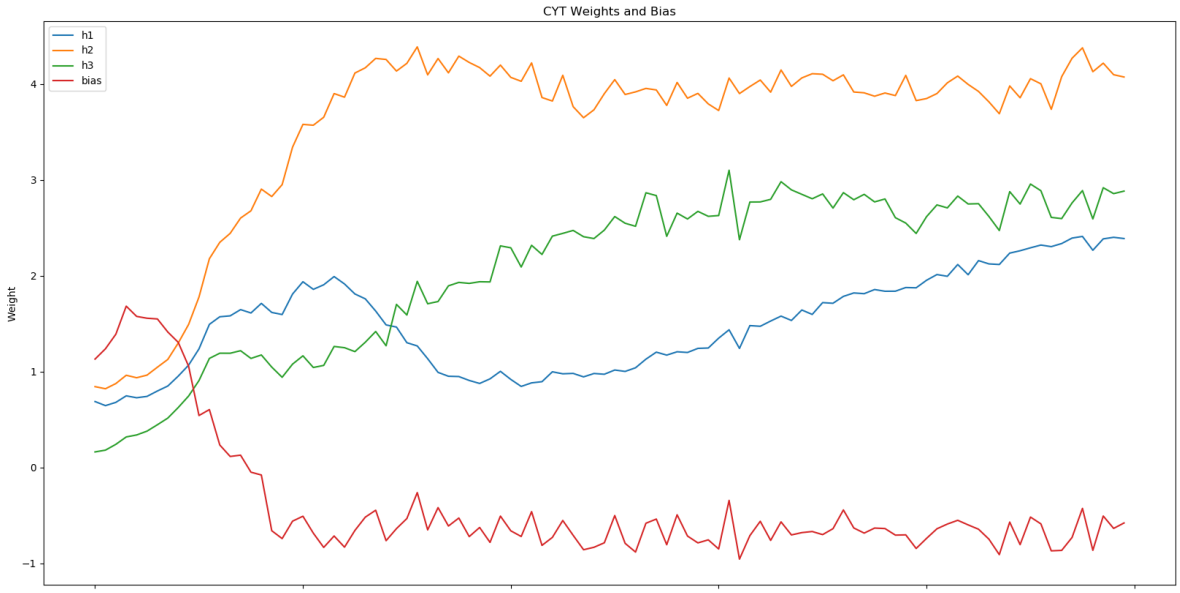
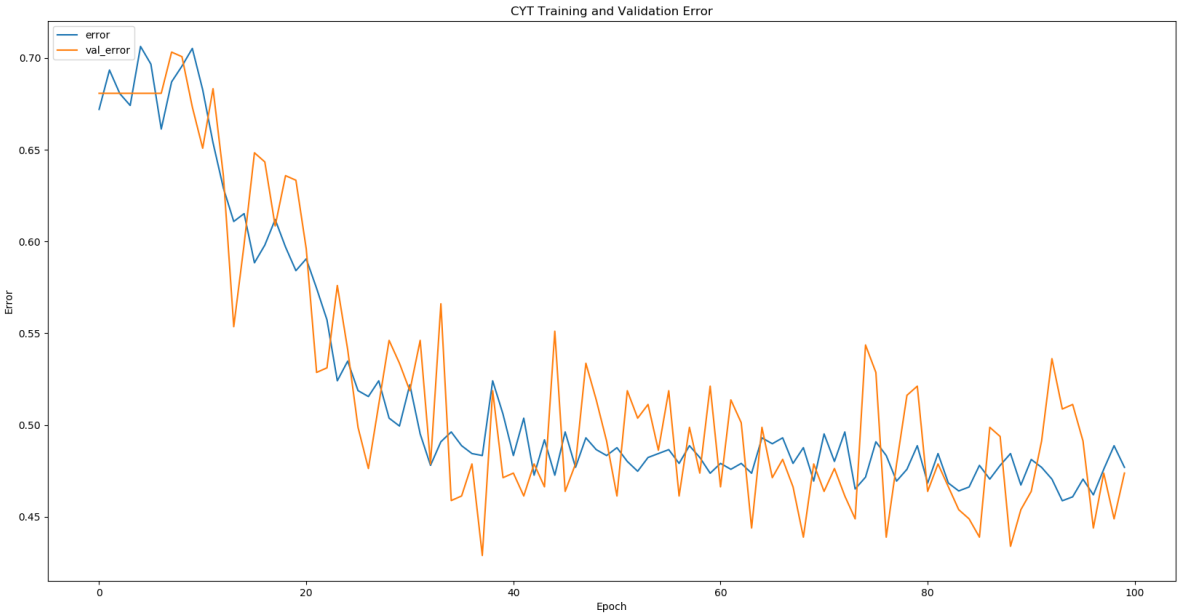
Homework 2

