**HW5 Report**

Zhongze Tang (zt67)

**Q1 Neural Network**

I add a bias unit to the input layer and the hidden layer respectively. So the network looks like:



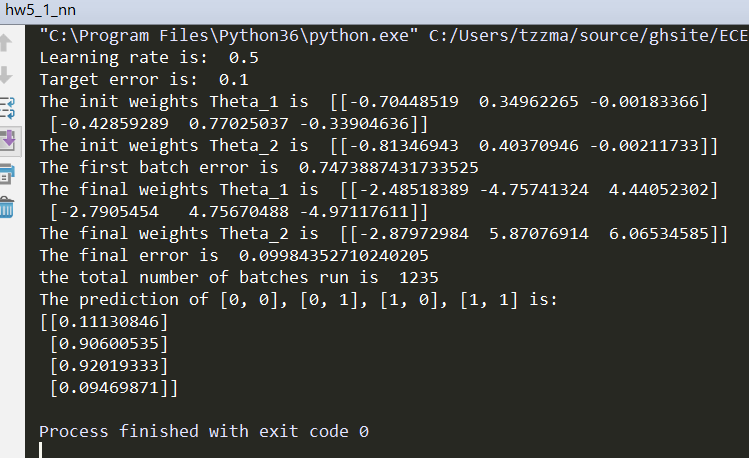
Note that sometimes the program won’t give you the result because the weights (theta1 and theta2) are initialized randomly, which may cause the cost function J won’t converge.

