Math for Machine Learning

Week 3.1: Basic Differentiation and Vector Calculus

By: Samuel Deng

Logistics & Announcements

Lesson Overview

Motivation for differential calculus. We ultimately want to solve *optimization problems*, which require finding *global minima.*

Single-variable differentiation review. In single-variable differentiation, the ***derivative*** is still a 1×1 "matrix" mapping change in input to change in output.

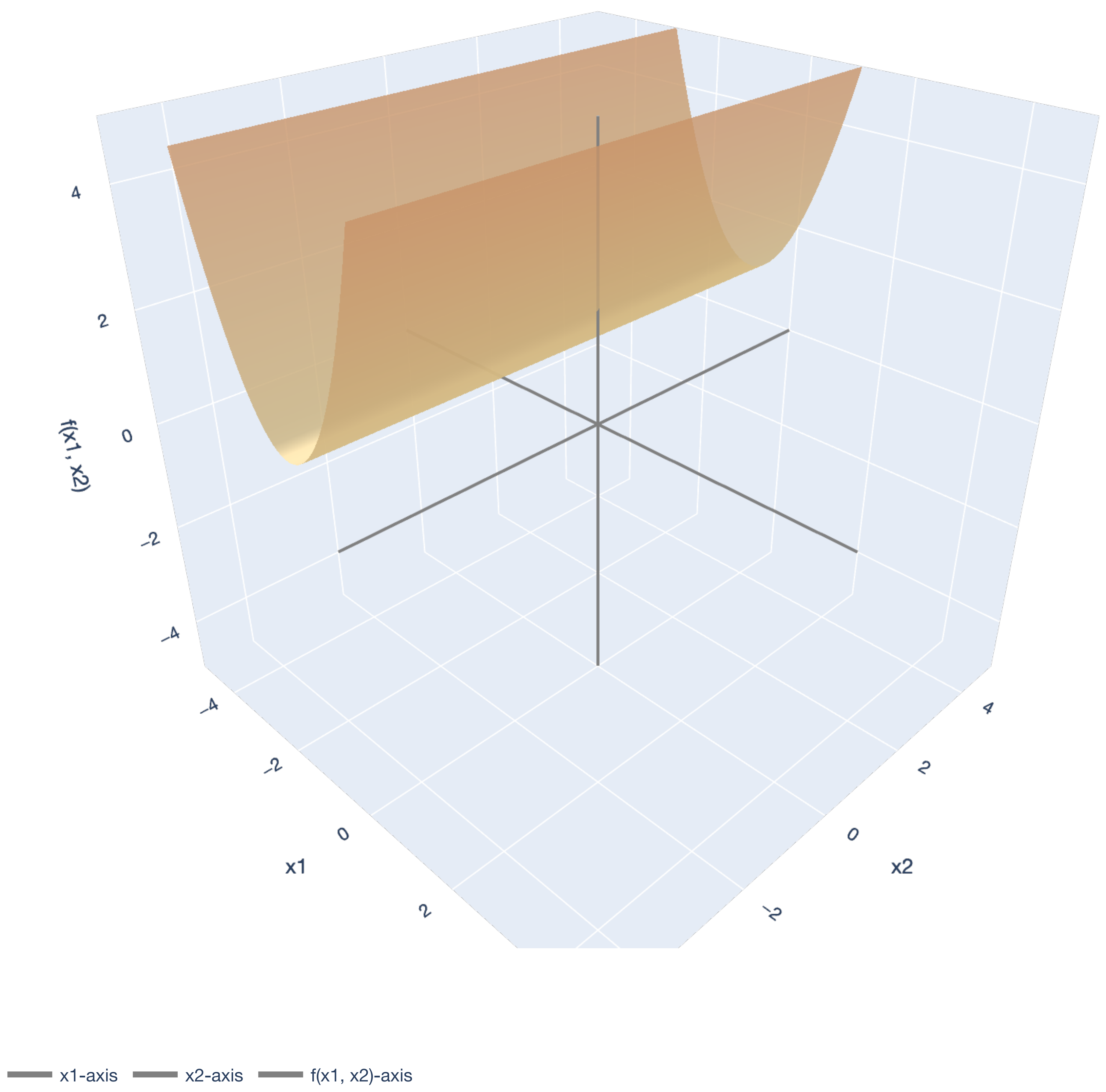
Multivariable differentiation. Derivatives in multiple variables become harder because we can approach from an infinite number of directions, not just two.

Total, directional, and partial derivatives. When a function is ***smooth*** it has a ***total derivative*** (it is ***differentiable***). In this case, the ***directional derivative*** and ***partial derivative*** is comes directly from the total derivative (Jacobian/gradient).

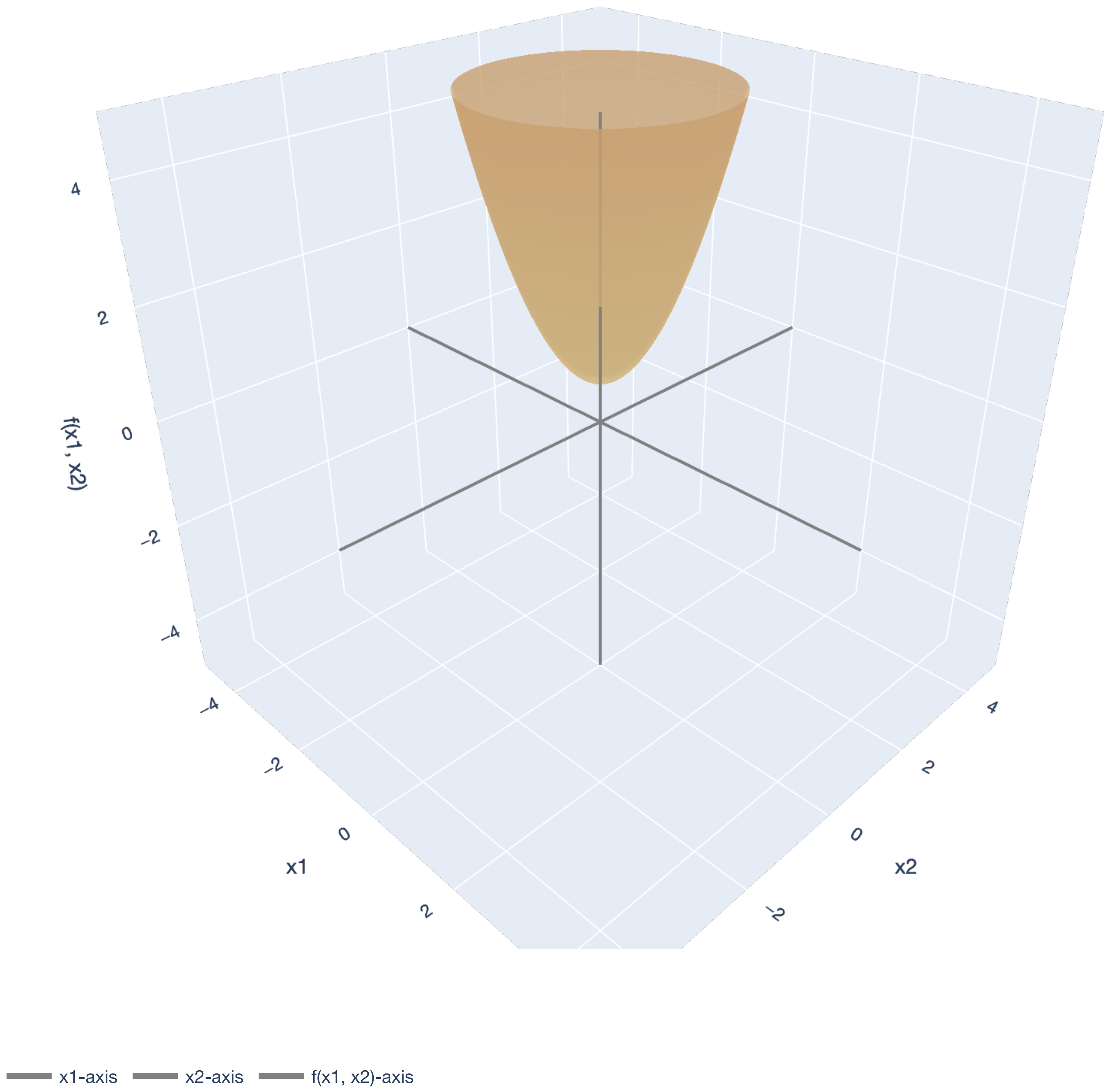
OLS: Optimization Perspective. We can solve OLS using differential calculus instead of linear algebra. We provide a heuristic derivation of the OLS estimator again.

Lesson Overview

Big Picture: Least Squares



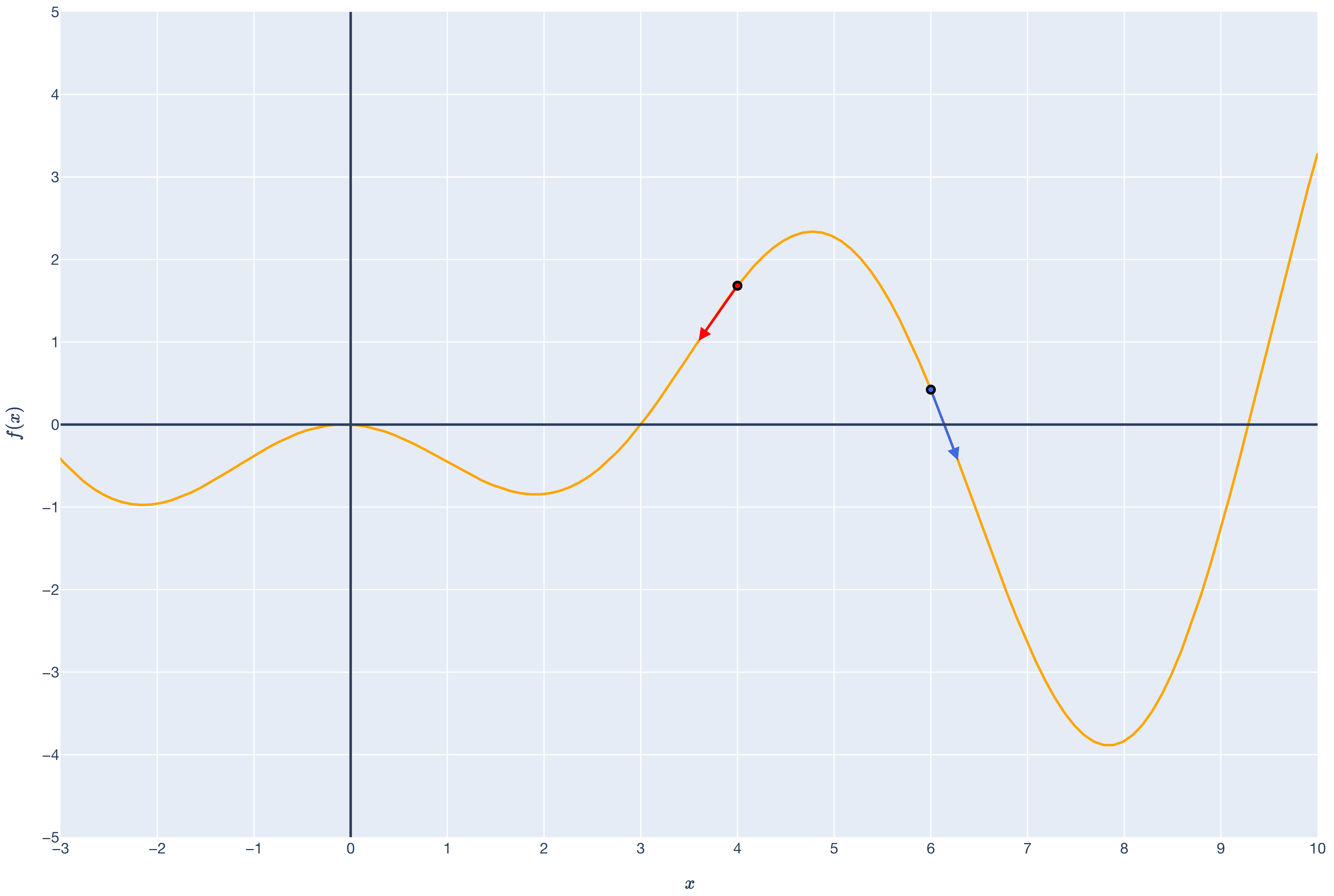
$$\lambda_1, \ldots, \lambda_d \geq 0$$

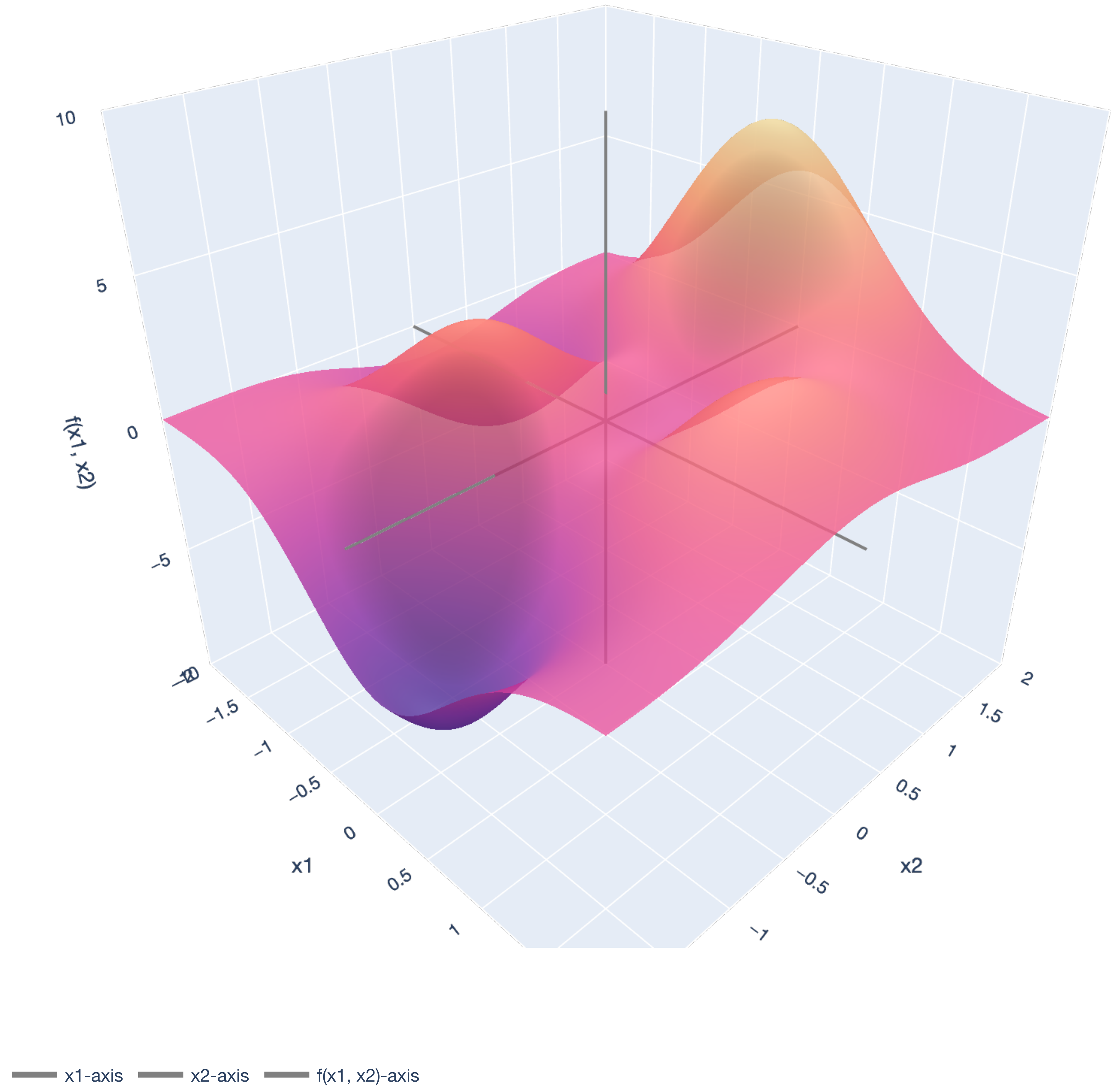


$$\lambda_1, \ldots, \lambda_d > 0$$

Lesson Overview

Big Picture: Gradient Descent





A Motivation for Calculus

Optimization

Motivation

Optimization in single-variable calculus

In much of machine learning, we design algorithms for well-defined *optimization problems.*

In an optimization problem, we want to minimize an ***objective function*** $f : \mathbb{R}^d \to \mathbb{R}$ with respect to a set of constraints $\mathscr{C} \subseteq \mathbb{R}^d$:

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & f(x) \\ \text{subject to} \quad & x \in \mathscr{C} \end{aligned}$$

Motivation

Optimization in single-variable calculus

In much of machine learning, we design algorithms for well-defined *optimization problems.*

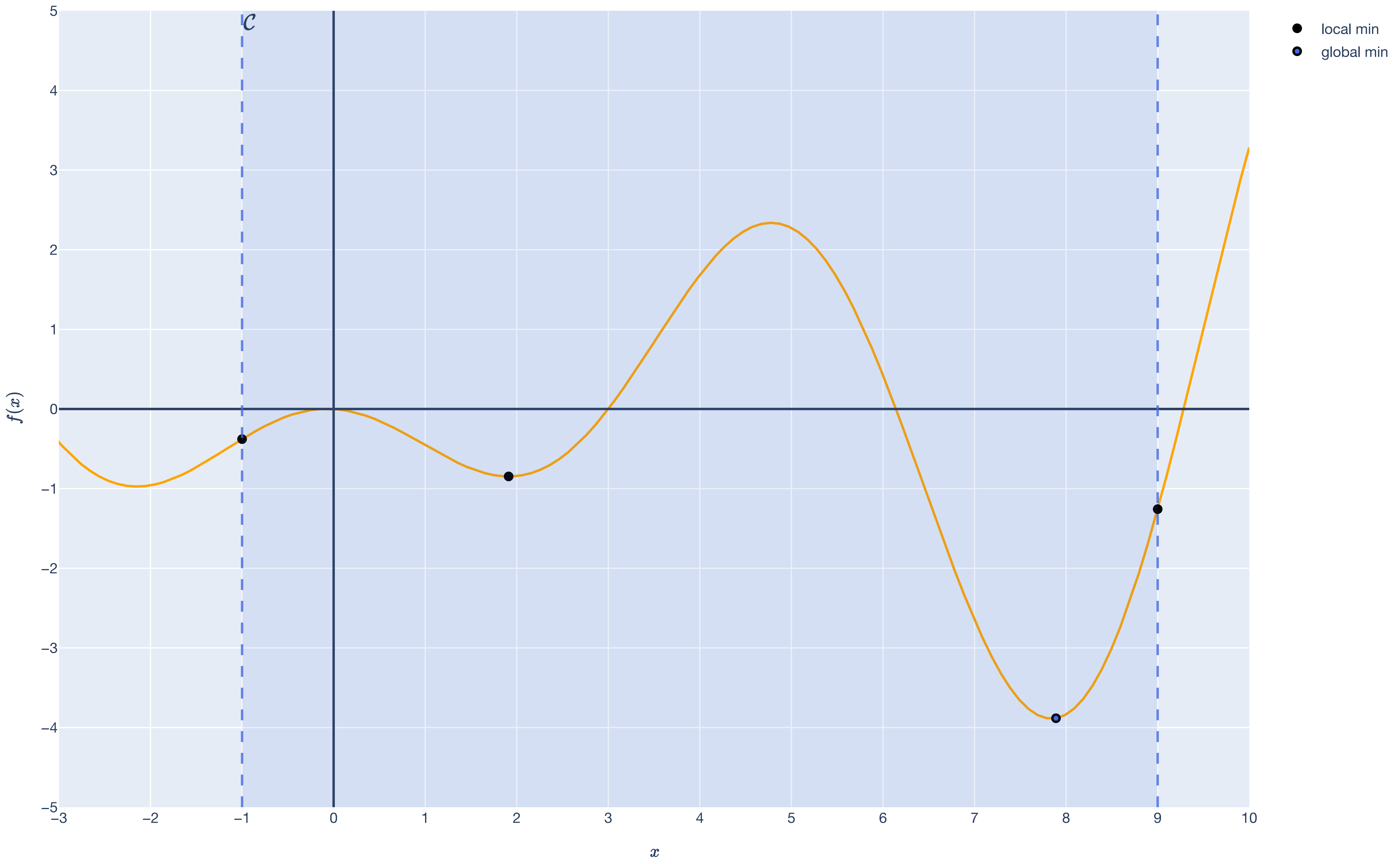
In an optimization problem, we want to minimize an ***objective function*** $f : \mathbb{R}^d \to \mathbb{R}$ with respect to a set of constraints $\mathscr{C} \subseteq \mathbb{R}^d$:

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & f(x) \\ \text{subject to} \quad & x \in \mathscr{C} \end{aligned}$$

How do we know how to do this from single-variable calculus?

Motivation

Optimization in single-variable calculus

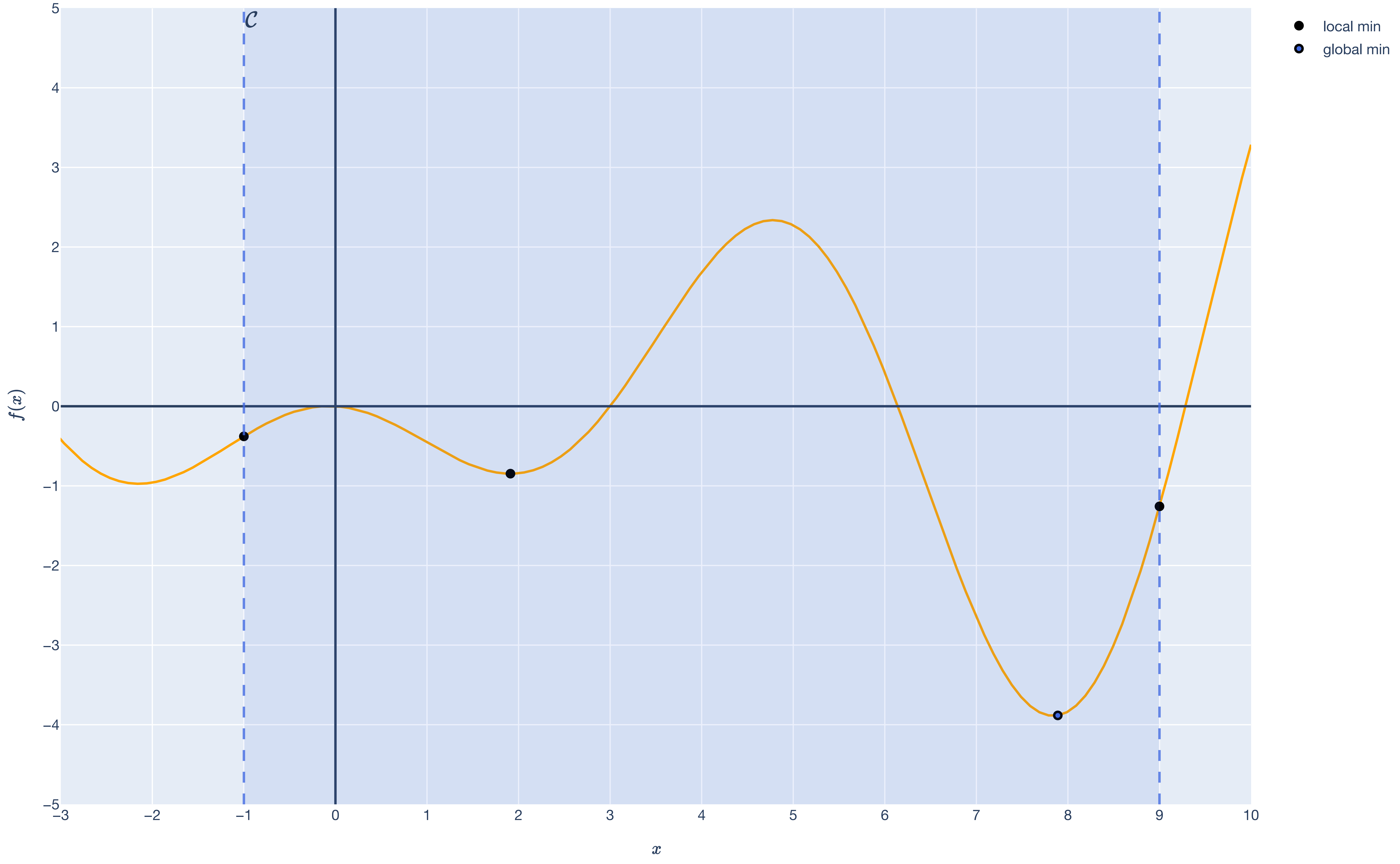


Motivation

Optimization in single-variable calculus

Ultimate goal: Find the *global minimum* of functions.

Intermediary goal: Find the *local minima.*



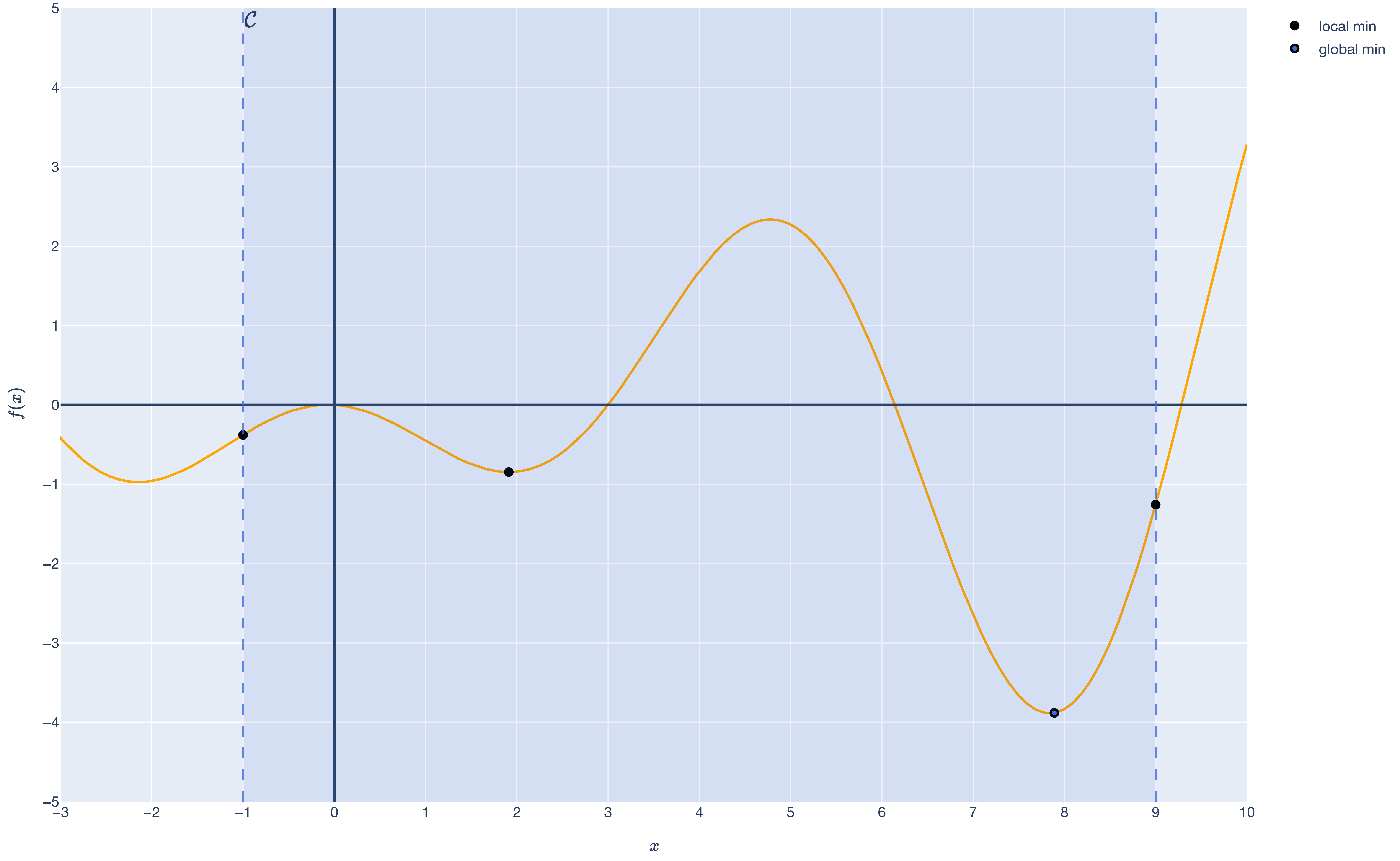
Motivation

Optimization in single-variable calculus

Ultimate goal: Find the *global minimum* of functions.

