Automatic Differentiation

Alfan F. Wicaksono

Fakultas Ilmu Komputer, Universitas Indonesia

Some slides are taken from “Alice’s Adventures in a Differentiable Wonderland” by **Simone Scardapane**.

Why not numerical differentiation?

- We could directly apply the definition of gradients to obtain a suitable numerical approximation of the gradient.

$$\partial_{x_i} f(\mathbf{x}) = \frac{\partial y}{\partial x_i} = \lim_{h \to 0} \frac{f(\mathbf{x} + h\mathbf{e}_i) - f(\mathbf{x})}{h},$$

butuh 2 function call

anggap kalau function tersebut punya parameter yang besar. Pasti lama banget

- However, each scalar value to be differentiated requires **2 function calls** in a naive implementation, making this approach unfeasible except for numerical checks over the implementation.

Why not symbolic differentiation?

- We can ask a symbolic engine to pre-compute the full, symbolic equation of the derivative. This is called symbolic differentiation.
- Finding an optimal implementation for the Jacobian which avoids any unnecessary computation is an **NP-complete task** (**optimal Jacobian accumulation**).

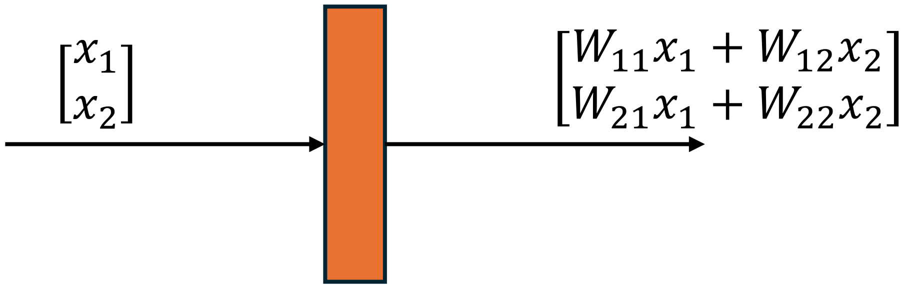
```python
import sympy as sp
x, a, b = sp.symbols('x a b')
y = a*sp.sin(x) + b*x*sp.sin(x)
sp.diff(y, x) # [Out]: acos(x)+bxcos(x)+bsin(x)
```

Box C.6.1: *Symbolic differentiation in Python using SymPy.*

$f(x, \boldsymbol{w})$ adalah sebuah fungsi "fully-connected layer" dengan **input vektor ~ (c)** dan **output vektor ~ (c)**; dan mempunyai parameter **W ~ (c, c)**.

Contoh, untuk c = 2:

loss.backward() itu isinya apa, akan dibahas



$$f(\boldsymbol{x}, \boldsymbol{W}) = \boldsymbol{W}\boldsymbol{x} = \begin{bmatrix} W_{11} & W_{12} \\ W_{21} & W_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

Saat proses optimization untuk mencari **W**, biasanya kita membutuhkan perhitungan dua buah kuantitas:

Input Jacobian : $\partial_{\boldsymbol{x}} f(\boldsymbol{x}, \boldsymbol{W})$

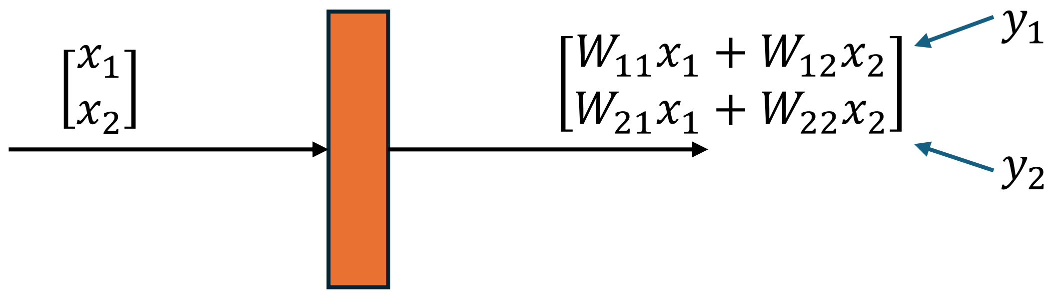
ada 2 jacobian yang bisa dihitugn:

bisa hitung turunan f terhadap x atau f terhadap w

Weight Jacobian : $\partial_{\boldsymbol{W}} f(\boldsymbol{x}, \boldsymbol{W})$

$f(x, \boldsymbol{w})$ adalah sebuah fungsi "fully-connected layer" dengan **input vektor ~ (c)** dan **output vektor ~ (c)**; dan mempunyai parameter **W ~ (c, c)**.

Contoh, untuk c = 2:



$$f(\boldsymbol{x}, \boldsymbol{W}) = \boldsymbol{W}\boldsymbol{x} = \begin{bmatrix} W_{11} & W_{12} \\ W_{21} & W_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

kalau kayak gini dia akan turunannya 4 parameter (fitur)

lihat polanya yang baris pertama itu y1, baris kedua itu y2
y ini itu kayak berapa baris yang kita miliki.

Untuk contoh di atas:

Weight Jacobian ~ (2, 2*2):

$$\partial_{\boldsymbol{W}} f(\boldsymbol{x}, \boldsymbol{W}) = \begin{bmatrix} \frac{\partial y_1}{\partial W_{11}} & \frac{\partial y_1}{\partial W_{12}} & \frac{\partial y_1}{\partial W_{21}} & \frac{\partial y_1}{\partial W_{22}} \\ \frac{\partial y_2}{\partial W_{11}} & \frac{\partial y_2}{\partial W_{12}} & \frac{\partial y_2}{\partial W_{21}} & \frac{\partial y_2}{\partial W_{22}} \end{bmatrix}$$

$$\partial_{\boldsymbol{W}} f(\boldsymbol{x}, \boldsymbol{W}) = \begin{bmatrix} x_1 & x_2 & 0 & 0 \\ 0 & 0 & x_1 & x_2 \end{bmatrix}$$

Input Jacobian ~ (2, 2):

input jacobian = W

$$\partial_{\boldsymbol{x}} f(\boldsymbol{x}, \boldsymbol{W}) = \boldsymbol{W} = \begin{bmatrix} W_{11} & W_{12} \\ W_{21} & W_{22} \end{bmatrix}$$

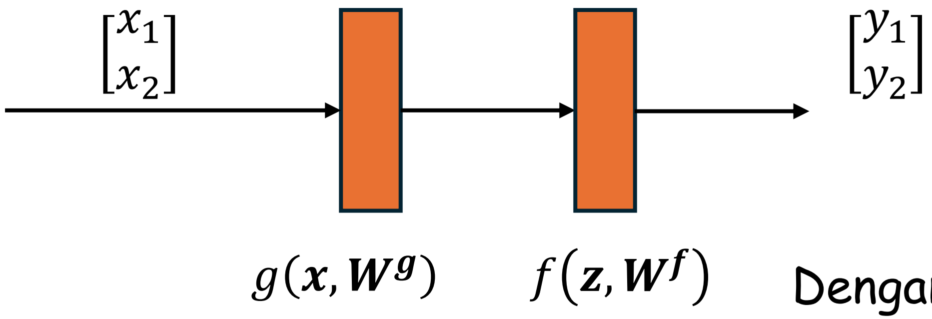
Setelah Anda mengetahui cara menghitung **input jacobian** dan **weight jacobian** dari sebuah fungsi **fully-connected layer** berparameter, sekarang kita mencoba jika terdapat 2 buah layer.

$\boldsymbol{W}^f$ bukan berarti **W** pangkat **f**; tetapi parameter **W** pada fungsi **f**. Fungsi **f** dan **g** punya parameter yang berbeda.

Sekarang, seandainya ada 2 layer:

$$f\left(\boldsymbol{z}, \boldsymbol{W}^f\right) = \boldsymbol{W}^f \boldsymbol{z} = \begin{bmatrix} W_{11}^f & W_{12}^f \\ W_{21}^f & W_{22}^f \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix}$$

$$g(\boldsymbol{x}, \boldsymbol{W}^g) = \boldsymbol{W}^g \boldsymbol{x} = \begin{bmatrix} W_{11}^g & W_{12}^g \\ W_{21}^g & W_{22}^g \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$



Dengan kata lain, ini adalah komposisi fungsi:

$$y = f\left(g(\boldsymbol{x}, \boldsymbol{W}^g), \boldsymbol{W}^f\right)$$

Biasanya, kita ingin menghitung $\partial \boldsymbol{y} / \partial \boldsymbol{W}^g$ dan $\partial \boldsymbol{y} / \partial \boldsymbol{W}^f$.

turunan terhadap w pawda fungsi g dan w pada fungsi f

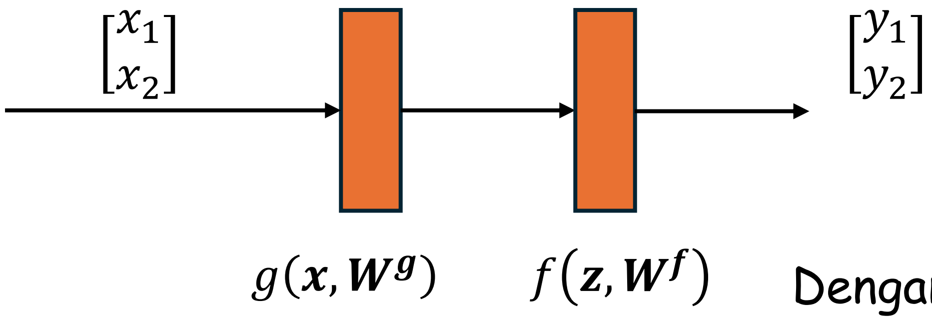
Bagaimana caranya?

$\boldsymbol{W}^f$ bukan berarti **W** pangkat **f**; tetapi parameter **W** pada fungsi **f**. Fungsi **f** dan **g** punya parameter yang berbeda.

Sekarang, seandainya ada 2 layer:

$$f(\boldsymbol{z}, \boldsymbol{W}^f) = \boldsymbol{W}^f \boldsymbol{z} = \begin{bmatrix} W_{11}^f & W_{12}^f \\ W_{21}^f & W_{22}^f \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix}$$

$$g(\boldsymbol{x}, \boldsymbol{W}^g) = \boldsymbol{W}^g \boldsymbol{x} = \begin{bmatrix} W_{11}^g & W_{12}^g \\ W_{21}^g & W_{22}^g \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$



Dengan kata lain, ini adalah komposisi fungsi:

$$y = f\left(g(\boldsymbol{x}, \boldsymbol{W}^g), \boldsymbol{W}^f\right)$$

Biasanya, kita ingin menghitung $\partial \boldsymbol{y} / \partial \boldsymbol{W}^g$ dan $\partial \boldsymbol{y} / \partial \boldsymbol{W}^f$.

Chain Rule again!

$$\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{W}^g} = \frac{\partial f}{\partial \boldsymbol{W}^g} = \frac{\partial f}{\partial g} \cdot \frac{\partial g}{\partial \boldsymbol{W}^g}$$

Input Jacobian f: $\partial_{\boldsymbol{x}} f(., \boldsymbol{W}^f)$

Weight Jacobian **g**: $\partial_{\boldsymbol{W}^g} g(., \boldsymbol{W}^g)$

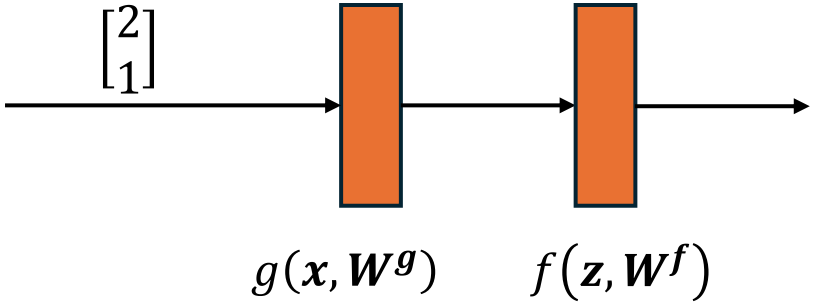
kalau udah akhir langsung aja diturunin karena

$$\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{W}^f} = \frac{\partial f}{\partial \boldsymbol{W}^f}$$

Weight Jacobian **f**: $\partial_{\boldsymbol{W}^f} f(., \boldsymbol{W}^f)$

ini simulasi SGD (Schotastic gradient descent)

Sekarang, dengan input tertentu dan parameter **W** diinisialisasi dengan nilai tertentu.



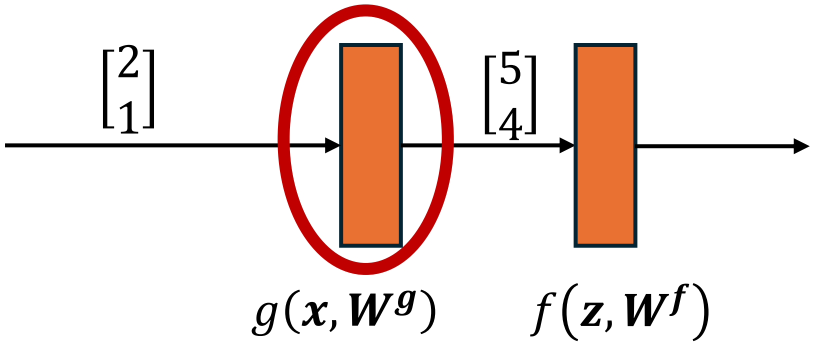
$$f(\boldsymbol{z}, \boldsymbol{W}^f) = \boldsymbol{W}^f \boldsymbol{z} = \begin{bmatrix}1 & 2\\1 & 1\end{bmatrix}\begin{bmatrix}z_1\\z_2\end{bmatrix}$$

$$g(\boldsymbol{x}, \boldsymbol{W}^g) = \boldsymbol{W}^g \boldsymbol{x} = \begin{bmatrix}2 & 1\\1 & 2\end{bmatrix}\begin{bmatrix}x_1\\x_2\end{bmatrix}$$

$$\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{W}^g} = \frac{\partial f}{\partial \boldsymbol{W}^g} = \frac{\partial f}{\partial g} \cdot \frac{\partial g}{\partial \boldsymbol{W}^g}$$

$$\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{W}^f} = \frac{\partial f}{\partial \boldsymbol{W}^f}$$

Sekarang, dengan input tertentu dan parameter **W** diinisialisasi dengan nilai tertentu.



$$\partial_{\boldsymbol{W}^g} g(\boldsymbol{z}, \boldsymbol{W}^g) = \begin{bmatrix} x_1 & x_2 & 0 & 0 \\ 0 & 0 & x_1 & x_2 \end{bmatrix}$$

$$\partial_{\boldsymbol{W}^g} g(\boldsymbol{z}, \boldsymbol{W}^g) = \begin{bmatrix} 2 & 1 & 0 & 0 \\ 0 & 0 & 2 & 1 \end{bmatrix}$$

$$f(\boldsymbol{z}, \boldsymbol{W}^f) = \boldsymbol{W}^f \boldsymbol{z} = \begin{bmatrix} 1 & 2 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix}$$

$$g(\boldsymbol{x}, \boldsymbol{W}^g) = \boldsymbol{W}^g \boldsymbol{x} = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} 2 \\ 1 \end{bmatrix} = \begin{bmatrix} 5 \\ 4 \end{bmatrix}$$

Buat sebuah matrix $\widehat{W^g}$ yang diinisialisasi dengan Weight Jacobian g:

$$\widehat{W^g} \leftarrow \begin{bmatrix} 2 & 1 & 0 & 0 \\ 0 & 0 & 2 & 1 \end{bmatrix}$$

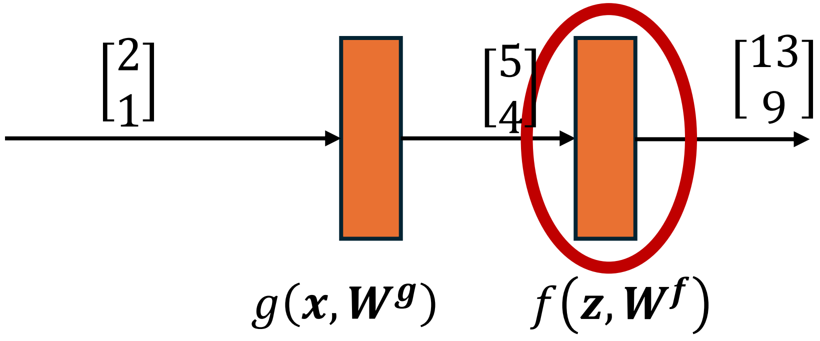
$$\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{W}^g} = \frac{\partial f}{\partial \boldsymbol{W}^g} = \frac{\partial f}{\partial g} \cdot \boxed{\frac{\partial g}{\partial \boldsymbol{W}^g}}$$

$$\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{W}^f} = \frac{\partial f}{\partial \boldsymbol{W}^f}$$

Sekarang, dengan input tertentu dan parameter **W** diinisialisasi dengan nilai tertentu.

$$f(\boldsymbol{z}, \boldsymbol{W}^f) = \boldsymbol{W}^f \boldsymbol{z} = \begin{bmatrix} 1 & 2 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 5 \\ 4 \end{bmatrix} = \begin{bmatrix} 13 \\ 9 \end{bmatrix}$$

$$g(\boldsymbol{x}, \boldsymbol{W}^g) = \boldsymbol{W}^g \boldsymbol{x} = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} 2 \\ 1 \end{bmatrix} = \begin{bmatrix} 5 \\ 4 \end{bmatrix}$$



$$\partial_{\boldsymbol{z}} f(\boldsymbol{z}, \boldsymbol{W}^f) = \begin{bmatrix} W_{11}^f & W_{12}^f \\ W_{21}^f & W_{22}^f \end{bmatrix}$$

$$\partial_{\boldsymbol{z}} f(\boldsymbol{z}, \boldsymbol{W}^f) = \begin{bmatrix} 1 & 2 \\ 1 & 1 \end{bmatrix}$$

$$\partial_{W^f} f(\boldsymbol{z}, \boldsymbol{W}^f) = \begin{bmatrix} z_1 & z_2 & 0 & 0 \\ 0 & 0 & z_1 & z_2 \end{bmatrix}$$

$$\partial_{W^f} f(\boldsymbol{z}, \boldsymbol{W}^f) = \begin{bmatrix} 5 & 4 & 0 & 0 \\ 0 & 0 & 5 & 4 \end{bmatrix}$$

