Lecture 14

Gradient Descent

DSC 40A, Fall 2024

The Midterm Exam is on Monday, Nov 4th!

- Randomized seat assignment is in the homework - look up your seat.
- 50 minutes, on paper, no calculators or electronics.
  - **You are allowed to bring one two-sided page of notes.**
- Content: Lectures 1-13, Homeworks 1-4, Groupworks 1-4.
- Prepare by practicing with old exam problems at practice.dsc40a.com.
  - Problems are sorted by topic!

Agenda

- Minimizing functions using gradient descent.
- Convexity.
- More examples.
    - Huber loss.
    - Gradient descent with multiple variables.

Question 🤔

Answer at q.dsc40a.com

Remember, you can always ask questions at q.dsc40a.com!

If the direct link doesn't work, click the "🤔 Lecture Questions" link in the top right corner of dsc40a.com.

The modeling recipe

1. Choose a model.

2. Choose a loss function.

3. Minimize average loss to find optimal model parameters.

Minimizing functions using gradient descent

Minimizing empirical risk

- Repeatedly, we've been tasked with **minimizing** the value of empirical risk functions.
  - Why? To help us find the **best** model parameters, h^* or w^*, which help us make the **best** predictions!
- We've minimized empirical risk functions in various ways.
  - $R_{\text{sq}}(h) = \dfrac{1}{n} \displaystyle\sum_{i=1}^{n} (y_i - h)^2$
  - $R_{\text{abs}}(w_0, w_1) = \dfrac{1}{n} \displaystyle\sum_{i=1}^{n} |y_i - (w_0 + w_1 x)|$
  - $R_{\text{sq}}(\vec{w}) = \dfrac{1}{n} \|\vec{y} - X\vec{w}\|^2$

Minimizing arbitrary functions

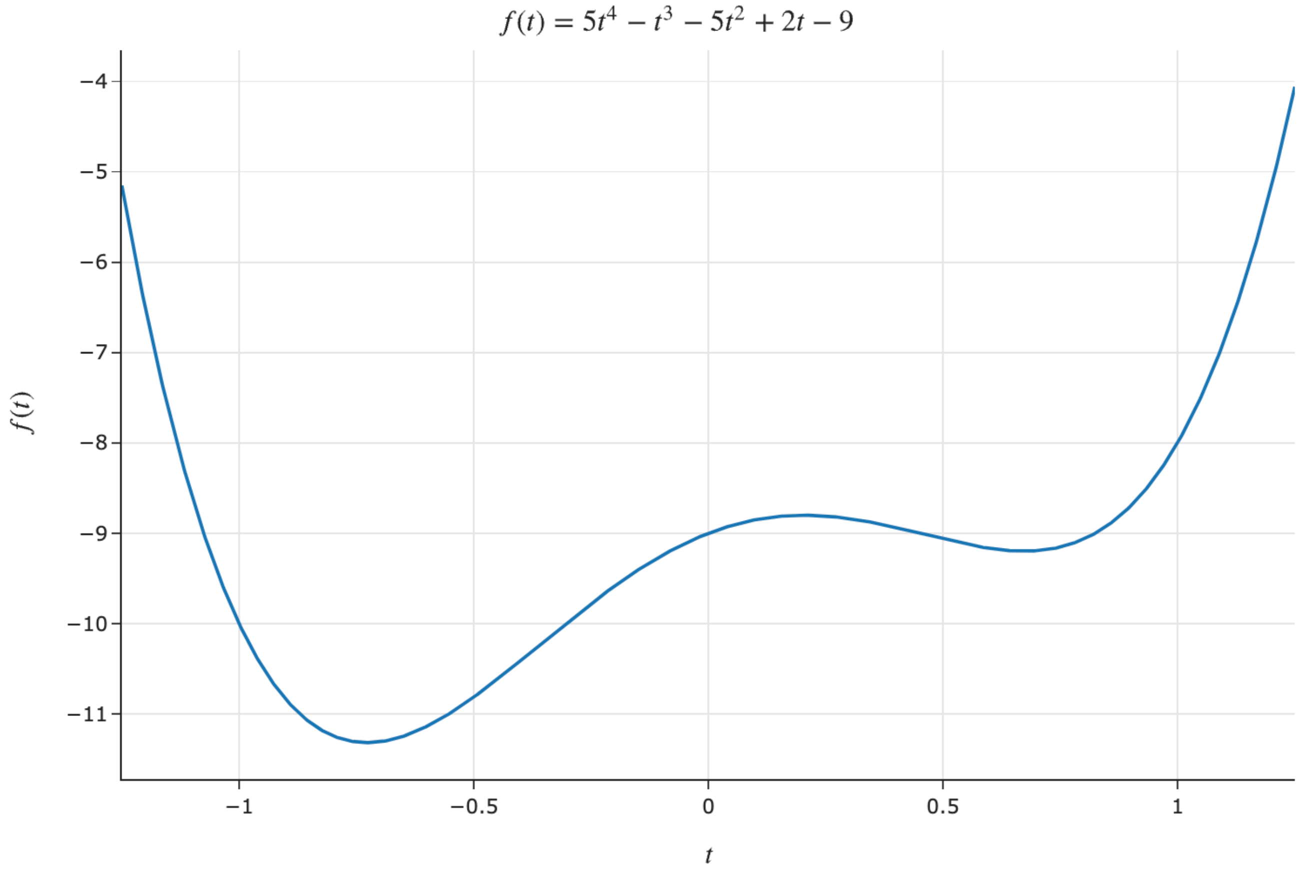
- Assume $f(t)$ is some **differentiable** single-variable function.
- When tasked with minimizing $f(t)$, our general strategy has been to:
  i. Find $\frac{df}{dt}(t)$, the derivative of f.
  ii. Find the input t^* such that $\frac{df}{dt}(t^*) = 0$.
- However, there are cases where we can find $\frac{df}{dt}(t)$, but **it is either difficult or impossible to solve** $\frac{df}{dt}(t^*) = 0$.

