中国近代第一所大学
FOUNDED IN 1895

天津大学

TIANJIN UNIVERSITY

硕士学位论文

M. S. DISSERTATION

学科专业：模式识别与智能系统
作者姓名：赵智超
指导教师：路志英 副教授

大津大学研究生院

2005 年 1 月

中文摘要

沙尘暴历史数据集具有场分布，维度高，数据量大的特点。而数据挖掘正是用来从大量数据中发现感兴趣的知识从而支持决策的良好方法。因此本文研究的主要内容就是如何用数据挖掘技术来进行沙尘暴的智能预报。围绕该项研究内容作了如下工作：

1 采用主成份分析和改进的聚类方法完成了数据样本的降维。两种方法的性能都比原有降维方法有较大提高。主成份分析方法在保证性能的前提下，运算简单，能在很短的时间内完成处理。改进的聚类方法是本文提出的一种将聚类与综合预报模型相结合的降维方法，在这几种降维方法中，其效果最佳。

2 本文采用改进的 BP 神经网络，k-最近邻法和支持向量机等方法建立了沙尘暴预报模型。改进的 BP 神经网络使用 Levenberg-Marquardt（LM）算法，权值训练过程具有快速收敛性，并获得最好的预报性能。k-最近邻法在保持较高的预报性能的前提下，具有最好的稳定性。支持向量机只是初步实现，虽然效果不甚理想，但表现出很好的避免过拟合能力，具有进一步改进的空间。

3 本文采用 Bayesian 规则化泛化技术对 LM 网络进行了泛化，消除了随机权初值对性能造成的不稳定的影响，成功地使 LM 网络的 CSI 值稳定在一个很小的范围，增强了系统的稳定性。

论文最后对所得系统的不足，如支持向量机核函数的选择及其参数的调整，进行了总结，并给出了进一步深入研究的方案，即典型样本和非典型样本的剪辑方法。

关键词：特征提取 LM 算法 k-最近邻法 支持向量机 神经网络泛化

Abstract

The dust storm historical data set has the characteristics of field distribution, high dimensionality, and huge data volume. Data mining is a good method used to discover knowledge of interest right from large amount of data to support decision-making. So the main content of research in this paper is how to use data mining technology to intelligently forecast dust storm. The works evolved in this content are listed below:

1 Dimensionality of data samples is reduced using principal component analysis (PCA) and the improved clustering method. The performance of both methods is enhanced much with respect to the original dimensionality reduction (DR) method. Under the condition of guaranteed performance, the operations of PCA are simple, and processing can be finished in short time. The improved clustering method is a DR method proposed in this paper, which combines clustering and ensemble forecast model. Among these DR methods, its performance is best.

2 The dust storm forecast models are created using the improved backpropagation neural network (BPNN), k-nearest neighbour, and support vector machine (SVM). The improved BPNN uses Levenberg-Marquardt (LM) algorithm, where the weights training process can converge rapidly, and the best performance is achieved. Under the condition of keeping relatively high forecast performance, k-nearest neighbour has the best stability. SVM is implemented preliminarily. Although the result is not ideal, it shows the better ability of avoiding overfitting.

3 LM network is generalized using Bayesian regularization. It can remove the instability effect of the random initial weights on performance, and successfully make the CSI value of the LM network stable in a small range.

In the end, the deficiencies of the obtained system, such as the selection of the SVM kernel function and the adaption of its parameters, are summarized. And further research scheme is given, i.e. the method of editing the typical and non-typical samples.

Key Words: feature extraction, LM algorithm, k-nearest neighbour, support vector machine, generalization of neural network

独创性声明

本人声明所呈交的学位论文是本人在导师指导下进行的研究工作和取得的研究成果，除了文中特别加以标注和致谢之处外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得 天津大学 或其他教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示了谢意。

学位论文作者签名：赵智超 签字日期： 05 年 1 月 10 日

学位论文版权使用授权书

本学位论文作者完全了解 天津大学 有关保留、使用学位论文的规定。特授权 天津大学 可以将学位论文的全部或部分内容编入有关数据库进行检索，并采用影印、缩印或扫描等复制手段保存、汇编以供查阅和借阅。同意学校向国家有关部门或机构送交论文的复印件和磁盘。

（保密的学位论文在解密后适用本授权说明）

学位论文作者签名：赵智超 导师签名：路志英

签字日期：05 年 1 月 10 日 签字日期：05 年 1 月 10 日

第一章 绪论

1.1 沙尘暴预报

1.1.1 沙尘暴预报的目标

沙尘暴是沙暴和尘暴两者兼有的总称，是指强风把地面大量沙尘卷入空中，使空气特别混浊，水平能见度低于 1Km 的天气现象[1]。专家研究指出，沙尘暴形成有 3 个基本条件，一是大风，这是形成沙尘暴的动力条件；二是地面上裸露的沙尘物质，它是沙尘暴的物质基础；三是不稳定的空气状态，这是重要的热力条件[2]。

沙尘暴使大片农田或受沙埋、或遭风蚀刮走沃土，或者农作物受霜冻之害，加剧土地沙漠化，对大气环境造成严重污染，对生态环境造成巨大破坏，对交通和供电线路产生重要影响，给人民生命财产造成严重损失[2]。

正因为这些严重危害，所以我们急需发展沙尘暴天气预报的新方法，建立具有较高业务应用价值的沙尘暴监测、预警系统，并及时发布信息，以利于提前安排好生产、交通和群众生活，尽可能减少损失。要达到这一目标，我们所要做的就是深入研究前面所提到的三个条件与沙尘暴产生之间存在的潜在关系，做到能够根据事先获得的天气状况信息，尽可能准确地预知沙尘暴的发生。

1.1.2 沙尘暴预报的研究现状

天气预报主要有两种方法，数值预报和统计预报。数值预报是在给定的初始条件和边界条件下，通过积分描述大气运动定律的流体动力学热力学方程组，得到未来时刻的气象要素分布。统计预报是利用大量历史气象观测数据中所蕴含的天气演变的内在规律，提取对未来天气有指示作用的情报。[3]

沙尘暴预报作为各种天气现象预报中的一种，也自然要采取这两种预报方法。目前沙尘暴预报的主要研究是数值预报。美国空军天气局（AFWA）使用 Colorado 大学的大气群落气溶胶与辐射模型（CARMA）对中东的沙尘暴进行预报。CARMA 的输入为第五代 Penn 州中尺度气象模型（MM5）[4]。其它沙尘气溶胶模型与此类似，都使用标准天气模型的数据作为输入。Malta 大学和 Athens 大学使用 Eta 天气模型的修改版对地中海地区的沙尘暴进行预报[5]。美国海军研究实验室使用海军气溶胶分析与预测系统（NAAPS）对沙尘进行每日预报，这一系统使用来自海军运行全球大气预测系统（NOGAPS）的每日天气预报产品[6]。由 Yaping Shao 提出的沙尘预报模型使用来自中国国家气象中心的

天气数据预报中国和东亚的沙尘暴[7]。

1.1.3 统计预报[3]

虽然目前大多数沙尘暴预报采用的是数值预报方法，但是沙尘暴统计预报方法的研究仍然是非常必要的。

首先，数值预报是建立在天气现象已知的因果关系基础之上的，但现有的人类认识只是自然界复杂联系的一小部分，因此数值预报必然因其不完整性而存在误差。而统计方法通过分析历史资料找出总体上的规律性，避免进行因果分析时不可避免的片面性。

其次，数值预报所建立的大气动力学模型是对大尺度运动趋势的模拟，因此数值预报是大形势的预报，局部的不一致性和细节都已经被平滑掉了。而各地的天气预报机构恰恰需要的是这些反映各地具体情况的信息。统计预报具有尺度上的可放缩性，当研究大范围的趋势时，可采用大范围的历史数据，当要做本地区的预报时，则使用本地的历史记录。但统计预报也有它的不足，对偶然的历史上未出现过的天气状况无法做出准确的预报。

再次，数值预报模型由大量非常复杂流体动力学公式组成，运行时需要进行大量的运算，因此对计算环境的要求非常高，通常只能在超级巨型机上部署，需要大量的建设和运行经费，而统计预报的计算量通常一台普通的工作站就可以应付。因此，数值预报多由一个国家或大区域的气象中心集中进行，将结果向全国统一发布；而各地区的气象台站在经费有限的情况下，可以用统计模型进行本地预报。

最后，从其它天气现象的预报发展来看，近年来的趋势是数值预报和统计预报的结合，即集成预报。所以，就必须对它进行统计预报的研究。

1.2 数据挖掘简介[8]

数据库中的知识发现（KDD）是从存储在数据库，数据仓库或其它信息储存库中的大量数据中发现感兴趣的知识的过程。而数据挖掘是数据库中知识发现过程中关键的一步。KDD 流程图如图 1－1 所示。许多人也将数据挖掘看成是 KDD 的同义词。

数据挖掘的功能用于指出数据挖掘任务中要发现的模式的种类。数据挖掘功能和它们可以发现的模式种类包括如下几种。

概念描述 对与数据关联的类或概念进行的有用的总结性描述。这些描述可以从数据表征或数据区分中得出。数据表征是对目标数据类的总体特性或特征

天气数据预报中国和东亚的沙尘暴[7]。

1.1.3 统计预报[3]

虽然目前大多数沙尘暴预报采用的是数值预报方法，但是沙尘暴统计预报方法的研究仍然是非常必要的。

首先，数值预报是建立在天气现象已知的因果关系基础之上的，但现有的人类认识只是自然界复杂联系的一小部分，因此数值预报必然因其不完整性而存在误差。而统计方法通过分析历史资料找出总体上的规律性，避免进行因果分析时不可避免的片面性。

其次，数值预报所建立的大气动力学模型是对大尺度运动趋势的模拟，因此数值预报是大形势的预报，局部的不一致性和细节都已经被平滑掉了。而各地的天气预报机构恰恰需要的是这些反映各地具体情况的信息。统计预报具有尺度上的可放缩性，当研究大范围的趋势时，可采用大范围的历史数据，当要做本地区的预报时，则使用本地的历史记录。但统计预报也有它的不足，对偶然的历史上未出现过的天气状况无法做出准确的预报。

再次，数值预报模型由大量非常复杂流体动力学公式组成，运行时需要进行大量的运算，因此对计算环境的要求非常高，通常只能在超级巨型机上部署，需要大量的建设和运行经费，而统计预报的计算量通常一台普通的工作站就可以应付。因此，数值预报多由一个国家或大区域的气象中心集中进行，将结果向全国统一发布；而各地区的气象台站在经费有限的情况下，可以用统计模型进行本地预报。

最后，从其它天气现象的预报发展来看，近年来的趋势是数值预报和统计预报的结合，即集成预报。所以，就必须对它进行统计预报的研究。

1.2 数据挖掘简介[8]

数据库中的知识发现（KDD）是从存储在数据库，数据仓库或其它信息储存库中的大量数据中发现感兴趣的知识的过程。而数据挖掘是数据库中知识发现过程中关键的一步。KDD 流程图如图 1—1 所示。许多人也将数据挖掘看成是 KDD 的同义词。

数据挖掘的功能用于指出数据挖掘任务中要发现的模式的种类。数据挖掘功能和它们可以发现的模式种类包括如下几种。

概念描述 对与数据关联的类或概念进行的有用的总结性描述。这些描述可以从数据表征或数据区分中得出。数据表征是对目标数据类的总体特性或特征

以从数据表征或数据区分中得出。数据表征是对目标数据类的总体特性或特征

天气数据预报中国和东亚的沙尘暴[7]。

1.1.3 统计预报[3]

虽然目前大多数沙尘暴预报采用的是数值预报方法，但是沙尘暴统计预报方法的研究仍然是非常必要的。

首先，数值预报是建立在天气现象已知的因果关系基础之上的，但现有的人类认识只是自然界复杂联系的一小部分，因此数值预报必然因其不完整性而存在误差。而统计方法通过分析历史资料找出总体上的规律性，避免进行因果分析时不可避免的片面性。

其次，数值预报所建立的大气动力学模型是对大尺度运动趋势的模拟，因此数值预报是大形势的预报，局部的不一致性和细节都已经被平滑掉了。而各地的天气预报机构恰恰需要的是这些反映各地具体情况的信息。统计预报具有尺度上的可放缩性，当研究大范围的趋势时，可采用大范围的历史数据，当要做本地区的预报时，则使用本地的历史记录。但统计预报也有它的不足，对偶然的历史上未出现过的天气状况无法做出准确的预报。

再次，数值预报模型由大量非常复杂流体动力学公式组成，运行时需要进行大量的运算，因此对计算环境的要求非常高，通常只能在超级巨型机上部署，需要大量的建设和运行经费，而统计预报的计算量通常一台普通的工作站就可以应付。因此，数值预报多由一个国家或大区域的气象中心集中进行，将结果向全国统一发布；而各地区的气象台站在经费有限的情况下，可以用统计模型进行本地预报。

最后，从其它天气现象的预报发展来看，近年来的趋势是数值预报和统计预报的结合，即集成预报。所以，就必须对它进行统计预报的研究。

1.2 数据挖掘简介[8]

数据库中的知识发现（KDD）是从存储在数据库，数据仓库或其它信息储存库中的大量数据中发现感兴趣的知识的过程。而数据挖掘是数据库中知识发现过程中关键的一步。KDD 流程图如图 1－1 所示。许多人也将数据挖掘看成是 KDD 的同义词。

数据挖掘的功能用于指出数据挖掘任务中要发现的模式的种类。数据挖掘功能和它们可以发现的模式种类包括如下几种。

概念描述 对与数据关联的类或概念进行的有用的总结性描述。这些描述可以从数据表征或数据区分中得出。数据表征是对目标数据类的总体特性或特征

的总结。数据区分是将目标类数据对象的总体特征与来自一个或一组对比类的对象的总体特征进行比较。

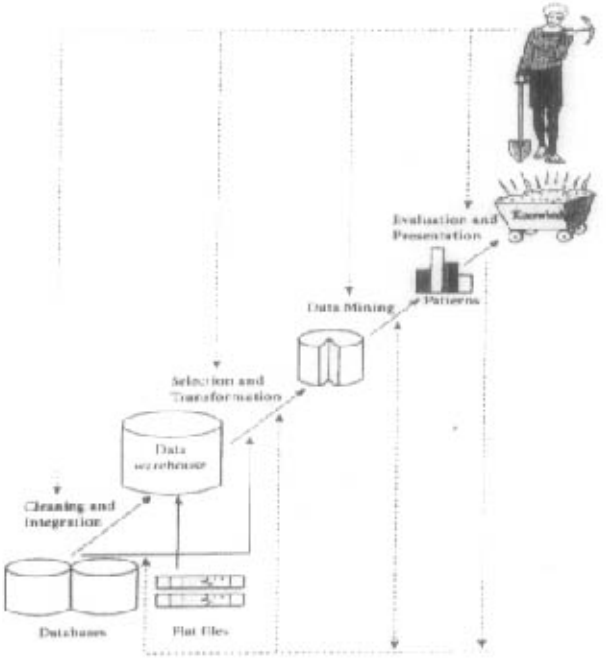


图 1－1 KDD 流程图

关联分析 发现显示给定数据集中频繁一起出现的属性一值条件的关联规则。关联规则的形式为 $X \Rightarrow Y$，即 $A_1 \wedge \cdots \wedge A_m \rightarrow B_1 \wedge \cdots \wedge B_n$，其中 A_i（$i \in \{1,\ldots,m\}$）和 B_j（$j \in \{1,\ldots,n\}$）是属性一值对。关联规则 $X \Rightarrow Y$ 可解释为满足条件 X 的数据可能出现结果 Y。

分类 找到描述和区分数据类或概念的一组模型或函数的过程，目的是为了能够使用模型来预测类标签未知的对象的类。所获得的模型是以一组训练数据的分析为基础的。但是在许多应用中，用户可能希望预测一些丢失的或无法获得的数据值，而不是类标签。当所预测的值是数值数据时，这一功能也经常特别被称为预测。

聚类 分析数据对象，不用咨询已知的类标签。聚类可以用来产生这些标签。对象聚类或分组是以最大化类内相似性，最小化类间相似性的原则为基础的。

离群值 数据库中包含的与数据的总体行为或模型不一致的数据对象。在有些应用中，例如故障检测，对罕见事件比对有规律发生的事件更感兴趣。对离群值数据的分析称为离群值挖掘。

进化分析 对行为随时间变化的对象的规律性或趋势进行描述和建模。

上面介绍了数据挖掘的几个主要功能。对于沙尘暴预报来说，可以利用聚类功能对沙尘暴样本进行数据预处理，利用分类预测功能建立沙尘暴预报模型。从前面的介绍可以看出，在数据挖掘领域，分类和预测的区别并不在于，分类是同时性的，预测有时间上的提前量；而是在于两者的输出不同，分类的输出是类标签，预测的输出是连续值。我们的沙尘暴预报只要求报出有没有沙尘暴，因此应该属于分类。时间上的提前性，通过时间序列分析给予保证（具体论述参见 3.2 节），即只要训练集的输入与目标输出之间存在提前量，在测试时输出就会比输入有提前量。

1.3 已有的工作基础

天津大学模式识别与智能系统研究室对该项内容的研究已有多年。其中，岳斌使用聚类方法从 684 维的数据样本中提取出 40 个特征，再用 BP 神经网络对其进行分类[9]。王汉芝用主成份分析将 40 个特征降为 10 个特征，再用模糊神经网络进行分类，对非典型样本使用另一个模糊神经网络进行分类[10]。

1.3.1 沙尘暴源数据集的特点

本文所采用的数据集是美国环境气象中心（NCEP）提供的，每天一个样本，每个样本的结构是相同的。该数据集有以下特点：

首先，鉴于我国沙尘暴发生的时间和地区特点，将 NCEP 资料圈定在我国西北部从初春到初夏的范围之内，详见表 1-1。

表 1-1 NCEP 资料范围

	范围	说明	沙尘暴日频数
时间	1981~1997 年 2 月 11 日~6 月 10 日	17 年初春至初夏	$\frac{\text{沙尘暴日}}{\text{非沙尘暴日}} = \frac{575}{1469} = 39.14\%$
地域	东经 70°~115° 跨度 45° 北纬 35°~55° 跨度 20°	俄罗斯、蒙古；新疆、内蒙古、甘肃、宁夏、青海、山西、陕西、河北、北京、天津、山东等省市	

