西南交通大学

硕士学位论文

基于LSSVM的短期交通流预测研究与应用

姓名：刘林

申请学位级别：硕士

专业：计算机应用技术

指导教师：戴齐

201105

摘要

短时交通流预测是实现智能交通控制与管理、交通流状态辨识和实时交通流诱导的前提及关键，也是智能化交通管理的客观需要。到目前为止，它的研究结果都不尽如人意。现有的以精确数学模型为基础的传统预测方法存在计算复杂、运算时间长、需要大量历史数据、预测精度不高等缺点。因此通过研究新型人工智能方法改进短期交通流预测具有一定的现实意义。本文在对现有短期交通流预测模型对比分析及交通流特性研究分析基础上，采用最小二乘支持向量机方法进行短期交通流预测模型，取得较好的效果。

支持向量机是一种新的机器学习算法，建立在统计学习理论的基础上，采用结构风险最小化原则，具有预测能力强、全局最优化以及收敛速度快等特点，相比较以经验风险化为基础的神经网络学习算法有更好的理论依据和更好的泛化性能。对于支持向量机模型而言，其算法相对简单、运算时间短、预测精度较高，比较适用于交通流预测研究，特别是在引入最小二乘理论后，计算简化为求解一个线性方程组，同时精度也能得到保证。

在最小二乘支持向量机理论的基础上、利用类似“滑动窗口”的概念，提出了一种新的在线算法，更新计算矩阵，并且通过“剪枝”，去除对模型影响较小的支持向量，并通过 MATLAB7.0 仿真实验验证了其算法有效性，在一定条件下，适用于短期交通流的预测。本文的主要工作如下：

1.首先第一步理论分析得到支持向量机的特点，然后再引入最小二乘方法，对支持向量机的解法进行改进。在此基础上，使用经典的“剪枝”法，并进行计算机仿真，证明其方法的科学性。最后针对此方法不能进行更新的问题，提出一种类似“滑动窗口”的概念，利用矩阵理论，增加或者删除系数矩阵，实现在线更新的功能。改进的在线"剪枝"算法可以在保证一定的时间开销的情况下，达到较好的预测精度。

2.最后通过一个实际的例子，以第三章的改进算法为核心，建立一个完整的短期交通流预测模型，并通过计算机仿真实验和误差分析，验证了模型的实用性和有效性。

关键词：短时交通流预测，支持向量机和 LSSVM，在线"剪枝"算法

Abstract

Short-term traffic flow forecasting is the premise and key of intelligent traffic control and management, traffic flow state realization and real-time traffic guidance and is also the impersonality necessity. But so far, its results are unsatisfactory. Traditional forecasting methods based on accurate mathematical models which are computing hard, spending-time and needing too many historical data but less predication precision are no ideal. Therefore, studying artificial intelligence methods have some reality meanings. Based on comparative analyses of the existing short-term traffic flow models and research of characteristics of traffic flow, the thesis establishes the goal: the models established can remedy the shortcoming of the existing short-term traffic flow forecasting models

Support Vector Machine(SVM) is the new machine learning method based on statics learning theory(STL).it has the characteristic which contain the structure Risk Minimization of Statistical Learning Theory, strong predication emotion, optimum result of all data point. Compared to artificial network based on experience Risk Minimization, it has the more theory basement and application foreground. And due to the SVM model, it has the simple algorithm and less time spending and high predication precision. so it has been applied on traffic predication.

This thesis based on Least Square Support Vector Machine theory use the principal "slide windows" to make a new online algorithm. The new algorithm can efficient dynamic renew the matrix and "pruning" the less efficient Support Vector. Using the MATLAB 7.0 to validate the new algorithm has feasibility. So it could be applied on Short-term traffic flow predication. Major jobs are as follow.

First it can get the feature of SVM through the chapter 2.then we get the least Square method to mend the SVM algorithm in chapter 3. Base on these theory and algorithm, we use the classical method "pruning" to prove the efficiency and scientific of this method through computer simulation. Last using the matrix theory and apply the principle like "slide window" to renew the row/arrange. So the exchange algorithm could achieve the predication precision while satisfy the time-consume.

In the last of this paper, we use the real data to prove the predication model which the kernel algorithm was mentioned in chapter3. we use the computer simulation and the error analysis to make sure the application and available of the model.

Keywords: Short-term traffic flow forecast SVM&LSSVM Online Pruning algorithm.

第一章 绪论

1.1 论文研究背景

根据资料显示，全美每年因为交通堵车而带来的汽油浪费和时间上的消耗对美国造成上千亿美元的经济损失[53]。我国北京、上海等大城市现在也因为交通问题，而陷入了交通堵塞的困境、严重影响了城市的运转效率、阻碍了经济的快速发展[54]。从上个世纪80年代起，西方国家开始研究各种方法，试图通过先进交通管理系统(ATMS) 、先进交通信息系统(ATIS) 、动态路径引导系统(DRGS)等各种智能交通系统来解决交通的堵塞问题[55]。实时的交通检测数据是这些方法的关键，但更重要的是通过这些数据建立预测模型，为动态路径引导系统和交通信息系统提供方便，从而缓解交通堵塞，减少污染、节约能源等目的。一个可靠、准确的交通预测信息是动态路径诱导系统的基础和关键。

时间序列的预测是对历史数据进行学习和总结得到一个非线性的映射，近似的得到数据中隐含的非线性机制，从而可以利用该映射进行时间序列的预测[1]。

近年来，数据挖掘中的时间序列的建模和预测一直是学术研究和实际应用领域的研究热点。在自然领域方面，河水的流量预测、流域的降水量预测、粮食的产量预测、太阳黑子数的研究预测等，还有社会领域中公路和铁路交通流量根据时间长短的预测，某一地区人口的增长量的预测、医院门诊诊断的人员数量变化预测。经济领域方面，股票价格指数预测、产品价格变化预测、以及国民收入预测。根据对许多有关时间序列预测方面的资料研究和分析，人们逐渐掌握了一些建模及预测的方法和规律。

和传统的统计学相比较统计学习理论(SLT) [10]是一种专门研究小样本情况下机器学习规律的理论。V.Vapnik 等人从上个世纪六，七十年代开始致力与此方法的研究，到现在处于正在发展和成熟阶段。同时也由于神经网络等机器学习方法在理论上没有实质性的进展，统计学习理论越来越受到广泛的关注。在 SLT 基础上产生的支持向量机(SVM)学习方法，其基本思想是通过内积函数定义的非线性变换将输入空间变换到一个高位空间，在这个高位空间中寻找输入变量和输出变量之间的一种非线性关系。SVM 有非常严格的理论基础，是基于结构风险最小化原则的方法，明显优于传统的基于风险最小化原则的常规神经网络方法。其算法是一个凸二次优化的问题，能够保证找到的是全局最优化解，能较好的解决小样本和非线性高维数等实际问题，问题的复杂度不由问题的维数所决定，有良好的推广能力。

1.2 国内外的研究现状

交通流预测在国际上一直是很活跃的，在过去的几十年里研究者们已经做了大量的工作。现在国内外关于交通流动态预测理论的研究取得了显著的成果。从 20 世纪 60 年代起，国内外的学者专家开始把在其他领域应用的成熟的预测模型用于短时交通流预测模型。早期的预测方法主要有：自回归滑动平均模型(ARMA),历史平均模型(HA),以及 Box-Con 等方法[14]。随着该领域研究的逐渐深入，出现了一批更复杂、精度更高的预测方法，这些模型可以分为五类：基于统计理论的模型，基于智能理论的模型，基于非线性预测理论的模型，基于微观交通仿真的模型和混合模型等[54]

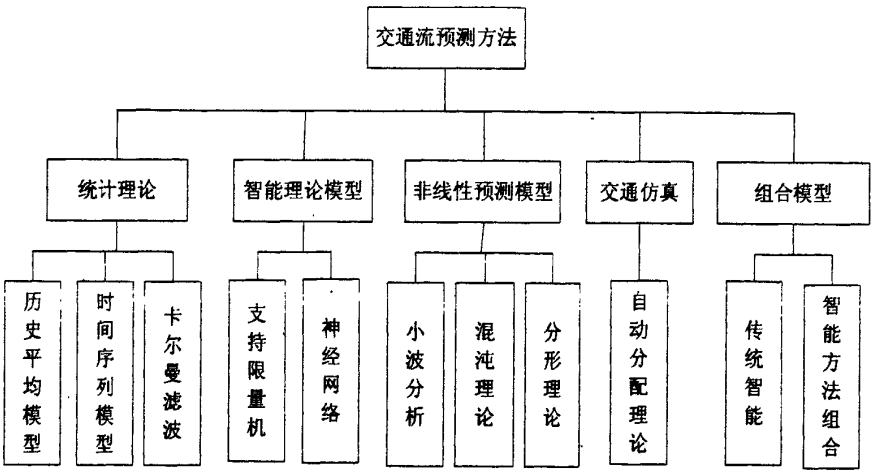

图 1-1 交通流预测模型示意图

通过近年来的研究成果表明，在短期交通流预测方面，智能理论模型是处于比较理想的位置。人工神经网络较卡尔曼滤波，MA 及 ARIMA 等模型有较好的预测效果。由于神经网络方法是基于渐进学习理论，同时利用梯度下降调节目标函数达到极小，易于陷入局部极值，也会导致过分学习和过拟合，收敛速度慢，使模型的泛化能力受到限制，在小样本的情况下，经验风险与实际风险的差异比较明显，在实际应用中受到的限制比较多。在这种情况下，针对神经网络方法的不足，Vapnik 等人提出了基于统计学习理论[10]为基础的支持向量机(SVM)方法，特别是在小样本，高维和非线性数据空间上有很好的推广能力，由于是解一个凸二次规划问题，在理论上得到的是全局最优解，并且相关研究证明[10]SVM 在 ITS 领域应用的可能性，根据使用的预测算法不同，分为 S-SVM,v-SVM 和 LSSVM(最小二乘支持向量机)等。

1.3 研究内容

通过对相关文献的研究，鉴于道路交通系统的非线性、复杂性和不确定性的基本特征，传统的基于统计理论的数学预测模型越来越不能满足实际的需要。近年来的研究成果表明，智能理论模型和关于几种方法集成的集成学习模型，相比较传统方法有更好的

效果。支持向量机是新兴的一种智能机器学习模型算法，是复杂非线性科学和人工智能科学的研究前沿，基于结构风险最小化的原理克服了神经网络容易陷入过学习和取得局部最小值的缺点。

综合国内外的交通流预测模型，通过引入支持向量机到交通流预测模型，并且使用改进的最小二乘支持向量机的方法，去尝试一种新的研究方法，新的思路去解决交通流预测问题。

本文的主要内容如下：

1. 先对交通流的特性进行分析，总结短期交通流预测模型应具备的特点。
2. 结合交通流的特性，建立交通流预测模型，并且在研究统计学习理论和支持向量回归原理的基础上，将支持向量回归方法用于交通流预测。
3. 对于标准的SVM使用最小二乘法，用这种变形的方法去建立新的LSSVM交通流预测模型，并且使用矩阵论的相关理论，优化解凸二次规划的性能。

1.4 本文的组织

第一章：绪论

首先分析本文的研究背景，再介绍了国内外的研究状态以及研究的必要性和内容。

第二章：短期交通流和支持向量机理论的分析

首先介绍并总结了交通流的特性和常用的短期交通流预测模型，然后简单的介绍支持向量机理论的基本原理和理论重点。

第三章：最小二乘支持向量机模型

首先介绍最小二乘理论，然后提出了线性和非线性的最小二乘算法，通过改进最小二乘支持向量机的算法，提高精度和运算速度，并且进行试验的验证。然后在经典算法的基础上面，针对经典算法不能实现动态的更新数据的问题，提出了一种类似"滑动窗口"概念的在线"剪枝"算法，在继承了经典算法的精度和时间效率的同时，能够“在线”更新数据。

第四章：基于LSSVM的交通流预测模型

通过第三章对改进的最小二乘支持向量机算法的学习研究。本章通过交通流预测的流程建立了一个基于LSSVM算法的预测模型，并通过实际的数据验证了改进后的算法在能够保证精度的同时，动态的更新数据，完成预测功能。

总结和展望

总结了全文，对全文的工作给出了评价，并给出以后要进行的工作。

第二章 短期交通流分析和 SVM 模型

2.1 短期交通流分析

交通流理论是运用物理和数学定理来描述交通流特征的理论、是研究交通流随时间和空间变化规律的模型和方法体系、使用数学或者物理上的原理对交通流各参数和其中的关系进行定性、定量分析去寻找交通变化的规律，从而为交通的规划和管理以及道路设计提供理论上的支持[11]。作为交通运输各个方面的理论支持，研究内容涵盖了描述交通流现象的各种理论思想、模型和方法。各种交通流模型就是交通流理论的核心研究内容[12]。揭示交通系统运行的基本规律是交通流预测的主要目的，为的是改善交通流的运行质量、提高交通运输的效率、保证交通运输的安全性。通过理论的分析和数学仿真相结合的方法去阐述交通现象和原理，使我们能更好的理解交通运行的特点和实质。

交通流变化过程是一个实时、非线性、多维且非平稳的随机过程，随着统计时间段的缩短，交通流变化的随机性和不确定性越来越强[6]。短时交通流的随机性和不确定性还与天气变化，交通事故和交通环境等因素有关系。为此提出了交通流的特性：

(1) 交通流的路网性：城市交通道路纵横交错且又是相互联通的，主干道和各个辅道交织在一起，每一条道路上面都有一定的交通流量，致使交通网形成一个有物质流动的网络。

(2) 交通流的周期性：由于人们出行呈现规律性，所以显示出交通需求的规律性，按年，月，周都能显示出交通流曲线的相似性。比如每一天的交通流的高峰都是出现在上下班时间，而其他时间段的变化呈现出的是相对平缓的变化率，这就体现出了交通流的周期性。在时间尺度上边，一周中工作日上下班的变化呈周期性变化，在空间尺度上面，相同地点交通流的变化情况也是随时间而周期变化的。

(3) 交通流的随机性：由于交通的需求和人们是息息相关的，出行者的目的不同，而且每一条道路的当时状况是不同的就决定了交通流呈现出很大的随机性，而且随着时间变得越来越短随机性会显著的增加。

(4) 交通流的时空相关性：交通流随时间变化而发生变化，并且不同路段的交通流在相同时间段内是不一定相同的。相同的路段在不同的时间段内交通流也不见的是相同的。

(5) 交通流的非线性和不确定性：一般在要研究的路段上面车辆的排列状态和车辆自身的状态、天气变化等因素都是不确定的，制约交通流的相关因素各种各样呈现出来的是非线性关系。

2.2 交通流预测模型要求

(1) 实时性：由于是短期交通流预测，故模型具有快速的计算能力和反应能力，

(2) 准确性：模型预测的结果是用于道路交通的运输管理，故预测对模型的精度要求比较高，不然不准确的信息将给交通管理系统带来不必要的错误。

(3) 可靠性：因为短期交通事故预测受到比较多的因素的影响（比如天气，事故，特殊事件等），所以交通流的预测模型应该具有很好的抗干扰能力。

(4) 时空相关性：交通流受到时间和空间两个变量的影响，但是在同一个位置，并且在相同的时间里面，人们的出行还是具有一定的规律性[13]。

2.3 短期交通流预测模型概述

智能交通控制、管理、交通状态辨识和交通诱导的核心技术就是交通流预测。它作为一项很重要基础理论，同时也是当今世界交通领域正在攻克的难关[56]。下面对现在几个主要的交通流预测模型做一个总结。

(1) 历史平均模型(History Average Model)[56]

历史模型的算法定义为：$V(\text{new}) = aV + (1-a)V(old)$。在公式中，$V(new)$代表的是路段在一定时间间隔内新的交通流量，$V(old)$代表的是该路段在一定的时间间隔内旧的交通流量，$V$是新旧时间间隔内增加的交通流，$a$为系数值。

在1981年就有美国的科学家把历史平均模型应用于城市交通控制系统。历史平均算法简单，参数少。可以在一定程度上解决不同时间段的交通流变化问题。但是它的静态预测不足，不能反应动态交通流的基本不确定性和非线性，无法抗干扰。

(2) 时间序列模型(Time-Series-Model)[56]

时间序列模型于1979年首次在交通预测领域被Amend和Cook提出来。Box和Jenkins有创立了(Box-Jenkns模型)又称为ARIMA(自回归移动平均模型)。这种模型使用广泛，它将某一个时刻的交通流量看成是一个一般的非平稳随机序列，带有3到6个参数。ARIMA模型在1984年被美国的科学家应用到高速公路交通流量预测中。模型是建立在大量不间断的数据的基础上，预测精度比较高，但是需要复杂的参数估计，并且参数不能移植。由于实际的情况和理论上面的差异，现实中数据往往出现残缺或者遗漏。导致模型精度下降，并且过于依赖大量历史数据。ARIMA时间序列模型适用于稳定的交通流，对于交通情况突发事件会出现预测延迟，精度下降的情况。

