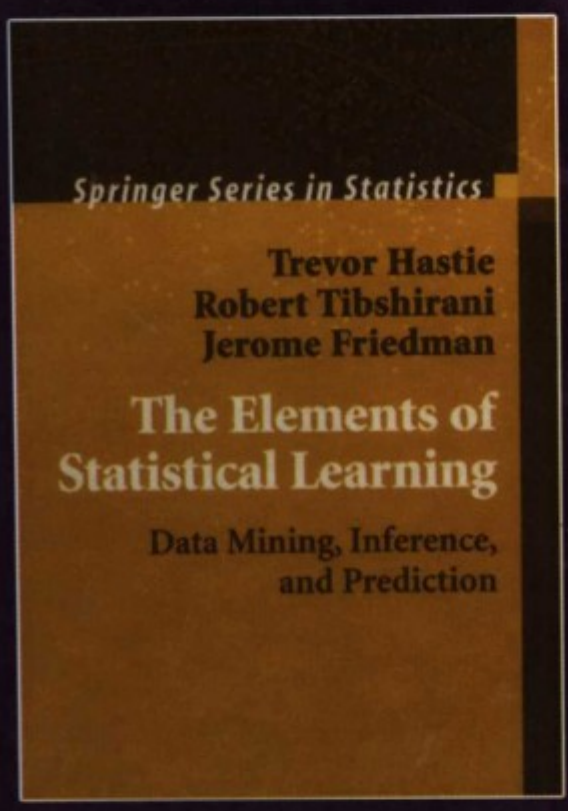
国外计算机科学教材系列

统计学习基础

——数据挖掘、推理与预测

The Elements of Statistical Learning

Data Mining, Inference, and Prediction



Trevor Hastie
[美] Robert Tibshirani 著
Jerome Friedman
范 明 柴玉梅 昝红英 等译

電子工業出版社
Publishing House of Electronics Industry
http://www.phei.com.cn

经典教材

统计学习基础

——数据挖掘、推理与预测

The Elements of Statistical Learning

Data Mining, Inference, and Prediction

随着计算机和信息时代的到来，统计问题的规模和复杂性都有了急剧增加。数据存储、组织和检索领域的挑战导致一个新领域“数据挖掘”的产生。数据挖掘是一个多学科交叉领域，涉及数据库技术、机器学习、统计学、神经网络、模式识别、知识库、信息提取、高性能计算等诸多领域，并在工业、商务、财经、通信、医疗卫生、生物工程、科学等众多行业得到了广泛的应用。

本书试图将学习领域中许多重要的新思想汇集在一起，并且在统计学的框架下解释它们。尽管有些数学细节是必要的，但本书强调的是方法和它们的概念基础，而不是理论性质。本书内容广泛，从有指导的学习（预测）到无指导的学习，应有尽有。包括神经网络、支持向量机、分类树和提升等主题，是同类书籍中介绍得最全面的，适合从事数据挖掘和机器学习研究的读者阅读。

作者简介

Trevor Hastie, Robert Tibshirani和Jerome Friedman都是斯坦福大学统计学教授，并在这个领域做出了杰出的贡献。Hastie和Tibshirani提出了广义加法模型，并出版了专著“Generalized Additive Models”。Hastie的主要研究领域为：非参数回归和分类、统计计算以及生物信息学、医学和工业的特殊数据挖掘问题。他提出了主曲线和主曲面的概念，并用S-PLUS编写了大量统计建模软件。Tibshirani的主要研究领域为：应用统计学、生物统计学和机器学习。他提出了套索的概念，还是“An Introduction to the Bootstrap”一书的作者之一。Friedman是CART、MARS和投影寻踪等数据挖掘工具的发明人之一。他不仅是位统计学家，而且是物理学家和计算机科学家，并先后在物理学、计算机科学和统计学的一流杂志上发表了论文80余篇。

ISBN 7-5053-9331-6

9 787505 393318 >

责任编辑：杜闽燕
封面设计：毛惠庚

本书贴有激光防伪标志，凡没有防伪标志者，属盗版图书

ISBN 7-5053-9331-6　定价：45.00元

国外计算机科学教材系列

统计学习基础

——数据挖掘、推理与预测

The Elements of Statistical Learning

Data Mining, Inference, and Prediction

Trevor Hastie
[美] Robert Tibshirani 著
Jerome Friedman

范 明 柴玉梅 昝红英 等译

電子工業出版社
Publishing House of Electronics Industry
北京 · BEIJING

内 容 简 介

计算和信息技术的飞速发展带来了医学、生物学、财经和营销等诸多领域的海量数据。理解这些数据是一种挑战，这导致了统计学领域新工具的发展，并延伸到诸如数据挖掘、机器学习和生物信息学等新领域。许多工具都具有共同的基础，但常常用不同的术语来表达。本书介绍了这些领域的一些重要概念。尽管应用的是统计学方法，但强调的是概念，而不是数学。许多例子附以彩图。本书内容广泛，从有指导的学习（预测）到无指导的学习，应有尽有。包括神经网络、支持向量机、分类树和提升等主题，是同类书籍中介绍得最全面的。

本书可作为高等院校相关专业本科生和研究生的教材，对于统计学相关人员、科学界和业界关注数据挖掘的人，本书值得一读。

版权贸易合同登记号　图字：01-2002-4937

图书在版编目（CIP）数据

统计学习基础——数据挖掘、推理与预测 /（美）黑斯蒂（Hastie, T.）等著；范明等译.
-北京：电子工业出版社，2004.1
（国外计算机科学教材系列）
书名原文：The Elements of Statistical Learning: Data Mining, Inference, and Prediction
ISBN 7-5053-9331-6

Ⅰ.统… Ⅱ.①黑… ②范… Ⅲ.统计学-教材 Ⅳ. C8

中国版本图书馆CIP数据核字（2003）第124311号

责任编辑：杜闽燕
印　　刷：北京兴华印刷厂
出版发行：电子工业出版社
　　北京市海淀区万寿路173信箱　邮编：100036
经　　销：各地新华书店
开　　本：787 × 1092　1/16　　印张：24.75　字数：634千字　彩插：22
印　　次：2004年1月第1次印刷
定　　价：45.00元

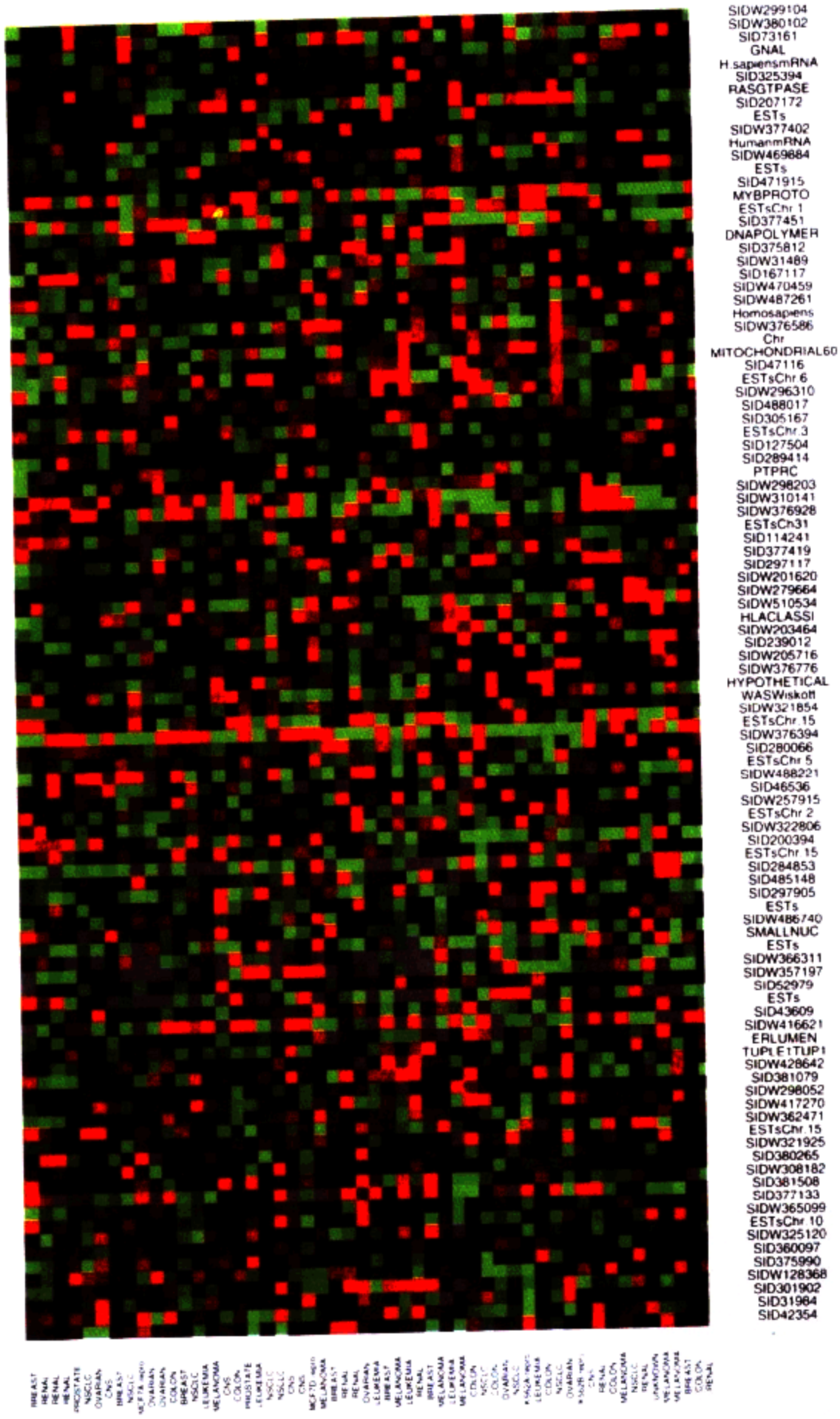


图 1.3

DNA 微阵列数据：人体瘤数据 6830 个基因（行）和 64 个样本（列）的表达水平矩阵。只显示 100 行的随机选样。显示的是热度图，从鲜绿（负，低显性）到鲜红（正，高显性）。遗漏的值为灰色。行和列以随机次序显示

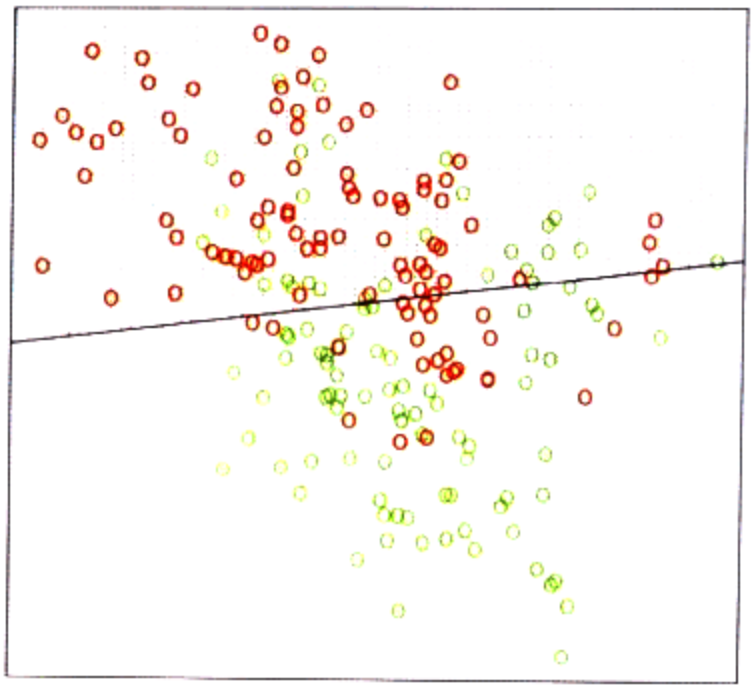


图 2.1

一个二维空间上的分类例子。类用二元变量编码（GREEN = 0，RED = 1），并且用线性回归拟合。直线是 $x^T\hat{\beta}=0.5$ 定义的判定边界。红色区域表示输入空间被分类为RED的部分，而绿色区域被分类为 GREEN

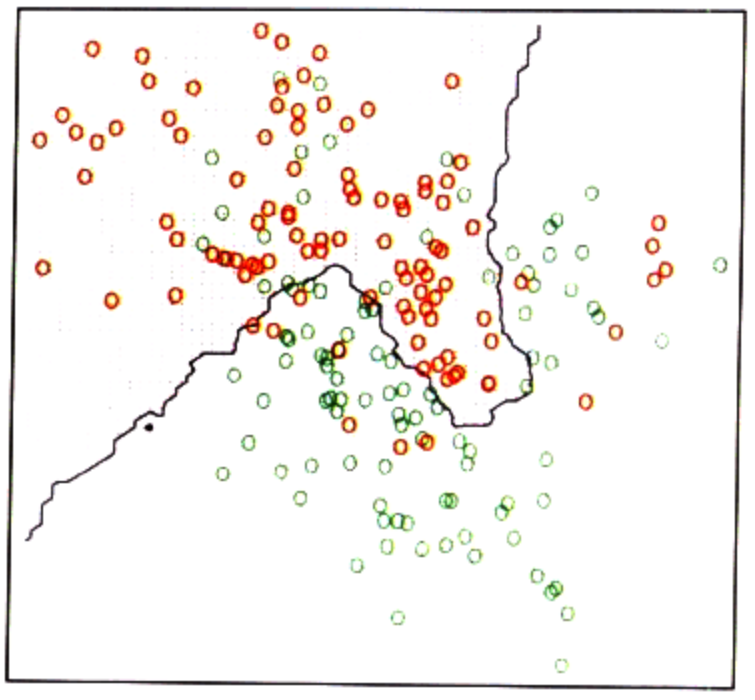


图 2.2

与图2.1相同的二维分类例子。类用二元变量编码（GREEN = 0，RED = 1），并用式（2.8）的 15- 最近邻平均拟合。因此，预测类用15- 最近邻的多数表决确定

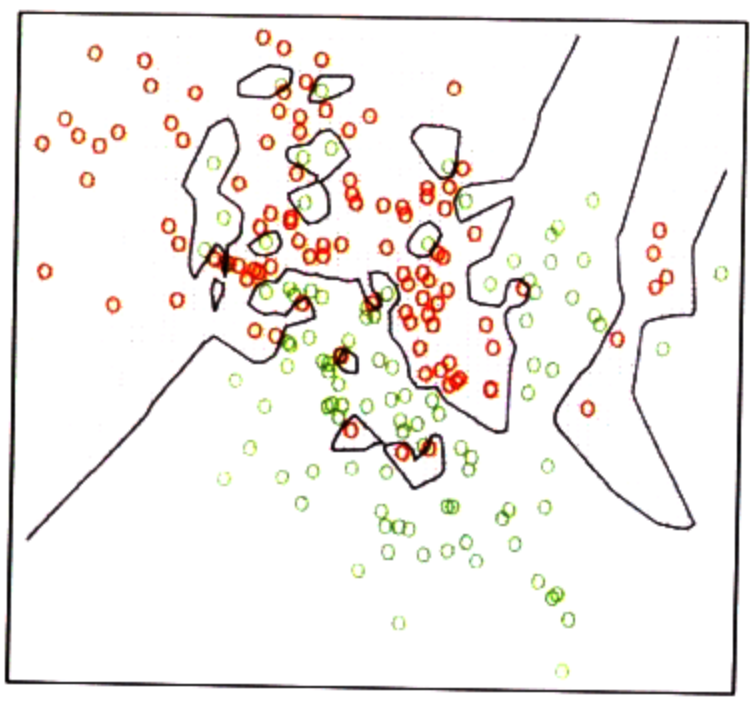


图 2.3

与图 2.1 相同的二维分类例子。类用二元变量编码（GREEN = 0，RED = 1），并用 1- 最近邻分类预测

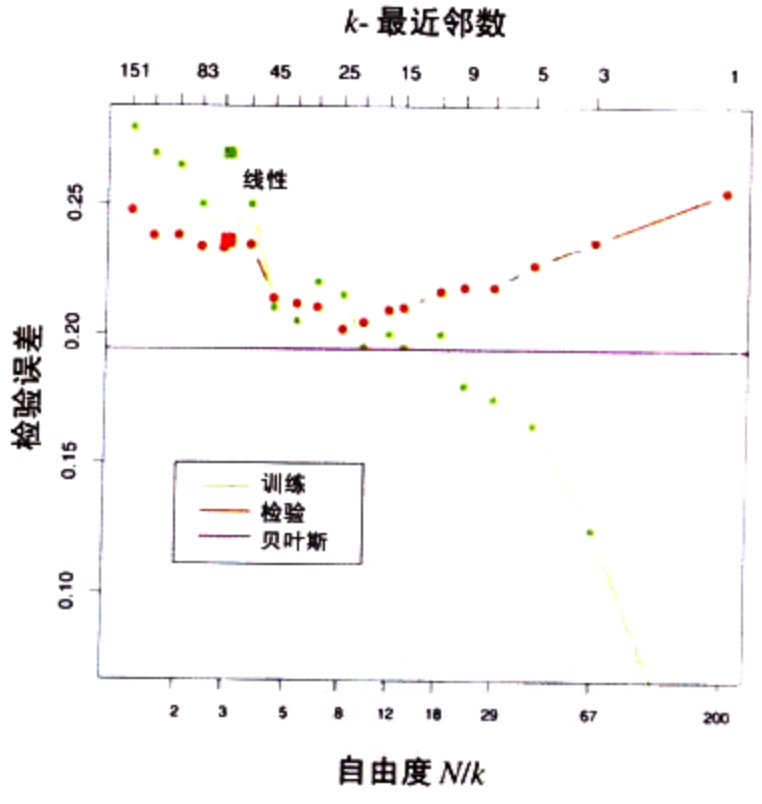


图 2.4

图2.1、图2.2和图2.3使用的模拟例子的误分类曲线。使用一个规模为 200 的训练样本和一个规模为 10 000 的检验样本。红色曲线是 k- 最近邻分类的检验误差，绿色曲线是训练误差。线性回归的结果是三自由度上较大的红色和绿色方块。紫色直线是最优的贝叶斯误差率

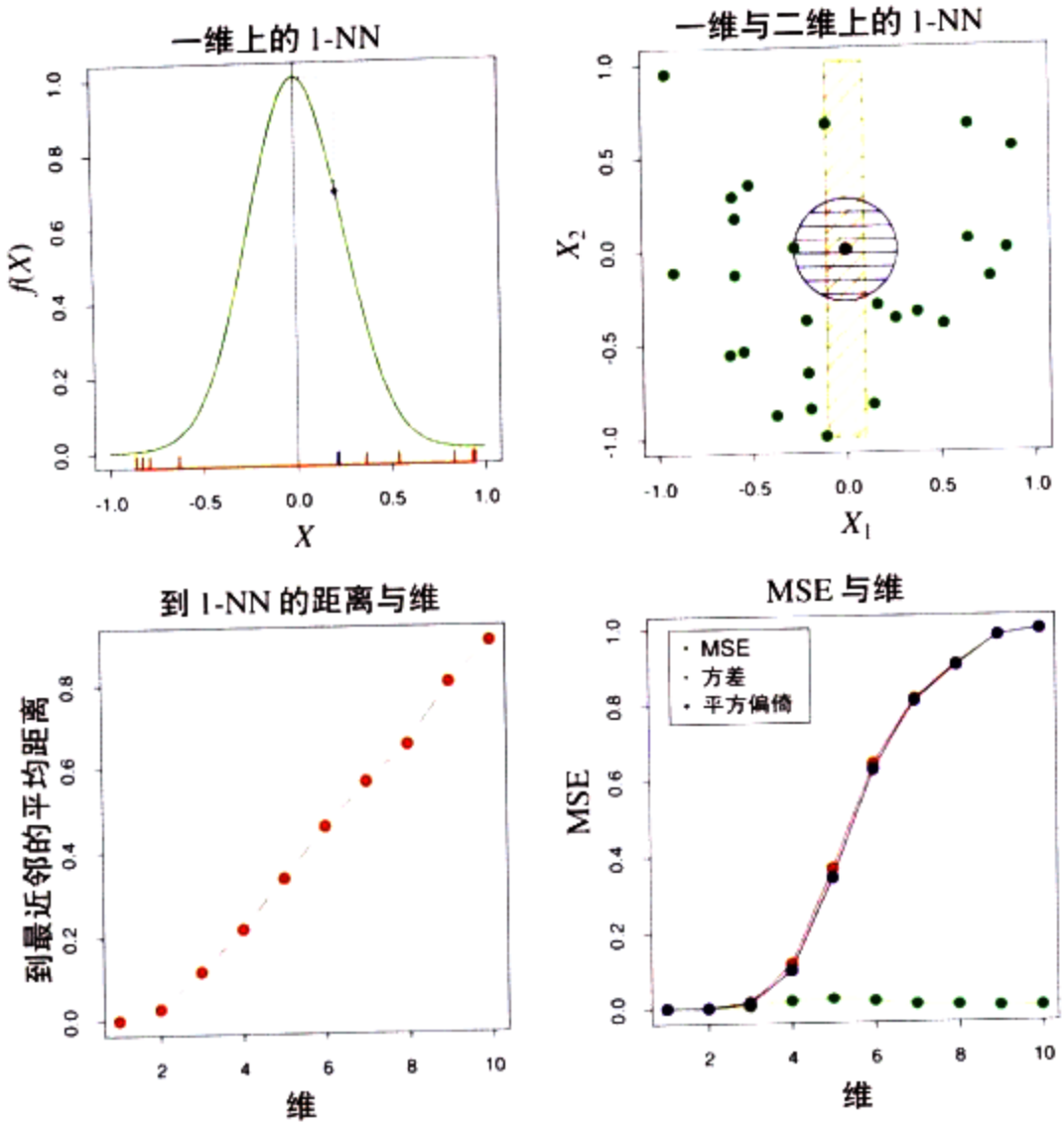


图 2.7

一个模拟例子，表明维灾难及其对MSE、偏倚和方差的影响。对于 $p=1, \cdots, 10$，输入特征值在$[-1, 1]^p$上均匀分布。左上角的图显示IR上的(无噪声)目标函数：$f(X)=e^{-8\|X\|^2}$，并图示1-最近邻对$f(0)$估计所产生的误差。训练点用蓝色粗体标记。右上角的图展示1-最近邻域的半径随维数p增加的原因。左下角的图展示1-最近邻域的平均半径。右下角的图显示作为维数p的函数，MSE、平方偏倚和方差的曲线

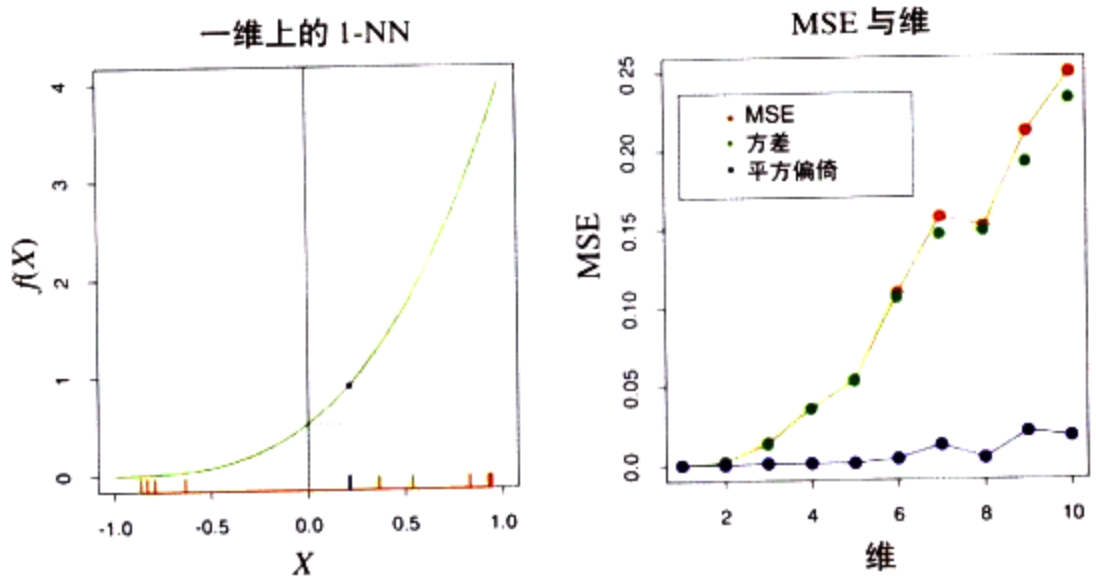


图 2.8

一个与图 2.7 具有相同设置的模拟例子。这里，除一个维为$f(X)=\frac{1}{2}(X_1+1)^3$外，函数为常数。方差占支配地位

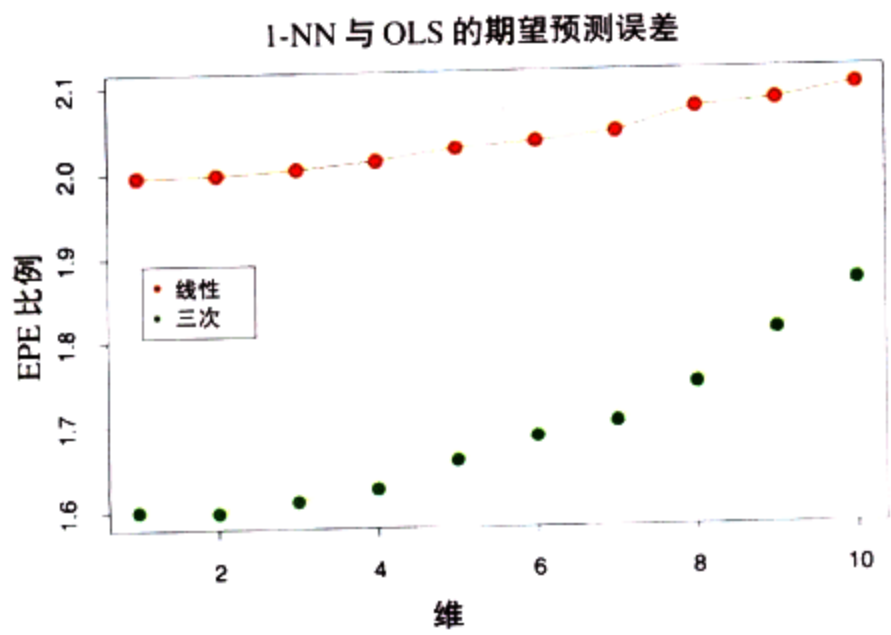


图 2.9

显示1-最近邻相对于最小二乘方关于模型$Y=f(X)+\varepsilon$的期望预测误差的曲线(在x_0=0)。对于红色曲线，$f(x)=x_1$；而对于绿色曲线，$f(x)=\frac{1}{2}(x_1+1)^3$

图 3.5

前列腺癌例子所有可能的子集模型。对每个子集容量，显示该容量的每个模型的残差平方和

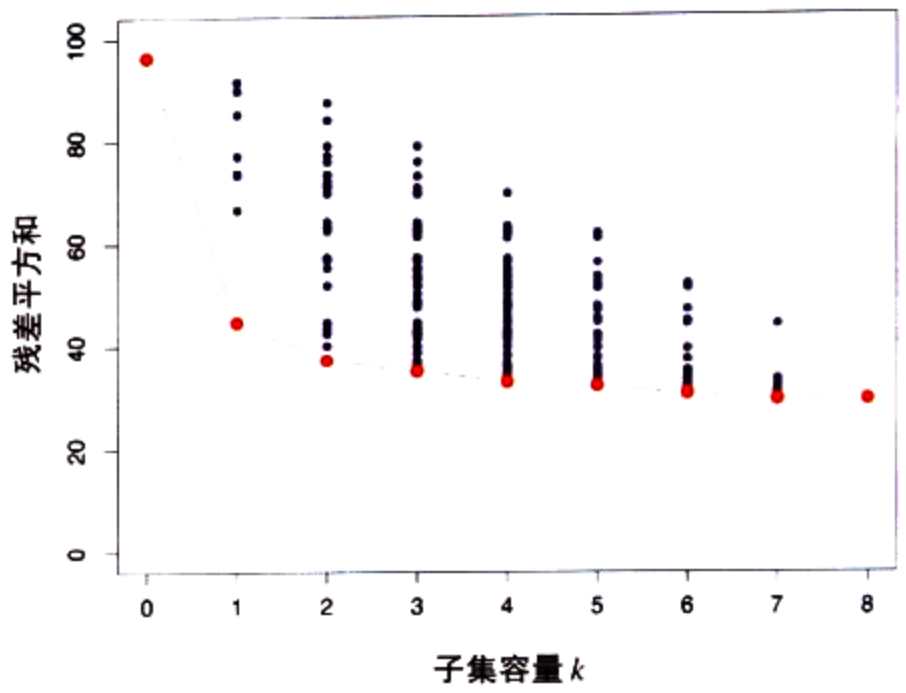
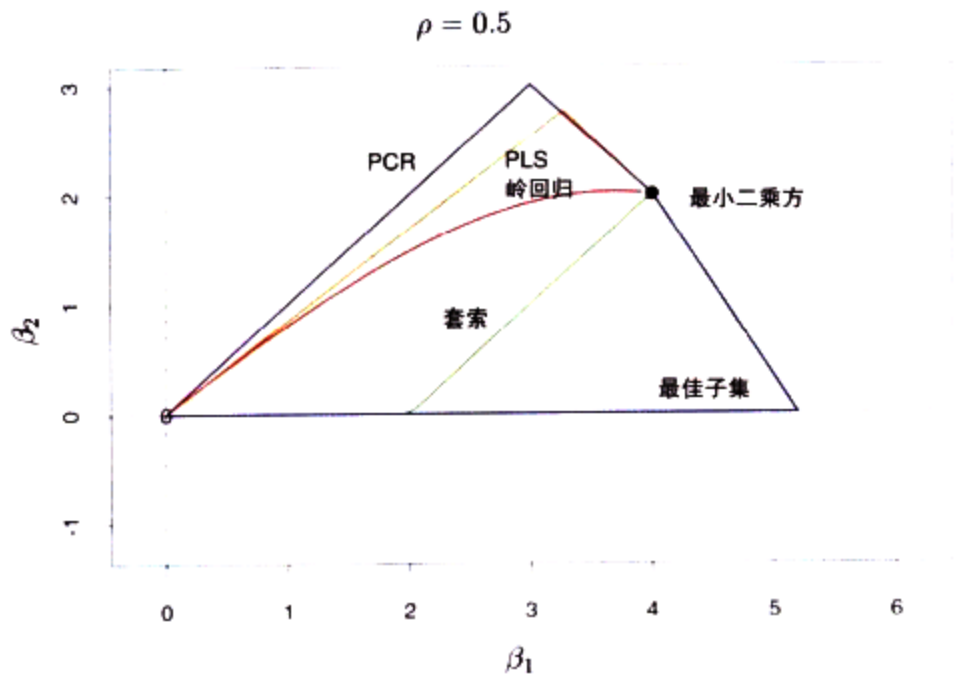
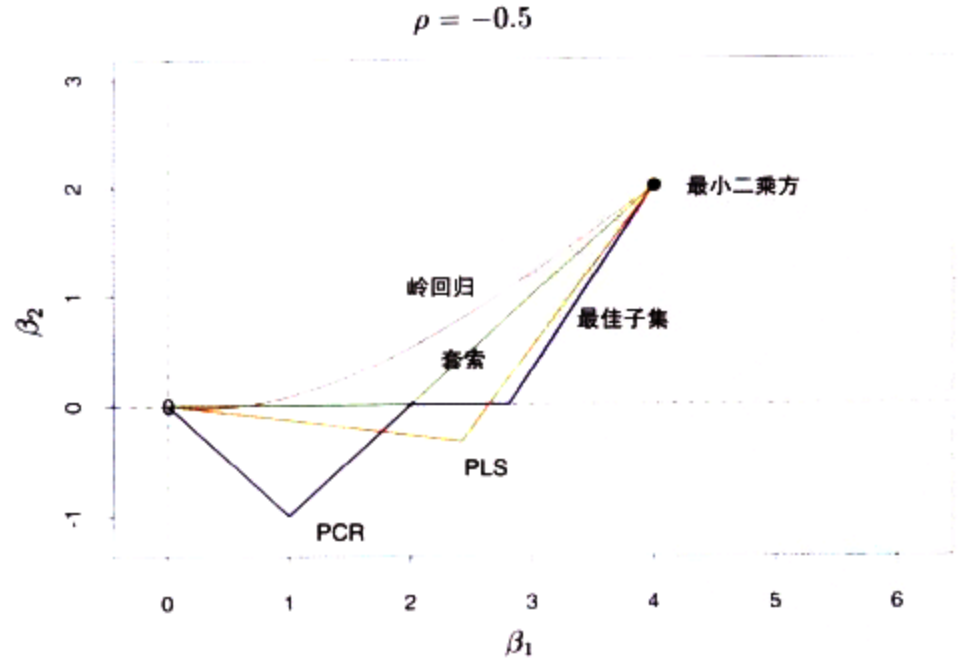


图 3.11

对于一个简单例子，不同方法的系数曲线图：两个具有相关度 ± 0.5 的输入，而实际的回归系数是 $\beta = (4, 2)$





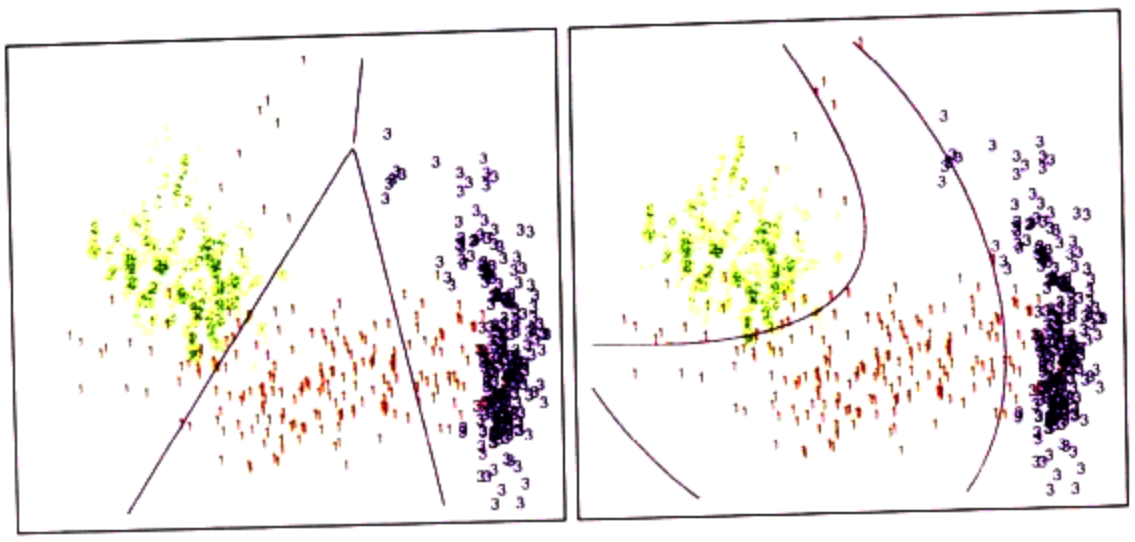


图 4.1

左图显示取自三个类的一些数据点，以及由线性判别分析找出的线性判定边界。右图显示二次判定边界。这些边界通过找出 5 维空间 $X_1, X_2, X_1X_2, X_1^2, X_2^2$ 中的线性边界得到。在该空间上的线性不等式是原空间中的二次不等式

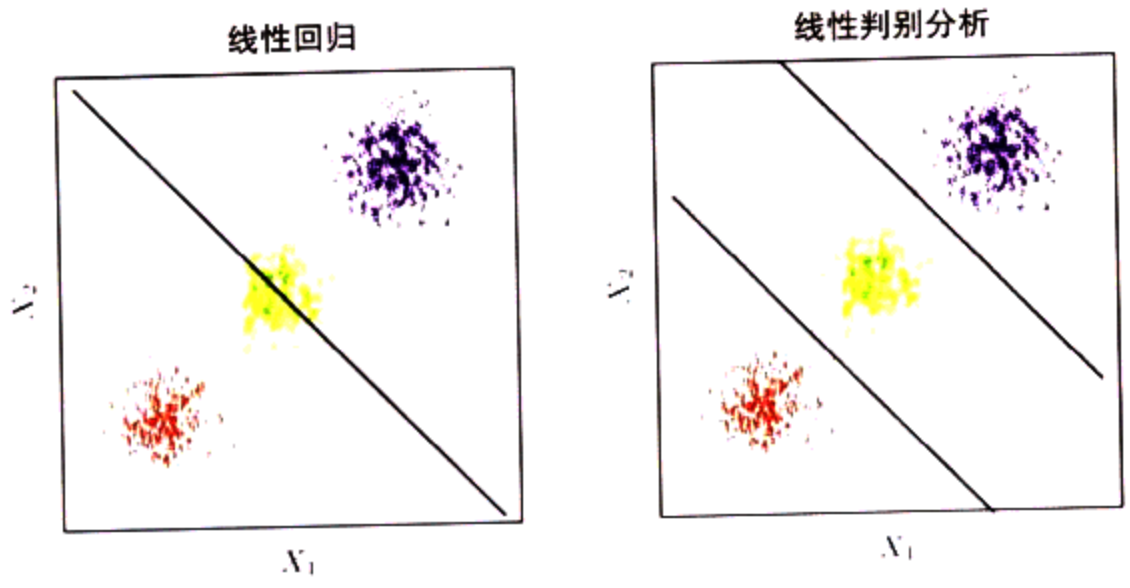


图 4.2

数据取自 IR^2 中的三个类，并容易被线性判定边界分开。右图显示被线性判别分析找到的边界。左图显示被指示响应变量的线性回归找出的边界。中间类完全被屏蔽（不占支配地位）

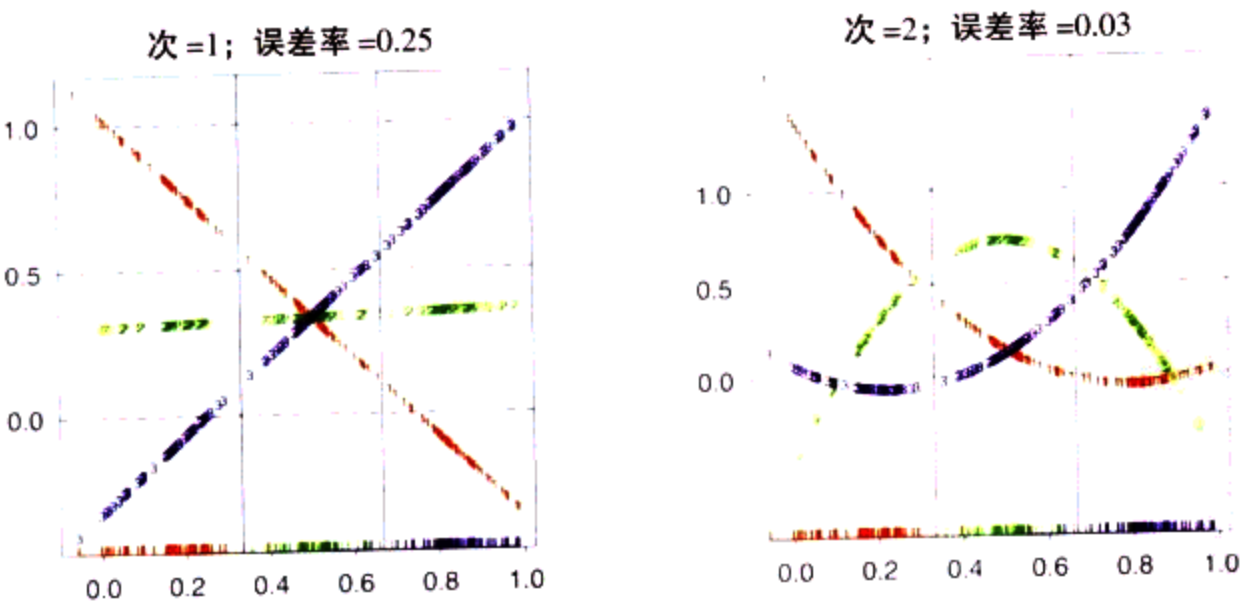


图 4.3

对于一个 3- 类问题，IR 上的线性回归的屏蔽作用。底部的底线图（rug plot）指示每个观测的位置和类隶属关系。每幅图上的三条曲线是 3- 类指示变量的拟合回归；例如，对于红色类，红色观测的 y_{red} 为 1，而绿色和蓝色观测的 y_{red} 为 0。每幅图的上方是训练误差率。对于该问题，贝叶斯误差率为 0.025，与 LDA 误差率一样

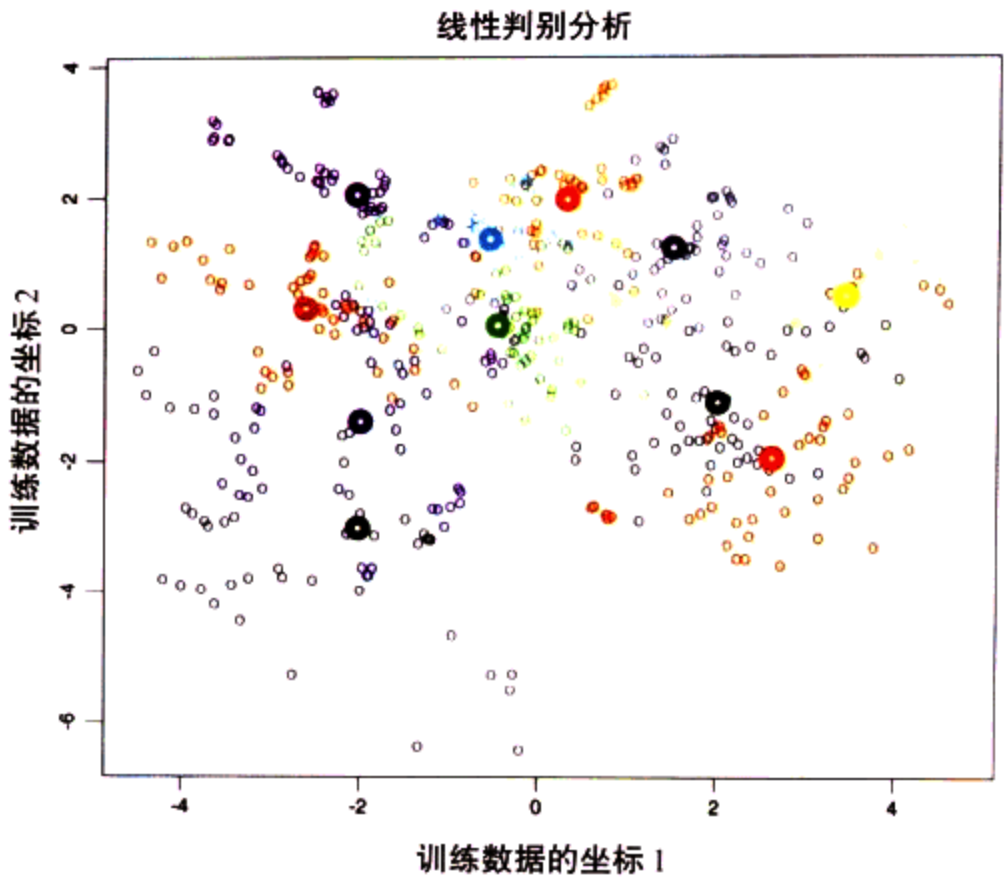


图 4.4

元音训练数据的二维图。有 11 个类，$X \in \mathrm{IR}^{10}$。这是 LDA 模型（见第 4.3.3 节）下的最佳视图。加重的圆是每个类的投影均值向量。类的重叠相当多

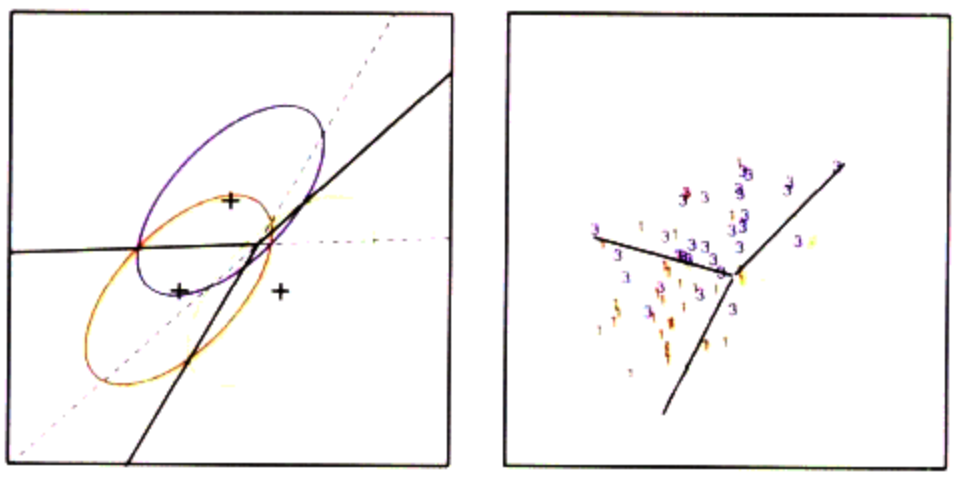


图 4.5

左图显示三个高斯分布，它们具有相同的协方差和不同的均值。图中包含的是每种情况围绕概率 95% 的常量密度围线。图中显示了每两个类之间的贝叶斯判定边界（虚线），而分离所有三个类的贝叶斯判定边界是粗实线（前者的子集）。在右图中，我们看到取自每个高斯分布的容量为 30 的样本，以及拟合的 LDA 判定边界

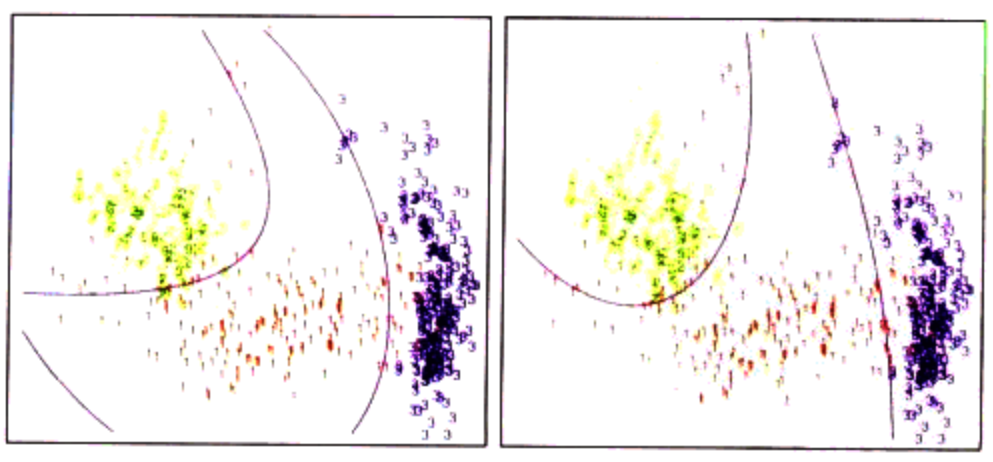


图 4.6

拟合二次边界的两种方法。对于图 4.1 的数据，左图显示二次判定边界（使用 5 维空间 $X_1, X_2, X_1X_2, X_1^2, X_2^2$ 上的 LDA 得到）。右图显示QDA发现的二次判定边界。差别很小，通常也是如此

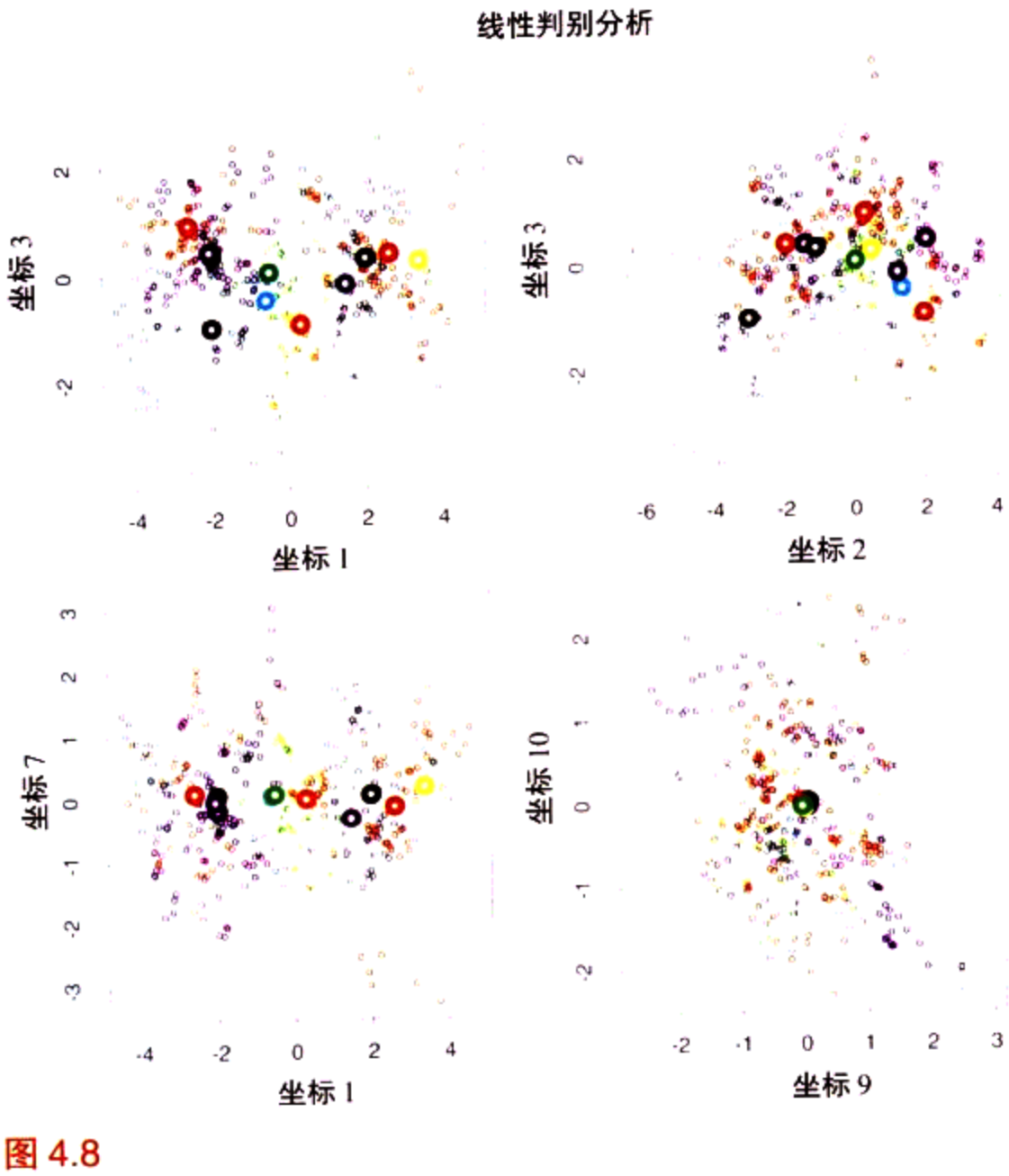


图 4.8

到标准变量对上的4个投影。注意，随标准变量的秩增加，形心变得集中。在右下图，它们似乎被叠加，并且类变得最乱

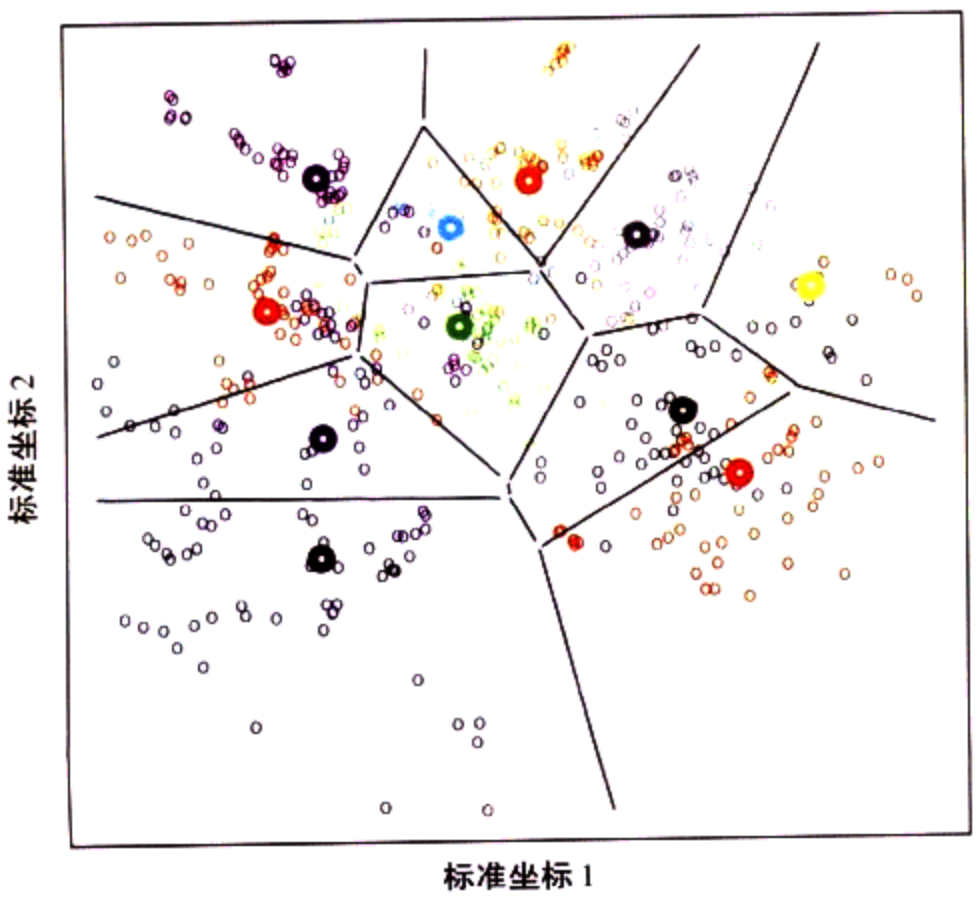


图 4.11

在前两个标准变量生成的二维子空间中元音训练数据的判定边界。注意，在任意较高维的子空间中，判定边界是较高维的仿射平面，不能用线表示

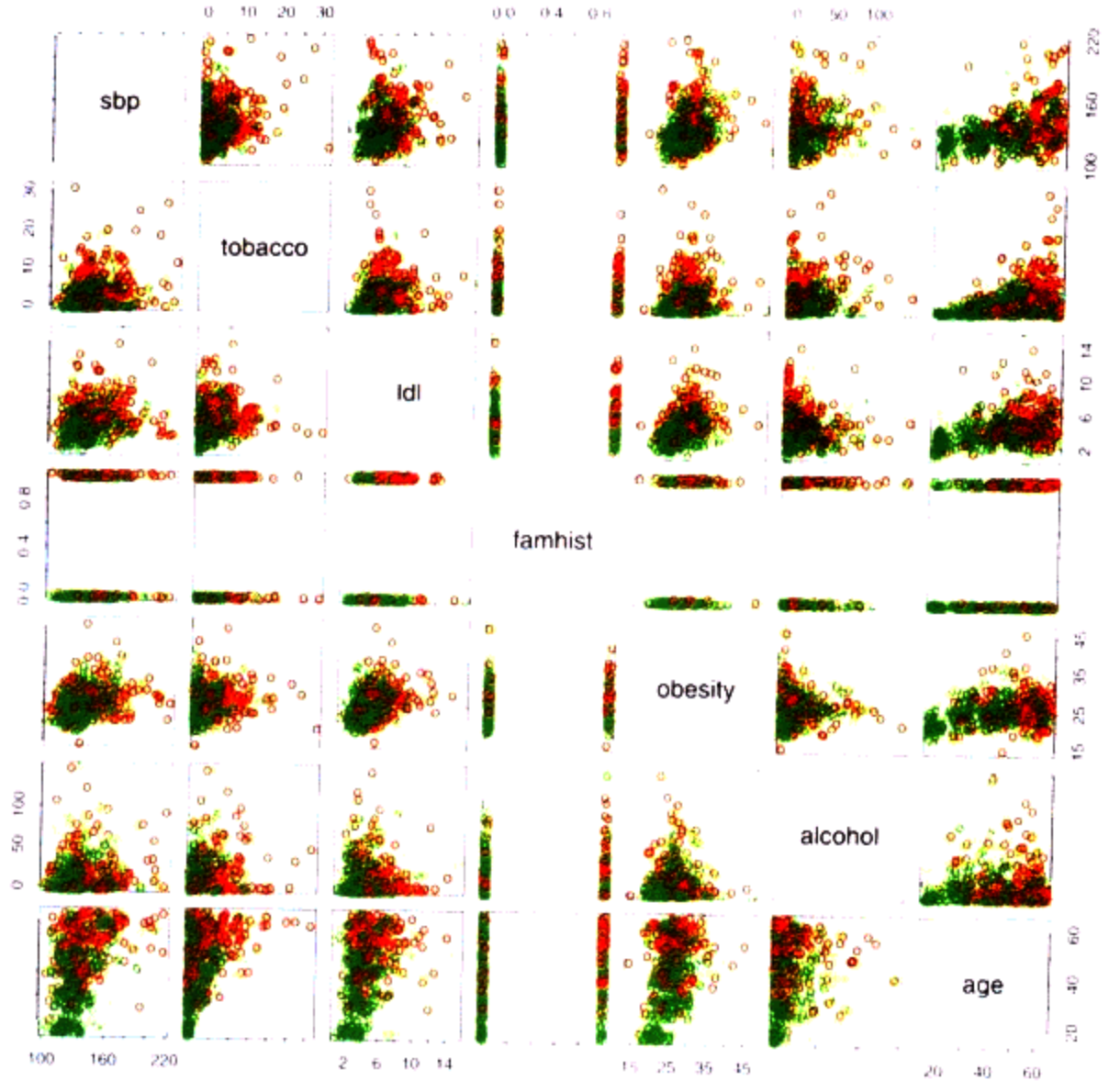


图 4.12

南非心脏病数据的散点图。每幅图显示一对风险因素，并且病例和控制用颜色编码（红色是病例）。心脏病家族史变量（famhist）是二元的（yes 或 no）

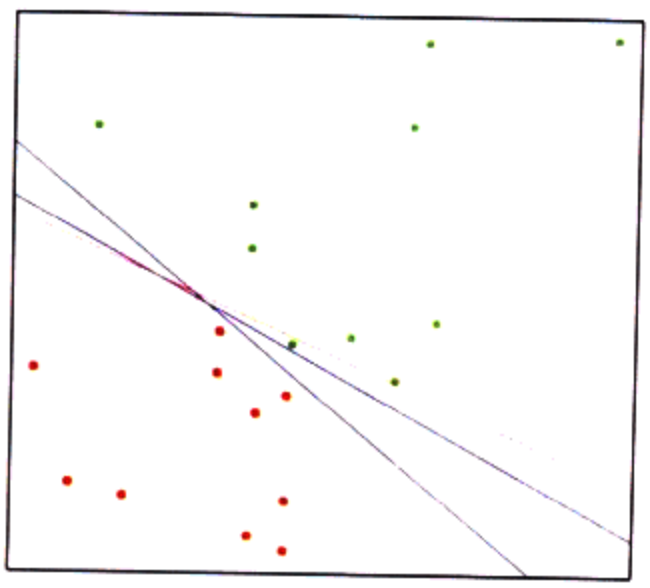


图 4.13

一个小例子，包含两个可被超平面分隔的类。橙色线是最小二乘方解，它将一个训练数据误分类。图中还显示了两个蓝色分隔超平面，它们被以不同的随机初始化的感知器学习算法找出

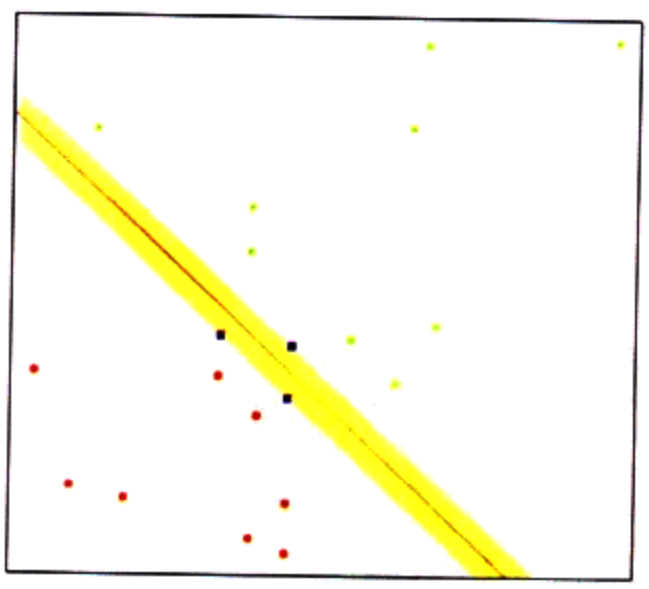


图 4.15

与图4.13相同的数据。阴影区域描述分离两个类的最大边缘。有三个支撑点，它们在边缘的边界上，而最佳分离超平面（蓝线）将隔离带一分为二。图中还显示了逻辑斯缔回归找出的边界（红线），它非常接近最佳分离超平面（见第12.3.3节）

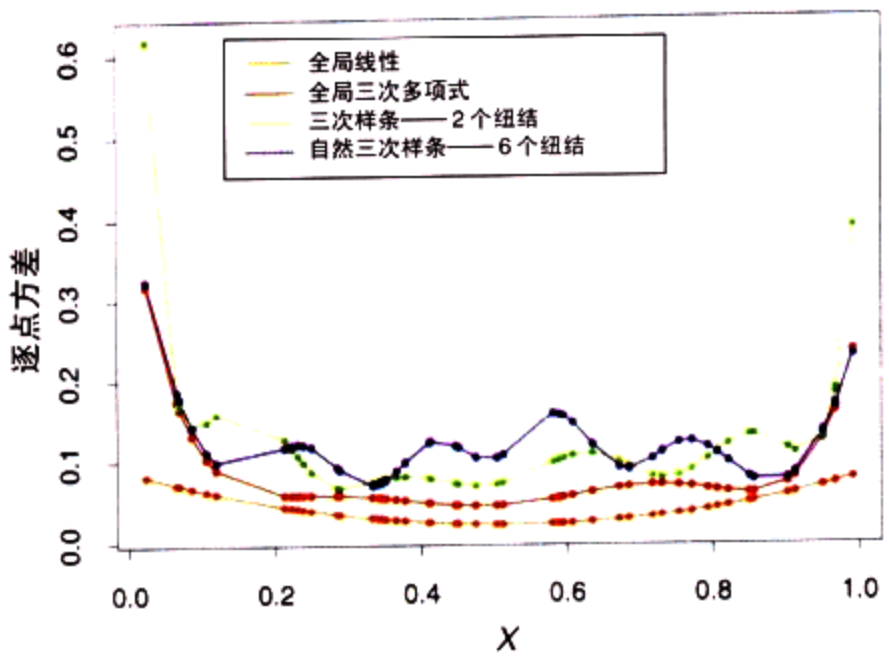
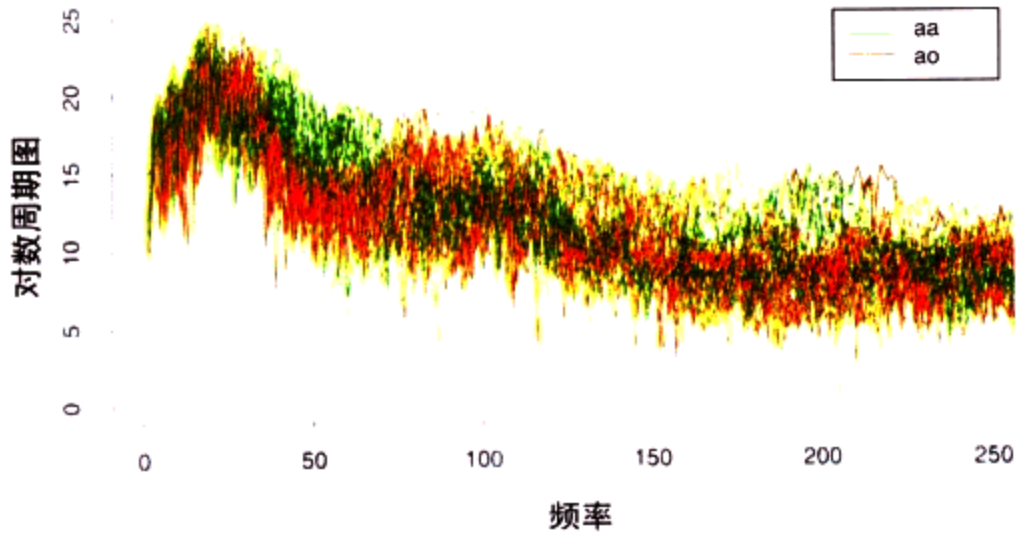


图 5.3

4个不同模型的逐点方差曲线。X包含50个点，随机地取自$U[0, 1]$，一个假定的误差模型具有常数方差。线性和二次多项式拟合分别具有2个和4个自由度，而三次样条和自然三次样条具有6个自由度。三次样条在0.33和0.66有两个纽结，而自然样条在0.1和0.9具有边界纽结，并且有4个内部纽结均匀地散布在它们中间



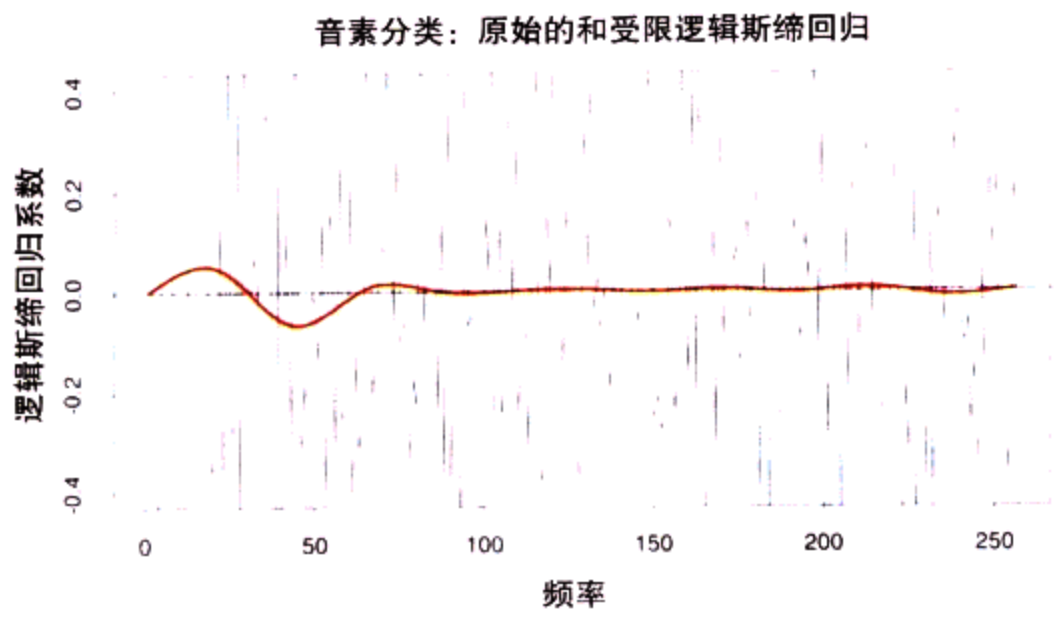
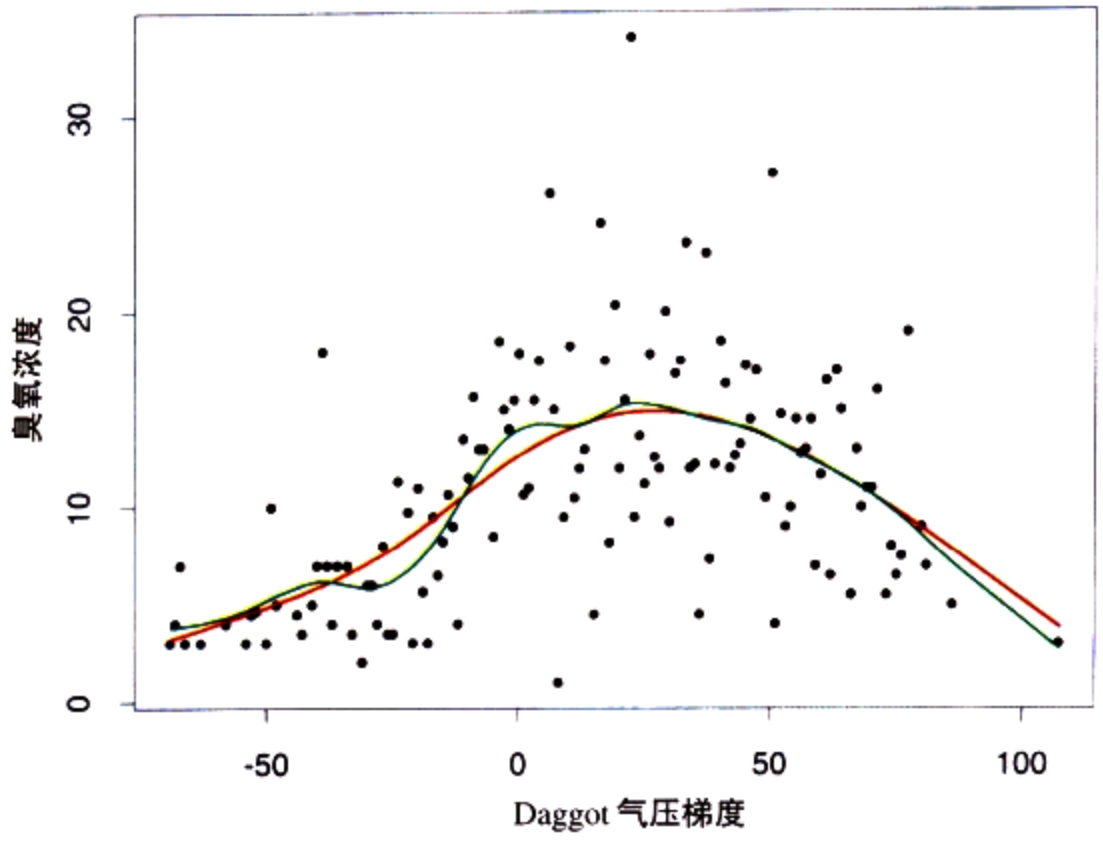


图 5.5

上图显示对数周期图；对于15个例子，每个音素“aa”和“ao”从695个“aa”和1022个“ao”中选样，对数周期图作为频率的函数显示。每个对数周期在256个均匀分布的频率点上测量。下面的图显示逻辑斯缔回归系数（作为频率的函数），使用256个对数周期图作为输入值，通过极大似然拟合数据。在红色曲线上，限制系数是光滑的，而在锯齿状灰色曲线上没有限制



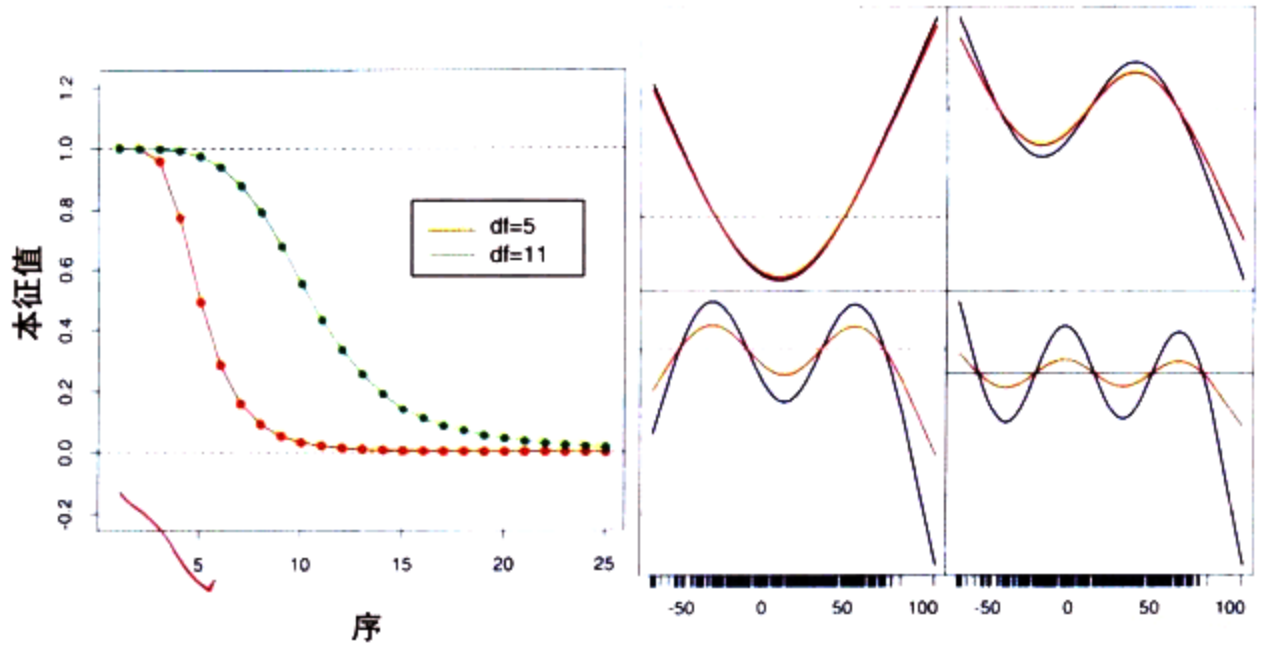


图 5.7

[上图]臭氧浓度作为Daggot气压梯度的函数的光滑样条拟合。两个拟合对应于光滑参数的不同值。光滑参数的选取是为了得到5个和15个由 $df_\lambda = \text{trace}(\mathbf{S}_\lambda)$定义的有效自由度。[左下图]两个光滑样条矩阵的前25个本征值。前两个本征值恰为1，并且所有的本征值都大于或等于零。[右下图]样条光滑子矩阵的第三个和第六个本征向量。在每条曲线中，$\mathbf{u}_k$ 都对照 $\mathbf{x}$ 绘制，并因此视为 x 的函数。图底部的底线指示数据点的出现。阻尼函数表示这些函数的光滑版本（使用5df光滑子）

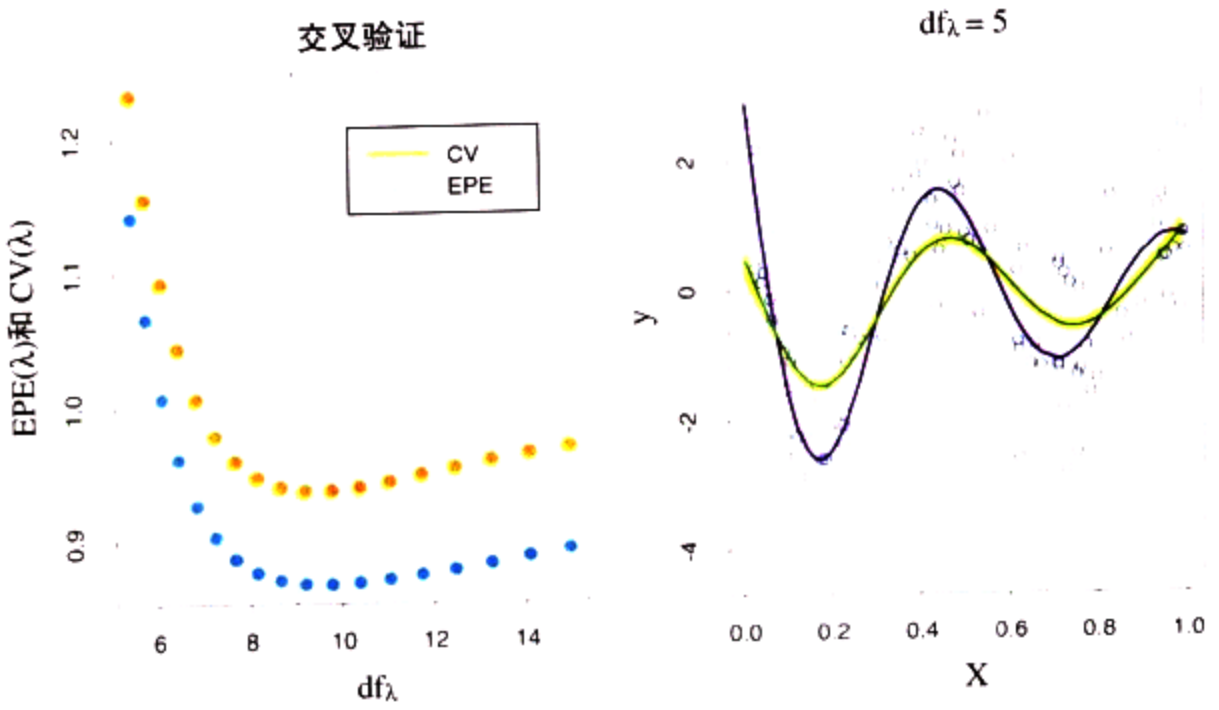


图 5.9

非线性加法误差模型（5.22）的实现，左上图显示其EPE(λ)和CV(λ)曲线。其他图对于不同的df_λ，显示数据、真实函数（紫色）和拟合曲线（绿色），其中黄色阴影是拟合曲线 ± 2 倍标准误差频带

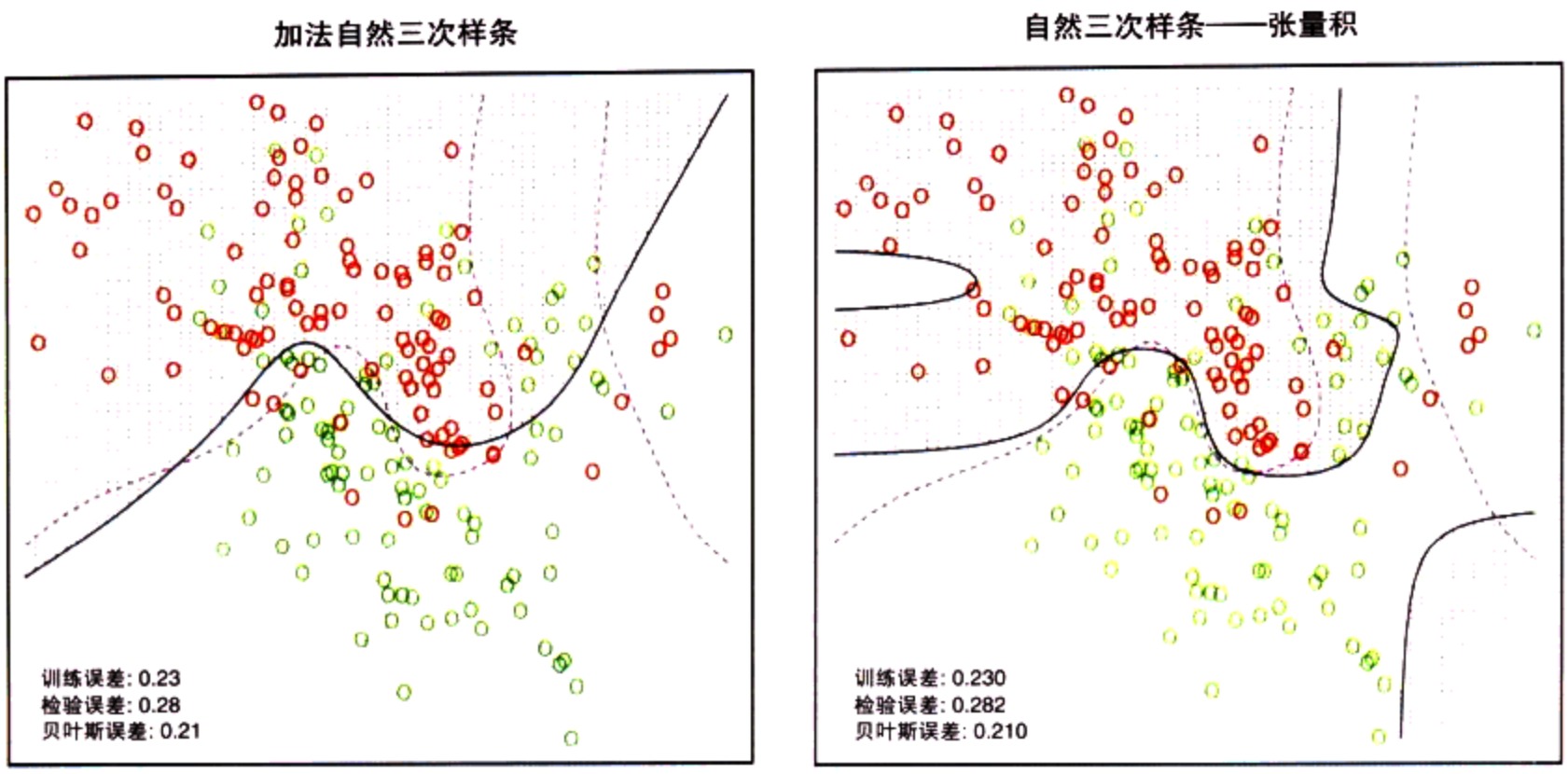


图 5.11

图 2.1 的模拟例子。左图显示加法逻辑斯缔回归模型的判定边界，在两个坐标上都使用自然样条（全部 df = 1 + (4 − 1) + (4 − 1) = 7）。右图显示在每个坐标上使用自然样条基张量积的结果（全部 df = 4 × 4 = 16）。紫色虚线边界是该问题的贝叶斯判定边界

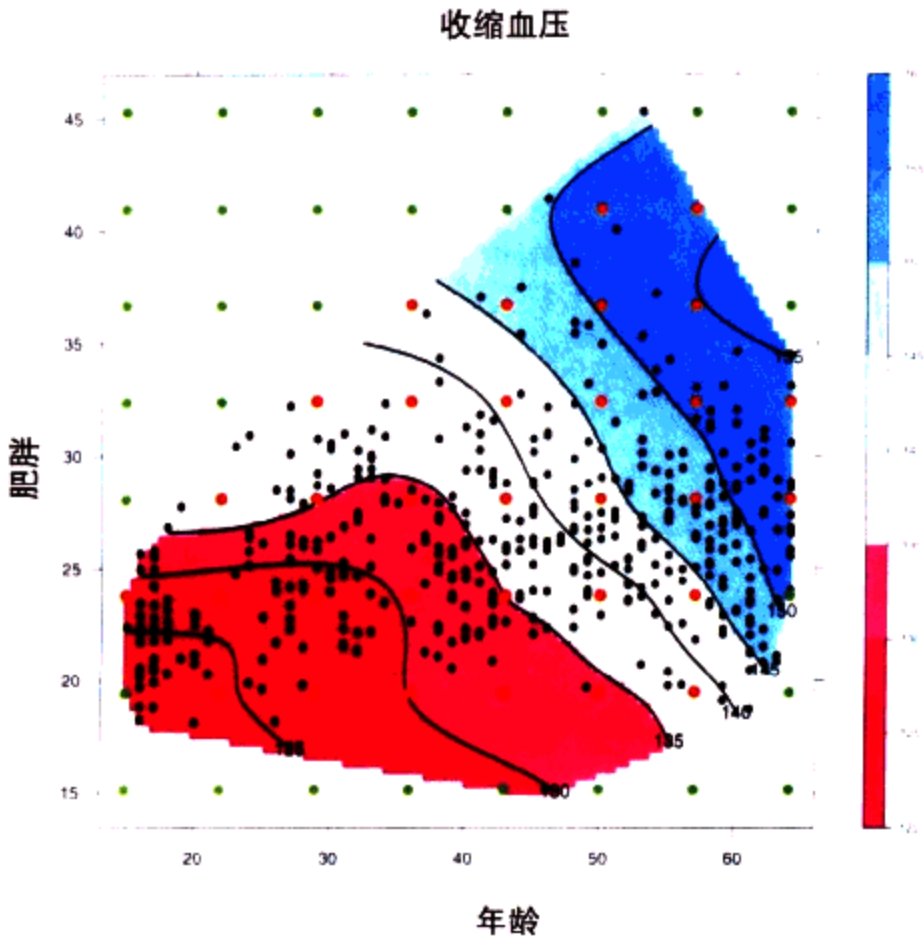


图 5.12

用围线图显示薄板样条拟合心脏病数据。响应是收缩血压（sbp），它作为年龄（age）和肥胖（obesity）的函数建模。图中标定有数据点以及用做纽结的点的格。谨慎使用来自数据凸包之内（红色）格的纽结，并忽略数据凸包之外（绿色）格的纽结

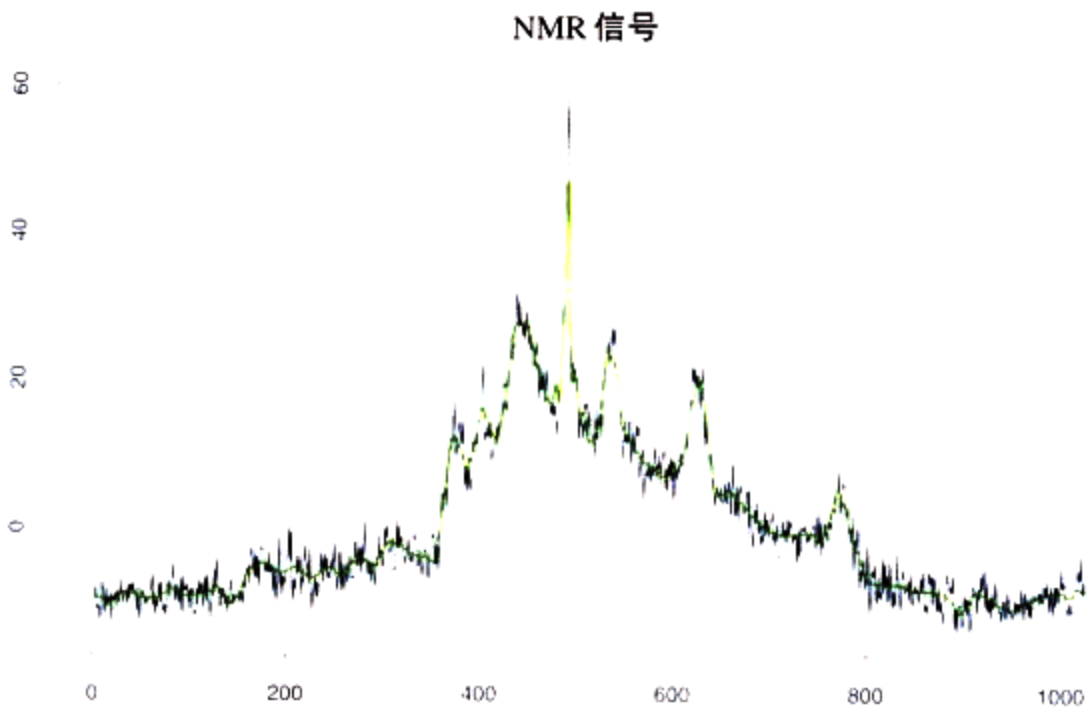
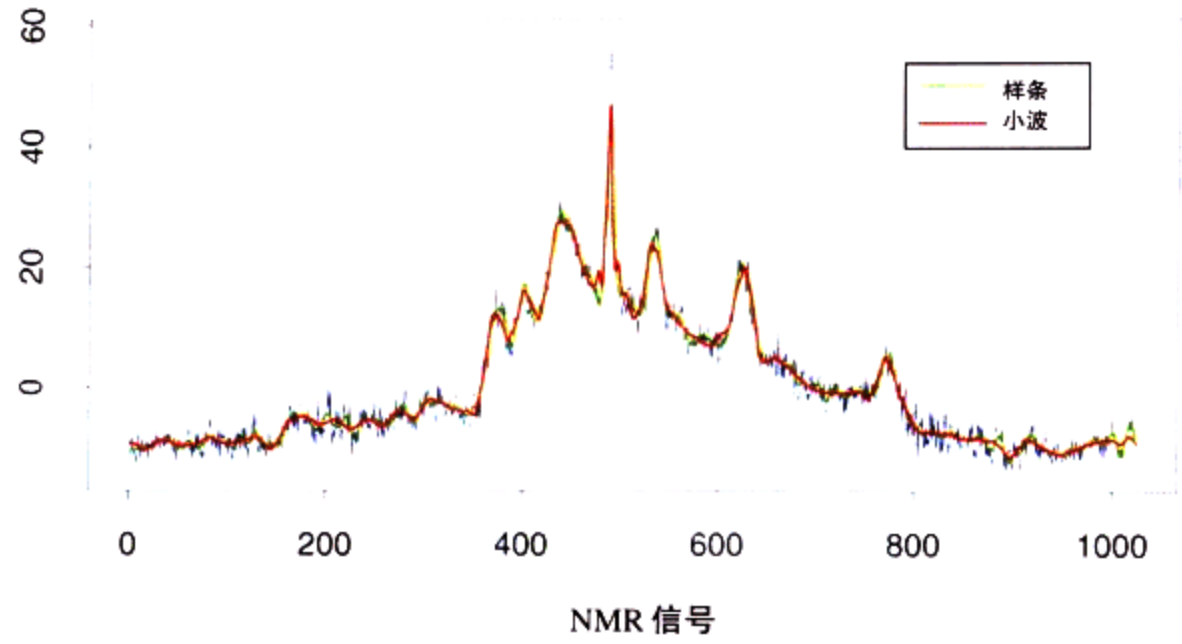


图 5.14

上图显示一个NMR信号，小波收缩版本以绿色叠加。左下图表示原信号的小波变换，使用symmlet-8基函数，取到V_4。每个系数用垂直条的高度（正的或负的）表示。右下图表示使用S-PLUS中的waveshrink函数收缩后的小波系数。waveshrink函数实现了Donoho和Johnstone的小波自适应SureShrink方法



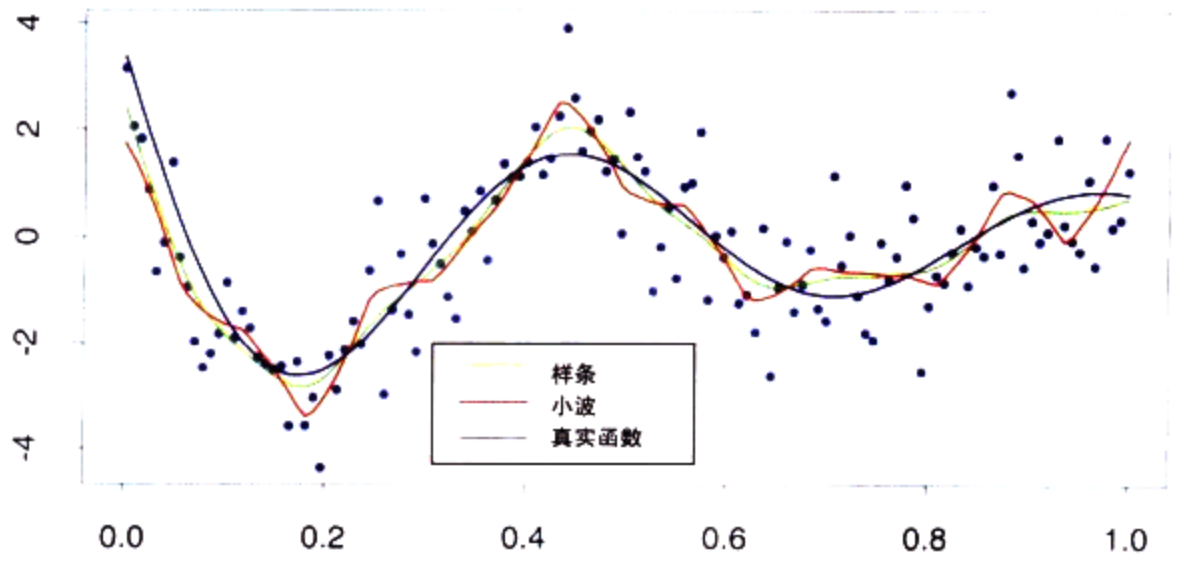


图 5.16

小波光滑与光滑样条比较的两个例子。每幅图给出SURE收缩小波拟合与交叉验证光滑样条拟合的比较

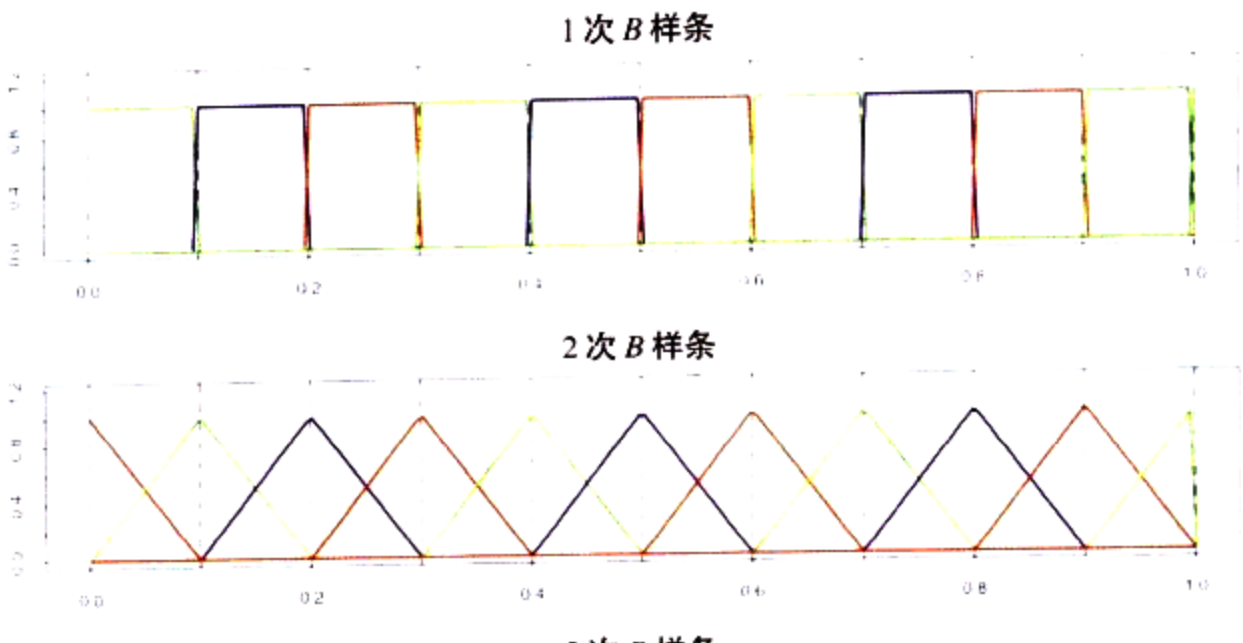


图 5.17

1次至4次 B 样条的序列，其中10个纽结均匀地分布在0到1之间。B 样条具有局部支集；它们在被 $M+1$ 个纽结生成的区间上非零

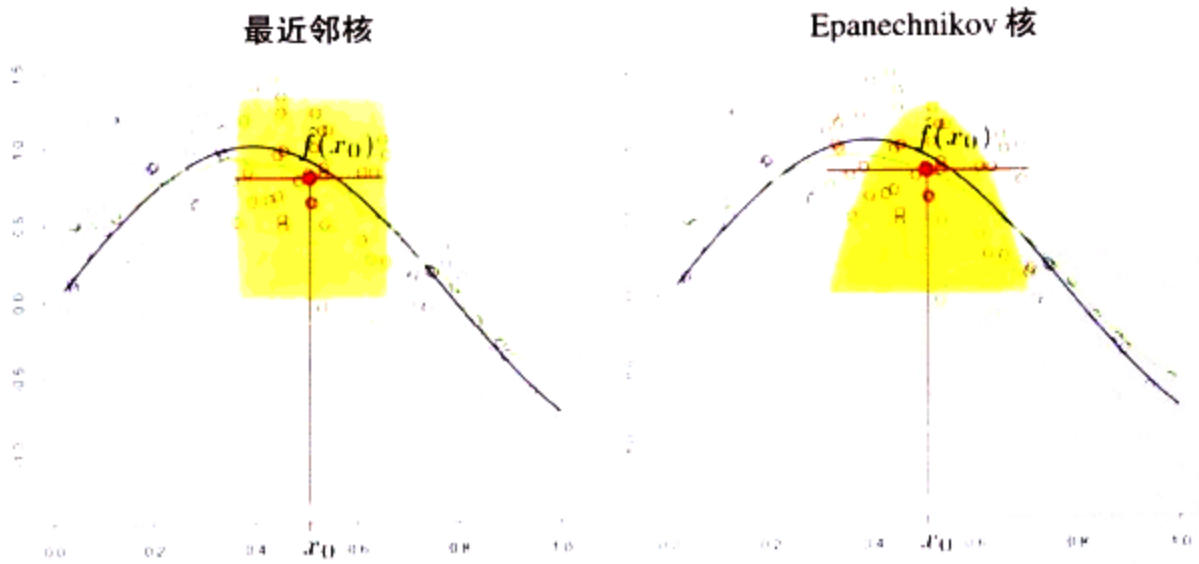


图 6.1

每幅图中的100个 x_i, y_i 对由具有高斯误差 $Y = \sin(4X) + \varepsilon$, $X \sim U[0,1]$, $\varepsilon \sim N(0, 1/3)$ 的蓝色曲线随机产生。在左图中，绿色曲线是30-最近邻移动均值（running-mean）光滑的结果。红色点是被拟合的常数 $\hat{f}(x_0)$，而橘黄色阴影圆指示那些对 x_0 上的拟合有贡献的观测。实心橘黄色区域指示赋予观测的权。在右图中，绿色曲线是核加权平均，使用（半个）窗口宽度为 $\lambda = 0.2$ 的 Epanechnikov 核

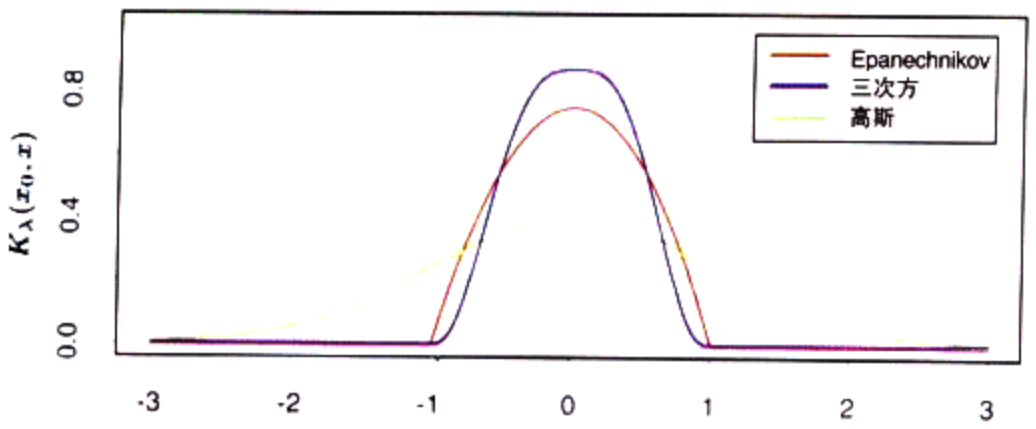


图 6.2

三种流行的局部光滑核的比较。每种都被调整得使积分为 1。三次方核是紧致的，并在其支集的边界上具有二阶连续导数，而 Epanechnikov 核没有。高斯核是连续可微的，但具有无限支集

图 6.3

局部加权平均在定义域边界上或接近定义域边界处存在偏倚问题。这里，真实函数是接近线性的，但是邻域中的大部分观测具有比目标点高的均值，因此尽管加权，它们的均值依然偏高。通过拟合局部加权线性回归（右图），该偏倚移至一阶

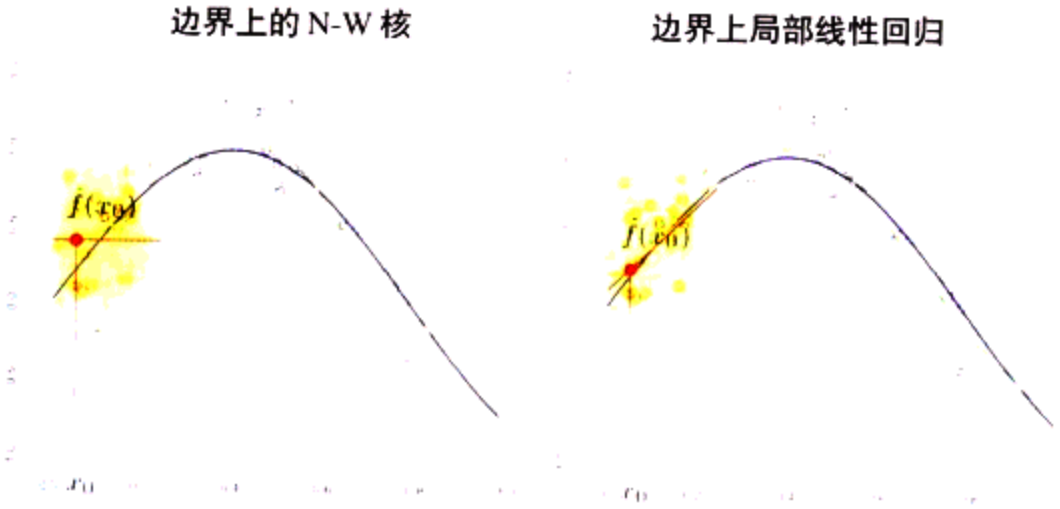


图 6.4

绿色点显示局部回归的等价核 $l_i(x_0)$。这些是 $\hat{f}(x_0)=\sum_{i=1}^{N} l_i(x_0)y_i$ 中的权，参照对应的 x_i 绘出。为了显示，这些已经过缩放，因为事实上它们的和为 1。由于橘黄色阴影区域是（重新缩放的）Nadaraya-Watson 局部平均的等价核，我们看到局部回归如何自动修改加权核，以校正由于光滑窗口中的不对称性而产生的偏倚

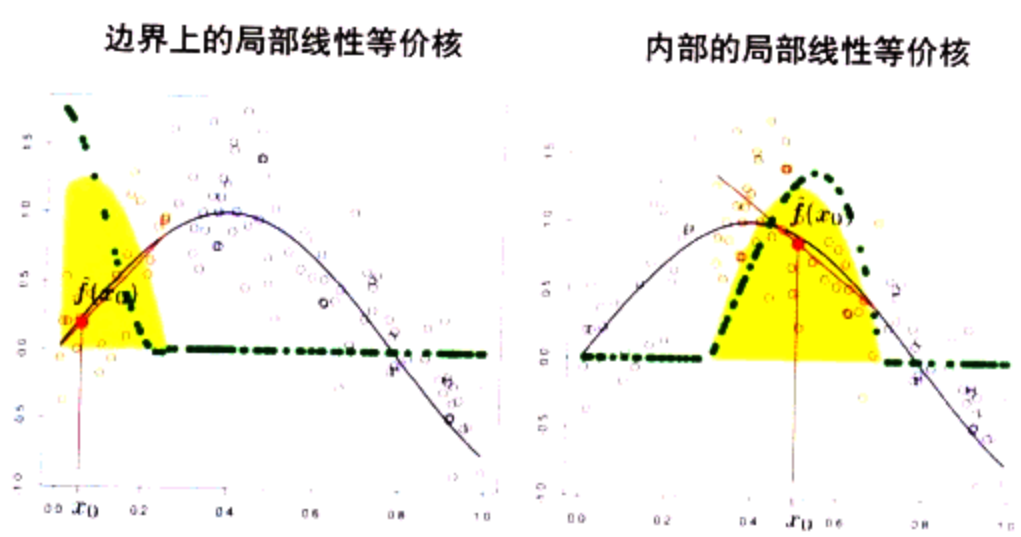
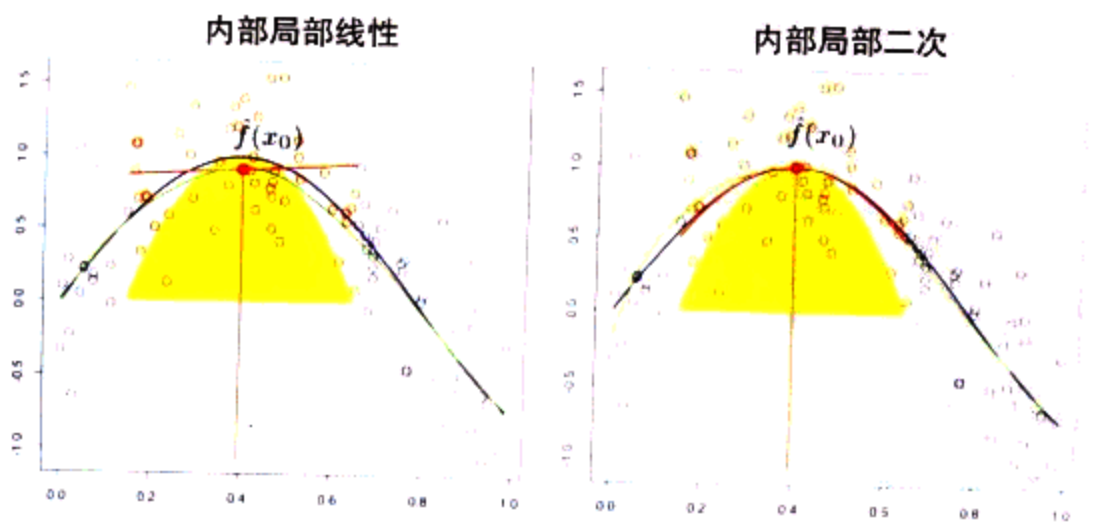


图 6.5

局部线性拟合在真实函数的弯曲部位表现出是有偏的。局部二次拟合趋向于消除这种偏倚



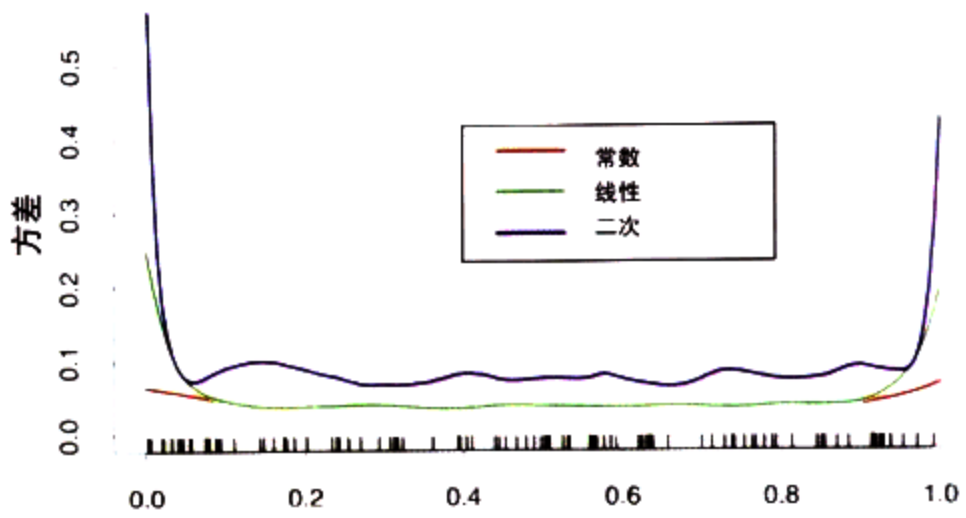


图 6.6

对于度量宽度（$\lambda = 0.2$）三次方核，局部常数、线性和二次回归的方差函数 $\| l(x) \|^2$

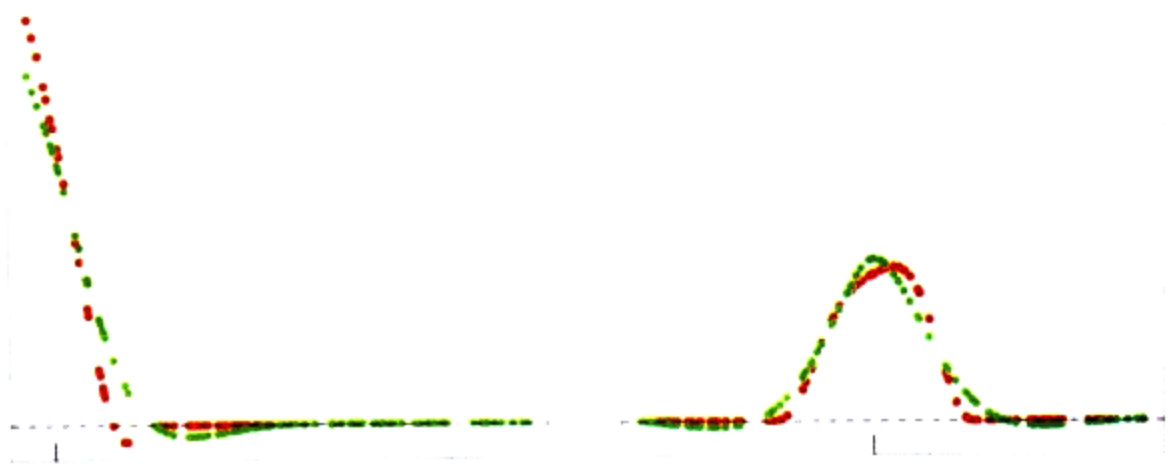


图 6.7

局部线性回归光滑法（三次方核，红色）和光滑样条（绿色）的等价核，具有匹配的自由度。竖直的短线指示目标点

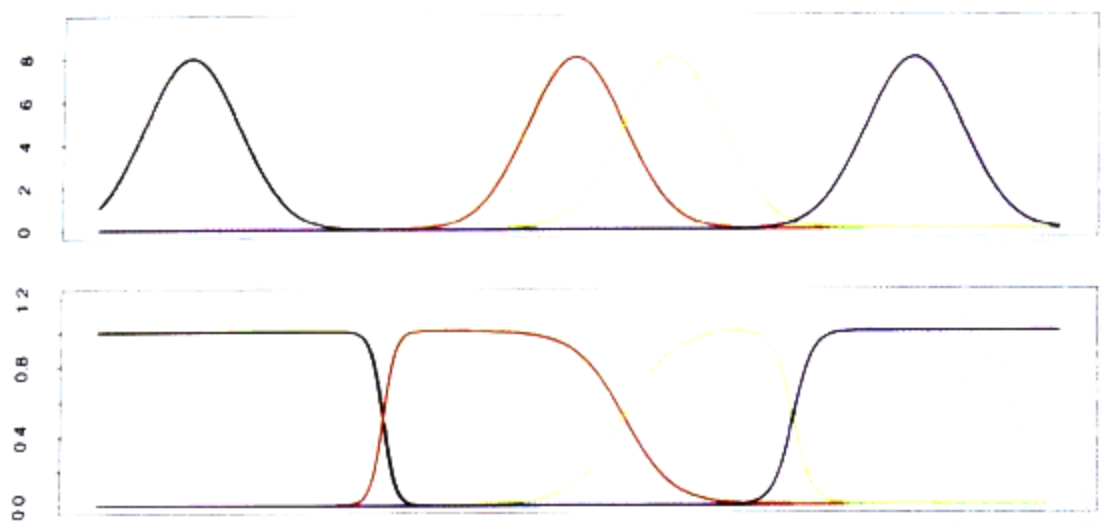


图 6.16

IR上具有固定宽度的高斯径向基函数可能产生洞（上图）。重新对高斯径向基函数标准化避免了该问题，并产生在某些方面类似于 B 样条的基函数

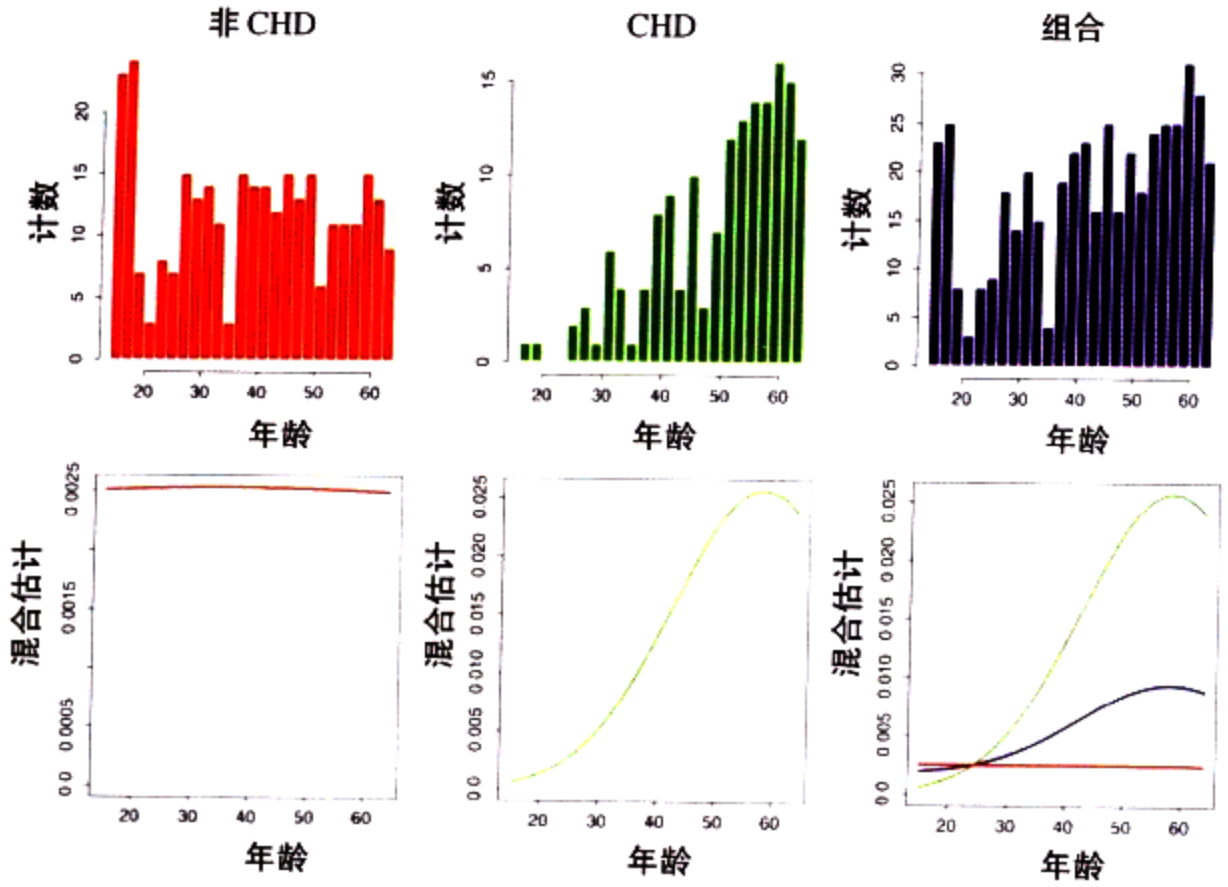


图 6.17

混合模型在心脏病风险因素研究中的应用。上行：分别是关于非 CHD 和 CHD 群的年龄(Age)的直方图和组合的年龄直方图。下行：高斯混合模型的估计支密度（左，中）；右下：估计支密度（绿色和红色），以及估计的混合密度（蓝色）。红色密度具有很大的标准差，并近似于均匀密度

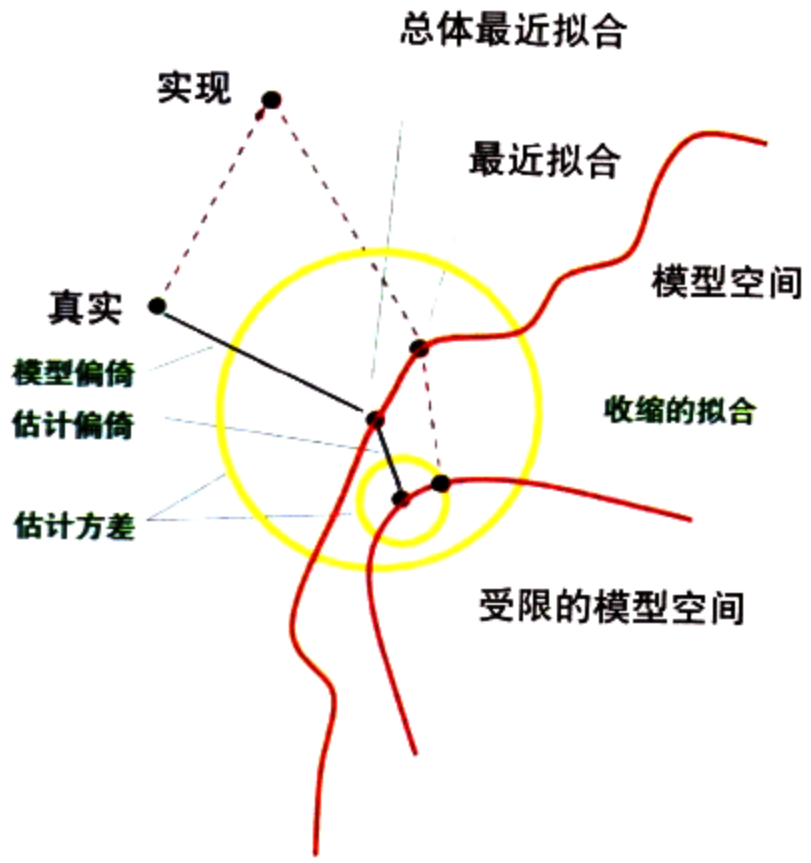


图 7.2

偏倚和方差变化示意图。模型空间是来自模型的所有可能预测的集合，“最近拟合”用黑点表示并加以标记。图中显示了与真实值的模型偏倚以及方差，由以标记为“总体最近拟合”的黑点为中心的黄色大圆指出。图中还显示了收缩或正则化拟合，它有额外的估计偏倚，但由于方差的减小，它具有较小的预测误差

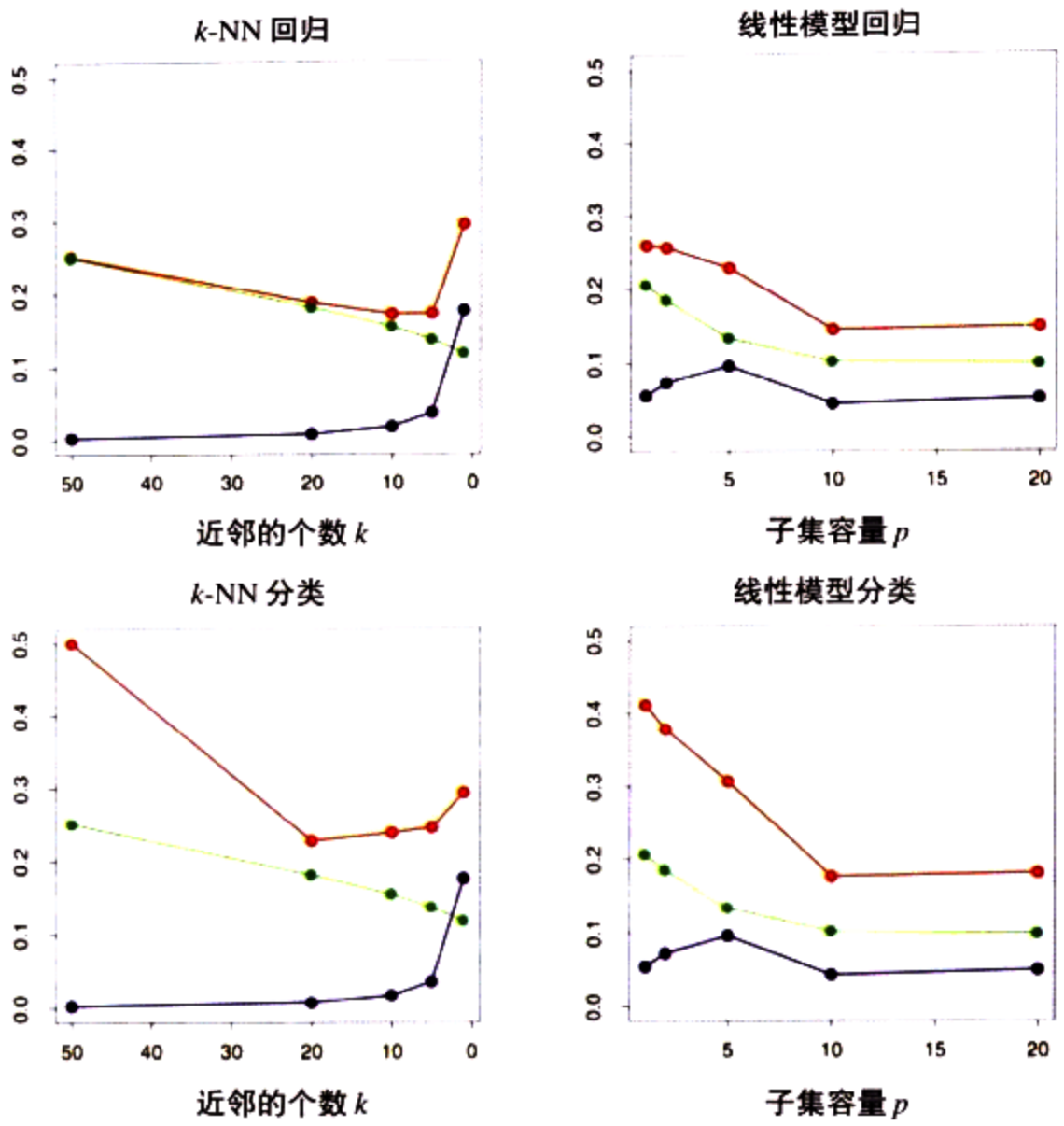


图 7.3

一个模拟例子的预测误差（红色）、平方偏倚（绿色）和方差（蓝色）。上两幅图是具有平方误差损失的回归，下两幅图是具有 0-1 损失的分类。模型是 k-最近邻（左）和容量为 p 的最佳子集回归（右）。方差和偏倚曲线在回归和分类中是相同的，但预测误差曲线不同

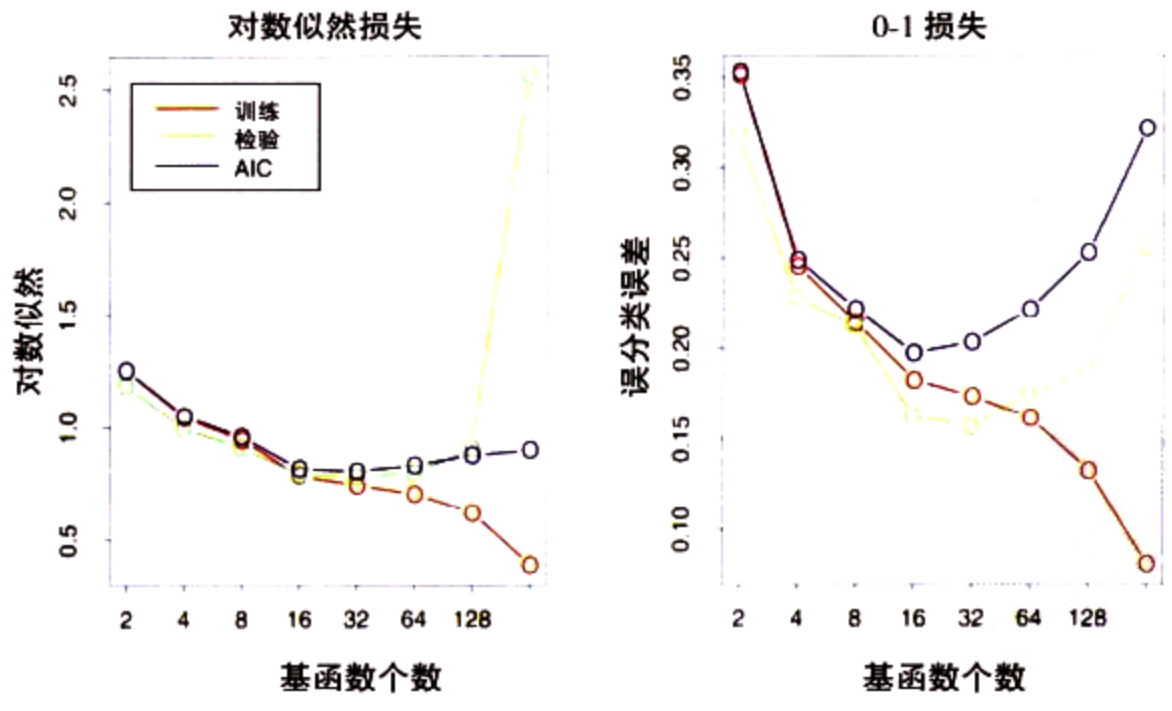


图 7.4

AIC 用于第 5.2.3 节音素识别例子的模型选择。逻辑斯缔回归系数函数 $\beta(f)=\sum_{m=1}^{M} h_m(f)\theta_m$ 被建模成 M 个样条基函数的展开式。在左图中，我们看到使用对数似然损失估计的 $\mathrm{Err}_{\mathrm{in}}$ 的 AIC 统计量。所包含的是基于一个独立检验样本的 Err 估计。除了极端地过分参数化情形外（对于 N=1000 个观测，M=256 个参数），它都做得很好。在右图中，对 0-1 损失做了同样的事情。尽管严格地说 AIC 公式不该在此处应用，但它应用于该情形却也很理想

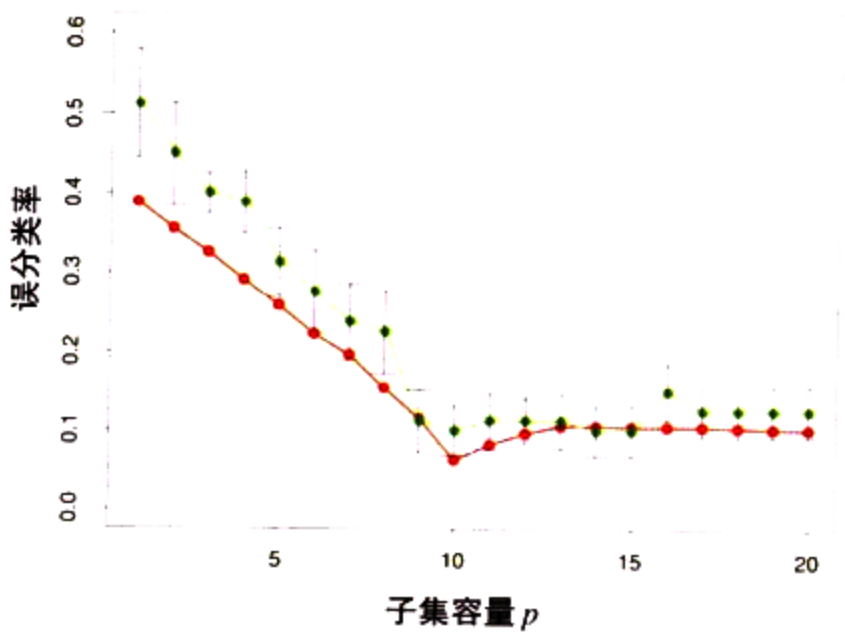
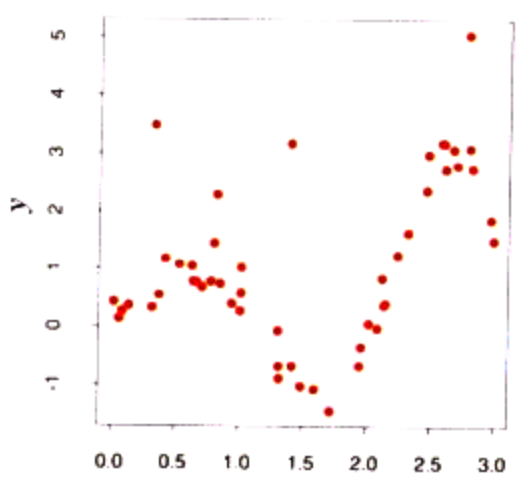


图 7.9

图7.3的底部右图的方案中，从单一训练集估计的预测误差（红色）和10折交叉验证曲线（绿色）



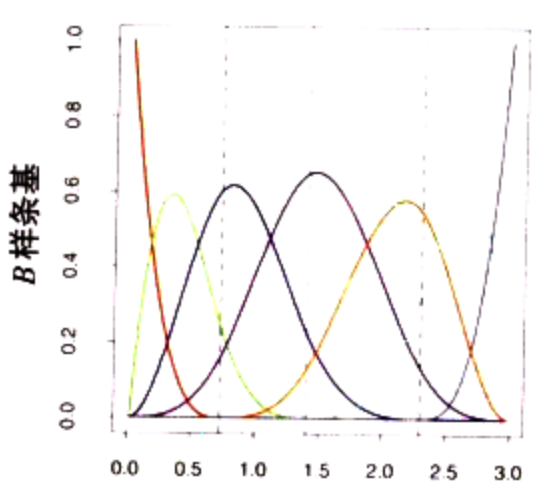


图 8.1

左图：光滑例子的数据。右图：7个 B 样条基函数的集合。垂直的虚线指出三个纽结的布局

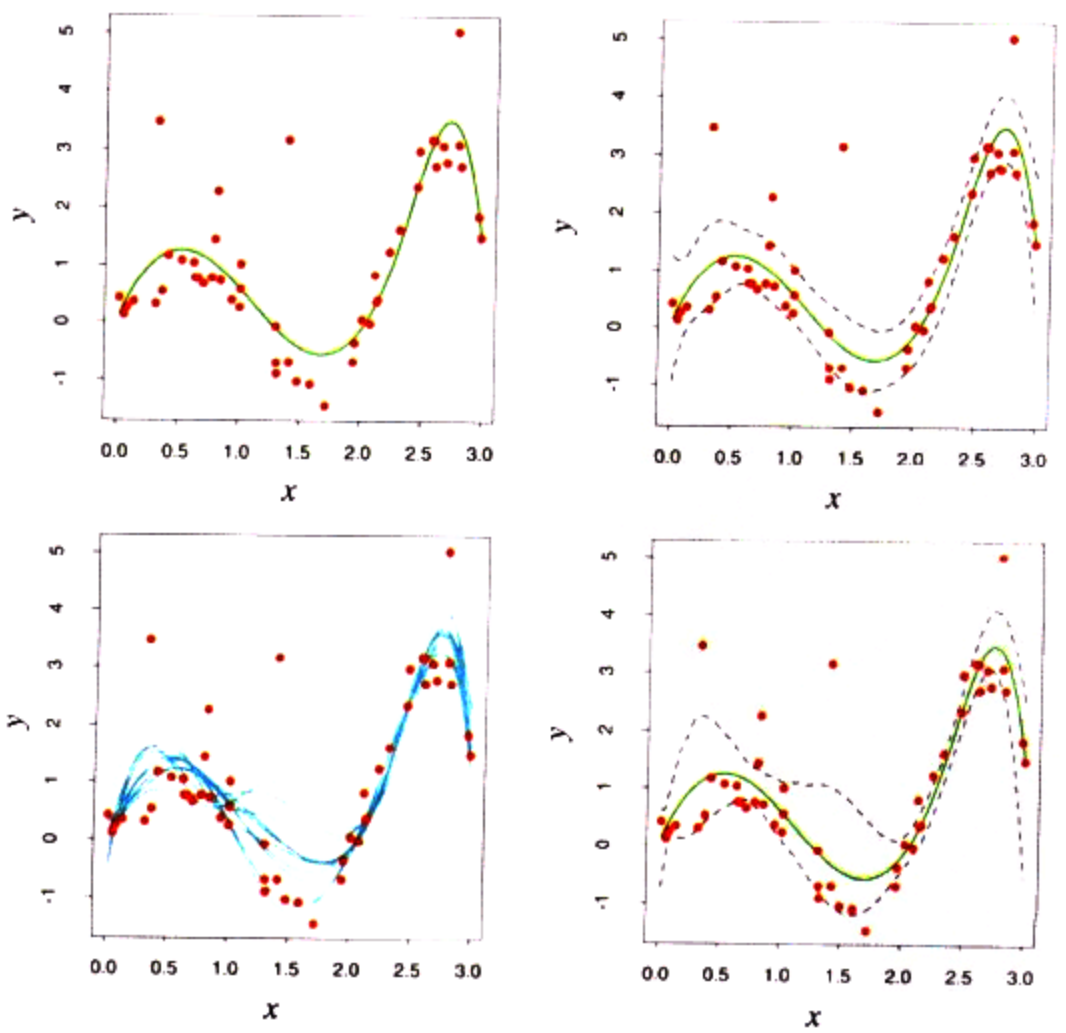


图 8.2

左上：数据的B样条光滑。右上：B 样条光滑加减 1.96 倍标准误差带。左下：B样条光滑的10个自助法重复实验。右下：从自助法分布计算的有95%标准误差带的 B 样条光滑

