分类号 TP3 密级

UDC 编号

中国科学院研究生院

硕士学位论文

支持向量机增量学习、样本选择及新模型研究

刘秋阁

指导教师 何清 副研究员

中国科学院计算技术研究所

申请学位级别 工学硕士 学科专业名称 计算机软件与理论

论文提交日期 2008 年 4 月 论文答辩日期 2008 年 6 月

培养单位 中国科学院计算技术研究所

学位授予单位 中国科学院研究生院

答辩委员会主席

Incremental Learning and Instance Seclection of SVM and New Classifier Model

by

Liu Qiuge

Dissertation submitted to

Graduate University of Chinese Academy of Sciences

in partial fulfillment of the requirements

for the degree of

Master of Science in Computer Science

Dissertation Supervisor: Professor He Qing

Institute of Computing Technology

Chinese Academy of Sciences

March 2008

声　　明

我声明本论文是我本人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除了文中特别加以标注和致谢的地方外，本论文中不包含其他人已经发表或撰写过的研究成果。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。

作者签名：　　日期：

论文版权使用授权书

本人授权中国科学院计算技术研究所可以保留并向国家有关部门或机构送交本论文的复印件和电子文档，允许本论文被查阅和借阅，可以将本论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存、汇编本论文。

（保密论文在解密后适用本授权书。）

作者签名：　　导师签名：　　日期：

本文受

国家自然科学基金“基于超曲面的覆盖分类算法与理论研究” No. 60675010；
863 高技术探索项目“基于感知机理的智能信息处理技术”2006AA01Z128；
国家自然科学基金“基于感知学习和语言认知的智能计算模型研究”No. 60435010；
973 项目子课题“语义网格资源描述模型、形式化理论和支撑技术”No. 2003CB317004；
973 项目子课题“非结构化信息(图像)的内容理解与语义表征”No. 2007CB311004；
北京市自然科学基金“海量高维、多类数据分类法研究及其应用”No. 4052025；

资助

摘 要

本论文所做工作是与支持向量机（Support Vector Machine - SVM）的在线增量学习、样本选择技术及新 SVM 分类模型相关的一些研究。

关于线性临近支持向量机（Proximal SVM – PSVM）存在着简单的训练算法，且适合进行增量学习；但非线性 PSVM（Nolinear Proximal SVM – NPSVM）的训练要求对一个大小为训练样本点个数平方的矩阵求逆，存在着空间复杂度过大的问题；而且 NPSVM 不能像线性情形那样进行增量学习。从这些问题出发，我们做了如下的工作。

- 为了使 NPSVM 能够更加快速的进行在线增量学习，本文基于一个新的非线性 PSVM 模型设计了一种新的增量学习算法。该算法从新模型解的形式出发，利用分块矩阵求逆公式，有效地利用 NPSVM 分类器的历史训练结果，减少了在线学习过程中的重复学习，完成增量学习过程。理论推导及实验结果显示在线学习过程中采用该算法不仅可以得到与批量学习相同的分类器、正确率；而且解决了重复学习的问题，可以显著地缩短训练时间。
- NPSVM 的空间复杂度是与样本个数的平方成正比的，为了处理大数据集的增量学习问题，本文设计了针对历史数据、新数据和非线性数据分类器的样本选择技术。在线学习过程中，该样本选择技术不仅能够选择出历史数据集中最具代表性的样本点，而且能够选择出新数据中最具价值的样本点；此外对于比较复杂的非线性分类器也特别设计了相应的样本选择方法。实验显示上述样本选择技术仅需付出较小时间代价，就可以有效地处理大样本集的在线学习问题，而且可以得到与利用全部样本进行训练的结果相近的正确率。
- 设计了一种新的非线性 SVM 学习算法——Extreme SVM（ESVM）。ESVM 是一种基于正则化最小二乘法的新的 SVM 分类器。与其它所有非线性 SVM 学习方法不同的是，ESVM 不是使用核函数来得到非线性分类器，而是显式地构造了一个非线性的随机映射函数将输入样本点映射到一个特征空间中，然后在该特征空间中学习一个线性的分类器。该方法基于单隐层前馈神经网络（Single hidden Layer Feedforward Networks - SLFNs）的学习机制，在保持 SLFNs 学习能力的前提下，其输入权重可以随机地确定而不需要训练，这样 SLFNs 隐层神经元的作用相当于一个映射函数。理论分析及实验结果表明: ESVM 可以有效地应用于大数据集的训练，不仅具有与 SVM 相当的正确率，而且极大地缩短了训练时间。另外，与 SLFNs 的学习算法 ELM 相比，ESVM 将正则化理论引入到 SLFNs 的训练中，具有比 ELM 更好的泛化能力。

关键词：分类学习算法；SVM；PSVM；增量学习算法；样本选择技术；ESVM

Incremental Learning and Instance Seclection Technique of SVM and New Classifier Model

Liu Qiuge

Directed By He Qing

In this paper we formulated our works about incremental learning and instance selection technique based on Support Vector Machine (SVM) and new SVM classifying model.

To obtain a linear Proximal SVM (PSVM) classifier, all that is needed is the inversion of a small matrix of the order of the input space dimension, typically of the order of 100 or less, even if there are millions of data points to classify. For a nonlinear classifier, however, a linear system of equations of the order of the number of data points needs to be solved. For larger datasets, we will be in face of the problem of large space complexity.

- An incremental learning method based on a new nonlinear PSVM model is proposed in this paper, utilizing which we can perform online learning of nonlinear PSVM classifier efficiently. Test results demonstrate that the computation time of this new NPSVM is shorter than standard NPSVM with similar accuracy, and this incremental learning meatod reduce training time evidently while still hold same correctness as that of learning in standard way.
- A data selection technique which can be used to solve large dataset online incremental learning problem is introduced in this paper. The selected set of data points, which can be discretionarily smaller the entire dataset, completely characterize a separation plane classifier. This makes it a useful incremental classification tool which maintains only a small fraction of a large data set before merging and processing it with new incoming data. Mathematical analysis and experimental results demonstrate the effectivity of our proposed technique both in batch mode and online learning situation.
- Through constructing a nonlinear mapping function explicitly, a new nonlinear support vector machine classifier is devised. It can be interpreted as a special regularized least squares, and the solution of it requires only a single system of linear equations, the order of which is independent to the size of training data set. The experimental results show that this new classifier can lead to shorter training time than standard SVM with comparable accuracy and better generalization performance than SLFN.

Keywords: Classifying algorithm, SVM, PSVM, Incremental learning algorithm, Instance selection algorithm, ESVM

目 录

摘　要 ..... **I**

目 录 ..... **V**

图目录 ..... **VIII**

表目录 ..... **IX**

第一章 引言 **11**

1.1 数据挖掘技术的出现 11
1.2 数据挖掘中的分类方法 12
1.2.1 分类问题的提出 12
1.2.2 决策树学习 13
1.2.3 贝叶斯分类 14
1.2.4 人工神经网络 14
1.2.5 关联规则分类 15
1.2.6 支持向量机 15
1.2.7 分类算法的评价标准 15
1.3 本文的贡献 16
1.4 本文的组织 17

第二章 理论基础 **19**

2.1 分类问题的表示 19
2.2 经验风险最小化 20
2.3 复杂性与推广能力 20
2.4 统计学习理论— 支持向量机的理论背景 20
2.4.1 函数集的 VC 维及推广性 21
2.4.2 结构风险最小化 22
2.5 支持向量机 22
2.6 支持向量机求解算法 26
2.6.1 化为无约束问题 26
2.6.2 内点算法 26
2.6.3 求解大型问题的算法 27
2.6.4 选块算法(Chunking) 27

2.6.5 分解算法（Decomposing） 28
2.6.6 序列最小最优化（Sequential Minimal Optimization, SMO）算法 28
2.7 增量支持向量机 28
2.8 支持向量机的一些进展 30
2.8.1 邻近支持向量机 —— Proximal SVM 30
2.8.2 大数据集 SVM 学习算法 30
2.8.3 极限学习机 —— Extreme Learning Machine 31

第三章 非线性邻近支持向量机的增量学习算法 33

3.1 研究工作基础 34
3.2 新的非线性 PSVM 分类机 38
3.3 非线性邻近支持向量机的增量学习算法 39
3.4 实验结果 41
3.5 总结及下一步的工作方向 42

第四章 PSVM 样本选择技术 43

4.1 研究工作摘要 43
4.2 PSVM 样本选择技术 45
4.3 实验结果 49
4.4 总结 51

第五章 新的非线性支持向量机学习模型 —— Extreme SVM 53

5.1 引言 53
5.2 相关工作回顾 55
5.2.1 单隐层前馈神经网络 55
5.2.2 极限学习算法 56
5.3 极限支持向量分类器——Extreme SVM 57
5.3.1 线性极限支持向量机分类器 57
5.3.2 非线性极限支持向量机分类器 58
5.3.3 极限支持向量机与正则化网络的关系 60
5.3.4 极限支持向量机与非线性 PSVM 之间的关系 61
5.3.5 极限支持向量机与极限学习机之间的关系 62
5.3.6 极限支持向量机与标准 SVM 之间的区别 62
5.4. 测试结果 62
5.5 总结 63

第六章 结束语 67

6.1 本文工作总结 67
6.2 下一步研究方向 68
参考文献 69
致　谢 i
作者简历 iii

图目录

图 3.1. 标准 SVM 33

图 3.2 邻近 SVM 34

图 4.1 历史数据样本选择技术流程图 46

图 4.2 添加了对新到数据集进行选择的样本选择技术流程图 47

图 4.1 在数据集 Iris 和 Ionosphere 上 NPSVM-I、NPSVM-II 和 NPSVM-III 的 10 折训练、测试正确率 51

表目录

表 3.1 NPSVM 和 NNPSVM 在 Ionosphere, Bupa Liver, Tic-Tac-Toe 上的平均召回率、10 折交叉测试正确率和训练时间，参数 v 由对训练数据集的 10 折交叉确认确定 41

表 3.2 NPSVM 和增量学习算法在 Ionosphere, Bupa Liver, Tic-Tac-Toe 上的召回率及训练时间；参数 v 由对训练数据集的 10 折交叉确认确定..... 42

表 4.1 线性 PSVM 和 PSVM-I 分类器在数据集：Iris，Ionosphere，Tic-Tac-Toe 上的 10 折训练、测试正确率； 50

表 4.2 在 4 个 UCI 公共数据集：Iris，Ionosphere，Tic-Tac-Toe 和 BUPA 上的 RSVM 的 10 折测试正确率，及利用增量学习算法和样本选择技术得到的 NPSVM-III 的 10 折测试正确率..... 51

表 5.1 ESVM，SVM，NPSVM 在 8 个数据集上的训练、测试正确率及训练时间比较 66

第一章 引言

近年来，数据挖掘引起了信息产业界极大的关注。究其原因，主要是很多领域，如商务管理、生产控制、市场分析、工程设计、科学探索等，积累了大量的数据，迫切需要将这些数据转换成有用的信息和知识。例如，随着电子商务的流行，零售业积累了大量的顾客购买历史记录，货物进出、消费与服务记录等数据，而且其数据量在不断的迅速膨胀。没有强有力的工具，理解分析这些快速增长的海量数据已经远远超出了人的能力。而数据挖掘技术正是从大量数据中提取或“挖掘”知识的技术，例如对于零售业来说，数据挖掘技术可以有助于识别顾客购买行为，发现顾客购买模式和趋势，改进服务质量，取得更好的顾客满意程度，设计更好的分销策略，从而为商家的决策制定提供强有力的支持。

根据可以挖掘什么类型的模式，数据挖掘技术可以划分为：用于数据一般特征汇总的概念描述技术，关联规则技术，用于对未知数据进行分类、预测的技术，对已知数据进行聚类分组的技术，异常数据发现技术，用于描述行为随时间变化的对象的规律的演变分析技术等等。本篇论文将专注于分类技术的研究。

1.1 数据挖掘技术的出现

数据挖掘是信息技术演化的一个自然结果：数据收集和数据库创建机制的早期开发成为稍后数据存储和检索、查询和事务处理有效机制开发的必备基础，随着提供查询和事务处理的大量数据库系统广泛付诸实践，数据分析和理解自然成为下一个目标。

自上世纪 60 年代以来,数据库和信息技术已经从原始的文件处理演化到复杂的数据库系统；70 年代以来，数据库系统的研究和开发已经从层次和网状数据库系统发展到开发关系数据库系统、数据建模工具、索引和数据组织技术；自 80 年代中期以来，出现了很多新的数据模型，如扩充关系模型、面向对象模型、对象-关系模型和演绎模型等，面向空间的、时间的、多媒体的、主动的、知识库在内应用的数据库系统百花齐放，设计分布性、多样性和数据共享问题被广泛研究，异种数据库和基于 Internet 的全球信息系统也已出现。

在 30 多年里，随着数据库和信息产业迅速的发展，快速增长的海量数据收集、存放在大型和大量数据库中，没有强有力的工具，理解它们已经远远超出了人的能力。重要的决定常常不是基于数据库中信息丰富的数据，而是给予决策者的直觉，因为决策者缺乏从海量数据中提取有价值知识的工具。

一种解决方案是专家系统技术，这种系统依赖用户或领域专家人工的将知识输入知识库，然而这一过程常常有偏差和错误，并且耗时、费用高。相对比，数据挖掘技术则可以自动的从大量数据中“挖掘”知识。

按照数据挖掘技术能够发现的模式类型，可以把主要的数据挖掘技术分为关联规则、聚类分析、概念描述、孤立点分析、数据演变分析、分类预测等类型。

数据可以与类或概念相关联，利用数据挖掘技术自动发现某个类或概念的简洁的描述被称为类/概念描述。例如，对一个超市中比较大的客户特征进行汇总描述，可能得到这些顾客的一般轮廓，如年龄在 30-40、有工作、有很好的信用等级等等；或对经常购买计算机产品的顾客和偶尔购买的顾客进行区分比较。

关联规则问题的提出最初的动机是超市的购物篮分析，即分析超市商品的销售数据，探寻顾客的购物行为规律。一个关联规则的例子就是：{黄油，面包}⇒{牛奶}（90%），表示购买黄油与面包的顾客中有90%也购买了牛奶。这条关联规则的前件是黄油与面包，后件是牛奶，90%是这条规则的可信度。这有助于超市管理者制定合适的销售策略。

聚类分析可以将对象划分为一些簇，使得同一簇中的对象具有很高的相似性，而与其他簇中的对象很不相似。聚类具有很多的应用，如聚类能帮助市场分析人员从客户基本库中发现不同的客户群；在生物学上用于推导动植物的分类等。

孤立点是指与数据的一般行为或模型不一致的数据对象，大部分数据挖掘方法将孤立点视为噪声或异常而丢弃。然而，在一些应用中，如信用卡欺骗检测，罕见的事件可能比正常出现的那些更有趣，而孤立点分析技术则可以检测出这些潜在的“异常”。

分类的目的是根据数据集的特点构造一个分类函数或分类模型（也常常称作分类器），该模型能把未知类别的样本映射到某一个给定类别。分类问题包括两个阶段：训练和预测。在训练阶段，分类算法从具有类别标记的训练实例中学到一个分类模型，该模型把训练实例映射到给定的类别；在预测阶段，利用该分类器对没有类别标记的实例预测其类别，预测的准确程度可以评价分类器的性能。分类给出的是数据对象离散的类别标记，而预测给出的是某些空缺的或不知道的数值数据（可以是连续的）。

本篇论文的工作主要是针对数据挖掘技术中的分类方法所做的研究，下面对各种主要的分类技术做一些简单地介绍。

1.2 数据挖掘中的分类方法

分类问题不是新问题，但是计算机的普及应用，特别是机器学习和数据挖掘的迅速发展赋予了他们新的意义，再次引起了人们热切地关注。分类是机器学习的一个重要任务和目标，是许多其它问题的基础，目前在研究和商业上的应用非常广泛。

1.2.1 分类问题的提出

为了完全确诊某些疾病，可能需要进行创伤性探测或者昂贵的手段。因此利用一些有关的容易获得的临床指标进行推断，是一项有意义地工作。美国 Cleveland Heart Disease Database 提供的数据，就是这方面工作的一个实例。在那里对 297 个病人进行了彻底地临床检测，确诊了他们是否有心脏病。同时也记录了他们的年龄、胆固醇等 13 项有关指标。他们希望根据这些临床资料，对新来的病人只检测这 13 项指标，就推断该病人是否

有心脏病。这类问题就属于分类问题。

一般地，可考虑n维空间上的分类问题，它包含n个指标（即$x \in R^n$）和l个样本点。记这l个样本点的集合为：

$$T = \{(x_1, y_1), ...(x_l, y_l)\} \in (X \times Y)^l . \quad (1.1)$$

其中$x_i \in X = R^n$是输入向量，$y_i \in Y = \{-1,1\}$是输出向量，$i = 1,...l$。这l个样本点组成的集合称为训练集，这时分类问题就是，对任意给定的一个新的模式x，根据训练集，推断它所对应的输出y是1还是-1。

确切地说，上述分类问题是分成两类的问题：两类分类问题；与此类似，还有分成多类的分类问题：多类分类问题。它们的不同之处仅在于前者的输出至取两个值，而后者则可取多个值，本论文中的分类问题默认为两类分类问题。

目前，具有代表性的分类技术主要有：决策树方法、贝叶斯分类、人工神经网络、遗传算法、基于案例的学习、粗糙集和模糊集方法等等。需要指出的是，任何一种分类方法都不是万能的。不同的具体问题，需要用不同的方法去解决。即使对于同一个问题，可能有多种分类算法。分类的效果一般和数据的特点有关，有些数据噪声大、有缺值、分布稀疏，有些属性是离散的而有些是连续值的，目前普遍认为不存在某种方法适合所有的数据。因此，对于一个特定问题和一类特定数据，需要评估具体算法的适应性。下面对几种主要的分类方法作简要地介绍。

1.2.2 决策树学习

决策树学习是经典的归纳学习算法，已经被成功地应用到很多领域如医疗诊断，贷款申请的信用风险评估等[Mitchell 97]。构造一个决策树分类器通常分为两步：树的生成和剪枝。树的生成采用自顶向下分而治之的方法：开始时所有记录都在根节点，然后基于一个启发式规则或一种统计度量选择一个最佳属性，递归地进行分割，不断向下分支，直到同一节点上的样本点都属于一个类别或者没有属性可用于对数据进行分割，从而完成决策树的构造。从生成的决策树中可以提取多条 IF-THEN 规则来提高分类模型的可读性，从根节点到叶节点的一条路径对应着一条合取规则，整个决策树对应着一组析取表达式规则。

到目前为止决策树有很多实现算法。最早的决策树算法是由 Hunt 等人于 1966 年提出的概念学习系统 CLS [Hunt 66]，以后的许多决策树算法都是对 CLS 算法的改进或由 CLS 衍生而来。1986 年，Quinlan 提出了著名的 ID3 算法 [Quinlan 86]。在此基础上，他又提出了 C4.5 算法 [Quinlan 93]。ID3 和 C4.5 算法的核心都是在决策树的各级节点上进行属性选择，选择的依据分别是信息增益(Information Gain)和增益比率(Gain Ratio)。C5.0 是 C4.5 的进一步改进版本，它对大数据库的处理速度更快，并增加了 Boost 等技术，而

且进一步提高了规则的可理解性。另一方面，Breiman 提出了基于 Gini 系数选择属性的 CART 方法 [Breiman 84]，为了适应处理大规模数据集的需要，后来又有若干改进的算法被相继提出，其中 SLIQ [Mehta 96]和 SPRINT [Shafer 96]是两个比较有代表性的算法。

1.2.3 贝叶斯分类

贝叶斯分类属于统计学分类方法，它是一类利用概率统计知识进行分类的算法。设每个实例包括一个决策属性和 n 个特征属性$\{x_1,\ldots,x_n\}$。贝叶斯网络由一个表示类别变量的节点 C 和若干个表示特征的节点 X_i 组成。对一个未知类别的样本 X，需要计算出 X 属于每一个类别 C_k 的概率 $P(C_k|X)$，然后选择其中概率最大的类别作为其类别。根据贝叶斯定理，由于 P(X)对于所有类为常数，最大化后验概率 $P(C_k|X)$可转化为最大化先验概率 $P(X|C_k)P(C_k)$，而先验概率一般可从训练数据集中近似求得。

构造贝叶斯分类器有很多种，实验表明，特征变量与类别变量直接关联的方法分类性能较好。Naive Bayesian(NB)分类器就是其中之一，它简单、应用广泛。NB 对问题进行了简化，假定假设各属性的取值互相独立，即满足 $P(X|C)=\prod P(xi|C)$。实际问题的特征属性通常不具有这样的性质，但 NB 在某些情况下仍然具有很好的效果[Mitchell 97]。

为了提高分类的准确性，可以通过多种反映特征属性之间相关性的方式来提高 NB 分类器的性能，如允许特征属性节点除了类别属性节点外还有其它父节点。但即使限定其它的父节点至多两个，求最优的贝叶斯分类器仍然是 NP 问题 [Chickering 95]。在最坏的情形，任何构造最优分类器的算法的复杂性都是特征数的指数次方。为此，[Friedman 96]中提出了 Tree Augmented Naive-Bayes(TAN)算法，它限定每个特征属性除了分类属性外至多有一个其它父节点，算法的复杂性为特征数的平方。

1.2.4 人工神经网络

神经网络是一种很好的函数逼近工具，在过去十几年里取得了飞速的发展，发展出了很多的模型及其改进，例如 BP, Hopfield, Kohonen, ART, RNN, KBANN, RBF 等等。神经网络是由大量处理单元组成的非线性自适应动力系统，具有学习能力、记忆能力、计算能力以及智能处理功能，并在不同程度和层次上模仿人脑神经系统的信息处理、存储及检索功能 [史 06]。典型的神经网络模型主要分三大类：以感知机、BP 模型等为代表的，用于分类、预测和模式识别的前馈式神经网络模型；以 Hopfield 的离散模型和连续模型为代表的，分别用于联想记忆和优化计算的反馈式神经网络模型；以 ART 模型、Kohonen 模型为代表的，用于聚类的自组织映射方法。神经网络方法的缺点是"黑箱"性，人们难以理解网络的学习和决策过程。

神经网络分类算法的重点是构造阈值逻辑单元，每个阈值逻辑单元是一个对象，它可以接受一组输入$\{x_1,x_2,\ldots,x_n\}$以及与其对应的权值$\{w_1,w_2,\ldots,w_n\}$，然后进行加权求和，计算 $x_1w_1+x_2w_2+\ldots+x_nw_n$，如果这个和达到或者超过了预先设定的阈值 w_0，则根据转移函数产生一个输出量。神经网络是基于经验风险最小化原则的学习算法，有一些固有的缺陷，比如层数和神经元个数难以确定，容易陷入局部极小，还有过学习现象，这些本

身的缺陷在 SVM 算法中可以得到很好的解决。

1.2.5 关联规则分类

关联规则挖掘是数据挖掘研究中一个重要的、高度活跃的领域。近年来，将关联规则挖掘用于分类问题，取得了很好的效果。Liu Bing 提出了著名的 CBA(Classification Based on Association)算法 [LiuB 98]，该算法把分类规则挖掘与关联规则挖掘整合在一起。CBA 算法分两个步骤构造分类器：第一步，发现所有形如 $x_i \wedge x_j => C_k$ 的关联规则，即右部为类别属性值的类别关联规则(Classification Association Rules，CAR)；第二步，从已发现的 CARs 中选择高优先度的规则来覆盖训练集，也就是说，如果有多条关联规则的左部相同，而右部为不同的类，则选择具有最高置信度的规则作为可能的规则。CBA 算法的优点是其分类准确度较高，在许多数据集上比 C4.5 更精确。此外，上述两步都具有线性可伸缩性。

1.2.6 支持向量机

Vapnik 等人从六、七十年代开始致力于统计学习理论方面研究，到九十年代中期，随着其理论的不断发展和成熟，也由于神经网络等学习方法在理论上缺乏实质性进展，统计学习理论开始受到越来越广泛的重视。统计学习理论是一种小样本统计理论，着重研究在小样本情况下的统计学习规律及学习方法性质。该理论针对小样本统计问题建立了一套新的理论体系，在这种体系下的统计推理规则不仅考虑了对渐近性能的要求，而且追求如何在现有的有限信息条件下得到最优的结果。SLT 为解决有限样本学习问题提供了一个统一的框架。它能将很多现有方法纳入其中，并对可学习性、正确性、过学习和欠学习、局部极小点等问题取得了较好的结果。同时，在 SLT 的基础上发展了一种新的通用学习算法——支持向量机(Support Vector Machine: SVM) [Vapnik 95]。通过学习算法，SVM 可以自动寻找出那些对分类有较好区分能力的支持向量，由此构造出的分类器可以使类与类之间的间隔最大化，因而有较好的适应能力和较高的分类准确率。