其次，在所圈定的地域范围内生成格点场。按照 NCEP 资料每 2.5 度记一格的数据格式，可知表 1-1 给定的选定区域，东西向跨 45°分为 18 格、南北向跨 20°分为 8 格。东西向和南北向的交叉记为一个格点，这样，一个格点场的

数据量就是 19*9=171 个，即每个格点场提供 19×9 的数据阵。

第三，样本是由度量不同物理量的多个格点场组成的。考虑到沙尘暴形成的基本条件，选取 500hpa 的高度、700hpa 的东南风和西北风、850hpa 的温度和比湿共五种格点场作原始数据源。然后，用 850hpa 的温度和比湿推导出 850hpa 的位温，再将东南风和西北风合并在一起，最终得到反映沙尘暴信息的三种物理场，即 500hpa 的高度(H5)场、700hpa 的两个风(UV)场和 850hpa 的位温(SE)场。这样，一个样本的总维数为 171×4=684。

1.3.2 沙尘暴预报系统基础

一、 自距平

自距平是获取场分布的形状特征的一种方法。设矩阵 A 代表一个场分布，A 的所有元素的平均值为 m，将 A 中的每个元素减去 m 所得到的矩阵即为 A 的自距平。

根据专家经验，将所有场转换为其自距平，这些距平场和高度场放到一起构成新的样本集（保留高度场是为了提供值信息）。

二、 挖掘沙尘暴典型模式

历史数据被划分成训练集和测试集两部分。通过训练集中的历史数据，挖掘沙尘暴出现的规律，然后利用这些规律指导沙尘暴的预报。测试集用来模拟系统真实应用时的场景，从中检验沙尘暴预报的准确性能。挖掘典型沙尘暴模式的具体过程：

首先，利用自组织特征映射网络聚类方法对训练集中的强沙尘暴日(出现沙尘暴的站点数目为 9 以上的沙尘暴日)样本进行聚类。根据专家经验，高度场分为两类；高度距平场分为三类、风距平场分为两类、位温距平场分为两类。这样，强沙尘暴日样本被聚为 2*3*2*2=24 种类型，聚类结果如表 1-2 所示。表中类别号的四位数字中，第一位到第四位依次表示高度场的类型、高度距平场的类型、风距平场的类型和位温距平场的类型。例如 1211 表示其样本属于高度场的第 2 类，高度距平场的第 3 类，风距平场的第 2 类和位温距平场的第 2 类。

沙尘暴有强沙尘暴和少站点沙尘暴（出现沙尘暴的站点数目为 10 以下）之分，在强沙尘暴中，又将站点数较多的沙尘暴界定为严重沙尘暴，据各子类中聚集的样本数和沙尘暴严重程度，大体有样本数为 0、样本数少、样本数多以及样本数据较多但含有特别严重的沙尘暴类型几种情况。

1. 对于样本数较多的子类，客观上反映了三种物理场各自特定格局的组合与沙尘暴天气的联系，应该作为典型的沙尘暴模式，如子类 1 和子类 10。

2. 而那些样本数较多但包含着历史上特别严重的沙尘暴天气的子类，其物理场组合格局不容忽略，也应作为典型的沙尘暴模式，如类别 0010 样本数（9 个）虽少于类别 0211（12 个），但其中包括站点数为 63 的严重沙尘暴样本，故选择为子类 2。

由此从 24 个聚类结果中筛选出 10 个沙尘暴子类（模式），如表 1-2 所示。

表 1-2 聚类结果

类别	样本数	站点数	备注	类别	样本数	站点数	备注
0000	35	10~80	子类 1	1000	16	10~58	子类 6
0001	4	13~55		1001	8	10~27	
0010	9	11~63	子类 2	1010	20	10~57	子类 7
0011	0			1011	1	18	
0100	8	10~35		1100	0		
0101	17	11~67	子类 3	1101	11	10~24	子类 8
0110	6	10~33		1110	15	10~37	子类 9
0111	0			1111	9	10~15	
0200	0			1200	0		
0201	15	10~25	子类 4	1201	1	18	
0210	17	10~41	子类 5	1210	1	10	
0211	12	10~21		1211	37	10~49	子类 10

三、提取沙尘暴特征

10 个沙尘暴子模式中，每个模式的三种物理场值的特征及其形的特征都能表征该类沙尘暴天气，因此，10 个模式下全部物理场值的特征和形的特征联合起来可以用来鉴别沙尘暴和非沙尘暴样本。于是

- 计算 10 个沙尘暴模式的高度场及其距平场的中心场，风距平场的中心场以及位温距平场的中心场共计 40 个 $(C_{ij}, i=1,2,\cdots,10, j=1,2,3,4)$。计算公式如下：

$$c_{ij}(k)=\frac{1}{n_i}\sum_{l=1}^{n_i}x_{ij}^{(l)}(k) \tag{1-1}$$

其中 $i=1,2,\cdots,10;\ j=1,2,3,4;\ k=\begin{cases}1,2,\cdots,342 & \text{风的中心场}\\ 1,2,\cdots 171 & \text{其他中心场}\end{cases}$；

n_i — 类 i 中的样本个数；

$x_{ij}^{(l)}(k)$ — 类 i 中第 l 个样本的第 j 个格点阵的第 k 个格点数据；

$c_{ij}(k)$ — 中心场 C_{ij} 的第 k 个格点数据。

• 计算样本的 4 个格点阵与每个子模式（共 10 个）的 4 个中心场的相似度，并将每一个相似度记为该样本的一个特征取值。40 个特征则构成样本的特征向量 Z。

$$Z=(z_{1,1},z_{1,2},z_{1,3},z_{1,4},\cdots,z_{10,1},z_{10,2},z_{10,3},z_{10,4})$$

$$z_{i,j}=\sqrt{\sum_{k=1}^{m}(x_j(k)-c_{ij}(k))^2} \tag{1-2}$$

其中 $i=1,2,\cdots,10;\ j=1,2,3,4;\ m=\begin{cases}342 & \text{风的中心场}\\ 171 & \text{其他中心场}\end{cases}$

$x_j(k)$ —样本第 j 个格点阵的第 k 个格点数据

四、模型建立与测试

预报模型选用 BP 神经网络，其输入层为 40 个神经元，输出层为 1 个神经元。在建立模型的过程中，沙尘暴情况下的输出设为 1；非沙尘暴情况下的输出设为 0。

模型建立后，利用测试集来检验模型的预报性能。检测过程中需要对测试集样本进行自距平和与聚类中心距离的求取，以提取特征，然后将特征输入已训练好的 BP 神经网络模型。模型的输出可决定预报是否出现沙尘暴。

衡量测试结果的指标气象部门常常采用临界成功指数（CSI），即

$$\mathrm{CSI}=\frac{c_f}{c_f+w_f}\times 100\% \tag{1-3}$$

其中，c_f 为正确报出的沙尘暴日数，w_f 是漏报与空报数之和。[11]

五、模型参数的调整

沙尘暴预报模型虽然已经确定，但模型中仍然存在可以调整的参数，如 BP 神经网络隐层的拓扑结构，训练样本的编辑等。通过这些参数的调整，可以改变测试结果 CSI 值。CSI 值的变化反映了系统预报性能的变化。因此，可以通过对某些参数的不断调整，找出它们的最佳取值，使系统预报性能达到最优。

首先，调整 BP 神经网络隐层的拓扑结构。表 1-3 为采用手工试探法试出的

各种拓扑结构的试报结果。其中 90~95 年样本为训练集。96~97 年样本为测试集。由表 1－3 可以看出：40*20*10*1 这种拓扑结构，试报得到的 CSI 值较高，是一种较优的拓扑结构。

表 1-3 各种拓扑结构的试报 CSI 值

第一隐层节点数	第二隐层节点数	试报 CSI(%)
40	20	12.7
30	30	11.1
20	20	21.7
△ 20	10	21.6
20	0	15.1

注：训练参数：学习率 0.4，惯性系数 0.02，迭代 1000 次。

其次，训练样本的编辑也会影响系统的性能，如表 1－4 所示。表 1－4 说明，虽然 90-95 年样本比较接近 96-97 的大气环境，但却未能包含所有沙尘暴类型，神经网对一些类型的沙尘暴缺乏学习机会，故不具有识别能力。适当增加训练集的样本数目，同时也增加了沙尘暴的类型，使模型预报水平提高。

表 1-4 训练样本对预报结果的影响

训练集合年限	样本情况			网络拓扑	拟和 CSI(%)	试报 CSI(%)
	样本总数(A)	沙尘暴样本数(B)	B/A			
90-95 年	715	153	0.214	40*20*10*1	100	22
△81-95 年	1788	548	0.306	40*20*9*1	68.4	25.9

注：训练参数：学习率 0.4，惯性常数 0.02，迭代 1000 次。

测试集合： 96-97 年样本(样本总数: 239 个，沙尘暴日样本 25 个，占总数的 10.5%)。

1.4 本文工作

本文所做工作是国家气象中心“沙尘天气中短期及短时预报系统”研制工作的一部分，任务是在岳斌和王汉芝所做的工作的基础上，进一步提取合理的特征，并建立合适的沙尘暴预报模型，使沙尘暴预报率达到一定的预报精度。

一. 已有工作的分析

1. 分场聚类组合的方法实现了样本聚类，从而使典型模式的挖掘成为可能。但是，组合方法制造了大量冗余，需要在组合结果中进行挑选，才能得到典型

模式。组合方法还可能产生典型模式的丢失。

2．建立了基于人工神经网的沙尘暴预报模型，并从样本、拓扑结构和参数等方面优化预报模型。但是，对神经网络的训练算法及其它分类方法缺乏更多的尝试和必要的分析。

3．仅注意了模型参数的优化，却忽视了模型性能的稳定性和泛化能力，缺乏对网络权值的分析。

二．本文主要工作

针对上述问题，本文做了如下的工作：

1．首先利用主成份分析将 684 维数据直接进行特征提取，实验证明，这种一次性的降维方法虽然简单，但效果比原有方法好。然后借鉴多信息源综合预报模型，用综合预报替换原方法的分场聚类组合方法，减少了冗余特征和信息丢失。

2．对神经网络的 BP 训练算法进行改进，实现了 Levenberg-Marquardt 算法的训练，加快了权值的收敛。另外尝试了两种分类方法——k-最近邻法和支持向量机。虽然性能都没有 LM 网络好，但 k-最近邻法能够稳定在较高的性能，支持向量机还有进一步改进的空间。

3．为提高 LM 网络的稳定性，采用 Bayesian 规则化方法对网络进行了泛化，成功地使系统的 CSI 指数稳定在一个较小的范围（0.34 ± 0.01）内。

第二章 数据预处理

2.1 数据预处理的主要方法[8]

数据预处理是数据挖掘前准备数据的几个步骤的总称，这些步骤都是为了改善数据的质量，进而改善挖掘的质量，增进挖掘过程的效率和简易性。这些步骤包括：数据清洗，数据集成，数据转换和数据削减，如图 2－1 所示。其中：

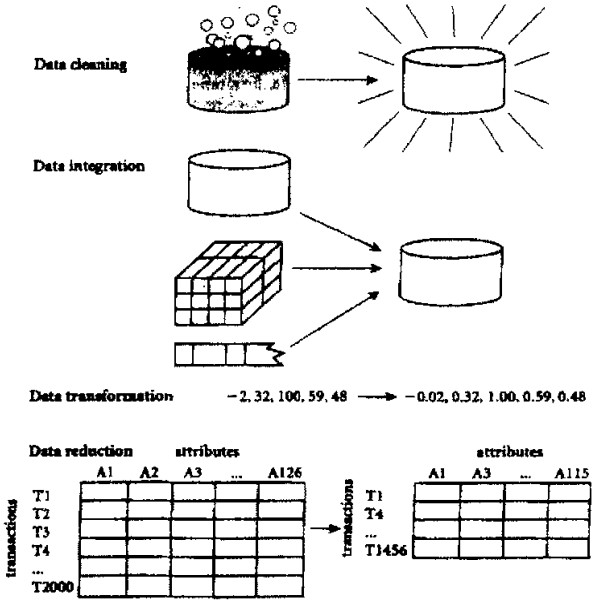


图 2－1 数据预处理的形式

数据清洗 具有不完整数据（缺少感兴趣的属性值或只包含聚集数据）的填充，消除或平滑噪声数据（包含误差或远离期望的离群值），识别或移除离群值，解决不一致数据（包含用于分类的代码在部门间的差异性）的作用。

数据集成 集成所需分析问题中涉及到多个数据源（包括数据库，数据立方体或文件等）。但是，在数据集成过程中，代表给定概念的某些属性可能在不同的数据库中有不同的名称，造成数据的不一致性和冗余，这可能减慢或搞乱知识发现过程。因此，必须采取措施避免数据集成中的冗余。

数据转换 该操作包括规范化和聚集，对挖掘过程的成功起很大作用。如果采用基于距离的挖掘算法（例如神经网络，最近邻分类器或聚类）进行数据分

析，就要对数据进行规范化（即放缩到特定的域，如[0, 1]），使这些挖掘算法得到更好的结果。

数据削减 具有减小数据集的大小，但又能产生相同（或几乎相同）的分析结果的作用。庞大的数据集肯定会减慢挖掘过程，因此，数据削减是数据挖掘过程中必不可少的环节。

本文所涉及的沙尘暴数据是根据 NCEP 原始数据，在一定的气象知识指导下，通过对观测数据中漏值，噪声，不一致性，多源数据集成时产生的冗余加工后得到的[9]。因此，本文不再需要进行数据清洗和数据集成处理，只需完成数据削减和数据转换步骤。其中数据削减步骤尤为重要，这是因为原始数据每个样本的维数太高（684 维），所以必须将维数削减，使挖掘过程不致过慢。

数据削减策略包括：

1 数据立方体聚集 在数据立方体的建造过程中，对数据应用聚集操作。

2 降维 检测并移除无关，弱相关或冗余的属性或维。

降维也被称为特征选择。最普通的一种特征选择方法，就是一个搜索符合事先约定的最优性准则的特征子集的过程。用符号 F_t 表示原始的 t 个特征的集合，用 F 表示任意一个特征子集，它所包含的特征个数 $|F|$ 等于需要的空间维数 d。用 $J(F)$ 表示选择过程中使用的准则函数。特征选择问题就变成一个寻找满足下面条件的子集 F^* 的过程。

$$J(F^*) = \max_{F \subseteq F_t, |F|=d} J(F)$$

这一过程需要解决的两个问题：准则函数的选择和最大值的搜索方法。在很多情况下，准则函数衡量类别的可分性，这时可以使用 Bhattacharyya 距离尺度或 AnovaF 统计方法。搜索方法包括最优方法和次优方法两类。穷尽搜索法和分枝定界法属于最优方法。分枝定界法要求准则函数必须是单调的，被舍弃特征的增加用树的形式表示。遗传算法，直接顺序搜索法和动态顺序搜索法属于次优方法。遗传算法在搜索中继承在前面搜索进程中发现的具有良好属性的父辈子集。直接顺序搜索法可添加进行，也可删除进行，相当于只进行树的一枝。动态顺序搜索法将添加和删除结合起来，是直接顺序搜索法和分枝定界法之间的折中方法。[12][29]

3 数据压缩 用编码机制来减少数据集的大小。它分成两类，有损数据压缩和无损数据压缩。两类的区别是看在压缩的过程中是否有信息损失。两种流行的有效的有损数据压缩方法为小波变换和主成份分析。

离散小波变换（DWT）是一种线性信号处理技术，当应用于一个数据向量时，它会将向量转换为一个相同长度的向量。它的有用性在于，小波变换后的数据可以截短。数据的压缩近似可以通过只存储一部分最强的小波系数得以保

留。DWT 与离散 Fourier 变换（DFT）紧密相关。但是通常，DWT 可以达到更好的有损压缩，即如果给定数据向量的 DWT 和 DFT 保留相同数目的系数，DWT 将提供对原始数据更准确的近似。不像 DFT，小波在空间上相当局部化，对局部细节的转换有帮助。DWT 有几个族，流行的小波变换族包括 Harr_2,Daubechies_4 和 Daubechies_6。应用离散小波变换的一般步骤使用层次金字塔算法，它在每次循环中二分数据，产生很快的计算速度。

主成份分析方法将在本文中使用，详细描述见 2.2 节。

4 数值削减　数据用另外的更小的数据表示来替换或估计。数值削减可以分为参数方法和非参数方法。

参数方法使用模型来估计数据，以便只需要存储数据参数，而不是真实数据。Log-线性模型就是一个例子，它近似离散多维概率分布。

非参数方法包括直方图，聚类和采样。1）直方图使用装箱来近似数据分布。属性 A 的直方图将 A 的数据分布划分为不相交的子集或桶。这些桶在水平轴上显示，桶的高度反映桶所代表值的平均频率。通常，代表给定属性的连续区间。属性值划分的规则包括：等宽，等深，方差最优和最大差分。其中，方差最优和最大差分是最准确和实用的。前面所述的单一属性直方图可以扩展用于多个属性。多维直方图可以获取属性间的依赖性。2）聚类方法将在本文中使用，详细描述见 2.3 节。3）采样是用数据量小得多的随机采样或子集来表示大数据集。可用的采样方式包括：不带替换的简单随机采样，带替换的简单随机采样，聚类采样，分层采样。用采样做数据削减的好处在于，获得样本的成本与样本的大小成正比，而不是数据集的大小。因此，采样复杂度与数据集的大小成亚线性。而其它数据削减技术至少需要遍历一次数据集。对于固定大小的样本集，采样复杂度只随数据维数的增加而线性增加，而直方图则随数据维数指数性增加。采样是削减过的数据集进一步精练的自然选择。

