To minimize a **differentiable** function f:

1. Pick a positive number, α. This number is called the **learning rate**, or **step size**.
   Think of α as a hyperparameter of the minimization process.

2. Pick an **initial guess**, $w^{(0)}$.

guess for w^*

3. Then, repeatedly update your guess using the **update rule**:

$$w^{(t+1)} = w^{(t)} - \alpha \frac{df}{dw}(w^{(t)})$$

negative because we want to move opposite the derivative

learning rate / step size

$\Rightarrow$ multiplier on derivative.

- Repeat this process until **convergence** – that is, when w doesn't change much from iteration to iteration.

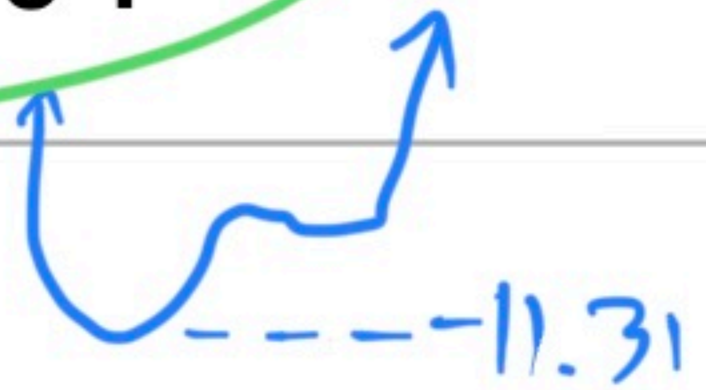
- This procedure is called **gradient descent**.

```
w = -0.726, f(w) = -11.3157, df/dw(w) = 0.0246
w = -0.7263, f(w) = -11.3157, df/dw(w) = 0.0182
w = -0.7265, f(w) = -11.3157, df/dw(w) = 0.0134
w = -0.7266, f(w) = -11.3157, df/dw(w) = 0.0099
w = -0.7267, f(w) = -11.3157, df/dw(w) = 0.0074
w = -0.7268, f(w) = -11.3157, df/dw(w) = 0.0054
w = -0.7268, f(w) = -11.3157, df/dw(w) = 0.004
w = -0.7269, f(w) = -11.3157, df/dw(w) = 0.003
w = -0.7269, f(w) = -11.3157, df/dw(w) = 0.0022
w = -0.7269, f(w) = -11.3157, df/dw(w) = 0.0016
w = -0.7269, f(w) = -11.3157, df/dw(w) = 0.0012
w = -0.727, f(w) = -11.3157, df/dw(w) = 0.0009
w = -0.727, f(w) = -11.3157, df/dw(w) = 0.0007
w = -0.727, f(w) = -11.3157, df/dw(w) = 0.0005
w = -0.727, f(w) = -11.3157, df/dw(w) = 0.0004
```

derivatives getting closer to 0!

converging at $w^* = -0.727$

height of the bottom



$y_1 = -4$

$y_2 = -2$

$y_3 = 2$

$y_4 = 4$

$h^{(0)} = 4$

$\alpha = \frac{1}{4}$

squared loss

$$R_{sq}(h) = \frac{1}{n} \sum_{i=1}^{n} (y_i - h)^2 \qquad n = 4$$

mean squared error

$$\frac{dR}{dh} = \frac{1}{n} \sum_{i=1}^{n} 2(y_i - h)(-1)$$

$$= -\frac{1}{2} \sum_{i=1}^{4} (y_i - h)$$

$$= -\frac{1}{2} \left(\sum_{i=1}^{4} y_i - 4h \right)$$

$$= -\frac{1}{2} (0 - 4h) = 2h$$

need this for grad. desc.!

$$-4 - 2 + 2 + 4 = 0$$

$$h^{(0)} = 4$$

$$h^{(1)} = h^{(0)} - \alpha \frac{dR}{dh}(h^{(0)})$$

$$= 4 - \frac{1}{\cancel{4}} \cdot \underbrace{2(\cancel{4})}_{\frac{dR}{dh}(h) = 2h,\ \text{so this is } 2 \times 4.}$$

$$= 4 - 2 = \boxed{2}$$

$$h^{(2)} = h^{(1)} - \alpha \frac{dR}{dh}(h^{(1)}) = 2 - \frac{1}{\cancel{4}} \cdot \cancel{2(2)} = \boxed{1}$$

- When is gradient descent *guaranteed* to converge to a global minimum? What kinds of functions work well with gradient descent?

