ANN Tutorial Examples, Q.J. Zhang

Exercise 1

For the MLP network with 1 input neuron, 2 hidden neurons with sigmoid functions, and 1 output neuron with linear function as shown in Figure A1, train the neural network using sample-by-sample backpropagation with learning rate λ = 0.1 and no momentum. Specifically, using the initial guess of weights in Figure A1, calculate the new weight values for 1 epoch. In the calculation, assume no scaling to x or y, and the training error is defined as

$$E(\mathbf{w}) = \frac{1}{2}\left(y(x, \mathbf{w}) - \hat{y}\right)^2$$

per sample and where $\hat{y}$ represents the training data corresponding to output neuron.

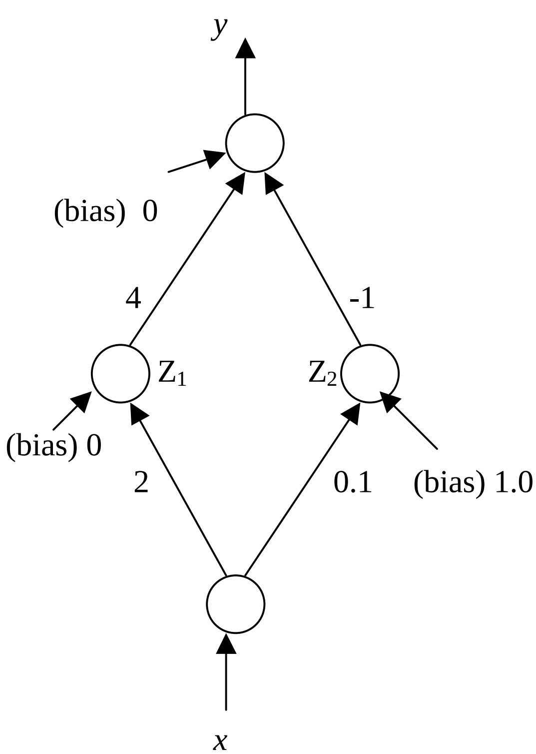

Figure A1

Training Data

x	$\hat{y}$
0	2
1	4

Solution:

(1) $y = y(x, \underset{\sim}{w})$ relation:

$$y = w^3_{10} + w^3_{11} z_1 + w^3_{12} z_2$$

$$= v_0 + v_1 z_1 + v_2 z_2$$

$$z_1 = \frac{1}{1 + e^{-(w^2_{10} + w^2_{11} x)}} = \frac{1}{1 + e^{-(w_0 + w_1 x)}}$$

$$z_2 = \frac{1}{1 + e^{-(w^2_{20} + w^2_{21} x)}} = \frac{1}{1 + e^{-(u_0 + u_1 x)}}$$

$$\underset{\sim}{w} = \begin{bmatrix} v_0 \\ v_1 \\ v_2 \\ w_0 \\ w_1 \\ u_0 \\ u_1 \end{bmatrix}$$

(2) $\frac{\partial y}{\partial \underset{\sim}{w}}$:

$$\frac{\partial y}{\partial v_0} = 1$$

$$\frac{\partial y}{\partial v_1} = z_1$$

$$\frac{\partial y}{\partial v_2} = z_2$$

③

$$\frac{\partial y}{\partial w_0} = v_1 \cdot \frac{\partial z_1}{\partial w_0} = v_1 z_1 (1-z_1)(-1)$$

Note: $$\frac{\partial \left(\frac{1}{1+e^{-r}}\right)}{\partial r} = \frac{-1}{(1+e^{-r})^2}(e^{-r})(-1)$$

$$= \left(\frac{1}{1+e^{-r}}\right)\left(\frac{e^{-r}+1-1}{1+e^{-r}}\right)$$

$$= \frac{1}{1+e^{-r}}\left(1 - \frac{1}{1+e^{-r}}\right)$$

$$\frac{\partial y}{\partial w_1} = v_1 z_1 (1-z_1) x$$

$$\frac{\partial y}{\partial u_0} = v_2 z_2 (1-z_2)$$

$$\frac{\partial y}{\partial u_1} = v_2 z_2 (1-z_2) x$$

(3) Initially:

$$\underset{\sim}{w} = \begin{bmatrix} 0 \\ 4 \\ -1 \\ 0 \\ 2 \\ 1 \\ 0.1 \end{bmatrix}$$