(3) Kalman滤波模型(Kalman Filtering Model)[56]

Kalman滤波理论是有Kalman在1960年提出，1984年美国的科学家Okutani和Stephandes将Kalman理论运用用到交通流预测模型里面，同时Vythotkaapc也提出了基于Kalman滤波理论的交通流预测模型。Kalman滤波因为具有预测因子，所以选择灵活，精度较高的优点，预测精度随时间的变化不大，健壮性比较好。但是由于模型是线性估计模型，当时间间隔过小，交通流的随机性和非线性性再加强，性能会变差，并且模型算法复杂，很难进行实时预测。

由于以上的交通流预测模型都无法进行实时预测，就已知的数据通过自我学习而预测未

来，而神经网络良好的自适应和自学习能力，使得在交通流预测领域里面应用越来越多。

(4) 神经网络模型（Neural Network Model）[56]

人工神经网络在上个世纪 40 年代由美国的科学家首先提出，在 1993 年和 1994 神经网络被 Dougherty 和 Clark 分别用于短时交通流预测。由于交通系统的复杂性，而神经网络采用“黑箱”式学习模型，适合交通流预测的应用。神经网络不需要任何的经验公式，能从已有的数据自动归纳规则，获得数据的内部规律，在拥有大量输入和输出样本的情况下，由内部“黑箱”的调整，便可以建立良好的输入和输出模型。现在的神经网络大概分为以下几类：BP 神经网络模型，单元神经网络模型等。

(5)混沌理论模型(Chaos Theory Based Model)

混沌是指一种类似于无规则的运动，指在非线性系统中，不用任何随机的因素亦可出现类似随机性行为。研究的目的是找出在类似于随机现象后面的简单规律，并且用这些简单规律来解决一大类复杂系统问题。混沌理论是用于短期预测而非长期预测。

以下是对上面各个模型以及小波模型优缺点的总结：

表 2-1 时间序列预测模型列表

模型	优点	缺点	适用范围
历史平均模型	模型简单运算速度快	精度差，不能解决非常规的交通预测问题	精度要求不高的静态模型
时间序列模型	建模简单，适合对不易建立精确度模型系统建模，在数据充分的情况下，预测效果较好	过多的依赖不间断的历史数据，当交通状态发生变化时，不能很好的反应，且不能移植	交通情况比较稳定的情况
KAlMAN 滤波模型	适应性好，可以处理平稳和非平稳的数据，模型具有线性，无偏，最小均方差性	线性模型，不宜预测非线性的交通流，输出结果有时延	交通情况较稳定，有时延的情况
神经网络模型	有实时性，可以实时的更新交通流网络，适合交通系统影响因素多的区域	训练过程复杂，数据量大，局限性大，结果不易被理解	适合情况复杂的交通系统
混沌理论模型	从实际的数据出发，得到系统的混沌参数，进行预测，精度高	理论复杂，只能用于短时交通预测	适合复杂，短时交通预测
小波预测模型	可移植性好，预测精度较高	理论复杂，计算量大	用于非线性，复杂交通系统

2.4 支持向量机理论基础和原理

2.4.1 机器学习的表示

机器学习的目的是根据给定的训练样本对某一个系统输入和输出之间的依赖关系的估计，使它能够对未知的输出做出尽可能的准确预测。一般表示为：变量 y 和 x 存在的未知依赖关系，即遵循某一未知的联合概率 $F(x,y)$，（x 和 y 之间的确定关系可以看作是其特例），机器学习问题就是根据 n 个独立同分布的观测样本[10]。

$$(x_1,y_1),(x_2,y_2),(x_3,y_3),\ldots\ldots(x_n,y_n) \tag{2-1}$$

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

$$R(w)=\int L(y,f(x,w))dF(x,y) \tag{2-2}$$

最小。其中，$\{f(x,w)\}$称作预测函数集，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有不同形式的损失函数，预测函数也称作学习函数，学习模型获学习机器[10]。

有三类基本的机器学习问题，即模式识别问题，函数逼近和概率密度估计。对模式识别问题，输出 y 是类别标号，两类情况下 $y=\{0,1\}$或$\{1,-1\}$，预测函数称作损失函数，损失函数可以定义为

$$L(y,f(x,\omega))=\begin{cases}0, & if \quad y=f(x,\omega)\\ 1, & if \quad y\neq f(x,\omega)\end{cases} \tag{2-3}$$

使风险最小就是 Bayes 决策中错误率最小。在函数逼近问题中，y 是连续变量(这里假设为单值函数),损失函数可定义为

$$L(y,f(x,y))=(y-f(x,y))^2 \tag{2-4}$$

即采用最小平方误差准则.而对概率密度估计问题，学习的目的是根据训练样本确定 x 的概率密度。估计的密度函数为 $p(x,w)$,则损失函数可以定义为

$$L(p(x,\omega))=-\log(x,\omega) \tag{2-5}$$

2.4.2 经验风险最小化

学习的目的在于期望风险最小化，但是我们可以用的信息只有样本(2-1), (2-2)的期望风险并无法计算，所以传统的学习方法中采用的经验风险最小化(ERM)准则，即用样本定义经验风险[10]。

$$R_{emp}(w)=\frac{1}{n}\sum_{i=1}^{n}L(y_i,f(x_i,w)) \tag{2-6}$$

对（2-2）式的估计，设计学习算法使它最小化，对损失函数(2-3),经验风险就是训练样本错误率；对(2-4)的损失函数，经验风险就是平方训练误差；而采用(2-5)的损失函数的 ERM 准则就是等价于最大似然函数方法。其实，使用 ERM 原则代替期望风险最小化并没有经过充分的

理论根据，只是在直观上面认为是合理的做法。比如当假设 n 趋于无穷大时(2-6)会接近于(2-2)，在很多问题上面样本的数目是有限个，那么在有限的样本下 ERM 准则得到的结果不见得是真实风险最小。

2.4.3 结构风险最小化

统计学习理论从 VC 维的概念出发[14]，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。关于两类分类问题，结论是：经验风险 $R_{emp}(w)$ 和实际风险 $R(w)$ 之间至少 $1-\eta$ 的概率满足如下两个关系式:

$$R(w) \le R_{emp}(w) + \phi(h/n) \tag{2-7}$$

$$\phi(h/n) = \sqrt{[\frac{h(\ln(2n/h)+1) - \ln(\eta/4)}{n}]} \tag{2-8}$$

其中 h 是函数集的 VC 维，n 是样本数。

表明当 h/l 较大时，学习机器的置信范围越大，导致的真实风险过大，会出现较大的误差。如果样本的数量比较多，使得 h/l 比较小，置信区间范围小，经验风险最小的最优解就会和真实值相当。

当样本数量是固定不变的时候，学习机器的 VC 维越小（也就是复杂度越小），置信区间就越小，真实值和检验风险之间的差值就最小。所以在设计分类器的时候，不但要使经验风险尽可能的小，还要尽量控制 VC 维，从而缩小置信区间。

结构风险最小化原理[15]：如果要使结构经验风险最小化，就要使不等式(7)中右边两项相互平衡，共同趋于最小；同时还要要求 h 的值尽可能的小，也就是置信区间最小。

我们由公式(2-7)，如果训练样本数 n 的大小，则控制结构风险 $R(w)$ 的参数有两个：$R_{emp}(w)$和h

(1) 函数集 $f(w,x)$ 决定机器学习的经验风险最小化，可以通过控制 ω 来控制经验风险。

(2) VC 维的维数 h 依赖于学习机器的函数集合，可以通过使函数集合结构化，建立 h 与各个函数子结构之间的关系，使函数集合同 VC 维 h 结合起来。

我们可以使用以下的方法使函数集合 $\{f(w,a), w \in \Gamma\}$(Γ是抽象参数的集合)结构化。考虑到函数子集的集合 $S_1 \subset S_2 \subset S_3 \subset S_4 \ldots \subset S_n \ldots \subset S$,其中，$S_k = \{f(w,a), a \in \Gamma_k\}$，而任何结构集合 S 中的元素 S_k 有一个有限的 VC 维 h_k,且$h_1 \le h_2 \le h_3 \ldots \le h_k \ldots h$。给定一组样本 (x_1,y_1)，(x_2,y_2)，$(x_3,y_3)\ldots(x_l,y_l)$,根据结构风险最小化原理在函数子集 S_k 中选择一个函数 $f(w,a_l^k)$ 来使经验风险最小化，同时确保 S_k 的置信区间是最小的。

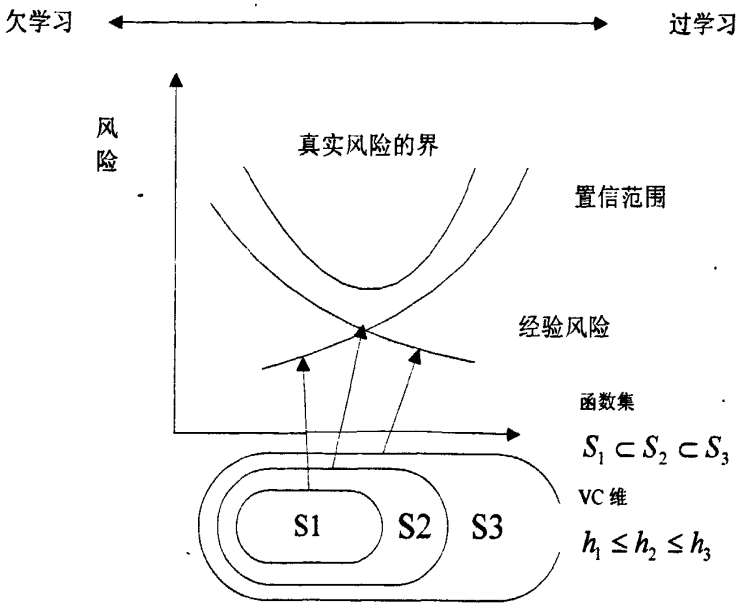

图 2-1 结构风险最小化示意图

2.4.4 支持向量回归机

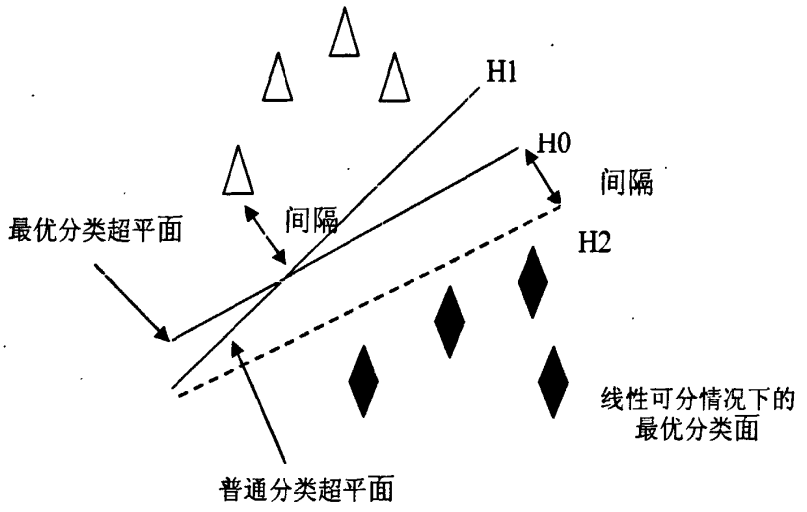

图 2-2 最优分类面示意图

(1)最优分类面

对一个给定训练样本集，假如是线性可分的，机器学习的结果是一个超平面，在二维的情况下，是直线或者判别函数，该超平面可以将训练样本分为正负两类[14]。我们由统计学习理论可以知道，最优超平面不但能将两类样本正确的分开，还可以是分类的间隔最大化。我们从一维推广到高位空间上面去，最优分化直线就是最优的分类面。

实心点和空心点代表两类样本，H_0为分类线，对于适当的给定划分函数的法向量ω,会有

两条极端的直线分别通过两类中离H_0最近的样本且平行于H_0的直线H_1，H_2。而这两条直线的距离叫做分类间隔(Margin)[10]。最优分类直线就是H_0不但能把那两个类分开，而且还使分类的间隔距离最大。用数学公式描述就是：

$$(w\times x)+b=0 \tag{2-9}$$

我们对线性方程进行归一化，使得对线性可分的样本集合(x_i,y_i),i = 1,....,n,$x\in R^d, y\in\{+1,-1\}$,满足$y_i[(\omega\times x_i)+b]-1\geq 0, i=1,....,n$。直接计算知道，此时相应的两条极端直线的距离，就是分类的间距等于$2/\|\omega\|$。极大化间隔思想反应到数学公式上面就是，解一个最优化的问题

$$\min_{\omega,b}\frac{1}{2}\|\omega\|^2$$

$$st.y_i((\omega.x_i)+b)\geq 1, i=1,2,.....,l \tag{2-10}$$

使间隔最大化也就是使$\|\omega\|^2$最小，并且使$\|\omega\|^2/2$最小的分类面叫做最优分类面，而H_1和H_2上的训练样本点就叫做支持向量机。使分类间隔最大实际上就是对泛化能力的度量。

支持向量机是建立在统计学习理论上面的机器学习算法，它分为线性和非线性两种

2.4.5 线性支持向量机

对于线性回归函数

$$f(x)=w\times x+b \tag{2-11}$$

数据集$(x_1,y_1),(x_2,y_2),....,(x_i,y_i),....,(x_N,y_N), x_i,y_i\in R$。为了保证$f(x)=w\times x+b$方程式的平滑，必须选取一个最小的$w$。假定存在函数$f$，取精度$\varepsilon$使得$(x_i,y_i)$的数据能够被估计，故求最小的$\omega$问题就可以转化成一个凸优化问题

$$\min\frac{1}{2}\|w\|^2$$

$$st.y_i-w\times x_i-b\leq\varepsilon 或者 w\times x_i+b-y_i\leq\varepsilon \tag{2-12}$$

由于处理函数f能够处理的数据超出了ε精度的估计范围，所以我们引入了松弛变量ξ_i,ξ_i^*，(2-12)就改写为如下所示的情况

$$\min\frac{1}{2}\|w\|^2+C\sum_{i=1}^{n}(\xi_i+\xi_i^8)$$

$$st.y_i-w\times x_i-b\leq\varepsilon 或者 w\times x_i+b-y_i\leq\varepsilon \tag{2-13}$$

$$\xi_i\geq 0,\xi_i^*\geq 0, i=1,2,.....,n$$

C是惩罚系数，ε为不敏感损失系数，ξ_i,ξ_i^*为松弛变量，其取值由下面的公式说确定

$$\xi_i^{(*)}=\begin{cases}0 & f(x_i)-y_i<\varepsilon\\ |f(x_i)-y_i|-\varepsilon & f(x_i)-y_i\geq\varepsilon\end{cases} \tag{2-14}$$

由对偶原理，建立拉格朗日方程，把(2-13)变成

$$l(w,\xi_i,\xi_i^*)=\frac{1}{2}\|w\|^2+C\sum_i^{\infty}(\xi_i+\xi_i^*)-\sum_i^{\infty}\alpha_i(\varepsilon+\xi_i^*-y_i+w\times x_i+b)-\sum_i^{\infty}\alpha_i^*(\varepsilon+\xi_i^*-y_i+\omega\times x_i+b)$$
$$-\sum_i^{\infty}(\eta_i\xi_i-\eta_i\xi_i^*) \tag{2-15}$$

对(2-15)式求偏导数

$$\frac{\partial l}{\partial w}=w-\sum_{i=1}^{\infty}(\alpha_k-\alpha_k^*)x_i=0$$
$$\frac{\partial l}{\partial b}=\sum_{i=1}^{\infty}(\alpha_k-\alpha_k^*)x_i=0$$
$$\frac{\partial l}{\partial \xi_i}=C-\alpha_i-\eta_i=0$$
$$\frac{\partial l}{\partial \xi_i^*}=C-\alpha_i^*-\eta_i^*=0$$

将(2-13)式按矩阵的形式表示：

$$\min\frac{1}{2}\begin{bmatrix}\alpha^t,(\alpha^*)^t\end{bmatrix}\begin{bmatrix}Q & -Q\\ -Q & Q\end{bmatrix}\begin{bmatrix}\alpha\\ \alpha^*\end{bmatrix}+\begin{bmatrix}\varepsilon e^t+y^t,\varepsilon e^t-y^t\end{bmatrix}\begin{bmatrix}\alpha\\ \alpha^*\end{bmatrix}$$
$$st.\begin{cases}\begin{bmatrix}e^t,-e^t\end{bmatrix}\begin{bmatrix}\alpha\\ \alpha^*\end{bmatrix}=0\\ 0\le\alpha,\alpha^*\le C\end{cases}$$

