

Newton's method and second-order optimisation

Ten problems on Newton's method: the step as the minimiser of the quadratic model and the Newton decrement, the square root of 2 by hand, the proof of quadratic convergence with the error recurrence made exact, a one-dimensional minimisation with the digits doubling, affine invariance against gradient descent, why Newton finds critical points rather than minima and when its step points uphill, the Gauss–Newton approximation for least squares, Levenberg–Marquardt damping as an interpolation between Newton and gradient descent, the secant step as the first quasi-Newton update, and a two-cycle that shows convergence is only local, with worked solutions and the mistakes that iterate on f instead of f' , treat Gauss–Newton as exact, take the Newton direction as a descent direction, assume convergence from any start, or expect rescaling the loss to rescale the step.

BEFORE YOU START

Gradient descent uses the slope; Newton's method also uses the curvature, and the difference is the difference between a rate of convergence that depends on how stretched the loss is and one that does not. The method is simple to state, minimise the quadratic model and jump to its minimum, and everything interesting is in what the model gets wrong: far from the solution the model is poor, at a saddle its minimum is a maximum, and for a least-squares loss half of its Hessian is usually not worth computing. These ten problems derive the step and the decrement, compute $\sqrt{2}$ and a one-dimensional minimiser by hand and watch the correct digits double, prove quadratic convergence and make the error recurrence exact for a square root, show that the step is unchanged by a change of coordinates while gradient descent's is not, follow the method to a maximum, derive Gauss–Newton and the damping that turns it into Levenberg–Marquardt, take a secant step as the first quasi-Newton update, and end with a cubic on which undamped Newton cycles forever. The five mistakes at the end are the ones that produce an iteration that still runs: Newton on f when the target is $f' = 0$, Gauss–Newton called exact, an uphill Newton direction fed to a line search, local convergence read as global, and a step that was expected to shrink with the loss.

- f is a twice-differentiable function of $x \in \mathbb{R}^n$ (or of a scalar x), with gradient ∇f and Hessian $H = \nabla^2 f$; a minimiser is x^* and the error at step k is $e_k = x_k - x^*$. The quadratic model of f at x is $m_x(\delta) = f(x) + \nabla f(x)^\top \delta + \frac{1}{2} \delta^\top H(x) \delta$, the second-order Taylor polynomial (the Taylor-series page).
- Newton's method for minimisation steps to the minimiser of the model: $x_{k+1} = x_k + \delta_k$ with $H(x_k) \delta_k = -\nabla f(x_k)$. For a scalar function this is $x_{k+1} = x_k - f'(x_k)/f''(x_k)$. The Newton decrement is $\lambda(x)^2 = \nabla f(x)^\top H(x)^{-1} \nabla f(x)$.

- Newton's method for a root of $g(x) = 0$ is $x_{k+1} = x_k - g(x_k)/g'(x_k)$; minimisation is root-finding on $g = f'$, since $g' = f''$. ξ denotes a point between x_k and x^* supplied by Taylor's theorem.
- Gradient descent is $x_{k+1} = x_k - \eta \nabla f(x_k)$ with step size $\eta > 0$. A symmetric matrix A is positive definite (PD, $A \succ 0$) if $v^\top A v > 0$ for all $v \neq 0$ (the positive-definite page); its eigenvalues are λ_i with $\lambda_{\min}$ and $\lambda_{\max}$ the extremes. A direction d is a descent direction at x if $\nabla f(x)^\top d < 0$.
- For least squares, $r(\theta) \in \mathbb{R}^N$ is the residual vector with Jacobian $J = \partial r / \partial \theta \in \mathbb{R}^{N \times p}$ (numerator layout, the Jacobians page), and $f(\theta) = \frac{1}{2} \|r(\theta)\|^2$. $\nabla^2 r_n$ is the $p \times p$ Hessian of the n -th residual.

Builds on: [Partial derivatives, gradients and Hessians](#), [Taylor series and linear approximation](#)

Problems

1. Show that the minimiser of the quadratic model $m_x(\delta)$ when $H(x) \succ 0$ difficulty $\cdot$ is $\delta^* = -H(x)^{-1} \nabla f(x)$, and that the model predicts the decrease $f(x) - m_x(\delta^*) = \frac{1}{2} \nabla f(x)^\top H(x)^{-1} \nabla f(x) = \frac{1}{2} \lambda(x)^2$. Why is $H \succ 0$ needed?

2. Derive the root-finding iteration $x_{k+1} = x_k - g(x_k)/g'(x_k)$ as the zero difficulty $\cdot$ of the tangent line, and apply it to $g(x) = x^2 - 2$ from $x_0 = 1$. Simplify the iteration, compute x_1, x_2, x_3 exactly and the errors $x_k - \sqrt{2}$ to two significant figures.

3. Let x^* be a root of g with $g'(x^*) \neq 0$ and g'' continuous. Show that difficulty $\dots$
$$e_{k+1} = \frac{g''(\xi_k)}{2g'(x_k)} e_k^2$$
 for some ξ_k between x_k and x^* , so that convergence is quadratic once it starts: $|e_{k+1}| \leq C|e_k|^2$ near x^* . For $g(x) = x^2 - a$, show the exact recurrence $e_{k+1} = e_k^2/(2x_k)$ and check it against Problem 2.

4. Minimise $f(x) = e^x - 2x$ by Newton's method from $x_0 = 0$. Write the difficulty $\dots$ iteration, compute x_1, x_2, x_3 to four decimals and the errors $x_k - x^*$ for $k = 1, \dots, 4$, and verify that e_{k+1}/e_k^2 approaches the constant of Problem 3.

