CS 7150: Deep Learning — Summer-Full 2020 — Paul Hand

Week 3 — Preparation Questions For Class
Due: Monday May 18, 2020 at 12:00 PM Eastern time via Gradescope

Name: [Sumeet Gajjar]
Collaborators: [None]

You may consult any and all resources in answering the questions. Your goal is to have answers that are ready to be shared with the class (or on a hypothetical job interview) as written.

Directions: Read the articles 'Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift' (original BN paper) and 'How Does Batch Normalization Help Optimization?' (Santurkar et al.) . Watch this video of Ian Goodfellow explaining batch normalization (3:46 - 13:16)

Question 1. *In the context of CNNs, what is Batch Normalization? In a BN layer, a variance is computed over a batch of images, but at test time there may be only a single image passed into the network. The variance of a quantity over only a single datapoint is undefined. How is this issue dealt with?*

Response:

Answer 1. *Batch normalization normalizes the input of the activations to achieve zero mean and unit variance. In context of CNN, the gamma and beta parameters learned during the training process is same for all the activations in a feature map. Thus the gamma and beta parameters are learned per feature map instead of per activation.*

During the test time or inference the input is normalized using the mean and variance of the population rather than using the mini-batch statistics.

Question 2. *What empirical benefits does batch normalization provide? Explain the experimental evidence for these benefits.*

Response:

Answer 2. *The empirical benefits of Batch normalization are as follows:*

- *Less number of training steps*

  In Figure 1.a, the authors plot the test accuracy vs the number of training steps. We can see that the Batch normalized network achieves a higher test accuracy with less number of training steps.

- *Higher learning rate*

  The authors mentioned in Section 4.2.1 that higher learning rate led to training speedup without any ill side-effects. In figure 2, plots of BN-x5 and BN-x30 networks indicates the same.

- *BN allows us to train networks with saturating non-linearities e.g. sigmoid*

  In section 4.2.2 the authors verify that the reduction in internal covariate shift allows deep networks with Batch Normalization to be trained when sigmoid is used as the non-linearity, despite the well-known difficulty of training such networks. In figure 2, its shown that BN-x5-Sigmoid achieves the accuracy of 69.8%. Without Batch Normalization, Inception with sigmoid never achieves better than 1/1000 accuracy.

- *Improves accuracy*

  In the Figure 2, the BN-x30 network achieves a better accuracy than Inception.

Using Batch normalization also shows the following benefits however their experimental evidence is not specified in the paper.

- *Allows us to be less careful about the initialization*
- *BN acts as a regularizer in some cases eliminating the need for dropout*

Question 3. *In the original BN paper, what evidence is provided that batch normalization works because it reduces internal covariate shift? Is this evidence convincing?*

Response:

Answer 3. *In section 4.1, the authors confirm the internal covariate shift reduces due to batch Normalization. The authors studied inputs to the sigmoid, in the original network N and batch normalized network over the course of training. In Fig. 1(b,c) the authors show, for one typical activation from the last hidden layer of each network, how its distribution evolves. The distributions in the original network change significantly over time, both in their mean and the variance, which complicates the training of the subsequent layers. In contrast, the distributions in the batch normalized network are much more stable as training progresses, which aids the training. And hence the authors conclude that the reduction in Internal covariate shift is the reason for Batch Normalization to work.*

However this is not very convincing since the graph contains a plot of a typical activation in the last hidden layer. The authors do not mention of observing similar effect in the rest of the layers. Using the plot for a single activation in the last hidden layer, we cannot generalize the reduction of Internal Covariate Shift for all activations in each of the Batch normalized hidden layers.

Question 4. *Explain the evidence that the Santurkar et al. paper uses to argue that BN's performance is not explained by reducing internal covariate shift.*

Response:

Answer 4. *Santurkar et al. in the figure 1.c shows that there is marginal change in the input distribution for layers 1 and 3 of unmodified VGG and VGG with BN.*

Further in the section 2.1 the authors propose a experiment to introduce noisy internal covariate shift(ICS) explicitly after the BN layer. Such noise injection produces a severe covariate shift that skews activations at every time. The figure 2 shows the finding that even after introducing noisy ICS the performance is better than the non BN network.

