Djk7HX

Lecture -22 Djk7HX

Lecture started with a brief information about project ~~for~~ submission, paper presentation ~~for~~ dates and submission and Homework 4 brief information and road map to follow to solve it.

* A problem:- The vanishing gradient:

With large no of layers $\left(\delta^{(l)}\right) \to 0$

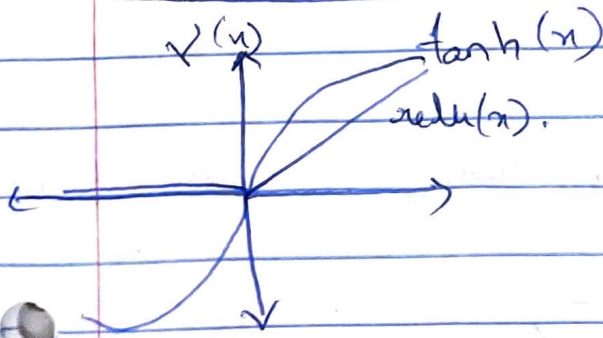
$$G_1^{(l)}(x_n) = \left[x^{(l)}\left(\delta^{(l)}\right)^T\right]$$

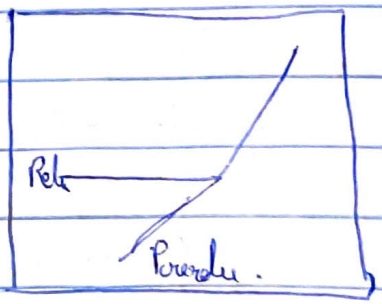
$$w^{(l)} \leftarrow w^{(l)} - \eta G_1^{(l)}$$

As $\delta^{(l)} \to 0$ then $G_1^{(l)} \to 0$.

So ~~the~~ When we try to update weight using above formula. There is no change noticed.

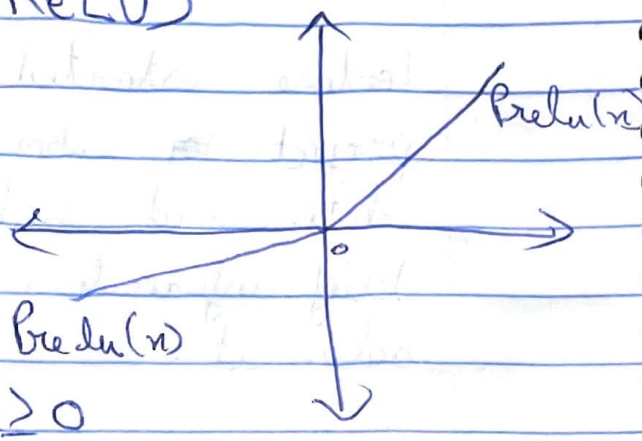
* A solution:- The ReLU (Rectified Linear Unit)





Benefit:- Helps in creating sparse values as derivative is 0 of values < 0

* Parameterised Relu (PReLU)



$$\frac{d}{dx} relu(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ 0 & \text{else} \end{cases}$$

$$Prelu(x) = \begin{cases} x & \text{if } x \geq 0 \\ ax & \text{else.} \end{cases}$$

↓

learnable parameter.

* A similar Problem :- Exploding Gradient.

Artificially scaling down ~~to a scale~~ so a norm is under budget.

* Breaking the symmetry during initialization.

Ways to handle are :-

1] Random initialization (Uniform Distribution)
2] Xavier initialization (Uniform distribution from network statistics)

* Dropout Regularization.

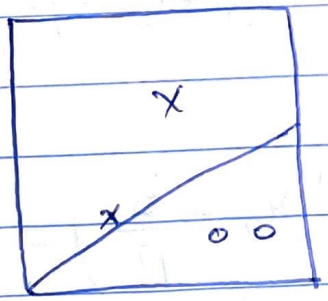
Masking few nodes at random

$$x_i^{(l)} = \theta\left(p_i^{(l)} s_i^{(l)}\right)$$

randomly masking the nodes by making the output 0.

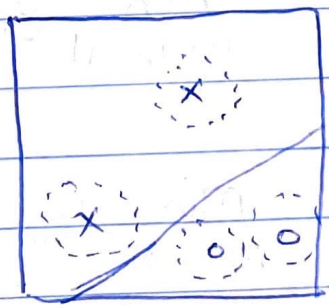
Dropout rate ~~needs~~ can be specified at each layer.

* Robustness to Noisy Data.



$E_{in} = 0$

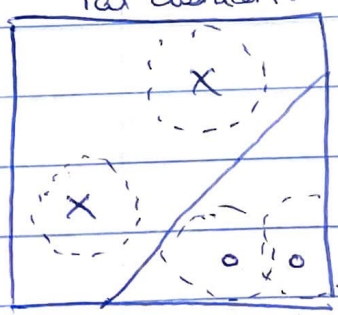
But less robust
(Can handle less error)



$E_{in} = 0$

more robust

Fat cushion.



$E_{in} = 0$

has the most robust
(Can handle
noisy data
more efficiently

Increasing the thickness effectively reduces the Hypothesis set.
Thick cushion makes our model more efficient it is equivalent to picking the hypothesis which is regularised.

Seperating Data.

$$\text{sign}\,(w^T x_n + b) = y_n$$

$$\equiv$$

$$y_n (w^T x_n + b) > 0 \qquad \text{(always +ve)}$$

$\Rightarrow$ there is a ρ such that.

$$\min_{n=1,\dots N} y_n (w^T x_n + b) = \rho$$

$$\min_{n=1,\dots N} y_n \left(\frac{w^T}{\rho} x_n + \frac{b}{\rho} \right) = 1.$$

rescaling the weights, identical hyperplane

Remember this fact.