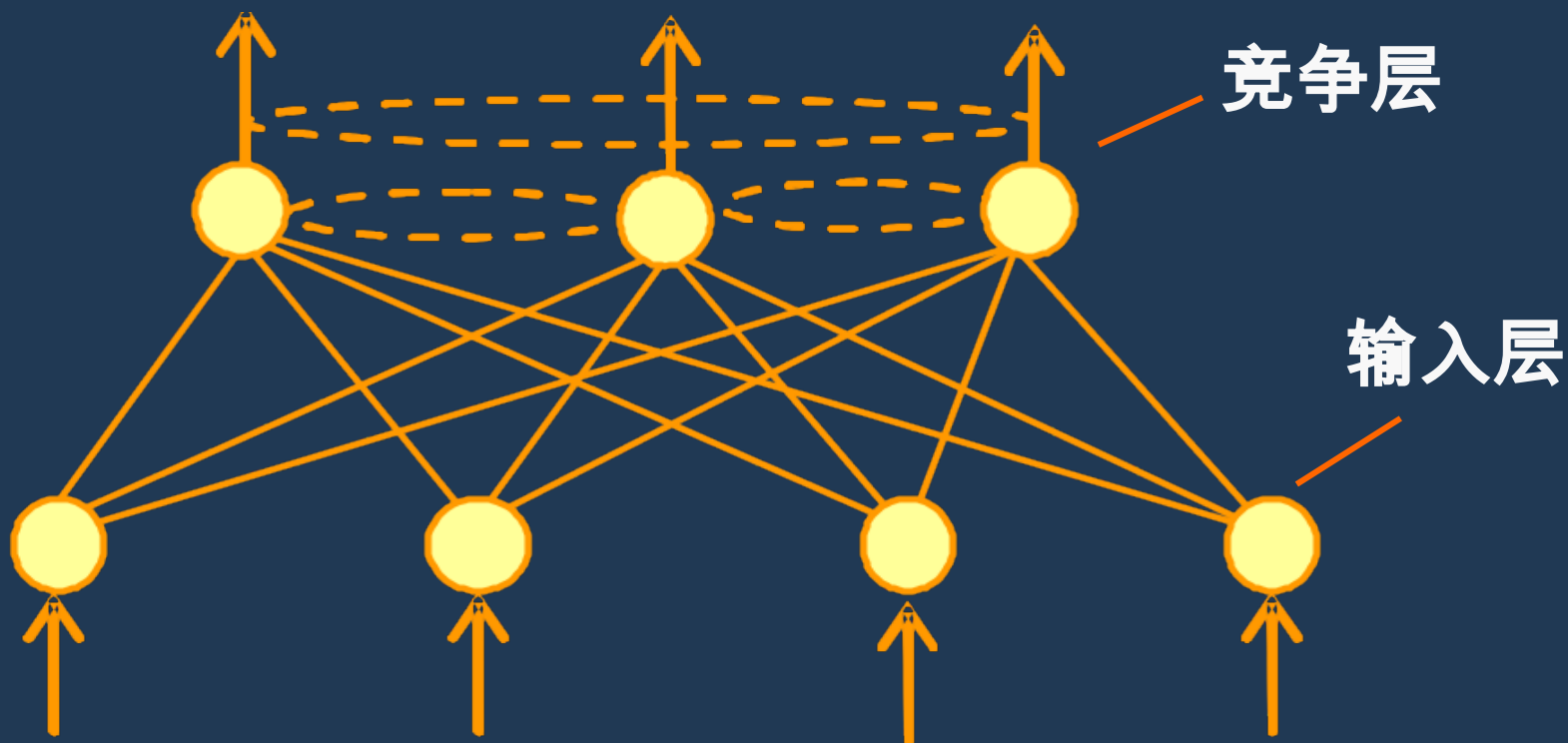


第四章 自组织神经网络

4.1 竞争学习的概念与原理

4.2 自组织特征映射神经网络

第四章 自组织神经网络



自组织神经网络的典型结构



第四章 自组织神经网络

自组织学习 (self-organized learning) :

通过自动寻找样本中的内在规律和本质属性，自组织、自适应地改变网络参数与结构。

自组织网络的自组织功能是通过竞争学习 (competitive learning) 实现的。



4.1 竞争学习的概念与原理

4.1.1 基本概念

分类——分类是在类别知识等导师信号的指导下，将待识别的输入模式分配到各自的模式类中去。

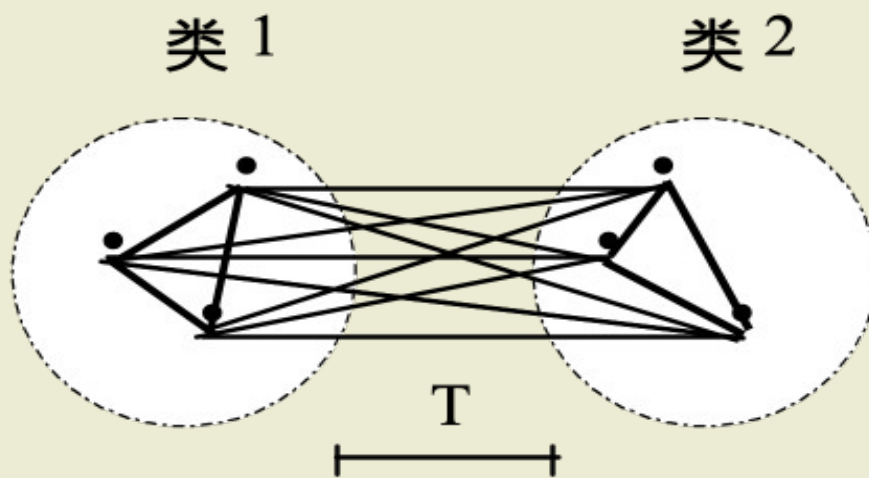
聚类——无导师指导的分类称为聚类，聚类的目的是将相似的模式样本划归一类，而将不相似的分离开。



4.1.1 基本概念

- 相似性测量—欧式距离法

$$\| \mathbf{X} - \mathbf{X}_i \| = \sqrt{(\mathbf{X} - \mathbf{X}_i)^T (\mathbf{X} - \mathbf{X}_i)}$$



(a) 基于欧式距离的相似性测量

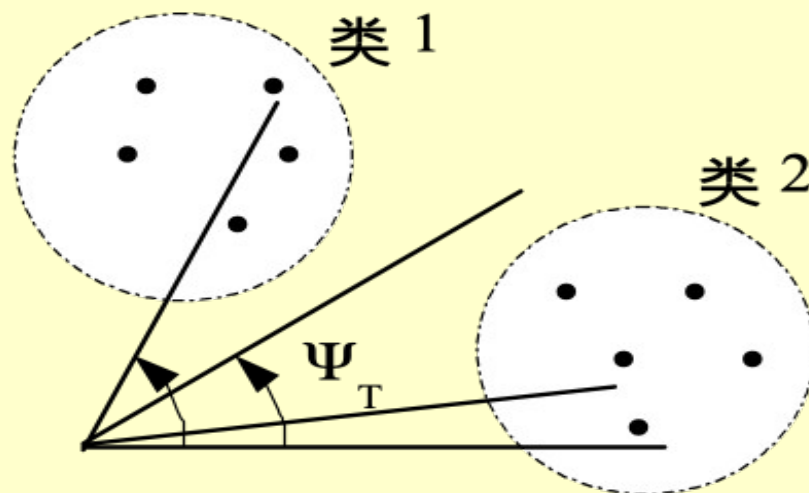
4.1.1 基本概



念

- 相似性测量—余弦法

$$\cos \psi = \frac{\mathbf{X}^T \mathbf{X}_i}{\|\mathbf{X}\| \|\mathbf{X}_i\|}$$



(b)基于余弦法的相似性测量

4.1.2 竞争学习原理

竞争学习规则—— Winner-Take-All

网络的输出神经元之间相互竞争以求被激活，结果在每一时刻只有一个输出神经元被激活。这个被激活的神经元称为竞争获胜神经元，而其它神经元的状态被抑制，故称为 **Winner Take All**。



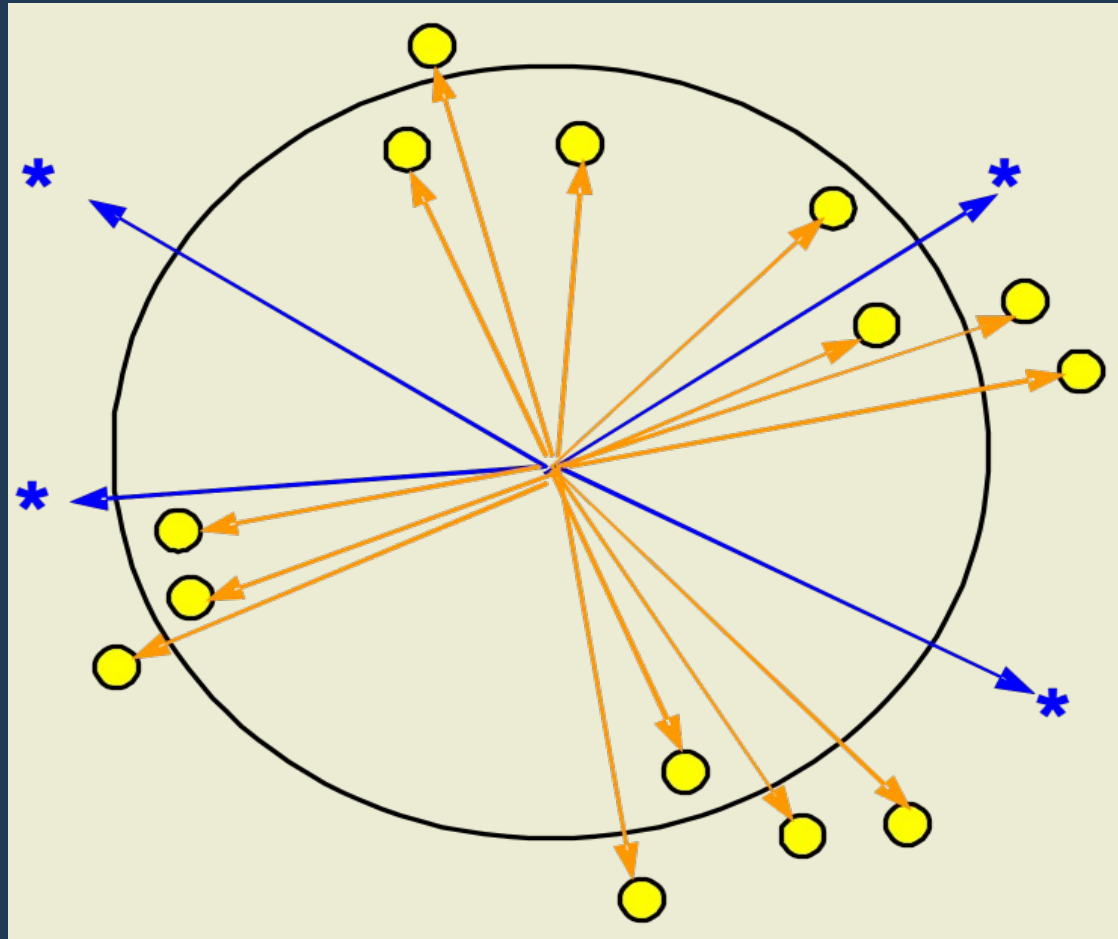
竞争学习规则—— Winner-Take-All

1. 向量归一化 首先将当前输入模式向量 \mathbf{X} 和竞争层中各神经元对应的内星向量 \mathbf{W}_j 全部进行归一化处理； ($j=1,2,\dots,m$)

