Course 1

Calculus

Week 1 _ Derivatives and Optimization

calculating the Slope

$$\text{Solpe} = \frac{\text{rise} \uparrow}{\text{ran} \rightarrow} \quad , \quad \text{Slope} = \frac{\text{change in distance } (\Delta x)}{\text{change in time } (\Delta t)}$$

Derivatives

$$\text{Slope} = \frac{\text{change in distance}}{\text{change in time}} \quad \begin{matrix}(\text{vertical change}) \\ (\text{horizontal change})\end{matrix} \longrightarrow \text{slope} = \frac{\Delta x}{\Delta t}$$

$$\text{slope at } \underline{\underline{\text{a point}}} = \frac{dx}{dt} \quad (\text{derivative})$$

Lagrange's notation

Notation: Function: $y = f(x)$ $\longrightarrow$ Derivative of f: $\frac{df(x)}{dx} = f'(x)$

Leibniz's notation

Some Common Derivatives

1) $y = f(x) = c$ (line) $\longrightarrow$ $f'(x) = 0$

2) $f(x) = ax + b$ (line) $\longrightarrow$ $f'(x) = a$

3) $f(x) = x^2$ (quadratic) $\longrightarrow$ $f'(x) = 2x$

4) $f(x) = x^3$ (cubic) $\longrightarrow$ $f'(x) = 3x^2$

5) $f(x) = 1/x$ $\longrightarrow$ $f'(x) = \frac{-1}{x^2} = -x^{-2}$

6) Power Func : $f(x) = x^n \longrightarrow f'(x) = n x^{n-1}$

* Inverse Function: $g(x)$ and $f(x)$ are inverses $\longrightarrow g(x) = f^{-1}(x)$

$\longrightarrow g(f(x)) = x$

7) Inverse : $f(x)$, $g(y)$ $\longrightarrow g'(y) = \frac{1}{f'(x)}$

are inverses

Ex: $f(x) = x^2$, $g(y) = \sqrt{y} = y^{1/2}$

$\downarrow$ $\downarrow$

$f'(x) = 2x$ $\quad g'(y) = \frac{1}{2} y^{-1/2} = \frac{1}{2\sqrt{y}}$

$\Rightarrow g'(y) = \frac{1}{f'(x)}$

$\rightsquigarrow$ at the point $(2,4)$: $g'(4) = \frac{1}{4}$ $\quad f'(2) = 4 \longrightarrow g'(y) = \frac{1}{f'(x)}$

8) Triaonometric Functions :

- $f(x) = \sin(x) \longrightarrow f'(x) = \cos(x)$
- $f(x) = \cos(x) \longrightarrow f'(x) = -\sin(x)$

9) Exponential : $f(x) = e^x \longrightarrow f'(x) = e^x$

* Logarithm : $\log(x)$ is the inverse of e^x

10) Logarithm : $f(x) = \log(x) \longrightarrow f'(x) = \frac{d}{dx} \log x = \frac{1}{x}$

* Differentiable Functions:

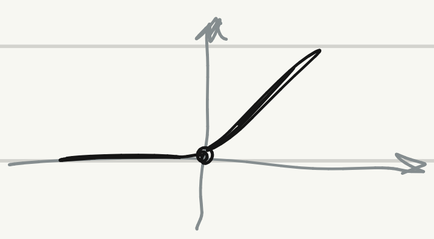
→ at a point: The derivative has to exist for that point.

→ at an interval: The derivative has to exist for every point in the interval.

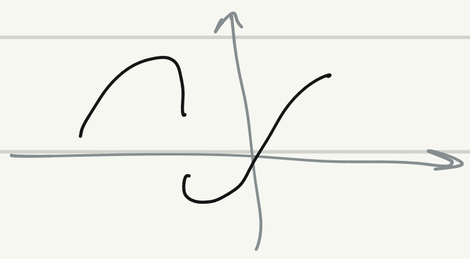
* Non Differentiable Functions:

Generally, when a Function has a corner or a cusp, the function is not differentiable at that point.

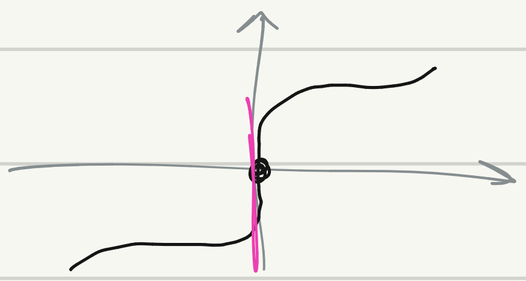
1) Corners / cusps



2) jump discontinuity



3) Vertical tangents



11) The Sum Rule:

$$f = g + h \longrightarrow f' = g' + h'$$

12) The product Rule:

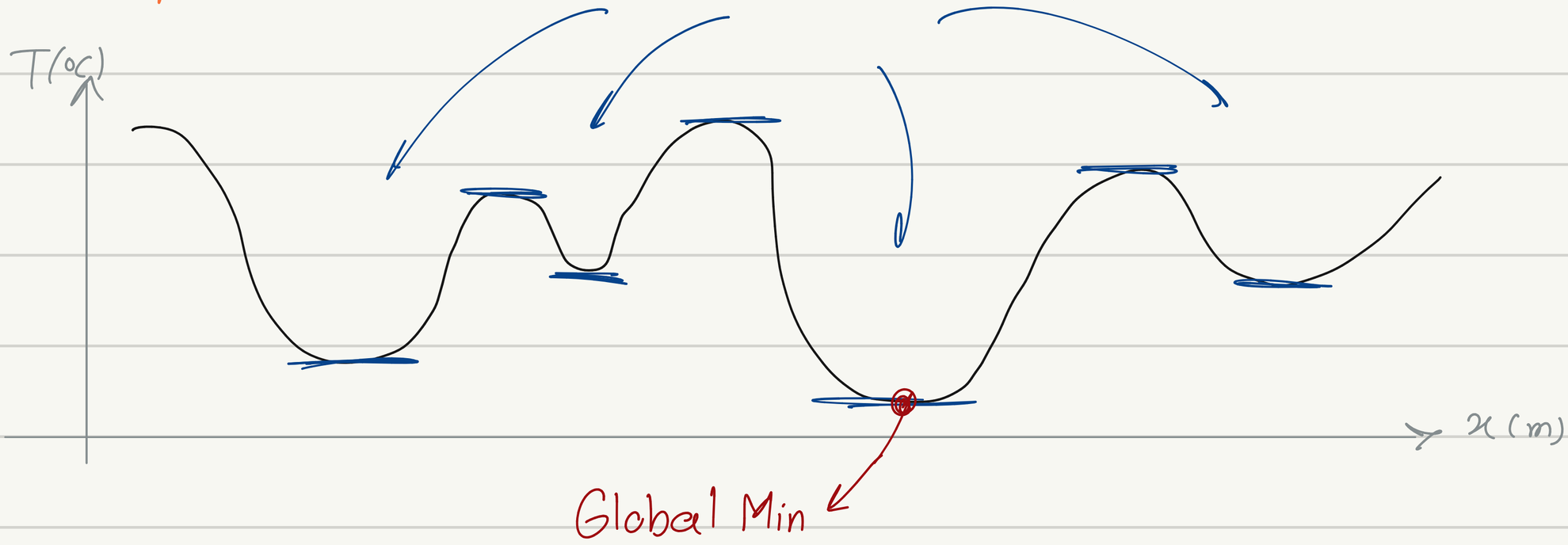
$$f = gh \longrightarrow f' = g'h + gh'$$

13) The chain Rule:

$$f(g(h(t))) \longrightarrow \frac{d}{dt} f(g(h(t))) = \frac{df}{dg} \cdot \frac{dg}{dh} \cdot \frac{dh}{dt} = f'(g(h(t))) \cdot g'(h(t)) \cdot h'(t)$$

Optimization

→ Multiple Minima :



→ optimization of squared loss :

- Total cost : $(x-a)^2 + (x-b)^2 \xrightarrow[\text{So}]{\text{Goal: Find min}} \frac{d}{dx}\left[(x-a)^2 + (x-b)^2\right] = 0$

Solution $\Longrightarrow$ $x = \frac{a+b}{2}$

- Cost Func : $(x-a)^2 + (x-b)^2 + (x-c)^2 \xrightarrow{\text{Good: minimize func}} \frac{d}{dx}\text{Cost Func} = 0$

Solution $\Longrightarrow$ $x = \frac{a+b+c}{3}$

So $\Longrightarrow$ Minimize $(x-a_1)^2 + (x-a_2)^2 + \dots + (x-a_n)^2$

Solution: $x = \frac{a_1 + a_2 + \dots + a_n}{n}$

→ optimization of log loss :

- Func : $g(p) = p^7(1-p)^3 \xrightarrow{\text{Goal : Minimize}} g'(p) = 0$ * But it's hard to solve.

↓ Better way

$G(p) = 7\log(p) - 3\log(1-p) \xrightarrow{\text{minimize}} G'(p) = 0$ * Easier to solve.

Week 2 : Gradients and Gradient Descent

Partial Derivatives

- Functions with one variable : $f(x)$ $\xrightarrow{\text{derivative}}$ $f'(x) = \frac{df}{dx}$

* We calculate derivative for one variable functions with using the rules that we've learned in previous chapter.

- Functions with two variables : $f(x,y)$ $\xrightarrow{\text{derivative}}$ $f'(x,y) = ?$

* We calculate derivative for two variables functions with Partial Derivatives.

What is partial derivatives ?!

Ex. $f(x,y) = x^2 + y^2$ $\longrightarrow$ $f'(x,y) = ?$

step 1: Treat y as a constant $\Rightarrow$ f become one variable function

step 2: Differentiate the function using the normal rules.

step3: Repeat step 1, 2 for the next variable (x).

$f(x,y) = x^2 + \underbrace{y^2}_{\text{constant}}$ and $f(x,y) = \underbrace{x^2}_{\text{constant}} + y^2$

$\frac{\partial f}{\partial x} = 2x$ $\frac{\partial f}{\partial y} = 2y$

⇒ Partial Derivatives Notation:

$f(x,y)$

partial derivative of f with respect to y.

partial derivative of f with respect to y.

$$f_x = \frac{\partial f}{\partial x} \qquad f_y = \frac{\partial f}{\partial y}$$

pronounced "die"

* The partial derivative is denoted by the symbol ∂, which replaces the roman letter d used to denote a full derivative.

Ex. $f(x,y) = 3x^2y^3$

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

• Gradient •

$f(x,y) \longrightarrow$ gradient: $\nabla f = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \\ \frac{\partial f}{\partial y} \end{bmatrix}$

Ex. Find the gradient of $f(x,y) = x^2 + y^2$ at $(2,3)$:

$$\nabla f = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \\ \frac{\partial f}{\partial y} \end{bmatrix} = \begin{bmatrix} 2x \\ 2y \end{bmatrix}\Bigg|_{(2,3)} = \begin{bmatrix} 4 \\ 6 \end{bmatrix}$$

Gradients and min / max

1) Functions of one variable : $f(x) \longrightarrow$ min is when slope $= 0$ ($f'(x) = 0$)

slope $\equiv$ derivative

2) Functions of two variable : $f(x,y) \longrightarrow$ min is when both slopes $= 0$

$f'(x,y) = 0$ means $\frac{\partial f}{\partial y} = 0$ and $\frac{\partial f}{\partial x} = 0$

Ex. $f(x) = x^2$

$f'(x) = 2x = 0$

$x = 0$ min

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

$\frac{\partial f}{\partial x} = 2x = 0$ / $\frac{\partial f}{\partial y} = 2y = 0$