Feed 1st training sample:

④

$$Z_1 = \frac{1}{1+e^0} = 0.5$$

$$Z_2 = \frac{1}{1+e^{-1}} = 0.731$$

$$y = 4 \times Z_1 - Z_2 = 1.269$$

$$E = \frac{1}{2}(y-2)^2$$

$$\frac{\partial E}{\partial \underset{\sim}{w}} = (y-2)\begin{bmatrix} 1 \\ Z_1 \\ Z_2 \\ V_1 Z_1(1-Z_1) \\ V_1 Z_1(1-Z_1)x \\ V_2 Z_2(1-Z_2) \\ V_2 Z_2(1-Z_2)x \end{bmatrix} = \begin{bmatrix} 1 \\ 0.5 \\ 0.731 \\ 1 \\ 0 \\ -0.197 \\ 0 \end{bmatrix} \times (1.269-2) = \begin{bmatrix} -0.731 \\ -0.366 \\ -0.534 \\ -0.731 \\ 0 \\ 0.144 \\ 0 \end{bmatrix}$$

$$\underset{\sim}{w} = \underset{\sim}{w} - \lambda \frac{\partial E}{\partial \underset{\sim}{w}} = \begin{bmatrix} 0 \\ 4 \\ -1 \\ 0 \\ 2 \\ 1 \\ 0.1 \end{bmatrix} - 0.1 \begin{bmatrix} -0.731 \\ -0.366 \\ -0.534 \\ -0.731 \\ 0 \\ 0.144 \\ 0 \end{bmatrix} = \begin{bmatrix} 0.073 \\ 4.037 \\ -0.947 \\ 0.073 \\ 2 \\ 0.986 \\ 0.1 \end{bmatrix}$$

(4) Feed next training sample:
with new values of $\underset{\sim}{w}$,

$$Z_1 = \frac{1}{1+e^{-2.073}} = 0.888$$

$$Z_2 = \frac{1}{1+e^{-1.086}} = 0.748$$

$$y = 0.073 + 4.037 \times 0.888 - 0.947 \times 0.748$$
$$= 2.950$$

$$\frac{\partial E}{\partial \underset{\sim}{w}} = (y-4)\begin{bmatrix} 1 \\ Z_1 \\ Z_2 \\ V_1 Z_1 (1-Z_1) \\ V_1 Z_1 (1-Z_1) x \\ V_2 Z_2 (1-Z_2) \\ V_2 Z_2 (1-Z_2) x \end{bmatrix} = -1.05 \begin{bmatrix} 1 \\ 0.888 \\ 0.748 \\ 0.402 \\ 0.402 \\ -0.179 \\ -0.179 \end{bmatrix} = \begin{bmatrix} -1.05 \\ -0.932 \\ -0.785 \\ -0.422 \\ -0.422 \\ 0.188 \\ 0.188 \end{bmatrix}$$

$$\underset{\sim}{w} = \underset{\sim}{w} - \lambda \frac{\partial E}{\partial \underset{\sim}{w}} = \begin{bmatrix} 0.073 \\ 4.037 \\ -0.947 \\ 0.073 \\ 2 \\ 0.986 \\ 0.1 \end{bmatrix} - 0.1 \begin{bmatrix} -1.05 \\ -0.932 \\ -0.785 \\ -0.422 \\ -0.422 \\ 0.188 \\ 0.188 \end{bmatrix} = \begin{bmatrix} 0.178 \\ 4.13 \\ -0.868 \\ 0.115 \\ 2.042 \\ 0.967 \\ 0.081 \end{bmatrix}$$

