Calculus for Artificial Intelligence

1. Power Rule

For a function $f(x, y) = x^n$ or $f(x, y) = y^n$:

$$\frac{\partial}{\partial x}(x^n) = nx^{n-1}$$
$$\frac{\partial}{\partial y}(y^n) = ny^{n-1}$$

Example:

$$f(x, y) = x^3$$
$$\frac{\partial f}{\partial x} = 3x^2$$

$$g(x, y) = y^4$$
$$\frac{\partial g}{\partial y} = 4y^3$$

2. Constant Multiple Rule

For a function $f(x, y) = c \cdot g(x, y)$, where c is a constant:

$$\frac{\partial}{\partial x}(c \cdot g(x, y)) = c \cdot \frac{\partial g(x,y)}{\partial x}$$
$$\frac{\partial}{\partial y}(c \cdot g(x, y)) = c \cdot \frac{\partial g(x,y)}{\partial y}$$

Example:

$$f(x, y) = 5x^2$$
$$\frac{\partial f}{\partial x} = 5 \cdot 2x = 10x$$
$$g(x, y) = 7y^3$$
$$\frac{\partial g}{\partial y} = 7 \cdot 3y^2 = 21y^2$$

3. Sum Rule

For a function $f(x, y) = g(x, y) + h(x, y)$:

$$\frac{\partial}{\partial x}(g(x,y) + h(x,y)) = \frac{\partial g(x,y)}{\partial x} + \frac{\partial h(x,y)}{\partial x}$$

$$\frac{\partial}{\partial y}(g(x,y) + h(x,y)) = \frac{\partial g(x,y)}{\partial y} + \frac{\partial h(x,y)}{\partial y}$$

Example:

$$f(x, y) = x^2 + y^2$$

$$\frac{\partial f}{\partial x} = \frac{\partial}{\partial x}(x^2) + \frac{\partial}{\partial x}(y^2) = 2x + 0 = 2x$$

$$g(x, y) = 3x^3 + 4y$$

$$\frac{\partial g}{\partial y} = \frac{\partial}{\partial y}(3x^3) + \frac{\partial}{\partial y}(4y) = 0 + 4 = 4$$

4. Product Rule

For a function $f(x, y) = g(x, y) \cdot h(x, y)$:

$$\frac{\partial}{\partial x}(g(x, y) \cdot h(x, y)) = g(x, y) \cdot \frac{\partial h(x,y)}{\partial x} + h(x, y) \cdot \frac{\partial g(x,y)}{\partial x}$$

$$\frac{\partial}{\partial y}(g(x, y) \cdot h(x, y)) = g(x, y) \cdot \frac{\partial h(x,y)}{\partial y} + h(x, y) \cdot \frac{\partial g(x,y)}{\partial y}$$

Simplified,

$$\frac{\partial f}{\partial x} = u \cdot \frac{\partial v}{\partial x} + v \cdot \frac{\partial u}{\partial x}$$

$$\frac{\partial f}{\partial x} = u \cdot \left(\frac{\partial}{\partial x} v\right) + v \cdot \left(\frac{\partial}{\partial x} u\right)$$

Example:

$$f(x, y) = (x^2) \cdot (y^3)$$

$$\frac{\partial f}{\partial x} = (x^2) \cdot \frac{\partial (y^3)}{\partial x} + (y^3) \cdot \frac{\partial (x^2)}{\partial x} = (x^2) \cdot 0 + (y^3) \cdot (2x) = 2xy^3$$

$$g(x, y) = (3x) \cdot (4y^2)$$

$$\frac{\partial g}{\partial y} = (3x) \cdot \frac{\partial (4y^2)}{\partial y} + (4y^2) \cdot \frac{\partial (3x)}{\partial y} = (3x) \cdot (8y) + (4y^2) \cdot 0 = 24xy$$

Partial Derivative with Respect to x

First, we compute the partial derivatives of $u(x, y) = x^2 + y^2$ and $v(x, y) = x^2y + y^2x$ with respect to x:

$$u(x, y) = x^2 + y^2$$
$$v(x, y) = x^2y + y^2x$$

$$\frac{\partial u}{\partial x} = 2x$$
$$\frac{\partial v}{\partial x} = \frac{\partial}{\partial x}(x^2y + y^2x) = 2xy + y^2$$

$$\frac{\partial f}{\partial x} = u \cdot \frac{\partial v}{\partial x} + v \cdot \frac{\partial u}{\partial x}$$
$$\frac{\partial f}{\partial x} = (x^2 + y^2)(2xy + y^2) + (x^2y + y^2x)(2x)$$

Simplifying this:

$$\frac{\partial f}{\partial x} = (x^2 + y^2)(2xy + y^2) + 2x(x^2y + y^2x)$$
$$\frac{\partial f}{\partial x} = 2x^3y + x^2y^2 + 2xy^3 + y^4 + 2x^3y + 2xy^3$$
$$\frac{\partial f}{\partial x} = 4x^3y + x^2y^2 + 4xy^3 + y^4$$

Partial Derivative with Respect to y

Now, we compute the partial derivatives of $u(x, y)$ and $v(x, y)$ with respect to y:

$$\frac{\partial u}{\partial y} = 2y$$
$$\frac{\partial v}{\partial y} = \frac{\partial}{\partial y}(x^2y + y^2x) = x^2 + 2yx$$

Applying the product rule:

$$\frac{\partial f}{\partial y} = u \cdot \frac{\partial v}{\partial y} + v \cdot \frac{\partial u}{\partial y}$$
$$\frac{\partial f}{\partial y} = (x^2 + y^2)(x^2 + 2yx) + (x^2y + y^2x)(2y)$$

Simplifying this:

$$\frac{\partial f}{\partial y} = (x^2 + y^2)(x^2 + 2yx) + 2y(x^2y + y^2x)$$
$$\frac{\partial f}{\partial y} = x^4 + 2x^3y + x^2y^2 + 2yx^2 + 2y^2x^2 + 2y^3x$$
$$\frac{\partial f}{\partial y} = x^4 + 2x^3y + x^2y^2 + 2x^2y + 2x^2y^2 + 2y^3x$$
$$\frac{\partial f}{\partial y} = x^4 + 2x^3y + 3x^2y^2 + 2yx^2 + 2y^3x$$

5. Quotient Rule

For a function $f(x, y) = \frac{g(x,y)}{h(x,y)}$:

$$\frac{\partial}{\partial x}\left(\frac{g(x,y)}{h(x,y)}\right) = \frac{h(x,y)\cdot\frac{\partial g(x,y)}{\partial x} - g(x,y)\cdot\frac{\partial h(x,y)}{\partial x}}{[h(x,y)]^2}$$

$$\frac{\partial}{\partial y}\left(\frac{g(x,y)}{h(x,y)}\right) = \frac{h(x,y)\cdot\frac{\partial g(x,y)}{\partial y} - g(x,y)\cdot\frac{\partial h(x,y)}{\partial y}}{[h(x,y)]^2}$$

Example:

$$f(x, y) = \frac{x^2}{y}$$

$$\frac{\partial f}{\partial x} = \frac{y\cdot\frac{\partial(x^2)}{\partial x} - x^2\cdot\frac{\partial y}{\partial x}}{y^2} = \frac{y\cdot(2x) - x^2\cdot 0}{y^2} = \frac{2xy}{y^2} = \frac{2x}{y}$$

$$g(x, y) = \frac{y^2}{x}$$

$$\frac{\partial g}{\partial y} = \frac{x\cdot\frac{\partial(y^2)}{\partial y} - y^2\cdot\frac{\partial x}{\partial y}}{x^2} = \frac{x\cdot(2y) - y^2\cdot 0}{x^2} = \frac{2xy}{x^2} = \frac{2y}{x}$$

Simplified,

$$f(x, y) = \frac{u(x,y)}{v(x,y)}$$

$$\frac{\partial f}{\partial x} = \frac{v\cdot\frac{\partial u}{\partial x} - u\cdot\frac{\partial v}{\partial x}}{v^2}$$

$$\frac{\partial f}{\partial x} = \frac{v\cdot\left(\frac{\partial}{\partial x}u\right) - u\cdot\left(\frac{\partial}{\partial x}v\right)}{v^2}$$

6. Chain Rule

For a function $z = f(g(x,y), h(x,y))$:

$$\frac{\partial z}{\partial x} = \frac{\partial z}{\partial g} \cdot \frac{\partial g}{\partial x} + \frac{\partial z}{\partial h} \cdot \frac{\partial h}{\partial x}$$
$$\frac{\partial z}{\partial y} = \frac{\partial z}{\partial g} \cdot \frac{\partial g}{\partial y} + \frac{\partial z}{\partial h} \cdot \frac{\partial h}{\partial y}$$

Example:

$$f(x,y) = e^{x^2+y^2}$$

Let $u = x^2 + y^2$, then $f = e^u$.

