***CIS 412 - ARTIFICIAL INTELLIGENCE***

***Project – 3***

**By Team 10:**

**Vishnu Vardhan Kumar Pallati**

**Nicholas Elliot**

**Balasubramanian Srinivas**

**Part 1: Decision Trees**

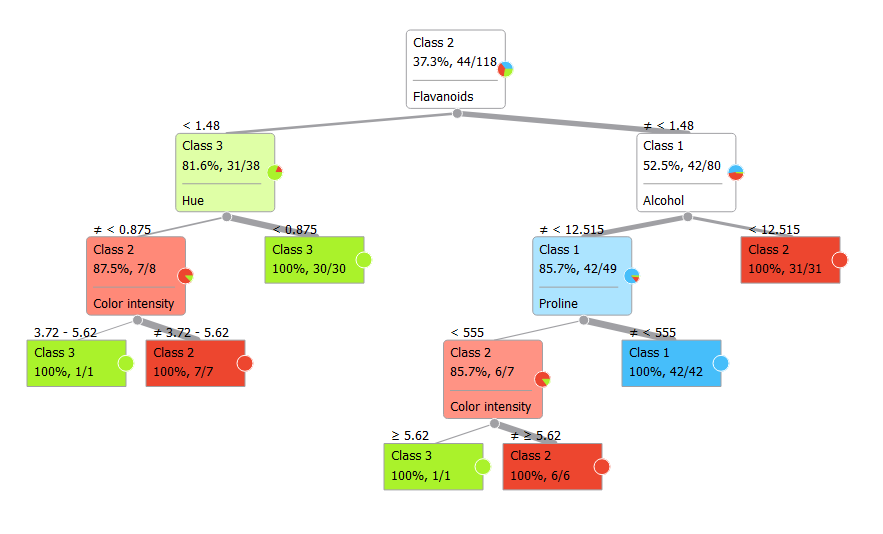
118 instances are taken for the training dataset and 60 instances are considered for the testing set.

Orange Data Mining tool was used to draw the trees and to perform all the data editing.

**Test 1 :** Grouping of data.

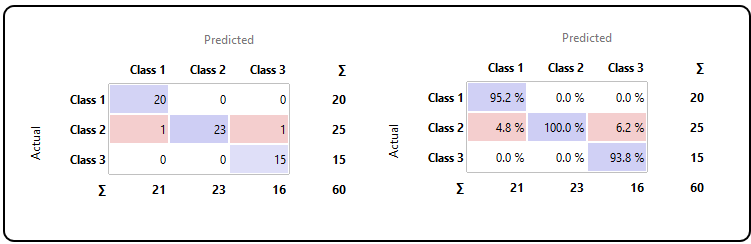
|  |  |  |  |
| --- | --- | --- | --- |
| 1) Alcohol | < 12.515 | 12.515 - 13.485 | ≥ 13.485 |
| 2) Malic acid | < 1.675 | 1.675 - 2.555 | ≥ 2.555 |
| 3) Ash | < 2.275 | 2.275 - 2.47 | ≥ 2.47 |
| 4) Alkalinity of ash | < 18.05 | 18.05 - 20.9 | ≥ 20.9 |
| 5) Magnesium | < 91.5 | 91.5 - 102.5 | ≥ 102.5 |
| 6) Total phenols | < 1.94 | 1.94 - 2.605 | ≥ 2.605 |
| 7) Flavonoids | < 1.48 | 1.48 - 2.645 | ≥ 2.645 |
| 8) Non Flavonoid phenols | < 0.285 | 0.285 - 0.405 | ≥ 0.405 |
| 9) Proanthocyanidins | < 1.355 | 1.355 - 1.84 | ≥ 1.84 |
| 10)Color intensity | < 3.72 | 3.72 - 5.62 | ≥ 5.62 |
| 11)Hue | < 0.875 | 0.875 - 1.065 | ≥ 1.065 |
| 12)OD280/OD315 of diluted wines | < 2.285 | 2.285 - 3.025 | ≥ 3.025 |
| 13)Proline | < 555 | 555 - 832.5 | ≥ 832.5 |

**Screenshot of decision tree:**



Additional Information about formation of the tree such as entropy, gain tables at each levels is present in the ~/AdditionalDocsDecision.docx in the zip folder.

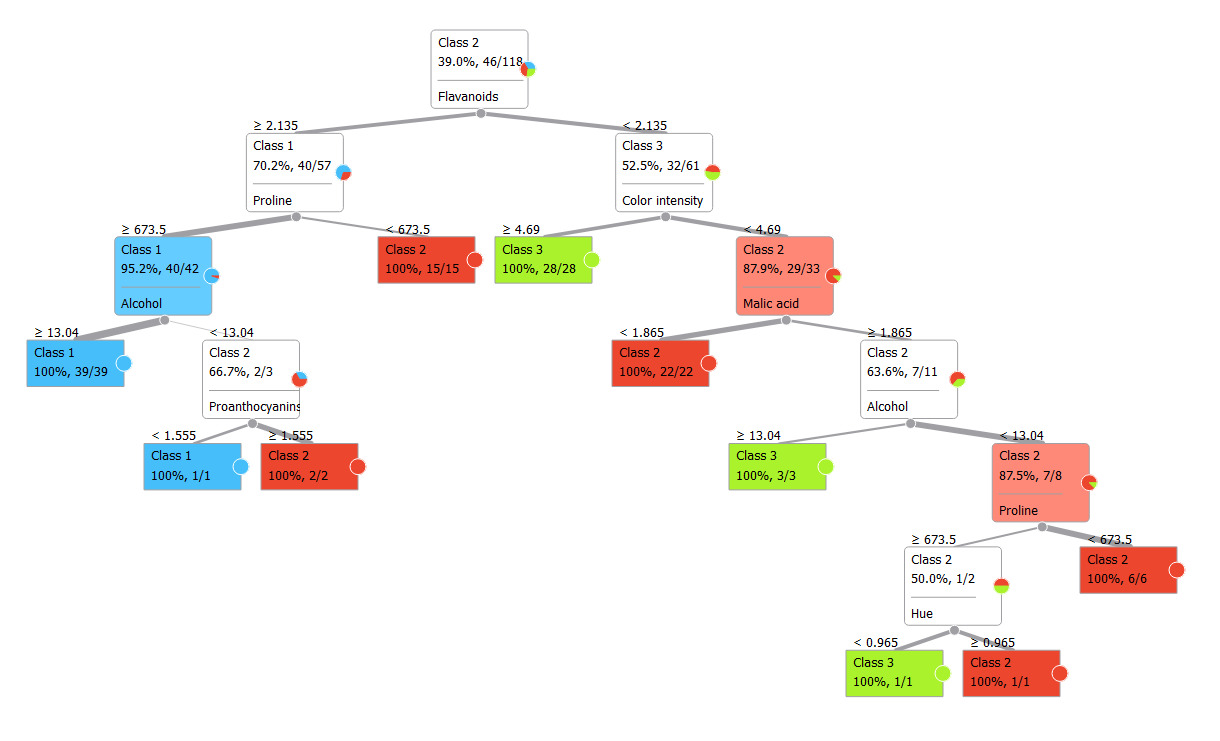
**Confusion Matrix:**



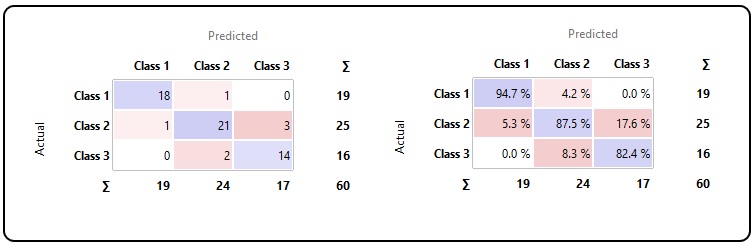
**Test 2 :**

|  |  |  |
| --- | --- | --- |
| 1) Alcohol | < 13.04 | ≥ 13.04 |
| 2) Malic acid | < 1.865 | ≥ 1.865 |
| 3) Ash | < 2.355 | ≥ 2.355 |
| 4) Alkalinity of ash | < 19.45 | ≥ 19.45 |
| 5) Magnesium | < 97.5 | < 97.5 |
| 6) Total phenols | < 2.355 | ≥ 2.35 |
| 7) Flavonoids | < 2.135 | ≥ 2.135 |
| 8) Non Flavonoid phenols | < 0.335 | ≥ 0.335 |
| 9) Proanthocyanidins | < 1.555 | ≥ 1.555 |
| 10)Color intensity | < 4.69 | ≥ 4.69 |
| 11)Hue | < 0.965 | ≥ 0.965 |
| 12)OD280/OD315 of diluted wines | < 2.775 | ≥ 2.775 |
| 13)Proline | < 673.5 | ≥ 673.5 |

Screenshot of the decision tree:



Confusion Matrix:

