

Jacobians and the chain rule

Ten problems on Jacobians of elementwise and affine maps, the order of the chain rule, vector-Jacobian products, and gradients through one and two layers, with worked solutions and the shape mistakes that hide inside them.

BEFORE YOU START

A backward pass is the chain rule applied to vectors and matrices. Each step multiplies by a Jacobian or its transpose, and the wrong answers come from four places: factors in the wrong order, a derivative evaluated at the wrong point, a Jacobian replaced by the vector that describes its action, and factors transposed until the shapes fit. These ten problems work through all four.

- The conventions are those of the previous page: vectors are columns, Jacobians are in numerator layout ($\partial y / \partial x$ is $m \times n$ for $y \in \mathbb{R}^m$, $x \in \mathbb{R}^n$), the gradient of a scalar is the transpose of its row derivative, a matrix gradient has the shape of the matrix, and $\| \cdot \|$ is the Euclidean norm.
- σ is the logistic sigmoid, $\sigma(t) = 1/(1 + e^{-t})$, applied elementwise to vectors. Its derivative is $\sigma'(t) = \sigma(t)(1 - \sigma(t))$, also applied elementwise, so $\sigma'(z) = \sigma(z) \odot (1 - \sigma(z))$ for a vector z .
- $\odot$ is the elementwise product, $(u \odot v)_i = u_i v_i$. $\text{diag}(v)$ is the diagonal matrix with v on the diagonal, and $\text{diag}(v) u = v \odot u$.
- One layer is $z = Wx + b$, $h = \sigma(z)$, with $W \in \mathbb{R}^{m \times n}$, $x \in \mathbb{R}^n$ and $b, z, h \in \mathbb{R}^m$. z is the pre-activation and h the output.
- The chain rule in numerator layout: for $y = g(u)$ and $u = f(x)$, $\partial y / \partial x = (\partial y / \partial u)(\partial u / \partial x)$. The outer Jacobian goes on the left. With $y \in \mathbb{R}^p$, $u \in \mathbb{R}^m$, $x \in \mathbb{R}^n$ the shapes are $(p \times m)(m \times n) = p \times n$.
- Rule of thumb: for a scalar loss you never need the full Jacobian of an elementwise layer, only its action on a vector, and that action is an elementwise product.

Builds on: [Matrix calculus conventions](#)

Problems

1. Let $h = \sigma(z)$ elementwise, $z \in \mathbb{R}^n$. Compute $\partial h / \partial z$ and state its shape. difficulty ·

2. Let $z = Wx + b$ with $W \in \mathbb{R}^{m \times n}$. Compute $\partial z / \partial x$ and $\partial z / \partial b$.

difficulty ·

3. Let $h = \sigma(Wx + b)$. Compute $\partial h / \partial x$ using the chain rule, with shapes at every step.

difficulty ··

4. Let $y = B(Ax)$ with $A \in \mathbb{R}^{m \times n}$, $B \in \mathbb{R}^{p \times m}$. Write $\partial y / \partial x$ as a product of Jacobians and check the shapes. Which order is forced?

difficulty ··

5. Let $L = \frac{1}{2} \|h - y\|^2$ with $h = \sigma(Wx + b)$ and y constant. Compute $\nabla_x L$, and write it without any diag matrix.

difficulty ··

6. For a scalar loss L and $h = \sigma(Wx + b)$, show that $\nabla_x L = (\partial h / \partial x)^\top \nabla_h L$ and simplify it to a form that never forms the $m \times m$ Jacobian of σ . Why does this matter when $m = 4096$?

