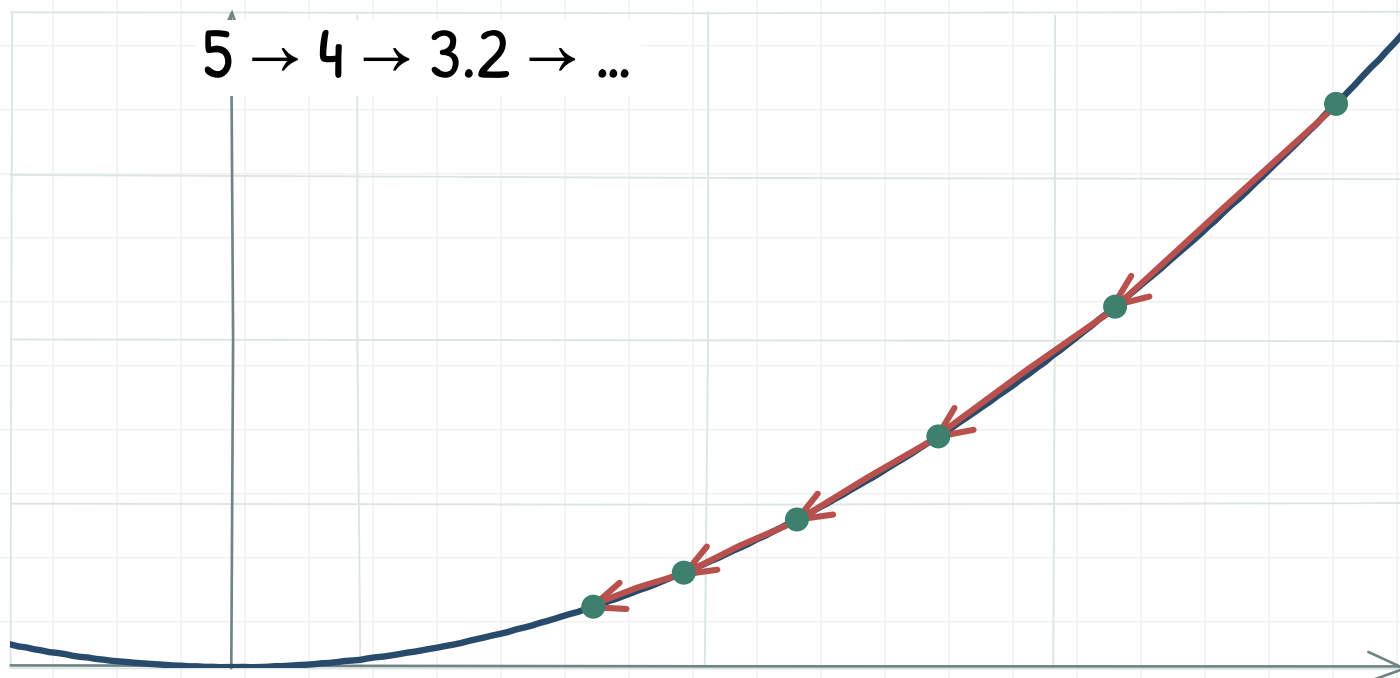


Độ dốc chỉ đường, lr chọn bước

Đứng trên $f(x)=x^2$, đo độ dốc rồi bước ngược lại.



Từ hình đến công thức

$\theta_{\text{mới}} = \theta - lr \cdot \nabla f(\theta)$, với $lr > 0$.

Tại $x=5$, gradient=10. Nếu $lr=0.1$ thì $x_{\text{mới}}=5-1=4$.

Độ dài bước = $lr \cdot \|\nabla f\|$. Gradient đổi theo vị trí nên bước không cố định.

GD chỉ nhìn địa hình tại chỗ, chưa biết đáy nằm ở đâu.

Qua đáy chưa chắc phân kỳ

Với x^2 , một bước chỉ là nhân x với $q=1-2lr$.

lr	q	Hành vi
0.1	+0.8	co cùng phía
0.5	0	đến đáy ngay
0.9	-0.8	co, đổi phía
1.0 / 1.1	-1 / -1.2	dao động / nổ

Từ hình đến công thức

$x_k = q^k x_0$. Từ mọi x_0 , hội tụ về 0 khi $|q| < 1 \Leftrightarrow 0 < lr < 1$.

