ONLINE MASTERS IN **DATA SCIENCE**

DSC 255 - MACHINE LEARNING FUNDAMENTALS

# FEEDFORWARD NEURAL NETS

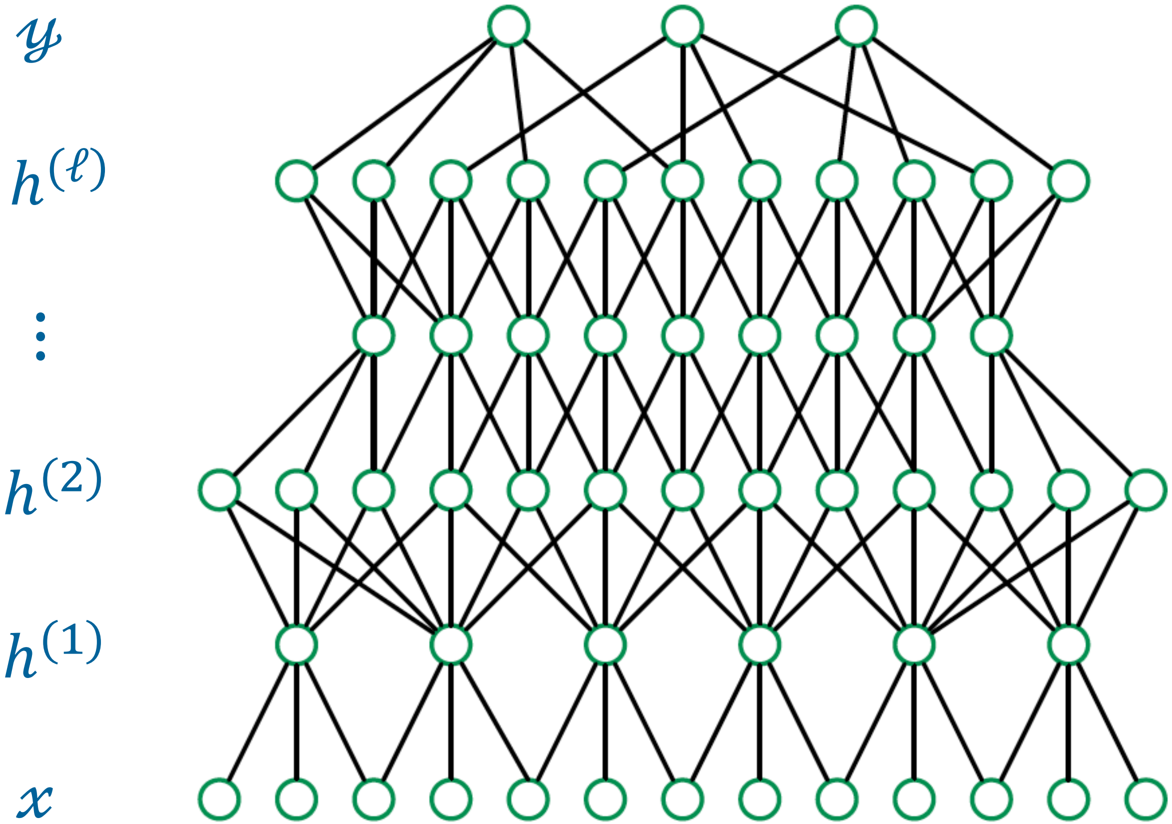
SANJOY DASGUPTA, PROFESSOR

UC San Diego

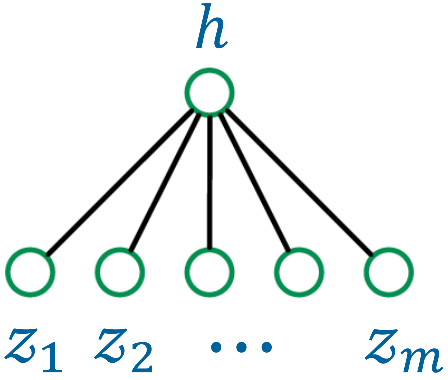
COMPUTER SCIENCE & ENGINEERING

HALICIOĞLU DATA SCIENCE INSTITUTE

# Feedforward Neural Nets

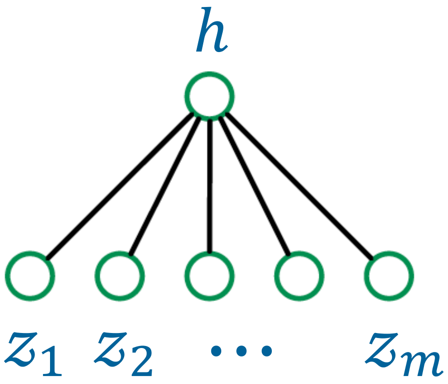


# The Value at a Hidden Unit




How is $h$ computed from $z_1, \ldots, z_m$?