Update matrix $\widehat{W^g}$ dikali dengan Input Jacobian **f**.

$$\widehat{W^g} \leftarrow \partial_{\boldsymbol{z}} f(\boldsymbol{z}, \boldsymbol{W}^f) \times \widehat{W^g}$$

$$\widehat{W^g} \leftarrow \begin{bmatrix} 1 & 2 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 2 & 1 & 0 & 0 \\ 0 & 0 & 2 & 1 \end{bmatrix} = \begin{bmatrix} 2 & 1 & 4 & 2 \\ 2 & 1 & 2 & 1 \end{bmatrix}$$

Buat sebuah matrix $\widehat{W^f}$ yang diinisialisasi dengan Weight Jacobian f:

$$\widehat{W^f} \leftarrow \begin{bmatrix} 5 & 4 & 0 & 0 \\ 0 & 0 & 5 & 4 \end{bmatrix}$$

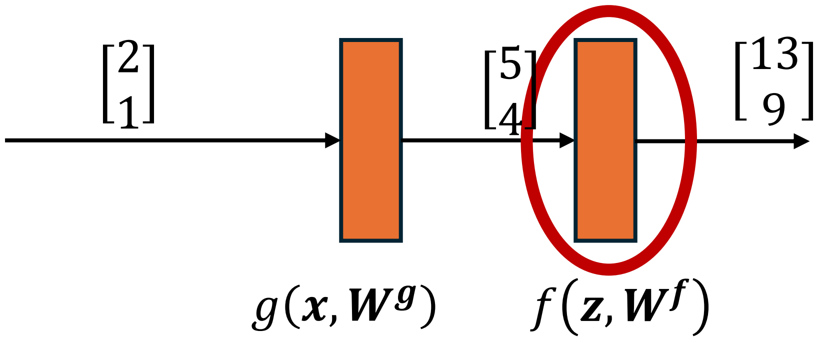
input jacobian

$$\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{W}^g} = \frac{\partial f}{\partial \boldsymbol{W}^g} = \boxed{\frac{\partial f}{\partial g} \cdot \frac{\partial g}{\partial \boldsymbol{W}^g}}$$

$$\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{W}^f} = \boxed{\frac{\partial f}{\partial \boldsymbol{W}^f}}$$

weight jacobian

Sekarang, dengan input tertentu dan parameter **W** diinisialisasi dengan nilai tertentu.



$$f(\boldsymbol{z}, \boldsymbol{W}^f) = \boldsymbol{W}^f \boldsymbol{z} = \begin{bmatrix}1 & 2\\1 & 1\end{bmatrix}\begin{bmatrix}5\\4\end{bmatrix} = \begin{bmatrix}13\\9\end{bmatrix}$$

$$g(\boldsymbol{x}, \boldsymbol{W}^g) = \boldsymbol{W}^g \boldsymbol{x} = \begin{bmatrix}2 & 1\\1 & 2\end{bmatrix}\begin{bmatrix}2\\1\end{bmatrix} = \begin{bmatrix}5\\4\end{bmatrix}$$

$$\frac{\partial y_1}{\partial W^g_{12}} \qquad \frac{\partial y_2}{\partial W^g_{21}}$$

turunan output terhadap sebuah parameter

$$\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{W}^g} = \frac{\partial f}{\partial \boldsymbol{W}^g} = \boxed{\frac{\partial f}{\partial g} \cdot \frac{\partial g}{\partial \boldsymbol{W}^g}} = \widehat{W^g}$$

$$\widehat{W^g} = \begin{bmatrix}2 & 1 & 4 & 2\\2 & 1 & 2 & 1\end{bmatrix}$$

$$\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{W}^f} = \boxed{\frac{\partial f}{\partial \boldsymbol{W}^f}} = \widehat{W^f}$$

$$\widehat{W^f} = \begin{bmatrix}5 & 4 & 0 & 0\\0 & 0 & 5 & 4\end{bmatrix}$$

Sekarang, kita perumum semuanya, dan bungkus dalam sebuah konsep yang Bernama "Automatic Differentiation"

Automatic Differentiation: Problem Statement

Consider a sequence of **l** primitive calls:

$$\boldsymbol{h_1} = f_1(\boldsymbol{x}, \boldsymbol{W_1})$$

$$\boldsymbol{h_2} = f_2(\boldsymbol{h_1}, \boldsymbol{W_2})$$

$$\vdots$$

$$\boldsymbol{h_l} = f_l(\boldsymbol{h_{l-1}}, \boldsymbol{W_l})$$

outputnya scalar (harus loss function)

This is called an evaluation trace of the program. Roughly, the first ***l* − *1* operations** can represent several layers of a differentiable model, operation ***l*** can be a per-input loss (e.g., cross-entropy).

Hence, the output of our program is always a scalar, since we require it for numerical optimization.

Automatic Differentiation: Problem Statement

Definition D.6.1 (Automatic differentiation) *Given a program $F(\mathbf{x})$ composed of a sequence of differentiable primitives,* ***automatic differentiation*** *(AD) refers to the task of simultaneously and efficiently computing all weight Jacobians of the program given knowledge of the computational graph and all individuals input and weight Jacobians:*

$$AD(F(\mathbf{x})) = \left\{ \partial_{\mathbf{w}_i} y \right\}_{i=1}^{l}$$

There are two major classes of AD algorithms, called **forward-mode** and **backward-mode**, corresponding to a different ordering in the composition of the individual operations.

We will also see that the **backward-mode** (called **back-propagation** in the neural networks' literature) is significantly more efficient in our context.

Automatic Differentiation: Forward Mode AD

$$\boldsymbol{h}_1 = f_1(\boldsymbol{x}, \boldsymbol{W}_1)$$

$$\boldsymbol{h}_2 = f_2(\boldsymbol{h}_1, \boldsymbol{W}_2)$$

$$\vdots$$

$$\boldsymbol{h}_l = f_l(\boldsymbol{h}_{l-1}, \boldsymbol{W}_l)$$

The idea is that every time we apply a primitive function, we initialize its corresponding weight Jacobian (called **tangent** in this context) $\widehat{\boldsymbol{W}}_i$, while simultaneously updating all previous tangent matrices.

What we want to compute:

I perkalian matriks di ujung

$$\frac{\partial \boldsymbol{h}_l}{\partial \boldsymbol{W}_1} = \frac{\partial \boldsymbol{h}_l}{\partial \boldsymbol{h}_{l-1}} \cdot \frac{\partial \boldsymbol{h}_{l-1}}{\partial \boldsymbol{h}_{l-2}} \cdots \frac{\partial \boldsymbol{h}_2}{\partial \boldsymbol{h}_1} \cdot \frac{\partial \boldsymbol{h}_1}{\partial \boldsymbol{W}_1}$$

$$\frac{\partial \boldsymbol{h}_l}{\partial \boldsymbol{W}_2} = \frac{\partial \boldsymbol{h}_l}{\partial \boldsymbol{h}_{l-1}} \cdot \frac{\partial \boldsymbol{h}_{l-1}}{\partial \boldsymbol{h}_{l-2}} \cdots \frac{\partial \boldsymbol{h}_2}{\partial \boldsymbol{W}_2}$$

$$\vdots$$

Input jacobian of f_i

$$\widehat{\boldsymbol{W}}_j \leftarrow \left[\frac{\partial \boldsymbol{h}_i}{\partial \boldsymbol{h}_{i-1}}\right] \widehat{\boldsymbol{W}}_j \qquad \forall j < i$$

sisanya itu input jacobian.

Jadi weight jacobian hanya muncul 1 kali

yaitu di akhir

$$\widehat{\boldsymbol{W}}_i \leftarrow \frac{\partial \boldsymbol{h}_i}{\partial \boldsymbol{W}_i}$$

dihitung dulu 1 kali

weight jacobiannya

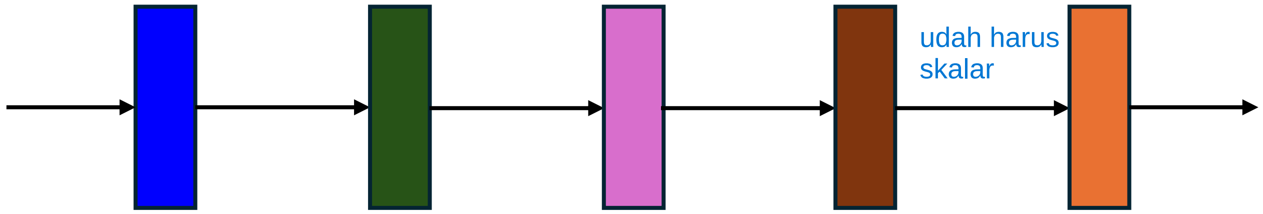
Weight jacobian of f_i

Automatic Differentiation: Forward Mode AD

$\boldsymbol{h_1} = f_1(\boldsymbol{x}, \boldsymbol{W}^{(1)})$

$\boldsymbol{h_3} = f_3(\boldsymbol{h_2}, \boldsymbol{W}^{(3)})$

$o = l(\boldsymbol{h_4}, \boldsymbol{y}) = (h_4 - y)^2$



$\boldsymbol{h_2} = f_2(\boldsymbol{h_1}, \boldsymbol{W}^{(2)})$

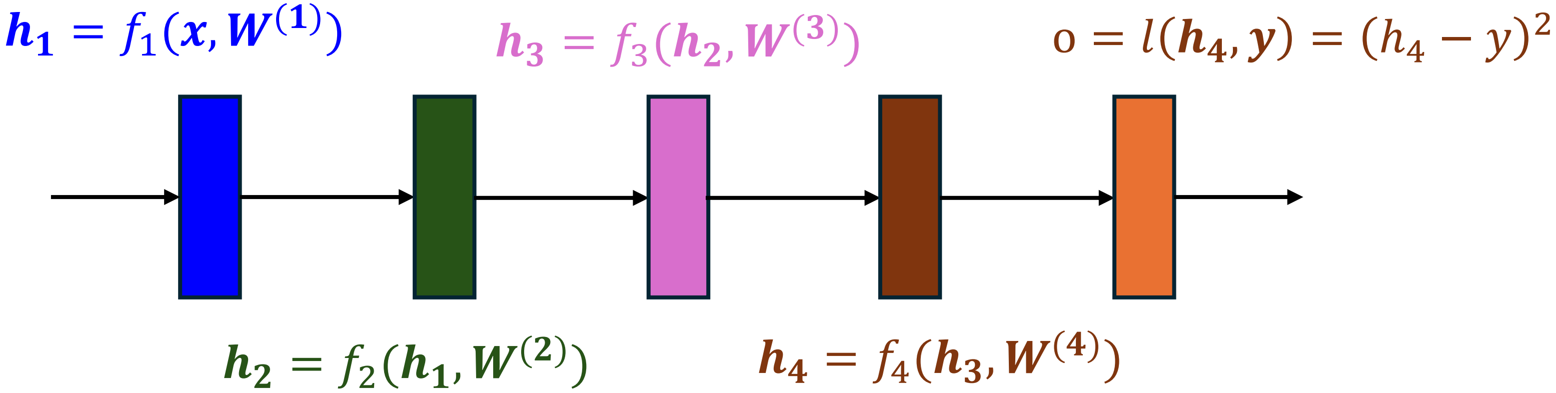
$\boldsymbol{h_4} = f_4(\boldsymbol{h_3}, \boldsymbol{W}^{(4)})$

ini itu udah loss functionnya di parameter terakhir (udah inputnya itu pred sama fx-1][])

Misal, Input adalah **vektor ~ (2)**.

Misal, 3 layer pertama adalah fully-connected models dengan **output ~ (2)**; layer ke-4 adalah fully connected model dengan output scalar; dan layer terakhir adalah fungsi loss, yaitu squared loss, yang mengembalikan scalar.

Automatic Differentiation: Forward Mode AD



$$f_1\left(\boldsymbol{x}, \boldsymbol{W}^{(1)}\right) = \boldsymbol{W}^{(1)}\boldsymbol{x} = \begin{bmatrix} W_{11}^{(1)} & W_{12}^{(1)} \\ W_{21}^{(1)} & W_{22}^{(1)} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

$$f_2\left(\boldsymbol{h_1}, \boldsymbol{W}^{(2)}\right) = \boldsymbol{W}^{(2)}\boldsymbol{h_1} = \begin{bmatrix} W_{11}^{(2)} & W_{12}^{(2)} \\ W_{21}^{(2)} & W_{22}^{(2)} \end{bmatrix} \begin{bmatrix} h_{11} \\ h_{12} \end{bmatrix}$$

$$f_3\left(\boldsymbol{h_2}, \boldsymbol{W}^{(3)}\right) = \boldsymbol{W}^{(3)}\boldsymbol{h_2} = \begin{bmatrix} W_{11}^{(3)} & W_{12}^{(3)} \\ W_{21}^{(3)} & W_{22}^{(3)} \end{bmatrix} \begin{bmatrix} h_{21} \\ h_{22} \end{bmatrix}$$

$$f_4\left(\boldsymbol{h_3}, \boldsymbol{W}^{(4)}\right) = \boldsymbol{W}^{(4)}\boldsymbol{h_3} = \begin{bmatrix} W_{11}^{(4)} & W_{12}^{(4)} \end{bmatrix} \begin{bmatrix} h_{31} \\ h_{32} \end{bmatrix}$$

$$l(h_4, \boldsymbol{y}) = (h_4 - y)^2$$

Total ada 14 parameter!

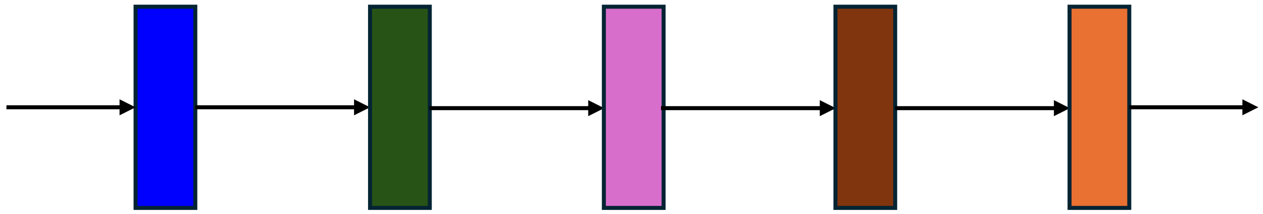
Automatic Differentiation: Forward Mode AD

$\boldsymbol{h_1} = f_1(\boldsymbol{x}, \boldsymbol{W}^{(1)})$ $\boldsymbol{h_3} = f_3(\boldsymbol{h_2}, \boldsymbol{W}^{(3)})$ $o = l(\boldsymbol{h_4}, \boldsymbol{y}) = (h_4 - y)^2$

$x = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

$y = 35$

ini y_true nya



$\boldsymbol{h_2} = f_2(\boldsymbol{h_1}, \boldsymbol{W}^{(2)})$ $\boldsymbol{h_4} = f_4(\boldsymbol{h_3}, \boldsymbol{W}^{(4)})$

$$f_1\left(\boldsymbol{x}, \boldsymbol{W}^{(1)}\right) = \boldsymbol{W}^{(1)}\boldsymbol{x} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

$$f_2\left(\boldsymbol{h_1}, \boldsymbol{W}^{(2)}\right) = \boldsymbol{W}^{(2)}\boldsymbol{h_1} = \begin{bmatrix} 1 & 2 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} h_{11} \\ h_{12} \end{bmatrix}$$

$$f_3\left(\boldsymbol{h_2}, \boldsymbol{W}^{(3)}\right) = \boldsymbol{W}^{(3)}\boldsymbol{h_2} = \begin{bmatrix} 1 & 1 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} h_{21} \\ h_{22} \end{bmatrix}$$

$$f_4\left(\boldsymbol{h_3}, \boldsymbol{W}^{(4)}\right) = \boldsymbol{W}^{(4)}\boldsymbol{h_3} = \begin{bmatrix} 2 & 1 \end{bmatrix} \begin{bmatrix} h_{31} \\ h_{32} \end{bmatrix}$$

$$l(h_4, \boldsymbol{y}) = (h_4 - y)^2$$

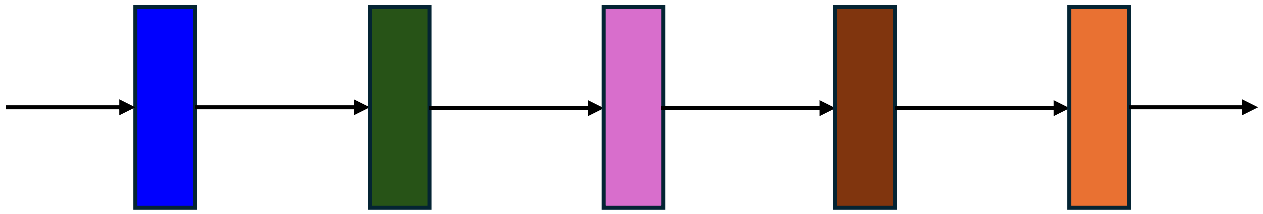
Misal, kita mulai dari inisialisasi parameter tertentu (random); dan untuk input x & y tertentu.

