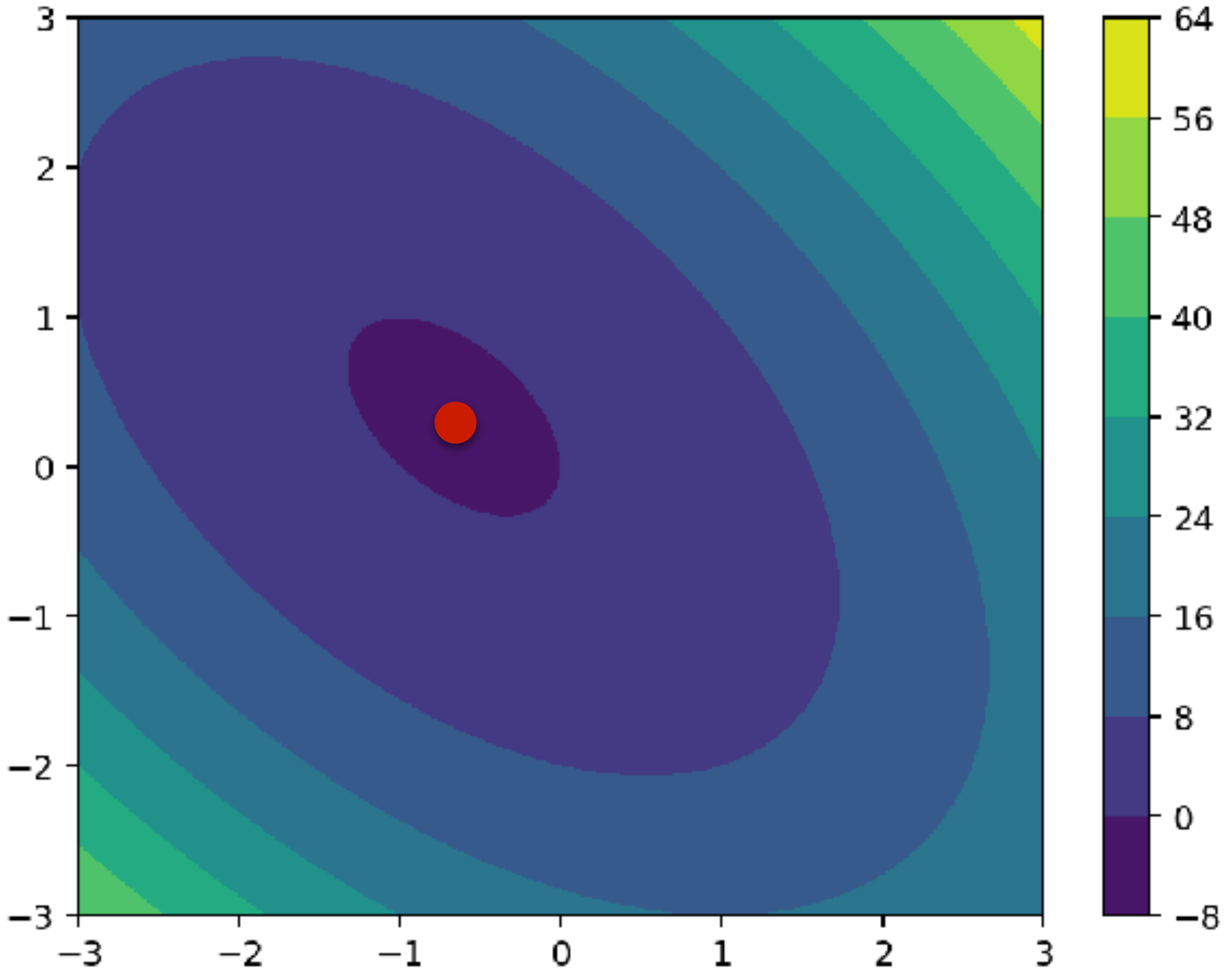
$$\min_x 5x^2 - 10x$$

$$2x + 4 = 0$$
$$-x < 0$$

Unfeasible

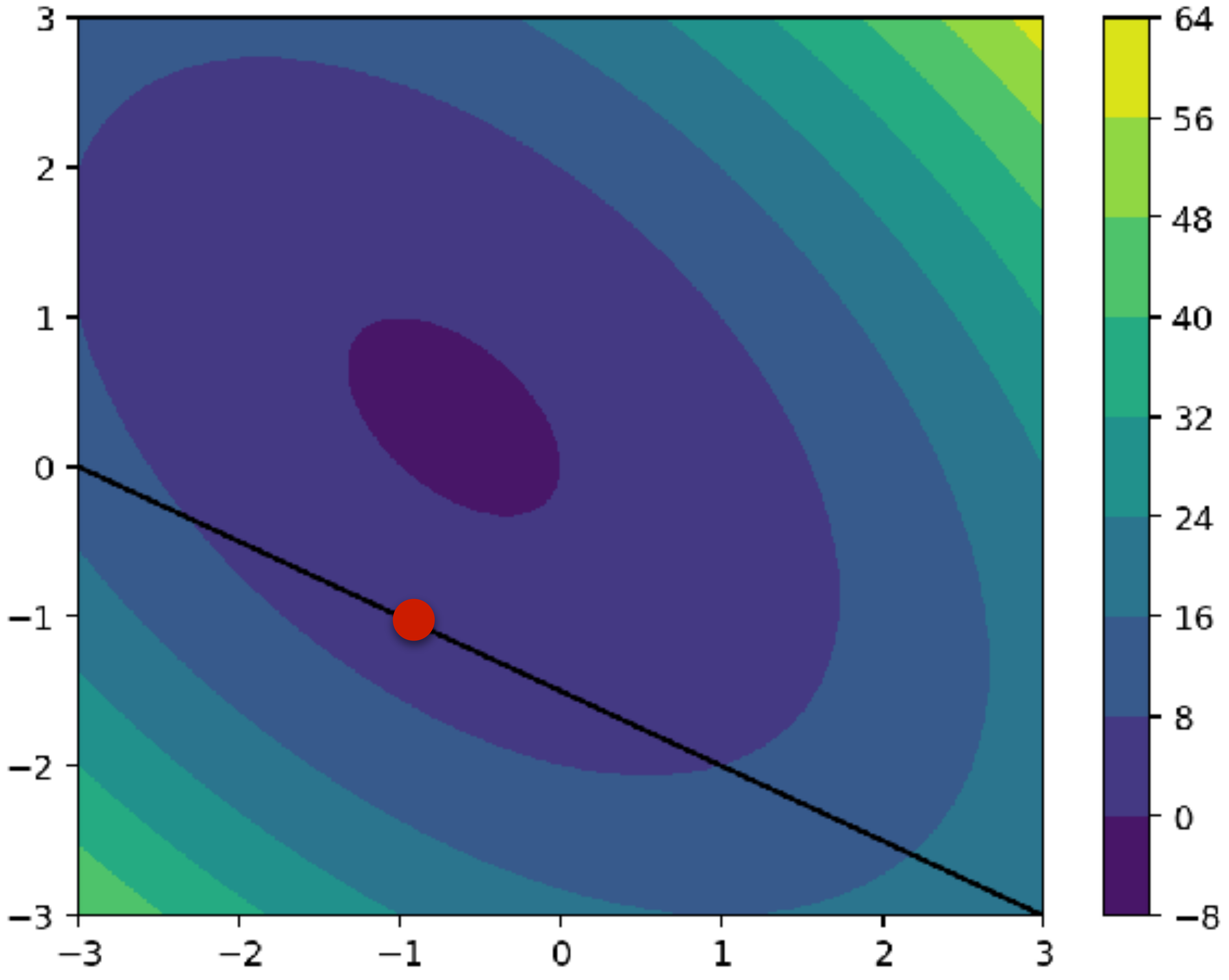


$$\min_{x_1,x_2} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}^T \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + 2x_1$$



$$\min_{x_1,x_2} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}^T \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + 2x_1$$

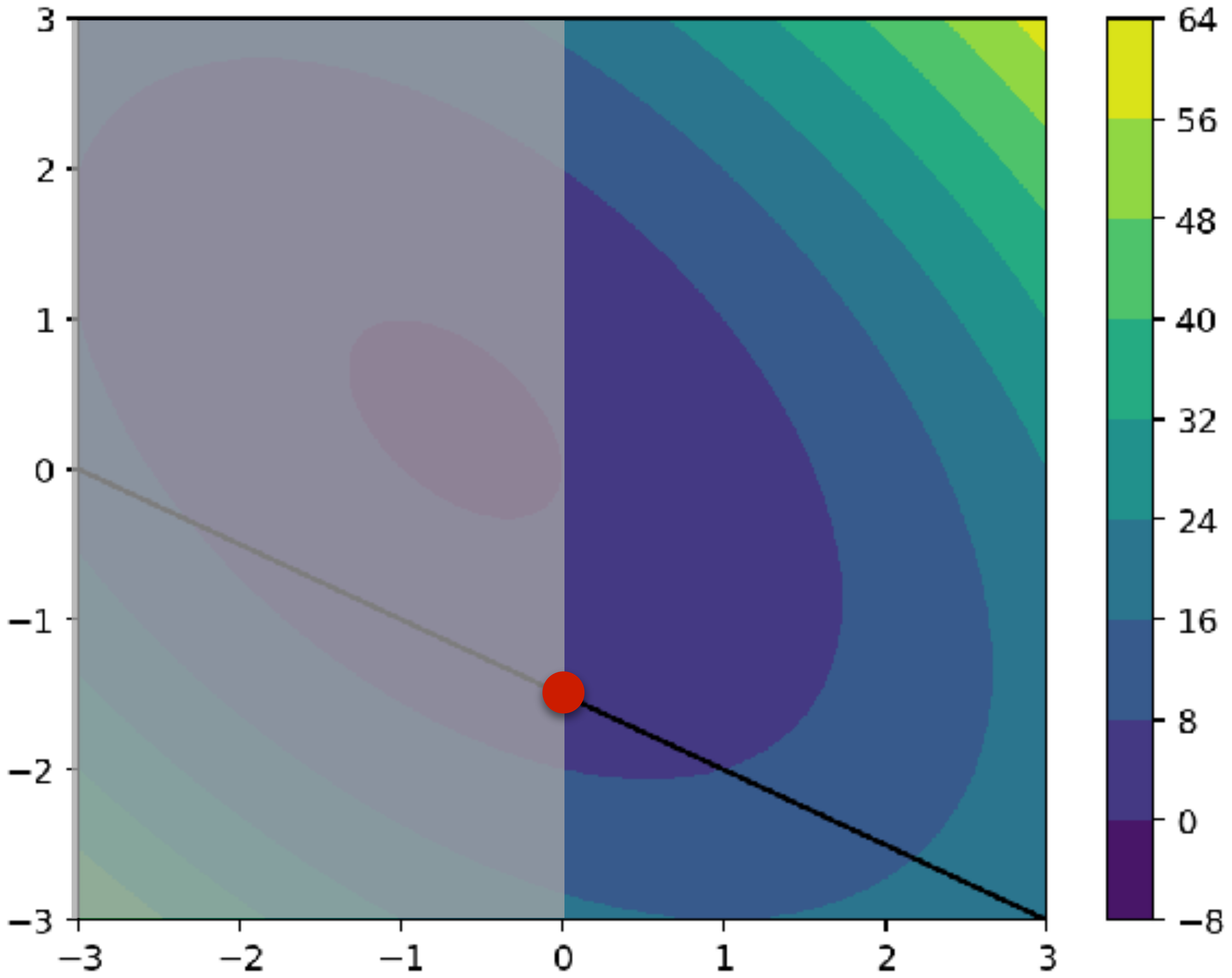
s.t. $x_1 + 2x_2 + 3 = 0$



$$\min_{x_1,x_2} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}^T \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + 2x_1$$

s.t. $x_1 + 2x_2 + 3 = 0$

$x_1 \geq 0$



Minimums (local and global)

