Assignment #1

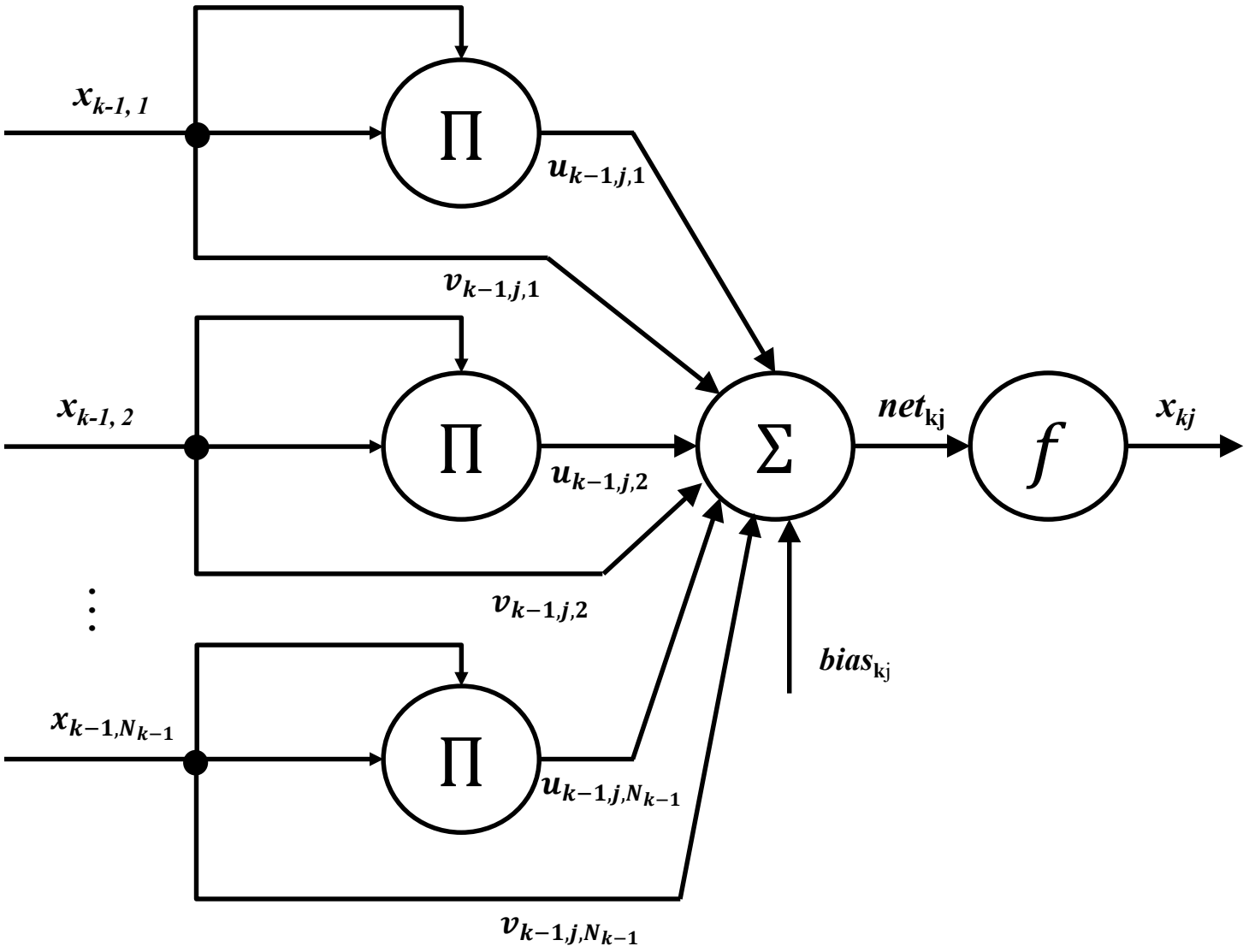
Problem 1: Suppose the output of each neuron in a multi-layer perceptron is:

$$x_{kj} = f\left(\sum_{i=1}^{N_{k-1}} \left(u_{kji} x_{k-1,i}^2 + v_{kji} x_{k-1,i}\right) + b_{kj}\right)$$

where both u_{kji} and v_{kji} are the weights connecting the i^{th} unit in the layer k-1 to the j^{th} unit in the layer k, b_{kj} is the bias of the j^{th} unit in the layer k, N_k is the number of units if $1 \leq k \leq M$ and $f(\cdot)$ is the sigmoidal activation function.