5 离散化和概念层次产生　属性的原始数据值用区间或更高概念级来替换。

2.2 主成份分析[13]

假设要压缩的数据由 N 个 d 维元组或数据向量组成。主成份分析或 PCA（也称 Karhunen-Loeve 或 K-L 方法）搜索 c 个 d 维正交向量，它们能够用于最佳地表示数据，其中 $c \leq d$。这样原始数据就被投影到一个小得多的空间，从而产生数据压缩。

几何上，主成份分析可以看作是坐标轴的旋转，将原始坐标系的坐标轴旋转成一组新的正交坐标轴，并按照它们占原始数据方差的多少排列这些坐标轴。

为了要找到描述数据的一组数量较少的潜在变量，希望最初的几个坐标轴占原始数据中方差的大多数。

主成份分析是一种数据驱动的方法。它没有假定数据内部存在不同的类，因此被描述成一种非监督的特征提取方法。

基本过程如下：

1. 规范化输入数据，以便每个属性落入相同的区间。这一步可以保证大定义域的属性不会凌驾小定义域的属性。
2. 寻找第一个主成份轴，使数据空间中的所有数据点在该轴上的投影的方差最大。
3. 在与第一个主成份轴垂直的超平面上寻找第二个主成份轴，使数据空间中的所有数据点在该轴上的投影的方差最大。如图 2－2 所示。
4. 在与第一个和第二个主成份轴都垂直的超平面上寻找第三个主成份轴，使数据空间中的所有数据点在该轴上的投影的方差最大。
5. 如此持续进行下去，直到得到第 d 个主成份轴。
6. 选择前 c 个主成份轴，将原始数据空间中的所有数据映射到这 c 个坐标轴上，所得到 c 维坐标值就是数据的压缩表示。

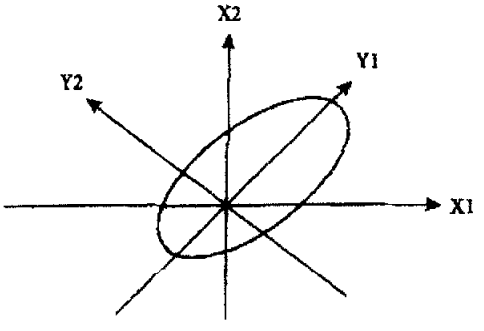


图 2－2 主成份分析示意图

椭圆代表数据集的分布，Y_1 和 Y_2 是前两个主成份轴。

按照上面所述步骤进行编程，在高维空间中搜索第一个主成份轴的过程就将拥有巨大的搜索空间，足以使程序无法运行下去。因此，上述方法只是原理上的说明，在实际操作中无法采用，必须使用 PCA 的优化算法，使搜索的过程变得简单且省时。

Hotelling 提出的寻找产生新变量的正交变换方法[14]，使新变量具有方差的极值。下面进行较详细的介绍。

设$x_1,\ldots,x_d$是原始数据坐标，ξ_i（i=1,...,d）是变换后的坐标，是原始变量的线性组合

$$\xi_i=\sum_{j=1}^{d}a_{ij}x_j \text{或} \xi=\mathbf{A}^T\mathbf{x} \tag{2.1}$$

其中：ξ和$\mathbf{x}$是随机向量，$\mathbf{A}$是系数矩阵。

考虑第一个变量ξ_1

$$\xi_1=\sum_{j=1}^{d}a_{1j}x_j$$

选择满足约束$\mathbf{a}_1^T\mathbf{a}_1=1$的$\mathbf{a}_1=(a_{11},a_{12},...,a_{1d})^T$使$\xi_1$方差最大。$\xi_1$的方差是

$$\begin{aligned}\text{var}(\xi_1)&=E[\xi_1^2]-E[\xi_1]^2=E[\mathbf{a}_1^T\mathbf{x}\mathbf{x}^T\mathbf{a}_1]-E[\mathbf{a}_1^T\mathbf{x}]E[\mathbf{x}^T\mathbf{a}_1]\\&=\mathbf{a}_1^T(E[\mathbf{x}\mathbf{x}^T]-E[\mathbf{x}]E[\mathbf{x}^T])\mathbf{a}_1=\mathbf{a}_1^T\boldsymbol{\Sigma}\mathbf{a}_1\end{aligned}$$

其中，$\boldsymbol{\Sigma}$是$\mathbf{x}$的协方差矩阵。寻找满足约束$\mathbf{a}_1^T\mathbf{a}_1=1$的$\mathbf{a}_1^T\boldsymbol{\Sigma}\mathbf{a}_1$的极值等价于寻找

$$f(\mathbf{a}_1)=\mathbf{a}_1^T\boldsymbol{\Sigma}\mathbf{a}_1-v\mathbf{a}_1^T\mathbf{a}_1$$

的无条件极值。其中v是拉各朗日乘子。对$\mathbf{a}_1$求导并令其等于0，得到

$$\boldsymbol{\Sigma}\mathbf{a}_1-v\mathbf{a}_1=0$$

显然，$\mathbf{a}_1$必须是$\boldsymbol{\Sigma}$的特征向量，v必须是$\boldsymbol{\Sigma}$的特征值，与$\mathbf{a}_1$对应。由于ξ_1的方差是

$$\text{var}(\xi_1)=\mathbf{a}_1^T\boldsymbol{\Sigma}\mathbf{a}_1=\mathbf{a}_1^Tv\mathbf{a}_1=v$$

我们希望该方差取最大值，因此选择$\boldsymbol{\Sigma}$最大的特征值作为v的值，$\mathbf{a}_1$取其对应的特征向量。这样，变量ξ_1就是第一个主成份，并对原始变量$x_1,\ldots,x_d$的任意线性函数都有最大方差。

获得第二个主成份$\xi_2=\mathbf{a}_2^T\mathbf{x}$的方法是：选择系数$\mathbf{a}_2$，使$\xi_2$在满足约束$\mathbf{a}_2^T\mathbf{a}_2=|\mathbf{a}_2|=1$且与$\mathbf{a}_1$垂直的条件下的方差$\mathbf{a}_2^T\boldsymbol{\Sigma}\mathbf{a}_2$取最大值。$\mathbf{a}_2$与$\mathbf{a}_1$垂直意味着

$$\mathbf{a}_2^T\mathbf{a}_1=0$$

再次使用拉各朗日待定乘子法，求$\mathbf{a}_2^T\mathbf{a}_2=1$，$\mathbf{a}_2^T\mathbf{a}_1=0$约束下，使$\mathbf{a}_2^T\boldsymbol{\Sigma}\mathbf{a}_2$取最大值的$\mathbf{a}_2$,即

$$\mathbf{a}_2^T\boldsymbol{\Sigma}\mathbf{a}_2-\mu\mathbf{a}_2^T\mathbf{a}_2-\eta\mathbf{a}_2^T\mathbf{a}_1$$

的无约束最大值。对$\mathbf{a}_2$求导并令其等于0，得到

$$2\boldsymbol{\Sigma}\mathbf{a}_2-2\mu\mathbf{a}_2-\eta\mathbf{a}_1=0 \tag{2.2}$$

两边都乘以$\mathbf{a}_1^T$，得到

$$2\mathbf{a}_1^T\boldsymbol{\Sigma}\mathbf{a}_2-\eta=0$$

因为$\mathbf{a}_2$与$\mathbf{a}_1$垂直，有$\mathbf{a}_1^T\boldsymbol{\Sigma}\mathbf{a}_2=0$，因此$\eta=0$。式(2.2)就变为

$$\boldsymbol{\Sigma}\mathbf{a}_2=\mu\mathbf{a}_2$$

这样，$\mathbf{a}_2$也是$\boldsymbol{\Sigma}$的特征向量。因为要寻找最大方差，$\mathbf{a}_2$必须是与余下的特征值

中的最大值相对应的特征向量。

继续推论下去，对于第 k 个主成份 $\xi_k = \mathbf{a}_k^T \mathbf{x}$，其 $\mathbf{a}_k$ 是与 Σ 的第 k 大的特征值对应的特征向量，ξ_k 的方差等于第 k 大特征值。

给出一个数据矩阵，为确定其主成份，不一定要形成样本协方差矩阵，可以使用奇异值分解。一个 $m \times n$ 阶矩阵 $\mathbf{Z}$ 可以写成以下形式：

$$\mathbf{Z} = \mathbf{USV}^T$$

设 $r = \min\{m,n\}$，$\mathbf{U}$ 是 $m \times r$ 阶矩阵，且 $\mathbf{U}^T\mathbf{U}$ 为单位矩阵。$\mathbf{V}$ 是 $r \times n$ 阶矩阵，且 $\mathbf{V}^T\mathbf{V}$ 为单位矩阵，其列向量 $\mathbf{v}_1,\ldots,\mathbf{v}_n$ 称为右奇异向量。$\mathbf{S} = diag(\sigma_1,\ldots,\sigma_r)$ 为奇异值 σ_i 的对角矩阵。$\mathbf{Z}$ 的奇异值是 $\mathbf{Z}^T\mathbf{Z}$ 的特征值的平方根。矩阵 $\mathbf{Z}$ 的右奇异向量是矩阵 $\mathbf{Z}^T\mathbf{Z}$ 的特征向量。

现在，我们要求协方差矩阵的特征向量，如果有一个矩阵 $\mathbf{Z}$ 使得 $\mathbf{Z}^T\mathbf{Z}$ 是协方差矩阵，那么我们就可以将 $\mathbf{Z}$ 奇异值分解，其右奇异向量就是协方差矩阵的特征向量，其奇异值就是协方差矩阵特征值的平方根。样本协方差矩阵可以写成如下形式：

$$\frac{1}{n-1}\sum_{i=1}^{n}(\mathbf{x}_i - \mathbf{m})(\mathbf{x}_i - \mathbf{m})^T = \frac{1}{n-1}\tilde{\mathbf{X}}^T\tilde{\mathbf{X}}$$

其中 $\tilde{\mathbf{X}} = \mathbf{X} - \mathbf{1m}^T$，$\mathbf{X}$ 是 $n \times d$ 阶的数据矩阵，$\mathbf{m}$ 是样本均值，$\mathbf{1}$ 是单位向量。因此可以设

$$\mathbf{Z} = \frac{1}{\sqrt{n-1}}\tilde{\mathbf{X}} \tag{2.3}$$

这样

$$\mathbf{Z}^T\mathbf{Z} = \frac{1}{n-1}\tilde{\mathbf{X}}^T\tilde{\mathbf{X}}$$

实际采用的主成份分析步骤是：

第一步，数据规范化；

第二步，用式(2.3)准备奇异值分解矩阵；

第三步，奇异值分解，得到的右奇异向量即为主成份变换系数向量，得到的奇异值的平方即为新变量的方差；

第四步，用式(2.1)求出坐标变换后得到的数据矩阵；

第五步，选择前 c 个变量，使得它们对应的方差之和占全部方差之和的比例达到满意的程度，或者将方差画成递减的谱线，寻找该谱线的拐点，保留拐点之前对应的变量。

本文中，数据削减采用主成份分析来降低数据量。对原数据中每一个样本的 684 维应用主成份分析，得到新变量的方差。然后画出方差谱线，如图 2－3

所示，寻找拐点（第 55 个变量）。由此可知，保留前 55 个新维，就可以保留足够多的信息。这样，数据就由原来的 684 维压缩为 55 维的特征。

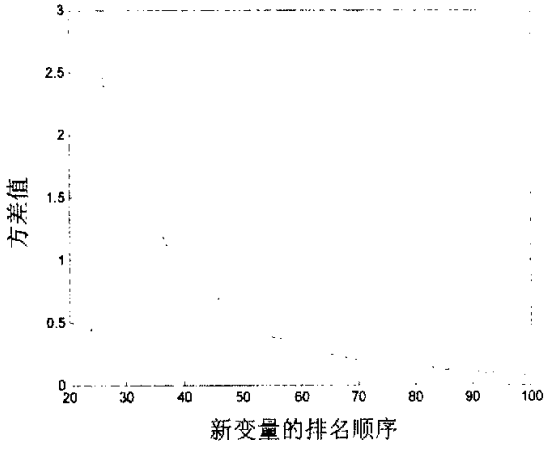


图 2－3 沙尘暴主成份的方差谱线

2.3 聚类方法的改进

聚类用于数据削减的基本思想是，通过聚类找出沙尘暴的几个典型模式，然后用数据样本到这些典型模式的距离作为原样本削减后的表示。但是，由于沙尘暴原始数据的维数为 171×4＝684 维（参见 1.3 节），如此高的维数对任何聚类方法来说都是非常沉重的负担，几乎无法实现。因此，必须找到一种间接的方法，使聚类能够进行。

在已有的工作基础中所采用的间接方法是：将分属于 4 个场的 684 维数据按照不同的场进行分拆（分拆后每个场包含 171 维数据），然后进行分场聚类，最后，将分场聚类结果再经过场间组合生成样本的聚类。

这一方法有效地解决了高维聚类问题，但仍然存在一些问题。主要的问题有：

（一） 分场聚出的类的场间组合不一定就是样本的聚类。上述方法对组合结果进行挑选，去除含样本少的组合，就是这个原因。

（二） 样本的聚类可能不能通过分场聚类的组合得到反映。分场聚类的组合具有的性质之一为，其中所有的样本在每个场的聚类中，都会被聚为一类。而样本的聚类不一定会满足这一点。也就是说，如果样本的一个聚类无法满足上述条件，即无法在所有场的聚类中聚为一类，它就无法表示成分场聚类组合的形式。当我们用分场聚类组合来获得样本的聚类时，样本的这个聚类就被漏

掉了。

（三） 由于使用了组合，即使每个场只聚出数量不多的几个类，经过组合，得到的样本类的数量也会成指数性增长。正因为这一点，原有的工作基础中对每个场只聚出 2 类或 3 类，无法反映更详细的分布信息。

由于原有方法存在的上述缺陷，促使我们对聚类方法做进一步的研究和改进。改进思想如下：

- 沿用了分场聚类的思路进行样本聚类；
- 计算各场的典型模式；
- 求样本中的各场数据与对应场的典型模式间的距离，作为原数据的压缩表示。

这一改进思想是受了文献[15][30]所述的综合预报模型的启发。它给我们的启发是：可以将各场的数据看成是不同的信息源，先用各场的数据分别进行预报，再对各场的预报结果进行综合预报。这样，综合预报就起到了综合各场提供的信息并作出取舍的作用。因为各场分别预报的输入只是本场信息，所以数据削减时采用分场聚类进行压缩后就不再需要组合。

这一方法能够克服原有方法的上述三个缺陷。这三个缺陷的根源是组合操作。改进方法采用综合预报的方式避免了组合的发生，使每个场可以聚出更多的类。

改进方法的具体执行步骤如下：

- 高度场（H5）,位温场（SE）和风场（UV）都进行距平运算，得到高度距平场（H5_dm），位温距平场（SE_dm）和风距平场(UV_dm)。将新得到的三个场和 H5 一起作为新的样本集。
- 在这个样本集中取出所有沙尘暴日的样本，构成新的样本数据 H5_sa、H5_dm_sa、UV_dm_sa、SE_dm_sa。
- 利用自组映射神经网络分别对 H5_sa、H5_dm_sa、UV_dm_sa、SE_dm_sa 进行聚类，每场聚出 5 类。
- 对每一类，求其类中心。设第 l 个目标场第 k 类所有样本为 $S_k^{(l)}{}_1$、$S_k^{(l)}{}_2$，…，$S_k^{(l)}{}_n$，该类中心为

$$M_k^{(l)} = \frac{1}{n}\sum_{j=1}^{n} S_k^{(l)}{}_j \qquad (l = 1,2,3,4)$$

- 分别计算第一步所得到的数据样本与聚类中心的距离，以获得单场特征。设 $a_{i,j}^{(l)}$ 是样本第 l 个场中的一个元素，$f_k^{(l)}$ 是第 l 个场的第 k 个特征，则有

$$f_k^{(l)} = \sqrt{\sum_{i,j}(a_{i,j}^{(l)} - M_k^{(l)})^2} \qquad (l = 1,2,3,4)$$

- 将来自同一场的5个特征作为BP神经网络的输入，输出为单场预报值。将此预报值与5个特征一起作为综合预报的输入。将4个场的输入放到一起，得到共24个综合预报输入。

整个系统的处理流程如图2—4所示。

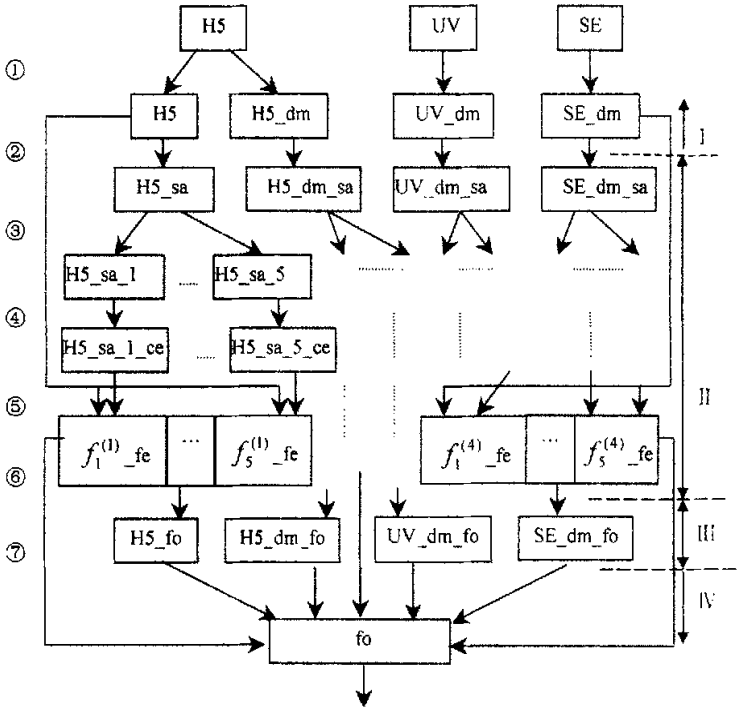


图2—4 ANN综合预报模型示意图

其中，sa代表沙尘暴日，ce代表聚类中心，fe代表特征，fo代表预报结果，第Ⅰ阶段是样本预处理，第Ⅱ阶段是单场特征提取，第Ⅲ阶段是单场预报，第Ⅳ阶段是综合预报，①是距平，②是沙尘暴日选择，③是聚类，④是聚类中心计算，⑤距离计算，⑥是单场预报，⑦是综合预报