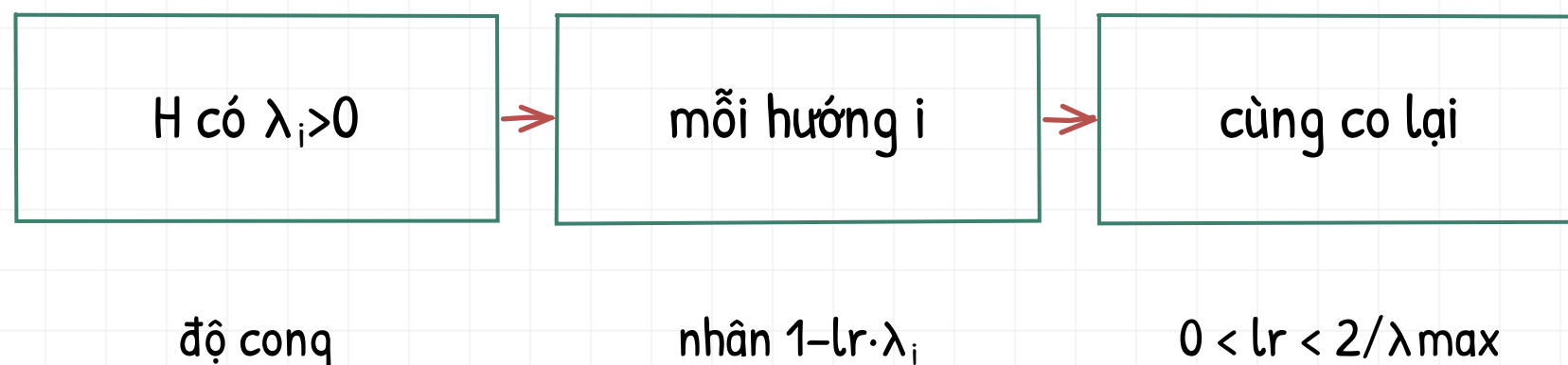
$lr=0.1$ và 0.9 có cùng tốc độ co: từ 5, sau 10 bước đều được 0.5369.

$lr=1.1$ cho $x_{10} \approx 30.9587$: đi qua đáy nhưng mỗi lần xa thêm.

Phân biệt đổi dấu với tăng độ lớn; đó là hai chuyện khác nhau.

Mỗi trục có một giới hạn

Với bát bậc hai, đổi sang cơ sở eigenvector để tách các hướng.



Từ hình đến công thức

$f(\theta) = \frac{1}{2}\theta^T H \theta$, H đối xứng xác định dương: $\theta_{\text{mới}} = (I - lrH)\theta$.

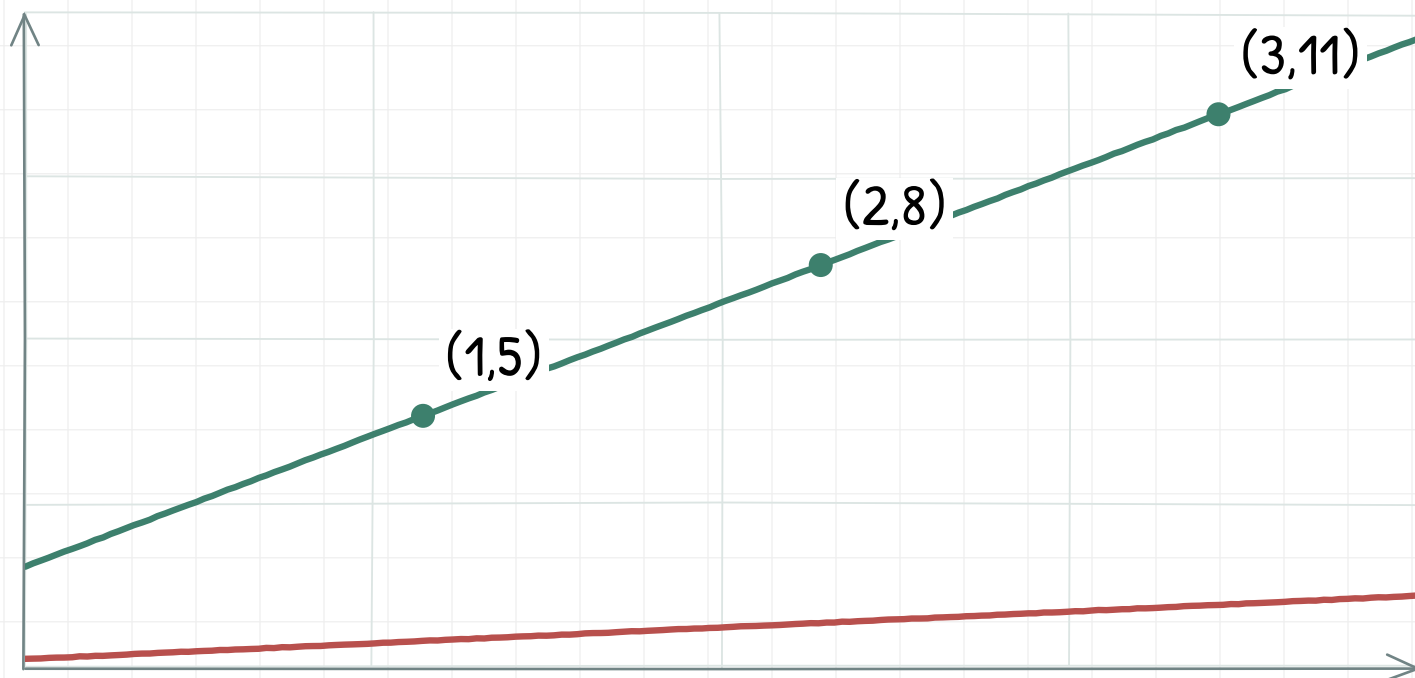
Mỗi thành phần eigenvector nhân $1 - lr \cdot \lambda_i$. Trục cong nhất hạn chế lr .