Automatic Differentiation: Forward Mode AD

$\boldsymbol{h_1} = f_1(\boldsymbol{x}, \boldsymbol{W^{(1)}})$ $\boldsymbol{h_3} = f_3(\boldsymbol{h_2}, \boldsymbol{W^{(3)}})$ $o = l(\boldsymbol{h_4}, \boldsymbol{y}) = (h_4 - y)^2$

$x = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

$y = 35$



$\boldsymbol{h_2} = f_2(\boldsymbol{h_1}, \boldsymbol{W^{(2)}})$ $\boldsymbol{h_4} = f_4(\boldsymbol{h_3}, \boldsymbol{W^{(4)}})$

karena loss function, kayaknya dia paling akhir itu weight nya itu 1x2 instead of 2x2

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(1)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{h_2}} \cdot \frac{\partial \boldsymbol{h_2}}{\partial \boldsymbol{h_1}} \cdot \frac{\partial \boldsymbol{h_1}}{\partial \boldsymbol{W}^{(1)}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(2)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{h_2}} \cdot \frac{\partial \boldsymbol{h_2}}{\partial \boldsymbol{W}^{(2)}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(3)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{W}^{(3)}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(4)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{W}^{(4)}}$$

Input dan output loss function sudah scalar. Turunannya juga scalar:

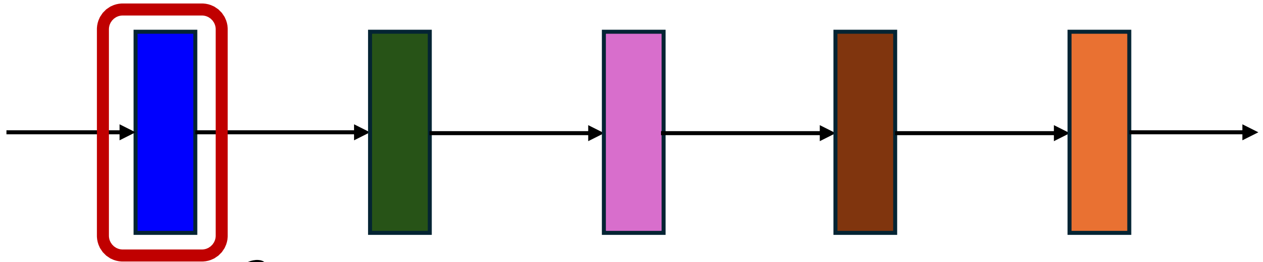
$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} = 2(h_4 - y) = 2(h_4 - 2)$$

Automatic Differentiation: Forward Mode AD

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} = 2(h_4 - 2)$$

$$x = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

$$y = 35$$



$$h_1 = \begin{bmatrix} 2 \\ 2 \end{bmatrix}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(1)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{h_2}} \cdot \frac{\partial \boldsymbol{h_2}}{\partial \boldsymbol{h_1}} \boxed{\frac{\partial \boldsymbol{h_1}}{\partial \boldsymbol{W}^{(1)}}}$$

$$\widehat{\boldsymbol{W}}_{\boldsymbol{1}} \leftarrow \frac{\partial \boldsymbol{h_1}}{\partial \boldsymbol{W}^{(1)}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(2)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{h_2}} \cdot \frac{\partial \boldsymbol{h_2}}{\partial \boldsymbol{W}^{(2)}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(3)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{W}^{(3)}}$$

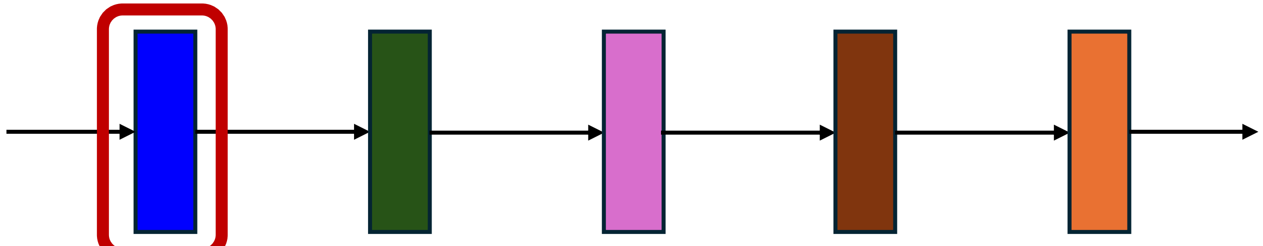
$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(4)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{W}^{(4)}}$$

Automatic Differentiation: Forward Mode AD

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} = 2(h_4 - 2)$$

$$x = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

$$y = 35$$



$$h_1 = \begin{bmatrix} 2 \\ 2 \end{bmatrix}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(1)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{h_2}} \cdot \frac{\partial \boldsymbol{h_2}}{\partial \boldsymbol{h_1}} \boxed{\frac{\partial \boldsymbol{h_1}}{\partial \boldsymbol{W}^{(1)}}}$$

$$\widehat{\boldsymbol{W}_1} \leftarrow \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(2)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{h_2}} \cdot \frac{\partial \boldsymbol{h_2}}{\partial \boldsymbol{W}^{(2)}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(3)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{W}^{(3)}}$$

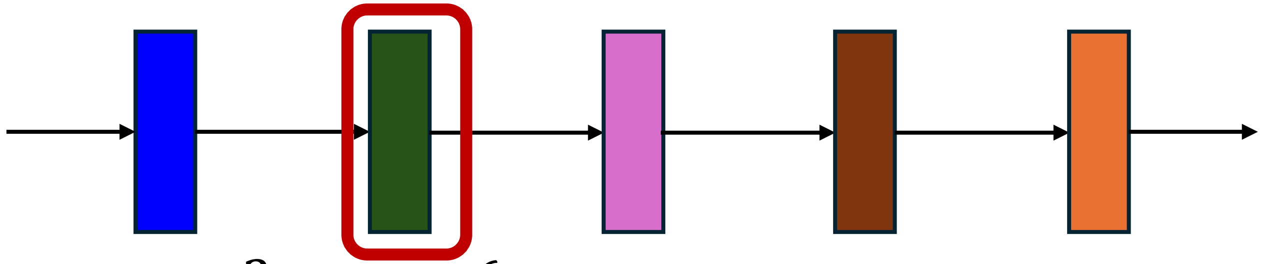
$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(4)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{W}^{(4)}}$$

Automatic Differentiation: Forward Mode AD

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} = 2(h_4 - 2)$$

$$x = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

$$y = 35$$



$$h_1 = \begin{bmatrix} 2 \\ 2 \end{bmatrix} \quad h_2 = \begin{bmatrix} 6 \\ 4 \end{bmatrix}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(1)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{h_2}} \cdot \boxed{\frac{\partial \boldsymbol{h_2}}{\partial \boldsymbol{h_1}} \cdot \frac{\partial \boldsymbol{h_1}}{\partial \boldsymbol{W}^{(1)}}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(2)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{h_2}} \cdot \boxed{\frac{\partial \boldsymbol{h_2}}{\partial \boldsymbol{W}^{(2)}}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(3)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{W}^{(3)}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(4)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{W}^{(4)}}$$

$$\widehat{\boldsymbol{W}}_{\boldsymbol{1}} \leftarrow \frac{\partial \boldsymbol{h_2}}{\partial \boldsymbol{h_1}} \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

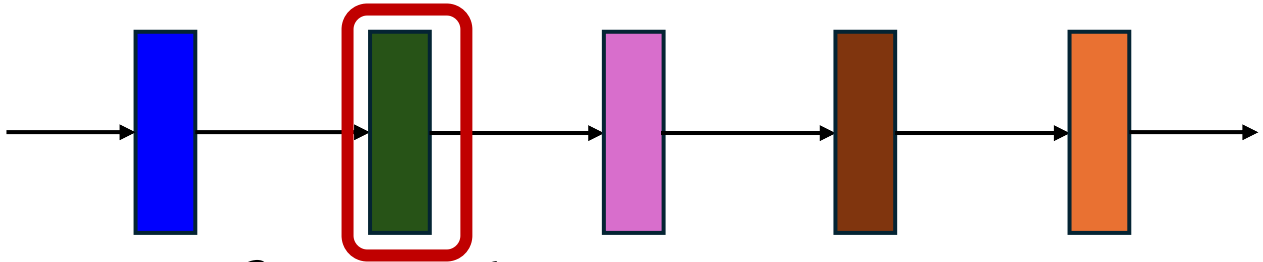
$$\widehat{\boldsymbol{W}}_{\boldsymbol{2}} \leftarrow \frac{\partial \boldsymbol{h_2}}{\partial \boldsymbol{W}^{(2)}}$$

Automatic Differentiation: Forward Mode AD

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} = 2(h_4 - 2)$$

$$x = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

$$y = 35$$



$$h_1 = \begin{bmatrix} 2 \\ 2 \end{bmatrix} \quad h_2 = \begin{bmatrix} 6 \\ 4 \end{bmatrix}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(1)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{h_2}} \cdot \boxed{\frac{\partial \boldsymbol{h_2}}{\partial \boldsymbol{h_1}} \cdot \frac{\partial \boldsymbol{h_1}}{\partial \boldsymbol{W}^{(1)}}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(2)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{h_2}} \cdot \boxed{\frac{\partial \boldsymbol{h_2}}{\partial \boldsymbol{W}^{(2)}}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(3)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{W}^{(3)}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(4)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{W}^{(4)}}$$

$$\widehat{\boldsymbol{W}}_{\boldsymbol{1}} \leftarrow \begin{bmatrix} 1 & 2 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

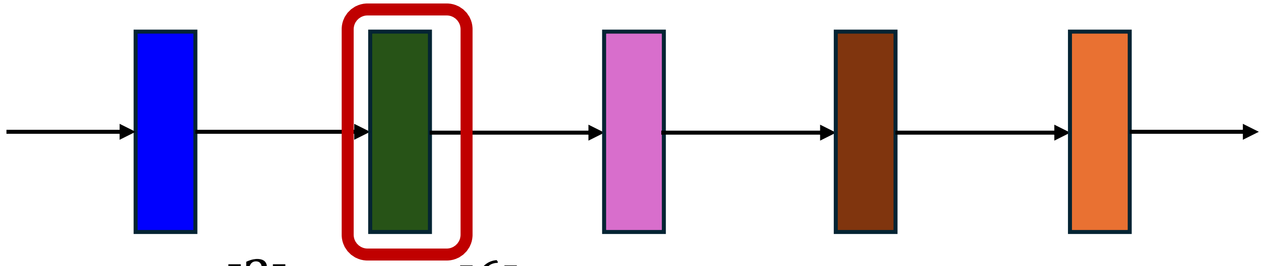
$$\widehat{\boldsymbol{W}}_{\boldsymbol{2}} \leftarrow \begin{bmatrix} 2 & 2 & 0 & 0 \\ 0 & 0 & 2 & 2 \end{bmatrix}$$

Automatic Differentiation: Forward Mode AD

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} = 2(h_4 - 2)$$

$x = \begin{bmatrix}1\\1\end{bmatrix}$

$y = 35$



$h_1 = \begin{bmatrix}2\\2\end{bmatrix}$ $h_2 = \begin{bmatrix}6\\4\end{bmatrix}$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(1)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{h_2}} \cdot \boxed{\frac{\partial \boldsymbol{h_2}}{\partial \boldsymbol{h_1}} \cdot \frac{\partial \boldsymbol{h_1}}{\partial \boldsymbol{W}^{(1)}}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(2)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{h_2}} \cdot \boxed{\frac{\partial \boldsymbol{h_2}}{\partial \boldsymbol{W}^{(2)}}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(3)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{W}^{(3)}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(4)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{W}^{(4)}}$$

$$\widehat{\boldsymbol{W}}_{\boldsymbol{1}} \leftarrow \begin{bmatrix}1 & 1 & 2 & 2\\1 & 1 & 1 & 1\end{bmatrix}$$

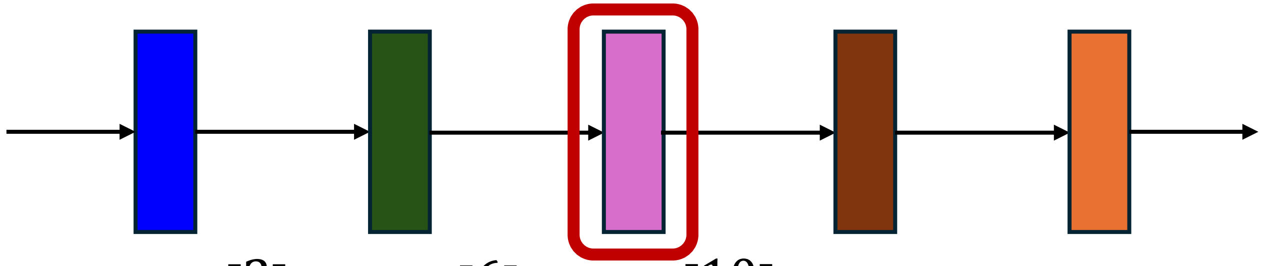
$$\widehat{\boldsymbol{W}}_{\boldsymbol{2}} \leftarrow \begin{bmatrix}2 & 2 & 0 & 0\\0 & 0 & 2 & 2\end{bmatrix}$$

Automatic Differentiation: Forward Mode AD

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} = 2(h_4 - 2)$$

$$x = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

$$y = 35$$



$$h_1 = \begin{bmatrix} 2 \\ 2 \end{bmatrix} \quad h_2 = \begin{bmatrix} 6 \\ 4 \end{bmatrix} \quad h_3 = \begin{bmatrix} 10 \\ 16 \end{bmatrix}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(1)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \boxed{\frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{h_2}} \cdot \frac{\partial \boldsymbol{h_2}}{\partial \boldsymbol{h_1}} \cdot \frac{\partial \boldsymbol{h_1}}{\partial \boldsymbol{W}^{(1)}}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(2)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \boxed{\frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{h_2}} \cdot \frac{\partial \boldsymbol{h_2}}{\partial \boldsymbol{W}^{(2)}}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(3)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \boxed{\frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{W}^{(3)}}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(4)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{W}^{(4)}}$$

$$\widehat{\boldsymbol{W}}_{\boldsymbol{1}} \leftarrow \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{h_2}} \begin{bmatrix} 1 & 1 & 2 & 2 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

$$\widehat{\boldsymbol{W}}_{\boldsymbol{2}} \leftarrow \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{h_2}} \begin{bmatrix} 2 & 2 & 0 & 0 \\ 0 & 0 & 2 & 2 \end{bmatrix}$$

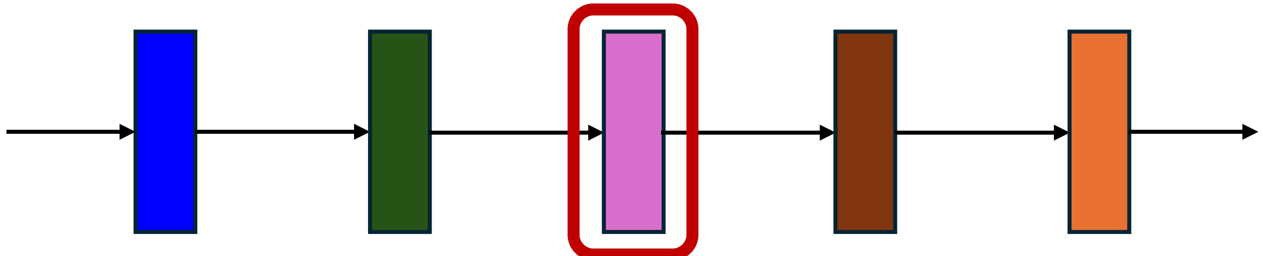
$$\widehat{\boldsymbol{W}}_{\boldsymbol{3}} \leftarrow \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{W}^{(3)}}$$

Automatic Differentiation: Forward Mode AD

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} = 2(h_4 - 2)$$

$$x = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

$$y = 35$$



$$h_1 = \begin{bmatrix} 2 \\ 2 \end{bmatrix} \quad h_2 = \begin{bmatrix} 6 \\ 4 \end{bmatrix} \quad h_3 = \begin{bmatrix} 10 \\ 16 \end{bmatrix}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(1)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \boxed{\frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{h_2}} \cdot \frac{\partial \boldsymbol{h_2}}{\partial \boldsymbol{h_1}} \cdot \frac{\partial \boldsymbol{h_1}}{\partial \boldsymbol{W}^{(1)}}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(2)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \boxed{\frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{h_2}} \cdot \frac{\partial \boldsymbol{h_2}}{\partial \boldsymbol{W}^{(2)}}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(3)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \boxed{\frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{W}^{(3)}}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(4)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{W}^{(4)}}$$

$$\widehat{\boldsymbol{W}}_{\boldsymbol{1}} \leftarrow \begin{bmatrix} 1 & 1 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 & 2 & 2 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

$$\widehat{\boldsymbol{W}}_{\boldsymbol{2}} \leftarrow \begin{bmatrix} 1 & 1 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} 2 & 2 & 0 & 0 \\ 0 & 0 & 2 & 2 \end{bmatrix}$$

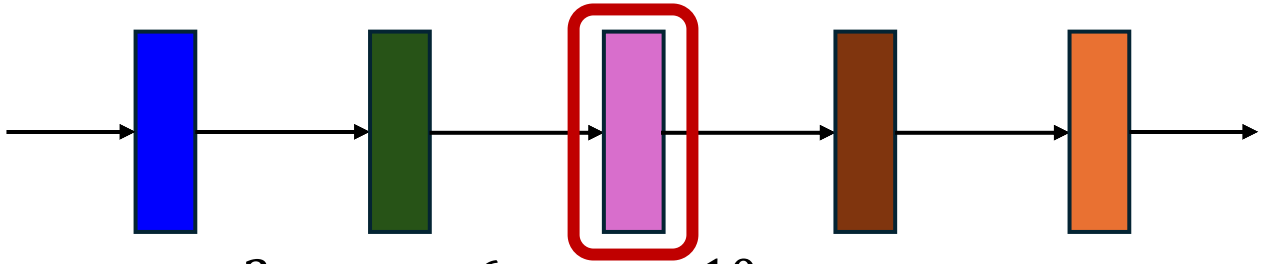
$$\widehat{\boldsymbol{W}}_{\boldsymbol{3}} \leftarrow \begin{bmatrix} 6 & 4 & 0 & 0 \\ 0 & 0 & 6 & 4 \end{bmatrix}$$

Automatic Differentiation: Forward Mode AD

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} = 2(h_4 - 2)$$

$x = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

$y = 35$



$h_1 = \begin{bmatrix} 2 \\ 2 \end{bmatrix}$ $h_2 = \begin{bmatrix} 6 \\ 4 \end{bmatrix}$ $h_3 = \begin{bmatrix} 10 \\ 16 \end{bmatrix}$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(1)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \boxed{\frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{h_2}} \cdot \frac{\partial \boldsymbol{h_2}}{\partial \boldsymbol{h_1}} \cdot \frac{\partial \boldsymbol{h_1}}{\partial \boldsymbol{W}^{(1)}}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(2)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \boxed{\frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{h_2}} \cdot \frac{\partial \boldsymbol{h_2}}{\partial \boldsymbol{W}^{(2)}}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(3)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \boxed{\frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{W}^{(3)}}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(4)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{W}^{(4)}}$$

$$\widehat{\boldsymbol{W}}_1 \leftarrow \begin{bmatrix} 2 & 2 & 3 & 3 \\ 3 & 3 & 5 & 5 \end{bmatrix}$$