5. Let $y = Bx + c$ with B invertible, and define $h(y) = f(x)$, that is $h(y) = f(B^{-1}(y - c))$. Show that $\nabla h(y) = B^{-\top} \nabla f(x)$ and $\nabla^2 h(y) = B^{-\top} \nabla^2 f(x) B^{-1}$, and that the Newton step in y is B times the Newton step in x , so the iterates correspond: $y_k = Bx_k + c$ for all k . Show that gradient descent does not have this property unless B is orthogonal. difficulty $\dots$

6. Apply Newton's method to $f(x) = x^3 - 3x$ from $x_0 = -\frac{1}{2}$: compute x_1 and x_2 and say where the iteration converges and what kind of point that is. Then, for $f(x, y) = x^2 - y^2$ at $(1, 2)$, compute the Newton step and show that it is not a descent direction. difficulty $\dots$

7. For $f(\theta) = \frac{1}{2} \|r(\theta)\|^2$ with $r : \mathbb{R}^p \rightarrow \mathbb{R}^N$ and Jacobian J , show that $\nabla f = J^\top r$ and $\nabla^2 f = J^\top J + \sum_{n=1}^N r_n \nabla^2 r_n$. The Gauss–Newton step drops the second term: $\delta_{\text{GN}} = -(J^\top J)^{-1} J^\top r$. Show that δ_{GN} is the least-squares solution of the linearised problem $\min_\delta \|r + J\delta\|^2$, and that for a linear residual $r(\theta) = A\theta - b$ Gauss–Newton equals Newton and reaches the solution in one step. difficulty $\dots$

8. The damped step is $\delta_\mu = -(H + \mu I)^{-1} \nabla f$ for $\mu \geq 0$ with $H \succeq 0$. Show that (a) $\delta_\mu \rightarrow \delta_{\text{Newton}}$ as $\mu \rightarrow 0$ when $H \succ 0$, and $\delta_\mu \approx -\frac{1}{\mu} \nabla f$ as $\mu \rightarrow \infty$; (b) δ_μ is a descent direction for every $\mu > 0$; (c) $\|\delta_\mu\|$ is decreasing in μ ; (d) δ_μ minimises $m_x(\delta) + \frac{\mu}{2} \|\delta\|^2$. difficulty $\dots$

9. The secant method replaces $g'(x_k)$ in Newton's iteration by the slope $B_k = \frac{g(x_k) - g(x_{k-1})}{x_k - x_{k-1}}$. Starting from $x_0 = 1, x_1 = \frac{3}{2}$ for $g(x) = x^2 - 2$, compute x_2 and x_3 and their errors. Show that B_k is the unique scalar satisfying the secant equation $B_k(x_k - x_{k-1}) = g(x_k) - g(x_{k-1})$, and that for $g = f'$ this is the one-dimensional case of the quasi-Newton condition $B_k s_k = y_k$ with $s_k = x_k - x_{k-1}$ and $y_k = \nabla f(x_k) - \nabla f(x_{k-1})$. difficulty $\dots$

10. Apply Newton's method to $g(x) = x^3 - 2x + 2$ from $x_0 = 0$ and from $x_0 = -2$. Compute the first iterates in each case, say what happens, and explain the two outcomes in terms of Problem 3. difficulty $\dots$

Answers

1. $\delta^* = -H(x)^{-1}\nabla f(x)$ and $f(x) - m_x(\delta^*) = \frac{1}{2}\nabla f(x)^\top H(x)^{-1}\nabla f(x) = \frac{1}{2}\lambda(x)^2$
2. $x_{k+1} = \frac{1}{2}(x_k + 2/x_k)$; $x_1 = \frac{3}{2}$, $x_2 = \frac{17}{12}$, $x_3 = \frac{577}{408}$, with errors 8.6×10^{-2} , 2.5×10^{-3} , 2.1×10^{-6}
3. $e_{k+1} = \frac{g''(\xi_k)}{2g'(x_k)}e_k^2$, quadratic convergence with $C \approx \frac{|g''(x^*)|}{2|g'(x^*)|}$; for $x^2 - a$, exactly $e_{k+1} = \frac{e_k^2}{2x_k}$, which reproduces Problem 2's errors
4. $x_{k+1} = x_k - 1 + 2e^{-x_k}$; $x_1 = 1$, $x_2 \approx 0.7358$, $x_3 \approx 0.6940$; errors 0.31, 0.043, 9.0×10^{-4} , 4.0×10^{-7} , with $e_{k+1}/e_k^2 \rightarrow \frac{1}{2}$
5. $\nabla h = B^{-\top}\nabla f$, $\nabla^2 h = B^{-\top}\nabla^2 f B^{-1}$, $\delta_y = B\delta_x$, so $y_k = Bx_k + c$ throughout; gradient descent is invariant only under orthogonal B
6. From $x_0 = -\frac{1}{2}$: $x_1 = -1.25$, $x_2 = -1.025$, converging to the local maximum $x = -1$; for $x^2 - y^2$ at $(1, 2)$ the Newton step $(-1, -2)^\top$ has $\nabla f^\top \delta = 6 > 0$, an ascent direction, and lands on the saddle
7. $\nabla f = J^\top r$; $\nabla^2 f = J^\top J + \sum_n r_n \nabla^2 r_n$; $\delta_{\text{GN}} = -(J^\top J)^{-1}J^\top r$ solves $\min_\delta \|r + J\delta\|^2$; for $r = A\theta - b$ it is the Newton step and reaches $(A^\top A)^{-1}A^\top b$ in one step
8. $\delta_\mu \rightarrow \delta_{\text{Newton}}$ as $\mu \rightarrow 0$ and $\delta_\mu \approx -\frac{1}{\mu}\nabla f$ as $\mu \rightarrow \infty$; $\nabla f^\top \delta_\mu < 0$ for all $\mu > 0$; $\|\delta_\mu\|$ decreases in μ ; $\delta_\mu = \arg \min_\delta (m_x(\delta) + \frac{\mu}{2}\|\delta\|^2)$
9. $x_2 = 1.4$, $x_3 \approx 1.41379$ with errors -1.4×10^{-2} and -4.2×10^{-4} ; $B_k = \frac{g(x_k) - g(x_{k-1})}{x_k - x_{k-1}}$ is the unique solution of $B_k s_k = y_k$, the secant equation
10. From $x_0 = 0$ Newton cycles $0 \rightarrow 1 \rightarrow 0 \rightarrow \dots$; from $x_0 = -2$ it converges to $x^* \approx -1.76929$ with errors 0.23, 0.031, 6.6×10^{-4} , 3.1×10^{-7}

Worked solutions

Problem 1

Show that the minimiser of the quadratic model $m_x(\delta)$ when $H(x) \succ 0$ is $\delta^* = -H(x)^{-1}\nabla f(x)$, and that the model predicts the decrease $f(x) - m_x(\delta^*) = \frac{1}{2}\nabla f(x)^\top H(x)^{-1}\nabla f(x) = \frac{1}{2}\lambda(x)^2$. Why is $H \succ 0$ needed?