Ngưỡng này chính xác cho bát bậc hai; không phải bảo đảm toàn cục cho mọi hàm.

Độ cong lớn buộc bước nhỏ. Eigenvalue âm cần chẩn đoán khác.

Một bước học từ ba điểm

Bắt đầu $w=b=0$, khớp ba điểm $(1,5)$, $(2,8)$, $(3,11)$.



— đường đúng: $3x+2$ — sau một bước

Từ hình đến công thức

$L = \text{mean}((wx+b-y)^2)$. $\partial L / \partial w = 2 \cdot \text{mean}(rx)$; $\partial L / \partial b = 2 \cdot \text{mean}(r)$.

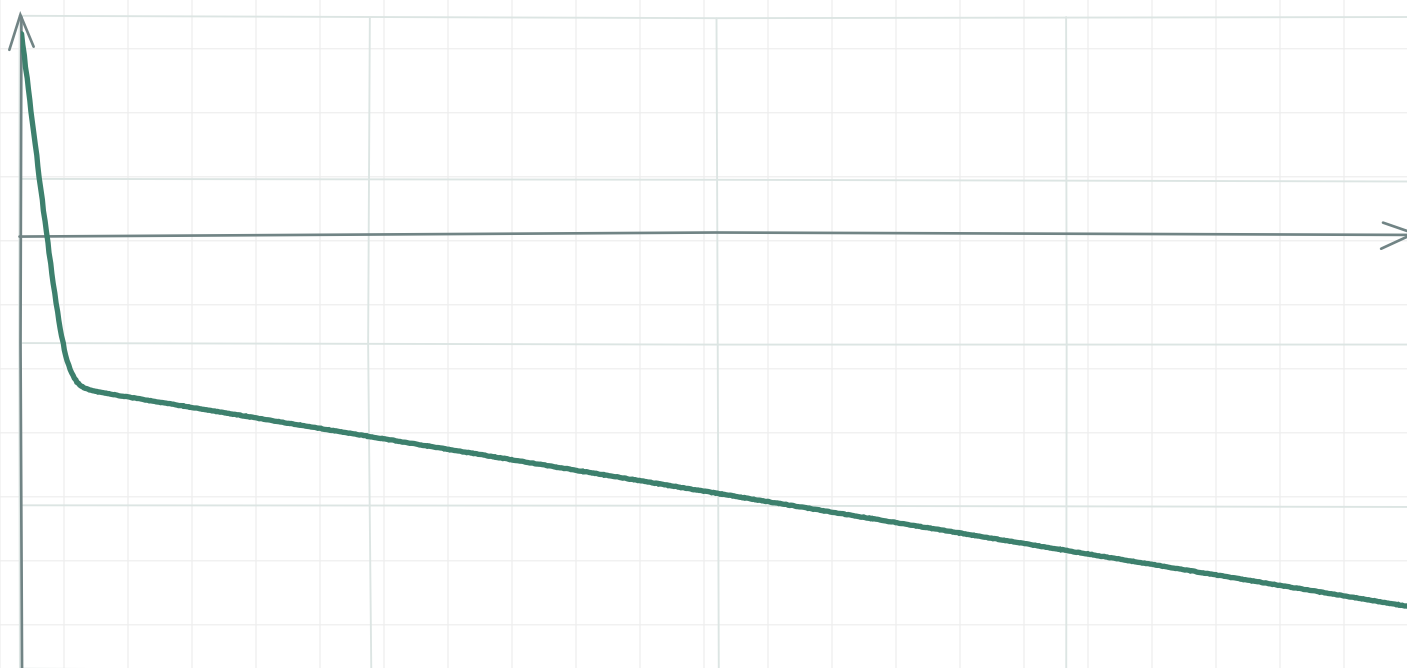
Ở bước đầu: $r = (-5, -8, -11)$, $L = 70$; $\text{gradient} = (-36, -16)$.

