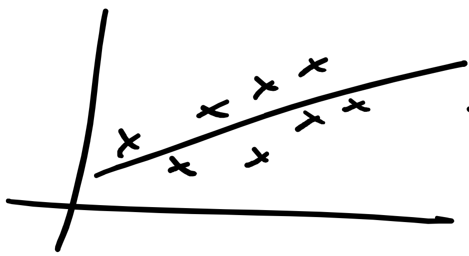
2025-07-10 - Explain the fundamental concepts in neural network and deep learning

10 July 2025 09:20

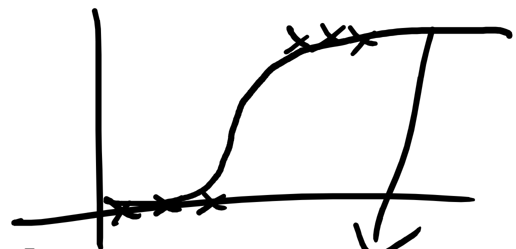
=> Activation Functions:

Linear Reg (Regression)

Logistic Reg (Classification)



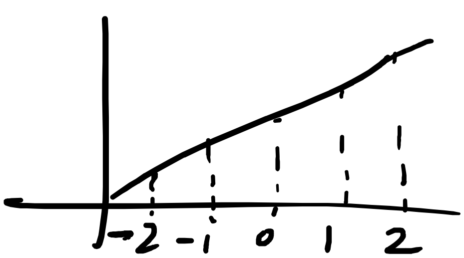
$\hat{y} = \beta_0 + \beta_1 x_i$

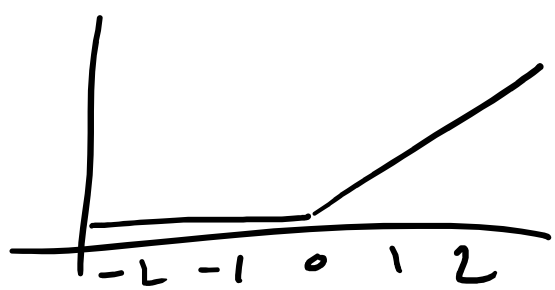


$\sigma(z)$

Activation Functions convert the linear equation to non-linear equation.

=> Rectified Linear Unit (ReLU)





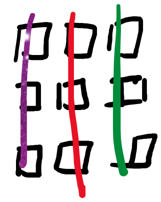
-> This layer cannot be used in o/p layer.

-> Relu (Tensor):

→ Relu (Tensor):

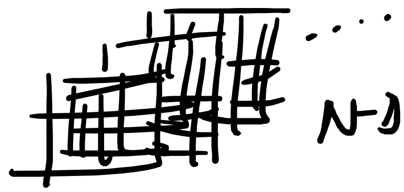
max (0, Tensor)

[.] 1-D Scalar



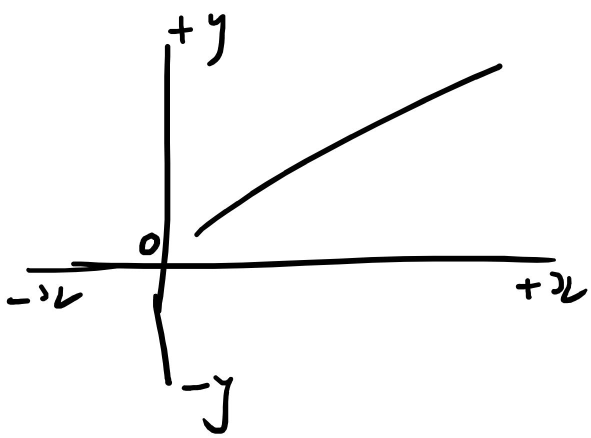
3-D Matrix

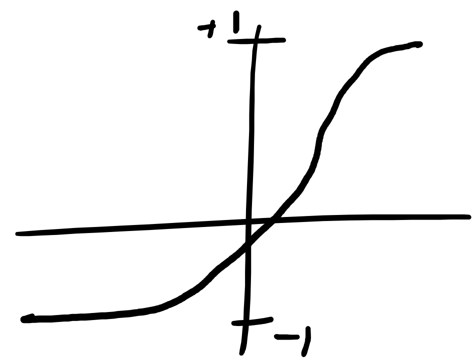
{ [ ] [ ] [ ] } 2-D Vector



N-D Tensor

→ Hyperbolic Tangent (TanH)





Cannot be used in o/p layer.

