南开大学

硕士学位论文

基于SOM的可视化聚类研究

姓名：赵辉

申请学位级别：硕士

专业：控制理论与控制工程

指导教师：王秀峰

20070501

摘要

随着计算机应用的普及，信息系统产生的数据量日益增大，迫切需要高效的数据挖掘工具，从大量原始数据中寻找有价值的知识模式。聚类分析是数据挖掘的重要工具之一。如何正确处理维度达到数百、数千的数据集合，从高维数据集中寻找潜在的、自然存在的聚类簇，这是当前聚类分析研究的热点。

本文针对聚类分析的热点和难点问题—高维海量聚类展开研究，提出了基于SOM的可视化聚类方法。主要工作有：

（1）对基于非监督学习神经网络的自组织聚类方法进行了深入研究。对其工作原理、聚类特点、评价准则以及存在问题进行了深入探讨。

（2）深入探讨了利用描述相关性的U矩阵的信息，改进聚类性能和可视化特性的可视化聚类方法；通过用经典和标准数据库数据仿真，验证了可视化聚类方法的有效性。并在入侵检测中进行了应用研究。

关键字： SOM特征映射　聚类　可视化　U矩阵

Abstract

With the wide usage of information technology, data generated from varies information systems become more and more, and the higher efficiency data mining tools was needed to fmd valuable knowledge patterns. Clustering analysis is a important method in data mining. It is a discovery process that groups a set of data such that the intra-cluster similarity is maximized and the inter-cluster similarity is minimized. Clustering of data in a large dimension space is of a great interest in many data mining applications. With high-dimensionality data sets, how to find the latentand nature clusters is more difficult and need to be resolved

In this thesis, a famous unsupervised learning neural network, Self-Organizing Map (SOM) was introduced that is proposed by Kohonen. Essentially, SOM is a feature projection, which can be applied in revealing the nonlinear statistical relationship hidden in the high-dimensional data.

The main contribution of this thesis consists of two parts. The first one is the use of the existing and development of novel SOM-based methods for clustering. The second contribution is new methods and measures for visualization of data based on SOM are proposed.

Keyword: Self-organizing map Clustering, Visualization U-matrix

南开大学学位论文电子版授权使用协议

（请将此协议书装订于论文首页）

论文《　　》系本人在南开大学工作和学习期间创作完成的作品，并已通过论文答辩。

本人系本作品的唯一作者（第一作者），即著作权人。现本人同意将本作品收录于“南开大学博硕士学位论文全文数据库”。本人承诺：已提交的学位论文电子版与印刷版论文的内容一致，如因不同而引起学术声誉上的损失由本人自负。

本人完全了解《南开大学图书馆关于保存、使用学位论文的管理办法》。同意南开大学图书馆在下述范围内免费使用本人作品的电子版：

本作品呈交当年，在校园网上提供论文目录检索、文摘浏览以及论文全文部分浏览服务（论文前16页）。公开级学位论文全文电子版于提交1年后，在校园网上允许读者浏览并下载全文。

注：本协议书对于“非公开学位论文”在保密期限过后同样适用。

院系所名称：

作者签名：

学号：

日期：　　年　　月　　日

南开大学学位论文版权使用授权书

本人完全了解南开大学关于收集、保存、使用学位论文的规定，同意如下各项内容：按照学校要求提交学位论文的印刷本和电子版本；学校有权保存学位论文的印刷本和电子版，并采用影印、缩印、扫描、数字化或其它手段保存论文；学校有权提供目录检索以及提供本学位论文全文或者部分的阅览服务；学校有权按有关规定向国家有关部门或者机构送交论文的复印件和电子版；在不以赢利为目的的前提下，学校可以适当复制论文的部分或全部内容用于学术活动。

学位论文作者签名：

年　月　日

经指导教师同意，本学位论文属于保密，在　　年解密后适用本授权书。

指导教师签名：		**学位论文作者签名：**	
解　密　时　间：	年　　月　　日		

各密级的最长保密年限及书写格式规定如下：

内部　5 年（最长 5 年，可少于 5 年）
秘密★10 年（最长 10 年，可少于 10 年）
机密★20 年（最长 20 年，可少于 20 年）

南开大学学位论文原创性声明

本人郑重声明：所呈交的学位论文，是本人在导师指导下，进行研究工作所取得的成果。除文中已经注明引用的内容外，本学位论文的研究成果不包含任何他人创作的、已公开发表或者没有公开发表的作品的内容。对本论文所涉及的研究工作做出贡献的其他个人和集体，均已在文中以明确方式标明。本学位论文原创性声明的法律责任由本人承担。

学位论文作者签名：

年　　月　　日

第一章 绪论

第一节 选题背景与意义

1.1.1 聚类简介

近几年来，人们利用信息技术生产和搜集数据的能力大幅度提高，千千万万个数据库被用于商业管理、政府办公科学研究和工程开发等等。要想使数据真正成为一个公司的资源，只有充分利用它为公司自身的业务决策和战略发展服务才行；否则大量的数据可能成为包袱甚至成为垃圾。因此，面对人们被数据淹没却饥饿于知识的挑战，数据挖掘和知识发现技术应运而生，并得以蓬勃发展，越来越显示出其强大的生命力。

数据挖掘(Data Mining)就是从大量的、不完全的、有噪声的、模糊的、随机的数据中，提取隐含在其中的、人们事先不知道的、但又是潜在有用的信息和知识的过程。人们把原始数据看作是形成知识的源泉，就像从矿石中采矿一样。原始数据可以是结构化的，如关系数据库中的数据；也可以是半结构化的，如文本图形、图像数据，甚至是分布在网络上的异构型数据；发现知识的方法可以是数学的，也可以是非数学的；可以是演绎的，也可以是归纳的；挖掘出的知识可以被用于信息管理、查询优化、决策支持、过程控制等，还可以用于数据自身的维护。因此，数据挖掘是一门很广义的交叉学科，涉及人工智能技术、统计技术与数据库技术等多种技术。它汇聚了不同领域的研究者，尤其是数据库、人工智能、数理统计、可视化、并行计算等方面的学者和工程技术人员。

聚类(Clustering)是数据挖掘中一种重要的挖掘任务和挖掘方法，它从数据库中寻找数据间的相似性并依此对数据进行分类，使得不同类中的数据尽可能相异，而同一类中的数据尽可能相似即“物以类聚”，从而优化大规模数据库的查询和发现数据中隐含的有用信息或知识。用数学语言表达就是，如果将 n 个样本 $x_1 \cdots x_n$ 的数据集 X 聚集成 c 个子类 $x_1 \cdots x_c$，则要求 $x_1 \cdots x_c$ 满足：

$$x_1 \cup x_2 \cup x_3 \cdots \cup x_c = X$$

$$x_i \cap x_j = \Phi, 1 \le i \ne j \le c$$

数据聚类在很多领域中有着广泛的应用，如模式识别、图象处理、数据压缩、空间数据分析、市场研究、WWW 上（WWW 上的文本分类；对 WEB 的日志数据聚类以后发现相似的访问模式）等等。迄今为止人们提出了很多聚类算法，例如分割的方法，层次的方法，基于密度的方法，基于网格的方法和基于模型的方法等，参见图 1.1。

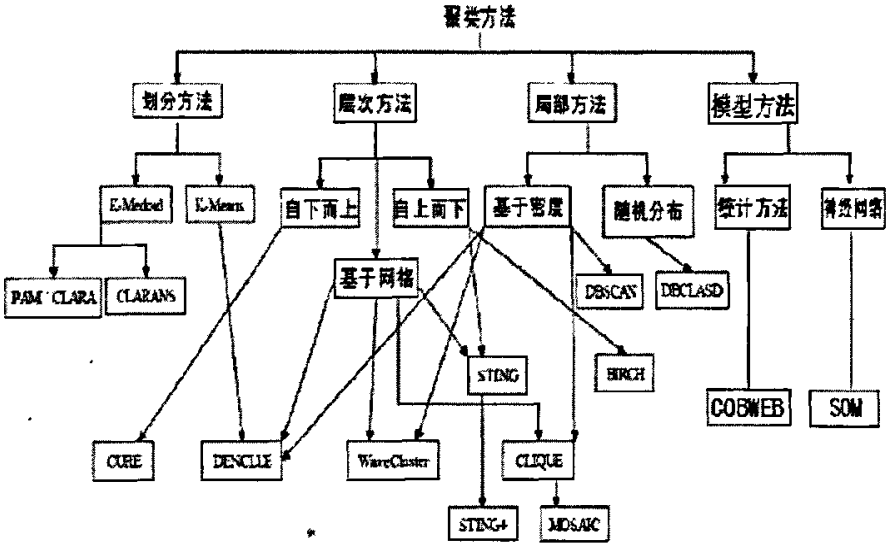


图 1.1 聚类方法架构

所有的聚类方法都具有各自的特点。有些以方法简单、执行效率高见长(如 k 平均)；有些对任意形状、大小的类识别能力强(如 CUBN)；有些能很好的过滤噪声数据(如 DBSCAN)。但这些方法都有各自的局限性。如 k 平均方法只能识别大小近似的球形类；CUBN, DBSCAN 的时间复杂度都为 $o(n^2)$。另外，很多聚类方法对输入参数过分敏感而且参数很难确定，这加重了用户的负担。因此，尽管已存在众多的聚类方法，人们仍在致力于研究聚类能力强、执行效率高、参数设置简单的聚类方法。

1.1.2 海量高维聚类问题

聚类是按照一定的要求和规律对事物进行区分和分类的过程，将数据集合分成落干类或簇，使得每个簇中的数据点之间最大程度地相似，而不同簇中的数据点最大程度不同。其原始样本数据是由被研究对象的物理、或形态、或化学、或组成等等性质加以数量化后形成的，也即是研究对象的这些性质加以数量化以后将形成模式空间的一个向量，即模式，并且用这个向量来代表该研究对象。在这个过程中没有任何关于分类的先验知识，没有教师指导，仅靠事物间的相似性作为类属划分的准则，因此属于无监督分类的范畴。

在实际问题的分析研究中，人们往往倾向于搜集尽可能多的被研究对象性质，并加以数量化形成模式空间中的自变量。这是因为首先，我们总是希望通过被研究对象的各种性质能尽量完整地、准确地描述被研究对象。其次，希望被研究对象各种性质能够被显示，如果被研究对象的一种性质能表达各类模式间的细微区别，那么性质越多就越能将各类模式区分开来。再次，当被研究对象的性状、行为或组成非常复杂时，往往也需要大量的性质或组成含量测定值才可能较为准确地说明被研究对象的特征。同时，随着科学技术和现代分析测量技术的迅速发展，使可以测定的性质越来越多，可以测出的组分含量越来越微小。这样导致描述研究对象的模式维数变得很高。所以，慢慢随着聚类分析应用的领域扩展和深入，高维聚类问题成为当前聚类分析研究的重点。典型的高维聚类应用如 web 挖掘、文本聚类、搜索引擎、客户分析等。

常用的聚类方法包括划分方法、层次方法、基于密度的方法、基于网格的方法和基于模型的方法。但在实际应用中，由于大多数的聚类方法在处理高维数据集时会出现高维失效问题，主要表现在：①高维空间中数据分布的稀疏性和不对称性使得基于 L_p 范数的距离度量无法正确衡量对象之间的相似程度[1]；

②多数算法的聚类结果的质量对参数设置非常敏感，处理高维数据集时，用户很难合理地设置参数，而且在缺乏背景知识且无法可视化聚类结果的情况下，用户也很难判断聚类结果的质量。为避免由于数据维度增长而导致聚类效果的恶化，常考虑在聚类分析中结合降维技术降低数据维数。对于海量数据，则需降低聚类算法复杂度或采用抽样技术。由于聚类分析算法本身的复杂性使得一般用户难以理解和接受聚类结果，可视化技术由于其直观性可以弥补聚类算法

复杂性的缺陷，从可视化的角度更直观地理解数据分析和聚类分析的整个过程。

第二节 国内外研究现状

1.2.1 技术综述

由于高维数据存在的普遍性，对高维数据的处理已经成为近几年研究的一个热点。为解决高维数据聚类，国内外一些专家学者提出了许多不同的方法。总的来说可以概括为三种[2]，即特征转换法、子空间聚类法及基于模式相似的聚类。

1 特征转换法(feature transformation)[3]

对于高维数据，可以采用属性转换或属性约简方法，以减少数据维度，然后利用传统的聚类算法在较低维的数据空间中完成聚类操作，如主成分分析(PCA)[4]、多维缩放(MDS)[5]、小波分析[6]等，都是普遍应用的降维方法。

在多变量统计分析中，PCA 经常被采用以降低数据维度[7][8]。它首先计算原始数据的协方差矩阵和该矩阵的前 k 个特征向量，然后把原始数据映射这 k 个主特征方向上，映射后的数据维很低，可以应用传统聚类算法如 k-means, AutoClass 等。PCA 在确定 k 值时有较大的难度，若设得太小，将使数据丢失很多有用的信息；若设得太大，虽然可以保留较多的特征信息，但对传统聚类算法来说维数太高，无法有效地聚类。另外，PCA 的计算量大，内存需求高，均为 $o(m^2)$，当 m 值很大时，其算法复杂度无法接受。

在信息获取领域，经常用以减少维度的技术是奇异值分解(Singular value decomposition, SVD)。类似 PCA 的潜在语义分析(Latent Semantic Indexing，LSI)[9]是与 PCA 类似的一种属性约简方法。与 PCA 使用协方差矩阵的特征向量不同的是，LSI 直接使用原始数据的奇异值分解技术。由于它不再计算协方差矩阵，其对内存和 CPU 的需求比 PCA 较少。

2 子空间聚类（subspace clustering methods）[10]

子空间聚类是从另一角度处理高维数据。由于直接在高维空间中寻找簇(Clusters)很困难，有些算法就把原始数据空间划分为不同的子空间，从子空间考察聚类的存在。和特征选择一样，子空间聚类需要使用一种搜索策略和评

测标准来筛选出需要聚类的簇，选择的搜索策略对聚类结果有很大的影响。根据搜索的方向的不同，可以将子空间聚类方法分成两大类：自上而下的子空间查询方法和自下而上的子空间查询方法。

自上而下的子空间查询方法首先在整个维上初始化一系列近似的簇，这时每个维赋予相似的权值。接着每个维相对于不同的类赋予不同的权值，将这些更新的权值使用于下一个循环来重新生成簇。这种方法在全局空间中需要重复多次复杂的聚类算法，导致时间复杂度较高。许多使用该策略的算法都利用采样技术来提高算法的性能。大部分至上而下的方法都需要输入一些参数，例如：簇的数量、子空间大小等。这些参数往往事先是不知道的，不同的输入参数往往会得到不相同的聚类结果。其中子空间的大小作为参数输入，这就在某种程度上导致至上而下的子空间聚类方法试图在相同或者相似大小的子空间中查找簇。其中具有代表性的算法有 PROCLUS 算法[11]和 ORCLUS 算法[12]。