SVM 的基本思想是通过某种事先选择的非线性映射将输入向量映射到一个高维特征空间，在这个空间中构造最优分类超平面。在高维特征空间中构造最优超平面，只需用到特征向量之间的内积计算，而我们可以使用核函数来计算原空间中的向量在特征空间中的点积，通过选用不同的核函数，可以构造输入空间中不同类型的非线性决策面的学习机，同时也克服了维数困难 [Vapnik 95]。

支持向量机算法的目的在于寻找一个超平面 H(d)，该超平面可以将训练集中的数据分开，且使两类边界沿垂直于该超平面方向的距离最大，故 SVM 法亦被称为最大边缘(Maximum Margin)算法。训练样本集中的大部分样本不是支持向量，移去或者减少这些样本对分类结果没有影响。SVM 算法对小样本、非线性和高维数据具有很好分类性能。

1.2.7 分类算法的评价标准

分类算法可以参考以下标准进行评价和比较：

- 预测的准确率：即模型正确地预测新的或先前未见过的样本类别的能力；
- 计算速度：分类的时间包括构造模型和使用模型进行预测的时间；
- 强壮性：即正确预测含有噪声和空缺值的数据集的能力；
- 可伸缩性：即对海量数据集进行有效构造模型的能力；
- 模型描述的简洁性和可解释性：模型描述愈简洁、愈容易理解，愈受欢迎。

为了提高分类的准确性、有效性和可伸缩性，在进行分类之前，通常要对数据进行预处理，如数据清理、相关性分析、数据变换等。需要指出的是，不存在适合所有数据的通用分类算法，在实际操作中，需要根据数据本身的特点评估算法的适应性，从而选出最合适的算法。

1.3 本文的贡献

本文的所介绍的研究成果包括支持向量机（SVM）的增量学习技术、样本选择技术及新的 SVM 分类器模型：设计了非线性临近支持向量机(Proximal Support Vector Machine PSVM）的增量学习算法；为能够处理大数据集的 PSVM 在线增量学习问题 ，设计了针对（非）线性 PSVM 的样本选择技术；最后提出了一种新的支持向量机分类器——Extreme SVM（ESVM），与标准 SVM 相比 ESVM 具有速度快、扩展性能好等优点。

1．非线性 PSVM 的增量学习算法

目前关于标准 SVM 的增量学习算法已经有很多的研究成果，不过根本的出发点是一致的，即利用支持向量具有代表整个训练数据集的能力这一性质，在增量学习过程中仅保留历史数据中的支持向量，然后利用这些支持向量对应的样本点同新样本一起训练。对于 Linear PSVM 有更好的结果，利用其解的特殊形式，可以随意地向数据集中添加、删除样本，同时有效地得到更新后的结果[Fung 01b]。而且线性 PSVM 的求解复杂度独立于数据集中样本点的个数，只跟输入空间的维度有关，通常样本点的维度要比样本点的个数小的多。

但是这些结果却不适用于非线性 PSVM (Nonlinear PSVM - NPSVM)。首先 PSVM 与标准支持向量机具有不同的几何意义，在 PSVM 中并不存在传统意义上的支持向量，因此传统 SVM 的增量学习算法对非线性 PSVM 并不适用，其次 NPSVM [Fung 01a]的解与线性 PSVM[Fung 01a]具有不同的形式，线性 PSVM 的增量算法也不适用。

为了使 NPSVM 能够更加快速的进行在线增量学习，本文基于一个新的非线性 PSVM 模型设计了一种新的增量学习算法。该算法从新模型解的形式出发，利用分块矩阵求逆公式，有效地利用 NPSVM 分类器的历史训练结果，减少了在线学习过程中的重复学习，完成增量学习过程。理论推导及实验结果显示在线学习过程中采用该算法不仅可以得到与批量学习相同的分类器、正确率；而且解决了重复学习的问题，可以显著地缩短训练时间。

2．PSVM 的样本选择技术

上述增量学习算法的空间复杂度是样本点个数的平方：$O\left(m^2\right)$，为了使对大容量数据集的分类器增量学习成为可能，本文针对 PSVM 设计了样本选择技术。标准支持向量机中的样本选择很简单：只需保留支持向量就可以了。但 PSVM 具有与标准 SVM 不同的优化问题表示形式、不同的几何意义，需要对其设计样本选择技术。

本文通过分析 PSVM 解的结构及不同位置的样本点对最终解的影响，来设计样本选择技术。我们发现与训练得到的拟合超平面距离较近的样本点具有较强的表达能力，因此可以选择出与拟合超平面距离最近一些点代表历史数据集。此外样本选择也应该存在于在线学习时添加新样本的过程中：对于新的数据集我们期望选择出对历史分类器最有价值的样本点，即只保留那些可能更新模型的样本点，去掉那些已被模型描述了的点。这可以通过保留距离拟合超平面最远一些点来得到。在非线性的情形下，为了得到更好的效果，我们同时把被新分类器错分的样本点添加到新数据集中。

实验显示上述样本选择技术仅需付出较小时间代价，就可以有效的处理大样本集的在线学习问题，而且可以得到较好的训练结果。

3．一种新的非线性 SVM 学习方法

ELM[Huang 04]是一种单隐层神经网络（SFLNs）的学习算法，通过随机选取输入权重实现了神经网络的快速学习。但它仍然是一种经验风险最小化框架下的学习算法，容易出现过学习的现象。注意到可以把 ELM 学习 SLFNs 输出权重的过程分为两步：1）隐层神经元将输入样本点映射到隐层输出向量；2）利用隐层输出向量作为训练集求解输出权重。这就与 SVM 训练非线性分类器的思想有些类似：通过核函数在某特征空间中训练线性分类器。本文将 SVM 与 SFLNs 的学习算法结合起来设计了一种新的非线性 SVM 模型——ESVM。

ESVM 是一种基于正则化最小二乘法的新的 SVM 分类器。与其他所有非线性 SVM 学习方法不同的是，ESVM 不是使用核函数来得到非线性分类器，而是显式地构造了一个非线性的随机映射函数将输入样本点映射到一个特征空间中，然后在该特征空间中学习一个线性的分类器。该方法基于单隐层前馈神经网络（Single hidden Layer Feedforward Networks - SLFNs）的学习机制，在保持 SLFNs 学习能力的前提下，其输入权重可以随机地确定而不需要训练，这样 SLFNs 隐层神经元的作用相当于一个映射函数。理论分析及实验结果表明: ESVM 可以有效地应用于大数据集的训练，不仅具有与 SVM 相当的正确率，而且极大地缩短了训练时间。另外，与 SLFNs 的学习算法 ELM 相比，ESVM 将正则化理论引入到 SLFNs 的训练中，具有比 ELM 更好的泛化能力。

1.4 本文的组织

本文从数据挖掘技术的出现出发，在本章介绍了数据挖掘技术的分类，典型的数据挖掘算法尤其是分类学习算法。

由于本文的主要工作是基于 SVM 的，所以在第二章中我们将对经典统计学习理论的支持向量机技术，PSVM 技术，以及它们的进展、存在的问题进行详细的介绍。

第三章提出了一种非线性 PSVM 的增量学习算法，同时给出了实验数据说明该算法的有效性。

第四章设计了一种在线学习过程中的 PSVM 样本选择技术，并且给出了验证其有效性的实验数据。

第五章给出了一种新的非线性 SVM 模型：Extreme SVM – ESVM。并将其与其它流行的学习算法进行理论比较分析，同时给出了测试实验数据，说明了与其它算法相比其具有良好的性能。

最后，对整个论文进行了总结，并给出了后续可能的研究方向。

第二章 理论基础

本章首先给出了分类问题的表示方法及分类器学习理论；然后重点描述了支持向量机学习算法；最后对与本文相关的支持向量机模型新进展做了简单地介绍。

2.1 分类问题的表示

考虑 n 维空间中的分类问题，已知 l 个样本点，每个样本点包含 n 个属性指标（即 $x \in R^n$ ），记 l 个样本点的集合为：

$$T = \{(x_1, y_1), ...(x_l, y_l)\} \in (X \times Y)^l . \tag{2.1}$$

其中 $x_i \in X = R^n$ 是输入向量，$y_i \in Y = \{-1,1\}$ 是输出向量，$i = 1,...l$ 。这 l 个样本点组成的集合称为训练集，此时分类问题就是，对任意给定的一个新的样本点 x ，根据训练集，推断它所对应的输出 y 是1还是 -1 。

分类学习算法的目的是根据给定的训练集对某系统输入输出之间的依赖关系求近似估计，使该估计能够对未知输入做出尽可能正确的预测。可以一般地表示为：变量 y 与 x 遵循某一未知的联合概率 $F(x, y)$ ，分类学习问题就是根据 l 个独立同分布观测样本：

$$(x_1, y_1), (x_2, y_2)...(x_l, y_l) . \tag{2.2}$$

在一组函数 $\{f(x, w)\}$ 中求一个最优的函数 $f(x, w_0)$ 来对依赖关系进行估计，使期望风险：

$$R(w) = \int L(y, f(x, w)) \mathrm{d}F(x, y) . \tag{2.3}$$

最小。其中，$\{f(x, w)\}$ 称为预测函数集，w 为函数的广义参数，$\{f(x, w)\}$ 可以表示任何函数集；$L(y, f(x, w))$ 为由于用 $f(x, w)$ 对 y 进行预测而造成的损失，不同类型的学习问题（如：模式识别，函数逼近或者概率密度估计）有不同形式的损失函数。

分类问题包括两个阶段：训练和预测。在训练阶段，分类算法利用具有类别标记的训练实例学到一个分类模型（称作分类器），该模型把训练实例映射到给定的类别；在预测阶段，利用该分类器对没有类别标记的（测试）实例预测其类别，预测的准确程度可以评价分类器的性能。

2.2 经验风险最小化

在上面的分类问题表达中，学习算法的目标是使期望风险（2.3）最小化，但是可以利用的信息只有有限个数的训练样本（2.2），这使得期望风险无法计算。传统的学习方法普遍采用经验风险最小化（Emprical Risk Minimization - ERM）准则，即定义如下的经验风险公式：

$$R_{emp}(w)=\frac{1}{2}\sum_{i=1}^{n}L\left(y_i, f(x_i,w)\right). \tag{2.4}$$

作为对期望风险（2.3）的估计，然后设计使经验风险（2.4）最小化的学习算法。但事实上，用ERM准则代替期望风险最小化并没有经过充分的理论论证，只是直观上想当然可能合理的做法，而实际上即使可以假定当样本数趋向于无穷大时经验风险（2.4）能够趋近于期望风险（2.3），很多问题中的样本数目离无穷大也相去甚远，那么在有限样本条件下ERM准则得到的结果能使真实风险也较小吗？

2.3 复杂性与推广能力

ERM准则不成功的一个例子是神经网络过学习问题。开始，很多注意力都集中在如何使（2.4）式中的$R_{emp}(w)$更小，但人们很快发现：极小化训练误差并不一定能导致好的预测效果。某些情况下，训练误差过小反而会导致推广能力的下降，即真实风险的增加，这就是过学习问题。之所以出现过学习现象，一是因为样本不充分，二是学习机器设计不合理。即，试图用一个十分复杂的模型去拟合有限的样本，导致丧失推广能力。学习机器的复杂性与推广性之间的这种矛盾同样可以在其他学习方法中看到。

由此可看出，有限样本情况下，经验风险最小并不一定意味着期望风险最小；学习机器的复杂性不但应于所研究的系统有关，而且要和有限数目的样本相适应。统计学习理论就是一种指导我们在小样本情况下建立有效的学习和推广方法的理论。

2.4 统计学习理论— 支持向量机的理论背景

SVM之前的机器学习方法的重要理论基础之一是统计学，传统统计学研究的是样本数目趋于无穷大时的渐进理论，现有学习方法也多是基于此假设。但在实际问题中，样本数目往往是有限的，因此一些理论上功能很优秀的学习方法实际中表现却可能不尽如人意。与传统统计学相比，统计学习理论[Vapnik 98], [Vapnik 95])（Statistical Learning Theory - SLT）是一种专门研究小样本情况下机器学习规律的理论。V.Vapnik 等人从六、七十年代开始致力于此方面的研究，到九十年代中期，随着其理论的不断发展和成熟，也由于神经网络等学习方法在理论上缺乏实质性进展，统计学习理论([Burges 98], [Vapnik 95])开始受到越来越广泛的重视。

统计学习理论是建立在一套较坚实的理论基础之上的，为解决有限样本学习问题提出

了一个统一的框架。它能将很多现有方法纳入其中，可以解决很多以前难以解决的问题（比如神经网络结构选择问题、局部极小点问题等）；同时，在这一理论基础上发展了一种新的通用学习方法——支持向量机（SVM）。

统计学习理论主要包括四个方面：

1. 经验风险最小化准则下统计学习一致性的条件；
2. 在这些条件下关于统计学习方法推广性界的结论；
3. 在这些界的基础上建立的小样本归纳推理准则；
4. 实现新准则的实际方法；

其中最有指导性的结论是推广性的界，与此相关的一个核心概念是VC维。

2.4.1 函数集的 VC 维及推广性

为了研究学习过程一致收敛的速度和推广性，统计学习理论定义了一系列有关函数集学习性能的指标，其中最重要的是VC维（Vapnik-Chervonenkis Dimension）：假如存在一个包含 h 个样本的样本集能够被一个函数集 $f(x,w)$ 中的函数按照所有可能的 2^n 种形式分为两类，则称该函数集能把样本数为 h 的样本集打散。VC维就是用这个函数集中的函数所能够打散的最大样本集中所包含的样本数目。VC维反映了函数集的学习能力，VC维越大则学习器越复杂。但目前尚没有通用的关于任意函数集VC维计算的理论，只对一些特殊的函数集知道其VC维：比如在 n 维实数空间中线形分类器和线形实函数的VC维是 $n+1$。对于一些较复杂的学习机器（如神经网络），其VC维除了与函数集（网络结构）有关外，还受学习算法等的影响，其确定更加困难。

统计学习理论系统地研究了各种类型的函数集经验风险和实际风险之间的关系，即推广性的界 [Vapnik 95]，关于两类问题结论是：对函数集中的所有函数，经验风险 $R_{emp}(w)$ 和实际期望风险 $R(w)$ 之间至少以 $1-\eta$ 的概率满足如下关系：

$$R(w) \le R_{emp}(w) + \sqrt{\frac{h\left(\ln(2n/h)+1\right)-\ln(\eta/4)}{2}},. \qquad (2.5)$$

其中 h 是函数集的ＶＣ维，n 是样本数。这一结论从理论上说明了学习机器的实际风险是由两部分组成的：一是经验风险（训练误差），另一部分称作置信范围，它和学习机器的ＶＣ维及训练样本数有关。可以简单地标示为：

$$R(w) \le R_{emp}(w) + \phi(h/n),. \qquad (2.6)$$

它表明，在有限训练样本下，学习机器的ＶＣ维越高（复杂性越高）则置信范围越大，导致实际风险与经验风险之间可能的差别越大。这就是为什么会出现过学习现象的原因。机器学习过程不但要使经验风险最小，还要使ＶＣ维尽量小以缩小置信范围，才能取得较小的实际风险，即对未来样本有较好的推广性。

需要指出，推广性的界是对于最坏情况的结论，在很多情况下是较松的，尤其当ＶＣ维较高时更是如此（Vapnik指出当$h/n>0.37$时这个界肯定是松弛的，当ＶＣ维无穷大时这个界就不再成立）。而且这个界仅指在对同一类学习函数进行比较时有效，可以指导从函数集中选择最优的函数，而在不同函数集之间比较却不一定成立，Vapnik指出寻找更好的反映学习机器能力的参数和得到更紧的界是学习理论今后的研究方向之一[Vapnik 95]。

2.4.2 结构风险最小化

从上面的结论看到，ERM准则当样本数目非常有限时是不合理的，我们需要同时最小化经验风险和置信范围。其实，在传统方法中，选择学习模型和选参的过程就是调整置信范围的过程，如果模型比较适合现有的训练样本，则可以取得较好的效果。但因为缺乏理论指导，这些方法只能依赖先验知识和经验，造成了如神经网络等方法对使用者“技巧”的过分依赖。

统计学习理论提出了一种新的策略，即把函数集构造成一个函数子集序列，使各个子集按照VC维的大小排列；在每个子集中寻找最小经验风险，在子集间折中考虑经验风险和置信范围，以取得实际风险的最小，这种思想称作结构风险最小化（SRM：Structural Risk Minimization）准则.统计学习理论还给出了合理的函数子集结构应满足的条件及在SRM准则下实际风险收敛的性质[Vapnik 95]。

实现SRM准则可以有两种思路，一是在每个子集中求最小经验风险，然后选择使最小经验风险和置信范围之和最小的子集，这种方法比较费时，当子集数目很大甚至无穷时是不可行。因此有第二种思路，即设计函数集的某种结构使每个子集中都能取得最小的经验风险（如使训练误差为 0 ），然后只需选择适当的子集使置信范围最小，则这个子集中使经验风险最小的函数就是最优函数。支持向量机实际上就是这种思想的具体实现。[Vapnik 95]中讨论了一些函数子集结构的例子和如何根据SRM准则对某些传统方法进行改进的问题。

2.5 支持向量机

SVM 是从线性可分情况下的最优划分超平面发展而来的：假设有m个样本训练点$(x_1,y_1),(x_2,y_2)...(x_m,y_m)\in R^n\times\{\pm1\}$，学习得到线性分类超平面：$x'w-r=0$，将样本分成两类：

$$x'w-r\begin{cases}>0 \text{ then } y=1\\<0 \text{ then } y=-1\end{cases}. \tag{2.7}$$

显然，超平面中的w和r乘以某一系数后仍能满足方程，不失一般性，对于所有的样本x_i，适当调整w和r，使$\|x'w-r\|$的最小值为 1。经过归一化处理得到：

$$x_i'w-r\begin{cases}\geq 1 \text{ 如果 } y_i=1\\ \leq -1 \text{ 如果 } y_i=-1\end{cases}. \tag{2.8}$$

或者等价地写成：

$$d_i\left(x_i'w-r\right)-1\geq 0,\ i=1,2,....m. \tag{2.9}$$

其中$d_i=\pm 1$表示样本x_i的类别，$x'w-r=\pm 1$为过正负类中距离分类超平面$x'w-r=0$最近的样本点且与之平行的超平面，它们之间的距离叫做分类间隔（margin）。所谓最优分类面就是要求划分超平面不但能将两类正确分开，而且使分类间隔最大。在归一化的条件下样本与超平面的最小距离为$\frac{|x'w-r|}{\|w\|}=\frac{1}{\|w\|}$，两类样本到超平面最小距离之和即分类间隔为$\frac{2}{\|w\|}$，使间隔最大等价于使$\frac{1}{2}\|w\|^2$最小。则我们可以通过求解如下二次优化问题得到目标解：

$$\begin{aligned}&\min_{(w,r,y)\in R^{m+1+m}}\ \frac{1}{2}w'w\\ &s.t.\ D(Aw-er)\geq e,\end{aligned}. \tag{2.10}$$

即满足条件（2.9）且使$\frac{1}{2}\|w\|^2$最小的划分超平面叫做最优分类面，$x'w-r=\pm 1$上的训练样本点就称作支持向量。

使分类间隔最大实际上就是对推广能力的控制，这是SVM的核心思想之一。统计学习理论指出，在N维空间中，设样本分布在一个半径为R的超球范围内，则满足条件$\|w\|\leq A$的正则超平面构成的指示函数集$f\left(x,w,r\right)=\mathrm{sgn}\left(x'w-r\right)$的VC维满足下面的界：

$$h\leq \min\left(\left[R^2A^2\right],N\right)+1. \tag{2.11}$$

因此使$\frac{1}{2}\|w\|^2$最小就是使VC维的上界最小，从而实现SRM准则中对函数复杂性的选择。

利用Lagrange优化方法可以把上述最优分类问题转化为其对偶问题，其Lagrange函数如下：

$$L\left(w,r,\alpha\right)=\frac{1}{2}w'w-\alpha'\left[D\left(Aw-er\right)-e\right]. \tag{2.12}$$

可得如下的KKT充分必要条件：

$$\begin{cases} \dfrac{\delta L}{\delta w} \Rightarrow w - A'D\alpha = 0 \\ \dfrac{\delta L}{\delta r} \Rightarrow \alpha De = 0 \\ \alpha\left[D\left(Aw - er\right) - e \right] = 0 \\ D\left(Aw - er\right) - e \geq 0, \alpha \geq 0 \end{cases} . \tag{2.13}$$

将（2.13）中的等式带入（2.10）得到其对偶问题：

$$\begin{aligned} \min_{\alpha} \quad & \frac{\alpha' DAA' D\alpha}{2} - \alpha' e \\ \text{s.t.} \quad & \alpha De = 0 \\ & \alpha \geq 0, \end{aligned} . \tag{2.14}$$

其中α为Langrangian乘子，由上述问题求得α的最优解，可得如下SVM分类超平面：

$$f\left(x\right) = \text{sgn}\left(\left(x \bullet A'\right) D\alpha - r\right). \tag{2.15}$$

对偶问题的最优解$\alpha = \left(\alpha_1, \alpha_2, ..., \alpha_m\right)$应使得每个样本$A_i$满足优化问题的KKT 条件：

$$\begin{aligned} & \alpha_i = 0 \Rightarrow y_i f\left(X_i\right) \geq 1 \\ & 0 < \alpha_i < v \Rightarrow y_i f\left(X_i\right) = 1. \\ & \alpha_i = v \Rightarrow y_i f\left(X_i\right) \leq 1 \end{aligned} \tag{2.16}$$

其中非零α_i对应的样本点为支持向量。

在线性不可分的情况下，可在条件（2.9）中增加一个松弛项$y_i \geq 0$：

$$d_i\left(x_i' w - r\right) - 1 + y_i \geq 0, \ i = 1, 2, ...m . \tag{2.17}$$

将二次优化改为：

$$\begin{aligned} \min_{(w,r,y) \in R^{m+1+m}} \quad & ve' y + \frac{1}{2} w' w \\ s.t. \ & D(Aw - er) + y \geq e . \\ & y \geq 0, \end{aligned} \tag{2.18}$$

即折中考虑最少错分样本和最大分类间隔，就得到广义最优分类面。其中v是一个常数，它控制对错分样本惩罚的程度。类似的，有如下Lagrange函数：

$$L\left(w, r, y, \alpha, \beta\right) = ve' y + \frac{1}{2} w' w - \alpha'\left[D\left(Aw - er\right) + y - e \right] - \beta' y . \tag{2.19}$$

对w, r, y分别求偏导得到如下ＫＫＴ条件：

$$\begin{cases} \dfrac{\delta L}{\delta y} \Rightarrow ve-\alpha-\beta=0 \\ \dfrac{\delta L}{\delta w} \Rightarrow w-A'D\alpha=0 \\ \dfrac{\delta L}{\delta r} \Rightarrow \alpha De=0 \\ \alpha\left[D\left(Aw-er\right)+y-e\right]=0, D\left(Aw-er\right)+y-e\geq 0 \\ \beta y=0, y\geq 0 \\ \alpha\geq 0, \beta\geq 0 \end{cases} \tag{2.20}$$

将其中关于w,r,y的等式带入Lagrange函数得到如下对偶问题：

$$\begin{aligned} \min_{\alpha} \quad & \frac{\alpha'DAA'D\alpha}{2}-\alpha'e \\ \text{s.t.} \quad & \alpha De=0 \\ & 0\leq\alpha\leq ve, \end{aligned} \quad . \tag{2.21}$$

α为Langrangian乘子，有上述问题求得α的最优解，可得如下SVM分类超平面：

$$f\left(x\right)=\operatorname{sgn}\left(\left(x\bullet A'\right)D\alpha-r\right). \tag{2.22}$$

对偶问题的最优解$\alpha=\left(\alpha_1,\alpha_2,...,\alpha_m\right)$应使得每个样本$A_i$满足优化问题的KKT 条件（2.20）其中非零$\alpha_i$对应的样本点为支持向量。

对于N维空间中的线性函数，其VC维为$N+1$，但根据（2.11）的结论，在$\|w\|\leq A$的约束下其VC维可能大大减小，即使在十分高维的空间中也可以得到较小VC维的函数集，以保证有较好的推广性，同时我们看到，通过把原问题转化为对偶问题，计算的复杂度不再取决于空间维数，而是取决于样本数，尤其是样本中的支持向量数。这些特点使有效的对付高维问题成为可能。