$(x,y) = (0,0)$ min

Optimization with gradients

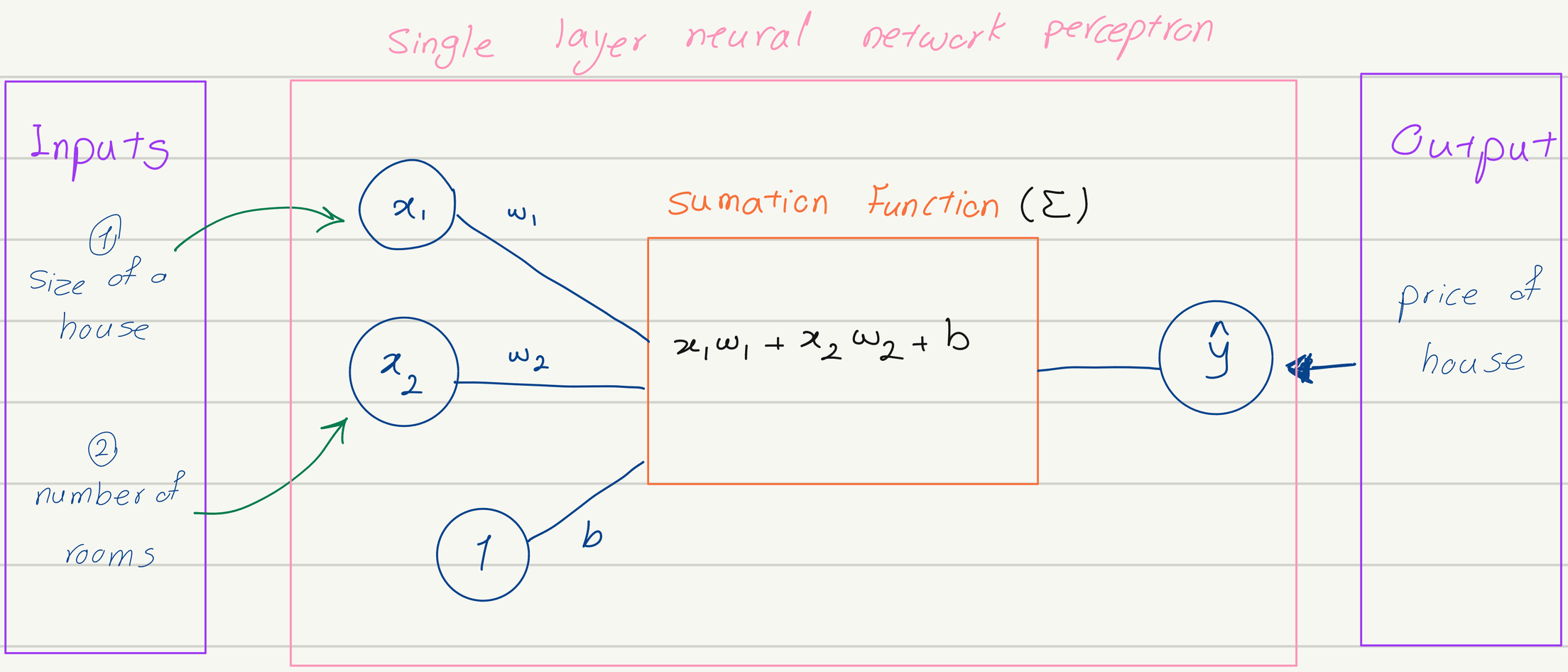
Week 3 : Optimization in Neural Networks and Newton Method

Regression with a perceptron

Features =

Ex. We want to predict the price of a house from 2 inputs:

1) the size of the house 2) number of rooms



* x_1 , x_2 : inputs / features * $\hat{y}$: output

* Σ : Sumation Function * 1 , b : A bias term

* w_1 , w_2 : weights that determine how important the input is for the output.

So: $\hat{y} = x_1 w_1 + x_2 w_2 + b$ ⟶ Goal: Find weights and bias that will optimise the predictions.

or: Reduce the errors in the predictions. → what's error ? The error is basically how far are you from the price of the house ?

Mean Squared Error

Error : subtract the prediction from the actual value

→ $y - \hat{y}$

actual ← y , $\hat{y}$ → prediction

⇒ The lesser the error , the bettey the model.

Error 's problem ? Errors can be positive or negetive so adding them can get us zero or small numbers that can get confusing . So for that we squar the Error.

Squared Error : $\frac{1}{2}(y - \hat{y})^2$

What is $\frac{1}{2}$? when we take the derivative of $(y - \hat{y})^2$, we get a lingering two . So we put $\frac{1}{2}$ there to cancel with that.

باقی ماندن

Loss function : for every point is : $L(y, \hat{y}) = \frac{1}{2}(y - \hat{y})^2$

⇒ Main Goal : Minimize the loss function

And as we said in last page : Find w_1, w_2, b that give $\hat{y}$ with the least error.

So to find optimal values for : w_1 , w_2 , b

→ we must use Gradient Descent.

learning rate

$$w_1 \longrightarrow w_1 - \alpha \frac{\partial L}{\partial w_1}$$

Loss function

$$w_2 \longrightarrow w_2 - \alpha \frac{\partial L}{\partial w_2}$$

$$b \longrightarrow b - \alpha \frac{\partial L}{\partial b}$$

$$w_1 \longrightarrow w_1 - \alpha(-x_1(y-\hat{y}))$$

$$w_2 \longrightarrow w_2 - \alpha(-x_2(y-\hat{y}))$$

$$b \longrightarrow b - \alpha(-(y-\hat{y}))$$

→ * we should do this many times *

Starting points

Find the following partial derivatives How?

Finding Partial Derivatives : using chain rule

Let's go * up and calculate gradient descent

■ Prediction Function:

$$\hat{y} = x_1 w_1 + x_2 w_2 + b$$

■ Loss Function:

$$L(y, \hat{y}) = \frac{1}{2}(y-\hat{y})^2$$

■ Partial Derivatives:

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

$$\frac{\partial L}{\partial w_1} = \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial w_1} = -(y-\hat{y})x_1$$

$$\frac{\partial L}{\partial w_2} = \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial w_2} = -(y-\hat{y})x_2$$

■ Calculation:

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

$$\frac{\partial \hat{y}}{\partial w_1} = x_1$$

$$\frac{\partial \hat{y}}{\partial b} = 1$$

$$\frac{\partial \hat{y}}{\partial w_2} = x_2$$

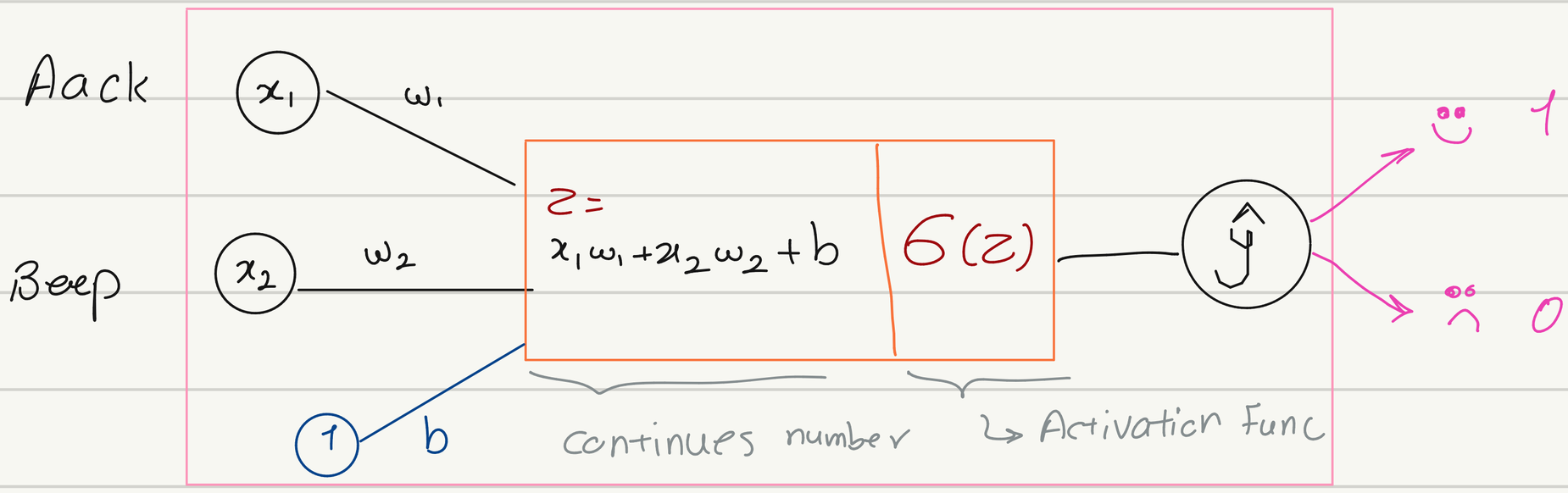
* Continue

Classification with a Perceptron

Example: Alien Speaking (Emotion)

Sentence	Aack	Beep	Mood
Aack Aack Aack!	3	0	Happy ☺
Beep Beep!	0	2	Sad ☹
Aack Beep Beep Beep!	1	3	Sad ☹
Aack Beep Aack!	2	1	Happy ☺

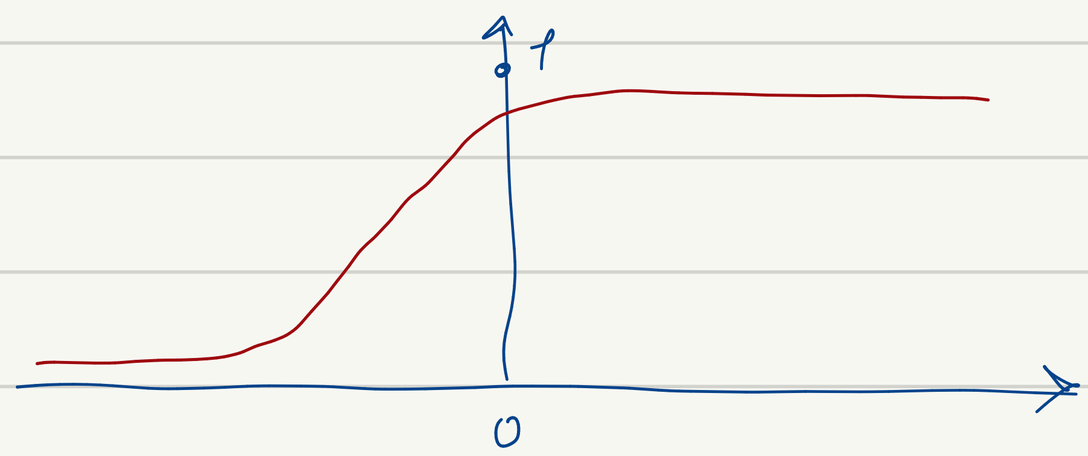
* Models take numbers not words.



$\sigma(z)$: Sigmoid of z. This Function is going to take all the numbers and crunch them into the interval 0, 1.

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

$\Rightarrow 0 < \sigma(z) < 1$



Derivative of a Sigmoid Function

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

So: $$\frac{d\sigma(z)}{dz} = \frac{d}{dz}(1+e^{-z})^{-1} = -1(1+e^{-z})^{-2}(e^{-z})(-1)$$

$$= \frac{1}{(1+e^{-z})^2} e^{-z} = \frac{e^{-z}+1-1}{(1+e^{-z})^2} = \frac{1+e^{-z}}{(1+e^{-z})^2} - \frac{1}{(1+e^{-z})^2}$$

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

$$= \sigma(z)\left(1-\sigma(z)\right)$$

* $\sigma(z) = \frac{1}{1+e^{-z}}$ $\xrightarrow{\text{derivative}}$ $\frac{d}{dz}\sigma(z) = \sigma(z)(1-\sigma(z))$ *

* The process and our goal is the same as Regression Except in 1) Activation Function and 2) Loss Function. In classification we use kind of Loss function called Log Loss:

- prediction Function:

$$\hat{y} = \sigma(w_1x_1 + w_2x_2 + b)$$

- Main Goal:

Find w_1, w_2, b that give $\hat{y}$ with the least error.

- Loss Function (Log Loss):

$$L(y,\hat{y}) = -y\ln(\hat{y}) - (1-y)\ln(1-\hat{y})$$

* $L(y,\hat{y})$ is a large number if y and $\hat{y}$ are far away from each other and small number if they are close to each other.

- Finding optimal values for w_1, w_2, b:

* Using Gradient Descent *

$$w_1 \longrightarrow w_1 - \alpha \frac{\partial L}{\partial w_1} \qquad w_1 \longrightarrow w_1 - \alpha\,(-x_1(y-\hat{y}))$$

$$w_2 \longrightarrow w_2 - \alpha \frac{\partial L}{\partial w_2} \qquad w_2 \longrightarrow w_2 - \alpha\,(-x_2(y-\hat{y}))$$

$$b \longrightarrow b - \alpha \frac{\partial L}{\partial b} \qquad b \longrightarrow b - \alpha\,(-(y-\hat{y}))$$

- calculating partial derivatives:

→ Prediction Function:

$$\hat{y} = \sigma(w_1 x_1 + w_2 x_2 + b)$$

→ Log Loss Function:

$$L(y, \hat{y}) = -y \ln(\hat{y}) - (1-y)\ln(1-\hat{y})$$

→ partial derivatives:

$$\frac{\partial L}{\partial w_1} = \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial w_1} = -(y-\hat{y})\,x_1$$

$$\frac{\partial L}{\partial w_2} = \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial w_2} = -(y-\hat{y})\,x_2$$

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

* Remember: $f(x) = \ln(x) \longrightarrow f'(x) = \frac{1}{x}$ / $\sigma(z) \longrightarrow \sigma'(z) = \sigma(z)(1-\sigma(z))$

