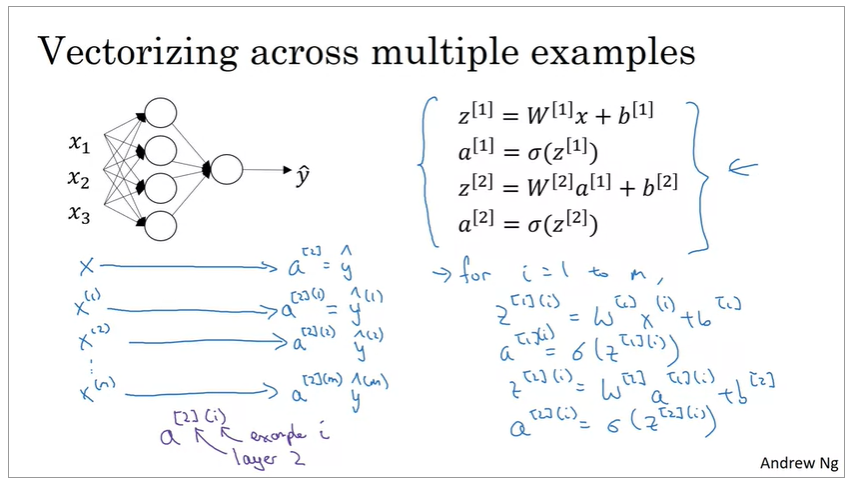
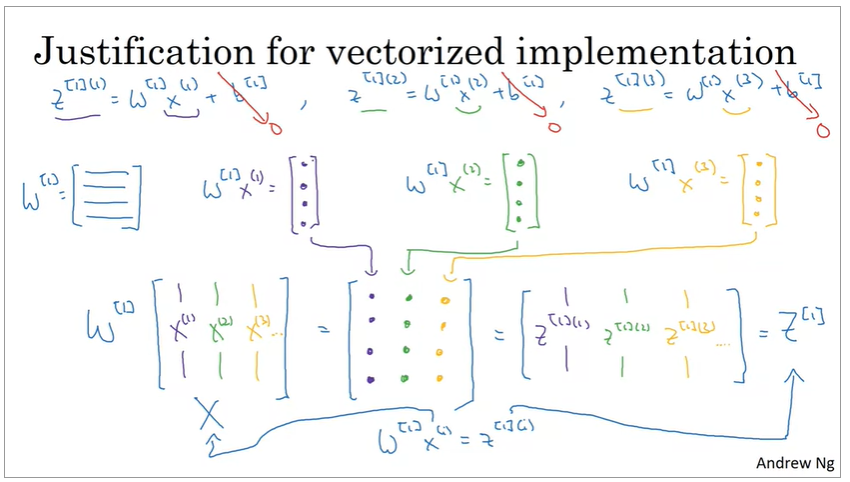
NEURAL NETWORK

với mỗi record đều sẽ dùng chung w[i] (i) để tối ưu dần các w này

và mỗi bước đều phải tính a[i] (i) giống như tính y^ nhưng là của một nút con của mạng

x là chiều dọc, X là các x được stack theo chiều ngang

w là chiều ngang, W là các w được stack chiều dọc

số row W chính là số nút (node) của hidden layer liền sau

mỗi row W là w (ngang) nhân vs x (dọc) là giá trị của một nút

=> vậy nên khi W.dot(x) sẽ được kết quả của toàn bộ lớp liền sau

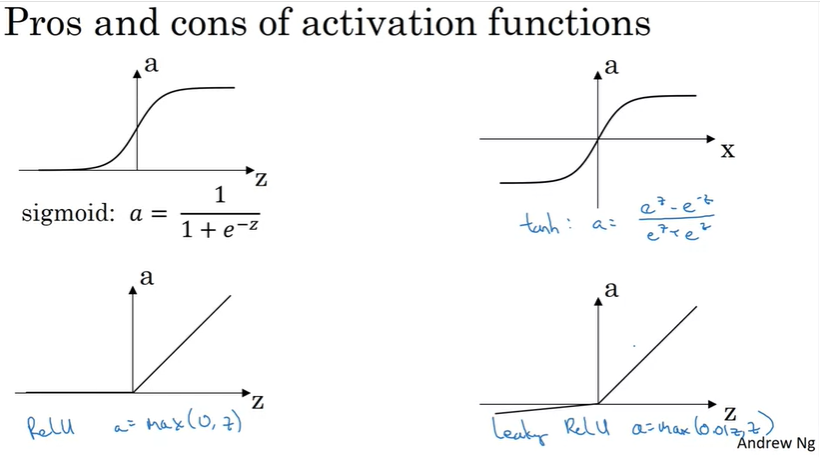
giải thích việc stacking với toàn tập train,