$\nabla_\delta m_x(\delta) = \nabla f(x) + H(x) \delta.$	$\nabla_\delta(g^\top \delta) = g$ and $\nabla_\delta(\frac{1}{2}\delta^\top H \delta) = H \delta$ for symmetric H (the matrix-calculus page).
$\nabla_\delta m_x = 0 \iff H \delta = -\nabla f(x) \iff \delta^* = -H^{-1}\nabla f(x).$	$H \succ 0$ is invertible (the positive-definite page's Problem 2).
$\nabla_\delta^2 m_x = H \succ 0$, so δ^* is the unique minimiser of m_x .	A quadratic with PD Hessian has one critical point and it is the global minimum (the positive-definite page's Problem 6).
$m_x(\delta^*) = f(x) - \nabla f^\top H^{-1}\nabla f + \frac{1}{2}\nabla f^\top H^{-1} H H^{-1}\nabla f = f(x) - \frac{1}{2}\nabla f^\top H^{-1}\nabla f.$	Substitute $\delta^* = -H^{-1}\nabla f$ into the model; $H^{-1} H H^{-1} = H^{-1}$.
$\delta^* = -H(x)^{-1}\nabla f(x)$ and $f(x) - m_x(\delta^*) = \frac{1}{2}\nabla f(x)^\top H(x)^{-1}\nabla f(x) = \frac{1}{2}\lambda(x)^2$	The Newton step is the jump to the bottom of the local bowl; the decrement $\lambda(x)^2$ is how deep the bowl says the bottom is, and $\frac{1}{2}\lambda^2$ is the usual stopping test, since it is affine-invariant (Problem 5) where $\ \nabla f\ $ is not. Without $H \succ 0$ step 2 still finds the critical point of the model, but step 3 fails: an indefinite H makes that critical point a saddle of the model and the step may go uphill (Problem 6), and a singular H leaves it undefined. The check compares δ^* against a direct minimisation of the model and the decrease against the model's values.

Problem 2

Derive the root-finding iteration $x_{k+1} = x_k - g(x_k)/g'(x_k)$ as the zero of the tangent line, and apply it to $g(x) = x^2 - 2$ from $x_0 = 1$. Simplify the iteration, compute x_1, x_2, x_3 exactly and the errors $x_k - \sqrt{2}$ to two significant figures.

The tangent line at x_k is $\ell(x) = g(x_k) + g'(x_k)(x - x_k)$.

The linear approximation of g at x_k (the Taylor-series page).

$$\ell(x) = 0 \iff x = x_k - g(x_k)/g'(x_k).$$

Solve the linear equation; $g'(x_k) \neq 0$ is needed.

$$\text{For } g(x) = x^2 - 2: x_{k+1} = x_k - \frac{x_k^2 - 2}{2x_k} = \frac{2x_k^2 - x_k^2 + 2}{2x_k} = \frac{1}{2} \left(x_k + \frac{2}{x_k} \right).$$

Put over the common denominator $2x_k$ and split the fraction.

$$x_1 = \frac{1}{2}(1 + 2) = \frac{3}{2}, x_2 = \frac{1}{2}\left(\frac{3}{2} + \frac{4}{3}\right) = \frac{17}{12},$$

$$x_3 = \frac{1}{2}\left(\frac{17}{12} + \frac{24}{17}\right) = \frac{577}{408}.$$

$$\frac{2}{3/2} = \frac{4}{3}, \frac{9+8}{6} = \frac{17}{6}; \frac{2}{17/12} = \frac{24}{17} \text{ and } \frac{17}{12} + \frac{24}{17} = \frac{289+288}{204} = \frac{577}{204}.$$

$$x_1 - \sqrt{2} \approx 8.6 \times 10^{-2}, x_2 - \sqrt{2} \approx 2.5 \times 10^{-3}, x_3 - \sqrt{2} \approx 2.1 \times 10^{-6}.$$

$\sqrt{2} = 1.41421356 \dots$ against 1.5, 1.416 and 1.41421569 $\dots$

$$x_{k+1} = \frac{1}{2} \left(x_k + 2/x_k \right); x_1 = \frac{3}{2}, x_2 = \frac{17}{12},$$

$$x_3 = \frac{577}{408}, \text{ with errors } 8.6 \times 10^{-2}, 2.5 \times 10^{-3}, 2.1 \times 10^{-6}$$

The number of correct digits roughly doubles each step: 1, 3, 6, and $x_4 = \frac{665857}{470832}$ is right to 12 digits. This is the Babylonian method, Newton's method before Newton, and the averaging form of step 3 explains it: x_k and $2/x_k$ sit on opposite sides of $\sqrt{2}$, and their mean is much closer than either. Problem 3 proves the doubling and makes the error recurrence exact for this g .

Problem 3

Let x^* be a root of g with $g'(x^*) \neq 0$ and g'' continuous. Show that $e_{k+1} = \frac{g''(\xi_k)}{2g'(x_k)} e_k^2$ for some ξ_k between x_k and x^* , so that convergence is quadratic once it starts: $|e_{k+1}| \leq C|e_k|^2$ near x^* . For $g(x) = x^2 - a$, show the exact recurrence $e_{k+1} = e_k^2/(2x_k)$ and check it against Problem 2.

$$0 = g(x^*) = g(x_k) + g'(x_k)(x^* - x_k) + \frac{1}{2}g''(\xi_k)(x^* - x_k)^2.$$

Taylor's theorem for g about x_k , evaluated at x^* , with the Lagrange remainder; $x^* - x_k = -e_k$.

$$g(x_k) = g'(x_k) e_k - \frac{1}{2}g''(\xi_k) e_k^2.$$

Rearrange step 1, using $(x^* - x_k) = -e_k$ and $(x^* - x_k)^2 = e_k^2$.

$$e_{k+1} = x_{k+1} - x^* = e_k - \frac{g(x_k)}{g'(x_k)} = e_k - e_k + \frac{g''(\xi_k)}{2g'(x_k)} e_k^2 = \frac{g''(\xi_k)}{2g'(x_k)} e_k^2.$$

Subtract x^* from the Newton update and substitute step 2 for $g(x_k)$; the linear terms cancel exactly.

Near x^* , $g'(x_k) \rightarrow g'(x^*) \neq 0$ and $g''(\xi_k) \rightarrow g''(x^*)$, so $|e_{k+1}| \leq C|e_k|^2$ with $C \approx \frac{|g''(x^*)|}{2|g'(x^*)|}$.

Continuity bounds the ratio once x_k is close; the error is squared each step, and if $C|e_k| < 1$ the sequence $C|e_k|$ is squared each step too, which is the doubling of correct digits.

