CS 615 Assignment2

Himanshu Gupta

May 2020

1 Theory

1.1

Given this hyperbolic tangent function:

$$tanh(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$$

In order to calculate the function:

$$\frac{\partial tanh(x\theta)}{\partial \theta_j}$$

Let the z = $x\theta$ so using chain rule the function becomes $\frac{\partial tanh(z)}{\partial \theta_j}$

$$\frac{\partial tanh(z)}{\partial \theta_j} = \frac{\partial tanh(z)}{\partial z_j} \frac{\partial (z_j)}{\partial \theta_j}$$

$$\frac{\partial tanh(z)}{\partial z_j} = \frac{((e^{z_j} + e^{-z_j})(e^{z_j} + e^{-z_j}) - (e^{z_j} - e^{-z_j})(e^{z_j} - e^{-z_j}))}{(e^{z_j} + e^{-z_j})^2} = \frac{((e^{z_j} + e^{-z_j})^2 - (e^{z_j} - e^{-z_j})^2)}{(e^{z_j} + e^{-z_j})^2} = 1 - \frac{(e^{z_j} - e^{-z_j})^2}{(e^{z_j} + e^{-z_j})^2}$$

$$= 1 - (tanh(z_j)^2)$$

$$\frac{\partial (z_j)}{\partial \theta_j} = \frac{\partial (x\theta)}{\partial \theta_j} = x_j$$

Using the above two results:

$$\frac{\partial tanh(z)}{\partial \theta_j} = (1 - (tanh(z_j)^2))(x_j)$$

1.2

1.2.1

$$\frac{\partial J}{\partial \theta_{j,k}} = \frac{\partial J}{\partial g(net_{o_k})} \cdot \frac{\partial g(net_{o_k})}{\partial net_{o_k}} \cdot \frac{\partial net_{o_k}}{\partial \theta_{jk}}$$

In this case, our J (objective function is cross entropy), the output layer has softmax activation function, and the hidden layer has the linear activation function.

The Cross Entropy objective function is given below:
J = $\sum_{k=1}^{K} -y_k log\hat{y}_k$

Since, we are using one-hot encoded distribution so only one y is 1. Let $y_a = 1$ then

J $= -log\hat{y}_a$

so $\frac{\partial J}{\partial g(net_{o_k})} = \frac{-1}{\hat{y}_a} = -\frac{\sum_{k=1}^{K} e^{net_{o_k}}}{e^{net_{o_a}}}$

$g(net_{o_k}) = \frac{e^{net_{o_a}}}{\sum_{k=1}^{K} e^{net_{o_k}}}$

if j $\neq$ a, $-\frac{g(net_{o_k})}{net_{o_k}} = -\frac{(e^{net_{o_a}})(e^{net_{o_k}})}{(\sum_{k=1}^{K} e^{net_{o_k}})^2}$

if j = a, $\frac{g(net_{o_k})}{net_{o_k}} = \frac{(e^{net_{o_a}})(\sum_{k=1}^{K} e^{net_{o_k}})-(e^{net_{o_a}})(e^{net_{o_a}})}{(\sum_{k=1}^{K} e^{net_{o_k}})^2}$

$\frac{\partial net_{o_k}}{\partial \theta_{j,k}} = x_j$ for j = a, otherwise zero

Therefore, using the results of the above equations:

Case 1: $j \neq a$

$\frac{\partial J}{\partial \theta_{j,k}} = (\frac{\sum_{k=1}^{K} e^{net_{o_k}}}{e^{net_{o_a}}})(\frac{(e^{net_{o_a}})(e^{net_{o_k}})}{(\sum_{k=1}^{K} e^{net_{o_k}})^2})(x_j) = (x_j)(\hat{y}_k)$