其中，$Q_{i,j}=(x_i,x_j),e=[1,...1]^T,\alpha_i^*$是拉格朗日乘子，由二次规划的求解

$$w=\sum_{i=1}^{\infty}(\alpha_i-\alpha_i^*)x_i \tag{2-16}$$

由 KKT 条件，最优解为：

$$\begin{cases}\alpha_i(\varepsilon+\xi_i-y_i+w\times x+b)=0\\ \alpha_i^*(\varepsilon+\xi_i-y_i+w\times x+b)=0\end{cases} \tag{2-17}$$

和

$$\begin{cases}(C-\alpha_i)\xi_i=0\\ (C-\alpha_i^*)\xi_i^*=0\end{cases} \tag{2-18}$$

求解可知，在不敏感区的样本点对应的α_i和α_i^*都为 0，而外部点对应有$\alpha_i=C$和$\alpha_i^*=C$，而在边界上，ξ_i和ξ_i^*都等于 0。

$$\begin{cases}b=y_i-w\times x_i-\varepsilon\ \alpha_i\in(0,C)\\ b=y_i-w\times x_i+\varepsilon\alpha_i^*\in(0,C)\end{cases} \tag{2-19}$$

通过(2-16)(2-17)(2-18)求解，可以计算出 b 的值，与$\alpha_i \neq 0$和$\alpha_i^* \neq 0$对应的样本x_i，即在不敏感区域边界上的样本，叫做支持向量，所以可以把(2-18)改写为：

$$w = \sum_{i=1}^{\infty}(\alpha_i - \alpha_i^*)x_i = \sum_{i \in SVs}(\alpha_i - \alpha_i^*)x_i \tag{2-20}$$

其中 SV 表示支持向量集合，所以

$$f(x) = \sum_{SV}(\alpha_i - \alpha_i^*)\langle x_i \times x \rangle + b$$

2.4.6 非线性支持向量机

非线性支持向量机的基本思想是通过一个非线性映射将数据映射到高维特征空间,在这个空间进行线性回归，其基本形式和线性支持向量机类似，只不过使用核函数来代替向量内积。

$$f(x) = \sum_{SV}(\alpha_i - \alpha_i^*)K(x_i \times x) + b \tag{2-21}$$

$K(x_i \times x) = \phi(x_i)^T \phi(x)$称为核函数。

2.5 本章小结

本章主要介绍了交通流的性质和基本的短期交通流预测模型。并且介绍了统计学习和支持向量机的相关理论，主要包括统计学习的机器学习方法，结构风险最小化，经验风险最小化等理论知识，然后在此基础上面，通过最优分类面引入了线性支持向量机和非线性支持向量机。是对整个统计学习理论和支持向量机理论有一个框架上面的认识，对后面使用最小二乘方法改进支持向量机，打下基础。

第三章 LSSVM 理论及算法

由上所述的支持向量机的理论，可知支持向量机最终是解一个凸二次规划问题，其计算方法相对比较复杂。本文在支持向量机理论的基础上面加入最小二乘方法，将二次规划问题转换为一个线性方程组的问题，简化其求解的过程。同时提出一种新的算法使其能够更新矩阵，使其在保证预测精度的情况下适用于动态预测。

3.1 LSSVM 的基本概念

最小二乘支持向量机(LSSVM)是比利时数学家 Suykens 等人提出对经典 SVM 算法的一种改进。他将标准的 SVM 算法中的不等式约束条件改为等式约束条件，以误差的平方项做为训练的损失函数，把一个凸二次规划问题转化为解一个二次线性方程组[19][20]。对于线性的回归问题采用线性函数

$$f(x)=w\times x+b \tag{3-1}$$

来拟合训练集 $\{(x_i,y_i)\}_{i=1}^{l}\in R^n\times R$，其中 l 为样本数。而在非线性情况下引入函数映射 $\varphi:R^n\to R^m$,所以 LSSVM 回归问题可以表达为：

$$\begin{aligned}&\min J(\omega,\xi)=\frac{1}{2}\|\omega\|^2+\frac{1}{2}\gamma\sum_{i=1}^{l}\xi_i^2\\&s.t.f(x)=\omega\times\varphi(x_i)+b+\xi_i i=1,2,...,l\end{aligned} \tag{3-2}$$

其中 ξ_i 为松弛变量，γ 是用于平衡拟合误差和模型复杂度的正则化参数。

对于优化问题(3-2)，引入拉格朗日函数，最后得到如下的方程式

$$L(\omega,b,\xi_i,a)=J-\sum_{i=1}^{l}a_i\left[\omega\times\varphi(x_i)+b+\xi_i-y_i\right] \tag{3-3}$$

公式中间 $\alpha_i(i=1,.....,n)$ 是拉格朗日乘子。并且由最优化条件，分别就 ω,b,ξ,a，求偏导数，并设为 0，有：

$$\begin{aligned}&\frac{\partial l}{\partial w}=0\Rightarrow w=\sum_{i=1}^{n}\alpha_i x_i\\&\frac{\partial l}{\partial b}=0\Rightarrow\sum_{i=1}^{n}\alpha_i=0\\&\frac{\partial l}{\partial e_k}=0\Rightarrow\alpha_k=Ce_k\quad k=1,...,n\\&\frac{\partial l}{\partial\alpha_k}=0\Rightarrow\omega^t x_k+b+e_k-y_k=0\quad k=1,....,n\end{aligned} \tag{3-4}$$

为了求最优α和b,有KKT条件可以得到

$$\begin{bmatrix} 0 & I^t \\ I & \Omega+\gamma^{-1}I \end{bmatrix}\begin{bmatrix} b \\ \alpha \end{bmatrix}=\begin{bmatrix} 0 \\ y \end{bmatrix} \tag{3-5}$$

其中，$y=[y_1,.....,y_l]^t$；$a=[a_1,.....,a_l]^t$；$I=[I_1,.....,I_l]^t$；Ω为核矩阵，第i行j列的元素为$\Omega_{ij}=\varphi(x_i)^t\varphi(x_j)$,i,j=1,....,l。最后将线性方程组(3-5)解出来的α和b代入(3-1)，可以表出最小二乘回归函数

$$f(x)=w^T\times x+b=\sum_{i=1}^{n}\alpha_i x_i^T+b \tag{3-6}$$

对于非线性问题的求解，与支持向量机一样，我们使用非线性的射影，引入核函数理论，将其转化到高维特征空间，变成高维空间的线性问题求解。也就是在(3-1)和(3-2)式中间的约束条件中通过一个非线性映射$\varphi(\bullet):R^m\to R^h$(这里$m$，$h$)分别表示输入样本空间和高维特征空间的维数，非常明显的是(3-1)式中ω的维数也相应的由m变为h。我们把优化问题映射到到高维特征空间中区讨论，由VC维理论可知，高维特征空间可能是无穷维的。所以对于非线性的映射$\varphi(\bullet)$的实际操作是困难的，并且有可能出现我们并不想看到的结果“维数灾难”。对于我们要研究的支持向量机，(3-4),(3-5)式中对于输入空间的数据点只涉及到其内积运算，所以我们引入能够满足Mercer条件的核函数：

$$K(x_i,x_j)=\left\langle\varphi(x_i)\bullet\varphi(x_j)\right\rangle$$

那么通过一系列的数学运算和公式处理，非线性最小二乘支持向量机模型可以转化为求解一个线性方程组的问题：

$$\begin{bmatrix} 0 & I^t \\ I & K+\gamma^{-1}I \end{bmatrix}\begin{bmatrix} b \\ \alpha \end{bmatrix}=\begin{bmatrix} 0 \\ y \end{bmatrix} \tag{3-7}$$

其中$K_{i,j}=K(x_i,x_j)=\left\langle\varphi(x_i)\bullet\varphi(x_j)\right\rangle$，$i,j=1,....,n$对应于(3-3)中间的第一个等式$\omega=\sum_{i=1}^{n}\alpha_i\varphi(x_i)$，则回归的估计函数为:

$$f(x)=w^T\varphi(x)+b=\sum_{i=1}^{n}\alpha_i K(x_i,x)+b \tag{3-8}$$

由于最小二乘支持向量机把经典的支持向量机的凸二次规划问题的复杂求解转化为一个线性方程组的求解，这极大地简化了计算的复杂度，提高了求解的速度。但是我们也要看到，由于使用了等式约束条件，最小二乘解只能保证局部的最小性，失去了支持向量机的一个重要特性:鲁棒性，并且，由于没有使用ε不敏感损失函数，从(3-4)式中的第三个式子：$\alpha_k=Ce_k$，$k=1,....,n$可以看出来支持向量的拉格朗日乘子的绝对值大小与训练点处的残差成正比，所以最小二乘支持向量机失去了支持向量机的别外一个重要特性：稀疏性。

3.2 LSSVM 的求解方法和算法设计

2000 年左右比利时藉数学家 Suykens 和 Vandwalle 提出了经典的最小二乘支持向量机的基本概念和理论。由于最小二乘方法使得支持向量机的求解过程相比较与经典的支持向量机解法更加的精简和快速。为了使 LS-SVM 的性能发挥到极致，学者们逐渐使用了剪枝法和 W-权向量表出法等来提高解的稀疏性或鲁棒性，使算法尽可能的达到最优。

在第二章中，我们已经通过理论说明，在标准的支持向量机中引入了ε不敏感损失函数后，解就具有稀疏性，但是当我们使用最小二乘这种方法的时候，却失去了这个特性。最后通过研究，比利时籍数学家 Suykens 等提出了剪枝法(Pruning 方法)[19][20]来求近似解。该算法主要的思想就是对于$|\alpha_i|$越小的样本，对函数的估计所做的贡献就越小的事实。对于每一次迭代的结果，剪枝一定比例的α_i(通常是训练数据集对结果最没有影响的 5%的样本),反复的迭代循环，直到最小二乘支持向量机的性能变差为止。

具体的算法实现如下[19]：

(1)输入 N=n 个训练样本点，n 是输入样本个数，对

N 个样本训练最小二乘支持向量机；

(2)对$|\alpha_i|$ (i=1,.....N)进行排序，剪枝 M 个$|\alpha_i|$最小的训练样本(一般来说，$M = N \times 5\%$)；

(3)设$N = N - M$,将剩余的N个训练样本组成新的训练样本；

(4)用最小二乘支持向量机训练削减过后的样本集；

(5)然后，回到第二步，反复迭代，直到最后的运算结果变差为止；

算法的目的是为了增强最小二乘支持向量机的解的稀疏性，但是我们从算法第四步可以看到，算法需要反复迭代所有样本构成的集合以及其部分子集，使其优化。算法的实际效果是以速度的相对损失换来求解的复杂度的降低（增加解的稀疏性），最小二乘支持向量机速度快的特点会相对有所损失。

针对 Suykens 算法的不足， G.C.Cawley 和 N.L.C Talbot 等数学家根据另外一种思路[21][22]—利用有限或者少量样本点完全或者不完全表出ω-权向量来求解最小二乘支持向量机。其基本思想：我们使用(3-3)的第一个表达式和(3-7)的回归估计函数，如果能用有限或者少量的$x_i (i \in S, S \subset \{1,2,...,n\} = I$,且$|S| < n$,$|\bullet|$, 表示元素的个数)，近似表示出$w \approx \sum_{i \in S} \alpha_i x_i$，这样就可以假设$\alpha_j = 0, j \in S$，这样就可以在子集$Q = \{(x_i, y_i) \mid i \in S\} \subset T$上使用最小二乘支持向量机进行训练。由于$|S| < n$,该方法在求解的速度上相比较而言快于剪枝法。算法简称为 ISLS-SVM

该方法存在的关键问题：这样的子集 S 是否存在以及如何确定 S。

对于线性系统而言，大型的数据集一般是可以找到一个子集 S，使得$S \subset \{1,2,...,n\}$且 S 的个数远远小于 I，使得$x_k = \sum \beta x_i$，又因为$w = \sum_{i=1}^{n} \alpha_i x_i$，可以由线性代数的相关知识我们可以推导出$w = \sum_{i \in S} \alpha_i' x_i$，这里$\alpha_i'$可以由$\beta$和$\alpha$线性表出，并且将$\alpha_i'$记为$\alpha_i, i \in S$即可

对于非线性系统而言，将 x_i 换为 $\varphi(x_i)$，由于向量集 $\{\varphi(x_1),....,\varphi(x_n)\}$ 中 φ 是非线性映射，那么求解这样的 S 是很困难的，一般情况下近似的表示为 $w \approx \sum_{i \in S} \alpha_i \varphi(x_i)$。

那么根据 G.C.Cawley 等国外的数学家提出的稀疏解思想，对于非线性系统问题进行一下理论上面的推导。

首先:假设子集 S 已经求出

第二步:

$$\min_{a,b} L(a,b) = \frac{1}{2}\sum_{i,j \in S} \alpha_i \alpha_j K(x_i, K_j) + \frac{C}{n}\sum_{i=1}^{n}\left(y_i - \sum_{j \in S} \alpha_j K(x_i, x_j) - b\right)^2 \tag{3-9}$$

第三步:对(8)求偏导数，并且分别乘以 $\frac{n}{2C}$,最后得到

$$\begin{cases} \dfrac{\partial L}{\partial \alpha_i} = 0 \Rightarrow \sum_{i \in S} \alpha_i \left(\dfrac{n}{2C} K_{ik} + \sum_{j=1}^{n} K_{ij} K_{jk}\right) + b\sum_{i=1}^{n} K_{ik} = \sum_{i=1}^{n} y_i K_{ik} \quad \forall k \in S \\ \dfrac{\partial L}{\partial b} = 0 \Rightarrow \sum_{i \in S} \alpha_i \sum_{j=1}^{n} K_{ij} + nb = \sum_{i=1}^{n} y_i \end{cases} \tag{3-10}$$

最后将(20)写成矩阵的形式，求解就等价于一个 $|S|+1$ 阶方程组的求解。

$$\begin{bmatrix} \mathrm{A} & \mathrm{B} \\ \mathrm{B}^T & n \end{bmatrix} \begin{pmatrix} \alpha \\ b \end{pmatrix} = \begin{pmatrix} \beta \\ \sum_{i=1}^{n} y_i \end{pmatrix} \tag{3-11}$$

其字母表达式为 $\mathrm{A}_{i,j} = \frac{n}{2C} K_{ij} + \sum_{k=1}^{n} K_{ki} K_{kj}, i, j \in S, \mathrm{B}_i = \sum_{j=1}^{n} K_{ij}, i \in S, \beta_i = \sum_{j=1}^{n} y_j K_{ij}, i \in S$

对于 w-权向量来求解最小二乘支持向量机来说，其理论思想要比剪枝法要领先，但是推导的数学公式要复杂不少，那么求解的过程也要复杂，实现起来相对比较困难，所以一般情况下，剪枝法的应用情况要多于 w-权向量。

3.3 算法分析实现及总结

3.3.1 算法分析和实现

对于标准的最小二乘支持向量机而言，就是解(3-6)的线性方程组，其具体的解法是迭代法：有经典的共轭梯度法或 SOR 迭代法，同时我们可以使用 Matlab 工具软件进行快速求导，其基本的计算复杂度为 $o(n^3)$。而对于 Suykens 提出的剪枝法其求解的方法和标准的解法类似，只是要进行反复的剪枝。所以时间的复杂度为 $o(kn^3)$，最后对于 G.C.Cawley 提出的 w-权向量稀疏解法来说，由于只需要很小的样本数据集就可以表示权向量，所以只需要求解 $|S|+1$ 阶的线性方程组(3-10)，故计算复杂度为 $o(|S|^3)$。显然如果

我们能够在样本集里面找到那样的S集合就可以了。

对标准的最小二乘支持向量算法和剪枝算法进行验证

模型介绍：测试样本集输入集

训练样本集$\{X,Y\}$，$X=(-3:0.2:3)^T$，$Y=\sin c(X)+normrnd(0,0.1,length(X),1)$,无误差结果$Y=\sin c(X)$，噪声选择 normrnd(0,0.1,length(X),(均值为 0，标准差为 0.1，length(X)行，1 列)模型的参数 r=10,核函数使用高斯径向基函数，其参数设置为$\sigma^2=0.2$。