- How do we choose a step size?

- How do we use gradient descent to minimize functions of multiple variables, e.g.:

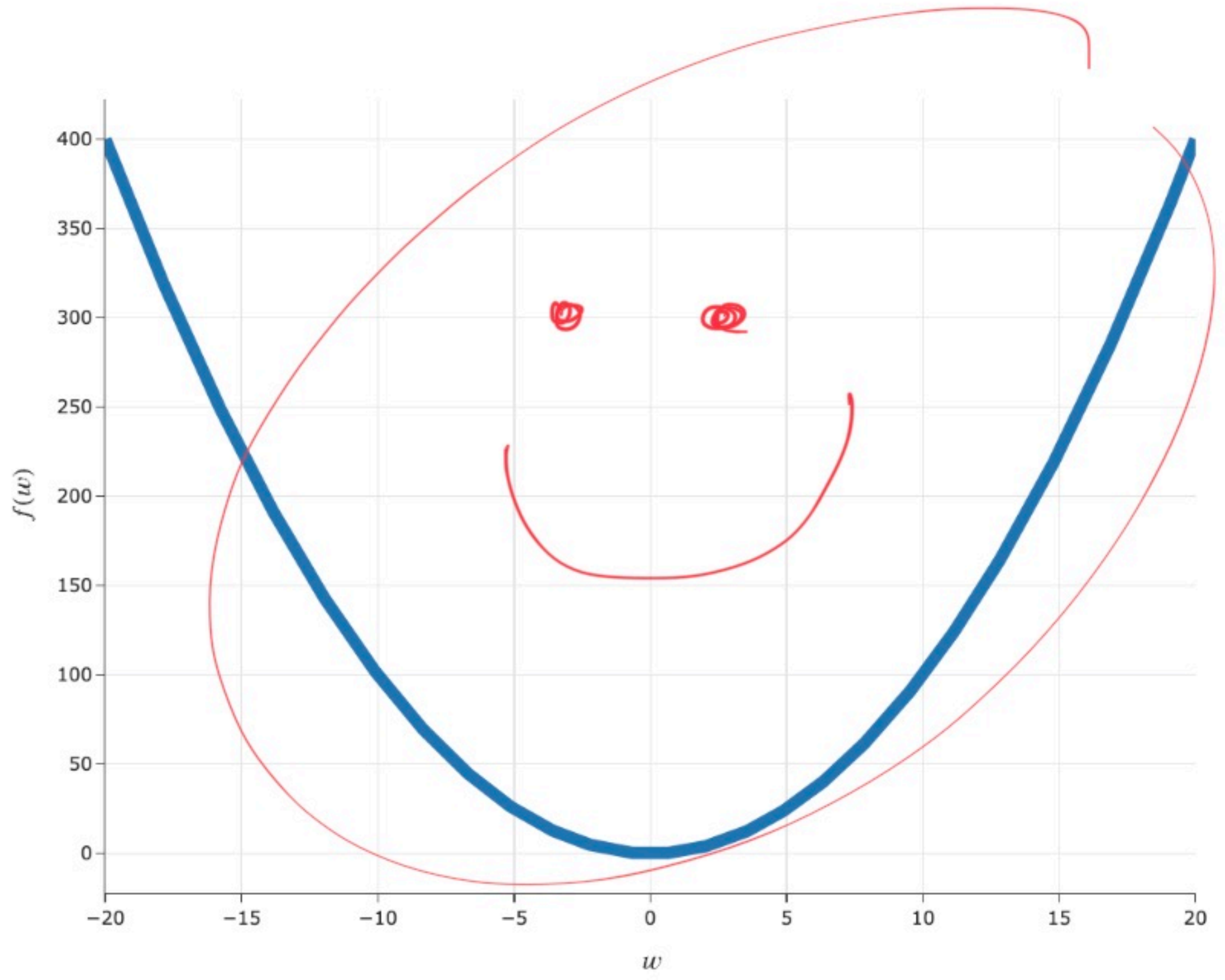
$$R_{\text{ridge}}(\vec{w}) = \frac{1}{n} \|\vec{y} - X\vec{w}\|^2 + \lambda \sum_{j=1}^{d} w_j^2$$