Similarly, Case 2: $j = a$

$\frac{\partial J}{\partial \theta_{j,k}} = -(\frac{\sum_{k=1}^{K} e^{net_{o_k}}}{e^{net_{o_a}}})(\frac{(e^{net_{o_a}})(\sum_{k=1}^{K} e^{net_{o_k}})-(e^{net_{o_a}})(e^{net_{o_a}})}{(\sum_{k=1}^{K} e^{net_{o_k}})^2})(x_j) = (x_j)(\hat{y}_k - 1)$

Hence, $\frac{\partial J}{\partial \theta_{j,k}} = x_j(\hat{y}_k - y_k)$

1.2.2

$$\frac{\partial J}{\partial \beta_{i,j}} = \frac{\partial J}{\partial g(net_o)} \cdot \frac{\partial g(net_o)}{\partial net_o} \cdot \frac{\partial net_o}{\partial g\left(net_{h_j}\right)} \cdot \frac{\partial g\left(net_{h_j}\right)}{\partial net_{h_j}} \cdot \frac{\partial net_{h_j}}{\partial \beta_{ij}}$$

The first two terms are already calculated in the previous question.

$$\frac{\partial net_o}{\partial g\left(net_{h_j}\right)} = \frac{\partial net_o}{\partial h_j} = \theta_j$$

As activation function of hidden layer is linear (g(net_{h_j}) = net_{h_j}).

$$\frac{\partial g\left(net_{h_j}\right)}{\partial net_{h_j}} = 1$$

$\frac{\partial net_{h_j}}{\partial \beta_{ij}} = x_i$

Combining all the results from the above equations, we get

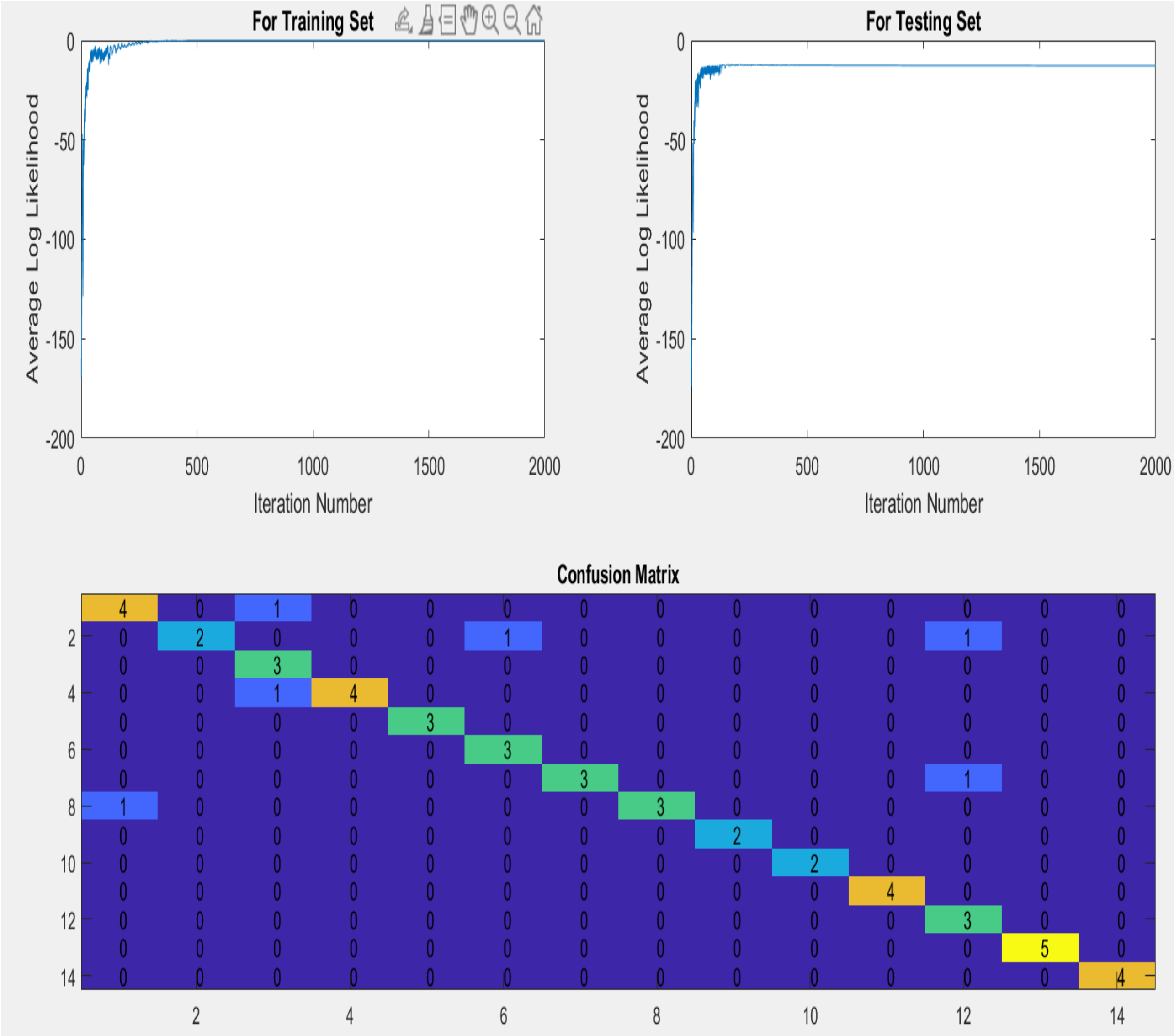
Case 1: j $\neq$ a , $\frac{\partial J}{\partial \beta_{ij}} = (\frac{\sum_{k=1}^{K} e^{net_{o_k}}}{e^{net_{o_a}}})(\frac{(e^{net_{o_a}})(e^{net_{o_k}})}{(\sum_{k=1}^{K} e^{net_{o_k}})^2})(\theta_j)(x_i) = (x_i)(\hat{y}_k)(\theta_j)$

