8-11-2021 Monday

week # 11 Lecture # 13

Activation Functions:-

① Sigmoid Function:

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

ⓐ Always +ve output $[0, 1]$

ⓑ Saturated output $-4 > < 0 < +4$

② Tanh Function:

$$Tanh(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}}$$

OR

$$Tanh(x) = 2\sigma(x) - 1$$

a) Zero Centered $-1 < 0 < 1$
output not always +ve.

b) Saturated output $-7 < 0 < +7$
atleast better than Sigmoid Function.

③ ReLU. Function:-

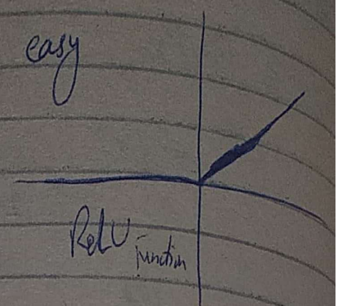
$$ReLU(x) = max(0, \omega x + b)$$

a) Convergence is fast

b) No -ve values as output

c) Easy to compute relatively easy

d) No saturation



e) ~~Dieng~~ Dying ReLU:

or some 50% neurons could be dead like there output is zero

4) Leaky ReLU Function.

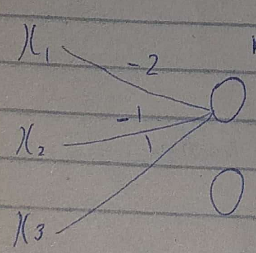
$$\text{Leaky ReLU}(x) = \max(0.01(z), z)$$

where $z = wx + b$

a) Now gradients aren't zero & output is also not zero.

e.g.:-

$\max(0.01(-2), -2) = \max(-0.02, -2)$
$= -0.02$



usually α is 0.01

$$\text{Leaky ReLU}(x) = \max(\alpha z, z)$$

where $z = wx + b$

5) PReLU Function :- Parametric ReLU.

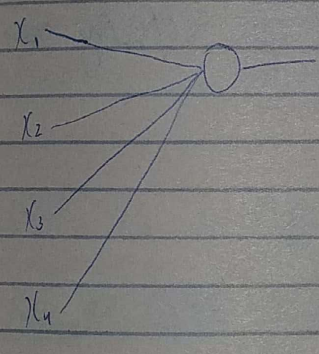
we learn our "α" as well

$$PReLU(x) = max(\alpha z, z)$$

$$\boxed{\frac{\text{output of Neurons Zero}}{\text{Total Neurons in layer}} = \frac{5}{10} = 0.5 \quad \text{Dead Neurons}}$$

Data Pre-Processing:-

Problem #1: Zero centered:



all inputs +ve , then gradients will also be +ve

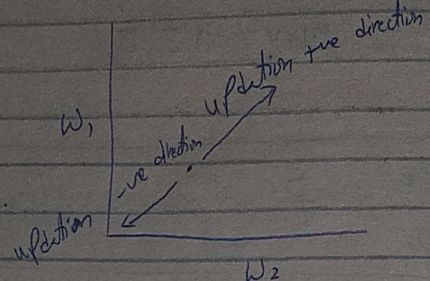
$f = wx$

$dw = \underset{+ve}{x}$

So our updation of weights will be in one direction.

$$W = w - \alpha dw$$

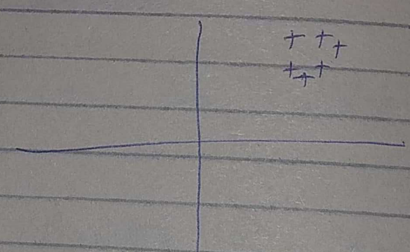
either in +ve or -ve direction.

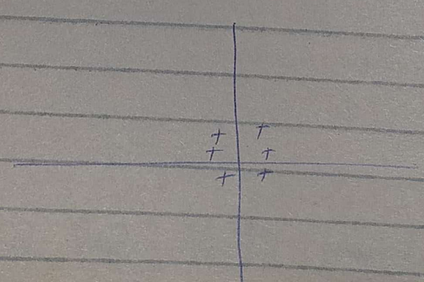


So, problem is weights updation exploration is not good.

<u>Solution:</u>

make Input Zero centered by subtracting mean from the input





Zero centered.

$$X = \underbrace{X - \mu}_{\text{mean}}$$

Now, how do we calculate our mean?

20	30	X_1
20	30	

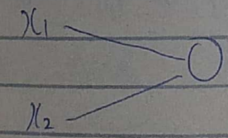
10	60	X_2
60	10	

70	80	X_3
90	100	

$\frac{20+10+70}{3}$	$\frac{30+60+80}{3}$	Mean Image
$\frac{20+60+90}{3}$	$\frac{30+10+100}{3}$	

subtract this from every input.

Problem #2 Normalization:-
inputs are at different scale



$x_1 \to 17 - 70$

$x_2 \to 0 - 100000$

So if we want to do Linear Regression

then x_2 would have more ~~weightage~~ impact & x_1 would have less impact in our output

Solution:

$$Z = \frac{X - X_{min}}{X_{max} - X_{min}}$$

e.g.

	$X - X_{min}$	$\frac{X - X_{min}}{X_{max} - X_{min}}$
3	0	0
4	1	1/7
5	2	2/7
8	5	5/7
9	6	6/7
10	7	1

max value = 1

min value = 0

Now, both x_1 & x_2 will be in between 0 - 1 values.

Now, in images case, we calculated min image by taking minimum in each dimension

10	20
20	10

from prev example.

similarly, we calculate max image, then we can do Normalization on images.

either apply #3 or (#1 & #2 together)

Problem #3 Standardization:

$$x_i = \frac{x_i - \mu}{\sigma}$$

σ = standard deviation

First we make it zero centered then we divide it with standard deviation of data set.

Now, input has 2 properties.

a) Zero centered - Mean zero

b) Standard deviation = 1

Problem #4: Initialization of weights: Xavier Initialization

when we initialize weights with zero, in forward or backward pass weights & outputs are same. This is called assymetric.

we need to break symmetry.

we initialize weights randomly

$$W = \boxed{np.random.randn(n, D)}$$

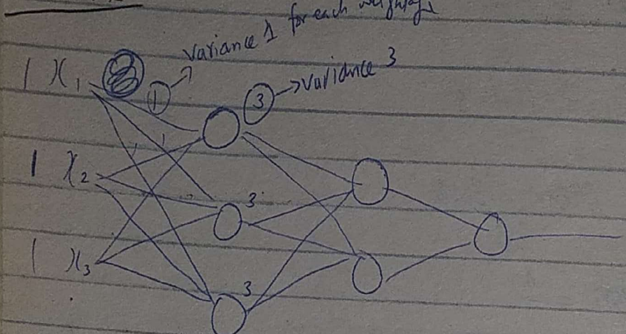
n = no. of inputs for this layer.

D = no. of neurons in this layer.

$(-1 - +1)$ from standard distribution. Standard deviation & variance is 1, before this solution, variance was increasing when moving in forward pass.

So weights standard deviation & variance is 1.
Now if inputs are all 1, Then output would be
3 & variance would be 3.
Now, overall outputs gradient explode...

Solution:



Solution is That we divide weights by $\sqrt{n}$, n is no. of inputs

$$W = np.random.randn(n, D)/\sqrt{n}$$

$Var(z) = n(1) = n$ before the solution

So that's why we divide by $\sqrt{n}$ so that standard deviation of output remains 1.

$\frac{W}{\sqrt{n}}$ when we divide by n, we make variance remain 1.