

Week 9 – Implementing the Perceptron (worth 4% of overall grade)

Aim

This is a completely different lab from your previous topics. This week, we are introducing machine learning, and in particular, implementing the perceptron. You will require week 7's lecture notes, and the aim is first to program your own perceptron, then to test it with simple logic gate data, and finally to carry out character recognition. You will find the perceptron python file as well as the datasets in week 8 of MyPlace.

Part 1: Implement the Perceptron

Step one is to implement the perceptron so that you can use it to train logic gate data

- The perceptron is implemented as a class (you can start from the template on MyPlace). Call this file "perceptron.py. Create a script that will create a Perceptron object. Implement the predict, train, and test methods. You will find the week 7 lecture notes useful
- Create a dataset and labels using a logic gate (And, Or, Not etc.). Remember to add a constant! Try testing it before training, and then train your logic data. Use print outputs to see the weights change. The trained perceptron will be able to accurately predict the output, given input. Try it with an XOR.
- You should ultimately be able to produce output similar to that on the right

```
C:\Users\andrew.abel\AppData\Local\Programs\Python\Python39-64\Scripts\python.exe perceptron.py
No. inputs: 3
Max iterations: 5
Learning rate: 0.1
actual 0 est 1
actual 0 est 1
actual 0 est 1
actual 1 est 1
Accuracy: 25.0
[0.33333333 0.33333333 0.33333333]
[-0.01666667 0.28333333 0.23333333]
[-0.16666667 0.23333333 0.13333333]
[-0.21666667 0.18333333 0.13333333]
[-0.21666667 0.18333333 0.13333333]
[-0.21666667 0.18333333 0.13333333]
[-0.21666667 0.18333333 0.13333333]
actual 0 est 0
actual 0 est 0
actual 0 est 0
actual 1 est 1
Accuracy: 100.0

Process finished with exit code 0
```

Part 2: Character Recognition

When you have successfully been able to recognise logic gate data, use your perceptron to recognise digits. Look at your lecture nodes for guidance

- The mnist train.csv file can be used to train your node. The mnist test.csv file can be used to test your node. Start by trying to recognise the number "7". What recognition rate can you get?
- Try to recognise other numbers.
- Visualise the weights before and after training, can you see the difference?

Part 3: Batch Learning

One way to improve your perceptron is to implement a new method called `train_batch`, which will implement batch training. Rather than updating the weights after every row of training data, instead, you can predict every item of training data, and then perform a single weight update.

- You should implement this as a new method named **`train_batch(self, training_data, labels)`**, and not delete the old one.

Submission Instructions

- Demonstrate your code to the lab demonstrators and answer any questions they have.