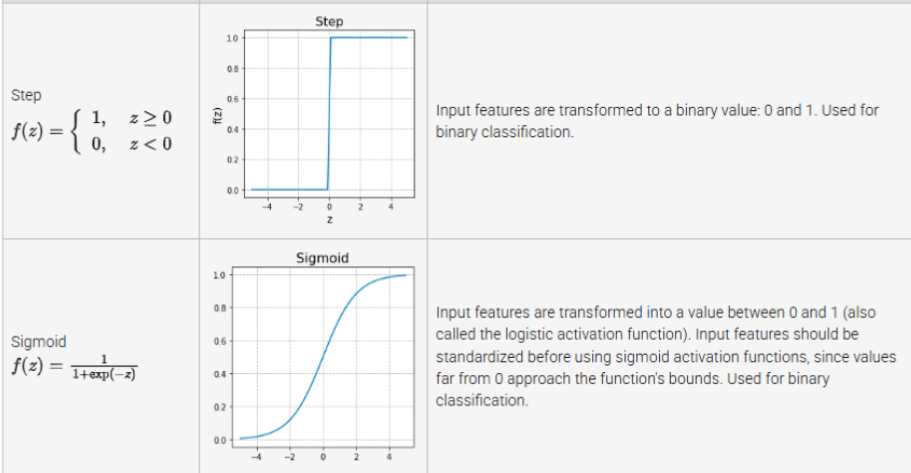
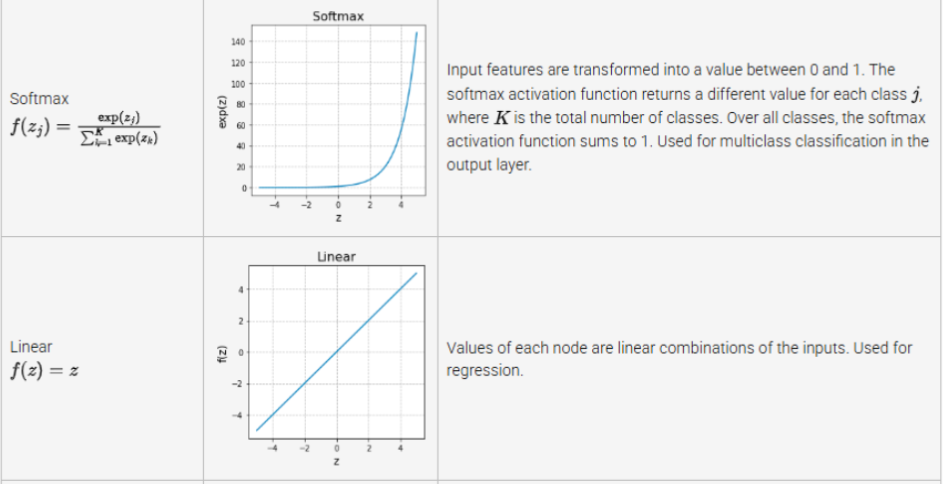
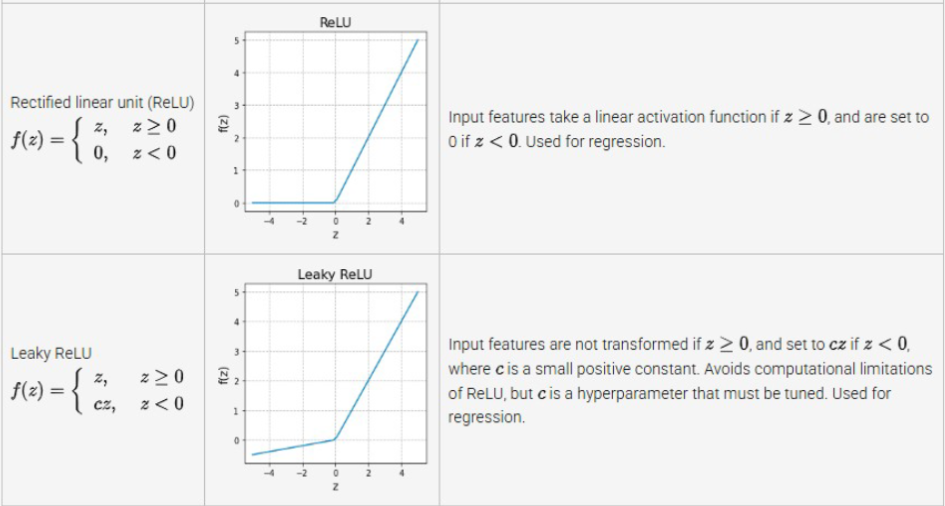
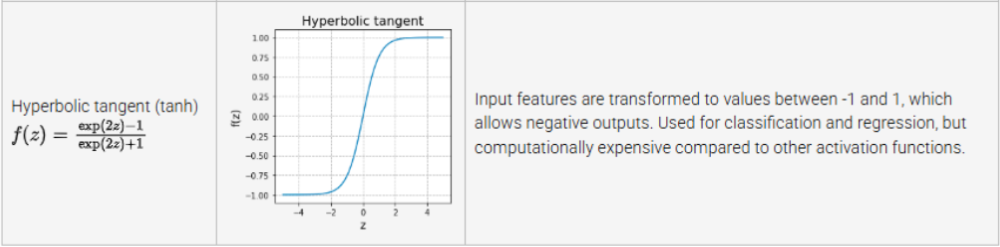
NN: mimic the brain to do computation