$lr = 0.01 \rightarrow (w, b) = (0.36, 0.16)$, $\text{loss mới} = 55.3408$.

Hai đạo hàm xuất phát từ chain rule, không cần autograd.

Đối chiếu với nghiệm tham chiếu

Cùng ba điểm, loss giảm khi $lr=0.01$; nghiệm đúng là $w=3, b=2$.



— ngang: bước k

— dọc: $\log_{10} \text{loss}$

Từ hình đến công thức

Viết A có các hàng $(1,1), (2,1), (3,1)$; $\theta=(w,b)$.

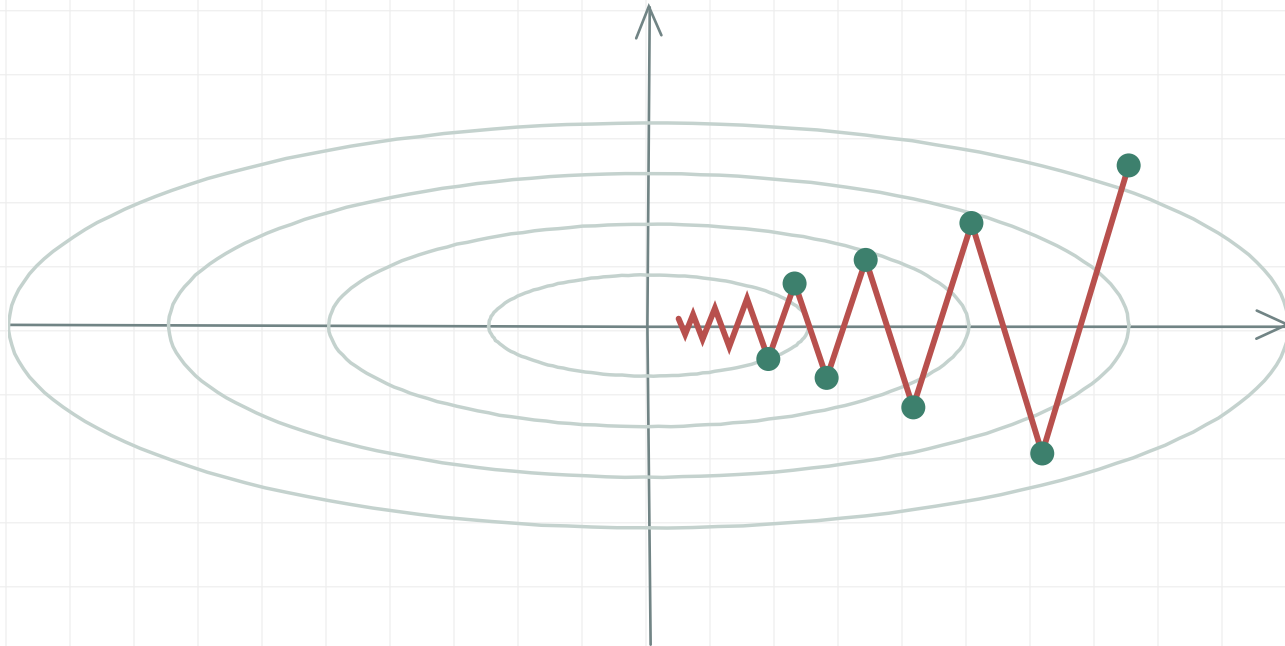
$\nabla L=(2/N)A^T(A\theta-y)$; $H=(2/N)A^TA$. Ngưỡng $lr \approx 0.180295$.

Capstone: lưu loss mỗi bước, so θ với $\text{lstsq}(A,y)$, rồi thử lr vượt ngưỡng.

Loss giảm là cần kiểm; còn phải kiểm gradient và nghiệm cuối.