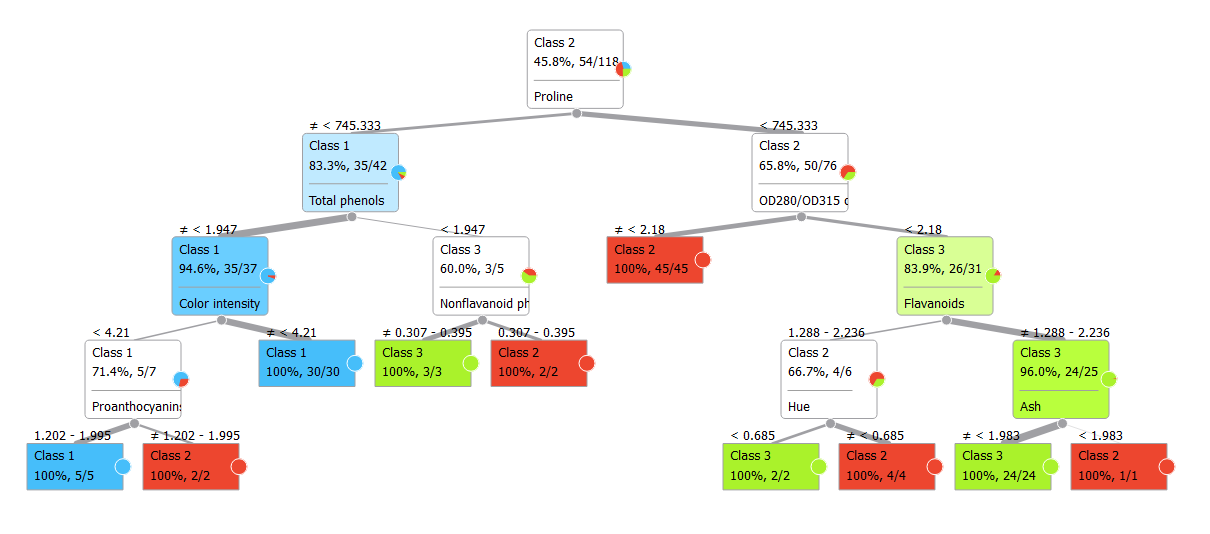


**Test 3:**

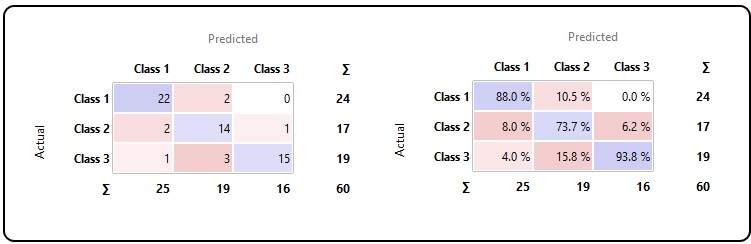
**Grouping of data:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1) Alcohol | < 11.98 | 11.98 - 12.93 | 12.93 - 13.88 | ≥ 13.88 |  |  |
| 2) Malic acid | < 2.43 | 2.43 - 4.11 | ≥ 4.11 |  |  |  |
| 3) Ash | < 1.98 | 1.98 - 2.61 | ≥ 2.61 |  |  |  |
| 4) Alkalinity of ash | < 15.45 | 15.45- 20-30 | 20.30 - 25.15 | ≥ 25.15 |  |  |
| 5) Magnesium | < 85.33 | 85.33 - 100.67 | 100.67 - 116.00 | 116.00 - 131.33 | 131.33 - 146.67 | ≥ 146.67 |
| 6) Total phenols | < 1.95 | 1.95 - 291 | ≥ 2.91 |  |  |  |
| 7) Flavonoids | < 1.29 | 1.29 - 2.24 | 2.24 - 3.18 | 3.18 - 4.13 | ≥ 4.13 |  |
| 8) Non Flavonoid phenols | < 0.22 | 0.22 - 0.31 | 0.31 - 0.40 | 0.40 - 0.48 | 0.48 - 0.57 | ≥ 0.57 |
| 9) Proanthocyanidins | < 1.20 | 1.20 - 1.99 | 1.99 - 2.79 | ≥ 2.79 |  |  |
| 10)Color intensity | < 4.21 | 4.21 - 7.14 | 7.14 - 10.07 | ≥ 10.07 |  |  |
| 11)Hue | < 0.68 | 0.68 - 0.89 | 0.89 - 1.09 | 1.09 - 1.30 | 1.30 - 1.50 | ≥ 1.50 |
| 12)OD280/OD315 of diluted wines | < 2.18 | 2.18 - 3.09 | ≥ 3.09 |  |  |  |
| 13)Proline | < 745.33 | 745.33 - 1212.67 | ≥ 1212.67 |  |  |  |

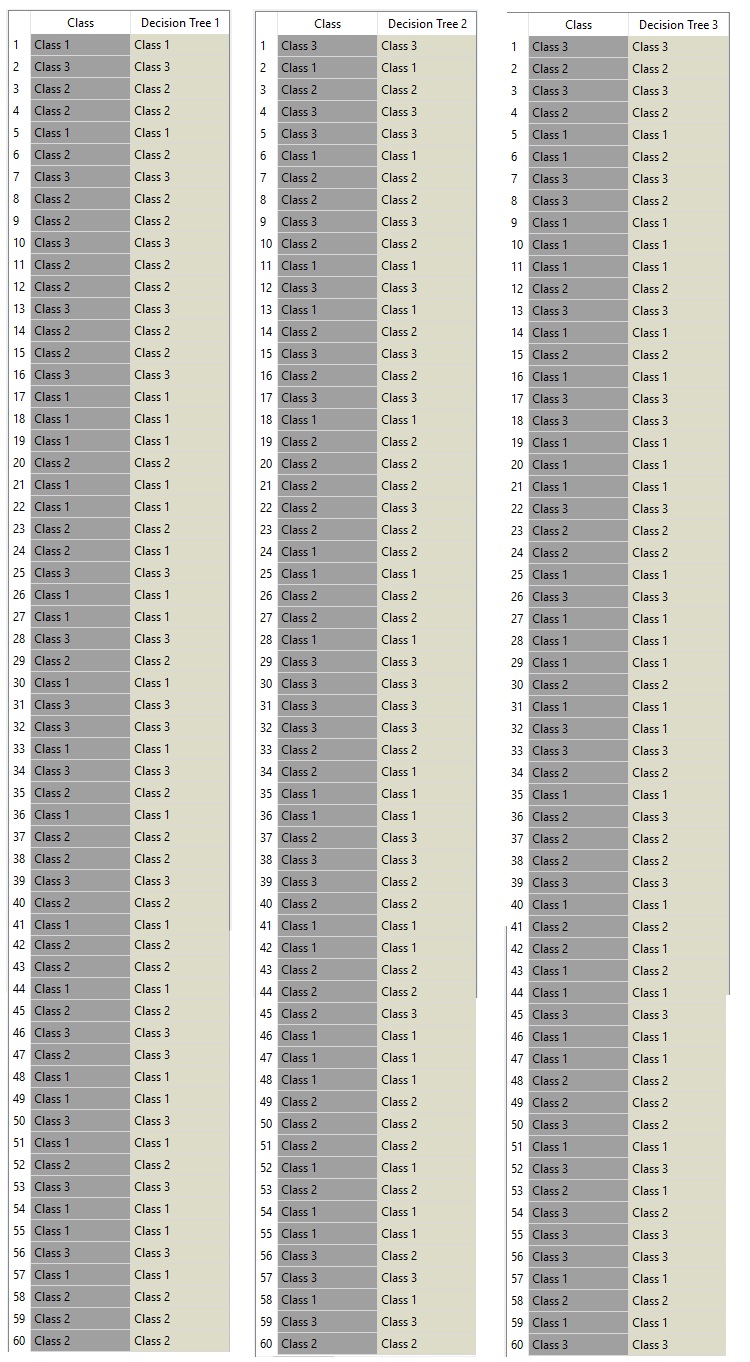
**Screenshot of the tree:**



**Confusion Matrix:**



**3) Correct classification percentage for the decision trees :**



Decision Tree 1: 2 Errors.

* 99% Accuracy .

Decision Tree 2 : 7 Errors

* 96% Accuracy .

Decision Tree 3 : 9 Errors

* 95% Accuracy .

**4) Comparative analysis of the classification rate of the Decision Tree:**

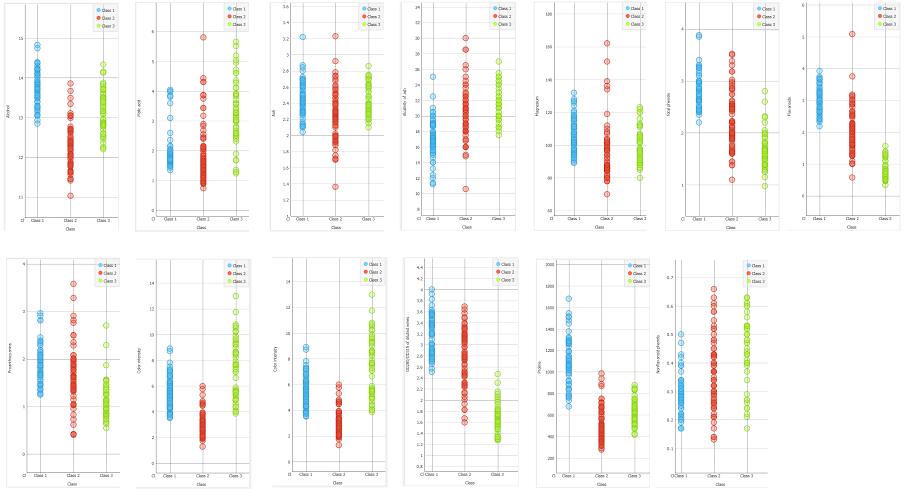
Firstly it is worth noting that the learning data for our decision trees have 100% classification rate. All of the errors were within the test data.

Our first decision tree discrete split the data three ways, with the ranges of all variables being evenly split so that all new ranges were of equal width. This seemed to have the best result of all three trees at a glance for classification as it was extremely accurate at 99%.

Our second decision tree discrete split the ranges of all variables evenly in half, meaning that the new ranges were both of equal width. This result proved still fairly accurate at 96%, still lower than our first decision tree but quite accurate

For our third discrete split of the ranges, we chose our own method. Looking at data graphs of each range of data per variable that we generated (as shown after the next paragraph), we were able to come up with custom ranges that we felt might assist in the tree classifying the data correctly. For example, looking at the alcohol data plot, we chose the ranges <11.98 since very little of the data fell below that line, 11.98-12.93 since the majority of data for class two and roughly half the data for class 3 lies in that range, 12.93-13.88 since the majority of class 1 and roughly half of the data class 3 have data within that range, and lastly anything greater than or equal to 13.88 since very little of the data was within this range and could be seen as outliers. This method proved to be less effective since our accuracy was 95% for decision tree 3, which was lower than both other decision trees.

Comparing the three, it is obvious that decision tree 1 has the highest accuracy of the three and is the best for this particular scenario. Splitting the ranges evenly into three parts allows the tree to most accurately classify the data with the least amount of errors. Decision tree two was also quite accurate but fell slightly short of decision tree 1, and decision tree three while designed by us in order to improve classification rate was the least accurate in classifying data, even if it seemed like an intelligent choice for the tree to make. We believe that this could be because of data points that are very close to other classes rather than their own, thus throwing off the trees from correct classification.

