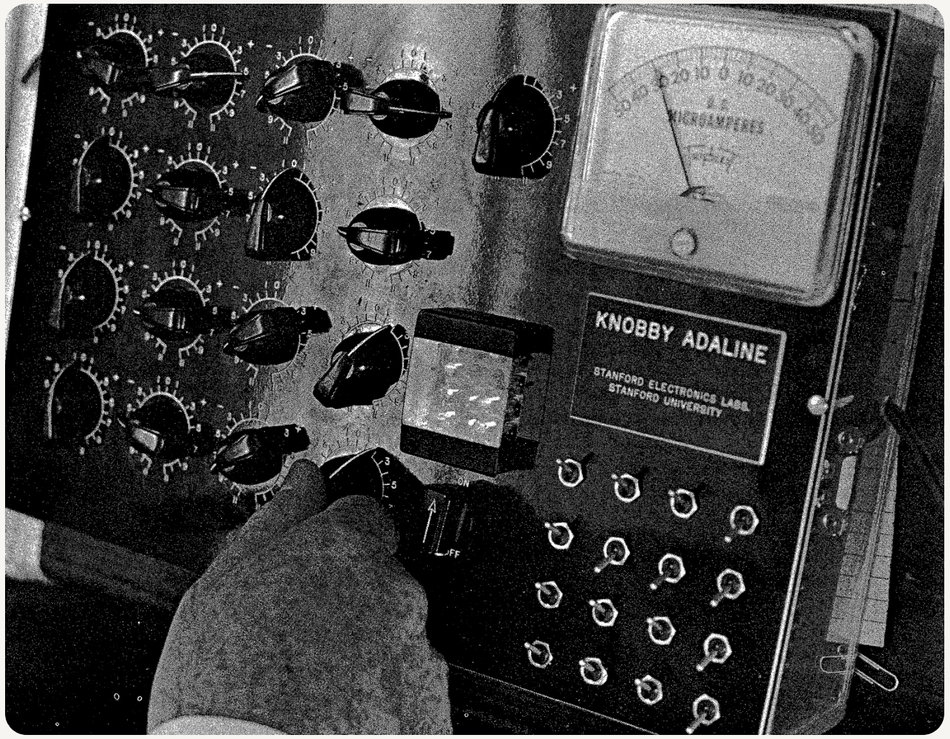
SIENA, NIPS-CERN, UFJF

Limiar

Redes neurais, do neurônio ao fractal

Aula 2: derivada, gradiente e gradiente descendente

02/10/2026. C∴ Chrysthofer A. A. Afonso



A Knobby Adaline, de Widrow e Hoff, que aprendia pelo gradiente: os botões são os pesos, e o medidor, o erro. Stanford Today, 1963, domínio público nos Estados Unidos

1 · REVISÃO

O que ficou da aula 1, de memória

1 O que o neurônio calcula?

2 O que a regra faz quando erra?

3 Por que o treino do E termina?

4 E no XOR?

A regra do perceptron aprende o E

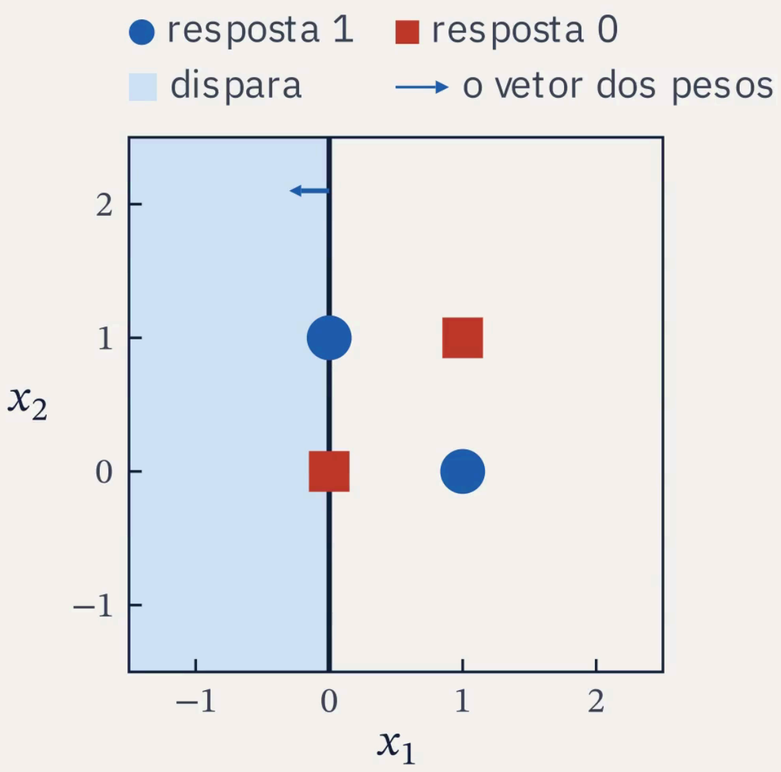
$$\vec{w} \leftarrow \vec{w} + \eta\,(t - y)\,\vec{x}$$

$$b \leftarrow b + \eta\,(t - y)$$

Época 6: nenhum erro.