$$f(t) = 5t^4 - t^3 - 5t^2 + 2t - 9$$

- Then what?

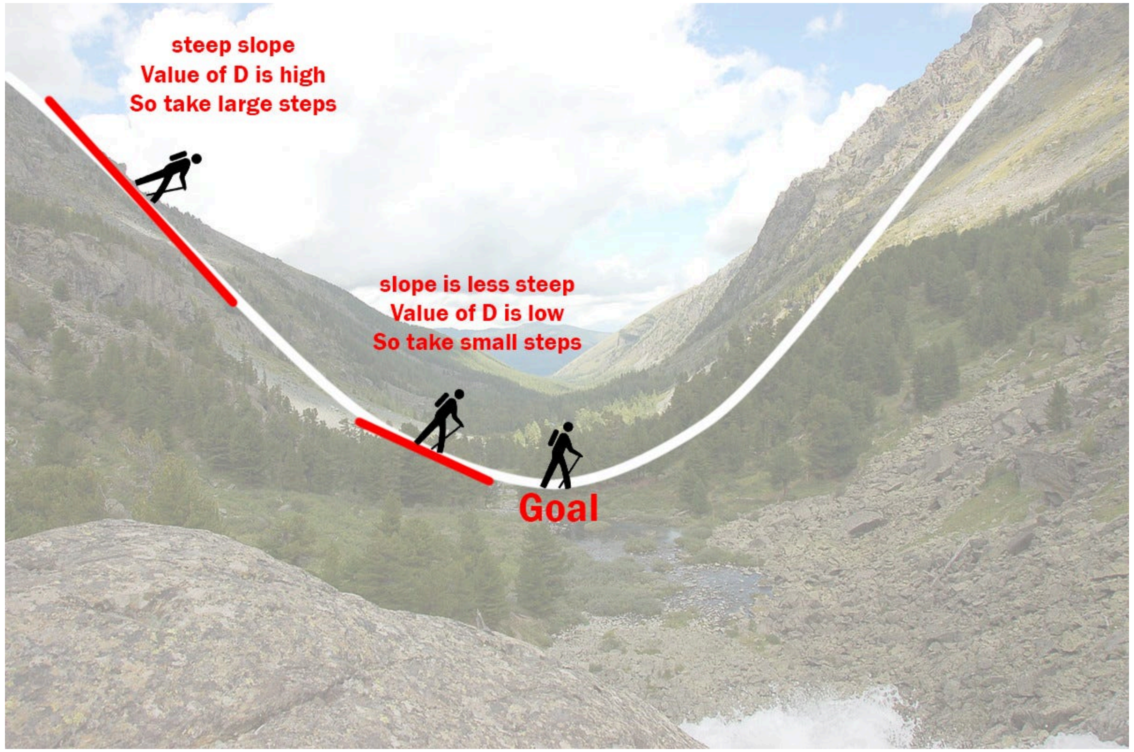
What does the derivative of a function tell us?

