Last Class ⟶ Optimizers ADAM, RMSPROP

Today's Agenda:-

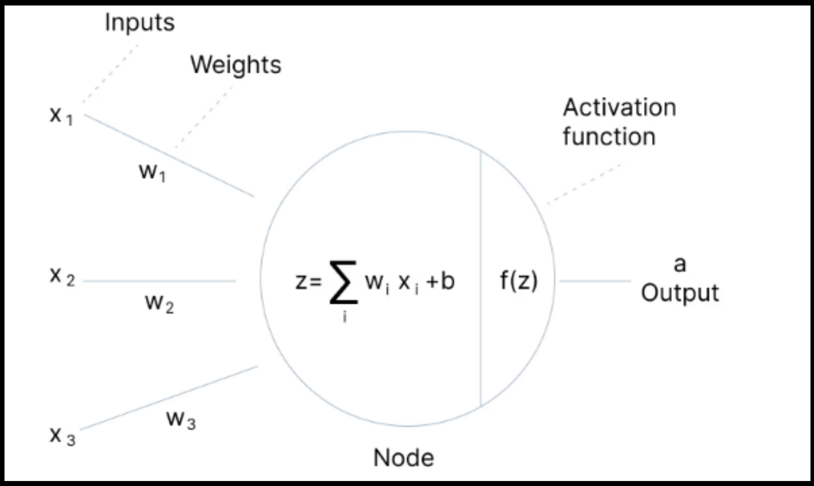
1) Activation Functions

(i) Step

(ii) Linear

(iii) Sigmoid

(iv) Relu



$z = wx + b = -5$

$f(z) = (wx + b)$ step function

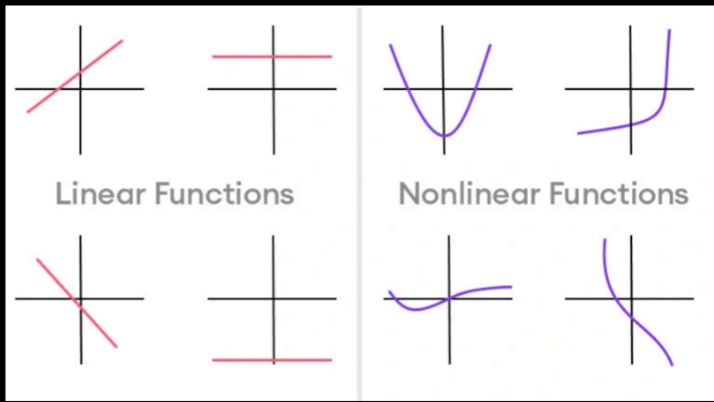
$= 0$ Mathematical Operations

$f(z) = (wx + b)$ Linear Function / Identity Function

$= (wx + b) = -5$

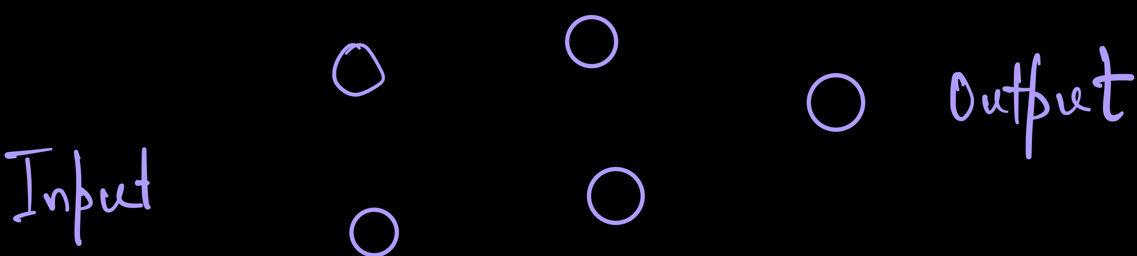
Why ?

(i) To non-linearity.



