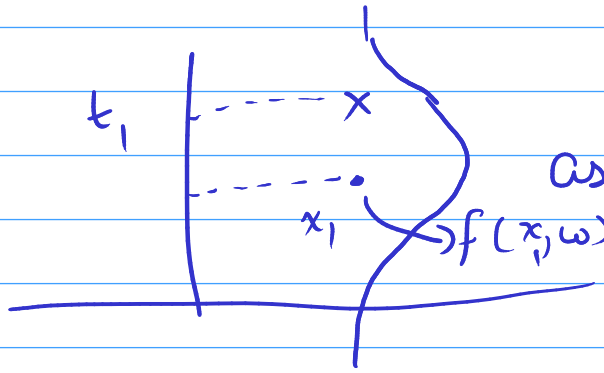
$$y/x \sim N(y | f(x,w), \tilde{I})$$

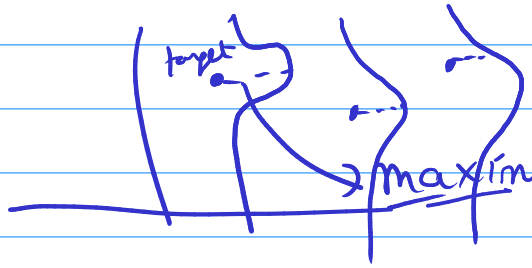


assume a normal distribution like this!

Consider covariance matrix to be I for now

now $f(X, w)$ is quite away from target

now lets have another w_2 such that



⇒ we need to find MLE (w)

Covariance matrix can also be determined

positive, semidefinite

you need a more complex neural network

What cost functions are good??

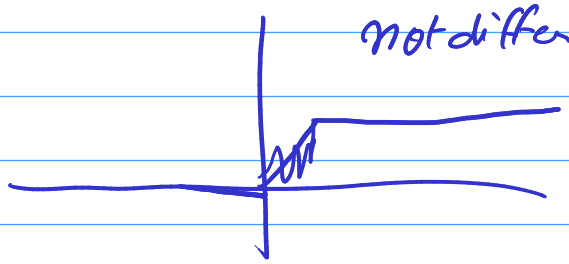
→ $P(y=1) = \sigma(w_0 + w_1 w + w_2 w \cdots + w_n w)$

what if

$p(y=1) = \max\{0, \min\{1, w^T h + b\}\}$

gradients are zero outside

not differentiable, also saturates



we need a better cost function !!

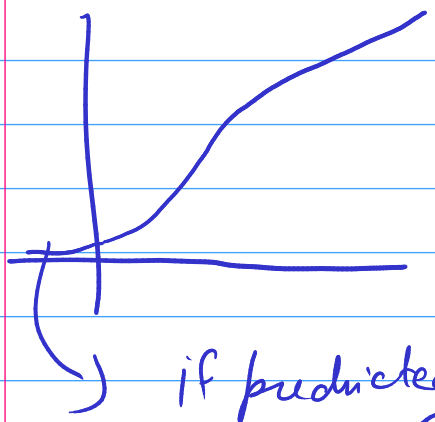
$-\log(P(\cdots))$ ...

$\Rightarrow$ we need a cost function that saturates @ correct classification & doesn't saturate @ incorrect classifications

$Y \mid X$	1	0
$P(Y \mid X)$	p	$1-p$

p is a bernoulli

Cost functions aren't forced, it's a consequence of an assumption of a normal distribution



if predicted output is low (-ve) but target is positive } Saturates

—(Z), if there is small change $Z \leq\!< 0$ then function won't change

i.e. $Z \to \infty$ or $Z \to -\infty$

and in an interval, it changes a lot!

My error function should change when im not near a global or local minima

in SGD, since you're doing it example by example

you want to improvise vs drastically

⇒ Function shouldn't saturate

Deriving Cost

$$\sigma(z) = \frac{1}{1+e^{-z}} = \frac{e^z}{1+e^z}$$

$$\frac{d\sigma(z)}{dz} = \sigma(z)(1-\sigma(z))$$

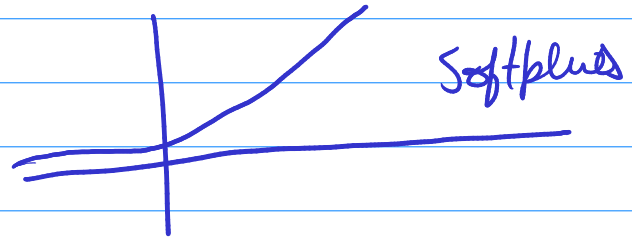
assume ← $\zeta(z) = \log(1+e^z)$ → softplus function