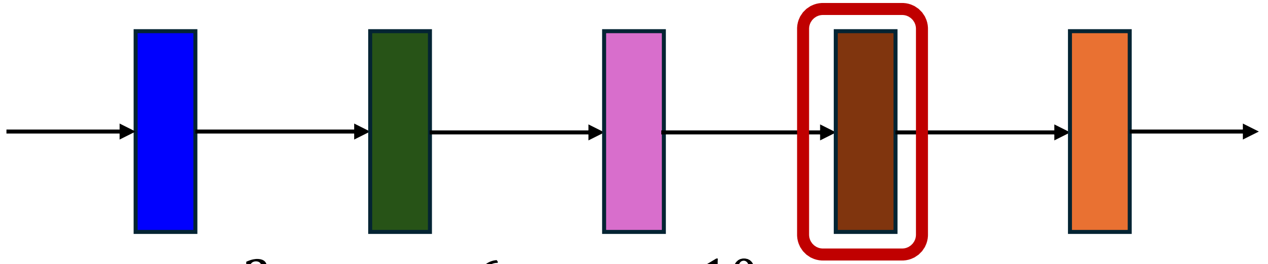
$$\widehat{\boldsymbol{W}}_2 \leftarrow \begin{bmatrix} 2 & 2 & 2 & 2 \\ 4 & 4 & 2 & 2 \end{bmatrix}$$

$$\widehat{\boldsymbol{W}}_3 \leftarrow \begin{bmatrix} 6 & 4 & 0 & 0 \\ 0 & 0 & 6 & 4 \end{bmatrix}$$

Automatic Differentiation: Forward Mode AD

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} = 2(h_4 - 2)$$

$$x = \begin{bmatrix}1\\1\end{bmatrix} \qquad y = 35$$



$$h_1 = \begin{bmatrix}2\\2\end{bmatrix} \quad h_2 = \begin{bmatrix}6\\4\end{bmatrix} \quad h_3 = \begin{bmatrix}10\\16\end{bmatrix} \quad h_4 = 36$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W^{(1)}}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \boxed{\frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{h_2}} \cdot \frac{\partial \boldsymbol{h_2}}{\partial \boldsymbol{h_1}} \cdot \frac{\partial \boldsymbol{h_1}}{\partial \boldsymbol{W^{(1)}}}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W^{(2)}}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \boxed{\frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{h_2}} \cdot \frac{\partial \boldsymbol{h_2}}{\partial \boldsymbol{W^{(2)}}}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W^{(3)}}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \boxed{\frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{W^{(3)}}}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W^{(4)}}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \boxed{\frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{W^{(4)}}}}$$

$$\widehat{\boldsymbol{W}}_1 \leftarrow \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \begin{bmatrix}2 & 2 & 3 & 3\\3 & 3 & 5 & 5\end{bmatrix}$$

$$\widehat{\boldsymbol{W}}_2 \leftarrow \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \begin{bmatrix}2 & 2 & 2 & 2\\4 & 4 & 2 & 2\end{bmatrix}$$

$$\widehat{\boldsymbol{W}}_3 \leftarrow \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \begin{bmatrix}6 & 4 & 0 & 0\\0 & 0 & 6 & 4\end{bmatrix}$$

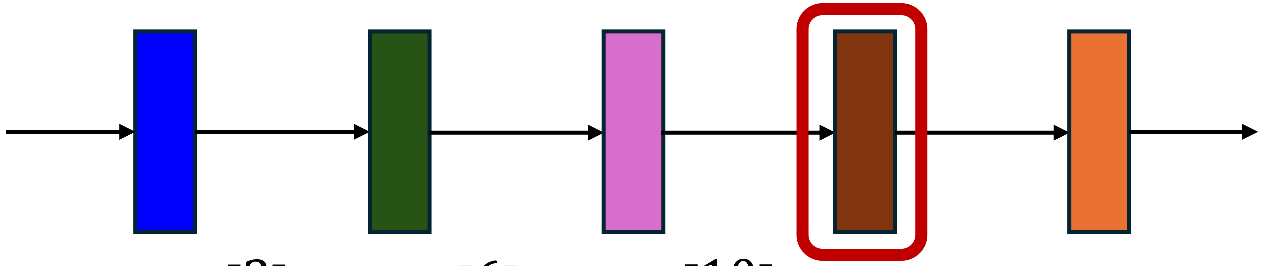
$$\widehat{\boldsymbol{W}}_4 \leftarrow \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{W^{(4)}}}$$

Automatic Differentiation: Forward Mode AD

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} = 2(h_4 - 2)$$

$$x = \begin{bmatrix}1\\1\end{bmatrix}$$

$$y = 35$$



$$h_1 = \begin{bmatrix}2\\2\end{bmatrix} \quad h_2 = \begin{bmatrix}6\\4\end{bmatrix} \quad h_3 = \begin{bmatrix}10\\16\end{bmatrix} \quad h_4 = 36$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(1)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \boxed{\frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{h_2}} \cdot \frac{\partial \boldsymbol{h_2}}{\partial \boldsymbol{h_1}} \cdot \frac{\partial \boldsymbol{h_1}}{\partial \boldsymbol{W}^{(1)}}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(2)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \boxed{\frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{h_2}} \cdot \frac{\partial \boldsymbol{h_2}}{\partial \boldsymbol{W}^{(2)}}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(3)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \boxed{\frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{W}^{(3)}}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(4)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \boxed{\frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{W}^{(4)}}}$$

$$\widehat{\boldsymbol{W}}_{\boldsymbol{1}} \leftarrow [2 \quad 1] \begin{bmatrix}2 & 2 & 3 & 3\\3 & 3 & 5 & 5\end{bmatrix}$$

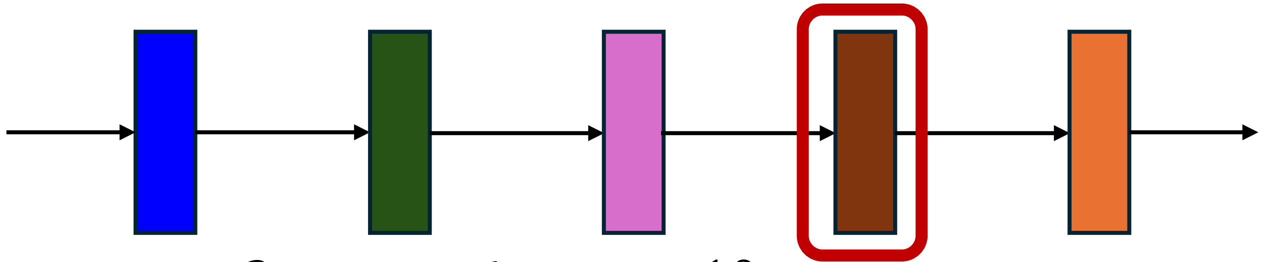
$$\widehat{\boldsymbol{W}}_{\boldsymbol{2}} \leftarrow [2 \quad 1] \begin{bmatrix}2 & 2 & 2 & 2\\4 & 4 & 2 & 2\end{bmatrix}$$

$$\widehat{\boldsymbol{W}}_{\boldsymbol{3}} \leftarrow [2 \quad 1] \begin{bmatrix}6 & 4 & 0 & 0\\0 & 0 & 6 & 4\end{bmatrix}$$

$$\widehat{\boldsymbol{W}}_{\boldsymbol{4}} \leftarrow [10 \quad 16]$$

Automatic Differentiation: Forward Mode AD

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} = 2(h_4 - 2)$$



$x = \begin{bmatrix}1\\1\end{bmatrix}$

$y = 35$

$h_1 = \begin{bmatrix}2\\2\end{bmatrix}$ $h_2 = \begin{bmatrix}6\\4\end{bmatrix}$ $h_3 = \begin{bmatrix}10\\16\end{bmatrix}$ $h_4 = 36$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(1)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \boxed{\frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{h_2}} \cdot \frac{\partial \boldsymbol{h_2}}{\partial \boldsymbol{h_1}} \cdot \frac{\partial \boldsymbol{h_1}}{\partial \boldsymbol{W}^{(1)}}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(2)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \boxed{\frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{h_2}} \cdot \frac{\partial \boldsymbol{h_2}}{\partial \boldsymbol{W}^{(2)}}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(3)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \boxed{\frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{W}^{(3)}}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(4)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \boxed{\frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{W}^{(4)}}}$$

$\widehat{\boldsymbol{W}}_{\boldsymbol{1}} \leftarrow [7 \quad 7 \quad 11 \quad 11]$

$\widehat{\boldsymbol{W}}_{\boldsymbol{2}} \leftarrow [8 \quad 8 \quad 6 \quad 6]$

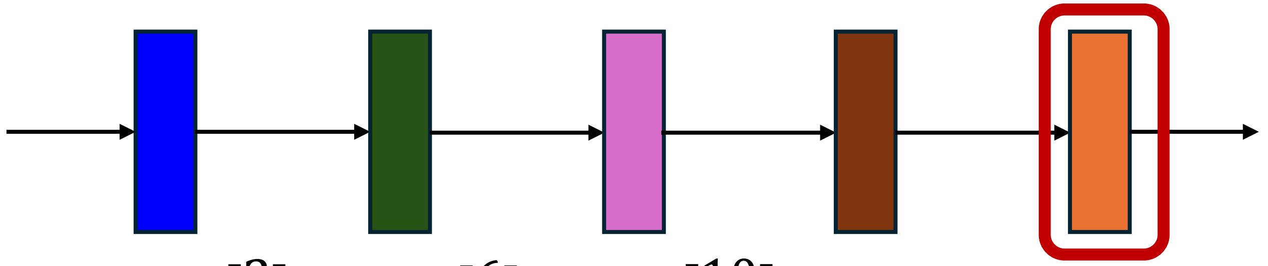
$\widehat{\boldsymbol{W}}_{\boldsymbol{3}} \leftarrow [12 \quad 8 \quad 6 \quad 4]$

$\widehat{\boldsymbol{W}}_{\boldsymbol{4}} \leftarrow [10 \quad 16]$

Automatic Differentiation: Forward Mode AD

$$x = \begin{bmatrix}1\\1\end{bmatrix}$$

$$y = 35$$



$$h_1 = \begin{bmatrix}2\\2\end{bmatrix} \quad h_2 = \begin{bmatrix}6\\4\end{bmatrix} \quad h_3 = \begin{bmatrix}10\\16\end{bmatrix} \quad h_4 = 36 \quad \mathrm{o} = 1$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(1)}} = \boxed{\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{h_2}} \cdot \frac{\partial \boldsymbol{h_2}}{\partial \boldsymbol{h_1}} \cdot \frac{\partial \boldsymbol{h_1}}{\partial \boldsymbol{W}^{(1)}}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(2)}} = \boxed{\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{h_2}} \cdot \frac{\partial \boldsymbol{h_2}}{\partial \boldsymbol{W}^{(2)}}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(3)}} = \boxed{\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{W}^{(3)}}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(4)}} = \boxed{\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{W}^{(4)}}}$$

$$\widehat{\boldsymbol{W}}_{\boldsymbol{1}} \leftarrow [476 \quad 476 \quad 748 \quad 748]$$

$$\widehat{\boldsymbol{W}}_{\boldsymbol{2}} \leftarrow [544 \quad 544 \quad 408 \quad 408]$$

$$\widehat{\boldsymbol{W}}_{\boldsymbol{3}} \leftarrow [816 \quad 544 \quad 408 \quad 272]$$

$$\widehat{\boldsymbol{W}}_{\boldsymbol{4}} \leftarrow [680 \quad 1088]$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} = 2(h_4 - 2) = 68$$

disini setiap kali ada N kali perkalian matriks, misalnya

sekarang ada 4 kali perkalian matriks pas di layer terakhir

If we have a sequence of matrix multiplications like bellow, where would you start in order to be efficient?

$$\begin{bmatrix} 2 & 1 \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

If we have a sequence of matrix multiplications like bellow, where would you start in order to be efficient?

selalu lebih efisien dari yang kecil karena nanti matriksnya lebih kecil.

$$\underbrace{\begin{bmatrix} 2 & 1 \end{bmatrix} \underbrace{\begin{bmatrix} 1 & 2 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}}_{2 \times 2 \times 4 = 16 \text{ basic operations}}}_{1 \times 2 \times 4 = 8 \text{ basic operations}}$$

Total 24 basic operations

If we have a sequence of matrix multiplications like bellow, where would you start in order to be efficient?

$$\begin{bmatrix} 2 & 1 \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

Total 12 basic operations!

1 x 2 x 2 = 4 basic operations

1 x 2 x 4 = 8 basic operations

Automatic Differentiation: Reverse Mode AD

$$\boldsymbol{h_1} = f_1(\boldsymbol{x}, \boldsymbol{W_1})$$

$$\boldsymbol{h_2} = f_2(\boldsymbol{h_1}, \boldsymbol{W_2})$$

$$\vdots$$

$$\boldsymbol{h_l} = f_l(\boldsymbol{h_{l-1}}, \boldsymbol{W_l})$$

Because matrix multiplication is associative, we can perform the computations in any order. In **Forward-AD**, we proceeded from **the right to the left**, since it corresponds to the ordering in which the primitive functions were executed. However, we can do better by noting two interesting aspects:

What we want to compute:

$$\frac{\partial \boldsymbol{h_l}}{\partial \boldsymbol{W_1}} = \frac{\partial \boldsymbol{h_l}}{\partial \boldsymbol{h_{l-1}}} \cdot \frac{\partial \boldsymbol{h_{l-1}}}{\partial \boldsymbol{h_{l-2}}} \cdots \frac{\partial \boldsymbol{h_2}}{\partial \boldsymbol{h_1}} \cdot \frac{\partial \boldsymbol{h_1}}{\partial \boldsymbol{W_1}}$$

$$\frac{\partial \boldsymbol{h_l}}{\partial \boldsymbol{W_2}} = \frac{\partial \boldsymbol{h_l}}{\partial \boldsymbol{h_{l-1}}} \cdot \frac{\partial \boldsymbol{h_{l-1}}}{\partial \boldsymbol{h_{l-2}}} \cdots \frac{\partial \boldsymbol{h_2}}{\partial \boldsymbol{W_2}}$$

$$\vdots$$

The leftmost term is a product between a vector and a matrix (which is a consequence of having a scalar term in output), which is computationally better than a product between two matrices.

Automatic Differentiation: Reverse Mode AD

$$\boldsymbol{h}_1 = f_1(\boldsymbol{x}, \boldsymbol{W}_1)$$

$$\boldsymbol{h}_2 = f_2(\boldsymbol{h}_1, \boldsymbol{W}_2)$$

$$\vdots$$

$$\boldsymbol{h}_l = f_l(\boldsymbol{h}_{l-1}, \boldsymbol{W}_l)$$

Because matrix multiplication is associative, we can perform the computations in any order. In **Forward-AD**, we proceeded from **the right to the left**, since it corresponds to the ordering in which the primitive functions were executed. However, we can do better by noting two interesting aspects:

What we want to compute:

$$\frac{\partial \boldsymbol{h}_l}{\partial \boldsymbol{W}_1} = \boxed{\frac{\partial \boldsymbol{h}_l}{\partial \boldsymbol{h}_{l-1}} \cdot \frac{\partial \boldsymbol{h}_{l-1}}{\partial \boldsymbol{h}_{l-2}}} \cdots \frac{\partial \boldsymbol{h}_2}{\partial \boldsymbol{h}_1} \cdot \frac{\partial \boldsymbol{h}_1}{\partial \boldsymbol{W}_1}$$

$$\frac{\partial \boldsymbol{h}_l}{\partial \boldsymbol{W}_2} = \boxed{\frac{\partial \boldsymbol{h}_l}{\partial \boldsymbol{h}_{l-1}} \cdot \frac{\partial \boldsymbol{h}_{l-1}}{\partial \boldsymbol{h}_{l-2}}} \cdots \frac{\partial \boldsymbol{h}_2}{\partial \boldsymbol{W}_2}$$

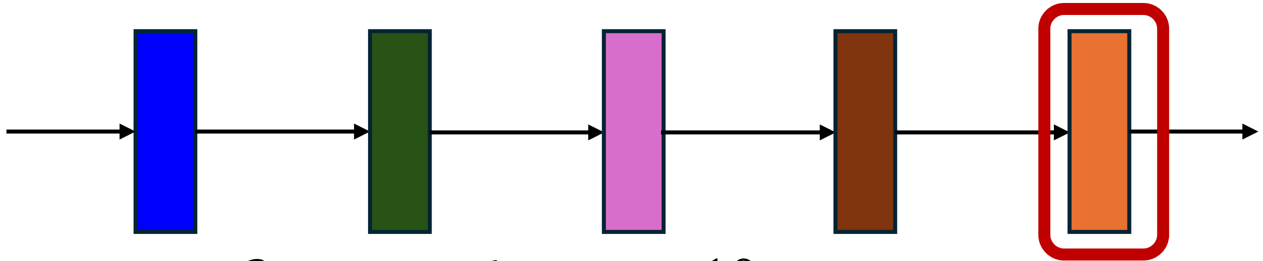
$$\vdots$$

The leftmost term is a product between a vector and a matrix (which is a consequence of having a scalar term in output), which is computationally better than a product between two matrices.