difficulty ··

7. Two layers: $h_1 = \sigma(W_1 x)$, $h_2 = W_2 h_1$, $L = \frac{1}{2} \|h_2 - y\|^2$. Compute $\nabla_x L$.

difficulty ···

8. Let $L = \frac{1}{2} \|Wx - y\|^2$ with $W \in \mathbb{R}^{m \times n}$. Compute $\nabla_W L$ and state its shape.

difficulty ···

9. Let $u = x / \|x\|_2$ for $x \neq 0$. Compute $\partial u / \partial x$.

difficulty ··

10. Compute $\nabla_x \|\sigma(Wx)\|_2^2$.

difficulty ···

Answers

1. $\partial h / \partial z = \text{diag}(\sigma'(z)), n \times n$
2. $\partial z / \partial x = W (m \times n); \partial z / \partial b = I_m$
3. $\partial h / \partial x = \text{diag}(\sigma'(z)) W, m \times n$
4. $\partial y / \partial x = BA, (p \times m)(m \times n) = p \times n$; AB is not defined unless $p = n$, and is wrong even then
5. $\nabla_x L = W^\top (\sigma'(z) \odot (h - y))$
6. $\nabla_x L = W^\top (\sigma'(z) \odot \nabla_h L)$; the diag never appears, so the elementwise layer costs $O(m)$ storage instead of the $O(m^2)$ of its Jacobian
7. $\nabla_x L = W_1^\top (\sigma'(z_1) \odot W_2^\top (h_2 - y))$
8. $\nabla_W L = (Wx - y) x^\top, m \times n$
9. $\partial u / \partial x = \frac{1}{\|x\|} (I - \hat{x} \hat{x}^\top), \hat{x} = x / \|x\|$
10. $\nabla_x \|\sigma(Wx)\|^2 = 2W^\top (\sigma'(z) \odot \sigma(z)), z = Wx$

Worked solutions

Problem 1

Let $h = \sigma(z)$ elementwise, $z \in \mathbb{R}^n$. Compute $\partial h / \partial z$ and state its shape.

$h_i = \sigma(z_i)$ for $i = 1, \dots, n$.	Elementwise means output i is computed from input i alone.
$\partial h_i / \partial z_j = \sigma'(z_i)$ if $i = j$, and 0 if $i \neq j$.	h_i contains no z_j with $j \neq i$, so those partials vanish; the remaining one is the ordinary derivative of σ at z_i .
The Jacobian has $\sigma'(z_1), \dots, \sigma'(z_n)$ on the diagonal and zeros elsewhere.	Entry (i, j) of $\partial h / \partial z$ is $\partial h_i / \partial z_j$, so step 2 fills it entry by entry.
$\partial h / \partial z = \text{diag}(\sigma'(z)), n \times n$	One row per h_i and one column per z_j , and both have n entries.

Problem 2

Let $z = Wx + b$ with $W \in \mathbb{R}^{m \times n}$. Compute $\partial z / \partial x$ and $\partial z / \partial b$.

$z_i = \sum_j W_{ij} x_j + b_i$.	Writing entry i out shows which terms contain each x_j and each b_k .
$\partial z_i / \partial x_j = W_{ij}$.	Only term j of the sum contains x_j , with coefficient W_{ij} ; b_i does not depend on x .
$\partial z_i / \partial b_k = 1$ if $i = k$, and 0 otherwise.	b_k appears only in z_k , with coefficient 1, and the sum does not depend on b .
$\partial z / \partial x = W$ ($m \times n$); $\partial z / \partial b = I_m$	Step 2 fills the $m \times n$ Jacobian with the entries of W ; step 3 fills the $m \times m$ Jacobian with ones on the diagonal. The previous page's $\partial(Ax) / \partial x = A$ is the case $b = 0$.

Problem 3

Let $h = \sigma(Wx + b)$. Compute $\partial h / \partial x$ using the chain rule, with shapes at every step.

Let $z = Wx + b$, so $h = \sigma(z)$.	Naming the pre-activation splits h into two maps whose Jacobians are Problems 1 and 2.
$\partial h / \partial x = (\partial h / \partial z)(\partial z / \partial x)$.	Chain rule in numerator layout: the Jacobian of the outer map, σ , goes on the left.
$\partial h / \partial z = \text{diag}(\sigma'(z))$, $m \times m$; $\partial z / \partial x = W$, $m \times n$.	Problem 1 with m in place of n , since z has m entries, and Problem 2.
$(m \times m)(m \times n) = m \times n$.	The inner dimensions agree, and the result has one row per h_i and one column per x_j , the shape $\partial h / \partial x$ must have.
$\partial h / \partial x = \text{diag}(\sigma'(z)) W$, $m \times n$	Row i of this product is row i of W scaled by $\sigma'(z_i)$: h_i depends on x only through z_i , and z_i depends on x through row i of W .