Please derive a back-propagation algorithm for multilayer quadratic perceptron (MLQP) in on-line or sequential mode.

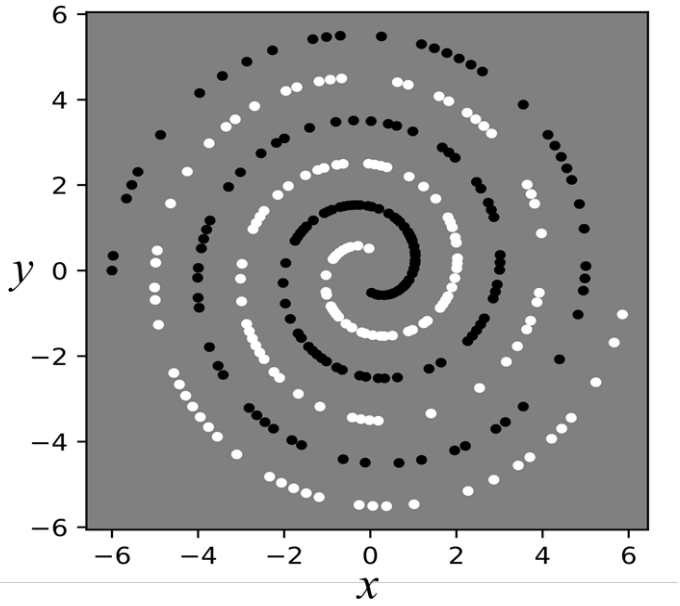
Multilayer Quadratic Perceptron (MLQP)



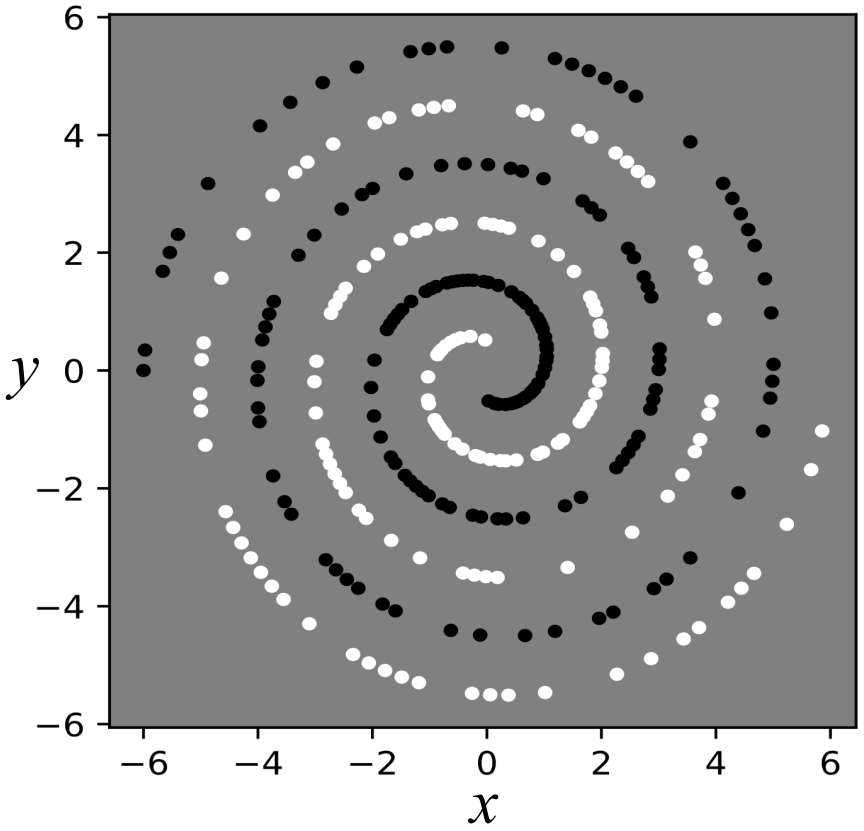
Assignment #1

Problem 2:

Please implement an on-line BP algorithm for MLQP (you can use any programming language), train an MLQP with one hidden layer to classify two spirals problem, and compare the training time and decision boundaries at three different learning rates .