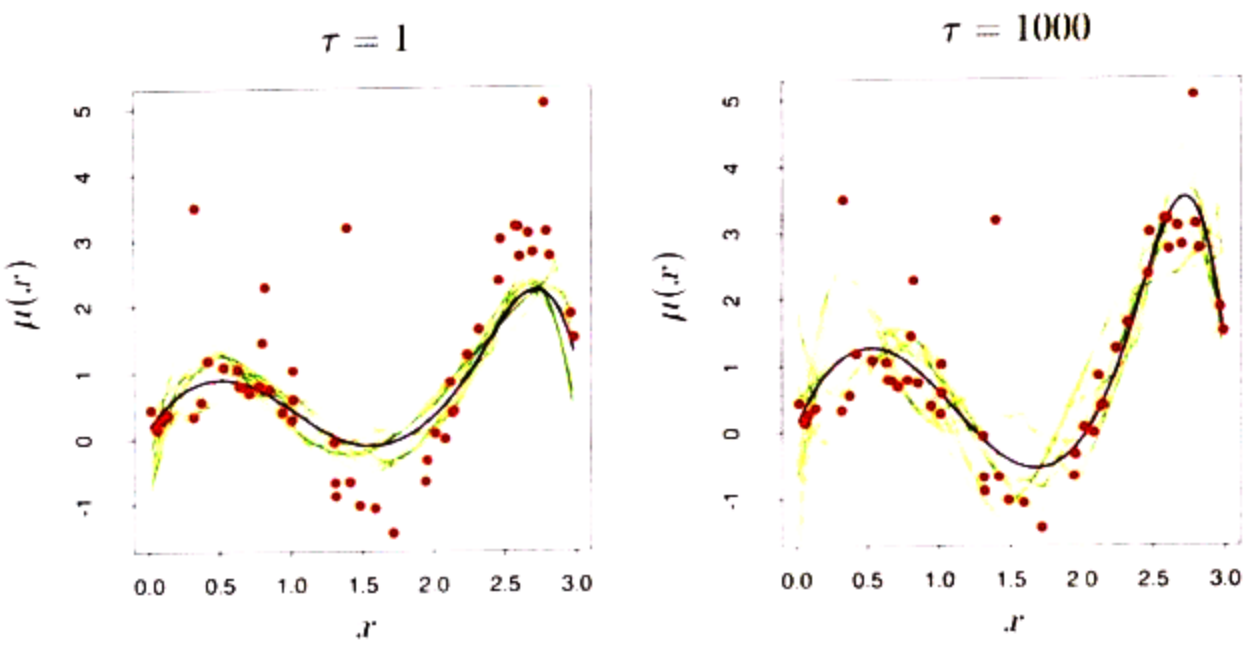


图 8.4

光滑例子：对于先验方差 τ 的两个不同的值，函数 $\mu(x)$ 的后验分布的 10 条曲线。紫色曲线是后验均值

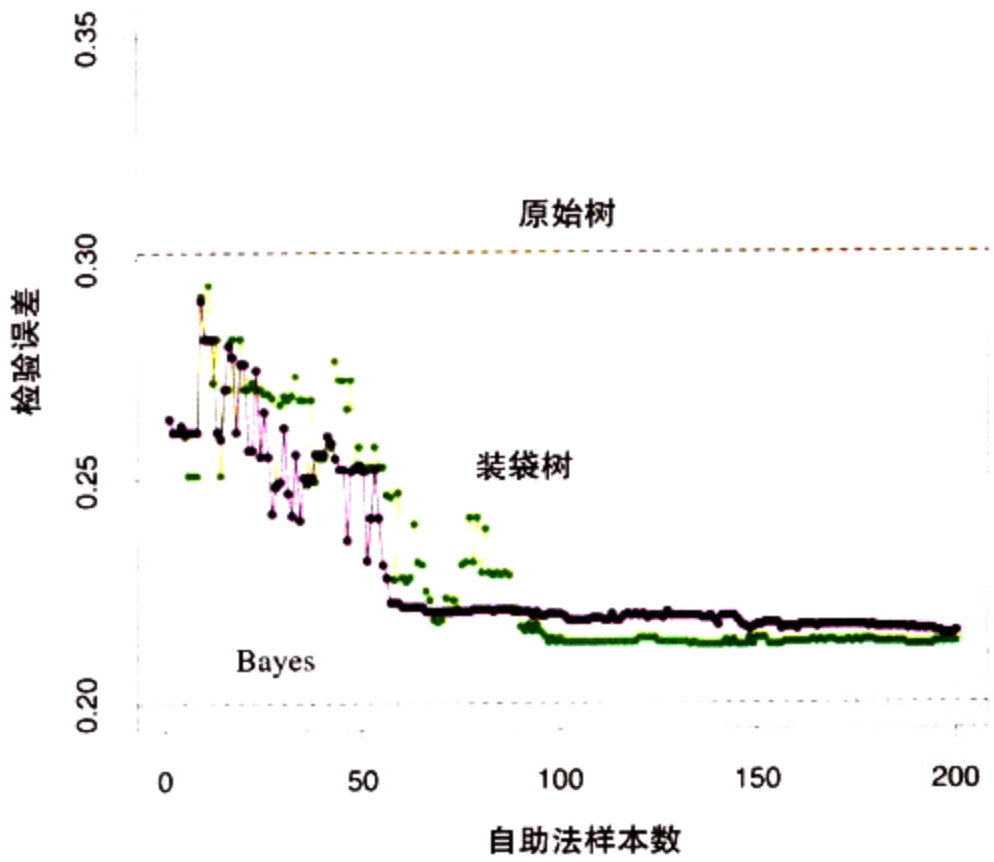


图 8.10

图8.9中装袋例子的误差曲线。所显示的是原始树和装袋树的检验误差，作为自助法样本数量的函数。绿色点对应于多数表决，而紫色点是概率的平均值

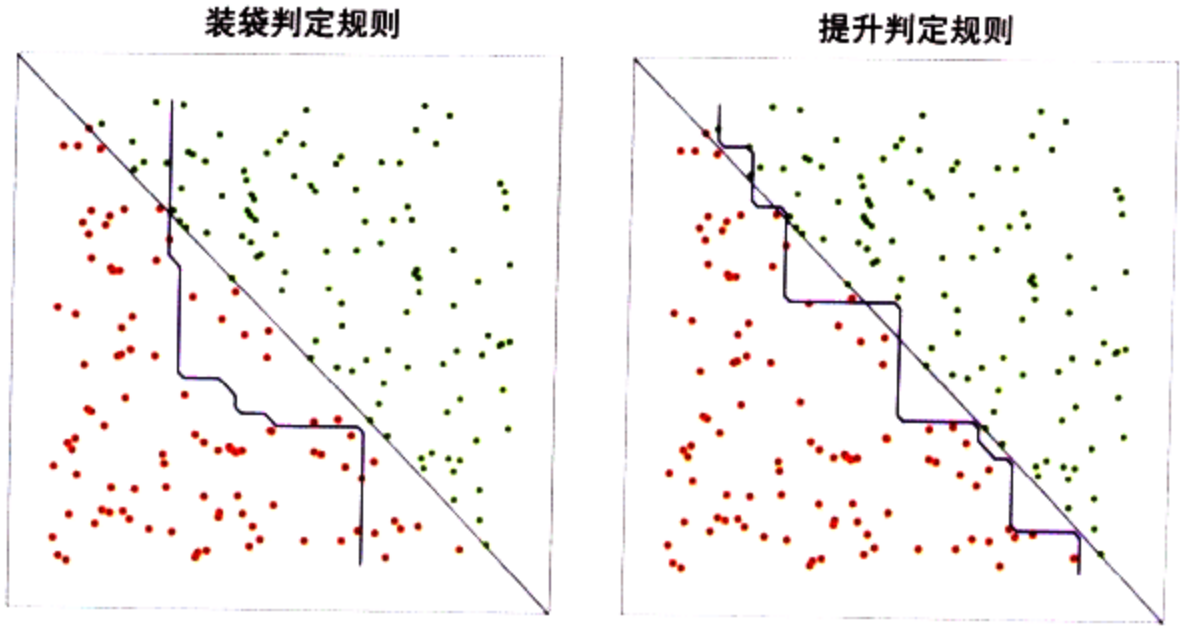


图 8.11

具有两个特征和两个类的数据，被一个线性边界分开。左图：对单个轴向分类子的判定规则装袋估计的判定边界。右图：由提升相同分类子的判定规则得到的估计判定边界。检验误差率分别为 0.166 和 0.065。提升方法将在第 10 章介绍

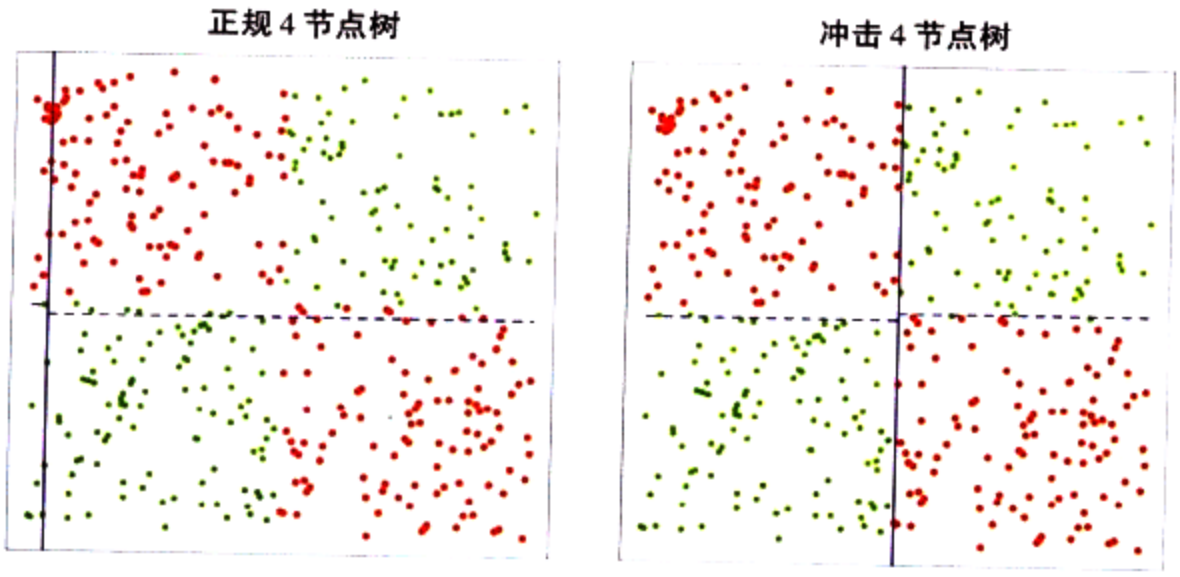
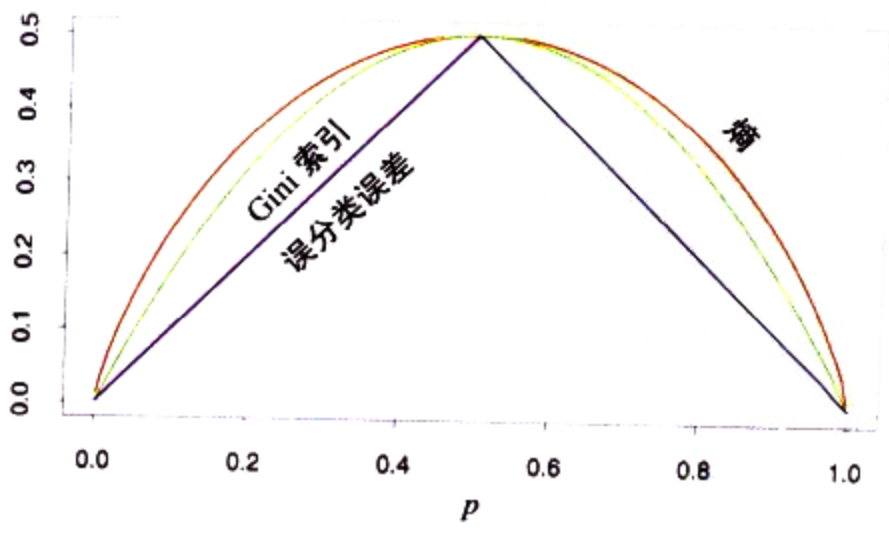


图 8.12

具有两个特征和两个类（绿色和红色）的数据，显示出纯交互效应。左图显示了由一个标准、贪心的树增长分裂方法的三次分裂所发现的划分。靠近左边界的蓝色垂线是最初的分裂，虚线是两个随后得到的分裂。算法不知道好的初始分裂在哪里，做出了一个很差的选择。右图则显示了冲击树增长算法 20 次所发现的接近最佳的分裂

图 9.3

2-类分类的节点非纯度量，是类 2 中的比例 p 的函数。互墒经过点（0.5,0.5）



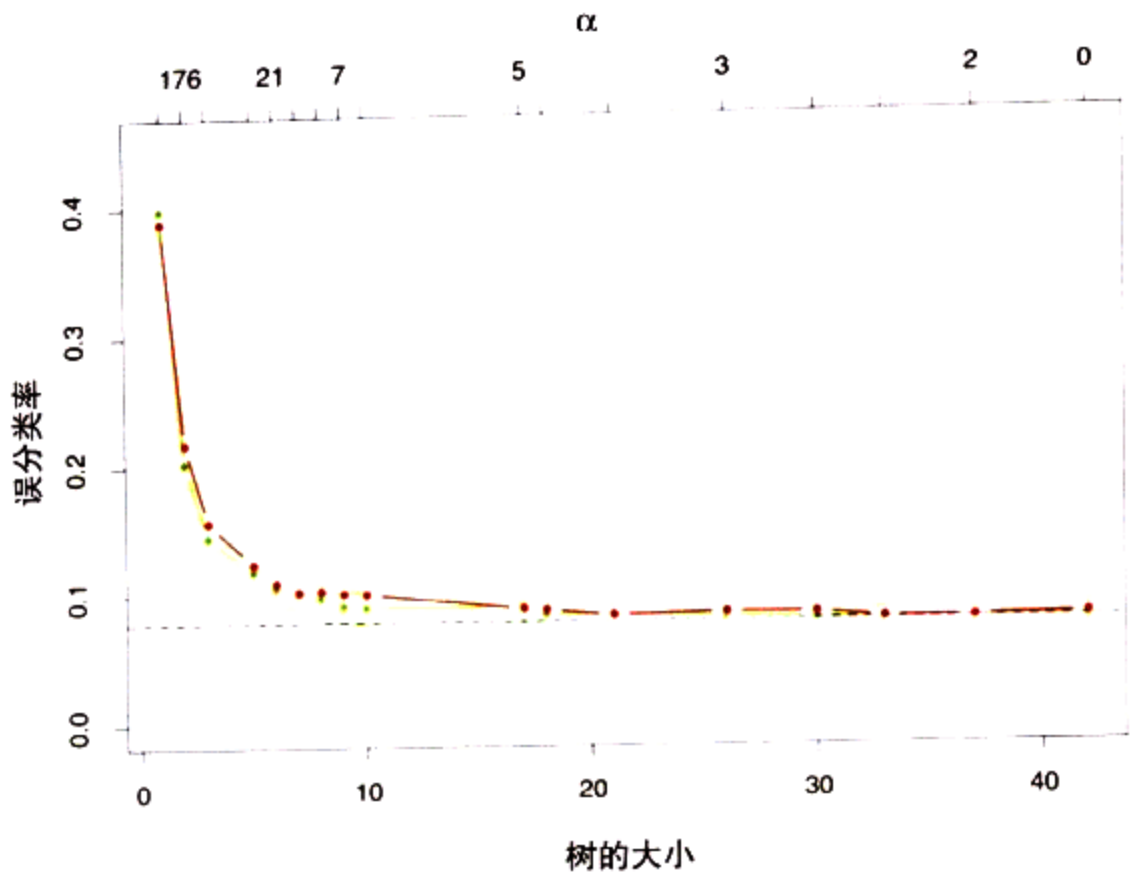


图 9.4

垃圾邮件例子的结果。绿色曲线是误分类率的10折交叉验证估计，误分类率是树规模的函数，有 ± 2 倍标准误差。极小值出现在大小约有 17 个端节点的树上。红色曲线是检验误差，它与CV 误差非常接近。交叉验证被 α 标引，在图中上方显示。显示在图底部树的大小是指剪枝后原始树的大小 $|T_\alpha|$

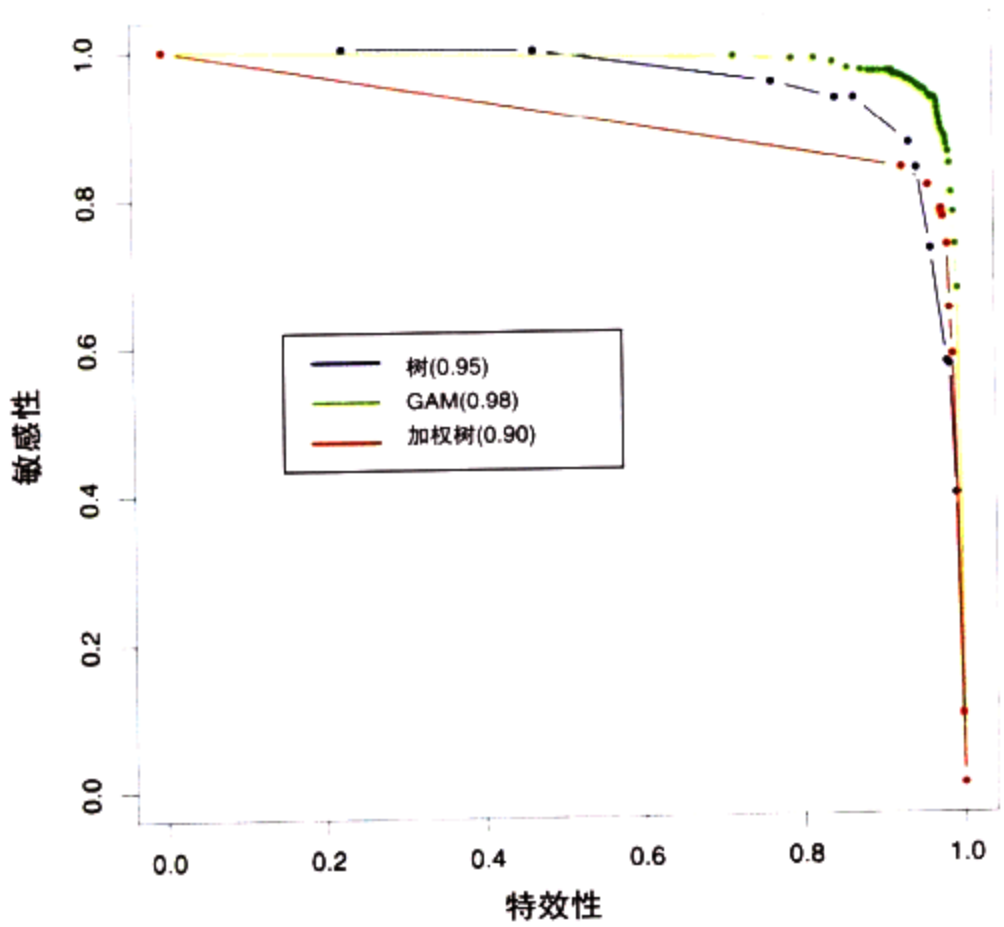


图 9.6

拟合垃圾邮件数据的分类规则的ROC 曲线。靠近右上角的曲线展示了较好的分类器。在此情况下，GAM 分类器优于树。对于较高的特效性，加权树比不加权树能获得较好的敏感性。插图中的数字表示曲线下面的面积

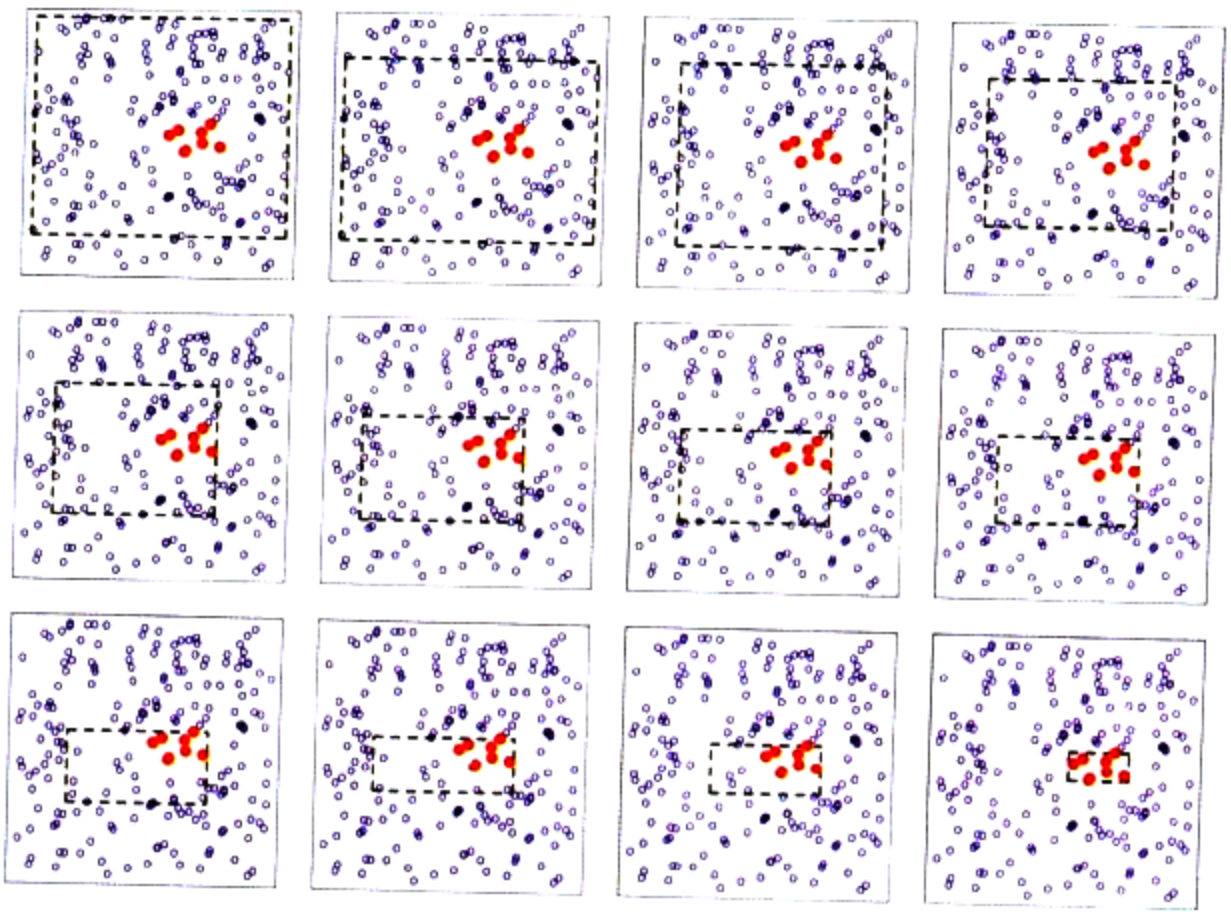


图 9.7

PRIM 算法图解。这里有两个类，分别用蓝点（类 0）和红点（类 1）指示。过程从包围所有数据的矩形（黑色虚线）开始，然后，按预先指定的量沿一条边剥除点，使得留在箱中的点的均值极大化。从左上角开始，显示剥除序列，直到最右下角纯红色区域被隔离为止

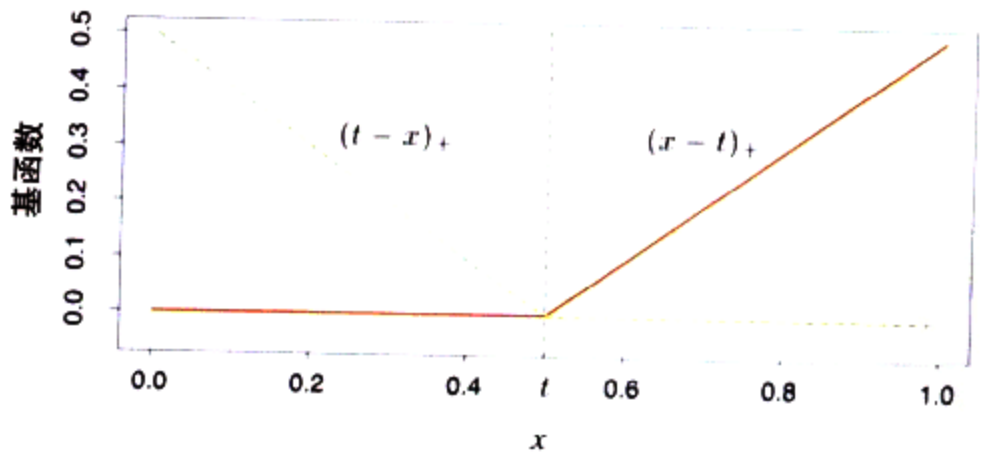


图 9.9

MARS 使用的基函数$(x-t)_+$（红色实线）和$(t-x)_+$（绿色虚线）

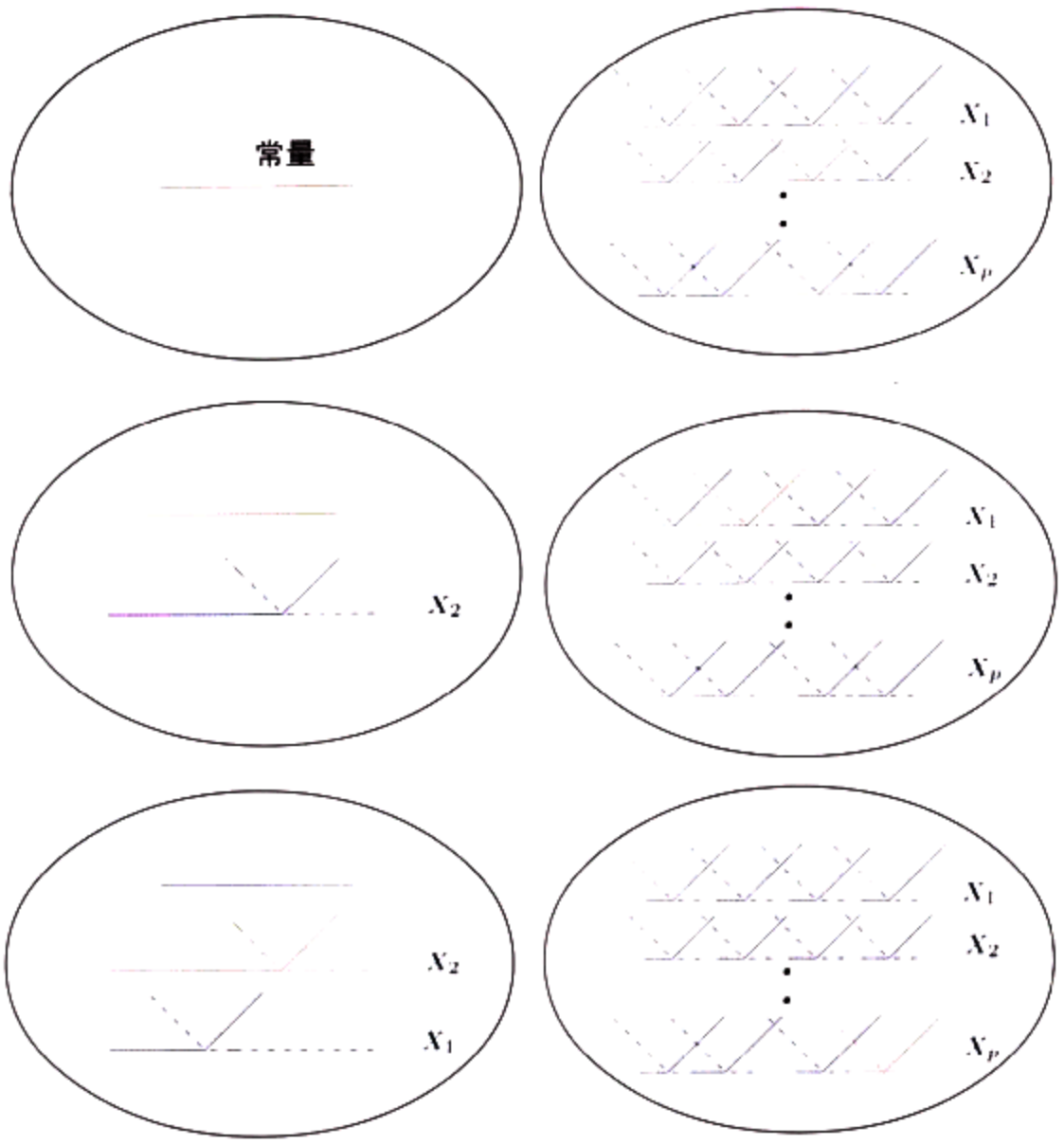


图 9.10

MARS 前向模型建立过程示意图。左侧是当前模型中的基函数：初始时，它是常量函数 $h(X) = 1$。右侧是在构造模型时需要考虑的全部候选基函数。它们是图 9.9 中所示的分段线性函数对，纽结 t 在每个预测 X_j 的全部惟一观测值 x_{ij} 上。在每一步，我们考虑候选对与模型中基函数的所有积。并将最大程度降低残差的积添加到当前模型中。上面我们描述了过程的前三个步骤，所选择的函数用红色表示

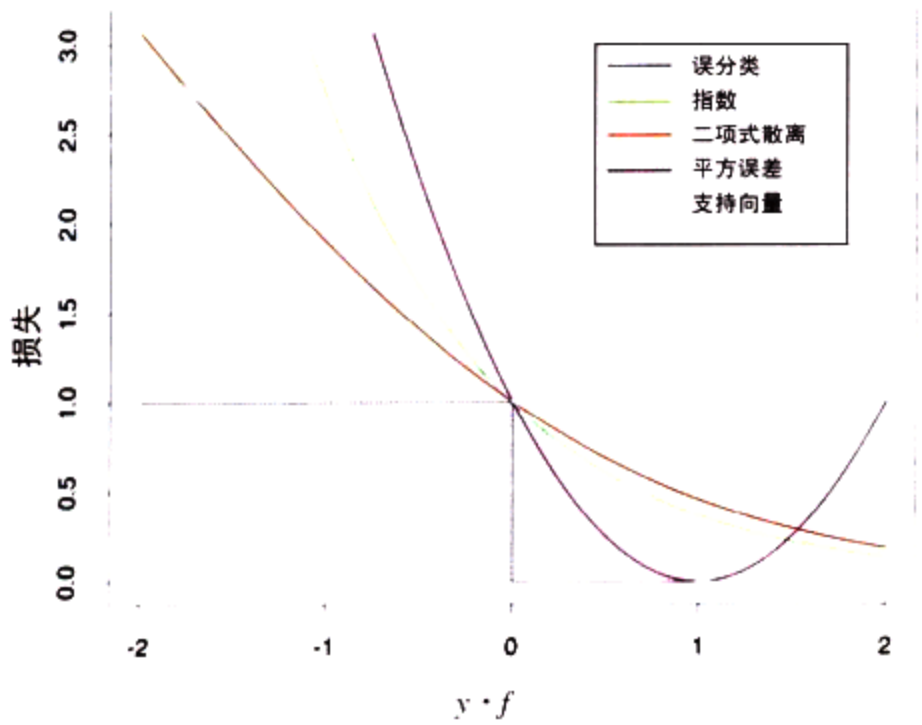


图 10.4

2- 类分类的损失函数。响应是 $y = \pm 1$；预测是 f，类预测是 $\text{sign}(f)$。损失是误分类：$I(\text{sign}(f) \neq y)$；指数：$\exp(-yf)$；二项式散离：$\log(1+\exp(-2yf))$；平方误差：$(y-f)^2$；支持向量：$(1-yf)I(yf>1)$（参见第 12.3 节）。每个函数已经被缩放以便经过点（0, 1）

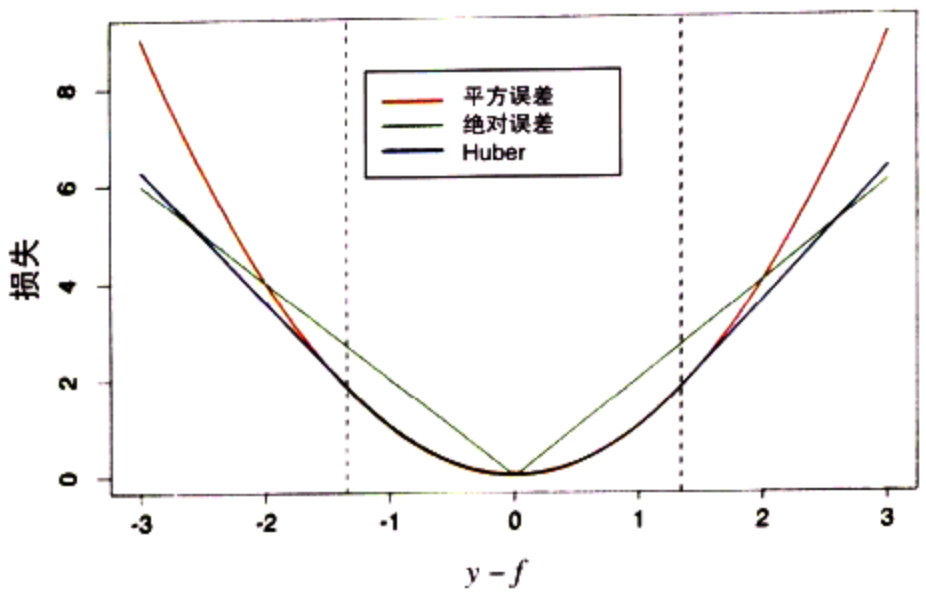


图 10.5

回归所用的三种损失函数的比较，绘制的曲线是$y-f$的函数。Huber损失函数结合了在0附近的平方误差损失和在$|y-f|$较大时的绝对误差损失的好的性质

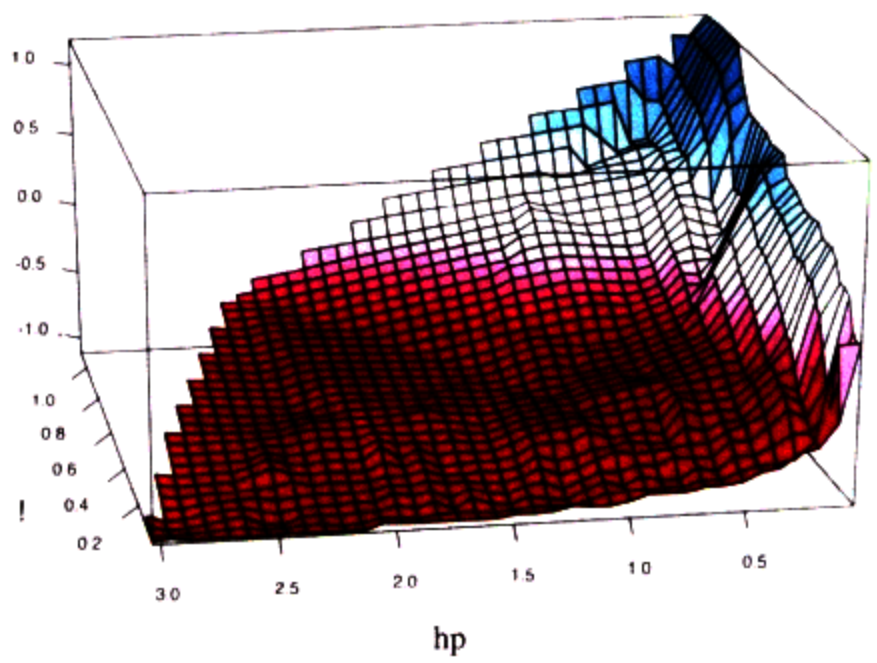


图 10.8

作为hp和!联合的频率函数，spam对email的对数几率的偏依赖

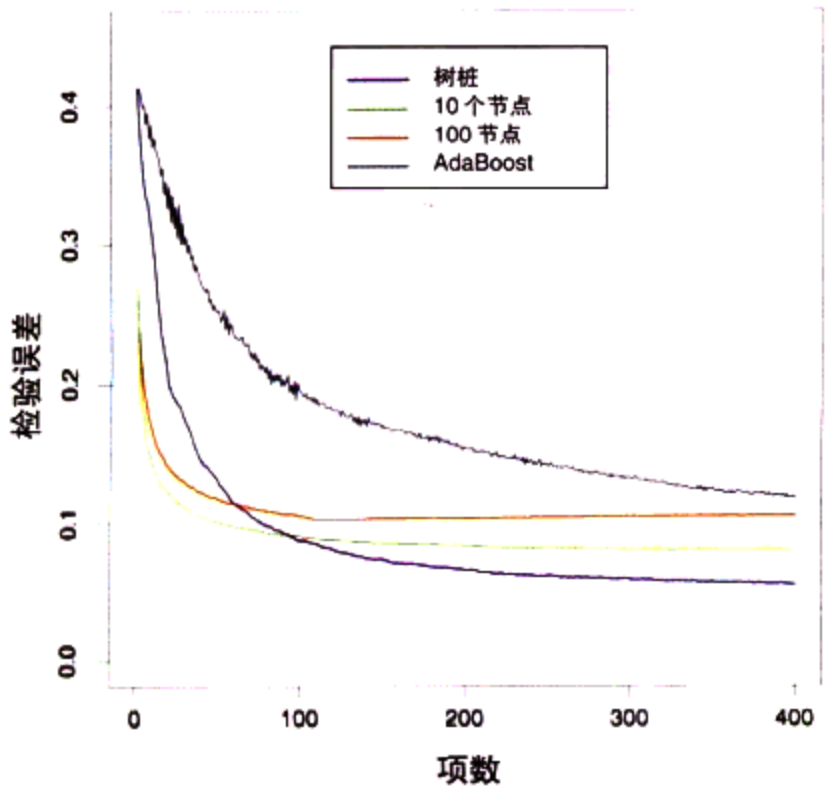


图 10.9

提升不同大小的树，用于图10.2使用的例子(10.2)。由于生成的模型是加法的，所以树桩性能最好。提升算法使用算法10.3中的二项式散离损失；为了比较，还显示了AdaBoost算法10.1

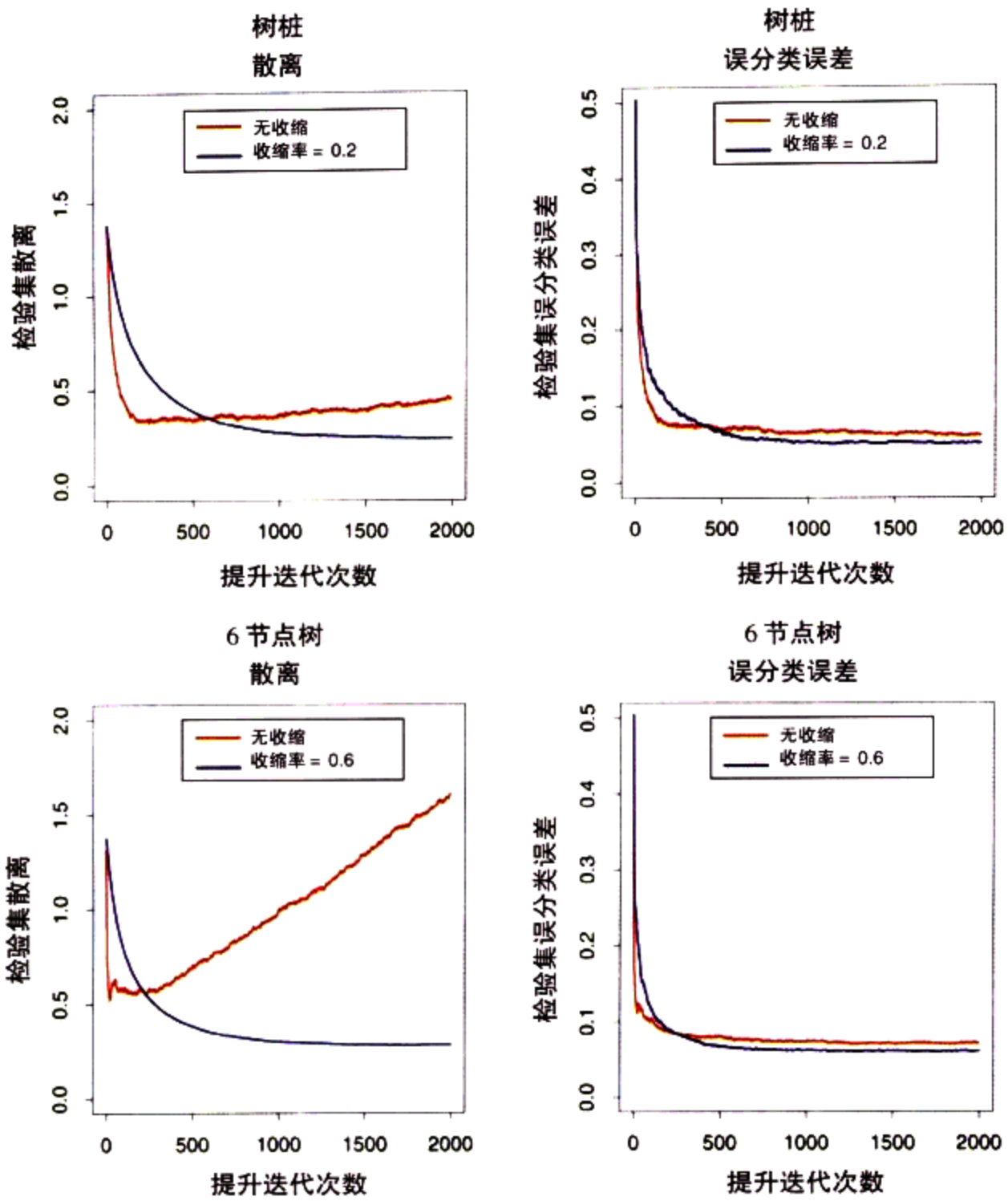


图 10.11

使用MART，图 10.9 模拟例子(10.2)的检验误差曲线。使用二项式散离、树桩或 6 节点树训练模型，有或没有收缩的情形。左侧报告检验散离，而右侧显示误分类误差。收缩的有利效果在所有情况中都能看得到，特别是对于左边显示的散离

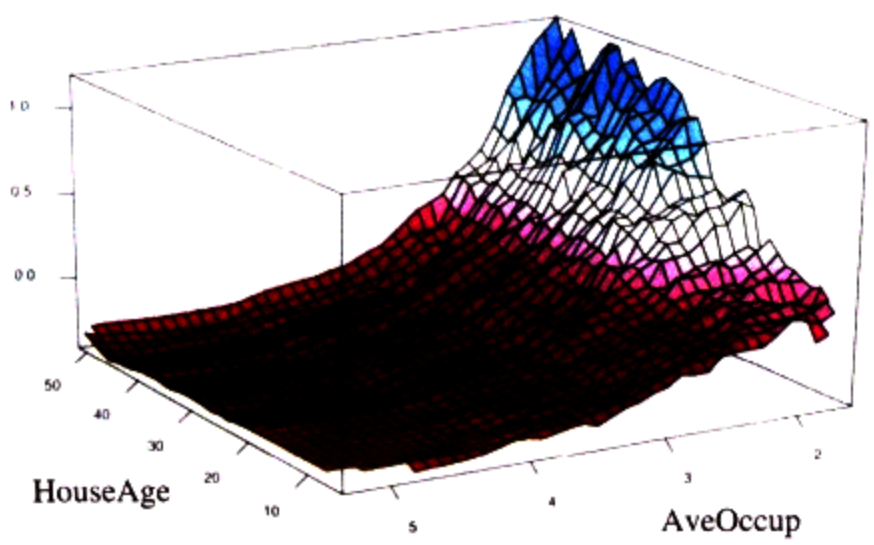


图 10.16

在中等房龄和平均面积上房价的偏依赖。在这些两变量间似乎存在着很强的交互

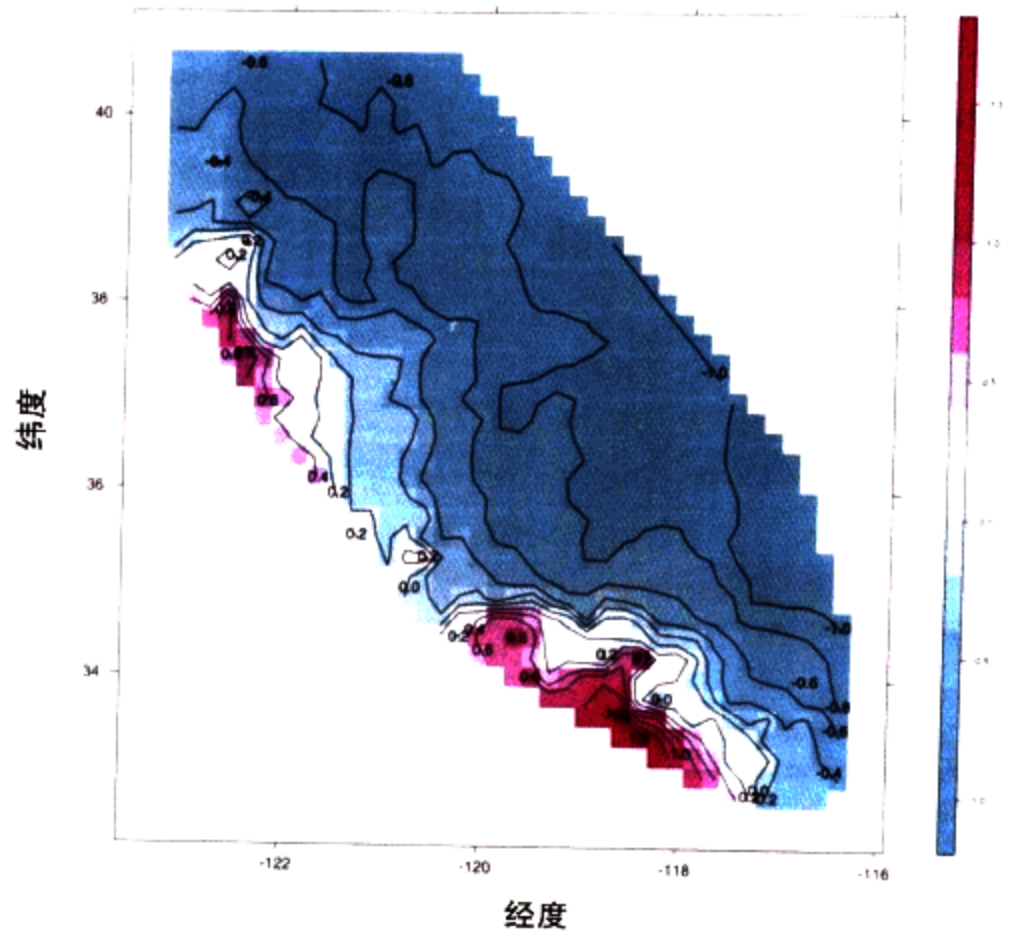


图 10.17
房价中值对加利福尼亚位置的偏依赖

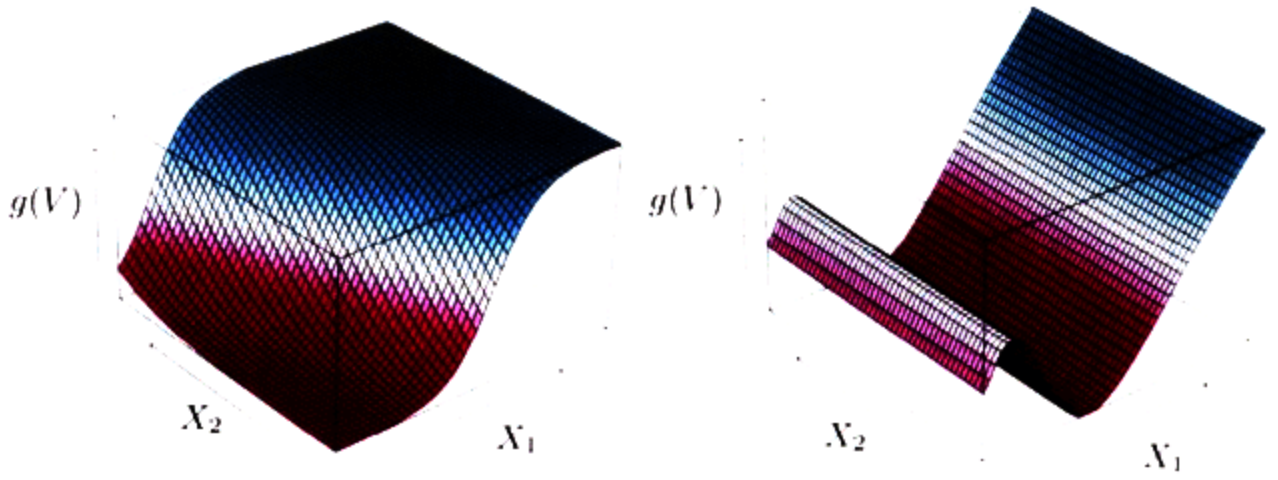
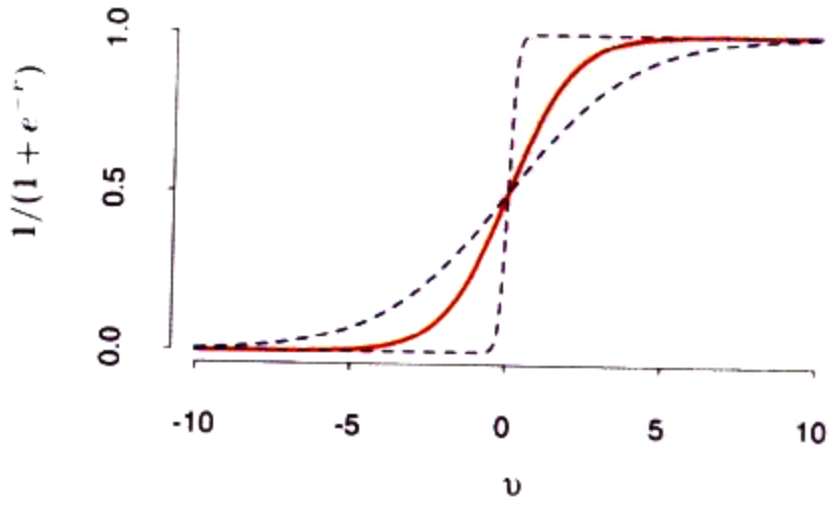


图 11.1
两个岭函数的透视图。左：$g(V)=1/\left[1+\exp(-5(V-0.5))\right]$，其中 $V=(X_1+X_2)/\sqrt{2}$。右：$g(V)=(V+0.1)\sin(1/(V/3+0.1))$，其中 $V=X_1$

图 11.3
S型函数 $\sigma(v)=1/(1+\exp(-v))$（红色曲线）的平面图。通常用于神经网络的隐藏层，所包含的是 $s=\frac{1}{2}$ 时的 $\sigma(sv)$（蓝色曲线）和 $s=10$ 时的 $\sigma(sv)$（紫色曲线）。缩放参数 s 控制激活率，我们可以看到大的 s 相当于在 $v=0$ 处的硬激活。注意到 $\sigma(s(v-v_0))$ 将激活阈值从 0 移动到 v_0



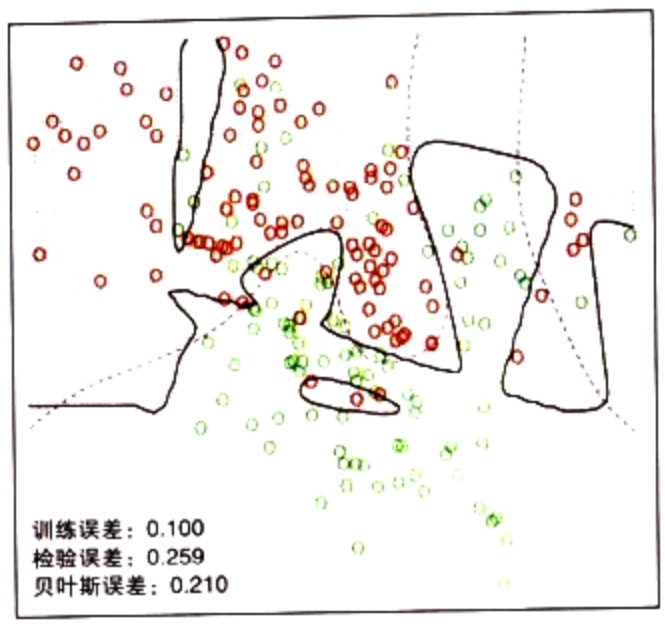


图 11.4

第 2 章混合例子上的神经网络。上图没有使用权衰减，并过分拟合训练数据。下图使用权衰减，并接近于贝叶斯误差率（紫色虚边界）。两者皆使用 softmax 激活函数和互熵误差

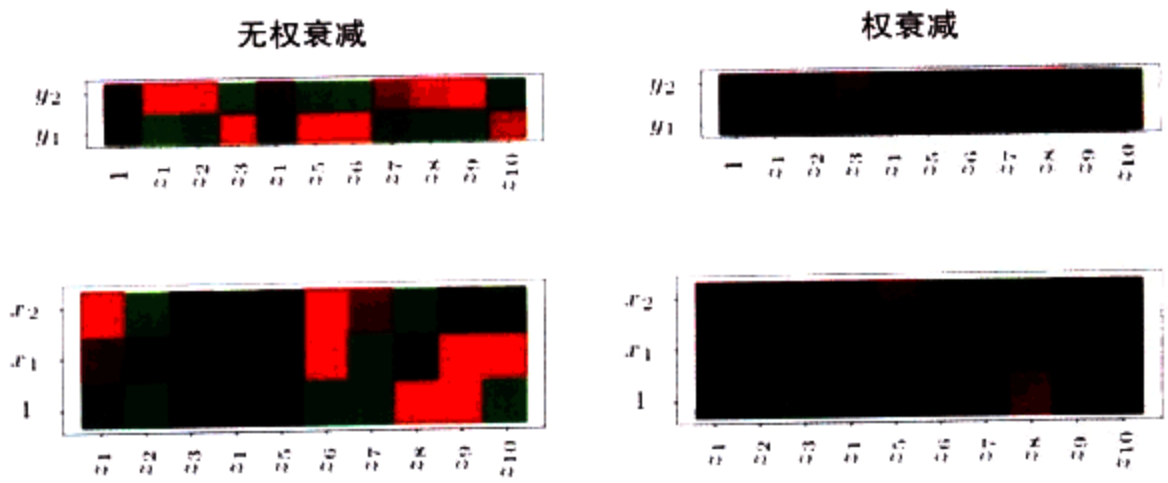


图 11.5

图 11.4 神经网络训练估计权值的热度图。显示的范围由鲜绿色（负的）到鲜红色（正的）

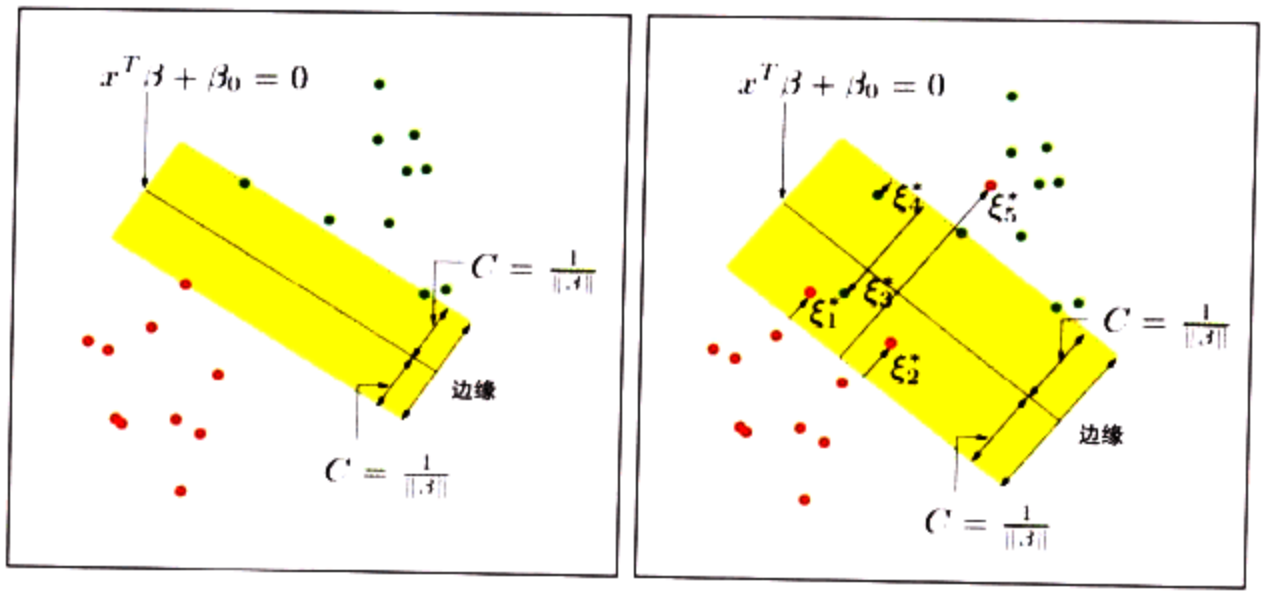
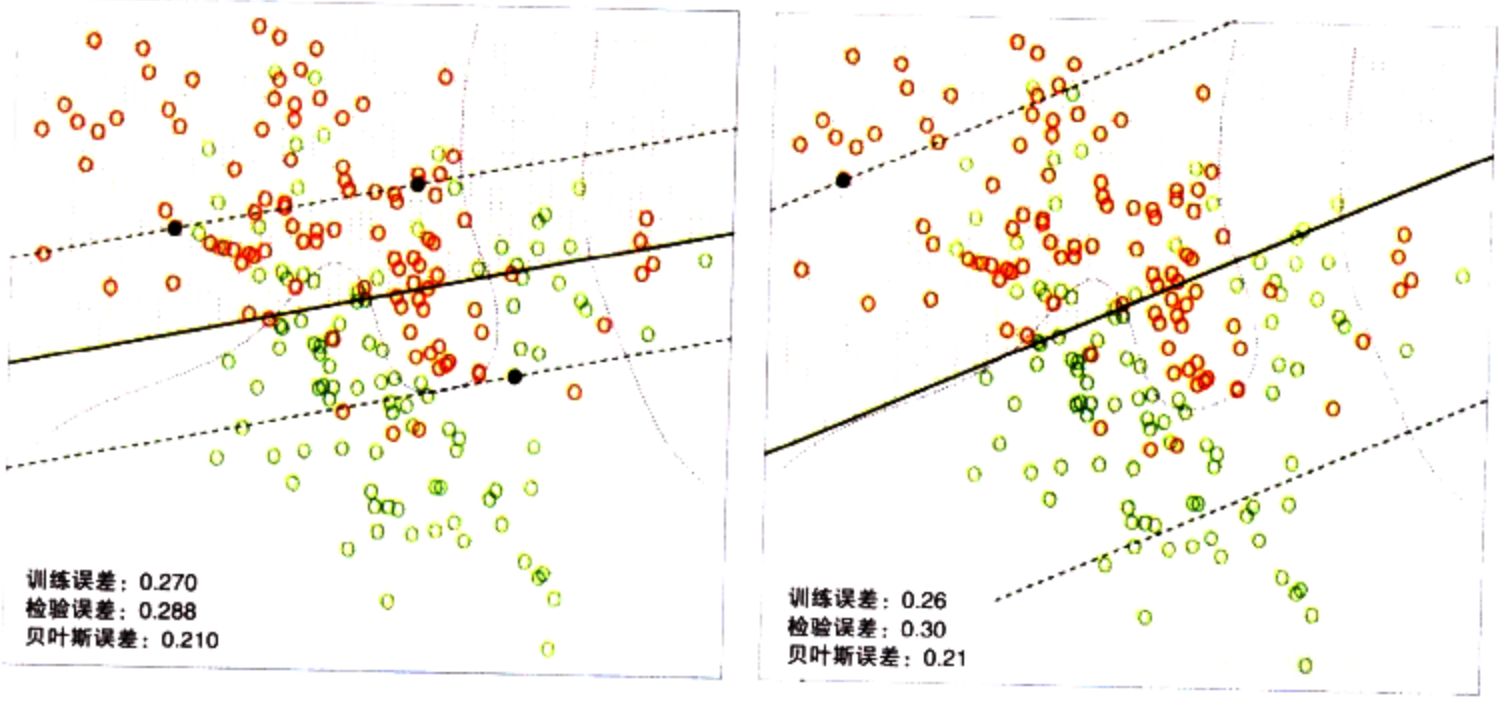


图 12.1

支持向量分类器。左图显示可分情况。判定边界是实线，而虚线界定宽度为 $2C=2/\|\beta\|$ 的阴影的最大边缘。右图显示不可分的（重叠）情况，标有 ξ_j^* 的点位于其边缘的错误侧，相差量 $\xi_j^* = C\xi_j$；在正确侧的点都有 $\xi_j^* = 0$。边缘被极大化，服从 $\sum \xi_i \leqslant$ 常量。因此，$\sum \xi_j^*$ 是在其边缘错误侧的点的总距离



$\gamma = 10000$　　$\gamma = 0.01$

图 12.2

对于两个不同的γ值，有两个重叠类的混合数据示例的线性支持向量边界。虚线指明了边缘，其中 $f(x) = \pm 1$。支持点（$\alpha_i > 0$）是在边缘错误侧上的全部点。黑实点是恰好落在边缘（$\xi_i = 0$，$\alpha_i > 0$）上的支持点。在左图中 62% 的观测是支持点，而在右图中 85% 的观测是支持点，背景上的紫色虚线是贝叶斯判定边界

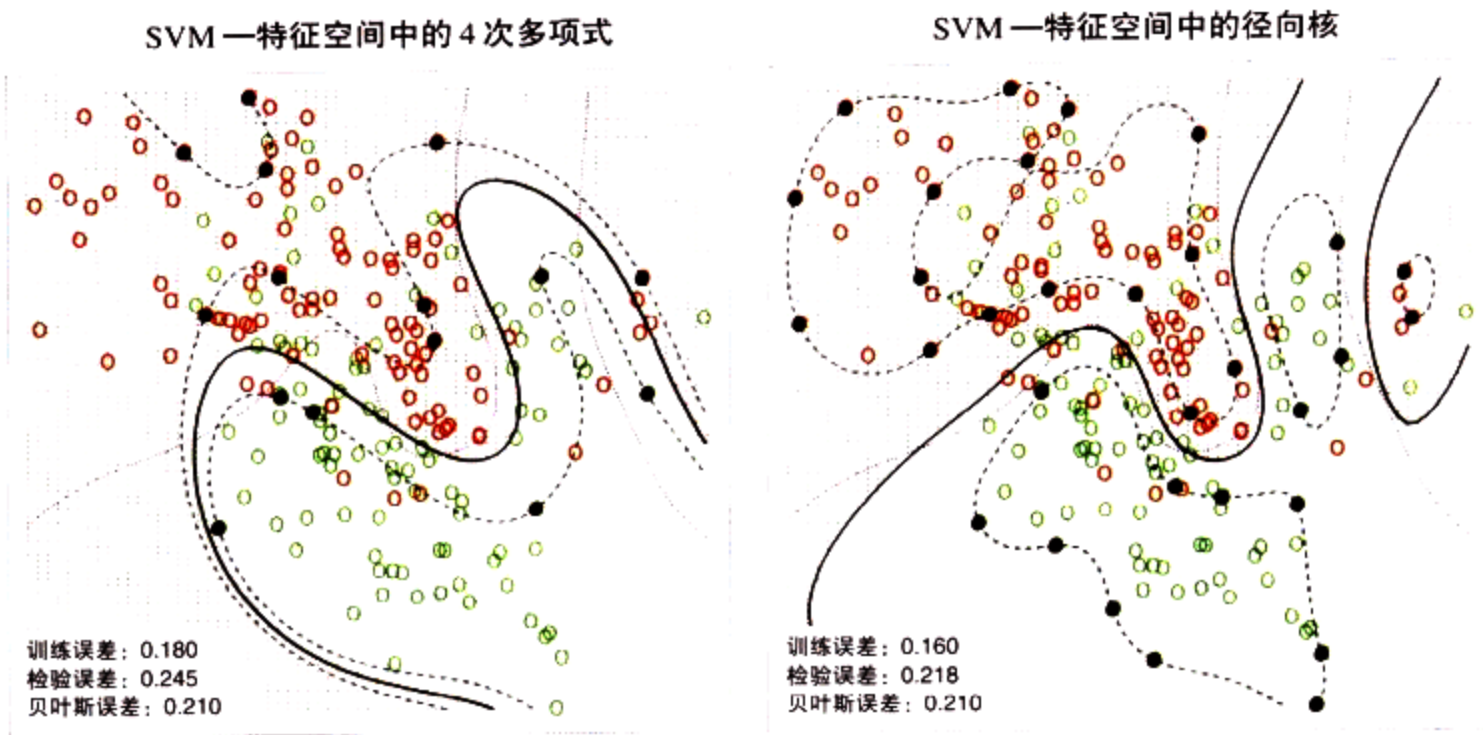


图 12.3

混合数据的两个非线性SVM。左图使用一个4次多项式核，右图使用径向基核。在每种情况下，调整γ以近似地实现最好检验误差性能，且γ=1时，两种情况做得都很好。径向核实现得最好（接近贝叶斯最优解）；给定由高斯混合产生的数据，与期望的结果一样。背景上的紫色虚线是贝叶斯判定边界

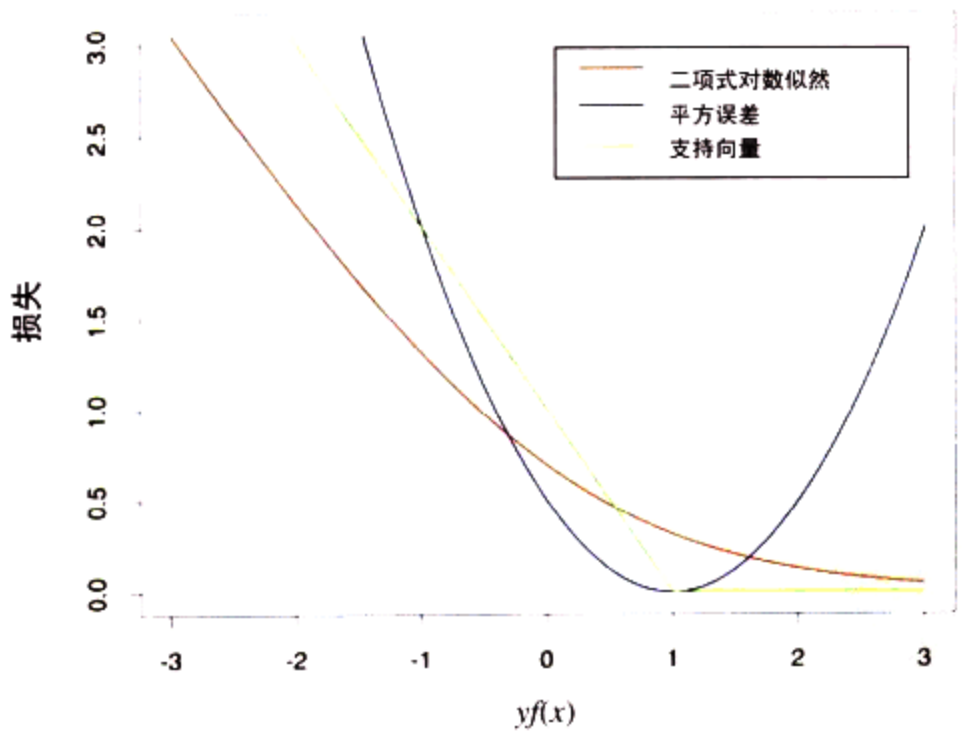


图 12.4

支持向量损失函数，与逻辑斯缔回归的（负的）对数似然损失和平方误差损失比较。所显示的都是yf而不是f的函数，因为在$y=+1$和$y=-1$之间三条曲线是对称的。对数似然与SVM有相同的渐近线，但是在内部是圆形的

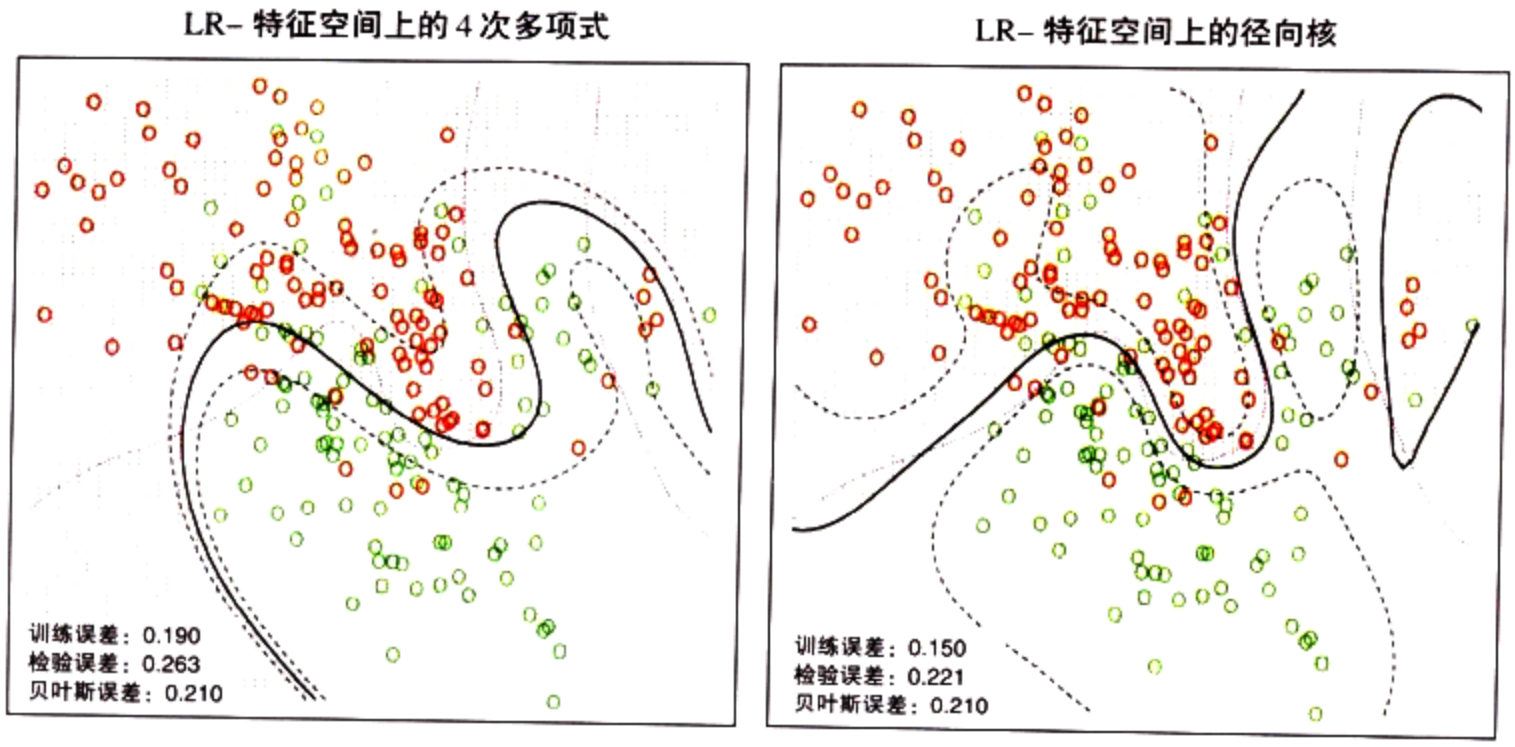


图 12.5

图 12.3 的 SVM 模型的逻辑斯缔回归版本，使用了同样的核，因此有同样的罚，但使用了对数似然损失函数，而不是SVM损失函数。两个虚线轮廓相当于 + 1 类的0.75 和 0.25 后验概率。背景上的紫色虚线是贝叶斯判定边界

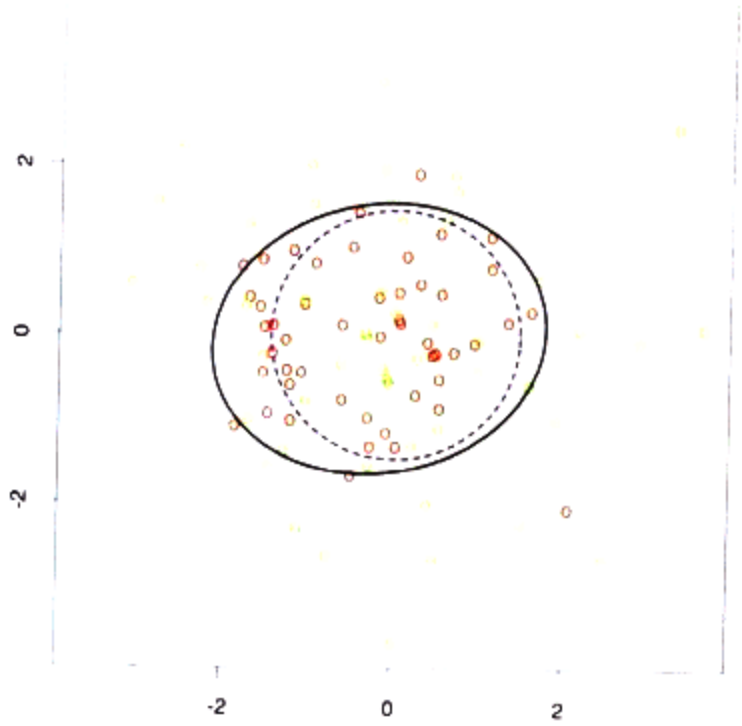


图 12.7

数据由 50 个点组成，每个数据点由 $N(0, I)$和 $N(0, \frac{9}{4}I)$产生。黑色实线椭圆是使用二次多项式回归的 FDA 发现的判定边界。紫色虚线圆是贝叶斯判定边界

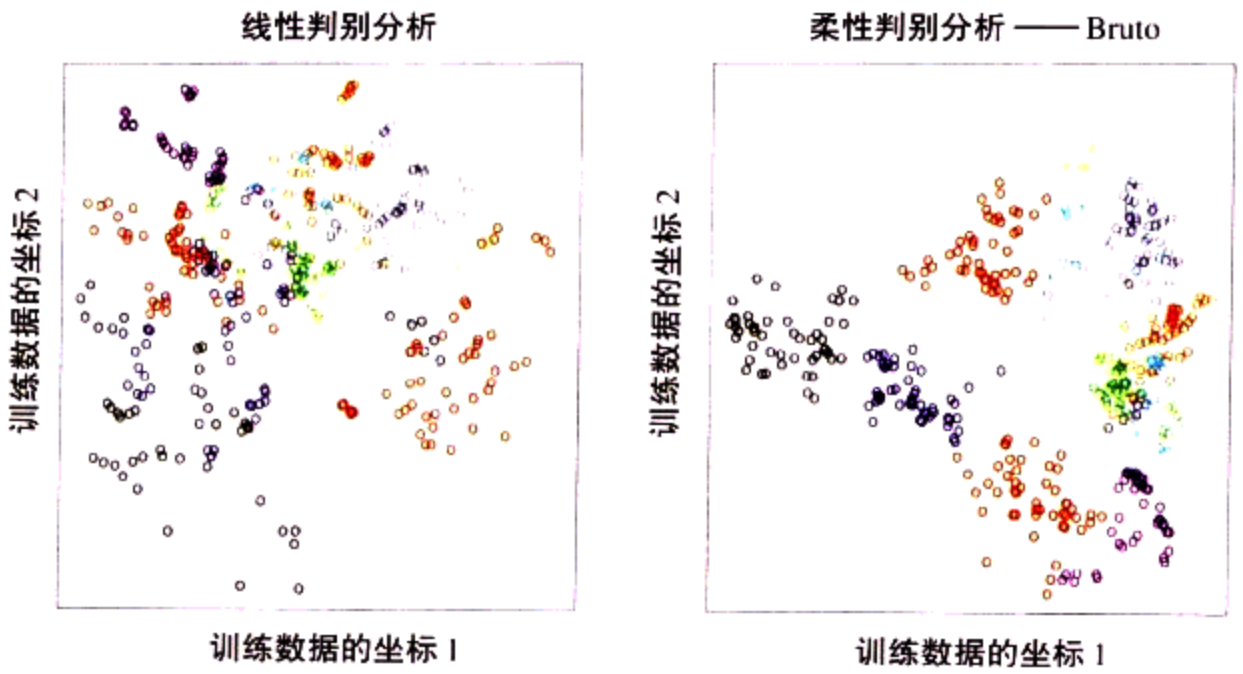


图 12.8

左图显示元音训练数据的前两个LDA正规变量。右图显示当FDA/BRUTO用于拟合模型时相应的投影；所描绘的是拟合回归函数$\hat{\eta}_1(x_i)$和$\hat{\eta}_2(x_i)$。注意改进的分离。字母标记元音

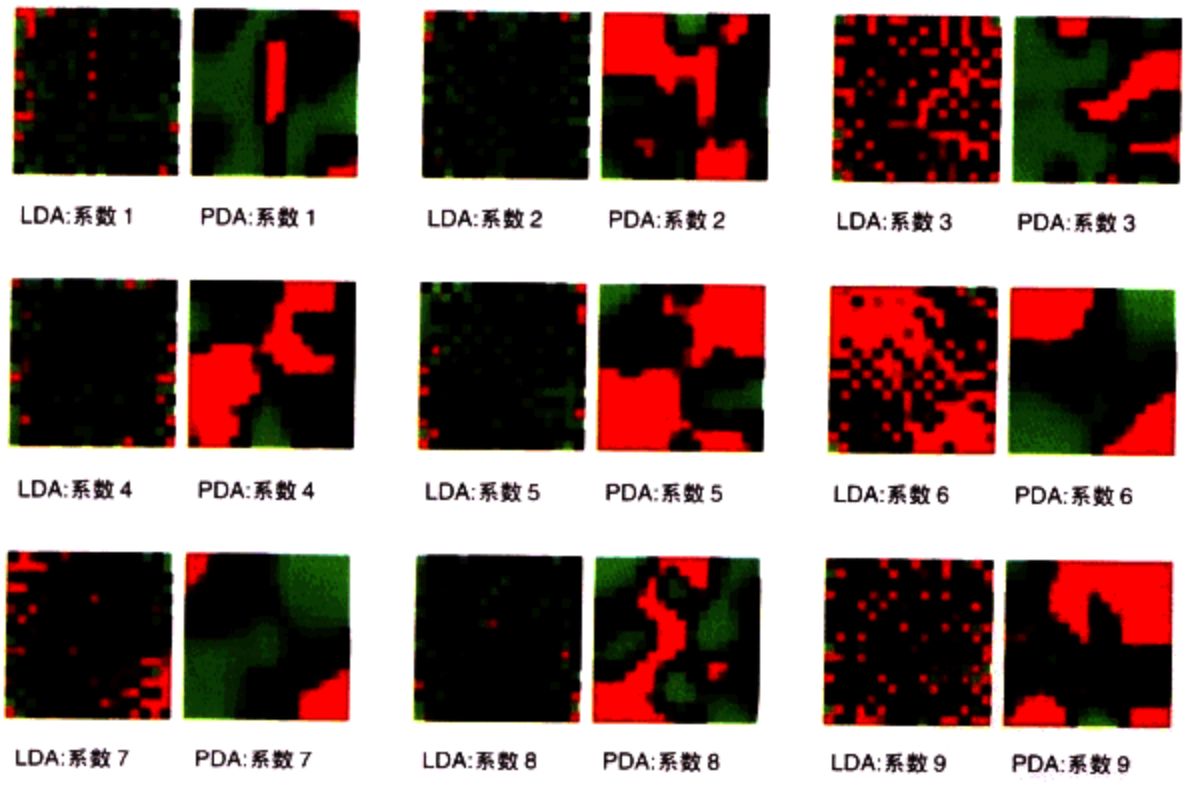


图 12.9

图像成对出现，表示数字识别问题中9个判别系数函数。每对中左边成员是LDA系数，而右边是PDA系数，已经正则化，以加强空间光滑性

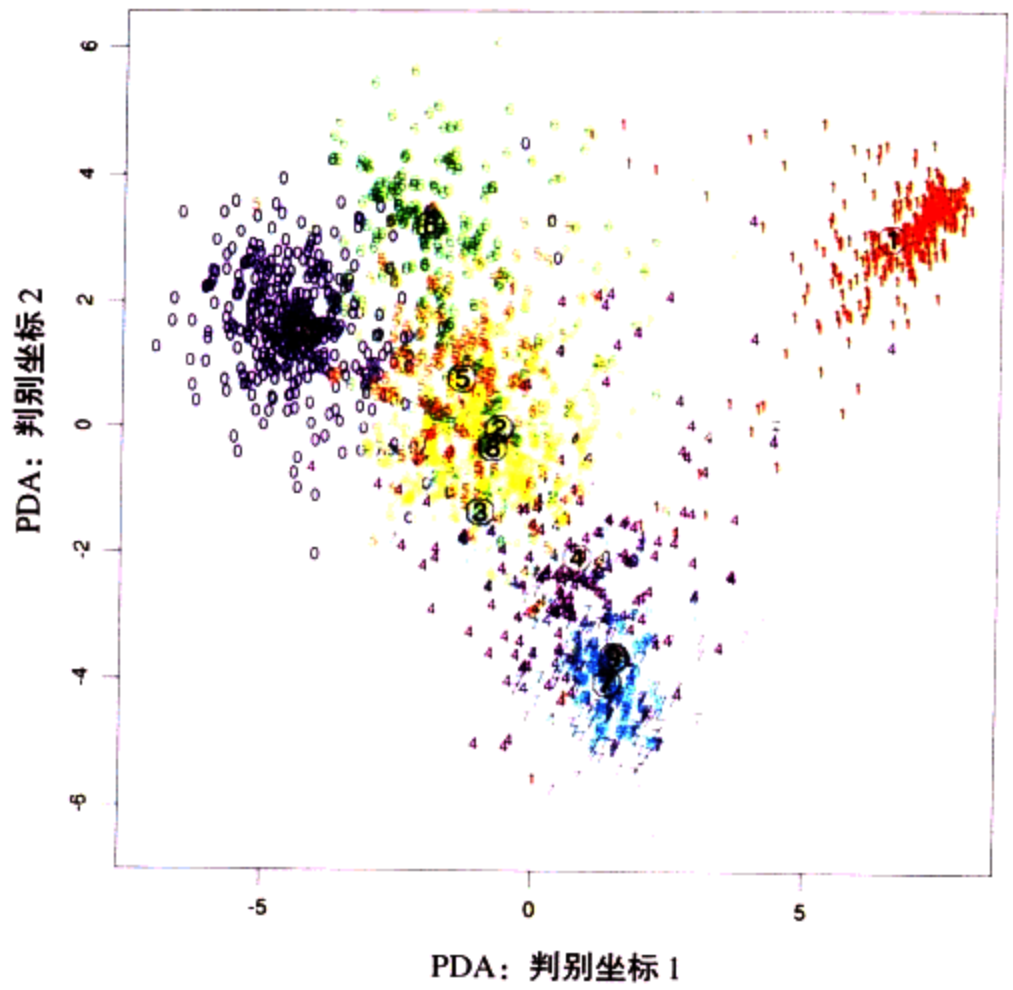


图 12.10

前两个处罚规范变量，是对检验数据求的值。圆圈指明了类的中心点。第一个坐标主要对比 0 和 1，而第二个对比 6 和 7/9

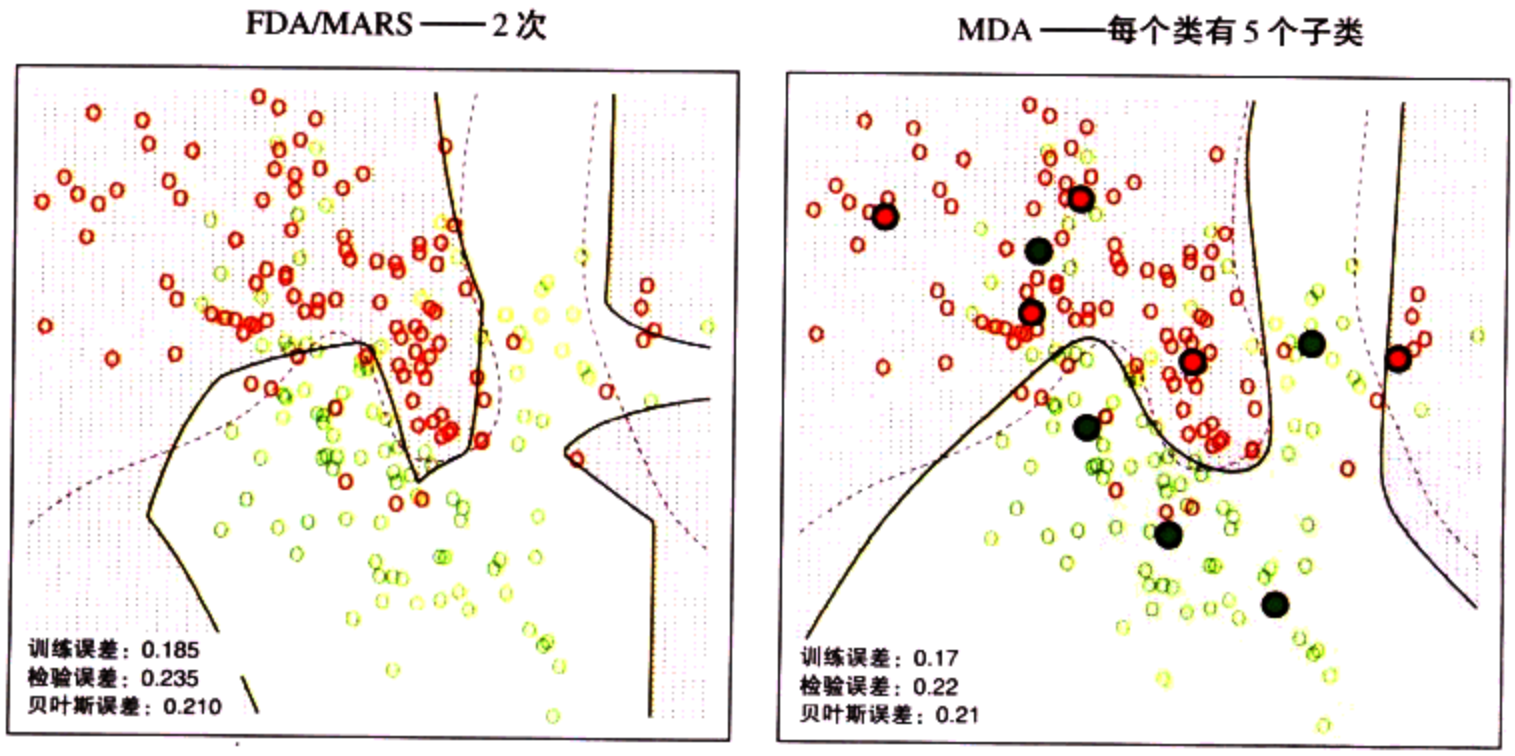


图 12.11

混合数据上的FDA 和MDA。上图使用FDA，以MARS作为回归过程。下图使用MDA，每个类有 5 个混合中心（已指出）。MDA 的解接近于贝叶斯最优解，它是由给定高斯混合数据可能期望的结果。背景上的紫色虚线是贝叶斯判定边界

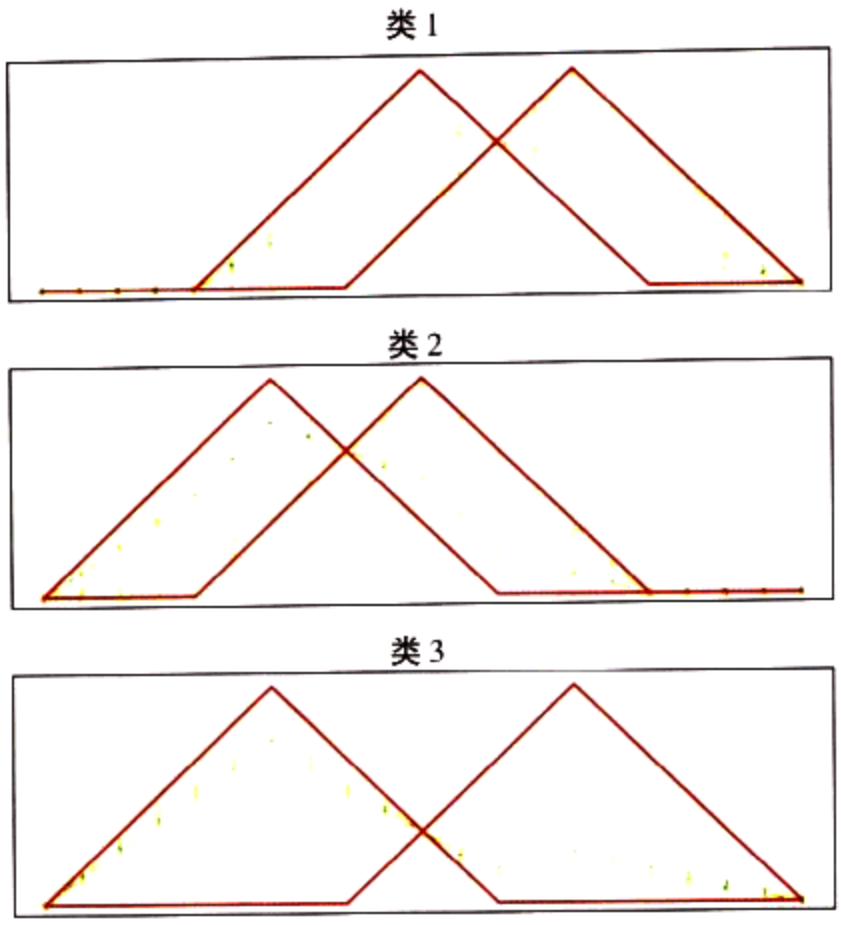


图 12.12

在高斯噪音加入前，一些由模型（12.62）产生的波形例子

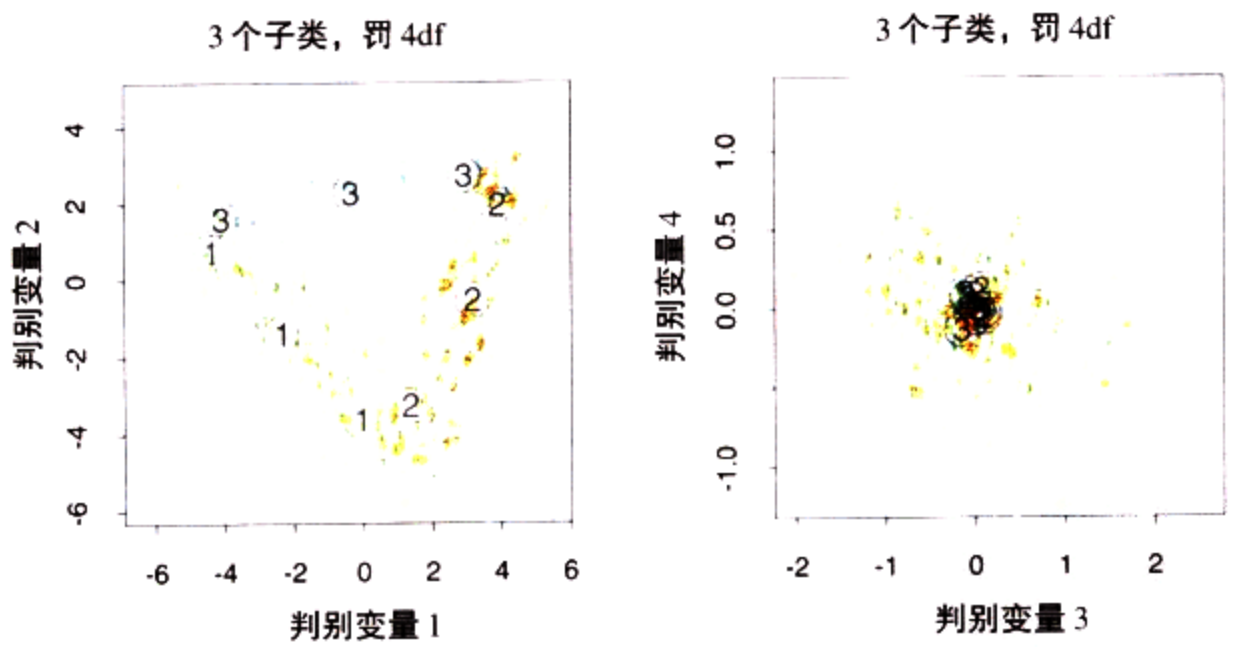


图 12.13

拟合波形模型样本的MDA模型的二维视图。点是独立的检验数据，投影到两个主标准坐标上（左图），以及第三个和第四个坐标上（右图）。图中指出了子类的中心

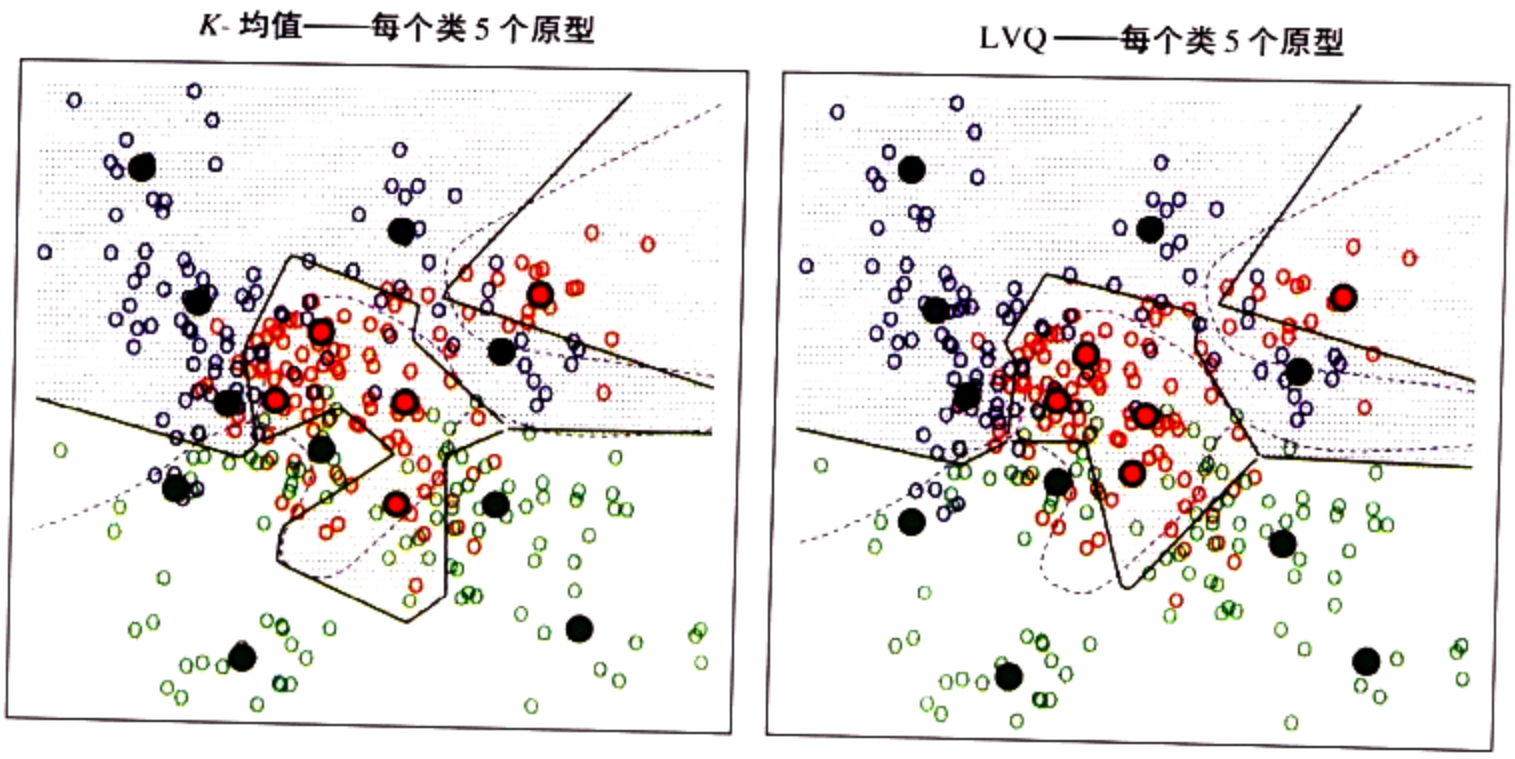


图 13.1

模拟例子，具有三个类，每个类5个原型。每个类中的数据由一个高斯混合产生。左图，通过在每个类中分别使用K-均值聚类算法找出原型。右图，LVQ算法（从一个K-均值解开始）将原型从判定边界移开。背景上的紫色虚线是贝叶斯判定边界

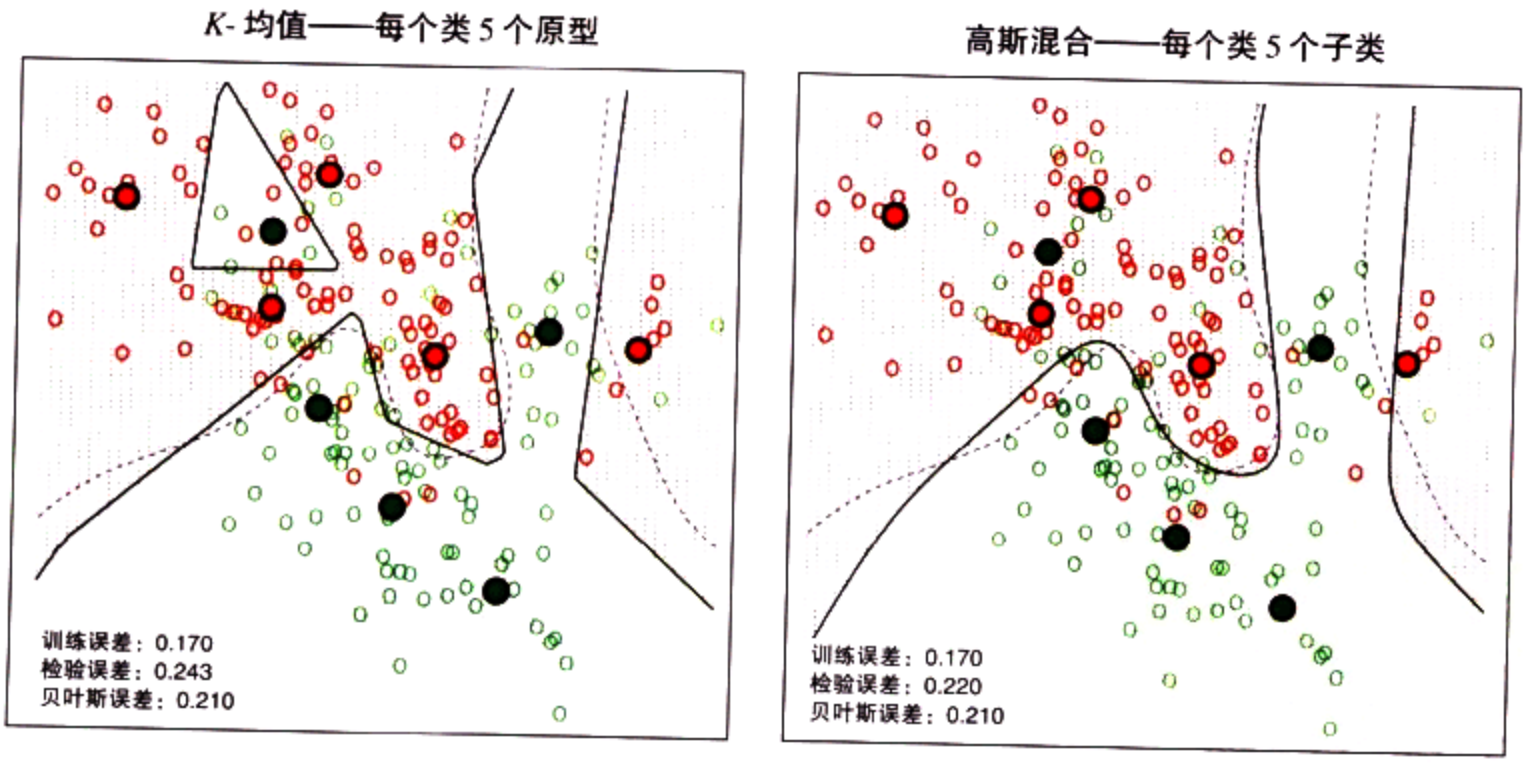


图 13.2

左图显示对混合数据例子应用的K-均值分类器。判定边界是分段线性的。右图显示高斯混合模型，所有分量高斯具有公共协方差。混合模型的EM算法开始于一个K-均值解。背景上的紫色虚线是贝叶斯判定边界

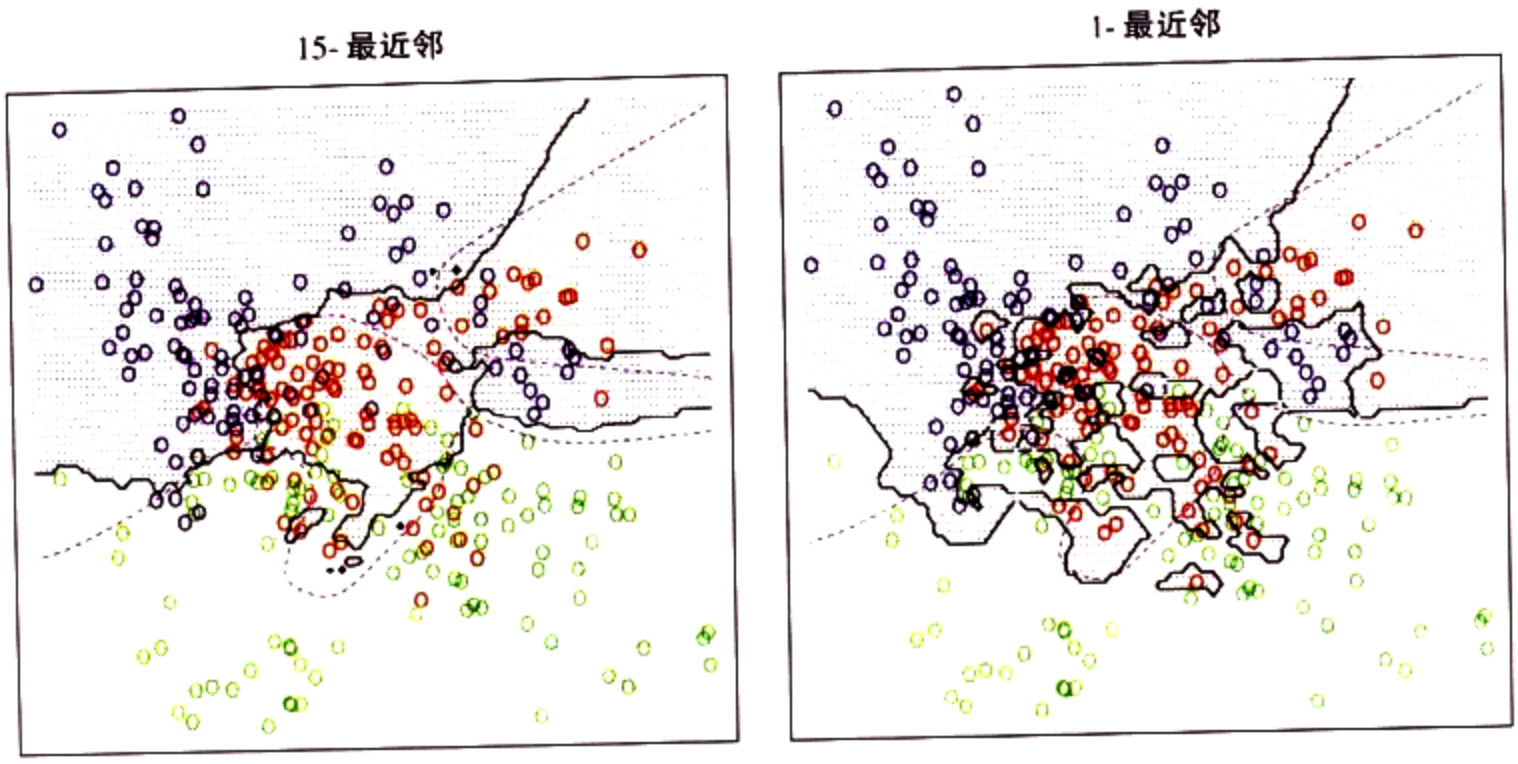


图 13.3

k- 最近邻分类器应用于图 13.1 的模拟数据。背景上的紫色虚线是贝叶斯判定边界

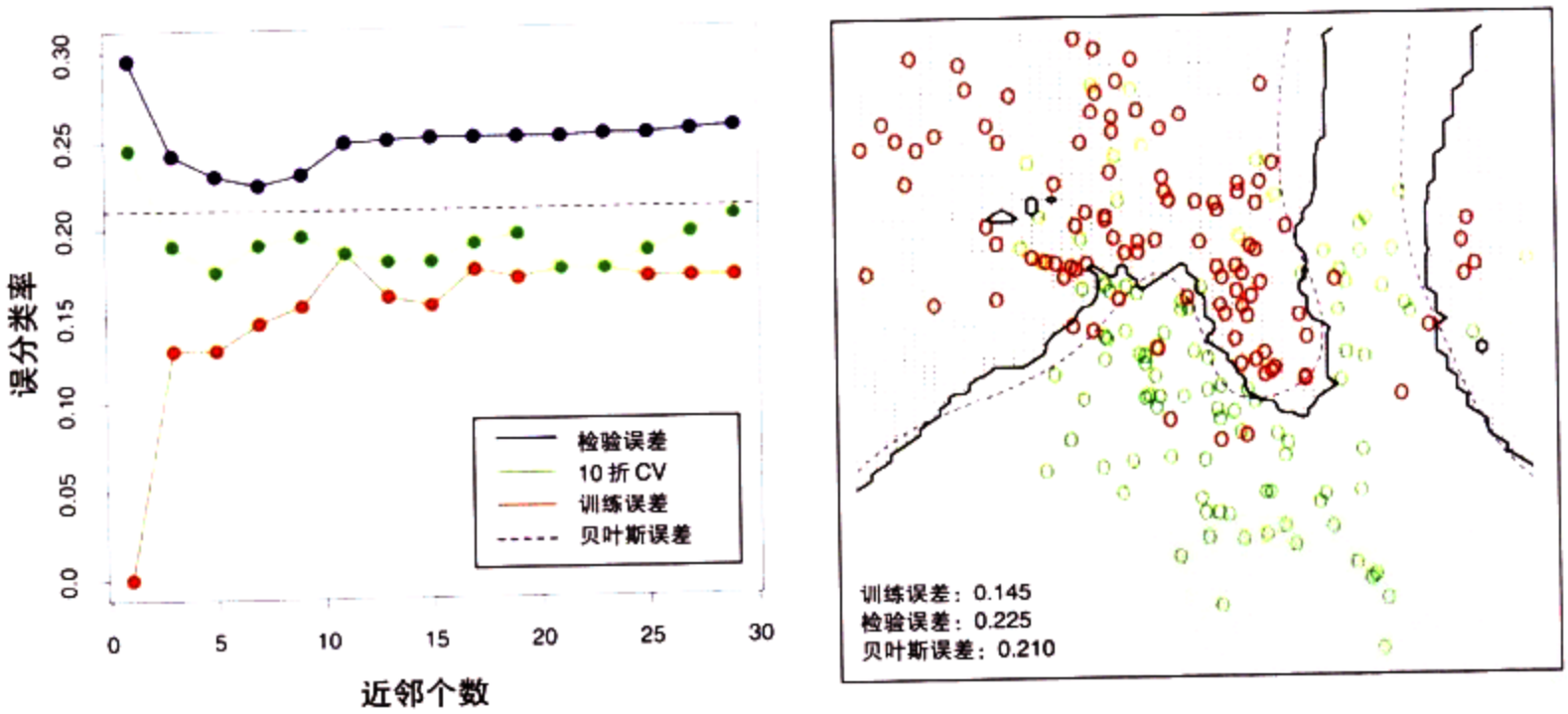


图 13.4

2- 类混合数据上的 *k*- 最近邻。左图显示误分类率，作为邻域大小的函数。右图显示 7- 最近邻的判定边界，关于极小化检验误差看上去它是最优的。背景中的紫色虚线是贝叶斯判定边界

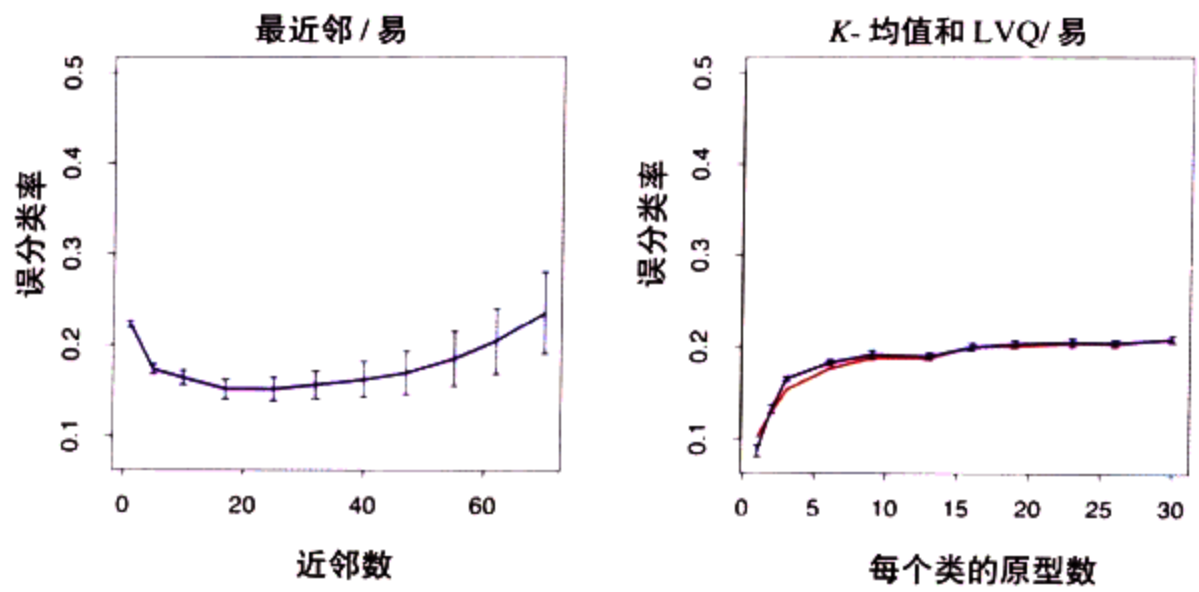


图 13.5

关于文中描述的两个模拟问题："易"和"难"问题，10 次实现上的最近邻，*K*-均值（蓝色）和 LVQ（红色）的误分类均值 ± 一个标准误差

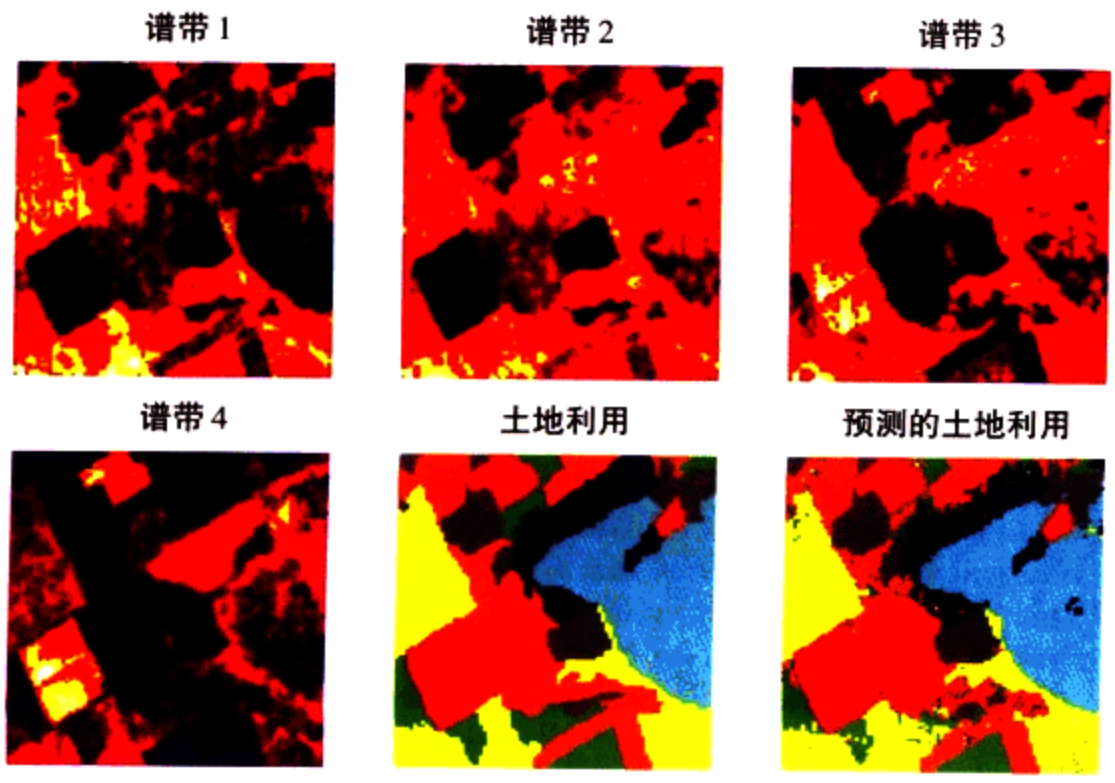


图 13.6

前 4 幅图是某农业区域 4 个谱带下的 LANDSAT 图像，用热度图描绘。其余两幅图给出实际的土地使用情况（彩色编码）和使用文中描述的5-最近邻规则预测的土地使用情况

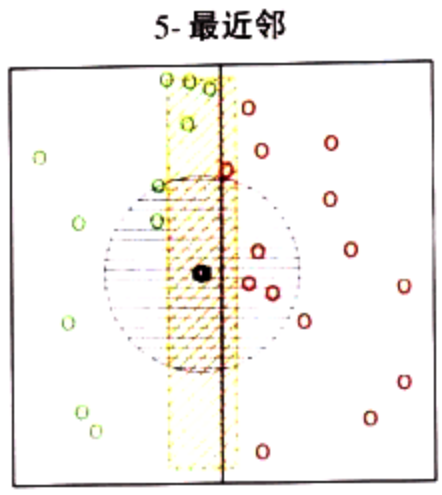


图 13.13

立方体上均匀分布的点，垂线将红色类和绿色类分隔开。垂直的带代表仅使用水平坐标发现目标点（实点）的最近邻的 5- 最近邻区域。球显示了使用两个坐标的 5- 最近邻区域，我们看到在这种情况下它已经伸展到红色类区域（而且该实例被错误类控制）

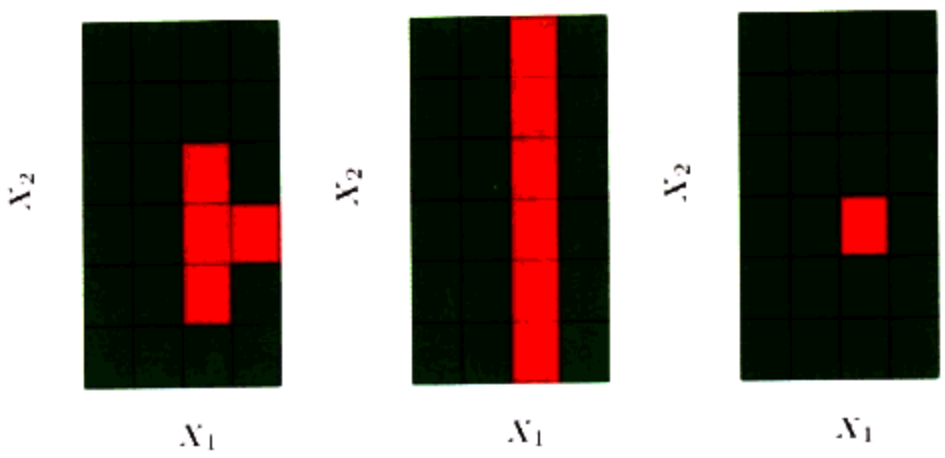


图 14.1

关联规则的简化。有两个输入 X_1 和 X_2，分别取 4 个和 6 个不同值。红色方块表示高密度区域。为了简化计算，我们假定导出的子集对应输入的一个值或所有值。在这个假定下，能够求得中间或右边的模式，但得不到左边的模式

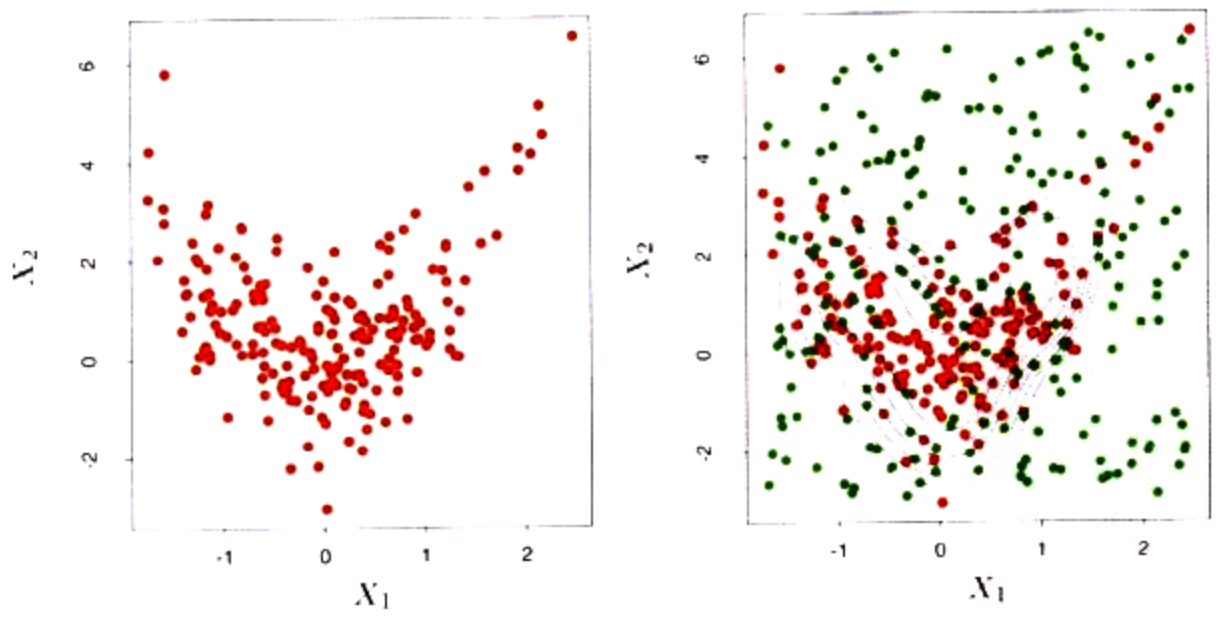


图 14.3

通过分类的密度估计。左图：200 个数据点的训练集。右图：训练集加上 200 个参考数据点，它们在包含训练数据的矩形框内均匀生成。训练样本标记为类 1，参考样本标记为类 0，并且用半参数化的逻辑斯缔回归模型拟合数据。图中显示了 $\hat{g}(x)$ 的一些等高线

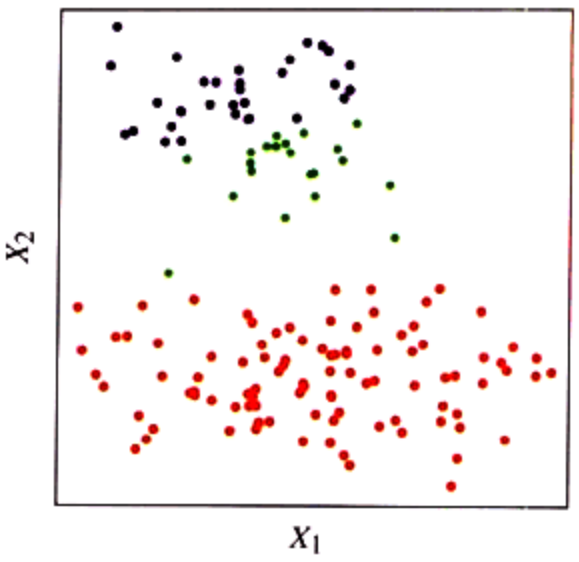


图 14.4

图中的模拟数据由 K-均值聚类算法聚类为三类（由红色、蓝色和绿色表示）

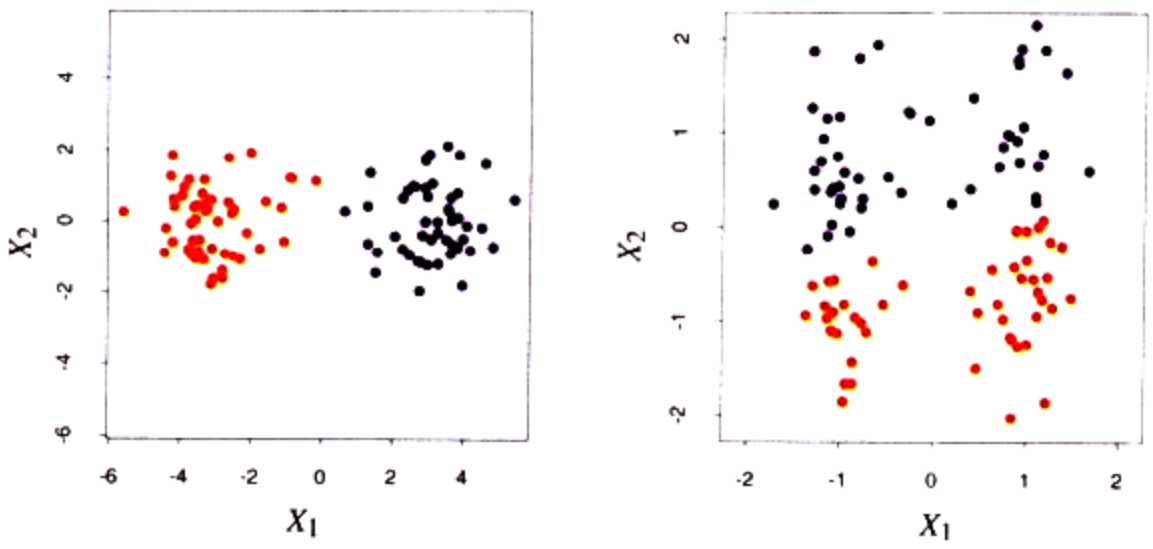


图 14.5

模拟数据：左图，对原始数据应用 K-均值聚类方法（K=2）。用两种颜色表明簇的成员。右图，在聚类前首先将特征做了标准化。这等价于使用特征权值 $1/[2 \cdot \mathrm{var}(X_j)]$。标准化使原来分割不错的组间界限模糊。注意，每幅图的横坐标和纵坐标使用相同的坐标单位

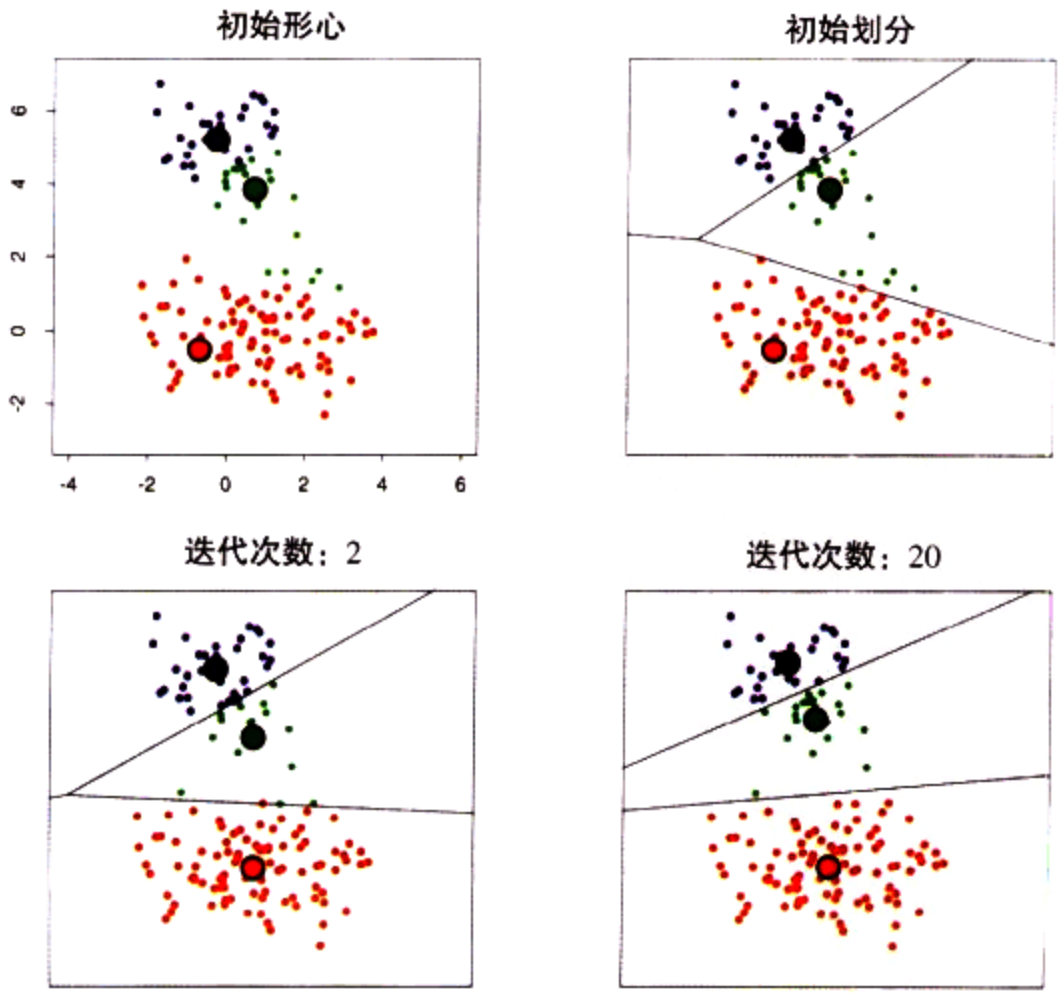


图 14.6

对图 14.4 中模拟数据，K-均值聚类算法的相继迭代

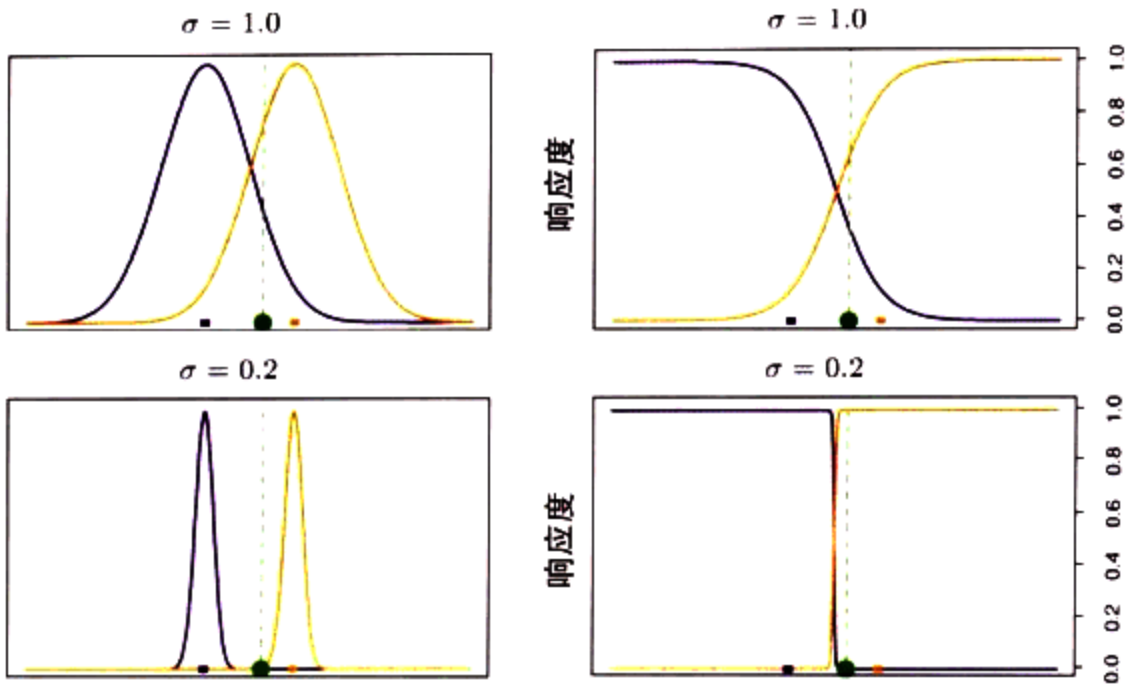


图 14.7

左图：实线是两个高斯密度 $g_0(x)$和 $g_1(x)$（蓝色和橙色），$x=0.5$ 处为单个数据点（绿色点）。彩色方块绘制在 $x=-1.0$ 和 $x=1.0$ 上，即在每个密度的均值上。右图：相对密度 $g_0(x)/(g_0(x)+g_1(x))$和 $g_1(x)/(g_0(x)+g_1(x))$，称做该数据点对每个簇的“响应度”。上图中高斯标准差 $\sigma=1.0$，下图中 $\sigma=0.2$。EM 算法使用这些“响应度”做每个数据点到两个簇中每一个的“软”指派。当 σ 相当大时，响应度可能接近 0.5（右上角的图中响应度为 0.36 和 0.64）。当 $\sigma\to 0$ 时，对于离目标点最近的簇中心，响应度→1，对于其他簇，响应度为 0。右下图中所示为“硬”指派

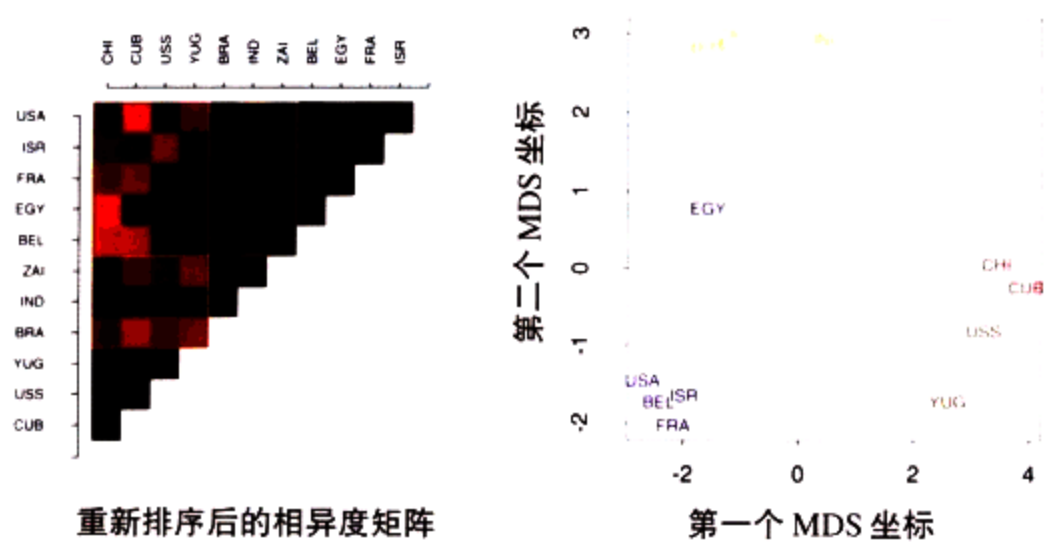


图 14.10

国家相异度的调查。左图：根据 3-中心点聚类排序并分组的相异度。热度图由最相似（深红色）到最不相似（浅红色）进行编码。右图：二维的多维定标图，3 种颜色表示 3- 中心点聚类的簇

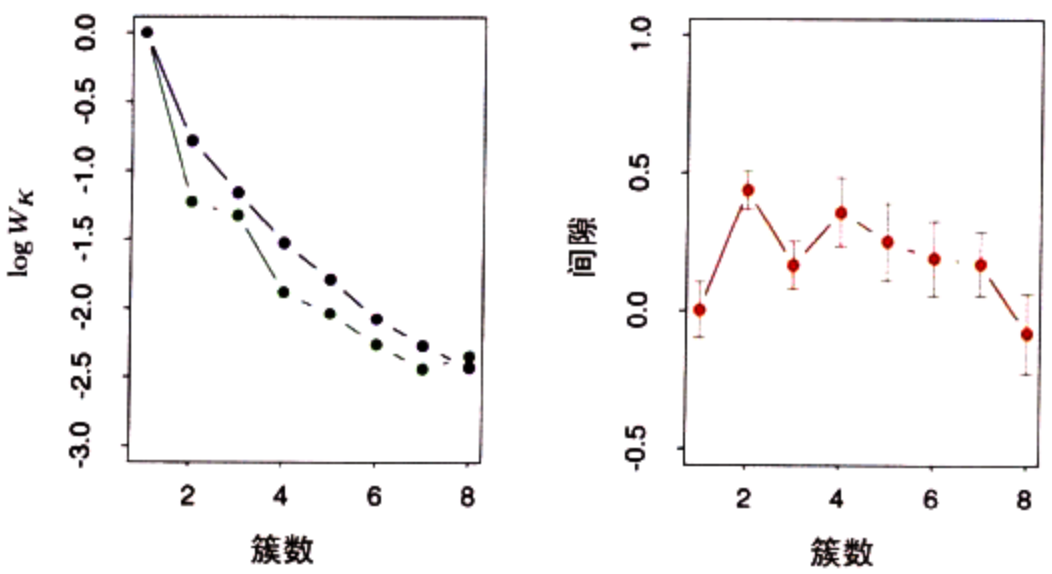


图 14.11

左图：对于图 14.4 中模拟数据，$\log W_k$ 的观测值（绿色）和期望值（蓝色）。两条曲线在 1 个簇时均等于 0。右图：间隙曲线，等于 $\log W_k$ 的观测值和期望值之差。间隙估计 K^* 是在最大值的一个标准差内产生间隙的最小 K 值；这里 $K^*=2$

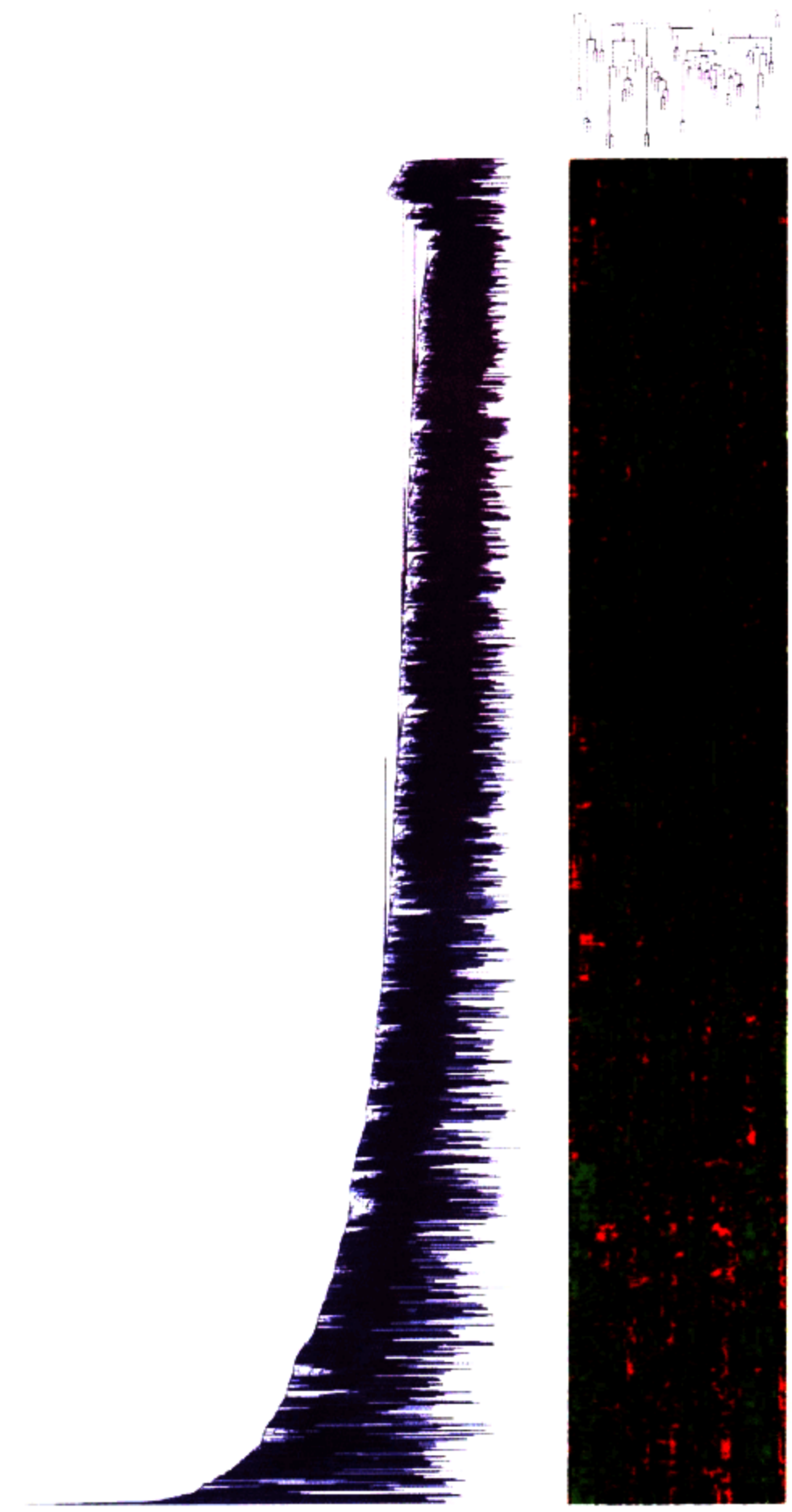


图 14.14

DNA 微阵列数据：平均连接分层聚类独立地应用于行（基因）和列（样本），以确定行和列的顺序。颜色由浅绿（阴性，低显性）变为浅红（阳性，高显性）

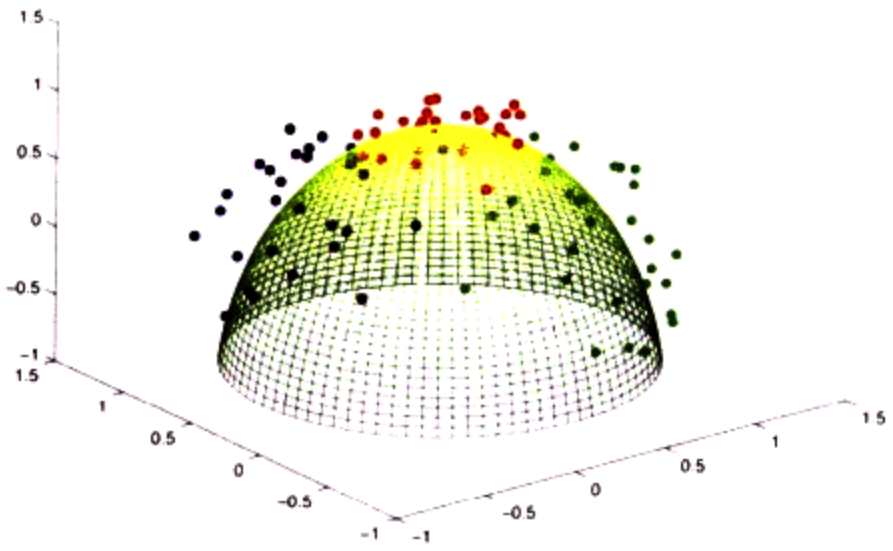


图 14.15

聚为三个类的模拟数据，接近一个半球面

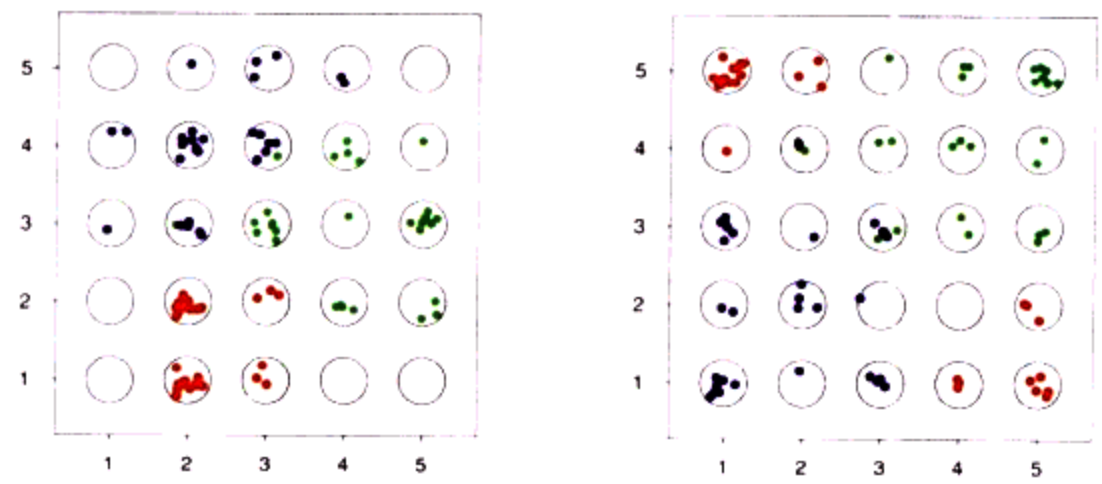


图 14.16

自组织映射用于半球面数据的例子。左图是初始格局，右图是最终结果。5 × 5 的原型网格由圆圈表示，投影到每个原型的点随机绘制在相应的圆圈内

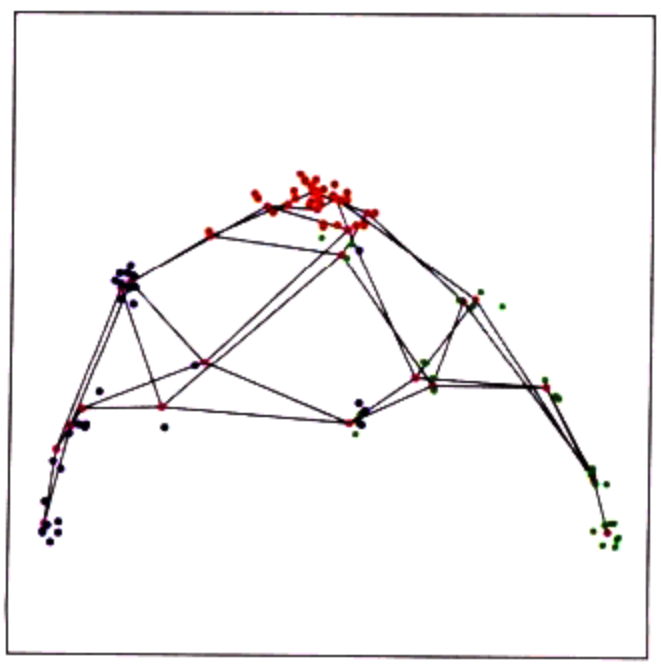


图 14.17

IR^3 中拟合SOM模型的线网表示。其中直线表示拓扑网格的水平边和垂直边，双线表示曲面对角地折回自身以模拟红色的点。聚类成员抖动以表示其颜色，紫色的点为节点中心

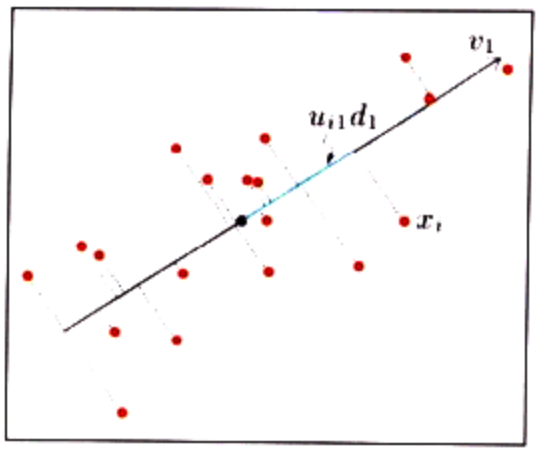
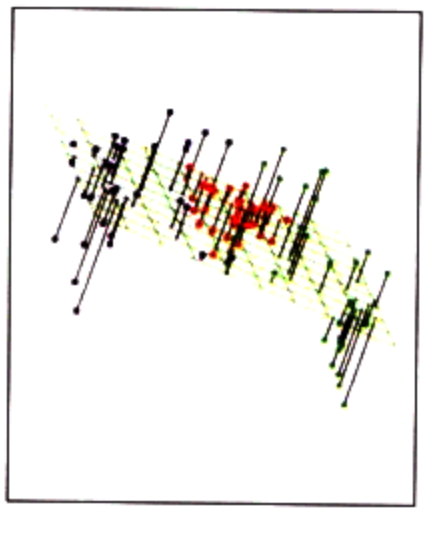


图 14.20

数据集的第一个线性主成分。该直线极小化每个点与其在该直线的正交投影的距离平方和



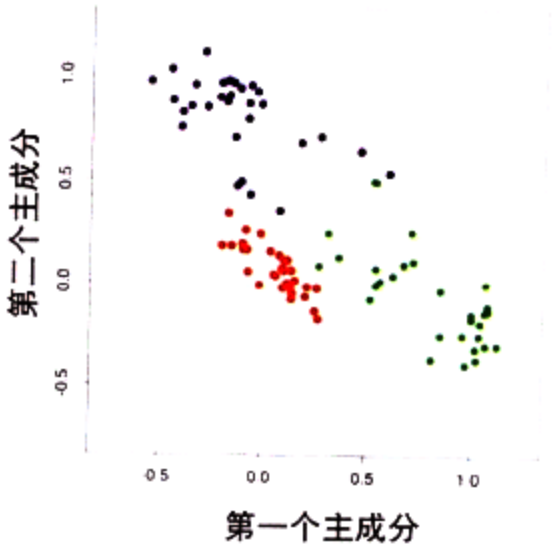
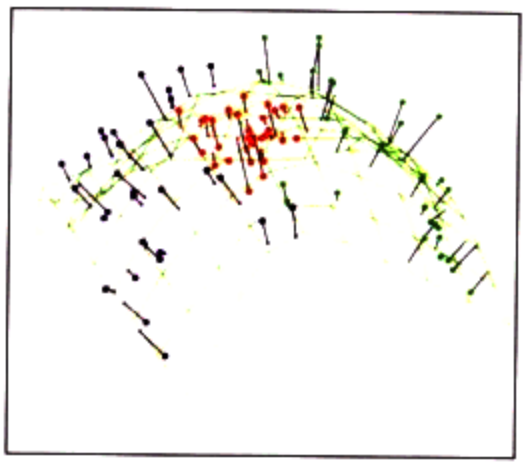


图 14.21

对半球面数据最好的秩 2 线性近似。右图显示投影点，其坐标由数据的前两个主成分 $\mathbf{U}_2\mathbf{D}_2$ 给出



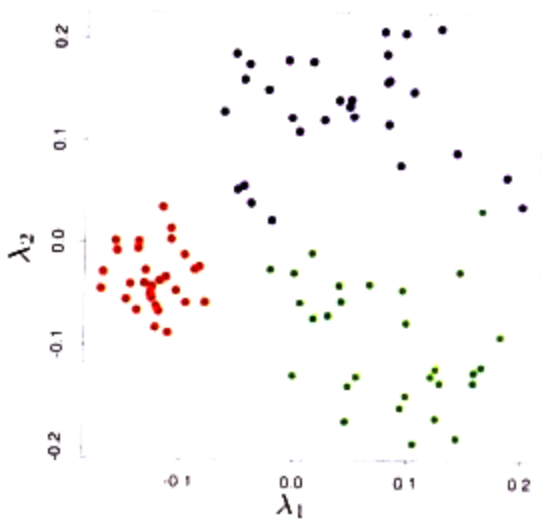


图 14.26

主曲面拟合半球面数据。左图：拟合的二维曲面。右图：数据点在曲面上的投影，结果在坐标 $\hat{\lambda}_1, \hat{\lambda}_2$ 上

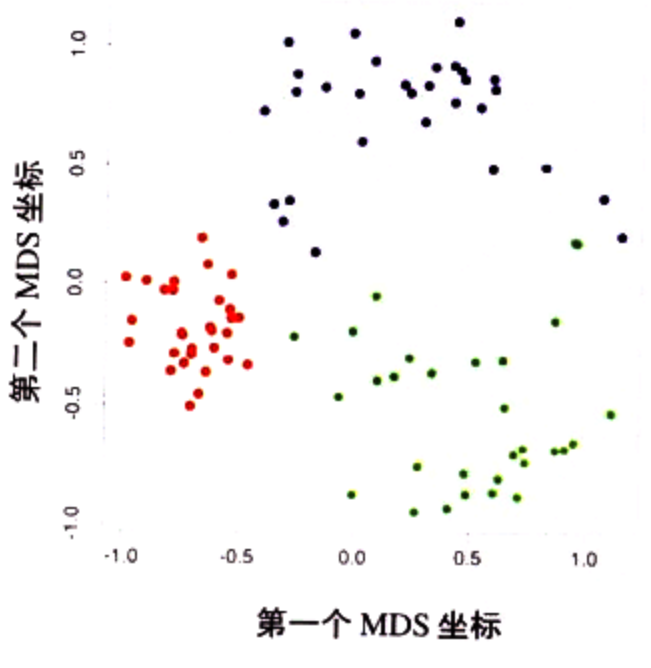


图 14.32

来自经典定标对半球面数据的前两个坐标

出 版 说 明

21世纪初的5至10年是我国国民经济和社会发展的重要时期，也是信息产业快速发展的关键时期。在我国加入WTO后的今天，培养一支适应国际化竞争的一流IT人才队伍是我国高等教育的重要任务之一。信息科学和技术方面人才的优劣与多寡，是我国面对国际竞争时成败的关键因素。

当前，正值我国高等教育特别是信息科学领域的教育调整、变革的重大时期，为使我国教育体制与国际化接轨，有条件的高等院校正在为某些信息学科和技术课程使用国外优秀教材和优秀原版教材，以使我国在计算机教学上尽快赶上国际先进水平。

电子工业出版社秉承多年来引进国外优秀图书的经验，翻译出版了“国外计算机科学教材系列”丛书，这套教材覆盖学科范围广、领域宽、层次多，既有本科专业课程教材，也有研究生课程教材，以适应不同院系、不同专业、不同层次的师生对教材的需求，广大师生可自由选择和自由组合使用。这些教材涉及的学科方向包括网络与通信、操作系统、计算机组织与结构、算法与数据结构、数据库与信息处理、编程语言、图形图像与多媒体、软件工程等。同时，我们也适当引进了一些优秀英文原版教材，本着翻译版本和英文原版并重的原则，对重点图书既提供英文原版又提供相应的翻译版本。

在图书选题上，我们大都选择国外著名出版公司出版的高校教材，如Pearson Education培生教育出版集团、麦格劳－希尔教育出版集团、麻省理工学院出版社、剑桥大学出版社等。撰写教材的许多作者都是蜚声世界的教授、学者，如道格拉斯·科默（Douglas E. Comer）、威廉·斯托林斯（William Stallings）、哈维·戴特尔（Harvey M. Deitel）、尤利斯·布莱克（Uyless Black）等。

为确保教材的选题质量和翻译质量，我们约请了清华大学、北京大学、北京航空航天大学、复旦大学、上海交通大学、南京大学、浙江大学、哈尔滨工业大学、华中科技大学、西安交通大学、国防科学技术大学、解放军理工大学等著名高校的教授和骨干教师参与了本系列教材的选题、翻译和审校工作。他们中既有讲授同类教材的骨干教师、博士，也有积累了几十年教学经验的老教授和博士生导师。

在该系列教材的选题、翻译和编辑加工过程中，为提高教材质量，我们做了大量细致的工作，包括对所选教材进行全面论证；选择编辑时力求达到专业对口；对排版、印制质量进行严格把关。对于英文教材中出现的错误，我们通过与作者联络和网上下载勘误表等方式，逐一进行了修订。

此外，我们还将与国外著名出版公司合作，提供一些教材的教学支持资料，希望能为授课老师提供帮助。今后，我们将继续加强与各高校教师的密切联系，为广大师生引进更多的国外优秀教材和参考书，为我国计算机科学教学体系与国际教学体系的接轨做出努力。

电子工业出版社

教材出版委员会

Preface of the Chinese Edition

We are very happy that our book has been translated into Chinese by Professors Fan, Chai and Zan. This means that our work has the possibility of reaching far more people than before—a prospect exciting for any scientist.

