EECS 245, Spring 2026

LEC 10 Gradients and Gradient Descent

→ Read: Ch. 8

Agenda

Ch 8.1-8.4 all in scope

- Recap: Big 3 gradient rules
- Minimizing $R_{sq}(\vec{w}) = \frac{1}{n}\|\vec{y} - X\vec{w}\|^2$ using gradients
- Using gradients to minimize functions algorithmically – gradient descent
- If time: convexity – not in scope for MT2, in scope for Final

Announcements

- Deadlines:
  - HW 7 today
  - HW 8 Sunday, no slip days
  - Lab 9 Monday
- Midterm 2 on Tuesday from 1-3PM! 2 new conceptual Q's
- No in-person lecture next Thursday – videos will be posted

Practice Conceptual Questions

It's often said that the best way to learn something is by teaching it. In IA interviews for this class, we usually ask conceptual questions like the ones below. It's a good idea to study for Midterm 2 by trying to explain the questions below to your peers and having them ask questions of your explanation.

1. Explain where the normal equations came from.
2. How does finding the line of best fit have anything to do with the normal equations?
3. Prove that the null space is a subspace.
4. Prove that every element in the column space is orthogonal to every element in the null space of A^T.
5. What does the rank-nullity theorem say?

Recap: Gradients

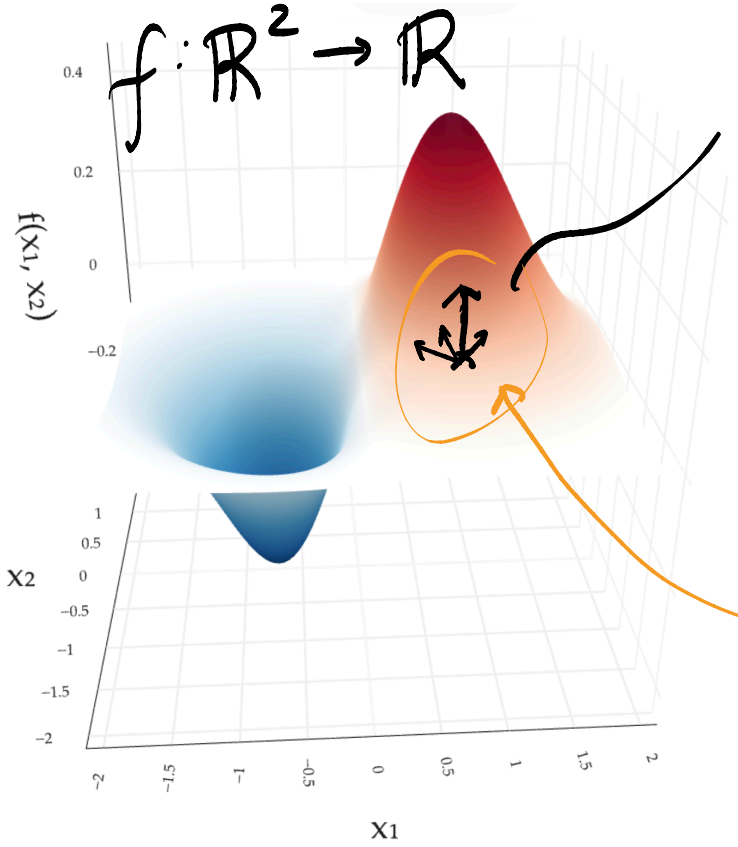
$f : \mathbb{R}^d \to \mathbb{R}$ "vector-to-scalar"

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

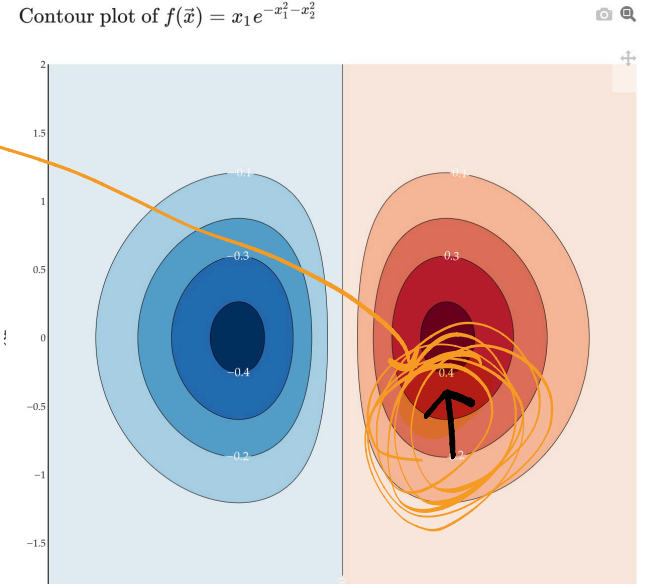
e.g. $f(\vec{x}) = (x_1 + x_2)^2 - 3\cos x_1$

$$\nabla f(\vec{x}) = \begin{bmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \end{bmatrix} = \begin{bmatrix} 2(x_1+x_2) + 3\sin x_1 \\ 2(x_1+x_2) \end{bmatrix}$$

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



function increases in lots of directions, but increases most quickly in the direction of the gradient.