→ calculating:

$$\frac{\partial L}{\partial \hat{y}} = \frac{-y}{\hat{y}} + \frac{1-y}{1-\hat{y}} = \frac{-y + y\hat{y} + \hat{y} - y\hat{y}}{\hat{y}(1-\hat{y})} = \frac{-(y-\hat{y})}{\hat{y}(1-\hat{y})}$$

$$\frac{\partial \hat{y}}{\partial w_1} = \hat{y}(1-\hat{y})\,x_1$$

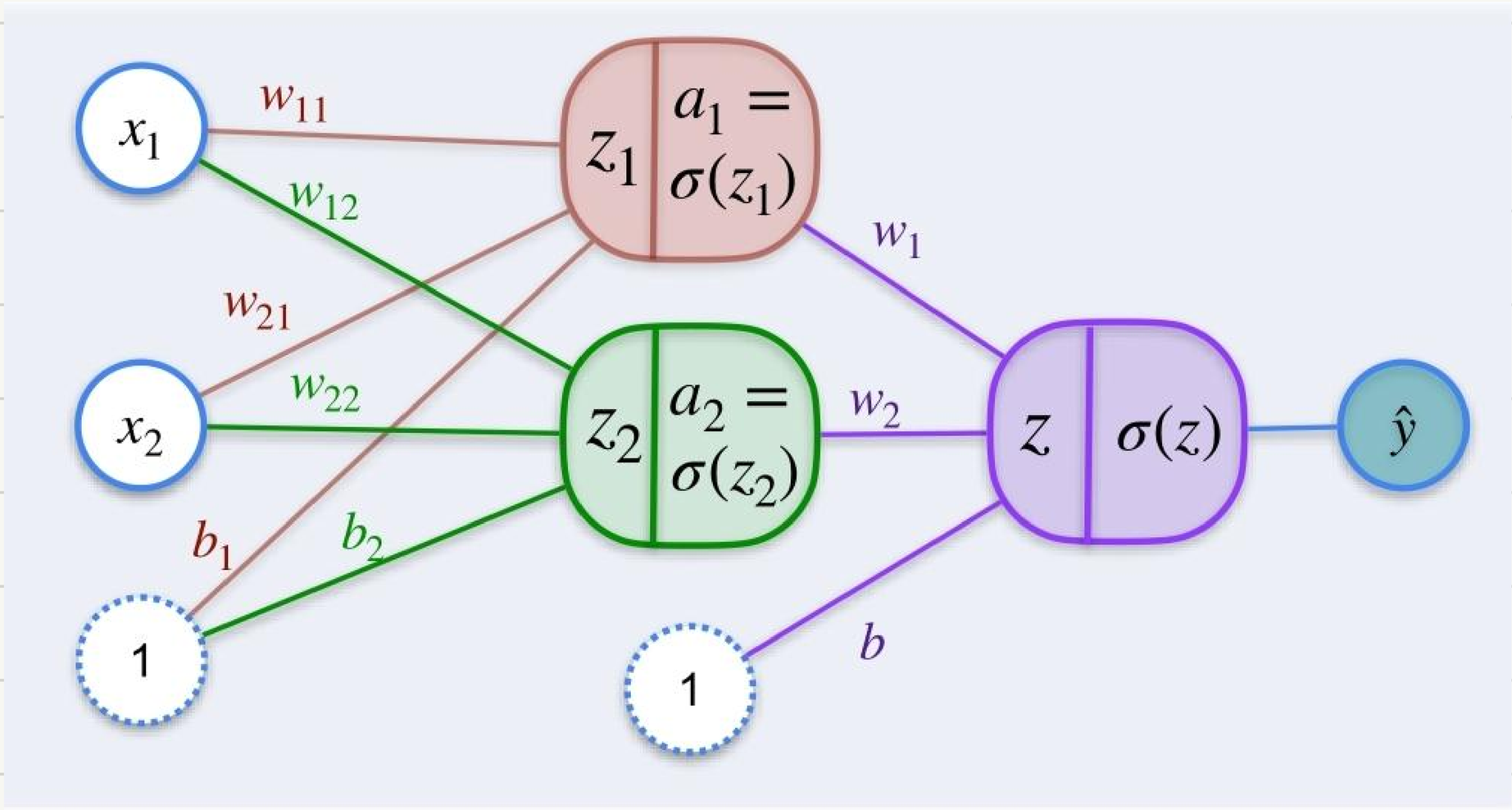
$$\frac{\partial \hat{y}}{\partial w_2} = \hat{y}(1-\hat{y})\,x_2$$

$$\frac{\partial \hat{y}}{\partial b} = \hat{y}(1-\hat{y})$$

* Derivative of the inside by w_1

let's multiply them.

2, 2, 1 Neural Network



Neural network of depth 2 :

- one input layer
- one hidden layer
- one output layer

$$a_1 = \sigma(z_1) \quad , \quad z_1 = x_1 w_{11} + x_2 w_{21} + b_1$$

$$a_2 = \sigma(z_2) \quad , \quad z_2 = x_1 w_{12} + x_2 w_{22} + b_2$$

$$\hat{y} = \sigma(z) \quad , \quad z = a_1 w_1 + a_2 w_2 + b$$

Log Loss : $L(y, \hat{y}) = -y \log(\hat{y}) - (1-y) \log(1-\hat{y})$

Goal : Adjust each of the weights and biases $\longrightarrow$ To reduce loss Function

How each one of the weights effects the Loss ?

for $w_{11}, w_{12}, w_{21}, w_{22}$ $\leftarrow \frac{\partial L}{\partial w_{ij}}$ / for w_1, w_2 $\leftarrow \frac{\partial L}{\partial w_i}$

: .. .. the biases .. .. .. ?

for b_1, b_2 $\leftarrow \frac{\partial L}{\partial b_i}$ / for b $\leftarrow \frac{\partial L}{\partial b}$

* using chain Rule.

$$L(y, \hat{y}) = -y\log(\hat{y}) - (1-y)\log(1-\hat{y})$$

$$\hat{y} = \sigma(z)$$

$$z = a_1 w_1 + a_2 w_2 + b$$

$$a_1 = \sigma(z_1)$$

$$z_1 = x_1 w_{11} + x_2 w_{21} + b_1$$

$$\frac{\partial L}{\partial w_{11}} = \frac{\partial z_1}{\partial w_{11}} \cdot \frac{\partial a_1}{\partial z_1} \cdot \frac{\partial z}{\partial a_1} \cdot \frac{\partial \hat{y}}{\partial z} \cdot \frac{\partial L}{\partial \hat{y}}$$

$$\frac{\partial L}{\partial w_{11}} = x_1 \cdot a_1(1-a_1) \cdot w_1 \cdot \cancel{\hat{y}(1-\hat{y})} \cdot \frac{-(y - \hat{y})}{\cancel{\hat{y}(1-\hat{y})}}$$

$$= -x_1 w_1 a_1 (1-a_1)(y - \hat{y})$$

Perform gradient descent with

$$w_{11} \rightarrow w_{11} - \alpha\text{-}x_1 w_1 a_1 (1-a_1)(y - \hat{y})$$

to find optimal value of w_{11} that gives the least error

$$L(y, \hat{y}) = -y\log(\hat{y}) - (1-y)\log(1-\hat{y})$$

$$\hat{y} = \sigma(z)$$

$$z = a_1 w_1 + a_2 w_2 + b$$

$$a_1 = \sigma(z_1)$$

$$z_1 = x_1 w_{11} + x_2 w_{21} + b_1$$

$$\frac{\partial L}{\partial w_{21}} = \frac{\partial z_1}{\partial w_{21}} \cdot \frac{\partial a_1}{\partial z_1} \cdot \frac{\partial z}{\partial a_1} \cdot \frac{\partial \hat{y}}{\partial z} \cdot \frac{\partial L}{\partial \hat{y}}$$

$$\frac{\partial L}{\partial w_{21}} = x_2 \cdot a_1(1-a_1) \cdot w_1 \cdot \cancel{\hat{y}(1-\hat{y})} \cdot \frac{-(y - \hat{y})}{\cancel{\hat{y}(1-\hat{y})}}$$

$$= -x_2 w_1 a_1 (1-a_1)(y - \hat{y})$$

Perform gradient descent with

$$w_{21} \rightarrow w_{21} - \alpha\text{-}x_2 w_1 a_1 (1-a_1)(y - \hat{y})$$

to find optimal value of w_{21} that gives the least error

$$L(y, \hat{y}) = -y\log(\hat{y}) - (1-y)\log(1-\hat{y})$$

$$\hat{y} = \sigma(z)$$

$$z = a_1 w_1 + a_2 w_2 + b$$

$$a_1 = \sigma(z_1)$$

$$z_1 = x_1 w_{11} + x_2 w_{21} + b_1$$

$$\frac{\partial L}{\partial b_1} = \frac{\partial z_1}{\partial b_1} \cdot \frac{\partial a_1}{\partial z_1} \cdot \frac{\partial z}{\partial a_1} \cdot \frac{\partial \hat{y}}{\partial z} \cdot \frac{\partial L}{\partial \hat{y}}$$

$$\frac{\partial L}{\partial b_1} = 1 \cdot a_1(1-a_1) \cdot w_1 \cdot \cancel{\hat{y}(1-\hat{y})} \cdot \frac{-(y - \hat{y})}{\cancel{\hat{y}(1-\hat{y})}}$$

$$= -w_1 a_1 (1-a_1)(y - \hat{y})$$

Perform gradient descent with

$$b_1 \rightarrow b_1 - \alpha(-w_1 a_1 (1-a_1)(y - \hat{y}))$$

to find optimal value of b_1 that gives the least error

In Summary (layer 1):

$$w_{11} \longrightarrow w_{11} + \alpha x_1 w_1 a_1 (1 - a_1)(y - \hat{y})$$

$$w_{12} \longrightarrow w_{11} + \alpha x_2 w_1 a_1 (1 - a_1)(y - \hat{y})$$

$$w_{21} \longrightarrow w_{21} + \alpha x_1 w_2 a_2 (1 - a_2)(y - \hat{y})$$

$$w_{22} \longrightarrow w_{22} + \alpha x_2 w_2 a_2 (1 - a_2)(y - \hat{y})$$

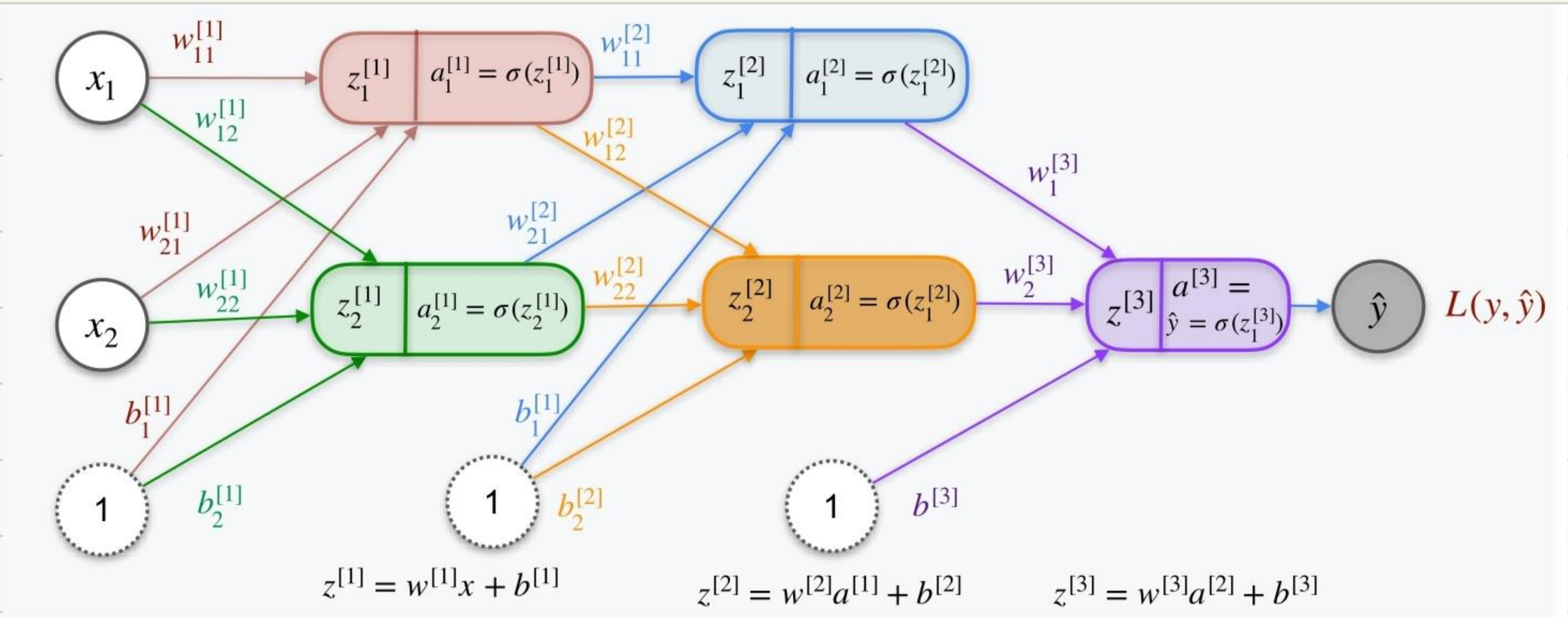
$$b_1 \longrightarrow b_1 + \alpha w_1 a_1 (1 - a_1)(y - \hat{y}) \quad , \quad b_2 \longrightarrow b_2 + \alpha w_2 a_2 (1 - a_2)(y - \hat{y})$$