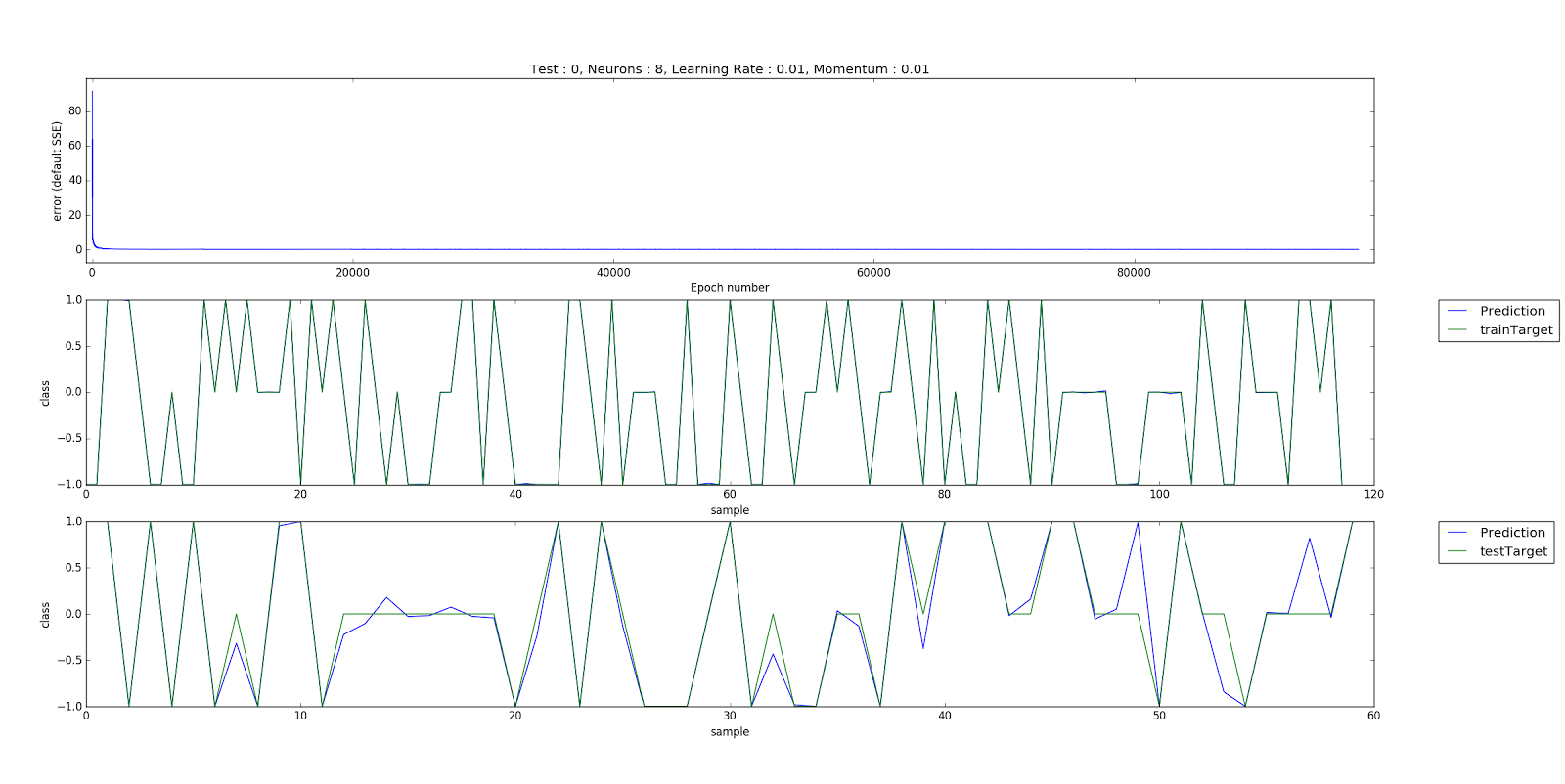


**Neural Networks :**

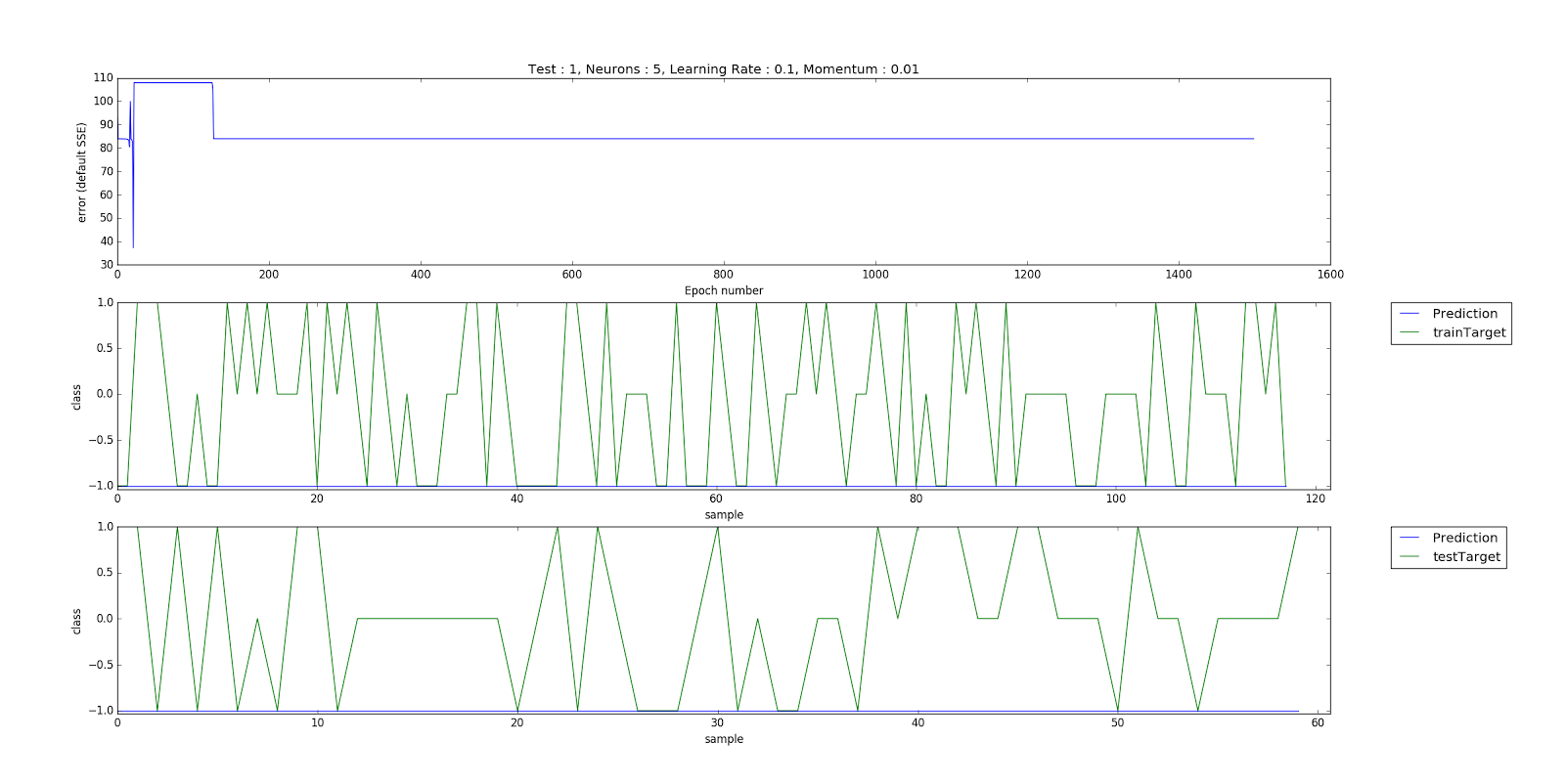
The code for Neural Networks is present in NN.py python file.

**5) Error and epochs for all different conditions**.

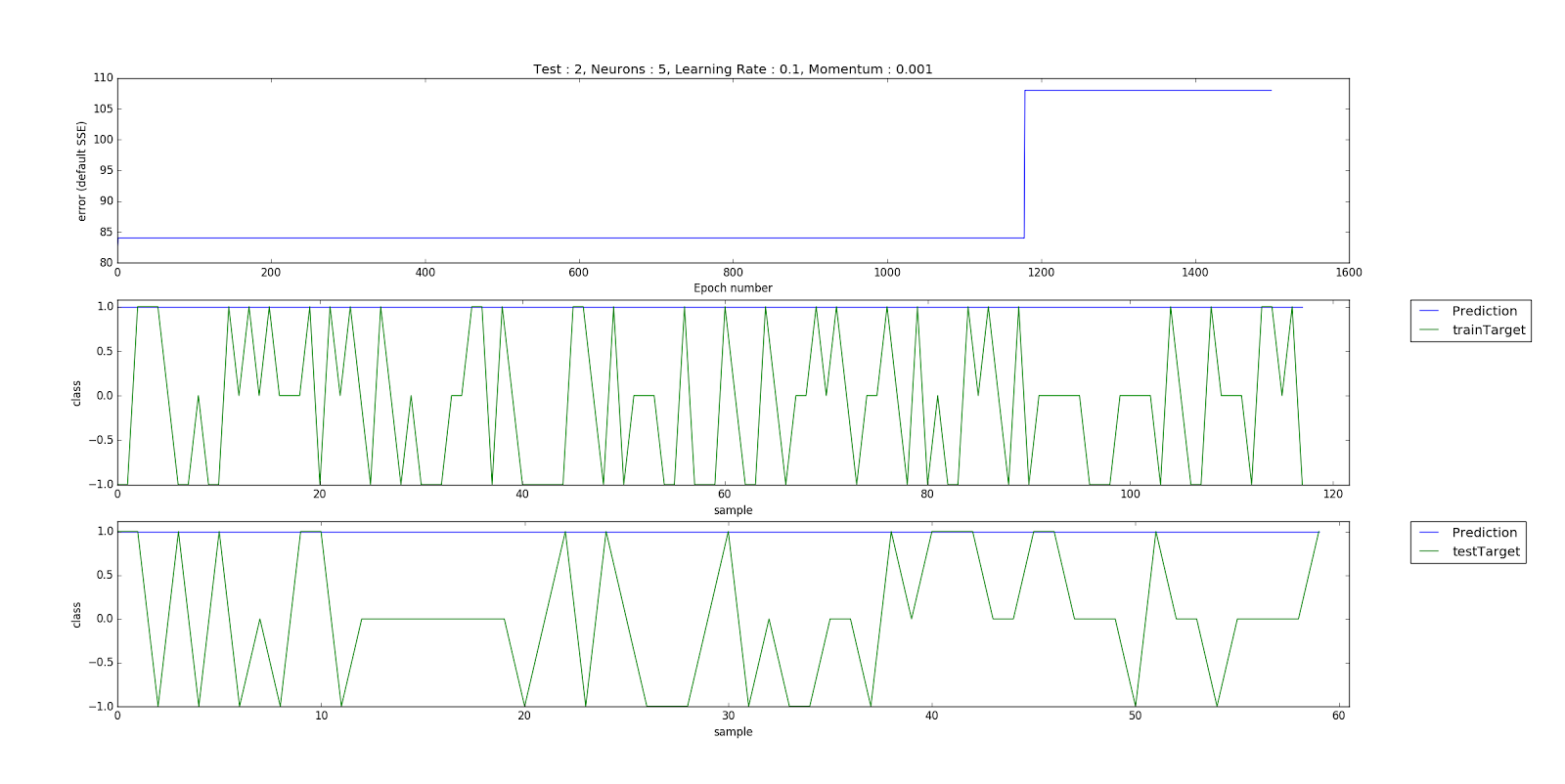
Test 0 :