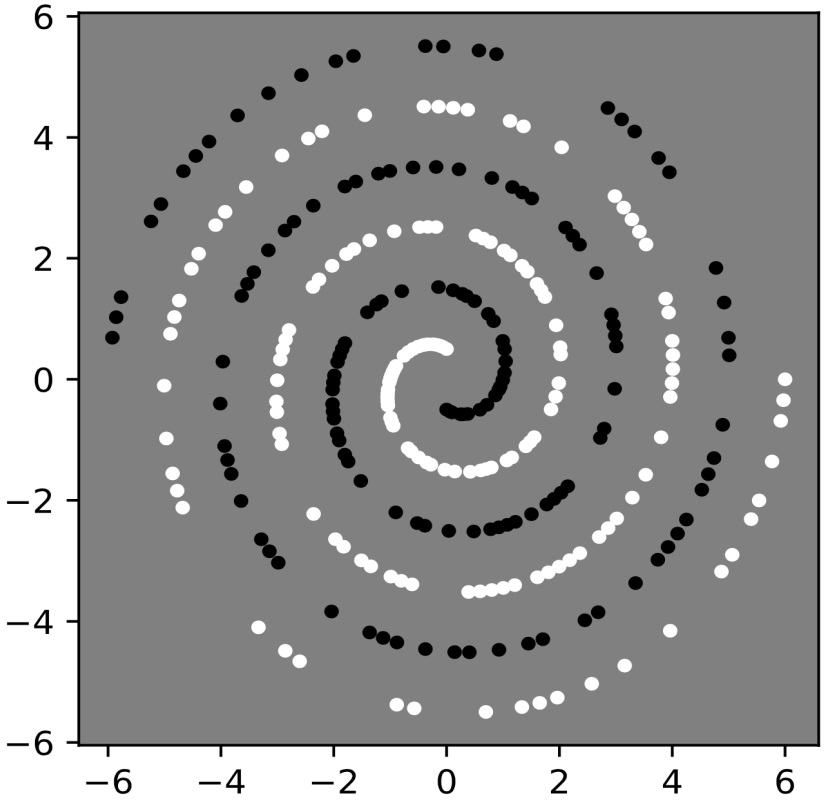
Training Dataset : 300 samples



x	y	Label
0.37682	-5.50887	1
0.51316	-1.51936	1
0.91197	0.83395	0
0.66642	-4.49772	0
-1.67144	-2.88335	1
-5.39738	2.31078	0
-4.87164	3.17704	0
1.96371	-0.17585	1
5.66311	-1.68343	1
-1.50417	2.22890	1
.....		

Label:
white point: 1
black point: 0

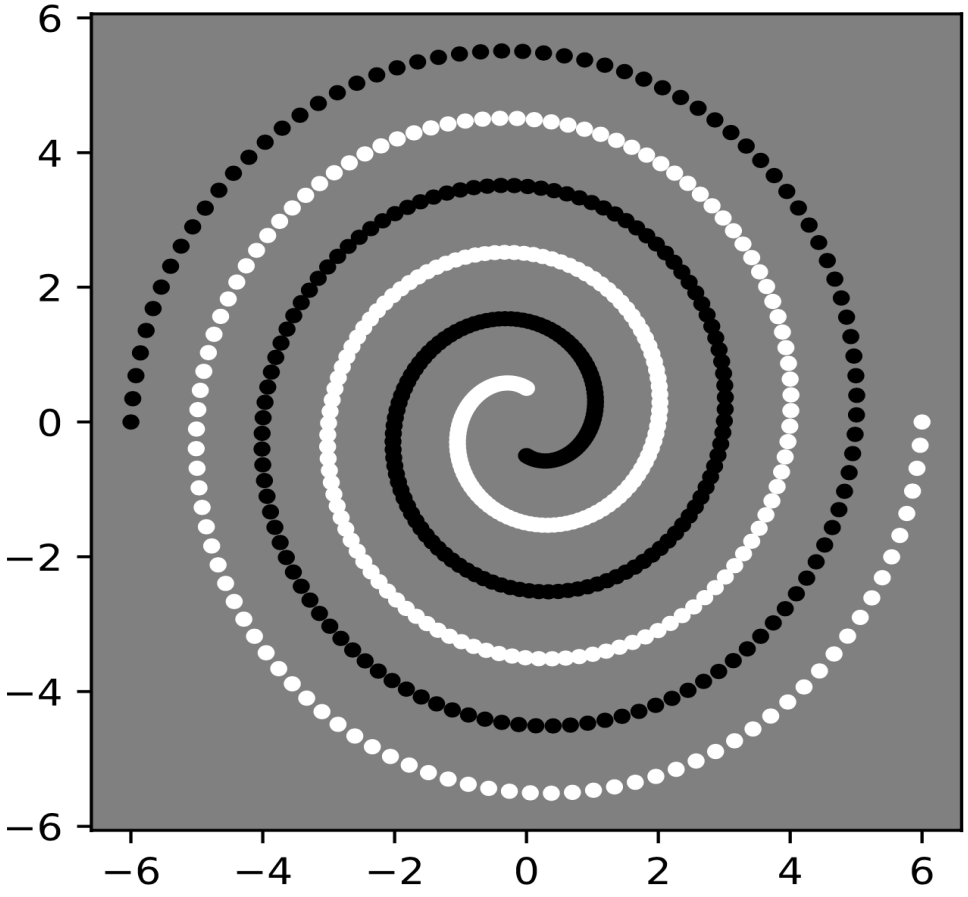
Test Dataset: 300 samples




```
 0.11797   4.49009 1
 0.60694  -1.50424 1
-0.48132   2.51455 1
 4.63888  -1.56676 0
-0.90070  -2.19894 0
-3.70490   4.36335 0
 0.37556   4.45745 1
-2.60388  -4.65960 1
 0.40389  -4.51038 0
 1.98898  -0.06272 1
```