Activations :- Positive gradient
Negative gradient

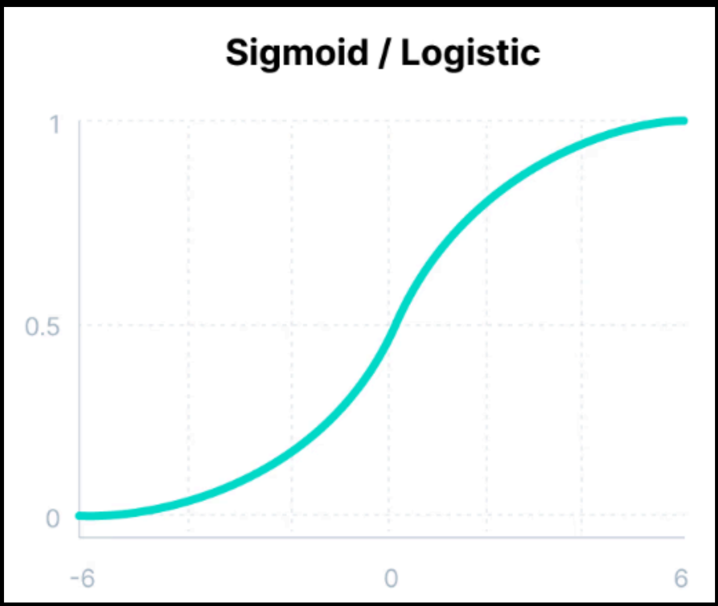
Neural Network:-

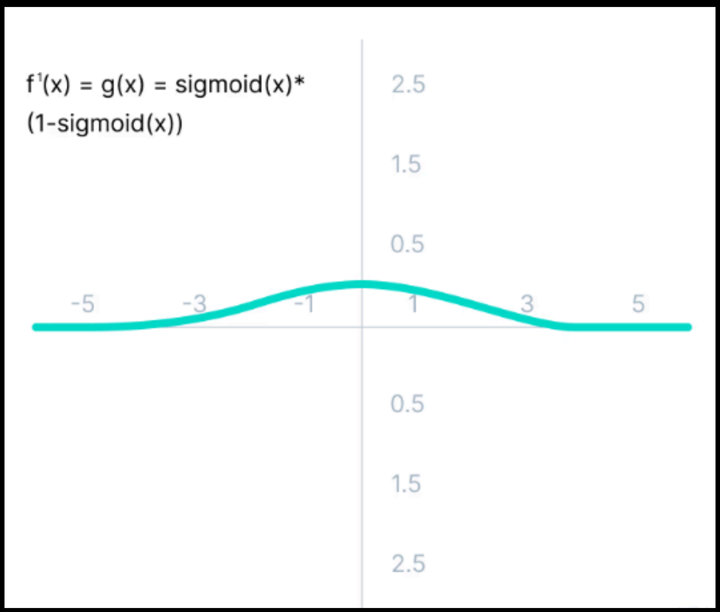


* Sigmoid *

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

$$f'(x) = \text{Sigmoid}(x) \cdot (1 - \text{Sigmoid}(x))$$