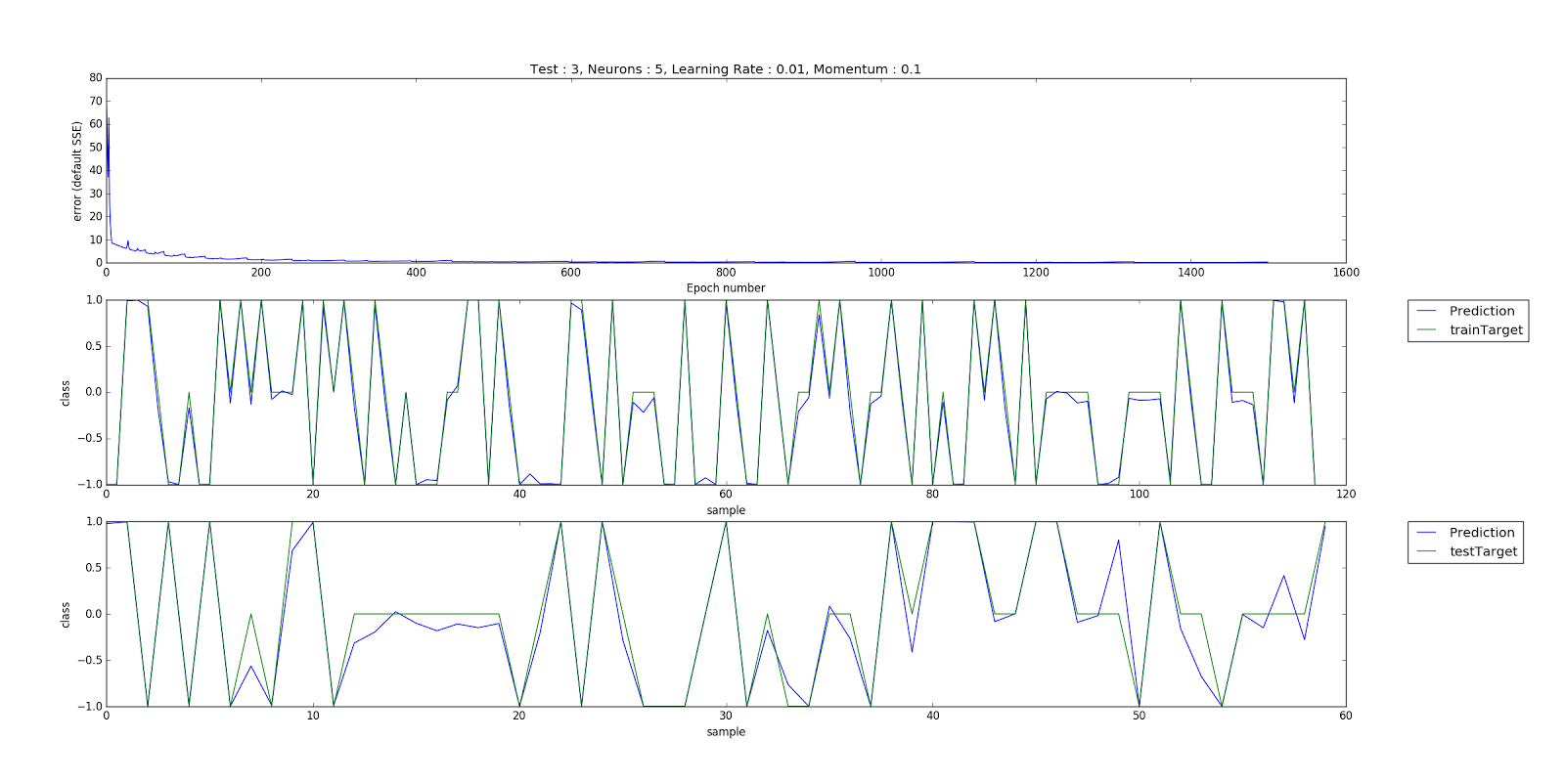
Test 1 :



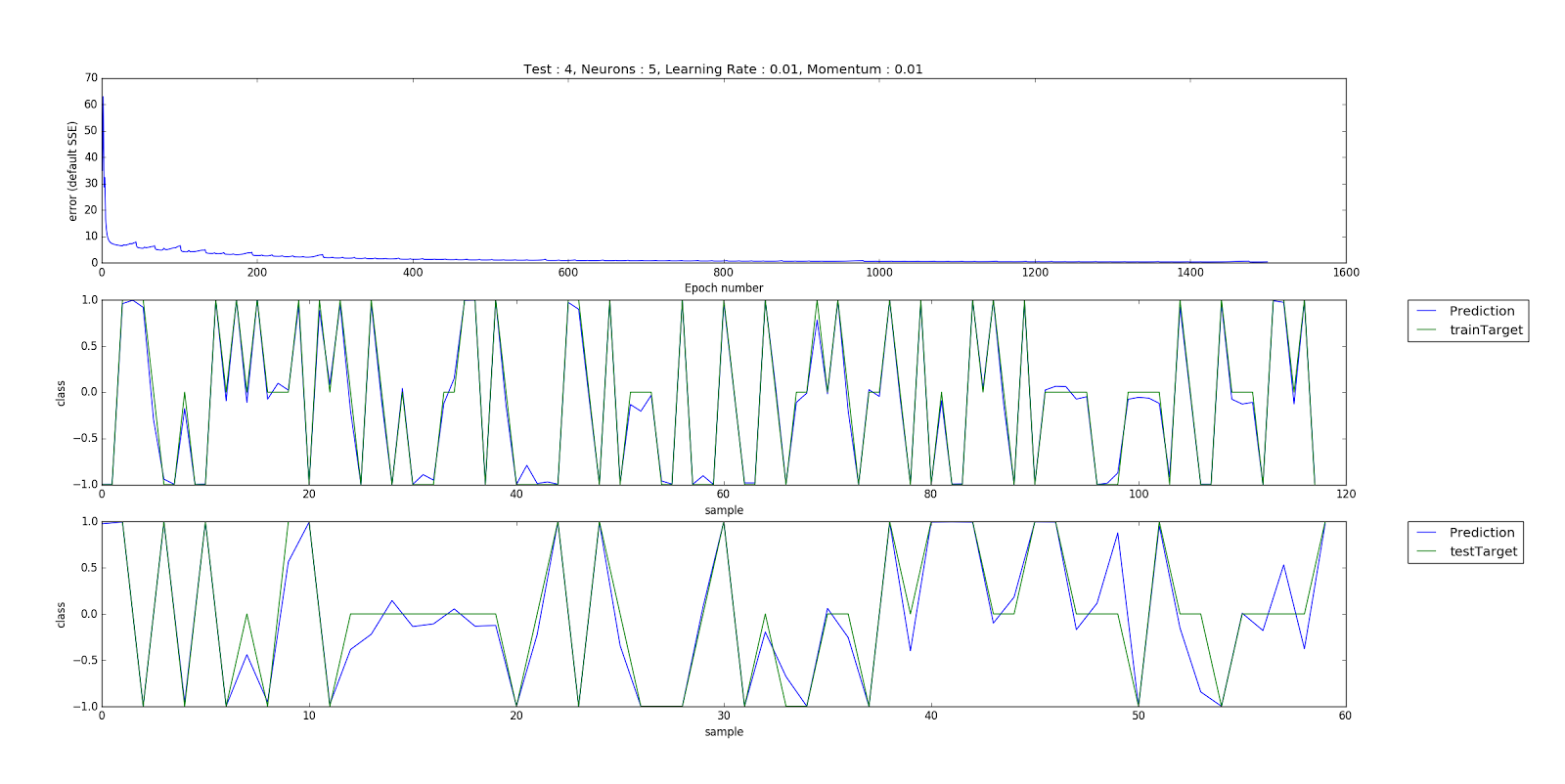
Test 2 :



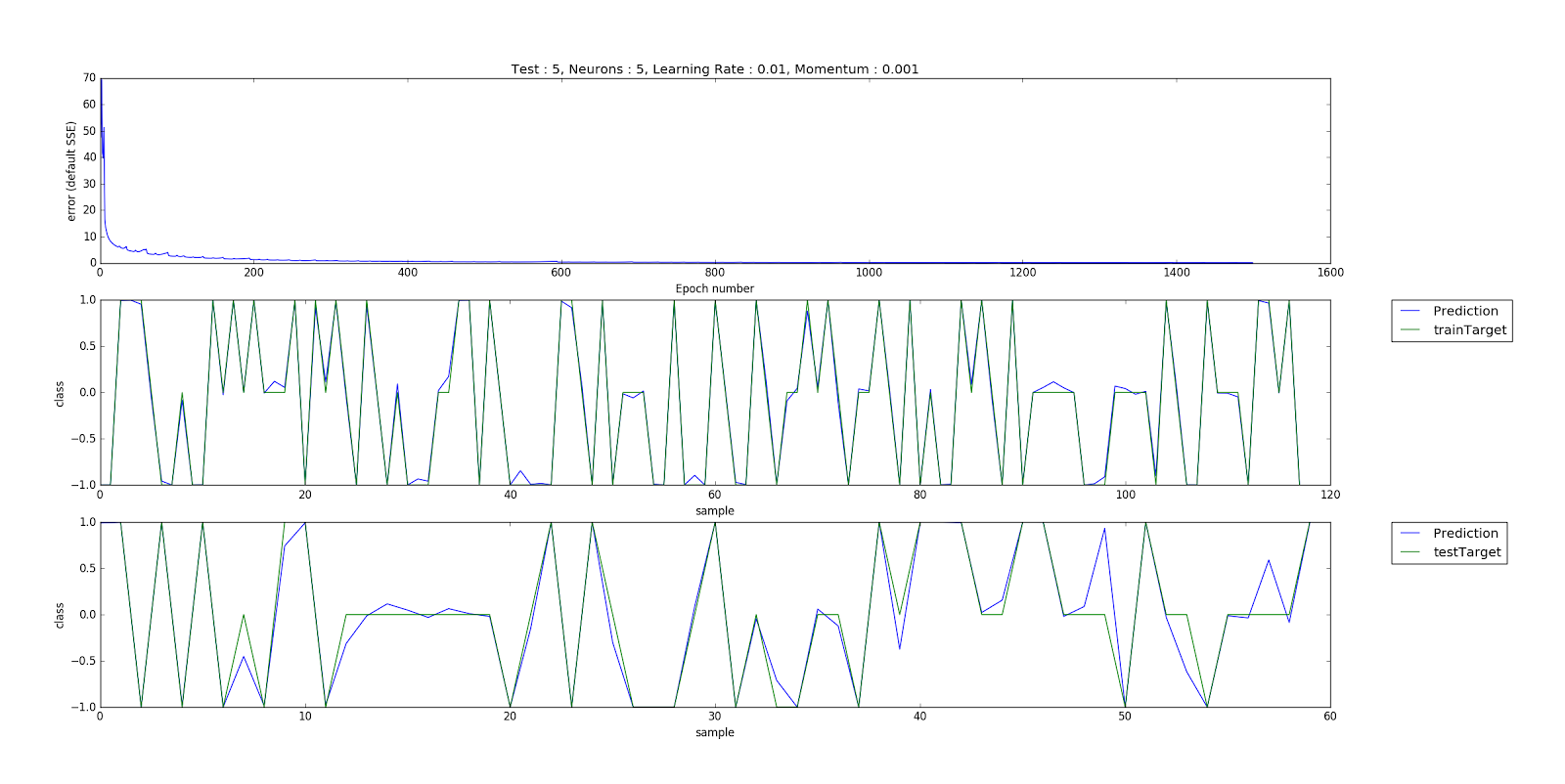
Test 3 :