Thung lũng hẹp tạo zigzag

$f = x^2 + 10y^2$: hướng y dốc gấp 10 lần về độ cong.



Từ hình đến công thức

$H = \text{diag}(2, 20)$, $\kappa = 10$; tỉ lệ hai bán trục đồng mức là $\sqrt{10}$, không phải 10.

$lr = 0.09$: x nhân 0.82; y nhân $-0.8 \rightarrow$ qua lại ngang thung lũng.

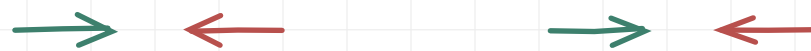
Tăng lr để tiến nhanh theo x có thể làm hướng y mất ổn định.

Một learning rate chung phải chịu giới hạn của hướng dốc nhất.

Momentum nhớ các bước trước

Tín hiệu cùng hướng tích lũy; dao động đổi dấu triệt tiêu một phần.

đổi dấu



triệt tiêu một phần

cùng dấu



tích lũy một hướng

Từ hình đến công thức

$$v_{\text{mới}} = \beta v + g; \theta_{\text{mới}} = \theta - lr \cdot v_{\text{mới}}, 0 \leq \beta < 1.$$

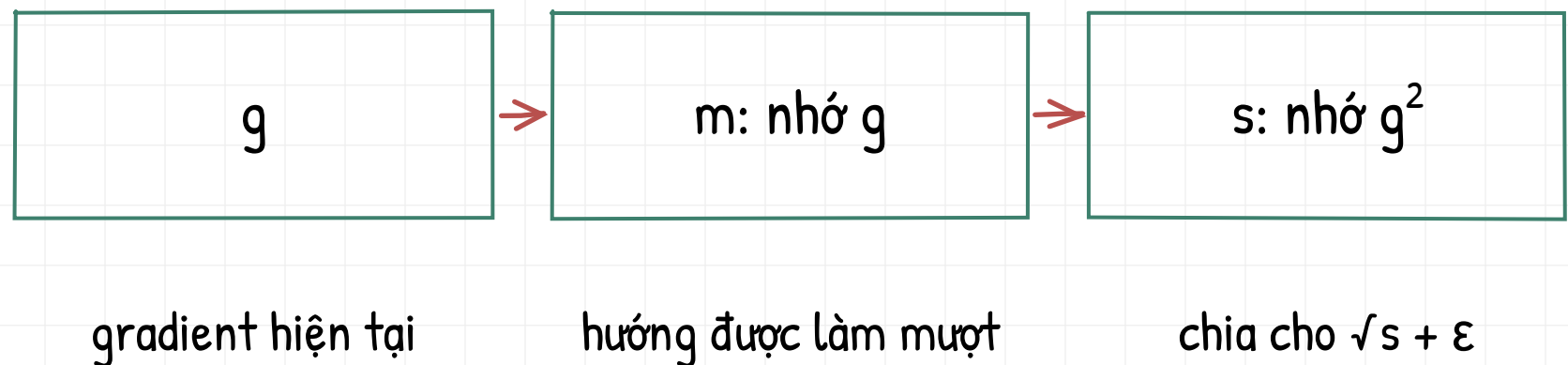
$v = g_t + \beta g_{t-1} + \beta^2 g_{t-2} + \dots$ là tổng có trọng số mũ.

Đây là phiên bản chưa nhân $(1-\beta)$. Khác quy ước sẽ đổi thang bước; vẫn phải chọn lr và β .

Momentum đổi cả đường đi; không bảo đảm mọi bước đều giảm loss.

RMSProp và Adam thêm gì?

Một bộ nhớ cho hướng; một bộ nhớ cho bình phương độ lớn gradient.



Từ hình đến công thức

RMSProp: $s \leftarrow \rho s + (1 - \rho)g^2$; $\theta \leftarrow \theta - lr \cdot g / (\sqrt{s} + \epsilon)$, tính từng phần tử.

Adam thêm $m \leftarrow \beta_1 m + (1 - \beta_1)g$; hiệu chỉnh $\hat{m} = m / (1 - \beta_1^t)$,
 $\hat{s} = s / (1 - \beta_2^t)$.

Bước Adam: $-lr \cdot \hat{m} / (\sqrt{\hat{s}} + \epsilon)$. s là moment bậc hai, không phải phương sai đã trừ trung bình.

Optimizer thích nghi vẫn cần lr và so với baseline đã tinh chỉnh.