Ryan Gosiaco 912819444

1. When considering the dataset as a whole (all of the classes together), there are 149 outliers detected by both LOF and Isolation Forest. LOF and Isolation Forest do not entirely agree, there are different outliers detected. LOF assumes that an outlier is a point whose local density (the typical distance a point can be reached by its neighbors) is substantially lower than their surrounding neighbors. It is essentially trying to cluster the data and see which points fall outside of the main clusters. Isolation Forest assumes that an outlier is a point that is easily separated. It is basically drawing lines to separate a point from the others, and an outlier is a point that requires less lines than other points. When running either method on the whole dataset rather than by class, both methods end up removing the least represented class (ERL) due to the class imbalance present in the dataset. I decided to use LOF because I felt that having samples that are grouped together will help train the model quicker.

2. The first graph is of the weight values for CYT per epoch. h1 stands for the first hidden node in hidden layer 2, h2 is the second hidden node in layer 2, and so forth. The second graph is the training and validation error for CYT per epoch.





3. The training error I got when training all of the samples (1483 samples) is 0.5172 after 100 epochs. I would have trained the model on the dataset without the outliers but the homework instructions specified all of the data (1484 samples) so that is what I did.
The final activation function formula for the CYT class is

$$\mathrm{a_{cyt}} = \mathrm{softmax}\left(w_{i0}^{(4)} + a_1^{(3)}w_{11}^{(4)} + a_2^{(3)}w_{12}^{(4)} + a_3^{(3)}w_{13}^{(4)}\right)$$

$$a_i^{(3)} = sigmoid\left(w_{i0}^{(3)} + a_1^{(2)}w_{i1}^{(3)} + a_2^{(2)}w_{i2}^{(3)} + a_3^{(2)}w_{i3}^{(3)}\right) \text{ for i } = 1,2,3$$

$$a_i^{(2)} = sigmoid\left(w_{i0}^{(2)} + x_1w_{i1}^{(2)} + x_2w_{i2}^{(2)} + x_3w_{i3}^{(2)} + x_4w_{i4}^{(2)} + x_5w_{i5}^{(2)} + x_6w_{i6}^{(2)} + x_7w_{i7}^{(2)} + x_8w_{i8}^{(2)}\right)$$

$$\text{for i } = 1,2,3$$

where $a_i^{(3)}$is each activation function for the second hidden layer and $a_i^{(2)}$is each activation function for the first hidden layer.

The weights from the input layer to the first hidden layer are:

[12.232405 , -1.7977852 , -4.381492], [7.54965 , 0.24999769, -4.170795],
[-7.0225554 , -25.454842 , -0.44205657], [-1.2620404 , -1.3720942 , -7.5584393],
[1.0309734 , 1.1974117 , 2.3233268], [0.20570916, 2.97804 , -5.7364373],
[-5.09074 , 2.2773097 , -3.0158384], [2.9651701 , -0.60071135, 9.817527]
which are in the form of $[w_{1k}, w_{2k}, w_{3k}]$ where k goes from 1 to 8.

The bias from the input layer to the first hidden layer are:

[-9.120532 , 8.581263 , 4.4262295] which is in the form $[w_{1k}, w_{2k}, w_{3k}]$ where k=0.