For $g(x) = x^2 - a$: $x_{k+1} - \sqrt{a} = \frac{1}{2} \left(x_k + \frac{a}{x_k} \right) - \sqrt{a} = \frac{x_k^2 - 2\sqrt{a}x_k + a}{2x_k} = \frac{(x_k - \sqrt{a})^2}{2x_k}.$

Common denominator, then recognise the perfect square. Here $g'' = 2$ is constant, so step 3's ξ_k drops out and the recurrence is exact.

Problem 2: $e_2 = e_1^2/(2x_1) = (0.0858)^2/3 = 2.45 \times 10^{-3}$ and $e_3 = e_2^2/(2x_2) = (2.453 \times 10^{-3})^2/2.8\bar{3} = 2.12 \times 10^{-6}.$

Step 5 with $x_1 = \frac{3}{2}$ and $x_2 = \frac{17}{12}$; both match Problem 2 to the digits shown.

$e_{k+1} = \frac{g''(\xi_k)}{2g'(x_k)} e_k^2$, **quadratic**
convergence with $C \approx \frac{|g''(x^*)|}{2|g'(x^*)|}$; **for**
 $x^2 - a$, **exactly** $e_{k+1} = \frac{e_k^2}{2x_k}$, **which**
reproduces Problem 2's errors

The proof is one line of Taylor: Newton solves the linear part of g exactly, so what remains is the quadratic remainder. Three things in it are conditions, not decorations: $g'(x^*) \neq 0$ (at a double root the convergence drops to linear with ratio $\frac{1}{2}$), x_k close enough that $C|e_k| < 1$ (Problem 10 shows what happens otherwise), and g'' bounded. For minimisation, $g = f'$ and the condition is $f''(x^*) \neq 0$, a non-degenerate minimum.

Problem 4

Minimise $f(x) = e^x - 2x$ by Newton's method from $x_0 = 0$. Write the iteration, compute x_1, x_2, x_3 to four decimals and the errors $x_k - x^*$ for $k = 1, \dots, 4$, and verify that e_{k+1}/e_k^2 approaches the constant of Problem 3.

$f'(x) = e^x - 2$ and $f''(x) = e^x$; $f' = 0$ at $x^* = \log 2 \approx 0.6931$, where $f'' = 2 > 0$, a minimum.

Differentiate; $e^x = 2$ has the unique solution $\log 2$, and a positive second derivative there makes it a strict minimum.

$$x_{k+1} = x_k - \frac{e^{x_k} - 2}{e^{x_k}} = x_k - 1 + 2e^{-x_k}.$$

Newton on $g = f'$ with $g' = f''$; divide each term of the numerator by e^{x_k} .

$x_1 = 0 - 1 + 2 = 1$; $x_2 = 1 - 1 + 2e^{-1} \approx 0.7358$; $x_3 \approx 0.7358 - 1 + 2e^{-0.7358} \approx 0.6940$.

$e^{-1} \approx 0.36788$ and $e^{-0.7358} \approx 0.47914$.

$e_1 \approx 0.3069$, $e_2 \approx 0.04261$, $e_3 \approx 8.95 \times 10^{-4}$, $e_4 \approx 4.0 \times 10^{-7}$.

Subtract $\log 2 = 0.693147\dots$; $x_4 = x_3 - 1 + 2e^{-x_3} \approx 0.6931476$.

$e_2/e_1^2 \approx 0.452$, $e_3/e_2^2 \approx 0.493$, $e_4/e_3^2 \approx 0.500$, and Problem 3's constant is $\frac{g''(x^*)}{2g'(x^*)} = \frac{e^{x^*}}{2e^{x^*}} = \frac{1}{2}$.

$g = e^x - 2$ has $g' = g'' = e^x$; the ratios approach $\frac{1}{2}$ from below as $x_k \rightarrow x^*$.

$x_{k+1} = x_k - 1 + 2e^{-x_k}$; $x_1 = 1$, $x_2 \approx$
 0.7358 , $x_3 \approx 0.6940$; **errors** 0.31 , 0.043 ,
 9.0×10^{-4} , 4.0×10^{-7} , **with** $e_{k+1}/e_k^2 \rightarrow \frac{1}{2}$

Correct digits: 0, 1, 3, 6, then 13 at x_5 .
 Each Newton step is a gradient step with
 the step size $\eta = 1/f''(x_k)$ chosen by the
 curvature: 1 at x_0 , $1/e$ at x_1 , and $\frac{1}{2}$ near
 the solution, which is why no learning
 rate appears. A fixed $\eta = \frac{1}{2}$ would
 converge only linearly, with ratio $|1 -$
 $\frac{1}{2}f''(x_k)|$, and $\eta = 1$ would overshoot: f''
 varies by a factor of e over the first step.
 Mistake 1 applies this iteration to f
 instead of f' .

Problem 5

Let $y = Bx + c$ with B invertible, and define $h(y) = f(x)$, that is $h(y) = f(B^{-1}(y - c))$. Show
 that $\nabla h(y) = B^{-\top} \nabla f(x)$ and $\nabla^2 h(y) = B^{-\top} \nabla^2 f(x) B^{-1}$, and that the Newton step in y is B
 times the Newton step in x , so the iterates correspond: $y_k = Bx_k + c$ for all k . Show that
 gradient descent does not have this property unless B is orthogonal.

$x = B^{-1}(y - c)$ has Jacobian $\partial x / \partial y =$
 B^{-1} , so $\nabla h(y) = (B^{-1})^\top \nabla f(x) =$
 $B^{-\top} \nabla f(x)$.

Chain rule for a scalar through a linear
 map: the gradient picks up the
 transpose of the Jacobian (the
 Jacobians page).

$\nabla^2 h(y) = B^{-\top} \nabla^2 f(x) B^{-1}$.

Differentiate step 1 again in y : $\nabla f(x)$
 changes by $\nabla^2 f(x) \partial x / \partial y = \nabla^2 f(x) B^{-1}$,
 with the constant $B^{-\top}$ in front.

Newton in y : $\delta_y =$
 $-(B^{-\top} H B^{-1})^{-1} B^{-\top} \nabla f =$
 $-B H^{-1} B^\top B^{-\top} \nabla f = -B H^{-1} \nabla f = B \delta_x$.

$(B^{-\top} H B^{-1})^{-1} = B H^{-1} B^\top$, the inverse
 of a product in reverse order; $B^\top B^{-\top} =$
 I .

