

6.4 自组织映射的无指导聚类

- 在缺少学习所需的先验知识时，需要神经网络具有自学习能力。
- Kohonen自组织映射图（Self-Organizing maps, 简称SOMs）
 - Teuvo Kohonen（1982）提出的。
 - 具有自学习功能，基于生理学和脑科学的研究成果。
 - 形式化神经网络的无指导聚类，形成了著名的Kohonen神经网络。
 - Kohonen 网络支持两层结构
 - 输入层包含输入向量节点；
 - 输入层节点与所有输出层节点具有加权连接。
 - 通过某种规则，使得在训练后，一个区域的所有节点对某种输入具有类似的输出，并且簇的概率分布与输入模式的概率分布相接近。
 - 输出层可以采取任何格式，但一般被组织为二维网格。

Kohonen网络

- Kohonen网络的功能就是通过自组织方法，用大量的样本训练数据来调整网络的权值，使得最后网络的输出能够反映样本数据的分布情况。

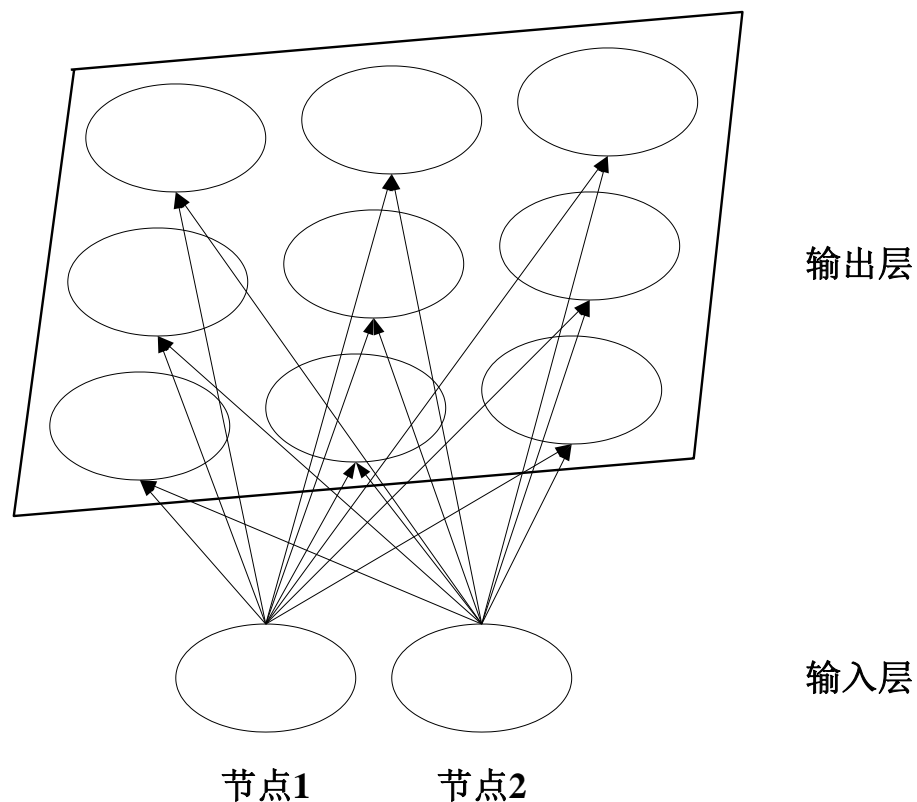
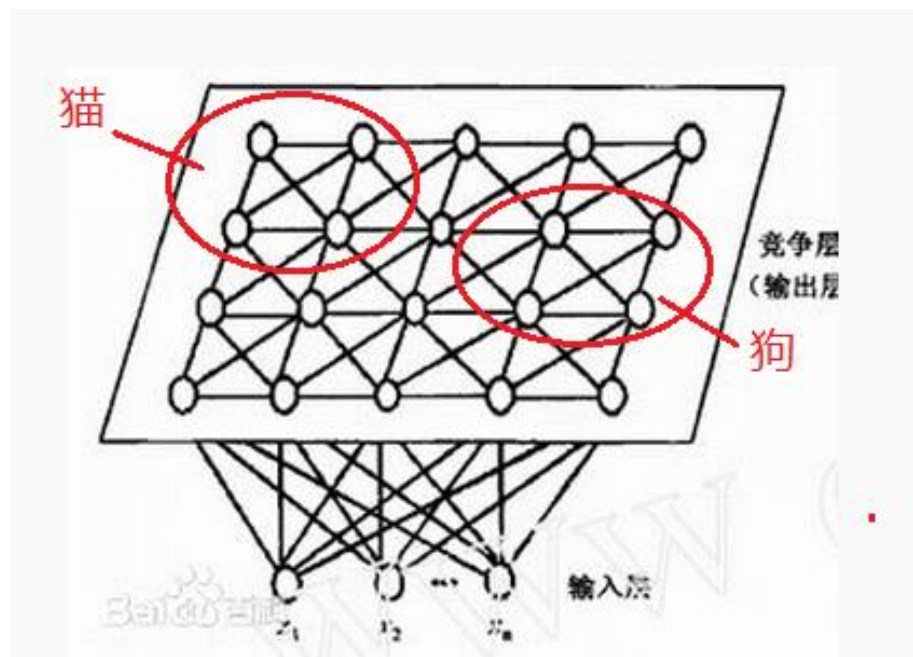


