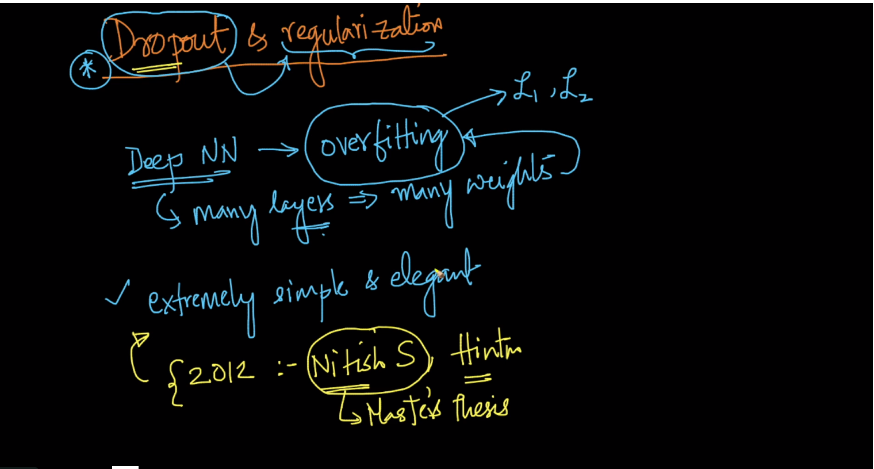
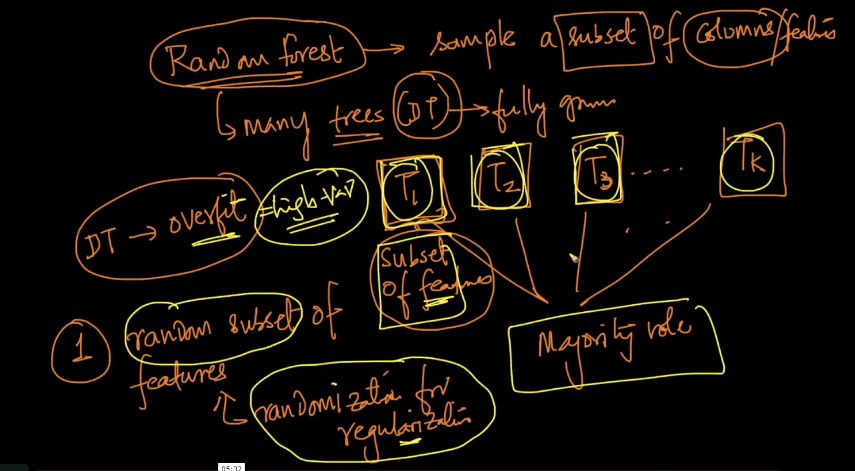
**Dropout layers & Regularization**

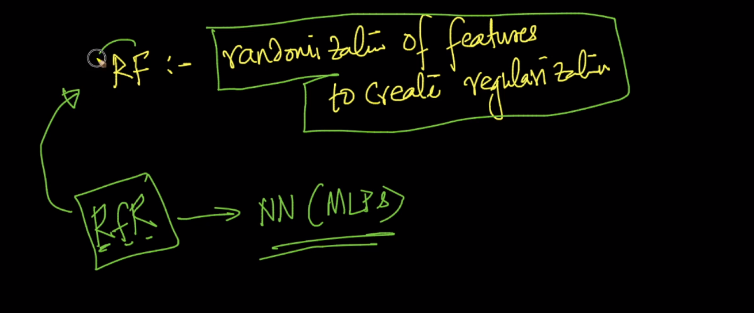
In Deep neural networks as there are many layers therefore there are many weights and this cause overfitting, we used regularization L1, L2 to overcome this problem

Another approach to overcome this problem is Dropout layers.



Dropout layers is inspired by the random forest as in this we sample a subset of features/columns to overcome high variance(overfitting) i.e we use randomization for regularization similar things we use in Dropout layer.