In section 2.2 the authors define ICS of activation i at time t as $\|G_{t,i} - G'_{t,i}\|_2$, where $G_{t,i}$ corresponds to the gradient of the layer parameters and $G'_{t,i}$ is the same gradient after all the previous layers have been updated with their new values. Here the authors show that the BN networks often show increase in ICS. Figure 3.b shows that the model with BN has worse ICS as compared to a model without BN despite performing well in terms of accuracy.

Question 5. *What is Santurkar et al.'s explanation of BN's performance. What experimental evidence is provided that batch normalization works because of this explanation? Is this evidence convincing?*

Response:

Answer 5. *According to Santurkar et al., BN makes the optimization landscape significantly smoother. This smoothness induces a more predictive and stable behavior of the gradients, allowing for faster training.*

This smoothness in landscape is achieved by reparametrizing the underlying optimization problem. This provides improvement in the Lipschitzness of the loss function. This reparametrization makes gradients of the loss more Lipschitz too, which implies that the gradients more reliable and predictive. In other words, the loss exhibits a significantly better "effective" β-smoothness

The authors provides the evidence for gradient predictiveness and "effective" β-smoothness in Figure 4. In fig 4.a the authors show how the loss changes as we move in the direction of gradients. In fig 4.c the authors demonstrate the effect of Batch Normalization on the stability/Lipschitzness of the gradients of the loss.

The authors also show the same performance benefit can be achieved if some other technique to fix the first order moments is used instead of Batch Normalization. One such technique was to normalizes using the average of l_p-norm (before shifting the mean), for $p = 1, 2$ and ∞.

The evidence seems convincing since the authors successfully challenges the conventional wisdom about Batch Normalization with experimental evidence and provides the appropriate results for their empirical findings of landscape smoothness and gradient predictiveness. The authors have successfully supported their empirical findings with plots and have provided all the necessary details to reproduce the results in the appendix section.

Question 6. *What theoretical results are made in the Santurkar et al. paper? Are these results convincing in explaining the behavior of batch normalization?*

Response:

Answer 6. *The authors makes the following theoretical result to explain the behavior of Batch Normalization:*

- *BN has an effect on the Lipschitzness of the loss, which makes the optimization landscape more predictable*
- *BN has an effect on the smoothness of the Optimization landscape.*
- *There is a minimax bound on weight-space Lipschitzness*
- *BN leads to favorable initialization.*

This results are convincing in explaining the behavior of BN. All the above results are stated as theorems in the paper and have a corresponding satisfactory proof. The experiments conducted by the authors seems to obey these mathematical properties stated by the theorem. Furthermore the details to reproduce the results is specified in the appendix section.

Question 7. *What explanation does Ian Goodfellow provide for batch normalization. What reasoning is provided for believing it?*

Response:

Answer 7. *According to Ian Goodfellow, Batch Normalization is not a optimization algorithm. Instead it's a way for designing the network which makes it easy to optimise. Every layer affects the statistics of activations in every layer that comes after it. Adding Batch Normalization enables us to decouple the mean and the standard deviation of activations of the succeeding layers. We have By having zero mean and unit variance fix.*

Ian provides an example of a linear neural network $a \rightarrow b \rightarrow c \rightarrow d \rightarrow e$. Here the value of a almost determines statistics of activations in d. The coupling is due to the fact that SGD is completely blind to interactions between multiple different variables.

Adding Batch normalization we are able to knock out some of the most important interactions between a layer and all the succeeding layers. In order to compute the statistics of e, we now only have to look at the γ and β parameters of that layer.

Question 8. *What is the point of having an explanation for why batch normalization works?*

Response:

Answer 8. *Understanding the roots of such a fundamental techniques as Batch Normalization will let us have a significantly better grasp of the underlying complexities of neural network training and, in turn, will inform further algorithmic progress in this context.*