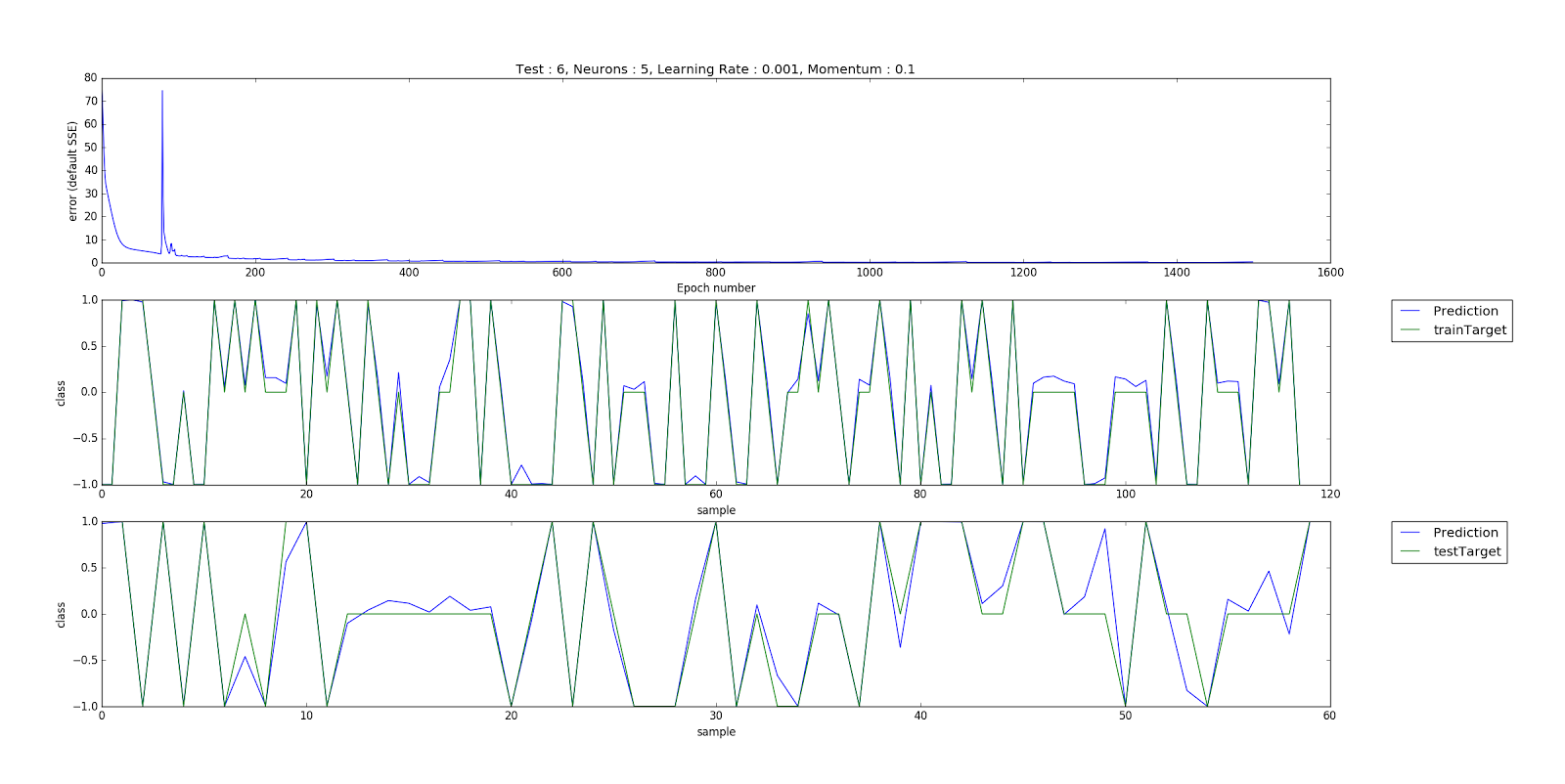
Test 4 :

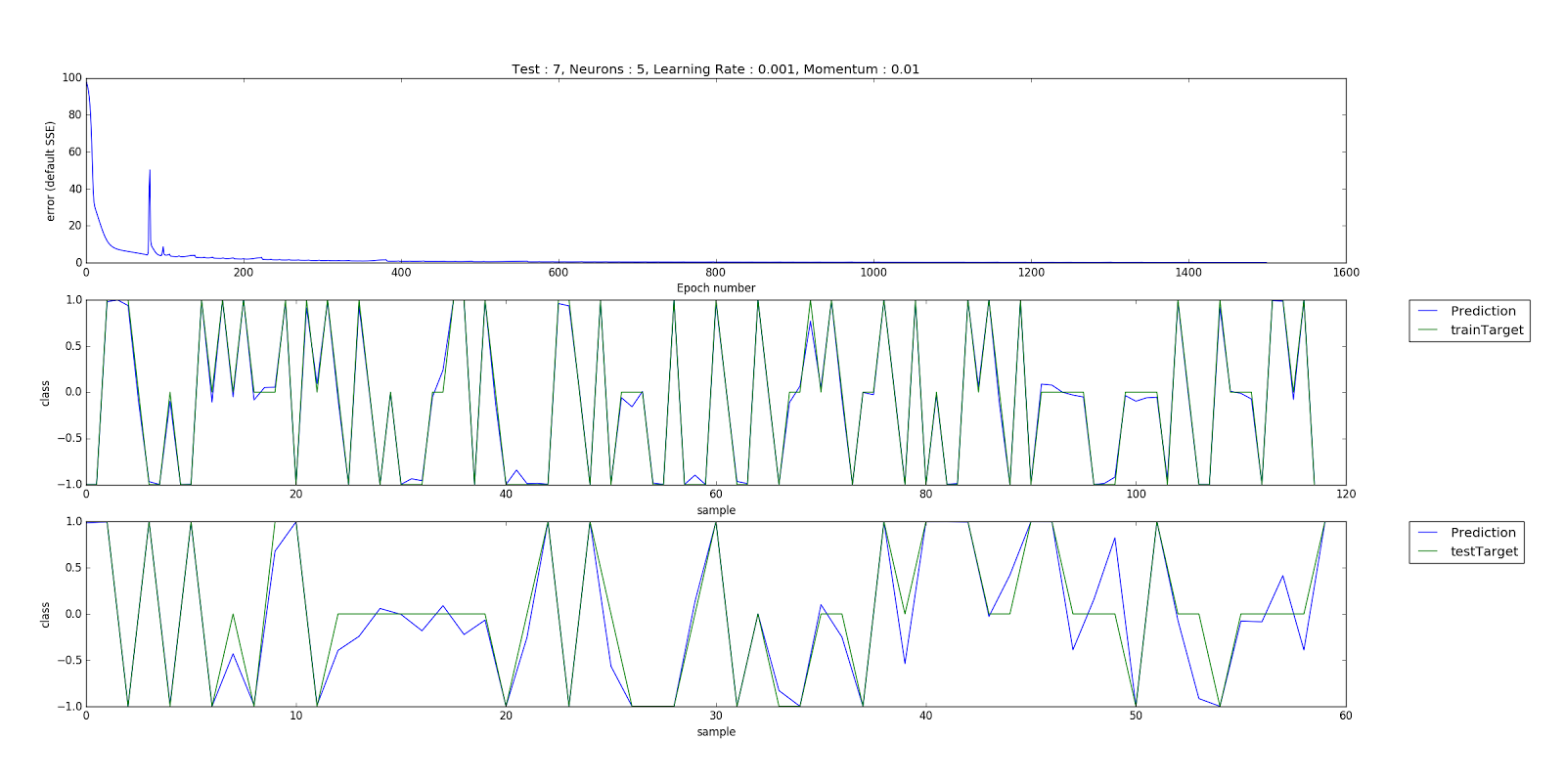


Test 5 :

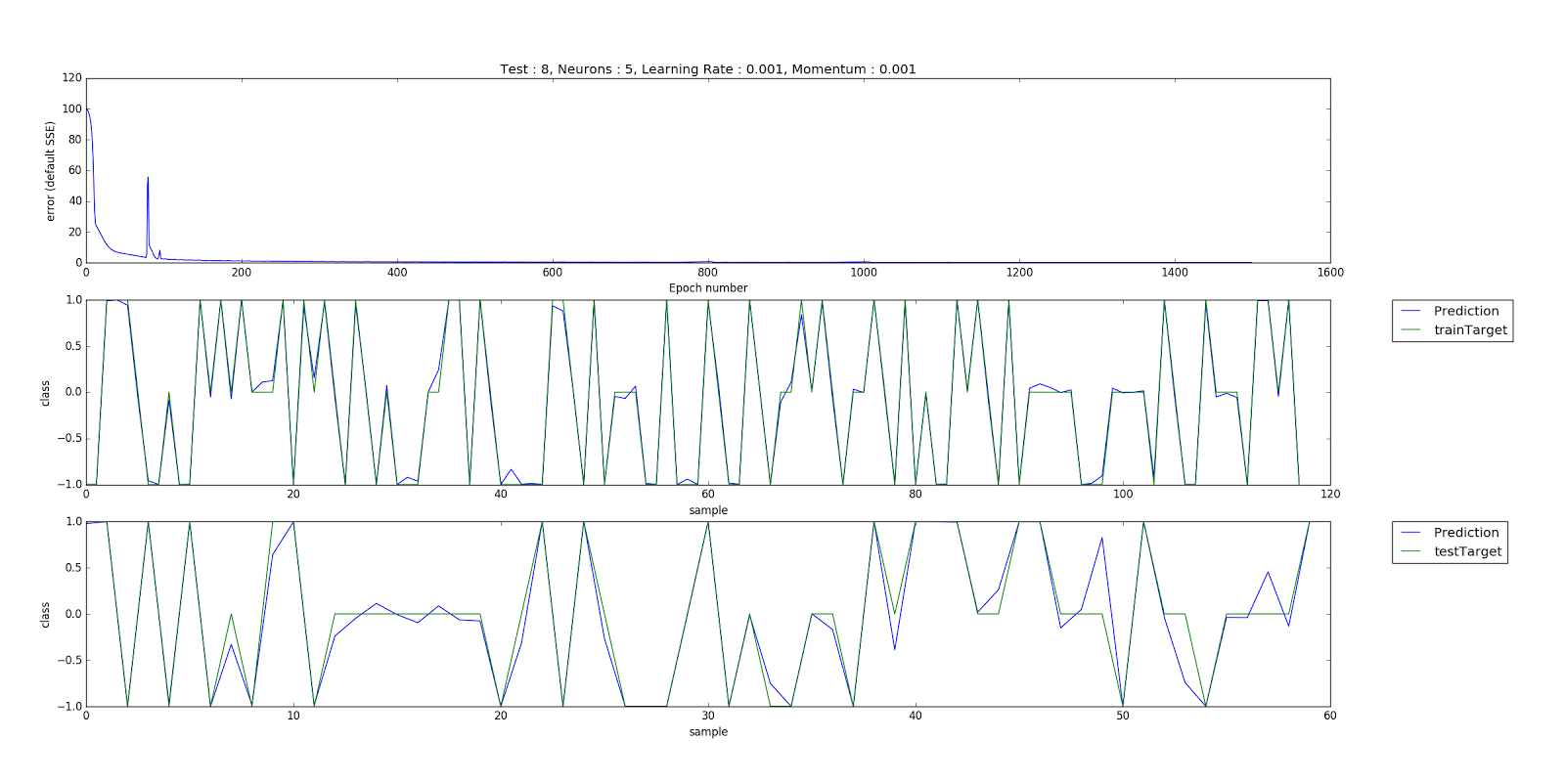


Test 6 :

