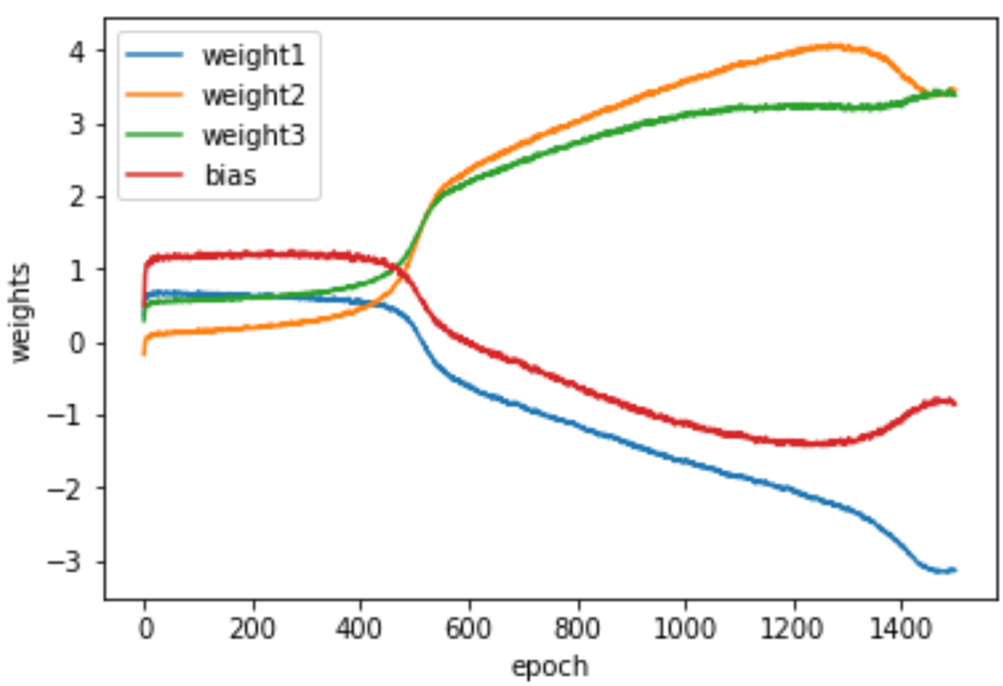
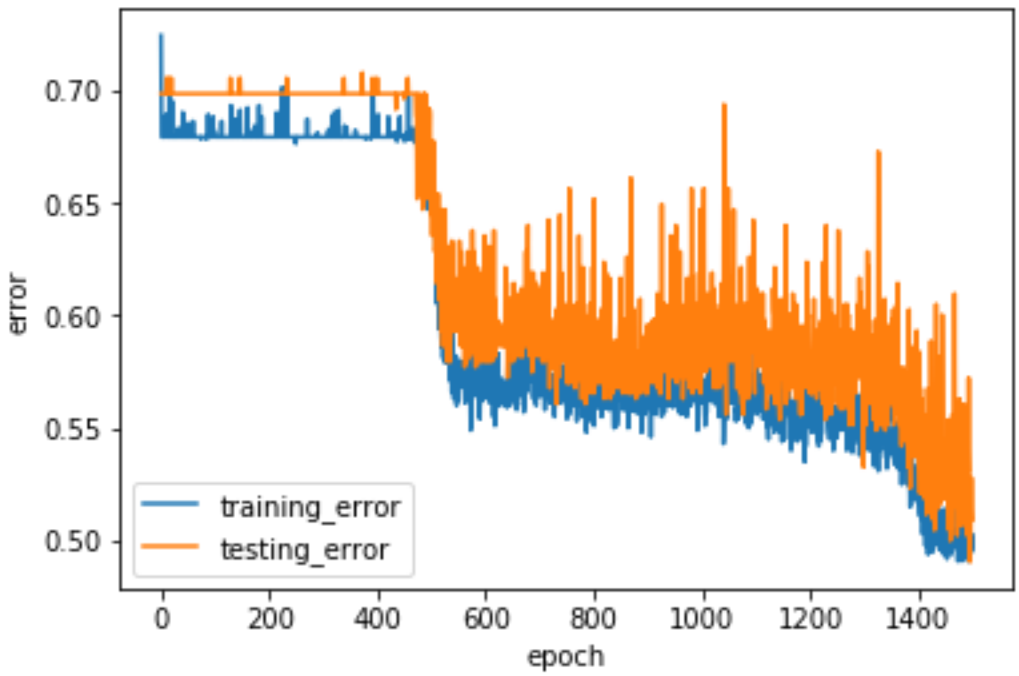
HW2 report
Xingwei Ji
914401326