$x^+ = \max\{0, x\}$ Softer version of max function

$z \to +\infty \quad \log(1+e^z) \to z$

$z \to -\infty \quad \log(1+e^{-z}) \to 0$

very nice

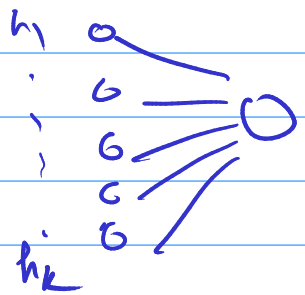


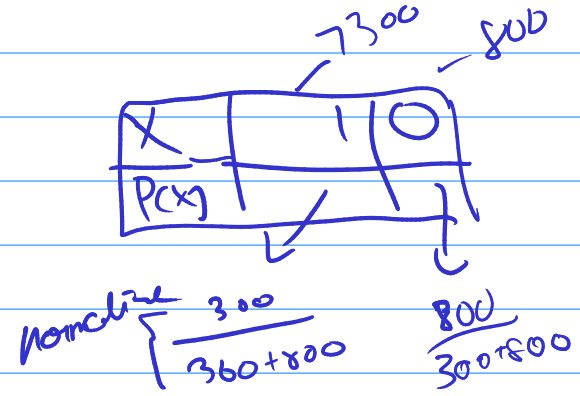
$$\zeta(1+e^{-z}) = \log\left(\frac{e^z+1}{e^z}\right) = \log\left(\frac{1}{\sigma(z)}\right)$$

$$= -\log(\sigma(z))$$

$$\log(\sigma(z)) = -\zeta(-z)$$

$$z = w_0 + w_1 h_1 + \cdots + w_k h_k$$





I want to combine z with target to arrive @ cost function that doesn't saturate

So they choose yz as that combination such that

$$\log(\hat{p}(y)) = yz = \hat{p}(y) = e^{yz}$$

$$p(y) = \frac{e^{yz}}{\sum_{y'} e^{y'z}} = \frac{e^{yz}}{1+e^{z}}$$

(Normalized)

(y takes values 0 & 1)

A "kind of" probability distribution

$$p(y=1) = \frac{e^{z}}{1+e^{z}} = \frac{1}{1+e^{-z}} \quad p(y=0) = \frac{1}{1+e^{z}}$$

$$\Rightarrow p(y) = \sigma((2y-1)z)$$

When $y=1$, $2y-1=1$

$y=0$, $\sigma(-z)$

$= \frac{1}{1+e^{-(-z)}} = \frac{1}{1+e^{z}}$

Cost function of regular logistic regression also saturates

$$p(y_1 \cdots y_m | x_1 \cdots x_m) = \prod_{i=1}^{m} p(y_i/x_i)$$

$$\max \log\left(\prod_{i=1}^{m} p(y_i/x_i)\right)$$

$$\max \sum_{i=1}^{m} \log(p(y_i/x_i))$$

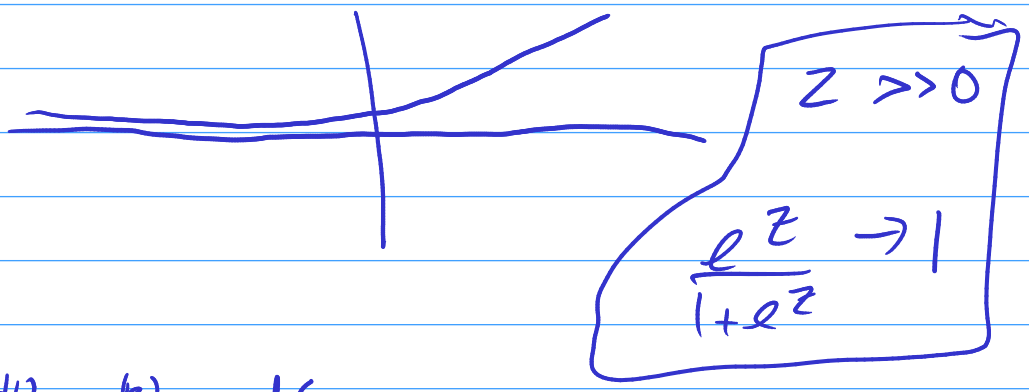
$$\min \sum_{i=1}^{m} -\log(p(y_i/x_i))$$