PROCLUS 是较早提出的至上而下的子空间聚类方法，采用随机采集样本的方法使得该算法在运行速度上有较大的提高；然而该方法需要输入聚类的数目和每个簇相关维的大小。当不同簇之间的大小相差较大时，该算法的聚类结果不是非常准确，该算法对输入的参数较为敏感。ORCLUS 算法是对 PROCLUS 算法的改进，可以找到任意形状的簇，但是该算法仍然对输入参数敏感，而且得到的聚类结果很难理解。

自下而上的子空间查询方式使用了关联规则方法中的先验性质[13]，这种类型的查询方法首先将每个维作为单独的项进行考虑，然后依次合并维并判断其子空间的优劣。其中具有代表性的算法有 CLIQUE 算法、ENCLUS 算法、MAFIA 算法以及 DOC 算法等。

CLIQUE (Clustering In QUEst)[14]算法是较早提出的子空间算法之一，是一种基于数值属性子空间聚类的高维聚类算法。它的主要特征是将基于密度的聚类方法与基于网格的聚类方法相结合，采用与关联规则挖掘中 Apriori 算法类似的维归约原理。该算法可以自动发现高维数据中的子空间，对数据项的输入顺序不敏感，无需假设任何规范的数据分布。在该算法中用子空间的覆盖作为衡量该子空间中数据聚类是否良好的标准，实际上这个标准并不能完全反映该子空间中聚类的好坏。例如根据覆盖的定义，含有一个高密度的稠密单元的子空间与含有多个低密度的稠密单元的子空间可能有相同的覆盖，但显然是前者的聚类质量更好[15]。由于该方法大大简化，导致聚类结果的精确性不是很高。

ENCLUS 算法[16]继承了 CLIQUE 算法的主要思想，但该方法并不直接测量子空间的密度或者覆盖，而是通过熵值加以判断。具有较坚实的数学基础，概括了一些其他算法，包括基于划分的、层次的、以及基于位置的方法，可以查找任意形状的聚类。但是，这个方法要求对密度参数和噪声阈值进行仔细的选择，因为这样的参数选择可能显著地影响聚类结果的质量。

MAFIA 算法[17]对 CLIQUE 的改动比较大。它从对数据的遍历开始，在每一个维建立适应网络(adaptive grids)。然后，MAFIA 通过向内存中整块读入数据，使用 1000 个箱格计算网格自方，然后合并这些箱格，得到最后的簇或聚类。该算法使用一个称为聚类优势因子的参数 a，选择数据列密度高于平均密度 a 信的箱格。从 q=1 候选密度区域(candidate dense units, CDUs)开始，该算法递归地向高维处理，每次都包含一次数据的遍历。研究指出，a 以在一个单向区间范围内取一系列的值。如果 q 是 CDU 的最高维度，MAFIA 的计算复杂度为 $o(const^q + qN)$。

DOC 算法[18]是一种基于密度的最佳映射聚类算法，该方法首先从数据集中选择一个随机数据项，然后找出该数据项所在的映射簇。重复上述方法多次后，得到一个最佳的数据项以及对应的相关维:然后从数据集中将满足条件的数据项划到该簇中。该方法克服了 PROCLUS 算法的一些缺陷，例如，该算法可以自动确定聚类的数目，可以得到不同大小的类。算法的运行时间随着数据集中数据的增加呈线性的增长，却随着数据维数的增加呈指数级增长。

子空间聚类算法较大的一个弊端是计算的复杂度，当数据维数很高并且要求较精确的聚类结果时，子空间的数目会急骤增长，对子空间中簇的搜索就会成为聚类操作的瓶颈，从而使算法失效。

3 基于模式相似的聚类

尽管自底向上策略和自顶向下策略中聚类的方式不同，但两者的相似性概念依然是基于传统的相似性尺度即距离，即相似的数据对象至少在维的子空间上有非常接近的值。但在很多实际应用中，还存在另外一种相似性，不同的数据对象尽管在维的某个子空间上的取值不是很接近，但却呈现出一致的模式如幅度变化相同或变化趋势相同。这种相似称为模式相似，模式相似的聚类正是基于这种相似性尺度。

Cheng 等首先提出了 bicluster 模型[19]以分析生物信息学的 DNA 阵列中基因

在某些方面的相似性。如果一组数据对象子集 U 在维集 D 上具有相同的趋势，则 U 和 D 就构成 bicluser。由于 DNA 阵列表达时趋势比值的接近更有利于分析某个基因与条件的关系，因此 bicluster 模型可以很好地解决生物数据的分析。在 bicluster 模型中，使用启发式随机算法来找到 bicluster。

尽管 bicluster 模型及其算法已用于生物信息学场合，但也有其局限性。bicluster 模型采用平均误差平方为测度衡量 bicluster，但平均误差平方不能很好地衡量一个 bicluster；另外此算法不能正确地找到重叠簇。为解决以上问题，Wang 等提出了 pCluter 模型[20]。一个 pCluter 由一组数据对象 U 和一组属性 D 组成，设$u1,u2 \in U, d1,d2 \in D$，则 pCluter 要求$u1$从$d1$变化到$d2$的趋势与$u2$从$d1$变化到$d2$的趋势相似。pCluter 算法采用深度优先的策略找到所用满足用户设定阈值的 pCluter。

1.2.2 本文研究思路

上述三种高维聚类方法各有利弊，各有其不同的适应领域。当数据维数较低时(小于 100)。可以使用属性转换加传统聚类技术，也可使用子空间聚类方法。当数据维数较高(大于 100)时；协同聚类较为有效，尤其是文本聚类、web 挖掘等领域。一些基于神经网络的聚类方法巧妙地处理了高维数据中的非线性，是一类比较有特色的降维聚类方法。其中，较有代表性的方法是自组织映射(Self-Organization Mapping，SOM)[21]。

Kohonen 提出的 Self-Organizing Mapping (SOM)特征映射是一种基于非监督学习的竞争网络，具有将高维数据可视化地表现在低维空间的能力，它能够揭示隐藏在高维数据中的复杂的非线性关系，并将其在低维空间中以简单的几何关系展现出来。关于 SOM 的理论分析和工程应用的研究在神经网络领域受到相当的重视[22][23][24]。本文将现有方法进行了较为全面的综述，然后分别从两条主线（聚类与可视化）开始进行研究，提出了可视化聚类方法，并分别用经典和标准数据库进行数据仿真。实验表明：该算法对于线性不可分与奇异点的检测具有很好的自适应性和鲁棒性，能够很好地表现出高维数据内部的结构。

本文的研究思路可以用图 1.20 所示。

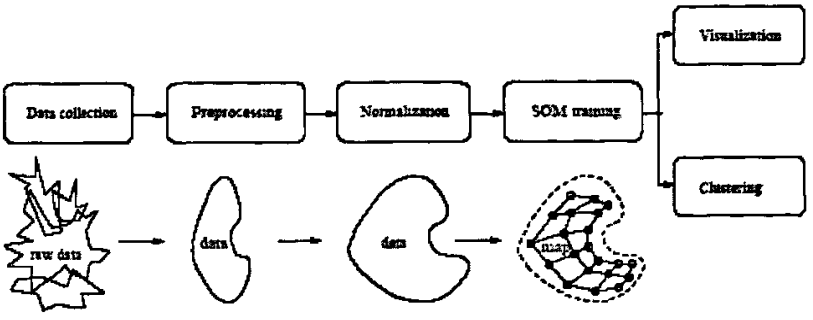


图 1.2 本文研究思路

第三节 论文组织结构

本文总共分为 5 章。

第 1 章是绪论，简略介绍了目前聚类技术的由来及现实意义，并给出了论文的主要研究内容和基本结构。

第 2 章主要分析了 SOM 网络，讨论 SOM 的基本性质，学习算法，收敛性和时间复杂度。讨论了它的各种变体。

第 3 章是讨论分析 SOM 算法的聚类，针对一般 SOM 算法时间复杂度高，聚类结果不易理解等缺点，引入 U 矩阵，提出了可视化聚类方法，实验表明有很好的聚类效果。

第 4 章详细分析了基于 SOM 的可视化技术，提出 SOM 的可视化模型，通过颜色对比的方法来起到可视化聚类的作用，通过在二维和多维数据集上进行仿真实验，验证了可视化聚类方法的有效性和实用价值。

第 5 章给出了本文的结论。列举了论文的主要研究内容和贡献点，并探讨了进一步研究的方向。

第二章 SOM 算法及其变种

第一节 基本 SOM 算法

2.1.1 基本竞争型人工网络

生理学实验证明，生物视网膜有许多特定细胞对特定的图形例如直线河源比较敏感。当视网膜中有若干接受单元受特定模式刺激时，就使大脑皮层中的特定神经元开始兴奋。这种由特定刺激引起局部兴奋的反射功能是受后天环境影响而形成的一种生理现象，是一种自学习的过程。竞争学习即使受此启发设计的，它使同一神经元层上各神经元相互之间进行竞争，最终将有一个神经元获胜，竞争胜利的神经元修改其与输入层的联接权值。在生物神经系统中，还存在侧抑制现象，即一个神经细胞兴奋后，通过它的分支会对周围其它神经细胞产生抑制。这种竞争抑制使神经细胞之间出现竞争，虽然开始阶段各神经细胞处于不同程度的兴奋状态，出于侧抑制的作用，各神经细胞之间相互竞争的最终结果是：兴奋作用最强的神经细胞所产生的侧抑制作用战胜了它周围所有其它细胞的抑制作用而最终“赢”了，而其周围的其它神经细胞则全“输”了。

基本竞争型人工神经网络由输入层和竞争层组成，输入层有 N 个神经元，竞争层有 M 个神经元，输入层和竞争层实现全互联。其网络基本结构如下：

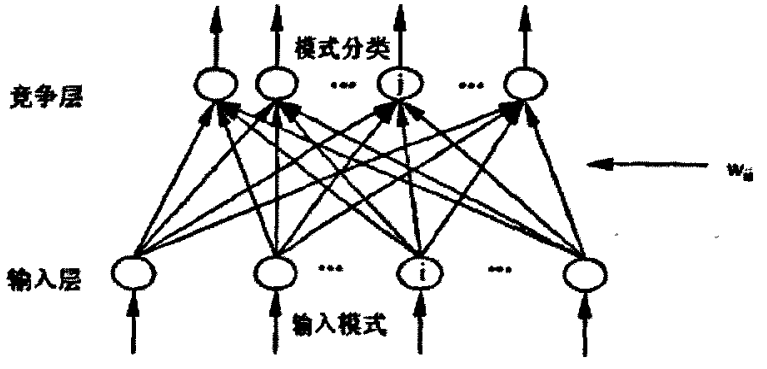


图 2.1 基本竞争网络结构

其中网络的连接权为$\{W_{ij}\}, i=1,2,...N; j=1,2,...M$,且约束条件为：$\sum_{i=1}^{N} w_{ij}=1$

在竞争层中，神经元之间互相竞争，最终只有一个或者几个神经元获胜，以适应当前样本。竞争胜利的神经元就代表着当前输入样本的分类模式。

竞争型网络的输入样本为二值向量，各元素取值 0 或 1。竞争层神经元 j 的输入值$S_j=\sum_{i=1}^{N} W_{ij}X_i$，按照“胜者为王”的原则(Hecht Nielsen, 1987)，以对应S_j最大的那个神经元作为胜者，使其输出为 1，表示该神经元在该样本的输入下兴奋(如果出现$S_i=S_j$，取 i 值小的那一个)。然后，修改与获胜神经元相连的各联接权：

$$w_{ij}=w_{ij}+\alpha(\frac{x_i}{m}-w_{ij}) \tag{2.1}$$

其中，α为学习参数，$0<\alpha\leq 1$，一般取为 0.01~0.03；m 为输入层中输出为 1 的神经元个数，即$m=\sum_{i=1}^{N} x_i$。

最后，当w_{ij}变化很小的时候，网络的拓扑结构也就确定下来了。然后在网络回想时，根据所记忆的学习模式对输入模式做出最近邻分类。

2.1.2 SOM 的两种训练算法

自组织特征映射网络也称为 Kohonen 网络，或者称为 Self-Organizing Feature Map(SOM)网络，它是由芬兰学者 Teuvo Kohonen 于 1981 年提出的。该网络是一个由全连接的神经元阵列组成的无教师自组织、自学习网络。Kohonen 认为，处于空间中不同区域的神经元有不同的分工，当一个神经网络接受外界输入模式时，将会分成不同的反应区域，各区域对输入模式具有不同的响应特性。

SOM 网络由输入层和竞争层组成，结构如图 2.2 所示。

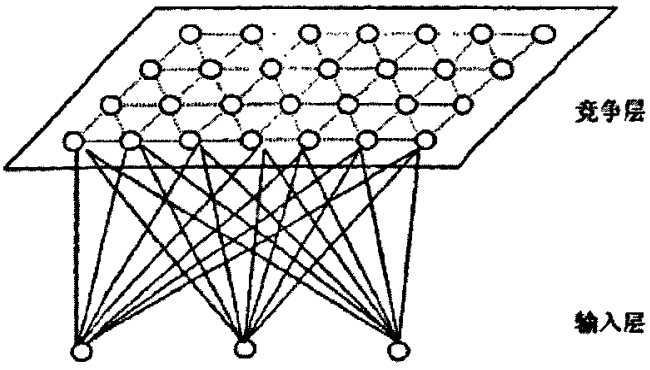


图 2.2 SOM 网络的结构

SOM 网络的一个典型特性就是可以在一维或者二维的处理单元阵列上，形成输入型号的特征拓扑分布，因此SOM网络具有抽取输入信号模式特征的能力。SOM 一般只包含有一维阵列和二维阵列，但也可以推广到多维处理单元阵列中去。由于研究及实用最多的是 2 维阵列，本文仅讨论 2 维阵列结构的 SOM 网络。SOM 网络模型由以下 4 个部分组成：

（1） 处理单元陈列。用于接受事件输入，并且形成对这些信号的“判别函数”

（2） 比较选择机制。用于比较“判别机制”，并选择一个具有最大函数输出值得处理单元（Winner Takes All 机制）

（3） 局部互联作用。用于同时激励被选择的处理单元及其最邻近的处理单元

（4） 自适应过程。用于修正被激励的处理单元的参数，以增加其对应于特定输入“判别函数”的输出值。

设 M 个神经元有序地排列在 2 维平面中(这一平面也被称为 SOM 网络的输出平面，即为竞争层)，如图 2.3 所示。SOM 网络还存在其他的排列方法，不同的排列方法对网络的学习方法没有影响。

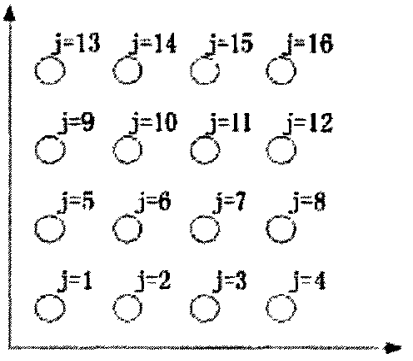


图 2.3 具有 16 个神经元的 SOM 网络

在图 2.3 中，各神经元在网络的输出平面上占据了固定的位置，因此可以对 m 个神经元进行编号(j=1,...,m)。每一个神经元有一个相应 n 维权矢量(也称为权重矢量)，记为 w_j。这里神经元权重矢量维数 n 和网络学习样本的维数一致。SOM 的训练方法主要有两种:连续训练方法和批量映射方法，下面分别予以介绍。