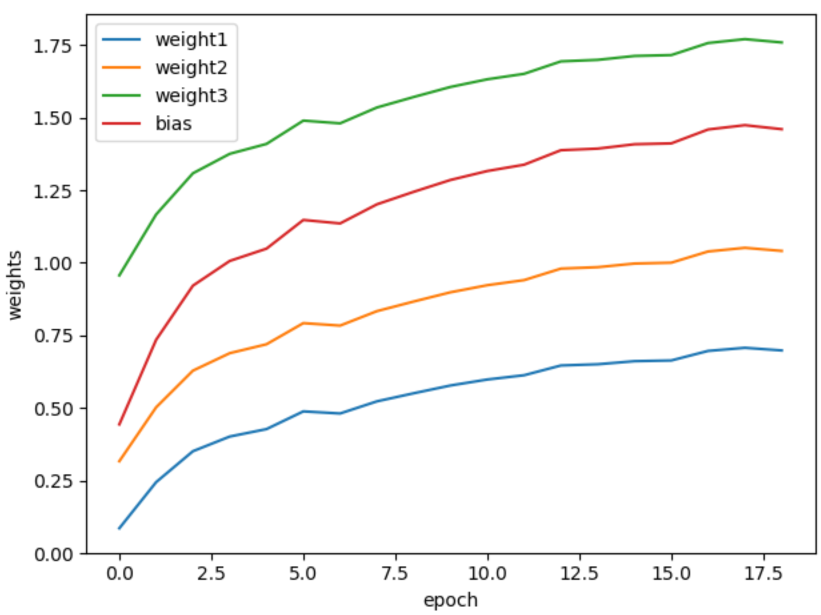
1. Outlier
    a. Yes, there are outliers in the dataset
    b. no, the isolation forest and LOF don't agree each other. I think the reason why they disagree is that the ways they detect outliers are different. LOF will draw circles from each point and see how many other points are included. This is called k numbers. IsoForest tries to draw random lines to separate data points. If a point takes fewer lines to separate, it would be an outlier.
    c. The local outlier factor (LOF) method scores points in a multivariate dataset whose rows are assumed to be generated independently from the same probability distribution.
    The isolation-based approach measures how isolated a point is without any assumption on the data distribution

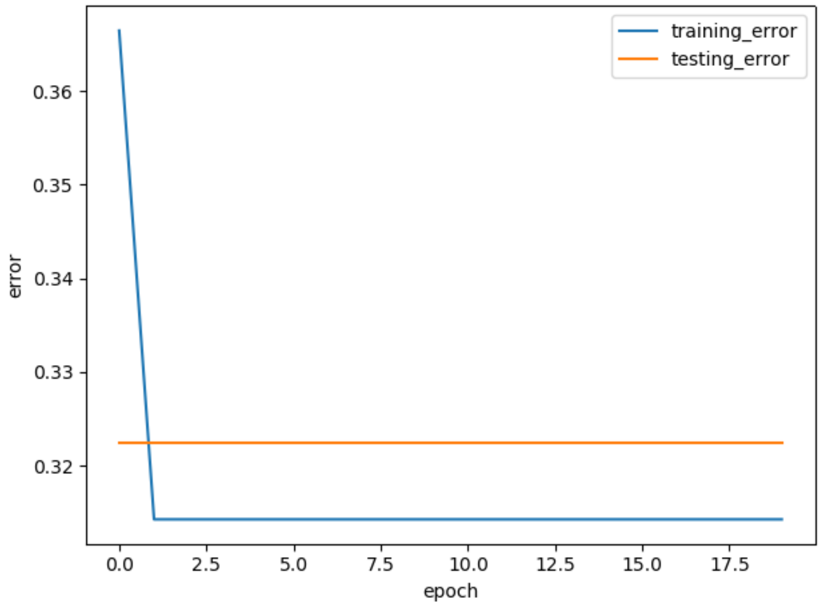
   I used Isolation forest to detect outliers. I separated the ERL data out and run outlier algorithm in the rest data because there are only five of ERL, and they are pretty far from the rest of other classes. As a result, the outlier detector will consider them as outliers and wipe out all ERL classes. I append the data of ERL class after the removal of outliers in the rest data and do the fitting.
2. Graph
   I am not sure which method to use, I used normal 10 classes classification and get the following result.(q_two_normal.py)





I also used CYT class versus all method which sets CYT class to 1 and anything else to 0.(q_two_onevsall.py)
Here is the graph:





3. The training error is 49%.

$W^1_{i,j}$:

i\j	1	2	3
1	-2.1102152	-1.4224404	-6.1384377
2	-0.95252943	-2.258863	-0.5459768
3	[-2.2399316	-1.5370386	-2.4138312
4	1.7532674	-2.2792144	3.2550364
5	1.2469106	3.9437149	1.2858084
6	-0.92298543	1.9879668	-1.9019877
7	2.1192098	-1.0134993	3.371361
8	0.65852046	1.9403076	1.36866
Bias weights	-0.26492488	1.042711	-1.2803181

W^2_{ji}:

i\j	1	2	3
1	1.5685434	2.3213522	2.5931113
2	3.6316113	3.2313075	-3.910231
3	6.425479	3.610899	6.657158
Bias weights	-3.2191505	-5.2541513	-3.000276

