支持向量回归的教程

Alex J. Smola, GMD 1 Bernhard Sch?olkopf, GMD 2

NeuroCOLT2技术报告系列

nc2 - tr - 1998 - 030

19983年10月

作为ESPRIT工作组的一部分制作

在神经与计算学习II中，

NeuroCOLT2 27150

欲了解更多信息，请访问NeuroCOLT网站

http://www.neurocolt.com

或电子邮件neurocolt@neurocolt.com

1smola@first.gmd.de GMD FIRST, Rudower Chaussee 5, 12489，德国柏林

23bs@first.gmd.de收到30-OCT-98GMD FIRST, Rudower Chaussee 5,12489，德国柏林

介绍1

摘要

在本教程中，我们将概述用于回归和函数估计的支持向量机(SV)的基本思想。此外，我们还总结了目前用于训练SV机器的算法，包括二次(或凸)编程部分和处理大数据集的先进方法。最后，我们提到了一些对标准SV算法的改进和扩展，并从SV的角度讨论了正则化和容量控制方面的问题。

1介绍

本文的目的是双重的。它应该作为一个独立的介绍支持向量回归为读者这一快速发展的研究领域的新。另一方面，它试图给出一个概述近年来的发展?

为此，我们决定将文章组织如下。我们首先在第1、2和3节中简要概述基本技术，并在第4节中简要总结一些图形和图表。第5节回顾了目前用于实际实现SV机器的算法技术。这可能是医生最感兴趣的。以下部分覆盖更高级的主题,比如扩展的基本SV算法(秒。6),SV机器和正规化理论之间的联系(秒。7),和方法进行模型选择和容量控制(秒。8)。我们的结论与开放式问题和问题的讨论和当前SV研究的方向。在这篇综述论文中提出的大多数结果已经在其他地方发表过，但是全面的介绍和一些细节是新的。

1.1历史背景

SV算法是60年代俄罗斯开发的广义肖像算法的非线性推广1 ?Vapnik和Lerner, 1963, Vapnik和Chervonenkis, 1964]。因此，它是基于统计学习理论或VC理论的框架，该理论已经发展了超过三十年的Vapnik, Chervonenkis和其他人Vapnik和Chervonenkis ?1974]， Vapnik ?1982, 1995]。简而言之，VC理论描述了学习机器的特性，使它们能够很好地概括到不可见的数据。

在其目前的形式，SV机是由Vapnik和他的同事在AT&T贝尔实验室开发的?Boser等人，1992年，Guyon等人，1993年，Cortes和Vapnik, 1995年，Sch?olkopf等人，1995,Vapnik等人，1997]。由于这一工业背景，SV研究目前已朝着真实世界的应用方向发展。最初的工作集中在OCR(光学字符识别)上。在短时间内，SV classi?在OCR和物体识别任务方面，ers成为了最具竞争力的系统?olkopf 1A类似的方法，但使用线性规划而不是二次规划，在美国同时采取，主要是由Mangasarian ?1964, 1969]。



介绍2

等人，1996年，Sch?Olkopf等人，1998]。一个关于SV classi的综合教程?ers已由Burges ?1998出版]。但在回归和时间序列预测应用中，也很快获得了优异的性能?ul et al .,

1997, Drucker等人，1997,Stitson等人，1999,Mattera和Haykin, 1999]。最近，在一年一度的神经信息处理系统会议上拍摄了一幅SV学习的最新状态快照。Olkopf等人，1999]。SV学习现在已经发展成为一个活跃的研究领域。而且，是不是在进入机器学习的标准方法工具箱的过程中?

1998, Cherkassky和Mulier, 1998，例如]。

1.2基本思路

假设我们有训练数据f(x1 y1):::(x ' y ')g x ?R，其中X表示输入模式的空间|，例如，Rd。这些可能是，例如，一些货币在随后几天的汇率，以及相应的计量指标。在“-SV regression Vapnik ?1995”中，我们的目标是找到一个函数f(x)，对于所有的训练数据，它与实际获得的目标yi的偏差最大，同时，它是尽可能大的。换句话说，我们不关心误差，只要它们小于”，但将不接受任何偏差大于这个。例如，如果你想确保在处理汇率时损失的不超过“钱”，这一点可能很重要。

出于教学的原因，我们首先描述线性函数f的情况，如下所示

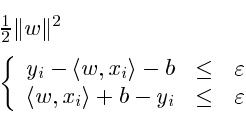


(1）

h在哪里?i表示x中的点积。在(1)的情况下，平整度意味着寻找小w。确保这一点的一种方法是最小化欧几里得范数2，即kwk2。形式上，我们可以把这个问题写成一个凸优化问题，要求:

最小化

受



（2）

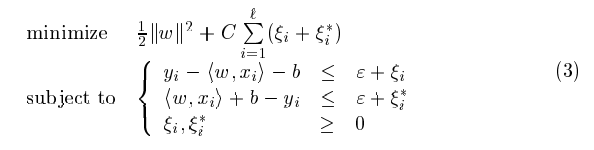
(2)中的默认假设是这样一个函数f确实存在，它以“精度”逼近所有对(xi yi)，或者换句话说，凸优化问题是可行的。然而，有时情况可能不是这样，或者我们也可能希望允许一些错误。类似于?Cortes和Vapnik, 1995]中的\软边际“损失函数，可以引入松弛变量?i ?i来处理不可行的优化约束

2参见?Smola, 1998]来概述说明这些功能的其他方法。

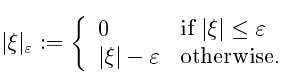


介绍3

因此，我们得到了?Vapnik, 1995]中所述的公式

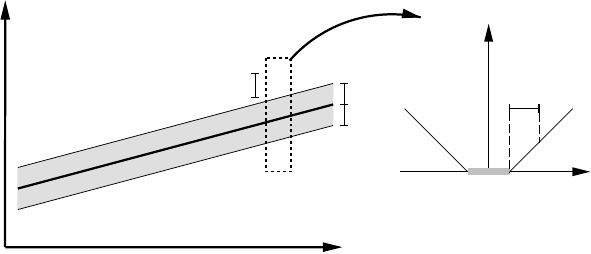


常数C > 0决定了交易?在f的?值和允许大于"的偏差的值之间。上述公式对应于处理所谓的“{不敏感损失函数j?j”所描述的



(4)

图1生动地描述了这种情况。在此范围内，只有阴影区域以外的点对成本有贡献，因为偏差以线性方式进行惩罚。结果表明，优化问题(3)可以得到更多的求解



ζ

x

xx

ζ

x

x

x

x

+ε  
0  
−ε

x

x

x

x

x

x

x

−ε

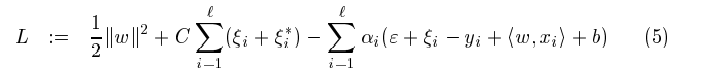
+ε



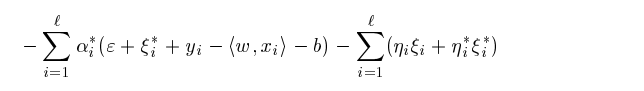
图1软裕度损失设置对应于线性SV机器。在它的双重公式中很容易。此外，正如我们将在第二节中看到的，对偶公式提供了将SV机器扩展到非线性函数的关键。因此，我们将使用使用拉格朗日乘数的标准二元化方法，如?Fletcher, 1989]所述。

1.3对偶公式和二次规划

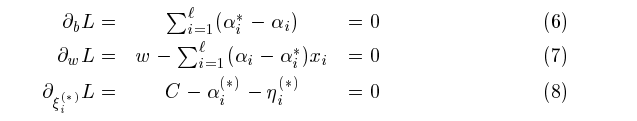
其关键思想是通过引入对偶变量集，从目标函数(在本文的其余部分将称为原始目标函数)和相应的约束构造一个拉格朗日函数。证明了该函数在最优解处对于原变量和对偶变量都有一个鞍点。详见?Goldstein, 1986, Mangasarian, 1969, McCormick, 1983, Vanderbei, 1997a]和5.2节中的解释。因此，我们进行如下工作:



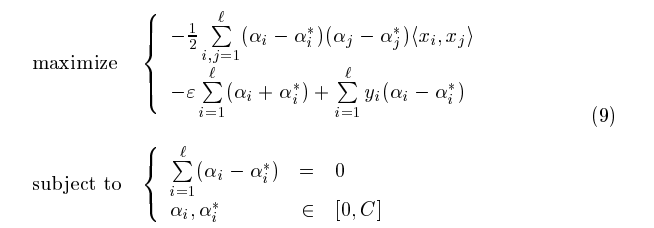
介绍4



据了解，(5)中的对偶变量必须满足正约束，即?i ?i ?i ?0.根据鞍点条件，L对原始变量(w b i i)的偏导数必须为零，以达到最优性。



将(6) (7) (8)代入(5)得到对偶优化问题



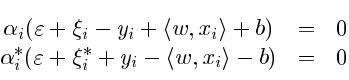
在推导(9)时，我们已经通过条件(8)消除了对偶变量i ?i，因为这些变量不再出现在对偶目标函数中，而只出现在对偶可行性条件中。式(7)可以改写为

X ' X ' w= (?我;i)xi，因此f(x) = (?我;?i)hxi xi + b:(10) i=1 i=1

这就是所谓的支持向量展开，即w完全可以描述为训练模式xi的线性组合。从某种意义上说，sv表示函数的复杂度与输入空间X的维数无关，而只取决于sv的个数。此外，完整的算法可以用数据之间的点积来描述。即使在计算f(x)时，我们也不需要显式地计算w(尽管在计算上它可能更偏向于e?Cient在线性设置)。这些观测结果将有助于建立非线性扩展的公式。

1.4计算b

到目前为止，我们忽略了计算b的问题。后者可以通过利用所谓的Karush{Kuhn{Tucker (KKT)条件来实现?Karush, 1939, Kuhn and Tucker, 1951]。这些状态表明，在最优解，对偶变量和约束条件之间的乘积必须消失。在SV的情况下，这意味着



（11）

内核5

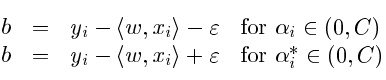
和



(C);我)?我= 0:

（12）

这使我们可以得出几个有用的结论。首先，只有对应的?(i) = C的样本(xi yi)位于f周围的“{不敏感管”之外。I = 0，也就是说，永远不可能有一组对偶变量I ? I，它们同时都是非零的，因为这需要在两个方向上都有非零松弛。最后，对于i 2 (0 C)我们得到i() = 0而且(11)中的第二个因子要消掉。因此b可以计算如下:



（13）

另一种计算b的方法将在内点优化的背景下讨论(参见第5节)。b是优化过程的副产品。因此，进一步的考虑应推迟到相应的一节。

需要注意的是SV资料片的稀疏性。从(11)可以得出，仅对于jf(xi);yij吗?“拉格朗日乘数可能是非零的，或者换句话说，对于{管内的所有样本(即图1中的阴影区域)i i消失:对于jf(xi);yij <”(11)中的第二个因子是非零的，因此i i必须为零，从而满足KKT条件。因此，我们有一个关于xi的w的稀疏展开(也就是说，我们不需要所有xi来描述w)。客户称为支持向量。

2内核

2.1预处理的非线性

下一步是使SV算法非线性。例如，这可以通过简单地用地图对训练模式xi进行预处理来实现?: X !F到某个特征空间F，如?Aizerman et al.， 1964, Nilsson, 1965]所述，然后应用标准SV回归算法。让我们简单看一下?Vapnik, 1995]中给出的一个例子。

例1 (R2中的二次特征)考虑地图?: R2 !R3和?2p 2??(x1 x2) = x1 2x1x2

(14）

在这种情况下，下标是指x2r2的分量。训练一个线性SV机器的预处理特征将产生一个二次函数。

虽然这种方法在上面的特定例子中似乎是合理的，但它可以很容易地排序，并成为更高的计算维数，不可行的多项式di?不同的特征是特征;dhigher + pp; 1 ?。这里d表示dim(X)， p表示多项式的次数。OCR的典型值

内核6

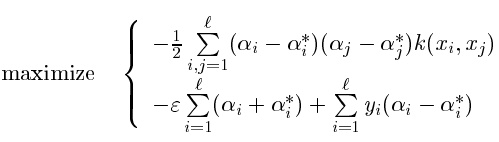
任务(表现良好)?olkopf等人，1995年，Sch?olkopf et al.， 1997, Vapnik, 1995]是p = 7 d = 28 ?28 = 784，大约对应于3:7 ?1016的特性。

2.2通过内核的隐式映射

很明显，这种方法根本不可行，我们必须用一种计算上更便宜的方法。关键的观察结果?Boser et al.， 1992]是对于例子1的feature map我们有

D ?2 p ?？2 p 2x 0 1x 02 x022

h (x) ? (x0)i = x1 2x1x2 x22 x1 = hx x0i2:(15)如前一章所述，SV算法只依赖于不同模式之间的点积。因此,苏?要知道并使用k(x x0):= h (x) (x0)i而不是(?)这使得我们可以将支持向量算法重写如下:



8

（16）

8 P '

受制于:<i=1(?我;?i) = 0

2 ? 0 C]

我?我

f(10)的展开式可以写成

X ' X ' w= (?我;?i)?(xi)因此f(x) = ?我;?i)k(xi x) + b:(17) i=1 i=1

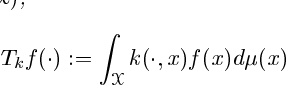
迪吗?对线性情况的推论是w不再是显式给出的。然而，由于Fischer-Riesz定理(见e.g. ?Riesz and Nagy, 1955)，它已经是唯一的de?弱意义下的内德乘以点乘hw ?(x)i。还要注意的是，在非线性设置下，优化问题对应于在特征空间而不是在输入空间中寻找证明函数。

2.3内核的条件

现在出现的问题是，哪个函数k(x x0)对应于某个特征空间f中的点积。以下定理描述了这些函数。



定理2 (Mercer ?1909])假设k 2 L1(X2)使得积分运算



(18)

内核7

是正的。让j2l2 (X)是Tk的特征函数与特征值j6 = 0相关联，并规范化使kjkl2 = 1，让j表示它的复共轭。然后

1.(j(T))j 2 ' 1。



3.x0 k (x) = P

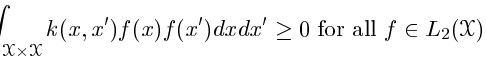


j2N

j j (x) (x0)适用于几乎所有(x x0)，其中系列

对几乎所有(x x0)都绝对一致收敛。不那么正式地说，这个定理的意思是

Z



(19）

我们可以把k(x0)写成某个特征空间中的点积。从这个条件中，我们可以得出一些关于籽粒组成的简单规则，这些规则也满足Mercer条件?Sch?Olkopf等人，1998]。在下文中，我们将这种函数称为k个允许SV核。

推论3(核的线性组合)设k1(x x0) k2(x x0)是允许的SV核，c1 c2 ?0,那么也



（20)

是一个容许核。

由于积分的线性性，这直接由(19)得到。推论4(核的积分)设s(x x0)是x ?X,那么

Z

k (x x0):=

(21）

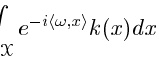


是一个允许的SV核。

这可以从(19)和(21)通过重新排列积分的顺序直接显示出来。我们现在提出一个必要的和苏?平移不变核的客户条件，即k(x x0):= k(x;x0)，由?Smola等人，1998d导出。定理5 (Smola, Scholkopf, and Muller ?1998d])一个平移不变核k(x x0) = k(x;x0)是一个可容许的SV核当且仅当傅里叶变换

Z

F ?k](!) = (2?)2



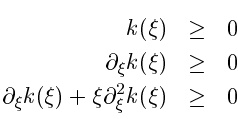
(22）

是负的。

内核8

我们将在第7节给出这个定理的证明和一些额外的解释。它基本上来源于插值理论?Micchelli, 1986]和正则化网络?Girosi等人，1993]。对于点{积类型的核，即k(x x0) = k(hx x0i)存在su?允许SV内核的客户条件。

定理6 (Burges ?1999])任何点核{积类型k(x0) = k(hx0i)必须满足



(23)

(24）

(25）

对于任何?？0以得到一个可接受的SV核。

注意，定理6中的条件只是必要的，而不是先验的。上面所述的规则对于实践人员来说是有用的工具，可以用来检查一个内核是否为可接受的SV内核，也可以用来实际构建新的内核。

2.4的例子

在?Poggio, 1975年，Sch?olkopf等人，1998]通过显式计算映射，表明齐次多项式核k与p2 N和

k (x x0) = hx x0知识产权

（26)

是合适的SV内核。从这个观察中，我们可以立即得出结论:Boser等人，1992,Vapnik, 1995]，这种类型的核

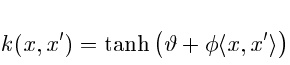
; ?