- **Goal**: Given a **differentiable** function $f(t)$, find the input t^* that minimizes $f(t)$.
- What does $\frac{d}{dt} f(t)$ mean?

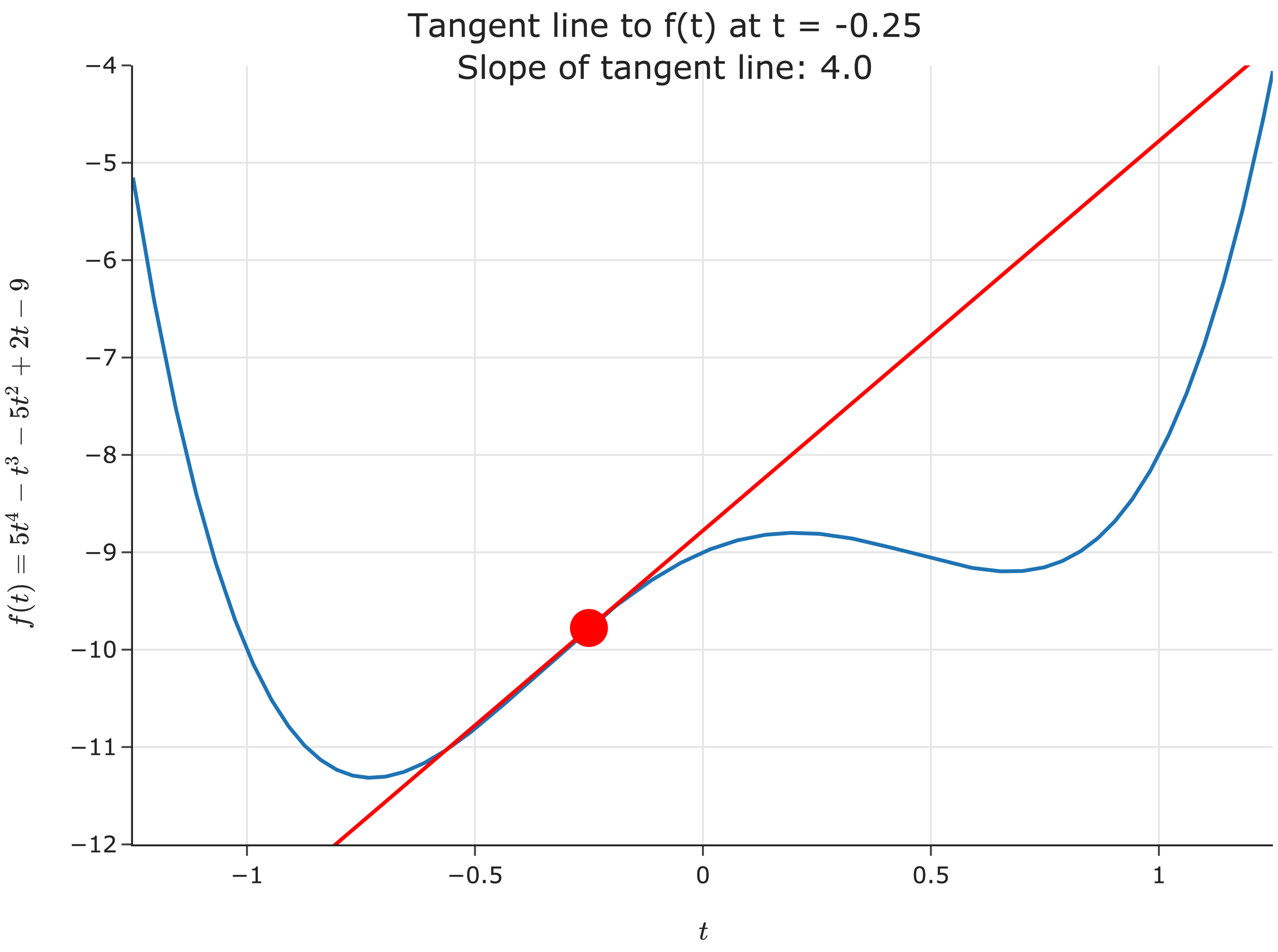


Let's go hiking!

- Suppose you're at the top of a mountain 🏔️ and need to get **to the bottom**.
- Further, suppose it's really cloudy ☁️, meaning you can only see a few feet around you.
- **How** would you get to the bottom?