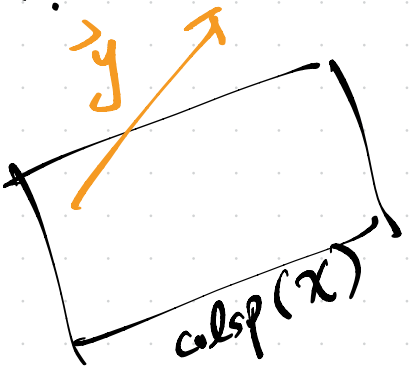


Ultimate goal: minimize mean squared error

$$R_{sq}(\vec{w}) = \frac{1}{n} \|\vec{y} - X\vec{w}\|^2$$

$\vec{y} - X\vec{w}$: error vector

Why? The $\vec{w}^*$ that minimizes $R_{sq}(\vec{w})$ will tell us how to make the best predictions!
($\vec{w}^*$ – optimal model parameters)



We know what the answer should be!

If X^TX invertible,

$$\vec{w}^* = (X^TX)^{-1}X^T\vec{y}$$

Recap: Big 3 gradient rules

① $f(\vec{x}) = \vec{a}^T\vec{x} = \vec{a}\cdot\vec{x}$ $\quad \vec{x}, \vec{a} \in \mathbb{R}^n$

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

② $f(\vec{x}) = \|\vec{x}\|^2$ $\quad \vec{x} \in \mathbb{R}^n$

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

③ $f(\vec{x}) = \vec{x}^T A \vec{x}$ "quadratic form" $\quad \vec{x} \in \mathbb{R}^n$, A: $n \times n$ matrix

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

e.g. $f(\vec{x}) = \|\vec{x}\|$

$\vec{x} \xrightarrow{\|\ \|^2} \|\vec{x}\|^2 \xrightarrow{\sqrt{\ }} \|\vec{x}\|$

g: vector to scalar

h: scalar to scalar

We would like to use the fact that

$\nabla(\|\vec{x}\|^2) = 2\vec{x}$

$f(\vec{x}) = h(g(\vec{x}))$

Recall, $\frac{d}{dx}\sqrt{x} = \frac{1}{2\sqrt{x}}$

$\nabla f(\vec{x}) = \frac{dh}{dx}(g(\vec{x}))\,\nabla g(\vec{x}) = h'(g(\vec{x}))\,\nabla g(\vec{x})$

Here, $\nabla\|\vec{x}\| = \nabla\left(\sqrt{\|\vec{x}\|^2}\right) = \frac{1}{2\sqrt{g(\vec{x})}}\nabla g(\vec{x}) = \frac{1}{2\|\vec{x}\|}\,2\vec{x} = \frac{\vec{x}}{\|\vec{x}\|}$

($\sqrt{g(\vec{x})} = \|\vec{x}\|$)

$f(\vec{x}) = \|\vec{x}\|^p$

$p \in \mathbb{R}$

Chapter 8.2

$f(\vec{x}) = h(g(\vec{x}))$

$h(x) = x^p \qquad g(\vec{x}) = \|\vec{x}\|$

$$\nabla f(\vec{x}) = \frac{dh}{dx}(g(\vec{x}))\,\nabla g(\vec{x})$$

$$= p\,g(\vec{x})^{p-1}\,\frac{\vec{x}}{\|\vec{x}\|} = \frac{p\,\|\vec{x}\|^{p-1}\,\vec{x}}{\|\vec{x}\|} = p\,\|\vec{x}\|^{p-2}\,\vec{x}$$

$$R_{sq}(\vec{w}) = \frac{1}{n}\|\vec{y} - X\vec{w}\|^2$$

$$= \frac{1}{n}(\vec{y} - X\vec{w}) \cdot (\vec{y} - X\vec{w})$$

$$= \frac{1}{n}\left(\vec{y}\cdot\vec{y} - \vec{y}\cdot(X\vec{w}) - (X\vec{w})\cdot\vec{y} + (X\vec{w})\cdot(X\vec{w})\right)$$

$$= \frac{1}{n}\left(\vec{y}\cdot\vec{y} - 2(X\vec{w})\cdot\vec{y} + (X\vec{w})^T(X\vec{w})\right)$$

$$= \frac{1}{n}\left(\vec{y}\cdot\vec{y} - 2\vec{w}^T(X^T\vec{y}) + \vec{w}^T X^T X \vec{w}\right)$$