(27）

也就是说，非齐次多项式核具有p2nc> 0也是可容许的。这可以通过将k重写为齐次核的和，并应用-得到

荷兰国际集团(ing)推论3。另一个核，看起来很吸引人，是双曲正切核;



（28)

因为它会产生神经网络类型的函数。通过应用定理6，我们可以检验?Burges, 1999]当# < 0或?< 0这个内核肯定不满足Mercer的条件3

平移不变核k(x x0) = k(x;X0)是相当广泛的。在?Aizerman et al.， 1964, Micchelli, 1986, Boser et al.， 1992中显示



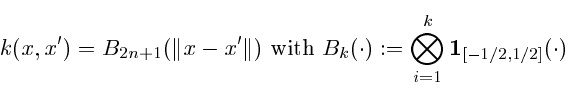
(29）

更直接的理解方式是。对于# < 0，考虑只支持KXK #=?的非负函数f(x)。在这种情况下，被积函数的支持是负的，因此积分本身也是负的。



成本函数9

是一个允许的SV核。此外，我们可以证明?Smola, 1996, Vapnik et al.， 1997]



(30）

B{2n+1阶样条，de?内德乘以单位逆函数的2n+1卷积，也是可容许的。最后，如果输入维数过低，还可以使用一些中间映射，并在第二步中应用一个内核。然而，在实际应用中，这种方法与传统内核相比几乎没有任何优势。

我们将把进一步的考虑推迟到第7节，在那里它将显示一些简单的莫迪?而SV算法可以推广到比Mercer算法条件更弱的核。

3成本函数

到目前为止，用于回归的SV算法可能看起来相当奇怪，几乎与其他现有的函数估计方法无关(例如?Huber, 1981, Stone, 1985, H?ardle, 1990, Hastie和Tibshirani, 1990, Wahba, 1990, Ripley, 1996])。然而，一旦转换为更标准的数学符号，我们将观察到与以前工作的联系。为了简单起见，我们将再次只考虑线性情况，因为使用前一章描述的核方法对非线性情况的扩展是直接的。

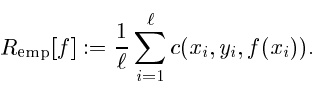
3.1风险功能

让我们稍微回顾一下第1.2节的情况。在那里，我们有一些训练数据X:= f(x1 y1):::(X ' y ')g X ?我们现在假设这个训练集是从某个概率分布P (x y)中得到的iid 4 .然而，我们的目标是找到一个函数f，使风险函数最小化(cf. ?Vapnik, 1982]) Z



（31)

(c(x y f(x))表示基于经验数据x的成本函数，决定我们将如何惩罚估计误差)。鉴于我们不知道概率度量dp(x y)，我们只能使用x来估计一个函数f，使R?f最小化]。一个可能的近似值是用经验估计代替积分，得到所谓的经验风险函数



（32)

第一个尝试是查找函数f0:= argminf2H Remp ?然而，如果H是非常丰富的，即它的容量非常高，对于

4独立且分布一致



成本函数10

实例在处理非常高维空间中的少量数据时，这可能不是一个好主意，因为它会导致over?因此泛化性能较差。因此，应该增加一个容量控制项，在SV情况下，结果是kwk2，从而导致规范化的风险功能?Tikhonov和Arsenin, 1977, Morozov, 1984, Vapnik, 1982]



（33）

> 0是一个所谓的正则化常数。许多算法，如正则化网络?Girosi等人，1993]或权值衰减网络?Bishop, 1995]最小化类似于(33).5的表达式

3.2最大似然模型和密度模型

现在问题来了，在(33)中应该使用哪个代价函数c(x y f(x))。在SV情况下的标准设置是，正如在1.2节中已经提到的，



（34)

很容易证明，用特定的损失函数(34)使(33)最小化等价于使唯一的di?因为C = 1=(')。

损失函数如jy;f(x)jp"和p > 1可能不是理想的，因为超线性增加会导致估计量的鲁棒性丧失(参见?Huber, 1981]):在这些情况下，代价函数的导数可能没有限制地增长。对于p < 1，损失函数是非凸的。

对于c(x y f(x)) = (y;f(x))2我们恢复了最小均方t方法，它不同于标准SV损失函数，导致一个矩阵反演而不是一个二次规划问题。

现在出现的问题是在(33)中应该使用哪个成本函数。一方面，我们要避免使用一个非常复杂的函数c，因为这可能导致di?狂热的优化问题。另一方面，我们应该使用最适合数据的特定成本函数。例如，我们可能会得到一个实际问题的代价函数c~，因此我们应该使用这个特定的函数。此外，在假设样本是由一个潜在的函数依赖性加上附加噪声yi = ftrue(xi) + ?i生成的，密度p(?)是最大的最优代价函数



(35）

可以这样看。估计的可能性

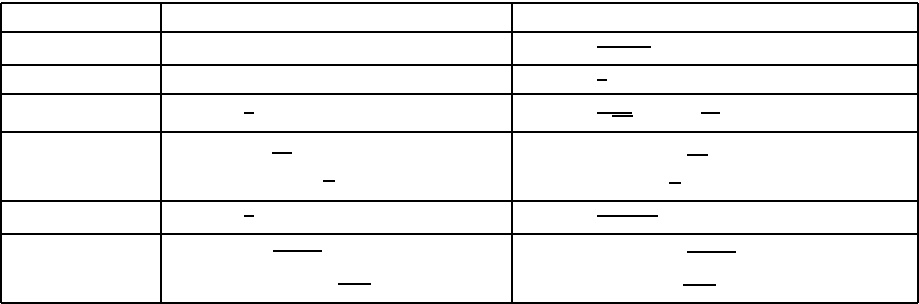


（36)

5参见?Smola, 1998]讨论了二次正则化函数的其他正则项和不变性性质。



成本函数11



密度模型

损失函数

{不敏感c(?) = j?j?P (?) = 2(1+1 ") exp(;j?)j”  
拉普拉斯方程c(?) = j?J p(?) = 21 exp(; J ?)j  
) c(?) = 12?2 p (?) = (p12? exp (; ?22   
) Huber c (?) = j  
1   
j;   
  
2 - 2   
otherwiseif j ?j ?？exp (; 2 ? ?2) 2 ?(?)吗?p (?)/ 如果j ?j ?？  
健壮的损失

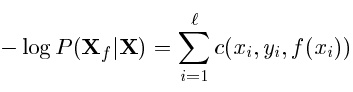
1

exp (?2, j ?j  
)否则多项式c(?) = (pj?)p(?) = (2;(1p=p) exp (; j ?摩根大通  
1 (?)p p  
c (?)= 如果j ?j ?？p (?)/ exp(;p??j ?？  
多项式 j ?;？p; p1否则 exp (?p, p 1;否则j ?)

表1在加性噪声和iid数据的假设下，常见的损耗函数和相应的密度模型为

' ' P (Xf jX) = Y P (f(xi)j(xi yi)) = Y P (f(xi)jyi);F (xi)):(37) i=1 i=1 i=1

P (Xf jX)最大化等同于最小化;logp (Xf jX)用(35)可以得到



（38)

这证明了这个说法。

然而，由这种推理得到的代价函数可能是非凸的。在这种情况下，我们必须找到一个凸代理来处理这种情况。Ciently(即:与e?客户实现相应的优化问题)。此外，回归的情况，即没有任何成本函数的知识，是不适当的de?从结构风险最小化的角度:风险只有量化才能最小化?通过成本函数(即偏差的惩罚)。最后，给定一个规格?从一个现实世界的问题，一个人应该尝试寻找一个尽可能接近这个代价函数的代理，因为它是性能wrt。这个特殊的成本函数最终会起作用。

表1概述了一些常见的密度模型和相应的损失函数，如de?(35)，而?图2包含相应函数的图。下面我们对c的唯一要求是，对于x (xi)和x (yi) f(xi)具有凸性。之所以提出这个要求，是因为我们想通过施加凸性来确保最小优化问题的存在性和唯一性(对于严格凸性)?Fletcher, 1989]。

3.3解方程

然而，为了简单起见，我们将额外假设c是对称的，并且在“”处有(最多)两个(为了对称)不连续点。0的

成本函数12

3.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  | 损失函数  密度 | |  |
| －2 | －1 | 0 | 1 | 2 |  |
|  |  |  | 损失函数  密度 | |  |

2．5

2

1．5

1

0．5

0

－3

3.

2．5

2

1．5

1

0．5

0 0

3.

3.

2．5

2

1．5

1

0．5

0

2．5

2

1．5

1

0．5

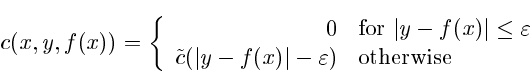
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  | 损失函数  密度 | |  |
| －2 | －1 | 0 | 1 | 2 |  |
|  |  |  | “损失函数”  密度 | |  |

－3

-3 -2 -1 0 1 2 3 -3 -2 -1 0 1 2 3

图2损失函数图及相应密度模型。左上:高斯，右上:拉普拉斯，左下:胡伯鲁棒，右下:“{不敏感

一阶导数，在区间内为零"”)。表1中的所有损失函数都属于这个类。因此c将呈现以下形式。



（39)

注意这与Vapnik的“(不敏感的损失)”相似。将这种特殊选择扩展到更一般的凸代价函数是相当简单的。对于区间内的非零成本函数?]使用另外一对松弛变量。此外，我们可以选择di?c~i, c~i和di?每个样本的“i”和“i”的不同值。在对偶公式中，以牺牲额外的拉格朗日乘子为代价，还可以考虑额外的不连续点。与(3)类似，我们得到了一个凸最小化问题?Smola等人，1998b。然而，我们将坚持使用(3)的符号，并将使用C，而不是通过和'进行规范化，因为它有助于清晰阐述。最小化1 kwk2 + C P ' (~ C (?i) + C ~(?))）

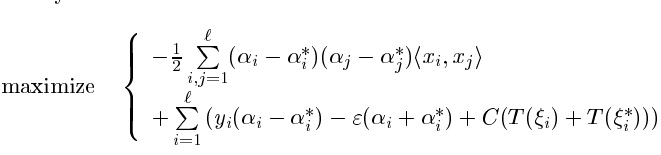
82 i=1 i服从 : Hw xii + b;咦?“+ ?i < yi;hw xii;b ?+ ?i ?i ?i ?0

(40）

成本函数13

同样，通过标准拉格朗日乘子技术，和j ?J "情况下，可以计算对偶优化问题。为了避免繁琐的表示法，我们将省略索引i和。

这个收益率



8

where<: w T (?):= =~icP=1(' ?() ?我;@ ? ?c~i ()?x)i >>< 8 iP=1 ' (?我;?i) = 0

(41）

服从> ?？C@ ? c ~(?) >:???

？0



3.4的例子

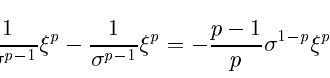
让我们考虑表1中的例子。我们将通过两个例子明确说明(41)如何进一步简化?把它变成一种实际有用的形式。在“{不区分大小写，即c~(?) = j?”j我们得到



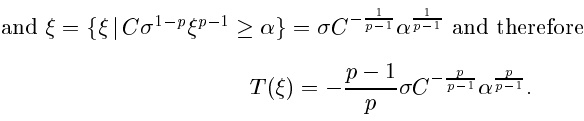
（42）

此外，我们可以从@?c~(?) = 1得出结论

？= inff ?j C ?g = 0，因此?:(43)对于分段多项式损失的情况，我们必须区分两个di?不同的情况| ?？？然后呢?> ?。在第一种情况下



(44）



（45)

在第二种情况下(?？我们?)



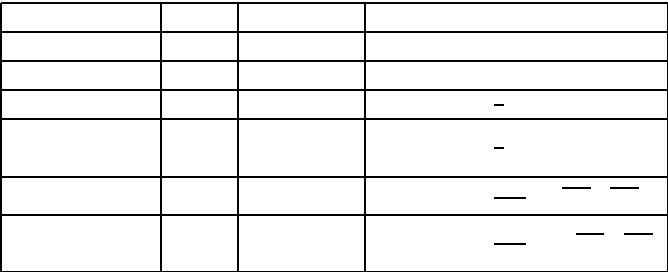
(46）

和

？= inff ?j C ?g = ?因此吗?2 ?0 c]:

(47）

更大的图景14



“{不敏感

”?

CT (?)

“6= 0 ?”2 ? c ?  
) =0拉普拉斯“=0 ?ct (?) = 0

高斯”= 0 ?2 ?01) ct (?) =;1？2  
  
Huber的稳健损失"=0 ?ct (?) =;12? c;2  
多项式”= 0 ?2 ?0 1) CT (?) =;p;p1C;p;2 ?0C] CT (?) =;p;p1?C;p;11 ?p;p1多项式

表2凸优化问题的条件取决于损失函数的选择。

这两种情况可以合并成



（48)

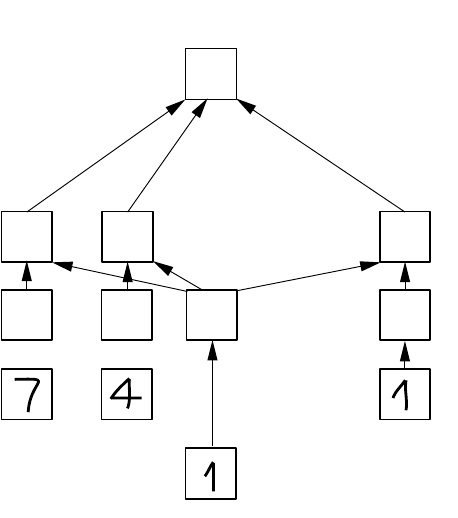
表2包含了关于?T(?)和di?不同成本函数。6注意，c~的最大斜率决定了?可行的区域，即s:= sup?2R+ @?c~(?) < 1导致?0 c]的紧凑区间。这意味着在?单一模式的影响是有界的，导致鲁棒估计?Huber, 1972]。人们也可以通过实验观察到SV机器的性能取决于signi?M?李国华，李国华等，1998 [j]。

对于成本函数的使用，除了“{不敏感的一个”之外，有必要注意一下。除非“6= 0，否则我们将失去稀疏分解的优势。”在数据很少的情况下，这可能是可以接受的，但否则将使预测步骤极其缓慢。因此，一个人将不得不在预测准确性的潜在损失与更快的预测之间进行交易。然而，请注意，这也是一个简化集算法，如?Burges, 1996a, Burges和Sch?1997年olkopf,原理图吗?Olkopf等人，1998b]可以应用于解决这一问题。

4更大的图景

在深入研究实现的算法细节之前，让我们先简单介绍一下?我们将回顾到目前为止描述的用于回归的SV算法的基本性质。图3包含了di?回归阶段的不同步骤。

输入模式(应该对其进行预测)通过映射映射到特征空间?。然后将训练模式图像在地图?下进行点积计算。这对应于在k(xi x)处求核k函数的值。最后用?i;我吗?。这，加上常数项b，产生?预测6表显示CT(?)而不是T(?)，因为前者可以直接插入相应的优化方程。



更大的图景15

输出Σ vi k (x,xi) + b

权重

(.)点积(Φ(x).Φ(xi))= k(x,xi)

映射向量Φ(x*i*),Φ(x)

支持向量x1…xn

测试向量x

Σ

*v1*

*v2*

*虚拟机*

（.)

（.)

．．．．．．

Φ(x1）

Φ(x2）

Φ(x)

Φ(x*n*）

.．.

图3 SV算法构建的回归机结构。

输出。这里描述的过程非常类似于三层神经网络中的回归，使用di?在SV情况下，输入层的权值由训练模式决定。

图4演示了SV算法如何从那些以给定精度近似原始数据的函数中选择? test函数。虽然只要求特征空间具有可操作性，但可以观察到，这些功能在输入空间中也非常适用。这是由于这样一个事实，核可以通过正则化算子与atness属性相关联。这将在第7节中更详细地解释。

最后? g。5表示在SV情况下近似质量和表示稀疏性之间的关系。逼近原始数据所需的精度越低，编码所需的sv就越少。非SV是冗余的，即即使在训练集中没有这些模式，SV机器也会构造完全相同的函数f。有人可能认为这可能是e?一种新的数据压缩方法，即只存储支持模式，从中可以完全重构估计。然而，这个简单的类比在高维数据的情况下是失败的，在存在噪声的情况下甚至更严重。在?Vapnik et al.， 1997]中可以看到，即使是中等近似质量，sv的数量也相当高，产生率比Nyquist抽样?Nyquist, 1928, Shannon, 1948]率差。

在图6中，我们可以观察到拉格朗日乘数作为力(?I ? I)在“{管”内拉动和推动回归。然而，这些力只能应用于回归接触甚至超过预定管的样品。这是KKT{条件的直接说明:要么回归在管内(因此条件满足?，因此拉格朗日乘数为0，或

优化算法16

1

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  | sinc x |  |
| -0.8 | -0.6 | -0.4 | -0.2 | 0 | 0．2 | 0．4 | 0．6 | 0．8 |  |
|  |  |  |  |  |  |  |  | Sinc x + 0.2  Sinc x - 0.2 |  |
|  |  |  |  |  |  |  | 近似 | |  |

0．8

0．6

0．4

0．2

0

-0.2

-0.4

-0.6

-0.8

－1

1

1

0．8

0．6

0．4

0．2

0

-0.2

-0.4

-0.6

-0.8

1

1

0．8

0．6

0．4

0．2

0

-0.2

-0.4

-0.6

-0.8

－1

0．8

0．6

0．4

0．2

0

-0.2

-0.4

-0.6

-0.8

1

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  | Sinc x + 0.1  Sinc x - 0.1 |  |
|  |  |  |  |  |  |  | 近似 | |  |
| -0.8 | -0.6 | -0.4 | -0.2 | 0 | 0．2 | 0．4 | 0．6 | 0．8 |  |
|  |  |  |  |  |  |  |  | Sinc x + 0.5  Sinc x - 0.5 |  |
|  |  |  |  |  |  |  | 近似 | |  |

1

1 -0.8 -0.6 -0.4 -0.2 0 0.2 0.4 0.8 1 -1 -0.8 -0.6 -0.4 -0.2 0 0.2 0.4 0.8 1

图4左上:原始函数sinc x，右上:以“= 0:1精度”近似(实线顶部和底部表示“{管，中间的虚线是回归”的大小)，左下:“= 0:2，右下:”= 0:5。

这个条件完全满足，力必须施加- i6 = 0或- i6 = 0来满足约束条件。Osuna et al.， 1997, Saunders et al.， 1998]。

5优化算法

虽然在过去的两年里已经有了大量的SV算法的实现，但我们只关注其中的几个算法，这些算法将在后面更详细地介绍。这个选择有些偏颇，因为它包含了作者最熟悉的这些算法。然而，我们认为这一概述包含了一些最e?对于那些想要自己编写SV机器代码的实践者来说，这是非常有用的。但在此之前，我们将布里?其他主要的优化包和策略。

优化算法17

IMAGE

图5左上:回归(实线)，数据点(小点)和sv(大点)的近似“= 0:5，右上角”= 0:2，左下“= 0:1，右下”= 0:02。注意sv数量的增加。

5.1实现

一个商业上可用的二次规划软件包是OSL ?IBM公司，1992]。它使用一种两阶段算法来最小化一个二次目标函数，该函数具有一个正的半?夜间二次coe吗?客户矩阵受线性约束。由于最优可能发生在可行域的内部，仅用单纯形法?Dantzig, 1962]不能解决QP问题。第一个子算法解决了一个近似LP问题，使用单纯形求解器，并在每次迭代中解决了一个相关的非常简单的QP问题。当连续逼近足够接近时，使用第二个子算法，它允许一个二次目标，并且从一个好的起始值快速收敛。另一个包,公司。Vanderbei等人，1994]用预测-校正步骤代替原-双对数势垒算法。?Lustig et al.， 1990, Mehrotra and Sun, 1992, Vanderbei, 1997b, 1994)。

另一个由斯坦福优化实验室(Stanford OptimizationLaboratory ?Murtagh and Saunders, 1983)开发的包MINOS使用了一个简化的梯度算法和一个准牛顿算法。约束由活动集策略处理。在整个过程中保持可行性。变量是classi?Ed分为基本的、超基本的和非基本的!在溶液中，有碱性和超碱性

优化算法18

1

0．8

样本插值

α

0．6

0．4

0．2

0

-0.2

-0.4

-0.6

-0.8

1

1

-0.8

-0.6

-0.4

-0.2

0

0．2

0．4

0．6

0．8

1

图6 -管(实间隔线)对近似(虚线)施加的力(虚线)。

变量远离其界限。零空间由一个矩阵张成，该矩阵由coe?矩阵的基本变量，使用稀疏因子分解。在主动约束流形上，对简化的Hessian保持了拟牛顿近似。

最后一个系统由Kaufman ?Bunch等人，1976,Bunch and Kaufman, 1977, 1980, Drucker等人，1997,Kaufmann, 1999]使用迭代自由集方法，从边界上的所有变量开始，并随着Karush Kuhn Tucker条件变得越来越违反而添加它们。这种方法的最大优点是不需要从一开始就计算完整的点积矩阵。相反，它是根据y进行评估的，与一次性解决整个优化问题相比，产生了性能改进。然而，其他算法也能穆迪吗?通过几种组块技术(见5.5节)来解决这个问题。

