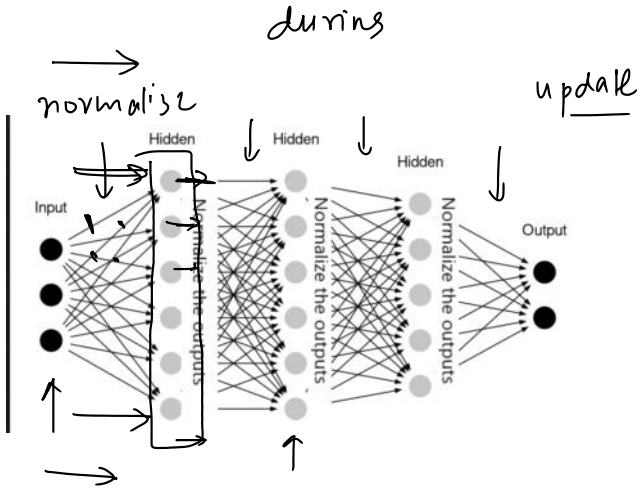
Batch Norm

03 January 2025 18:40



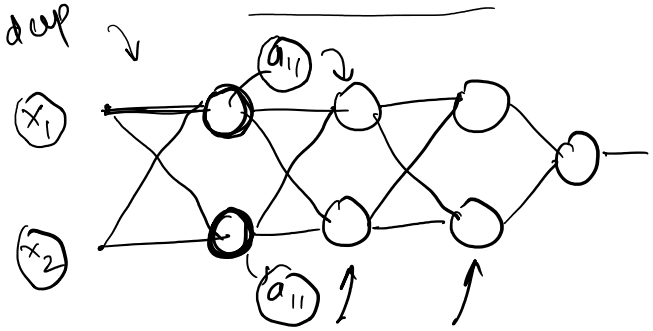
- Applied to Hidden Layers:
  - Typically applied to the hidden layers of a neural network, but not to the output layer.
- Applied After Linear Layers and Before Activation Functions:
  - Normalizes the output of the preceding layer (e.g., after nn.Linear) and is usually followed by an activation function (e.g., ReLU).
- Normalizes Activations:
  - Computes the mean and variance of the activations within a mini-batch and uses these statistics to normalize the activations.
- Includes Learnable Parameters:
  - Introduces two learnable parameters, gamma (scaling) and beta (shifting), which allow the network to adjust the normalized outputs.
- Improves Training Stability:
  - Reduces internal covariate shift, stabilizing the training process and allowing the use of higher learning rates.
- Regularization Effect:
  - Introduces some regularization because the statistics are computed over a mini-batch, adding noise to the training process.
- Consistent During Evaluation:
  - During evaluation, BatchNorm uses the running mean and variance accumulated during training, rather than recomputing them from the mini-batch.

What is Batch Norm?

27 June 2022 11:00

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

↝ It consists of normalizing activation vectors from hidden layers using the mean and variance of the current batch. This normalization step is applied right before (or right after) the nonlinear function.

