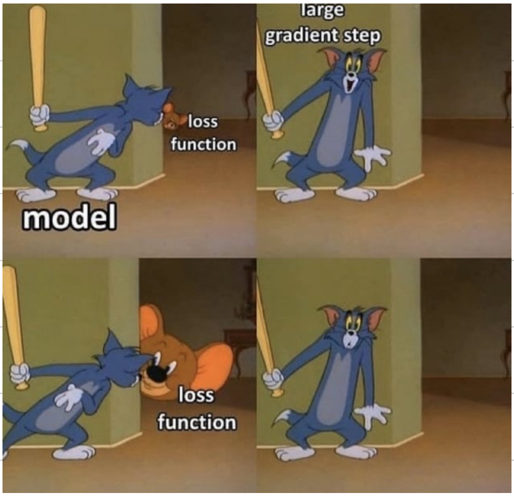
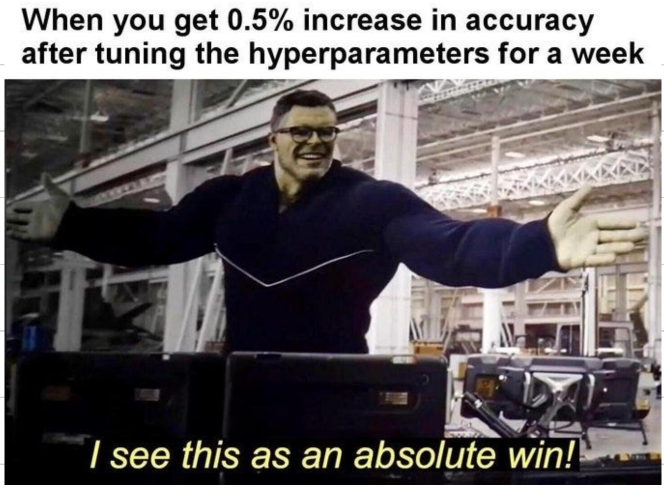
Session - 9 # Hyperparameter - Tuning Feb 26, 2024





Agenda

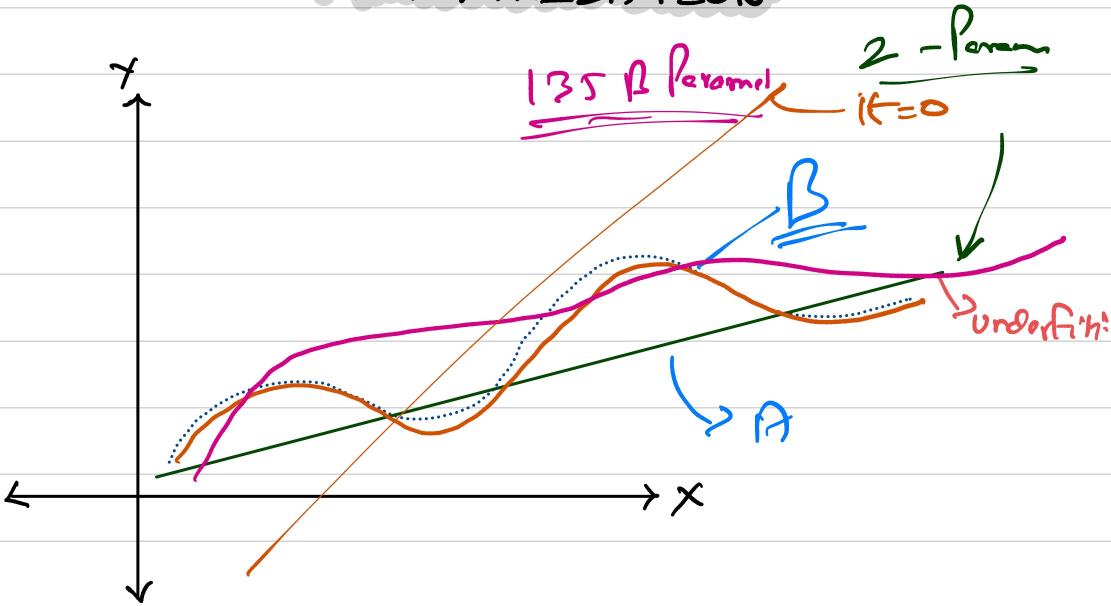
1. Regularization
2. Dropout
3. Batch Normalization

