Feedforward neural nets

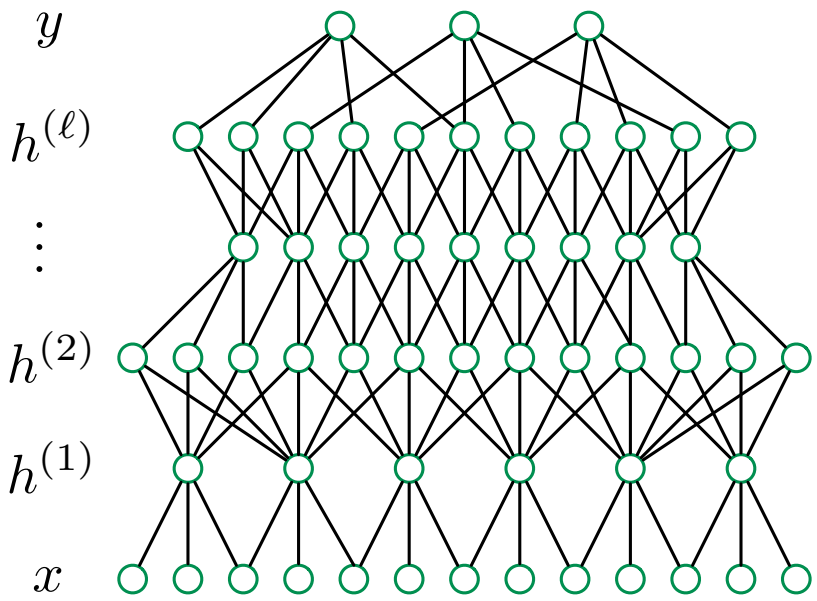
Sanjoy Dasgupta

University of California, San Diego

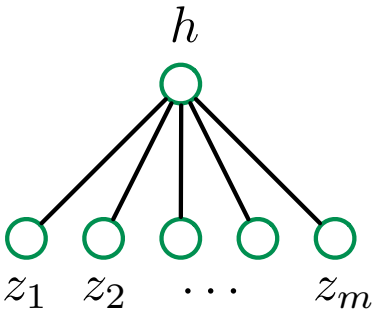
Topics we'll cover

1. The architecture
2. The functions
3. The effect of depth

The architecture



The value at a hidden unit

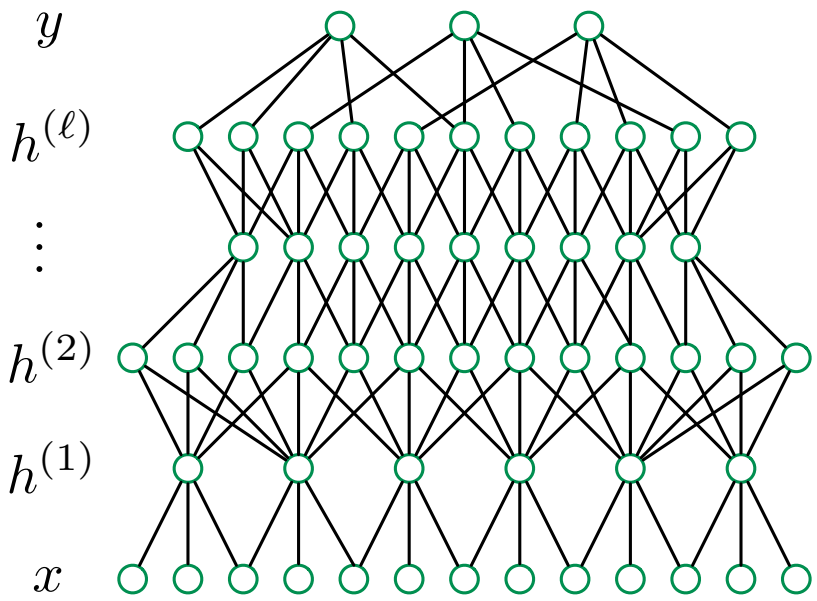


How is h computed from $z_1, \ldots, z_m$?

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

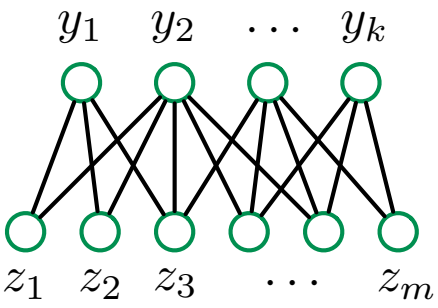
$$\sigma(u) = \begin{cases} u & \text{if } u \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

Why do we need nonlinear activation functions?



The output layer

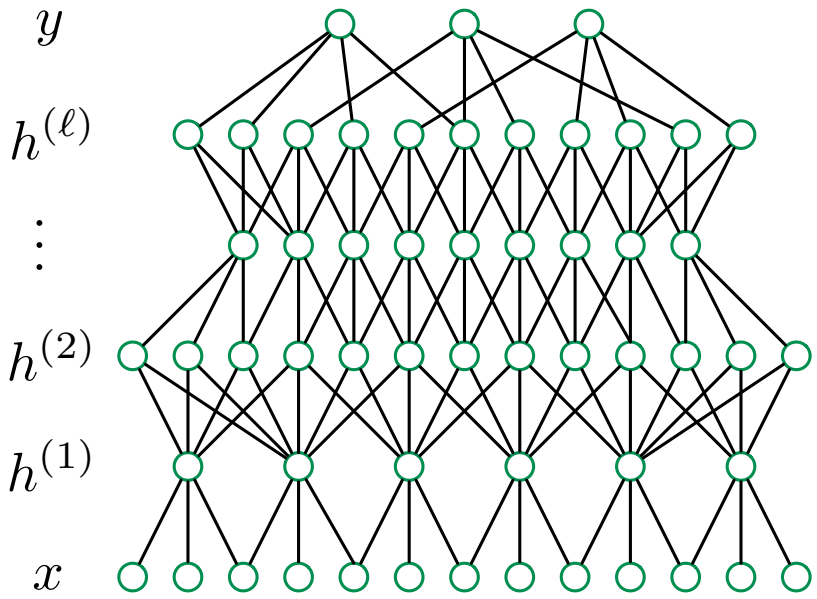
Classification task with k labels: want k probabilities summing to 1.



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

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

The complexity



The effect of depth

- Universal approximator
  Any function can be arbitrarily well approximated by a neural net with one hidden layer.

- Concerns about size
  To fit certain classes of functions:
  - Either: one hidden layer of enormous size
  - Or: multiple hidden layers of moderate size