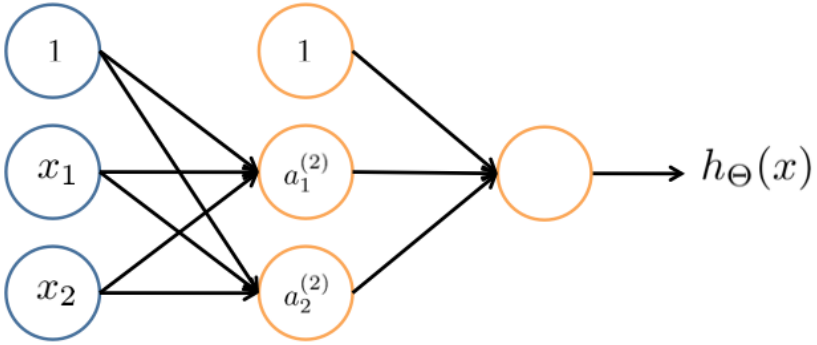
ASSIGNMENT 4

ANN: Multiclass problem

Assignment 4 (multiclass problem)

- Recommended order:
  - `prediction:`
    - **Checkpoint**: accuracy ≈ 97.52% (with theta values provided)
  - `sigmoidGradient`
  - `randInitializeWeights`
    - **Checkpoint**: values are properly initialized, i.e. (-ε, ε)
  - `nnCostFunction`
    - **Checkpoint**: relative difference
    - **Checkpoint**: cost debugging parameters (w/ λ = 10)

prediction



$$h_\theta(x) = x * \theta^T$$

$$a_1^{(2)} = g(\Theta_{10}^{(1)} x_0 + \Theta_{11}^{(1)} x_1 + \Theta_{12}^{(1)} x_2)$$

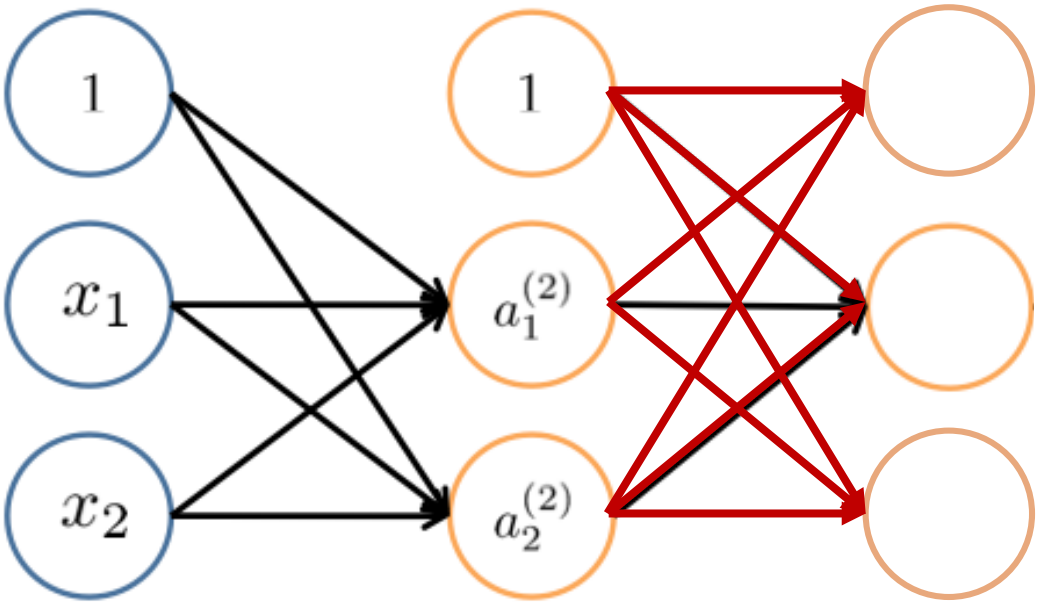
$$a_2^{(2)} = g(\Theta_{20}^{(1)} x_0 + \Theta_{21}^{(1)} x_1 + \Theta_{22}^{(1)} x_2)$$

$$h_\Theta(x) = g(\Theta_{10}^{(2)} a_0^{(2)} + \Theta_{11}^{(2)} a_1^{(2)} + \Theta_{12}^{(2)} a_2^{(2)})$$

- Calculation per layer (forward propagation)!
- You will (probably) need to handle the values of *p* (check the predictions before moving on)

prediction

- Multiclass network



$h_\Theta(x)$

sigmoidGradient

- You will need:

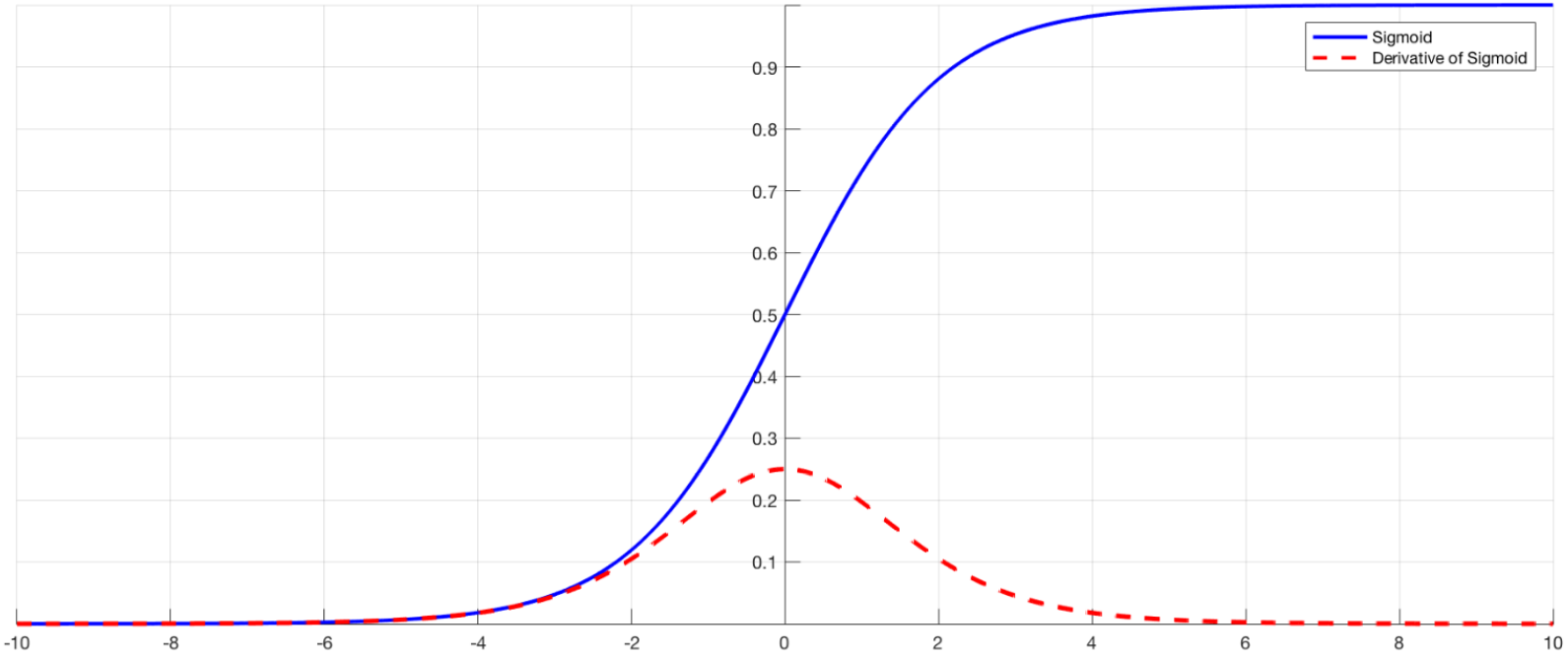
$$g'(x) = \frac{d}{dx} g(x) = g(x)(1 - g(x))$$

$$where \qquad g(x) = \frac{1}{1 + e^{-\theta^T}}$$

- And we can prove mathematically that:

$$g'(x) = g(x) * (1 - g(x))$$

sigmoidGradient

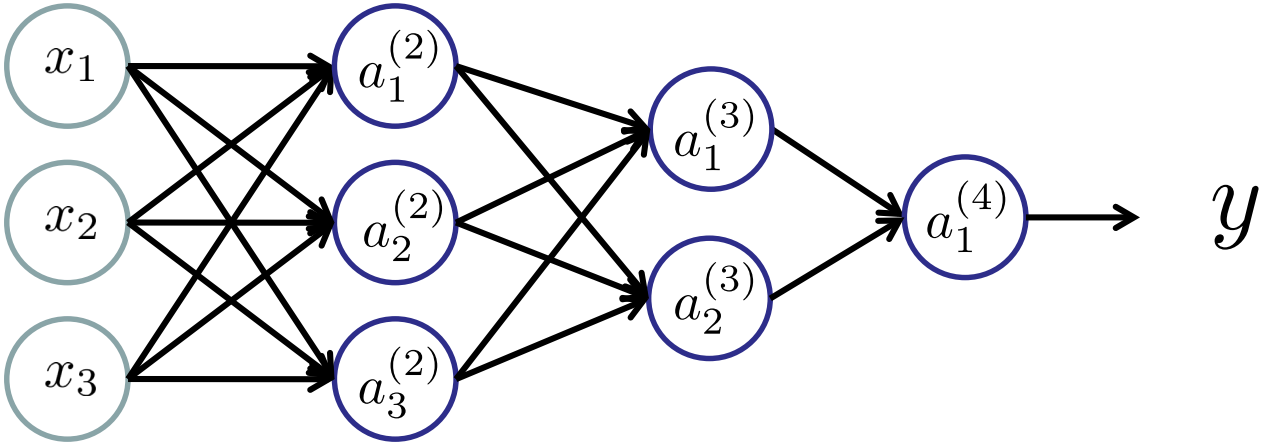


https://towardsdatascience.com/derivative-of-the-sigmoid-function-536880cf918e

randInitializeWeights

- Initialize all network weights to small random values -> (-ε, ε)
- You need a matrix of (initialized) weights
- Verify it!!!!

nnCostFunction



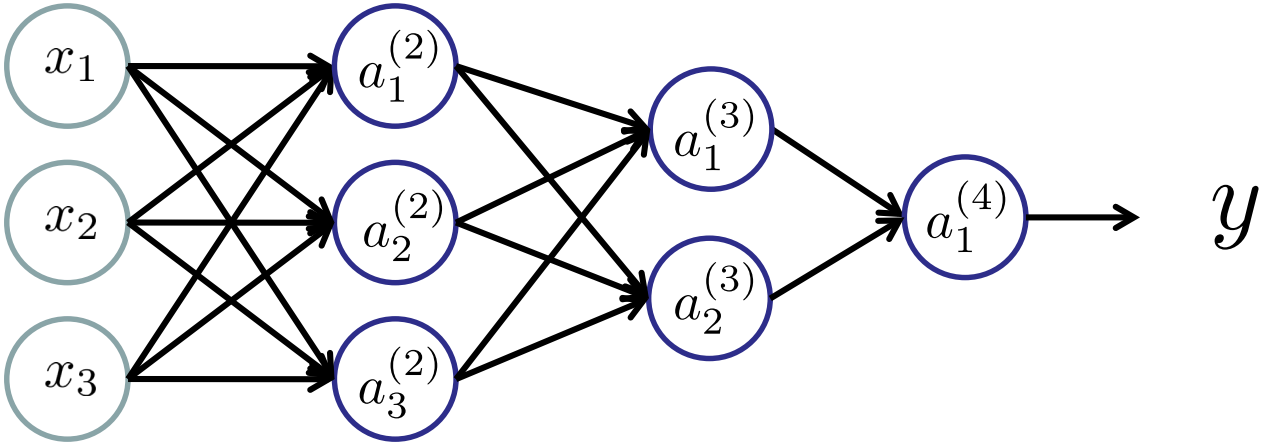
$$\delta^{(4)} = a^{(4)} - y \quad \text{where} \quad a^{(4)} = h\_\theta(x)$$

$$\delta^{(3)} = (\theta^{(3)})^T \delta^{(4)} * g'(z^{(3)})$$

$$\delta^{(2)} = (\theta^{(2)})^T \delta^{(3)} * g'(z^{(2)})$$

~~$\delta^{(1)}$~~

nnCostFunction



(from backprop.) Update each network weight θ_{kh}:

$$\theta_{ij} \leftarrow \frac{1}{m} * \Delta_{ij} + \lambda * \theta_{ij}$$

where

$$\Delta_{ij} = \Delta_{ij} + a_j \, \delta_i$$

nnCostFunction

- **Checkpoint**: relative difference
- **Checkpoint**: cost debugging parameters (w/ $\lambda = 10$)

Assignment 4

- Implement the ANN proposed
- Try to understand the code and the process followed
- Follow the instructions included as comments (extra validations)

- A **short** document discussing the concepts proposed in Section 2 Evaluation