- **Question**: Why **can't** we use gradient descent to find $\vec{w}^*_{\text{LASSO}}$?

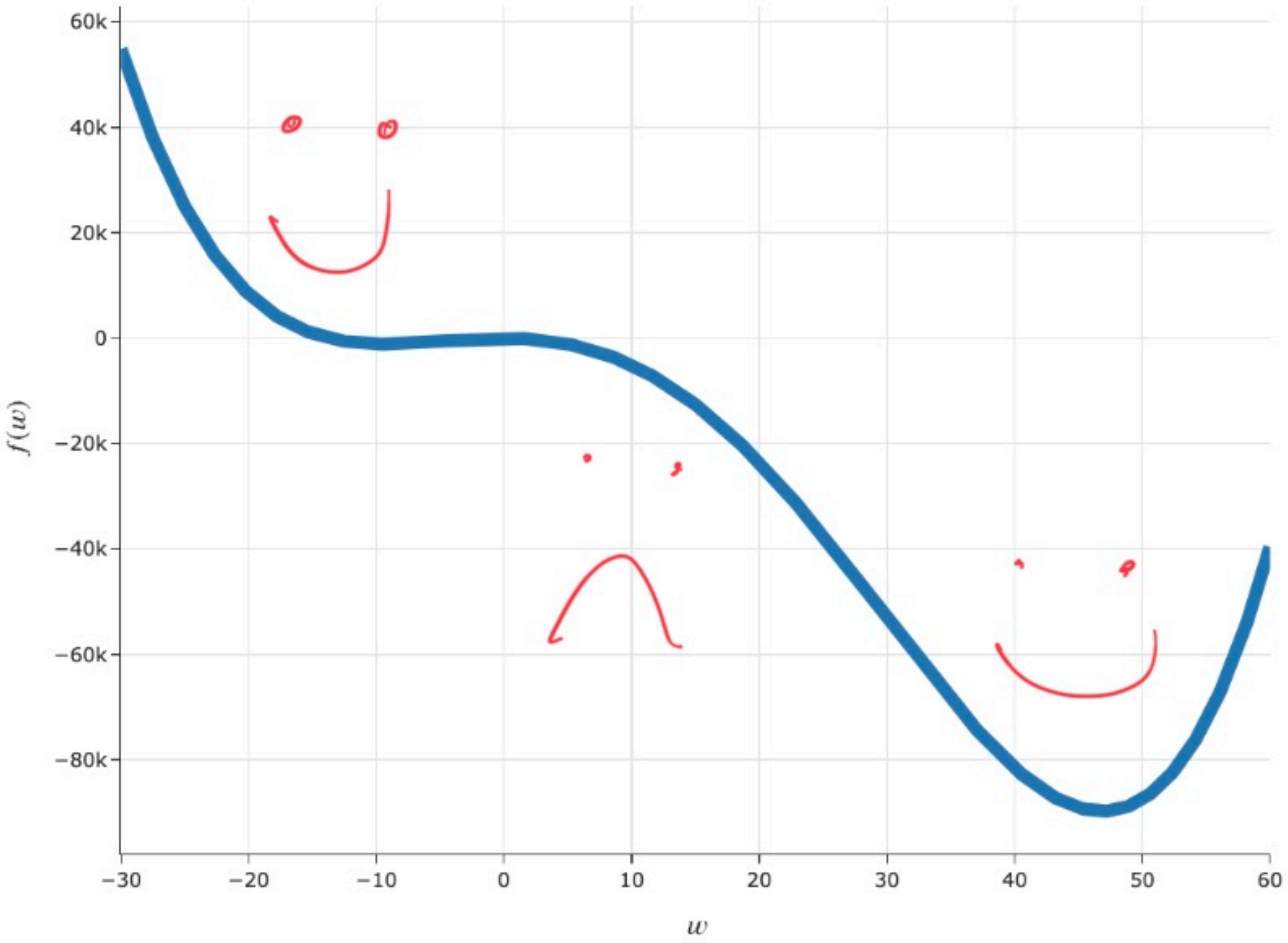
$$R_{\text{LASSO}}(\vec{w}) = \frac{1}{n} \|\vec{y} - X\vec{w}\|^2 + \lambda \sum_{j=1}^{d} |w_d|$$