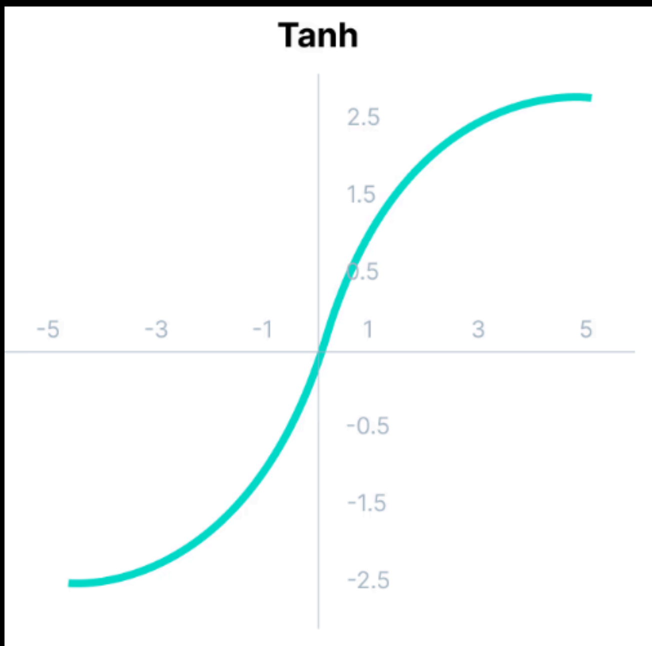


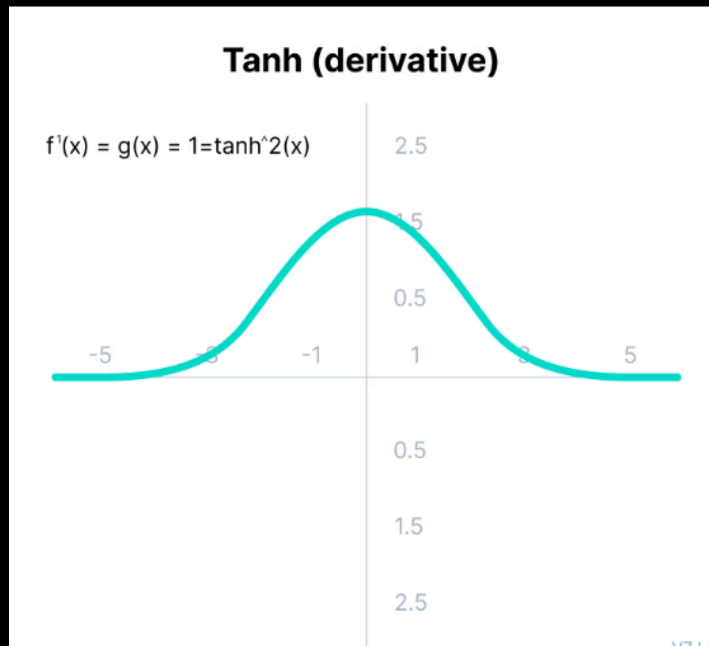


$W_{new} \approx W_{old}$, No learning
No weight update

Vanishing Gradient Problem $(-3, 3)$

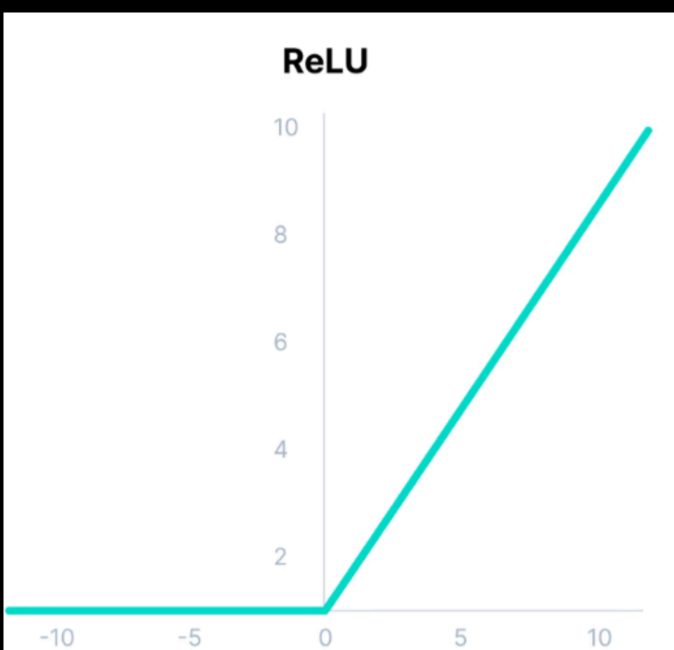
$$f(x) \text{ Tanh} = \frac{e^x - e^{-x}}{e^x + e^{-x}}$$