Problem 4

Let $y = B(Ax)$ with $A \in \mathbb{R}^{m \times n}$, $B \in \mathbb{R}^{p \times m}$. Write $\partial y / \partial x$ as a product of Jacobians and check the shapes. Which order is forced?

Let $u = Ax$, so $y = Bu$.	Naming the intermediate vector makes y a composition of two linear maps, which is the form the chain rule takes.
$\partial u / \partial x = A$, $m \times n$; $\partial y / \partial u = B$, $p \times m$.	The Jacobian of a linear map is its matrix (Problem 2 with $b = 0$).
$\partial y / \partial x = (\partial y / \partial u)(\partial u / \partial x) = BA$.	The outer map is B , so its Jacobian goes on the left.

$(p \times m)(m \times n) = p \times n$. The other order, $(m \times n)(p \times m)$, needs $n = p$.	Only BA has inner dimensions that agree for every m, n, p , and its result is $p \times n$, one row per y_i and one column per x_j . The shapes force this order.
$\partial y / \partial x = BA$, $(p \times m)(m \times n) = p \times n$; AB is not defined unless $p = n$, and is wrong even then	Directly, $y = (BA)x$, whose Jacobian is BA . When $p = n$, AB is $m \times m$, not $p \times n$; when all three sizes are equal it has the right shape, but $AB \neq BA$ in general.

Problem 5

Let $L = \frac{1}{2} \|h - y\|^2$ with $h = \sigma(Wx + b)$ and y constant. Compute $\nabla_x L$, and write it without any diag matrix.

Let $z = Wx + b$, so $h = \sigma(z)$ and $L = \frac{1}{2}(h - y)^\top (h - y)$.	The squared norm is the inner product of $h - y$ with itself, which the product rule can differentiate.
$dL = (h - y)^\top dh$, so $\partial L / \partial h = (h - y)^\top$, $1 \times m$.	Product rule gives two equal scalar terms, and the $\frac{1}{2}$ cancels the 2; y is constant.
$\partial h / \partial x = \text{diag}(\sigma'(z)) W$, $m \times n$.	Problem 3.
$\partial L / \partial x = (h - y)^\top \text{diag}(\sigma'(z)) W$, with shapes $(1 \times m)(m \times m)(m \times n) = 1 \times n$.	Chain rule with the outer Jacobian, that of L with respect to h , on the left.
$\nabla_x L = W^\top \text{diag}(\sigma'(z)) (h - y)$, with shapes $(n \times m)(m \times m)(m \times 1) = n \times 1$.	The gradient is the transpose of the row; transposing a product reverses it, and a diagonal matrix is its own transpose.
$\text{diag}(\sigma'(z)) (h - y) = \sigma'(z) \odot (h - y)$.	A diagonal matrix times a vector scales entry i of the vector by diagonal entry i , which is exactly the elementwise product.
$\nabla_x L = W^\top (\sigma'(z) \odot (h - y))$	Substitute step 6 into step 5. Shape check: $(n \times m)(m \times 1) = n \times 1$, the shape of x .

Problem 6

For a scalar loss L and $h = \sigma(Wx + b)$, show that $\nabla_x L = (\partial h / \partial x)^\top \nabla_h L$ and simplify it to a form that never forms the $m \times m$ Jacobian of σ . Why does this matter when $m = 4096$?

$\partial L / \partial x = (\partial L / \partial h)(\partial h / \partial x)$, with shapes
 $(1 \times m)(m \times n) = 1 \times n$.

L depends on x only through h , so the chain rule applies with L as the outer map.

$\nabla_x L = (\partial h / \partial x)^\top (\partial L / \partial h)^\top =$
 $(\partial h / \partial x)^\top \nabla_h L$.

Transpose both sides: the gradient is the transposed row, a transposed product reverses its order, and $(\partial L / \partial h)^\top$ is by definition $\nabla_h L$.

Let $z = Wx + b$. Then $(\partial h / \partial x)^\top =$
 $(\text{diag}(\sigma'(z)) W)^\top = W^\top \text{diag}(\sigma'(z))$.