关于标准MATLAB优化工具箱的使用，需要谨慎注意。虽然它提供了令人满意的，虽然低于平均表现在classi?正离子任务，它似乎对回归任务(对于超过100个样本的问题)没有太大用处，因为1是e?有效地处理一个大小为2 '的优化问题，其中至少一半的Hessian特征值消失。这也是为什么不带o是有利的原因?SV回归|的上架产品

优化算法19

根据特定的SV情况调整优化策略可以实现相当大的提速。

5.2基本概念

大多数算法依赖于凸优化中的对偶理论的结果。虽然我们已经在第1.2节提到了一些基本的概念，但为了方便起见，我们将介绍brie?Y复查没有证明的核心结果。这些是特别需要的，以导出内点算法。详情和证明见?Fletcher, 1989]。

唯一性每一个严格凸约束优化问题都有一个唯一解。这意味着sv不像神经网络那样受到局部极小值问题的困扰。7唯一性性质可以这样理解:假设存在两点，即x1和x2，在这两点上得到(原目标，即目标)函数f(x)的最小值。由于问题是严格凸的，所有点x:= x1 +(1;)x2是可行的，即满足解的流形上的约束条件。而且f(x) < f(x1)+(1;)F (x2)对于2(0 1)由于是凸性的。这与初始假设f(x1) = f(x2)都是约束优化问题的最小值相矛盾。同样的推理也表明不存在局部极小值。

拉格朗日函数由原始目标函数(应该最小化的函数)减去约束条件和相应拉格朗日乘数之间的所有乘积的和(如:Goldstein, 1986, Fletcher, 1989)给出。优化可以看作是拉格朗日函数wrt的最小化。原始变量或者最大化wrt。拉格朗日乘数，即对偶变量。因此它在原始变量和对偶变量的最优解处有一个鞍点。通常拉格朗日函数只是作为推导对偶目标函数的理论工具(参见第1.2节)。

对偶目标函数它是通过最小化关于原始变量的拉格朗日函数，然后消去后者而得到的。因此，它可以单独写成二元变量(即。

拉格朗日乘数)，并导致对偶最大化问题。对偶差距对于可行的原始变量和对偶变量，原始目标

对于较大且噪声较大的问题(例如100,000个模式或更多)，几乎不可能找到优化方程的精确最小值。这是由于必须使用子集选择算法，因此不可能对训练集进行联合优化，而且全局最优值只能接近一定的精度，比如0:001。然而，神经网络还有一个额外的问题，人们甚至不能确定它是一个正在接近的全局最小值，因为有许多指数级的局部最小值?Minsky和Papert, 1969]。此外，不能说明解的绝对质量，即当前变量集到最优解的最大距离。然而，所有这些推理只在凸代价函数的情况下是有效的。



优化算法20

而不是对偶目标函数。当且仅当我们处于最优解时，等式就得到了。因此，对偶性差距是一种度量(就目标函数而言)当前变量集已接近最优解的程度。它直接来源于拉格朗日函数的鞍点性质。

卡鲁什库恩塔克(KKT)条件一组既可行又满足的原变量和对偶变量。es KKT条件是最优解决方案(即约束?对偶变量= 0)。违反KKT项的和(通过构造拉格朗日函数)确定了对偶间隙的大小。这使得计算后者非常容易。

一个简单的直觉，看看为什么\约束?dual variable = 0”可以发现，对于违逆约束，可以任意增加对偶变量，从而使拉格朗日函数任意增大。然而，这与鞍点性质相矛盾。考虑一个简单的例子:一个盒子里有一个受重力作用的球。只有在盒子的表面(即约束)，球(即变量)接触到盒子(可行区域的域)，力可以被施加(即产生非零拉格朗日乘数)。这个量是由目标函数(势能)的梯度在约束条件(盒子的面)上的投影给出的。

上面提到的结果将使我们能够找到一个e?凸优化问题的客户解。

5.3内点算法

简而言之，内点算法的思想是计算优化问题的对偶(在我们的情况下对偶的初始设置)，并同时解决这两个问题。这是通过逐步强制KKT条件迭代找到可行解，并使用原始和对偶目标函数之间的对偶差距来确定当前变量集的质量来完成的。我们将描述的算法的特殊之处是一个原始的{双路径{在?Vanderbei, 1994]中描述的一个。

5.3.1原始{双重配方

为了避免繁琐的表示法，我们将考虑稍微一般化的问题，最后将结果专门化到SV情况。可以理解的是，除非另有说明，像?表示向量，i表示其中的第i个分量。

最小化12q(?) + (c ?服从于，服从于= b

l ? ?u

(49）

优化算法21

与c ?lu2rn, a2rn ?m, b 2rm，分量保持向量之间的不等式q(?)是?的凸函数?(在可行区域，即满足约束条件的区域)。现在我们加入松弛变量除去所有的不等式除了正约束。这是:受制于A?= b吗?；G = l ?+ t = u最小化12q(?) + c ?？)

(50) g t ?0 ?免费的

(50)的沃尔夫对偶是

最大12 ?q(?);(@ ~ (?) ??+ (b ?Y) + (l ?z);(u ?S)以12 + c (?)(是的)> + s = z (51)

z ?0 y免费

并且我们得到KKT条件，即



(52）

正如我们所知，一个必要的和苏?求最优解的首要条件是原/对偶变量同时满足(50)和(51)的可行性条件和KKT条件(52)。现在我们将继续迭代地解这个方程组。

5.3.2解方程

我们将采取一种叫做路径的方法，也就是说，我们不会试图满足(52)，而是试图解决莫迪?Ed版本代替了一些> 0在第一个位置，并在迭代时减少。

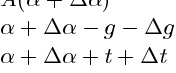
gizi =



（53)

可是它还是很稀。邪教的非线性方程组(50)，(51)，(53)的准确求解。然而，我们对精确解|不感兴趣，相反，我们的目标是找到一个更可行的解，然后减少并继续迭代。这可以通过线性化上述系统，并通过预测器{校正器方法求解得到的方程，直到对偶间隙足够小。这意味着我们将解决线性化系统的变量，一旦|，这是预测步骤|，然后将这些变量代入二次项在#，并再次解决线性化系统(校正)。其优点是，我们将得到近似相等的性能通过尝试解决二次系统直接，前提是第2项是足够小的。因此我们解了方程组

(?+ # ?)

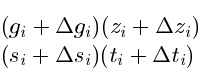


= b = l = u

q(?) + 1@2q(?)#?；((y + # y))> + s + #s = z + #z

22

(54）



＝

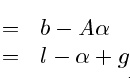
＝

优化算法22

对于#中的变量。这种方法在?Vanderbei, 1994]中作了详细的描述。我们得到了



？



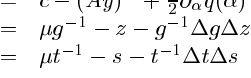
？

#？+ #t = u;?(# y)> + # z;#年代;12 @2q (?) # ?= c;(是的)> + 12@ q(?) + s;z =: ?# G + #z =

t; 1 # t + #年代

的

(55)其中g;1表示向量(1=g1::: 1=gn)， t类推。并且表示这两个向量的分量乘积所产生的向量g;1z和t;1s。解出我们得到的



z ?

=：

=：

=：

=：

=：

德?不

因此



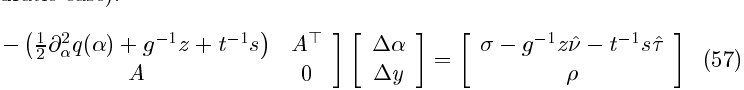
# t = s; 1 t (?年代;^:= ?；z, 1 g ?Z ?^:= ?；t s; 1 ?年代

1 # z = g; z (^ ?；#?) #s = t;；? ^):

（56)

现在我们可以制定简化的KKT{系统(见?Vanderbei, 1994)

二次案例):;



5.3.3迭代策略

对于预测器{校正器方法，我们进行如下操作。在预测器步骤中，求解系统(56)和(57)= 0,rhs上的所有#{项都设置为0，即?z = z ?s = s。在校正步骤中，再次求出?z和?s以及(56)和(57)。因为(57)的二次项不是a?在预测器校正步骤的指导下，我们只需要求二次矩阵的逆一次。这最好通过手动旋转12@2q(?) + g来实现;1 z + t;1s部分，因为它是正的。

接下来，使用此类迭代步骤获得的#中的值来更新?S t z:::。为了确保变量满足正约束，步长?使变量最多移动1;？它们到正正交线边界的初始距离。[范德贝，1994]一套?= 0:05。

另一个启发式用于计算，参数决定多少KKT{条件应该强制执行。显然，我们的目标是尽可能快地减少，但如果我们碰巧选择它太小，方程的条件将急剧恶化。一个已被证明有效的设置是



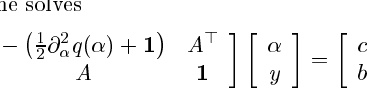
(58）

优化算法23

(58)背后的基本原理是使用KKT条件(53)的满意度的平均值作为参考点，然后迅速减少，如果我们足够远的正正交的边界，所有的变量(除y)都约束到。

最后，我们必须给出一个好的初始值。与?Van- derbei, 1994]类似，我们选择(57)的正则化版本来确定初始条件。一个解决了

；



(59）

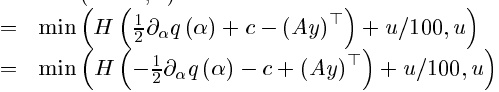
和

X = max(X u=100)

g = min (?；l u)

t = min (?u; ? ?u ) ?？z

(60）

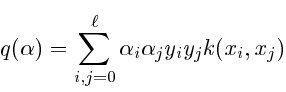


年代

其中H(:)表示Heavyside函数，即对于x > 0 H(x) = 1，否则H(x) = 0。

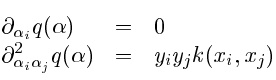
5.3.4 SV回归的特殊考虑

目前所描述的算法可以应用于SV模式识别和回归估计。模式识别的标准设置



（61）

因此



（62)

例如，Hessian是稠密的，我们唯一能做的就是计算它的cholesky因子分解来计算(57)。然而，在SV回归的情况下，我们有

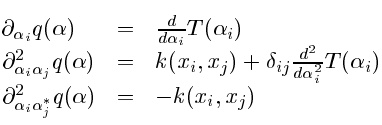


｀

q (?) =

X X (?我;我)?j;?j k(xi xj) + 2C T (?i) + T (?I) (63) I ?j = 1 i = 1

因此



(64）

@ 2ijq (?) @ 2ijq (?)因此，我们处理的是类型为M的矩阵:=K;+KD K;+ kd0其中D D0是对角矩阵。通过应用正交变换，M实际上可以通过对a求逆得到

优化算法24

”?'矩阵而不是2 ' ?2的系统。这正是直接实现优化算法而不是使用通用优化器所能获得的额外优势。可以证明，在实际应用中，Smola等人可以使用e?Ciently作为“{不敏感损失函数”的特例。

最后请注意，由于我们同时解决原始和对偶优化问题，我们也计算初始SV优化问题对应的参数。这个观察是有用的，因为它允许我们直接获得常数项b，即通过设置b = y。参见?Smola, 1998]

5.4有用的技巧

在进一步进行二次优化算法之前，让我们简单介绍一下?你是否提到一些有用的技巧，可以应用于后面描述的所有算法，并且可能有意义?斜面冲击尽管他们的简单。它们在某种程度上源自于内点方法的思想。

培训与迪吗?由于几个原因(模型选择，控制支持向量的数量，等等)，可能会发生一个人必须训练一个SV机器di?不同的正则化参数C，但其他设置完全相同。如果参数Ci不是太di?因此，利用拉格朗日乘子的重新缩放值(即i ?i)作为新的优化问题的起点是有利的。为了满足莫迪，重新调整比例是必要的吗?ed约束。因此,一个人

新= C ?Cnew ?old和类似的bnew = CCNew bold: (65) old old



假设原始目标的(主要)凸部分q(?)是部分。然而，后者的线性尺度项占主导的CCnew2 old2，这是客观的快于函数的线性(一个在实践中得到负值，但凸项只能是非负的)，重新缩放值仍然是一个更好的起点?＝ ０．在实践中，使用顺序最小化算法可以观察到大约95%的整体训练时间的提速，cf. ?Smola, 1998]。

当正则化参数相同但di?核函数的不同(但相似)的宽度参数。看到了吗?Cristianini等人，1998]在不同的背景下的细节。



在原始可行变量和对偶可行变量的情况下，以下是原始可行变量之间的联系

优化算法25

双目标函数成立:

双目标=原目标;X(gizi + siti) (66

这可以从拉格朗日函数的构造中立即看出。在回归估计中(带有“{不敏感损失函数”)有

X

2 +max(0 f(x);(y + ?)) (C;?)3 x 66;最小值(0 f(十二);(我易+吗? 2))?I I gizi + siti = 4 +min(0 (yi;?我);f (xi)) (C;?i) 5 77: (67) i

我

；马克斯(0(易;?我);f (xi)) ?我

因此，相对于最优解的收敛性可以用对偶间隙来表示。e ?有效的停止规则是需要

P g z +s t



(68）

为了一些精度?这个条件非常符合原始对偶内点路径跟随算法的精神，收敛性是根据符号的数量来衡量的。斜面(将是(68)的十进制对数)，这一约定也将在本论述的后续部分中被采用。

5.5子集选择算法

目前所描述的凸规划算法可以直接用于中等大小(多达3000个)的样本数据集，而无需进一步修改。-阳离子。然而，在大型数据集上，它是di?由于内存和CPU的限制，计算点积矩阵k(xi xj)并将其保存在内存中。一个简单的计算表明，例如存储NIST OCR数据库的点积矩阵(60000个样本)在单一精度(4字节)将消耗687兆字节，已经利用了k(xi xj) = k(xj xi)的事实。这将额外需要大致相同数量的内存和64个Ter- a?操作(分别计算乘法和加法)似乎是不现实的，至少在当前的处理器速度下。更糟糕的是，内点算法需要大约15次Cholesky迭代直到收敛，产生近1015个操作点。因此必须加上更多的e?客户方式为大数据集。

5.5.1分块

第一个解决方案是由Vapnik在1982年提出的，它依赖于只有SVs与假设的基本形式相关的观察结果。换句话说，如果我们只得到sv，我们将获得完全相同的假设，就好像我们有全部的训练集。因此，如果我们事先知道SV集，而且这也会进入内存

优化算法26

可以直接解决简化问题，从而处理signi?不能更大的数据集。问题是我们在解决问题之前不知道SV集。解决方案是从一个任意的子集开始，一个进入内存的第一个块，在它上训练SV算法，保持SVs，并将当前估计器会产生错误的数据(即位于当前回归“{管”之外的数据)块起来。然后重新培训系统，并不断迭代，直到培训后KKT{条件满足?所有的样品。

5.5.2高级工作集算法

基本的分块算法只是推迟了处理大数据集的根本问题，这些大数据集的点积矩阵不能适当地保存在内存中。因此，解决方案是?Osuna等人，1997]只使用变量的子集作为工作集，并在冻结其他变量的同时针对它们优化问题。这种方法在?Osuna et al.， 1996, Joachims, 1999, Saunders et al.， 1998]中针对模式识别的情况进行了详细的描述

我们将把这一论述适用于具有凸代价函数的回归。这一项很简单，因为只有非{二次部分会出现i (?i)+ T (?我)。为了不失一般性，我们假设“6= 0和?”2 ?0 C]，其他情况可作为特殊情况处理。首先，当所有其他变量保持不变时，我们将为工作集提取一个简化的优化问题。标记Sw f1::: ' g为工作集，Sf f1:: ' g为使用Sw Sf = f1:: ' g和Sw \ Sf = ?的?xed集。写(41)作为一个优化问题，只有在Sw的条件下><>8 >;1 P (?；?)（？；?)hxxi2 iij j i j使> + j2(S?w我;我)?彝族;P (?j;? j) hxi xji !>>:+ i2PSw j2Sf (;"（？i + ?i + C (T (?i) + T (?i)))受<8:i2PSw(?我;?i) =;我;？我)

?i 2 ?0 C

(69）

；P (?我;?i) P (?j;?j)hxi xji与等式约束;P (?我;因此，我们只需要通过与已添加的集合i2Sw j2Sf i2Sf ?i的耦合来更新线性项。很容易看到，最小化(69)也减少了(41)完全相同的数量。如果我们选择的变量KKT{条件不满足?我们保证严格降低总体目标函数，同时保持所有变量可行。此外，目标函数的下限为0。表3描述了算法。

在Osuna等人，1997，第3.3节中，这种推理被用来证明这个子集选择过程保证在数个步骤中收敛。在线性规划的背景下，为了处理大型数据集，Bradley和Mangasarian使用了类似的技术。



优化算法27

(1)初始化?我吗?我= 0

(2)任意选择工作集

(3)重复

(3.1)计算耦合项(线性和常数)

(3.2)求解简化优化问题

(3.3)从变量中选择newSw ?我吗?我不满足

(4)直到KKT工作条件ssetsw =?