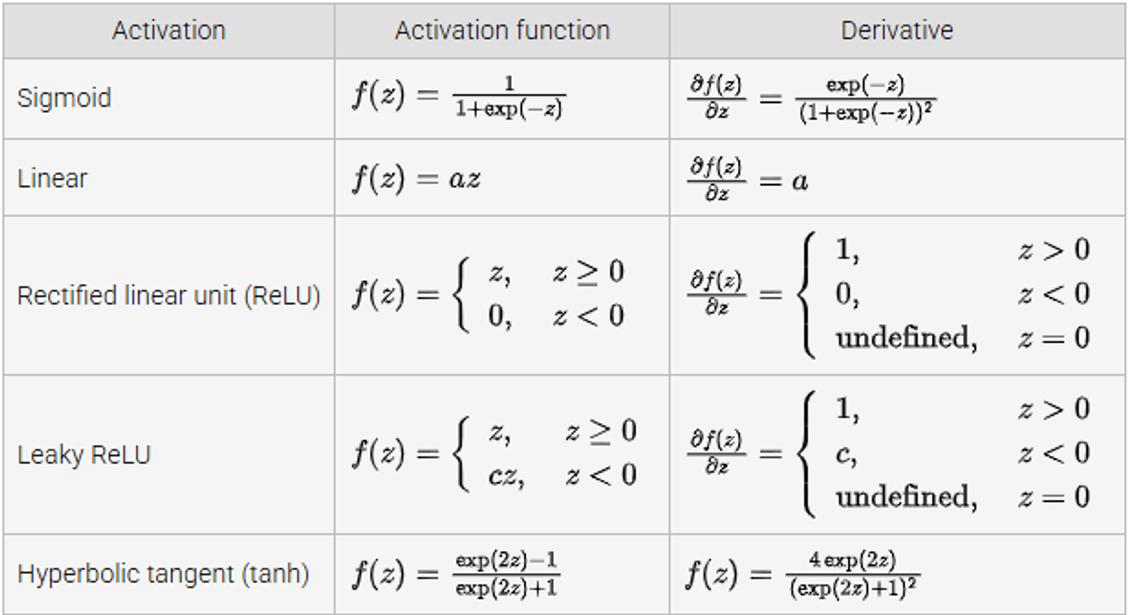
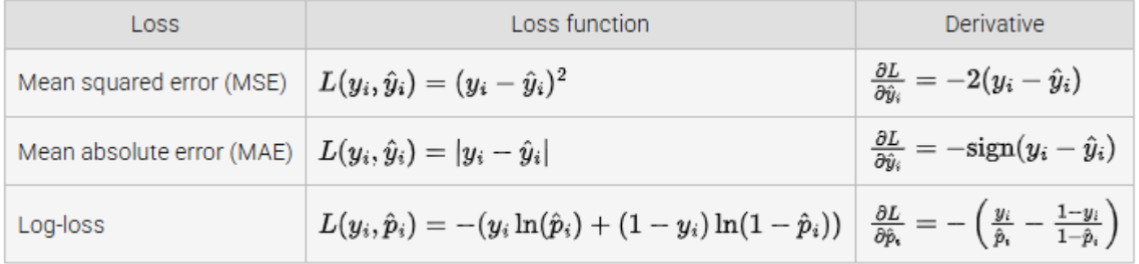