want to find $\nabla R_{sq}(\vec{w})$

X : $n \times d$ matrix

$\vec{y} \in \mathbb{R}^n$

$\vec{w} \in \mathbb{R}^d$

$\vec{u} \cdot \vec{v} = \vec{u}^T \vec{v}$

Aside:

$$(X\vec{w}) \cdot \vec{y}$$

($X\vec{w}$ is $\vec{u}$, $\vec{y}$ is $\vec{v}$)

$$= \vec{w}^T X^T \vec{y}$$

so, $(X\vec{w}) \cdot \vec{y} = \vec{w} \cdot (X^T \vec{y})$

$$R_{sq}(\vec{w}) = \frac{1}{n}\left(\vec{y}\cdot\vec{y} - \underbrace{2\vec{w}^T(X^T\vec{y})}_{\vec{a}} + \vec{w}^T \boxed{X^T X}\vec{w}\right)$$

$$\nabla R_{sq}(\vec{w}) = \frac{1}{n}\left(-2X^T\vec{y} + \underbrace{\left((X^TX)+(X^TX)^T\right)}_{2X^TX}\vec{w}\right)$$

$$= -\frac{2}{n}\left(X^T\vec{y} - X^TX\vec{w}\right)$$

set $\nabla R_{sq}(\vec{w})$

$$-\frac{2}{n}\left(X^T\vec{y} - X^TX\vec{w}\right) = \vec{0}$$

$$X^T\vec{y} - X^TX\vec{w} = \vec{0}$$

the normal equation! $\rightarrow$ $X^TX\vec{w} = X^T\vec{y}$

same condition for $\vec{w}$ we saw before with orthogonality and projections!

$$X^T X \vec{w} = X^T \vec{y}$$

all $\vec{w}$'s that minimize

$$R_{sq}(\vec{w}) = \frac{1}{n} \|\vec{y} - X\vec{w}\|^2$$

satisfy

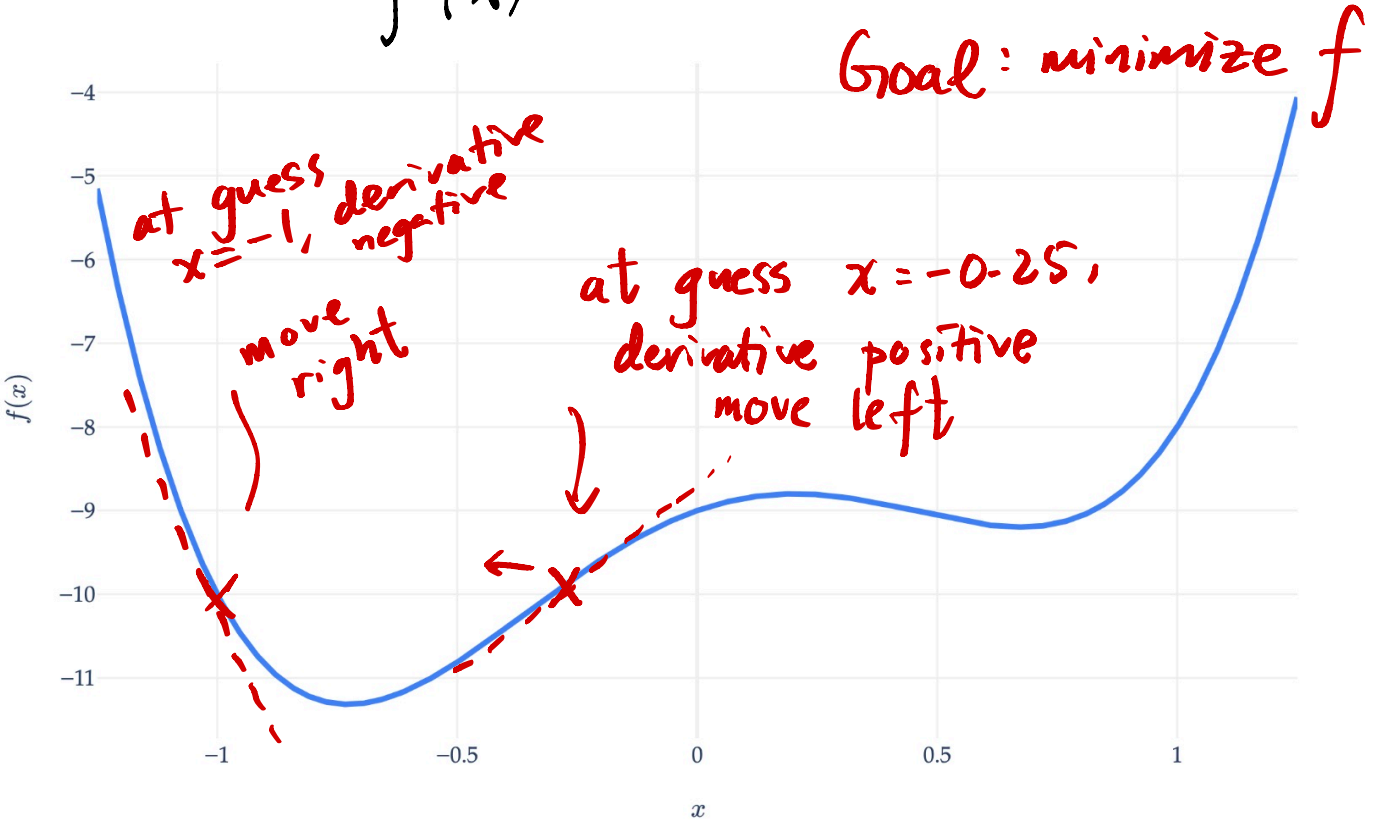
→ if $X^T X$ invertible, there is a unique best $\vec{w}^*$:

$$\vec{w}^* = (X^T X)^{-1} X^T \vec{y}$$

→ otherwise, infinitely many — see Chapter 6.4

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

$$f'(x) = 20x^3 - 3x^2 - 10x + 2$$



Gradient descent

Goal: Minimize $f : \mathbb{R}^d \to \mathbb{R}$, a differentiable function

First, choose initial guess, $\vec{x}^{(0)}$, and a learning rate / step size, $\alpha > 0$

Update guess using update rule:

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

vector

Terminate when $\| \nabla f(\vec{x}^{(t)}) \| \leq 0.001$ (tolerance)

Many examples
in Ch 8.3 – go play with them!

Chapter 8.3

Consider the following function.

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

1. Is $f(\vec{x})$ a quadratic form?
2. 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~~ one iterations of gradient descent. What is $\vec{x}^{(2)}$?

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

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

$$\vec{x}^{(0)} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad \alpha = \frac{1}{3}$$

$$\vec{x}^{(1)} = \vec{x}^{(0)} - \alpha \nabla f\left(\vec{x}^{(0)}\right)$$

$$= \begin{bmatrix} 0 \\ 0 \end{bmatrix} - \frac{1}{3}\begin{bmatrix} 2(-2) + 2 \\ -2(-3) \end{bmatrix} = \begin{bmatrix} 2/3 \\ -2 \end{bmatrix}$$

Issues with gradient descent:

numerical method

- Can get trapped at a local minimum
- step size too large → may not converge

Gradient descent is usually used for finding optimal model parameters, i.e. for empirical risk minimization

e.g. $R_{sq}(\vec{w}) = \frac{1}{n} \|\vec{y} - X\vec{w}\|^2$

since we have a "closed-form" solution for $\vec{w}^*$, we don't strictly need GD here

Let's try it!

$$R_{sq}(\vec{w}) = \frac{1}{n} \|\vec{y} - X\vec{w}\|^2$$

$$\nabla R_{sq}(\vec{w}) = \underbrace{-\frac{2}{n}\left(X^T\vec{y} - X^TX\vec{w}\right)}_{\text{from earlier today}}$$

Start with $\vec{w}^{(0)}$

$$\vec{w}^{(t+1)} = \vec{w}^{(t)} - \alpha\left(-\frac{2}{n}\left(X^T\vec{y} - X^TX\vec{w}^{(t)}\right)\right)$$