图6.3 具有2两个输入层节点的3×3 Kohonen网络

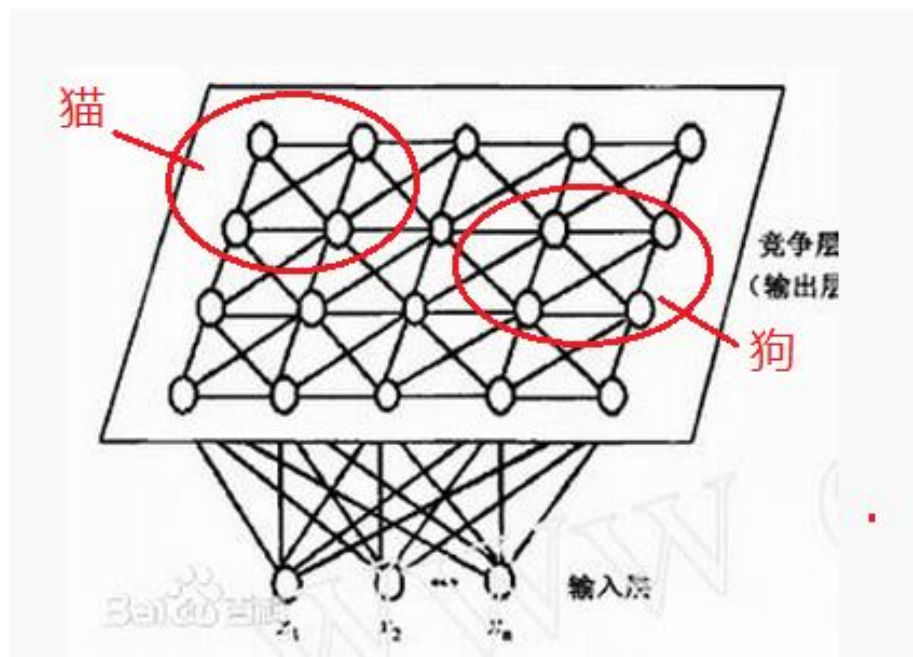
Kohonen网络

- 自组织神经网络是两层结构，包括输入层和竞争层。为什么要用这种结构？
- 因为Kohonen根据生理学的发现，认为神经网络在接受外界输入时，将会分成不同的区域，不同的区域对不同的模式具有不同的响应特征。如下图，人对猫和狗的概念储存在不同区域。



Kohonen网络

- 训练：输入层是输入样本的特征，通过不断地将所有样本数据反复输入网络进行训练，最终权值将会趋于稳定。学习方法为竞争学习。
- 测试：将新的样本输入，比如"猫"的样本，那么在图中猫的区域内容点的值输出最大，说明正确识别了"猫"。



Kohonen网络

- 假如输入的数据样本在N为空间上均匀分布（样本点在高维空间呈格子状），初始化时的权重 w 为随机，那么输出层也是随机的。如图(1)。通过不断的迭代修改权值 w ，输出层慢慢也呈现出均匀格子结构，如(1)-(8)变化。这个例子说明SOM神经网络可以把高维空间映射到平面，从而实现降维。