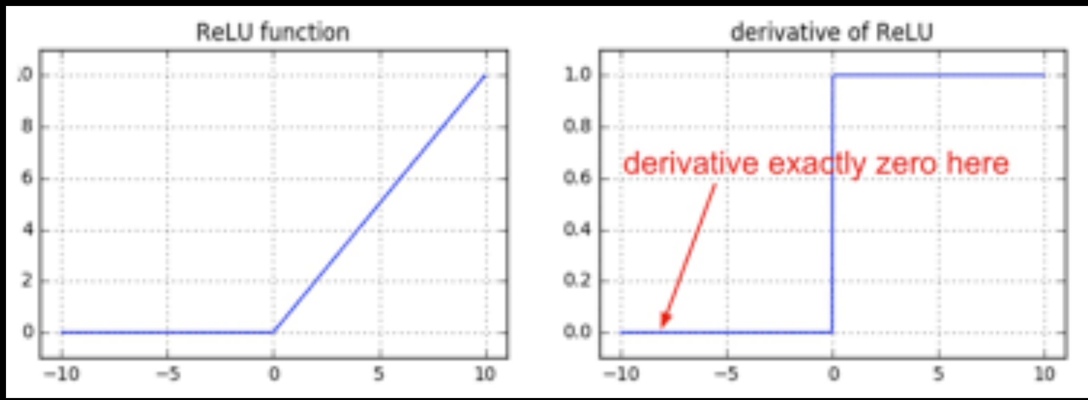


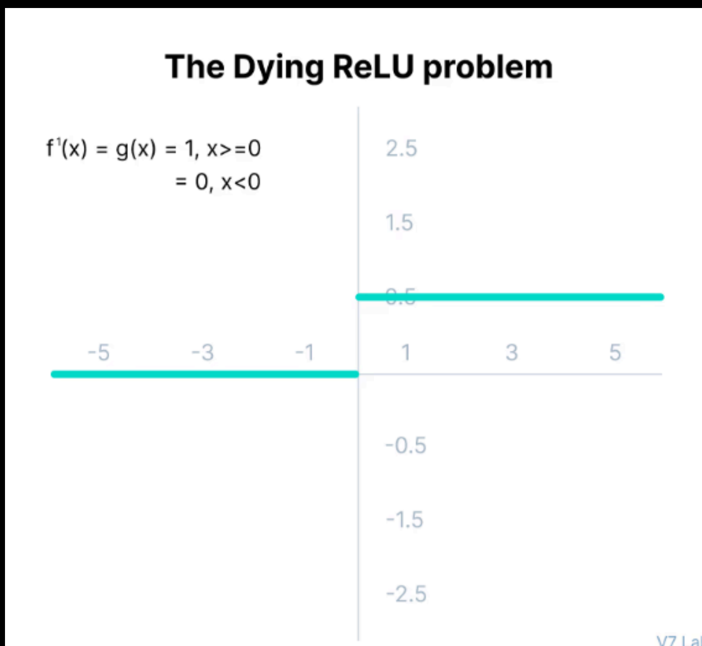
* Vanishing Gradient Problems *

RELU → Rectified Linear Units



$$f(x) = \max(0, x)$$





(i) Negative gradients = zero

Backprop, some neurons are not updated.

Dead Neurons

$z = wx + b$

if $z < 0$ for all samples:-

Output $= 0$ learning X

gradient $= 0$

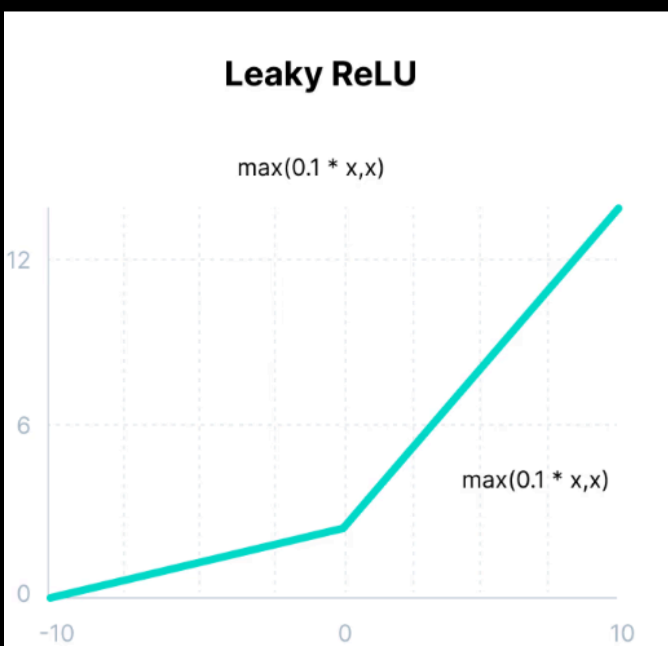
Why?