Update until $\|\nabla R_{sq}(\vec{w}^{(t)})\| \leq$ tolerance

```
t =     0 | w^(0) = [   0.000,    0.000] | mse = 5523.7231 | ||grad|| = 1229.8426
t =   500 | w^(500) = [  19.804,    0.102] | mse =  314.8511 | ||grad|| =  3.5279
t =  1000 | w^(1000) = [  36.133,    4.200] | mse =  260.7135 | ||grad|| =  3.0582
t =  1500 | w^(1500) = [  50.289,    2.551] | mse =  220.0323 | ||grad|| =  2.6510
t =  2000 | w^(2000) = [  62.559,    1.121] | mse =  189.4630 | ||grad|| =  2.2980
t =  2500 | w^(2500) = [  73.196,   -0.118] | mse =  166.4919 | ||grad|| =  1.9920
t =  3000 | w^(3000) = [  82.416,   -1.193] | mse =  149.2306 | ||grad|| =  1.7268
t =  3500 | w^(3500) = [  90.409,   -2.124] | mse =  136.2598 | ||grad|| =  1.4969
t =  4000 | w^(4000) = [  97.338,   -2.931] | mse =  126.5130 | ||grad|| =  1.2976
t =  4500 | w^(4500) = [ 103.344,   -3.631] | mse =  119.1889 | ||grad|| =  1.1248
t =  5000 | w^(5000) = [ 108.551,   -4.237] | mse =  113.6852 | ||grad|| =  0.9750
t =  5500 | w^(5500) = [ 113.064,   -4.763] | mse =  109.5496 | ||grad|| =  0.8452
t =  6000 | w^(6000) = [ 116.976,   -5.219] | mse =  106.4419 | ||grad|| =  0.7327
t =  6500 | w^(6500) = [ 120.368,   -5.614] | mse =  104.1067 | ||grad|| =  0.6351
t =  7000 | w^(7000) = [ 123.308,   -5.957] | mse =  102.3519 | ||grad|| =  0.5505
t =  7500 | w^(7500) = [ 125.856,   -6.254] | mse =  101.0333 | ||grad|| =  0.4772
t =  8000 | w^(8000) = [ 128.065,   -6.511] | mse =  100.0447 | ||grad|| =  0.4137
t =  8500 | w^(8500) = [ 129.980,   -6.734] | mse =   99.3078 | ||grad|| =  0.3586
t =  9000 | w^(9000) = [ 131.640,   -6.928] | mse =   98.7383 | ||grad|| =  0.3108
t =  9500 | w^(9500) = [ 133.079,   -7.095] | mse =   98.3179 | ||grad|| =  0.2694
t = 10000 | w^(10000) = [ 134.327,   -7.241] | mse =   98.0020 | ||grad|| =  0.2336
t = 10500 | w^(10500) = [ 135.407,   -7.367] | mse =   97.7646 | ||grad|| =  0.2025
t = 11000 | w^(11000) = [ 136.345,   -7.476] | mse =   97.5862 | ||grad|| =  0.1755
t = 11500 | w^(11500) = [ 137.158,   -7.571] | mse =   97.4521 | ||grad|| =  0.1521
t = 12000 | w^(12000) = [ 137.862,   -7.653] | mse =   97.3514 | ||grad|| =  0.1319
t = 12500 | w^(12500) = [ 138.473,   -7.724] | mse =   97.2757 | ||grad|| =  0.1143
t = 13000 | w^(13000) = [ 139.002,   -7.785] | mse =   97.2188 | ||grad|| =  0.0991
t = 13500 | w^(13500) = [ 139.461,   -7.839] | mse =   97.1761 | ||grad|| =  0.0859
t = 14000 | w^(14000) = [ 139.859,   -7.885] | mse =   97.1440 | ||grad|| =  0.0744
```