Automatic Differentiation: Reverse Mode AD

$$x = \begin{bmatrix}1\\1\end{bmatrix}$$

$$y = 35$$



$$h_1 = \begin{bmatrix}2\\2\end{bmatrix} \quad h_2 = \begin{bmatrix}6\\4\end{bmatrix} \quad h_3 = \begin{bmatrix}10\\16\end{bmatrix} \quad h_4 = 36 \quad o = 1$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(1)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h}_4} \cdot \frac{\partial \boldsymbol{h}_4}{\partial \boldsymbol{h}_3} \cdot \frac{\partial \boldsymbol{h}_3}{\partial \boldsymbol{h}_2} \cdot \frac{\partial \boldsymbol{h}_2}{\partial \boldsymbol{h}_1} \cdot \frac{\partial \boldsymbol{h}_1}{\partial \boldsymbol{W}^{(1)}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(2)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h}_4} \cdot \frac{\partial \boldsymbol{h}_4}{\partial \boldsymbol{h}_3} \cdot \frac{\partial \boldsymbol{h}_3}{\partial \boldsymbol{h}_2} \cdot \frac{\partial \boldsymbol{h}_2}{\partial \boldsymbol{W}^{(2)}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(3)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h}_4} \cdot \frac{\partial \boldsymbol{h}_4}{\partial \boldsymbol{h}_3} \cdot \frac{\partial \boldsymbol{h}_3}{\partial \boldsymbol{W}^{(3)}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(4)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h}_4} \cdot \frac{\partial \boldsymbol{h}_4}{\partial \boldsymbol{W}^{(4)}}$$

Ini adalah contoh bagian yang cukup dikomputasi sekali lalu disimpan dan digabung untuk komputasi selanjutnya.

Automatic Differentiation: Reverse Mode AD

$$\boldsymbol{h}_1 = f_1(\boldsymbol{x}, \boldsymbol{W}_1)$$

$$\boldsymbol{h}_2 = f_2(\boldsymbol{h}_1, \boldsymbol{W}_2)$$

$$\vdots$$

$$\boldsymbol{h}_{l-1} = f_{l-1}(\boldsymbol{h}_{l-2}, \boldsymbol{W}_{l-1})$$

$$\boldsymbol{L} = l(\boldsymbol{h}_{l-1}, \boldsymbol{W}_l)$$

Step 1: Forward Pass

Differently from F-AD, we start by executing the entire program to be differentiated, **storing all intermediate outputs**.

Step 2: Backward Pass

Initialize a vector $\tilde{\boldsymbol{h}} = \frac{\partial \boldsymbol{L}}{\partial \boldsymbol{h}_{l-1}}$

For index *i* ranging in *l, l-1 , l-2, …, 2, 1*:

$$\partial_{W^{(i)}} \boldsymbol{L} = \tilde{\boldsymbol{h}} \times \frac{\partial \boldsymbol{h}_i}{\partial \boldsymbol{W}^{(i)}}$$

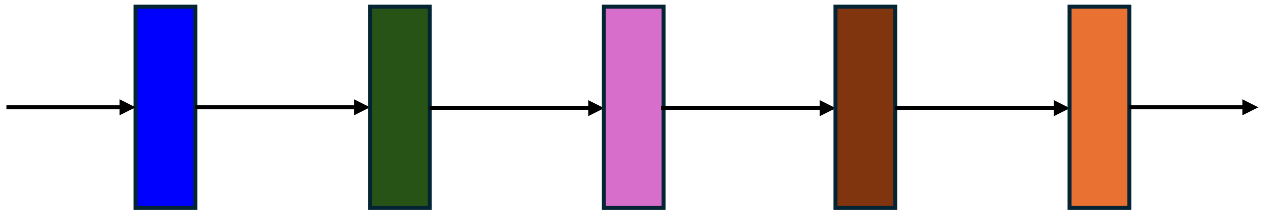
$$\tilde{\boldsymbol{h}} \leftarrow \tilde{\boldsymbol{h}} \times \frac{\partial \boldsymbol{h}_i}{\partial \boldsymbol{h}_{i-1}}$$

We update our "back-propagated" input Jacobian.

Automatic Differentiation: Reverse Mode AD

$x = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

$y = 35$



$h_1 = \begin{bmatrix} 2 \\ 2 \end{bmatrix}$

2x2

2x4

Storage: [ ]

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(1)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h}_4} \cdot \frac{\partial \boldsymbol{h}_4}{\partial \boldsymbol{h}_3} \cdot \frac{\partial \boldsymbol{h}_3}{\partial \boldsymbol{h}_2} \cdot \frac{\partial \boldsymbol{h}_2}{\partial \boldsymbol{h}_1} \cdot \frac{\partial \boldsymbol{h}_1}{\partial \boldsymbol{W}^{(1)}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(2)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h}_4} \cdot \frac{\partial \boldsymbol{h}_4}{\partial \boldsymbol{h}_3} \cdot \frac{\partial \boldsymbol{h}_3}{\partial \boldsymbol{h}_2} \cdot \frac{\partial \boldsymbol{h}_2}{\partial \boldsymbol{W}^{(2)}}$$

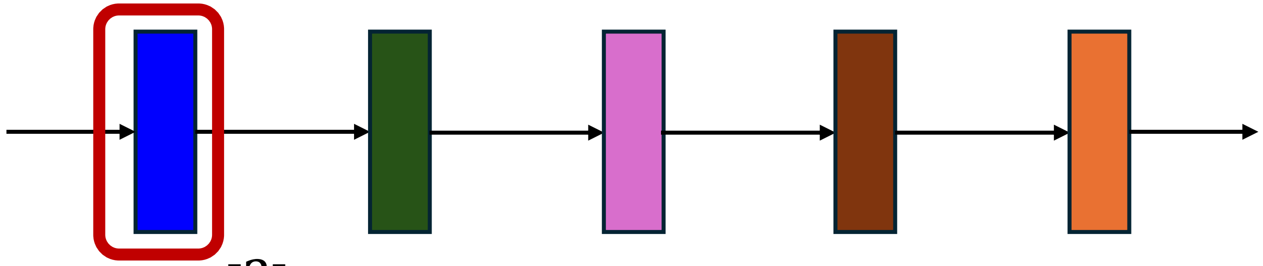
$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(3)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h}_4} \cdot \frac{\partial \boldsymbol{h}_4}{\partial \boldsymbol{h}_3} \cdot \frac{\partial \boldsymbol{h}_3}{\partial \boldsymbol{W}^{(3)}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(4)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h}_4} \cdot \frac{\partial \boldsymbol{h}_4}{\partial \boldsymbol{W}^{(4)}}$$

Automatic Differentiation: Reverse Mode AD

$x = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

$y = 35$



$h_1 = \begin{bmatrix} 2 \\ 2 \end{bmatrix}$

Forward Pass

Storage: $[h_1 = \begin{bmatrix} 2 \\ 2 \end{bmatrix} \quad]$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(1)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{h_2}} \cdot \frac{\partial \boldsymbol{h_2}}{\partial \boldsymbol{h_1}} \cdot \frac{\partial \boldsymbol{h_1}}{\partial \boldsymbol{W}^{(1)}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(2)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{h_2}} \cdot \frac{\partial \boldsymbol{h_2}}{\partial \boldsymbol{W}^{(2)}}$$

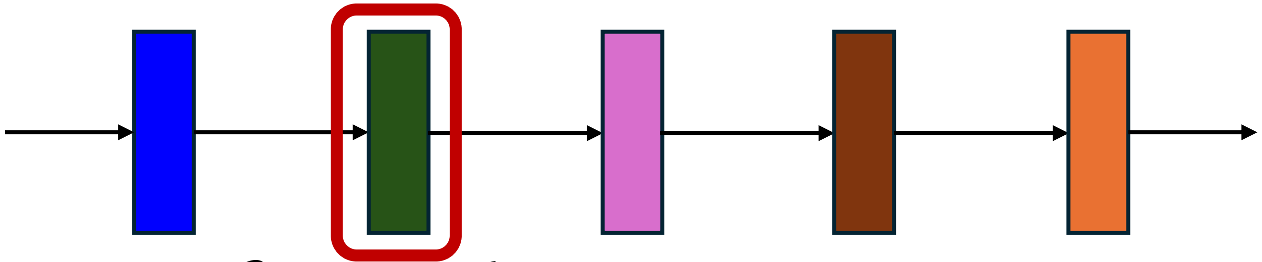
$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(3)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{W}^{(3)}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(4)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{W}^{(4)}}$$

Automatic Differentiation: Reverse Mode AD

$x = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

$y = 35$



$h_1 = \begin{bmatrix} 2 \\ 2 \end{bmatrix}$ $h_2 = \begin{bmatrix} 6 \\ 4 \end{bmatrix}$

Forward Pass

Storage: $[h_1 = \begin{bmatrix} 2 \\ 2 \end{bmatrix} \,|\, h_2 = \begin{bmatrix} 6 \\ 4 \end{bmatrix} \,|\quad|\quad|\quad]$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(1)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{h_2}} \cdot \frac{\partial \boldsymbol{h_2}}{\partial \boldsymbol{h_1}} \cdot \frac{\partial \boldsymbol{h_1}}{\partial \boldsymbol{W}^{(1)}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(2)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{h_2}} \cdot \frac{\partial \boldsymbol{h_2}}{\partial \boldsymbol{W}^{(2)}}$$

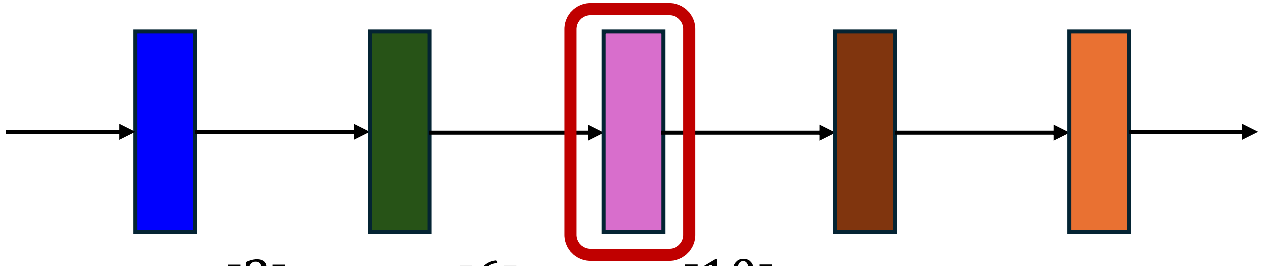
$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(3)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{W}^{(3)}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(4)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{W}^{(4)}}$$

Automatic Differentiation: Reverse Mode AD

$x = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

$y = 35$



$h_1 = \begin{bmatrix} 2 \\ 2 \end{bmatrix}$ $h_2 = \begin{bmatrix} 6 \\ 4 \end{bmatrix}$ $h_3 = \begin{bmatrix} 10 \\ 16 \end{bmatrix}$

Forward Pass

Storage: $\left[h_1 = \begin{bmatrix} 2 \\ 2 \end{bmatrix} \middle| h_2 = \begin{bmatrix} 6 \\ 4 \end{bmatrix} \middle| h_3 = \begin{bmatrix} 10 \\ 16 \end{bmatrix} \middle| \quad \middle| \quad \right]$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(1)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{h_2}} \cdot \frac{\partial \boldsymbol{h_2}}{\partial \boldsymbol{h_1}} \cdot \frac{\partial \boldsymbol{h_1}}{\partial \boldsymbol{W}^{(1)}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(2)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{h_2}} \cdot \frac{\partial \boldsymbol{h_2}}{\partial \boldsymbol{W}^{(2)}}$$

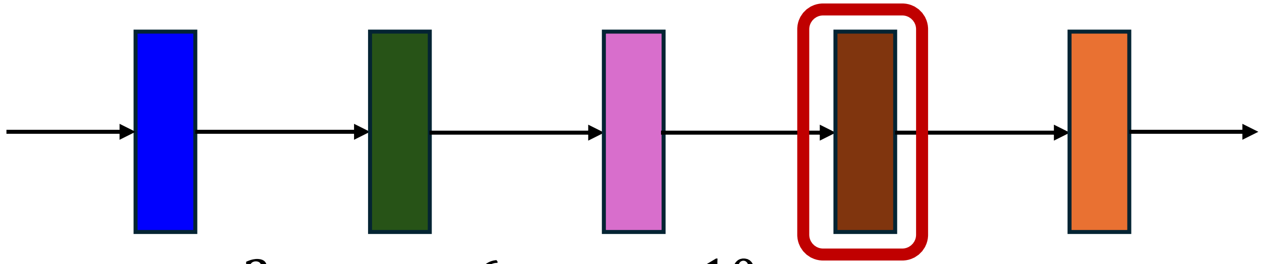
$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(3)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{W}^{(3)}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(4)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{W}^{(4)}}$$

Automatic Differentiation: Reverse Mode AD

$x = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

$y = 35$



Forward Pass

$h_1 = \begin{bmatrix} 2 \\ 2 \end{bmatrix}$ $h_2 = \begin{bmatrix} 6 \\ 4 \end{bmatrix}$ $h_3 = \begin{bmatrix} 10 \\ 16 \end{bmatrix}$ $h_4 = 36$

Storage: $[\, h_1 = \begin{bmatrix} 2 \\ 2 \end{bmatrix} \mid h_2 = \begin{bmatrix} 6 \\ 4 \end{bmatrix} \mid h_3 = \begin{bmatrix} 10 \\ 16 \end{bmatrix} \mid h_4 = 36 \mid \quad]$

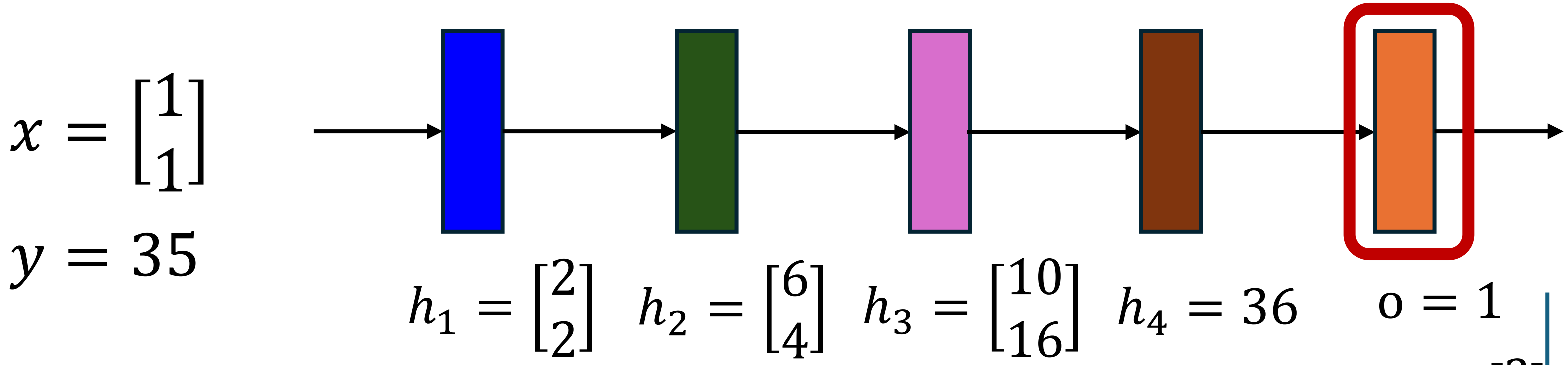
$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(1)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{h_2}} \cdot \frac{\partial \boldsymbol{h_2}}{\partial \boldsymbol{h_1}} \cdot \frac{\partial \boldsymbol{h_1}}{\partial \boldsymbol{W}^{(1)}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(2)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{h_2}} \cdot \frac{\partial \boldsymbol{h_2}}{\partial \boldsymbol{W}^{(2)}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(3)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{W}^{(3)}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(4)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{W}^{(4)}}$$

Automatic Differentiation: Reverse Mode AD



Forward Pass

Storage: $[h_1 = \begin{bmatrix} 2 \\ 2 \end{bmatrix} \,|\, h_2 = \begin{bmatrix} 6 \\ 4 \end{bmatrix} \,|\, h_3 = \begin{bmatrix} 10 \\ 16 \end{bmatrix} \,|\, h_4 = 36 \,|\, o = 1]$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(1)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{h_2}} \cdot \frac{\partial \boldsymbol{h_2}}{\partial \boldsymbol{h_1}} \cdot \frac{\partial \boldsymbol{h_1}}{\partial \boldsymbol{W}^{(1)}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(2)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{h_2}} \cdot \frac{\partial \boldsymbol{h_2}}{\partial \boldsymbol{W}^{(2)}}$$

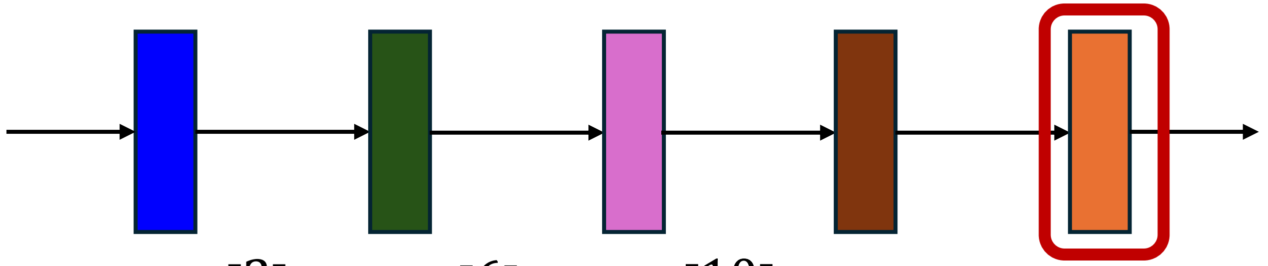
$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(3)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{h_3}} \cdot \frac{\partial \boldsymbol{h_3}}{\partial \boldsymbol{W}^{(3)}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(4)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h_4}} \cdot \frac{\partial \boldsymbol{h_4}}{\partial \boldsymbol{W}^{(4)}}$$

Automatic Differentiation: Reverse Mode AD

$x = \begin{bmatrix}1\\1\end{bmatrix}$

$y = 35$



$h_1 = \begin{bmatrix}2\\2\end{bmatrix}$ $h_2 = \begin{bmatrix}6\\4\end{bmatrix}$ $h_3 = \begin{bmatrix}10\\16\end{bmatrix}$ $h_4 = 36$