stack từ input lẫn các lớp mạng đều stack

từ đây mình sẽ tính sum theo chiều ngang

để tối ưu dần dần các nút mạng của mình

=> nên W.dot(X) sẽ là toàn bộ kết quả được stack lại theo chiều ngang

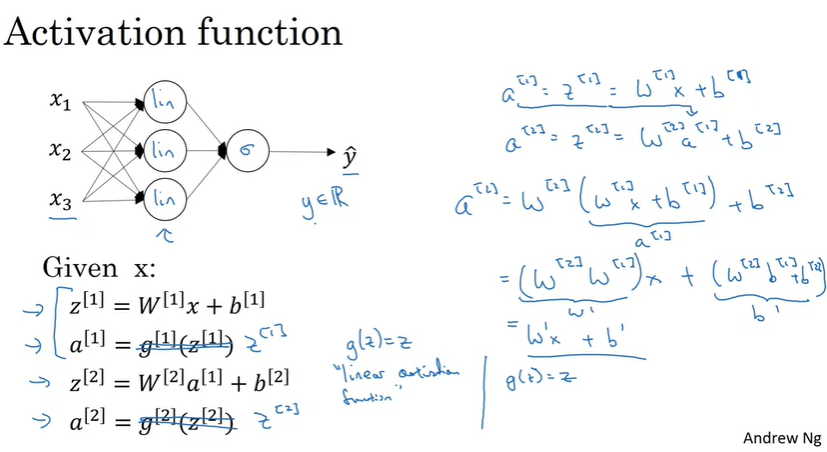


sigmoid chỉ tốt với binary classification (đúng - sai, 0 - 1)

tanh ok nhất vì hội tụ đủ thứ, nhưng chậm

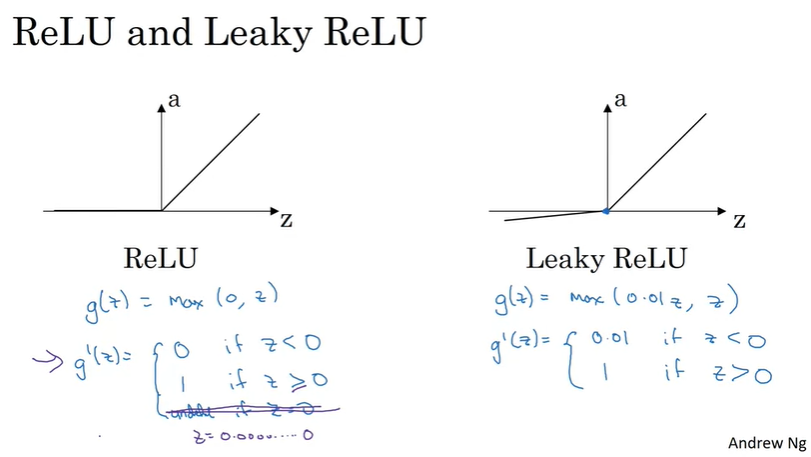
ReLU = max (0, z) nhanh nhưng số 0 và gần 0 sẽ là vấn đề

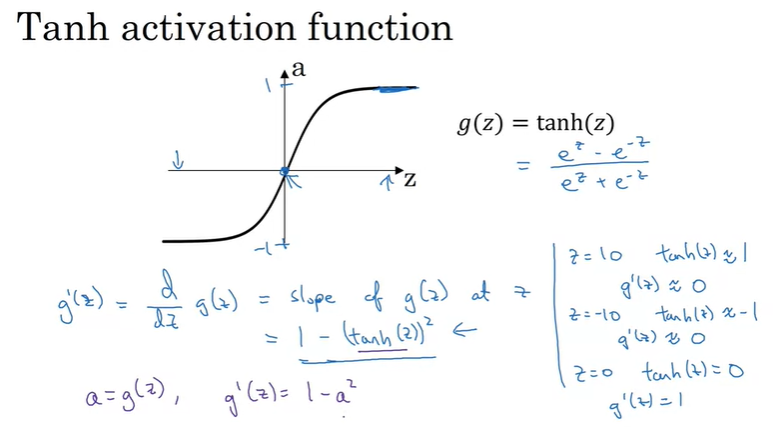
leaky ReLU = max(0.01 \* z, z) cải thiện của relu, có giá trị âm