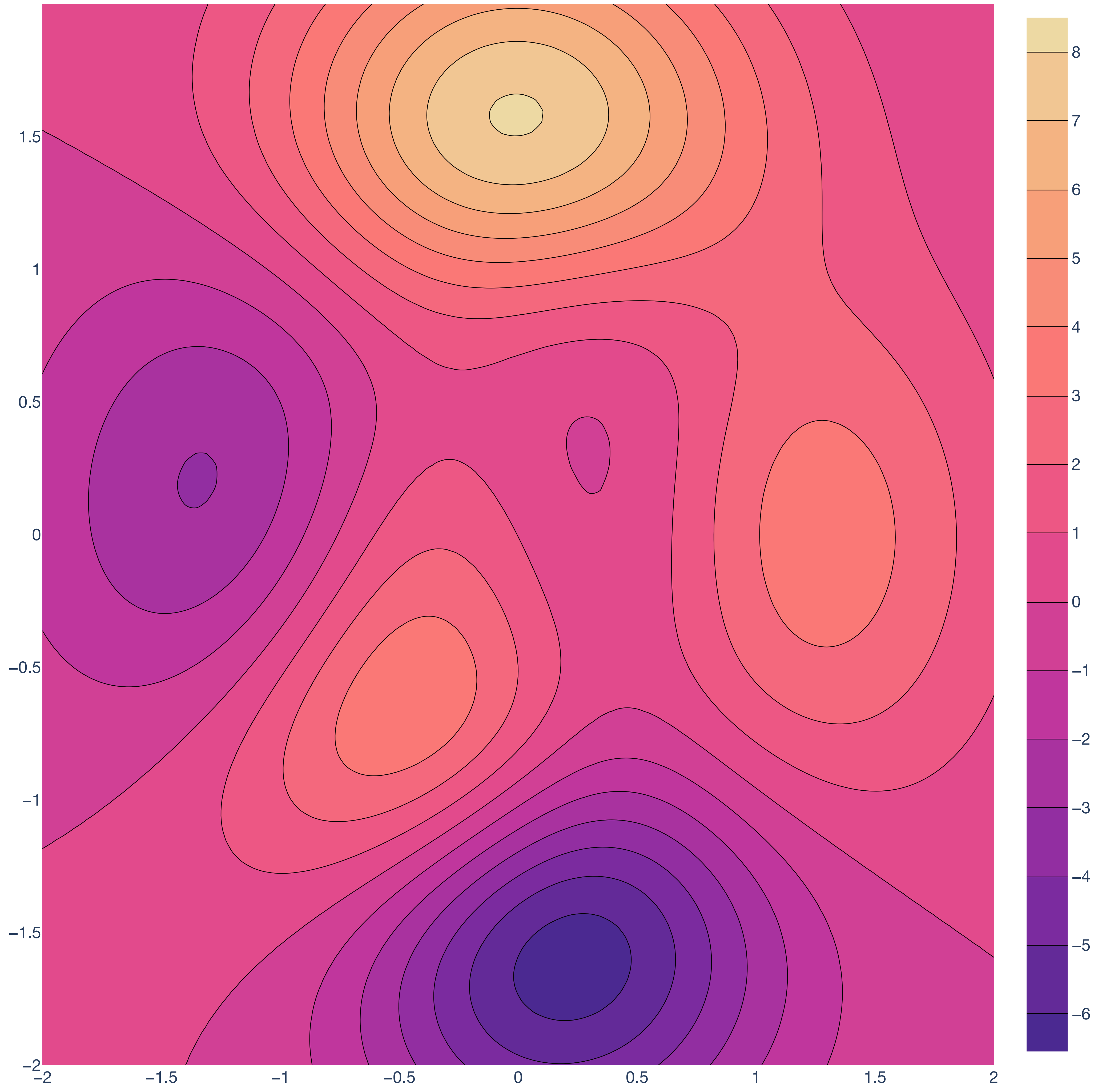
Intermediary goal: Find the *local minima.*

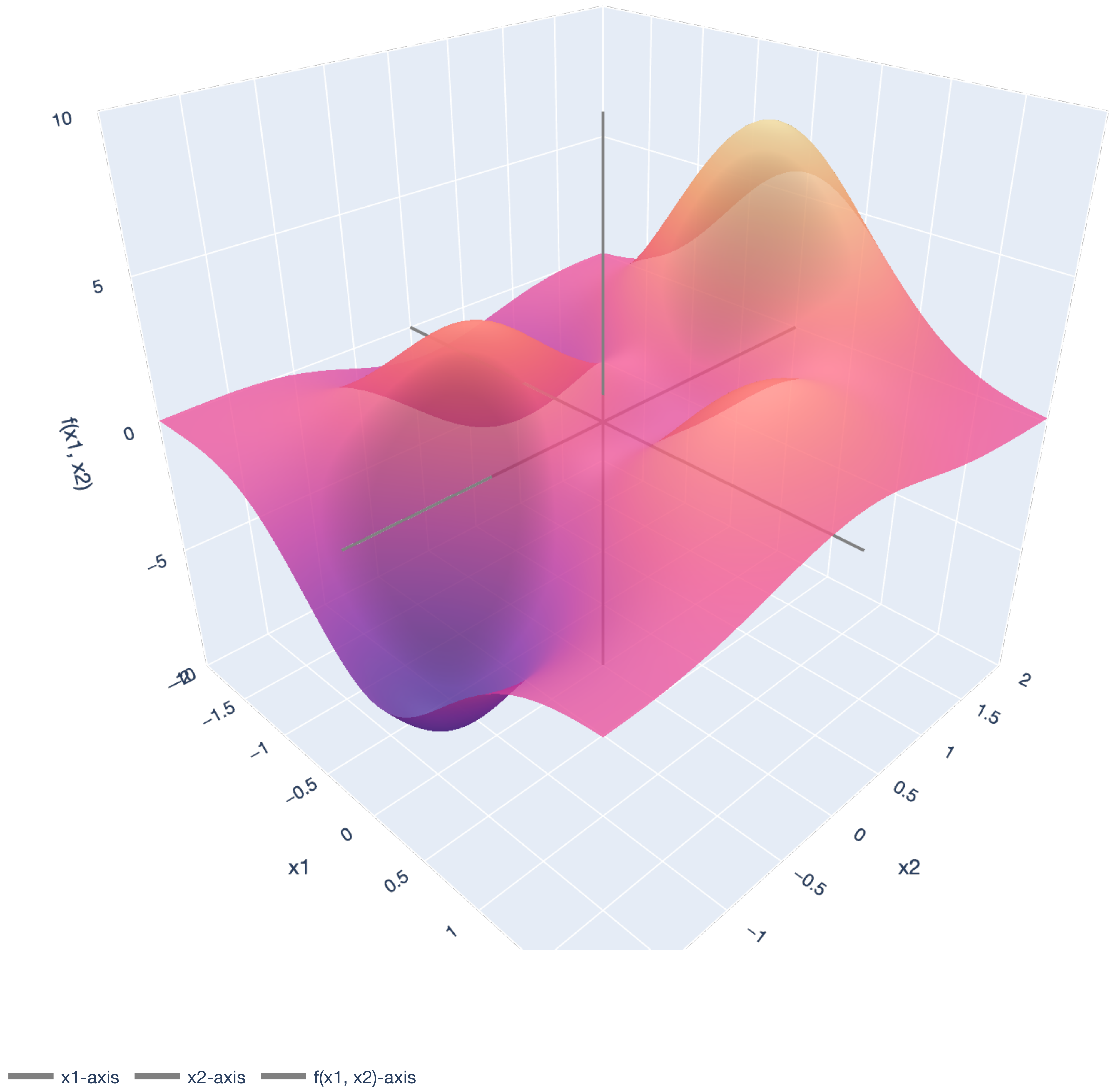
Derivatives give us the direction of steepest descent!



Motivation

Optimization in multi-variable calculus





Single-variable Differentiation

Review of (some) single-variable calculus

Single-variable Differentiation

Difference quotient

For a function $f : \mathbb{R} \to \mathbb{R}$, the ***difference quotient*** computes the slope between two points x and $x + \delta$:

$$\frac{\delta y}{\delta x} := \frac{f(x+\delta) - f(x)}{\delta}$$

Single-variable Differentiation

Difference quotient

For a function $f : \mathbb{R} \to \mathbb{R}$, the ***difference quotient*** computes the slope between two points x and $x + \delta$:

$$\frac{\delta y}{\delta x} := \frac{f(x+\delta) - f(x)}{\delta}$$

Throughout, δ denotes "change in the inputs." For any two points $x, y \in \mathbb{R}$, we can write $\delta = y - x$.

For a linear function, this is the slope *everywhere.*

Single-variable Differentiation

Difference quotient

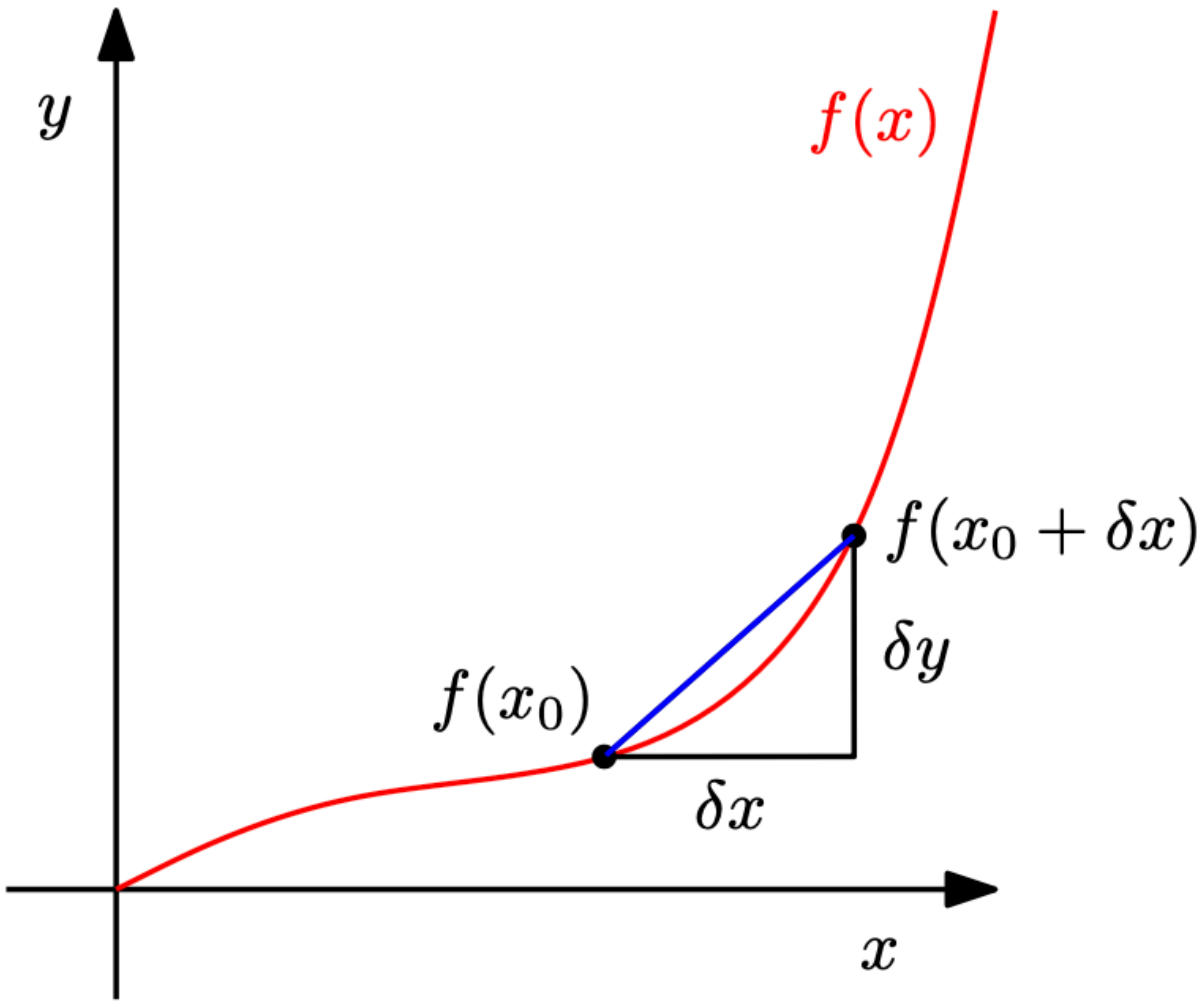
Example. $f(x) = -2x$

Example. $f(x) = x^2 - 2x + 1$

Single-variable Differentiation

$f : \mathbb{R} \to \mathbb{R}$

$$\frac{\delta y}{\delta x} := \frac{f(x + \delta x) - f(x)}{\delta x}$$



Single-variable Differentiation

Definition of the derivative

For a function $f : \mathbb{R} \to \mathbb{R}$, the ***derivative*** of f at the point x is the value

$$\frac{df}{dx} := \lim_{\delta \to 0} \frac{\delta x}{\delta y} = \lim_{\delta \to 0} \frac{f(x + \delta) - f(x)}{\delta},$$

if the limit exists.

In this lecture, we will assume that all functions are *everywhere differentiable.* Not always the case, e.g. $f(x) = \quad x \quad$.

We will also denote this as $f'(x)$ or $\nabla f(x)$.

Important: The derivative is defined *at a point*!

Single-variable Differentiation

Definition of the derivative

For a function $f : \mathbb{R} \to \mathbb{R}$, the ***derivative*** of f at the point x is the value

$$\frac{df}{dx} := \lim_{\delta \to 0} \frac{\delta x}{\delta y} = \lim_{\delta \to 0} \frac{f(x+\delta) - f(x)}{\delta},$$

if the limit exists.

Single-variable Differentiation

Definition of the derivative

For a function $f : \mathbb{R} \to \mathbb{R}$, the ***derivative*** of f at the point x is the value

$$\frac{df}{dx} := \lim_{\delta \to 0} \frac{\delta x}{\delta y} = \lim_{\delta \to 0} \frac{f(x + \delta) - f(x)}{\delta},$$

if the limit exists.

In this lecture, we will assume that all functions are *everywhere differentiable.* Not always the case, e.g. $f(x) = \quad x \quad$.

We will also denote this as $f'(x)$ or $\nabla f(x)$.

Important: The derivative is defined *at a point*!

Single-variable Differentiation

Definition of the derivative

Example. $f(x) = -2x$

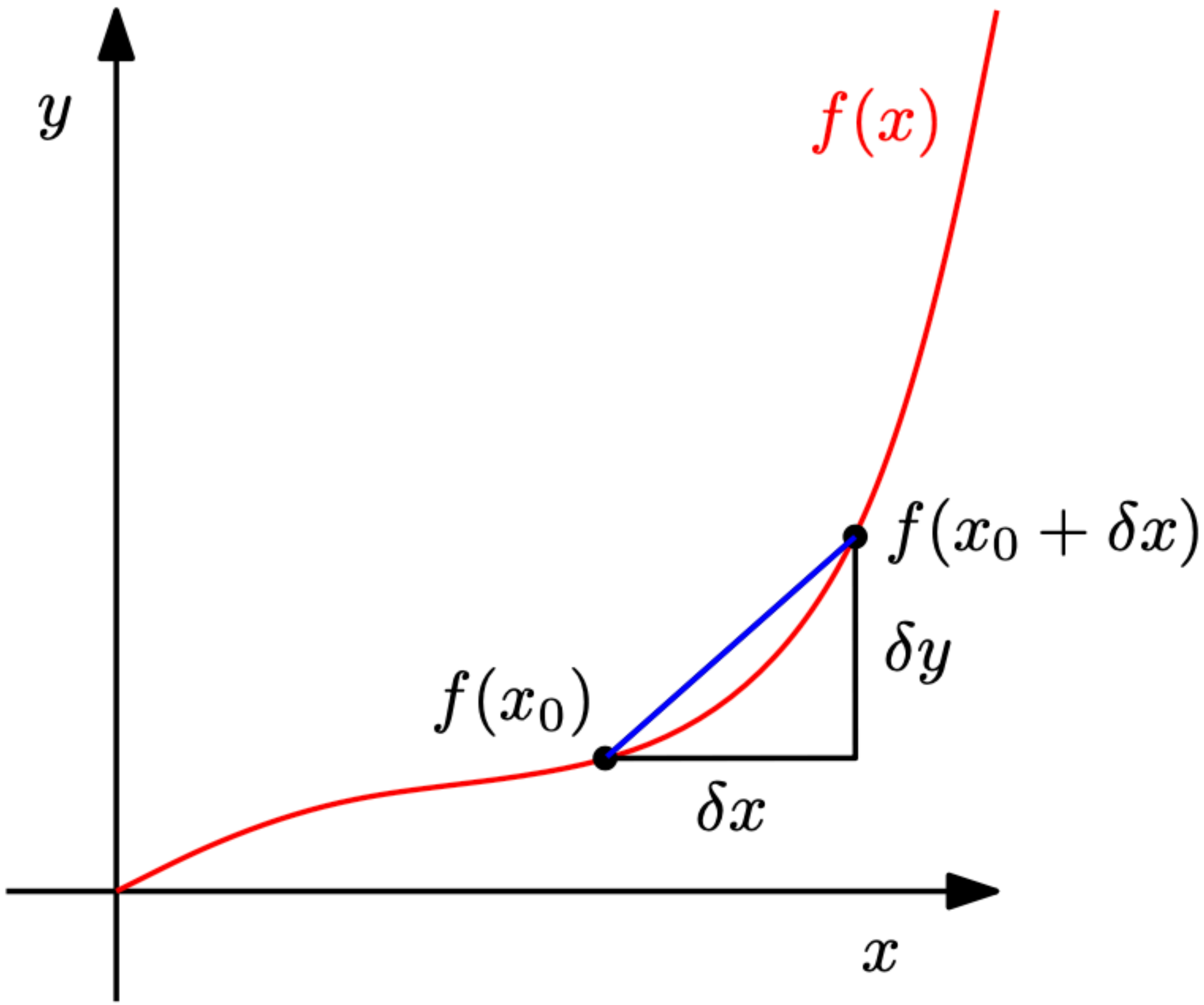
Example. $f(x) = x^2 - 2x + 1$

Single-variable Differentiation

$f : \mathbb{R} \to \mathbb{R}$

Get used to thinking, for all x that are "close" to x_0:

$$\nabla f(x_0)(x - x_0) \approx f(x) - f(x_0)$$



The derivative gives a good local, linear approximation to the change in $f(x)$.

Single-variable Differentiation

$f : \mathbb{R} \to \mathbb{R}$

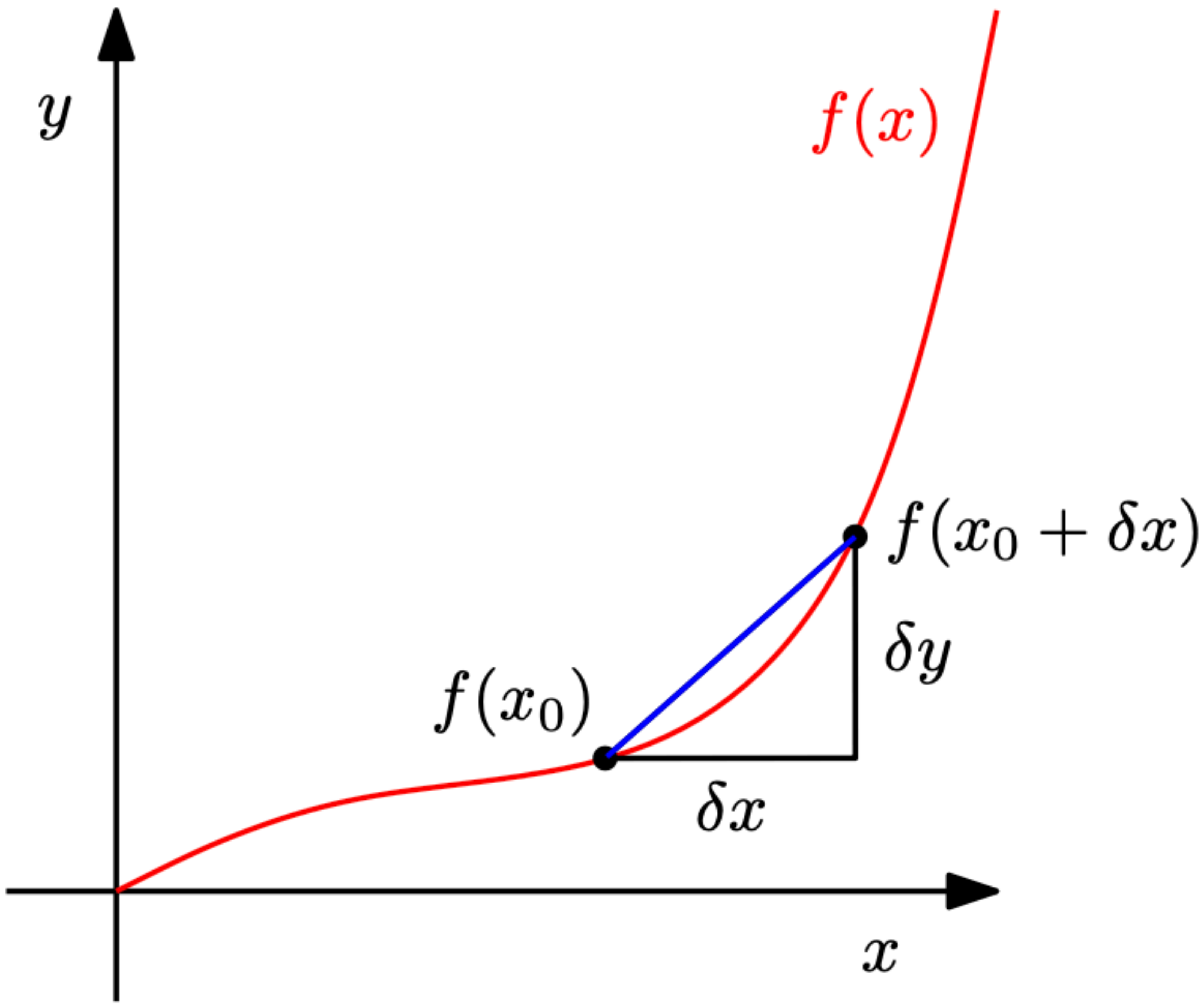
Get used to thinking, for all x that are "close" to x_0:

$$\nabla f(x_0)(x - x_0) \approx f(x) - f(x_0)$$

We can always write the "target point" as $x = x_0 + \delta$.

$$\nabla f(x_0) \cdot \delta \approx f(x_0 + \delta) - f(x_0)$$

The derivative gives a good local, linear approximation to the change in $f(x)$.



Single-variable Differentiation

Review: basic derivative rules

Product rule:

$$\nabla\left(f(x)g(x)\right) = g(x)\,\nabla f(x) + f(x)\,\nabla g(x)$$

Quotient rule:

$$\nabla\left(\frac{f(x)}{g(x)}\right) = \frac{g(x)\,\nabla f(x) - f(x)\,\nabla g(x)}{g(x)^2}$$

Sum rule:

$$\nabla\left(f(x) + g(x)\right) = \nabla f(x) + \nabla g(x)$$

Chain rule:

$$\nabla(g(f(x)) = \nabla(g \circ f)(x) = \nabla g(f(x))\,\nabla f(x)$$

Linearity

Review from linear algebra

Linearity is the central property in linear algebra. Cooking is linear.

Bacon, egg, cheese (on roll)	*Bacon, egg, cheese (on bagel)*	*Lox sandwich*
1 egg	1 egg	0 egg
1 slice of cheese	1 slice of cheese	0 slice of cheese
1 slice bacon	1 slice bacon	0 slice bacon
1 Kaiser roll	0 Kaiser roll	0 Kaiser roll
0 cream cheese	0 cream cheese	1 cream cheese
0 slices of lox	0 slices of lox	2 slices of lox
0 bagel	1 bagel	1 bagel

Linearity

Review from linear algebra

Linearity is the central property in linear algebra. A function ("transformation") $T : \mathbb{R}^d \to \mathbb{R}^n$ is ***linear*** if T satisfies these two properties for any two vectors $\mathbf{a}, \mathbf{b} \in \mathbb{R}^d$:

$$T(\mathbf{a} + \mathbf{b}) = T(\mathbf{a}) + T(\mathbf{b})$$

$$T(c\mathbf{a}) = cT(\mathbf{a}) \text{ for any } c \in \mathbb{R}.$$

Linearity

Review from linear algebra

Linearity is the central property in linear algebra. A function ("transformation") $T : \mathbb{R} \to \mathbb{R}$ is ***linear*** if T satisfies these two properties for any two vectors $a, b \in \mathbb{R}$:

$$T(a + b) = T(a) + T(b)$$

$$T(ca) = cT(a) \text{ for any } c \in \mathbb{R}.$$

Single-variable Differentiation

Linearity and differentiation

Why do we like linear transformations?

$$\nabla f(x_0)(x - x_0) \approx f(x) - f(x_0)$$

Recall: $T(x + y) = T(x) + T(y)$ and $T(cx) = cT(x)$.

Derivative exploits the fact that, on small scales, things behave linearly!

Single-variable Differentiation

Linearity and differentiation

The derivative is a linear transformation that maps changes in x to changes in y. We like linear transformations!

$$T : \text{change in x} \rightarrow \text{change in y}$$

$$\nabla f(x_0)(x - x_0) \approx f(x) - f(x_0)$$

Single-variable Differentiation

Linearity and differentiation

The derivative is a linear transformation that maps changes in x to changes in y. We like linear transformations!

$$T : \text{change in x} \rightarrow \text{change in y}$$

$$\nabla f(x_0)(x - x_0) \approx f(x) - f(x_0)$$

Consider the function $f(x) = x^2$. The derivative of f at $x = 1$ is $\nabla f(1) = 2$.

The derivative is nothing more than a 1×1 matrix in single-variable differentiation: $\nabla f(1) = [2]$.

A goal of differential calculus, for us, is to replace nonlinear functions with linear approximations!

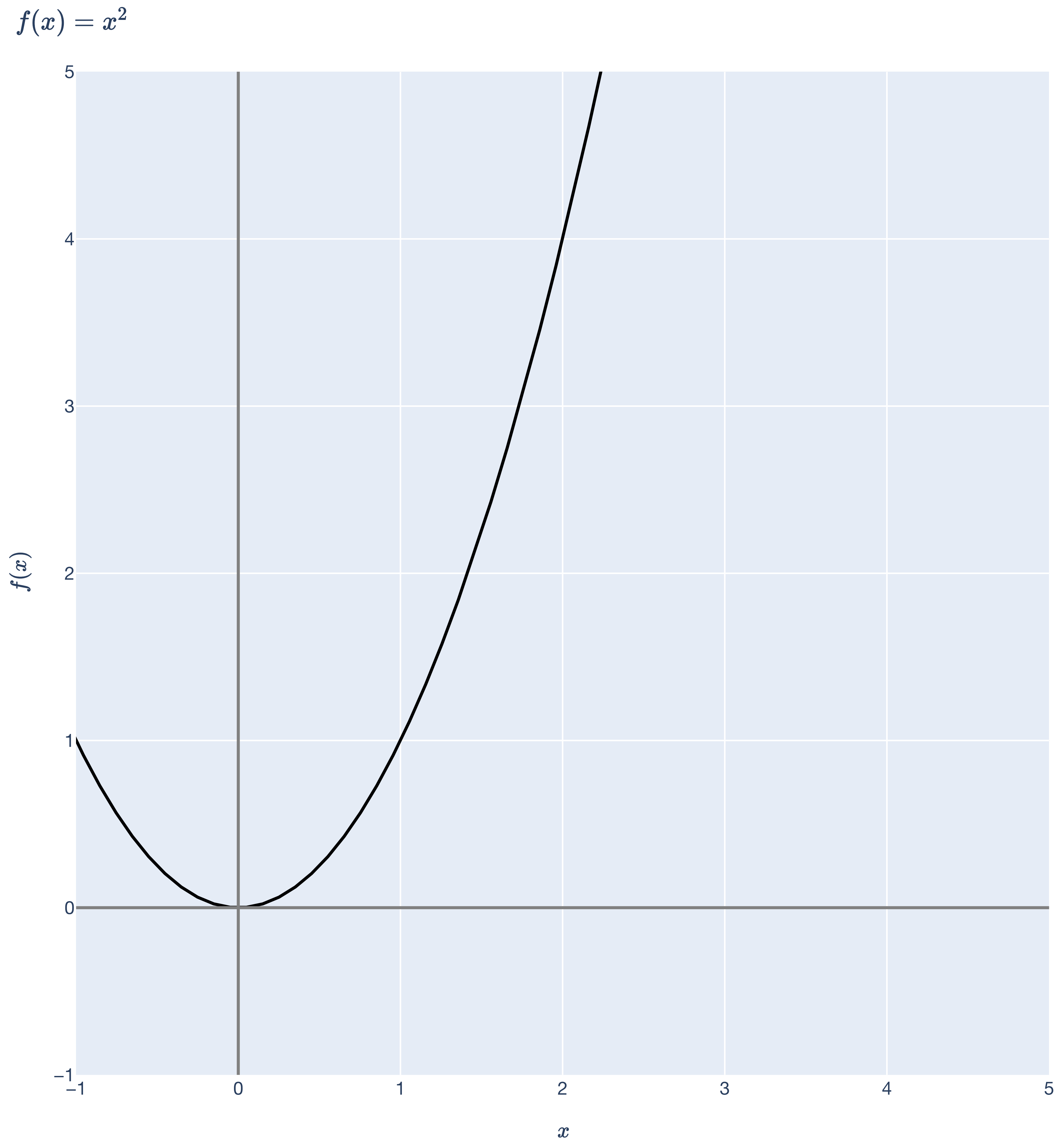
Single-variable Differentiation

Linearity and differentiation

Calculate some examples of $\nabla f(1) \cdot (x - 1)$.

Consider the function $f(x) = x^2$.

The derivative of f at $x = 1$ is $\nabla f(1) = 2$.



Single-variable Differentiation

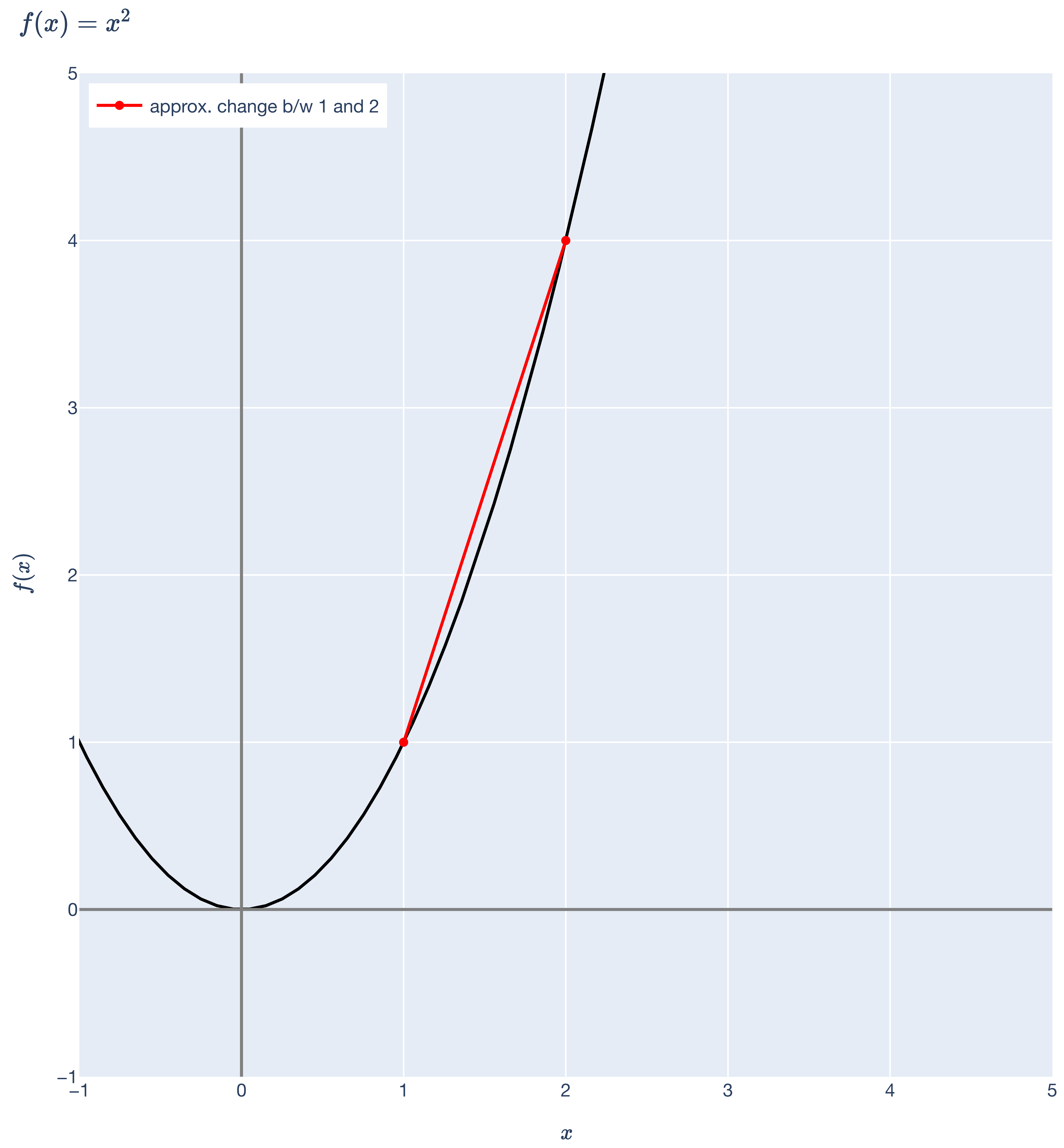
Linearity and differentiation

Calculate some examples of $\nabla f(1) \cdot (x - 1)$.

Consider the function $f(x) = x^2$.