Backward Pass

Storage: $\left[h_1 = \begin{bmatrix}2\\2\end{bmatrix} \middle| h_2 = \begin{bmatrix}6\\4\end{bmatrix} \middle| h_3 = \begin{bmatrix}10\\16\end{bmatrix} \middle| h_4 = 36 \middle| \boxed{o = 1} \right]$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(1)}} = \boxed{\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h}_4}} \frac{\partial \boldsymbol{h}_4}{\partial \boldsymbol{h}_3} \cdot \frac{\partial \boldsymbol{h}_3}{\partial \boldsymbol{h}_2} \cdot \frac{\partial \boldsymbol{h}_2}{\partial \boldsymbol{h}_1} \cdot \frac{\partial \boldsymbol{h}_1}{\partial \boldsymbol{W}^{(1)}}$$

$\widetilde{\boldsymbol{h}} \leftarrow \partial \boldsymbol{l} / \partial \boldsymbol{h}_4$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(2)}} = \boxed{\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h}_4}} \frac{\partial \boldsymbol{h}_4}{\partial \boldsymbol{h}_3} \cdot \frac{\partial \boldsymbol{h}_3}{\partial \boldsymbol{h}_2} \cdot \frac{\partial \boldsymbol{h}_2}{\partial \boldsymbol{W}^{(2)}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(3)}} = \boxed{\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h}_4}} \frac{\partial \boldsymbol{h}_4}{\partial \boldsymbol{h}_3} \cdot \frac{\partial \boldsymbol{h}_3}{\partial \boldsymbol{W}^{(3)}}$$

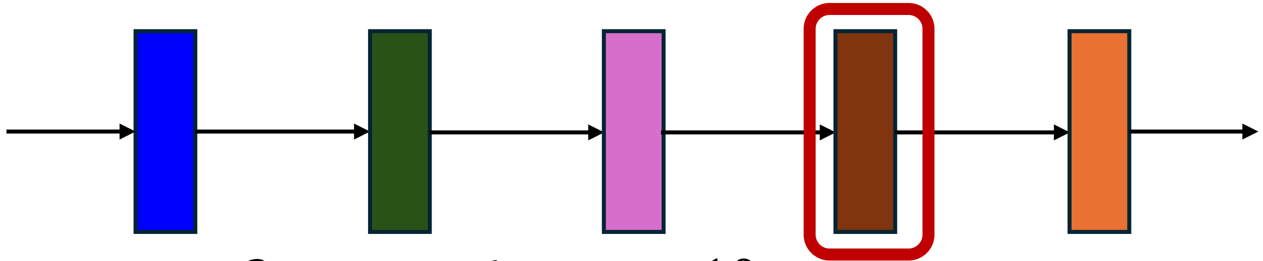
$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(4)}} = \boxed{\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h}_4}} \frac{\partial \boldsymbol{h}_4}{\partial \boldsymbol{W}^{(4)}}$$

Automatic Differentiation: Reverse Mode AD

Backward Pass

$x = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

$y = 35$



$h_1 = \begin{bmatrix} 2 \\ 2 \end{bmatrix}$ $h_2 = \begin{bmatrix} 6 \\ 4 \end{bmatrix}$ $h_3 = \begin{bmatrix} 10 \\ 16 \end{bmatrix}$ $h_4 = 36$ $o = 1$

Storage: $\left[h_1 = \begin{bmatrix} 2 \\ 2 \end{bmatrix} \middle| h_2 = \begin{bmatrix} 6 \\ 4 \end{bmatrix} \middle| h_3 = \begin{bmatrix} 10 \\ 16 \end{bmatrix} \middle| \boxed{h_4 = 36} \middle| o = 1 \right]$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(1)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h}_4} \frac{\partial \boldsymbol{h}_4}{\partial \boldsymbol{h}_3} \frac{\partial \boldsymbol{h}_3}{\partial \boldsymbol{h}_2} \cdot \frac{\partial \boldsymbol{h}_2}{\partial \boldsymbol{h}_1} \cdot \frac{\partial \boldsymbol{h}_1}{\partial \boldsymbol{W}^{(1)}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(2)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h}_4} \frac{\partial \boldsymbol{h}_4}{\partial \boldsymbol{h}_3} \frac{\partial \boldsymbol{h}_3}{\partial \boldsymbol{h}_2} \cdot \frac{\partial \boldsymbol{h}_2}{\partial \boldsymbol{W}^{(2)}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(3)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h}_4} \frac{\partial \boldsymbol{h}_4}{\partial \boldsymbol{h}_3} \frac{\partial \boldsymbol{h}_3}{\partial \boldsymbol{W}^{(3)}}$$

$$\frac{\partial \boldsymbol{l}}{\partial \boldsymbol{W}^{(4)}} = \frac{\partial \boldsymbol{l}}{\partial \boldsymbol{h}_4} \frac{\partial \boldsymbol{h}_4}{\partial \boldsymbol{W}^{(4)}}$$

DONE!

$\tilde{\boldsymbol{h}} \leftarrow \partial \boldsymbol{l} / \partial \boldsymbol{h}_4$

$\partial_{\boldsymbol{W}^{(4)}} \boldsymbol{l} = \tilde{\boldsymbol{h}} \times \partial \boldsymbol{h}_4 / \partial \boldsymbol{W}^{(4)}$

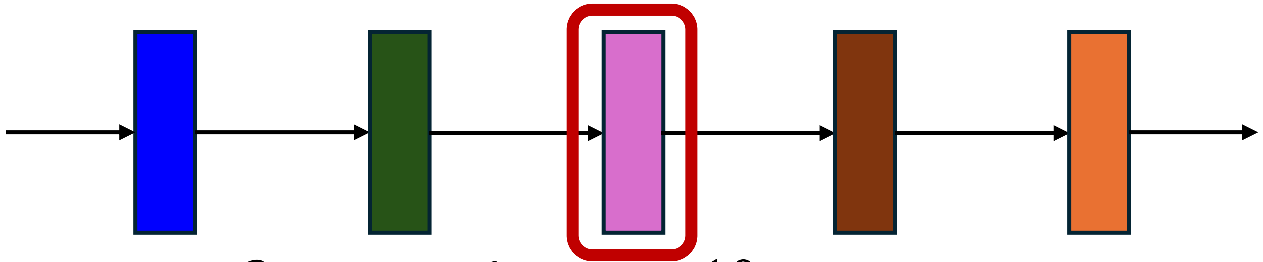
$\tilde{\boldsymbol{h}} \leftarrow \tilde{\boldsymbol{h}} \times \partial \boldsymbol{h}_4 / \partial \boldsymbol{h}_3$

Automatic Differentiation: Reverse Mode AD

Backward Pass

$x = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

$y = 35$



$h_1 = \begin{bmatrix} 2 \\ 2 \end{bmatrix}$ $h_2 = \begin{bmatrix} 6 \\ 4 \end{bmatrix}$ $h_3 = \begin{bmatrix} 10 \\ 16 \end{bmatrix}$ $h_4 = 36$ $o = 1$

Storage: $\left[h_1 = \begin{bmatrix} 2 \\ 2 \end{bmatrix} \middle| h_2 = \begin{bmatrix} 6 \\ 4 \end{bmatrix} \middle| \boxed{h_3 = \begin{bmatrix} 10 \\ 16 \end{bmatrix}} \middle| h_4 = 36 \middle| o = 1 \right]$

$$\frac{\partial l}{\partial W^{(1)}} = \frac{\partial l}{\partial h_4} \frac{\partial h_4}{\partial h_3} \frac{\partial h_3}{\partial h_2} \cdot \frac{\partial h_2}{\partial h_1} \cdot \frac{\partial h_1}{\partial W^{(1)}}$$

$$\frac{\partial l}{\partial W^{(2)}} = \frac{\partial l}{\partial h_4} \frac{\partial h_4}{\partial h_3} \frac{\partial h_3}{\partial h_2} \cdot \frac{\partial h_2}{\partial W^{(2)}}$$

$$\frac{\partial l}{\partial W^{(3)}} = \frac{\partial l}{\partial h_4} \frac{\partial h_4}{\partial h_3} \frac{\partial h_3}{\partial W^{(3)}}$$ **DONE!**

$$\frac{\partial l}{\partial W^{(4)}} = \frac{\partial l}{\partial h_4} \frac{\partial h_4}{\partial W^{(4)}}$$ **DONE!**

$\tilde{h} \leftarrow \partial l / \partial h_4$

$\partial_{W^{(4)}} l = \tilde{h} \times \partial h_4 / \partial W^{(4)}$

$\tilde{h} \leftarrow \tilde{h} \times \partial h_4 / \partial h_3$

$\partial_{W^{(3)}} l = \tilde{h} \times \partial h_3 / \partial W^{(3)}$

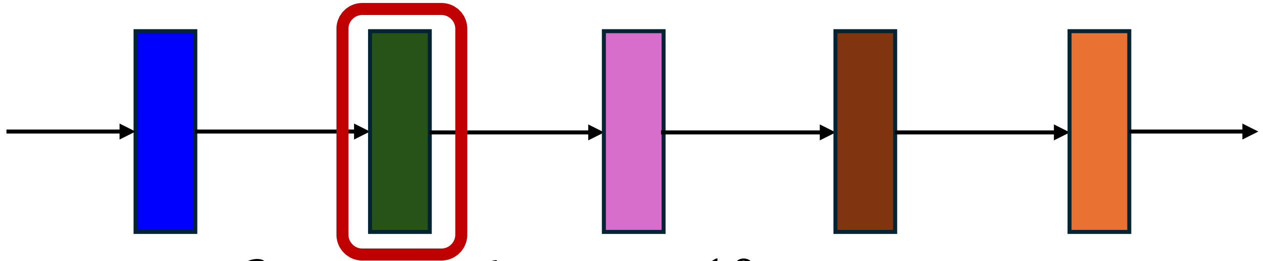
$\tilde{h} \leftarrow \tilde{h} \times \partial h_3 / \partial h_2$

Automatic Differentiation: Reverse Mode AD

Backward Pass

$x = \begin{bmatrix}1\\1\end{bmatrix}$

$y = 35$



$h_1 = \begin{bmatrix}2\\2\end{bmatrix}$ $h_2 = \begin{bmatrix}6\\4\end{bmatrix}$ $h_3 = \begin{bmatrix}10\\16\end{bmatrix}$ $h_4 = 36$ $o = 1$

Storage: $[h_1 = \begin{bmatrix}2\\2\end{bmatrix} \,|\, \boxed{h_2 = \begin{bmatrix}6\\4\end{bmatrix}} \,|\, h_3 = \begin{bmatrix}10\\16\end{bmatrix} \,|\, h_4 = 36 \,|\, o = 1]$

$$\frac{\partial l}{\partial W^{(1)}} = \frac{\partial l}{\partial h_4} \cdot \frac{\partial h_4}{\partial h_3} \cdot \frac{\partial h_3}{\partial h_2} \cdot \boxed{\frac{\partial h_2}{\partial h_1}} \cdot \frac{\partial h_1}{\partial W^{(1)}}$$

$$\frac{\partial l}{\partial W^{(2)}} = \frac{\partial l}{\partial h_4} \cdot \frac{\partial h_4}{\partial h_3} \cdot \frac{\partial h_3}{\partial h_2} \cdot \frac{\partial h_2}{\partial W^{(2)}}$$ DONE!

$$\frac{\partial l}{\partial W^{(3)}} = \frac{\partial l}{\partial h_4} \cdot \frac{\partial h_4}{\partial h_3} \cdot \frac{\partial h_3}{\partial W^{(3)}}$$ DONE!

$$\frac{\partial l}{\partial W^{(4)}} = \frac{\partial l}{\partial h_4} \cdot \frac{\partial h_4}{\partial W^{(4)}}$$ DONE!

$\tilde{h} \leftarrow \partial l / \partial h_4$

$\partial_{W^{(4)}} l = \tilde{h} \times \partial h_4 / \partial W^{(4)}$

$\tilde{h} \leftarrow \tilde{h} \times \partial h_4 / \partial h_3$

$\partial_{W^{(3)}} l = \tilde{h} \times \partial h_3 / \partial W^{(3)}$

$\tilde{h} \leftarrow \tilde{h} \times \partial h_3 / \partial h_2$

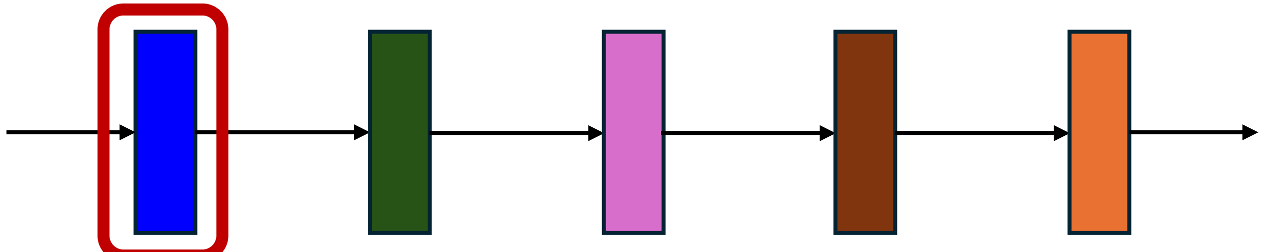
$\partial_{W^{(2)}} l = \tilde{h} \times \partial h_2 / \partial W^{(2)}$

$\tilde{h} \leftarrow \tilde{h} \times \partial h_2 / \partial h_1$

Automatic Differentiation: Reverse Mode AD

$x = \begin{bmatrix}1\\1\end{bmatrix}$

$y = 35$



$h_1 = \begin{bmatrix}2\\2\end{bmatrix}$ $h_2 = \begin{bmatrix}6\\4\end{bmatrix}$ $h_3 = \begin{bmatrix}10\\16\end{bmatrix}$ $h_4 = 36$ $o = 1$

Backward Pass

Storage: $\left[\boxed{h_1 = \begin{bmatrix}2\\2\end{bmatrix}} \middle| h_2 = \begin{bmatrix}6\\4\end{bmatrix} \middle| h_3 = \begin{bmatrix}10\\16\end{bmatrix} \middle| h_4 = 36 \middle| o = 1\right]$

$$\frac{\partial l}{\partial W^{(1)}} = \frac{\partial l}{\partial h_4} \frac{\partial h_4}{\partial h_3} \frac{\partial h_3}{\partial h_2} \boxed{\frac{\partial h_2}{\partial h_1}} \frac{\partial h_1}{\partial W^{(1)}}$$

DONE!

$$\frac{\partial l}{\partial W^{(2)}} = \frac{\partial l}{\partial h_4} \frac{\partial h_4}{\partial h_3} \frac{\partial h_3}{\partial h_2} \frac{\partial h_2}{\partial W^{(2)}}$$

DONE!

$$\frac{\partial l}{\partial W^{(3)}} = \frac{\partial l}{\partial h_4} \frac{\partial h_4}{\partial h_3} \frac{\partial h_3}{\partial W^{(3)}}$$

DONE!

$$\frac{\partial l}{\partial W^{(4)}} = \frac{\partial l}{\partial h_4} \frac{\partial h_4}{\partial W^{(4)}}$$

DONE!

$\tilde{h} \leftarrow \partial l / \partial h_4$

$\partial_{W^{(4)}} l = \tilde{h} \times \partial h_4 / \partial W^{(4)}$

$\tilde{h} \leftarrow \tilde{h} \times \partial h_4 / \partial h_3$

$\partial_{W^{(3)}} l = \tilde{h} \times \partial h_3 / \partial W^{(3)}$

$\tilde{h} \leftarrow \tilde{h} \times \partial h_3 / \partial h_2$

$\partial_{W^{(2)}} l = \tilde{h} \times \partial h_2 / \partial W^{(2)}$

$\tilde{h} \leftarrow \tilde{h} \times \partial h_2 / \partial h_1$

$\partial_{W^{(1)}} l = \tilde{h} \times \partial h_1 / \partial W^{(1)}$

yang kotak merah hanya dihitung sekali, jadi ktia selalu update h.

ini adalah algoritma dari

loss.backward()

Automatic Differentiation

Computationally, Reverse-Mode AD is significantly more efficient than Forward-Mode AD.

In particular, both operations in **backward steps** at Reverse-Mode AD are **vector-matrix products** scaling only linearly in all shape quantities.

The tradeoff is that executing **Reverse-Mode AD requires a large amount of memory**, since all intermediate values of the primal program must be stored on disk with a suitable strategy

Automatic Differentiation

Specific techniques, such as **gradient checkpointing**, can be used to improve on this tradeoff by increasing computations and partially reducing the memory requirements.

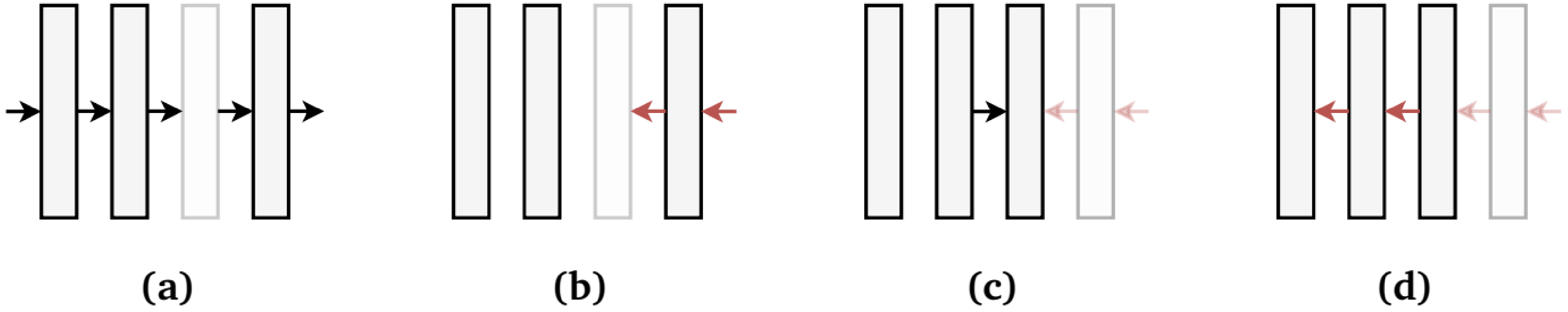


Figure F.6.1: *An example of* ***gradient checkpointing****. (a) We execute a forward pass, but we only store the outputs of the first, second, and fourth blocks (****checkpoints****). (b) The backward pass (red arrows) stops at the third block, whose activations are not available. (c) We run a second forward pass starting from the closest checkpoint to materialize again the activations. (d) We complete the forward pass. Compared to a standard backward pass, this requires 1.25x more computations. In general, the less checkpoints are stored, the higher the computational cost of the backward pass.*

Vector-Jacobian Products

Looking at the backward step in the R-AD algorithm, we can make an interesting observation: the only operation we need is a product between a row vector **v** and a Jacobian of **f** (either the input or the weight Jacobian). We call these two operations the vector-Jacobian products (**VJPs**) of **f(x)**.

