January 2020

The Real World Weight Cross Entropy Loss Function

So they say that cross Entropy which is the standard Loss function does not do well when a class is underrepresented. So they present a new Loss function the Real World Cost function.

So normal catigorical cross entropy: for one example.

eq.0 $$-\sum_{\substack{\text{num} \\ \text{classes}}} \underset{\substack{\text{prob} \\ \text{of} \\ \text{lable}}}{y^k} \cdot \underbrace{\log(h_\theta(x))}_{\substack{\text{prob of lable} \\ \text{from model}}}$$

now people also said (not these guys) lets give more weight to the loss of the underrepresented classes

eq1 $$-\sum \underset{\substack{\uparrow \\ \text{weight} \\ \text{for class } K}}{w_k} \cdot y^k \cdot \log(h_\theta(x))$$

So then these guy come arond and propose a seprat weight for missing positive VS missing negative so.... that's pretty much What they do.

$$eq.2 \quad -\sum_{\text{num classes}}^{K} \underbrace{W^{k}_{mcfp}}_{\text{marginal cost of making a false posative}} \times y^{k} \times \log(h_\theta(x,k)) + \sum^{K'} \underbrace{W_{mcfn}}_{\text{marginal cost of makin a false negatvie}} \times y^{k} \times \log(1-h_\theta(x,K))$$

$K' \neq K$

So to test this they take MNIST ~7,000 of each class for say class "0" zero they lable the first 630 them as zero + the rest are left out. do this for the next 630 + so on that way 10 dataset + then 100 if done for each number. so for one dataset 630 (posative) / 63,000 (negative) so 1% posatives so very squed data

They set the MCFN High 2000 (cost of not identifying rare desies) +

the MCFP not so High. 100 (set to cost of a retest)

In the results they found that they had fewer false negatives but more false postives.

Then they find that in the real world this would result in lower cost. this makes sense if you set the marginal costs correctly then we would optimize for that.

☆ Note in the Binary case this is equivilant to doing eq.1 where the weight is a ratio of MFNC/MFPC

☆ Note the marginal cost are not Learned params But set By experts of the domain

find similar result in the multiclass case