Neural network with backpropagation

December 19, 2021

Forward propagation

We consider a network consisting of L layers: one input layer, one output layer, and L-2 hidden layers. A given layer j of the network with s_j units can be represented in terms of two vectors, $\mathbf{z}^{(j)}$ and $\mathbf{a}^{(j)}$, whereas the propagation of data from the input to the output layers is given in terms of L-1 weight matrices $\mathbf{\Theta}$.

For a given input instance with feature vector $\mathbf{x}$, for the first layer ($j = 1$) the vector $\mathbf{a}$ is set as

$$\mathbf{a}^{(1)} = (1, \mathbf{x})^{\mathrm{T}}. \tag{1}$$

For $L > j > 1$, $\mathbf{z}^{(j)}$ and $\mathbf{a}^{(j)}$ can be obtained following

$$\mathbf{z}^{(j)} = \mathbf{\Theta}^{(j-1)} a^{(j-1)} \tag{2}$$

with

$$\mathbf{a}^{(j)} = \Big(1, g(\mathbf{z}^{(j)})\Big)^{\mathrm{T}} = \Big(1, g(z_1^{(j)}), g(z_2^{(j)}), ..., g(z_{s_j}^{(j)})\Big)^{\mathrm{T}}. \tag{3}$$

The activation function g is given by the sigmoid function

$$g(z) = \frac{e^z}{1 + e^z}. \tag{4}$$

The last layer does not have a bias unit, such that the output layer is given by

$$\mathbf{a}^{(L)} = g(\mathbf{z}^{(L)}) \equiv \hat{\mathbf{y}}. \tag{5}$$

Cost function

For m instances in the training set, the regularized cost function of the network is defined as

$$J(\{\boldsymbol{\Theta_i}\}) = -\frac{1}{m}\sum_m \mathbf{y}_m \cdot \log(\hat{\mathbf{y}_m}) - (1-\mathbf{y}_m)\cdot\log(1-\hat{\mathbf{y}}_m) + \frac{\alpha}{2m}\sum_j \sum_{i=2} \sum_{l=1} \left(\Theta_{il}^{(j)}\right)^2 \tag{6}$$

Backpropagation

Given a training instance $\mathbf{y}$, the error of the last layer ($j = L$) is set to

$$\boldsymbol{\delta}^{(L)} = \hat{\mathbf{y}} - \mathbf{y}. \tag{7}$$

For $L > j > 1$, the error associated to each layer is given by

$$\boldsymbol{\delta}^{(j)} = \left(\tilde{\boldsymbol{\Theta}}^{(j)}\right)^{\mathrm{T}} \boldsymbol{\delta}^{(j+1)} \circ g'(\mathbf{z}^{(j)}), \tag{8}$$

where $\circ$ denotes element-wise multiplication and $\tilde{\boldsymbol{\Theta}}^{(j)}$ corresponds to the weight matrix j without the first column. This is needed to exclude the bias unit from the backpropagation, which is not connected to the input unit.

The error associated with the weight matrix j is given by a matrix $\mathbf{D}^{(j)}$, which is defined as

$$D_{il}^{(j)} = \frac{\partial J}{\partial \Theta_{il}^{(j)}}. \tag{9}$$

Using the error of each individual layer, the error associated with the weight matrix j can be defined as

$$D_{il}^{(j)} = \begin{cases} \frac{1}{m}\Delta_{il}^{(j)} & \text{if } i = 0 \\ \frac{1}{m}\Delta_{il}^{(j)} + \frac{\alpha}{m}\Theta_{il}^{(j)} & \text{else.} \end{cases} \tag{10}$$

where the matrix $\boldsymbol{\Delta}^{(j)}$ accumulates the errors when going through the training set with m instances

$$\boldsymbol{\Delta}^{(j)} = \boldsymbol{\Delta}^{(j)} + \boldsymbol{\delta}^{(j+1)} \left(\mathbf{a}^{(j)}\right)^{\mathrm{T}}. \tag{11}$$