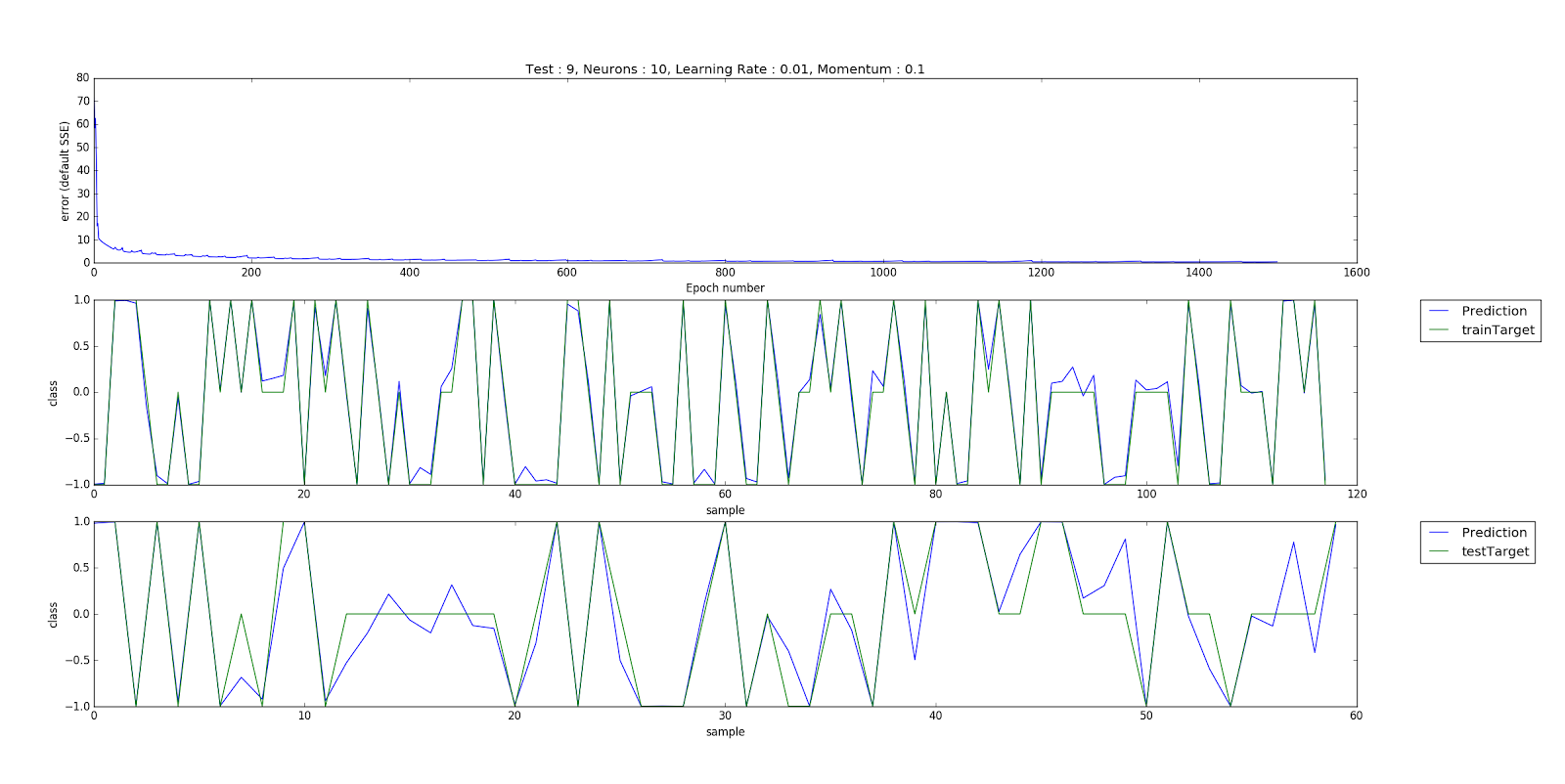


Test7 : 

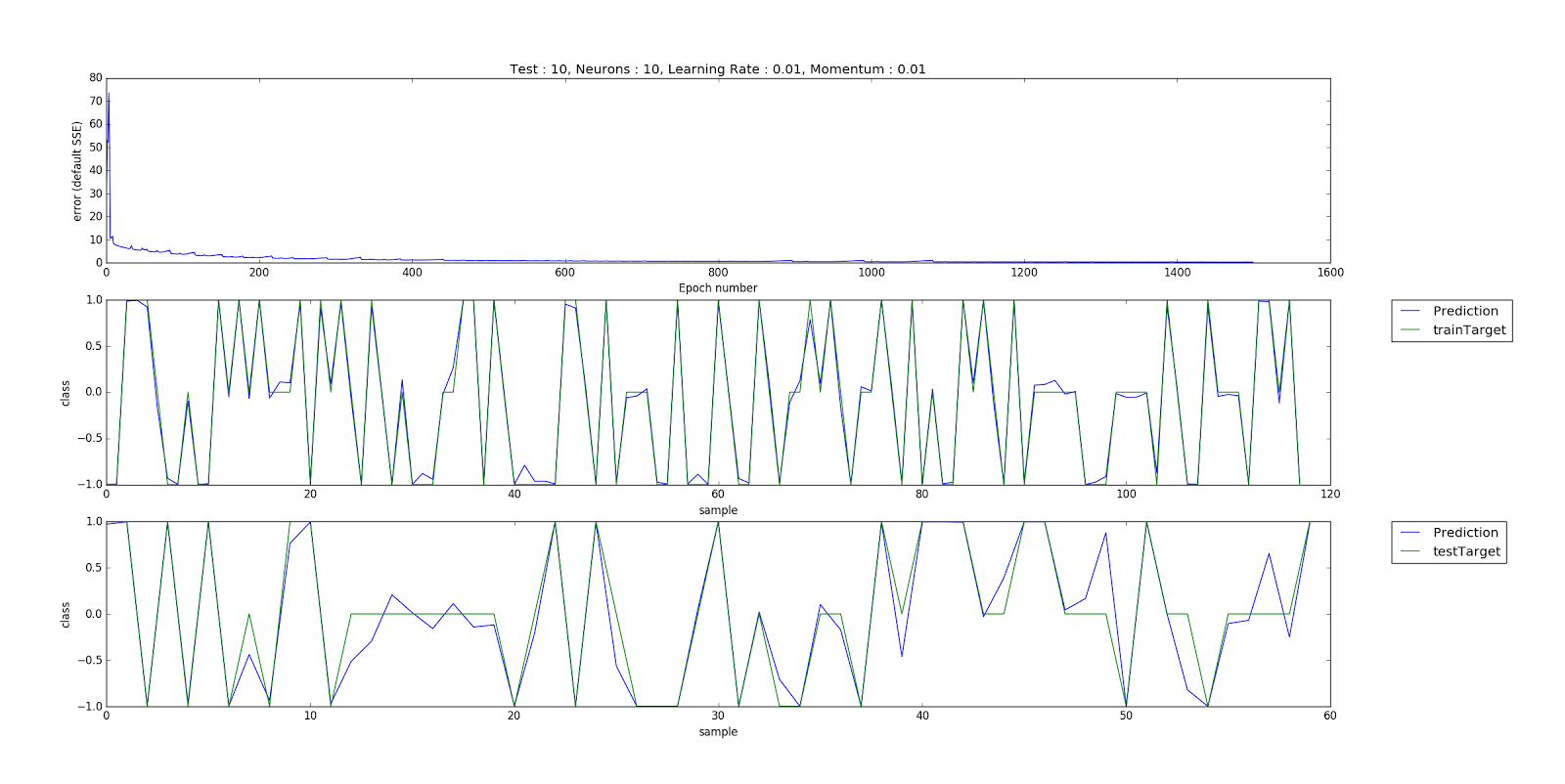
Test 8 :



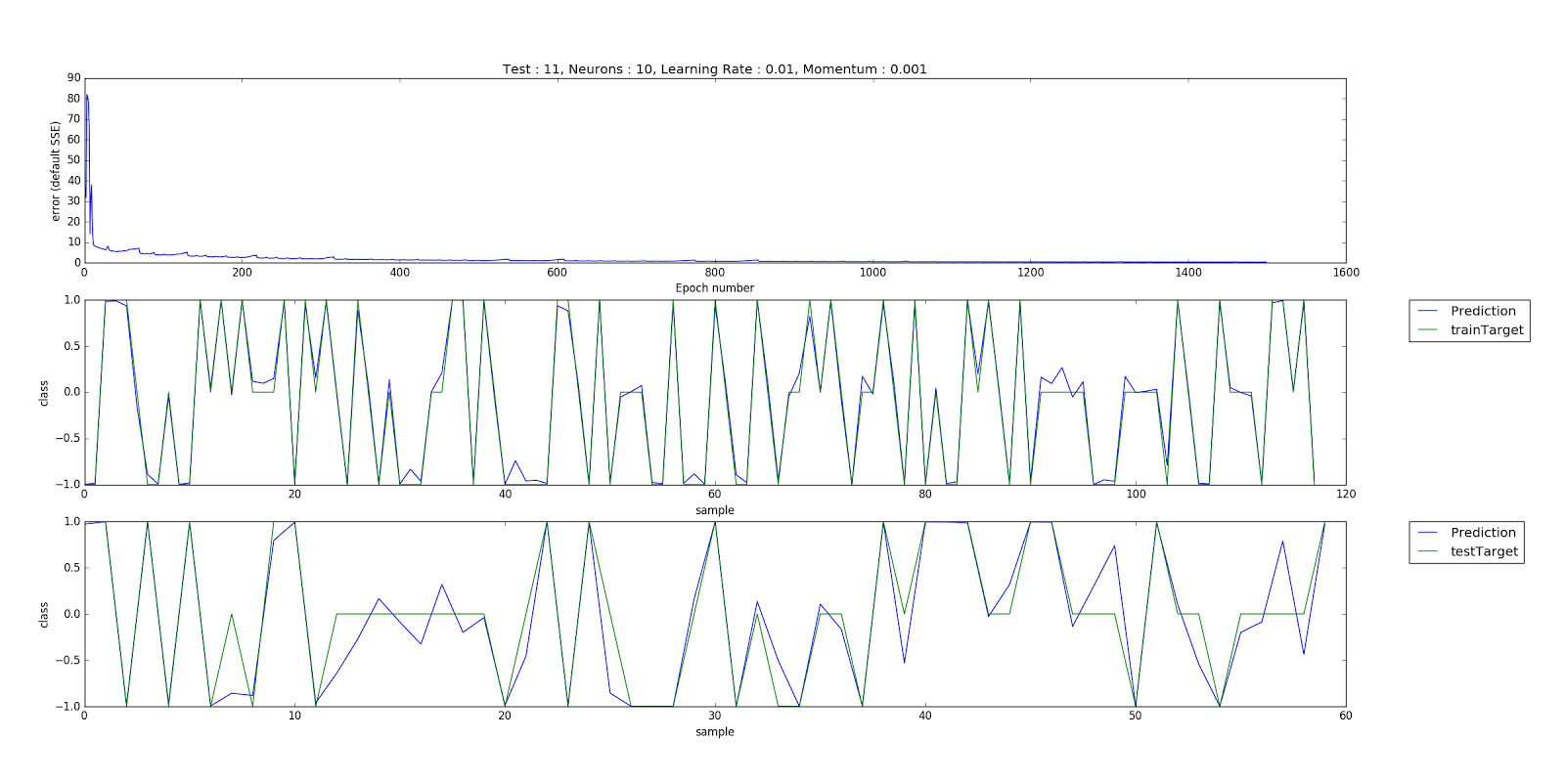
Test 9 :