absolute value NOT differentiable

What makes a function convex?



A **convex** function ✅.



A **non-convex** function ❌.

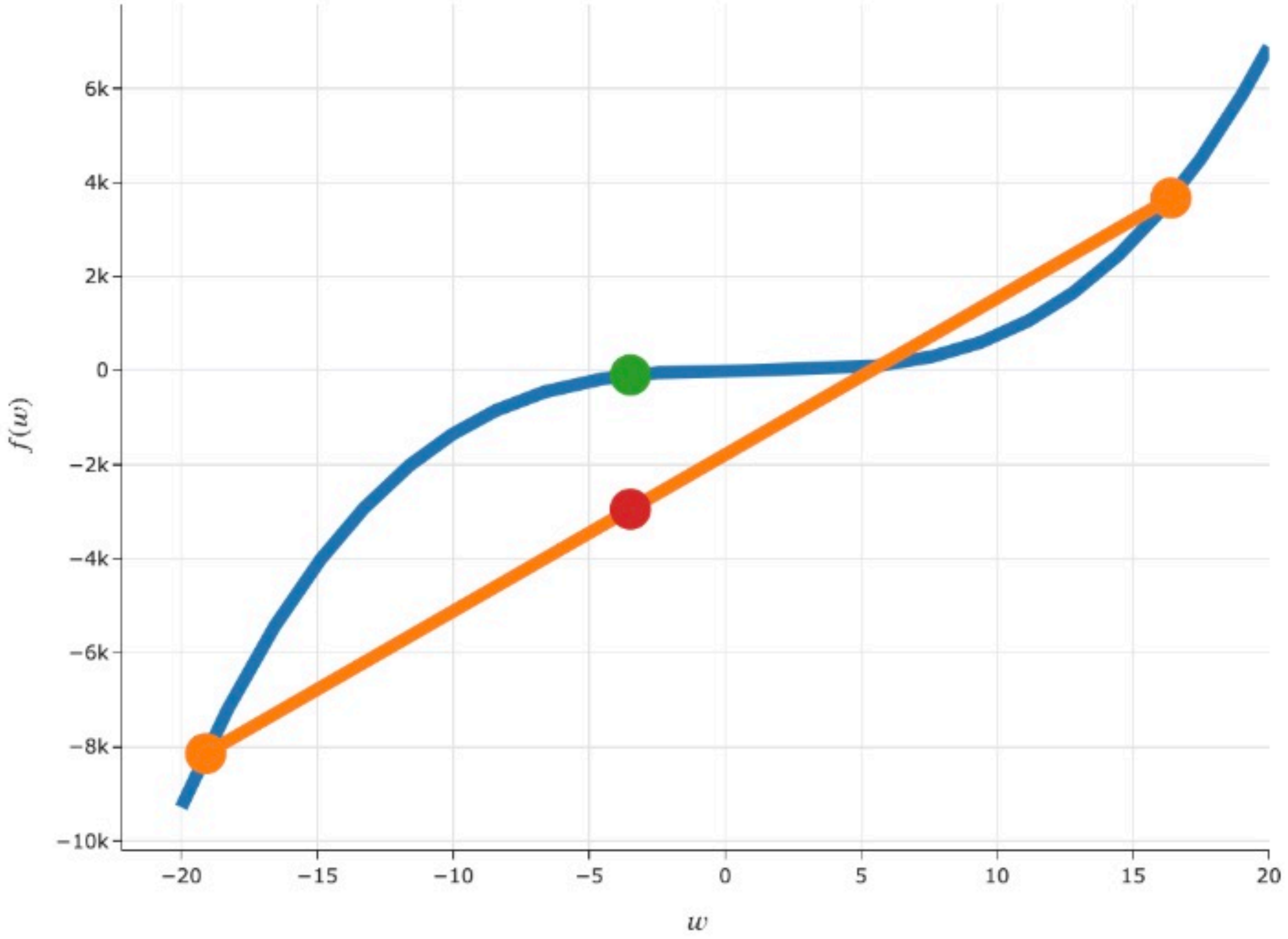
Formal definition of convexity

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