对非线性问题，可以通过非线性变换转化为某个高维空间中的线性问题，在变换空间中求最优分类面。这种变换可能比较复杂，这种思路在一般情况下不易实现，但是注意到，在上面的对偶问题中，不论是优化函数还是分类函数都只涉及到训练样本之间的内积运算，这样在高维空间实际上只需进行内积运算，而这种内积运算是可以用原空间中的函数实现的，我们甚至没有必要知道变换的形势。根据范函的有关理论，只要一种核函数$K\left(x_i,x_j\right)$满足Mercer条件，它就对应某一变换空间中的内积。因此在最优分类面中采用适当的内积函数$K\left(x_i,x_j\right)$就可以实现某一非线性变换后的线性分类器，而计算复杂度却没有增加。此时目标函数(2.21)变为：

$$\begin{aligned} \min_{\alpha} \quad & \frac{\alpha' DK\left(A,A'\right)D\alpha}{2} - \alpha' e \\ \text{s.t.} \quad & \alpha De = 0 \\ & 0 \le \alpha \le ve, \end{aligned} \quad . \tag{2.23}$$

而相应的分类函数也变为：

$$f\left(x\right) = \text{sgn}\left(K\left(x,A'\right)D\alpha - r\right). \tag{2.24}$$

概括地说，支持向量机就是首先通过用内积函数定义的非线性变换将输入空间变换到一个高维空间，在这个空间中求（广义）最优分类面。

2.6 支持向量机求解算法

从上一节我们可以看到，支持向量机已经把分类问题归结为一个约束最优化问题：一个带有线性约束的凸二次规划问题。

2.6.1 化为无约束问题

求解约束问题的途径之一，是把它转化为一个或一系列无约束问题，对支持向量机中的凸二次规划问题也可以采用这一途径，具体可见（SSVM [Lee 99]），然后我们就可以利用基本的无约束问题解法对其进行求解。典型的有最速下降法，但最速下降法的效果一般来说是很不理想的，因为某点x_k处的负梯度方向$-\nabla f\left(x_k\right)$仅仅在$x_k$附近局部来看是最速下降的，从全局来看并不一定是最优的路径，比如可能导致整个行进路径成锯齿形。此外Newton法也是一种求解无约束问题的解法，它利用解的一阶必要条件$\nabla f\left(x\right)=0$导出一个从任意点x_1出发的迭代公式。当目标函数是正定二次函数时，Newton法仅需一次迭代就能到达极小点x^*，或者目标函数满足一定条件而且初始点x_1充分接近极小点x^*时Newton法会以很小的收敛速率收敛到极小点，这是Newton法的最大优点。但当初始点x_1距极小点x^*较远时，所产生的点列可能收敛不到x^*，甚至算法的某次迭代反而使目标函数值增大，此外Newton法的计算量也比较大。为此有很多对Newton法的改进：变度量法（BFGS）、共轭梯度法、Newton-PCG算法等等。

2.6.2 内点算法

对线性规划问题和凸二次规划问题来说，内点算法主要是原仿射尺度法和原-对偶算法。原仿射尺度法的基本策略是：取一个内点解，对解空间作一适当的变换，使现行解置于变换空间的中点，然后沿着变换后约束的零空间中的最速下降方向移动，为了保持其为

内点解，始终不能移到非负象限的面上；然后再取逆变换，将改进解移回原来的解空间作为一个新的内点解。重复以上过程直到最优性条件或其他停机准则得到满足。虽然原仿射尺度法用得很好，但是不能证明它是多项式时间算法，为此人们又提出了一种具有多项式时间的原仿射尺度法——带有壁垒函数的原仿射尺度法。原对偶算法和带有壁垒函数的原仿射尺度法有密切的关系：它们的搜索方向是两条不同的，但是等价的通向满足KKT条件解的代数路径上的Newton方向。

2.6.3 求解大型问题的算法

上面介绍的两种类型算法，原则上都可以直接应用于求解支持向量机中的凸二次规划问题，但这些算法都要存储与训练集相应的核矩阵，而储存核矩阵所需的内存是随着训练机中样本点的个数m的平方增长的。另外这些算法往往包含大量的矩阵运算，所需的运算时间往往过长。上述事实迫使人们涉及专门针对支持向量机的新算法，而支持向量机中的最优化问题具有一些非常好的特性，如解的稀疏性和最优化问题的凸性等，这些性质使得构造快速的专用算法成为可能。专用算法的一个共同特点是：将大规模的原问题分解成若干小规模的子问题，按照某种迭代策略，反复求解子问题，构造出原问题的近似解，并使该近似解逐渐收敛到原问题的最优解。由于子问题的选区和迭代策略的不同，可以有不同的算法，如选块算法(Chunking)[Vapnik 98]、分解算法(Decomposing)（[Chang 04]、[Osuna 97]、[Platt 99]、[Vishwanathan 03]）和序列最小优化算法(Sequential Minimal Optimization - SMO) [Platt 99]。

由于支持向量机的解仅依赖于支持向量，因此如果我们事先知道哪些是支持向量对应的样本点，就可以仅保留它们像应的样本点，而从训练集中删去其他样本点。对缩小后的训练集进行学习可以得到同样的决策函数，显然这一事实对于大规模的实际问题非常重要，因为它往往是稀疏的，只有支持向量不是很多，就有可能只需求解规模不是很大的优化问题，然而我们事先并不知道哪些样本点是支持向量哪些不是，一般来说需要采用启发式方法寻找。最简单的启发式方法是选块算法。

2.6.4 选块算法(Chunking)

我们称训练集T中的任意一个子集为“块”。该方法从训练集的任意一个子集或者“块”出发，在该块上应用标准的优化算法求解得到Lagrange乘子：$\alpha=(\alpha_1,\alpha_2...,\alpha_m)'$，然后用下列方式调整当前块为新块：（1）保留该块中与非零α_j对应的训练点，舍去块中的其他训练点；（2）利用得到的决策函数来检测训练集中除去该块后的所有训练点，把其中M个违背KKT条件最严重的点加到新块中去，再在新块上求解对偶问题，重复以上过程，其中每一个新的子问题的α初始值都取为前一个子问题的解，最后直到满足某一个停机准则为止。这种方法的优点是当支持向量数很少时，能大大提高运算速度，然而，如果支持向量个数本身就比较多，随着算法迭代次数的增多，所选的块也会越来越大，算法就变得十分

缓慢了。

2.6.5 分解算法（Decomposing）

分解算法与选块算法的不同之处在于他每次只更新一定数量的Lagrange乘子，而其他的乘子保持不变，因此每次把一个样本点加入到工作集中去，就必须舍去另外一个样本点，迭代过程只是将工作集之外的样本点中的一部分“情况最糟的样本点”与工作集中的一部分样本点进行等量交换。即使支持向量的个数超过工作集的大小，也不改变工作集的规模，即与选块算法不同，该算法的目的不是找出所有的支持向量，从而在相应的约束上解决问题，而是每次只针对很小的训练子集来求解。

2.6.6 序列最小最优化（Sequential Minimal Optimization, SMO）算法

序列最小最优化算法是分解算法中，将工作集大小设置为2时的特殊情形，即每次迭代过程中只调整相应于两个样本点(x_i, y_i)和(x_j, y_j)的α_i和α_j，它只求解一个具有两个变量的最优化问题。实际上，这使得工作集的规模已经建到最小，原因是在对偶问题中存在一个等式约束：$\alpha De = 0$。只要变动一个乘子α_i，就至少必须同时调整另一个乘子来保证不违反该约束。在工作集规模为2时，两个乘子存在解析解不需要迭代的求解二次规划问题，因此每一步执行的速度会很快，但是子问题的规模和整个算法需要迭代的次数是一对矛盾，SMO将工作集的规模建到最小，一个直接的后果就是迭代次数的增加，尽管如此，与其它算法相比该算法常表现出整体的快速收敛性质。另外，该算法还具有其它的一些重要性质：如不需要把核矩阵存储在内存中，没有矩阵运算，容易实现等等。

虽然Chunking、Decomposing和SMO能够解决大样本的训练问题，但当样本数目很大时迭代时间也将很长。人们发现可以利用支持向量的特殊性质，通过增量学习的途径也可以解决大样本的学习问题。

2.7 增量支持向量机

增量学习技术作为一种智能知识发现技术，已经得到了广泛的研究。它与传统的学习技术相比，优越性在于它不仅可以舍弃无用样本并减小训练集，而且可以充分利用学习的历史结果，使学习具有了延续性，可以用于对海量数据集的学习也可以用于实时地再现学习，很多学者基于传统的学习方法提出了新的增量学习算法([Zhang 99], [Zhang 95])。但是由于传统学习算法不能保证很好的泛化能力，常常陷于对问题的过学习和局部最小等现象，因而基于传统学习方法的增量学习算法通常得不到原问题较好的结果。

支持向量机（Support Vector Machine）是一种新的机器学习技术，基于统计学习理论的坚实基础，有着很强的学习能力和较好的泛化性能；学习采用优化方法得到的结果是全局最优解，不会产生传统方法中的过学习和局部最小等问题。而且其学习结果可以用支持向量集来表示，通常支持向量集是学习样本集的一小部分，但它却充分体现了整个样本集的模型属性。因此利用支持向量集的这种性质人们设计了很多种增量学习算法。

从（2.21）可以看出$w=\sum_{i=1}^{n}\alpha_i d_i \phi(x_i)$，一般情况下在此$w$的展开式中，大多数系数$\alpha_i$为零值，因此对$w$的值并没有影响，也不会影响分类的结果。而对$w$的确定有贡献的仅仅是非零值的$\alpha_i$所对应的$x_i$，也就是“SV”（Support Vector 支持向量）。因此SV集充分描述了模型的特征，对SV集的训练等价于对整个数据集的训练，在大多数情况下训练集中SV的数量只占训练样本集的很少一部分，因此可以使用SV集取代训练样本集进行分类学习，使得在不影响分类精度的同时极大地减少训练时间。

Syed等人在[Syed 99]中就利用SV集的这种性质提出了一种支持向量机的增量学习算法：首先把一个比较大的数据集分割为很多小子块，然后依次对每一个子块进行训练，在对每一个子块训练完毕后仅仅保留在该步的训练中得到的支持向量，并把这些支持向量加入到下一步的训练集中。这种算法同前面提到的选块算法、分解算法有些相似之处，但增量算法对每个样本仅仅进行一次判断，以决定它是否是支持向量，一旦某个样本在增量学习的一步中被丢弃则不会被再一次加入到训练集中。而选块算法、分解算法则可能对每个样本进行多次迭代以决定最终解。因此增量方法可以看成是分块算法的一种近似。

之后有很多对这种增量算法的改进，不过基本思想都是一致的，即利用SV集来代替训练集。如：萧嵘等提出的SVM 增量学习算法[Xiao 01]:构造了一个再分类-再训练循环,每次总是首先利用训练得到的模型对训练样本进行分类，然后仅把误分样本引入和SV 样本一起进行训练,直到误分样本比例小于系统设定的阈值，并通过权值的调整在多次训练中逐步积累起关于样本空间分类特性的知识，使得对样本有选择的遗忘成为可能。

但是增量学习算法每次把误分样本加入训练集中进行训练,这样是不全面的。由KKT 条件,只有违背KKT 条件的新增训练样本才能使原SV 集发生变化。违背KKT 条件的样本可以分为3 类:

(1) 位于分类间隔中,与本类在分类边界同侧,可以被原分类器正确分类的样本;

(2) 位于分类间隔中,与本类在分类边界异侧,被原分类器分类错误的样本;

(3) 位于分类间隔外,与本类在分类间隔异侧,被原分类器分类错误的样本;

可见,加入新增样本得到新的SVM 分类器时,分类错误只是样本违反KKT 条件的特定情况,所以KKT 条件比分类函数的分类判断更合理。Syed 等提出的增量学习算法将训练集中的所有非SV 样本抛弃,然后加入新增样本进行训练,这样也是不全面的,因为可能会丢失原来样本集中的信息。虽然SV 集在某些情况下可以代替原来的训练样本集,但随着新样本的加入,最优分类面会发生变化。为此人们又设计了一种新的增量学习算法，以是否违背KKT 条件为判断依据,违背则加入新的训练集,否则就放入测试集中;将训练集中的非SV 样本不做丢弃处理,而是把它放入测试集中,使它有可能再次符合条件进入训练集,得到了改进后的增量学习算法。

上述这些算法均作了一个假设即：数据是良好分布的，但现实中并不总是这种情况。在[Rüping 01]中给出了一个例子说明了在增量学习的不同阶段中如果数据的分布相差很

大会导致错误率很大的模型。在传统的批量学习方式中这是支持向量机的一个很好的性质：它说明了支持向量机有很好的抗噪声的能力，但是在增量学习的情况下，当数据的分布变化很大时，噪声很可能是需要考虑的旧支持向量。为此Stefan Rüping对增量算法提出了一点改进：通过对旧支持向量的误差加一个更大的权重，来增大旧支持向量的影响力。

2.8 支持向量机的一些进展

以 Vapnik 的统计学习理论为基础的支持向量机学习算法（SVM）出现以后，由于其对泛化性能的保证，受到了学术界极大的关注。为了得到更快的训练速度、更好的扩展性能，研究人员提出了很多新的 SVM 训练算法，而且在统计学习框架下，出现了很多新的 SVM 模型。下面根据本文的内容对部分相关工作进行简单地介绍。

2.8.1 邻近支持向量机 —— Proximal SVM

经典 SVM 通过在原空间或特征空间中构造一个超平面将正负两类数据点划分开来，同时最大化过两类边界划分超平面的间隔，以获得良好的推广能力。与此不同，邻近支持向量机（Proximal SVM —— PSVM）通过对两类数据点构造平行的“拟合”超平面来实现构造分类器的目的，未知类别样本点的类别根据它与哪个类拟合超平面的距离较近来确定。PSVM 是一种利用正则化最小二乘法实现分类的算法，可以看成正则化网络的一种特殊形式。PSVM（非）线性分类器的训练只需求解一个线性方程组即可，而不需要经典 SVM 比较耗时的二次优化问题的求解，非常的简单、快速。我们将在第三章对相关工作进行详细地介绍。

2.8.2 SVM 学习算法的发展

经典支持向量机的训练有 $O\left(m^2\right)$ 的空间复杂度，其中 m 是训练集中输入样本点的个数。因此当训练集很大时，经典 SVM 的学习方法在计算上是不可行的。为了解决这个问题，常用的方法是采取迭代的方式求得最优解的一个近似解。

为了减小 SVM 训练的时间复杂度，一种流行的技术是采用核矩阵的低阶近似如：Nyström 方法[Williams 01]、贪婪近似[Smola 00]、取样技术[Achlioptas 02]、矩阵分解技术[Fine 01]等。但是对于非常大的训练数据集，上述技术得到的核矩阵仍然太大而难以处理。

另外一种提升核方法处理大数据集能力的技术是分块[Vapnik 98]或更加复杂的分解技术（[Chang 04]、[Osuna 97]、[Platt 99]、[Vishwanathan 03]）。但是分块算法需要对全部 Lagrange 对偶变量非零的点进行优化，所以仍有可能遇到核矩阵过大的困难。[Osuna 97]里建议只对一个固定大小的子集（工作集）进行优化，工作集外的数据点的对偶变量保持不变。SMO 算法（Sequential Minimal Optimization）[Platt, 1999]把分解式算法推向了极致，将原优化问题分解为一系列极小的二次优化，即每次仅对两个输入数据点进行操作，更新其对偶变量值。

一种更加极端的方法是完全不采用二项式优化的方式求解，而采用线性方程组的求解方法如 Mangasarian 等人提出的 PSVM[Fung 01]、Lagrangian SVM (LSVM) [Mangasarian 01b]、[Fung 03]等。但其非线性模型仍需要求解一个$m \times m$矩阵的逆。[Kao 04]和[Yang 05]分别针对线性 SVM 和利用 Gaussian 核的非线性 SVM，提出了处理大数据集 SVM 的训练方法。

同分解类算法处理大数据集方法的思想类似，另外一类SVM学习算法在对输入样本集训练之前就对训练数据进行选择，以减小训练数据的规模。如：[Pavlov 00b]中提出的利用Boosting的方法合并多个子SVM分类器，每个子SVM分类器只利用全部训练数据集中的一小部分数据进行训练，[Collobert 02]则利用一个基于神经网络的方法来合并这些子分类SVM器；Lee和Mangasarian则提出了一种Reduced SVM[Lee 01]，RSVM通过随机选择核矩阵中的一些列来构造大数据集的分类器；除了随机选择之外，也可采取Active学习([Schohn 00], [Tong 00]), Squashing ([Pavlov 00a])，Editing ([Bakir 05])或者聚类([Boley 04], [Yu 03])等更加智能的方法来对训练数据集进行采样；其它相似的方法包括Kernel Adatron([Friess 98]）和SimpleSVM([Vishwanathan 03])等。[Tresp 01]和[Schölkopf 02]的第10章提供了更加完整的研究进展，感兴趣的读者可以参考。

实践中，当前 SVM 的实现时间复杂度一般在$O(m)$和$O(m^{2.3})$之间[Platt 99]；利用[Collobert 02]中提到的并行混合技术复杂度可以进一步降低到$O(m)$。但这些都是经验的观察结果而不是理论证明的结果。

2.8.3 极限学习机 —— Extreme Learning Machine

目前神经网络的学习速度与人们期望的相比差很远，这也成为神经网络在现实应用中一个主要的瓶颈。学习速度之所以很慢，两个主要的原因是：（1）目前神经网络的训练算法普遍采用基于梯度下降进行迭代更新的方法；（2）学习算法需要学习神经网络的所有权重参数。同经典神经网络学习算法不同的是，极限学习机（Extreme Learning Machine - ELM）[Huang 04]随机产生单隐层前馈神经网络（Single hidden Layer Feedforward neural Networks - SLFNs）输入权重的参数值，仅仅需要对 SLFNs 的输出权重进行求解。这使得 ELM 具有极快的速度，而且也避免了局部极小值等传统 SLFNs 学习算法的困难。

ELM 算法并不是基于结构风险最小化的学习算法，但在本文中我们将把 ELM 随机构造输入权重的思想引入到 SVM 中。我们将在第五章对 ELM 进行较详细的介绍。

第三章 非线性邻近支持向量机的增量学习算法

Fung G., Mangasarian O 在正则化最小二乘、正则化网络的基础上提出了一种新的支持向量机（SVM）算法：PSVM。经典的 SVM 通过在原空间或特征空间中构造一个超平面将正负两类数据点划分开来，同时最大化过两类边界的划分超平面的间隔，以获得良好的推广能力。与此不同，邻近支持向量机（Proximal SVM —— PSVM）通过对两类数据点构造平行的“拟合”超平面来实现构造分类器的目的，未知类别样本点的类别根据它与哪个类样本的拟合超平面较近来确定。如图 3.1、图 3.2 所示。

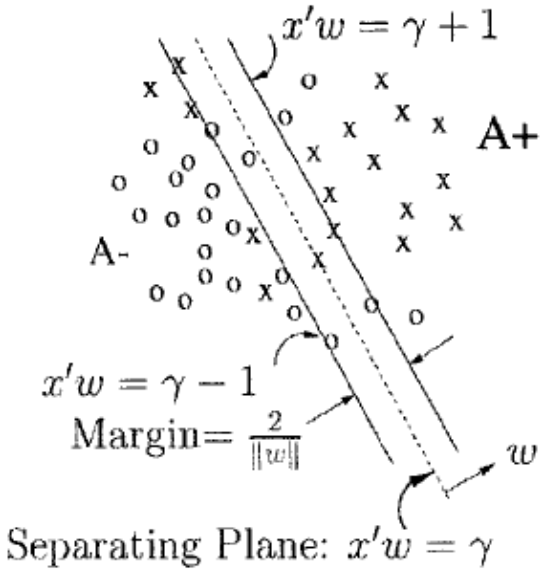

图 3.1. 标准 SVM

PSVM 是一种利用正则化最小二乘法实现分类的算法，可以看成正则化网络的一种特殊形式。PSVM（非）线性分类器的训练只需求解一个线性方程组即可，而不需要经典 SVM 比较耗时的二次优化问题的求解，非常的简单、快速。

线性 PSVM 的训练存在着一个简单的算法，比较容易进行增量学习；但非线性 PSVM（NPSVM）要求对一个大小为训练样本点个数平方的线性方程组进行求解。因此它存在着与标准 SVM 相似的空间复杂度问题，而且 NPSVM 不能像线性情形那样进行增量学习。

针对 NPSVM 存在的问题，我们给出了自己的解决方案。我们给出了一种新的 NPSVM 分类器形式，在这个分类器的基础上我们设计了一种新的 NPSVM 增量学习算法。这种增量学习算法可以有效的应用于增量学习情景中，当新数据到来时可以有效的利用历史训练结果，而不必对全部数据重新训练一遍，并且最终得到与利用批量方式学习相同的结果。

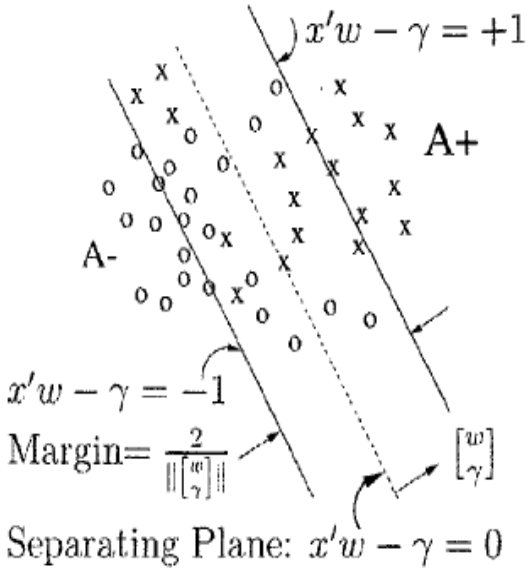

图 3.2 邻近 SVM

为了叙述的方便，首先把这里将用到的数学表示符号声明如下：所有的向量默认均为列向量，上标'表示转置，向量 x 和 y 在空间中的点积表示为：$x'\cdot y$，向量 x 的 2-norm 表示为 $\|x\|=\sqrt{x'\cdot x}$；这里我们利用矩阵 $A[m\times n]$ 表示 n 维空间 R^n 中的 m 个训练样本；对角线元素为 ±1 的对角矩阵 $D[m\times n]$ 的对角线上的元素声明了 m 个训练样本的类别是 $+1$ 或 -1；对于矩阵 $A\in R^{m\times n}$ 和 $B\in R^{n\times l}$，核函数 $k(A,B)$ 将 $R^{m\times n},R^{n\times l}$ 映射到 $R^{m\times l}$；默认情况下我们将使用下面的 Gaussian 核：

$$\left(K(A,B)\right)_{i,j}=\varepsilon^{-\mu\left\|A_i'-B_j\right\|^2},i=1...m,j=1...l$$

其中 μ 是正常量，ε 表示自然对数的底；e 为元素为1的任意维向量（维度根据上下文确定）；w,r 分别为分类超平面的方向系数和偏置，y 为松弛向量，参数 $v\geq0$ 用于控制模型复杂度和准确率之间的平衡；I 表示单位向量。

3.1 研究工作基础

Fung和Mangsarian提出了一种新的训练速度很快的支持向量机模型[Fung 01a]：邻近支持向量机(Proximal Support Vector Machine PSVM)。在PSVM中 $x'w-r=\pm1$ 不再是分隔平面，而是与两类点平均距离最近的平面，通过优化算法在最大化 $x'w-r=\pm1$ 间隔和最小化误差之间找到一个平衡解。PSVM的出发点是如下的优化函数（[Mangasarian 01a]，[Mangasarian 01b]）：

$$\min_{(w,r,y)\in R^{m+1+m}} v\frac{1}{2}\|y\|^2+\frac{1}{2}(w'w+r^2) \quad s.t. \quad D(Aw-er)+y\geq e, \tag{3.1}$$

注意到与标准支持向量机不同的是这里目标函数最小化 y 的二次模，并且考虑了偏移量 r（从而得到严格凸的优化问题），这里并不需要对 y 的非负约束，因为如果存在 y_i 为负那么将它置零后我们有更优的目标函数而且约束仍然满足。大量的实验结果表明([Lee 99], [Mangasarian 01a]，[Mangasarian 01b],[Mangasarian 99],[Lee 01])由该优化得到的模型同标准支持向量机相比有类似的准确率，而且由于该优化是严格凸的，这就保证了全局解的唯一存在性。

PSVM的关键在于将不等式约束变为等式约束：

$$\min_{(w,r,y)\in R^{m+1+m}} v\frac{1}{2}\|y\|^2+\frac{1}{2}(w'w+r^2) \quad s.t. \quad D(Aw-er)+y=e, \tag{3.2}$$