The derivative of f at $x = 1$ is $\nabla f(1) = 2$.

$\nabla f(1)(2-1) = [2](2-1) = 2 \approx$ change in $f(x)$ between 1 and 2



Single-variable Differentiation

Linearity and differentiation

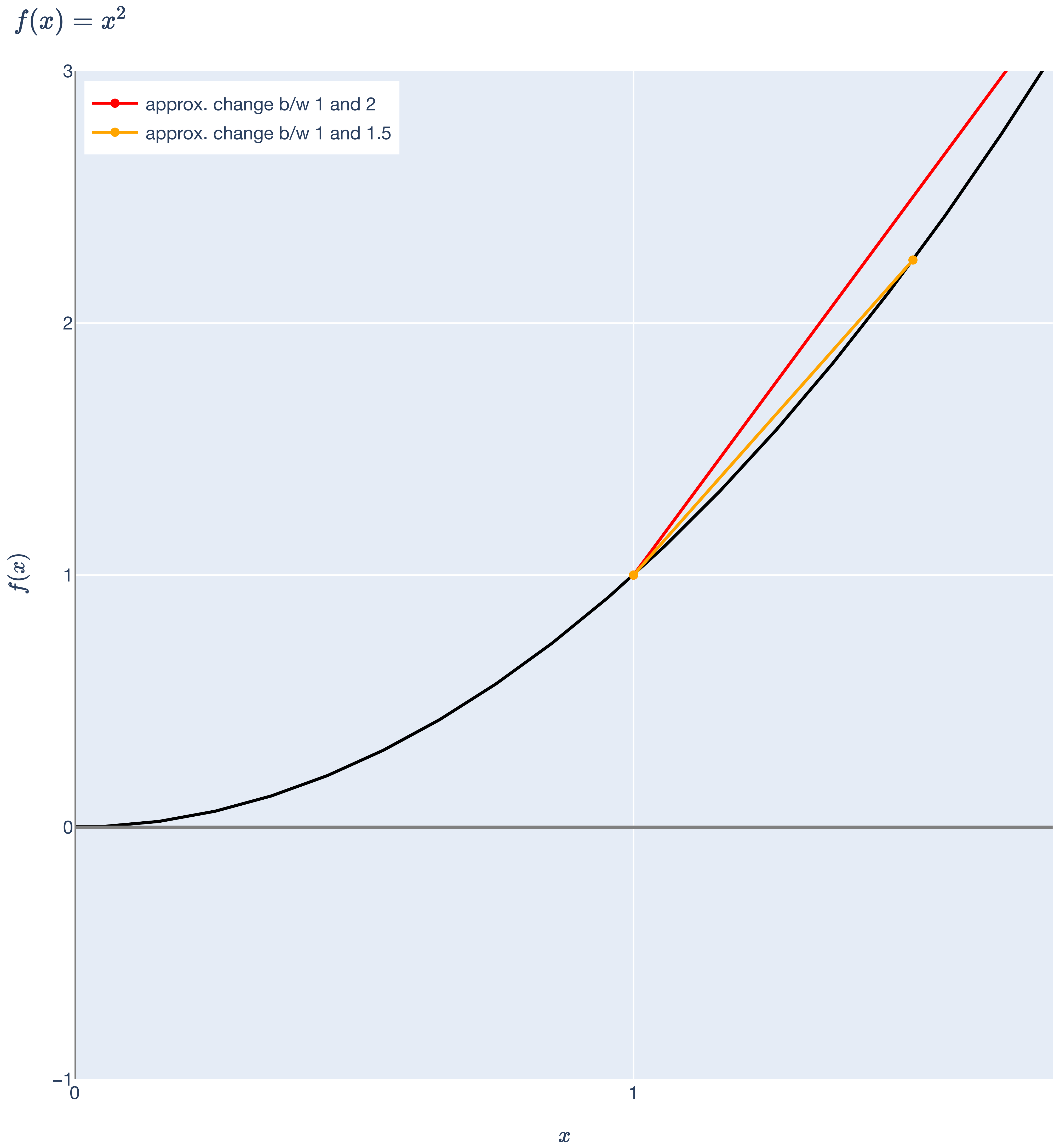
Calculate some examples of $\nabla f(1) \cdot (x - 1)$.

Consider the function $f(x) = x^2$.

The derivative of f at $x = 1$ is $\nabla f(1) = 2$.

$\nabla f(1)(2 - 1) = [2](2 - 1) = 2 \approx \text{change in } f(x) \text{ between } 1 \text{ and } 2$

$\nabla f(1)(1.5 - 1) = [2](1.5 - 1) = 1 \approx \text{change in } f(x) \text{ between } 1 \text{ and } 1.5$



Single-variable Differentiation

Linearity and differentiation

Calculate some examples of $\nabla f(1) \cdot (x - 1)$.

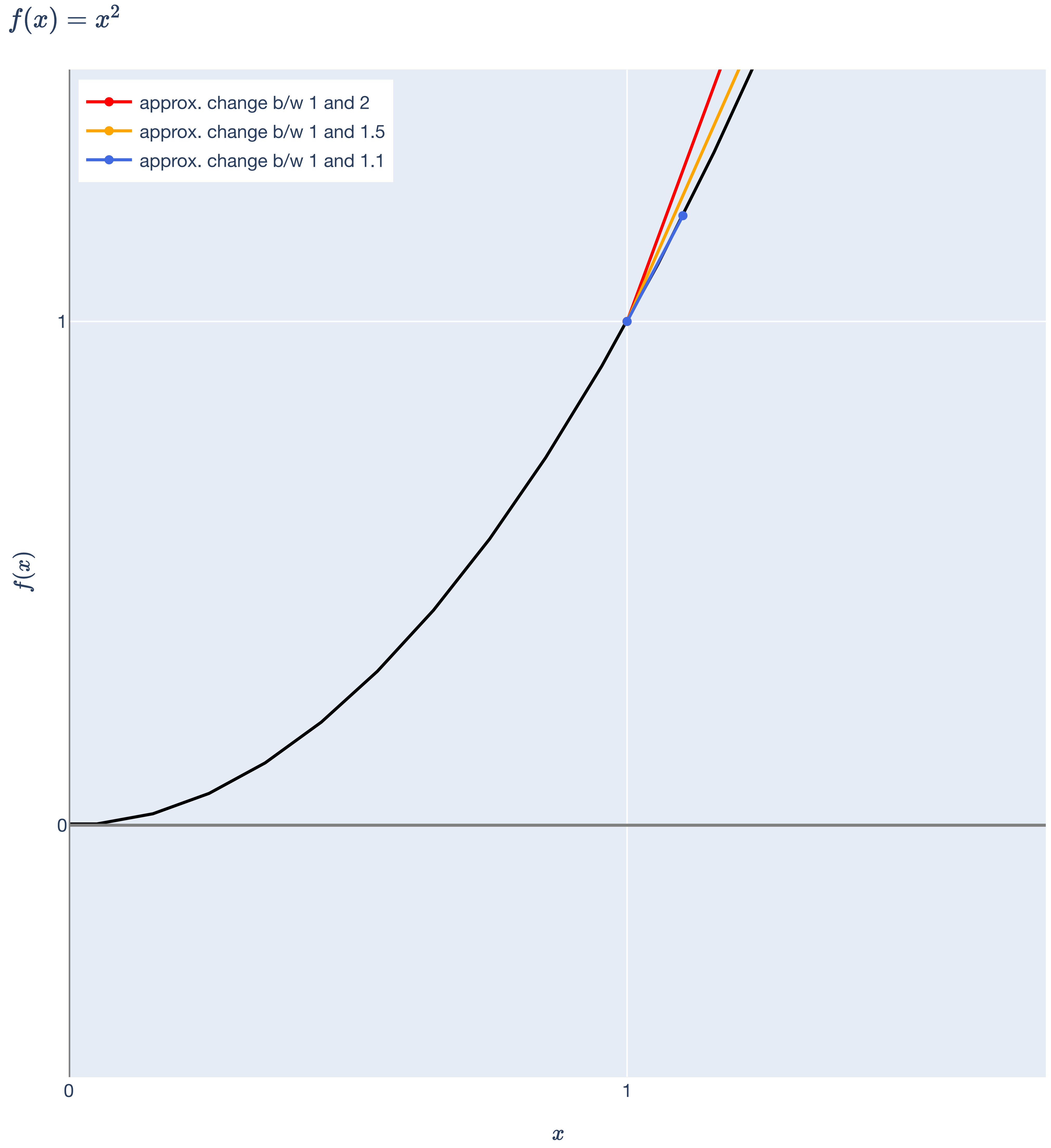
Consider the function $f(x) = x^2$.

The derivative of f at $x = 1$ is $\nabla f(1) = 2$.

$\nabla f(1)(2 - 1) = [2](2 - 1) = 2 \approx \text{change in } f(x) \text{ between } 1 \text{ and } 2$

$\nabla f(1)(1.5 - 1) = [2](1.5 - 1) = 1 \approx \text{change in } f(x) \text{ between } 1 \text{ and } 1.5$

$\nabla f(1)(1.1 - 1) = [2](1.1 - 1) = 0.2 \approx \text{change in } f(x) \text{ between } 1 \text{ and } 1.1$



Single-variable Differentiation

Linearity and differentiation

The derivative is a linear transformation that maps changes in x to changes in y.
We like linear transformations!

$$T : \text{change in x} \rightarrow \text{change in y}$$

$$\nabla f(x_0)(x - x_0) \approx f(x) - f(x_0)$$

The derivative is nothing more than a 1×1 matrix in single-variable differentiation.

Multivariable Differentiation

Review of multivariable notions of derivative

Multivariable Differentiation

Scalar-valued vs. vector-valued functions

$f : \mathbb{R}^d \to \mathbb{R}$ is a ***scalar-valued*** multivariable function, $\mathbf{f} : \mathbb{R}^d \to \mathbb{R}^n$ is a ***vector-valued*** multivariable function.

$$\mathbf{f}(\mathbf{x}_0) = (f_1(\mathbf{x}_0), \ldots, f_n(\mathbf{x}_0)) \,.$$

But $\mathbf{f}$ is just made up of n scalar-valued functions.

Upshot: Just treat vector-valued functions as a collection of n scalar-valued functions, and deal with each coordinate individually.

Multivariable Differentiation

Big picture: total, partial, and directional derivatives.

The ***total derivative*** (or just derivative) of $\mathbf{f}$ at $\mathbf{x}_0$ is a linear transformation $D\mathbf{f}(\mathbf{x}_0) : \mathbb{R}^d \to \mathbb{R}^n$.

The ***gradient*** of f at $\mathbf{x}_0$ is the vector $\nabla f(\mathbf{x}_0) \in \mathbb{R}^d$ associated with the total derivative of a scalar-valued $f : \mathbb{R}^d \to \mathbb{R}$.

The ***Jacobian*** of $\mathbf{f}$ at $\mathbf{x}_0$ is the $n \times d$ matrix $\nabla \mathbf{f}(\mathbf{x}_0)$ associated with the total derivative of a vector-valued $\mathbf{f} : \mathbb{R}^d \to \mathbb{R}^n$.

The ***directional derivative*** of $\mathbf{f}$ at $\mathbf{x}_0$ in the direction $\mathbf{v} \in \mathbb{R}^d$ is the derivative applied to $\mathbf{v}$:
$\nabla \underbrace{\mathbf{f}(\mathbf{x}_0)}_{n\times d} \underbrace{\mathbf{v}}_{d\times 1}$, via matrix-vector multiplication.

The ***i'th partial derivative*** of $\mathbf{f}$ at $\mathbf{x}_0$ is the directional derivative in the unit basis direction $\mathbf{e}_i \in \mathbb{R}^n$.

Multivariable Differentiation

Why is multivariable differentiation harder to pin down than single-variable differentiation?

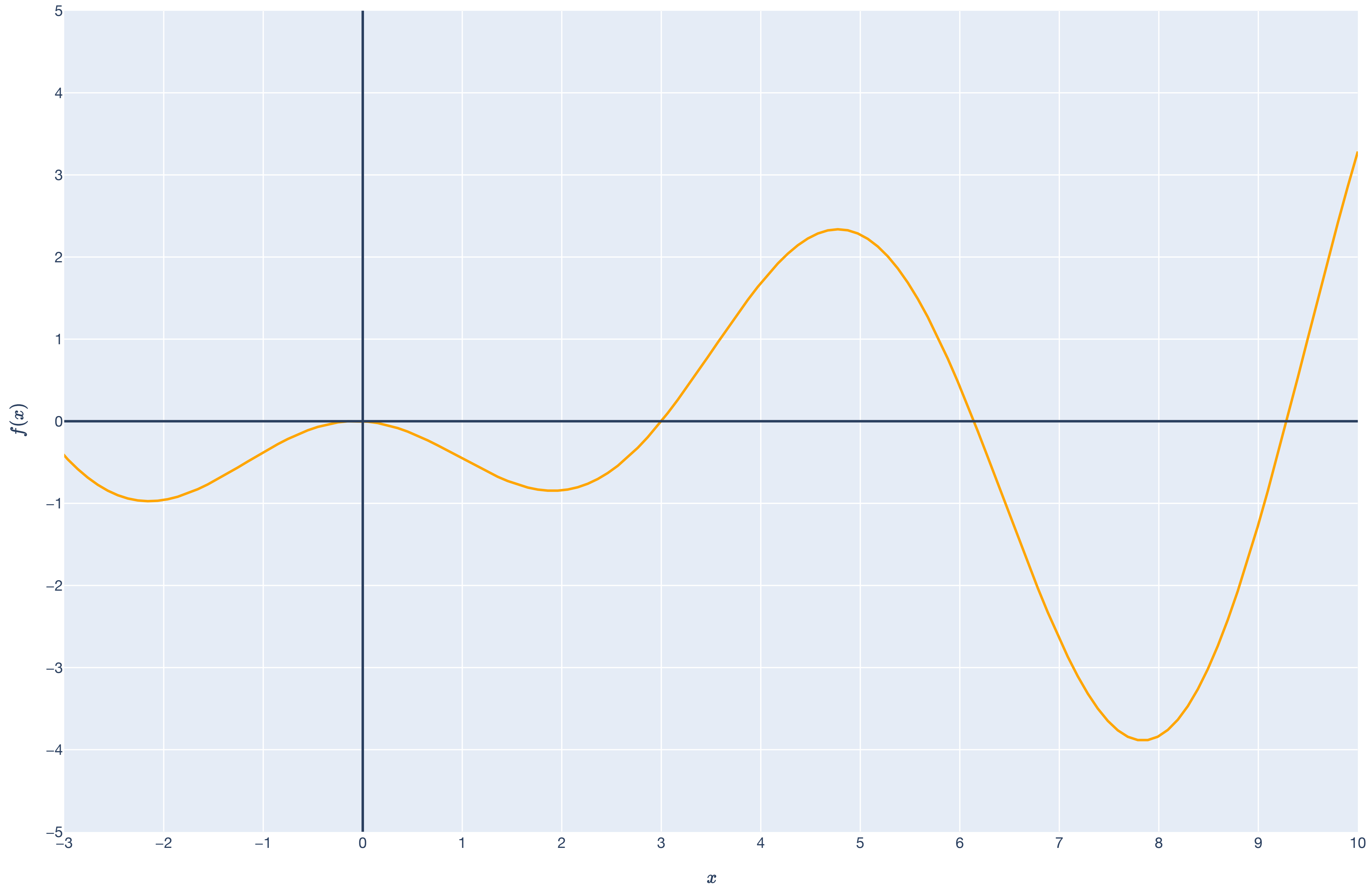
In $\mathbb{R}$, there are only two directions from which we can approach x_0 (on a standard Cartesian plane, the "left" and the "right").

In $\mathbb{R}^n$, we can approach $\mathbf{x}_0$ from infinitely many directions!

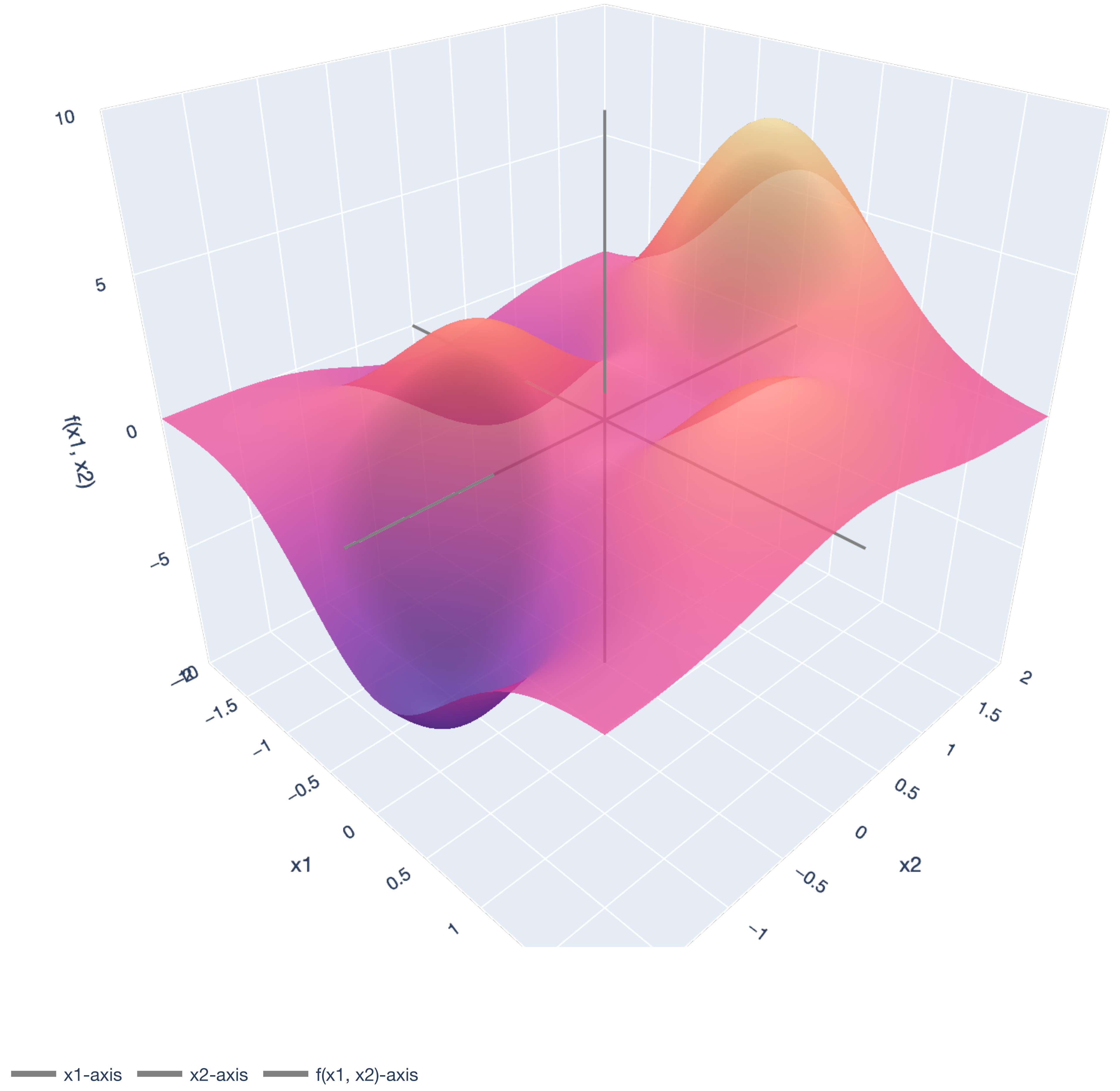
Multivariable Differentiation

Approach directions

$$f: \mathbb{R} \to \mathbb{R}$$

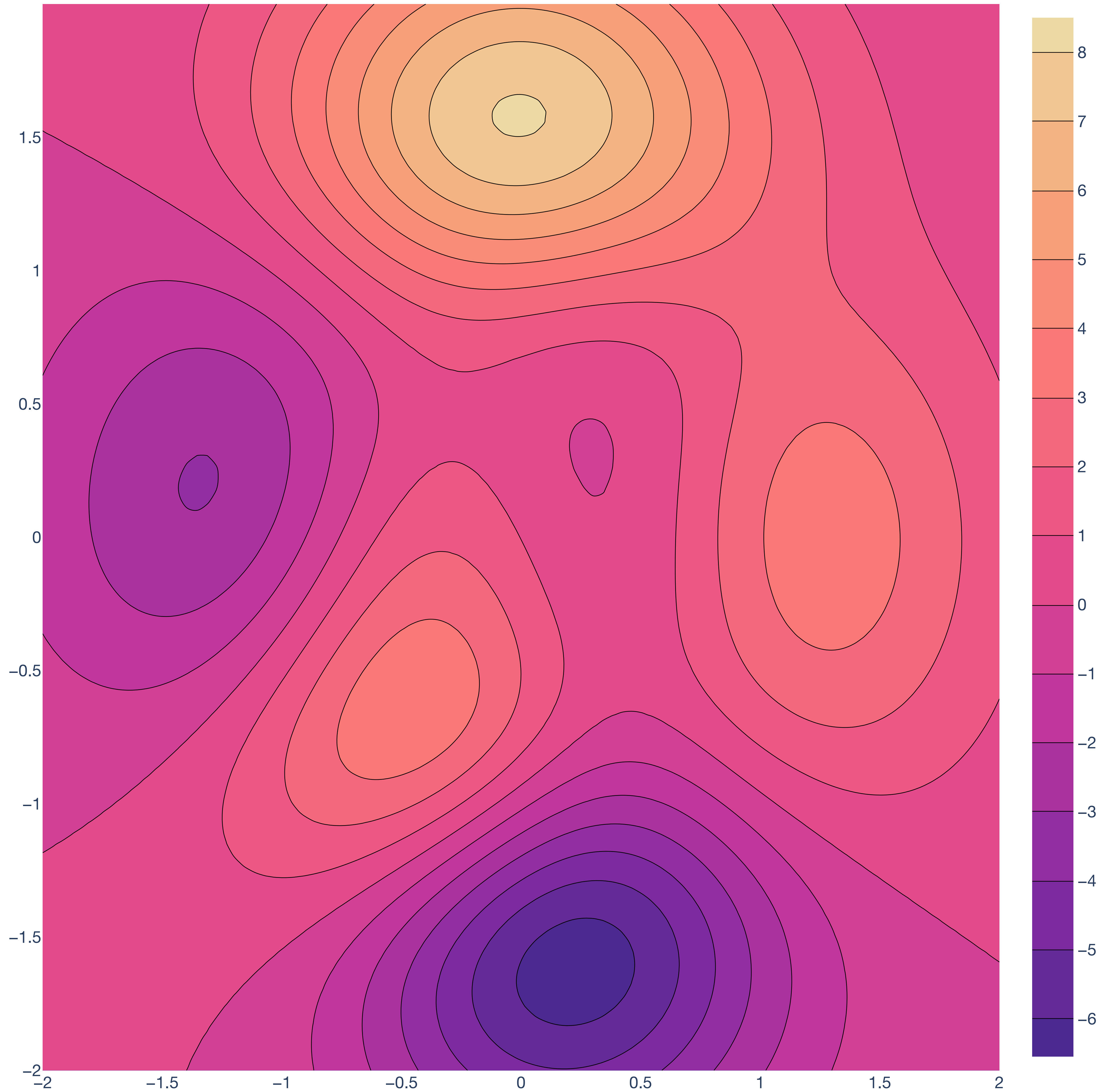


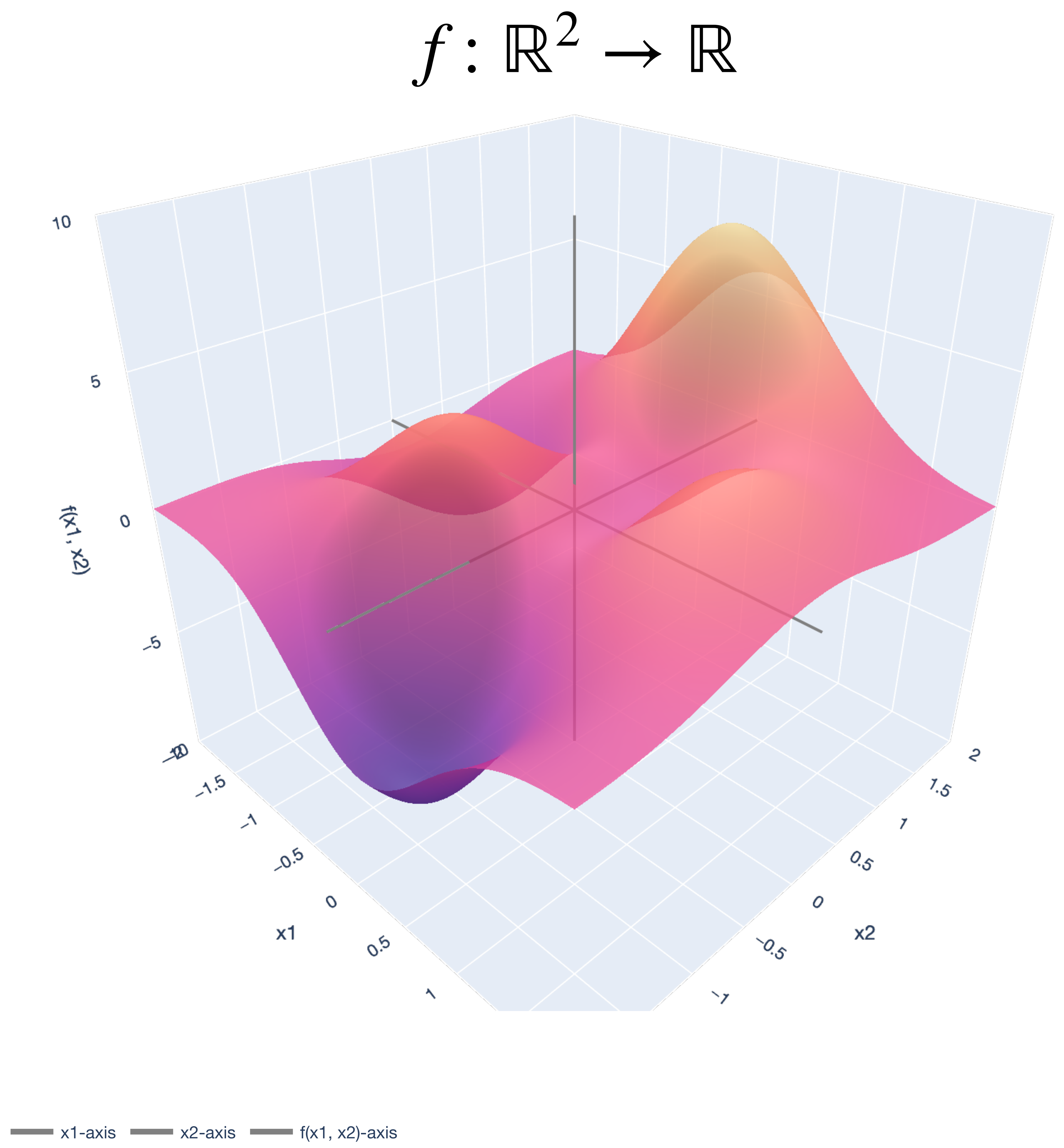
$$f: \mathbb{R}^2 \to \mathbb{R}$$



Multivariable Differentiation

Approach directions





Multivariable Differentiation

Directional and partial derivatives

Multivariable Differentiation

Directional and partial derivatives

For $\mathbf{f} : \mathbb{R}^d \to \mathbb{R}^n$ and point $\mathbf{x}_0 \ldots$

The ***directional derivative*** is change in $\mathbf{f}$ when we approach $\mathbf{x}_0$ from the direction defined by some vector $\mathbf{v}$.

The ***ith partial derivative*** is change in $\mathbf{f}$ when we approach $\mathbf{x}_0$ from the standard basis direction $\mathbf{e}_i$.

Multivariable Differentiation

Directional derivative

Let $\mathbf{f} : \mathbb{R}^d \to \mathbb{R}^n$ be a function. The ***directional derivative*** of $\mathbf{f}$ at $\mathbf{x}_0$ in the direction $\mathbf{v} \in \mathbb{R}^d$ is

$$\lim_{\delta \to 0} \frac{\mathbf{f}(\mathbf{x}_0 + \delta \mathbf{v}) - \mathbf{f}(\mathbf{x}_0)}{\delta}.$$

Multivariable Differentiation

Partial derivative

Let $\mathbf{e}_i$ be the ith standard basis vector in $\mathbb{R}^d$.

The ***ith partial derivative*** of $\mathbf{f}$ at $\mathbf{x}_0$ is the directional derivative in the direction $\mathbf{e}_i$, also written as:

$$\lim_{\delta \to 0} \frac{\mathbf{f}(\mathbf{x}_0 + \delta \mathbf{e}_i) - \mathbf{f}(\mathbf{x}_0)}{\delta}.$$

Multivariable Differentiation

Partial derivative

The *ith partial derivative* of $\mathbf{f}$ at $\mathbf{x}_0$ can also be written:

$$\frac{\partial \mathbf{f}}{\partial x_i}(\mathbf{x}_0) := \lim_{\delta \to 0} \frac{\mathbf{f}(\mathbf{x}_0 + \delta \mathbf{e}_i) - \mathbf{f}(\mathbf{x}_0)}{\delta} = \lim_{\delta \to 0} \frac{\mathbf{f}(x_{0,1}, \ldots, x_{0,i} + \delta, \ldots x_{0,n}) - \mathbf{f}(x_{0,1}, \ldots, x_{0,i}, \ldots, x_{0,n})}{\delta}$$

Mechanically: take the derivative of variable x_i while keeping all the others constant.