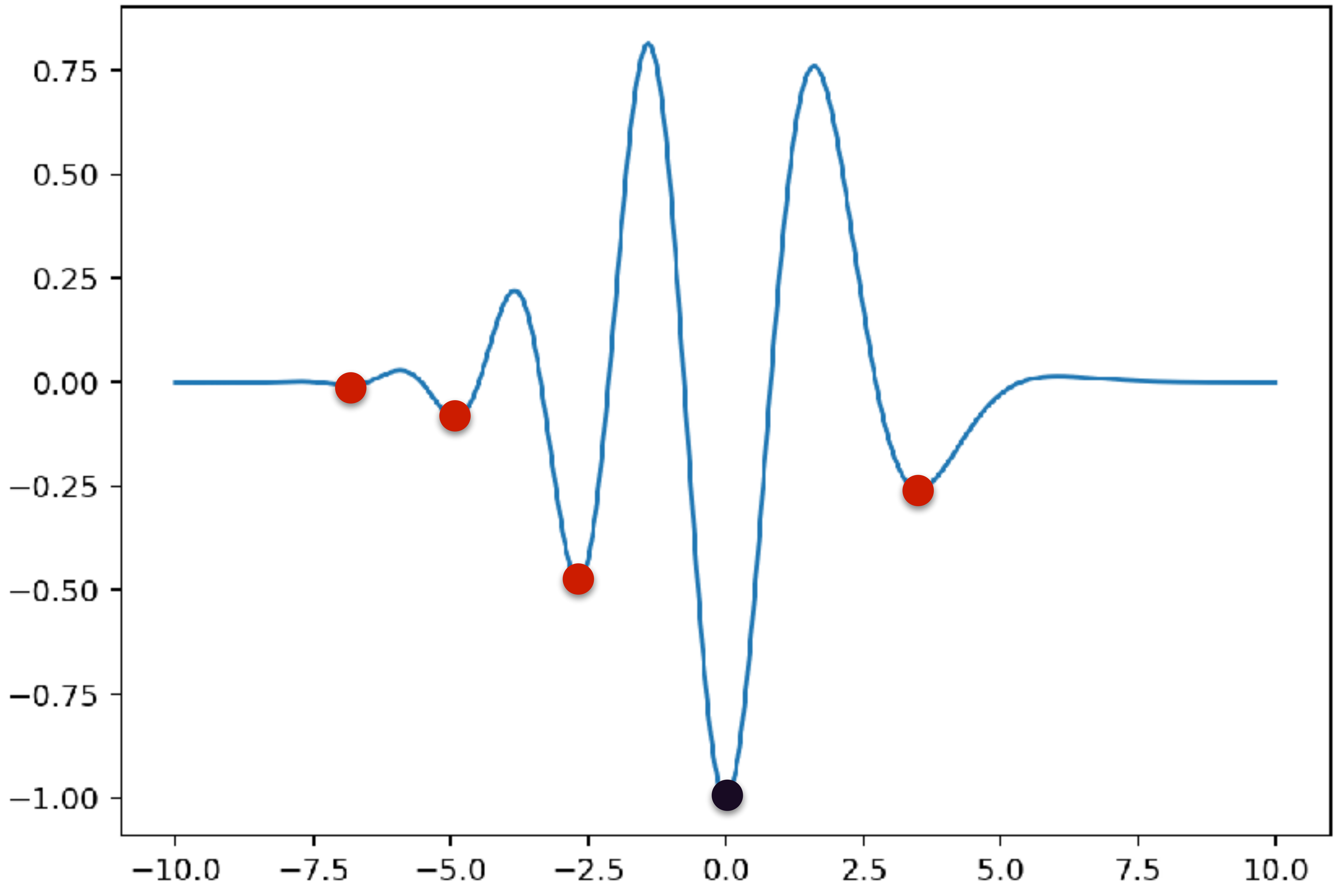
A point x^* is a <u>global minimizer</u> if $f(x^*) \leq f(x)$ for all (admissible) x

A point x^* is a <u>local minimizer</u> if there is a neighborhood $\mathcal{N}$ of x^* such that $f(x^*) \leq f(x)$ for all $x \in \mathcal{N}$

A point x^* is a <u>strict local minimizer</u> (or strong local minimizer) if there is a neighborhood $\mathcal{N}$ of x^* such that $f(x^*) < f(x)$ for all $x \in \mathcal{N}$ with $x \neq x^*$

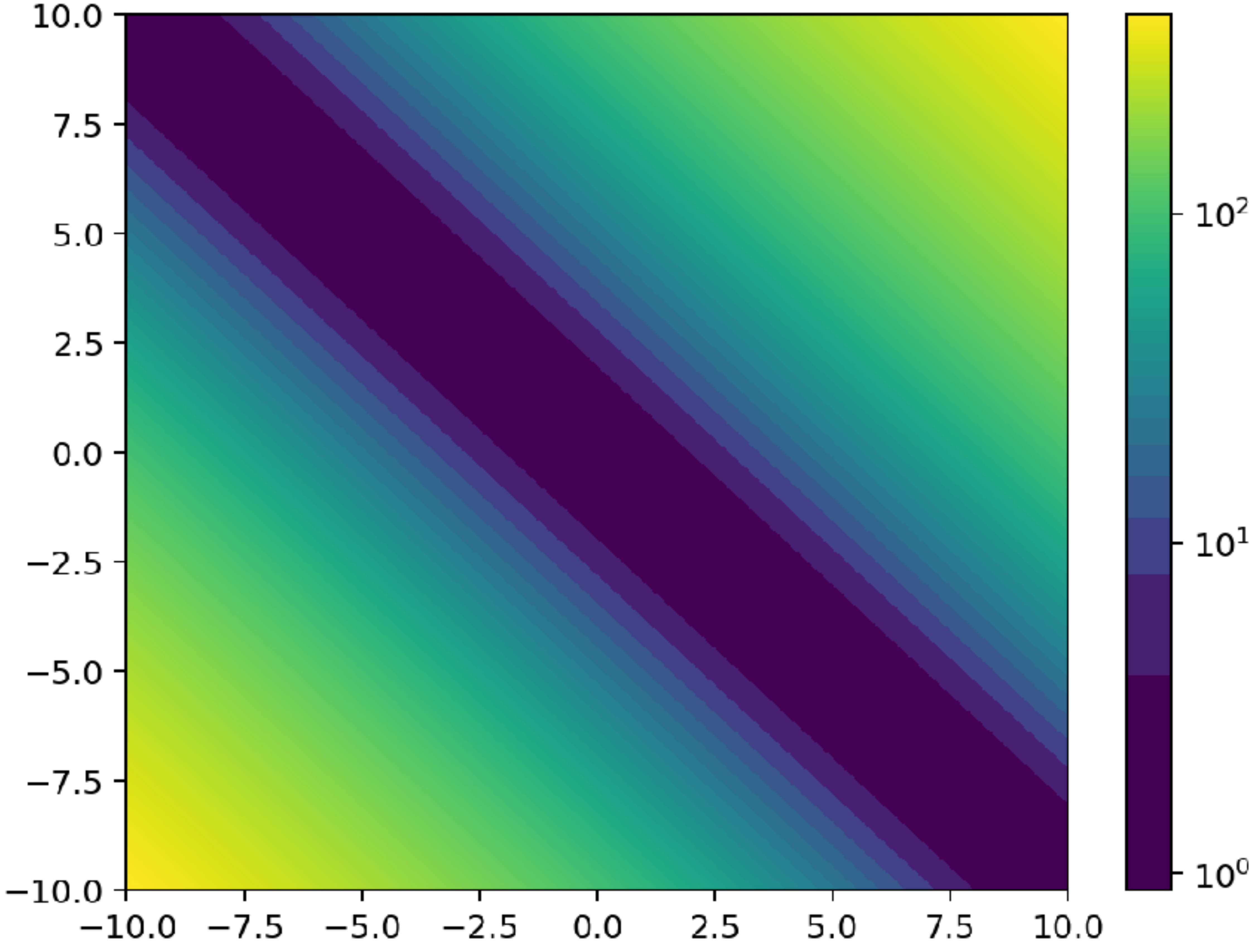
Minimums (local and global)

$$-\cos(2x - 0.1x^2)\mathrm{e}^{-0.1x^2}$$



Minimums (local and global)

$$\min_{x,y} \begin{pmatrix} x \\ y \end{pmatrix}^T \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \begin{pmatrix} x \\ y \end{pmatrix}$$