This finishes 1st epoch □

Exercise 2:

This exercise is the same as Exercise 1 except that we use the Conjugate Gradient method instead of backpropagation, and the training error here is:

$$E = \frac{1}{2}\sum_{i=1}^{P}(y(x_i, w) - \hat{y}_i)^2$$

where index i means training data sample #i, $i = 1, 2, \ldots, P$; P is the total number of samples in training data; and $\hat{y}_i$ is the target value for neural network output neuron, i.e., $\hat{y}_i$ is a value of sample #i in the training data. Assume a step length $\lambda = 0.1$.

Solutions:

Steps (1) & (2): Same as Exercise 1.

(3) Initially:

$$\underset{\sim}{w} = \begin{bmatrix} 0 \\ 4 \\ -1 \\ 0 \\ 2 \\ 1 \\ 0.1 \end{bmatrix}$$

Feed 1st training sample, $x = 0$,

$z_1 = 0.5$

$z_2 = 0.731$

$y = 1.269$

$$\frac{\partial y}{\partial \underset{\sim}{w}} = \begin{bmatrix} 1 \\ z_1 \\ z_2 \\ v_1 z_1 (1 - z_1) \\ v_1 z_1 (1 - z_1) x \\ v_2 z_2 (1 - z_2) \\ v_2 z_2 (1 - z_2) x \end{bmatrix} = \begin{bmatrix} 1 \\ 0.5 \\ 0.731 \\ 1 \\ 0 \\ -0.197 \\ 0 \end{bmatrix}$$

Feed 2nd Sample : $x = 1$,

$$z_1 = \frac{1}{1+e^{-z}} = 0.881$$

$$z_2 = \frac{1}{1+e^{-1.1}} = 0.75$$

$$y = 4 \cdot z_1 - z_2 = 2.774$$

$$\frac{\partial y}{\partial \underset{\sim}{w}} = \begin{bmatrix} 1 \\ z_1 \\ z_2 \\ v_1 z_1(1-z_1) \\ v_1 z_1(1-z_1)x \\ v_2 z_2(1-z_2) \\ v_2 z_2(1-z_2)x \end{bmatrix} = \begin{bmatrix} 1 \\ .881 \\ .75 \\ .419 \\ .419 \\ -.1875 \\ -.1875 \end{bmatrix}$$

$$\underset{\sim}{\nabla} E = \frac{\partial E}{\partial \underset{\sim}{w}} = \sum_{i=1}^{p} (y - y_i) \frac{\partial y}{\partial \underset{\sim}{w}}$$

$$(1.269-2)\begin{bmatrix} 1 \\ .5 \\ .731 \\ 1 \\ 0 \\ -.197 \\ 0 \end{bmatrix} + (2.774-4)\begin{bmatrix} 1 \\ .881 \\ .75 \\ .419 \\ .419 \\ -.1875 \\ -.1875 \end{bmatrix}$$

$$= \begin{bmatrix} -1.957 \\ -1.446 \\ -1.454 \\ -1.245 \\ -0.514 \\ .374 \\ .23 \end{bmatrix}$$

For 1st epoch:

$$\underset{\sim}{h} = -\nabla \underset{\sim}{E}$$

$$\underset{\sim}{w} = \underset{\sim}{w} + \lambda \underset{\sim}{h} = \begin{bmatrix} 0 \\ 4 \\ -1 \\ 0 \\ 2 \\ 1 \\ 0.1 \end{bmatrix} - 0.1 \begin{bmatrix} -1.957 \\ -1.446 \\ -1.454 \\ -1.245 \\ -.514 \\ .374 \\ .23 \end{bmatrix}$$

$$= \begin{bmatrix} 0.1957 \\ 4.1446 \\ -0.8546 \\ 0.1245 \\ 2.0514 \\ 0.9626 \\ 0.077 \end{bmatrix}$$

This Finishes 1st epoch

$\square$