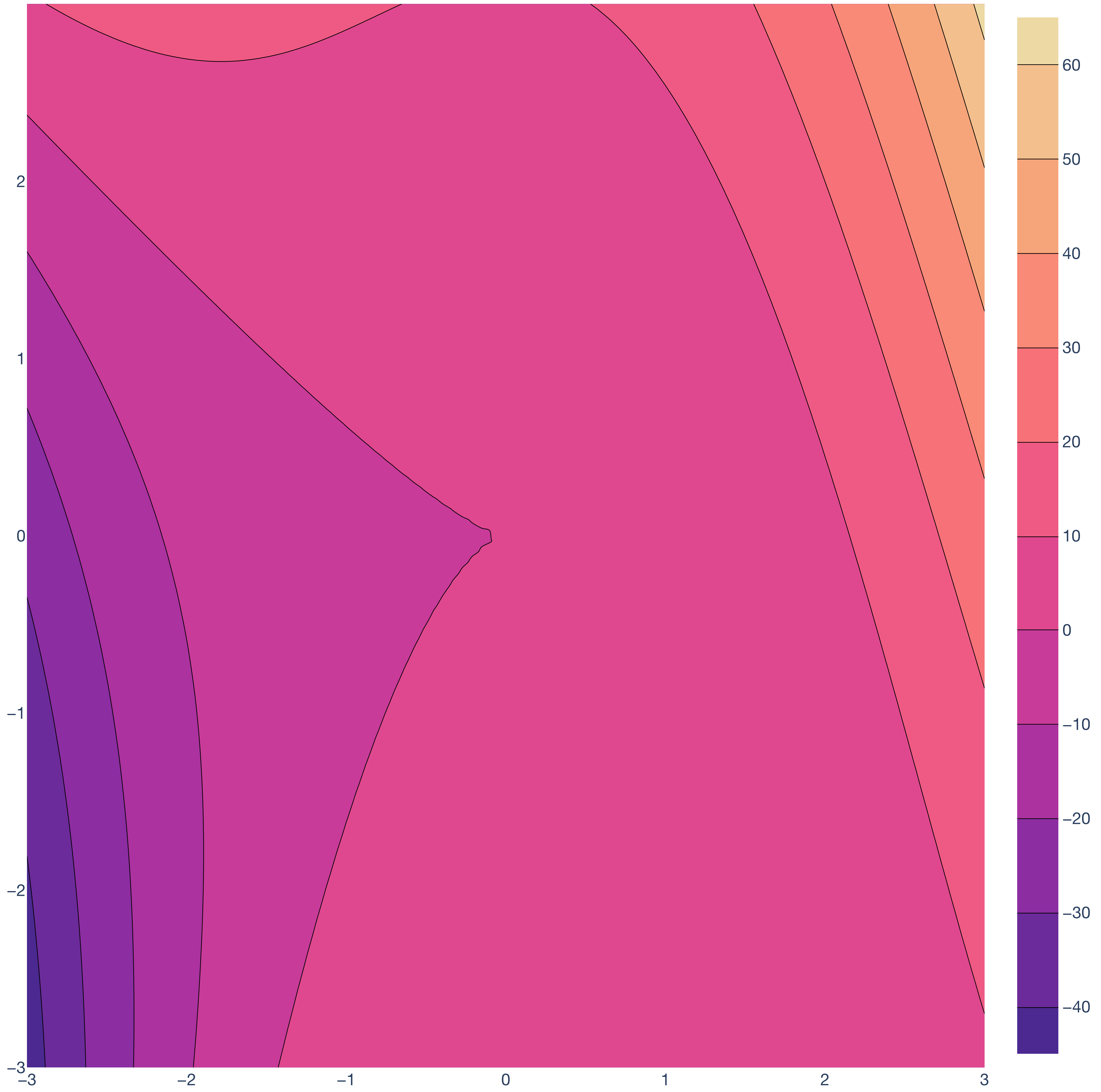
Multivariable Differentiation

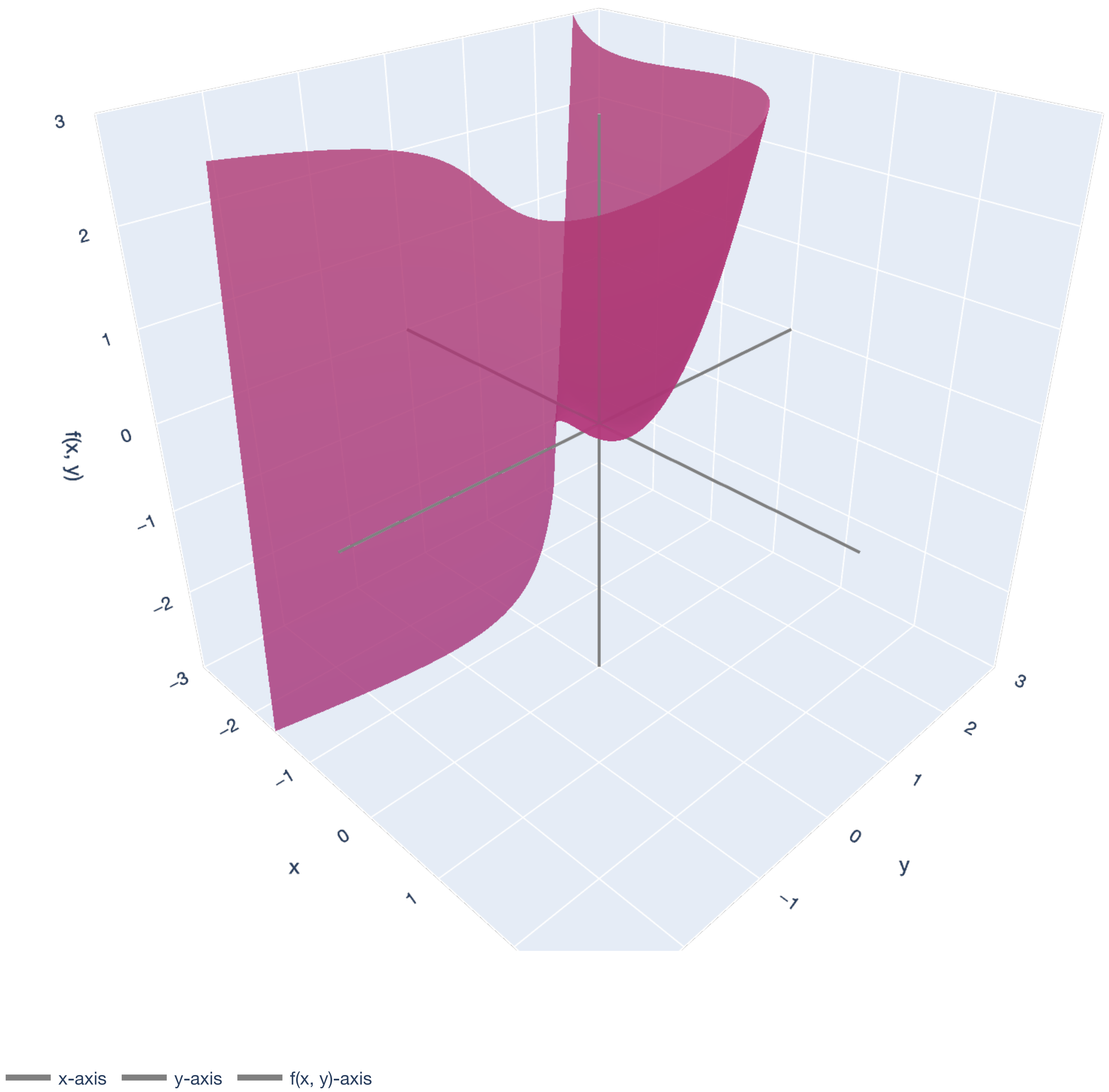
Example: $f(x, y) = x^3 + x^2y + y^2$

Example. Compute the partial derivatives of $f : \mathbb{R}^2 \to \mathbb{R}$ defined by $f(x, y) = x^3 + x^2y + y^2$. What are the partial derivatives at $(1,2)$?

Multivariable Differentiation

Example: $f(x, y) = x^3 + x^2y + y^2$





Multivariable Differentiation

Examples

Example. Compute the partial derivatives of $\mathbf{f} : \mathbb{R}^2 \to \mathbb{R}^2$ defined by $f(x, y) = (x^2y, \cos y)$. What are the partial derivatives at $(1,2)$?

Multivariable Differentiation

Total derivatives

Multivariable Differentiation

Jacobian and gradient idea

The ***gradient*** is the vector in $\mathbb{R}^d$ that contains the partial derivatives of $f : \mathbb{R}^d \to \mathbb{R}$ as each entry.

The ***Jacobian*** $n \times d$ matrix that contains the partial derivatives of $\mathbf{f} : \mathbb{R}^d \to \mathbb{R}^n$, collected column-by-column.

Viewing $\mathbf{f}$ as a collection of n functions $\mathbf{f} = (f_1, \ldots, f_n)$, the Jacobian is also what we get by "stacking" all the gradients top-to-bottom in a matrix.

Multivariable Differentiation

Gradient

Let $f : \mathbb{R}^d \to \mathbb{R}$ be a function. The ***gradient*** of f at $\mathbf{x}_0$ is the vector $\nabla f(\mathbf{x}_0) \in \mathbb{R}^d$ composed of all the partial derivatives of f at $\mathbf{x}_0$:

$$\nabla f(\mathbf{x}_0) := \begin{bmatrix} \frac{\partial f}{\partial x_1}(\mathbf{x}_0) \\ \vdots \\ \frac{\partial f}{\partial x_n}(\mathbf{x}_0) \end{bmatrix}$$

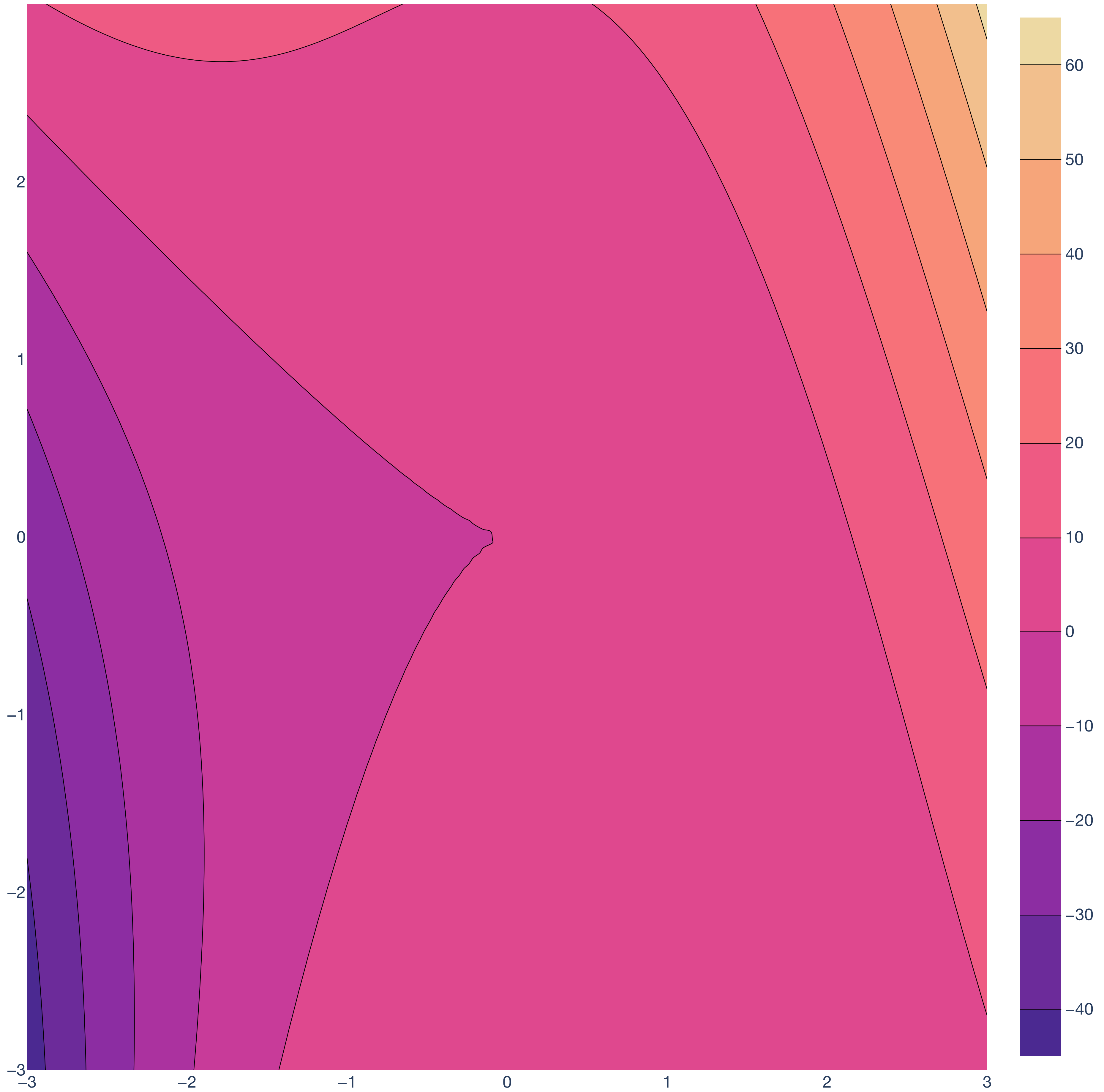
Multivariable Differentiation

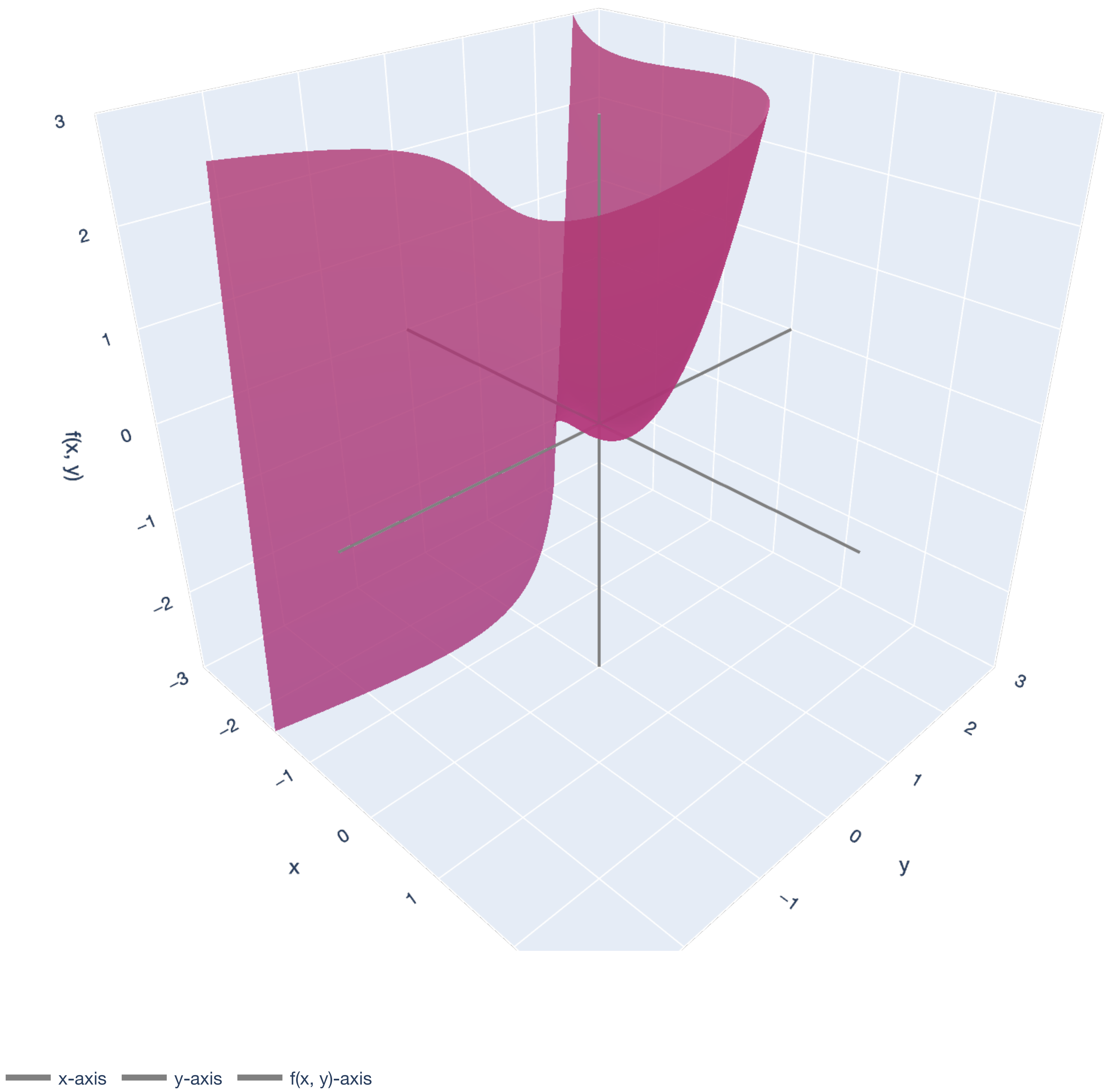
Gradient

Example. What's a formula for the gradient of $f(x, y) = x^3 + x^2y + y^2$?

Multivariable Differentiation

Example: $f(x, y) = x^3 + x^2y + y^2$





Multivariable Differentiation

Jacobian

Let $\mathbf{f} : \mathbb{R}^d \to \mathbb{R}^n$ be a function. The ***Jacobian*** of $\mathbf{f}$ at $\mathbf{x}_0$ is the $n \times d$ matrix composed of all the partial derivatives of $\mathbf{f}$ at $\mathbf{x}_0$:

$$\nabla \mathbf{f}(\mathbf{x}_0) := \begin{bmatrix} \frac{\partial f_1}{\partial x_1}(\mathbf{x}_0) & \cdots & \frac{\partial f_1}{\partial x_n}(\mathbf{x}_0) \\ \vdots & & \vdots \\ \frac{\partial f_m}{\partial x_1}(\mathbf{x}_0) & \cdots & \frac{\partial f_m}{\partial x_n}(\mathbf{x}_0) \end{bmatrix} = \begin{bmatrix} \leftarrow & \nabla f_1(\mathbf{x}_0)^\top & \rightarrow \\ \vdots & \vdots & \vdots \\ \leftarrow & \nabla f_n(\mathbf{x}_0)^\top & \rightarrow \end{bmatrix}$$

Multivariable Differentiation

Jacobian

Example. What's the Jacobian of $f(x, y) = (x^2y, \cos y)$?

“Local” to a Point

Definition of an open ball/neighborhood

Let $\mathbf{x} \in \mathbb{R}^d$ be a point. For some real value $\delta > 0$, the ***open ball*** or ***neighborhood of radius*** δ around $\mathbf{x}$ is the set of all points:

$$B_\delta(\mathbf{x}) := \{\mathbf{a} \in \mathbb{R}^d : \|\mathbf{x} - \mathbf{a}\| < \delta\}.$$

“Local” to a Point

Definition of an open ball/neighborhood

Example. Consider $\mathbf{x} = (1,1) \in \mathbb{R}^2$. What is the open ball of radius $\delta = 1$ around $\mathbf{x}$?

"Local" to a Point

Definition of an open ball/neighborhood

Example. Consider $\mathbf{x} = (1,1) \in \mathbb{R}^2$. What is the open ball of radius $\delta = 1$ around $\mathbf{x}$?

An open ball lets us approach $\mathbf{x}$ from all directions.

Multivariable Differentiation

Total Derivative

The ***total derivative*** is the linear transformation that "best approximates" the *local* change in $\mathbf{f}$ at a point $\mathbf{x}_0$.

The total derivative, like the univariate derivative, takes "change in $\mathbf{x}$" and outputs "change in $\mathbf{y}$."

Recall: $\nabla f(x_0)(x - x_0) \approx f(x) - f(x_0)$

Multivariable Differentiation

Total Derivative

Let $\mathbf{f} : \mathbb{R}^d \to \mathbb{R}^n$ be a function and let $\mathbf{x}_0 \in \mathbb{R}^d$ be a point. If there exists a linear transformation $D\mathbf{f}_{\mathbf{x}_0} : \mathbb{R}^d \to \mathbb{R}^n$ such that

$$\lim_{\vec{\delta}\to 0} \frac{1}{||\vec{\delta}||_2} \left(\left(\mathbf{f}(\mathbf{x}_0 + \vec{\delta}) - \mathbf{f}(\mathbf{x}_0) \right) - D\mathbf{f}_{\mathbf{x}_0}(\vec{\delta}) \right) = \mathbf{0},$$

then $\mathbf{f}$ is ***differentiable*** at $\mathbf{x}_0$ and has the unique ***(total) derivative*** $D\mathbf{f}_{\mathbf{x}_0}$.

As we get closer to $\mathbf{x}_0$ from any direction $\vec{\delta}$, the change $\mathbf{f}(\mathbf{x}_0 + \vec{\delta}) - \mathbf{f}(\mathbf{x}_0)$ can be approximated by $D\mathbf{f}_{\mathbf{x}_0}$.

Multivariable Differentiation

Total Derivative

Good news: in many cases, we don't have to deal with the clunky expression

$$\lim_{\vec{\delta}\to 0} \frac{1}{||\vec{\delta}||_2} \left(\left(\mathbf{f}(\mathbf{x}_0 + \vec{\delta}) - \mathbf{f}(\mathbf{x}_0) \right) - D\mathbf{f}_{\mathbf{x}_0}(\vec{\delta}) \right) = \mathbf{0},$$

because we can replace $D\mathbf{f}_{\mathbf{x}_0}$ by the Jacobian/gradient for all "nice" functions (the functions we usually care about)!

The "nice" functions is the class of ***continuously differentiable (smooth)*** functions.

Multivariable Differentiation

Smoothness and consequences

Multivariable Differentiation

Smoothness

A function $\mathbf{f} : \mathbb{R}^d \to \mathbb{R}^n$ is ***continuously differentiable*** if all of the partial derivatives of $\mathbf{f}$ exist and are continuous.

AKA: $\mathscr{C}^1$ *functions,* and the collection of all such functions are the class $\mathscr{C}^1$.

Generally: $\mathscr{C}^p$ for some $p \geq 1$ are the ***p-times continuously differentiable*** functions.

Multivariable Differentiation

Smoothness

Theorem (Sufficient criterion for differentiability). If $\mathbf{f} : \mathbb{R}^d \to \mathbb{R}^n$ is a $\mathscr{C}^1$ function, then $\mathbf{f}$ is differentiable, and its total derivative is equal to its Jacobian matrix.

Multivariable Differentiation

Directional derivatives from total derivative

Theorem (Computing directional derivatives). If $\mathbf{f} : \mathbb{R}^d \to \mathbb{R}^n$ is differentiable with $n \times d$ Jacobian matrix $\nabla \mathbf{f}(\mathbf{x}_0)$, the directional derivative of $\mathbf{f}$ at $\mathbf{x}_0$ in the direction $\mathbf{v} \in \mathbb{R}^d$ is given by the matrix-vector product:

$$\underbrace{\nabla \mathbf{f}(\mathbf{x}_0)}_{n \times d} \underbrace{\mathbf{v}}_{d \times 1} .$$

Remember from our linear algebra lectures: multiplying a vector by a matrix is applying a *linear transformation* to that vector!

Multivariable Differentiation

Gradient as direction of steepest ascent

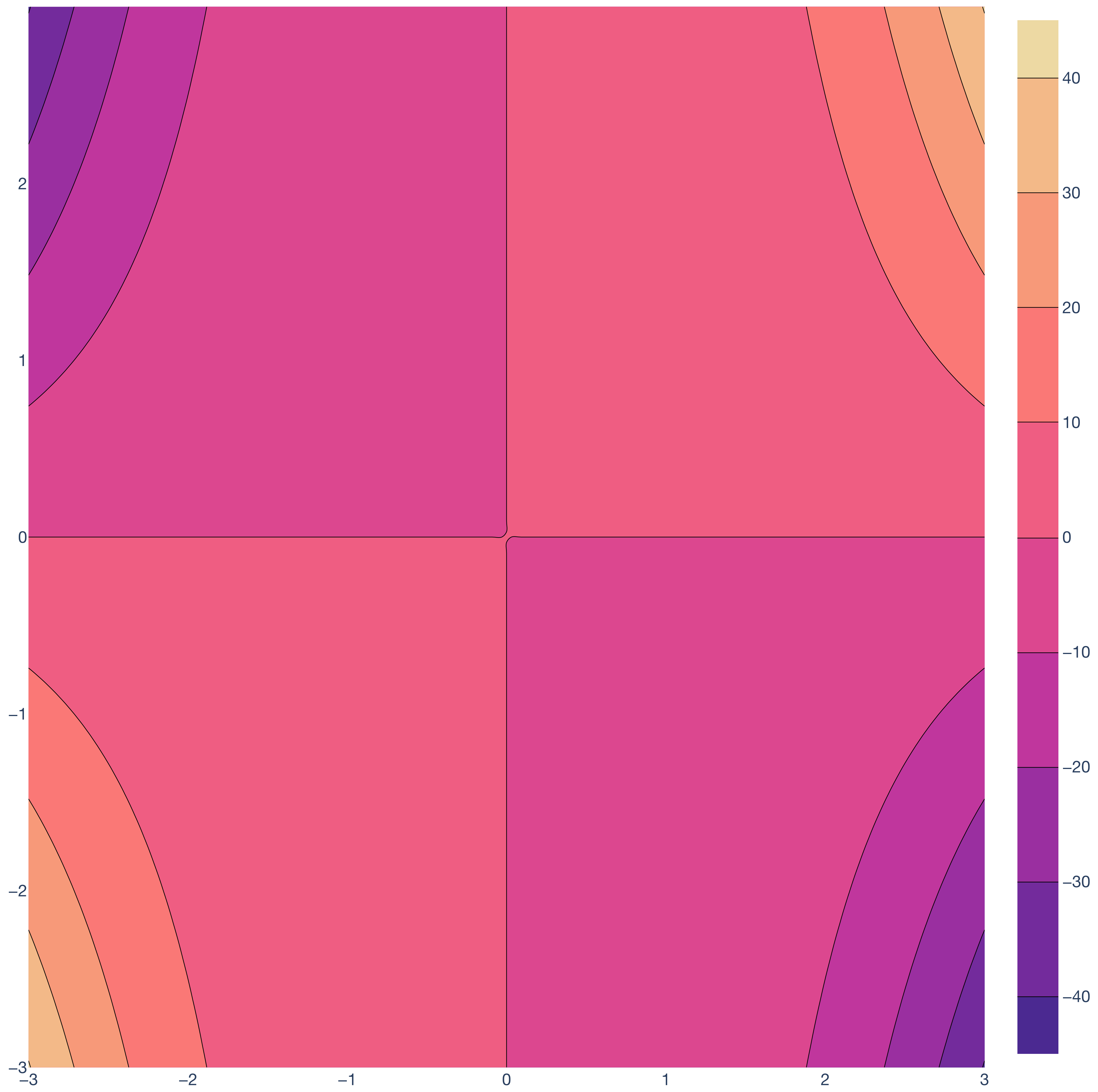
Theorem (Gradient and direction of steepest ascent). Let $f : \mathbb{R}^d \to \mathbb{R}$ be differentiable at $\mathbf{x}_0 \in \mathbb{R}^d$. If $\mathbf{v} \in \mathbb{R}^d$ is a *unit* vector making angle θ with the gradient $\nabla f(\mathbf{x}_0)$, then:

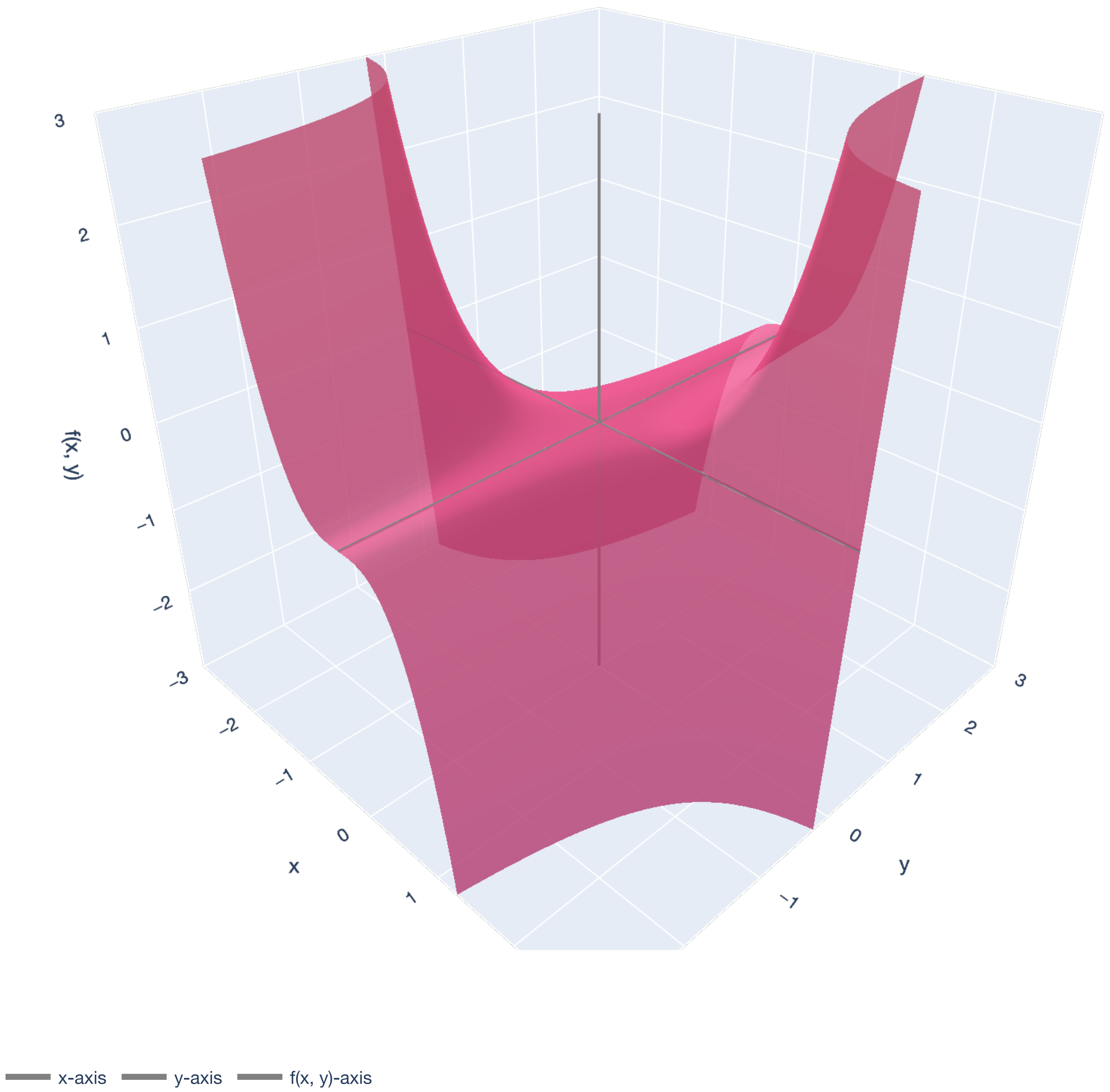
$$\nabla f(\mathbf{x}_0)^\top \mathbf{v} = \|\nabla f(\mathbf{x}_0)\| \cos\theta.$$

Gradient is the direction of *steepest ascent* at the rate $||\nabla f(\mathbf{x}_0)||$!

Multivariable Differentiation

Example: $f(x, y) = (1/2)x^3y$





Multivariable Differentiation

Big picture: how do all these objects connect?

The ***total derivative*** is a linear transformation that maps “changes in inputs” to “changes in outputs.”

When we apply a total derivative to a vector, think of mapping the “change” represented by that vector to a “change” in output space.

The ***partial derivative*** tells us how our function changes in each basis vector direction. The ***directional derivative*** tells us change in any direction.

For all the “smooth” ***continuously differentiable*** functions we care about, the total derivative is given by the ***Jacobian*** matrix (the ***gradient*** for scalar-valued functions).

Applying the Jacobian/gradient to a vector is the same as matrix-vector multiplication!

Multivariable Differentiation

Big picture: how do all these objects connect?