época	erros	$\vec{w}$	b
1	2	(1, 1)	0
2	3	(2, 1)	−1
3	3	(2, 1)	−2
4	2	(2, 2)	−2
5	1	(2, 1)	−3
6	0	(2, 1)	−3

No XOR, a regra nunca para

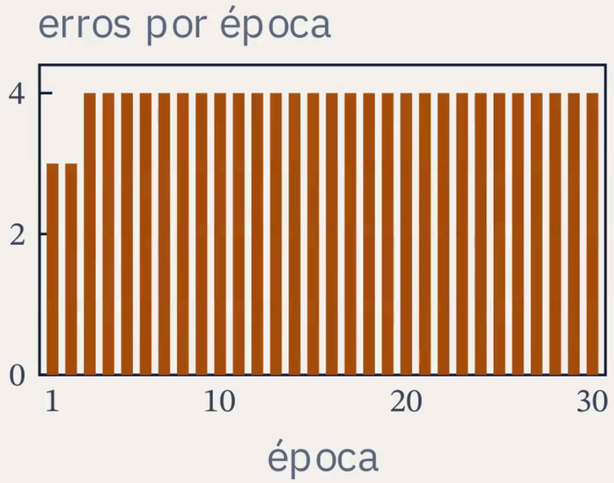


Época 30: ainda 4 erros.

Nenhuma reta separa o XOR.

$\vec{w} = (-1,\ 0) \qquad b = 0$

iguais no fim de toda época



2 · SEMINÁRIO

Os vídeos e o perceptron de E e OU

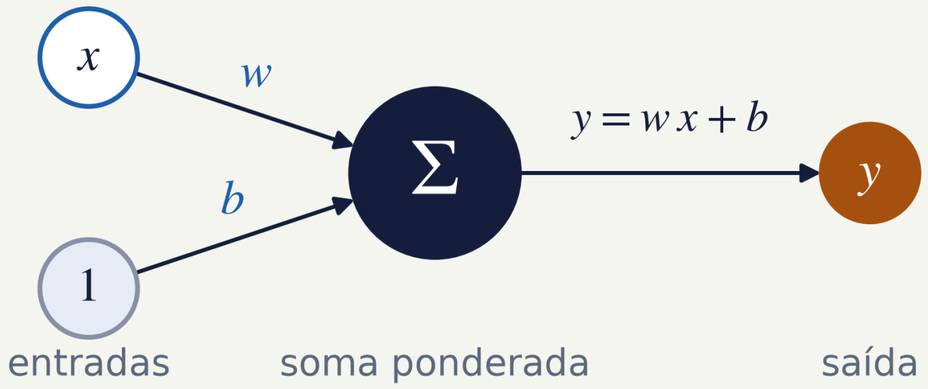
Os vídeos

- neurônio entre 0 e 1, contra 0 ou 1: o que muda?
- o que o vídeo chama de custo?
- por que a derivada num instante parece um paradoxo?

O código

- o perceptron de E e OU, linha a linha
- a tabela bate com a do slide das épocas?
- uma pergunta de cada um

O neurônio sem o degrau

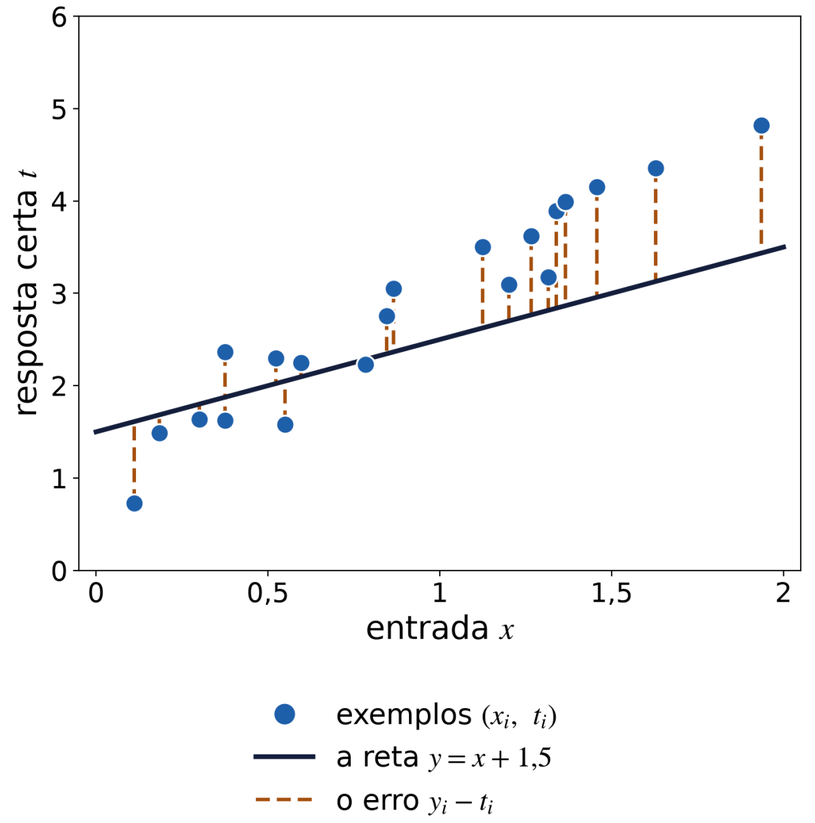


$$y = w\,x + b$$

Com uma entrada, o neurônio é uma reta: w é a inclinação, e b, a altura em que ela corta o eixo.

Sem o degrau, o erro $y - t$ tem tamanho: diz quanto e para que lado o neurônio errou.