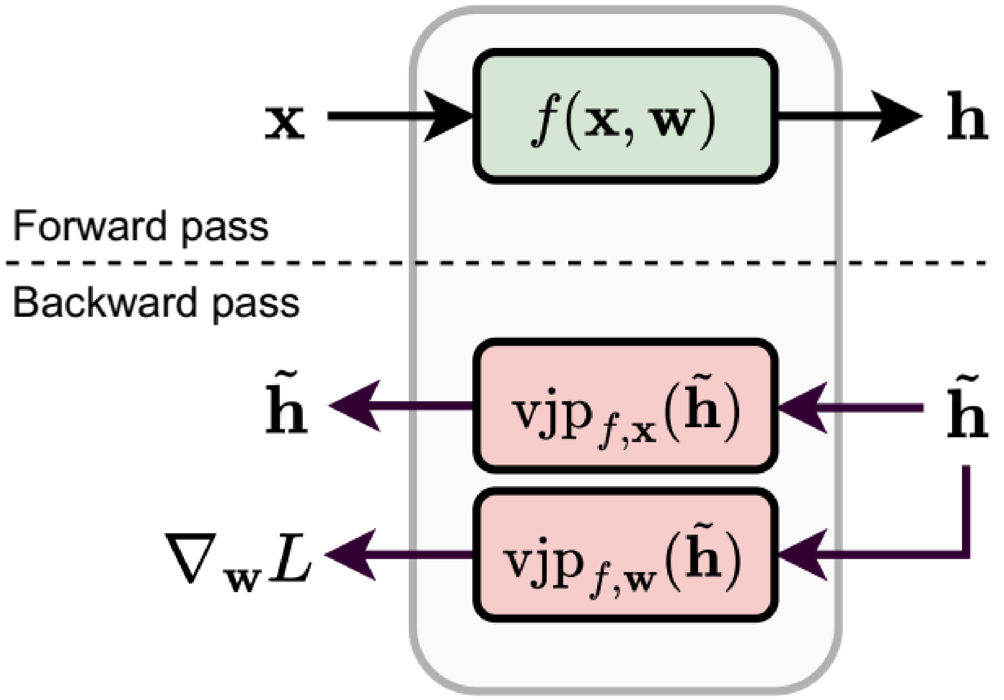
Definition D.6.2 (Vector-Jacobian product (VJP)) *Given a function* $\mathbf{y} = f(\mathbf{x})$*, with* $\mathbf{x} \sim (c)$ *and* $\mathbf{y} \sim (c')$*, its VJP is another function defined as:*

$$\mathrm{vjp}_f(\mathbf{v}) = \mathbf{v}^\top \partial f(\mathbf{x}) \tag{E.6.4}$$

where $\mathbf{v} \sim (c')$*. If* f *has multiple parameters* $f(\mathbf{x}_1, \ldots, \mathbf{x}_n)$*, we can define* n *individual VJPs denoted as* $\mathrm{vjp}_{f,\mathbf{x}_1}(\mathbf{v})$*, ...,* $\mathrm{vjp}_{f,\mathbf{x}_n}(\mathbf{v})$*.*

Vector-Jacobian Products: How to implement R-AD?

Figure F.6.2: *For performing R-AD, primitives must be augmented with two VJP operations to be able to perform a backward pass, corresponding to the input VJP* (E.6.5) *and the weight VJP* (E.6.6). *One call for each is sufficient to perform the backward pass through the primitive, corresponding to* (E.6.7)-(E.6.8).



$$\mathrm{vjp}_{f,\mathbf{x}}(\mathbf{v}) = \mathbf{v}^\top \partial_{\mathbf{x}} f(\mathbf{x}, \mathbf{w})$$
$$\mathrm{vjp}_{f,\mathbf{w}}(\mathbf{v}) = \mathbf{v}^\top \partial_{\mathbf{w}} f(\mathbf{x}, \mathbf{w})$$

Contoh implementasi sederhana Reverse-Mode AD

Contoh kode dari A. Karpathy
https://github.com/karpathy/micrograd

$$f(a,b) = a^2 + 2ab + b^2$$

$$g(a,b) = a + b$$

$$h(g) = g^2$$

$$f(a,b) = h(g(a,b))$$

$$\frac{\partial f(a,b)}{\partial a} = 2a + 2b$$

$$\frac{\partial f(a,b)}{\partial b} = 2a + 2b$$

$$\frac{\partial f(a,b)}{\partial a} = \frac{\partial h}{\partial a} = \frac{\partial h}{\partial g} . \frac{\partial g}{\partial a}$$

$$\frac{\partial f(a,b)}{\partial b} = \frac{\partial h}{\partial b} = \frac{\partial h}{\partial g} . \frac{\partial g}{\partial b}$$

$$f(a,b) = a^2 + 2ab + b^2$$

```
a = Value(...)
b = Value(...)
c = a + b
d = c ** 2
d.backward()
print(a.grad)    # ∂d/∂a
print(b.grad)    # ∂d/∂b
```

https://github.com/karpathy/micrograd

```python
class Value:
    """ stores a single scalar value and its gradient """

    def __init__(self, data, _children=()):
        self.data = data
        self.grad = 0
        # internal variables used for autograd graph construction
        self._backward = lambda: None
        self._prev = set(_children)

    def __add__(self, other):
        other = other if isinstance(other, Value) else Value(other)
        out = Value(self.data + other.data, (self, other), '+')

        def _backward():
            self.grad += out.grad
            other.grad += out.grad
        out._backward = _backward

        return out
```

https://github.com/karpathy/micrograd

```python
def __mul__(self, other):
    other = other if isinstance(other, Value) else Value(other)
    out = Value(self.data * other.data, (self, other))

    def _backward():
        self.grad += other.data * out.grad
        other.grad += self.data * out.grad
    out._backward = _backward

    return out

def __pow__(self, other):
    assert isinstance(other, (int, float)), "only supporting int/float powers for now"
    out = Value(self.data**other, (self,))

    def _backward():
        self.grad += (other * self.data**(other-1)) * out.grad
    out._backward = _backward

    return out
```

https://github.com/karpathy/micrograd

```python
def backward(self):

    # topological order all of the children in the graph
    topo = []
    visited = set()
    def build_topo(v):
        if v not in visited:
            visited.add(v)
            for child in v._prev:
                build_topo(child)
            topo.append(v)
    build_topo(self)

    # go one variable at a time and apply the chain rule to get its gradient
    self.grad = 1
    for v in reversed(topo):
        v._backward()
```

$$f(a,b) = a^2 + 2ab + b^2$$

```
a = Value(2)
b = Value(3)
c = a + b
d = c ** 2
d.backward()
print(a.grad)    # ∂d/∂a
print(b.grad)    # ∂d/∂b
```

$$f(a,b) = a^2 + 2ab + b^2$$

```
a = Value(2)
b = Value(3)
c = a + b
d = c ** 2
d.backward()
print(a.grad)    # ∂d/∂a
print(b.grad)    # ∂d/∂b
```

a

```
_prev = ()
data = 2
grad = 0
_backward =
    lambda: None
```

$$f(a,b) = a^2 + 2ab + b^2$$

```
a = Value(2)
b = Value(3)
c = a + b
d = c ** 2
d.backward()
print(a.grad)    # ∂d/∂a
print(b.grad)    # ∂d/∂b
```

a

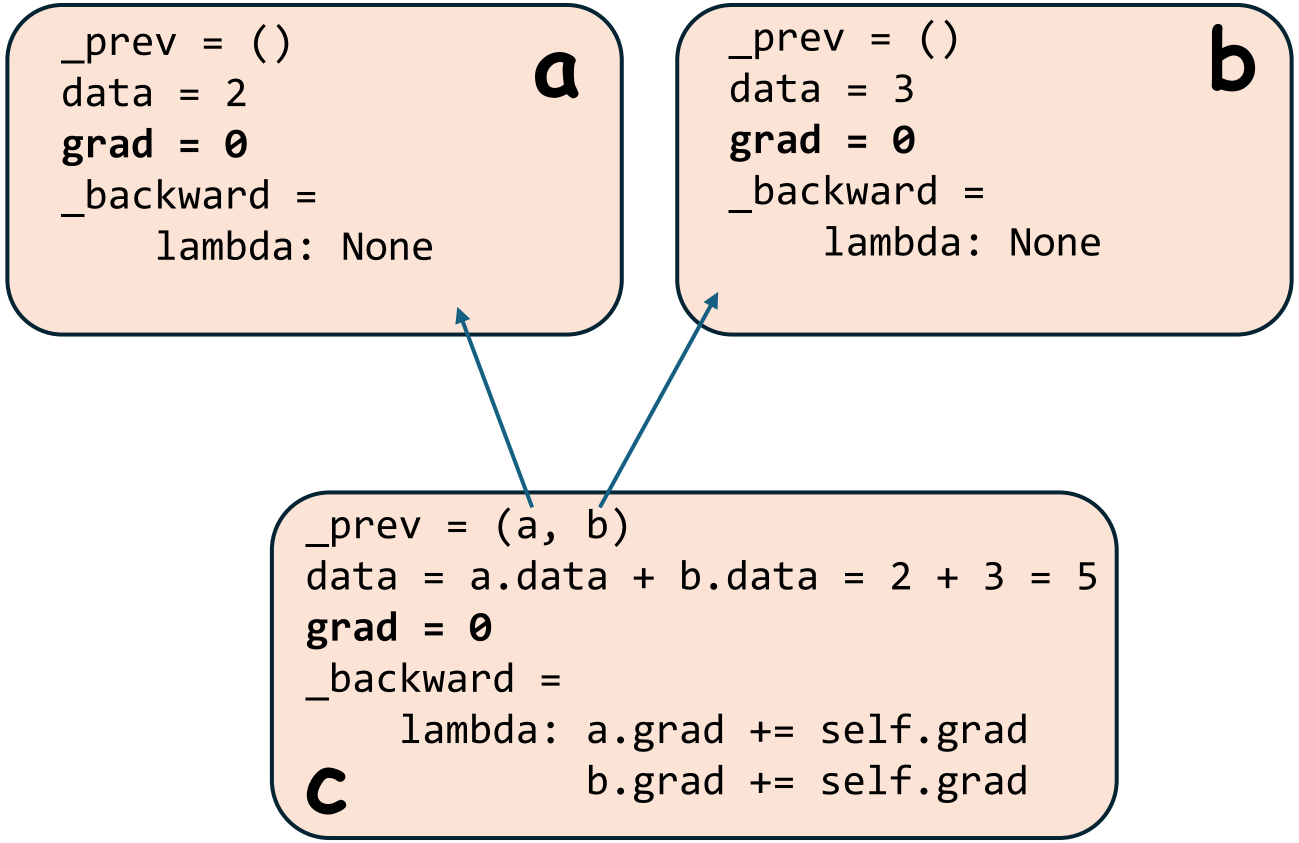
```
_prev = ()
data = 2
grad = 0
_backward =
    lambda: None
```

b

```
_prev = ()
data = 3
grad = 0
_backward =
    lambda: None
```

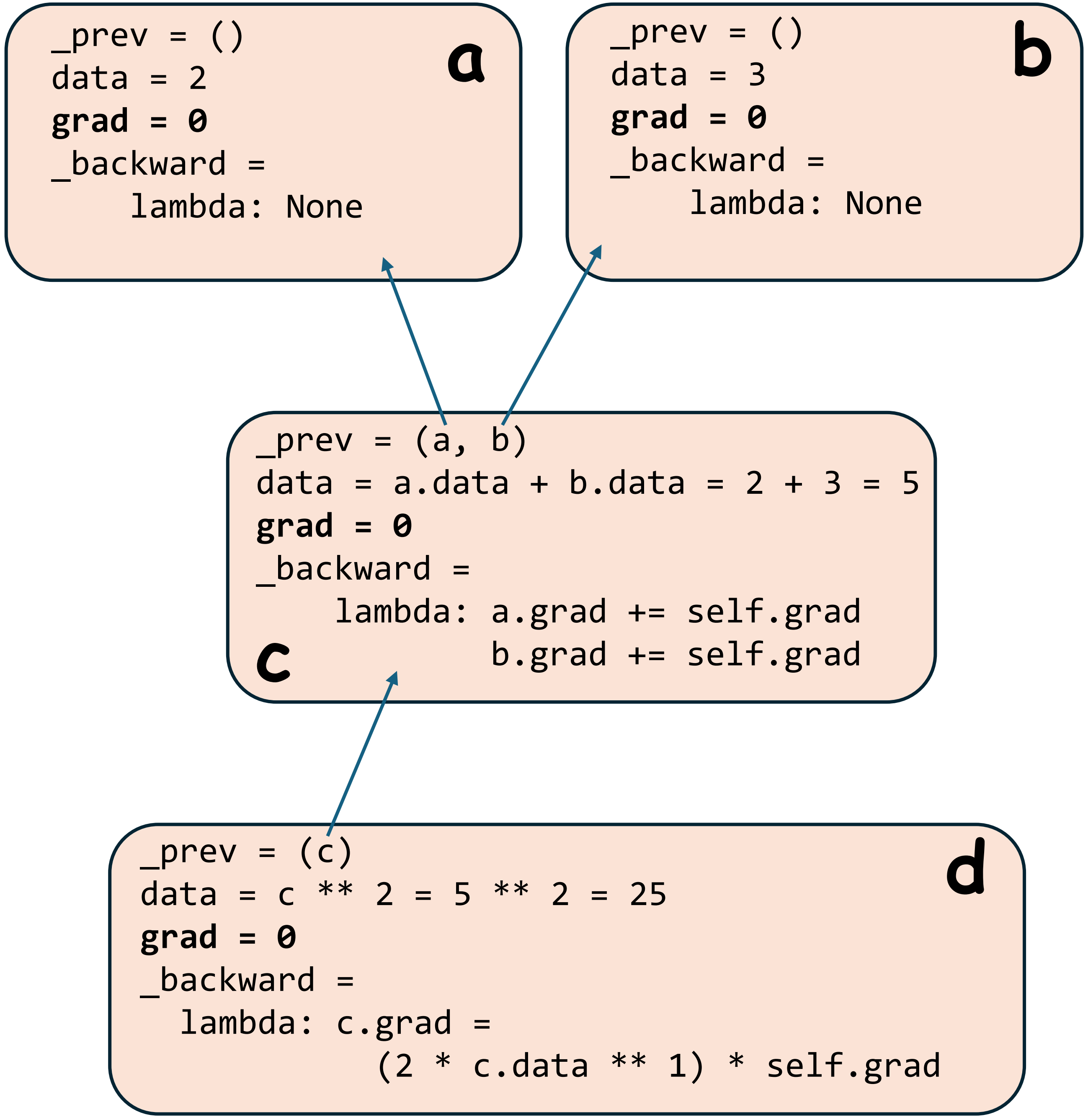
$$f(a,b) = a^2 + 2ab + b^2$$

```
a = Value(2)
b = Value(3)
c = a + b
d = c ** 2
d.backward()
print(a.grad)    # ∂d/∂a
print(b.grad)    # ∂d/∂b
```



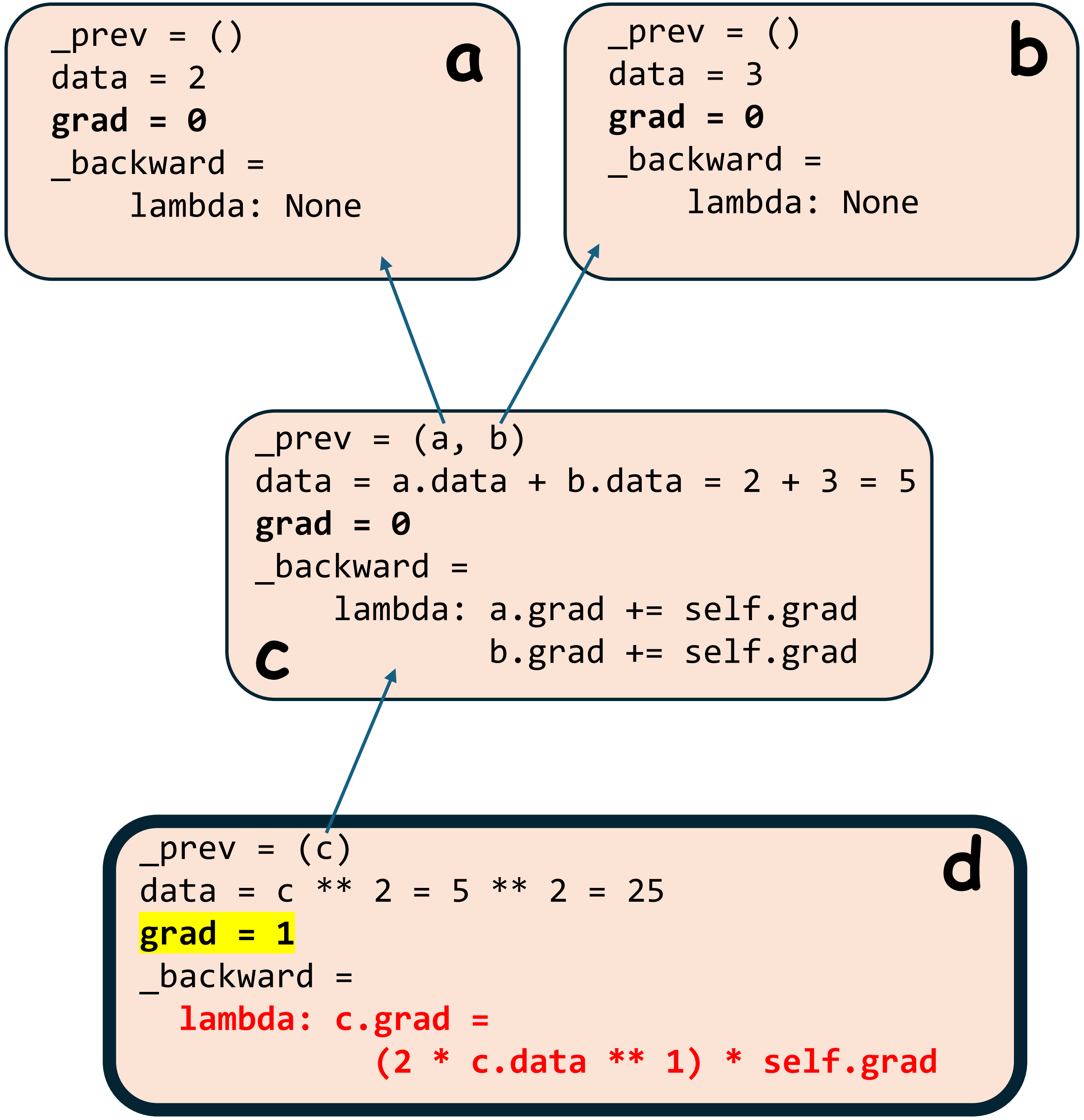
$$f(a,b) = a^2 + 2ab + b^2$$

```
a = Value(2)
b = Value(3)
c = a + b
d = c ** 2
d.backward()
print(a.grad)    # ∂d/∂a
print(b.grad)    # ∂d/∂b
```