Mini-batch: một hướng có nhiều

Mỗi điểm cho một gradient; lấy mẫu đều cho trung bình đúng trong kỳ vọng.

Điểm	g_i tại $(w,b)=(0,0)$
(1,5)	(-10, -10)
(2,8)	(-32, -16)
(3,11)	(-66, -22)
trung bình	(-36, -16)

Từ hình đến công thức

Lấy mẫu đều và lấy mean: $E[g_{\text{batch}} \mid \theta] = \nabla L(\theta)$.

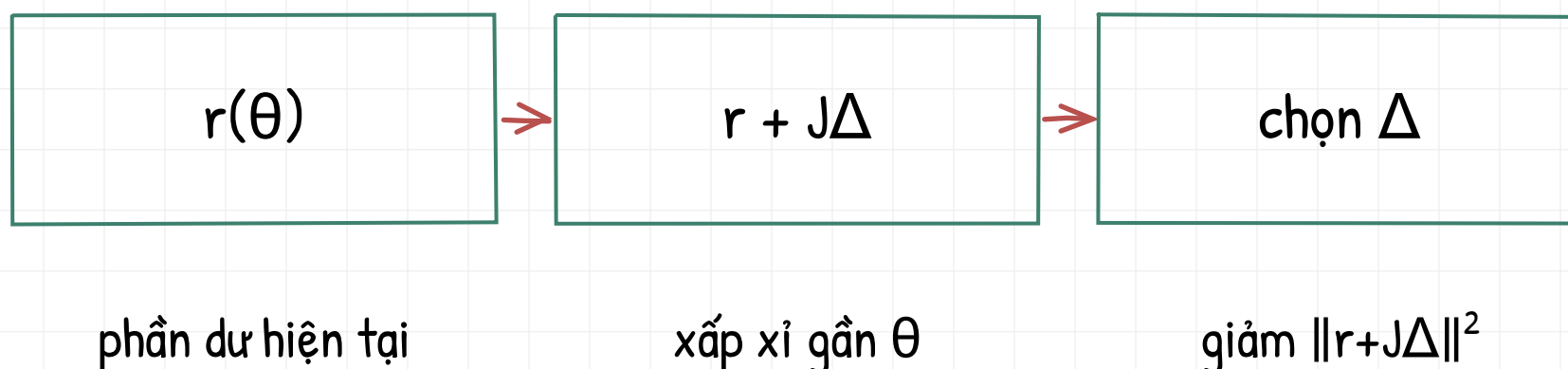
Không chệch chưa đủ bảo đảm hội tụ: còn điều kiện về hàm, phương sai và lịch lr.

Nhiều có thể giúp thoát một số vùng; quá lớn cũng cản việc ổn định gần nghiệm.

Batch nhỏ không tự động tốt hơn; xét cả ngân sách và chất lượng nghiệm.

Dùng độ cong để chọn bước

Tuyến tính hoá phần dư, rồi giải bài bình phương tối thiểu tại chỗ.



Từ hình đến công thức

Gauss-Newton: giải $J^T J \Delta = -J^T r$. Với $r = A\theta - y$, A đủ hạng cột: một bước ra nghiệm LS.

LM: $(J^T J + \lambda I) \Delta = -J^T r$, $\lambda > 0$. Tăng λ hạn chế bước; kiểm loss để nhận hoặc bỏ.

Newton tối ưu giải $H \Delta = -\nabla L$; Newton tìm nghiệm $r=0$ giải $J \Delta = -r$. Hai bài toán khác nhau.

Dùng solve / least squares, không tính nghịch đảo tường minh trong code.

Tự kiểm tra khi đi tiếp

Che đáp án, thử nói bằng lời

1. Với x^2 , lr nào hội tụ từ mọi điểm đầu?
2. $lr=0.9$ đổi phía mỗi bước: có phân kỳ?
3. Ngưỡng $2/\lambda_{\max}$ chính xác trong phạm vi nào?
4. Bước đầu hồi quy ba điểm cho (w,b) nào?
5. Gradient mini-batch không chệch đã đủ hội tụ?

Đáp nhanh

1. $0 < lr < 1$; $lr=0.5$ đến đáy ngay.
2. Không: hệ số -0.8 , độ lớn co dần.
3. Hàm bậc hai với Hessian đối xứng xác định dương.
4. $(0.36, 0.16)$, loss 55.3408, với $lr=0.01$.
5. Chưa: cần giả thiết về mục tiêu, nhiễu và bước học.