O erro quadrático médio



$$L(w,\ b) = \frac{1}{N}\sum_{i=1}^{N}(y_i - t_i)^2$$

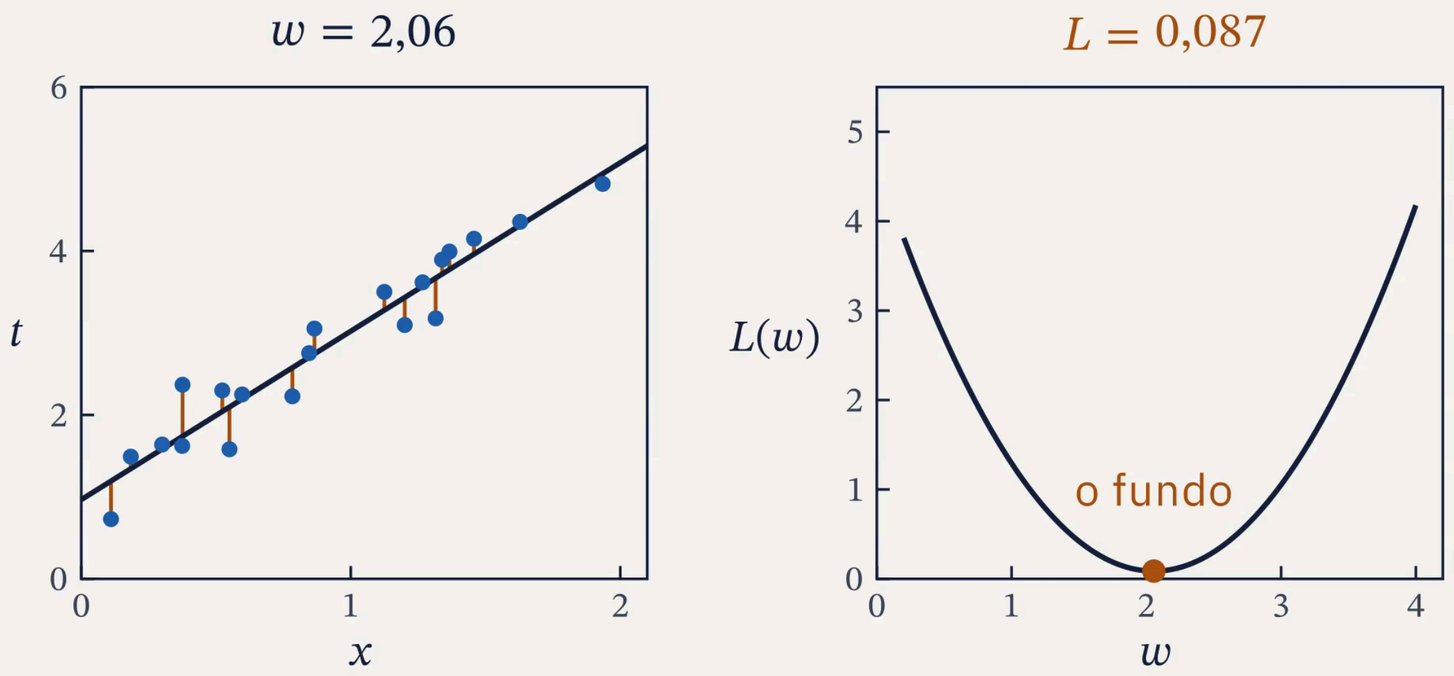
L de *w* e *b* é a média, de *i* igual a 1 até *N*, dos quadrados dos erros

$$y_i = w\,x_i + b$$

A reta da figura, $y = x + 1{,}5$, tem $L = 0{,}556$. A melhor reta tem $L = 0{,}087$.

A perda depende do peso

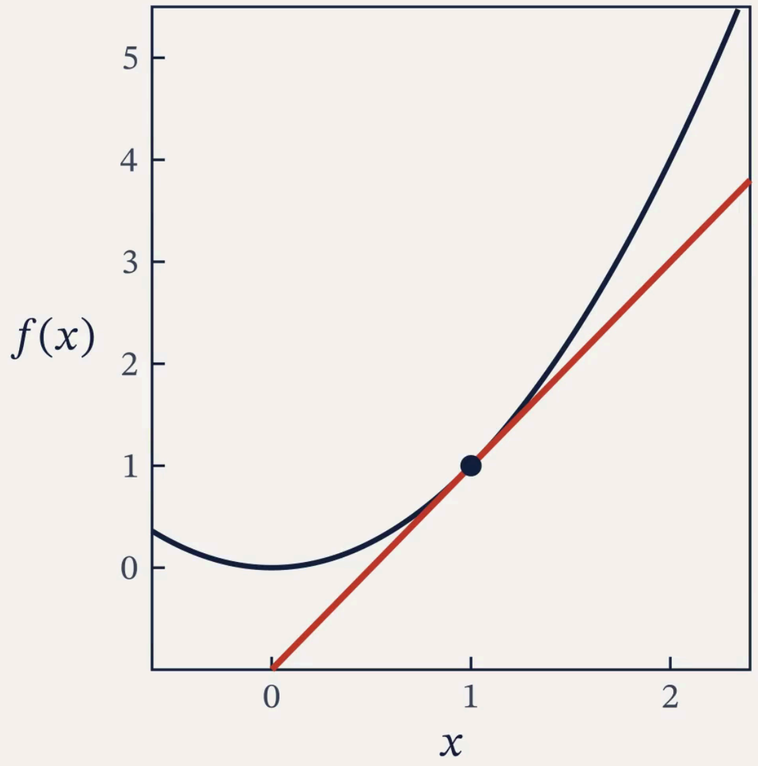
No fundo da curva, a melhor reta: $w = 2{,}056$



3 · O neurônio sem o degrau

Para que lado mexer w, e quanto, para a perda cair?

