

## Improving Neural Network performance

### 1) Vanishing Gradients

- Activation functions
- Weight initialization

### 2) Overfitting

- Reduce complexity / Increase Data
- Dropout layers
- Regularization ( $L_1$  and  $L_2$ )
- Early stopping

### 3) Normalization

- Normalizing inputs
- Batch Normalization
- Normalizing Activations

### 4) Gradient checking and clipping

### 5) optimizers:-

- momentum
- Adagrad
- RMSprop
- Adam

### 6) learning rate Scheduling

### 7) Hyperparameter Tuning

- No of hidden layers
- nodes / layers
- Batch size

## Early Stopping in Neural Networks:

If we train our data in very higher epochs it will lead us in overfitting. So, to get free from overfitting we will implement Early Stopping concept of the keras.

Hanno model train garda if overfitting vayo, accuracy improve vayena, loss decrease vayena vaye. Early Stopping ley automatically epoches rokdiya ta hami lai final result diya (parameters values)

keras ma we use call backs to implement the Early Stopping.



Callback = EarlyStopping (

monitor = "val\_loss",

min\_delta = 0.5

patience = 5

verbose = 1 (0 or 1)

mode = "auto" (fix it)

restore\_best\_weights = True

.)

### Arguments:

• monitor = "val\_loss". It will look for val loss either our val-loss is decreasing or not.

• min\_delta = 0.5. paila epoch ko best val\_loss se current epoch ko val loss ko value diff value  $\geq 0.5$  vayo vane matra further epoch me train hunxa natra early stopping triggered hunxa ani further training hunna.

### Example:

1<sup>st</sup> epoch: val\_loss = 5

2<sup>nd</sup> epoch: val\_loss = 4.5 (prev. <sup>best</sup> current = (5 - 4.5) = 0.5) train continues

3<sup>rd</sup> epoch: val\_loss = 6 (prev. <sup>best</sup> current = (4.5 - 6) = -1.5) training stops  
(As -1.5 is less than 0.5 so, further training stops)

• patience = 5. 5 ota epoch samma kura val\_loss (ghatna) improve hunxa ki nai.

• verbose = 1. Displays output as early stopped applied on screen. 0 gives no information in screen about early stopping applied

• mode = auto. keras auto detects what to do. min, max also can be set. val\_loss min vako observe gar vaneko max ma chai accuracy improve gareko observe gar vaneko tesko lagi (val-accuracy rakhu parxa monitor ma)

• restore\_best\_weights = True. yesle chai paila kunai epoch ma hamro accuracy aako or hamro loss ghatko wala parameters la save or set garxa.