1) Zero gradients for negative inputs

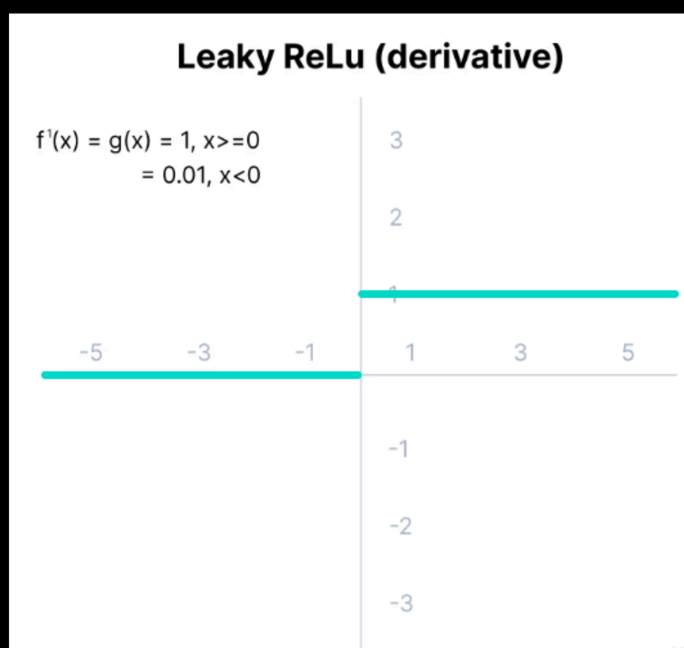
2) Bad weight initialization

3) Imbalanced Data

Leaky Relu



$$f(x) = \max(0.1x, x)$$

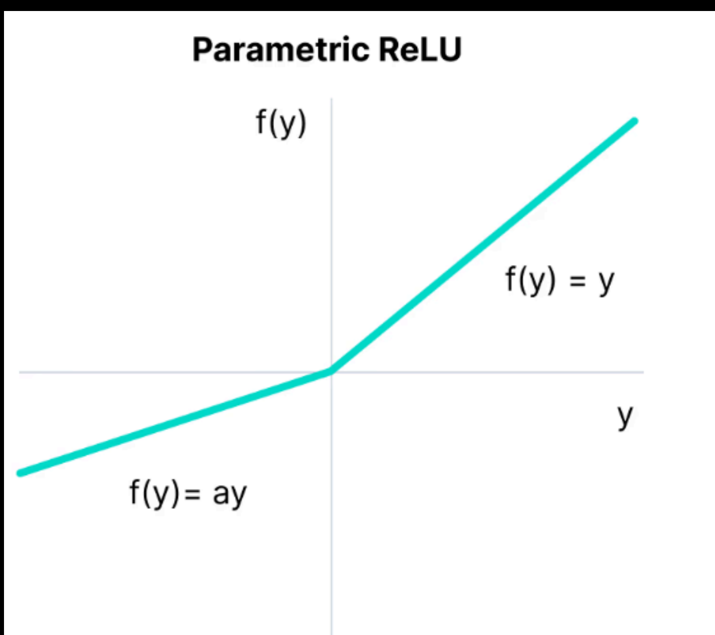


Gradient of our negative input will be very small value.

* learning of the NN will be slow, time consuming *

$W_{old} \approx W_{new}$

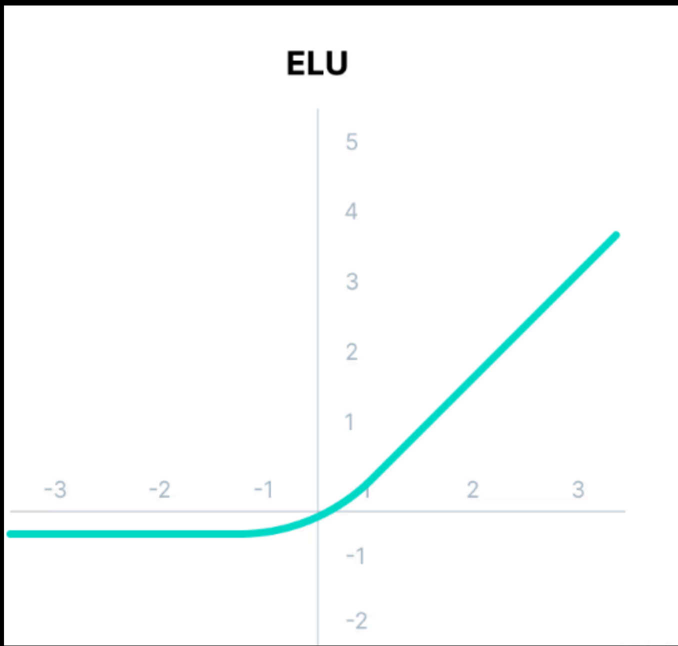
PRELU (Parametric Relu)



$f(x) = \max(ax, x)$

a = slope parameter for our negative values.