Problem 3, transposed; a diagonal matrix is its own transpose.

$\nabla_x L = W^\top (\text{diag}(\sigma'(z)) \nabla_h L)$.

Multiplying right to left means the $m \times m$ matrix only ever acts on the vector $\nabla_h L$, so the matrix $\text{diag}(\sigma'(z)) W$ is never needed.

$\text{diag}(\sigma'(z)) \nabla_h L = \sigma'(z) \odot \nabla_h L$.

A diagonal matrix acting on a vector is an elementwise product: m multiplications, and only the m numbers $\sigma'(z)$ are stored.

$\nabla_x L = W^\top (\sigma'(z) \odot \nabla_h L)$; **the diag never appears, so the elementwise layer costs $O(m)$ storage instead of the $O(m^2)$ of its Jacobian**

At $m = 4096$ the Jacobian of σ has $4096^2 \approx 16.8$ million entries, about 67 MB in 32-bit floats, per example and per layer, and all but 4096 of them are zero. The elementwise form stores 4096 numbers, and the remaining work is the product with $W^\top$, which the backward pass pays anyway.

Problem 7

Two layers: $h_1 = \sigma(W_1 x)$, $h_2 = W_2 h_1$, $L = \frac{1}{2} \|h_2 - y\|^2$. Compute $\nabla_x L$.

Let $z_1 = W_1 x$, so $h_1 = \sigma(z_1)$. Take $x \in \mathbb{R}^n$, $W_1 \in \mathbb{R}^{k \times n}$, $W_2 \in \mathbb{R}^{p \times k}$, so $z_1, h_1 \in \mathbb{R}^k$ and $h_2, y \in \mathbb{R}^p$.	σ' has to be evaluated at the pre-activation, so it needs a name; the sizes let every product below be checked.
$\nabla_{h_2} L = h_2 - y$, $p \times 1$.	The previous page's $\nabla_x \ Ax - b\ ^2 = 2A^\top(Ax - b)$ with $A = I$, halved.
$\nabla_{h_1} L = W_2^\top (h_2 - y)$, with shapes $(k \times p)(p \times 1) = k \times 1$.	Problem 6's rule: the gradient with respect to a map's input is its transposed Jacobian times the gradient with respect to its output, and $\partial h_2 / \partial h_1 = W_2$ (Problem 2).
$\nabla_{z_1} L = \sigma'(z_1) \odot \nabla_{h_1} L$, $k \times 1$.	The same rule through σ : the transposed Jacobian is $\text{diag}(\sigma'(z_1))$, and it acts on a vector as an elementwise product.
$\nabla_x L = W_1^\top \nabla_{z_1} L$, with shapes $(n \times k)(k \times 1) = n \times 1$.	The same rule once more, with $\partial z_1 / \partial x = W_1$.
$\nabla_x L = W_1^\top (\sigma'(z_1) \odot W_2^\top (h_2 - y))$	Substitute steps 3 and 4 into step 5. Evaluated from the inside out, this is backpropagation: the last layer's transpose acts first and the first layer's acts last.

Problem 8

Let $L = \frac{1}{2} \|Wx - y\|^2$ with $W \in \mathbb{R}^{m \times n}$. Compute $\nabla_W L$ and state its shape.

Let $r = Wx - y$. Then $L = \frac{1}{2} \sum_i r_i^2$ with $r_i = \sum_j W_{ij} x_j - y_i$, so $L = \frac{1}{2} \sum_i \left(\sum_j W_{ij} x_j - y_i \right)^2$.	A matrix gradient is defined entry by entry, so write L in terms of the entries W_{ij} .
--	--

$\partial L / \partial W_{ij} = r_i \partial r_i / \partial W_{ij}.$	W_{ij} sits in row i of W , which only builds r_i , so only the i -th square depends on it; its derivative is r_i times the derivative of r_i .
$\partial r_i / \partial W_{ij} = x_j.$	In $\sum_{j'} W_{ij'} x_{j'}$, only the $j' = j$ term contains W_{ij} , with coefficient x_j .
$(\nabla_W L)_{ij} = (Wx - y)_i x_j.$	Entry (i, j) of $\nabla_W L$ is $\partial L / \partial W_{ij}$ (the previous page's convention).
A matrix with entries $a_i b_j$ is the outer product $ab^\top$.	$(ab^\top)_{ij} = a_i b_j$: a supplies the row index and b the column index.
$\nabla_W L = (Wx - y) x^\top, m \times n$	Step 5 with $a = Wx - y$ and $b = x$. Shape check: $(m \times 1)(1 \times n) = m \times n$, the shape of W , so $W - \eta \nabla_W L$ makes sense.