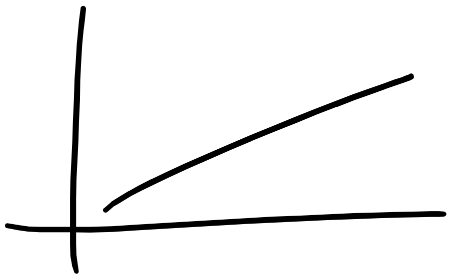
$$\text{Tanh} = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}}$$

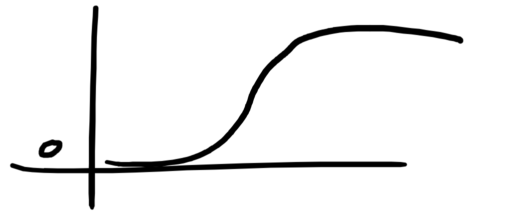
① Range b/w

-1 to 1

=> **Sigmoid** (Logit Function) : Binary Classification

I-1 yields probabilistic o/p. (0 to 1)





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

z represent linear equation

=> Always used in o/p layer.

=> **Softmax** : Categorical Classification -

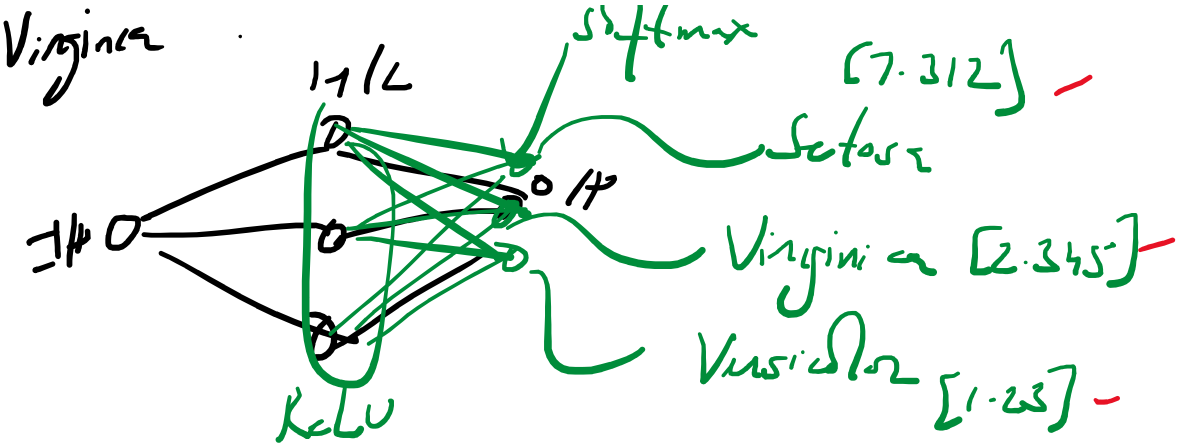
Type of cumulative probabilistic function.

=> Iris

Setosa Virginica Versicolor

11/2 /Softmax [7.312]

Setosa Virginica.