guesses change a lot

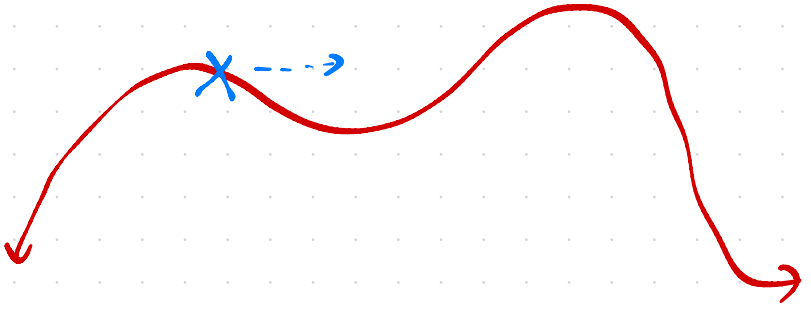
guesses change very little

approaching theoretical best MSE

!!grad!! → 0

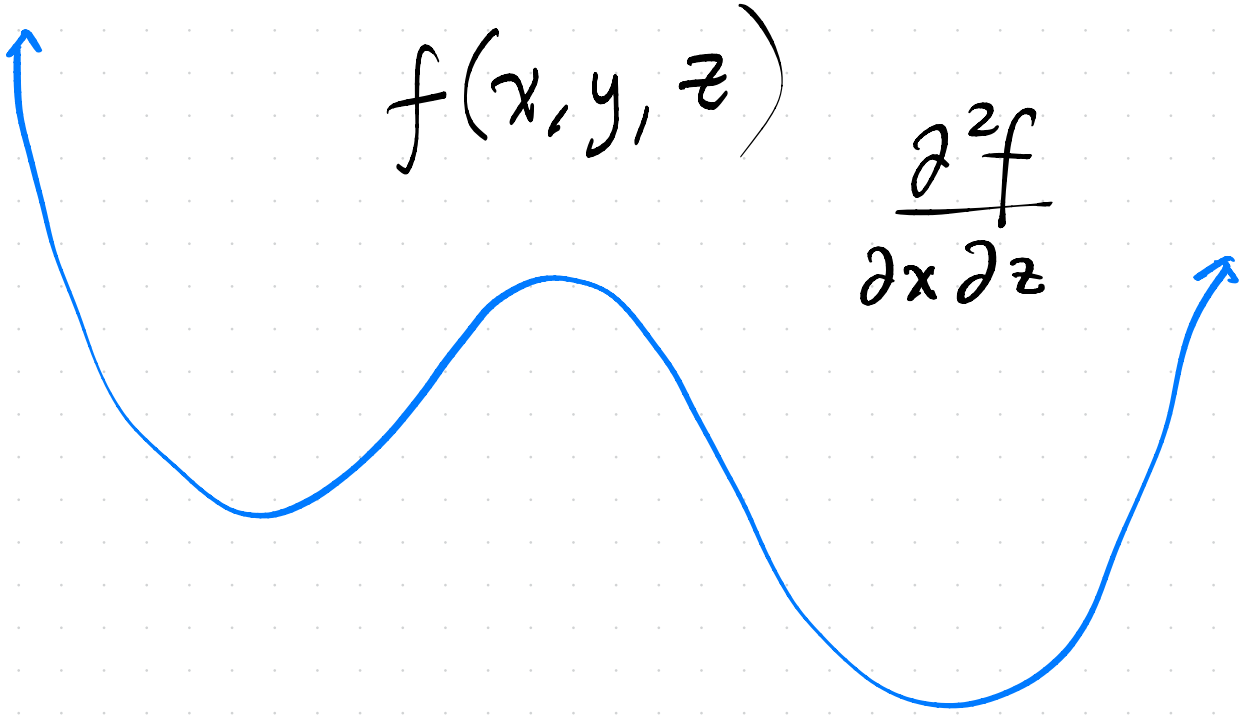
CONTENTS

- Simple Linear R
- Another Examp
- Issues of Scale

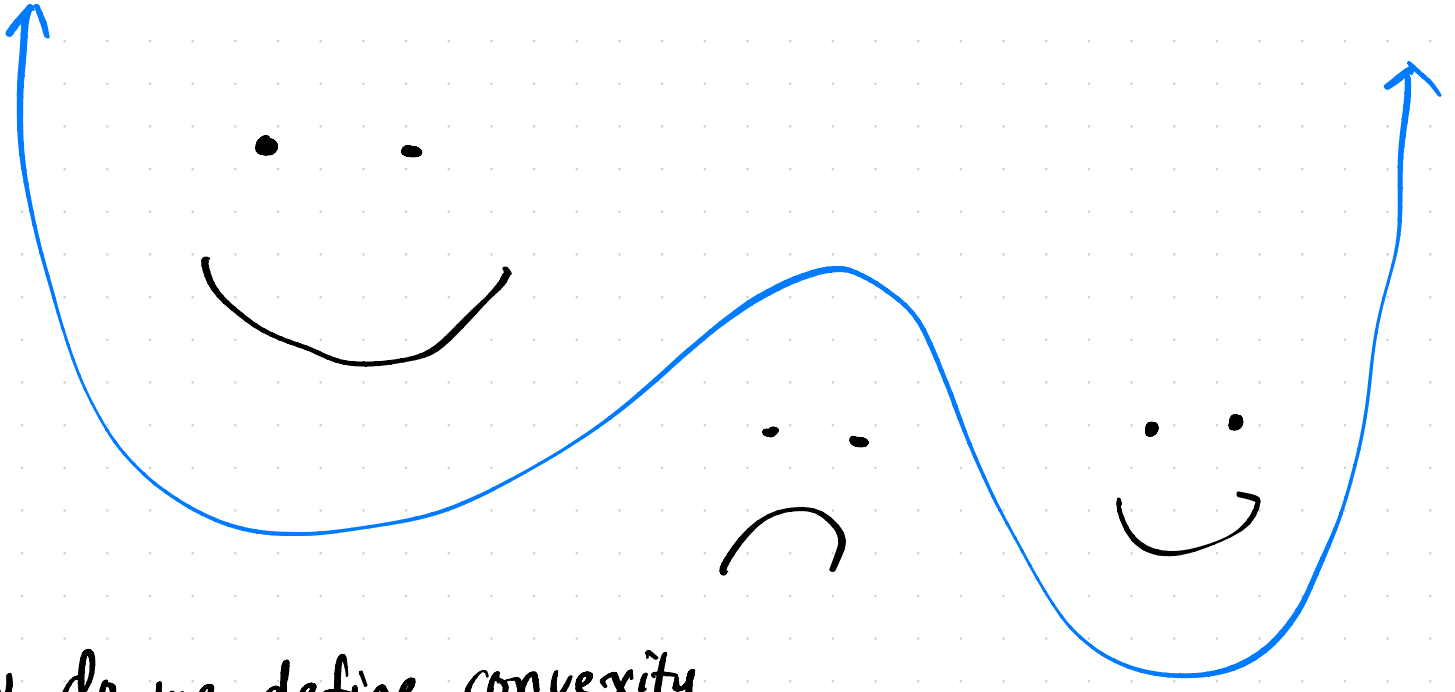


Recap: we use gradient descent to minimize differentiable functions (vector-to-scalar or scalar-to-scalar) when we don't have a closed-form minimizer.