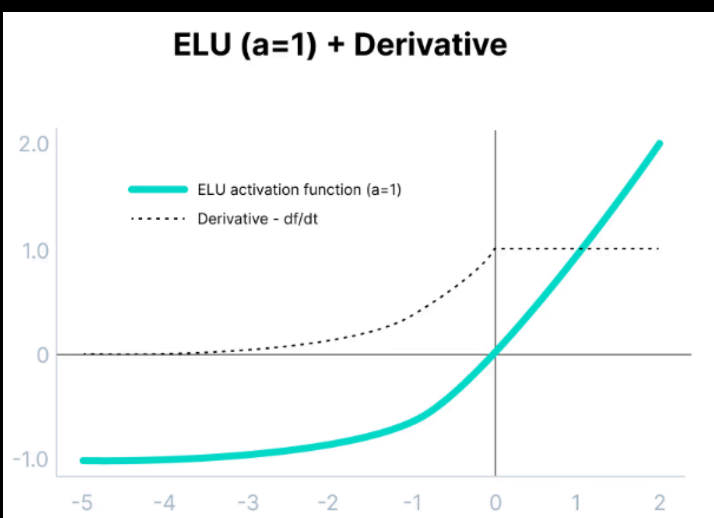
Exponential Linear Units (ELU)



modify the slope of the negative gradients.

log curve

$$f(x) = \begin{cases} x & \text{for } x \geq 0 \\ \alpha(e^{x} - 1) & \text{for } x < 0 \end{cases}$$