If $y_k = Bx_k + c$ then $y_{k+1} = y_k + \delta_y =$
 $Bx_k + c + B\delta_x = B(x_k + \delta_x) + c =$
 $Bx_{k+1} + c$.

Induction on k with step 3; it starts
 because $y_0 = Bx_0 + c$ by choice.

Gradient descent in y : $y_{k+1} = y_k - \eta B^{-\top} \nabla f$, while the image of the x -step is $B(x_k - \eta \nabla f) + c = y_k - \eta B \nabla f$. These agree for every ∇f only when $B^{-\top} = B$, that is $B^\top B = I$.

Compare the two updates term by term; equality for all gradient vectors forces the matrices to be equal.

$\nabla h = B^{-\top} \nabla f$, $\nabla^2 h = B^{-\top} \nabla^2 f B^{-1}$, $\delta_y = B \delta_x$, so $y_k = B x_k + c$ **throughout**; **gradient descent is invariant only under orthogonal B**

Newton's method does not care how the problem is parameterised: rescale a feature by 1000, change units, rotate the axes, and it takes the same path, because the Hessian's change undoes the gradient's. Gradient descent sees the scaling as a change in conditioning (the positive-definite page's Problem 9), which is why feature normalisation, batch norm and Adam's per-coordinate scaling matter for first-order methods and are irrelevant to Newton. The decrement $\lambda^2 = \nabla f^\top H^{-1} \nabla f$ is invariant for the same reason, so it is the right stopping criterion. Mistake 5 is the special case $B = cI$.

Problem 6

Apply Newton's method to $f(x) = x^3 - 3x$ from $x_0 = -\frac{1}{2}$: compute x_1 and x_2 and say where the iteration converges and what kind of point that is. Then, for $f(x, y) = x^2 - y^2$ at $(1, 2)$, compute the Newton step and show that it is not a descent direction.

$f'(x) = 3x^2 - 3$ and $f''(x) = 6x$; the critical points are $x = \pm 1$, with $f''(1) = 6 > 0$ (minimum) and $f''(-1) = -6 < 0$ (maximum).

Differentiate; $3x^2 - 3 = 0$ at $x = \pm 1$; the sign of f'' classifies them.

$$x_1 = -\frac{1}{2} - \frac{3 \cdot \frac{1}{4} - 3}{6 \cdot (-\frac{1}{2})} = -\frac{1}{2} - \frac{-2.25}{-3} = -\frac{1}{2} - 0.75 = -1.25.$$

Newton on f' ; both numerator and denominator are negative, so the step is to the left, away from the minimum at $+1$.

$x_2 = -1.25 - \frac{3 \cdot 1.5625 - 3}{-7.5} = -1.25 + \frac{1.6875}{7.5} = -1.25 + 0.225 = -1.025.$	The same iteration; the errors from -1 are 0.25 and then 0.025, already shrinking quadratically.
The iteration converges to $x = -1$, the local maximum.	Newton solves $f' = 0$ and converges to whichever critical point it starts near; with $f''(x_0) < 0$ the quadratic model is an upside-down parabola and its "minimiser" is a maximiser.
For $f(x, y) = x^2 - y^2$: $\nabla f = (2x, -2y)^\top = (2, -4)^\top$ and $H = \text{diag}(2, -2)$, so $\delta = -H^{-1}\nabla f = -(1, 2)^\top = (-1, -2)^\top$.	Partial derivatives; H is constant and indefinite, and $H^{-1} = \text{diag}(\frac{1}{2}, -\frac{1}{2})$.
$\nabla f^\top \delta = 2 \cdot (-1) + (-4) \cdot (-2) = -2 + 8 = 6 > 0$.	A positive directional derivative: f increases along δ, so δ is an ascent direction. The step lands on $(0, 0)$, the saddle, where $f' = 0$.
From $x_0 = -\frac{1}{2}$: $x_1 = -1.25$, $x_2 = -1.025$, converging to the local maximum $x = -1$; for $x^2 - y^2$ at $(1, 2)$ the Newton step $(-1, -2)^\top$ has $\nabla f^\top \delta = 6 > 0$, an ascent direction, and lands on the saddle	Newton's method finds critical points; which kind depends on the sign of the curvature it sees, and a loss with saddles, which every neural network has, hands it a step that may go up. The test is $\nabla f^\top \delta = -\nabla f^\top H^{-1}\nabla f$, which is negative for every nonzero gradient exactly when $H \succ 0$. The fixes all modify H: add μI with $\mu > -\lambda_{\min}$ (Problem 8), flip the negative eigenvalues (saddle-free Newton), or use Gauss–Newton, whose $J^\top J$ is PSD by construction (Problem 7). Mistake 3 hands the uphill direction to a line search.

Problem 7

For $f(\theta) = \frac{1}{2}\|r(\theta)\|^2$ with $r : \mathbb{R}^p \rightarrow \mathbb{R}^N$ and Jacobian J , show that $\nabla f = J^\top r$ and $\nabla^2 f = J^\top J + \sum_{n=1}^N r_n \nabla^2 r_n$. The Gauss–Newton step drops the second term: $\delta_{\text{GN}} = -(J^\top J)^{-1} J^\top r$. Show that

δ_{GN} is the least-squares solution of the linearised problem $\min_{\delta} \|r + J\delta\|^2$, and that for a linear residual $r(\theta) = A\theta - b$ Gauss–Newton equals Newton and reaches the solution in one step.

$$f = \frac{1}{2} \sum_n r_n^2, \text{ so } \frac{\partial f}{\partial \theta_j} = \sum_n r_n \frac{\partial r_n}{\partial \theta_j} = (J^\top r)_j.$$

Chain rule on each square; $J_{nj} = \partial r_n / \partial \theta_j$, so the sum over n is the j -th entry of $J^\top r$.

$$\frac{\partial^2 f}{\partial \theta_i \partial \theta_j} = \sum_n \frac{\partial r_n}{\partial \theta_i} \frac{\partial r_n}{\partial \theta_j} + \sum_n r_n \frac{\partial^2 r_n}{\partial \theta_i \partial \theta_j} = (J^\top J)_{ij} + \left(\sum_n r_n \nabla^2 r_n \right)_{ij}.$$

Product rule on $r_n \partial r_n / \partial \theta_j$: differentiate the first factor to get the Jacobian product, the second to get the residual times its Hessian.

