

# Machine Learning: Representing Neural Networks

Sidharth Baskaran

June 2021

## Non-linear hypotheses

- Computer vision - section matrix of pixel intensities correspond to an image
  - Denote + and – for affirming if an image fits a classification
  - Would need nonlinear hypothesis
  - Feature vector  $x$  - pixel intensities in a column vector
- Example
  - Assume  $50 \times 50$  pixel images - 2500 pixels
    - \*  $n = 2500$  features
  - Quadratic features would mean  $\sim 3$  mil features
    - \* number of features =  $50^2 + C(50^2, 2)$  as need all possible ways of 2 terms from features in addition to number of features present

## Neural Network Model

- Neuron structure
  - Dendrite - input wires
  - Computation in nucleus
  - Axon - output wires
- Neuron model - logistic unit
  - Input wires from features  $x$  through computation to output  $h_\theta(x)$
  - Sigmoid activation function  $g(z) = \frac{1}{1+e^{-z}}$
  - Parameters  $\theta$  are same as weights
- Layers of neural networks
  - Layer 1 of features/inputs
  - Layer 2 of bias units - is *hidden* as is not an output
  - Layer 3 is the output

$$[x_0 x_1 x_2 x_3] \rightarrow [a_1^{(2)} a_2^{(2)} a_3^{(2)}] \rightarrow h_\theta(x)$$

Node values are

$$\begin{aligned} a_1^{(2)} &= g(\Theta_{10}^{(1)} x_0 + \Theta_{11}^{(1)} x_1 + \Theta_{12}^{(1)} x_2 + \Theta_{13}^{(1)} x_3) \\ a_2^{(2)} &= g(\Theta_{20}^{(1)} x_0 + \Theta_{21}^{(1)} x_1 + \Theta_{22}^{(1)} x_2 + \Theta_{23}^{(1)} x_3) \\ a_3^{(2)} &= g(\Theta_{30}^{(1)} x_0 + \Theta_{31}^{(1)} x_1 + \Theta_{32}^{(1)} x_2 + \Theta_{33}^{(1)} x_3) \\ h_\Theta(x) = a_1^{(3)} &= g(\Theta_{10}^{(2)} a_0^{(2)} + \Theta_{11}^{(2)} a_1^{(2)} + \Theta_{12}^{(2)} a_2^{(2)} + \Theta_{13}^{(2)} a_3^{(2)}) \end{aligned}$$

Arguments are  $z_c^{(k)}$  where  $c$  is which element of the layer and  $k$  is the layer.

- Notation

- $a_i^{(j)}$  is activation of unit  $i$  in layer  $j$
- $\Theta^{(j)}$  is matrix of weights controlling function mapping from layer  $j$  to layer  $j + 1$
- If network has  $s_j$  units in layer  $j$ ,  $s_{j+1}$  units in layer  $j+1$ , then  $\Theta^{(j)}$  is of dimension  $s_{j+1} \times (s_j + 1)$
- \*  $x_0$  and  $\Theta_0^{(j)}$  bias nodes are not shown in a NN diagram
- Vectorized
  - Arguments of  $g$ ,  $z_c^{(k)} = \theta^{(k)} x$
  - Can let  $a^{(k)} = g(z^{(k)}) = \Theta^{(k)} a^{(k)}$
  - $x = \begin{bmatrix} x_0 \\ x_1 \\ \dots \\ x_n \end{bmatrix}$ ,  $z^{(j)} = \begin{bmatrix} z_1^{(j)} \\ z_2^{(j)} \\ \dots \\ z_n^{(j)} \end{bmatrix}$
  - Thus  $z^{(j)} = \Theta^{(j-1)} a^{(j-1)}$

## Multiclass Classification

- One-vs-all method extension
- Multiple output units for multiple classifications
- $h_{\Theta}(x)$  is a vector