Searching for the minimum

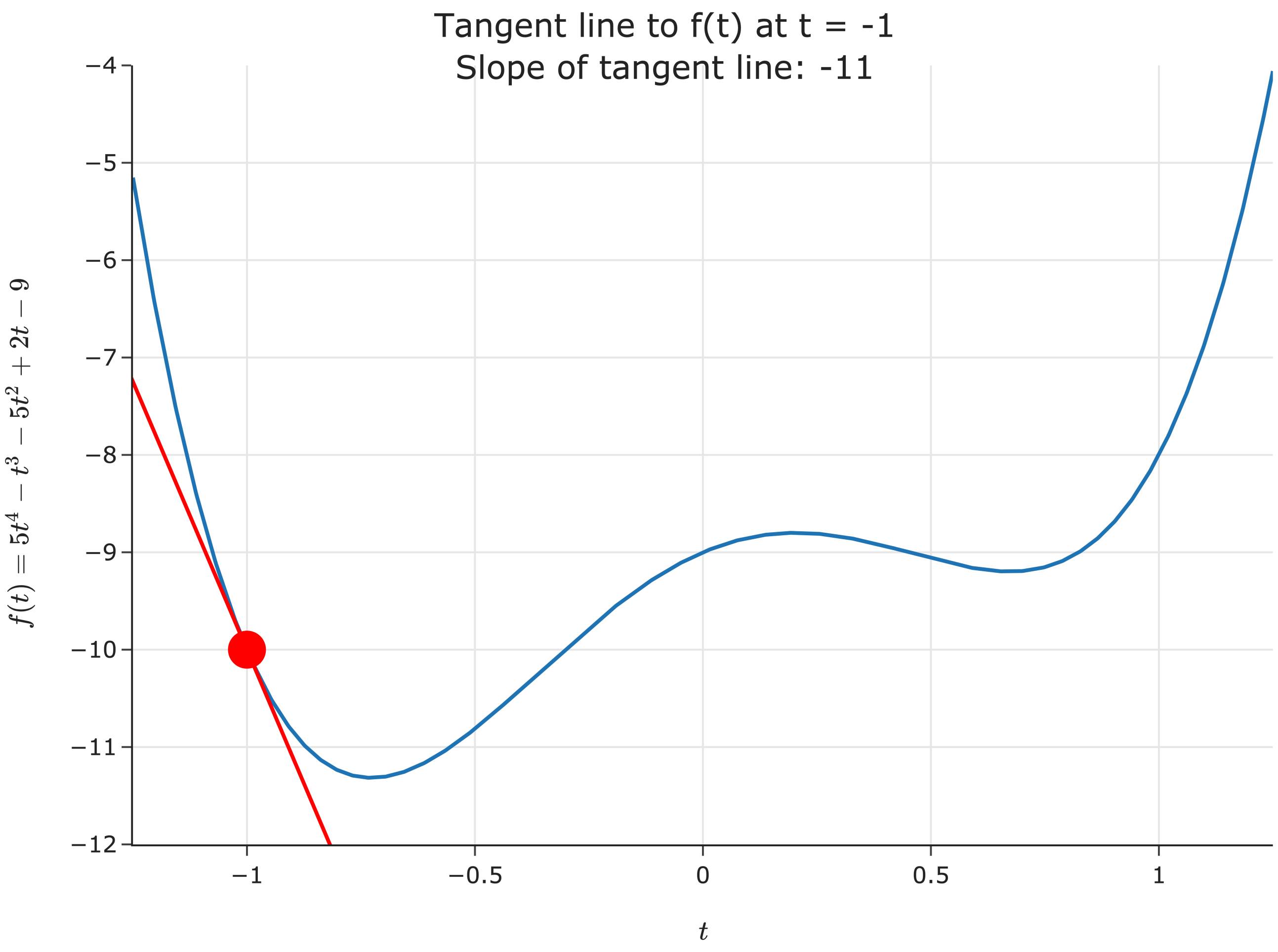


Suppose we're given an initial *guess* for a value of t that minimizes $f(t)$.

If the **slope of the tangent line at $f(t)$** is **positive** 📈:

- Increasing t **increases** f.
- This means the minimum must be to the **left** of the point $(t, f(t))$.
- Solution: **Decrease** t ⬇️.

Searching for the minimum



Suppose we're given an initial *guess* for a value of t that minimizes $f(t)$.

If the **slope of the tangent line at $f(t)$** is **negative** 📉:

- Increasing t **decreases** f.
- This means the minimum must be to the **right** of the point $(t, f(t))$.
- Solution: **Increase** t ⬆️.

Intuition

- To minimize $f(t)$, start with an initial guess t_0.
- Where do we go next?
  - If $\frac{df}{dt}(t_0) > 0$, **decrease** t_0.
  - If $\frac{df}{dt}(t_0) < 0$, **increase** t_0.
- One way to accomplish this:

$$t_1 = t_0 - \frac{df}{dt}(t_0)$$

Gradient descent

To minimize a **differentiable** function f:

- Pick a positive number, α. This number is called the **learning rate**, or **step size**.
- Pick an **initial guess**, t_0.
- Then, repeatedly update your guess using the **update rule**:

$$t_{i+1} = t_i - \alpha \frac{df}{dt}(t_i)$$

- Repeat this process until **convergence** – that is, when t doesn't change much.
- This procedure is called **gradient descent**.

What is gradient descent?

- Gradient descent is a numerical method for finding the input to a function f that minimizes the function.
- Why is it called **gradient** descent?
    - The gradient is the extension of the derivative to functions of multiple variables.
    - We will see how to use gradient descent with multivariate functions next class.
- What is a **numerical** method?
    - A numerical method is a technique for approximating the solution to a mathematical problem, often by using the computer.
- Gradient descent is **widely used** in machine learning, to train models from linear regression to neural networks and transformers (includng ChatGPT)!

See this notebook for a demo!

Gradient descent and empirical risk minimization

- While gradient descent can minimize other kinds of differentiable functions, its most common use case is in **minimizing empirical risk**.
- For example, consider:
    - The constant model, $H(x) = h$.
    - The dataset $-4, -2, 2, 4$.
    - The initial guess $h_0 = 4$ and the learning rate $\alpha = \frac{1}{4}$.
- **Exercise**: Find h_1 and h_2.