We have many Chinese speaking graduate students at Stanford Statistics, and they have assured us that these translation authors have done a very good job.

We take this opportunity to wish all our Chinese colleagues well, and hope they find this text useful.

We also can only hope that our book gets as warm a welcome in the East as it has done in the West.

With best wishes from Trevor Hastie, Rob Tibshirani and Jerome Friedman

Stanford, October 2003

中译本序

我们的书被范明教授、柴玉梅副教授和昝红英讲师翻译成中文，我们感到非常高兴。这意味我们的工作将有机会被更多人所了解——对于任何科学家，这都是令人期待和兴奋的。

在斯坦福大学统计学系，我们有许多讲中文的研究生，他们使我们确信几位译者的翻译非常出色。

借此机会，我们向所有的中国同仁问好，并希望他们喜欢本书。

热切期待我们的书在东方也能像在西方一样受到热烈欢迎。

致以良好祝愿！

Trevor Hastie, Rob Tibshirani, Jerome Friedman

2003 年 10 月于斯坦福

译 者 序

数据挖掘是一个多学科交叉领域，涉及数据库技术、机器学习、统计学、神经网络、模式识别、知识库、信息提取、高性能计算等诸多领域，并在工业、商务、财经、通信、医疗卫生、生物工程、科学等众多行业得到广泛的应用。在我们已经拥有收集、存储、查询大量数据的手段之后，理解这些数据、从大量数据中发现有用的知识自然成为各行各业的需要，同时也向各领域的研究者提出了新的挑战。需要是发明之母，是科学研究和科学发现的源泉。直面挑战是研究者的本能，也是激发科学研究的巨大动力。

基于知识背景，我们是从数据库或机器学习进入数据挖掘领域的。译者之一曾翻译过 Jiawei Han 和 Micheline Kamber 所著的《数据挖掘：概念与技术》一书。我们为数据挖掘领域的进展感到鼓舞，也在试图为推进数据挖掘的进展贡献微薄之力。随着对数据挖掘理解和研究的深入，我们越来越感到知识海洋的浩瀚。一个问题不断在我们的脑海中浮现：数据挖掘的数学或统计学基础是什么？

曾有几位学者向我们推荐过本书，我们也想认真读一读，但一直未能如愿。当电子工业出版社的编辑希望我们翻译一本关于数据仓库的著作时，我们表示对“The Elements of Statistical Learning: Data Mining, Inference, and Prediction”一书更感兴趣。真是无巧不成书！电子工业出版社买下了这本专著的中文版版权，并在物色翻译人员。当问起是否愿意翻译时，我们欣然同意。

收到本书英文版后，我们迫不及待地打开，真想一口气读完。然而，粗略浏览之后，我们倒吸了一口凉气。放在面前的是斯坦福大学三位统计学家 Trevor Hastie, Robert Tibshirani 和 Jerome Friedman 的力作。想读这本书是一回事，而翻译完全是另一回事。一时间，我们后悔不迭。然而，瓷器活是揽下了，有无金刚钻都只好一试。

用“临时抱佛脚”来形容我们翻译前的准备工作再恰当不过了。我们迅速浏览了一些概率论与数理统计和统计学方面的书籍，并多方寻求帮助；与此同时，我们开始通读原著。最初，进行本书的翻译有些像“逼上梁山”。然而，随着翻译工作的进展，我们才真正感到这是一种享受，因为我们已经被这本书深深地吸引。我们从书中学到了许多，并且可以肯定地说，这本书对我们未来的研究必将产生重要影响。

正如作者所言，本书试图将学习领域中许多重要的新思想汇集在一起，并且在统计学的框架下解释它们。然而，作者强调的是方法和它们的概念基础，而不是数学性质。因此，读者只要学过一门统计学的基础课程，涵盖包括线性回归在内的基本内容，阅读本书就不会有太大的困难。

本书特别适合从事数据挖掘和机器学习研究的读者阅读。尽管作者是统计学家，但他们在过去八年中一直参加神经网络、数据挖掘和机器学习会议，他们的统计学观点可以帮助读者从不同角度更好地理解学习。

全书共 14 章。第 1 章到第 6 章和第 7 章的第 7.1 节至第 7.3 节由范明翻译，昝红英校对；第 7 章其余部分和第 8 章到第 13 章由柴玉梅和王黎明翻译及校对；第 14 章由昝红英翻译，范

明校对。陈国勋认真阅读了全部译文初稿，规范了专业术语的译法并订正了一些错误。范明通读全部译稿，并最后定稿。译者还参照本书 Web 页提供的勘误表，对书中的印刷错误和疏漏进行了更正。

译者感谢电子工业出版社的工作者。在许多出版社都忙于出版“畅销书”的时候，他们坚持引进名著，是他们的远见使得本书中文版能够及时与读者见面。

译者感谢本书的三位作者 Trevor Hastie，Robert Tibshirani 和 Jerome Friedman 教授为中译本撰写序言。当请他们为中译本写序时，他们欣然同意，并坚持要在中译本出版之前先阅读部分译稿。我们按照作者的要求寄去部分章节的译稿，50 天后 Hastie 发来了他们为中译本写的序。当看到三位作者在中译本序中对译文的评价“We have many Chinese speaking graduate students at Stanford Statistics, and they have assured us that these translation authors have done a very good job.”后，我们感觉受到了最高褒奖，同时也被三位作者一丝不苟的科学精神所折服。

这里向为本书翻译做出贡献的所有人表示感谢。这是一本统计学习专著，不仅涉及统计学，而且涉及机器学习、数据挖掘。书中的例子更是取材广泛，涉及医学、生命科学、电子、语音识别等众多领域。郑州大学林诒勋教授、施仁杰教授、陈绍春教授和北京大学计算语言学研究所的于江生博士对一些数学术语的译法提出了宝贵建议；郑州大学医学院张雪培、戴丽萍博士为医学术语的翻译提供了帮助。译者的一些学生也分别阅读了部分译稿，提出了一些有益的建议。还要感谢我们的家人，感谢他(她)们的理解与支持。

作为一本交叉学科的专著，在翻译的过程中，时常需要面对新的知识。尽管我们反复讨论、多次修改，力求译文准确，但仍难免出现差错。此外，由于译者水平有限，译文中的不当之处也在所难免。译文中的错误当然应当由译者负责。但我们真诚地希望同行和读者不吝赐教。如果能把你的意见和建议发往 mfan@zzu.edu.cn，我们将不胜感激。

译　者

2003 年 6 月于郑州大学

前　言

我们被信息淹没，但却缺乏知识。

——Rutherford D. Roger

统计学领域不断受到来自科学界和产业界问题的挑战。早期，这些问题通常来自农业和工业实验，且规模相对较小。随着计算机和信息时代的到来，统计问题的规模和复杂性都有了急剧的增加。数据存储、组织和检索领域的挑战导致一个新领域“数据挖掘”的产生；生物和医学方面的统计和计算问题开创了“生物信息学”。许多领域都产生了海量数据，而统计学家的工作就是理解这些数据：提取重要的模式和趋势，理解这些数据“说瞬么”。我们称此为：从数据中学习。

从数据中学习的难题引发了统计科学的革命。由于计算扮演了重要角色，毫不奇怪，许多成果都是由计算机科学和工程学等其他领域的研究者做出的。

我们考虑的学习问题可以粗略地分为有指导的和无指导的。对于有指导学习，目标是根据一些输入度量预测一个结果度量值。对于无指导学习，没有结果度量，其目标是描述输入度量集合中的关联和模式。

在本书中，我们试图将学习领域中许多重要的新思想汇集在一起，并且在统计学的框架下解释它们。尽管有些数学细节是必要的，但我们强调的是方法和它们的概念基础，而不是理论性质。我们希望本书不仅能吸引统计学家，而且能吸引更广泛领域的研究者和实践者。

正如从统计学之外的研究者那里学到了许多知识一样，我们的统计学观点也可以帮助其他人更好地理解学习的不同方面。

任何事物都没有真正正确的解释，解释是为人们理解而服务的一种媒介。解释的价值是使得他人可以更富有成果地思考。

——Andreas Buja

这里要向为本书的构思和完成做出贡献的所有人员表示感谢。David Andrews，Leo Breiman，Andreas Buja，John Chambers，Bradley Efron，Geoffrey Hinton，Werner Stuetzle 和 John Tukey 对我们的工作具有重要影响。Balasubramanian Narasimhan 为我们提出了许多建议，在一些计算问题上给予了帮助，并维护了一个良好的计算环境。Shin-Ho Bang 帮我们绘制了大量的图形。Lee Wilkinson 为彩图绘制提出了宝贵意见。

Trevor Hastie
Robert Tibshirani
Jerome Friedman
斯坦福，加利福尼亚
2001 年 5 月

恬静的统计学家改变了我们的世界；不是通过发现新的事实或者开发新技术，而是通过改变我们的推理、实验和观点的形成方式……

——Ian Hacking

目　录

第1章 绪 论

统计学习在科学、财经和工业等许多领域都起着至关重要的作用。下面是一些学习问题的例子：

- 预测一个因心脏病发作而住院的病人是否会再次心脏病发作。这种预测基于人口统计、饮食和对该病人的临床检查。
- 根据公司的业绩和经济学数据，预测今后6个月的股票股价。
- 从数字化的图像，识别手写的邮政编码中的数字。
- 根据患者血液的红外线光谱，估计糖尿病患者血液中的葡萄糖含量。
- 根据临床和人口统计学变量，确定前列腺癌风险因素。

学习科学在统计学、数据挖掘和人工智能等领域起着关键的作用，同时也与工程学和其他学科有交叉。

本书介绍从数据中学习。典型地，有结果度量，通常是量化的(如股票价格)或分类的(如心脏病发作或不发作)，我们希望根据一组特征(feature)(如饮食和临床检查)对其进行预测。假设有训练数据集(training set of data)，借此观察对象集(如人)的结果和特征度量。使用这些数据建立预测模型或学习器(learner)，使我们可以预测新的未知对象的结果。一个好的学习器可以精确地预测这种结果。

上面描述的例子称为有指导学习(supervised learning)问题。之所以称它为“有指导的”，是因为有结果变量指导学习过程。在无指导学习(unsupervised learning)问题中，只能观察特征，而没有结果度量。我们的任务只是描述数据组织或聚类的方式。本书大部分讨论的是有指导学习；关于无指导学习问题的研究不多，仅在最后一章介绍。

下面是本书讨论的实际学习问题的一些例子。

例1 垃圾邮件

本例的数据包括4601封电子邮件信息，研究预测电子邮件是否为垃圾邮件。目标是设计一个垃圾邮件自动检测器，在把邮件放进用户信箱之前过滤掉垃圾邮件。对于所有4601封电子邮件，都知道真实结果(电子邮件类型)email或spam，以及电子邮件中最常出现的57个词和标点符号的相对频率。这是一个有指导学习问题，结果类变量为email/spam。该问题也称分类(classification)问题。

表1.1列出了单词和字符，并显示了它们在email和spam之间的最大平均差异。

表1.1 电子邮件信息中指定的单词或字符的平均百分比。选取显示email和spam之间差别最大的单词和字符

	george	you	your	hp	free	hpl	!	our	re	edu	remove
spam	0.00	2.26	1.38	0.02	0.52	0.01	0.51	0.51	0.13	0.01	0.28
email	1.27	1.27	0.44	0.90	0.07	0.43	0.11	0.18	0.42	0.29	0.01

我们的学习方法必须决定使用哪些特征以及如何使用。例如，可以使用下面的规则：

```
if (%george < 0.6) & (%you > 1.5)   then spam
                                    else email
```

规则的另一种形式可以是：

```
if (0.2 * %you - 0.3 * %george) > 0   then spam
                                      else email
```

对于该问题，并非所有错误都是等同的。我们想避免过滤掉好的电子邮件，尽管使垃圾邮件通过不是所希望的，但也不会导致太大问题。本书将讨论处理该学习问题的多种不同方法。

例 2　前列腺癌

该例的数据如图 1.1 所示，取自 Stamey 等人(1989)的研究。该研究考察了 97 位准备做前列腺根治术病人的前列腺特殊抗原(prostate specific antigen，PSA)水平与一些临床指标之间的相关性。

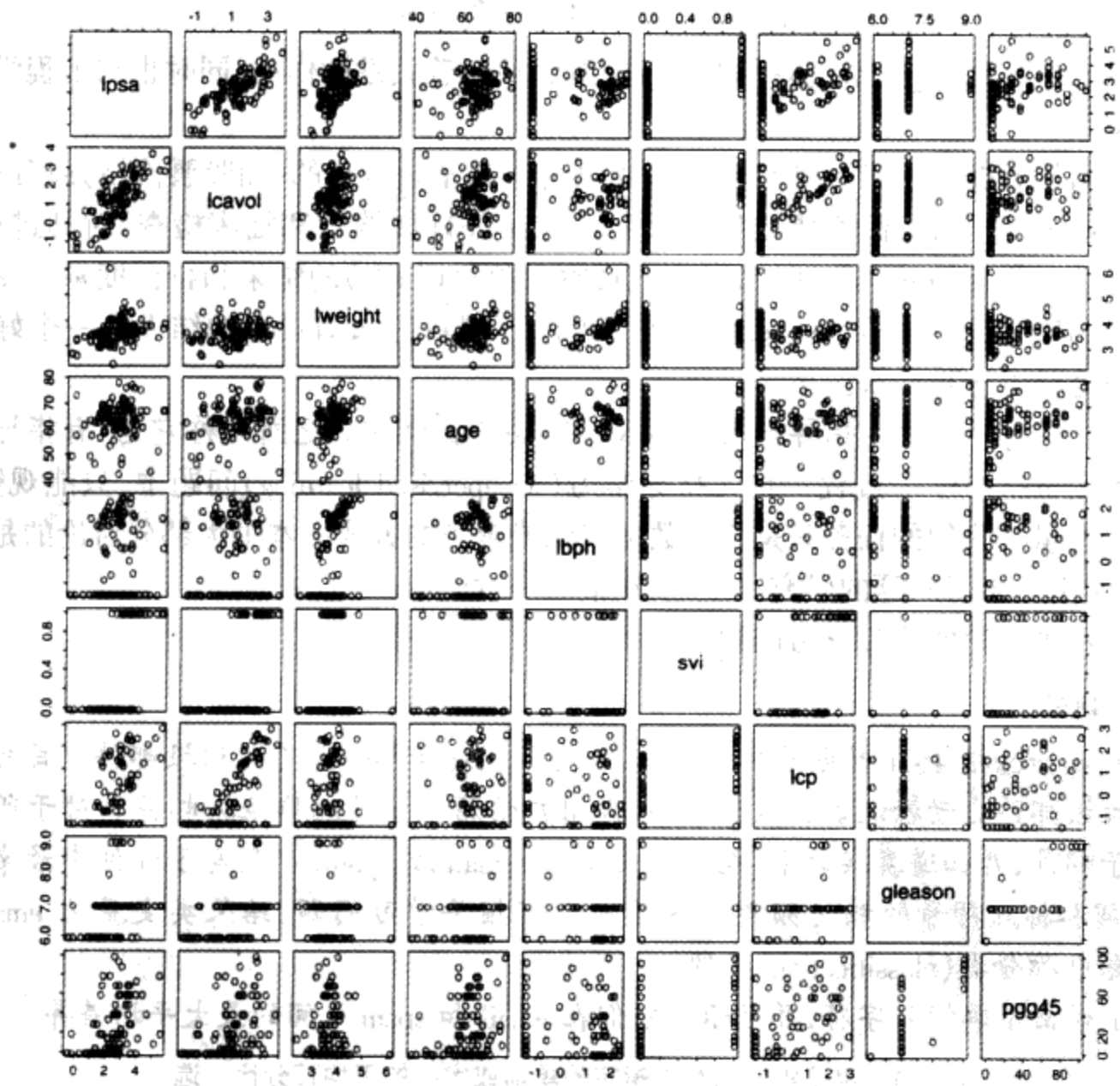


图 1.1　前列腺癌数据的散点图矩阵。第一行依次显示对每个预测子的响应。其中 svi 和 gleason 两个预测子是分类的

本例的目的是从一些指标预测 PSA 的记录值 lpsa。指标包括肿瘤体积记录值 lcavol、前列腺重量记录值 lweight、年龄 age、良性前列腺增生量 lbph、精囊浸润 svi、包膜穿透记录值 lcp、Gl-

eason 积分 gleason 和 Gleason 积分 4 或 5 所占的百分比 pgg45。图 1.1 是这些变量的散点图矩阵。一些指标与 lpsa 的相关性是显而易见的,但是靠肉眼构造一个好的预测模型很困难。这是一个有指导学习问题,称为回归问题(regression problem),因为输出度量是定量的。

例 3 手写体数字识别

该例的数据取自美国邮政信封上手写的邮政编码。每幅图像从一个五位的邮政编码截取,隔离成单个数字。图像是 16×16 的八位灰度图,每个点的亮度从 0 到 255。一些样本图像在图 1.2 中给出。

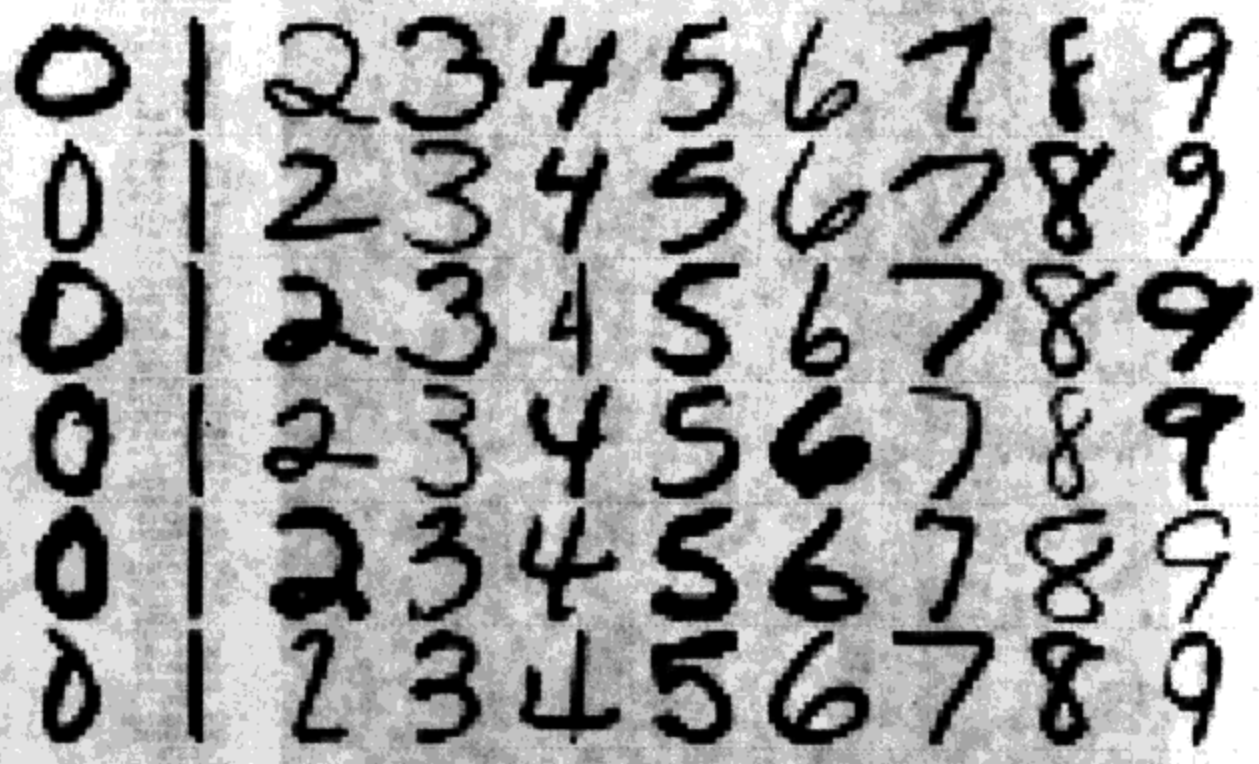


图 1.2 取自美国邮政信封的手写体数字

图像已被规范化,使它们具有大致相同的尺寸和方位。本例的任务是从 16×16 的点亮度矩阵,快速准确地识别图像{0,1,…,9}。如果足够准确,结果算法就可以用于信件自动分拣过程。这是一个分类问题,要求错误率很低,以避免邮件误投。为了获得低误差率,有些对象可以归入“不知道”类,并通过手工分拣。

例 4 DNA 表达微阵列分析

DNA 代表脱氧核糖核酸,是构成人类染色体的基本物质。通过测定出现在细胞某基因中的 mRNA(信使核糖核酸)总量,可用 DNA 微阵列来测量细胞中的基因表达。微阵列是生物学的一项突破性技术,便于同时对细胞单个样本中数以千计的基因进行定量研究。

下面介绍 DNA 微阵列如何工作。数千基因的核苷酸序列印在一个玻璃片上。一个目标样本和一个参照样本用红绿染色标记,并均与玻璃片上的 DNA 杂交。通过荧光检查器,测量每个位点上 RNA 的记录(红/绿)强度。结果是数千个值,通常在 -6~6 之间,测量目标样本中每个基因相对于参照样本的表达水平。正值表示目标样本的表达水平高于参照样本的表达水平,负值相反。

基因表达数据集将一系列 DNA 微阵列实验的表达值收集在一起,每一列代表一个实验。因此,有数千行代表个体基因,数十列代表样本。在图 1.3 的特例中,有 6830 个基因(行)和 64 个样本(列),为了简洁,只显示了 100 行随机选样。该图以热度图(heat map)的形式显示数据,颜色变化从绿(负)到红(正)。样本取自不同病人的 64 个癌瘤。

图 1.3 DNA 微阵列数据：人体瘤数据 6830 个基因(行)和 64 个样本(列)的表达水平矩阵。只显示100行的随机选样。显示的是热度图，从鲜绿(负，低显性)到鲜红(正，高显性)。遗漏的值为灰色。行和列以随机次序显示(见彩页)

我们面对的挑战是理解基因和样本是如何组织的。典型的问题包括：

(a) 根据基因的表达图解，哪些样本最相似？

(b) 根据样本的表达图解，哪些基因最相似？

(c) 对于某些癌样本，某些基因显示很高(或很低)的表达水平吗？

可以将该任务看做回归问题，它具有两个分类预测变量——基因和样本；响应变量是表达

水平。然而，将它视为无指导学习问题可能更有用。例如，对于上面的问题(a)，想像样本为 6830 维空间中的点，我们要按某种方法对它们进行聚类(cluster)。

读者对象

本书是为众多领域的研究者和学生编写的。这些领域包括：统计学、人工智能、工程学、财经和其他一些领域。假定读者至少已经学过一门统计学的基础课程，涵盖包括线性回归在内的基本内容。

我们并不试图囊括全部学习方法，而是介绍一些最重要的技术。同样值得注意的是，本书着重介绍基本概念和思想，借此研究者可以评价学习方法。我们试图以直观的风格写这本书，强调的是概念而不是数学细节。

作为统计学家，本书的表述自然反映我们的背景和专业领域。然而，在过去的八年中，我们一直参加神经网络、数据挖掘和机器学习会议，并且思想深受这些令人激动的领域的影响，这种影响在我们的研究和本书中随处可见。

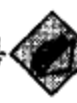
本书组织

我们的观点是：在试图掌握更复杂的方法之前，必须理解简单的方法。因此，在第 2 章给出有指导学习的概览之后，在第 3 章和第 4 章讨论回归和分类的线性方法。在第 5 章，介绍了样条函数、小波和单个预测器的正则/罚(regularization/penalization)方法。第 6 章涵盖核方法(kernel method)和局部回归。这两组方法都是构建高维学习技术的构件。模型的评估和选择是第 7 章的主题，涵盖偏倚和方差、过分拟合(overfitting)概念和选择模型的交叉验证方法。第 8 章讨论模型推理和平均，包括最大似然概述、贝叶斯推理和自助法(bootstrap)、EM 算法、Giblbs 选样和装袋(bagging)。一个相关的过程称为提升(boosting)，是第 10 章的重点。

从第 9 章到第 13 章，介绍了一系列有指导学习的结构化方法，其中第 9 章和第 11 章涵盖回归，第 12 章和第 13 章集中讨论分类。最后，在第 14 章介绍无指导学习方法。

在每章的结尾，我们都讨论对数据挖掘应用十分重要的计算考虑，包括计算规模如何随观测和预测量增加而变化。每章都以文献注释结束，给出材料的引用背景。

我们建议首先顺序阅读第 1 章到第 4 章。第 7 章也是必读的，它涵盖了与所有学习方法相关的核心概念。除此之外，本书的其余部分可以顺序或选择性地阅读，取决于读者的兴趣。

符号  指出技术上的难点段，可以跳过而不影响讨论。

本书网站

本书的网址是 http://www-stat.stanford.edu/ElemStatLearn，该网站提供了大量资源，包括本书使用的许多数据库。

写给教师

我们已经成功地使用本书作为一个两季度课程的基本内容；如果再添加一些辅助材料，甚至可以用于三季度的系列课程。每章的最后提供了习题。重要的是让学生找到好的软件工具解决这些问题。我们在教学中使用 S-PLUS 程序设计语言。

第 2 章　有指导学习概述

2.1　引言

第 1 章介绍的前三个例子有一些共同点：每个例子都有一个可以看做输入的变量集，它们被度量或预置；这些输入对一个或多个输出有影响；每个例子的目标都是使用输入来预测输出的值。这称为有指导的学习。

我们使用了更现代的机器学习语言。在统计学文献中，通常称输入为预测子(predictor)；该术语将与输入替换使用。经典地，称输入为独立变量(independent variable)，称输出为响应(response)，或更经典地，称输出为依赖变量(dependent variable)。

2.2　变量类型和术语

这些例子中的输出本质上是不同的。在预测葡萄糖含量的例子中，输出是定量的(quantitative)度量；其中一些度量值比其他值大，并且度量值相近的在本质上也相近。在 R. A. Fisher 提出的著名的艾里斯(Iris)判别例子中，输出是定性的(qualitative)(艾里斯的种族)，并假定值取自有限集$\mathcal{G} = \{Virginica, Setosa \text{ 和 } Versicolor\}$。在手写数字的例子中，输出是 10 个不同数字类$\mathcal{G} = \{0,1,\cdots,9\}$中的一个。这两个例子中，类中都没有显式的序，并且实际上通常用描述性的标记，而不是用数来记这些类。定性变量通常称为分类(categorical)或离散(discrete)变量，也称因素(factor)。

对于两种类型的输出，考虑使用输入来预测输出是有意义的。给定今天和昨天的大气测量，我们想预测明天的臭氧层状况。给定手写数字的数字化图像点的灰度值，预测它的类标号。

输出变量类型的差异引发对预测任务的命名约定：预测定量输出称为回归(regression)，而预测定性输出称为分类(classification)。我们将看到，这两类任务具有许多共同点。特殊地，它们都可以看做函数逼近任务。

输入也有不同的度量类型，每个都可以有定量的和定性的输入变量。这些也造成所用预测方法类型上的差别：有些方法明显最适合定量输入，有些最适合定性输入，而有些同时适合二者。

第三种变量类型是有序分类的(ordered categorical)，如 small，medium 和 large。这里，值之间有序，但不希望有度量(medium 和 small 之间的差不必与 large 和 medium 之间的差相同)。这些将在第 4 章进一步讨论。

典型地，定性变量用数值编码刻画。最简单的情况是只有两个类，如“成功”或“失败”，“存活”或“死亡”。这些常常用单个二进位数字 0 和 1 表示，或者用 -1 和 1 表示。由于很快就会介绍到的原因，这种数字编码有时称为目标(target)。当类多于两个时，可有多种选择。最有

用和最常用的编码是通过哑变量(dummy variable)。这里，K 级定性变量用 K 个二元变量或二进位的向量表示，该向量一次只有一位被“置位”。尽管有更多的压缩编码模式可用，但是哑变量在因素级是对称的。

典型地，我们用符号 X 表示输入变量。如果 X 是向量，则其分量可以用 X_j 访问。定量的输出变量用 Y 表示，定性输出用 G(表示组)表示。引用变量的整体时，我们使用大写字母，如 X、Y 或 G。观测值用小写字母表示；因此，X 的第 i 个观测值记做 x_i(这里，x_i 也是标量或向量)。矩阵用粗体大写字母表示。例如，N 个输入 p 向量 $x_i(i = 1, \cdots, N)$的集合将用 $N \times p$ 的矩阵 $\mathbf{X}$ 表示。通常，向量不用粗体，除非它们有 N 个分量；这种约定将第 i 个观测值的输入 p 向量 x_i 与变量 X_j 上所有观测值组成的 N 向量 x_j 相区别。由于假定所有向量都是列向量，所以 $\mathbf{X}$ 的第 i 行是向量 x_i 的转置 x_i^T。

现在，学习任务可以不严格地表述为：给定输入向量 X 的值，对输出 Y 的值做出一个好的预测，记为 $\hat{Y}$(读做 y 帽)。如果 Y 在 IR 中取值，则 $\hat{Y}$ 也如此；同样，对于分类输出，$\hat{G}$ 也应当在与 G 相关联的同样集合$\mathcal{G}$上取值。

对于 2 -类变量 G，一种方法是将该二元编码目标记为 Y，然后把它按定量输出对待。预测 $\hat{Y}$ 将落在[0, 1]中，而我们可以根据是否有$\hat{y} > 0.5$ 来设置 $\hat{G}$ 的类标号。这种方法可以拓广到 K 级定性输出。

我们需要数据来构造预测规则，通常需要大量数据。假定有观测值的集合(x_i, y_i)或$(x_i, g_i), i = 1, \cdots, N$，称为训练数据(training data)。使用它们，构造预测规则。

2.3　两种简单预测方法：最小二乘方和最近邻法

本节详细讨论了两种简单但有效的预测方法：使用最小二乘方的线性模型拟合和 k -最近邻预测规则。线性模型对结构做了大量假定，并产生稳定但可能不精确的预测。k -最近邻对结构做了适度的假定，其预测常常是精确的，但可能不稳定。

2.3.1　线性模型和最小二乘方

在过去的 30 年中，线性模型一直是统计学的主要支柱，并且现在依然是我们最重要的工具之一。给定一个输入向量 $X = (X_1, X_2, \cdots, X_p)$，通过以下模型来预测输出 Y：

$$\hat{Y} = \hat{\beta}_0 + \sum_{j=1}^{p} X_j \hat{\beta}_j \tag{2.1}$$

项 $\hat{\beta}_0$ 是截距，在机器学习中也称偏置(bias)。通常，在 X 中包含一个常数变量 1，在系数向量 $\hat{\beta}$ 中包含 $\hat{\beta}_0$ 是方便的。这样，向量形式的线性模型可以写成内积：

$$\hat{Y} = X^T \hat{\beta} \tag{2.2}$$

其中，X^T 表示向量或矩阵的转置(X 是列向量)。这里对单个输出建模，因此 $\hat{Y}$ 是标量。一般来说，$\hat{Y}$ 可以是 K 向量。在这种情况下，β 是 $p \times K$ 的系数矩阵。在 $(p+1)$维输入 - 输出空间中，$(X, \hat{Y})$表示一个超平面。如果 X 中包含常量，则超平面包含原点，是一个子空间。如果 X 不含常量，则超平面是一个仿射集，切 Y 轴于点 $(0, \hat{\beta}_0)$。从现在起，我们假定截距包含

在 $\hat{\beta}$ 中。

如果被视为 p 维输入空间的函数，则 $f(X) = X^T\beta$ 是线性的，而梯度 $f'(X) = \beta$ 是输入空间中的向量，指向上升最陡峭的方向。

如何用线性模型拟合训练数据集呢？有许多不同的方法，但迄今为止最流行的是最小二乘方（least square）。在这种方法下，我们选择系数 β，使得残差的平方和最小：

$$\mathrm{RSS}(\beta) = \sum_{i=1}^{N}(y_i - x_i^T\beta)^2 \tag{2.3}$$

$\mathrm{RSS}(\beta)$是参数的二次函数，因此极小值总是存在，但可能不惟一。解用矩阵形式最容易刻画。上式可以写为：

$$\mathrm{RSS}(\beta) = (\mathbf{y} - \mathbf{X}\beta)^T(\mathbf{y} - \mathbf{X}\beta) \tag{2.4}$$

其中，$\mathbf{X}$ 是 $N \times p$ 的矩阵，每行是一个输入向量，而 $\mathbf{y}$ 是训练数据集中输出的 N 向量。关于 β 微分，我们得到标准方程（normal equation）：

$$\mathbf{X}^T(\mathbf{y} - \mathbf{X}\beta) = 0 \tag{2.5}$$

如果 $\mathbf{X}^T\mathbf{X}$ 是非奇异的，则惟一解由下式给出：

$$\hat{\beta} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y} \tag{2.6}$$

并且第 i 个输入 x_i 的拟合值为$\hat{y}_i = \hat{y}(x_i) = x_i^T\hat{\beta}$。在任意输入 x_0 上，预测是$\hat{y}(x_0) = x_0^T\hat{\beta}$。整个拟合面被 p 个参数$\hat{\beta}$ 刻画。直观地，我们似乎并不需要很大的数据集来拟合这种模型。

下面考察一个用线性模型分类的例子。图 2.1 显示了输入对 X_1 和 X_2 上的训练数据的散点图。数据是模拟的，而现在我们不必关心模拟模型。输出类变量 G 有两个值 GREEN 或 RED，如散点图所示。两个类都有 100 个点。用线性回归拟合这些数据，响应变量 Y 用 0 和 1 分别表示 GREEN 和 RED。拟合值 $\hat{Y}$ 根据如下规则转换到拟合类变量 $\hat{G}$：

$$\hat{G} = \begin{cases} \mathrm{RED} & \text{如果 } \hat{Y} > 0.5 \\ \mathrm{GREEN} & \text{如果 } \hat{Y} \leqslant 0.5 \end{cases} \tag{2.7}$$

IR^2 中点的集合根据$\{x : x^T\hat{\beta} > 0.5\}$分类为 RED，如图 2.1 所示。两个预测类由判定边界$\{x : x^T\hat{\beta} = 0.5\}$分开；在此情况下，边界是线性的。可以看到，对于这些数据，判定边界两边都存在一些误分类。或许我们的线性模型太僵硬，或许这种错误不可避免。注意，这些是训练数据本身的误差，我们并未提及数据的来源。考虑两种可能的情况：

情况 1：每个类的训练数据都由二元高斯分布产生，这些高斯分布具有不相关的分量和不同的均值。

情况 2：每个类的训练数据都来自 10 个低方差的高斯分布的混合，个体均值本身服从高斯分布。

混合高斯分布可以用生成模型很好地解释。首先产生一个离散变量，决定高斯分布使用的分量，然后从选定的密度产生观测值。在每类一个高斯分布的情况下，正如将在第 4 章中介绍的，最好的情况就是线性判定边界，并且我们的估计几乎是最优的。区域重叠是不可避免的，并且此后的预测数据也被这种重叠所困扰。

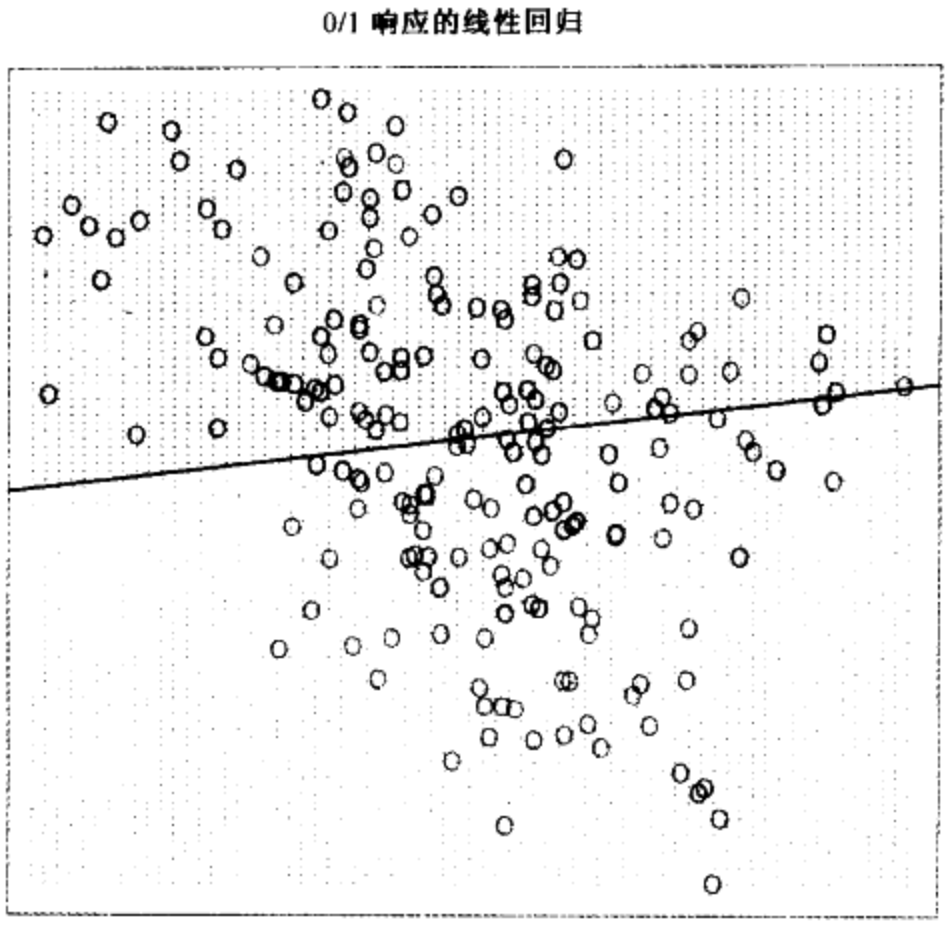


图 2.1　一个二维空间上的分类例子。类用二元变量编码(GREEN = 0,RED = 1),并且用线性回归拟合。直线是 $x^T\hat{\beta} = 0.5$定义的判定边界。红色区域表示输入空间被分类为RED的部分,而绿色区域被分类为GREEN(见彩页)

在紧密聚集高斯分布混合的状况下,情况截然不同。看来线性判定边界不是最优的,并且事实上它也不是。最优判定边界是非线性的、不相交的,因此更难得到它。

现在,考察另一种分类和回归过程。在某种意义下,它在线性模型谱系的另一端,并且更适合第二种情况。

2.3.2　最近邻法

最近邻法使用训练集 $\mathcal{T}$ 在输入空间中最邻近 x 的观测值形成 $\hat{Y}$。特殊地,拟合 $\hat{Y}$ 的 k-最近邻定义为:

$$\hat{Y}(x) = \frac{1}{k} \sum_{x_i \in N_k(x)} y_i \tag{2.8}$$

其中,$N_k(x)$是 x 的邻域,由训练样本中最邻近 x 的 k 个点 x_i 定义。邻近性意味有一种度量,不妨先假定这种度量为欧氏距离。换句话说,找出输入空间中与 x 最邻近的 k 个观测值 x_i,并对它们的响应取平均值。

在图 2.2 中,我们使用与图 2.1 相同的训练数据,并使用二元编码响应的 15-最近邻平均作为拟合方法。这样,$\hat{Y}$ 是邻域中 RED 所占的比例,并且如果 $\hat{Y} > 0.5$ 则置 $\hat{G}$ 为类 RED,这等价于在邻域中的多数表决。着色区域表示输入空间的所有点被该规则分类为 GREEN 或 RED(本例通过对输入空间细栅格上的求值过程找出)。我们看到,分隔 GREEN 和 RED 的判定边界更不规则,并且反映被一个类支配的局部聚类。

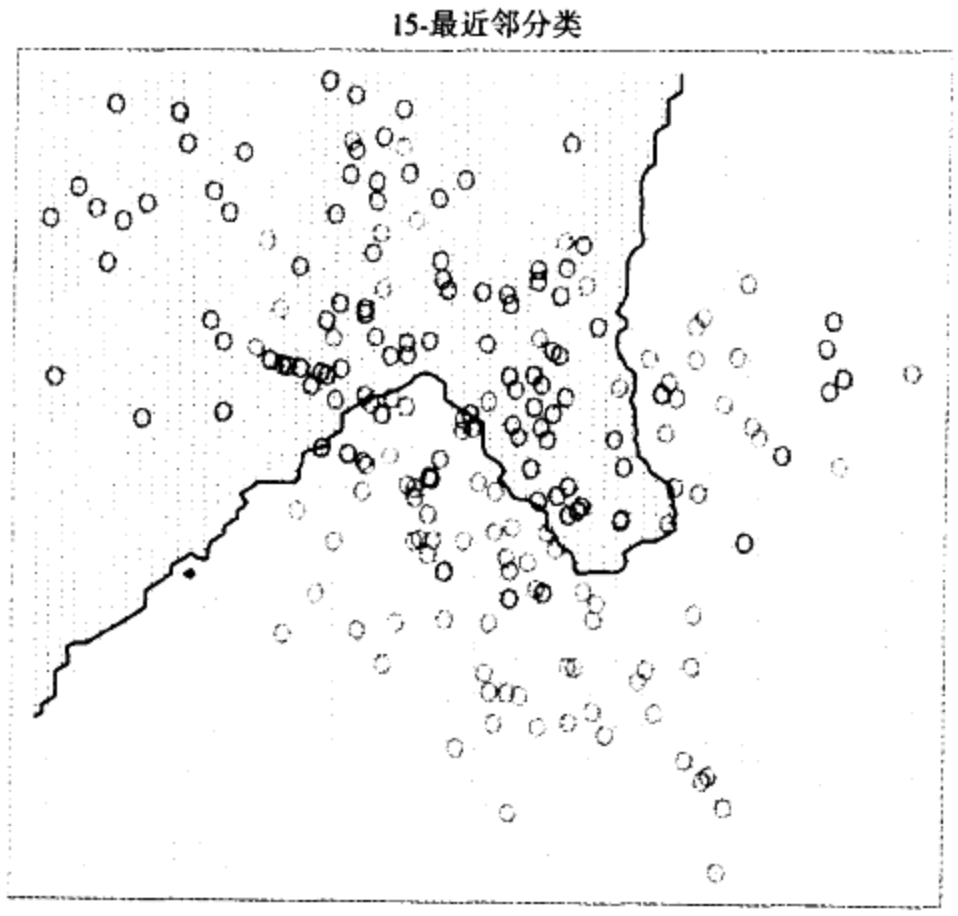


图 2.2 与图 2.1 相同的二维分类例子。类用二元变量编码(GREEN = 0,RED = 1),并用式(2.8)的15-最近邻平均拟合。因此,预测类用15-最近邻的多数表决确定(见彩页)

图 2.3 显示了 1 -最近邻分类的结果:$\hat{Y}$ 被赋予训练数据中 x 最邻近的点 x_ℓ 的值 y_ℓ。在此情况下,分类区域相对容易计算,并对应于训练数据的 Voronoi 嵌图(tessellation)。每个点 x_i 有一个相关联的方格,界定其最邻近的输入点的区域。对于方格中的所有点 x,$\hat{G}(x) = g_i$。与前面的相比,判定边界更加不规则。

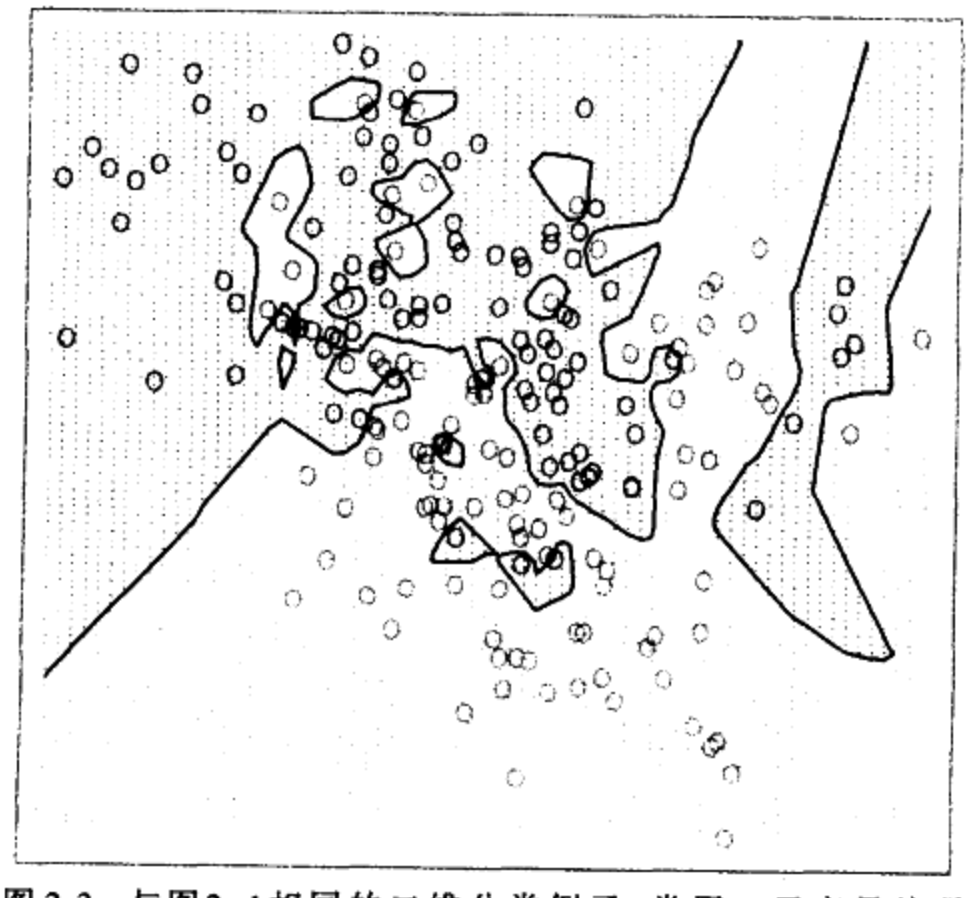


图 2.3 与图2.1相同的二维分类例子。类用二元变量编码(GREEN = 0, RED = 1),并用1-最近邻分类预测(见彩页)

定义 k-最近邻平均方法与定义定量输出 Y 的回归的方法完全相同，尽管 $k=1$ 是不太可能的选择。

在图 2.2 中，我们看到被错误分类的训练数据比图 2.1 少得多。这不应该给我们太多慰藉，因为图 2.3 中没有一个训练数据被错误地分类。少许的考虑暗示：对于 k-最近邻拟合，训练数据上的误差可能近似地是 k 的增函数，并对于 $k=1$ 取 0。一个独立的检验集应当为我们比较不同方法提供更满意的手段。

与最小二乘方拟合的 p 个参数相比，k-最近邻拟合似乎只有一个参数，即邻居的个数 k。尽管如此，k-最近邻有效的参数个数是 N/k，一般远大于 p，并随 k 增加而减小。为明白其中的原因，应注意：如果邻域不重叠，则有 N/k 个邻域，每个邻域需要配一个参数(均值)。

还要清楚，我们不能在训练数据集上使用误差的平方和作为选择 k 的标准，因为这样将总是选取 $k=1$。对于上面的情况 2，k-最近邻方法更合适；而对于高斯数据，k-最近邻的判定边界将会不必要地过于杂乱。

2.3.3　从最小二乘方到最近邻

最小二乘方的线性判定边界非常光滑，并且对于拟合显然是稳定的。看来它确实过分依赖如下假定：线性判定边界是合适的。用我们后面将要阐明的术语来说，它具有低方差和潜在的高偏倚。

另一方面，k-最近邻过程看上去不依赖对基础数据的任何严格假定，并能适合任何情况。然而，判定边界的任何特定子部分都依赖于少数输入点和它们的特定位置，并因而是摆动和不稳定的——高方差和低偏倚。

每种方法都有它自己的适用范围。特殊地，线性回归更适合上面的情况 1，而最近邻更适合于情况 2。该是揭开神话的时候了！事实上，数据是由一个模型产生的，介于二者之间，但更接近情况 2。首先，我们从一个二元高斯分布 $N((1,0)^T,\mathbf{I})$产生 10 个均值 m_k，并标记该类为 GREEN。类似地，再从 $N((0,1)^T,\mathbf{I})$提取 10 个并标记为类 RED。然后，对于每个类，按如下方法产生 100 个观测：对于每个观测，我们以 1/10 的概率随机选取一个 m_k，然后产生一个 $N(m_k,\mathbf{I}/5)$。这样就对每个类产生一个高斯聚类混合。图 2.4 展示了对 10 000 个由该模型产生的新观测分类的结果。我们将最小二乘方的结果与某 k 值区间的 k-最近邻的结果进行比较。

当今使用的大量流行技术大部分都是这两个简单过程的变种。事实上，1-最近邻(所有方法中最简单的)赢得了低维问题市场的大部分份额。下面列出了加强这些简单过程的一些方法：

- 核方法(kernel method)使用随至目标点的距离平滑地递减到 0 的权，而不是 k-最近邻所用的有效 0/1 权。
- 在高维空间中，修改距离核，以强调某变量比其他变量更重要。
- 局部回归通过局部加权最小二乘方，而不是局部拟合常数拟合线性模型。
- 线性模型拟合原输入的基展开，可以得到任意复杂的模型。
- 投影寻踪(projection pursuit)和神经网络模型由非线性变换的线性模型的和组成。

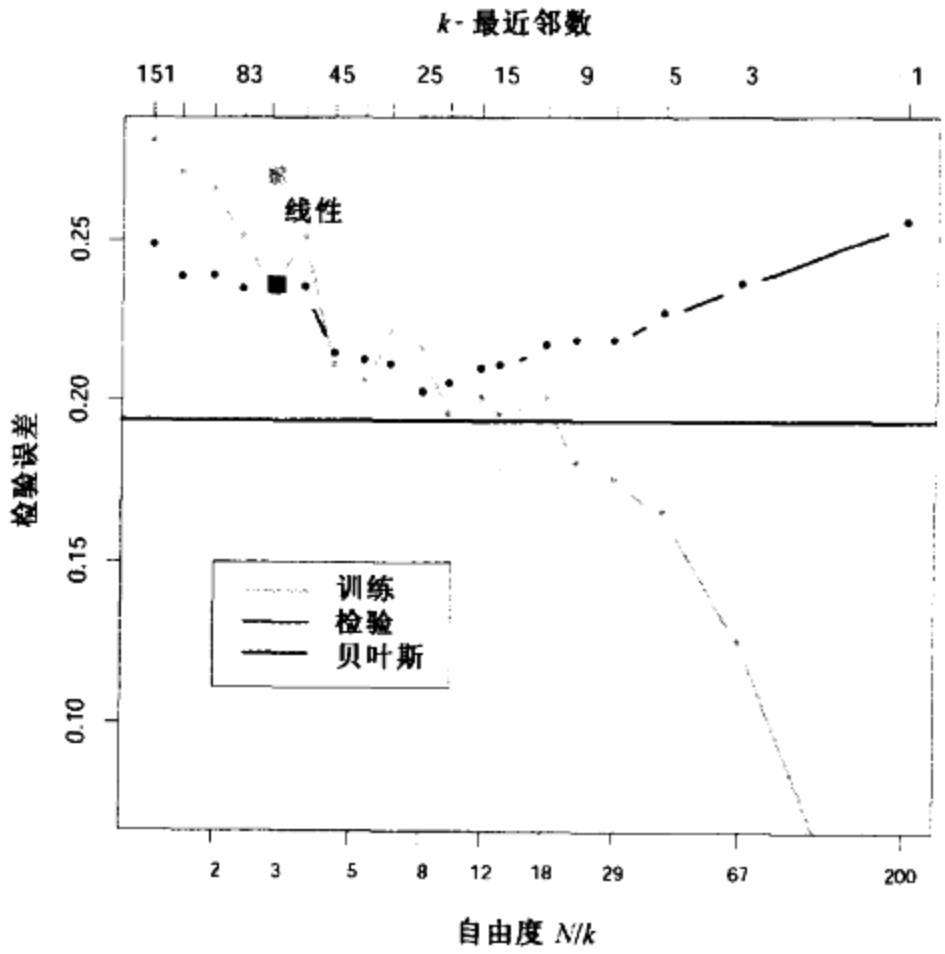


图 2.4 图 2.1、图 2.2 和图 2.3 使用的模拟例子的误分类曲线。使用一个规模为 200 的训练样本和一个规模为10 000的检验样本。红色曲线是 k- 最近邻分类的检验误差，绿色曲线是训练误差。线性回归的结果是三自由度上较大的红色和绿色方块。紫色直线是最优的贝叶斯误差率(见彩页)

2.4 统计判决理论

本节将阐述一点理论，为模型开发(如迄今为止一直非形式地讨论的那些)提供一个框架。首先考虑定量输出的情况，并置身于随机变量和概率空间世界。设 $X \in \mathbb{R}^p$ 是实数值随机输入向量，$Y \in \mathbb{R}$ 是实数值随机输出变量，具有联合分布 $\Pr(X, Y)$。我们寻找一个函数 $f(X)$，给定输入 X 的值预测 Y。该理论需要一个损失函数(loss function) $L(Y, f(X))$来处罚预测误差，而到目前为止最通用、最方便的是平方误差损失(squared error loss)：$L(Y, f(X)) = (Y - f(X))^2$。这给了我们一个选取 f 的标准——期望(平方)预测误差：

$$\begin{aligned} \mathrm{EPE}(f) &= \mathrm{E}(Y - f(X))^2 && (2.9) \\ &= \int (y - f(x))^2 \Pr(dx, dy) && (2.10) \end{aligned}$$

通过在 X 上取条件[①]，可以将 EPE 写为：

$$\mathrm{EPE}(f) = \mathrm{E}_X \mathrm{E}_{Y|X} \left([Y - f(X)]^2 | X \right) \tag{2.11}$$

我们只需要对 EPE 逐点极小化：

$$f(x) = \mathrm{argmin}_c \mathrm{E}_{Y|X} \left([Y - c]^2 | X = x \right) \tag{2.12}$$

① 这里，取条件实际上是分解联合密度 $\Pr(X, Y) = \Pr(Y|X)\Pr(X)$，其中 $\Pr(Y|X) = \Pr(Y, X)/\Pr(X)$，并相应分裂二元积分。

解是条件期望：

$$f(x) = \mathrm{E}(Y|X = x) \tag{2.13}$$

也称回归(regression)函数。这样，当使用平均平方误差度量最好时，任意点 $X = x$ 上 Y 的最好预测是条件均值。

最近邻法试图使用训练数据直接实现这一点。在每个点 x，我们可能寻找输入 $x_i = x$ 的所有y_i 的平均值。由于在任意点 x 一般最多只有一个观测，我们有：

$$\hat{f}(x) = \mathrm{Ave}(y_i|x_i \in N_k(x)) \tag{2.14}$$

其中，“Ave”表示平均，而 $N_k(x)$是邻域，包含 T 中 k 个距 x 最近的点。这里发生了两次近似：

- 通过在样本数据上求平均值，对期望取近似值。
- 在点上取条件放宽为在“靠近”目标点的某区域上取条件。

当训练样本的容量 N 很大时，邻域中的点多半靠近 x，并且随 k 增大，平均值趋向于稳定。事实上，在联合概率分布 $\Pr(X, Y)$适度正则的条件下，可以证明：随 $N, k \to \infty$ 使得 $k/N \to 0, \hat{f}(x) \to \mathrm{E}(Y|X = x)$。考虑到这一点，既然已经有了普适近似，为什么还要进一步找呢？通常，我们没有非常大的样本。如果线性或某种更结构化的模型是合适的，通常我们可以得到比 k-最近邻更稳定的估值，尽管这种知识也需要从数据中学习。还有一些问题，有时还很严重。在 2.5 节我们会看到，随维数 p 增大，k-最近邻域的度量规模也增大。这样，硬要用最近邻域替代取条件，我们将失败得很惨。收敛性依然成立，但收敛速度随维数增加而降低。

如何将线性回归纳入该框架呢？最简单的办法是假定回归函数 $f(x)$在其参数上是近似线性的：

$$f(x) \approx x^T\beta \tag{2.15}$$

这是一种基于模型的方法——我们为回归函数指定了一个模型。把 $f(x)$的这个线性模型插入式(2.9)的 EPE 并进行微分，理论上可以解出 β：

$$\beta = [\mathrm{E}(XX^T)]^{-1}\mathrm{E}(XY) \tag{2.16}$$

注意：我们没有在 X 上取条件，而是使用函数关系的知识将 X 的值合并。最小二乘方解式(2.6)实际上是用训练数据上的平均值替换式(2.16)中的期望值。

这样，通过平均，k-最近邻和最小二乘方最终都得到近似条件期望。但是，它们对模型的假定截然不同：

- 最小二乘方假定 $f(x)$可以用一个全局线性函数很好地近似。
- k-最近邻假定$f(x)$可以用一个局部常量函数很好地近似。

尽管后者看上去更可取，但是，我们已经看到必须为这种灵活性付出高昂代价。

本书介绍的许多更现代的技术都是基于模型的，尽管比严格的线性模型灵活得多。例如，加法模型(additive model)假定：

$$f(X) = \sum_{j=1}^{p} f_j(X_j) \tag{2.17}$$

这保持了线性模型的可加性，但每个坐标函数 f_j 是任意的。其结果是，加法模型的最佳估计使用诸如 k -最近邻技术，同时对每个坐标函数逼近单变量条件期望。通过强加某种(通常是不现实的)模型假定(在这种情况下是可加性)，估计高维空间的条件期望问题就可以化解。

你对式(2.11)的标准满意吗？ 如果用 $L_1: E|Y-f(X)|$ 取代损失函数 L_2 会怎样？ 在这种情况下，解是条件中位数(median)：

$$\hat{f}(x) = \text{median}(Y|X=x) \tag{2.18}$$

这是一种不同的定位度量，并且它的估计比条件平均值更健壮。L_1 标准的导函数不连续，这限制了它的广泛使用。其他更稳定的损失函数将在后面的章节中提及，但是平方误差在分析上最方便、最流行。

当输出是分类变量 G 时，我们怎么做？ 除需要不同的损失函数处罚预测误差外，做法相同。假设估计 $\hat{G}$ 在可能类的集合$\mathcal{G}$中取值。损失函数可以用一个 $K \times K$ 的矩阵 $\mathbf{L}$ 表示，其中 $K = \text{card}(\mathcal{G})$。$\mathbf{L}$ 的对角线上为 0，其他位置上非负，其中 $L(k,\ell)$ 是将属于类$\mathcal{G}_k$ 的观测分类为 $\mathcal{G}_\ell$ 的代价。通常使用0-1损失函数，所有误分类的代价都取 1 个单位。期望预测误差是：

$$\text{EPE} = \text{E}[L(G,\hat{G}(X))] \tag{2.19}$$

其中，期望仍根据联合分布 $\Pr(G,X)$ 取。再取条件，EPE 可以写成：

$$\text{EPE} = \text{E}_X \sum_{k=1}^{K} L[\mathcal{G}_k,\hat{G}(X)]\Pr(\mathcal{G}_k|X) \tag{2.20}$$

只需要对 EPE 逐点极小化：

$$\hat{G}(x) = \text{argmin}_{g\in\mathcal{G}} \sum_{k=1}^{K} L(\mathcal{G}_k,g)\Pr(\mathcal{G}_k|X=x) \tag{2.21}$$

使用 0-1 损失函数，上式化简为：

$$\hat{G}(x) = \text{argmin}_{g\in\mathcal{G}}[1-\Pr(g|X=x)] \tag{2.22}$$

或简单写为：

$$\hat{G}(X) = \mathcal{G}_k, \text{ 如果} \Pr(\mathcal{G}_k|X=x) = \max_{g\in\mathcal{G}} \Pr(g|X=x) \tag{2.23}$$

这个合理的解称为贝叶斯分类器(Bayes classifier)，它使用条件(离散)分布 $\Pr(G,X)$，将输入分到最可能的类。图 2.5 展示模拟例子的贝叶斯最优判定边界。贝叶斯分类的误差率称为贝叶斯率(Bayes rate)。

我们再次看到 k -最近邻分类直接逼近这个解——除了点上的条件概率放宽成点的邻域上的条件概率，以及用训练样本比例估计概率外，最近邻域中的多数表决事实上就是贝叶斯分类。

假定对于 2 -类问题选用哑变量方法，并用二元变量 Y 对 G 编码，后随平方误差损失估计。如果$\mathcal{G}_1$ 对应于 $Y=1$，则 $\hat{f}(X) = E(Y|X) = \Pr(G=\mathcal{G}_1|X)$。类似地，对于 K -类问题，$E(Y_k|X) = \Pr(G=\mathcal{G}_k|X)$。这表明我们的哑变量回归过程，后随用最大拟合值分类，是表示贝叶斯分类器的另一方法。尽管该理论是对的，但在实践上还可能出问题，这取决于所用的回归模型。例如，当使用线性回归时，$\hat{f}(X)$不一定为正，使用它作为概率估计可能有问题。第 4 章将讨论对 $\Pr(G|X)$建模的各种方法。

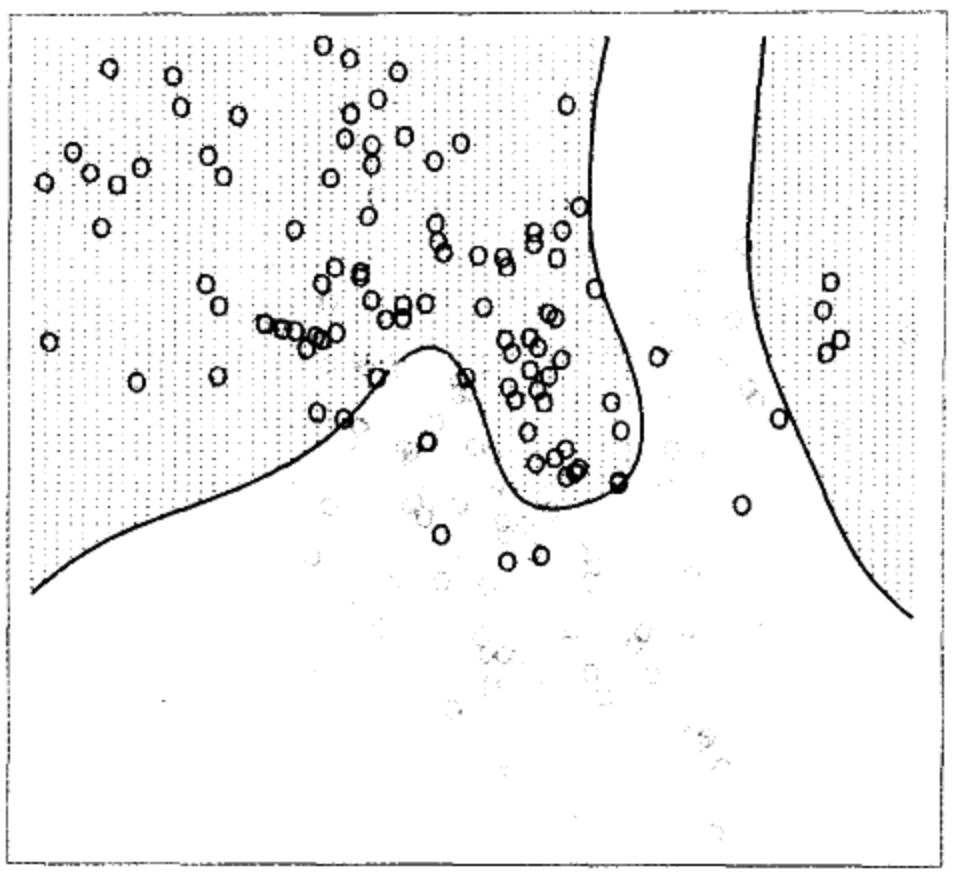


图 2.5　图 2.1、图 2.2 和图 2.3 的模拟例子的最优贝叶斯判定边界。由于每个类的生成密度是已知的，边界可以准确计算(见习题 2.2)

2.5　高维空间的局部方法

迄今为止，我们已经考察了预测的两种学习技术：稳定但存在偏倚的线性模型和不太稳定、但显然偏倚较小的 k-最近邻估计。似乎有了合理大的训练数据集，使用 k-最近邻平均总能逼近理论上的最佳条件期望，因为我们应当能够找到接近任意 x 的相当大的观测值邻域，并对它们取平均。该方法与我们的直觉在高维空间将失败，这种现象通常称为“维灾难”(curse of dimensionality)(Bellman，1961)。该问题有多种表现形式，这里将考察几种。

考虑输入在 p 维单位超立方体(见图 2.6)上均匀分布的最近邻过程。假定我们选取目标点的超立方体邻域，覆盖观测的一部分 r。由于这对应于单位体积的部分 r，故预期的边长为 $e_p(r) = r^{1/p}$。在 10 维空间，$e_{10}(0.01) = 0.63$，$e_{10}(0.1) = 0.80$，而每个输入的整个变程才是 1.0。这样，为得到数据的 1%或 10%以形成局部平均，我们必须覆盖每个输入变量变程的 63%或 80%。这样的邻域不再是“局部的”。大幅度降低 r 也无济于事，因为取平均值的观测越少，拟合的方差就越大。

高维空间中稀疏选样的另一个问题是所有样本点都靠近样本的边沿。考虑均匀分布在以原点为中心的 p 维单位球上的 N 个数据点。假定我们考虑原点上最近邻估计。从原点到最近数据点的中位数距离由下面的表达式给出(见习题 2.3)：

$$d(p,N)=\left(1-\frac{1}{2}^{1/N}\right)^{1/p} \tag{2.24}$$

对于到最近点的平均距离，存在更复杂的表达式。对于 $N=500, p=10, d(p,N)\approx 0.52$，超过到边界的一半。这样，大部分数据点更靠近样本空间的边界，而不是靠近其他数据点。提出这个问题是因为靠近训练样本边沿的预测更加困难。我们必须由邻近样本点外推，而不是在它们之间内插。

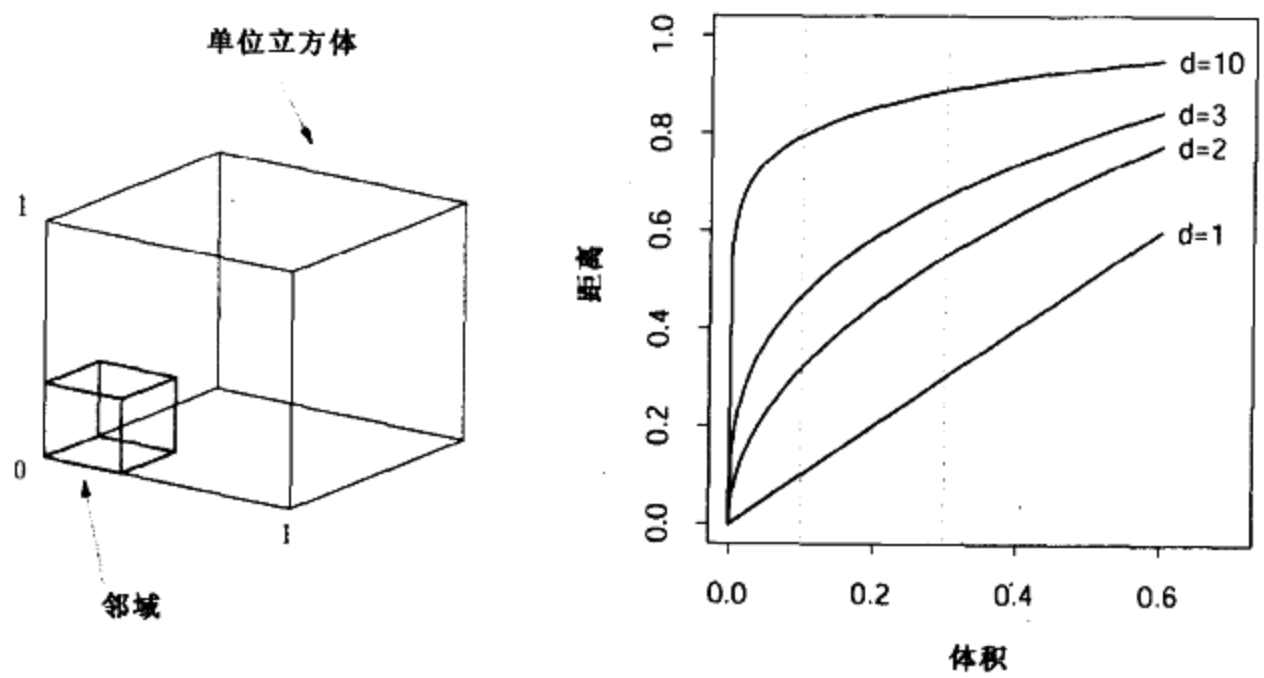


图 2.6 用单位立方体上的均匀分布数据的子立方体邻域，可以很好地解释“维灾难”。右边的图显示了对于不同的维数 p，覆盖数据的部分 r 所需要的子立方体的边长。在10维空间，为覆盖10%的数据，我们需要覆盖每个坐标的80%

维灾难的另一个现象是选样密度与 $N^{1/p}$ 成比例，其中 p 是输入空间的维数，而 N 是样本容量。这样，如果 $N_1 = 100$ 提供单输入问题的稠密样本，则 $N_{10}=100^{10}$ 是具有 10 个输入问题的相同选样密度所需要的样本容量。这样，在高维空间，所有可用的训练样本就稀疏地散布在输入空间。

让我们构造另一个均匀分布的例子。假定有 1000 个训练样本 x_i，均匀分布在 $[-1,1]^p$ 上。假定 X 和 Y 之间的真正联系是：

$$Y=f(X)=e^{-8\|X\|^2}$$

而没有任何观测误差。我们在检验点 $x_0=0$ 使用 1 -最近邻规则预测 y_0。记训练数据集为 $\mathcal{T}$。在所有 1000 个这样的样本上取平均值，可以计算我们的过程在 x_0 上的期望预测误差。由于该问题是确定性的，对 $f(0)$ 的估计是均方误差(MSE)：

$$\begin{aligned}\mathrm{MSE}(x_0) &= \mathrm{E}_{\mathcal{T}}[f(x_0)-\hat{y}_0]^2 \\ &= \mathrm{E}_{\mathcal{T}}[\hat{y}_0-\mathrm{E}_{\mathcal{T}}(\hat{y}_0)]^2+[\mathrm{E}_{\mathcal{T}}(\hat{y}_0)-f(x_0)]^2 \\ &= \mathrm{Var}_{\mathcal{T}}(\hat{y}_0)+\mathrm{Bias}^2(\hat{y}_0)\end{aligned} \tag{2.25}$$

图 2.7 显示了这种安排。我们已经将 MSE 分解成两部分：方差和平方偏倚。这种做法随后就会熟悉。这种分解总是可能的，并且常常是有用的，称为偏倚 - 方差分解(bias-variance decomposition)。除非最近的近邻在 0 上，否则该例中 $\hat{y}_0$ 比 $f(0)$ 小，因此平均估计向下偏斜。方差是由 1 -最近邻的选样方差造成的。在低维空间并且 $N=1000$ 时，最近的近邻非常接近于 0，

因此偏倚和方差都很小。随着维数增加，最近的近邻趋向于偏离目标点，因此偏倚和方差都将出现。当 $p = 10$ 时，对于 99% 以上的样本，最近的近邻离原点的距离大于 0.5。这样，随 p 增加，估计多半趋向于 0，因此和偏倚一样，MSE 稳定于 1.0，而方差开始下降（该例的人工安排）。

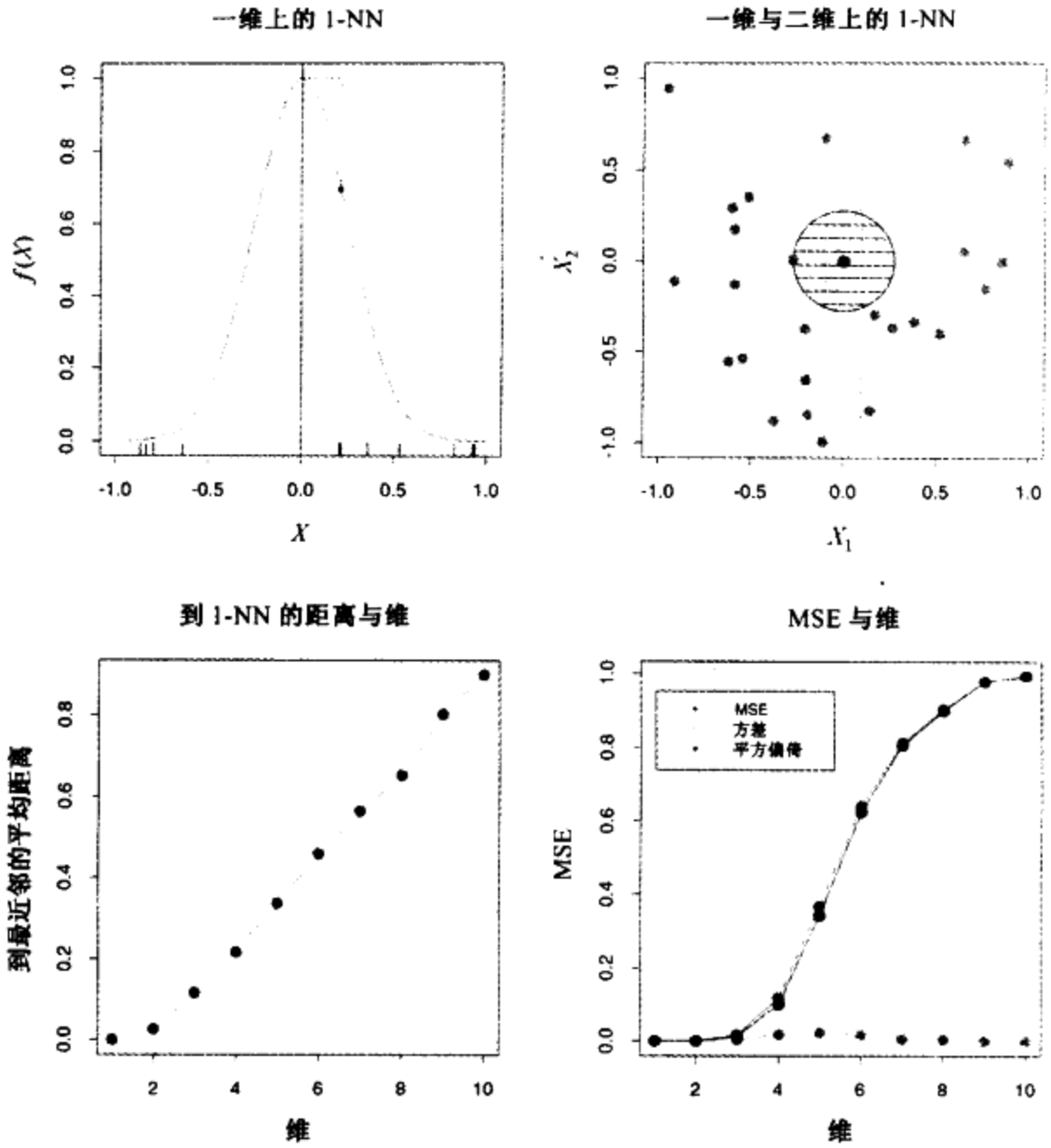


图 2.7　一个模拟例子，表明维灾难及其对 MSE、偏倚和方差的影响。对于 $p = 1, \cdots, 10$，输入特征值在 $[-1,1]^p$ 上均匀分布。左上角的图显示 $\mathbb{R}$ 上的（无噪声）目标函数：$f(X) = e^{-8\|X\|^2}$，并图示 1-最近邻对 $f(0)$ 估计所产生的误差。训练点用蓝色粗体标记。右上角的图展示 1-最近邻域的半径随维数 p 增加的原因。左下角的图展示 1-最近邻域的平均半径。右下角的图显示作为维数 p 的函数，MSE、平方偏倚和方差的曲线（见彩页）

尽管这是一个精心编排的例子，但类似的现象常常发生。许多变量的函数复杂性都随维数指数增加；并且，如果你希望以与低维函数相同的精度估计这样的函数，所需要的训练数据集的大小也将呈指数增长。这个例子中，函数是所涉及的所有 p 个变量的复杂交互作用。

偏倚项取决于距离是不争的事实，但它不一定总是对 1-最近邻占支配地位。例如，如果函数总是只涉及少数几个维（见图 2.8），则方差可能取而代之，占支配地位。

另一方面，假定我们知道 Y 和 X 之间的联系是线性的：

$$Y = X^T\beta + \varepsilon \tag{2.26}$$

其中，$\varepsilon \sim N(0,\sigma^2)$，并且我们用最小二乘方拟合训练数据。对于任意测试点 x_0，我们有，$\hat{y}_0 = x_0^T\hat{\beta}$，记为 $\hat{y}_0 = x_0^T\beta + \sum_{i=1}^{N}\ell_i(x_0)\varepsilon_i$，其中 $\ell_i(x_0)$是 $\mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}x_0$ 的第 i 个元素。由于在该模型下最小二乘方估计是无偏的，于是有：

$$\begin{aligned}\mathrm{EPE}(x_0) &= \mathrm{E}_{y_0|x_0}\mathrm{E}_{\mathcal{T}}(y_0-\hat{y}_0)^2 \\ &= \mathrm{Var}(y_0|x_0)+\mathrm{E}_{\mathcal{T}}[\hat{y}_0-\mathrm{E}_{\mathcal{T}}\hat{y}_0]^2+[\mathrm{E}_{\mathcal{T}}\hat{y}_0-\mathrm{E}_{\mathcal{T}}y_0]^2 \\ &= \mathrm{Var}(y_0|x_0)+\mathrm{Var}_{\mathcal{T}}(\hat{y}_0)+\mathrm{Bias}^2(\hat{y}_0) \\ &= \sigma^2+\mathrm{E}_{\mathcal{T}}x_0^T(\mathbf{X}^T\mathbf{X})^{-1}x_0\sigma^2+0^2\end{aligned} \tag{2.27}$$

这里，预测误差中又出现了一个附加的方差 σ^2，因为目标函数不是确定的。不存在偏倚，并且方差取决于 x_0。但是，如果 N 很大，$\mathcal{T}$ 是随机选择的，假定 $\mathrm{E}(X)=0$，则 $\mathbf{X}^T\mathbf{X}\to N\mathrm{Cov}(X)$，并且有：

$$\begin{aligned}\mathrm{E}_{x_0}\mathrm{EPE}(x_0) &\sim \mathrm{E}_{x_0}x_0^T\mathrm{Cov}(X)^{-1}x_0\sigma^2/N+\sigma^2 \\ &= \mathrm{trace}[\mathrm{Cov}(X)^{-1}\mathrm{Cov}(x_0)]\sigma^2/N+\sigma^2 \\ &= \sigma^2(p/N)+\sigma^2\end{aligned} \tag{2.28}$$

我们看到期望 EPE 作为 p 的函数线性增长，斜率为 σ^2/N。如果 N 很大或σ^2 很小，方差的增长可以忽略(在确定情况下为 0)。通过对拟合模型强加一些较强的限制，避免了维灾难。

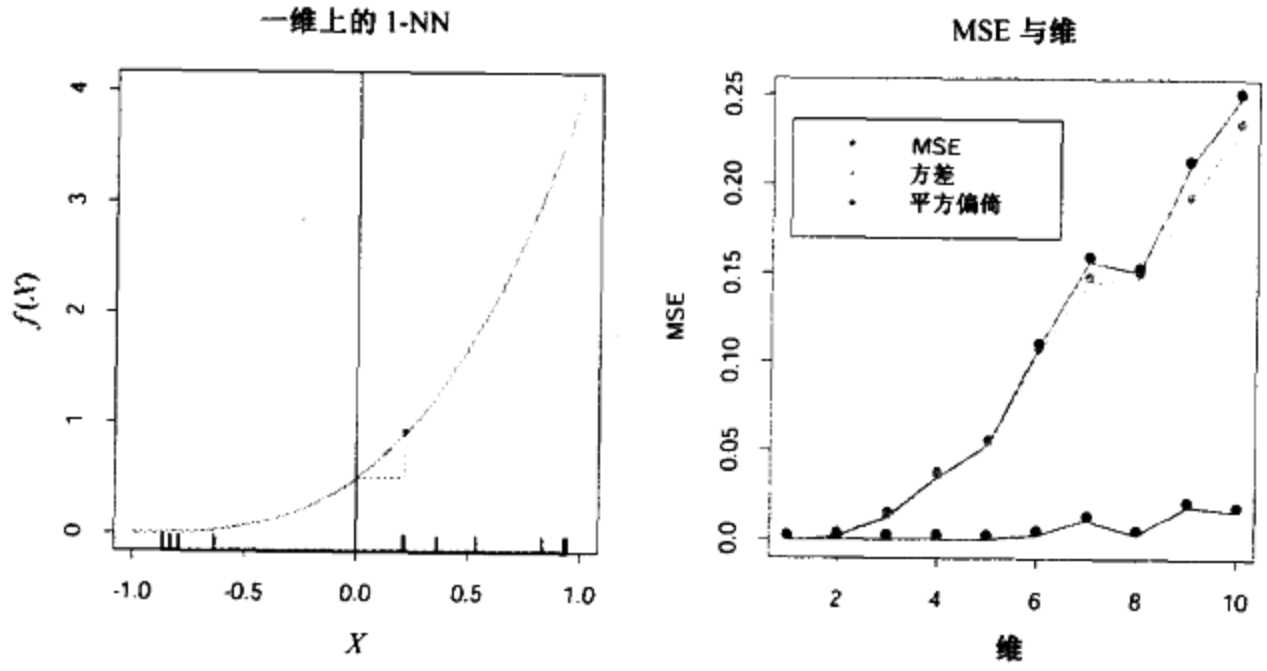


图 2.8 一个与图 2.7 具有相同设置的模拟例子。这里，除一个维为 $f(X)=\frac{1}{2}(X_1+1)^3$ 外，函数为常数。方差占支配地位(见彩页)

图 2.9 在两种情况下比较 1 -最近邻和最小二乘方。两种情况下，都有 $Y=f(X)+\varepsilon$；与前面一样，X 均匀分布，而 $\varepsilon \sim N(0,1)$。样本容量为 $N=500$。对于红色曲线，$f(x)$在第一个坐标上是线性的；对于绿色曲线，和图 2.8 一样，是三次的。图中显示的是 1 -最近邻和最小二乘方的相对 EPE。对于线性情况，相对 EPE 大约从 2 开始。在此情况下，最小二乘方是无偏的，并且如上所述，EPE 略大于 $\sigma^2=1$。1 -最近邻的 EPE 总是大于2，因为在此情况下，$\hat{f}(x_0)$的方差至少是 σ^2，并且随着最近的近邻飘离目标点，比例随维数增加。对于立方情况，最小二乘方是有偏的，这降低了比例。显然，我们可以杜撰一些例子，其中最小二乘方的偏倚可能超过方差，而 1 -最近邻将是赢家。

依赖于严格的假定，线性模型完全没有偏倚，并且方差可以忽略；而 1 -最近邻的误差相当

大。然而,如果假定是错误的,所有的断言都不成立,而 1 -最近邻可能占优势。我们将会看到,在严格的线性模型和极端灵活的 1 -最近邻模型之间有一个完整的模型谱系,每个都有自己的假定和偏倚。这些模型的提出是试图通过一些严格的假定,避免复杂性随维数指数增长。

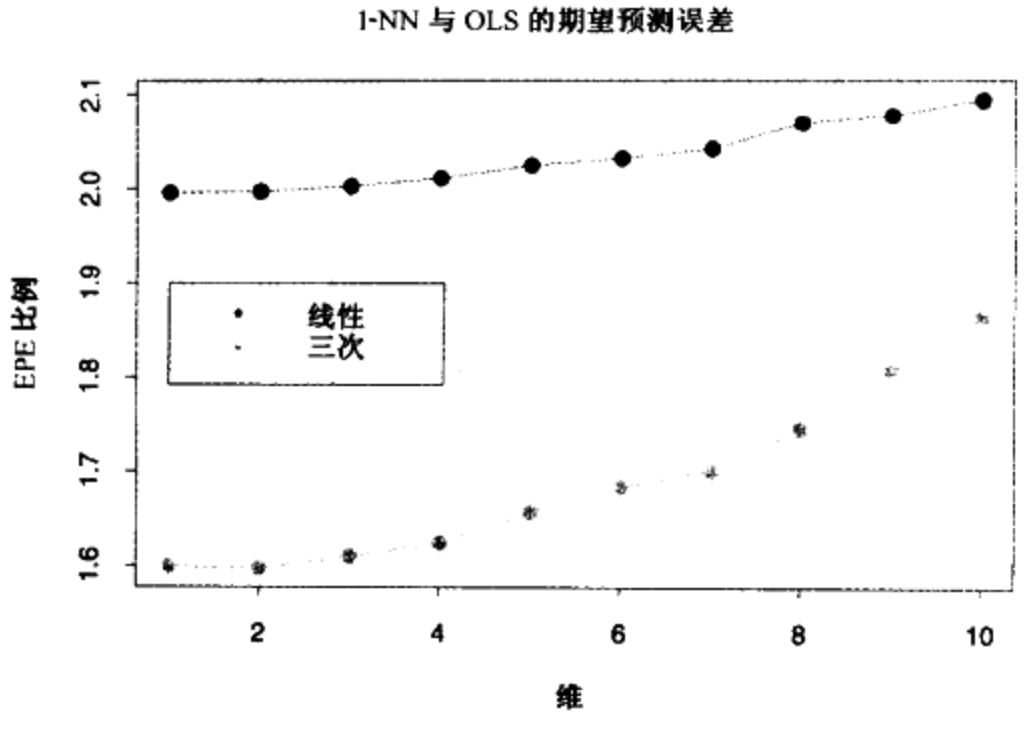


图 2.9　显示 1 -最近邻相对于最小二乘方关于模型 $Y = f(X) + \varepsilon$ 的期望预测误差的曲线(在 $x_0 = 0$)。对于红色曲线,$f(x) = x_1$;对于绿色曲线,$f(x) = \frac{1}{2}(x_1 + 1)^3$(见彩页)

在深入讨论之前,我们先稍微详细阐述一下统计模型的概念,并看看它们如何纳入预测框架。

2.6　统计模型、有指导学习和函数逼近

我们的目标是:对于预测输入和输出之间联系的函数 $f(x)$,找到一个有用的逼近 $\hat{f}(x)$。在2.4 节的理论框架中看到,平方误差损失将我们引向定量响应的回归函数 $f(x) = E(Y|X = x)$。最近邻这类方法可以看做是该条件期望的直接估计,但我们已经看到它们至少在两种情况下可能失败:

- 如果输入空间的维数很高,最近的近邻不一定靠近目标点,并可能导致较大误差。
- 如果知道存在特殊结构,则可以用来降低估值的偏倚和方差。

我们期望对 $f(x)$使用其他类型的模型,在许多情况下是为了克服维问题而专门设计的;而这里将讨论一个把它们纳入预测问题的框架。

2.6.1　联合分布 Pr(X, Y)的统计模型

假定我们的数据实际上来自统计模型:

$$Y = f(X) + \varepsilon \tag{2.29}$$

其中,随机误差 ε 满足 $E(\varepsilon) = 0$ 并独立于 X。注意,对于该模型,$f(x) = E(Y|X = x)$,并且事实上条件分布 $Pr(Y|X)$仅通过条件均值 $f(x)$依赖于 X。

加法误差模型是一种对真实情况的有用逼近。对于大部分系统，输入－输出对(X, Y)不具备确定的联系$Y=f(X)$。通常，存在其他未测量的变量，包括测量误差，它们也影响Y。加法模型假定我们可以通过误差ε，由确定性的联系捕获所有偏倚。

对于一些问题，确定性的联系确实存在。机器学习研究的许多分类问题都是这种形式，其中响应面可以想像成一张$\mathbb{R}^p$上的着色图。训练数据由图中着色的实例$\{x_i, g_i\}$组成，而目标是能够对任意点着色。这里，函数是确定性的，通过训练点的位置x引进随机性。现在暂时不研究该问题，而是先看看如何用基于误差的模型的适当技术来处理它们。

式(2.29)中的假定(误差是独立的同分布)不是严格必要的，但当在EPE标准中对平方误差一致地取平均值时，它确实在我们的下意识之中。有了这样一个模型，使用最小二乘方作为模型估计的数据标准[如式(2.1)]就变得很自然。可以做一些简单的修订，以避开独立性假定；例如，可以有$\mathrm{Var}(Y|X=x)=\sigma(x)$，而此时均值和方差都取决于$X$。一般来说，条件分布$\Pr(Y|X)$可以以复杂的方式依赖于$X$，但加法误差模型排除了这些。

迄今为止，我们一直专注定量的响应。通常，加法模型不用于定性输出G。对于定性问题，目标函数$p(X)$是条件密度$\Pr(G|X)$，并被直接建模。例如，对于2-类数据，通常合理地假定数据来自独立的二值实验，其中一个特定结果的概率为$p(X)$，另一个的概率为$1-p(X)$。这样，如果Y是G的0-1编码版本，则$\mathrm{E}(Y|X=x)=p(x)$，但是方差也取决于x：$\mathrm{Var}(Y|X=x)=p(x)[1-p(x)]$。

2.6.2 有指导学习

在使用更多的统计学行话之前，我们先从机器学习的角度提供函数拟合范例。为简单起见，假定误差是可加的，并且模型$Y=f(X)+\varepsilon$是一个合理的假设。有指导的学习试图通过一个“教师”由实例学习f。在系统的学习阶段，输入和输出装配成一个观测的训练集$\mathcal{T}=\{(x_i, y_i), i=1,\cdots,N\}$。观测到的输入值$x_i$同时提供给一个称为学习算法的人工系统(通常是一个计算机程序)，该系统也产生一个输出$\hat{f}(x_i)$响应该输入。学习算法具有一个性质：它可以调整输入/输出联系$\hat{f}$，以响应原来的输出和产生的输出之间的差$y_i-\hat{f}(x_i)$。该过程称为通过实例学习(learning by example)。一旦学习过程完成，希望人工的和实际的输出足够接近，使得算法对于实际中可能遇到的所有输入集有用。

2.6.3 函数逼近

上一节介绍的学习范例是研究机器学习(模拟人的推理)和神经网络(对人脑的生物学模拟)领域有指导学习问题的动机。应用数学和统计学接受的方法是函数逼近和估计的观点。这里，数据对(x_i, y_i)被视为$(p+1)$维欧氏空间中的点。函数$f(x)$的定义域对应于p维输入子空间，并通过一个诸如$y_i=f(x_i)+\varepsilon_i$的模型与数据建立联系。一般地，输入可以是混合类型，但为了方便，本章假定定义域是p维欧氏空间$\mathbb{R}^p$。目标是，给定在$\mathcal{T}$中的表示，对于$\mathbb{R}^p$某区域中的所有x，得到$f(x)$的一个有用逼近。尽管不如学习方法吸引人，将有指导的学习处理成函数逼近问题，有利于将欧氏空间的几何概念和概率推理的数学概念用于该问题。这是本书采用的方法。

我们将遇到的许多逼近都与一个参数集θ有关，该参数集可以调整，以适合手头的数据。例

如,线性模型 $f(x)=x^T\beta$ 有 $\theta=\beta$。另一类有用的逼近可以用线性基展开式(linear basis expansion)表示:

$$f_\theta(x)=\sum_{k=1}^{K}h_k(x)\theta_k \tag{2.30}$$

其中,h_k 是输入向量 x 的函数或变换的适当集合。传统的例子是多项展开式和三角展开式,其中 h_k 可以是 x_1^2, $x_1x_2^2$, $\cos(x_1)$等。我们也会遇到非线性展开式,如神经网络模型常见的S型(sigmoid)变换,

$$h_k(x)=\frac{1}{1+\exp(-x^T\beta_k)} \tag{2.31}$$

正如在线性模型中所做的那样,我们可以通过对残差的平方和(θ 的函数)

$$\mathrm{RSS}(\theta)=\sum_{i=1}^{N}(y_i-f_\theta(x_i))^2 \tag{2.32}$$

极小化,用最小二乘方估计 f_θ 中的参数 θ。对于加法误差模型,这看上去是一个合理的标准。用函数逼近的术语,想像参数化的函数是 $p+1$ 维空间的曲面,而我们观察的是它的有噪声实现。当 $p=2$ 时容易图示,垂直坐标是输出 y,如图2.10所示。噪声在输出坐标上,因而我们要找出参数集合使得拟合曲面尽可能接近被观测的点。这里,接近用 $\mathrm{RSS}(\theta)$ 中的垂直误差平方和度量。

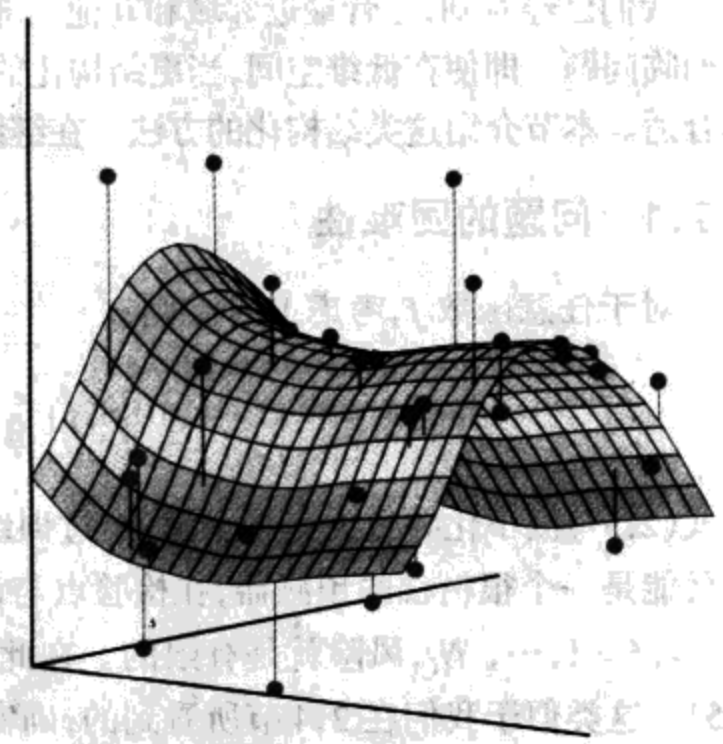


图2.10　双输入函数的最小二乘方拟合。选取 $f_\theta(x)$ 的参数,使得垂直误差的平方和最小

对于线性模型,我们得到极小化问题的一个简单的封闭形式的解。对于基函数方法,如果基函数本身不含隐藏的参数也能如此;否则,求解需要进行迭代或数值优化。

尽管最小二乘方通常是最方便的,但它不是惟一使用的标准,并且在某些情况下没有多大意义。一个更通用的评估原则是最大似然估计(maximum likelihood estimation)。假定有随机样本 $y_i, i=1,\cdots,N$,选自某参数 θ 标定的密度函数 $\mathrm{Pr}_\theta(y)$。观测样本的对数概率是:

$$L(\theta)=\sum_{i=1}^{N}\log \mathrm{Pr}_\theta(y_i) \tag{2.33}$$

最大似然原则假定 θ 最合理的值是使观测样本的概率最大的那些。对于加法误差模型 $Y=f_\theta(X)+\varepsilon$, $\varepsilon\sim N(0,\sigma^2)$,最小二乘方等价于使用条件似然

$$\Pr(Y|X,\theta)=N(f_\theta(X),\sigma^2) \tag{2.34}$$

的最大似然。这样,尽管附加的正态性假定看上去更严格,但结果相同。数据的对数似然是:

$$L(\theta) = -\frac{N}{2}\log(2\pi) - N\log\sigma - \frac{1}{2\sigma^2}\sum_{i=1}^{N}(y_i - f_\theta(x_i))^2 \tag{2.35}$$

并且惟一涉及 θ 的项在最后，它是 RSS(θ)乘以一个负数。

一个更有趣的例子是定量输出 G 的回归函数 $\Pr(G|X)$的多项式似然。给定 X，对于被参数向量 θ 定标的每个类的条件概率，假定有一个模型 $\Pr(G=\mathcal{G}_k|X=x)=p_{k,\theta}(x)$，$k=1,\cdots N$，则该对数似然[又称互熵(cross entropy)]是：

$$L(\theta) = \sum_{i=1}^{N}\log p_{g_i,\theta}(x_i) \tag{2.36}$$

并且，最大化时它产生在该似然意义下最符合数据的 θ 值。

2.7 结构化回归模型

我们已经看到，尽管最近邻域和其他局部方法直接关注于给定点上的函数，但在高维空间它们面临问题。即使在低维空间，当更结构化的方法可以使得数据的使用更有效时，它们也可能并不合适。本节介绍这类结构化的方法。在继续进行之前，我们先进一步讨论对这类方法的需求。

2.7.1 问题的困难性

对于任意函数 f，考虑 RSS 准则：

$$\mathrm{RSS}(f) = \sum_{i=1}^{N}(y_i - f(x_i))^2 \tag{2.37}$$

对式(2.37)极小化导致无穷多个解：经过训练点(x_i, y_i)的任意函数 $\hat{f}$ 都是解。任何选定的解都可能是一个很糟糕的预测器，在检验点与训练点不同。如果在每个 x_i 的值上有多个观测对 $x_i, y_{i\ell}, \ell=1,\cdots, N_i$，风险就是有限的。在此情况下，解经过每个 x_i 上的$y_{i\ell}$的平均值(见习题2.5)。这类似于我们在 2.4 节所看到的；确实，式(2.37)是式(2.11)的有限样本版本。如果样本的容量 N 足够大，使得确保重复并稠密地安排，这些解都可能趋向于极限条件期望。

为了对有限的 N 得到有用的结果，我们必须将式(2.37)符合条件的解限制在一个较小的函数集中。如何决定限制的特性是基于数据之外的考虑。有时，这些限制通过 f_θ 的参数表示编码，或者显式或隐式地在学习模型本身构建。这些受限制的解类型是本书的主题。然而，有件事情应当清楚：加在 f 上，导致式(2.37)惟一解的任何限制实际上并没有消除因多解导致的不确定性。存在无限多个可能的限制，每个限制导致一个惟一的解。这样，不确定性只是简单地转移到约束的选取。

一般地，大部分学习方法施加的约束都可以视为这种或那种复杂性限制。通常，这意味输入空间小邻域上的某种规则性。即，对于所有的输入点 x，在某种度量下，它们都彼此足够接近，$\hat{f}$ 显示出某种特殊的结构性，如近似常数、线性或低阶多项式。这样，估值就可以通过在邻域中取平均值或多项式拟合得到。

约束的强度被邻域的大小所左右。邻域越大，约束越强，并且解对于约束的特定选择就越敏感。例如，在无穷小的邻域中的局部常数拟合已不再是约束；在非常大的邻域上的局部线性

拟合几乎是全局的线性模型，并且限制很强。

约束的特性取决于所使用的度量。有些方法，如核与局部回归和基于树的方法，直接指定度量和邻域的大小。迄今为止讨论的最近邻方法基于如下假定：函数为局部常量；靠近目标输入 x_0，函数变化不大，并因此可以对邻近的输出取平均值，产生 $\hat{f}(x_0)$。其他方法，如样条函数、神经网络和基函数方法，隐式地定义邻域的局部特性。在第 5.4.1 节，我们将讨论等价核（equivalent kernel）概念（见图 5.8），它对输出上线性的方法描述这种局部依赖性。在许多情况下，这些等价核就像前面讨论的加权核——在目标点达到峰值并由它平滑地下降。

迄今为止，有一个事实应当清楚：任何试图在一个各向同性的小邻域产生局部变化的函数的方法都将在高维空间遇到问题——维灾难。反之，克服维数问题的所有方法都有一个相关联的（通常是隐含的或自适应的）度量邻域的标准。这些标准基本上不允许邻域同时在所有方向上都很小。

2.8　受限的估计方法类

根据所施加的限制的特点，各种非参数回归技术或学习方法可以分成一些不同的种类。这些类是截然不同的，并且确实有一些方法可以归入多个类。由于详尽的讨论将在后面章节给出，这里我们只做一个简略概述。每个类都有与之相关联的一个或多个参数，有时适当地称之为光滑（smoothing）参数，它们控制局部邻域的实际大小。这里主要介绍三大类。

2.8.1　粗糙度罚和贝叶斯方法

有一类函数被具有粗糙度罚的显式罚 RSS(f)控制：

$$\mathrm{PRSS}(f;\lambda)=\mathrm{RSS}(f)+\lambda J(f) \tag{2.38}$$

对于在小输入区域变化太快的函数 f，用户选择的泛函 $J(f)$将很大。例如，流行的一维输入空间三次光滑样条（cubic smoothing spline）是罚最小二乘方准则的解：

$$\mathrm{PRSS}(f;\lambda)=\sum_{i=1}^{N}(y_i-f(x_i))^2+\lambda\int[f''(x)]^2\mathrm{d}x \tag{2.39}$$

这里，粗糙度罚控制 f 的二阶导函数的值，而罚量由 $\lambda\geqslant 0$ 控制。对于 $\lambda=0$，没有强加的罚，并且任意插值函数都可以使用；而对于 $\lambda=\infty$，只允许 x 上的线性函数。

可以对任意维上的函数构造罚泛函 J，并且可以创建特定的版本来利用特定结构。例如，使用加法罚 $J(f)=\sum_{j=1}^{p}J(f_j)$ 与加法函数 $f(X)=\sum_{j=1}^{p}f_j(X_j)$ 一起创建具有光滑坐标函数的加法模型。类似地，对于自适应选择的方向 α_m，投影寻踪回归（projection pursuit regression）模型有 $f(X)=\sum_{m=1}^{M}g_m(\alpha_m^{T}X)$，并且每个函数 g_m 都可以有一个相关联的粗糙度罚。

罚函数，或正则化（regularization）方法表达了我们的先验信念：所寻找的函数类型具有某种光滑性，并且确实可以纳入贝叶斯框架。罚泛函 J 对应于对数先验分布，PRSS($f;\lambda$)对应于对数后验分布，并且对 PRSS($f;\lambda$)极小化实际上是找出后验众数。我们将在第 5 章讨论粗糙度罚，将在第 8 章讨论贝叶斯方法。

2.8.2 核方法和局部回归

这些方法可以看做通过明确说明局部邻域的特性和局部拟合的正则函数，显式地提供回归函数的估计或条件期望。局部邻域由核函数(kernel function) $K_\lambda(x_0, x)$指定，它将权赋予 x_0 周围区域中的点 x(见图 6.1)。例如，高斯核具有基于高斯密度函数的权函数：

$$K_\lambda(x_0, x) = \frac{1}{\lambda}\exp\left[-\frac{\|x - x_0\|^2}{2\lambda}\right] \tag{2.40}$$

并且把随 x_0 到它们的欧氏距离的平方指数衰减的权赋给点。参数 λ 对应于高斯密度函数的方差，并控制邻域的宽度。核估计最简单的形式是 Nadaraya-Watson 加权平均：

$$\hat{f}(x_0) = \frac{\sum_{i=1}^{N} K_\lambda(x_0, x_i)y_i}{\sum_{i=1}^{N} K_\lambda(x_0, x_i)} \tag{2.41}$$

一般地，我们可以将 $f(x_0)$的局部回归估计定义为 $f_{\hat{\theta}}(x_0)$，其中 $\hat{\theta}$ 极小化：

$$\mathrm{RSS}(f_\theta, x_0) = \sum_{i=1}^{N} K_\lambda(x_0, x_i)(y_i - f_\theta(x_i))^2 \tag{2.42}$$

而 f_θ 是某种参数化函数，如低阶多项式。例子有：

- $f_\theta(x) = \theta_0$，常数函数；这导致式(2.41)Nadaraya -Watson 估计。
- $f_\theta(x) = \theta_0 + \theta_1 x$ 给出流行的局部线性回归模型。

最近邻方法可以想像为具有更多数据依赖度量的核方法。确实，k-最近邻的度量是：

$$K_k(x, x_0) = I(\|x - x_0\| \leqslant \|x_{(k)} - x_0\|)$$

其中，$x(k)$是到 x 的距离中秩为 k 的训练观测值，而 $I(S)$是集合 S 的指示符。

当然，在高维空间这些方法需要修改，以避免维灾难。各种调整将在第 6 章讨论。

2.8.3 基函数和字典方法

这类方法包括熟悉的线性和多项式展开，但更重要的是包括多种更灵活的模型。f 的模型是基函数的线性展开式：

$$f_\theta(x) = \sum_{m=1}^{M} \theta_m h_m(x) \tag{2.43}$$

其中，每个 h_m 都是输入 x 的函数，而这里的术语线性是指参数 θ 的作用。该类包含了大量各种不同类型方法。在某些情况下，基函数序列是指定的，如总次数为 M 的 x 上的多项式基。

对于一维 x，K 次多项式样条函数可以用 M 个样条基函数的适当序列表示，依次被 $M-K$ 个纽结(knot)确定。它们产生纽结间分段的 K 次多项式，并在纽结上 $K-1$ 次连续相交。作为一个例子，考虑线性样条函数或分段线性函数。直观地，一个满意的基函数包括 $b_1(x) = 1$，$b_2(x) = x$，$b_{m+2}(x) = (x - t_m)_+$，$m = 1, \cdots, M-2$，其中 t_m 是第 m 个纽结，而 z_+ 表示正的部分。样条基的张量积可以用于多维输入(见 5.2 节和第 9 章的 CART 和 MARS 模型)。参数 θ 可以是多项式的总次数或样条纽结的数目。

径向基函数(radial basis function)是在特定形心上对称的 p 维核，

$$f_\theta(x) = \sum_{m=1}^{M} K_{\lambda_m}(\mu_m, x)\theta_m \tag{2.44}$$

例如，高斯核 $K_\lambda = (\mu, x) = e^{-\|x-\mu\|^2/2\lambda}$ 很流行。

必须确定径向基函数具有的形心 μ_m 和标度 λ_m。样条基函数有纽结，通常我们也希望数据指出它们。包含这些参数将使回归问题从直接的线性问题变为组合困难的非线性问题。在实践中，采取诸如贪心算法和两阶段过程等捷径。第 6.7 节将讨论这样一些算法。

具有线性输出权的单层前馈神经网络模型可以看做自适应基函数方法。该模型形如：

$$f_\theta(x) = \sum_{m=1}^{M} \beta_m \sigma(\alpha_m^T x + b_m) \tag{2.45}$$

其中，$\sigma(x) = 1/(1 + e^{-x})$ 称为激活(activation)函数。这里，像在投影寻踪模型中一样，需要确定方向 α_m 和偏置 b_m，并且它们的估计是计算的中心内容。细节将在第 11 章给出。

这些自适应选取基函数的方法也称字典(dictionary)方法。这里，我们有一个候选基函数的无限集或字典 $\mathcal{D}$ 可供选择，并且通过使用某种搜索机制建立模型。

2.9　模型选择和偏倚 – 方差权衡

上面介绍的所有模型和后面章节中将要讨论的一些其他模型都有一个光滑(smoothing)或复杂性(complexity)参数需要确定：

- 罚项的乘数。
- 核的宽度。
- 或基函数的个数。

对于光滑样条，参数 λ 定标模型包括从直线拟合到插值模型。类似地，局部 m 次多项式模型包括从 m 次全局多项式(当窗口无限大)到内插拟合(当窗口的尺寸收缩为 0)。这意味着也不能使用训练数据上的残差平方和来确定这些参数，因为这样做我们总是选中那些内插拟合，并因此具有零残差。这样的模型多半不能很好地预测。

k-最近邻回归拟合 $\hat{f}_k(x_0)$ 很好地阐明了影响这种逼近预测能力的竞争力。假定数据源自一个模型 $Y = f(X) + \varepsilon$，其中 $\mathrm{E}(\varepsilon) = 0, \mathrm{Var}(\varepsilon) = \sigma^2$。为简单起见，我们假定样本中 x_i 的值预先给定(非随机的)。在 x_0 的期望预测误差也称检验(test)误差或泛化(generalization)误差，可以分解成：

$$\begin{aligned}
\mathrm{EPE}_k(x_0) &= \mathrm{E}[(Y - \hat{f}_k(x_0))^2 | X = x_0] \\
&= \sigma^2 + [\mathrm{Bias}^2(\hat{f}_k(x_0)) + \mathrm{Var}_{\mathcal{T}}(\hat{f}_k(x_0))] \qquad (2.46) \\
&= \sigma^2 + \left[f(x_0) - \frac{1}{k}\sum_{\ell=1}^{k} f(x_{(\ell)})\right]^2 + \frac{\sigma^2}{k} \qquad (2.47)
\end{aligned}$$

括号中的下标 (ℓ) 指定 x_0 的最近邻的序列。

该表达式中有三项。第一项 σ^2 是不可约的(irreducible)误差(新检测目标的方差),既便知道实际的 $f(x_0)$,我们也无法控制该误差。

第二项和第三项在我们的控制之中,并组成估计 $f(x_0)$ 时 $\hat{f}_k(x_0)$ 的均方误差。均方误差被分解成偏倚和方差两部分。偏倚项是实际均值 $f_k(x_0)$ 与估计的期望值之差的平方 $[\mathrm{E}_{\mathcal{T}}(\hat{f}_k(x_0)) - f(x_0)]^2$,其中期望对训练数据中的随机性取平均值。如果实际函数相当光滑,该项多半会随 k 增加。对于较小的 k,少量最近邻将具有接近 $f(x_0)$ 的值 $f(x_{(\ell)})$,从而它们的平均值将接近 $f(x_0)$。随着 k 的增长,近邻将进一步远离,从而什么情况都可能发生。

这里,方差项是简单的平均方差,并随 k 增加而减小。从而,随 k 变化,需要在偏倚和方差之间权衡。

更一般地,随着我们的过程模型的复杂度增加,方差趋向于增加,而平方偏倚趋向于减小。随着模型的复杂度降低,情况相反。对于 k-最近邻方法,模型的复杂度被 k 控制。

通常,我们希望这样选择模型的复杂度:在偏倚和方差之间权衡,使检验误差最小。检验误差的一个显而易见的估计是训练误差(training error) $\frac{1}{N}\sum_i (y_i - \hat{y}_i)^2$。遗憾的是,训练误差不是检验误差的一个好的估计,因为它不能适当解释模型的复杂性。

图 2.11 展示随模型复杂度的变化,检验和训练误差的典型特点。当提高模型的复杂性(即更严格地拟合数据)时,训练误差趋向于减小。然而,过分拟合使得模型自适应过分适合训练数据,而不能很好地泛化(即具有较大的检验误差)。在此情况下,预测 $\hat{f}(x_0)$ 将具有较大方差,如式(2.46)的最后一项所示。反之,如果模型不够复杂,它将拟合不足并可能具有较大偏倚,又导致较差的泛化。在第 7 章,我们将讨论评估预测方法检验误差的方法,从而对于给定的预测方法和训练数据评估模型复杂度的最佳值。

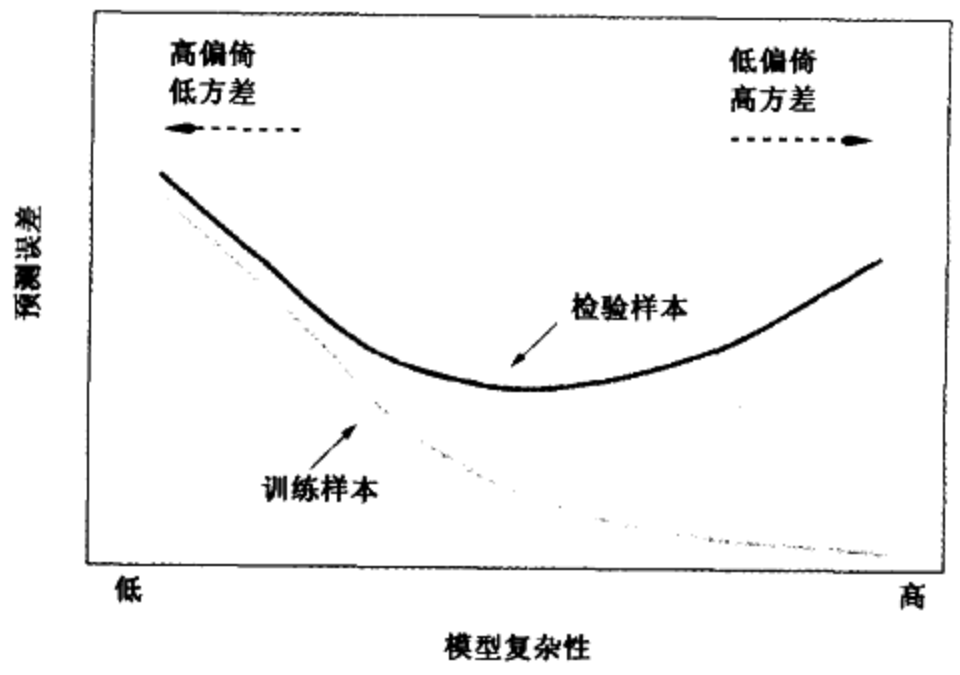


图 2.11 检验和训练误差作为模型复杂性的函数

文献注释

关于学习问题的一些好的书籍有 Duda 等人(2000),Bishop(1995),Ripley(1996),Cherkassky 和 Mulier(1998),Vapnik(1996)的著作。本章的部分内容基于 Friedman(1994b)。

习题

2.1 假定 K-类的每一个都有一个相关联的目标 t_k。其中，t_k 是向量，除第 k 个位置为 1 外全为 0。证明：如果 $\hat{y}$ 的元素和为 1，分类到 $\hat{y}$ 的最大元素相当于选取最近的目标 $\min_k \| t_k - \hat{y} \|$。

2.2 试述如何计算图 2.5 中模拟例子的贝叶斯判定边界。

2.3 推导式(2.24)。

2.4 在第 2.5 节中讨论的边沿影响问题不是有界域均匀抽样独有的。考虑取自球形多项分布 $X \sim N(0, \mathbf{I}_p)$ 的输入。从任意样本点到原点的平方距离服从具有均值 p 的 χ_p^2 分布。考虑取自该分布的预测点 x_0，并设 $a = x_0 / \| x_0 \|$ 是一个相关的单位向量。设 $z_i = a^T x_i$ 为每个训练点在该方向上的投影。

(a) 证明 z_i 分布在 $N(0, 1)$ 上，具有到原点的期望平方距离 1，而目标点具有到原点的期望平方距离 p。

(b) 对于 $p = 10$，证明从训练数据的中心到检验点的期望距离是 3.1 倍标准差，而所有训练点沿方向 a 具有期望距离 1。从而，大部分预测点位于训练集的边沿。

2.5 考虑一个回归问题，x_i 是输入，y_i 是输出，而 $f_\theta(x)$ 是被最小二乘方拟合的参数模型。证明：如果观测在 x 存在结或恒等值，则拟合可以通过简化加权最小二乘方问题得到。

2.6 假定我们有一个容量为 N 的样本，样本中的对 x_i, y_i 独立同分布地取自如下分布：

$$x_i \sim h(x), \text{设计密度}$$
$$y_i = f(x_i) + \epsilon_i, f \text{是回归函数}$$
$$\epsilon_i \sim (0, \sigma^2) \text{（均值 0，方差 } \sigma^2\text{）}$$

为 f 构造一个在 y_i 上线性的估计器：

$$\hat{f}(x_0) = \sum_{i=1}^{N} \ell_i(x_0; \mathcal{X}) y_i$$

其中，权 $\ell_i(x_0; \mathcal{X})$ 不依赖于 y_i，而依赖于 x_i 的整个训练序列，记做 $\mathcal{X}$。

(a) 证明线性回归和 k-最近邻域回归属于这类估计器。对于这两种情况，给出 $\ell_i(x_0; \mathcal{X})$。

(b) 将条件均方误差

$$\mathrm{E}_{y|\mathcal{X}}(f(x_0) - \hat{f}(x_0))^2$$

分解成条件平方偏倚和条件方差。和 $\mathcal{X}$ 一样，y 代表 y_i 的整个训练序列。

(c) 将均方误差(无条件)

$$\mathrm{E}_{y,\mathcal{X}}(f(x_0) - \hat{f}(x_0))^2$$

分解成平方偏倚和方差。

(d) 建立以上两种情况的平方偏倚和方差的联系。

2.7 在 zipcode 数据上，比较线性回归和 k-最近邻域分类的性能。特殊地，只考虑数字 2 和 3，$k = 1, 3, 5, 7$ 和 15。对每种选择，给出训练和检测误差。zipcode 数据可以从本书的 Web 网站 www-stat.stanford.edu/ElemStatLearn 中得到。

第3章　回归的线性方法

3.1　引言

线性回归模型假定回归函数 $E(Y|X)$ 在输入 $X_1,\cdots,X_p$ 上是线性的。线性模型在统计学的计算机前时代已有了很大的发展，但是即便在当今的计算机时代，依然有充足的理由研究并使用它们。它们简单，并且常常对输入如何影响输出提供充分和可解释的描述。对于预测，它们通常远胜过非线性模型，特别是在训练数据数量较少，信噪比较低或稀疏数据的情况下更是如此。最后，线性方法可以用在变换后的输入上，并且这能大大扩展它们的应用范围。这些拓广有时称为基函数方法，将在第5章讨论。

本章将介绍回归的线性方法，而下一章讨论分类的线性方法。对于某些主题，本章将相当详细地讨论，因为我们确信对于理解非线性模型，理解线性模型是至关重要的。事实上，许多非线性技术正是这里讨论的线性方法的直接拓广。

3.2　线性回归模型和最小二乘方

正如第2章所述，我们有输入向量 $X=(X_1,\cdots,X_p)$，并希望预测实数值输出 Y。线性回归模型形如：

$$f(X)=\beta_0+\sum_{j=1}^{p}X_j\beta_j \tag{3.1}$$

线性模型假定回归函数 $E(Y|X)$ 是线性的，或者假定线性模型是一个合理的近似。这里，β_j 是未知参数或系数，而变量 X_j 可能来自不同的源：

- 定量输入。
- 定量输入的变换，如对数、方根或平方。
- 基展开，如 $X_2=X_1^2, X_3=X_1^3$，导致多项式表示。
- 定性输入级的数值或“哑”编码。例如，如果 G 是5级因素输入，我们可以创建 $X_j, j=1,\cdots,5$，使得 $X_j=I(G=j)$。通过级依赖的常量集，这一组 X_j 表现了 G 的效果，因为在 $\sum_{j=1}^{5}X_j\beta_j$ 中，X_j 中的一个为1，其他为0。
- 变量间的交互作用。如 $X_3=X_1\cdot X_2$。

无论 X_j 的源是什么，模型在其参数上都是线性的。

典型地，我们有一个训练数据集 $(x_1,y_1),\cdots,(x_N,y_N)$，通过它们估计参数 β。每个 $x_i=(x_{i1},x_{i2},\cdots,x_{ip})^T$ 是第 i 个数据的特征度量向量。最流行的估计方法是最小二乘方，它选择