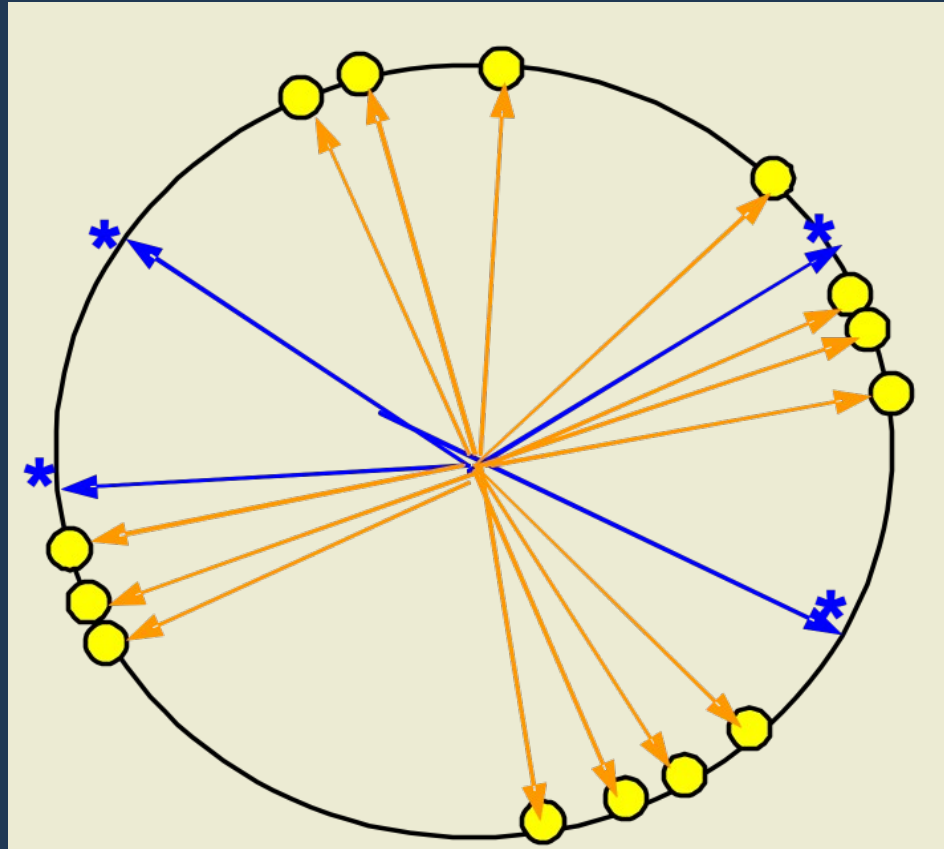
$$\hat{\mathbf{X}} = \frac{\mathbf{X}}{\|\mathbf{X}\|} = \left(\frac{x_1}{\sqrt{\sum_{j=1}^n x_j^2}} \quad \dots \quad \frac{x_n}{\sqrt{\sum_{j=1}^n x_j^2}} \right)^T$$



向量归一化之 **前**



向量归一化之后



竞争学习原理

竞争学习规则——Winner-Take-All

2. 寻找获胜神经元 当网络得到一个输入模式向量时，竞争层的所有神经元对应的内星权向量均与其进行相似性比较，并将最相似的内星权向量判为竞争获胜神经元。

欲使两单位向量最相似，须使其点积最大。即：

$$\hat{\mathbf{W}}_{j^*}^T \hat{\mathbf{X}} = \max_{j \in \{1, 2, \dots, m\}} (\hat{\mathbf{W}}_j^T \hat{\mathbf{X}})$$



竞争学习规则—— Winner-Take-All

$$\|\hat{\mathbf{X}} - \hat{\mathbf{W}}_{j^*}\| = \min_{j \in \{1, 2, \dots, m\}} \left\{ \|\hat{\mathbf{X}} - \hat{\mathbf{W}}_j\| \right\}$$

$$\|\hat{\mathbf{X}} - \hat{\mathbf{W}}_{j^*}\| = \sqrt{(\hat{\mathbf{X}} - \hat{\mathbf{W}}_{j^*})^T (\hat{\mathbf{X}} - \hat{\mathbf{W}}_{j^*})}$$

$$= \sqrt{\hat{\mathbf{X}}^T \hat{\mathbf{X}} - 2\hat{\mathbf{W}}_{j^*}^T \hat{\mathbf{X}} + \hat{\mathbf{W}}_{j^*}^T \hat{\mathbf{W}}_{j^*}} = \sqrt{2(1 - \mathbf{w}_{j^*}^T \hat{\mathbf{X}})}$$

从上式可以看出，欲使两单位向量的欧式距离最小，须使两向量的点积最大。即：

$$\hat{\mathbf{W}}_{j^*}^T \hat{\mathbf{X}} = \max_{j \in \{1, 2, \dots, m\}} (\hat{\mathbf{W}}_j^T \hat{\mathbf{X}})$$



竞争学习规则——胜者为王 (Winner-Take-All)

3. 网络输出与权值调整

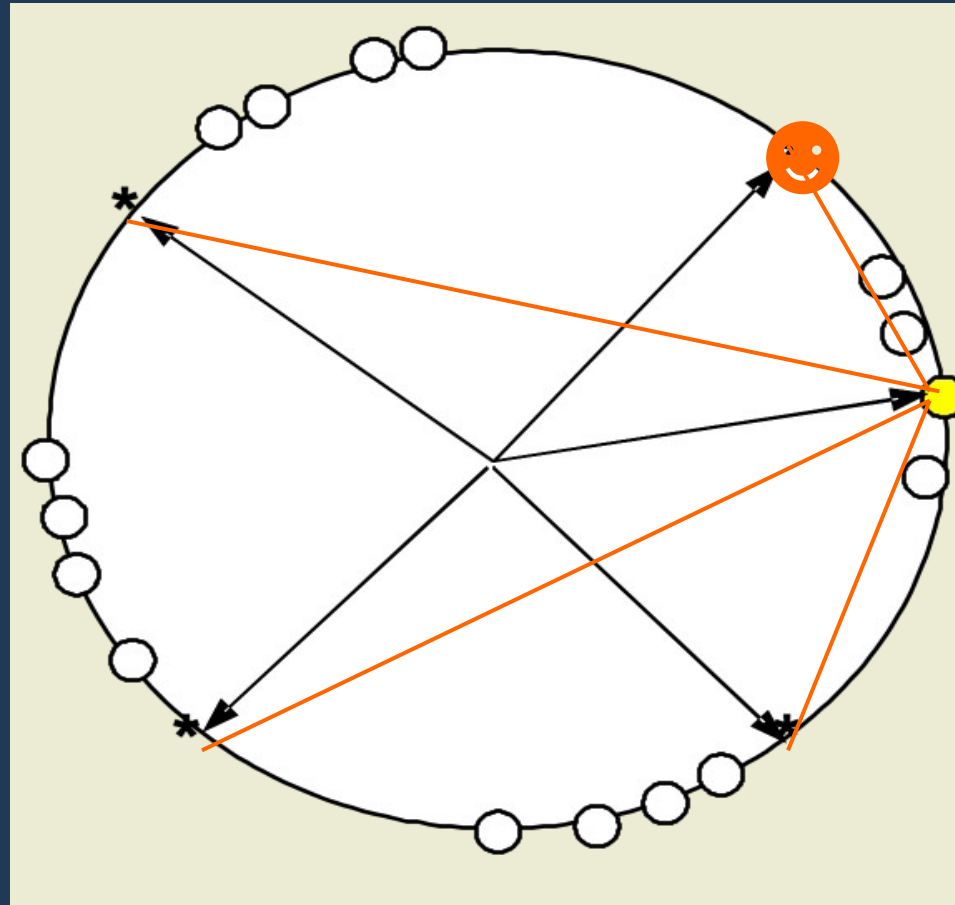
$$o_j(t+1) = \begin{cases} 1 & j = j^* \\ 0 & j \neq j^* \end{cases}$$

$$\left\{ \begin{array}{l} W_{j^*}(t+1) = \hat{W}_{j^*}(t) + \Delta W_{j^*} = \hat{W}_{j^*}(t) + \mu(t)(\hat{X} - \hat{W}_{j^*}) \\ \mathbf{W}_j(t+1) = \hat{\mathbf{W}}_j(t) \quad j \neq j^* \end{array} \right.$$

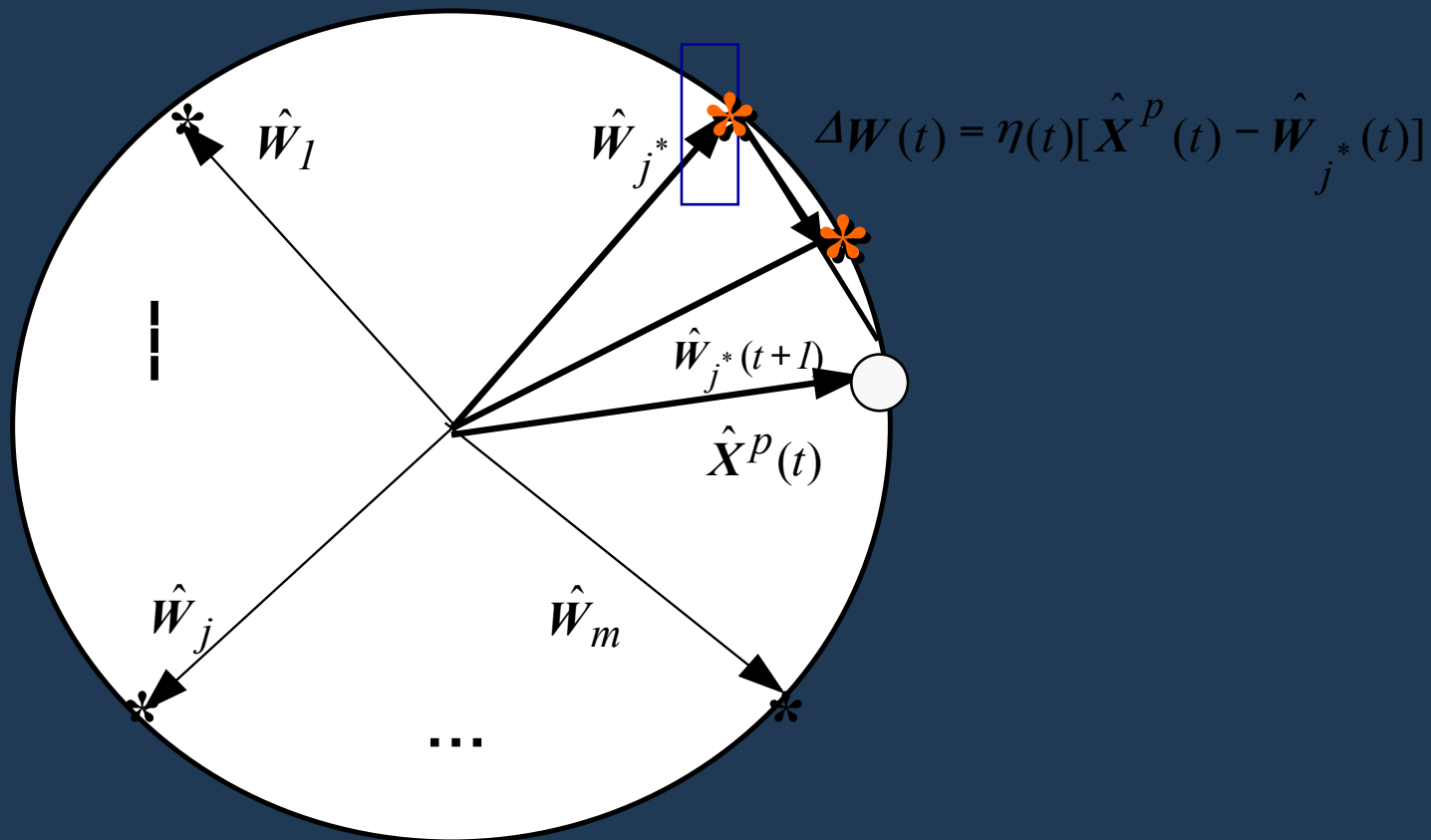
步骤 3 完成后回到步骤 1 继续训练，直到学习率衰减到 0。



竞争学习的几何意义



竞争学习的几何意义



例 4.1 用竞争学习算法将下列各模式分为 2 类：

$$\mathbf{X}^1 = \begin{pmatrix} 0.8 \\ 0.6 \end{pmatrix} \quad \mathbf{X}^2 = \begin{pmatrix} 0.1736 \\ -0.9848 \end{pmatrix} \quad \mathbf{X}^3 = \begin{pmatrix} 0.707 \\ 0.707 \end{pmatrix} \quad \mathbf{X}^4 = \begin{pmatrix} 0.342 \\ -0.9397 \end{pmatrix} \quad \mathbf{X}^5 = \begin{pmatrix} 0.6 \\ 0.8 \end{pmatrix}$$

