when initialize weights, try not to intitialize all of the weights as zeros because then all the activation outputs will be the same and the network cannot learn something interesting

instead, randomly anitialize something thats between [-epsilon, epsilon], maybe [-1, 1]

logistic regression uses sigmoid function as activation. however, the cost function as a result has to log function to make sure the cost function is convex

for multi-class classification, typically multi-class k >= 3 because for k <= 2 it's just binary classification. the output layer of the multi-class classification is vector of size k. e.g., [0,1,0,0]

the non-linear nature of a network is an output of the non-linear activation function

regularization is used to make sure the weights are minimize in order to make sure the network does not overfit.

note the distinction between decision boundary and the sigmoid function