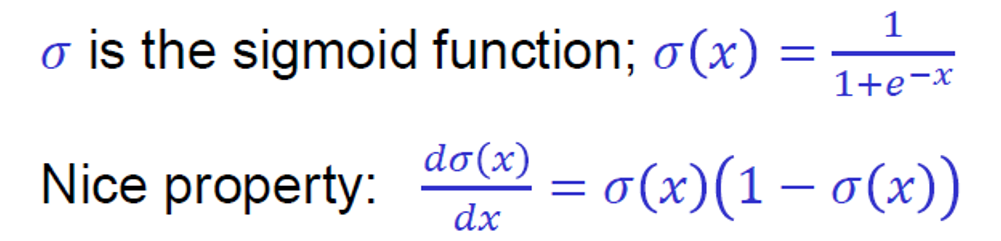
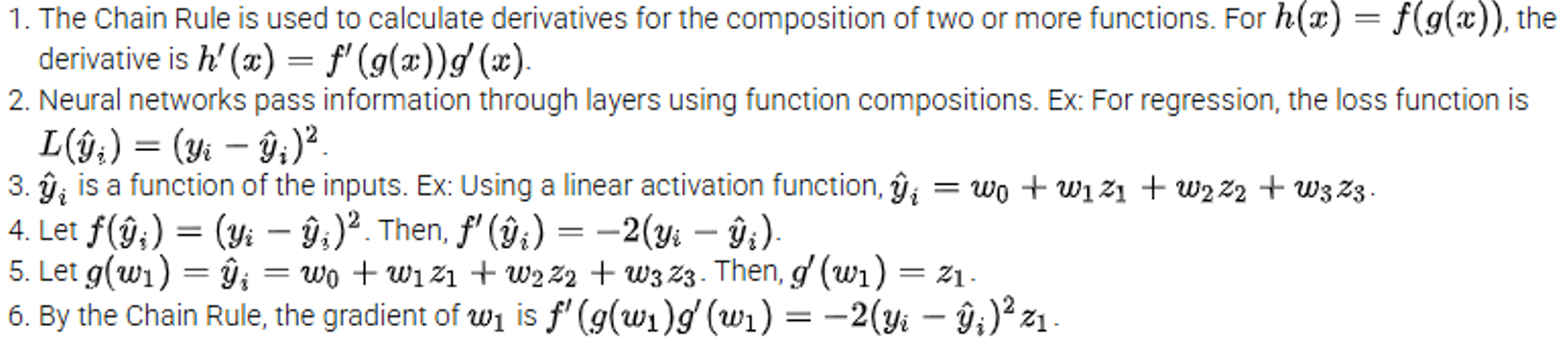






Gradient descent is an optimization algorithm that adjusts weights based on the loss function's derivative.