表3工作集算法的基本结构。

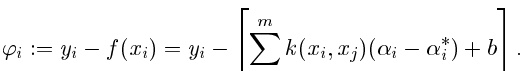
然而,这个结论是不正确的,作为一个严格单调递减序列(如n1)由某个常数有界从下面c0(例如,1)不一定会收敛于c0 |也容易可能收敛于一些c1 > c0(在本例中为0),甚至更糟的是:在一个?夜数的步骤。

然而，我们应该记住，即使没有证据证明在一个nite数的步骤收敛，在许多情况下，这种算法证明在实践中是有用的。它是为数不多的方法之一(除了?Kaufmann, 1999, Platt, 1999])，可以处理的问题，其二次部分不完全进入内存。但在实践中，必须采取特别的预防措施，以避免收敛停滞。关键部分是表3中算法的步骤(3:3)，即选择哪个工作集Sw。

5.5.3关于最优性的说明

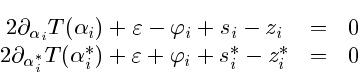
为了方便起见，KKT条件在莫迪?教育形式。表示当前假设在样本xi处所产生的误差，即

2 3



j = 1

将(51)的可行性条件改写为i ?i yield



(70）

(71）

所有的I 2 f1::: mg与子子斯斯?0.现在我们需要找到一组对偶可行变量，这是通过

= max(2@ iT (?i) + ";'i 0) si =;分钟(2子=;@ iT (?i) + ";“我0)?i ' T (?I) + " + ' I 0?如果=;最小2@ i T (?因此，KKT条件(52)可以转化为

(72）



?izi = 0 and (C;?我)如果= 0

(73）

优化算法28

可以选择所有违反(73)中某些条件的变量i ?i进行进一步优化。在大多数情况下，尤其是在优化算法的初始阶段，这组模式比任何Sw的实际大小都要大得多。不幸的是，Osuna et al.， 1997]对于如何选择Sw的信息很少。这里提出的启发式是对?Joachims, 1999]回归的一种改编。

5.5.4选择规则

与价值函数方法类似，El-Bakry et al.， 1996]，其思想是选择那些最违背(71)和(73)的变量，从而对可行性差距贡献最大。因此一个德?什么是得分变量

(74）

i:= gizi + siti



通过构造，Pi ?i是可行性间隙的大小(参见(67)对于“{不敏感损失”的情况)。通过减小这一差距，接近最优解(上界为原目标，下界为对偶目标函数)。因此，选择规则是选择那些?i最大的模式

最后，请注意像赋值sticky{?ags (cf. ?Burges, 1998])到边界处的变量，因此e?有效地解决较小的子问题，或完全从训练集中删除相应的模式，同时考虑它们的耦合?缓慢地减小必须解决的问题的大小，从而导致明显的加速。Joachims, 1999, Burges, 1996b]缓存已经计算过的点积矩阵项可能有一个意义?不能影响性能。

5.6顺序最小优化

最近提出了一种算法|顺序最小优化(SMO)|，该算法通过迭代选择大小为2的子集，并针对它们优化目标函数，将分块发挥到极致。据报道，它比经典的分块(第5.5.1节)速度快几个数量级(高达1000倍)，并表现出更好的缩放特性(通常高达一个数量级)。关键是，对于2个工作集，优化子问题可以解析地解决，而无需显式调用二次优化器。



9有些算法用?i代替



(75）

(76）



其中H(:)表示Heavyside函数，如果其参数为正则为1，否则为0。可以看到，i = 0, i0 = 0和?i00 = 0相互暗示。然而，只有?i给出了变量i对可行性差距大小的贡献的度量。

优化算法29

虽然很容易得到的模式识别由Platt ?1999]，一个简单的模仿最初的推理，以获得回归估计的扩展。这是为了方便起见，将在下面的|中做的，完整的算法被描述(包括伪代码，参看附录A)。阳离子包括模式依赖的正则化，收敛控制通过符号的数量?不能?当然，还有莫迪?对两变量优化问题进行了分析求解。

注意，这个推理只适用于SV回归，“不敏感损失函数|，对于大多数其他凸代价函数，限制二次规划问题的显式解决是不可能的。”

博览会收益如下:? rst必须推导出(莫迪? ed)边界条件约束2指数(我j)子问题回归(5.6.1节),下一个可以继续解决优化问题分析(参见5.6.2节),和?纳人检查,选择规则的哪一部分需要修改吗?Ed使该方法适用于回归(第5.6.3节)。

5.6.1模式相关正则化

考虑两个指标(i j)的约束优化问题(69)。模式相关的正则化意味着Ci可能是di?每一种模式(甚至可能是di?为方便起见，也可以得到类i?阳离子情况给出|这些是直接下降在替换对应的方程式在?Platt, 1999])。德?Ne一个辅助变量s:= yiyj为classi?阳离子(这里是yi2f1;1g)对于回归，我们必须区分四个di?不同情况下:(?我? j) (?I ?j) ?我? j) (?我? j)。这里，对于第一个和最后一个情况，设置s = 1，否则设置s =;1。由此可以从求和约束中得到



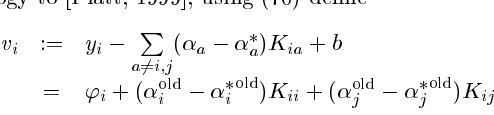
(77）

抚慰心灵?阳离子,

（？我;?i) + (?j;?j) = (?iold;?i old) + (?oldj;? jold): = ?(78)的回归。利用?(j) 2 ?0 Cj()]产量?(i) 2 ?如表4和表5所示。

5.6.2回归的解析解

接下来，我们必须解析地解决两个变量的优化问题(实际上，在回归情况下，我们必须考虑四个变量| ?i ?i ?j ?j)。在类推?Platt, 1999]，使用(70)de?不



(79）

因此

FORMULA

优化算法30

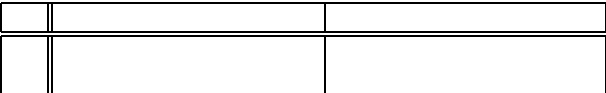
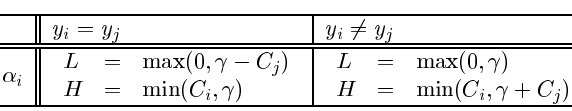


表4分类可行区域边界。



j ?



LHL

H

j ?  
= max (0 ?；Cj = max(0 ?  
) = min (Ci ?)H = min(Ci Cj + ?  
) = max (0; ?)L = max(0;?；Cj)

= min (Ci; ?+ Cj) H = min(Ci;?)

表5回归可行域边界。

现在(69)限制为(i j)可以重写如下:?i > kiii Kij ?i;我最大化> ?j;? jKji Kjj吗?j;j: + v2i (?我;?i) + vj(?j;? j);“(?I + ? I + ?j + ?j)(81)以(?我;?i) + (?j;j) = ??i ?i ?j ?j 2 ?0 C

下一个要通过求和约束来消除j j。忽略独立于?(i)项的项获得10

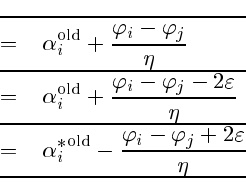
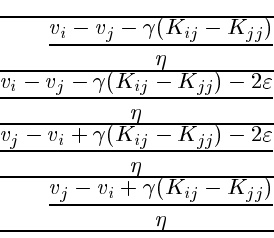
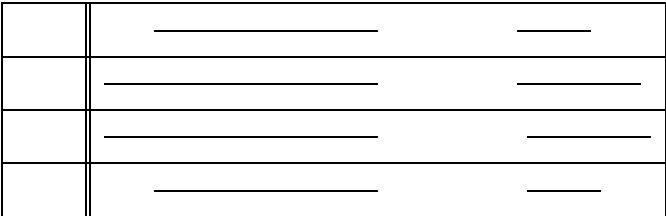
最大化;+ (12 ? ?我;我;?？I)(I)v2i (;Kii vj +;K ?jj (K;ij 2;KKij) jj。))”(?)I + ? I)(1;s)(82)根据?(i) 2 ?L() H()]:

(82)对?i或?i的无约束最大值见表6。这里的速记?:= Kii + Kjj;2 kij使用。它可能会发生，对于一对下标(ij)最初选择的象限，例如(?I ?j)是具有最优解的那个。在这种情况下，我们也必须检查其他象限。如果两个变量中的一个达到0边界|，就会发生这种情况。根据表6，需要计算带有(out)星号的变量的对应值。如果整体最优值在另一个象限，则最多重复两次。幸运的是，与梯度/误差向量的整体更新成本相比，额外的计算成本可以忽略不计，每次成功优化的更新成本为O(m)

10注意(82)只适用于?i?i = j ?j = 0。

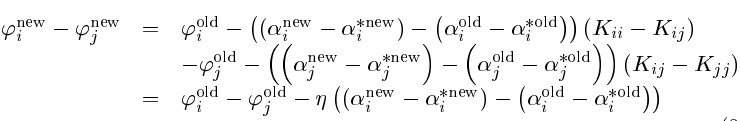


优化算法31



？  
= ?i 旧的;“我;? “j

表6二次规划问题的无约束极大值。



的一步。所有需要重新计算的是'newi;'newj，可以找到为;新 ;老? ?(冢;Kij)

（83）

最后一个等式是用(78)推导出来的。

由于数值的不稳定性，可能会发生?< 0。那样的话?应该设置为0。负的?不允许，如k(??)必须满足默瑟的条件。在那种情况下?＝ ０．i的最优值在边界H或l上，我们可以通过观察梯度，或者简单地通过计算目标函数在端点上的值来确定，取哪一个端点。

5.6.3回归选择规则

最后，必须选择指标(ij)，使目标函数最大化。再一次，SMO ?Platt, 1999，第12.2.2节]为classi?阳离子将被模仿。这意味着选择了两循环方法来最大化目标函数。外部循环遍历所有违反KKT条件的模式，首先只遍历那些既不在上边界上也不在下边界上具有拉格朗日乘数的模式，一旦它们都满足?ed，在所有违反KKT条件的模式中，以确保在整个数据集上的自一致性。11这样就解决了选择索引i的问题。

现在对于j:为了向最小值迈进一大步，我们在?i中寻找大的步骤。因为它在计算上很昂贵?对于所有可能的对(i j)，选择启发式使表6表达式中分子的绝对值最大化(即j 'I;'jj和j'i;'j 2 'j，取决于是否存在星号)。为此，我们选择了绝对值最大值对应的指数j。

11在实现子集的完全自一致性之前，迭代完整的违反KKT的数据集有时是有用的，尤其是在处理噪声数据时。否则，大量的计算资源将花费在使不是全局自一致的子集保持自一致上。这就是为什么在伪代码中，当只有不到10%的非绑定变量发生改变时，已经启动了一个全局循环。



优化算法32

如果这种启发式碰巧失败，换句话说，如果这个选择取得的进展很小，那么将按照以下方式查看所有其他索引j(在?Platt, 1999]中，这被称为\第二选择层次结构)。

1.所有与非限定例子对应的索引j都被查看，寻找一个可以继续前进的例子。

2.如果第一个启发式没有成功，所有其他的样本都会被分析，直到找到一个可以取得进展的例子。

3.如果前面的两个步骤都失败了，SMO将继续到下一个索引i。

有关这些启发式的更详细讨论，请参见?Platt, 1999]。

与内点算法不同，SMO不自动提供b的值。但是，可以像第1.4节那样，通过仔细查看得到的拉格朗日乘子?(i)来选择b。如果至少有一个变量?(i)和?(j)在边界内，则可以利用(13)。在不发生这种情况的极少数情况下，存在一个完整的允许阈值区间(比如?bibj])。因此只需取两者的平均值:b = bi+2bj。

5.6.4符号的数量?不能数字和可行性差距

通过本质上最小化一个有约束的原始优化问题，不能保证对偶目标函数随着每一步迭代而增加。然而，我们知道目标函数的最小值属于区间对偶目标

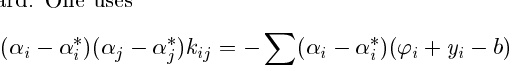
也在区间(maxj i对偶目标j)原始目标i为所有i。一个?i原始目标i]为所有迭代?因此，步骤i使用后者来确定当前解决方案的质量。

从预测误差出发计算原目标函数



我吗?j

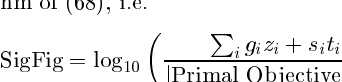
很简单。一个使用X



我

(84）

即德?Nition 'i以避免矩阵{向量乘法与点积矩阵。对偶目标函数可以通过KKT条件计算(见(66))。signi的数量?最后，斜面计算为(68)的十进制对数，即。



j + 1

（85）

加上常数1是为了避免被零除数。为了节省计算成本，原目标函数和对偶目标函数的计算只在每100步内进行。附录A包含SMO回归的伪代码。

在对实现SV机器的方法进行了相当长的讨论之后，让我们现在考虑一些SV回归中更高级的主题。

如何设计出同时减少原目标函数和对偶目标函数的子集选择优化算法仍然是一个悬而未决的问题。问题是，这通常涉及到许多与样本大小顺序相同的对偶变量，这使得这种尝试不切实际。



主题变奏曲/

6主题变奏曲

存在大量的算法莫迪?阳离子的SV算法，使其适合于规范?C设置(逆问题)，di?对于线性规划(凸组合)，有几种度量容量和约简的方法，它们分别是?控制容量和容量的方法有多种。Erent模型类(半参数建模)。

6.1的逆问题

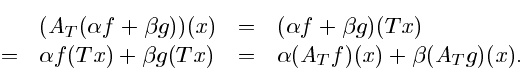
演讲的SV优化问题我们已经假设现在我们给出了直接测量,即一个训练集样本(xi yi)基于我们必须想出一个估计f,最小化风险的功能(31)。现在让我们考虑这样一种情形:我们不能观察“xi”，但可以观察其他一些相应的点“xsi”;我们也不能观察“yi”，但可以观察“ysi”。假设我们知道元素xsi是由xi通过一个(可能是非线性的)变换T生成的:

xsi = Txi:

对应的变换AT作用于f，

FORMULA

对于函数f g和coe是线性的吗?字母系数?？我们有



（86）

(87)

（88）

了解AT后，我们可以使用数据来估计底层的功能依赖关系。出于几个原因，这可能比直接估计转换数据中的依赖关系更可取。例如，在某些情况下，我们指定?是否想估计原始功能，就像在磁共振成像的情况下?Vapnik等人，1997]。

此外，我们可能有多个经过转换的数据集，但只估计一个潜在的依赖关系。这些数据集可能会消失。er的大小!此外，我们可能还需要关联di?有估算误差的不同成本?不同类型的测量，例如，如果我们相信他们的di?er在可靠性。最后，如果我们掌握了转换的知识，我们也可以利用它来改进估计。特别是当变换很复杂时，原始函数可能更容易估计。

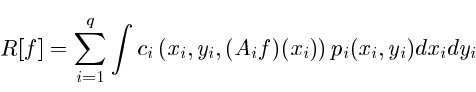
一个显著的例子是将带拖车的卡车倒车到给定位置的问题?Nguyen and Widrow, 1989, Geva et al.， 1992]。这个问题是一个复杂的类别?正离子问题(方向盘左或右)时表示在笛卡尔坐标!然而，在极坐标中，它是线性可分的。

我们不局限于只作用于f的辐的算子的情况，但对于一般的线性算子，我们将上述形式化为

主题变奏曲

遵循。我们考虑成对的观察值(x?{y?{)，采样点为x?{对应的目标值y?多重{index &{:= (i i0)表示获得目标值的过程!第二项I0 运行的观察1:::' i。在下面，它被理解为没有bar的变量对应于多个{索引的适当条目。这将帮助我们避免多个求和符号。

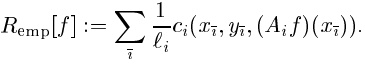
让我们假设这些样本是独立于q对应的概率分布绘制的，其密度分别为p1(x1 y1)::: pq(xq yq)。与(31)类似，我们想估计一个函数f，使风险函数?olkopf, 1998)



问Z

（89）

是最小化。注意我们可能(并且在大多数情况下会)有di?每个PDF pi的成本函数ci。与3.1节中的推理类似，我们替换R?F]由实证风险函数



X1

(90）

德?规范的风险函数(33)Rreg?然而，F]保持不变。然而,为了避免病理情况下(例如运营商Ai可能需要在输入空间二阶导数,会,当然,消失在一个简单的线性设置),我们将处理映射到特征空间在这一节中,即f (x) = hw ?我+ b (x)。类似地,我们获得以下优化问题(40)。

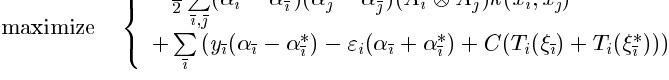
最小化1 kwk2 + C P(~ci(??{) + C ~i(?)) 82 < y?{;Ai hw ? {? (x ?{)我;b] {?“我+ ? ?{受制于:A??{i??Hw ?(x?{)i + b];y ?{?“我+ ? ?{(91)

?{ ?0

表达式Ai ?hw ?(x?{)i;b]必须读作(Aif)({)，即运算符Ai应用于特征空间的点积和常数函数b。宁(Ai ?Aj) k (?)作为一个函数，通过应用Ai到k作为第一个的函数，Ai到k作为第二个参数的函数给出

主题变奏曲

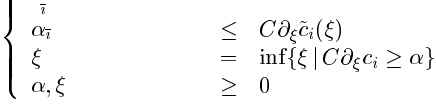
产生一个类似于(41)的优化问题。8 <; 12 P ? {| ? (? ?{;?？{) (? ?|;| ?)(人工智能吗?Aj) k (x ?{x | ?)



f (?) = P (?；?)一个k(x ?) + b其中Ti(?):= c~?{i(?) {?;@ ?{~ ci (?我)?｛

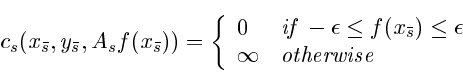
（

（92）



在下文中，我们将讨论一些使用(89)中所包含的多算子方程来整合领域知识的例子。

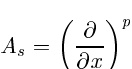
例7(估计函数的附加约束)假设我们有关于函数值在某些点xsi的附加知识，例如:?？f (xsi) ?？对于一些吗?？> 0。这可以通过将这些点添加为一个额外的集合Xs = f(xs1 0)::: (Xs 's 0)g、一个算子as:= 1和一个代价函数来实现



（93）

这些额外的硬约束导致了类似于(2)的优化问题。Olkopf, 1998a]的细节。

函数f的单调性和凸性，以及关于f的导数的其他约束条件，可以用类似的方法加以实现。在这种情况下，我们使用



(94)

而不是上面使用的As = 1。当然，这需要di?f的函数展开式的可积性。

同样地，人们可以使用这些方法来合并虚拟例子?olkopf et al.， 1996](这里T将是数据上的对称算子)，使用它处理提示?olkopf et al.， 1998]或3D断层重建?Vapnik et al.， 1997。关于这些技巧的详细描述可以在?Smola和Sch?Olkopf, 1998a]和其中的参考。

6.2凸组合和 `1{规范

到目前为止，所有的算法都涉及到凸规划，至多是二次规划。然而，有人可能会认为将问题简化为可以应用线性规划技术的情况。这是可以的

主题变奏曲

Weston等人，1999]用于SV模式识别和回归案例。关键是把(33)用



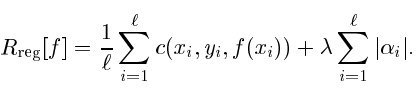
(95）

k在哪里?K1表示“1在coe?字母系数空间。因此，我们使用SV内核扩展

习f (x) = = 1 ?Ik (xi x) + b

(96）

迪?通过最小化控制容量的方法



（97）

对于“{不敏感损失函数”，这导致了一个线性规划问题。然而，在其他情况下，问题仍然是一个二次或一般凸问题，因此可能不会产生预期的计算优势。因此，我们将只限于推导线性规划问题在j?j”成本函数。申论(97)产量

最小化

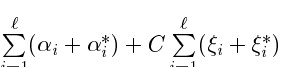
i8

= 1

我

我

i=1 ><> i;P ' (?j;？J)k(x jx i);b ?+ ?i j=1 (98) subject to: >>?摩根大通= 1”我?（？j ?;我吗??j)k(x jx i) + b;y我?+ ?i ?0



与经典SV情况不同的是，将(98)转化为它的Wolfe对偶并不能改善优化问题的结构。因此，最好直接求解(98)，这可以很容易地由线性优化器来完成，例如?Dantzig, 1962, Lustig et al.， 1990, Vanderbei, 1997a]。

在?Weston等人，1999]中，线性SV方法的一个类似变体被用来估计一条线上的密度。然而，他们提到，这可能不是一个好概念，以这种方式控制容量SV机器。然而，人们可以证明?Smola等人，1998e]，人们可以获得泛化误差的边界，它表现出比经典SV情况更好的比率(就熵数而言)?Williamson等人，1998]。