A derivada: quanto a saída muda



$$f(x) = x^2$$

$$f'(1) = \lim_{h \to 0} \frac{f(1+h) - f(1)}{h} = 2$$

a tangente

A derivada de x^2 pela definição

$$\frac{(x+h)^2 - x^2}{h} = \frac{2xh + h^2}{h} = 2x + h$$

$$f(x) = x^2 \;\Rightarrow\; f'(x) = 2x$$

TRÊS REGRAS QUE BASTAM

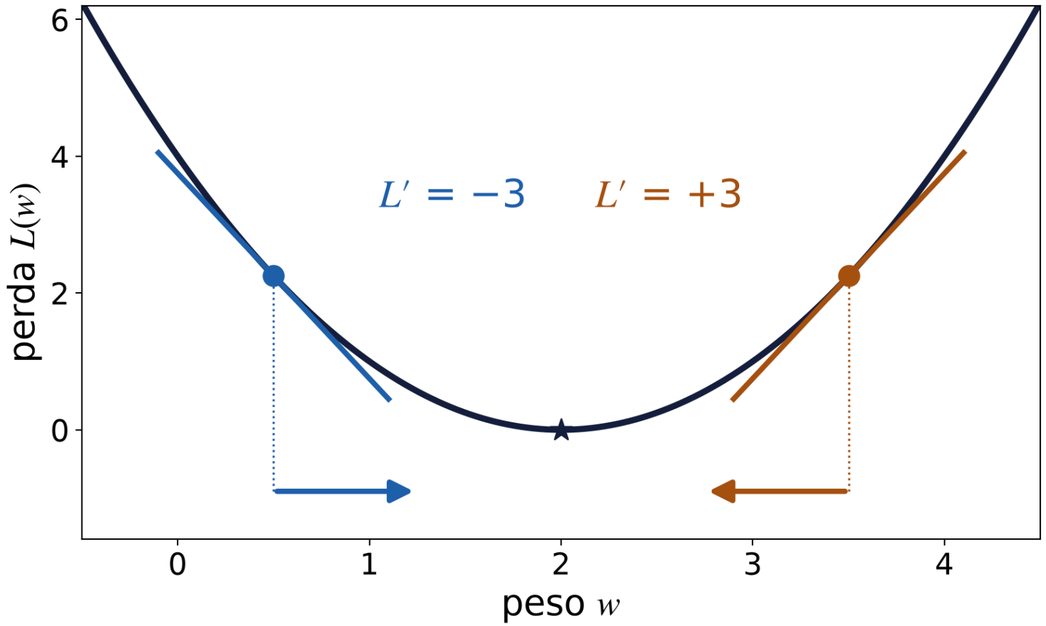
$(c)' = 0 \qquad (c\,w)' = c \qquad (f+g)' = f' + g'$

c é um número fixo; *f* e *g*, duas funções

h	$\frac{(1+h)^2 - 1}{h}$
1	3
0,5	2,5
0,1	2,1
0,01	2,01

Em x = 1, a razão vale 2 + h: a derivada numérica.

A derivada diz para onde descer



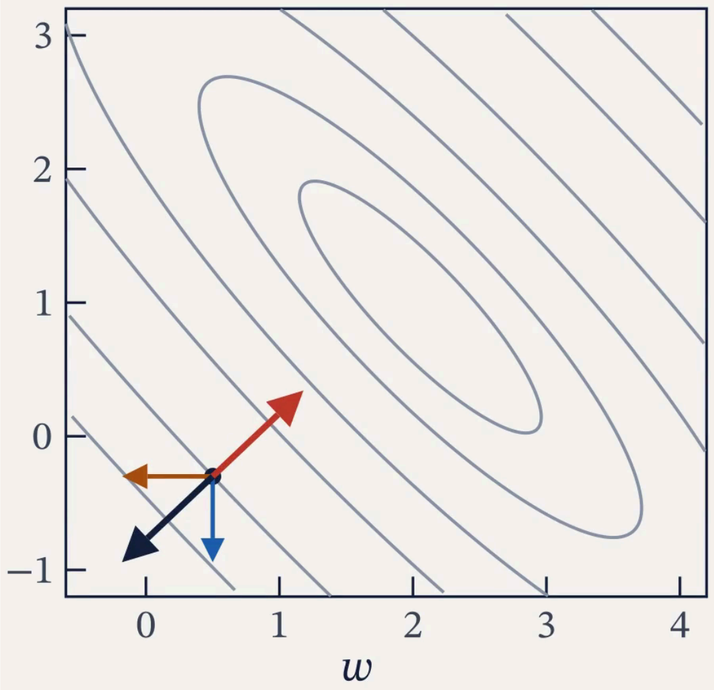
$$w \leftarrow w - \eta\, L'(w)$$

$L' < 0$: a perda cai para a direita, e w aumenta.

$L' > 0$: a perda cai para a esquerda, e w diminui.