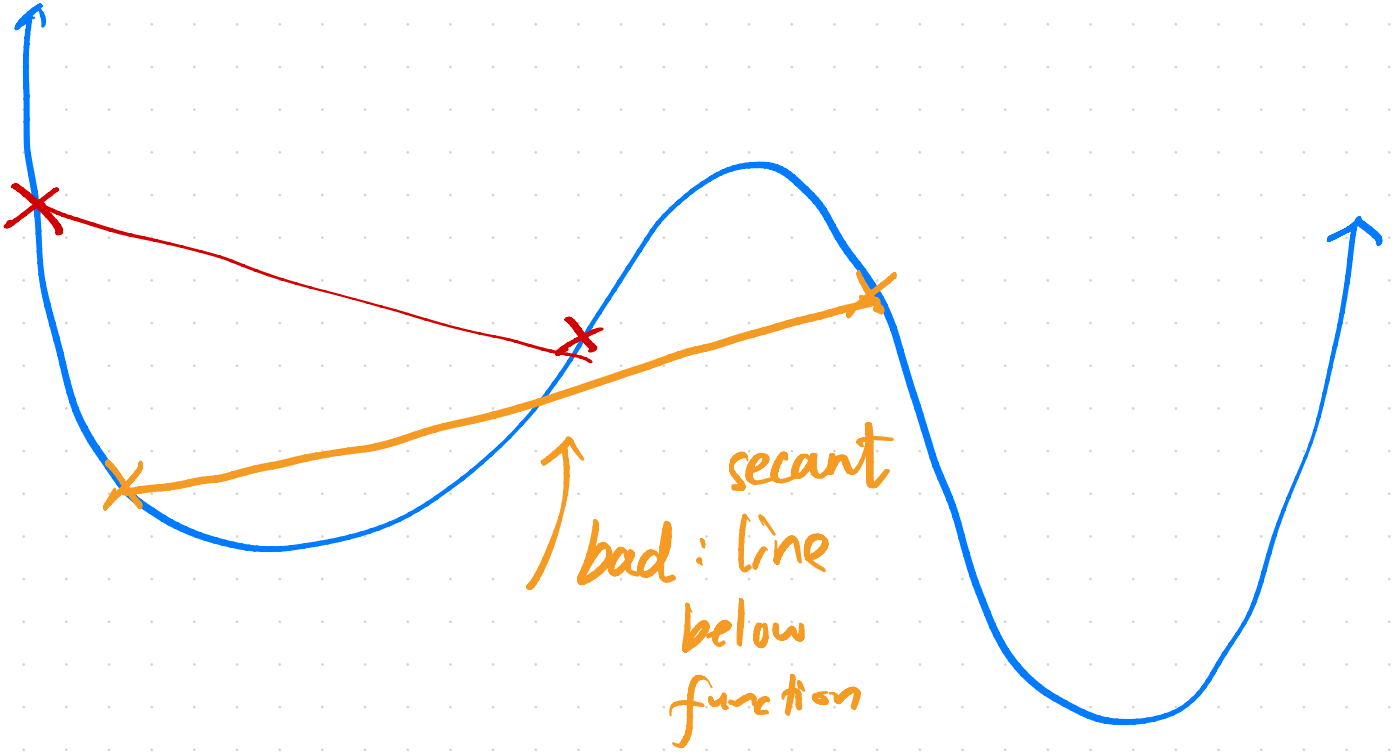
Usually, the function is average loss / empirical risk, so the minimizer is a vector of optimal model parameters.