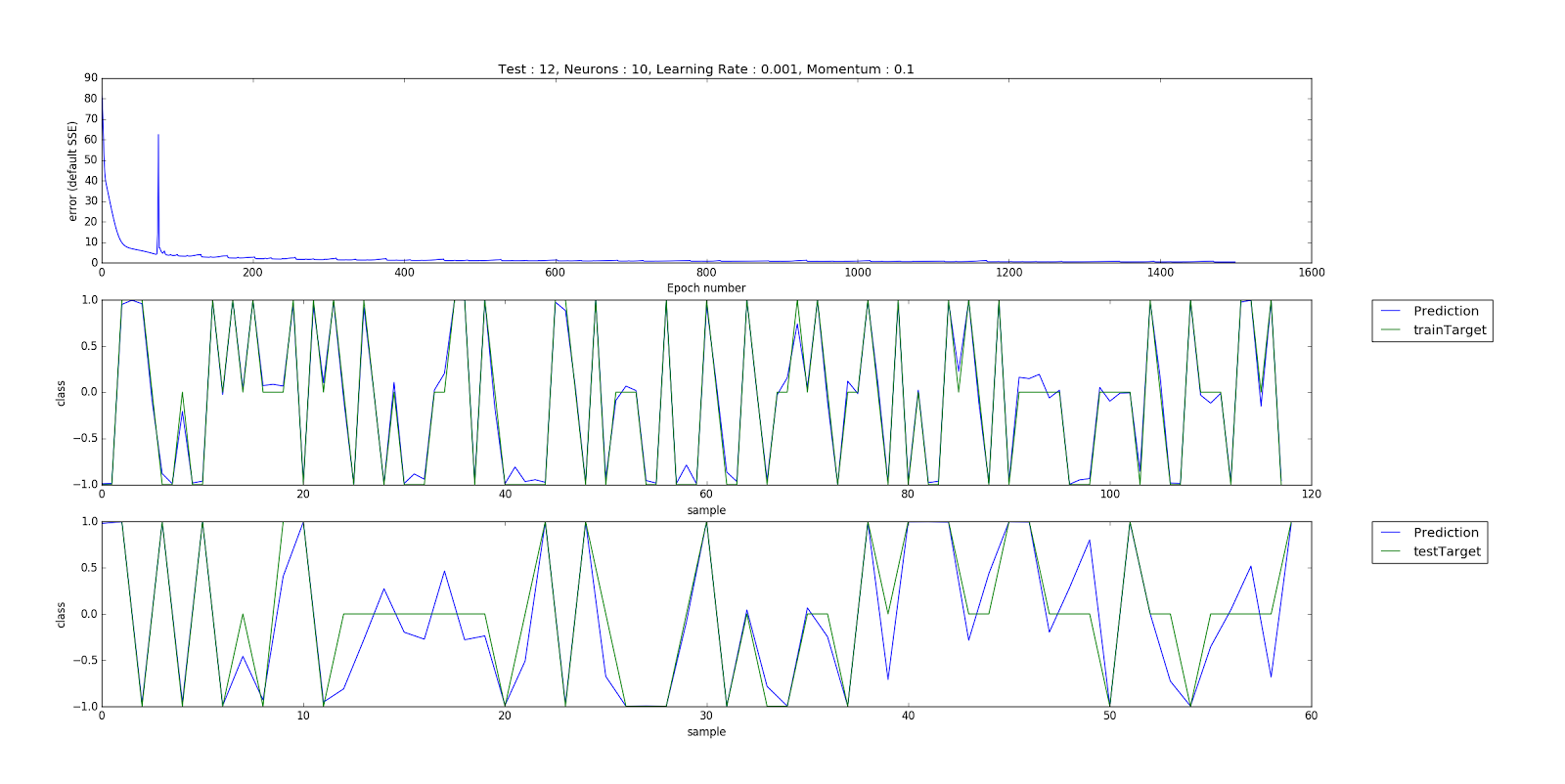
Test 10 :



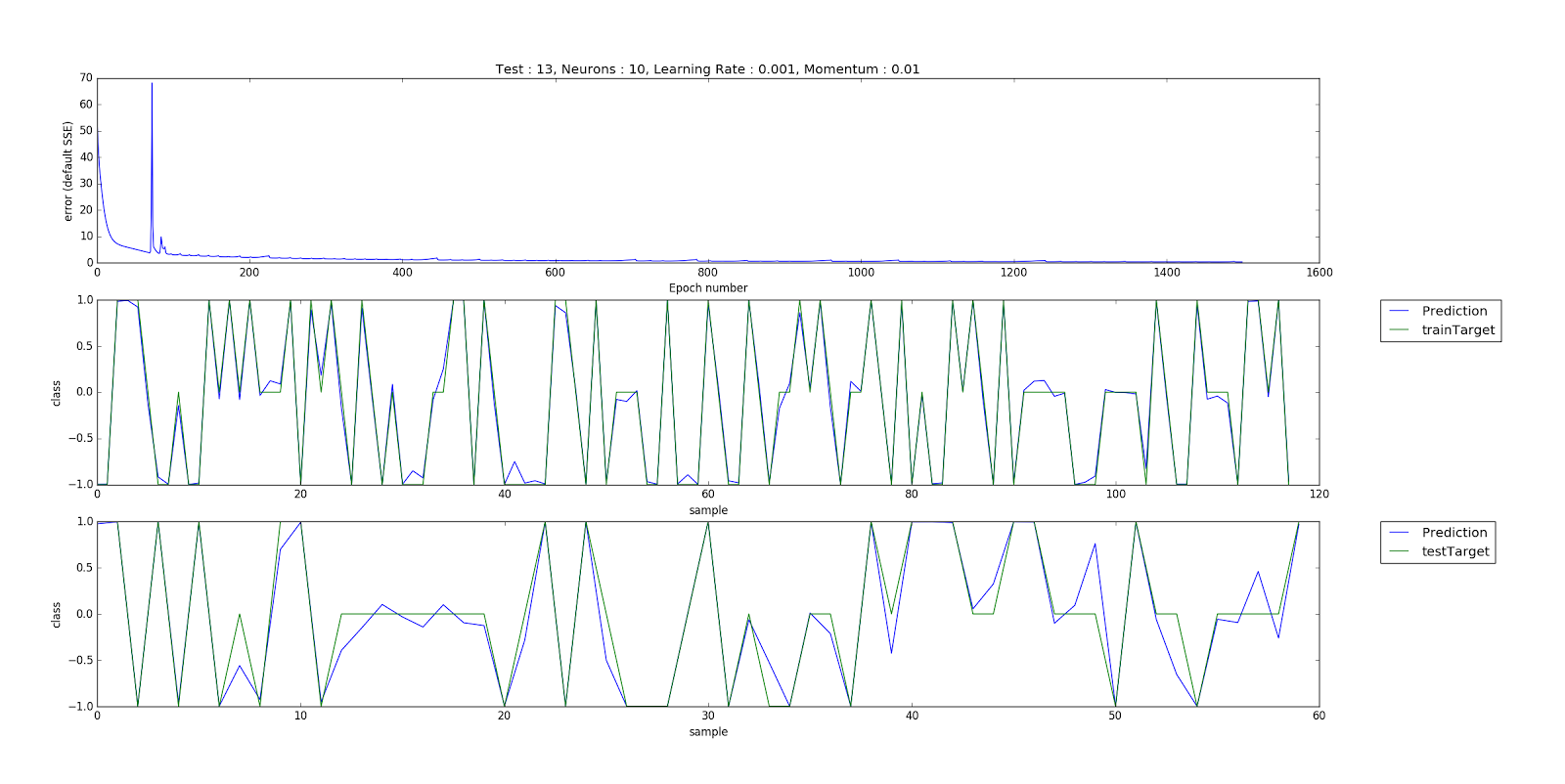
Test 11 :



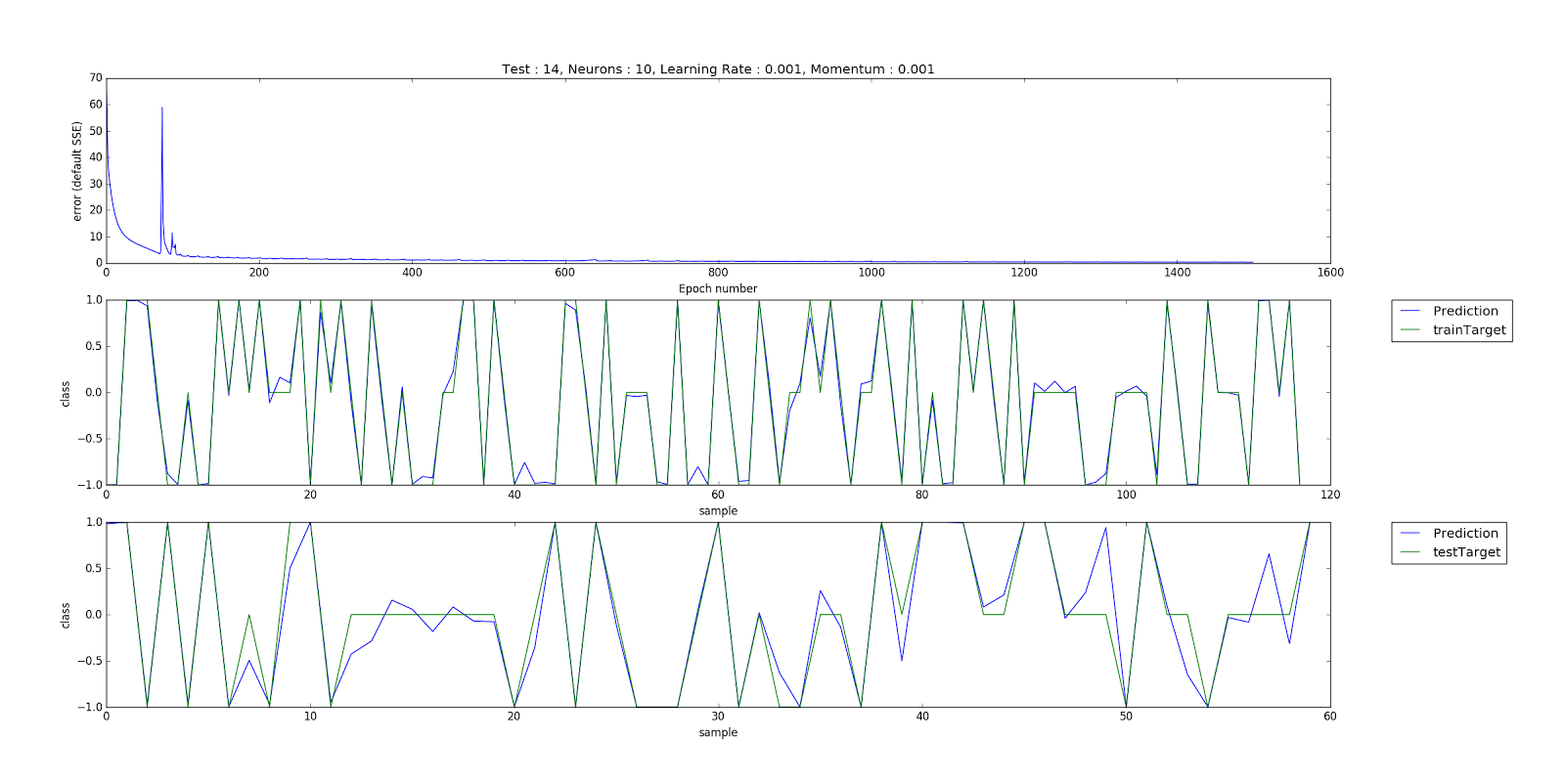
Test 12 :