Longe do fundo, a inclinação é grande, e o passo também; perto, os passos encolhem sozinhos.

Derivada parcial: um corte na superfície



$$\vec{\nabla} L = (-5{,}66;\ -5{,}36)$$

→ $\partial L/\partial w = -5{,}66$

→ $\partial L/\partial b = -5{,}36$

→ o gradiente: a subida

→ contra ele: a descida

As duas derivadas parciais da perda

$$\frac{\partial L}{\partial w} = \frac{2}{N}\sum_{i=1}^{N}(y_i - t_i)\,x_i$$

derivada parcial de L em relação a w: só w varia, e b fica parado

$$\frac{\partial L}{\partial b} = \frac{2}{N}\sum_{i=1}^{N}(y_i - t_i)$$

DE ONDE VEM

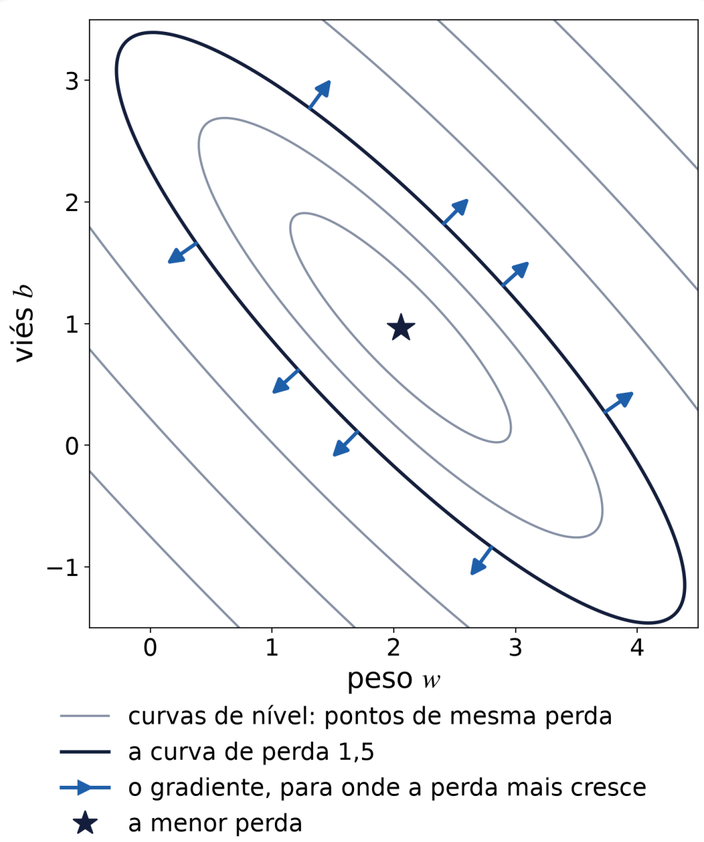
$$(wx_i + b - t_i)^2 = w^2x_i^2 + 2wx_i(b - t_i) + (b - t_i)^2$$

é uma parábola em w; derivando em w, sai $2(y_i - t_i)\,x_i$

CONFERIDO

em (0,5; −0,3), a fórmula e a derivada numérica dão (−5,6643; −5,3552)

O gradiente



$$\vec{\nabla} L = \left(\frac{\partial L}{\partial w}, \frac{\partial L}{\partial b} \right)$$

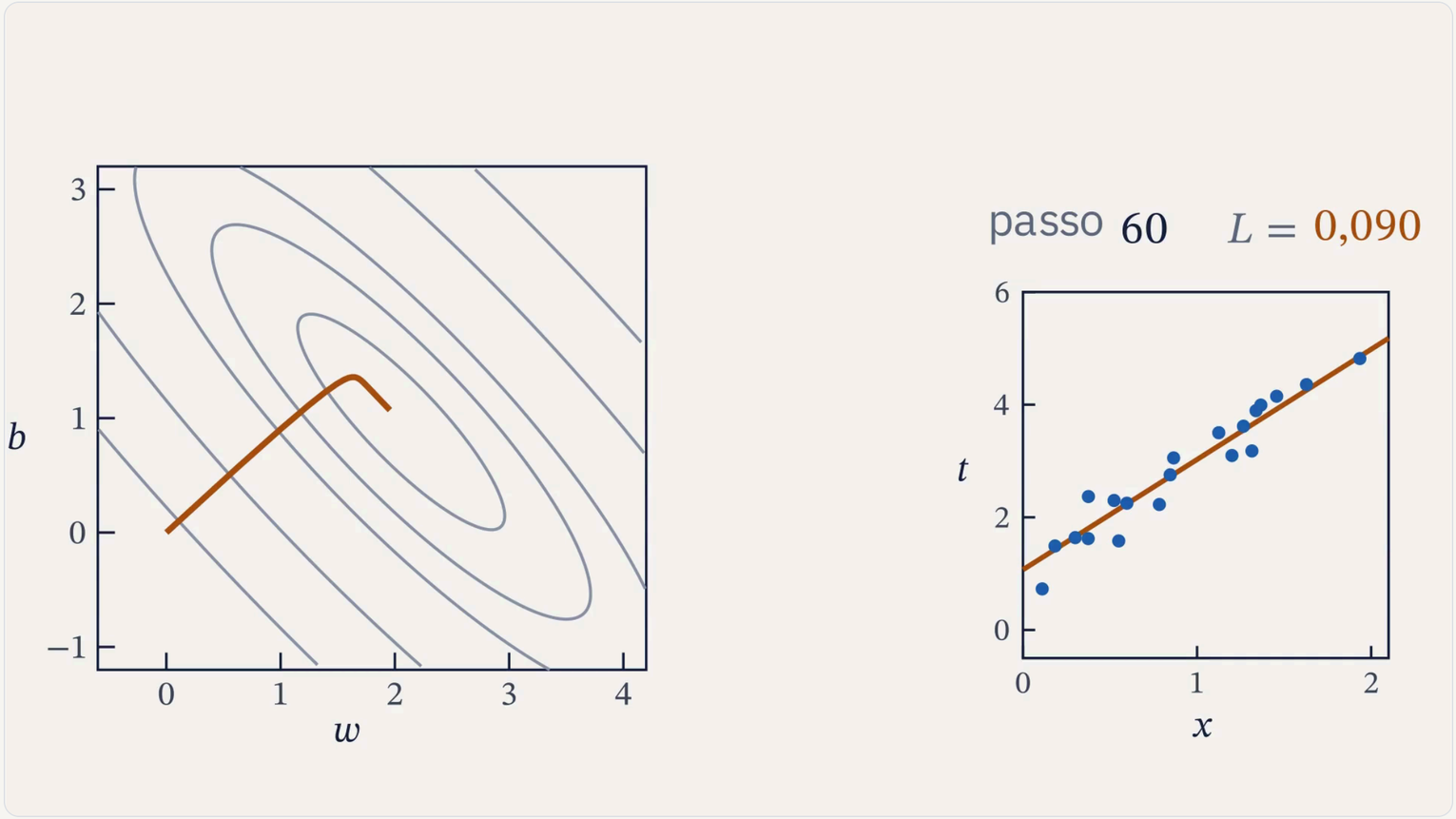
nabla L, o gradiente: o vetor das derivadas parciais