Any point on the line $x = y$ is a global minimizer and a local minimizer but none of these points are strict local minimizers

Notation

If $f(x) : \mathbb{R}^n \rightarrow \mathbb{R}$ then we write its gradient $\nabla f(x)$ as the vector of derivatives

$$\nabla f(x) = \begin{pmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \\ \vdots \\ \frac{\partial f}{\partial x_n} \end{pmatrix}$$

The matrix of second partial derivatives of f, called the Hessian, is defined as

$$\nabla^2 f(x) = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2^2} & \cdots \frac{\partial^2 f}{\partial x_2 \partial x_n} & \\ \vdots & \vdots & & \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \frac{\partial^2 f}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{bmatrix}$$

Note that the Hessian is symmetric, since $\frac{\partial^2 f}{\partial x_i \partial x_j} = \frac{\partial^2 f}{\partial x_j \partial x_i}$ for all $i, j = 1, \cdots, n$

Notation

When $f(x) : \mathbb{R}^n \to \mathbb{R}^m$ is vector valued, we define $\nabla f(x)$ to be the $n \times m$ matrix whose ith column is $\nabla f_i(x)$, which means that

$$\nabla f(x) = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_2}{\partial x_1} & \cdots & \frac{\partial f_m}{\partial x_1} \\ \frac{\partial f_1}{\partial x_2} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_m}{\partial x_2} \\ \vdots & \vdots & & \\ \frac{\partial f_1}{\partial x_n} & \frac{\partial f_2}{\partial x_n} & \cdots & \frac{\partial f_m}{\partial x_n} \end{bmatrix}$$

Often we might want to work with the transpose of the gradient, a matrix of dimension $m \times n$. This matrix is called the Jacobian is often written $J(x)$. For example, the Jacobian from the forward kinematics function of a robot.

Recognizing a local minimum

$$\min_x f(x)$$

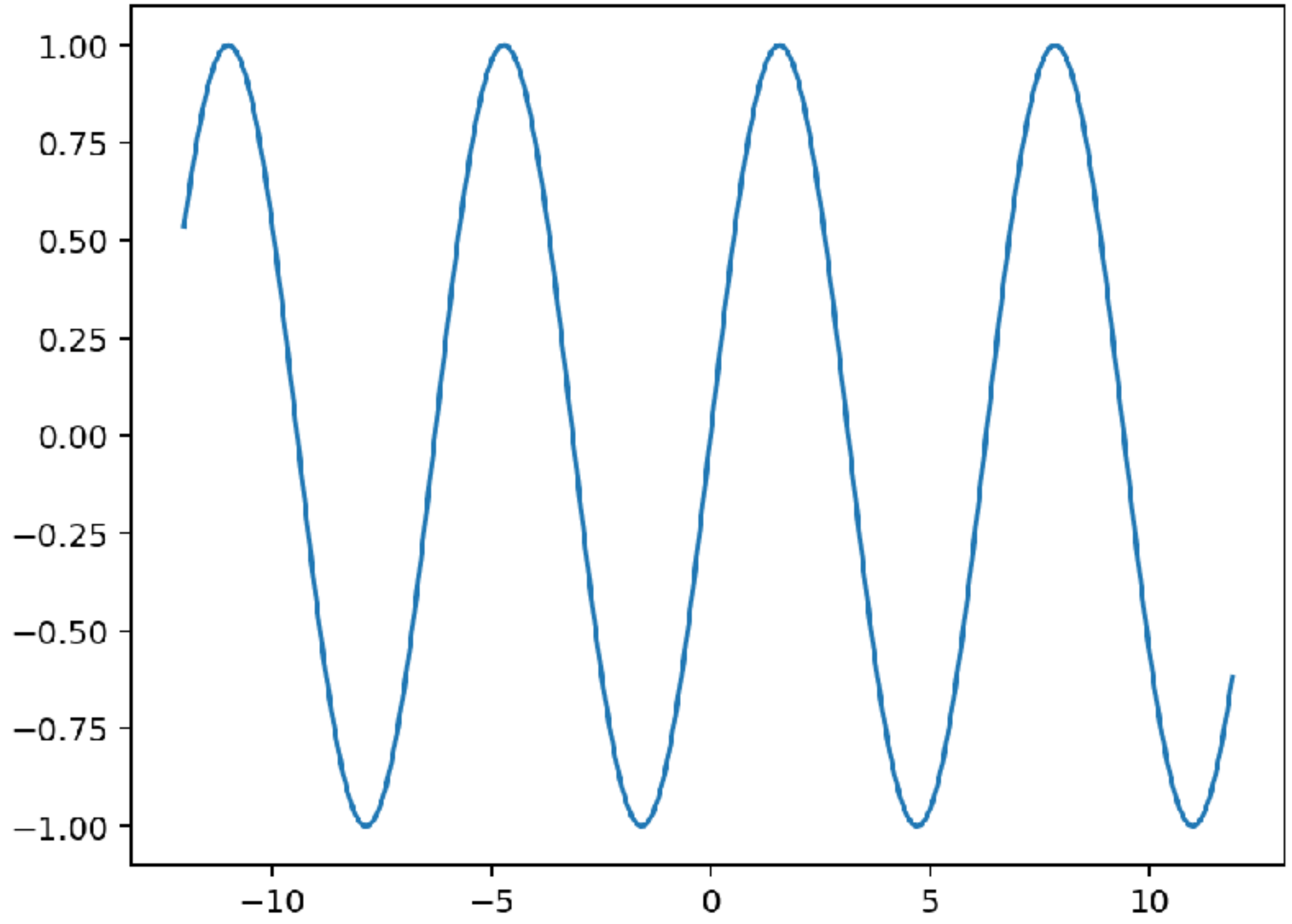
First order necessary conditions: If x^* is a local minimizer and f is continuously differentiable in an open neighborhood of x^*, then $\nabla f(x^*) = 0$

Second order necessary conditions: If x^* is a local minimizer and $\nabla^2 f$ exists and is continuous in an open neighborhood of x^*, then $\nabla f(x^*) = 0$ and $\nabla^2 f(x^*)$ is positive semidefinite

Second order sufficient conditions: Suppose that $\nabla^2 f$ is continuous in an open neighborhood of x^* and that $\nabla f(x^*) = 0$ and $\nabla^2 f(x^*)$ is positive definite. Then x^* is a strict local minimizer of f.