.....

Overall Dataset



Assignment #1

Problem 3:

a) Divide the two spirals problem into four or nine sub-problems randomly and with prior knowledge, respectively

b) Train MLQP on these sub-problems and construct two min-max modular networks

a) Compare training time and decision boundaries of the above two min-max modular networks

Two Spiral Formula

- A general formula of Archimedean spiral in polar coordinates:

$$r = a + b\theta$$

- In this case, the formulas for postive and negative samples are:

$$\theta = r\pi$$

$$\theta = r\pi + \pi$$

- In Python, the rectangular coordinates can be calculated for each radius r by following expression:

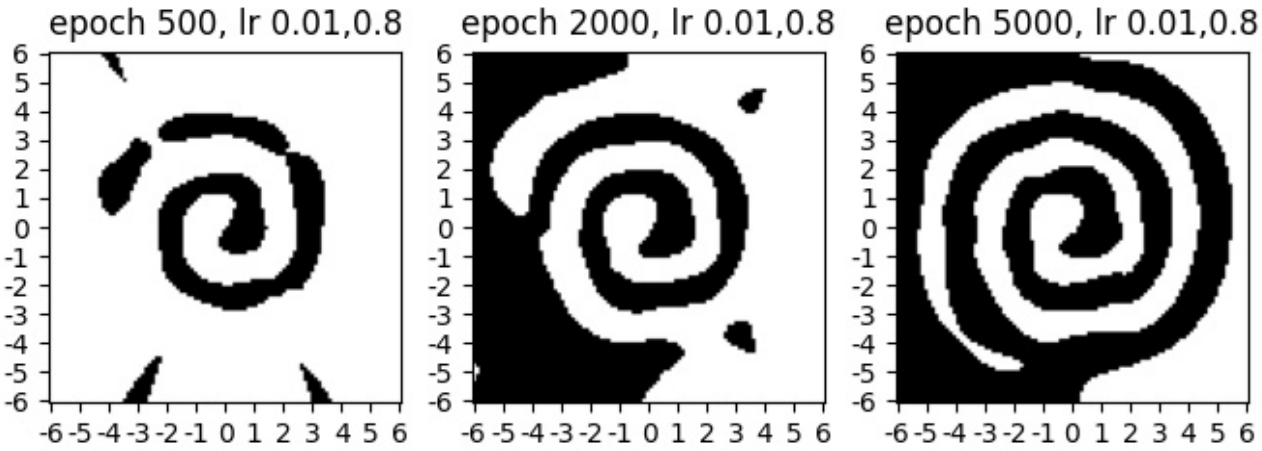
```
theta = r * math.pi
X1 = (r * math.cos(theta), r * math.sin(theta))
X2 = (-r * math.cos(theta), -r * math.sin(theta))
```

Visualization of Decision Boundary

- Use the library Matplotlib in Python

```
import matplotlib.pyplot as plt
```

- Uniformly sample points on a square area and color every point according to your classifier's prediction
- Below is an example of decision boundary at different epochs



平时作业评分标准

1. 平时作业的评分标准是结果的正确性和程序的可执行性
2. 作业报告要求用LaTex、Word等电子文档工具书写，不接受手写拍照的报告
3. 迟交平时作业按下列比列降低评分等级：
    a) 1天之内：降低10%
    b) 2天之内：降低20%
    c) 3天之内：降低30%
    d) 超过三天不再接受
4. 严禁抄袭！一旦发现零分处理！抄袭者和被抄者同等责任

作业提交截止日期: 2022年3月13日