Cost func: $y(0) = -\log p(y|x)$

$$= -\log(\sigma((2y-1)z)) = \zeta(-(2y-1)z)$$

$= \zeta((1-2y)z)$

remember ζ is the softplus function



$w^{(k+1)} = w^{(k)} - \eta \frac{d\mathcal{E}}{dw}$

gradient should be small // $z \gg 0$, & prediction is & is correct

this ζ function achieves that, as, if $y=1$

$\zeta((1-2y)z)$

$\zeta(-z) \approx 0$ since $z \gg 0$

$z \gg 0$, $y=0$, you want high gradients

then $\zeta(z) = \to z$

you get

Why for

$z \ll 0$ $y=0$ then you get $\zeta(z) \to 0$

$z \ll 0$ $y=1$ $\zeta(-z) \to (-z)$

Why that cost function
WTF is wrong with what I have?!

$$P(y_1=1/x_1) = \sigma(z_1)$$

$$P(y_1=0/x_1) = 1-\sigma(z_1)$$

$$P(y_i) = (\sigma(z_i))^{(y_i)} (1-\sigma(z_i))^{(1-y_i)}$$

$$p(y_1, y_2, \dots y_m | x_1, x_2 \dots x_m) = \prod_{i=1}^{n} (p(y_i/x_i))$$

$$= \prod_{i=1}^{m} (\sigma(z_i))^{y_i} (1-\sigma(z_i))^{1-y_i}$$

$$= \log (p(y_1, y_2 \dots y_m | x_1, x_2 \dots x_m))$$

$$\max \sum_{i=1}^{m} \left[y_i \log(\sigma(z_i)) + (1-y_i) \log(1-\sigma(z_i)) \right]$$

$$= \min \sum_{i=1}^{n} -\left(y_i \log \sigma(z_i) + (1-y_i) \log(1-\sigma(z_i)) \right)$$

$z >> 0$ $y=1$ → saturates

$z >> 0$ $y=0$ also saturates ;-;

$\sigma(z)$ saturates

⇒ $\log(1-\sigma(z))$ saturates (ultimately, when $\sigma(z)$ doesn't change much at all)

⇒ problem!!

Up next: Softmax, Order & how to deal with them!

Alright, Lets say m outputs are there

Then how do we predict

$(w^T h + b)$ is input

$(w^T h + b) = z_i = \log \hat{p}(y=i/x)$

unnormalized log probabilities

$\Rightarrow e^{w^T h + b} = \hat{p}(y=i/x)$

$$p(y=i/x) = \frac{e^{z_i}}{\sum_{i=1}^{m} e^{z_i}}$$ for m classes

Why distribution? This is a multinomial dist.

→ so we can "turn" z_i into a probability this way.

→ we dont normally maximize THE probabilities so we ~~not~~ maximize the (unnormalized) log probabilities

then we normalize

Softmax activation $= \frac{e^{z_i}}{\sum_{i=1}^{m} e^{z_i}}$

$$\min\left(-\log \prod_{i=1}^{m} \left(p(Y=2|x_i)\right)^{\textcircled{1}} \left(P(Y=2|x)\right)^{0} \left(P(Y=2|x)\right)^{1}\right)$$

(1 one hot encoding)

(1 since this is class 2; 0, since this is class 1)

It is however not for sure that you'll hit local minima after some iterations

we might need early stopping

Other activation funcs

ReLU: differentiability problem @ '0' take either LHS, or RHS derivative

Leaky ReLU

$h_i = g(Z_A) = \max(0, z_i) + \alpha_i \min(0, z_i)$

$\alpha = -1 \Rightarrow$ absolute value activation function

0

0

0

0

0

0

0

0

0

take max of each $w^T x + b$ in each division and all neurons in that division have max of division in that output (maxout)

This is maxout

Don't use sigmoid, tanh(z) saturates lesser, use that instead
(make sure you shift origin of this graph)

radial basis

softplus $\log(1+e^z)$

hard tanh $\max(-1, \min(1, a))$