这个变化虽然很简单，但是确极大的简化了求解的步骤：我们可以给出一个显示的解析解，而对于标准支持向量机来说这是不可能的。公式(3.2)的几何意义同标准SVM有所不同，这里 $x'w-r=\pm 1$ 不再是分隔超平面而是最邻近平面，在其周围分布着大多数的点，目的是使两类点到它们的平均距离最小，同时为了保证有较好的推广性能，通过最小化 $(w'w+r^2)$ 尽可能的扩大两个超平面之间的距离。

公式(3.2)的KKT充分必要最优条件，可以通过将如下的Lagrange函数对 (w,r,y) 求偏导得到：

$$L(w,r,y,u)=v\frac{1}{2}\|y\|^2+\frac{1}{2}(w'w+r^2)-u'(D(Aw-er)+y-e). \tag{3.3}$$

这里， $u\in R^m$ 是公式(3.2)的等式约束对应的Lagrange对偶乘子：

$$\begin{cases}\dfrac{\partial L}{\partial w}=0\rightarrow w=A'Du\\ \dfrac{\partial L}{\partial r}=0\rightarrow r=-e'Du\\ \dfrac{\partial L}{\partial y}=0\rightarrow vy=u\end{cases}. \tag{3.4}$$

将(3.4)中的前三个等式带入等式约束中，我们可以得到如下关于 Du 解的表达式：

$$\begin{aligned} Du &= (\frac{I}{v} + AA' + ee')^{-1} De = (\frac{I}{v} + EE')^{-1} De \\ E &= [A, -e] \end{aligned} . \quad (3.5)$$

由于公式(3.5) 包含了一个$m \times m$矩阵的逆，使用 Sherman-Morrison-Woodbury (SMW)公式 [Golub 96]可以得到如下Du解的另一个表达式:

$$Du = v(I - E(\frac{I}{v} + E'E)^{-1} E')De . \quad (3.6)$$

利用(3.6)通过一系列的数学转换我们可以得到 (w,r) 下列解：

$$\begin{bmatrix} w \\ r \end{bmatrix} = \left(\frac{I}{v} + E'E \right)^{-1} E'De . \quad (3.7)$$

(3.7)中仅仅包含了一个维数相对可能较小的$(n+1)\times(n+1)$矩阵：$\frac{I}{v} + E'E$ 。因此线性PSVM的训练是很快的，它仅需求解一个线性方程便可得到(w,r)的解，而不是二次优化。而且我们需要在内存中仅仅包括： $m \times (n+1)$ 维的矩阵 E ， $(n+1)\times(n+1)$ 维的矩阵 $E'E$ 和 $(n+1)\times 1$ 维向量 $d = E'De$ 。通常输入空间的维数相对较小（小于10^3），因此即使训练集很大，PSVM也能够在较短的时间内给出分类结果。

容易看出算法的存储复杂度受限于$m \times (n+1)$维的矩阵E，为了能够处理海量数据集，[Fung 01b]中利用$E'E$和$d = E'De$的计算性质给出了一种增量学习方法，利用这种增量方法我们可以方便得向训练集中添加样本或从训练集中取出样本，并且高效的更新模型而不用重新训练。我们假定当前的分类器的输入数据集是$E \in R^{m\times(n+1)}$，对角矩阵$D \in R^{m\times m}$的对角线上的± 1值给出了每个点的类别信息。此时我们希望从数据集E中丢弃一部分样本，将这部分样本用$E^1 \in R^{m^1\times(n+1)}$表示，它是$E$的子集， $D^1 \in R^{m^1\times m^1}$ 是与E^1对应的子集；同时我们又有一些新的样本需要加入到训练集中，我们用$E^2 \in R^{m^2\times(n+1)}$和$D^2 \in R^{m^2\times m^2}$分别表示新的训练集以及其分类信息。[Fung 01b]中给出了如下的增量公式：

$$[w,r]' = (\frac{I}{v} + E'E - E^i{}' \times E^i + E^{i+1}{}' \times E^{i+1})^{-1} (E'De - E^i{}' D^i e + E^{i+1}{}' D^{i+1} e) \quad (3.8)$$

我们可以看到在(3.8)中对每块数据$E^i \in R^{m^i\times(n+1)}$，我们只需存储一个$(n+1)\times(n+1)$维的矩阵$E^i{}' \times E^i$和一个$(n+1)\times 1$维的向量$E^i{}' D^i e$。因此(3.8)与样本的个数无关仅仅与维数有关，

进而我们可以很方便的向训练集中添加或去除任意数量的样本。而且这种方法使我们能够处理任意大的样本集：只需将一个大的样本集分成很多子块，然后将每个子块依次以 $E^i{}'\times E^i$ 和 $E^i{}'D^ie$ 的形式加入到模型中即可。由于计算 $E^i{}'\times E^i$ 需要 $2(n+1)^2m^i$ 步计算；计算 $E^i{}'D^ie$ 需要 $2(n+1)m^i$ 步计算，因此这种增量算法的时间复杂度仅仅是参与训练的样本集的样本数的线性函数 $m=\sum m^i$ 。

对于非线性情形[Fung 01a]中给出了利用某一核函数 $K(.,.)$ 得到的非线性PSVM：

$$\begin{aligned}\min_{(u,r,y)\in R^{m+1+m}} \quad & v\frac{1}{2}\|y\|^2+\frac{1}{2}(u'u+r^2)\\ s.t. \quad & D(K(A,A')Du-er)+y=e,\end{aligned} \tag{3.9}$$

令 $K=K(A,A')$，(3.9)的Lagrange函数可表示为：

$$L(u,r,y,s)=v\frac{1}{2}\|y\|^2+\frac{1}{2}(u'u+r^2)-s'(D(KDu-er)+y-e), \tag{3.10}$$

这里 s 是(3.9)中的等式约束对应的Lagrange乘子，最优化条件可由将Lagrange函数对 (u,r,y) 求偏导得到：

$$\begin{cases}\dfrac{\partial L}{\partial u}=0\rightarrow u=DK'Ds\\ \dfrac{\partial L}{\partial r}=0\rightarrow r=-e'Ds\\ \dfrac{\partial L}{\partial y}=0\rightarrow vy=s\end{cases} \tag{3.11}$$

把(3.11)三个等式带入到等式约束中我们可以得到如下 Ds 的解析解：

$$Ds=(\frac{I}{v}+KK'+ee')^{-1}De \tag{3.12}$$

把(3.12)关于 s 的等式代入(35)中就可得到 (u,r) 的解析解。此时的分类超平面可以表示为：

$$\begin{aligned}K(x',A')Du-r&=K(x',A')DDK(A,A')Ds+e'Ds\\&=\left(K(x',A')K(A,A')+e'\right)Ds=0\end{aligned} \tag{3.13}$$

对应的分类器可表示为：

$$\left(K\left(x',A'\right)K\left(A,A'\right)+e'\right)Ds\begin{cases}>0,\text{then } x\in A+\\ <0 \text{ then } x\in A-\\ =0 \text{ then } x\in A- \text{ or } x\in A+\end{cases} \tag{3.14}$$

和线性分类器情形不同由于 K 是一个 $m\times m$ 维的方阵，这里SMW公式将不再适用。所以这里我们遇到了两方面的困难，其一是优化方面的困难：(3.9)是一个包含 $m+1$ 个变量的二次优化公式，(3.12)则需求解一个 $m\times m$ 维矩阵的逆,而 m 表示样本的个数，这个数值通常会比较大；其次是分类器使用方面的困难：我们可以看出(3.14)依赖于整个数据集 A，这样当数据集很大时模型的存储会占用很大的空间，而且使用它对一个新的数据点分类也会非常的慢。

为解决这两方面的问题，Lee等人在[Lee 01]中给出了一种RSVM的算法：通过从数据集中随机选取 $\bar{m}$ 个样本点组成训练集 $\bar{A}$，将 $m\times m$ 维的核矩阵 $K=K\left(A,A'\right)$ 缩小为一个 $m\times\bar{m}$ 维的矩阵 $K=K(A,\bar{A}')$，而 $\bar{m}$ 可以取得很小，如总样本数的 m 的1%。这种方法不仅可以用于解决大数据集的计算问题，而且有时也会产生出泛化性能更好的模型（由于只使用了一部分样本，避免了过学习现象的发生）。但是由于在增量学习中样本点是逐步添加的，这种技术并不适用于增量学习，而且其内存复杂度还是偏高即当 m 很大时 $\bar{m}$ 必须取很小的值才可以。

而我们的目的正是设计一种适用于非线性 PSVM 的增量学习算法，以解决 PSVM 大数据集的非线性学习和在线学习问题。

3.2 新的非线性 PSVM 分类机

Fung 等人在[Fung 01]中给出的非线性模型的解析解同线性模型有不同的形式，线性模型的增量算法不适用于非线性模型，我们有必要设计一种新的非线性 PSVM 模型，使得在这种模型上构造增量学习算法成为可能。这种 Nonlinear PSVM 应满足以下条件：

- 其应该具有 PSVM 的几何意义，其模型应该是严格凸的等式约束二次优化，求解过程同 Linear PSVM,、标准 Nonlinear PSVM 一样快速（或者更优）；
- 这种分类器模型应具有与标准 SVM、PSVM 相似的对泛化性能的保证；
- 能够利用其解的形式方便地构造增量算法。

经典 Nonlinear PSVM 解的形式如（3.12）所示，同线性 PSVM 解的形式（3.6）不同，这里 Sherman-Morrison-Woodbury 公式并不起作用。考虑标准支持向量机中处理非线性分类器学习的方法：利用一个非线性变换函数 $\phi\left(x\right)$: $R^n\to R^{\tilde{n}}$ 把输入空间 R^n 中的样本点映射到高维的特征空间 $R^{\tilde{n}}$ 中，由于在高维空间中训练数据集线性可分的可能性更大一些，可以通过在高维特征空间 $R^{\tilde{n}}$ 中构造线性分类器来实现构造原空间中非线性分类器

的目的。SVM 的训练仅会用到向量之间的内积计算，这使得可以利用核函数来实现高维空间的内积计算，从而避免了维数灾难。

与 SVM 中的方法类似，这里我们首先使用一个非线性转换函数$\phi(x)$: $R^n \to R^{\tilde{n}}$，把输入空间R^n中的样本点映射到高维的特征空间$R^{\tilde{n}}$中，然后在$R^{\tilde{n}}$中构造 PSVM 线性分类器。具体地，新非线性分类器的模型如下所示：

$$\begin{aligned} \min_{(w,r,y)\in R^{m+1+m}} \; & v\frac{1}{2}\| y \|^2 + \frac{1}{2}\left(w'w + r^2\right) \\ s.t. \quad & D\left(\phi(A)\mathrm{w} - er\right) + y = e, \end{aligned} \quad . \tag{3.15}$$

为了同[Fung 01]中的 NPSVM 进行区分，我们将模型(3.15)命名为 Simple NPSVM (S-NPSVM)。

我们可以得到如下 S-NPSVM 的 KKT 最优化充分必要条件：

$$\begin{aligned} & w = \phi(A)'Du, \\ & r = -e'Du, \\ & vy = u, \\ & D(\phi(A)w - er) + y - e = 0 \end{aligned} \quad . \tag{3.16}$$

其中$u \in R^m$是（3.15）式中等式约束的 Lagrange 对偶乘子。 将（3.16）中前三个等式代入到第四个等式中，我们可以得到对偶变量Du的解：

$$Du = (I/v + \phi(A)\phi(A)' + ee')^{-1}De = (I/v + K + ee')^{-1}De \quad . \tag{3.17}$$

其中K是核函数$K\left(A, A'\right) = \phi(A)\phi(A)'$的缩写。与 NPSVM 的解(3.12)相比，S-NPSVM 的解（3.17）去掉了两个核矩阵的乘积，因此 S-NPSVM 的求解速度会更快一些。

其对应的分类器的形式如下：

$$f\left(x\right) = \mathrm{sgn}\left(K\left(x', A'\right)Du - r\right) \quad . \tag{3.18}$$

我们可以看到它也存在一个解析解（通过求解一个线性方程组）。通过这种方式构造的分类器继承了 PSVM 训练速度快的优点，而且可以方便地构造增量学习算法，测试结果显示它具有比较好的泛化性能。

3.3 非线性邻近支持向量机的增量学习算法

目前关于标准支持向量机的增量算法有很多的研究，不过根本的出发点是一致的，即利用支持向量可以代表整个数据集的训练结果这一性质，每一步仅仅保留相应的支持向量，然后同新样本一起训练。对于 Linear PSVM 有更好的结果，由增量公式（3.8）可以看出，利用这个公式可以随意地向数据集中添加或删除样本，同时可以得到与利用批量学习方式同样的结果，另外我们注意到这个公式独立于数据集中样本点的个数，它只

跟样本的维数有关，而通常维数要比样本点的个数小的多。但是这些结果却不适用于非线性 PSVM，首先 PSVM 与标准支持向量机具有不同的几何意义，在 PSVM 中并不存在传统意义上的支持向量，因此传统 SVM 的增量算法对非线性 PSVM 并不适用，其次非线性 PSVM 的解如前所述具有和线性 PSVM 不同的形式，线性 PSVM 的增量算法也不适用。这里我们就要利用上一节提到的 S-NPSVM 模型构造一种适用于非线性 PSVM 的增量算法。

要设计出一种增量算法，即当我们获得新的样本后要以一种有效的方式更新历史模型，具体地说我们有以下问题需要解决：

- 利用新的样本和旧样本的训练结果构造新的分类器；
- 应避免对旧数据集的重复训练，以缩短训练时间；
- 得到的模型应与利用新旧全部样本进行训练得到的结果相似。

从（3.17）式可以看出S-NPSVM的求解主要依赖于对一个 $m\times m$ 维矩阵逆的求解，而利用分块矩阵求逆的性质，我们可以得到S-NPSVM的增量学习算法。在一个典型的在线学习情形下，假设当前我们已经对训练数据集 $A_1 \in R^{m_1\times n}$ 及其类别信息 $D_1 \in R^{m_1\times m_1}$ 进行了训练并得到了其解，及相应矩阵的逆：

$$A_{11}^{-1} = \left(I/v + K\left(A_1, A_1'\right) + ee'\right)^{-1} . \tag{3.19}$$

现在我们需要将一批新的训练数据 $A_2 \in R^{m_2\times n}$ 及其类别信息矩阵 $D_2 \in R^{m_2\times m_2}$ 加入到训练数据集中，即我们需要根据新的训练数据 $A_2 \in R^{m_2\times n}$ 来调整原来学习得到的分类器。根据公式（3.17），此时我们需要求解由历史数据 A_1 和新数据 A_2 构成的训练数据集 $A_t = [A_1', A_2']'$ 对应的对偶变量 Du 的解，即计算下面这个矩阵的逆：

$$\begin{aligned} &A_{tt} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \\ &\text{where } A_{12} = \left(K\left(A_1, A_2'\right) + ee'\right), A_{22} = \left(I/V + K\left(A_2, A_2'\right) + ee'\right) \text{ and } A_{21} = A_{12}' \end{aligned} . \tag{3.20}$$

由于我们已经得到了 A_{11} 的逆 A_{11}^{-1}，利用分块矩阵求逆的公式，我们可以增量的计算 $(m_1+m_2)\times(m_1+m_2)$ 维矩阵 A_{tt} 的逆：

$$A_{tt}^{-1} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}^{-1} = \begin{bmatrix} A_{11}^{-1} + X & Y \\ Y' & T \end{bmatrix} . \tag{3.21}$$

其中 T， X 和 Y 由以下等式给出：

$$\begin{aligned} &T = (A_{22} - A_{21}A_{11}^{-1}A_{12})^{-1}, \\ &Y = -A_{11}^{-1}A_{12}T, \\ &X = -YA_{12}'A_{11}^{-1} \end{aligned} . \tag{3.22}$$

对公式（3.22）进行分析可以看出：利用增量学习的方式计算矩阵 A_{tt} 的逆，我们需要分别计算 $m_1 \times m_2$ 维的矩阵 A_{12}， $A_{22} \in R^{m_2 \times m_2}$， $T \in R^{m_2 \times m_2}$ （需要求解一个矩阵的逆），$X \in R^{m_1 \times m_1}$ 和 $Y \in R^{m_1 \times m_2}$ 的值；时间复杂度约为 $O\left(m_2^3\right) + 2m_1 m_2^2 + 2m_1^2 m_2$，而批量学习的方式的时间复杂度大致为 $O\left((m_1 + m_2)^3\right)$。因此在保持学习所得结果不变的同时，增量学习明显的减少了训练所需的计算量。

3.4 实验结果

本节所有的测试结果，都是在 731Mhz 主频的 Pentium III 处理器和 512M 内存配置的机器上完成的。为了完成该实验，我们从 UCI 机器学习公共数据库中选择了三个用于衡量分类算法性能的公共测试数据集：Ionosphere, Bupa Liver, Tic-Tac-Toe，这四个公共数据集是线性不可分的，因此比较适用于使用非线性分类器进行训练。

在表一中，我们将 S-NPSVM[Liu 07a]同 NPSVM [Fung 01]进行比较，这里两个模型的参数设置均由对测试数据集的 10-折交叉确认的平均结果决定。表一中的数据表明，S-NPSVM 的训练时间要比 NPSVM 短（这也验证了我们前面的理论分析），而且具有与 NPSVM 相当的训练、测试正确率。

在表二中我们对提出的增量学习算法进行测试验证。为了模拟在线学习中增量学习的情景，我们首先把每个测试数据集分成大小相当的两个子块，然后利用 S-NPSVM 对其中一个子块进行训练得到一个初始的模型，然后将第二个子块中的数据做为在线学习中新到的数据进行处理。有两种方法来完成这个任务，第一种方法是利用 S-NPSVM 或 NPSVM 重新进行训练，这相当于直接利用批量学习的方法来完成在线学习的任务；第二种方法是利用增量学习算法直接将新数据加入到分类器中，从而避免重复训练。从表二中的测试结果我们可以看出，这两种方法可以得到同样的训练结果（同样的模型，同样的测试、训练正确率）即该增量学习技术对模型的正确率没有影响；但是增量学习的方法可以显著的减少所需的训练时间。更进一步，可以想象如果我们把测试数据集划分为更多的部分，所节省的时间将更加的显著。

表 3.1 NPSVM 和 NNPSVM 在 Ionosphere, Bupa Liver, Tic-Tac-Toe 上的平均召回率、10 折交叉测试正确率和训练时间，参数 v 由对训练数据集的 10 折交叉确认确定

	Data Set $m \times n$	Ionosphere 351×34	Bupa Liver 345×6	Tic-Tac-Toe 958×9
NPSVM	Recall	99.43%	80.23%	100%
	Test	93.17%	66.87%	99.06%
	Time(Sec.)	0.16	0.16	2.32
S-NPSVM	Recall	99.94%	84.73%	100%
	Test	94.80%	67.27%	99.06%
	Time(Sec.)	0.14	0.13	1.56

表3.2 NPSVM和增量学习算法在Ionosphere, Bupa Liver, Tic-Tac-Toe上的召回率及训练时间；参数ν由对训练数据集的10折交叉确认确定

	Data Set $m \times n$	Ionosphere 351×34	Bupa Liver 345×6	Tic-Tac-Toe 958×9
INPSVM	Recall Time(Sec.)	99.72% 0.21	83.48% 0.21	100.00% 3.75
NNPSVM	Recall Time(Sec.)	99.72% 0.33	83.48% 0.33	100.00% 5.03

3.5 总结及下一步的工作方向

在这一章中我们给出了一种新的NPSVM分类器模型——S-NPSVM，在这个分类器的基础上我们设计了一种新的NPSVM增量学习算法。这种增量学习算法可以有效的应用于增量学习情景中，当新数据到来时可以有效的利用历史训练结果，而不必对全部数据重新训练一遍，并且最终得到与利用批量方式学习相同的结果。

测试结果表明，S-NPSVM具有与NPSVM相比更少的训练时间和相似的正确率；同时增量学习算法可以有效的应用于在线学习的情景当中，避免重复学习，减少训练所需的时间，同时得到与批量学习方式相同的分类模型。

为了处理不同类别样本个数差距过大时带来的训练准确度的问题，[Fung 01c]提出了PSVM的一种变形通过对不同类样本的误差进行加权，这种方法同样可以很方便的扩展到S-NPSVM中。

利用本章中提到的增量学习算法，我们可以有效的快速进行在线学习。但是该增量学习算法并没有解决NPSVM的空间复杂度问题，由公式（3.20）和（3.21）可以看出S-NPSVM对空间的要求仍然是与样本个数的平方成正比。因此与经典SVM的不能有效的对大数据集进行学习的困境类似，我们需要一种方法来处理大数据集的训练难题。

对于大数据集的训练，已经有了很多的研究成果，如经典SVM的分块训练方法：SMO，通过采取对原始问题分解、迭代的方式得到优化问题的近似最优解；而PSVM的样本随机选择技术RSVM则通过随机选择一部分样本得到一个子核矩阵来进行训练。但是所有这些方法并不能有效的应用于在线学习的环境中。在下一章里我们为PSVM设计了样本选择技术，其可以应用于线性、非线性PSVM模型的在线学习中，用于解决大数据集的PSVM在线学习问题。

第四章 PSVM 样本选择技术

在处理海量数据集时，我们可以利用上一章的增量学习算法，把数据集分成很多的子块进行增量学习，但由于增量学习算法的空间复杂度与样本个数的平方成正比，随着训练的进行，所需的空间就会越来越大，所以我们需要一种能够选择出对最终模型最有价值的样本的样本选择技术，就像标准支持向量机中那样：只需保留支持向量就可以得到与利用全部样本进行训练相同的模型。

通过分析 PSVM 解的结构，以及不同位置的样本点对最终解的影响，本章给出了一种有效的样本选择技术。该样本选择技术可以根据训练结果，从训练数据集中选择出 PSVM 的“支持向量”，即用这些样本进行训练可以给出与利用全部样本进行训练相似的模型；该技术适用于线形和非线性 PSVM；不仅能够在将新增数据添加到训练集之前对增量学习中的历史数据进行选择，而且能够对新添加的样本进行选择，淘汰那些对最终结果没有影响的样本。

实验数据及理论分析表明该样本选择技术能够有效地应用于批量学习或在线学习中：选择出的样本能够比较好地代表历史数据的性质；因此也在很大程度上解决了 PSVM 在线学习中的空间复杂度问题。

4.1 研究工作摘要

近些年，随着人们所收集的数据量越来越大，增长速度越来越快，以及在线学习的应用，要求 SVM 算法不仅能够得到良好的训练结果，而且能够进行增量学习。另一方面，出现了很多种标准支持向量机[Vapnik 95]的变种，如 Proximal SVM [Fung 01]，Reduced SVM [Lee 01]，Least Squares SVM [Suykens 99]等，这些变种 SVM 模型的共同点是：将标准 SVM 中的不等式约束改为等式约束，训练时只需求解一个线性方程组，从而避免了比较耗时的二次优化的求解，而且具有与经典 SVM 相似的正确率。由于 PSVM 的求解具有代表性（严格凸的二次优化模型），而且具有最快的速度[Fung 01a]，本章我们专注于 PSVM 的样本选择技术。

Fung 在[Fung 01b]中为线性 PSVM 设计了有效的增量学习算法，而且线性 PSVM 分类器学习的代价独立与样本点的个数，但是由于非线性 PSVM（Nonlinear PSVM - NPSVM）具有与线性 PSVM 不同的解的形式，该方法并不能适用于 NPSVM，NPSVM 的学习算法具有$O\left(m^2\right)$的空间复杂度，其中m是样本点的个数。为了能够对大数据集训练 NPSVM 分类器，Lee 等人在[Lee 01]中设计了一种 Reduced SVM（RSVM）技术，它通过从样本点中随机地选择出一部分样本，在核矩阵中仅保留其对应的列，从而构成一个矩形形状的核矩阵（矩阵的每行仅包含所选择出的样本对应的点积），来进行求解。但是 RSVM 技术并不能应用于在线学习问题中，因为在线学习的开始阶段我们并不知道全

部的样本。当样本集比较大时，RSVM 对空间的要求仍然很大。

为了有效地进行在线学习，避免重复训练，在上一章中我们设计了新的 NPSVM 模型：S-NPSVM，在保证正确率与 NPSVM [Fung 01]相当的前提下，S-NPSVM 缩短了训练时间；从 S-NPSVM 解的形式出发，利用分块矩阵求逆公式，设计了一种新的增量学习算法[Liu 07a]，其能够有效地利用历史训练结果，缩短增量学习的训练时间。但是该增量学习算法的空间复杂度与训练样本个数的平方成正比，因此不能有效地处理样本个数很大的情况。