Problem 9

Let $u = x / \|x\|_2$ for $x \neq 0$. Compute $\partial u / \partial x$.

$u = x \ x\ ^{-1}.$	A vector times a scalar function of x , which the product rule handles.
$du = \ x\ ^{-1} dx + x d(\ x\ ^{-1}).$	Product rule; both factors depend on x .
$d(\ x\ ^{-1}) = -\ x\ ^{-2} d\ x\ = -\ x\ ^{-3} x^\top dx.$	Scalar chain rule for $t \mapsto 1/t$, then $d\ x\ = x^\top dx / \ x\ $, which is the previous page's $\nabla_x \ x\ = x / \ x\ $ written as a differential.
$du = \ x\ ^{-1} dx - \ x\ ^{-3} x x^\top dx.$	Substitute step 3; $x (x^\top dx) = (xx^\top) dx$ by associativity.
$\partial u / \partial x = \frac{1}{\ x\ } \left(I - \frac{xx^\top}{\ x\ ^2} \right), n \times n.$	The Jacobian is the matrix that multiplies dx ; factor out $1/\ x\ $.

Let $\hat{x} = x/\ x\ $, the unit vector along x (it equals u). Then $xx^\top/\ x\ ^2 = \hat{x}\hat{x}^\top$.	Split $\ x\ ^2$ into one $\ x\ $ for each factor of x .
$\partial u/\partial x = \frac{1}{\ x\ }(I - \hat{x}\hat{x}^\top)$, $\hat{x} = x/\ x\ $	Sanity check: scaling x does not change u , so the Jacobian must send the direction of x to zero, and $(I - \hat{x}\hat{x}^\top)\hat{x} = \hat{x} - \hat{x} = 0$.

Problem 10

Compute $\nabla_x \|\sigma(Wx)\|_2^2$.

Let $z = Wx$ and $h = \sigma(z)$, so the function is $f = \ h\ ^2 = h^\top h$.	Naming the pre-activation and the output splits f into a layer (Problem 3) and a squared norm (previous page).
$\nabla_h f = 2h = 2\sigma(z)$.	The previous page's $\nabla_x(x^\top Ax) = (A + A^\top)x$ with $A = I$.
$\nabla_x f = (\partial h/\partial x)^\top \nabla_h f$.	Problem 6: the gradient with respect to x is the transposed Jacobian of h times the gradient with respect to h .
$(\partial h/\partial x)^\top = W^\top \text{diag}(\sigma'(z))$.	Problem 3 with $b = 0$, transposed.
$\nabla_x f = W^\top (\sigma'(z) \odot 2\sigma(z))$, with shapes $(n \times m)(m \times 1) = n \times 1$.	The diagonal matrix acts on $2\sigma(z)$ as an elementwise product (Problem 6).
$\nabla_x \ \sigma(Wx)\ ^2 = 2W^\top (\sigma'(z) \odot \sigma(z))$, $z = Wx$	The scalar 2 moves out of the elementwise product and the matrix product. For the sigmoid, $\sigma'(z) \odot \sigma(z) = h \odot h \odot (1 - h)$, which is how it is computed from the saved forward output.

Where this goes wrong

1. Multiplying the Jacobians in the wrong order

The forward pass applies A first and B second, and scalars commute, so writing the derivatives in the order the maps run looks harmless.