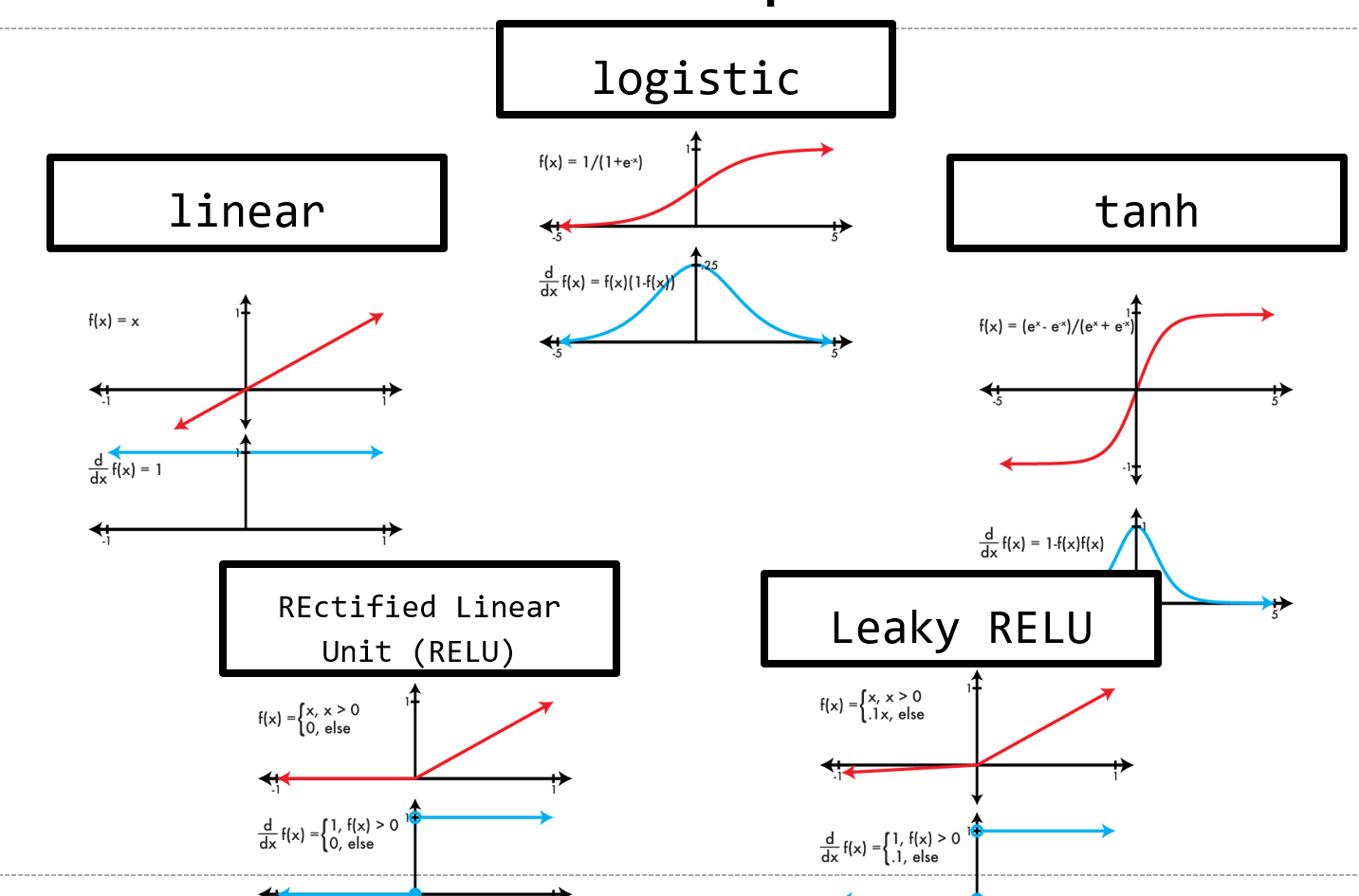




Make weak model more powerful: boosting!

Make strong model less likely to overfit: regularization

Neural networks are prone to overfitting



η = [.0001 - .01]

λ = .0005  
m = .9  
φ = leaky relu

Convolutional neural networks

Convolutional Layer:

Input: an image  
Processing: convolution with multiple filters  
Output: an image, # channels = # filters

Output still weighted sum of input (w/ activation)

Pooling Layer:

Input: an image  
Processing: pool pixel values over region  
Output: an image, shrunk by a factor of the stride

Hyperparameters:  
 What kind of pooling? Average, mean, max, min  
 How big of stride? Controls downsampling  
 How big of region? Usually not much bigger than stride

Most common: 2x2 or 3x3 maxpooling, stride of 2