$\|r + J\delta\|^2$ is a quadratic in δ with gradient $2J^\top(r + J\delta)$, zero when $J^\top J\delta = -J^\top r$.

The least-squares page's normal equations with design matrix J and target $-r$; $J^\top J \succ 0$ when J has full column rank.

$$\delta_{\text{GN}} = -(J^\top J)^{-1} J^\top r \text{ is that solution.}$$

Step 3 solved for δ ; it is the Newton step for the model in which r is replaced by its linearisation $r + J\delta$.

For $r = A\theta - b$: $J = A$ and $\nabla^2 r_n = 0$, so $\nabla^2 f = A^\top A = J^\top J$ and $\delta_{\text{GN}} = \delta_{\text{Newton}} = -(A^\top A)^{-1} A^\top (A\theta - b) = (A^\top A)^{-1} A^\top b - \theta$.

The dropped term is exactly zero for a linear residual; expand the product.

$\theta + \delta_{\text{GN}} = (A^\top A)^{-1} A^\top b$, the least-squares solution, from any θ .

The θ cancels; one step lands on the minimiser because the model is exact.

$\nabla f = J^\top r$; $\nabla^2 f = J^\top J + \sum_n r_n \nabla^2 r_n$;
 $\delta_{\text{GN}} = -(J^\top J)^{-1} J^\top r$ **solves** $\min_\delta \|r + J\delta\|^2$; **for** $r = A\theta - b$ **it is the Newton step and reaches** $(A^\top A)^{-1} A^\top b$ **in one step**

Gauss–Newton keeps the half of the Hessian that is cheap (first derivatives only) and always PSD, and drops the half that needs second derivatives of every residual and can be indefinite. The dropped term is small when the residuals are small at the solution (a good fit) or the model is nearly linear, and then Gauss–Newton converges almost as fast as Newton; with large residuals it converges only linearly or not at all (Mistake 2). In practice the step is computed by solving $J\delta \approx -r$ by QR or the SVD rather than forming $J^\top J$, whose condition number is the square of J 's. This is the iteration behind nonlinear least squares, bundle adjustment and the natural-gradient methods that use a Gauss–Newton matrix in place of the Hessian.

Problem 8

The damped step is $\delta_\mu = -(H + \mu I)^{-1} \nabla f$ for $\mu \geq 0$ with $H \succeq 0$. Show that (a) $\delta_\mu \rightarrow \delta_{\text{Newton}}$ as $\mu \rightarrow 0$ when $H \succ 0$, and $\delta_\mu \approx -\frac{1}{\mu} \nabla f$ as $\mu \rightarrow \infty$; (b) δ_μ is a descent direction for every $\mu > 0$; (c) $\|\delta_\mu\|$ is decreasing in μ ; (d) δ_μ minimises $m_x(\delta) + \frac{\mu}{2} \|\delta\|^2$.

(a) As $\mu \rightarrow 0$, $(H + \mu I)^{-1} \rightarrow H^{-1}$ when H is invertible; as $\mu \rightarrow \infty$, $(H + \mu I)^{-1} = \frac{1}{\mu} (I + H/\mu)^{-1} \rightarrow \frac{1}{\mu} I$.

Matrix inversion is continuous at an invertible matrix; factor μ out and note $H/\mu \rightarrow 0$.

(b) $\nabla f^\top \delta_\mu = -\nabla f^\top (H + \mu I)^{-1} \nabla f < 0$ for $\nabla f \neq 0$.

$H + \mu I \succ 0$ for $\mu > 0$ since its eigenvalues are $\lambda_i + \mu > 0$, and the inverse of a PD matrix is PD (the positive-definite page's Problem 2).

(c) With $H = Q\Lambda Q^\top$ and $g = Q^\top \nabla f$:
 $\|\delta_\mu\|^2 = \sum_i \frac{g_i^2}{(\lambda_i + \mu)^2}$, and each term decreases as μ grows.

$(H + \mu I)^{-1} = Q(\Lambda + \mu I)^{-1} Q^\top$ and Q preserves norms; the denominators increase with μ .

$$(d) \nabla_{\delta} (m_x(\delta) + \frac{\mu}{2} \|\delta\|^2) = \nabla f + H\delta + \mu\delta = 0 \iff (H + \mu I)\delta = -\nabla f.$$

Problem 1's gradient plus $\nabla_{\delta}(\frac{\mu}{2}\delta^{\top}\delta) = \mu\delta$; the Hessian $H + \mu I \succ 0$ makes the critical point the minimum.

$\delta_{\mu} \rightarrow \delta_{\text{Newton}}$ **as** $\mu \rightarrow 0$ **and** $\delta_{\mu} \approx -\frac{1}{\mu}\nabla f$
as $\mu \rightarrow \infty$; $\nabla f^{\top}\delta_{\mu} < 0$ **for all** $\mu > 0$;
 $\|\delta_{\mu}\|$ **decreases in** μ ; $\delta_{\mu} =$
 $\arg \min_{\delta} (m_x(\delta) + \frac{\mu}{2} \|\delta\|^2)$

Damping interpolates between Newton (trust the model, long step) and gradient descent with step $1/\mu$ (distrust it, short step along the gradient), and by (d) it is the model's minimiser under a penalty on the step length, the Lagrangian form of a trust region. Levenberg–Marquardt applies this to Gauss–Newton, $(J^{\top}J + \mu I)\delta = -J^{\top}r$, raising μ after a step that increases f and lowering it after one that decreases f ; because $J^{\top}J \succeq 0$, (b) guarantees every step is downhill. For an indefinite H the same algebra works once $\mu > -\lambda_{\min}$, which is the standard repair for Problem 6's uphill step. Note that $H + \mu I$ is not $(1 + \mu)H$: damping turns the step towards the gradient, it does not just shorten it, and (c) is the only sense in which it is a shrinkage.

Problem 9

The secant method replaces $g'(x_k)$ in Newton's iteration by the slope $B_k = \frac{g(x_k) - g(x_{k-1})}{x_k - x_{k-1}}$.

