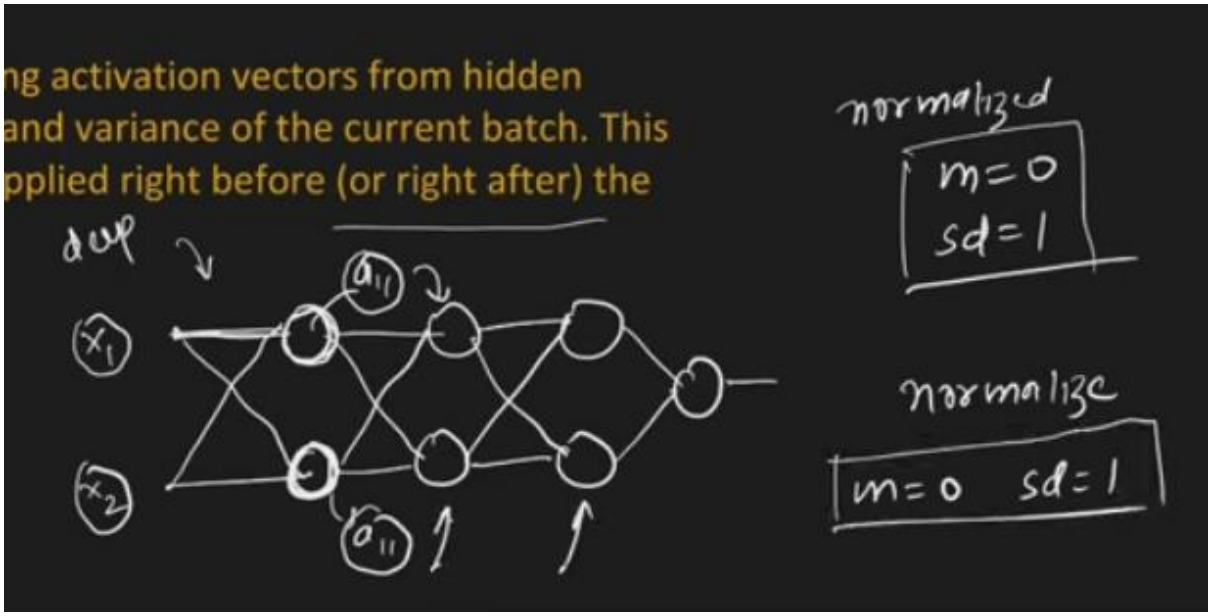
Batch Normalization

Batch-Normalization is an algorithmic method which makes training of Deep Neural Network (DNN) faster and more stable.

It consists of normalizing activation vectors from hidden layers using the mean and variance of current batch.

This Normalization step is applied right before (or right after) the Non Linear(Activation) Function.

Batch Normalization karnese Training bhi fast hota hai & Stable bhi hojata hai.



Jo Input layer hota hai mtlb inputs hote hai

Unka Mean = 0 rehta hai & SD 1 rehta hai

& jbh vo pass hote hai hidden layers mai

Then suppose 3 Hidden layer hai

So jo 1st Hidden layer mai input gaya tou, 1st Hidden layer mai jo input hota uska jo Ouput hota hai

Vo Output ,

Agle Hidden Layer ka mtlb,2nd Hidden Layer ka Input hota hai,

& jo 1st Hidden layer ka output hai vo barbar change hote rehta So barbar changed inputs milta agle hidden layer ko so yeah Kindoff Unstability hai jishe hum kehte hai Internal Covariate Shift.