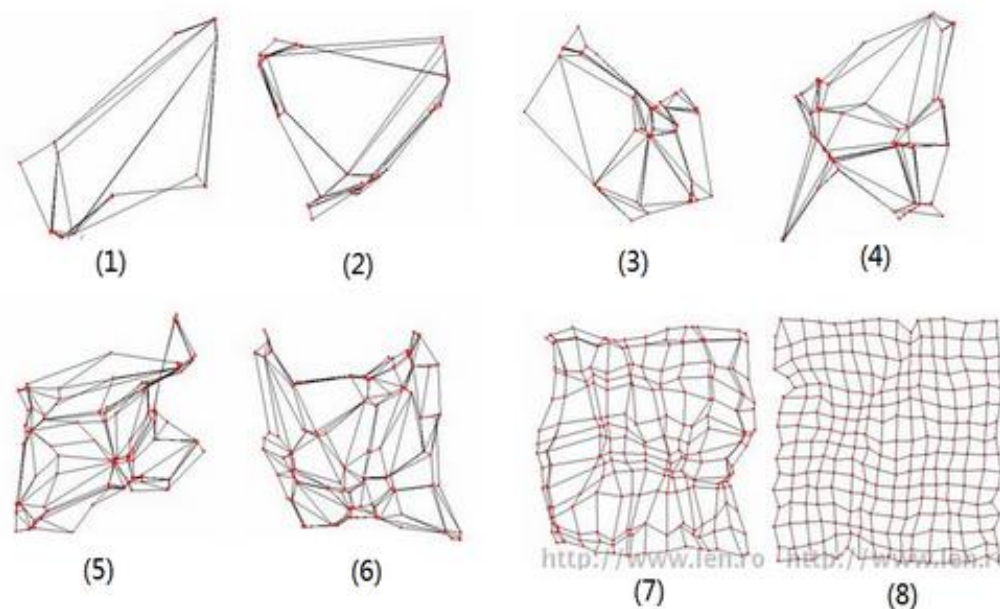


图 (1)至(8)为不断迭代权值 w ，计算出输出层的结果

网络自组织算法

- 网络开始训练时，某个输出结点能对某一类模式作出特别的反应，以代表该模式类，但这里规定二维平面上相邻的结点能对实际模式分布中相近的模式类作出特别的反应。当某类数据模式输入时，对其某一输出结点给予最大的刺激，以指示该类模式的所属区域，而同时对获胜结点周围的一些结点给予较大的刺激。
- 当输入模式从一个模式区域移到相邻的模式区域时，二维平面上的获胜结点也从原来的结点移到其相邻的结点。因此，从Kohonen网络的输出状况，不但能判断输入模式所属的类别并使输出结点代表某一类模式，还能够得到整个数据区域的大体分布情况，即从样本数据中抓到所有数据分布的大体本质特性。

网络自组织算法

- 为了能使二维输出平面上相邻的输出结点对相近的输入模式类作出特别反应，在训练过程中需定义获胜结点的邻域结点。
- 假设本次获胜结点为 N_j ，它在 t 时刻的邻域结点用 $NE_j(t)$ 表示， $NE_j(t)$ 包含以结点 N_j 为中心而距离不超过某一半径的所有结点。随着训练的进行， $NE_j(t)$ 的半径逐渐缩小，最后只包含获胜结点 N_j 自身，即在训练初始阶段，不但对获胜的结点做权值调整，也对其较大范围内的几何邻接结点做相应的调整，而随着训练过程的继续，与输出结点相连的权向量也越来越接近其代表的模式类。
- 此时，对获胜结点进行较细微的权值调整时，只对其几何邻域接近的结点（包括其自身）进行相应调整，直到最后只对获胜的结点本身做细微的权值调整。在训练结束后，几何上相近的输出结点所连接的权向量既有联系（类似性）又互相区别，从而保证对于某一类输入模式，获胜结点能做出最大响应，而相邻结点做出较大响应。几何上相邻的结点代表特征上相近的模式类别。