TARGET ENCODING

X	Y (Sold / NotSold)		Target_En(X_tr)
Maruti	1	$\frac{1+1+0}{3}$	2/3
Maruti	1		2/3
Maruti	0		2/3
BMW	0	$\frac{1+0}{2}$	0.5
BMW	1		0.5
RR	1	$\frac{1+1}{2}$	1
RR	1		1

X	Y (Price)		Target_En(X_tr)
Maruti	1.2	$\frac{1.2+3.4+5.2}{3}$	
Maruti	3.4		
Maruti	5.2		
BMW	40	$\frac{100}{2}$	50
BMW	60		50
RR	800		
RR	260		

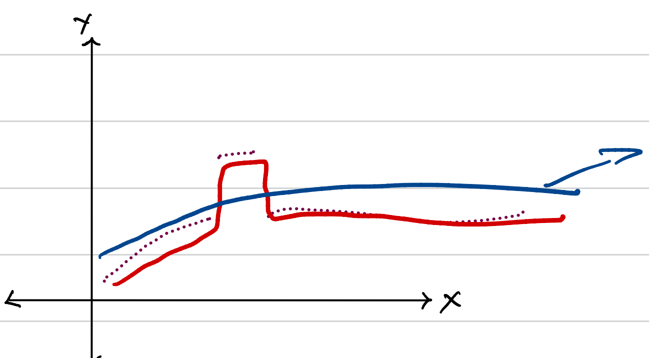
Regularization



it: 0   mse → ↑ : 100

it: 5   mse : 20

Loss-function + λ (Constraint)



Penalize wt metrics

Mr:

$$\sum_{i=1}^{n} L(\hat{y}, y) + \frac{\lambda}{2n} \cdot \underbrace{\sum_{j=0}^{m} ||w_j||^2}_{L2}$$

$$+ \frac{\lambda}{21} \sum_{j=0}^{n} |w_j|$$

$L2 \longrightarrow$ squashe matrix (defuse)

$L1 \longrightarrow$ sparse matrix

Howt $\rightarrow$ compatible with NN

no of layers

Frobenius Norm

Reg : $\dfrac{\lambda}{2n} \sum_{k=1}^{k=L} \boxed{\underbrace{||w_F^k||^2}}$

no of neurons in prev layer

no of neurons in next layer

$$||w_F^k||^2 = \sum_{i=1}^{n^{k-1}} \sum_{j=1}^{n^k} (w_{ij}^k)^2$$

Ea. $w^k = w^k - \eta * \dfrac{dL}{dw}$

$$\frac{dL}{dw} = \underbrace{dw}_{\frac{\partial L}{\partial w}} + \frac{\lambda}{\cancel{2}n}(\cancel{2} * w^k) \rightarrow \frac{\partial ||w^k||^2}{\partial w^k}$$