Aponta para onde a perda mais cresce. Descer mais depressa é andar contra ele.

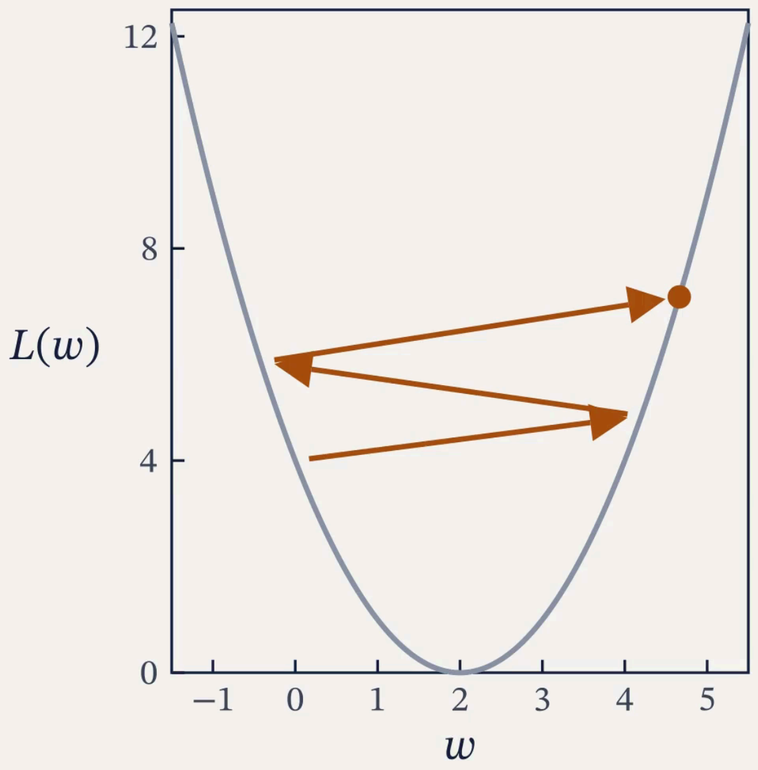
DE ONDE VEM O NOME: GRADIENTE

do latim gradiens, que anda, de gradior, dar passos, de gradus, passo; a mesma raiz de degrau, a função da aula 1.