自组织神经网络：Kohonen网络训练算法

- (1) 权连接初始化，对所有从输入结点到输出结点的连接权值赋予随机的小数，置时间计数 $t = 0$ 。
- (2) 对网络输入模式
- (3) 计算输入 $x_k = (x_1, x_2, \dots, x_n)$ 与全部输出结点所连接向量的距离：

$$x_k$$

- (4) 具有最小距离的结点竞争获胜：
$$d_j = \sum_{i=1}^n (x_i^k - w_{ij})^2, i \in \{1, 2, \dots, n\}, j \in \{1, 2, \dots, m\}$$

$$N_{j*}$$

$$d_{j*} = \min_{j \in \{1, 2, \dots, m\}} \{d_j\}$$

自组织神经网络：Kohonen网络训练算法

- (5) 调整输出结点 j^* 所连接的权值及 N_{j^*} 几何邻域 $NE_{j^*}(t)$ 内的结点所连权值：

- $\Delta w_{ij} = \eta(t)(x_i^k - w_{ij}), N_j \in NE_{j^*}(t), i \in \{1, 2, \dots, n\}$
- (6) 若还有输入样本数据, $t = t+1$, 转 (2)。

- 在算法中, $\eta(t)$ 是一种可变学习速度, 随时间而衰减, 表示随着训练过程的进行, 权值调整的幅度越来越小, 以使竞争获胜结点所连的权向量代表模式的本质属性。 $NE_{j^*}(t)$ 也随时间而收缩, 最后当 t 充分大时, $NE_{j^*}(t) = \{j^*\}$, 即只训练获胜结点本身以实现 Kohonen 网络的自组织特征映射功能。
- $\eta(t)$ 和 $NE_{j^*}(t)$ 都有许多不同的形式, 在具体训练过程中可根据不同的数据分布进行选择和设计。