→ not on MT 2,
but on final exam



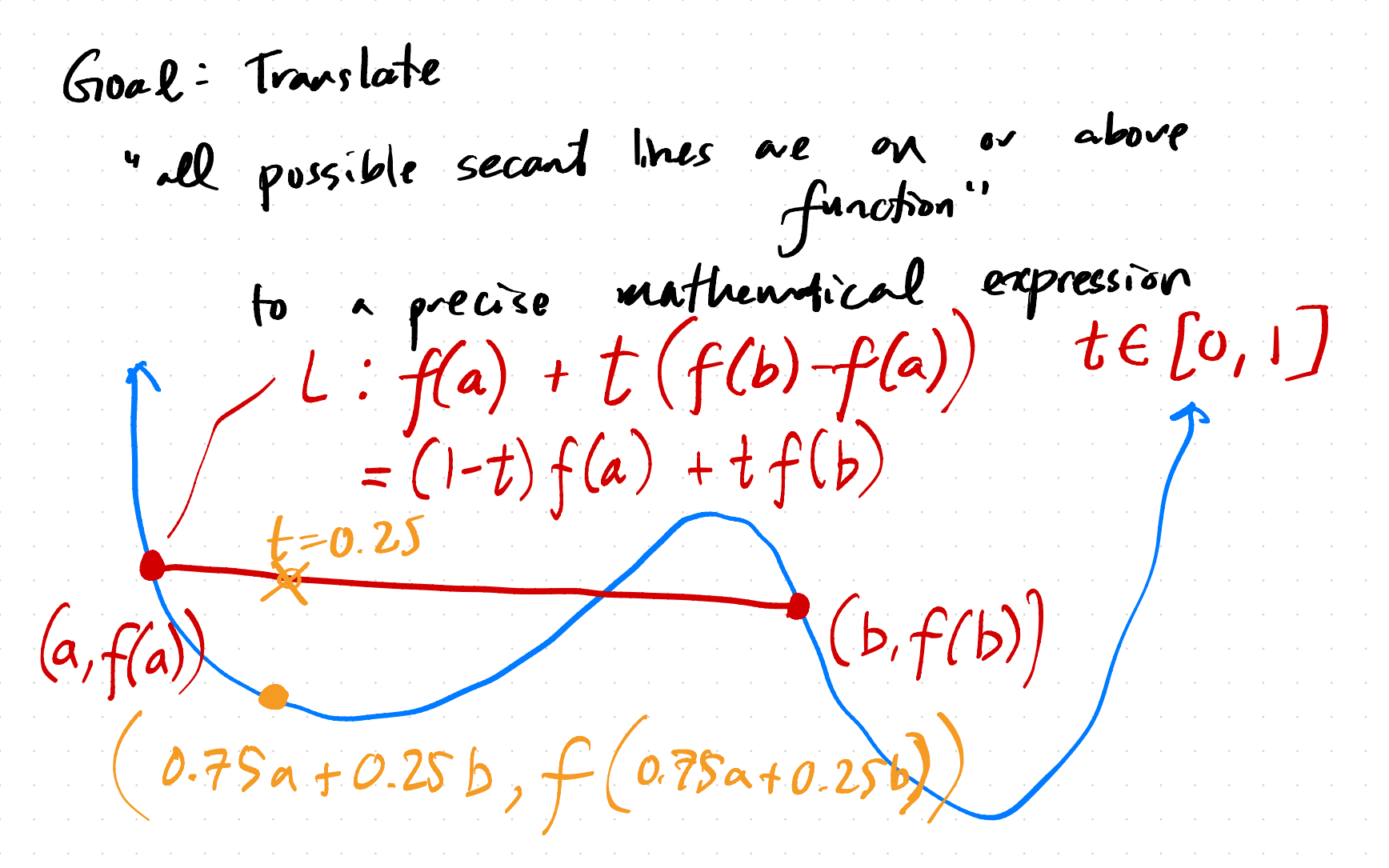
How do we define convexity without using derivatives?


secant
bad: line
below
function

Goal: Translate

"all possible secant lines are on or above function"

to a precise mathematical expression



Suppose $f : \mathbb{R}^d \to \mathbb{R}$ is a function.

f is convex if

$$f\big((1-t)\vec{x} + t\vec{y}\big) \leq (1-t)f(\vec{x}) + t f(\vec{y})$$

f: function; $(1-t)f(\vec{x}) + t f(\vec{y})$: line

for all $t \in [0,1]$

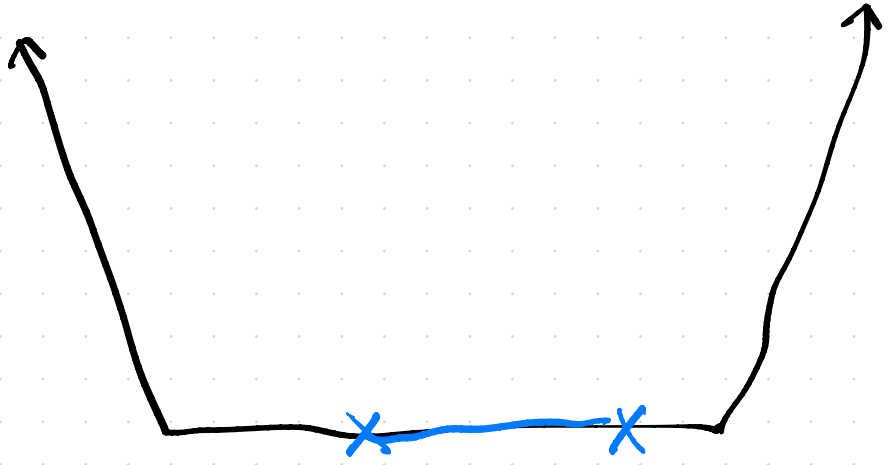
and all $\vec{x}, \vec{y}$ in domain of f.

Why is this useful?

- If we can prove function convex, it doesn't have any "trap" local minimums

→ GD won't get stuck

→ see video in 8.5



convex ?

yes !

strictly convex ?

no !

e.g. mean squared error,

$$R_{sq}(\vec{w}) = \frac{1}{n} \|\vec{y} - X\vec{w}\|^2$$

always convex

only strictly convex if X's columns are linearly independent.