1 Preamble

The neural network (function approximator) is just a chain of geometric transformations (functions) each parametrized by $W \in \mathcal{R}^{n_x \times n_h}$ on $X \in \mathcal{R}^{m \times n_x}$. Note that m is the number of examples in the dataset, n_x is the input size (i.e., output of the previous layer), and n_h is number of hidden units in the current layer. Also θ refers to the weights W and biases b in the network. Learning occurs by updating the parameters iteratively via mini-batch gradient descent.

2 Equations

Here I define the operations that occur for a single hidden layer fully connected neural network with the activation function of the last layer as the sigmoid function $\sigma(\cdot)$ during training.

2.1 Single Hidden Layer Neural Network

$$\begin{aligned} NeuralNet_\theta(X) &= \sigma((XW^{[1]} + b^{[1]})W^{[2]} + b^{[2]}) \\ &= f(g(w)) \end{aligned} \tag{1}$$

where f, g, and w are defined as follows:

$$\begin{aligned} g_{\theta^{[2]}}(A) &= AW^{[2]} + b^{[2]} \\ w_{\theta^{[1]}}(A) &= AW^{[1]} + b^{[1]} \\ f(t) = \sigma(t) &= \frac{1}{1 + e^{-t}} \end{aligned} \tag{2}$$

2.2 Neural Network Prediction (Forward Pass)

$$\hat{y} \leftarrow NeuralNet_\theta(X) \tag{3}$$

2.3 Mean Squared Error Loss Function

$$\mathcal{L}_\theta(X) = \text{MSE}_\theta(X) = \frac{1}{N} \sum_{i=1}^{N} (\hat{y}^{(i)} - y^{(i)}) \tag{4}$$

2.4 Gradient Update

$$\theta_i \leftarrow \theta_i - \eta(\nabla_\theta \mathcal{L}_\theta(\hat{y}, y)) \tag{5}$$