系数 $\beta=(\beta_0,\beta_1,\cdots,\beta_p)^T$，以极小化残差的平方和：

$$\begin{aligned}\mathrm{RSS}(\beta) &= \sum_{i=1}^{N}(y_i-f(x_i))^2 \\ &= \sum_{i=1}^{N}\Big(y_i-\beta_0-\sum_{j=1}^{p}x_{ij}\beta_j\Big)^2\end{aligned} \tag{3.2}$$

从统计学角度，如果训练观测(x_i,y_i)从总体中独立地随机抽取，该准则是合理的。即使 x_i 不是随机抽取的，如果y_i 条件独立于给定的 x_i，该准则依然有效。图 3.1 给出对(X,Y)所处的 $\mathbb{R}^{p+1}$空间中的最小二乘方拟合的几何图解。注意：式(3.2)并不假定模型(3.1)的有效性，它只是找出数据的最好线性拟合。直观上，最小二乘方拟合是令人满意的，不管数据源于何处；该准则度量平均拟合偏离。

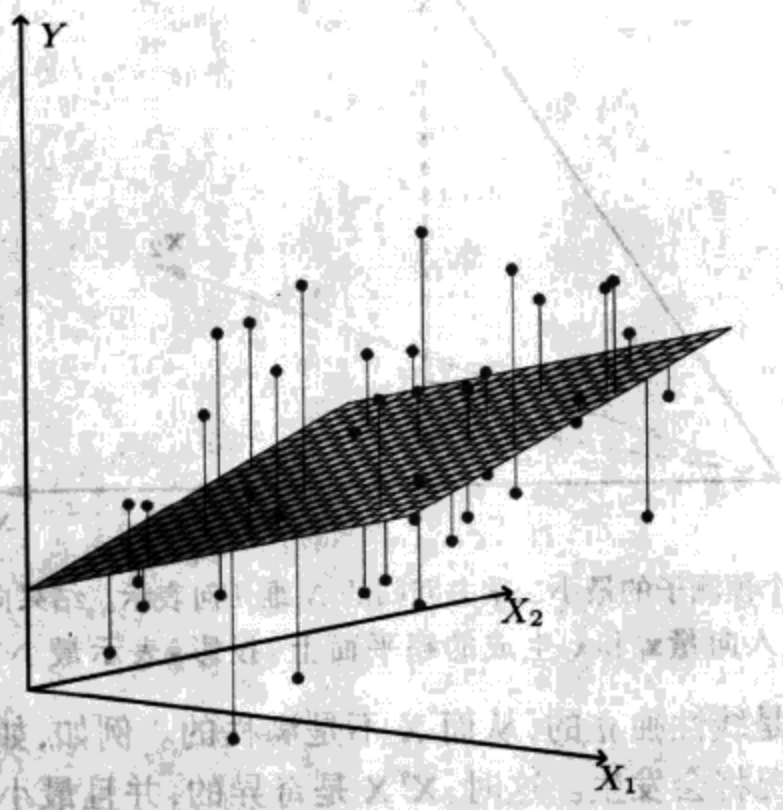


图 3.1　线性最小二乘方拟合($X\in\mathbb{R}^2$)。我们寻找 X 的线性函数，它极小化来自 Y 的残差平方和

如何将式(3.2) 极小化？记 $\mathbf{X}$ 为 $N\times(p+1)$ 矩阵，每行代表一个输入向量(第 1 个位置有一个 1)。类似地，设 $\mathbf{y}$ 是训练数据集里的输出 N 向量，则残差的平方和可以写成：

$$\mathrm{RSS}(\beta)=(\mathbf{y}-\mathbf{X}\beta)^T(\mathbf{y}-\mathbf{X}\beta) \tag{3.3}$$

这是 $p+1$ 个参数的二次函数。关于 β 微分，我们得到：

$$\begin{aligned}\frac{\partial \mathrm{RSS}}{\partial\beta} &= -2\mathbf{X}^T(\mathbf{y}-\mathbf{X}\beta) \\ \frac{\partial^2\mathrm{RSS}}{\partial\beta\partial\beta^T} &= 2\mathbf{X}^T\mathbf{X}\end{aligned} \tag{3.4}$$

暂时假定 $\mathbf{X}$ 是列满秩的，从而 $\mathbf{X}^T\mathbf{X}$ 是正定的。令第一个微分等于零：

$$\mathbf{X}^T(\mathbf{y}-\mathbf{X}\beta)=0 \tag{3.5}$$

得到惟一解：

$$\hat{\beta}=(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y} \tag{3.6}$$

在输入向量 x_0 上的预测值由 $\hat{f}(x_0)=(1:x_0^T)\hat{\beta}$ 给出，在训练输入上的拟合值是：

$$\hat{\mathbf{y}}=\mathbf{X}\hat{\beta}=\mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y} \tag{3.7}$$

其中，$\hat{y}_i=\hat{f}(x_i)$。出现在式(3.7)中的矩阵 $\mathbf{H}=\mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T$ 有时称为“帽”矩阵，因为它在 $\mathbf{y}$ 上加了一个“帽”。

图 3.2 展示了最小二乘方估计的不同几何表示，这次在 $\mathbb{R}^N$ 上。用 $\mathbf{x}_0,\mathbf{x}_1,\cdots,\mathbf{x}_p$ 记 $\mathbf{X}$ 的列向量，其中 $\mathbf{x}_0\equiv\mathbf{1}$。在下面的大部分地方，第一列像其他列那样处理。这些列向量生成 $\mathbb{R}^N$ 的一个子空间，也称 $\mathbf{X}$ 的列空间。通过选取 $\hat{\beta}$ 使得残差向量 $\mathbf{y}-\hat{\mathbf{y}}$ 正交于该子空间，我们对 $\mathrm{RSS}(\beta)=\|\mathbf{y}-\mathbf{X}\beta\|^2$ 极小化。该正交性在式(3.5)中表达，并且结果估计 $\hat{\mathbf{y}}$ 是 $\mathbf{y}$ 在该子空间上的正交投影(orthogonal projection)。帽矩阵 $\mathbf{H}$ 计算该正交投影，因此也称投影矩阵。

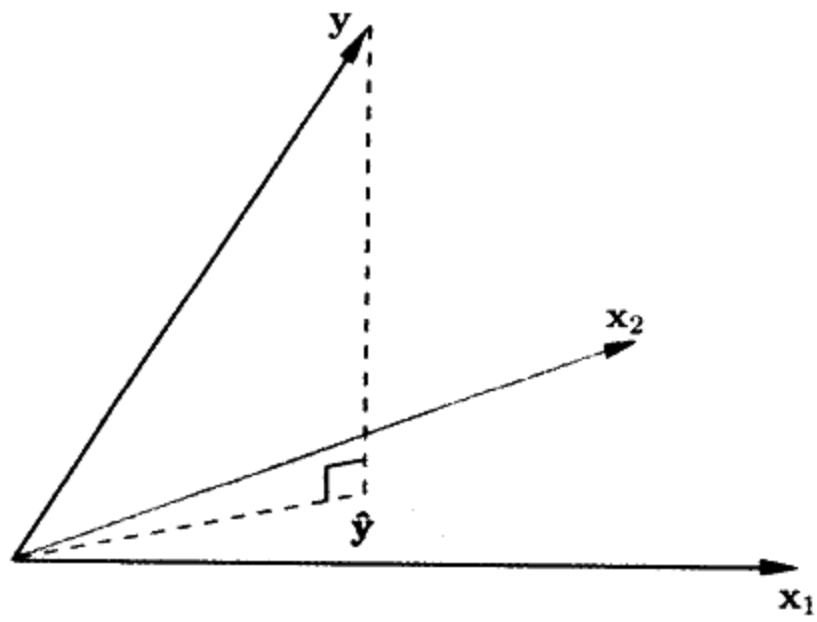


图 3.2 具有两个预测子的最小二乘方回归的 N 维几何表示。结果向量 $\mathbf{y}$ 正交地投影到由输入向量 $\mathbf{x}_1$ 和 $\mathbf{x}_2$ 生成的超平面上。投影 $\hat{\mathbf{y}}$ 表示最小二乘方预测向量

$\mathbf{X}$ 的列向量可能不是线性独立的，从而 $\mathbf{X}$ 不是满秩的。例如，如果两个输入是完全相关的(如 $\mathbf{x}_2=3\mathbf{x}_1$)，这种情况将会发生。这时，$\mathbf{X}^T\mathbf{X}$ 是奇异的，并且最小二乘方系数 $\hat{\beta}$ 不是惟一确定的。然而，拟合值 $\hat{\mathbf{y}}=\mathbf{X}\hat{\beta}$ 仍然是 $\mathbf{y}$ 到 $\mathbf{X}$ 的列空间上的投影，只不过存在多种用 $\mathbf{X}$ 的列向量表示 $\mathbf{y}$ 投影的方法。当一个或多个定性输入使用冗余方式编码时，非满秩的情况更是经常出现。通常，有一种自然的方法解决非惟一表示问题：重新编码或删除 $\mathbf{X}$ 中的冗余列。大部分回归软件包检测这些冗余，并自动地实现某种删除策略。秩亏也可能在信号和图像分析中出现，那里输入个数 p 可能超过训练实例个数 N。在这种情况下，通常使用过滤减少特征，否则用正则化控制拟合(见第 5.2.3 节)。

到目前为止，我们只对数据的实际分布做了极小假定。为了确定 $\hat{\beta}$ 的选样性质，现在假定观测 y_i 是不相关的，并具有常数方差 σ^2，而 x_i 是固定的(非随机的)。最小二乘方参数估计的方差－协方差矩阵容易由式(3.6)导出，并由下式给出：

$$\mathrm{Var}(\hat{\beta})=(\mathbf{X}^T\mathbf{X})^{-1}\sigma^2 \tag{3.8}$$

典型地，方差 σ^2 的估计由下式给出：

$$\hat{\sigma}^2=\frac{1}{N-p-1}\sum_{i=1}^{N}(y_i-\hat{y}_i)^2$$

其中，分母中用 $N-p-1$ 而不是 N 使得 $\hat{\sigma}^2$ 是 σ^2 的无偏估计：$\mathrm{E}(\hat{\sigma}^2)=\sigma^2$。

为了导出关于参数和模型的推论，我们需要附加的假定。现在，假定式(3.1)是均值的正确模型，即 Y 的条件期望在 X_1，…，X_p 上是线性的。还假定 Y 的散离在其期望周围是可加的和高斯的。因此：

$$\begin{aligned} Y &= \mathrm{E}(Y|X_1,\cdots,X_p)+\varepsilon \\ &= \beta_0+\sum_{j=1}^{p}X_j\beta_j+\varepsilon \end{aligned} \tag{3.9}$$

其中，误差 ε 是高斯随机变量，期望为 0，方差为 σ^2，记做 $\varepsilon\sim N(0,\sigma^2)$。

在式(3.9)下，容易证明：

$$\hat{\beta}\sim N(\beta,(\mathbf{X}^T\mathbf{X})^{-1}\sigma^2) \tag{3.10}$$

这是一个多元正态分布，其均值向量和方差－协方差矩阵如上所示。而

$$(N-p-1)\hat{\sigma}^2\sim\sigma^2\chi^2_{N-p-1} \tag{3.11}$$

是具有 $N-p-1$ 自由度的 χ^2 分布。此外，$\hat{\beta}$ 和 $\hat{\sigma}^2$ 是统计独立的。我们使用这些分布性质，形成参数 β_j 的假设检验和置信区间。

为检验特定系数 $\beta_j=0$ 的假设，我们形成标准系数或 Z-得分：

$$z_j=\frac{\hat{\beta}_j}{\hat{\sigma}\sqrt{v_j}} \tag{3.12}$$

其中，v_j 是 $(\mathbf{X}^T\mathbf{X})^{-1}$ 的第 j 个对角线元素。在原假设 $\beta_j=0$ 下，z_j 服从分布 t_{N-p-1}（具有自由度 $N-p-1$ 的 t 分布），因此(绝对值)大的 z_j 值将导致拒绝该原假设。如果 σ 已知，则 z_j 将具有标准正态分布。随着样本数量增加，t 分布的尾分位数和标准正态分布之间的差可以忽略；因此，通常我们使用正态量(见图 3.3)。

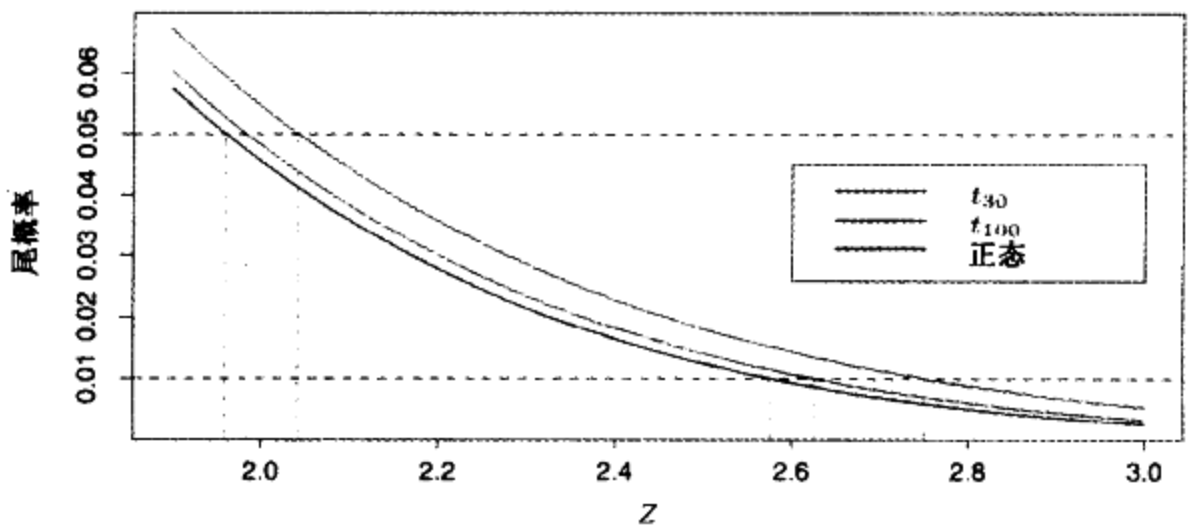


图 3.3　三个分布 t_{30}、t_{100} 和标准正态分布的尾概率 $\Pr(|Z|>z)$。图中显示的是在 $p=0.05$ 和 0.01 水平下的检验显著性的分位数。对于 N 大于 100，t 分布的尾分位数和标准正态之间的差可以忽略

通常，我们需要同时检验一组系数的显著性。例如，为了检验一个 k 级分类变量是否可以从一个模型排除，需要检验用于表示其哑变量的系数是否全都可以置 0。这里使用 F 统计量：

$$F = \frac{(\mathrm{RSS}_0 - \mathrm{RSS}_1)/(p_1 - p_0)}{\mathrm{RSS}_1/(N - p_1 - 1)} \tag{3.13}$$

其中，RSS_1 是具有 $p_1 + 1$ 个参数的较大模型的最小二乘方拟合的残差平方和，而 RSS_0 是嵌套的具有 $p_0 + 1$ 个参数，而 $p_1 - p_0$ 个参数被约束为 0 的较小模型的残差平方和。F 统计量度量较大模型中每个附加的参数导致的残差平方和的改变，并被 σ^2 的估计正态化。在高斯假定和较小模型正确的原假设下，F 统计量具有分布 $F_{p_1-p_0,N-p_1-1}$。可以证明式(3.12)中的 z_j 等价于从模型中删除一个系数 β_j 的 F 统计量(见习题 3.1)。对于较大的 N，$F_{p_1-p_0,N-p_1-1}$ 的分位数逼近 $\chi^2_{p_1-p_0}$ 的分位数。

类似地，我们可以隔离式(3.10)中的 β_j，得到 β_j 的 $1-2\alpha$ 置信区间：

$$(\hat{\beta}_j - z^{(1-\alpha)} v_j^{\frac{1}{2}} \hat{\sigma},\ \hat{\beta}_j + z^{(1-\alpha)} v_j^{\frac{1}{2}} \hat{\sigma}) \tag{3.14}$$

其中，$z^{(1-\alpha)}$ 是正态分布的 $1-\alpha$ 百分位数：

$$\begin{aligned} z^{(1-0.025)} &= 1.96 \\ z^{(1-0.05)} &= 1.645 \text{等} \end{aligned}$$

因此，报告 $\hat{\beta} \pm 2 \cdot \mathrm{se}(\hat{\beta})$ 的标准操作达到将近 95% 的置信区间。即便高斯误差假设不成立，该区间也近似正确，并随样本容量 $N \to \infty$ 而收敛于 $1-2\alpha$。

用类似的方法，我们能够得到整个参数向量 β 的近似置信集，即：

$$C_\beta = \{\beta | (\hat{\beta} - \beta)^T \mathbf{X}^T \mathbf{X} (\hat{\beta} - \beta) \leqslant \hat{\sigma}^2 \chi^{2\ (1-\alpha)}_{p+1}\} \tag{3.15}$$

其中，$\chi^{2\ (1-\alpha)}_{\ell}$ 是自由度 ℓ 上分布的 $1-\alpha$ 百分位数：例如，$\chi^{2\ (1-0.05)}_5 = 11.1$，$\chi^{2\ (1-0.1)}_5 = 9.2$。$\beta$ 的该置信集为真实的函数 $f(x) = x^T\beta$ 产生对应的置信区间，即 $\{x^T\beta | \beta \in C_\beta\}$(见习题 3.2)。关于该置信区间的一个例子，见第 5.2.2 节图 5.4。

3.2.1 例：前列腺癌

该例的数据取自 Stamey 等人(1989)的研究。他们考察准备做前列腺根治手术的病人的前列腺特殊抗原水平与一些临床指标之间的相关性。变量是肿瘤体积记录(lcavol)、前列腺重量记录(lweight)、年龄(age)、良性前列腺增生量(lbph)、精囊浸润(svi)、包膜穿透记录(lcp)、Gleason 积分(gleason)和 Gleason 4 或 5 分所占的百分比(pgg45)。表 3.1 给出的预测子的相关矩阵表现出许多强相关性。第 1 章的图 1.1 是散点图矩阵，显示变量两两之间的图形。我们看到 svi 是二进制变量，而 gleason 是有序的分类变量。例如，看到 lcavol 和 lcp 都表现出与响应 lpsa 的强联系，并且相互之间也具有强联系。我们需要拟合联合效应，揭示预测和响应之间的联系。

表3.1 前列腺癌数据中的预测子的相关性

	lcavol	lweight	age	lbph	svi	lcp	gleason
lweight	0.300						
age	0.286	0.317					
lbph	0.063	0.437	0.287				

（续表）

	lcavol	lweight	age	lbph	svi	lcp	gleason
svi	0.593	0.181	0.129	-0.139			
lcp	0.692	0.157	0.173	-0.089	0.671		
gleason	0.426	0.024	0.366	0.033	0.307	0.476	
pgg45	0.483	0.074	0.276	-0.030	0.481	0.663	0.757

在对预测子规格化，使它们具有单位方差之后，用线性模型拟合前列腺特殊抗原记录 lpsa。我们随机地将数据集分成大小为 67 的训练集和大小为 30 的检验集。对训练集使用最小二乘方估计，产生估值、标准误差和 Z-得分，如表 3.2 所示。Z-得分在式(3.12)中定义，度量从模型中删除变量的效果。Z-得分的绝对值大于 2 在 5% 水平上是近似显著的。（对于我们的例子，有 9 个参数，并且 t_{67-9} 分布的 0.025 尾分位数是 ±2.002!）预测子 lcavol 显示最强的效应，lweight 和 svi 也是强的。注意，一旦 lcavol 在模型中，lcp 就不是显著的（当用于一个不含 lcavol 的模型时，lcp 是强显著的）。使用 F 统计量(3.13)，也能检验一次排除多个项的情况。例如，考虑删除表 3.2 中所有非显著的项，即 age，lcp，gleason 和 pgg45。可以得到：

$$F = \frac{(32.81 - 29.43)/(9 - 5)}{29.43/(67 - 9)} = 1.67 \tag{3.16}$$

它的 p 值为 0.17($\Pr(F_{4,58} > 1.67) = 0.17$)，因此不是显著的。

检验数据上的均值预测误差是 0.545。相比之下，使用 lpsa 的平均训练值的预测具有检验误差 1.050，称为“基本误差率”。因此，线性模型将基本误差率大约降低 50%。稍后，我们将回到该例，比较各种选择和收缩方法。

表 3.2　线性模型拟合前列腺癌数据。Z-得分是系数除其标准误差式(3.12)。粗略地，绝对值大于2的 Z-得分在 $p = 0.05$ 水平上是显著非零的

项	系数	标准误差	Z-得分
截距	2.48	0.09	27.66
lcavol	0.68	0.13	5.37
lweight	0.30	0.11	2.75
age	-0.14	0.10	-1.40
lbph	0.21	0.10	2.06
svi	0.31	0.12	2.47
lcp	-2.9	0.15	-1.87
gleason	-0.02	0.15	-0.15
pgg45	0.27	0.15	1.74

3.2.2 高斯 – 马尔可夫定理

统计学最著名的结果之一断言:在所有的线性无偏估计中,参数 β 的最小二乘方估计具有最小方差。这里将准确地陈述它,并且揭示限制无偏估计不一定是明智的选择。这将引导我们在本章的后面讨论诸如岭回归等有偏估计。我们关注参数 $\theta = a^T\beta$ 的任意线性组合的估计;例如,预测 $f(x_0) = x_0^T\beta$ 是这种形式。$a^T\beta$ 的最小二乘方估计是:

$$\hat{\theta} = a^T\hat{\beta} = a^T(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y} \tag{3.17}$$

考虑 $\mathbf{X}$ 固定,这是响应向量 $\mathbf{y}$ 的线性函数 $\mathbf{c}_0^T\mathbf{y}$。如果假定线性模型是正确的,则 $a^T\hat{\beta}$ 就是无偏的,因为:

$$\begin{aligned} \mathrm{E}(a^T\hat{\beta}) &= \mathrm{E}(a^T(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}) \\ &= a^T(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{X}\beta \\ &= a^T\beta \end{aligned} \tag{3.18}$$

高斯 – 马尔可夫定理告诉我们:如果有 $a^T\beta$ 的其他无偏估计 $\tilde{\theta} = \mathbf{c}^T\mathbf{y}$,即 $\mathrm{E}(\mathbf{c}^T\mathbf{y}) = a^T\beta$,则:

$$\mathrm{Var}(a^T\hat{\beta}) \leqslant \mathrm{Var}(\mathbf{c}^T\mathbf{y}) \tag{3.19}$$

证明(见习题 3.3)使用三角不等式。为简单起见,我们用单个参数 $a^T\beta$ 的估计陈述该结论。但是,借助于一些定义,可以对整个参数向量 β 陈述它(见习题 3.3)。

在估计 θ 时,考虑估计 $\tilde{\theta}$ 的均方误差:

$$\begin{aligned} \mathrm{MSE}(\tilde{\theta}) &= \mathrm{E}(\tilde{\theta} - \theta)^2 \\ &= \mathrm{Var}(\tilde{\theta}) + [\mathrm{E}(\tilde{\theta}) - \theta]^2 \end{aligned} \tag{3.20}$$

第一项是方差,而第二项是平方偏倚。高斯 – 马尔可夫定理暗示在所有的无偏线性估计中,最小二乘方估计具有最小的均方误差。然而,可能存在有偏估计,具有更小的均方误差。这种估计以偏倚的较小增加换取方差较大的减小。有偏估计使用广泛。将最小二乘方的某些系数收缩到 0 或设置为 0,都可能导致有偏估计。在本章的后面,我们讨论一些例子,包括变量子集选择和岭回归。从更实际的角度讲,大部分模型是失真的,因而是有偏的;选取正确的模型旨在取得偏倚和方差之间的平衡。我们将在第 7 章更加详尽地讨论这些问题。

如第 2 章所述,均方误差与预测精度密不可分。考虑在输入点 x_0 的新的响应预测:

$$Y_0 = f(x_0) + \varepsilon_0 \tag{3.21}$$

估计 $\tilde{f}(x_0) = x_0^T\tilde{\beta}$ 的期望预测误差是:

$$\begin{aligned} \mathrm{E}(Y_0 - \tilde{f}(x_0))^2 &= \sigma^2 + \mathrm{E}(x_0^T\tilde{\beta} - f(x_0))^2 \\ &= \sigma^2 + \mathrm{MSE}(\tilde{f}(x_0)) \end{aligned} \tag{3.22}$$

因此,期望预测误差和均方误差只相差一个常数 σ^2,表示新的观测 y_0 的方差。

3.3 从简单的一元回归到多元回归

具有 $p > 1$ 个输入的线性模型(3.1)称为多元线性回归模型(multiple linear regression model)。正如我们在本节将要阐明的,对于一元($p = 1$)线性模型的估计,该模型的最小二乘方

估计(3.6)已被透彻理解。

首先,假定有一个无截距的一元模型,即:

$$Y = X\beta + \varepsilon \tag{3.23}$$

最小二乘方估计和残差是:

$$\begin{aligned} \hat{\beta} &= \frac{\sum_1^N x_i y_i}{\sum_1^N x_i^2} \\ r_i &= y_i - x_i\hat{\beta} \end{aligned} \tag{3.24}$$

用传统的向量记号,令 $\mathbf{y} = (y_1, \cdots, y_N)^T$, $\mathbf{x} = (x_1, \cdots, x_N)^T$, 并定义:

$$\begin{aligned} \langle \mathbf{x}, \mathbf{y} \rangle &= \sum_{i=1}^{N} x_i y_i \\ &= \mathbf{x}^T\mathbf{y} \end{aligned}$$

为 $\mathbf{x}$ 和 $\mathbf{y}$ 的内积(inner product)[①],则有:

$$\begin{aligned} \hat{\beta} &= \frac{\langle \mathbf{x}, \mathbf{y} \rangle}{\langle \mathbf{x}, \mathbf{x} \rangle} \\ \mathbf{r} &= \mathbf{y} - \mathbf{x}\hat{\beta} \end{aligned} \tag{3.25}$$

正如我们将看到的,这个简单的一元回归为多元最小二乘方回归提供了基本构件。假定输入 $\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_p$(数据矩阵 $\mathbf{X}$ 的列)是正交的,即对于所有的 $j \neq k$,有 $\langle \mathbf{x}_j, \mathbf{x}_k \rangle = 0$。容易验证:多元最小二乘方估计 $\hat{\beta}$ 等于 $\langle \mathbf{x}_j, \mathbf{y} \rangle / \langle \mathbf{x}_j, \mathbf{x}_j \rangle$——一元估计。换句话说,当输入是正交的时,模型的参数估计相互间没有影响。

对于平衡的、设计的实验,正交输入最常出现(其中,正交性是强制的),但对于观测数据,输入几乎都不是正交的。因此,我们必须将它们正交化,以便进一步应用这一思想。假定有一个截距和单个输入 $\mathbf{x}$,则 $\mathbf{x}$ 的最小二乘方系数形如:

$$\hat{\beta}_1 = \frac{\langle \mathbf{x} - \bar{x}\mathbf{1}, \mathbf{y} \rangle}{\langle \mathbf{x} - \bar{x}\mathbf{1}, \mathbf{x} - \bar{x}\mathbf{1} \rangle} \tag{3.26}$$

其中,$\bar{x} = \sum_i x_i / N$,而 $\mathbf{1} = \mathbf{x}_0$ 是 N 个观测的向量。可以把估计(3.26)看成简单回归(3.25)两次应用的结果。步骤如下:

1. 在 $\mathbf{1}$ 上对 $\mathbf{x}$ 回归,产生残差 $\mathbf{z} = \mathbf{x} - \bar{x}\mathbf{1}$;
2. 在残差 $\mathbf{z}$ 上对 $\mathbf{y}$ 回归,产生 $\hat{\beta}_1$ 系数。

在该过程中,"在 $\mathbf{a}$ 上对 $\mathbf{b}$ 回归"意指 $\mathbf{a}$ 上无截距的 $\mathbf{b}$ 的简单一元回归,产生系数 $\hat{\gamma} = \langle \mathbf{a}, \mathbf{b} \rangle / \langle \mathbf{a}, \mathbf{a} \rangle$,残差向量 $\mathbf{b} - \hat{\gamma}\mathbf{a}$。我们称 $\mathbf{b}$ 是 $\mathbf{a}$ 的调整,或关于 $\mathbf{a}$ 的"正交化"。

步骤 1 关于 $\mathbf{x}_0 = 1$ 正交化 $\mathbf{x}$。步骤 2 是使用正交预测子 $\mathbf{1}$ 和 $\mathbf{z}$ 的简单一元回归。图 3.4 对两个一般的输入 $\mathbf{x}_1$ 和 $\mathbf{x}_2$ 图示了该过程。正交化不改变 $\mathbf{x}_1$ 和 $\mathbf{x}_2$ 生成的子空间,只是产生一个表示该子空间的正交基。

---

① 内积符号启发将线性回归拓广到不同的度量空间和概率空间。

该方法拓广到 p 个输入的情况，如算法 3.1 所示。注意，步骤 2 中的输入 $\mathbf{z}_0, \mathbf{z}_1, \cdots, \mathbf{z}_{j-1}$ 是正交的，因此所计算的简单回归系数实际上也是多元回归系数。

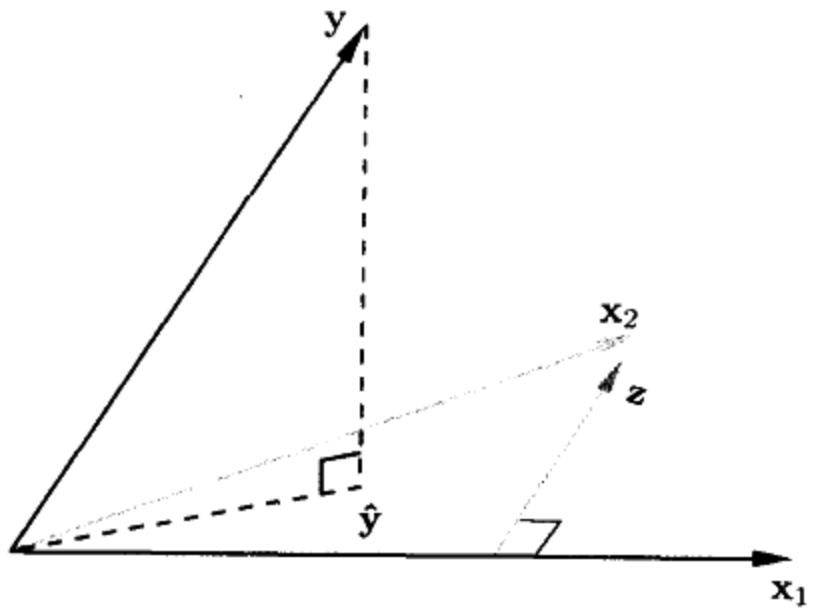


图 3.4 通过输入正交化的最小二乘方回归。向量 $\mathbf{x}_2$ 在向量 $\mathbf{x}_1$ 上回归，残差为向量 $\mathbf{z}$。$\mathbf{z}$ 上 $\mathbf{y}$ 的回归产生 $\mathbf{x}_2$ 的多元回归系数。$\mathbf{y}$ 在 $\mathbf{x}_1$ 和 $\mathbf{z}$ 上的投影加在一起，产生最小二乘方拟合 $\hat{\mathbf{y}}$

算法 3.1 通过相继正交化回归

1. 初始化 $\mathbf{z}_0 = \mathbf{x}_0 = 1$

2. 对于 $j = 1, 2, \cdots, p$

   在 $\mathbf{z}_0, \mathbf{z}_1, \cdots, \mathbf{z}_{j-1}$ 上对 $\mathbf{x}_j$ 回归，产生系数 $\hat{\gamma}_{\ell j} = \langle \mathbf{z}_\ell, \mathbf{x}_j \rangle / \langle \mathbf{z}_\ell, \mathbf{z}_\ell \rangle$，其中 $\ell = 0, 1, \cdots, j-1$，并产生残差向量 $\mathbf{z}_j = \mathbf{x}_j - \sum_{k=0}^{j-1} \hat{\gamma}_{kj} \mathbf{z}_{k-1}$

3. 在残差 $\mathbf{z}_p$ 上对 $\mathbf{y}$ 回归，产生估计 $\hat{\beta}_p$

该算法的结果是：

$$\hat{\beta}_p = \frac{\langle \mathbf{z}_p, \mathbf{y} \rangle}{\langle \mathbf{z}_p, \mathbf{z}_p \rangle} \tag{3.27}$$

重新安排步骤 2 中的残差，我们看到每个 $\mathbf{x}_j$ 是 $\mathbf{z}_k (k \leqslant j)$ 的线性组合。由于 $\mathbf{z}_j$ 都是正交的，它们形成 $\mathbf{X}$ 的列空间的基，因此到该子空间的最小二乘方投影是 $\hat{\mathbf{y}}$。由于只有 $\mathbf{x}_p$ 涉及 $\mathbf{z}_p$（系数为 1），我们看到系数(3.27)确实是 $\mathbf{y}$ 在 $\mathbf{x}_p$ 上的多元回归系数。这一关键结果揭示多元回归中相关输入的影响。注意，同样是重新安排 $\mathbf{x}_j$，它们中的任何一个都可以在最后一个位置，并且类似的结果成立。因此，更一般地说，我们证明了第 j 个多元回归系数是 $\mathbf{y}$ 在 $\mathbf{x}_{j \cdot 012 \cdots (j-1)(j+1) \cdots, p}$ 上的一元回归系数，在 $\mathbf{x}_0, \mathbf{x}_1, \cdots, \mathbf{x}_{j-1}, \mathbf{x}_{j+1}, \cdots, \mathbf{x}_p$ 上对 $\mathbf{x}_j$ 回归后的残差：

在 $\mathbf{x}_j$ 关于 $\mathbf{x}_0, \mathbf{x}_1, \cdots, \mathbf{x}_{j-1}, \mathbf{x}_{j+1}, \cdots, \mathbf{x}_p$ 调整后，多元回归系数 $\hat{\beta}_j$ 提供 $\mathbf{x}_j$ 在 $\mathbf{y}$ 上的附加分布。

如果 $\mathbf{x}_p$ 与其他的某些 $\mathbf{x}_k$ 高度相关，残差向量 $\mathbf{z}_p$ 将接近于 0。由式(3.27)，系数 $\hat{\beta}_p$ 将很不稳定。对于相关集中的所有变量也都如此。由式(3.27)，还可以得到方差估计(3.8)的一个替代公式：

$$\mathrm{Var}(\hat{\beta}_p) = \frac{\sigma^2}{\langle \mathbf{z}_p, \mathbf{z}_p \rangle} = \frac{\sigma^2}{\|\mathbf{z}_p\|^2} \tag{3.28}$$

换言之，估计 $\hat{\beta}_p$ 可以达到的精度取决于向量 $\mathbf{z}_p$ 的长度；这表明 $\mathbf{x}_p$ 在多大程度上不能用其他的 $\mathbf{x}_k$ 解释。

算法 3.1 称做多元回归的 Gram -Schmidt 过程，并且也是计算估计的有用数值策略。从中可以得到的不仅是 $\hat{\beta}_p$，还能得到整个多元最小二乘方拟合，如习题 3.4 所示。

算法 3.1 的步骤 2 可以用矩阵形式表示：

$$\mathbf{X} = \mathbf{Z}\Gamma \tag{3.29}$$

其中，$\mathbf{Z}$(依次)以 $\mathbf{z}_j$ 为列，而 Γ 是上三角矩阵，具有元素值 $\hat{\gamma}_{kj}$。引入对角矩阵 $\mathbf{D}$，其中 $\mathbf{D}$ 的第 j 个对角线元素为 $D_{jj} = \|\mathbf{z}_j\|$，得到：

$$\begin{aligned} \mathbf{X} &= \mathbf{Z}\mathbf{D}^{-1}\mathbf{D}\Gamma \\ &= \mathbf{Q}\mathbf{R} \end{aligned} \tag{3.30}$$

即 $\mathbf{X}$ 的 QR 分解。这里，$\mathbf{Q}$ 是 $N \times (p+1)$的正交矩阵，$\mathbf{Q}^T\mathbf{Q} = \mathbf{I}$，而 $\mathbf{R}$ 是$(p+1) \times (p+1)$的上三角矩阵。

QR 分解为 $\mathbf{X}$ 的列空间提供了一个方便的正交基。例如，容易看出最小二乘方解由下式给出：

$$\hat{\beta} = \mathbf{R}^{-1}\mathbf{Q}^T\mathbf{y} \tag{3.31}$$

$$\hat{\mathbf{y}} = \mathbf{Q}\mathbf{Q}^T\mathbf{y} \tag{3.32}$$

式(3.31)容易求解，因为 $\mathbf{R}$ 是上三角矩阵(见习题 3.4)。

3.3.1　多元输出

假定有多个输出 $Y_1, Y_2, \cdots, Y_K$，希望由输入 $X_0, X_1, X_2, \cdots, X_p$ 预测它们。假设对于每个输出，我们有一个线性模型：

$$Y_k = \beta_{0k} + \sum_{j=1}^{p} X_j\beta_{jk} + \varepsilon_k \tag{3.33}$$

$$= f_k(X) + \varepsilon_k \tag{3.34}$$

给定 N 个训练实例，可以将该模型写成矩阵形式：

$$\mathbf{Y} = \mathbf{X}\mathbf{B} + \mathbf{E} \tag{3.35}$$

这里，$\mathbf{Y}$ 是 $N \times K$ 响应矩阵，第 ik 项为 y_{ik}，$\mathbf{X}$ 是 $N \times (p+1)$输入矩阵，$\mathbf{B}$ 是$(p+1) \times K$ 参数矩阵，而 $\mathbf{E}$ 是 $N \times K$ 误差矩阵。一元损失函数(3.2)的一个直接拓广是：

$$\mathrm{RSS}(\mathbf{B}) = \sum_{k=1}^{K}\sum_{i=1}^{N}(y_{ik} - f_k(x_i))^2 \tag{3.36}$$

$$= \mathrm{tr}[(\mathbf{Y} - \mathbf{X}\mathbf{B})^T(\mathbf{Y} - \mathbf{X}\mathbf{B})] \tag{3.37}$$

最小二乘方估计与先前的具有完全相同的形式：

$$\hat{\mathbf{B}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y} \tag{3.38}$$

因此，第 k 个结果系数恰好是 $\mathbf{y}_k$ 在 $\mathbf{x}_0, \mathbf{x}_1, \cdots, \mathbf{x}_p$ 上的回归的最小二乘方估计。多元输出不影响其他输出的最小二乘方估计。

如果式(3.33)中的误差 $\varepsilon = (\varepsilon_1, \cdots, \varepsilon_K)$ 是相关的，则修改式(3.36)，以适合于多元版本看来是合理的。特殊地，假定 $\mathrm{Cov}(\varepsilon) = \mathbf{\Sigma}$，则多元加权准则：

$$\mathrm{RSS}(\mathbf{B}; \mathbf{\Sigma}) = \sum_{i=1}^{N} (y_i - f(x_i))^T \mathbf{\Sigma}^{-1} (y_i - f(x_i)) \tag{3.39}$$

自然地由多元高斯定理产生。这里，$f(x)$ 是向量函数 $(f_1(x), \cdots, f_K(x))$，而 y_i 是观测 i 的 K 个响应向量。然而，可以证明这个解也由式(3.38)给出；K 个分离的回归，忽略相关性(见习题3.9)。如果 $\mathbf{\Sigma}_i$ 在观测中是变化的，情况就截然不同，$\mathbf{B}$ 的解不再是可分解的。

在第 3.4.6 节中将继续讨论多元输出问题，并考虑可以把回归组合的情况。

3.4 子集选择和系数收缩

有两个原因，使得我们常常对最小二乘方估计(3.6)不满意。

- 第一个原因是预测精度(prediction accuracy)：最小二乘方估计通常具有低偏倚和高方差。有时可以通过将某些系数收缩到 0 或设置为 0 来提高预测精度。通过这样的处理，牺牲一些偏倚，而降低被预测值的方差，从而提高总体预测精度。
- 第二个原因是解释(interpretation)：存在大量预测子时，通常希望确定一个表现出最强影响的较小子集。为了得到“大印象”，我们情愿牺牲某些小的细节。

本节将介绍一些变量选择和系数收缩方法。

3.4.1 子集选择

在该方法下只保留变量的一个子集，而将其余变量从模型中删除。最小二乘方回归用来评估留下的输入的系数。有多种不同策略用于选择子集。最佳子集回归(best subset regression)对每个 $k \in \{0, 1, 2, \cdots, p\}$，找出的容量为 k 的子集，它们具有最小残差平方和(3.2)。一个有效的算法——跳跃和约束(leaps and bounds)过程(Furnival和Wilson，1974)——使得它对于高达 30 或 40 的 p 是可行的。图 3.5 显示前列腺癌例子的所有子集模型。较低的边界表示应当由最佳子集方法选择模型。注意，容量为 2 的最佳子集不必包含容量为 1 的最佳子集中的变量(对于该例，所有的子集是嵌套的)。最佳子集曲线(图 3.5 中红色边界)必然是递减的，因此不能用来选取子集容量 k。如何选取 k 的问题涉及偏倚和方差之间的权衡，并且有许多可用的准则。典型地，我们这样选取模型，它极小化期望预测误差的估计。我们将该问题的讨论推迟到第 7 章。

可以寻找一条通过可能子集的好的路径，而不是搜索所有可能的子集(对于比 40 大很多的 p，这是不可行的)。逐步前向选择(forward stepwise selection)由截距开始，并依次将对拟合改进最大的预测子添加到模型中。假定当前模型有 k 个输入，用参数估计 $\hat{\beta}$ 表示，并且添加一个预测子导致估计 $\tilde{\beta}$。拟合的改进通常基于 F 统计量(3.13)，

$$F = \frac{\mathrm{RSS}(\hat{\beta}) - \mathrm{RSS}(\tilde{\beta})}{\mathrm{RSS}(\tilde{\beta})/(N - k - 2)} \tag{3.40}$$

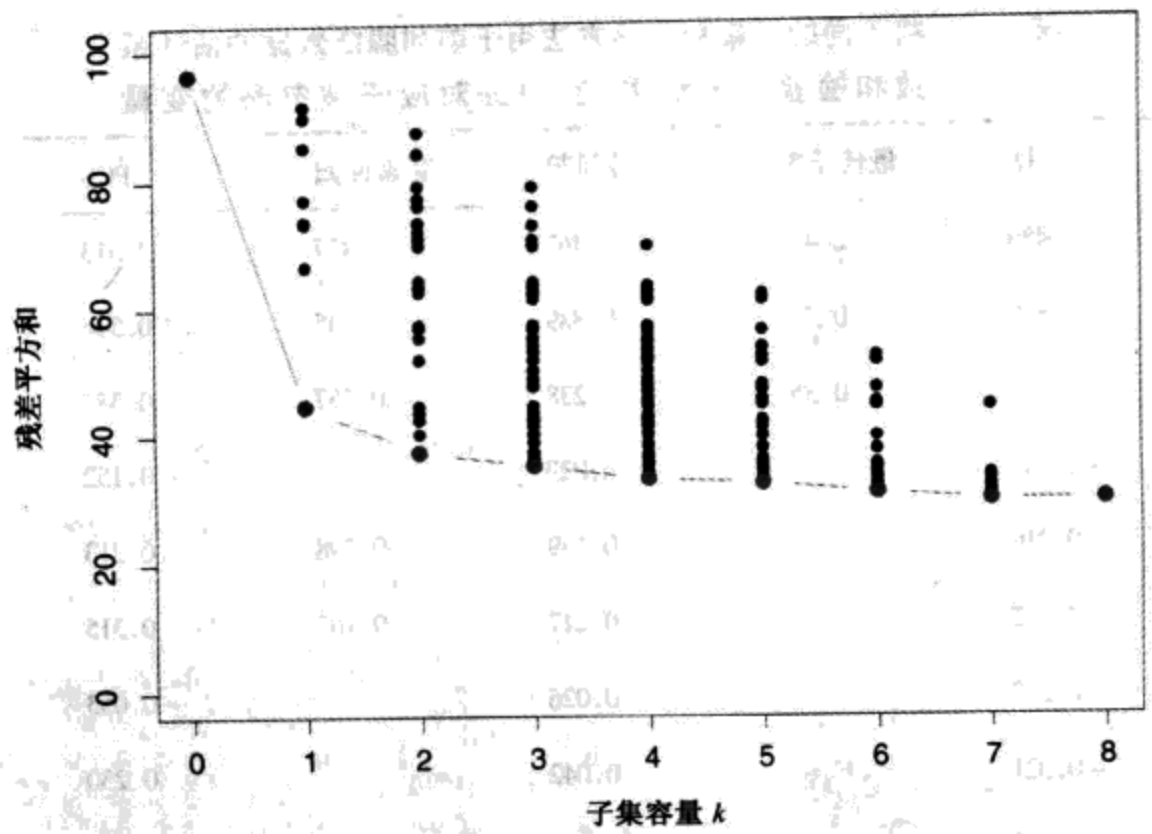


图3.5　前列腺癌例子中所有可能的子集模型。对每个子集容量，显示该容量的每个模型的残差平方和(见彩页)

一个典型的策略是顺序地添加产生最大 F 值的预测子。当没有一个预测子产生的 F-比率大于 $F_{1,N-k-2}$ 分布的第90个或第95个百分位数时停止。

逐步后向选择(backward stepwise selection)从整个模型开始，并依次删除预测子。与前向选择一样，它使用如式(3.40)的 F-比率选取待删除的预测子。在此情况下，我们每一步都删除产生最小 F 值的预测子；当删除后模型中的每个预测子产生的 F 值都大于第90个或第95个百分位数时停止。后向选择仅当 $N > p$ 时才能使用，而前向选择总是可以使用的。还有混合逐步选择策略，它在每一步同时考虑前向和后向移动，并做“最好”的移动。这需要一个参数来设置“添加”移动超过“删除”移动的阈值。

F-比率停止规则只提供了模型搜索的局部控制，并不试图在所考察的模型序列中找到最好的模型。使用所有子集选择，可以从序列中选取极小化期望预测误差的模型。这在第7章讨论，并用下面的例子解释。

3.4.2　前列腺癌数据例子(续)

表3.3列出取自一些不同的选择和收缩模型的系数。它们是使用搜索所有子集的最佳子集选择、岭回归、套索(lasso)、主成分回归和部分最小二乘方。每个模型都有一个复杂性参数，并且它的选取是为了极小化基于10折交叉验证的预测误差估计；交叉验证的细节将在第7.10节给出。简略地说，10折交叉验证随机地将训练数据分成10个相等的部分。学习方法拟合数据的十分之九，而预测误差在剩下的十分之一上计算。依次对每份十分之一数据执行这一过程，并对十个预测误差估计取平均值。注意，我们已经将这些数据分成容量为67的训练集和容量为30的检验集。交叉验证用于训练集，这是因为选择收缩参数也是训练过程的一部分。在这里，检验集用于评估所选模型的性能。

表 3.3 将不同的子集和收缩方法用于前列腺癌数据的估计系数和检验误差结果。空白处对应于被忽略的变量

项	LS	最佳子集	岭回归	套索回归	PCR	PLS
截距	2.480	2.495	2.467	2.477	2.513	2.452
lcavol	0.680	0.740	0.389	0.545	0.544	0.440
lweight	0.305	0.367	0.238	0.237	0.337	0.351
age	-0.141		-0.029		-0.152	-0.017
lbph	0.210		0.159	0.098	0.213	0.248
svi	0.305		0.217	0.165	0.315	0.252
lcp	-0.288		0.026		-0.053	0.078
gleason	-0.021		0.042		0.230	0.003
pgg45	0.267		0.123	0.059	-0.053	0.080
检验误差	0.586	0.574	0.540	0.491	0.527	0.636
标准误差	0.184	0.156	0.168	0.152	0.122	0.172

估计预测误差曲线在图 3.6 中给出。大部分曲线在其最小值附近的大范围内是非常平坦的。图中包含每个估计误差率的标准误差频带，它们基于使用交叉验证计算的 10 个误差估计。使用“一个标准误差”(one-standard-error)规则——选取在一个最小标准差之内的最节省的模型(见第 7.10 节)。这样的规则正视现实：折中曲线是有误差的估计，并因而采取保守的方法。

最佳子集选择选取 lcavol 和 lweight 两个预测子。表 3.3 的最后两行给出检验集上的平均预测误差(及其标准误差)。

3.4.3 收缩方法

通过保留预测子的一个子集而丢弃其他预测子，子集选择产生一个模型。该模型是可解释的，并可能具有比完整模型更低的预测误差。然而，由于它是一个离散过程(变量或者保留，或者丢弃)，它常常表现出高方差，因此不能降低整个模型的预测误差。收缩方法更连续，并且不会因变量多而过多地降低性能。

岭回归

岭回归(ridge regression)通过对其容量加罚来收缩回归系数。岭系数极小化罚残差平方和，

$$\hat{\beta}^{\text{ridge}} = \underset{\beta}{\operatorname{argmin}} \left\{ \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j\right)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \right\} \tag{3.41}$$

这里，$\lambda \geqslant 0$ 是控制收缩量的复杂度参数：λ 值越大，收缩量越大。系数向 0 收缩(并相互收缩)。通过参数的平方和来加罚的思想也用于神经网络，那里称为权衰减(weight decay)(参见第 11 章)。

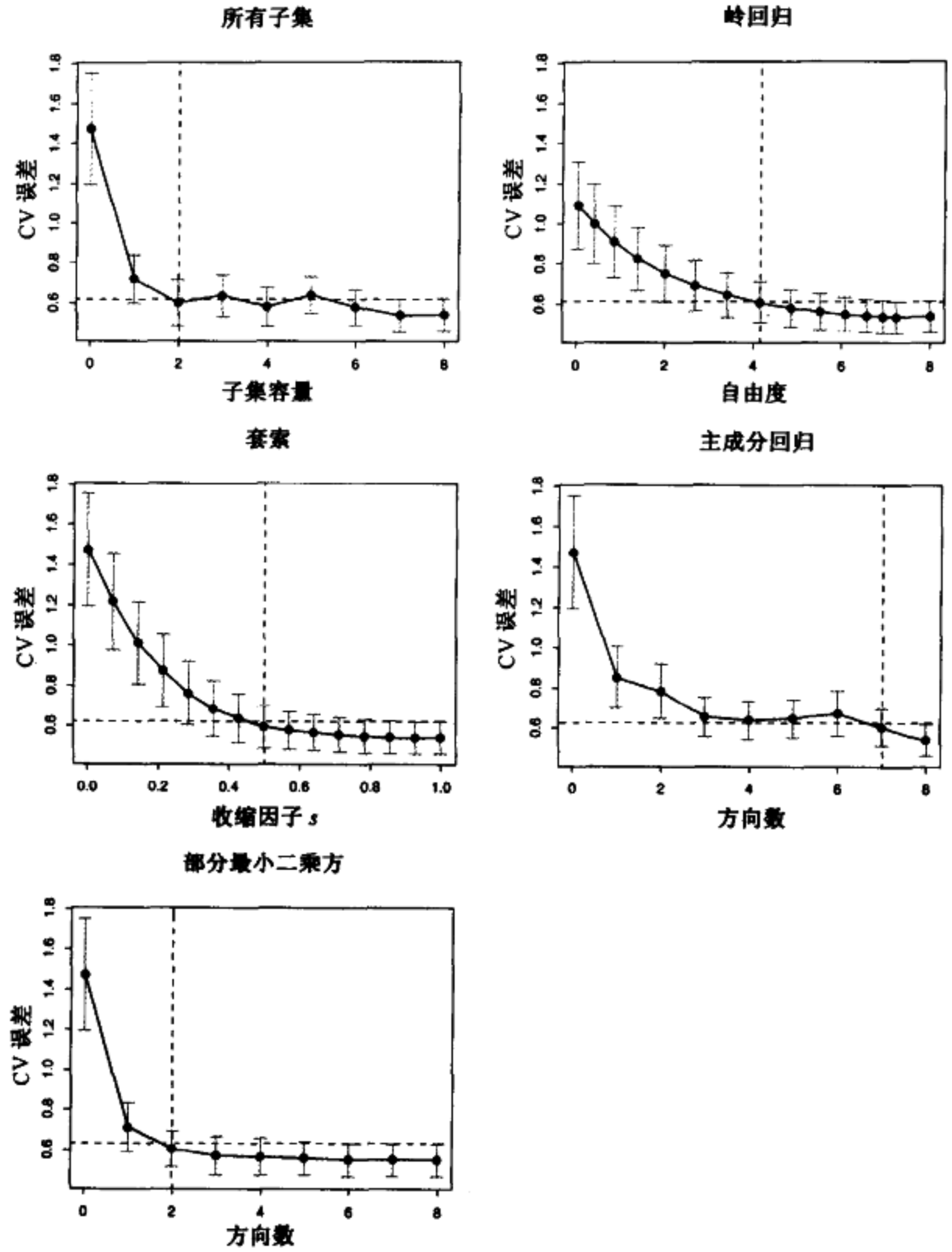


图 3.6　不同选择和收缩方法的估计预测误差曲线及其标准误差。每条曲线是对应模型复杂性参数的函数 。水平轴的选取使得模型的复杂性从左到右递增。预测误差估计和它们的标准误差通过10折交叉验证得到；细节将在第7.10节给出。选取在一个最佳标准误差之内的复杂度最小的模型

表达岭回归问题的一个等价方法是：

$$\hat{\beta}^{\text{ridge}} = \underset{\beta}{\operatorname{argmin}} \sum_{i=1}^{N} \Big(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \Big)^2 \quad \text{受限于} \sum_{j=1}^{p} \beta_j^2 \leqslant s \tag{3.42}$$

这清楚地表达了参数上的量约束。在式(3.41)的参数 λ 和式(3.42)的 s 之间存在一个一一对应。当线性回归模型中存在多个相关变量时，它们的系数确定性变差，并呈现高方差。在一个

变量上的很大的正系数可能被在其相关变量上类似大小的负系数抵消。通过在系数上施加一个量约束，如式(3.42)，可以避免这种现象发生。

在输入缩放时，岭解是不等价的，因此在解式(3.41)之前，通常要对输入标准化。

此外，注意截距 β_0 被排除在罚项之外。截距的罚将使得过程依赖于 Y 的原点选择；即，给每个目标y_i 加上一个常数 c 将不会简单地导致结果移动相同的量 c。可以证明(见习题 3.5)使用中心化的输入(每个 x_{ij}用 $x_{ij}-\bar{x}_j$ 替换)重新设置参数之后，式(3.41)可以分成两部分。我们用 $\bar{y}=\sum_1^N y_i/N$ 估计β_0。其余的系数使用中心化的 x_{ij}，通过无截距的岭回归估计得到。此后，假定已进行中心化，从而输入矩阵 $\mathbf{X}$ 有 p 列(而不是 $p+1$ 列)。

将准则(3.41)写成矩阵形式，

$$\mathrm{RSS}(\lambda)=(\mathbf{y}-\mathbf{X}\beta)^T(\mathbf{y}-\mathbf{X}\beta)+\lambda\beta^T\beta \tag{3.43}$$

容易看出岭回归的解是：

$$\hat{\beta}^{\mathrm{ridge}}=(\mathbf{X}^T\mathbf{X}+\lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y} \tag{3.44}$$

其中，$\mathbf{I}$ 是 $p\times p$ 的单位矩阵。注意，选取二次罚 $\beta^T\beta$，岭回归的解又是 $\mathbf{y}$ 的线性函数。这个解在 $\mathbf{X}^T\mathbf{X}$ 反演之前，将一个正常数加到 $\mathbf{X}^T\mathbf{X}$ 的对角线上。这使得问题非奇异，即使 $\mathbf{X}^T\mathbf{X}$ 不是满秩的。这正是当初统计学引进岭回归的主要动机(Hoerl 和 Kennard，1970)。传统地，岭回归的介绍从定义式(3.44)开始。我们通过式(3.41)和式(3.42)诱导它，因为这样能够洞察岭回归如何工作。

图 3.7 给出前列腺癌例子的岭系数估计，曲线作为 df (λ)的函数绘制。df (λ)是罚 λ 蕴涵的有效自由度(effective degrees of freedom)[定义在式(3.50)]。

对于正交输入，岭回归估计只不过是最小二乘方估计的缩放版本；即 $\hat{\beta}^{\mathrm{ridge}}=\gamma\hat{\beta}$。这里，$0\leqslant\gamma\leqslant1$是式(3.41)中 λ 的简单函数；详见第 3.4.5 节。

适当选择先验分布，岭回归也可以作为后验分布的均值或众数导出。设$y_i\sim N(\beta_0+x_i^T\beta,\sigma^2)$，每个参数 β_j 都独立地分布在$N(0,\tau^2)$上。假定τ^2 和 σ^2 已知，则 β 的(负)对数后验密度等于式(3.41)花括号中的表达式，其中 $\lambda=\sigma^2/\tau^2$(见习题 3.6)。这样，岭估计是该后验分布的众数；由于该分布是高斯分布，它也是后验均值。

中心化输入矩阵 $\mathbf{X}$ 的奇异值分解(singular value decomposition，SVD)使我们可以进一步洞察岭回归的特点。该分解在许多统计方法的分析中都特别有用。$N\times p$ 矩阵 $\mathbf{X}$ 的 SVD 具有如下形式：

$$\mathbf{X}=\mathbf{U}\mathbf{D}\mathbf{V}^T \tag{3.45}$$

这里，$\mathbf{U}$ 和 $\mathbf{V}$ 是 $N\times p$ 和$p\times p$ 正交矩阵，$\mathbf{U}$ 的列生成 $\mathbf{X}$ 的列空间，而 $\mathbf{V}$ 的列生成 $\mathbf{X}$ 的行空间。$\mathbf{D}$ 是 $p\times p$ 对角矩阵，对角线上元素 $d_1\geqslant d_2\geqslant\cdots\geqslant d_p\geqslant0$ 称做 $\mathbf{X}$ 的奇异值。

使用奇异值分解，经过某些简化，可以将最小二乘方拟合向量写成：

$$\begin{aligned}\mathbf{X}\hat{\beta}^{\mathrm{ls}} &= \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}\\ &= \mathbf{U}\mathbf{U}^T\mathbf{y}\end{aligned} \tag{3.46}$$

注意，$\mathbf{U}^T\mathbf{y}$ 是 $\mathbf{y}$ 关于正交基 $\mathbf{U}$ 的坐标。此外，注意与式(3.32)的类似性；$\mathbf{Q}$ 和 $\mathbf{U}$ 一般是 $\mathbf{X}$ 的列空间的不同基(见习题 3.8)。

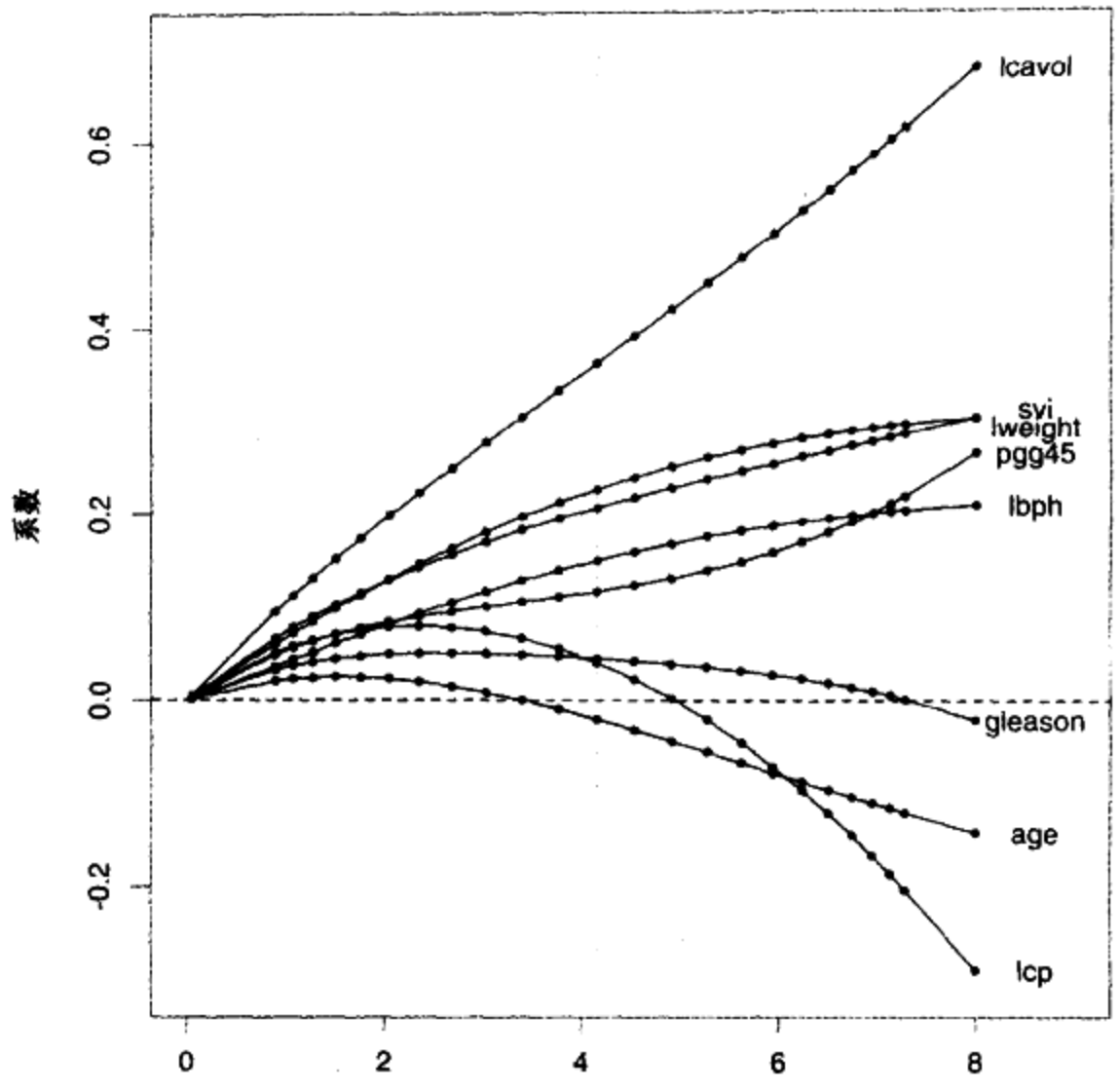


图 3.7　前列腺癌例子的岭系数随协调参数 λ 变化的曲线。系数按有效自由度 df(λ) 绘出。一条竖线绘在 df = 4.16 处，这是交叉验证选取的值

现在，岭解是：

$$\begin{aligned}\mathbf{X}\hat{\beta}^{\text{ridge}} &= \mathbf{X}(\mathbf{X}^T\mathbf{X}+\lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y} \\ &= \mathbf{U}\,\mathbf{D}(\mathbf{D}^2+\lambda\mathbf{I})^{-1}\mathbf{D}\,\mathbf{U}^T\mathbf{y} \\ &= \sum_{j=1}^{p}\mathbf{u}_j\frac{d_j^2}{d_j^2+\lambda}\mathbf{u}_j^T\mathbf{y}\end{aligned} \tag{3.47}$$

其中，$\mathbf{u}_j$ 是 $\mathbf{U}$ 的列。注意，由于 $\lambda \geqslant 0$，我们有 $d_j^2/(d_j^2+\lambda) \leqslant 1$。和线性回归一样，岭回归计算 $\mathbf{y}$ 关于正交基 $\mathbf{U}$ 的坐标。然后按因子 $d_j^2/(d_j^2+\lambda)$ 收缩这些坐标。这意味较大的收缩量用在具有较小 d_j^2 的基向量。

一个小 d_j^2 值意味着什么？中心化矩阵 $\mathbf{X}$ 的 SVD 是表示 $\mathbf{X}$ 中变量的主成分(principal components)的另一种方式。样本的协方差矩阵由 $\mathbf{S}=\mathbf{X}^T\mathbf{X}/N$ 给出，而由式(3.45)有：

$$\mathbf{X}^T\mathbf{X} = \mathbf{V}\mathbf{D}^2\mathbf{V}^T \tag{3.48}$$

这是 $\mathbf{X}^T\mathbf{X}$ 的本征分解(eigen decomposition)(也是 $\mathbf{S}$ 的本征分解，相差一个因子 N)。本征向量 v_j 也称 $\mathbf{X}$ 的主成分(或 Karhumen-Loeve)方向。第一个主成分方向 v_1 具有如下性质：*在 $\mathbf{X}$ 的列*的所有正规化线性组合中，$\mathbf{z}_1=\mathbf{X}v_1$ 具有最大的样本方差。容易看出，该样本方差是：

$$\mathrm{Var}(\mathbf{z}_1) = \mathrm{Var}(\mathbf{X}v_1) = \frac{d_1^2}{N} \tag{3.49}$$

并且事实上 $\mathbf{z}_1 = \mathbf{X}v_1 = \mathbf{u}_1 d_1$。导出变量 $\mathbf{z}_1$ 称为 $\mathbf{X}$ 的第一主成分，并因此称 $\mathbf{u}_1$ 是正规化的第一主成分。随后的主成分 $\mathbf{z}_j$ 最大方差为 d_j^2/N，受到与前面的主成分正交约束。反过来，最后的主成分具有最小方差。因此，最小的奇异值 d_j 对应于 $\mathbf{X}$ 的列空间中具有最小方差的方向，而岭回归在这些方向收缩最多。

图 3.8 展示二维空间某些数据点的主成分。如果考虑该区域上的线性曲面拟合（Y 轴垂直于页面），数据的布局允许我们在长方向上比在短方向上以更高的精度确定它的梯度。岭回归防止短方向上梯度估计可能出现高方差。暗含的假定是：在输入的高方差方向，响应趋向于变化最大。通常，这是一个合理的假定，但并不一般地成立。

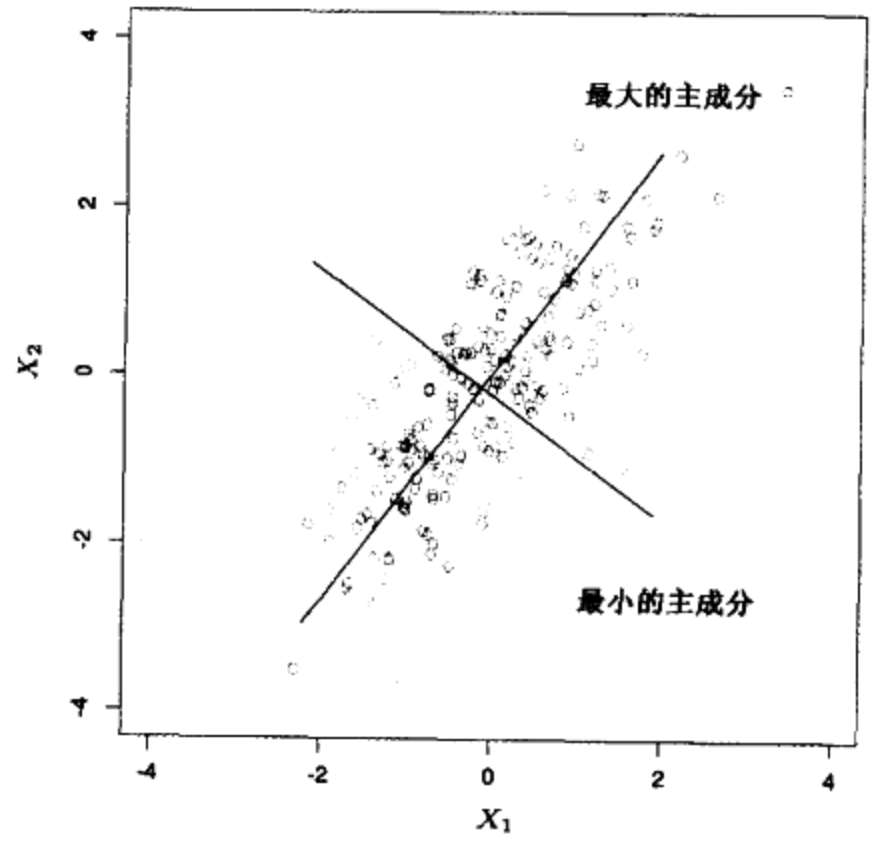


图 3.8 一些输入数据点的主成分。最大的主成分是最大化投影数据方差的方向，而最小的主成分最小化该方差。岭回归将y投影到这些成分，然后收缩系数，低方差的成分比高方差的成分收缩更多

在图 3.6 中，我们已经绘制了估计预测误差关于量

$$\begin{aligned} \mathrm{df}(\lambda) &= \mathrm{tr}[\mathbf{X}(\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T] \\ &= \sum_{j=1}^{p} \frac{d_j^2}{d_j^2 + \lambda} \end{aligned} \tag{3.50}$$

的关系曲线。正如第 7.6 节讨论的，这个单调减函数是岭回归拟合的有效自由度。注意，当 $\lambda = 0$（非正则化）时，$\mathrm{df}(\lambda) = p$，并且随 $\lambda \to \infty$，$\mathrm{df}(\lambda) \to 0$。在图 3.6 中，最小值出现在 $\mathrm{df}(\lambda) = 4.16$。表 3.3 显示岭回归将整个最小二乘方估计的检验误差降低了一点。

套索

套索（lasso）是一种收缩方法，像岭回归一样，但具有微妙而重要的差别。套索估计由下式定义：

$$\begin{aligned}\hat{\beta}^{\text{lasso}} &= \underset{\beta}{\operatorname{argmin}} \sum_{i=1}^{N}\Big(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j\Big)^2 \\ &\text{受限于} \sum_{j=1}^{p} |\beta_j| \leqslant t\end{aligned} \tag{3.51}$$

正如岭回归，我们可以通过将预测子标准化来对常量 β_0 重新参数化；$\hat{\beta}_0$ 的解是 $\bar{y}$，而此后，我们拟合不含截距的模型(见习题 3.11)。

注意与岭回归问题(3.42)的类似性：L_2 岭罚 $\sum_1^p \beta_j^2$ 被 L_1 套索罚 $\sum_1^p |\beta_j|$ 取代。后一个约束使得解在 y_i 上非线性，并使用二次规划算法计算它们。由于该约束的特性，使得 t 充分小将导致某些系数恰好为 0。这样，套索做了某种连续的子集选择。如果选取 t 大于 $t_0 = \sum_1^p |\hat{\beta}_j|$ (其中，$\hat{\beta}_j = \hat{\beta}_j^{\text{ls}}$，最小二乘方估计)，则套索估计是这些 $\hat{\beta}_j$ 的估计。另一方面，例如对于 $t = t_0/2$，最小二乘方系数平均收缩大约 50%。然而，这种收缩的特性并非显而易见，我们将在第 3.4.5 节进一步考察它。像变量子集选择的子集容量，或岭回归的罚参数一样，t 应当自适应地选取，从而极小化期望预测误差估计。

在图 3.6 中，为便于解释，我们绘制了套索预测误差估计关于标准化参数 $s = t/\sum_1^p |\hat{\beta}_j|$ 的曲线。值 $\hat{s} \approx 0.50$ 被十折交叉验证选取；这导致将三个系数设置为 0(见表 3.3 的第 5 列)。结果模型具有最低的检验误差，略低于完全最小二乘方模型，但是检验误差估计的标准误差(见表 3.3 的最后一行)相当大。

图 3.9 显示随标准化调整参数 $s = t/\sum_1^p |\hat{\beta}_j|$ 变化，套索系数的变化。在 $s = 1.0$，这是最小二乘方估计；随 $s \to 0$，它们递减到 0。该递减不是严格单调的，尽管在该例如此。在 $s = 0.5$(该值由交叉验证选取)处画了一条竖线。

3.4.4　使用导出输入方向的方法

在许多情况下，我们有大量的输入，它们常常是很相关的。本节的方法产生原始输入 X_j 的少量线性组合 Z_m，$m = 1, \cdots, M$，然后在回归中用 Z_m 替代 X_j 作为输入。这类方法随线性组合的构造方法不同而异。

主成分回归

在该方法中，所用的线性组合 Z_m 是主成分，如前面第 3.4.3 节定义的。

主成分回归对某 $M \leqslant p$，形成导出的输入列 $\mathbf{z}_m = \mathbf{X}v_m$，然后在 $\mathbf{z}_1, \mathbf{z}_2, \cdots, \mathbf{z}_M$ 上对 $\mathbf{y}$ 回归。由于这些 $\mathbf{z}_m$ 是正交的，所以该回归就是一元回归的和：

$$\hat{\mathbf{y}}^{\text{pcr}} = \bar{y} + \sum_{m=1}^{M} \hat{\theta}_m \mathbf{z}_m \tag{3.52}$$

其中，$\hat{\theta}_m = \langle \mathbf{z}_m, \mathbf{y} \rangle / \langle \mathbf{z}_m, \mathbf{z}_m \rangle$。由于每个 $\mathbf{z}_m$ 都是原来的 $\mathbf{x}_j$ 的线性组合，可以用 $\mathbf{x}_j$ 的系数表示解(3.52)(见习题 3.12)：

$$\hat{\beta}^{\text{pcr}}(M) = \sum_{m=1}^{M} \hat{\theta}_m v_m \tag{3.53}$$

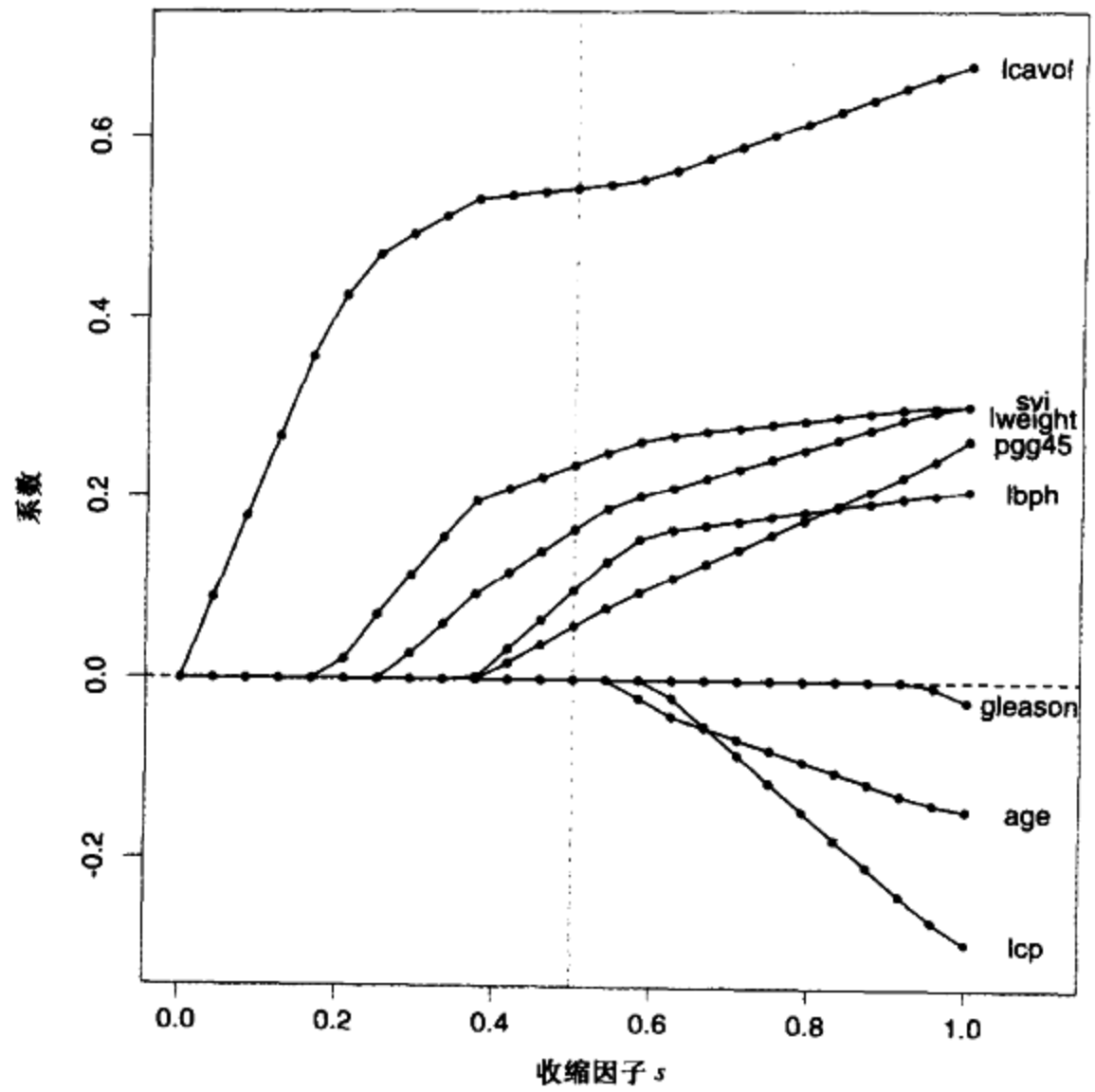


图 3.9 套索系数分布图，系数随调整参数 t 变化。系数作为 $s = t / \sum_1^p |\hat{\beta}_j|$ 的函数绘制。一条竖线绘在 $s = 0.5$ 处，这是交叉验证选取的值。与图3.7相比，套索的曲线到达0，而岭回归的曲线不到0

与岭回归一样，主成分依赖于输入的定标。因此，通常首先要对输入标准化。注意，如果 $M = p$，我们回到通常的最小二乘方估计，因为 $\mathbf{Z} = \mathbf{UD}$ 的列生成 $\mathbf{X}$ 的列空间。对于 $M < p$，得到约化的回归。我们看到主成分回归与岭回归非常类似，二者都凭借输入矩阵的主成分操作。岭回归收缩主成分的系数(见图 3.10)，收缩量更依赖于对应的本征值的大小；而主成分回归丢弃 $p - M$ 个最小本征值成分，如图 3.10 所示。

在图 3.6 中，我们看到交叉验证建议 7 项作为主成分；结果模型与表 3.3 中岭回归具有大致相同的检验误差。

部分最小二乘方

该技术也构造输入的线性组合的集合，用于回归。但是，与主成分回归不同，(除 $\mathbf{X}$ 之外)它还在构造中使用 $\mathbf{y}$。假定 $\mathbf{y}$ 是中心化的，并且每个 $\mathbf{x}_j$ 是标准化的，具有均值 0 和方差 1。PLS(部分最小二乘方)从计算 $\mathbf{y}$ 在每个 $\mathbf{x}_j$ 上的一元回归系数 $\hat{\varphi}_{1j}$(即 $\hat{\varphi}_{1j} = \langle \mathbf{x}_j, \mathbf{y} \rangle$)开始。由此，我们构造导出的输入 $\mathbf{z}_1 = \sum \hat{\varphi}_{1j} \mathbf{x}_j$，它是第一个部分最小二乘方方向。因此，在每个 $\mathbf{z}_m$ 的构造中，输入按它们在 $\mathbf{y}$ 上的单变量效应加权。结果 $\mathbf{y}$ 在 $\mathbf{z}_1$ 上回归，产生系数，然后将 $\mathbf{x}_1, \cdots, \mathbf{x}_p$ 关于 $\mathbf{z}_1$ 正交化。继续该过程，直至得到 $M \leqslant p$ 个方向。按照这种方式，部分最小二乘方产生一

个导出输入或方向序列 $\mathbf{z}_1, \mathbf{z}_2, \cdots, \mathbf{z}_M$。与主成分回归一样，如果构造所有 $M = p$ 个方向，我们又得到一个等价于通常的最小二乘方估计的解；使用 $M < p$ 个方向产生约化的回归。该过程的完整描述在算法 3.2 中。

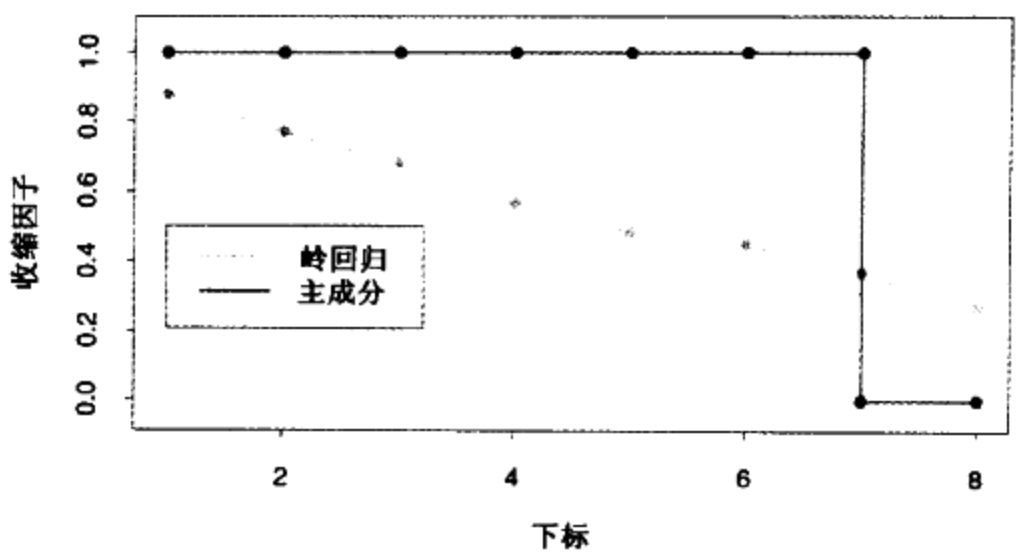


图 3.10　岭回归(浅色)使用式(3.47)中的收缩因子 $d_j^2/(d_j^2+\lambda)$，收缩主成分的系数。主成分回归(深色)截掉它们。图中显示的是对应于图3.6的收缩和截断模式，作为主成分下标的函数

算法 3.2　部分最小二乘方

1. 标准化每个 $\mathbf{x}_j$，使得它具有均值 0，方差 1。置 $\hat{\mathbf{y}}^{(0)} = 1\bar{y}, \mathbf{x}_j^{(0)} = \mathbf{x}_j, j = 1, \cdots, p$

2. 对于 $m = 1, 2, \cdots, p$

   - $\mathbf{z}_m = \sum_{j=1}^{p} \hat{\varphi}_{mj}\mathbf{x}_j^{(m-1)}$，其中 $\hat{\varphi}_{mj} = \langle \mathbf{x}_j^{(m-1)}, \mathbf{y}\rangle$
   - $\hat{\theta}_m = \langle \mathbf{z}_m, \mathbf{y}\rangle / \langle \mathbf{z}_m, \mathbf{z}_m\rangle$
   - $\hat{\mathbf{y}}^{(m)} = \hat{\mathbf{y}}^{(m-1)} + \hat{\theta}_m \mathbf{z}_m$
   - 对每个 $\mathbf{x}_j^{(m-1)}$ 关于 $\mathbf{z}_m$ 正交化：$\mathbf{x}_j^{(m)} = \mathbf{x}_j^{(m-1)} - [\langle \mathbf{z}_m, \mathbf{x}_j^{(m-1)}\rangle / \langle \mathbf{z}_m, \mathbf{z}_m\rangle]\mathbf{z}_m, j = 1, 2, \cdots, p$

3. 输出拟合向量序列 $\{\hat{\mathbf{y}}^{(m)}\}_1^p$。由于 $\{\mathbf{z}_\ell\}_1^m$ 在原来的 $\mathbf{x}_j$ 上是线性的，因此 $\hat{\mathbf{y}}^{(m)} = \mathbf{X}\hat{\beta}^{\text{pls}}(m)$ 也是线性的。这些线性系数可以通过 PLS 变换序列重新获得

在前列腺癌例子中，交叉验证选取图 3.6 中的 $M = 2$ 个 PLS 方向。这产生表 3.3 最右列给出的模型。

部分最小二乘方解决的优化问题是什么？由于它使用响应 $\mathbf{y}$ 构造它的方向，它的解是 $\mathbf{y}$ 的非线性函数。可以证明，与主成分回归不同，部分最小二乘方寻找具有高方差并且与响应高度相关的方向(Stone 和 Brooks，1990；Frank 和 Friedman，1993)。特殊地，第 m 个主成分方向 v_m 解决：

$$\max_{\substack{\alpha=1 \\ v_\ell^T \mathbf{S}\alpha = 0, \ell = 1, \cdots, m-1}} \text{Var}(\mathbf{X}\alpha) \tag{3.54}$$

其中，$\mathbf{S}$ 是样本 $\mathbf{x}_j$ 的协方差矩阵。条件 $v_\ell^T \mathbf{S}\alpha = 0$ 确保 $\mathbf{z}_m = \mathbf{X}\alpha$ 与前面的所有线性组合 $\mathbf{z}_\ell = \mathbf{X}v_\ell$ 不相关。第 m 个 PLS 方向 $\hat{\varphi}_m$ 解决：

$$\max_{\substack{|\alpha|=1 \\ \hat{\varphi}_l^T \mathbf{S}\alpha=0, l=1,\cdots,m-1}} \mathrm{Corr}^2(\mathbf{y}, \mathbf{X}\alpha)\mathrm{Var}(\mathbf{X}\alpha) \tag{3.55}$$

进一步分析揭示，方差趋向于占支配地位，因而部分最小二乘方的行为非常像岭回归和主成分回归。我们将在下一节进一步讨论该问题。

如果输入矩阵 $\mathbf{X}$ 是正交的，则部分最小二乘方在 $m=1$ 步后找到最小二乘方估计。此后的步骤不起作用，因为对于 $m>1$，$\hat{\varphi}_{mj}$ 为 0（见习题 3.13）。还可以证明：对于 $m=1, 2, \cdots, p$，PLS 系数序列代表计算最小二乘方解的共轭梯度序列（见习题 3.16）。

3.4.5 讨论：选择和收缩方法比较

有一些简单的设置，使我们能够更好地理解上面讨论的方法之间的联系。考虑一个例子，它具有两个相关输入 X_1 和 X_2，相关度为 ρ。假定实际的回归系数是 $\beta_1=4$ 和 $\beta_2=2$。图 3.11 显示不同方法的系数随调整参数变化的曲线。上图有 $\rho=0.5$，下图有 $\rho=-0.5$。岭回归和套索回归的调整参数在一个连续区域变化，而最佳子集、PLS 和 PCR 只取两个离散步骤到最小二乘方解。在上图中，从原点开始，岭回归收缩系数，直到最终收敛于最小二乘方解。PLS 和 PCR 表现出与岭回归类似的行为，尽管它们是离散的，并且更极端。最佳子集走过了头，然后回溯。相对于其他方法，套索的行为则介于它们之间。对于负相关（下图），PLS 和 PCR 还是粗略地追踪岭回归的路径，而所有的方法相互之间更相似。

从贝叶斯理论角度，我们可以进一步洞察这些方法。假定采用高斯先验分布：

$$\beta \sim N(0, \tau I) \tag{3.56}$$

我们看到岭回归估计 $\hat{\beta}^{\mathrm{ridge}}$ 是后验众数（和均值）。这揭示了一个有趣的事情：先验分布（3.56）只是 β 长度的函数，而不是其方向的函数。因此，岭回归的低方差方向收缩不是基于偏向高方差的先验分布；该收缩带来的方差降低是由于输入矩阵 $\mathbf{X}$ 中的相关性。

回想一下，岭回归收缩所有的方向，但是低方差方向收缩更多。主成分回归留下 M 个高方差方向，而丢弃其他的。因此，其蕴涵的先验将较高的概率加在 M 个高方差方向，而将 0 概率加在 $p-M$ 个低方差方向。有趣的是，可以证明：部分最小二乘方也趋向于收缩低方差方向，但实际上可以膨胀某些较高方差的方向。这可能使 PLS 有些不稳定，并导致它具有略高于岭回归的预测误差。详尽的研究在 Frank 和 Friedman（1993）的论文中。这些作者断言：对于极小化预测误差，岭回归通常比变量子集选择、主成分回归和部分最小二乘方更可取。然而，在后两种方法上的改进是微不足道的。

综上所述，PLS、PCR 和岭回归的行为类似。岭回归可能更可取，因为它是平滑地收缩，而不是离散地收缩。

现在，我们关注岭回归、套索和子集回归。当输入矩阵 $\mathbf{X}$ 正交时，三个过程都有显式解。每种方法都对最小二乘方估计 $\hat{\beta}_j$ 施加一个简单的变换，如表 3.4 所示。岭回归按比例收缩。最佳子集保留 M 个最大的系数，而套索通过一个常数因子变换每个系数，在 0 上截断。这称为“取软阈值”，并用在第 5.9 节的基于小波的光滑方法中。注意，套索公式中的阈值参数 γ 是定义式（3.51）中界限 t 的一个 1-1 变换。

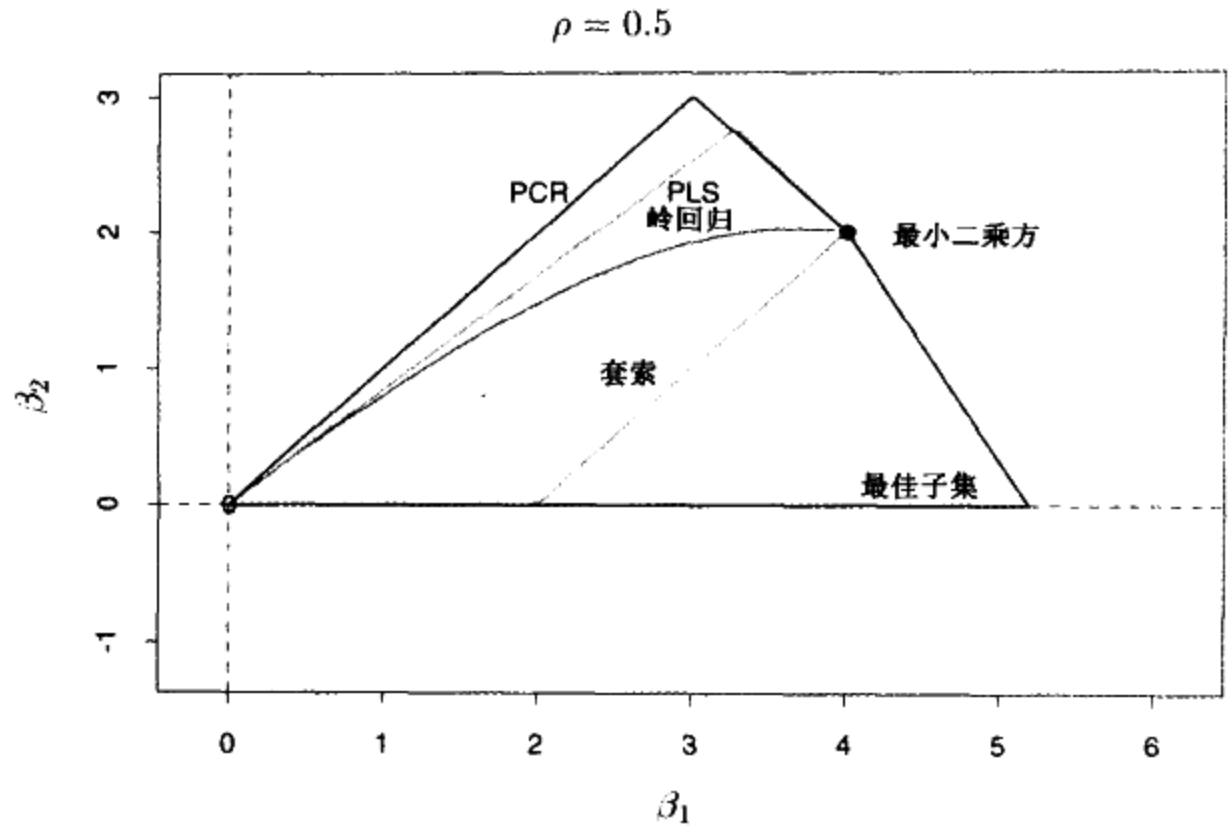


图 3.11　对于一个简单例子，不同方法的系数曲线图：两个具有相关度 ±0.5的输入，而实际的回归系数是 $\beta = (4,2)$（见彩页）

表 3.4　**X 的列正交情况下 β_j 的估计。λ，M 和 γ 是相应技术选取的常数。sign 是符号函数，产生其参数的符号（±1），而 x_+ 表示 x 的“正的部分”**

估计法	公式
最佳子集（容量 M）	$\hat{\beta}_j$，如果 $\text{rank}(\lvert\hat{\beta}_j\rvert) \leqslant M$
岭回归	$\hat{\beta}_j/(1+\lambda)$
套索回归	$\text{sign}(\hat{\beta}_j)(\lvert\hat{\beta}_j\rvert - \gamma)_+$

回到非正交的情况。有些图可以帮助我们理解它们的联系。图 3.12 描述只有两个参数的套索(左)和岭回归(右)。残差的平方和具有椭圆围线,中心在完全最小二乘方估计。岭回归的约束域是圆 $\beta_1^2+\beta_2^2 \leqslant t^2$,而套索回归的约束域是菱形 $|\beta_1|+|\beta_2| \leqslant t$。两种方法都找出第一个点,它是椭圆围线与约束区域的接触点。与圆不同的是,菱形有尖角;如果解在尖角上,则有一个参数 β_j 等于零。当 $p>2$ 时,菱形变成了菱形体,有许多角、直线边和面,估计参数为 0 的可能性更大。

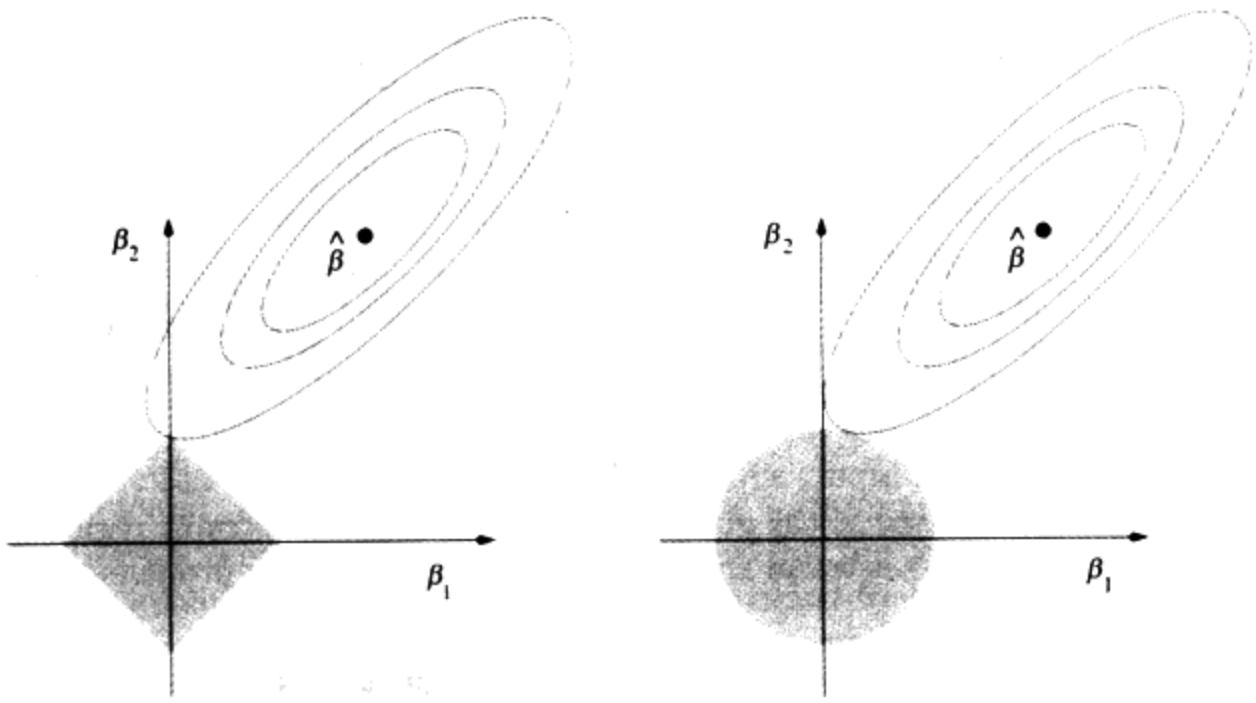


图 3.12 套索(左)和岭回归(右)的估计图。显示的是误差和约束函数的围线。实心区域分别是约束区域 $|\beta_1|+|\beta_2| \leqslant t$ 和 $\beta_1^2+\beta_2^2 \leqslant t^2$,而椭圆是最小二乘方误差函数的围线

我们可以拓广岭回归和套索回归,并将它们看成贝叶斯估计。对于 $q \geqslant 0$,考虑准则:

$$\tilde{\beta}=\underset{\beta}{\operatorname{argmin}}\left\{\sum_{i=1}^{N}\left(y_i-\beta_0-\sum_{j=1}^{p}x_{ij}\beta_j\right)^2+\lambda\sum_{j=1}^{p}|\beta_j|^q\right\} \tag{3.57}$$

对于两个输入的情况,$\sum_j |\beta_j|^q$ 的常数值围线如图 3.13 所示。

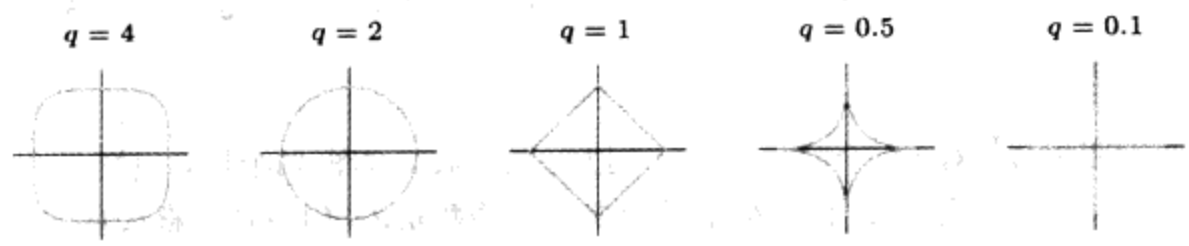


图 3.13 给定 q,常数值 $\sum_j |\beta_j|^q$ 的围线

想像 $|\beta_j|^q$ 是 β_j 的对数先验密度,这些也是参数先验分布的等值线。值 $q=0$ 对应于变量子集选择,罚简单地是非零参数的个数;$q=1$ 对应于套索,而 $q=2$ 对应于岭回归。注意,对于 $q \leqslant 1$,先验分布在方向上不是均匀的,而是在坐标方向上积聚了更多质量。对应于 $q=1$ 的先验分布对每个输入是一个独立的二重指数(或拉普拉斯)分布,密度为 $(1/2\tau)\exp(-|\beta|)/\tau$ 并且 $\tau=1/\lambda$。$q=1$(套索)是使得约束区域为凸区域的最小 q,非凸的约束区域使得优化问题更加困难。

在这种观点下，套索、岭回归和最佳子集选择都是具有不同先验分布的贝叶斯估计。然而需要注意，它们是作为后验众数（即后验的最大者）导出的。正如贝叶斯估计，后验均值更常用。岭回归也是后验均值，但套索和最佳子集不是。

重新考虑准则(3.57)，我们可能试图使用除 0，1 或 2 之外的其他 q 值。的确，我们甚至可以由数据估计 q。据我们所知，对该问题尚未研究。

3.4.6　多元输出收缩和选择

正如第 3.3.1 节提到的，多元输出线性模型的最小二乘方估计是简单地对每个输出分别的最小二乘方估计。

为了将选择和收缩方法应用于多元输出情况，可以将一元技术分别用于每个输出，或同时用于所有输出。例如，对于岭回归，我们可以使用不同的参数值 λ，将式(3.44) 应用于结果矩阵 Y 的每一列；或者使用相同的参数值 λ，将该公式应用于所有列。前一种策略允许对不同的输出使用不同的正则化量，但需要 k 个单独的正则化参数 $\lambda_1, \cdots, \lambda_k$ 的估计；而后者允许使用所有 k 个输出估计惟一的正则化参数 λ。

其他更复杂的收缩和选择策略利用不同响应的相关性，对于多元输出情况可能是有帮助的。例如，假定在输出之间，我们有：

$$Y_k = f(X) + \varepsilon_k \tag{3.58}$$

$$Y_\ell = f(X) + \varepsilon_\ell \tag{3.59}$$

即，式(3.58)和式(3.59)在它们的模型中具有共同的组成部分 $f(X)$。显然，在此情况下，我们应当汇聚 Y_k 和 Y_l 上的观测来估计共同的 f。

组合响应是标准相关分析（canonical correlation analysis，CCA）的核心。CCA 是为多元输出开发的一种数据归约技术。类似于 PCA，CCA 找出 $\mathbf{x}_j$ 的不相关线性组合序列 $\mathbf{X}v_m$（$m = 1, \cdots, M$）和响应 $\mathbf{y}_k$ 的不相关线性组合的对应序列 $\mathbf{Y}u_m$，使得相关度

$$\mathrm{Corr}^2(\mathbf{Y}u_m, \mathbf{X}v_m) \tag{3.60}$$

相继最大化。注意，最多找出 $M = \min(K, p)$ 个方向。前面的标准响应变量是那样一些线性组合（导出的响应），它们最好地被 $\mathbf{x}_j$ 预测。相反，尾随的标准变量不能很好地被 $\mathbf{x}_j$ 预测，从而是被丢弃的候选。CCA 的解使用样本交叉协方差矩阵 $\mathbf{Y}^T\mathbf{X}/N$ 的广义 SVD 来计算（假定 $\mathbf{Y}$ 和 $\mathbf{X}$ 已经中心化，见习题 3.18）。

降秩回归（reduced-rank regression）（Izenman，1975；van der Merwe 和 Zidek，1980）用显式合并信息的回归模型将该方法形式化。给定误差协方差 $\mathrm{Cov}(\varepsilon) = \mathbf{\Sigma}$，解下面受限的多元回归问题：

$$\hat{\mathbf{B}}^{\mathrm{rr}}(m) = \underset{\mathrm{rank}(\mathbf{B})=m}{\mathrm{argmin}} \sum_{i=1}^{N} (y_i - \mathbf{B}^T x_i)^T \mathbf{\Sigma}^{-1} (y_i - \mathbf{B}^T x_i) \tag{3.61}$$

用估计 $\mathbf{Y}^T\mathbf{Y}/N$ 替换 $\mathbf{\Sigma}$，可以证明（见习题 3.19）解由 $\mathbf{Y}$ 和 $\mathbf{X}$ 的 CCA 给出：

$$\hat{\mathbf{B}}^{\mathrm{rr}}(m) = \hat{\mathbf{B}}\mathbf{U}_m\mathbf{U}_m^- \tag{3.62}$$

其中，$\mathbf{U}_m$ 是 $\mathbf{U}$ 的 $K \times m$ 子矩阵，由 $\mathbf{U}$ 的前 m 列组成；而 $\mathbf{U}$ 是左标准向量 $u_1, u_2, \cdots, u_M$ 的 $K \times$

M 矩阵。$\mathbf{U}_m^-$ 是它的广义逆。将该解写成：

$$\hat{\mathbf{B}}^{\mathrm{rr}}(M) = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T(\mathbf{Y}\mathbf{U}_m)\mathbf{U}_m^- \tag{3.63}$$

我们看到，降秩回归在合并的响应矩阵 $\mathbf{Y}\mathbf{U}_m$ 上进行线性回归，然后将系数（因此拟合也一样）映射回原来的响应空间。降秩拟合由下式给出：

$$\begin{aligned}\hat{\mathbf{Y}}^{\mathrm{rr}}(m) &= \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}\mathbf{U}_m\mathbf{U}_m^- \\ &= \mathbf{H}\mathbf{Y}\mathbf{P}_m\end{aligned} \tag{3.64}$$

其中，$\mathbf{H}$ 是通常的线性回归投影算子，$\mathbf{P}_m$ 是 m 秩 CCA 响应投影算子。尽管 $\mathbf{\Sigma}$ 的更好估计是 $(\mathbf{Y}-\mathbf{X}\hat{\mathbf{B}})^T(\mathbf{Y}-\mathbf{X}\hat{\mathbf{B}})/(N-pK)$，我们可以证明它的解相同（见习题 3.20）。

降秩回归通过截断 CCA 借助于响应而加强。Breiman 和 Friedman（1997）利用 $\mathbf{X}$ 和 $\mathbf{Y}$ 之间标准变量的成功收缩，开发了降秩回归的一个光滑版本。他们的提议具有如下形式[与式（3.62）比较]：

$$\hat{\mathbf{B}}^{\mathrm{c+w}} = \hat{\mathbf{B}}\mathbf{U}\mathbf{\Lambda}\mathbf{U}^{-1} \tag{3.65}$$

其中，$\mathbf{\Lambda}$ 是对角收缩矩阵（“c + w”表示“乳块”和“乳清”，是为他们的过程起的名字）。基于在总体上的最优预测，他们证明 $\mathbf{\Lambda}$ 的对角线元素为：

$$\lambda_m = \frac{c_m^2}{c_m^2 + \frac{p}{N}(1-c_m^2)},\ m = 1,\cdots,M \tag{3.66}$$

其中，c_m 是第 m 个标准相关系数。注意，随着输入变量数目和样本容量之比 p/N 减小，收缩因子趋向于 1。Breiman 和 Friedman（1997）基于训练数据和交叉验证提出了一个修改版本，但主要形式相同。这里，拟合响应具有如下形式：

$$\hat{\mathbf{Y}}^{\mathrm{c+w}} = \mathbf{H}\mathbf{Y}\mathbf{S}^{\mathrm{c+w}} \tag{3.67}$$

其中，$\mathbf{S}^{\mathrm{c+w}} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^{-1}$ 是响应收缩算子。

Breiman 和 Friedman（1997）还建议在 Y 空间和 X 空间都进行收缩。这导致如下形式的混合收缩模型：

$$\hat{\mathbf{Y}}^{\mathrm{ridge,c+w}} = \mathbf{A}_\lambda\mathbf{Y}\mathbf{S}^{\mathrm{c+w}} \tag{3.68}$$

其中，$\mathbf{A}_\lambda = \mathbf{X}(\mathbf{X}^T\mathbf{X}+\lambda\mathbf{I})^{-1}\mathbf{X}^T$ 是岭回归收缩算子，如式（3.46）所示。他们的论文及其讨论包含更多细节。

3.5 计算考虑

最小二乘方拟合通常通过矩阵 $\mathbf{X}^T\mathbf{X}$ 的 Cholesky 分解或 $\mathbf{X}$ 的 QR 分解来做。给定 N 个观测和 p 个特征，Cholesky 分解需要 $p^3 + Np^2/2$ 操作，而 QR 分解需要 Np^2 操作。依赖于 N 和 p 的相对大小，有时 Cholesky 分解较快；另一方面，它可能稳定性较差（Lawson 和 Hansen，1974）。套索计算需要二次规划；例子见 Murray 等人（1981）的论文。

文献注释

线性回归在许多统计学书籍中讨论，如 Seber（1984），Weisberg（1980）和 Mardia 等人（1979）

的著作。岭回归由 Hoerl 和 Kennard(1970)引进,而套索由 Tibshirani(1996)提出的著作。部分最小二乘方由 Wold(1975)引进。收缩方法的比较可以在 Copas(1983),Frank 和 Friedman(1993)中找到。

习题

3.1 证明从模型中删除单个系数的 F 统计量(3.13)等于对应的 z- 得分(3.12)的平方。

3.2 给定两个变量 X 和 Y 上的数据,考虑拟合一个三次多项式回归模型 $f(X) = \sum_{j=0}^{3}\beta_j X^j$。除绘制拟合曲线外,还想绘出曲线的 95% 置信带宽。考虑下面两种方法:

1. 在每个点 x_0,对于线性函数 $a^T\beta = \sum_{j=0}^{3}\beta_j x_0^j$,形成 95% 置信区间。
2. 形成式(3.15)中 β 的 95% 置信集,由此产生 $f(x_0)$ 的置信区间。

这些方法有何不同? 哪种带宽可能宽一些? 做一个小型模拟实验,比较两种方法。

3.3 (a) 证明高斯 - 马尔可夫定理:参数 $a^T\beta$ 的 $\tilde{V}$ 最小二乘方估计的方差不大于参数 $a^T\beta$ 的任何线性无偏估计(见第 3.2.2 节)。

(b) 矩阵不等式 $\mathbf{B} \leq \mathbf{A}$ 成立,如果 $\mathbf{A} - \mathbf{B}$ 是半正定的。证明:如果 $\hat{\mathbf{V}}$ 是 β 的 $\tilde{\mathbf{V}}$ 最小二乘方估计的方差 - 协方差矩阵,$\tilde{\mathbf{V}}$ 是其他无偏估计的方差 - 协方差矩阵,则 $\hat{\mathbf{V}} \leq \tilde{\mathbf{V}}$。

3.4 表明如何由 Gram-Schmidt 过程(见算法 3.1)的一趟得到最小二乘方系数向量。用 $\mathbf{X}$ 的 QR 分解提供你的解。

3.5 考虑岭回归问题(3.41)。证明该问题等价于问题:

$$\hat{\beta}^c = \underset{\beta^c}{\operatorname{argmin}}\left\{\sum_{i=1}^{N}\left[y_i - \beta_0^c - \sum_{j=1}^{p}(x_{ij} - \bar{x}_j)\beta_j^c\right]^2 + \lambda\sum_{j=1}^{p}\beta_j^{c2}\right\} \tag{3.69}$$

给出 β^c 与式(3.41)中原来 β 之间的对应。说明解关于修改后准则的特点。

3.6 证明在高斯先验分布 $\beta \sim N(0,\tau\mathbf{I})$ 和高斯选样模型 $\mathbf{y} \sim N(\mathbf{X}\beta,\sigma^2\mathbf{I})$ 下,岭回归估计是后验分布的均值(和众数)。找出岭回归公式中正则化参数 λ 与方差 τ 和 σ^2 之间的联系。

3.7 假定 $y_i \sim N(\beta_0 + x_i^T\beta,\ \sigma^2)$,$i = 1,2,\cdots,N$,并且每个参数 β_j 都服从分布 $N(0,\tau^2)$,相互独立。假定 σ^2 和 τ^2 已知,证明 β 的(负)对数后验密度正比于 $\sum_{i=1}^{N}(y_i - \beta_0 - \sum_j x_{ij}\beta_j)^2 + \lambda\sum_{j=1}^{p}\beta_j^2$,其中 $\lambda = \sigma^2/\tau^2$。

3.8 考虑非中心化的 $N \times (p+1)$ 矩阵 $\mathbf{X}$ 的 QR 分解和中心化的 $N \times p$ 矩阵 $\tilde{\mathbf{X}}$ 的 SVD。证明 $\mathbf{Q}_2$ 和 $\mathbf{U}$ 生成相同的子空间;其中,$\mathbf{Q}_2$ 是 $\mathbf{Q}$ 的子矩阵,去掉 $\mathbf{Q}$ 的第一列。在什么情况下它们相同,取决与符号交换吗?

3.9 证明多元线性回归问题(3.39)的解由式(3.38)给出。如果每个观测的协方差矩阵 $\mathbf{\Sigma}_i$ 不同,会怎么样?

3.10 证明岭回归的估计可以通过在增广数据集上的一般最小二乘方回归得到。我们用 p 个附加的行 $\sqrt{\lambda}\mathbf{I}$ 增广中心化的矩阵 $\mathbf{X}$,并用 p 个 0 增广 $\mathbf{y}$。通过引进响应值为 0 的人工数据,拟合过程强制将这些系数收缩为 0。这涉及 Abu-Mostafa(1995)的提示思想,其中模型的约束通过添加满足它们的人工数据实例实现。

3.11 考虑套索问题(3.51)。证明它等价于问题：

$$\hat{\beta}^c = \underset{\beta^c}{\operatorname{argmin}}\left\{\sum_{i=1}^{N}\left[y_i - \beta_0^c - \sum_{j=1}^{p}(x_{ij} - \bar{x}_j)\beta_j^c\right]^2 + \lambda\sum_{j=1}^{p}|\beta_j^c|\right\} \tag{3.70}$$

给出 β^c 与式(3.51)中原来的 β 之间的对应。对修改后的标准讨论解的特性。

3.12 推导表达式(3.53)，并证明 $\hat{\beta}^{\text{pcr}}(p) = \hat{\beta}^{\text{ls}}$。

3.13 证明：在正交的情况下，PLS 在 $m = 1$ 步后停止，因为在算法 3.2 第 2 步中，后继的 $\hat{\varphi}_{mj}$ 都是 0。

3.14 推导表 3.4 中的表目——正交情况下估计法的显式形式。

3.15 在第 1 章讨论的 spam 数据上，重复表 3.3 的分析。

3.16 阅读共轭梯度算法(例如，Murray 等人的论文，1981)，并建立这些算法与部分最小二乘方之间的联系。

3.17 证明 $\|\hat{\beta}^{\text{ridge}}\|$ 随其调整参数 $\lambda \to 0$ 而递增。对套索和部分最小二乘方估计，同样的性质成立吗？对于后者，考虑“调整参数”为算法的相继步。

3.18 考虑标准协相关问题(3.60)。证明：首对标准变量 u_1 和 v_1 解决如下广义 SVD 问题：

$$\max_{\substack{u^T(\mathbf{Y}^T\mathbf{Y})u=1\\ v^T(\mathbf{X}^T\mathbf{X})v=1}} u^T(\mathbf{Y}^T\mathbf{X})v \tag{3.71}$$

证明解由 $u_1 = (\mathbf{Y}^T\mathbf{Y})^{-\frac{1}{2}}u_1^*$ 和 $v_1 = (\mathbf{X}^T\mathbf{X})^{-\frac{1}{2}}v_1^*$ 给出，其中 u_1^* 和 v_1^* 是下式中的首项左、右奇异向量：

$$(\mathbf{Y}^T\mathbf{Y})^{-\frac{1}{2}}(\mathbf{Y}^T\mathbf{X})(\mathbf{X}^T\mathbf{X})^{-\frac{1}{2}} = \mathbf{U}^*\mathbf{D}^*\mathbf{V}^{*T} \tag{3.72}$$

证明整个序列 $u_m, v_m, m = 1,2,\cdots,\min(K,p)$ 也由式(3.72)给出。

3.19 证明：$\boldsymbol{\Sigma}$ 用 $\mathbf{Y}^T\mathbf{Y}/N$ 估计，降秩回归问题的解由式(3.62)给出。提示：将 $\mathbf{Y}$ 变换成 $\mathbf{Y}^* = \mathbf{Y}\boldsymbol{\Sigma}^{-\frac{1}{2}}$，并使用标准向量 u_m^* 来求解。证明 $\mathbf{U}_m = \boldsymbol{\Sigma}^{-\frac{1}{2}}\mathbf{U}_m^*$，并且广义逆是 $\mathbf{U}_m^- = \mathbf{U}_m^{*T}\boldsymbol{\Sigma}^{\frac{1}{2}}$。

3.20 证明：如果 $\boldsymbol{\Sigma}$ 用更自然的量 $(\mathbf{Y} - \mathbf{X}\hat{\mathbf{B}})^T(\mathbf{Y} - \mathbf{X}\hat{\mathbf{B}})/(N - pK)$ 来估计，习题 3.19 的解并不改变。

第 4 章　分类的线性方法

4.1　引言

本章我们回到分类问题并关注分类的线性方法。由于预测子 $G(x)$在离散集合$\mathcal{G}$上取值，所以总可以根据分类将输入空间分割成标定的区域集合。在第 2 章我们看到，这些区域的边界可以是粗糙的或光滑的，取决于预测函数。对于一类重要过程，这些判定边界是线性的；这就是我们所说的分类的线性方法的含义。

有一些不同的方法用来求线性判定边界。在第 2 章，我们用线性回归模型拟合类指示变量，并分类到最大拟合。假设有 K 个类，为方便起见，记做 $1,2,\cdots,K$；第 k 个指示响应变量的拟合线性模型为 $\hat{f}_k(x)=\hat{\beta}_{k0}+\hat{\beta}_k^T x$。类 k 和类 ℓ 之间的判定边界是满足 $\hat{f}_k(x)=\hat{f}_\ell(x)$的点集合，即集合 $\{x:(\hat{\beta}_{k0}-\hat{\beta}_{\ell 0})+(\hat{\beta}_k-\hat{\beta}_\ell)^T x=0\}$——一个仿射集或超平面[①]。既然对任意两个类都如此，因此输入空间被分成具有分段超平面判定边界的常数分类区域。这种回归方法是下面分类方法的一种：它对每个类建立一个判别函数(discriminant function)$\delta_k(x)$，并将 x 分类到其判别函数具有最大值的类。用后验概率 $\Pr(G=k\mid X=x)$建模的方法也属于这一类。显然，如果 $\delta_k(x)$或 $\Pr(G=k\mid X=x)$在 x 上是线性的，则判定边界也是线性的。

实际上，为了使判定边界为线性的，我们需要 $\delta_k(x)$或 $\Pr(G=k\mid X=x)$的某个单调变换是线性的。例如，如果有两个类，一种流行的后验概率模型是：

$$\begin{aligned}\Pr(G=1|X=x)&=\frac{\exp(\beta_0+\beta^T x)}{1+\exp(\beta_0+\beta^T x)}\\ \Pr(G=2|X=x)&=\frac{1}{1+\exp(\beta_0+\beta^T x)}\end{aligned}\tag{4.1}$$

这里，单调变换是分对数(logit)变换：$\log[p/(1-p)]$，并且事实上我们看到：

$$\log\frac{\Pr(G=1|X=x)}{\Pr(G=2|X=x)}=\beta_0+\beta^T x\tag{4.2}$$

判定边界是对数几率(log-odds)为 0 的点的集合，是由 $\{x\mid\beta_0+\beta^T x=0\}$定义的超平面。我们讨论两种导致线性对数几率或分对数的非常流行但不同的方法：线性判别分析和线性逻辑斯缔回归。尽管在推导方面不相同，但它们的本质差别在于线性函数拟合训练数据的方式。

一种更直接的方法是显式地将类之间的边界建立成线性模型。对于 p 维输入空间的 2-类问题，相当于用超平面对判定边界建模——换言之，用一个法向量和一个割点对判定边界建模。我们将考察两种寻找“分离超平面”的方法。第一种是 Rosenblatt 的著名的感知器(perceptron)模型(1958)和算法，如果存在，该算法找出训练数据中的分离超平面。第二种方法归

① 严格地说，超平面经过原点，而仿射集不必经过原点。有时我们忽略这一区别，一般称做超平面。

功于 Vapnik(1996),如果存在最佳分离超平面(optimally separating hyperplane),找出最佳分离超平面,否则找出一个超平面,它极小化训练数据中重叠的某种度量。这里,我们处理可分的情况,而将不可分情况的讨论推迟到第 12 章。

尽管本章整篇都是讨论线性判定边界,但仍有一些拓广值得考虑。例如,通过包括变量的平方和叉积 X_2^1, X_2^2,…, X_1X_2,…扩充变量集 X_1,…, X_p,从而添加 $p(p+1)/2$ 个附加的变量。把增广空间上的线性函数映射成原空间上的二次函数——从而将线性判定边界映射成二次判定边界。图 4.1 说明了这一思想。数据是相同的:左图使用二维空间的线性判定边界,而右图使用如上所述的增广的 5 维空间的线性判定边界。该方法可以与任何基变换 $h(X)$一起使用,其中 $h:\mathbb{R}^p \mapsto \mathbb{R}^q$, $q > p$, 将在本章稍后讨论。

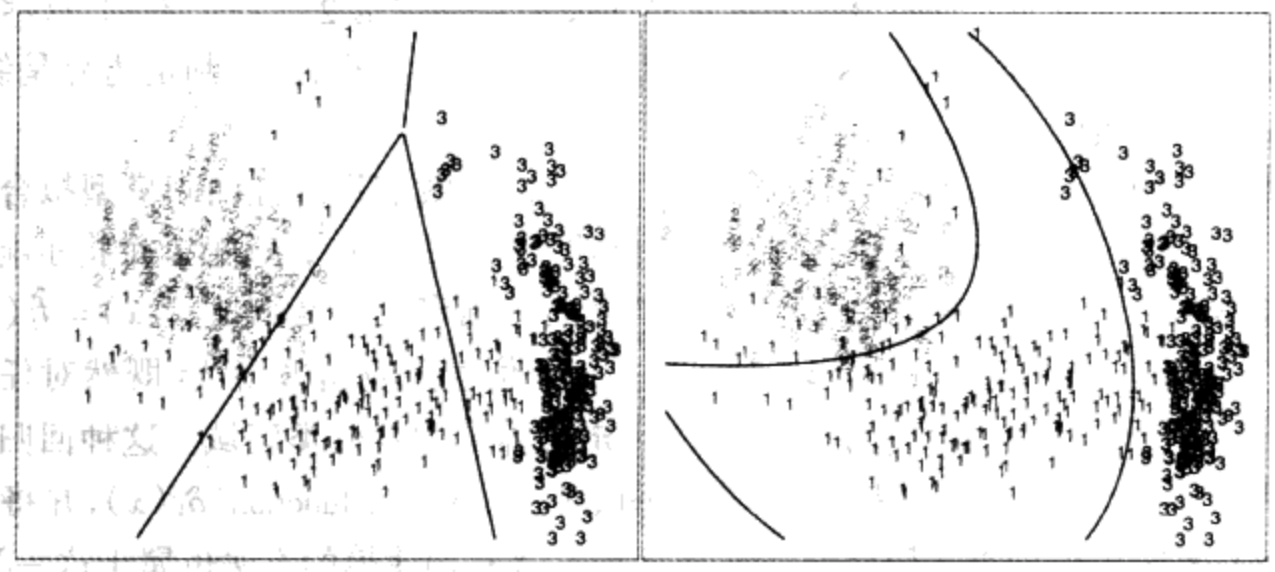


图 4.1 左图显示取自三个类的一些数据点,以及由线性判别分析找出的线性判定边界。右图显示二次判定边界。这些边界通过找出5维空间 X_1, X_2, X_1X_2, X_1^2, X_2^2 中的线性边界得到。在该空间上的线性不等式是原空间中的二次不等式(见彩页)

4.2 指示矩阵的线性回归

下面,每个响应类型通过一个指示变量编码。这样,如果$\mathcal{G}$有 K 个类,则有 K 个这样的指示变量 Y_k, $k = 1,\cdots,K$;其中,如果 $G = k$,则 $Y_k = 1$,否则 $Y_k = 0$。这些指示变量收集在向量 $Y = (Y_1,\cdots,Y_K)$中,并且这些指示变量的 N 个训练实例形成一个 $N \times K$ 的指示响应矩阵(indicator response matrix)$\mathbf{Y}$。$\mathbf{Y}$ 是 0 和 1 的矩阵,每行只有一个 1。我们同时用线性回归模型拟合 $\mathbf{Y}$ 的每一列,并且拟合由下式给出:

$$\hat{\mathbf{Y}} = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y} \tag{4.3}$$

第 3 章给出了线性回归的更多细节。注意,对于每个响应列 $\mathbf{y}_k$,我们有一个系数向量,因此有一个$(p+1) \times K$ 的系数矩阵 $\hat{\mathbf{B}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}$。这里,$\mathbf{X}$ 是模型矩阵,$p+1$ 列对应于 p 个输入,而首列的 1 对应于截距。

一个输入为 x 的新的观测按如下办法分类:

- 计算拟合输出 $\hat{f}(x) = [(1,x)\hat{\mathbf{B}}]^T$,它是一个 K 向量;
- 识别最大分量,并按下式分类:

$$\hat{G}(x) = \text{argmax}_{k \in \mathcal{G}} \hat{f}_k(x) \tag{4.4}$$

该方法的理由是什么？一个相当形式化的理由是将该回归看做条件期望的估计。对于随机变量 Y_k，$E(Y_k \mid X = x) = \Pr(G = k \mid X = x)$，因此，每个 Y_k 的条件期望看来是一个合理的目标。实际问题是：严格的线性回归模型在多大程度上近似于条件期望？换句话说，$\hat{f}_k(x)$是后验概率 $\Pr(G = k \mid X = x)$的合理拟合吗？更重要的是，这有关系吗？

可以直接验证，只要模型中有截距(**X** 中取 1 的列)，对于任意 x，$\sum_{k \in \mathcal{G}} \hat{f}_k(x) = 1$。然而，$\hat{f}_k(x)$可能为负或大于 1，并且有些的确会这样。这是线性回归严格性质的必然结果，特别是当我们在训练数据包之外进行预测时更是如此。这些干扰并不妨碍该方法发挥作用，并且事实上对于许多问题，它产生与更标准的线性分类方法类似的结果。如果允许在输人的基展开 $h(X)$上进行线性回归，该方法将导致相容的概率估计。随着训练数据集容量 N 的增大，我们相应地包含更多的基元素，使得基函数上的线性回归逼近条件期望。我们将在第 5 章讨论这些方法。

一种过于简化的观点是为每个类构造一个目标 t_k，其中 t_k 是 $K \times K$ 单位矩阵的第 k 列。我们的预测问题是试图为观测重新产生合适的目标。使用与前面相同的编码，如果 $g_i = k$，则观测 i 的响应向量y_i(**Y** 的第 i 行)具有值$y_i = t_k$。然后，我们可以用最小二乘方拟合该线性模型：

$$\min_{\mathbf{B}} \sum_{i=1}^{N} \| y_i - [(1, x_i)\mathbf{B}]^T \|^2 \tag{4.5}$$

该标准是拟合向量到它们的目标的欧氏距离平方和。新的观测的分类通过如下方法实现：计算它的拟合向量 $\hat{f}(x)$，并将它分到最近的目标：

$$\hat{G}(x) = \underset{k}{\text{argmin}} \| \hat{f}(x) - t_k \|^2 \tag{4.6}$$

这与前面的方法完全相同：

- 平方和范数准则正是多元响应线性回归准则，只是观察角度稍微不同。由于平方范数本身是平方和，分量解耦(decouple)并可以对每个元素重新安排成分离的线性模型。注意，这是可能的，因为模型中没有什么东西将不同的响应捆绑在一起。
- 容易看出，最近目标分类规则(4.6)与最大拟合分量准则(4.4)恰好相同，但要求拟合值和为 1。

当类的个数 $K \geqslant 3$，特别是当 K 很大时，回归方法还有严重的问题。由于回归模型的严格性，一些类可能被其他类屏蔽。图 4.2 解释了 $K = 3$ 时的一种极端情况。三个类被线性判定边界正确地分隔，但线性回归却完全丢失了中间的类。

在图 4.3 中，我们已经将数据投影到连接三个形心的线上(在此情况下，没有正交方向的信息)，并且包含了三个响应变量 Y_1，Y_2 和 Y_3，并对它们编码。三条回归线(左图)已包括在内，我们看到对应于中间类的线是水平的，并且它的拟合值不占支配地位。这样，类 2 的观测或者分到类 1，或者分到类 3。右图使用二次回归，而不是线性回归。对于这个简单例子，二次而不是线性拟合(至少对于中间类)将解决该问题。然而，可以看到，如果不是三个类，而是四

个类像这样排成一条线，二次回归仍然不行，还需要三次回归。一个不严格的一般规则是：如果 $K \geqslant 3$ 个类排成一条线，则可能需要高达 $K-1$ 次多项式项对它们求解。注意，这些是沿导出方向，经过形心的多项式，它们可以具有任意的方向。因此，为了对最坏情况求解，在 p 维输入空间，我们将需要 $K-1$ 次一般的多项式项和叉积，总共 $O(p^{K-1})$ 项。

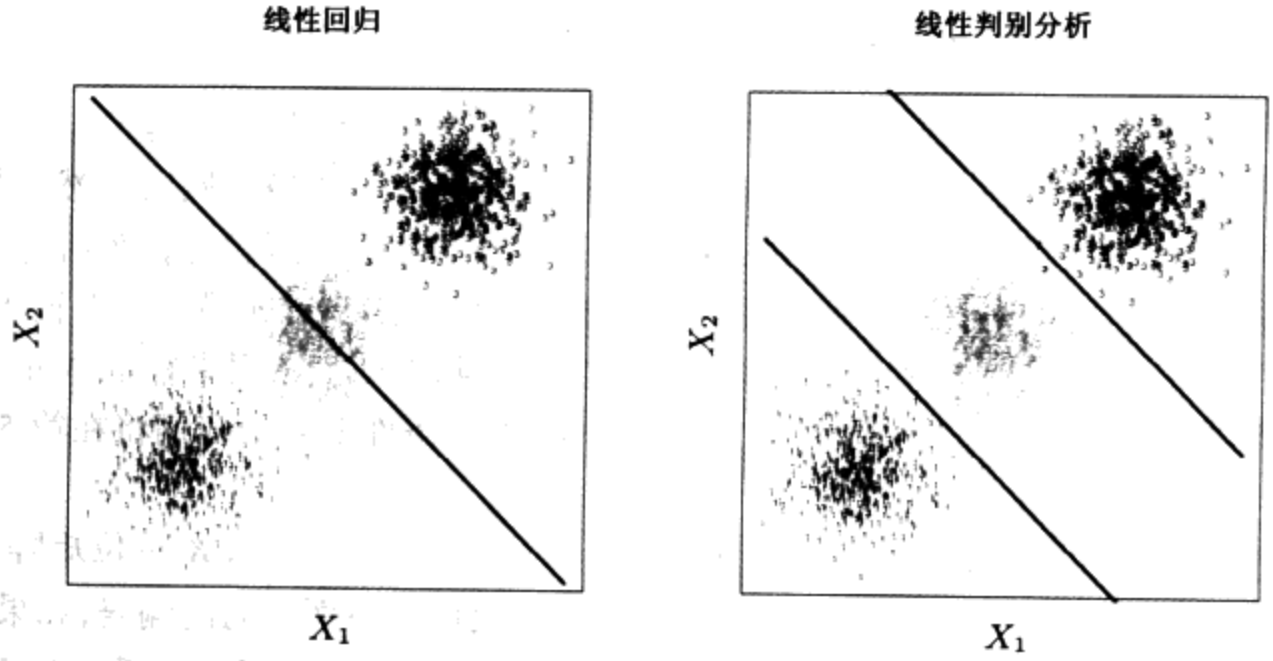


图 4.2 数据取自 $\mathbb{R}^2$ 中的三个类，并容易被线性判定边界分开。右图显示被线性判别分析找到的边界。左图显示被指示响应变量的线性回归找出的边界。中间类完全被屏蔽(不占支配地位)(见彩页)

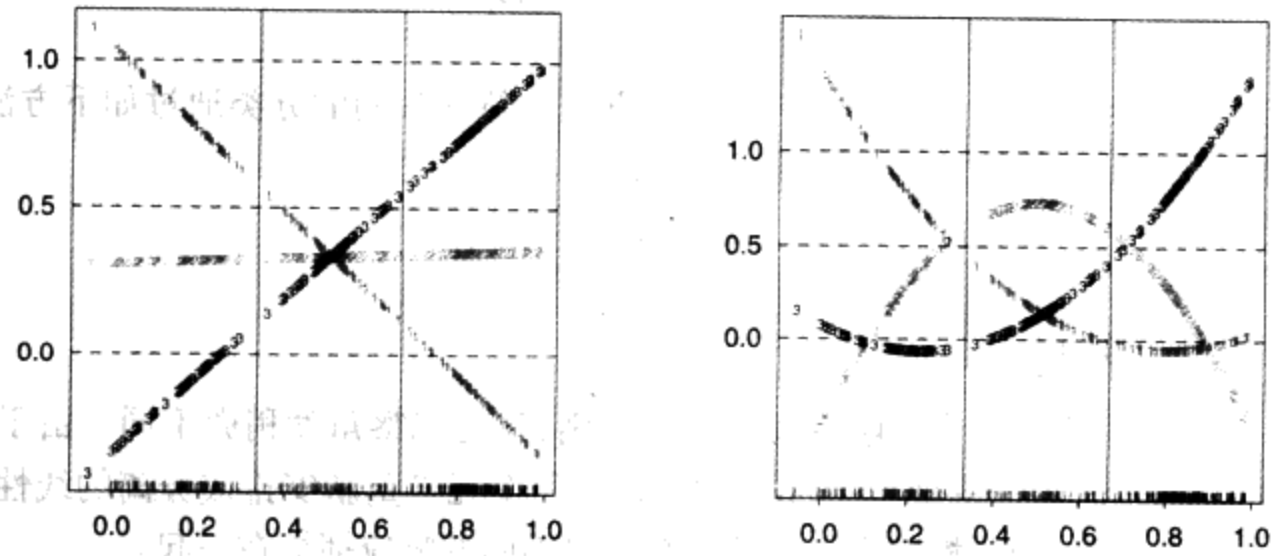


图 4.3 对于一个 3-类问题，IR 上的线性回归的屏蔽作用。底部的底线图(rug plot)指示每个观测的位置和类隶属关系。每个图上的三条曲线是 3-类指示变量的拟合回归；例如，对于红色类，红色观测的 y_{red} 为 1，而绿色和蓝色观测的 y_{red} 为 0。每幅图的上方是训练误差率。对于该问题，贝叶斯误差率为 0.025，与 LDA 误差率一样(见彩页)

这是一个极端例子，但对于较大的 K 和较小的 p，这种屏蔽现象自然会出现。作为一个更实际的图解，图 4.4 是元音识别问题训练数据到信息较多的 2 维子空间上的投影。在 $p=10$ 维空间中有 $K=11$ 个类。这是一个困难的分类问题，并且最好的方法在检验数据上大约有 40% 的误差率。要点汇总在表 4.1；线性回归具有 67% 的误差率，而它的一个近亲(线性判别分析)具有 56% 的误差率。看来屏蔽对此情况具有不良影响。虽然本章的其他方法也都是基于 x 的线性函数，但它们以避免屏蔽问题的方式使用线性函数。