$\mathscr{C}^1$ function $\implies$ total derivative = Jacobian/gradient

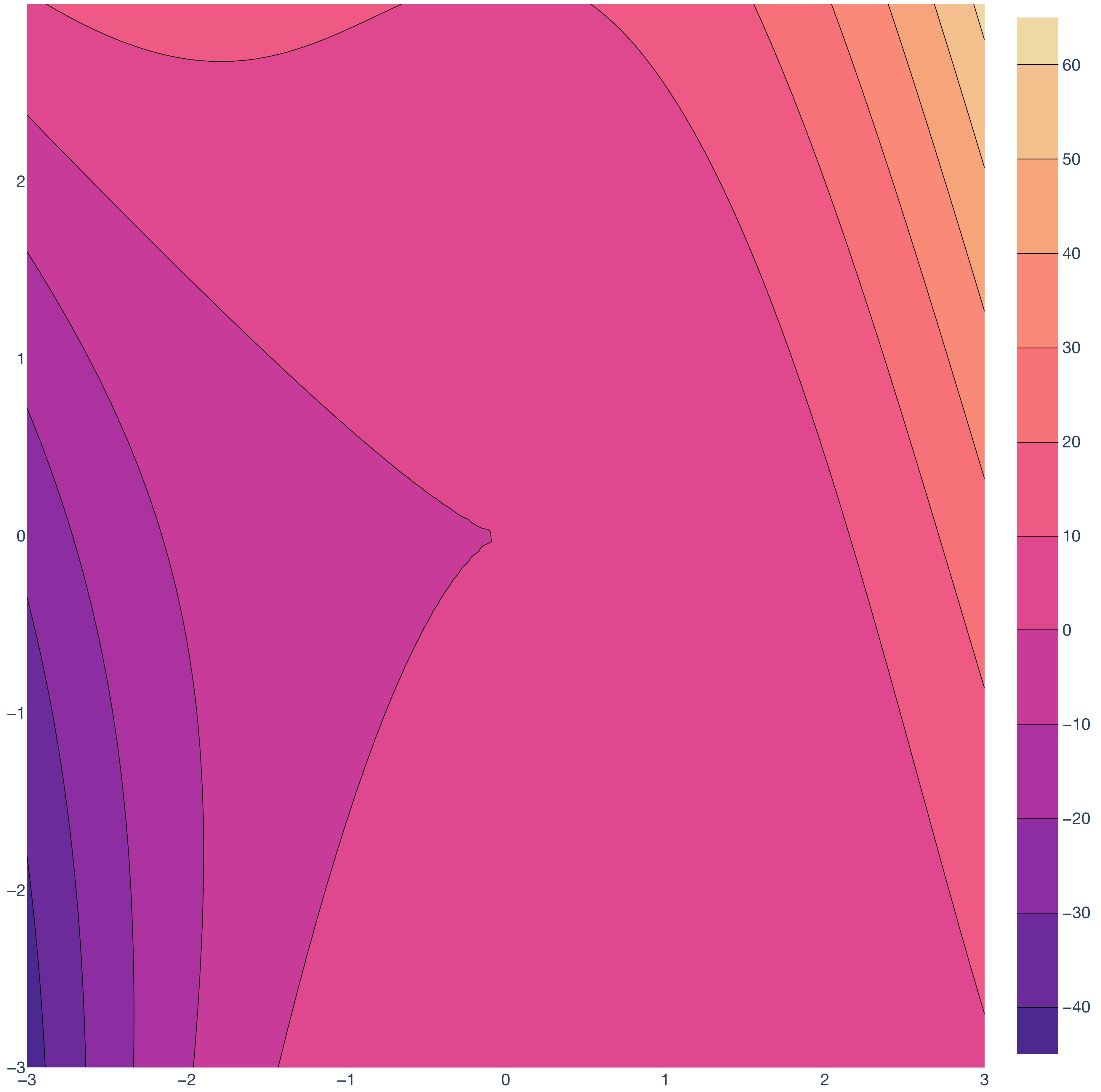
$\implies$ all directional/partial derivatives from matrix-vector product!

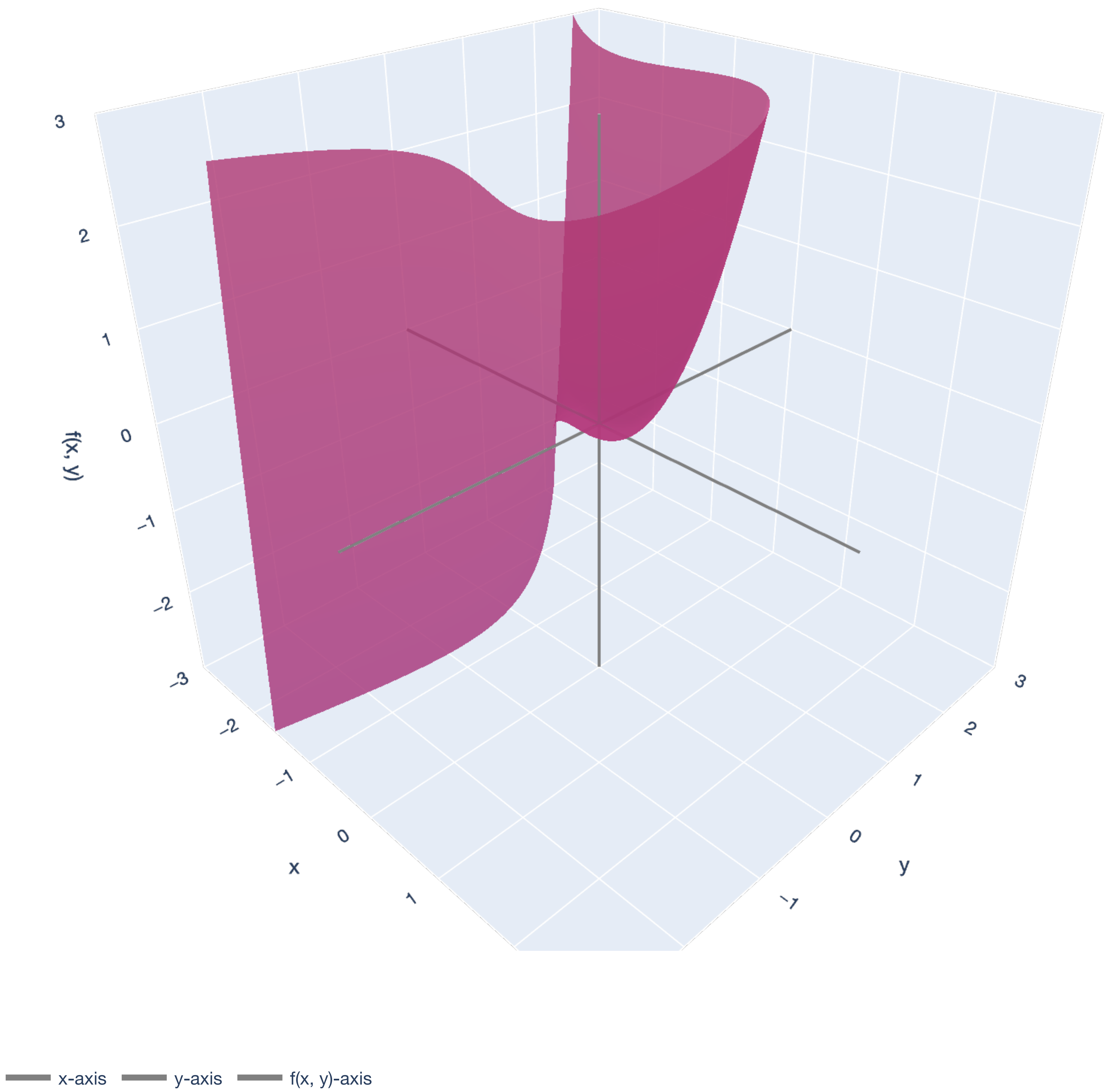
$\nabla \mathbf{f}(\mathbf{x}_0)\mathbf{v}$ for Jacobian ($\mathbf{f} : \mathbb{R}^d \to \mathbb{R}^n$)

$\nabla f(\mathbf{x}_0)^\top \mathbf{v}$ for gradient ($f : \mathbb{R}^d \to \mathbb{R}$)

Multivariable Differentiation

Example: $f(x, y) = x^3 + x^2y + y^2$





Multivariable Differentiation

The Hessian and the “Second Derivative”

Multivariable Differentiation: Hessian

Hessian matrix

The ***Hessian*** is the "second derivative" for scalar-valued multivariable functions. It is a matrix. For *really* smooth functions, it is symmetric.

The Hessian contains the local "second-order" information, or *curvature* of the function. It describes how "bowl-shaped" the function is around a point.

Note: The Hessian is only defined for scalar-valued functions $f : \mathbb{R}^n \to \mathbb{R}$.

Multivariable Differentiation: Hessian

Hessian matrix for $f : \mathbb{R}^2 \to \mathbb{R}$

The ***Hessian*** matrix for $f : \mathbb{R}^2 \to \mathbb{R}$ is the 2×2 matrix of all second-order partial derivatives:

$$\nabla^2 f(\mathbf{x}_0) = \begin{bmatrix} \dfrac{\partial^2 f}{\partial x_1^2} & \dfrac{\partial^2 f}{\partial x_1 \partial x_2} \\ \dfrac{\partial^2 f}{\partial x_2 \partial x_1} & \dfrac{\partial^2 f}{\partial x_2^2} \end{bmatrix}$$

$\dfrac{\partial^2 f}{\partial x_i^2}$ is the second partial derivative of f with respect to x_i.

$\dfrac{\partial^2 f}{\partial x_i \partial x_j}$ is the partial derivative from differentiating w.r.t. x_j first and then differentiating w.r.t. x_i.

Multivariable Differentiation: Hessian

Hessian matrix for $f : \mathbb{R}^n \to \mathbb{R}$

The ***Hessian*** matrix for $f : \mathbb{R}^n \to \mathbb{R}$ is the $n \times n$ matrix of all second-order partial derivatives.

Multivariable Differentiation: Hessian

Equality of mixed partials

Theorem (Equality of mixed partials). If $f : \mathbb{R}^n \to \mathbb{R}$ is a *twice continuously differentiable* function (i.e., in class $\mathscr{C}^2$), then, for all pairs (i, j):

$$\frac{\partial^2 f}{\partial x_i \partial x_j} = \frac{\partial^2 f}{\partial x_j \partial x_i}.$$

This means that for $\mathscr{C}^2$ functions, the Hessian is a symmetric matrix.

$\mathscr{C}^2$, the class of ***twice continuously differentiable*** functions, is the collection of all functions whose second-order partial derivatives all exist and are continuous.

Multivariable Differentiation

Wrap-up example

Consider the function $\mathbf{f} : \mathbb{R}^2 \to \mathbb{R}^3$ given by

$$\mathbf{f}(x, y) := \begin{pmatrix} \frac{1}{2}x^3y & 2x^2y^2 & xy \end{pmatrix}.$$

Is $\mathbf{f}$ smooth (i.e. in $\mathscr{C}^1$)? How about $\mathscr{C}^2$? What does that tell us?

Multivariable Differentiation

Wrap-up example

Consider the function $\mathbf{f} : \mathbb{R}^2 \to \mathbb{R}^3$ given by

$$\mathbf{f}(x, y) := \begin{pmatrix} \frac{1}{2}x^3y & 2x^2y^2 & xy \end{pmatrix}.$$

What's the *formula for* the Jacobian of $\mathbf{f}$?

What's the *formula for* the gradient of $f_1(x, y) = \dfrac{1}{2}x^3y$? What is the Jacobian/gradient at $\mathbf{x}_0 = (1,2)$?

Multivariable Differentiation

Wrap-up example

Consider the function $\mathbf{f} : \mathbb{R}^2 \to \mathbb{R}^3$ given by

$$\mathbf{f}(x, y) := \begin{pmatrix} \frac{1}{2}x^3y & 2x^2y^2 & xy \end{pmatrix}.$$

What's the total derivative of $\mathbf{f}$ at $\mathbf{x}_0 = (1,0)$?

Multivariable Differentiation

Wrap-up example

Consider the function $\mathbf{f} : \mathbb{R}^2 \to \mathbb{R}^3$ given by

$$\mathbf{f}(x, y) := \begin{pmatrix} \frac{1}{2}x^3y & 2x^2y^2 & xy \end{pmatrix}.$$

What's the directional derivative of $\mathbf{f}$ at $\mathbf{x}_0$ in the direction $\mathbf{v} = (1,1)$?

How about in the direction $\mathbf{e}_1$?

Multivariable Differentiation

Common Derivative Rules

Multivariable Differentiation

Basic derivative rules

Same as single-variable differentiation rules, but we need to "type-check" dimensions.

Let $\dfrac{\partial}{\partial \mathbf{x}}$ be the differentiation "operator."

Derivatives of $\mathbf{f} : \mathbb{R}^d \to \mathbb{R}^n$ from reasoning about each scalar-valued $f_1, \ldots, f_n$.

Multivariable Differentiation

Sum Rule

For $f : \mathbb{R}^d \to \mathbb{R}$ and $g : \mathbb{R}^d \to \mathbb{R}$:

$$\frac{\partial}{\partial \mathbf{x}}(f(\mathbf{x}) + g(\mathbf{x})) = \frac{\partial f}{\partial \mathbf{x}} + \frac{\partial g}{\partial \mathbf{x}}$$

Multivariable Differentiation

Product Rule

For $f : \mathbb{R}^d \to \mathbb{R}$ and $g : \mathbb{R}^d \to \mathbb{R}$:

$$\frac{\partial}{\partial \mathbf{x}}(f(\mathbf{x})g(\mathbf{x})) = \frac{\partial f}{\partial \mathbf{x}}g(\mathbf{x}) + f(\mathbf{x})\frac{\partial g}{\partial \mathbf{x}}$$

Multivariable Differentiation

Chain Rule

For $f : \mathbb{R}^d \to \mathbb{R}$ and $g : \mathbb{R} \to \mathbb{R}$:

$$\frac{\partial}{\partial \mathbf{x}}(g \circ f)(\mathbf{x}) = \frac{\partial}{\partial \mathbf{x}} g(f(\mathbf{x})) = \frac{\partial g}{\partial f} \frac{\partial f}{\partial \mathbf{x}}$$

Multivariable Differentiation

Example of chain rule

Example. Let $g : \mathbb{R}^2 \to \mathbb{R}$ be defined as $g(y_1, y_2) = y_1^2 + 2y_2$. Let $\mathbf{f} : \mathbb{R}^2 \to \mathbb{R}^2$ be defined as $\mathbf{f}(x_1, x_2) := \begin{pmatrix} \sin(x_1) + \cos(x_2) & x_1 x_2^3 \end{pmatrix}$.

We can also write this as:

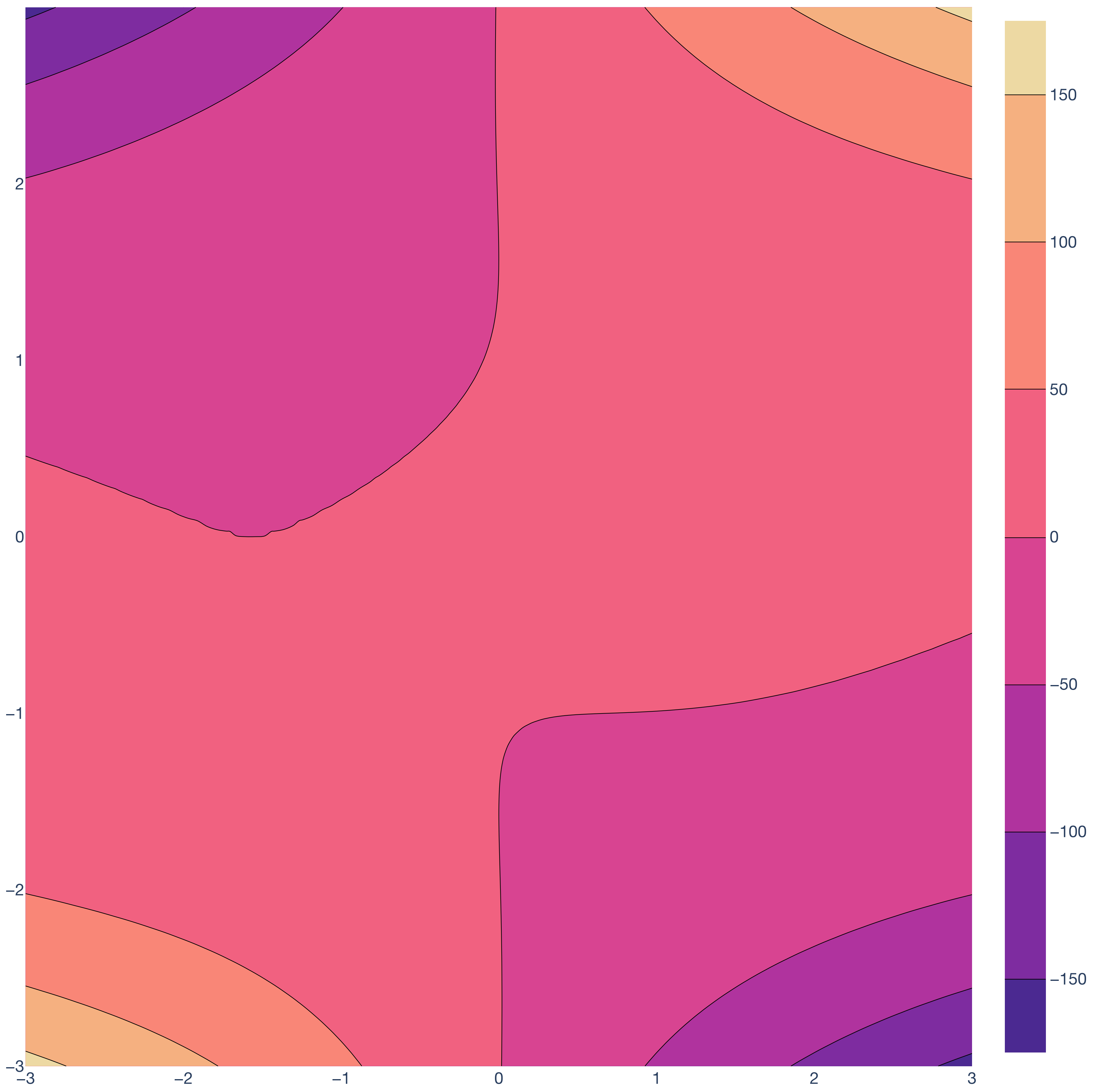
$$g(\mathbf{f}(\mathbf{x})) = (g \circ \mathbf{f})(x_1, x_2) = (\sin(x_1) + \cos(x_2))^2 + 2(x_1 x_2^3)$$

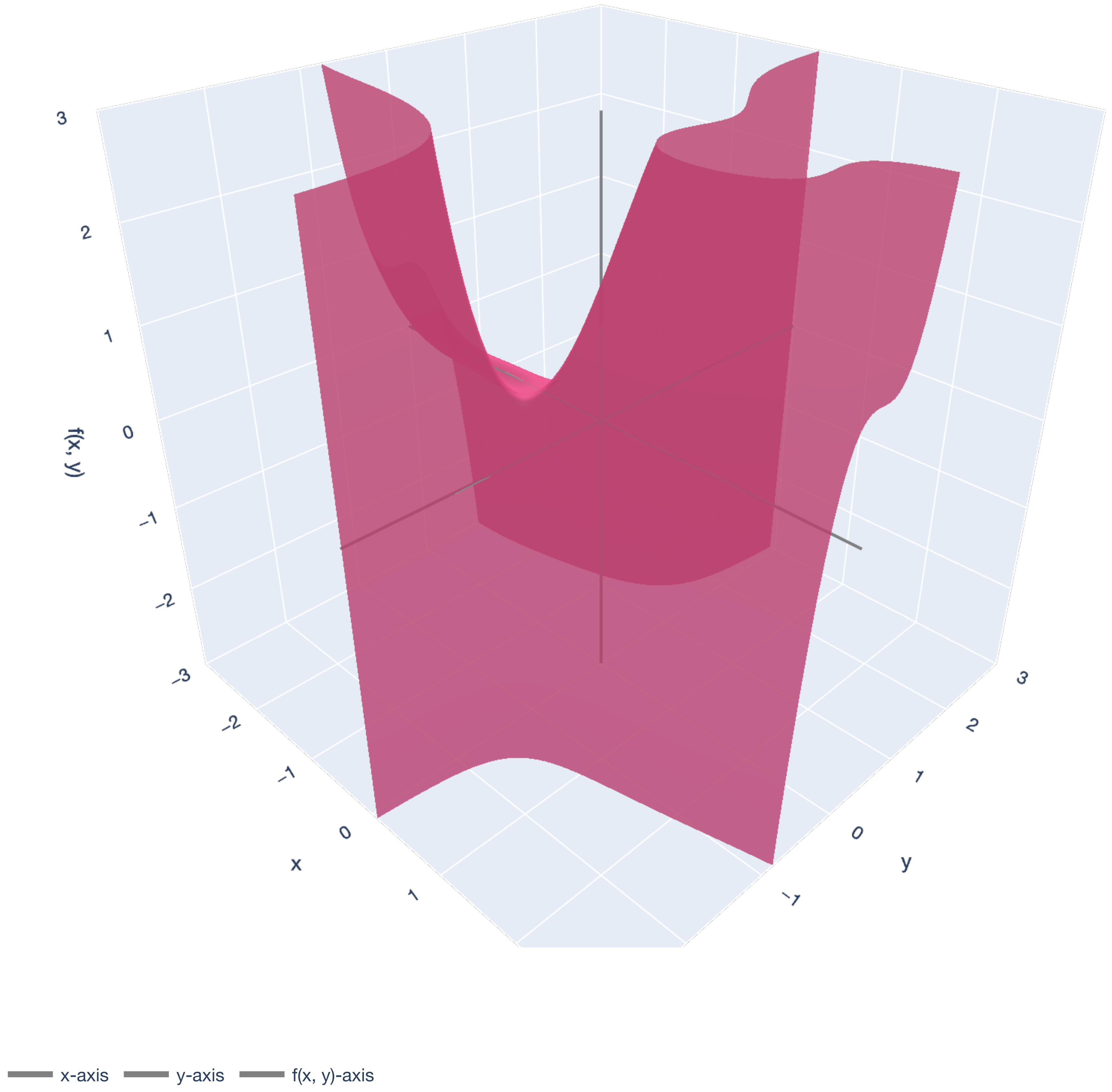
What is $\dfrac{\partial (g \circ \mathbf{f})}{\partial \mathbf{x}}$?

Multivariable Differentiation

Example of chain rule

$$g(\mathbf{f}(\mathbf{x})) = (g \circ \mathbf{f})(x_1, x_2) = (\sin(x_1) + \cos(x_2))^2 + 2(x_1 x_2^3)$$





“Matrix Calculus”

Useful identities in machine learning

$$\frac{\partial \mathbf{x}^\top \mathbf{a}}{\partial \mathbf{x}} = \mathbf{a}$$

$$\frac{\partial \mathbf{a}^\top \mathbf{x}}{\partial \mathbf{x}} = \mathbf{a}$$

$$\frac{\partial \mathbf{A}\mathbf{x}}{\partial \mathbf{x}} = \mathbf{A}$$

$$\frac{\partial \mathbf{x}^\top \mathbf{A}\mathbf{x}}{\mathbf{x}} = (\mathbf{A} + \mathbf{A}^\top)\mathbf{x}$$

More in *The Matrix Cookbook* (Petersen and Pederson, 2012).

“Matrix Calculus”

Example

Why $\dfrac{\partial \mathbf{x}^\top \mathbf{a}}{\partial \mathbf{x}} = \mathbf{a}$?

Why do we get $\dfrac{\partial \mathbf{a}^\top \mathbf{x}}{\partial \mathbf{x}}$ “for free?”

Least Squares

Optimization Perspective

Regression
Setup

<u>Observed:</u> Matrix of *training samples* $\mathbf{X} \in \mathbb{R}^{n \times d}$ and vector of *training labels* $\mathbf{y} \in \mathbb{R}^d$.

$$\mathbf{X} = \begin{bmatrix} \uparrow & & \uparrow \\ \mathbf{x}_1 & \cdots & \mathbf{x}_d \\ \downarrow & & \downarrow \end{bmatrix} = \begin{bmatrix} \leftarrow & \mathbf{x}_1^\top & \rightarrow \\ & \vdots & \\ \leftarrow & \mathbf{x}_n^\top & \rightarrow \end{bmatrix}.$$

<u>Unknown:</u> *Weight vector* $\mathbf{w} \in \mathbb{R}^d$ with weights $w_1, \ldots, w_d$.

<u>Goal:</u> For each $i \in [n]$, we predict: $\hat{y}_i = \mathbf{w}^\top \mathbf{x}_i = w_1 x_{i1} + \ldots + w_d x_{id} \in \mathbb{R}$.

Choose a weight vector that “fits the training data”: $\mathbf{w} \in \mathbb{R}^d$ such that $y_i \approx \hat{y}_i$ for $i \in [n]$, or:

$$\mathbf{X}\mathbf{w} = \hat{\mathbf{y}} \approx \mathbf{y}\,.$$

Regression

Setup

<u>**Goal:**</u> For each $i \in [n]$, we predict: $\hat{y}_i = \mathbf{w}^\top \mathbf{x}_i = w_1 x_{i1} + \ldots + w_d x_{id} \in \mathbb{R}$.

Choose a weight vector that "fits the training data": $\hat{\mathbf{w}} \in \mathbb{R}^d$ such that $y_i \approx \hat{y}_i$ for $i \in [n]$, or:

$$\mathbf{X}\hat{\mathbf{w}} = \hat{\mathbf{y}} \approx \mathbf{y}\,.$$

To find $\hat{\mathbf{w}}$, we follow the *principle of least squares.*

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w} \in \mathbb{R}^d} \ \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2$$

Least Squares

OLS Theorem

<u>Theorem (Ordinary Least Squares).</u> Let $\mathbf{X} \in \mathbb{R}^{n \times d}$ and $\mathbf{y} \in \mathbb{R}^n$. Let $\hat{\mathbf{w}} \in \mathbb{R}^d$ be the least squares minimizer:

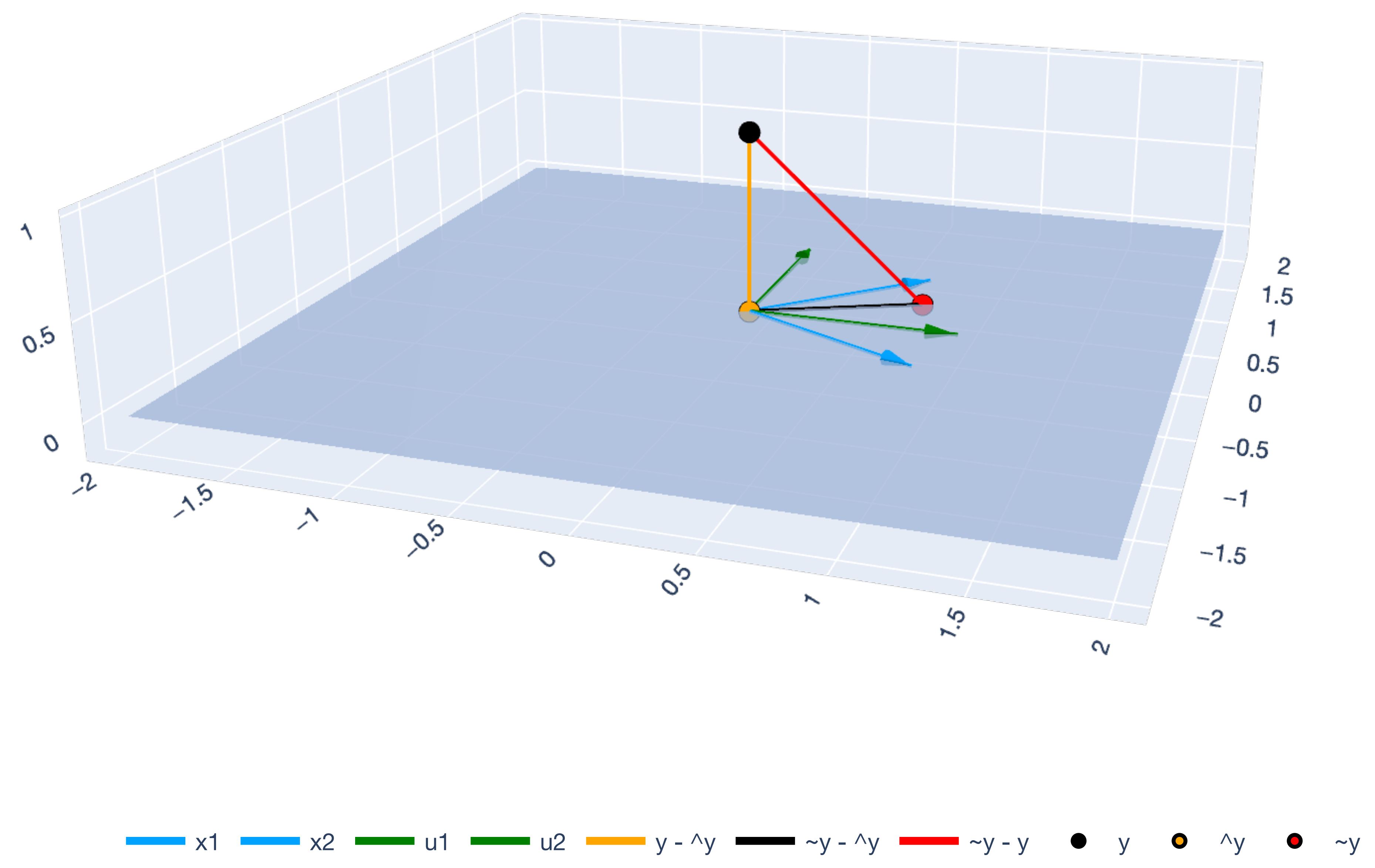
$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w} \in \mathbb{R}^d} \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2$$

If $n \geq d$ and $\operatorname{rank}(\mathbf{X}) = d$, then:

$$\hat{\mathbf{w}} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}.$$

To get predictions $\hat{\mathbf{y}} \in \mathbb{R}^n$:

$$\hat{\mathbf{y}} = \mathbf{X}\hat{\mathbf{w}} = \mathbf{X}(\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}.$$



Least Squares

OLS Theorem

<u>Theorem (Ordinary Least Squares).</u> Let $\mathbf{X} \in \mathbb{R}^{n \times d}$ and $\mathbf{y} \in \mathbb{R}^n$. Let $\hat{\mathbf{w}} \in \mathbb{R}^d$ be the least squares minimizer:

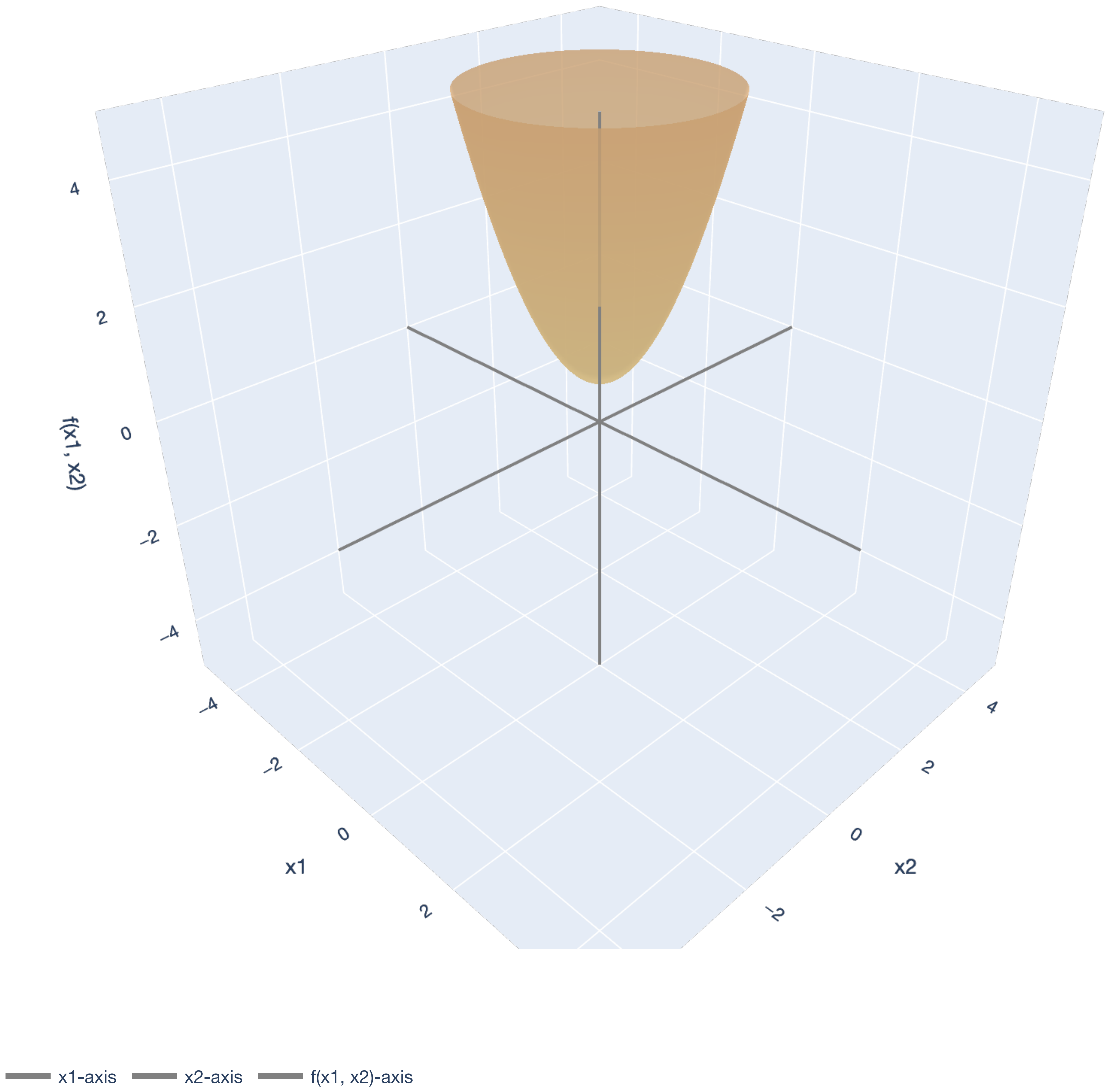
$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w} \in \mathbb{R}^d} \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2$$

If $n \geq d$ and $\mathrm{rank}(\mathbf{X}) = d$, then:

$$\hat{\mathbf{w}} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}.$$

To get predictions $\hat{\mathbf{y}} \in \mathbb{R}^n$:

$$\hat{\mathbf{y}} = \mathbf{X}\hat{\mathbf{w}} = \mathbf{X}(\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}.$$



Least Squares

Optimization Problem

Let $\mathbf{X} \in \mathbb{R}^{n \times d}$ and $\mathbf{y} \in \mathbb{R}^{n}$. Let $\hat{\mathbf{w}} \in \mathbb{R}^{d}$ be the least squares minimizer:

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w} \in \mathbb{R}^{d}} \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2$$

What if we consider this as an optimization problem instead?