$$\frac{\partial u}{\partial x} = 2x$$
$$\frac{\partial f}{\partial u} = e^u$$
$$\frac{\partial f}{\partial x} = \frac{\partial f}{\partial u} \cdot \frac{\partial u}{\partial x} = e^{x^2+y^2} \cdot 2x = 2xe^{x^2+y^2}$$

Derivative	Integral (Antiderivative)
$\frac{d}{dx}n = 0$	$\int 0\ dx = C$
$\frac{d}{dx}x = 1$	$\int 1\ dx = x + C$
$\frac{d}{dx}x^n = nx^{n-1}$	$\int x^n\ dx = \frac{x^{n+1}}{n+1} + C$
$\frac{d}{dx}e^x = e^x$	$\int e^x\ dx = e^x + C$
$\frac{d}{dx}\ln x = \frac{1}{x}$	$\int \frac{1}{x}\ dx = \ln x + C$
$\frac{d}{dx}n^x = n^x \ln x$	$\int n^x\ dx = \frac{n^x}{\ln n} + C$
$\frac{d}{dx}\sin x = \cos x$	$\int \cos x\ dx = \sin x + C$
$\frac{d}{dx}\cos x = -\sin x$	$\int \sin x\ dx = -\cos x + C$
$\frac{d}{dx}\tan x = \sec^2 x$	$\int \sec^2 x\ dx = \tan x + C$
$\frac{d}{dx}\cot x = -\csc^2 x$	$\int \csc^2 x\ dx = -\cot x + C$
$\frac{d}{dx}\sec x = \sec x \tan x$	$\int \tan x \sec x\ dx = \sec x + C$
$\frac{d}{dx}\csc x = -\csc x \cot x$	$\int \cot x \csc x\ dx = -\csc x + C$
$\frac{d}{dx}\arcsin x = \frac{1}{\sqrt{1-x^2}}$	$\int \frac{1}{\sqrt{1-x^2}}\ dx = \arcsin x + C$
$\frac{d}{dx}\arccos x = -\frac{1}{\sqrt{1-x^2}}$	$\int -\frac{1}{\sqrt{1-x^2}}\ dx = \arccos x + C$
$\frac{d}{dx}\arctan x = \frac{1}{1+x^2}$	$\int \frac{1}{1+x^2}\ dx = \arctan x + C$
$\frac{d}{dx}\operatorname{arc cot} x = -\frac{1}{1+x^2}$	$\int -\frac{1}{1+x^2}\ dx = \operatorname{arc cot} x + C$
$\frac{d}{dx}\operatorname{arc sec} x = \frac{1}{x\sqrt{x^2-1}}$	$\int \frac{1}{x\sqrt{x^2-1}}\ dx = \operatorname{arc sec} x + C$
$\frac{d}{dx}\operatorname{arc csc} x = -\frac{1}{x\sqrt{x^2-1}}$	$\int -\frac{1}{x\sqrt{x^2-1}}\ dx = \operatorname{arc csc} x + C$

Backpropagation in Neural Networks

Backpropagation

Main Steps

Forward propagation and backpropagation are distinct steps within the training process of a neural network, they are closely related and interdependent. But forward propagation is indeed a part of the backpropagation process.

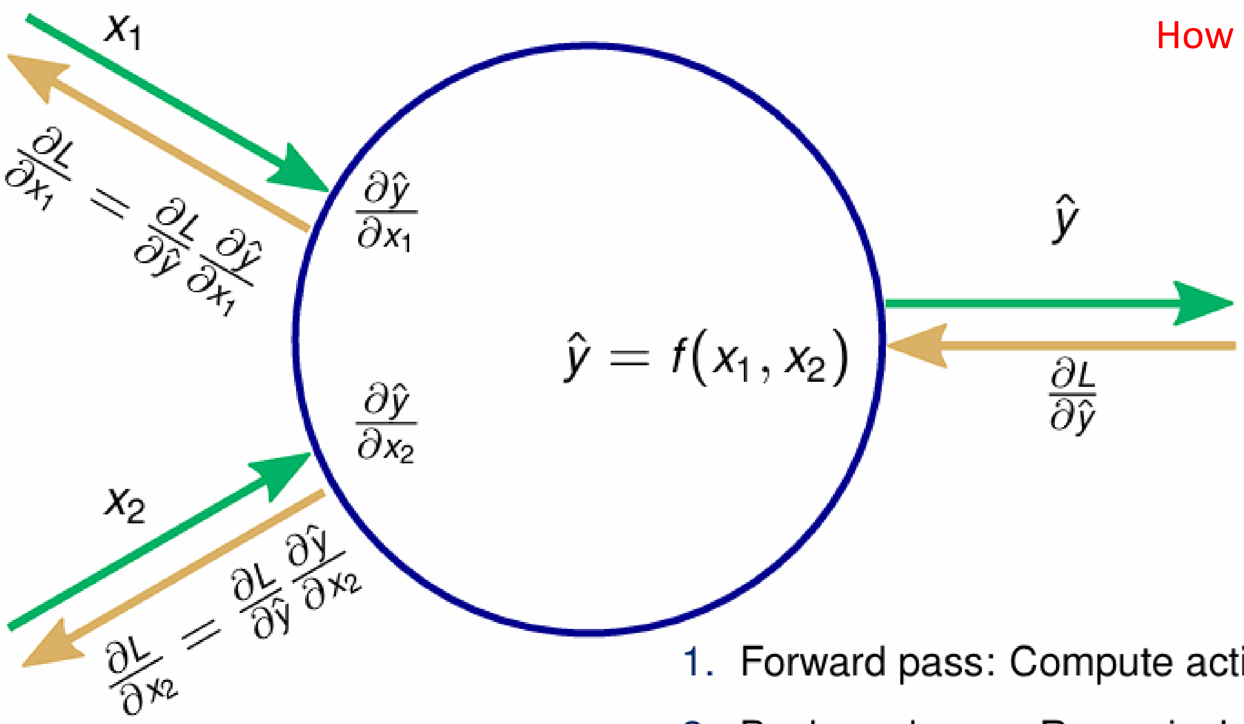
1. **Forward Pass:**
   - Compute the output of the network for a given input.
   - Calculate the cost (loss) using the cost function.

2. **Backward Pass:**
   - Compute the gradient of the loss function concerning the output of the network.
   - Propagate the gradients back through the network to compute the gradients concerning each weight and bias.

3. **Weight Update:**
   - Adjust the weights and biases using the gradient descents.

Backpropagation

The function $\hat{y} = f(x_1, x_2)$ represents the network's output for given inputs



How to calculate derivatives in complex neural networks?

- **Finite Differences:** Numerical approximation method.
- **Analytic Derivative:** Direct computation using calculus.

FD, $$f'(x) \approx \frac{f(x+h) - f(x)}{h}$$

Chain Rule, $$\frac{\partial L}{\partial \theta} = \frac{\partial L}{\partial y} \cdot \frac{\partial y}{\partial \theta}$$

Der. w.r.s to W: $$\frac{\partial L}{\partial w_l} = \frac{\partial L}{\partial z_l} \cdot \frac{\partial z_l}{\partial w_l}$$

$$\frac{\partial L}{\partial w_l} = \frac{\partial L}{\partial a_l} \cdot \frac{\partial a_l}{\partial z_l} \cdot \frac{\partial z_l}{\partial w_l}$$

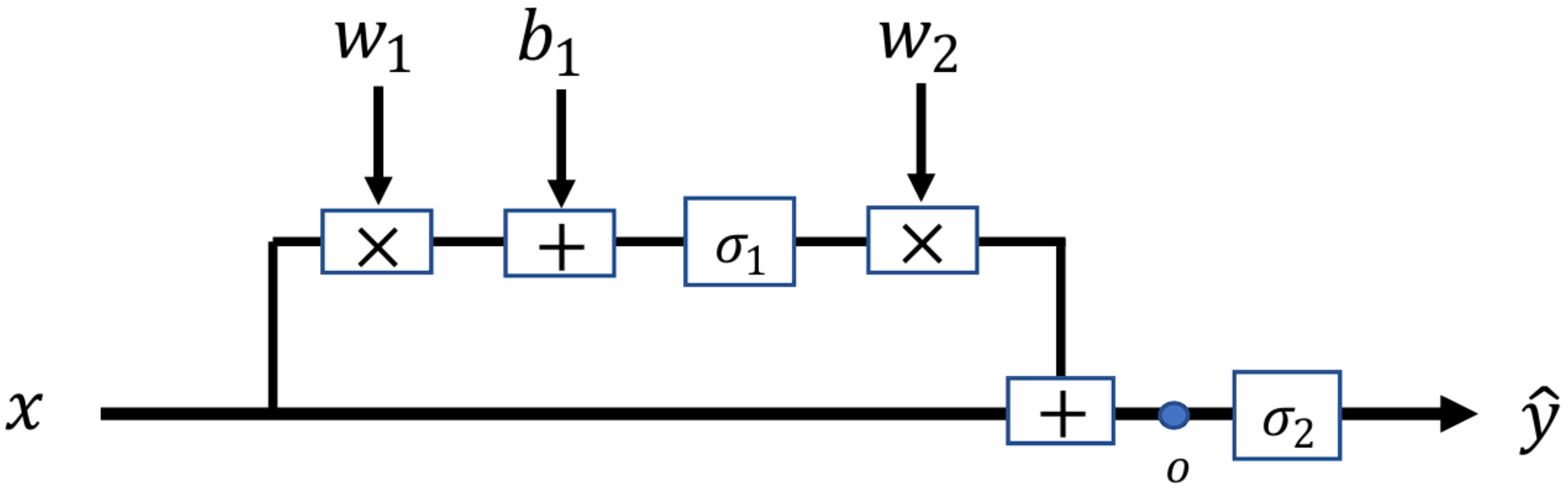
1. Forward pass: Compute activations
2. Backward pass: Recursively apply chain rule

$\frac{\partial L}{\partial \hat{y}}$ represents the gradient of the loss L with respect to the predicted output $\hat{y}$.

