DS-GA 1008: Deep Learning, Spring 2019
Homework Assignment 3

Lekha Iyengar

March 2019

1.1 - Dropout

a. List the torch.nn module corresponding to 2D dropout.

Dropout2d is the torch.nn module corresponding to dropout. Its parameters are the dropout probability 'p' and 'inplace'. p is the probability of an element to be zero-ed. If inplace is set to True, the dropout operation is done inplace.

b. Read on what dropout is and give a short explanation on what it does and why it is useful.

Dropout is a regularization technique which is used to prevent overfitting. The technique involves temporarily dropping nodes (hidden and input) along with all its incoming and outgoing connections. A node at training time is present with probability p (dropped with probability 1-p). For the input nodes, however, the optimal probability of retention is usually closer to 1 At test time, the node is always present and the weights are multiplied by p.
During training, dropout uses several reduced networks. This prevents the nodes from co-adapting too much. This model combination also improves the performance of the network. "Although dropout alone gives significant improvements, using dropout along with max-norm regularization, large decaying learning rates and high momentum provides a significant boost over just using dropout. A possible justification is that constraining weight vectors to lie inside a ball of fixed radius makes it possible to use a huge learning rate without the possibility of weights blowing up. As the learning rate decays, the optimization takes shorter steps, thereby doing less exploration and eventually settles into a minimum." [1]

1.2 - Batch Norm

a. What does mini-batch refer to in the context of deep learning?

Usually the total amount of training data available is quite large. It is computationally expensive to use the entire batch at each step to train the algorithm. We prefer to use a small number of training instances which constitute a mini-batch in one iteration. Training over a mini-batch constitutes an iterator and training over all the mini batches constitutes an epoch.

b. Read on what batch norm is and give a short explanation on what it does and why it is useful.

The distribution of each layer's inputs changes during training, as the parameters of the previous layers change. To avoid this problem we need to use smaller learning rates and carefully initialize parameters. This slows down the training. Networks converge faster if the inputs have been whitened (zero mean, unit variances) and are uncorrelated [2] [3]. Batch normalization reduces the dependence of gradients on the scale of the parameters or on their initial values (reduces internal covariate shift) by fixing the means and variances of layer inputs. This allows us to use much higher learning rates without the risk of divergence. It also allows each layer to learn independently of other layers.It also acts as a regularizer, in some cases eliminating the need for Dropout. It normalizes the output of previous activation layer by subtracting the batch mean and dividing by the batch standard deviation. The gradient descent algorithm does the denormalization if required (to minimize loss function) by changing only these two weights for each activation, instead of losing the stability of the network by changing all the weights.

Links

- Read only link to Overleaf Project

References

[1] Nitish Srivastava, Geoffrey E. Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal oDropout: A Simple Way to Prevent Neural Networks from Overfittingf Machine Learning Research*, 15(1):1929–1958, 2014.

[2] Yann LeCun, Léon Bottou, Genevieve B. Orr, and Klaus-Robert Müller. Efficient backprop. In *Neural Networks: Tricks of the Trade, This Book is an Outgrowth of a 1996 NIPS Workshop*, pages 9–50, London, UK, UK, 1998. Springer-Verlag.

[3] Simon Wiesler and Hermann Ney. A convergence analysis of log-linear training. In J. Shawe-Taylor, R. S. Zemel, P. L. Bartlett, F. Pereira, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 24*, pages 657–665. Curran Associates, Inc., 2011.