Starting from $x_0 = 1$, $x_1 = \frac{3}{2}$ for $g(x) = x^2 - 2$, compute x_2 and x_3 and their errors. Show that B_k is the unique scalar satisfying the secant equation $B_k(x_k - x_{k-1}) = g(x_k) - g(x_{k-1})$, and that for $g = f'$ this is the one-dimensional case of the quasi-Newton condition $B_k s_k = y_k$ with $s_k = x_k - x_{k-1}$ and $y_k = \nabla f(x_k) - \nabla f(x_{k-1})$.

$$g(x_0) = -1, g(x_1) = \frac{1}{4}, B_1 = \frac{\frac{1}{4} - (-1)}{\frac{3}{2} - 1} = \frac{5/4}{1/2} = \frac{5}{2}, \text{ so } x_2 = x_1 - \frac{g(x_1)}{B_1} = \frac{3}{2} - \frac{1/4}{5/2} = \frac{3}{2} - \frac{1}{10} = 1.4.$$

The slope of the chord through the two most recent points replaces the tangent slope $2x_1 = 3$.

$$g(x_2) = 1.96 - 2 = -0.04, B_2 = \frac{-0.04 - 0.25}{1.4 - 1.5} = \frac{-0.29}{-0.1} = 2.9, \text{ so } x_3 = 1.4 - \frac{-0.04}{2.9} \approx 1.41379.$$

The same step with the points x_1, x_2 .

Errors: $x_2 - \sqrt{2} \approx -1.4 \times 10^{-2}$, $x_3 - \sqrt{2} \approx -4.2 \times 10^{-4}$, and the next is $x_4 - \sqrt{2} \approx 2.1 \times 10^{-6}$.

Subtract 1.41421356...; the signs alternate because the chord's slope lags the tangent's.

$B_k(x_k - x_{k-1}) = g(x_k) - g(x_{k-1})$ has the unique solution $B_k = \frac{g(x_k) - g(x_{k-1})}{x_k - x_{k-1}}$ when $x_k \neq x_{k-1}$.

A linear equation in one unknown with nonzero coefficient.

With $g = f'$ this reads $B_k s_k = y_k$ where $s_k = x_k - x_{k-1}$ and $y_k = f'(x_k) - f'(x_{k-1})$: the curvature estimate is the change in gradient per unit change in position.

Rename; in one dimension the gradient is f' and the Hessian is the scalar f'' , which B_k approximates by a difference quotient.

$x_2 = 1.4$, $x_3 \approx 1.41379$ **with errors** -1.4×10^{-2} **and** -4.2×10^{-4} ; $B_k = \frac{g(x_k) - g(x_{k-1})}{x_k - x_{k-1}}$ **is the unique solution of** $B_k s_k = y_k$, **the secant equation**

The secant method needs no second derivative and converges superlinearly, with order $\phi = \frac{1+\sqrt{5}}{2} \approx 1.618$ rather than 2: the errors here, 8.6×10^{-2} , 1.4×10^{-2} , 4.2×10^{-4} , 2.1×10^{-6} , gain about 1.6 times as many digits per step as the last. In n dimensions the equation $B_k s_k = y_k$ has infinitely many solutions for the matrix B_k , and BFGS picks the one closest to the previous estimate that stays symmetric and positive definite; that choice is all that separates it from this problem, and it is why a quasi-Newton method converges superlinearly without ever forming a Hessian.

Problem 10

Apply Newton's method to $g(x) = x^3 - 2x + 2$ from $x_0 = 0$ and from $x_0 = -2$. Compute the first iterates in each case, say what happens, and explain the two outcomes in terms of Problem 3.

$g'(x) = 3x^2 - 2$. From $x_0 = 0$: $g(0) = 2$,
 $g'(0) = -2$, so $x_1 = 0 - \frac{2}{-2} = 1$.

The tangent at 0 has slope -2 and crosses zero at $x = 1$.

$g(1) = 1 - 2 + 2 = 1$, $g'(1) = 1$, so $x_2 =$
 $1 - \frac{1}{1} = 0$.

The tangent at 1 has slope 1 and crosses zero at $x = 0$.

$x_3 = 1$, $x_4 = 0$, and so on: the iteration cycles between 0 and 1 for ever and never converges.

Steps 1 and 2 repeat exactly; neither point is near a root, since $g > 0$ on $[0, 1]$ and in fact for every $x > -1.77$.

From $x_0 = -2$: $g(-2) = -8 + 4 + 2 = -2$,
 $g'(-2) = 10$, $x_1 = -2 + 0.2 = -1.8$;
 $g(-1.8) = -0.232$, $g'(-1.8) = 7.72$, $x_2 \approx$
 $-1.8 + 0.03005 = -1.76995$; then $x_3 \approx$
 -1.7692927 , $x_4 \approx -1.7692924$.

Two full Newton steps and two more carried to the digits shown; the real root is $x^* \approx -1.7692924$.

Errors from -2 : 0.23 , 3.1×10^{-2} , $6.6 \times$
 10^{-4} , 3.1×10^{-7} : quadratic, with e_{k+1}/e_k^2
settling near $\frac{g''(x^*)}{2g'(x^*)} = \frac{6x^*}{2(3x^{*2} - 2)} =$
 $\frac{-10.6}{2 \cdot 7.39} \approx -0.72$.

Problem 3's constant with $g'' = 6x$ and $g'(x^*) \approx 7.39$; the error signs alternate because the constant is negative.

From $x_0 = 0$ Newton cycles $0 \rightarrow 1 \rightarrow 0 \rightarrow \dots$; from $x_0 = -2$ it converges to $x^* \approx -1.76929$ with errors 0.23, 0.031, 6.6×10^{-4} , 3.1×10^{-7}

Problem 3 promises quadratic convergence only once $C|e_k| < 1$ with C evaluated near the root; from $x_0 = 0$ the error is 1.77, g' changes sign between x_0 and x^* (at $x = -\sqrt{2/3}$), and the tangent line is a useless model of g over that distance. The cubic has one real root and two complex ones, and the pair $\{0, 1\}$ is an attracting cycle of the Newton map: a whole interval of starts is pulled into it. For minimisation the same pathology appears when f'' changes sign between the start and the minimiser. The practical cures are to damp the step (Problem 8) or to accept the full step only when it reduces $|g|$ or f by enough, a backtracking line search, and then to let the full step take over near the solution, where Problem 3 applies; Mistake 4 reads Problem 3 as a global guarantee.

Where this goes wrong

1. Newton's iteration applied to f instead of f'

The root-finding formula is the one everyone remembers, and the function to hand is the loss.

For a root of g : $x_{k+1} = x_k - g(x_k)/g'(x_k)$

Right so far: Problem 2.

“To minimise f , run Newton on f .”