$$\eta(t) \quad NE_{j^*}(t)$$

【例6.5】

使用图6.3所示的神经网络结构和输入实例举例说明自组织学习方法。

具有3个输出层节点的Kohonen网络

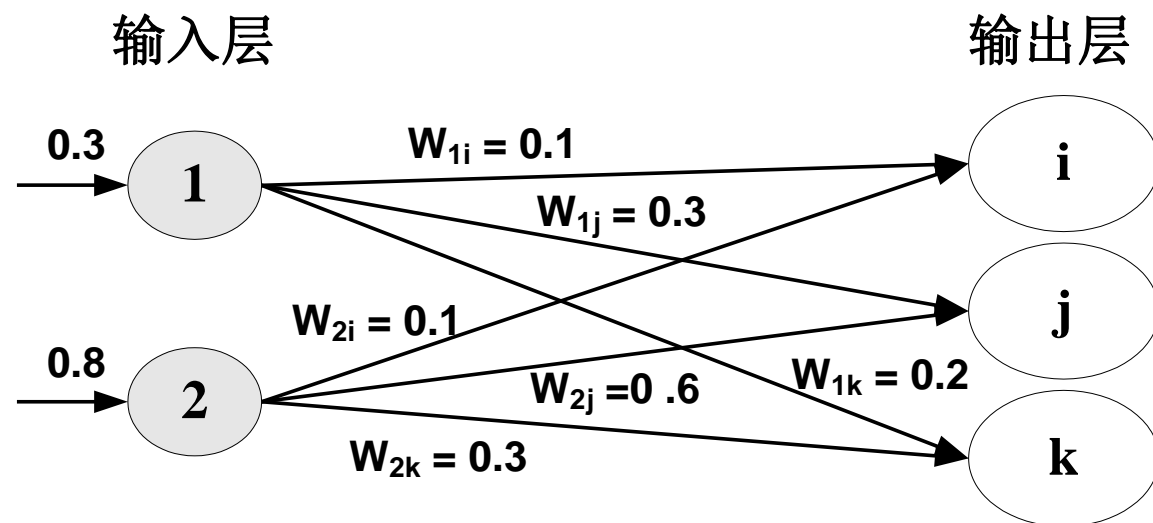


图6.4 具有3个输出层节点的Kohonen网络

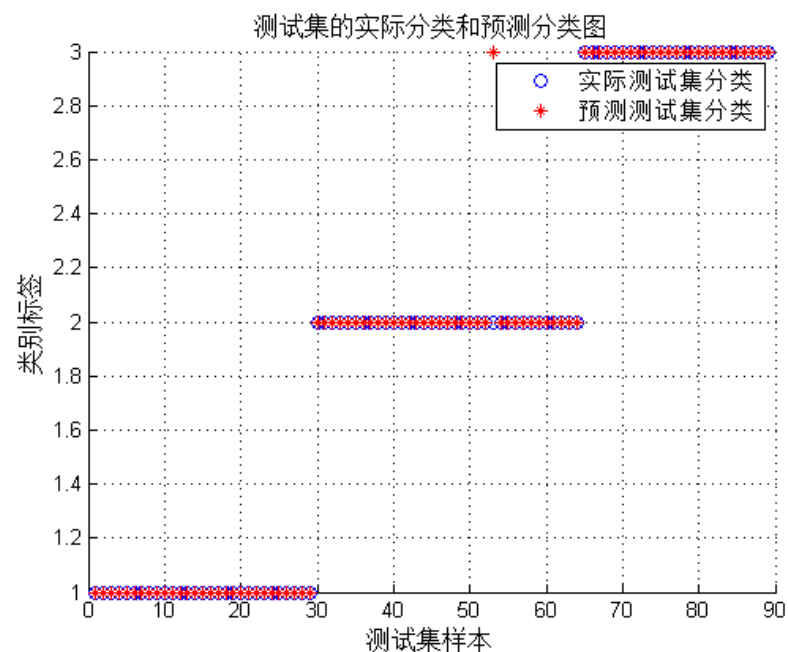
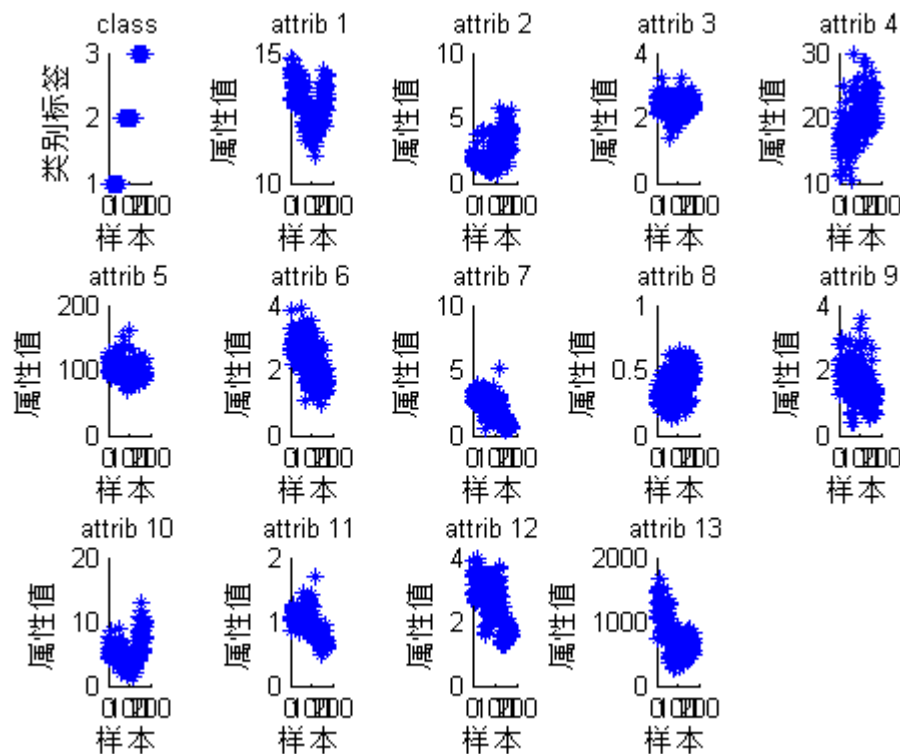
步骤1: 找出获胜输出节点

