Student ID : 20194293

Name : Go, Kyeong Ryeol

[AI 502] Batch Normalization : Accelerating Deep Network Training by Reducing Internal Covariate Shift

1. Paper Summary

When the distribution of each layer's input varies, training a deep neural network gets hard as it needs to continuously adapt to the new distribution. Therefore, the elaborate settings for learning rate and the parameter initialization are inevitable. This paper is dealing with this 'Internal Covariate Shift' problem and resolves it by the technique named as "Batch Normalization" that enables higher learning rates and nice regularization by its stochasticity.

To reduce internal covariate shift, several methods for whitening the activations are suggested. Some previous works have changed the parameters of the optimization algorithm to depend on the network activation values. However, the author argues that the gradient step may raise any change neither to the output of layers nor the loss while the model parameters are continuously updated to grow indefinitely.

This paper considered the way of performing the whitening by directly modifying the network architecture to involve the normalization step not only on a particular training example but on all examples. However, it is very computationally expensive as it requires computing the covariance matrix and its inverse square root for every example and also the Jacobians with respect to all the training examples to conduct the backpropagation.

Therefore, the author made the following two assumptions. First, each scalar feature is normalized independently rather than jointly. To maintain what the layer was supposed to represent, they introduced two additional parameters for each feature dimension so that it can play a role as an identity transform. Furthermore, the normalization is conducted in a mini-batch unit rather than using all the training examples. This can be easily rationalized as the sample mean and variance are the unbiased estimators.

The experiments are mainly focused on image classification tasks with public datasets like MNIST and ImageNet. It was empirically shown that batch normalization helps the network train faster, achieve higher accuracy and makes the distribution more stable and reduces the internal covariate shift. Also, it can be further improved by several modifications like follows; Increase learning rate / Remove dropout / Reduce the L2 weight regularization / Accelerate the learning rate decay / Remove the local response normalization / Shuffle training examples more thoroughly / Reduce the photometric distortions.

Before I move on to the discussion, there are two important additional notes to refer from this paper. First, while during the inference, they used the estimates from the population rather than from the mini-batch for the output to only depend on the input deterministically. Second, Batch Normalization is conducted immediately before the non-linear activation, which is more likely to have a symmetric, non-sparse and gaussian-like distribution.

2. Discussion

Here, I want to offer three discussion points. To begin with, instead of changing the batch size, how can we deliberately control the regularization power of the Batch Normalization? I suggest to use it along with the dropout or to compute the mean and variance among multiple features. Or the median may substitute the mean in the standard normal transformation. Furthermore, stratified sampling can be utilized when dividing the training examples to mini-batches. Next, how does it perform with other normalization method? Transforming the data distribution to standard normal is the most common way, but still there are many other methods like min-max scaling that are also widely used.