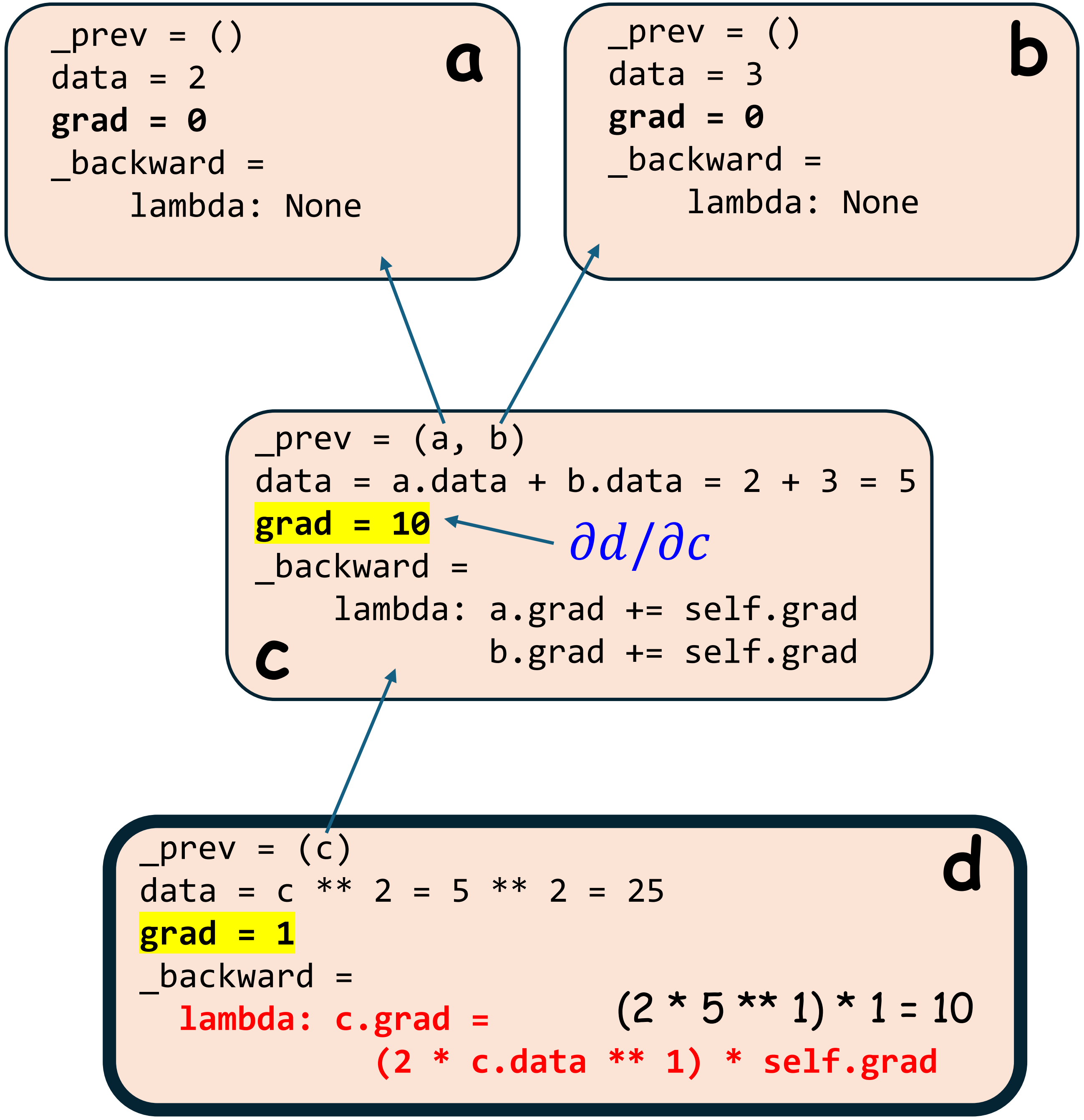
$$f(a, b) = a^2 + 2ab + b^2$$

```
a = Value(2)
b = Value(3)
c = a + b
d = c ** 2
d.backward()
print(a.grad)    # ∂d/∂a
print(b.grad)    # ∂d/∂b
```



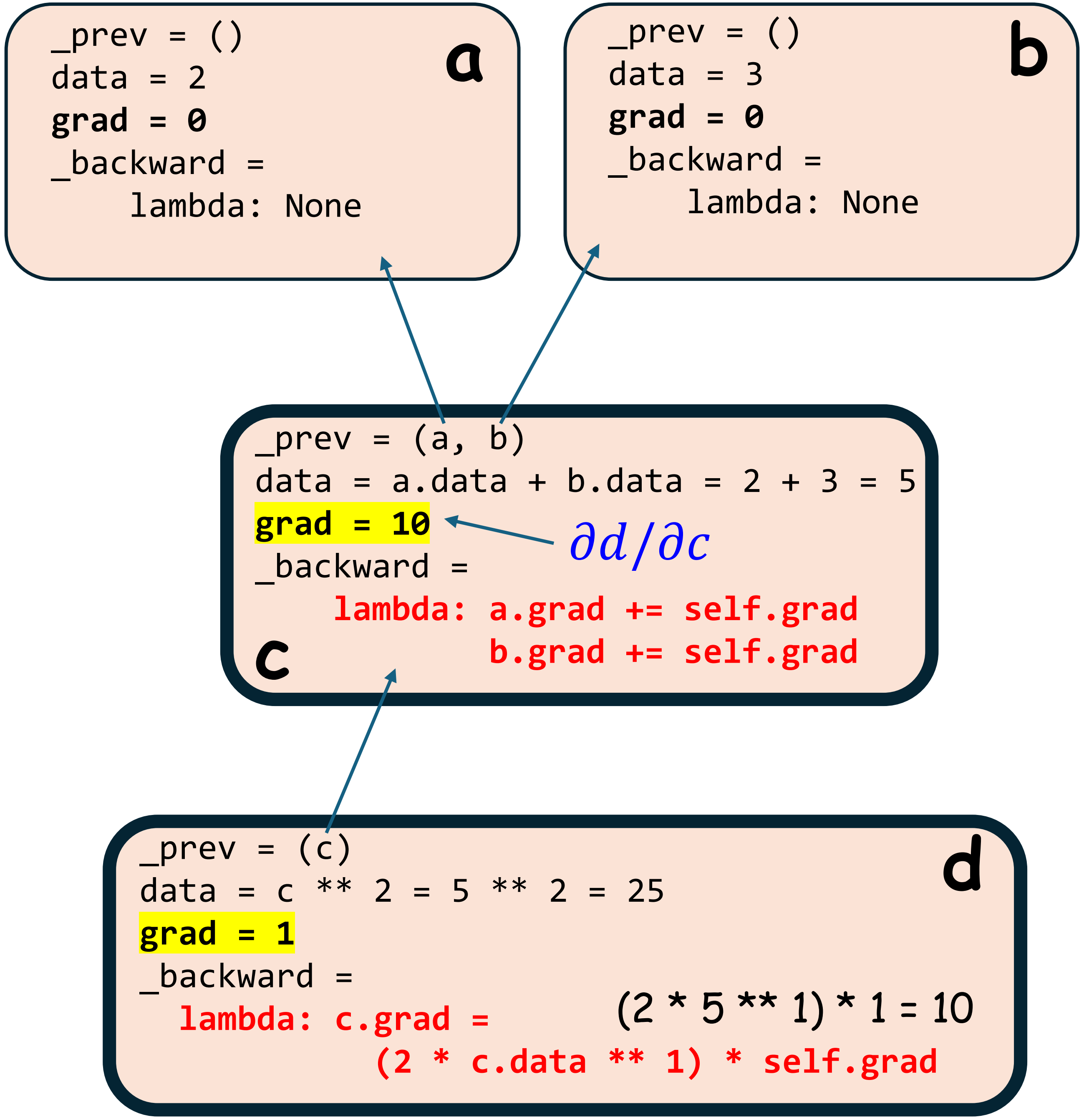
$$f(a,b) = a^2 + 2ab + b^2$$

```
a = Value(2)
b = Value(3)
c = a + b
d = c ** 2
d.backward()
print(a.grad)    # ∂d/∂a
print(b.grad)    # ∂d/∂b
```



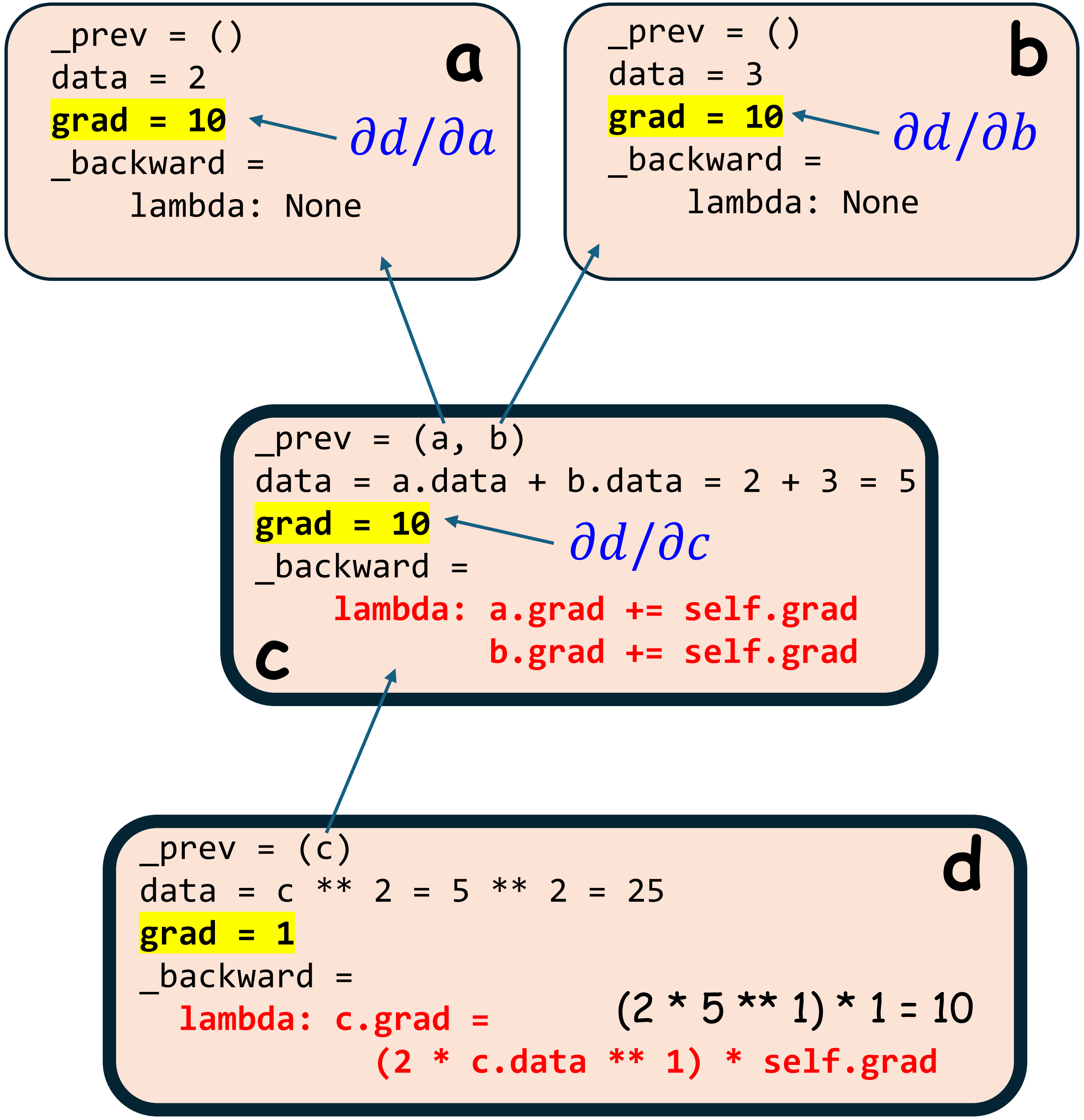
$$f(a,b) = a^2 + 2ab + b^2$$

```
a = Value(2)
b = Value(3)
c = a + b
d = c ** 2
d.backward()
print(a.grad)    # ∂d/∂a
print(b.grad)    # ∂d/∂b
```



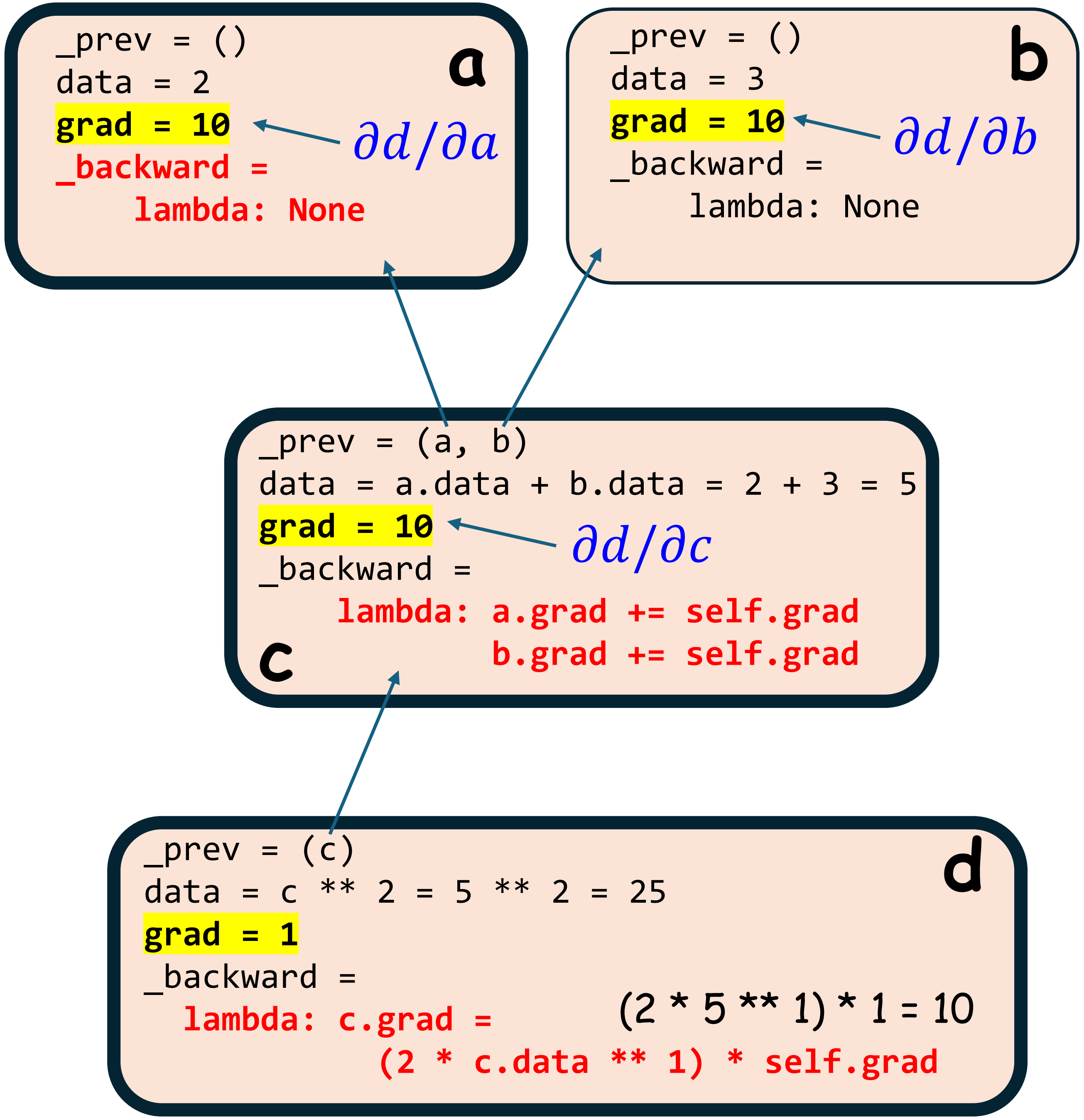
$$f(a, b) = a^2 + 2ab + b^2$$

```
a = Value(2)
b = Value(3)
c = a + b
d = c ** 2
d.backward()
print(a.grad)    # ∂d/∂a
print(b.grad)    # ∂d/∂b
```



$$f(a,b) = a^2 + 2ab + b^2$$

```
a = Value(2)
b = Value(3)
c = a + b
d = c ** 2
d.backward()
print(a.grad)     # ∂d/∂a
print(b.grad)     # ∂d/∂b
```



$$f(a,b) = a^2 + 2ab + b^2$$

```python
a = Value(2)
b = Value(3)
c = a + b
d = c ** 2
d.backward()
print(a.grad)    # ∂d/∂a
print(b.grad)    # ∂d/∂b
```