（a）连续 SOM 训练算法(Sequential training algorithm)

（1） 权连接初始化

对所有从输入节点到输出节点的连结权值 w_j 赋以较小的权值。可供选择的方法有

- 随机初始化（random initialization）：电脑随机产生一些小数赋予权值
- 样本初始化（sample initialization）：随机选取输入样本属性值赋予权值

（2） 网络输入模式为

$$X=(x_1,x_2,\cdots x_n) \tag{2.2}$$

（3） 训练学习过程

每一个训练步骤中，计算输入的样本向量 X 与全部输出节点连接权向量 W_j 的距离。根据“胜者为王”的竞争规则，具有最小距离的节点 c 竞争获胜，即与输入模式最相近的神经元，称为最匹配神经元（Best-Matching Unit，BMU）

$$\|X-w_c\|=\min_j\{\|X-w_j\|\} \tag{2.3}$$

更新最匹配神经元以及它所在拓扑邻域内神经元的权值，使他们逼近输入

样本向量，具体示意可用图 2.4 说明。

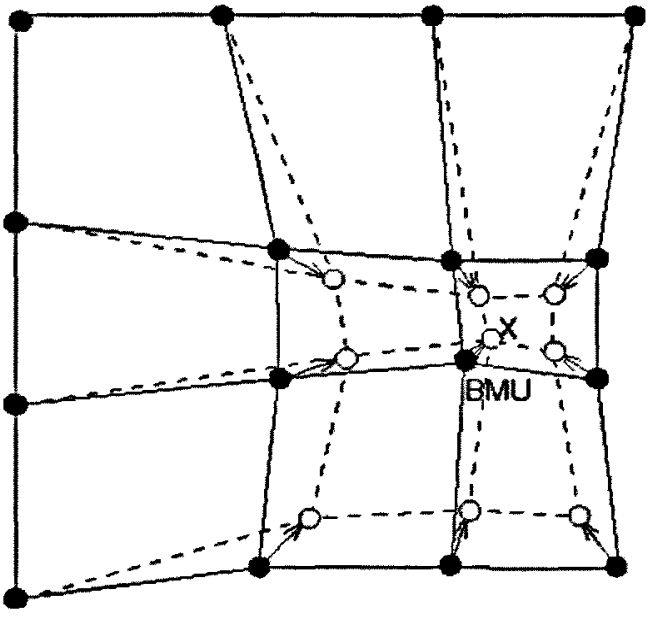


图 2.4 BMU 及其邻域神经元向输入样本向量逼近

SOM 权值向量的更新规则如下

$$w_j(t+1)=w_j(t)+h_{cj}(t)[X(t)-w_j(t)] \tag{2.4}$$

式中 t 为迭代次数，$X(t)$ 为 t 时刻输入样本向量，$h_{cj}(t)$ 为 t 时刻“优胜者单元”c 的邻域核函数。邻域核函数是一个随时间非增的函数，它定义了输入样本在 SOM 网络邻域空间上的影响程度。核函数主要由两部分组成：邻域函数 $h(d,t)$ 和学习速率函数 $a(t)$

$$h_{cj}(t)=h(\|r_c-r_i\|,t)a(t) \tag{2.5}$$

其中 r 代表神经元在图片网格中的位置。一般来说，$a(t)$ 在跌代开始阶段，可以将学习速率设置较大（接近 1），而在后期将学习速率设置的较小（接近 0）。也可以设置为随 t 而线性或者指数地下降。较常用取法为 $a(t)=0.9(1-t/1000)$。$h(d,t)$ 是我们所定义的邻域函数，是以获胜神经元为中心的一个区域。在跌代开

始阶段，一般邻域范围较大，例如取 6 近邻、4 近邻等，随着跌代的进行邻域范围逐渐变小，最后可收敛到只有获胜神经元。另一种做法是把 $h(d,t)$ 取为高斯函数 $\exp(-\frac{\|r_c-r_i\|^2}{2\delta^2(t)})$，其有效范围随着高斯分布方差的变小而缩小。文献[35]对 SOM 网络参数初始值的选择、学习速率函数和邻域函数的设计、以及参数自适应学习等内容进行了较为深入的探讨，可供参考。

随着学习的不断进行，学习率 $a(t)$ 将不断减小，邻域 $h(d,t)$ 也将不断缩小，所有权向量将在输入向量空间相互分离，各自代表输入空间的一类模式，这就是 Kohonen 网络特征自动识别聚类功能。

（b）批量映射法(Batch map)

批量映射法的重复训练过程基于一个固定的点，在训练过程中，所有训练数据的最佳匹配点 BMU 都可以通过上式(2.3)获得，但是原型矢量更新公式变为：

$$w_j=\frac{\sum_{i=1}^{n}h_{c,j}x_i}{\sum_{i=1}^{n}h_{c,j}} \tag{2.6}$$

新的原型矢量用于表示整个数据样本的平均权系数，每个样本数据的权系数为其 BMU(c)邻居核函数 $h_{c,j}$ 的值。初始的 SOM 可以看成是批量映射算法的随机表示。

2.1.3 质量评估

构建 SOM 的方法多种多样，不同的方法构建的 SOM 效果也不尽相同。如何评估一个 SOM 的优劣是 SOM 研究的一个重要方向，目前，人们常使用变形尺度(Distortion Measure)来衡量 SOM 的好坏。现在有许多方法都使用代价函数或能量函数来确定 SOM 的优化程度，从 SOM 的定义可以看出，通常情况下，SOM 的代价函数并不具有梯度性。假设数据集是离散的且邻居核固定，则 SOM 的变形尺度为：

$$E_d = \sum_{i=1}^{n}\sum_{j=1}^{m} h_{c,j} \left\| x_i - w_j \right\|^2 \tag{2.7}$$

当任意样本数据的 BMU 发生变化时，能量函数也发生微小的变化。当能量函数的值最小时，SOM 达到最优。

另一种衡量 SOM 优劣的指标是量化误差(quantization error)，可以用下式表示：

$$E_q = \sum_{i=1}^{n} \left\| x_i - w_j \right\|^2 \tag{2.8}$$

2.1.4 矢量量化

原型矢量用于近似表示原始的样本数据，其常用的初始化方法主要有随机法、矢量量化法等。随机法比较简单，就是随机产生和原始变量维数相同的原型矢量，这种方法的缺点是 SOM 的训练过程非常耗时，通常在原始变量的维数不高且数据量较小的时候使用。下面重点介绍矢量量化法。

1) 经典的矢量量化算法

所谓矢量量化算法就是找到一个原型矢量数据集 w_i，i=1,...,m。用它最大程度的近似表示原始的数据集。最著名的算法 k-均值(k-means)算法，它可以很方便的找到原型矢量数据集，并使其量化误差最小。原型矢量的数据密度决定于待训练样本数据密度，总的来说，它满足下式：

$$P(m) \propto p(x)^{\frac{d}{d+r}} \tag{2.9}$$

其中，d 为数据的维数，p(x)和 p(m)是输入数据和原型矢量对应的概率密度函数。量化操作将原始数据集减少为一个只有较少数据的原型矢量数据集。这个原型矢量的功能非常强大，它可以在减少计算代价的前提下获取近似的结果，现已经成功的用于聚类分析、投影变换、去噪等多种用途，尤其是在数据挖掘和数据可视化方面，它的应用前景更是广泛，本文后续章节将作详细介绍，这里先不赘述。

2) SOM 量化算法

SOM 与 k-均值算法非常接近，如果 SOM 中邻居核的值在最佳匹配点 BMU 上为 1，而在其它地方为 0，则 SOM 就变成自适应 k-均值算法了。

经典量化算法与 SOM 量化算法的不同之处在于 SOM 为每个网格单元的邻居都执行了一个局部的平滑操作，这个平滑操作建立了原型矢量之间的顺序性，当邻居半径随着训练过程的进行逐渐递减时，它实际上模拟了训练方案的一个退火过程，这就使得整个 SOM 量化的过程更加健壮。

样本数据的量化具有非常重要的作用，它能够反映数据集中数据的重要性。

2.1.5 矢量投影

矢量投影的目的是寻找原始高维数据所对应低维坐标，使这些低维坐标能够保持原始数据之间距离。多维比例缩放法（MDS）就是一种很经典的矢量投影方法，当降低原始数据的维数时，它能够很好的保持原始信息之间的关系。该方法的最小误差函数可由下式表示：

$$E=\sum_{i=1}^{n}\sum_{j=1}^{n}(d_{ij}-d_{ij}')^2 \tag{2.10}$$

其中，d_{ij}是样本数据 i，j 在输入空间的距离，d_{ij}'为样本数据 i，j 在输出空间的投影坐标上所对应的距离。

SOM 同样具有上述矢量投影的性质，其变形尺度

$$E_d=\sum_{i=1}^{n}\sum_{j=1}^{m}h_{c,j}\left\|x_i-w_j\right\|^2=\sum_{i=1}^{n}\sum_{j=1}^{m}h(d_{ij}')d_{ij}^2 \tag{2.11}$$

其中，h(.)为邻居核函数，它随着d_{ij}'的变化单调递减，在输出空间的距离越小，则该变量越重要。另一方面，许多映射单元上都对应很多数据，投影数据在输出空间上的距离还与数据的密度相关。因此，SOM 并不注重保持原始数据的距离，相反，它是将原型矢量在预先定义的网格上排序，这样就通过投影在网格上保留了邻居集。

2.1.6 SOM 网络的收敛性

SOM 网络的许多数学性质已经被许多学者研究过。这些研究者包括 Kohonen(1982；1989；1990；1993)、Ritter(1988；1992)、Flangen(1996)、Amarl(1983)、Kraaij1ad(1992)、Maker(1995)和 Cherkassky(1991)，等等。遗憾的是，SOM 网络

实现起来较容易，且大量实验证明能产生很好的效果，但理论分析却困难，目前仍尚有许多问题有待于进一步研究。在这里仅介绍 Ritter(1988; 1992)关于 SOM 网络学习算法收敛性的研究结果。

Ritter 将 SOM 网络的学习算法描述为一个 Markov 随机过程，这是因为 SOM 网络的学习算法的收敛过程是由输入信号序列驱动的一个随机过程。再应用描述这一过程的分布函数随时间演化的 Foker-Planck 方程，就可以得到 SOM 网络学习过程的收敛条件为：

$$\lim_{l\to\infty}\int a(l)dl=\infty \text{ 且 } \lim_{l\to\infty}a(t)=0$$

当满足上述条件时，SOM 网络的学习过程肯定可以收敛到一个平衡点。

2.1.7 计算复杂度分析

假设已经定义了二维的网格，且权值向量已经初始化，则可以使用下列伪代码来表示连续 SOM 训练算法。

```
for (i=0; i<N; i++) {                    /*遍历原始样本，时间复杂度 O(N)*/
bmu=-1;min=1000000;
for (j=0; j<M; j++) {                    /*寻找最佳匹配单元 BMU：O(3Md)*/
  dist=0;
  for (k=0; k<d; k++) { dx=X[i][k]-M[j][k]; dist+=dx*dx; }
  if (dist<min) { min=dist; bmu=j;}
}
for (j=0; j<M; j++)  {                   /*更新权向量：O(3M+3Md)*/
   h = alpha*exp(delta(bmu,j)/rad);      /*高斯类型核邻域函数*/
   for (k=0; k<d; k++)   M(j,k)=h*(M[j][k]-X[i][k]);
}
   }                                     /*总时间复杂度：O(6NMd+3NM)*/
```

在上面的伪代码中，X[i][k]是第 i 个样本数据的第 k 个变量(或者是属性)，M[j] [k]是第 j 个网格单元的第 k 个变量，delta 函数是预先求得的 BMU 和 j 单元之间距离的平方，rad 函数为在 t 时刻的邻域半径乘以-2。

从上面的伪代码可以看出，寻找最佳匹配单元的运算需要 3NMd 次，更新原型矢量的计算步骤需要 3NM(d+1)次，其中 N 是样本数据的个数，M 是网格单

元的个数，d 是输入空间的维数，则上述过程的总计算复杂度为 O(NMd)。

在实际应用中，SOM 的计算复杂度主要由网格单元的个数决定。如果所选得网格单元与$\sqrt{N}$成正比，则网格训练的复杂度为$O(N^{1.5}d)$[25]。当然网络结构的节点数选择从原则上来说是任意的，几乎和样本数据无关，它取决于实际问题的应用，专家建议取10^2数量级(如取 12*8, 16*12 等)。已有研究基于优化搜索 BMU 算法的基础上[26][27][28]，将算法的时间复杂度从 O(Md)改进到 O(log(M)d)。此算法亦可作为并行算法[29][30]，所有这些努力使得将 SOM 算法应用于大规模数据集成为可能。

第二节 SOM 算法的各种变体

2.2.1 基于动态确定神经元数目的改进

由前面介绍可知，SOM 算法作为一种通用的聚类方法，理论上可以完成任意输入空间向量的特征自动提取。但是将该算法应用于特定的数据集时却存在两个问题。一是竞争层神经元个数 M 应预先指定。这种网络结构上的限制大大影响了网络的收敛速度，往往要进行若干次不同 M 值的仿真训练才能最终确定适合于特定应用的网络结构。二是预先确定好的输出层拓扑结构可能无法很好的反应输入数据的特征。为此，人们提出了多种在训练过程中动态确定网络形状和单元数目的解决方案，比较有代表性的有：Alahakoon 提出的 GSOM[31]，GSOM 在初始时，竞争层由 4 个神经元构成正方形结构，在训练过程中，对于每一个输入样本 x，计算其获胜结点 c 的累计误差 TE，若 TE 大于预先指定的生长阈值 GT，则在 c 的邻域内找一空闲位置生成一个新结点，若 c 的邻域内无空闲位置，则将 c 的累计误差 TE 分配给其邻域内的结点。其不足是不能按需要方便地在合适的位置生成新结点；王莉等提出的树型动态增长模型 TGSOM，它与 GSOM 的不同在于它可以按需要方便地在任意合适位置生成新结点，克服了 GSOM 的缺点；Fritzke 提出了增长细胞结构 (Growing Cell Structure，GCS)[32] 算法，GCS 算法从一个由 3 个神经元构成的三角形结构开始，记录下每个神经元获胜的次数，在下一周期开始前，选出获胜次数最多的神经元，在其最大的一

边上增加一个含初始权值的新结点，并重新计算新结点及各邻接结点的获胜次数，同时，可根据结点的获胜次数进行结点的删除操作。Choi 等提出了自组织、自创造的神经网络模型(Self-creating and Organizing Neural Networks，SCONN)[33]，SCONN 在初始时存在一个激活水平足够高的根结点，找出输入向量 x 的最佳匹配单元 c，然后比较$|x-w_c|$与 c 的激活水平。若前者大于后者，生成一个 c 的子结点以匹配 x；否则，修正 c 及邻域结点的权值。