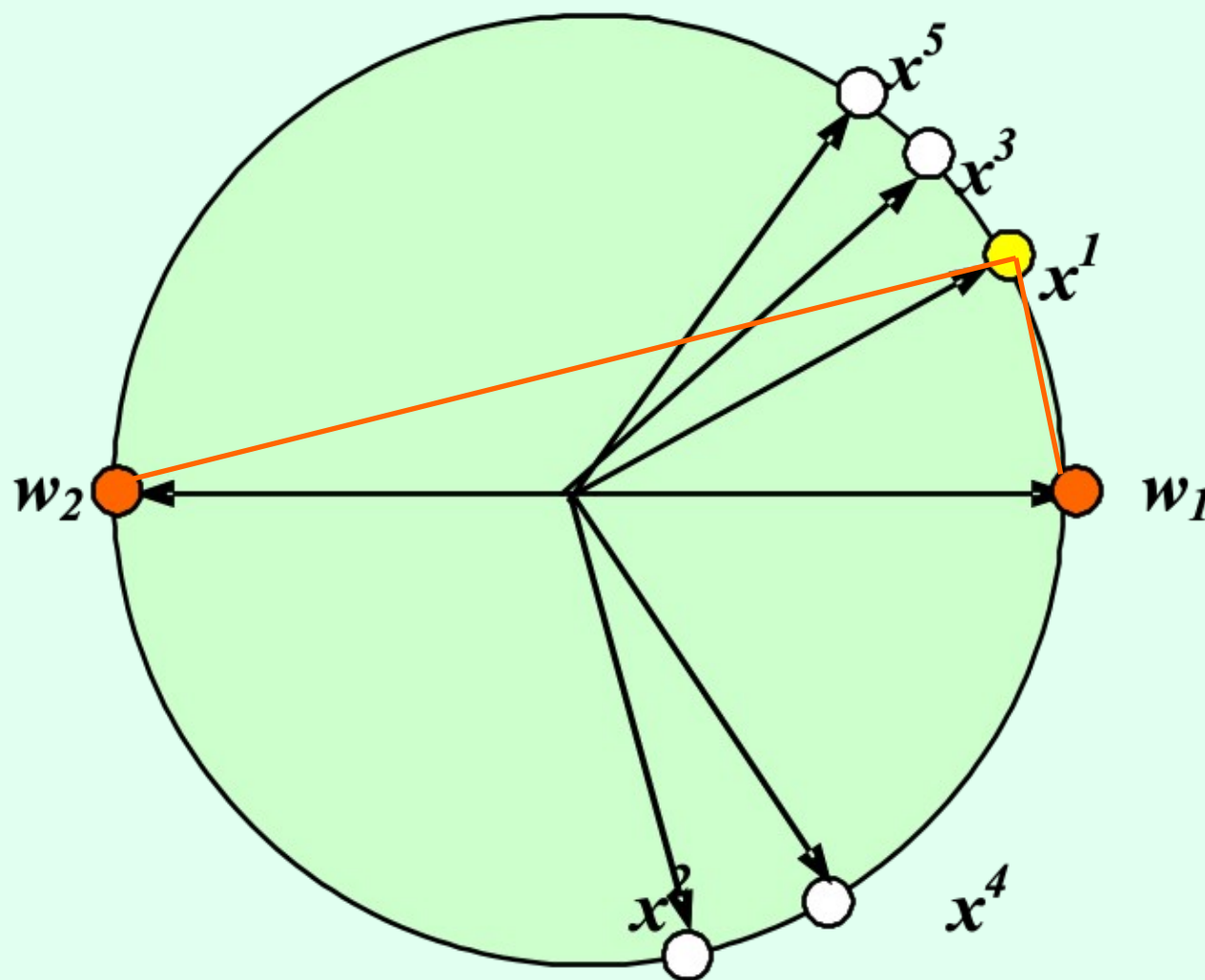
解：为作图方便，将上述模式转换成极坐标形式：

$$\mathbf{X}^1 = 1 \angle 36.89^\circ \quad \mathbf{X}^2 = 1 \angle -80^\circ \quad \mathbf{X}^3 = 1 \angle 44.5^\circ \quad \mathbf{X}^4 = 1 \angle -70^\circ \quad \mathbf{X}^5 = 1 \angle 53.13^\circ$$

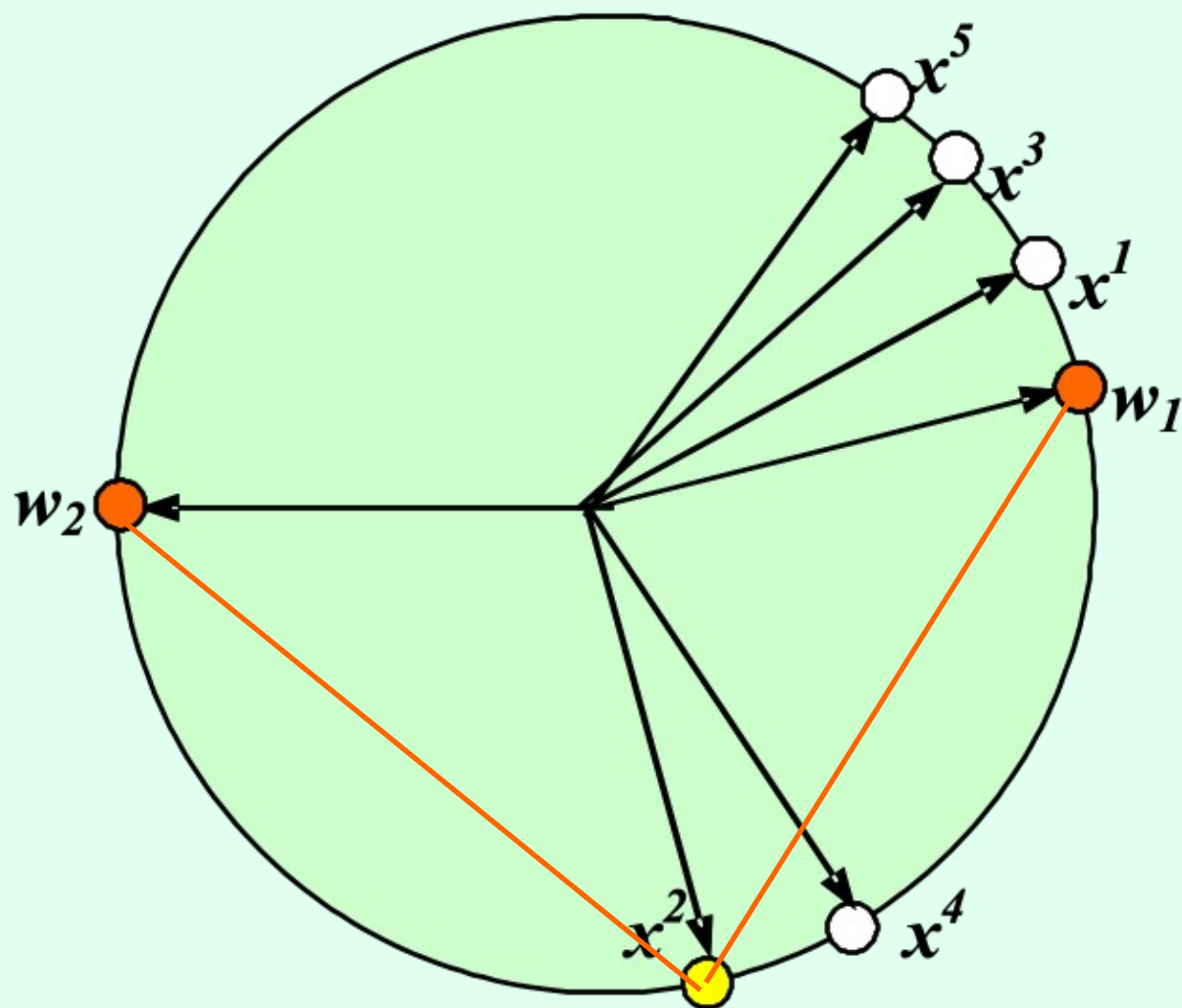
竞争层设两个权向量，随机初始化为单位向量：

$$\mathbf{W}_1(0) = \begin{pmatrix} 1 \\ 0 \end{pmatrix} = 1 \angle 0^\circ \quad \mathbf{W}_2(0) = \begin{pmatrix} -1 \\ 0 \end{pmatrix} = 1 \angle 180^\circ$$