由此可知，每个场聚为5类，与其对应的有5个距离。用这5个距离作为特征进行单场预报，将预报结果与这5个特征一起作为综合预报的输入，所以，每个场为综合预报提供6个输入，4个场所提供的总输入为$6\times4=24$个。这样，数据维数就由原来的684维压缩为24维，少于原有方法所得的40维。

2.4 小结

本文在数据预处理阶段使用了两种数据削减的方法，即主成份分析和对原有聚类方法的改进。这两种方法均是通过 Matlab 编程实现的。为了比较这两种方法和原有方法的优劣，我们采用相同的分类方法——BP 神经网络，相同的训练集，相同的测试集。因此，影响系统结果好坏的因素就只在于它们的预处理算法。衡量结果好坏的指标采用 CSI，其定义见 1.3.2 节。

实验结果如表 2－1 所示。

表 2－1 三种数据削减方法效果的比较

数据削减方法	聚类	主成份分析	聚类改进
CSI	0.22	0.26	0.28
运行时间(s)	－－	102.359	175.11

注：原有聚类方法的典型模式选择为手动进行，故无法计量运行时间

由于所用的 BP 神经网络是全连接，没有经过泛化，所以网络结果不稳定。也就是说，对于同一预处理方法，如果进行多次实验，各次所得的 CSI 值差异很大。表中所给出的 CSI 值是多次实验的平均值。

从表中可以看出，本文所使用的两种方法都比原有聚类方法的效果好。其中改进后的聚类方法效果最好。主成份分析方法虽然效果较差，但它的运算步骤简单，运行所需时间较少,因此适用于对快速性要求较高的场合。在后面分类方法的研究中，需要统一的预处理方法，以比较各种分类方法的好坏。所以，我们采用这里证实的最好的预处理方法，即改进后的聚类。

第三章 分类器的设计与实现

预报模型的主体是对提取出的特征进行预报的分类器，因此，预报模型的设计与实现的主要工作是围绕分类器的设计与实现进行的。

3.1 主要分类方法[16]

分类是找到描述和区分数据类或概念的一组模型或函数的过程，目的是为了能够使用模型来预测未知对象的类。数据分类一般分两步完成[8]，即：

第一步，创建一个模型来描述一组预定义的数据类或概念。其方法是：

- 确定训练样本集；
- 通过对训练样本的有监督的学习（即模型的学习是受到监督的，因为已经告知了每个训练样本属于哪一类）或无监督的学习（每个训练样本的类标签是未知的，要学习的一组类或类数可能事先也不知道），建立模型。

第二步，利用模型进行分类预测。其方法是：

- 使用带类标签样本的一个测试集估计模型的预测准确性。
- 如果认为模型的准确性是可以接受的，就可以用这个模型来分类未来数据元组或对象，这些未来对象的类标签是未知的。

机器学习，专家系统，统计学和神经生物学的研究者提出了许多分类方法，这些方法对模型信息已知程度的要求是不同的，有些模型要求大量的模型信息（比如概率密度，分布类型，类别标记等），有些模型则只要求有类标签。下面就按照这种要求从多到少的顺序介绍各种分类方法。

3.1.1 贝叶斯决策理论

该理论适用于研究模式类的概率结构完全知道的理想情况。虽然这种情况可能很少出现在实际中，但是它为我们提供了能与其他分类器作对比的一个评价依据，即“最优(贝叶斯)分类器”。

1 基本思想

贝叶斯决策论的基本思想非常简单。为最小化总体风险，总是选择那些能够最小化条件风险 $P(\alpha \mid \mathbf{x})$ 的行为。尤其是，为了最小化分类问题中的误差概率，总是选择那些使后验概率 $P(\omega_j \mid \mathbf{x})$ 最大的类别。贝叶斯公式

$$P(\omega_j \mid \mathbf{x}) = \frac{p(\mathbf{x} \mid \omega_j)P(\omega_j)}{\sum_i p(\mathbf{x} \mid \omega_i)P(\omega_i)}$$

允许我们通过先验概率 $P(\omega_j)$ 和条件密度 $p(\mathbf{x}|\omega_j)$ 来计算后验概率。如果对在模式 ω_i 中所做的误分的惩罚与模式 ω_j 的不同，那么在做出判决行为之前必须先根据该惩罚函数对后验概率加权。

2 贝叶斯置信网

贝叶斯置信网是贝叶斯决策理论的扩展，也称因果网，它以图形的形式来表示特征分量间的因果依赖性，如图 3－1 所示。它采用了有向无环图的拓扑形式，每一个结点都具有方向性，且没有循环结点。当变量的任意子集被箝位为某些已知值的时候，通过贝叶斯推理计算，每一个结点都可以获得一个概率值。例如结点 A 处于状态 a 时的概率为 $P(a)$。

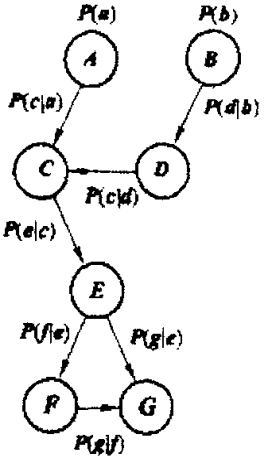


图 3－1 贝叶斯置信网

3.1.2 最大似然和贝叶斯参数估计

这两种方法适用于模式类的概率结构未知，但一般的分布类型已知的情况。由于若干参数值未知，造成概率分布的不确定。因此，为获得最好的分类效果，必须估计出正确的参数值。

参数估计问题是统计学中的经典问题，并且已经有了一些具体的解决方法。最大似然估计和贝叶斯估计是两种很有效的常用方法。虽然这两个方法得到的结果通常是很接近的，但这两个方法的本质却有很大差别。

1 最大似然估计　把待估计的参数看作是确定性的量，只是取值未知。其最佳估计就是使得产生已观测到的样本(即训练样本)的概率为最大的那个值。

2 贝叶斯估计　该方法与最大似然估计不同的是，把待估计的参数看成是符合某种先验概率分布的随机变量。在对样本进行观测的过程中，把先验概率

密度转化为后验概率密度，这样就利用样本的信息修正了对参数的初始估计值。在贝叶斯估计中，一个典型的效果就是，每得到新的观测样本，都使得后验概率密度函数变得更加尖锐，使其在待估参数的真实值附近形成最大的尖峰。这个现象就称为“贝叶斯学习”过程。

3.1.3 非参数方法

非参数方法是针对没有参数化的先验分布形式的任何知识进行处理的。分类器必须基本上只利用输入训练样本自身提供的信息来工作。在模式识别领域中，非参数估计方法有两种最基本的途径。第一种途径：估计概率密度函数，并将它用于后面的分类中。典型方法有Parzen窗方法和它的一种硬件实现方式——概率神经网络(PNN)。第二种途径：不估计具体的概率密度函数，直接根据样本进行分类。代表方法为k-最近邻方法和松弛网络。

1 Parzen窗方法　通过计算落入不同宽度的窗中的训练样本数目，估计特征的概率密度函数。

2 k-最近邻方法　将在本文中使用，详细描述见 3.4 节。

3 模糊分类方法　通过使用对“类别隶属度函数”的启发式选择和启发式的合取规则得到分类函数。然而这种技术的适用范围仅仅局限于当训练样本过少，或者特征数量比较少，或者设计者的知识是从先验置信得出的场合。

4 松弛方法　建立包围在原型样本点周围的“吸引盆”（半径为可调整参数的d 维输入特征空间中的超球体）。如果一个测试样本点位于一个吸引盆中，它的类别就被归于这个盆所属的类别。RCE网络是其中的一种方法，这个算法通过调整吸引盆尺寸，使其尽可能多地包含周围同一类别的训练样本点。

3.1.4 判别函数法

判别函数法是研究参数估计的一般方法。在此假定所谓的“判别函数”具有一种十分特殊的形式——线性形式。利用判别函数参数值的增量学习规则，寻找线性判别函数的问题可被形式化为极小化准则函数的问题。以分类为目的的准则函数可以是样本风险，或者是训练误差，即对训练样本集进行分类所引起的平均损失。

1 感知器算法　通过调整参数来提高与ω_1的样本的内积，而降低与ω_2的样本的内积，以使准则函数极小化。

2 梯度下降法　两种著名方法分别是：Kiefer-Wolfowitz算法，它是对回归函数的极小化；Robbins-Monro算法，它是寻找回归函数的根。

3 支持向量机　详细描述参见 3.5 节。

3.1.5 多层神经网络

详细描述参见 3.3 节。

3.1.6 随机搜索技术

当一个模式识别问题涉及离散模型，或者有过高的复杂度，常规的解析方法或梯度下降算法都无能为力时，那么可以尝试采用随机搜索技术，即在某个层次上运用随机性去搜索模型参数。具体方法有：

1 模拟退火　来源于物理学中的金属退火处理，由下述过程构成：随机扰动系统，同时逐渐降低系统的随机程度，直到最终得到一个最优解。

2 Boltzmann算法　通过训练网络的互连权，使得最终得到正确输出的概率提高。这种算法一方面基于模拟退火，另一方面又运用了Kullback-Liebler散度的梯度下降过程。

3 遗传算法和遗传规划　基于进化的搜索方法，它能够在设计者指定的空间中进行高度并行化的随机搜索。遗传算法中的基本表达方式是二进制位串，或称为染色体，而在遗传规划中则采用的是计算机代码片断。种群的差异性是通过复制，交叉和变异等遗传算子来实现的。

3.1.7 非度量方法

这种方法不再基于统计模型，转而研究利用逻辑规则来表达非度量数据的分类问题。非度量数据由语义属性的列表构成。这种列表可以是有序（如串）的，也可以是无序的，具体实现方法：

1 CART，ID3 和C4.5　这些方法均是基于树的方法，根据回答一系列问题(经常是二值的)来进行分类。设计者选择问题的形式，并从根结点开始，将结点分支，使其表达的“纯度”增加，进而生长起一棵树。业已发展了多种可选用的不纯度函数指标，其中熵不纯度的应用范围最广。为避免过拟合现象的出现，可采用分枝停止技术(声明一个结点称为叶节点，不纯度接近零)，或者对树进行剪枝处理以获得不纯度最小化的叶节点。

2 基于文法的方法　假定串是由某种特定规则产生的。这类规则可以用文法表达。一个文法G由一个字母表、中间符号、一个初始符号以及最为关键的重写规则集组成。4 种不同类型的文法对符号转换的性质作了不同的假设。“分析”的作用是输入一个串x，判断它是否属于由文法G产生的语言，如果是，则导出它。

3 基于规则的系统　采用命题逻辑或者一阶谓词逻辑来表达模式类别。广

义地说，规则可以通过连续运用非常复杂的复合规则来“序贯覆盖”训练样本集的方法来学习。

3.1.8 基于实例的推理

存储训练样本为复杂的符号描述。当给出要分类的新实例时，基于实例的推理器首先在历史实例中查找是否存在一个相同的训练样本。如果存在，则返回这一实例所伴随的解决方案；如果不存在相同的实例，则基于实例的推理器首先搜索具有与新实例相似成份的训练实例，然后组合相邻训练实例的解决方案，以便提出新实例的解决方案，基于实例推理的结构框图如图 3－2 所示。

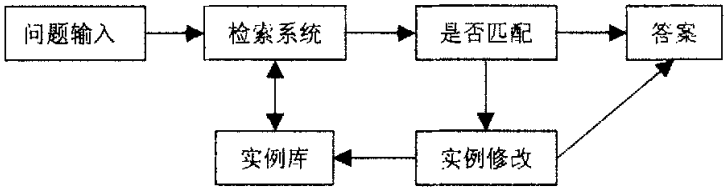


图 3－2 基于实例推理系统的基本结构

3.1.9 粗糙集理论

粗糙集理论可用于分类，以发现不精确或噪声数据中的结构性关系。它应用于离散值属性。粗糙集理论基于给定训练数据中等价类的建立。组成等价类的所有样本对于描述数据的属性是等价的。给定类的粗糙集的定义由两个集来近似：低近似集和高近似集。低近似集由绝对属于这一类的数据样本组成，高近似集由不可描述为不属于这一类的数据样本组成。

3.1.10 其它方法[8]

1 ARCS或关联规则聚类系统　在训练集上挖掘关联规则，使其将定量属性的区间与类别属性的类标签相关联。然后对生成的规则进行聚类，同一聚类中的定量属性相邻区间可以合并。

2 关联分类　找出既频繁又准确的所有可能规则，使用启发式方法来创建分类器，其中发现的规则按照信任度和支持度的降序来组织。当分类新样本时，使用满足样本的第一条规则来分类它。

3 CAEP（聚集浮现模式分类）[31]　为每个类寻找满足给定支持度和成长率阈值的浮现模式。当分类新样本时，对于每个类，聚集新样本中出现的该类的浮现模式的区分能力，得到该类的规范化分数

$$norm\_score(t,i)=\frac{1}{base\_score(i)}\sum_{e\subseteq t,e\in E_i}\frac{GR(e)}{GR(e)+1}*supp_{D_i}(e)$$

其中，t是一个新样本，i是第i类，$base_score(i)$是第i类的基分数，E_i是第i类中所有浮现模式所组成的集合，e是E_i中的一个浮现模式，$GR(e)$是e的成长率，D_i是所有属于第i类的样本组成的数据集，$supp_{D_i}(e)$是e在D_i中的支持度。具有最大规范化分数的类决定新样本的类标签。

3.2 基于时间序列的预报分析[17]

时间序列是一个变量在相等时间间隔上的一组有序的取值。对时间序列的分析有两个主要的目标：(a) 识别观测序列所代表的现象的本质。(b) 预测时间序列变量的未来值。这两个目标都需要识别和描述观测时间序列模式。一旦建立了模式，就可以解释并将它与其它数据集成。不管对现象的物理解释理解的深度，都可以用已识别的模式推断预测未来事件。

时间序列模型的拟合是一项非常复杂的工作。有很多模型拟合的方法，包括：指数平滑，Box-Jenkins ARIMA 模型和人工神经网络（ANN）。下面一一介绍这些方法。

3.2.1 指数平滑

指数平滑是早期提出的一种很简单的时间序列模型，由 C.C. Holt 于 1957 年提出，Winters 于 1965 年泛化了这一方法以包含季节性。因此，这一方法也被称为 Holt-Winters 方法。Holt-Winters 方法有 3 个更新方程，这些方程用来将更大的权值赋予新近的观测值，将更小的权值赋予久远的观测值。这些权值按固定比率呈几何衰减。

3.2.2 ARIMA 模型

Box 和 Jenkins 在 1976 年提出了 ARIMA 方法。这一方法组合了自回归和移动平均两种方法。这使得它在许多领域获得了巨大的流行性，研究实践证实了它的实力和灵活性。但是正因如此，ARIMA 是一种复杂的技术，不容易使用，其结果依赖于研究者的专业等级。建造 Box-Jenkins 时间序列模型有三个主要阶段：模型识别，估计和验证。ARIMA 和指数平滑都可用作单元时间序列模型，即由相等时间增量上序列记录的标量观测值所组成的时间序列。但是 ARIMA 还可以用作多元时间序列模型，这时它又称为向量 ARIMA 模型。

3.2.3 人工神经网络（ANN）[18]

ANN 的架构是以生物神经系统的当前理解为基础的，试图通过简单计算单元的稠密互联达到良好的性能。ANN 提供了几个可贵的特征:

- 它从数据中推断出解，不用数据中规律性的先验知识。网络直接从实例中学习模式间的相似性。
- ANN 可从先前的例子泛化到新的例子。ANN 也很善于从包含无关数据的输入中抽象出本质特性。
- 它是非线性的，即于线性技术相比，它能更准确地解决一些复杂问题。
- ANN 是高度并行的，与其它方法相比，它能更快的执行。

时间序列分析方法用于统计天气预报时，单元时间序列方法的局限性在于，它所结合的唯一信息是这一变量过去的取值提供的，而用于预报的多个因子的观测值无法派上用场。因此，指数平滑方法就无法用于统计天气预报。

大部分时间序列建模过程陷入多元 ARIMA 模型的框架。为了最优地拟合 ARIMA 类型的模型到一组时间序列，数据必须是平稳的，且服从正态分布。当开发 ANN 模型时，不需要知道数据的统计分布，ANN 的内部结构也会隐式地考虑数据中的不稳定性。不像 ARIMA 类型的模型，ANN 因泛化对噪声数据相对不敏感。文献[19]显示，相对于线性 ARMAX（带外生输入的自回归移动平均）时间序列方法，非线性 ANN 模型对密西西比州 Collins 附近中尺度叶河盆的 R-R 关系提供了更好的表示。

由此可以看出，人工神经网络在用于统计天气预报的时间序列分析方面，性能优于多元 ARIMA。由于目前尚未发现用 ANN 时间序列分析方法进行沙尘暴预报，因此，本文采用 ANN 方法对沙尘暴进行时间序列预报分析。

3.3 BP 神经网络的改进[20]

神经网络由并行运算的简单元素组成。这些元素是由生物神经系统所启发的。与自然中一样，网络函数主要是由元素间的连接决定的。我们可以通过调整元素间的连接（权）的值，训练一个神经网络执行一个特定的函数。这一过程如图 3－3 所示。其中，网络的调整是基于输出与目标的比较，直到网络输出匹配目标。通常使用许多这样的输出/目标对来训练一个网络。