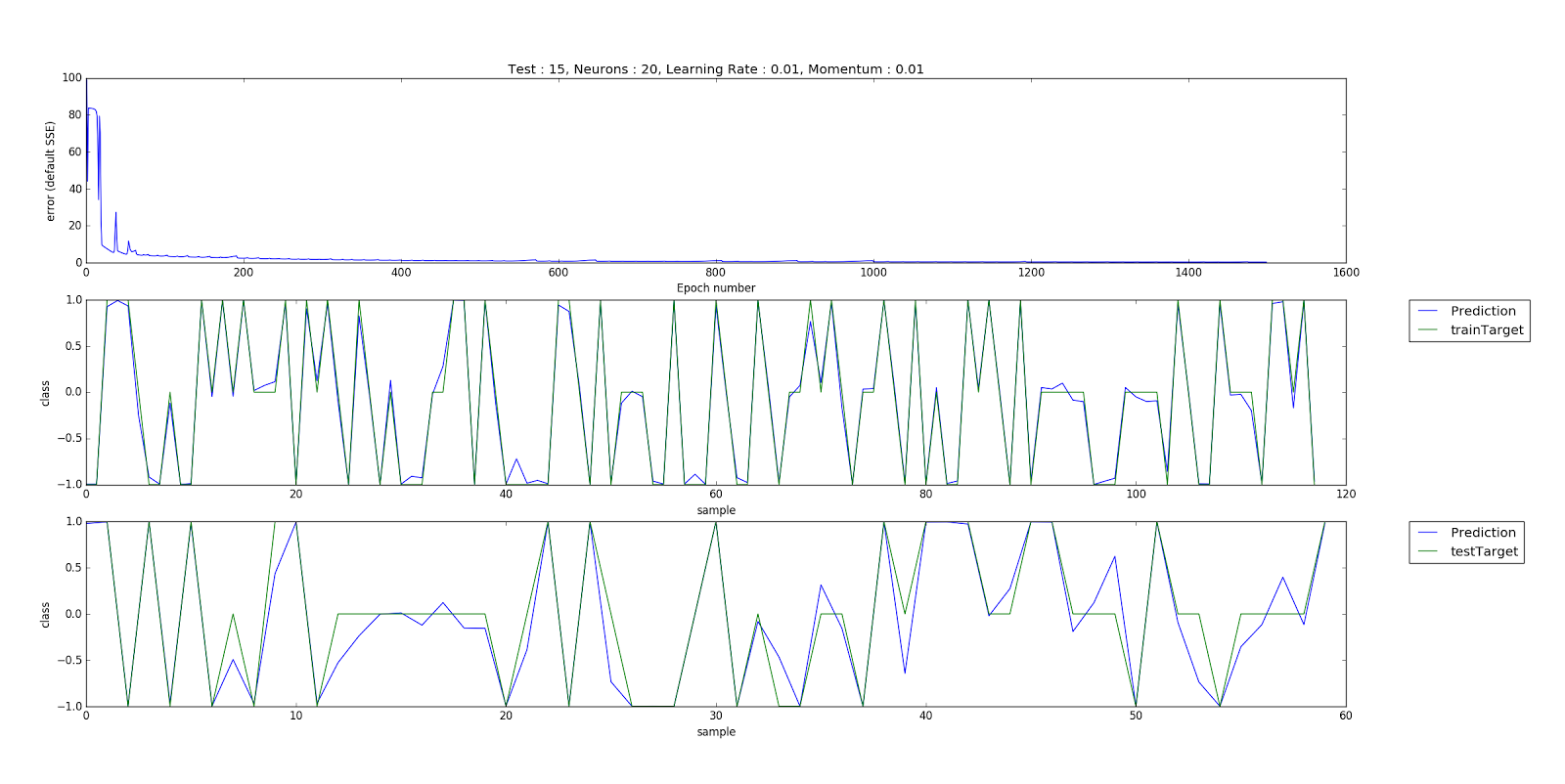
Test 13 :



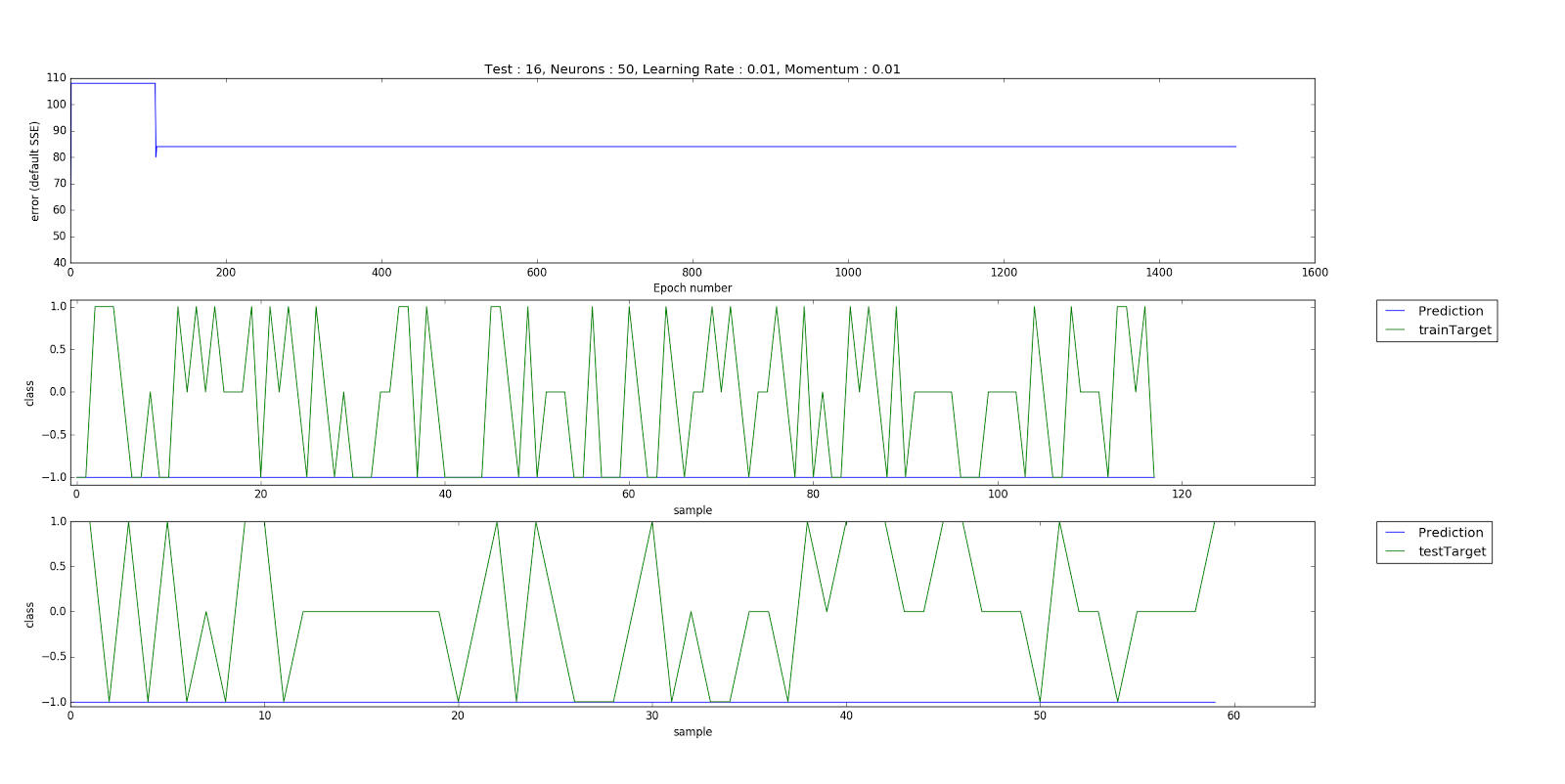
Test 14 :



Test 15 :



Test 16 :



**6) Correct Classification rate for all different condition:**

1. Test 0: 98%
   1. Neurons: 8, learning rate: 0.01, momentum: 0.01
2. Test 1: 35%
   1. Neurons: 5, learning rate: 0.1, momentum: 0.01
3. Test 2: 24%
   1. Neurons: 5, learning rate: 0.1, momentum: 0.001
4. Test 3: 98%
   1. Neurons: 5, learning rate: 0.01, momentum: 0.1
5. Test 4: 97%
   1. Neurons: 5, learning rate: 0.01, momentum: 0.01
6. Test 5: 98%
   1. Neurons: 5, learning rate: 0.01, momentum: 0.001
7. Test 6: 98%
   1. Neurons: 5, learning rate: 0.001, momentum: 0.1
8. Test 7: 97%
   1. Neurons: 5, learning rate: 0.001, momentum: 0.01
9. Test 8: 98%
   1. Neurons: 5, learning rate: 0.001, momentum: 0.001
10. Test 9: 96%
    1. Neurons: 10, learning rate: 0.01, momentum: 0.1
11. Test 10: 96%
    1. Neurons: 10, learning rate: 0.01, momentum: 0.01
12. Test 11: 95%
    1. Neurons: 10, learning rate: 0.01, momentum: 0.001
13. Test 12: 94%
    1. Neurons: 10, learning rate: 0.001, momentum: 0.1
14. Test 13: 96%
    1. Neurons: 10, learning rate: 0.001, momentum: 0.01
15. Test 14: 97%
    1. Neurons: 10, learning rate: 0.001, momentum: 0.001
16. Test 15: 96%
    1. Neurons: 20, learning rate: 0.01, momentum: 0.01
17. Test 16: 35%
    1. Neurons: 50, learning rate: 0.01, momentum: 0.01

**7) Comparative Analysis of classification rates of NN**

From the data we gathered from our tests we found that a learning rate of 0.1 was insufficient for the network to be able to properly classify the data within the correct output value (1, 2, or 3). We also found from our data that the more neurons we added to the network layer, the less accurate in classifying them it became. We noticed that changing the momentum gave a less significant change than changing the learning rate. While having our learning rate be 0.01 and our momentum at 0.01 appeared to be best, this may not be when the network “learns” the most. Some of the data obviously falls between categories and with more neurons and more drastic changes to momentum we noticed that more of the data was falling between categories with the Neural Network, meaning that, to a point, adding more neurons allows it to classify things more accurately in the sense that data with attributes between outputs will be placed between them as opposed to one or the other. In the sense of our problem, we had the most accurate results with 8 neurons, learning rate of 0.01, and momentum of 0.01