Least Squares

Optimization Problem

Let $\mathbf{X} \in \mathbb{R}^{n \times d}$ and $\mathbf{y} \in \mathbb{R}^n$. Let $\hat{\mathbf{w}} \in \mathbb{R}^d$ be the least squares minimizer:

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w} \in \mathbb{R}^d} \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2$$

What if we consider this as an optimization problem instead?

$$f : \mathbb{R}^d \to \mathbb{R}$$

$$f(\mathbf{w}) = \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2$$

Least Squares

Optimization Problem

$$f : \mathbb{R}^d \to \mathbb{R}$$

$$f(\mathbf{w}) = \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2$$

Least Squares

Least Squares Objective

Before, we called this the ***squared error*** or ***sum of squared residuals***…

$$f : \mathbb{R}^d \to \mathbb{R}$$

$$f(\mathbf{w}) = ||\mathbf{X}\mathbf{w} - \mathbf{y}||^2$$

We can also consider this the *objective function* of an optimization problem: the ***least squares objective***.

Least Squares

Least Squares Objective in $\mathbb{R}$

$$f : \mathbb{R} \to \mathbb{R}$$

$$f(\mathbf{w}) = \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 \implies f(w) = \|w\mathbf{x} - \mathbf{y}\|^2$$

Least Squares

Least Squares Objective in $\mathbb{R}$

Consider the dataset $\mathbf{x} = (1, -1)$ and $\mathbf{y} = (3, -3)$, where $n = 2, d = 1$.

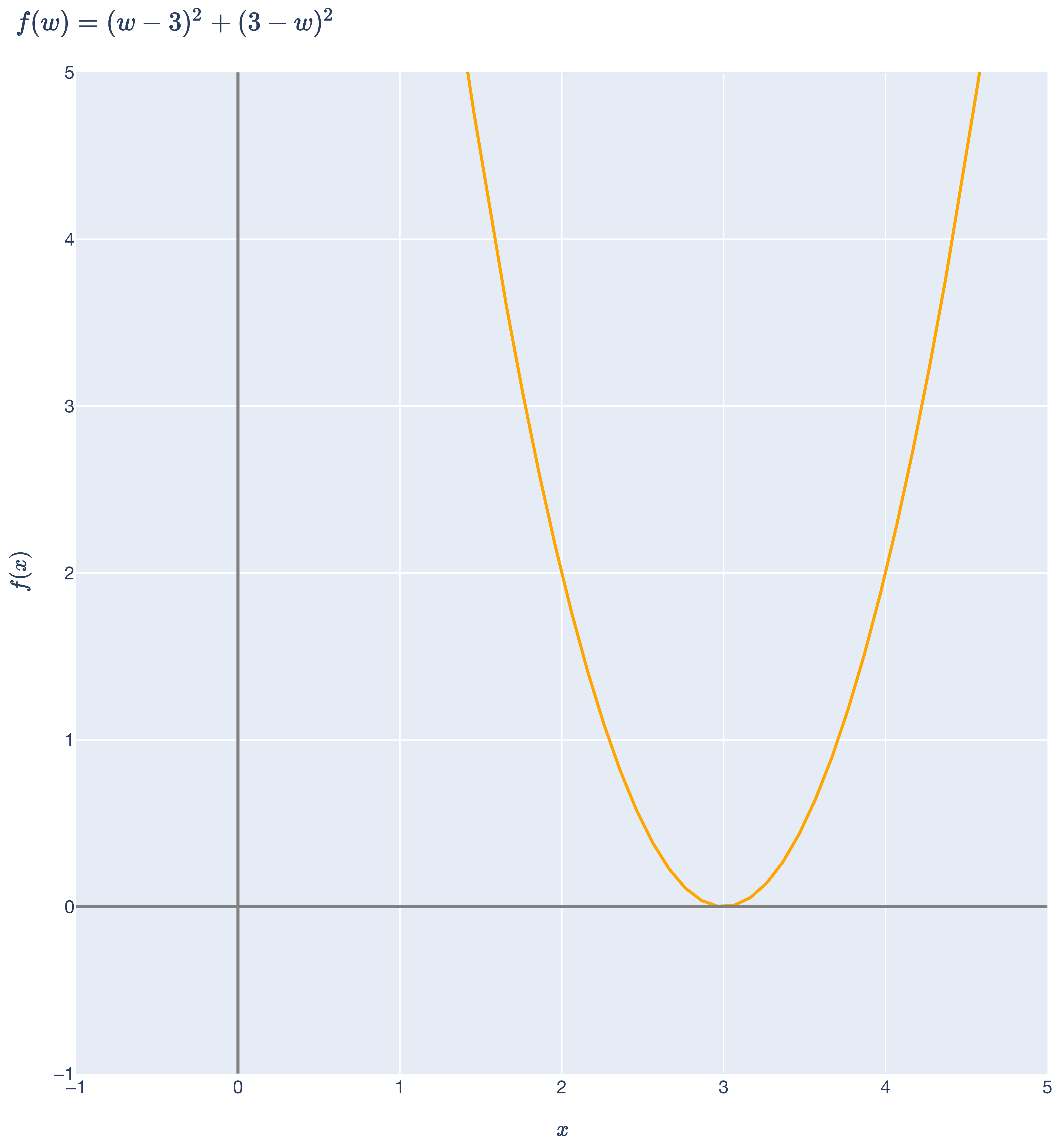
$$f(w) = \|w\mathbf{x} - \mathbf{y}\|^2$$

Least Squares

Least Squares Objective in $\mathbb{R}$

Consider the dataset $\mathbf{x} = (1, -1)$ and $\mathbf{y} = (3, -3)$, where $n = 2, d = 1$.

$$f(w) = \|w\mathbf{x} - \mathbf{y}\|^2$$



Least Squares

Least Squares Objective in $\mathbb{R}^2$

$$f : \mathbb{R}^2 \to \mathbb{R}$$

$$f(\mathbf{w}) = \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2$$

Least Squares

Least Squares Objective in $\mathbb{R}^2$

Consider the dataset $\mathbf{X} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ and $\mathbf{y} = \begin{bmatrix} -1 \\ 1 \end{bmatrix}$, where $n = 2, d = 2$.

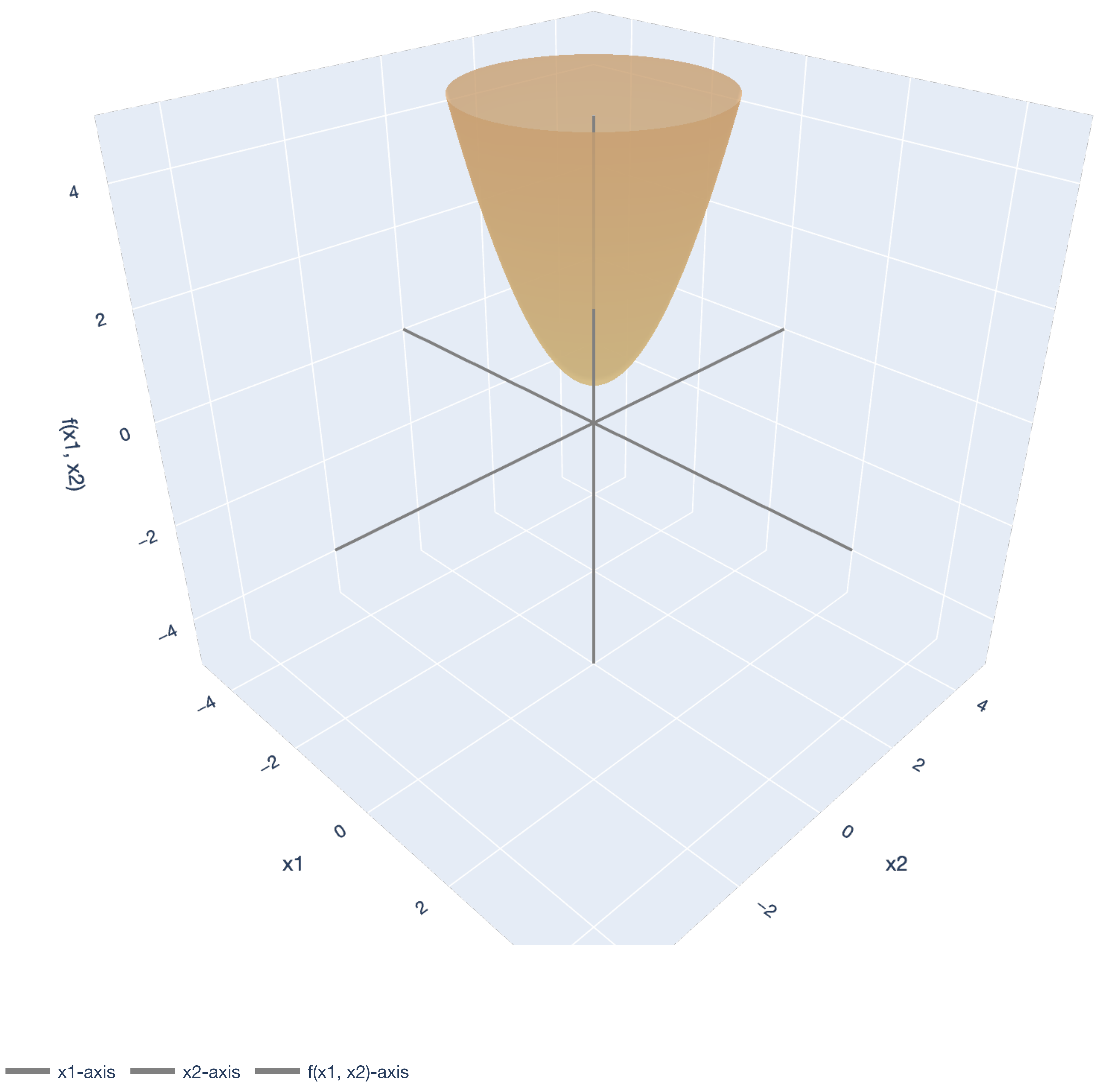
$$f(\mathbf{w}) = \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2$$

Least Squares

Least Squares Objective in $\mathbb{R}^2$

Consider the dataset $\mathbf{X} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ and $\mathbf{y} = \begin{bmatrix} -1 \\ 1 \end{bmatrix}$, where $n = 2, d = 2$.

$$f(\mathbf{w}) = \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2$$



Least Squares

Least Squares Objective in $\mathbb{R}^2$

Consider the dataset $\mathbf{X} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$ and $\mathbf{y} = \begin{bmatrix} -1 \\ 1 \end{bmatrix}$, where $n = 2, d = 2$.

$$f(\mathbf{w}) = \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2$$

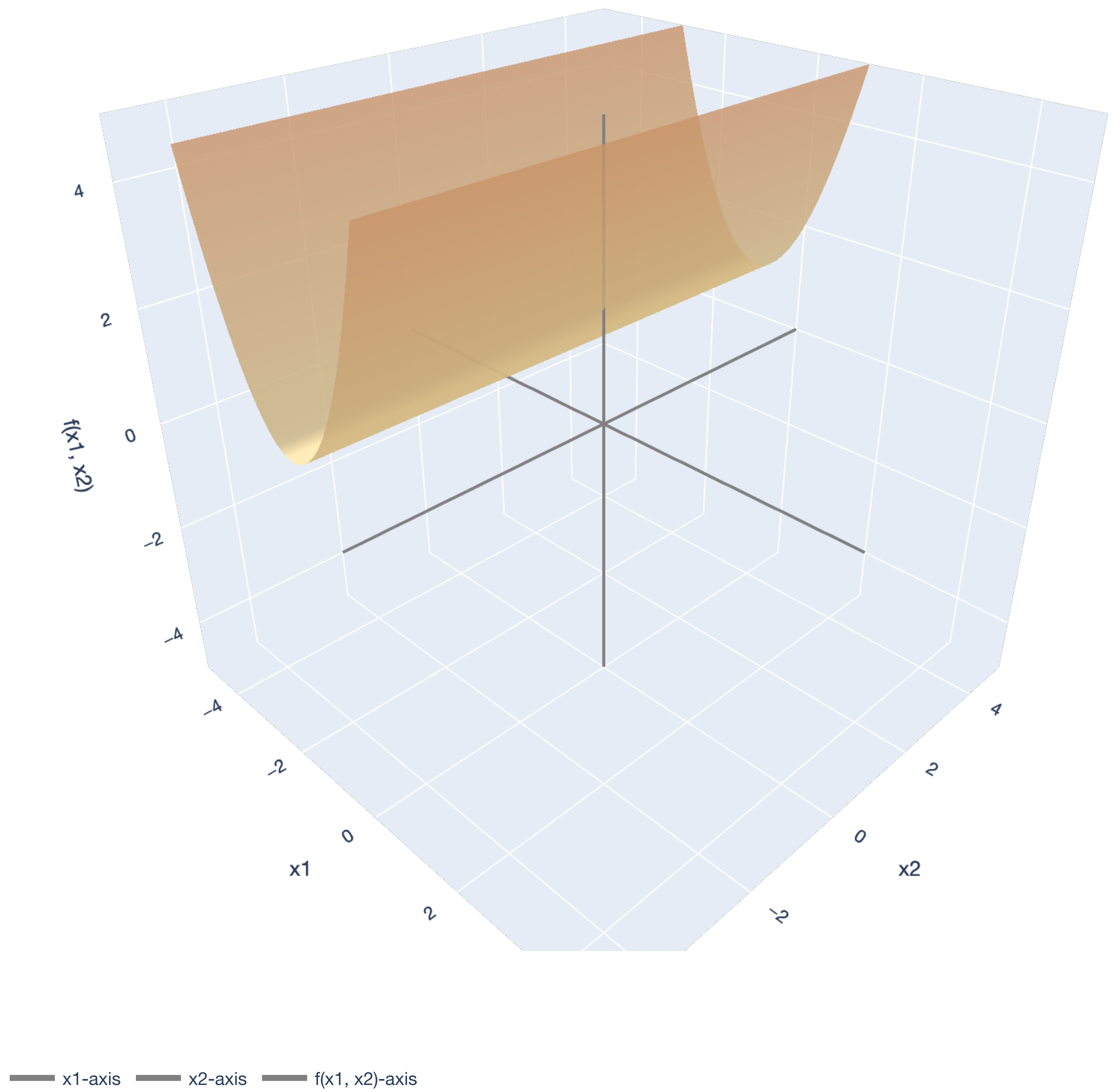
Least Squares

Least Squares Objective in $\mathbb{R}^2$

Consider the dataset $\mathbf{X} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$

and $\mathbf{y} = \begin{bmatrix} -1 \\ 1 \end{bmatrix}$, where $n = 2, d = 2$.

$$f(\mathbf{w}) = \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2$$



Least Squares

OLS from Optimization

Theorem (Ordinary Least Squares). Let $\mathbf{X} \in \mathbb{R}^{n \times d}$ and $\mathbf{y} \in \mathbb{R}^n$. Let $\hat{\mathbf{w}} \in \mathbb{R}^d$ be the least squares minimizer:

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w} \in \mathbb{R}^d} \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2$$

If $n \geq d$ and $\operatorname{rank}(\mathbf{X}) = d$, then:

$$\hat{\mathbf{w}} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}.$$

Least Squares

OLS from Optimization

Theorem (Full rank and eigenvalues). Let $\mathbf{A} \in \mathbb{R}^{d \times d}$ be a square matrix with all real eigenvalues $\lambda_1, \ldots, \lambda_d \in \mathbb{R}$.

$$\operatorname{rank}(\mathbf{A}) = d \iff \lambda_i > 0 \text{ for all } i \in [d].$$

Least Squares

Review: How did we optimize in 1D?

Recall from single variable calculus: how did we optimize a function like:

$$f(w) = 4w^2 - 4w + 1?$$

Least Squares

Review: How did we optimize in 1D?

Recall from single variable calculus: how did we optimize a function like:

$$f(w) = 4w^2 - 4w + 1?$$

First derivative test. Take the derivative $f'(w)$ and set equal to 0 to find candidates for optima, $\hat{w}$.

Least Squares

Review: How did we optimize in 1D?

Recall from single variable calculus: how did we optimize a function like:

$$f(w) = 4w^2 - 4w + 1?$$

First derivative test. Take the derivative $f'(w)$ and set equal to 0 to find candidates for optima, $\hat{w}$.

Second derivative test. Check $f''(\hat{w}) > 0$ for minimum; check $f''(\hat{w}) < 0$ for maximum.

Least Squares

OLS from Optimization

Let $\mathbf{X} \in \mathbb{R}^{n \times d}$ and $\mathbf{y} \in \mathbb{R}^n$. Consider the function $f : \mathbb{R}^d \to \mathbb{R}$,

$$f(\mathbf{w}) = \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2.$$

Least Squares

OLS from Optimization

Let $\mathbf{X} \in \mathbb{R}^{n \times d}$ and $\mathbf{y} \in \mathbb{R}^n$. Consider the function $f : \mathbb{R}^d \to \mathbb{R}$,

$$f(\mathbf{w}) = \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2.$$

Expand the squared norm:

$$\begin{aligned} f(\mathbf{w}) &= \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 \\ &= (\mathbf{X}\mathbf{w} - \mathbf{y})^\top(\mathbf{X}\mathbf{w} - \mathbf{y}) \\ &= \mathbf{w}^\top\mathbf{X}^\top\mathbf{X}\mathbf{w} - 2\mathbf{w}^\top\mathbf{X}^\top\mathbf{y} + \mathbf{y}^\top\mathbf{y} \end{aligned}$$

Quadratic Forms
Review

A function $f : \mathbb{R}^2 \to \mathbb{R}$ is a ***quadratic form*** if it is a polynomial with terms of all degree two:

$$f(x) = ax^2 + 2bxy + cy^2\,.$$

We can rewrite this in matrix form:

$$f(x, y) = \begin{bmatrix} x & y \end{bmatrix} \begin{bmatrix} a & b \\ b & c \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

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

Least Squares

OLS from Optimization

Let $\mathbf{X} \in \mathbb{R}^{n \times d}$ and $\mathbf{y} \in \mathbb{R}^n$. Consider the function $f : \mathbb{R}^d \to \mathbb{R}$,

$$f(\mathbf{w}) = \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2.$$

Expand the squared norm:

$$f(\mathbf{w}) = \mathbf{w}^\top \mathbf{X}^\top \mathbf{X}\mathbf{w} - 2\mathbf{w}^\top \mathbf{X}^\top \mathbf{y} + \mathbf{y}^\top \mathbf{y}$$

This is a quadratic function, with the quadratic form:

$$\mathbf{w}^\top \mathbf{X}^\top \mathbf{X}\mathbf{w}$$

Positive Semidefinite (PSD) Matrices

Review

A square matrix $\mathbf{A} \in \mathbb{R}^{d \times d}$ is ***positive semidefinite (PSD)*** if…

$$\text{there exists } \mathbf{X} \in \mathbb{R}^{n \times d} \text{ such that } \mathbf{A} = \mathbf{X}^\top \mathbf{X}.$$

$$\updownarrow$$

$$\text{all eigenvalues of } \mathbf{A} \text{ are nonnegative: } \lambda_1 \geq 0, \ldots, \lambda_d \geq 0.$$

$$\updownarrow$$

$$\mathbf{x}^\top \mathbf{A} \mathbf{x} \geq 0 \text{ for any } \mathbf{x} \in \mathbb{R}^d.$$

Least Squares

OLS from Optimization

Let $\mathbf{X} \in \mathbb{R}^{n \times d}$ and $\mathbf{y} \in \mathbb{R}^n$. Consider the function $f : \mathbb{R}^d \to \mathbb{R}$,

$$f(\mathbf{w}) = \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2.$$

Expand the squared norm:

$$f(\mathbf{w}) = \mathbf{w}^\top \mathbf{X}^\top \mathbf{X}\mathbf{w} - 2\mathbf{w}^\top \mathbf{X}^\top \mathbf{y} + \mathbf{y}^\top \mathbf{y}$$

This is a quadratic function, with the quadratic form:

$$\mathbf{w}^\top \mathbf{X}^\top \mathbf{X}\mathbf{w}$$

We know that $\mathbf{X}^\top \mathbf{X}$ is PSD.

Least Squares

OLS from Optimization

Let $\mathbf{X} \in \mathbb{R}^{n \times d}$ and $\mathbf{y} \in \mathbb{R}^n$. Consider the function $f : \mathbb{R}^d \to \mathbb{R}$,

$$f(\mathbf{w}) = \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2.$$

Expand the squared norm:

$$f(\mathbf{w}) = \mathbf{w}^\top \mathbf{X}^\top \mathbf{X}\mathbf{w} - 2\mathbf{w}^\top \mathbf{X}^\top \mathbf{y} + \mathbf{y}^\top \mathbf{y}$$

This is a quadratic function, with the quadratic form:

$$\mathbf{w}^\top \mathbf{X}^\top \mathbf{X}\mathbf{w}$$

Even better: $\mathrm{rank}(\mathbf{X}) = d$*, so* $\mathrm{rank}(\mathbf{X}^\top \mathbf{X}) = d$ *and therefore* $\lambda_1, \ldots, \lambda_d > 0$ *and* $\mathbf{X}^\top \mathbf{X}$ *is positive definite!*

“Matrix Calculus”

Useful identities in machine learning

$$\frac{\partial \mathbf{x}^\top \mathbf{a}}{\partial \mathbf{x}} = \mathbf{a}$$

$$\frac{\partial \mathbf{a}^\top \mathbf{x}}{\partial \mathbf{x}} = \mathbf{a}$$

$$\frac{\partial \mathbf{A}\mathbf{x}}{\partial \mathbf{x}} = \mathbf{A}$$

$$\frac{\partial \mathbf{x}^\top \mathbf{A}\mathbf{x}}{\mathbf{x}} = (\mathbf{A} + \mathbf{A}^\top)\mathbf{x}$$

More in *The Matrix Cookbook* (Petersen and Pederson, 2012).

Least Squares

OLS from Optimization

$$f(\mathbf{w}) = \mathbf{w}^\top \mathbf{X}^\top \mathbf{X}\mathbf{w} - 2\mathbf{w}^\top \mathbf{X}^\top \mathbf{y} + \mathbf{y}^\top \mathbf{y}$$

"First derivative test." Take the gradient.

$$\nabla_{\mathbf{w}} f(\mathbf{w}) = \nabla_{\mathbf{w}}(\mathbf{w}^\top \mathbf{X}^\top \mathbf{X}\mathbf{w}) - \nabla_{\mathbf{w}}(2\mathbf{w}^\top \mathbf{X}^\top \mathbf{y}) + \nabla_{\mathbf{w}} \mathbf{y}^\top \mathbf{y} \text{ (sum rule)}$$

Least Squares

OLS from Optimization

$$f(\mathbf{w}) = \mathbf{w}^\top \mathbf{X}^\top \mathbf{X}\mathbf{w} - 2\mathbf{w}^\top \mathbf{X}^\top \mathbf{y} + \mathbf{y}^\top \mathbf{y}$$

"First derivative test." Take the gradient.

$$\nabla_{\mathbf{w}} f(\mathbf{w}) = \nabla_{\mathbf{w}}(\mathbf{w}^\top \mathbf{X}^\top \mathbf{X}\mathbf{w}) - \nabla_{\mathbf{w}}(2\mathbf{w}^\top \mathbf{X}^\top \mathbf{y}) + \nabla_{\mathbf{w}} \mathbf{y}^\top \mathbf{y} \text{ (sum rule)}$$

$$\nabla_{\mathbf{w}}(\mathbf{w}^\top \mathbf{X}^\top \mathbf{X}\mathbf{w}) = 2(\mathbf{X}^\top \mathbf{X})\mathbf{w} \text{ because } \frac{\partial \mathbf{x}^\top \mathbf{A}\mathbf{x}}{\mathbf{x}} = (\mathbf{A} + \mathbf{A}^\top)\mathbf{x}$$

Least Squares

OLS from Optimization

$$f(\mathbf{w}) = \mathbf{w}^\top \mathbf{X}^\top \mathbf{X}\mathbf{w} - 2\mathbf{w}^\top \mathbf{X}^\top \mathbf{y} + \mathbf{y}^\top \mathbf{y}$$

"First derivative test." Take the gradient.

$\nabla_{\mathbf{w}} f(\mathbf{w}) = \nabla_{\mathbf{w}}(\mathbf{w}^\top \mathbf{X}^\top \mathbf{X}\mathbf{w}) - \nabla_{\mathbf{w}}(2\mathbf{w}^\top \mathbf{X}^\top \mathbf{y}) + \nabla_{\mathbf{w}} \mathbf{y}^\top \mathbf{y}$ (sum rule)

$\nabla_{\mathbf{w}}(\mathbf{w}^\top \mathbf{X}^\top \mathbf{X}\mathbf{w}) = 2(\mathbf{X}^\top \mathbf{X})\mathbf{w}$ because $\dfrac{\partial \mathbf{x}^\top \mathbf{A}\mathbf{x}}{\mathbf{x}} = (\mathbf{A} + \mathbf{A}^\top)\mathbf{x}$

$\nabla_{\mathbf{w}}(2\mathbf{w}^\top \mathbf{X}^\top \mathbf{y}) = 2\mathbf{X}^\top \mathbf{y}$ because $\dfrac{\partial \mathbf{a}^\top \mathbf{x}}{\partial \mathbf{x}} = \mathbf{a}$

Least Squares

OLS from Optimization

$$f(\mathbf{w}) = \mathbf{w}^\top \mathbf{X}^\top \mathbf{X}\mathbf{w} - 2\mathbf{w}^\top \mathbf{X}^\top \mathbf{y} + \mathbf{y}^\top \mathbf{y}$$

"First derivative test." Take the gradient.

$\nabla_{\mathbf{w}} f(\mathbf{w}) = \nabla_{\mathbf{w}}(\mathbf{w}^\top \mathbf{X}^\top \mathbf{X}\mathbf{w}) - \nabla_{\mathbf{w}}(2\mathbf{w}^\top \mathbf{X}^\top \mathbf{y}) + \nabla_{\mathbf{w}} \mathbf{y}^\top \mathbf{y}$ (sum rule)

$\nabla_{\mathbf{w}}(\mathbf{w}^\top \mathbf{X}^\top \mathbf{X}\mathbf{w}) = 2(\mathbf{X}^\top \mathbf{X})\mathbf{w}$ because $\dfrac{\partial \mathbf{x}^\top \mathbf{A}\mathbf{x}}{\mathbf{x}} = (\mathbf{A} + \mathbf{A}^\top)\mathbf{x}$

$\nabla_{\mathbf{w}}(2\mathbf{w}^\top \mathbf{X}^\top \mathbf{y}) = 2\mathbf{X}^\top \mathbf{y}$ because $\dfrac{\partial \mathbf{a}^\top \mathbf{x}}{\partial \mathbf{x}} = \mathbf{a}$

$\nabla_{\mathbf{w}} \mathbf{y}^\top \mathbf{y} = 0$

Least Squares

OLS from Optimization

$$f(\mathbf{w}) = \mathbf{w}^\top \mathbf{X}^\top \mathbf{X}\mathbf{w} - 2\mathbf{w}^\top \mathbf{X}^\top \mathbf{y} + \mathbf{y}^\top \mathbf{y}$$

"First derivative test." Take the gradient.