Gradiente descendente



A taxa de aprendizado



$$w \leftarrow w - \eta\, L'(w)$$

$$L'(w) = 2\,(w - 2)$$

$\eta = 0{,}1$ devagar: fator 0,8

$w - 2$: -2 $-1{,}6$ $-1{,}28$ $-1{,}02$

$\eta = 0{,}9$ oscila e chega: fator $-0{,}8$

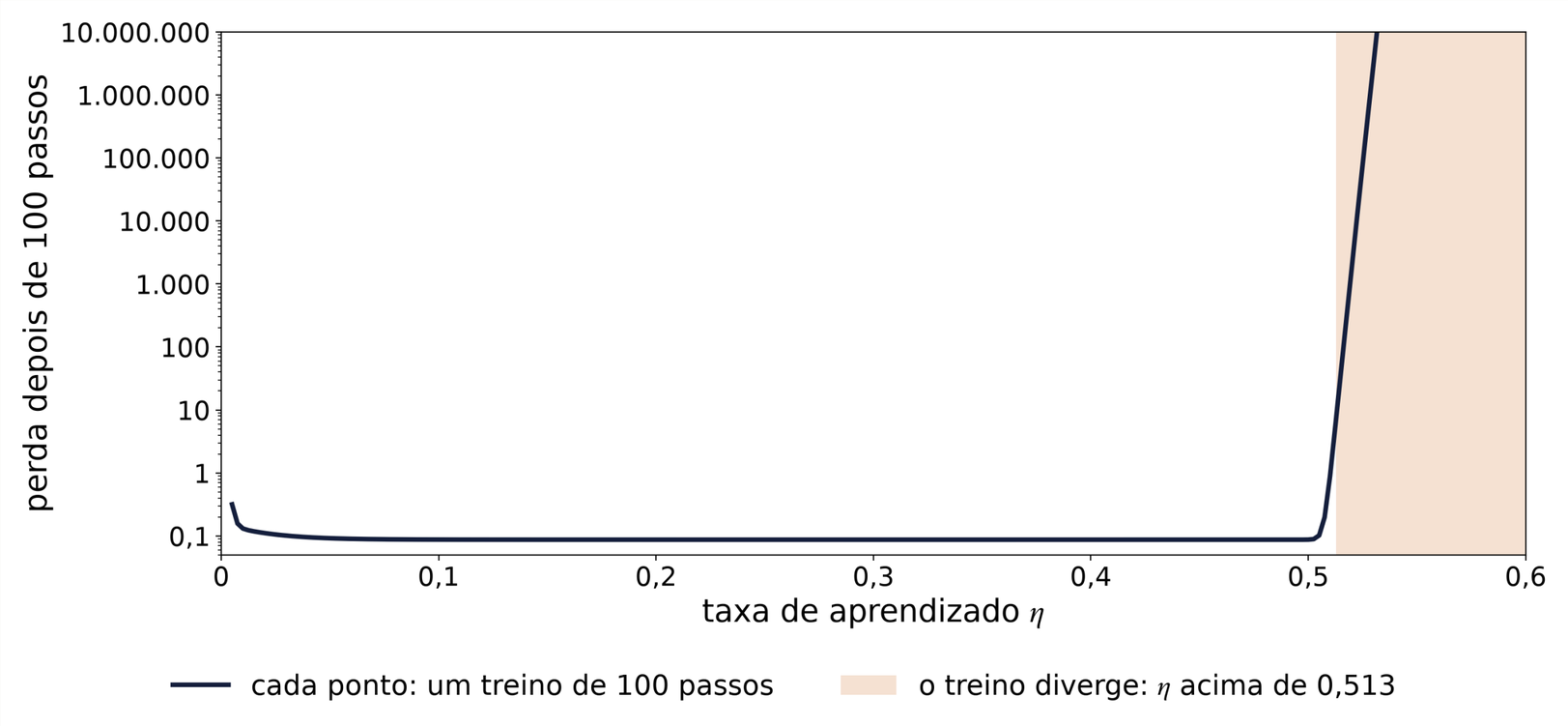
$w - 2$: -2 $1{,}6$ $-1{,}28$ $1{,}02$

$\eta = 1{,}05$ diverge: fator $-1{,}1$

$w - 2$: -2 $2{,}2$ $-2{,}42$ $2{,}66$

$$|1 - 2\eta| < 1 \iff 0 < \eta < 1$$

Um limiar de novo



Com a reta, o treino converge até η ≈ 0,513 e explode acima. No projeto de dezembro, uma taxa por camada, e a fronteira vira uma curva: no artigo de Sohl-Dickstein, fractal.

1847: Cauchy e as órbitas

> *“Para achar os valores de x, y, z, … que satisfazem a equação u = 0, basta fazer a função u decrescer indefinidamente, até que ela se anule.”*
>
> Augustin-Louis Cauchy, Comptes rendus da Academia de Ciências de Paris, 18 de outubro de 1847, tradução livre

- o problema: a órbita de um astro, com seis incógnitas
- o passo: cada variável anda contra a própria derivada parcial
- o tamanho do passo, pequeno o bastante: a taxa de aprendizado

$$x \leftarrow x - \theta X$$ *θ*, teta, é o passo; *X*, a derivada parcial em *x*

Augustin-Louis Cauchy, 1789 a 1857. Louis Grégoire e Deneux, por volta de 1840, domínio público

7 · DE ONDE VEM

1960: a Adaline de Widrow e Hoff

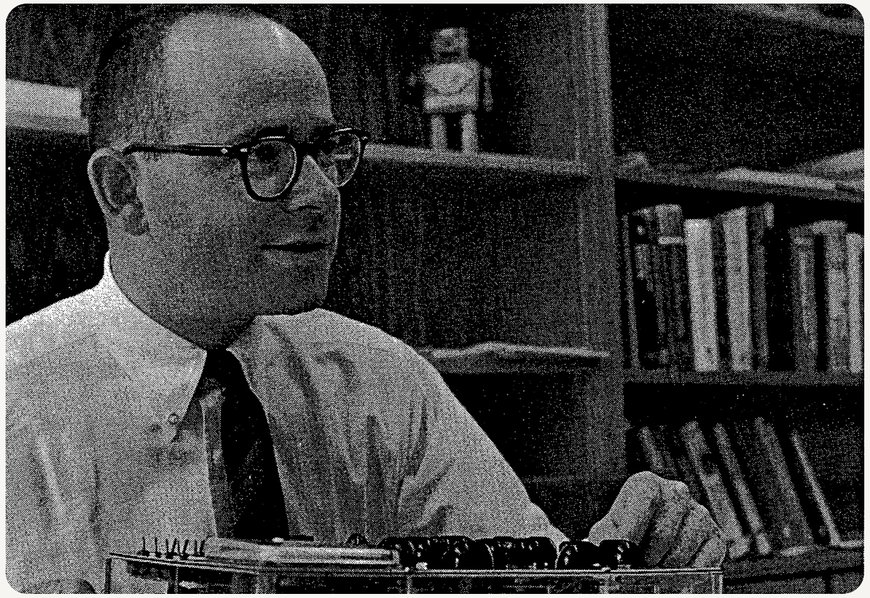
- Adaline, de adaptive linear: um neurônio linear em circuito
- 16 chaves de entrada, 17 pesos em botões e um medidor do erro
- do tamanho de uma lancheira, nas palavras dos autores

