# Introduction to Deep Learning for Scientists and Engineers

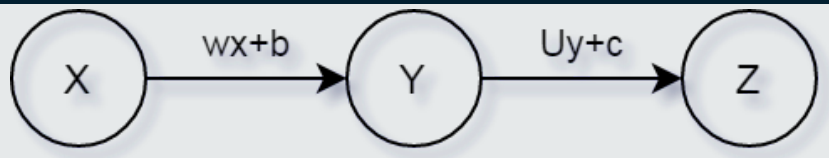
Abhijat Vatsyayan [1]

August 14, 2020

# Summary

# Multiple layers

## Layers (composition) of $ax + b$ style functions




$$y = \mathbb{W}x + b$$

$$\begin{aligned} z &= \mathbb{U}y + c = \mathbb{U}(\mathbb{W}x + b) + c \\ &= (\mathbb{U}\mathbb{W})x + (\mathbb{U}b + c) = \mathbb{V}x + d \end{aligned}$$

## Introducing non-linearity with element-wise sigmoid

$$y = \sigma(\mathbb{W}x + b)$$

$$z = \sigma(\mathbb{U}y + c)$$

# What can these functions (neural networks) do?

*A standard multilayer feed-forward network with a locally bounded piecewise continuous activation function can approximate any continuous function to any degree of accuracy if and only if the network's activation function is not a polynomial.*

*-Leshno et al.,1993*

*In particular, we show that arbitrary decision regions can be arbitrarily well approximated by continuous feed forward neural networks with only a single internal, hidden layer and any continuous sigmoidal nonlinearity.*

*-Cybenko, 1989*

# Multi-layer, feed forward, non-polynomial

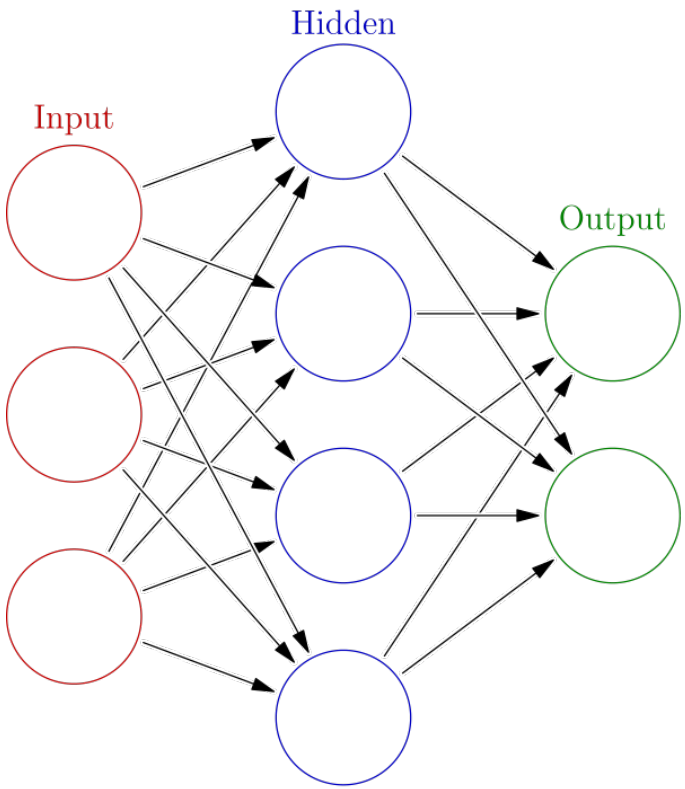
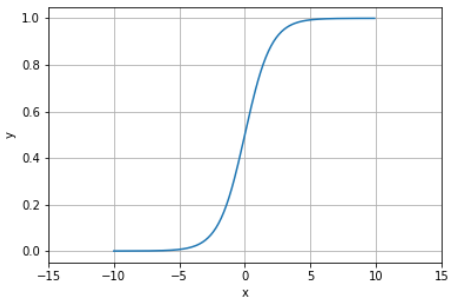


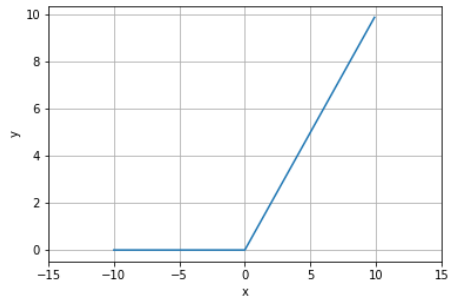

Image from https://en.wikipedia.org/wiki/Artificial_neural_network

# Activation functions

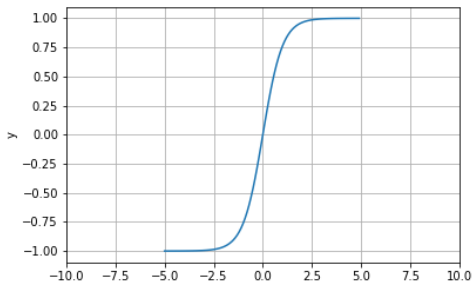
$$f(x) = \frac{1}{1 + e^{-x}}$$



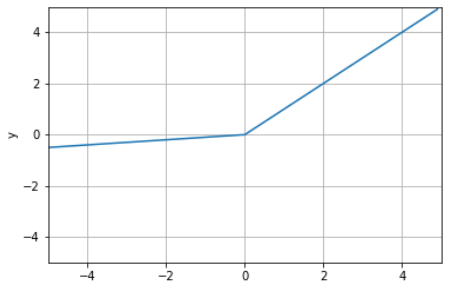

$$f(x) = max(x, 0)$$




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




$$f(x) = \begin{cases} x, & \text{if } x \geq 0 \\ 0.1x, & \text{otherwise} \end{cases}$$



# References I