标准算法的效果图

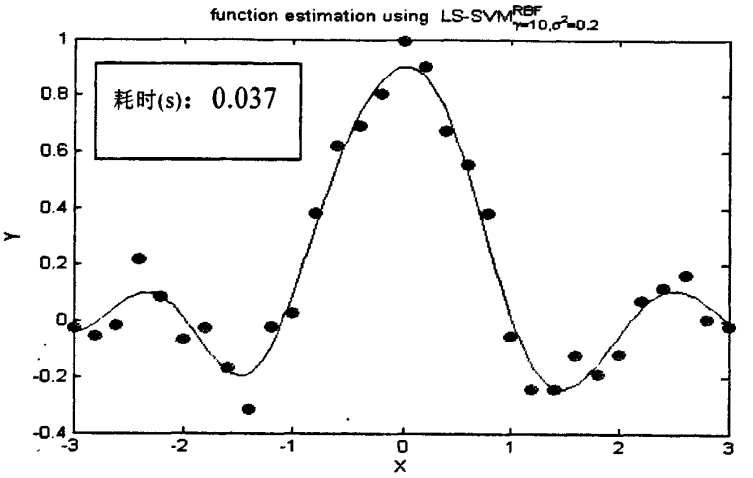

图 3-1 LSSVM 曲线拟合图

剪枝算法实现的效果图：

总共有 31 组训练样本，支持向量列表及误差见附录 表 3-1

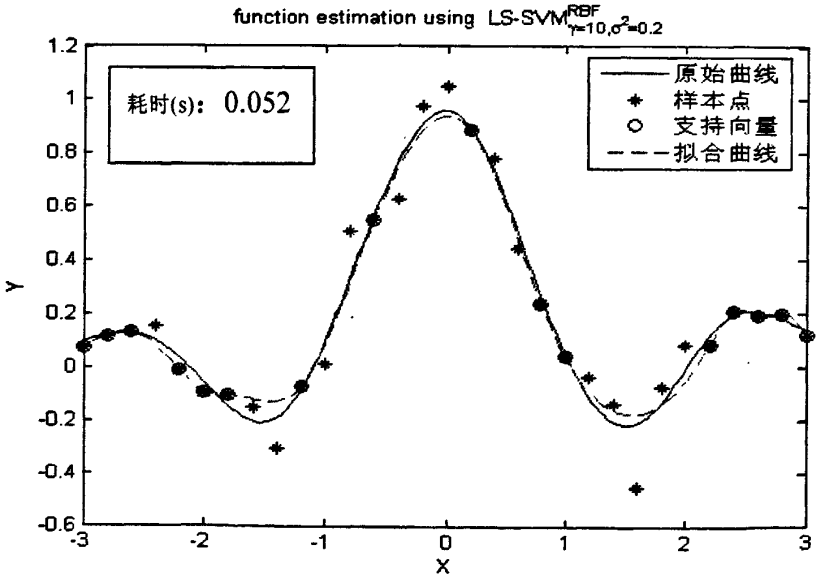

图 3-2“剪枝”后拟合曲线与原曲线比较

分析：

当训练样本为 30 的时候，我们通过标准的剪枝法去求支持向量机时。消耗的时间大于直接拟合曲线，但是我们通过剪枝，减去α值小的的那些支持向量，然后再拟合曲线，最后得出的拟合曲线和我们要求的曲线的形式和趋势是相吻合。达到了我们需要的效果。

下面是在训练区内的数据拟合误差计算和分析,使用如下三种检验方法[43]

平均相对误差：$merr=\frac{1}{N}\sum_{t}\frac{Y_p-Y_r}{Y_r}=0.065$

平均绝对相对误差：$merr=\frac{1}{N}\sum_{t}\frac{|Y_p-Y_r|}{Y_r}=0.107$

最大绝对相对误差：$\max err=\max\left|\frac{Y_p-Y_r}{Yr}\right|=0.406$

3.3.2 算法总结

从误差分析和曲线拟合的图形，我们可以得出。就经典的“剪枝”算法来说，其计算精度是完全满足需求，同时由于要进行"剪枝"计算，其计算过程是要进行反复的排序、迭代故其计算的复杂度要相对增加，在时间的消耗上是略有增加的。从实验上已经证明经典的"剪枝"算法性能优良，时间开销在一个可以控制的范围内，不需要这么多的数据点就可以达到预测拟合效果。

3.3.3 交通流特性与算法应用相结合的分析

根据本文第二章 2.2 节交通流的特性，我们可以知道，交通流一般有如下的几个特点：

1. 实时性：要求对交通流的预测有快速的响应能力
2. 精确性：由于预测是服务于管理和决策，对精度要求较高。
3. 时空相关性：交通流在数学上面可以定义为在时间坐标上面的连续的流数据。

针对最小二乘支持向量机的计算精度较高，计算相对较快（由于是解一个线性方程组的过程）的特点可以满足交通流预测的要求，故可以通过交通流预测的流程建立一个交通流预测的模型，将最小二乘支持向量机方法引入到交通流预测领域。同时由于交通流具有实时性的特点，本文通过引入矩阵理论改进计算的方法使预测模型可以进行动态数据更新，满足交通流预测精度的要求。因为矩阵计算会额外的增加时间和空间的开销影响到算法计算速率和计算精确度，所以在使用矩阵理论改进算法的同时减少矩阵求逆的次数和求逆矩阵的个数，以保证计算时间复杂度不至于太大，从而影响模型的效率。对于经典“剪枝”法在计算中保证计算精度的优点在改进算法中也要得到保留。

3.4 算法的改进(在线"剪枝"解法)

3.4.1 建立在线模型的必要性分析

对于现存的预测模型来说，大多数文献[19,58,59]建立系统的离线预测模型形式，然后通过优化算法进行在线计算，但是在实际的工作中，系统的数域的变化往往使得离线模型不能够准确的描述系统的实际状况，从而不能做出准确的预测。为了能使模型能够准确的反应系统的当前状态，达到实时预测的目的，应该使用不断获取的最新数据建立起反映系统当前状态的最新模型，也就是在线模型。

文献[60]提出了一种在线的高斯学习方法，其主要思想：在训练过程中，当增加一个训练样本时，对应的训练参数的维数增加一维，同时使用为增加时的样本的线性组合来表示当前的参数。为了保证训练过程的运动性，在增加样本的时候就删除一对基向量，保证参数的维数不变。文献[61]对这种在线的学习方法做了改进，通过对地基测量问题的仿真表明新的算法能够准确、快速地预测非线性、时变函数。基于这种思想，对于经典的 LSSVM 的剪枝算法而言，其计算的时间消耗较少，同时其预测精度也令人满意。综合上面所述，将在线的思想引入到 LSSVM 的训练过程并建立起完整的预测模型。使这种模型能够在保持其计算时间少，精度高等特点的同时，还能够完成在线的预测功能。

3.4.2 在线模型的算法理论

算法是在"剪枝"算法的基础上，使用类似“滑动窗口”的概念，进行在线的更新预测模型训练样本逐渐的向未知逼近，达到预测的目的。

对于增量式学习来讲，样本集是递增的，样本集$\{(x_i,y_i)\}i=1,,,t$(t表示一个自然数)随着时间 t 的递增，学习的样本集可以表示为$\{x(t),y(t)\}$，其中，$x(t)=[x_1,x_2,,,,,x_t],y(t)=(y_1,y_2,,,,y_t)^T,x_t\in R^n,y_t\in R$。这样的核矩阵$Q$，待求的拉格朗日乘子$\alpha$和常偏差$b$都是 t 的函数，表示如下:

$$Q_t(i,j)=k(x_i,x_j),i,j=1,2,...t \qquad \alpha(t)=(\alpha_1,\alpha_2,...,\alpha_t)^T,b(t)=b_t$$

得到表达式

$$f(x)=\sum_{k=1}^{N}\alpha_k K(x_k,x)+b \tag{3-12}$$

最后将要求的最小二乘矩阵化简成如下的形式

$$\begin{bmatrix}0 & 1^T\\ 1 & K_N+U_N\end{bmatrix}\bullet\begin{bmatrix}b\\ \alpha\end{bmatrix}=\begin{bmatrix}0\\ Y_N\end{bmatrix} \tag{3-13}$$

令$P_N=K_N+U_N$，求解方程组得到模型的系数

$$\begin{cases} b = \dfrac{1^T P_N^{-1} Y_N}{1^T P_N^{-1} 1} \\ \alpha = P_N^{-1}(Y_N - 1 \times b) \end{cases} \tag{3-14}$$

$$P_N = \begin{bmatrix} k(x_1,x_1) & ..... & k(x_1,x_N) \\ ..... & ..... & .... \\ k(x_N,x_1) & ..... & k(x_N,x_N) \end{bmatrix} + diag\left(\frac{1}{ru_1},\frac{1}{ru_2},.....,\frac{1}{ru_N}\right) \quad k(x_i,x_i)=1$$

$$= \begin{bmatrix} 1+\dfrac{1}{ru_1} & k(x_1,x_2) & ... & k(x_1,x_N) \\ k(x_2,x_1) & 1+\dfrac{1}{ru_2} & ... & k(x_2,x_N) \\ ... & ... & ... & ... \\ k(x_N,x_1) & k(x_N,x_2) & ... & 1+\dfrac{1}{ru_N} \end{bmatrix} \tag{3-15}$$

对于要实现在线的更新训练样本，我们可以使用类似"滑动窗口"概念的技术，进行增量计算。

下面引入两个矩阵的理论

引理 1[23][24]

设矩阵 $A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}$,若矩阵$A$可逆，则其逆矩阵为：

$$A^{-1} = \begin{bmatrix} (A_{11} - A_{12}A_{22}^{-1}A_{21})^{-1} & -A_{11}^{-1}A_{12}(A_{22} - A_{21}A_{11}^{-1}A_{12})^{-1} \\ -A_{22}^{-1}A_{21}(A_{11} - A_{12}A_{22}^{-1}A_{21})^{-1} & (A_{22} - A_{21}A_{11}^{-1}A_{12})^{-1} \end{bmatrix} \tag{3-16}$$

引理 2[23][24]

若矩阵 $A, C, A+BCD$是可逆的，则

$$(A+BCD)^{-1} = A^{-1} - A^{-1}B(DA^{-1}B + C^{-1})^{-1}DA^{-1}$$

把加号变成减号，变形得到：

$$(A-BCD)^{-1} = A^{-1} + A^{-1}B(DA^{-1}B - C^{-1})^{-1}DA^{-1} \tag{3-17}$$

由(3-16)(3-17)可以得到

令 $P_{N+1} = \begin{bmatrix} P_N & V_{N+1}^T \\ V_{N+1} & u_{N+1} \end{bmatrix}$, $\begin{cases} V_{N+1} = [k(x_1,x_{N+1}),.....,k(x_N,x_{N+1})]^T \\ u = k(x_{N+1},x_{N+1}) + \dfrac{1}{ru_{N+1}} \end{cases}$，由矩阵的理论可以推导出

$$P_{N+1}^{-1} = \begin{bmatrix} P_N & V_{N+1}^T \\ V_{N+1} & u_{N+1} \end{bmatrix}^{-1} = \begin{bmatrix} \rho_1^{-1} & -P_N^{-1}V_{N+1}^T\rho_2^{-1} \\ -u_{N+1}^{-1}V_{N+1}\rho_2^{-1} & \rho_2^{-1} \end{bmatrix} \tag{3-18}$$

对于 P_{N+1}^{-1} 可以由 P_N^{-1} 及核函数矩阵表示出来，只需要计算 P_N，P_N^{-1}，u_{N+1}^{-1}，V_{N+1}^T，V_{N+1} 无需对 P_{N+1}^{-1} 求逆，而 u_{N+1}^{-1}，V_{N+1}^T，V_{N+1} 或者是一维的向量，或者是一个数，所以降低了计算的难度，最后将 P_{N+1}^{-1} 代入(3-14),然后再代入(3-16)就可以得到我们需要的结果。

上面是对矩阵做增量算法，当我们增加了一个元素并通过计算得到我们想要的结果后，为了保持矩阵的维数大小不变，我们将要删除一个对系统影响最小的支持向量。

对于 N+1 维矩阵来说，可以把它改写成如下的形式：

$$P_{N+1}=\begin{bmatrix} f_{N+1} & F_{N+1}^T \\ F_{N+1} & P_N^* \end{bmatrix} \tag{3-19}$$

$$f_{N+1}=k(x_1,x_1)+\frac{1}{ru_1},\quad F_{N+1}=[k(x_2,x_1),.....,k(x_N,x_1)]^T$$

$$P_N^*=\begin{bmatrix} 1+\frac{1}{ru_2} & ... & k(x_2,x_N) \\ ... & ... & ... \\ k(x_{N+1},x_2) & ... & 1+\frac{1}{ru_{N+1}} \end{bmatrix}$$，设 p_N^* 为删除一组数据后的矩阵，Y_N^* 为相应剩余数据，只要求出 p_N^* 就可以更新系统模型。

$$P_{N+1}^{-1}=\begin{bmatrix} Q_1 & q_{1,N} \\ q_{1,N}^T & Q_N \end{bmatrix},\quad Q_1\in R^{1x1},q_{1,N}\in R^{1XN},Q_N\in R^{NXN} \tag{3-20}$$

$$\begin{bmatrix} f_{N+1} & F_{N+1} \\ F_{N+1}^T & P_N^* \end{bmatrix}=\begin{bmatrix} Q_1 & q_{1,N} \\ q_{1,N}^T & Q_N \end{bmatrix}^{-1}=\begin{bmatrix} (Q_1-q_{1,N}Q_N^{-1}q_{1,N})^{-1} & -Q_1^{-1}q_{1,N}(Q_N-q_{i,N}^TQ^{-1}q_{1,N})^{-1} \\ -Q_N^{-1}q_{1,N}^T(Q_1-q_{1,N}Q_N^{-1}q_{1,N}^T)^{-1} & (Q_N-q_{1,N}Q_1^{-1}q_{1,N})^{-1} \end{bmatrix}$$

由此就可以得到：

$$P_N^*=(Q_N-q_{1,N}^TQ_1^{-1}q_{1,N})^{-1}，\text{化简得到}\ P_N^{*-1}=(Q_N-q_{1,N}^TQ_1^{-1}q_{1,N}) \tag{3-21}$$

然后将其带入(3-14),再带入(3-16)式就可以得到删除一个元素后的 N 为矩阵的值。

3.4.3 算法的实现

上一节是对算法的理论推导，接下来是对理论进行具体的实现..。其步骤如下：

(1) 先确定时间 T。通过部分样本 $(x_1,y_1),(x_2,y_2),....,(x_i,y_i)$ 先计算出核矩阵(稀疏矩阵)，确定其大小

(2) 在一个时间限内，采样得到新的输入输出值(x_k, y_k)，k 表示当前总的数据量，如果$k \le N$,则算法执行(3);如果$N \le k \le N+1$,则算法执行(4),如果$N+1 \le k$，则算法执行(5)

(3) 由(3-15)计算P_N，求出P_N^{-1}代入(3-14),更新模型参数b, α.。然后执行(2),并取$k = k+1$。

(4) 由(3-16)和(3-18)计算出P_{N+1}^{-1}，然后带回(3-14)更新模型参数b, α，执行(2)

(5) 由(3-19)得到P_N^{-1}，然后由(3-20)把P_N^{-1}分解成$P_{N+1}^{-1} = \begin{bmatrix} Q_1 & q_{1.N} \\ q_{1.N}^T & Q_N \end{bmatrix}$，最后由(3-20),(3-21)计算删除一组数据后的P_N^{*-1}，并带入(3-14)，更新模型参数b, α，删除样本集合中的一组数据。执行(2)

(6) T 时间到，程序停止。

为了完整的证明在线"剪枝"算法的整个过程，下面设计了两个模型进行算机仿真实验。

模型 1 介绍：

测试样本集输入集训练样本集$\{X, Y\}$，$X = (-5:0.2:5)^T$，$Y = \sin c(X) + normrnd(0, 0.1, length(X), 1)$，无误差结果$Y = \sin c(X)$，噪声选择 normrnd(0,0.1,length(X),(均值为 0，标准差为 0.1，length(X)行，1 列)模型的参数 r=10,核函数使用高斯径向基函数，其参数设置为$\sigma^2 = 0.2$。模型的参数 r=10,共有 80 组训练样本。

算法的效果图如下:

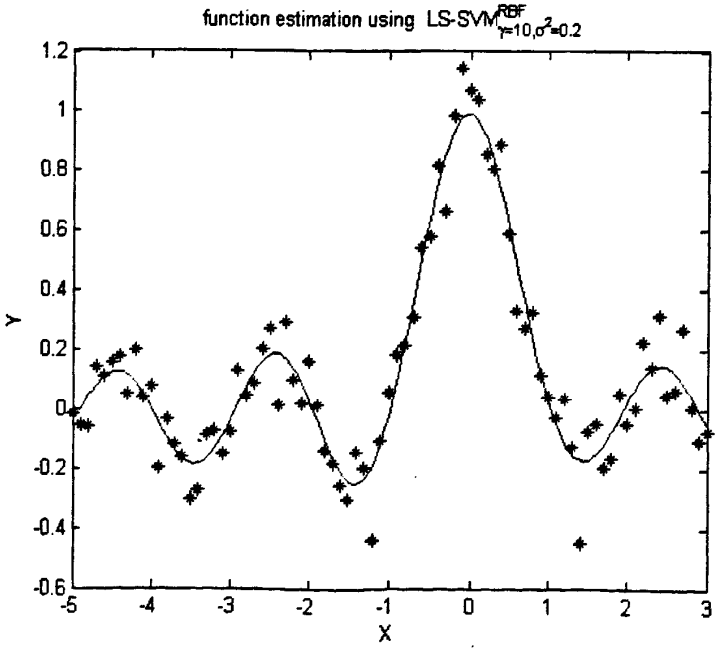

图 3-3

先通过 80 个样本点将曲线拟合出来。

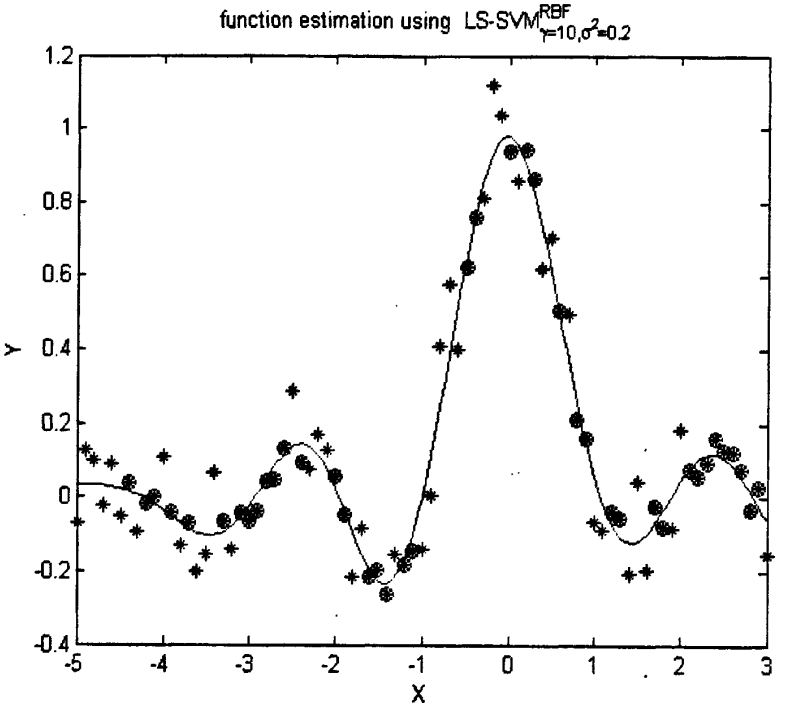

图 3-4

然后从前 80 个样本中找出其中的支持向量。支持向量的个数不大于 40 个

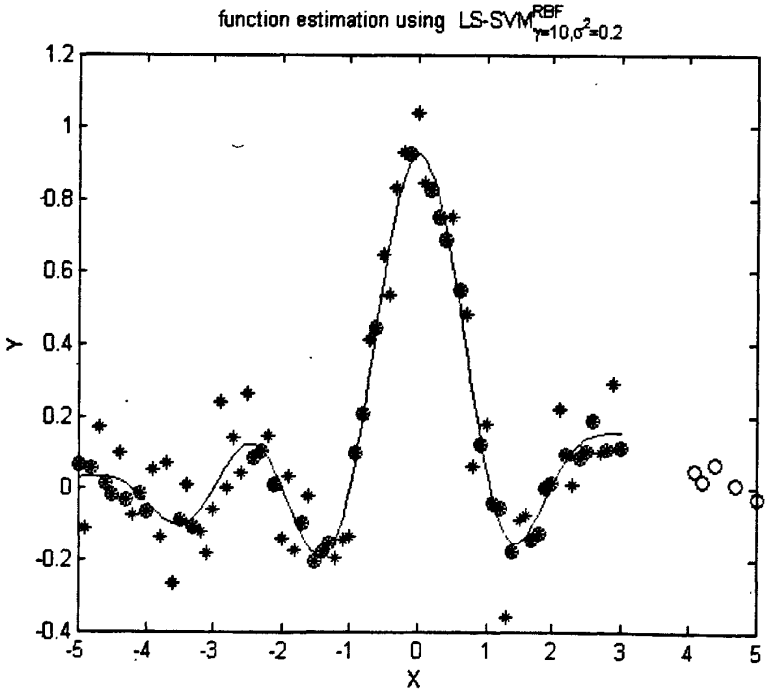

图 3-5

最后通过后 20 个样本点，将余下的支持向量找出来。

上面的实验可以清楚的表明在线"剪枝"算法进行动态"剪枝"的过程

第一步:样本输入

第二步:建立计算模型，并找出支持向量

第三步:通过在线"剪枝"模型，将新输入的样本点代入模型进行计算得到最新的"剪枝"支持向量。

模型 2 介绍:

测试样本集输入集训练样本集 $\{X,Y\}$，$X=(-5:0.2:5)^T$，$Y=\sin c(X)+normrnd(0,0.02,length(X),1)$，无误差结果 $Y=\sin c(X)$，噪声选择 normrnd(0,0.1,length(X),(均值为 0，标准差为 0.02，length(X)行，1 列)模型的参数 r=10,核函数使用高斯径向基函数，其参数设置为 $\sigma^2=0.2$。模型的参数 r=10,共有 80 组训练样本,20 个检测样本。

算法的效果图如下:

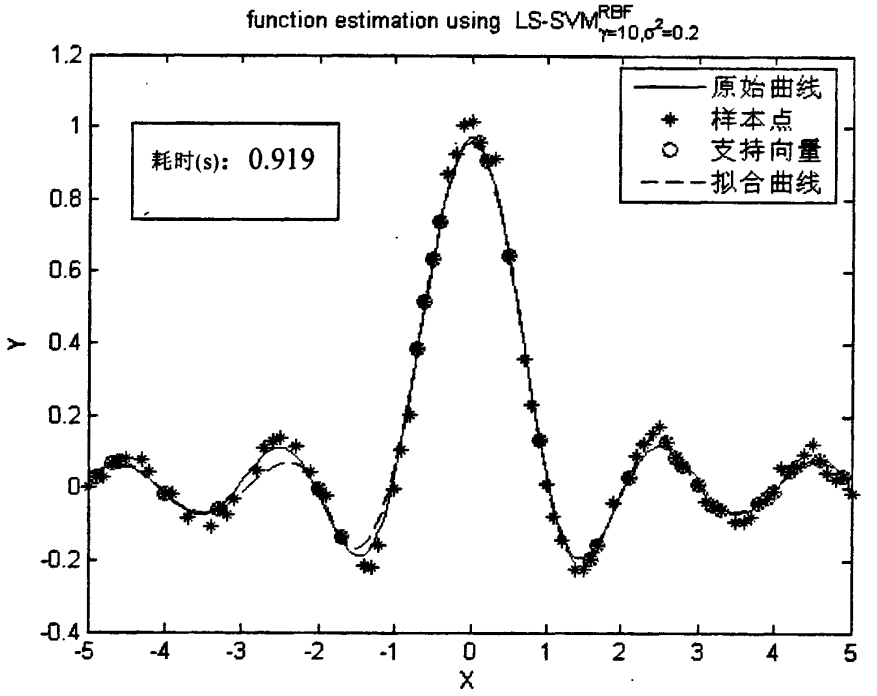

图 3-6“新”算法拟合曲线效果图

最后将样本点中所有的支持向量找出来，拟合出曲线，达到预测功能。

误差分析：最后得到的支持向量和最终的分析结果

表 3-2 剪枝后支持向量列表预测误差

X	Y(计算值 r)	Y(测量值 p)	误差(Yp-Yr)	误差((Yp-Yr)/Yr)
3	0.0112	0.0098	-0.0014	-0.125
3.2	-0.0436	-0.0436	0	0
3.3	-0.0595	-0.0605	-0.001	-0.016
3.8	-0.039	-0.039	0	0
3.9	-0.0259	-0.026	0.001	0.0038
4	-0.0101	-0.0121	-0.002	0.198
4.2	0.0474	0.0494	0	0
4.3	0.0625	0.0625	0	0
4.6	0.0811	0.0775	-0.0036	-0.0443
4.9	0.0355	0.0312	-0.0043	-0.1211

可以看出在向前推测的预测趋势反应出变化趋势，从图上就可以看出误差情况比较理想，我们把训练样本点的输入作为预测的输入，预测输出一般也基本和训练样本的输出相符合，虽然通过样本训练的拟合曲线一般和原始曲线是不完全重合，但是实验的效果却较好，可以表明支持向量机的拟合预测适合小样本、数据比较密集数据集的预测。

下面是数据误差计算和分析

平均相对误差：$merr=\frac{1}{N}\sum_{t}\frac{Y_{p.}-Y_r}{Y_r}$=-0.0104

平均绝对相对误差：$merr=\frac{1}{N}\sum_{t}\frac{|Y_p-Y_r|}{Y_r}$=0.05

最大绝对相对误差：$\max err=\max\left|\frac{Y_p-Y_r}{Yr}\right|$=0.198

3.5 与 RBF 神经网络预测结果的比较

文献[65]将神经网络这种人工智能的方法引入到城市交通流预测中，通过对 BP 神经网络，RBF 神经网络，GRNN 神经网络三种神经网络的对比得出结论：相对于其他两种神经网络模型，RBF 神经网络模型在计算时间和预测精度方面两方面综合起来预测效果最优。文献[45]将 LSSVM 方法引入土木工程测量方面，通过建立离线最小二乘支持向量机模型，使用等步长预测优化方法进行滚动在线预测，与 RBF 神经网络进行对比试验，得出结论：支持向量机模型在预测效果方面要好于 RBF 神经网络。

针对文章提出的在线"剪枝"方法，使用 RBF 神经网络模型进行对比试验，最后通过实验的结果来证明在线"剪枝"方法的预测效果的好坏。

(1) RBF 神经网络模型[43]

RBF 神经元模型结构如图所示。RBF 基函数神经元的变换函数为高斯函数，其输出值为高斯函数的计算输出值，输入值为输入和权值向量之间的$\|dist\|$距离作为输出

$$R(\|dist\|) = e^{-\|dist\|^2}$$

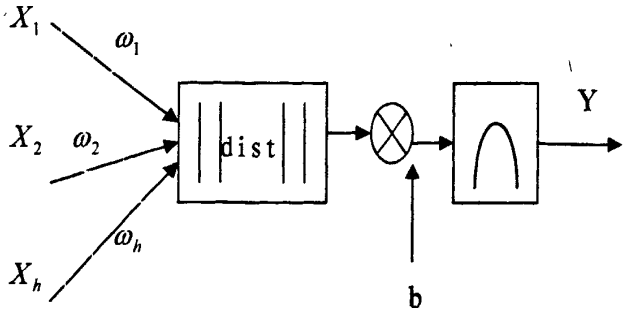

图 3-7 有 h 个 RBF 函数神经元

RBF 函数网络是由输入层，隐含层，输出层构成的三层前向网络，隐含层是以 RBF 函数为激励函数，一般为高斯函数。

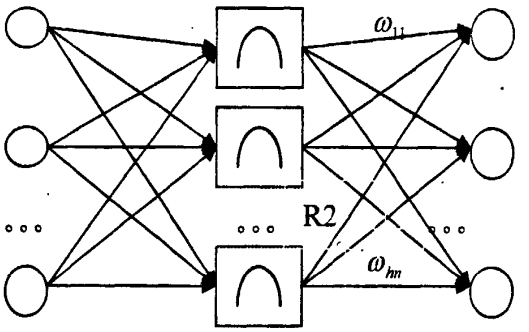

图 3-8 RBF 神网络结构

隐含层每个神经元与输入层相连的权值向量 WLi 和输入向量 Xi(表示第 i 个输入向量)，之间的距离乘上阈值作为本身的输入。

第 i 个神经元的输入：$k_i^q = \sqrt{\sum_j (w1_{ji} - x_j^q)} \times b1_i$

第 i 个神经元的输出：$\exp(-(\|w1_i - x^q\| \times b1_i)^2)$

(2) RBF 神经网络仿真比较

使用 RBF 神经网络可以用于函数逼近，使用 Matlab 工具包 Newrb 可以从训练样本中快速的建立一个 0 误差的预测模型。本文使用 RBF 神经网络和最小二乘向量机预测过程进行比较。

为了比较的公平，本文选用训练样本输入与上面最小二乘支持向量机的实验相同。

表 3-3 RBF 神经网络预测误差

X	Y(计算值 r)	Y(测量值 p)	误差(Yp-Yr)	误差((Yp-Yr)/Yr)
3	0.0112	0.0103	-0.0009	-0.08
3.2	-0.0436	-0.0421	0.0015	-0.034
3.3	-0.0595	-0.0611	-0.0016	0.026
3.8	-0.039	-0.0407	-0.0017	0.0435
3.9	-0.0259	-0.0294	-0. 0035	0.1351
4	-0.0101	-0.0142	-0.0041	0.4059
4.2	0.0474	0.0494	0.002	0.0421
4.3	0.0625	0.0602	-0.0023	-0.0368
4.6	0.0811	0.0793	-0. 0018	-0. 0221
4.9	0.0355	0.0394	0. 0039	0. 1098

平均相对误差：$merr = \frac{1}{N}\sum_{t}\frac{Y_p - Y_r}{Y_r} = 0.058$

平均绝对相对误差：$merr = \frac{1}{N}\sum_{t}\frac{|Y_p - Y_r|}{Y_r} = 0.093$

最大绝对相对误差：$\max err = \max\left|\frac{Y_p - Y_r}{Yr}\right| = 0.4059$

表 3-4 两种预测模型的结果对比

预测方法	平均相对误差	平均绝对相对误差	最大绝对相对误差
基于"剪枝"LSSVM 预测方法	0.0104	0.05	0.198
基于 RBF 神经网络预测方法	0.058	0.093	0.406

通过对 LSSVM 的剪枝算法和 RBF 神经网络的仿真实验，由表 3-4 可知，无论是平均相对误差、平均绝对相对误差还是最大绝对相对误差使用最小二乘支持向量机的预测结果要好于使用 RBF 神经网络。

3.5 本章小结

本章首先介绍了最小二乘支持向量机的基本原理，然后通过公式的推导，得出一般的最小二乘支持向量机的模型，接着又介绍了 LSSVM 算法的计算步骤，提出了两种基本的算法，剪枝法和 w-权向量法。最后在剪枝法的基础上面，提出了一种类似“滑动窗口”概念可以进行在线更新稀疏解法，并且通过仿真实验，对结果进行了验证，

同时，我们也从前人的经验可以得出，虽然 G.C.Cawley 等人提出的 w-权向量法在理论上面已经证明相对于剪枝法要更为的有效，但是其算法却相对更加复杂，特别是要在高维空间中去找一组基向量来做近似计算是比较困难的，同时还要在低维空间和高维

空间中进行映射计算，所以该方法的实际使用并不广泛，这样一个问题要需要在以后的研究中区解决。

当然对于最小二乘支持向量机的研究还有许多方法需要我们去深入研究，比如现在对于核函数的参数的选择，在业界就有很多方法运用于其中，比如说在参数的选择中使用粒子群算法或者小波技术等

第四章 基于 LSSVM 的短时交通流预测模型

交通流预测大致分为以下四个部分：

(1)采集交通流原始数据

(2)交通数据的预处理/数据的清洗

(3)将处理过后的数据输入预测模型

(4)将得到的数据应用到各种交通管理系统中

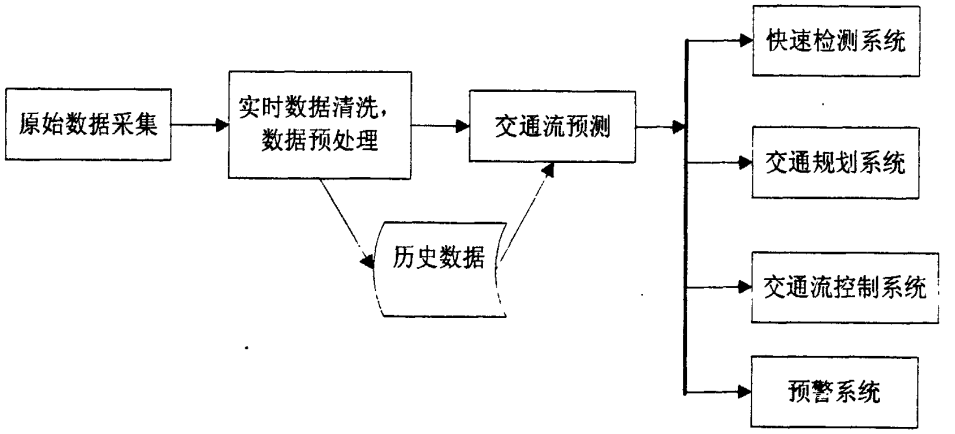

图 4-1 交通流预测流程示意图

本章在总结了交通流的数据流程和改进的支持向量机模型的基础上，重点对交通流数据的清洗和预测进行了研究，将最小二乘支持向量机引入到预测模型里面，并对其解法进行改进，最后通过实验进行验证。