Update Second Layer:

$$w_1 \longrightarrow w_1 + \alpha a_1 (y - \hat{y}) \quad , \quad w_2 \longrightarrow w_2 + \alpha a_2 (y - \hat{y})$$

$$b \longrightarrow b + a (y - \hat{y})$$

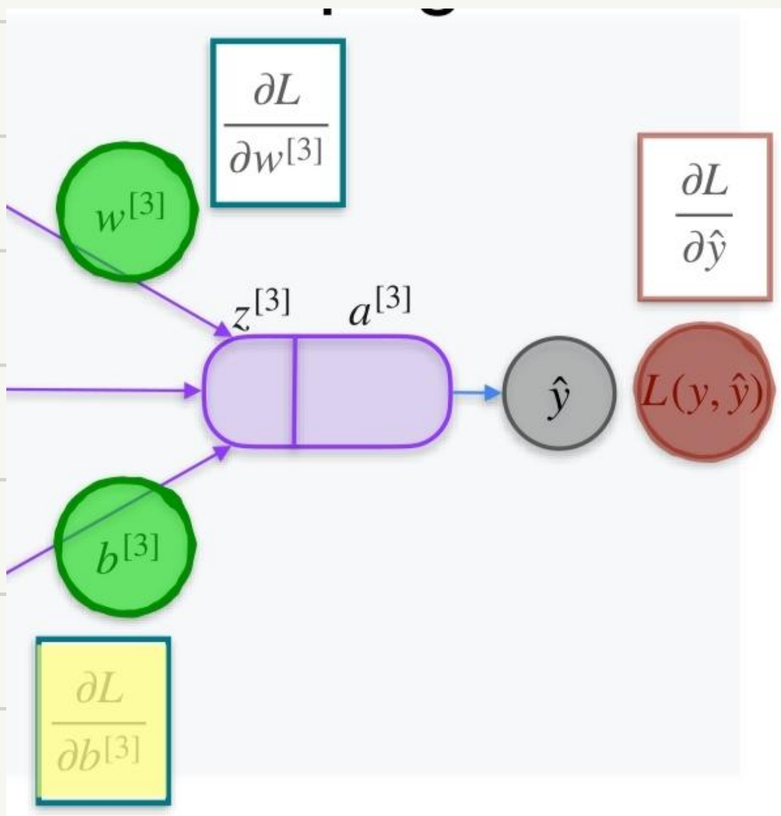
- Training Model -



$$L(y, \hat{y}) = -y\log(\hat{y}) - (1 - y)\log(1 - \hat{y})$$

we want to minimize

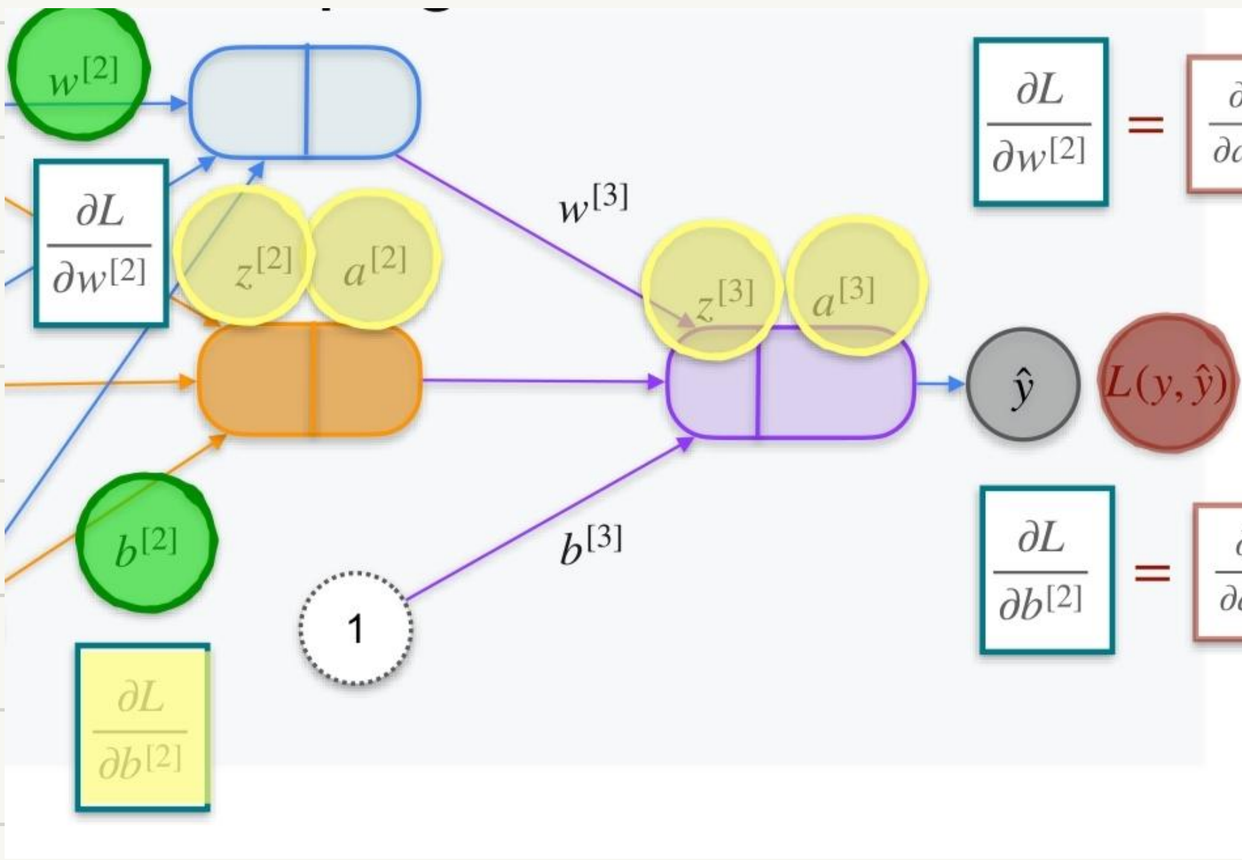
- **Back Propagation** : is a gradient estimation method commonly used for training neural networks to compute the network parameter updates.



$$\frac{\partial L}{\partial w^{[3]}} = \frac{\partial z^{[3]}}{\partial w^{[3]}} \cdot \frac{\partial a^{[3]}}{\partial z^{[3]}} \cdot \frac{\partial L}{\partial a^{[3]}}$$

$$\frac{\partial L}{\partial b^{[3]}} = \frac{\partial z^{[3]}}{\partial b^{[3]}} \cdot \frac{\partial a^{[3]}}{\partial z^{[3]}} \cdot \frac{\partial L}{\partial a^{[3]}}$$

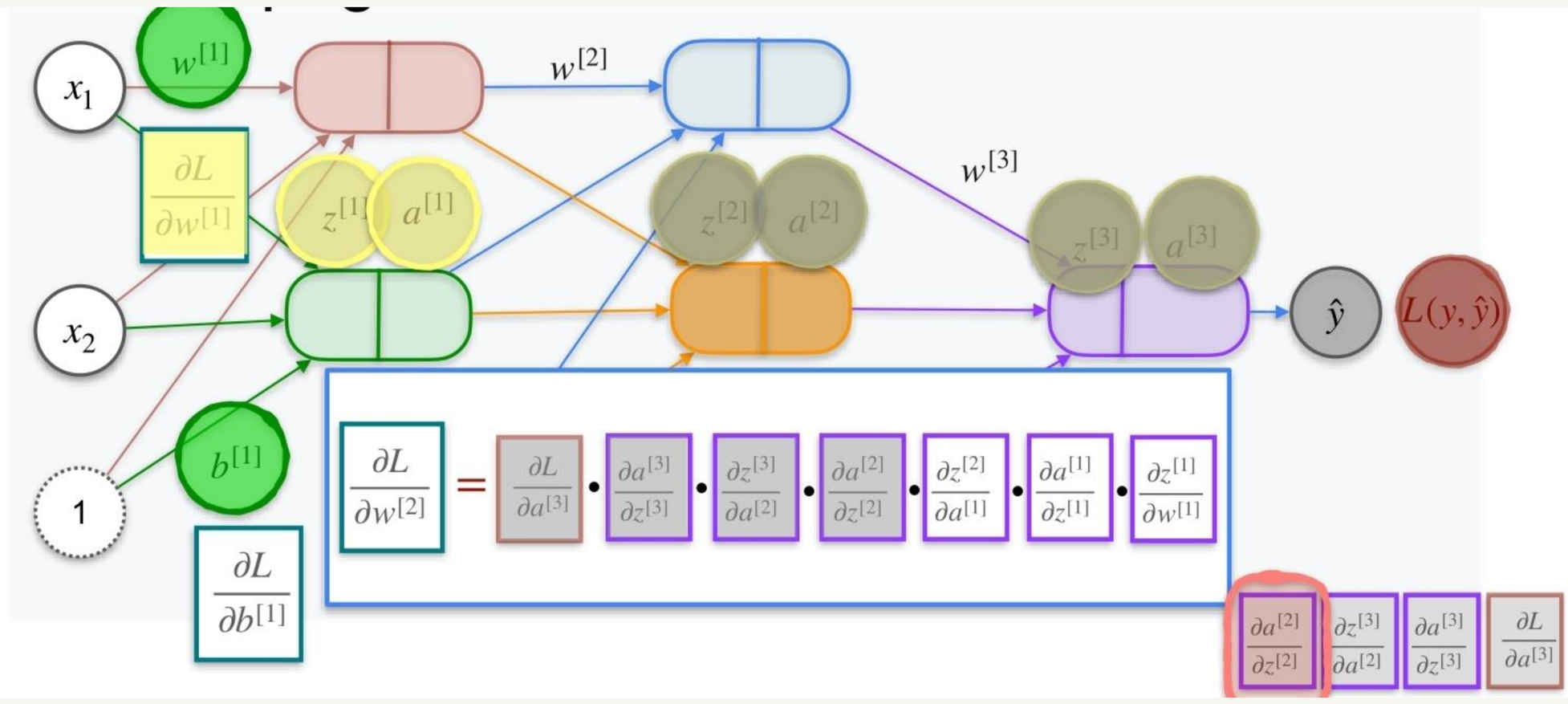
$\frac{\partial a^{[3]}}{\partial z^{[3]}}$ $\frac{\partial L}{\partial a^{[3]}}$



$$\frac{\partial L}{\partial w^{[2]}} = \frac{\partial L}{\partial a^{[3]}} \cdot \frac{\partial a^{[3]}}{\partial z^{[3]}} \cdot \frac{\partial z^{[3]}}{\partial a^{[2]}} \cdot \frac{\partial a^{[2]}}{\partial z^{[2]}} \cdot \frac{\partial z^{[2]}}{\partial w^{[2]}}$$

$$\frac{\partial L}{\partial b^{[2]}} = \frac{\partial L}{\partial a^{[3]}} \cdot \frac{\partial a^{[3]}}{\partial z^{[3]}} \cdot \frac{\partial z^{[3]}}{\partial a^{[2]}} \cdot \frac{\partial a^{[2]}}{\partial z^{[2]}} \cdot \frac{\partial z^{[2]}}{\partial b^{[2]}}$$

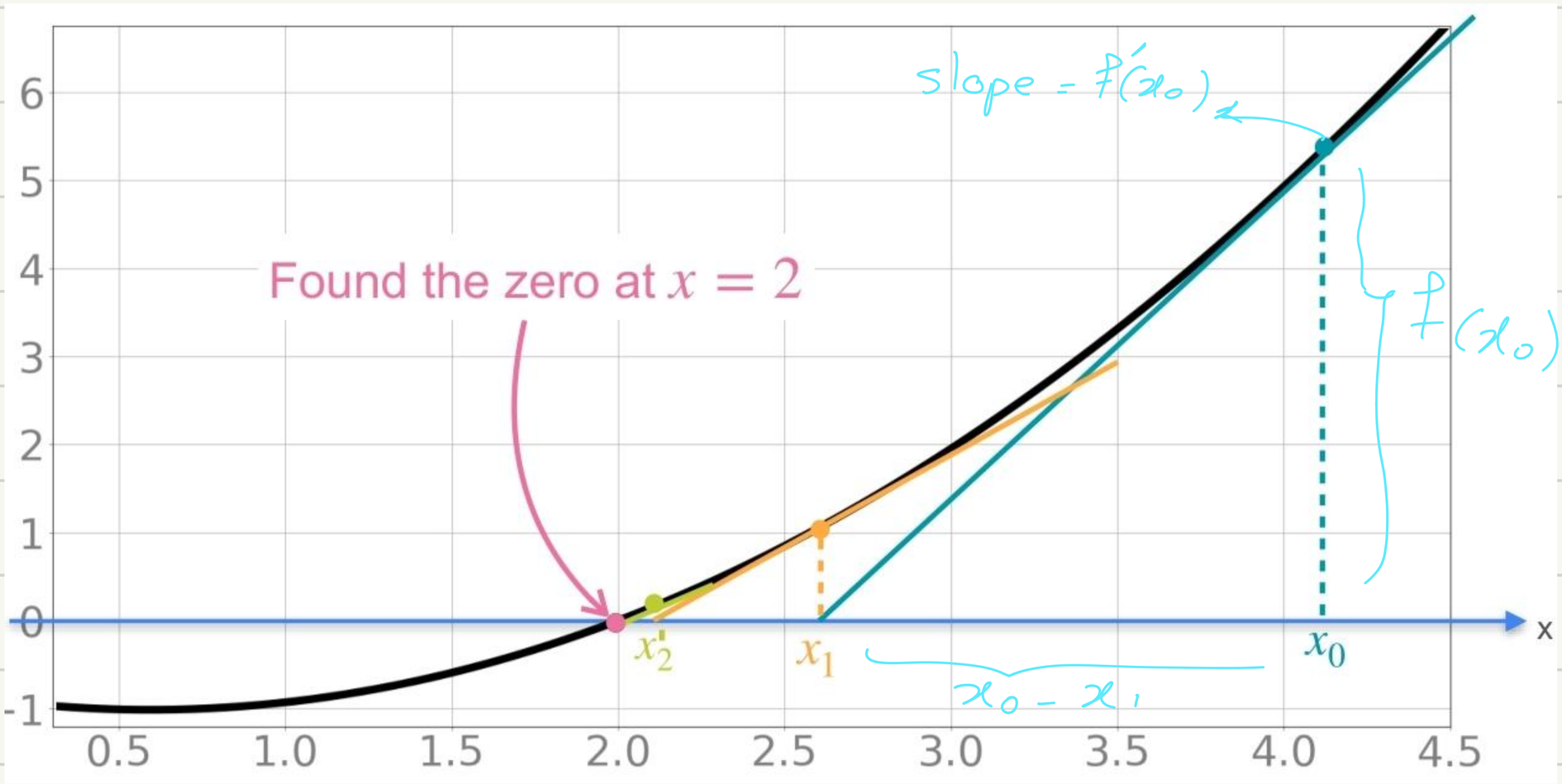
$\frac{\partial a^{[3]}}{\partial z^{[3]}}$ $\frac{\partial L}{\partial a^{[3]}}$



$$\frac{\partial L}{\partial w^{[2]}} = \frac{\partial L}{\partial a^{[3]}} \cdot \frac{\partial a^{[3]}}{\partial z^{[3]}} \cdot \frac{\partial z^{[3]}}{\partial a^{[2]}} \cdot \frac{\partial a^{[2]}}{\partial z^{[2]}} \cdot \frac{\partial z^{[2]}}{\partial a^{[1]}} \cdot \frac{\partial a^{[1]}}{\partial z^{[1]}} \cdot \frac{\partial z^{[1]}}{\partial w^{[1]}}$$