总的来说，这类算法与普通 SOM 网络的不同点在于：网络结构是动态生成的；持续不变的自适应能力；相同点在于：固定的网络维数；邻域的确定取决于网络的拓扑结构。

2.2.2 基于匹配神经元策略的改进

Kohonen 竞争学习机制经常会使得竞争层中有些结点始终不能获胜，尽管 SOM 采用拓扑结构来克服此缺点，但并不是非常有效，为此提出了很多克服此缺点的算法，比较典型的有：SOFM-CV，SOFM-C，ESOM(Expanding Self-organizing Map)[34]，DSOM[35][36]。SOFM-CV 的思想是：把 SOM 网络的权值都初始化为$1/\sqrt{n}$(n 为输入向量的维数)，每个输入向量 X 要经过如下修正后：$ax+(1-a)/\sqrt{n}$ (a 随时间从 0 逐渐增大)，再输入网络。SOFM-C 即带“良心”的竞争学习 SOFM，它的基本思想是给每个竞争层结点设置一个阈值，每次使竞争获胜的神经元的阈值增加，使经常获胜的神经元获胜的机会减小。ESOM 的思想是把更新获胜结点 c 及其邻域结点的权值公式(2.4)修改为下式：

$$w_{ij}(t+1)=c_j(t)(w_{ij}(t)+\eta(t)(x_i-w_{ij}(t)) \qquad (2.6)$$

其中$c_j(t)\geq 1$，由$w_{ij}(t)$，x_i，$\eta(t)$确定。在 TASOM 中，每个神经元都有自己的学习率和邻域函数，并且能根据学习时间自动地调整学习率和邻域的大小。DSOM 的思想是把内源性一氧化氮(NO)的四维动态扩散特性和其在长时间学习过程中的增强作用应用到 SOM 中，输入向量 x 输入网络后，以某种规则(评价函数)确定竞争层中一组获胜神经元，称为亚兴奋神经元簇。并把每一个亚兴奋神经元作为 NO 的扩散源。然后计算各亚兴奋神经元所处位置的 NO 浓度，则 NO 浓度最高的神经元为最终获胜单元。

2.2.3 SOM 算法和其他算法的组合

比较有代表性的组合算法有：Xiao 等提出了把 SOM 和微粒群优(Particle swarm optimization，PSO)算法结合用来对基因数据进行聚类[37]，先用 SOM 算法对基因数据进行聚类，得到一组权值，然后用此权值初始化 PSO 算法，用 PSO 算法对此聚类结果进行优化。Sankar 等提出了把粗糙集和 SOM 结合的 RSOM 算法，它先用粗糙集理论中的依赖规则获得输入数据的大致聚类情况等知识，然后通过这些知识来确定 SOM 网络的结构，并对 SOM 权值进行初始化，用 SOM 网络对结果进行训练、优化。Hussin 等提出了把 SOM 和自适应共振理论(Adaptive Resonance Theory，ART)模型相结合用来对文档进行聚类[38]，先用 SOM 算法对文档进行划分，然后用 ART 对所有的划分进行聚类。孙放等提出了把 SOM 和多层感知器(Multilayer Perceptron, MLP)结合进行语音识别[39]，首先用 SOM 算法进行语音特征矢量量化(VQ)，用轨迹图训练 MLP 网络，相当于建立好了参数模板，用此参数模板就可以进行语音识别。

除了已经介绍的几种以外，还有许多由 SOM 算法发展而来的变体，例如 ASSOM[40]（Adaptive-Subspace SOM)；最近发展起来了一种新兴的 SOM 变体—WEBSOM[41]，主要用于对 Internet 上存在的海量文档进行组织，在所有已经实现的应用系统中，最多的已经对大约 700 万份文档作了有效的组织。

第三章 基于 SOM 算法的聚类分析

第一节 概述

3.1.1 聚类简介

聚类分析是研究数据间逻辑上或物理上的相互关系的技术，它通过一定的规则将数据集划分为在性质上相似的数据点构成的若干个类。聚类分析的结果不仅可以揭示数据间的内在联系与区别，同时也为进一步的数据分析与知识发现提供了重要的依据，如数据间的关联规则，分类模式以及数据的变化趋势等。

绝大多数聚类算法都属于硬划分，，即把每个待处理的对象严格地划分到某个类中，每个数据样本精确的、独一无二地属于某一类别。在这类算法中，隶属度不是 1 就是 0，而现实中大多数的对象并没有严格的属性，而往往是模糊不清的，因此这种硬划分并不能真正地反应对象和类的实际关系，这又促使人们提出了模糊聚类算法。

下图是一个简单的聚类示意图。其中 A，B，C，D 分别表示不同的聚类簇，其中 D 类中又包含了 A，B 两类簇，在聚类分析上又称之为层级聚类。

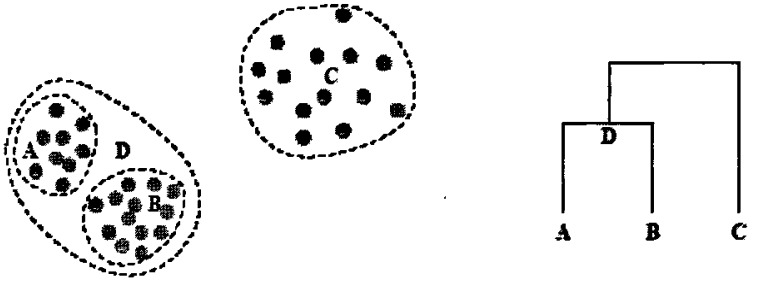


图 3.1 聚类分析示意图

3.1.2　相似度定义

在模式识别问题研究中，人们常常按照“相似的样本在空间中的距离越近”这个前提假设来进行分类。但是，由于样本空间在输入空间中的距离并不能很好地反映样本间的相似程度，因此，期望通过变换尺度来寻找特征空间，在此空间内，变换后的样本呈现出样本间距离与相似一致，使得训练数据在转换后的特征空间中能较好刻画类的几何分布。因此，寻找到能够保持样本间相似程度的距离尺度变成为了关键。其中，样本原始属性变量张成的空间称为输入空间，经过尺度变换后张成的空间成为特征空间。

在实际使用中，我们更多考虑的是 d 维样本向量间的欧式距离，但距离的概念本身要广义的多，在这一节将详细讨论各种可能的相似距离，这也是模式识别中核心问题。总的来说，刻画样本点之间的相似性主要有距离尺度函数和相似尺度函数两类[42]。

- 距离尺度函数：它将样本集的 d 个特征看成张成 d 维空间的维度，而样本集中使用的每个样本都是 d 维空间中的一个点，而样本点之间的相似性可以使用某种距离来表示——距离较近的样本点性质较相似，距离较远的样本点则差异较大。
- 相似尺度函数：通过函数给出两个样本间的相似程度，两个样本越相似，则相似系数值越接近 1；反之，则相似系数值接近 0。这样就可以使用相似数值来刻画样本点的相似程度。

3.1.2.1　距离尺度函数

聚类尺度 $D(\bullet,\bullet)$ 在本质上是一个函数，它给出了两个向量之间标量距离的大小。具体地，设 $X \subseteq R^d$ 是样本点集合，如果对于任意的样本向量 x，y 和 z，函数 $D: X \times X \to R$ 都满足以下四个条件，则称为距离尺度函数。

（1）　非负性　$D(x,y) \geq 0$

（2）　自反性　$D(x,x) = 0$

（3）　对称性　$D(x,y) = D(y,x)$

（4）　三角不等式　$D(x,y)+D(y,z)\geq D(x,z)$

有些距离函数不满足三角等式（4），但满足减弱的条件（4'），从广义的角度也可被成为距离，在[23]中被称为极端距离。

（4'）对于一切 z，　$D(x,y)\leq \max(D(x,z),D(z,y))$ 都被满足

经常使用的满足这些条件的距离尺度函数有：欧式距离、City Block 距离、明氏距离、马氏距离、兰氏距离等。下面简单介绍一下这些距离尺度函数。

明氏距离：定义在 d 维空间上的广义距离，通常也被称为 L_k 范数，具体定义方程为：$D_k(\vec{x},\vec{y})=(\sum_{i=1}^{d}|\vec{x}_i-\vec{y}_i|^k)^{1/k}$

当 k 取 1，2 和正无穷大是，则分别得

（1）　City Block 距离，亦称绝对值距离、Manhattan 距离

（2）　欧式（Euclid）距离

（3）　切比雪夫（Chebyshev）距离

在实际应用中，欧式距离(L_2 范数)和 City Block 距离（L_1 范数）最为常用。

其中，欧式距离的一个优点是当坐标轴进行平移和正交旋转时，欧式距离保持不变。

马氏（Mahalanobis）距离：$D(\vec{x},\vec{y})=(\vec{x}-\vec{y})^T\sum^{-1}(\vec{x}-\vec{y})$

其中 $\sum$ 是样本集合的协反差矩阵，是样本总体分布的协方差估计量。马氏距离是明氏距离的改进，它对于一些线性变换具有不变性，克服明氏距离受量纲影响的缺点，也部分克服了多重相关性。

但马氏距离也有以下缺点：在大规模样本集合上计算整个样本空间的马氏距离效果不甚理想，而且均值和协方差距阵的逆很耗时间，一般使用各个类的协方差矩阵来计算各自的马氏距离，而这种计算又会导致距离尺度只能作用在局部样本空间，因此作为大规模数据集的全局距离尺度不太合适。但其局部性使其在最近邻分类算法中表现优异。

兰氏（lance）距离[43,44]

$$D(\vec{x},\vec{y})=\sum_i \frac{|\vec{x}_i-\vec{y}_i|}{|\vec{x}_i+\vec{y}_i|}$$

它克服了明氏距离受维度单位影响大的缺点，但没有考虑多重相关性。

3.1.3　相似尺度函数

聚类分析中不仅要把样本点聚类，在有些场合还需要对特征变量进行聚类。特征变量之间的相似性测度除了使用上述的距离函数之外，更常用的是相似系数函数。如果一个函数 $S: X\times X\to[-1,-1]$ 满足以下条件，我们就称之为相似系数函数。

（1）　$|S(x,y)|\le 1$

（2）　$S(x,x)=1$

（3）　$S(x,y)=S(y,x)$

$|S(x,y)|$ 越接近 1，两个特征变量间的关系越密切。经常采用的相似系数有夹角余弦和相关系数两种：

夹角余弦：

$$S(\vec{x},\vec{y})=\frac{<\vec{x},\vec{y}>}{\sqrt{<\vec{x},\vec{x}>\times<\vec{y},\vec{y}>}}$$

夹角余弦经常使用在文本分类的向量空间模型，它是受相似性的启发而来的，夹角余弦函数忽略了各个向量的绝对长度，着重从形状方面考虑他们之间关系。当两个向量的方向相近时，夹角余弦值较大，反之则较小。特殊地，当两个向量平行时，夹角余弦值为 1；而正交时余弦值为 0。

相关系数：

$$S(\vec{x},\vec{y})=\frac{\sum_i(\vec{x}_i-\bar{\vec{x}})\times(\vec{y}_i-\bar{\vec{y}})}{\sqrt{(\sum_i(\vec{x}_i-\bar{\vec{x}})^2\times(\sum_i(\vec{y}_i-\bar{\vec{y}})^2)}}$$

相关系数是对向量作标准化后的夹角余弦。它表示两个向量的线性相关程度。

$c_k = \frac{1}{N_k}\sum_{x_i \in C_k} x_i$ 基于上述距离尺度的定义，聚类距离具体可以分为二类：模式类内距离（within-cluster distances）和类间距离（between-clusters distances）。见表 3.1。类内距离 $S(\bullet)$ 代表着自身类内的方差偏移变化，而类间距离 $d(\bullet,\bullet)$ 则定义着两类之间的差异情况。$x_i, x_{i'} \in C_k, i \neq i', x_j \in C_l, k \neq l$。$N_k$ 是聚类 C_k 的样本数，是类 C_k 的中心。

表 3.1 各种聚类距离列表

类内距离 $S(C_k)$	
平均值(average)	$S_a = \frac{\sum_{i,i'} \lVert x_i - x_{i'} \rVert}{N_k(N_k-1)}$
最近邻(nearest neighbor)	$S_{nn} = \frac{\sum_i \min_{i'} \{\lVert x_i - x_{i'} \rVert\}}{N_k}$
质心(centroid)	$S_c = \frac{\sum_i \lVert x_i - x_k \rVert}{N_k}$
方差(variance)	$S_v = \sum_i \lVert x_i - x_k \rVert^2$
类间距离 $d(C_k, C_l)$	
单联接(single linkage)	$d_s = \min_{i,j} \{\lVert x_i - x_j \rVert\}$
全联接(complete linkage)	$d_{co} = \max_{i,j} \{\lVert x_i - x_j \rVert\}$
均联接(average linkage)	$d_a = \frac{\sum_{i,j} \lVert x_i - x_j \rVert}{N_k N_l}$
质心(centroid)	$d_{ce} = \lVert c_k - c_l \rVert$

在表中，S_{nn} 通过计算每个点与其邻域之间的均值估计了类的密度特性。与

邻域之间的距离特性提供了局部数据点的密度分布，这常常用来发现聚类的边缘特征。

3.1.4　尺度变换

通常，我们不能直接对原始变量进行比较处理，常常需要在聚类前对变量预先作一些尺度变化。已有研究表明，变量的尺度变化对于聚类结果有着很重要的影响。本文将介绍 2 种最常用的尺度变换方法：Z 变换和比例缩放。

（1）标准化的 Z-变换过程如下：

i)计算样本数据集均值

$$m=\frac{1}{n}(x_1+x_2+\cdots+x_n)$$

ii)计算样本数据集的绝对偏差 s

$$s=\frac{1}{n}(|x_1-m|+|x_2-m|+\cdots+|x_n-m|)$$

iii）将原始数据样本 Z 变换：

$$Z_i=\frac{x_i-m}{s}$$

（2）比例缩放法

$$x_{ij}=\frac{x_{ij}-\min_j(x_{ij})}{\max_j(x_{ij})-\max_j(x_{ij})}$$

有研究表明比例缩放在聚类重要比 z 变化有效，但是在离群点的检测上，常常要比 Z 变化敏感[45]。

3.1.5　聚类有效性

由于聚类结果的多样化，其结果的解释也往往因人而异，如何给出一个统一的指标来评价聚类结果的优劣就成为了一个很重要的问题。针对这个问题，文献[46]提出了 30 个有效指标用来评估聚类结果。

常用来表征是一个好聚类结果的通常是具有很小的类内部距离 $S(C_i)$ 同时类间距 $d(C_i,C_j)$ 很大。可以用式 3.1 表示。其值越小，表示聚类效果越好。

$$\frac{S(C_i)+S(C_j)}{d(C_i,C_j)}, \forall j \neq i \tag{3.1}$$

通俗地讲：就是各个簇之间紧密结合，而不同类别之间差距很大。

另一个常用指标是Davies-Bouldin指数。由式3.2表示。

$$I_{DB}=\frac{1}{C}\sum_{i=1}^{C}\max_{j\neq i}\{\frac{S_c(C_i)+S_c(C_j)}{d_{ce}(C_i,C_j)}\} \tag{3.2}$$