Rearrange into vector $\begin{bmatrix} - \\ - \\ - \end{bmatrix}$ } Cumulation 0–1

argmax ← $\begin{bmatrix} 0.63 \\ \cancel{0.07} \\ 0.4 \end{bmatrix}$ } 1

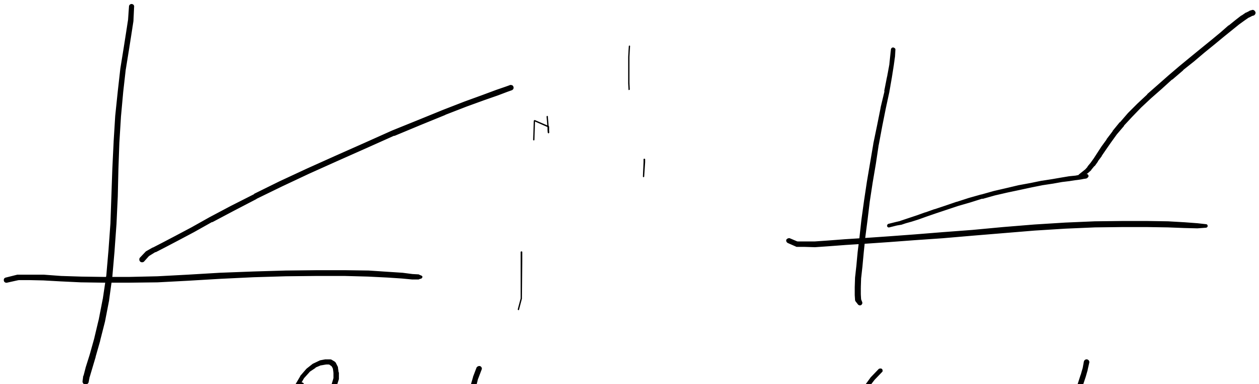
Softmax is always used in o/p layer in all cases.

$$\text{Softmax} = \frac{e^{x}}{\sum_{i}^{n} e^{x_i}}$$

=> **Leaky ReLU**: Nonlinearity fn

Always used in the hidden layers

Always used in the hidden layers



Parameter α (0.005 to 0.05)

leakyR(Tensor):
    max(Tensor $\times \alpha$, Tensor)

=> <u>Basics of model</u> :

$W \cdot x + b$

Activations $\quad a^{[\,]}$

Parametric learning
(Always linear)

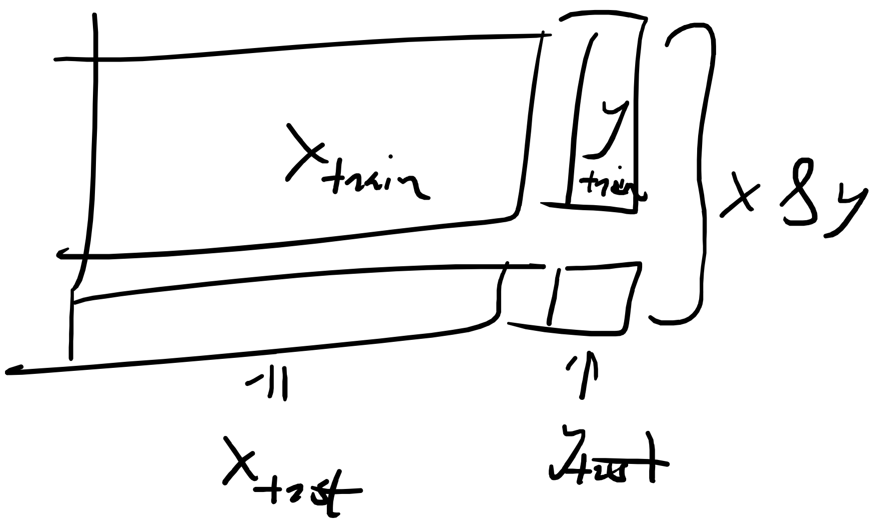
W - Weights
x - Explanatory Variable

b – Bias

⇒ Loss Functions

↳ Mean Squared Error:

Linear Equations



Error = (Predicted – Truth Value)

$$\text{Error} = (\hat{y} - y_i)$$

$$\text{Sum of Error} = \sum_{i=1}^{n} (\hat{y} - y_i)$$

$$\text{Sum of Squared Errors (SSE)} = \sum_{i=1}^{n} (\hat{y} - y_i)^2$$

Sum of Squared Errors (SSE) $\sum_{i=1} (\hat{y} - y)$

$$MSE = \frac{\sum_{i=1}^{n} (\hat{y} - y_i)^2}{N}$$