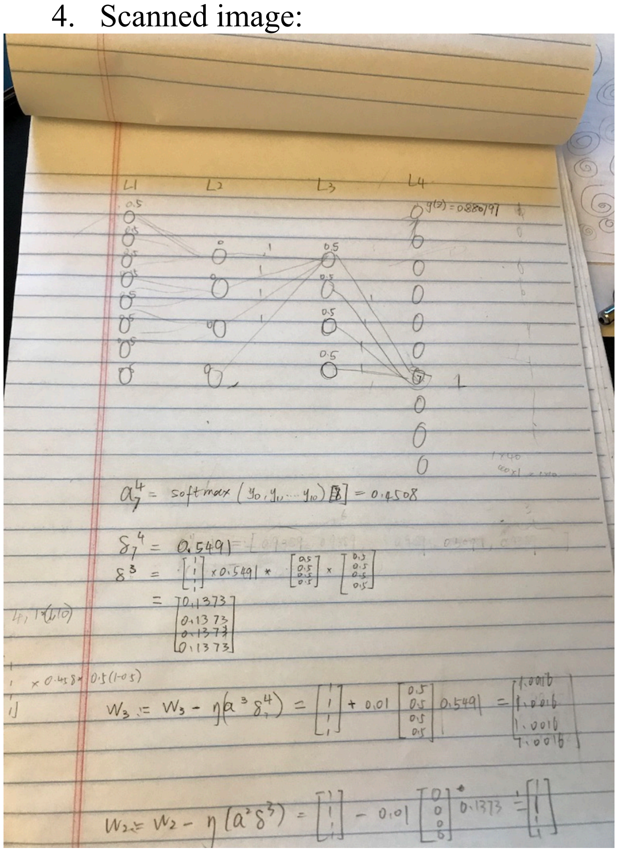
i\j	1	2	3	4	5	6	7	8	9	10
1	3.4098232	-2.2465796	-5.4599202	-4.7452083	-1.4367447	4.4570074	1.5751253	1.7695584	-1.5391823	5.3518766
2	3.3591673	-1.6011828	-2.2017803	-3.7939258	-1.1232570	-1.4322712	-1.8251935	5.8142719	-4.7003850	1.2893586
3	2.1966605	-9.2606425e-01	9.5691539e-02	-3.7555292	-6.5038025e-01	-5.4527917	6.8054771	5.5964011e-01	8.4109461e-01	1.1623959e-01
Bias weights	-3.1517823	0.13407792	1.1397935	3.375799	1.9995346	0.13785586	-0.8446069	-2.443163	0.50435436	-0.8518435

The formula of activation function is

$$P(y = j \mid \mathbf{x}) = \frac{e^{\mathbf{x}^\mathsf{T}\mathbf{w}_j}}{\sum_{k=1}^{K} e^{\mathbf{x}^\mathsf{T}\mathbf{w}_k}}$$

Where w_j is (W11, W12, W13, Bias1)
Wk is W^3_{ij}

4. Scanned image:



It matches the output:

```
array([[1.0001142e+00, 4.0733942e-04, 4.0733942e-04],
       [1.0001142e+00, 4.0733942e-04, 4.0733942e-04],
       [1.0001142e+00, 4.0733942e-04, 4.0733942e-04]], dtype=float32),
array([1.0002284e+00, 8.1467885e-04, 8.1467885e-04], dtype=float32),
array([[-3.3461282e-04, -3.3461282e-04, -3.3461282e-04, -3.3461282e-04,
        -3.3461282e-04, -3.3461282e-04,  1.0030115e+00, -3.3461282e-04,
        -3.3461282e-04, -3.3461282e-04],
       [-1.8103975e-04, -1.8103975e-04, -1.8103975e-04, -1.8103975e-04,
        -1.8103975e-04, -1.8103975e-04,  1.0016294e+00, -1.8103975e-04,
        -1.8103975e-04, -1.8103975e-04],
       [-1.8103975e-04, -1.8103975e-04, -1.8103975e-04, -1.8103975e-04,
        -1.8103975e-04, -1.8103975e-04,  1.0016294e+00, -1.8103975e-04,
        -1.8103975e-04, -1.8103975e-04]], dtype=float32),
array([-3.6207950e-04, -3.6207950e-04, -3.6207950e-04, -3.6207950e-04,
       -3.6207950e-04, -3.6207950e-04,  1.0032587e+00, -3.6207950e-04,
       -3.6207950e-04, -3.6207950e-04], dtype=float32)]
```

5. Table:

Layer\Node	**3**	**6**	**9**	**12**
1	0.464953	0.455607	0.448598	0.436915
2	0.577102	0.492990	0.483644	0.483644
3	0.698598	0.628504	0.642523	0.602803

The optimal config is 1 hidden layer with 12 hidden nodes. The relationship between number of layers, number of nodes and generalization error is that the more layer, the higher the error. The more nodes, the less the error.

6. The predicted class is 7, so it's NUC
7. We can use the error rate which will tell us the miss predicted of the model. The uncertainty is 0.43