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

the line between f(a) and f(b)

a point on f between a and b



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

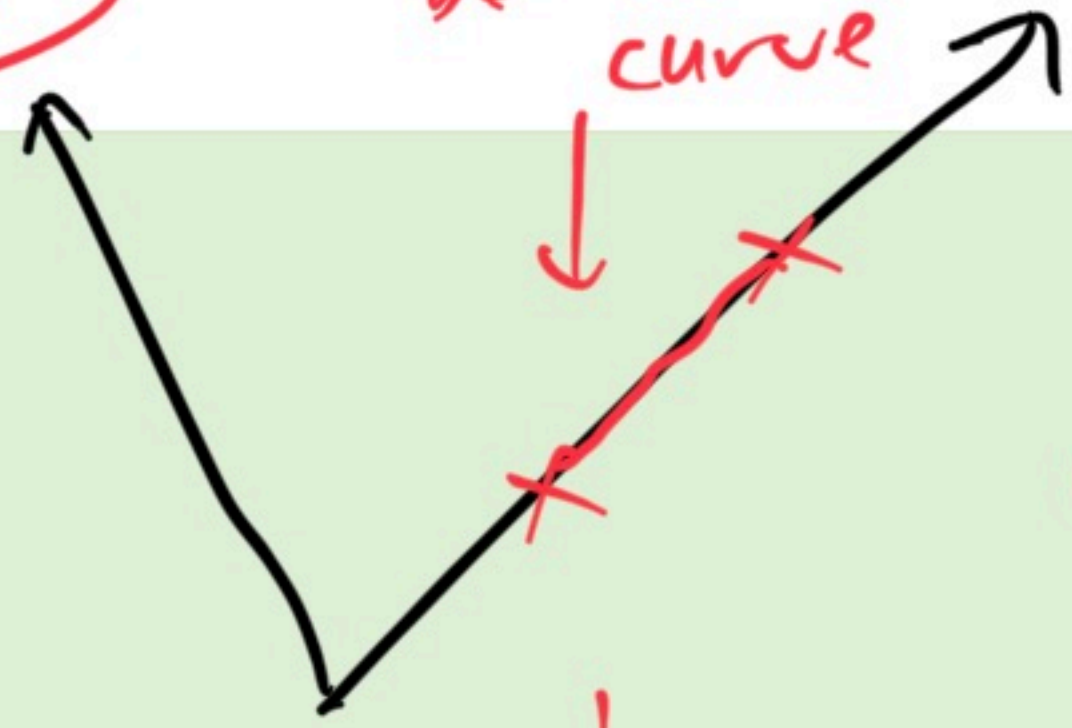
Activity

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.