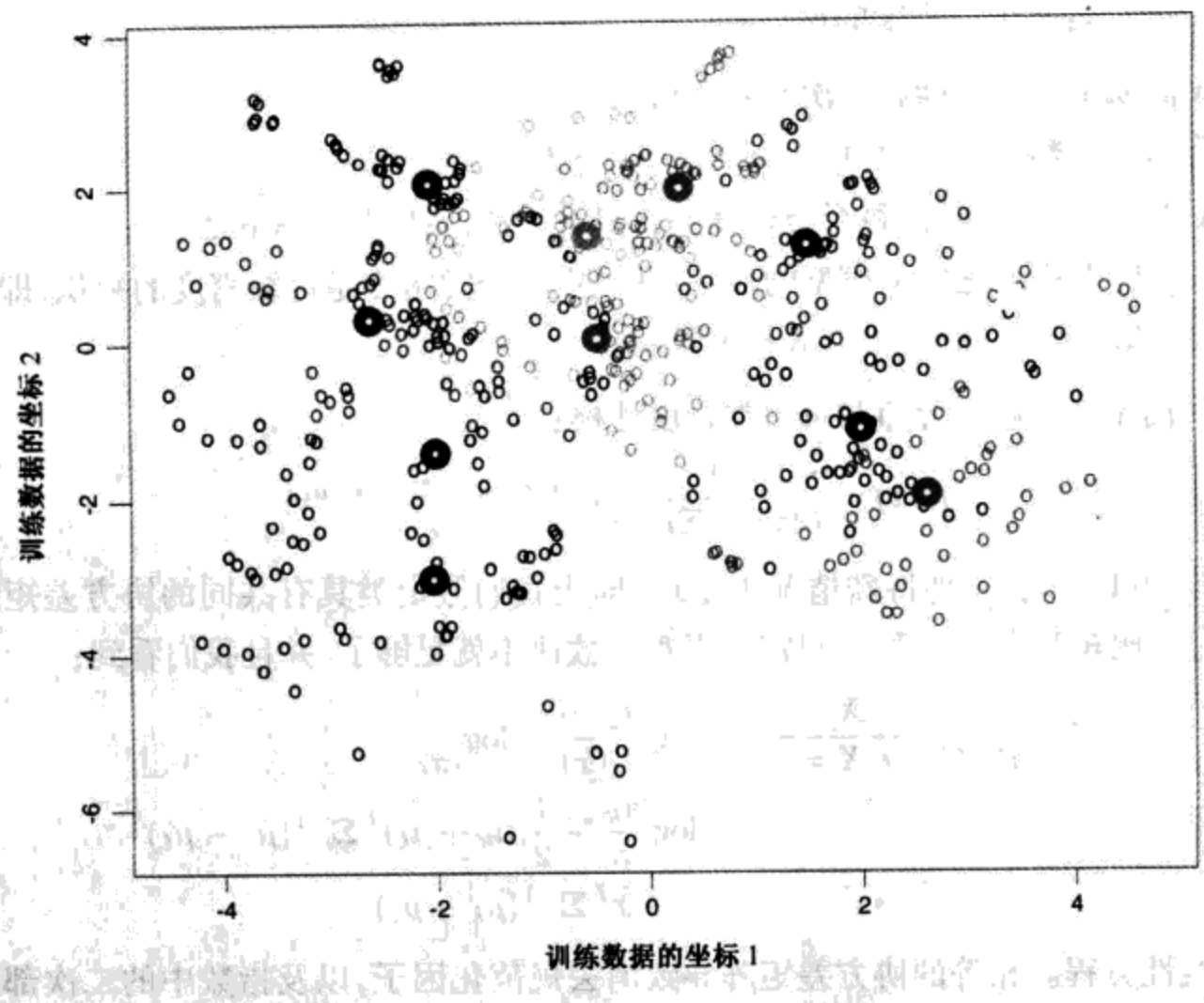


图 4.4 元音训练数据的二维图。有 11 个类，$X \in \mathbb{R}^{10}$。这是 LDA 模型(见第 4.3.3 节)下的最佳视图。加重的圆是每个类的投影均值向量。类的重叠相当多(见彩页)

表 4.1 在元音数据上，使用各种线性技术的训练和检验误差率。在 10 维空间上有11个类，其中3个类占方差的90%(通过光滑分析)。我们看到线性回归受屏蔽的负面影响，训练和检验误差增加了10%以上

技术	误差率	
	训练	检验
线性回归	0.48	0.67
线性判别分析	0.32	0.56
二次判别分析	0.01	0.53
逻辑斯缔回归	0.22	0.51

4.3 线性判别分析

由分类的判定理论(见第 2.4 节)可知，为了最优分类，我们需要知道后验概率 $\Pr(G|X)$。设 $f_k(x)$是类 $G=k$ 中 X 的类条件密度，而 π_k 是类 k 的先验概率，并且 $\sum_{k=1}^{K}\pi_k=1$。贝叶斯定理的简单应用给出：

$$\Pr(G=k|X=x)=\frac{f_k(x)\pi_k}{\sum_{\ell=1}^{K}f_\ell(x)\pi_\ell} \tag{4.7}$$

我们看到，从分类能力讲，有了 $f_k(x)$ 几乎等价于有了 $\Pr(G=k|X=x)$。

许多技术都是基于类密度的模型：

- 使用高斯密度的线性和二次判别分析；
- 允许非线性判定边界的更灵活的高斯混合模型（见第 6.8 节）；
- 一般非参数密度估计，每个类密度允许最大的灵活性（见第 6.6.2 节）；
- 朴素贝叶斯模型是以上模型的变种，并假定每个类密度是边缘密度的乘积，即假定输入在每个类上条件独立（见第 6.6.3 节）。

假定我们用多元高斯分布对每个类密度建模：

$$f_k(x)=\frac{1}{(2\pi)^{p/2}|\mathbf{\Sigma}_k|^{1/2}}e^{-\frac{1}{2}(x-\mu_k)^T\mathbf{\Sigma}_k^{-1}(x-\mu_k)} \tag{4.8}$$

线性判别分析（LDA）在一种特殊情况中出现，即当我们假定类具有共同的协方差矩阵 $\mathbf{\Sigma}_k=\mathbf{\Sigma}\ \forall k$ 时。为了比较两个类 k 和 ℓ，只需要考察对数比率就足够了，并且我们看到：

$$\begin{aligned}\log\frac{\Pr(G=k|X=x)}{\Pr(G=\ell|X=x)}&=\log\frac{f_k(x)}{f_\ell(x)}+\log\frac{\pi_k}{\pi_\ell}\\&=\log\frac{\pi_k}{\pi_\ell}-\frac{1}{2}(\mu_k+\mu_\ell)^T\mathbf{\Sigma}^{-1}(\mu_k-\mu_\ell)\\&\quad+x^T\mathbf{\Sigma}^{-1}(\mu_k-\mu_\ell)\end{aligned} \tag{4.9}$$

是 x 上的线性方程。相等的协方差矩阵导致消去规范化因子，以及指数中的二次部分。该线性对数似然函数意味类 k 和 ℓ 之间的判定边界[集合 $\Pr(G=k|X=x)=\Pr(G=\ell|X=x)$]在 x 上是线性的；在 p 维空间上是一个超平面。当然，这对于任意两个类都成立，因而所有判定边界都是线性的。如果将 $\mathbb{R}^p$ 分割成一些区域，这些区域分成类 1，类 2 等，它们将被超平面分开。图 4.5（左图）展示了一个理想化例子，其中有三个类，而 $p=2$。这里，数据源自于三个具有共同协方差的高斯分布。我们已经在图中绘出对应于 95% 的最高概率密度围线，以及类形心。注意，判定边界不是连接形心线段的中垂线。如果协方差 $\mathbf{\Sigma}$ 是球形 $\sigma^2\mathbf{I}$，并且类先验相等，则判定边界将是连接形心线段的中垂线。

由式（4.9）我们看到：线性判别函数

$$\delta_k(x)=x^T\mathbf{\Sigma}^{-1}\mu_k-\frac{1}{2}\mu_k^T\mathbf{\Sigma}^{-1}\mu_k+\log\pi_k \tag{4.10}$$

是判定规则的等价描述，其中 $G(x)=\operatorname{argmax}_k\delta_k(x)$。

在实践中，我们不知道高斯分布的参数，而需要用训练数据估计它们：

- $\hat{\pi}_k=N_k/N$，其中 N_k 是类 k 的观测数；
- $\hat{\mu}_k=\sum_{g_i=k}x_i/N_k$；
- $\hat{\mathbf{\Sigma}}=\sum_{k=1}^{K}\sum_{g_i=k}(x_i-\hat{\mu}_k)(x_i-\hat{\mu}_k)^T/(N-K)$。

图 4.5（右图）显示基于一个容量为 30 的样本的估计判定边界，该样本取自 3 个高斯分布。图 4.1 是另一例子，但那里类不是高斯分布的。

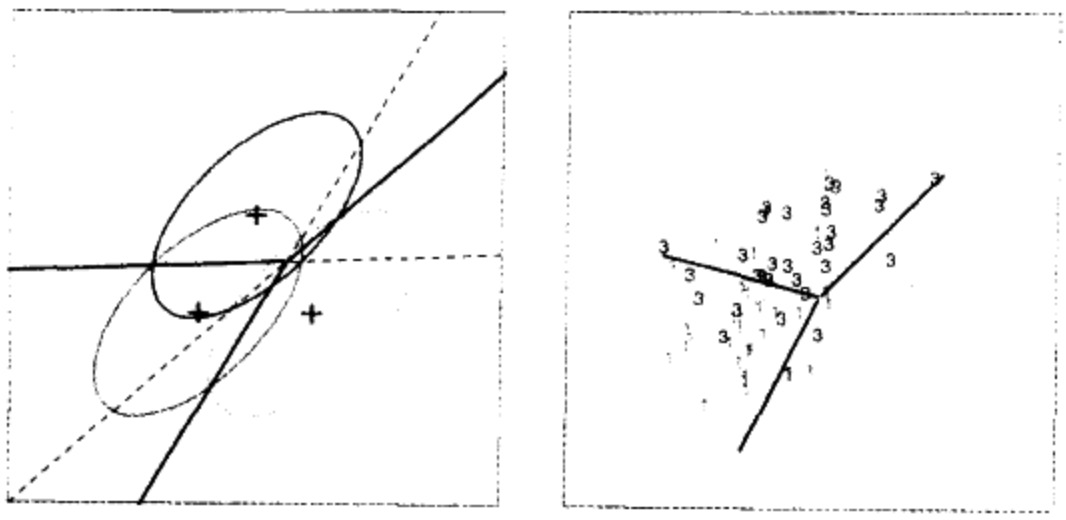


图 4.5　左图显示三个高斯分布，它们具有相同的协方差和不同的均值。图中包含的是每种情况围绕概率95%的常量密度围线。图中显示了每两个类之间的贝叶斯判定边界(虚线)，而分离所有三个类的贝叶斯判定边界是粗实线(前者的子集)。在右图中，我们看到取自每个高斯分布的容量为30的样本，以及拟合的LDA判定边界(见彩页)

对于两个类，线性判别分析与用最小二乘方分类[如式(4.5)]之间存在一个简单的对应。LDA 规则将 x 分到类 2，如果

$$x^T\hat{\mathbf{\Sigma}}^{-1}(\hat{\mu}_2-\hat{\mu}_1)>\frac{1}{2}\hat{\mu}_2^T\hat{\mathbf{\Sigma}}^{-1}\hat{\mu}_2-\frac{1}{2}\hat{\mu}_1^T\hat{\mathbf{\Sigma}}^{-1}\hat{\mu}_1+\log(N_1/N)-\log(N_2/N) \tag{4.11}$$

否则，将 x 分到类 1。假定我们将两个类的目标分别用 +1 和 −1 编码。容易证明：最小二乘方的系数向量正比于式(4.11)给定的 LDA 方向(见习题 4.2)。事实上，对于目标的任意(相异)编码，这种对应都出现；见习题 4.2。然而，除非 $N_1=N_2$，否则截距不同，因而结果决策规则也不同。

由于通过最小二乘方推导 LDA 方向未对特征使用高斯假定，LDA 方向的适用性可以扩充到高斯数据之外的领域。然而，式(4.11)给定的特定截距或割点的推导确实需要高斯数据。这样，根据经验选择极小化给定数据集的训练误差的割点很有意义。我们发现这在实践上是可行的，但尚未见到文献提及。

当类多于两个时，LDA 与类指示矩阵的线性回归不同，并且它避免了与那种方法有关的屏蔽问题(Hastie 等人，1994)。回归和 LDA 之间的对应可以通过第 12.5 节讨论的最佳得分(optimal scoring)建立。

回到一般判别问题(4.8)，如果不假定 $\mathbf{\Sigma}_k$ 相等，则式(4.9)中的相约不出现；特殊地，x 的二次部分依然存在。于是，我们得到二次判别函数(quadratic discriminant function，QDA)，

$$\delta_k(x)=-\frac{1}{2}\log|\mathbf{\Sigma}_k|-\frac{1}{2}(x-\mu_k)^T\mathbf{\Sigma}_k^{-1}(x-\mu_k)+\log\pi_k \tag{4.12}$$

两个类 k 和 ℓ 之间的判定边界由二次方程 $\{x:\delta_k(x)=\delta_\ell(x)\}$ 给出。

图 4.6 展示了一个例子(取自图 4.1)；其中，三个类是高斯混合分布(见第 6.8 节)，而判定边界用 x 的二次方程逼近。这里，我们图示了拟合二次边界的两种流行方法。右图使用这里介绍的 QDA，而左图使用扩张的 5 维二次多项式空间上的 LDA。差别一般很小；作为 LDA 的

一种方便的替代，QDA 是更可取的方法[①]。

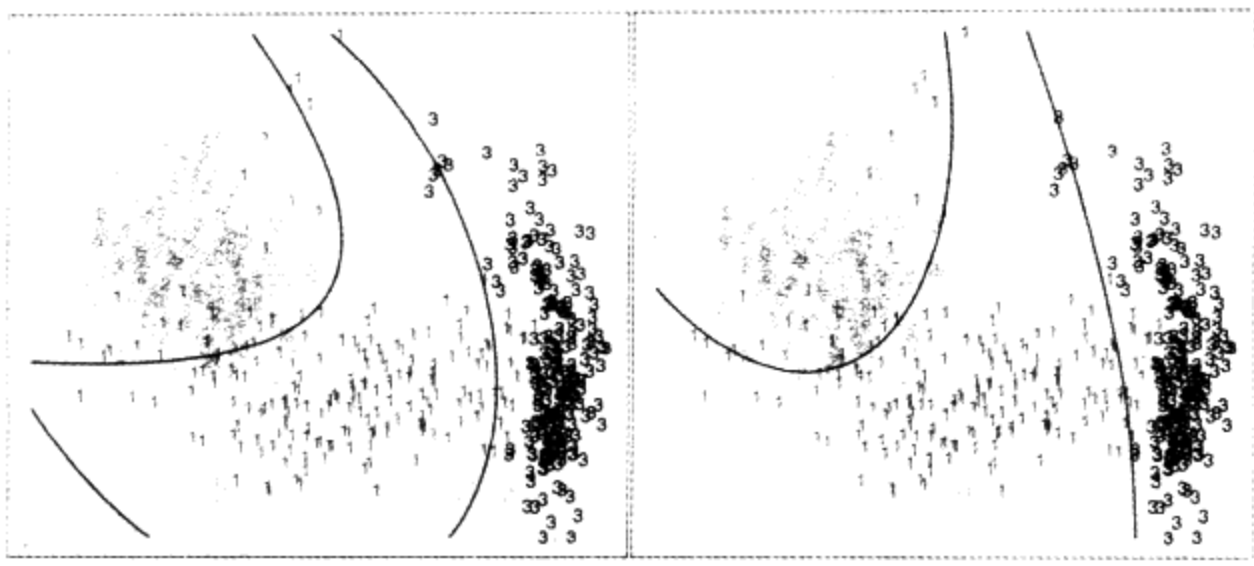


图 4.6 拟合二次边界的两种方法。对于图 4.1 的数据，左图显示二次判定边界（使用5维空间 X_1，X_2，X_1X_2，X_1^2，X_2^2 上的LDA得到）。右图显示QDA发现的二次判定边界。差别很小，通常也是如此（见彩页）

除了必须对每个类分别估计协方差矩阵外，QDA 估计与 LDA 估计类似。当 p 较大时，这可能意味参数急剧增加。由于判定边界是密度参数的函数，计算参数的数目必须小心。对于LDA，看来有 $(K-1)\times(p+1)$ 个参数，这是因为我们只需要判别函数的差 $\delta_k(x)-\delta_K(x)$，其中 K 是某个预先选定的类（这里选定最后一个），而每个差需要 $p+1$ 个参数[②]。类似地，对于QDA，有 $(K-1)\times\{p(p+3)/2+1\}$ 个参数。对于大量各种各样的分类任务，LDA 和 QDA 都具有良好的性能。例如，在 STATLOG 项目中（Michie 等人，1994），对于 22 个数据集中的 7 个，LDA 在最好的 3 个分类法中，QDA 对 4 个数据集在最好的 3 个分类法中，并且对于 10 个数据集，LDA 和 QDA 会有一个在最好的 3 个分类法中。这两种技术都被广泛使用，而本书着重讨论 LDA。看来，不管奇特的工具多么时尚，我们总可以使用这两种简单的工具。这就提出了一个问题：为什么 LDA 和 QDA 有这么好的记录？看来不是因为数据是近似高斯分布的，并且对于 LDA，协方差近似相等。更可能的原因是数据只能支持简单的判定边界（如线性或二次边界），并且通过高斯模型的估计是稳定的。这是偏倚－方差折中——我们可以容忍线性判定边界的偏倚，因为它们能够以比奇特的替代方法低得多的方差进行估计。对于 QDA，这一观点不那么可信，因为它本身也许会有许多参数，尽管可能比非参数方法少。

4.3.1 正则化的判别分析

Friedman（1989）提出了一种 LDA 和 QDA 之间的折中方案，像 LDA 一样，该折中方案允许将 QDA 的各个协方差向一个共同的协方差收缩。这些方法与岭回归非常类似。正则化的协方差矩阵形如：

---

① 对于该图和本书中大多数类似的图，我们通过一种穷举围线法计算判定边界。在点的细格上计算判定规则，然后使用求围线算法计算边界。

② 尽管我们拟合协方差矩阵 $\hat{\boldsymbol{\Sigma}}$，以计算 LDA 判别函数，但估计计算判定边界所需的 $O(p)$ 个参数所需要的是它的一个约化函数。

$$\hat{\mathbf{\Sigma}}_k(\alpha) = \alpha\hat{\mathbf{\Sigma}}_k + (1-\alpha)\hat{\mathbf{\Sigma}} \tag{4.13}$$

其中，$\mathbf{\Sigma}$是合并的协方差矩阵，与 LDA 中使用的相同。这里，$\alpha \in [0, 1]$使得 LDA 和 QDA 之间的模型成为一个连续统，并且需要指定。在实践中，α可以根据模型在验证数据上的性能，或通过交叉验证选取。

图 4.7 显示 RDA 应用于元音数据的结果。训练误差和检验误差都随α增大而减小，尽管当α大于 0.9 时检验误差急剧上升。训练误差和检验误差之间的较大差距部分地是由于在少数个体存在许多重复度量，在训练集和检验集上不同。

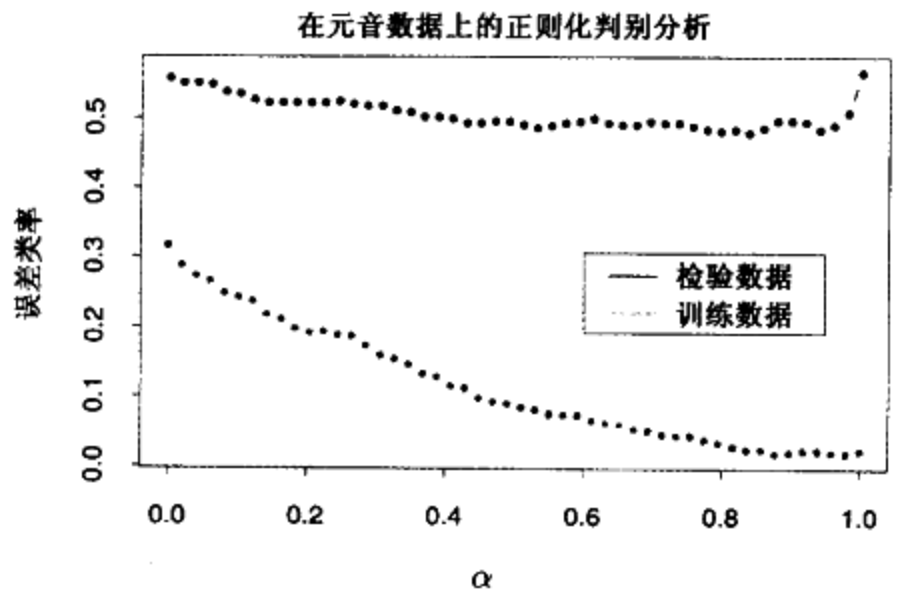


图 4.7　用一系列$\alpha \in [0, 1]$值，使用正则化判别分析，元音数据的检验误差和训练误差。检验数据的最佳值出现在$\alpha = 0.9$附近，接近于二次判别分析

类似的修改允许 $\hat{\mathbf{\Sigma}}$本身向一个标量协方差收缩，

$$\hat{\mathbf{\Sigma}}(\gamma) = \gamma\hat{\mathbf{\Sigma}} + (1-\gamma)\hat{\sigma}^2\mathbf{I} \tag{4.14}$$

其中，$\gamma \in [0, 1]$。用 $\hat{\mathbf{\Sigma}}(\gamma)$替换式(4.13)中的 $\hat{\mathbf{\Sigma}}$导致一族更一般的协方差 $\hat{\mathbf{\Sigma}}(\alpha, \gamma)$，由两个参数表征。

在第 12 章中我们将讨论 LDA 的其他正则化版本，当数据来自数字化模拟信号或图像时，它们更合适。在这些情况下，特征是高维和相关的，并且可以将 LDA 系数正则化，使之在原信号域是光滑或稀疏的。这将导致更好的拓广并使得系数更容易解释。

4.3.2　LDA 计算

作为下一主题的导引，我们暂时偏离主题，讨论一下 LDA，特别是 QDA 所需要的计算。通过将 $\hat{\mathbf{\Sigma}}$或 $\hat{\mathbf{\Sigma}}_k$ 对角化，它们的计算可以简化。对于后者，假定计算每个 $\hat{\mathbf{\Sigma}}_k = \mathbf{U}_k\mathbf{D}_k\mathbf{U}_k^T$ 的本征分解，其中 $\mathbf{U}_k$ 是$p \times p$ 正交的，而 $\mathbf{D}_k$ 是正本征值 $d_{k\ell}$的对角矩阵。则式(4.12)$\delta_k(x)$的成分是：

- $(x-\hat{\mu}_k)^T\hat{\mathbf{\Sigma}}_k^{-1}\ (x-\hat{\mu}_k) = [\mathbf{U}_k^T(x-\hat{\mu}_k)]^T\mathbf{D}_k^{-1}[\mathbf{U}_k^T(x-\hat{\mu}_k)]$；
- $\log|\hat{\mathbf{\Sigma}}_k| = \sum_\ell \log d_{k\ell}$。

按照上面列出的计算步骤，LDA 分类法可以用如下两步实现：

- 关于共同的协方差估计 $\hat{\boldsymbol{\Sigma}}$ 球形化数据：$X^* \leftarrow \mathbf{D}^{-1/2}\mathbf{U}^T X$，其中 $\hat{\boldsymbol{\Sigma}} = \mathbf{U}\mathbf{D}\mathbf{U}^T$。则 X^* 共同的协方差估计就是该等式。
- 分类到变换后空间中最近的类形心，以类的先验概率 π_k 的效应为模。

4.3.3 降秩线性判别分析

迄今为止，作为一种受限的高斯分类法，我们讨论了 LDA。它的流行部分因为附加的限制允许我们观察数据的富含信息的低维投影。

p 维输入空间的 K 个形心在一个小于或等于 $K-1$ 维的仿射子空间中，并且如果 p 比 K 大得多，则可以相当可观地降低维数。此外，为了确定最近的形心，我们可以忽略正交于该子空间的距离，因为它们对每个类所起的作用相等。这样，也可以将 X^* 投影到这个形心生成的子空间 H_{K-1}，并在那里做距离比较。这样，LDA 中存在一个基本的维归约，即最多需要在 $K-1$ 维子空间上考虑数据。例如，如果 $K=3$，使得我们可以在一个 2 维图上观察数据，用颜色对每个类编码。这样就不必放弃 LDA 分类所需要的任何信息。

如果 $K>3$ 会怎么样？我们可以寻找某个 $L<K-1$ 子空间 $H_L \subseteq H_{K-1}$，在某种意义下对 LDA 是最佳的。Fisher 定义最佳的含义是投影后的形心在方差意义下尽可能分散。这实际上等价于找形心本身的主成分子空间（主成分的简要介绍见第 3.4.4 节，而更详细的讨论见第 14.5.1 节）。图 4.4 显示元音数据的这种最佳二维子空间。这里，在 10 维输入空间上有 11 个类，每个类是一个不同的元音。在此情况下，形心需要整个空间，因为 $K-1=p$，但我们显示了一个最佳 2 维子空间。维是有序的，因此可以顺序计算附加的维。图 4.8 显示了 4 对附加的坐标，也称标准（canonical）或判别（discriminant）变量。

概括地说，寻找 LDA 的最佳子空间序列涉及如下步骤：

- 计算 $K \times p$ 的类形心矩阵 $\mathbf{M}$ 和公共协方差矩阵 $\mathbf{W}$（关于类内协方差）；
- 使用 $\mathbf{W}$ 的本征分解计算 $\mathbf{M}^* = \mathbf{M}\mathbf{W}^{-1/2}$；
- 计算 $\mathbf{M}^*$ 的协方差矩阵 $\mathbf{B}^*$（$\mathbf{B}$ 表示类间协方差）和它的本征分解 $\mathbf{B}^* = \mathbf{V}^* \mathbf{D}_B \mathbf{V}^{*T}$。$\mathbf{V}^*$ 的列 v_ℓ^* 从第一个到最后一个依次定义最佳子空间坐标。

将所有这些操作结合在一起，第 ℓ 个判别变量由 $Z_\ell = v_\ell^T X$ 给出，这里，$v_\ell = \mathbf{W}^{-1/2} v_\ell^*$。

Fisher 通过不同的途径得到了这个分解，完全没有涉及高斯分布。他提出了以下问题：

寻找线性组合 $Z = a^T X$，使得类间方差相对于类内方差极大化。

类间方差是 Z 的类均值的方差，而类内方差是关于均值的合并方差。图 4.9 表明该标准为什么是合理的。尽管连接形心的方向尽可能地分开均值（即，极大化类间方差），但是由于协方差的特性，投影后的类之间仍然有相当大的重叠。通过同时考虑协方差，可以找到极小重叠的方向。

Z 的类间方差是 $a^T\mathbf{B}a$，而类内方差是 $a^T\mathbf{W}a$，$\mathbf{W}$ 在前面已定义，而 $\mathbf{B}$ 是类形心矩阵 $\mathbf{M}$ 的协方差矩阵。注意，$\mathbf{B}+\mathbf{W}=\mathbf{T}$，其中 $\mathbf{T}$ 是 X 的全协方差矩阵，忽略类信息。

因此，Fisher 的问题实际是极大化瑞利商（Rayleigh quotient）：

$$\max_a \frac{a^T\mathbf{B}a}{a^T\mathbf{W}a} \tag{4.15}$$

或等价地：

$$\max_a a^T \mathbf{B} a, \quad \text{受限于} \quad a^T \mathbf{W} a = 1 \tag{4.16}$$

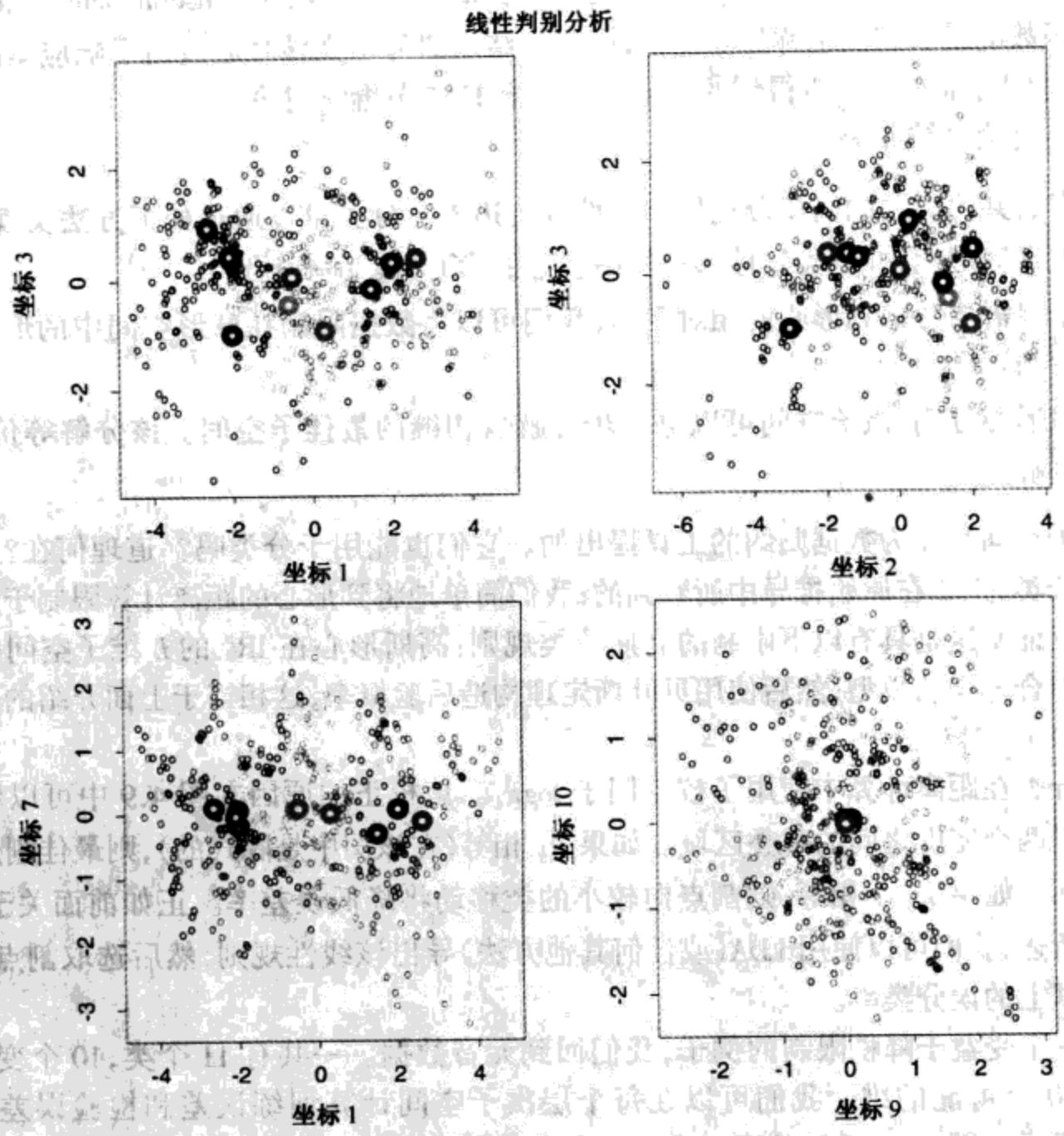


图 4.8　到标准变量对上的 4 个投影。注意，随标准变量的秩增加，形心变得集中。在右下图，它们似乎被叠加，并且类变得最乱（见彩页）

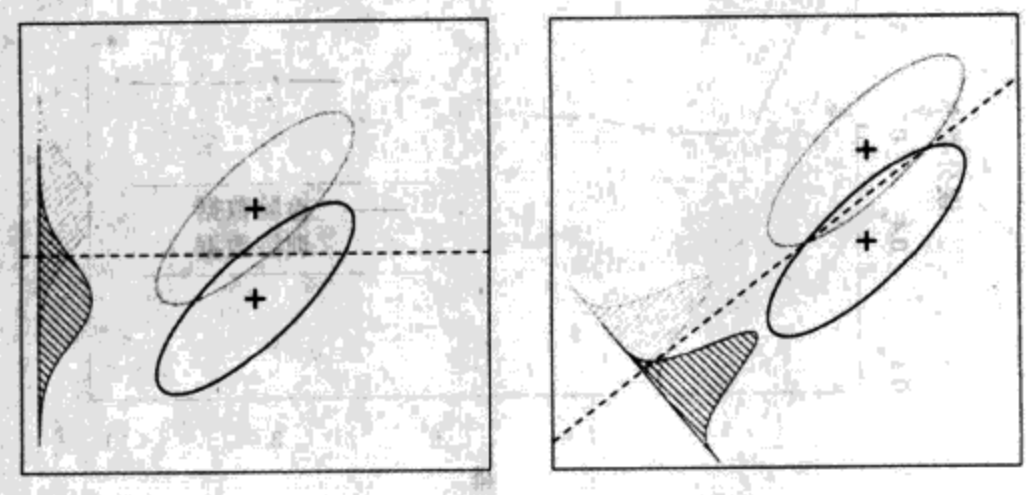


图 4.9　尽管连接形心线定义了极大形心伸展方向，但是由于协方差的原因，投影后的数据依然重叠（左图）。对于高斯数据，判别方向使重叠极小化（右图）

这是一个广义本征值问题，其中 a 由 $\mathbf{W}^{-1}\mathbf{B}$ 的最大本征值给定。不难证明(见习题 4.1)最优的 a_1 等于上面定义的 v_1。类似地，我们可以找出下一个方向 a_2，在 $\mathbf{W}$ 中正交于 a_1，使得 $a_2^T\mathbf{B}a_2/a_2^T\mathbf{W}a_2$ 最大；这个解是 $a_2 = v_2$，如此等等。a_l 称为判别坐标(discriminant coordinate)，不要与判别函数混淆。它们也称标准变量，因为该结果的另一推导是通过指示响应矩阵 $\mathbf{Y}$ 在预测矩阵 $\mathbf{X}$ 上的标准相关分析得到的。我们将在第 12.5 节继续讨论。

将上述讨论总结如下：

- 具有公共协方差的高斯分类导致线性判定边界。分类可以通过如下方法实现：关于 $\mathbf{W}$ 对数据球形化，并分类到球形空间中最近的形心(模 $\log\pi_k$)。
- 由于只需要考虑到形心的相对距离，我们可以将数据限制在球形空间中的形心生成的子空间。
- 按照形心分割，该子空间可以进一步分解成相继的最佳子空间。该分解等价于 Fisher 的分解。

归约子空间是作为数据归约的工具提出的。它们也能用于分类吗？道理何在？显然，它们能用于分类，正如在原始推导中所看到的；我们简单地将到形心的距离计算限制于选定的子空间。可以证明这是具有以下限制的高斯分类规则：高斯形心在 IR^p 的 L 维子空间中。通过极大似然拟合这样的模型，然后使用贝叶斯定理构造后验概率，这相当于上面介绍的分类规则(见习题 4.8)。

高斯分类在距离计算中使用了校正因子 $\log\pi_k$。该校正的原因在图 4.9 中可以看出。误分类率基于两个密度之间的重叠区域。如果 π_k 相等(在该图中是隐式的)，则最佳割点在投影均值的中间。如果 π_k 不相等，则割点向较小的类移动将降低误差率。正如前面关于 2-类问题的讨论所述，我们可以使用 LDA(或任何其他方法)导出该线性规则，然后选取割点，以极小化训练数据上的误分类率。

作为一个受益于降秩限制的例子，我们回到元音数据。一共有 11 个类，10 个变量，因此分类法有 10 个可能的维。我们可以在每个层次子空间计算训练误差和检验误差，结果在图 4.10中显示。图 4.11 显示基于二维 LDA 解的分类法的判定边界。

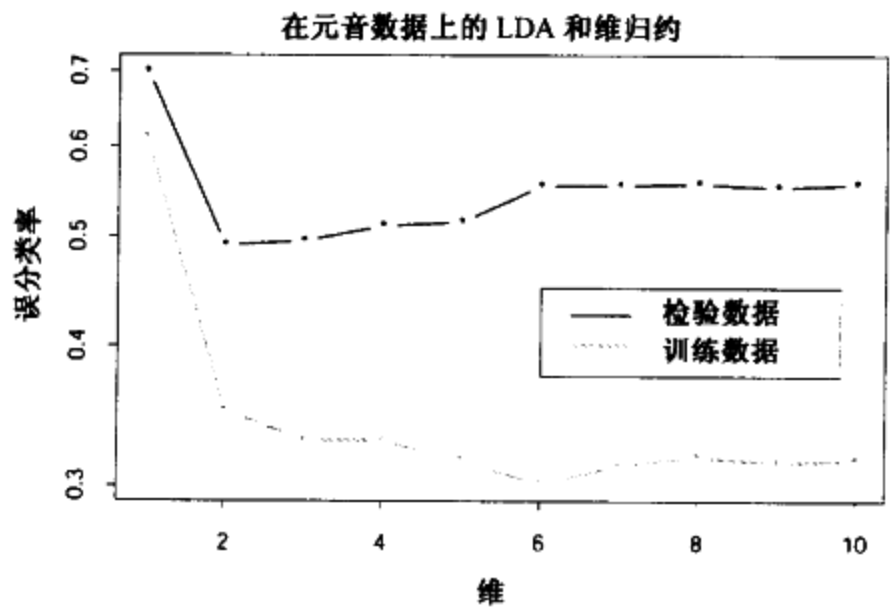


图 4.10 元音数据的训练和检验误差率，作为判别子空间的维的函数。在此例中，最好的误差率在二维。图4.11显示该空间中的判定边界

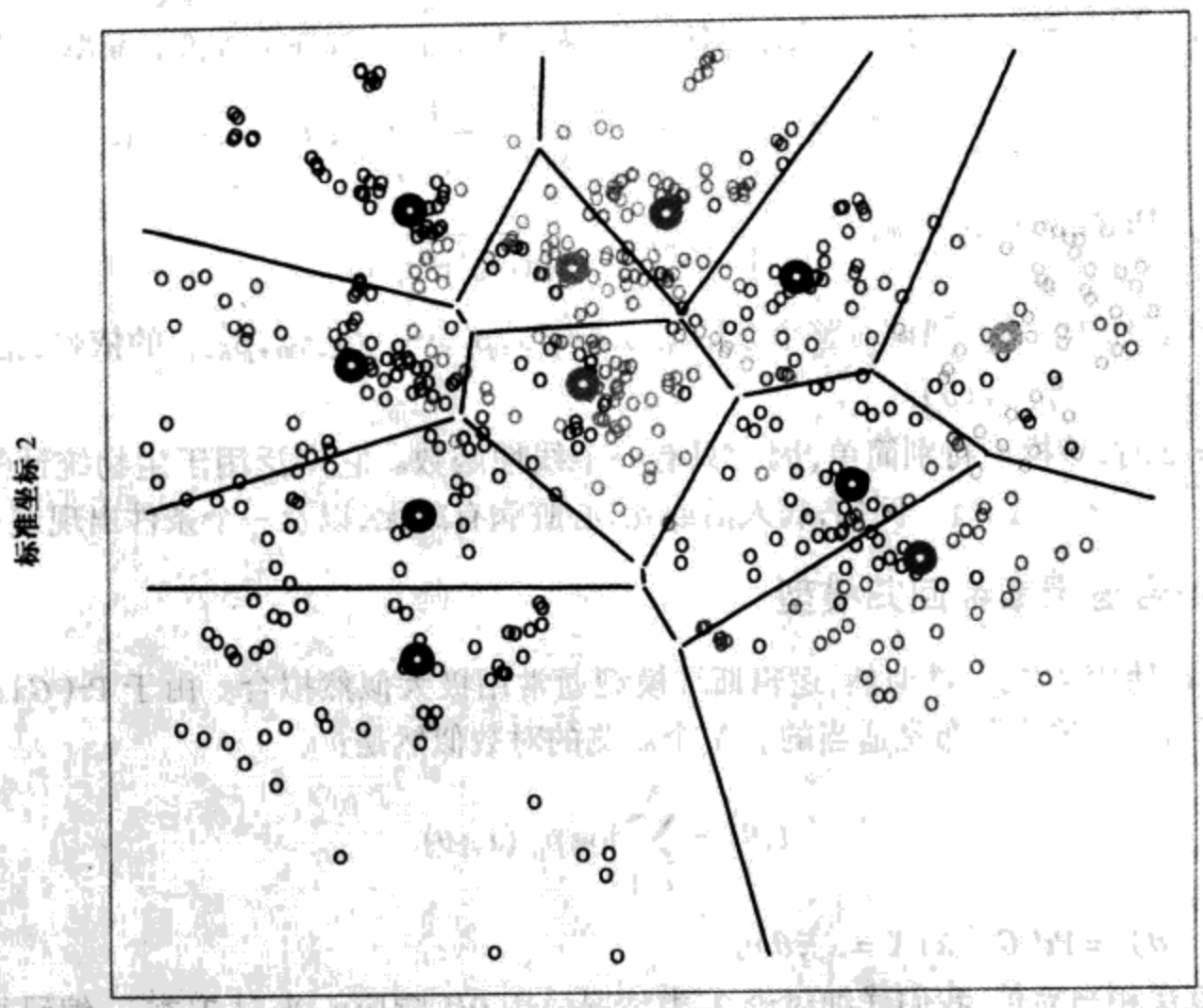


图 4.11　在前两个标准变量生成的 2 维子空间中元音训练数据的判定边界。注意，在任意较高维的子空间中，判定边界是较高维的仿射平面，不能用线表示(见彩页)

在 Fisher 的降秩判别分析和指示响应矩阵的回归之间存在紧密联系。这样，LDA 实际上是回归，后随 $\hat{\mathbf{Y}}^T\mathbf{Y}$ 的本征分解。在两个类的情况下，只有单个判别变量，它等于 $\hat{\mathbf{Y}}$ 的每一列的标量乘积。这些联系将在第 12 章做进一步讨论。一个相关的事实是：如果将原来的预测子 $\mathbf{X}$ 变换到 $\hat{\mathbf{Y}}$，则使用 $\hat{\mathbf{Y}}$ 的 LDA 等价于原空间上的 LDA(见习题 4.3)。

4.4　逻辑斯缔回归

逻辑斯缔回归(logistic regression)模型源于这样一种愿望：通过 x 的线性函数对 K 个类的后验概率建模，而同时确保它们的和为 1，并都在[0,1]中。该模型具有如下形式：

$$\begin{aligned}
\log\frac{\Pr(G=1|X=x)}{\Pr(G=K|X=x)} &= \beta_{10}+\beta_1^T x\\
\log\frac{\Pr(G=2|X=x)}{\Pr(G=K|X=x)} &= \beta_{20}+\beta_2^T x\\
&\vdots\\
\log\frac{\Pr(G=K-1|X=x)}{\Pr(G=K|X=x)} &= \beta_{(K-1)0}+\beta_{K-1}^T x
\end{aligned} \tag{4.17}$$

该模型用 $K-1$ 个对数几率或分对数变换确定(反映概率和为 1 的约束)。尽管模型使用最后一

个类作为几率中的分母，但分母的选择是任意的，因为估计在该选择下等价。简单的计算得到：

$$\begin{aligned}\Pr(G=k|X=x) &= \frac{\exp(\beta_{k0}+\beta_k^T x)}{1+\sum_{\ell=1}^{K-1}\exp(\beta_{\ell 0}+\beta_\ell^T x)},\ k=1,\cdots,K-1\\ \Pr(G=K|X=x) &= \frac{1}{1+\sum_{\ell=1}^{K-1}\exp(\beta_{\ell 0}+\beta_\ell^T x)}\end{aligned} \tag{4.18}$$

而它们的和显然为 1。为强调对整个参数集 $\theta=\{\beta_{10},\beta_1^T,\cdots,\beta_{(K-1)0},\beta_{K-1}^T\}$ 的依赖，记该概率为 $\Pr(G=k|X=x)=p_k(x;\theta)$。

当 $K=2$ 时，该模型特别简单，因为只有一个线性函数。它广泛用于生物统计学，那里二元响应（两个类）相当普遍。例如，病人活或死，心脏病有或无，以及一个条件出现或不出现。

4.4.1 拟合逻辑斯缔回归模型

给定 X，使用 G 的条件似然，逻辑斯缔模型通常用极大似然拟合。由于 $\Pr(G|X)$ 完全指定该条件分布，多项式分布是适当的。N 个观测的对数似然是：

$$\ell(\theta)=\sum_{i=1}^{N}\log p_{g_i}(x_i;\theta) \tag{4.19}$$

其中，$p_k(x_i;\theta)=\Pr(G=k|X=x_i;\theta)$。

由于算法相当简单，我们详细讨论 2-类情况。用 0/1 响应 y_i 来对 2-类 g_i 编码很方便，其中，当 $g_i=1$ 时 $y_i=0$，而当 $g_i=2$ 时 $y_i=1$。设 $p_1(x;\theta)=p(x;\theta)$，而 $p_2(x;\theta)=1-p(x;\theta)$。对数似然可以写成：

$$\begin{aligned}\ell(\beta) &= \sum_{i=1}^{N}\left\{y_i\log p(x_i;\beta)+(1-y_i)\log(1-p(x_i;\beta))\right\}\\ &= \sum_{i=1}^{N}\left\{y_i\beta^T x_i-\log(1+e^{\beta^T x_i})\right\}\end{aligned} \tag{4.20}$$

这里，$\beta=\{\beta_{10},\beta_1\}$，并且我们假定输入向量 x_i 包含一个常数项 1，以便接纳截距。

为极大化对数似然，我们令它的导数等于零。这些得分方程是：

$$\frac{\partial\ell(\beta)}{\partial\beta}=\sum_{i=1}^{N}x_i(y_i-p(x_i;\beta))=0 \tag{4.21}$$

这是 $p+1$ 个 β 上的非线性方程。注意，由于 x_i 的第一个分量为 1，第一个得分方程可确定 $\sum_{i=1}^{N}y_i=\sum_{i=1}^{N}p(x_i;\beta)$；类 1 的期望数与观测数匹配（类 2 也如此）。

为解得分式(4.21)，我们使用 Newton-Raphson 算法，这需要二阶导数或 Hessian 矩阵：

$$\frac{\partial^2\ell(\beta)}{\partial\beta\partial\beta^T}=-\sum_{i=1}^{N}x_i x_i^T p(x_i;\beta)(1-p(x_i;\beta)) \tag{4.22}$$

以 β^{old} 开始，单个 Newton-Raphson 更新是：

$$\beta^{\text{new}} = \beta^{\text{old}}-\left(\frac{\partial^2\ell(\beta)}{\partial\beta\partial\beta^T}\right)^{-1}\frac{\partial\ell(\beta)}{\partial\beta} \tag{4.23}$$

其中，导数在 β^{old} 处计算。

将得分和 Hessian 写成矩阵形式是方便的。设 $\mathbf{y}$ 表示 y_i 值向量，$\mathbf{X}$ 是 x_i 值的 $N \times (p+1)$ 矩阵，$\mathbf{p}$ 是其第 i 个元素为 $p(x_i;\beta^{old})$ 的拟合概率向量，而 $\mathbf{W}$ 是权的 $N \times N$ 对角矩阵，第 i 个对角线元素为 $p(x_i;\beta^{old})(1-p(x_i;\beta^{old}))$。则 $\frac{\partial \ell(\beta)}{\partial \beta} = \mathbf{X}^T(\mathbf{y}-\mathbf{p})$，$\frac{\partial^2 \ell(\beta)}{\partial \beta \partial \beta^T} = -\mathbf{X}^T\mathbf{W}\mathbf{X}$。

这样，Newton-Raphson 步骤是：

$$\begin{aligned}\beta^{new} &= \beta^{old} + (\mathbf{X}^T\mathbf{W}\mathbf{X})^{-1}\mathbf{X}^T(\mathbf{y}-\mathbf{p}) \\ &= (\mathbf{X}^T\mathbf{W}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{W}\left(\mathbf{X}\beta^{old} + \mathbf{W}^{-1}(\mathbf{y}-\mathbf{p})\right) \\ &= (\mathbf{X}^T\mathbf{W}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{W}\mathbf{z}\end{aligned} \tag{4.24}$$

在第二行和第三行，我们使用响应

$$\mathbf{z} = \mathbf{X}\beta^{old} + \mathbf{W}^{-1}(\mathbf{y}-\mathbf{p}) \tag{4.25}$$

重新将 Newton-Raphson 步骤表示成加权最小二乘方步骤。式(4.25)有时也称调整的响应。这些方程重复地求解，因为在每一步迭代 $\mathbf{p}$ 都变化，从而 $\mathbf{W}$ 和 $\mathbf{z}$ 也变化。该算法称做迭代加权最小二乘方(IRLS)，因为每次迭代都要解加权最小二乘方问题：

$$\beta^{new} \leftarrow \arg\min_{\beta}(\mathbf{z}-\mathbf{X}\beta)^T\mathbf{W}(\mathbf{z}-\mathbf{X}\beta) \tag{4.26}$$

尽管不能保证收敛，但是看来 $\beta=0$ 是迭代过程的一个好的初值。典型地，因为对数似然是凹的，该算法确实收敛，但可能过头。在少数情况下，对数似然递减，步长减半将保证收敛性。

对于多类($K \geqslant 3$)情况，牛顿算法也可以用迭代加权最小二乘方算法表达，但用 $K-1$ 个响应的向量，并且每个观测一个非对角的权矩阵。后者排除了简化的算法，但在此情况下它是数值的，更便于直接与扩展的向量 θ 一起计算(见习题 4.4)。

逻辑斯缔回归模型更多地用做数据分析和推理工具，其目标是理解输入变量在解释结果中的作用。典型地，许多模型适合用于寻找简洁的模型，它们涉及变量的一个子集，或许还有交叉项。下面的例子解释了所涉及的一些问题。

4.4.2　例：南非心脏病

这里，我们提供一个二元数据分析，解释逻辑斯缔回归模型传统的统计学应用。图 4.12中的数据是冠心风险因素研究(Coronary Risk-Factor Study，CORIS)基本调查数据的一个子集。CORIS 在南非 Western Cape 的三个农村地区进行(Rousseauw 等人，1983)。该研究的目标是建立高发地区缺血性心脏病的风险因素。数据取自年龄在 15 岁到 64 岁之间的白人男性，响应变量是调查时是否发生心肌梗塞(MI)(该地区 MI 的总体发病率为 5.1%)。在我们的数据中有 160 个病例，302 个对照样本。在 Hastie 和 Tibshirani(1987)中有对这些数据更详细的描述。

我们用极大似然拟合该模型，产生的结果列在表 4.2 中。

表 4.2　逻辑斯缔回归拟合南非心脏病数据的结果

	系数	标准误差	Z-得分
(截距)	-4.130	0.964	-4.285
sbp	-0.006	0.006	1.023

（续表）

	系数	标准误差	Z得分
tobacco	0.080	0.026	3.034
ldl	0.185	0.057	3.219
famhist	0.939	0.225	4.178
obesity	-0.035	0.029	-1.187
alcohol	0.001	0.004	0.136
age	0.043	0.010	4.184

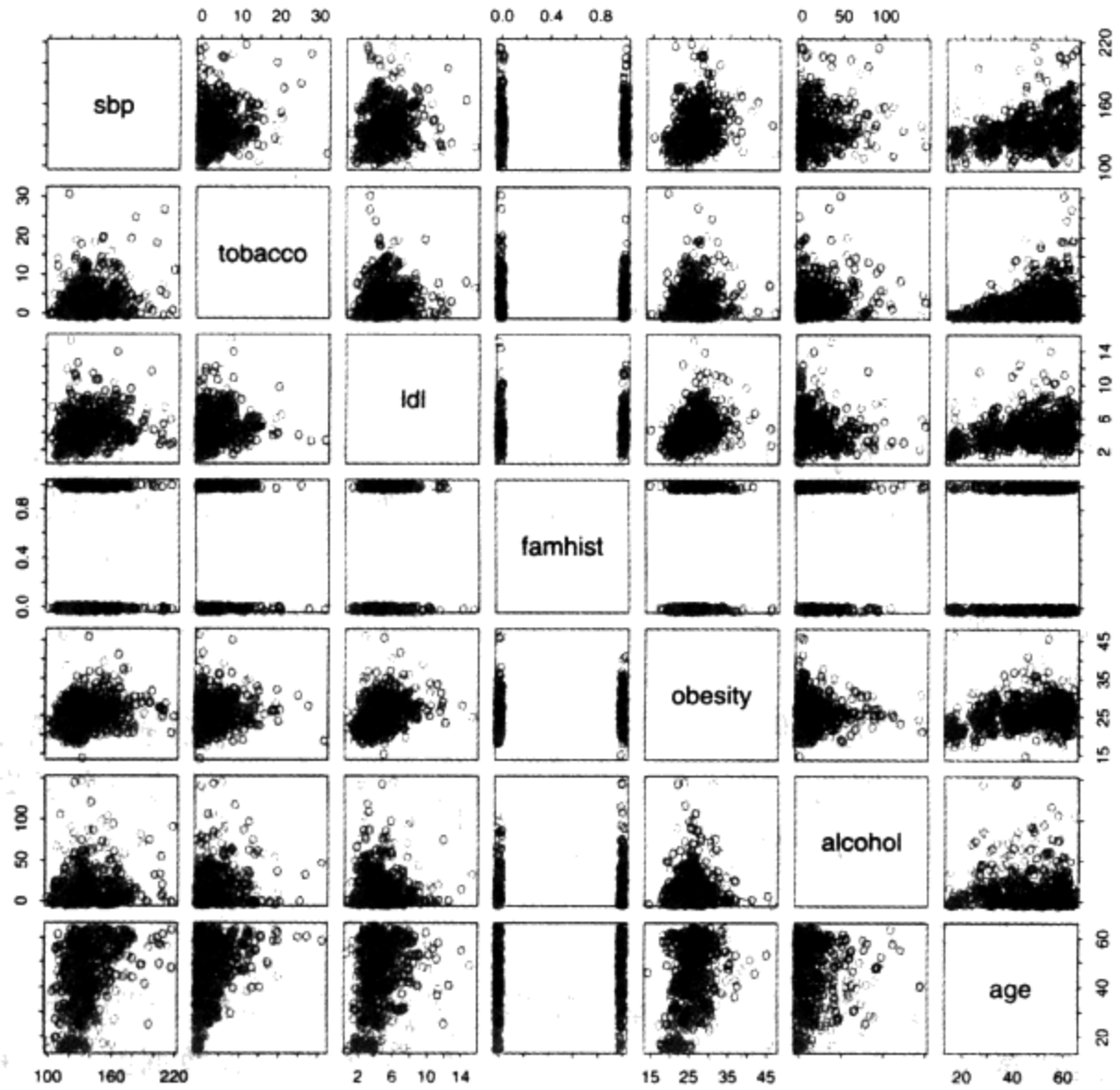


图 4.12 南非心脏病数据的散点图。每幅图显示一对风险因素，并且病例和控制用颜色编码（红色是病例）。心脏病家族史变量（famhist）是二元的（yes或no）（见彩页）

表 4.2 的汇总数据包含模型中每个系数的 Z-得分（系数除以标准误差）；一个非显著的 Z-得分表明该系数可以从模型中删除。每一个形式地对应于原假设的一个检验：有问题系数为 0，其余系数非 0（也称为 Wald 检验）。绝对值近似大于 2 的 Z-得分在 5% 水平上是显著的。

系数表中的有些结果令人吃惊，必须小心解释。收缩血压（sbp）是不显著的！肥胖（obesity）也不显著，并且它的符号为负。这种混乱是由预测子集合之间的相关性导致的。就自身而言，sbp和obesity都是显著的，并具有正号。然而，在一些其他相关的变量存在时，它们就不再需要（甚至取负号）。

现在，分析家可能要做某种模型选择；找出变量的一个子集，该子集能够充分解释其对冠心病流行的联合影响。一种处理方法是删除最不显著的系数，并重新拟合模型。重复该过程，直到不能从模型中删除任何项为止。这产生表 4.3 给出的模型。

表 4.3　逐步逻辑斯缔回归拟合南非心脏病数据的结果

	系数	标准误差	Z 得分
（截距）	-4.204	0.498	-8.45
tobacco	0.081	0.026	3.16
ldl	0.168	0.054	3.09
famhist	0.924	0.223	4.14
age	0.044	0.010	4.52

一种更好但更费时的策略是对删除一个变量的每个模型重新进行拟合，然后进行散离分析（analysis of deviance），决定排除哪个变量。拟合模型的残散离是其对数似然的负二倍，而两个模型之间的散离是它们的残散离之差（与平方和类似）。该策略给出与上面相同的最终结果。

如何解释系数，例如，tobacco（烟草）的系数 0.081（标准误差 = 0.026）？烟草按一生的用量以公斤度量，对于对照样本，中值为 1.0 kg，而对于病例，中值为 4.1 kg。这样，一生烟草的用量增加 1.0 kg，患冠心病的几率增加 $\exp(0.081) = 1.084$ 或 8.4%。结合标准误差，我们得到一个近似的 95% 置信区间 $\exp(0.081 \pm 2 \times 0.026) = (1.03, 1.14)$。

第 5 章将再次考虑这些数据。那里，我们将看到有些变量具有非线性效应，适当地建模时不能从模型中排除。

4.4.3　二次逼近和推理

极大似然参数估计 $\hat{\beta}$ 满足自相容联系：它们是加权最小二乘方拟合的系数，其中响应是：

$$z_i = x_i^T \hat{\beta} + \frac{(y_i - \hat{p}_i)}{\hat{p}_i(1 - \hat{p}_i)} \tag{4.27}$$

而权是 $w_i = \hat{p}_i(1 - \hat{p}_i)$，它们都依赖于 $\hat{\beta}$ 本身。除了提供一个方便的算法外，这种与最小二乘方的联系还能提供：

- 加权的残差平方和是熟悉的 Pearson χ^2 统计量

$$\sum_{i=1}^{N} \frac{(y_i - \hat{p}_i)^2}{\hat{p}_i(1 - \hat{p}_i)} \tag{4.28}$$

  对散离的二次近似。
- 渐近似然定理表明，如果模型是正确的，则 $\hat{\beta}$ 是相容的（即收敛到真正的 β）。
- 中心极限定理表明，$\hat{\beta}$ 的分布收敛到 $N(\beta, (\mathbf{X}^T\mathbf{W}\mathbf{X})^{-1})$。通过模仿正规的定理推理，这个和其他渐近可以直接从加权最小二乘方拟合推出。
- 对于逻辑斯缔回归模型，模型的建立可能代价很高，因为每个拟合模型都需要迭代。流行的捷径是 Rao 得分检验（检验项包含）和 Wald 检验（用于检验项排除）。它们都不需

要迭代拟合，而是基于当前模型的极大似然拟合。它们实际上是使用相同的权，向加权的最小二乘方拟合添加项或从中删除项。这样的计算可以有效地进行，而不必重新计算整个加权的最小二乘方拟合。

软件实现可以利用这些联系。例如，S-PLUS 中的广义线性建模软件(包含逻辑斯缔回归，作为二次模型簇的一部分)就充分地利用了它们。GLM(广义线性模型)对象可以看做线性模型对象，并且所有可以用于线性模型的工具都自动地可以使用。

4.4.4 逻辑斯缔回归还是 LDA

在第 4.3 节，我们发现类 k 和 K 之间的对数后验几率是 x 的线性函数(4.9)：

$$\begin{aligned}\log\frac{\Pr(G=k|X=x)}{\Pr(G=K|X=x)} &= \log\frac{\pi_k}{\pi_K}-\frac{1}{2}(\mu_k+\mu_K)^T\mathbf{\Sigma}^{-1}(\mu_k-\mu_K)\\ &\quad +x^T\mathbf{\Sigma}^{-1}(\mu_k-\mu_K)\\ &= \alpha_{k0}+\alpha_k^T x\end{aligned} \tag{4.29}$$

这种线性是类密度的高斯假设，以及公共协方差矩阵假设的推论。依据构造，线性逻辑斯缔模型(4.17)具有线性分对数：

$$\log\frac{\Pr(G=k|X=x)}{\Pr(G=K|X=x)}=\beta_{k0}+\beta_k^T x \tag{4.30}$$

看起来模型是相同的。尽管它们具有完全相同的形式，但不同之处在于估计线性系数的方法。逻辑斯缔回归模型更一般，因为它做的假设少一些。我们可以将 X 和 G 的联合密度写成：

$$\Pr(X,G=k)=\Pr(X)\Pr(G=k|X) \tag{4.31}$$

其中，$\Pr(X)$表示输入 X 的边缘密度。对于 LDA 和逻辑斯缔回归，右边的第二项都具有分对数线性形式：

$$\Pr(G=k|X=x)=\frac{e^{\beta_{k0}+\beta_k^T x}}{1+\sum_{\ell=1}^{K-1}e^{\beta_{\ell 0}+\beta_\ell^T x}} \tag{4.32}$$

这里，我们又任意选取最后一个类来引用。

逻辑斯缔回归模型允许 X 的边缘密度为任意密度函数 $\Pr(X)$，并通过极大化条件似然——具有概率 $\Pr(G=k|X)$的多项式似然——拟合 $\Pr(G|X)$的参数。尽管 $\Pr(X)$完全被忽略，但是我们可以将这个边缘密度看做使用在每个观测放置 $1/N$ 质量的经验分布函数，以完全非参数和无限制的方式进行估计。

对于 LDA，我们通过极大化全对数似然拟合参数，基于联合概率：

$$\Pr(X,G=k)=\phi(X;\mu_k,\mathbf{\Sigma})\pi_k \tag{4.33}$$

其中，ϕ 是高斯密度函数。标准正态理论容易导致第 4.3 节给出的估计$\hat{\mu}_k$，$\hat{\mathbf{\Sigma}}$和$\hat{\pi}_k$。由于逻辑斯缔形式(4.29)的线性参数是高斯参数的函数，通过插入对应的估计，我们得到它们的极大似然估计。然而，与条件似然不同，边缘密度 $\Pr(X)$在这里确实起作用。它是一个混合密度

$$\Pr(X)=\sum_{k=1}^{K}\pi_k\phi(X;\mu_k,\mathbf{\Sigma}) \tag{4.34}$$

也涉及这些参数。

这个附加的成分/限制可以起什么作用呢？依赖附加的模型假设，我们具有更多的关于参数的信息，因此能够更有效地估计它们（较低的方差）。如果真实的 $f_k(x)$ 是高斯的，则在最坏的情况下，忽略似然的边缘部分按误差率渐近地引起大约 30% 的有效性损失（Efron，1975）。也就是说：多用 30% 的数据，条件似然将做得同样好。

例如，远离判定边界的观测（被逻辑斯缔回归降低权值）在估计公共协方差矩阵中起作用。这不是一个好消息，因为它还意味 LDA 对于离群点不健壮。

由该混合公式显然可以看出，即便不带类标号的观测也包含关于参数的信息。通常，产生类标号是昂贵的，而未分类的观测容易得到。依赖强模型假设（如这里所做的），我们可以使用两种类型的信息。

边缘似然可以看做正规化算子，在某种意义下要求类密度是通过该边缘可见的。例如，如果 2-类逻辑斯缔回归模型中的数据可以很好地被一个超平面分开，则参数的极大似然估计是不确定的（即无限的，见习题 4.5）。对于同样的数据，LDA 系数将是确定的，因为边缘似然将不允许这些退化。

在实践中，这些假设不成立，并且 X 的某些分量常常是定性变量。通常认为逻辑斯缔回归比 LDA 更安全、更健壮，它依赖于较少的假设。我们的经验是：这些模型给出非常类似的结果，即便不适当地使用 LDA（如，用于定性预测）时也是如此。

4.5　分离超平面

我们看到线性判别分析和逻辑斯缔回归都用类似但稍微不同的方式估计线性判定边界。本章的其余部分将介绍分离超平面分类法。这些过程构造线性判定边界，试图显式地、尽可能好地将数据分到不同的类。它们为第 12 章讨论的支持向量分类法奠定了基础。本节的数学味比前几节稍微浓一些。

图 4.13 显示 $\mathbb{R}^2$ 中两个类的 20 个数据点。这些数据可以被一个线性边界分隔开。图中显示的是无限多个可能的分离超平面中的两个（蓝线）。橙色线是该问题的最小二乘方解，通过对 X 上的 −1/1 响应 Y 回归（有截距）得到；该直线由下式给出：

$$\{x : \hat{\beta}_0 + \hat{\beta}_1 x_1 + \hat{\beta}_2 x_2 = 0\} \tag{4.35}$$

这个最小二乘方解不能很好地分隔这些点，它分错了一个点。这也是 LDA 找到的边界，因为在 2-类情况下，LDA 与线性回归等价（见第 4.3 节和习题 4.2）。

在上世纪 50 年代后期，像式（4.35）这样计算输入特征的线性组合，并返回符号的分类法在工程文献中称做感知器（perceptron）（Rosenblatt，1958）。感知器为 20 世纪 80 年代和 90 年代的神经网络模型奠定了的基础。

在继续讨论之前，我们稍微偏离主题，回顾一下向量代数。图 4.14 绘出了由方程 $f(x) = \beta_0 + \beta^T x = 0$ 定义的超平面或仿射集 L，由于在 $\mathbb{R}^2$ 中，这是一条直线。

下面，我们列出一些性质：

1. 对于 L 中的任意两个点 x_1 和 x_2，$\beta^T(x_1 - x_2) = 0$，从而 $\beta^* = \beta / \|\beta\|$ 是到面 L 的法向量。

2. 对于 L 中的任意点 x_0，$\beta^T x_0 = -\beta_0$。

3. 任意点 x 到 L 的有符号的距离由下式给出：

$$\begin{aligned} \beta^{*T}(x - x_0) &= \frac{1}{\|\beta\|}(\beta^T x + \beta_0) \\ &= \frac{1}{\|f'(x)\|} f(x) \end{aligned} \tag{4.36}$$

这里，$f(x)$正比于从 x 到$f(x)=0$定义的超平面的有符号距离。

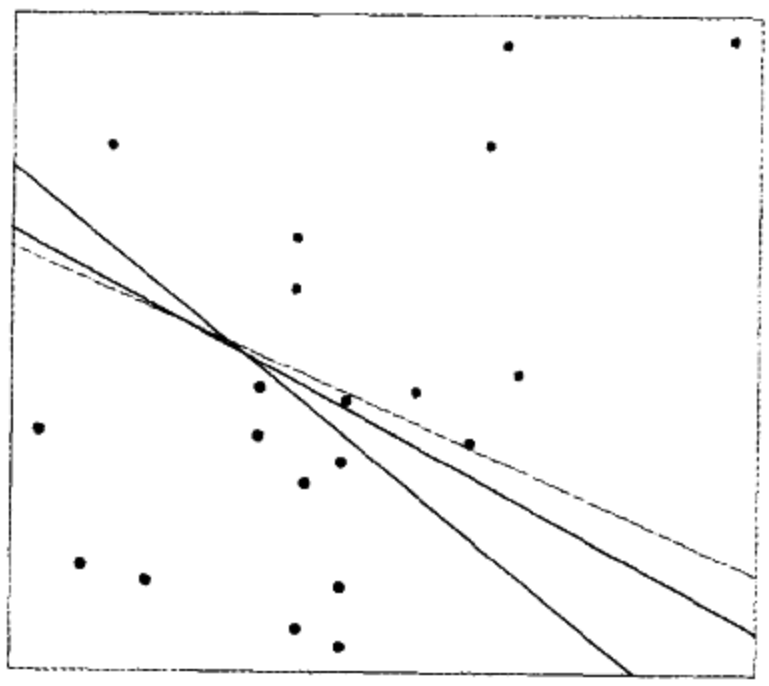


图 4.13 一个小例子，包含两个可被超平面分隔的类。橙色线是最小二乘方解，它将一个训练数据误分类。图中还显示了两个蓝色分隔超平面，它们被以不同的随机初始化的感知器学习算法找出(见彩页)

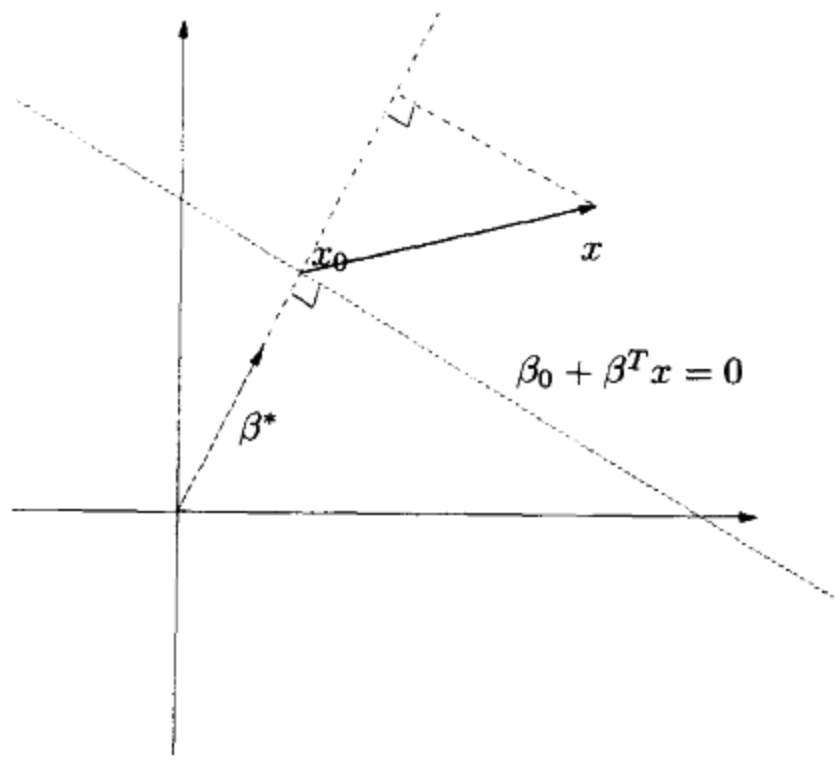


图 4.14 超平面(仿射集)的线性代数

4.5.1 Rosenblatt 的感知学习算法

感知学习算法(perceptron learning algorithm)试图通过极小化误分类点到判定边界的距离来

找出分离超平面。如果响应$y_i = 1$被误分类，则 $x_i^T\beta + \beta_0 < 0$，而对于误分类的响应$y_i = -1$则相反($x_i^T\beta + \beta_0 > 0$)。目标是极小化

$$D(\beta, \beta_0) = -\sum_{i\in\mathcal{M}} y_i(x_i^T\beta + \beta_0) \tag{4.37}$$

其中，$\mathcal{M}$ 是误分类点的下标集。这个量是非负的，并正比于误分类点到 $\beta^T x + \beta_0 = 0$ 定义的判定边界的距离。(假定$\mathcal{M}$ 是固定的)梯度由

$$\partial\frac{D(\beta, \beta_0)}{\partial\beta} = -\sum_{i\in\mathcal{M}} y_i x_i \tag{4.38}$$

$$\partial\frac{D(\beta, \beta_0)}{\partial\beta_0} = -\sum_{i\in\mathcal{M}} y_i \tag{4.39}$$

给出。事实上，该算法使用随机梯度下降法(stochastic gradient descent)极小化该分段线性准则。这意味不是计算每个观测的梯度分布和，然后在负梯度方向前进一个步长，而是在访问每个观测之后取步长。因此，按照某种次序访问误分类的观测，并使用下式更新参数 β：

$$\begin{pmatrix}\beta\\ \beta_0\end{pmatrix} \leftarrow \begin{pmatrix}\beta\\ \beta_0\end{pmatrix} + \rho\begin{pmatrix}y_i x_i\\ y_i\end{pmatrix} \tag{4.40}$$

这里，ρ 是学习率，不失一般性，在此情况下可以取 1。如果类是线性可分隔的，可以证明：在有限步之后，算法收敛于一个分离超平面(见习题 4.6)。图 4.13 显示小例子的两个解，每个都开始于一个不同的随机猜测。

该算法有不少问题，总结在 Ripley(1996)中：

- 当数据可分时，存在许多解，并且找到哪个解依赖于初值。
- “有限”步的步数可能很大。间隔越小，所需时间越长。
- 当数据不可分时，算法不收敛，并周期变化。该周期可能很长，因此难以检测。

第二个问题通常可以通过如下方法解决：不是在原空间寻找超平面，而是创建原变量的一些基函数变换，得到扩大的空间，在扩大的空间中寻找超平面。这类似于通过使多项式的次数足够高，将多项式回归问题的残差降低到 0。理想的分割并非总能得到：例如，如果来自两个不同的类共享相同的输入。这可能不是所期望的，因为结果模型可能过分拟合，从而不能很好地泛化。在下一节的末尾，我们将回到该问题。

第一个问题的一个相当优雅的解是对分离超平面做附加的约束。

4.5.2　最佳分离超平面

最佳分离超平面(optimal separating hyperplane)分隔两个类，并极大化到两个类的最近点的距离(Vapnik，1996)。这不仅使分离超平面问题的解惟一，而且通过极大化训练数据上两个类之间的边缘，得到在检验数据上更好的分类性能。

我们需要拓广标准(4.37)。考虑优化问题：

$$\begin{aligned}&\max_{\beta,\beta_0,||\beta||=1} C\\ &\text{受限于 } y_i(x_i^T\beta + \beta_0) \geqslant C,\ i = 1, \cdots, N\end{aligned} \tag{4.41}$$

条件集确保从 β 和 β_0 定义的判定边界到所有点的带符号的距离至少为 C，并且寻找最大的 C 和相关联的参数。可以去掉条件 $\|\beta\| = 1$，用下式取代该条件：

$$\frac{1}{\|\beta\|} y_i(x_i^T\beta + \beta_0) \geqslant C \tag{4.42}$$

（它重新定义 β_0）或等价地：

$$y_i(x_i^T\beta + \beta_0) \geqslant C\|\beta\| \tag{4.43}$$

因为对于任何满足这些不等式的 β 和 β_0，其非负倍数也满足这些不等式，我们可以任意地置 $\|\beta\| = 1/C$。这样，式(4.41)等价于：

$$\begin{gathered}\min_{\beta,\beta_0} \frac{1}{2}\|\beta\|^2 \\ \text{受限于 } y_i(x_i^T\beta + \beta_0) \geqslant 1,\ i = 1, \cdots, N\end{gathered} \tag{4.44}$$

根据式(4.36)，该约束在判定边界周围定义了一个宽度为 $1/\|\beta\|$ 的空隔离带或边缘。因此，我们选取 β 和 β_0 来极大化它的宽度。这是一个凸优化问题（具有线性不等式约束的二次准则）。拉格朗日原始函数是：

$$L_P = \frac{1}{2}\|\beta\|^2 - \sum_{i=1}^{N} \alpha_i[y_i(x_i^T\beta + \beta_0) - 1] \tag{4.45}$$

令导数为 0，可以得到：

$$\beta = \sum_{i=1}^{N} \alpha_i y_i x_i \tag{4.46}$$

$$0 = \sum_{i=1}^{N} \alpha_i y_i \tag{4.47}$$

替换到式(4.45)中，得到 Wolfe 对偶：

$$\begin{gathered}L_D = \sum_{i=1}^{N} \alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{k=1}^{N} \alpha_i\alpha_k y_i y_k x_i^T x_k \\ \text{受限于 } \alpha_i \geqslant 0\end{gathered} \tag{4.48}$$

这个解可以通过在正卦限极大化 L_D 得到。这是一个简单的凸优化问题，可以使用标准软件求解。此外，该解必须满足 Karush-Kuhn-Tucher 条件，它包括式(4.46)、式(4.47)、式(4.48)和

$$\alpha_i[y_i(x_i^T\beta + \beta_0) - 1] = 0\ \forall i \tag{4.49}$$

由此可以看出：

- 如果 $\alpha_i > 0$，则 $y_i(x_i^T\beta + \beta_0) = 1$；换句话说，$x_i$ 在隔离带边界上。
- 如果 $y_i(x_i^T\beta + \beta_0) > 1$，则 x_i 不在隔离带边界上，并且 $\alpha_i = 0$。

由式(4.46)我们看到，解向量 β 用支撑点（support point）x_i 的线性组合定义。这些支撑点通过 $\alpha_i > 0$ 定义为在隔离带的边界上。图 4.15 显示了小例子的最佳超平面，有三个支撑点。同样，通过对任意的支撑点解式(4.49)得到 β_0。

最佳分离超平面产生一个函数 $\hat{f}(x) = x^T\hat{\beta} + \hat{\beta}_0$，用于对新的观测分类：

$$\hat{G}(x) = \text{sign}\hat{f}(x) \tag{4.50}$$

尽管没有一个训练观测落在边缘中(根据构造),但是对于检验观测未必如此。直观地,训练数据上的大边缘将导致对检验数据更好地分隔。

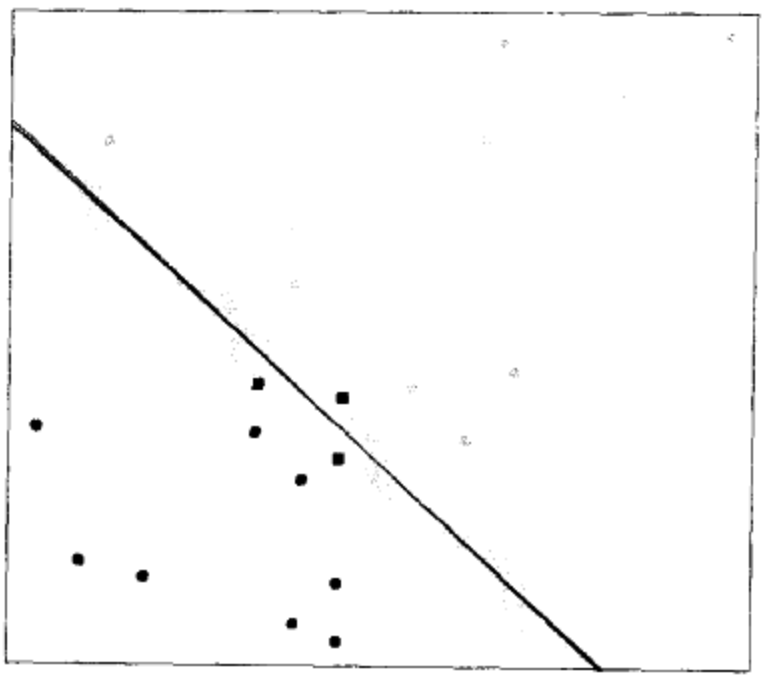


图 4.15　与图 4.13 相同的数据。阴影区域描述分离两个类的最大边缘。有三个支撑点,它们在边缘的边界上,而最佳分离超平面(蓝线)将隔离带一分为二。图中还显示了逻辑斯缔回归找出的边界(红线),它非常接近最佳分离超平面(见第12.3.3节)(见彩页)

用支撑点描述解似乎暗示最佳超平面更关注有价值的点,并且对模型的误描述更健壮。另一方面,LDA 解依赖于所有数据,即便点离判定边界很远。然而,需要注意的是,这些支撑点的识别需要使用所有的数据。当然,如果类的确是高斯的,则 LDA 是最优的,而分离超平面将为关注类边界上的数据(噪声)付出代价。

图 4.15 中给出的是该问题的逻辑斯缔回归解,用极大似然拟合。在该情况下,两个解是类似的。当分离超平面存在时,逻辑斯缔回归总能找到它,因为在此情况下,对数似然总可以降低到 0(见习题 4.5)。逻辑斯缔回归解与分离超平面解还具有一些相同的定性特征。系数向量由输入特征上的零均值线性响应的加权最小二乘方拟合定义,并且靠近判定边界的点的权要比较远的点的权大。

当数据不可分时,问题没有可行解,需要其他形式化方法。我们还可以使用基变换加大空间,但这可能通过过分拟合导致人工的分离。在第 12 章,我们将讨论更吸引人的替代方法,称做支持向量机(support vector machine),它允许重叠,但是极小化重叠的程度。

文献注释

关于分类的好的通用教材包括 Duda 等人(2000)、Hand(1981)、Mclachlan(1992)和 Ripley(1996)的著作。Mardia 等人(1979)给出了线性判别分析的简单扼要的讨论。Michie 等人(1994)在基准数据库上比较了大量流行的分类法。线性分离超平面在 Vapnik(1996)中讨论。我们给出的感知器学习算法基于 Ripley(1996)。

习题

4.1 描述如何通过变换到标准本征值问题来解广义本征值问题：$\max a^T\mathbf{B}a$，条件为 $a^T\mathbf{W}a=1$。

4.2 假定有特征 $x\in\mathrm{IR}^p$，一个2-类响应，类大小分别为 N_1 和 N_2，且编码分别为 $-N/N_1$ 和 N/N_2 的目标。

(a) 证明 LDA 规则将样本分类到类 2，如果

$$x^T\hat{\mathbf{\Sigma}}^{-1}(\hat{\mu}_2-\hat{\mu}_1)>\frac{1}{2}\hat{\mu}_2^T\hat{\mathbf{\Sigma}}^{-1}\hat{\mu}_2-\frac{1}{2}\hat{\mu}_1^T\hat{\mathbf{\Sigma}}^{-1}\hat{\mu}_1+\log(\frac{N_1}{N})-\log(\frac{N_2}{N})$$

否则分类到类 1。

(b) 考虑最小二乘方准则极小化

$$\sum_{i=1}^{N}(y_i-\beta_0-\beta^Tx_i)^2 \tag{4.51}$$

证明(化简后)解 $\hat{\beta}$ 满足

$$\left[(N-2)\hat{\mathbf{\Sigma}}+\frac{N_1N_2}{N}\hat{\mathbf{\Sigma}}_B\right]\beta=N(\hat{\mu}_2-\hat{\mu}_1) \tag{4.52}$$

其中，$\hat{\mathbf{\Sigma}}_B=(\hat{\mu}_2-\hat{\mu}_1)(\hat{\mu}_2-\hat{\mu}_1)^T$。

(c) 由此证明 $\hat{\mathbf{\Sigma}}_B\beta$ 在方向 $(\hat{\mu}_2-\hat{\mu}_1)$ 上，并且

$$\hat{\beta}\propto\hat{\mathbf{\Sigma}}^{-1}(\hat{\mu}_2-\hat{\mu}_1) \tag{4.53}$$

因此，最小二乘方回归系数等于 LDA 系数，相差一个因子。

(d) 证明该结论对于任意(不同的)类编码均成立。

(e) 找出解 $\hat{\beta}_0$，从而预测值 $\hat{f}=\hat{\beta}_0+\hat{\beta}^Tx$。考虑如下规则：如果 $\hat{y}_i>0$，则分类到类 2；否则分类到类 1。证明这与 LDA 规则不同，除非两个类具有相同的观测个数。

(Fisher, 1936; Ripley, 1996)

4.3 假定通过线性回归将原预测子 $\mathbf{X}$ 变换到 $\hat{\mathbf{Y}}$。详细地说，设 $\hat{\mathbf{Y}}=\mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}=\mathbf{X}\hat{\mathbf{B}}$，其中 $\mathbf{Y}$ 是指示响应矩阵。类似地，对于任意输入 $x\in\mathrm{IR}^p$，我们得到变换后的向量 $(\hat{y}=\hat{\mathbf{B}}^Tx\in\mathrm{IR}^K)$。证明：使用 $\hat{\mathbf{Y}}$ 的 LDA 等价于在原空间的 LDA。

4.4 考虑具有 K 个类的多元分对数模型(4.17)。设 β 是 $(p+1)(K-1)$ 向量，包含所有系数。定义输入向量 x 的适合该向量化系数矩阵的一个合适扩展版本。导出极大化多项式对数似然的 Newton-Raphson 算法，并描述你如何实现该算法。

4.5 对于 $x\in\mathrm{IR}$，考虑 2-类逻辑斯缔回归问题。如果两类的样本 x_i 被一个点 $x_0\in\mathrm{IR}$ 分隔，指出斜率和截距参数的极大似然估计的性质。将该结果推广到以下情况：(a) $x\in\mathrm{IR}^p$(见图 4.15)，(b)多于两个类。

4.6 假定在 IR^p 中的一般位置上有 N 个点 x_i，这些点具有类标号 $y_i\in\{-1,1\}$。证明感知器算法在有限步收敛到分离超平面：

(a) 记超平面为 $f(x)=\beta_1^Tx+\beta_0=0$，或更紧凑的形式 $\beta^Tx^*=0$，其中 $x^*=(x,1)$，而

$\beta=(\beta_1,\beta_0)$。设 $z_i=x_i^*/\|x_i^*\|$。证明可分性蕴涵存在 β_{opt} 使得对于任意 i 有 $y_i\beta_{opt}^T z_i \geqslant 1\ \forall i$。

(b) 给定当前的 β_{old}，感知器算法识别出 z_i 被误分类，并产生更新 $\beta_{new}\leftarrow\beta_{old}+y_i z_i$。证明 $\|\beta_{new}-\beta_{opt}\|^2\leqslant\|\beta_{old}-\beta_{opt}\|^2-1$，从而算法在不超过 $\|\beta_{start}-\beta_{opt}\|^2$ 步内收敛到分离超平面(Ripley, 1996)。

4.7 考虑准则：

$$D^*(\beta,\beta_0)=-\sum_{i=1}^{N} y_i(x_i^T\beta+\beta_0) \tag{4.54}$$

即式(4.37)的拓广，我们在所有观测上求和。考虑极小化 D^* 满足 $\|\beta\|=1$。简要描述该准则。它能解决最佳超平面问题吗?

4.8 考虑多元高斯模型 $X\mid G=k\sim N(\mu_k,\mathbf{\Sigma})$，附加的限制为 $\text{rank}\{\mu_k\}_1^K=L<\max(K-1,\ p)$。对于 μ_k 和 $\mathbf{\Sigma}$，推导被约束的 MLE。证明贝叶斯分类规则等价于在约化子空间上用 LDA 分类(Hastie 和 Tibshirani, 1996b)。

4.9 写一个计算机程序，通过拟合每类一个高斯模型进行二次判别分析。在元音数据上试运行它，并计算在检验数据上的误分类率。元音数据可以在本书的网站 www-stat.stanford.edu/ElemStatLearn 上找到。

第5章　基展开与正则化

5.1　引言

我们已经对回归和分类使用了输入特征的线性模型。线性回归、线性判别分析、逻辑斯缔回归和分离超平面都依赖于一个线性模型。实际上，真实函数 $f(X)$ 是 X 的线性函数的情况非常罕见。对于回归问题，通常 $f(X)=\mathrm{E}(Y|X)$ 在 X 上是非线性和非可加的，而用一个线性模型表示通常很方便，并且有时是必要和近似的。方便是因为线性模型容易解释，并且是 $f(X)$ 的一阶泰勒近似。有时必要是因为当 N 很小或 p 很大时，线性模型可能是我们能够用来拟合数据而又不会过分拟合的惟一模型。分类方面也类似，线性的贝叶斯最佳判定边界蕴涵 $\Pr(Y=1|X)$ 的某个单调变换在 X 上是线性的。这显然是一个逼近。

在本章和下一章，我们超越线性限制，讨论一些流行的模型。本章的核心思想是用附加的变量（X 的变换）增广/替换输入向量 X，然后在新的导出的输入特征空间上使用线性模型。

记 $h_m(X):\mathrm{IR}^p\mapsto\mathrm{IR}$ 为 X 的第 m 个变换，$m=1,\cdots,M$。然后，建立 X 的线性基展开（linear basis expansion）模型：

$$f(X)=\sum_{m=1}^{M}\beta_m h_m(X) \tag{5.1}$$

该方法的优点是，一旦确定了基函数 h_m，则模型在这些新变量上是线性的，并且拟合过程与以前一样。

一些简单但广泛使用的 h_m 的例子如下：

- $h_m(X)=X_m$，$m=1,\cdots,p$，恢复原来的线性模型。
- $h_m(X)=X_j^2$ 或 $h_m(X)=X_jX_k$ 允许我们用多项式项增广输入，得到高阶泰勒展开式。注意，变量的个数随多项式的次数指数增长。p 个变量上完全的二次模型需要 $O(p^2)$ 个平方和叉积项；更一般地，对于 d 次多项式需要 $O(p^d)$ 个。
- $h_m(X)=\log(X_j),\sqrt{X_j},\cdots$ 允许单个输入上的其他非线性变换。更一般地，我们可以使用涉及多个输入的类似函数，如 $h_m(X)=\|X\|$。
- $h_m(X)=I(L_m\leqslant X_k<U_m)$，$X_k$ 的区间指示子。通过将 X_k 的区间划分成 M_k 这样不重叠的区间，导致一个 X_k 上逐段常数的模型。

有时，待处理的问题需要特定的基函数 h_m，如对数或幂函数。然而，我们更多地将基展开作为一种装置使用，以便得到 $f(X)$ 的更灵活表示。多项式是后者的一个示例，尽管它受其全局特性的限制：调整系数得到一个区域中的函数形式，可能导致函数在远处剧烈地摆动。本章将讨论一族更有用的分段多项式（piecewise-polynomial）和样条（spline），它们允许局部多项式表

示。还将讨论小波(wavelet)基,它对信号和图像建模特别有用。这些模型产生了一个词典 $\mathcal{D}$,它包含大量基函数,远远超过拟合我们的数据的需要。使用该词典中的基函数,需要一种控制模型复杂度的方法。有三种常用的方法:

- 限制法:在处理之前确定函数类的限制。可加性是一个例子,这里假定模型具有如下形式:

$$\begin{aligned} f(X) &= \sum_{j=1}^{p} f_j(X_j) \\ &= \sum_{j=1}^{p}\sum_{m=1}^{M_j} \beta_{jm} h_{jm}(X_j) \end{aligned} \tag{5.2}$$

  模型的规模被每个分量函数 f_j 所使用的基函数的个数 M_j 所限制。
- 选择法:自适应地扫描词典,只选取那些对模型拟合具有显著贡献的基函数 h_m。这里,第 3 章讨论的变量选择技术是有用的。贪心方法(greedy approache),如 CART、MARS 和提升都可以归入该类。
- 正则化法:我们使用整个词典,但限制系数。岭回归是正则化方法的一个简单例子,而套索既是正则化方法,又是选择方法。下面,讨论这些方法以及更复杂的正则化模型。

5.2　分段多项式和样条

在第 5.7 节之前,我们假定 X 是一维的。一个分段多项式 $f(X)$可以用如下方法得到:将 X 的定义域划分成连续区间,在每个区间用一个多项式表示 f。图 5.1 显示两个简单的分段多项式。第一个是分段常数,有三个基函数:

$$h_1(X) = I(X < \xi_1), \quad h_2(X) = I(\xi_1 \leqslant X < \xi_2), \quad h_3(X) = I(\xi_2 \leqslant X)$$

由于这些函数定义在完全不相交的区间上,所以模型 $f(X) = \sum_{m=1}^{3} \beta_m h_m(X)$ 的最小二乘方估计实际上是 $\hat{\beta}_m = \bar{Y}_m$,即 Y 在第 m 个区间上的均值。

图 5.1 的右上图显示了一个分段线性拟合。需要三个附加的基函数:$h_{m+3} = h_m(X)X$, $m = 1, \cdots, 3$。除特殊情况外,我们一般引用第三幅图(左下图),它也是分段线性的,但在两个纽结上限制为连续的。这些连续性限制导致参数上的线性约束;例如,$f(\xi_1^-) = f(\xi_1^+)$蕴涵 $\beta_1 + \xi_1\beta_4 = \beta_2 + \xi_1\beta_5$。在此情况下,由于有两个限制,我们期望得到两个参数,忽略四个自由参数。

一种处理该情况更直接的方法是使用结合约束的基函数:

$$h_1(X) = 1, \quad h_2(X) = X, \quad h_3(X) = (X - \xi_1)_+, \quad h_4(X) = (X - \xi_2)_+$$

其中 t_+ 表示正的部分。函数 h_3 显示在图 5.1 的右下图。通常,我们偏爱光滑一些的函数,这可以通过提高局部多项式的次数来实现。图 5.2 显示一系列分段三次多项式拟合相同的数据,提高了在纽结的连续阶数。图 5.2 的右下图中的函数是连续的,并且在纽结上具有连续的一、二阶导数。它称为三次样条(cubic spline)。强制更高阶的连续性导致全局立方多项式。不难证明(见习题 5.1)下面的基函数表示一个在ξ_1 和ξ_2 处具有纽结的三次样条:

$$
\begin{aligned}
h_1(X) &= 1, \quad h_3(X) = X^2, \quad h_5(X) = (X-\xi_1)_+^3 \\
h_2(X) &= X, \quad h_4(X) = X^3, \quad h_6(X) = (X-\xi_2)_+^3
\end{aligned} \tag{5.3}
$$

这 6 个基函数对应于函数的 6 维线性空间。快速检查证实参数计数:(3 个区域) × (4 个参数/区域) – (2 个纽结) × (3 个约束/纽结) = 6。

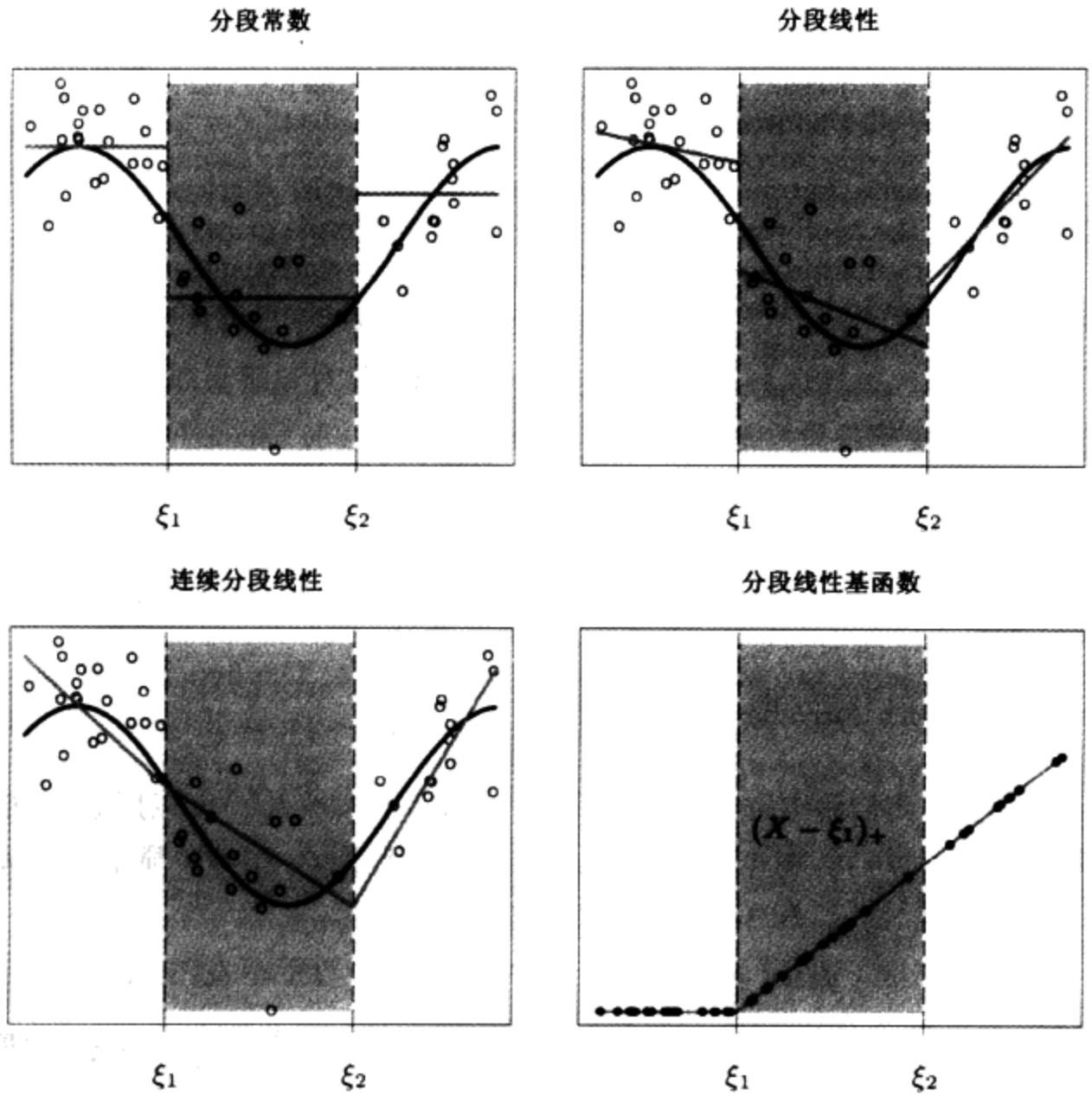


图 5.1 左上图显示分段常数函数拟合人工数据。虚垂线指示两个纽结ξ_1和ξ_2的位置。深色曲线表示真实的函数,由它产生具有高斯噪声的数据。另外两幅图显示分段线性函数拟合相同的数据——右上图没有限制,而左下图限制在纽结上连续。右下图显示分段线性基函 数$h_3(X) = (X-\xi_1)_+$,在ξ_1上连续。黑色的点指示样本的求值$h_3(x_i)$, $i = 1, \cdots, N$

更一般地,一个具有纽结$\xi_j (j = 1, \cdots, K)$的 M 次样条是一个 M 次分段多项式,并具有高达 $M-2$ 阶连续导函数。三次样条有 $M = 3$。事实上,图 5.1 中的分段常数函数是 1 次样条,而连续的分段线性函数是 2 次样条。同样,截尾幂基集的一般形式是:

$$
\begin{aligned}
h_j(X) &= X^{j-1}, \ j = 1, \cdots, M \\
h_{M+\ell}(X) &= (X-\xi_\ell)_+^{M-1}, \ \ell = 1, \cdots, K
\end{aligned}
$$

可以断言,三次样条是肉眼看不出纽结上不连续的最低阶样条。除非对光滑的导函数感兴趣,否则很少有理由需要 3 次以上的样条。在实践中,最广泛使用的样条次数为 $M = 1, 2$ 和 4。

这些纽结固定的样条又称回归样条(regression spline)。我们需要选定样条的次数、纽结数和它们的布局。一种简单的方法是:用基函数的个数或自由度对一族样条参数化,并用观测 x_i 来决定纽结的位置。例如,S-PLUS 中的表达式 bs(x,df = 7)产生三次样条函数基矩阵。这些样条函数在 x 中的 N 个观测上计算,具有 7 - 3 = 4[①] 个内部纽结,在 x 的适当百分位(在此情况下为第 20,40, 60 和 80 个百分位)。然而,可以更直接用 bs(x, degree = 1, knots = c(0.2, 0.4,0.6))产生具有三个内部纽结的线性样条的基,并返回一个 $N \times 4$ 矩阵。

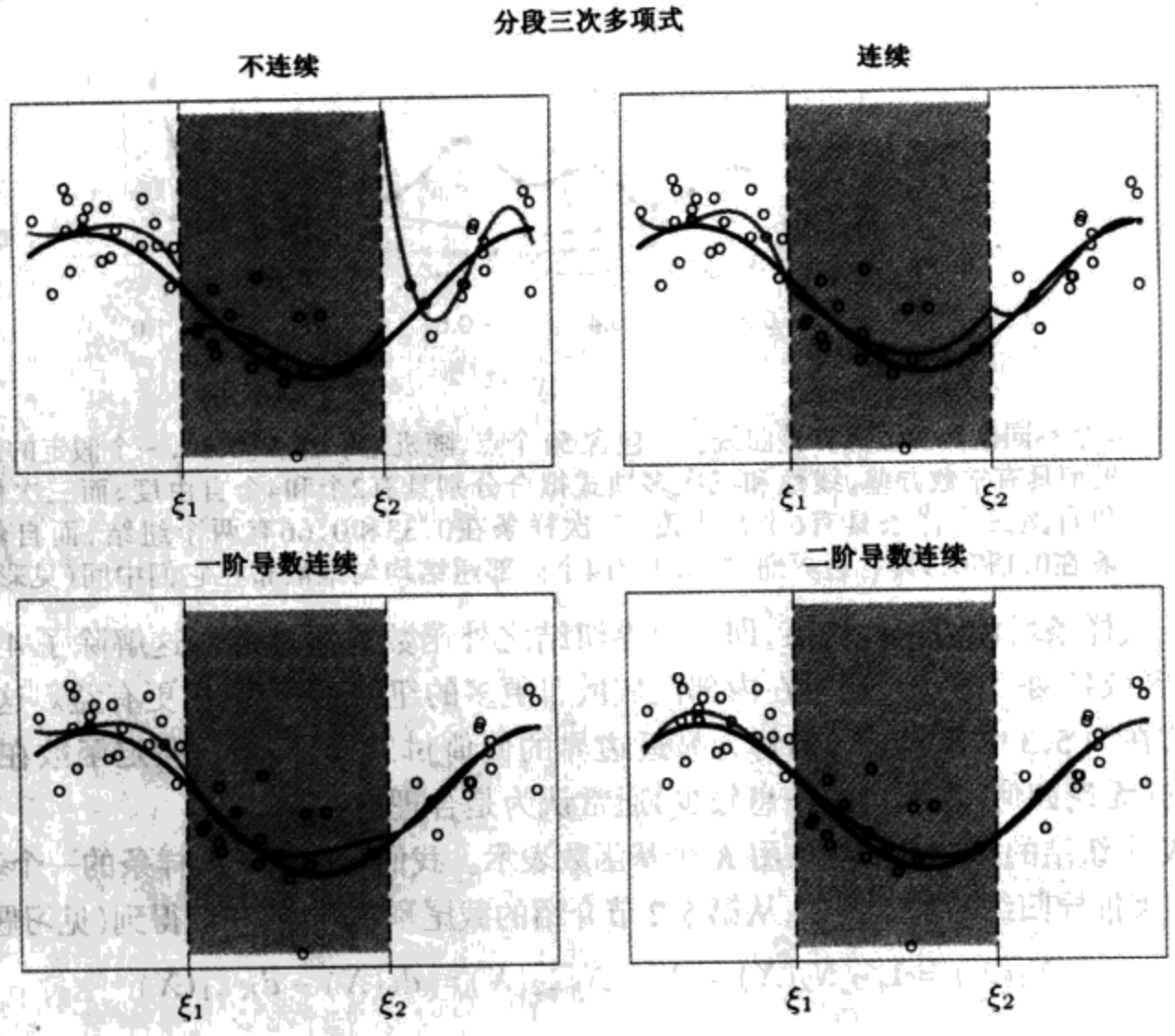


图 5.2　一系列分段三次多项式,具有递增的连续阶

由于特定次数和纽结序列的样条函数空间是一个向量空间,存在一些表示它们的等价基(与一般多项式一样)。虽然截尾幂基在概念上是简单的,但它的数值性质并不太吸引人:大数的幂可能导致严重的舍入问题。本章附录介绍了 B 样条基,即便纽结数 K 很大它也能够有效地计算。

5.2.1　自然三次样条

我们知道,多项式拟合的行为在靠近边界处趋于不稳定, 并且外推可能是危险的。对于样条,这些问题更加严重。在同一区间,与对应的全局多项式相比,越过边界纽结的多项式拟合更加不受控制。这可以方便地用最小二乘方样条函数拟合的逐点方差概述(关于这些方差的计算细节,见下一节的示例)。图 5.3 比较了各种不同模型的逐点方差。靠近边界的方差激

① 一个具有 4 个纽结的三次样条是 8 维的。函数 bs()忽略了基中的常数项, 因为像这样的项通常包含在模型的其他项中。

增是显而易见的，并且对于三次样条，情况最糟糕。

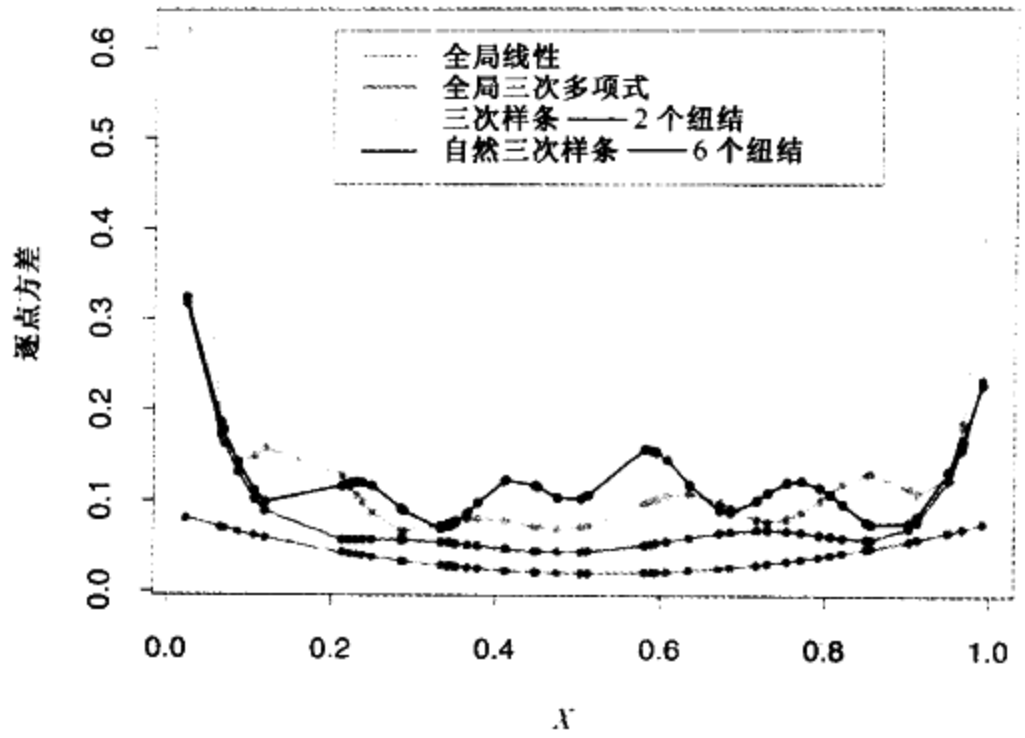


图 5.3 4个不同模型的逐点方差曲线。X 包含 50 个点，随机地取自 $U[0,1]$，一个假定的误差模型具有常数方差。线性和三次多项式拟合分别具有2个和4个自由度，而三次样条和自然三次样条具有6个自由度。三次样条在0.33和0.66有两个纽结，而自然样条在0.1和0.9具有边界纽结，并且有4个内部纽结均匀地散布在它们中间(见彩页)

自然三次样条增加了一些约束，即在边界纽结之外函数是线性的。这解除了 4 个自由度(两个边界区域各两个约束)，通过在内部区域散布更多的纽结，丢弃它们更有益。这种折中以方差的形式在图 5.3 中显示。我们将为靠近边界的偏倚付出代价，但是假定函数在靠近边界处是线性的(无论如何，在那里的信息较少)通常认为是合理的。

具有 K 个纽结的自然三次样条用 K 个基函数表示。我们可以从三次样条的一个基开始，并通过边界约束推导归约的基。例如，从第 5.2 节介绍的截尾幂基开始，我们得到(见习题 5.4)：

$$N_1(X) = 1, \quad N_2(X) = X, \quad N_{k+2}(X) = d_k(X) - d_{K-1}(X) \tag{5.4}$$

其中

$$d_k(X) = \frac{(X-\xi_k)_+^3 - (X-\xi_K)_+^3}{\xi_K - \xi_k} \tag{5.5}$$

可以认为当 $X \geqslant \xi_K$ 时，这些基函数的二、三阶导函数为零。

5.2.2 例：南非心脏病(续)

在第 4.4.2 节，我们用线性逻辑斯缔回归模型拟合南非心脏病数据。这里，使用自然样条考察函数的非线性性。模型的函数形式是：

$$\text{logit}[\Pr(\text{chd}|X)] = \theta_0 + h_1(X_1)^T\theta_1 + h_2(X_2)^T\theta_2 + \cdots + h_p(X_p)^T\theta_p \tag{5.6}$$

其中，每个 θ_j 是系数向量乘以它们相关联的自然样条基函数 h_j 的向量。

对于模型中的每个项，我们使用四个自然样条基。例如，设 X_1 表示 sbp，则 $h_1(X_1)$是由四个基函数组成的基。这实际上蕴涵三个而不是两个内部纽结(在 sbp 的分数位上均匀地选取)，加上在极端数据上的两个边界纽结，因为我们排除了每个 h_j 中的常数项。

由于 famhist 是 2 级因子，我们简单地用一个二元变量或哑变量对它编码，并且将它与模型拟合中的单个系数相关联。

更简洁地，可以把基函数(和常数项)的所有 p 个向量组合在一个大向量 $h(X)$ 中，从而模型简化为 $h(X)^T\theta$，参数的总数为 $\mathrm{df} = 1 + \sum_{j=1}^{p} \mathrm{d}f_j$，即每个分量项中参数的和。每个基函数在 N 个样本上计算，结果是 $N \times \mathrm{df}$ 基矩阵 $\mathbf{H}$。此时，模型与其他线性逻辑斯缔模型一样，并可以使用第 4.4.1 节介绍的算法。

我们执行逐步后向删除过程，从模型中删除项，同时保持每项的组群结构，而不是一次删除一个系数。AIC 统计(见第 7.5 节)用于项删除，并且对于留在最终模型中的所有项，如果从模型中删除将导致 AIC 增加(见表 5.1)。图 5.4 显示由逐步回归选出的最终模型。对于每个变量 X_j，显示的函数是 $\hat{f}_j(X_j) = h_j(X_j)^T\hat{\theta}_j$。协方差矩阵 $\mathrm{Cov}(\hat{\theta}) = \mathbf{\Sigma}$ 用 $\hat{\mathbf{\Sigma}} = (\mathbf{H}^T\mathbf{W}\mathbf{H})^{-1}$ 估计，其中 $\mathbf{W}$ 是取自逻辑斯缔回归的权对角矩阵。因此，$v_j(X_j) = \mathrm{Var}[\hat{f}_j(X_j)] = h_j(X_j)^T\hat{\mathbf{\Sigma}}_j h_j(X_j)$ 是 $\hat{f}_j$ 的逐点方差函数，其中 $\mathrm{Cov}(\hat{\theta}_j) = \hat{\mathbf{\Sigma}}_j$ 是 $\hat{\mathbf{\Sigma}}$ 的适当子矩阵。每个图的阴影区域由 $\hat{f}_j(X_j) \pm 2\sqrt{v_j(X_j)}$ 定义。

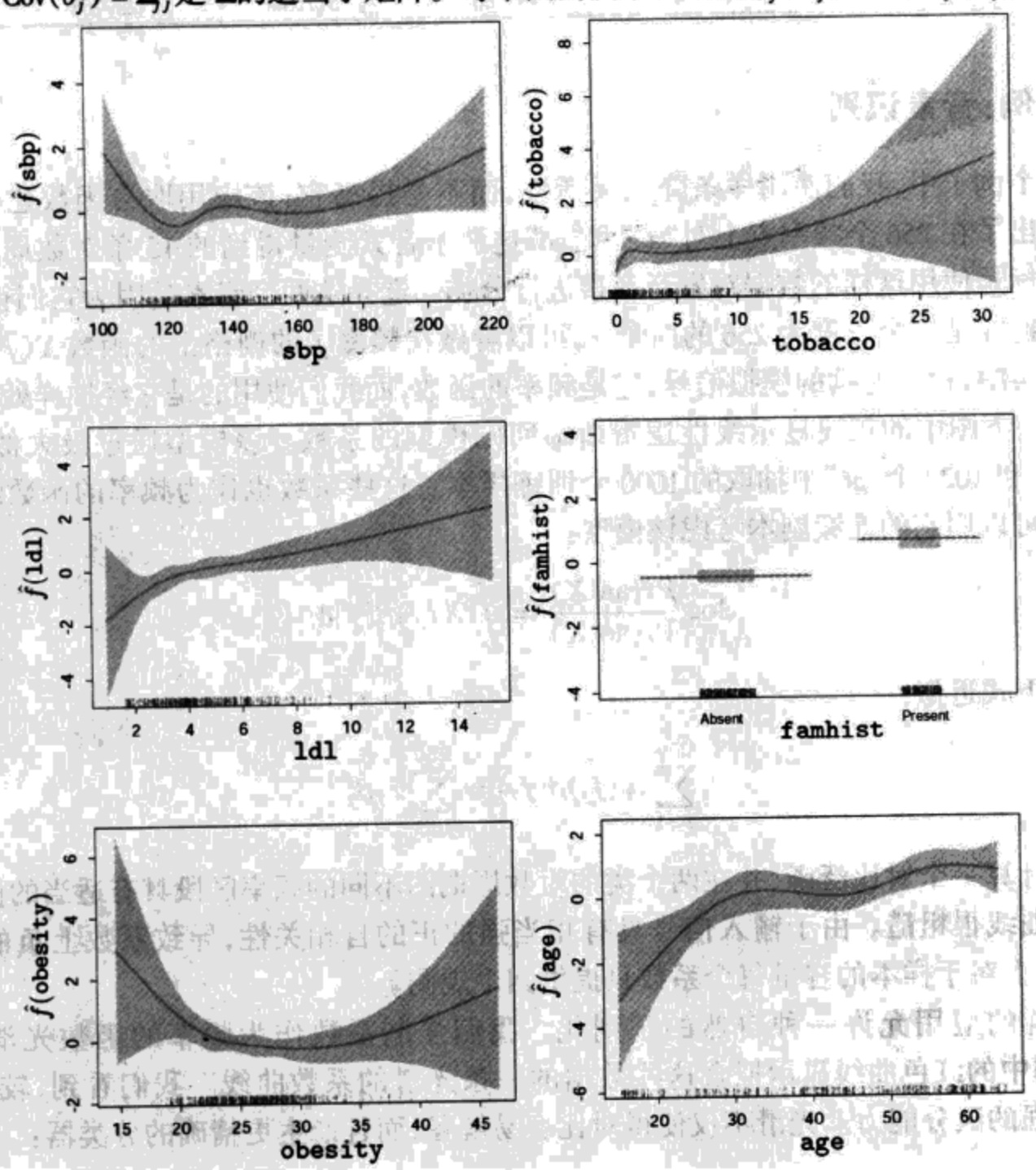


图 5.4　拟合的自然样条函数，最终模型中的每项被逐步过程选取。图中包含逐点标准误差频带。每幅图底部的底线指出该变量的每个样本值的位置

AIC 统计比似然率检验(散离检验)稍微宽松。sbp 和 obesity(肥胖)都包含在该模型中,而它们不在线性模型中。该图说明了原因,因为它们的作用是固有非线性的。开始,这些结果可能有些令人吃惊,但解释就在数据的来源中。有时,这些测量是在病人患心脏病之后才做的,而此时他们多半已经受益于健康的饮食和生活方式,因此表面上看在 sbp 和 obesity 的低值处增加了危险。表 5.1 给出选取的模型的结果汇总。

表 5.1 逐步删除自然样条项后的最终的逻辑斯缔模型。标记为"LRT"的列是当该项从模型中删除时的似然率检验统计,并且是从整个模型(标记为"none")删除时的散离改变

项	Df	散离	AIC	LRT	P 值
none		458.09	502.09		
sbp	4	467.16	503.16	9.076	0.059
botacco	4	470.48	506.48	12.387	0.015
ldl	4	472.39	508.39	14.307	0.006
famhist	1	479.44	521.44	21.356	0.000
obesity	4	466.24	502.24	8.147	0.086
age	4	481.86	517.86	23.768	0.000

5.2.3 例:音素识别

在这个例子中,我们使用样条降低灵活性,而不是提高它;该应用取自函数建模。图 5.5 的上图给出了在 256 个频率点上对"aa"和"ao"这两个音素测量得到的 15 条对数周期分布图。本例的目标是使用这样的数据对发音音素进行分类。选择这两个音素是因为它们很难分辨。

输入特征是一个长度为 256 的向量 x,可以看做在频率 f 的栅格上的函数 $X(f)$ 的求值向量。在现实中,存在连续的模拟信号,它是频率的函数,而我们使用的是它经抽样处理的版本。

图 5.5 下图中的灰线显示线性逻辑斯缔回归模型的系数。该模型通过极大似然拟合从 695 个"aa"和 1022 个"ao"中抽取的 1000 个训练样本。这些系数也作为频率的函数绘制,而事实上我们可以用它的连续副本考虑该模型:

$$\log \frac{\Pr(\text{aa}|X)}{\Pr(\text{ao}|X)} = \int X(f)\beta(f)\mathrm{d}f \tag{5.7}$$

它可以用下式近似:

$$\sum_{j=1}^{256} X(f_j)\beta(f_j) = \sum_{j=1}^{256} x_j\beta_j \tag{5.8}$$

系数计算一个对比泛函,并在两个类的对数周期图不同的频率区段具有适当的值。

灰色曲线很粗糙。由于输入信号具有相当强的正的自相关性,导致系数上负的自相关。此外,实际上对于样本的容量每个系数只提供四个观测。

像这样的应用允许一种自然的正则化。我们强制系数作为频率的函数光滑地变化。图 5.5下图中的红色曲线显示拟合这些数据的一条光滑的系数曲线。我们看到,较低的频率表现出最强的区分能力。光滑不仅使得对比容易解释,而且产生更精确的分类器:

	原始的	正则的
训练误差	0.080	0.185
检验误差	0.255	0.158

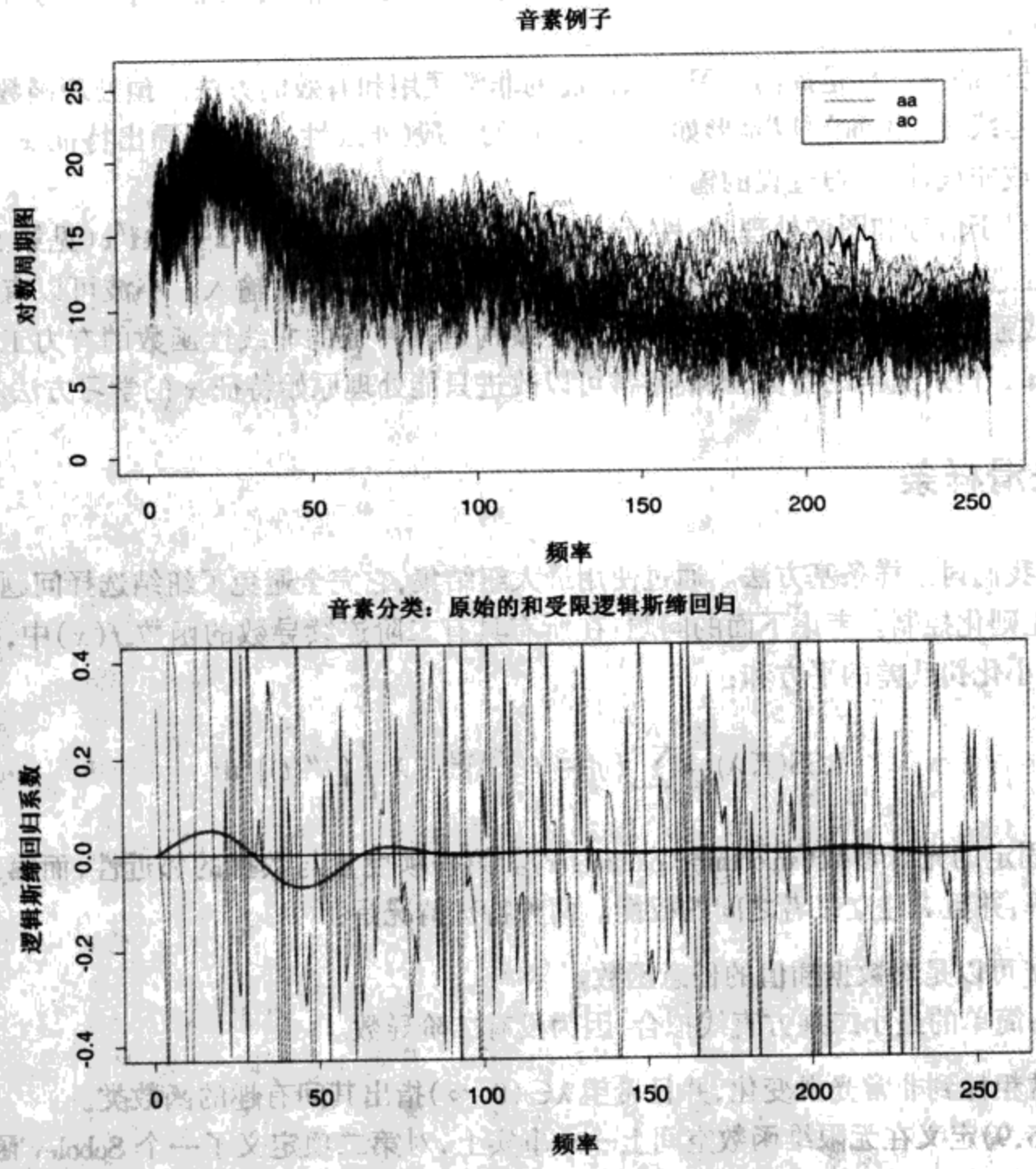


图 5.5　上图显示对数周期图；对于 15 个例子，每个音素“aa”和“ao”从 695 个“aa”和 1022 个“ao”中选样，对数周期图作为频率的函数显示。每个对数周期在256个均匀分布的频率点上测量。下面的图显示逻辑斯缔回归系数(作为频率的函数)，使用256个对数周期图作为输入值，通过极大似然拟合数据。在红色曲线上，限制系数是光滑的，而在锯齿状灰色曲线上没有限制(见彩页)

光滑的红色曲线通过使用一种非常简单的自然三次样条得到。我们可以将系数函数表示成样条展开式 $\beta(f)=\sum_{m=1}^{M} h_m(f)\theta_m$。在实践中，这意味 $\beta=\mathbf{H}\theta$，其中 $\mathbf{H}$ 是自然三次样条的 $p \times M$ 基矩阵，定义在频率集上。这里，我们使用 $M=12$ 个基函数，纽结均匀地分布在代表频率的整数 1，2，…，256 上。由于 $x^T\beta=x^T\mathbf{H}\theta$，我们可以简单地用过滤后的版本 $x^*=\mathbf{H}^Tx$ 替换输入向量 x，并通过 x^* 上的线性逻辑斯缔回归拟合 θ。这样，红色曲线是 $\hat{\beta}(f)=h(f)^T\hat{\theta}$。

5.3 过滤和特征提取

在上一个例子中，我们构造了一个 $p \times M$ 基矩阵 $\mathbf{H}$，并将特征向量 x 变换成新的特征向量 $x^* = \mathbf{H}^T x$。然后，特征的这些过滤后的版本用做学习过程的输入：在上例中，学习过程是线性逻辑斯缔回归。

高维特征的预处理是提高学习算法性能的非常通用和有效的方法。预处理函数不必是线性的（上例是线性的），而可以是形如 $x^* = g(x)$ 的一般（非线性）函数。导出特征 x^* 可以用做任意（线性或非线性）学习过程的输入。

例如，对于信号和图像处理，一种流行的方法是通过小波变换 $x^* = \mathbf{H}^T x$（见第 5.9 节）对原始的特征进行变换，然后使用 x^* 作为神经网络（见第 11 章）的输入。小波可以有效地捕获离散跃变或强势，而神经网络是为预测目标变量构造其特征的非线性函数的有力工具。通过利用领域知识构造适当的特征，我们常常可以改进只能处理原始特征 x 的学习方法。

5.4 光滑样条

这里，我们讨论样条基方法。通过使用最大纽结集，它完全避免了纽结选择问题。拟合的复杂性被正则化控制。考虑下面的问题：在所有具有二阶连续导数的函数 $f(x)$ 中，找出一个函数，它极小化罚残差的平方和：

$$\mathrm{RSS}(f,\lambda) = \sum_{i=1}^{N} \{y_i - f(x_i)\}^2 + \lambda \int \{f''(t)\}^2 \mathrm{d}t \tag{5.9}$$

其中，λ 是固定的光滑参数（smoothing parameter）。第一项度量与数据的邻近性，而第二项惩罚函数的曲率，并且 λ 建立二者之间的权衡。两种特殊情况是：

$\lambda = 0$：f 可以是对数据插值的任意函数。

$\lambda = \infty$：简单的最小二乘方直线拟合，因为没有二阶导数。

这些从非常粗糙到非常光滑变化，并且希望 $\lambda \in (0, \infty)$ 指出其间有趣的函数类。

准则(5.9)定义在无限维函数空间上——事实上，对第二项定义了一个 Sobolev 函数空间。值得注意的是，可以证明式(5.9)具有一个显式的、有限维的、惟一的最小化，它就是自然三次样条，纽结在 $x_i (i = 1, \cdots, N)$ 的惟一值上（见习题 5.7）。看上去这一族方法过于参数化，因为这里有多达 N 个纽结，这意味 N 个自由度。然而，罚项转换成样条系数上的罚，它们向着线性拟合收缩。

由于解是自然样条，可以把它写为：

$$f(x) = \sum_{j=1}^{N} N_j(x)\theta_j \tag{5.10}$$

其中，$N_j(x)$ 是表示该族自然样条的基函数的 N 维集合（见第 5.2.1 节和习题 5.4）。这样，该准则归约成：

$$\mathrm{RSS}(\theta,\lambda)=(\mathbf{y}-\mathbf{N}\theta)^T(\mathbf{y}-\mathbf{N}\theta)+\lambda\theta^T\boldsymbol{\Omega}_N\theta \tag{5.11}$$

其中，$\{\mathbf{N}\}_{ij}=N_j(x_i)$，而 $\{\boldsymbol{\Omega}_N\}_{jk}=\int N_j''(t)N_k''(t)\mathrm{d}t$。容易看出，解是：

$$\hat{\theta}=(\mathbf{N}^T\mathbf{N}+\lambda\boldsymbol{\Omega}_N)^{-1}\mathbf{N}^T\mathbf{y} \tag{5.12}$$

即广义岭回归。拟合的光滑样条由下式给出：

$$\hat{f}(x)=\sum_{j=1}^{N}N_j(x)\hat{\theta}_j \tag{5.13}$$

关于光滑样条的有效计算技术在本章附录中讨论。

图 5.6 显示光滑样条拟合青少年骨质密度(BMD)数据。响应是两次相继检查(通常相隔一年)之间脊骨 BMD 的相对变化。数据按性别着色(深色点代表男性，浅色点代表女性)，并且分别被两条曲线拟合。简单的汇总更清楚地揭示女性的增长比男性大约早两年。在两种情况下，光滑参数 λ 大约为 0.000 22；它的选取在下一节讨论。

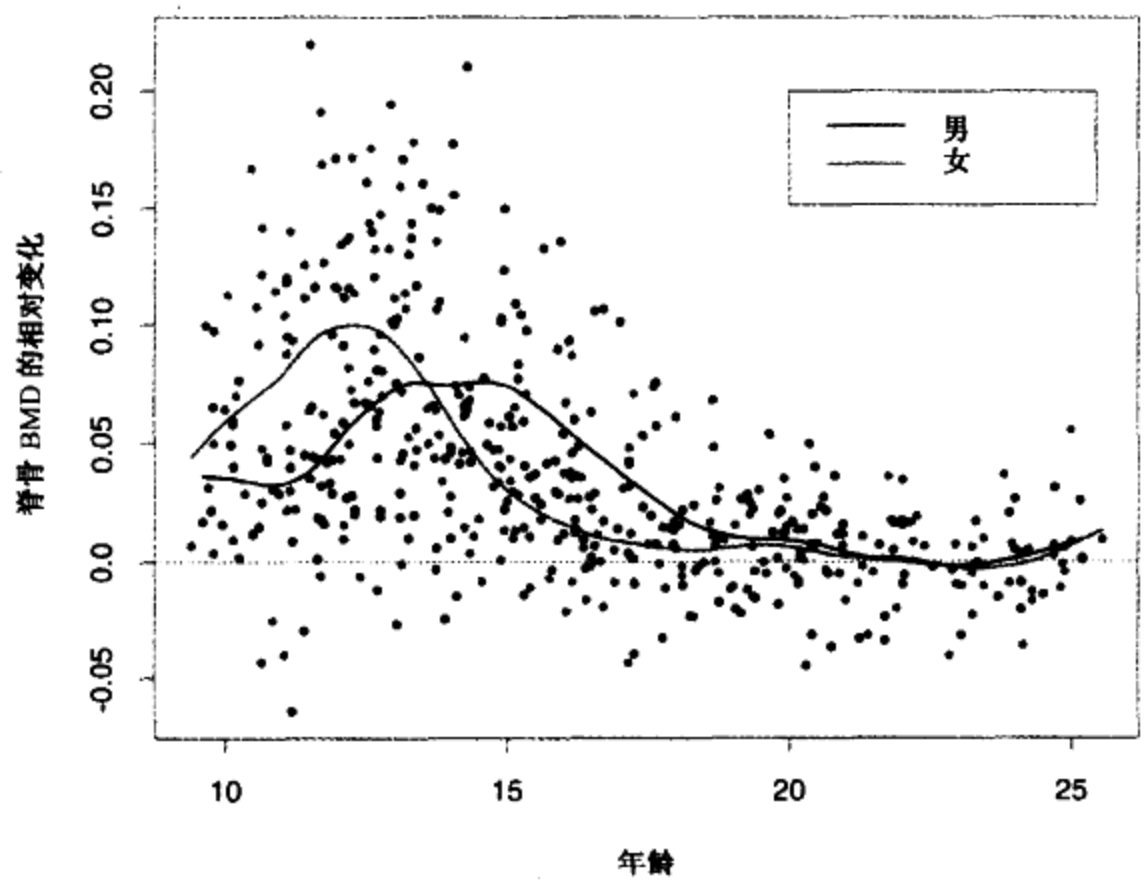


图 5.6　作为年龄的函数，响应是青少年脊骨骨质密度的相对变化。分别用光滑样条拟合男性和女性，λ≈0.000 22；它的选取对应于大约12个自由度

5.4.1　自由度和光滑矩阵

我们还未指出如何为光滑样条选取 λ。在本章稍后，将介绍使用诸如交叉验证等技术的自动方法。本节讨论预先设定光滑量的直观方法。

具有预先选定 λ 的光滑样条是线性光滑子(linear smoother)(与线性算子中一样)的一个例子。这是因为式(5.12)中估计的参数是 y_i 的线性组合。记训练预测子 x_i 上的拟合值 $\hat{f}(x_i)$ 的 N 向量为 $\hat{\mathbf{f}}$，则：

$$\begin{aligned}\hat{\mathbf{f}} &= \mathbf{N}(\mathbf{N}^T\mathbf{N} + \lambda\boldsymbol{\Omega}_N)^{-1}\mathbf{N}^T\mathbf{y} \\ &= \mathbf{S}_\lambda\mathbf{y}\end{aligned} \tag{5.14}$$

拟合又是 $\mathbf{y}$ 上线性的，并且有限线性算子 $\mathbf{S}_\lambda$ 称做光滑子矩阵(smoother matrix)。这种线性性质的一个推论是 $\hat{\mathbf{f}}$ 由 $\mathbf{y}$ 产生并不依赖于 $\mathbf{y}$ 本身，$\mathbf{S}_\lambda$ 仅依赖于 x_i 和 λ。

在更传统的最小二乘方拟合领域，线性算子是众所周知的。设$\mathbf{B}_\xi$是 M 个三次样条基函数的 $N \times M$ 矩阵，在 N 个训练点 x_i 上求值，具有纽结序列ξ，而 $M << N$。则拟合样条值向量由下式给出：

$$\begin{aligned}\hat{\mathbf{f}} &= \mathbf{B}_\xi(\mathbf{B}_\xi^T\mathbf{B}_\xi)^{-1}\mathbf{B}_\xi^T\mathbf{y} \\ &= \mathbf{H}_\xi\mathbf{y}\end{aligned} \tag{5.15}$$

这里，线性算子$\mathbf{H}_\xi$是一个投影算子，在统计学也称帽矩阵(hat matrix)。在$\mathbf{H}_\xi$和 $\mathbf{S}_\lambda$ 之间有一些重要的类似和差异：

- 它们都是对称的、半正定的矩阵。
- $\mathbf{H}_\xi\mathbf{H}_\xi = \mathbf{H}_\xi$(幂等的)，而 $\mathbf{S}_\lambda\mathbf{S}_\lambda \preceq \mathbf{S}_\lambda$，意味着通过一个半正定矩阵，右部超过左部。这是 $\mathbf{S}_\lambda$ 收缩特性的一个结果，我们将在下面进一步讨论。
- $\mathbf{H}_\xi$的秩为 M，而 $\mathbf{S}_\lambda$ 的秩为 N。

表达式 $M = \text{trace}(\mathbf{H}_\xi)$给出投影空间的维数，它也是基函数的个数，因而是拟合涉及的参数的个数。根据类推，我们定义光滑样条的有效自由度(effective degrees of freedom)为

$$\text{df}_\lambda = \text{trace}(\mathbf{S}_\lambda) \tag{5.16}$$

$\mathbf{S}_\lambda$ 的对角线元素之和。这个非常有用的定义使得我们可以用更直观的方法对光滑样条参数化，并且还可以用一致的方式对其他一些光滑法参数化。例如，在图 5.6 中，对每条曲线指定 $\text{df}_\lambda = 12$，并且通过解 $\text{trace}(\mathbf{S}_\lambda) = 12$ 推出对应的 $\lambda \approx 0.000\,22$。有许多理由支持这种自由度定义，这里将讨论其中的一些。

由于 $\mathbf{S}_\lambda$ 是对称的(和半正定的)，它具有本征分解。在继续讨论之前，将 $\mathbf{S}_\lambda$ 写成 Reinsch 形式是方便的

$$\mathbf{S}_\lambda = (\mathbf{I} + \lambda\mathbf{K})^{-1} \tag{5.17}$$

其中，$\mathbf{K}$ 不依赖于 λ(见习题 5.9)。由于 $\hat{\mathbf{f}} = \mathbf{S}_\lambda\mathbf{y}$，解

$$\min_{\mathbf{f}}(\mathbf{y} - \mathbf{f})^T(\mathbf{y} - \mathbf{f}) + \lambda\mathbf{f}^T\mathbf{K}\mathbf{f} \tag{5.18}$$

$\mathbf{K}$ 被视为罚矩阵(penalty matrix)，而 $\mathbf{K}$ 中的二次形式确实具有平方二次差分的加权和表示。$\mathbf{S}_\lambda$ 的本征分解是：

$$\mathbf{S}_\lambda = \sum_{k=1}^{N}\rho_k(\lambda)\mathbf{u}_k\mathbf{u}_k^T \tag{5.19}$$

而

$$\rho_k(\lambda) = \frac{1}{1 + \lambda d_k} \tag{5.20}$$

并且 d_k 是 $\mathbf{K}$ 的对应本征值。

图 5.7(上图)显示将三次光滑样条用于某空气污染数据(128 个观测)的结果。给出了两个拟合:较光滑的拟合对应于较大的罚 λ,而较粗糙的拟合对应于较小的罚。下图给出本征值(左下)和对应的光滑子矩阵的某些本征向量(右下)。本征表示的一些要点如下:

- 本征向量不受 λ 变化的影响,因而被 λ 索引的整个光滑样条族(对于一个特定的序列 $\mathbf{x}$)具有相同的本征向量。
- $\mathbf{S}_\lambda \mathbf{y} = \sum_{k=1}^{N} \mathbf{u}_k \rho_k(\lambda) \langle \mathbf{u}_k, \mathbf{y} \rangle$ 从而光滑样条通过关于(整个)基 $\{\mathbf{u}_k\}$ 分解 $\mathbf{y}$,并使用 $\rho_k(\lambda)$ 微分地收缩贡献来进行操作。这与基回归方法形成明显差异。基回归方法的分量或者留下,或者收缩到 0——像上面 $\mathbf{H}_\xi$ 这样的投影矩阵有 M 个本征值等于 1,而其他为 0。因此,光滑样条称做收缩(shrinking)光滑法,而回归样条是投影(projection)光滑法(见图 3.10)。
- $\mathbf{u}_k$ 的序列,按 $\rho_k(\lambda)$ 的递减排列,看来增加了复杂度。确实,它们具有递增次数多项式的零交叉(zero-crossing)行为。由于 $\mathbf{S}_\lambda \mathbf{u}_k = \rho_k(\lambda)\mathbf{u}_k$,我们看每个本征向量本身如何被光滑样条收缩:复杂度越高,收缩越多。如果 X 的定义域是周期的,则 $\mathbf{u}_k$ 是不同频率上的正弦和余弦函数。
- 前两个本征值总是 1,并且它们对应于 x 上线性函数的二维本征空间(见习题 5.11),永远不被收缩。
- 本征值 $\rho_k(\lambda) = 1/(1+\lambda d_k)$ 是罚矩阵 $\mathbf{K}$ 的本征值 d_k 的逆函数,被 λ 调节;λ 控制 $\rho_k(\lambda)$ 递减到 0 的速率。$d_1 = d_2 = 0$ 并且线性函数没有罚。
- 可以使用基向量 $\mathbf{u}_k$(Demmler-Reinsch 基)对光滑样条重新参数化。在此情况下,光滑样条解决:

$$\min_{\boldsymbol{\theta}} \|\mathbf{y} - \mathbf{U}\boldsymbol{\theta}\|^2 + \lambda \boldsymbol{\theta}^T \mathbf{D} \boldsymbol{\theta} \tag{5.21}$$

其中,$\mathbf{U}$ 具有列 $\mathbf{u}_k$,而 $\mathbf{D}$ 是具有元素 d_k 的对角矩阵。

- $\mathrm{df}_\lambda = \mathrm{trace}(\mathbf{S}_\lambda) = \sum_{k=1}^{N} \rho_k(\lambda)$。对于投影光滑,所有本征值为 1,每个对应投影子空间的一个维。

图 5.8 显示了一个光滑样条矩阵,行按 x 排序。这种表示的带状特点暗示光滑样条是一种局部拟合方法,很像第 6 章的局部加权回归过程。其中的右图详细显示了 $\mathbf{S}$ 的选定行,称做等价核(equivalent kernel)。随 $\lambda \to 0$,$\mathrm{df}_\lambda \to N$,并且 $\mathbf{S}_\lambda \to \mathbf{I}$,即 N 维恒等矩阵。随 $\lambda \to \infty$,$\mathrm{df}_\lambda \to 2$,并且 $\mathbf{S}_\lambda \to \mathbf{H}$,即 $\mathbf{x}$ 上线性回归的帽矩阵。

5.5　光滑参数的自动选择

回归样条的光滑参数包括样条的次数、纽结个数和位置。对于光滑样条,我们只有罚参数 λ 需要选择,因为纽结在所有训练 X 上,并且在实践中总是使用三次样条。

对于回归样条,除非强制进行某些简化,否则选择纽结的位置和个数可能是组合复杂的。

第9章的MARS过程使用具有某种近似的贪心算法，实现一种实际的折中。这里不做深入讨论。

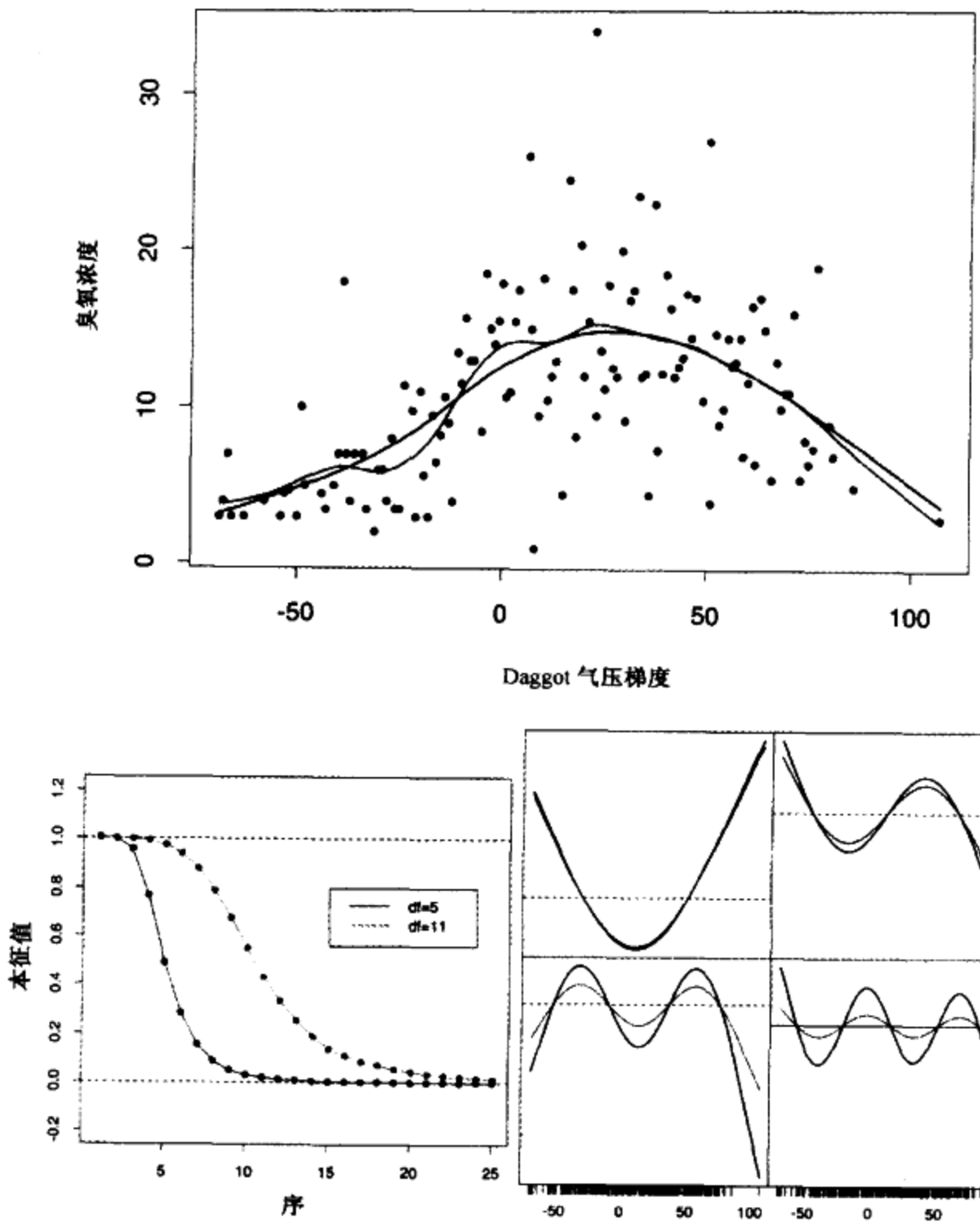


图5.7 [上图]臭氧浓度作为Daggot气压梯度的函数的光滑样条拟合。两个拟合对应于光滑参数的不同值。光滑参数的选取是为了得到5个和11个由$df_\lambda = \mathrm{trace}(\mathbf{S}_\lambda)$定义的有效自由度。[左下图]两个光滑样条矩阵的前25个本征值。前两个本征值恰为1，并且所有的本征值都大于或等于零。[右下图]样条光滑子矩阵的第三个和第六个本征向量。在每条曲线中，$\mathbf{u}_k$都对照x绘制，并因此视为x的函数。图底部的底线指示数据点的出现。阻尼函数表示这些函数的光滑版本(使用5df光滑子)(见彩页)

5.5.1 固定自由度

对于光滑样条，由于$df_\lambda = \mathrm{trace}(\mathbf{S}_\lambda)$是$\lambda$上单调的，我们可以反转该联系，并通过固定df来确定λ。在实践中，这可以通过简单的数值方法实现。例如，在S-PLUS中，可以使用smooth.spline(x,y,df=6)指定光滑量。这促使采用更传统的模型选择方法：试验多个不同的df

值,并根据近似的 F-检验、残差图或其他更主观的标准选择一个。以这种方法使用 df 提供了一种比较一些不同光滑方法的一致方法。这在广义加法模型(见第 9 章)中特别有用。在广义加法模型中,多种光滑方法可以同时使用。

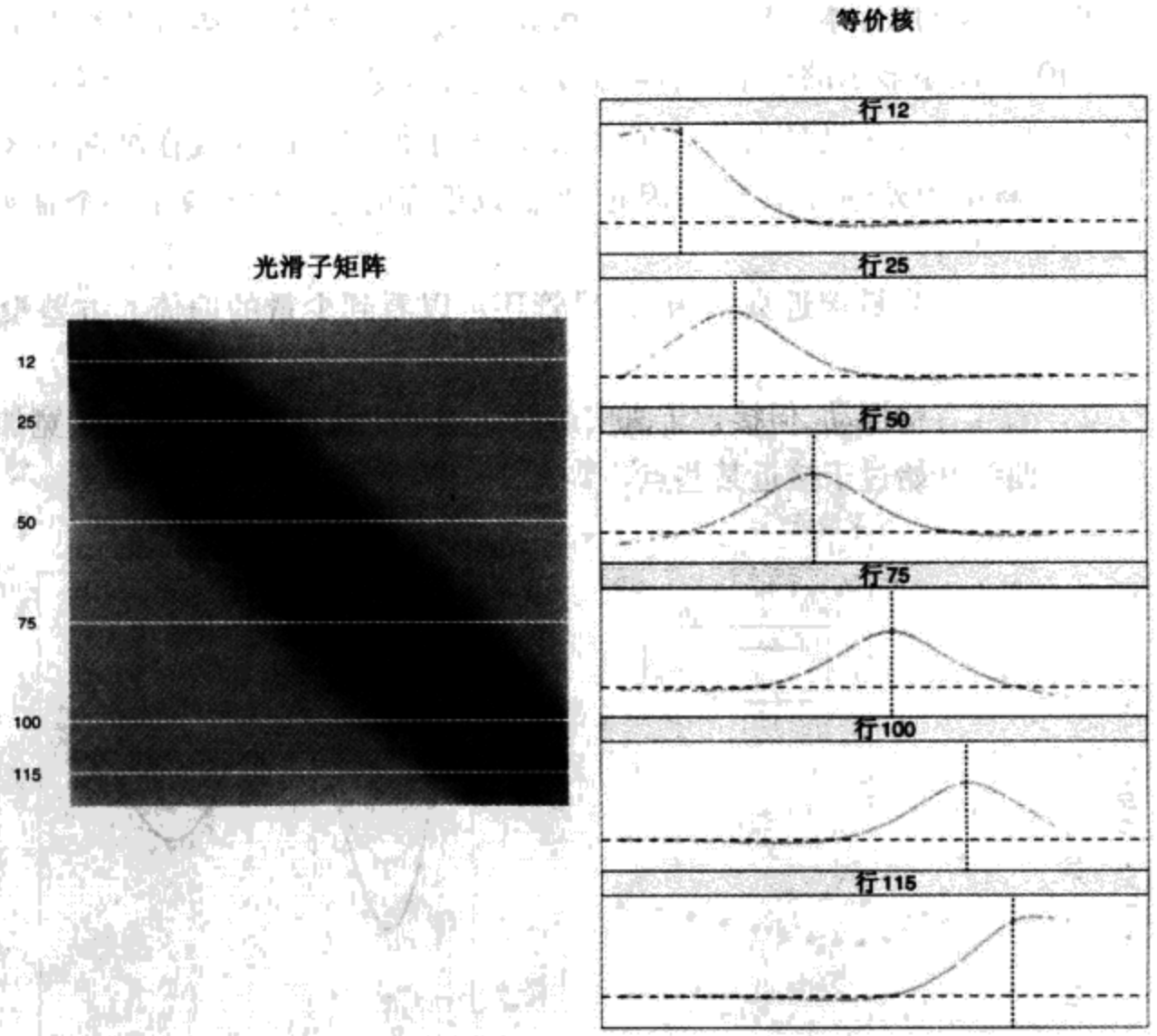


图 5.8　光滑样条的光滑矩阵是近似带状的,预示具有局部支集的等价核。左图将S的元素表示成图像。右图详细显示指定行的等价核或权函数

5.5.2　偏倚 – 方差权衡

图 5.9 显示,在下面的简单例子上使用光滑样条时选择 df_λ 的影响:

$$\begin{aligned} Y &= f(X) + \varepsilon \\ f(X) &= \frac{\sin(12(X+0.2))}{X+0.2} \end{aligned} \tag{5.22}$$

其中,$X \sim U[0, 1]$,而 $\varepsilon \sim N(0, 1)$。训练样本包含 $N = 100$ 个 x_i 和 y_i 对,独立地从该模型抽取。

对于三个不同的 df_λ 值,图中显示了拟合样条。图中的黄色阴影区域表示 $\hat{f}_\lambda$ 的逐点标准差;即,我们对 $\hat{f}_\lambda(x) \pm 2 \cdot \mathrm{se}(\hat{f}_\lambda(x))$之间的区域加了阴影。由于 $\hat{\mathbf{f}} = \mathbf{S}_\lambda \mathbf{y}$,

$$\begin{aligned} \mathrm{Cov}(\hat{\mathbf{f}}) &= \mathbf{S}_\lambda \mathrm{Cov}(\mathbf{y}) \mathbf{S}_\lambda^T \\ &= \mathbf{S}_\lambda \mathbf{S}_\lambda^T \end{aligned} \tag{5.23}$$

对角线元素包含训练数据 x_i 上的逐点方差。偏倚由下式给出:

$$\begin{aligned} \text{Bias}(\hat{\mathbf{f}}) &= \mathbf{f} - \text{E}(\hat{\mathbf{f}}) \\ &= \mathbf{f} - \mathbf{S}_\lambda \mathbf{f} \end{aligned} \tag{5.24}$$

其中，$\mathbf{f}$ 是真实的 f 在训练数据 X 上的求值(未知)向量。期望和方差都是关于从模型(5.22)中重复抽取的容量 $N=100$ 的样本。$\text{Var}(\hat{f}_\lambda(x_0))$和 $\text{Bias}(\hat{f}_\lambda(x_0))$可以用类似的方式在任意点 x_0 计算(见习题 5.10)。图中显示的三个拟合给出关于选定参数的偏倚 - 方差权衡的图解。

$\text{df}_\lambda = 5$：样条拟合不足，并且显然裁减了高峰，填充了低谷。这导致在高曲率区域偏倚非常大。标准误差频带非常窄，因此，我们以很高的可靠性形成了一个真实函数的很大偏倚的估计！

$\text{df}_\lambda = 9$：这里，拟合函数最接近真实函数，尽管还可以看到少量的偏倚。方差没有明显的增加。

$\text{df}_\lambda = 15$：拟合函数有些摆动，但接近于真实函数。摆动也是造成标准误差带宽加宽的原因——曲线开始过于接近某些点。

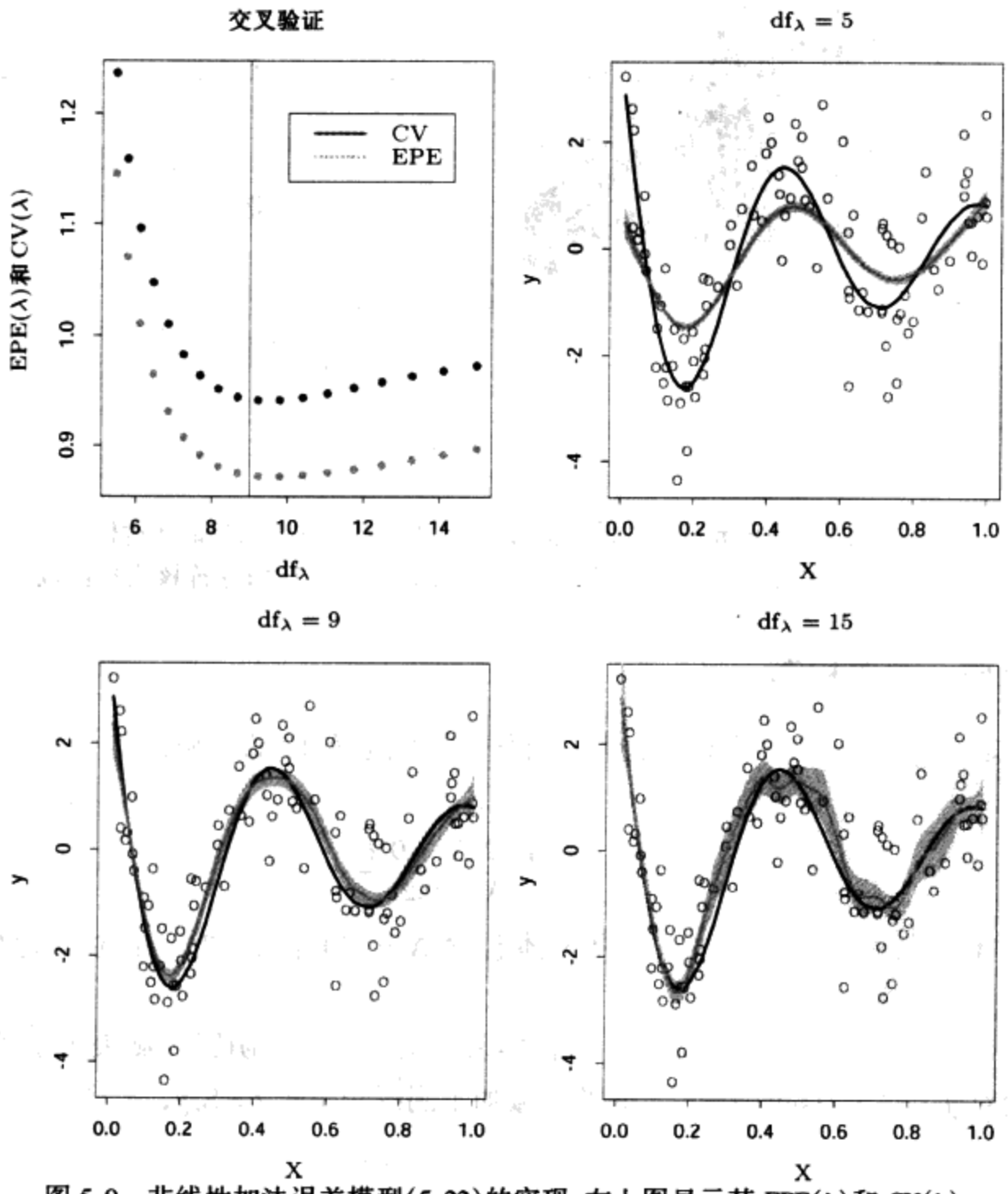


图 5.9 非线性加法误差模型(5.22)的实现，左上图显示其 EPE(λ)和 CV(λ)曲线。其他图对于不同的 df_λ，显示数据、真实函数(紫色)和拟合曲线(绿色)，其中黄色阴影是拟合曲线 ± 2倍标准误差频带(见彩页)

注意，在这些图中，我们观察数据的单个实现和每种情况的拟合样条，而偏倚涉及期望 $\mathrm{E}(\hat{f})$。我们将计算类似曲线的问题留做习题(见习题 5.10)，那里还要显示偏倚。中间的曲线看上去“恰好”，它实现了偏倚和方差之间很好的折中。

综合的平方预测误差(EPE)将偏倚和方差组合在一个公式中：

$$\begin{aligned}\mathrm{EPE}(\hat{f}_\lambda) &= \mathrm{E}(Y-\hat{f}_\lambda(X))^2 \\ &= \mathrm{Var}(Y)+\mathrm{E}\left[\mathrm{Bias}^2(\hat{f}_\lambda(X))+\mathrm{Var}(\hat{f}_\lambda(X))\right] \\ &= \sigma^2+\mathrm{MSE}(\hat{f}_\lambda)\end{aligned} \tag{5.25}$$

注意，这是训练样本(引出 $\hat{f}_\lambda$)和(独立选取的)预测点 (X,Y) 的值上的平均。EPE 是一个有趣的自然量，它确实建立了偏倚和方差之间的平衡。图 5.9 左上图的蓝点表明 $\mathrm{df}_\lambda=9$ 是恰当的位置。

由于我们不知道真实的函数，从而没有 EPE 可用，因而需要一个估计。该问题将在第 7 章中更详细地讨论，而诸如 K 折交叉验证、GCV 和 C_p 等技术都是常用的。在图 5.9 中，我们给出了 N 折(留一)交叉验证曲线：

$$\mathrm{CV}(\hat{f}_\lambda) = \sum_{i=1}^{N}(y_i-\hat{f}_\lambda^{(-i)}(x_i))^2 \tag{5.26}$$

$$= \sum_{i=1}^{N}\left(\frac{y_i-\hat{f}_\lambda(x_i)}{1-S_\lambda(i,i)}\right)^2 \tag{5.27}$$

对于每个 λ 值，可以通过原拟合值和 $\mathbf{S}_\lambda$ 的对角线元素 $S_\lambda(i,i)$ 计算(见习题 5.13)。

EPE 和 CV 曲线具有类似的形状，但是，整条 CV 曲线在 EPE 曲线的上方。对于某些实现正好相反，并且作为 EPE 曲线的估计，整条 CV 曲线是近似无偏的。

5.6　无参逻辑斯缔回归

第 5.4 节的光滑样条问题(5.9)是以回归形式提出的。将该技术转移到其他领域通常是直接的。这里，我们考虑具有单个量化输入变量 X 的逻辑斯缔回归。模型是：

$$\log\frac{\Pr(Y=1|X=x)}{\Pr(Y=0|X=x)}=f(x) \tag{5.28}$$

蕴涵：

$$\Pr(Y=1|X=x)=\frac{\mathrm{e}^{f(x)}}{1+\mathrm{e}^{f(x)}} \tag{5.29}$$

光滑地拟合 $f(x)$ 导致条件概率 $\Pr(Y=1|x)$ 的一个光滑估计，这可以用于分类和风险评估。

构造罚对数似然准则：

$$\begin{aligned}\ell(f;\lambda) &= \sum_{i=1}^{N}[y_i\log p(x_i)+(1-y_i)\log(1-p(x_i))]-\frac{1}{2}\lambda\int\{f''(t)\}^2\mathrm{d}t \\ &= \sum_{i=1}^{N}\left[y_i f(x_i)-\log(1+\mathrm{e}^{f(x_i)})\right]-\frac{1}{2}\lambda\int\{f''(t)\}^2\mathrm{d}t\end{aligned} \tag{5.30}$$

其中，缩写 $p(x) = \Pr(Y = 1 \mid x)$。该表达式的第一项是基于二项分布的对数似然（参见第 4 章）。类似于第 5.4 节的论据表明最佳的 f 是有穷维上的自然样条，纽结在 x 的惟一值上。这意味我们有 $f(x) = \sum_{j=1}^{N} N_j(x)\theta_j$。计算一、二阶导数：

$$\frac{\partial \ell(\theta)}{\partial \theta} = \mathbf{N}^T(\mathbf{y} - \mathbf{p}) - \lambda \mathbf{\Omega}\theta \tag{5.31}$$

$$\frac{\partial^2 \ell(\theta)}{\partial \theta \partial \theta^T} = -\mathbf{N}^T\mathbf{W}\mathbf{N} - \lambda \mathbf{\Omega} \tag{5.32}$$

其中，$\mathbf{p}$ 是 N 向量，具有元素 $p(x_i)$，而 $\mathbf{W}$ 是权 $p(x_i)(1 - p(x_i))$ 的对角矩阵。一阶导数(5.31)在 θ 上是非线性的，因此需要使用类似于第 4.4.1 节的迭代算法。使用如式(4.23)的 Newton-Raphson 和线性逻辑斯缔回归(4.24)，更新方程可以写为：

$$\begin{aligned} \theta^{\text{new}} &= (\mathbf{N}^T\mathbf{W}\mathbf{N} + \lambda\mathbf{\Omega})^{-1}\mathbf{N}^T\mathbf{W}\left(\mathbf{N}\theta^{\text{old}} + \mathbf{W}^{-1}(\mathbf{y} - \mathbf{p})\right) \\ &= (\mathbf{N}^T\mathbf{W}\mathbf{N} + \lambda\mathbf{\Omega})^{-1}\mathbf{N}^T\mathbf{W}\mathbf{z} \end{aligned} \tag{5.33}$$

也可以用拟合值表示该更新：

$$\begin{aligned} \mathbf{f}^{\text{new}} &= \mathbf{N}(\mathbf{N}^T\mathbf{W}\mathbf{N} + \lambda\mathbf{\Omega})^{-1}\mathbf{N}^T\mathbf{W}\left(\mathbf{f}^{\text{old}} + \mathbf{W}^{-1}(\mathbf{y} - \mathbf{p})\right) \\ &= \mathbf{S}_{\lambda,w}\mathbf{z} \end{aligned} \tag{5.34}$$

参照式(5.12)和式(5.14)，我们看到更新是用加权光滑样条拟合响应 $\mathbf{z}$（见习题 5.12）。

式(5.34)的形式具有启发性。它试图用任意无参（加权的）回归算子替换 $\mathbf{S}_{\lambda,w}$，并得到一族一般的无参逻辑斯缔回归模型。尽管这里的 x 是一维的，但该过程可以自然地拓广到高维 x。这些扩充是广义加法模型的核心，我们将在第 9 章讨论。

5.7 多维样条函数

迄今为止，我们一直关注一维样条函数模型。每种方法都有类似的多维方法。假定 $X \in \mathbb{R}^2$，有表示坐标 X_1 的函数基 $h_{1k}(X_1)$，$k = 1, \cdots, M_1$，并类似地对坐标 X_2 有 M_2 个函数 $h_{2k}(X_2)$ 的集合。则由

$$g_{jk}(X) = h_{1j}(X_1)h_{2k}(X_2),\ j = 1, \cdots, M_1,\ k = 1, \cdots, M_2 \tag{5.35}$$

定义的 $M_1 \times M_2$ 维张量积基（tensor product basis）可以用来表示二维函数：

$$g(X) = \sum_{j=1}^{M_1}\sum_{k=1}^{M_2}\theta_{jk}g_{jk}(X) \tag{5.36}$$

图 5.10 显示使用 B 样条的张量积基。和以前一样，系数可以用最小二乘方拟合。这可以拓广到 d 维，但注意基的维数将呈指数增长——又出现维灾难。第 9 章讨论的 MARS 过程是一种贪心算法，只包含那些最小二乘方必需的张量积。

图 5.11 显示了在取自第 2 章的模拟分类例子上，加法和张量积（自然）样条的差别。逻辑斯缔模型 $\text{logit}[\Pr(T \mid x)] = h(x)^T\theta$ 是对二元响应的拟合，并且估计判定边界是围线 $h(x)^T\hat{\theta} = 0$。张量积基可以在判定边界上获得更大的灵活性，但引进了一些似是而非的结构。

一维光滑样条也能（通过正则化）拓广到高维。假定有 y_i 和 x_i 对，$x_i \in \mathbb{R}^d$，并且我们寻求

d 维回归函数$f(x)$。这种想法建立问题：

$$\min_f \sum_{i=1}^{N} \{y_i - f(x_i)\}^2 + \lambda J[f] \tag{5.37}$$

其中，J 是稳定 $\mathbb{R}^d$ 上函数f 的罚泛函。例如，对于 $\mathbb{R}^2$ 上的函数，一维粗糙度罚(5.9)的一个自然拓广是：

$$J[f] = \int\int_{\mathbb{R}^2} \left[\left(\frac{\partial^2 f(x)}{\partial x_1^2} \right)^2 + 2 \left(\frac{\partial^2 f(x)}{\partial x_1 \partial x_2} \right)^2 + \left(\frac{\partial^2 f(x)}{\partial x_2^2} \right)^2 \right] \mathrm{d}x_1 \mathrm{d}x_2 \tag{5.38}$$

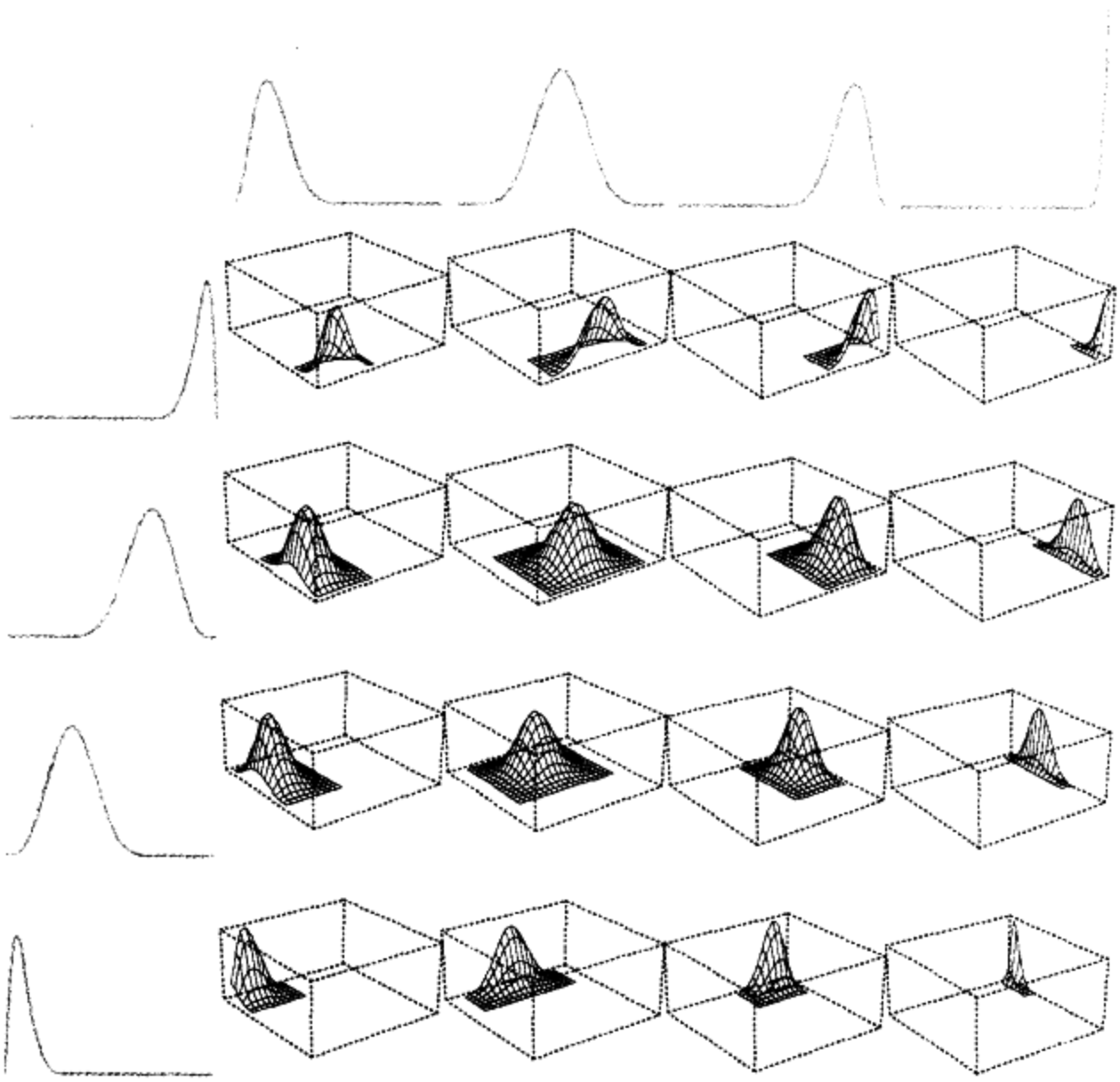


图 5.10　B 样条的张量积基，显示某些选定的对。每个二维函数是对应的一维边缘函数的张量积

利用该罚优化(5.37)，产生一个光滑的二维曲面，称为薄板样条。它与一维三次光滑样条有一些相同的性质：

- 随 $\lambda \to 0$，解趋向于一个插值函数[具有最小罚(5.38)的插值函数]；
- 随 $\lambda \to \infty$，解趋向于最小二乘方平面；
- 对于 λ 的中间值，解可以用基函数的线性展开式表示，其系数可以通过一种广义岭回归形式得到。

解形如：

$$f(x) = \beta_0 + \beta^T x + \sum_{j=1}^{N} \alpha_j h_j(x) \tag{5.39}$$

其中，$h_j(x) = \eta(\| x - x_j \|)$，而 $\eta(z) = z^2 \log z^2$。这些 h_j 是径向基函数(radial basis function)的例子，将在下一节更详细地讨论。通过将式(5.39)插入到式(5.37)中求出系数，将问题归结为有限维罚最小二乘方问题。对于有限的罚，系数 α_j 必须满足一个线性约束集，见习题 5.14。

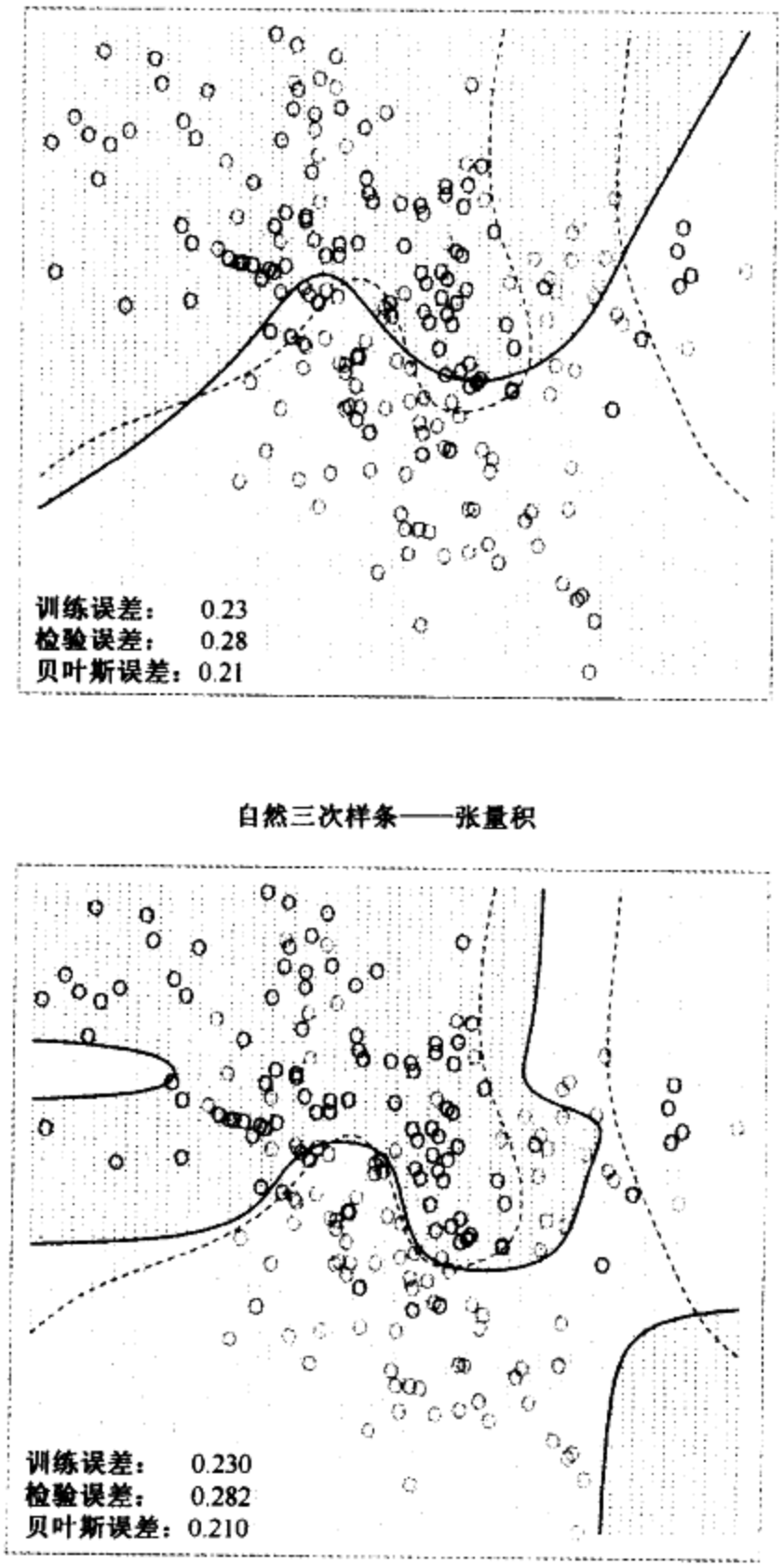


图 5.11 图 2.1 的模拟例子。上图显示加法逻辑斯缔回归模型的判定边界，在两个坐标上都使用自然样条(全部df = 1 + (4 − 1) + (4 − 1) = 7)。下图显示在每个坐标上使用自然样条基张量积的结果(全部df = 4 × 4 = 16)。紫色虚线边界是该问题的贝叶斯判定边界(见彩页)

对于任意维数 d，薄板样条的定义更加一般，它使用了更一般的 J。

有许多混合方法在实践中很流行，它们都谋求计算和概念上的简洁性。与一维光滑样条不同，薄板样条的计算复杂度为 $O(N^3)$，因为一般没有稀疏结构可以利用。然而，与一元光滑样条一样，我们可以从显著少于解(5.39)指出的 N 个纽结出发。在实践中，通常使用覆盖定义域的纽结格就足够了。正如前面一样，对归约后的展开式计算罚。使用 K 个纽结将计算复杂度降低到 $O(NK^2 + K^3)$。图 5.12 显示用薄板样条拟合某些心脏病风险因子，曲面以围线图显示。所显示的是输入特征的位置，以及拟合中使用的纽结。注意，λ 通过 $\mathrm{df}_\lambda = \mathrm{trace}(S_\lambda) = 15$ 指定。

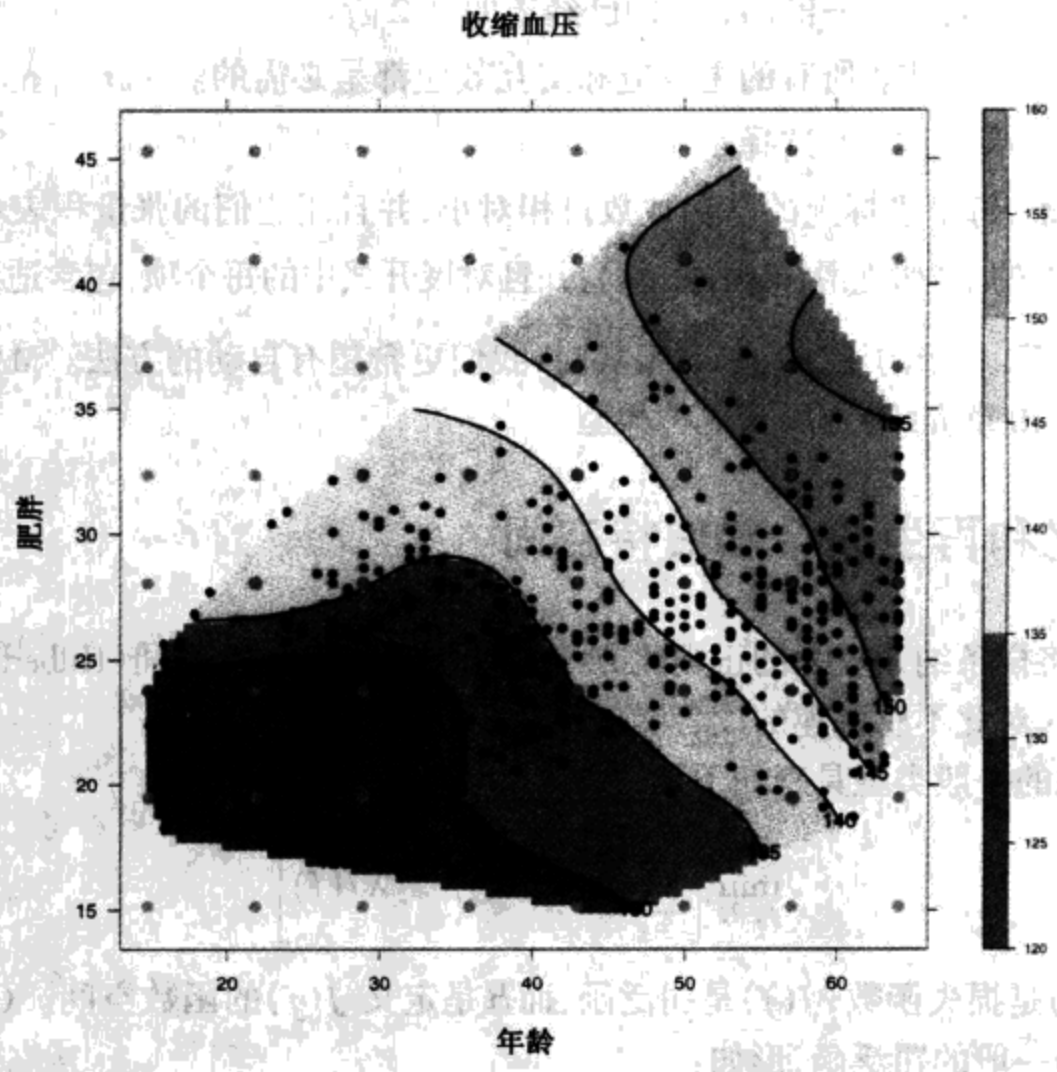


图 5.12　用围线图显示薄板样条拟合心脏病数据。响应是收缩血压(sbp)，它作为年龄(age)和肥胖(obesity)的函数建模。图中标定有数据点以及用做纽结的点的格。谨慎使用来自数据凸包之内(红色)格的纽结，并忽略数据凸包之外(绿色)格的纽结(见彩页)

更一般地，可以将 $f \in \mathbb{R}^d$ 表示成任意大的基函数集的展开式，并通过形如式(5.38)的正则化控制复杂性。例如，可以通过形成所有一元光滑样条基函数对的张量积[如式(5.38)]来构造基；例如，使用第 5.9.2 节推荐的一元 B 样条作为成分。这导致基函数个数随维数指数增长，因而必须减少每个坐标上的函数个数。

第 9 章介绍的加法样条模型是一类受限的多维样条。它们也能用这种一般形式表示，即存在罚 $J[f]$ 使得解具有形式 $f(X) = \alpha + f_1(X_1) + \cdots + f_d(X_d)$，并且每个函数 f_j 是一元样条。在此情况下，罚多少有些退化。而假定 f 是加法的则更自然，从而简单地在每个分量函数上强加一个附加的罚：

$$\begin{aligned} J[f] &= J(f_1 + f_2 + \cdots + f_d) \\ &= \sum_{j=1}^{d} \int f_j''(t_j)^2 \mathrm{d}t_j \end{aligned} \tag{5.40}$$

这些自然延伸到 ANOVA 样条分解，

$$f(X) = \alpha + \sum_j f_j(X_j) + \sum_{j<k} f_{jk}(X_j, X_k) + \cdots \tag{5.41}$$

其中，每个成分是所需维的样条。要做出许多选择：

- 交互作用的最大阶——前面，我们已经表明到 2 阶。
- 包含哪些项——并非所有的主效应和交互效应都是必需的。
- 使用什么表示——一些选择是：
  - 回归样条，每个坐标上的基函数数目相对小，并且用它们的张量积表示交互作用。
  - 一个完整的基(如光滑样条中的基)，并且对展开式中的每个项，包含适当的正则化子。

在许多情况下，可能的维(特征)数很大时，我们更希望有自动的方法。MARS 和 MART(分别在第 9 章和第 10 章介绍)都属于这种类型。

5.8 正则化和再生核希尔伯特空间

本节，我们将样条纳入较大的正则化方法和再生核希尔伯特空间(Hilbert space)问题。本节的技术性很强，对此不感兴趣的读者和初学者可以跳过。

正则化问题的一般类型具有如下形式：

$$\min_{f \in \mathcal{H}} \left[\sum_{i=1}^{N} L(y_i, f(x_i)) + \lambda J(f) \right] \tag{5.42}$$

其中，$L(y, f(x))$是损失函数，$J(f)$是罚泛函，而$\mathcal{H}$是定义 $J(f)$的函数空间。Girosi 等人(1995)介绍了一种相当一般的罚泛函，形如：

$$J(f) = \int_{\mathbb{R}^d} \frac{|\tilde{f}(s)|^2}{\tilde{G}(s)} \mathrm{d}s \tag{5.43}$$

其中，$\tilde{f}$ 表示f 的傅里叶变换，而 $\tilde{G}$ 是一个正函数，随 $\| s \| \to \infty$ 递减到 0。基本思想是 $1/\tilde{G}$ 增加f 的高频分量的罚。在某些附加的假定下，他们证明解具有如下形式：

$$f(X) = \sum_{k=1}^{K} \alpha_k \phi_k(X) + \sum_{i=1}^{N} \theta_i G(X - x_i) \tag{5.44}$$

其中，ϕ_k 生成罚泛函 J 的零空间，而 G 是 $\tilde{G}$ 的逆傅里叶变换。光滑样条和薄板样条包含在该框架之中。这个解的令人瞩目的特点是：准则(5.42)定义在无穷维空间上，而解是有穷维的。在下一节，我们将介绍一些特殊的例子。

5.8.1　用核拓广函数空间

对于式(5.42)形式的问题,一个重要的子类是用正定核 $K(x,y)$拓广,而对应的函数空间 $\mathcal{H}_K$ 称做再生核希尔伯特空间(reproducing kernel Hilbert space, RKHS)。罚泛函 J 也用核定义。我们给出这类模型的简略介绍,取自 Wahba(1990)和 Girosi 等人(1995)的论文,而在 Evgeniou 等人(2001)的论文中有一个很好的概述。

假定 K 具有本征展开:

$$K(x,y)=\sum_{i=1}^{\infty}\gamma_i\phi_i(x)\phi_i(y) \tag{5.45}$$

其中,$\gamma_i\geqslant 0$, $\sum_{i=1}^{\infty}\gamma_i^2<\infty$。$\mathcal{H}_K$ 的元素具有这些本征函数的一个展开式:

$$f(x)=\sum_{i=1}^{\infty}c_i\phi_i(x) \tag{5.46}$$

约束为:

$$\|f\|_{\mathcal{H}_K}^2\overset{\text{def}}{=}\sum_{i=1}^{\infty}c_i^2/\gamma_i<\infty \tag{5.47}$$

其中,$\|f\|_{\mathcal{H}_K}$是 K 导出的范数。式(5.42)中空间$\mathcal{H}_K$ 的罚泛函定义为平方范数 $J(f)=\|f\|_{\mathcal{H}_K}^2$。罚 $J(f)$可以解释为广义岭罚;其中,展开式(5.45)中具有大本征值的函数得到的罚少,反之亦然。

重写式(5.42),我们有:

$$\min_{f\in\mathcal{H}_K}\left[\sum_{i=1}^{N}L(y_i,f(x_i))+\lambda\|f\|_{\mathcal{H}_K}^2\right] \tag{5.48}$$

或等价地:

$$\min_{\{c_j\}_1^{\infty}}\left[\sum_{i=1}^{N}L(y_i,\sum_{j=1}^{\infty}c_j\phi_j(x_i))+\lambda\sum_{j=1}^{\infty}c_j^2/\gamma_j\right] \tag{5.49}$$

可以证明(Wahba,1990,见习题 5.15):式(5.48)的解是有穷维的,并具有如下形式:

$$f(x)=\sum_{i=1}^{N}\alpha_iK(x,x_i) \tag{5.50}$$

基函数 $h_i(x)=K(x,x_i)$(作为第一个变元的函数)称做 x_i 在$\mathcal{H}_K$ 中的估值表示(representer of evaluation),因为对于 $f\in\mathcal{H}_K$,容易看出$\langle K(\cdot,x_i),f\rangle_{\mathcal{H}_K}=f(x_i)$。类似地,$\langle K(\cdot,x_i),K(\cdot,x_j)\rangle_{\mathcal{H}_K}=K(x_i,x_j)$($\mathcal{H}_K$ 的再生性质),从而对于 $f(x)=\sum_{i=1}^{N}\alpha_iK(x,x_i)$,

$$J(f)=\sum_{i=1}^{N}\sum_{j=1}^{N}K(x_i,x_j)\alpha_i\alpha_j \tag{5.51}$$

根据式(5.50)和式(5.51),式(5.48)归结为有穷维准则:

$$\min_{\alpha} L(\mathbf{y}, \mathbf{K}\alpha) + \lambda \alpha^T \mathbf{K}\alpha \tag{5.52}$$

我们使用了向量记号，其中 $\mathbf{K}$ 是 $N \times N$ 矩阵，第 ij 个元素为 $K(x_i, x_j)$。可以使用简单的数值算法来优化式(5.52)。这种将无穷维问题式(5.48)或式(5.49)归结为有穷维优化问题的现象在支持向量机领域(见第 12 章)称为核性质(kernel property)。

这类模型有一个贝叶斯解释，其中 f 被解释为具有先验协方差函数 K 的零均值的固定高斯过程的实现。本征分解过程产生一系列具有相关协方差 γ_j 的正交本征函数 $\phi_j(x)$。典型的情况是"光滑的"函数 ϕ_j 具有较大的先验方差，而"粗糙的"函数 ϕ_j 具有较小的先验方差。式(5.48)中的罚是先验对联合似然的贡献，而具有较小先验方差的分量的罚较大[与式(5.43)比较]。

为简单起见，我们处理了这种情况，其中$\mathcal{H}$的所有成员[如式(5.48)中的成员]都是罚的。更一般地，我们可能希望一些$\mathcal{H}$中的分量不是罚的，如第 5.4 节的三次光滑样条的线性函数。第 5.7 节的多维薄板样条和张量积样条也都属于这一类。在这种情况下，有一种更方便的表示，即 $\mathcal{H} = \mathcal{H}_0 \oplus \mathcal{H}_1$，其中零空间 $\mathcal{H}_0$ 由低阶多项式组成，它们不是罚的。罚变成 $J(f) = \| P_1 f \|$，其中 P_1 是 f 在 $\mathcal{H}_1$ 上的正交投影。解形如 $f(x) = \sum_{j=1}^{M} \beta_j h_j(x) + \sum_{i=1}^{N} \alpha_i K(x, x_i)$，其中第一项表示在 $\mathcal{H}_0$ 中的展开式。根据贝叶斯定理的观点，$\mathcal{H}_0$ 中成分的系数具有不恰当的先验，具有无限方差。

5.8.2 RKHS 的例子

上面的机制通过核 K 和损失函数 L 的选取导出。首先考虑平方误差损失函数的回归。在此情况下，式(5.48)特指罚最小二乘方，并且解可以用两种等价方法刻画，对应于式(5.49)或式(5.52)：

$$\min_{\{c_j\}_1^\infty} \sum_{i=1}^{N} \left(y_i - \sum_{j=1}^{\infty} c_j \phi_j(x_i) \right)^2 + \lambda \sum_{j=1}^{\infty} \frac{c_j^2}{\gamma_j} \tag{5.53}$$

无穷维的、广义岭回归问题，或

$$\min_{\alpha} (\mathbf{y} - \mathbf{K}\alpha)^T (\mathbf{y} - \mathbf{K}\alpha) + \lambda \alpha^T \mathbf{K}\alpha \tag{5.54}$$

α的解可以简单地得到：

$$\hat{\alpha} = (\mathbf{K} + \lambda \mathbf{I})^{-1} \mathbf{y} \tag{5.55}$$

并且

$$\hat{f}(x) = \sum_{j=1}^{N} \hat{\alpha}_j K(x, x_j) \tag{5.56}$$

拟合值的 N 向量由下式给出：

$$\begin{aligned} \hat{\mathbf{f}} &= \mathbf{K}\hat{\alpha} \\ &= \mathbf{K}(\mathbf{K} + \lambda \mathbf{I})^{-1} \mathbf{y} \end{aligned} \tag{5.57}$$

$$= (\mathbf{I} + \lambda \mathbf{K}^{-1})^{-1} \mathbf{y} \tag{5.58}$$

估计(5.57)在空间统计学中还作为高斯随机域的克瑞精(kriging)估计出现(Cressie, 1993)。比较式(5.58)和光滑样条拟合(5.17)。

罚多项式回归

核 $K(x,y)=(\langle x,y\rangle+1)^d$ $(x,y\in \mathrm{IR}^p)$(Vapnik, 1996)具有 $M=\binom{p+d}{d}$ 个本征函数,它们生成总次数为 d 的 IR^p 上的多项式空间。例如,当 $p=2$ 和 $d=2$ 时,$M=6$,并且

$$K(x,y) = 1+2x_1y_1+2x_2y_2+x_1^2y_1^2+x_2^2y_2^2+2x_1x_2y_1y_2 \tag{5.59}$$

$$= \sum_{m=1}^{M} h_m(x)h_m(y) \tag{5.60}$$

而

$$h(x)^T=(1,\sqrt{2}x_1,\sqrt{2}x_2,x_1^2,x_2^2,\sqrt{2}x_1x_2) \tag{5.61}$$

可以用 M 个正交本征函数和 K 的本征值表示 h

$$h(x)=\mathbf{V}\mathbf{D}_\gamma^{\frac{1}{2}}\phi(x) \tag{5.62}$$

其中,$\mathbf{D}_\gamma=\mathrm{diag}(\gamma_1,\gamma_2,\cdots,\gamma_M)$,而 $\mathbf{V}$ 是 $M\times M$ 矩阵并且是正交的。

假定我们希望解罚多项式回归问题

$$\min_{\{\beta_m\}_1^M}\sum_{i=1}^{N}\left(y_i-\sum_{m=1}^{M}\beta_m h_m(x_i)\right)^2+\lambda\sum_{m=1}^{M}\beta_m^2 \tag{5.63}$$

将式(5.62)代换到式(5.63)中,得到式(5.53)形式的待优化表达式(见习题 5.16)。

基函数的个数 $M=\binom{p+d}{d}$ 可能很大,通常比 N 大得多。由式(5.55)可知,如果使用解函数的核表示,只须计算核 N^2 次,并可以在 $O(N^3)$ 操作内计算解。

这种简洁性并非明显。式(5.61)中的每个多项式 h_m 从 K 的特定形式继承了一个伸缩因子,这对式(5.63)中的罚具有影响。

高斯径向基函数

在前面的例子中,核的选取是因为它提供了一个多项展开式,并可以方便地计算高维内积。在这个例子中,核的选取是因为它在表达式(5.50)中的函数形式。

高斯核 $K(x,y)=e^{-\|x-y\|^2/2\sigma^2}$ 与平方误差损失导致一个回归模型,它是高斯径向基函数的展开式

$$h_j(x)=e^{-\|x-x_j\|^2/2\sigma^2},\ j=1,\cdots,N \tag{5.64}$$

每一个的中心都在一个训练数据 x_i 上。系数用式(5.54)估计。我们已经知道(Girosi 等人, 1995)薄板样条(见第 5.7 节)是径向基函数的展开式,由核

$$K(x,y)=\|x-y\|^2\log(\|x-y\|) \tag{5.65}$$

产生。径向基函数将在第 6.7 节更详细地讨论。

支持向量分类法

第 12 章的支持向量机对于 2-类分类问题具有形式 $f(x)=\alpha_0+\sum_{i=1}^{N}\alpha_i K(x,x_i)$，其中参数的选取是为了极小化

$$\min_{\alpha_0,\alpha}\left\{\sum_{i=1}^{N}[1-y_i f(x_i)]_+ + \lambda\alpha^T\mathbf{K}\alpha\right\} \tag{5.66}$$

其中，$y_i\in\{-1,1\}$，而 $[z]_+$ 表示 z 的正的部分。这可以视为具有线性约束的二次优化问题，并需要二次规划算法求解。支持向量(support vector)的名字源于如下事实：典型地，许多 $\hat{\alpha}_i=0$ [由于(式 5.66)中损失函数的分段零(piecewise-zero)性质]，因而 $\hat{f}$ 是 $K(\cdot,x_i)$ 的子集上的展开式(详见第 12.3.3 节)。

5.9 小波光滑

我们已经看到使用基函数字典的两种不同的操作模式。对于回归样条，或者使用主题知识，或者自动地选择基函数的一个子集。诸如 MARS(见第 9 章)这样更加自适应的过程可以捕获光滑性和非光滑性。对于光滑样条，我们使用完整的基函数，但朝向光滑性收缩系数。

典型地，小波使用完全标准正交基表示函数，但朝着稀疏表示选择和收缩系数。正如光滑函数可以用少量样条基函数表示一样，具有少量孤立隆起的平坦函数可以用少量(隆起)基函数表示。小波基在信号处理和压缩方面非常流行，因为它们可以用有效的方式表示光滑或局部隆起的函数——一种称为时间和频率局部性(time and frequency localization)的现象。相比之下，传统的傅里叶基只允许频率的局部性。

在给出细节之前，让我们考察图 5.13 左部的 Haar(哈尔)小波，以得到小波光滑如何起作用的直观思想。垂直轴指示小波的标度(频率)，从底部的低标度到顶部的高标度。在每一个标度，小波并排地“填充”，完全填满时间轴：我们只显示一个选定的子集。通过最小二乘方，小波拟合这些基的系数，然后丢弃(过滤掉)较小的系数。由于在每个标度都有许多基函数，它可以使用需要的那些，而丢弃不需要的那些，以得到时间和频率的局部性。Haar 小波简单易懂，但对于大部分用途它不够光滑。图 5.13 右部的 symmlet 小波具有相同的正交性，但比较光滑。

图 5.14 显示核磁共振(nuclear magnetic resonance，NMR)信号，看上去它由一些光滑分量和孤立的尖峰，加上一些噪声组成。使用 symmlet 基的小波变换如图 5.14 的左下图所示。小波系数按行排列，从底部的最低标度到顶部的最高标度。每条线段的长度指出系数的大小。其中的右下图显示取阈值后的小波系数。阈值过程[在下面的式(5.68)给出]是软阈值规则，与出现在线性回归的套索过程(见第 3.4.3 节)中的相同。注意，一些较小的系数已经设置为 0。上图中的绿色曲线是取阈值后的系数的反向变换，这是原始信号的光滑版本。在下一节，我们将给出该过程的细节，包括小波的构造和阈值规则。

5.9.1 小波基和小波变换

本节，将介绍小波构造和过滤的细节。小波基通过单个定标函数 $\phi(x)$(也称 father)的变换和放大产生。图 5.15 中上两幅图的曲线是 Haar 和 symmlet-8 定标函数。Haar 基很容易理

解，特别是对于具有方差或树分析经验的人更是如此，因为它产生分段常数表示。这样，如果 $\phi(x) = I(x \in [0, 1])$，则 $\phi_{0,k}(x) = \phi(x - k)$（$k$ 是一个整数）为在这些整数上具有跳跃的函数产生一个正交基。我们称它为参考空间（reference space）V_0。扩张 $\phi_{1,k}(x) = \sqrt{2}\phi(2x - k)$ 形成在 1/2 长度区间上分段常数函数空间 $V_1 \supset V_0$ 的一个正交基。事实上，更一般地，我们有 $\cdots \supset V_1 \supset V_0 \supset V_{-1} \supset \cdots$，其中每个 V_j 由 $\phi_{j,k} = 2^{j/2}\phi(2^j x - k)$ 生成。

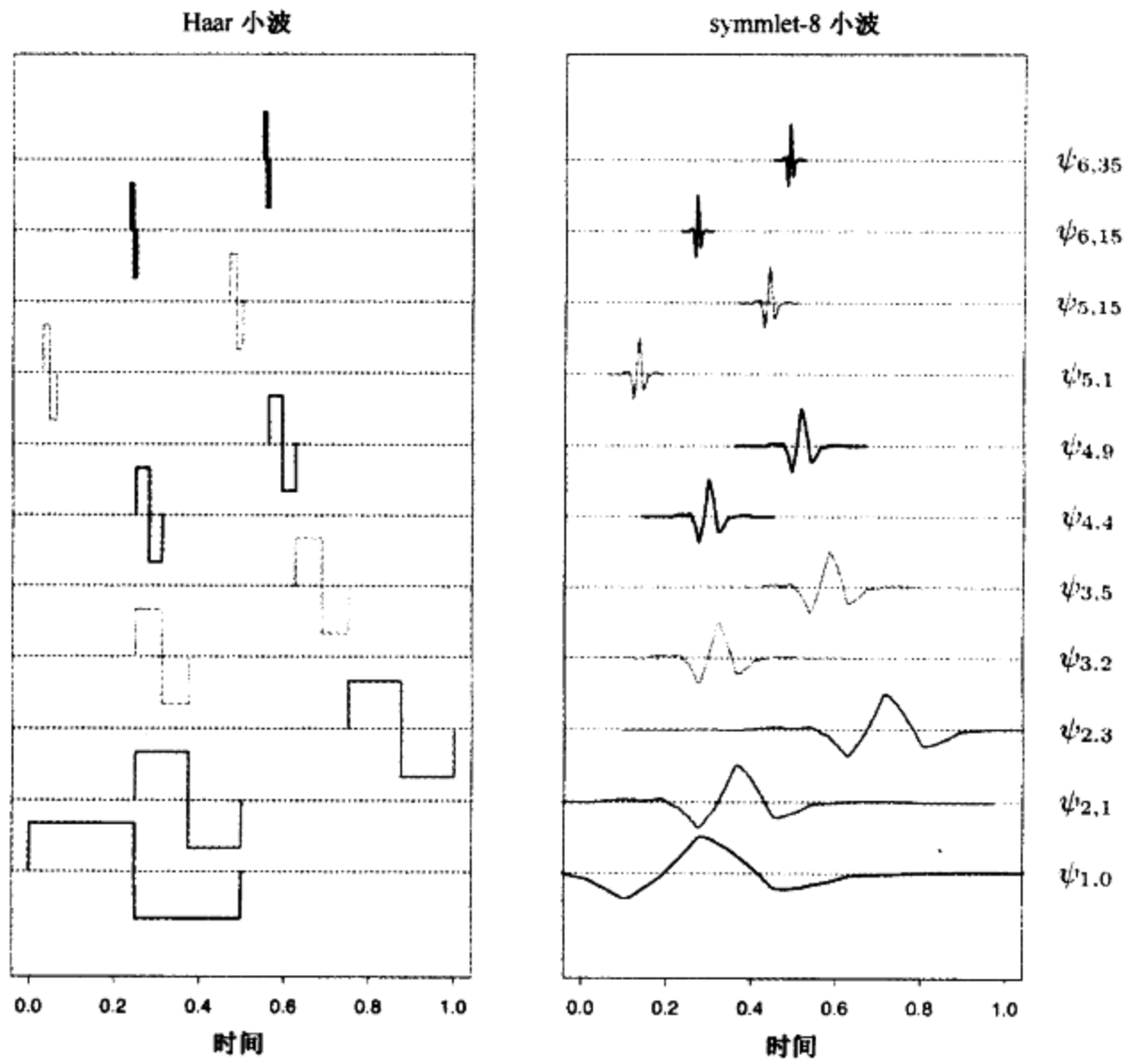


图 5.13　对于 Haar 和 symmlet 族，不同变换和放大选择的小波。函数已经过缩放，以适于显示

现在看小波的定义。在方差分析中，均值对 μ_1 和 μ_2 通常用总均值 $\mu = 1/2(\mu_1 + \mu_2)$ 和对比 $\alpha = 1/2(\mu_1 - \mu_2)$ 表示。如果对比 α 很小，则可以简化，因为可以将它设置为 0。用类似的方法，可以用 V_j 中的分量，加上 V_j 到 V_{j+1} 的正交补 W_j 中的分量来表示 V_{j+1} 中的函数，记做 $V_{j+1} = V_j \oplus W_j$。W_j 中的分量表示细节（detail），我们可能希望将其中的某些分量设置为 0。不难看出由母小波（mother wavelet）$\psi(x) = \phi(2x) - \phi(2x - 1)$ 产生的函数 $\psi(x - k)$ 形成了 Haar 族 W_0 的正交基。类似地，$\psi_{j,k} = 2^{j/2}\psi(2^j x - k)$ 形成 W_j 的基。

既然 $V_{j+1} = V_j \oplus W_j = V_{j-1} \oplus W_{j-1} \oplus W_j$，因此除了用 j 层细节和 j 层粗糙分量表示函数外，后者还可以进一步分解成 $(j - 1)$ 层的细节和粗糙分量，如此下去。最后，我们得到形如 $V_J = V_0 \oplus W_0 \oplus W_1 \oplus \cdots \oplus W_{J-1}$ 的表示。图 5.13 显示特定的小波 $\psi_{j,k}(x)$。

注意，由于这些空间是正交的，因此所有的基函数也是正交的。事实上，如果定义域是离

散的，具有 $N=2^J$ 个(时间)点，这正是我们所能得到的。在 j 层有 2^j 个基元素，并且在 W_j 中总共有 2^J-1 个基元素，在 V_0 中有一个。这种结构化的正交基支持多分辨率分析(multiresolution analysis)，我们将在下一节解释。

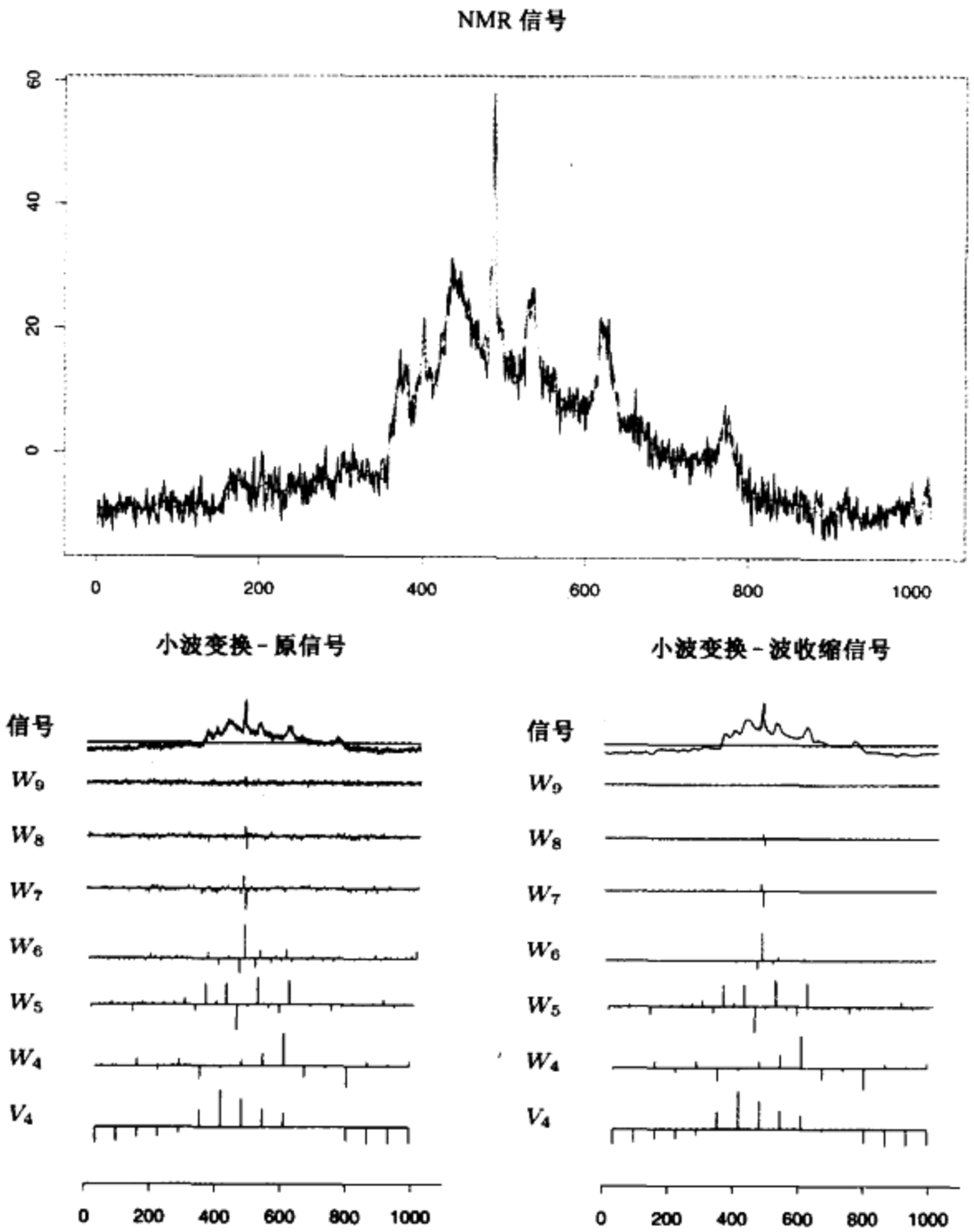


图 5.14 上图显示一个 NMR 信号，小波收缩版本以绿色叠加。左下图表示原信号的小波变换，使用symmlet-8基函数，取到 V_4。每个系数用垂直条的高度(正的或负的)表示。右下图表示使用S-PLUS中的waveshrink函数收缩后的小波系数。waveshrink函数实现了Donoho和Johnstone的小波自适应SureShrink方法(见彩页)

尽管对于理解上述构造是有帮助的，但是Haar基通常太粗糙，不适合实际应用。幸而，已经发明了一些更好的小波基。图 5.13 和图 5.15 包含了Daubechies symmlet-8基。这个基具有比对应的Haar基更光滑的元素，但有一些权衡：

- 每个小波有一个支集，涵盖 15 个相继时间区间，而不是Haar基的一个。更一般地，symmlet-p族具有$2p-1$个相继区间的支集。支集越宽，小波衰减到 0 的时间越长，因而

就越光滑。注意，有效的支集看来非常狭窄。

- symmlet-p小波$\phi(x)$具有 p 个消失瞬间，即：

$$\int \phi(x)x^j \mathrm{d}x = 0,\ j = 1, \cdots, p$$

一个推论是，在 $N = 2^J$ 个时间点上的任意 p 次多项式恰好在 V_0 中再生(见习题 5.17)。在此意义下，V_0 等价于光滑样条罚的零空间。Haar 小波具有一个消失瞬间，因而 V_0 可以再生任意常数函数。

symmlet-p标度函数是若干小波产生族中的一个。其操作类似于 Haar 基的操作：

- 如果 V_0 由 $\phi(x-k)$生成，则对于某过滤系数 $h(k)$，$V_1 \supset V_0$ 由 $\phi_{1,k}(x) = \sqrt{2}\phi(2x-k)$和 $\phi(x) = \sum_{k\in Z} h(k)\phi_{1,k}(x)$ 生成。
- W_0 由 $\psi(x) = \sum_{k\in Z} g(k)\phi_{1,k}(x)$ 生成，过滤系数为 $g(k) = (-1)^k h(1-k)$。

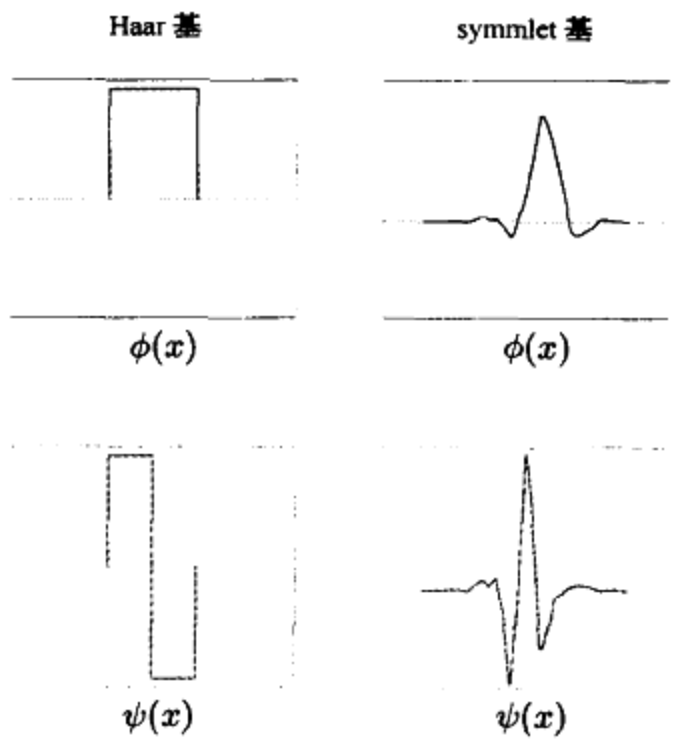


图 5.15　Haar 和 symmlet 父(标度)小波$\psi(x)$和母小波$\psi(x)$

5.9.2　自适应小波过滤

当离散的信号、图像或时间序列等数据在均匀的格上度量时，小波特别有用。我们将关注一维情况，并且有 $N = 2^J$ 个格点。设 $\mathbf{y}$ 是响应向量，$\mathbf{W}$ 是 $N \times N$ 的正交小波基矩阵，在 N 个均匀分布的观测上计算。则称 $\mathbf{y}^* = \mathbf{W}^T\mathbf{y}$ 是 $\mathbf{y}$ 的小波变换(wavelet transform)(并且是完全最小二乘方回归系数)。一种流行的自适应小波拟合方法称做 SURE 收缩[Stein Unbiased Risk Estimation，Stein无偏风险估计(Donoho和Johnstone，1994)]：

$$\min_{\boldsymbol{\theta}} ||\mathbf{y} - \mathbf{W}\boldsymbol{\theta}||_2^2 + 2\lambda||\boldsymbol{\theta}||_1 \tag{5.67}$$

这与第 3 章的套索准则相同。由于 $\mathbf{W}$ 是正交的，将导致简单的解：

$$\hat{\theta}_j = \mathrm{sign}(y_j^*)(|y_j^*| - \lambda)_+ \tag{5.68}$$

最小二乘方系数向 0 变换，并在 0 上截断。拟合函数（向量）则由逆小波变换（inverse wavelet transform）$\hat{\mathbf{f}} = \mathbf{W}\hat{\boldsymbol{\theta}}$ 给出。

λ 的一个简单选择是 $\lambda = \sigma\sqrt{2\log N}$，其中 σ 是噪声的标准偏差的估计。可以给出该选择的一些动机。由于 $\mathbf{W}$ 是一个正交变换，如果 $\mathbf{y}$ 的元素是白噪声（独立的高斯变量，具有均值 0，方差 σ^2），则 $\mathbf{y}^*$ 也是。进一步说，如果随机变量 Z_1，Z_2，…，Z_N 是白噪声，$|Z_j|(j=1,\cdots,N)$ 的期望最大值近似为 $\sigma\sqrt{2\log N}$。因此，所有小于 $\sigma\sqrt{2\log N}$ 的系数多半是噪声，并被设置为 0。

空间 $\mathbf{W}$ 可以是任意正交函数的基：多项式、自然样条或余弦。使得小波特殊是所用基函数的特定形式，它支持时间和频率的局部表示。

让我们再考察图 5.14 中的 NMR 信号。小波变换使用symmlet-8基计算。注意，系数不是一路传递到 V_0，而是停止于具有 16 个基函数的 V_4。随着我们进入每个细节层，除表现尖刺行为的位置外，系数变小。小波系数表现出信号的时间局部性（每层的基函数相互变换）和频率局部性。每个放大会将细节增加一个因子 2，并在此意义下对应于将传统傅里叶表示的频率加倍。事实上，对小波的进一步数学理解揭示，在特定标度上的小波具有受限于有限变程或倍频程的傅里叶变换。

图 5.14 右下图中的收缩/截断使用本节引言介绍的 SURE 方法实现。正交的 $N \times N$ 基矩阵 $\mathbf{W}$ 的列是小波基函数在 N 个时间点上求值。特殊地，在此情况下，有 16 列对应于 $\phi_{4,k}(x)$，而其余的作用于 $\psi_{j,k}(x)$，$j=4,\cdots,11$。在实践中，λ 依赖于噪声方差，必须通过数据估计（如最高层系数的方差）。

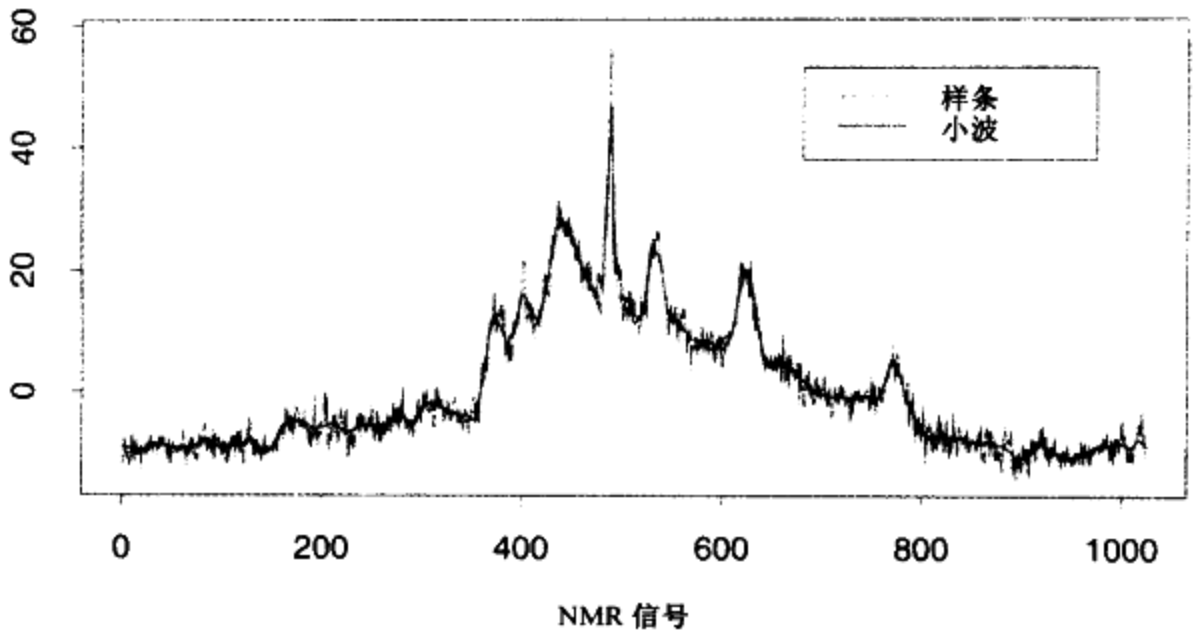
注意 SURE 准则（5.67）和光滑样条准则（5.21）之间的类似性：

- 二者都是由粗糙到细节分层地构造，尽管小波在每一级分辨率内都是时间局部的。
- 通过强加微分收缩常数 d_k，样条偏向于光滑函数。SURE 收缩的早期版本同等地处理所有标度。S + wavelets函数waveshrink()有许多选项，其中一些允许微分收缩。
- 样条 L_2 罚导致纯收缩，而 SURE L_1 罚进行收缩和选择。

更一般的光滑样条利用光滑性实现原信号的压缩，而小波利用稀疏性实现原信号的压缩。图 5.16 利用两个本质上不同的例子，比较了小波拟合（使用SURE收缩）和光滑样条拟合（使用交叉验证）。对于上图的NMR数据，光滑样条到处引进细节，以便保留孤立尖峰中的细节；而小波拟合很好地局部化尖峰。在下图中，真实的函数是光滑的，并且噪声相对很高。小波拟合带来一些多余和不必要的摆动——为多余的自适应性付出的方差代价。

小波变换不是通过如 $\mathbf{y}^* = \mathbf{W}^T\mathbf{y}$ 中的矩阵乘法实现的。事实上，使用精巧的棱锥模式，$\mathbf{y}^*$ 可以在 $O(N)$ 时间得到，甚至比快速傅里叶变换（FFT）的 $N\log(N)$ 还快。尽管对一般构造的介绍已经超出本书范围，但对于 Haar 基容易看出（见习题 5.18）。类似地，小波逆变换 $\mathbf{W}\hat{\boldsymbol{\theta}}$ 也是 $O(N)$。

这只是对小波这个内容广泛、成长快速的领域的简要介绍。存在大量建立在小波之上的数学和计算库。现代图像压缩通常使用二维小波。



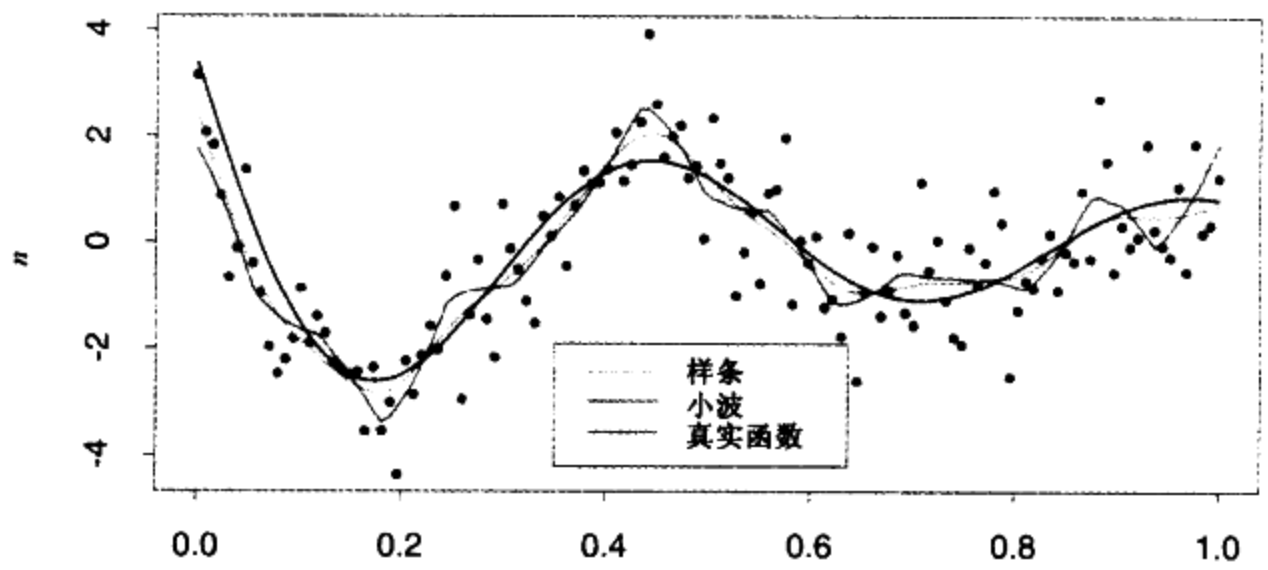


图 5.16　小波光滑与光滑样条比较的两个例子。每幅图给出 SURE 收缩小波拟合与交叉验证光滑样条拟合的比较(见彩页)

文献注释

在de Boor(1978)中详细讨论了样条和 B 样条。Green和Silverman(1994)、Wahba(1990)介绍了光滑样条和薄板样条的详尽处理;后者还涵盖了再生核希尔伯特空间。对于一些使用RKHS方法的无参回归技术之间的联系,见Girosi等人(1995)和Evgeniou等人(2000)的论文。对泛函数据建模(见第 5.2.3 节)详见Ramsay和Silverman(1997)。

Daubechies(1992)是小波的经典和数学的处理。其他有用的资料包括 Chui(1992) 和 Wickerhauser(1994)。Donoho和Johnstone(1994)基于统计学估计框架开发了 SURE 收缩和选择技术,另见Vidakovic(1999)。Bruce和Gao(1996)是有用的应用导论,其中还介绍了 S-PLUS 中的小波软件。

习题

5.1 证明式(5.3)中的截尾幂基函数为具有指定的两个纽结的三次样条提供了一个基。

5.2 设 $B_{i,M}(x)$是本章附录中式(5.76)到式(5.77)定义的 M 阶 B 样条。

(a) 归纳地证明:对于 $x \notin [\tau_i, \tau_{i+M}]$, $B_{i,M}(x) = 0$。这表明三次 B 样条的支集最多 5 个纽结。

(b) 归纳地证明:对于 $x \in (\tau_i, \tau_{i+M})$, $B_{i,M}(x) > 0$。B 样条在它的支集内部为正。

(c) 使用归纳法证明: $\sum_{i=1}^{K+M} B_{i,M}(x) = 1, \forall x \in [\xi_0, \xi_{K+1}]$。

(d) 证明:$B_{i,M}$是$[\xi_0, \xi_{K+1}]$上 $M(M-1$ 次)阶分段多项式,仅在纽结$\xi_1, \cdots, \xi_K$ 上有断点。

(e) 证明:M 阶 B 样条基函数是 M 个均匀随机变量的卷积的密度函数。

5.3 写一个程序,重新产生图 5.3。

5.4 考虑具有 K 个内部纽结的三次样条截尾幂序列表示。设

$$f(X) = \sum_{j=0}^{3} \beta_j X^j + \sum_{k=1}^{K} \theta_k (X - \xi_k)_+^3 \tag{5.69}$$

证明自然三次样条的自然边界条件(见第 5.2.1 节)蕴涵系数上的如下线性约束:

$$\begin{aligned} \beta_2 &= 0, \quad \sum_{k=1}^{K} \theta_k = 0 \\ \beta_3 &= 0, \quad \sum_{k=1}^{K} \xi_k \theta_k = 0 \end{aligned} \tag{5.70}$$

因此导出基式(5.4)和式(5.5)。

5.5 写一个程序,使用二次判别分析(见第 4.3 节)对phoneme数据分类。由于有许多相关特征,应当首先使用自然三次样条的光滑基(见第 5.2.3 节)过滤它们。预先确定 5 种纽结数目和位置的不同选择,并使用 10 折交叉验证确定最终选择。phoneme数据可以从本书网站www-stat.stanford.edu/ElemStatLearn得到。

5.6 假定希望拟合具有已知周期 T 的周期函数。描述应如何修改截尾幂序列基,以实现你的目标。

5.7 光滑样条的推导(Green和Silverman,1994)。设 $N \geqslant 2$, g 是对偶 $\{x_i, z_i\}_1^N$ 的三次自然样条插值,其中 $a < x_1 < \cdots < x_N < b$。这是一个自然样条,在每个 x_i 处有一个纽结;将它看做 N 维函数空间,我们可以确定系数,使得它准确地对序列 z_i 插值。设 $\tilde{g}$是$[a, b]$上其他可微函数,对这 N 个点插值。

(a) 设 $h(x) = \tilde{g}(x) - g(x)$。使用 g 上的边界条件和分部积分证明:

$$\int_a^b g''(x)h''(x)\mathrm{d}x = -\sum_{j=1}^{N-1} g'''(x_j^+)\{h(x_{j+1}) - h(x_j)\} \tag{5.71}$$

(b) 使用 g 是自然三次样条的事实证明该表达式为 0,因而

$$\int_a^b \tilde{g}''(t)^2\mathrm{d}t \geqslant \int_a^b g''(t)^2\mathrm{d}t$$

(c) 证明仅当 h 在$[a, b]$上恒为 0 时等式成立。

(d) 考虑罚最小二乘方问题

$$\min_f \left[\sum_{i=1}^{N}(y_i - f(x_i))^2 + \lambda \int_a^b f''(t)^2 \mathrm{d}t\right]$$

使用(b)证明极小必定是一个在每个 x_i 上具有纽结的三次样条。

5.8 在本章附录中,我们表明如何使用 B 样条的$(N + 4)$维基,有效地计算光滑样条。使用定义在 $N-2$ 个内部纽结上的 B 样条的$(N+2)$维基,描述一个稍微简化的策略。

5.9 导出光滑样条的 Reinsch 形式 $\mathbf{S}_\lambda = (\mathbf{I} + \lambda \mathbf{K})^{-1}$。

5.10 导出 $\mathrm{Var}(\hat{f}_\lambda(x_0))$和 $\mathrm{bias}(\hat{f}_\lambda(x_0))$的表达式。使用例(5.22),创建图 5.9 的一个版本,其中显示 $\hat{f}_\lambda(x)$的均值和若干(逐点)分位数。

5.11 证明:对于光滑样条,$\mathbf{K}$ 的零空间由 X 上的线性函数生成。

5.12 描述如下问题解的特性:

$$\min_f \mathrm{RSS}(f, \lambda) = \sum_{i=1}^{N} w_i\{y_i - f(x_i)\}^2 + \lambda \int \{f''(t)\}^2 \mathrm{d}t \tag{5.72}$$

其中 $w_i \geqslant 0$ 是观测权。

当训练数据在 X 上有结时,描述光滑样条问题(5.9)解的性质。

5.13 你已经用光滑样条 $\hat{f}_\lambda$ 拟合 N 个对偶(x_i, y_i)的样本。假定用对偶 x_0, $\hat{f}_\lambda(x_0)$增广你的原始样本,并重新拟合,描述结果。用它导出 N 折交叉验证式(5.26)。

5.14 导出薄板样条展开式(5.39)中在α_j 上的约束,确保罚 $J(f)$是有限的。如何用其他方法确保罚有限?

5.15 本题推导第 5.8.1 节提出的一些结果。假定 $K(x, y)$满足条件(5.45),并设 $f(x) \in \mathcal{H}_K$。证明:

(a) $\langle K(\cdot, x_i), f\rangle_{\mathcal{H}_K} = f(x_i)$

(b) $\langle K(\cdot, x_i), K(\cdot, x_j)\rangle_{\mathcal{H}_K} = K(x_i, x_j)$

(c) 如果 $g(x) = \sum_{i=1}^{N} \alpha_i K(x, x_i)$,则

$$J(g) = \sum_{i=1}^{N}\sum_{j=1}^{N} K(x_i, x_j)\alpha_i\alpha_j$$

假定设 $\tilde{g}(x) = g(x) + \rho(x)$,其中 $\rho(x) \in \mathcal{H}_K$,并且在$\mathcal{H}_K$,中正交于每个 $K(x, x_i)$, $i = 1, \cdots, N$。证明:

(d)

$$\sum_{i=1}^{N} L(y_i, \tilde{g}(x_i)) + \lambda J(\tilde{g}) \geqslant \sum_{i=1}^{N} L(y_i, g(x_i)) + \lambda J(g) \tag{5.73}$$

当且仅当 $\rho(x) = 0$ 时等式成立。

5.16 考虑岭回归问题(5.53),并假定 $M \geqslant N$。假定你有一个核 K,它计算内积 $K(x, y) = \sum_{m=1}^{M} h_m(x) h_m(y)$。

(a) 导出式(5.62)。给定 K,如何计算矩阵 $\mathbf{V}$ 和 $\mathbf{D}_y$? 据此证明式(5.63)与式(5.53)等价。

(b) 证明:

$$\begin{aligned}\hat{\mathbf{f}} &= \mathbf{H}\hat{\boldsymbol{\beta}} \\ &= \mathbf{K}(\mathbf{K}+\lambda\mathbf{I})^{-1}\mathbf{y}\end{aligned} \tag{5.74}$$

其中,$\mathbf{H}$ 是求值 $h_m(x_i)$ 的 $N\times M$ 矩阵,而 $\mathbf{K}=\mathbf{H}\mathbf{H}^T$ 是内积 $h(x_i)^T h(x_j)$ 的 $N\times N$ 矩阵。

(c) 证明:

$$\begin{aligned}\hat{f}(x) &= h(x)^T\hat{\boldsymbol{\beta}} \\ &= \sum_{i=1}^{N} K(x,x_i)\hat{\boldsymbol{\alpha}}_i\end{aligned} \tag{5.75}$$

而 $\hat{\boldsymbol{\alpha}}=(\mathbf{K}+\lambda\mathbf{I})^{-1}\mathbf{y}$。

(d) 如果 $M<N$,如何修改你的解?

5.17 symmlet-p 小波基的标度函数 $\phi(x)$ 有多达 p 阶瞬间。证明这意味 p 次多项式恰好在 V_0 中表示。[注:V_0 是第 5.9.1 节中定义的参考空间。]

5.18 证明长度为 $N=2^J$ 的信号的 Haar 小波变换可以在 $O(N)$ 时间计算。

样条的计算考虑

在这个附录中,我们介绍表示多项式样条的 B 样条基。还将讨论它们在光滑样条计算中的使用。

附录:B 样条

在开始之前,我们需要扩充第 5.2 节定义的纽结序列。设 $\xi_0<\xi_1$ 和 $\xi_K<\xi_{K+1}$ 是两个边界纽结;典型地,它们定义了希望对样条求值的定义域。现在,定义扩展的纽结序列 τ,使得:

- $\tau_1\leqslant\tau_2\leqslant\cdots\leqslant\tau_M\leqslant\xi_0$;
- $\tau_{j+M}=\xi_j, j=1,\cdots,K$;
- $\xi_{K+1}\leqslant\tau_{K+M+1}\leqslant\tau_{K+M+2}\leqslant\cdots\leqslant\tau_{K+2M}$。

越出边界的那些附加纽结的实际值是任意的,并且通常令它们分别等于 ζ_0 和 ζ_{K+1}。

对于纽结序列 τ,$m\leqslant M$,记 $B_{i,m}(x)$ 为第 i 个 m 次 B 样条基函数。它们用均差递归地定义如下:

$$B_{i,1}(x) = \begin{cases}1 & \text{如果}\tau_i\leqslant x<\tau_{i+1}\\ 0 & \text{其他}\end{cases} \tag{5.76}$$

其中,$i=1,\cdots,K+2M-1$。这些也称为 Haar 基函数。

$$B_{i,m}(x) = \frac{x-\tau_i}{\tau_{i+m-1}-\tau_i}B_{i,m-1}(x)+\frac{\tau_{i+m}-x}{\tau_{i+m}-\tau_{i+1}}B_{i+1,m-1}(x) \tag{5.77}$$

其中,$i=1,\cdots,K+2M-m$。

这样,当 $M=4$ 时,$B_{i,4}(i=1,\cdots,K+4)$ 是纽结序列 ξ 的 $K+4$ 个三次 B 样条基函数。该递归过程可以继续,将产生任意次样条的 B 样条基。图 5.17 显示直至 4 次 B 样条的序列,纽

结在点 *0.0*, 0.1, …, 1.0 上。由于我们创建了一些重复的纽结，因此需要小心避免被 0 除。要是我们接受约定：如果 $\tau_i = \tau_{i+1}$，则 $B_{i,1} = 0$；根据归纳法，如果 $\tau_i = \tau_{i+1} = \cdots = \tau_{i+m}$，则 $B_{i,m} = 0$。还要注意，在上述构造中，对于阶 $m < M$、具有纽结 ξ 的 B 样条基，只需要子集 $B_{i,m}$，$i = M - m + 1, \cdots, M + K$。

为了完全理解这些函数的性质，并且证明它们确实对该纽结序列生成三次样条空间，需要包括均差在内的附加的数学机制。习题 5.2 论述了这些问题。

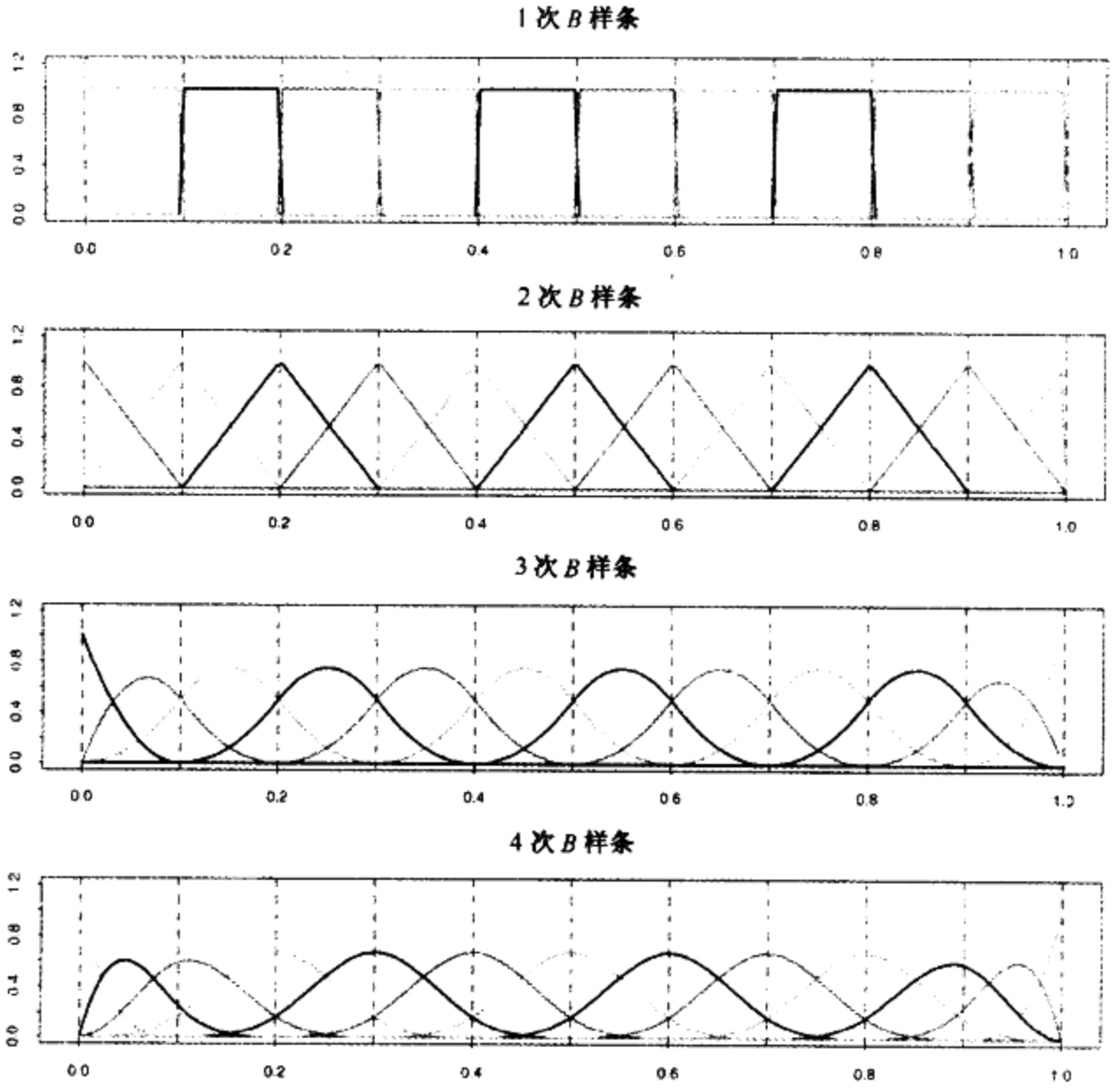


图 5.17　1 次至 4 次 B 样条的序列，其中 10 个纽结均匀地分布在 0 到 1 之间。B 样条具有局部支集，它们在由 $M + 1$ 个纽结生成的区间上非零（见彩页）

事实上，B 样条的辖域比这里声称的大，并且必须处理纽结重复。如果在上述序列 τ 的构造中有一个重复的内部纽结，并和以前一样产生 B 样条序列，则结果基生成的分段多项式空间在该重复纽结上少一阶连续导数。一般地，如果除重复的边界纽结外，我们包含内部纽结 ξ_j $1 \leqslant r_j \leqslant M$ 次，则在 $x = \xi_j$ 上最低阶不连续导数将是 $M - r_j$ 阶。这样，对于没有重复纽结的三次样条，$r_j = 1, j = 1, \cdots, K$，并在每个内部纽结上三阶 $(4-1)$ 导数是不连续的。重复第 j 个纽结三次，导致不连续的 1 阶导数；重复 4 次，导致不连续的零阶导数，即函数在 $x = \xi_j$ 上是不连续的。这正是在边界上发生的；我们重复边界纽结 M 次，因此样条在边界纽结上不连续（即越出边界无定义）。

B 样条的局部支集具有重要的计算含义，特别是当纽结数 K 很大时。具有 N 个观测和 $K+M$ 个变量(基函数)的最小二乘方计算需要 $O(N(K+M)^2+(K+M)^3)$ 次 flop(浮点操作)。如果 K 略小于 N，则导致 $O(N^3)$ 算法；对于很大的 N，这是不能接受的。如果 N 个观测有序，则在 N 个点求值的 $K+M$ 个 B 样条基函数组成的 $N\times(K+M)$ 回归矩阵的许多元素为 0。这可以用来将计算复杂性降低到 $O(N)$。我们将在下一节进一步讨论该问题。

光滑样条计算

尽管自然样条(见第 5.2.1 节)为光滑样条提供了一个基，但是在较大的无约束的 B 样条空间上更便于计算。我们记 $f(x)=\sum_1^{N+4}\gamma_jB_j(x)$，其中 γ_j 是系数，而 B_j 是三次 B 样条基函数。解看上去与以前的一样：

$$\hat{\gamma}=(\mathbf{B}^T\mathbf{B}+\lambda\Omega_B)^{-1}\mathbf{B}^T\mathbf{y} \tag{5.78}$$

不同的是我们用 $N\times(N+4)$ 矩阵 $\mathbf{B}$ 替换了 $N\times N$ 矩阵 $\mathbf{N}$，而类似地，$(N+4)\times(N+4)$ 罚矩阵 $\mathbf{\Omega}_B$ 替换了 $N\times N$ 维矩阵 $\mathbf{\Omega}_N$。尽管表面看没有边界导数约束，但是事实上通过给越过边界任意非零导数无限权，罚项自动施加了这些约束。实践中，$\hat{\gamma}$ 被限制在有限子空间，罚在其上总是有限的。

由于 $\mathbf{B}$ 的列是求值的 B 样条，在 X 的有序值上从左到右依次计算，并且三次 B 样条具有局部支集，$\mathbf{B}$ 是较低的 4-带的。结果是，矩阵 $\mathbf{M}=(\mathbf{B}^T\mathbf{B}+\lambda\mathbf{\Omega})$ 是 4-带的，因而它的 Cholesky 分解 $\mathbf{M}=\mathbf{L}\mathbf{L}^T$ 易于计算。然后，通过回代，可以解 $\mathbf{L}\mathbf{L}^T\gamma=\mathbf{B}^T\mathbf{y}$ 得到 γ，因而在 $O(N)$ 次操作得到解。

在实践中，当 N 很大时，不必使用所有 N 个内部纽结，并且任何合理的稀释(thinning)策略都可以节省计算时间，而对拟合的影响可以忽略。例如，S-PLUS 中的 smooth.spline 函数使用一种近似的对数策略：如果 $N<50$，则使用所有纽结；但是，即便 $N=5000$，也只使用 204 个纽结。

第6章 核 方 法

本章,我们将讨论一类回归技术。通过在每个查询点 x_0 分别拟合不同但简单的模型,它们灵活地估计 $\mathbb{R}^p$ 上的回归函数 $f(X)$。其基本思想是:仅使用靠近目标点 x_0 的观测拟合简单的模型,并且使结果估计函数在 $\mathbb{R}^p$ 上是光滑的。这种局部化通过加权函数或核 $K_\lambda(x_0, x_i)$ 来实现。加权函数根据 x_i 到点 x_0 的距离赋予 x_i 一个权。通常,核 K_λ 用参数 λ 标引,而 λ 指示邻域的宽度。原则上,这些基于内存的方法需要少量或不需要训练;所有工作都在求值时完成。惟一需要由训练数据确定的参数是 λ。然而,模型是整个训练数据集上的模型。

本章讨论更一般的基于核的技术。这些技术与其他章节的结构化方法有着紧密联系,并且对于密度估计和分类是有用的。

6.1 一维核光滑方法

在第2章,我们导出 k-最近邻平均

$$\hat{f}(x) = \text{Ave}(y_i | x_i \in N_k(x)) \tag{6.1}$$

作为回归函数 $\text{E}(Y|X=x)$的估计。这里,$N_k(x)$是平方距离最邻近 x 的 k 个点的集合,而 Ave 表示取平均值(均值)。其基本思想是,放宽条件期望的定义(见图6.1左部),并在目标点的邻域上计算平均值。在图6.1中,我们使用30-最近邻——x_0 上的拟合是30个对偶的平均值,这些对偶的 x_i 值最接近 x_0。绿色曲线绘制出我们在不同 x_0 值上使用该定义的轨迹。绿色曲线是起伏不平的,因为 $\hat{f}(x)$在 x 上不连续。随着从左向右移动 x_0,k-最近邻域保持不变,直到某个点 x_i 到 x_0 的右边比邻域中的最远点 $x_{i'}$ 到 x_0 的左边更近,此时用 x_i 替代 $x_{i'}$。式(6.1)中的平均值离散地改变,导致不连续的 $\hat{f}(x)$。

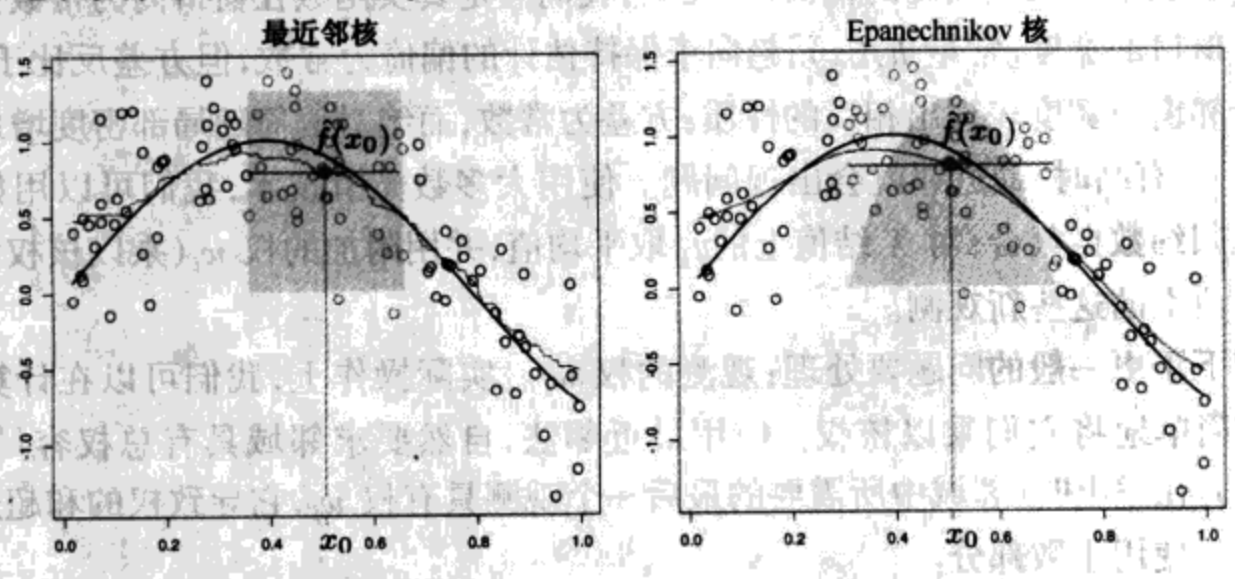


图6.1 每幅图中的100个 x_i, y_i 对由具有高斯误差 $Y = \sin(4X) + \varepsilon$, $X \sim U[0,1]$, $\varepsilon \sim N(0,1/3)$的蓝色曲线随机产生。在左图中,绿色曲线是30-最近邻移动均值(running-mean)光滑的结果。红色点是被拟合的常数 $\hat{f}(x_0)$,而橘黄色阴影圆指示那些对 x_0 上的拟合有贡献的观测。实心橘黄色区域指示赋予观测的权。在右图中,绿色曲线是核加权平均,使用(半个)窗口宽度为 $\lambda = 0.2$ 的 Epanechnikov 核(见彩页)

这种不连续很不好，并且是不必要的。我们可以把随其到目标点的距离平滑衰减的权赋给邻域中的点，而不是将相同的权赋给邻域中的所有点。右图给出了一个例子，使用称为Nadaraya-Watson的核加权平均：

$$\hat{f}(x_0) = \frac{\sum_{i=1}^{N} K_\lambda(x_0, x_i) y_i}{\sum_{i=1}^{N} K_\lambda(x_0, x_i)} \tag{6.2}$$

使用 Epanechnikov 二次核：

$$K_\lambda(x_0, x) = D\left(\frac{|x - x_0|}{\lambda}\right) \tag{6.3}$$

其中

$$D(t) = \begin{cases} \frac{3}{4}(1 - t^2) & \text{如果} |t| \leqslant 1 \\ 0 & \text{其他} \end{cases} \tag{6.4}$$

现在，图 6.1 右部的拟合函数是连续的，并且相当光滑。随着从左向右移动目标点，点进入邻域的初始权为 0，而后其贡献缓慢递增(见习题 6.1)。

在右图中，我们使用了大小为 $\lambda = 0.2$ 的度量窗口，用于核拟合，其大小并不随目标点 x_0 的移动而改变，而 30-最近邻光滑窗口自适应 x_i 的局部密度。然而，也可以对核使用自适应的邻域，但需要使用更一般的记号。设 $h_\lambda(x_0)$是一个宽度函数(被 λ 标引)，它确定 x_0 的邻域宽度。则更一般地，我们有：

$$K_\lambda(x_0, x) = D\left(\frac{|x - x_0|}{h_\lambda(x_0)}\right) \tag{6.5}$$

在式(6.3)中，$h_\lambda(x_0) = \lambda$ 是常数。对于 k-最近邻域，邻域的大小 k 取代了 λ，并且有 $h_k(x_0) = |x_0 - x_{[k]}|$，其中，$x_{[k]}$是第 k 个最邻近 x_0 的 x_i。

在实践中，有一些细节必须留意：

- 必须确定光滑参数 λ，它们确定局部邻域的宽度。较大的 λ 意味较低的方差(在更多的观测上取平均)，但较高的偏倚(本质上，我们假定真实函数在窗口内为常数)。
- 度量窗口的宽度[常量 $h_\lambda(x)$]趋向于保持估计的偏倚为常数，但方差反比于局部密度。最近邻窗口宽度表现出相反的性质：方差为常数，而绝对偏倚随局部密度增加而减少。
- 当 x_i 上有结时，最近邻就会出现问题。使用大多数光滑技术，我们可以用如下方法简单地归约数据集合：对 X 结值上的y_i 取平均值，并用附加的权 w_i(乘以核权)补充 x_i 的惟一值上的这些新观测。
- 这留下了更一般的问题要处理：观测的权 w_i。实际操作上，我们可以在计算加权平均之前简单地将它们乘以核权。使用最近邻法，自然要求邻域具有总权容量 k(相对于 $\sum w_i$)。在溢出时(邻域中所需要的最后一个观测具有权 w_j，它导致权的和超过预计 k)，则可以使用小数部分。
- 边界问题：度量邻域趋向于在边界上包含较少的点，而最近邻域变得较宽。
- Epanechnikov 核具有紧致支集(在与最近邻窗口容量一起使用时是需要的)。另一种流行的紧致核是基于三次方函数：

$$D(t)=\begin{cases}(1-|t|^3)^3 & \text{如果 } |t|\leqslant 1\\ 0 & \text{其他}\end{cases} \tag{6.6}$$

这在顶部较平坦(像最近邻箱),并且在其支集的边界上可微。高斯密度函数 $D(t)=\phi(t)$是一种流行的非紧致核,标准偏差起窗口大小的作用。图 6.2 比较了这三种核。

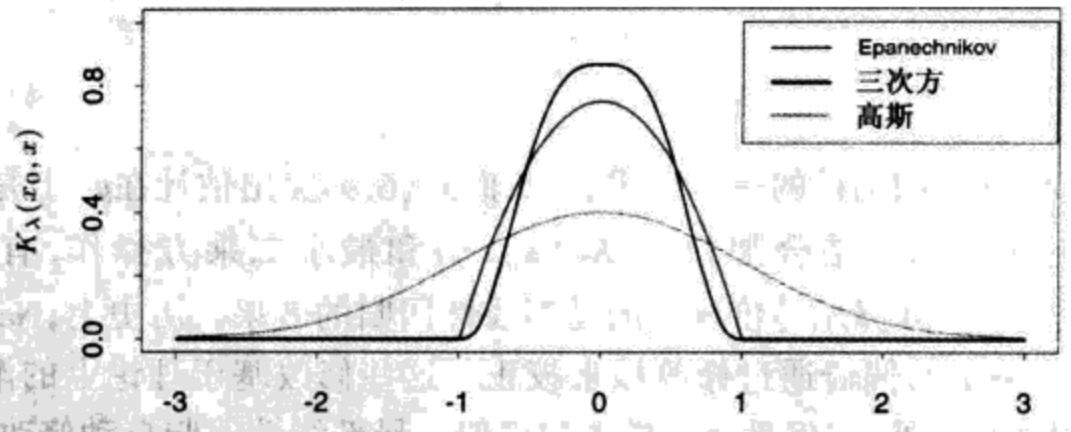


图 6.2 三种流行的局部光滑核的比较。每种都调整得使积分为 1。三次方核是紧致的,并在其支集的边界上具有二阶连续导数,而Epanechnikov核没有。高斯核是连续可微的,但具有无限支集(见彩页)

6.1.1 局部线性回归

我们已经稳步地从粗略的移动平均前进到通过使用核加权的光滑变化的局部加权平均。然而,光滑核拟合依然有问题,如图 6.3(左图)所示。由于核在边界区域上的不对称性,局部加权平均可能在定义域边界上出现严重的偏倚。通过直线,而不是常数拟合,可以将该偏倚移至一阶,见图 6.3(右图)。实际上,如果 X 的值不是等距排列,该偏倚也可能在定义域内部出现(基于同样的原因,但通常不严重)。局部加权线性回归又将产生一阶校正。

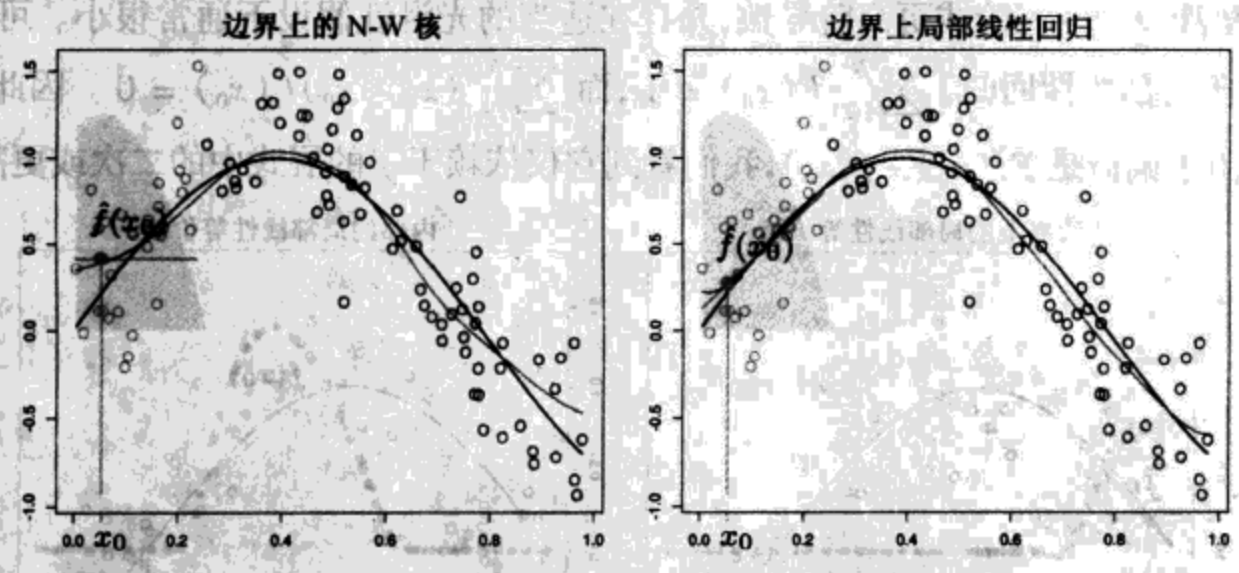


图 6.3 局部加权平均在定义域边界上或接近定义域边界处存在偏倚问题。这里,真实函数是接近线性的,但是邻域中的大部分观测具有比目标点高的均值,因此尽管加权,它们的均值依然偏高。通过拟合局部加权线性回归(右图),该偏倚移至一阶(见彩页)

局部加权回归在每个目标点 x_0 解一个单独的加权最小二乘方问题:

$$\min_{\alpha(x_0),\beta(x_0)}\sum_{i=1}^{N}K_\lambda(x_0,x_i)\,[y_i-\alpha(x_0)-\beta(x_0)x_i]^2 \tag{6.7}$$

估计则是 $\hat{f}(x_0)=\hat{\alpha}(x_0)+\hat{\beta}(x_0)x_0$。注意,尽管我们用整个线性模型拟合该区间的数据,但只

使用它计算在单个点 x_0 上的拟合。

定义向量值函数 $b(x)^T=(1, x)$。设 $\mathbf{B}$ 是 $N\times 2$ 回归矩阵，第 i 行为 $b(x_i)^T$，而 $\mathbf{W}(x_0)$ 是 $N\times N$ 对角矩阵，第 i 个对角线元素为 $K_\lambda(x_0, x_i)$。则

$$\hat{f}(x_0) = b(x_0)^T(\mathbf{B}^T\mathbf{W}(x_0)\mathbf{B})^{-1}\mathbf{B}^T\mathbf{W}(x_0)\mathbf{y} \tag{6.8}$$

$$= \sum_{i=1}^{N} l_i(x_0)y_i \tag{6.9}$$

式(6.8)给出局部线性回归估计的一个显式表示，而式(6.9)突出估计在 y_i 上是线性的[$l_i(x_0)$不涉及 $\mathbf{y}$]。这些权 $l_i(x_0)$ 结合加权核 $K_\lambda(x_0, \cdot)$ 和最小二乘方操作，有时称做等价核(equivalent kernel)。图 6.4 显示等价核上的局部线性回归的效果。历史上，Nadaraya-Watson 和其他局部平均核方法中的偏倚通过修改核来校正。这些修改基于理论上的渐近均方误差考虑，并且除了实现冗长之外，仅仅是有限样本的近似。局部线性回归自动修改该核，将偏倚准确地调整到一阶；这种现象被戏称为自动核木工(automatic kernel carpentry)。使用局部回归的线性性和真实函数 f 在 x_0 附近的级数展开，考虑 $\mathrm{E}\hat{f}(x_0)$的如下展开式：

$$\mathrm{E}\hat{f}(x_0) = \sum_{i=1}^{N} l_i(x_0)f(x_i)$$

$$= f(x_0)\sum_{i=1}^{N} l_i(x_0) + f'(x_0)\sum_{i=1}^{N}(x_i - x_0)l_i(x_0) \tag{6.10}$$

$$+\frac{f''(x_0)}{2}\sum_{i=1}^{N}(x_i - x_0)^2 l_i(x_0) + R$$

其中，余项 R 涉及 f 的三阶或更高阶导数，并且在适当的光滑性假设下通常很小。可以证明(见习题 6.2)对于局部线性回归，$\sum_{i=1}^{N} l_i(x_0) = 1$，而 $\sum_{i=1}^{N}(x_i - x_0)l_i(x_0) = 0$。因此，中项等于 $f(x_0)$，并且由于偏倚是 $\mathrm{E}\hat{f}(x_0) - f(x_0)$，我们看到它仅依赖于 f 的展式中的二次或更高次项。

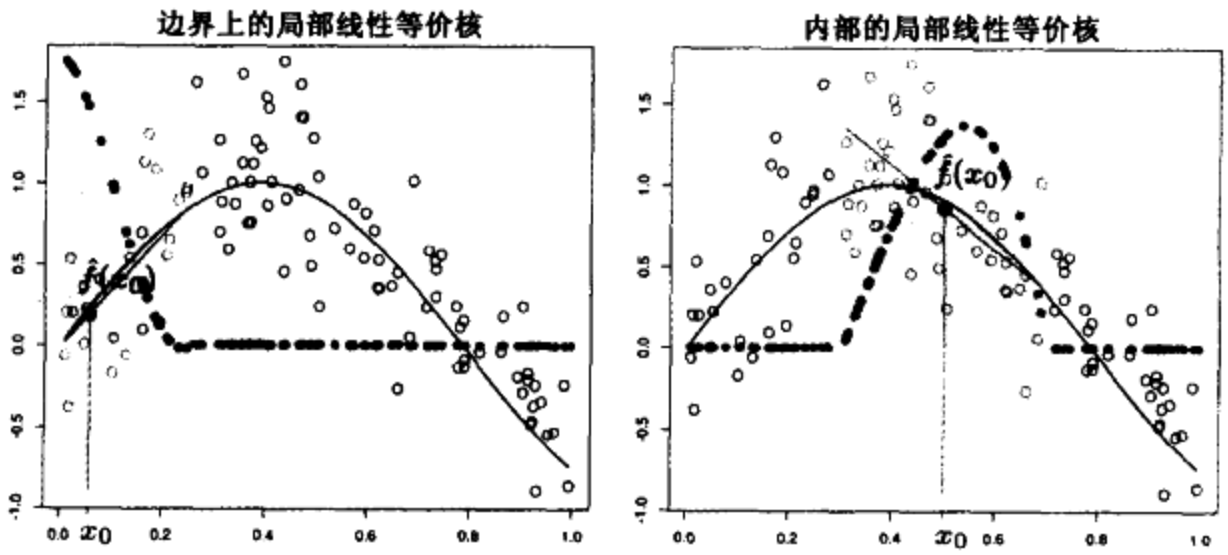


图 6.4 绿色点显示局部回归的等价核 $l_i(x_0)$。这些是 $\hat{f}(x_0) = \sum_{i=1}^{N} l_i(x_0)y_i$ 中的权，参照对应的 x_i 绘出。为了显示，这些已经过缩放，因为事实上它们的和为 1。由于橘黄色阴影区域是(重新缩放的)Nadaraya-Watson 局部平均的等价核，我们看到局部回归如何自动修改加权核，以校正由于光滑窗口中的不对称性而产生的偏倚(见彩页)

6.1.2 局部多项式回归

为什么止步于线性拟合？我们可以用解 $\hat{f}(x_0)=\hat{\alpha}(x_0)+\sum_{j=1}^{d}\hat{\beta}_j(x_0)x_0^j$ 拟合任意 d 次局部多项式拟合

$$\min_{\alpha(x_0),\beta_j(x_0),\,j=1,\cdots,d}\sum_{i=1}^{N}K_\lambda(x_0,x_i)\left[y_i-\alpha(x_0)-\sum_{j=1}^{d}\beta_j(x_0)x_i^j\right]^2 \tag{6.11}$$

事实上，像式(6.10)那样的展开式告诉我们，偏倚仅包含 $d+1$ 次或更高次分量(见习题 6.2)。图 6.5 显示局部二次回归。局部线性回归在真实函数弯曲的区域趋向于是有偏的，这种现象称做截峰填谷。通常，局部二次回归可以校正这种偏倚。

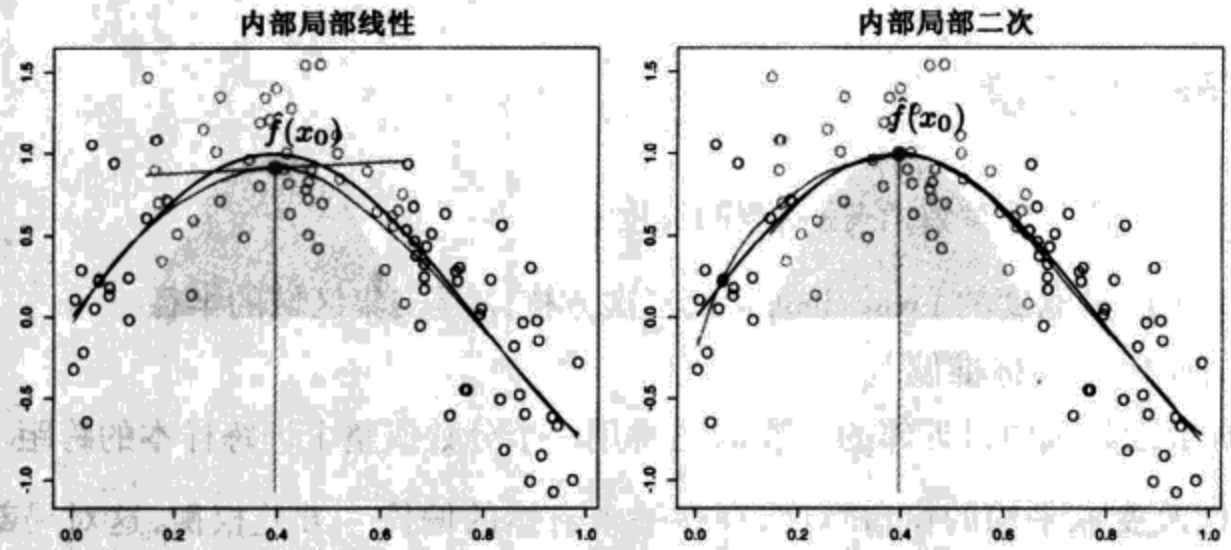


图 6.5 局部线性拟合在真实函数的弯曲部位表现出是有偏的。局部二次拟合趋向于消除这种偏倚(见彩页)

当然，为降低偏倚需要付出代价，而代价是方差的增大。图 6.5 右图中的拟合有稍多的摆动，特别是在尾部。假定模型为 $y_i=f(x_i)+\varepsilon_i$，其中 ε_i 是独立的同分布，均值为 0，而方差为 σ^2，则 $\mathrm{Var}(\hat{f}(x_0))=\sigma^2\|l(x_0)\|^2$，其中 $l(x_0)$ 是 x_0 上等价核权的向量。可以证明(见习题 6.3) $\|l(x_0)\|$ 随 d 增加，因而在选择多项式次数存在偏倚和方差的折中。图 6.6 显示零次、一次和二次局部多项式的方差曲线。关于该问题的一些至理名言汇总如下：

- 局部线性拟合以适当的方差为代价，可以显著减缓边界上的偏倚。局部二次拟合对边界上的偏倚改进不大，但方差增大了许多。
- 局部二次拟合对于降低定义域内部因曲率导致的偏倚最起作用。
- 渐近分析表明，奇次局部多项式优于偶次多项式。主要原因是 MSE 渐近地被边界影响所支配。

尽管做了一些修补，并从边界上的局部线性拟合过渡到内部的局部二次拟合可能有帮助，但并不推荐采用这种策略。通常，应用将支配拟合的多项式次数。例如，如果我们感兴趣于插值，则对边界更感兴趣，而局部线性拟合可能更可靠。

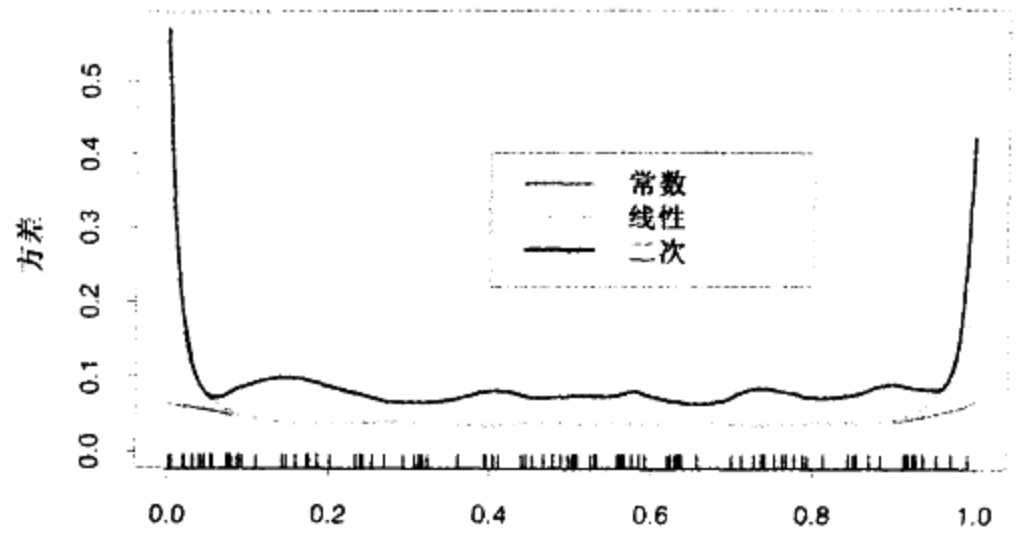


图 6.6 对于度量宽度($\lambda = 0.2$)三次方核,局部常数、线性和二次回归的方差函数 $\| l(x) \|^2$(见彩页)

6.2 选择核的宽度

在每个核 K_λ 中,λ 是参数,它控制核的宽度:

- 对于具有度量宽度的 Epanechnikov 或三次方核,λ 是支集区域的半径。
- 对于高斯核,λ 是标准偏差。
- λ 是 k-最近邻域中最近邻的个数 k,通常用一个分数或整个训练样本的跨距 k/N 表示。

随着我们改变取平均的窗口宽度,存在一个自然的偏倚 - 方差权衡,这对局部平均最明显:

- 如果窗口比较窄,$\hat{f}(x_0)$是少数靠近 x_0 的 y_i 的平均值,并且其方差相对较大——接近个体 y_i 的方差。偏倚趋向于较小,还是因为每个 $E(y_i) = f(x_i)$应当接近于 $f(x_0)$。
- 如果窗口较宽,则由于取平均值的影响,相对于任意 y_i 的方差,$\hat{f}(x_0)$的方差较小。偏倚将比较高,因为我们现在使用的观测 x_i 距 x_0 较远,并且不能保证 $f(x_i)$接近于 $f(x_0)$。

类似的讨论用于局部回归估计,例如局部线性回归估计:随着宽度趋向于 0,估计趋向于一个对训练数据插值的分段线性函数[①];随着宽度趋向于无穷大,拟合趋向于对数据的全局线性最小二乘方拟合。

第 5 章关于为光滑样条选择正则参数的讨论也适用于这里,不再重复。局部回归光滑法是线性估计法;$\hat{\mathbf{f}} = \mathbf{S}_\lambda \mathbf{y}$ 中的光滑子矩阵由等价核(6.8)建立,并且第 ij 个元素是 $\{\mathbf{S}_\lambda\}_{ij} = l_i(x_j)$。留一交叉验证特别简单(见习题 6.7),像广义交叉验证 C_p(见习题 6.10)和 k 折交叉验证一样。有效自由度仍然定义为 $\text{trace}(\mathbf{S}_\lambda)$,并且可以用来调整光滑量。图 6.7 比较了光滑样条和局部线性回归的等价核。局部回归光滑法具有 40% 的跨距,导致 $\text{df} = \text{trace}(\mathbf{S}_\lambda) = 5.86$。光滑样条经调整,具有相同的 df,并且它们的等价核性质非常相似。

① 对于均匀分布的 x_i;对于不规则分布的 x_i,情况可能恶化。

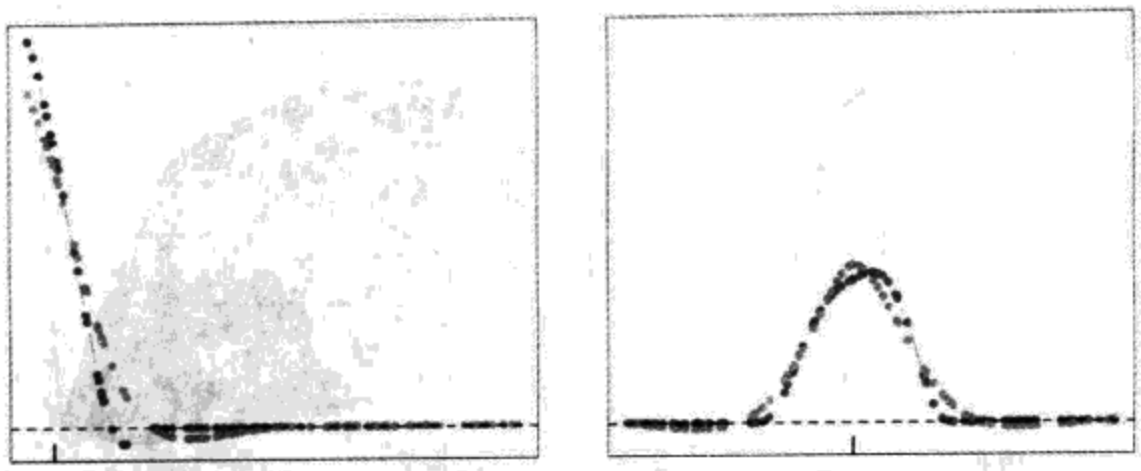


图 6.7 局部线性回归光滑法(三次方核,红色)和光滑样条(绿色)的等价核,具有匹配的自由度。竖直的短线指示目标点(见彩页)

6.3 $\mathbb{R}^p$ 上的局部回归

核光滑和局部回归都可以自然地拓广到二维或更高的维上。Nadaraya-Watson 核光滑法使用由 p 维核提供的权局部地拟合常数。局部线性回归使用由 p 维核提供的权,通过加权最小二乘方局部地拟合 X 上的超平面。其实现很简单,并且通常我们更愿意使用局部常数拟合,因为它在边界上具有很好的性能。

设 $b(X)$是 X 上最高次数为 d 的多项式项的向量。例如,当 $d=1,p=2$时,有$b(X)=(1,X_1,X_2)$;当 $d=2$时,有 $b(X)=(1,X_1,X_2,X_1^2,X_2^2,X_1X_2)$;而一般地,当 $d=0$时,有 $b(X)=1$。在每个 $x_0\in\mathbb{R}^p$,解

$$\min_{\beta(x_0)}\sum_{i=1}^{N}K_\lambda(x_0,x_i)(y_i-b(x_i)^T\beta(x_0))^2 \tag{6.12}$$

产生拟合 $\hat{f}(x_0)=b(x_0)^T\hat{\beta}(x_0)$。典型地,核是径向函数,如径向 Epanechnikov 核或三次方核

$$K_\lambda(x_0,x)=D\left(\frac{\|x-x_0\|}{\lambda}\right) \tag{6.13}$$

其中,$\|\cdot\|$是欧几里德范数(Euclidean norm)。由于欧几里德范数依赖于每个坐标上的单位,在光滑前,将每个预测子标准化最有意义。例如,将每个预测子标准化为单位标准差。

虽然边界影响在一维光滑是个问题,但是在二维或更高维,它们的问题更大,因为边界上的点所占的比例更大。事实上,维灾难的表象之一就是靠近边界的点所占的比例随维数增长到 1。直接修改核,以适应二维边界很棘手,特别是对于不规则的边界。局部多项式回归无缝地将边界调整到任意维上的期望阶。图 6.8 显示在某些具有不寻常预测子(星状)的天文研究观测上的局部线性回归。这里,边界很不规则,并且随着逼近边界,拟合曲面还必须在数据越来越稀疏的区域上插值。