$\nabla_{\mathbf{w}} f(\mathbf{w}) = \nabla_{\mathbf{w}}(\mathbf{w}^\top \mathbf{X}^\top \mathbf{X}\mathbf{w}) - \nabla_{\mathbf{w}}(2\mathbf{w}^\top \mathbf{X}^\top \mathbf{y}) + \nabla_{\mathbf{w}} \mathbf{y}^\top \mathbf{y}$ (sum rule)

$\nabla_{\mathbf{w}}(\mathbf{w}^\top \mathbf{X}^\top \mathbf{X}\mathbf{w}) = 2(\mathbf{X}^\top \mathbf{X})\mathbf{w}$ because $\dfrac{\partial \mathbf{x}^\top \mathbf{A}\mathbf{x}}{\mathbf{x}} = (\mathbf{A} + \mathbf{A}^\top)\mathbf{x}$

$\nabla_{\mathbf{w}}(2\mathbf{w}^\top \mathbf{X}^\top \mathbf{y}) = 2\mathbf{X}^\top \mathbf{y}$ because $\dfrac{\partial \mathbf{a}^\top \mathbf{x}}{\partial \mathbf{x}} = \mathbf{a}$

$\nabla_{\mathbf{w}} \mathbf{y}^\top \mathbf{y} = 0$

$$\implies \nabla_{\mathbf{w}} f(\mathbf{w}) = 2(\mathbf{X}^\top \mathbf{X})\mathbf{w} - 2\mathbf{X}^\top \mathbf{y}$$

Least Squares

OLS from Optimization

$$f(\mathbf{w}) = \mathbf{w}^\top \mathbf{X}^\top \mathbf{X}\mathbf{w} - 2\mathbf{w}^\top \mathbf{X}^\top \mathbf{y} + \mathbf{y}^\top \mathbf{y}$$

"First derivative test." Take the gradient.

$$\nabla_{\mathbf{w}} f(\mathbf{w}) = 2(\mathbf{X}^\top \mathbf{X})\mathbf{w} - 2\mathbf{X}^\top \mathbf{y}.$$

Set it equal to $\mathbf{0}$.

$$2(\mathbf{X}^\top \mathbf{X})\mathbf{w} - 2\mathbf{X}^\top \mathbf{y} = \mathbf{0} \implies \mathbf{X}^\top \mathbf{X}\mathbf{w} = \mathbf{X}^\top \mathbf{y}$$

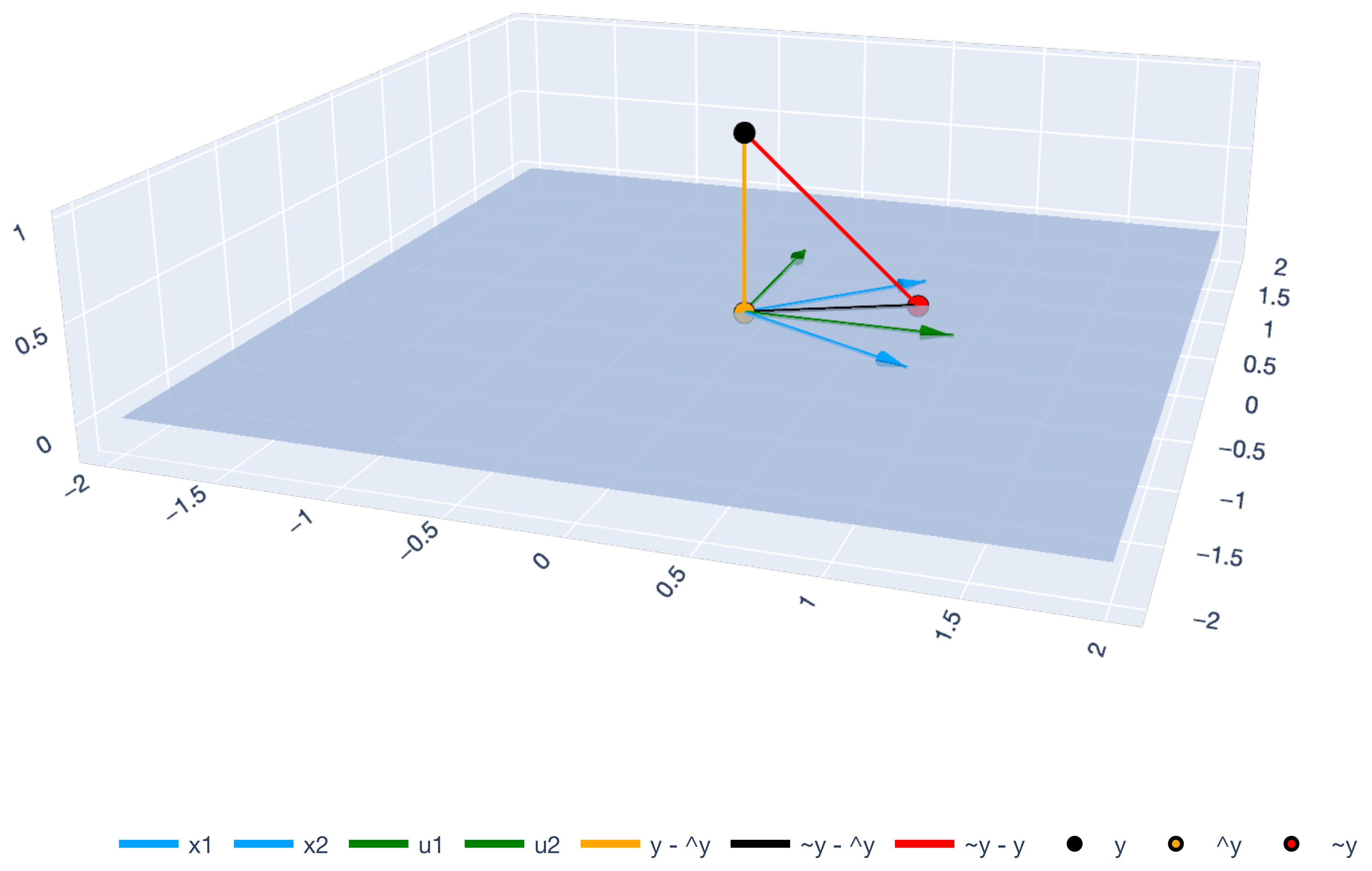
We have again obtained the ***normal equations***!

Least Squares

Obtaining normal equations from linear algebra

Because $\hat{\mathbf{y}} - \mathbf{y}$ is perpendicular to $\mathrm{span}(\mathrm{col}(\mathbf{X}))$, we obtain the *normal equations:*

$$\mathbf{X}^\top \mathbf{X} \hat{\mathbf{w}} = \mathbf{X}^\top \mathbf{y}.$$



Least Squares

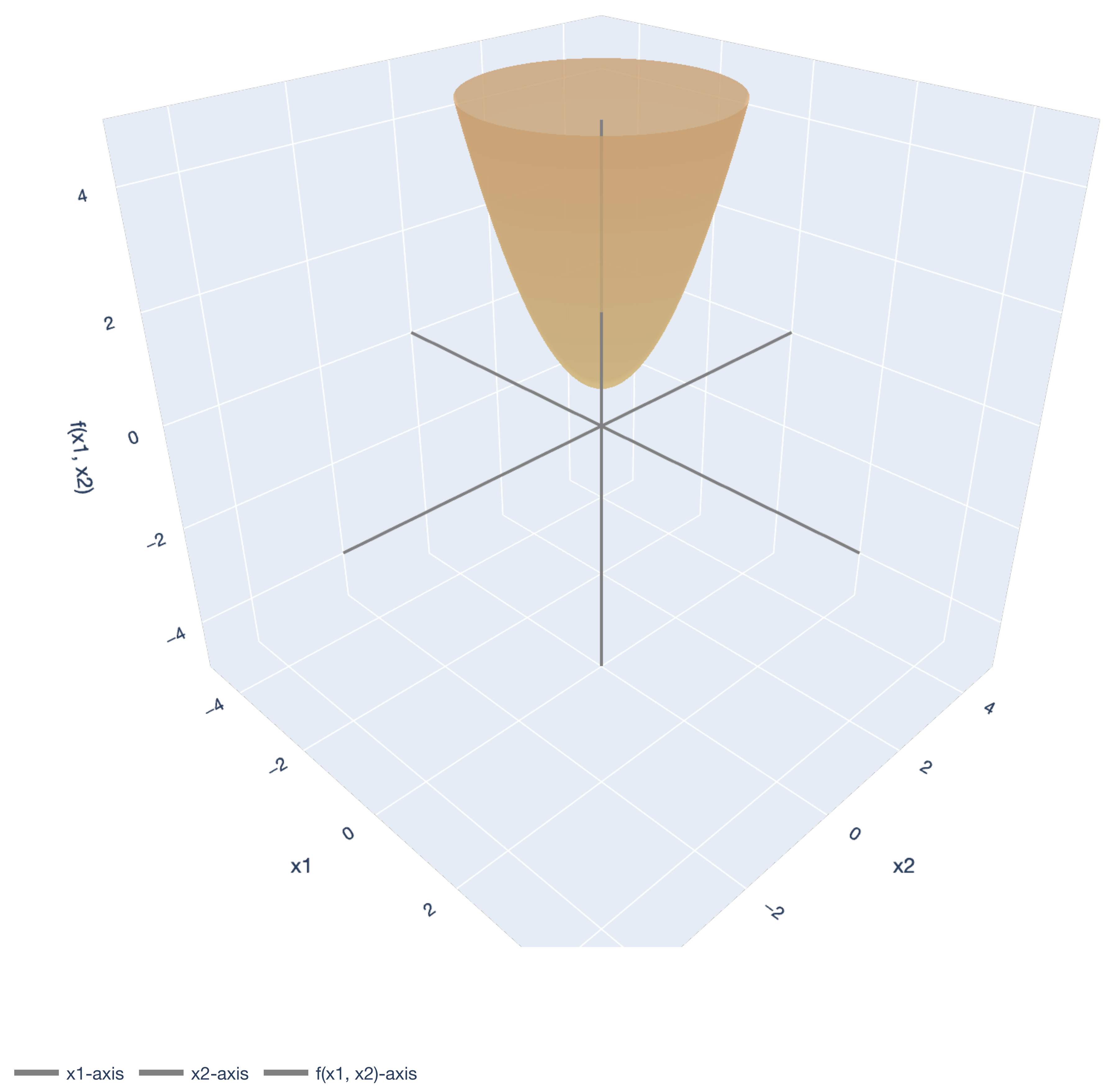
Obtaining normal equations from optimization

Because the gradient is

$$\nabla_{\mathbf{w}} f(\mathbf{w}) = 2(\mathbf{X}^\top \mathbf{X})\mathbf{w} - 2\mathbf{X}^\top \mathbf{y},$$

setting it equal to $\mathbf{0}$, we obtain the *normal equations*:

$$\mathbf{X}^\top \mathbf{X}\hat{\mathbf{w}} = \mathbf{X}^\top \mathbf{y}.$$



Least Squares

OLS from Optimization

$$f(\mathbf{w}) = \mathbf{w}^\top \mathbf{X}^\top \mathbf{X}\mathbf{w} - 2\mathbf{w}^\top \mathbf{X}^\top \mathbf{y} + \mathbf{y}^\top \mathbf{y}$$

"First derivative test." Take the gradient.

$$\nabla_{\mathbf{w}} f(\mathbf{w}) = 2(\mathbf{X}^\top \mathbf{X})\mathbf{w} - 2\mathbf{X}^\top \mathbf{y}.$$

Set it equal to $\mathbf{0}$.

$$2(\mathbf{X}^\top \mathbf{X})\mathbf{w} - 2\mathbf{X}^\top \mathbf{y} = \mathbf{0} \implies \mathbf{X}^\top \mathbf{X}\mathbf{w} = \mathbf{X}^\top \mathbf{y}$$

Because $\mathrm{rank}(\mathbf{X}) = d$, we know $\mathrm{rank}(\mathbf{X}^\top \mathbf{X}) = d$ and $\mathbf{X}^\top \mathbf{X}$ is invertible. Solve the normal equations to get a *candidate* for the minimizer:

$$\hat{\mathbf{w}} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}.$$

Least Squares

OLS from Optimization

Objective: $f(\mathbf{w}) = \mathbf{w}^\top \mathbf{X}^\top \mathbf{X}\mathbf{w} - 2\mathbf{w}^\top \mathbf{X}^\top \mathbf{y} + \mathbf{y}^\top \mathbf{y}$

Gradient: $\nabla_{\mathbf{w}} f(\mathbf{w}) = 2(\mathbf{X}^\top \mathbf{X})\mathbf{w} - 2\mathbf{X}^\top \mathbf{y}.$

Candidate minimizer: $\hat{\mathbf{w}} = (\mathbf{X}^\top \mathbf{X})^{-1}\mathbf{X}^\top \mathbf{y}.$

Least Squares

OLS from Optimization

Objective: $f(\mathbf{w}) = \mathbf{w}^\top \mathbf{X}^\top \mathbf{X}\mathbf{w} - 2\mathbf{w}^\top \mathbf{X}^\top \mathbf{y} + \mathbf{y}^\top \mathbf{y}$

Gradient: $\nabla_{\mathbf{w}} f(\mathbf{w}) = 2(\mathbf{X}^\top \mathbf{X})\mathbf{w} - 2\mathbf{X}^\top \mathbf{y}.$

Candidate minimizer: $\hat{\mathbf{w}} = (\mathbf{X}^\top \mathbf{X})^{-1}\mathbf{X}^\top \mathbf{y}.$

"Second derivative test." Take the *Hessian* of $f(\mathbf{w})$.

$$\nabla^2_{\mathbf{w}} f(\mathbf{w}) = 2\mathbf{X}^\top \mathbf{X}.$$

Least Squares

OLS from Optimization

Objective: $f(\mathbf{w}) = \mathbf{w}^\top \mathbf{X}^\top \mathbf{X}\mathbf{w} - 2\mathbf{w}^\top \mathbf{X}^\top \mathbf{y} + \mathbf{y}^\top \mathbf{y}$

Gradient: $\nabla_{\mathbf{w}} f(\mathbf{w}) = 2(\mathbf{X}^\top \mathbf{X})\mathbf{w} - 2\mathbf{X}^\top \mathbf{y}.$

Candidate minimizer: $\hat{\mathbf{w}} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}.$

"Second derivative test." Take the *Hessian* of $f(\mathbf{w})$.

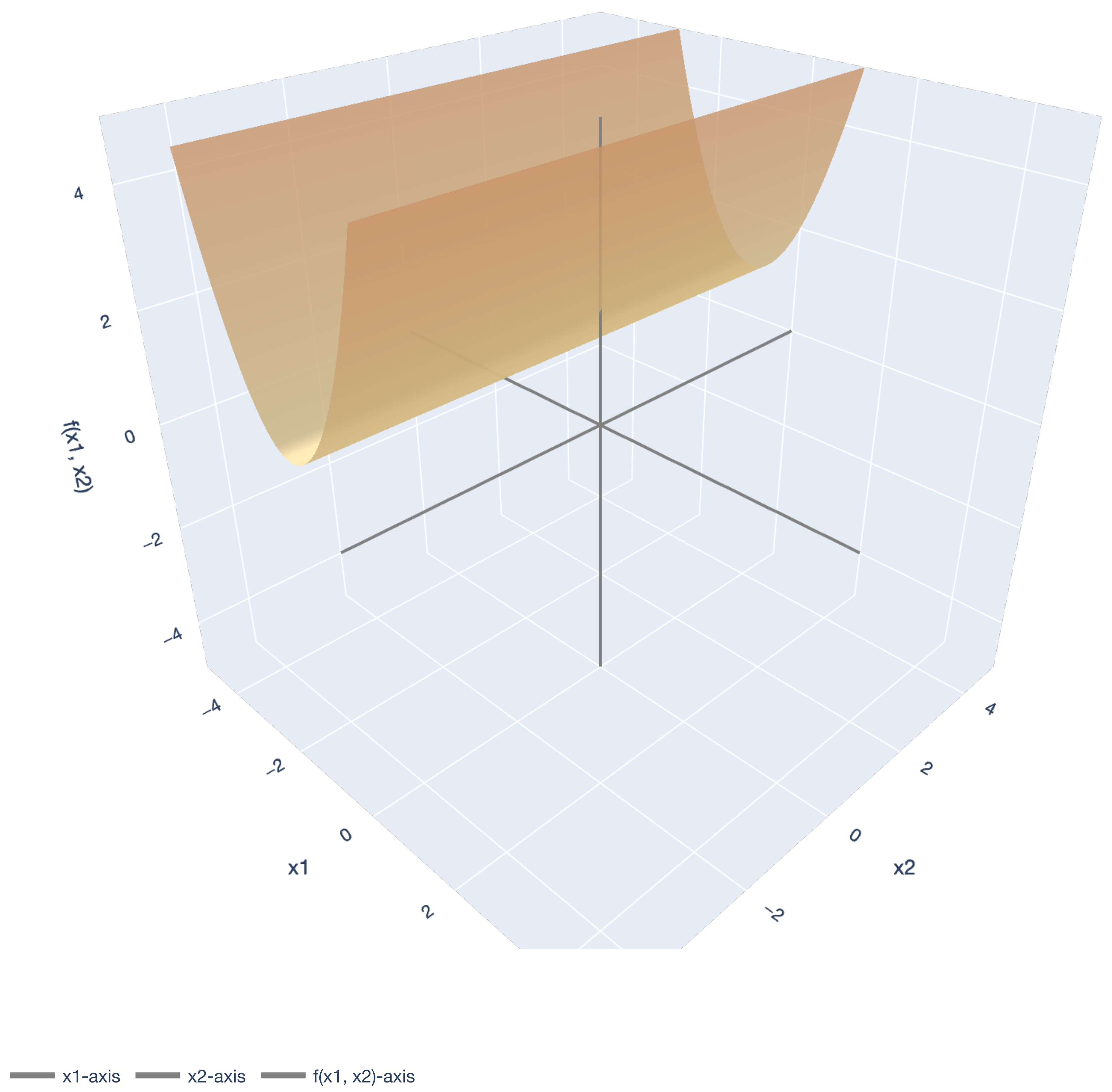
$$\nabla^2_{\mathbf{w}} f(\mathbf{w}) = 2\mathbf{X}^\top \mathbf{X}.$$

$$\mathrm{rank}(\mathbf{X}) = d \implies \mathrm{rank}(\mathbf{X}^\top \mathbf{X}) = d \implies \lambda_1, \ldots, \lambda_d > 0$$

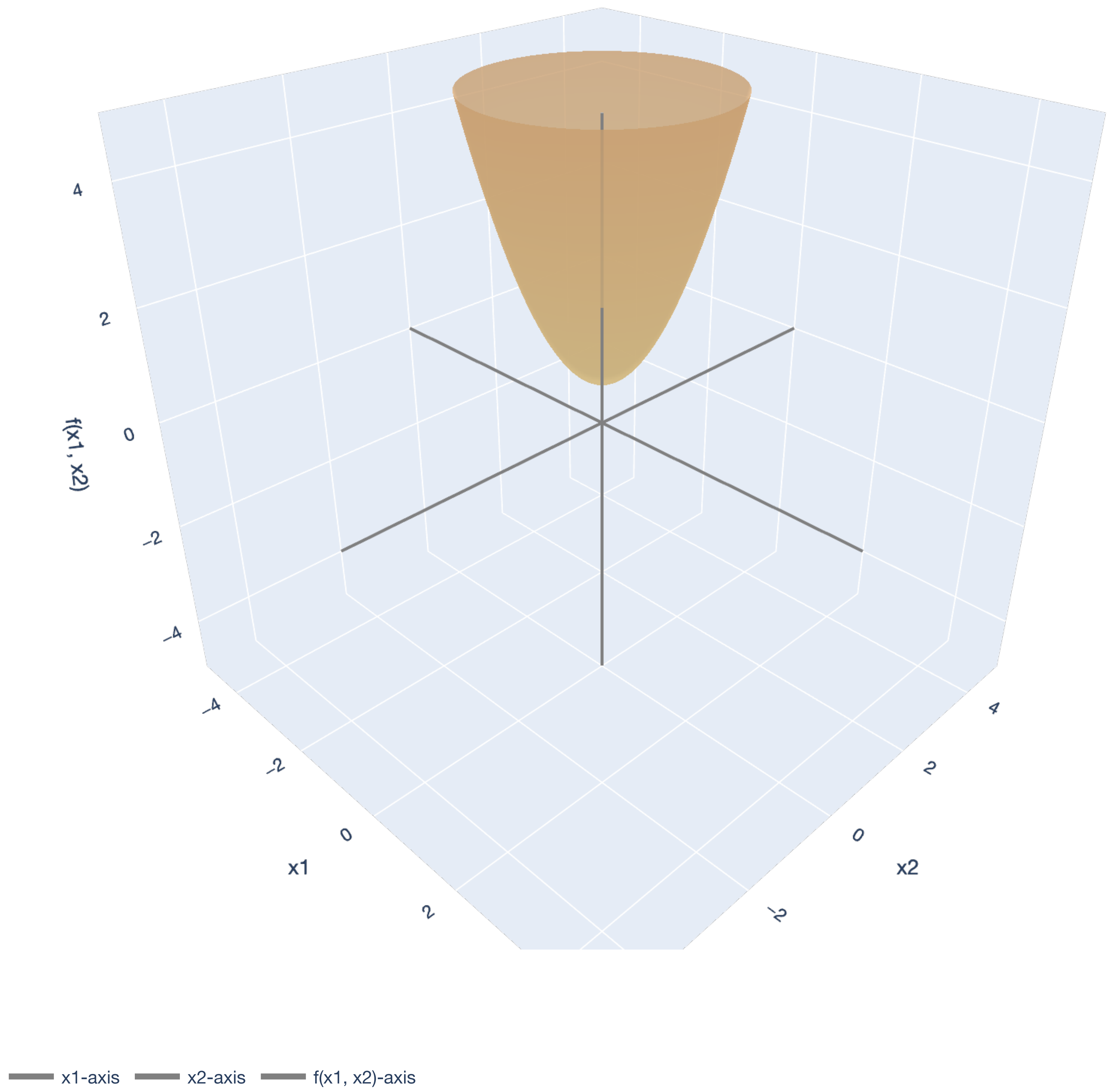
$\implies$ $\mathbf{X}^\top \mathbf{X}$ is positive definite!

PSD and PD Quadratic Forms

“Proof by graph”



$$\lambda_1, \ldots, \lambda_d \geq 0$$



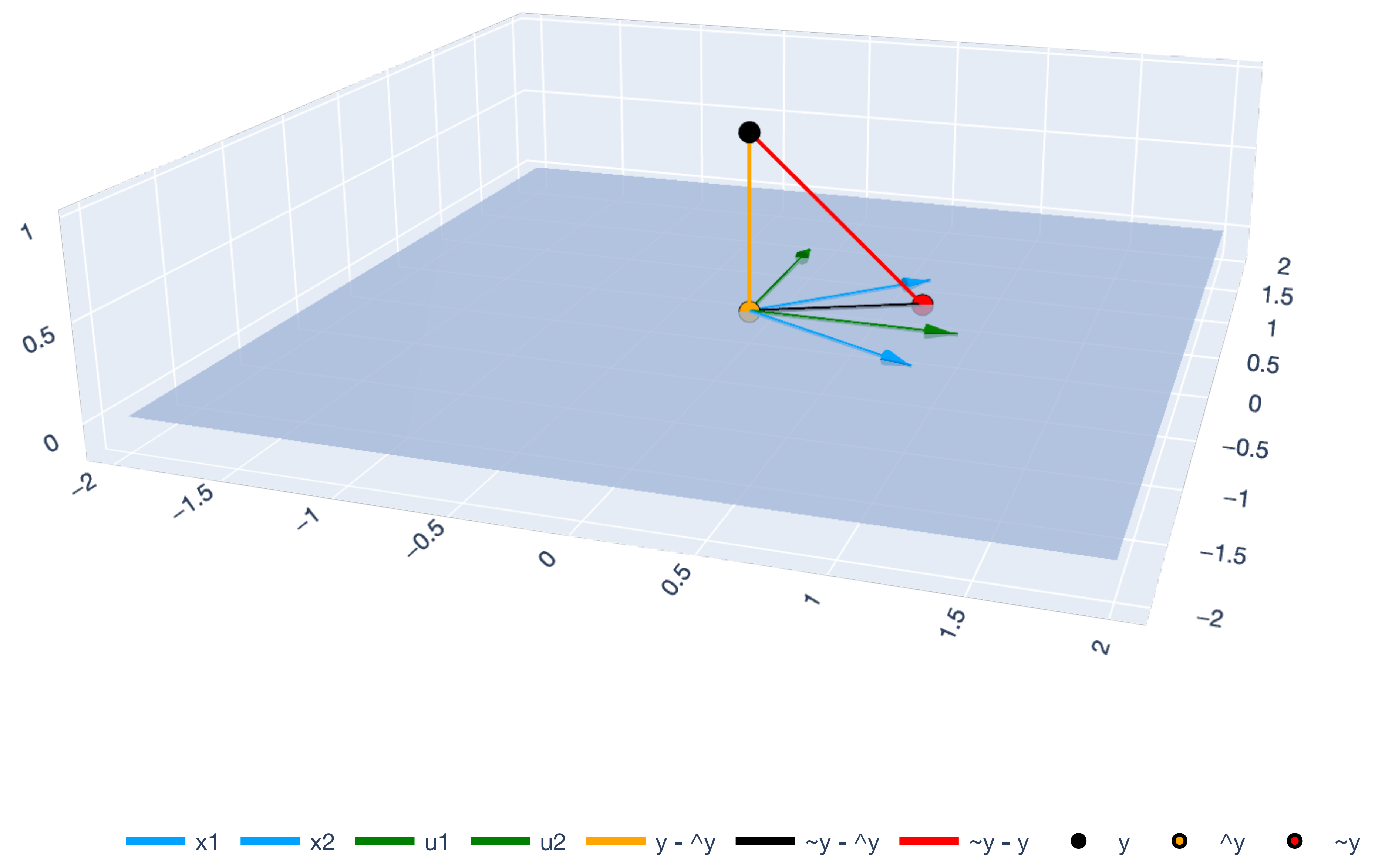
$$\lambda_1, \ldots, \lambda_d > 0$$

Least Squares

Showing $\hat{\mathbf{w}}$ is the minimizer from linear algebra

By Pythagorean Theorem, any other vector $\tilde{\mathbf{y}} \in \text{span}(\text{col}(\mathbf{X}))$ gives a larger error:

$$\|\hat{\mathbf{y}} - \mathbf{y}\|^2 \leq \|\tilde{\mathbf{y}} - \mathbf{y}\|^2.$$



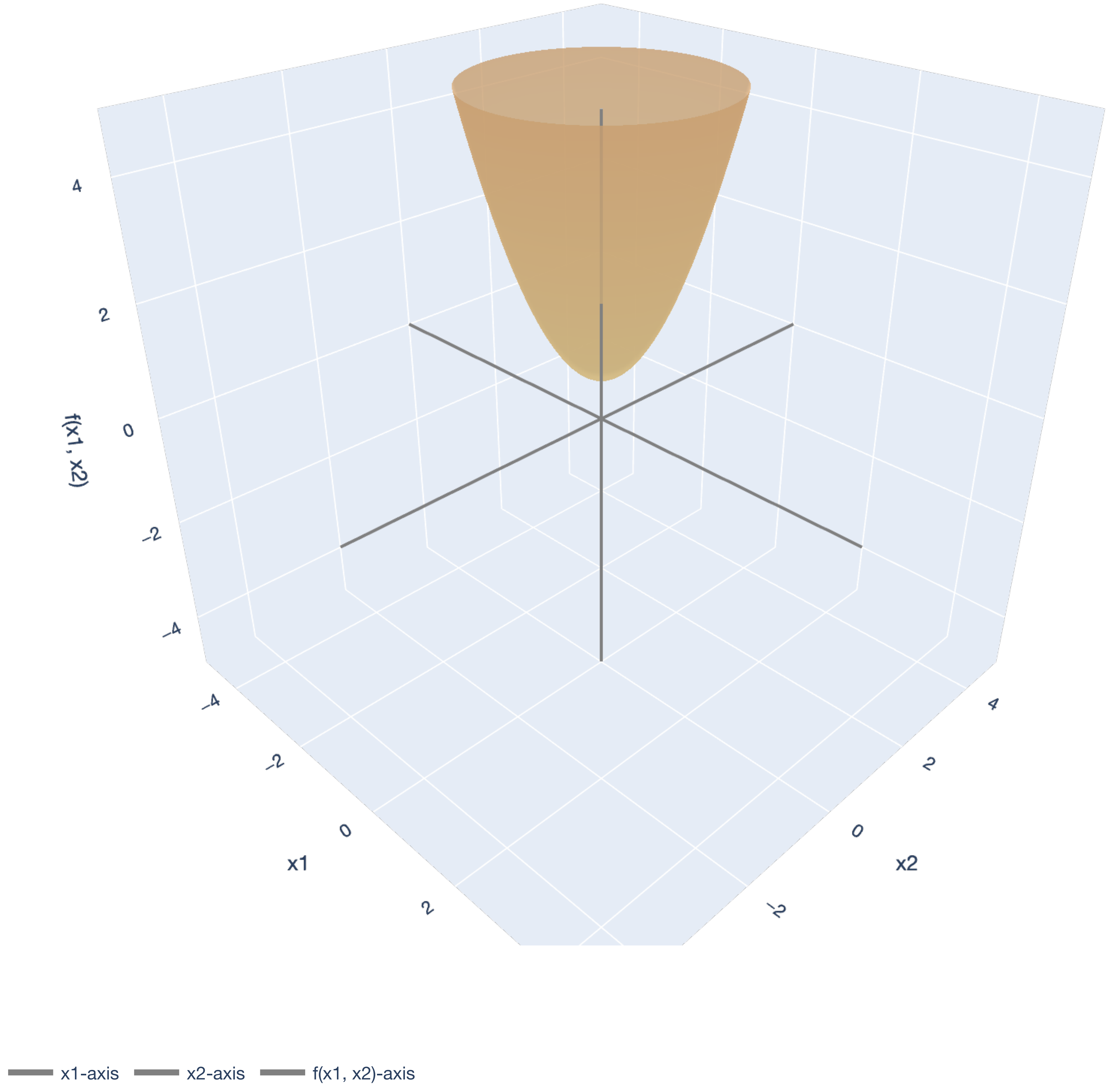
Least Squares

Showing $\hat{\mathbf{w}}$ is the minimizer from optimization

Because the Hessian of $f(\mathbf{w})$ is

$$\nabla^2_{\mathbf{w}} f(\mathbf{w}) = 2\mathbf{X}^\top\mathbf{X},$$

and we assumed $\mathrm{rank}(\mathbf{X}) = d$, the matrix $\mathbf{X}^\top\mathbf{X}$ must be positive definite, and $f(\mathbf{w})$ therefore has a “positive” second derivative (Hessian).