第二节　基于 SOM 的聚类分析

3.2.1　SOM 聚类

上一章已经基本介绍了 SOM 的算法过程，现今该算法已被广泛的应用于数据的聚类。它通过在竞争层相似格点的自组织聚合，而自发形成各种聚类。图 3.2 是 SOM 聚类的简单示意。N 个原始样本通过 SOM 算法训练，相似样本将自发的聚集在 M 个网格上，进而形成 C 个簇类。

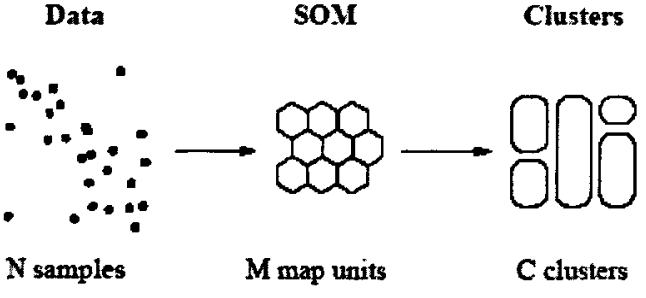


图 3.2　SOM 聚类示意图

从本质上说，SOM是一种矢量量化算法。这类算法一般都是通过迭代，使得原始数据集的重构误差达到最小化。下表3.2是一些常见的矢量量化算法。其主要不同点在于聚类中心点权值的更新。

表 3.2　一些矢量量化算法

算法	区别
K-means[47]	只有样本向量最匹配的聚类中心被更新

最大熵[49]	根据样本距离，更新所有聚类中心
神经气[49]	根据样本距离排序，更新所有聚类中心
SOM	根据样本向量的最佳匹配，更新类中心及其邻域

基本SOM聚类方法主要分为2步：

1. 通过SOM训练，在拓扑层形成相应原始数据的投影分布
2. 根据自组织原理，一些具有相似特性的数据将自发的聚集在一起，从而达到分类目的

通过下面这个例子来简单说明SOM的聚类过程。样本数据采用我国某地区的10个土壤样本，每个样本用7个理化指标表示其形状，原始数据如表3.3所示。其中，全磷，全氮，有机质为百分比，耕层单位cm。

表3.3　土壤样本及形状

序号	土壤类型	全氮	全磷	有机质	PH	代换量	耕层	密度
1	薄层黏底白浆化黑土	0.270	0.142	6.46	5.5	35.8	21	1.03
2	厚层黏底黑土	0.171	0.115	3.46	6.3	33.0	60	078
3	薄层黏底黑土	0.114	0.101	2.43	6.4	26.5	25	1.13
4	厚层黏土厚土	0.173	0.123	3.30	5.8	28.9	65	1.09
5	薄层黏土厚土	0.145	0.131	3.28	6.0	28.5	25	0.98
6	厚层草甸厚土	0.173	0.140	3.45	5.8	33.4	60	0.98
7	中层草甸黑土	0.250	0.177	5.51	7.2	42.5	45	0.93
8	薄层草甸黑土	0.237	0.189	5.37	6.1	32.9	27	1.00
9	薄层沟谷地草甸黑土	0.319	0.227	7.04	5.8	35.9	24	1.03
10	厚层平地草甸土	0.163	0.124	3.73	6.2	30.6	61	1.28

本次实验我们创建的网络竞争层为6*4的结构，输入为7维数值指标，共10个样本数据点。分别设置训练步数10、100和1000，得到聚类结果如下表3.4。

表3.4　聚类结果

训练步数	聚类结果									
10	1	24	1	24	1	24	3	1	1	24

100	19	22	1	12	1	22	4	8	19	24
1000	12	24	1	6	2	23	16	8	20	11

对结果分析发现，当训练步数为10时，样本序号为1、3、5、8和9的分为一类，与表3.3进行对比，可知这5组数据样本都是属于薄层的黑土；样本序号为2、4、6和10的分为一类，而这些都是属于厚层的黑土。而序号为7的数据单独成为一类，它是中层的黑土。由此可见，网络已经对样本进行了初步分类，这种分类虽然准确但不够精确。当训练次数为100时，分类结果更加细化了。但是，随着分类次数的提高，分类结果就没有意义了（当训练次数超过1000次时，样本各自成为一类，没有实现聚类）。

SOM将高维数据映射到一个二维网格中，数据集合的分布以二维数据的形式表现出来，相似程度较高的神经元聚集在一起，在网格中形成一个个聚类的数据云。当神经元的数据急剧增加的时候，受SOM 自身特点以及实际环境的影响，

数据云将相互重叠从而难于分辨，无法得到有用的聚类信息。另外，随着神经元个数的增加，算法的计算复杂度也越来越高。为此文献[50]引入了U矩阵[51]来对 SOM 算法进行预处理，通过在二次聚类将一些可能对结果产生不良结果的神经元剔除，减小计算的复杂度，并且消除了噪音和孤立点数据对最终的分析结果的影响。通过对Wine Recognition数据集的8个随机实验，结果表明经过U-矩阵的SOM聚类错分率降低了2.656%。

本文接下来将介绍U矩阵的简单原理。但本文的目的并不仅仅局限于对SOM的数据预处理这么简单，而是期待着通过 U 矩阵这个媒介来挖掘发现原始数据集的一些内在特性。

3.2.2　U矩阵

图3.3是SOM网络训练结果的U矩阵示意图。图中小柱体表示竞争层相邻神经元权向量之间的距离，小柱体高度表示距离值大小。U矩阵方法是通过三维空间来观察SOM网络的聚类训练结果的。

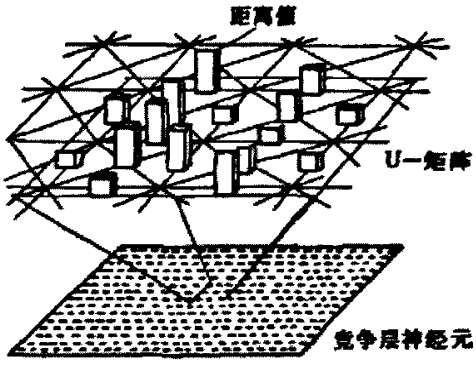


图 3.3　U 矩阵示意图

对于 SOM 网络竞争层的每个神经元 n_i，它都拥有一个二维坐标和一个邻域即时 $N(i)$，如图 3.4 所示。图中显示了 2 种基本的竞争层拓扑结构，邻域函数在训练过程中逐渐减小。

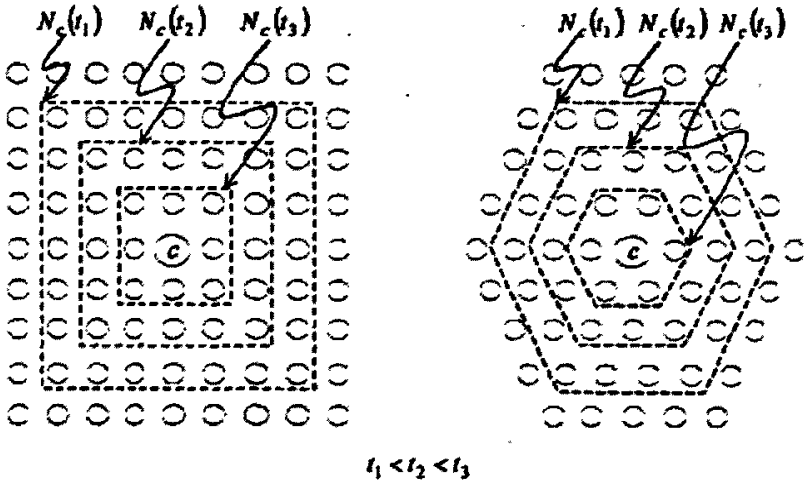


图 3.4　竞争层邻域函数变化示意图

在 U 矩阵中由式 3.3 定义每个神经元的高度：

$$uh(i)=\frac{1}{n}\sum_{j} d(w_i,w_j), j \in N(i), n=|N(i)| \tag{3.3}$$

其物理意义表征着邻域内神经元与相邻神经元权向量之间距离值总合的平

均，当然亦可以用邻域内神经元距离的最大值来表示。

在U矩阵中，如果神经元的$uh(i)$值比较小，就说明它和周围的神经元比较相似，和周围的神经元属于一个数据簇；相反，如果$uh(i)$值比较大，就说明它和周围的神经元相似度很低，它可能处于一个数据簇的边缘或簇间的“分隔地带”。

这样，将神经元高度值用灰度表示，在二维平面上显示自组织网络训练结果。颜色越深表示距离越远，颜色浅则表示两个神经元之间的距离近。所以通过观察高峰、低谷以及竞争获胜神经元的分布情况就可以识别输入向量的聚类结果。颜色浅的区域可以看成是聚类，而颜色深的区域则是边界或孤立[52]。

图3.5是SOM网络的一个U矩阵表达示意图。其中，竞争层上每个神经元的高度值用不同的颜色值表示。

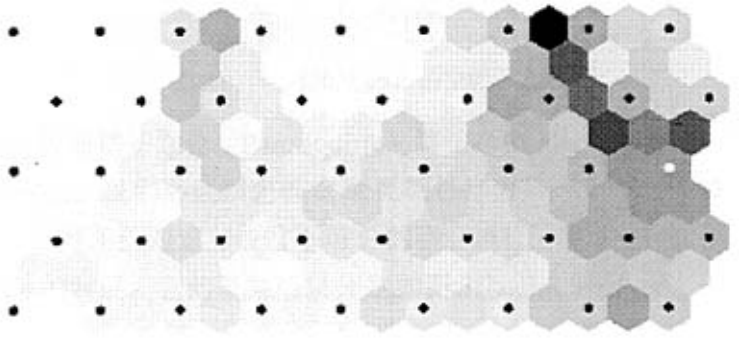


图3.5　SOM网络的U矩阵表达

3.2.3　结合U矩阵的SOM聚类仿真

为了说明结合U矩阵的SOM聚类有效性，本文将对通过一组人造数据来说明，如图3.6所示。聚类对象是一对相交的圆环，每个环有500个数据点，维数为3，聚类数为2。

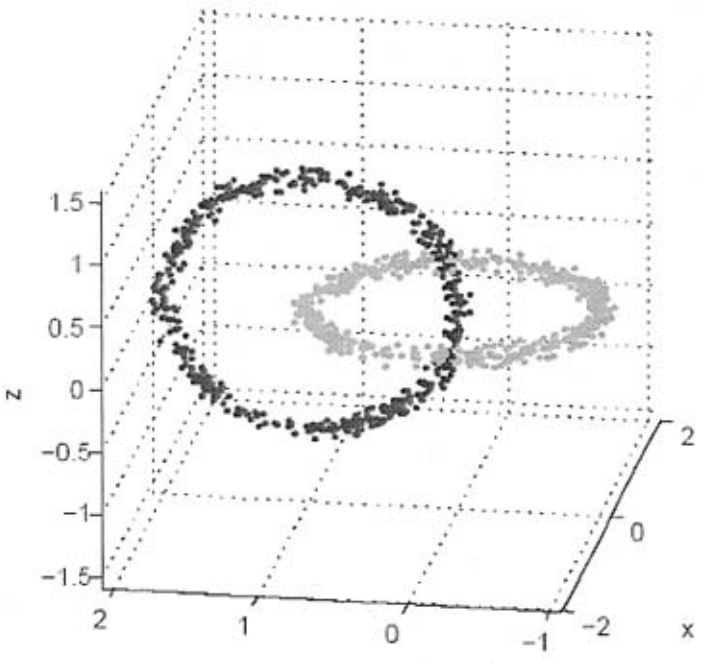


图 3.6　一对相交圆环的空间示意图

K-means 算法属于划分（Partitioning）算法，由 Hartigan 在 1975 年提出。在其训练过程前，我们必须预先确定聚类数目作为其一个训练参数。已有研究表明，K-means 并不能对这种线性不可分的事物进行分类，当它的预定聚类参数分别是 2 和 4 的时候，它的聚类结果分别如图 3.7，3.8 所示。

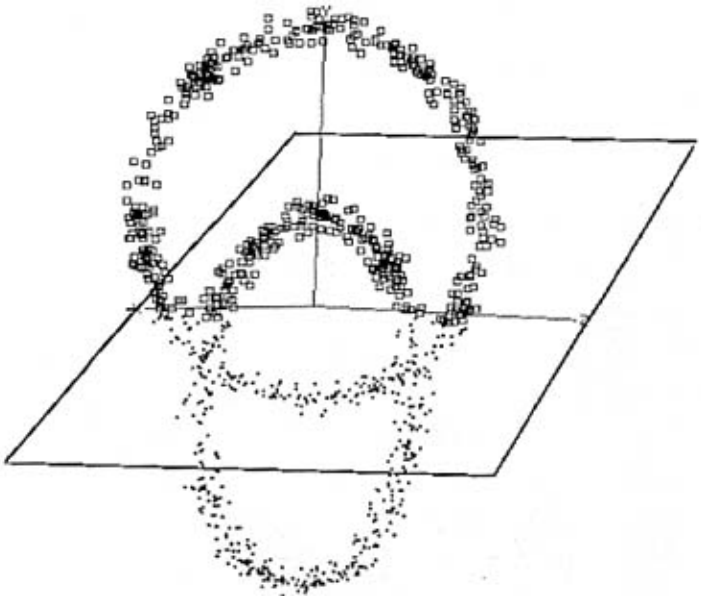


图 3.7　聚类参数为 2 时，K-means 的聚类结果

图 3.8　聚类参数为 4 时，K-means 的聚类结果

从图 3.7，我们可以看出，当聚类参数为 2 时，交叉圆环数据的聚类结果被一个超平面从中切开，这也是一般向量支持机用于样本空间分割的方法[53]。它并不能将数据集正确的分开，而是错误的将数据集从两环中间阶段。这至少从直观意义上来说，是不正确的。从图 3.8 我们发现，当聚类参数预设为 4 时，它的结果表示实际上是 2 个超平面的 2 次切割所致。

为此，我们应用 SOM 算法分析该数据集，在实验中发现，当我们直接对数据进行处理时，常常受困于计算速度与训练次数的选择，而且几乎很难得出一个满意的结果。为此，我们结合 U 矩阵，通过在竞争层上计算每个神经元高度，当多个神经元在拓扑层上延展开来后，得到了图 3.9。