SO Batch Normalization yeahi krta hai Normal distribution banadeta jishse aageh stability milljata

So vahi Output ko hum normalize krte hai mtlb mean =0 & SD=1

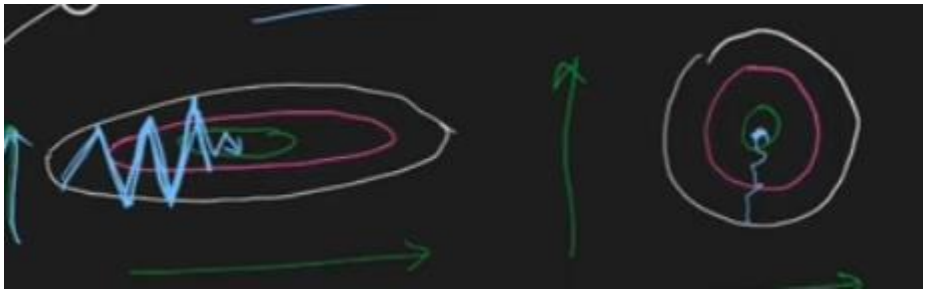
So ese hum saare hidden layers ke output ko Normalize krte jaate hai

Ki SO further Hidden layers ko input mille vo Normalize rheah

Ese kaha jaata hai

Saare Inputs ko same scale pe rehna chaye or else Normalize krdo Mean =0 SD =1 rheah

Why to do this?



Cost Function Contour

Har Hidden Layer Ke Output ko Normal Distribution mai transform krrhe hai , jishse Internal Covariate Shift kamm horha hai & stability millrhi

& zyaada Learning Rate Use karke hum Covergence pe Pochjaate hai

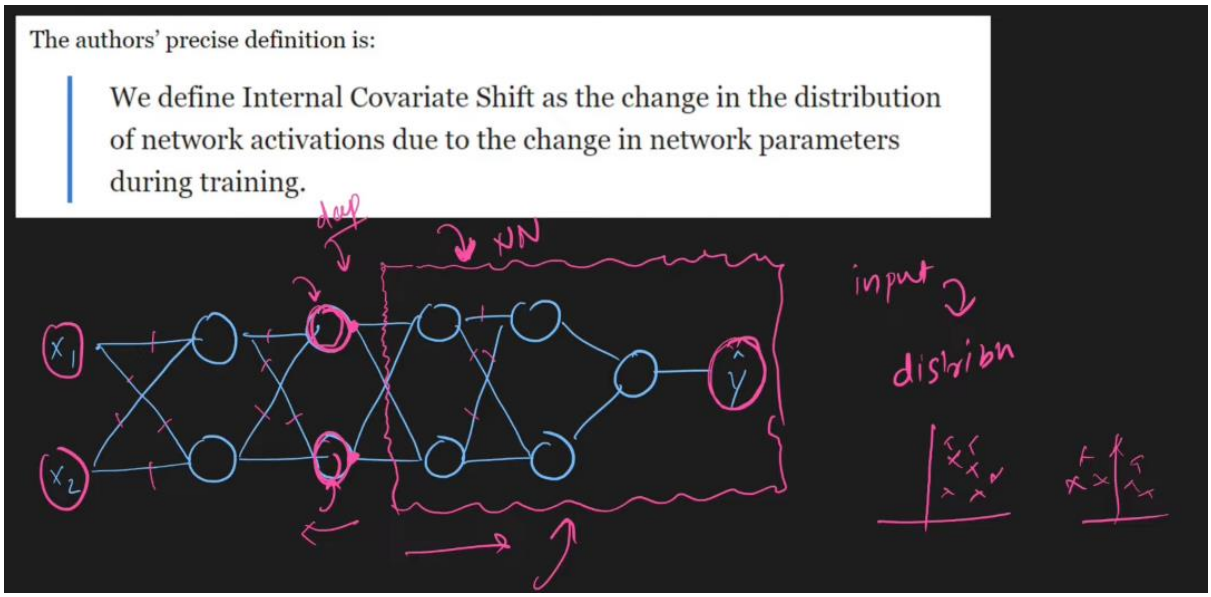
Agar batch normalization use nahi krrhe hai hum,