为了能有效地进行大数据集 NPSVM 分类器的在线学习，本章我们为 PSVM 设计了样本选择技术。该样本选择技术可以根据训练结果，从训练数据集中选择出 PSVM 的“支持向量”，即用这些样本进行训练可以给出与利用全部样本进行训练相似的模型；适用于线形或非线性 PSVM；不仅能够在将新增数据添加到训练集之前对增量学习中的历史数据进行选择，而且能够对新添加的样本进行选择，淘汰那些对最终结果没有影响的样本。最后我们在该样本选择技术的基础上给出了 PSVM 在线学习的框架。

利用该样本选择技术及上一章的增量学习算法，S-NPSVM[Liu 07a]被赋予了处理大数据集训练的能力；而且由于 PSVM[Fung 01]模型本身的简单特性，相比经典 SVM 在速度上也获得了很大的提升。

理论分析及在机器学习公共数据集上的实验结果表明：该样本选择技术能够有效地应用在批量学习或在线学习中，即选择出的样本能够有效的代表整个历史训练数据集或新增训练数据集的特征。

在第二节我们将介绍 PSVM 的样本选择技术，包括对历史数据的选择，对新增数据的选择，对非线性分类器学习的处理方法等，给出应用上一章的增量学习算法和本章样本学习技术的 PSVM 在线学习流程框架，该流程能够处理大数据集的在线学习；在第四节中我们给出了实验测试结果；最后我们在第五节对本章工作进行总结。

为了叙述的方便，首先把这里将用到的数学表示符号声明如下：这里所有的向量默认均为列向量，上标'表示转置，向量 x 和 y 在空间中的点积表示为：$x'\cdot y$，向量 x 的 2-norm 表示为 $\|x\|=\sqrt{x'\cdot x}$；这里我们利用矩阵 $A[m\times n]$ 表示 n 维空间 R^n 中的 m 个训练样本；对角线元素为 ± 1 的对角矩阵 $D[m\times n]$ 的对角线上的元素声明了 m 个训练样本的类别是 $+1$ 或 -1；对于矩阵 $A\in R^{m\times n}$ 和 $B\in R^{n\times l}$，核函数 $k(A,B)$ 将 $R^{m\times n}, R^{n\times l}$ 映射到 $R^{m\times l}$；默认情况下我们将使用下面的 Gaussian 核：

$$\left(K(A,B)\right)_{i,j}=\varepsilon^{-\mu\|A_i'-B_j\|^2}, i=1...m, j=1...l$$

其中 μ 是正常量，ε 表示自然对数的底；e 为元素为1的任意维向量（维度根据上下文确定）；w,r 分别为分类超平面的方向系数和偏置，y 为松弛向量，参数 $v\geq 0$ 用于控制模型

复杂度和正确率之间的平衡；I 表示单位向量。

4.2 PSVM 样本选择技术

要利用上一章提到的增量学习算法，我们仍需解决两个问题：首先从公式（3.20）和（3.21）的形式可以看出，该模型的存储复杂度是样本点个数的平方，即$O\left(m^2\right)$；其次，对未知类别的点x利用公式（3.18）进行分类时，需要做大量的运算。

对大数据集训练算法的研究，已经有了很多的成果，如经典 SVM 的分块训练方法：SMO，通过采取对原始问题分解、迭代的方式得到优化问题的近似最优解；而 PSVM 的样本随机选择技术 RSVM 则通过随机选择部分样本得到一个矩形核矩阵来进行训练。但是所有这些方法都不能有效地应用于在线学习的环境中。

为了使对大数据集的增量训练成为可能，本节给出了 PSVM 的样本选择技术，使得丢弃一部分对最终模型影响很小的样本成为可能。首先考虑标准支持向量机[Vapnik 95]的情况，当数据是分批加到训练数据集时，我们可以将历史数据中的支持向量取出来作为历史数据的代表（[Syed 99], [Ruping 01]）。之所以可以这样做是由于仅以支持向量作为训练集进行训练可以得到与利用全部样本集进行训练同样的分类器结果。由此启发，我们期望设计一种根据训练结果对样本进行选择的方法，它应该能够选择出最具代表性的 num（个数由用户指定）个样本。所选择出的样本应与经典 SVM 中支持向量的作用类似，即对这些样本进行训练可以得到与原分类器类似的模型。这种样本选择技术还可以应用到上一节中提到的增量学习方法中，当新数据到来时，仅仅选择历史数据集或新数据集中部分样本点来进行合并，从而解决增量学习过程中大数据集对存储空间的要求过大的问题。

在标准支持向量机中样本的选择很简单：只需保留支持向量（对偶变量的值为非零的点）就可以得到与利用全部样本进行训练相同的模型。但是 PSVM 或 S-NPSVM 具有与标准 SVM 所不同的几何意义，在 PSVM 中对偶变量u的取值是松弛变量的y常数倍，即 $y:vy=u$，而y通常是非零的（因为很少有样本点会恰巧落到拟合超平面上），这也就决定了 PSVM 不存在 SVM 中对偶变量非零的支持向量的概念。因此我们需要新的样本选择技术。

由[Gale 1960]中线性等式的理论可以得知，最多只需要$n+1$个线性独立的数据点，这可以唯一的确定参数$(w,r)\in R^{n+1}$的值，即仅需要一小部分数据就可以描述解(w,r)；对于非线性情况我们也可以做相似的推论。

首先分析 PSVM 解的结构，及不同位置的样本点对最终解的影响。在[Fung 01]中定义了ε支持向量的概念，即某数据点A_i，如果其松弛变量y_i的绝对值小于ε，我们则称它们为ε支持向量。试验表明，当ε取较小的正数时（如我们将ε设置为某一个较小的值，

使得大约10%的样本点是ε支持向量），利用这些ε支持向量进行训练得到的 PSVM 或 S-NPSVM 分类器能够具有与利用全部样本进行训练得到的分类器相似的正确率。在公式（3.15）中，y_i可以理解为样本点A_i到拟合超平面距离的一种量度，y_i的绝对值越大样本点A_i与拟合平面之间的距离越大。我们可以推断，在拟合平面附近的点相比离拟合平面较远的点具有更好的描述能力，因此当ε为一个适当的较小的实数值时，我们可以选择ε-*支持向量*作为全部样本点的一个代表。S-NPSVM是在特征空间中构造一个线性模型，ε-*支持向量*的概念及上述推论可以很容易的推广到S-NPSVM中。

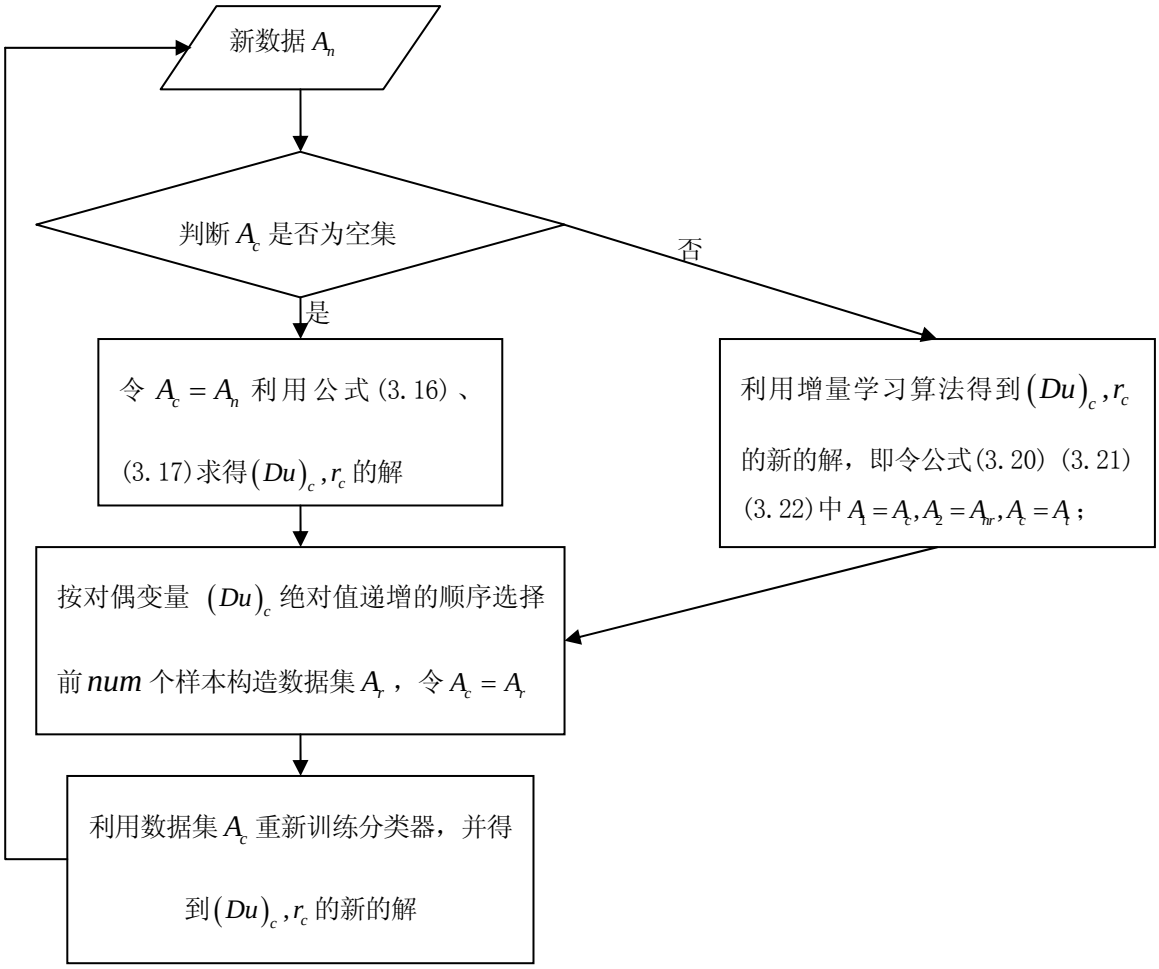

图 4.1 历史数据样本选择技术流程图

由上所述，我们可以选择与两条拟合超平面距离最近一些点，做为整个训练数据集的代表。可以认为，最靠近拟合超平面的点具有与 SVM 中支持向量类似的性质，即具有最好的概括描述能力，对这些样本进行训练可以得到与原分类器类似的模型。在线学习过程中，在对新数据集通过增量学习算法进行学习前，我们可以从历史数据中选择出部分ε支持向量作为历史数据的代表，从而减少增量学习算法对空间复杂度的要求。

对历史数据的选择过程可以用框图 4.1 表示。首先我们说明一下采用的表示方法，假设我们已经对数据集 $A_c \in R^{m_c \times n}$ 及其类别信息 $D_c \in R^{m_c \times m_c}$ 进行训练得到了分类器：$f_c(x) = \mathrm{sgn}\left(K(x', A_c')(Du)_c - r_c\right)$，现在到来了一批新的数据 $A_n \in R^{m_n \times n}$ 及其类别信息 $D_n \in R^{m_n \times m_n}$ 需要加入到分类器中。我们用 $A_r \in R^{m_r \times n}$ 表示从历史数据中选择出的样本，$A_{nr} \in R^{m_{nr} \times n}$ 表示从新数据中选择出的样本，$A_m \in R^{m_m \times n}$ 表示错分的样本。初始时我们有如下初值：$A_c = \varnothing, A_r = \varnothing, A_m = \varnothing, (Du)_c = 0, r_c = 0$。 最后，令 num 表示每一步需要选择出的样本的个数。

在增量学习的过程中，对于历史数据我们的目的是选择出最具代表性的样本，而样本选择也可以存在于添加新样本点的过程中。对于新到的数据，我们期望保留那些可能更新模型的样本点，去掉那些已经被模型所概括了的点。

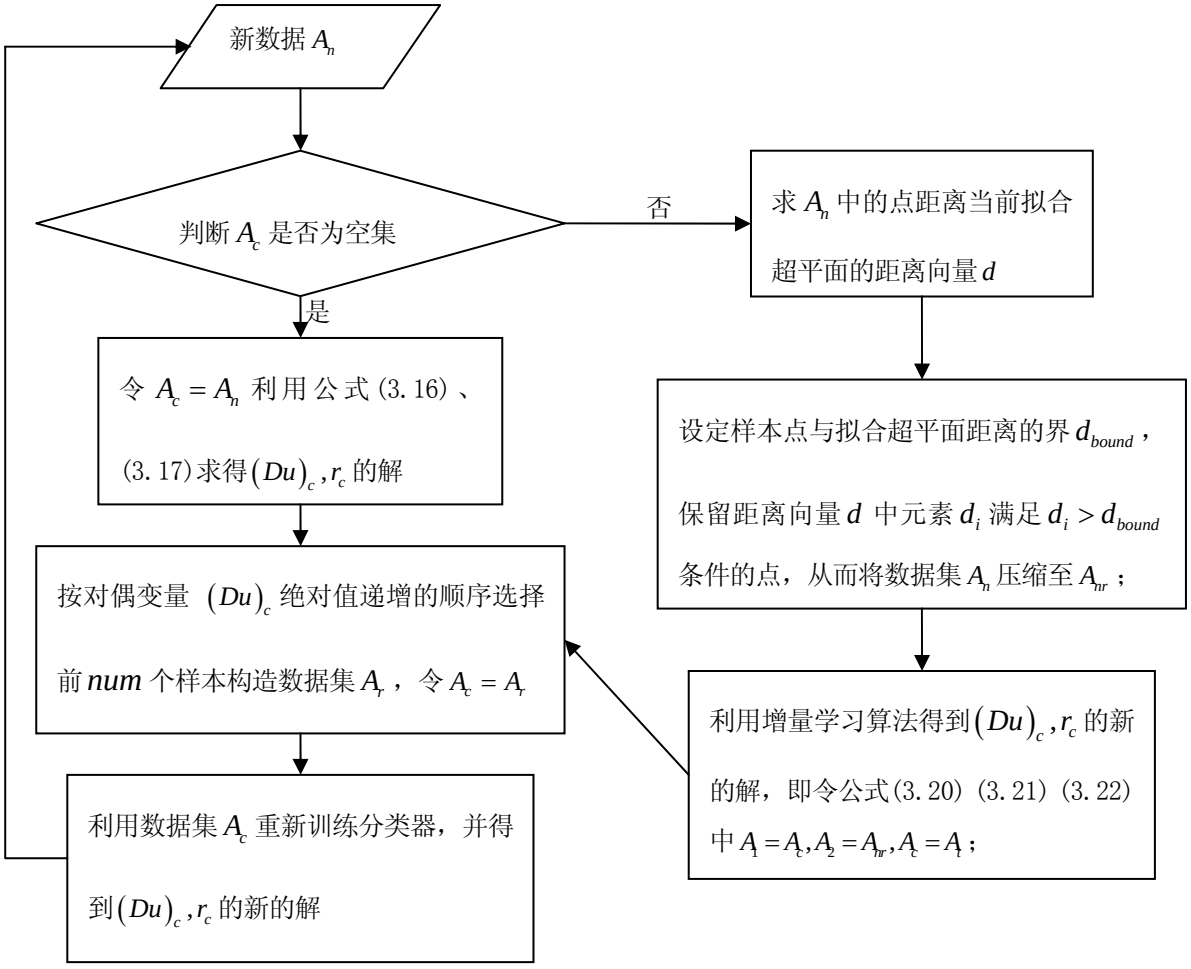

图 4.2 添加了对新到数据集进行选择的样本选择技术流程图

既然仅仅那些没有被已有分类器准确描述的样本点是有价值的，因此我们同样可以根据样本点与拟合超平面之间的距离来对新数据进行压缩，即仅仅那些远离拟合超平面的点被考虑到总体数据集中。这样做不仅可以进一步的减小内存需求，而且新数据和选择出的历史数据的容量不会相差太大。当然很有可能存在一些更优的方法，值得我们去研究。添加了对增量学习中新到数据集进行选择过程的样本选择技术流程图如图 4.2 所示。

上述样本选择技术可以有效的应用于线性情形，但是对于非线性情形，实验结果显示，要想得到满意的结果所需选择的样本数要比线性情形多很多。这也说明了非线性分类期的描述要比线性分类期的描述复杂很多。

为了处理这个问题，在上述针对线性分类器的基于距离的样本选择技术选择出的样本的基础上，非线性分类器的样本选择技术分两步进行：（1）与线性情况类似对历史数据、新到数据进行选择，期望选择出数据对整个模型有很好的描述能力；（2）对选择出的样本进行训练得到一个新的分类器（该分类器可以用于下一步的增量学习），利用该分类器对这个训练数据集进行测试，然后把测试错误的样本作为新样本添加到第一步选择出的数据集中。由于错分样本的个数通常比较少，利用增量学习公式(3.20)，(3.21)和(3.22)该方法导致的代价增加一般较小，而且实验结果显示该方法可以显著的提升非线性情况下的测试正确率。

注意到公式（3.16）中有 $vy = u$，这也说明我们可以直接由比较不同的样本点对应的对偶变量 u_i 的取值来确定不同的样本点到拟合超平面的距离大小。

根据前面的对样本选择技术的描述，我们可以设计如下的能过对大数据集进行处理的增量学习步骤。

1） 等待，直到有新的数据集 A_n 需要学习；

2） 如果 $A_c \neq \varnothing$，令 $A_n = A_n \cup A_m$ 并跳转到第 7 步，否则令 $A_c = A_n$；

3） 利用公式(3.16)、(3.17)求得 $(Du)_c, r_c$ 的解；

4） 按照样本点 Lagrange 对偶变量 $(Du)_c$ 的绝对值递增的顺序选择前 num 个样本构造数据集 A_r，并令 $A_c = A_r$；

5） 利用数据集 A_c 重新训练分类器，并得到 $(Du)_c, r_c$ 的新的解；

6） 利用分类器 $f_c(x)$ 对全部样本进行测试，获得错分的样本 A_m，转移到步骤 1；

7） 通过下述公式获得新数据集 A_n 距离当前拟合超平面的距离向量 d：

$$d = \left| f\left(A_n\right) \right| = \left| \operatorname{sgn}\left(K\left(A_n', A_c'\right)\left(Du\right)_c - r_c \right) \right| \qquad (8)$$

8) 设定一个界 d_{bound} 作为样本点与拟合超平面距离的界，一个可能的选择是 $\left(Du\right)_c$ 中元素绝对值的最大值；

9) 将数据集 A_n 压缩至 A_{nr}，仅保留对应距离向量 d 中元素 d_i 满足 $d_i > d_{bound}$ 条件的点；

10) 将 A_{nr} 作为增量学习中新增数据集，利用增量学习算法得到 $\left(Du\right)_c, r_c$ 的新的解，即令公式(12)、(13)、(14)中 $A_1 = A_c, A_2 = A_{nr}, A_c = A_t$；跳转到第 4 步（此时 A_{nr} 已经包含进 A_c）。

由上述步骤可以看出由样本选择所带来的时间代价增加约为 $O\left(num^3\right)$，而 num 一般可以比较小。但是它降低了对空间的要求，允许用户按照内存条件对训练参数进行调整，因此可以满足大数据集的增量学习的要求。进一步注意到 A_t 是一个 $R^{num \times n}$ 的矩阵，因此也减小了利用公式（3.18）进行分类时所需的计算。

4.3 实验结果

在这一节中，我们将在 4 个机器学习分类领域的公共测试数据集：Ionosphere，BUPA Liver，Tic-Tac-Toe 和 Iris 上，对本章提出的各种样本选择技术进行测试。本节中，所得到的结果数据的测试环境为：MATLAB 7.0，2.80GHz Pentium IV CPU，512 内存；我们在同样的机器上使用标准 SVM 的 C 语言工具包：LIBSVM 来模拟经典 SVM 的性能。实验中，SVM 使用的核函数是高斯径向基核函数：$\exp\left(-\left\|x-y\right\|^2 / \delta^2\right)$，其中的参数均通过 10 折交叉确认确定。

我们将测试 5 种分类器的性能，来说明本节提出的样本选择技术的有效性。这些分类器的不同之处在于所采用的样本选择技术，及它们是线性模型还是非线性模型。这 5 种分类器中包含 2 种线性分类器模型：一种是普通的 PSVM 分类器[Fung 01]，这里用 PSVM 表示；另外一种是应用样本选择技术后，对选择出的样本进行训练得到的线性分类器，这里用 PSVM-I 表示。另外 3 种分类器是非线性分类器：第一种是上一章中提出的 S-NPSVM 分类器，这里用 NPSVM-I 表示[Liu 07a]；第二种是对 NPSVM-I 利用本节提出的样本选择技术后，对选择出的样本进行训练得到的分类器，这里用 NPSVM-II 表示；最后一种分类器是在 NPSVM-II 样本选择技术中考虑错分样本影响，这里用 NPSVM-III 表示。

我们将 10 次 10 折测试的平均召回率、测试正确率的结果分别列在表 4.1、表 4.2 和

图 4.1 中。我们可以根据这些测试结果，做出如下观察结果：

1. 从表 4.1 和图 4.1 中，我们可以看出 PSVM-I 和 NPSVM-III 具有与 PSVM、NPSVM-I 相似的（甚至更好的）召回率、测试正确率，当所选择的样本数占总样本数的比例较大时（如 30%），这种现象尤其明显；
2. 当选择出的样本点的个数占总样本个数的比例较小时（如小于 10%），从图 4.1 中我们可以看出，NPSVM-II 的性能比较差，为了能得到较为满意的 NPSVM-II 测试正确率结果，需要选择出比线性分类器情景大很多的样本点的个数；
3. 从图 4.1 中我们可以看出，NPSVM-III 的正确率的曲线要明显的高与 NPSVM-II，这表明将分类错误的样本考虑到非线性分类器的样本选择技术中能够明显的提升所选择出的样本集的质量，对于 Ionosphere 数据集这一点尤其明显；
4. 从图 4.1 中我们可以看出，NPSVM-II 和 NPSVM-III 的召回率、测试正确率随着选择出的样本点的个数的递增而增加。

从表 4.2 中我们给出了本节提出的样本选择技术在线增量学习的过程中的应用效果。为了模拟在线学习的过程，我们首先将整个数据集划分为很多的子部分，对其中一个子部分进行训练，然后每次添加一个子部分来模拟在线学习过程中新到的数据。我比较了三个 NPSVM-III 分类器和 RSVM 分类器[Lee 01] 的结果（注意 RSVM 并不适用于增量学习，表 4.2 中的 RSVM 的测试结果数据通过随机选择整个数据集中 10%的样本作为训练数据集学习得到），这三个 NPSVM-III 分类器在在线增量学习过程中，每一步分别选择全部已知的 10%，20%和 30%。从表 5.2 中的数据我们可以看出，本节提出的样本选择技术可以有效地应用于在线增量学习过程中，它具有与 RSVM 相似的正确率，而且其正确率随着所选择样本个数的增加而递增。

表 4.1 线性 PSVM 和 PSVM-I 分类器在数据集：Iris，Ionosphere，Tic-Tac-Toe 上的 10 折训练、测试正确率；

Data Set $m \times n$		Iris 150×4	Ionosphere 351×34	Tic-Tac-Toe 958×9
PSVM	Train	73.93%	90.76%	98.33%
	Test	71.33%	86.08%	98.33%
PSVM-I(0.1)	Train	76.52%	84.74	98.33%
	Test	74.67%	82.05	98.32%
PSVM-I(0.2)	Train	75.70%	89.43%	98.33%
	Test	73.33%	86.06%	98.33%
PSVM-I(0.3)	Train	73.93%	89.30%	98.33%
	Test	71.33%	86.63%	98.32%

表 4.2 在 4 个 UCI 公共数据集：Iris，Ionosphere，Tic-Tac-Toe 和 BUPA 上的 RSVM 的 10 折测试正确率，及利用增量学习算法和样本选择技术得到的 NPSVM-III 的 10 折测试正确率

Data Set $m \times n$	Iris 150×4	Ionosphere 351×34	Tic-Tac-Toe 958×9	BUPA Liver 345×6
RSWM	95.33%	92.31%	98.50%	74.86%
NPSVM-III (10%)	95.33%	93.44%	98.23%	68.69%
NPSVM-III (20%)	97.33%	95.15%	98.54%	75.65%
NPSVM-III (30%)	**98.00%**	**95.73%**	**98.64%**	**77.39%**