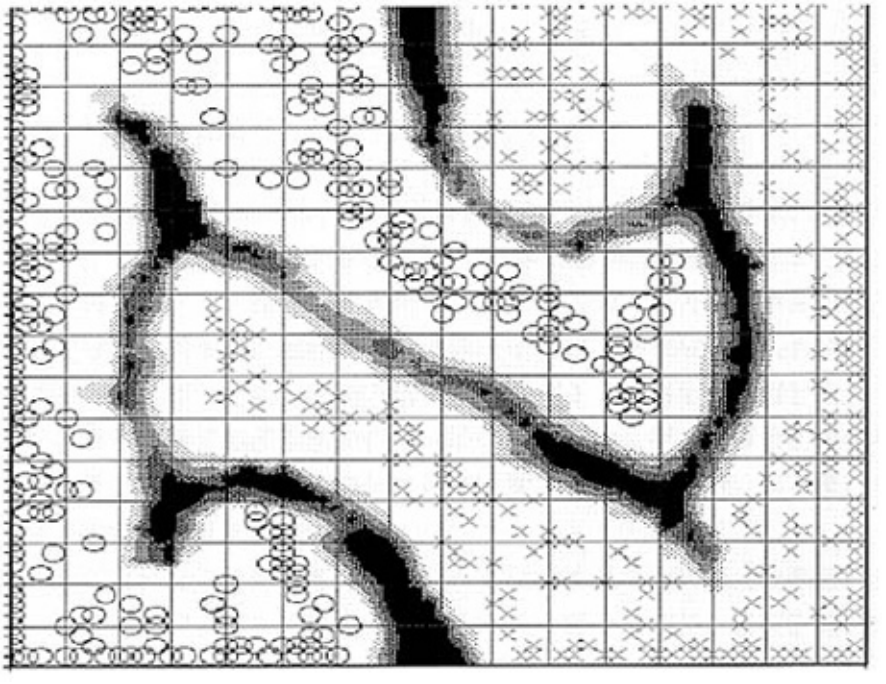


图 3.9　SOM 算法的 U 矩阵示意图

从图中，我们发现竞争层中的神经元自发的在中间形成了一道“墙”——即拥有相对较高的神经元高度，它将原始数据在低维的拓扑映射很好的分为 2 类。这也正是我们需要的结果。

这么一副美丽的“图画”，开始引导着我们进行下一步的研究——原始样本在低维拓扑的可视化聚类。

第四章　基于 SOM 的可视化聚类

第一节　可视化技术

4.1.1　什么是可视化

所谓可视化(Visualization)，牛津英语词典解释为“构成头脑情景的能力或过程，或不可直接觉察的某种东西的视觉”。该术语也指将本来不可见的东西生成可见图像的过程。有人指出：可视化是一系列的转换，这种转换将原始模拟数据转换成可显示的图像，这种转换的目的在于将信息转换成可被人类感应系统所领悟的格式。可视化成为一种方法与技术应用于有关科学和工程技术各个领域，开始于利用计算机图形来加强信息的传递和理解。随后，计算机图像处理技术和计算机视觉也成功地用来处理各类医学图像和卫星图片，以帮助人们理解和利用各类图像数据。

可视化的前身是计算机图形学，今天它已经发展成为研究用户界面、数据表示、处理算法和显示方式等一系列问题的一个综合性学科。根据侧重面的不同，可视化可以分成三个分支：科学计算可视化(Scientific Visualization 或 Visualization in Scientific Computation—ViSC)、数据可视化(Data Visualization)和信息可视化(Information Visualization)。

科学计算可视化是把计算中涉及的和所产生的数字信息转变为直观的、以图像或图形信息表示的、随时间和空间变化的物理现象或物理量呈现在研究者面前，使他们能够观察到模拟和计算，即看到传统意义上不可见的事物或现象；同时还提供与模拟和计算的视觉交互手段。它的实质是运用计算机图形学和图像处理技术，将科学计算过程中产生的数据及计算结果转换为图像，在屏幕上显示出来并进行交互处理，其核心是三维数据场的可视化。科学计算可视化侧重的是科学和工程领域数据的可视化问题。

数据可视化比科学计算可视化具有更加广泛的内容，它不仅包含工程领域数据的可视化，还包含其他领域(如经济、金融、商业等)中数据的可视化。数据可视化概念首先来自科学计算可视化，科学家们不仅需要通过图形图像来分析

由计算机算出的数据，而且需要了解在计算过程中数据的变化。数据可视化可以实现对计算和编程过程的引导和控制，通过交互手段改变过程所依据的条件，并观察其影响。随着计算机技术的发展，数据可视化的概念大为扩展，它不仅包括科学计算数据的可视化，而且包括工程数据和测量数据的可视化。它涉及到计算机图形学、图像处理、计算机辅助设计、多媒体技术和虚拟现实技术、计算机视觉和人机交互等多个领域。

近年来，随着网络技术和电子商务的发展，提出了信息可视化的要求。信息可视化是源于数据可视化，我们可以通过数据可视化技术来发现大量金融、通信和商业信息数据中的隐含规律，从而为决策提供依据。在科学计算可视化中，显示的对象涉及标量、矢量和张量等不同类别的空间数据，研究的重点放在如何真实、快速地显示三维数据场；而在信息可视化中，显示的对象主要是多维的标量数据。

4.1.2　多维信息可视化

所谓多维信息，通常是指变量形式为 $F=f(x_1,x_2,\cdots,x_n)$（n>3）的数据，每个数据含有 3 个维度以上的信息。多维信息的可视化大都采用降维的方式来实现。通常，需要把高维的信息转换到人类视觉能够感知的二维或者三维空间。降维的过程可以是线性的，也可以是非线性的。虽然线性的方法简单明了，但是对于真正复杂的高维信息却很难表达，因此人们开始寻求用非线性的方法来实现多维信息的可视化。非线性的降维方法认为现实数据中大部分点集的有效维比它的空间维小，也就是说一个高维数据中往往有些维度的信息是不明显的。有效维通常由样品数据的显著因子决定。即使点云不能放置在一个低维的线性子空间中，它也具有一个低维的非线性结构与其对应。

由于多维信息的复杂性，很难用简单的标准对现有多维信息可视化技术进行分类。笔者根据可视化技术的目的、类型以及数据的维数，将多维信息可视化技术分为如下三类。

(1)基于 2 变量的多维可视化技术

这种方法由基本的 2 变量显示以及可同步观察这个 2 变量显示的视图组成，它通常作为统计方法使用。基于 2 变量的多维可视化技术充分考虑到了视觉与数据拟合度之间的关系，但这种方法能够处理的数据量较少，一般只能是几百

个数据项，其图形通常由二维点或者线划图的变化关系表示。

(2)基于多变量的显示技术

该技术是近来空间多维信息可视化技术的基础，它绝大部分都是采用通过高速图形计算生成的彩色图形来表示的。这种方法处理的数据量一般比较大，且可以处理复杂数据类型的多维信息。

(3)动画技术

动画技术是一种功能强大的多维信息可视化技术。这种方法采用不同的电影动画技术以及标量可视化模型来表示多维信息。从理论上说，如果数据能够表示成显示双向关系的时间序列，则任何一帧单一画面都能够扩展为动画。

总的来说，国外在多维信息可视化算法研究方面比较活跃，新的研究成果不断出现，其产品化的软件也越来越多，比较著名有美国 Advanced Visual System(简称 AVS)公司的 AVS，Research Systems 公司的 IDL，Silicon Graphics 公司的 IRIS Explorer IBM 公司的 Visualization Data Explorer 等，这些软件都能够较好的支持多维信息的显示。而国内对于多维信息的可视化研究，尤其是算法研究，仍处于起步阶段。

4.1.3　一个简单的可视化实例

本节将介绍一种基本的可视化技术——散点图，并以此来说明多维信息的可视化，并为后续章节 SOM 可视化的“出场”作铺垫。

XY 散点图（也称为散点图）将值序列显示为一组点。值由点在图表空间中的位置表示。类别由图表中的不同点表示。散点图通常用于比较跨类别的非重复值。 有三种类型的散点图：XY 散点图、折线散点图以及平滑线散点图。

XY 散点图：

XY 散点图根据值序列的 X 值和 Y 值将每个值序列显示为图表空间中的数据点。虽然类别组和序列组是可选的，但是必须至少选择其中一种，才能在图表中显示有意义的数据。

折线散点图：

除了数据点是由直线连接外，折线散点图与 XY 散点图相同。

平滑线散点图：

除了数据点是由曲线连接外，折线散点图与 XY 散点图相同。

实验采用 Iris 数据集，共 3 类：Iris Setosa，Iris Versicolour，Iris Virginica。每类个 50 条数据，每条数据有 5 个属性值，其中 4 个数值属性（单位 cm）：sepal length，sepal width，petal length，petal width，一个类属性。你可以通过站点 http://www.ics.uci.edu/~mlearn/databases/iris/取得样本数据。

下表 4.1 是它的数学性质统计表。

表 4.1　Iris 数据集数据统计特性

	最小值	最大值	均值	方差	类相关系数
sepal length	4.3	7.9	5.84	0.83	0.7826
sepal width	2.0	4.4	3.05	0.43	-0.4194
petal length	1.0	6.9	3.76	1.76	0.9490
petal width	0.1	2.5	1.20	0.76	0.9565

由于散点图是通过样本数据属性的其中某两个特性组成 X，Y 轴来观察样本。因此，对于 d 维样本来说，就有 $d(d-1)/2$ 幅散点图。图 4.1，4.2 是其中两幅。从图 4.1 发现，样本数据点的分布，较为杂乱，并不能很好的发现数据点之间的类别。而从图 4.2 却较为完美的体现了样本类间的差别。

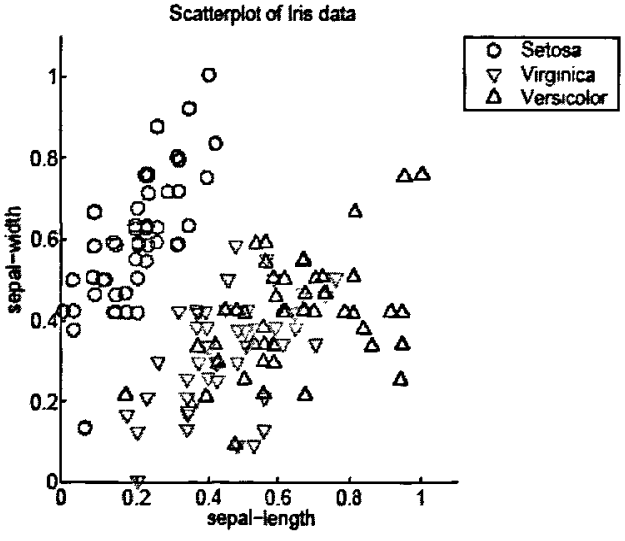


图 4.1　sepal width- sepal length 散点图

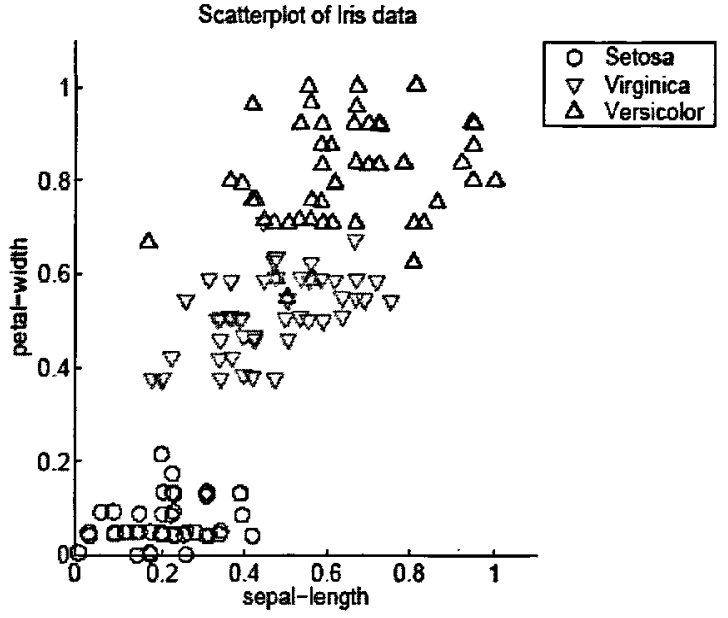


图 4.2　petal width- sepal length 散点图

作为一种可视化手段，散点图以其简单明了的原理而受到大众的欢迎，对于低维数据的表示是一个较为有效的办法。但在高维数据样本的表征时，却由于需要多幅散点图来说明，且表示结果并不一定能如上图 4.2 所示很好的将数据类别区分。而从上一章的分析，我们知道 SOM 却可以实现这个目标。下一节，我们将介绍 SOM 的多维可视化方法。

第二节　SOM 的多维可视化方法

4.2.1　概述

在多维信息的可视化中，SOM 的作用主要体现在两个方面：

(1)在聚类分析和多变量可视化中，SOM 的原型矢量（即神经元权值）可以看成样本数据的代表，原始的样本数据被替换成了维数较低的数据集，因此，我们可以认为根据 SOM 原型矢量获得的可视化信息与原始数据是保持一致的。

(2)在其它的技术中，将 SOM 看成是数据的模型，我们可以将数据与这个

模型进行比较，也可以根据模型反演各种样本数据集。

基于 SOM 的多维可视化模型可以用图 4.3 表示。多维信息经过 SOM 变换，可以用多种方式表示降维后的信息，用户能够通过交互的手段观察数据的相关性。下面分别介绍 SOM 进行多维信息可视化的实现方法。

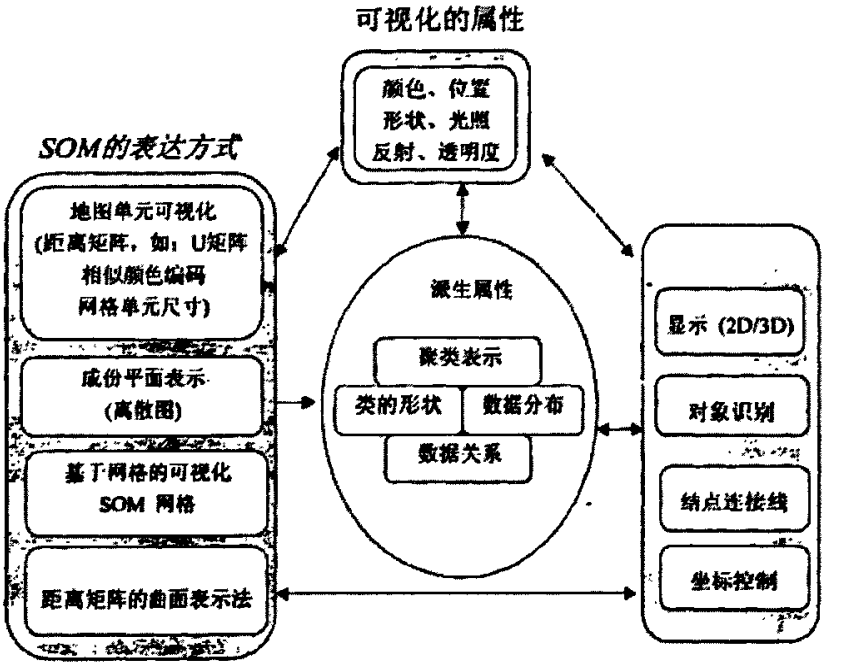


图 4.3　SOM 的可视化模型

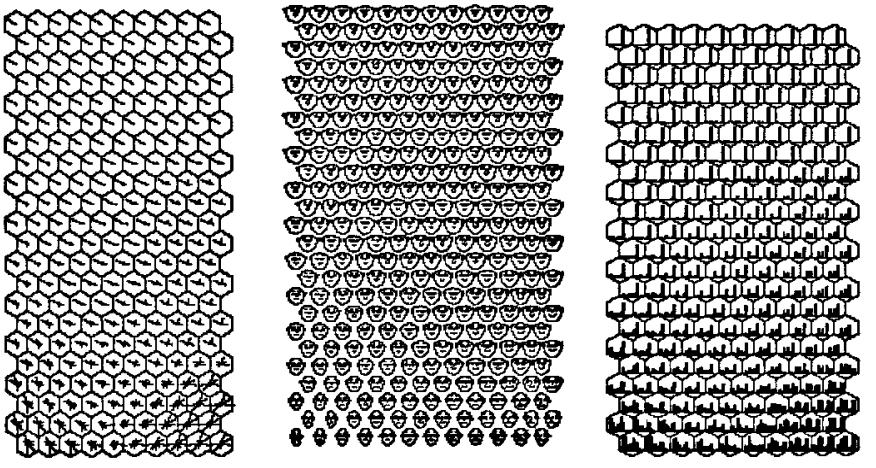
4.2.2　SOM 可视化平台

SOM 形成了一个数据的低维映射，有序的映射网格可以表示原始的样本数据，相邻区域的数据相似度较高，而距离较远的数据，其相似性也很小。总的来说，地图网格具有以下属性：