反向传播（BP）是最小均方（LMS）算法的泛化，可用于训练多层神经网络。与LMS学习法则一样，反向传播也是一个近似最陡下降算法。其中的性能指标是均方差。LMS和反向传播的唯一区别是导数的计算方式。LMS用于单层线性网络中，误差是网络权值的显式线性函数，其对权值的导数很容易计算出

来。而在具有非线性传递函数的多层网络中，网络权值和误差间的关系更为复杂，需要LMS的泛化BP来解决。BP算法是通过算子链规则来计算该导数的。

反向传播算法的过程总结如下：

第一步，传播输入向前通过网络：

$$\mathbf{a}^0 = \mathbf{p}$$

$$\mathbf{a}^{m+1} = \mathbf{f}^{m+1}(\mathbf{W}^{m+1}\mathbf{a}^m + \mathbf{b}^{m+1}) \qquad m = 0,1,...,M-1$$

$$\mathbf{a} = \mathbf{a}^M$$

其中$\mathbf{a}^i$是第i层的输出向量，$\mathbf{a}$为网络的输出向量，$\mathbf{p}$是网络的输入向量，M是网络的层数，$\mathbf{f}^i$是第i层的传递函数，$\mathbf{W}^i$是第i层的权值矩阵，$\mathbf{b}^i$是第i层的偏置向量。

第二步，传播敏感性反向通过网络：

$$\mathbf{s}^M = -2\dot{\mathbf{F}}^M(\mathbf{n}^M)(\mathbf{t}-\mathbf{a})$$

$$\mathbf{s}^m = \dot{\mathbf{F}}^m(\mathbf{n}^m)(\mathbf{W}^{m+1})^T\mathbf{s}^{m+1} \qquad m = M-1,...,2,1$$

其中，$\mathbf{s}^i$是第i层的敏感性向量，$\dot{\mathbf{F}}^i$是第i层传递函数的导数矩阵，$\mathbf{n}^i$是第i层的净输入向量，$\mathbf{t}$是网络的目标向量。

最后，使用近似最陡下降准则更新权值和偏置：

$$\mathbf{W}^m(k+1) = \mathbf{W}^m(k) - \alpha\mathbf{s}^m(\mathbf{a}^{m-1})^T$$

$$\mathbf{b}^m(k+1) = \mathbf{b}^m(k) - \alpha\mathbf{s}^m$$

其中，k表示第k步的值，α是学习率。

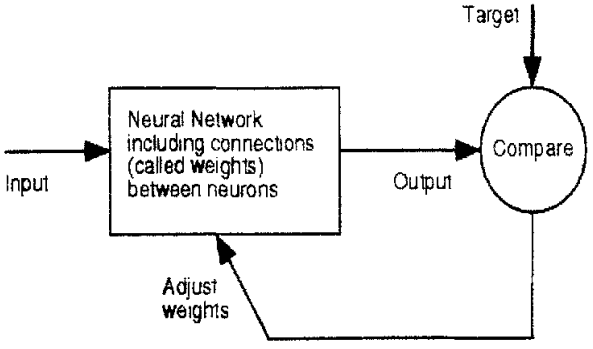


图3－3 神经网络权值的调整过程

反向传播算法是神经网络研究的主要突破。但是，基本算法对于大部分实际应用都太慢了。多层网络的性能面可能有许多局部最小值点，曲率在参数空间的不同区域可能有很大变化。一些反向传播的变种提供了明显的加速，并使

算法更实用。

动量方法是基于这样的观察：如果可以平滑掉轨迹上的振荡，收敛可能得到改善。动量方法使用一个低通滤波器做到这一点。可变学习率方法通过在平缓表面上增加学习率，在斜度增加时减小学习率，可以加速收敛。共轭梯度法是一种数值优化方法，是最陡下降法和Newton法之间的折中算法。

本文所采用的反向传播的改进算法是Levenberg-Marquardt算法。它是Newton方法的变种。Newton方法设计用来最小化作为其它非线性函数平方和的函数。这一点非常适合于神经网络训练，因为性能指标就是均方误差。

优化性能指标$F(\mathbf{x})$的Newton方法是

$$\mathbf{x}_{k+1}=\mathbf{x}_k-\mathbf{A}_k^{-1}\mathbf{g}_k \tag{3.1}$$

其中，$\mathbf{x}$是权值和偏置向量，$\mathbf{A}_k\equiv\nabla^2F(\mathbf{x})|_{\mathbf{x}=\mathbf{x}_k}$，$\mathbf{g}_k\equiv\nabla F(\mathbf{x})|_{\mathbf{x}=\mathbf{x}_k}$

设$F(\mathbf{x})$是误差函数的平方和，即

$$F(\mathbf{x})=\sum_{i=1}^{N}v_i^2(\mathbf{x})=\mathbf{v}^T(\mathbf{x})\mathbf{v}(\mathbf{x})$$

其中，$\mathbf{v}$是误差向量函数。那么梯度的第j个元素将是

$$[\nabla F(\mathbf{x})]_j=\frac{\partial F(\mathbf{x})}{\partial x_j}=2\sum_{i=1}^{N}v_i(\mathbf{x})\frac{\partial v_i(\mathbf{x})}{\partial x_j}$$

所以梯度可以写成矩阵形式

$$\nabla F(\mathbf{x})=2\mathbf{J}^T(\mathbf{x})\mathbf{v}(\mathbf{x}) \tag{3.2}$$

其中

$$\mathbf{J}(\mathbf{x})=\begin{bmatrix}\frac{\partial v_1(\mathbf{x})}{\partial x_1} & \frac{\partial v_1(\mathbf{x})}{\partial x_2} & \cdots & \frac{\partial v_1(\mathbf{x})}{\partial x_n}\\ \frac{\partial v_2(\mathbf{x})}{\partial x_1} & \frac{\partial v_2(\mathbf{x})}{\partial x_2} & \cdots & \frac{\partial v_2(\mathbf{x})}{\partial x_n}\\ \cdots & \cdots & \cdots & \cdots\\ \frac{\partial v_N(\mathbf{x})}{\partial x_1} & \frac{\partial v_N(\mathbf{x})}{\partial x_2} & \cdots & \frac{\partial v_N(\mathbf{x})}{\partial x_n}\end{bmatrix} \tag{3.3}$$

是Jacobian矩阵。接着找出Hessian矩阵。$\nabla^2F(\mathbf{x})$矩阵的第k,j个元素将是

$$[\nabla^2F(\mathbf{x})]_{k,j}=\frac{\partial^2F(\mathbf{x})}{\partial x_k\partial x_j}=2\sum_{i=1}^{N}\left\{\frac{\partial v_i(\mathbf{x})}{\partial x_k}\frac{\partial v_i(\mathbf{x})}{\partial x_j}+v_i(\mathbf{x})\frac{\partial^2v_i(\mathbf{x})}{\partial x_k\partial x_j}\right\}$$

$\nabla^2F(\mathbf{x})$矩阵可表达为矩阵形式

$$\nabla^2 F(\mathbf{x}) = 2\mathbf{J}^T(\mathbf{x})\mathbf{J}(\mathbf{x}) + 2\mathbf{S}(\mathbf{x})$$

其中

$$\mathbf{S}(\mathbf{x}) = \sum_{i=1}^{N} v_i(\mathbf{x})\nabla^2 v_i(\mathbf{x})$$

如果假设$\mathbf{S}(\mathbf{x})$很小（因为它是$\mathbf{x}$的高阶无穷小），可以得到$\nabla^2 F(\mathbf{x})$的近似为

$$\nabla^2 F(\mathbf{x}) \cong 2\mathbf{J}^T(\mathbf{x})\mathbf{J}(\mathbf{x}) \tag{3.4}$$

将式（3.4）和式（3.2）代入到式（3.1），就获得了Gauss-Newton法[32]：

$$\mathbf{x}_{k+1} = \mathbf{x}_k - [2\mathbf{J}^T(\mathbf{x}_k)\mathbf{J}(\mathbf{x}_k)]^{-1} 2\mathbf{J}^T(\mathbf{x}_k)\mathbf{v}(\mathbf{x}_k)$$

$$= \mathbf{x}_k - [\mathbf{J}^T(\mathbf{x}_k)\mathbf{J}(\mathbf{x}_k)]^{-1}\mathbf{J}^T(\mathbf{x}_k)\mathbf{v}(\mathbf{x}_k)$$

注意到Gauss-Newton法的好处是不需要再计算二阶导数。但它有一个问题是Hessian矩阵$\mathbf{H} = \mathbf{J}^T\mathbf{J}$可能是不可逆的。通过使用下面的修改来近似Hessian矩阵，使这一问题得到解决

$$\mathbf{G} = \mathbf{H} + \mu\mathbf{I}$$

设$\mathbf{H}$的特征值和特征向量分别为$\{\lambda_1, \lambda_2, ..., \lambda_n\}$和$\{z_1, z_2, ..., z_n\}$，则

$$\mathbf{G}\mathbf{z}_i = [\mathbf{H} + \mu\mathbf{I}]\mathbf{z}_i = \mathbf{H}\mathbf{z}_i + \mu\mathbf{z}_i = \lambda_i\mathbf{z}_i + \mu\mathbf{z}_i = (\lambda_i + \mu)\mathbf{z}_i$$

因此$\mathbf{G}$的特征向量与$\mathbf{H}$相同，$\mathbf{G}$的特征值是$(\lambda_i + \mu)$。通过增大μ直到对所有的i有$(\lambda_i + \mu) > 0$，可以使$\mathbf{G}$变为正定，因此矩阵将是可逆的。这导出了Levenberg-Marquardt算法：

$$\mathbf{x}_{k+1} = \mathbf{x}_k - [\mathbf{J}^T(\mathbf{x}_k)\mathbf{J}(\mathbf{x}_k) + \mu_k\mathbf{I}]^{-1}\mathbf{J}^T(\mathbf{x}_k)\mathbf{v}(\mathbf{x}_k) \tag{3.5}$$

这个算法有一个很有用的特征：当μ_k增加时，它接近小学习率的最陡下降算法

$$\mathbf{x}_{k+1} \cong \mathbf{x}_k - \frac{1}{\mu_k}\mathbf{J}^T(\mathbf{x}_k)\mathbf{v}(\mathbf{x}_k) = \mathbf{x}_k - \frac{1}{2\mu_k}\nabla F(\mathbf{x})$$，对于大μ_k

而当μ_k减少到零时，算法成为Gauss-Newton法。

算法开始时设置μ_k为某一小值。如果不能得到$F(\mathbf{x})$的更小值，μ_k就乘以某一因子$\vartheta > 1$。最终$F(\mathbf{x})$应减小，因为μ_k增大使方法更接近最陡下降。如果产生了$F(\mathbf{x})$的更小值，那么μ_k就除以ϑ，以便算法接近Gauss-Newton法，能够提供更快的收敛。这一算法提供了Newton法的速度和最陡下降法的保证收敛之间的良好折中。

Levenberg-Marquardt算法的关键一步是Jacobian矩阵的计算。误差向量为

$$\mathbf{v}^T = \begin{bmatrix} e_{1,1} & e_{2,1} & \cdots & e_{S^M,1} & e_{1,2} & \cdots & e_{S^M,Q} \end{bmatrix}$$

其中，Q为训练集中的样本个数，S^i为第i层的神经元个数。参数向量为

$$\mathbf{x}^T = \left[w_{1,1}^1 \quad w_{1,2}^1 \quad \cdots \quad w_{S^1,R}^1 \quad b_1^1 \quad \cdots \quad b_{S^1}^1 \quad w_{1,1}^2 \quad \cdots \quad b_{S^M}^M\right]$$

其中，R为输入向量的维数。因此，如果将这些代入式（3.3），多层神经网络的的Jacobian矩阵可以写成

$$\mathbf{J}(\mathbf{x}) = \begin{bmatrix} \frac{\partial e_{1,1}}{\partial w_{1,1}^1} & \frac{\partial e_{1,1}}{\partial w_{1,2}^1} & \cdots & \frac{\partial e_{1,1}}{\partial w_{S^1,R}^1} & \frac{\partial e_{1,1}}{\partial b_1^1} & \cdots \\ \frac{\partial e_{2,1}}{\partial w_{1,1}^1} & \frac{\partial e_{2,1}}{\partial w_{1,2}^1} & \cdots & \frac{\partial e_{2,1}}{\partial w_{S^1,R}^1} & \frac{\partial e_{2,1}}{\partial b_1^1} & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \frac{\partial e_{S^M,1}}{\partial w_{1,1}^1} & \frac{\partial e_{S^M,1}}{\partial w_{1,2}^1} & \cdots & \frac{\partial e_{S^M,1}}{\partial w_{S^1,R}^1} & \frac{\partial e_{S^M,1}}{\partial b_1^1} & \cdots \\ \frac{\partial e_{S^M,2}}{\partial w_{1,1}^1} & \frac{\partial e_{S^M,2}}{\partial w_{1,2}^1} & \cdots & \frac{\partial e_{S^M,2}}{\partial w_{S^1,R}^1} & \frac{\partial e_{S^M,2}}{\partial b_1^1} & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \end{bmatrix}$$

计算Jacobian的元素，如果x_l是权值

$$[\mathbf{J}]_{h,l} = \tilde{s}_{i,h}^m \times a_{j,q}^{m-1} \tag{3.6}$$

如果x_l是偏置

$$[\mathbf{J}]_{k,l} = \tilde{s}_{i,h}^m \tag{3.7}$$

其中，$\tilde{s}$是Marquardt敏感性

$$\tilde{\mathbf{S}}_q^M = -\dot{\mathbf{F}}^M(\mathbf{n}_q^M) \tag{3.8}$$

$$\tilde{\mathbf{S}}_q^m = \dot{\mathbf{F}}(\mathbf{n}_q^m)(\mathbf{W}^{m+1})^T \tilde{\mathbf{S}}_q^{m+1} \tag{3.9}$$

Levenberg-Marquardt反向传播算法可总结如下：

1．向网络提供所有输入，计算相应网络输出和误差。计算所有输入误差的平方和$F(\mathbf{x})$；

2．在初始化式（3.8）后，用循环关系式（3.9）计算Marquardt敏感性。用式（3.6）和（3.7）计算Jacobian矩阵的元素；

3．解式（3.5），以获得$\mathbf{x}_{k+1}$；

4．重新计算在$\mathbf{x}_{k+1}$下的误差平方和。如果这个新平方和比第一步中的更小，则将μ除以ϑ，并转到第一步。如果平方和没有减小，则将μ乘以ϑ，跳到第三步，重新计算$\mathbf{x}_{k+1}$。

这一过程的停止条件可以设为$F(\mathbf{x})$的值降到某一阈值以下，或者循环达到

给定的循环次数。

本文中，LM算法采用Matlab神经网络工具箱实现。数据预处理采用 2.3 节所述的聚类方法的改进，其中数据由 684 维压缩到 24 维。沙尘暴预报中，采用了 3.2 节所述的时间序列分析预报方法，即增加了当天发生沙尘暴的站点数作为分类器的输入。因此，神经网络的输入层有 25 个神经元。输出层有一个神经元，其输出决定预报第二天有无沙尘暴。为防止网络出现过拟合，网络采用一个隐层[21]。可以通过调节隐层的神经元数目，来使网络的性能达到最优。网络参数 μ 的初值设为 0.001，ϑ 设为 10，循环次数设为 100。

3.4 k-最近邻法[16][8]

最近邻分类器是以类比学习为基础的。令训练集 $D^N=\{\mathbf{x}_1,...\mathbf{x}_N\}$，训练样本用n维数值属性来描述。每个样本 $\mathbf{x}_i$ 代表n维空间中的一个点，所属的类别均已知。对于测试样本点 $\mathbf{x}$，在集合 D^N 中距离它最近的点记为 $\mathbf{x}'$。那么，最近邻规则的分类方法就是把点 $\mathbf{x}$ 分为 $\mathbf{x}'$ 所属的类别。最近邻规则是次优的方法，通常的误差率比最小可能误差率(即贝叶斯误差率)要大。然而，在无限训练样本的情况下，这个误差率至多不会超过贝叶斯误差率的两倍。

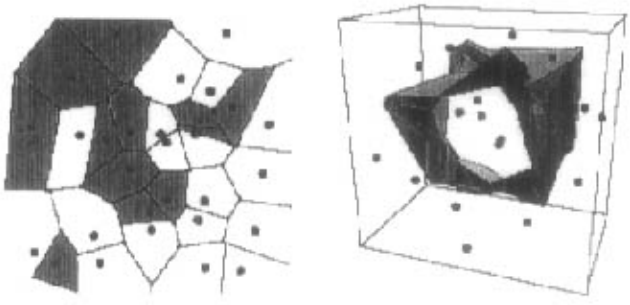
最近邻规则能够很好地工作的原理在于：赋予最近邻点的类标签 θ' 是一个随机变量。$\theta'=\omega_i$ 的概率无非就是后验概率 $P(\omega_i\mid\mathbf{x}')$。当样本个数非常大的时候，有理由认为 $\mathbf{x}'$ 距离 $\mathbf{x}$ 足够近，使得 $P(\omega_i\mid\mathbf{x}')\approx P(\omega_i\mid\mathbf{x})$。因为这恰好就是状态位于 ω_i 的概率，因此最近邻规则自然是真实概率的一个有效的近似。如果我们定义 $\omega_m(\mathbf{x})$ 为

$$P(\omega_m\mid\mathbf{x})=\max_i P(\omega_i\mid\mathbf{x})$$

那么贝叶斯规则总是选取 $\omega_m(\mathbf{x})$ 作为分类结果。最近邻规则允许我们把特征空间分成一个个的网格单元。每一个单元中的点，到最近邻 $\mathbf{x}'$ 的距离都比到别的样本点的距离要大。因此，这个小单元中的任意点的类别就与最近邻 $\mathbf{x}'$ 的类别相同。这称为空间Voronoi网格（参见图 3－4）。对于(a)图二维情况，最近邻算法使输入空间划分为Voronoi微元，每个微元贴上其所包含训练点的类别。对于(b)图的三维情况，微元是三维的，决策边界像晶体的表面。虽然需要更详细严谨的理论分析，但这些粗略的感性的观测结果使我们认识到，最近邻规则有比较好的结果并不是偶然的。