Lingering questions

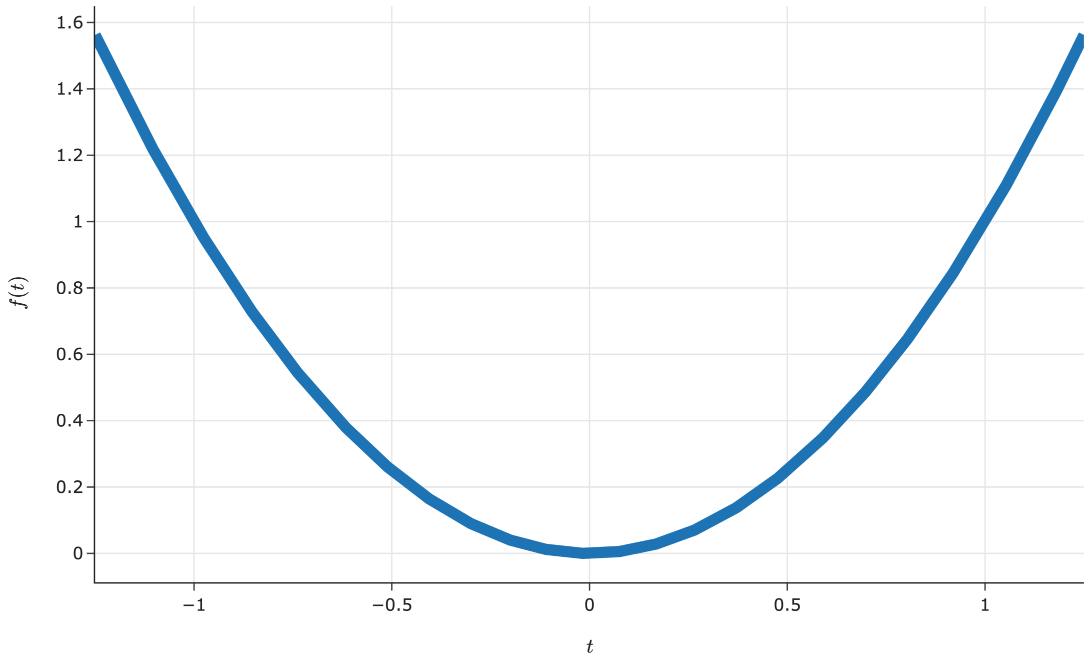
Now, we'll explore the following ideas:

- When is gradient descent *guaranteed* to converge to a global minimum?
    - What kinds of functions work well with gradient descent?
- How do I choose a step size?
- How do I use gradient descent to minimize functions of multiple variables, e.g.:

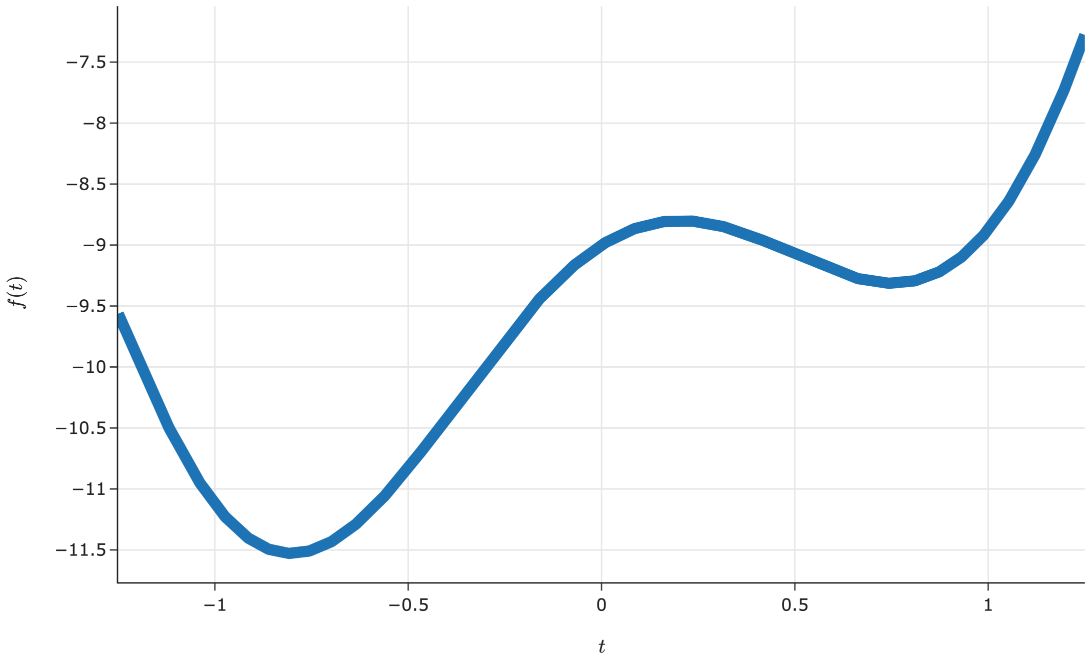
$$R_{\text{sq}}(w_0, w_1) = \frac{1}{n} \sum_{i=1}^{n} (y_i - (w_0 + w_1 x_i))^2$$

When is gradient descent guaranteed to work?

