Semi-supervised learning: loss fn

Salimans et al 2016 describe approach to loss fns. If you have a N way softmax classifier, then use a N+1 classifier to allow for a fake image. Then note only need a N way classifier - subtracting a constant does not change softmax probabilities

output neurons are x_1, x_2, x_3

Probability = $\frac{e^{x_i}}{\sum_j e^{x_j}}$

subtract a, so have $x_1 - a$, $x_2 - a$, $x_3 - a$

gives $\frac{e^{x_i - a}}{\sum_j e^{x_j - a}} = \frac{e^{x_i} \cancel{e^{-a}}}{\cancel{e^{a}} \sum_j e^{x_j}} = \frac{e^{x_i}}{\sum_j e^{x_j}}$

So set $y_1 = x_1 - x_3$, $y_2 = x_2 - x_3$, $y_3 = 0$ Therefore N+1 neuron is hardwired to 0

$P(\text{image fake}) = \frac{e^0}{\sum_{j=1}^{N} e^{y_j} + e^0} = \frac{1}{\sum_j e^{y_j} + 1}$

$P(\text{image real}) = \frac{\sum e^{y_j}}{\sum e^{y_j} + 1}$

Loss for a real image = $-\log(\text{prob real})$

$= -\left[\sum e^{x_i} - \log\left(\sum e^{x_i} + 1\right)\right]$

$= -\log \sum e^{x_i} + \log\left(\sum e^{x_i} + 1\right)$

Loss for a fake image = $-\log(\text{prob fake})$

$= -\left[\log 1 - \log\left(\sum e^{x_i} + 1\right)\right]$

$= \log\left(\sum e^{x_i} + 1\right)$

1) Loss is the cross entropy loss.

2) For a mini-batch, sum the loss.