

Question. Using your own words, describe the binary classifier Perceptron and provide its pseudocode.

The binary classifier Perceptron consists of a trainer and a tester.

The Perceptron trainer reads the data from the input file, shuffles the order of the data and then computes the activation. If the activation is incorrect, the trainer updates the weights. This operation is repeated until optimum weights are found. Then they are saved in a file so that the Perceptron tester could access them.

As for the Perceptron tester, using the weights that were calculated by the Perceptron trainer and the unlabeled data from an input file, it calculates the activation and then classifies the data instance according to the result.

Perceptron Pseudocode

PerceptronTrain(Data)

$W_{\text{Data}} \leftarrow 0$, for all $d = 1 \dots \text{Data}$

learningRate $\leftarrow 0.05$

Epochs $\leftarrow 0$

allowedMisses $\leftarrow 0$ *//for stopping the updating process if there is no improvement.*

While (True) **do**

 numOfMisses $\leftarrow 0$

for (x,y) **in** Data **do** *//x = input, y = output.*

 Act $\leftarrow \sum_{d=1}^{\text{Data}} w_d x_d$

If ya ≤ 0 **then** *//if misclassified.*

$w_d \leftarrow w_d + x_d * \text{learningRate}$, for all $d = 1 \dots D$ *//Updating weights.*

 numOfMisses = numOfMisses + 1

end if

end for

 Epochs = Epochs + 1

If Epoch % 200 == 0 **then**

 allowedMisses = allowedMisses + 1 *//Decrease accuracy as time moves.*

end if

if numOfMisses \leq allowedMisses **then** *//Stopping criteria.*

return W_{data}

end if

end while

PreceptronTest(Data, W_{data})

Act $\leftarrow \sum_{d=1}^{Data} w_d x_d$

If Act < 0 **then**

return negativeClass

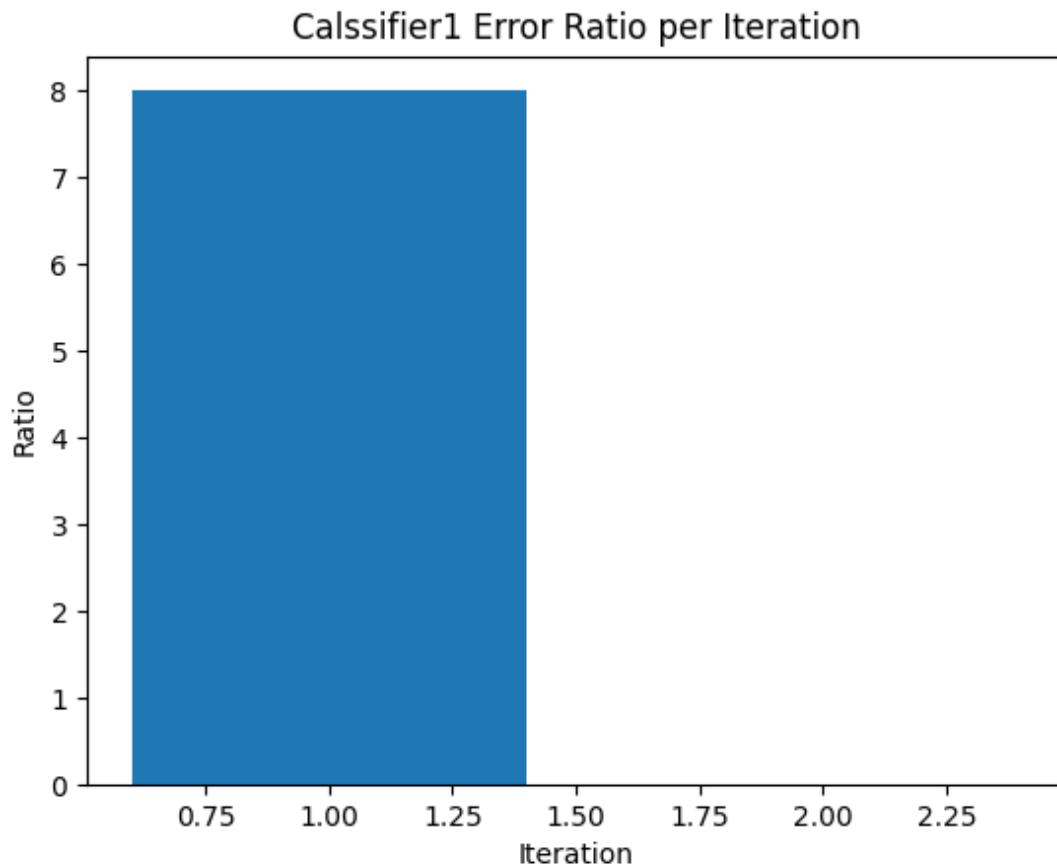
Else

return positiveClass

end if

Question. Plot the train error rate and test error rate against the number of iterations.