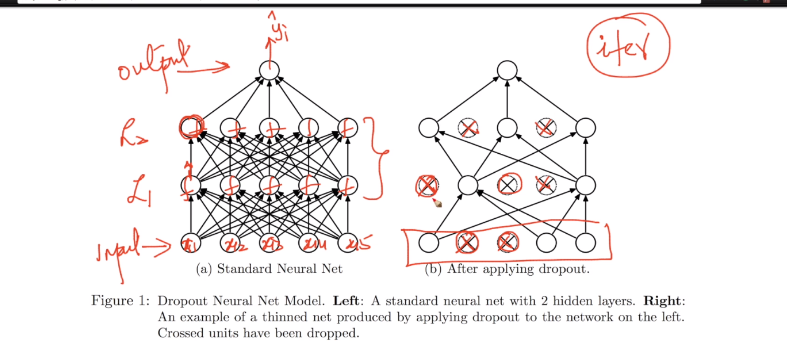


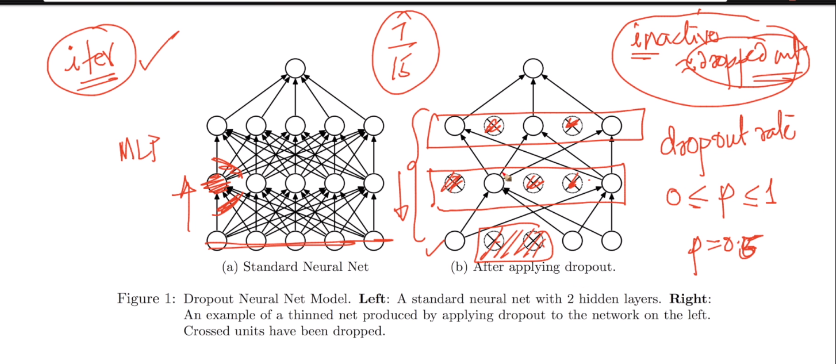


In dropout we inactive some neurons or inputs to randomized for regularization.

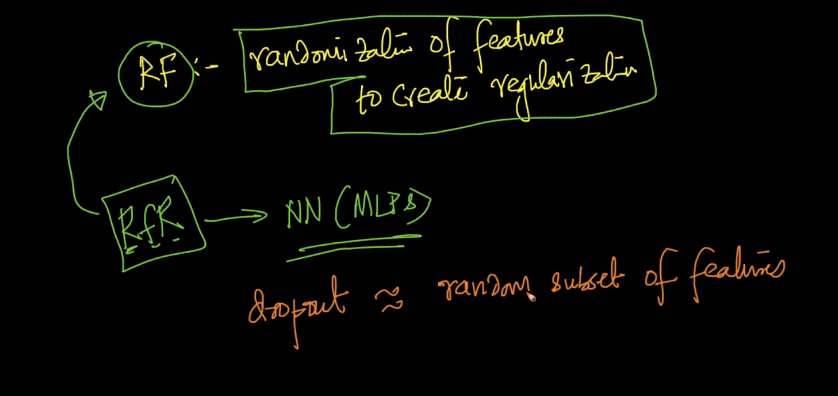
There is drop out rate which is between 0 and 1 which means how much percentage of neurons to be inactive(disconnect all the input and output connections) in a layers as in below image we use p = 0.5 that means 50% of neurons to be inactive in layers.

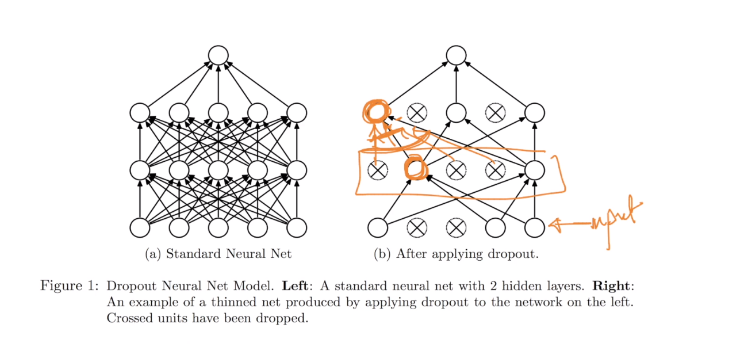
SO in each iteration it selects random 50% and it selects another or different neurons in next iteration





Dropout is just like a random subset of features because for any neuron all incomings are features only.