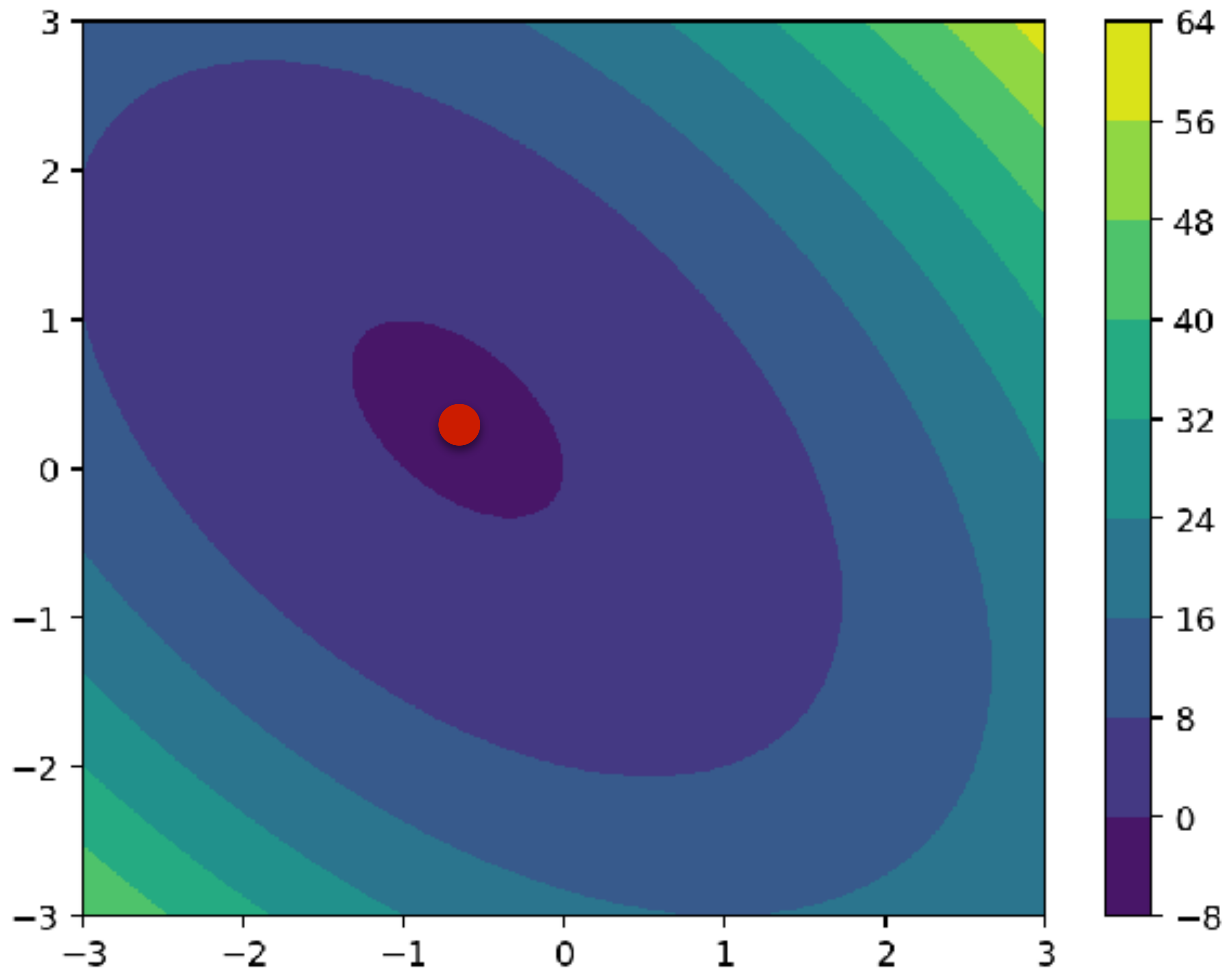
Example

$$\min_x \sin(x)$$



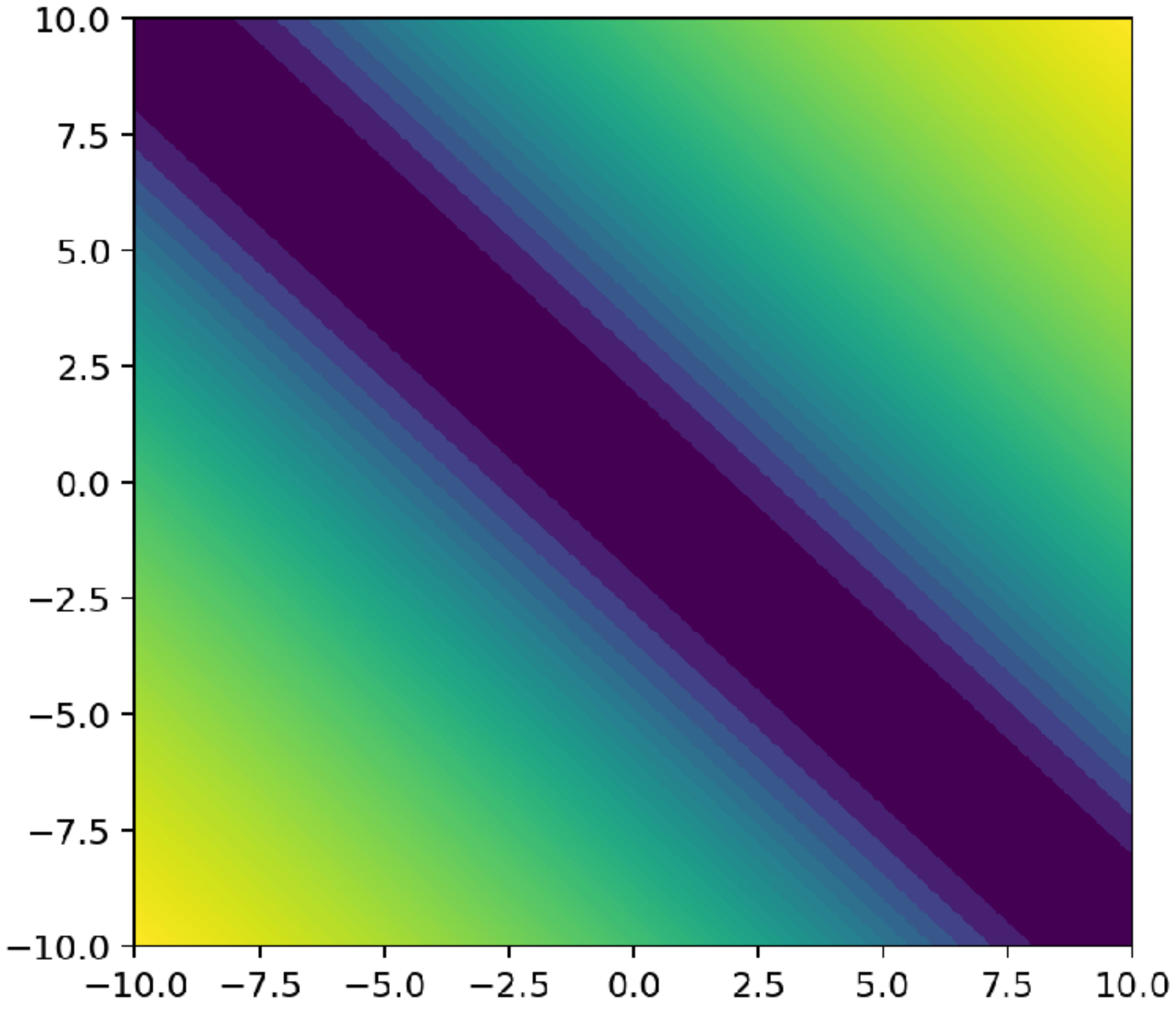
Example

$$\min_{x_1, x_2} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}^T \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + 2x_1$$