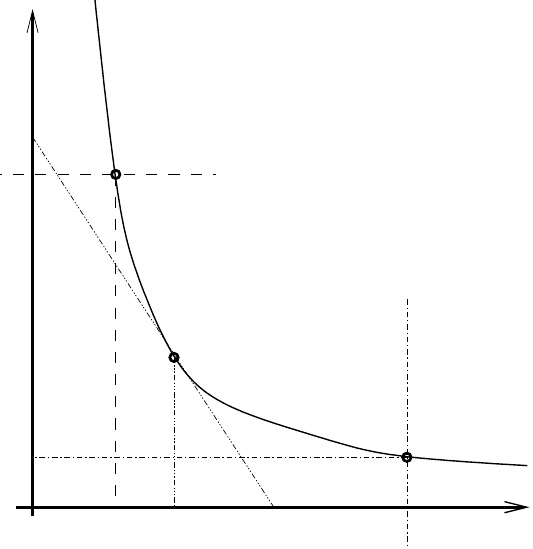
最后值得一提的是，存在类似的SV分类方法。看到了吗?Bennett, 1999年，Frie'和Harrison, 1998年。

6.3三种算法合为一图

到目前为止，我们所考虑的所有算法都产生一个值(kwk2或kk1)，一旦正则化参数C和数据都给出，就可以使用该值来应用模型选择。所以kwk2和k?k1是随机的

主旋律变奏曲

变量。然而，如果我们想应用结构风险最小化，我们应该事先指定假设类，但在这里提出的情况下，模型的层次结构是依赖于数据的。这并不是我们想要的。当然，人们可以使用适当的方法来处理这种额外的技术复杂性? shaw - taylor等人，1996a,b]但至少在这一点上，有一些方法可以避免这个问题。我们将不得不求助于这些方法。当应用模型选择标准时，原因是不同的(细节将在第8节给出)。



经验主义错误

1

2

3.

w1 w2

w3

w | | | |

图7三种风险最小化算法。

图7描述了经验误差与考虑的相应模型选择参数(这里是kwk22或k?k1)之间的依赖关系。显然，当模型选择参数指标为假设类的嵌套子集的层次结构时，函数是单调递减的。换句话说，|一组假设类被保证至少和它们的子集在一个规格上的表现一样好(就训练错误而言)?c的问题。

标准SV算法可以用图7中的情形2来描述。通过指定C或一个规格?这是什么行业?在模型复杂性和13之间有人可能会说，结构在第二种方式上也是依赖于数据的，即通过内核扩展的形式。我们只使用函数k(xi??)，因此只是函数的一个子集。然而，这并不是一个主要的限制，正如我们可以证明的那样?Kimeldorf和Wahba, 1971]在二次正则化器的情况下，特征空间F中所有展开的最优解正是SV展开中给出的那个。所以1是e?有效地搜索整个特征空间以得到最优解。因此，将w从F中的某个约束区域中选取是合理的。



主旋律变奏曲

训练错误，也就是解决

减少Remp ?F] +21 kwk22特征空间正则化k?k1凸性正则化。有效地C决定对(Remp?F] w? F])是由图上的切线条件决定的。这个观察结果可以用来选择C的值，而不需要验证集，通过确定C与VC理论中导出的风险界限一致?olkopf, 1997]。

然而，结构也可以指定?通过最小化Remp?f]

1 2 kwk2 2 ?C0特征空间正则化(100)受k?k1吗?C 0凸性正则化。对于某个正常数C0。这再次导致了一个凸的或线性的Smola等人的编程问题，并且对应于图7中的情形3。

最后，可以考虑在优化问题中允许一定程度的误差。这将导致在图7中通过求解最小化e1 2kkw?kk21 2特征凸空间正则化(101)服从Remp?f) ?C00:

尽管这些优化算法可能会失效。方程的表述方式是，在相应的条件下，它们都得到相同的假设。这也是为什么情况2是最常用的一个原因。它在实际中表现得更好?Vapnik, 1995]。

6.4不灵敏管的自动调谐

除了经典的模型选择问题，即如何指定交易对象?在经验误差和模型复杂度之间还存在着代价函数的最优选择问题。特别是对于“-不敏感的成本函数，我们仍然有选择适当的参数”的问题，以实现良好的性能与SV机器。

Smola等人，1998a]表明噪声水平和SV回归方法的最佳“-参数之间存在线性相关性。然而，这需要我们对噪声模型有所了解。虽然在某些特殊情况下可能是这样，但一般不能这样假设。因此，尽管提供了理论见解，但这一结论本身在实践中并不是特别有用。而且，如果我们真的知道噪声模型，我们很可能不会选择{不敏感代价函数，而是选择相应的最大似然损失函数。

然而，有一种方法可以构造自动调整SV的SV机器，而且，至少是渐进地，有一个预定的采样点的分数作为SV。该方法原理图?Olkopf等人，1998a]与第6.3节提出的观点有关。我们修改(33)使“变成”

主旋律变奏曲

优化问题的一个变量，在原目标函数中包含一个额外的项，它试图最小化”。换句话说

最小化R ?f]:= Remp?f) + 2 (102)对一些人来说?> 0。因此(40)就变成(又一次在'和C之间进行通常的转换)

最小化1 2kwk2 + C P ' (~ C (?)+ c ~ (?)) +’吗?”8 i=1 i i

< yi;hw xii;b ?“+ ?i(103)取决于:h?w?Xii + b;咦?

？

“+ ?i i i 0

注意，这适用于任何带有“{不敏感区”的凸损失函数。然而，为了便于说明，我们将坚持使用标准的j?j”损失函数。计算(103)产量的对偶最大化;12p ' (?我;我)?j;?j)k(xi xj) + P ' yi(?我;?我)

我吗?j = 1 i = 1



（104)

8 > P '

> >: ?iP=1 ' (??我+吗?我)?C ?' 2 ?0 c]

我

我

除了具有能够自动确定的优势”，(104)还有另一个优势。它可以用来预先指定sv的数量:定理8 (Scholkopf, Bartlett, Smola，和Williamson ?1998a)

1.？是误差分数的上界。

2.？是sv分数的下界。

3.假设数据是由分布p(x y) = p(x)p(yjx)的iid生成的，并具有连续的条件分布p(yjx)。渐近的概率是1 ?等于sv的分数和误差的分数。

因此，与标准的“-SV回归”不同，所谓的?-SV机器包含了一种直接控制sv数量的方法。有关该方法的详细信息和性能描述，请参阅原始出版物。

从本质上讲，?-SV回归在“-SV回归”基础上进行了改进，允许管宽自动适应数据。然而，到目前为止，保留下来的是管的形状。然而，我们可以更进一步

主题变奏曲/

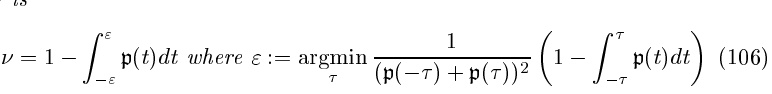
以及使用非恒定宽度的参数管模型，导致几乎相同的优化问题?Olkopf等人，1998a]。

结合?-SV回归的概念与结果的渐近最优选择?对于给定的噪声模型，Smola等人，1998a]提出了如何调整的指导方针。只要已知噪声模型的类别(如高斯或拉普拉斯)。

备注9(?的最佳选择)用p表示具有单位方差的概率密度，用p表示由p生成的一组噪声模型

P:= ??p = ?1 p;?y ? ?: ( 105)

此外，假设数据是由分布p(x y) = p(x)p(y;F (x))和p(y;F (x))是连续的，即由一个潜在的函数依赖F生成，被附加噪声损坏。那么在一致收敛的假设下，渐近最优适应参数?是



证据见?斯莫拉，1998]。对于多项式噪声模型，即exp(;j?jp)类型的密度，可以计算相应的(渐近地)?的最优值。他们得到了表7。

多项式度p 1 2 3 4 5最优?1 0.5405 0.2909 0.1898 0.1384最优?单位方差0 0.6120 1.1180 1.3583 1.4844多项式度p 6 7 8 9 10最优?最优?单位方差为1.5576 1.6035 1.6339 1.6551 1.6704

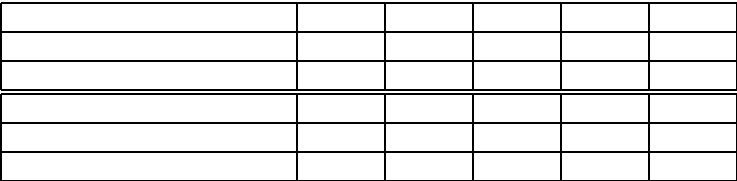


表7最佳?然后呢?对于不同程度的多项式加性噪声。

我们通过注意?-SV回归与削减估计量的思想相关来结束这一节。你可以证明回归不在?如果我们干扰了管子外面的点。因此，回归基本上是通过丢弃一定比例的异常值(speci?通过?，并从剩余的点计算回归估计?Olkopf等人，1998]。

6.5半参数SV机器

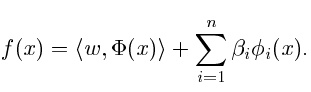
SV机器的优势之一是它们是非参数技术，无需预先指定基函数的数量。事实上，对于许多使用的核(不是多项式核)，如高斯rbf{核，它可以显示?Micchelli, 1986, Dyn, 1987] SV机器是通用逼近器。请注意，也有类似的证据

主题变奏曲

而对于神经网络呢?Hornik等人，1989]。然而，与SV机器不同的是，这在实践中并没有多大帮助，因为神经网络可能会陷入局部最小值，而SV机器有一个实际可实现的算法来精确地找到局部最小值。

虽然这在一般情况下是有利的，但参数化模型本身就是有用的技术。特别是如果一个人碰巧对这个问题有额外的了解，不利用它将是极其愚蠢的。例如，可能会出现e?数据的特征用一组基函数f?1(?)::: ?n(?)g的线性组合来描述。其次，也可能是用户希望有一个可理解的模型，而不是sacri?c的准确性。例如，生命科学领域的许多人倾向于偏好线性模型。这可能是构建半参数模型的一些动机，半参数模型既易于理解(对于参数部分)，又具有良好的性能(对于额外的非参数形式主义)。半参数模型的一个很好的参考文献是?Bickel等人，1994]。

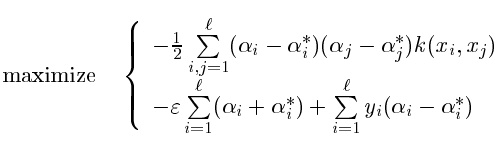
一种常用的方法是用参数模型对数据进行t，然后训练非参数部分的误差，即非参数部分对误差进行t。我们可以证明，这只在非常有限的情况下有用。在一般情况下，不可能在给定的一类中找到最好的模型。通过这样做可以得到不同的成本函数。更好的方法是解决一个凸优化问题，就像在标准SV机器，然而di?一个允许函数的集合



（107）

注意，这并不是那么多的di?如果我们设n = 1且?1(?)= 1，则与原始设置的值不同。求解相应的优化方程得到

8



（108）

受<:8 iP=1 ' (?我;?i)?j(xi) = 0对所有1 ?j ?n

我?我

2 ? 0 C]

单一等式约束(由于常数o?set)被转化为多重等式约束。强烈建议在求解(108)时使用原始的{dual方法，因为这(参见5.3节)会自动生成coe?Cients ?i作为(108)中等式约束的对偶变量。

半参数模型可用于时间序列预测，例如使用简单的线性自回归模型进行非线性增强，或也可进行假设检验，是否指定?所考虑的C参数模型为just ?艾德。请注意，类似的模型已经在光滑样条的背景下被提出和探索了?Wahba, 1990, 1999]。而且这

正规化42

背景在“有条件的肯定的de?nite内核(见?Smola等人，1998d]和第7.4节)。有关这个扩展的更多细节，请参阅原始工作?Smola等人，1998c]。

7正规化

到目前为止，我们还没有考虑到规格?映射的C属性?把它作为构造非线性回归函数的一个方便的技巧。在某些情况下，映射是由内核隐式给出的，因此映射本身及其许多属性都被忽略了。此外，我们希望对内核映射有更深入的了解，以便能够为一个规范选择合适的内核?c任务，也许通过这样做，同时结合之前的知识?Olkopf等人，1998]。最后，feature map似乎违背了维数的诅咒?Bellman, 1961]通过在一些更高维的空间中通过地图使问题看起来更容易。

在这一节中，我们将重点讨论SV方法与以前的技术(如正则化网络?Girosi等人，1993年)之间的联系。这将使我们了解内核如何工作的机制。特别地，我们将展示SV机器本质上是正则化网络(RN)，它对代价函数进行了巧妙的选择，其内核是相应正则化算子的格林函数。由于本文的性质，我们只能对这些问题做一个简要的说明。读者可以参考Smola等人，1998d]。

7.1正则化网络

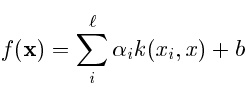
让我们布里干酪?让我们复习rn的基本概念。就像在(33)中，我们也最小化了一个规范化的风险函数。然而，我们对加强功能空间的一致性不感兴趣，而是尝试优化一些di。函数在输入空间中的光滑性有多种准则。因此我们得到



（109）

这里P表示?Tikhonov and Arsenin, 1977]意义上的正则化算子，即P是正半ide?nite算子将考虑的函数f的Hilbert空间H映射到点积空间D，从而得到表达式hPf ?Pgi很好?例如，通过选择一个适当的算子来惩罚f的大变化，可以减少众所周知的over?剪e ?等。Aronszajn, 1950, Kimeldorf and Wahba, 1971, Saitoh, 1988, Girosi, 1998]。

利用f关于对称函数k(xi xj)的展开式(注意，k，就像凸组合的情况一样，不需要满?你美世的条件),



(110）

正规化43

以及“{不敏感代价函数”，这导致了一个类似于sv的二次规划问题。通过计算沃尔夫的双重和使用



(111）

我们得到了什么?= D; 1 k (?；?)，?？是最小1 (?；) > KD; 1 k (?；?);（？；> y ?);“P”(?)I + ? I) 2 I =1

P '

受(?我;?i) = 0 ?i ?i 2 ?0 C

(112)不幸的是，这个问题的设置并没有保持在coe?cients，作为一个潜在的关于i和i的稀疏分解，被D;1K破坏了，它通常不是对角的。

7.2格林函数

比较(9)与(112)导致的问题是，如果和在什么条件下这两种方法可能是等效的，因此也在什么条件下正则化网络可能导致稀疏分解(即只有几个扩展coe?Cients ?i在f中会消失?er从0)。苏吗?cient14条件为D = K(因此KD;1K = K)，即

k(xi xj) = h(Pk)(xi:) ?(Pk)(xj:)i (self consistency):(113)我们现在的目标是解决以下两个问题:

1.给定一个正则化算子P和一个核k，使使用k的SV机器不仅在特征空间中加强?性，而且对应于最小化以P作为正则化算子的正则化风险函数。

2.给定一个SV核k，和一个正则化算子P，使得使用这个核的SV机器可以被视为使用P的正则化网络。

这两个问题可以通过使用?Girosi等人在1993年所描述的Green函数的概念来解决。这些函数已经在解di?为方程。对于我们的目的，它是su?得到P P的格林函数Gxi(x)满足



（114）

这里，?xi(x)是?{分布(不要与克罗内克符号混淆?ij)，它具有hf ?第十二? = f (xi)。核与正则化算子之间的关系在以下命题中形式化:14在K不具有满秩D的情况下，只需是K的像的逆，例如伪逆就是这样一个矩阵。



正规化44

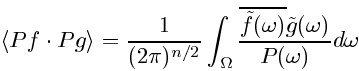
第10号提案(Smola and Scholkopf ?1998b)

设P是正则化算子，G是P P的格林函数。然后G是Mercer Kernel，使D = k SV机器使用G最小化风险函数(109)，P作为正则化算子。

下面我们将以两种方式利用这个关系:计算一个给定的正则化算子P的Green函数，并从一个给定的核k推导正则化算子。

7.3平移不变内核

让我们现在更具体些?我们可以考虑正则化算子P^，它可以在傅里叶空间中写成乘法形式



Z f ~ (!) ~ g (!)

？

（115）

其中f~(!)表示f(x)的傅里叶变换，P (!) = P(;!)实数，非负且收敛于0，对于j!j !1和(:= supp?P(!)]。较小的P(!)值对应于相应频率的强衰减。因此，小值的P(!)代表大值的P !由于f~的高频分量对应于f的快速变化。P(!)描述了P的?lter性质P |注意，对于P(!) = 0没有衰减发生，因为这些频率已被排除在积分域之外。

正则化算子de?在傅里叶空间中利用(115)，可以通过利用P (!) = P (;!) = P (!

1 Z



(2) Rn n = 2

(116）

为对应的满足平移不变性的格林函数，即



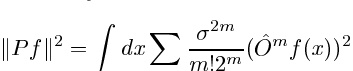
（117）

这给了我们一个e?用于分析SV内核和它们所展示的容量控制类型的cient工具。事实上，上面是Bochner定理的一个特殊情况，Bochner, 1959]，说明一个正测度的傅里叶变换构成一个正的希尔伯特施密特核。

例11(高斯核)

继?Girosi et al.， 1993]中描述的?Yuille and Grzywacz, 1988]之后，我们可以看到

Z



米

(118）

O^2m = #m, O^2m+1 = r#m， #为拉普拉斯算子，r为梯度算子，得到高斯核(图8)

K (x) = exp2x?k2 2：



（119）

正规化45

此外，我们可以提供等效的2 用其傅里叶性质表示P，即P (!) = exp(;？k2 ！k2)上升到一个乘法常数。用高斯RBF核训练SV机器?Sch?Olkopf等人，1997]cor- response to最小化规格?C代价函数，具有类型为(118)的正则化算子。还记得吗，(118)意味着f的所有导数都被惩罚(我们有一个伪?得到一个非常平滑的估计。这也解释了在这种情况下SV机器的良好性能，因为在某些高维空间中选择?at函数并不明显会对应于低维空间中的简单函数，如?Smola等人，1998d]对Dirichlet核函数所示。

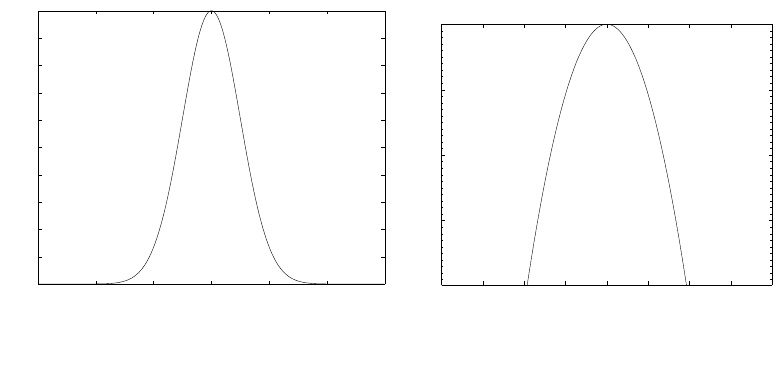


图8左:标准差为0的高斯核:  
5.右:核的傅里叶变换。

k (x)

1

0．9

0．8

0．7

0．6

0．5

0．4

0．3

0．2

0．1

0

−3

100

10−50

k(ω)

−100年10

10−150

−2

−1

0 x

1

2

3.