Convex functions



A **convex** function ✅



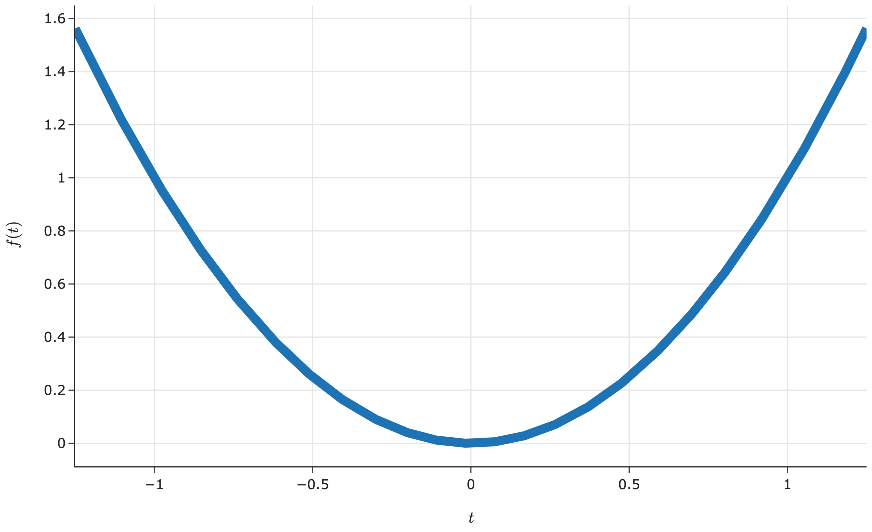
A **non-convex** function ❌

Convexity

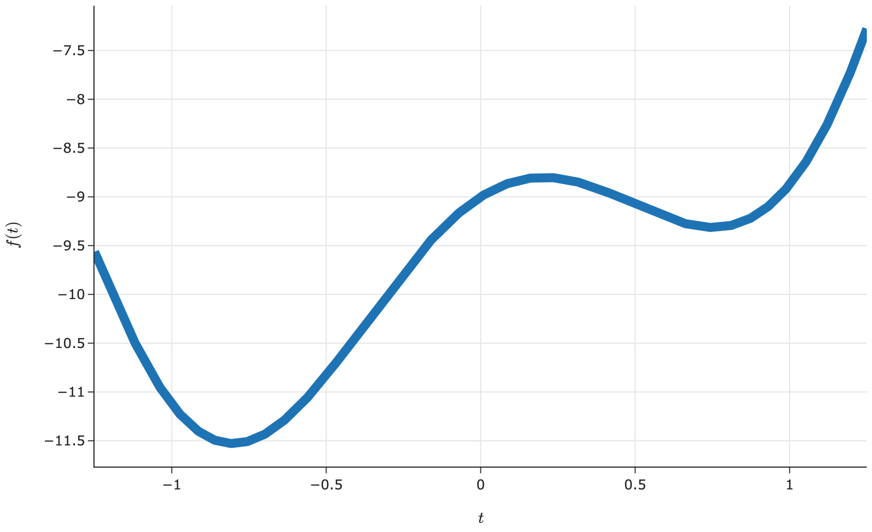
- A function f is **convex** if, for **every** a, b in the domain of f, the line segment between:

$$(a, f(a)) \text{ and } (b, f(b))$$

does not go below the plot of f.



