Machine Learning for Many-Body Physics
Homework 1

Thomas Koutsikos

April 2024

1 Backpropagation

In this problem, you will derive the equations needed to calculate the gradient of the cost function in a feed forward neural network using backpropagation. This gradient is used within learning algorithms (such as gradient descent) that train the neural network.

Recall that the output from the j$^{\text{th}}$ neuron in layer ℓ is given by

$$a_j^{(\ell)} = g_\ell\left(z_j^{(\ell)}\right), \tag{1}$$

where g_ℓ is a non-linear activation function and

$$z_j^{(\ell)} = \sum_{i=1}^{n_{\ell-1}} a_i^{(\ell-1)} W_{ij}^{(\ell)} + b_j^{(\ell)}, \tag{2}$$

with $W_{ij}^{(\ell)}$ representing the weights and $b_j^{(\ell)}$ the biases of the network. Assume that the cost function C can be expressed as a sum over the dataset $\mathcal{D}$ as

$$C = \frac{1}{|\mathcal{D}|} \sum_{\mathbf{x}\in\mathcal{D}} c(\mathbf{x}), \tag{3}$$

and define the notation

$$\delta_j^{(\ell)} = \frac{\partial c}{\partial z_j^{(\ell)}}. \tag{4}$$

a) Show that the quantity $\delta_j^{(\ell)}$ can be expressed as

$$\delta_j^{(L)} = \frac{\partial c}{\partial a_j^{(L)}} g_L'\left(z_j^{(L)}\right), \tag{5}$$

$$\delta_j^{(\ell)} = \sum_{k=1}^{n_{\ell+1}} \delta_k^{(\ell+1)} W_{kj}^{(\ell+1)T} g_\ell'\left(z_j^{(\ell)}\right) \quad (\text{for } \ell < L). \tag{6}$$

We begin with

$$\delta_j^{(L)} = \frac{\partial c}{\partial z_j^{(L)}} = \sum_{k=1}^{n_L} \frac{\partial c}{\partial a_k^{(L)}} \frac{\partial a_k^{(L)}}{\partial z_j^{(L)}} = \sum_{k=1}^{n_L} \frac{\partial c}{\partial a_k^{(L)}} \frac{\partial}{\partial z_j^{(L)}} g_L(z_k^{(L)}) = \frac{\partial c}{\partial a_j^{(L)}} g_L' \left(z_j^{(L)} \right) \tag{7}$$

For the second one we have

$$\delta_j^{(\ell)} = \frac{\partial c}{\partial z_j^{(\ell)}} = \sum_{k=1}^{n_{\ell+1}} \frac{\partial c}{\partial z_k^{(\ell+1)}} \frac{\partial z_k^{(\ell+1)}}{\partial z_j^{(\ell)}} \quad (\text{for } \ell < L) \Leftrightarrow \tag{8}$$

$$\delta_j^{(\ell)} = \sum_{k=1}^{n_{\ell+1}} \delta_k^{(\ell+1)} \frac{\partial}{\partial z_j^{(\ell)}} \left(\sum_{i=1}^{n_\ell} a_i^{(\ell)} W_{ik}^{(\ell+1)} + b_k^{(\ell+1)} \right) \Leftrightarrow \tag{9}$$

$$\delta_j^{(\ell)} = \sum_{k=1}^{n_{\ell+1}} \delta_k^{(\ell+1)} \frac{\partial}{\partial z_j^{(\ell)}} \left(\sum_{i=1}^{n_\ell} g_\ell \left(z_i^{(\ell)} \right) W_{ik}^{(\ell+1)} \right) \Leftrightarrow \tag{10}$$

$$\delta_j^{(\ell)} = \sum_{k=1}^{n_{\ell+1}} \delta_k^{(\ell+1)} g_\ell' \left(z_j^{(\ell)} \right) W_{jk}^{(\ell+1)} = \sum_{k=1}^{n_{\ell+1}} \delta_k^{(\ell+1)} W_{kj}^{(\ell+1)T} g_\ell' \left(z_j^{(\ell)} \right). \tag{11}$$

1b) Show that the partial derivatives of the cost function with respect to the network's weights and biases can be calculated as

$$\frac{\partial c}{\partial W_{ij}^{(\ell)}} = a_i^{(\ell-1)} \delta_j^{(\ell)}, \tag{12}$$

$$\frac{\partial c}{\partial b_j^{(\ell)}} = \delta_j^{(\ell)}. \tag{13}$$

For the first one we have

$$\frac{\partial c}{\partial W_{ij}^{(\ell)}} = \sum_{k=1}^{n_\ell} \frac{\partial c}{\partial z_k^{(\ell)}} \frac{\partial z_k^{(\ell)}}{\partial W_{ij}^{(\ell)}} = \sum_{k=1}^{n_\ell} \delta_k^{(\ell)} \frac{\partial}{\partial W_{ij}^{(\ell)}} \left(\sum_{m=1}^{n_{\ell-1}} a_m^{(\ell-1)} W_{mk}^{(\ell)} + b_k^{(\ell)} \right) = a_i^{(\ell-1)} \delta_j^{(\ell)} \tag{14}$$

For the second one we have

$$\frac{\partial c}{\partial b_j^{(\ell)}} = \sum_{k=1}^{n_\ell} \frac{\partial c}{\partial z_k^{(\ell)}} \frac{\partial z_k^{(\ell)}}{\partial b_j^{(\ell)}} = \sum_{k=1}^{n_\ell} \delta_k^{(\ell)} \frac{\partial}{\partial b_j^{(\ell)}} \left(\sum_{m=1}^{n_{\ell-1}} a_m^{(\ell-1)} W_{mk}^{(\ell)} + b_k^{(\ell)} \right) = \delta_j^{(\ell)} \tag{15}$$