# The Value at a Hidden Unit




How is $h$ computed from $z_1, \ldots, z_m$?

- $h = \sigma(w_1 z_1 + w_2 z_2 + \cdots + w_m z_m + b)$
- $\sigma(\cdot)$ is a nonlinear **activation function**, e.g., "rectified linear"

$$\sigma(u) = \begin{cases} u & \text{if } u \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

## Common Activation Functions

- Threshold function or Heaviside step function

$$\sigma(z) = \begin{cases} 1 & \text{if } z > 0 \\ 0 & \text{otherwise} \end{cases}$$

- Sigmoid

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

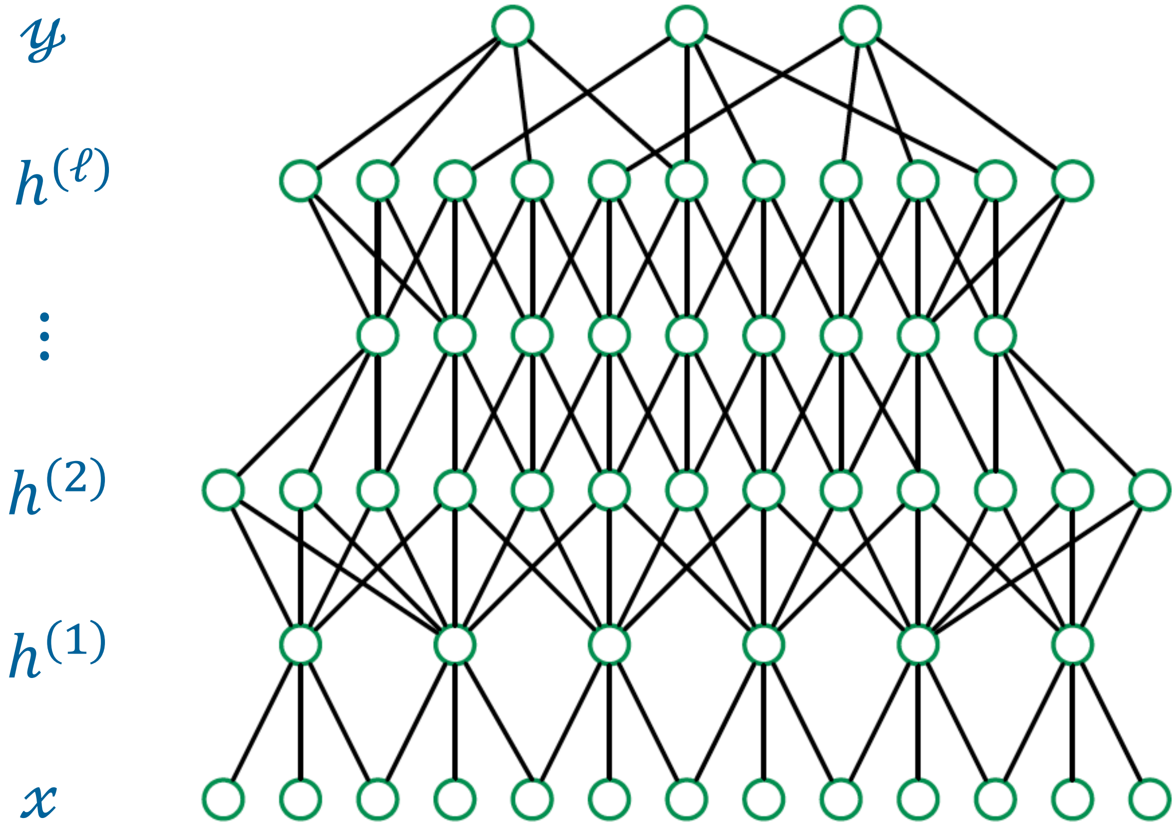
- Hyperbolic tangent

$$\sigma(z) = \tanh(z)$$

- ReLU (rectified linear unit)

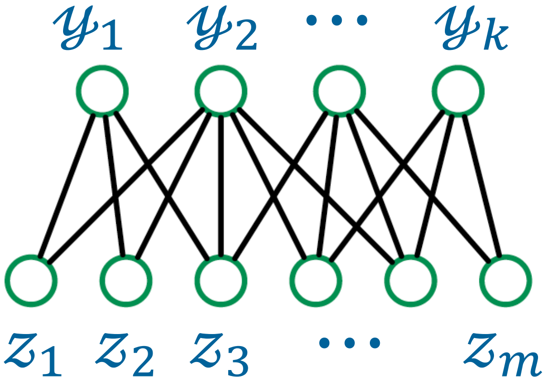
$$\sigma(z) = \max(0, z)$$

# Why do we Need Nonlinear Activation Functions?



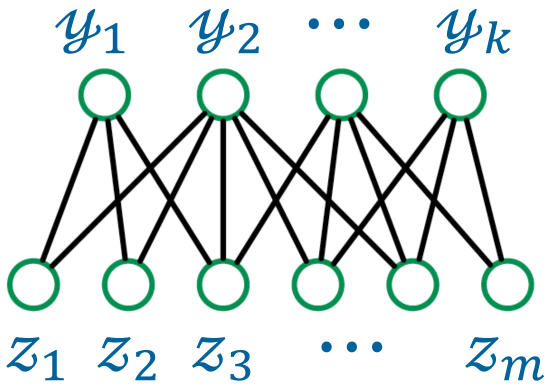
## The Output Layer

Classification with $k$ labels: want $k$ probabilities summing to 1.