最近邻规则的一个推广就是k-最近邻规则。就像我们从这个规则的名称本身所能期望的那样，这个规则将一个测试数据点 $\mathbf{x}$ 分类为与它最接近的k个近邻中出现最多的那个类别(如图 3-5)。k-最近邻查询始于测试点，生长一个球形区

域，直到它包含k个训练样本，用这些样本的多数选票给测试点贴标签。在两类问题中，为了避免出现两类最近邻一样多的情况，通常令k取奇数。



(a) 二维Voronoi网格　　(b) 三维Voronoi网格

图 3－4 空间Voronoi网格

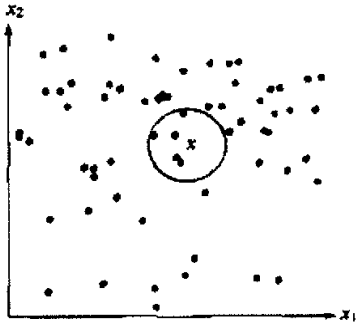


图 3－5 k-最近邻规则示意图

研究k-最近邻规则的原理来源于前面的有关自然概率的观察。首先注意到如果k值固定。并且允许训练样本个数趋向于无穷大，那么，所有的这k个近邻都将收敛于 $\mathbf{x}$。这样，如同最近邻规则一样，k个近邻的标记都是随机变量，概率 $P(\omega_i \mid \mathbf{x})$ 都是互相独立的。假设 $P(\omega_m \mid \mathbf{x})$ 是较大的那个后验概率，那么根据贝叶斯分类规则，我们总是选取类别 ω_m。最近邻规则则以概率 $P(\omega_m \mid \mathbf{x}')$ 选取类别 ω_m。而根据k-最近邻规则，只有当k个最近邻中的大多数的标记为此，才判决为类别 ω_m。做出这样选择的概率为

$$\sum_{i=(k+1)/2}^{k} \binom{k}{i} P(\omega_m \mid \mathbf{x})^i [1 - P(\omega_m \mid \mathbf{x})]^{k-i}$$

通常，k的值越大，选择类别 ω_m 的概率也越大。

接近性由Euclidean距离来定义，这里两个点 $X=(x_1,x_2,...,x_n)$ 和 $Y=(y_1,y_2,...,y_n)$ 之间的Euclidean距离是

$$d(X,Y)=\sqrt{\sum_{i=1}^{n}(x_i-y_i)^2}$$

将未知样本赋给它的k个最近邻中最普遍的类。当k=1 时，将未知样本赋给模式空间中离它最近的训练样本所属的类。

最近邻分类器是基于实例的或懒惰的学习器，因为它存储所有训练样本，并不建立分类器，直到需要将新样本分类。这一点与热切学习方法（比如决策树归纳和反向传播）相对，这些方法在收到要分类的新样本之前创建泛化模型。当与给定无标签样本相比较的潜在邻居的数目很大时，懒惰学习器可能引发昂贵的计算成本。因此，它需要高效的索引技术。懒惰学习器在训练时比热切方法快，但是在分类时慢，因为所有计算那时都要延迟。不像决策数归纳和反向传播，最近邻分类器赋给每个属性相等的权值。当数据中有许多无关的属性时，这样做可能导致混乱。

本文中，k-最近邻算法采用Matlab实现。数据预处理采用 2.3 节所述的聚类方法的改进，将数据由 684 维压缩到 24 维。沙尘暴预报中，采用了 3.2 节所述的时间序列分析预报方法，即增加了当天发生沙尘暴的站点数作为分类器的输入。因此，Euclidean距离是在 25 维空间中计算出来的。k-最近邻法是以距离为基础的分类器，所以提取出的特征在送入分类器之前，必须先进行规范化。将样本一半作为训练集，另一半作为测试集。可以通过调节k值，来使分类器的性能达到最优。

3.5 支持向量机[22]

支持向量机（Support Vector Machine，缩写为SVM）是基于线性机优化泛化性理论的训练方法，但它依赖于对数据的预处理，即在更高维的空间表达模式，并且通常比原来的特征空间的维数高很多。通过适当的到一个足够高维的非线性映射 $\varphi(\cdot)$，数据(属于两类)总能被一个超平面分割。如果假设每个模式 $\mathbf{x}_k$ 变换到 $\mathbf{y}_k=\varphi(\mathbf{x}_k)$，我们就把问题变为如何选择 $\varphi(\cdot)$。不同的泛化性定义启发了不同的算法，比如优化最大边沿裕量，最大软边沿裕量或最稀疏分开超平面。

支持向量机中最简单也是提出最早的模型是最大边沿裕量分类器。它的原理为：对n个模式中的每一个，$k=1,2,...,n$，根据模式属于 ω_1 或是 ω_2，我们分别令 $z_k=\pm 1$，增广空间 $\mathbf{y}$ 上的判别函数就是

$$g(\mathbf{y})=\mathbf{a}^T\mathbf{y}+m \qquad (3.10)$$

其中，**a** 为权向量。这里的权向量和变换后的模式向量都是增广的。这样，一个分隔超平面保证

$$z_k g(\mathbf{y}_k) \geq 1, \quad k = 1,...,n \tag{3.11}$$

如图 3－6 所示。

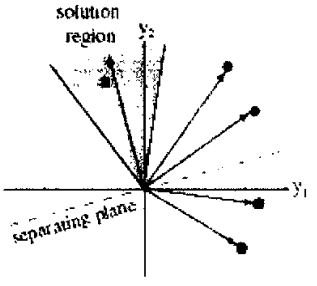


图 3－6 特征空间中的权向量和分隔超平面

设间隔b是到判定超平面的任何正的距离。训练一个支持向量机的目标是找到一个具有最大边沿裕量的分隔平面；如果边沿裕量越大，得到的分类器也越好。从超平面到(变换后的)模式**y** 的距离是$|g(\mathbf{y})|/\|\mathbf{a}\|$，如果正的间隔b存在的话，由式（3.11）推出

$$\frac{z_k g(\mathbf{y}_k)}{\|\mathbf{a}\|} \geq b, k = 1,...,n$$

我们的目标就是找到一个使得b最大化的权向量**a**。当然，解向量可以任意地伸缩，同时保持超平面不变，这样就保证了我们加上的限制条件$b\|\mathbf{a}\| = 1$，也就是方程（3.10）、（3.11）的解是$\|\mathbf{a}\|^2$的极小值。

支持向量是使式（3.11）等号成立的(变换后的)模式向量，也就是说，支持向量是接近超平面的(如图 3－7)。支持向量是那些定义最优分割超平面的训练样本，也是那些最难被分类的模式。所以说，训练一个支持向量机就是要找到最优超平面，即与最近训练模式距离最大的超平面。图 3-7 中，设离超平面距离为b，三个实心点为支持向量。非形式地说，它们就是对求解分类任务的最富有信息的模式。

总而言之，给定一个线性可分的训练样本

$$S = ((\mathbf{y}_1, z_1),...,(\mathbf{y}_n, z_n))$$

求解优化问题

$$\text{minimise} \quad <\mathbf{a}\cdot\mathbf{a}>$$

$$\text{subject to} \quad z_i(<\mathbf{a}\cdot\mathbf{y}_i> + m) \geq 1, \quad i = 1,...,n$$

可以得到超平面$(\mathbf{a},m)$，它实现了边沿裕量为$b=1/\|\mathbf{a}\|$的最大边沿裕量超平面。

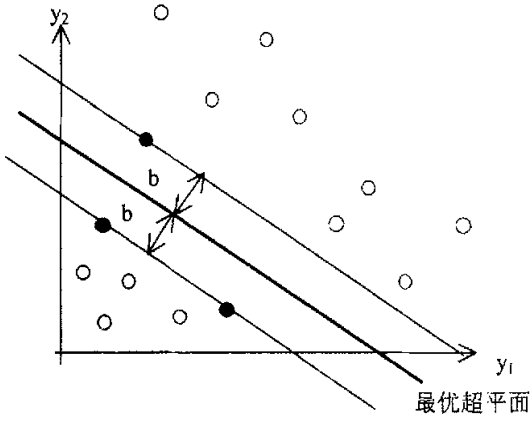


图 3－7 支持向量示意图

采用拉各朗日策略将优化问题转化为相应的对偶问题，原始的拉各朗日函数为

$$L(\mathbf{a},m,\boldsymbol{\alpha})=\frac{1}{2}\langle\mathbf{a}\cdot\mathbf{a}\rangle-\sum_{i=1}^{n}\alpha_i[z_i(\langle\mathbf{a}\cdot\mathbf{y}_i\rangle+m)-1]$$

这里$\alpha_i\geq 0$为拉各朗日乘子。通过对相应的$\mathbf{a}$和m求偏导，可以找到相应的对偶形式

$$\frac{\partial L(\mathbf{a},m,\boldsymbol{\alpha})}{\partial\mathbf{a}}=\mathbf{a}-\sum_{i=1}^{n}z_i\alpha_i\mathbf{y}_i=0$$

$$\frac{\partial L(\mathbf{a},m,\boldsymbol{\alpha})}{\partial m}=\sum_{i=1}^{n}z_i\alpha_i=0$$

将得到的关系式$\mathbf{a}=\sum_{i=1}^{n}z_i\alpha_i\mathbf{y}_i$代入到原始拉各朗日函数，得到

$$L(\mathbf{a},m,\boldsymbol{\alpha})=\frac{1}{2}\sum_{i,j=1}^{n}z_iz_j\alpha_i\alpha_j\langle\mathbf{y}_i\cdot\mathbf{y}_j\rangle-\sum_{i,j=1}^{n}z_iz_j\alpha_i\alpha_j\langle\mathbf{y}_i\cdot\mathbf{y}_j\rangle+\sum_{i=1}^{n}\alpha_i$$

$$=\sum_{i=1}^{n}\alpha_i-\frac{1}{2}\sum_{i,j=1}^{n}z_iz_j\alpha_i\alpha_j\langle\mathbf{y}_i\cdot\mathbf{y}_j\rangle$$

替换显示拉各朗日函数可以描述为训练点的线性组合，应用优化理论自然导出了对偶表示。在应用核函数的过程中需要对偶表示。

总结上述推导得到：考虑一个线性可分的训练集

$$S=((\mathbf{y}_1,z_1),...,(\mathbf{y}_n,z_n))$$

并假定参数$\boldsymbol{\alpha}^*$是下面二次规划问题的解

$$\text{maximise } W(\boldsymbol{\alpha})=\sum_{i=1}^{n}\alpha_i-\frac{1}{2}\sum_{i,j=1}^{n}z_iz_j\alpha_i\alpha_j\langle\mathbf{y}_i\cdot\mathbf{y}_j\rangle$$

$$\text{subject to } \sum_{i=1}^{n}z_i\alpha_i=0 \quad , \quad \alpha_i\geq 0 \, , \quad i=1,...,n$$

则权向量$\mathbf{a}^*=\sum_{i=1}^{n}z_i\alpha_i^*\mathbf{y}_i$实现了边沿裕量为

$$b=1/\|\mathbf{a}^*\|$$

的最大边沿裕量超平面。

m的值没有出现在对偶问题中，利用原始约束可以找到m^*

$$m^*=-\frac{\max_{z_i=-1}(\langle\mathbf{a}^*\cdot\mathbf{y}_i\rangle)+\min_{z_i=1}(\langle\mathbf{a}^*\cdot\mathbf{y}_i\rangle}{2}$$

可以注意到特征空间中的样本集$S=((\mathbf{y}_1,z_1),...,(\mathbf{y}_n,z_n))$在最终的二次优化问题中只以内积的形式使用。也就是说，样本集从低维的原始数据$\mathbf{x}_i$映射到高维空间的$\mathbf{y}_i$后，除了参与彼此的内积运算外，并没有参与任何其它的运算。因此，我们可以将从低维到高维的非线性映射函数$\varphi(\cdot)$和两个特征向量的内积函数$\langle\cdot\rangle$合并为一个函数，这个函数就是核函数。

定义：核是一个函数K，对所有$\mathbf{x},\mathbf{z}\in X$，满足

$$K(\mathbf{x},\mathbf{z})=\langle\varphi(\mathbf{x})\cdot\varphi(\mathbf{z})\rangle$$

这里φ是从X到特征空间Y的映射。

看起来似乎首先需要创建一个复杂的特征空间，然后计算出空间的内积，最后寻找一种直接的方式用原始输入计算内积值。而实际上往往是直接定义一个核函数，通过它隐式地定义特征空间。利用这种方式，不仅在计算内积时，而且在学习器的设计中都可以避开特征空间。要强调的是，为输入空间定义一个核函数通常比创立一个复杂的特征空间更自然。

在实现这个思路之前，首先要决定函数$K(\mathbf{x},\mathbf{z})$的哪些性质对于确定它是否适合某个特征空间是必要的。明显的是，函数必须是对称的

$$K(\mathbf{x},\mathbf{z})=\langle\varphi(\mathbf{x})\cdot\varphi(\mathbf{z})\rangle=\langle\varphi(\mathbf{z})\cdot\varphi(\mathbf{x})\rangle=K(\mathbf{z},\mathbf{x})$$

$K(\mathbf{x},\mathbf{z})$是真正对应于特征映射φ的核函数的充要条件是Mercer定理：令X是R^n的紧子集。假定K是连续对称函数，存在积分算子$T_K:L_2(X)\to L_2(X)$，使得

$$(T_Kf)(\cdot)=\int_X K(\cdot,\mathbf{x})f(\mathbf{x})d\mathbf{x}$$

是正的，也就是

$$\int_{X\times X} K(\mathbf{x},\mathbf{z})f(\mathbf{x})f(\mathbf{z})d\mathbf{x}d\mathbf{z} \geq 0 \quad \forall f \in L_2(X)$$

然后扩展 $K(\mathbf{x},\mathbf{z})$ 到一个一致收敛的序列（在 $X\times X$ 上），这个序列由 T_K 的特征函数 $\varphi_j \in L_2(X)$ 构成，归一化使得 $\|\varphi_j\|_{L_2}=1$，并且 $\lambda_j \geq 0$，则有

$$K(\mathbf{x},\mathbf{z}) = \sum_{j=1}^{\infty} \lambda_j \varphi_j(\mathbf{x})\varphi_j(\mathbf{z})$$

满足Mercer条件的流行核函数有：Gauss核

$$\exp(-\|\mathbf{x}_i - \mathbf{x}_j\|^2 / 2\sigma^2)$$

其中，σ 为分散性参数。r阶多项式核

$$(\mathbf{x}_i^T \mathbf{x}_j + 1)^r$$

S形函数

$$\tanh(\kappa \mathbf{x}_i^T \mathbf{x}_j + \theta)$$

在本文中，SVM算法采用Matlab编程实现，二次规划使用的是Matlab内嵌的二次规划函数。数据预处理采用 2.3 节所述的聚类方法的改进，其中数据由 684 维压缩到 24 维。沙尘暴预报中，采用了 3.2 节所述的时间序列分析预报方法，即增加了当天发生沙尘暴的站点数作为分类器的输入。因此，支持向量机将从 25 维空间映射到高维空间。所用的核函数为上述的Gauss核。可以通过调节分布性参数 σ，来使支持向量机的性能达到最优。

3.6 小结

本文尝试了用三种不同的分类方法进行未来一天的预报。这三种分类方法是：反向传播的改进 Levenberg-Marquardt 算法，k-最近邻法和支持向量机。三种方法的实现都是通过 Matlab 平台编程的。为了比较这三种方法和原有方法的优劣，三种方法的数据预处理均采用最好的降维方法——改进后的聚类（参见第二章）。因此，所有分类器的输入都为 25 维的特征向量。这样，影响系统结果好坏的唯一因素就在于它们的不同之处——分类方法。实验结果如表 3－1 所示。

表 3－1 四种分类方法预报结果

分类方法	BP	LM	k-最近邻	SVM
CSI	0.30	0.36	0.34	0.22

由于所用的 BP 和 LM 神经网络是全连接，没有经过泛化，所以网络结果不稳定。表中所给出的 CSI 值是多次实验的最佳值。

BP 网络取得最佳值时隐层神经元的个数为 9。LM 网络取得最佳值时隐层神经元的个数为 15。k-最近邻法取得最佳值时 k 为 71。SVM 取得最佳值时 σ 为 3990。

从表中可以看出，LM 和 k-最近邻法都比原有方法要好。支持向量机方法效果不好，原因在于选用的是最常用的核，但不一定是最适用于这一问题的核。进一步的改进计划见第五章。k-最近邻法虽然效果不如 LM，但运行稳定，多次测试的 CSI 值不变。

第四章 神经网络的泛化

4.1 主要泛化方法

上一章中我们谈到利用 LM 神经网络建立的沙尘暴预报模型的性能是几个方法中最好的。但美中不足的是，它和其它神经网络建模一样，具有不稳定性。也就是说，同一个网络，在结构，训练集和训练参数相同的情况下，每次训练得出的网络性能有很大不同。其原因在于：（1）网络中有许多权值在训练过程中得不到训练。这是由于所用训练样本数目相对于权值的数目要少造成的。（2）数据的内在关系本质上可能是不完全连接，而我们的初始假设认为网络为全连接。这样，网络中那些本来不必存在的权值在训练过程中自然得不到训练。（3）在 BP 一类的算法中，网络的初始权值是在训练开始前随机赋给的。如果训练过程中存在不完全训练，则训练完成后，有些权值就仍然会保持原有初值。初始权值赋值的随机性造成了不完全训练结果的多样性，进一步导致了网络性能的不稳定性。

