



# SNS COLLEGE OF TECHNOLOGY

(An Autonomous Institution)

Coimbatore – 35.



## DEPARTMENT OF BIOMEDICAL ENGINEERING

### UNIT 2

#### BATCH NORMALIZATION

##### What is Batch Normalization?

Batch Normalization is a technique used to improve the training of deep [neural networks](#). Introduced by Sergey Ioffe and Christian Szegedy in 2015, batch normalization is used to normalize the inputs of each layer in such a way that they have a mean output activation of zero and a [standard deviation](#) of one. This normalization process helps to combat issues that deep neural networks face, such as internal covariate shift, which can slow down training and affect the network's ability to generalize from the training data.

##### Understanding Internal Covariate Shift

Internal covariate shift refers to the change in the distribution of network activations due to the update of weights during training. As deeper layers depend on the outputs of earlier layers, even small changes in the initial layers can amplify and lead to significant shifts in the distribution of inputs to deeper layers. This can result in the need for lower learning rates and careful parameter initialization, making the training process slow and less efficient.

##### How Batch Normalization Works

Batch normalization works by normalizing the output of a previous activation layer by subtracting the batch mean and dividing by the batch standard deviation. After this step, the result is then scaled and shifted by two learnable parameters, gamma and beta, which are unique to each layer. This process allows the model to maintain the mean activation close to 0 and the activation standard deviation close to 1.

The normalization step is as follows:

1. Calculate the mean and [variance](#) of the activations for each feature in a mini-batch.
2. Normalize the activations of each feature by subtracting the mini-batch mean and dividing by the mini-batch standard deviation.

3. Scale and shift the normalized values using the learnable parameters gamma and beta, which allow the network to undo the normalization if that is what the learned behavior requires.

Batch normalization is typically applied before the [activation function](#) in a network layer, although some variations may apply it after the activation function.

### **Benefits of Batch Normalization**

Batch normalization offers several benefits to the training process of deep neural networks:

- **Improved Optimization:** It allows the use of higher learning rates, speeding up the training process by reducing the careful tuning of parameters.
- **Regularization:** It adds a slight noise to the activations, similar to dropout. This can help to regularize the model and reduce overfitting.
- **Reduced Sensitivity to Initialization:** It makes the network less sensitive to the initial starting weights.
- **Allows Deeper Networks:** By reducing internal covariate shift, batch normalization allows for the training of deeper networks.

### **Batch Normalization During Inference**

While batch normalization is straightforward to apply during training, it requires special consideration during inference. Since the mini-batch mean and variance are not available during inference, the network uses the moving averages of these statistics that were computed during training. This ensures that the normalization is consistent and the network's learned behavior is maintained.

### **Challenges and Considerations**

Despite its benefits, batch normalization is not without challenges:

- **Dependency on Mini-Batch Size:**

The effectiveness of batch normalization can depend on the size of the mini-batch. Very small batch sizes can lead to inaccurate [estimates](#) of the mean and variance, which can destabilize the training process.

- **Computational Overhead:** Batch normalization introduces additional computations and parameters into the network, which can increase the complexity and computational cost.
- **Sequence Data:**

Applying batch normalization to [recurrent neural networks](#) and other architectures that handle sequence data can be less straightforward and may require alternative approaches.

## **Conclusion**

Batch normalization has become a widely adopted practice in training deep neural networks due to its ability to accelerate training and enable the construction of deeper architectures. By addressing the issue of internal covariate shift, batch normalization helps to stabilize the learning process and improve the performance of neural networks. As with any technique, it comes with its own set of challenges, but the overall benefits have solidified its place as a standard tool in the [deep learning](#) toolkit.

## **References**

Ioffe, S., & Szegedy, C. (2015). Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. arXiv:1502.03167.