This gradient measures how much the loss L changes in response to a small change in the predicted output $\hat{y}$.

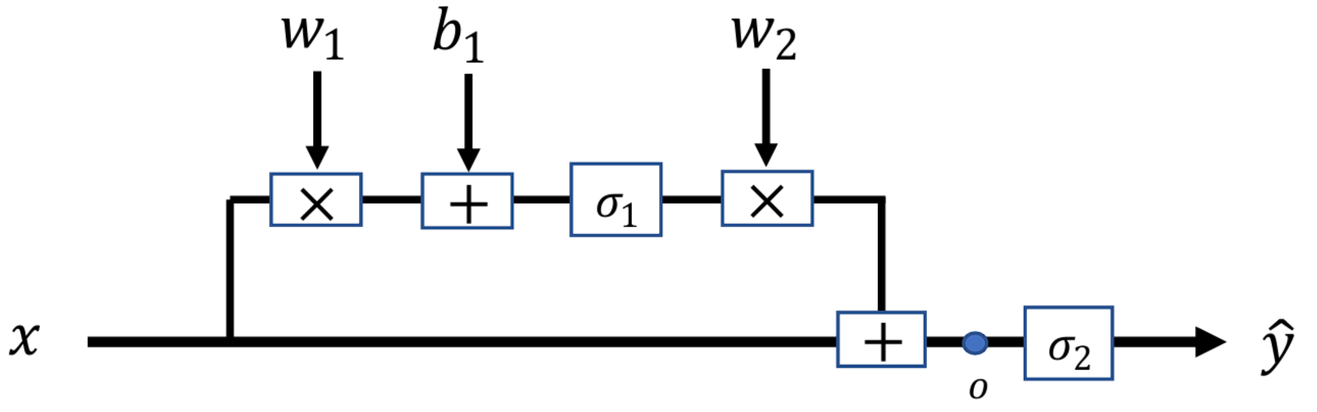
Forward Propagation

Given is the following network $f(x) = \hat{y}$ receiving $x \in \mathbb{R}$ as input to compute a prediction $\hat{y} \in \mathbb{R}$. It uses two weights $w_1, w_2 \in \mathbb{R}$, one bias $b_1 \in \mathbb{R}$ and two sigmoid activations denoted as functions $\sigma_1(\cdot)$ and $\sigma_2(\cdot)$ shown in the figure below. The network is trained using the L_2 norm, defined as $L(y, \hat{y}) = \|\hat{y} - y\|_2^2$ with labels $y \in \{0, 1\}$. The boxes are mathematical operations, while $\times$ represents the multiplication, $+$ the addition and σ the sigmoid activation. $o \in \mathbb{R}$ marks an intermediate result as indicated in the figure.



Forward Propagation

1. **Input to Hidden Layer:**

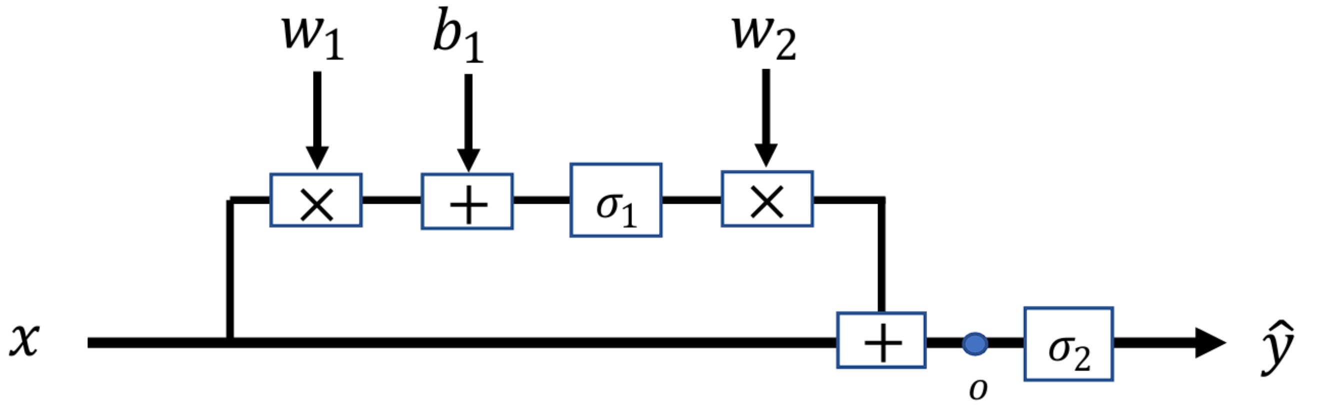


- Input: x
- Weight: w_1
- Bias: b_1
- Activation function: σ_1

The first step involves calculating the intermediate result after the first multiplication and addition:

$$o_1 = \sigma_1(w_1 \cdot x + b_1)$$

Forward Propagation



2. **Hidden Layer to Output:**

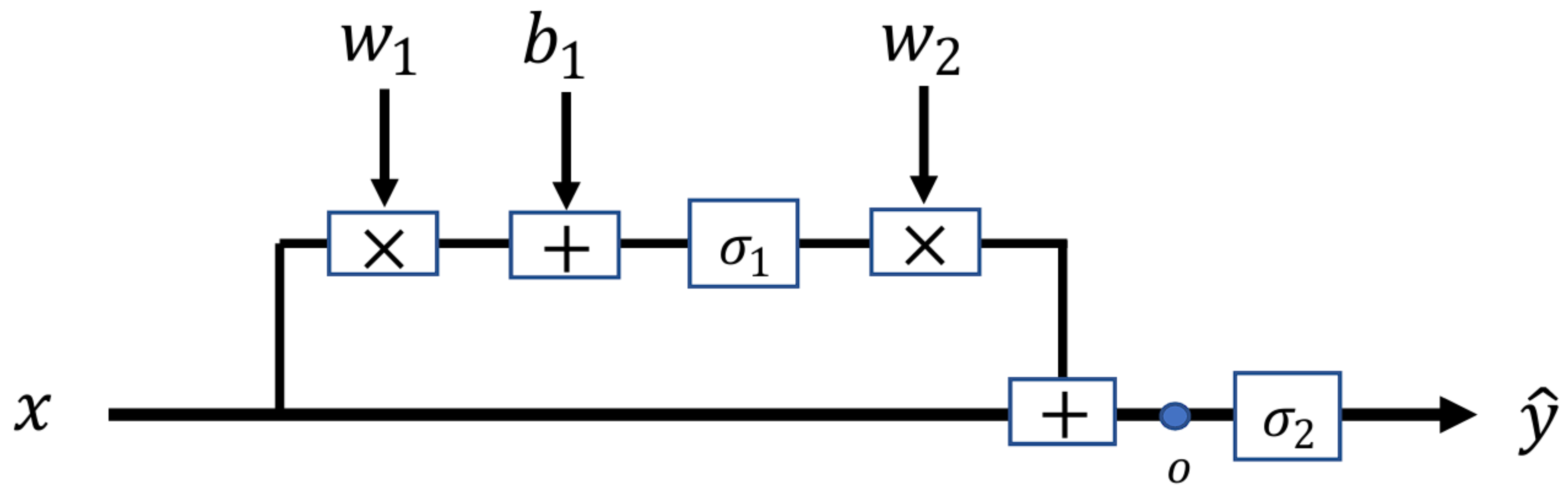
- Intermediate result from hidden layer: o_1
- Weight: w_2
- Activation function: σ_2

The second step involves multiplying the intermediate result o_1 by w_2:

$$\hat{y} = \sigma_2(x + w_2 \cdot \sigma_1(w_1 \cdot x + b_1))$$

Let’s See an Example!