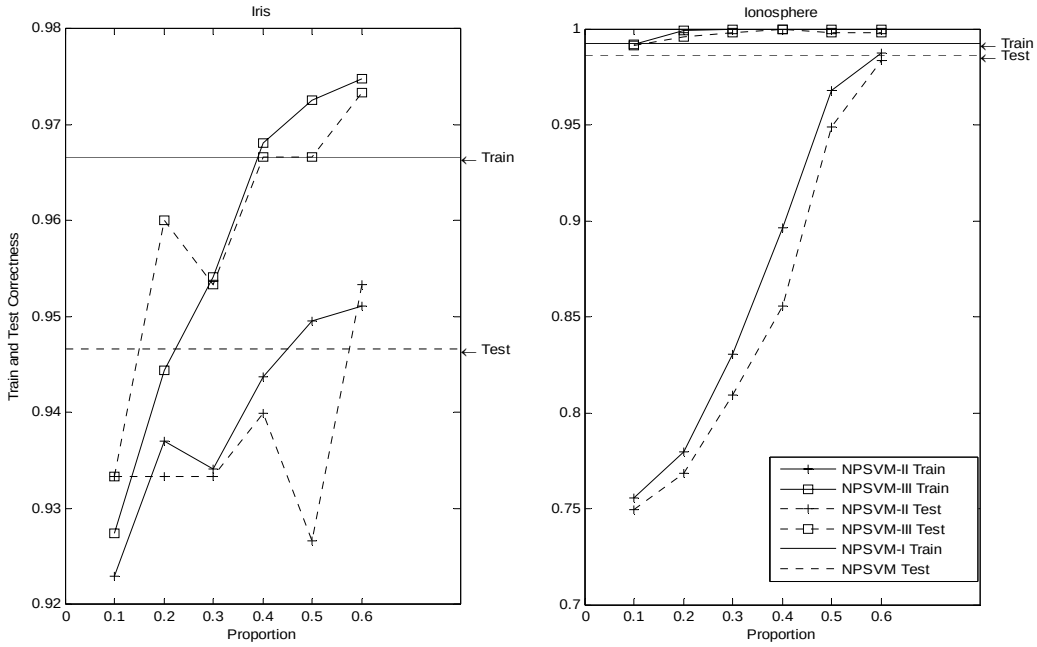

图 4.1 在数据集 Iris 和 Ionosphere 上 NPSVM-I、NPSVM-II 和 NPSVM-III 的 10 折训练、测试正确率

4.4 总结

利用上一节中提到的增量学习算法，我们可以有效的进行在线学习。但是该增量学习算法并没有解决 NPSVM 的空间复杂度问题，S-NPSVM 对空间的要求仍然是与样本个数的平方成正比，因此我们需要一种方法来处理大数据集在线学习的难题。

在本章，我们设计了一种根据训练结果对样本进行选择的方法，它试图选择出对于训练结果最具代表性 *num*（由用户指定）个样本。所选择出的样本与经典 SVM 中的支持向量的作用类似，对这些样本进行训练可以得到与原分类器类似的模型。这种样本选择技术可以应用到上一节中提到的增量学习方法中，当新数据到来时，仅仅选择历史样本的部分数据来进行合并，从而解决增量学习中大数据集对空间要求过大的问题。

对于经典 SVM 来说，样本选择非常简单，我们只需保留对偶变量值为非零的点（支持向量）即可。但是一般 PSVM 或 S-NPSVM 的松弛变量的值是非零的，这也就决定了 PSVM 不存在 SVM 中支持向量的概念。

在实验及理论分析的基础上我们推断出，在拟合平面附近的点相比离拟合平面较远的点具有更好的描述能力，我们可以利用这些点作为全部样本点的一个代表。在增量学习的过程中，对于历史数据我们的目的是选择出最具代表性的样本，而对于新到的数据，仅仅那些没有被已有分类器准确描述的样本点是有价值的因此我们同样可以根据样本点与拟合超平面之间的距离来对新数据进行压缩，即仅仅那些远离拟合超平面的点被考虑到总体数据集中。对于非线性分类器的描述要比线性分类器的描述复杂很多。为了处理这个问题，我们在上述基于距离选择出的样本的基础上，进一步把样本集中被错分的样本作为新样本一并添加到选择出的样本集中。由于错分样本的个数通常比较少，利用增量学习公式该方法导致的代价增加一般较小。

实验显示上述样本选择技术仅需付出较小时间代价，就可以有效的处理大样本集的在线学习问题，而且可以得到较好的训练结果。

第五章 新的非线性支持向量机学习模型 —— Extreme SVM

现有的 SVM 学习算法几乎都是通过使用核函数的方法来处理非线性分类器的学习，可以将核函数所起的作用分成两步来理解：1）将输入训练样本点映射到一个高维的特征空间中（包括无穷维），期望训练数据集在高维的特征空间中有更好的线性可分的可能性，映射所采用的映射函数是隐含在核函数中的；2）核函数以输入空间中的两个训练样本点作为输入，输出它们在特征空间中的点积。由于 SVM 算法的学习过程仅需要用到训练样本点的点积，所以我们可以通过核函数对训练数据集在特征空间中的 SVM 线性分类器（即原空间中的非线性分类器）进行学习。

本章，通过显式地构造一个非线性映射函数代替核函数将训练样本点映射到特征空间中，设计了一种新的非线性支持向量机（SVM）学习算法：Extreme SVM（ESVM）。该非线性映射函数同输入权重随机生成的 SFLNs 中的隐层神经元将输入样本点映射到隐层输出向量的作用类似。

从理论的角度来说，ESVM 是一种基于正则化最小二乘法的分类学习算法，可以被理解为一种特殊形式的正则化网络，它可以提供比神经网络算法（Extreme Learning Machine - ELM）更好的泛化能力；而且对它的训练仅需要一个求解一个阶数与训练样本个数无关的线性方程组，因此相比其它非线性 SVM 算法，ELM 更加的简单、快速。实验结果证实了我们的算法在保持正确率的前提下，可以显著的降低非线性 SVM 的训练时间；而且具有比 ELM 更好的推广能力。

5.1 引言

过去的研究结果表明，通过随机生成单隐层神经网络（Single hidden Layer Feedforward Networks - SLFNs）隐层神经元的输入权重和使用几乎任意的非零激活函数，SLFNs 可以一致地拟合任意的紧密输入集上的连续函数（[Sartori 91], [Huang 03]）。

在神经网络学习能力研究成果的基础上，极限学习机（Extreme Learning Machine - ELM）[Huang 04]根据某一概率分布随机生成 SFLNs 隐层神经元的输入权重（简称输入权重），使得隐层神经元的输出权重（简称输出权重）可以通过一个线性方程组的最小二乘解容易地获得，而不需要像传统的 SFLNs 算法那样进行迭代计算。实验表明 ELM 的运行速度要比其他传统的 SLFNs 算法（如 BP）算法快上千倍[Huang 04]，并且避免了局部极小值等问题。但是 ELM 是一种基于经验风险最小化（Empirical Risk Minimization - ERM）的算法，根据统计学习理论（[Vapnik 95], [Vapnik 98], [Bertero 86]），该类算法容易得到过拟合的模型。

在这篇文章中，对 2 类非线性分类器学习问题，通过利用输入权重随机生成的 SFLNs 显示地构造一个映射函数，我们设计了一种新的非线性支持向量机（SVM）分类器，它提

供了比 SFLNs 的 ELM 算法更好的控制模型复杂度的能力，具有更好的推广能力。该算法可以看成正则化网络（[Tikhonov 97], [Evgeniou 00a], [Evgeniou 00b]）的一种特殊的形式：与经典的 SVM 通过未知类别的样本点在划分超平面的正负两侧对其进行分类不同，该算法首先对两类样本点构造两个平行的拟合超平面，然后在分类时根据未知类别的样本点与哪条拟合超平面的距离较近来实现分类的目的，其它采用这种构造分类器方法的 SVM 算法包括 PSVM [Fung 01]，Multisurface PSVM [Mangasarian 06]，Least Squares SVM [Suykens 99]等。

但是，所有其它 SVM 非线性分类器的训练都使用了一个核函数来实现输入样本点在特征空间中的点积；而且算法的求解都需要计算一个大小与样本点个数平方成正比的矩阵的逆，因此当样本数量稍大（数千个）时，上述算法存在严重的空间复杂度过大的问题。与此不同，这本章中，我们利用利用输入权重随机生成的 SFLNs 显式构造的非线性映射函数，显示地计算两个训练样本点在特征空间中的点积；并且在此基础上设计了一种更加简单快捷的非线性支持向量机训练算法，对它的训练只需求解一个大小与样本点个数无关的线性方程组（通常阶小于 200）。从实验结果中我们可以看到与它具有其他 SVM 学习算法相似的测试正确率，但是具有更快的速度。由于其具有极快的速度，在本篇论文里，我们将该算法称为极限支持向量机（Extreme SVM - ESVM）。

本章的内容按照如下的方式组织：在第 2 节，我们将介绍 SLFNs 的体系结构，并且对 ELM 算法进行简单的回顾；在第 3 节，我们将详细的介绍 ESVM 算法，并将它与其它相关的理论进行比较；最后在第 4 节中，通过在标准分类问题测试数据集上的实验结果表明：ESVM 具有比 ELM 更好的泛化能力，而且在保持正确率的前提下显著降低了非线性 SVM 学习算法的训练复杂度。

为了叙述的方便，首先把本章将用到的数学表示符号声明如下：这里所有的向量默认均为列向量，上标'表示转置，向量 x 和 y 在空间中的点积表示为：$x'\cdot y$，向量 x 的 2-norm 表示为 $\|x\|=\sqrt{x'\cdot x}$；这里我们利用矩阵 $A[m\times n]$ 表示 n 维空间 R^n 中的 m 个训练样本；对角线元素为 ± 1 的对角矩阵 $D[m\times n]$ 的对角线上的元素声明了 m 个训练样本的类别是 $+1$ 或 -1；对于矩阵 $A\in R^{m\times n}$ 和 $B\in R^{n\times l}$，核函数 $k(A,B)$ 将 $R^{m\times n}\cdot R^{n\times l}$ 映射到 $R^{m\times l}$；默认情况下我们将使用下面的 Gaussian 核：

$$\left(K(A,B)\right)_{i,j}=\varepsilon^{-\mu\left\|A_i'-B_j\right\|^2},i=1...m,j=1...l$$

其中 μ 是正常量，ε 表示自然对数的底；e 为元素为 1 的任意维向量（维度根据上下文确定）；w,r 分别为分类超平面的方向系数和偏置，y 为松弛向量，参数 $v\geq 0$ 用于控制模型复杂度和正确率之间的平衡；I 表示单位向量。

5.2 相关工作回顾

本节我们将对本章中将用到的单隐层神经网络（SLFNs）的体系结构和 ELM 算法做一些简单的介绍。

5.2.1 单隐层前馈神经网络

这里令训练数据集为包含 m 个输入有序对的集合：$\{a_j, d_j\}, 1 \le j \le m$，其中 $a_j \in R^n$ 是第 j 个输入训练向量，$d_j \in R^{\tilde{n}}$ 是该输入向量的目标输出向量。那么一个单层神经网络的输出可以表示为：

$$O^1 = G\left(A^1 W^1\right) \ . \tag{5.1}$$

其中 $A^1 := \left[a_1^1, ..., a_m^1\right]' \in R^{m\times(n+1)}$ 为输入矩阵，a_i^1 是由输入向量 a_i 加上单位 1 构成的向量，即 $a_i^1 = [a_i', 1]'$；$O^1 := \left[o_1^1, ..., o_m^1\right]' \in R^{m\times\tilde{n}}$ 为神经元的输出矩阵，$o_i^1 \in R^{\tilde{n}\times 1}, 1 \le i \le m$ 为对应输入向量 a_i 的神经元输出向量；$W^1 := \left[w_1^1, ..., w_m^1\right]' \in R^{(n+1)\times\tilde{n}}$ 为神经网络的权重矩阵。函数 $G(Z)$ 接受一个矩阵 Z 作为输入，输出是利用神经网络的激活函数 $g(x)$ 作用到矩阵 Z 的每一个元素 z_{ij} 上后产生的与 Z 同等大小的矩阵。

多层神经网络包含多个平行的神经元层次，相互之间利用带有权重的前馈方式连接在一起。利用 $\#k$ 来表示多层神经网络第 k 层所包含的神经元的个数，第 k 层的输出矩阵可以如下表示：

$$O^k = G\left(A^k W^k\right) \ . \tag{5.2}$$

与单层神经网络的情形类似，这里 $A^k := \left[a_1^k, ..., a_m^k\right]' \in R^{m\times(\#(k-1)+1)}$ 为输入矩阵，a_i^k 是由前一层（$k-1$ 层）神经元的输出向量加上单位 1 构成的第 k 层向量即 $a_i^1 = \left[a_i^{k-1}', 1\right]'$；$O^1 := \left[o_1^1, ..., o_m^1\right]' \in R^{m\times\#k}$ 为神经元的输出矩阵，$o_i^k \in R^{\#k\times 1}, 1 \le i \le m$ 为对应输入向量 a_i 的第 k 层神经元的输出向量；$W^k := \left[w_1^k, ..., w_{\#k}^k\right]' \in R^{(\#(k-1)+1)\times\#k}$ 为神经元的权重矩阵。

采用类似的表示方式，单隐层神经网络（SLFNs）的隐层神经元输出向量可以由（5.1）式表示。输出层神经元的输出这里我们将表示为如下的形式：

$$O^2 = A^2 W^2, \ . \tag{5.3}$$

其中

$$A^2=\left[O^1,e\right]=\left[G\left(A^1W^1\right),e\right]\ . \tag{5.4}$$

本章，我们称A^2为隐层神经元输出矩阵，W^1,W^2分别命名为 SLFNs 的输入、输出权重。

研究结果表明，包含$m-1$个隐层神经元的单隐层神经网络可以拟合任意的包含m个输入点的训练集（[Baum 88], [Nilsson 65], [Huang 91]）。更进一步，文献([Sartori 91], [Huang 03], [Huang 04])指出在不影响神经网络拟合能力的前提下，随机生成带有N个隐层神经元的 SLFNs 的隐层神经元的输入权重，能够以任意小的误差拟合N个输入样本，这意味着神经网络的输入权重W^1可以随机产生，而不必利用传统的算法（如 BP）训练。

5.2.2 极限学习算法

考虑如下的 2 类分类问题：训练数据集包含m个分布在n维空间R^n上数据点，这里用矩阵$A=\left[a_1^{'}...a_m^{'}\right]\in R^{m\times n}$来表示，其中$a_i\in R^n$表示第$i$个输入样本；样本点$a_i$的类别信息由一个对角矩阵$D\in R^{m\times m}$对角线上的元素$d_{ii}$（$+1$或$-1$）指定，即若$d_{ii}=+1$则$a_i$为正类，反之则为负类。特别注意到对于两类分类问题，SLFNs 根据输出层神经元输出值的正负来做出分类的决定，因此只需要一个神经元就足够了，即$\#2=1$。

在(([Sartori 91], [Huang 03])的理论基础之上，极限学习算法（Extreme Learning Machine - ELM）随机地确定输入权重（矩阵W^1），并将求解 SLFNs 输出权重的问题等价于求解下述优化问题：

$$\min_{W^2}F\left(W^2\right)=\left\|A^2W^2-De\right\|^2. \tag{5.5}$$

其中A^2同（5.4）的定义。

ELM 的核心思想在于不再像传统的 SLFNs 训练算法那样迭代地训练神经网络全部的参数，取而代之随机地生成输入权重矩阵W^1。从式（5.5）可以看出 ELM 算法的训练，等价于由矛盾线性方程组$A^2W^2=De$的极小范数二乘解确定输出权重矩阵W^2，这可以通过隐层输出矩阵A^2的伪逆简单地求得：

$$\hat{W}^2=A^2\ De. \tag{5.6}$$

其中A^2 是隐层神经元输出矩阵的伪逆[Serre 02]。

从表达式（5.6）我们可以看出 ELM 的目标是极小化表达式$A^2W^2=O^2$的经验风险，虽然 ELM 得到的是所有最小二乘解当中模最小的解，它对模型复杂度的控制仍然比较弱，

根据 Vapnik 的理论（[Vapnik 95],[Vapnik 98]），ELM 算法可以看成一种基于经验风险最小化（Emprical Risk Minimization - ERM）原则的算法，容易产生过拟合的模型。从第四节的测试结果我们可以看出，当隐层节点数相对较多时，这种泛化性能比较差的现象尤其明显。

我们可以将 ELM 学习 SLFNs 的过程理解为两个步骤：首先，随机地生成 SLFNs 输入权重，将输入训练样本映射到隐层神经元的输出向量；然后，利用隐层神经元输出向量通过公式（5.6）求得 SLFNs 输出权重w^2的极小范数极小二乘解。

基于这种观察，下一节设计了一种新的基于非线性 SVM 分类器—Extreme SVM（ESVM）。与以往 SVM 算法利用核函数实现非线性分类器的方法不同，ESVM 首先利用输入权重随机生成的 SLFNs 的隐层神经元将输入训练样本映射到一个特征空间中；然后在该特征空间中执行一个基于正则化最小二乘法的线性分类器算法。从理论上分析，ESVM 算法是一种基于结构风险最小化（Structural Risk Minimization - SRM）原则的算法，比 ELM 具有更好的泛化性能，而且从测试结果我们可以看出，在正确率相当的情况下，ESVM 的训练时间明显低于其他 SVM 算法。

5.3 极限支持向量分类器——Extreme SVM

这一节，我们将介绍一种新的非线性支持向量机学习算法——极限支持向量机（Extreme SVM - ESVM）；并将 ESVM 与其它相关理论、算法进行比较。

5.3.1 线性极限支持向量机分类器

线性 ESVM 与线性 PSVM [Fung 01] 具有相同的形式，但是为了表述的方便我们将相关的公式作如下的推导说明。

对于上一节中提到的两类分类问题，线性ESVM试图找到超平面$x'w-r=\pm 1$，其中w,r分别是斜率和相对于原点的偏移，这里其不再是分隔超平面而是最邻近平面，在它周围分布着大多数的点。ESVM可以用如下的等式约束二次优化来表示：

$$\begin{aligned}\min_{(w,r,y)\in R^{m+1+m}} \; & v\frac{1}{2}\|y\|^2+\frac{1}{2}(w'w+r^2) \\ s.t. \quad & D(Aw-er)+y=e,\end{aligned} \tag{5.7}$$

其中v是一个正参数。表达式（5.7）将标准支持向量机[Vapnik 95]的不等式约束变为了等式约束,这个变化虽然很简单，但是确极大地简化了求解的步骤：我们可以给出一个显式的解析解，而对于标准支持向量机来说这是不可能的。可以看出（5.7）与线性 PSVM（3.2）式具有相同的形式。

对于某一未知类别的样本x，线性 ESVM 的分类器可以表示成如下形式：

$$x'w-r\begin{cases}>0, \text{ then } x\in A+;\\ <0, \text{ then } x\in A-;\\ =0, \text{ then } x\in A+ \text{ or } x\in A-;\end{cases}. \tag{5.8}$$

下一节通过在非线性映射函数衍生的特征空间中求解线性 ESVM 模型（5.7），我们构造了非线性的 ESVM 模型。

5.3.2 非线性极限支持向量机分类器

为了得到非线性 ESVM 分类器，我们首先利用一个非线性的转换函数 $\Phi(x): R^n \to R^{\tilde{n}}$ 将 R^n 空间中的输入样本点映射到一个有限维度的特征空间 $R^{\tilde{n}}$ 中；然后在该特征空间中利用线性 ESVM 的二次优化求得原输入空间中的非线性分类器；具体地非线性极限支持向量分类器的求解可以表示为如下的等式二次规划问题：

$$\begin{aligned}&\min_{(w,r,y)\in R^{\tilde{n}+1+m}} v\frac{1}{2}\|y\|^2+\frac{1}{2}(w'w+r^2)\\ &s.t. \quad D(\phi(A)\mathrm{w}-er)+y=e,\end{aligned}. \tag{5.9}$$

本章稍后将详细介绍(5.9)式中的映射函数：$\Phi(x): R^n \to R^{\tilde{n}}$，类似的模型我们在[Liu 07a]中也曾经使用过。优化问题（5.9）的 Lagrangian 对偶函数可以表示为：

$$L(w,r,y,u)=\frac{v}{2}\|y\|^2+\frac{1}{2}\left\|\begin{bmatrix}w\\ r\end{bmatrix}\right\|^2-s'(D(\Phi(A)w-er)+y-e). \tag{5.10}$$

其中 $s\in R^m$ 是优化问题(5.9)中等式约束的Lagrangian对偶乘子。令Lagrangian等式(5.10)对 (w,r,y,s) 的导数分别等于零，可以给出如下 KKT 最优条件：

$$\begin{aligned}&w=\Phi(A)'Ds,\\ &r=-e'Ds,\\ &vy=s,\\ &D(\Phi(A)w-er)+y-e=0\end{aligned}. \tag{5.11}$$

将（5.11）中前 3 个等式替换到最后一个等式中，我们可以得到如下对偶变量 Ds 的显式解析解：

$$Ds=(\frac{I}{v}+\phi(A)\phi(A)'+ee')^{-1}De=(\frac{I}{v}+E_\Phi E_\Phi')^{-1}De. \tag{5.12}$$

其中 $E_\Phi=[\Phi(A),-e]\in R^{m\times\tilde{n}}$。

几乎所有以往的非线性 SVM 算法都利用了一个计算输入向量 x, y 在某一特征空间

中点积的核函数$K(x',y)$（如 RBF 核，多项式核等），来计算式（5.12）中的输入样本矩阵的乘积：$\phi(A)\phi(A)'$。但是核函数采用的是哪种非线性转换函数$\Phi(x)$，以及该非线性映射函数的相关性质是未知的。

与此不同，如第二节最后所述，这里我们将利用输入权重随机生成的 SLFNs 的隐层神经元显式地构造一个随机非线性映射函数$\Phi(x)$，从而显示的计算$\phi(A)\phi(A)'$。具体地转换函数$\Phi(x):R^n \to R^{\tilde{n}}$可以表示为如下的形式：

$$\begin{aligned}\phi(x) &= G(W^1x^1) \\ &= \left(g\left(\sum_{j=1}^{n} W_{1j}^1 x_j + W_{1(n+1)}^1\right),...g\left(\sum_{j=1}^{n} W_{\tilde{n}j}^1 x_j + W_{\tilde{n}(n+1)}^1\right)\right).\end{aligned} \tag{5.13}$$

其中$x \in R^n$是输入向量，$x^1 = [x',1]'$，$W^1 \in R^{\tilde{n}\times n}$是一个元素按照某一非平凡的分布随机产生的矩阵，激活函数$G(.)$具有与（5.1）中激活函数同样的定义。注意，这里我们可以将x_1和W^1分别理解为前面提到的 SLFNs 中的输入向量和输入权重，$\phi(x)$是隐层神经元的输出向量。

从式（5.12）可以看出对偶变量Ds解的表达式里，包含求解一个大（$m\times m$，m是样本点的个数）矩阵逆的运算，因此仍然存在空间复杂度过大的问题。我们可以通过利用 SMW [Golub 96]公式对表达式（5.12）进行变换，得到对偶变量Ds解新的表达式：

$$Ds = v(I - E_\phi(\frac{I}{v} + E_\phi' E_\phi)^{-1} E_\phi')De\,. \tag{5.14}$$

注意，如果我们将式（5.14）代入到 KKT 条件式（5.11）中，就可以得到如下关于$[w,r]$的解析解：

$$[w,r]' = \left(\frac{I}{v} + E_\Phi' E_\Phi\right)^{-1} E_\Phi' De \tag{5.15}$$

式（5.15）仅需对一个$(\tilde{n}+1)\times(\tilde{n}+1)$维的矩阵求逆，而$\tilde{n}$是特征空间的维度，其取值一般会比较小（通常小于 200）并且独立于训练样本点的个数m。

对一个未知类别的测试样本点x，采用映射函数$\phi(x)$的非线性 ESVM 分类器，可以由下式表示：

$$\phi(x)'w-r\begin{cases}>0, \text{ then } x\in A+;\\ <0, \text{ then } x\in A-;\\ =0, \text{ then } x\in A+ \text{ or } x\in A-;\end{cases} \tag{5.16}$$

与线性 ESVM 分类器（5.8）相比，非线性 ESVM 分类器首先将待测试点 x 映射到特征空间中的向量 $\phi(x)$，然后对向量 $\phi(x)$ 进行分类。

现在我们可以给出非线性 ESVM 的算法：

算法：极限支持向量分类机 已知 $m\times n$ 维的矩阵 A 给定了包含 m 个 n 维空间中的训练样本点；对角矩阵 $D\in R^{m\times m}$ 的对角元素给出了这些样本的类别信息（正类为 +1，否则为 −1）。我们可以按照如下的步骤生成非线性的 ESVM 分类器：