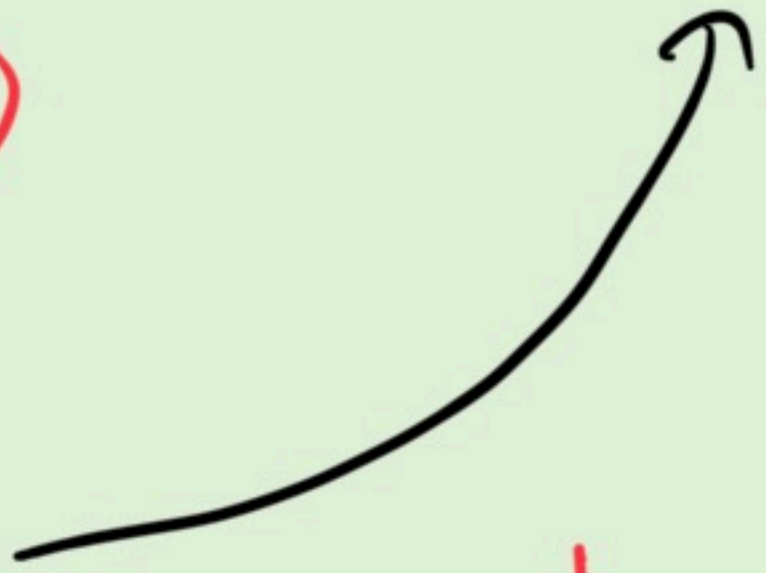
A

line can be on the curve



convex!

B



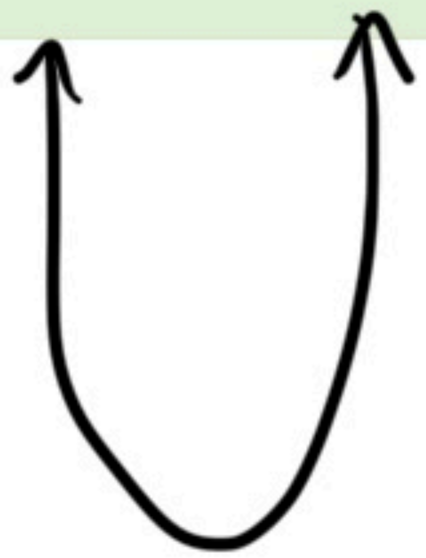
convex!

C

line below = BAD!

non-convex

D



even degree : convex

odd degree : non-convex

can take derivative twice!

Second derivative test for convexity

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

$$\frac{d^2 f}{dw^2}(w) \geq 0, \quad \underbrace{\forall}_{\text{for all}}\, w$$

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

$$\frac{df}{dx} = 4x^3 \qquad \frac{d^2f}{dx^2} = 12x^2 \geq 0 \quad \forall x$$

localhost

$$\vec{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

$$\nabla f(\vec{x}) = \begin{bmatrix} 2(x_1 - 1) \\ -2(x_2 - 3) \end{bmatrix}$$

gradient of f: vector of partial derivatives.

Minimizing functions of multiple variables

- Consider the function:

$$f(x_1, x_2) = (x_1 - 2)^2 + 2x_1 - (x_2 - 3)^2$$

- It has two **partial derivatives**: $\frac{\partial f}{\partial x_1}$ and $\frac{\partial f}{\partial x_2}$.

  See the annotated slides for what they are and how we find them.

$$\underbrace{\frac{\partial f}{\partial x_1}}_{\text{treat } x_2 \text{ as constant!}} = 2(x_1 - 2) + 2 = 2(x_1 - 1)$$

$$\frac{\partial f}{\partial x_2} = -2(x_2 - 3)$$

?