4.1 交通流预测的流程

4.1.1 最小二乘支持向量机的预测过程

使用 LSSVM 来进行预测的一般过程也就是求解一个线性方程组的问题，确定其中的参数，并不断的更新参数，拟合出曲线，通过曲线发展的趋势去做预测功能

一般的流程图如下：

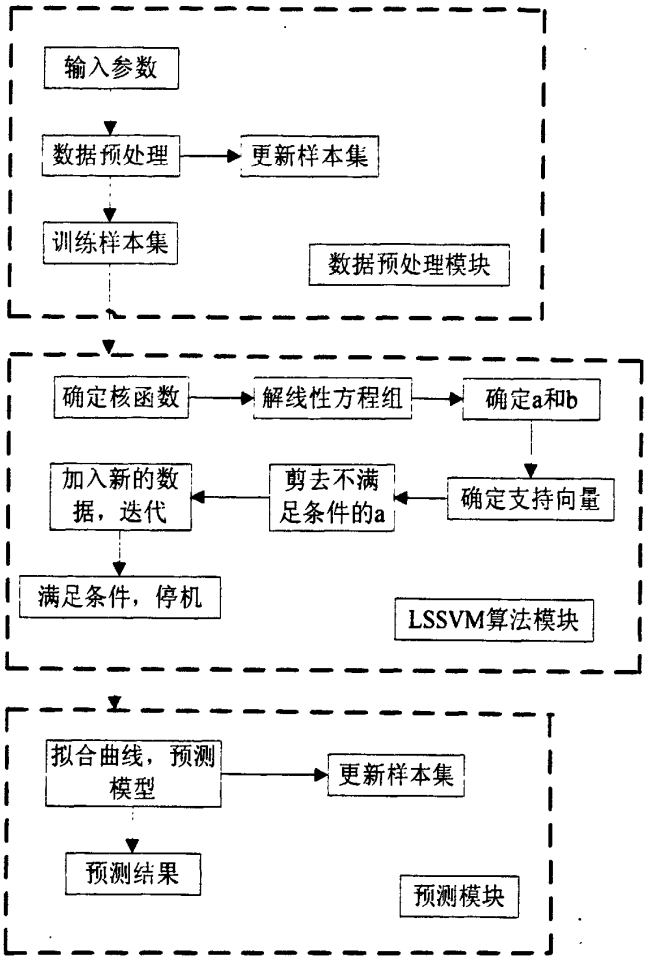

图 4-2 最小二乘支持向量机预测流程图

4.1.2 实时数据的采集

原始数据的采集是通过各种仪器在某一条路段上面实时的采集单位时间段内，有多少辆小汽车通过，然后将数据记录在电脑中（具体过程略）。

4.1.3 实时数据的清洗

我们对从检测器获得的动态数据进行分析可以知道，使交通实时数据出现偏差的情况有两个主要的方面，首先就是检测器或者传输线路等硬件方面出现故障而引起数据的错误，这种情况下数据一般是丢失或是数据失真。别外一种情况是由于道路中突然出现的各种事故对当时的交通状况产生了干扰，出现了异常的波动，导致大量的噪声。

(1)缺失值补偿

由于交通流的数据通常不是完全离散而相互孤立的，所以有一个渐进的连续变化过程。每一个相邻时间点的数据都能够反应出该时间点上面交通流数据的特性。根据交通

流相似原理估计，也可以根据相关缺失项时间段内交通流量的变化趋势估计当时的交通流量，可以使用缺失值去取代正确值，从而保证有完整的数据。我们可以设定一个阈值来表示缺失数据的个数，当小于阈值，就使用前一个时刻的数据进行代替；当等于阈值，使用一天前的数据来修补。

(2)异常值纠正

计算路口不同时间段交通流量的数学期望 E(x)和方差 D(x),采用限幅滤波法对异常值进行修正[18]

$$\bar{x}_n = \begin{cases} E(x)+D(x) & |x_n - E(x)+D(x)| > a \\ x_n & |x_n - E(x)+D(x)| \le a \end{cases} \tag{4-1}$$

其中，x_n是 n 时段的流量数据，a 为限幅范围。

(3)去噪

由于交通流量的扰动性很明显，其数据变化有了很大的噪声。由实际统计得知，在一个观察路口流量远没有饱和的时候，相邻两个信号的流量能够达到 78%的变化，一般情况下能够在 20%以内[18]。文章用某一个信号时刻为中心的几个信号的平均值作为该信号的流量数据，即采用算术平均滤波方法实现噪声去除。

$$\bar{x}_n = (x_{n-2} + x_{n-1} + x_n + x_{n+1} + x_{n+2})/5$$

其中，n为流量数据的个数，x_n为该时刻流量的记录值，$\bar{x}_n$为该时刻的实际值。

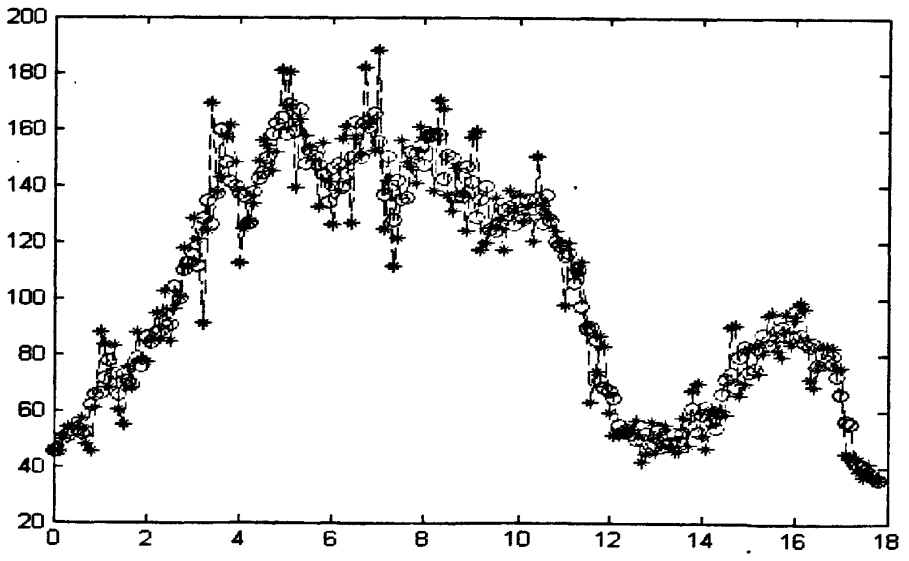

图 4-3 数据清洗示意图（十字表示原数据点，圆圈代表清洗后的点）

(4)提取数据

由于时间序列变化特点规律是在交通流量中存在的，路段上面前几个时间段的流量和当前的流量有着必然的联系。所以就可以利用当前路段的前几个时间段的流量数据去做预测，设$v_i(t)$表示路段 t 时刻的交通流量。我们采用当前时段和前 n 个时间段的交通

流量作为样本，进行输入和输出。

4.1.4 建立 lssvm 预测模型

(a)确定核函数

在第二章的最后，我们引入了核函数的概念，现在我们详细的论述核函数与它的概念。

[1]对于给定输入空间中的样本集：

$(x_1,y_1),(x_2,y_2),...,(x_i,y_i),x_i \in R^N, y_i \in R,(i=1,2....,l)$ 对于这些数据存在一个映射 $\phi(x)$，满足关系:$R^N \to H: x_i \to \phi(x_i)$,在输入空间中存在一个函数 $K(x_i,x_j)$,而且它可以表示为从输入空间到特征空间的映射函数 $\phi(x)$ 的内积，也就是 $K(x_i,x_j)=\{\phi(x_i),\phi(x_j)\}$,我们称为核函数。

一般情况下，可以使用以下几种函数为核函数：

1) 多项式核函数：

$K(x,y)=((x\cdot y)+c)^d$，其中 $c \geq 0, d$ 是任意正整数 (4-2)

2) 高斯径向基核函数：

$$K(x_i,x)=\exp\left\{\frac{-\|x-x_i\|^2}{2\sigma^2}\right\} \quad i=1,2,\ldots,s \tag{4-3}$$

3) 傅里叶核函数：

$$K(x,y)=\frac{1-q^2}{2(1-2q\cos(x-y)+q^2)}$$，其中，$x,y \in R, 0<$常数$q<1, q$为 (4-4)

任何函数只要满足 Mercer 条件，都可以用来作为支持向量机的核函数，采用不同的函数为核函数，可以构造实现输入空间中不同类型的非线性决策面的学习机器[25]。最常用的核函数是 RBF 高斯径向基核函数。

(b)建立模型的方程，求解参数 α,b,确定支持向量，并剪枝

由第三章，我们已经从最小二乘向量机的理论中推出了最后的结论，就是求解一个线性方程组，其形式如下：

$$\begin{bmatrix} 0 & I^t \\ I & K+\gamma^{-1}I \end{bmatrix}\begin{bmatrix} b \\ \alpha \end{bmatrix}=\begin{bmatrix} 0 \\ y \end{bmatrix} \tag{4-5}$$

$K_{i,j}=K(x_i,x_j)=\langle \varphi(x_i)\bullet\varphi(x_j)\rangle$, $i,j=1,....,n$，Ω为核矩阵，第 i 行 j 列的元素为 $\Omega_{ij}=\varphi(x_i)'\varphi(x_j)$,i,j=1,....,l。我们将数据输入到核函数中，再将核函数组合成核矩阵，最后转化为一个线性方程组，也就是上面的方程组形式。解这个方程组得到 α,b 其中 α 就是我们要求得的支持向量，取得支持向量后，我们依据 $|\alpha|$ 越小，对回归支持函数的贡献越小的思想，“剪枝”一定比例的 α。缩减一定比例的样本点(一般情况是 5%),然后反复

迭代，直到算法性能变差为止。

(c)拟合曲线，评价结果

根据求出的支持向量α，然后拟合出趋势曲线，并通过误差分析得出最后的结论，也就是通过最小二乘法求解支持向量，然后“剪枝”一部分多余的支持向量，用剩下的支持向量去拟合成曲线在满足一定误差的情况下去做预测。

4.1.4 评价指标

为了评价标准算法和改进后的算法的预测性能和比较仿真实验的结果，我们使用一下三个指标来说明[43]

(1) 平均相对误差：$merr = \frac{1}{N}\sum_{t}\frac{Y_p - Y_r}{Y_r}$ (4-6)

(2)平均绝对相对误差：$merr = \frac{1}{N}\sum_{t}\frac{|Y_p - Y_r|}{Y_r}$ (4-7)

(3)最大绝对相对误差：$\max err = \max\left|\frac{Y_p - Y_r}{Yr}\right|$ (4-8)

4.2 实验 1 结果和分析

4.2.1 结果分析

实验选用加州伯克利大学，交通工程系 carlos Doganzo 教授个人网站上面(http://www.ce.berkeley.edu/~daganzo/index.htm)，1998 年美国 San Pablo Dam 高速公路十二号出口，下午两点到晚上 8 点之间的交通流量的数据值,每隔两分钟采集一次交通流量数据。先把开始时间 14:02:00 设置为 0.1,然后将 14:04:00 设置为 0.2，直到最后将 9:58:00 转化为 17.8 为止。(X 轴代表时间坐标，Y 轴代表车流量大小)

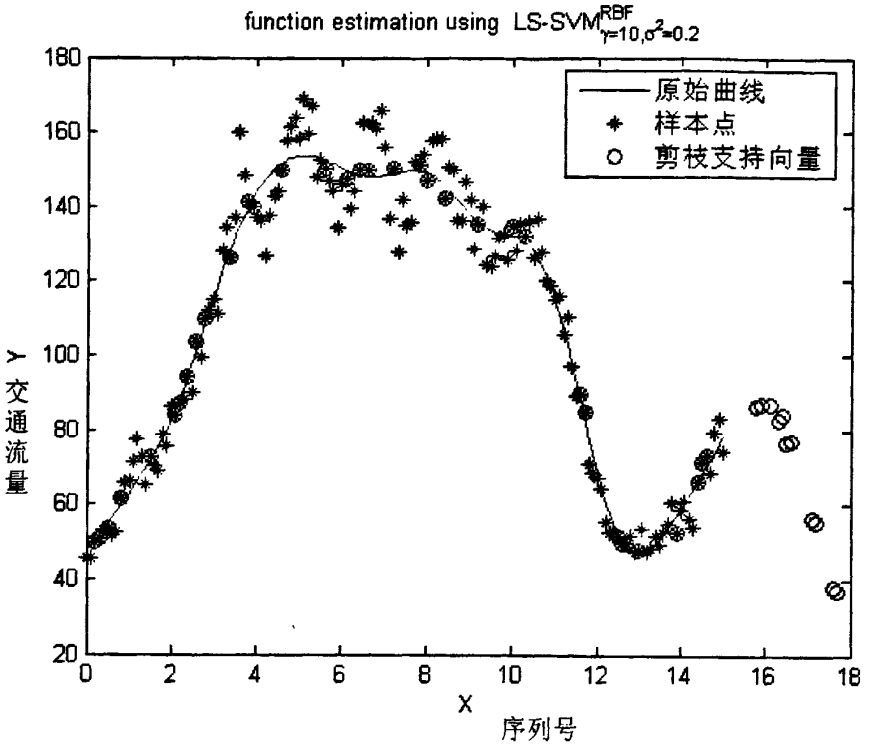

图 4-4 找出整个区间的支持向量

上图先使用样本点，先建立模型，取出支持向量，拟合出趋势曲线，然后在把检测样本带入已经建立好的模型，得出新的支持向量，通过这些新的支持向量，我们可以得到预测值，和将要拟合的曲线的趋势。

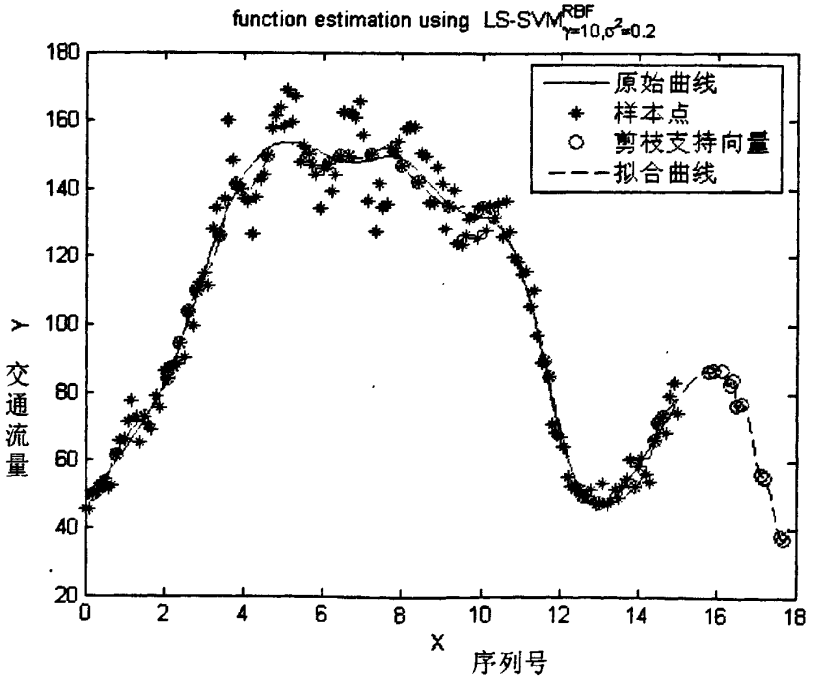

图 4-5 通过测试样本中选出的支持向量拟合出预测曲线

通过对预测值(也就是预测的支持向量)进行曲线的拟合，可以得出预测的趋势。达到预测的效果。

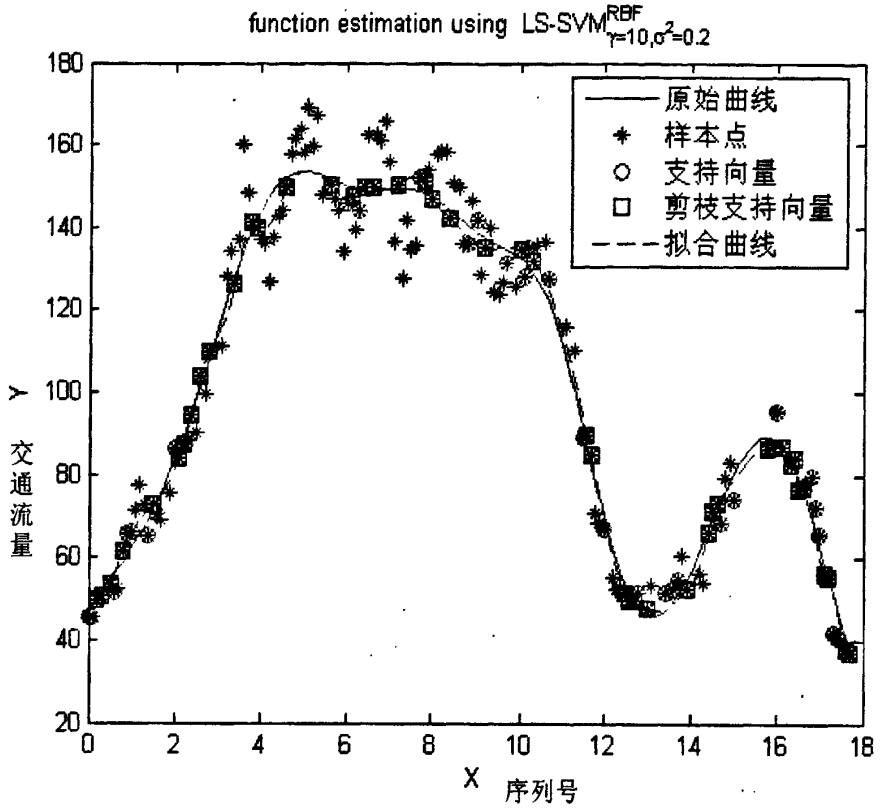

图 4-6 完成整个在线"剪枝"预测过程

最后通过在线的剪枝算法，去掉对曲线影响小的支持向量，建立模型，最后通过模型计算检测要使用的预测样本，取出其中的支持向量，得到离散点运动的趋势，把模型计算出的所有支持向量拟合成曲线，就达到了预测的效果。通过第三章的理论证明，这种算法最后得出的结果是可以接受的，但是当数据量相对较大的，其计算的时间开销也会相应的上升。时效比的效果就不会向相对较小数据集是那么明显。