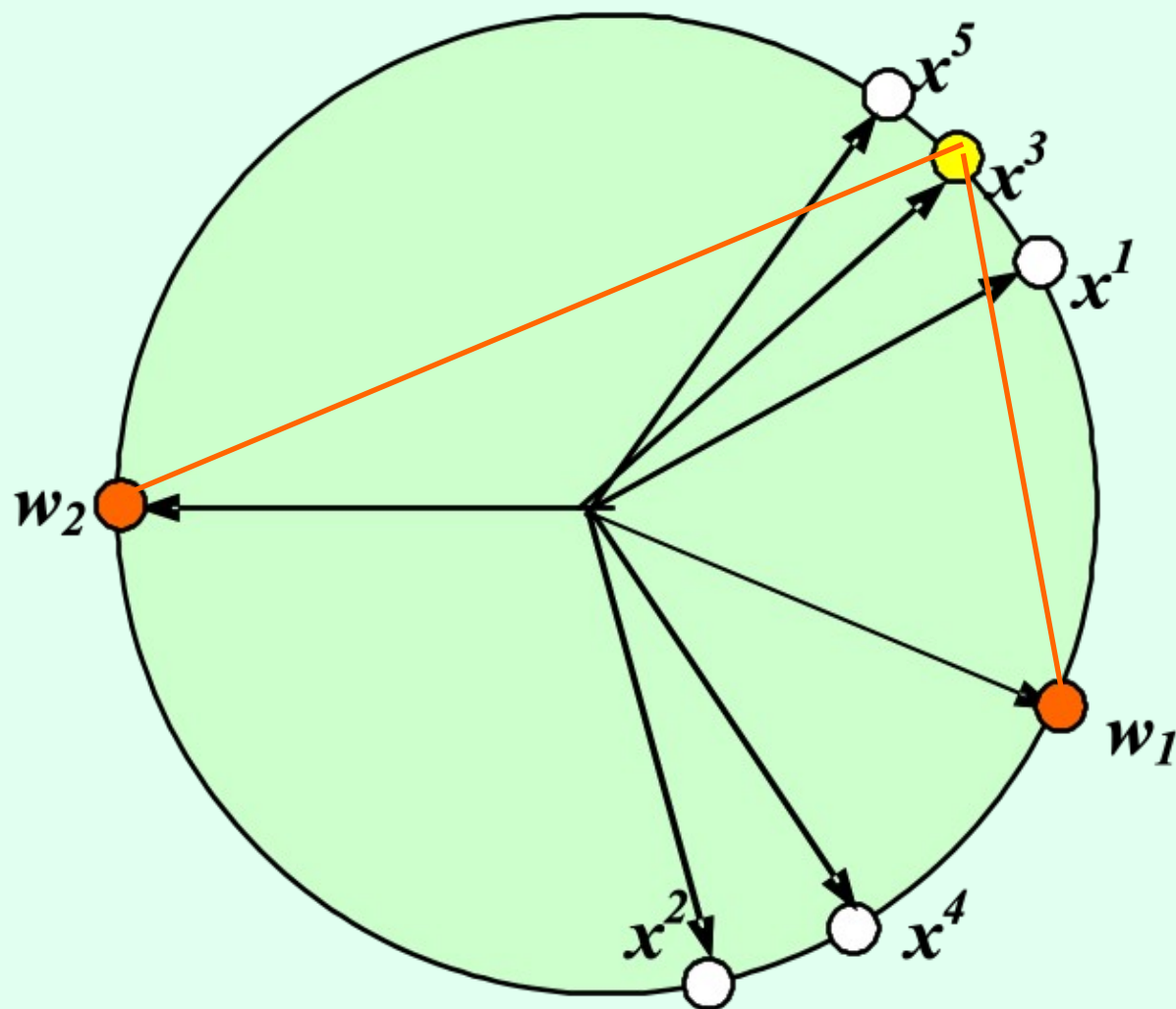




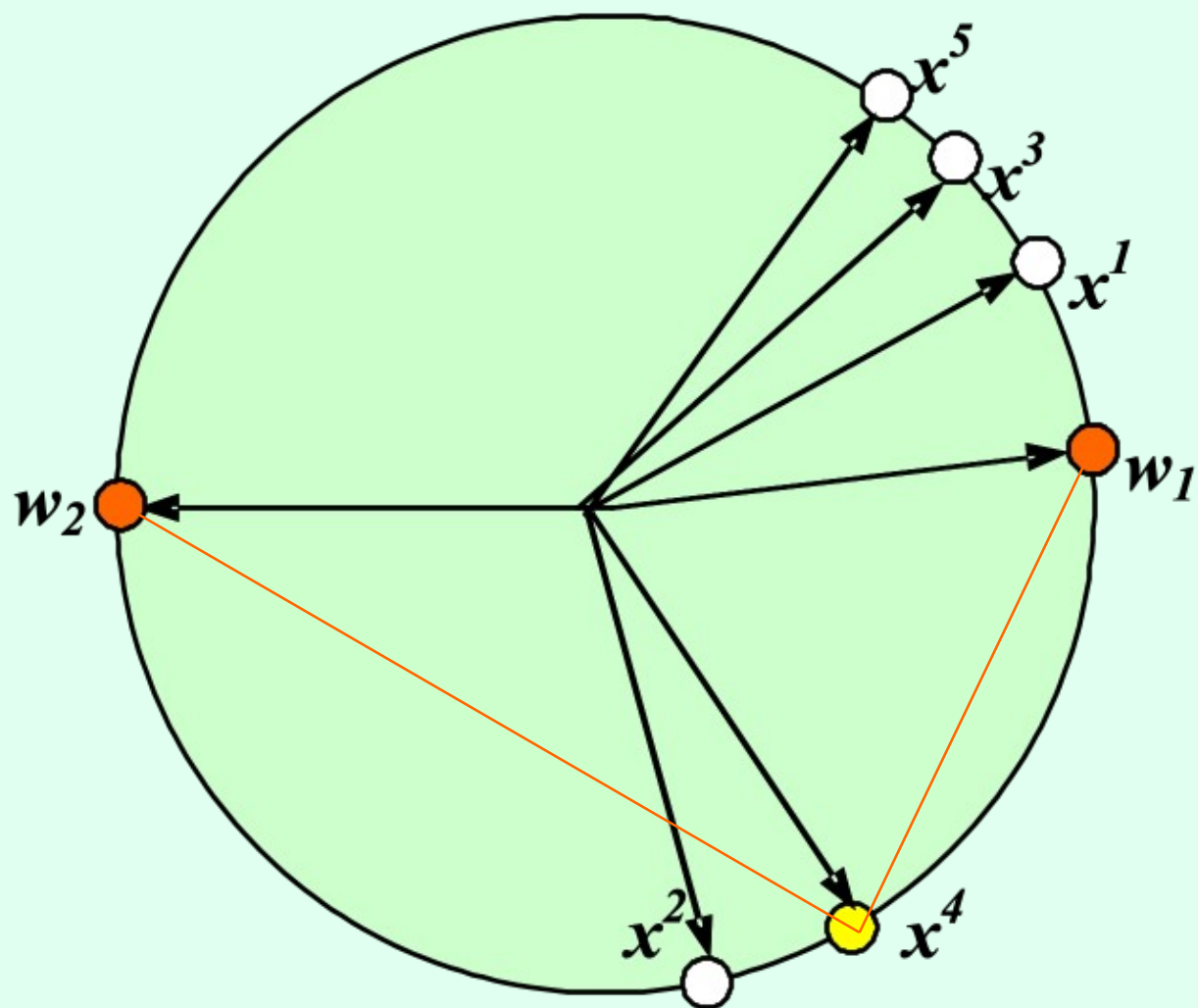
训练次数	w	w
1	18.43°	-180°
2	-30.8°	-180°
3	7°	-180°
4	-32°	-180°
5	11°	-180°
6	24°	-180°
7	24°	-130°
8	34°	-130°
9	34°	-100°
10	44°	-100°
11	40.5°	-100°
12	40.5°	-90°
13	43°	-90°
14	43°	-81°
15	47.5°	-81°
16	42°	-81°
17	42°	-80.5°
18	43.5°	-80.5°
19	43.5°	-75°
20	48.5°	-75°



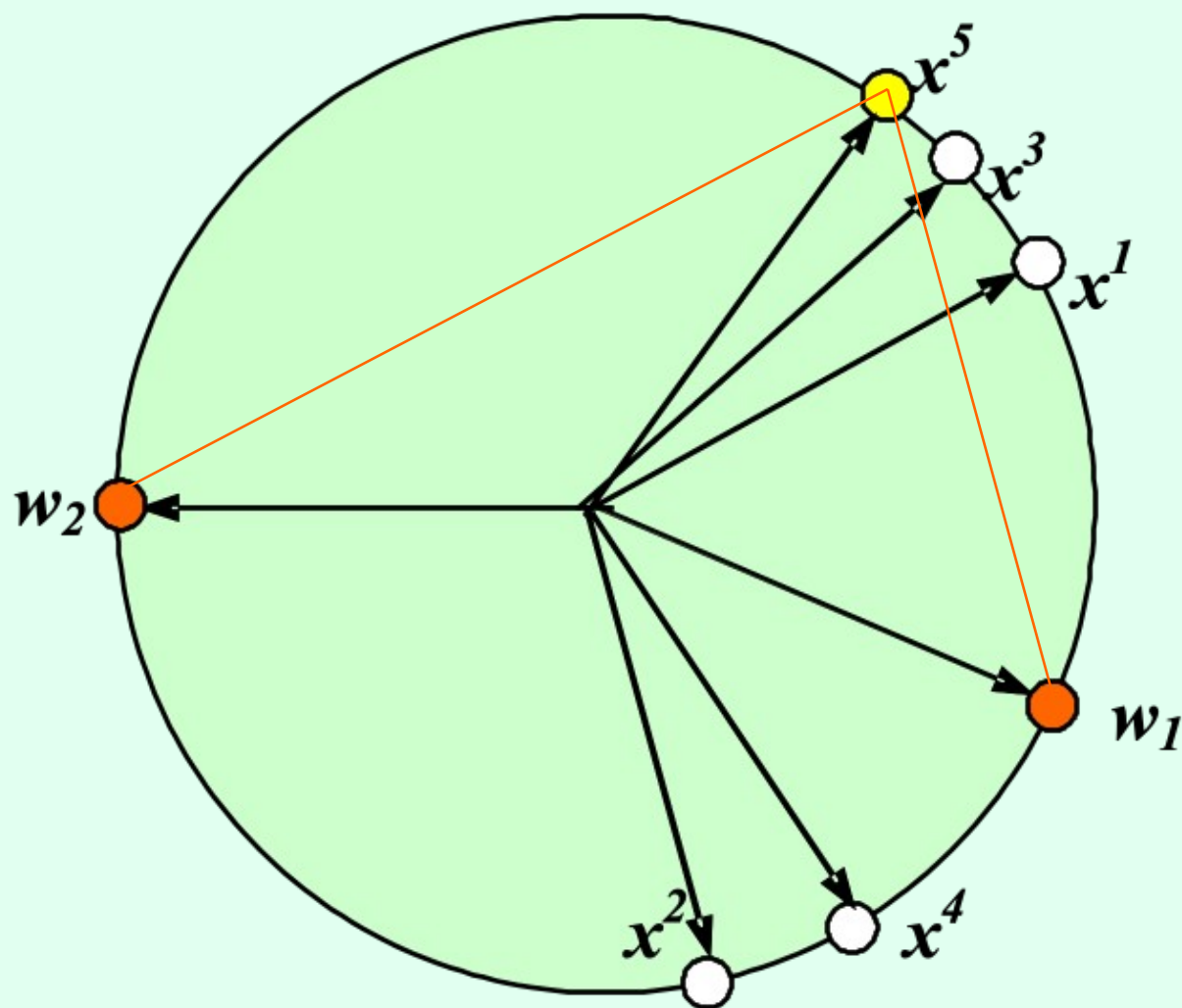
训练次数	w	w
1	18.43°	-180°
2	-30.8°	-180°
3	7°	-180°
4	-32°	-180°
5	11°	-180°
6	24°	-180°
7	24°	-130°
8	34°	-130°
9	34°	-100°
10	44°	-100°
11	40.5°	-100°
12	40.5°	-90°
13	43°	-90°
14	43°	-81°
15	47.5°	-81°
16	42°	-81°
17	42°	-80.5°
18	43.5°	-80.5°
19	43.5°	-75°
20	48.5°	-75°