在维数比二维或三维高很多的空间上,局部回归不太有用。我们已经详细讨论了维问题,如在第 2 章。如果整个样本的规模不随 p 指数增长,就不可能在维增长的同时维持局部性(⇒低偏倚)和邻域中的样本大小相当(⇒低方差)。在维数较高的空间中,$\hat{f}(X)$的可视化显示变得非常困难,而这正是光滑的基本目标之一。尽管图 6.8 的散点云(scatter-cloud)和线网(wire-frame)图看上去很吸引人,但除非很粗略,它很难解释结果。从数据分析角度来看,条件图(conditional plot)更加有用。

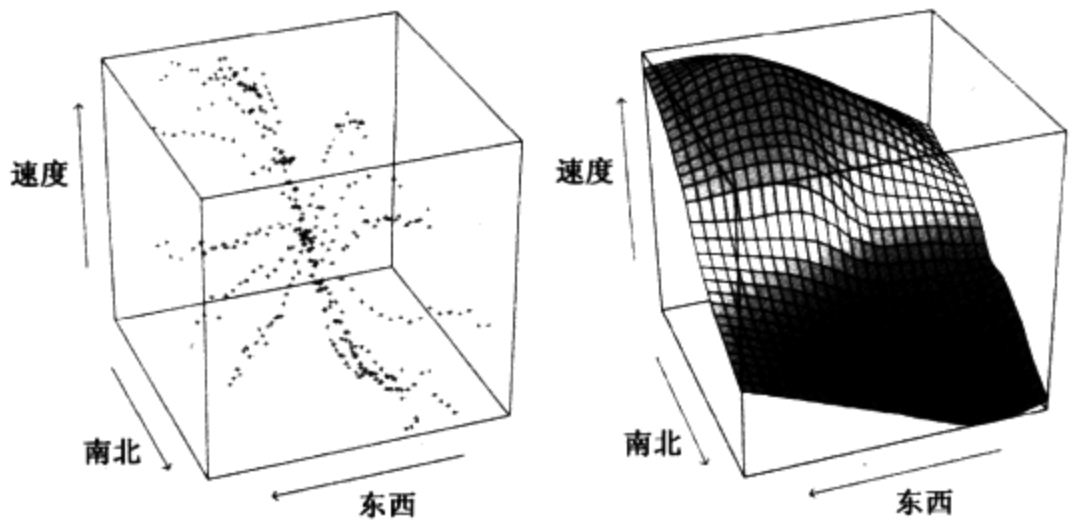


图 6.8 左图显示三维数据，其中响应是在星系上的速度观测，而两个预测子记录在天体上的位置。不寻常的“星状”设计指出所用的观测方法，并导致极不规则的边界。右图显示 $\mathbb{R}^2$ 上局部线性回归的结果，使用包含15%数据的最近邻窗口

图 6.9 显示具有三个预测子的某环境数据分析。这里，格子中将臭氧作为放射性的函数显示，以另外两个变量温度和风速为条件。然而，以变量值为条件实际上蕴涵局部于该值（与局部回归一样）。图 6.9 的每个显示条上方指示每个条件值在该显示条的取值区间。图中显示数据子集（响应与其余变量），并用一维局部回归拟合该数据。尽管这与观察拟合的三维曲面的切片很不相同，但对于理解数据的联合行为也许更有用。

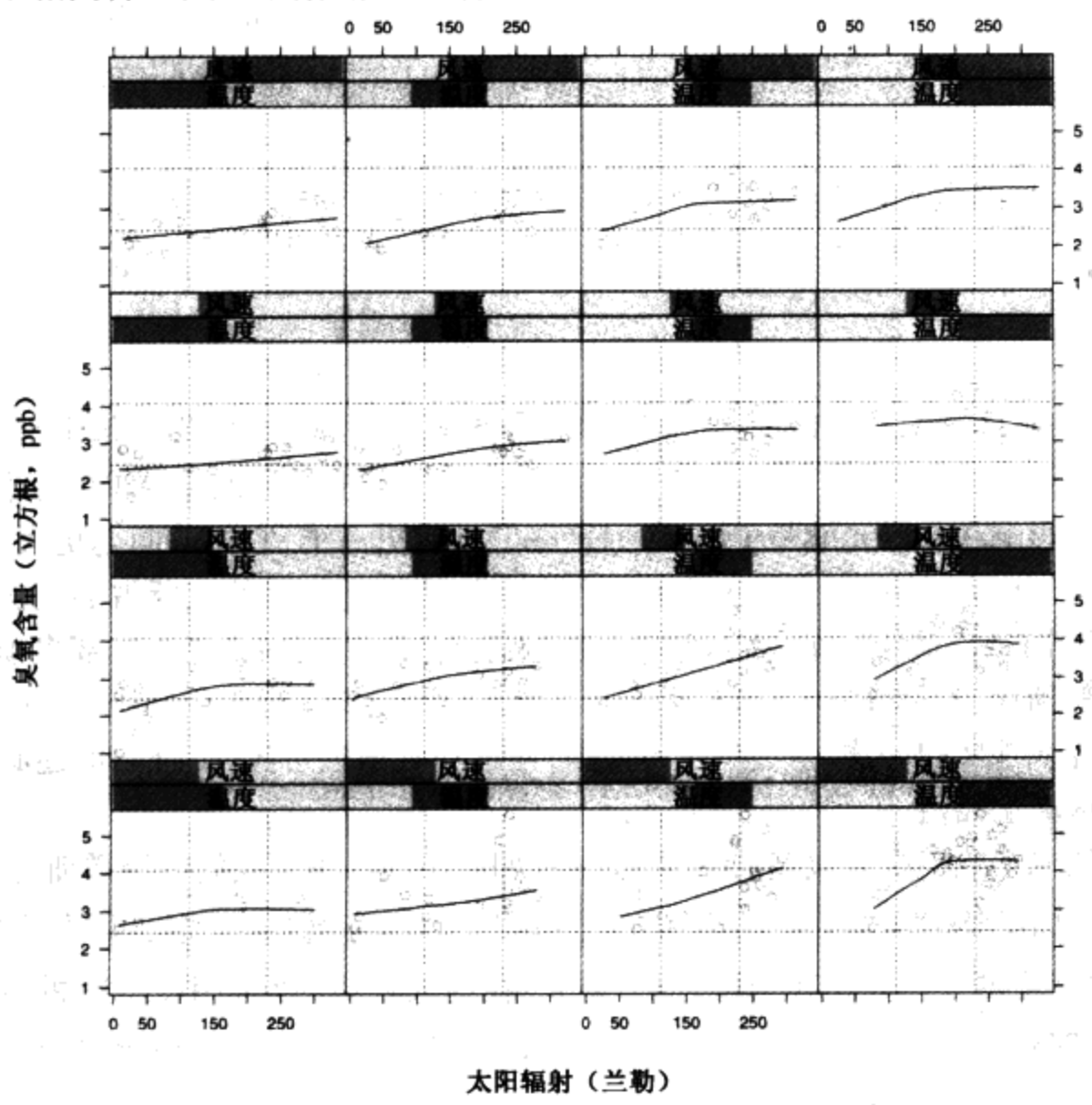


图 6.9 三维光滑的例子。响应是臭氧浓度（立方根），而三个预测子是温度、风速和放射性。图中将臭氧作为放射性的函数显示，以另外两个变量温度和风速为条件。每幅图大约包含每个条件变量值域的40%。每幅图中的曲线是一元局部线性回归，拟合图中的数据

6.4 $\mathbb{R}^p$ 上结构化局部回归模型

当维数与样本量的比例不合适时,局部回归没有多大帮助,除非我们愿意对模型做某些结构化假设。本书的很多内容都是讨论结构化回归和分类模型的。这里将关注一些直接与核方法相关的方法。

6.4.1 结构化核

一种方法是修改核。默认的球形核(6.13)将相等的权赋予每个坐标,因而一种自然的默认策略是将每个变量标准化到单位标准差。更一般的方法是使用一个半正定的矩阵 **A** 对不同的坐标加权:

$$K_{\lambda,A}(x_0,x)=D\left(\frac{(x-x_0)^T\mathbf{A}(x-x_0)}{\lambda}\right) \tag{6.14}$$

整个坐标或方向可以降序排列,或通过在 **A** 上施加适当的限制而忽略。例如,如果 **A** 是对角的,则可以通过增加或减少 A_{jj} 来增大或减小个体预测子 X_j 的影响。通常,预测子很多,并且高度相关(如数字模拟信号或图像中的那些)。可以使用预测子的协方差函数对较少关注高频对比度的度量 **A** 剪裁(见习题 6.4)。已经提出一些从多维核学习参数的提议。例如,第 11 章讨论的投影寻踪回归模型就是这种提议,其中 **A** 的低秩版本蕴涵 $\hat{f}(X)$ 的岭函数。**A** 的更一般的模型是繁琐的,而我们偏爱下面讨论的回归函数的结构化形式。

6.4.2 结构化回归函数

我们试图拟合 $\mathbb{R}^p$ 中的回归函数 $E(Y|X)=f(X_1,X_2,\cdots,X_p)$,其中,每级都可能出现交。很自然的是考虑如下形式的方差分析(ANOVA)分解

$$f(X_1,X_2,\cdots,X_p)=\alpha+\sum_j g_j(X_j)+\sum_{k<\ell} g_{k\ell}(X_k,X_\ell)+\cdots \tag{6.15}$$

然后通过删除某些高阶项引入结构。加法模型假定只有主影响项:$f(X)=\alpha+\sum_{j=1}^p g_j(X_j)$;二阶模型具有总次数最多为 2 的交叉项,依次类推。在第 9 章,我们介绍迭代的反向拟合(backfitting)算法来拟合这种低阶交模型。例如,在加法模型中,如果除第 k 项之外的所有项已知,则可以通过 X_k 上的局部回归 $Y-\sum_{j\neq k} g_j(X_j)$ 来估计 g_k。可以依次对每个函数重复这样的处理,直到收敛。重要的细节是:在任意阶段,所需要的就是一维局部回归。同样的思想可以用于拟合低维 ANOVA 分解。

这些结构化模型的一种重要的特殊情况是变系数模型(varying coefficient model)。例如,假定我们将 X 的 p 个预测子划分到集合 $(X_1,X_2,\cdots,X_q)$ $(q<p)$ 中,而其余的变量收集在向量 Z 中。然后,假定条件线性模型

$$f(X)=\alpha(Z)+\beta_1(Z)X_1+\cdots+\beta_q(Z)X_q \tag{6.16}$$

对于给定的 Z,这是一个线性模型,但是每个系数可能随 Z 变化。很自然的是通过局部加权最小二乘方拟合这种模型:

$$\min_{\alpha(z_0),\beta(z_0)}\sum_{i=1}^N K_\lambda(z_0,z_i)\left(y_i-\alpha(z_0)-x_{1i}\beta_1(z_0)-\cdots-x_{qi}\beta_q(z_0)\right)^2 \tag{6.17}$$

图 6.10 用人体主动脉测量图表述了这种思想。长期以来人们一直认为主动脉随年龄增厚。这里，我们建立模型，将主动脉直径(diameter)作为年龄(age)的函数，但允许系数随性别(gender)和动脉的深度(depth)变化。我们分别对男性和女性使用局部回归模型。主动脉在较高区域随年龄明显地变厚，该关系随着到主动脉的距离而减弱。图 6.11 显示了截距和斜率作为深度的函数。

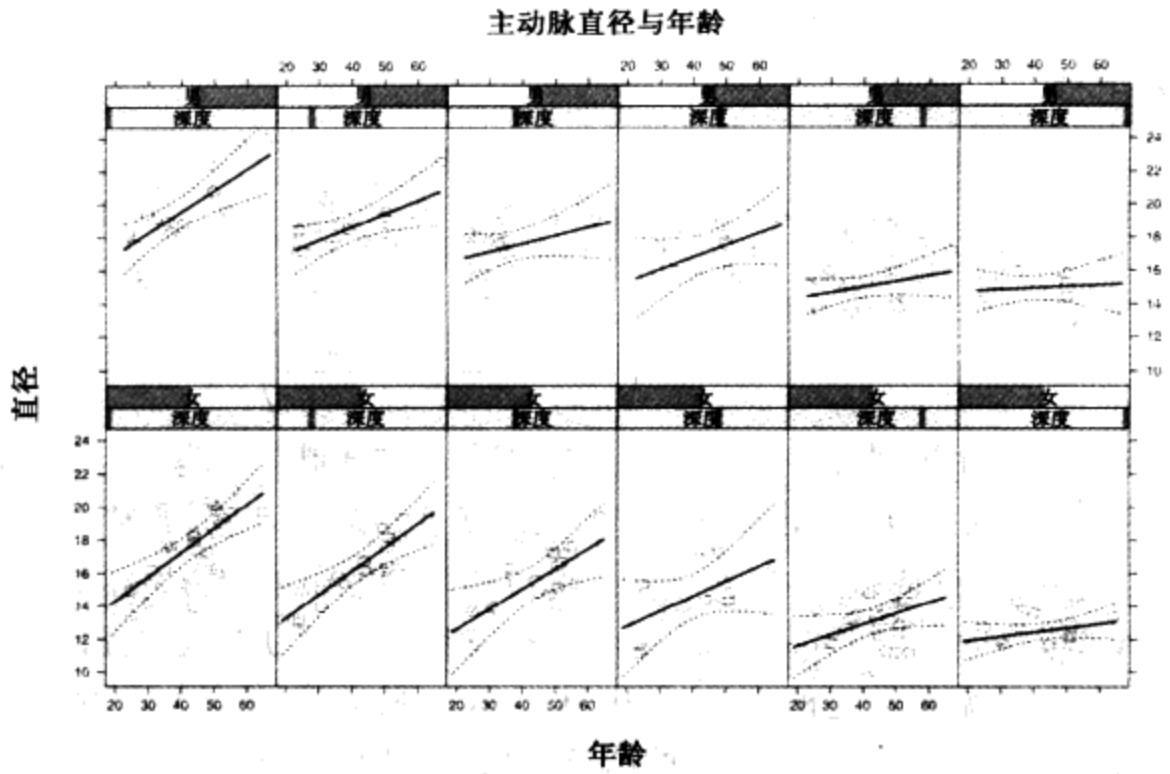


图 6.10 在每个方格中，主动脉直径作为年龄的线性函数。模型的系数随性别和主动脉深度变化(左边靠近顶部，右边靠近下部)。该线性模型的系数存在明显的趋势

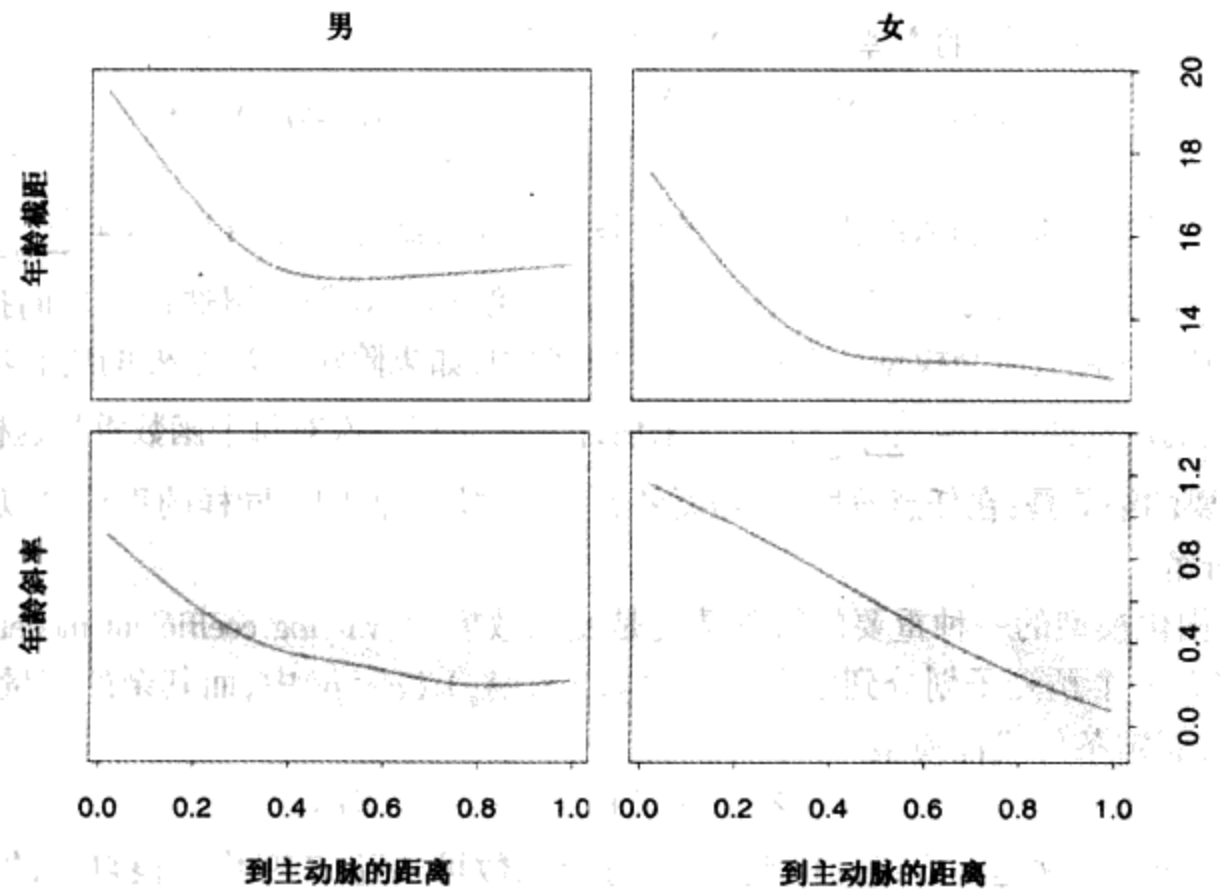


图 6.11 年龄的截距和斜率作为到主动脉距离的函数，分别考虑男性和女性。阴影带宽指示一个标准误差

6.5 局部似然和其他模型

局部回归和变系数模型的概念相当广:如果拟合方法辅以观测权,任何参数模型都可以做成局部的。下面是一些例子:

- 与每个观测y_i相关联的是参数$\theta_i = \theta(x_i) = x_i^T\beta$,在协变量$x_i$上是线性的,并且$\beta$的推断基于对数似然$l(\beta) = \sum_{i=1}^{N} l(y_i, x_i^T\beta)$。通过使用局部于$x_0$的似然推断$\theta(x_0) = x_0^T\beta(x_0)$,可以更灵活地对$\theta(X)$建模:

$$l(\beta(x_0)) = \sum_{i=1}^{N} K_\lambda(x_0, x_i) l(y_i, x_i^T\beta(x_0))$$

  许多似然模型,特别是包括逻辑斯缔和对数线性模型在内的广义线性模型族都涉及线性形式的协变量。局部似然允许将全局线性模型放宽为局部线性模型。
- 除了与θ关联的变量不同于定义局部似然使用的变量之外,与上面的一样:

$$l(\theta(z_0)) = \sum_{i=1}^{N} K_\lambda(z_0, z_i) l(y_i, \eta(x_i, \theta(z_0)))$$

  例如,$\eta(x, \theta) = x^T\theta$可以是$x$上的线性模型。通过极大化局部似然,这将拟合变系数模型$\theta(z)$。
- k阶自回归时间级数具有形式$y_t = \beta_0 + \beta_1 y_{t-1} + \beta_2 y_{t-2} + \cdots + \beta_k y_{t-k} + \varepsilon_t$。记滞后集(lag set)为$z_t = (y_{t-1}, y_{t-2}, \cdots, y_{t-k})$,该模型看上去像一个标准线性模型$y_t = z_t^T\beta + \varepsilon_t$,并且通常用最小二乘方拟合。使用核$K(z_0, z_t)$,通过局部最小二乘方拟合允许模型根据级数的短期历史而变化。这不同于更传统的动态线性模型,它们通过开窗口的时间变化。

作为局部似然的一种解释,考虑第4章的多类线性逻辑斯缔回归模型(4.32)的局部版本。数据由特征x_i和相关联的分类响应$g_i \in \{1, 2, \cdots, J\}$组成,而线性模型具有如下形式:

$$\Pr(G = j | X = x) = \frac{e^{\beta_{j0} + \beta_j^T x}}{1 + \sum_{k=1}^{J-1} e^{\beta_{k0} + \beta_k^T x}} \tag{6.18}$$

对于这种J类模型,局部似然可以记做:

$$\sum_{i=1}^{N} K_\lambda(x_0, x_i) \Big\{ \beta_{g_i 0}(x_0) + \beta_{g_i}(x_0)^T (x_i - x_0) - \log \Big[1 + \sum_{k=1}^{J-1} \exp\left(\beta_{k0}(x_0) + \beta_k(x_0)^T (x_i - x_0)\right) \Big] \Big\} \tag{6.19}$$

注意:

- 在第一行中,以g_i为下标,以提取适当的分子;
- 根据模型的定义,$\beta_{J0} = 0, \beta_J = 0$;

● 我们已经对 x_0 上的局部回归中心化，使得 x_0 上的拟合后验概率简单地为：

$$\hat{\Pr}(G=j|X=x_0)=\frac{e^{\hat{\beta}_{j0}(x_0)}}{1+\sum_{k=1}^{J-1}e^{\hat{\beta}_{k0}(x_0)}} \tag{6.20}$$

该模型可以用于适当低的维上灵活的多类分类，尽管已经报告在高维邮政编码分类问题上该模型取得了成功。使用核光滑方法的广义加法模型（见第 9 章）与此十分接近，并通过假定回归函数具有加法结构而避免维问题。

作为一个简单解释，我们用 2-类局部线性逻辑斯缔模型拟合第 4 章的心脏病数据。图 6.12显示一元局部逻辑斯缔模型（分别地）拟合两个风险因素。当数据本身不能提供多少可视信息时，这是一种检测非线性的有用的显示装置。在该例中，数据中一个未预料的异常被发现。使用传统的方法，该异常可能不会引起注意。

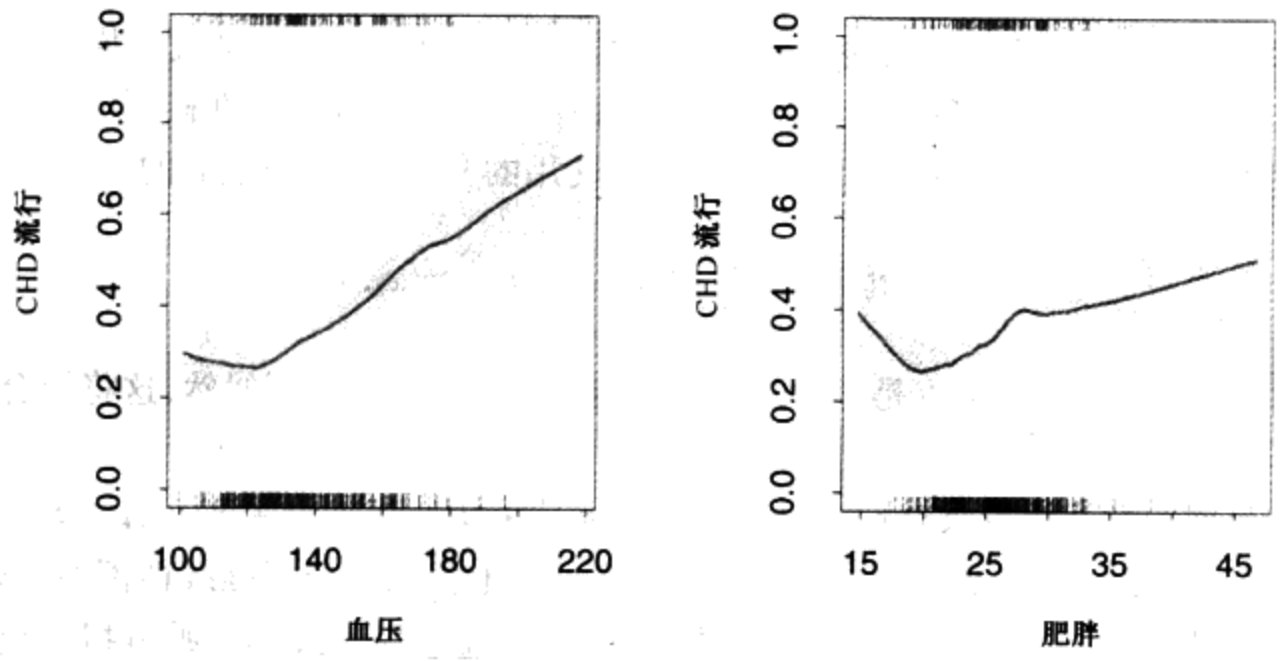


图 6.12 对于南非心脏病数据，每幅图显示二元响应 CHD（冠心病）作为一个风险因素的函数。对于每幅图，我们使用局部线性逻辑斯缔回归模型，计算拟合的CHD流行。区间低端CHD流行的非预期增长是因为这些数据涉及以前的情况，并且有些研究对象已经过治疗，降低了血压和体重。图中的阴影区域指示估计的逐点标准误差带

由于 CHD 是二元指示符，我们可以通过简单地光滑该二元响应，而不直接借助于似然公式来估计条件概率 $\Pr(G=j|x_0)$。这实际上是拟合一个局部常数逻辑斯缔回归模型（见习题 6.5）。为了利用局部线性光滑的偏倚准则，在不受限的分对数尺度下运算更自然。

典型地，使用逻辑斯缔回归，我们计算参数估计以及它们的标准误差。这也可以局部地进行，并且如图所示，还可以产生关于拟合估计的逐点标准误差带。

6.6 核密度估计和分类

核密度估计是一个无指导学习过程，历史上早于核回归。它也很自然地导致一族简单的非参数分类过程。

6.6.1 核密度估计

假定我们有从概率密度 $f_X(x)$提取的随机样本 $x_1, \cdots, x_N$,并希望估计点 x_0 上的 f_X。为简单起见,假定 $X \in \mathbb{R}$。和以前的论证一样,一种自然的局部估计具有如下形式:

$$\hat{f}_X(x_0) = \frac{\#x_i \in \mathcal{N}(x_0)}{N\lambda} \tag{6.21}$$

其中,$\mathcal{N}(x_0)$是 x_0 周围宽度为 λ 的较小度量邻域。该估计是颠簸的,而光滑的 Parzen 估计更可取

$$\hat{f}_X(x_0) = \frac{1}{N\lambda}\sum_{i=1}^{N} K_\lambda(x_0, x_i) \tag{6.22}$$

因为它使用随到 x_0 的距离递减的权处理邻近 x_0 的观测。在此情况下,K_λ 的通常选择是高斯核 $K_\lambda(x_0, x) = \phi(|x - x_0|/\lambda)$。图 6.13 显示高斯核密度拟合 CHD 组群的收缩血压样本值。设 ϕ_λ 表示具有均值 0 和标准差 λ 的高斯密度,则式(6.22)具有如下形式:

$$\begin{aligned} \hat{f}_X(x) &= \frac{1}{N}\sum_{i=1}^{N} \phi_\lambda(x - x_i) \\ &= (\hat{F} \star \phi_\lambda)(x) \end{aligned} \tag{6.23}$$

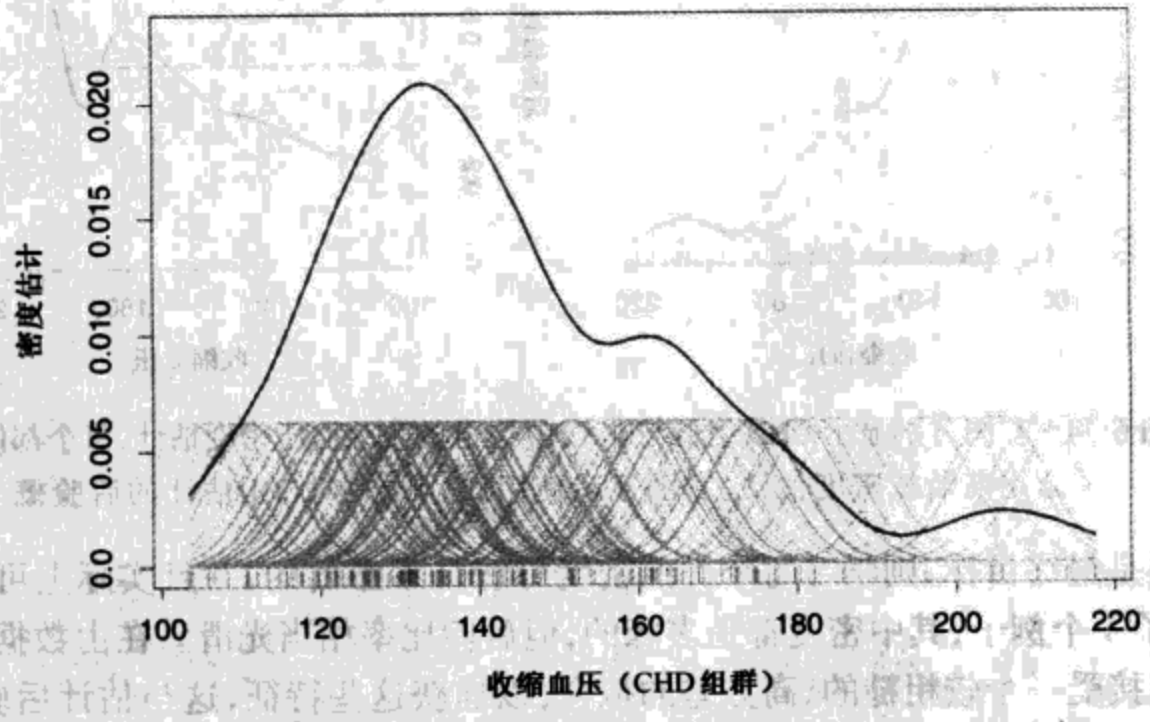


图 6.13 收缩血压的核密度估计(CHD 组群)。每个点上的密度估计是该点上每个核的平均贡献。我们已经将核缩小了一个因子10,以便于显示

这是样本经验分布 $\hat{F}$ 与 ϕ_λ 的卷积。分布 $\hat{F}(x)$在每个观测 x_i 上放置质量 $1/N$,并且是颠簸的;在 $\hat{f}_X(x)$,通过添加独立的高斯噪声到每个观测 x_i,我们有光滑的 $\hat{F}$。

Parzen 密度估计等价于局部平均,并且沿着局部回归的思路已经提出一些改进[关于密度的对数尺度,见 Loader(1999)]。这里不再深入讨论。在 $\mathbb{R}^p$ 中,高斯密度估计的一个自然拓广是在式(6.23)中使用高斯积核:

$$\hat{f}_X(x_0) = \frac{1}{N(2\lambda^2\pi)^{\frac{p}{2}}} \sum_{i=1}^{N} e^{-\frac{1}{2}(||x_i - x_0||/\lambda)^2} \tag{6.24}$$

6.6.2 核密度分类

可以利用非参数密度估计，直接使用贝叶斯定理进行分类。假定对于 J 类问题，我们分别在每个类上拟合非参数密度估计 $\hat{f}_j(X), j = 1, \cdots, J$，并且还有每个类的先验 $\hat{\pi}_j$ 的估计（通常是样本的比例），则

$$\hat{\Pr}(G = j | X = x_0) = \frac{\hat{\pi}_j \hat{f}_j(x_0)}{\sum_{k=1}^{J} \hat{\pi}_k \hat{f}_k(x_0)} \tag{6.25}$$

图 6.14 使用该方法估计 CHD 流行，进行心脏病风险因素研究，并且应当与图 6.12 的左图比较。它们的主要不同出现在图 6.14 右图的高 SBP 区域。在该区域，两个类的数据都是稀疏的，并且由于高斯核密度估计使用度量核，在这些区域的密度估计偏低，并且质量很差（高方差）。局部逻辑斯缔回归模型(6.20)使用具有 k-NN 带宽的三次方核；这有效地加宽了该区域的核，并且使用局部线性假设光滑了估计（在分对数尺度）。

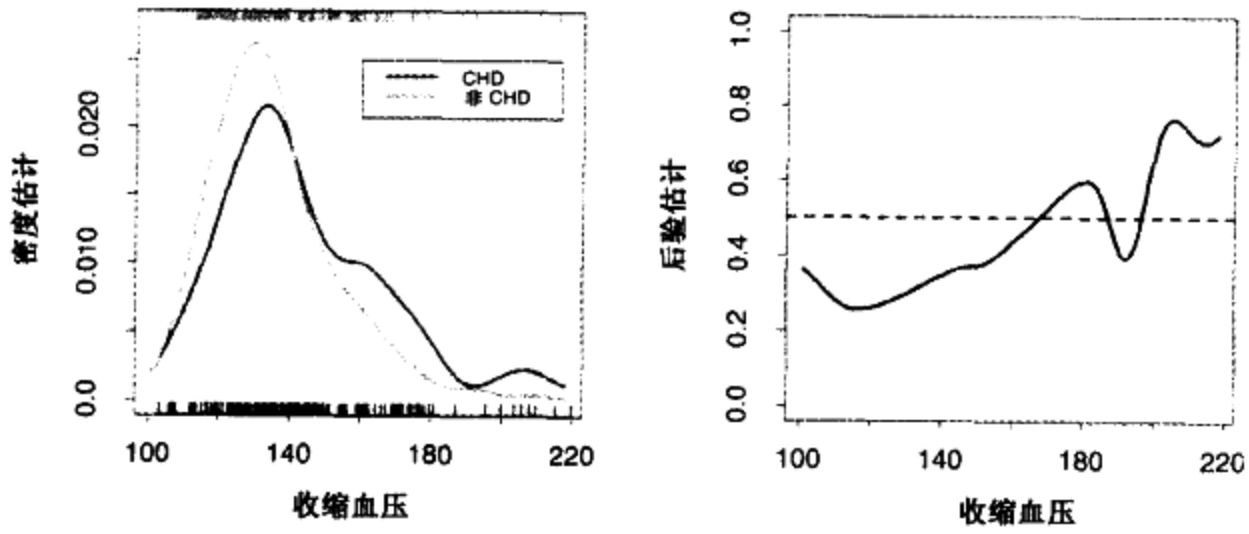


图 6.14 左图分别显示 CHD 组群与非 CHD 组群的收缩血压密度估计，每个都使用高斯核密度估计。右图显示使用式(6.25)对CHD估计的后验概率

如果分类是最终目标，则学习每个类的密度可能是不必要的，并且实际上可能产生误导。图 6.15 显示了一个例子，其中密度都是多峰的，但后验比率相当光滑。在由数据学习密度时，我们可能决定接受一个较粗糙的、高方差的拟合，以捕获这些特征，这与估计后验概率的目的无关。事实上，如果分类是最终目标，我们只需要估计相当靠近判定边界的后验（对于两个类，这就是集合 $\{x \mid \Pr(G = 1 \mid X = x) = 1/2\}$）。

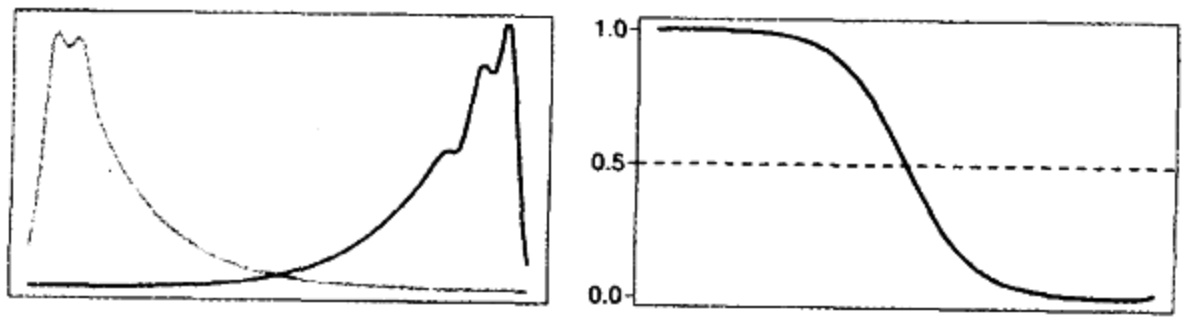


图 6.15 总体类密度可能具有有趣的结构（左），它在后验概率形成后消失（右）

6.6.3 朴素贝叶斯分类法

这是一种多年来一直流行的技术,尽管它是朴素的(也称"傻瓜贝叶斯")。当特征空间的维数 p 很高,使得密度估计很难时,它特别合适。朴素贝叶斯模型假定,给定类 $G=j$,特征 X_k 是独立的:

$$f_j(X)=\prod_{k=1}^{p} f_{jk}(X_k) \tag{6.26}$$

尽管该假定一般并不成立,但是它确实大大简化了估计:

- 单个类条件边缘密度 f_{jk} 可以使用一维核密度估计分别地估计。事实上,这是原始的朴素贝叶斯过程的拓广。原始的朴素贝叶斯过程使用一元高斯分布表示这些边缘。
- 如果 X 的分量 X_j 是离散的,则可以用适当的直方图估计。这为特征向量中混合变量类型提供了一种无缝方法。

尽管这些假定是非常理想化的,但是朴素贝叶斯分类法的性能通常远胜过更复杂的分类法。原因涉及图 6.15:尽管单个类密度的估计可能是有偏的,但是这种偏倚可能对后验概率影响不大,特别是在靠近判定边界的区域。事实上,为了获得"朴素"假设赢得的方差减少,问题可能能够承受一定的偏倚。

从式(6.26)出发,我们可以导出分对数变换(使用类 J 作为基):

$$\begin{aligned}\operatorname{logit}\frac{\Pr(G=\ell|X)}{\Pr(G=J|X)} &= \log\frac{\pi_\ell f_\ell(X)}{\pi_J f_J(X)}\\ &= \log\frac{\pi_\ell\prod_{k=1}^{p} f_{\ell k}(X_k)}{\pi_J\prod_{k=1}^{p} f_{Jk}(X_k)}\\ &= \log\frac{\pi_\ell}{\pi_J}+\sum_{k=1}^{p}\log\frac{f_{\ell k}(X_k)}{f_{Jk}(X_k)}\\ &= \alpha_\ell+\sum_{k=1}^{p} g_{\ell k}(X_k)\end{aligned} \tag{6.27}$$

这具有广义加法模型的形式,将在第 9 章详细介绍。该模型以相当不同的方法拟合;这些不同在习题(6.9)中研究。朴素贝叶斯与广义加法模型之间的联系类似于线性判别和逻辑斯缔回归之间的联系(见第 4.4.4 节)。

6.7 径向基函数和核

在第 5 章,函数用基函数的展开式表示:$f(x)=\sum_{j=1}^{M}\beta_j h_j(x)$。使用基展开式灵活建模的技术包括选择一族适当的基函数,然后通过选择、正则化或二者共同控制表示的复杂性。有些基函数族具有局部定义的函数;例如,B 样条局部地定义在 $\mathbb{R}$ 上。如果在特定区域需要更多的灵活性,则该区域需要用更多的基函数表示(在 B 样条中,转换成更多的纽结)。$\mathbb{R}$ 局部基函数的张量积产生局部于 $\mathbb{R}^p$ 的基函数。并非所有的基函数都是局部的——例如,样条的

截尾幂基，神经网络中使用的 S 型基函数(见第 11 章)。复合函数仍然能够表现局部行为，因为系数的特定符号和值导致全局影响的抵消。例如，对于相同的函数空间，截尾幂基具有一个等价的 B 样条基；在此情况下恰是抵消。

通过在局部于目标点 x_0 的区域拟合简单的模型，核方法获得了灵活性。局部性通过加权核 K_λ 实现，并且个体观测得到权 $K_\lambda(x_0, x_i)$。

通过将核函数 $K_\lambda(\xi, x)$处理为基函数，径向基函数将这些思想结合在一起。这导致模型

$$\begin{aligned} f(x) &= \sum_{j=1}^{M} K_{\lambda_j}(\xi_j, x)\beta_j \\ &= \sum_{j=1}^{M} D\left(\frac{||x-\xi_j||}{\lambda_j}\right)\beta_j \end{aligned} \tag{6.28}$$

其中，每个基元素通过位置或原型参数 ξ_j 和缩放参数 λ_j 指定。D 的通常选择是标准高斯密度函数。存在一些学习参数$\{\lambda_j, \xi_j, \beta_j\}(j = 1, \cdots, M)$的方法。为简单起见，我们将关注回归的最小二乘法，并使用高斯核。

- 关于所有的参数，优化平方和：

$$\min_{\{\lambda_j,\xi_j,\beta_j\}_1^M} \sum_{i=1}^{N}\left(y_i - \beta_0 - \sum_{j=1}^{M}\beta_j \exp\left\{-\frac{(x_i-\xi_j)^T(x_i-\xi_j)}{\lambda_j^2}\right\}\right)^2 \tag{6.29}$$

  通常称该模型为 RBF 网络，它是第 11 章讨论的 S 型神经网络的替代方案；ξ_j 和 λ_j 起权的作用。该标准是非凸的，具有多个局部极小，而优化算法类似于神经网络使用的那些算法。

- 分别由 β_j 估计$\{\lambda_j, \xi_j\}$。给定前者，后者的估计是简单的最小二乘方问题。通常，核参数 λ_j 和 ξ_j 用无指导的方法选择，仅使用 X 的分布。方法之一是用高斯混合密度模型拟合训练 x_i，产生中心 ξ_j 和缩放 λ_j。其他更特殊的方法使用聚类确定原型 ξ_j，并将$\lambda_j = \lambda$作为超参数处理。这些方法的明显缺点是条件分布 $\Pr(Y|X)$，特别是 $E(Y|X)$并未说明这种做法的关注点在哪里。在积极的一面，它们的实现非常简单。

尽管减小参数集并假定 $\lambda_j = \lambda$ 取常数值看上去吸引人，但它可能具有副作用——产生洞——$\mathbb{R}^p$ 的区域，那里核都没有适当的支集，如图 6.16(上图)所示。重新对径向基函数标准化，

$$h_j(x) = \frac{D(||x-\xi_j||/\lambda)}{\sum_{k=1}^{M} D(||x-\xi_k||/\lambda)} \tag{6.30}$$

将避免该问题(下图)。

$\mathbb{R}^p$ 上的 Nadaraya-Watson 核回归估计(6.2)可以看做重新正规化的径向基函数的展开式：

$$\begin{aligned} \hat{f}(x_0) &= \sum_{i=1}^{N} y_i \frac{K_\lambda(x_0, x_i)}{\sum_{i=1}^{N} K_\lambda(x_0, x_i)} \\ &= \sum_{i=1}^{N} y_i h_i(x_0) \end{aligned} \tag{6.31}$$

其中，基函数 h_i 位于每个观测和系数y_i；即，$\xi_i = x_i$，$\hat{\beta}_i = y_i$，$i = 1, \cdots, N$。

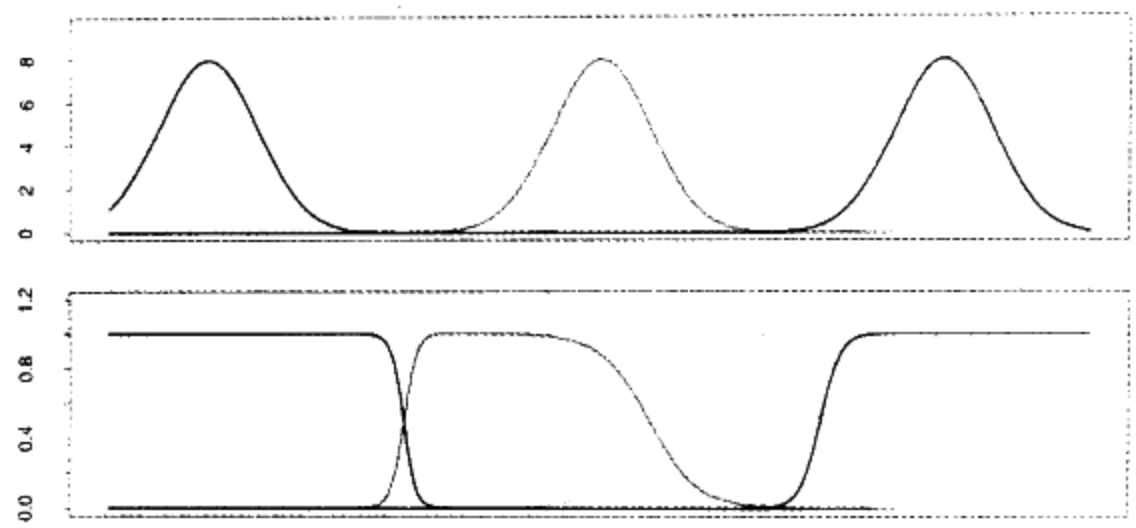


图 6.16 IR 上具有固定宽度的高斯径向基函数可能产生洞(上图)。重新对高斯径向基函数标准化避免了该问题,并在某些方面产生类似于 B 样条的基函数(见彩页)

6.8 密度估计和分类的混合模型

对于密度估计,混合模型是一种有用的工具,并且可以看做某种类型的核方法。高斯混合模型具有如下形式:

$$f(x)=\sum_{m=1}^{M}\alpha_m\phi(x;\mu_m,\mathbf{\Sigma}_m) \tag{6.32}$$

具有混合比例 α_m, $\sum_m \alpha_m = 1$,并且每个高斯密度具有均值 μ_m 和协方差矩阵 $\mathbf{\Sigma}_m$。一般地,混合模型可以用任意支密度取代式(6.32) 中的高斯密度:迄今为止,高斯混合模型最流行。

通常,参数用极大似然拟合,使用第 8 章介绍的 EM 算法。一些特殊情况是:

- 如果协方差矩阵限制为标量:$\mathbf{\Sigma}_m = \sigma_m\mathbf{I}$,则式(6.32) 具有径向基展开式形式。
- 此外, 如果 $\sigma_m = \sigma > 0$ 是固定的,并且 $M \uparrow N$,则对式(6.32)的极大似然估计逼近核密度式(6.22),其中 $\hat{\alpha}_m = 1/N$ 而 $\hat{\mu}_m = x_m$。

使用贝叶斯定理,每个类上分离的混合密度导致产生 $\Pr(G|X)$ 的灵活的模型,这将在第 12 章详细讨论。

图 6.17 显示混合模型在心脏病风险因素研究中的应用。上面一行分别是关于非 CHD 和 CHD 组群的年龄(Age)的直方图,而组合在右边。使用组合数据,我们拟合两个分量的形如式(6.32)的混合模型,其中不限制(标量)$\mathbf{\Sigma}_1$ 和 $\mathbf{\Sigma}_2$ 必须相等。拟合通过 EM 算法(见第 8 章)完成:注意, 该过程并不使用 CHD 标号知识。结果估计是:

$$\begin{array}{lll}\hat{\mu}_1 = 36.4 & \hat{\Sigma}_1 = 157.7 & \hat{\alpha}_1 = 0.7\\ \hat{\mu}_2 = 58.0 & \hat{\Sigma}_2 = 15.6 & \hat{\alpha}_2 = 0.3\end{array}$$

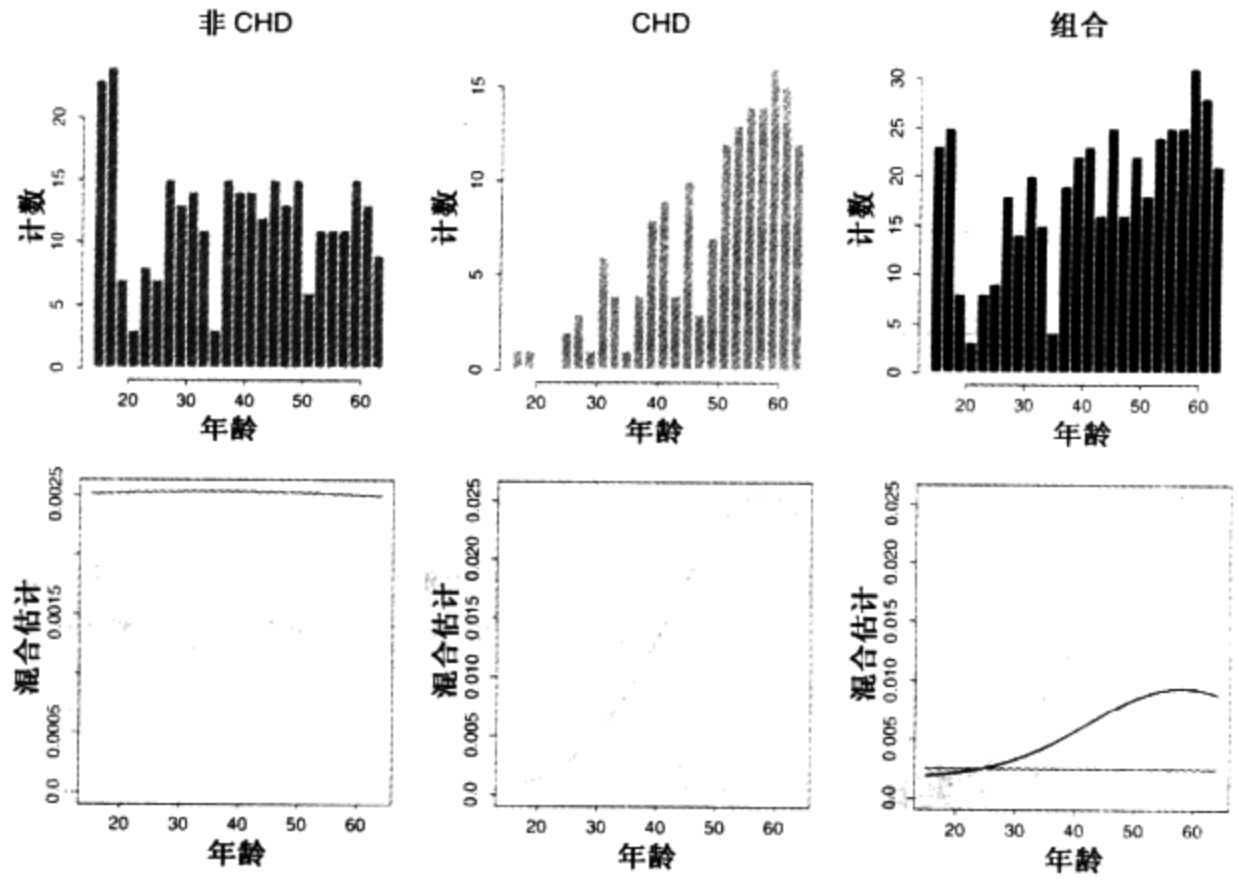


图 6.17 混合模型在心脏病风险因素研究中的应用。上行:分别是关于非 CHD 和 CHD群的年龄(Age)的直方图和组合的年龄直方图。下行:高斯混合模型的估计支密度(左,中);右下:估计支密度(绿色和红色),以及估计的混合密度(蓝色)。红色密度具有很大的标准差,并近似于均匀密度(见彩页)

支密度 $\phi(\hat{\mu}_1, \hat{\mathbf{\Sigma}}_1)$ 和 $\phi(\hat{\mu}_2, \hat{\mathbf{\Sigma}}_2)$ 在左下图和中下图显示。右下图显示这些支密度(绿色和红色),以及估计的混合密度(蓝色)。

混合模型还提供了观测 i 属于分量 m 的概率估计:

$$\hat{r}_{im} = \frac{\hat{\alpha}_m \phi(x_i; \hat{\mu}_m, \hat{\Sigma}_m)}{\sum_{k=1}^{M} \hat{\alpha}_k \phi(x_i; \hat{\mu}_k, \hat{\Sigma}_k)} \tag{6.33}$$

其中,x_i 是本例中的 Age。假定我们限定每个值 $\hat{r}_{i2}$,从而定义 $\hat{\delta}_i = I(\hat{r}_{i2} > 0.5)$。则可以将每个观测的 CHD 分类与混合模型进行比较,如下:

		混合模型	
		$\hat{\delta} = 0$	$\hat{\delta} = 1$
CHD	NO	232	70
	Yes	76	84

尽管混合模型不使用 CHD 标号,但它做了相当好的工作,发现了两个 CHD 子组群。线性逻辑斯缔回归使用 CHD 作为响应,当使用极大似然拟合这些数据时,达到相同的误差率(32%,见第 4.4 节)。

6.9 计算考虑

核与局部回归和密度估计是基于内存的方法:模型是整个训练数据集,并且拟合在求值和

预测时进行。对于一些实时应用,可能导致这类方法不切实际。

除非过分简化(如平方核),拟合单个观测 x_0 的计算开销是 $O(N)$flops(浮点运算)。通过比较,M 个基函数的展开式每次求值的开销为 $O(M)$,而典型地,$M \sim O(\log N)$。基函数方法初始化的开销至少是 $O(NM^2 + M^3)$。

核方法的光滑参数 λ 通常离线确定。例如使用交叉验证,开销为 $O(N^2)$flops。

局部回归的通常实现,如 S-PLUS 中的 loess 函数和 locfit 过程(Loader,1999),使用三角割分方案减少计算量。它们在 M 个精心选取的位置上计算拟合[$O(NM)$],然后使用混合技术在其他位置进行插值拟合[每次求值 $O(M)$]。

文献注释

核方法有大量文献,我们并不试图加以概述。我们只是指出少量好的参考文献,它们本身包含了大量文献。Loader(1999)给出局部回归和似然的详尽叙述,并且还介绍了拟合这些模型的软件发展情况。Fan 和 Gijbels(1996)从理论层面讨论了这些模型。Hastie 和 Tibshirani(1990)在加法模型的背景下讨论了局部回归。Silverman(1986)和 Scott(1992)给出了很好的密度估计综述。

习题

6.1 证明具有固定度量带宽 λ 和高斯核的 Nadaraya-Watson 核光滑是可微的。对于Epanechnikov 核,怎么样? 对于具有自适应的最近邻带宽 $\lambda(x_0)$的 Epanechnikov 核,又如何?

6.2 证明对于局部线性回归,$\sum_{i=1}^{N}(x_i - x_0)l_i(x_0) = 0$。定义 $b_j(x_0) = \sum_{i=1}^{N}(x_i - x_0)^j l_i(x_0)$。证明对于任意次的局部多项式回归(包括局部常数),$b_0(x_0) = 1$。证明对于 k 次局部多项式回归,对于所有的 $j \in \{1,2,\cdots, k\}$,$b_j(x_0) = 0$。对于偏倚,这蕴涵什么?

6.3 证明 $\| l(x) \|$(见第 6.1.2 节)随局部多项式的次数增加。

6.4 假定 p 个预测子 X 取自在 p 个均匀分布横坐标值上相对光滑的模拟曲线选样。记预测子的条件协方差矩阵为 $\mathrm{Cov}(X \mid Y) = \mathbf{\Sigma}$,并假定它随 Y 改变不大。讨论式(6.14)中度量的 Mahalanobis 选择 $\mathbf{A} = \mathbf{\Sigma}^{-1}$的特点。这与 $\mathbf{A} = \mathbf{I}$ 比较会怎么样? 你如何构造核 $\mathbf{A}$,它(a)距离度量中高频分量的权减少;(b)完全忽略它们?

6.5 证明拟合形如式(6.19)的局部常数多项式分对数模型相当于使用具有核权 $K_\lambda(x_0, x_i)$的 Nadaraya-Watson核光滑法,对每个类分别光滑二元响应指示符。

6.6 假定你只有拟合局部回归的软件,但可以指定哪些单项式包含在拟合中。怎样使用该软件拟合某些变量的变系数模型?

6.7 对于局部多项式回归,导出留一交叉验证残差平方和的表达式。

6.8 假定对于连续响应 Y 和预测子 X,我们使用多元高斯核估计法对 X, Y 的联合概率建模。注意,在此情况下,核应当是积核 $\phi_\lambda(X)\phi_\lambda(Y)$。证明由该估计导出的条件均值 $E(Y \mid X)$ 是 Nadaraya-Watson 估计。通过为连续的 X 和离散的 Y 的联合概率估计提供适当的核,将该结果推广到分类。

6.9 考察朴素贝叶斯模型(6.27)和广义加法逻辑斯缔回归模型在(a)模型假设和(b)估计方面的区别。如果所有的变量 X_k 是离散的,对应的 GAM 如何?

6.10 假设有 N 个样本,由模型 $y_i = f(x_i) + \varepsilon_i$ 产生,ε_i 是独立的同分布,具有均值 0 和方差 σ^2,并假设 x_i 是固定的(非随机的)。我们使用一个具有光滑参数 λ 的线性光滑法(局部回归、光滑样条等)估计 f。这样,拟合值向量由 $\hat{\mathbf{f}} = \mathbf{S}_\lambda \mathbf{y}$ 给定。对于 N 个输入值上的新响应,考虑样本内预测误差

$$\mathrm{PE}(\lambda) = \mathrm{E}\frac{1}{N}\sum_{i=1}^{N}(y_i^* - \hat{f}_\lambda(x_i))^2 \tag{6.34}$$

证明训练数据上的平均平方残差 $\mathrm{ASR}(\lambda)$是对 $\mathrm{PE}(\lambda)$的有偏估计(乐观的),而

$$C_\lambda = \mathrm{ASR}(\lambda) + \frac{2\sigma^2}{N}\mathrm{trace}(\mathbf{S}_\lambda) \tag{6.35}$$

是无偏的。

6.11 证明对于高斯混合模型(6.32),似然在 $+\infty$ 最大,并说明原因。

6.12 编写一个计算机程序,进行局部判别分析。在每个查询点 x_0,训练数据从加权核接收权 $K_\lambda(x_0, x_i)$,而线性判定边界(见第 4.3 节)的成分通过加权平均计算。在 zipcode 数据上试运行你的程序。对于 5 个预先选定的 λ 值序列显示训练和检验误差。zipcode 数据可以在本书的网站 www-stat.Stanford.edu/ElemStatLearn 得到。

第7章　模型评估与选择

7.1　引言

学习方法的泛化(generalization)性能涉及它在独立的检验数据上的预测能力。在实践中，性能评估尤为重要，因为它指导学习方法或模型的选择，并为我们提供最终选定模型的质的度量。

本章将介绍并解释性能评估的关键技术，并展示如何使用它们选择模型。我们以偏倚、方差和模型复杂性之间的相互影响的讨论开始。

7.2　偏倚、方差和模型复杂性

图7.1显示了评估学习方法泛化能力的重要问题。该图与图2.11完全相同；因为它太重要了，所以我们在此处再次选用它。首先考虑定量或区间标度响应。我们有一个目标变量 Y，一个输入向量 X 和一个已由训练样本估计的预测模型 $\hat{f}(X)$。度量 Y 和 $\hat{f}(X)$ 之间的误差的损失函数记做 $L(Y, \hat{f}(X))$。典型的选择是：

$$L(Y,\hat{f}(X)) = \begin{cases} (Y-\hat{f}(X))^2 & \text{均方误差} \\ |Y-\hat{f}(X)| & \text{绝对误差} \end{cases} \tag{7.1}$$

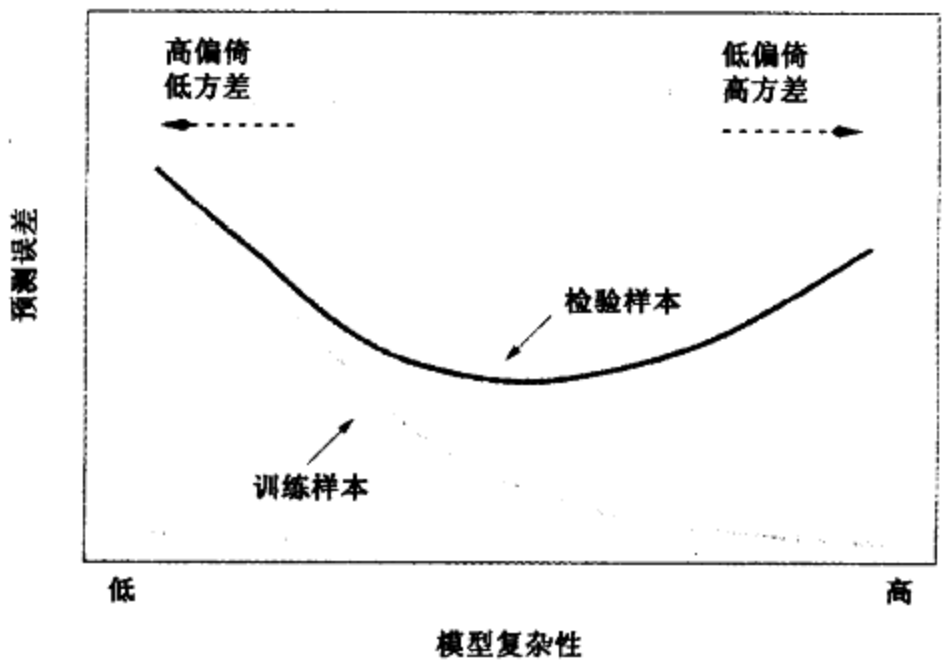


图7.1　检验样本和训练样本误差随模型的复杂性而变化

检验误差(test error)也称泛化误差(generalization error)，它是在独立的检验样本上的期望预测误差：

$$\mathrm{Err} = \mathrm{E}[L(Y,\hat{f}(X))] \tag{7.2}$$

其中，X 和 Y 都是从它们的联合分布（总体）中随机抽取的。注意，该期望对任意随机对象取平均值，包括产生 $\hat{f}$ 的训练样本的随机性。训练误差（training error）是在训练样本上的平均损失：

$$\overline{\text{err}} = \frac{1}{N}\sum_{i=1}^{N} L(y_i, \hat{f}(x_i)) \tag{7.3}$$

我们希望知道估计模型 $\hat{f}$ 的检验误差。随着模型越来越复杂，它能够适应更复杂的结构（偏倚减少），但估计误差有所增加（方差增大）。其间存在最佳模型复杂性，它产生最小检验误差。

遗憾的是，训练误差不是检验误差的一种好的估计，如图 7.1 所示。训练误差随模型的复杂性而减小，如果将模型的复杂性增加到足够大，典型地训练误差会减小到 0。然而，具有零训练误差的模型过分拟合训练数据，泛化性能通常很差。

对于在集合$\mathcal{G}$的 K 个值（为方便起见，记做 1，2，…，K）上取值的定性或分类响应 G，情况是类似的。通常，我们对概率 $p_k(X) = \Pr(G = k \mid X)$[或某单调变换 $f_k(X)$]建模，而后 $\hat{G}(X) = \arg\max_k \hat{p}_k(X)$。在某些情况下，如 1-最近邻分类（参见第 2 章和第 13 章），我们直接产生 $\hat{G}(X)$。典型的损失函数是：

$$L(G, \hat{G}(X)) = I(G \neq \hat{G}(X)) \quad \text{0-1 损失} \tag{7.4}$$

$$\begin{aligned} L(G, \hat{p}(X)) &= -2\sum_{k=1}^{K} I(G = k)\log \hat{p}_k(X) \\ &= -2\log \hat{p}_G(X) \quad \text{对数似然} \end{aligned} \tag{7.5}$$

该对数似然有时称为互熵（cross-entropy）损失或散离（deviance）。

检验误差仍然由期望误分类率 $\text{Err} = \text{E}[L(G, \hat{G}(X))]$或 $\text{Err} = \text{E}[L(G, \hat{p}(X))]$给出。训练误差是样本模拟，例如，对于该模型则是样本的对数似然

$$\overline{\text{err}} = \frac{-2}{N}\sum_{i=1}^{N} \log \hat{p}_{g_i}(x_i) \tag{7.6}$$

对数似然可以作为一般响应密度，如泊松分布、γ 分布、指数、对数正态分布和其他分布的损失函数。如果 $\Pr_{\theta(X)}(Y)$是 Y 的密度，被依赖于预测 X 的参数 $\theta(X)$标引，则

$$L(Y, \theta(X)) = -2 \cdot \log \Pr_{\theta(X)}(Y) \tag{7.7}$$

定义中的“2”使得高斯分布的对数似然损失与平方误差损失相匹配。

为便于阐述，在本章的其余部分我们将用 Y 和$f(X)$表示上述所有情况，因为我们主要关注定量响应（平方误差损失）。对于其他情况，适当的变换是显然的。

本章，我们介绍一些估计模型的检验误差曲线的方法。典型地，我们的模型具有调整参数或参数α，由此可以将预测记做 $\hat{f}_\alpha(x)$。调整参数改变模型的复杂性，并且我们希望找到极小化误差的α值；即产生图 7.1 误差曲线的极小值。这样，为简化符号，我们省略对 $\hat{f}(x)$的α依赖。

重要的是，要注意事实上我们有两个目标：

模型选择：估计不同模型的性能，以便选出（近似）最好的模型。

模型评估：已经选定最终的模型，估计它在新数据上的预测误差（泛化误差）。

如果我们的数据量很大，对于以上两个问题的最好方法是随机地将数据集分成三部分：训

练集、验证集(validation set)和检验集(test set)。训练集用于拟合模型,验证集用于估计模型选择的预测误差,检验集用于最终选定的模型泛化误差的评估。理想地,检验集应当保存在"保险库"中,直到数据分析结束时才拿出来用。如若不然,就要假定我们重复地使用检验集,选择具有最小检验集误差的模型。最终选定的模型的检验集误差将低于真实的检验误差,有时非常显著。

很难给出一个一般规则,指明三个部分各占多少观测,因为这依赖于数据的信噪比和训练样本的容量。一个典型的划分可能是 50%用于训练,而验证和检验各占 25%:

训练集	验证集	检验集

本章的方法是为没有足够的数据划分成三部分而设计的。同样,很难给出一个规则,说明多少数据是足够的;这依赖于基础函数的信噪比和拟合数据的模型的复杂性。

本章的方法或者解析地(AIC, BIC, MDL, SRM),或者通过有效样本重用(交叉验证或自助法)近似地实现验证。除了它们在模型选择中的使用外,我们还考察每种方法在多大程度上提供最终选定模型的检验误差的可靠估计。

在进入这些议题之前,我们先更详细地考察检验误差的性质和偏倚 - 方差权衡。

7.3　偏倚 - 方差分解

和第 2 章一样,如果假定 $Y = f(X) + \varepsilon$,其中 $\mathrm{E}(\varepsilon) = 0$, $\mathrm{Var}(\varepsilon) = \sigma_\varepsilon^2$,使用平方误差损失,可以导出在任意输入点 $X = x_0$ 上,回归拟合 $\hat{f}(X)$的期望预测误差表达式:

$$\begin{aligned}\mathrm{Err}(x_0) &= E[(Y - \hat{f}(x_0))^2 | X = x_0] \\ &= \sigma_\varepsilon^2 + [E\hat{f}(x_0) - f(x_0)]^2 + E[\hat{f}(x_0) - E\hat{f}(x_0)]^2 \\ &= \sigma_\varepsilon^2 + \mathrm{Bias}^2(\hat{f}(x_0)) + \mathrm{Var}(\hat{f}(x_0)) \\ &= \text{不可约的误差} + \text{偏倚}^2 + \text{方差}\end{aligned} \tag{7.8}$$

第一项是目标在其真正均值 $f(x_0)$附近的方差,除非 $\sigma_\varepsilon^2 = 0$,否则无论我们对 $f(x_0)$的估计多么好,也不能避免它。第二项是平方偏倚,是我们估计的平均值与真正均值之间的差异;最后一项是方差,是 $\hat{f}(x_0)$在其均值附近的期望平方差。通常,$\hat{f}$ 的模型越复杂,(平方)偏倚越小,但方差越大。

对于 k-最近邻回归拟合,这些表达式具有简洁形式:

$$\begin{aligned}\mathrm{Err}(x_0) &= E[(Y - \hat{f}_k(x_0))^2 | X = x_0] \\ &= \sigma_\varepsilon^2 + \left[f(x_0) - \frac{1}{k}\sum_{\ell=1}^{k} f(x_{(\ell)})\right]^2 + \sigma_\varepsilon^2/k\end{aligned} \tag{7.9}$$

这里,为简单起见,我们假定训练输入 x_i 是固定的,而y_i 是随机的。近邻数 k 与模型的复杂性逆相关。对于较小的 k, 估计 $\hat{f}_k(x)$可以更好地自适应于 $f(x)$。随 k 增加,偏倚[$f(x_0)$和 k-最近邻中的$f(x)$平均值之间差的平方]将增加,而方差则减少。

对于线性模型拟合 $\hat{f}_p(x) = \hat{\beta}^T x$,其中参数向量 β 具有 p 个分量,被最小二乘方拟合,我们有:

$$\begin{aligned} \mathrm{Err}(x_0) &= E[(Y-\hat{f}_p(x_0))^2|X=x_0] \\ &= \sigma_\varepsilon^2 + [f(x_0)-E\hat{f}_p(x_0)]^2 + \|\mathbf{h}(x_0)\|^2\sigma_\varepsilon^2 \end{aligned} \tag{7.10}$$

这里，$\mathbf{h}(x_0)$是线性权的 N 向量，产生拟合 $\hat{f}_p(x_0)=x_0^T(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$，因而 $\mathrm{Var}[\hat{f}_p(x_0)]=\|\mathbf{h}(x_0)\|^2\sigma_\varepsilon^2$。该方差随 x_0 变化，其(在样本值 x_i 上的)平均值是$(p/N)\sigma_\varepsilon^2$，从而样本内(in-sample)误差为：

$$\frac{1}{N}\sum_{i=1}^{N}\mathrm{Err}(x_i)=\sigma_\varepsilon^2+\frac{1}{N}\sum_{i=1}^{N}[f(x_i)-E\hat{f}(x_i)]^2+\frac{p}{N}\sigma_\varepsilon^2 \tag{7.11}$$

这里，模型复杂性直接与参数的个数 p 相关。

岭回归拟合 $\hat{f}_\alpha(x_0)$的检验误差 $\mathrm{Err}(x_0)$在形式上等价于式(7.10)，不同在于方差项中的线性权：$\mathbf{h}(x_0)=\mathbf{X}(\mathbf{X}^T\mathbf{X}+\alpha\mathbf{I})^{-1}x_0$。偏倚项也不同。

对于诸如岭回归这样的线性族，我们可以将偏倚进一步分解。设 β_* 表示对 f 的最佳拟合线性逼近：

$$\beta_*=\arg\min_\beta \mathrm{E}\left(f(X)-\beta^TX\right)^2 \tag{7.12}$$

这里，期望是用关于输入变量 X 的分布来取的。则可以将平方偏倚写成：

$$\begin{aligned} E_{x_0}\left[f(x_0)-E\hat{f}_\alpha(x_0)\right]^2 &= E_{x_0}\left[f(x_0)-\beta_*^Tx_0\right]^2+E_{x_0}\left[\beta_*^Tx_0-\mathrm{E}\hat{\beta}_\alpha^Tx_0\right]^2 \\ &= \mathrm{Ave}[\text{模型偏倚}]^2+\mathrm{Ave}[\text{估计偏倚}]^2 \end{aligned} \tag{7.13}$$

右端第一项是平均平方模型偏倚(model bias)，最佳拟合线性逼近与真实函数之间的误差。第二项是平均平方估计偏倚(estimation bias)，平均估计 $\mathrm{E}(\hat{\beta}^Tx_0)$与最佳线性拟合逼近之间的误差。

对于使用一般最小二乘方的线性模型拟合，估计偏倚为 0。对于受限的拟合，如岭回归，它是正的，而我们用它来换取方差的减小。模型偏倚的减少只能通过包含模型中变量的交叉项和变换，将线性模型类扩大到更丰富的模型类来实现。

图 7.2 显示了偏倚－方差权衡的策略。对于线性模型，模型空间是来自 p 个输入预测的所有线性预测的集合，而标记“最近拟合”的黑点是 β_*^Tx。蓝色阴影区域指示误差 σ_ε，通过它，我们可以看到训练样本的真实值。

图中还显示了最小二乘方拟合的方差，由中心在标记为“总体最近拟合”的黑点处的黄色圆圈指出。现在，如果用较少的预测子拟合模型，或通过将系数向 0 收缩对系数正则化，我们将得到图中显示的“收缩的拟合”。该拟合还有估计偏倚，因为它不是模型空间的最近拟合。另一方面，它具有较小的方差。如果方差的减少能够超过(平方)偏倚的增加，那么这是值得的。

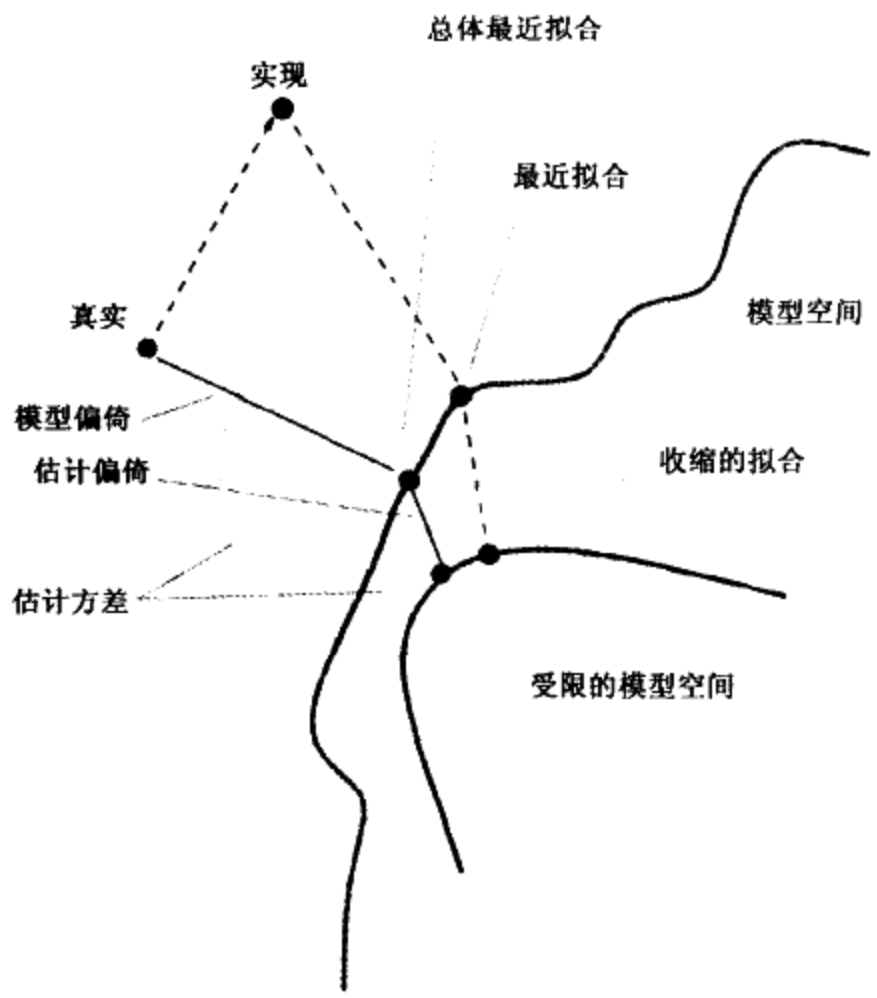


图 7.2　偏倚和方差变化示意图。模型空间是来自模型的所有可能预测的集合，“最近拟合”用黑点表示并加以标记。图中显示了与真实值的模型偏倚以及方差，由以标记为“总体最近拟合”的黑点为中心的黄色大圆指出。图中还显示了收缩或正则化拟合，它有额外的估计偏倚，但由于方差的减小，它具有较小的预测误差（见彩页）

7.3.1　例：偏倚 – 方差权衡

图 7.3 显示了两个模拟例子的偏倚 – 方差权衡。有 50 个观测，20 个预测子，均匀地分布在超立方体$[0, 1]^{20}$上。情况如下：

图 7.3 的左列图：如果 $X_1 \leqslant 1/2$，则 Y 等于 0；而当 $X_1 > 1/2$ 时，Y 等于 1，并且使用 k-最近邻。

图 7.3 的右列图：若 $\sum_{j=1}^{10} X_j$ 大于 5，则 Y 等于 1，否则 Y 等于 0，并且使用容量为 p 的最佳子集线性回归。

上两幅图是使用平方误差损失的回归；下两幅图是使用 0-1 损失的分类。这些图显示预测误差（红色）、平方偏倚（绿色）和方差（蓝色），所有的计算都是在一个很大的检验样本上进行的。

对于回归问题，偏倚和方差相加，产生预测误差曲线。对于 k-最近邻，大约在 $k = 5$ 时曲线取最小值；而对于线性模型，$p \geqslant 10$ 时取最小值。对于分类损失（下两幅图）可以看到一些有趣的现象。偏倚和方差曲线与上两幅图一样，而现在预测误差与误分类率有关。我们看到，预测误差不再是平方偏倚与方差的和。对于 k-最近邻分类，随近邻数增加到 20，尽管平方偏倚还在上升，但预测误差将降低或不变。对于线性模型分类，与回归一样，$p \geqslant 10$ 时取最小值，但在 $p = 1$ 上模型的改进则更加引人注目。我们看到偏倚和方差在决定预测误差时似乎相互影响着。

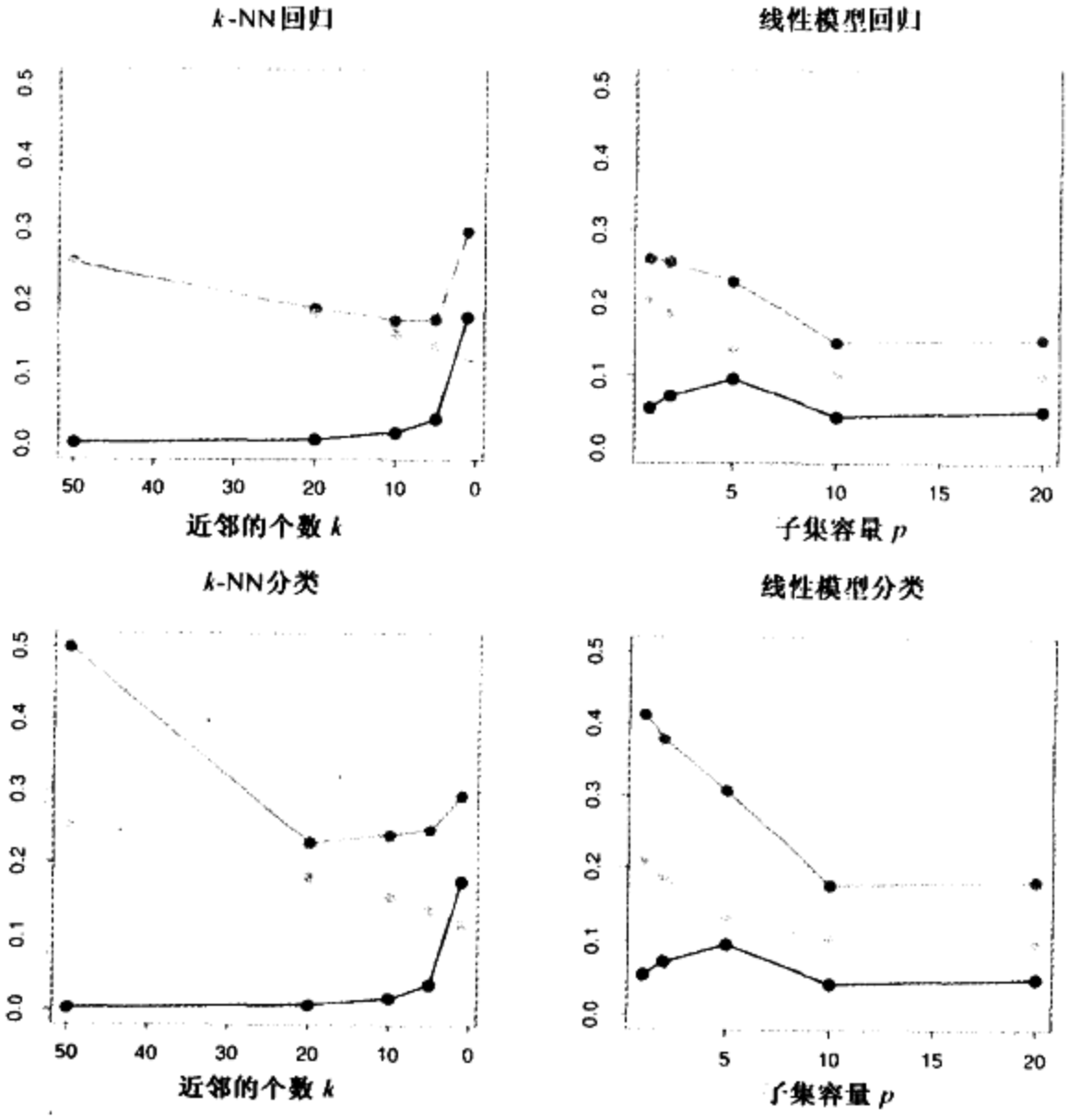


图 7.3 一个模拟例子的预测误差(红色)、平方偏倚(绿色)和方差(蓝色)。上两幅图是具有平方误差损失的回归,下两幅图是具有0-1损失的分类。模型是k-最近邻(左)和容量为p的最佳子集回归(右)。方差和偏倚曲线在回归和分类中是相同的,但预测误差曲线不同(见彩页)

为什么会出现这种情况?对第一种现象有一个简单的解释。假设在给定的输入点,类 1 的真正概率是 0.9,而我们估计的期望值是 0.6。那么平方偏倚$(0.6-0.9)^2$是相当大的,但由于我们做了正确的判断,使得预测误差为 0。换句话说,使得我们在判定边界正确一侧的估计误差并无妨碍。习题 7.2 分析了这种现象,并展示了偏倚和方差之间的交互作用。

总体上看,0-1 损失与平方误差损失的偏倚 - 方差权衡表现形式有所不同。这意味:在两种情况下,调整参数的最佳选择可能很不相同。正如以下各节所述,调整参数的选择应当基于预测误差的估计。

7.4 训练误差率的乐观性

典型地,训练误差率

$$\overline{\text{err}} = \frac{1}{N}\sum_{i=1}^{N} L(y_i, \hat{f}(x_i)) \tag{7.14}$$

将小于实际误差 $\text{Err} = E[L(Y, \hat{f}(X))]$,因为相同的数据被用于拟合方法和评估它的误差。通

常，拟合方法适应于训练数据，因此，表面上的或训练的误差$\overline{\text{err}}$将是泛化误差 Err 的过分乐观估计。

部分差异在于计算点出现在何处。Err 是一种样本外(extra-sample)误差，因为检验特征向量不必与训练特征向量一致。当我们关注的不是 Err，而是样本内(in-sample)误差

$$\text{Err}_{\text{in}} = \frac{1}{N}\sum_{i=1}^{N} \text{E}_{\mathbf{y}}\text{E}_{\mathbf{Y}^{\text{new}}} L(Y_i^{\text{new}}, \hat{f}(x_i)) \tag{7.15}$$

时，$\overline{\text{err}}$的乐观性就容易理解了。

记号 $\mathbf{Y}^{\text{new}}$指出，在每个训练点 x_i，$i = 1, 2, \cdots, N$，我们观测 N 个新的响应值。这里把乐观性(optimism)定义为 Err_{in}和训练误差$\overline{\text{err}}$之间的期望差：

$$\text{op} \equiv \text{Err}_{\text{in}} - \text{E}_{\mathbf{y}}(\overline{\text{err}}) \tag{7.16}$$

它一般是正的，因为作为预测误差的估计，$\overline{\text{err}}$通常是偏低的。

对于平方误差、0-1 和其他损失函数，可以很一般地证明：

$$\text{op} = \frac{2}{N}\sum_{i=1}^{N} \text{Cov}(\hat{y}_i, y_i) \tag{7.17}$$

其中 Cov 代表协方差。这样，导致$\overline{\text{err}}$过低地估计实际误差的总量取决于y_i 对它自己预测的影响程度。我们对数据拟合得越狠，$\text{Cov}(\hat{y}_i, y_i)$就越大，从而乐观性增加。习题 7.4 对平方误差损失证明了这一结果，其中$\hat{y}_i$ 是回归拟合值。对于 0-1 损失，$\hat{y}_i \in \{0,1\}$是 x_i 上的分类；对于熵损失，$\hat{y}_i \in [0,1]$是类 1 在 x_i 上的拟合概率。

概括地说，我们有重要关系：

$$\text{Err}_{\text{in}} = \text{E}_{\mathbf{y}}(\overline{\text{err}}) + \frac{2}{N}\sum_{i=1}^{N} \text{Cov}(\hat{y}_i, y_i) \tag{7.18}$$

如果$\hat{y}_i$ 是通过一个有 d 个输入或基函数的线性拟合得到的，则这个表达式可以简化。例如，对于加法误差模型 $Y = f(X) + \epsilon$，

$$\sum_{i=1}^{N} \text{Cov}(\hat{y}_i, y_i) = d\sigma_\epsilon^2 \tag{7.19}$$

从而

$$\text{Err}_{\text{in}} = \text{E}_{\mathbf{y}}\overline{\text{err}} + 2 \cdot \frac{d}{N}\sigma_\epsilon^2 \tag{7.20}$$

乐观性随我们使用的输入或基函数的个数 d 而呈线性增加，但随训练样本容量的增加而降低。对于其他误差模型，如二值数据和互熵损失，式(7.20)近似地成立。

估计预测误差一个明显的方法是估计乐观性，然后把它加到训练误差率$\overline{\text{err}}$中去。下一节介绍的方法——AIC、BIC 和其他方法，对于在其参数上是线性的一类特殊估计，就是这样做的。

相比之下，本章后面将介绍的交叉验证和自助法直接估计样本外误差 Err。这些通用工具可以与任意损失函数，以及非线性的和自适应的拟合技术一起使用。

通常，并不直接关心样本内误差，因为未来的特征值多半不大可能与它们的训练集值一致。但是，对于模型之间的对比，样本内误差是方便的，并且通常会导致有效的模型选择。因为误差的相对(不是绝对的)大小是重要的。

7.5 样本内预测误差的估计

样本内误差估计的一般形式是：

$$\widehat{\mathrm{Err}}_{in} = \overline{\mathrm{err}} + \widehat{\mathrm{op}} \tag{7.21}$$

其中，$\widehat{\mathrm{op}}$是乐观性估计。

在平方误差损失下拟合 d 个参数时，可以使用式(7.20)，导致产生所谓的 C_p 统计量：

$$C_p = \overline{\mathrm{err}} + 2 \cdot \frac{d}{N}\hat{\sigma}_\varepsilon^2 \tag{7.22}$$

这里，$\hat{\sigma}_\varepsilon^2$ 是噪声方差的估计，是从低偏倚模型的均方误差得到的。使用这一准则，我们用一个正比于所使用基函数的个数的因子来调整训练误差。

当使用对数似然损失函数时，Akaike 信息准则是一个类似的、但更具一般性的 Err_{in} 的估计。它依赖于一个类似于式(7.20)的联系，该联系随 $N \to \infty$ 渐近地成立：

$$-2 \cdot \mathrm{E}[\log \Pr_{\hat{\theta}}(Y)] \approx -\frac{2}{N} \cdot \mathrm{E}[\mathrm{loglik}] + 2 \cdot \frac{d}{N} \tag{7.23}$$

这里，$\Pr_\theta(Y)$是 Y 的密度族(包括“真实”密度)，$\hat{\theta}$ 是 θ 的极大似然估计，而“loglik”是极大对数似然：

$$\mathrm{loglik} = \sum_{i=1}^{N} \log \Pr_{\hat{\theta}}(y_i) \tag{7.24}$$

例如，对于逻辑斯缔回归模型，使用二项式对数似然，我们有：

$$\mathrm{AIC} = -\frac{2}{N} \cdot \mathrm{loglik} + 2 \cdot \frac{d}{N} \tag{7.25}$$

对于高斯模型(假定方差 $\sigma_\varepsilon^2 = \hat{\sigma}_\varepsilon^2$ 已知)，AIC 统计量等价于 C_p，因而我们把它们统称为 AIC。

为了在模型选择中使用 AIC，我们在所考虑的模型集上简单地选择给出最小 AIC 的模型。对于非线性的和其他复杂的模型，需要用模型复杂度的某种度量来代替 d。我们将在第 7.6 节讨论。

给定一个由调整参数α标引的模型 $f_\alpha(x)$集，用$\overline{\mathrm{err}}(\alpha)$和 $d(\alpha)$分别表示模型的训练误差和参数的个数。然后，对这个模型集我们定义：

$$\mathrm{AIC}(\alpha) = \overline{\mathrm{err}}(\alpha) + 2 \cdot \frac{d(\alpha)}{N}\hat{\sigma}_\varepsilon^2 \tag{7.26}$$

函数 AIC(α)提供了检验误差曲线的一个估计，并且寻找使其极小化的调整参数$\hat{\alpha}$。我们最终选择的模型是 $f_{\hat{\alpha}}(x)$。注意，如果自适应地选择基函数，则式(7.19)不再成立。例如，如果总共有 p 个输入，并且选择具有 $d < p$ 个输入的最佳拟合线性模型，则乐观性将超过$(2d/N)\sigma_\varepsilon^2$。换言之，通过选择具有 d 个输入的最佳拟合模型，拟合的有效的参数个数将大于 d。

图 7.4 显示了第 5.2.3 节音素识别例子中 AIC 的运行情况。输入向量是元音读音的对数周期图，被量化到 256 个均匀分布的频率上。线性逻辑斯缔回归模型用于预测音素类，系数函数为 $\beta(f) = \sum_{m=1}^{M} h_m(f)\theta_m$，它是 M 个样条基函数的展开式。对任意给定的 M，自然三次样条基用于 h_m，并且有在频率范围内均匀选择的纽结（这样，$d(\alpha) = d(M) = M$）。对于熵和 0-1 损失，使用 AIC 选择基函数的个数将近似地极小化 Err(M)。

简单公式：

$$(2/N)\sum_{i=1}^{N}\mathrm{Cov}(\hat{y}_i, y_i) = (2d/N)\sigma_\epsilon^2$$

对具有加法误差和平方误差损失的线性模型完全成立，而对线性模型和对数似然则近似成立。特殊地，该公式对 0-1 损失并不一般成立（Efron，1986），尽管许多作者仍然在此情况下使用它（见图 7.4 中的右图）。

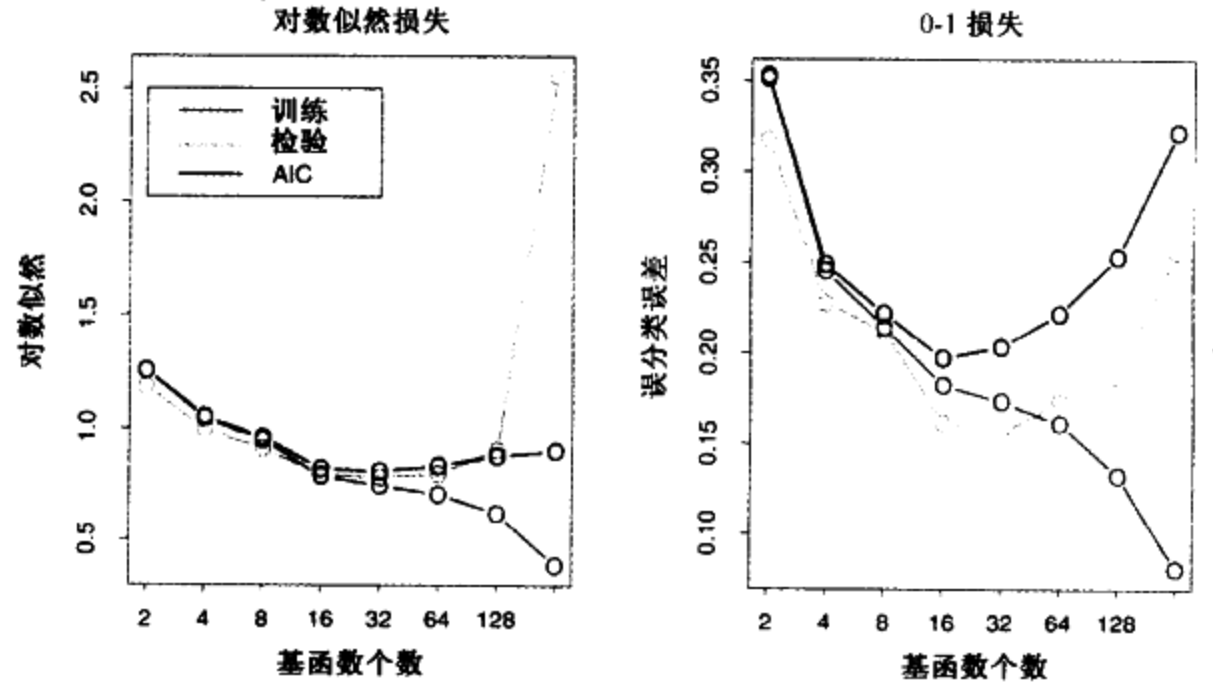


图 7.4　AIC 用于第 5.2.3 节音素识别例子的模型选择。逻辑斯缔回归系数函数 $\beta(f) = \sum_{m=1}^{M} h_m(f)\theta_m$ 被建模成 M 个样条基函数的展开式。在左图中，我们看到使用对数似然损失估计的 $\mathrm{Err}_{\mathrm{in}}$ 的 AIC 统计量。所包含的是基于一个独立检验样本的 Err 估计。除了极端地过分参数化情形外（对于 $N = 1000$ 个观测，$M = 256$ 个参数），它都做得很好。在右图中，对 0-1 损失做了同样的事情。尽管严格地说 AIC 公式不该在此处应用，但它应用于该情形却也很理想（见彩页）

7.6　有效的参数个数

“参数的个数”这一概念可以拓广，特别，可以拓广到使用正则化拟合的模型中。假设我们把结果 $y_1, y_2, \cdots y_N$ 放到一个向量 $\mathbf{y}$ 中，并且对预测 $\hat{\mathbf{y}}$ 也类似地处理，则线性拟合方法可以记做：

$$\hat{\mathbf{y}} = \mathbf{S}\mathbf{y} \tag{7.27}$$

其中，$\mathbf{S}$ 是 $N \times N$ 矩阵，依赖于输入向量 x_i，但不依赖于 y_i。线性拟合方法包括原空间或导出的基函数集上的线性回归和使用二次收缩的光滑方法，如岭回归和三次光滑样条。则有效的

参数个数(effective number of parameters)定义为:

$$d(\mathbf{S}) = \text{trace}(\mathbf{S}) \tag{7.28}$$

它是 $\mathbf{S}$ 的对角线元素之和。注意,如果 $\mathbf{S}$ 是到一个由 M 个特征生成的基集的正交投影矩阵,则 $\text{trace}(\mathbf{S}) = M$。这表明 $\text{trace}(\mathbf{S})$ 确实是代替 d 作为 C_p 统计量(7.22)中参数个数的正确的量(见习题 7.4 和习题 7.5)。我们在第 5.4.1 节中有关于 $d = \text{trace}(\mathbf{S})$ 更加详细的推导。

对于像神经网络那样的模型,我们用权衰减罚(正则化) $\alpha \sum_m w_m^2$ 对误差函数 $R(w)$ 极小化,有效的参数个数有如下形式:

$$d(\alpha) = \sum_{m=1}^{M} \frac{\theta_m}{\theta_m + \alpha} \tag{7.29}$$

其中,θ_m 是 Hessian 矩阵 $\partial^2 R(w)\partial w \partial w^T$ 的本征值。如果在解上对误差函数取二项式逼近,则式(7.29)符合式(7.28)(Bishop, 1995)。

7.7 贝叶斯方法和 BIC

和 AIC 一样,贝叶斯信息准则(BIC)可应用于通过对数似然极大化实现的拟合过程中,BIC 的一般形式是:

$$\text{BIC} = -2 \cdot \text{loglik} + (\log N) \cdot d \tag{7.30}$$

BIC 统计量(乘以 1/2)也是所谓的 Schwartz 准则(Schwartz, 1979)。

在高斯模型下,假设方差 σ_ε^2 已知,$-2 \cdot \text{loglik}$ 等于 $\sum_i (y_i - \hat{f}(x_i))^2 / \sigma_\varepsilon^2$(相差一个常数因子),对于平方误差损失,它等于 $N \cdot \overline{\text{err}} / \sigma_\varepsilon^2$。因此,我们有:

$$\text{BIC} = \frac{N}{\sigma_\varepsilon^2}\left[\overline{\text{err}} + (\log N) \cdot \frac{d}{N}\sigma_\varepsilon^2\right] \tag{7.31}$$

这样,BIC 与 AIC(C_p)成比例,因子 2 被 $\log N$ 所取代。假设 $N > e^2 \approx 7.4$,BIC 倾向于更多地惩罚复杂模型,在选择中偏爱较简单的模型。如同 AIC,σ_ε^2 通常由低偏倚模型的均方误差来估计。对于分类问题,使用互熵作为误差度量,多项式对数似然的使用导致了与 AIC 的相似联系。然而需要注意,误分类误差度量在 BIC 中并不升高,因为在任意概率模型下它并不对应于数据的对数似然。

尽管与 AIC 相似,但 BIC 动机却极其不同。它起源于使用贝叶斯方法进行模型选择,我们现在就讨论它。

假设有一个候选模型集 $\mathcal{M}_m$ ($m = 1, 2, \cdots, M$)和对应的模型参数 θ_m,我们希望在它们之中选择一个最佳模型。假定对于每个模型 $\mathcal{M}_m$ 的参数,有一个先验分布 $\Pr(\theta_m \mid \mathcal{M}_m)$,则给定模型的后验概率是:

$$\begin{aligned} \Pr(\mathcal{M}_m|\mathbf{Z}) &\propto \Pr(\mathcal{M}_m) \cdot \Pr(\mathbf{Z}|\mathcal{M}_m) \\ &\propto \Pr(\mathcal{M}_m) \cdot \int \Pr(\mathbf{Z}|\theta_m, \mathcal{M}_m)\Pr(\theta_m|\mathcal{M}_m)\mathrm{d}\theta_m \end{aligned} \tag{7.32}$$

其中,$\mathbf{Z}$ 表示训练数据 $\{x_i, y_i\}_1^N$。为了比较两个模型 $\mathcal{M}_m$ 和 $\mathcal{M}_\ell$,我们形成后验几率:

$$\frac{\Pr(\mathcal{M}_m|\mathbf{Z})}{\Pr(\mathcal{M}_\ell|\mathbf{Z})}=\frac{\Pr(\mathcal{M}_m)}{\Pr(\mathcal{M}_\ell)}\cdot\frac{\Pr(\mathbf{Z}|\mathcal{M}_m)}{\Pr(\mathbf{Z}|\mathcal{M}_\ell)} \tag{7.33}$$

如果几率大于 1,我们选择模型 m,否则选择 ℓ。最右边的量

$$\mathrm{BF}(\mathbf{Z})=\frac{\Pr(\mathbf{Z}|\mathcal{M}_m)}{\Pr(\mathbf{Z}|\mathcal{M}_\ell)} \tag{7.34}$$

叫做贝叶斯因子(Bayes factor),它是数据对后验几率的贡献。

典型地,我们假设模型上的先验是均匀的,使得 $\Pr(\mathcal{M}_m)$是常量。我们需要逼近 $\Pr(\mathbf{Z}|\mathcal{M}_m)$的某种方法。一种对式(7.32)中积分的拉普拉斯逼近,后随某些其他简化(Ripley,1996,第 64 页)给出:

$$\log\Pr(\mathbf{Z}|\mathcal{M}_m)=\log\Pr(\mathbf{Z}|\hat{\theta}_m,\mathcal{M}_m)-\frac{d_m}{2}\cdot\log N+O(1) \tag{7.35}$$

此处,$\hat{\theta}_m$ 是极大似然估计,d_m 是模型$\mathcal{M}_m$ 中自由参数的个数。如果定义损失函数为:

$$-2\log\Pr(\mathbf{Z}|\hat{\theta}_m,\mathcal{M}_m)$$

则它等价于式(7.30)的 BIC 准则。

这样,选择具有极小 BIC 的模型等价于选择具有最大(近似)后验概率的模型。但这种框架给予我们更多的好处。给定 BIC_m($m=1,2,\cdots,M$),如果对 M 个模型的集合计算 BIC 准则,那么可以把每个模型$\mathcal{M}_m$ 的后验概率估计为:

$$\frac{e^{-\frac{1}{2}\cdot\mathrm{BIC}_m}}{\sum_{\ell=1}^{M}e^{-\frac{1}{2}\cdot\mathrm{BIC}_\ell}} \tag{7.36}$$

这样,我们不仅可以估计最好的模型,而且可以评估所考虑模型的相关指标。

对于模型选择这一目的,AIC 和 BIC 之间没有明确的选择。作为选择准则,BIC 是渐近相容的。这里的意思是给定一个模型族,包括真实模型,当样本容量 $N\to\infty$ 时,BIC 选择正确模型的概率将趋向于 1。而对 AIC 情况则不是这样,当 $N\to\infty$ 时,AIC 倾向于选择过于复杂的模型。另一方面,对于有限的样本,BIC 选择的模型通常过于简单,因为它对复杂性有较大的罚。

7.8　最小描述长度

最小描述长度(MDL)方法给出了形式上与 BIC 方法完全相同的选择准则,但它源自于最优编码。首先我们回顾一下数据压缩的编码理论,然后将它应用于模型选择。

把我们的数据 z 想像为信息,想对它编码并发送给另外某个人(接收者)。我们把模型看做是对数据编码的方法,并且将选择最节俭的模型,即对于传输,它是最短的码。

首先假设要传输的信息是 z_1, z_2,$\cdots$,z_m。我们的码使用长度为 A 的有限字母表:例如,可以使用长度 $A=2$ 的二进制编码{0, 1}。这是一个有 4 种可能信息的例子,其二进制编码为:

信息	z_1	z_2	z_3	z_4
码	0	10	110	111

(7.37)

这种码是一种所谓的瞬间前缀码:任何一个码都不是另一个码的前缀,并且接收者(他知道全部可能的码)确切地知道什么时候信息被全部发送。我们的讨论就针对这样的瞬间前缀码。

可以使用式(7.37)中的编码,或者可以排列这些码,例如对于 z_1, z_2, z_3, z_4 使用码 110, 10,111,0。怎样决定使用哪一种编码呢? 这取决于我们将发送的每一条信息的频率。例如,如果发送 z_1 最频繁,则对 z_1 使用最短的码 0 是有意义的。使用这种策略(较短的码用于较频繁的信息)平均信息长度将比较短。

一般来说,如果信息发送的概率是 $\Pr(z_i)$ ($i = 1, 2, \cdots, 4$),则著名的香农定理告诉我们,应当使用长度 $l_i = -\log_2 \Pr(z_i)$ 的码,并且平均信息长度满足:

$$\mathrm{E}(\text{长度}) \geqslant -\sum \Pr(z_i) \log_2 (\Pr(z_i)) \tag{7.38}$$

上式右端也称分布 $\Pr(z_i)$ 的熵。当概率满足 $p_i = A^{-l_i}$ 时,不等式将变成等式。在我们的例子中,如果分别有 $\Pr(z_i) = 1/2, 1/4, 1/8, 1/8$,则式(7.37)中显示的编码是最优的并且能够达到熵的下界。

一般地,不能达到下界,但像哈夫曼编码方案这样的过程可以很接近该下界。注意,对于无限的信息集,熵可以用 $-\int \Pr(z) \log_2 \Pr(z) \mathrm{d}z$ 代替。

从这个结果我们发现:

为了传递具有概率密度函数 $\Pr(z)$ 的随机变量 z,我们需要大约 $-\log_2 \Pr(z)$ 位的信息。

以后我们改变记法:将 $\log_2 \Pr(z)$ 改为 $\log \Pr(z) = \log_e \Pr(z)$;这是为了方便,且仅引进了一个无关紧要的乘法常量。

现在,将这一结果应用于模型选择问题。我们有一个以 θ 为参数的模型 M 和包括输入、输出的数据 $\mathbf{Z} = (\mathbf{X}, \mathbf{y})$。令该模型下输出的(条件)概率是 $\Pr(\mathbf{y} \mid \theta, M, \mathbf{X})$,假设接收者知道全部输入,并且我们希望传送输出。那么传送输出所需要的信息长度是:

$$\text{长度} = -\log \Pr(\mathbf{y}|\theta, M, \mathbf{X}) - \log \Pr(\theta|M) \tag{7.39}$$

它是给定输入的目标值的对数概率。第二项是传送模型参数 θ 的平均码长,而第一项是传送模型和实际目标值之间的偏差的平均码长。例如,假设我们有单一目标 $y \sim N(\theta, \sigma^2)$,参数 $\theta \sim N(0,1)$,并且无输入(为了简化),则信息长度是:

$$\text{长度} = \text{常量} + \log \sigma + \frac{(y-\theta)^2}{\sigma^2} + \frac{\theta^2}{2} \tag{7.40}$$

注意,由于 y 比较集中于 θ 附近,所以较小的 σ 就是较短的信息长度。

MDL 原理表明:我们应当选择能够极小化式(7.39)的模型。我们把式(7.39)视为(负的)对数后验分布,从而极小化描述长度等价于极大化后验概率。因此,作为对对数后验概率逼近而导出的 BIC 规则,也可以看做是用极小描述长度进行(近似的)模型选择的策略。

注意,我们忽略了对随机变量 z 编码的精度。使用有限的编码长度,我们无法精确地对连续变量编码。但是如果在容差 δz 之内对 z 编码,则所需的信息长度就是区间 $[z, z+\delta z]$ 上的概率的对数;如果 δz 较小,则它将被 $\delta z \Pr(z)$ 很好地近似。由于 $\log \delta z \Pr(z) = \log \delta z + \log \Pr(z)$,这意味着我们恰好可以忽略常量 $\log \delta z$,并且使用 $\log \Pr(z)$ 作为信息长度的度量,就像我们上面所做的一样。

前面用于模型选择的 MDL 观点表明,我们应当选择具有极大后验概率的模型。然而,许多贝叶斯方法宁愿通过从后验分布中抽样来进行推理。

7.9　Vapnik-Chernovenkis 维

使用样本内误差估计的困难在于需要说明在拟合中使用的参数的个数(或复杂度)d。尽管在第 7.6 节引进的有效的参数个数对某些非线性模型是有用的,但它还不够一般。Vapnik-Chernovenkis(VC)理论提供复杂度的一般度量,并给出乐观性的相关界限。这里,我们给出该理论的简略回顾。

假设有一个函数类$\{f(x,\alpha)\}$,由参数向量α标引,$x\in \mathrm{IR}^p$。现在假设f是一个指示函数,即取值 0 或 1。如果$\alpha=(\alpha_0,\alpha_1)$并且$f$是线性指示函数$I(\alpha_0+\alpha_1^T x>0)$,则说类$f$的复杂度是参数的个数$p+1$看来是合理的。但是,如果$\alpha$是任意实数且$x\in \mathrm{IR}$,那么$f(x,\alpha)=I(\sin\alpha\cdot x)$会怎样?函数$\sin(50\cdot x)$显示在图 7.5 中。这是一个摆动很大的函数,随着频率α的增加变得更加粗糙,但它只有一个参数:尽管如此,断言它的复杂性低于$p=1$维上的线性指示函数$I(\alpha_0+\alpha_1 x)$看来并不合理。

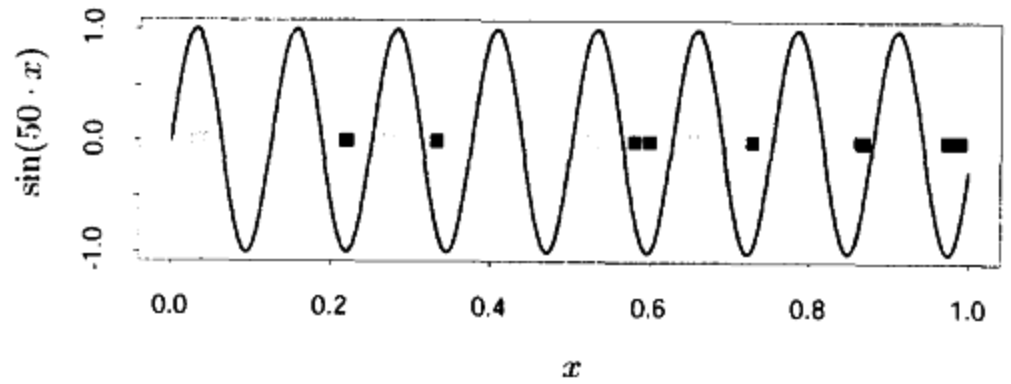


图 7.5　实线是函数$\sin(50\cdot x)$,$x\in[0,1]$。方块(实心的)和方块(空心的)刻画了相关联的指示器函数$I(\sin(\alpha x)>0)$怎样通过选择一个合适的高频率α才能够分散(分离)一个相当大的数目的点

Vapnik-Chernovenkis 维是一种通过估价一个函数类的成员可能有多大摆动来度量该函数类的复杂性的方法。

类$\{f(x,\alpha)\}$的 VC 维被定义成可以被$\{f(x,\alpha)\}$的成员分散的点的最大个数(在某种结构中)。

一个点集合被说成是被一个函数类分散,如果不管我们怎样对每个点赋一个二值标号,这个类的成员都能够正确地分散它们。

图 7.6 说明了平面中的线性指示函数的 VC 维是 3 而不是 4,因为 4 个点不能被一个直线集所分散。通常,p 维中的线性指示函数的 VC 维 $p+1$,它也是自由参数的个数。另一方面,还可以说明族$\sin(\alpha x)$有无限的 VC 维,如图 7.5 指出的。通过适当地选择α,任意点集都可以被这个类所分散(见习题 7.7)。

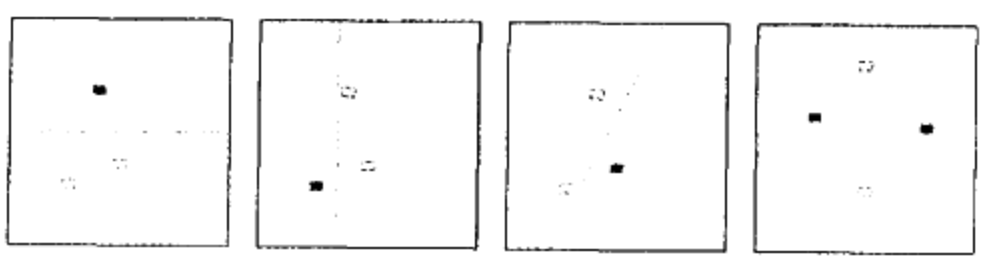


图 7.6　前三幅图显示了在平面中直线类可以分散 3 个点。最后一幅图显示平面中直线类无法分散4个点,因为没有直线能够将空心点置于线的一侧而将实心点置于另一侧。因此,平面上直线类的VC维是3。注意,非线性曲线类可以分散4个点,因此有大于3的VC维

至此,我们仅讨论了指示函数的 VC 维,但这可以扩展到实值函数上。实值函数类$\{g(x,\alpha)\}$的 VC 维被定义为指示类$\{I(g(x,\alpha)-\beta>0)\}$的 VC 维,其中 β 在 g 的值域上取值。

可以在构造样本内预测误差的估计中使用 VC 维,已有各种不同的可用结果。使用 VC 维的概念,可以证明在使用一个函数类时的训练误差的乐观性结果。这种结果的一个例子如下。如果我们使用 VC 维为 h 的函数类$\{f(x,\alpha)\}$拟合 N 个训练点,则在训练集上至少有概率$1-\eta$:

$$\begin{aligned}
\text{Err} &\leqslant \overline{\text{err}} + \frac{\epsilon}{2}\left(1+\sqrt{1+\frac{4\cdot\overline{\text{err}}}{\epsilon}}\right) \quad (\text{二类分类}) \\
\text{Err} &\leqslant \frac{\overline{\text{err}}}{(1-c\sqrt{\epsilon})_+} \quad (\text{回归}) \qquad (7.41) \\
&\text{其中} \quad \epsilon = a_1\frac{h[\log(a_2N/h)+1]-\log(\eta/4)}{N}
\end{aligned}$$

这些上界对所有成员 $f(x,\alpha)$同时成立,并且都取自于 Cherkassky 和 Mulier(1998)。他们建议值 $c=1$。对于回归,他们建议 $a_1=a_2=1$;而对于分类,他们没提出建议,但给出 $a_1=4, a_2=2$ 对应于最坏情况。上界表明乐观性随 h 增加而随 N 减少,与式(7.20)给出的 AIC 校正 d/N 的定性一致。然而,式(7.41)中的结果更强些:不是对每个固定函数 $f(x,\alpha)$给出期望的乐观性,它们是对全部函数 $f(x,\alpha)$给出可能的上界,因此允许在类中搜索。

Vapnik 结构风险极小化(structural risk minimization, SRM)方法拟合一个递增 VC 维 $h_1<h_2<\cdots$的嵌套模型序列,然后选择具有最小上界值的模型。

我们注意到像式(7.41)中这样的上界通常很宽松,但是如果相对的(非绝对的)检验误差的大小是重要的,这并不妨碍它们作为模型选择的好准则。这种方法的主要缺点在于计算函数类的 VC 维的困难。通常仅可以获得一个 VC 维的粗略上界,并且可能未必是适当的。结构风险极小化程序可以成功地实现的例子是支持向量分类,在第 12.2 节讨论。

7.9.1 例(续)

图 7.7 显示了使用 AIC、BIC 和 SRM 为图 7.3 的例子选择模型规模的结果。对于标有 KNN 的例子,模型标引α指邻域大小,而对于标有回归的例子,α指子集容量。使用每种选择方法(如 AIC),我们估计了最佳模型$\hat{\alpha}$,并找出了它在检验集上的真正预测误差 $\text{Err}(\hat{\alpha})$。对相同的训练集,我们计算了最好和最差的可能模型选择的预测误差 $\min_\alpha \text{Err}(\alpha)$和 $\max_\alpha \text{Err}(\alpha)$。盒图显示量

$$100\times\frac{\text{Err}(\hat{\alpha})-\min_\alpha \text{Err}(\alpha)}{\max_\alpha \text{Err}(\alpha)-\min_\alpha \text{Err}(\alpha)}$$

的分布,它表示使用所选择的模型相对于最佳模型的误差。对于线性回归模型,复杂性用特征的数量度量,它也是线性分类器的 VC 维。对于 k-最邻近,我们使用了量 N/k。这是对复杂性的粗略估计,我们不知道它是否相当于 VC 维。对于式(7.41)中的常量,我们使用了 $a_1=a_2=1$。SRM 的结果随不同的常量改变,该选择给出了最有利的结果。对于误分类误差,我们对最少限制模型(对于 KNN,$k=5$,因为 $k=1$ 导致 0 训练误差)使用 $\hat{\sigma}_\epsilon^2=[N/(N-d)]\cdot\overline{\text{err}}(\alpha)$。看来 AIC 准则在全部 4 种方案中都做得很好,尽管对 0-1 损失缺乏理论支持。BIC 也可以,而 SRM 的性能好坏掺半。

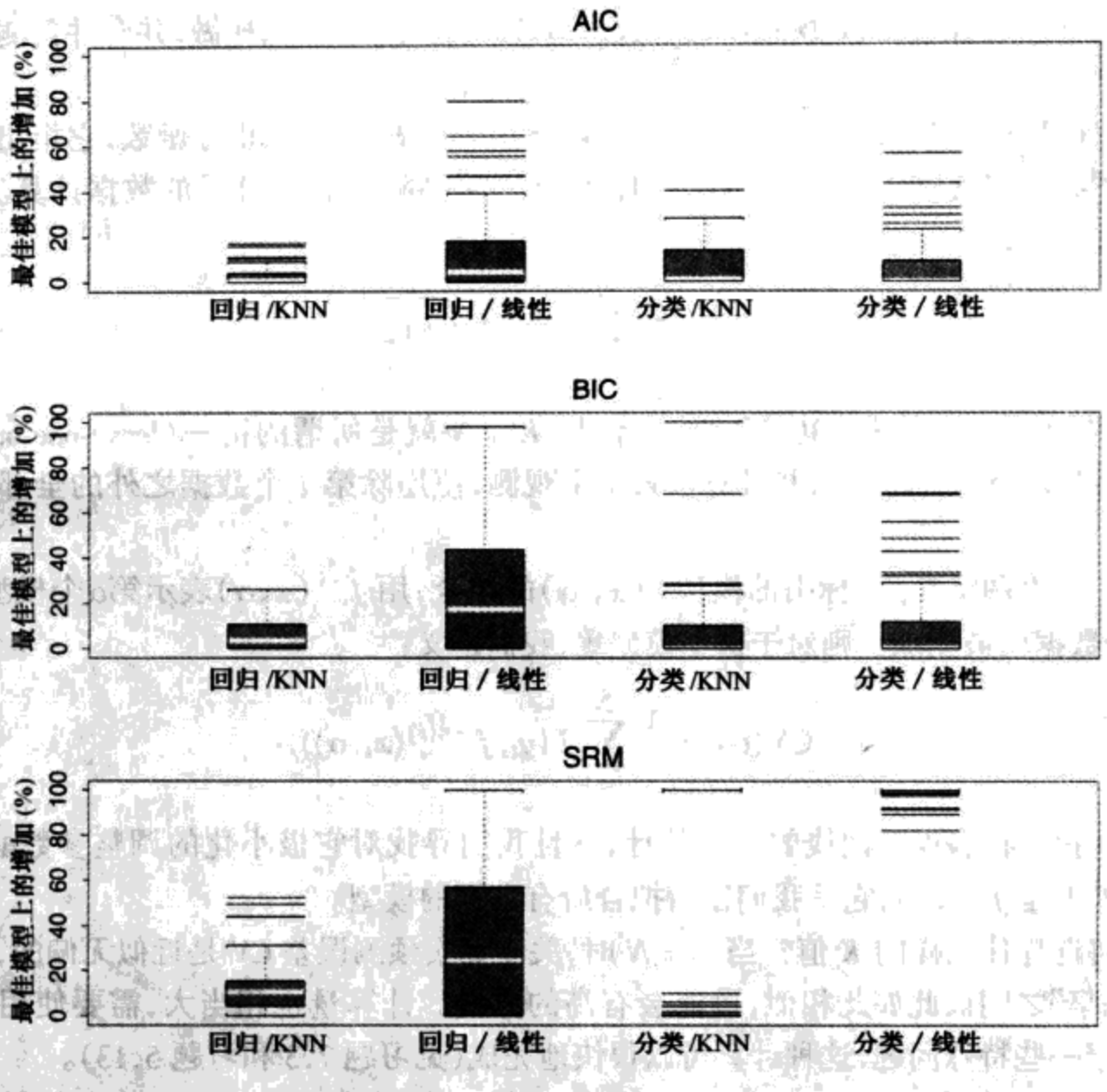


图 7.7　盒图显示相对误差 $100 \times [\mathrm{Err}(\hat{\alpha}) - \min_\alpha \mathrm{Err}(\alpha)]/[\max_\alpha \mathrm{Err}(\alpha) - \min_\alpha \mathrm{Err}(\alpha)]$ 在图 7.3 的 4 种情况下的分布。这是使用选择的模型相对于最佳模型的误差。在每幅盒子图中显示 100 个训练集，每个训练集合的容量为 50，误差在容量为 500 的检验集上计算

7.10　交叉验证

估计预测误差最简单且最广泛使用的方法可能就是交叉验证。这种方法直接估计样本外误差 $\mathrm{Err} = \mathrm{E}[L(Y, \hat{f}(X))]$。当方法 $\hat{f}(X)$ 用于 X 和 Y 的联合分布的独立检验样本时，它就是泛化误差。

理想地，如果我们有足够的数据，将把确认集置于一旁，并用它评价我们的预测模型的性能。但由于数据通常都很缺乏，所以这样做常常是不可能的。为了解决这一问题，K 折交叉验证使用部分可用数据拟合模型，而用不同的部分检验它。我们将数据分成容量大致相等的 K 部分，如 $K = 5$，该方案看上去如下所示：

1	2	3	4	5
训练	训练	检验	训练	训练

对于第 k 部分（上面的第 3 部分），我们用模型拟合数据的其他 $K - 1$ 部分，并当预测第 k

部分数据时，计算拟合模型的预测误差。我们对 $k=1, 2, \cdots, K$ 这样做，并合并预测误差的 K 个估计。

这里给出更详细的说明。令 $\kappa: \{1, \cdots, N\} \mapsto \{1, \cdots, K\}$ 是一个指标函数，它指出观测 i 被随机指派到其上的划分。用 $\hat{f}^{-k}(x)$ 表示拟合函数，用删除第 k 部分后的数据计算。那么，预测误差的交叉验证估计是：

$$\mathrm{CV} = \frac{1}{N}\sum_{i=1}^{N} L(y_i, \hat{f}^{-\kappa(i)}(x_i)) \tag{7.42}$$

典型地，K 的选择是 5 或者 10(见下面)。情形 $K=N$ 就是所谓的留一(leave-one-out)交叉验证。在这种情况下 $\kappa(i)=i$，并且对于第 i 个观测，使用除第 i 个数据之外的全部数据计算拟合。

给定一个由调整参数α标引的模型 $f(x, \alpha)$的集合，用 $\hat{f}^{-k}(x,\alpha)$表示第α个模型拟合，它的第α部分数据已被删除。则对于这个模型集，我们定义：

$$\mathrm{CV}(\alpha) = \frac{1}{N}\sum_{i=1}^{N} L(y_i, \hat{f}^{-\kappa(i)}(x_i, \alpha)) \tag{7.43}$$

函数 CV(α)提供检验误差曲线的一个估计，并且我们寻找对它极小化的调整参数$\hat{\alpha}$。我们最终选择的模型是 $f(x,\hat{\alpha})$，它是我们以后拟合所有数据的模型。

我们将选择什么样的 K 值？当 $K=N$ 时，关于真实预测误差 CV 是近似无偏的，但是由于 N 个“训练集”之间彼此如此相似，可能会有高的方差。计算量也相当大，需要使用学习方法 N 次。对于一些特殊问题，这种计算可以很快地完成(见习题 7.3 和习题 5.13)。

另一方面，比如说，对于 $K=5$，CV 有较低的方差。但偏倚可能是一个问题，依赖于学习方法的性能如何随训练集的容量而变化。图 7.8 显示一个给定任务上的分类器的假想“学习曲线”，是 1 - Err 作为训练集容量 N 的函数曲线图。当训练集的容量增加到 100 个观测时，分类器的性能有所改进，进一步增加训练集到 200 个观测，仅带来很微小的改进。如果我们的训练集有 200 个观测，从图 7.8 可以看到，5 折交叉验证估计容量为 160 的训练集上的分类器的性能实际上与容量为 200 的训练集的性能相同。这样，交叉验证将不会有太大的偏倚。然而，如果训练集有 50 个观测，5 倍交叉验证将在容量为 40 的训练集上估计分类器的性能，并且从图中看到，它将是 1 - Err 的一个过低估计。因此，作为 Err 的估计，交叉验证将是向上偏倚的。

概括地说，如果学习曲线在给定训练集容量的情况下有相当大的斜率，则 5 或 10 折交叉验证将过分估计真实预测误差。这种偏倚在实际当中是否是缺点取决于目标。另一方面，留一交叉验证有较低的偏倚，但有较高的方差。总之，5 或 10 折交叉验证已被推荐为较好的折中方案。

图 7.9 显示了图 7.3 的底部右图方案中，从单一训练集估计的预测误差曲线和 10 折交叉验证曲线。这是一个两类分类问题，它使用了子集容量为 p 的最佳子集回归线性模型。标准误差条被显示出来，它们是 10 个部分中的每一部分的各自误分类误差率的标准误差。尽管 CV 曲线超过 10 时相当平坦，但两个曲线都在 $p=10$ 时有最小值。通常“1-标准误差”规则与交叉验证一起使用，其中，我们选择最节俭的模型，它的误差与最好模型的误差相比不超过 1-标准误差。这里，看上去应当选择大约有 $p=9$ 个预测子的模型，而实际模型使用 $p=10$。

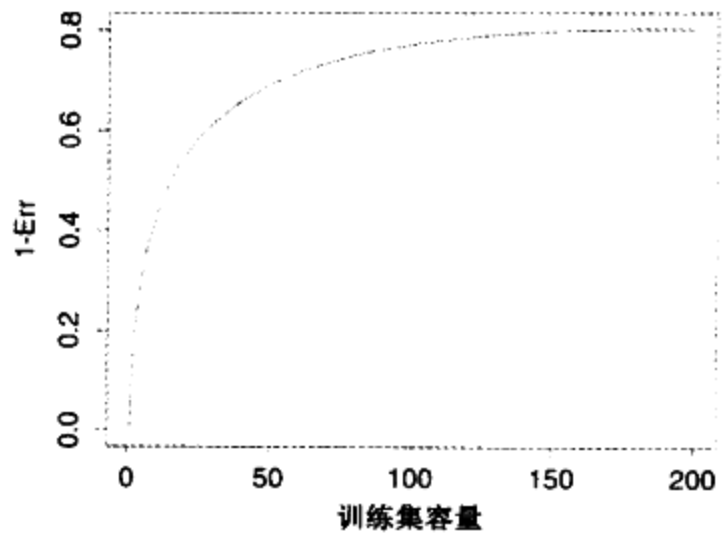


图 7.8　一个给定任务上的分类器假想学习曲线：它是 1 − Err 作为训练集容量 N 的函数曲线图。对于一个200个观测的数据集，5折交叉验证使用容量为160的训练集。它的作用非常像整个数据集。然而，对于50个观测的数据集，5折交叉验证将使用容量为40的训练集，且这将导致预测误差相当严重地过分估计

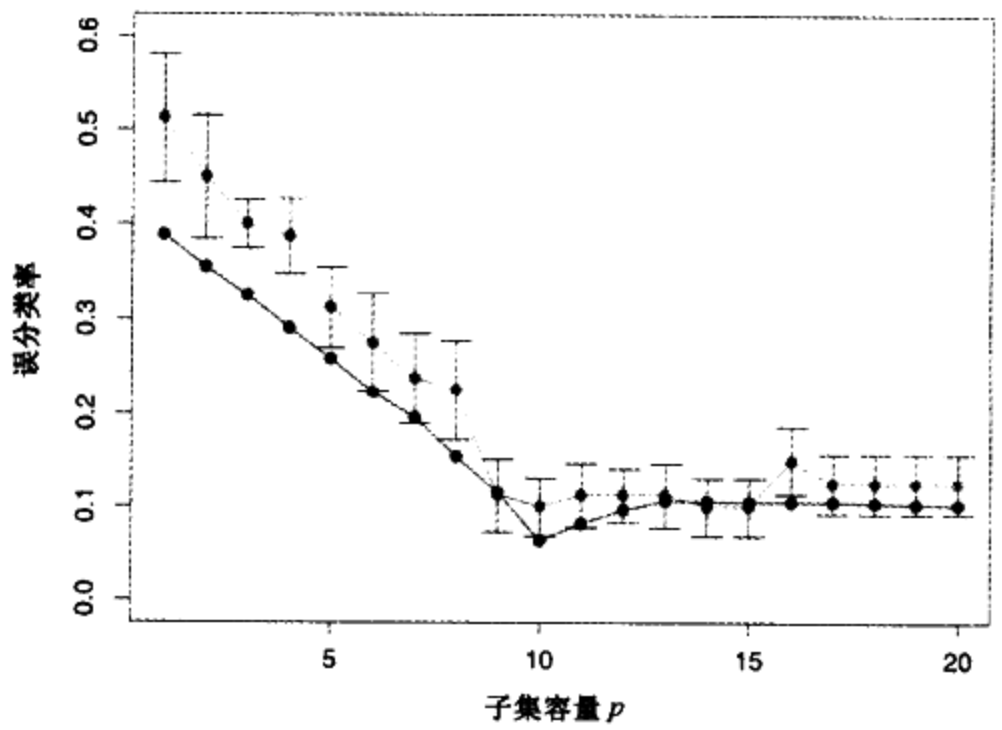


图 7.9　图 7.3 的底部右图的方案中，从单一训练集估计的预测误差(红色)和10折交叉验证曲线(绿色)(见彩页)

对于平方误差损失下的线性拟合，广义交叉验证(generalized cross-validation)提供了一种对留一交叉验证方便的逼近。和第 7.6 节中定义的一样，线性拟合方法可以写成：

$$\hat{\mathbf{y}} = \mathbf{S}\mathbf{y} \tag{7.44}$$

现在，对许多线性拟合方法，

$$\frac{1}{N}\sum_{i=1}^{N}[y_i - \hat{f}^{-i}(x_i)]^2 \quad = \quad \frac{1}{N}\sum_{i=1}^{N}\left[\frac{y_i - \hat{f}(x_i)}{1 - S_{ii}}\right]^2 \tag{7.45}$$

其中，S_{ii}是 $\mathbf{S}$ 的第 i 个对角元素(见习题 7.3)。GCV 逼近是：

$$\text{GCV} = \frac{1}{N}\sum_{i=1}^{N}\left[\frac{y_i - \hat{f}(x_i)}{1 - \text{trace}(\mathbf{S})/N}\right]^2 \tag{7.46}$$

量 trace(**S**)是有效的参数个数,和第 7.6 节中定义的一样。

对于某些情况,那里 **S** 的迹可能比元素 S_{ii} 更易计算,GCV 有计算上的优势。在光滑问题中,GCV 也可以缓和交叉验证光滑不足的趋势。GCV 和 AIC 之间的相似性可以从逼近 $1/(1-x)^2 \approx 1+2x$ 中看出(见习题 7.6)。

7.11 自助法

自助法是评估统计精度的一般工具。首先,我们一般地介绍自助法,然后说明怎样用它估计样本外预测误差。

假设有一个模型,拟合训练数据集。记训练集为 $\mathbf{Z}=(z_1,z_2\cdots,z_N)$,其中 $z_i=(x_i,y_i)$。自助法的基本思想是:从训练数据中有放回地随机抽取数据集,每个数据集的容量与原训练集相同。这样做 B 次(比如说,$B=100$),产生 B 个自助法数据集,如图 7.10 所示。然后,对每个自助法数据集重新拟合模型,并检查 B 次重复实验上的拟合行为。

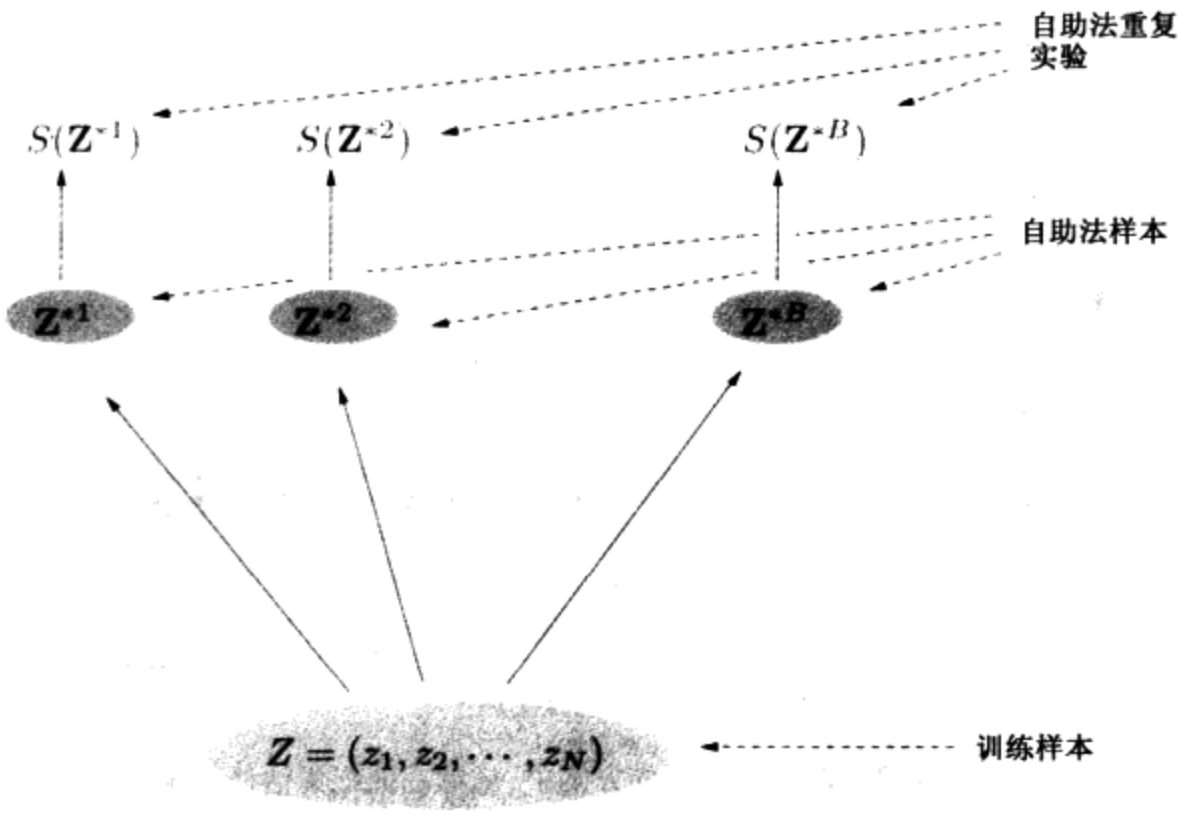


图 7.10 自助法过程图解。我们希望评估由数据集计算的 $S(\mathbf{Z})$的统计精度。B 个容量为 N的训练集$\mathbf{Z}^{*b}(b=1,\cdots,B)$从原始数据集中有放回地抽取。量$S(\mathbf{Z})$由每个自助法训练集计算,并且值$S(\mathbf{Z}^{*1}),\cdots,S(\mathbf{Z}^{*B})$用于评估$S(\mathbf{Z})$的统计精度

图中,$S(\mathbf{Z})$是由数据 **Z** 计算的任意量,例如,某输入点上的预测。从自助法抽样,我们可以估计 $S(\mathbf{Z})$的分布的任意性质,例如,它的方差:

$$\widehat{\mathrm{Var}}[S(\mathbf{Z})]=\frac{1}{B-1}\sum_{b=1}^{B}(S(\mathbf{Z}^{*b})-\bar{S}^{*})^2 \tag{7.47}$$

其中,$\bar{S}^{*}=\sum_b S(\mathbf{Z}^{*b})/B$。注意:$\widehat{\mathrm{Var}}[S(\mathbf{Z})]$可以看做是从数据$(z_1,z_2,\cdots,z_N)$的经验分布函数 $\hat{F}$ 抽样下,$S(\mathbf{Z})$的方差的蒙特卡罗估计。

