

Neural Network

Wenhe Li wl1508

```
(base) D:\MachineLearning\assignment4\python
-----Naive python-----
Time: 4.945989370346069
Error rate: 0.024800000000000044
Loss: 0.08688856037475011
-----NP python-----
Time: 1.7770001888275146
Error rate: 0.024800000000000044
Loss: 0.0868885603747501
-----Less Weight-----
Time: 0.7659871578216553
Error rate: 0.1412
-----ReLU-----
Time: 0.870018720626831
Error rate: 0.18500000000000005
-----ReLU & Less Weight-----
Time: 0.7259976863861084
Error rate: 0.24219999999999997
```

Above is the screenshot for **execution time**, **error rate**, and **loss** (for naive py only) By using numpy, we significantly reduce the **execution time** while doing matrix related operation.

By putting less weights, we make the model contain less information so that it will result in a drop of accuracy. As for the **ReLU** activation, compared with **softmax**, the parameters are trained upon **softmax** activation. Therefore, using **ReLU** will naturally reduce the accuracy. That's why **softmax** is better than **ReLU** and **Less Weight** will lead to a lower accuracy.