The gradient vector

- If $f(\vec{x})$ is a function of multiple variables, then its **gradient**, $\nabla f(\vec{x})$, is a vector containing its partial derivatives.

$$\vec{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}_n$$

- Example:

$$f(\vec{x}) = (x_1 - 2)^2 + 2x_1 - (x_2 - 3)^2$$

$$\nabla f(\vec{x}) = \begin{bmatrix} 2(x_1 - 1) \\ -2(x_2 - 3) \end{bmatrix}$$

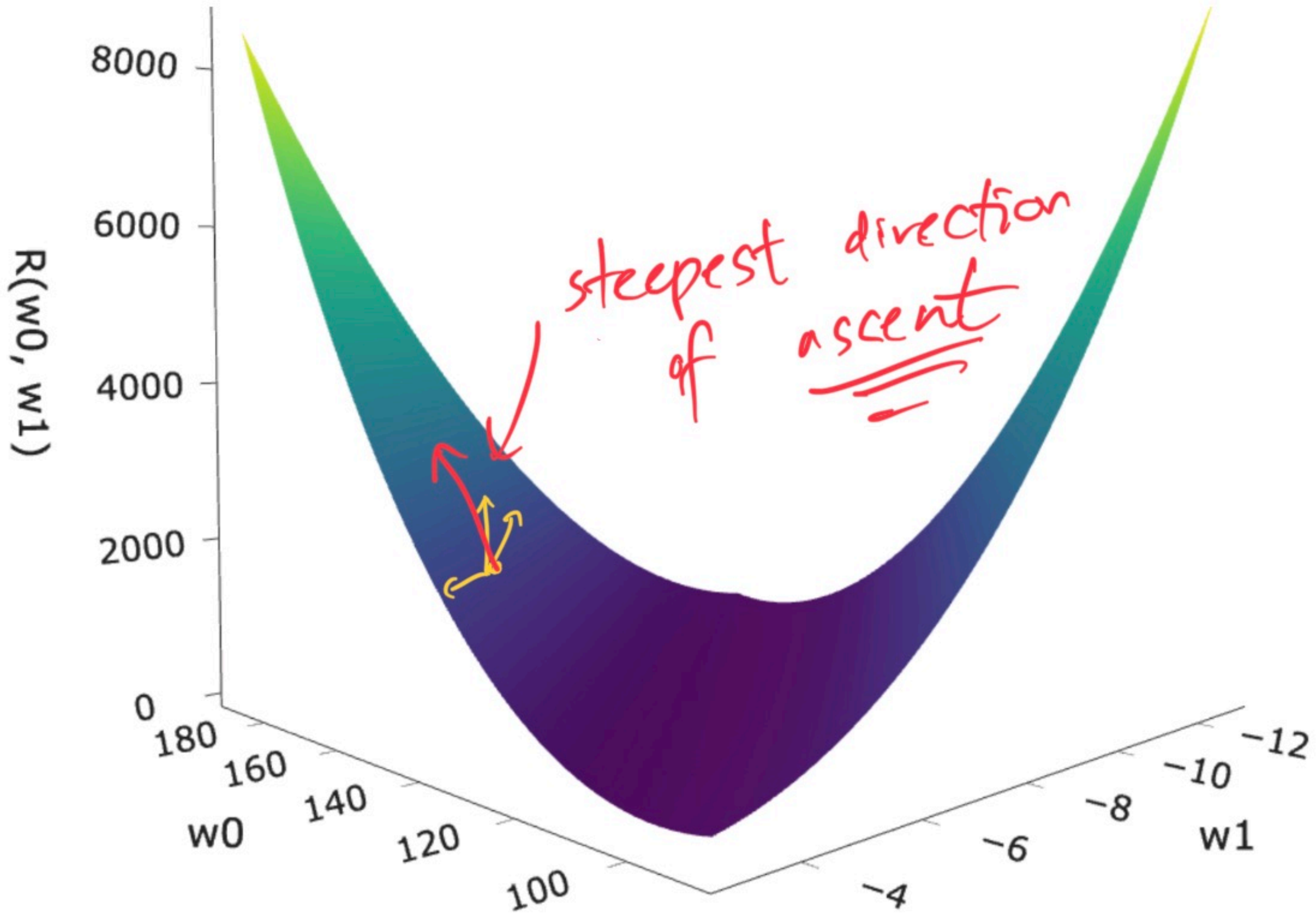
$$\vec{x}^T \vec{x} = \begin{bmatrix} x_1 & x_2 & \cdots & x_n \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = x_1^2 + x_2^2 + \cdots + x_n^2$$