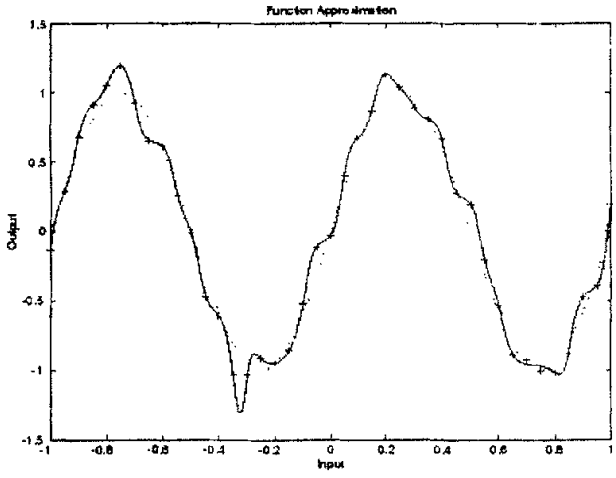
本章的目的就是要探讨克服网络不稳定性的方法。要克服网络的不稳定性，就必须铲除上面所述的造成网络不稳定性的几个原因。当训练样本数过少时，可以增加采样。但是，当训练样本数一定，且无法再增加时，可通过减小神经网络的权值数目来提高相对样本数目。减小权值数目不但可以解决全连接的问题，同时，还可以消除训练样本中的测量误差对网络造成的不良影响。权值的减少，造成网络的表达能力下降，因此记忆训练集中有噪声样本的可能性更小。这样，网络对新样本就能作出更好的分类，这种能力就是泛化。要提高网络的稳定性，本质上就是要提高网络的泛化能力。

利用所建模型对测试数据集进行预报，当结果正确时，网络被称作泛化性能良好。术语“泛化性”是从哲学中借用过来的。学习过程可以看作是一个曲线拟合问题。

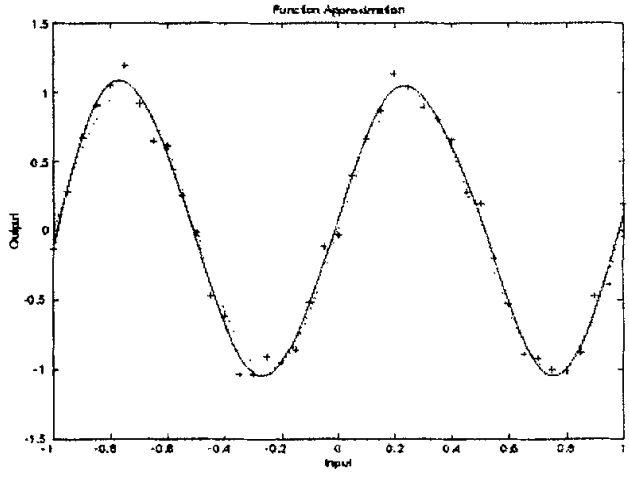
一个 1－20－1 的神经网络的响应如图 4－1 所示。通过对神经网络的训练来近似有噪声的正弦函数。图中虚线表示潜在的正弦函数，带误差的测量值用“＋”号给出，神经网络的输出拟合曲线用实线表示。很显然，图 4－1(a)所示网络的泛化性能较差，而图 4－1(b)所示网络的泛化性能良好。

当输入样本与训练网络的实例稍有不同时，泛化性能良好的网络将产生一个正确的输入－输出映射，如图 4－1(b)所示。但是，当网络结构过于复杂时，得出的网络可能会记住训练数据中的噪声。也就是说，它可能找到一个

仅在训练数据中出现的特征，但对于要建模的潜在函数，这个特征却不是真实的。这种现象称为过拟合现象或过训练现象。当网络过训练时，它就失去了在相似输入－输出模式间的泛化能力。



（a）过拟合



（b）泛化性良好

图 4－1 过拟合与泛化的比较

提高网络泛化性的方法有很多。其中包括：

1 通过减少隐层神经元，来达到减少网络权值数目的方法

该方法是一种经验性的方法，也是最简单的方法。如果网络中的权值数比总训练点数 n 还多，那么一个训练点还不够训练一个权值，必然会出现不完全训练。因此，必须使权值数比总训练点数 n 小得多。在全连接网络中，隐单元数决定了网络中总的权值数。一个简便的经验规则就是选取隐单元的

个数，使得网络的总权值数大致为 n / 10。这在很多实际问题中部取得较好的效果。

2 Bayesian 规则化将在本文中使用，详细描述见 4.2 节。

3 早期停止方法

该项技术中，可用的数据被分为三个子集。第一个子集是训练集，它被用来计算梯度，更新网络的权值和偏置。第二个子集是验证集。在训练的过程中，我们可以监控验证集的误差。在训练的初始阶段，验证误差与训练集误差一样，通常会下降。但是，当网络开始过拟合数据时，验证集误差一般会开始上升。当造成验证误差连续增加的循环次数达到指定循环数目时，训练停止，返回验证误差最小值时的权值和偏置。第三个子集是测试集，用来比较不同的模型。在训练过程中计算测试集误差也很有用。如果测试集误差达到最小值的循环数，与验证集误差达到最小值的循环数显著不同，那么这可能表明数据集的划分不合理。早期停止所获得的泛化拟合函数不如规则化所获得的函数平滑。

4 对不必要的权值连接进行剪枝[23] 很自然会想到，训练之后应该去掉那些幅值最小的权值。这种基于幅值的剪枝法也还行得通，但是可以证明，它不是最优的。因为有时小幅值的权值对于训练数据的学习也是非常关键的。

wald 统计法是一种优化的剪枝方法，其基本思想是：估计出模型中的某个参数的重要性，然后就可以消除最不重要的参数。而在神经网络中，这样的参数可以是某个权值。最佳脑损伤(optimal brain damage，OBD)算法和它的派生算法最佳脑外科(optimal brain surgeon，OBS)是这一思想的具体实现。它们利用二阶近似来预测训练误差对于某个特定权值的依赖程度，并且消除那些能导致训练误差的增加量最小的权值。

OBD 和 OBS 的基本方法是相同的，即将网络训练到权值 $\mathbf{w}^*$，使误差达到局部极小值，于是，导致训练误差增量最小的那个权值将被剪枝掉。

对于整个权值矢量的某个变化 $\delta\mathbf{w}$，预计的误差函数的增量为

$$\delta J = \underbrace{\left(\frac{\partial J}{\partial \mathbf{w}}\right)^T}_{\approx 0}\delta\mathbf{w} + \frac{1}{2}\delta\mathbf{w}^T\underbrace{\frac{\partial^2 J}{\partial \mathbf{w}^2}}_{=\mathbf{H}}\delta\mathbf{w} + \underbrace{O(\|\delta\mathbf{w}\|^3)}_{\approx 0}$$

其中 $\mathbf{H}$ 是 Hessian 矩阵。第一项可以去掉，这是因为网络目前正处于一个局部误差极小值处；忽略三阶以及更高阶项。在假定只去掉一个权值的限定下，最小化该函数的一般解为

$$\delta\mathbf{w} = -\frac{w_q}{\left[\mathbf{H}^{-1}\right]_{qq}}\mathbf{H}^{-1}\mathbf{u}_q \qquad (4.1)$$

$$L_q = \frac{1}{2}\frac{w_q^2}{[\mathbf{H}^{-1}]_{qq}} \tag{4.2}$$

这里，$\mathbf{u}_q$是权值空间中沿着第 q 个方向的单位向量，L_q是权值 q 的显著性的一个近似，也即，消除权值 q 并且其他权值通过式（4.1）进行更新所引起的训练误差的增量。

OBS 算法的步骤如下：

1 **begin initialize** $\mathbf{w}, \theta$（误差阈值停止条件）
2 训练一个适当大的网络达到最小误差
3 **do** 计算 $\mathbf{H}^{-1}$
4 $q^* \leftarrow$使式（4.2）最小的 q
5 将q^*代入式（4.1），求得$\delta\mathbf{w}\big|_{q^*}$，然后$\mathbf{w} \leftarrow \mathbf{w} - \delta\mathbf{w}\big|_{q^*}$
6 **until** $J(\mathbf{w}) > \theta$
7 **return** w
8 **end**

4.2 Bayesian 规则化

规则化(regularization)技术的一个常用作法是构造一个新的准则函数。用于训练前馈神经网络的典型性能函数是网络误差的均方和，即

$$J_{pat} = mse = \frac{1}{N}\sum_{i=1}^{N}(e_i)^2 = \frac{1}{N}\sum_{i=1}^{N}(t_i - a_i)^2$$

其中，N为训练集的样本数，t_i为第i个样本的目标输出，a_i为第i个样本的实际输出。而新准则函数不仅取决于典型的训练误差，还取决于分类器的复杂程度。更确切的说，新的准则函数对高度复杂的模型进行惩罚；在该准则下寻找极小值的过程也就是将训练集上的误差与复杂度进行折中和平衡的过程。形式上，可将新误差写成原来训练集上的误差再加上一个规则项，该项表示对解的约束或期望的属性，可由下式表示

$$J = \beta J_{pat} + \alpha J_{reg}$$

其中，参数β和α的大小决定了规则项作用的强弱程度，J_{reg}为

$$J_{reg} = msw = \frac{1}{n}\sum_{j=1}^{n} w_j^2$$

其中，n为网络中权值与偏置的总数，w_j为权值或偏置的值。显然权值衰减技术可用于这种形式，其中当权值将较大时，J_{reg}的值也较大。使用这个性能函数会使网络具有更小的权值和偏置。这将迫使网络响应更平滑，更不可能出现过拟合现象。

规则化的问题是很难确定性能比参数β和α的最优值。如果我们把参数设为$\beta >> \alpha$，可能会出现过拟合现象。如果设为$\beta << \alpha$，网络将不能充分地拟合训练数据。Bayesian 规则化是一套例行过程，它能够自动地设定规则化参数。

MacKay 提出的 Bayesian 规则化方法假设网络中的权值和偏置是服从 Gaussian 分布的随机变量，找出使数据集概率最大的α和β值。详细的推导过程见文献[24]。

下面是 Bayesian 规则化方法完成参数优化所需的步骤：

0. 初始化α，β和权值。我们选择设置$\alpha = 0$，$\beta = 1$。在第一步的训练后，目标函数的参数可以从初始设置恢复。
1. 执行一步 Levenberg-Marquardt 算法（见 3.3 节）以最小化目标函数$F(\mathbf{w}) = \beta J_{pat} + \alpha J_{reg}$。
2. 计算参数的有效数目$\gamma = n - 2\alpha tr(\mathbf{H}^{-1})$。$\mathbf{H}$的计算使用 Hessian 的 Gauss-Newton 近似。这一近似可以从 Levenberg-Marquardt 训练算法中获得：$\mathbf{H} = \nabla^2 F(\mathbf{w}) = 2\beta \mathbf{J}^T \mathbf{J} + 2\alpha \mathbf{I}_n$，其中$\mathbf{J}$是训练集误差的 Jacobian 矩阵。
3. 计算目标函数参数新的估计$\alpha = \dfrac{\gamma}{2J_{reg}(\mathbf{w})}$和$\beta = \dfrac{n-\gamma}{2J_{pat}(\mathbf{w})}$。
4. 循环第一步到第三步，直到收敛或达到指定的循环次数。

在本文中，Bayesian规则化所实现的泛化是建立在 3.3 节所述的LM改进神经网络基础上的。Bayesian规则化算法采用Matlab神经网络工具箱实现。

至此，网络具有了自动压缩不必要权值的功能，所以，我们不必担心初始设定的网络结构是否过于复杂。即便初始网络设为全连接，而隐层神经元数目又很高，也会使那些无关的权值通过训练被迫趋近于零。这时，我们反而会担心初始网络结构设置过于简单，以致于该有的重要网络权连接都无法得到保障。因此，这时设置初始网络结构的指导思想与以前大不一样，它要使网络尽可能地实现所有可能的连接。在本文中，我们设置的隐层神经元数目与输入层的神经元数目相同，都为 25 个。

实验中，为了和没有使用泛化的LM网络做比较，采用与之相同的训练集

和测试集，相同的μ值，ϑ值和循环次数。经过多次实验，结果表明，泛化后的网络有很好的稳定性，多次实验的CSI值稳定在0.34 ± 0.01。相比较而言，没有经过泛化的LM网络多次实验的CSI值变化幅度高达 0.21~0.36。因此，可以说，通过泛化可以使网络稳定性得到很大的提高。

第五章 结论与展望

5.1 本文结论

天气系统是一个包含有多种空间尺度、时间尺度和多种天气要素的多维复杂的系统，气象预报是一个很复杂的问题。近年来，数据挖掘技术越来越多的应用于气象预报问题。本文主要运用数据挖掘技术就“沙尘暴预报”的客观建模问题展开研究。

首先将数据样本降维。采用的主成份分析方法，将样本空间坐标轴变换，使变换后的各维按方差从大到小排列，取靠前的维作为代表样本的特征。对原有聚类降维方法进行改进。将原有方法中分场聚类组合生成样本典型模式，改为生成各场的典型模式，然后进行分场预报和综合预报。

接着，实现几种分类器。改进过的 BP 神经网络，使用 LM 算法使其具有快速收敛性。通过调整网络结构，对分类器进行优化。k 最近邻法，通过离待预报样本距离最近的若干训练样本的类标签，决定待预报样本的归属。通过调整最近邻数目，对分类器进行优化。支持向量机，将在低维空间中线性不可分的数据映射到高维的线性可分空间中，然后在高维空间中找到使类间空白最大的超平面作为分类器。通过调整核函数的参数，对分类器进行优化。最终进行性能比较，改进聚类降维方法与 LM 网络分类器联合使用，可以取得最佳效果，令 CSI 指标达到 0.36，同时维度降到 25 维。

然后，针对 LM 网络性能波动很大的特点，采用 Bayesian 规则化泛化技术，使不必要的权值尽可能地小，消除了随机权初值对性能造成的不稳定的影响。

最后，对所得到的系统的不足进行总结，找出将来可以进一步深入研究的领域。所得系统的不足包括支持向量机核函数的选择及其参数的调整，典型样本和非典型样本的剪辑方法。

沙尘暴预报是气象领域的一个崭新课题，实现对沙尘暴的有效预报需要多方的不懈努力和探索。消除沙尘暴危害最根本的办法还是有效的防止沙尘暴的发生，这需要全国，乃至全球人民的共同努力。

随着研究的深入，越发感觉到自己知识的浅薄，所以整个研究过程也是我不断学习和充实自己的过程。因时间和水平所限，特别是气象知识的不足，论文中尚有许多待完善的地方，希望大家提出宝贵意见。

5.2 问题分析

在以上几章所述的主要工作中仍然存在一些未解决的问题。这些问题主要包括：

1 在 2.2 节所述的主成份分析是一种最基本的降维方法。它的基本思想是对原数据进行线性变换，使新的坐标轴按重要性排序。因此，这种方法存在两点主要弊端，即：线性变换和无类标签监督。线性变换只能使原坐标轴旋转，找到方差最大的方向。而非线性变换可以使原坐标轴扭曲，从而可能找到方差更大的数据表示方法。无类标签监督的降维只会找到整个数据集方差最大的方向，但真正对分类有帮助的不是整个数据集最分散，而是两类分开得最明显。这就是说，在降维阶段就要考虑训练集类标签的指导作用，使得所找的方向上两类样本的斜方差最大，而不是整个数据集的方差。

2 在 3.2 节提出的时间序列分析预报中，要使用连续几天的数据作为神经网络的输入。而在我们实际的应用中，只增加了预报量提前一天的数据，因为每天压缩后的特征数仍然很大。另外，时间序列预报可以在提前一个时间单位预报的基础上作出提前更多时间单位的预报。我们这里的时间单位是天，因此，应该可以利用提前一天的预报结果作为输入，作出提前两天的预报。以此类推，作出提前多天的预报。但是，由于时间问题，尚未实现该功能。

3 在 3.5 节所述的支持向量机分类方法中，核函数及其参数的选择至关重要，本文对此仅作了初步的尝试。我们直接选择了使用最广泛的 Gauss 径向基函数作为核函数，其参数采用手工调整。实验结果证明性能并不理想。因此，需要根据具体的解决问题慎重选择核函数，同时采用优化技术确定核函数的参数。

4 在 4.2 节所述的 Bayesian 规则化，虽然取得了一定的稳定效果，但稳定的 CSI 值（0.34）较不稳定时的最高值 0.36 有所下降。如果看一下训练结束后的权值矩阵，可以发现，虽然算法迫使无关的权值尽可能的小，但最终的权值矩阵并没有一个单元真正为零。这说明网络中仍残余着一些不稳定因素，这是 CSI 值较最优有所下降的原因。另外，泛化的方法我们只实现了 Bayesian 规则化一种，无法与其它泛化方法比较优劣，从而找出最佳的方法。

5 最后，在本文中，分类方法都是单独使用的。其实，每种分类方法都有其局限性。只有将多种方法结合起来使用，才能取长补短，使分类性能达到最优。

5.3 展望

针对 5.2 节分析出的问题，这里作出如下展望：

一. 非线性和有监督的降维方法

多维放缩[16]是一种非线性的降维方法。其基本思想为：为了体现原来数据之间的关系，在低维空间上点与点的距离要与原始空间上点与点的距离(或相似性)相互对应。设在 n 个样本 $\mathbf{x}_1,...,\mathbf{x}_n$ 之间是可以计算距离的。令 $\mathbf{y}_i$ 为 $\mathbf{x}_i$ 在低维空间上的映像，δ_{ij} 代表 $\mathbf{x}_i$ 和 $\mathbf{x}_j$ 的距离，d_{ij} 为 $\mathbf{y}_i$ 和 $\mathbf{y}_j$ 的距离。现在我们寻求 $\mathbf{y}_1,...,\mathbf{y}_n$(即映像点的集合)构型，同时要求映像点之间的 $n(n-1)/2$ 个距离 d_{ij} 要尽量接近对应的原始距离 δ_{ij}。通常不可能对所有的距离都实现 $d_{ij}=\delta_{ij}$，所以我们需要下面的准则函数用来比较并选择几个不同候选答案。

$$J_{ef}=\frac{1}{\sum_{i<j}\delta_{ij}}\sum_{i<j}\frac{(d_{ij}-\delta_{ij})}{\delta_{ij}}$$