4.2.2 误差处理

下表是模型计算后的支持向量

相对误差公式：$Y_{计算值} - Y_{真实值}$

绝对误差公式：$\dfrac{Y_{计算值} - Y_{真实值}}{Y_{真实值}}$

训练区间 37 个支持向量列表及误差见附录 表 4-1。

我们从效果图和训练区的数据分析表中可以看出来，在训练区间也就是用来建立模型的区间，数据的补漏预测也就是拟合曲线与原始曲线之间，误差效果比较好。

这里先进行训练区间内的数据预测和误差计算分析

(1) 平均相对误差: $merr = \frac{1}{N}\sum_{t}\frac{Y_p - Y_r}{Y_r}$=0.102

(2) 平均绝对相对误差: $merr = \frac{1}{N}\sum_{t}\frac{|Y_p - Y_r|}{Y_r}$=0.006

(3) 最大绝对相对误差: $\max err = \max\left|\frac{Y_p - Y_r}{Yr}\right|$=0.022

当模型更新时，输入测试数据集，计算得出其支持向量，然后通过曲线拟合得到离散点的运动趋势，下面是预测值的误差分析

表 4-2 预测区间支持向量列表

序号	X	Y 真实值	Y 计算值	相对误差	绝对误差
1	16.1	86.876	86.93	0.054	6.21E-04
2	16.3	82.638	82.776	0.138	1.66E-03
3	16.4	84.1	83.766	-0.334	-3.970E-03
4	16.5	76.75	77.05	0.3	3.9E-03
5	16.6	77.16	76.9	-0.26	-3.36E-03
6	17.1	56.52	56.82	0.3	5.3E-03
7	17.2	55.426	55.426	0	0.00E+00
8	17.6	38.256	38.256	0	0.00E+00
9	17.7	37.14	37.14	0	0.00E+00

我们从图形和误差分析的情况可以得出，模型在做预测时的效果和误差结果都可以达到我们的要求。

(1) 平均相对误差: $merr = \frac{1}{N}\sum_{t}\frac{Y_p - Y_r}{Y_r}$=0.022

(2) 平均绝对相对误差: $merr = \frac{1}{N}\sum_{t}\frac{|Y_p - Y_r|}{Y_r}$=0.002

(3) 最大绝对相对误差: $\max err = \max\left|\frac{Y_p - Y_r}{Yr}\right|$=0.005

从误差分析上面可以看出，各项预测误差指标值不大，预测的效果在可以接受的范围内。

4.3 实验 2 结果和分析

从上面的试验中，我们可以得出结论当单位时间的交通流量的数值比较大时，其预

测的结果一般较好。但是在单位时间交通流量不是特别大，但是因为其地理位置情况比较特殊的道路，对其进行预测是有意义的。这个时候模型的预测精度能否达到高密度交通流预测的效果，需要进行仿真实验来验证。

4.3.1 结果分析

实验的数据同样采用 carlos Doganzo 教授个人网站上面(http://www.ce.berkeley.edu/~daganzo/index.htm)上面的数据，本实验使用 San Pablo Dam 高速公路五号出口的数据，由于五号出口不是交通干线，但是是一个比较重要的支线，对它进行预测，也具有很重要的现实意义。

对于实时采集的数据，首先进行无量纲的 0 转化为 17.8 为止(X 轴代表的是时间，Y 轴代表的是车流量)然后将将时间对应的归一化，先把开始时间 14:02:00 设置为 0.1,然后将 14:04:00 设置为 0.2，直到最后将 9:58:0 交通流数据代入模型进行计算。

其效果如下图所示：

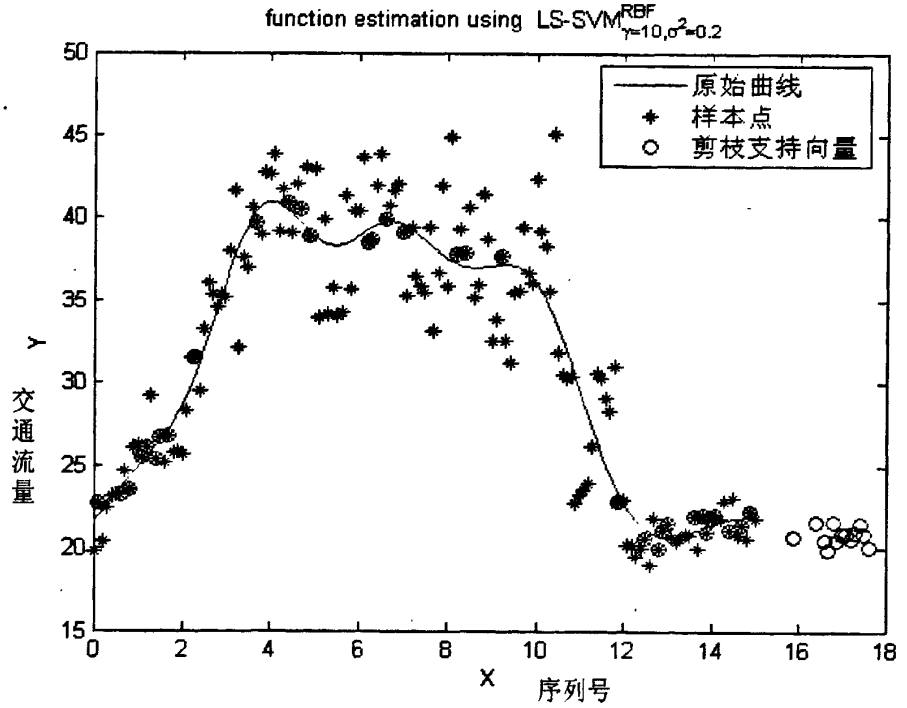

图 4-7 找出整个区间的支持向量

先将训练样本代入模型当中，通过计算得出支持向量，并通过“剪枝”得到我们拟合曲线所需要的支持向量，并且通过支持向量拟合出曲线

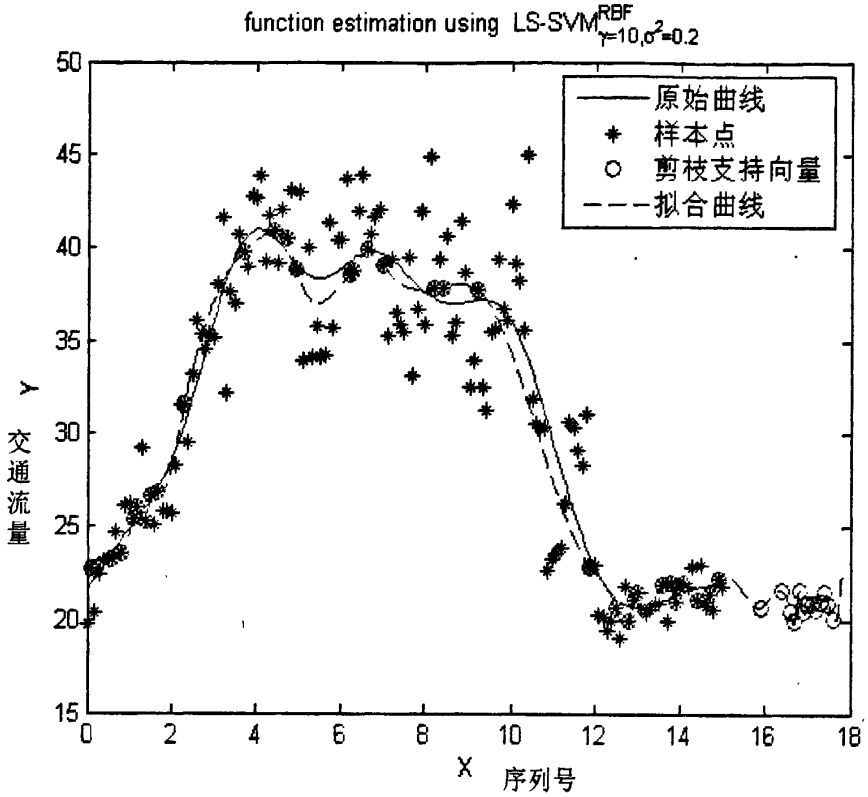

图 4-8 通过测试样本中选出的支持向量拟合出预测曲线

然后我们通过剪枝后的支持向量拟合出趋势曲线，达到得出预测效果的目的。

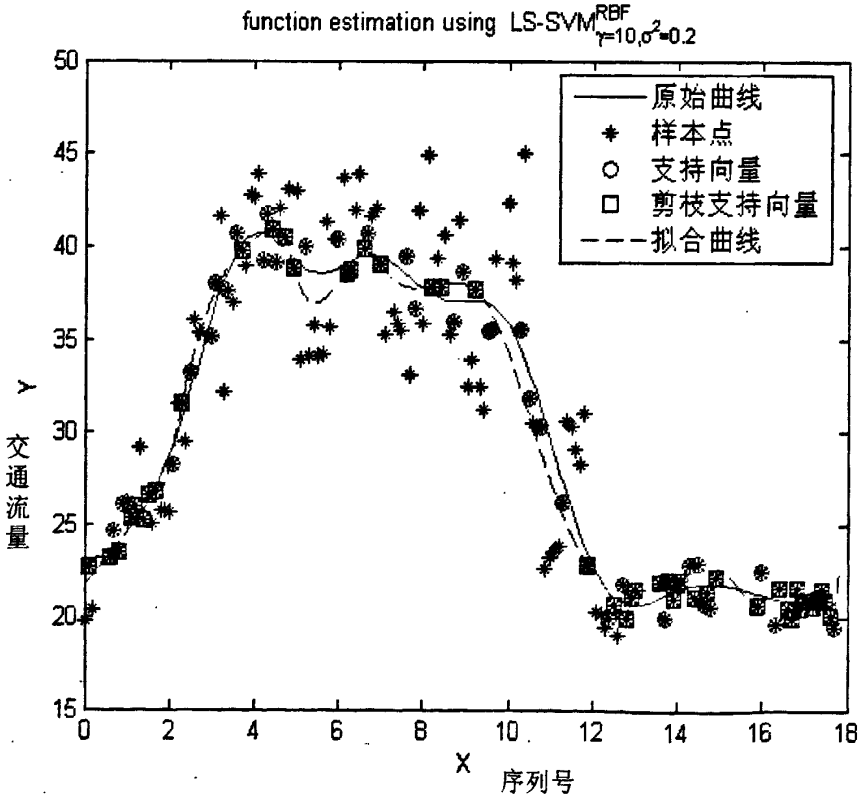

图 4-9 完成整个在线"剪枝"预测过程

由第三章的理论证明，这种算法最后得出的结果是可以接受的，但是当数据量相对较大的，其计算的时间开销也会相应的上升。时效比的效果就不会向相对较小数据集是那么明显。

4.3.2 误差分析和总结

对于误差分析和处理，我们也是用同样的标准和公式

相对误差公式：$Y_{计算值} - Y_{真实值}$

绝对误差公式：$\frac{Y_{计算值} - Y_{真实值}}{Y_{真实值}}$

训练区间 37 个支持向量列表及误差见附录 表 4-3。

我们从以上数据表中可以看出当交通流的流量不太大，并且数据比较分散的时候，模型的预测误差比较大，要高于交通比较稠密的时候

这里先进行训练区间内的数据预测和误差计算分析

(1) 平均相对误差: $merr = \frac{1}{N}\sum_{t}\frac{Y_p - Y_r}{Y_r} = 0.174$

(2) 平均绝对相对误差 $merr = \frac{1}{N}\sum_{t}\frac{|Y_p - Y_r|}{Y_r} = 0.011$

(3) 最大绝对相对误差: $\max err = \max\left|\frac{Y_p - Y_r}{Yr}\right| = 0.058$

当模型更新时，输入测试数据集，计算得出其支持向量，然后通过曲线拟合得到离散点的运动趋势，下面是预测值的误差分析

表 4-4 实验 2 预测区间支持向量列表

序号	X	Y 真实值	Y 计算值	相对误差	绝对误差
1	16.8	21.616	22.34	0.594	2.74E-02
2	16.9	20.5	20.12	-0.38	-1.81E-02
3	17	20.892	21.25	0.358	1.71E-02
4	17.1	20.916	19.86	-1.056	-5.04E-02
5	17.2	20.59	21.71	1.12	5.4E-02
6	17.3	21.042	20.83	-0.212	-1E-02
7	17.4	21.476	19.86	-1.616	-7.52E-02
8	17.5	20.916	22.87	1.954	9.34E-02
9	17.6	20.126	20.59	0.464	2.3E-02

我们从图形和误差分析的情况可以得出，模型在做预测时的效果和误差结果都可以达到我们的要求。

(1) 平均相对误差: $merr = \frac{1}{N}\sum_{t}\frac{Y_p - Y_r}{Y_r}$=0.136

(2) 平均绝对相对误差 $merr = \frac{1}{N}\sum_{t}\frac{|Y_p - Y_r|}{Y_r}$=0.007

(3) 最大绝对相对误差: $\max err = \max\left|\frac{Y_p - Y_r}{Yr}\right|$=0.09

从图形和误差分析中，我们可以看到，当交通流的流量比较稀疏并且数据点也较为分散，预测模型会出现比较大的误差，预测的效果不是太令人满意，所以模型还存在需要改进和进一步的研究。

4.4 本章小结

本章的出发点是通过一个实际的交通流的例子来验证在前面的所提出的最小二乘支持向量机的理论，在进行交通流预测的时候能够在保证精度的时候，能提高性能效率。

我们使用了加州伯克利大学，交通工程系 carlos Doganzo 教授个人网站上面(http://www.ce.berkeley.edu/~daganzo/index.htm)，1998 年美国 San Pablo Dam 高速公路，下午两点到晚上 8 点之间的交通流量的数据值，通过 Matlab7.0 软件进行仿真实验验证了模型的真实性和模型的准确性。一方面实验表明了 LSSVM 模型是可以用于交通流预测，而且向前预测的效果反而要好于补漏预测的效果(从示意图上就可以看出，在训练区外的支持向量比较密集)。同时别一方面 LSSVM 模型相比其他模型来说精度略有提高，但是算法本身有局限性，不能再较大的数据上面做预测，这也是它的局限性。

总结和展望

本文在对现有的交通流模型的总结基础及交通流特性研究的分析上面，针对支持向量机模型在预测精度上要优于其他类大多型的模型，同时其模型建立的复杂度相对于现在的集成预测模型来说要相对简单，计算规模也不是很大。通过使用最小二乘法原理改进支持向量机模型，进一步简化计算方法提高效率。通过对短时交通流预测进行应用，验证了最小二乘支持向量机模型的优越性和可靠性。主要工作如下:

(1). 总结了短时交通流的特性，先分析各种时间序列模型，评价了各个模型的性能和缺点。提出了使用支持向量机的理论运用到短期交通流预测模型中，并总结了支持向量机在交通流预测中的优缺点，针对支持向量机最后还要求解一个凸二次规划问题，其算法有一定的复杂性，需要反复的迭代，故将最小二乘理论引入支持向量机中在保证精度的情况下使算法的计算复杂度大大的降低。

(2). 当我们在支持向量机模型中使用最小二乘方法的时候，其实就是将约束条件由不等式改写为等式，将凸二次规划问题转化为线性方程组的求解问题，也就是最后求解一个线性方程组，就可以通过模型来进行预测。但是新的问题也出现了，由于解的是个线性方程组，其系数矩阵丧失了稀疏性，解法依然比较繁杂，效率较低。通过研究发现支持向量中有一部分对最后的结果影响较小，故“剪枝”这些支持向量最后不影响最终的结果。剪枝算法就产生了。

(3).“剪枝”算法固然可以提高算法的效率，并且降低计算的复杂度。但是它的缺点是不能对矩阵进行更新，但是短时交通流预测是一个动态的数据流，需要动态更新数据。为了满足预测的需要，本文最后使用类似“滑动窗口”的概念，提出了在线稀疏解的新算法，在系数矩阵中通过一次增加或者删除一个满足条件的支持向量，并且反复的计算稀疏矩阵，最后得到整个预测序列的支持向量。使用这种算法就可以在保证精度的情况下，以相对较小的时间开销，完成在线的短时交通流预测

最小二乘支持向量机理论应用在短时交通流预测中确实有其他模型不具备的优点，对短时交通流预测而言是一种较新的尝试。但是最小二乘支持向量机来源于支持向量机理论，它也有支持向量机方法的固有缺点。

(1). 对于核函数的参数而言，本文是通过反复的实验得到的，并没有通过理论推导最优解，在以后的研究和应用当中，可以使用小波，蚁群算法等求的最优解。
本文只是使用等间距的数据进行建模和预测，没有就非等距的数据进行进行处理。在以后的工作中，应该考虑非等间距的情况，使模型的预测具有更广泛性

(2). 在使用新算法做预测研究的实验过程中，出现了比较有意思的现象，当使用构造函数生成数据时会出现，在预测区间中单个数据点的误差相对较大，但是整个区间误差

和却在可接受范围内。但是在实际的数据集中间，在预测区间单个的数据误差和区间误差和都分别满足误差条件，这个现象还没有得到解释，需要进一步研究。

算法对于噪声比较大的情况，预测效果不理想，以后应该加强对抗干扰方面的研究。

致谢

首先由衷的感谢我的导师戴齐副教授，我的论文是在他精心指导和热心关怀下完成的。在我三年的交大求学生涯中老师严谨治学、勤勉踏实、忘我执着的工作态度，严谨科学的工作作风、平易近人的高尚品德，对我在以后做人、治学、工作和生活等多方面产生了很大的影响，是我学习榜样，这段难忘的经历将是我一生的财富。在此，谨向导师致以崇高的敬意和真诚的感谢。

其次，衷心感谢我的母亲，是她哺育我成长，教会我做人，引导我树立远大的志向，勉励我努力进取，我所取得的每一个进步都浸透着她的心血。

在论文的写作过程中，实验室及寝室同学以及其他专业实验室的同学给予我热心帮助和支持，那些在一起生活、学习研究的日子是我永恒的美好回忆.在此，向所有给予我鼓励和帮助的老师及同学致以崇高的敬意和由衷的感谢!