A REGRA, UM EXEMPLO POR VEZ

$$\vec{w} \leftarrow \vec{w} + \eta\,(t - z)\,\vec{x}$$

a do perceptron com *z*, a soma antes do degrau, no lugar de *y*

Virou o algoritmo LMS e o começo dos filtros adaptativos: aqui, caixa-preta, só a história.



Bernard Widrow com uma Adaline. Stanford Today, 1963, domínio público nos Estados Unidos

8 · O CÓDIGO

No laptop: a reta por gradiente descendente

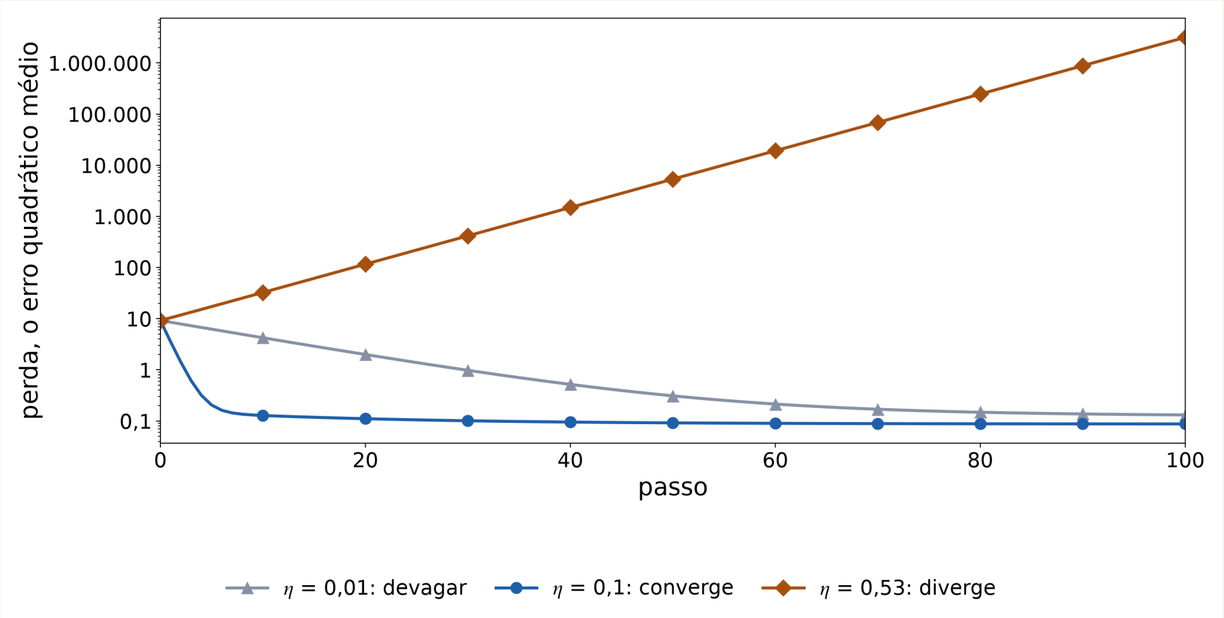
1. Sortear 20 pontos em volta de $t = 2x + 1$, com a semente fixa
2. Escrever a perda, o erro quadrático médio
3. Escrever as duas derivadas parciais e conferir com a derivada numérica
4. O laço: 100 passos, guardando a perda de cada um
5. Treinar com várias taxas e achar a que diverge

O NUMPY QUE RESOLVE

```
rng = np.random.default_rng(2)
x = rng.uniform(0, 2, 20)
y = w * x + b
np.mean((y - t) ** 2)
```

com arrays, a conta vale para os 20 pontos de uma vez, sem laço

O que a entrega pede



A perda a cada passo, uma curva por taxa, e a taxa a partir da qual o treino diverge. Cada um com os próprios pontos: o limiar muda com eles.

```
python aulas/02-gradiente/codigo/reta.py
```

9 · FECHAMENTO

Até 16/10: duas semanas, sem aula em 09/10

Estudo, até 16/10

- 3Blue1Brown, álgebra linear, capítulo 3
- 3Blue1Brown, álgebra linear, capítulo 4
- Nielsen, capítulo 1, até a arquitetura das redes

Entrega até 16/10

- a perda a cada passo, para várias taxas
- a taxa a partir da qual o treino diverge
- uma pergunta sobre o estudo

Seminário da aula 3

- o código da entrega, linha a linha
- o que é uma transformação linear?
- por que uma matriz guarda uma transformação?
- multiplicar matrizes: uma depois da outra