因为准则函数只牵涉点间距离，所以构型做刚体运动将不会改变它们的值。而且，它们都经过了归一化，所以原始数据点整体缩放不会影响它们的最小值。

一旦选取了准则函数后，就可以定义最优构型，即能够最小化准则函数的映像点集合。这个集合可以通过标准的梯度下降法求解：先给出 $\mathbf{y}_1,...,\mathbf{y}_n$ 的初始值，然后沿着准则函数下降最快的方向去调整 $\mathbf{y}_i$。因为低维空间的距离是 $d_{ij}=\|\mathbf{y}_i-\mathbf{y}_j\|$，它关于 $\mathbf{y}_i$ 的梯度就是沿着 $\mathbf{y}_i-\mathbf{y}_j$ 方向的单位向量，所以很容易得到准则函数的梯度

$$\nabla_{\mathbf{y}_k}J_{ef}=\frac{2}{\sum_{i<j}\delta_{ij}}\sum_{l\neq k}\frac{d_{kl}-\delta_{kl}}{\delta_{kl}}\frac{\mathbf{y}_k-\mathbf{y}_l}{d_{kl}}$$

初始构型可以随机选取，或者以任何使映像点散布方便的方式选取。如果映像点是在 d 维空间中，则一个简单而有效地获得初始值的方法就是只取原始样本向量对应方差最大的前 d 个分量。

有监督的降维从本质上讲，目的是使在最小维数特征空间中异类模式点相距较远(类间距离较大)，而同类模式点相距较近(类内距离较小)。在实现上述目标时，往往需要首先制定特征提取的准则，可直接以反映类内类间距离的函数作为准则，或直接以误判概率最小作为准则，也可以用类别判决函数作为准则，还可以构造与误判概率有关的判据来刻画特征对分类识别的贡献或者有效性。

基于离差阵的方法[25]属于直接以反映类内类间距离的函数作为准则。设 $\mathbf{S}_w$ 和 $\mathbf{S}_b$ 分别为原始特征空间中的类内和类间离差矩阵

$$\mathbf{S}_w = \sum_{i=1}^{c} P_i \frac{1}{N_i} \sum_{k=1}^{N_i} (\mathbf{x}_k^{(i)} - \mathbf{m}^{(i)})(\mathbf{x}_k^{(i)} - \mathbf{m}^{(i)})^T$$

$$\mathbf{S}_b = \sum_{i=1}^{c} P_i (\mathbf{m}_i - \mathbf{m})(\mathbf{m}_i - \mathbf{m})^T$$

其中，c 为类别数，P_i 是第 i 类的先验概率，N_i 是第 i 类的样本数，$\mathbf{x}_k^{(i)}$ 是第 i 类的第 k 个样本，$\mathbf{m}^{(i)}$ 是第 i 类的均值，$\mathbf{m}$ 是样本均值。$\mathbf{S}_w^*$ 和 $\mathbf{S}_b^*$ 分别为变换特征空间中的类内和类间离差矩阵。基于离差阵的准则函数为

$$J(\mathbf{W}) = tr[(\mathbf{S}_w^*)^{-1}\mathbf{S}_b^*]$$

$$= tr[(\mathbf{W}^T\mathbf{S}_w\mathbf{W})^{-1}(\mathbf{W}^T\mathbf{S}_b\mathbf{W})] = tr[\mathbf{S}_w^{-1}\mathbf{S}_b] = \sum_{i=1}^{n} \lambda_i$$

其中，$\mathbf{W}$ 为变换矩阵，λ_i 为 $\mathbf{S}_w^{-1}\mathbf{S}_b$ 的特征值。由此可以得出，取前 d 个较大的特征值所对应的特征矢量 $\mathbf{w}_i$(i=1，2．…，d)构造特征提取矩阵 $\mathbf{W}$ 对 $\mathbf{x}$ 作变换 $\mathbf{y} = \mathbf{W}^T\mathbf{x}$，这时对于给定的 d 所得到的准则 $J(\mathbf{W})$ 达最大值。

以上分别介绍了两种常用的非线性降维和有监督降维的方法，但是如何将两者结合起来，形成非线性有监督的降维方法还有待研究。

二．与降维相结合的时间序列预报

在前面的分析中，我们可以看出，阻碍进行时间序列预报的真正原因是降维所得到的特征维数仍然太高。以前面所述最佳的降维方法——改进后的聚类为例，降维后的特征向量为 24 维，加上预报量当天的数据，一共 25 维。如果将连续三天的数据作为输入，针对后面一天的天气进行沙尘暴预报，那么分类器的输入就会高达 75 维。更不要说，预报后面两天或三天的情况。

但是，我们看到，这 75 维并非完全独立。它是 25 维特征向量在三天内的变化情况序列。在这三天中，可能有些特征并没有多少变化，可能有些特征的变化是彼此一致的。这样可以说明，这 75 维数据中，有许多数据都是相关的，可能仍然存在大量的数据冗余。因此，这 75 维数据也有继续降维的必要。这样，我们将要做的就是，对前面改进聚类得到特征的多天组合再进行降维。有了这一步，分类器的输入维数就会降下来，5.1 节的第一个问题就可以解决了。

三．支持向量机核函数的选取及其参数的优化

SVM 的核函数的构造有多种途径：

可以从核函数中构造新的核函数。这时新的核函数必须满足 3.5 节所述的 Mercer 定理的条件。已证明满足该定理的核操作如下：

$$K(\mathbf{x},\mathbf{z}) = K_1(\mathbf{x},\mathbf{z}) + K_2(\mathbf{x},\mathbf{z})$$

$$K(\mathbf{x},\mathbf{z}) = aK_1(\mathbf{x},\mathbf{z})$$
$$K(\mathbf{x},\mathbf{z}) = K_1(\mathbf{x},\mathbf{z})K_2(\mathbf{x},\mathbf{z})$$
$$K(\mathbf{x},\mathbf{z}) = f(\mathbf{x})f(\mathbf{z})$$
$$K(\mathbf{x},\mathbf{z}) = K_3(\varphi(\mathbf{x}),\varphi(\mathbf{z}))$$
$$K(\mathbf{x},\mathbf{z}) = \mathbf{x}^T\mathbf{B}\mathbf{z}$$
$$K(\mathbf{x},\mathbf{z}) = p(K_1(\mathbf{x},\mathbf{z}))$$
$$K(\mathbf{x},\mathbf{z}) = \exp(K_1(\mathbf{x},\mathbf{z}))$$

其中，K_1和K_2是$X\times X$上的核，$a\in R^+$，$f(\cdot)$是X上的实值函数，$\varphi: X\to R^m$，K_3是$R^m\times R^m$上的核，$\mathbf{B}$是一个对称半正定的$n\times n$阶矩阵，$p(x)$是正系数多项式。

从特征空间中构造核函数。从特征空间开始，通过内积得到核函数。在这种情况下，只要内积的选取符合内积的定义，就不需要检查 Mercer 定理的条件。

直接从数据中求得核。具体内容可参考[26]和[27]。

四. 新的泛化方法

文献[28]中提出的一种新的泛化方法——早停 Bayesian 规则化（EBR），它通过用早期停止算法预训练初始网络，能够增强 ANN Bayesian 规则化（BR）的性能。应用这一方法于前馈神经网络的规则化，以回归三个基准数据序列。交叉验证误差和训练次数都比标准的 Bayesian 规则化有显著下降。

在 MacKay 的原始工作中，BR 训练过程始于一个随机初始网络，它的权值是从给定的先验分布中采样出来的。但是，但是这些不同的初始网络可能收敛于不同的局部最优，并表现出不同的收敛属性和网络性能。这里建议的良好预训练的初始网络有两个好处：第一，初始网络的 Hessian 矩阵经过更好的调节，改善了收敛属性；第二，初始网络距离选定的最优值更近，缩短了优化路径，减少了昂贵的 BR 训练量（$O(\mathbf{w}^3)$）

EBR 的步骤描述如下：

Step 1: 设$i=0$。随机初始化$\mathbf{w}_{ES}^0$。

Step 2: SCG(缩小共轭梯度)在内 ES 循环中优化$\mathbf{w}_{ES}^i\to\mathbf{w}_{ES}^{i+1}$。$i=i+1$。

Step 3: 检查$\mathbf{w}_{ES}^i$的验证误差ev_{ES}^i，将最佳权值保存记录$\mathbf{w}_{ES}^i\to\mathbf{w}_{ES}^{best}$。如果超过$stop_{ES}$个循环后没有观测到更好的$\mathbf{w}_{ES}^i$，设置$\mathbf{w}_{ES}^{best}\to\mathbf{w}_{BR}$并向前到 Step 4；否则，返回 Step 2。

Step 4: 设置$i=0$。随机初始化α^0

Step 5: SCG(缩小共轭梯度)在内 BR 循环中优化$\mathbf{w}_{BR}^i\to\mathbf{w}_{BR}^{i+1}$。执行证据最大化算法以更新$\alpha^i\to\alpha^{i+1}$。$i=i+1$。

Step 6: 检查$\mathbf{w}'_{BR}$的验证误差ev'_{BR}，将最佳权值保存记录$\mathbf{w}'_{BR} \rightarrow \mathbf{w}^{best}_{BR}$。如果超过$stop_{BR}$个循环后没有观测到更好的$\mathbf{w}'_{BR}$，停止并输出$w^{best}_{BR}$；否则，返回 Step 5。

五．多分类方法的综合预报

在已有的工作基础中，王汉芝的论文采用了一种多次预报的方法。在这一方法中，首先用模糊神经网对数据进行分类。对于有些无法正确分类的非典型样本，再重新对它们进行聚类特征提取，然后用一个新的模糊神经网对非典型样本进行分类。这一方法取得了一定的分类性能提升，但也存在一些问题。对全部数据的分类和对非典型样本的分类采用的是相同的分类方法。既然这一方法在第一次分类中，已经证明对非典型样本无能为力，在第二次分类中就没有必要再使用相同的分类方法。

我们可以将第二次分类所使用的方法换成另一种分类方法。这样做的基本思想是，先用一种分类方法对整个样本进行分类，对于无法正确分类的样本，再采用另一种分类方法，利用其相对于第一种方法的优势进行分类。这样可以做到兼俱两种分类器的长处，使分类的效果达到最佳。

参考文献

[1] 赵兴梁, 甘肃特大沙尘暴的危害与对策, 中国沙漠，1993，13(3):1-7

[2] 王式功 董光荣，沙尘暴研究的进展，中国沙漠，2000，20(4): 349~356

[3] 丑纪范 许以平，天气预报，北京：气象出版社，1985

[4] B.H. Barnum, N.S. Winstead, J. Werely, A. Hacola, P.R. Colarco, O.B. Toon, P. Ginoux, G. Brooks, L. HasselBarth, B. Toth, Forecasting dust storms using the CARMA-dust model and MM5 weather data, Environmental modelling and software, 2004, 19: 129-140

[5] S. Nickovic, S. Dobricic, A model for long-range transport of desert dust, Monthly weather, 1996, 124: 2537-2544

[6] T.F. Hogan, T.E. Rosmond, The description of the Navy operational global atmospheric prediction system's spectral forecast model, Monthly weather, 1991, 119: 1786-1815

[7] Y. Shao, A model for mineral dust emission, Geophys, 2001, 106: 20239-20254

[8] J. Han, M. Camber, Data mining: concepts and techniques, Morgan Kaufmann, 2000

[9] 岳斌，基于神经网络的沙尘暴预报模型的研究与应用：[硕士学位论文]，天津：天津大学，2002

[10] 王汉芝，基于模糊神经网的沙尘暴预报模型：[硕士学位论文]，天津：天津大学，2004

[11] 张承福，人工神经网络在天气预报中的应用研究，气象人工智能专辑（一），1994

[12] J.P. Marques, Pattern recognition: concepts, methods, and applications, Springer-Verlag, 2001

[13] A.R. Webb, Statistical pattern recognition, second edition, John Wiley & sons, 2002

[14] H. Hotelling, Analysis of a complex of statistical variables into principal components, Journal of educational Psychology, 1993, 24:417-444

[15] 路志英，赵智超，郝为，林孔元，刘还珠，基于人工神经网络的多模型综合预报方法，计算机应用，2004，24(4): 50-51, 88

[16] R.O. Duda, P.E. Hart, D.G. Stork, Pattern classification (second edition), Wiley-Interscience, 2000

[17] G. Box, G.M. Jenkins, G. Reinsel, Time series analysis: forecasting & control, Pentice hall, 1994

[18] C.M. Zealand, D.H. Burn, S.P. Simonovic, Short term streamflow forecasting

using artificial neural networks, Journal of hydrology, 1999, 214:32-48

[19] K. Hsu, H.V. Gupta, S. Sorooshian, Artificial neural network modeling of the rainfall-runoff process, Water resource research, 1995, 31(10): 2517-2530

[20] M.T. Hagan, H.B. Demuth, M.H. Beale, Neural network design (electrical engineering), Brooks Cole, 1995

[21] S. Ranjithan, J.W. Eheart, J.H. Garrett Jr, Neural network based screening for groundwater reclamation under uncertainty, Water Resour. Res., 1993, 29(3): 563-574

[22] N. Christianini, J. Shawe-Tayor, An introduction to support vector machines and other kernel-based learning method, Cambridge university press, 2000

[23] B. hassibi, D.G. Stork, G.J. Wolff, Optimal brain surgeon and general network pruning, Proceedings of the IEEE international joint conference on neural networks, 1992, 2: 441-444

[24] F.D. Foresee, M.T. Hagan, Gauss-Newton application to Bayesian learning, International conference on neural networks, 1997: 1930-1935

[25] 孙即祥，王晓华，钟山，张帆，史慧敏，模式识别中的特征提取与计算机视觉不变量，北京：国防工业出版社，2001

[26] T.S. Jaakkola, D. Haussler, Exploiting generative models in discrimitive classifiers, Advances in neural information processing systems, MIT press, 1998

[27] N. Christianini, J. Shawe-Taylor, C. Campbell, Dynamically adapting kernels in support vector machines, Advances in neural information processing systems, Mit press, 1998

[28] Z.S.H. Chan, H.W. Ngan, A.B. Rad, Improving Bayesian regularization of ANN via pre-training with early stopping, Neural processing letters, 2003, 18: 29-34

[29] 边肇祺，张学工等，模式识别，北京：清华大学出版社，1999

[30] 郝为，基于计算智能的多模型气象综合预报：[硕士学位论文]，天津：天津大学，2000

[31] G. Dong, X. Zhang, L. Wong, J. Li, CAEP: classification by aggregating emerging patterns, Proceeding DS'99, LNAI 1721, Tokyo, Japan, 1999

[32] L.E. Scales, Introduction to non-linear optimization, New York: Springer-Verlag, 1985

[33] B.E. Boser, I.M. Guyon, V.N. Vapnik, A training algorithm for optimal margin classifiers, Proceedings of the 5th annual ACM workshop on computational learning thoery, 1992: 144-152

[34] C. Cortes, V. Vapnik, Support vector networks, Machine learning, 1995. 20: 273-297

[35] J. Shawe-Tayor, P.L. Bartlett, R.C. Williamson, M. Anthony, Structural risk minimization over data-dependent hierachies, IEEE transactions on information thoery, 1998, 44(5): 1926-1940

[36] J. Shawe-Taylor, N. Christianini, Margin distribution and soft margin, Advances

in large margin classifiers, MIT Press, 1999

[37] V. Vapnik, The nature of statistical learning theory, Springer-Verlag, 1995

[38] V. Vapnik, Statistical learning theory, Wiley,1998

[39] O.L. Mangasarian, Generalized support vector machines, Technical report mathematical Programming TR 98-14, University of Wisconsin in Madison, 1998

[40] J.E. Jackson, A user's guide to principal components, New York: Wiley

[41] W.H. Press, S.A. Teukolosky, W.T. Vetterling, B.P. Flannery, Numerical recipes in C: the art of scientific computing, UK: Cambridge university press, 1996

[42] B.B. Hubbard, The world according to wavelets, Wellesley, MA: A.K. Peters, 1996

[43] M. Muralikrishna, D.J. DeWitt, Equi-depth histograms for estimating selectivity factors for multi-dimensional queries, Proc. 1988 ACM-SIGMOD Int. Conf. management of data, 1988:28-36

[44] Y. Le Cun, J.S. Denker, S.A. Solla, Optimal brain damage, Advances in neural information processing system 2, San Matoe, CA: Morgan Kauffman, 1990

[45] W.S. Sarle,Stopped training and other remedies for overfitting, Proceedings of the 27th symposium on interface, 1995

发表论文和参加科研情况说明

一．论文发表及录用情况

《基于人工神经网络的多模型综合预报方法》　　2004 年 4 月　　计算机应用

根据天气系统非线性变化及天气变化受大气多种内外因素综合影响的特点，本文提出了用 ANN 的前馈网络(BP 算法)串入竞争自组织映射网络(SOM 网络)方法对同一预报量进行不同结构类型的 MOS 模型、动力诊断模型和人工智能模型的综合预报。利用这一系统对样本进行了“先聚类后训练”的预报。结果表明，BP+SOM 网络实现多模型(异型)综合预报系统具有很好的应用前景。

二．参加科研情况

基于数据挖掘的沙尘暴预报，利用模式识别、人工智能等，在 Matlab 编程平台上建立沙尘暴预报模型，成功地解决了数据降维、特征提取及综合等较难的课题，模型具有较高预报准确率。

致谢

本论文是在导师路志英副教授和林孔元教授的悉心指导下完成的。在两年多的研究生学习过程中，始终得到这两位老师的热切关怀与悉心指导。二位老师对模式识别与智能系统学科的知识均有深刻的领悟。路老师对待事业认真、严谨的敬业精神，林老师在大方向的指引和开放的思维方式，二位老师的言传身授，都使我获益非浅，是我终生的精神财富。在此对路老师、林老师表示衷心的感谢。

感谢课题组的刘正光教授、王萍教授、杨正瓴副教授等给我的热心指点和帮助。

最后还要向帮助过我的课题组的各位同学表示真诚的谢意。