Forward Propagation



$\sigma(z) = \frac{1}{1+e^{-z}}$

Given:

- $x = 1.0$
- $y = 1$
- $w_1 = 0.5$
- $w_2 = -2$
- $b_1 = -0.5$

Question:

Assume you have an input sample $x = 1.0$ with associated label $y = 1$, the weights $w_1 = 0.5$, $w_2 = -2$ and the bias $b_1 = -0.5$. Compute the loss for these numbers.

Forward Propagation

1. **Compute the intermediate value o_1:**

$$o_1 = \sigma_1(w_1 \cdot x + b_1) = \sigma_1(0.5 \cdot 1.0 + (-0.5)) = \sigma_1(0.5 - 0.5) = \sigma_1(0)$$

   Using the sigmoid function $\sigma(z) = \frac{1}{1+e^{-z}}$:

$$\sigma_1(0) = \frac{1}{1+e^0} = \frac{1}{2} = 0.5$$

2. **Compute the intermediate value o:**

$$o = x + w_2 \cdot o_1 = 1.0 + (-2) \cdot 0.5 = 1.0 - 1.0 = 0$$

3. **Compute the output $\hat{y}$:**

$$\hat{y} = \sigma_2(o) = \sigma_2(0) = \frac{1}{1+e^0} = \frac{1}{2} = 0.5$$

$\sigma(z) = \frac{1}{1+e^{-z}}$

Given:

- $x = 1.0$
- $y = 1$
- $w_1 = 0.5$
- $w_2 = -2$
- $b_1 = -0.5$

Forward Propagation

4. **Compute the loss L:**

   Using the Mean Squared Error (MSE) loss function:

$$L = (\hat{y} - y)^2 = (0.5 - 1)^2 = (-0.5)^2 = 0.25$$

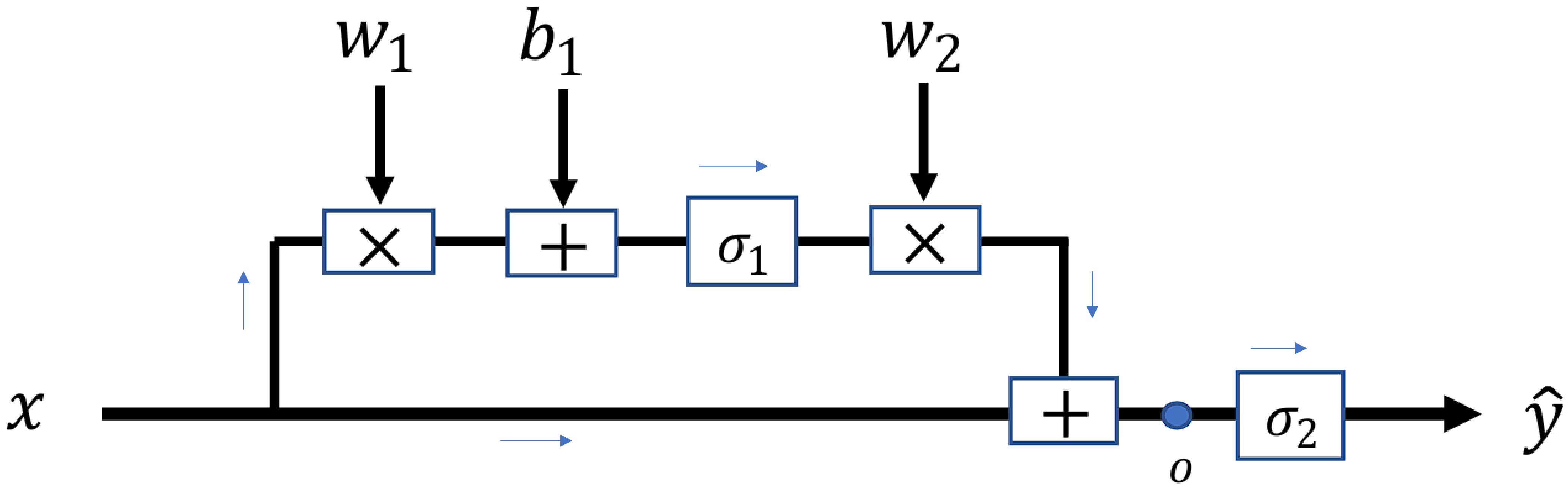
$\sigma(z) = \frac{1}{1+e^{-z}}$

Given:

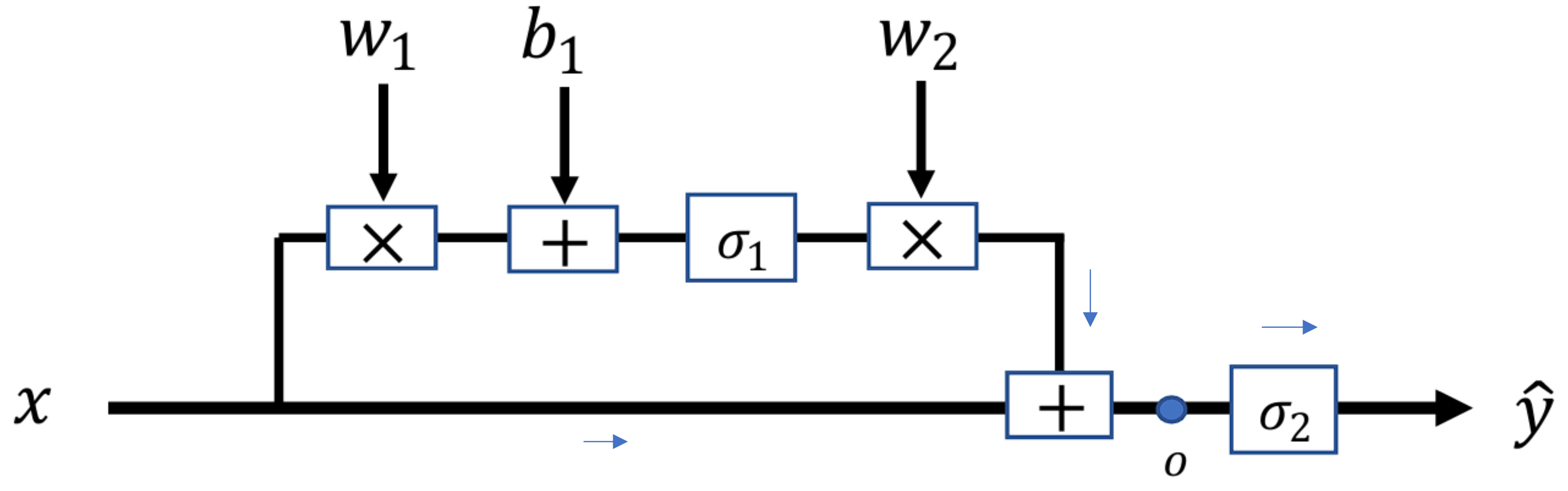
- $x = 1.0$
- $y = 1$
- $w_1 = 0.5$
- $w_2 = -2$
- $b_1 = -0.5$

Backpropagation using Chain Rule

Backpropagation



Backpropagation



$$\frac{\partial L}{\partial \hat{y}} = ?$$

$$\frac{\partial L}{\partial o} = ?$$

$$\frac{\partial L}{\partial w_2} = ?$$

$$\frac{\partial L}{\partial b_1} = ?$$

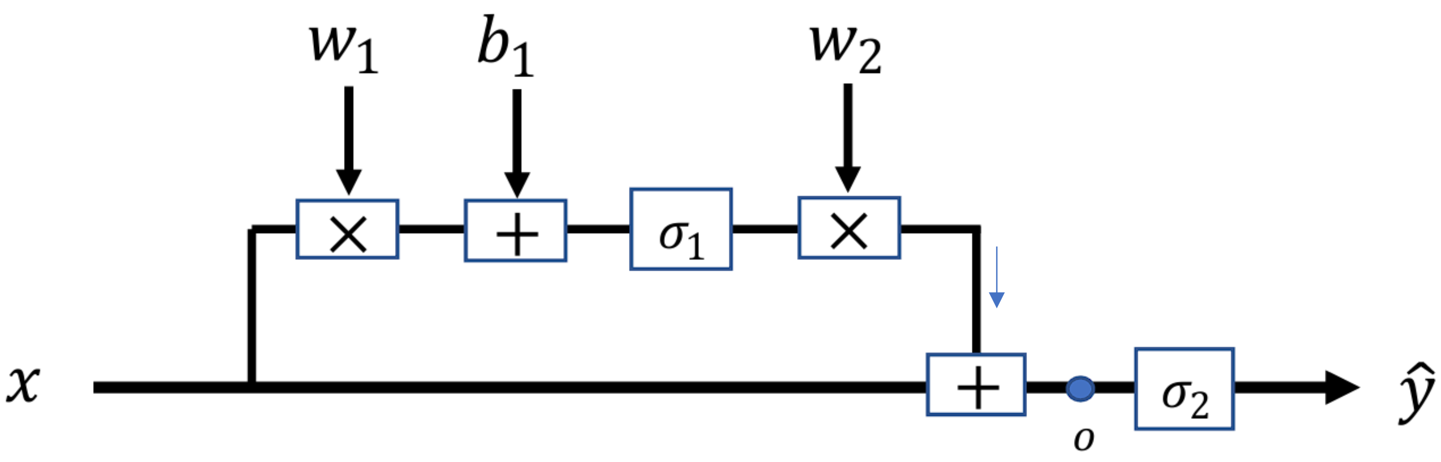
$$\frac{\partial L}{\partial w_1} = ?$$

$$\frac{\partial L}{\partial x} = ?$$

Question:

Derive the partial derivatives for the network f listed below. If necessary, define auxiliary components, which may simplify your answers.