最后，感谢西南交大和信息学院对我的培养，感谢参加论文评审和答辩的各位老师、专家!

参考文献

[1] 刘运荣. 基于支持向量机的时间序列预测研究 哈尔滨工程大学 2008

[2] 李英红. 支持向量分类机的核函数研究 重庆大学 2009

[3] 段永健. 基于时间序列与支持向量机的信号识别模型及预测 山东大学 2010

[4] 任双桥. 支持向量机理论与应用 国防科技大学 博士论文 2006

[5] C.J.C Burges A tutorial on Support vector Machine for pattern recognition Data Mining and Discovery 1998

[6] T.Joachims Advances in Kernel Methods:Support vector Learning chapter Marking large Scale SVM learning Practical chap 11 1999

[7] J.S Taylor Nello Cristianini Kernel Methods for Pattern Analysis 北京 机械工业出版社

[8] 吴涛 核函数的性质，方法及其在障碍检测中的应用 国防科技大学 博士论文 2003

[9] 张小云，刘允才. 高斯核支持向量机的性能分析[C] 计算机工程 2003 29(8).23-25

[10] 张学工.关于统计学理论和支持向量机 自动化学报 2000 26(1) 32-42

[11] 姚智胜，邵春福.道路交通状态预测研究[J].哈工大学报,2009,41(4):247-249.

[12] 王进，史其信.短时交通流预测模型综述[J].中国公共安全,2005,1(1):92-98.

[13] 王大鹏.基于支持向量机的公路车流量数据分析与预测模型 哈尔滨工程大学 2006

[14] 王凡.基于支持向量机的交通流预测方法研究 大连理工大学 博士论文 2010

[15] 袁晓丽，王耀南.基于混沌化算法的支持向量机参数选取方法[J] 控制与决策 2006

[16] 朱国强，刘士荣，俞金寿.支持向量机及其在函数逼近中的应用[J].华东理工大学学报，2002，18(5):555－559，568

[17] Fan Wang,Gouzhen Tan,Chao Deng,etal.Real-time traffie flow foreeasting model and parameter seleetion based on.InProeeedingsofthe7thWorldCongresson

[18] IntelligentControlandAutomation，2008

[19] 李存军.基于集成神经网络的城市道路交通流量融合预测研究[D〕.成都:西南交通大学，2005

[20] J.A.K.Suykens,L.Lukas,J.Vandewalle. Sparse approximation using least squares support vector machines. IEEE Int Symposium on Circuits and Systems. Geneva. 2000:11757-11760

[21] J.A.K.Suykens,J.De Brabanter,L.Lukas,J.Vandewalle.Weighted least squares support vector machines:robustness and sparse approximation. Neurocomputing 2002,48:85-105

[22] Gavin C.Cawley,Nicola L.C.Talbot.Improved sparse least-squares support vector

machines.Neurocomputing 2002,48:1025-1031

[23] Gavin C.Cawley,Nicola L.C.Talbot.A Greedy Training Algorithm for Sparse

[24] Least-Squares Support Vector Machines.Berlin:Springer-Verlag,2002:681-686

[25] 王松贵,杨振海,著.广义逆矩阵及其应用.北京:北京工业大学出版社,1996:165-172

[26] Lehel C, Manfred O,Sparse on-Line Gaussian procession [J].Neural Computation 2002,14(3):641-668

[27] 邓乃扬，田英杰.数据挖掘中的新方法—支持向量机.北京:科学出版社，2004:18 一25 页

[28] 刘江华，程君实，陈佳品. 支持向量机训练算法综述［J 上海交通大学信息存储研究中心 2002:2

[29] 郑春红. 支撑矢量机应用的关键技术研究[D].西安电子科技大学 博士论文 2005

[30] 姚智胜,邵春福,熊志华等. 基于主成分分析和支持向量机的道路网短时交通流量预测[J] 吉林大学学报(工学) 2008

[31] 杨建华,于小宁. 高速公路动态交通流支持向量机预测模型[J] 西安工业大学学报 2009

[32] 李大中,韩 璞,王 臻. 基于支持向量机和粒子群算法的生物质气化过程建模与优化[J] 华北电力大学学报 2009

[33] 黎兴宝，潘丰.基于遗传混沌算法的 LSSVM 参数优化及应用[J]计算机与应用化学 2010

[34] 段永健. 基于时间序列与支持向量机的信号识别模型及预测[D]山东大学 硕士论文 2010

[35] 张敬磊,王晓原. 基于非线性组合模型的交通流预测方法[J]计算机工程 2010 五月

[36] 刘太安. 最小二乘支持向量机组合优化算法研究[J]计算机科学 2008 Vol.34 NO.7(专刊)

[37] 朱家元, 张恒喜等.最小二乘支持向量机算法研究[J]计算机科学 2003 vol.30 No.7

[38] 王海峰,胡德金.最小二乘支持向量机的一种稀疏化算法[J]计算机工程与应用 2005.33

[39] 周博韬,李安贵.最小二乘支持向量机的一种改进算法[J]南昌大学学报(理科版) 2006 12

[40] 吴宗亮,窦 衡.一种新的最小二乘支持向量机稀疏化算法[J]计算机应用 2009 6

[41] 赵会,黄景涛.一种稀疏最小二乘支持向量机[J]计算机工程与应用 2009 45(26)

[42] 余艳芳,高大启.一种改进的最小二乘支持向量机及其应用[J]计算机工程与科学 Vol.28,No.2,2006

[43] 宋海鹰,桂卫华,阳春华.稀疏最小二乘支持向量机及其应用[J]信息与控制 2008

Vol.37,No.3

[44] 刘新旺,殷建平,张国敏,罗 棻,詹宇斌. 基于最小二乘支持向量机的特征增量学习算法[J]计算机工程与科学 2008 Vol.30,No12

[45] 郭得令.基于 LSSVM 的围岩位移非线性预测应用研究[D] 武汉理工大学 硕士论文 2006

[46] 周欣然,滕召胜,易钊.构造稀疏最小二乘支持向量机的快速剪枝算法[J]电机与控制学报 2009 Vol.13 No4

[47] 宁伟.非线性最小二乘测量平差与空间数据误差分析[D]山东大学 博士论文 2006

[48] Syed Nas,Liu H, Sung K from incremental learing to model independent instance seletion-a support vector machine approach[R].Singapore: national university of Singapore

[49] Ralaivolal Alche-Buc, F.Incremental Support Vector Machine Learning: A local approach[J].lecture Notes in Computer science Springer 2001

[50] E.Osuna, R.Frend and F.Girosi Training Support Vector Machines An application to face detection IEEE CVPR 97

[51] Marcelo Espinoza,J.A.K.Suykens,B.D.Moor.Load Forecasting Using Fixed-Size Least Squares Support Vector Machines.Springer-Verlag Berlin Heidelberg, 2005:1018-1026

[52] Michele Benzi.Preconditioning Techniques for Large Linear Systems:A SurveyJournal of Computational Physics,2002,182:418-477.

[53] L.Hoegaerts,J.A.K.Suykens,J.Vandewalle,et al.A Comparison of Pruning Algorithms for Sparse Least Squares Support Vector Machines.Berlin: Springer-Verlag, 2004:1247-1253

[54] GautmaaT，MandieDP，VanHulleMM. ADifferential Entropy Based Method for Determining the OPtimal Embedding Parameters of a Signal Proc of the Int Co

[55] nf onAeousties,Speeehand Signal Proeessing. HongKong，2003，6:29 一 32

[56] 王进,史其信 短时交通流预测模型综述[J] 智能交通 2005 年 6 月号第一卷

[57] 郭敏,肖翔,蓝金辉 道路交通流短时预测方法综述[J] 控制理论与应用 2009 年第 28 卷第 6 期

[58] 高慧,赵建玉,贾磊. 短时交通流预测方法综述[J] 济南大学学报(自然科学版) Vol.22 No.1

[59] 刘静,关伟. 交通流预测方法综述[J] 公路交通科技 2004 年 3 月 Vol.21 No.3

[60] 蔡岩. 基于灰色预测模型的短期交通流预测研究[D] 成都西南交通大学 2009

[61] Liu B, Su H Y. Predictive control algorithm based on least squares support vector machines[J]. Control and Decision 2004,19(12):1439-1402

[62] Long C E, Polisetty P K, Gatzke E P. Nonliner model predicitive control using deterministic global optimization[J]. Journal of Process Control 2006,16(6)635-643

[63] Lehel C, Manfred O, Sparse on-line Gaussian process[J].Neural Computation 2002,14(3):641-668

[64] Wang H, Pi D Y, Sun Y X. Weighted online SVM regression algorithm and its application[C].International conference on advances in natural computation 2005

[65] 刘丽娜 城市道路交通流量短时预测的研究 [D] 北京邮电大学 硕士论文 2010

附录

表 3-1

序号	X	Y 真实值	预测值	误差	相对误差
1	-3	0.07731	0.0728	-0.0045	-0.0618
2	-2.8	0.11762	0.1254	0.0078	0.062
3	-2.6	0.12962	0.1193	-0.0069	-0.057
4	-2.2	0.0165	0.0232	0.0067	0. 406
5	-2	-0.0527	-0.0673	-0.0146	0.216
6	-1.8	-0.0998	-0.0923	-0.0066	-0.071
7	-1.2	-0.0747	-0.0781	-0.0034	0.0435
8	-0.6	-0.5455	-0.623	0.078	-0.142
9	0.2	0.8844	0.8832	-0.001	-0.001
10	0.8	0.2359	0.2371	0.0012	0.005
11	1	0.0419	0.0512	0.0093	0.1816
12	2.2	0.1213	0.1267	0.0054	0.0445
13	2.4	0.2007	0.2412	0.0405	0.2017
14	2.6	0.1929	0.2317	0.0388	0.1674
15	2.8	0.1932	0.1938	0.0006	0.003
16	3	0.1231	0.1245	0.0014	0.0112

表 4-1

序号	X	Y 真实值	Y 计算值	相对误差	绝对误差
1	0.2	49.958	50.12	0.162	3.14E-03
2	0.3	50.798	50.88	0.082	1.61E-03
3	0.5	53.41	53.72	0.31	5.77E-03
4	0.8	61.588	61.3	-0.288	-4.69E-03
5	1.5	72.954	72.82	-0.134	-1.84E-03
6	2.1	84.146	84.33	0.181	2.14E-03
7	2.2	87.262	87.57	0.308	3.51E-03
8	2.4	94.186	94.39	0.204	2.16E-03
9	2.6	103.71	103.41	-0.3	-2.90E-03
10	2.8	109.77	109.17	-0.6	-5.49E-03
11	3.4	126.19	126.49	0.3	2.31E-03
12	3.8	141.34	141.34	0	0.00E+00
13	3.9	139.8	139.8	0	0.00E+00
14	4.6	149.51	150.25	0.74	4.92E-03
15	5.6	150.26	151.31	0.05	3.30E-04
16	6.1	147.43	149.87	2.44	1.60E-02
17	6.4	149.59	149.59	0	0.00E+00
18	6.6	149.51	149.51	0	0.00E+00
19	7.2	150.26	150.9	0.64	4.24E-03
20	7.8	151.09	149.89	-1.2	-8.00E-03
21	8	147.15	147.15	0	0
22	8.4	142.17	145.06	2.89	0.0199
23	9.2	135.18	136.12	0.94	6.90E-03
24	10	134.64	133.82	-0.82	-6.12E-03
25	10.3	131.89	129.11	-2.78	-2.15E-02
26	11.6	89.912	89.91	-0.002	-2.22E-05
27	11.7	84.87	84.97	0.1	1.17E-03
28	12.5	51.654	51.654	0	0.00E+00
29	12.6	49.676	49.676	0	0.00E+00
30	12.7	49.742	49.742	0	0.00E+00
31	13	48.084	48.084	0	0.00E+00
32	13.9	52.482	53.248	0.766	1.43E-02

33	14.4	66.058	66.058	0	0.00E+00
34	14.5	71.154	69.804	-1.35	-1.93E-02
35	14.6	73.298	73.125	-0.017	-2.36E-03
36	15.8	86.536	87.127	0.591	6.78E-03
37	15.9	86.71	87.305	0.595	6.81E-03

表 4-3

序号	X	Y 真实值	Y 计算值	相对误差	绝对误差
1	0.1	22.12	22.78	0.66	2.90E-02
2	0.6	23.216	23.5	0.284	1.20E-02
3	0.8	23.532	24.15	0.618	2.55E-02
4	1.1	25.22	25.46	0.24	9.51E-03
5	1.2	26.034	25.32	0.714	2.80E-02
6	1.4	25.342	25.83	0.488	1.86E-02
7	1.5	26.692	26.47	-0.222	8.38E-03
8	1.7	26.832	26.92	0.088	3.26E-03
9	2.3	31.534	31.78	0.246	7.74E-03
10	3.7	39.77	39.7	-0.07	-1.76E-03
11	4.4	40.916	40.2	-0.716	1.78E-02
12	4.7	39.52	40.524	1.004	2.54E-02
13	4.9	38.91	39.27	0.36	9.16E-03
14	6.2	38.506	38.96	0.454	1.16E-02
15	6.3	38.722	39.26	0.538	1.37E-02
16	6.6	39.932	39.78	-0.152	-3.82E-03
17	7	39.106	39.34	0.234	5.94E-03
18	8.2	37.79	37.65	-0.14	-3.71E-03
19	8.4	37.878	37.59	-0.288	-7.66E-03
20	12.5	20.008	21.24	1.232	5.88E-03
21	12.8	21.112	21.18	0.068	3.21E-03
22	12.9	21.516	20.97	-0.546	-2.60E-03
23	13	20.13	21.892	1.762	8.75E-02
24	13.6	21.19	22.02	0.83	3.93E-02
25	13.8	20.998	21.27	0.272	1.27E-02
26	13.9	21.87	21.77	-0.1	-4.60E-03

27	14	21.996	21.79	-0.206	-9.45E-03
28	14.1	21.370	21.10	-0.268	-1.25E-02
29	14.4	21.374	21.96	0.586	2.66E-02
30	12.5	20.008	21.24	1.232	5.88E-02
31	12.8	21.112	21.18	0.068	3.21E-03
32	14.7	21.140	22.24	1.108	5.24E-02
33	14.9	20.726	21.94	1.214	5.53E-02
34	15.9	20.880	21.62	0.746	3.57E-02
35	16.4	20.542	20.85	0.308	1.47E-02
36	16.6	21.160	19.94	-1.212	-5.72E-02
37	16.7	21.616	20.86	-0.756	-6.81E-03

攻读硕士学位期间发表的论文

[1] 刘林 分布式数据仓库元数据的安全模式探讨. 中国西部科技 2009,5