**You can change the learning\_rate and target\_error in Line 97 and 98.**

**Results**

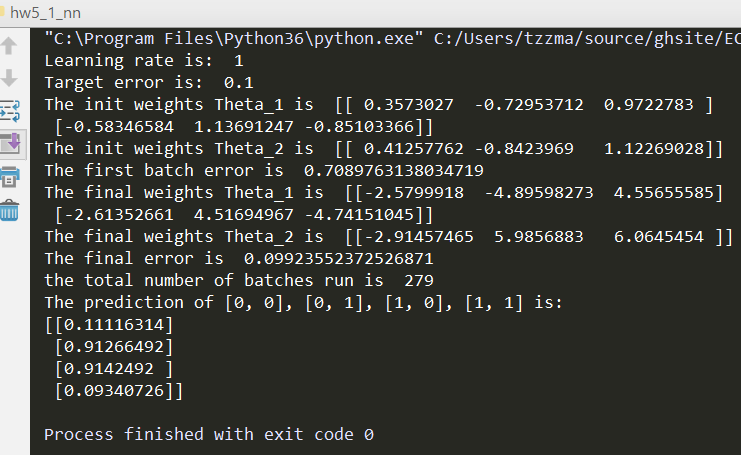
**Learning rate is: 0.5**

**Target error is: 0.1**



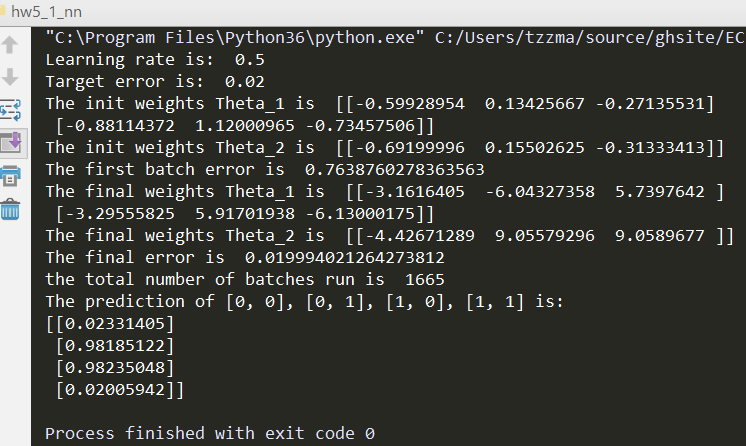
**Learning rate is: 1**

**Target error is: 0.1**



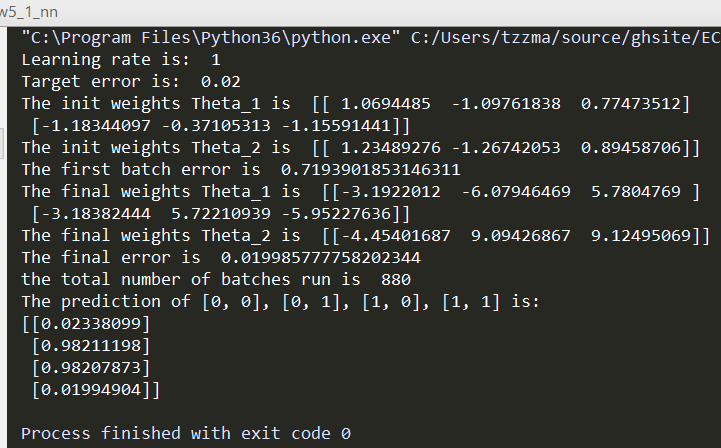
**Learning rate is: 0.5**

**Target error is: 0.02**



**Learning rate is: 1**

**Target error is: 0.02**



**Learning rate in [0.05, 0.1, 0.3, 0.5, 0.75, 1]**

The init weights Theta\_1 is [[-0.38404047 -0.06897211 1.12252285]

[ 0.99525219 -0.78971388 0.35975224]]

The init weights Theta\_2 is [[-0.44345173 -0.07964214 1.29617774]]

**Learning rate is: 0.05**

**Target error is: 0.1**

The first batch error is 0.7174708075735095

The final weights Theta\_1 is [[-2.27973707 5.47942452 5.48520361]

[ 6.06182938 -4.05718349 -4.05844979]]

The final weights Theta\_2 is [[-8.85899284 6.16701676 6.06162017]]

The final error is 0.09998439355685633

**the total number of batches run is 12035**

Learning rate is: 0.1

Target error is: 0.1

The first batch error is 0.7174708075735095

The final weights Theta\_1 is [[-2.27984016 5.47958173 5.4853568 ]

[ 6.06207454 -4.05734265 -4.05860811]]

The final weights Theta\_2 is [[-8.85945217 6.16729371 6.06195859]]

The final error is 0.09996820523917618

**the total number of batches run is 6019**

**Learning rate is: 0.3**

**Target error is: 0.1**

The first batch error is 0.7174708075735095

The final weights Theta\_1 is [[-2.27994594 5.4797204 5.4854824 ]

[ 6.06225125 -4.05745214 -4.05871494]]

The final weights Theta\_2 is [[-8.85998035 6.16759386 6.06237466]]

The final error is 0.09995153130685704

**the total number of batches run is 2008**

**Learning rate is: 0.5**

**Target error is: 0.1**

The first batch error is 0.7174708075735095

The final weights Theta\_1 is [[-2.28032037 5.48028715 5.48603387]

[ 6.06313856 -4.05802736 -4.05928699]]

The final weights Theta\_2 is [[-8.86165711 6.16860239 6.0636132 ]]

The final error is 0.09989264894379402

**the total number of batches run is 1206**

**Learning rate is: 0.75**

**Target error is: 0.1**

The first batch error is 0.7174708075735095

The final weights Theta\_1 is [[-2.28074798 5.4809295 5.48665797]

[ 6.06414795 -4.05868079 -4.05993664]]

The final weights Theta\_2 is [[-8.86358181 6.16975724 6.06503851]]

The final error is 0.09982531382072798

**the total number of batches run is 805**

**Learning rate is: 1**

**Target error is: 0.1**

The first batch error is 0.7174708075735095

The final weights Theta\_1 is [[-2.27965468 5.47914069 5.4848656 ]

[ 6.06116554 -4.05671692 -4.05797233]]

The final weights Theta\_2 is [[-8.85901294 6.16690594 6.06181359]]

The final error is 0.09999633955642356

**the total number of batches run is 604**

We can safely conclude that 1 is the best choice for learning rate because the total number of running times of it is the least.

**Q2 A Web Page**

Just designed as the homework required, but more details. For example, the table will reset itself when the shape, radius or height changes, and the height input textbox will be disabled when the user chooses sphere.