A **convex** function ✅

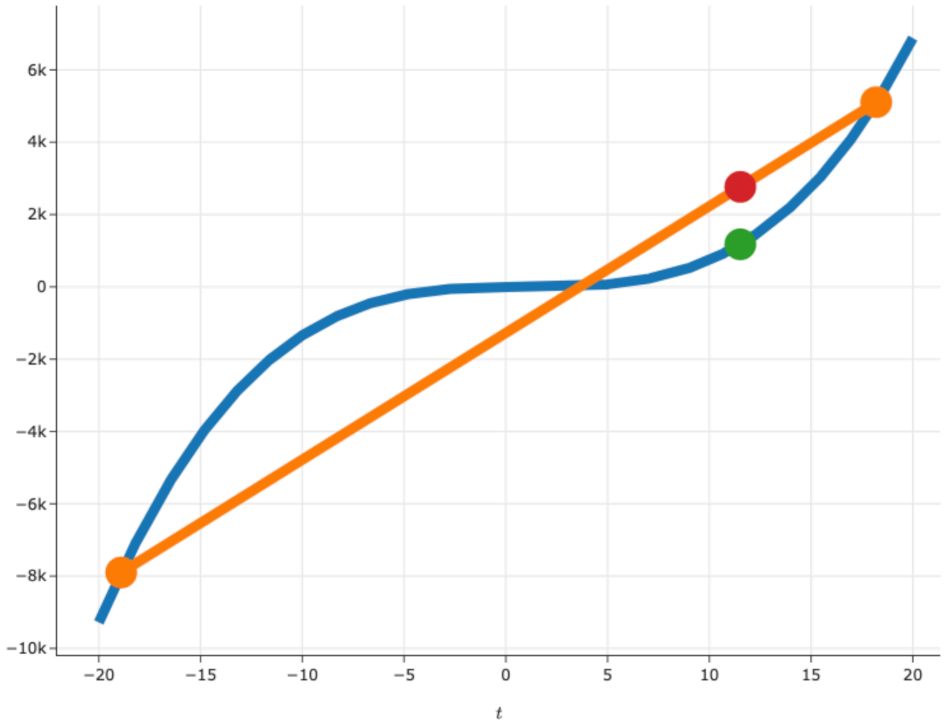


A **non-convex** function ❌

Formal definition of convexity

- A function $f : \mathbb{R} \to \mathbb{R}$ is **convex** if, for **every** a, b in the domain of f, and for every $t \in [0, 1]$:

$$\boxed{(1-t)f(a) + tf(b) \geq f((1-t)a + tb)}$$



- This is a formal way of restating the definition from the previous slide.

Question 🤔

Answer at q.dsc40a.com

Which of these functions are **not** convex?

- A. $f(x) = |x|$.
- B. $f(x) = e^x$.
- C. $f(x) = \sqrt{x - 1}$.
- D. $f(x) = (x - 3)^{24}$.
- E. More than one of the above are non-convex.

Second derivative test for convexity

- If $f(t)$ is a function of a single variable and is **twice** differentiable, then $f(t)$ is convex **if and only if**:

$$\frac{d^2 f}{dt^2}(t) \geq 0, \quad \forall\, t$$

- Example: $f(x) = x^4$ is convex.

Why does convexity matter?

- Convex functions are (relatively) easy to minimize with gradient descent.
- **Theorem**: If $f(t)$ is convex and differentiable, then gradient descent converges to a **global minimum** of f, as long as the step size is small enough.
- **Why?**
  - Gradient descent converges when the derivative is 0.
  - For convex functions, the derivative is 0 only at one place – the global minimum.
  - In other words, if f is convex, gradient descent won't get "stuck" and terminate in places that aren't global minimums (local minimums, saddle points, etc.).

Nonconvex functions and gradient descent

- We say a function is **nonconvex** if it does not meet the criteria for convexity.
- Nonconvex functions are (relatively) difficult to minimize.
- Gradient descent **might** still work, but it's not guaranteed to find a global minimum.
    - We saw this at the start of the lecture, when trying to minimize $f(t) = 5t^4 - t^3 - 5t^2 + 2t - 9$.

Choosing a step size in practice

- In practice, choosing a step size involves a lot of trial-and-error.
- In this class, we've only touched on "constant" step sizes, i.e. where α is a constant.

$$t_{i+1} = t_i - \alpha \frac{df}{dt}(t_i)$$

- **Remember**: α is the "step size", but the amount that our guess for t changes is $\alpha \frac{df}{dt}(t_i)$, not just α.
- In future courses, you'll learn about "decaying" step sizes, where the value of α decreases as the number of iterations increases.
    - Intuition: take much bigger steps at the start, and smaller steps as you progress, as you're likely getting closer to the minimum.