Backpropagation



Let $\sigma'(x)$ be the derivative of $\sigma(x)$, which are defined as

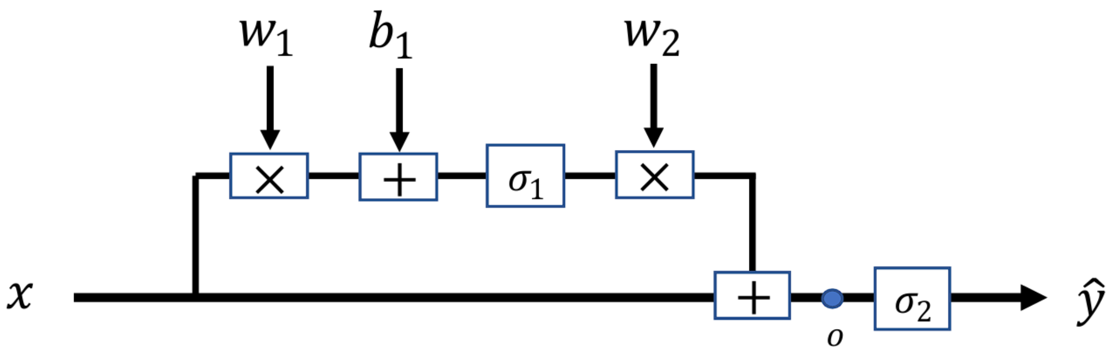
$$\sigma(x) = \frac{1}{1+e^{-x}}$$
$$\sigma'(x) = \sigma(x)(1-\sigma(x))$$

Loss = $\|\hat{y} - y\|_2^2 = (\hat{y} - y)^2$

Solution:

$$\frac{\partial L}{\partial \hat{y}} = 2(\hat{y} - y)$$
$$\frac{\partial L}{\partial o} = \frac{\partial L}{\partial \hat{y}} \sigma_2'(x + w_2\sigma_1(w_1x + b_1))$$
$$\frac{\partial L}{\partial w_2} = \frac{\partial L}{\partial o} \sigma_1(w_1x + b_1)$$
$$\frac{\partial L}{\partial b_1} = \frac{\partial L}{\partial o} w_2\sigma_1'(w_1x + b_1)$$
$$\frac{\partial L}{\partial w_1} = \frac{\partial L}{\partial o} w_2\sigma_1'(w_1x + b_1)x$$
$$\frac{\partial L}{\partial x} = \frac{\partial L}{\partial o} + \frac{\partial L}{\partial o} w_2\sigma_1'(w_1x + b_1)w_1$$

Backpropagation



Loss = $\|\hat{y} - y\|_2^2 = (\hat{y} - y)^2$

$o = w_2\sigma_1(w_1x + b_1) + x$

1. Gradient of Loss with Respect to $\hat{y}$

$$\frac{\partial L}{\partial \hat{y}} = 2(\hat{y} - y)$$

2. Gradient of Loss with Respect to o

$$\frac{\partial L}{\partial o} = \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial o}$$

$$\frac{\partial L}{\partial o} = \frac{\partial L}{\partial \hat{y}} \cdot \sigma_2'(o)$$

$$\frac{\partial L}{\partial o} = 2(\hat{y} - y) \cdot \sigma_2'(o)$$

$$\frac{\partial L}{\partial o} = 2(\hat{y} - y) \cdot \sigma_2'(x + w_2\sigma_1(w_1x + b_1))$$

3. Gradient of Loss with Respect to w_2

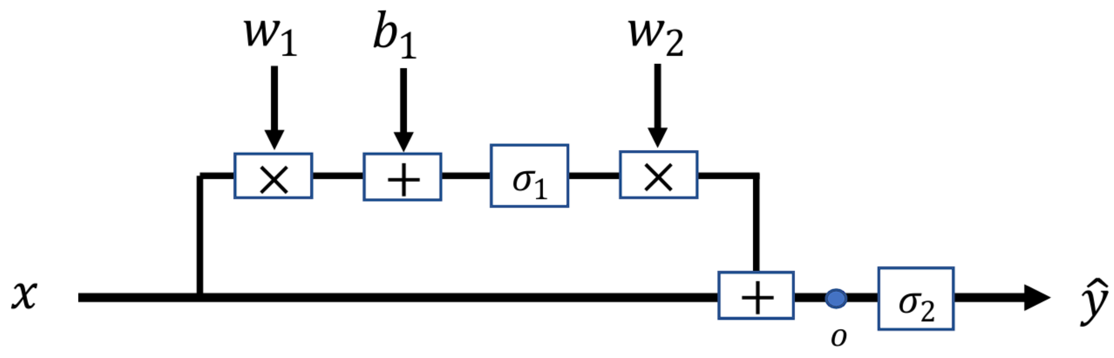
$$\frac{\partial L}{\partial w_2} = \frac{\partial L}{\partial o} \cdot \frac{\partial o}{\partial w_2}$$

$$\frac{\partial L}{\partial w_2} = \frac{\partial L}{\partial o} \cdot \sigma_1(w_1 x + b_1)$$

4. Gradient of Loss with Respect to b_1

$$\frac{\partial L}{\partial b_1} = \frac{\partial L}{\partial o} \cdot \frac{\partial o}{\partial b_1}$$

$$\frac{\partial L}{\partial b_1} = \frac{\partial L}{\partial o} \cdot w_2 \cdot \sigma_1'(w_1 x + b_1)$$



Loss = $\|\hat{y} - y\|_2^2 = (\hat{y} - y)^2$

$o = w_2 \sigma_1(w_1 x + b_1) + x$

5. Gradient of Loss with Respect to w_1

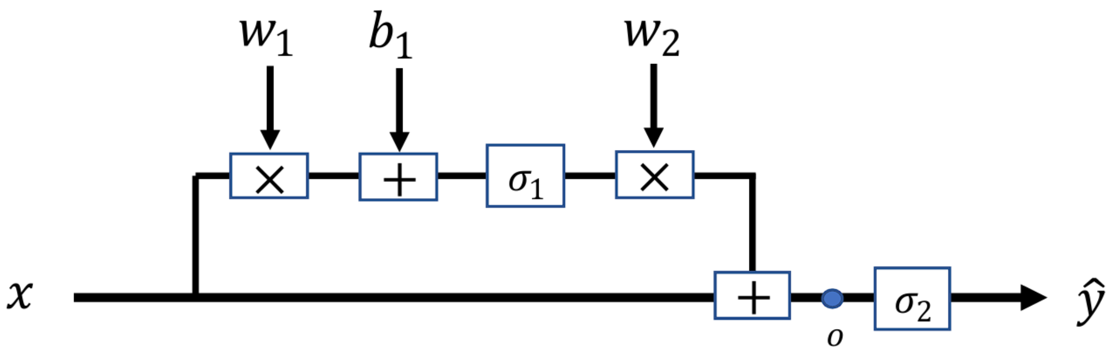
$$\frac{\partial L}{\partial w_1} = \frac{\partial L}{\partial o} \cdot \frac{\partial o}{\partial w_1}$$

$$\frac{\partial L}{\partial w_1} = \frac{\partial L}{\partial o} \cdot w_2 \cdot \sigma_1'(w_1 x + b_1) \cdot x$$

6. Gradient of Loss with Respect to x

$$\frac{\partial L}{\partial x} = \frac{\partial L}{\partial o} \cdot \frac{\partial o}{\partial x}$$

$$\frac{\partial L}{\partial x} = \frac{\partial L}{\partial o} \cdot \left(1 + w_2 \cdot \sigma_1'(w_1 x + b_1) \cdot w_1\right)$$



Loss = $\|\hat{y} - y\|_2^2 = (\hat{y} - y)^2$

$o = w_2 \sigma_1(w_1 x + b_1) + x$

Update the Weights and Bias using Gradient Descent

Backpropagation

Update the Weights and Bias using Gradient Descent

Using a learning rate α:

$$w_1^{new} = w_1^{old} - \alpha \frac{\partial L}{\partial w_1}$$

$$b_1^{new} = b_1 - \alpha \frac{\partial L}{\partial b_1}$$

$$w_2^{new} = w_2^{old} - \alpha \frac{\partial L}{\partial w_2}$$

- $w_1^{new} = 0.4875$
- $b_1^{new} = -0.5125$
- $w_2^{new} = -1.9875$

For example, if $\alpha = 0.1$:

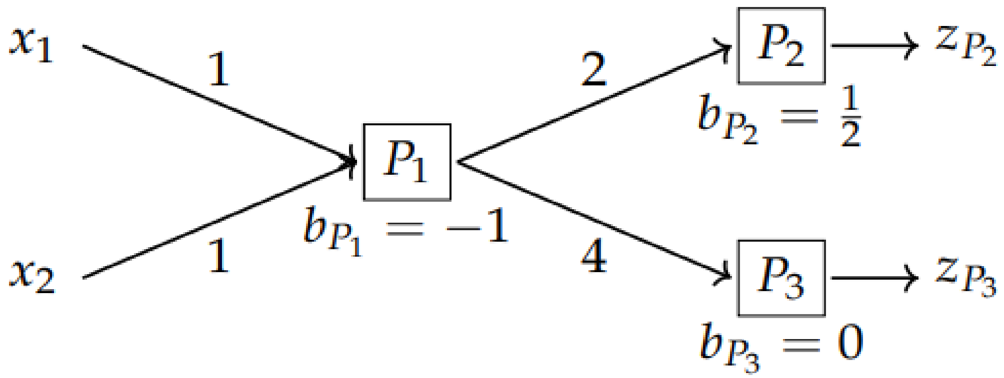
$$w_1^{new} = 0.5 - 0.1 \times 0.125 = 0.5 - 0.0125 = 0.4875$$

$$b_1^{new} = -0.5 - 0.1 \times 0.125 = -0.5 - 0.0125 = -0.5125$$

$$w_2^{new} = -2 - 0.1 \times (-0.125) = -2 + 0.0125 = -1.9875$$

Backpropagation in ANN

Consider the following network with 3 neurons P_1, P_2, P_3



with initial weights (as denoted in the graph)

$$w_{P_1 x_1} = 1,\ w_{P_1 x_2} = 1,\ w_{P_2 P_1} = 2,\ w_{P_3 P_1} = 4$$

initial biases (as denoted in the graph) $b_{P_1} = -1$, $b_{P_2} = \frac{1}{2}$, $b_{P_3} = 0$, and activation functions

Backpropagation in ANN

$$\psi_{P_1}(t) = \frac{1}{1+3^{-t}}, \;\; \psi_{P_2}(t) = t^2, \;\; \psi_{P_3}(t) = t^2.$$

You may use without proof that the derivative of ψ_{P_1} is given by

$$\psi'_{P_1}(t) \approx \psi_{P_1}(t)(1 - \psi_{P_1}(t)).$$

Let

$$\theta = (w_{P_1x_1}, w_{P_1x_2}, w_{P_2P_1}, w_{P_3P_1}, b_{P_1}, b_{P_2}, b_{P_3})^T$$

and let $f_\theta(x) = (z_{P_2}, z_{P_3})^T \in \mathbb{R}^2$ denote the output of the network using parameters θ and input $x = (x_1, x_2)^T \in \mathbb{R}^2$. Consider the loss function $C(\theta; x, y) = \frac{1}{2}||f_\theta(x) - y||^2$ for a given training pair (x, y).

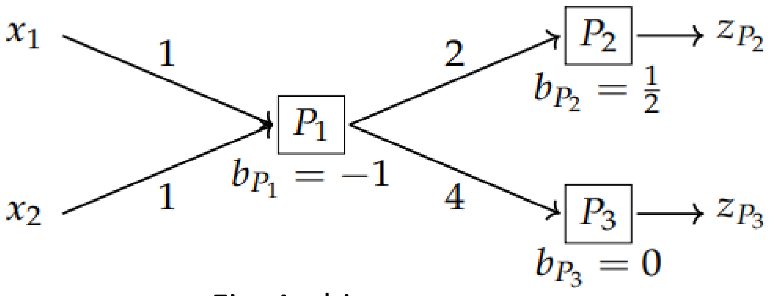
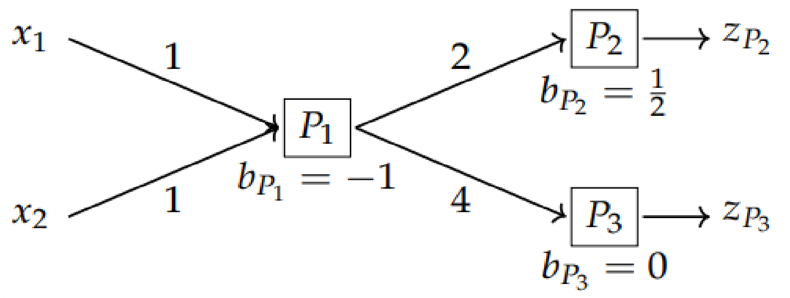


Fig: Architecture

a) Perform one training iteration using the input data $x^1 = \begin{pmatrix} -1 \\ 3 \end{pmatrix}$, $y^1 = \begin{pmatrix} 3 \\ 8 \end{pmatrix}$, and step size $\eta = 0.1$. State the updated weights and biases.



b) Assume that you are given a second point (x^2, y^2) with

$$\nabla C(\theta; x^2, y^2) = (6, 2, 1, 5.5, 10, -4, 2)^T.$$

What are the updated weights and biases if you use both points and the mean squared loss function in the first training iteration instead of only using x^1 as in a) (again with stepsize $\eta = 0.1$)?

Solution

$$\frac{\partial C}{\partial b_{P_2}} = ?$$
$$\frac{\partial C}{\partial b_{P_3}} = ?$$
$$\frac{\partial C}{\partial b_{P_1}} = ?$$
$$\frac{\partial C}{\partial w_{P_2 P_1}} = ?$$
$$\frac{\partial C}{\partial w_{P_3 P_1}} = ?$$
$$\frac{\partial C}{\partial w_{P_1 x_1}} = ?$$
$$\frac{\partial C}{\partial w_{P_1 x_2}} = ?$$

We update the parameters with a gradient step

$$\theta^{new} = \theta - \eta \nabla C(\theta) = \begin{pmatrix} 1 \\ 1 \\ 2 \\ 4 \\ -1 \\ 0.5 \\ 0 \end{pmatrix} - 0.1 \cdot \begin{pmatrix} -6 \\ 18 \\ 3 \\ 4,5 \\ 6 \\ 4 \\ 6 \end{pmatrix} = \begin{pmatrix} 1,6 \\ -0,8 \\ 1,7 \\ 3,55 \\ -1,6 \\ 0,1 \\ -0,6 \end{pmatrix}.$$

The updated parameters are therefore given by

$$w_{P_1 x_1}^{new} = 1,6 \quad w_{P_1 x_2}^{new} = -0,8 \quad w_{P_2 P_1}^{new} = 1,7 \quad w_{P_3 P_1}^{new} = 3,55$$
$$b_{P_1}^{new} = -1,6 \quad b_{P_2}^{new} = 0,1 \quad b_{P_3}^{new} = -0,6.$$

b) Using a second data point we have to compute the averaged gradient $\overline{\nabla}C(\theta)$ and use it for the update step. The averaged gradient is given by

$$\overline{\nabla}C(\theta) = \frac{1}{2}(\nabla(\theta, x^1, y^1) + \nabla(\theta, x^2, y^2)) = \frac{1}{2}\begin{pmatrix} 0 \\ 20 \\ 4 \\ 10 \\ 16 \\ 0 \\ 8 \end{pmatrix} = \begin{pmatrix} 0 \\ 10 \\ 2 \\ 5 \\ 8 \\ 0 \\ 4 \end{pmatrix}.$$