1. 按照某一非平凡概率分布随机生成矩阵 $W^1\in R^{\tilde{n}\times(n+1)}$，并选择几乎任意的一个非线性函数作为激活函数 $g(.)$（最常用的如 Signum 函数）构造映射函数 $\phi(x)$（5.13）；
2. 定义 $E_\phi=\left[\phi(A),-e\right]$，其中 e 是一个由单位1构成的 $m\times 1$ 维的向量；
3. 选择参数 ν 的值，利用公式（5.15）求得参数 $\begin{bmatrix}w\\ r\end{bmatrix}$ 的解；
4. 利用公式（5.16）对某个未知类别的点 x 分类；

注意，对于非常巨大的训练集（数万条数据），我们可以将训练集 A 划分为多个部分：$A_i, 2<i<m$，并令 $E_{\Phi i}=\left[\Phi(A_i),e\right]$，式（5.15）可以按照如下的方式进行增量计算：

$$E_\Phi'E_\Phi=\sum E_{\Phi i}'E_{\Phi i}, E_\Phi'De=\sum E_{\Phi i}'D_ie \tag{5.17}$$

5.3.3 极限支持向量机与正则化网络的关系

极限支持向量机（ESVM）可以看成一种特殊形式的正则化网络（Regularization Network - RN）。从式（5.8）和（5.16）可以看出，在 ESVM 中超平面 $x'w-r=\pm 1$ 和 $\phi(x)'w-r=\pm 1$ 不再是两类样本点的划分面（标准 SVM 的情形），而是对两类样本点的拟合超平面，在它周围分布着大部分的样本点，而分类时则根据测试样本点 x 与哪条拟合超平面较近做出决定。因此，ESVM 是利用拟合回归（输入是训练样本点，每个样本点的目标输出是 +1 或 −1，根据其所属的类别确定）的方法来构造分类器，采用类似分类器构造方法的算法包括 PSVM [Fung 01a]，LSSVM [Suykens 99]。

从稀疏数据构造拟合函数是一个经典的不适定问题，典型的处理这种问题的方法是

正则化理论([Tikhonov 97], [Bertero 88], [Bertero 86])。正则化理论将不适定的拟合问题形式化为寻找最小化下面表达式的泛函 f ，这样一个变元问题：

$$\min_{f\in \mathrm{H}} : \frac{1}{l}\sum_{i=1}^{m} V\left(D_{ii}, f\left(x_i\right)\right) + \lambda \left\| f \right\|_K^2 \tag{5.18}$$

其中$V\left(.,.\right)$代表损失函数，$\left\| f \right\|_K^2$是在正定核函数K定义的再生核 Hilbert 空间H中的范数，λ是正则化参数 [Wahba 90]。对比 ESVM 的优化问题（5.9）和公式（5.18）我们可以看出 ESVM 是正则化理论（5.18）的一种具体化的形式：在（5.9）中损失函数$V\left(.,.\right)$被定义成平方误差的形式，正定核函数K被定义为对输入向量非线性映射后在特征空间中的点积的形式，即：$K\left(x, y\right) = \phi\left(x\right)^{'} \cdot \phi\left(y\right)$。

文献[Evgeniou 00a]、[Evgeniou 00b]指出，正则化网络（Regularization Network - RN）提供了控制模型复杂度的能力而且，如同支持向量机，也可以由结构风险最小化（SRM）原则推出。因此根据 Vapnik 的理论 ([Vapnik 82], [Vapnik 95], [Vapnik 98])，我们可以期望，ESVM 的学习结果不仅能够很好的适用于训练数据集，而且对于未知类别的测试数据也有很好的预测能力。

5.3.4 极限支持向量机与非线性 PSVM 之间的关系

如 5.3.1 和 5.3.2 节所述，线性 ESVM 与线性 PSVM [Fung 01] 具有相同的形式，但它们却具有不同的非线性分类器优化问题。

在[Fung 01]中非线性 PSVM 具有如下的形式：

$$\begin{aligned} &\min_{(u,r,y)\in R^{n+1+m}} \; v\frac{1}{2}\| y \|^2 + \frac{1}{2}(u'u + r^2) \\ &s.t. \quad D(K\left(A, A^{'}\right) - er) + y = e, \end{aligned} \tag{5.19}$$

同线性 PSVM 的公式（3.2）相比，（5.19）将优化目标变量w替换为它的对偶变量：$w = A^{'}Du$，并将向量点积（线性核）替换为非线性的核函数$K\left(A, A^{'}\right)$。通过式（5.19）的 KKT 最优充分必要条件，我们可以得到对偶变量Ds的显示解：

$$Ds = \left(\frac{1}{v}I + KK^{'} + ee^{'}\right)^{-1} De. \tag{5.20}$$

比较非线性 ESVM 的解（5.12）和非线性 PSVM 的解（5.20）我们可以看出，（5.12）不需要核矩阵的乘积：$K \cdot K^{'}$；由于核矩阵K是一个$m \times m$（其中m是训练数据集中样本点的个数）的矩阵，SMW 公式对（5.20）不起作用，因此（5.20）势必包含求解一个很大的$m \times m$维矩阵的逆，这就遇到了与经典 SVM 类似的空间复杂度问题。与之不同的

是 ESVM 的解（5.12）仅需求解一个大小为 $\tilde{n}\times\tilde{n}$ 的矩阵的逆（$\tilde{n}$ 是特征空间的维度，它独立于样本点的个数。实验结果表明，在保持正确率相当的情况下，$\tilde{n}$ 通常可以远小于 m）。

5.3.5 极限支持向量机与极限学习机之间的关系

如上面所述，我们可以把 ESVM 与 ELM 的学习过程分为两个步骤：首先，输入训练数据集中的样本点被 SLFNs 的隐层神经元（ELM）或非线性映射函数 $\phi(\cdot)$（ESVM）映射到某一特征空间中；然后，在该特征空间中执行相应的线性算法。进一步的，可以看出 ESVM 中的映射函数（5.13）构造方式同 SFLNs 中的隐层神经元是类似的。但是 ELM 与 ESVM 的学习过程是非常不同的。

ESVM 的解是等式 $D\left(\phi(A)^{'}w-er\right)=e$ 的正则化最小二乘解，而 ELM 的解是式（5.5）的极小范数最小二乘解。ELM 的优化目标（5.5）式中的 A^2 可以看成由 ESVM 中的 $\left[\phi(A),e\right]$ 构成，并且 $W^2=\left[w^{'},-r^{'}\right]^{'}$。

如前所述，ELM 试图极小化 SLFNs 在训练数据集上的经验风险，对模型复杂度的控制很弱；因此，根据 Vapnik 的理论，ELM 容易得到过拟合的模型。ESVM 则通过使用正则化技术避免了过拟合的问题，实验结果也表明绝大部分时间里 ESVM 可以得到比 ELM 更好的泛化性能。

5.3.6 极限支持向量机与标准 SVM 之间的区别

ESVM 和 SVM 都可以看成是 Vapnik 的结构风险最小化（SRM）理论框架下的算法，它们之间主要有如下两方面的不同：

1. 与经典 SVM 不同，ESVM 基于正则化最小二乘方法构造拟合函数来实现分类器，非线性分类器的训练过程只需求解一个线性方程组（5.14）即可，非常的简单、快速，而其代价小的多。
2. 经典 SVM 利用一个核函数 K 来训练非线性分类器，该核函数对应的映射函数是未知的。而 ESVM 则显式地构造了一个映射函数（5.13）$\phi: R^n \to R^{\tilde{n}}$，这使得 ESVM 的求解只需计算一个大小为 $\tilde{n}\times\tilde{n}$ 的矩阵的逆即可。

5.4. 测试结果

在这一节中，我们将在 8 个机器学习分类领域的公共测试数据集上，将本章提出的 ESVM 算法与经典 SVM 算法、非线性 PSVM 算法及 ELM 算法的性能进行比较。本节中，所得的 ESVM 和 ELM 算法的结果数据的测试环境为：MATLAB 7.0，2.80GHz Pentium IV CPU，512 内存；我们在同样的机器上使用标准 SVM 的 C 语言工具包：LIBSVM 来模拟经典 SVM 的性能。实验中，经典 SVM 使用的核函数是高斯径向基核函数：

$\exp\left(-\|x-y\|^2/\delta^2\right)$，ESVM 和 ELM 中的激活函数使用的是简单的 Sigmoidal 函数：$g(x)=1/\left(1+\exp(-x)\right)$。为了比较的公平，本节在比较 ESVM 和 ELM 算法时，ESVM 中特征空间的维度$\tilde{n}$和 ELM 算法中 SFLNs 的隐层神经元的个数保持一致。

本节的数值实验所使用了 8 个来自机器学习公共测试数据库 UCI [Murphy 92]，Stalog 和 Delve 中的分类问题数据集，包括：australian，breast-cancer，diabetes，heart，ionosphere，liver-disorder，sonar，splice 等。

我们对所得到的测试数据结果，作如下的说明：

1. **图 5.1 至图 5.4：ESVM 与 ELM 在 8 个公共测试数据集上的泛化性能的比较。**在图 5.1 至 5.4 中，我们在 8 个公共数据集上比较了 ESVM 和 ELM 算法的测试正确率。这里的测试正确率的结果均为 10 次 10 折测试结果的平均值（随机选取整个数据集中 10%的数据作为测试数据），ESVM 中的参数v由 10 折交叉确认确定。从图中可以明显地看出：在全部的 8 个测试数据集上，ESVM 的泛化性能要明显优于 ELM，尤其是当隐层节点数较多时，ESVM 对分类器的泛化性能有较好的保证。从图中我们还可以明显的看出，随着隐层结点个数$\tilde{n}$由低到高逐步变化，ELM 算法所得的测试正确率首先逐步增高（由于学习能力的增强），当$\tilde{n}$在某一区域取值时到达峰值，然后逐步降低（由于学习能力过强导致过拟合现象的出现）。
2. **表 5.1：ESVM 与经典 SVM，非线性 PSVM 在 8 个公共数据集上的训练、测试正确率、时间的比较。**在表 5.1 中我们利用同样的 8 个公共数据集，对 ESVM 算法、经典 SVM 算法[Vapnik 95]和非线性 PSVM 算法[Fung 01]的训练时间，训练、测试正确率等指标进行比较。这里算法中的参数设置均通过十折交叉确认确定；所有的测试结果（包括召回率，测试正确率，训练时间等）均为 10 次 10 折测试的平均结果。进一步，为了更好的观察 ESVM 的特性，我给出了当特征空间维度$\tilde{n}$取不同值时 ESVM 算法的测试结果数据。从表中我们可以看出 ESVM 具有与经典 SVM，非线性 PSVM 相似的正确率，但明显减小了非线性分类器的学习时间。特别的，对于 splice 数据集由于非线性 PSVM 的空间复杂度太大，没有给出相应的测试数据。

5.5 总结

在本章中我们提出了一种新的基于正则化最小二乘的非线性支持向量机学习算法——Extreme SVM（ESVM）。与经典 SVM 利用核函数训练非线性分类器不同，ESVM 利用输入权重随机生成的 SLFNs 显示的构造了一个非线性转换函数$\Phi(x): R^n \to R^{\tilde{n}}$，并在此转换函数对应的特征空间中应用线性 ESVM 分类器算法得到原空间的非线性分类器。ESVM 的训练仅需求解一个较小的线性方程组，其阶数与训练样本数无关；从而避免了标准 SVM 优化问题求解算法的复杂度问题。

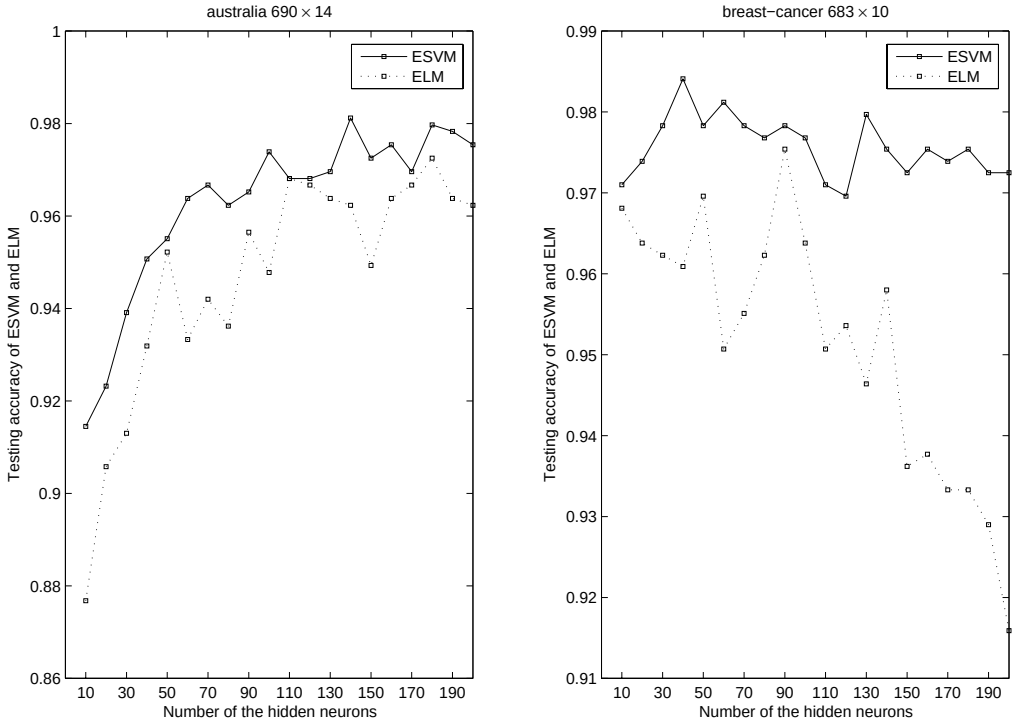

图 5.1 ESVM 特征空间的维度（ELM 隐层节点数）由 10 变化到 200 时，在 australia，brest-cancer 上的测试正确率

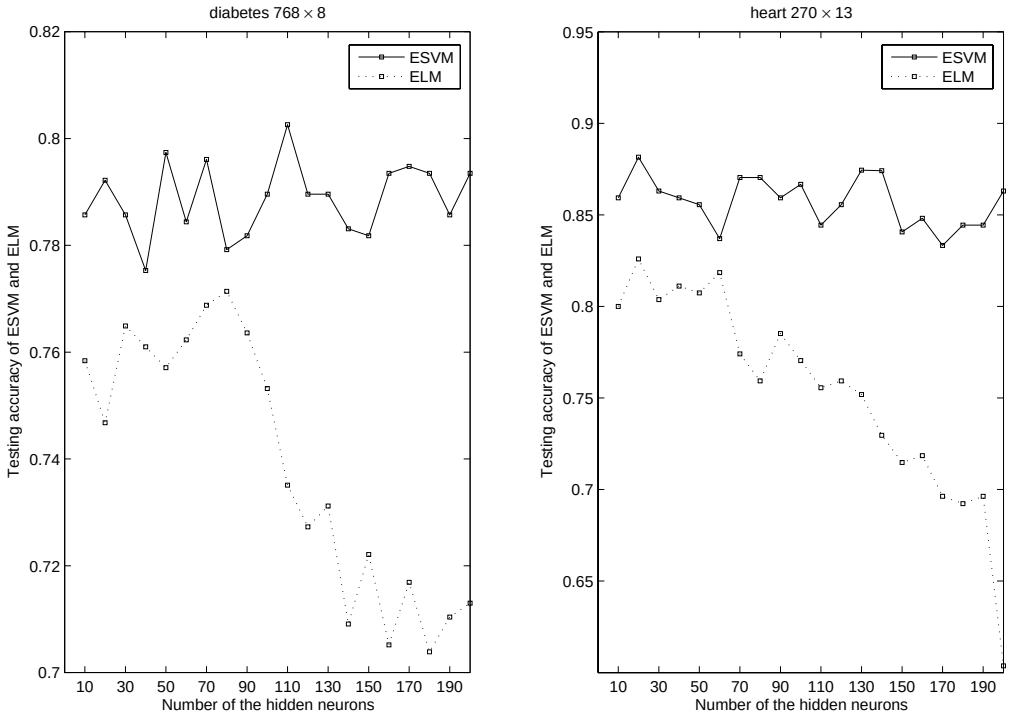

图 5.2 ESVM 特征空间维度（ELM 隐层节点数）由 10 变化到 200 时，在 diabetes，heart 上的测试正确率

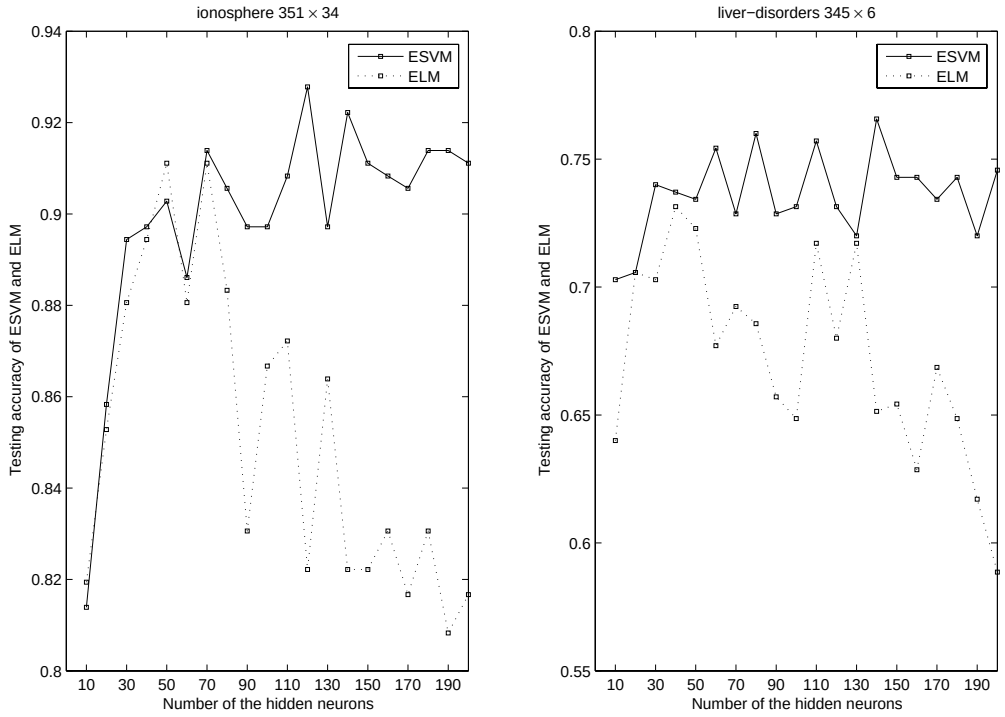

图 5.3 ESVM 特征空间维度（ELM 隐层节点数）由 10 变化到 200 时，在 ionosphere，liver-disorders 上的测试正确率

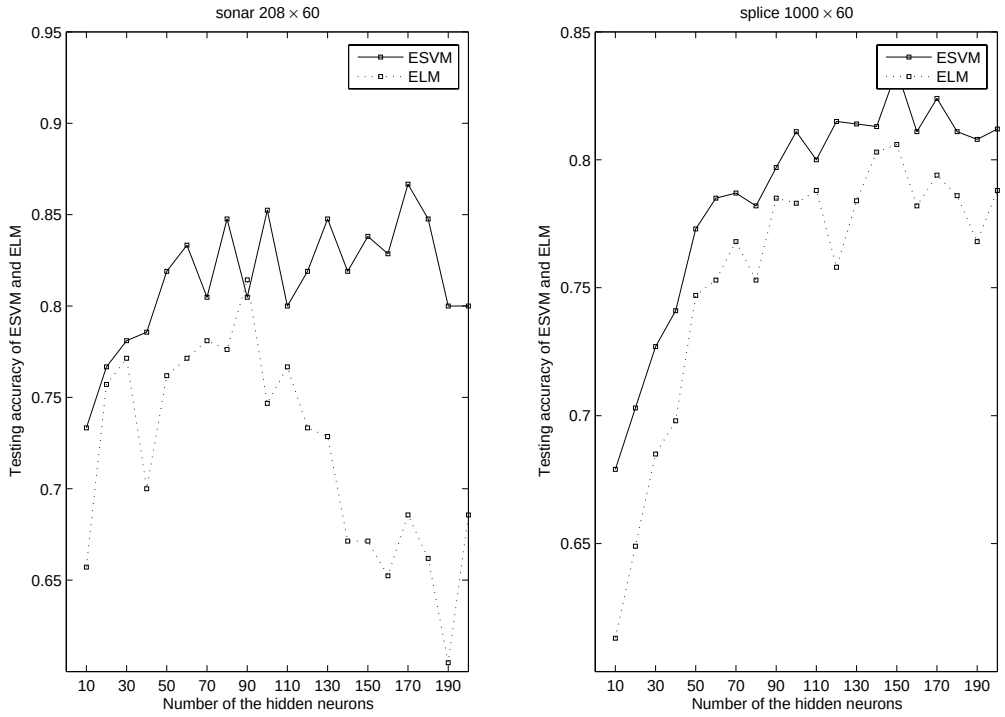

图 5.4 ESVM 特征空间维度（ELM 隐层节点数）由 10 变化到 200 时，在 sonar，splice 上的测试正确率

表 5.1 ESVM，SVM，NPSVM 在 8 个数据集上的训练、测试正确率及训练时间比较

Datasets	ESVM Train Test Time						SVM Train Test Time	NPSVM Train Test Time
	20	60	100	140	180	200		
Australia	91.26%	96.09%	97.86%	98.42%	99.19%	99.28%	92.59%	100%
690 × 14	92.32%	96.38%	97.39%	98.12%	97.97%	97.54%	83.91%	96.52%
	0.0047	0.0141	0.0219	0.0469	0.0703	0.0828	0.1703	0.3297
breast-cancer	97.08%	97.54%	97.77%	97.74%	97.92%	97.85%	96.73%	97.48%
683 × 10	97.39%	98.12%	97.68%	97.68%	97.54%	97.25%	96.63%	97.73%
	0	0.0125	0.0281	0.0453	0.0672	0.0781	0.125	0.3281
diabetes	78.17%	80.46%	79.15%	80.81%	79.83%	85.33%	77.47%	79.15%
768 × 8	79.22%	78.44%	78.96%	80.81%	79.83%	85.33%	75.78%	77.48%
	0.0078	0.0172	0.0313	0.0516	0.0766	0.0906	0.1689	0.4406
heart	85.76%	85.35%	88.72%	89.26%	90.12%	84.73%	96.75%	83.29%
270 × 13	88.15%	83.70%	86.67%	87.41%	84.44%	86.30%	75.56%	82.96%
	0.0047	0.063	0.0109	0.0219	0.0313	0.0344	0.0312	0.0297
ionosphere	85.62%	94.19%	96.67%	96.32%	94.19%	97.46%	100%	99.37%
351 × 34	85.83%	88.61%	89.72%	92.22%	91.39%	91.11%	92.02%	94.87%
	0.0031	0.0094	0.0156	0.0281	0.0344	0.0437	0.0610	0.0626
liver	75.13%	75.35%	77.23%	78.16%	76.32%	74.97%	80.58%	76.75%
345 × 6	70.57%	75.43%	73.14%	76.57%	74.29%	74.57%	72.49%	73.34%
	0.0016	0.0063	0.0156	0.0234	0.0359	0.0453	0.05	0.0581
sonar	81.18%	90.43%	90.91%	99.89%	99.57%	87.49%	100%	100%
208 × 60	76.67%	83.33%	85.24%	81.90%	84.76%	80%	74.04%	89.47%
	0.0016	0.0031	0.0141	0.0172	0.0313	0.0281	0.0405	0.0156
splice	68.31%	80.08%	83.99%	86.63%	88.44%	86.17%	100%	-
1000 × 60	70.30%	78.50%	81.10%	81.30%	81.10%	81.20%	56.9%	-
	0.0063	0.0234	0.0484	0.0703	0.1	0.1141	1.25	-

实验结果表明，ESVM 由于考虑了对模型复杂度的控制，因此可以得到具有比 ELM 的训练结果更好的泛化性能的分类器；而且在测试正确率相当的条件下，明显的降低了 SVM 算法的训练时间。

第六章 结束语

6.1 本文工作总结

分类是机器学习的一个重要任务和目标，是许多其它问题的基础，目前在研究和商业上的应用非常广泛。分类问题包括两个阶段：训练和预测。在训练阶段，分类算法从具有类别标记的训练实例中学到一个分类模型（称作分类器），期望该模型能把训练实例映射到给定的类别；在预测阶段，利用该分类器对没有类别标记的（测试）实例预测其类别，预测的准确程度可以评价分类器的性能。

