Hrishikesh Hippalgaonkar

[hippalga@usc.edu](mailto:hippalga@usc.edu)

**EE 559 Homework Week 6**

1. **Result**

|  |
| --- |
| **C:\Users\Hrishikesh\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Weight1.jpg** |
| 1. Synthetic dataset 1 |
| **C:\Users\Hrishikesh\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Weight2.jpg** |
| 1. Synthetic dataset 2 |
| **C:\Users\Hrishikesh\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Weight3.jpg** |
| 1. Synthetic dataset 3 |
| **Figure 1. Weights, Training Error and Testing Error** |

|  |  |
| --- | --- |
| C:\Users\Hrishikesh\AppData\Local\Microsoft\Windows\INetCache\Content.Word\training1.png | C:\Users\Hrishikesh\AppData\Local\Microsoft\Windows\INetCache\Content.Word\testing1.png |
| 1. Synthetic dataset 1 (Training) | 1. Synthetic dataset 1 (Testing) |
| C:\Users\Hrishikesh\AppData\Local\Microsoft\Windows\INetCache\Content.Word\training2.png | C:\Users\Hrishikesh\AppData\Local\Microsoft\Windows\INetCache\Content.Word\testing2.png |
| 1. Synthetic dataset 2 (Training) | 1. Synthetic dataset 2 (Testing) |
| C:\Users\Hrishikesh\AppData\Local\Microsoft\Windows\INetCache\Content.Word\training3.png | C:\Users\Hrishikesh\AppData\Local\Microsoft\Windows\INetCache\Content.Word\testing3.png |
| 1. Synthetic dataset 3 (Training) | 1. Synthetic dataset 3 (Testing) |
| **Figure 2. Decision Boundary and decision regions** | |

The training and testing error rate for Synthetic dataset 1 using MDTCM was 21% and 24%, and for synthetic dataset 2, the error rates were 3% and 4%. We can observe that these error rates are significantly greater than the error rates observed in the Perceptron approach.

It can also be observed that the MDTCM was not able to provide a good decision boundary for synthetic dataset 1, which resulted in the high error rate. On the other hand, the Perceptron approach gave a very good decision boundary and thus, a very low error rate.

If the data is linearly separable, MDTCM may or may not be able to separate the data points, but the Perceptron will definitely separate the data points correctly.

Unlike Perceptron, MDTCM is not a learning algorithm. MDTCM uses just the distance from class means to classify the data. Even if any of the data points are misclassified, the MDTCM has no way to learn from its error. On the other hand, Perceptron tries to achieve the lowest error possible by iteratively applying a “learning rule”, thus rectifying most of its error.