$y = B(Ax); \partial(Ax)/\partial x = A$ and $\partial(Bu)/\partial u = B$	Right so far: both Jacobians are correct, and only their order is still to be decided.
“ A is applied first and B second, so the derivative is A then B .”	The analogy that causes the mistake: reading the product in the order the forward pass runs, as if it were the scalar $g'(f(x)) f'(x)$, whose factors commute.
$\partial y/\partial x = AB \times$	In numerator layout the inner Jacobian goes on the right: $y = (BA)x$, so the Jacobian is BA . The shapes refuse AB : $(m \times n)(p \times m)$ exists only when $n = p$, and even then it is $m \times m$ instead of $p \times n$.

2. Evaluating σ' at the output instead of the pre-activation

For the sigmoid, $\sigma'(z) = \sigma(z)(1 - \sigma(z)) = h(1 - h)$, so the derivative is usually computed from the saved output h . Written as a function call, that invites a slip.

$z = Wx + b, h = \sigma(z), L = \frac{1}{2} \ h - y\ ^2,$ $\nabla_h L = h - y$	Right so far: this is steps 1 and 2 of Problem 5.
$\sigma'(z) = h \odot (1 - h)$, “so the sigmoid's derivative is a function of h ”	The identity is true; reading it as “ σ' takes h ” is the analogy that causes the mistake.

$$\nabla_x L = W^\top (\sigma'(h) \odot (h - y)) \quad \text{X}$$

σ' is a function of z : the chain rule evaluates the derivative of σ at the point σ was applied to. $\sigma'(h) = \sigma(h)(1 - \sigma(h))$ applies the sigmoid twice. The correct line is $W^\top (\sigma'(z) \odot (h - y))$, which is $W^\top (h \odot (1 - h) \odot (h - y))$. The slip survives because for $h \in (0, 1)$, $\sigma'(h)$ lies between about 0.197 and 0.25, so the gradient has a plausible size. A finite-difference check catches it at once, and so does noticing that the factor never drops below about 0.197, where the true $\sigma'(z)$ goes to 0 for large $|z|$.

3. Writing the Jacobian of an elementwise layer as a vector

Problem 6 says never to build $\text{diag}(\sigma'(z))$ in code, and the shorthand that replaces it leaks back into the derivation.

$$h = \sigma(z), L = \frac{1}{2} \|h - y\|^2, \partial L / \partial h = (h - y)^\top, 1 \times m$$

Right so far: the row derivative of the loss with respect to h .

“The $m \times m$ Jacobian is diagonal, so keep only the diagonal.”

The analogy that causes the mistake: the storage advice of Problem 6, which is about the product, is applied to the Jacobian itself.

$$\partial h / \partial z = \sigma'(z) \text{ (a vector) } \quad \text{X}$$

An $m \times 1$ vector is not the Jacobian of an m -vector with respect to an m -vector. The chain rule then gives $(h - y)^\top \sigma'(z)$, a 1×1 inner product where a $1 \times m$ row belongs. As a Jacobian, in a derivation, write $\text{diag}(\sigma'(z))$, $m \times m$. As a vector-Jacobian product, in code or a gradient formula, write $\sigma'(z) \odot g$ for the incoming gradient g . Both say the same thing; the bare vector says neither.

4. Transposing until the shapes fit

The scalar rule $\frac{d}{dw} \frac{1}{2}(wx - y)^2 = (wx - y)x$ says there are two factors, $Wx - y$ and x , and it is tempting to arrange them in whichever order multiplies.

$L = \frac{1}{2}\|Wx - y\|^2$ with $W \in \mathbb{R}^{m \times n}$; the factors are $Wx - y$ ($m \times 1$) and x ($n \times 1$)

Right so far: these are the two factors in the gradient.

“($Wx - y$)[⊤] x does not multiply unless $m = n$, so transpose the other one.”

The analogy that causes the mistake: the scalar rule gives the factors but not their arrangement, and the shapes are used to pick one.

$$\nabla_W L = x (Wx - y)^\top \quad \text{✗}$$

It multiplies, but it is $n \times m$, the transpose of W 's shape, so $W - \eta \nabla_W L$ is undefined unless $m = n$, and wrong even then. The index derivation (Problem 8) gives $\partial L / \partial W_{ij} = (Wx - y)_i x_j$: the row index belongs to $Wx - y$ and the column index to x , so the gradient is $(Wx - y) x^\top$. The gradient must have W 's shape, and that fixes which factor is the column.