Similarly, Case 2: j = a, $\frac{\partial J}{\partial \beta_{ij}} = -(\frac{\sum_{k=1}^{K} e^{net_{o_k}}}{e^{net_{o_a}}})(\frac{(e^{net_{o_a}})(\sum_{k=1}^{K} e^{net_{o_k}})-(e^{net_{o_a}})(e^{net_{o_a}})}{(\sum_{k=1}^{K} e^{net_{o_k}})^2})(x_i)(\theta_j)$

$= (x_i)(\hat{y}_k - 1)(\theta_j)$

Hence, $\frac{\partial J}{\partial \beta_{i,j}} = (x_i)(\hat{y}_k - y_k)(\theta_j)$

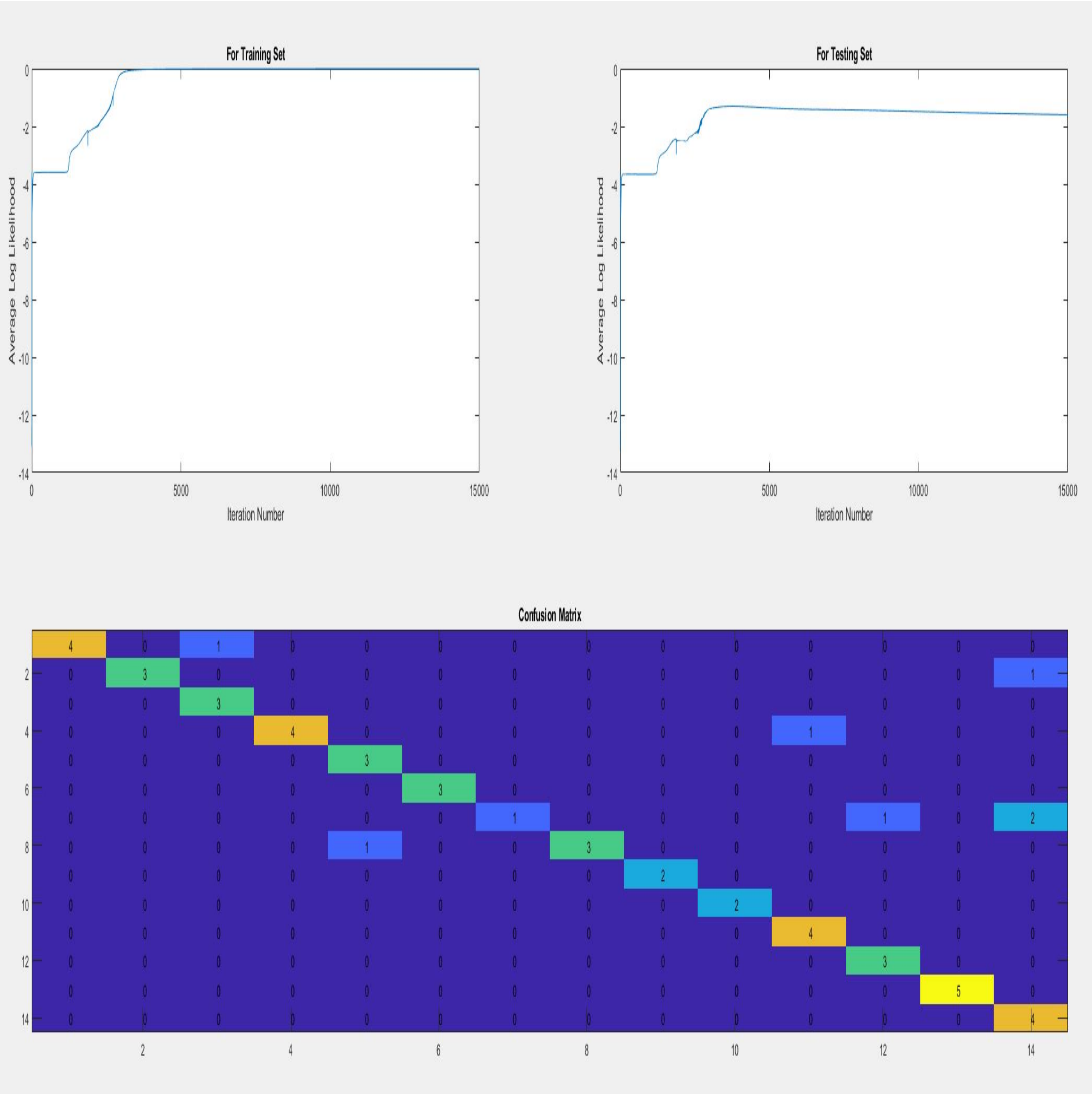
2 Shallow ANN



The initial values of θs are randomly chosen between [0,1], then made even smaller by multiplying with 0.001. Accuracy = 88.24%, Termination Criteria is max iterations = 2000, Learning Rate (eta) = 0.8, and L2 Regularization term (alpha) = 0.00001.