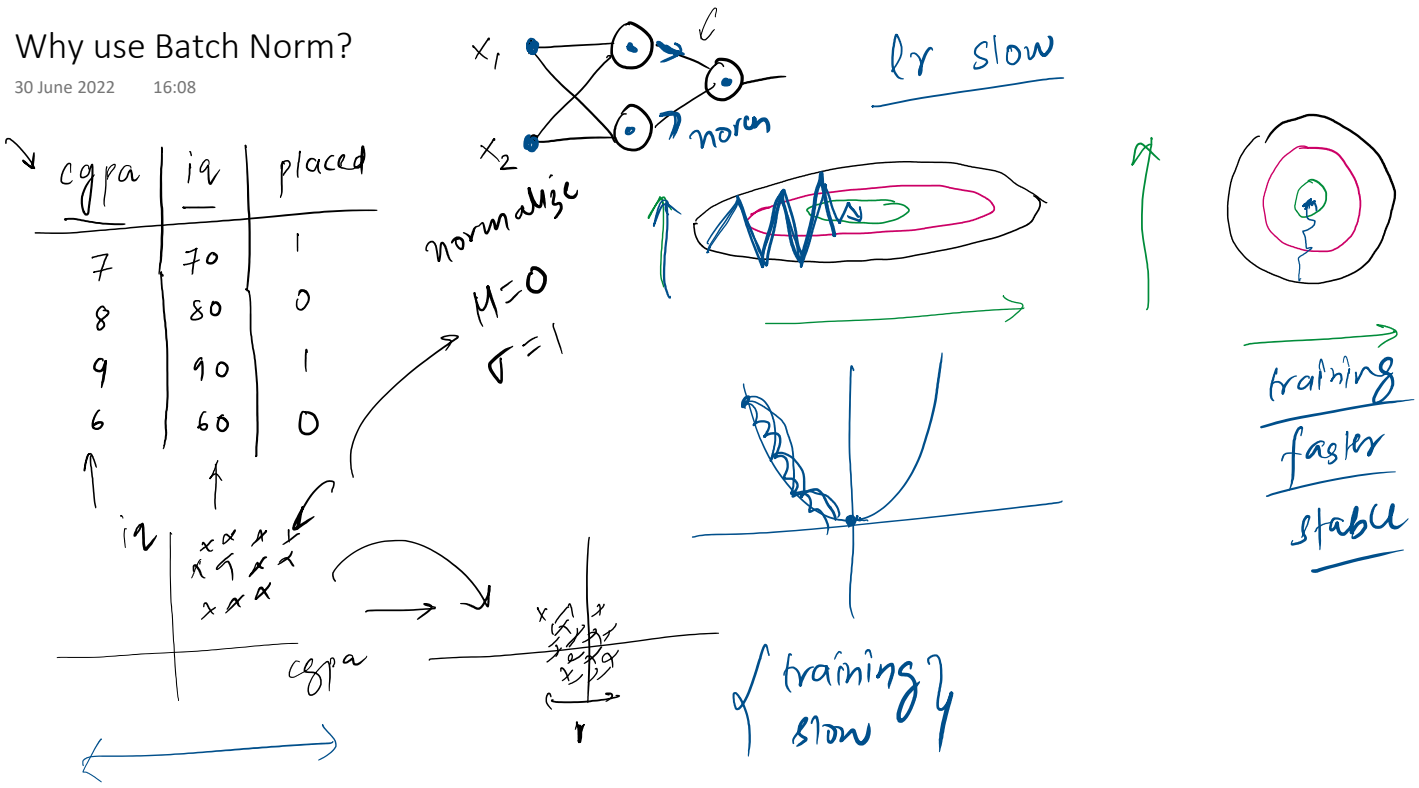


normalized
m=0
sd=1

normalize
m=0 sd=1

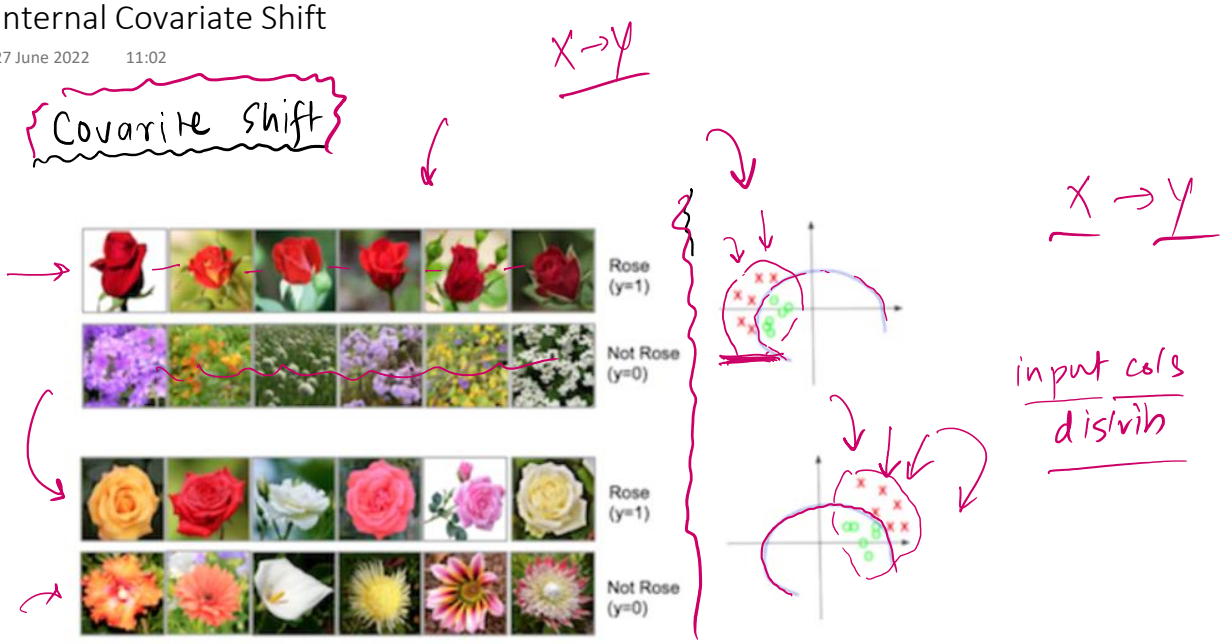
Why use Batch Norm?