步骤2: 校正权值向量, 奖赏获胜输出节点

步骤3: 完成聚类

【实例1】葡萄酒分类识别

- 数据：意大利同一产地的三种不同品种的葡萄酒的13种化学成分。
- 应用：通过学习建立神经网络对未知葡萄酒进行分类。

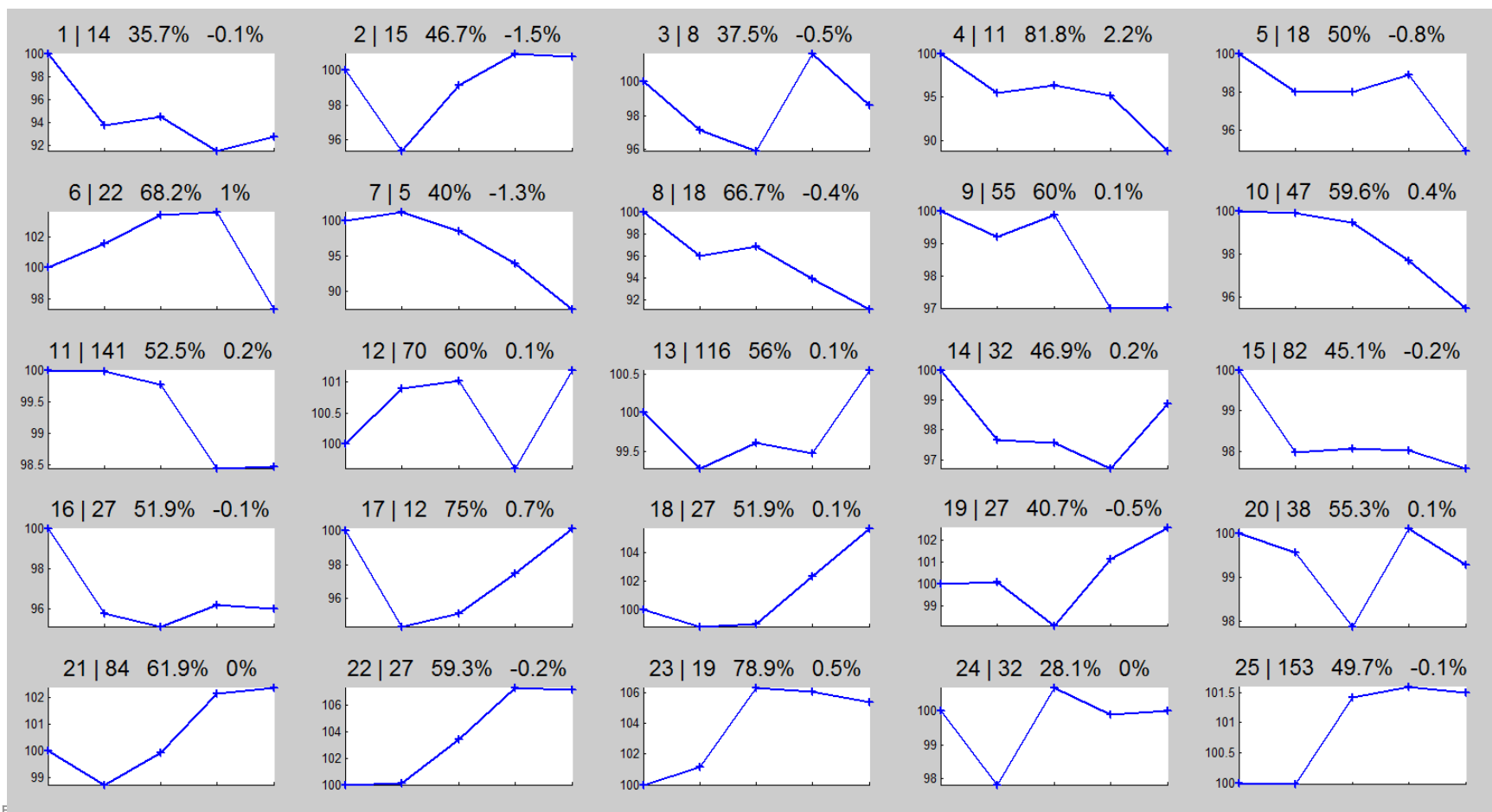


【实例2】通过识别K线形态预测股市

- 在证券市场中，最基本的元素是k线图。K线有很多组合形态，比如经典W底，多方炮之类，种类繁多，高手也无法完全识别。
- 能否用神经网络来自主学习并识别K线形态呢？
- 为简单起见，这里限定考虑4个交易日构成的形态，且不考虑成交量，数据为2006年至2016年的上证指数每日涨跌幅，共2550组数据，共识别出50种K线形态（可以人为选择输出的数量，数量少了，不够精细，误差会很大；数量多了，冗余会多。）