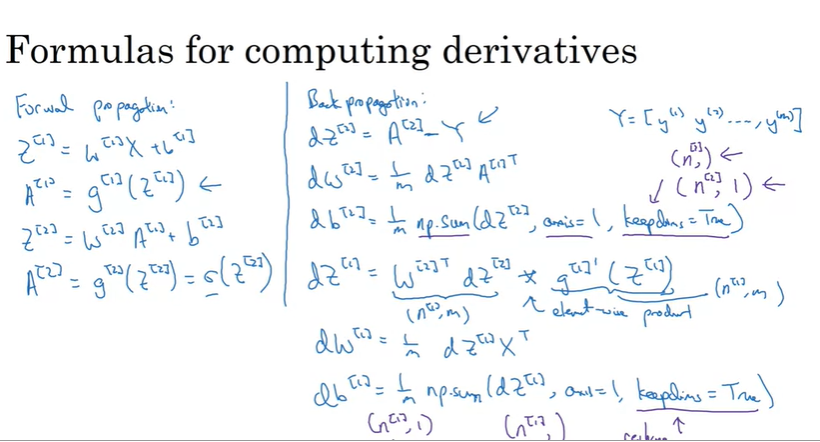


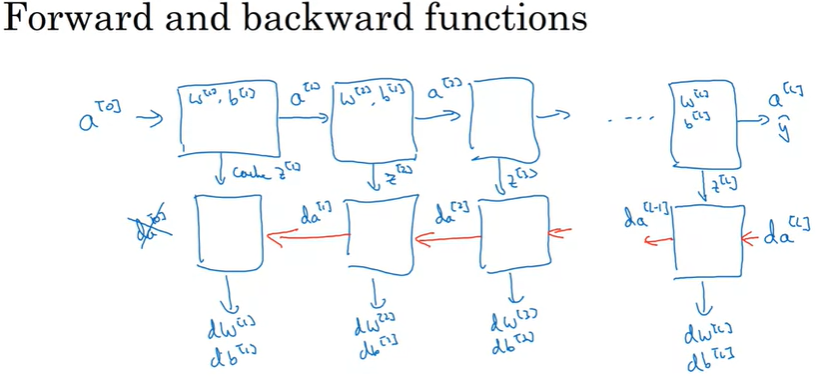
lý do không chọn linear vì nếu là linear thì mạng neuron trở nên vô nghĩa

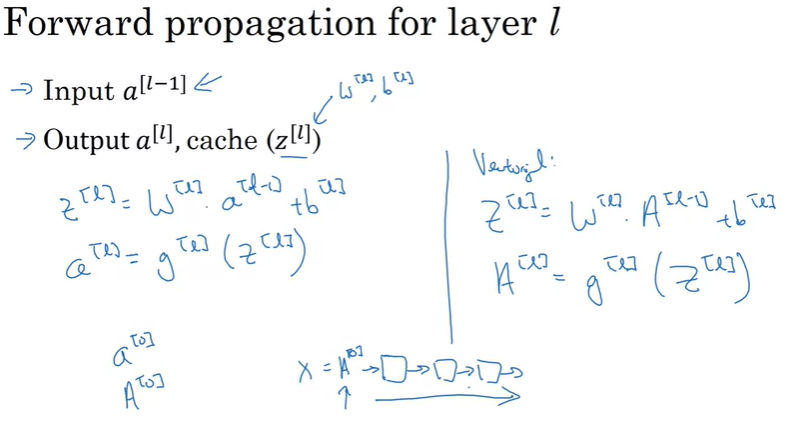
vì thực ra mình chỉ đang tối ưu một W' chính là tích của W mỗi lớp mạng

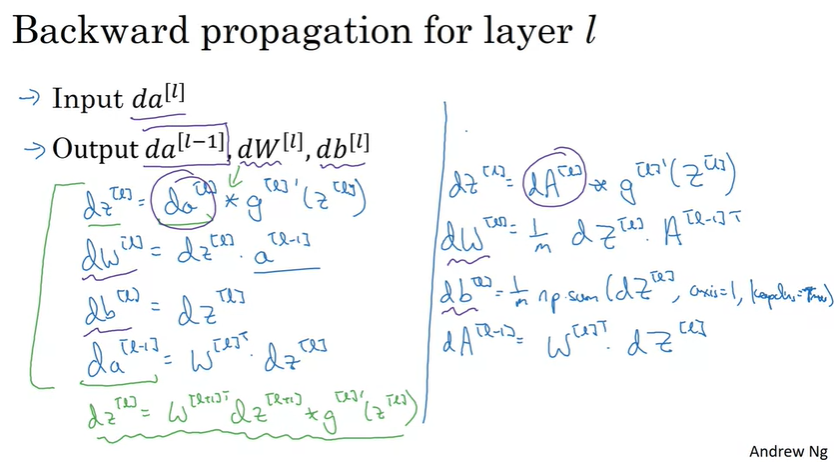












tổng