30 June 2022 16:08



Internal Covariate Shift

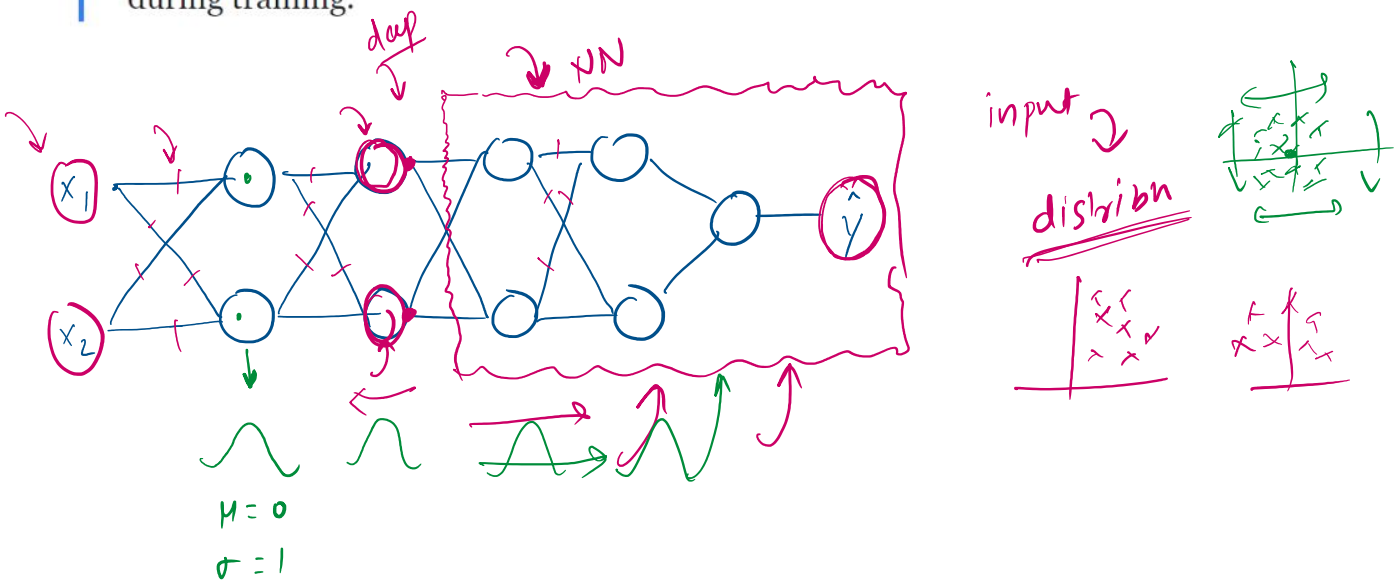
27 June 2022 11:02

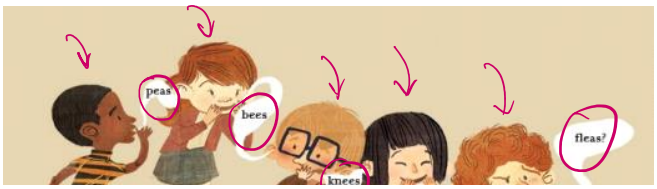


learning rate Xavier parameter init

The authors' precise definition is:

> We define Internal Covariate Shift as the change in the distribution of network activations due to the change in network parameters during training.