−20

−15

−10

−5

0 2π⋅ω

5

10

15

20.

在一般平滑性假设下，高斯核函数往往具有良好的性能，尤其在没有额外的数据知识可用的情况下，应该考虑使用高斯核函数。更多的信息和例子，例如关于Bq样条核、Dirichlet核、期刊核等，见?Smola等，1998d]。

现在出现的问题是选择哪个内核。让我们想想两个极端的情况。

1.假设我们已经知道我们想要估计的函数的功率谱Pow(!)的形状。在这种情况下，我们选择k，使k~匹配功率谱?

2.如果我们碰巧对给定的数据了解很少，那么一般的平滑假设是一个合理的选择。因此，我们可能需要选择一个高斯核函数(参见示例11)。如果计算时间很重要，那么可以考虑使用紧凑支持的内核，例如使用Bq{样条内核(见(30))。这种选择将导致许多矩阵元素kij = k(xi;xj)消失。

通常的情况是在两个极端情况之间，我们有一些有限的先验知识。有关使用先验知识选择内核的更多信息，请参见?Sch?Olkopf等人，1998]。

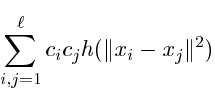
正规化46

7.4条件肯定性

与rn的连接允许我们利用一种连接，由?Madych和Nelson, 1990]所显示，?Girosi等人，1993]指出。主要的陈述是，有条件的肯定de?nite (cpd)函数生成允许的SV内核。这是非常有用的，因为是cpd的性质往往比Mercer的条件更容易验证，特别是当结合Schoenberg和Micchelli关于cpd和完全单调函数之间的联系的结果?Schoenberg, 1938a,b, Micchelli, 1986]。此外，cpd函数导致了一类不一定满足Mercer条件的SV内核。德?国家12(有条件地肯定de?夜间功能)

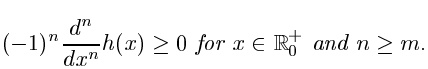
连续函数h, de?内德?0 1)，据说是有条件的正de?nite (cpd)对于任意不同的点x1::: x ' 2rn是二次形式

(120）



当标量c1::: c '满足p ' i=1 c ip(xi) = 0时，对于Rn上所有小于m次的多项式p。

德?一个函数h(x)被称为m阶的完全单调函数



（121）

可以证明?Schoenberg, 1938a,b, Micchelli, 1986]，一个函数h(x2)是有条件的正de?Nite当且仅当h(x)是完全相同阶的单调。这为检查函数是否为cpd提供了一个(有时更简单)标准。这些工具可以用来推导出一个简单的检查条件，用于测试一个内核是否为可接受的SV内核。

14号提案(Smola, Scholkopf, and Muller ?1998d)

德?ne \*nm是Rn上小于m次多项式的空间。每个m阶cpd函数h通过设k(xi xj):= h(kxi;xjk2)。换句话说，由m阶cpd函数生成的核可用于SV机器，前提是由核产生的?估计正交于次数低于m的多项式。这可以通过半参数模型实现，如6.5节所述。这就产生了可行域中的一个凸优化问题。唯一的问题是，由于点积矩阵的特征值是负的，所以问题可能不会凸到可行区域之外。因此我们需要从点积矩阵中投影出多项式部分。关于算法实现的详细信息见?Smola等人，1998d]。

正规化47

因此，人们可以使用?Girosi等人，1993]在正则化网络的背景下提出的内核作为SV内核:

高斯(m = 0) (122) k(x y) =;kx;Yk2 + c2多重二次方程(m = 1) (123) k(x y) = pkx;日元k2+ c2 逆多重二次方程(m = 0) (124) k(x y) = kx;Yk2 lnkx;Yk薄板样条(m = 2) (125)



7.5再现核希尔伯特空间

在核和正则化之间存在着一种比目前所指出的更深的联系。它与核希尔伯特空间(RKHS)再生理论有关。这些方法最初是在Aronszajn, 1944, 1950年引入的，在Parzen, 1961年被用于时间序列预测。关于RKHS理论的现代解释见? saito, 1988, Small and McLeish, 1994]。为了方便起见，我们召回de?RKHS的国家，在?Aronszajn, 1950]。德?nnition 15(再现核希尔伯特空间)设F是一个函数?内德在集合E上，形成一个希尔伯特空间(复数或实空间)。x x02e的函数k(xx0)称为F上的再现核

1.对于每一个x, k(x ?)作为其第二个参数的函数属于F。

2.对于每一个x2e和每一个f2f

F (x) = hf(?) k(x ?)i(再现性):

（126）

在其他几个性质中，如唯一性、存在性、可加性和投影性，在?Moore, 1916, Aronszajn, 1950]中证明了矩阵Kij:= k(xi xj)是一个正半核?夜间矩阵。而且，对于每一个正函数k(x x0)(即每一个满足Mercer条件的函数)恰好对应于一类具有唯一二次形式的函数，形成一个Hilbert空间，并允许k(x x0)作为其再生核。

换句话说，在RKHS和允许的SV核之间存在一一对应关系。由于RKHS元素的复制特性，我们得到了特别的



（127）

如果我们要求正则化算子P将函数映射到k给出的RKHS中，则其本质上与(113)相同。?Girosi, 1998]采用了这种方法，得到了与?Smola等人，1998d相似的结果。

RKHS的理论特别有用，因为它使我们能够以一种非常优雅的形式陈述和证明关于展开式最优性的结果。一个中心命题就是所谓的表征者定理。

正规化48

定理16 (Kimeldorf·? 1971],考克斯和奥沙利文吗? 1990)让F再生核希尔伯特空间理论是一个实值函数与复制内核k。E表示X训练集,并让+:= F # 1::: # mg m 2 N是一组函数等E矩阵j: # = # (xj)最大等级。然后

f2span (?) + h ?h2F

argmin

f0 =

Remp吗?f] + khk2F

（128）

我们有f0 2张成空间(+ fk(x1:)::: k(x ':)g)

这个定理表明，对于任何代价函数，RKHS中的最优展开式是de?用基函数de?内，即k(xi ?)。这是一种更一般的方式，表示可以用映射的输入数据表示SV优化问题的解决方案。

Williams, 1998]中指出了与高斯过程(GP)的一个有趣的联系，其中表明，只要我们使用二次损失函数，使用完整贝叶斯后验均值的最大后验近似，GP回归产生与SV机器|相同的结果。内核kP最小值(x x0)在这种情况下起着GP协方差函数的作用| i?j k(ξxj) ?我?J表示模型的可能性最大化。基本上，任何可接受的SV核也可以用于GP回归，反之亦然，这可能打开一种直接比较和/或适应di?这两种技术的模型选择方案是不同的。

7.6支持向量和稀疏性

我们以SV机器和稀疏表示之间的连接来结束本节。该报告主要基于?Girosi, 1998]。简而言之，其思想是使用?Olshausen和Field, 1996]的正则化风险函数，并使用di?erent Hilbert空间(即RKHS de?ned by k)，以获得类似于SV机器的扩展。

其出发点是使用大量冗余的基函数集，即所谓的字典来进行逼近和插值。这个概念在小波理论中被引入(参见?Meyer, 1992, Daubechies, 1992)。问题是选择一个小的原子子集，即该字典的元素，产生一个目标函数g(?)的几乎最好的近似或估计。有三种主要的方法来进行这样的分解。

第一个是匹配的追求，如?Mallat和Zhang, 1993]。这里以迭代的方式从字典中一次一个地选择最优函数，即通过添加字典中的原子来最大限度地减少逼近误差。这种方法在计算上是很困难的。由于优化步骤只在局部进行，所以在稀疏性方面是次优的。存在一个与此方法等效的SV，在?Sch?olkopf等，1998c]，该方法通过迭代选择最优原子，解决了用字典(函数k(x ?) x2 E)的原子以贪婪的方式最优重构特征空间元素(如RKHS)的问题。

正规化49

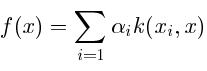
另一种方法，称为最佳基算法，在?Coifman和Wickerhauser, 1992]中描述，其中一个人手头有一个大的(通常是正交的)基字典，并试图选择最佳的基来分解估计。Vetterli and Kovacevic, 1995]虽然在信号处理方面有很好的性能，但目前还不清楚它如何也可以应用于RKHS方法和SV算法。

最后是基础追求，如?Chen, 1995, Chen等人，1995]所述。其想法是选择一个扩展，以便coe?消费者有最小的“1标准”，而不是框架理论中选择的“2标准”。这，或者更准确地说，在?Olshausen和Field, 1996]中的公式将被用来证明SV机器和稀疏分解之间的等价性。要了解更多关于稀疏性的细节，可以参考关于小波的教科书，例如?Strang和Nguyen, 1996]。

因此，我们从已知的f(17)和sparsi?ed risk function of ?Olshausen and Field, 1996]，即

X

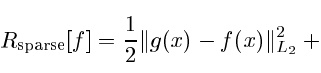
（129）



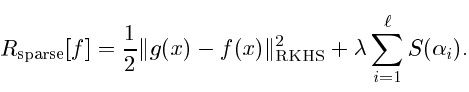
i = 1

在SV的情况下，我们将使用S(?) = j?j,然而迪吗?可以使用不同的函数S来实现coe?字母系数?。最小化(130)会导致凸规划问题，这些问题可以在未来得到解决。

为了推导出与SV机器相似的方程，必须用L2距离w.r.t.来代替?Olshausen和Field, 1996]用RKHS距离?Girosi, 1998]。因此,莫迪吗?Ed稀疏风险函数如下:

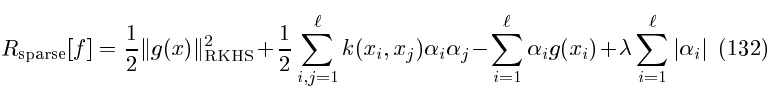


（130）



（131）

此外，我们要求在RKHS中g和f在常数函数1上的投影为0 (hf 1i = hg 1i = 0)，核函数在1上的投影归一化为某个常数(例如P1)，即hk(x ?) 1i = 1。这些约束转化为附加条件“i=1 ?i = 0”。将(129)插入(131)并利用(126)和(127)产量



这与计算沃尔夫对偶对“{不敏感硬损失情况”时得到的方程相同，如(2)所述。

?Girosi, 1998，附录3]还包含一个证明的草图，说明如何将这种推理应用于标准SV“{不敏感损失函数情况”。读者可参阅原出版物了解更多详情。

容量控制50

8容量控制

到目前为止，所有的推理都是基于假设存在确定模型选择参数的方法，如正则化常数或径向基核的长度尺度。我们将布里干酪吗?y简述如何对SV机器和类似的基于内核的算法执行容量控制。以下论述的主要部分是基于?Williamson等人，1998年]。我们建议读者阅读原文以获得详细的阐述，因为完整的描述将大约使当前论文的长度增加一倍。

简而言之，这些证明是基于一个在统计学习理论领域显然是新奇的观点。假设类被描述为一个线性算子映射，从一个可能的?将特征空间中的一维单位球转化为一维空间。然后通过算子的熵数确定类的覆盖数。这些数字表征了算子的紧度，可以用由机器使用的核函数归纳出的积分算子的特征值为界。

熵数，覆盖数，VCSS{引理和退火熵

让我们首先介绍一组基本工具。第一个成分是泛化误差的界，它是熵数n或它的逆函数，即模型类的覆盖数n(?)。一个典型的一致收敛结果具有一般形式

h我

P ' fR;多少钱?c1(的)E N (?F ' 21”)e; ? ?' = c2:

(133)

其中'为样本数量，Remp为经验风险，R为预期风险。常数c2 ?c1(m)取决于环境。例如，在可实现模型的情况下，Alon等人，1997]我们有c1(') = 12 ' ?= 2 c2 = 36。这些边界16通常用于将右边设为?解出' = ' (?吗?)这被称为样本复杂度。上述结果可用于给出泛化误差结果，将其应用于使用标准技术的损失函数诱导类?Bartlett等，1996,Lemma 17]， a ?Williamson近距离观察等(133)al.， 1998]。表明相关的量是ln E ?N(?F ' ' 1) ?这很重要，或者更准确地说，它的大小相对于。然而，为了使具有度规d(X? y)的集合X的覆盖数N(? X?d)为X的最小元素个数xi，使任意X 2x与这些元素之间的最小距离小于等于?，即极小d(X? xi) ?。X (w.r.t. d)的熵值是n(X) = n(X d) = inff?> 0: N (X ? ? ?d) ng。因此N (? ?F ?' '1)表示?{模型类F w.r.t.的覆盖编号为' '1 指标。我们组N”(? ?F):= sup16xThe 1?:::?x ' rst N(结果??F ?' 1 ")。这一类型由Vapnik和Chervonenkis(见例证:Vapnik和Chervonenkis, 1971)发表，然后又“被Sauer ?1972]和Shelah ?1972独立地重新发现”。因此，人们可以把这个原始结果称为VCSS引理，而不只是叫它绍尔引理。



容量控制51

该表达式以退火熵h i为界，更符合实际应用

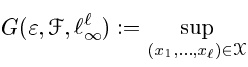


（134)

这是由于对数的凸性。最后,期望

被所有可能的“{样本的sup，即h i ln N(?F ' '1）

（135）



一个被称为生长函数的量。这个量通常又由VC维的表达式限定。参见?Vapnik, 1982, 1995, Anthony, 1997, Vidyasagar, 1997, Williamson, 1998, Anthony和Bartlett, 1999]了解更多关于跳跃机制如何工作的细节。

下面我们将只讨论生长函数，或者更准确地说，我们将利用它



(136)

特别有趣的是Nm(?)的函数逆。F)，即结果的熵数可以更容易地用后一种符号表示。在8.4节中描述的关键思想涉及Nm(?)的直接计算。F)以不涉及组合维(如VC-或脂肪粉碎维)的方式。

8.2为什么不在SV回归中使用VC维?有人可能认为VC维是SV回归的一个合理概念。正如我们将在一个小示例中显示的那样，情况并非如此。问题是VC维不包含任何尺度信息，因此在大多数情况下过于保守。

命题17(高斯rbf{带in的核?夜VC维)表示

r是任意正数，而c2rn是紧集。函数类

F:= ffjf = X ik(xi ?) with xi 2 C X ?i?jk(ξxj) ?1g (137) I I ?j

这里k是一个高斯rbf核?夜间VC维。

证明苏吗?证明对于每个' 2n，存在一个集合

X = f(x1 y1):::(X ' y ')g C ?f;1 1g大小'(138)，可以被非零边距粉碎。考虑一个集合X，其中p不包含重复的xi。可以证明存在一个函数f~ = i ?~ik(xi ?)使得f(xi) = yi对于所有的1 ?我吗?’，因为matrixPk(xi xj)的全秩为

p rbf-kernels ?Micchelli, 1986]。现在设r p:= i?j ~我?~jk(xi xj) > 0 and f:= 1=rf~。绕过构造f 2f和f(xi) = 1=ryi，因此X被1=r的边距粉碎。对于任意'，f有?夜间VC维



容量控制52

即使C是紧凑的。

因此VC维在SV回归中并不合适，因为特征空间中权重向量的长度正是SV方法中使用的数量。因此，绝对有必要使用与尺度相关的量，如(level) fat VC维，或直接应用于泛函分析工具的更基本的量，如熵和覆盖数，而不必绕道通过某些组合推理

8.3关于熵数的有用定理

有人可能会问，使用熵数n而不是它们的逆函数，生长函数(也称为覆盖数)n (?F ' m1）.事实上，计算也可以用后者进行，但是，代价是相当复杂的符号。在处理运算符时，熵数是一个比覆盖数更自然的量(它们是连续的)。一个简单的例子将说明这一点。

用n(F)表示某个函数类F的熵数，则该函数类的熵数由某个常数c缩放为n(cF) = jcj?n(F)。用覆盖数表示的符号完全隐藏了这个关系。

除了函数类(即集合)的熵数外，我们还需要操作符的熵数概念，因为我们将尝试通过连接操作符来构造函数类。一个算子的熵数T: A !B n(T)是de?ned为单位球UA图像的熵数，即?(TUA)。下面的定理将会派上用场。

设n2n, X是希尔伯特空间，并表示S: X !' n1 一个线性算子。那么存在一个常数c，使得:

2 n (S) ?ckSk ?n;1 log ?1 + mn?;12 (139)

当从特征空间映射到m时，这个定理是有用的1．

定理19 (Carl and Stephani ?1990])2吗??？?？j ??？?？0是一个非负数和的非递增序列

Dx = (?对于x = (x1 x2::: xj:::) 2 ' p(140)是' p到自身的对角运算符，由序列(?j)j生成。对于所有的n2n，

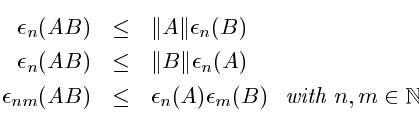
一口n;1j (1 ?2 ? ?? j) 1j ?n (D) ?6一口n; 1j (1 ?2 ? ?? j) 1如:(141)

同样的事实也适用于SV分类。然而，在那里，在?Vapnik, 1995]中引入了一个标度敏感量，从而避免了上述问题。从本质上导出了脂肪粉碎VC维的界限。



容量控制53

我们将利用这个结果来利用规范?C内核并不需要将数据映射成球，而是一些受约束的对象，比如半径快速衰减的超椭球。下面的命题将用于连接几个操作符来生成集，这些集尽可能接近我们想要限制熵数的实际集。命题20(算子串联的熵数)假设X, Y, Z是巴拿赫空间，A: X !Y, b: Y !Z是线性算子。然后是AB: X !Z满足



（142)

（143)

（144）

8.4如何在SV回归中使用熵数

策略如下。首先，导出语句来约束特征空间中映射图像的形状。然后利用它构造运算符，最后，当映射回m时应用到莫里定理1，即在m{样本上计算函数。

为此，我们需要一个语句来限制我们可以对特征空间中的数据图像执行的操作类型，即?(X)。这是默瑟定理的直接结果。

命题21 (Mapping ?(X) into ' 2， ?Williamson et al.， 1998])回想一下定理2的符号。此外，设A为对角线映射

FORMULA

FORMULA

（145)

然后A pmaps jaj?()j X2)切成2。这个结果的结果是，不存在轴平行椭球E (2ckp包含侧边长j的(也)轴平行六面体B)j，这样E将包含?(X)。也就是当且仅当?(X) E:因此?(X)包含平行六面体转角附近的一组非零元素测度。一旦我们知道?(X) \?我们可以用这个结果来构造一个从' 2的单位球到椭球E的逆映射，使?(X) E如下图所示。