The update is given by

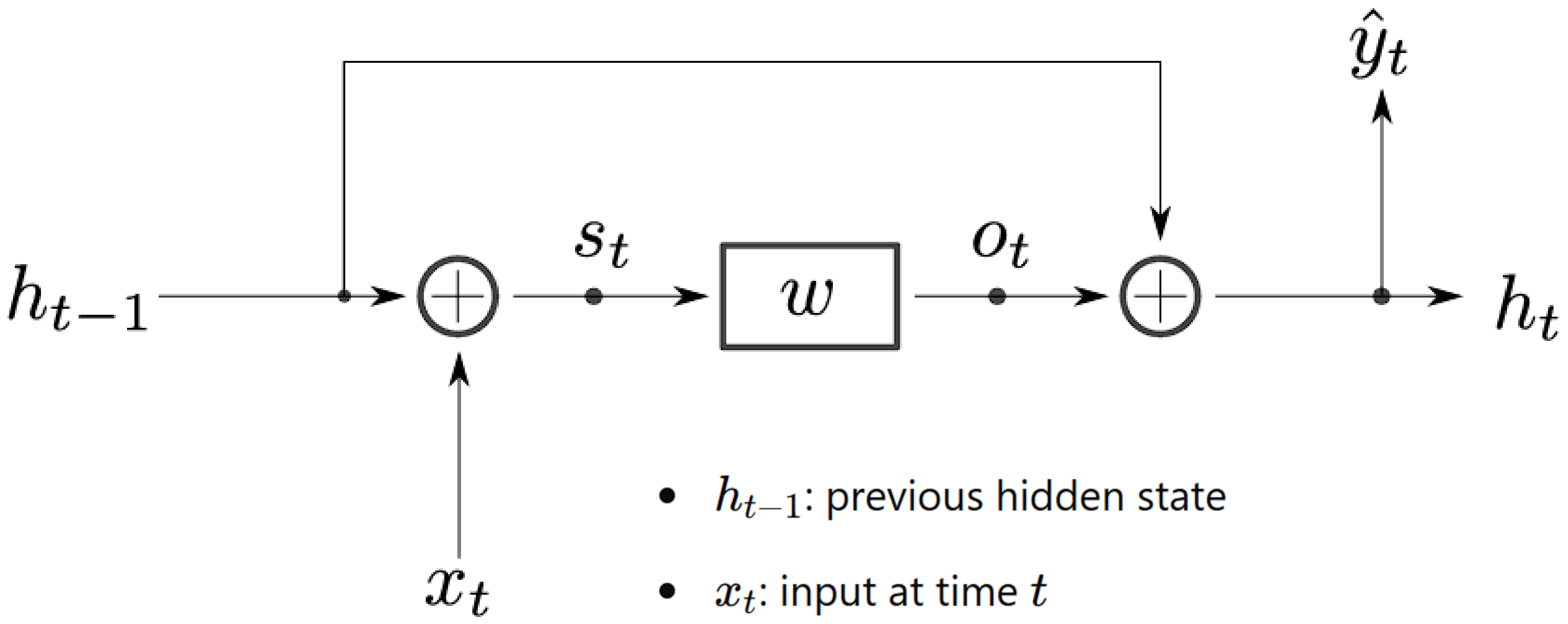
$$\theta^{new} = \theta - \eta\overline{\nabla}C(\theta) = \begin{pmatrix} 1 \\ 1 \\ 2 \\ 4 \\ -1 \\ 0.5 \\ 0 \end{pmatrix} - 0.1 \cdot \begin{pmatrix} 0 \\ 10 \\ 2 \\ 5 \\ 8 \\ 0 \\ 4 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 1,8 \\ 3,5 \\ -1,8 \\ 0,5 \\ -0,4 \end{pmatrix}.$$

The updated parameters are therefore given by

$$w^{new}_{P_1x_1} = 1 \quad w^{new}_{P_1x_2} = 0 \quad w^{new}_{P_2P_1} = 1,8 \quad w^{new}_{P_3P_1} = 3,5$$
$$b^{new}_{P_1} = -1,8 \quad b^{new}_{P_2} = 0,5 \quad b^{new}_{P_3} = -0,4.$$

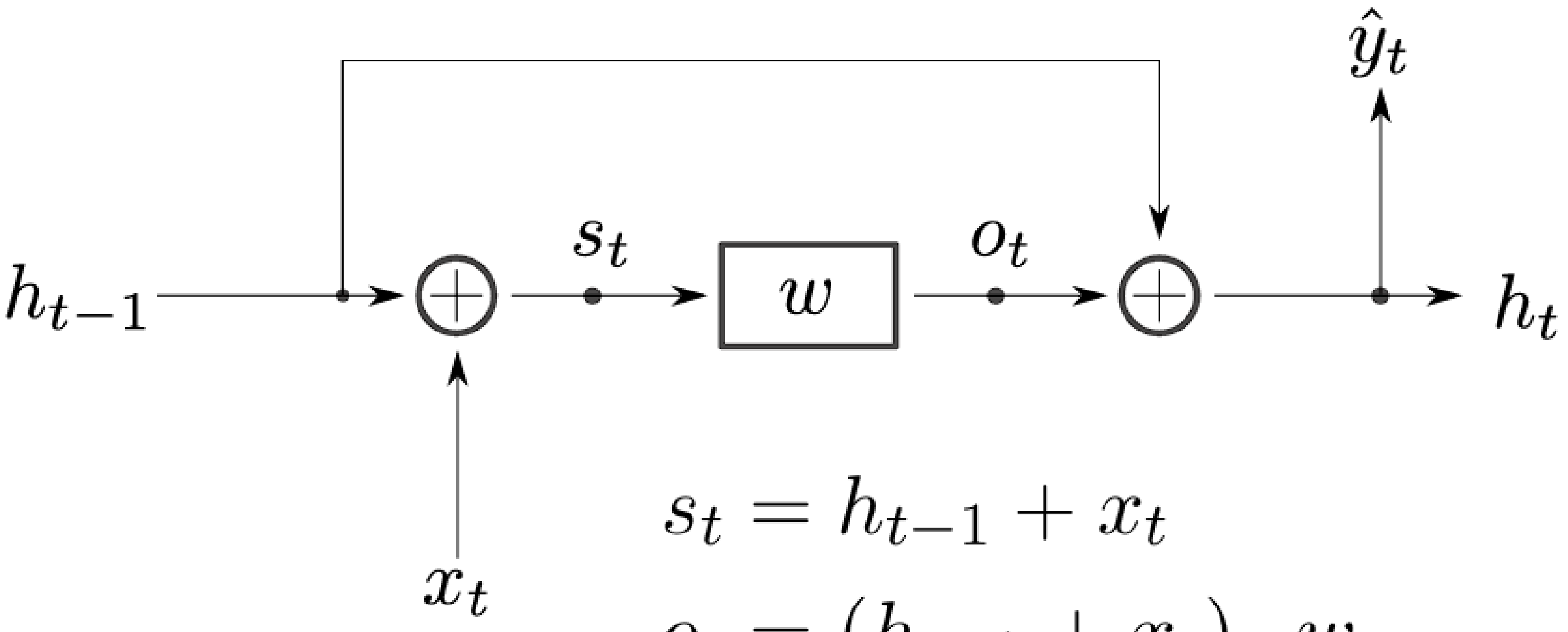
Backpropagation in RNN

Backprop in RNN



- h_{t-1}: previous hidden state
- x_t: input at time t
- w: weight
- y_t: true output
- $\hat{y}_t$: predicted output

Backpropagation in RNN



$$s_t = h_{t-1} + x_t$$

$$o_t = (h_{t-1} + x_t) \cdot w$$

$$f(x_t, h_{t-1}) = \underbrace{(h_{t-1} + x_t) \cdot w}_{o_t} + h_{t-1} = \hat{y}_t$$

Backpropagation in RNN

$$\frac{\partial L}{\partial \hat{y}_t} = 2(\hat{y}_t - y_t)$$

$$\frac{\partial L}{\partial o_t} = \frac{\partial L}{\partial \hat{y}_t} \cdot \frac{\partial \hat{y}_t}{\partial o_t} + \frac{\partial L}{\partial h_t} \cdot \frac{\partial h_t}{\partial o_t} = \frac{\partial L}{\partial \hat{y}_t} + \frac{\partial L}{\partial h_t}$$

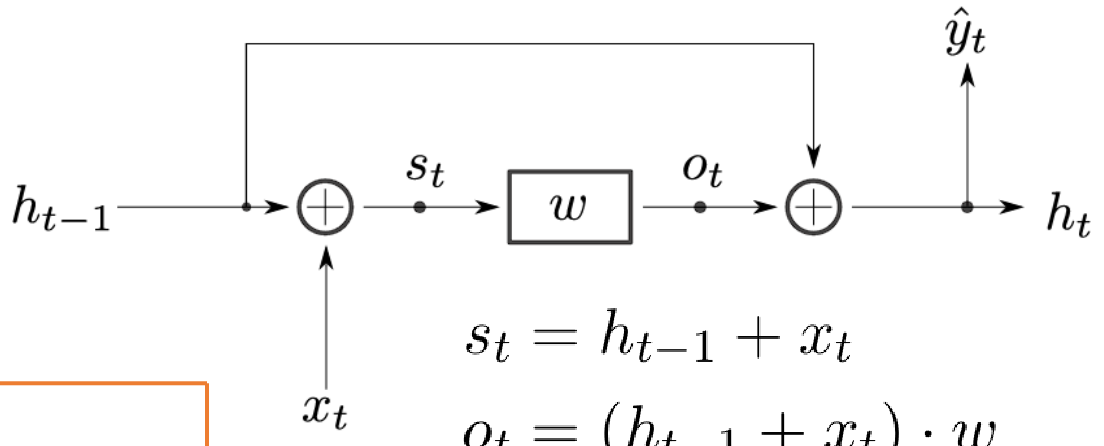
$$\frac{\partial L}{\partial w_t} = \frac{\partial L}{\partial o_t} \cdot \frac{\partial o_t}{\partial w_t} = \frac{\partial L}{\partial o_t} \cdot (h_{t-1} + x_t) = s_t \cdot \frac{\partial L}{\partial o_t}$$