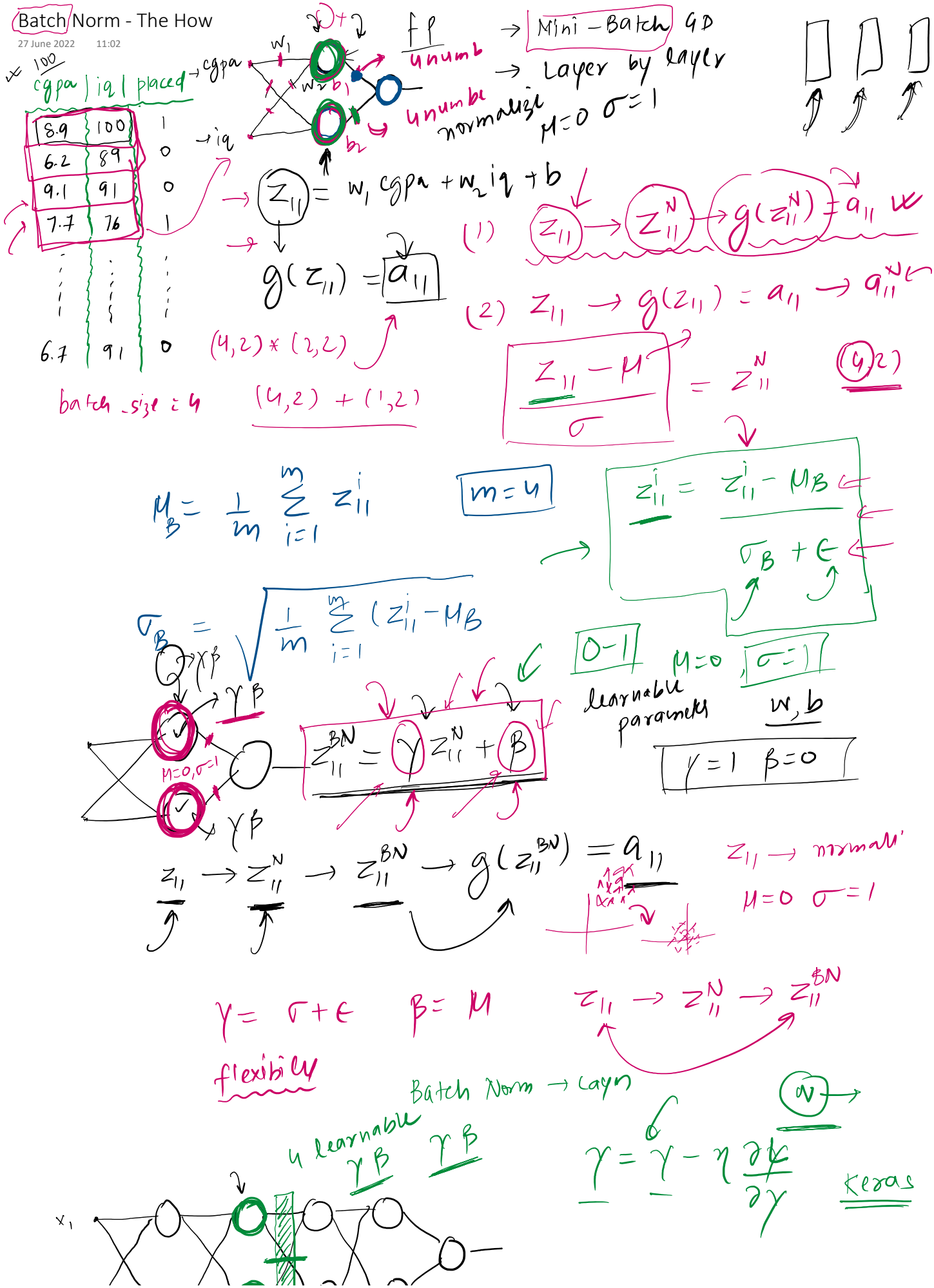


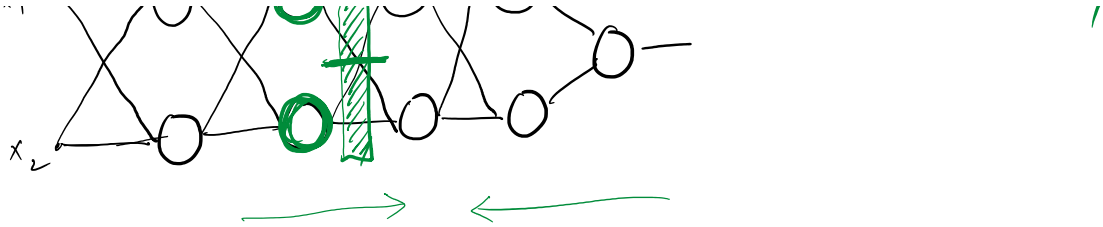



bees
knees
cheese
fleas?