# The Output Layer

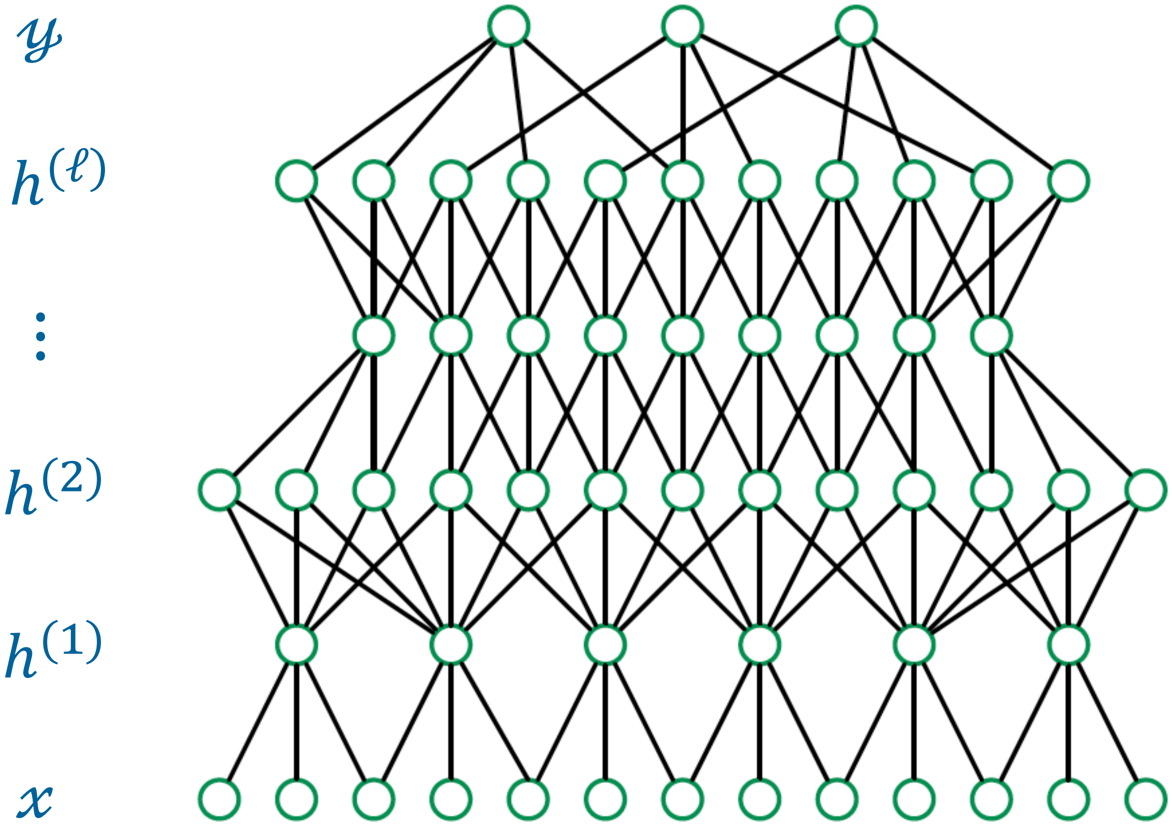
## Classification with $k$ labels: want $k$ probabilities summing to 1.




- $y_1, \ldots, y_k$ are linear functions of the parent nodes $z_i$.
- Get probabilities using **softmax:**

$$\Pr(label\ j) = \frac{e^{y_j}}{e^{y_1} + \cdots + e^{y_k}}.$$

# The Complexity



## Approximation Capability

**Let $f: \mathbb{R}^d \longrightarrow \mathbb{R}$ be any continuous function.**
**There is a neural net with one hidden layer that approximates $f$ arbitrarily well.**

# Approximation Capability

**Let $f: \mathbb{R}^d \longrightarrow \mathbb{R}$ be any continuous function.**
**There is a neural net with one hidden layer that approximates $f$ arbitrarily well.**

- The hidden layer may need a lot of nodes.

- The benefit of depth: for certain classes of functions, you need:
  - *Either:* one hidden layer of enormous size
  - *Or:* multiple hidden layers of moderate size