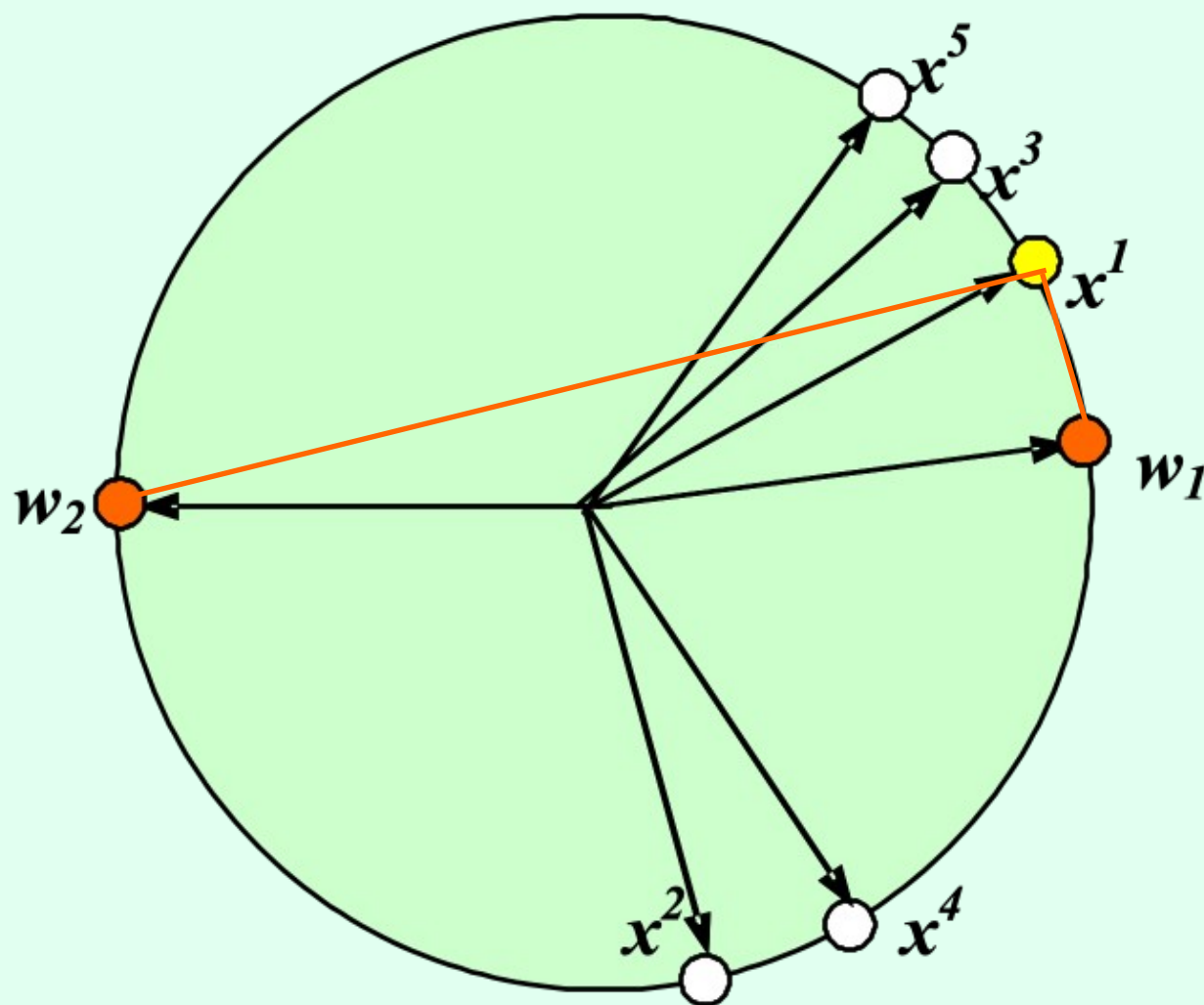
训练次数	W	W
1	18.43°	-180°
2	-30.8°	-180°
3	7°	-180°
4	-32°	-180°
5	11°	-180°
6	24°	-180°
7	24°	-130°
8	34°	-130°
9	34°	-100°
10	44°	-100°
11	40.5°	-100°
12	40.5°	-90°
13	43°	-90°
14	43°	-81°
15	47.5°	-81°
16	42°	-81°
17	42°	-80.5°
18	43.5°	-80.5°
19	43.5°	-75°
20	48.5°	-75°



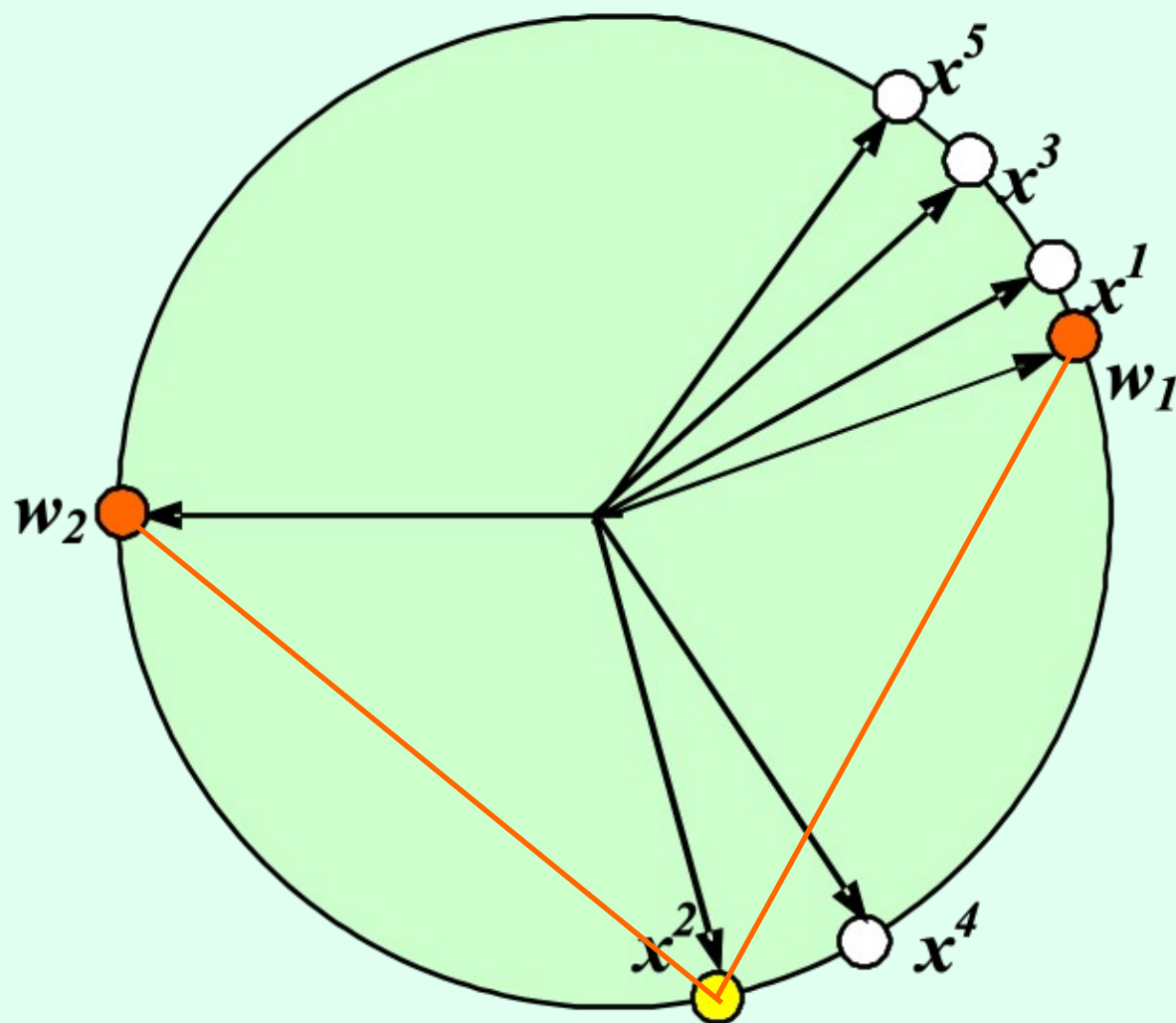
训练次数	w	w
1	18.43°	-180°
2	-30.8°	-180°
3	7°	-180°
4	-32°	-180°
5	11°	-180°
6	24°	-180°
7	24°	-130°
8	34°	-130°
9	34°	-100°
10	44°	-100°
11	40.5°	-100°
12	40.5°	-90°
13	43°	-90°
14	43°	-81°
15	47.5°	-81°
16	42°	-81°
17	42°	-80.5°
18	43.5°	-80.5°
19	43.5°	-75°
20	48.5°	-75°



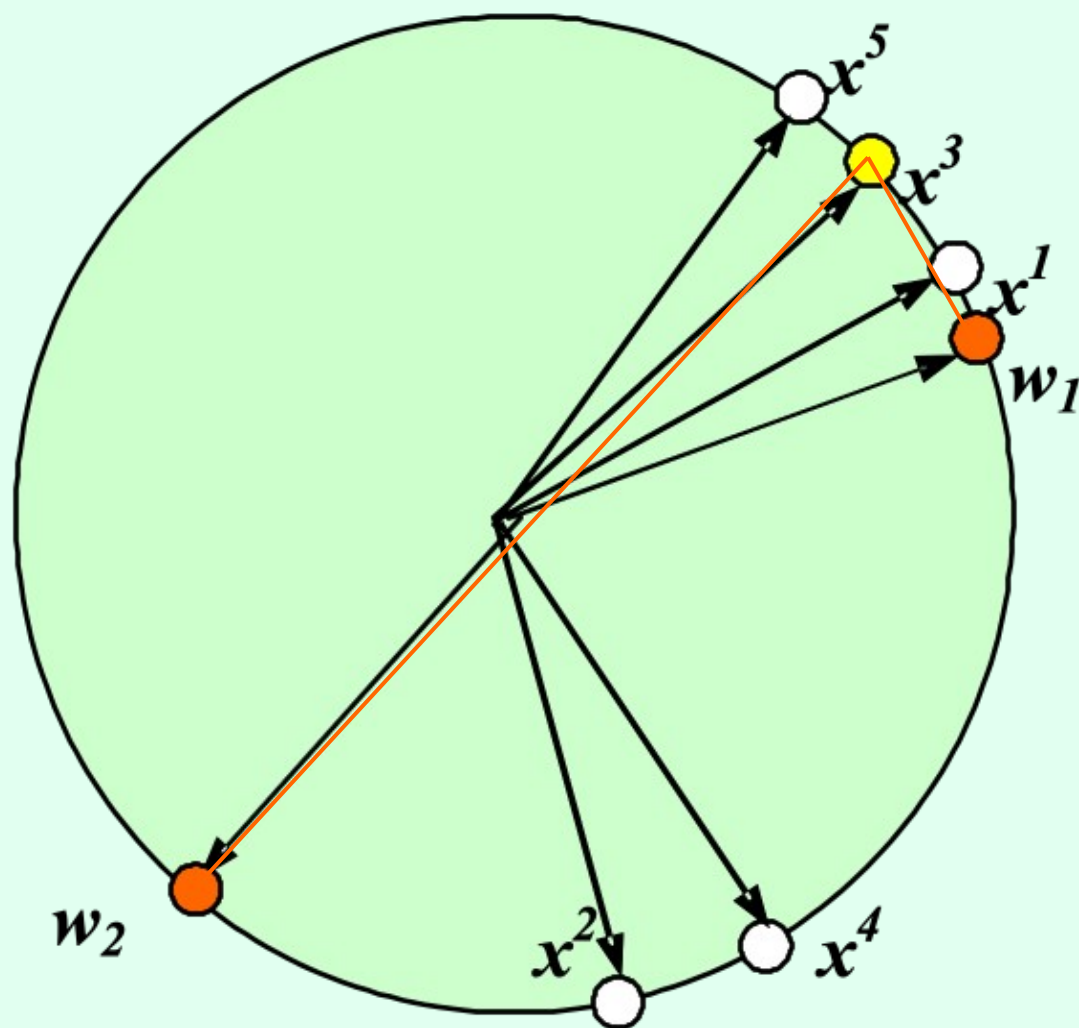
训练次数	W	W
1	18.43°	-180°
2	-30.8°	-180°
3	7°	-180°
4	-32°	-180°
5	11°	-180°
6	24°	-180°
7	24°	-130°
8	34°	-130°
9	34°	-100°
10	44°	-100°
11	40.5°	-100°
12	40.5°	-90°
13	43°	-90°
14	43°	-81°
15	47.5°	-81°
16	42°	-81°
17	42°	-80.5°
18	43.5°	-80.5°
19	43.5°	-75°
20	48.5°	-75°