Batch Norm - The How

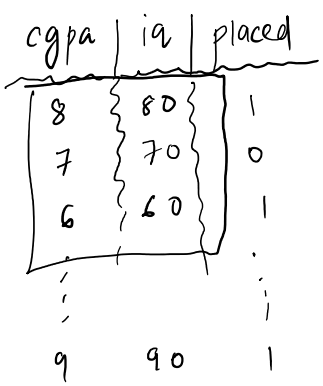
27 June 2022 11:02

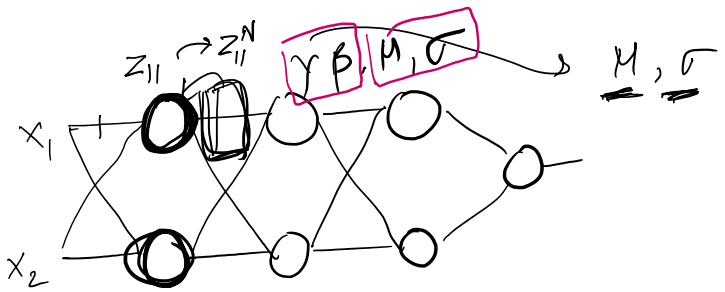




Batch Norm during test

27 June 2022 11:03





testing
query
point

Exp weighted moving avg.

EWMA

25 times

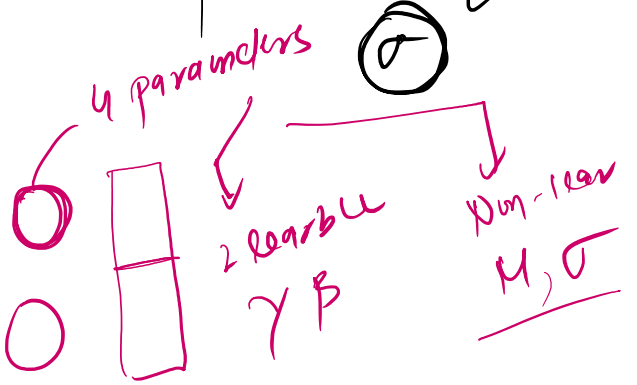
1 batch

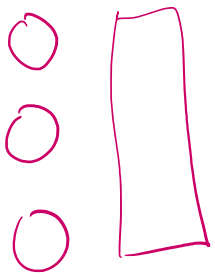
2

$\mu_{B1}, \mu_{B2} \ldots \mu_{B25}$

μ

EWMA





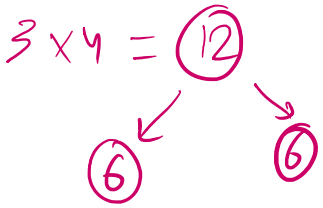
Why gamma and beta is there ?
Neural networks do not always want: zero mean and unit variance. Example: A layer may need activations centered around 5 Or need a larger/smaller variance to represent confidence.

If we force everything to mean 0 and variance 1:The network loses representational power
So normalization alone is too restrictive.

After normalization, we apply:
y = gamma * normalized + beta.
Where:
γ (gamma) → scale (controls variance)
β (beta) → shift (controls mean)
These are learnable parameters.

The network can now:
Keep normalized values as-is
OR scale them up/down
OR shift them left/right
OR completely undo normalization if needed.

If the best thing is no normalization at all:
γ → √variance
β → mean
The network can learn to cancel normalization.



Normalization standardizes → γ and β restore flexibility
Normalization standardizes activations; γ and β give the model back expressive power.

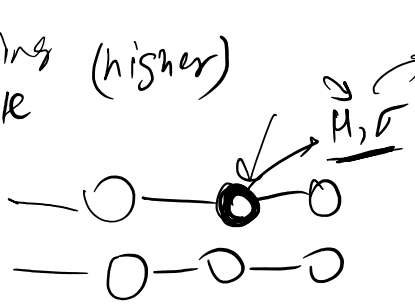
Advantages

27 June 2022 11:02

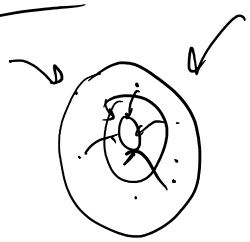
1) Stable → hyperpara → wider range of values

2) faster → learning rate (higher)

3) Regularizer → dropout



randomness ↓ noise → overfitting



4) Weight init impact reduce