- 网格的形状是预先定义的，这就保证了每个网格单元具有唯一的位置和相同的大小，因此各种不同的图形可以很方便的插入到网格中，而不会引起相互重叠。图 4.4 是一个地图网格的示意图，网格中填充了能够表示多维信息的扇形图案。
- 拓扑结构的数据密度是原始数据密度的近似。地图网格提供了一种自动

调节数据密度的方法。从图 4.5 可以看出，用颜色表示的 PCA 法不能显示全部数据(特别是数据密集的区域)，而 SOM 则能够方便的显示全部数据。

- 地图网格保持了数据之间的拓扑关系，但是不能体现原始数据之间的相对距离，存在着一定的拓扑映射误差。
- 地图网格在原始空间形成了一个非线性程度非常高的 2 维折叠，因此，很难根据原始的样本数据解释输出空间中各个坐标轴的含义。



(a)扇子图　　(b)Chernoff 脸　　(c)柱状图

图 4.4　地图网格示意图

(a)PCA 的可视化　　(b)颜色表示　　(c) U 矩阵

图 4.5　PCA 与 SOM 的可视化表示

每个网格单元都有一套相关的属性数据集，如拓扑空间上原型矢量的数据值等。SOM 可视化的方法在所有网格单元上都显示这些数据，因此允许对不同的网格单元进行比较。通常，几个显示不同属性数据值的网格单元都是并排排

列的。按照地图网格的坐标可以将可视化的结果串联到一块，但是这些具体的坐标并没有清晰的物理意义。

4.2.3 聚类可视化

数据云形状和聚类结构的可视化技术主要是通过矢量的投影变换实现的。由于 SOM 网格的形状都是预先定义的，因此，它并不适合于显示数据云，原型矢量必须通过其它投影方法映射到低维空间。除了坐标的物理转换外，人们还采用颜色编码技术来实现数据云的可视化。

在 SOM 中，最常用的数据云形状和聚类结构可视化的方法是距离矩阵法(Distance Matrices)，这种方法采用下式计算各个单元 i 和其各邻居单元 N_i 之间的距离：

$$D_i = \{\|m_i - m_j\| \mid j \in N_i, j = i\}$$

每个网格单元的距离(可取平均值)采用特殊的颜色进行编码，这样就可以实现数据云聚类结构的可视化。

此外，由于网格的结构都是规则的，因此，还可以采用标准距离矩阵(Unified Distance Matrix，简称 U 矩阵)来实现对所有网格单元与其邻居之间距离的可视化。用 U 矩阵可以很容易的在网格单元及其所有邻居之间标定一个可视化的符号，如可以用特定的颜色来表示网格单元与它所有邻居之间的距离。通常情况下，用较深的颜色表示网格单元之间距离较远，也就是说它们所对应的输入空间的原始数据差异很大，而用较浅的颜色来表示单元之间距离很近，即它们所对应的原始数据具有很大的相似性。U 矩阵可以反映数据集的聚类关系，由于原型矢量的映射与原始数据的随机密度函数相关，邻居距离 D_i 与数据密度成反比，因此聚类结构的边界形态可以看作是山脊，它代表距离远的单元。聚类结构的边界将距离近的网格单元分开，被分开的网格单元的形态类似于山谷。在上一章，已经简单说明了它的性能。

第三节　SOM 可视化聚类仿真

4.3.1　二维数据

本实验选取一个二维数据集，如图 4.6 所示。样本点数 n=770。数据样本包含密度不同的 6 个类，其中 4 个可以看作噪声数据。该数据集的聚类也可以看作是奇异点检测。

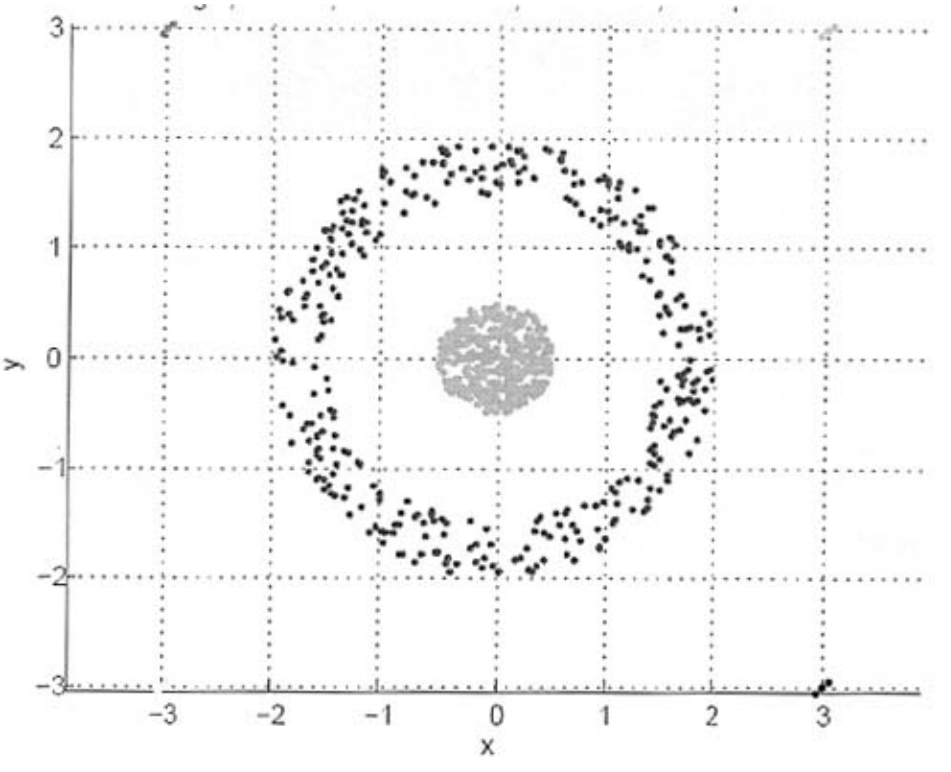


图 4.6　二维数据样本点分布图

通过对样本点数据的 Z-变化和归一化，竞争层网格取为 50*82，采用高斯邻域函数，开始邻域半径为 24，并最终逐渐到 1，学习速率从初始的 0.5 逐渐减小到 0.1。

得到仿真结果如图 4.7 所示。从图中我们可以清晰地发现，在拓扑层上的神经元从大体上分成了 2 个大类加 4 个小类（小圈表示奇异点）。其中的白点表示着相应的最佳匹配神经元（BMU）。

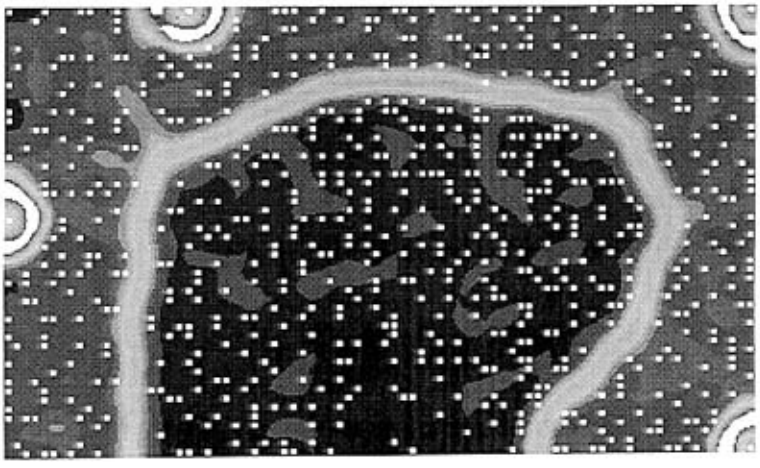


图 4.7　二维数据样本的仿真结果

4.3.2　高维数据

实验采用KDD-99 数据集(Knowledge Discovery in Databases)，你可以通过站点http://kdd.ics.uci.edu/databases/kddcup99/取得数据。该数据集用于评估在网络入侵检测算法的优劣性。数据集中的攻击包括以下几个类：DoS, R2L(Remote to Local), U2R(User to Root)和Probing。每一次网络连接都可以用41个特征来表示。

DoS 攻击是计算机网络安全中的一个重要难点，其主要通过占用计算机的有限进程来达到中断服务的目的。基于攻击策略，攻击目标可能是文件系统空间，网络带宽或是网络连接。从而达到计算机不能提供正常服务的目的。

根据文献[54][55]分析，我们从 41 个特征向量中选取其中最重要几个特征用来分析检测 DoS 攻击。分别是：

1. Duration
2. Source bytes
3. Destination bytes
4. Count
5. Same service rate
6. Connections with SYN errors
7. Connections-Same service-SYN errors
8. Destination-Host-SYN error rate

9. Destination-Host-Same-Service error rate

为了评估算法的有效性，我们定义 2 个指标：检测率（Detection rate，DR）和误报率（False alarm rate，FAR），见下式。

$$DR = \frac{TP}{TP + FN}, FAR = \frac{FP}{TN + FP}$$

其中

TP：攻击者样本被正确区分到攻击类中的数量

TN：正常情况样本被区分为正常类的数量

FP：正常情况样本被区分到攻击类的数量

FN：攻击者样本被分类至正常类的数量

有效的聚类结果应该是尽可能小的误报率同时又有一个比较高的正确检测率。

本实验采用的网络结构为 160*180，训练步数 50，起始邻域半径取为 79，最终邻域半径为 1。其测试结果如表 4.2，4.3 所示。

表 4.2 测试数据数据集

数据子集	正常连接数	总的 DoS 连接数	DoS 攻击数
数据集 1	29126	10697	3695 smurf 2002 back 5000 neptune
数据集 2	10517	3000	1000 smurf 1000 back 1000 neptune
数据集 3	30180	9816	3695 smurf 2002 back 4000 neptune 20 pod 99 teardrop

表 4.3 测试结果

评价实验	训练数据集	测试数据集	训练参数	DR	FAR
实验 1	数据集 1	数据集 2	160*180/50	98.2%	0.9%
实验 2	数据集 3	数据集 2	160*180/50	99.5%	1.1%

这个结果要比文献[56]中具有更好的 FAR 值。

第五章 总结与展望

第一节 全文总结

随着信息时代的到来，人们每天都将接触到大量的信息。传统的信息处理方式已经很难适应这种海量信息的分析和处理。对于复杂而又庞大的空间信息而言，这个问题显得尤其突出。人们迫切需要有新的手段来分析和处理这些海量的空间信息，从中寻找空间数据的分布规律，帮助人们对与空间信息相关的项目进行辅助决策。

数据挖掘应运而生，它能有效处理数据，挖掘数据中潜在模式，为实际工作和生活提供更多更有用的知识，发挥数据的战略价值。聚类知识发现是数据挖掘的重要工具之一。高维聚类分析是当前聚类知识发现的研究热点，对多种数据应用有重要影响。本文围绕高维聚类分析，针对其相关的关键技术，提出了 SOM 映射聚类算法，并相应研究了聚类问题中常见的一些问题：比如奇异点检测和非线性不可分问题、高维聚类与可视化结果的表达等。

论文的主要工作可以归纳为以下 2 点：

（1） 对基于非监督学习神经网络的自组织聚类方法进行了较为深入研究。对其工作原理、聚类特点、评价准则以及存在的问题进行了较为深入地探讨。

（2） 深入探讨了利用描述相关性的 U 矩阵信息，改进 SOM 算法的聚类性能和可视化特性。通过用经典和标准数据库数据仿真，验证了可视化聚类方法的有效性，并在入侵检测中进行了应用研究。

SOM 作为无监督分析数据的一种有力工具，主要在于其两个很重要的特性：向量化与投影分析，投影特性实现了可视化，而数据向量化为 SOM 的聚类提供了方法基础。

作为可视化工具，SOM 使用映射机制可以显示不同特性的数据。将 SOM 方法引入到多维信息可视化中，它能够很好的保持数据之间的邻居关系，而不是保持数据项之间的距离关系。SOM 的原型矢量可以看成是样本数据的代表，也就是说通过降维以后，SOM 原型矢量获得的可视化信息与原始数据是保持一

致的；此外，我们也可以将SOM看成是数据的模型，可以将数据与这个模型进行比较，也可以根据模型反演各种样本数据。经过SOM训练后可以通过地图网格、成份平面、U矩阵等多种形式展现多维信息，人们可以在此基础上方便地分析多维信息。

事实上，SOM算法提供了如图5.1所示的一个数据提炼过程。即通过SOM算法与其他算法特别是可视化算法的结合，为人们方便了解一些未知的数据提供了便捷的窗口，它通过提取样本数据点内在特性将会得到很大的应用。

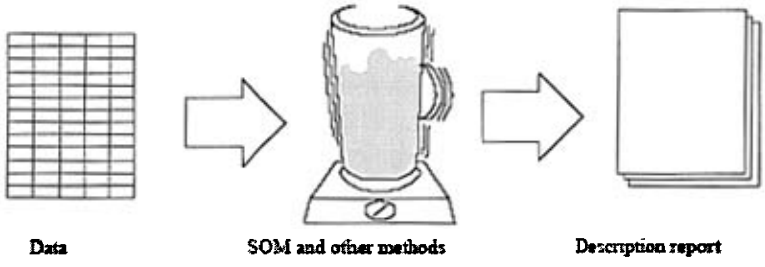


图5.1　数据提取过程

第二节　存在的问题和研究前景展望

高维数据聚类技术的研究方兴未艾，有待研究的内容极为丰富。由于时间的限制，本文研究内容中还有以下几方面值得进一步的深入研究：

1、拓扑映射。在SOM映射过程中，尽管可以保持SOM的邻域特性，但是却不可保持其距离特性，实际上是存在着映射误差的，现今好多学者已经提出了一些指标，来实现拓扑映射误差的最小化，如C-measure，U-Ranking Measure。作为高维样本数据与低维拓扑分布的连接，我个人觉得这是一个很有意思的研究。

2、聚类方法的完善。映射聚类是处理高维数据聚类问题的有效方法，本文简单的介绍了基于SOM的聚类算法，但并没有就算法过程中属性的选择、聚类质量的评价等算法本身的优化去考虑，这个还有待研究

3、数据可视化。可视化技术能简单、自观地反映数据规律，配合高维聚类算法将能加强人机交互功能，充分利用人类知识指导聚类过程。它的技术改进或许会更具有一些商业价值。

致 谢

在整个毕业设计中，我的导师王秀峰教授自始至终给予了我悉心的指导，他用渊博的学识，敏锐的思维和丰富的研究经验，给我提供了很多思路和方法，更重要的是，王老师宽松开明的态度，使我在课题研究最困难的时候依然能够坚持下去。

感谢李朝晖老师在带我教学实习的时候，让我有机会可以与本科生接触，辅导线性代数，让我在锻炼研究能力的同时也锻炼了其他方面的能力。

感谢饶一梅老师，张瀚，边学峰等师兄对我生活与学习上的帮助。感谢郑士芹师姐一直以来对我的帮助和教诲。感谢计算智能与金融信息系统实验室所有师长与同学，3年时间的共处让我们互相了解，增进友谊。

感谢我的家人，不管我的人生路上是风雨交加还是阳光灿烂，你们的关心和支持是我坚持学习和工作的动力。正是你们的关爱让我充满勇气和自信，前行的道路举步维艰，但有你们相伴，我不会退缩。

感谢所有认识或不认识的善良人们，在我撰写论文期间给予我的各种有形或无形的帮助，我的天空因你们而变得精彩和美好，我追寻梦想的脚步因你们而变得轻盈和坚定。

最后，衷心希望实验室可以一步一个脚印，坚持不断的发展下去，越来越好。

赵辉

2007年5月19日

于南开大学伯苓楼

参考文献

[1] BERCHTOLDS, Bohm C, KEIMDA , etal. A cost model for nearest neighbor search in high-dimensional data space[A]. In: Proceedings of the sixteenth ACM SIGMOD Symposium on Principles of database systems[C]. Tucson, Arizona: ACM Press, 1997.