Computationally expensive

Classification Problem

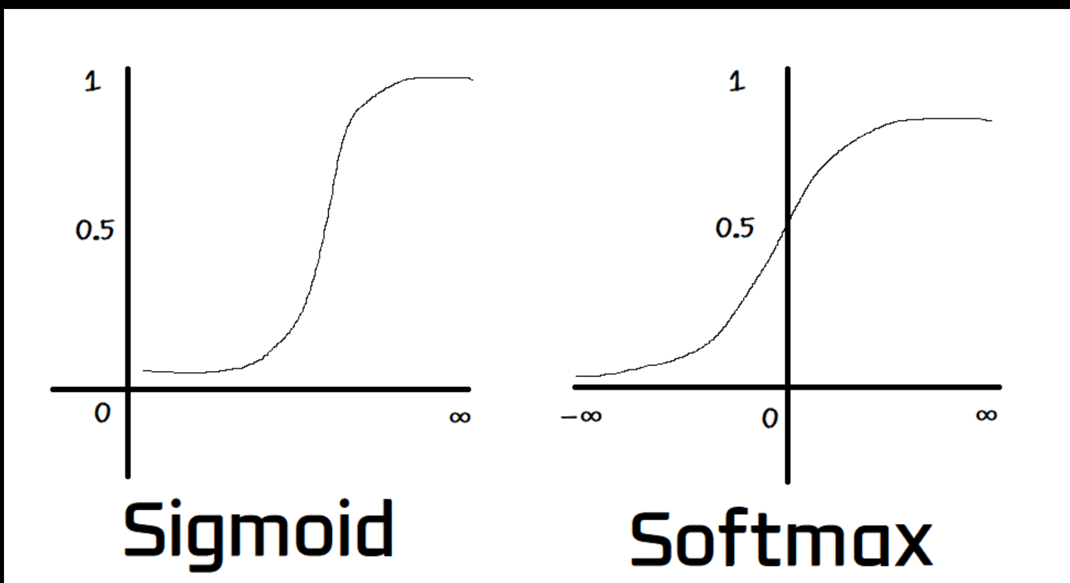
Binary Classification

(i) Sigmoid

Multiclass

(ii) Softmax

Softmax



$$f(x) = \frac{\exp(z_i)}{\sum_j \exp(z_j)}$$

3 classes

a, b, c

$$= \frac{e^a}{e^a + e^b + e^c}$$

Softmax = combination of multiple sigmoids

4 class example $a = \{0.8,\ 0.7,\ 0.6,\ 0.1\}$

Relative Probability

Sigmoid OR Softmax

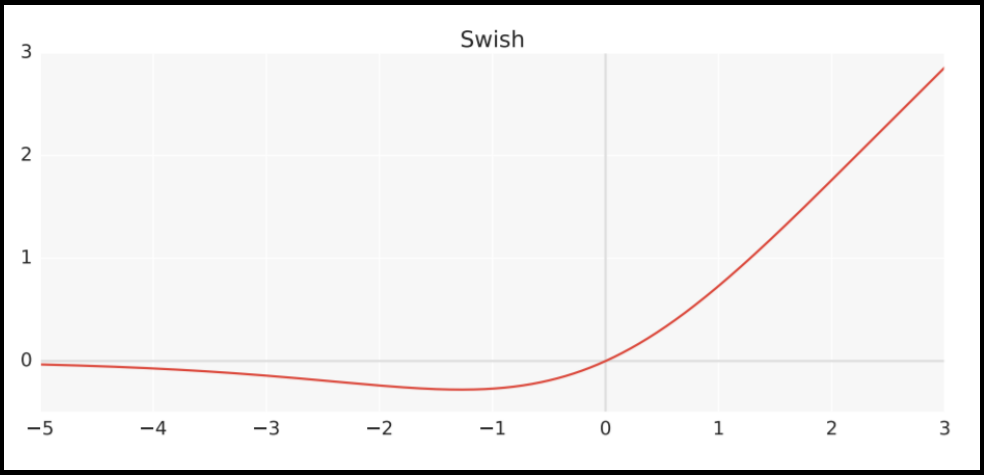
→ output layer

Swish Activation

Self gated Activation

Google

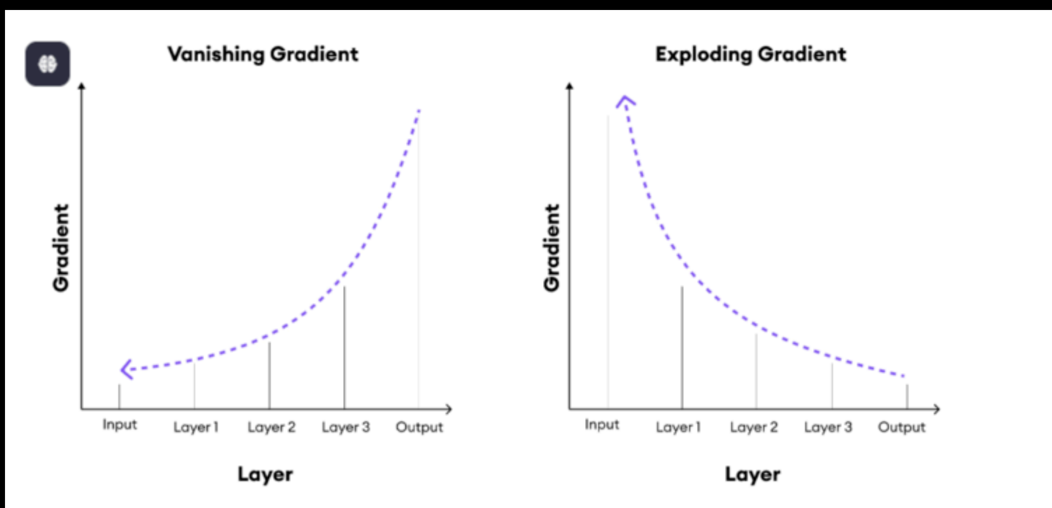
→ Human brain



$$f(x) = x \cdot \text{Sigmoid}(x)$$

Vanishing Gradient Problem

Exploding Gradient Problem



$$W_{new} \approx W_{old}$$

$$W_{new} > W_{old}$$

gradient will be very big, reaching towards infinity

i) $W_{new} = W_{old}$

$+ 0.058$

66

gradient $= 660000804$

Generalization of Neural Networks

$\longrightarrow$ Dropout