$$\frac{\partial L}{\partial w} = \sum_t \frac{\partial L}{\partial w_t}$$

$$\frac{\partial L}{\partial s_t} = \frac{\partial L}{\partial o_t} \cdot \frac{\partial o_t}{\partial s_t} = \frac{\partial L}{\partial o_t} \cdot w$$

$$\frac{\partial L}{\partial x_t} = \frac{\partial L}{\partial s_t} \cdot \frac{\partial s_t}{\partial x_t} = \frac{\partial L}{\partial s_t}$$

$$\frac{\partial L}{\partial h_{t-1}} = \frac{\partial L}{\partial o_t} + \frac{\partial L}{\partial s_t}$$



$$s_t = h_{t-1} + x_t$$

$$o_t = (h_{t-1} + x_t) \cdot w$$

$$\hat{y}_t = h_{t-1} + o_t = h_t$$

Backpropagation in RNN

1. Gradient of the Loss with respect to $\hat{y}_t$:

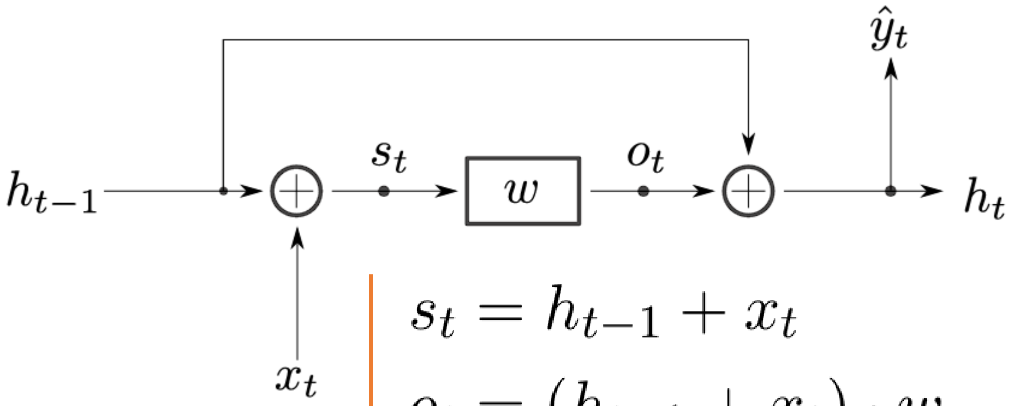
$$\frac{\partial L}{\partial \hat{y}_t} = 2(\hat{y}_t - y_t)$$

2. Gradient of the Loss with respect to o_t:

$$\frac{\partial L}{\partial o_t} = \frac{\partial L}{\partial \hat{y}_t} \cdot \frac{\partial \hat{y}_t}{\partial o_t} + \frac{\partial L}{\partial h_t} \cdot \frac{\partial h_t}{\partial o_t} = \frac{\partial L}{\partial \hat{y}_t} + \frac{\partial L}{\partial h_t}$$

3. Gradient of Loss w.r.t. w_t:

$$\frac{\partial L}{\partial w_t} = \frac{\partial L}{\partial o_t} \cdot \frac{\partial o_t}{\partial w_t} = \frac{\partial L}{\partial o_t} \cdot (h_{t-1} + x_t) = s_t \cdot \frac{\partial L}{\partial o_t}$$



$$s_t = h_{t-1} + x_t$$

$$o_t = (h_{t-1} + x_t) \cdot w$$

$$\hat{y}_t = h_{t-1} + o_t = h_t$$

$$\frac{\partial L}{\partial h_t} = \frac{\partial L}{\partial \hat{y}_t} \cdot \frac{\partial \hat{y}_t}{\partial h_t}$$

$$\frac{\partial \hat{y}_t}{\partial h_t} = 1$$

$$\frac{\partial L}{\partial h_t} = 2(\hat{y}_t - y_t) = 2(h_t - y_t)$$

Backpropagation in RNN

4. **Gradient of Loss w.r.t. w:**

$$\frac{\partial L}{\partial w} = \sum \frac{\partial L}{\partial w_t}$$

5. **Gradient of Loss w.r.t. s_t:**

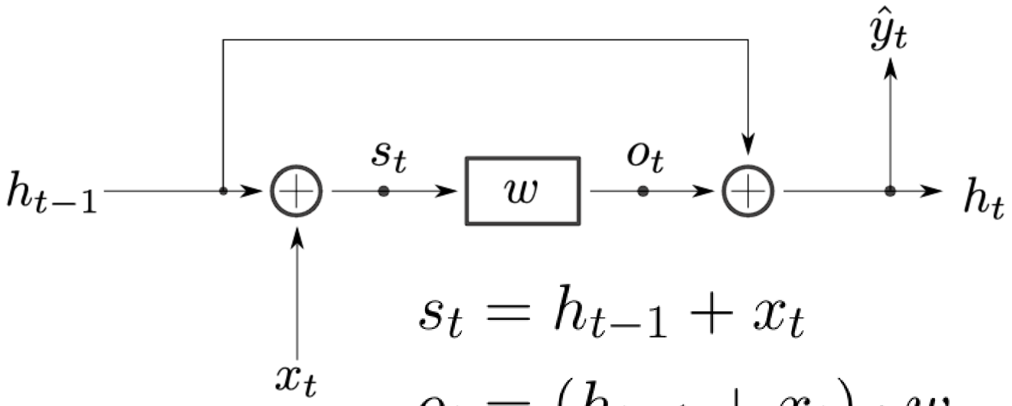
$$\frac{\partial L}{\partial s_t} = \frac{\partial L}{\partial o_t} \cdot \frac{\partial o_t}{\partial s_t} = \frac{\partial L}{\partial o_t} \cdot w$$

6. **Gradient of Loss w.r.t. x_t:**

$$\frac{\partial L}{\partial x_t} = \frac{\partial L}{\partial s_t} \cdot \frac{\partial s_t}{\partial x_t} = \frac{\partial L}{\partial s_t}$$

7. **Gradient of Loss w.r.t. h_{t-1}:**

$$\frac{\partial L}{\partial h_{t-1}} = \frac{\partial L}{\partial o_t} + \frac{\partial L}{\partial s_t}$$



$$s_t = h_{t-1} + x_t$$

$$o_t = (h_{t-1} + x_t) \cdot w$$

$$\hat{y}_t = h_{t-1} + o_t = h_t$$

$$\frac{\partial L}{\partial o_t} = \frac{\partial L}{\partial \hat{y}_t} + \frac{\partial L}{\partial h_t}$$

Update Weights using Gradient Descent

Gradient Descent Update Rule

The update rule using gradient descent is: $w \leftarrow w - \eta \cdot \frac{\partial L}{\partial w}$

1. **Update for w:**

$$w \leftarrow w - \eta \cdot s_t \cdot \frac{\partial L}{\partial o_t}$$

Here,

$$\frac{\partial L}{\partial o_t} = \frac{\partial L}{\partial \hat{y}_t} \cdot \frac{\partial \hat{y}_t}{\partial o_t} + \frac{\partial L}{\partial h_t} \cdot \frac{\partial h_t}{\partial o_t} = \frac{\partial L}{\partial \hat{y}_t} + \frac{\partial L}{\partial h_t}$$

Substituting $\frac{\partial L}{\partial o_t}$:

$$w \leftarrow w - \eta \cdot s_t \cdot (2(\hat{y}_t - y_t) + \frac{\partial L}{\partial h_t})$$

Gradient Descent Update Rule

The update rule using gradient descent is: $w \leftarrow w - \eta \cdot \frac{\partial L}{\partial w}$

2. **Update for x_t:**

$$x_t \leftarrow x_t - \eta \cdot \frac{\partial L}{\partial s_t}$$

Using $\frac{\partial L}{\partial s_t} = w \cdot \frac{\partial L}{\partial o_t}$:

$$x_t \leftarrow x_t - \eta \cdot w \cdot (2(\hat{y}_t - y_t) + \frac{\partial L}{\partial h_t})$$

Here,

$$\frac{\partial L}{\partial h_t} = 2(\hat{y}_t - y_t) = 2(h_t - y_t)$$

Gradient Descent Update Rule

The update rule using gradient descent is: $w \leftarrow w - \eta \cdot \frac{\partial L}{\partial w}$

3. **Update for h_{t-1}:**

$$h_{t-1} \leftarrow h_{t-1} - \eta \cdot \frac{\partial L}{\partial h_{t-1}}$$

Since $\frac{\partial L}{\partial h_{t-1}} = 2(\hat{y}_t - y_t)$ directly from $\hat{y}_t = h_t$:

$$h_{t-1} \leftarrow h_{t-1} - \eta \cdot 2(\hat{y}_t - y_t)$$