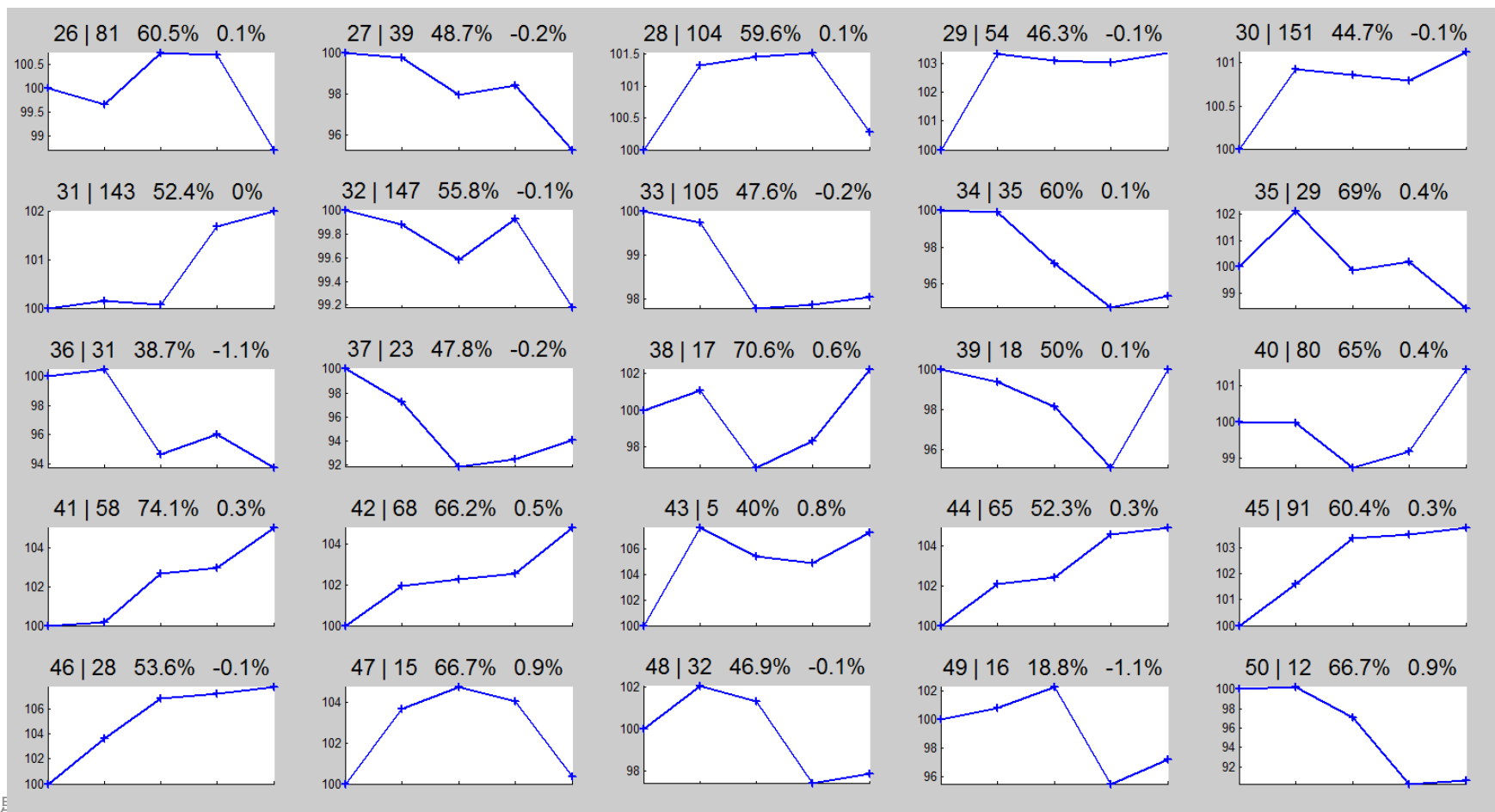
【实例2】通过识别K线形态预测股市

- 1-25 号形态。标题含义：序号 | 出现次数 次日上涨概率 平均涨幅



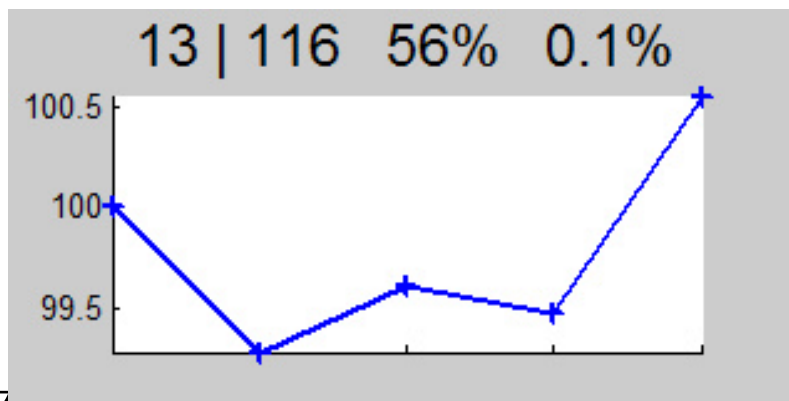
【实例2】通过识别K线形态预测股市

- 26-50号形态。标题含义：序号 | 出现次数 次日上涨概率 平均涨幅



【实例2】通过识别K线形态预测股市

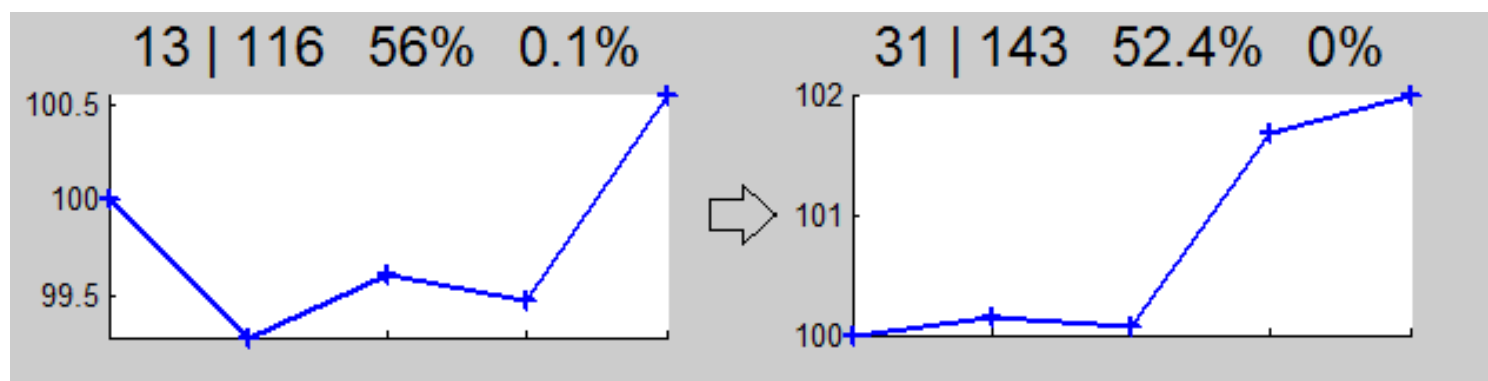
- 有了所有的基本形态后，行情数据便被转换成了一系列的状态。如图1中的13号即是大家所熟知的W底：



- 可以看到，样本数据中共出现该形态116次，次日上涨概率为56%，平均涨幅为0.1%

【实例2】 通过识别K线形态预测股市

- 而接下来可能的一种转换模式为：
- 13 号 转化为 31号，即：



【实例2】 通过识别K线形态预测股市

- 更近一步程序可以得到各状态出现后次日的胜率和期望收益等参数，而有了所需的参数，可以根据凯利公式计算出科学的仓位，以此来均衡收益与风险。
- 凯利公式是利用预期收益，亏损，即胜率等信息计算仓位的方法。

【实例2】通过识别K线形态预测股市

- 举个例子，我提取了09年到14年8月这段熊市的数据进行网络训练，然后用训练好的程序对之后一段时间的正常行情进行预测。
- 蓝色是真实数据，红色是预测值，可以看到中后部的一段，有很大的误差，其对应的是14年8月29日到9月05日的六连阳，可以这么理解，即这几天的走势完全超出预期，往往是特定事件导致的结果，需要深刻思考其产生的原因。