- Example:

$$\forall i, \quad \frac{\partial f}{\partial x_i} = 2x_i$$

$$f(\vec{x}) = \vec{x}^T \vec{x}$$

$$\nabla f(\vec{x}) = 2\vec{x}$$



At any given point, there are many directions in which you can go "up", but there's only one "steepest direction up", and that's the direction of the gradient!

Gradient descent for functions of multiple variables

$\nabla f(\vec{x}^{(t)})$ is the direction of steepest ascent,

- Example:

$$f(x_1, x_2) = (x_1 - 2)^2 + 2x_1 - (x_2 - 3)^2$$

$$\nabla f(\vec{x}) = \begin{bmatrix} 2(x_1 - 1) \\ -2(x_2 - 3) \end{bmatrix}$$

$-\nabla f(\vec{x}^{(t)})$ is the direction of steepest descent

- The minimizer of f is a vector, $\vec{x}^* = \begin{bmatrix} x_1^* \\ x_2^* \end{bmatrix}$.

- We start with an initial guess, $\vec{x}^{(0)}$, and step size α, and update our guesses using:

$$\vec{x}^{(t+1)} = \vec{x}^{(t)} - \alpha \nabla f(\vec{x}^{(t)})$$

Activity

goal: minimize to find (x_1^*, x_2^*)

$$f(x_1, x_2) = (x_1 - 2)^2 + 2x_1 - (x_2 - 3)^2$$

some bowl-ish thing in 3D

$$\nabla f(\vec{x}) = \begin{bmatrix} 2(x_1 - 1) \\ -2(x_2 - 3) \end{bmatrix}$$

$$\vec{x}^{(1)} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} - \frac{1}{3}\begin{bmatrix} 2(0-1) \\ -2(0-3) \end{bmatrix}$$

$$= \begin{bmatrix} 0 \\ 0 \end{bmatrix} - \frac{1}{3}\begin{bmatrix} -2 \\ 6 \end{bmatrix}$$

$$\vec{x}^{(t+1)} = \vec{x}^{(t)} - \alpha \nabla f(\vec{x}^{(t)})$$

$$\vec{x}^{(1)} = \begin{bmatrix} 2/3 \\ -2 \end{bmatrix}$$

Given an initial guess of $\vec{x}^{(0)} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$ and a step size of $\alpha = \frac{1}{3}$, perform **two** iterations of gradient descent. What is $\vec{x}^{(2)}$?

updated!

$$\vec{x}^{(2)} = \begin{bmatrix} \frac{2}{3} \\ 2 \end{bmatrix} - \frac{1}{3}\begin{bmatrix} 2\left(2/3 - 1\right) \\ -2(-2 - 3) \end{bmatrix} = \ldots$$

empirical risk:

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

- This is a function of multiple variables, and is differentiable, so it has a gradient!

$$\nabla R(\vec{w}) = \begin{bmatrix} -\frac{2}{n} \sum_{i=1}^{n} (y_i - (w_0 + w_1 x_i)) \\ -\frac{2}{n} \sum_{i=1}^{n} (y_i - (w_0 + w_1 x_i)) x_i \end{bmatrix}$$

- **Key idea**: To find $\vec{w}^* = \begin{bmatrix} w_0^* \\ w_1^* \end{bmatrix}$, we *could* use gradient descent!

- Why would we, when closed-form solutions exist?

often more efficient!

inverting $X^T X$ is costly