$$= dw + \frac{\lambda}{n} * w^k$$

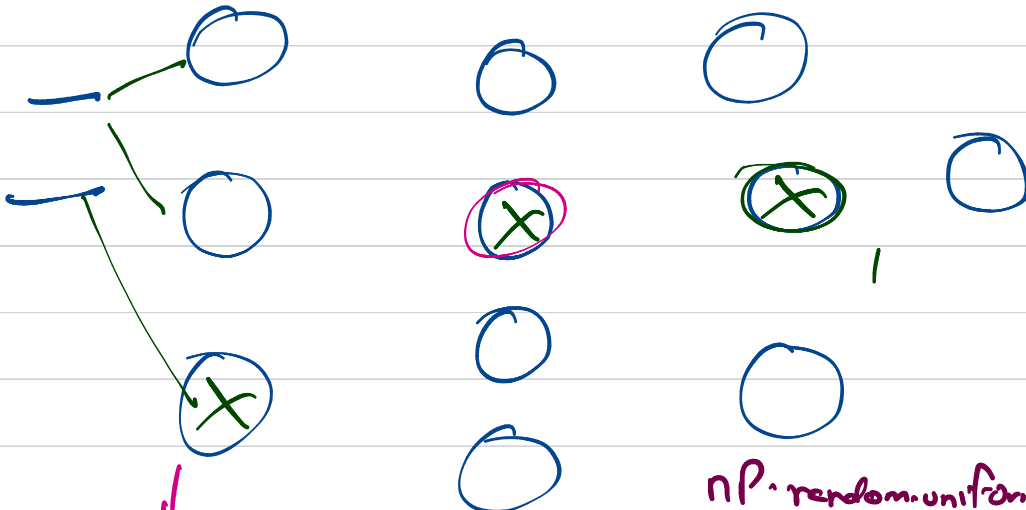
$$w^k = w^k - \eta\left(dw + \frac{\lambda * w^k}{n}\right)$$

$$= w^k - \eta * dw - \eta * \frac{\lambda w^k}{n}$$

$$= \underbrace{w^k\left(1 - \eta * \frac{\lambda}{n}\right)}_{\text{Reduction}} - \eta * dw$$

<u>decay</u> $\left(1 - \eta * \frac{\lambda}{n}\right)$ = weight decay

DROPOUT



np.random.uniform(

0–1 → every single iteration

dropout 0.2

$$\frac{1200}{24}$$

$$\frac{1200}{24}$$

100 (20 drop)

80%

10000 → $10000 \times \frac{80}{100} = 8K$

dropout → 0.2

Train a model

2

8K Para

$20/30 \quad \left(\frac{10}{30} \right)$

$30 \times p = 10$

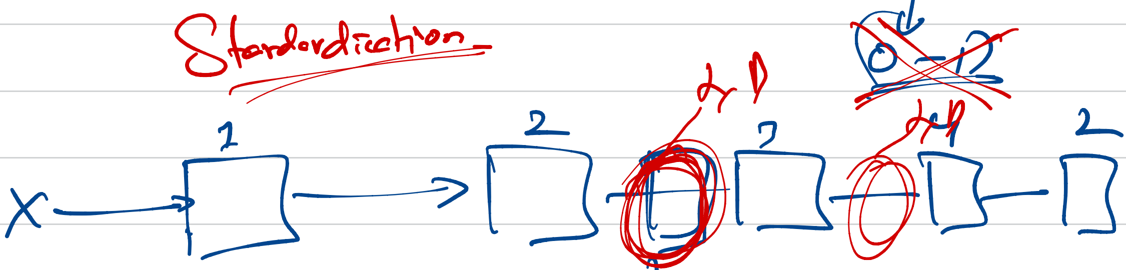
0

$w \rightarrow w/(1-\text{dropout})$

$= \frac{w}{0.8}$

Batch Normalization

Standardization



$\mu, \sigma \rightarrow$ 0 — mean
1 — var

$$z = \frac{x - \mu}{\sigma}$$

Restricting NN to generalize

$\alpha \times z + \beta \rightarrow$ shift

Scale

$\infty \rightarrow 0$

$0 - 6 \times 2$

$\alpha, \beta \rightarrow$ Trainable Parameters

$\rightarrow$ Standardize

relu (w x + b)

256 → 1024 Param

512 (mean, std) / 512 (α/β)

μ σ

α β

moving average across all batch (for that neuron)

1000 batch

$\mu = 1$

EMA 0.9

if: 1 → $\mu_1, \sigma_1, \left(\frac{x-\mu_1}{\sigma}\right)$

$\mu = 0.9 \times 1 + (1-\alpha) \times \mu_1$

if: 2 → $\mu_2, \sigma_2, \left(\frac{x-\mu_2}{\sigma_2}\right)$

$0.9 \times \mu + (1-\alpha) \times \mu_2$

$\vdots$

$0.9 \times \mu + (1-\alpha) \times \mu_i$