$\frac{\partial a^{[2]}}{\partial z^{[2]}}$ $\frac{\partial z^{[3]}}{\partial a^{[2]}}$ $\frac{\partial a^{[3]}}{\partial z^{[3]}}$ $\frac{\partial L}{\partial a^{[3]}}$

Newton's Method



$$\frac{f(x_0)}{x_0 - x_1} = f'(x_0) \longrightarrow x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}$$

So in general : $x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$

Newton's Method for Optimization

Newton's Method

Goal: find a zero of $f(x)$

1) Start with some x_0

2) Update :

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$$

3) Repeat 2 until you find the root

NM for Optimization

Goal: minimize $g(x)$ → find zeros of $g'(x)$

$f(x) \longmapsto g'(x)$, $f'(x) \longmapsto (g'(x))'$

1) Start with some x_0

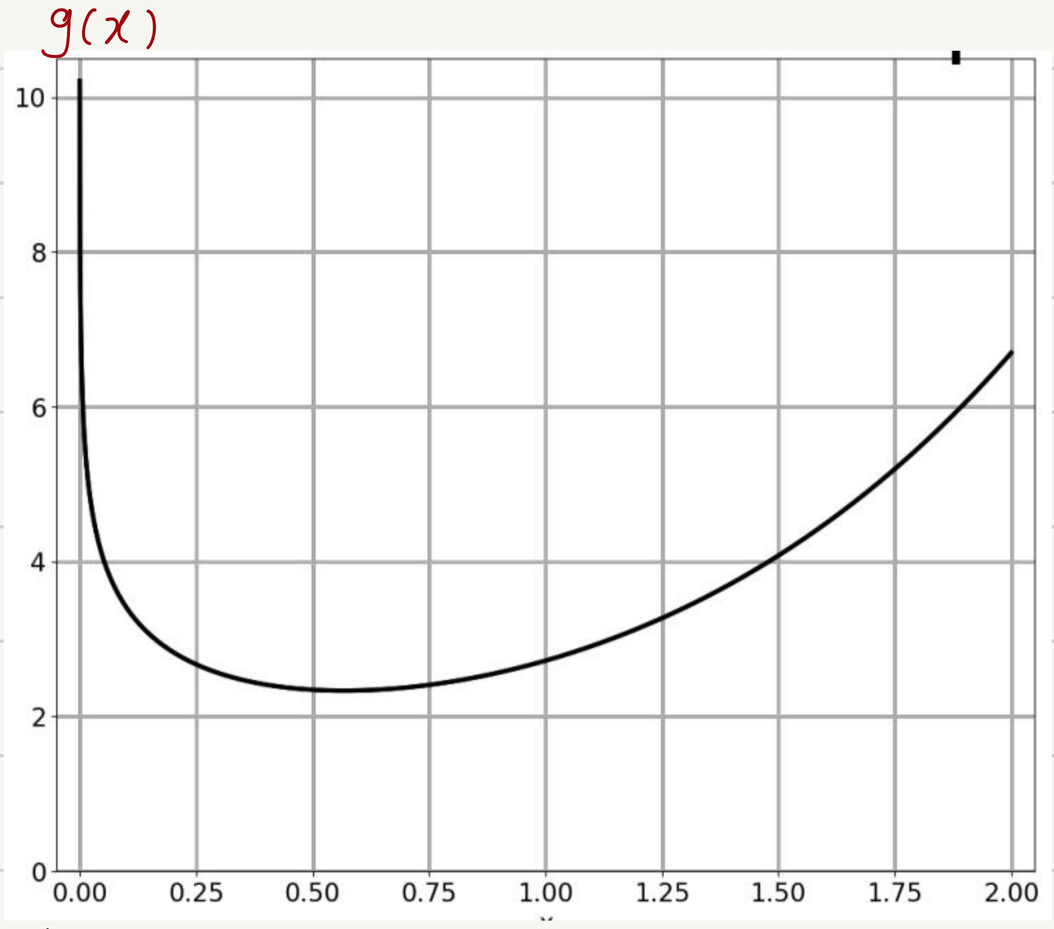
2) Update :

$$x_{k+1} = x_k - \frac{g'(x_k)}{(g'(x_k))'}$$

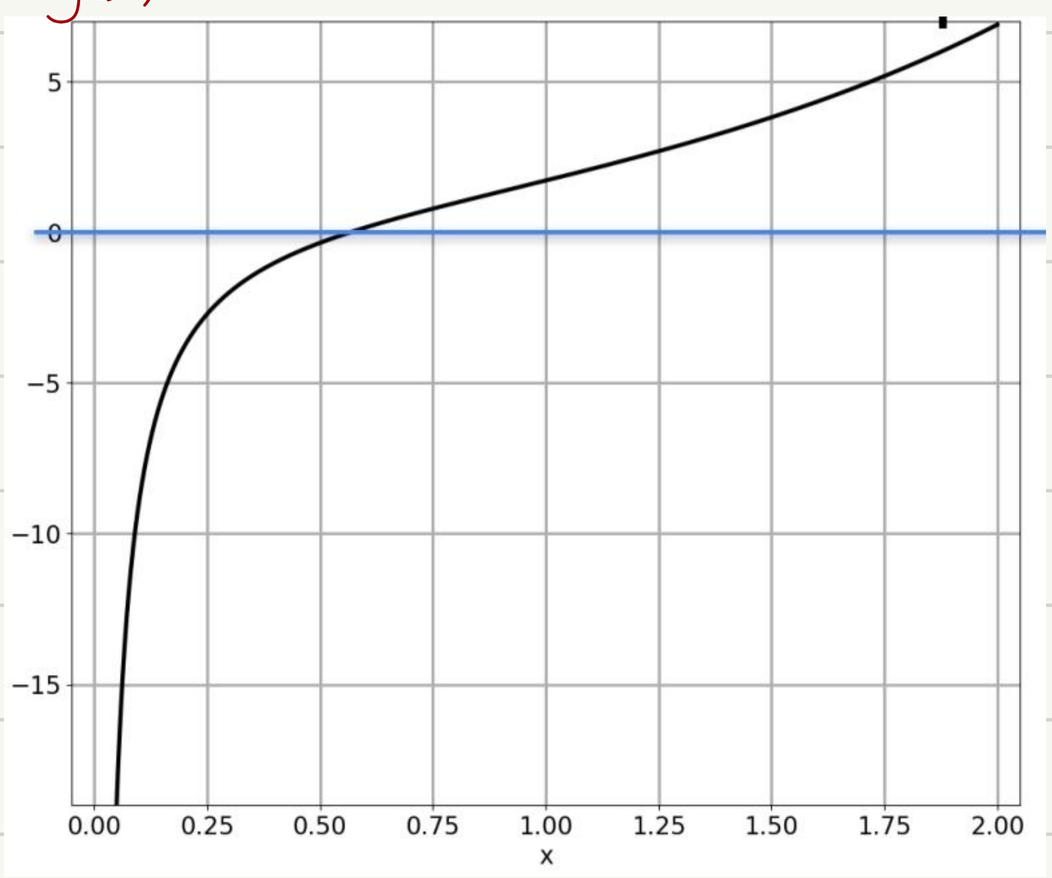
3) Repeat 2 until you find the root

Example : $g(x) = e^x - \log(x)$, $g'(x) = e^x - \frac{1}{x} = f(x)$

$g(x)$

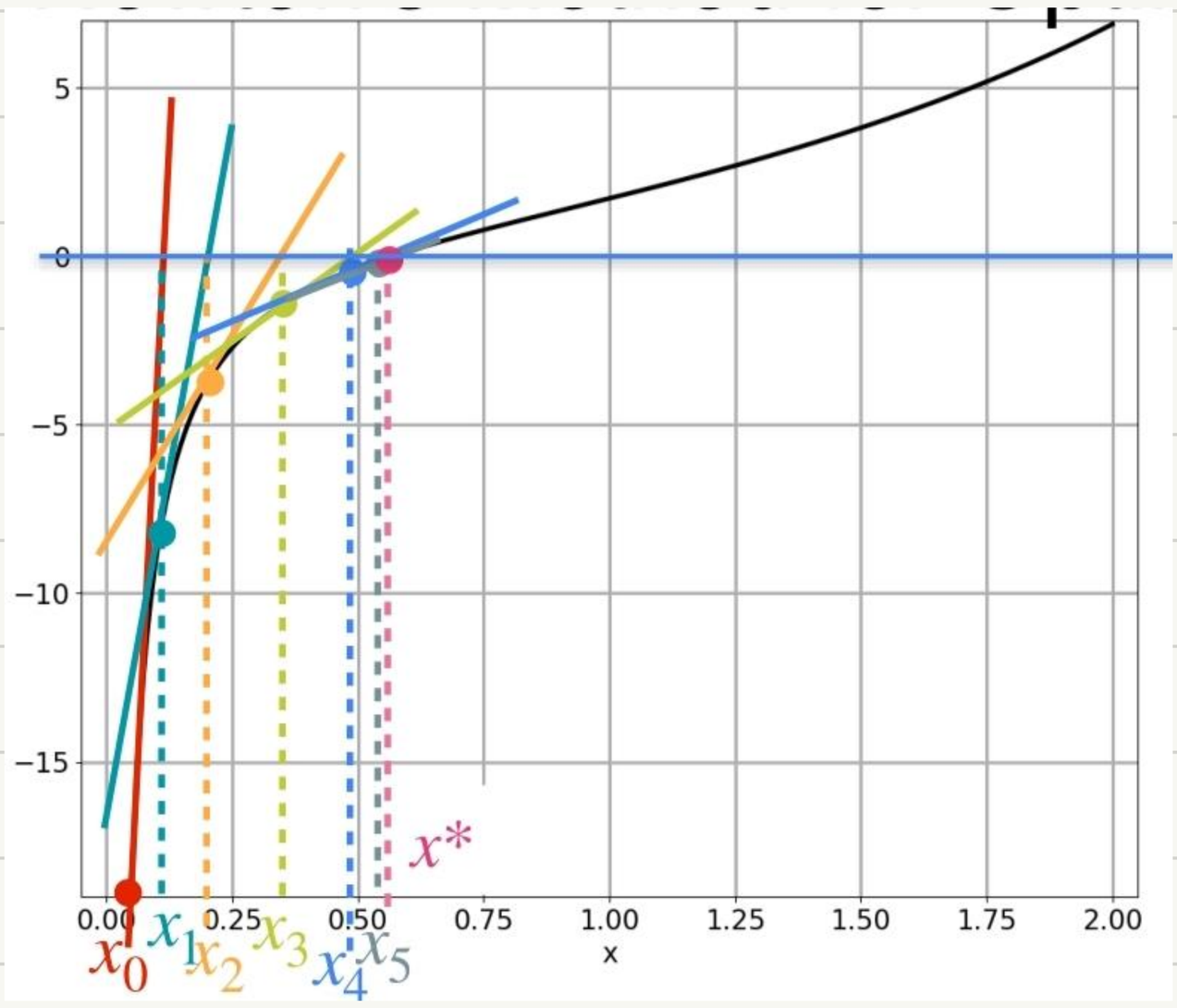


$g'(x)$



→ Min $\Rightarrow x^* = 0.5671$

→ Lets Start : $f'(x) = (g'(x))' = e^x + \frac{1}{x^2}$



$x_0 = 0.05$

$x_1 = x_0 - \frac{g'(x_0)}{(g'(x_0))'} = 0.097$

$x_2 = x_1 - \frac{g'(x_1)}{(g'(x_1))'} = 0.183$

$x_3 = x_2 - \frac{g'(x_2)}{(g'(x_2))'} = 0.320$

$x_4 = x_3 - \frac{g'(x_3)}{(g'(x_3))'} = 0.477$

$x_5 = x_4 - \frac{g'(x_4)}{(g'(x_4))'} = 0.558$

$x_6 = x^* = x_5 - \frac{g'(x_5)}{(g'(x_5))'} = 0.567$

min point of $g(x)$

Second Derivative → used for one variable

Newton's Method: $x_{k+1} = x_k - \frac{g'(x_k)}{(g'(x_k))'}$

$(g'(x_k))'$ → Second derivative

Leibniz notation: $\frac{d^2f(x)}{dx^2} = \frac{d}{dx}\left(\frac{df(x)}{dx}\right)$