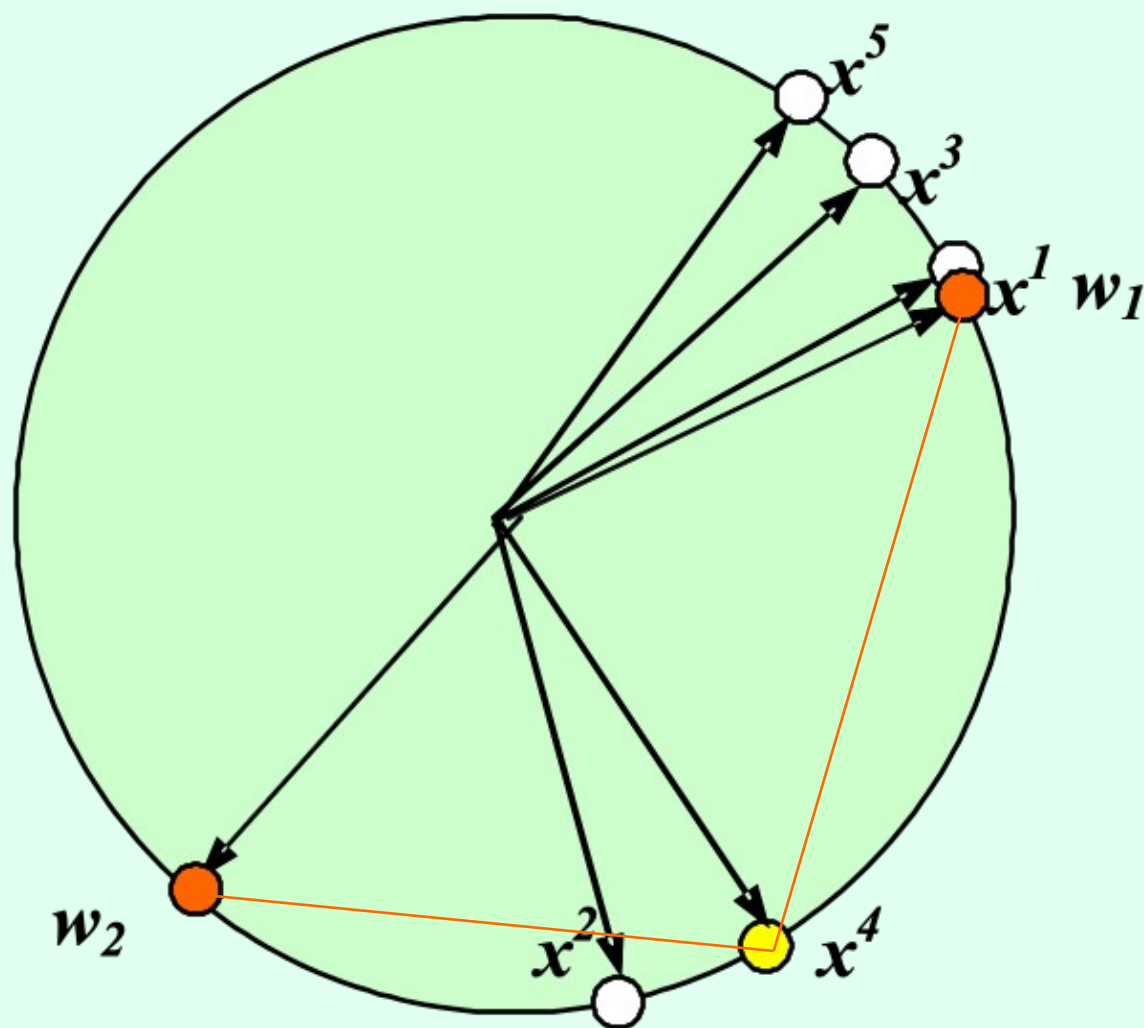
训练次数	w	w
1	18.43°	-180°
2	-30.8°	-180°
3	7°	-180°
4	-32°	-180°
5	11°	-180°
6	24°	-180°
7	24°	-130°
8	34°	-130°
9	34°	-100°
10	44°	-100°
11	40.5°	-100°
12	40.5°	-90°
13	43°	-90°
14	43°	-81°
15	47.5°	-81°
16	42°	-81°
17	42°	-80.5°
18	43.5°	-80.5°
19	43.5°	-75°
20	48.5°	-75°



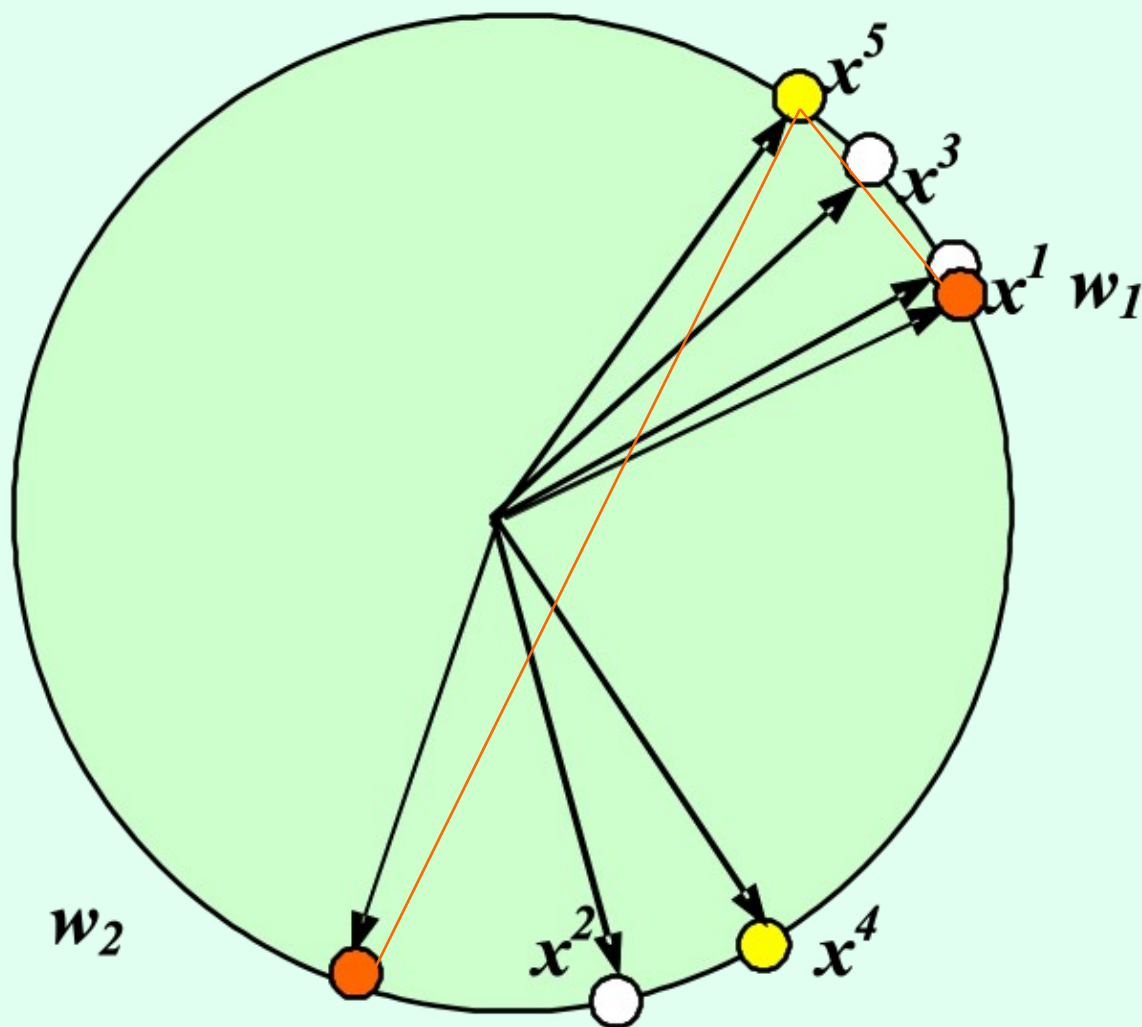
训练次数	w	w
1	18.43°	-180°
2	-30.8°	-180°
3	7°	-180°
4	-32°	-180°
5	11°	-180°
6	24°	-180°
7	24°	-130°
8	34°	-130°
9	34°	-100°
10	44°	-100°
11	40.5°	-100°
12	40.5°	-90°
13	43°	-90°
14	43°	-81°
15	47.5°	-81°
16	42°	-81°
17	42°	-80.5°
18	43.5°	-80.5°
19	43.5°	-75°
20	48.5°	-75°



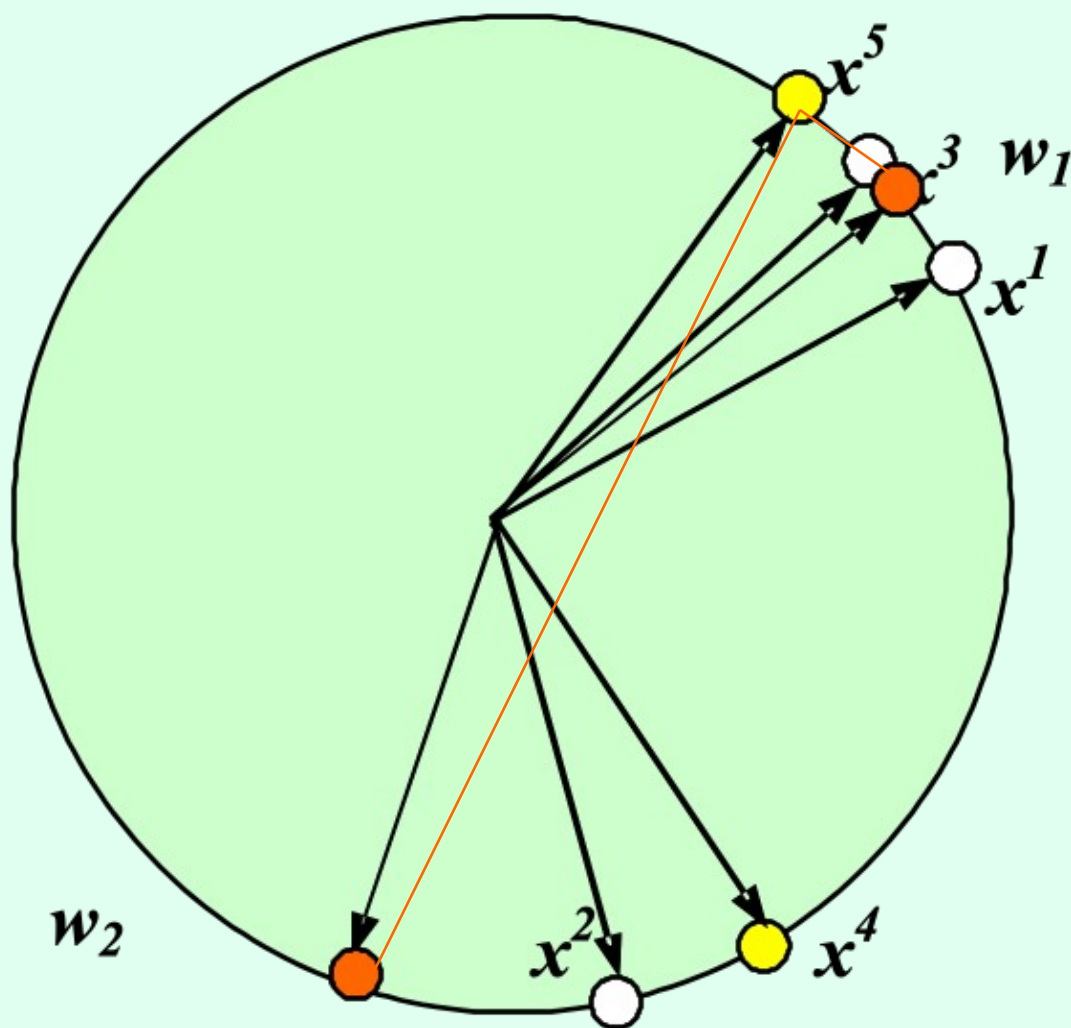
训练次数	w	w
1	18.43°	-180°
2	-30.8°	-180°
3	7°	-180°
4	-32°	-180°
5	11°	-180°
6	24°	-180°
7	24°	-130°
8	34°	-130°
9	34°	-100°
10	44°	-100°
11	40.5°	-100°
12	40.5°	-90°
13	43°	-90°
14	43°	-81°
15	47.5°	-81°
16	42°	-81°
17	42°	-80.5°
18	43.5°	-80.5°
19	43.5°	-75°
20	48.5°	-75°



