Brain Optimal Damage

November 20, 2022

1

The idea of pruning network weights after training, was first introduced by LeCun, et al. at paper **Optimal Brain Damage**. In this paper they suggest Taylor series approximation of degree 2 of the loss function to be used as a metric to find importance (saliency) of each weight in the network. And then they suggest to prune the weights with the lowest importance. Pruning in this context is done by setting the weights to zero and freezing them.

1.1

So first, we find the effect of perturbing each weight on the loss function. Note that by setting a parameter to zero, the perturbation amount of that parameter is equal to the negative of value of that parameter itself.

Taylor series approximation of the loss function is given by:

$$\delta E = \sum_i g_i \delta u_i + \frac{1}{2} \sum_{i,j} h_{ij} \delta u_i \delta u_j + \mathcal{O}(\|\delta U\|^3) \tag{1}$$

Where u_is are weights and U is the vector of all weights. g_i is the gradient of the loss function with respect to the ith weight, h_{ij} is the Hessian of the loss function with respect to the ith and jth weights, and δU is the vector of perturbations of the weights:

$$g_i = \frac{\partial E}{\partial u_i} \tag{2}$$

$$h_{ij} = \frac{\partial^2 E}{\partial u_i \partial u_j} \tag{3}$$

So by considering $\delta u_i = -u_i$ to make ith weight zero, we can find weights that have the least effect on the loss function and can set them to zero.

But more practical way of using Eq. 1 will be introduced in the next subsection.

1.2

As you know, computing the Hessian of the loss function is computationally expensive and of a $O(n^2)$ complexity. So the paper uses another simplification, by assuming the Hessian to be diagonal. And as mentioned, pruning is done after training the network, so it will be on a local minima of the loss function and the gradients (g_is) will be zero. And Eq. 1 will be simplified to:

$$\delta E = \frac{1}{2} \sum_i h_{ii} \delta u_i^2 \tag{4}$$

So saliency of weight u_i is given by:

$$s_i = \frac{1}{2} h_{ii} u_i^2 \tag{5}$$

and we can prune the weight(s) with the lowest saliency.

So we can summarize the whole process as follows:

1. Train the network until reaching a good solution (local minima of the loss function).
2. Compute saliency values for each weight.
3. Prune some of the weights with the lowest saliency.
4. Go to step 1.