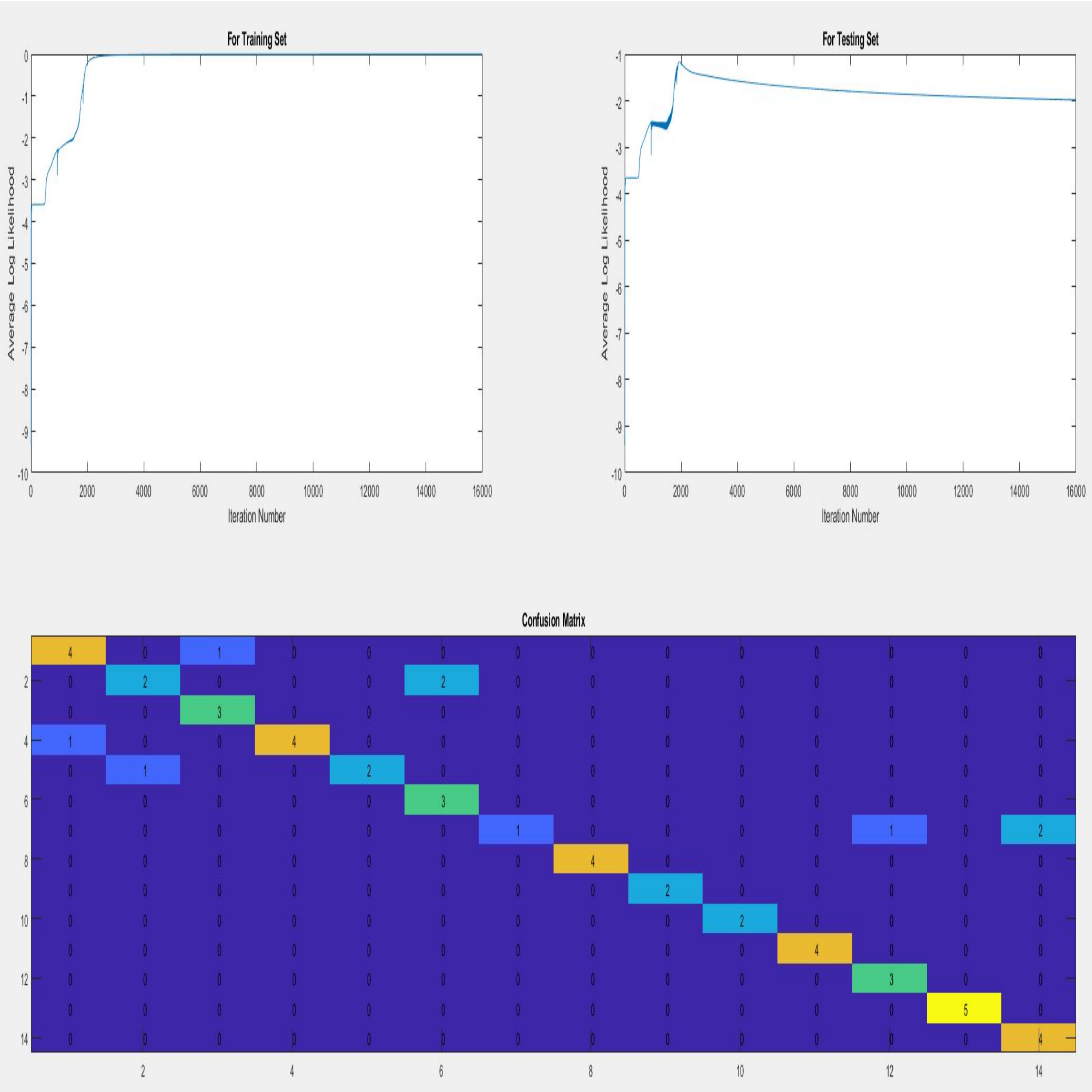
3 Multi-layers ANN

3.1 Network Configuration 1



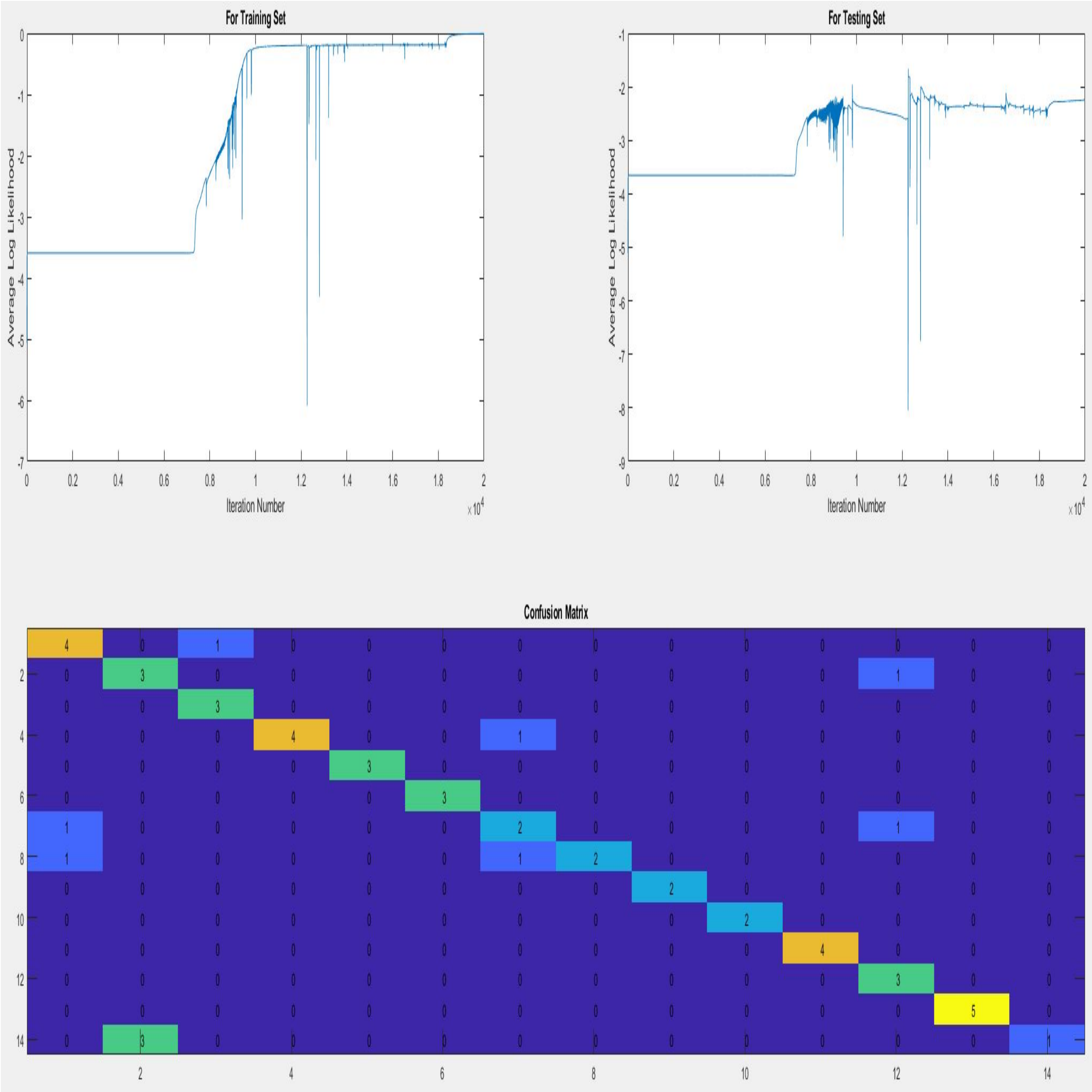
The initial values of θs are randomly chosen between [0,1], then made even smaller by multiplying with 0.001. Hidden Layers nodes are [800, 400, 200], Learning rate (eta) = 0.55, Termination Criteria is max iterations = 15000, Accuracy = 86%, and L2 regularization term (alpha) = 0.00001.