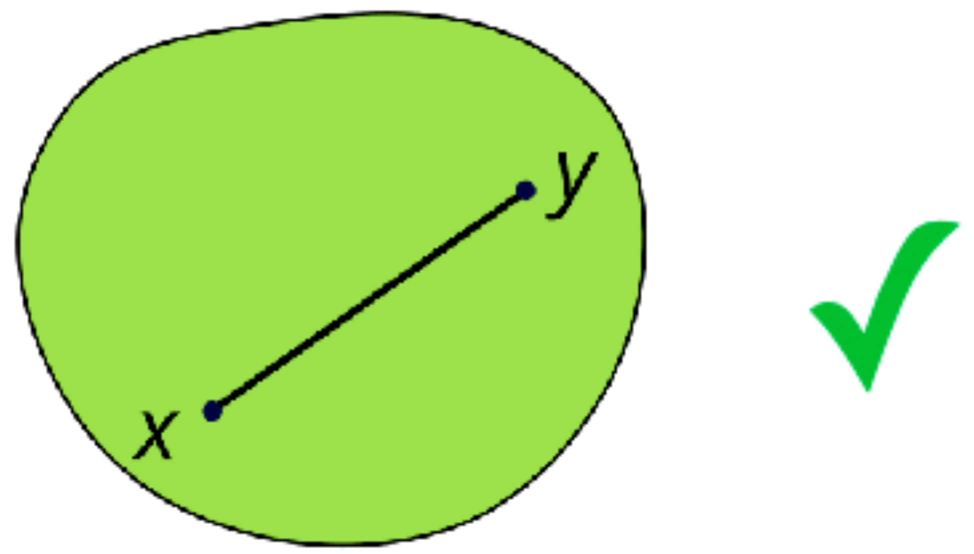


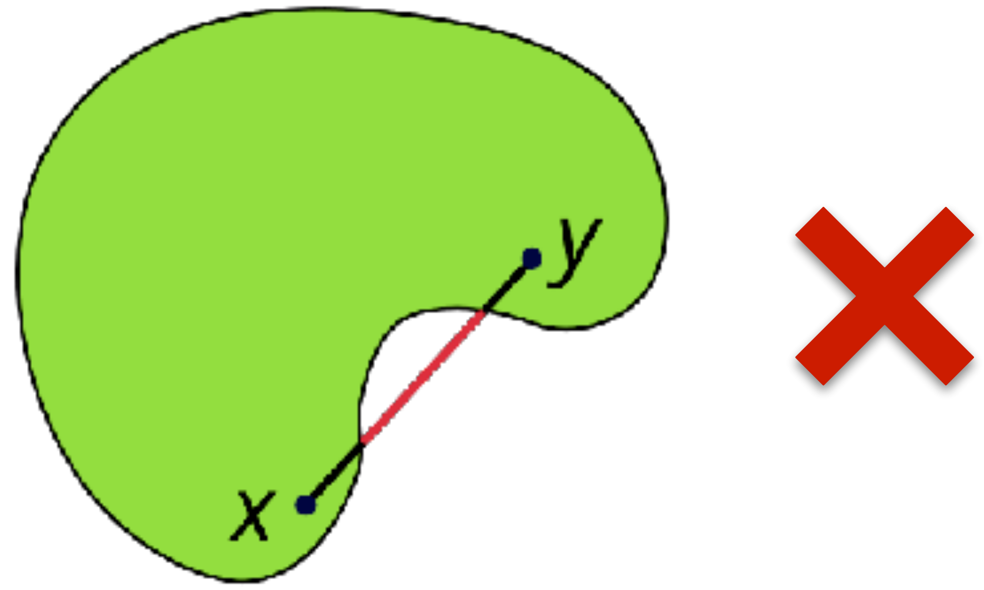
Example

$$\min_{x,y} \begin{pmatrix} x \\ y \end{pmatrix}^T \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \begin{pmatrix} x \\ y \end{pmatrix}$$