Least Squares

OLS Theorem

<u>Theorem (Ordinary Least Squares).</u> Let $\mathbf{X} \in \mathbb{R}^{n \times d}$ and $\mathbf{y} \in \mathbb{R}^n$. Let $\hat{\mathbf{w}} \in \mathbb{R}^d$ be the least squares minimizer:

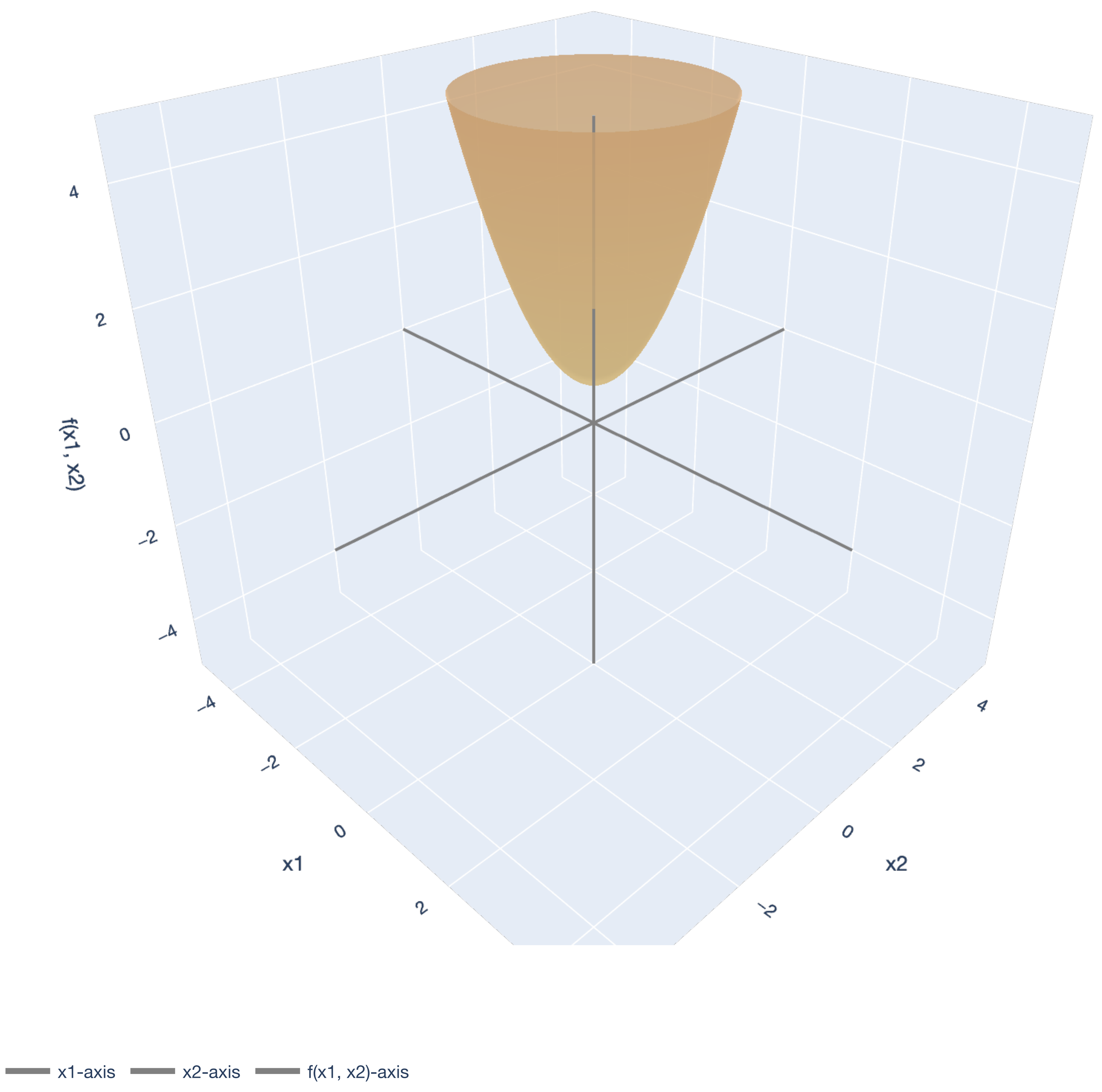
$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w} \in \mathbb{R}^d} \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2$$

If $n \geq d$ and $\mathrm{rank}(\mathbf{X}) = d$, then:

$$\hat{\mathbf{w}} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}.$$

To get predictions $\hat{\mathbf{y}} \in \mathbb{R}^n$:

$$\hat{\mathbf{y}} = \mathbf{X}\hat{\mathbf{w}} = \mathbf{X}(\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}.$$



Gradient Descent

Preview of the Algorithm

Multivariable Differentiation

Gradient as direction of steepest ascent

Theorem (Gradient and direction of steepest ascent). Let $f : \mathbb{R}^d \to \mathbb{R}$ be differentiable at $\mathbf{x}_0 \in \mathbb{R}^d$. If $\mathbf{v} \in \mathbb{R}^d$ is a *unit* vector making angle θ with the gradient $\nabla f(\mathbf{x}_0)$, then:

$$\nabla f(\mathbf{x}_0)^\top \mathbf{v} = \|\nabla f(\mathbf{x}_0)\| \cos \theta.$$

Gradient is the direction of *steepest ascent* at the rate $\|\nabla f(\mathbf{x}_0)\|$!

Gradient Descent

Algorithm

Input: Function $f : \mathbb{R}^n \to \mathbb{R}$. Initial point $\mathbf{x}_0 \in \mathbb{R}^n$. Step size $\eta \in \mathbb{R}$.

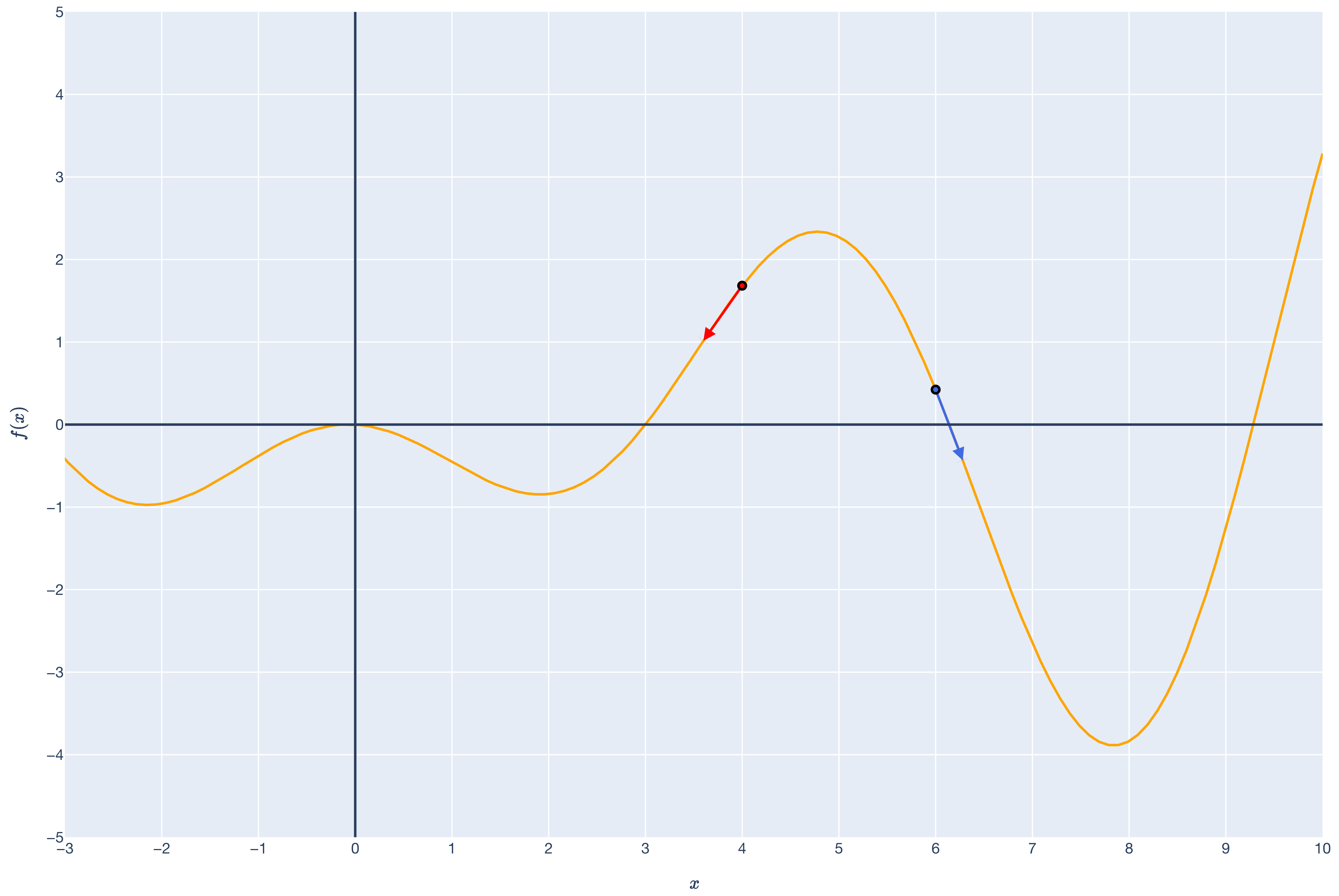
For $t = 1,2,3,\ldots$

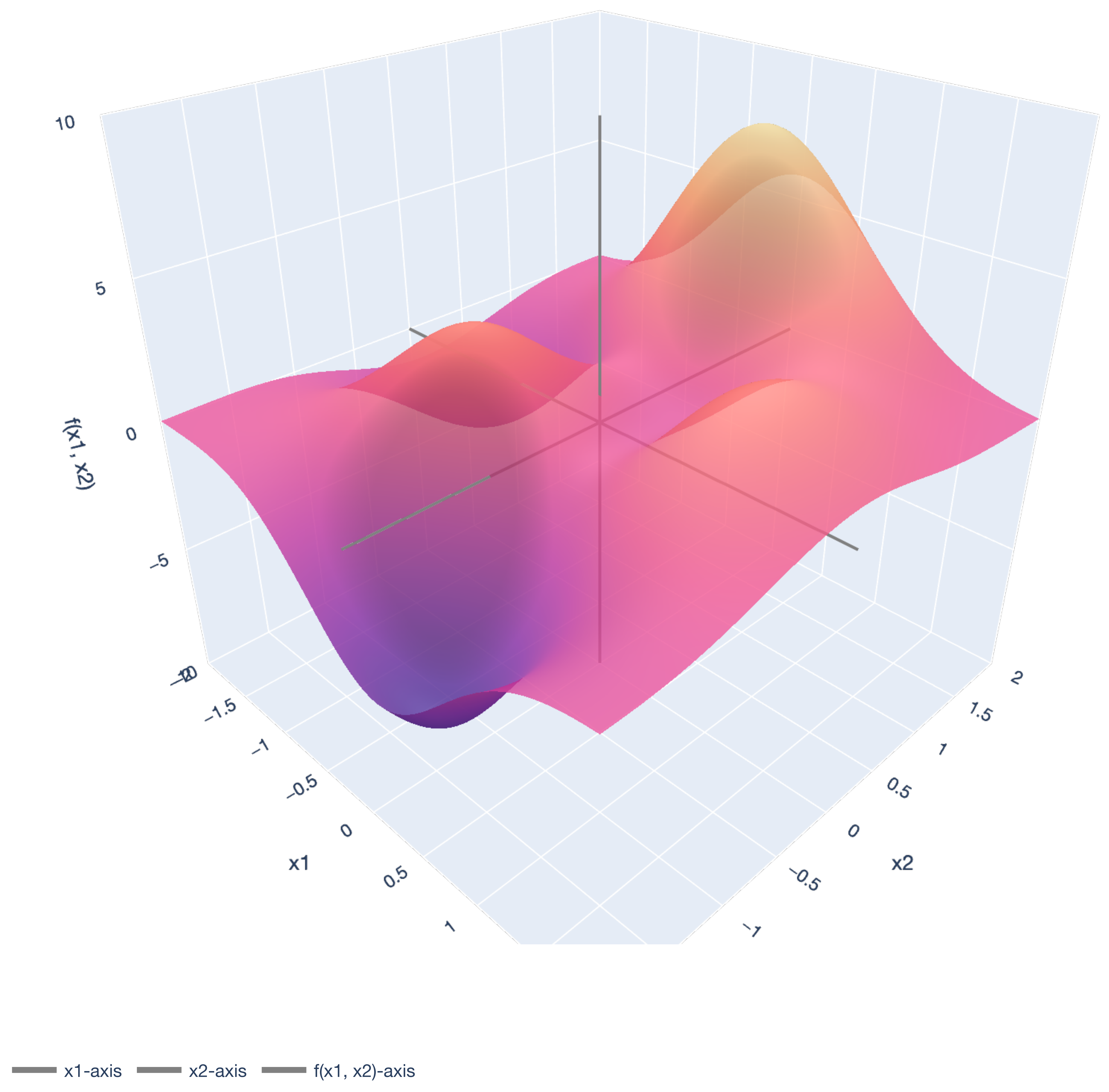
Compute: $\mathbf{x}_t \leftarrow \mathbf{x}_{t-1} - \eta \nabla f(\mathbf{x}_{t-1})$.

If $\nabla f(\mathbf{x}_t) = 0$ or $\mathbf{x}_t - \mathbf{x}_{t-1}$ is sufficiently small, then **return** $f(\mathbf{x}_t)$.

Gradient Descent

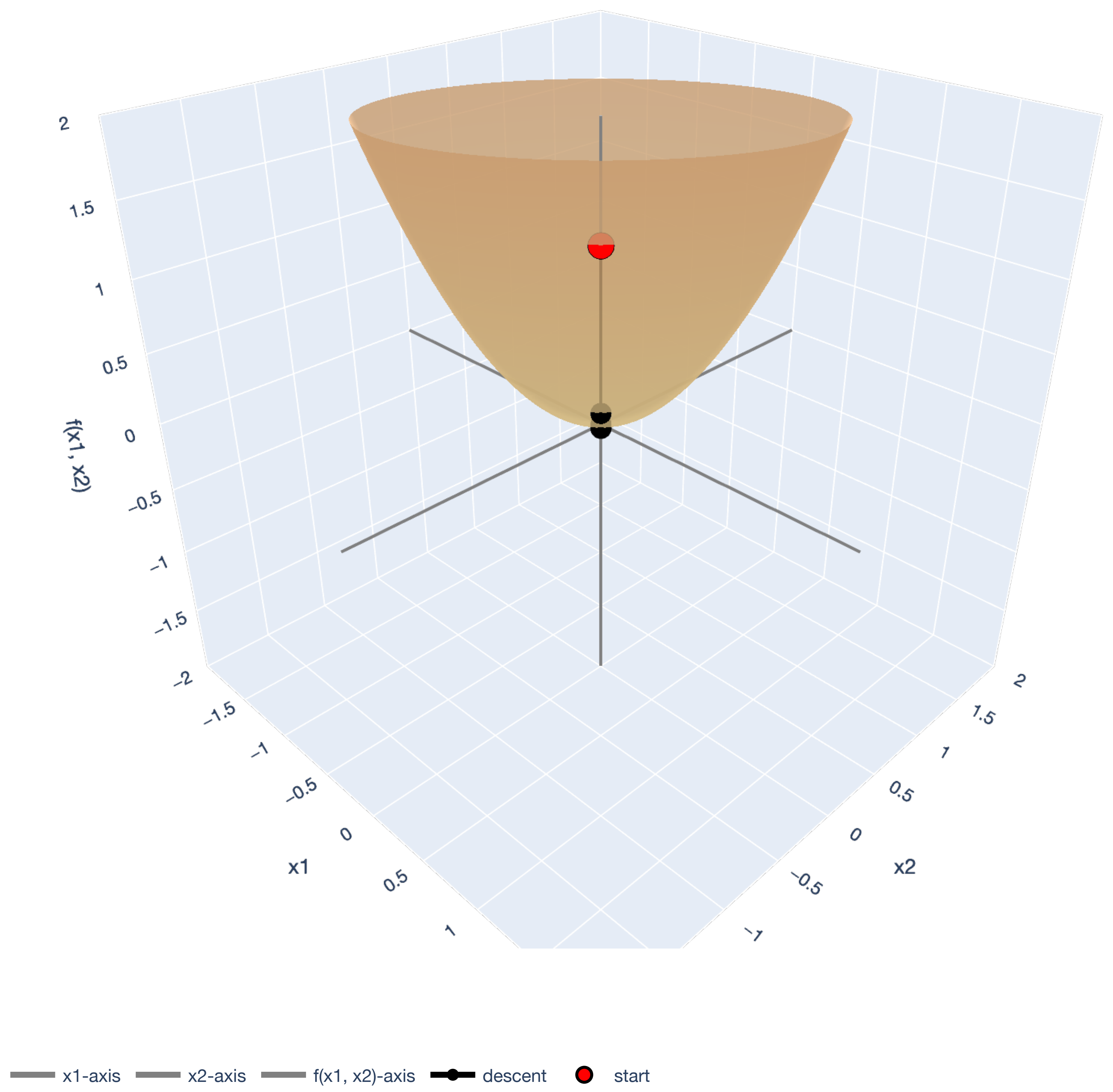
Preview

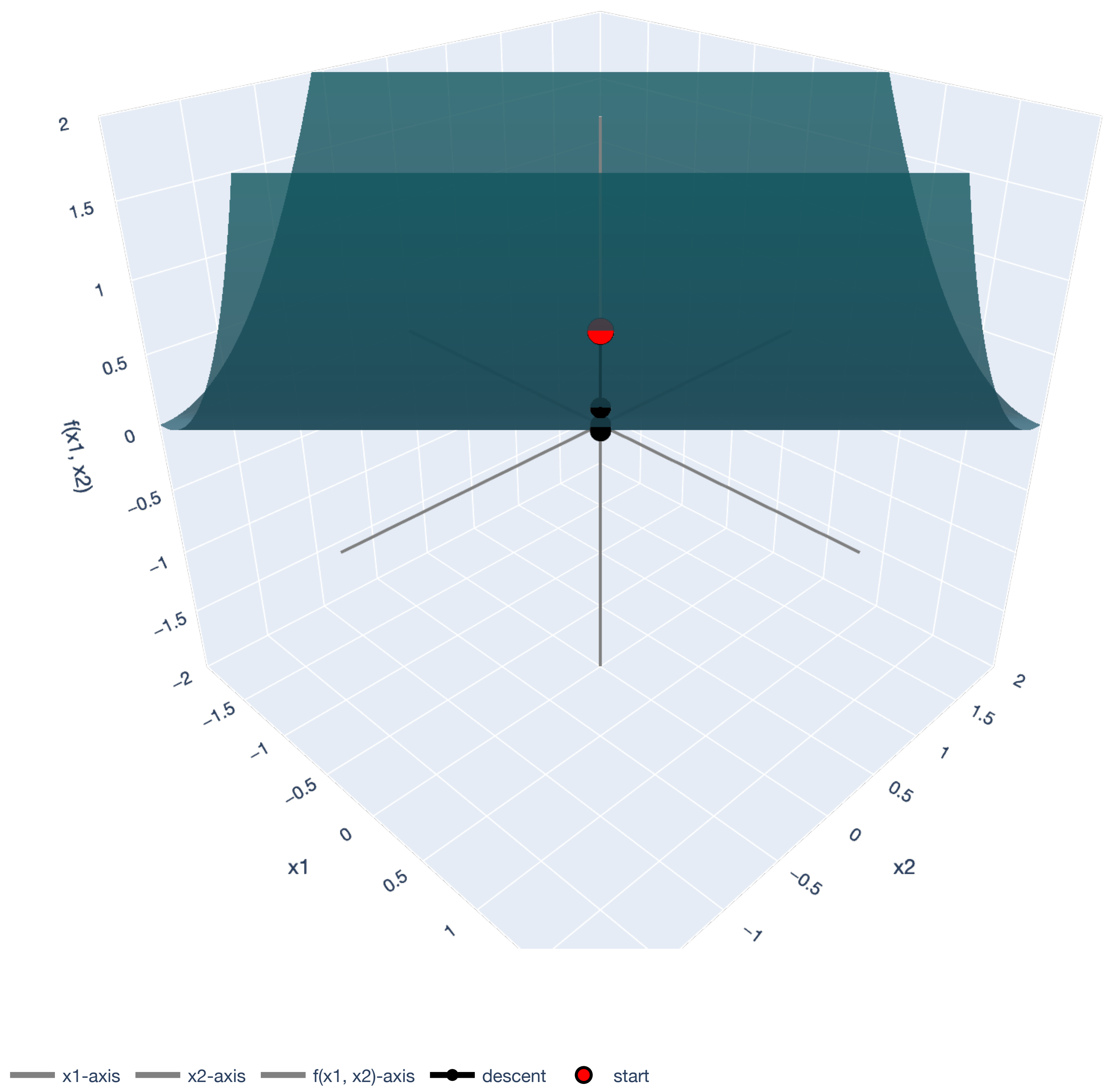


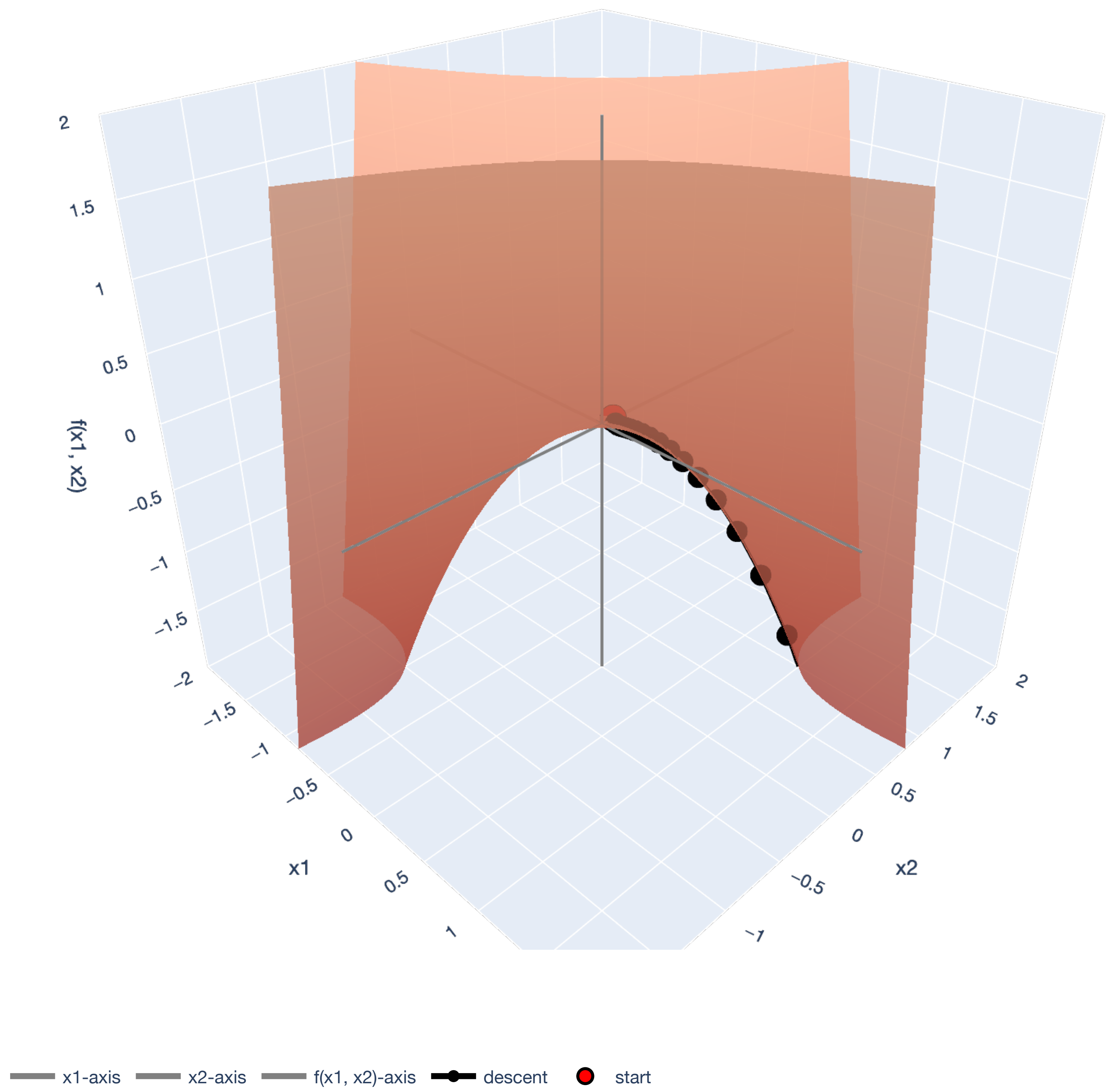


Gradient Descent

Preview



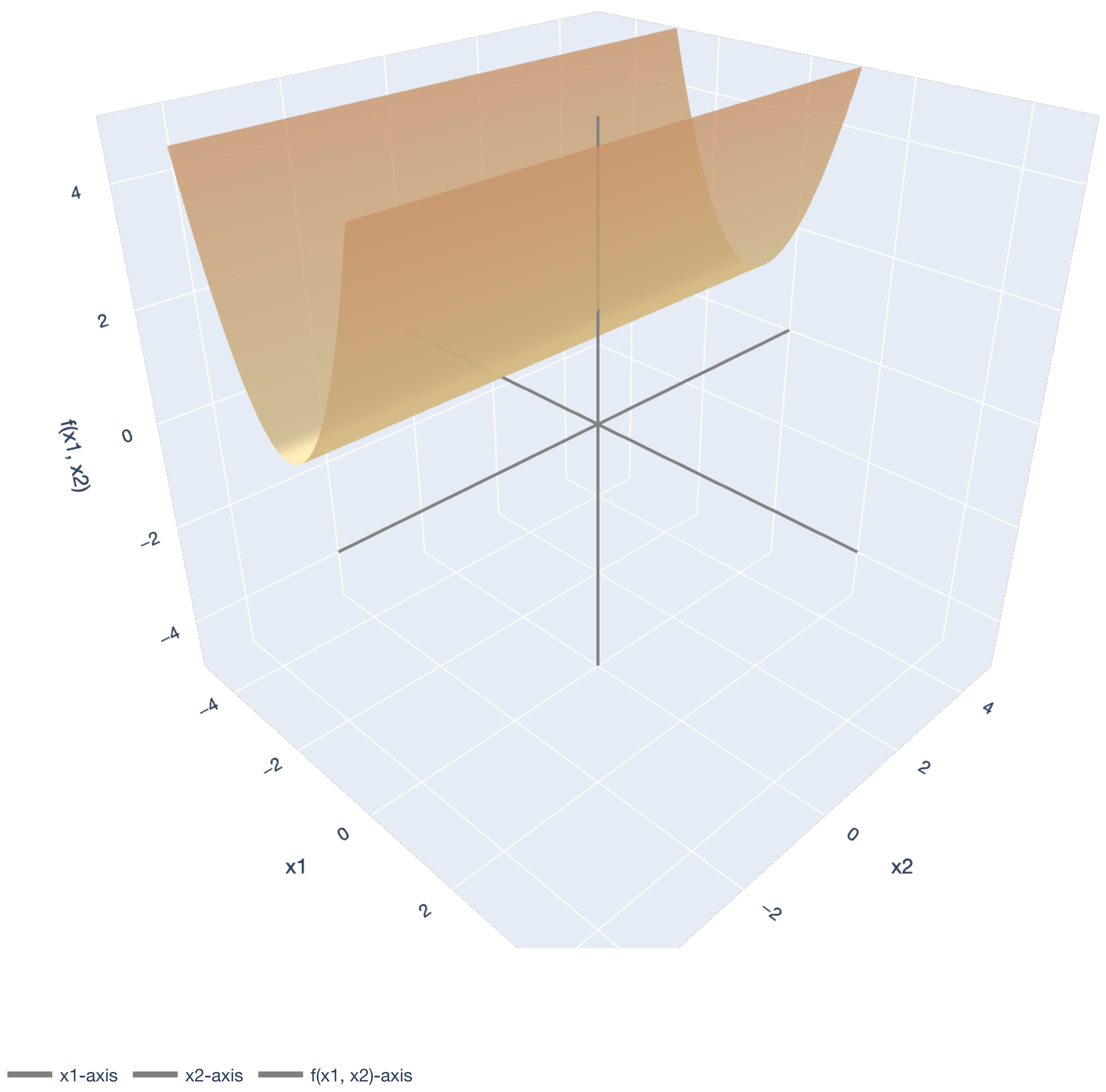




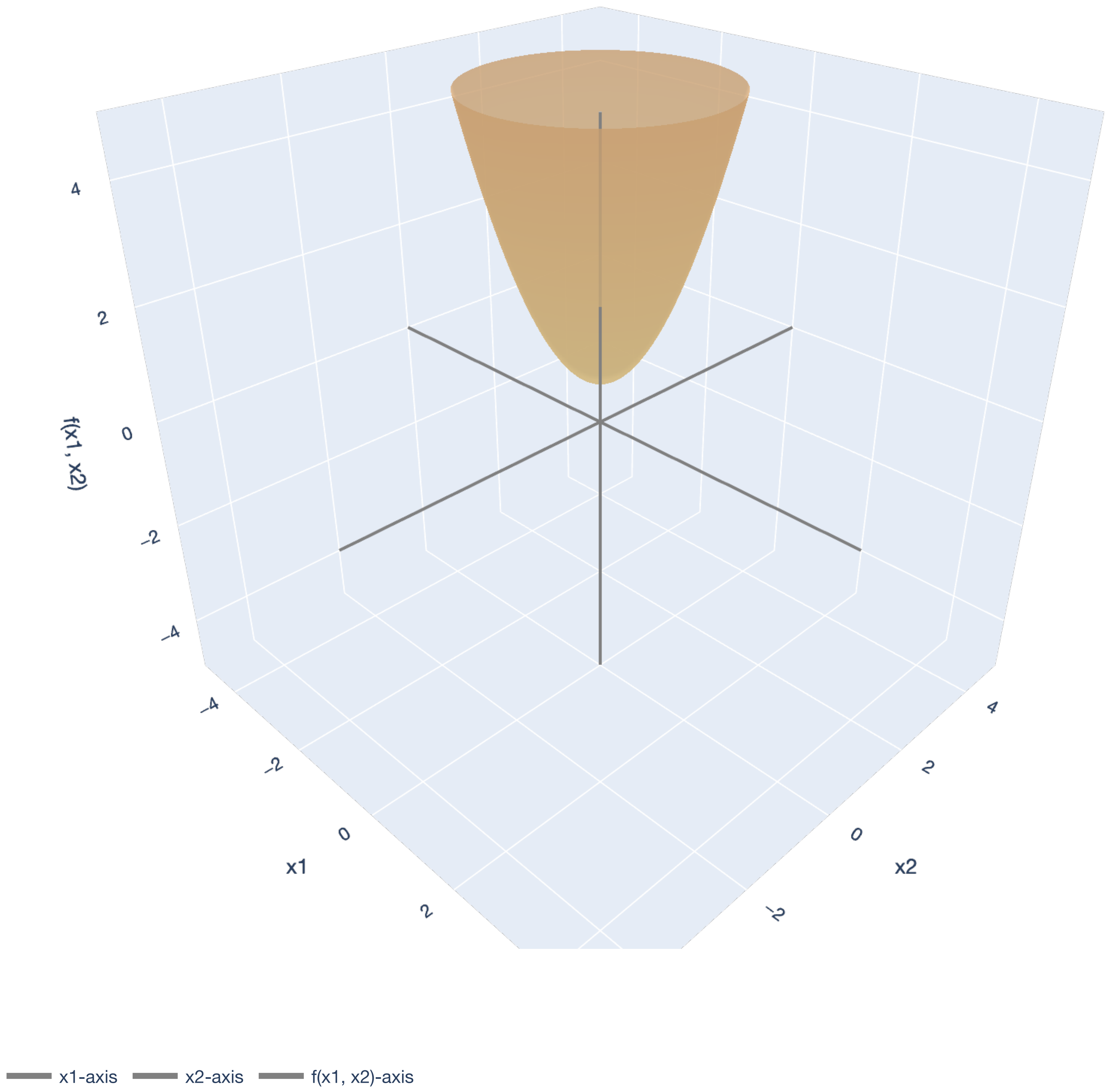
Lesson Overview

Lesson Overview

Big Picture: Least Squares



$$\lambda_1, \ldots, \lambda_d \geq 0$$



$$\lambda_1, \ldots, \lambda_d > 0$$

Lesson Overview

Big Picture: Gradient Descent