

```

1  #!/usr/bin/python3
2
3  import matplotlib.pyplot as plot
4  import matplotlib.lines as mlines
5
6  from perceptron import Perceptron
7
8  def graph_line(p1, p2, label, color):
9      ax = plot.gca()
10     xmin, xmax = ax.get_xbound()
11
12     if(p2[0] == p1[0]):
13         xmin = xmax = p1[0]
14         ymin, ymax = ax.get_ybound()
15     else:
16         ymax = p1[1]+(p2[1]-p1[1])/(p2[0]-p1[0])*(xmax-p1[0])
17         ymin = p1[1]+(p2[1]-p1[1])/(p2[0]-p1[0])*(xmin-p1[0])
18
19     l = mlines.Line2D([xmin,xmax], [ymin,ymax], color=color, label=label)
20     ax.add_line(l)
21     return l
22
23  def get_point(per, x):
24     return (x, per.getY(x))
25
26  def each(data):
27     return range(len(data))
28
29  def getData(i):
30     parse = lambda s, i: float(s.split(' ')[i])
31     with open('perceptrondat{}.format(i)) as inf:
32         data = [[parse(l, 0), parse(l, 1)] for l in inf.readlines()]
33     outputs = [i for l in each(data)]
34     return data, outputs
35
36  data1, ans1 = getData(1)
37  data2, ans2= getData(2)
38
39  print(data1)
40
41  plot.scatter([e[0] for e in data1], [e[1] for e in data1], marker='o')
42  plot.scatter([e[0] for e in data2], [e[1] for e in data2], marker='^')
43
44
45  xarray = data1+data2
46  outputs = ans1+ans2
47  perceptron = Perceptron(xarray, outputs, False)
48
49  #for o,xar in zip(outputs,xarray):
50  #    print(o,perceptron.getValue(xar),"Success: "+str(o == perceptron.getValue(xar)))
51
52  for i in range(30):
53     perceptron.train()
54     p1 = get_point(perceptron, 0)
55     p2 = get_point(perceptron, 1)
56     if i != 29:
57         graph_line(p1, p2, '{}'.format(i), 'yellow')
58     else:
59         graph_line(p1, p2, '{}'.format(i), 'red')
60
61  plot.show()
62

```