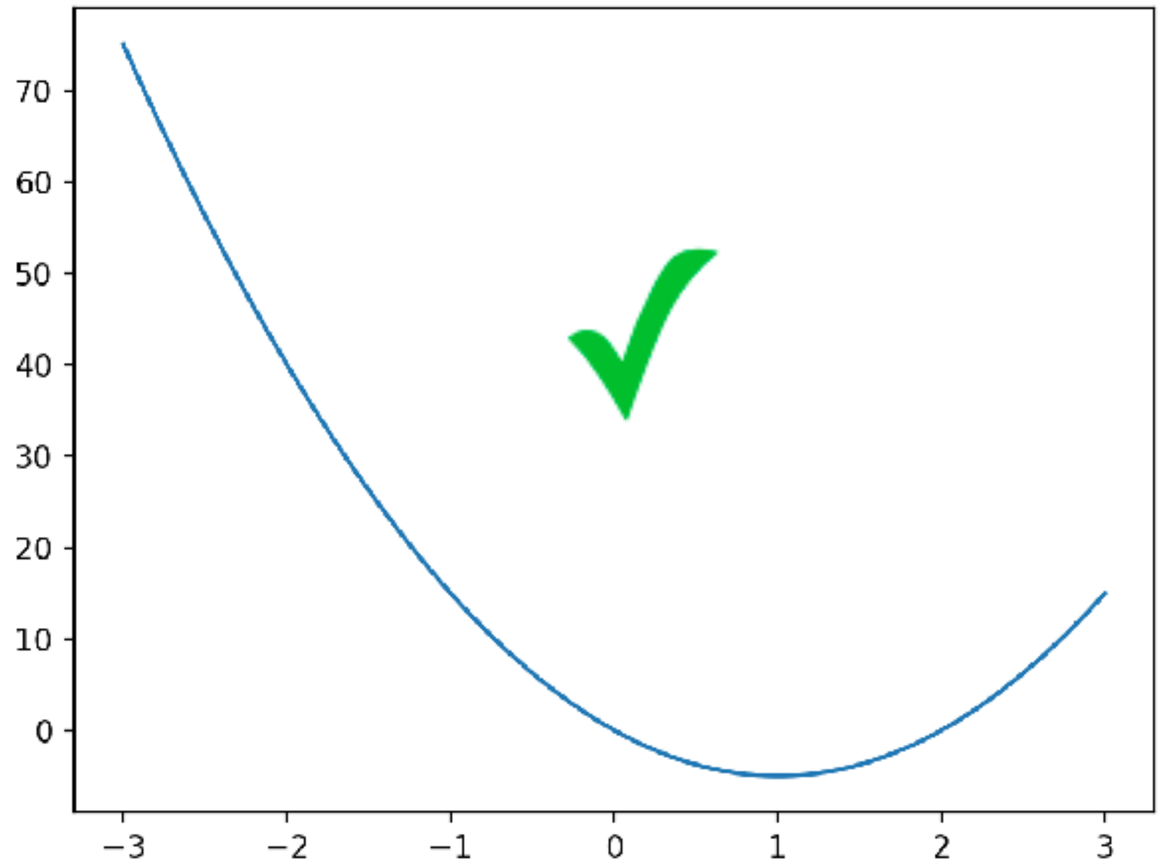
Special case: convex functions

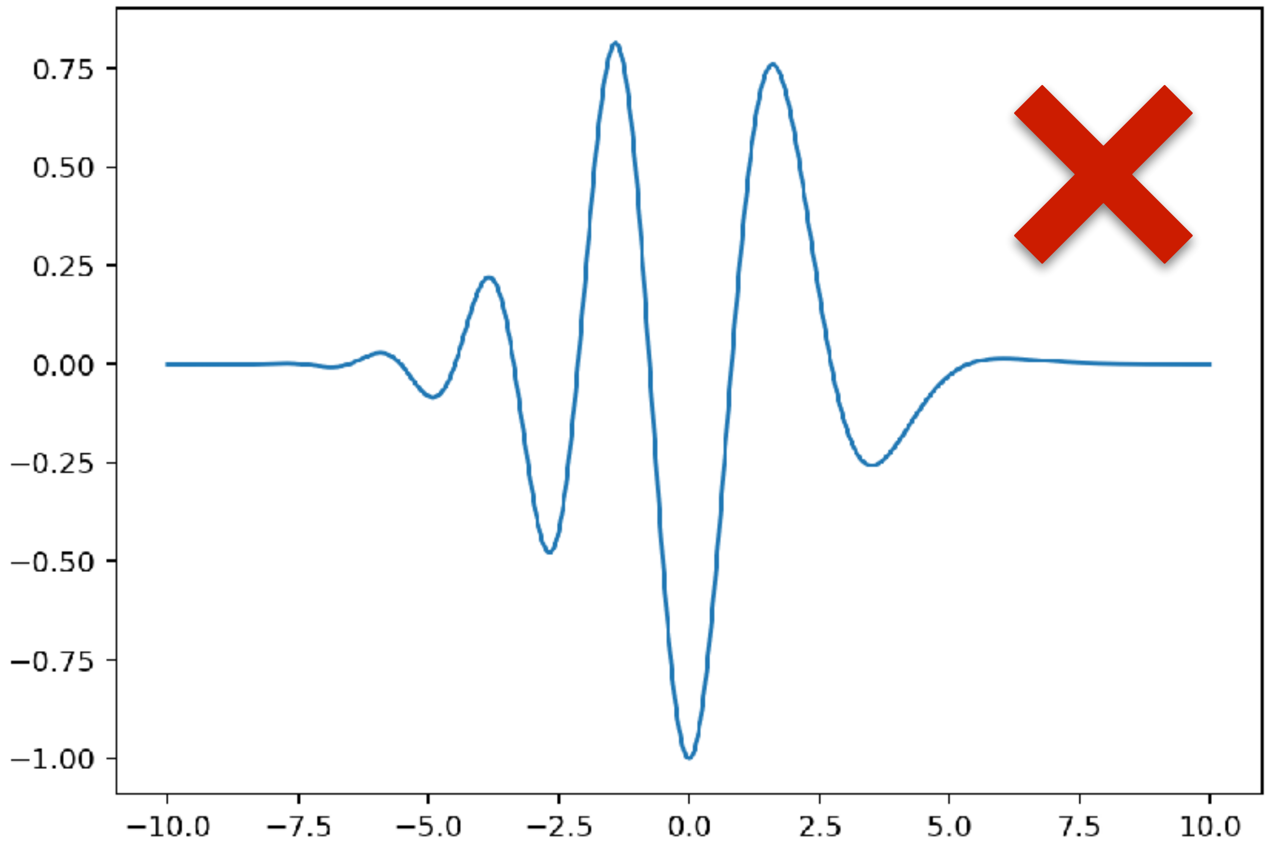




A subset C of a vector space S is convex if for any $x_1, x_2 \in C$ then $tx_1 + (1-t)x_2$ is also in C for any $t \in [0, 1]$. It means that for any two points in C, any point on the segment between these points is also in C.

A scalar function $f(x)$ defined over a convex set $x \in C$ is convex if for any $x_1, x_2 \in C$ we have $f(tx_1 + (1-t)x_2) \leq tf(x_1) + (1-t)f(x_2)$ for any $t \in [0, 1]$.





Special case: convex functions

A differentiable function $f(x)$ defined on a convex domain is convex if and only if $f(x) \geq f(y) + \nabla f(y)^T(x - y)$ for all x and y

A twice differentiable function $f(x)$ defined on a convex domain is convex if and only if its Hessian $\nabla^2 f(x)$ is positive semidefinite on the interior of the convex set

Special case: convex functions

When f is convex, any local minimizer x^* is a global minimizer of f. If in addition f is differentiable, then any point x^* for which $\nabla f(x^*) = 0$ is a global minimizer of f.