Then humme Low Learning Rate set karna padta hai

Covariate Shift

Internal Covariate Shift

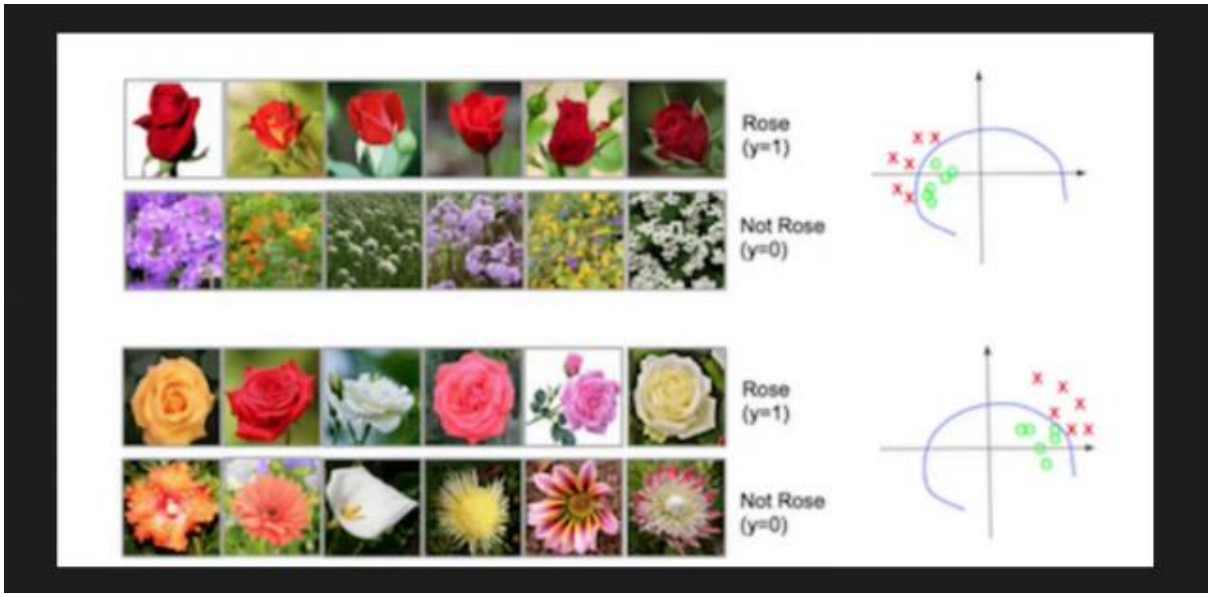
We define Internal Covarite Shift as the change in distribution of network activation due to change in network parameters during training.



https://youtu.be/uUPvvCHelD0?si=Z54zDc6yj-2RWuCe

Let's First understand Covariate Shift?

➔



The main idea behind Batch Normalization is to normalize the input of each layer in a neural network by subtracting the mean and dividing by the standard deviation of the mini-batch.

This helps address the internal covariate shift problem, which refers to the change in the distribution of the input to a neural network's layers during training.

By stabilizing and normalizing the inputs, Batch Normalization has been shown to accelerate training, allow for higher learning rates, and sometimes act as a form of regularization.

Batch Normalization is typically applied before the activation function in a neural network layer. It has been observed to have several benefits:

1. **Faster Convergence: BN reduces the internal covariate shift, which helps in faster convergence during training.**
2. **Higher Learning Rates: BN allows the use of higher learning rates without the risk of divergence or saturation.**
3. **Regularization: BN has a slight regularization effect due to the normalization process, reducing the need for other regularization techniques like dropout.**
4. **Reduced Sensitivity to Weight Initialization: BN makes the network less sensitive to the choice of initial weights.**

However, it's important to note that Batch Normalization may not always be beneficial in every situation, and its performance depends on the specific characteristics of the problem and the architecture of the neural network. Additionally, there are variations of Batch Normalization, such as Layer Normalization and Group Normalization, which are designed to address certain limitations of Batch Normalization in specific scenarios.

Summary

Batch Normalization Makes Training Very Stable

1. **Training becomes fast**
2. **Regularization effect becoz of randomness**
3. **Weight initialization ka Negative Impact Reduce krta hai**

& Batch Normalization Helps in Regularization it is also one of the advanatage

Keras Implementation