Vapnik 等人从六、七十年代开始致力于统计学习理论方面研究，到九十年代中期，随着其理论的不断发展和成熟，也由于神经网络等学习方法在理论上缺乏实质性进展，统计学习理论开始受到越来越广泛的重视。统计学习理论(SLT)是一种小样本统计理论，着重研究在小样本情况下的统计规律及学习方法性质。该理论针对小样本统计问题建立了一套新的理论体系，在这种体系下的统计推理规则不仅考虑了对渐近性能的要求，而且追求如何在现有的有限信息条件下得到最优的结果。SLT 为解决有限样本学习问题提供了一个统一的框架。它能将很多现有方法纳入其中，并对可学习性、正确性、过学习和欠学习、局部极小点等问题取得了较好的结果。同时，在 SLT 的基础上发展了一种新的通用学习算法——支持向量机(SVM)，SVM 算法对小样本、非线性和高维数据具有很好分类性能，是目前分类问题上最流行的算法之一。但由于在计算上，SVM 借助二次规划求解分类器需要一个 $n \times n$ 维的内积矩阵（其中 n 是样本个数），所需要的计算开销是相当大的，从计算理论上分析，在个人计算机上，用 SVM 技术处理样本个数的规模界限一般为 4,000 个，因而解决海量数据的分析与处理是几乎不可能的。

本论文中的工作主要是关于支持向量机（SVM）的大数据集训练、增量学习及新的分类器模型的一些工作，我们为临近支持向量机（Proximal Support Vector Machine PSVM）的非线性模型设计了增量学习算法；然后为使其能够处理大数据集的在线增量学习，我们设计了样本选择技术；最后我们提出了一种新的非线性支持向量机分类算法——Extreme Support Vector Machine（ESVM），ESVM 具有速度快、扩展性能好等优点。具体的作如下的总结：

- **非线性 PSVM 的增量学习算法**：为了使 NPSVM 能够更加有效地进行在线增量学习，本文设计了一种新的增量学习算法。该增量学习算法基于一个新的非线性 PSVM 模型，由该模型的解的形式，我们可以利用分块矩阵求逆公式有效的利用 NPSVM 分类器的历史训练结果，减少在线学习过程中的重复学习，使得学习具有延续性。理论推导及实验结果显示在线学习过程中采用该增量学习算法不仅可以得到与批量学习相同的分类器、正确率；而且由于该增量学习算法去除了重复学习的问题，可以显著的缩短训练时间。
- **PSVM 的样本选择技术**：NPSVM 的空间复杂度是与样本个数的平方成正比的，为了

处理大数据集的增量学习问题，本文设计了针对历史数据、新数据和非线性分类器的样本选择技术。在线学习过程中，该样本选择技术不仅能够选择出历史数据集中最具代表性的样本点，而且能够选择出新数据中最具价值的样本点；此外对于难以描述的非线性分类器也特别设计了相应的样本选择方法。实验显示上述样本选择技术仅需付出较小的时间代价，就可以有效地处理大样本集的在线学习问题，而且可以得到与利用全部样本进行训练的结果相近的正确率。

- **一种新的非线性 SVM 学习方法——ESVM：** 本文设计了一种新的非线性 SVM 分类学习算法——Extreme SVM（ESVM）。ESVM 是以正则化最小二乘为基础的一种 SVM 分类器，与其它所有非线性 SVM 学习算法不同的是，ESVM 不是使用核函数来训练非线性分类器，而是显式地构造了一个非线性随机映射函数将输入样本点映射到一个特征空间中，然后在该特征空间中学习一个线性的分类器。该方法基于单隐层前馈神经网络（Single hidden Layer Feedforward Networks - SLFNs）的学习机制：在保持 SLFNs 学习能力的前提下，其输入权重可以随机地确定而不需要训练，这样 SLFNs 隐层神经元的作用相当于一个映射函数。理论分析及实验结果表明: ESVM 可以有效地应用于大数据集的训练，不仅具有与 SVM 相当的正确率，而且极大地缩短了训练时间。另外，与 SLFNs 的学习算法 ELM 相比，ESVM 将正则化理论引入到 SLFNs 的训练中，具有比 ELM 更好的泛化能力。

6.2 下一步研究方向

针对 SVM 所构造的分类器对数据分布的描述能力方面有很多的工作需要做；另一方面 ESVM 的相关问题也值得进一步深入研究。具体说来主要包括以下内容：

1、SVM 与基于覆盖的分类算法的结合问题：以“最佳划分”为目标的传统分类方法如 SVM 有很好的理论基础，但存在着训练代价过高、误识率高等缺点；而以“认识”事物为目的的基于覆盖的分类方法则可以比较好的解决这些问题，但其泛化性能缺乏数学性质上的保证。我们期望能找到两者之间的结合点，从而能够提升基于覆盖分类方法的预测能力。

2、极小样本集的相关性质：极小样本集控制着最终模型的形式及性能，我们期望能在极小样本集的一般概念，与 PAC 样本复杂度理论的关系等方向做一些尝试。

3、ESVM 相关问题的深入研究：我们可以看出 ESVM 具有良好的性质，但仍然有很多的涉及更加本质的工作值得进一步去做：

1. SLFN 中的激活函数、隐层神经元输入参数对分类器性能的影响，及其与核函数、核优化理论的关系；
2. ESVM 对大数据集分类能力的分析；
3. ESVM 中两步学习方式（即首先映射然后学习），可以推广到很多不同的（几乎任意）学习算法中用于学习非线性模型。

参考文献

[Tsang 07] I. W. Tsang, A. Kocsor, J. T. Kwok. Simpler core vector machines with enclosing balls. ICML, 2007

[Williams 01] C. K. I. Williams and M. Seeger. Using the Nyström method to speed up kernel machines. In T. Leen, T. Dietterich, and V. Tresp, editors, Advances in Neural Information Processing Systems 13, Cambridge, MA, 2001. MIT Press.

[Smola 00] A. Smola and B. Schölkopf. Sparse greedy matrix approximation for machine learning. In Proceedings of the Seventeenth International Conference on Machine Learning, pages 911–918, Stanford, CA, USA, June 2000.

[Achlioptas 02] D. Achlioptas, F. McSherry, and B. Schölkopf. Sampling techniques for kernel methods. In T. G.Dietterich, S. Becker, and Z. Ghahramani, editors, *Advances in Neural Information Processing Systems 14*, Cambridge, MA, 2002. MIT Press.

[Fine 01] S. Fine and K. Scheinberg. Efficient SVM training using low-rank kernel representations. Journal of Machine Learning Research, 2:243–264, December 2001.

[Vapnik 98] V. Vapnik. *Statistical Learning Theory*. Wiley, New York, 1998.

[Chang 04] C.-C. Chang and C.-J. Lin. *LIBSVM: a Library for Support Vector Machines*, 2004. Softwareavailable at http://www.csie.ntu.edu.tw/˜cjlin/libsvm;

[Osuna 97] E. Osuna, R. Freund, and F. Girosi. Training support vector machines: an application to face detection. In *Proceedings of Computer Vision and Pattern Recognition*, pages 130–136, San Juan, Puerto Rico, June 1997.

[Platt 99] J. C. Platt. Fast training of support vector machines using sequential minimal optimization. In B. Sch¨olkopf, C. Burges, and A. Smola, editors, *Advances in Kernel Methods – Support Vector Learning*, pages 185–208. MIT Press, Cambridge, MA, 1999.

[Vishwanathan 03] S. V. N. Vishwanathan, A. J. Smola, andM. N.Murty. Simple SVM. In *Proceedings of the Twentieth International Conference onMachine Learning*, pages 760–767, Washington, D.C., USA, August 2003.

[Fung 03] G. Fung and O. L. Mangasarian. Finite Newton method for Lagrangian support vector machine classification. *Neurocomputing*, 55:39–55, 2003.

[Mangasarian 01a] O. L.Mangasarian and D. R.Musicant. Active set support vector machine classification. In T. Leen, T. Dietterich, and V. Tresp, editors, *Advances in Neural Information Processing Systems 13*, pages 577–583, Cambridge, MA, 2001a. MIT Press.

[Mangasarian 01b] O. L. Mangasarian and D. R. Musicant. Lagrangian support vector machines. *Journal of Machine Learning Research*, 1:161–177, 2001b.

[Kao 04] W.-C. Kao, K.-M. Chung, C.-L. Sun, and C.-J. Lin. Decomposition methods for linear support vector machines. *Neural Computation*, 16:1689–1704, 2004.

[Yang 05] C. Yang, R. Duraiswami, and L. Davis. Efficient kernel machines using the improved fast Gauss transform. In *Advances in Neural Information Processing Systems 17*, Cambridge, MA, 2005. MIT Press

[Pavlov 00a] D. Pavlov, J. Mao, and B. Dom. Scaling-up support vector machines using boosting algorithm. In Proceedings of the International Conference on Pattern Recognition, volume 2, pages 2219– 2222, Barcelona, Spain, September 2000a.

[Collobert 02] R. Collobert, S. Bengio, and Y. Bengio. A parallel mixture of SVMs for very large scale problems. Neural Computation, 14(5):1105–1114, May 2002.

[Lee 01] Y.-J. Lee and O. L. Mangasarian. RSVM: Reduced support vector machines. In Proceeding of the First SIAM International Conference on Data Mining, 2001.

[Schohn 00] G. Schohn and D. Cohn. Less is more: Active learning with support vector machines. In Proceedings of the Seventeenth International Conference on Machine Learning, pages 839–846, San Francisco, CA, USA, 2000. Morgan Kaufmann.

[Tong 00] S. Tong and D. Koller. Support vector machine active learning with applications to text classification. In Proceedings of the 17th International Conference onMachine Learning, pages 999–1006, San Francisco, CA, USA, 2000. Morgan Kaufmann.

[Pavlov 00b] D. Pavlov, D. Chudova, and P. Smyth. Towards scalable support vector machines using squashing. In Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pages 295–299, Boston, Massachusetts, USA, 2000b.

[Bakir 05] G. H. Bakir, J. Weston, and L. Bottou. Breaking SVM complexity with cross-training. In Advances in Neural Information Processing Systems 17, Cambridge, MA, 2005. MIT Press.

[Boley 04] D. Boley and D. Cao. Training support vector machine using adaptive clustering. In Proceedings of the SIAM International Conference on Data Mining, pages 126–137, Lake Buena Vista, FL, USA, April 2004.

[Yang 03] H. Yu, J. Yang, and J. Han. Classifying large data sets using SVM with hierarchical clusters. In Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pages 306–315, Washington DC, USA, 2003.

[Friess 98] T. Friess, N. Cristianini, and C. Campbell. The Kernel-Adatron algorithm: a fast and simple learning procedure for support vector machines. In Proceedings of the Fifteenth International Conference on Machine Learning, pages 188–196, Madison, Wisconsin, USA, July 1998.

[Vishwanathan 03] S. V. N. Vishwanathan, A. J. Smola, andM. N.Murty. SimpleSVM. In Proceedings of the Twentieth International Conference onMachine Learning, pages 760–767, Washington, D.C., USA, August 2003.

[Tresp 01] V. Tresp. Scaling kernel-based systems to large data sets. Data Mining and Knowledge Discovery, 5(3):197–211, 2001.

[Schölkopf 02] B. Schölkopf and A. J. Smola. Learning with Kernels. MIT Press, Cambridge, MA, 2002.

[Platt 99] J. C. Platt. Fast training of support vector machines using sequential minimal optimization. In B. Schölkopf, C. Burges, and A. Smola, editors, Advances in Kernel Methods – Support Vector Learning, pages 185–208. MIT Press, Cambridge, MA, 1999.

[Collobert 02] R. Collobert, S. Bengio, and Y. Bengio. A parallel mixture of SVMs for very large scale problems. Neural Computation, 14(5):1105–1114, May 2002.

[Smola 04] A. Smola and B. Sch¨olkopf. A tutorial on support vector regression. Statistics and Computing, 14 (3):199–222, August 2004.

[Joachims 99] T.Joachims. Making large-scale support vector machine learning practical. In B. Schölkopf, C. Burges, and A. Smola, editors, Advances in Kernel Methods – Support Vector Learning, pages 169–184. MIT Press, Cambridge, MA, 1999.

[Garey 79] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NPCompleteness*. W. H. Freeman, 1979.

[Vazirani 01]V. V. Vazirani. *Approximation Algorithms*. Springer, 2001.

[Liu 07a] Q. Liu, Q. He and Z. Shi, “Incremental Nonlinear Proximal Support Vector Machine”, ISNN 2007, Springer-Verlag Berlin, Nanjing, 2007 pp.336-341.

[Fung 01a] G. Fung, and O.L. Mangasarian, “Proximal Support Vector Machine Classifiers”, Proceedings of the seventh ACM SIGKDD international conference on Knowledge Discovery and Data Mining, ACM Press, San Francisco, 2001 pp. 77-86.

[Huang 04] G.B. Huang, QinYu Zhu, and C.K. Siew, Extreme Learning Machine: A New learning Scheme of Feedforward Neural Networks, IJCNN. 2004.

[Fung 01b] G. Fung., Mangasarian O.: Incremental Support Vector Machine Classification. Data Mining Institute Technical Report 01-08, Computer Sciences Department, University of Wisconsin , 2001.

[Vapnik 95] V. Vapnik. The Nature of Statistical Learning Theory. New York: Springer Verlag, 1995.

[Burges 98] C.J.C.Burges. A tutorial on support vector machines for pattern recognition. Knowledge Discovery and Data Mining, 1998: 121-167.

[Lee 99] Yuh-Jye Lee and O. L. Mangasarian. SSVM: A smooth support vector machine.Technical Report 99-03, Data Mining Institute, Computer Sciences Department, University of Wisconsin, Madison, Wisconsin, September 1999. Computational Optimization and Applications 20(1), October 2001.

[Syed 99] Nadeem Ahmed Syed, Huan Liu, Kah kay Sung. Incremental Learning with Support

Vector Machines. In Proc. Int. Joint Conf. on Artificial Intelligence(IJCAI-99), 1999

[Xiao 01] XIAO Rong, WANG Ji-cheng, SUN Zheng-xing, ZHANG Fu-yan. An Incremental SVM Learning Algorithm α-ISVM. Jounal of Software Vol.12, 2001.

[Rüping 01] Stefan Rüping. Incremental Learning with Support Vector Machines.Proceedings of the 2001 IEEE International Conference on Data Mining, 2001

[Mangasarian 99] O. L. Mangasarian, D. R. Musicant. Successive overrelaxation for support vector machines. IEEE Transactions on Neural Networks,1999: 1032-1037.

[Golub 96] G.H.Golub, C.G.Van Loan. Matrix Computations. The john Hopkins University press, Baltimore, Maryland,1996.

[Zhang 95] Zhang L, Wu FC, Zhang B, Han M. A learning and synthesis algorithm of multi-layered feed forwardneural networks. Journal of Software, 1995: 440~448.

[Zhang 99] Zhang L, Zhang B, Yin HF. An alternative covering design algorithm of multi-layer neural networks. Journal of Software, 1999: 737~742.

[Fung 01c] Fung G., Mangasarian O.: Multicategory Proximal Support Vector Classifiers. Submitted to Machine Learning Journal 2001

[Suykens 99] J. Suykens, and J. Vandewalle, "Least Squares Support Vector Machine Classifiers", Neural Processing Letters, Springer, 1999, pp. 293-300

[Liu 07b]Qiuge Liu, Qing He, and Zhongzhi Shi. Incremental Nonlinear Proximal Support Vector Machine, in D. Liu et al. (Eds.): ISNN 2007, LNCS 4493, Part III, pp. 336 – 341, 2007

[Ruping 01] S. Ruping, "Incremental Learning with Support Vector Machines", ICDM'01, 2001, pp.641-642.

[Gale 1960] D. Gale, "The Theory of Linear Economic Models", McGraw-Hill Book Company, New York, 1960.

[Sartori 91]M.A. Sartori and P.J. Antsaklis, A simple method to derive bounds on the size and to train multilayer neural networks, *IEEE Trans. Neural Networks*. vol. 2, pp. 34-43. 1991.

[Huang 03] G.B. Huang, L. Chen, and C.K. Siew, Universal approximation using incremental feedforward networks with arbitrary input weights, *ICIS*. 2003.

[Huang 04] G.B. Huang, QinYu Zhu, and C.K. Siew, Extreme Learning Machine: A New learning Scheme of Feedforward Neural Networks, *IJCNN*. 2004.

[Vapnik 95] V.N. Vapnik, The Nature of Statistical Learning Theory. Springer, New York, 1995.

[Vapnik 98] V.N. Vapnik, Statistical Learning Theory, Wiley, New York, 1998.

[Bertero 86] M. Bertero, Regularization methods for linear inverse problems, Inverse Problems, Springer, Berlin, 1986.

[Tikhonov 97] A.N. Tikhonov and V.Y. Arsenin, Solutions of Ill-posed Problems, *W.H. Winston, Washington, DC*, 1997.

[Evgeniou 00a] T. Evgeniou, M. Pontil, and T. Poggio, Regularization networks and support vector machines, *Advances in Computational Mathematics*, pp. 13:1-50, 2000.

[Evgeniou 00b] T. Evgeniou, M. Pontil, and T. Poggio, Regularization networks and support vector machines, *Advances in Large Margin Classi¯ers*, pp. 171-203, 2000.

[Mangasarian 06]O.L. Mangasarian and E.W. Wild, Multisurface proximal support vector machine classification via generalized eigenvalues, *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 28(1):69-74, 2006.

[Nilsson 65] N.J. Nilsson, Learning Machine. New York: McGraw-Hill, 1965.

[Baum 88] E.B. Baum, On the capabilities of multilayer perceptions, *J. Complexity*. vol.4, pp. 193-215, 1988.

[Huang 91]S.C. Huang and Y.F. Huang, Bounds on number of hidden neurons in multilayer perceptrons, *IEEE Trans. Neural Networks*, vol. 2, pp. 47-55, Springer-Verlag, 1991.

[Serre 02] D. Serre, Matrices: Theory and applications, Springer-Verlag New York, Inc, 2002.

[Bertero 88] M. Bertero, T. Poggio and V. Torre, Ill-posed problems in early vision, Proc. IEEE, pp.869-889, 1988.

[Wahba 90] G. Wahba, Splines Models for Observational Data, Series in Applied Mathematics, Vol. 59, SIAM, Philadelphia, PA, 1990.

[Vapnik 82] V.N. Vapnik, Estimation of Dependences Based on Empirical Data, Springer, Berlin, 1982.

[Murphy 92] P.M. Murphy and D.W. Aha, UCI repository of machine learning databases, 1992.

[Mitchell 97] Mitchell T. M.: Machine Learning. McGraw-Hill, 1997.

[Quinlan 86] Quinlan J. R.: Induction of decision tree. Machine Learning, 1986, (1):81-106.

[Quinlan 93] Quinlan J. R.: C4.5: Programs for Machine Learning. Morgan Haufmann Pub., Inc., Los Altos, California, 1993.

[Breiman 84] Breiman L., Friedman J. H., Olshen R. A., and Stone C. J.: Classification and Regression Trees. Wadsworth, Belmont, 1984.

[Mehta, 96] Mehta M., Agrawal R., Rissanen J.: SLIQ: A Fast Scalable Classifier for Data Mining. Lecture Notes in Computer Sci. Proc. of the 5th Int. Conf. on Extending Database Tech., pp.18-33, 1996.

[Shafer 96] Shafer J. C., Agrawal R., Mehta M.: SPRINT: A Scalable Parallel Classifier for Data Mining. Proc. of the 22nd Int. Conf. on Very Large Databases, 1996.

[Hunt 66] Hunt E. B., Marin J., Stone P. T.: Experments in Induction. Academic Press, 1966.

[Chickering 95] Chickering D. M.: Learning Bayesian networks in NP-complete. Lecture Notes in Statistics, 1995.

[Friedman 96] Friedman M., Goldszmidt T.: Building Classifers Using Bayesian Networks.

Proc. of 13th National conf. on AI, 1996.

[LiuB 98] Liu B., Hsu W., Ma Y.: Integrating Classification and Association Rule Mining. Agrawal R. Proc. of the 4th Int. Conf. on Knowledge Discovery and DataMining, NY, USA: AAAI Press, pp. 80-86, 1998.

[史 06] 史忠植：智能科学，清华大学出版社，2006 年 8 月。

致 谢

马上就要完成硕士学位的毕业论文了，在过去的 2 年多的读书学习过程中，有很多人给了我帮助和支持，没有他们我是没有可能完成这篇硕士毕业论文的。

首先，感谢我的导师何清副研究员以及课题组长史忠植研究员。感谢何老师在入学时接收我，使我能够在智能科学课题组里读研；感谢何老师对我学习过程中的帮助；感谢何老师对我开题、中期、答辩及写文章过程中的悉心指导和宝贵建议，没有这些指导建议我不可能写出目前已发表的文章；感谢何老师给了我非常好的研究环境，包容了我研究过程中的固执；何老师使我始终认识到我还有很多工作没有做好，还有很长的路要走；何老师严谨细致的作风和平易近人的品质将一直是我学习、生活中的榜样。我要感谢史老师把我介绍给何老师，使我获得了读研究生的机会，史老师对学生的关爱照顾常常使我受宠若惊；史老师每天的工作时间、每年的工作安排，都给了我极大的震动，史老师的敬业精神将是我一生学习的典范；感谢史老师为我提供的良好的学习、科研环境，史老师严谨的治学态度、渊博的专业知识与敏锐的学术洞察力让我真正领略到了一个科学家的风范。我有很多工作没有做好，深感对不起老师对我的期望。

感谢中国科学院智能信息处理重点实验室的老师和同学们，他们给了我很大的帮助和支持。他们是胡宏副研究员、施智平博士、蒙祖强副教授、彭晖副教授、常亮同学、赵卫中同学、马慧芳同学、曾立同学、黄瑞同学、罗杰文同学、林芬同学、李志清同学、李志欣同学、石志伟同学、陈立民同学、杨来同学、张大鹏同学、庄福振同学、林欢欢同学、张颖同学、刘曦同学、张子云同学、韩旭同学、牛温佳同学、叶飞同学、石川同学、罗平同学、陈明同学、张素兰同学、史春奇同学、邱莉榕同学、王茂光同学、张志勇同学、谭力同学、万常林同学、杨柳同学、何潇潇同学、史俊同学、余清同学、尹超同学、以及诸葛海研究员、眭跃飞研究员、曹存根研究员、李华研究员、胡兰萍女士、田卫平女士等。能够与你们一起工作学习，使我收益匪浅，也是我的幸运。

感谢计算所研究生部的李琳老师、宋守礼老师、张晓辉老师、周世佳老师等在生活和学习上的诸多关心和帮助。

最后，特别要感谢我的父母，他们给了我所有的信任，时刻牵挂着我的学习、生活，我会努力使他们不再为我担心、过的更加高兴，祝父母身体健康。

作者简历

姓名：刘秋阁　　性别：男　　出生日期：1983.09.19　　籍贯：山东

2005.09 -- 2008.7　　中国科学院计算技术研究所 计算机软件与理论 硕士研究生
2001.09 -- 2005.7　　山东大学 计算机软件与理论 本科

【攻读硕士学位期间发表的论文】

- Qiuge Liu, Qing He, and Zhongzhi Shi. Extreme Support Vector Machine, in PAKDD'08，accepted(EI 源).
- Qiuge Liu, Qing He, and Zhongzhi Shi. Incremental Nonlinear Proximal Support Vector Machine, in D. Liu et al. (Eds.): ISNN 2007, LNCS 4493, Part III, pp. 336 – 341, 2007(EI 已收录).
- Qiu-ge Liu, Qing He, Zhong-zhi Shi. Data Selection for Nonlinear Proximal Support Vector Machine, Third International Conference on Natural Computation，Vol.1,pp.120-124.（EI 已收录）
- 刘秋阁，何清，史忠植. 一种新的非线性支持向量机分类算法.CAAI-12, 北京邮电大学出版社, 2007, 190-195.

【攻读硕士学位期间参加的科研项目】

1. 国家自然科学基金“基于感知学习和语言认知的智能计算模型研究”No. 60435010，2005.1-2008.12；
2. 国家自然科学基金“基于超曲面的覆盖分类算法与理论研究” No. 60675010，2007.1-2009.12；
3. 863 高技术探索项目“基于感知机理的智能信息处理技术”2006AA01Z128，2006.12.1-2008.12.31；
4. 973 项目子课题“语义网格资源描述模型、形式化理论和支撑技术”No. 2003CB317004，2003.1-2007.12；
5. 973 项目子课题“非结构化信息(图像)的内容理解与语义表征”No. 2007CB311004；
6. 北京市自然科学基金“海量高维、多类数据分类法研究及其应用”No. 4052025, 2005.1-2007.12。