3.2 Network Configuration 2



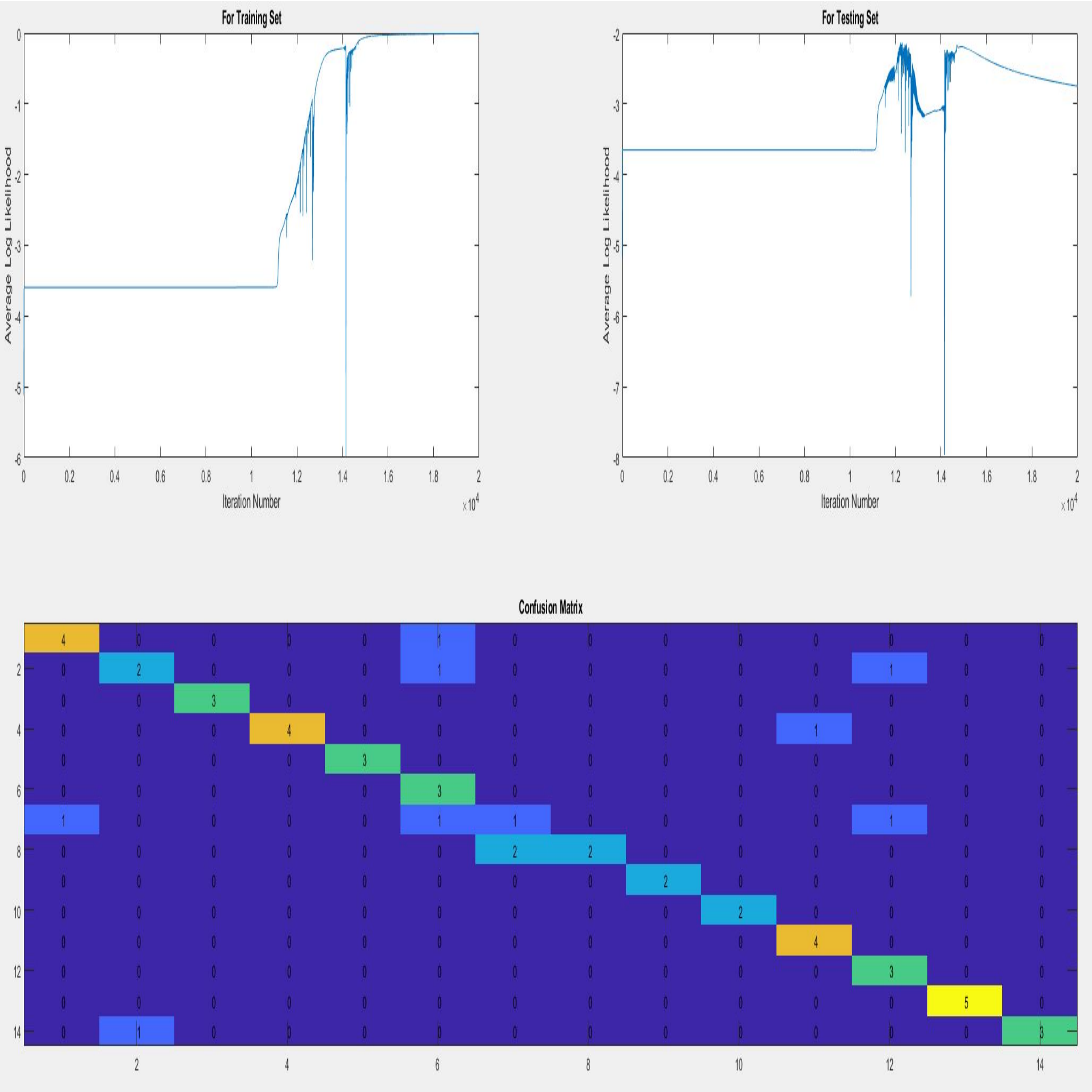
The initial values of θs are randomly chosen between [0,1], then made even smaller by multiplying with 0.001. Hidden layers nodes are [1000,400,100], Accuracy = 84%, Learning rate (eta) = 0.8, Termination Criteria is max iterations = 16000, and L2 regularization term (alpha) = 0.00001.

3.3 Network Configuration 3



The initial values of θs are randomly chosen between [0,1], then made even smaller by multiplying with 0.001. Hidden layers nodes are [1200 400 100 50], Accuracy = 80%, Learning rate (eta) = 0.91, Termination Criteria is max iteration = 20000 and L2 regularization term (alpha) = 0.00001.

3.4 Network Configuration 4



The initial values of θs are randomly chosen between [0,1], then made even smaller by multiplying with 0.001. Hidden layers are [1000,200,100,50], 80.39%, Learning Rate (eta) = 0.92, Termination Criteria is max iteration = 20000 and L2 regularization term (alpha) = 0.00001.