Lagrange notation: $f''(x)$

Understanding Second Derivative

$x \longrightarrow$ Distance

$v \longrightarrow$ Velocity $\longrightarrow \frac{dx}{dt}$

$a \longrightarrow$ Acceleration $\longrightarrow \frac{dv}{dt} = \frac{d^2x}{dt^2}$ (second derivative)

Curvature

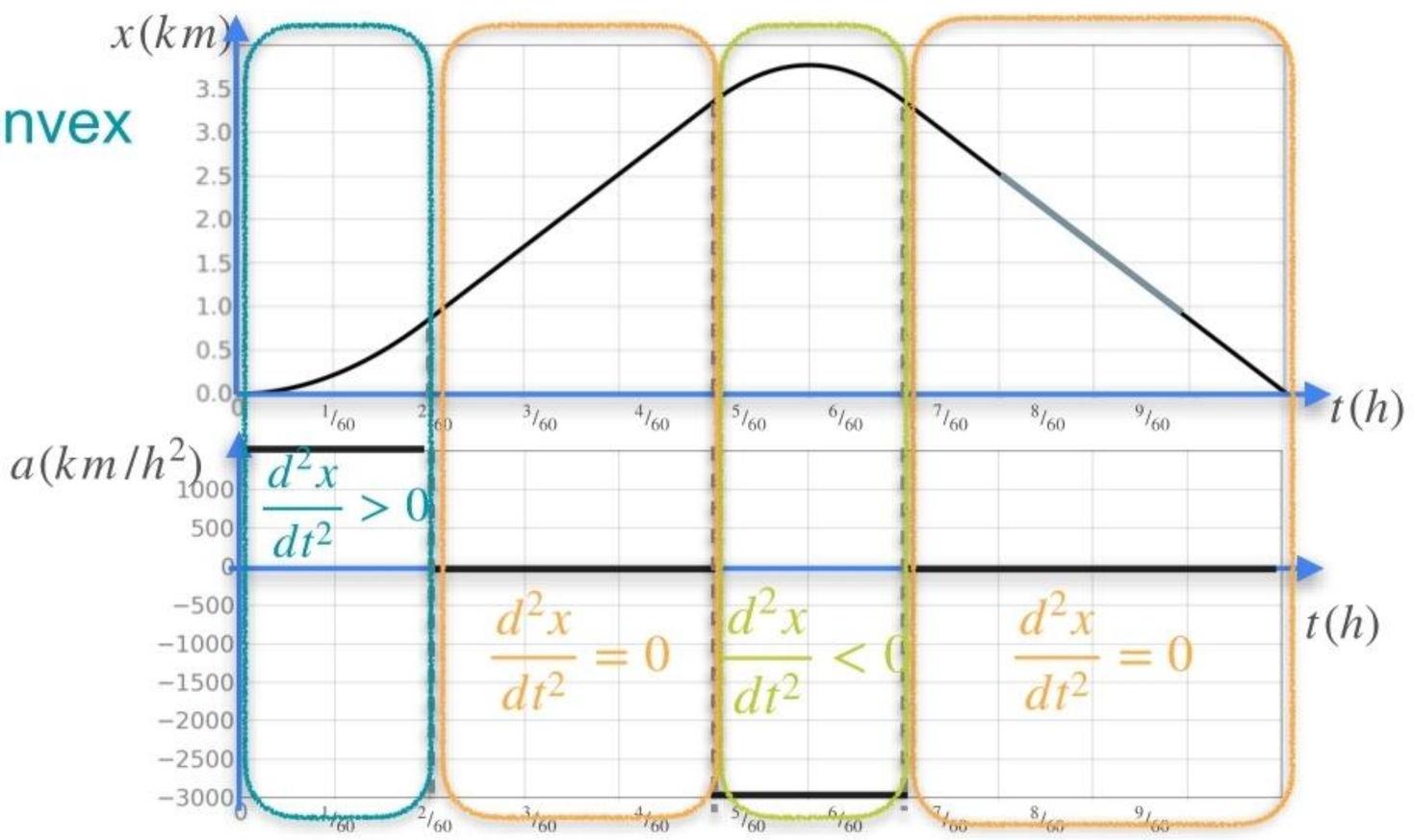
$\frac{d^2x}{dt^2} > 0$ Concave up or convex

$f''(0) > 0$

$\frac{d^2x}{dt^2} < 0$ Concave down

$f''(0) < 0$

$\frac{d^2x}{dt^2} = 0$ Need more information

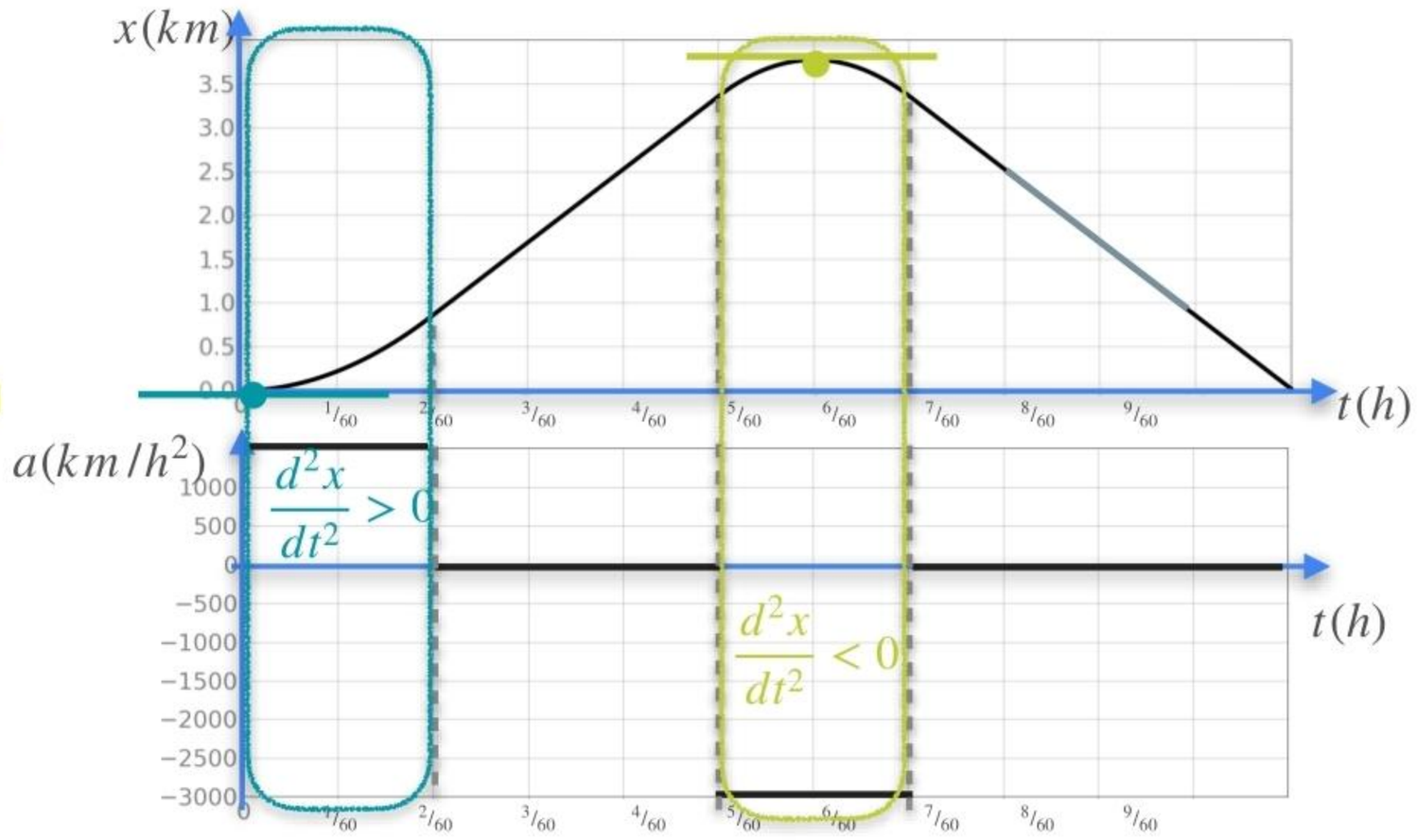


Second Derivative and Optimization

$\frac{d^2x}{dt^2} > 0$ (Local) Minimum

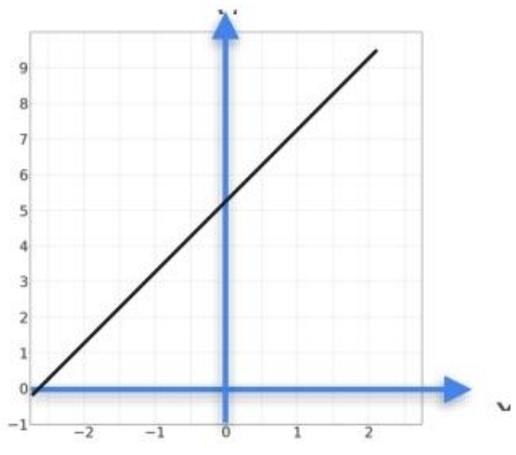
$\frac{d^2x}{dt^2} < 0$ (Local) maximum

$\frac{d^2x}{dt^2} = 0$ Inconclusive



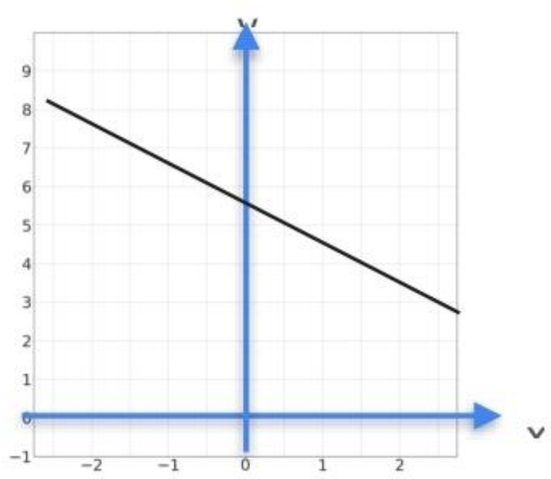
* Curvature *

First derivative



Increasing

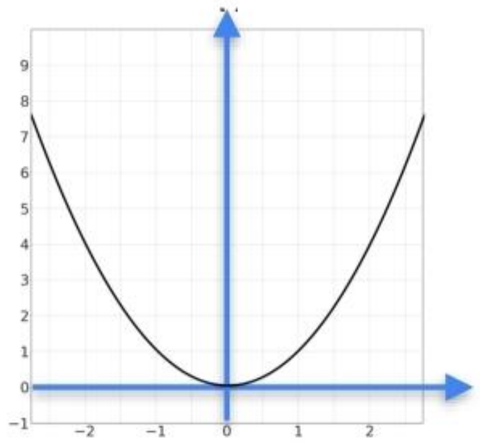
$f'(0) > 0$



Decreasing

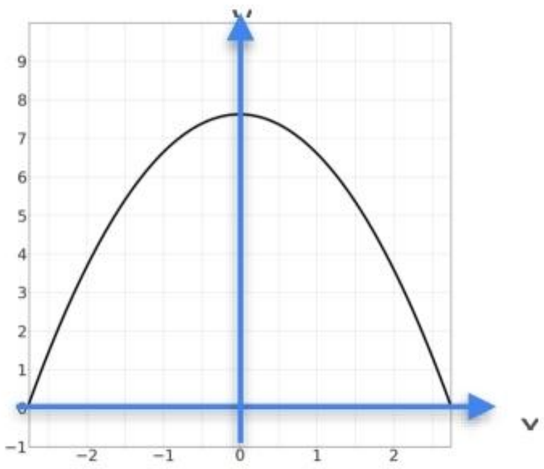
$f'(0) < 0$

Second derivative



Concave up

$f''(0) > 0$



Concave down

$f''(0) < 0$

- The Hessian (هِسْيَن) → used for multiple variables

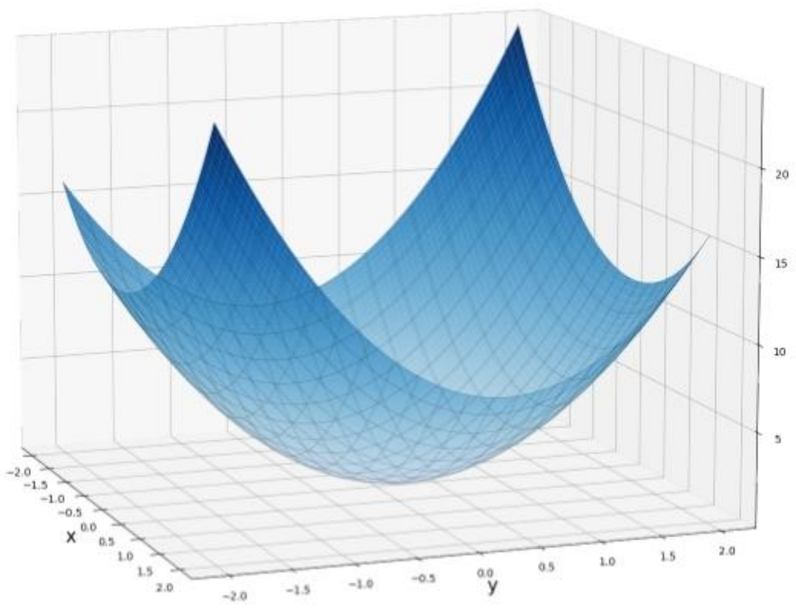
Second Derivative

	1 variable		2 variables	
Function	$f(x)$		$f(x, y)$	
First derivative	$f'(x)$	Rate of change of $f(x)$	$f_x(x, y)$ $f_y(x, y)$	Rate of change w.r.t x Rate of change w.r.t y
Second derivative	$f''(x)$	Rate of change of the rate of change of $f(x)$	???	$H = \begin{bmatrix} f_{xx}(x,y) & f_{xy}(x,y) \\ f_{yx}(x,y) & f_{yy}(x,y) \end{bmatrix}$