Recognizing a minimum in constrained optimization

Recognizing a minimum in constrained optimization

$$\min_x f(x)$$

subject to

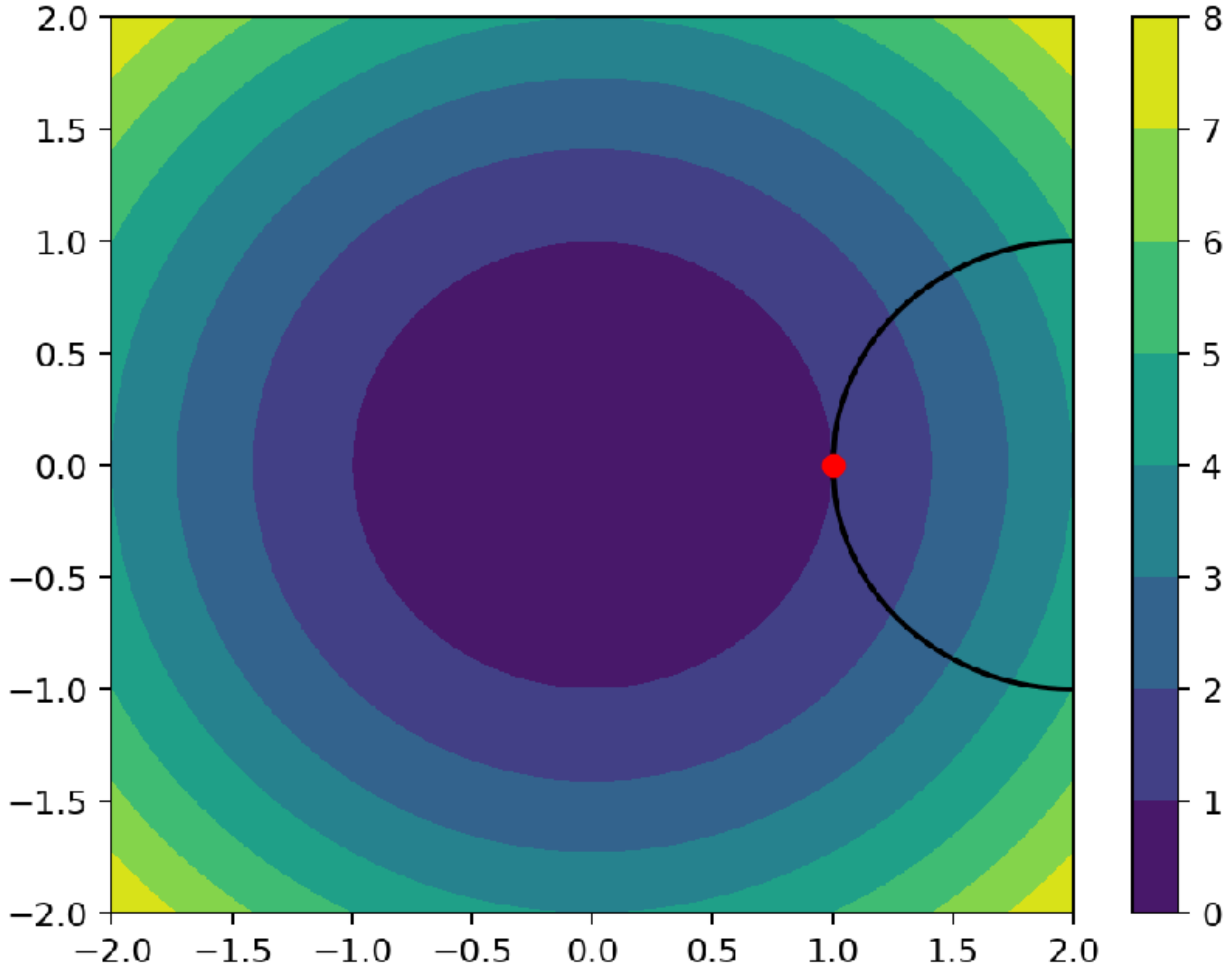
$$g(x) = 0$$

$$h(x) \leq 0$$

Recognizing a minimum in constrained optimization

$$\min_{x_1, x_2} x_1^2 + x_2^2$$

$$\text{subject to} \quad (x_1 - 2)^2 + x_2^2 - 1 = 0$$



Linear independence constraint qualification (LICQ)

$$\min_x f(x) \qquad \text{subject to} \qquad \begin{aligned} g(x) &= 0 \\ h(x) &\leq 0 \end{aligned}$$

The active set $\mathcal{A}(x)$ of any feasible point x consists of the equality constraints and the inequality constraints for which $h_i(x) = 0$

Given a point x and the active set of constraints $\mathcal{A}(x)$ we say that the linear independence constraint qualification (LICQ) holds if the gradients of all the active constraints are linearly independent

Karush Kuhn Tucker conditions of optimality

$$\min_x f(x) \qquad \text{subject to} \qquad \begin{array}{l} g(x) = 0 \\ h(x) \leq 0 \end{array}$$

We define the Lagrangian as $L(x, \lambda, \mu) = f(x) + \lambda^T g(x) + \mu^T h(x)$

The vectors λ and μ are called the Lagrange multipliers

First order necessary conditions (KKT conditions)
Suppose that x^* is a local solution and that the LICQ holds at x^* (and that f, g_i and h_i are continuously differentiable). Then there are Lagrange multiplier vectors λ^* and μ^* such that the following conditions are satisfied

$$\begin{aligned} \nabla_x L(x^*, \lambda^*, \mu^*) &= 0 \\ g(x^*) &= 0 \\ h(x^*) &\leq 0 \\ \mu_i &\geq 0 \ \ \forall i \\ \mu_i h_i(x^*) &= 0 \ \ \forall i \end{aligned}$$

Complementarity condition

$$\mu_i h_i(x^*) = 0 \ \ \forall i$$

Either $\mu_i^* = 0$ or $h_i(x^*) = 0$. It means that a Lagrange multiplier for the inequality cannot be non-zero ”away” from the boundary of the constraint, i.e. only *active contraints* can have non-zero multipliers.

Lagrange multipliers and sensitivity

Optimal control of linear systems with quadratic costs

$$\begin{aligned}
&\min_{x_n, u_n} \sum_{n=0}^{N} x_n^T Q x_n + u_n^T R u_n \\
\text{subject to } \; & x_{n+1} = A x_n + B u_n \\
& x_0 \text{ given}
\end{aligned}$$