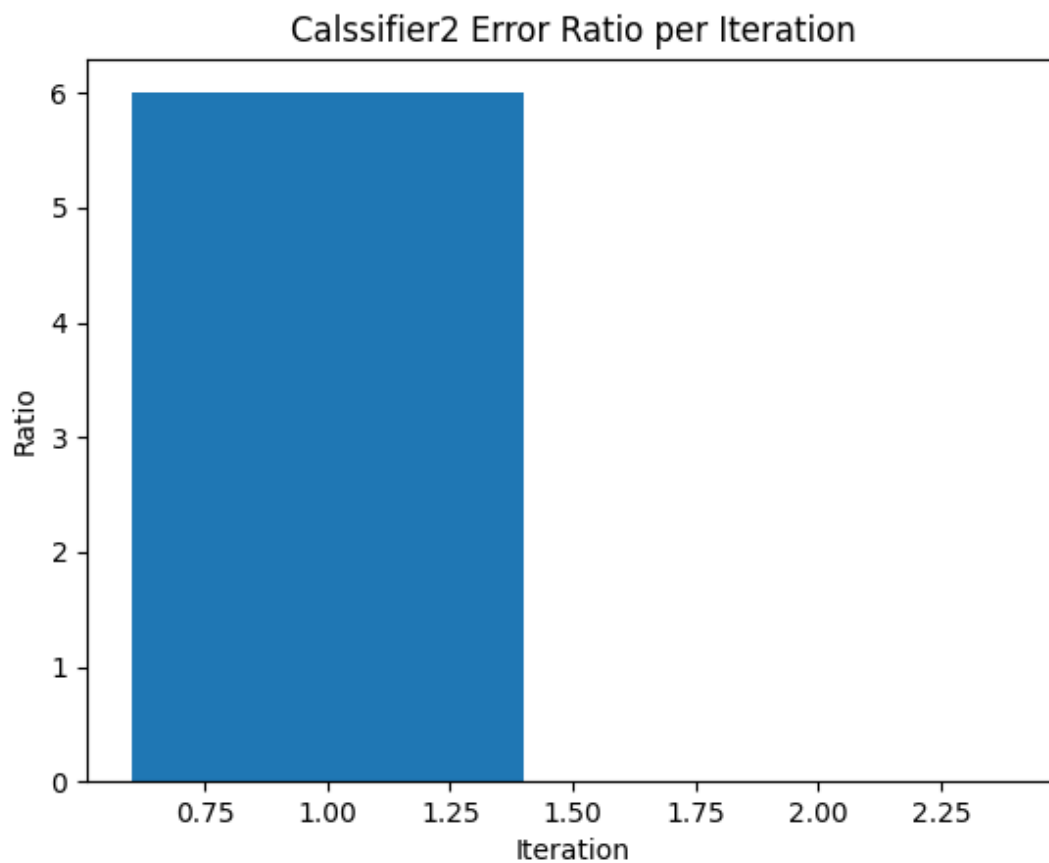
Figure 1



x=2.250 y=7.41

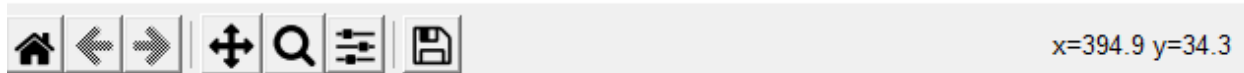
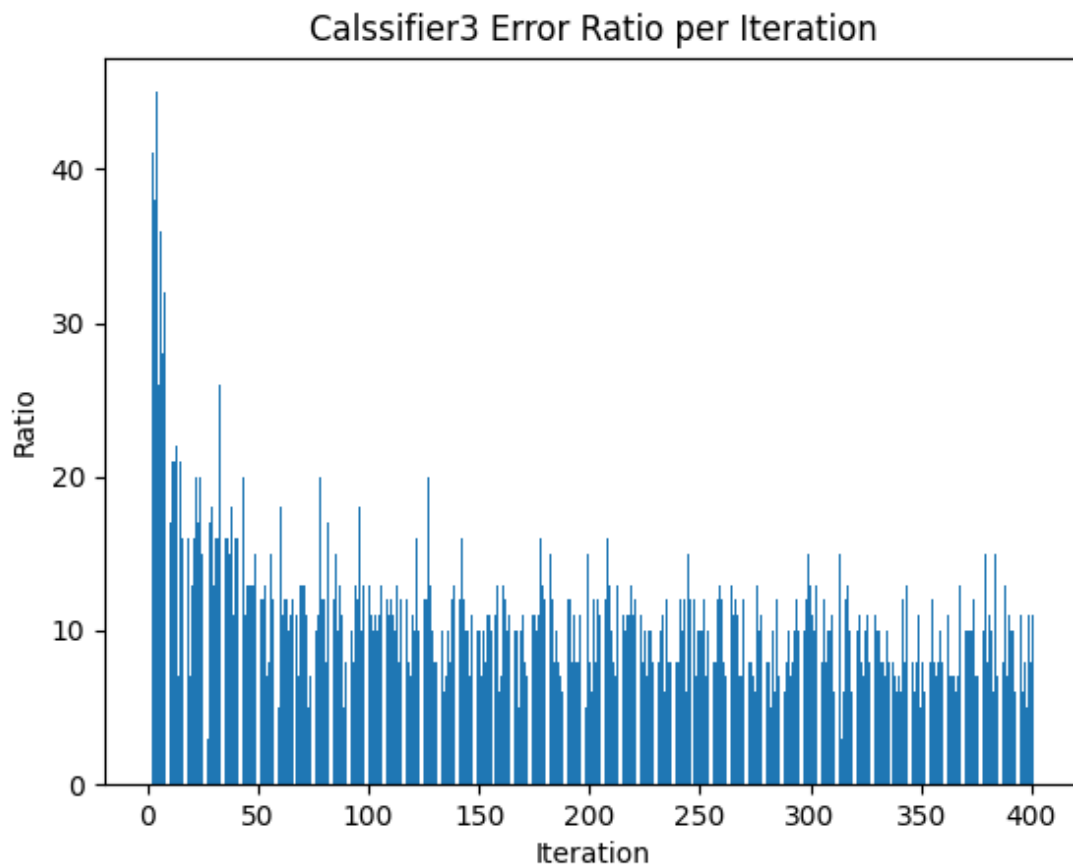
This figure is related to classifier1 which discriminates between class1 and class2. At the 2nd iteration the error ratio is 0%. This is because the data of the two classes are very different so the Perceptron learns to distinguish between them easily.