The weights from the first hidden layer to the second hidden layer are:

[2.1103613, -7.449023 , -5.4305844], [-7.1810985, -9.323466 , 1.2976029],

[3.1821432, -2.5067208, 6.9078407] with the form $[w_{1k}, w_{2k}, w_{3k}]$ where k goes from 1 to 3.

The bias from the first hidden layer to the second hidden layer are:

[-0.87308806, 0.9993863 , -0.05872086] which is in the form $[w_{1k}, w_{2k}, w_{3k}]$ where k=0.

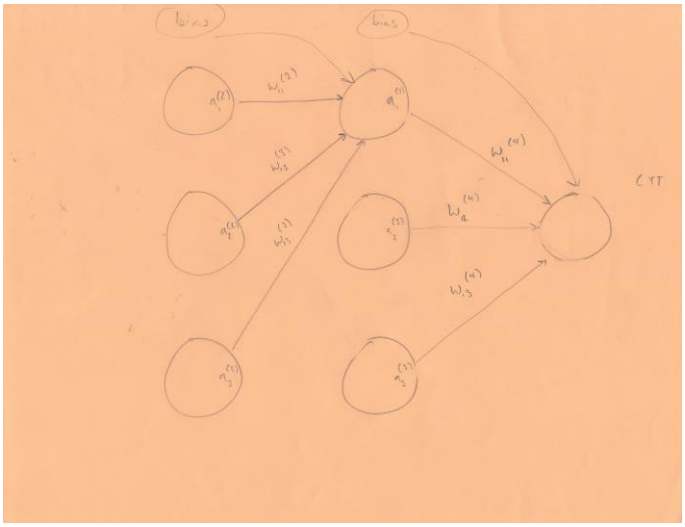
The weights from the second hidden layer to the output layer are:

[2.3352048 , 0.23535927, 3.8726716 , -3.1838393 , -0.22626811, -3.164714 , -1.3814996 , 3.95636, -2.48736 , 0.1124379],
[5.196895 , -0.7593069 , 2.8684623 , -6.479663 , -4.724142 , -6.9529085 , 6.0733094 , 2.7969959, 2.1669145 , 0.77876115],
[4.1115484 , -4.4512115 , -4.464537 , -6.2618675 , -0.50626224, 7.4318547 , 0.27980572, 4.4472375, 0.30765137, 1.9564188]
with the form $[w_{1k}, w_{2k}, w_{3k}, w_{4k}, w_{5k}, w_{6k}, w_{7k}, w_{8k}, w_{9k}, w_{10k}]$ where k goes from 1 to 3.

The bias from the second hidden layer to the output layer are:

[-2.4761837, 0.4777775, 0.8006015, 3.9217372, 2.0319924, -2.4957752, 0.7204365, -2.5188813, 0.2861757, -0.628655]
with the form $[w_{1k}, w_{2k}, w_{3k}, w_{4k}, w_{5k}, w_{6k}, w_{7k}, w_{8k}, w_{9k}, w_{10k}]$ where k=0.

4.Unfortunately, I ran out of time but here are the weights I would be looking for.



5. Since the grid search focuses on determining what combination of hidden layers and nodes is ideal, I set the bias initializer of each layer in each model to a RandomNormal initializer with the same seed to hopefully remove some of the potential randomness when training the models. The optimal configuration that I found is 1 hidden layer with 12 hidden nodes which resulted in an error of 0.41645885. The general relationship between the number of hidden layers and number of hidden nodes with the generalization error is that as you increase model complexity (increasing the number of hidden layers or hidden nodes), the error tends to increase.

6. Using the model from Problem 3 which was trained on the full dataset. I predicted [1.9569211e-01, 5.6513300e-06, 2.5874836e-04, 1.3355308e-06, 1.3789815e-03, 5.3582368e-03, 2.0145176e-02, 7.6462394e-01, 9.3840651e-04, 1.1597392e-02] which means the unknown sample belongs to the class “POX” because my labels are encoded as ['CYT', 'EXC', 'ME1', 'ME2', 'ME3', 'MIT', 'NUC', 'POX', 'VAC', ‘ERL’].

7.