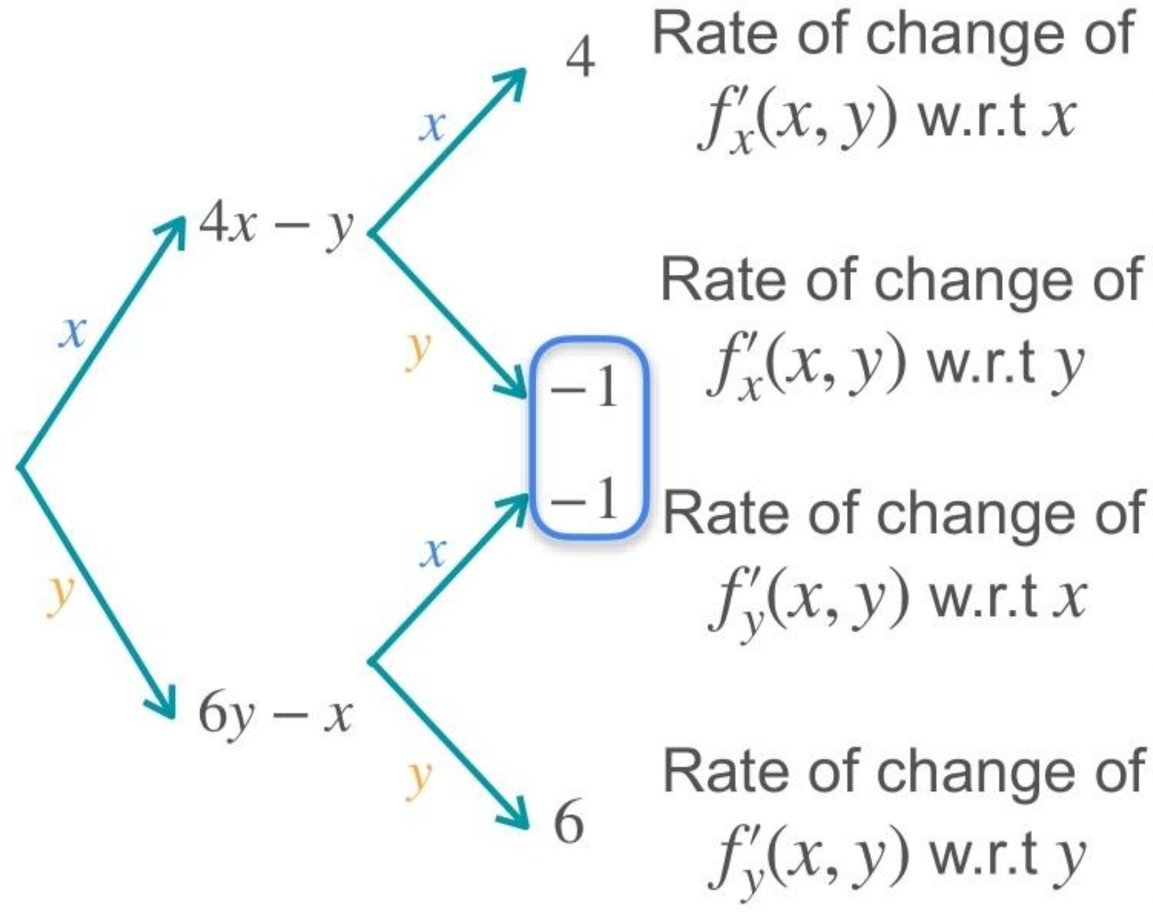
$$\nabla f = \begin{bmatrix} f_x(x, y) \\ f_y(x, y) \end{bmatrix}$$

Hessian Matrix

Example:



$$f(x, y) = 2x^2 + 3y^2 - xy$$



What does this mean?

Rate of change of $f_x(x, y)$ w.r.t x

Rate of change of $f_y(x, y)$ w.r.t y

Change in the change in the function w.r.t tiny changes in x and y

Same idea as with one variable!

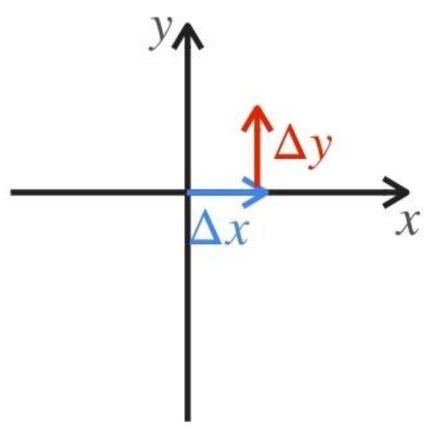
Rate of change of $f_x(x, y)$ w.r.t y

Rate of change of $f_y(x, y)$ w.r.t x

1. Change in the slope along one coordinate axis w.r.t tiny changes along an orthogonal coordinate axis

2. They are the same!

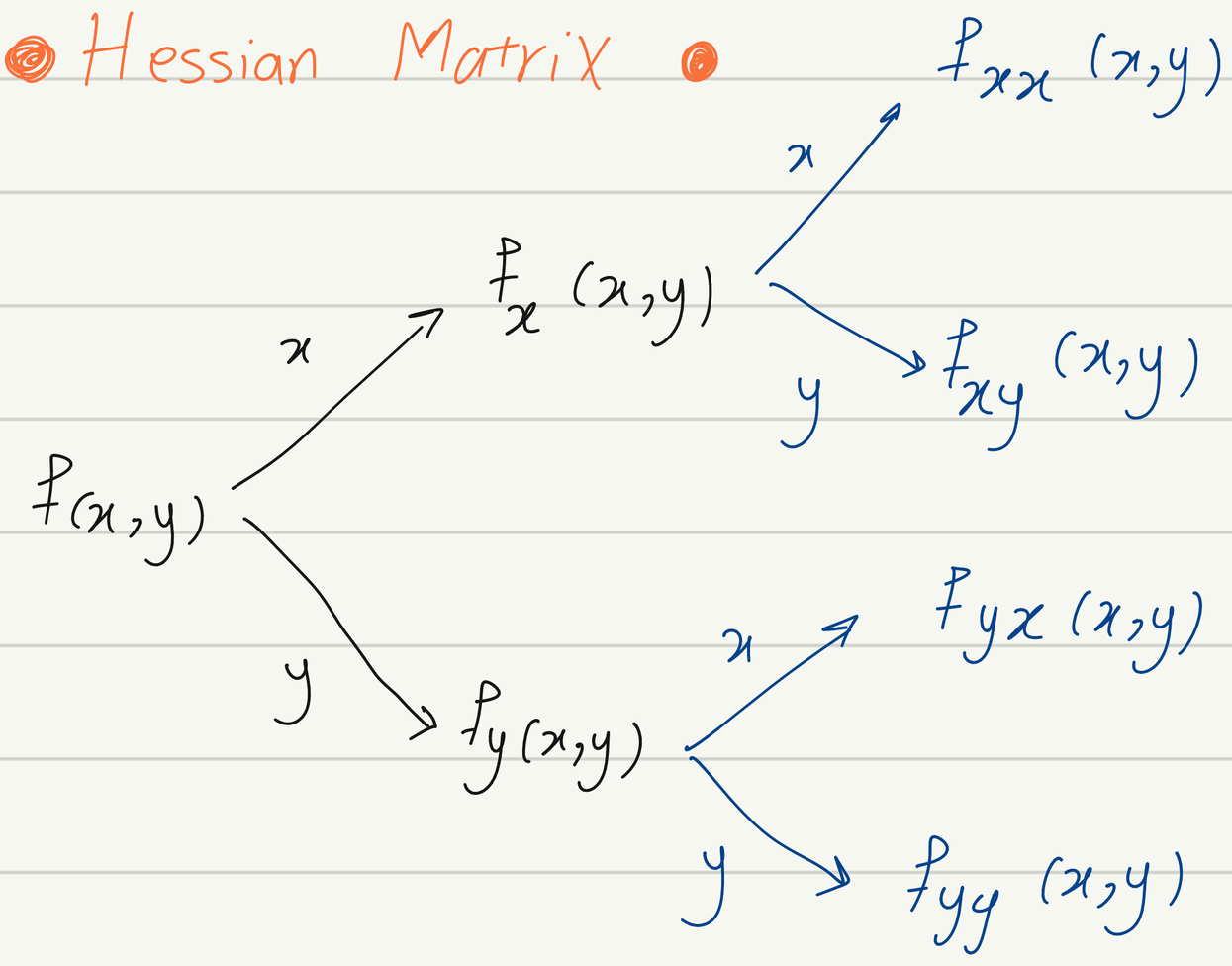
(In most cases)



✳ Notation ✳

	Leibinz's notation	Lagrange's notation
Rate of change of $f'_x(x,y)$ w.r.t x	$\frac{\partial^2 f}{\partial x^2}$	$f_{xx}(x,y)$
Rate of change of $f'_y(x,y)$ w.r.t y	$\frac{\partial^2 f}{\partial y^2}$	$f_{yy}(x,y)$
Rate of change of $f'_x(x,y)$ w.r.t y	$\frac{\partial^2 f}{\partial x \partial y}$	$f_{xy}(x,y)$
Rate of change of $f'_y(x,y)$ w.r.t x	$\frac{\partial^2 f}{\partial y \partial x}$	$f_{yx}(x,y)$