X



？? (TX)



A;1 k A kk

k kU”2

（146）

E英国k

因此，我们寻找一个算子A: ' 2 !' 2使A(U)2)?E:我们可以通过构造这样的A来确保这一点



(147）

容量控制54

与RA:= Ckk(p j=aj)jk ' 2。从命题21可以得出，所有RA < 1的算子A将满足A(U '2)?E:我们称这种比例(逆)算子为可容许的。让我们先假设已知b的形状，以便最终处理特征空间中具有有限权向量的一类函数(即kwk ?Rw)在m{样本Xm:= fx1::: xmg X上求值，我们需要引入一个求值算子S?(Xm)。可以理解为?(Xm)表示从X到特征空间的元素映射。

S ? (Xm): U '2 ！“米1 w S ? (Xm): 7 !(hw ?(x1)i::: hw ?(xm)i)(148)是S?(xm) (RwU ')的熵值2我们对SV机器感兴趣。例如，我们可以应用定理18用k?(xi)k为S的范数定界。这将导致与Vapnik ?1995]所获得的模式识别界限相似的界限，不过，由于这些估计需要在脂肪粉碎维度上限定?n，因此对数项略好一些。然而，一个人可以通过利用e形状的信息做得更好，Eq.(149)显示了一个人可以遵循的进一步推理。



U '2trw (Xm   
Xx Xx) Xx x  
RwU”2 一个

/“米  
xx < 1  
(1 (Xm));

RwE

(149）

xx

特别是Williamson等人推出了以下界

? n (S ? (Xm) RwU ' 2) ?Rw ? n (SA); 1 ? (Xm)) ?n 1 ?n 22正吗?n1 n2nr w ?n1(SA;1?(Xm(150)))?n2(A):这可能是由于特征空间中的线性，即

年代? (Xm) x = SA;1Ax ? (Xm):

（151）

因此，总体策略是获得E的一个界(或估计)，然后应用(150)来获得总体熵数，然后将这些熵数依次代入(136)中引入的误差界。

包围(X)的形状

如前所述，第一个估计E形状的方法是在Guyon等人，1993年，Vapnik, 1995年进行的。假设E是球的形状。结果呢?Sch?Olkopf et al.， 1995]指出后者的半径r可以通过求解二次规划问题来估计。然而，这种方法有一个根本的缺点，这并没有在?Vapnik, 1995]中得到解决:半径r是作为训练数据给出的样本m的估计。为了给出可靠的边界，我们必须给出半径为2m{从相同分布中抽取样本iid的上限

容量控制55

m{样本。因此，我们不能简单地将r代入定理18以获得良好的界限，而是例如在幸运框架中做一个陈述?到目前为止，在SV文献中主要忽略了这一点。

类似的方法可以应用于估计包围数据的椭球的几个半径，从而由于操作符A而得到改进的边界。必须采取有效的措施来确保从m样本中获得的估计值能够对2m样本进行良好的预测。看到了吗?Williamson et al.， 1998, shaw - taylor and Williamson, 1999, Smola, 1998]在这个话题上有更多的细节。

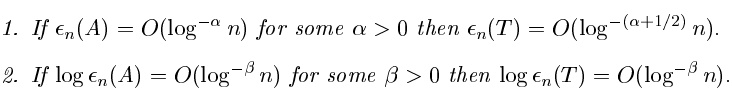
最后存在一种不需要任何运气参数的边界?(X)解析方法。Williamson等人证明，对于某些核，E的半径多项式地甚至是指数地衰减。简单地说，对于平移不变核函数，其半径本质上衰减速度与核函数的傅里叶谱一样快。例如，高斯rbf核k(x y) = exp(;kx;yk2)产生的半径像rj / exp(;?j2)一样衰减。再次参见?威廉姆森等人，1998]的细节。我们现在布里干酪吗?你想想怎么做?’2)渐近地依赖于E.18的半径

命题22(指数{多项式衰减，Williamson et al. ?1998])假设k是j = ?2e的Mercer核;J呢?？p > 0。然后ln ?;n 1(A: ' 2 !' 2) = O(lnp+1 pn)

也可以证明上述命题中的速率是渐近紧的。这些利率可以与定理18结合成一个以?n为界的总体利率。

引理23 (Rate bounds on ?n, Williamson et al. ?1998])假设k是Mercer核，并假设a是与它相关联的缩放算子de?内德(147)。





这个引理表明，在第一种情况下，Maurey定理允许T的熵数指数渐近改进，而在第二种情况下，它是a?Ords none(因为熵数衰减得很快)。简而言之，我们总能得到比Maurey定理更好的收敛速度因为我们不是在处理任意映射到?夜间维空间。另一方面，从VC{维论证(cf. ?Alon et al.， 1997])中获得的速率充其量只能与Maurey定理的速率相比较。因此，我们可以观察到使用内核可以得到一个信号?不能改进超过当前的“可用”界限。

由于篇幅的限制，这里不可能更详细地解释这些结果(包括常量)。特别是我们把自己限制在 18E和由核k引出的特征值j之间的联系由定理21决定



讨论56

这篇论述概述了如何计算学习速率。然而，对于一个成功的学习算法来说，常数的良好估计(不仅仅是速率)是至关重要的。我们建议读者参考?Williamson等人，1998]来计算后一种情况，以及通过计算在这篇简要大纲中简单优化的内容来获得更严格边界的算法。

9的讨论

由于在SV研究领域已经做了相当多的工作，所以几乎不可能编写一个关于SV回归的教程，其中包括所有对该领域的贡献。这也完全超出了教程的范围，而应该归入有关这个问题的教科书(参见例如Sch?olkopf et al. ?1999]为当前技术状态的快照，或Vapnik ?1998]为统计学习理论的概述)。作者仍然希望这项工作提供了一个对SV研究现状不太偏颇的看法。我们(除其他外)故意省略了下列主题。

9.1缺少话题

首先，我们应该提到数学规划领域。从一个完全的di?目前已经开发出的透视算法在其思想上与SV机器非常相似。一个好的入门可能是?布拉德利等人，1998]。另见?Mangasarian, 1964, 1969, Street and Mangasarian, 1995]。Bennett ?1999]对数学规划与SV机器之间的联系进行了全面的讨论。

另一个最近被追求的想法是用SV机器进行密度估计?Weston et al.， 1999, Vapnik, 1999]。这里利用了累积分布函数是单调递增的这一事实，它的值可以用变量con?通过选择di?在损失函数中的不同值。

此外，人们可能会考虑选择di?同时计算内核宽度。在标准SV情况下，这几乎是不可能的，除非由de?得到新的核函数作为di?不同比例的。这是由于一旦选择了一个正则化算子，最小化正则化风险函数的解决方案必须扩展为相应的P P ?Kimeldorf和Wahba, 1971, Cox和O'Sullivan, 1990]的green函数。因此，人们不得不求助于线性规划。Weston等人? ?1999]使用内核字典，不像Chen等人? ?1995]。

最后，本综述的重点是方法和理论，而不是应用。这样做是为了限制博览会的规模。技术的状态，甚至记录的表现报告在?M?uller等人，1997,Drucker等人，1997,Stitson等人，1999,Mattera和Haykin, 1999]。在许多情况下，神经网络方法也可能达到类似的效果，然而，只有当许多参数得到最佳调整时，这在很大程度上取决于实验者的技能。换句话说，一个人不应该认为SV机器是一个\银弹。”另一方面，因为只有很少的关键

讨论57

参数(例如正则化和内核宽度)，使用SV机器可能更容易实现相同的结果。

9.2开放问题

作为一种非常活跃的组织，仍存在许多有待进一步研究解决的问题。在那之后，算法开发似乎已经找到了一个更稳定的阶段，其中最重要的似乎是尝试从规范?核函数的C性质。在这种情况下，SV机器，或类似的方法源于线性规划正则化器，是否会导致最令人满意的结果将是有趣的。

此外，还必须设计某种类型的“幸运框架”(Shawe-Taylor et al.， 1996a)，用于多个模型选择参数，类似于贝叶斯统计中的多个超参数和自动关联检测?MacKay, 1991, Bishop, 1995]，以使SV机器更少地依赖于实验者的技能。利用正则化算子、高斯过程和先验之间的桥梁(见Williams, 1998])来说明SV机器的贝叶斯风险边界，以便将预测与VC理论的预测进行比较，也是值得的。此外，在SV机器环境下开发的优化技术也可以用于处理高斯过程设置中的大型数据集。

先验知识似乎是SV回归中的另一个重要问题。而不变性可以通过虚拟SV机制和特征空间的限制以原则性的方式包含在模式识别中。1997年olkopf,原理图吗?Olkopf等人，1998]，它仍然不清楚如何(可能)更微妙的属性，作为回归所需，可以处理e?在?Smola和Sch?然而，这可能是如何实现这一目标的第一个迹象。

还应该考虑简化集方法来加速大型数据集的预测(可能也包括训练)阶段。这个主题非常重要，因为数据挖掘应用程序需要能够处理数据库的算法，这些数据库通常比SV回归的当前实际大小至少大一个数量级(100万个样本)。

还有很多方面，比如更多的依赖于数据的泛化界限，e?客户训练算法，自动内核选择程序，向量值回归，以及许多已经进入标准神经网络工具包的技术，都将在未来被考虑。

确认

这项工作得到了DFG的部分资助(# Ja 379/71)。作者感谢Chris Burges, Klaus-Robert M?uller, Vladimir Vapnik, Jason Weston, Robert Williamson, Andreas Ziehe提供有益的讨论和评论。

SMO回归58的伪代码

SMO回归的伪代码

目标=期望输出向量点=训练点矩阵

过程takeStep (i1、i2)

If (i1 == i2)返回0

alpha1, alpha1\* = i1的拉格朗日乘数y1 =目标?i1)

phi1 = SVM输出点?I1] - y1(错误缓存中)

Kernel (point?i1]，point?i1]) Kernel (point?i1]，point?i2]) Kernel (point?i2]，point?i2) 2\*k12-k11-k22

= alpha1 - alpha1\* + alpha2 - alpha2\*

k11 =

k12的=

k22 =

η=

γ

%我们假设>为0。否则，你必须重复完整的%推理类似(计算L和H %的目标函数，并决定哪一个最大

Case1 = case2 = case3 = case4 = finished = 0 alpha1old = alpha1, alpha1old\* = alpha1\* alpha2old = alpha2, alpha2old\* = alpha2\* delta\_phi = phi1 - phi2

完成时!

%这个循环最多传递三次

% case变量需要避免尝试两次小的更改，如果(case1 == 0) &&

(alpha1 > 0 || (alpha1\* == 0 && deltaphi > 0)) && (alpha2 > 0 || (alpha2\* == 0 && deltaphi < 0)))

计算L, H (wrt。α1、alpha2)

如果L < H

A2 = 2 - δ /eta

a = min(a, H)

a = max(L, a)

A1 = alpha1 - (a2 - alpha2)

如果变化大于某些eps，则更新alpha1、alpha2

其他的

完成= 1

endif

case1 = 1 ?

Elseif (case2 == 0) &&

(alpha1 > 0 || (alpha1\* == 0 > 2)) && (alpha2\* > 0 || (alpha2 == 0 && deltaphi > 2))计算L, H (wrt。α1,alpha2 \*)

SMO回归59的伪代码

如果L < H

A2 = 2\* + (- 2)/

a = min(a, H)

a = max(L, a)

A1 = α 1 + (a2 - α 2\*)

如果更改大于某些eps，则更新alpha1, alpha2\*

完成= 1

endif

例2 = 1 ?

Elseif (case3 == 0) &&

(alpha1\* > 0 || (alpha1 == 0 && deltaphi < 2)) && (alpha2 > 0 || (alpha2\* == 0 && deltaphi < 2))计算L, H (wrt。α1 \*,alpha2)

如果L < H

A2 = 2 - (- 2)/

a = min(a, H)

a = max(L, a)

A1 = alpha1\* + (a2 - alpha2)

如果更改大于某些eps，则更新alpha1\*， alpha2

完成= 1

endif

case3 = 1 ?

Elseif (case4 == 0) &&

(alpha1\* > 0 || (alpha1 == 0 && deltaphi < 0)) && (alpha2\* > 0 || (alpha2 == 0 && deltaphi > 0))

计算L, H (wrt。α1 \*,alpha2 \*)

如果L < H

A2 = 2\* + delta /eta

a = min(a, H)

a = max(L, a)

a = α 1\* - (a - α 2\*)

如果更改大于某些eps，则更新alpha1\*， alpha2\*

完成= 1

endif

case4 = 1 ?

其他的

完成= 1

endif

更新deltaphi

endwhile

更新阈值以反映拉格朗日乘数的变化更新错误缓存使用新的拉格朗日乘数

如果alpha1(\*)、alpha2(\*)的变化大于某些eps

返回1

SMO回归60的伪代码

其他的

返回0

endif endprocedure

过程examineExample (i2)

y2 =目标?i2)

2, 2\* =拉格朗日乘数的i2, C2\* =约束的i2

phi2 =支持向量机输出点?I2] - y2(在错误缓存中)

if ((phi2 > && alpha2\* < C2\*) ||

(phi2 < && alpha2\* > 0) ||

(-phi2 > && alpha2 < C2) ||

(-phi2 > && alpha2 > 0))

if (number of non-zero & non-C alpha > 1) i1 =第二选择启发式的结果if takeStep(i1,i2)返回1

endif

循环所有非零和非c alpha，随机开始i1 =当前alpha的身份

如果takeStep(i1,i2)返回1

endloop

循环遍历所有可能的i1，并使用随机start i1 = Loop变量

如果takeStep(i1,i2)返回1

endloop endif

返回0 endprocedure

主程序:

初始化alpha和alpha\*数组为零初始化阈值为零

numChanged = 0

examineAll = 1

SigFig = -100

LoopCounter = 0

而((numChanged > 0 | examineAll) | (SigFig < 3)) LoopCounter++

numChanged = 0 ?

如果(examineAll)

循环I遍历所有训练示例

numChanged + = examineExample(我)

其他的

循环I遍历那些alpha不为0且不为C的例子

引用61年

numChanged += examineExample(I) endif

if (mod(LoopCounter, 2) == 0)

MinimumNumChanged = max(1, 0.1\*NumSamples) else

MinimumNumChanged = 1

endif

if (examineAll == 1)

examineAll = 0

elseif (numChanged < MinimumNumChanged) examineAll = 1

endif

endwhile

endmain

参考文献

y s . Abu-Mostafa。提示。神经网络计算，7(4):639{671,1995。

m·a·艾泽曼，e·m·布雷弗曼和l·i·罗佐诺。模式识别学习中势函数法的理论基础。自动化和远程控制，25:821{837,1964。

N.阿隆，S. Ben-David, N. Cesa-Bianchi和D. Haussler。尺度敏感维度、一致收敛性和可学习性。地球物理学报，44(4):615{631,1997。

m·安东尼。文章中学习的概率分析?社会神经网络:PAC模型及其变体。神经计算研究，1:1{47,1997。http://www.icsi.berkeley.edu/ ~ jagota / nc。

M.安东尼和P. Bartlett。艺术的学习理论脸部用的神经网络。剑桥大学出版社，1999年。

n Aronszajn。产品的研究及其应用。Proc。剑桥费罗斯。Soc。39:133{153年,1944年。

n Aronszajn。再生核理论。反式。阿米尔。数学。Soc。， 68: 337{404, 1950。

P.巴特利特，P.朗，R.威廉姆森。脂肪粉碎与实值函数的可学性。计算机与系统科学，52(3):434{452,1996。

右眼传达员。自适应控制过程。普林斯顿大学出版社，新泽西州普林斯顿，1961年。

班尼特。结合支持向量和数学规划方法进行归纳。在b .原理图吗?olkopf, C.J.C. Burges和A.J. Smola编辑，

引用62年

核心方法的进展- SV学习，307页{326，剑桥，MA, 1999。麻省理工学院出版社。

P.J. Bickel, C.A.J. Klaassen, Y. Ritov和J.A. Wellner。E ?半参数模型的客户端和自适应估计。J.霍普金斯出版社，巴尔的摩，ML, 1994年。

c . m .主教。模式识别的神经网络。克拉伦登出版社，牛津，1995年。

美国博赫纳。傅里叶积分的讲座。普林斯顿大学出版社，新泽西州普林斯顿，1959年。

B. E. Boser, I. M. Guyon和V. N. Vapnik。一种最优边界分类器的训练算法。D. Haussler，编辑，第五届ACM COLT年度研讨会，144页{152，匹兹堡，宾夕法尼亚州，1992年。ACM出版社。

P. Bradley和O. Mangasarian。海量数据识别通过线性支持向量机。数学编程技术报告98-05，威斯康星大学麦迪逊，1998。

P. S.布拉德利，U. M.法耶兹和O. L.曼加萨里安。数据挖掘:概述和优化机会。技术报告98-01，威斯康星大学，计算机科学系，麦迪逊，1998。INFORMS计算期刊，已提交。

J. R. Bunch和L. Kaufman。计算惯性和求解对称线性系统的一些稳定方法。计算数学，31:163{179,1977。

J. R. Bunch和L. Kaufman。inde的计算方法?二次规划问题。线性代数及其应用，页341{370,1980。

J. R. Bunch, L. Kaufman和B. Parlett。对称矩阵的分解。数学学报，27:95{109,1976。

c·j·c·伯吉斯。Simpli吗?支持向量决策规则。第十三届机器学习国际会议，第71页{77，圣马特奥，加利福尼亚州，1996年a。摩根考夫曼。

C. J. C. Burges和B. Sch?olkopf。提高支持向量学习机的准确性和速度。M. Mozer, M. Jordan和T. Petsche，编辑，神经信息处理系统进展9,375页381，剑桥，麻省，1997。麻省理工学院出版社。

C.J.C. Burges。私人通信,1996 b。

C.J.C. Burges。用于模式识别的支持向量机教程。知识发现与数据挖掘，1998。在出版社。

引用63年

C.J.C. Burges。基于核方法的几何和不变性。在b .原理图吗?olkopf, C.J.C. Burges和A.J. Smola，编辑，核方法的进展|支持向量学习，89页116，剑桥，马萨诸塞州，1999。麻省理工学院出版社。

Carl和I. Stephani。熵，紧性，和算子的近似。剑桥大学出版社，英国剑桥，1990年。

陈。基础的追求。博士论文，斯坦福大学统计学系，1995年。

S. Chen, D. Donoho和M. Saunders。原子根据追求分解。技术报告479，斯坦福大学统计学系，1995年。V. Cherkassky和F. Mulier。从数据中学习。威利，纽约，1998年。R.R. Coifman和M.V. Wickerhauser。基于熵的最佳基选择算法。《信息理论与应用》，1992。C. Cortes和V. Vapnik。支持向量网络。《学习》，1995年第20期。