训练次数	w	w
1	18.43°	-180°
2	-30.8°	-180°
3	7°	-180°
4	-32°	-180°
5	11°	-180°
6	24°	-180°
7	24°	-130°
8	34°	-130°
9	34°	-100°
10	44°	-100°
11	40.5°	-100°
12	40.5°	-90°
13	43°	-90°
14	43°	-81°
15	47.5°	-81°
16	42°	-81°
17	42°	-80.5°
18	43.5°	-80.5°
19	43.5°	-75°
20	48.5°	-75°



训练次数	w	w
1	18.43°	-180°
2	-30.8°	-180°
3	7°	-180°
4	-32°	-180°
5	11°	-180°
6	24°	-180°
7	24°	-130°
8	34°	-130°
9	34°	-100°
10	44°	-100°
11	40.5°	-100°
12	40.5°	-90°
13	43°	-90°
14	43°	-81°
15	47.5°	-81°
16	42°	-81°
17	42°	-80.5°
18	43.5°	-80.5°
19	43.5°	-75°
20	48.5°	-75°



训练次数	w	w
1	18.43°	-180°
2	-30.8°	-180°
3	7°	-180°
4	-32°	-180°
5	11°	-180°
6	24°	-180°
7	24°	-130°
8	34°	-130°
9	34°	-100°
10	44°	-100°
11	40.5°	-100°
12	40.5°	-90°
13	43°	-90°
14	43°	-81°
15	47.5°	-81°
16	42°	-81°
17	42°	-80.5°
18	43.5°	-80.5°
19	43.5°	-75°
20	48.5°	-75°

4.2 自组织特征映射神经网络 (Self-Organizing feature Map)

- 1981 年芬兰 Helsinki 大学的 T.Kohonen 教授提出一种自组织特征映射网，简称 SOM 网，又称 Kohonen 网。
- Kohonen 认为：一个神经网络接受外界输入模式时，将会分为不同的对应区域，各区域对输入模式具有不同的响应特征，而且这个过程是自动完成的。自组织特征映射正是根据这一看法提出来的，其特点与人脑的自组织特性相类似。



SOM 网的生物学基础

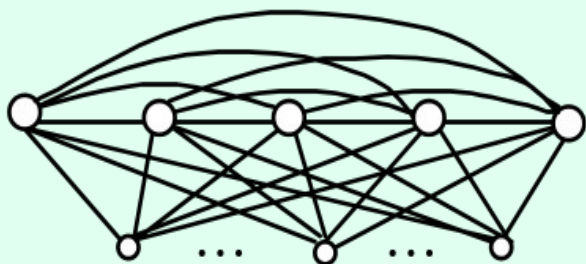
生物学研究的事实表明，在人脑的感觉通道上，神经元的组织原理是**有序排列**。因此当人脑通过感官接受外界的特定时空信息时，大脑皮层的**特定区域兴奋**，而且类似的外界信息在对应区域是**连续映象**的。

对于某一图形或某一频率的特定兴奋过程，神经元的有序排列以及对外界信息的连续映象是自组织特征映射网中竞争机制的生物学基础。

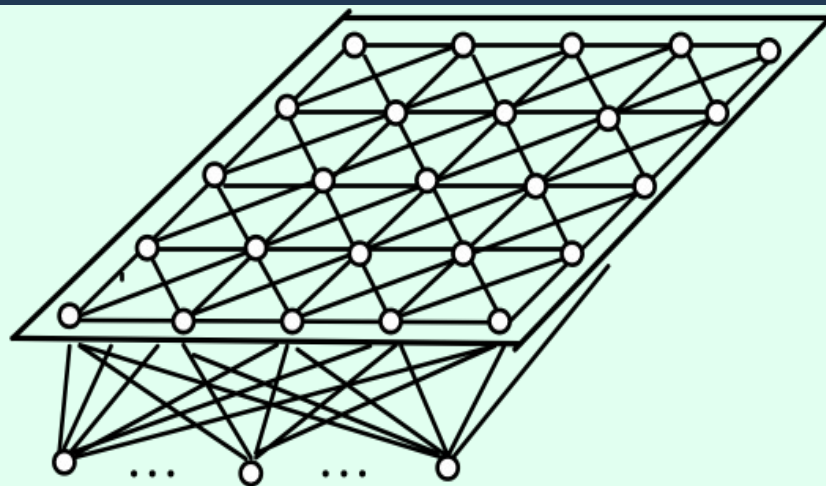


SOM网的拓扑结构

SOM网共有两层，输入层模拟感知外界输入信息的视网膜，输出层模拟做出响应的大脑皮层。



(a)一维线阵

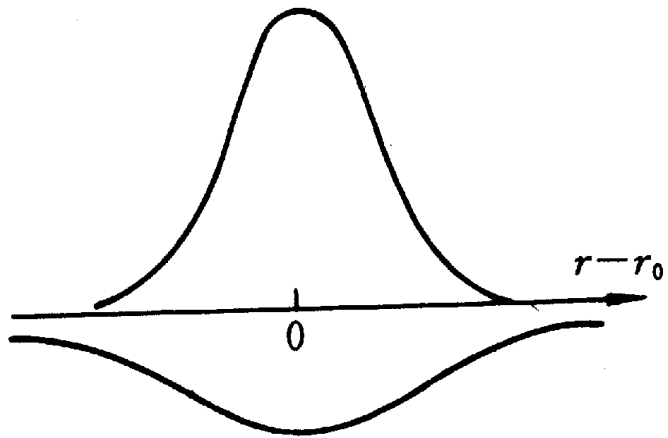


(b)二维平面线阵

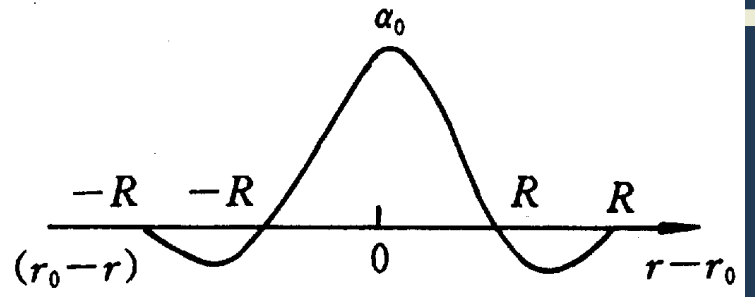
SOM 网的权值调整域

SOM 网的获胜神经元对其邻近神经元的影响是由近及远，由兴奋逐渐转变为抑制，因此其学习算法中不仅获胜神经元本身要调整权向量，它周围的神经元在其影响下也要程度不同地调整权向量。这种调整可用以下函数表示：

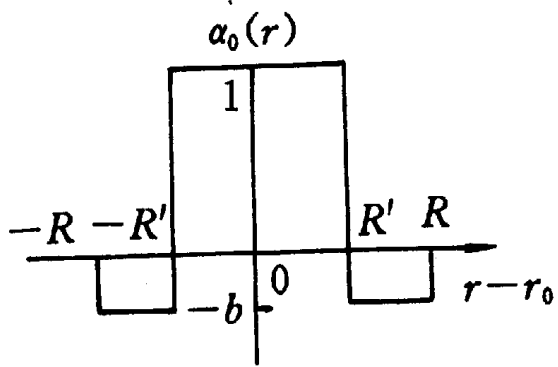




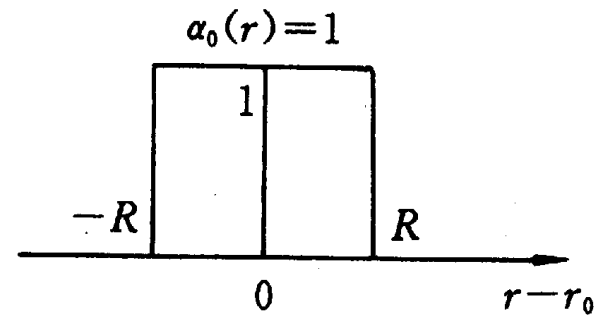
(a)



(b)



(c)



(d)



SOM 网的权值调整域

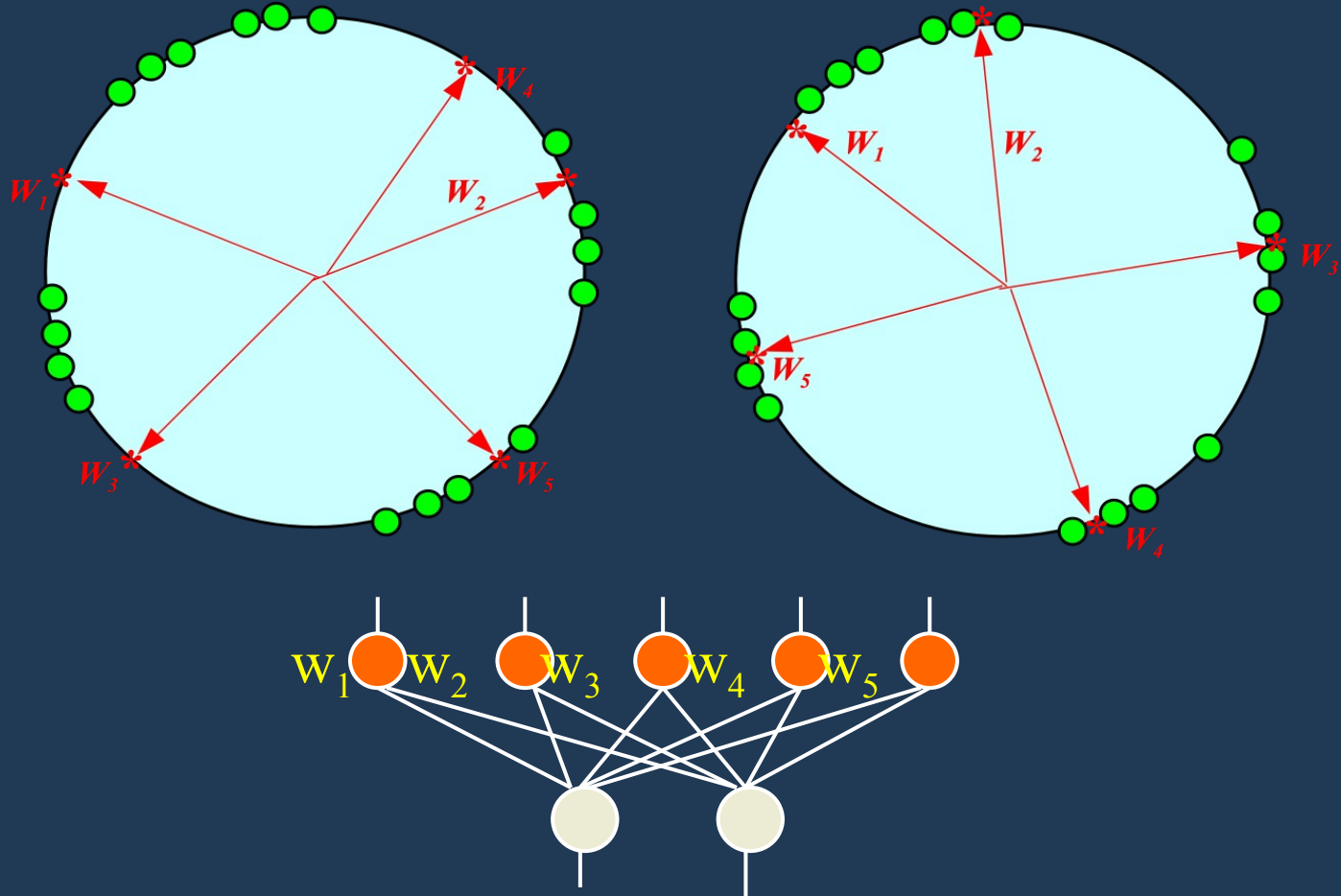
以获胜神经元为中心设定一个邻域半径，该半径圈定的范围称为**优胜邻域**。在 SOM 网学习算法中，优胜邻域内的所有神经元均按其离开获胜神经元的距离远近不同程度地调整权值。