● Hessian Matrix ●



$$\nabla f_x = \begin{bmatrix} f_{xx}(x,y) \\ f_{xy}(x,y) \end{bmatrix}$$

$$\nabla f_y = \begin{bmatrix} f_{yx}(x,y) \\ f_{yy}(x,y) \end{bmatrix}$$

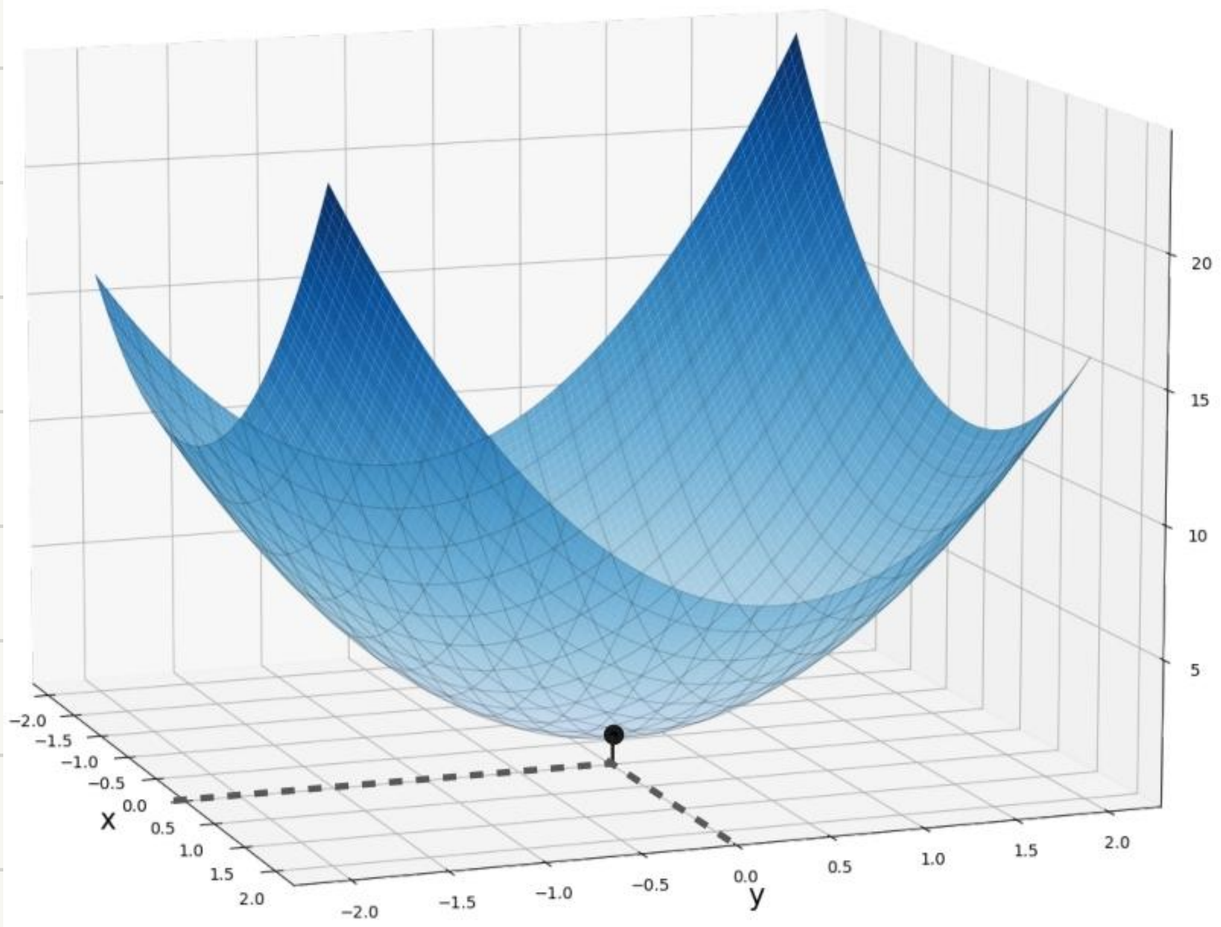
$$H = \begin{bmatrix} f_{xx}(x,y) & f_{xy}(x,y) \\ f_{yx}(x,y) & f_{yy}(x,y) \end{bmatrix} = \begin{bmatrix} \nabla f_x^T \\ \nabla f_y^T \end{bmatrix}$$

Hessian Matrix : All information about second derivatives

Back to previous page ↵

• Hessians and Concavity •

1) Concave up : both of eigen values are positive



$$f(x,y) = 2x^2 + 3y^2 - xy$$

$$H(0,0) = \begin{bmatrix} 4 & -1 \\ -1 & 6 \end{bmatrix}$$

$$\det(H(0,0) - \lambda I) = \det\left(\begin{bmatrix} 4-\lambda & -1 \\ -1 & 6-\lambda \end{bmatrix}\right)$$

$$= (4-\lambda)(6-\lambda) - (-1)(-1)$$

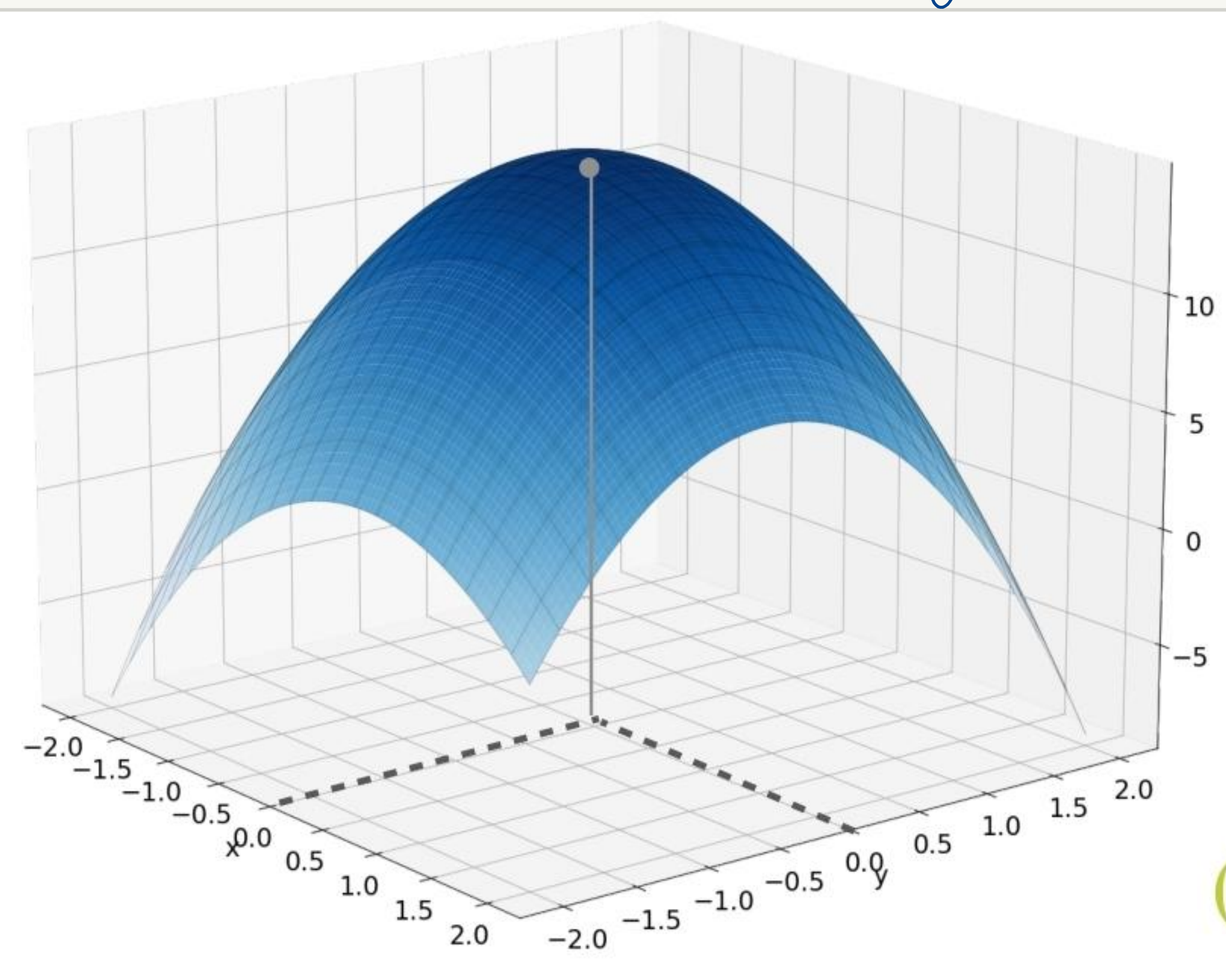
$$= \lambda^2 - 10\lambda + 23$$

$$\lambda_1 = 6.41$$
$$\lambda_2 = 3.59$$

> 0

(0,0) is a minimum!

2) Concave Down : both of eigen values are negative



$$f(x,y) = -2x^2 + -3y^2 - xy + 15$$

$$\nabla f(x,y) = \begin{bmatrix} -4x - y \\ -x - 6y \end{bmatrix}$$

$$H(0,0) = \begin{bmatrix} -4 & -1 \\ -1 & -6 \end{bmatrix}$$

$$\det(H(0,0) - \lambda I) =$$

$$(-4-\lambda)(-6-\lambda) - (-1)(-1)$$

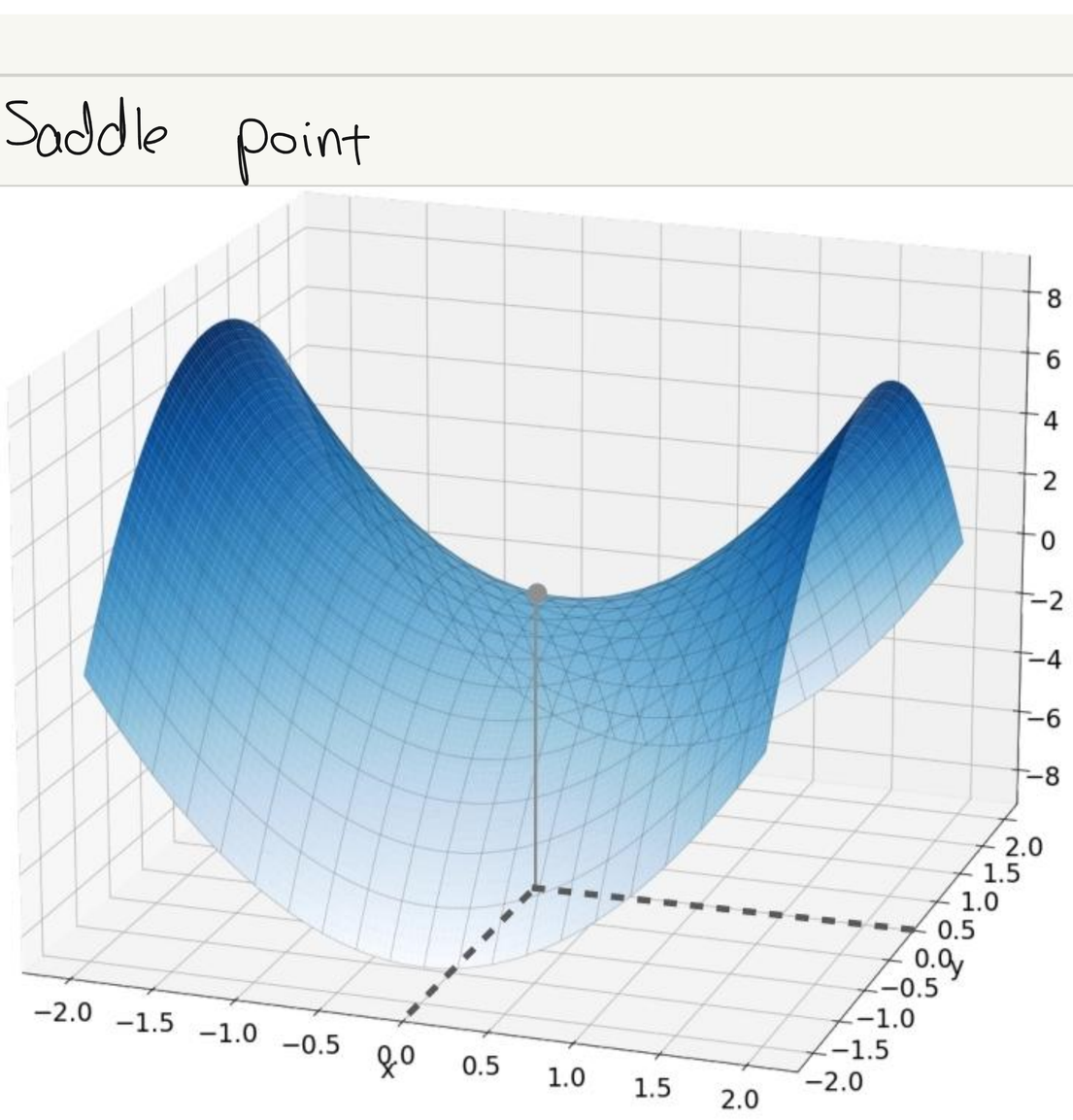
$$= \lambda^2 + 10\lambda + 23$$

$$\lambda_1 = -3.49$$
$$\lambda_2 = -6.41$$

< 0

(0,0) is a maximum!

3) Saddle point



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

$$\nabla f(x,y) = \begin{bmatrix} 4x \\ -4y \end{bmatrix}$$

$$H(0,0) = \begin{bmatrix} 4 & 0 \\ 0 & -4 \end{bmatrix}$$

$$\det(H(0,0) - \lambda I) =$$

$$(4-\lambda)(-4-\lambda) - 0$$

< 0

$$\lambda_1 = -4$$
$$\lambda_2 = 4$$

> 0

(0,0) is saddle point

Summary

	1 variable $f(x)$	2 variables $f(x,y)$	More variables $f(x_1, x_2, \ldots, x_n)$
(Local) minima	Happy face $f''(x) > 0$	Upper paraboloid $\lambda_1 > 0 \,\&\, \lambda_2 > 0$	All $\lambda_i > 0$
(Local) maxima	Sad face $f''(x) < 0$	Down paraboloid $\lambda_1 < 0 \,\&\, \lambda_2 < 0$	All $\lambda_i < 0$
Need more information	$f''(x) = 0$	Saddle point $\lambda_1 > 0 \,\&\, \lambda_2 < 0$ $\lambda_1 < 0 \,\&\, \lambda_2 > 0$ Or some $\lambda_i = 0$	Some $\lambda_i > 0$ and some $\lambda_j < 0$ OR At least one $\lambda_i = 0$

Newton's Method

1 variable : $$x_{k+1} = x_k - \frac{f'(x_k)}{(f'(x_k))'}$$

$$x_{k+1} = x_k - f''(x_k)^{-1} \cdot f'(x_k)$$

2 variables : $$\begin{bmatrix} x_{k+1} \\ y_{k+1} \end{bmatrix} = \begin{bmatrix} x_k \\ y_k \end{bmatrix} - H^{-1}(x_k, y_k)\, \nabla f(x_k, y_k)$$

$H^{-1}(x_k, y_k)$: 2×2 ; $\nabla f(x_k, y_k)$: 2×1

So: changing the position is wrong : ~~$\nabla f(x,y) \cdot H^{-1}(x,y)$~~