Figure 1



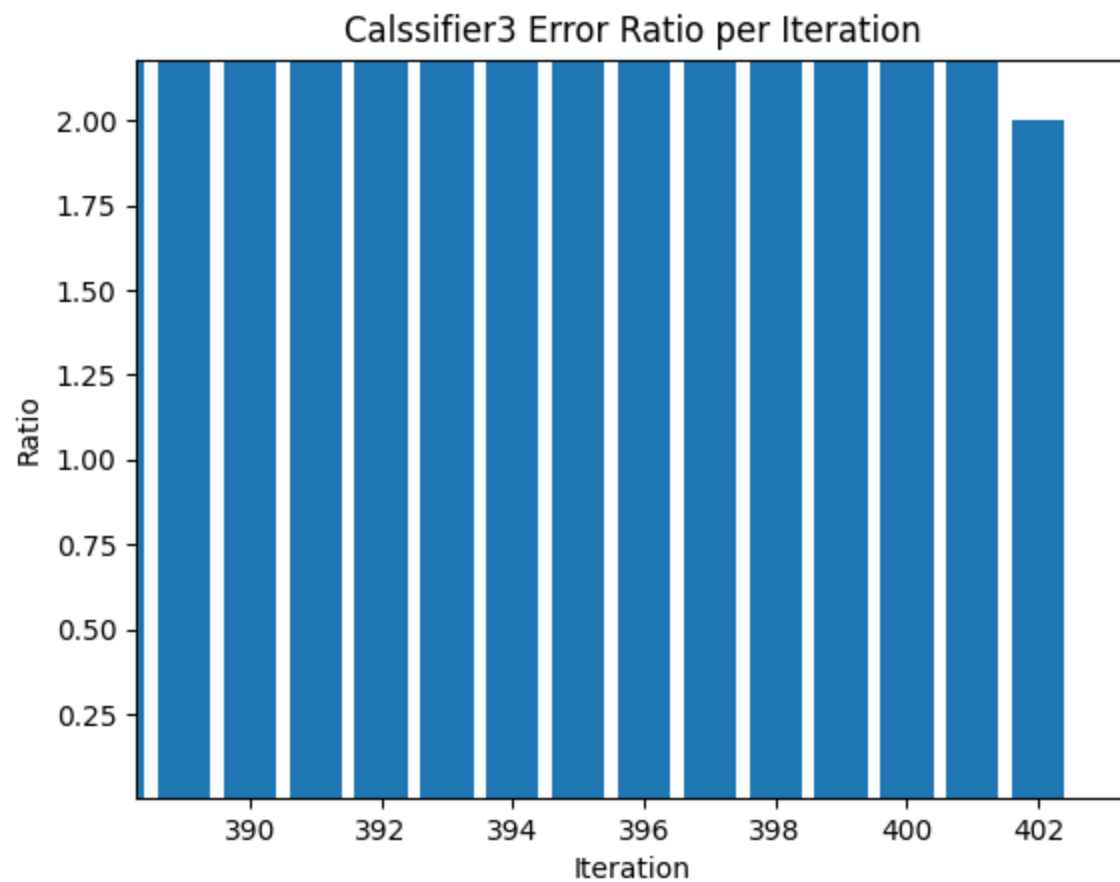
This figure is related to classifier2 which discriminates between class1 and class3. This case is the same as the previous one. The data is very different so the learning is done quickly.

Figure 1



This figure is related to classifier3 which discriminates between class2 and class3. The data of the two classes is more similar to those before and the learning takes more iteration, here it took 402 iterations to find the best result.

Note that it is not always that the last iteration is the best. Sometimes an older iteration produces better results.



Zoomed in image of the figure shows that at iteration 402 the error ratio was 2%.