[2] Pavel Berkhin. Survey of Clustering Data Mining Techniques[J]. Accrue Software,2002

[3] C Ding X He, H Zha etal. Adaptive dimension reduction for clustering high dimensional data[C]. In: Data Mining 2002 Proceedings, Second IEEE International Conference on 2002: 147~154.

[4] J.E.Jackson. A User's Guide To Principal Components. John Wiley&Sons,1991

[5] A.K.Jain and R.C.Dubes. Algorithms for Clustering Data. Prentice Ha11,1988

[6] C. Sheikholeslani, S.Chatterjee, and A. Zhang. WaveCluster:A Multi-Resolution Clustering Approach for Very Large Spatial Databases. In Proceedings of the 24th VLDB conference. August 1998.

[7] Chen Ning, Chen An, Zhou Long-xiang. An Effective Clustering Algorithm in Large Tansaction Databases [J]，软件学报，2001, 12 (4) :475~484.

[8] Joliffe I. Principal Component Analysis. Springer-Verlag, NewYork,NY. 1986.

[9] M. Wberry, S.T.dumais, and G.W.O'Brien. Using linear algebra for intelligent information retrieval. SIAM Revies,1995(37):573~595

[10] Parsons L, HaGue E, Liu H. Subspace Clustering for High Dimensional Data: A Review[J]. SIGKDD Explorations 2004:6(l):90~105.

[11] C.C.Aggarwal, J.L.Wolf, P.S.Yu, C.Procopiuc, and J.S.Park. Fast Algorithms for Projected Clustering[C], Proc. ACM SIGMOD, 1999:61~72.

[12] C.C.Aggarwal, P.S.Yu, Finding Generalized Projected Clusters in High Dimensional Space[C], Proc.2000 ACM SIGMOD Int. Conf. Management of Data(SIGMOD'00), 2000:70~81.

[13] Jiawei Han, Micheline Katnber，数据挖掘概念与技术，机械工业出版社，2001.

[14] R.Agrawal, J.Gehrke, D.Gunopulos, and P.Raghavan. Automatic Subspace Clustering of High Dimensional Data for Data Mining Applications[C], Proc. ACM SIGMOD, 1998: 94~105, ACM Press.

[15] 杨风召.高维数据挖掘中若干关键问题的研究：[博士学位论文]. 上海:复旦大学计算机与信息技术系，2003.

[16] Cheng C., Fu A, and Zhang Y. Entropy-based subspace clustering for mining numerical data[C]. In Proceedings of the 5th ACM SIGKDD, 1999:84~93, ACM Press.

[17] S.Goil, H.Nagesh, and A.Choudhary. Mafla: Efficient and scalable subspace clustering for very large data sets. Technical Report CPDC-TR-9906-010, Northwestern University, 2145

Sheridan Road, Evanston IL, 60208, June 1999.

[18] Man Lung Yiu, Nikos Mamoulis. Iterative Projected Clustering by Subspace Mining. IEEE Trans. On KNOWLEDGE AND DATA ENGINEERING, 2005, 17(2):176~189.

[19] Cheng ChurchBiclustering of expression date[C]. In: Proceedings of International Conference on Intelligent System for Molecular Biology, 2000.

[20] Wang Wang, Yang Yu. Clustering by pattern similarity in large data sets[C]. In: Proceedings of ACM SIGMOD International Conference on Management of Data 2002.

[21] Kohonen T. Self-organizing maps, 3rd ed. Berlin: Springer-Vrelag, 2001.

[22] Kraaijveld M A, Mao J, and Jain A K. A nonlinear projection method based an Kohonen topology preserving maps [J]. IEEE Trans. Neural Networks. 1995, 6: 548~559

[23] Oja E. A simplified neuron model as a principal component analyzer [J]. J. Math. Biol. 1982, 15:267~273

[24] Baldi P and Hornic K. Neural networks and principal component analysis from examples without local minima [J]. IEEE Trans. Neural Networks. 1989, 2: 53~58.

[25] Juha Vesanto and Esa Alhoniemi. Clustering of the Self-Organizing Map [J]. In IEEE Transactions on Neural Networks, 2000, 11 (3) :586~600.

[26] Teuvo Kohonen and Panu Somervuo. Self-organizing maps of symbol strings[J]. Neurocomputing, 1998(21):19~30.

[27] Samuel Kaski. Fast winner search for SOM-based monitoring and retrieval of high-dimensional data[C]. In Proceedings of ICANN99, Ninth International Conference on Artificial Neural Networks, 1999(2):940~945. IEEE, London.

[28] Teuvo Kohonen, Samuel Kaski, Krista Lagus, Jarkko Salojärvi, Jukka Honkela, Vesa Paatero, and Antti Saarela. Self organization of a massive document collection [J]. IEEE Transactions on Neural Networks, 2000(11):574~585

[29] Teuvo Kohonen, Samuel Kaski, Krista Lagus, Jarkko Salojärvi, Jukka Honkela, Vesa Paatero, and Antti Saarela. Self organization of a massive text document collection [J]. In Erkki Oja and Samuel Kaski, editors, Kohonen Maps, 1999:171~182. Elsevier, Amsterdam.

[30] R. D. Lawrence, G. S. Almasi, and H. E. Rushmeier. A Scalable Parallel Algorithm for Self-Organizing Maps with Applications to Sparse Data Problems [J]. Data mining and knowledge discovery, 1999, 3(2):171~195.

[31] Alahakoon D, Halgamuge S K. Dynamic Self-organizing Maps with Controlled Growth for Knowledge Discovery [J]. IEEE Transactions on Neural Networks, 2000, 11(3): 601-614.

[32] Fritzke B. Growing Cell Structures-A Self-organizing Network for Unsupervised and Supervised Learning [J]. Neural Network, 1994,7(9): 1411~1460.

[33] Choi D, Park S. Self-creating and Organizing Neural Networks[J]. IEEE Transactions on Neural Networks, 1994, 5(4): 561-575.

[34] H.Jin etal. Expanding self- Organizing map for data visulation and cluster analysis[J]. Information Sciences. 2004:157~173.

[35] DeSieno D. Adding a Conscience to Competitive Learning[J]. IEEE International Conference

on Neural Networks, 1988, 1(6): 117~124.

[36] 尹峻松, 胡德文, 陈 爽等. DSOM: 一种基于NO 时空动态扩散机理的新型自组织模型[J]. 信息科学, 2004, 34(10): 1094-1109.

[37] Xiao X. Gene Clustering Using Self-organizing Maps and Particle Swarm Optimization[C]. Parallel and Distributed Processing Symposium, 2003, 4.

[38] Hussin M F, Kamel M. Document Clustering Using Hierarchical SOMART Neural Network[J]. Proceedings of the International Joint Conference on Neural Networks, 2003, 3(6): 2238-2242.

[39] 孙 放, 胡光锐, 高 军. SOM 结合 MLP 的神经网络语音识别系统[J]. 数据采集与处理, 1996, 11(2): 119-122.

[40] Kaski S, Venna J , Kohonen T. Coloring that reveals high-dimensional structures in data[J]. International Conference on Neural Information Processing, 1999, 2: 729~734.

[41] Kaski S, Honkela T, Lagus K, Kohonen T. Websom-self-organizing maps of document collention[J]. Neurocomputing, 1998, 21(1-3):101~117.

[42] Duda, R.O, Hart, P.E, and Stork, D.G. Pattern Classification (2nd Endition). 李宏东，姚天翔等译，北京：机械工业出版社，2003

[43] Eidenberger, H. Distance Measure for MPEG-7 based Retrieval. Proceeding of the 5th ACM SIGMM international workshop on Multimedia information retrieval, 2003.15(6):130~137.

[44] Lance, G. N., and Williams, W. T. Mixed Data Classificatory Programs. Agglomerative Systems Australian Computer Journal, 1967, 9:373~380.

[45] Glenn W. Milligan and Martha C. Cooper. A study of standardation of variables in cluster analysis. Journal of Classification, 1988, 5:181~204.

[46] Glenn W. Milligan and Martha C. Cooper. An Examination of Procedures for Determining the Number of Clusters in a Data Set. Psychometrika, 1985, 50(2):159~179.

[47] C.M. Bishop. Neural Networks for Pattern recognition. Oxford University Press, 1955.

[48] K.Rose, F. Gurewitz, and G. Fox. Statistical mechanics and phase transitions in clustering. Physical Rev. Lett., 1990, 65(8):945-948.

[49] T.M. Martinetz, S.G. Berkovich, and K.J. Schulten. "neural-gas" network for vector quantization and its application to time-series prediction. IEEE Transaction on Neural Networks, 1993, 4(4):558~569.

[50] 马光志，杨春. 利用 U 矩阵对 SOM 网络的处理. 计算机工程与设计. 2006,2:654-655.

[51] A. Ultsch and H. P. Siemon. Kohonen's self organizing feature maps for exploratory data analysis. In Proc. INNC'90, Int. Neural Network Conf., 1990, 305-308. Dordrecht, Netherlands, Kluwer.

[52] Thanaknrn Naenna. Data Mining Application for Self-Organizing Maps [D]. New York: Rensselaer Polytechnic Institute, 2003.

[53] 谢茂强. 分类问题中的新类识别与分类器更新问题方法研究. 南开大学博士学位论文，2007.

[54] S. Mukkamala, A. H. Sung, "Identifying significant Features for Network Forensic Analysis

Using Artificial Intelligent Techniques", International Journal of Digital Evidence, Winter 2003, Vol. 1, Issue 4.

[55] C. Douligeris, A. Mitrokotsa, "DDoS attacks and defense mechanisms: classification and state-of-the-art", Computer Networks, 2004, 44 (5):643~666.

[56] A. Ultsch, "Maps for visualization of high-dimensional Data Spaces", Proc. WSOM, Kyushu, Japan, 2003, 225-230.

[57] Tenenbaum J B, Silvam V D, and Langford J C. A global geometric framework for nonlinear dimensionality reduction. Science. 2000, 290: 2319~2323

[58] 任若恩，王慧文. 多元统计数据分析-理论、方法、实例. 北京：国防工业出版社，1997.

[59] Borg I and Groenen P. Modern multidimensional scaling: theory and application. New York: Springer-Uerlag. 1997

[60] 杨行峻，郑君里.人工神经网络与盲信号处理.北京:清华大学出版社，2003

[61] Noecker, M., Moerchen, F., Ultsch, A.: Fast and reliable ESOM ESOM learning, In Proc. ESANN. 2006.

[62] Ultsch, A., Moerchen, F.: ESOM-Maps: tools for clustering, visualization, and classification with Emergent SOM, Technical Report Dept. of Mathematics and Computer Science, University of Marburg, Germany, No. 46, 2005.

[63] Ultsch, A., Herrmann, L.: Architecture of emergent self-organizing maps to reduce projection errors, In Proc. ESANN 2005, 1~6.

[64] Ultsch, A.: Maps for the Visualization of high-dimensional Data Spaces, In Proc. WSOM'03, Kyushu, Japan, 2003, 225-230

[65] 杨行峻，郑君里. 人工神经网络.北京:高等教育出版社，1992

个人简历

一、基本情况

姓　　名：赵辉

性　　别：男

出生年月：1981 年 10 月

籍　　贯：浙江余姚

专　　业：控制理论与控制工程

二、教育背景

2004 年 9 月至今：　南开大学计算智能与金融信息系统实验室
攻读控制理论与控制工程硕士学位

2000 年 9 月~2004 年 7 月：　北京理工大学自动控制系
攻读自动化学士学位

三、发表论文

1．赵辉，王秀峰. LM 算法在传感器数据融合中的应用. 传感器技术. 2005

基于SOM的可视化聚类研究

作者：赵辉
学位授予单位：南开大学
被引用次数：1次

参考文献(2条)

1. 尹峻松,胡德文,陈爽,周宗潭 DSOM：一种基于NO时空动态扩散机理的新型自组织模型[期刊论文]-中国科学E辑 2004(10)

2. 马光志,杨春 利用U矩阵对SOM网络的处理[期刊论文]-计算机工程与设计 2006(04)

本文读者也读过(9条)

1. 俞蓓.王军.叶施仁.YU Bei.WANG Jun.YE Shi-Ren 基于近邻方法的高维数据可视化聚类发现[期刊论文]-计算机研究与发展2000,37(6)

2. 袁云 基于SOM的可视化聚类挖掘应用研究[学位论文]2009

3. 杨一山 基于样条模型重建体数据场的造型研究[学位论文]2007

4. 廖广兰.史铁林.姜南.刘世元.Liao Guanglan.Shi Tielin.Jiang Nan.Liu Shiyuan 基于SOM网络的特征选择技术研究[期刊论文]-机械工程学报2005,41(2)

5. 蔡小琳 AdS背景下ⅡB超弦的KRR参数化[学位论文]2006

6. 鲁毅.张雪峰 论宇宙的无限性[会议论文]-2002

7. 陈丹 基于SOM的多维数据可视化研究与实现[学位论文]2010

8. 孙艳玲 论法律意识形态[学位论文]2010

9. 白耀辉.陈明.BAI Yao-hui.CHEN Ming 利用自组织特征映射神经网络进行可视化聚类[期刊论文]-计算机仿真 2006,23(1)

引证文献(1条)

1. 魏冬冬,陈迎春,孙刚 基于神经网络的RANN竞争模型在内饰视觉工效研究中的应用[期刊论文]-复旦学报（自然科学版） 2014(05)