优胜邻域开始定得很大，但其大小随着训练次数的增加不断收缩，最终收缩到半径为零。



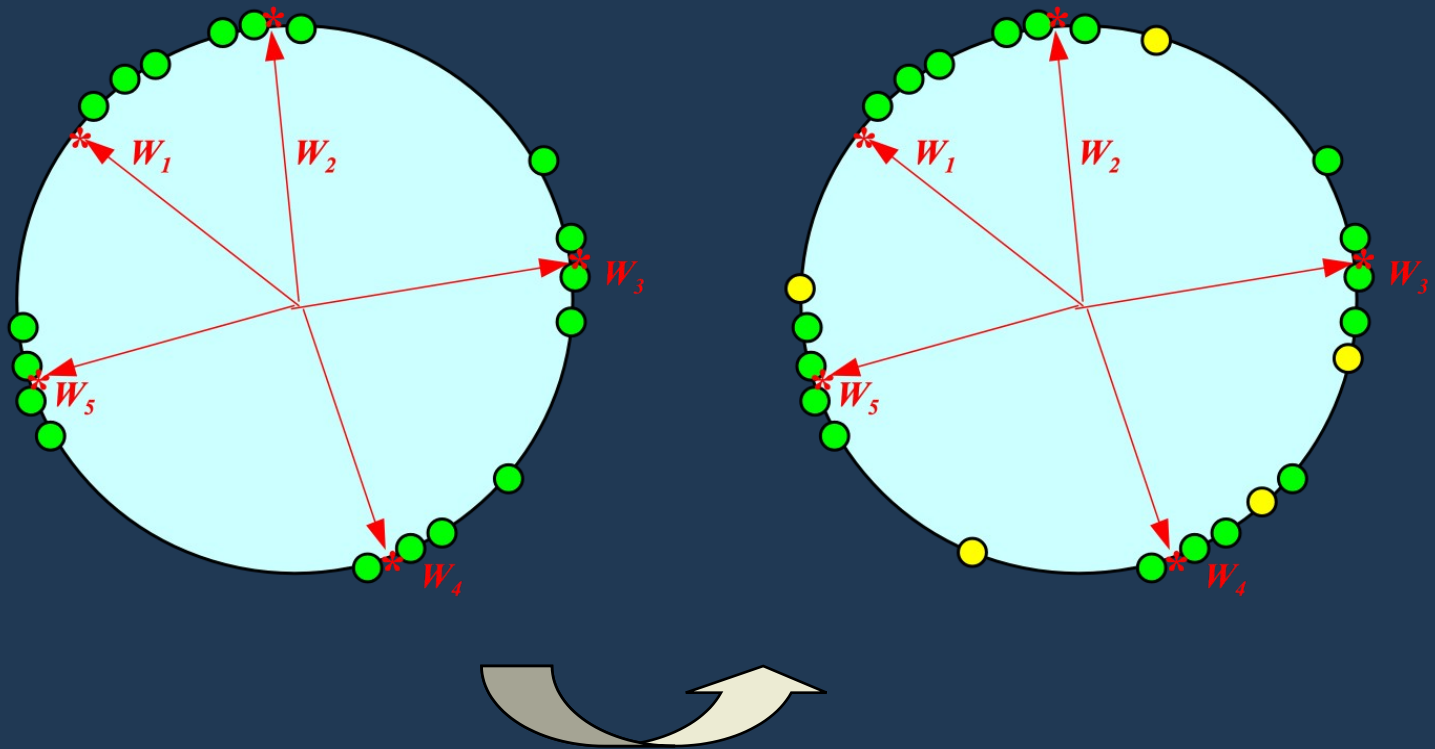
SOM网的运行原理

□ 训练阶段



SOM网的运行原理

□ 工作阶段



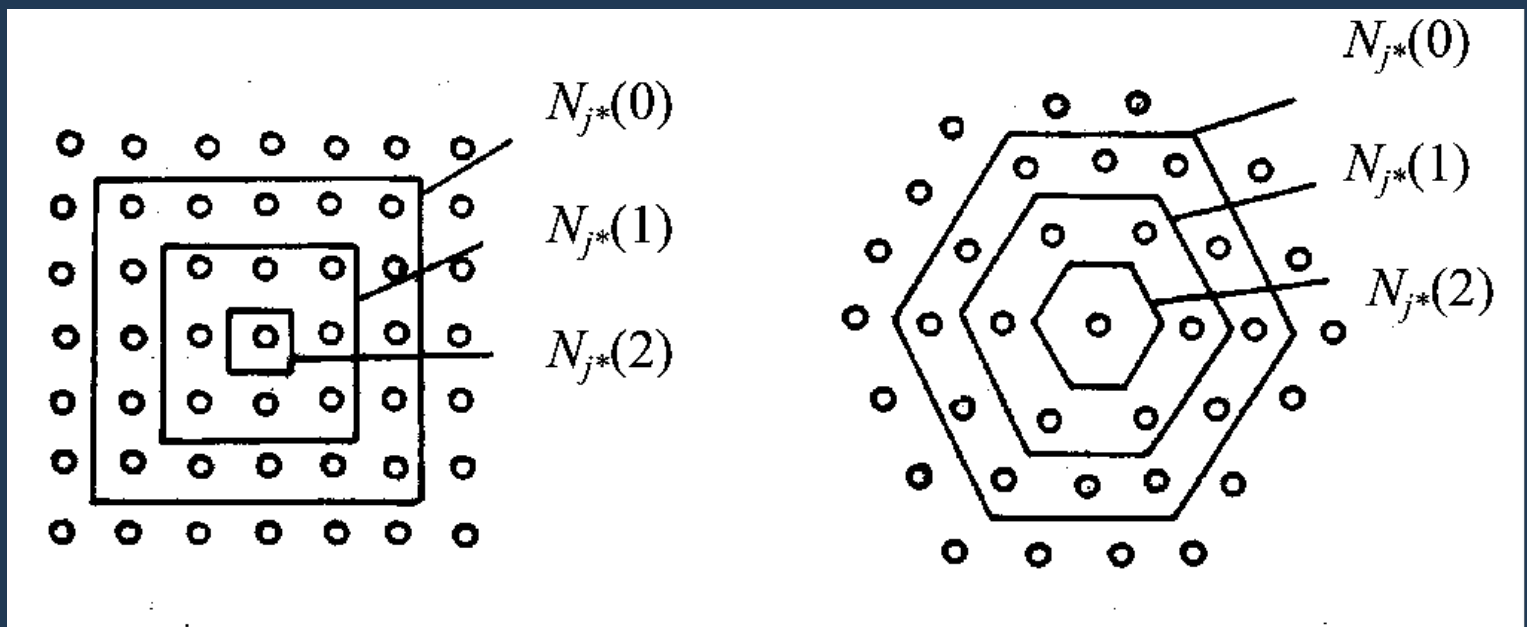
SOM网的学习算法

Kohonen 学习算法

- (1) 初始化 对输出层各权向量赋小随机数并进行归一化处理，得到 $\hat{\mathbf{W}}_j, j=1,2,\dots,m$ ；建立初始优胜邻域 $N_{j^*}(0)$ ；学习率 η 赋初始值。
- (2) 接受输入 从训练集中随机选取一个输入模式并进行归一化处理，得到 $\hat{\mathbf{X}}^p, p \in \{1,2,\dots,P\}$ 。
- (3) 寻找获胜节点 计算 $\hat{\mathbf{X}}^p$ 与 $\hat{\mathbf{W}}_j$ 的点积， $j=1,2,\dots,m$ ，从中选出点积最大的获胜节点 j^* 。
- (4) 定义优胜邻域 $N_{j^*}(t)$ 以 j^* 为中心确定 t 时刻的权值调整域，一般初始邻域 $N_{j^*}(0)$ 较大，训练过程中 $N_{j^*}(t)$ 随训练时间逐渐收缩。

SOM网的学习算法

Kohonen 学习算法



SOM网的学习算法

Kohonen 学习算法

(5) 调整权值 对优胜邻域 $N_{j^*}(t)$ 内的所有节点调整权值：

$$w_{ij}(t+1) = w_{ij}(t) + \eta(t, N)[x_i^p - w_{ij}(t)]$$

式中， $\eta(t, N)$ 是训练时间 t 和邻域内第 j 个神经元与获胜神经元 j^* 之间的拓扑距离 N 的函数，该函数一般有以下规律：

$$t \uparrow \rightarrow \eta \downarrow, N \uparrow \rightarrow \eta \downarrow$$

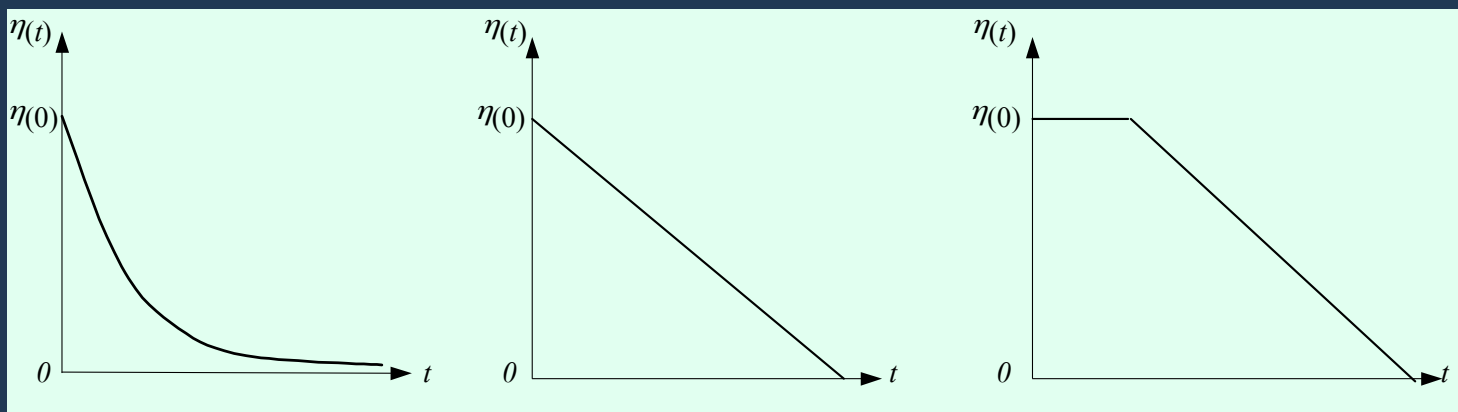


SOM网的学习算法

Kohonen 学习算法

(5) 调整权值

$$\eta(t, N) = \eta(t)e^{-N}$$



(6) 结束检查 学习率是否衰减到零或某个预定的正小数？



Kohonen 学习算法程序流程