The habit that causes the mistake: Problem 4 minimised by solving $f' = 0$, so the function fed to the root-finder is f' , and the division is by f'' .

$$x_{k+1} = x_k - f(x_k)/f'(x_k) \text{ with } f(x) = e^x - 2x \text{ ✗}$$

This looks for a root of f , and f has none: its minimum value is $f(\log 2) = 2 - 2 \log 2 \approx 0.61 > 0$. From $x_0 = 0$: $x_1 = 0 - 1/(-1) = 1$, $x_2 = 1 - (e - 2)/(e - 2) = 0$, a two-cycle like Problem 10's, and from other starts the iterates wander. The correct iteration is Problem 4's $x_{k+1} = x_k - (e^{x_k} - 2)/e^{x_k}$. The same confusion in n dimensions divides by the gradient instead of the Hessian, which is not even an operation; the step that exists is $-H^{-1}\nabla f$, and its one-dimensional shadow is $-f'/f''$.

2. Gauss–Newton taken to be the exact Newton step

For a linear residual the two coincide, and the equality is carried over to the nonlinear case.

$$\nabla f = J^\top r \text{ and, for } r = A\theta - b, \nabla^2 f = J^\top J$$

Right so far: Problem 7, steps 1 and 5.

“The Hessian of a sum of squares is $J^\top J$.
”

The shortcut that causes the mistake: step 2 of Problem 7 has a second term, $\sum_n r_n \nabla^2 r_n$, that vanishes only when the residuals are linear in θ or zero.

$\nabla^2 f = J^\top J$ for $r_n(\theta) = \theta_1 e^{\theta_2 t_n} - y_n$, so
Gauss–Newton converges quadratically ✗

For this residual $\partial^2 r_n / \partial \theta_1 \partial \theta_2 = t_n e^{\theta_2 t_n}$ and $\partial^2 r_n / \partial \theta_2^2 = \theta_1 t_n^2 e^{\theta_2 t_n}$, so the true Hessian is $J^\top J + \sum_n r_n \nabla^2 r_n$, and the check finds the second term far from negligible away from a good fit. Gauss–Newton is Newton on the linearised residual, not on f ; its convergence is quadratic only if the residuals vanish at the solution, linear with a rate set by the size of the dropped term otherwise, and it can diverge for large-residual problems, which is the case Levenberg–Marquardt's damping (Problem 8) exists to rescue. The thing that is exactly true is weaker and more useful: $J^\top J$ is always PSD, so the Gauss–Newton direction is always downhill.

3. Feeding the Newton direction to a line search without checking its sign

A line search makes any step safe, provided it is a direction along which the function decreases.

$\delta = -H^{-1} \nabla f$ and a line search picks the step length α along δ

Right so far: the standard damped Newton template.

“The line search will handle any problems with the step.”

The habit that causes the mistake: a line search shortens a step; it cannot reverse one, and it assumes $\nabla f^\top \delta < 0$.

At $(1, 2)$ for $f = x^2 - y^2$, search along
 $\delta = (-1, -2)^\top$ ✗

$\nabla f^\top \delta = 6 > 0$ (Problem 6): f increases for every small $\alpha > 0$, so a backtracking search halves α until it underflows and the iteration stalls, or an exact search returns $\alpha = 0$. The direction is uphill because $H = \text{diag}(2, -2)$ is indefinite, and no step length fixes a wrong direction. The repair is to the matrix, not the search: use $H + \mu I$ with $\mu > 2$ (Problem 8), which makes $\nabla f^\top \delta_\mu < 0$, or replace the negative eigenvalue by its absolute value, which gives $\delta = (-1, 2)^\top$ with $\nabla f^\top \delta = -10$. Checking the sign of $\nabla f^\top \delta$ before searching costs one dot product.

4. Reading quadratic convergence as convergence from any start

Newton's method is the fast one, and fast is heard as reliable.

$|e_{k+1}| \leq C|e_k|^2$ near a root with $g'(x^*) \neq 0$ Right so far: Problem 3.

“Newton's method converges quadratically, so it converges.”

The habit that causes the mistake: Problem 3's bound only shrinks the error when $C|e_k| < 1$, which is a statement about where x_k is, not about the method.

For $g(x) = x^3 - 2x + 2$, Newton from $x_0 = 0$ converges quadratically to the root $\times$

It cycles $0 \rightarrow 1 \rightarrow 0$ for ever (Problem 10), and from $x_0 = 0.5$ it wanders before landing in the same cycle; only starts near enough to $x^* \approx -1.769$, where g' keeps its sign, get the promised doubling of digits. In minimisation the analogue is a start on the far side of an inflection point, where f'' has the wrong sign and the step goes to a maximum (Problem 6). Gradient descent with a small enough step decreases f from any start on a smooth bounded-below function; Newton trades that guarantee for speed, and the globalisation strategies, damping, line searches and trust regions, are what buy the guarantee back. In practice a first-order method brings the iterate into the basin and the second-order method finishes.

5. Expecting the Newton step to scale with the loss

Dividing a summed loss by the number of examples makes the gradient N times smaller, and the learning rate has to be retuned; the same is expected of the Newton step.

For $\tilde{f} = cf$ with $c > 0$: $\nabla \tilde{f} = c\nabla f$ and $\nabla^2 \tilde{f} = c\nabla^2 f$

Right so far: a constant comes through both derivatives.

“Scaling the loss by c scales every step by c .”

The habit that causes the mistake: true for gradient descent, where $-\eta \nabla \tilde{f} = -c\eta \nabla f$; Problem 5 showed Newton ignores reparameterisations, and this is the simplest one.

The Newton step for $\tilde{f}$ is c times the Newton step for f ✕

$-(c\nabla^2 f)^{-1}(c\nabla f) = -\frac{1}{c}c(\nabla^2 f)^{-1}\nabla f =$
 $-(\nabla^2 f)^{-1}\nabla f$: the c cancels, and the step is identical, which is the regression page's observation that Newton's step for logistic regression is the same for the mean loss and the summed loss. Nothing needs retuning when the loss is rescaled, and no learning rate exists to retune, which is one of the method's attractions. The step that does scale is the damped one of Problem 8, because μI is not multiplied by c : $-(cH + \mu I)^{-1}c\nabla f = -(H + \frac{\mu}{c}I)^{-1}\nabla f$, so rescaling the loss is equivalent to rescaling the damping, and Levenberg–Marquardt's μ has to be chosen relative to the size of $J^\top J$.