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: **Saurabh Kiran Vaidya**
Collaborators: Sumeet Gajjar

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:

In context of CNN, Batch Normalization jointly normalizes all activations in a mini-batch over all locations. This is done to preserve the convolutional property so that different elements of the same feature map, at different locations, are normalized in the same way. And γ and β parameters are learned per each feature map instead of per activation.

In test phase, we calculte fixed values of mean and variance computed on the entire training set. To compute mean and variance , we use E[x] over all x in train set and Var[x] = m/m-1 * $E_B[Var_B]$ where m is batch size and Var_B is sample variance for each batch. We replace the output of BN(x) by computing $\widehat{x}$ with these fixed values and scaling it with γ and translating with β which were learned during training.

Thus for a single datapoint, the variances are calculated by weighted Expectation of variances of mini batches during training phase.

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

Response: Batch Normalization provides benefits such as fewer steps of learning meaning faster convergence, allows using higher learning rate and using saturating non-linearities as well without problems of vanishing or exploding gradients. Additionally, due to BatchNorm, we also get better accuracy and in some cases no or reduced dropout and regularization.

To provide evidence for higher learning rates, quicker convergence and better performance, the authors apply Batch Normalization(BN) layers to Inception network used in ImageNet. They then make certain additional tweaks to their BN model(BNx-5: 5 times initial leraning rate, reduced dropout) and then compare the standard Inception, BN-Baseline and BNx-5. As seen in

Fig 2 in the paper, BNx-5 needed 14 times fewer steps in training than the original Inception model to reach its benchmark performance. Further, a tweaked BN model with 30 times learning rate, trains quicker and achieves higher performance of 74.8 % over 72.2 % of standard Inception model.

To show how Batch Norm can allow using saturating non-linearities, authors train a Batch Norm applied network of Inception with sigmoid activation and show that it achieves 69% accuract verses using sigmoid on standard Inception which is never better than 1/1000.

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: To provide evidenec that Batch Normalization works because it reduces internal covariate shift, authors train a 3 hidden layer network on MNIST dataset with and without Batch normalizaion. Then they take one activation from some layer and show the distribution of inputs to that unit's sigmoid in both networks. As per the distribution plot seen in fig 1.b and 1.c its seen that the distribution is more stable in network with Batch Norm.

This evidence however does not seem convincing to me as maybe instead of a single unit, they could have shown how much the distribution of inputs changes over the the entire layer and for a deeper network.

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: The authors train three different networks: standard, Batch Normalization network and Batch normalization network with added noise. A random non-zero mean and non-unit variance noise is added to the output of batch normalization layer. This noise distribution changes at each time step and thus this network has sever covariate shift in its input at each time step.

This noisy BN network still performs better than a standard non-BN network. Moreover, the performance difference between noisy and regular BN networks is also non-existent. Thus authors provide a strong evidence that inspite of increased covariate shift, Batch Normalization had the same benefits. Thus BN's performance was not explained by reducing covariate shift.

Further, authors show that for a particular layer, the evolution of distribution of input to that layer does not change significantly in standard network versus a batch normalization network as seen in fig 1.c.

Moreover, authors also present internal covariate shift(ICS) can be looked as difference in gradient of loss of each layers before and after updates to all previous layers. The show that model trained with BN shows an increase in this ICS. Thus from an optimization point of view BN does not reduce ICS.

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: The authors explain that Batch Norm makes the optimization landscape significantly smoother. This smoothness induces a more predictive and stable behavior of the gradients, allowing for faster training. This smoothness is achieved because authors say that BN improves

the Lipschitzness of the loss function. Meaning, the loss function changes at a smaller rate and thus the gradients also change at a smaller rate. This improve the predictiveness of the gradients enabling faster learning.

To demonstrate the stability of loss function, they compute gradient of loss at a given step in training and measure how the loss would change as we move in the direction of gradient. As seen in fig 4.a, the loss function of vanilla network has long range of values compared to BN network. Moreover, fig 4.b shows that there is a significant difference in the predictiveness of the gradient between BN and vanilla networks. The l_2 distance between gradient value of loss at a given point in training with all the gradient values of loss at different points along the gradient direction is smaller than vanilla network.

The evidence seems convincing enough since if loss landscape is smoother, there aren't many 'kinks', flat regions and sharp minima. Thus there will be a reduction in the Lipschitz constant of gradients and they wont change their values in unpredictable way or even vanish or explode. Since the landscape is smooth the magnitude of the gradient can be a good prediction of much we expect the loss to fall in that direction. This belief can then allow us to take bigger steps in the direction of that gradient as there would not be plateau regions, sharp drops and flatter valleys.

This is solidified when authors show that techniques other than batch normalization that have the same effect of loss landscape smoothness also provide the same benefit of faster convergence as BN.

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: The paper presents theoretical results on Batch Norm showing a better Lipschitz constant for loss, having a minimax bound on the weight-space Lipschitzness, and more predictiveness of gradients. They also show that BN leads to favorable initialization. BN not just scales the loss landscape but also preserves the local minimas of the original landscape

These results are convincing because they provide mathematical theorems and their experimental observations seem to comply with properties of those theorems

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

Response: Ian Goodfellow explains that Batch normalization is not an optimization algorithm itself but it helps in designing better and easy optimizations for neural networks. Batch Normalization leads us to have standard values of low order moments for a layer which is dependent of weights from all of the previous layers.

The reason to believe this is without batch normalization SGD is oblivious to the dependency of a layer's output on all its previous layers. He states that normalization removes this dependency of a layer's output on all of its subsequent layers. Thus SGD only looks at the γ and β parameters to fix the lower order statistics of a deeper layer.

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

Response: Having an explanation for why batch normalization works will help us in understanding the complexities of neural network training. Knowing how batch normalization achieved

faster convergence can help us get clarity on implicit and hidden interactions that take place between layers and their dependency on each others outputs. It can also help us to design better algorithms.