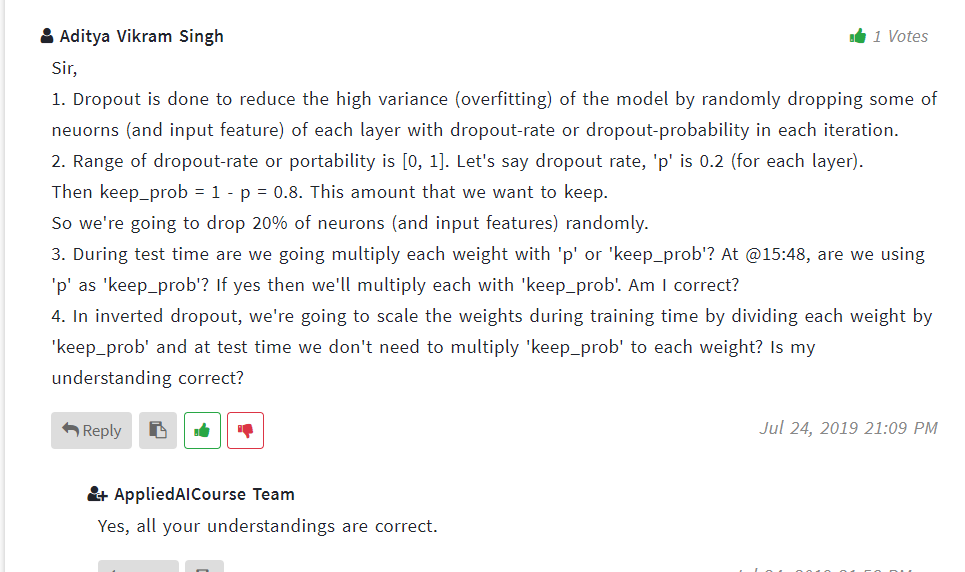


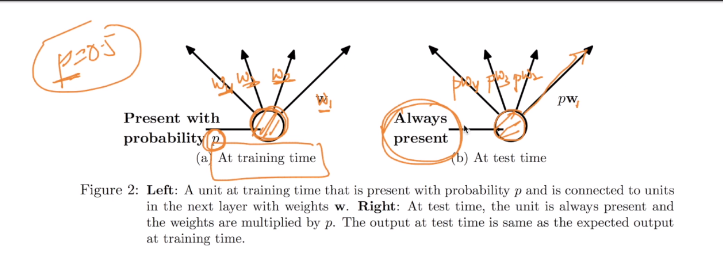
So this things happen at training time but what happens at test time.

So in training time every neuron is present with probability p mean p percentage of time it appears.

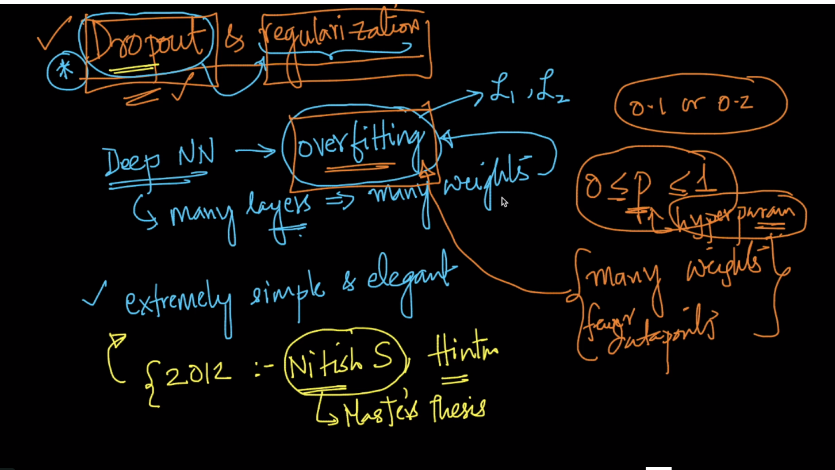
In testing time we multiply p of neuron with weights which means this weight present at p percentage of time

**Note : this p is probability not drop out rate p both are different.**

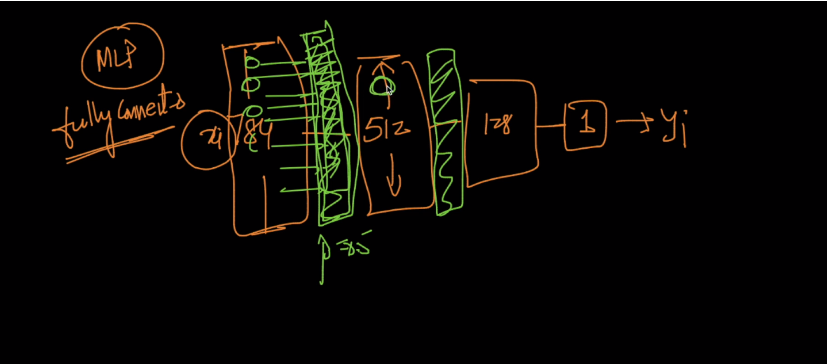




Therefore dropout rate p is a hyperparameter here if overfitting is large then we have to choose large p.



We can also think of this as dropout layer because we select some neurons from layer and we can think this as selected neuron is a new layer



Comments :