D.考克斯和F.奥沙利文。惩罚似然和相关估计的渐近分析。《统计年报》，1990年第18期。

C.坎贝尔和J.肖-泰勒。支持向量机中动态调整内核。技术报告NC-TR-98-017，英国伦敦大学皇家霍洛威学院，1998。

Dantzig G B。线性规划与扩展。普林斯顿大学出版社，新泽西州普林斯顿，1962年。

即Daubechies。关于小波的十讲。CBMS-NSF应用数学区域会议系列。SIAM，费城，宾州，1992。1990年在马萨诸塞州洛厄尔举行的CBMS-NSF小波和应用会议上的说明。

德鲁克，C. J. C. Burges, L. Kaufman, A. Smola和V. Vapnik。支持向量回归机器。M. Mozer, M. Jordan和T. Petsche，编辑，《神经信息处理系统进展》9,155页{161，剑桥，马萨诸塞州，1997。麻省理工学院出版社。

用径向函数插值散点数据。崔国强，舒梅克，乌特拉斯，编辑，多元近似主题。纽约学术出版社，1987年。

A. El-Bakry, R. Tapia, R. Tsuchiya和Y. Zhang。非线性规划的牛顿内点法的公式和理论。J.最优化理论与应用，89:507{541,1996。

r·弗莱彻。实用的优化方法。约翰·威利父子公司，纽约，1989年。

引用64年

T.T. Frie和R.F. Harrison。线性规划支持向量机的模式类?阳离子和回归估计以及集合约简算法。TR RR-706，女子大学?高龄,她?古人,英国,1998年。

S. Geva, J. Sitte和G. Willshire。一个神经元卡车靠背{上部。神经网络国际联合会议，巴尔的摩，ML, 1992。IEEE。

f . Girosi。稀疏近似和支持向量机之间的等价性。神经网络学报，10(6):1455{1480,1998。

F.吉洛西，M.琼斯和T.波吉欧。先验、稳定器和基函数:从正则化到径向、张量和加性样条。人工智能备忘录第1430号，麻省理工学院，1993年。

h·戈尔茨坦。经典力学。爱迪生-韦斯利，雷丁，马萨诸塞州，1986年。I.盖伊恩，B.博瑟和V.瓦普尼克。自动容量调整非常大的vc维类?Stephen jose -e Hanson、Jack D. Cowan和C. Lee Giles编辑，《神经信息处理系统的进展》，第5卷，第147页。摩根·考夫曼，加州圣马特奥，1993年。

w·H ? ardle。应用非参数回归，《计量经济社会专题》第19卷。剑桥大学出版社，1990年。

t.j. Hastie和r.j. Tibshirani。广义加性模型，关于统计和应用概率专著43卷。查普曼与霍尔出版社，伦敦，1990年。

微积分。神经网络:一个综合的基础。麦克米伦，纽约，1998年。第二版。

K. Hornik, M. Stinchcombe和H. White。多层前馈网络是一种通用的逼近器。神经网络，2:359{366,1989。

p•j•休伯。稳健性统计:综述。安。中央集权。, 1972年43:1041。

p•j•休伯。健壮的统计数据。约翰·威利父子，纽约，1981年。

IBM公司。IBM优化子例程库指南和参考。IBM系统杂志，1992年31日。sc23 - 0519。

最大化策略优化公司。使用CPLEX可调用库。手册,1994年版。约阿希姆。使大规模支持向量机学习成为现实。在b .原理图吗?olkopf, C.J.C. Burges，和A.J. Smola，编辑，核方法的进展|支持向量学习，169页184，剑桥，马萨诸塞州，1999。麻省理工学院出版社。

w . Karush。带不等式作为边约束的多变量函数的极小值。硕士论文，芝加哥大学数学系，1939年。

引用65年

l·考夫曼。解决了支持向量分类中出现的二次规划问题。在b .原理图吗?olkopf, C.J.C. Burges，和A.J. Smola，编辑，核方法的进展|支持向量学习，第147页{168，剑桥，马萨诸塞州，1999。麻省理工学院出版社。

G.S. Kimeldorf和G. Wahba。随机过程的贝叶斯估计和样条平滑之间的对应关系。安。数学。中央集权。， 2: 495{502, 1971。

h。w。库恩和a。w。塔克。非线性规划。第二届伯克利数学统计和概率研讨会，第481页，伯克利，1951年。加州大学出版社。

i。j。拉斯蒂格，r。e。马斯滕，和d。f。香诺。线性规划中mehrotra预测-校正内点法的实现。普林斯顿技术报告SOR 90-03。普林斯顿大学土木工程与运筹系，1990年。

大卫·j·c·麦凯。贝叶斯模型与神经网络。博士论文，计算与神经系统，加州理工学院，帕萨迪纳，CA, 1991。

麦迪奇和纳尔逊。多元插值和条件正de?夜间的功能。2计算数学，54(189):211{230,1990。

S. Mallat和Z. Zhang。时频字典中的匹配追踪。信号处理技术，2007。

开环Mangasarian。线性规划的线性和非线性模式分离。运筹学，13:444{452,1964。

开环Mangasarian。非线性规划。纽约麦格劳-希尔，1969年。

D. Mattera和S. Haykin。支持向量机用于混沌系统的动态重构。在b .原理图吗?olkopf, C.J.C. Burges和A.J. Smola，编辑，核方法的进展|支持向量学习，第211页{242，剑桥，马萨诸塞州，1999。麻省理工学院出版社。

Maurey。在:Remarques sur un resultat non - publi。e de B. Maurey”的G. Pisier。在数学中心，编辑，Seminarie d' analyses fonctionelle 1980{1981, Palaiseau, 1981。

他的麦考密克。非线性规划:理论，算法，和应用。Wiley-Interscience，纽约，纽约，1983年。

S. Mehrotra和J. Sun。关于(原{对偶)内点方法的实现。中国机械工程，2017,29(4):574{681,1992。

引用66年

j·默瑟。正负型函数及其与积分方程理论的联系。费罗斯。反式。罗伊。Soc。伦敦，A 209: 415{446, 1909。

y迈耶。小波和运营商。剑桥大学出版社，1992年。由赫尔曼首次在法国出版，编辑部。, 1990年巴黎。

c.a Micchelli。散乱数据的插值:距离矩阵和条件正解?夜间的功能。构造近似，2:11{22,1986。

M. L. Minsky和S. Papert。感知器。麻省理工学院出版社，马萨诸塞州剑桥，1969年。再见摩尔。正确的正厄米矩阵。公牛。阿米尔。数学。Soc。23(59): 66{67年,1916年。

诉Morozov。解决不正确问题的方法。施普林格1 - 1984。

K.-R。M ?uller, A. Smola, G. R?atsch, b .原理图吗?olkopf, J. Kohlmorgen和V. Vap- nik。用支持向量机预测时间序列。在W. Gerstner, A. Germond, M. Hasler和j . d。摇身Nicoud、编辑社会神经网络| ICANN'97, 999页{1004，柏林，1997。计算机科学课堂讲稿，第1327卷。

穆塔夫和桑德斯。MINOS 5.1用户指南。技术报告SOL 83{20R，斯坦福大学，CA，美国，1983。修订后的1987人。

D. Nguyen和B.寡妇。卡车司机:神经网络中自我学习的一个例子。神经网络国际联合会议论文集，第3卷，第357页，1989年。

新泽西州尼尔森。学习机器:可训练模式分类系统的基础。McGraw{山,1965年。

尼奎斯特。电报传输理论中的若干主题。反式。《a.i.e.e.》，第617页，644页，1928年。

b.a Olshausen和D.J. Field。通过学习自然图像的稀疏编码而产生的简单细胞接受特性。自然,381:607{609,1996。

E. Osuna, R. Freund和F. Girosi。支持向量机:培训和应用。麻省理工学院人工智能实验室AIM-1602技术报告, 1996年。

E. Osuna, R. Freund和F. Girosi。一种改进的支持向量机训练算法。J. Principe, L. Gile, N. Morgan和E. Wilson，编辑，信号处理的神经网络VII | 1997年IEEE研讨会论文集，第276页285页，纽约，1997年。IEEE。

大肠Parzen。连续参数时间序列的回归分析。《第四届伯克利数理统计与概率专题讨论会论文集》，第一卷，第469页。

引用67年

j·普拉特。基于序列最小优化的支持向量机快速训练。在b .原理图吗?olkopf, C.J.C. Burges，和A.J. Smola，编辑，内核方法的进展|支持向量学习，第185页{208，剑桥，MA, 1999。麻省理工学院出版社。

方法的。最优非线性联想回忆。生物控制，19:201{209,1975。

F. Riesz和B.S. Nagy。功能分析。Frederick Ungar出版公司，1955年。

最初里普利。模式识别与神经网络。剑桥大学出版社，1996年。

美国Saitoh。核再生理论及其应用。朗文Sci - enti ?c & Technical, Harlow，英格兰，1988年。

n·萨奥尔。集族的密度。组合理论学报，13(5):583{147,1972。

C. Saunders, M.O. Stitson, J. Weston, L. Bottou, B. Sch?olkopf和A. Smola。支持向量机|参考手册。技术报告CSD-TR-98- 03，英国伦敦大学皇家霍洛威学院计算机科学系，1998年。

I.J.勋伯格。度量空间和完全单调函数。安。的数学。39:811{841年,1938年。

I.J.勋伯格。度量空间和正de?夜间的功能。反式。阿米尔。数学。Soc。, 44:522{536、1938 b。

b .原理图? olkopf。支持向量的学习。欧登堡，慕尼黑，1997。b .原理图吗?olkopf, P. Bartlett, A. Smola和R. Williamson。支持向量回归与自动精度控制。L. nikklasson, M. Bod-en，和T. Ziemke，编辑，第八届人工神经网络国际会议论文集，神经计算透视，111{116，柏林，1998a。施普林格-。

b .原理图吗?olkopf, C. Burges和V. Vapnik。提取给定任务的支持数据。U. M. Fayyad和R. Uthurusamy主编，第一届知识发现和数据挖掘国际会议论文集，门洛帕克，1995。AAAI出版社。

b .原理图吗?olkopf, C. Burges和V. Vapnik。在支持向量学习机器中加入不变性。在C. von der Malsburg, W. von Seelen, J. C. Vorbr?ugen和B. Sendho?、编辑、摇身社会神经网络| ICANN'96，第47页{52，柏林，1996。计算机科学课堂讲稿，第1112卷。

b .原理图吗?olkopf, C.J.C. Burges和A.J. Smola。核方法的进展|支持向量学习。麻省理工学院出版社，马萨诸塞州剑桥，1999年。

引用68年

b .原理图吗?olkopf, P. Knirsch, A. Smola和C. Burges。支持向量核扩展的快速近似，以及聚类作为特征空间的近似的解释。P. Levi, M. Schanz, r - j。阿勒斯和F.梅，编辑，文集1998 | 20。《dagm专题讨论会》，第124页{132页，柏林，1998年b。施普林格。

b .原理图吗?olkopf, S. Mika, A. Smola, G. R?atsch, K.-R。M ? ul。基于近似预图像的核主成分分析模式重建。L. Niklasson, M. Bod-en, T. Ziemke，编辑，第八届Arti?社会神经网络，神经计算透视，第147页{152，柏林，1998c。施普林格-。

b .原理图吗?olkopf, P. Y. Simard, A. J. Smola和V. N. Vapnik。支持向量核中的先验知识。M. I. Jordan, M. J. Kearns和S. A. Solla编辑，《神经信息处理系统的进展》，第10卷，第640页，剑桥，马萨诸塞州，1998年。麻省理工学院出版社。

b .原理图吗?olkopf, A. Smola和K.-R。M ? ul。作为核特征值问题的非线性分量分析。神经网络计算，2008。b .原理图吗?olkopf, A. Smola和R. Williamson。一种新的参数化支持向量机。1998年EUROCOLT。提交。

b .原理图吗?olkopf, K. Sung, C. Burges, F. Girosi, P. Niyogi, T. Poggio和V. Vap- nik。高斯核支持向量机与径向基函数的比较。IEEE反式。的迹象。加工，42(5):565{2765,1997。刚建成时香农。交流的数学理论。贝尔系统技术期刊，27:379{423,623{656,1948。

J.肖-泰勒，P.巴特利特，R.威廉姆森和M.安东尼。结构风险最小化框架。技术报告NC-TR-96-032，英国伦敦大学皇家霍洛威学院，1996a。

J.肖-泰勒，P.巴特利特，R.威廉姆森和M.安东尼。数据依赖层次结构上的结构风险最小化。技术报告NC-TR-96- 053，英国伦敦大学皇家霍洛威学院，1996b。

J. shaw - taylor和R.C. Williamson。类的泛化性能?根据观察到的覆盖数。1999年EUROCOLT论文集。提交。

示拉。一个组合的问题!模型和理论的稳定性和秩序?nitary语言。奶嘴?c数学学报，41:247{261,1972。c。g。Small和d。l。McLeish。概率与统计推理中的希尔伯特空间方法。概率与数理统计。John Wiley & Sons，纽约，纽约，1994年。

A.斯莫拉，村田，B. Sch?olkopf, K.-R。M ? ul。支持向量机“-损失”的渐近最优选择。在尼克拉森，波德恩，

引用69年

T. Ziemke，编辑，第八届Arti?社会神经网络，神经计算展望，第105页{110，柏林，1998a。施普林格-。

A.斯莫拉，B. Sch?olkopf, K.-R。M ? ul。支持向量回归的一般成本函数。T. Downs, M. Frean和M. Gallagher，第九届澳大利亚神经网络会议主编，第79页{83，布里斯班，澳大利亚，1998b。昆士兰大学。

a . j . Smola。支持向量学习机器的回归估计。硕士论文，理工大学?在米?unchen, 1996年。

a . j . Smola。学习与内核。博士论文，理工大学?1998年,在柏林。

A. J.斯莫拉和B. Sch?olkopf。提出了一种基于核的模式识别、回归、逼近和算子反演方法。算法，22:211{231,1998a。

A.J.斯莫拉，t.f Frie'和b.s cholkopf。半参数支持向量机和线性规划机。《神经信息处理系统进展》11。麻省理工学院出版社,1998 c。在出版社。

A.J.斯莫拉和B. Sch?olkopf。从正则化算子到支持向量核。《神经信息处理系统进展》10，第343页，圣马特奥，加州，1998年b。

a·j·斯莫拉，b·施?olkopf, K.-R。M ? ul。正则化算子和支持向量核之间的联系。神经网络，11:637{649,1998d。

A.J.斯莫拉，R.C.威廉姆森和b.s cholkopf。正则化主流形的泛化边界。技术报告NC-TR-98-027，英国伦敦大学皇家霍洛威学院，1998e。提交给EUROCOLT 99。M. Stitson, A. Gammerman, V. Vapnik, V. Vovk, C. Watkins和J. Weston。支持向量回归与方差分析的分解内核。在b .原理图吗?olkopf, C.J.C. Burges和A.J. Smola，编辑，核方法的进展|支持向量学习，第285页292，剑桥，马萨诸塞州，1999。麻省理工学院出版社。

c . j .石头。加性回归和其他非参数模型。统计年报，13:689{705,1985。

G. Strang和T. Nguyen。小波和滤波器组。韦尔斯利{剑桥大学出版社,1996年。

wn街和O.L. Mangasarian街。通过容忍训练改进泛化。技术报告MP-TR-95-11，威斯康星大学，麦迪逊，1995。A. N.吉洪诺夫和V. Y. Arsenin。问题的解决。温斯顿，华盛顿特区，1977年。

引用70年

R. J. Vanderbei, A. Duarte和B. Yang。几种内点法的算法和数值比较。技术报告SOR-94-05，普林斯顿大学统计与运算学项目，1994年。r Vanderbei。LOQO:二次规划的内部点代码。关键词:数理统计，运筹学，统计分析r Vanderbei。线性规划:基础与扩展。Kluwer学术出版社，Hingham, MA, 1997年a。

r Vanderbei。LOQO用户手册3.10。技术报告SOR-97-08，统计与运算学，普林斯顿大学，新泽西州，1997b。

诉Vapnik。统计学习理论的本质。施普林格,纽约,1995年。诉Vapnik。统计学习理论。威利,纽约,1998年。

诉Vapnik。关于函数估计的支持向量法的三种注释。在b .原理图吗?olkopf, C.J.C. Burges，和A.J. Smola，编辑，核心方法的进展|支持向量学习，第25页42，剑桥，马萨诸塞州，1999。麻省理工学院出版社。

V. Vapnik和A. Chervonenkis。关于一类感知器的注释。自动化和远程控制，25,1964。

V. Vapnik和A. Chervonenkis。模式识别理论?俄语]。Nauka,莫斯科,1974年。(德文翻译:W. Wapnik & A. Tscherwo- nenkis，《Zeichenerkennung理论》，学术委员会，柏林，1979年)。V. Vapnik, S. Golowich和A. Smola。支持向量方法的函数逼近，回归估计，和信号处理。M. Mozer, M. Jordan和T. Petsche，编辑，《神经信息处理系统进展》9,281页{287，剑桥，MA, 1997。麻省理工学院出版社。

V. Vapnik和A. Lerner。使用广义肖像方法的模式识别。自动化与远程控制，24,1963。

v . n . Vapnik。基于经验数据的依赖性估计。施普林格- Verlag，柏林，1982年。

V. N. Vapnik和A. Y. Chervonenkis。事件相对频率与概率的一致收敛性。Probab理论。及其应用，16(2):264{280,1971。

M.维特里和J.科瓦切维奇。小波与子带编码。普伦蒂斯霍尔,1995年。

维德雅瑟格。学习和概括的理论。施普林格，纽约，1997。

g·丁。观测数据的样条模型。应用数学系列，第59卷，SIAM，费城，1990。

引用71年

g·丁。支持向量机，再现核希尔伯特空间和随机GACV。在b .原理图吗?olkopf, C.J.C. Burges，和A.J. Smola，编辑，核方法的进展|支持向量学习，69页88，剑桥，马萨诸塞州，1999。麻省理工学院出版社。

J. Weston, A. Gammerman, M. Stitson, V. Vapnik, V. Vovk，和C. Watkins。支持向量密度估计。在b .原理图吗?olkopf, C.J.C. Burges和A.J. Smola，编辑，核方法的进展|支持向量学习，293页{306，剑桥，马萨诸塞州，1999。麻省理工学院出版社。

C.K.I.威廉姆斯。用高斯过程预测:从线性回归到线性预测及超越。图形模型中的学习和推理，1998。NCRG/97/012的技术报告。

威廉姆森司令部。统计学习理论与非线性系统辨识的一些结果。非线性控制系统设计研讨会1998 (NOLCOS98)，第2卷，第443页。IFAC,爱思唯尔,1998年。R.C.威廉姆森、A.J.斯莫拉和b.s cholkopf。基于紧致算子熵数的正则化网络和支持向量机的泛化性能。技术报告NC-TR-98-019，英国伦敦大学皇家霍洛威学院，1998。

A. Yuille和N. Grzywacz。运动相干理论。《计算机视觉国际会议论文集》，第344页，华盛顿特区，1988年。IEEE计算机协会出版社。