Submission for Deep Learning Exercise

Team: shallow_learning_group
Students: Batuhan Karaca

November 28, 2023

Pen and Paper tasks

1) Data Augmentation

1.

Translation, scaling, rotation, shearing and reflection for symmetric numbers (i.e. 0, 8 and some of the 1 instances where the articulating tip is insignificant).

2.

Reflection for asymmetric numbers will not work as their reflected instances will not correspond to an instance anymore. Furthermore, the color operations (i.e. inverting, color-mapping) is unnecessary due to the fact that people generally tend to normalize the color values (to simplify the task as well as avoiding exploding values in the network) as a preprocessing step.

2) Early Stopping

1.

According to the book, early stopping occurs when the validation loss is at minimum theoretically (which is less feasible with noisy curves in stochastic optimizers).

2.

Overfitting occurs when the training loss still decreases (fitting to the noise and details of the training data), but test/validation loss starts to increase, and vice-versa for accuracy (the scores start to diverge). In other words, the model becomes so complex that it cannot adapt to new data anymore. The regularization techniques try to avoid this complexity of the model. For example, weight regularization tries to create sparser networks to reduce complexity. In a similar fashion, early stopping avoids more complex networks by stopping before the overfitting occurs.

Another approach is given in the book. It is shown that given certain conditions (i.e. number of steps τ, learning rate ϵ, regularization parameter α), the early stopping is equivalent to L_2 regularization. Furthermore, proper initialization is also important as the proof is given for the values in the neighborhood of the optimal value.

3) Discussion on Weight Distributions Before and After Various Regularization Techniques

As I pointed out in the code file, I tried both normalizing and limiting the values between -1 and 1. However, the information I got was less doing either of them. Therefore I decided not doing any of them.

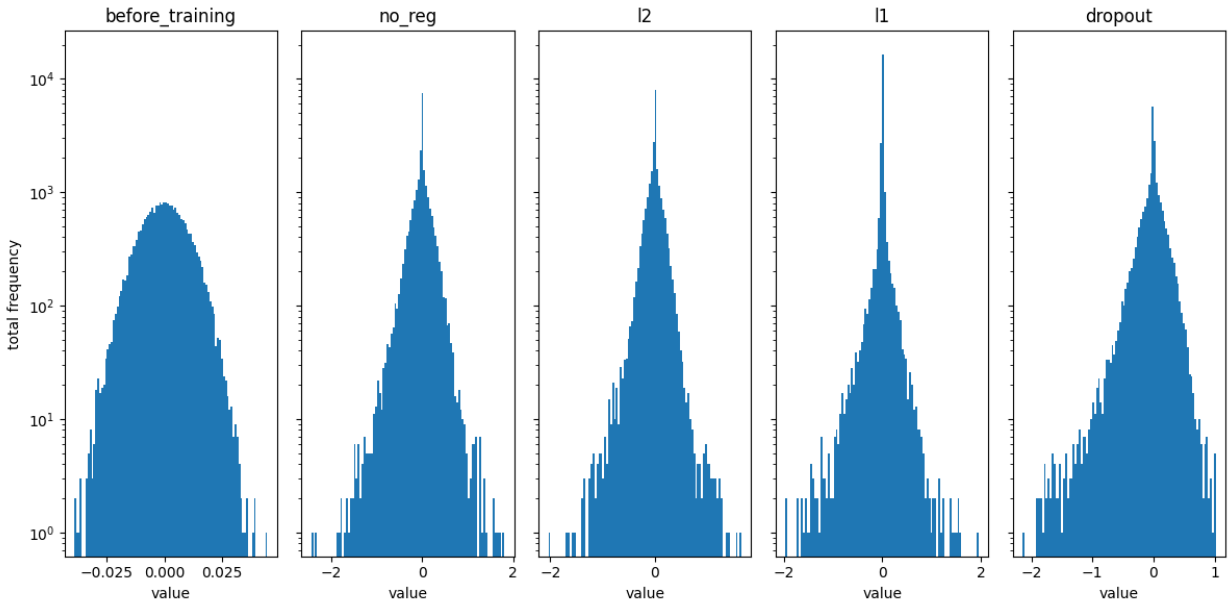

Figure 1: Plots of weight distributions before and after various regularization techniques

As figure 1 shows, due to `numpy.random.randn`, the weights had standard normal distribution before training. We observe that after training, the distributions had wider range. This could be due to the fact that as more layers are used, the weights in the initial layers tend to have larger gradients in the backpropagation step. This problem becomes more significant with deeper networks, also known as the exploding gradient problem.

Compared to before training, after training, all networks became sparser. It is true that regularization creates sparser networks. We also observe that no use of regularization acts like a L_2 regularizer. In the previous section, we discussed that early stopping is equivalent to L_2 regularization. We also discussed that early stopping is done before validation/evaluation score diverges from training score (overfitting). Looking at Table 1, we see that this is *roughly* (due to the noise in data and stochastic gradient descent) the case. The noise results in small fluctuations in the scores, which can be omitted to an extent. The experiment is done until the scores have not diverged yet, or about to diverge.

We also observe that L_1 regularization has created a sparser network compared to the previous two (or just L_2 regularization since they both act similar). For each weight w, the derivative of L_1 norm ($\alpha|w|$) is $\alpha sgn(w)$ whereas derivative of L_2 norm ($\frac{\alpha}{2}w^2$) is αw. For $|w| < 1$, we observe that magnitude of derivative of L_2 is smaller than that of L_1 (Note that the weights are mostly very small in all the graphs, i.e. $|w| << 1$). As w apporaches 0, the gap of derivatives widen, and L_2 regularization has less effect than before. Therefore, in the same amount of steps (i.e. 20 in the experiment), L_1 will have more weights closer to 0, creating that pointier graph.

The asymmetry of the graph of dropout regularization could be due to asymmetry in sampling. In other words, some inputs became activated more frequently than the others that this reflected in their corresponding weights.

Number of epochs	Training accuracy	Evaluation accuracy
1	0.8659	0.9468
2	0.9397	0.9529
3	0.9532	0.9575
4	0.9595	0.9551
5	0.9636	0.9585
6	0.9671	0.9622
7	0.9710	0.9537
8	0.9721	0.9621
9	0.9753	0.9606
10	0.9771	0.9674
11	0.9778	0.9638
12	0.9778	0.9652
13	0.9803	0.9634
14	0.9811	0.9614
15	0.9815	0.9620
16	0.9845	0.9659
17	0.9844	0.9672
18	0.9844	0.9664
19	0.9863	0.9663
20	0.9850	0.9609

Table 1: Accuracies for each epoch when no regularization is used