我们应该怎样用自助法估计预测误差呢?一种方法是用被考虑的模型拟合自助法样本

集，然后求出它对原训练集的预测精度。如果 $\hat{f}^{*b}(x_i)$ 是 x_i 上的预测值，来自模型对第 b 个自助法数据集的拟合，则我们的估计是：

$$\widehat{\mathrm{Err}}_{\mathrm{boot}} = \frac{1}{B}\frac{1}{N}\sum_{b=1}^{B}\sum_{i=1}^{N} L(y_i, \hat{f}^{*b}(x_i)) \tag{7.48}$$

然而，容易看出 $\widehat{\mathrm{Err}}_{\mathrm{boot}}$ 一般不能提供较好的估计。原因是自助法数据集起训练样本的作用，而原始训练集起检验样本的作用，而这两种样本有共同的观测。这种重叠使过分拟合预测看上去超常地好，这也是交叉验证显式地对训练样本和检验样本使用不重叠数据的原因。作为例子，考虑一个用于两类分类问题的 1-最近邻分类法。每个类有相同个数的观测，其中的特征和分类标号实际上是独立的。那么真实误差率是 0.5。但是，对自助法估计 $\widehat{\mathrm{Err}}_{\mathrm{boot}}$ 的贡献将是 0，除非观测 i 不出现在自助法样本 b 中。在后一种情况下，它将有正确的期望值 0.5。现在：

$$\begin{aligned} \Pr\{\text{观测} i \in \text{自助法样本} b\} &= 1 - \left(1 - \frac{1}{N}\right)^N \\ &\approx 1 - e^{-1} \\ &= 0.632 \end{aligned} \tag{7.49}$$

因此，$\widehat{\mathrm{Err}}_{\mathrm{boot}}$ 的期望大约是 $0.5 \times 0.368 = 0.184$，远远低于正确的误差率 0.5。

通过模仿交叉验证，可以获得一个较好的自助法估计。对每个观测，我们仅计算不包含该观测的自助法样本的预测。预测误差的留一自助法估计由下式定义：

$$\widehat{\mathrm{Err}}^{(1)} = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{|C^{-i}|}\sum_{b\in C^{-i}} L(y_i, \hat{f}^{*b}(x_i)) \tag{7.50}$$

这里，C^{-i} 是不包含观测 i 的自助法样本 b 的指标集，而 $|C^{-i}|$ 是这种样本的数量。在计算 $\widehat{\mathrm{Err}}^{(1)}$ 时，我们或者必须选择足够大的 B，以保证所有 $|C^{-i}|$ 大于 0，或者可以恰好略去式(7.50)中 $|C^{-i}|$ 的对应值为 0 的项。

留一自助法解决了 $\widehat{\mathrm{Err}}_{\mathrm{boot}}$ 的过分拟合问题，但却存在交叉验证讨论中提到的训练集容量偏倚的问题。在每个自助法样本中不同观测的平均个数大约是 $0.632 \cdot N$，所以它的偏倚将大致相当于 2 折交叉验证的偏倚。这样，如果学习曲线在样本容量 $N/2$ 上有相当大的斜率，则作为真实误差的估计，留一自助法将偏高。

“.632 估计子”的设计用于缓解这种偏倚。它由下式定义：

$$\widehat{\mathrm{Err}}^{(.632)} = .368 \cdot \overline{\mathrm{err}} + .632 \cdot \widehat{\mathrm{Err}}^{(1)} \tag{7.51}$$

.632 估计子的推导很复杂；直观地，它把留一自助法估计向下压向训练误差率，因此减少它的上偏倚。常量 .632 的使用与式(7.49)有关。

.632 估计子在“轻拟合”情况下工作得很好，但在过分拟合情形可能失败。这里有一个引自 Breiman 等人(1984)论文中的例子。假设有两个等尺寸的类，其目标独立于类标号，并且我们应用 1-最邻近规则。那么，$\overline{\mathrm{err}} = 0$，$\widehat{\mathrm{Err}}^{(1)} = 0.5$，这样 $\widehat{\mathrm{Err}}^{(.632)} = .632 \times 0.5 = .316$。然而，真实误差率是 0.5。

可以通过考虑过分拟合量来改进 .632 估计子。首先，定义 γ 是无信息误差率(no-information error rate)：如果输入和类标号是独立的，则它就是我们的预测规则的误差率。γ 的一个估计通过在目标 y_i 和预测子 x_i 的所有可能组合上计算预测规则

$$\hat{\gamma} = \frac{1}{N^2}\sum_{i=1}^{N}\sum_{i'=1}^{N} L(y_i, \hat{f}(x_{i'})) \tag{7.52}$$

得到。

例如，考虑对分法分类问题：令 $\hat{p}_1$ 是观测到的等于 1 的响应 y_i 的比例，而 $\hat{q}_1$ 是观测到的等于 1 的预测 $\hat{f}(x_{i'})$ 的比例。则

$$\hat{\gamma} = \hat{p}_1(1-\hat{q}_1) + (1-\hat{p}_1)\hat{q}_1 \tag{7.53}$$

使用类似于 1-最近邻的规则，有 $\hat{p}_1 = \hat{q}_1$，$\hat{\gamma}$ 的值是 $2\hat{p}_1(1-\hat{p}_1)$。式(7.53)的多类型泛化是 $\hat{\gamma} = \sum_\ell \hat{p}_\ell(1-\hat{q}_\ell)$。

使用它，相对过分拟合率(relative overfitting rate)定义为：

$$\hat{R} = \frac{\widehat{\mathrm{Err}}^{(1)} - \overline{\mathrm{err}}}{\hat{\gamma} - \overline{\mathrm{err}}} \tag{7.54}$$

这是一个在 0 和 1 之间变化的量。如果没有过分拟合($\widehat{\mathrm{Err}}^{(1)} = \overline{\mathrm{err}}$)，其值为 0；如果过分拟合等于无信息值 $\hat{\gamma} - \overline{\mathrm{err}}$，其值为 1。最后，我们用下式定义“.632 + ”估计子：

$$\begin{aligned} \widehat{\mathrm{Err}}^{(.632+)} &= (1-\hat{w})\cdot\overline{\mathrm{err}} + \hat{w}\cdot\widehat{\mathrm{Err}}^{(1)} \\ \text{其中 } \hat{w} &= \frac{.632}{1-.368\hat{R}} \end{aligned} \tag{7.55}$$

权值 w 在 .632(如果 $\hat{R}=0$)和 1(如果 $\hat{R}=1$)之间变化，从而 $\widehat{\mathrm{Err}}^{(.632+)}$ 在 $\widehat{\mathrm{Err}}^{(.632)}$ 和 $\widehat{\mathrm{Err}}^{(1)}$ 之间变化。同样，式(7.55)的导出也很复杂：粗略地说，它在留一自助法和依赖于过分拟合量的训练误差率之间做出了折中。对于类标号独立于输入的 1-最近邻问题，$\hat{w} = \hat{R} = 1$，从而有 $\widehat{\mathrm{Err}}^{(.632+)} = \widehat{\mathrm{Err}}^{(1)}$，它具有正确的期望值 0.5。在其他具有较少过分拟合的问题中，$\widehat{\mathrm{Err}}^{(.632+)}$ 取 $\overline{\mathrm{err}}$ 和 $\widehat{\mathrm{Err}}^{(1)}$ 之间的某个值。

7.11.1 例(续)

图 7.11 显示 5 折交叉验证和 .632 + 自助法估计在图 7.7 的 4 个相同问题上的结果。和图 7.7—样，图 7.11 显示 $100\cdot[\mathrm{Err}_{\hat{\alpha}} - \min_\alpha \mathrm{Err}(\alpha)]/[\max_\alpha \mathrm{Err}(\alpha) - \min_\alpha \mathrm{Err}(\alpha)]$的盒图，即，使用选定的模型相对于最佳模型的误差。在每幅盒图中表示了 20 个不同的训练集。两种度量总的来说都很好，与图 7.7 中的 AIC 相比几乎相同或略差一些。

我们的结论是：关于这些特殊问题和拟合方法，AIC、交叉验证或自助法的极小化产生的模型都相当接近于可用的最佳模型。注意，对于模型选择，任何度量都可能是有偏的，并且只要偏倚不改变方法的相对性能，这种偏倚并无妨碍。例如，对任意度量增加一个常量将不改变结果模型的选择。然而，对许多自适应的、非线性的技术(如树)，有效的参数个数的估计是非常困难的。这使得诸如 AIC 等方法不再实用，我们可选的方法只有交叉验证和自助法。

一个不同的问题是：每种方法估计检验误差的效果如何？平均来看，AIC 准则在 4 种情况上对它所选择的模型的预测误差的过高估计分别为 38%、37%、51%和 30%；BIC 的表现类似。相比较，交叉验证对误差的过高估计分别为 1%、4%、0%和 4%；自助法与此大致相同。因此，如果需要估计检验误差的精度，交叉验证或自助法度量计算的额外工作是值得的。借助于诸

如树等其他拟合方法，交叉验证和自助法对真实误差的过低估计可能在 10% 左右，因为最佳树的搜索受验证集的影响非常大。仅在这些情况下，一个独立的检验集将提供检验误差的无偏估计。

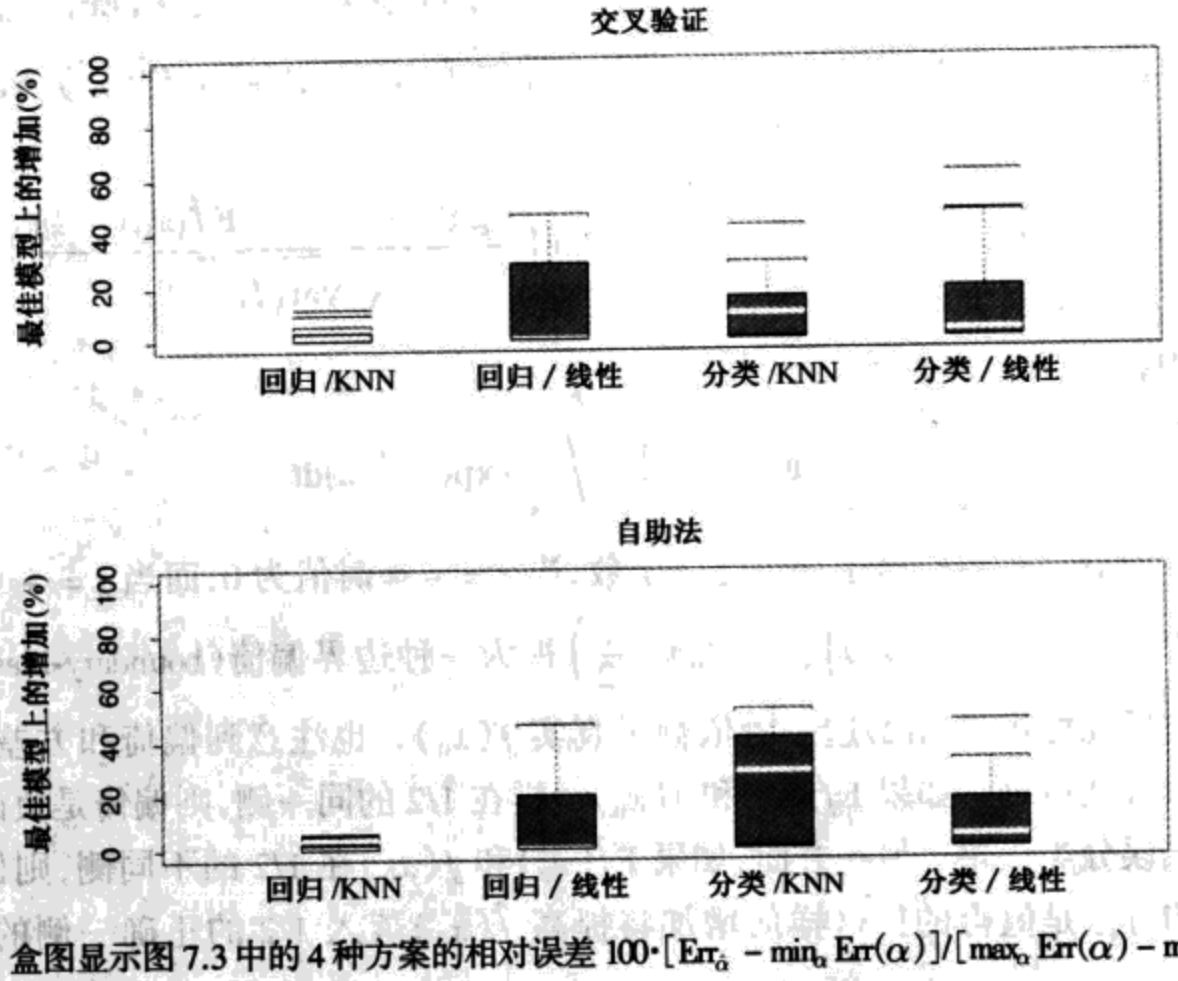


图 7.11　盒图显示图 7.3 中的 4 种方案的相对误差 $100\cdot[\mathrm{Err}_{\hat{\alpha}} - \min_\alpha \mathrm{Err}(\alpha)]/[\max_\alpha \mathrm{Err}(\alpha) - \min_\alpha \mathrm{Err}(\alpha)]$ 的分布。它是使用所选择的模型相对于最佳模型的误差。在每幅盒图中有 20 个训练集

文献注释

交叉验证的主要参考文献是 *Stone*(1974)、*Stone*(1977)和 *Allen*(1977)。*AIC* 是由 *Akaike*(1973)提出的，而 *BIC* 是由 *Schwartz*(1979)引进的。*Madigan* 和 *Raftery*(1994)给出了贝叶斯模型选择的综述。*MDL* 准则出自 *Rissanen*(1983)。*Cover* 和 *Thomas*(1991)包含了编码理论和复杂性的较好叙述。*VC* 维在 *Vapnik*(1996)中介绍。*Stone*(1977)证明 *AIC* 和留一交叉验证近似等价。广义交叉验证由 *Golub* 等人(1979)和 *Wahba*(1980)描述，该主题的进一步讨论可以在 *Wahba*(1990)的专著中找到，也可以参考 *Hastie* 和 *Tibshirani*(1990)的第 3 章。自助法源自于 *Efron*(1979)，综述参见 *Efron* 和 *Tibshirani*(1993)。*Efron*(1983)提出了一些预测误差的自助法估计，包括优化和 .632 估计。*Efron*(1986)比较了 *CV*、*GCV* 和自助法估计的误差率。使用交叉验证和自助法进行模型选择在 *Breiman* 和 *Spector*(1992)、*Breiman*(1992)、*Shao*(1996)和 *Zhang*(1993)中研究。.632+ 估计子由 *Efron* 和 *Tibshirani*(1997)提出。

习题

7.1　推导样本内误差估计(7.20)。

7.2　对于 0-1 损失，$Y \in \{0,1\}$ 且 $\Pr(Y=1 \mid x_0) = f(x_0)$，证明：

$$\begin{aligned} \mathrm{Err}(x_0) &= \Pr(Y \neq \hat{G}(x_0)|X = x_0) \\ &= \mathrm{Err}_{\mathrm{B}}(x_0) + |2f(x_0) - 1|\Pr(\hat{G}(x_0) \neq G(x_0)|X = x_0) \end{aligned} \tag{7.56}$$

其中，$\hat{G}(x) = I\left(\hat{f}(x) > \frac{1}{2}\right)$，$G(x) = I(f(x) > \frac{1}{2})$是贝叶斯分类器，$\mathrm{Err}_{\mathrm{B}}(x_0) = \Pr(Y \neq G$
$(x_0|X = x_0)$是 x_0 上不可约的贝叶斯误差。使用近似 $\hat{f}(x_0) \sim N(E\hat{f}(x_0), \mathrm{Var}(\hat{f}(x_0))$，证明：

$$\Pr(\hat{G}(x_0) \neq G(x_0)|X = x_0) \approx \Phi\left(\frac{\mathrm{sign}(\frac{1}{2} - f(x_0))(\mathrm{E}\hat{f}(x_0) - \frac{1}{2})}{\sqrt{\mathrm{Var}(\hat{f}(x_0))}}\right) \tag{7.57}$$

在上式中，

$$\Phi(t) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{t} \exp(-t^2/2)\mathrm{d}t$$

是累积高斯分布函数。这是一个递增函数，当 $t = -\infty$时值为 0，而当 $t = +\infty$时值为 1。我们可以将 $\mathrm{sign}\left(\frac{1}{2} - f(x_0)\right)\left(\mathrm{E}\hat{f}(x_0) - \frac{1}{2}\right)$视为一种边界偏倚(boundary-bias)项，因为它仅通过它所在的边界(1/2)这一侧依赖于真实 $f(x_0)$。也注意到偏倚和方差以乘法方式而非加法方式组合。如果 $\mathrm{E}f(x_0)$和 $f(x_0)$一样在 1/2 的同一侧，则偏倚是负的，且降低方差将降低误分类误差。另一方面，如果 $\mathrm{E}\hat{f}(x_0)$和 $f(x_0)$在 1/2 的不同侧，则偏倚是正的，并且增加方差是值得的！这样的增加将提高 $\hat{f}(x_0)$落入 1/2 的正确一侧的机会(Friedman，1997)。

7.3 令 $\hat{\mathbf{f}} = \mathbf{S}\mathbf{y}$ 是 $\mathbf{y}$ 的线性光滑。

(a) 如果 S_{ii} 是 $\mathbf{S}$ 的第 i 个对角元素，证明：对于从最小二乘方投影和三次光滑样条产生的 $\mathbf{S}$，交叉验证过的残差可以写成：

$$y_i - \hat{f}^{-i}(x_i) = \frac{y_i - \hat{f}(x_i)}{1 - S_{ii}} \tag{7.58}$$

(b) 使用这一结果证明 $|y_i - \hat{f}^{-i}(x_i)| \geqslant |y_i - \hat{f}(x_i)|$。

(c) 找出对任意光滑子 $\mathbf{S}$ 使结果(7.58)成立的一般条件。

7.4 在平方误差损失情形下，考虑样本内预测误差(7.15)和训练误差$\overline{\mathrm{err}}$：

$$\begin{aligned} \mathrm{Err}_{\mathrm{in}} &= \frac{1}{N}\sum_{i=1}^{N} \mathrm{E}_{Y^{\mathrm{new}}}\mathrm{E}_{\mathbf{y}}(Y_i^{\mathrm{new}} - \hat{f}(x_i))^2 \\ \overline{\mathrm{err}} &= \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{f}(x_i))^2 \end{aligned}$$

在每个表达式和展开式中增加和减去 $f(x_i)$和 $\mathrm{E}\hat{f}(x_i)$。由此建立的训练误差中的乐观性是：

$$\frac{2}{N}\sum_{i=1}^{N}\mathrm{Cov}(\hat{y}_i, y_i)$$

和式(7.17)中的一样。

7.5 关于线性光滑子 $\hat{\mathbf{y}} = \mathbf{S}\mathbf{y}$，证明：

$$\sum_{i=1}^{N} \mathrm{Cov}(\hat{y}_i, y_i) = \mathrm{trace}(\mathbf{S})\sigma_\epsilon^2 \tag{7.59}$$

这证明了它用做有效的参数个数是正确的。

7.6 使用逼近 $1/(1-x)^2 \approx 1 + 2x$ 揭示 C_p/AIC(7.22)和 GCV(7.46)之间的联系，其主要差别是用于估计噪声方差 σ_ϵ^2 的模型。

7.7 证明函数集 $\{I(\sin(\alpha x) > 0)\}$ 对于任意 ℓ，能够分散如下直线上的点：

$$z^1 = 10^{-1}, \cdots, z^\ell = 10^{-\ell} \tag{7.60}$$

因此，类 $\{I(\sin(\alpha x) > 0)\}$ 的 VC 维是无限的。

7.8 对于第 3 章的前列腺数据，执行最佳子集线性回归分析，和表 3.3(从左数第三列)中一样。计算预测误差的 AIC、BIC、5 折和 10 折交叉验证及自助法 .632 估计，并讨论结果。

第8章　模型推理和平均

8.1　引言

在本书中，模型的拟合（学习）多半通过极小化平方和（对于回归），或者通过极小化互熵（对于分类）来实现。实际上，这两种极小化都是极大似然拟合方法的实例。

本章将全面阐述极大似然方法和关于推理的贝叶斯方法。第7章介绍的自助法将在此背景下讨论，并讨论它与极大似然和贝叶斯方法的联系。最后，我们给出模型平均与改进的一些相关技术，包括委员会方法、装袋、堆栈、冲击（bumping）等。

8.2　自助法和极大似然法

8.2.1　一个光滑例子

通过从训练数据中选样，自助法提供一种评估不确定性的直接计算方法。这里，我们用一个简单的一维光滑问题解释自助法，并说明它与极大似然的联系。

记训练数据为 $\mathbf{Z} = \{z_1, z_2, \cdots, z_N\}$，其中 $z_i = (x_i, y_i)$，$i = 1, 2, \cdots, N$。这里，x_i 是一维输入，y_i 是输出，或者是连续的，或者是分类的。作为一个例子，考虑图8.1左图中显示的 $N = 50$ 个数据点。

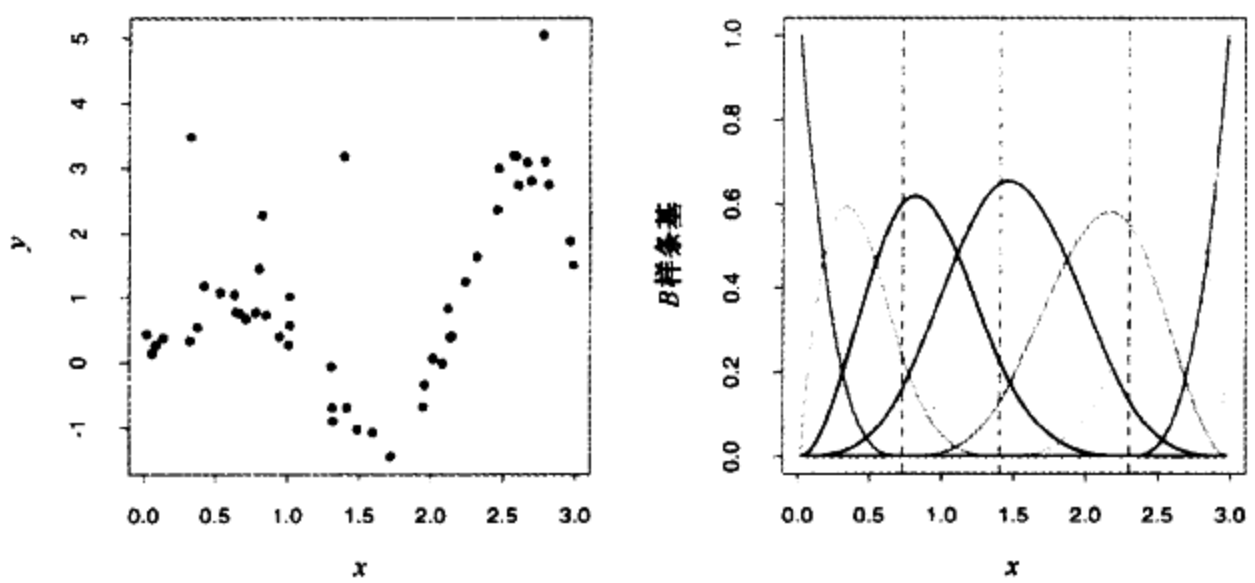


图8.1　左图：光滑例子的数据。右图：7个 B 样条基函数的集合。垂直的虚线指出三个纽结的布局（见彩页）

假设我们决定用有三个纽结的三次样条拟合这些数据，三个纽结位于 X 值的四分位数上。这是一个7维线性函数空间，并且可以用 B 样条基函数的线性展开式表示（见第5.9.2节）：

$$\mu(x) = \sum_{j=1}^{7} \beta_j h_j(x) \tag{8.1}$$

这里，$h_j(x)$，$j = 1,2,\cdots,7$ 是 7 个函数，如图 8.1 右图所示。可以把$\mu(x)$看做表示条件均值 $\mathrm{E}(Y \mid X' = x)$。

令 $\mathbf{H}$ 是 $N \times 7$ 矩阵，第 ij 个元素为 $h_j(x_i)$。通过对训练集上的平方误差极小化得到的 β 的通常估计由下式给出：

$$\hat{\beta} = (\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T\mathbf{y} \tag{8.2}$$

相应的拟合 $\hat{\mu}(x) = \sum_{j=1}^{7} \hat{\beta}_j h_j(x)$ 在图 8.2 的左上图显示。

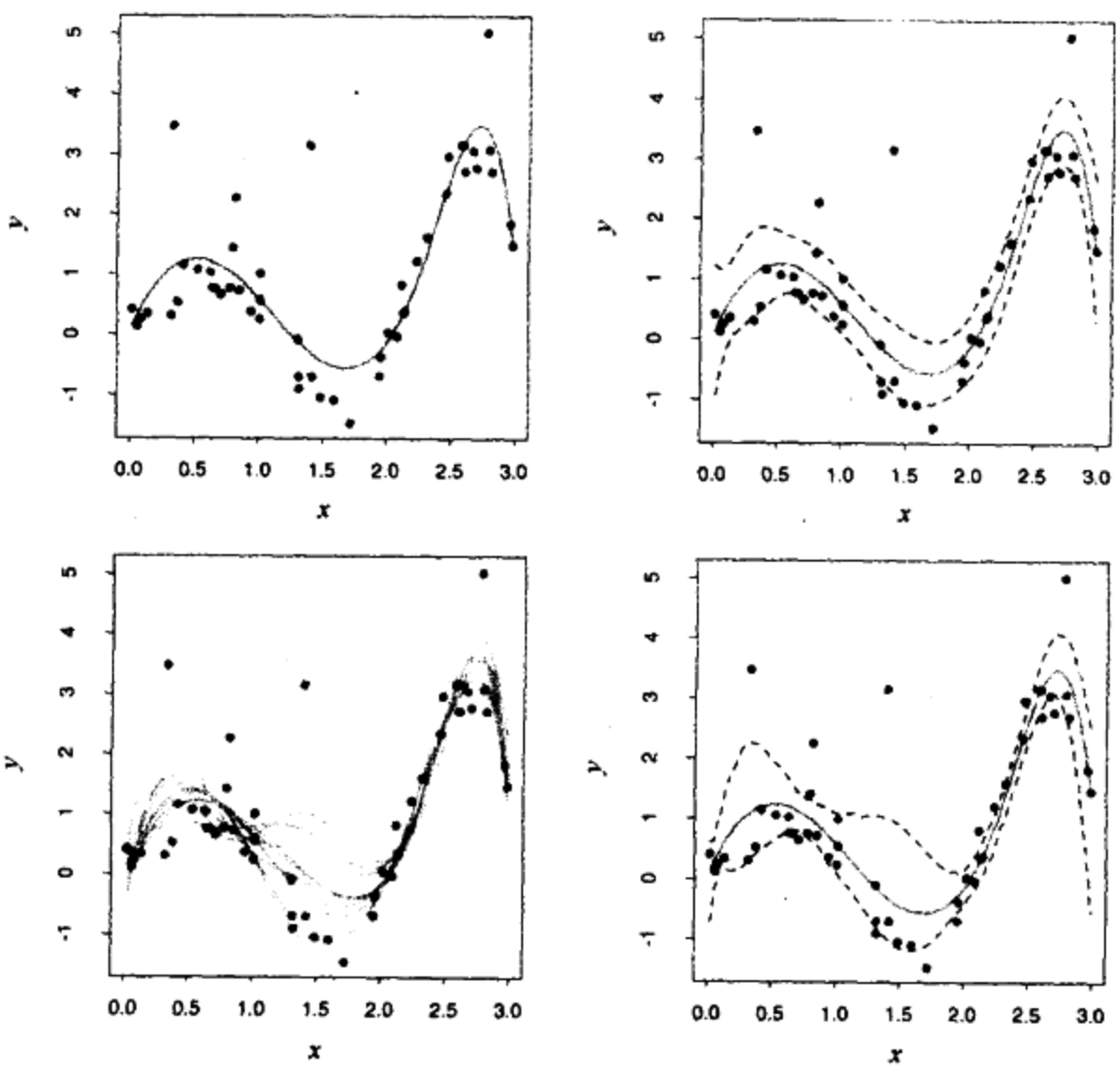


图 8.2　左上：数据的 B 样条光滑。右上：B 样条光滑加减 1.96 倍标准误差带。左下：B 样条光滑的 10 个自助法重复实验。右下：从自助法分布计算的有95%标准误差带的B样条光滑(见彩页)

$\hat{\beta}$ 的估计协方差矩阵是：

$$\widehat{\mathrm{Var}}(\hat{\beta}) = (\mathbf{H}^T\mathbf{H})^{-1}\hat{\sigma}^2 \tag{8.3}$$

其中，我们用 $\hat{\sigma}^2 = \sum_{i=1}^{N}(y_i - \hat{\mu}(x_i))^2/N$ 估计噪声方差。令 $h(x)^T = (h_1(x), h_2(x),\cdots, h_7(x))$，预测$\hat{\mu}(x) = h(x)^T\hat{\beta}$ 的标准误差是：

$$\widehat{\mathrm{se}}[\hat{\mu}(x)] = [h(x)^T(\mathbf{H}^T\mathbf{H})^{-1}h(x)]^{\frac{1}{2}}\hat{\sigma} \tag{8.4}$$

在图 8.2 的右上图，我们绘出$\hat{\mu}(x) \pm 1.96 \cdot \widehat{\mathrm{se}}[\hat{\mu}(x)]$。由于 1.96 是标准正态分布的

97.5% 的点，因此，它们提供$\mu(x)$的大约 $100-2\times2.5\% = 95\%$ 的逐点置信带。

这里将解释我们应该怎样在这个例子中应用自助法。我们抽取 B 个数据集，每个数据集的容量为 $N=50$，从训练数据有放回地抽样，抽样单位是 $z_i=(x_i,y_i)$。对每个自助法数据集 $\mathbf{Z}^*$，我们拟合一个三次样条$\hat{\mu}^*(x)$；来自 10 个这种样本的拟合显示在图 8.2 的左下图中。使用 $B=200$ 个自助法样本，从每个 x 上的百分位数形成 95% 逐点置信带：我们在每个 x 处找出 $(2.5\%\times200)$第五个最大和最小值。这些被绘制在图 8.2 的右下图中。置信带看上去与右上图相似，在端点略宽一点。

实际上，在最小二乘方估计(8.2)和(8.3)、自助法和极大似然之间存在着紧密的联系。假设我们进一步假定模型误差是高斯的

$$\begin{aligned} Y &= \mu(X)+\varepsilon;\ \ \varepsilon\sim N(0,\sigma^2) \\ \mu(x) &= \sum_{j=1}^{7}\beta_j h_j(x) \end{aligned} \tag{8.5}$$

上面介绍的自助法采用从训练数据中有放回地抽样，称为非参数自助法(nonparametric bootstrap)。这确实意味着该方法是“模型自由的”，因为它使用原始的数据，而不是特殊的参数模型来产生新的数据集。考虑自助法的一个变形，称之为参数自助法(parametric bootstrap)。这里，我们通过对预测值增加高斯噪声来模拟新的响应：

$$y_i^*=\hat{\mu}(x_i)+\varepsilon_i^*;\ \ \varepsilon_i^*\sim N(0,\hat{\sigma}^2);\ \ i=1,2,\cdots,N \tag{8.6}$$

该过程重复 B 次，例如 $B=200$。结果自助法数据集形如$(x_1,y_1^*),\cdots,(x_N,y_N^*)$，并在每一个数据集上重新计算 B 样条光滑。随自助法样本的数目趋向无穷大，这种方法盅得到的置信带将与右上图中的最小平方置信带恰好相同。由自助法样本 $\mathbf{y}^*$ 估计的函数是$\hat{\mu}^*(x)=h(x)^T(\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T\mathbf{y}^*$，并且具有分布

$$\hat{\mu}^*(x)\sim N(\hat{\mu}(x),h(x)^T(\mathbf{H}^T\mathbf{H})^{-1}h(x)\hat{\sigma}^2) \tag{8.7}$$

注意，该分布的均值是最小二乘方估计，而且标准差与近似公式(8.4)相同。

8.2.2 极大似然推理

这表明：在前面的例子中参数自助法与最小二乘方是一致的，因为模型(8.5)具有加法高斯误差。一般地，参数自助法并非与最小二乘方一致，而是与极大似然一致，我们现在对其进行综述。

从为观测指定概率密度或概率质量函数开始

$$z_i\sim g_\theta(z) \tag{8.8}$$

在该表达式中，θ 表示一个或多个控制 Z 分布的未知参数。这称为 Z 的参数模型(parametric model)。作为一个例子，如果 Z 满足正态分布，均值为μ，方差为 σ^2，则

$$\theta=(\mu,\sigma^2) \tag{8.9}$$

并且

$$g_\theta(z)=\frac{1}{\sqrt{2\pi}\sigma}e^{-\frac{1}{2}(z-\mu)^2/\sigma^2} \tag{8.10}$$

极大似然基于似然函数(likelihood function),由下式给出:

$$L(\theta;\mathbf{Z})=\prod_{i=1}^{N}g_\theta(z_i) \tag{8.11}$$

它是模型 g_θ 下观测的数据的概率。该似然的定义仅取决于一个正乘数,我们已取此正乘数为1。固定数据 $\mathbf{Z}$,我们把 $L(\theta;\mathbf{Z})$看做 θ 的函数。

将 $L(\theta;\mathbf{Z})$的对数记为:

$$\begin{aligned}\ell(\theta;\mathbf{Z}) &= \sum_{i=1}^{N}\ell(\theta;z_i)\\ &= \sum_{i=1}^{N}\log g_\theta(z_i)\end{aligned} \tag{8.12}$$

有时简记为 $\ell(\theta)$。该表达式称为对数似然,而每个值 $\ell(\theta;z_i)=\log g_\theta(z_i)$称为对数似然分量。极大似然方法选取值 $\theta=\hat{\theta}$ 以极大化 $\ell(\theta;\mathbf{Z})$。

似然函数可以用于评估 $\hat{\theta}$ 的精度。我们需要更多定义。得分函数(score function)由下式定义:

$$\dot{\ell}(\theta;\mathbf{Z})=\sum_{i=1}^{N}\dot{\ell}(\theta;z_i) \tag{8.13}$$

其中,$\dot{\ell}(\theta;z_i)=\partial\ell(\theta;z_i)/\partial\theta$。假设似然在参数空间内部取极大值,$\dot{\ell}(\hat{\theta};\mathbf{Z})=0$。信息矩阵(information matrix)是:

$$\mathbf{I}(\theta)=-\sum_{i=1}^{N}\frac{\partial^2\ell(\theta;z_i)}{\partial\theta\partial\theta^T} \tag{8.14}$$

当 $\mathbf{I}(\theta)$在 $\theta=\hat{\theta}$ 处计算时,通常称它为观测信息(observed information)。费希尔信息(Fisher information 或期望信息)是:

$$\mathbf{i}(\theta)=\mathrm{E}_\theta[\mathbf{I}(\theta)] \tag{8.15}$$

最后,令 θ_0 表示 θ 的真实值。

一个标准结果指出,当 $N\to\infty$ 时,极大似然估计算子的抽样分布以正态分布为极限:

$$\hat{\theta}\to N(\theta_0,\mathbf{i}(\theta_0)^{-1}) \tag{8.16}$$

这里,我们正从 $g_{\theta_0}(z)$中独立地抽样。这表明 $\hat{\theta}$ 的抽样分布可以用下式近似:

$$N(\hat{\theta},\mathbf{i}(\hat{\theta})^{-1})\ \text{或}\ N(\hat{\theta},\mathbf{I}(\hat{\theta})^{-1}) \tag{8.17}$$

其中,$\hat{\theta}$ 表示观测数据的极大似然估计。

$\hat{\theta}_j$ 的标准误差的相应估计由下式得到:

$$\sqrt{\mathbf{i}(\hat{\theta})_{jj}^{-1}}\qquad\text{且}\qquad\sqrt{\mathbf{I}(\hat{\theta})_{jj}^{-1}} \tag{8.18}$$

θ_j 的置信点可以从式(8.17)的两个近似中的任何一个来构造。这样的置信点分别具有如下形式:

$$\hat{\theta}_j - z^{(1-\alpha)} \cdot \sqrt{\mathbf{i}(\hat{\theta})_{jj}^{-1}} \quad \text{或} \quad \hat{\theta}_j - z^{(1-\alpha)} \cdot \sqrt{\mathbf{I}(\hat{\theta})_{jj}^{-1}}$$

其中，$z^{(1-\alpha)}$是标准正态分布的$1-\alpha$百分位。更精确的置信区间可以使用如下 χ^2 近似

$$2[\ell(\hat{\theta}) - \ell(\theta_0)] \sim \chi_p^2 \tag{8.19}$$

从似然函数导出。其中，p 是θ 中的分量的数目。结果 $1-2\alpha$置信区间是满足 $2[\ell(\hat{\theta}) - \ell(\theta_0)] \leqslant \chi_p^{2(1-2\alpha)}$ 的所有θ_0 的集合，其中 $\chi_p^{2(1-2\alpha)}$ 是具有p 个自由度χ^2 分布的$1-2\alpha$百分位。

让我们回到光滑例子，看看极大似然产生了什么。参数是 $\theta = (\beta, \sigma^2)$。对数似然是：

$$\ell(\theta) = -\frac{N}{2}\log\sigma^2 2\pi - \frac{1}{2\sigma^2}\sum_{i=1}^{N}(y_i - h(x_i)^T\beta)^2 \tag{8.20}$$

极大似然估计通过置$\partial\ell/\partial\beta = 0$ 和$\partial\ell/\partial\sigma^2 = 0$ 得到，给出：

$$\begin{aligned} \hat{\beta} &= (\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T\mathbf{y} \\ \hat{\sigma}^2 &= \frac{1}{N}\sum(y_i - \hat{\mu}(x_i))^2 \end{aligned} \tag{8.21}$$

它与式(8.2)中和式(8.3)下面给出的一般估计相同。

$\theta = (\beta, \sigma^2)$的信息矩阵是分块对角的，对应于 β 的块是：

$$\mathbf{I}(\beta) = (\mathbf{H}^T\mathbf{H})/\sigma^2 \tag{8.22}$$

所以估计方差$(\mathbf{H}^T\mathbf{H})^{-1}\hat{\sigma}^2$ 与最小二乘方估计(8.3)一致。

8.2.3 自助法与极大似然

本质上，自助法就是非参数或参数极大似然的计算机实现。自助法优于极大似然公式之处是，它允许我们在没有公式可用时计算标准误差和其他量的极大似然估计。

在我们的例子中，假设用交叉验证自适应地选择定义 B 样条的纽结的数和位置，而不是预先固定它们。用 λ 表示纽结的集合和它们的位置，则标准误差和置信带将解决 λ 的自适应选择，但无法解析地做这件事。使用自助法，计算 B 样条光滑，对每个自助法样本自适应地选择纽结。结果曲线的百分位从目标中的噪声和 $\hat{\lambda}$ 中捕获变异性。在这个特殊的例子中，置信带(没有显示)与固定的 λ 带比看起来没有太大差别。但在其他更多地使用自适应的问题中，这可能具有值得注意的重要影响。

8.3 贝叶斯方法

在用于推理的贝叶斯方法中，我们为给定参数的数据指定抽样模型 $\Pr(\mathbf{Z} \mid \theta)$(密度或概率质量函数)，并为参数 $\Pr(\theta)$指定先验分布，这些参数反映我们看到数据之前关于 θ 的知识。然后，计算后验分布：

$$\Pr(\theta|\mathbf{Z}) = \frac{\Pr(\mathbf{Z}|\theta)\cdot\Pr(\theta)}{\int \Pr(\mathbf{Z}|\theta)\cdot\Pr(\theta)\mathrm{d}\theta} \tag{8.23}$$

它表示在我们看到数据之后关于 θ 更新的知识。为了理解这种后验分布，可以由它选样，或者通过计算其均值或众数来汇总。贝叶斯方法与标准的("频率论的")推理方法不同，它使用

一个先验分布来表达看到数据之前的不确定性，并在看到数据之后允许残余的不确定性以后验分布形式来表示。

通过如下预测分布(predictive distribution)：

$$\Pr(z^{new}|\mathbf{Z}) = \int \Pr(z^{new}|\theta)\cdot\Pr(\theta|\mathbf{Z})\mathrm{d}\theta \tag{8.24}$$

后验分布还为预测未来观测 z^{new} 的值提供了基础。

相反，极大似然方法使用极大似然估计计算的数据密度 $\Pr(z^{new}|\hat{\theta})$ 预测未来的数据。与预测分布(8.24)不同，该方法不会解决估计 θ 的不确定性。

让我们用光滑例子，大致看一看贝叶斯方法。从式(8.5)给出的参数模型开始，并假设此时 σ^2 是已知的。假设观测到的特征值 x_1，x_2，…，x_N 是固定的，从而数据中的随机性惟一地来自在其均值$\mu(x)$附近变化的y。

我们需要的第二个要素是先验分布。函数上的分布相当复杂：一种方法是使用高斯过程先验，在其中指定任意两个函数值$\mu(x)$和$\mu(x')$的先验协方差(Wahba，1990；Neal，1996)。

这里，我们采取一种较简单的方法：通过考虑$\mu(x)$的一个有限的 B 样条基，可以提供系数 β 的一个先验分布，而且这还隐含地定义了一个$\mu(x)$的先验分布。我们选择中心在 0 处的高斯先验分布：

$$\beta \sim N(0, \tau\mathbf{\Sigma}) \tag{8.25}$$

先验相关矩阵 $\mathbf{\Sigma}$ 和方差τ的选择在下面讨论。$\mu(x)$的隐过程先验也因此是高斯的，具有协方差核：

$$\begin{aligned} K(x,x') &= \mathrm{cov}[\mu(x),\mu(x')] \\ &= \tau\cdot h(x)^T\mathbf{\Sigma}h(x') \end{aligned} \tag{8.26}$$

β 的后验分布也是高斯的，均值和协方差是：

$$\begin{aligned} \mathrm{E}(\beta|\mathbf{Z}) &= \left(\mathbf{H}^T\mathbf{H} + \frac{\sigma^2}{\tau}\mathbf{\Sigma}^{-1}\right)^{-1}\mathbf{H}^T\mathbf{y} \\ \mathrm{cov}(\beta|\mathbf{Z}) &= \left(\mathbf{H}^T\mathbf{H} + \frac{\sigma^2}{\tau}\mathbf{\Sigma}^{-1}\right)^{-1}\sigma^2 \end{aligned} \tag{8.27}$$

$\mu(x)$相应的后验值是：

$$\begin{aligned} \mathrm{E}(\mu(x)|\mathbf{Z}) &= h(x)^T\left(\mathbf{H}^T\mathbf{H} + \frac{\sigma^2}{\tau}\mathbf{\Sigma}^{-1}\right)^{-1}\mathbf{H}^T\mathbf{y} \\ \mathrm{cov}[\mu(x),\mu(x')|\mathbf{Z}] &= h(x)^T\left(\mathbf{H}^T\mathbf{H} + \frac{\sigma^2}{\tau}\mathbf{\Sigma}^{-1}\right)^{-1}h(x')\sigma^2 \end{aligned} \tag{8.28}$$

我们怎样选择先验相关矩阵 $\mathbf{\Sigma}$ 呢？在某些情况下，该先验可以根据关于参数的主题知识来选择。这里要说的是，函数$\mu(x)$应该是光滑的，通过用 B 样条的低维光滑基表示μ已经保证了这一点。因此，可以取先验相关矩阵为单位矩阵 $\mathbf{\Sigma} = \mathbf{I}$。当基函数的数目较大时，这样做可能不够充分，而附加的光滑性可以通过对 $\mathbf{\Sigma}$ 强加一些限制来加强；这与光滑样条的情况完全一样(见第 5.8.1 节)。

图 8.3 显示了$\mu(x)$的对应先验的 10 条曲线。为了产生函数$\mu(x)$的后验值，我们从它的

后验(8.27)产生值 β'，给出对应的后验值 $\mu'(x) = \sum_1^7 \beta'_j h_j(x)$。图 8.4 中显示了 10 条这样的后验曲线。两个不同的值 1 和 1000 用于先验方差 τ。注意，图 8.4 的右图看上去和图 8.2 左下图中的自助法分布极其相似。这种相似性不是偶然的。当 $\tau \to \infty$ 时，后验分布(8.27)和自助法分布(8.7)一致。另一方面，当 $\tau = 1$ 时，图 8.4 左图中的后验曲线 $\mu(x)$ 比自助法曲线光滑，因为我们已经在光滑性上强加了更多的先验权值。

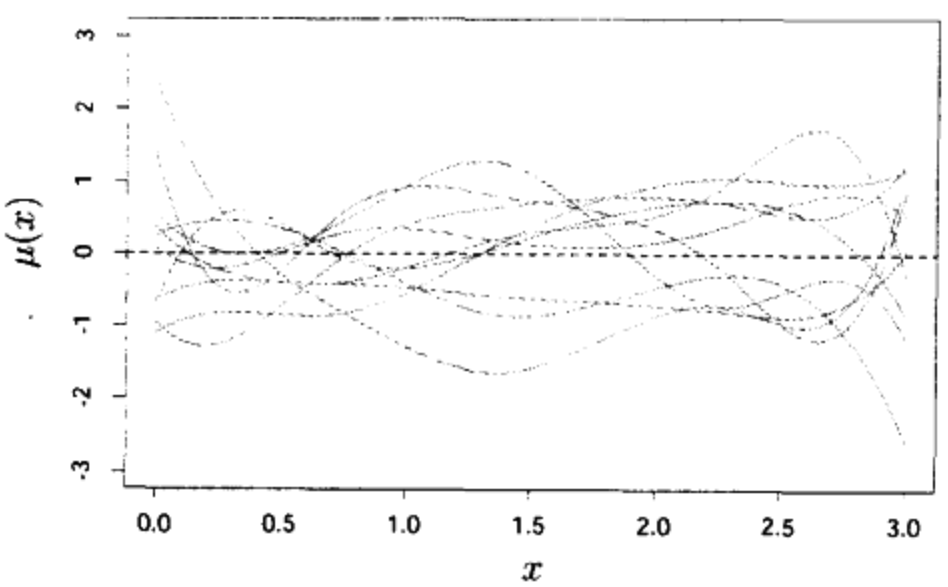


图 8.3 光滑例子：函数 $\mu(x)$ 的高斯先验分布的 10 条曲线

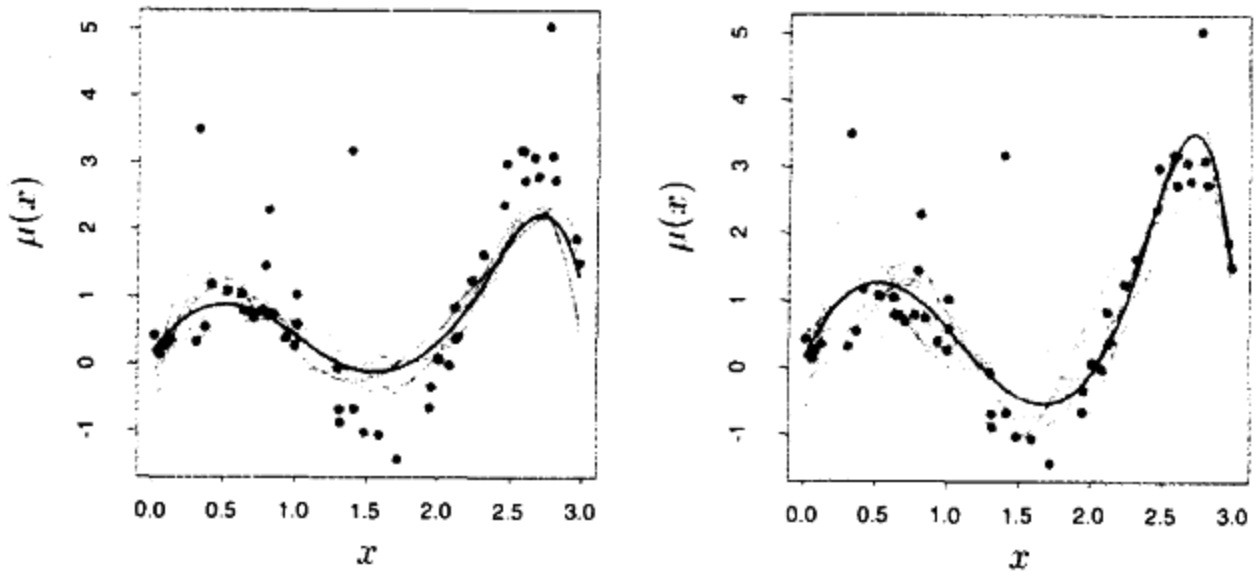


图 8.4 光滑例子：对于先验方差 τ 的两个不同的值，函数 $\mu(x)$ 的后验分布的 10 条曲线。紫色曲线是后验均值(见彩页)

当 $\tau \to \infty$ 时，分布(8.25)称为 θ 的无信息先验(noninformative prior)。在高斯模型中，极大似然和参数自助法分析与对自由参数使用无信息先验的贝叶斯分析趋于一致。这是因为使用常量先验，后验分布与似然成正比。这种对应也能扩展到非参数情形，那里，非参数自助法与无信息贝叶斯分析近似；第 8.4 节给出了详细的描述。

然而，从贝叶斯的观点看，我们已经做了一些不合适事情。我们使用了 σ^2 的无信息(常量)先验，并在后验中用极大似然估计 $\hat{\sigma}^2$ 代替它。更标准的贝叶斯分析也在 σ 上置一个先验(典型地，$g(\sigma) \propto 1/\sigma$)，计算 $\mu(x)$ 和 σ 的联合后验，然后取消 σ，而不是仅取出后验分布的极大值("MAP"估计)。

8.4　自助法和贝叶斯推理之间的联系

首先考虑一个非常简单的例子。在该例中，我们从如下正态分布中观测单个观测 z：

$$z \sim N(\theta, 1) \tag{8.29}$$

为了对 θ 进行贝叶斯分析，我们需要指定一个先验。最方便和最常见的选择可能是 $\theta \sim N(0, \tau)$，它给出后验分布：

$$\theta|z \sim N\left(\frac{z}{1+1/\tau}, \frac{1}{1+1/\tau}\right) \tag{8.30}$$

现在，我们取的 τ 越大，后验就变得越集中在极大似然估计 $\hat{\theta} = z$ 周围。随 $\tau \to \infty$ 取极限，获得一个无信息（常量）先验，而后验分布是：

$$\theta|z \sim N(z, 1) \tag{8.31}$$

这和参数自助法分布相同。在参数自助法分布下，我们从选样密度 $N(z,1)$ 的极大似然估计中产生自助法值 z^*。

有三种要素使得这种对应起作用：

1. θ 的无信息先验的选取。
2. 对数似然 $\ell(\theta; \mathbf{Z})$ 对数据 $\mathbf{Z}$ 的依赖性仅通过极大似然估计 $\hat{\theta}$，因此我们可以记该对数似然为 $\ell(\theta; \hat{\theta})$。
3. θ 和 $\hat{\theta}$ 的对数似然的对称性，即 $\ell(\theta; \hat{\theta}) = \ell(\hat{\theta}; \theta) +$ 常量。

性质 2 和性质 3 本质上仅对高斯分布成立。然而，这些性质对多项式分布也近似地成立，导致非参数自助法和贝叶斯推理的对应，我们在下面加以阐述。

假设有一个离散样本空间，具有 L 个类。令 w_j 是样本点落在类 j 中的概率，$\hat{w}_j$ 是类 j 的观测比例。令 $w = (w_1, w_2, \cdots, w_L)$，$\hat{w} = (\hat{w}_1, \hat{w}_2, \cdots, \hat{w}_L)$。记我们的估计子为 $S(\hat{w})$；看做 w 的先验分布，具有参数 a 的对称狄利克雷（Dirichlet）分布：

$$w \sim \mathrm{Di}_L(a1) \tag{8.32}$$

即，先验概率质量函数正比于 $\prod_{\ell=1}^{L} w_\ell^{a-1}$。则 w 的后验密度是：

$$w \sim \mathrm{Di}_L(a1 + N\hat{w}) \tag{8.33}$$

其中，N 是样本容量。令 $a \to 0$，得到无信息先验：

$$w \sim \mathrm{Di}_L(N\hat{w}) \tag{8.34}$$

现在，通过从数据中有放回地抽样得到的自助法分布可以表示为从多项式分布中抽样分类比例。特殊地，

$$N\hat{w}^* \sim \mathrm{Mult}(N, \hat{w}) \tag{8.35}$$

其中，$\mathrm{Mult}(N, \hat{w})$ 表示多项式分布，具有概率质量函数 $\left(N\hat{w}_1^*, \overset{N}{\cdots}, N\hat{w}_L^*\right) \prod \hat{w}_\ell^{N\hat{w}_\ell^*}$。该分布类似于上面的后验分布，有相同的支集、相同的均值和几乎相同的协方差矩阵。因此，$S(\hat{w}^*)$ 的自助法分布将逼近 $S(w)$ 的后验分布。

在这种意义上，自助法分布为我们的参数提供了一个(近似的)非参数的、无信息的后验分布。但这种自助法分布是不费力获得的——不必形式地指定先验分布，也不必从后验分布中抽样。因此，可以把自助法分布看做是"穷人的"贝叶斯后验。通过扰动数据，自助法逼近扰动参数的贝叶斯效应，而且实现更加简单。

8.5 EM 算法

对于简化较难的极大似然问题，EM 算法是一种常用工具。首先，我们用简单的混合模型介绍它。

8.5.1 二分量混合模型

本节，我们介绍一种用于密度估计的简单混合模型，以及用于实现极大似然估计相关的 EM 算法。这与贝叶斯推理的 Gibbs 选样方法有自然的联系。混合模型在本书的多个章节中论述过，特别是在第 6.8 节、第 12.7 节和第 13.2.3 节。

图 8.5 的左图显示了表 8.1 中的 20 个虚构数据点的直方图。

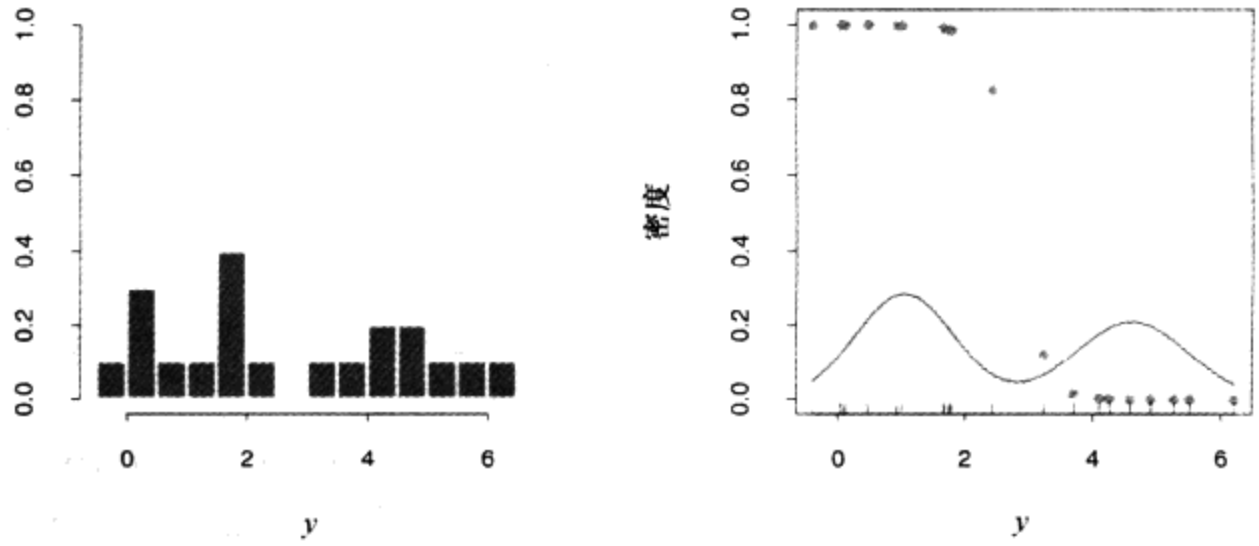


图 8.5 混合例子。左图：数据的直方图。右图：高斯密度(实线)和观测左 y 的分量密度的响应度(虚线和圆点)的极大似然拟合，是 y 的函数

表 8.1 用于图 8.5 中二分量混合模型例子的 20 个虚构数据点

-0.39	0.12	0.94	1.67	1.76	2.44	3.72	4.28	4.92	5.53
0.06	0.48	1.01	1.68	1.80	3.25	4.12	4.60	5.28	6.22

我们要对数据点的密度建模。由于明显的双峰性，高斯分布并不合适。这里似乎有两种分离的基本体制，因此我们用两个正态分布混合对 Y 建模：

$$\begin{aligned} Y_1 &\sim N(\mu_1, \sigma_1^2) \\ Y_2 &\sim N(\mu_2, \sigma_2^2) \\ Y &= (1-\Delta)\cdot Y_1 + \Delta \cdot Y_2 \end{aligned} \tag{8.36}$$

其中，$\Delta \in \{0, 1\}$，$\Pr(\Delta = 1) = \pi$。这个生成表示是清楚的：以概率 π 产生 $\Delta \in \{0,1\}$，然后，依赖于结果，产生 Y_1 或 Y_2。令 $\phi_\theta(x)$ 表示正态密度，参数为 $\theta = (\mu, \sigma^2)$。则 Y 的密度是：

$$g_Y(y) = (1-\pi)\phi_{\theta_1}(y) + \pi\phi_{\theta_2}(y) \tag{8.37}$$

现在，假设我们希望通过极大似然用该模型拟合图 8.5 中的数据。参数是：

$$\theta = (\pi, \theta_1, \theta_2) = (\pi, \mu_1, \sigma_1^2, \mu_2, \sigma_2^2) \tag{8.38}$$

基于 N 个训练实例的对数似然是：

$$\ell(\theta; \mathbf{Z}) = \sum_{i=1}^{N} \log[(1-\pi)\phi_{\theta_1}(y_i) + \pi\phi_{\theta_2}(y_i)] \tag{8.39}$$

由于需要求对数中的项的和，直接极大化 $\ell(\theta; \mathbf{Z})$ 是相当困难的。然而，我们有一种比较简单的方法。与式(8.36)一样，我们考虑取值 0 或 1 的本征变量 Δ_i：如果 $\Delta_i = 1$，则 Y_i 取自模型 2，否则取自模型 1。假设我们知道诸 Δ_i 的值，则对数似然将是：

$$\begin{aligned} \ell_0(\theta; \mathbf{Z}, \boldsymbol{\Delta}) &= \sum_{i=1}^{N} [(1-\Delta_i)\log\phi_{\theta_1}(y_i) + \Delta_i \log\phi_{\theta_2}(y_i)] \\ &\quad + \sum_{i=1}^{N} [(1-\Delta_i)\log\pi + \Delta_i \log(1-\pi)] \end{aligned} \tag{8.40}$$

且 μ_1 和 σ_1^2 的极大似然估计值将是 $\Delta_i = 0$ 的那些数据样本的均值和方差；类似地，μ_2 和 σ_2^2 的将是 $\Delta_i = 1$ 的那些数据样本的均值和方差。

由于诸 Δ_i 的值实际上是未知的，所以我们以迭代方式来处理，用如下期望值替换式(8.40)中的每个 Δ_i：

$$\gamma_i(\theta) = \mathrm{E}(\Delta_i | \theta, \mathbf{Z}) = \Pr(\Delta_i = 1 | \theta, \mathbf{Z}) \tag{8.41}$$

该式也称为模型 2 关于观测 i 的响应度。对于该高斯混合分布的特殊情形，我们使用一个称做 EM 算法的过程。该过程在算法 8.1 中给出。在期望步，我们对每个模型做每个观测的软赋值：根据每个模型下训练点的相对密度，使用参数的当前估计对响应度赋值。在极大化步，这些响应度用于加权极大似然拟合，以更新参数的估计。

算法 8.1　二分量高斯混合的 EM 算法

1. 取参数 $\hat{\mu}_1, \hat{\sigma}_1^2, \hat{\mu}_2, \hat{\sigma}_2^2, \hat{\pi}$ 的初值(参见正文)

2. 期望步：计算响应度：

$$\hat{\gamma}_i = \frac{\hat{\pi}\phi_{\hat{\theta}_2}(y_i)}{(1-\hat{\pi})\phi_{\hat{\theta}_1}(y_i) + \hat{\pi}\phi_{\hat{\theta}_2}(y_i)}, \quad i = 1, 2, \cdots, N \tag{8.42}$$

3. 极大化步：计算加权均值和方差：

$$\hat{\mu}_1 = \frac{\sum_{i=1}^{N}(1-\hat{\gamma}_i)y_i}{\sum_{i=1}^{N}(1-\hat{\gamma}_i)} \qquad \hat{\sigma}_1^2 = \frac{\sum_{i=1}^{N}(1-\hat{\gamma}_i)(y_i-\hat{\mu}_1)^2}{\sum_{i=1}^{N}(1-\hat{\gamma}_i)}$$

$$\hat{\mu}_2 = \frac{\sum_{i=1}^{N}\hat{\gamma}_i y_i}{\sum_{i=1}^{N}\hat{\gamma}_i} \qquad \hat{\sigma}_2^2 = \frac{\sum_{i=1}^{N}\hat{\gamma}_i(y_i-\hat{\mu}_1)^2}{\sum_{i=1}^{N}\hat{\gamma}_i}$$

和混合概率 $\hat{\pi} = \sum_{i=1}^{N}\hat{\gamma}_i / N$

4. 重复步骤 2 和步骤 3 直到收敛

构造$\hat{\mu}_1$和$\hat{\mu}_2$的初值的一个好方法是简单地随机选择两个y_i。可以置$\hat{\sigma}_1^2$和$\hat{\sigma}_2^2$等于整体样本方差$\sum_{i=1}^{N}(y_i-\bar{y})^2/N$。混合比例$\hat{\pi}$可以从值0.5开始。

注意，当我们在任意数据上设置一个无限高度尖峰时，即对某个i，$\hat{\mu}_i=y_i$，并且$\hat{\sigma}_i^2=0$时，似然的实际极大化才出现。这给出了无穷似然，但不是一个有用的解。因此，我们实际上在寻找该似然的一个好的局部极大值，满足$\hat{\sigma}_1^2,\hat{\sigma}_2^2>0$。对更复杂的情况，可能有多个满足$\hat{\sigma}_1^2,\hat{\sigma}_2^2>0$的局部极大值。在我们的例子中，用参数的一些不同初值(都满足$\hat{\sigma}_k^2>0.5$)运行EM算法，并选取产生最高的极大化似然的那个运行。图8.6显示了EM算法极大化对数似然的进展。表8.2上显示$\pi=\sum_i\hat{\gamma}_i/N$，它是在选取的EM算法的迭代时类2中的观测所占比例的极大似然估计。

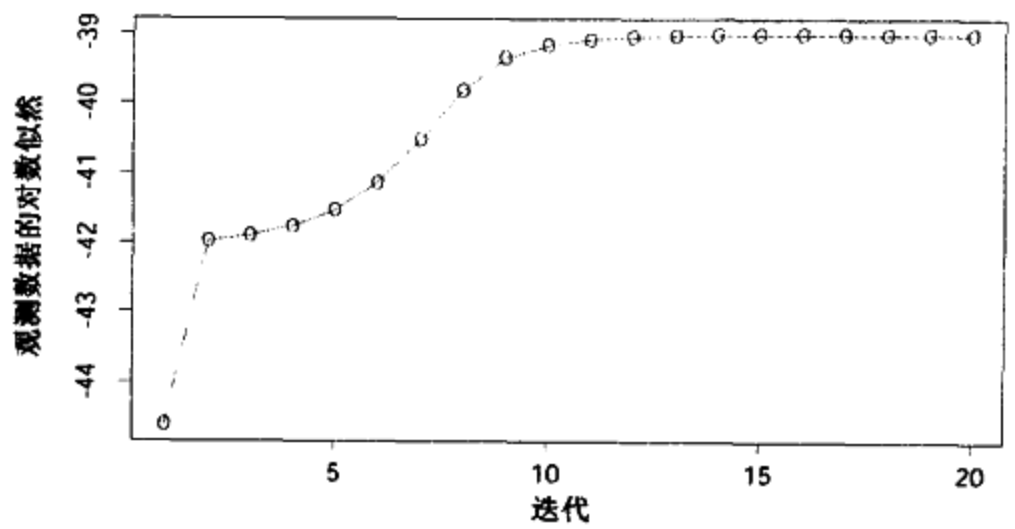


图8.6 EM算法：观测数据的对数似然，作为迭代次数的函数

表8.2 关于混合例子的EM算法的选择的迭代

迭代	$\hat{\pi}$
1	0.485
5	0.493
10	0.523
15	0.544
20	0.546

最终的极大似然估计是：

$$\hat{\mu}_1=4.62 \qquad \hat{\sigma}_1^2=0.87$$
$$\hat{\mu}_2=1.06 \qquad \hat{\sigma}_2^2=0.77$$
$$\hat{\pi}=0.546$$

图8.5右图显示通过该过程估计的高斯混合密度(红色实线)和响应度(绿色虚线)。注意，混合模型对有指导学习也是有用的；在第6.7节，我们展示了高斯混合模型如何导出径向基函数的版本。

8.5.2 通用EM算法

上述过程是EM(或Baum-Welch)算法的一个例子。EM算法用于在某些问题类中似然的

极大化。这些问题的似然极大化是困难的，但通过使用本征（未观测到的）数据加大样本可以使其简单些。这叫做数据增广（data augmentation）。这里，本征数据是模型隶属关系 Δ_i。在其他问题中，本征数据是实际数据，它们应当被观测，但被遗漏了。

算法 8.2 给出了 EM 算法的一般形式。我们的观测数据是 $\mathbf{Z}$，具有依赖于参数 θ 的对数似然 $\ell(\theta;\mathbf{Z})$。本征或遗漏的数据是 $\mathbf{Z}^m$，所以完全数据是 $\mathbf{T}=(\mathbf{Z},\mathbf{Z}^m)$，具有对数似然 $\ell_0(\theta;\mathbf{T})$，$\ell_0$ 基于完全密度。在混合问题中，$(\mathbf{Z},\mathbf{Z}^m)=(\mathbf{y},\Delta)$，$\ell_0(\theta;\mathbf{T})$在式(8.40)中给出。

算法 8.2　EM 算法

1. 由参数的初值 $\hat{\theta}^{(0)}$ 开始
2. 期望步：在第 j 步，计算：

$$Q(\theta',\hat{\theta}^{(j)})=\mathrm{E}(\ell_0(\theta';\mathbf{T})|\mathbf{Z},\hat{\theta}^{(j)}) \tag{8.43}$$

作为哑元 θ' 的函数

3. 极大化步：确定新的估计 $\hat{\theta}^{(j+1)}$，作为 θ' 上的 $Q(\theta',\hat{\theta}^{(j)})$的极大化
4. 重复步骤 2 和步骤 3 直到收敛

在我们的混合模型例子中，$\mathrm{E}(\ell_0(\theta';\mathbf{T})|\mathbf{Z},\hat{\theta}^{(j)})$就是简单地将式(8,40)中的 Δ_i 用响应度 $\hat{\gamma}_i(\hat{\theta})$替换，步骤 3 的极大化正好是加权的均值和方差。

现在，给出 EM 算法在一般情况下起作用的原因。

由于

$$\Pr(\mathbf{Z}^m|\mathbf{Z},\theta')=\frac{\Pr(\mathbf{Z}^m,\mathbf{Z}|\theta')}{\Pr(\mathbf{Z}|\theta')} \tag{8.44}$$

我们有：

$$\Pr(\mathbf{Z}|\theta')=\frac{\Pr(\mathbf{T}|\theta')}{\Pr(\mathbf{Z}^m|\mathbf{Z},\theta')} \tag{8.45}$$

用对数似然表示，我们有 $\ell(\theta';\mathbf{Z})=\ell_0(\theta';\mathbf{T})-\ell_1(\theta';\mathbf{Z}^m|\mathbf{Z})$，其中，$\ell_1$ 基于条件密度 $\Pr(\mathbf{Z}^m|\mathbf{Z},\theta')$。关于参数 θ 支配的 $\mathbf{T}|\mathbf{Z}$ 的分布取条件期望给出：

$$\begin{aligned}\ell(\theta';\mathbf{Z}) &= \mathrm{E}[\ell_0(\theta';\mathbf{T})|\mathbf{Z},\theta]-\mathrm{E}[\ell_1(\theta';\mathbf{Z}^m|\mathbf{Z})|\mathbf{Z},\theta]\\ &\equiv Q(\theta',\theta)-R(\theta',\theta)\end{aligned} \tag{8.46}$$

在 M(极大化)步，EM 算法在 θ' 上极大化 $Q(\theta',\theta)$，而不是实际的目标函数 $\ell(\theta';\mathbf{Z})$。为什么它能成功地极大化 $\ell(\theta';\mathbf{Z})$呢？注意，$R(\theta^*,\theta)$是密度的对数似然的期望值(用 θ^* 标引)，涉及 θ 标引的相同密度，因此，当 $\theta^*=\theta$ 时（根据 Jensen 不等式）作为 θ^* 的函数被极大化(见习题 8.2)。这样，如果 θ' 极大化 $Q(\theta',\theta)$，则有：

$$\begin{aligned}\ell(\theta';\mathbf{Z})-\ell(\theta;\mathbf{Z}) &= [Q(\theta',\theta)-Q(\theta,\theta)]-[R(\theta',\theta)-R(\theta,\theta)]\\ &\geqslant 0\end{aligned} \tag{8.47}$$

因此，EM 迭代永远也不会降低对数似然。

这也清楚地表明在 M 步完全极大化不是必需的：我们只需要找到一个值 $\hat{\theta}^{(j+1)}$，使得作为它的第一个自变量的函数 $Q(\theta',\hat{\theta}^{(j)})$是递增的，即 $Q(\hat{\theta}^{(j+1)},\hat{\theta}^{(j)})>Q(\hat{\theta}^{(j)},\hat{\theta}^{(j)})$。该过程称为 GEM(广义 EM)算法。

8.5.3 作为极大化 - 极大化过程的 EM

从不同的角度，可以将 EM 过程看做联合极大化算法。考虑函数：

$$F(\theta', \tilde{P}) = \mathrm{E}_{\tilde{P}}[\ell_0(\theta'; \mathbf{T})] - \mathrm{E}_{\tilde{P}}[\log \tilde{P}(\mathbf{Z}^m)] \tag{8.48}$$

这里，$\tilde{P}(\mathbf{Z}^m)$是本征数据 $\mathbf{Z}^m$ 上的任意分布。在混合模型例子中，$\tilde{P}(\mathbf{Z}^m)$包含概率 $\gamma_i = \Pr(\Delta = 1 | \theta, \mathbf{Z})$的集合。注意，由式(8.46)[①]，在 $\tilde{P}(\mathbf{Z}^m) = \Pr(\mathbf{Z}^m | \mathbf{Z}, \theta')$上求值的 F 是观测数据的对数似然。函数 F 扩展对数似然的定义域，使得它的极大化变得更加容易。

通过固定一个自变量而在另一个上极大化，EM 算法可以看做是 θ' 上的 F 和 $\tilde{P}(\mathbf{Z}^m)$的联合极大化方法。对于固定的 θ'，$\tilde{P}(\mathbf{Z}^m)$上的极大化可以表示为：

$$\tilde{P}(\mathbf{Z}^m) = \Pr(\mathbf{Z}^m | \mathbf{Z}, \theta') \tag{8.49}$$

(见习题 8.3)。这是由 E(期望)步[例如，混合模型例子中的式(8.42)]计算的分布。在 M(极大化)步，我们固定 $\tilde{P}$，在 θ' 上对 $F(\theta', \tilde{P})$极大化：由于第二项不包括 θ'，所以它与对第一项 $E_{\tilde{P}}[\ell_0(\theta'; \mathbf{T}) | \mathbf{Z}, \theta]$的极大化相同。

最后，由于当 $\tilde{P}(\mathbf{Z}^m) = \Pr(\mathbf{Z}^m | \mathbf{Z}, \theta')$时，$F(\theta', \tilde{P})$和观测数据的对数似然一致，所以前者的极大化实现了后者的极大化。图 8.7 显示了这一过程。EM 算法的这一观点导致了轮流极大化过程。例如，不必一次性极大化全部本征数据参数，而可以在 M 步轮流地一次极大化它们中的一个。

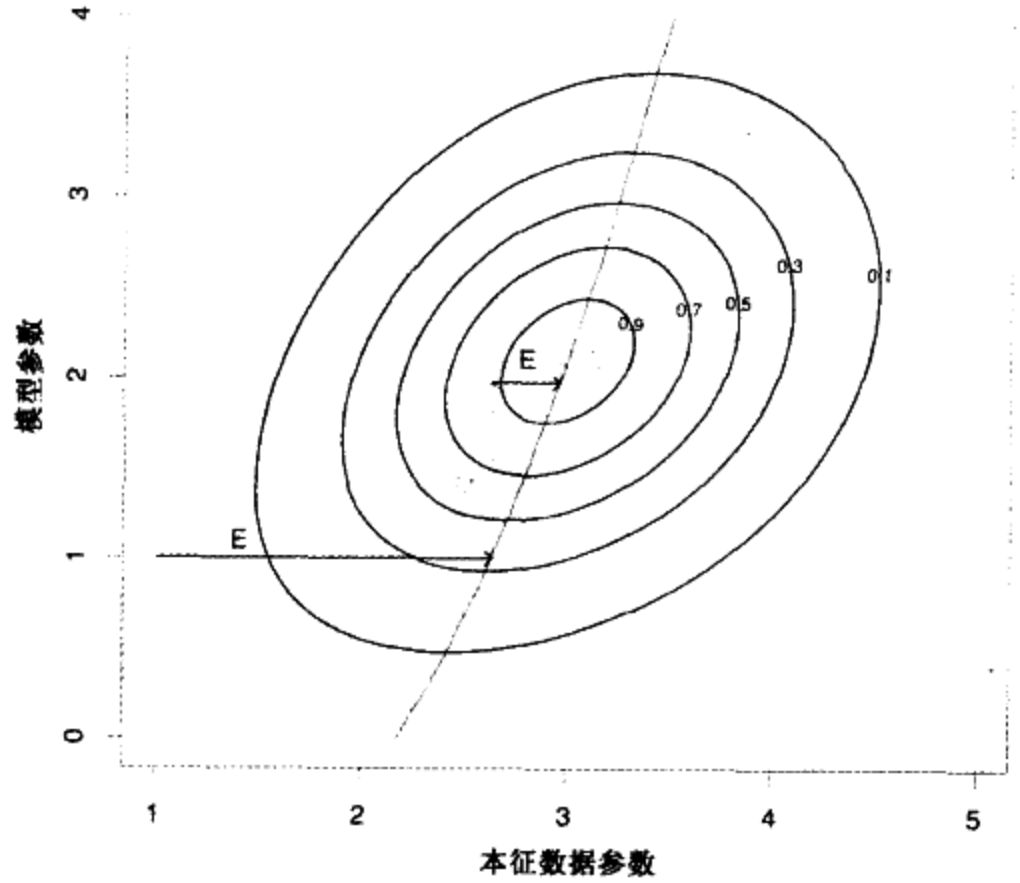


图 8.7 EM 算法的极大化 - 极大化观点。图中显示的是(增广的)观测数据的对数似然 $F(\theta', \tilde{P})$的周线。E 步等价于极大化本征数据分布的参数上的对数似然。M步在对数似然的参数上对其极大化。曲线对应于观测数据的对数似然，是通过对每个θ'值极大化$F(\theta', \tilde{P})$得到的曲线

① 式(8.46)对所有的 θ 成立，包括 $\theta = \theta'$。

8.6　从后验中抽样的 MCMC

已经定义了贝叶斯模型，我们想从结果后验分布中抽取样本，以便进行关于参数的推理。除简单模型外，这通常是一个困难的计算问题。本节，我们讨论后验抽样的马尔可夫链蒙特卡罗(Markov chain MonTe Carlo, MCMC)方法。我们将看到 Gibbs 抽样，一个 MCMC 过程，与 EM 算法有紧密的联系：其主要差别在于它从条件分布中抽样，而不是对它们极大化。

首先考虑如下抽象问题。我们有随机变量 $U_1, U_2, \cdots, U_K$，而希望从它们的联合分布中抽取一个样本。假设这很难做，但容易从条件分布 $\Pr(U_j \mid U_1, U_2, \cdots, U_{j-1}, U_{j+1}, \cdots, U_K)$，$j = 1, 2, \cdots, K$ 模拟。Gibbs 取样过程从每个分布轮流地选择一个来模拟，并且当该过程稳定时，提供联合分布的一个样本。该过程在算法 8.3 中定义。

算法 8.3　Gibbs 取样法

1. 取初值 $U_k^{(0)}, k = 1, 2, \cdots, K$
2. 对于 $t = 1, 2, \cdots$ 重复：
   对于 $k = 1, 2, \cdots, K$，从下式产生 $U_k^{(t)}$：
   $\Pr(U_k^{(t)} \mid U_1^{(t)}, \cdots, U_{k-1}^{(t)}, U_{k+1}^{(t-1)}, \cdots, U_K^{(t-1)})$
3. 继续步骤 2，直到 $(U_1^{(t)}, U_2^{(t)}, \cdots, U_K^{(t)})$ 的联合分布不再改变

在正则条件下，可以证明该过程最终稳定，并且结果随机变量确实是 $U_1, U_2, \cdots, U_K$ 的联合分布的一个样本。尽管对不同的 t，样本 $(U_1^{(t)}, U_2^{(t)}, \cdots, U_K^{(t)})$ 显然不是独立的，但还是可以得到联合分布的一个样本。更形式地，Gibbs 抽样过程产生一个马尔可夫链，它的平稳分布是实际的联合分布，因此有术语“马尔可夫链蒙特卡罗”。毫不奇怪，实际的联合分布在该过程下是平稳的，因为后继步骤使得诸 U_k 的边缘分布保持不变。

注意，我们不需要知道条件密度的显示形式，但确实要能从它们中抽样。当过程达到平稳之后，变量的任意子集的边缘密度都可以用样本值的密度估计来近似。然而，如果有条件密度 $\Pr(U_k \mid U_\ell, \ell \neq k)$ 的显示形式，U_k 的边缘密度的更好估计可以从下式得到(见习题 8.4)：

$$\widehat{\Pr}_{U_k}(u) = \frac{1}{(M - m + 1)} \cdot \sum_{t=m}^{M} \Pr(u \mid U_\ell^{(t)}, \ell \neq k) \tag{8.50}$$

这里，我们已对序列的最后 $M - m + 1$ 个成员求平均值，在达到平稳以前考虑了初始“烙印”阶段。

现在，再回到贝叶斯推理。我们的目标是：给定数据 $\mathbf{Z}$，从参数的联合后验中抽取样本。如果给定其他参数和 $\mathbf{Z}$，能容易地从每个参数的条件分布中抽样，则 Gibbs 抽样将是用的。一个例子，关于高斯混合分布问题，在下面详述。

后验的 Gibbs 抽样与指数族模型的 EM 算法之间有着紧密的联系。其关键是将 EM 过程的本征数据 $\mathbf{Z}^m$ 看做 Gibbs 抽样法的另一个参数。对于高斯混合分布问题，我们把参数取为 $(\theta, \mathbf{Z}^m)$。为简单起见，我们固定方差 σ_1^2, σ_2^2 和它们的极大似然值的混合比例 π，使得 θ 中的未知参数只有均值 μ_1 和 μ_2。混合分布问题的 Gibbs 抽样法在算法 8.4 中给出。我们看到，除进行抽样而不是极大化外，步骤 2(a)和步骤 2(b)与 EM 过程的步骤 E 和 M 是相同的。在步骤

2(a),Gibbs 抽样过程模拟来自分布 $\Pr(\Delta_i|\theta,\mathbf{Z})$中的本征数据 Δ_i,而不是计算极大似然的响应度 $\gamma_i = \mathrm{E}(\Delta_i|\theta,\mathbf{Z})$。在步骤 2(b),我们不是计算后验 $\Pr(\mu_1,\mu_2,\mathbf{\Delta}|\mathbf{Z})$的极大值,而是根据条件分布 $\Pr(\mu_1,\mu_2|\mathbf{\Delta},\mathbf{Z})$来模拟。

算法 8.4 混合分布的 Gibbs 抽样

1. 取初始值 $\theta^{(0)} = (\mu_1^{(0)},\mu_2^{(0)})$

2. 对于 $t = 1, 2,\cdots,$ 重复:

(a) 对于 $i = 1,2,\cdots,N$, 由式(8.42)产生 $\Delta_i^{(t)} \in \{0,1\}$, 其中 $\Pr(\Delta_i^{(t)} = 1) = \hat{\gamma}_i(\theta^{(t)})$

(b) 置

$$\hat{\mu}_1 = \frac{\sum_{i=1}^N (1-\Delta_i^{(t)})\cdot y_i}{\sum_{i=1}^N (1-\Delta_i^{(t)})}$$

$$\hat{\mu}_2 = \frac{\sum_{i=1}^N \Delta_i^{(t)}\cdot y_i}{\sum_{i=1}^N \Delta_i^{(t)}}$$

并产生$\mu_1^{(t)} \sim N(\hat{\mu}_1,\hat{\sigma}_1^2)$和$\mu_2^{(t)} \sim N(\hat{\mu}_2,\hat{\sigma}_2^2)$

3. 继续步骤 2,直到$(\mathbf{\Delta}^{(t)},\mu_1^{(t)},\mu_2^{(t)})$的联合分布不再改变

图 8.8 显示了 Gibbs 抽样的 200 次迭代,均值参数μ_1(下)和μ_2(上)在左图显示,类 2 观测的比例 $\sum_i \Delta_i/N$ 在右图显示。在每幅图中,水平虚线绘制在极大似然估计值$\hat{\mu}_1,\hat{\mu}_2$ 和 $\sum_i \hat{\gamma}_i/N$ 处。这些值似乎很快稳定下来,并且稳定地分布在极大似然值周围。

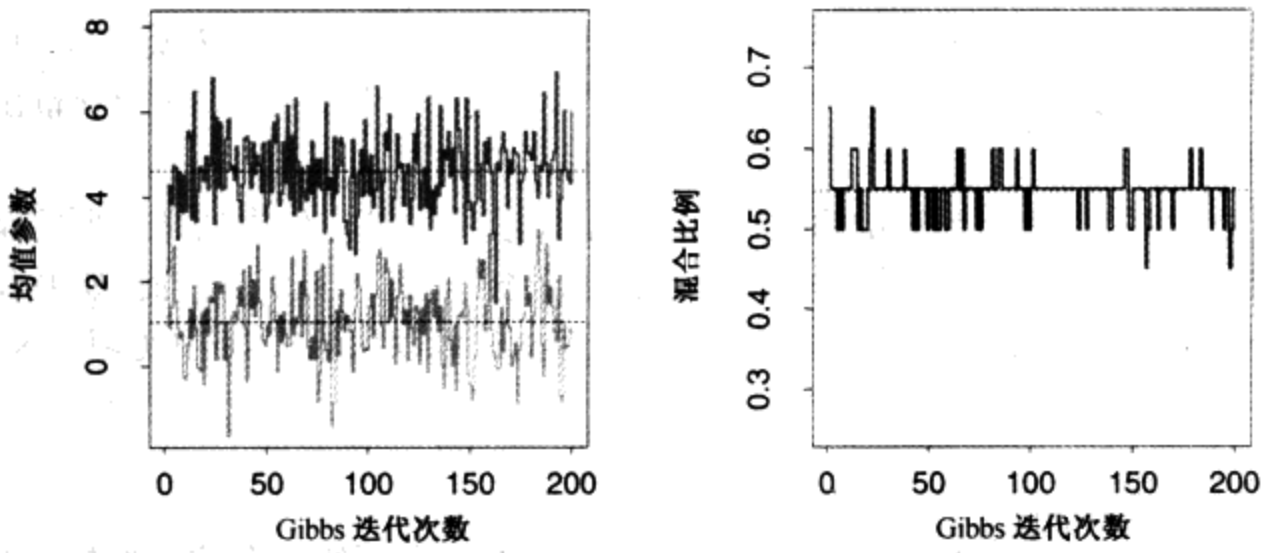


图 8.8 混合模型的例子。左图:Gibbs 抽样的两个均值参数的 200 个值;一条水平线绘制在极大化似然估计$\hat{\mu}_1,\hat{\mu}_2$ 处。右图:关于 200 个 Gibbs 抽样迭代,$\Delta_i = 1$的值所占的比例;一条水平线绘制在 $\sum_i \hat{\gamma}_i/N$处

上面的混合分布模型是一种简化模型,旨在清楚地解释 Gibbs 抽样和 EM 方法之间的联系。更实际地,我们应当考虑方差 σ_1^2,σ_2^2 和混合比例 π 上的先验分布,并采取不同的 Gibbs 抽样,从它们的后验分布抽样,在其他参数上取条件。也可以合并均值参数的真(有信息的)先验。这些先验不能是假的,否则会导致退化的后验,并且所有的混合权都加到一个分量上。

Gibbs 抽样仅仅是当前开发的从后验分布中抽样的若干过程中的一种。给定其他参数,它使用每个参数的条件抽样,并且当问题的结构使这种抽样易于实现时,它是有用的。其他方法

不需要这样的结构，如 Metropolis-Hastings 算法。这些方法和其他计算的贝叶斯方法已经应用于复杂的学习算法，如高斯过程模型和神经网络。详细内容可以在本章的文献注释中找到。

8.7　装袋

前面，我们引进的自助法是作为评估参数估计或预测精度的方法。这里将说明怎样使用自助法改进估计或预测本身。在第 8.4 节，我们考察了自助法和贝叶斯方法之间的联系，并且发现自助法均值近似于一个后验平均，装袋将进一步利用这种联系。

首先考虑回归问题。假设我们用一个模型拟合训练数据 $\mathbf{Z} = \{(x_1, y_1), (x_2, y_2), \cdots, (x_N, y_N)\}$，得到输入 x 上的预测 $\hat{f}(x)$。自助法聚集或装袋(bagging)对自助法样本集上的预测求平均，从而降低其方差。对每个自助法样本 $\mathbf{Z}^{*b}$，$b = 1, 2, \cdots, B$，我们用模型去拟合，产生预测 $\hat{f}^{*b}(x)$。装袋估计由下式定义：

$$\hat{f}_{\text{bag}}(x) = \frac{1}{B}\sum_{b=1}^{B}\hat{f}^{*b}(x) \tag{8.51}$$

用 $\hat{\mathcal{P}}$ 表示经验分布，它在每个数据点 (x_i, y_i) 上取相等的概率 $1/N$。事实上，“真”装袋估计由 $E_{\hat{\mathcal{P}}}\hat{f}^*(x)$ 定义，其中 $\mathbf{Z}^* = (x_1^*, y_1^*), (x_2^*, y_2^*), \cdots, (x_N^*, y_N^*)$，且每个 $(x_i^*, y_i^*) \sim \hat{\mathcal{P}}$。式(8.51)是真装袋估计的蒙特卡罗估计，当 $B \to \infty$ 时逼近它。

仅当原来的估计是非线性的，或者是数据的自适应函数时，装袋估计(8.51)与 $\hat{f}(x)$ 不同。例如，为了对第 8.2.1 节的 B 样条光滑装袋，我们在每个 x 值上对图 8.2 左下图中的曲线取平均值。如果我们固定输入，则 B 样条光滑在该数据上是线性的；因此，如果使用式(8.6)的参数自助法抽样，则当 $B \to \infty$ 时，$\hat{f}_{\text{bag}}(x) \to \hat{f}(x)$(见习题 8.5)。因此，装袋方法正好重新产生图 8.2左上图的原始光滑。如果使用非参数自助法装袋，则这种相同将是近似的。

一个更有趣的例子是回归树，在那里 $\hat{f}(x)$ 表示输入向量 x 上的树预测(回归树在第 9 章介绍)。典型地，每个自助法树将涉及不同于原始特征的特征，并且有不同数目的末端结点。装袋估计是 B 棵树在 x 上的平均预测。

现在，假设我们的树产生了一个 K-类响应的分类子 $\hat{G}(x)$。这里，考虑基本指示向量函数 $\hat{f}(x)$ 是有用的，该函数的值为一个 1 和 $K-1$ 个 0 的向量，使得 $\hat{G}(x) = \arg\max_k \hat{f}(x)$。那么，装袋的估计 $\hat{f}_{\text{bag}}(x)$ (8.51)是一个 K 向量 $(p_1, p_2, \cdots, p_K)$。其中 p_k 等于树在 x 处预测类 k 的比例。把这些看做类概率的估计，则我们预测的类就是 B 棵树中具有最高“得票”的那一个，$\hat{G}_{\text{bag}}(x) = \arg\max_k \hat{f}_{\text{bag}}(x)$。

对于许多分类子 $\hat{G}(x)$(包括树)，都存在基本函数 $\hat{f}(x)$，它估计 x 处的类概率。另一种装袋策略是对它们取平均，而不是用指示向量，并且这趋向于产生具有低方差的装袋估计，特别是对较小的 B 更是如此(见图 8.10)。

8.7.1　例：具有模拟数据的树

我们产生了一个容量 $N = 30$ 的样本，具有两个类和 $p = 5$ 个特征，每个遵守标准高斯分布，其两两间的相关性为 0.95。响应 Y 根据 $\Pr(Y = 1 \mid x_1 \leqslant 0.5) = 0.2, \Pr(Y = 1 \mid x_1 >$

0.5) = 0.8 产生。贝叶斯误差是 0.2。容量为 2000 的检验样本也从相同的总体中产生。我们用分类树(分类树将在第 9 章中说明)拟合训练样本和 200 个自助法样本。不使用剪枝。图 8.9显示原始树和 5 棵自助法树。注意,所有的树是不同的,具有不同的分裂特征和不同割点。原始树和装袋树的检验误差显示在图 8.10 中。在这个例子中,由于预测子的相关性,这些树具有较高的方差。装袋成功地光滑了这种方差,并因此而降低了检验误差。

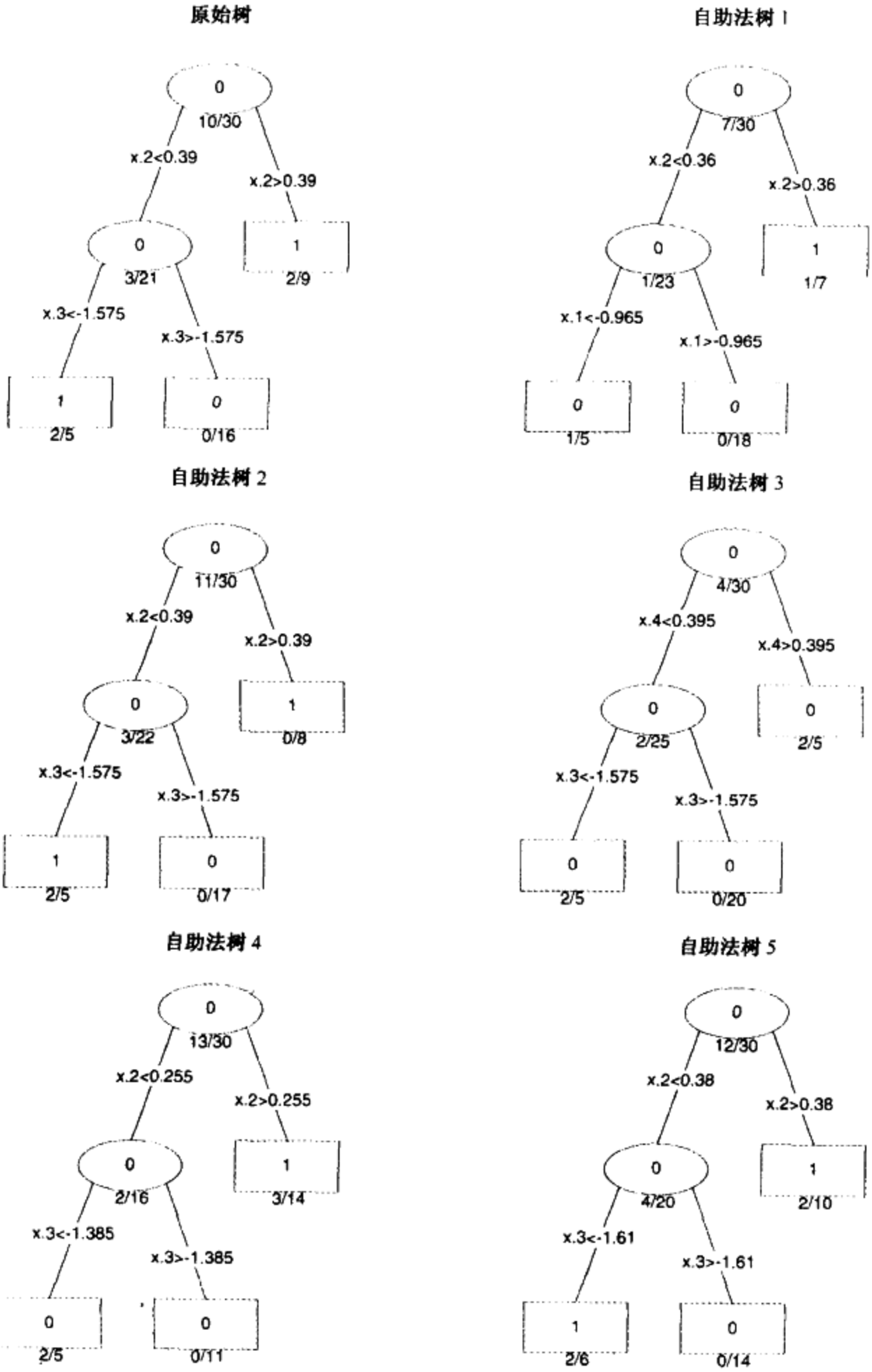


图 8.9 模拟数据集上的装袋树。左上图显示的是原始树。其余的图显示了自助法样本上的 5 棵树

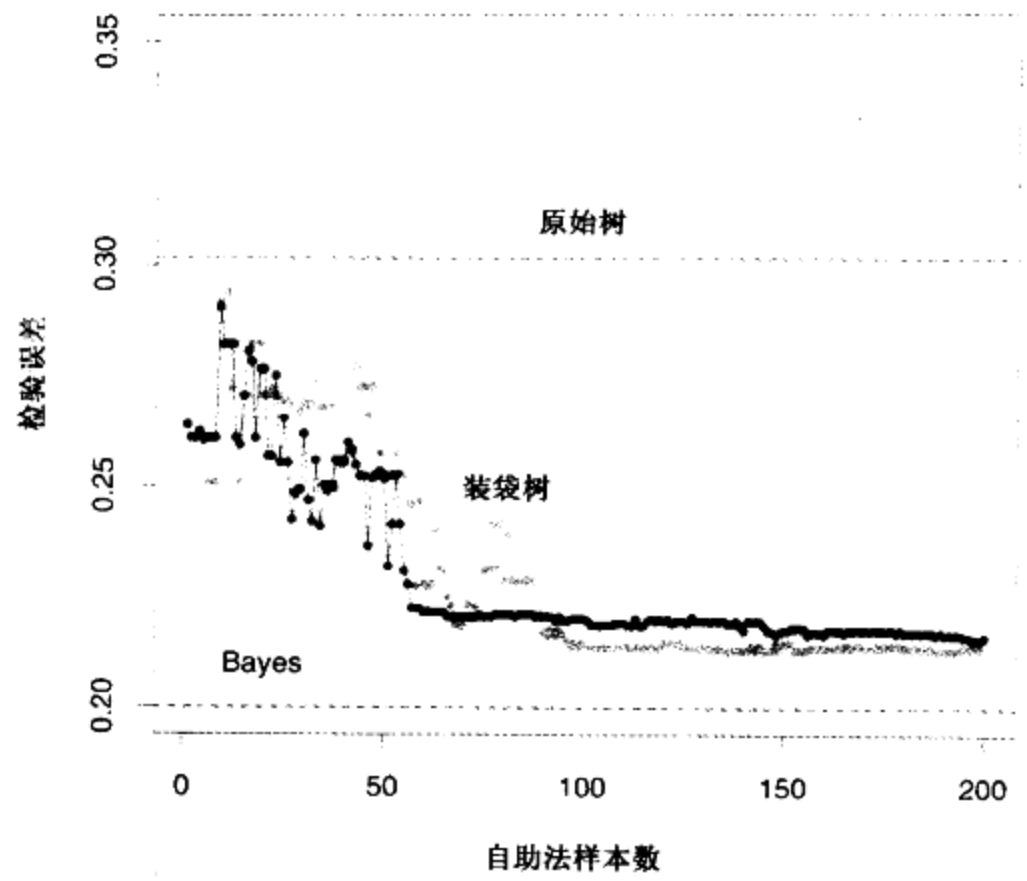


图 8.10　图 8.9 中装袋例子的误差曲线。所显示的是原始树和装袋树的检验误差，作为自助法样本数量的函数。绿色点对应于多数表决，而紫色点是概率的平均值(见彩页)

装袋可以大大降低不稳定过程(如树)的方差，导致预测性能的改进。一个简单的讨论可以表明为什么装袋在平方误差损失下会有所帮助。简单地说，是由于求平均会降低方差而保持偏倚不变。

假设我们的训练观测 (x_i, y_i)，$i = 1, 2, \cdots, N$ 是从分布 $\mathcal{P}$ 中独立抽取的，并考虑理想的聚集估计子 $f_{ag}(x) = \mathrm{E}_{\mathcal{P}} \hat{f}^*(x)$。这里，$x$ 是固定的，而自助法数据集 $\mathbf{Z}^*$ 包含从 $\mathcal{P}$ 中抽样的观测 x_i^*, y_i^*，$i = 1, 2, \cdots, N$。注意 $f_{ag}(x)$ 是一个装袋估计，它从实际总体 $\mathcal{P}$ 中而不是数据中抽取样本。它不是一个在实际中可以使用的估计，但对于分析很方便。我们有：

$$\begin{aligned} \mathrm{E}_{\mathcal{P}}[Y - \hat{f}^*(x)]^2 &= \mathrm{E}_{\mathcal{P}}[Y - f_{ag}(x) + f_{ag}(x) - \hat{f}^*(x)]^2 \\ &= \mathrm{E}_{\mathcal{P}}[Y - f_{ag}(x)]^2 + \mathrm{E}_{\mathcal{P}}[\hat{f}^*(x) - f_{ag}(x)]^2 \\ &\geqslant \mathrm{E}_{\mathcal{P}}[Y - f_{ag}(x)]^2 \end{aligned} \tag{8.52}$$

右侧的额外误差源于 $\hat{f}^*(x)$ 的方差，在其均值 $f_{ag}(x)$ 附近。因此，真总体聚集不会增加均方误差。这表明，装袋(从训练数据抽取样本)通常会降低均方误差。

上面的讨论对于 0-1 损失下的分类不成立，因为偏倚和方差不具有可加性。在其处理中，装袋一个好的分类子可以使它变得更好，但装袋一个差的分类子可能使它变得更差。这里有一个简单例子，它使用随机化规则。假设对所有 x，$Y = 1$，并且分类子 $\hat{G}(x)$ 以 0.4 的概率预测 $Y = 1$(对所有的 x)，以 0.6 的概率预测 $Y = 0$(对所有的 x)。那么，$\hat{G}(x)$ 的误分类误差是 0.6，但装袋分类子的误分类率是 1.0。

注意，当我们对一个模型装袋时，模型中的任何简单结构都将失去。例如，一个装袋的树已不再是树。对于模型的解释，这显然是一个缺点。通常，像最近邻法那样较稳定的过程不太受装袋影响。遗憾的是，由于强调可解释性，受益于装袋方法最大的不稳定模型都不稳定，而可解释性在装袋过程中丢失了。

图 8.11 显示了一个例子，其中装袋不起作用。所显示的 100 个数据点具有两个特征和两个类，被灰色线性边界 $x_1 + x_2 = 1$ 分隔。我们选择单个轴向分裂作为分类子 $\hat{G}(x)$，选择沿 x_1 或 x_2 分裂，最大限度地降低训练样本的误分类误差。

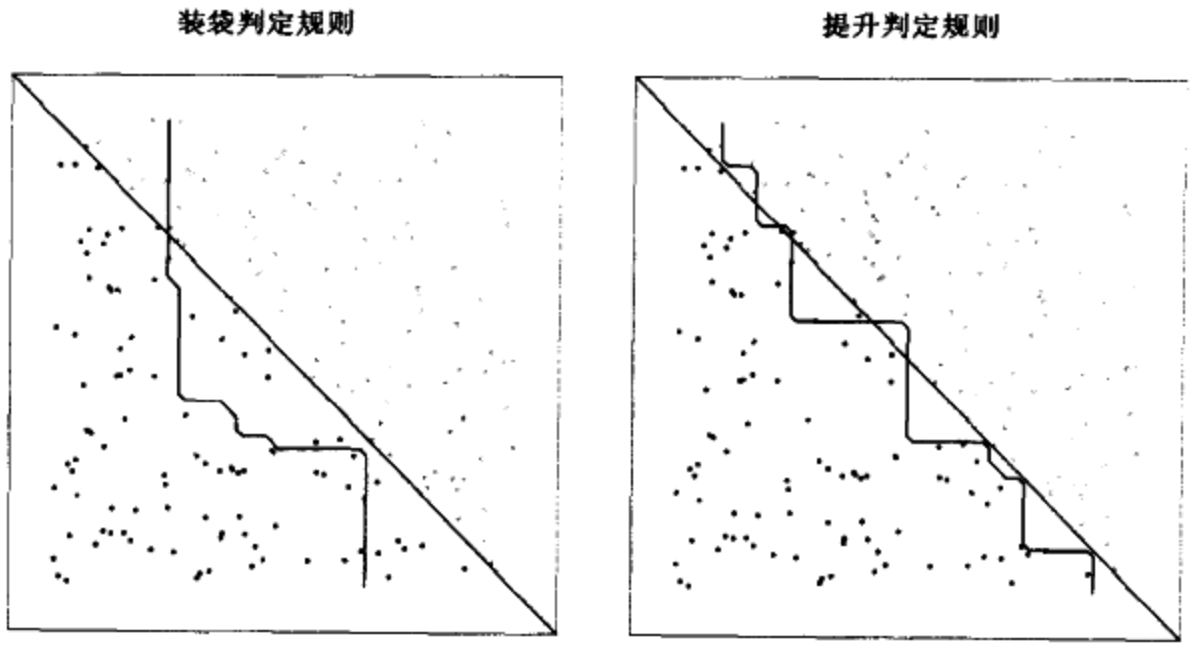


图 8.11 具有两个特征和两个类的数据，被一个线性边界分开。左图：对单个轴向分类子的判定规则装袋估计的判定边界。右图：由提升相同分类子的判定规则得到的估计判定边界。检验误差率分别为0.166和0.065。提升方法将在第10章介绍（见彩页）

左图的蓝色曲线显示在 $B = 50$ 个自助法样本上，对 0-1 判定规则装袋得到的判定边界。它不能很好地捕获真正的边界。从训练数据导出的单个分裂规则，在接近 0 处分裂（x_1 或 x_2 区间的中值），因此对远离中心的没有什么作用。这里，对概率而不是对分类求平均没有帮助。装袋从单个分裂规则来估计期望类概率，即在多个重复上求平均值。在上述模型的一个版本中，可以证明：对所有的 x_1 和 x_2，它的误差为 0.5（见习题 8.1）。注意，由装袋计算的期望类概率不能在任何一个单一重复上实现，其道理与一个妇女不能有 2.4 个孩子相同。在这种意义下，装袋一定程度上增大了各基本分类子的模型空间。然而，对于该例或需要更大地放大模型的其他一些例子，装袋都没有帮助。“提升”是做这件事的一种方法，将在第 10 章介绍。右图中的判定界是提升过程的结果，粗略地看它占据了对角边界。

8.8 模型平均和堆栈

在第 8.4 节，我们从非参数贝叶斯分析的角度，把估计子的自助法值看做对应参数的近似后验值。从这个角度看，装袋估计(8.51)是一个近似的后验贝叶斯均值。相反，训练样本估计 $\hat{f}(x)$对应于后验众数。由于后验均值（不是众数）极小化平方误差损失，因此装袋方法可能经常会减少均方误差也就不足为奇了。

这里，我们更加一般地讨论贝叶斯模型平均化。对于训练集 $\mathbf{Z}$，我们有候选模型的集合 $\mathcal{M}_m$，$m = 1, \cdots, M$。这些模型可能是具有不同参数值的同类模型（如线性回归中的子集），或是相同任务的不同模型（如神经网络和回归树）。

假设 ζ 是某感兴趣的量，例如，在某些固定特征值 x 上的预测 $f(x)$。ζ 的后验分布是：

$$\Pr(\zeta|\mathbf{Z}) = \sum_{m=1}^{M} \Pr(\zeta|\mathcal{M}_m, \mathbf{Z})\Pr(\mathcal{M}_m|\mathbf{Z}) \tag{8.53}$$

具有后验均值

$$\mathrm{E}(\zeta|\mathbf{Z}) = \sum_{m=1}^{M} \mathrm{E}(\zeta|\mathcal{M}_m, \mathbf{Z})\Pr(\mathcal{M}_m|\mathbf{Z}) \tag{8.54}$$

该贝叶斯预测是每个预测的加权平均，其权值与每个模型的后验概率成比例。

该公式导致一些不同的模型平均化策略。委员会方法(committee method)取每个模型预测的简单平均，本质上是对每个模型取相等的概率。更进一步，第 7.7 节的推演表明 BIC 准则可以用于估计模型的后验概率。当不同的模型源自相同的参数模型，具有不同的参数值时，可以使用 BIC 准则。BIC 将权值赋给每个模型，权值大小取决于模型的拟合程度和使用参数的多少。我们也可以充分地实现贝叶斯方法。如果每一个模型 $\mathcal{M}_m$ 有参数 θ_m，我们记：

$$\begin{aligned} \Pr(\mathcal{M}_m|\mathbf{Z}) &\propto \Pr(\mathcal{M}_m)\cdot\Pr(\mathbf{Z}|\mathcal{M}_m) \\ &\propto \Pr(\mathcal{M}_m)\cdot\int \Pr(\mathbf{Z}|\theta_m, \mathcal{M}_m)\Pr(\theta_m|\mathcal{M}_m)\mathrm{d}\theta_m \end{aligned} \tag{8.55}$$

原则上，我们可以指定先验概率 $\Pr(\theta_m \mid \mathcal{M}_m)$，并通过式(8.55)来计算后验概率，用做模型平均化的权值。然而，与更简单的 BIC 逼近相比，没有实际证据能够表明所有这些努力是值得的。

我们怎样从频率论者的角度逼近模型平均呢？给定预测 $\hat{f}_1(x), \hat{f}_2(x), \cdots, \hat{f}_M(x)$，在平方误差损失下，可以寻找权值 $w = (w_1, w_2, \cdots, w_M)$，使得：

$$\hat{w} = \underset{w}{\operatorname{argmin}}\, \mathrm{E}_{\mathcal{P}}\left[Y - \sum_{m=1}^{M} w_m \hat{f}_m(x)\right]^2 \tag{8.56}$$

这里，输入值 x 是固定的，而数据集 $\mathbf{Z}$(和目标 Y)中的 N 个观测分布在 $\mathcal{P}$ 上。解是 $\hat{F}(x)^T \equiv [\hat{f}_1(x), \hat{f}_2(x), \cdots, \hat{f}_M(x)]$ 上 Y 的总体线性回归：

$$\hat{w} = \mathrm{E}_{\mathcal{P}}[\hat{F}(x)\hat{F}(x)^T]^{-1}\mathrm{E}_{\mathcal{P}}[\hat{F}(x)Y] \tag{8.57}$$

现在，完全回归比任何单一模型都有较小的误差

$$\mathrm{E}_{\mathcal{P}}\left[Y - \sum_{m=1}^{M} \hat{w}_m \hat{f}_m(x)\right]^2 \leqslant \mathrm{E}_{\mathcal{P}}\left[Y - \hat{f}_m(x)\right]^2 \ \forall m \tag{8.58}$$

因此，在总体级，组合模型绝对不会使事情变得更糟。

当然，总体线性回归(8.57)不可用，因而代之以训练集上的线性回归是自然的。但是，存在一些简单的例子，该方法不能成功。例如，设有 M 个输入，如果 $\hat{f}_m(x)(m = 1, 2, \cdots, M)$ 表示 m 个输入的最佳子集的预测，则线性回归将全部权值置于极大模型上，即 $\hat{w}_M = 1, \hat{w}_m = 0, m < M$。问题是我们没有通过考虑它们的复杂性(输入的数目 m)，将每个模型置于相同的立足点。

堆栈泛化(stacked generalization)，或者堆栈(stacking)是解决该问题的一种方法。令 $\hat{f}_m^{-i}(x)$ 是 x 上的预测，使用模型 m，应用于删除第 i 个训练观测后的数据集。权的堆栈估计由 y_i 在 $\hat{f}_m^{-i}(x_i)(m = 1, 2, \cdots, M)$ 上的最小二乘方线性回归得到。更详细地，堆栈权由下式给出：

$$\hat{w}^{st} = \underset{w}{\operatorname{argmin}} \sum_{i=1}^{N} \left[y_i - \sum_{m=1}^{M} w_m \hat{f}_m^{-i}(x_i) \right]^2 \tag{8.59}$$

最终的预测是 $\sum_m \hat{w}_m^{st} \hat{f}_m(x)$。通过使用交叉验证预测 $\hat{f}_m^{-i}(x)$，堆栈避免了将不合理的高权值赋予具有高复杂度的模型。更好的结果可以通过限定权值非负并且和为 1 得到。如果我们像在式(8.54)中一样，将权值解释为后验模型概率，似乎是一个合理的限制，而且它将导致易于处理的二次规划问题。

在堆栈和通过留一交叉验证(见第 7.10 节)进行的模型选择之间有着密切的联系。如果我们限制式(8.59)的极小化为有一个单位权其余为 0 的权向量 w，将导致选择具有最小留一交叉验证误差的模型 $\hat{m}$。堆栈不是选择单个模型，而是按估计最优权值组合它们。这通常将导致更好的预测，但可解释性不如从 M 个模型中只选取一个好。

实际上，堆栈思想比上面介绍的要更一般些。可以使用任意的学习方法，而不仅仅是线性回归，像在式(8.59)中一样去组合模型；权也可以依赖输入位置 x。用这种办法，学习方法一个被"堆"在另一个之上，以改进预测性能。

8.9 随机搜索：冲击

本章介绍的最后一种方法不涉及模型平均或组合，而是一种找出单个较好模型的技术。冲击(bumping)使用自助法抽样随机地考察模型空间。对于拟合方法找出许多局部极小值的问题，冲击可以帮助这种方法避免陷入较差的解。

与装袋一样，抽取自助法样本并且用一个模型拟合每个样本。但是，我们不是对预测求平均，而是选择由自助法样本估计的模型，它最好地拟合训练数据。更详细地说，抽取自助法样本 $\mathbf{Z}^{*1}, \cdots, \mathbf{Z}^{*B}$，并用我们的模型拟合其中的每一个，产生输入点 x 上的预测 $\hat{f}^{*b}(x)$，$b = 1, 2, \cdots, B$。然后，我们选择这样的模型，在原始训练集上求平均，它产生最小的预测误差。例如，对于平方误差，选择从自助法样本 $\hat{b}$ 得到的模型，其中：

$$\hat{b} = \arg\min_b \sum_{i=1}^{N} [y_i - \hat{f}^{*b}(x_i)]^2 \tag{8.60}$$

对应的模型预测是 $\hat{f}^{*\hat{b}}(x)$。按惯例，我们也在自助法样本集中包含原始训练样本，以便在它具有最低训练误差时，可以自由地选取原始模型。

通过扰动数据，冲击方法试图将拟合过程移向模型空间的较好区域。例如，如果少量数据点导致该过程找到较差的解，则任何省略这些数据点的自助法样本都将产生较好的解。

另一个例子，考虑图 8.12 中的分类数据，众所周知的异或(XOR，exclusive or)问题。这里有两个类(绿色的和红色的)和两个输入特征，其中特征表现出纯交互效应。通过在 $x_1 = 0$ 处分裂数据，而后在 $x_2 = 0$ 处分裂每个结果层(或者反过来)，基于树的分类法能够实现正确的判别。但是，贪心而短见的 CART 算法(见第 9.2 节)试图在两个特征上发现最佳分裂，然后分裂结果层。由于数据的平衡特性，在 x_1 或 x_2 上的所有初始分裂看上去都是无用的，并且该过程在顶层本质上产生了一个随机分裂。关于这些数据所发现的实际分裂显示在图 8.12 的左图中，通过从数据中自助法抽样，冲击打破了这些类中的平衡，使用合理的自助法样本数量(这

里是 20)，它将偶然地至少产生一棵树，其初始分裂靠近 $x_1 = 0$ 或 $x_2 = 0$。仅用 20 个自助法样本，冲击发现了接近于最佳的分裂，显示在图 8.12 的右图中。如果我们增加一些独立于类标号的噪声特征，这种贪心的树增长算法的不足将更加严重。树增长算法就无法将 x_1 或 x_2 与其他值区分开，并且产生严重的丢失。

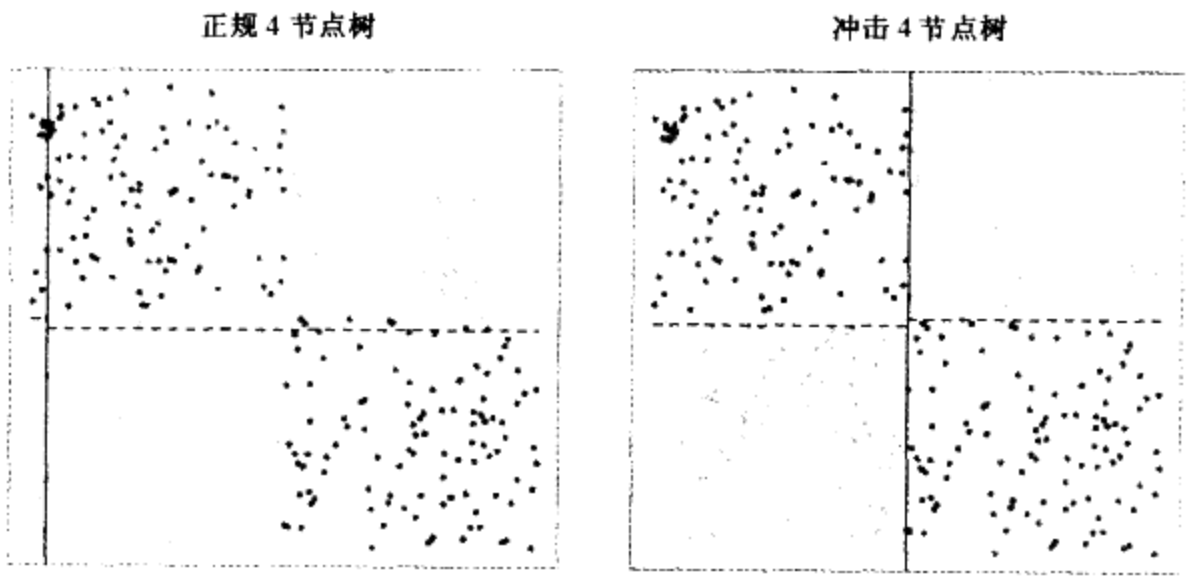


图 8.12　具有两个特征和两个类(绿色和红色)的数据，显示出纯交互效应。左图显示了由一个标准、贪心的树增长分裂方法的三次分裂所发现的划分。靠近左边界的蓝色垂线是最初的分裂，虚线是两个随后得到的分裂。算法不知道好的初始分裂在哪里，做出了一个很差的选择。右图则显示了冲击树增长算法20次所发现的接近最佳的分裂(见彩页)

由于冲击比较训练数据上不同的模型，所以必须保证模型具有大致相同的复杂性。对于树，这意味着在每个自助法样本上用相同数目的端节点来增长树。可能是由于缺少光滑性，在优化拟合准则有困难的问题中，冲击可能会有些帮助。其诀窍是，在自助法样本上优化一个不同的、更加方便的准则，然后选择在训练样本上对所希望准则产生最好结果的模型。

文献注释

有许多经典的统计推理书籍：Cox 和 Hinkley(1974)和 Silvey(1975)给出非技术性陈述。自助法源自于 Efron(1979)，并在 Efron 和 Tibshirani(1993)及 Hall(1992)中有更全面的介绍。一本较好的关于贝叶斯推理的现代书籍是 Gelman 等(1995)。贝叶斯方法在神经网络方面应用的明晰说明由 Neal(1996)给出。Gibbs 抽样统计应用源于 Geman 和 Geman(1984)、Gelfand 和 Smith(1990)，相关的工作由 Tanner 和 Wong(1987)给出。马尔科夫链蒙特卡罗方法，包括Gibbs抽样和Metropolis-Hastings算法，在Spiegelhalter等(1976)中讨论。EM 算法出自Dempster等(1997)；正如该文讨论所澄清的，存在一些非常相关的早期工作。作为罚完全数据对数似然的联合极大化方案的 EM 观点由 Neal 和Hinton(1998)予以阐明；他们相信Csiszar和Tusnády(1984)及Hathaway(1986)较早注意到这种联系。装袋由Breiman(1996a)提出。堆栈出自于Wolpert(1992)；Breiman(1996b)包含了易于被统计学家理解的讨论。Leblanc和Tibshirani(1996)介绍了基于自助法的堆栈的变形。在贝叶斯框架下求模型的平均最近被Madigan和Raftery(1994)提倡。冲击是由Tibshirani和 Knight(1999)提出的。

习题

8.1 考虑一个单位正方形，并通过值分别在每个轴的 1/3 和 2/3 处的两条直线将它分割成 9 个大小相等的小正方形。线 $x_1 + x_2 = 1$ 之下是类 1，其余的是类 2。

考虑单个分裂分类子，它在四条线（两个在 x_1 上的 1/3 处和 2/3 处，对 x_2 类似）之一处产生分裂，全部四个分裂具有相同的概率。

(a) 分析对判定规则和概率估计装袋的效果，并证明前者误差率为 0.5，而后者具有贝叶斯零误差率。

(b) 计算单个分裂概率估计及其装袋版本的偏倚和方差。由此判定装袋的正效应是否可以减少偏倚和方差。

8.2 令 $r(y)$ 和 $q(y)$ 是概率密度函数。詹生（Jensen）不等式指出：对于随机变量 X 和凸函数 $\phi(x)$，有 $\mathrm{E}[\phi(X)] \geqslant \phi[\mathrm{E}(X)]$。使用詹生不等式证明：当 $r(y) = q(y)$ 时，

$$\mathrm{E}_q \log[r(Y)/q(Y)] \tag{8.61}$$

作为 $r(y)$ 的函数被极大化。由此证明 $R(\theta, \theta) \geqslant R(\theta', \theta)$，如式(8.46)所述。

8.3 在使得 $\tilde{P}(\mathbf{Z}^m) \geqslant 0$ 并且 $\sum_{\mathbf{Z}^m} \tilde{P}(\mathbf{Z}^m) = 1$ 的分布 $\tilde{P}(\mathbf{Z}^m)$ 上，考虑对数似然(8.48)的极大化。使用拉格朗日乘子证明解是条件分布 $\tilde{P}(\mathbf{Z}^m) = \Pr(\mathbf{Z}^m \mid \mathbf{Z}, \theta')$，同式(8.49)中一样。

8.4 使用关系式

$$\Pr(A) = \int \Pr(A|B) d(\Pr(B))$$

证明估计(8.50)。

8.5 考虑第 8.7 节的装袋方法。令我们的估计 $\hat{f}(x)$ 是第 8.2.1 节中的 B 样条光滑子 $\hat{\mu}(x)$。考虑式(8.6)的参数自助法，将其应用于该估计子。证明：如果对 $\hat{f}(x)$ 装袋，使用参数自助法产生自助法样本，则当 $B \to \infty$ 时，装袋估计 $\hat{f}_{\mathrm{bag}}(x)$ 收敛于原始估计 $\hat{f}(x)$。

8.6 提出图 10.4 中每个损失函数到多个类（大于 2）的泛化，并设计一个合适的图比较它们。

8.7 考虑图 5.6 的骨质密度数据。

(a) 用三次光滑样条函数拟合脊椎 BMD 的相关变化，作为年龄的函数。使用交叉验证估计最优光滑量。构造基本函数的逐点 90% 置信带。

(b) 通过式(8.28)，计算真实函数的后验均值和协方差，并与(a)中获得的后验带比较。

(c) 像在图 8.2 的左下图一样，计算拟合曲线的 100 次自助法复制。将其结果与(a)和(b)得到的结果进行比较。

第 9 章　加法模型、树和相关方法

本章，我们开始讨论一些有指导学习的特定方法。这些技术都对未知的回归函数假定一种（不同的）结构化形式，并借此巧妙地解决了维灾难。当然，它们也为不能详细说明模型而付出了必要的代价，因此在每一种情况下都必须做出权衡。现在继续进行在第 3 章到第 6 章终止的讨论。我们介绍 5 种相关的技术：广义加法模型、树、多元自适应回归样条、忍耐规则归纳方法和专家分层混合方法。

9.1　广义加法模型

在许多数据分析中，回归方法都起着重要的作用，它提供预测、分类规则和数据分析工具，以理解不同输入的重要性。

尽管传统的线性模型极其简单，但常常不适应下列情况：在现实中，作用常常不是线性的。在前面的章节中，我们讨论了使用预定义的基函数实现非线性作用的技术。本节将介绍更加自动、灵活的统计学方法，可以用于定义和刻画非线性回归的效果。这些方法称做“广义加法模型”。

在回归的框架下，广义加法模型具有如下形式：

$$\mathrm{E}(Y|X_1, X_2, \cdots, X_p) = \alpha + f_1(X_1) + f_2(X_2) + \cdots + f_p(X_p) \tag{9.1}$$

按照惯例，X_1，X_2，$\cdots$，X_p 表示预测子，Y 是输出；诸 f_j 是未指定的光滑（“非参数的”）函数。如果想用基函数的展开式（如第 5 章中的）对每个函数建模，则结果模型可以用简单的最小二乘方拟合。这里的方法不同：我们使用散点图光滑子（例如，三次非参数光滑样条或核光滑子）拟合每个函数，并提供一个同时估计所有 p 个函数的算法（见第 9.1.1 节）。

对于 2-类分类问题，回顾一下在第 4.4 节中讨论的二元数据的逻辑斯缔回归模型。通过线性回归模型和分对数连接（logit link）函数：

$$\log\left(\frac{\mu(X)}{1-\mu(X)}\right) = \alpha + \beta_1 X_1 + \cdots + \beta_p X_p \tag{9.2}$$

将二元响应 $\mu(X) = \Pr(Y = 1 | X)$ 的均值与预测子联系起来。

加法（additive）逻辑斯缔回归模型用更一般的函数形式替换每一个线性项：

$$\log\left(\frac{\mu(X)}{1-\mu(X)}\right) = \alpha + f_1(X_1) + \cdots + f_p(X_p) \tag{9.3}$$

其中，f_j 仍然是未指定的光滑函数。虽然函数 f_j 的非参数形式使模型更加灵活，但它仍然保持可加性，并且允许我们用与前面完全相同的方式解释模型。加法逻辑斯缔回归模型是广义加法模型的一个例子。一般地，响应 Y 的条件均值 $\mu(X)$ 通过连接函数 g：

$$g[\mu(X)] = \alpha + f_1(X_1) + \cdots + f_p(X_p) \tag{9.4}$$

和预测子的加法函数联系起来。

连接函数的典型例子如下：

- $g(\mu) = \mu$是恒等连接，用于高斯响应数据的线性模型和加法模型。
- $g(\mu) = \text{logit}(\mu)$或 $g(\mu) = \text{probit}(\mu)$，概率单位连接函数，用于对二项式概率建模。概率单位函数是逆高斯积累分布函数：$\text{probit}(\mu) = \Phi^{-1}(\mu)$。
- $g(\mu) = \log(\mu)$用于泊松计数数据的对数线性模型或对数加法模型。

这些函数都源于指数族抽样模型，该族还包括 γ 分布和负二项式分布。这些族产生了著名的广义线性模型类，它们都以相同的方式扩展为广义加法模型。

使用一个以散点图光滑子为基本构件的算法，可以以灵活的方式估计函数 f_j。估计函数 $\hat{f}_j$ 能够揭示 X_j 的作用中可能的非线性性。并非所有的 f_j 函数都必须是非线性的。我们可以很容易地把线性形式和其他参数形式与非线性项混合；当某些输入是定性的变量(因子)时，这是必要的。非线性项也不限于主要效应；在两个或多个变量，或者在因子 X_k 的每一级 X_j 上的分离曲线中，都可以有非线性分量。这样，下面每一条都符合要求：

- $g(\mu) = X^T\beta + \alpha_k + f(Z)$ ——半参数(semiparametric)模型，其中 X 是一个被线性建模的预测向量，α_k 是第 k 级定性输入 V 的效应，而预测子 Z 的效应被非参数地建模。
- $g(\mu) = f(X) + g_k(Z)$——k 仍指明定性输入 V 的级，并由此为 V 和 Z 的效应建立一个交互项 $g(V, Z) = g_k(Z)$。
- $g(\mu) = f(X) + g(Z, W)$，其中 g 是两个特征上的非参数函数。

加法模型可以在多种情况下取代线性模型，例如时间序列的加法分解：

$$Y_t = S_t + T_t + \varepsilon_t \tag{9.5}$$

其中 S_t 是季节分量，T_t 是趋势，而 ε 是误差项。

9.1.1 拟合加法模型

本节，我们介绍拟合加法模型的模算法及其泛化。其构件是以一种灵活的方式拟合非线性效应的散点图光滑子。为了使讨论更加具体，使用第 5 章介绍的三次光滑样条作为我们的散点图光滑子。

加法模型有如下形式：

$$Y = \alpha + \sum_{j=1}^{p} f_j(X_j) + \varepsilon \tag{9.6}$$

其中，误差项 ε 均值为 0。给定观测 x_i, y_i，可以对该问题指定一个类似于第 5.4 节的罚平方和标准(5.9)的标准，

$$\text{PRSS}(\alpha, f_1, f_2, \cdots, f_p) = \sum_{i=1}^{N}\left\{y_i - \alpha - \sum_{j=1}^{p} f_j(x_{ij})\right\}^2 + \sum_{j=1}^{p}\lambda_j \int f_j''(t_j)^2 \mathrm{d}t_j \tag{9.7}$$

其中 $\lambda_j \geqslant 0$ 是调整参数。可以证明式(9.7)的极小化是一个加法三次样条模型；每个函数 f_j 是分量 X_j 上的三次样条，纽结在 $x_{ij}, i = 1, \cdots, N$ 的每个惟一值上。然而，如果不对模型做进一步

限制，解就不惟一。常量α不是确定的，因为我们可以对每个函数 f_j 加上或减去任意的常量，并相应地调整α。标准的做法是假设对于任意 j，$\sum_1^N f_j(x_{ij}) = 0$——函数在数据上的平均值为 0。容易看出，在这种情况下，$\hat{\alpha} = \text{ave}(y_i)$。如果除这个限制外，输入矩阵（第 ij 个元素为 x_{ij}）是非奇异的，则式(9.7)是一个严格的凸准则，并且极小值惟一。如果矩阵是奇异的，则分量 f_j 的线性部分不能惟一地确定（而非线性部分能惟一地确定）（Buja 等人，1989）。

进一步，还存在着发现解的简单迭代过程。置$\hat{\alpha} = \text{ave}(y_i)$，而且它一直不变。我们将三次光滑样条 S_k 应用于目标 $\{y_i - \hat{\alpha} - \sum_{j\neq k}\hat{f}_j(x_{ij})\}_1^N$，作为 x_{ik}的函数，得到一个新估计 $\hat{f}_k$。依次对每个预测子计算新估计，在计算 $y_i - \hat{\alpha} - \sum_{j\neq k}\hat{f}_j(x_{ij})$ 时，使用其他函数 $\hat{f}_j$ 的当前估计。继续该过程，直到估计 $\hat{f}_j$ 稳定为止。该过程在算法 9.1 中给出，称为“反向拟合”，其结果拟合类似于线性模型的多元回归。

原则上，算法 9.1 步骤 2 中的第二步不是必需的，因为光滑样条拟合一个均值为 0 的响应具有均值 0（见习题 9.1）。实践中，机器舍入会引起滑动，建议做一些调整。

算法 9.1　加法模型的反拟合算法

1. 初始化：$\hat{\alpha} = \frac{1}{N}\sum_1^N y_i, \hat{f}_j \equiv 0, \forall\, i,j$

2. 循环：$j = 1, 2, \cdots, p, \cdots, 1, 2, \cdots, p, \cdots,$

$$\hat{f}_j \leftarrow S_j\left[\{y_i - \hat{\alpha} - \sum_{k\neq j}\hat{f}_k(x_{ik})\}_1^N\right]$$

$$\hat{f}_j \leftarrow \hat{f}_j - \frac{1}{N}\sum_{i=1}^N \hat{f}_j(x_{ij})$$

直到函数 $\hat{f}_j$ 的变化小于预先指定的阈值

同样的算法能够以完全相同的方式用于其他拟合方法，只要指定适当的光滑算子 S_j：

- 其他一元回归光滑子，如局部多项式回归和核方法；
- 产生多项式拟合、分段常量拟合、参数样条拟合、级数和傅里叶拟合的线性回归算子；
- 更复杂的算子，如二阶或高阶交的面光滑子，或有周期效应的周期性光滑子。

如果我们只在训练点考虑光滑子 S_j 的操作，则可以用一个 $N \times N$ 的算子矩阵 $\mathbf{S}_j$ 表示它（见第 5.4.1 节）。这样，第 j 项的自由度（近似地）就由 $\text{df}_j = \text{trace}[\mathbf{S}_j] - 1$ 来计算，与第 5 章和第 6 章讨论过的光滑子的自由度类似。

对于一大类光滑算子 $\mathbf{S}_j$，反向拟合与求解确定线性方程组的高斯 - 塞德尔算法等价。详细内容见习题 9.2。

对于逻辑斯缔回归模型和其他广义加法模型，适当的准则是罚对数似然。为了对它极大化，将反向拟合过程与似然极大法联合应用。通常在广义线性模型中极大化对数似然的 Newton-Raphson例程可以改写成 IRLS（迭代加权最小二乘方）算法。这涉及反复在协方差上拟合工作响应变量的加权线性回归；每个回归产生一个新的参数估计的值，它依次产生新的工作响应和权值，且该过程是迭代的（见第 4.4.1 节）。在广义加法模型中，可以简单地用一个加权的反向拟合算法替换加权线性回归。下面，我们将对逻辑斯缔回归更详细地描述该算法，而更

一般的描述在 Hastie 和 Tibshirani(1990)的第 6 章中。

9.1.2 例:加法逻辑斯缔回归

或许,在医学研究中最广泛使用的模型是二元数据的逻辑斯缔模型。在这种模型中,输出 Y 可以用 0 或 1 编码,其中 1 表示事件发生(如,死亡或病变),而 0 表示事件没有发生。我们希望建立模型 $\Pr(Y = 1|X)$,它是给出病情预报因子 $X = (X_1, \cdots, X_p)$ 值的事件概率。通常,目标是理解病情预报因子的作用,而不是对新的个体分类。逻辑斯缔模型也用于风险筛选;在那里,我们是对估计类概率感兴趣。除了医学中的应用外,信用风险筛选也是很常见的应用。

广义加法逻辑斯缔模型有如下形式:

$$\log \frac{\Pr(Y=1|X)}{\Pr(Y=0|X)} = \alpha + f_1(X_1) + \cdots + f_p(X_p) \tag{9.8}$$

函数 $f_1, f_2, \cdots, f_p$ 通过Newton-Raphson过程中的反向拟合算法来估计,由算法 9.2 给出。

算法 9.2 加法逻辑斯缔回归模型的局部评分算法

1. 计算初始值:$\hat{\alpha} = \log[\bar{y}/(1-\bar{y})]$,其中 $\bar{y} = \text{ave}(y_i)$ 是类 1 的样本比例,并对任意 j,置 $\hat{f}_j \equiv 0$
2. 定义 $\hat{\eta}_i = \hat{\alpha} + \sum_j \hat{f}_j(x_{ij})$ 并且 $\hat{p}_i = 1/[1+\exp(-\hat{\eta}_i)]$

   迭代:

   (a)构造工作目标变量

$$z_i = \hat{\eta}_i + \frac{(y_i - \hat{p}_i)}{\hat{p}_i(1-\hat{p}_i)}$$

   (b)构造权值 $w_i = \hat{p}_i(1-\hat{p}_i)$

   (c)使用加权反向拟合算法,用加法模型拟合具有权值 w_i 的目标 z_i。这给出了新的估计 $\hat{\alpha}, \hat{f}_j, \forall j$
3. 继续步骤 2,直到函数变化低于预先指定的阈值

算法 9.2 步骤 2 中的加法模型拟合需要一个加权的散点图光滑子。大多数光滑过程都可以接受观测权值(见习题 5.12);详细内容参考 Hastie 和Tibshirani(1990)的第 3 章。

使用第 4.4 节的多元分对数公式,可以进一步拓广加法逻辑斯缔回归模型,以处理多个类。尽管该公式是式(9.8)的直接扩展,但拟合这种模型的算法将更加复杂。详细内容参考 Yee 和 Wild(1996),VGAM 软件现在可从以下网站得到:

`http://www.stat.auckland.ac.nz/~yee`。

例:预报垃圾邮件

我们把广义加法模型应用于第 1 章介绍的垃圾邮件数据。这些数据包括来自 4601 个邮件中的信息,用于研究筛选垃圾邮件。该数据公开发布在网站ftp.ics.uci.edu上,由位于加州 Palo Alto的Hewlett-Packard 实验室的George Forman提供。

响应变量是二元的,具有值 email 或 spam;有 57 个预测子,如下所述:

- 48 个定量的预测子——邮件中与给定单词相匹配的单词的百分比。例子包括business, address, internet, free 和george,其主旨是这些可以针对用户定制。
- 6 个定量的预测子——邮件中与给定字符相匹配的字符的百分比。这些字符是 ch;,ch(,

ch[，ch!，ch $和 ch #。

- 不间断的大写字母序列的平均长度：CAPAVE。
- 不间断的大写字母序列的最大长度：CAPMAX。
- 不间断的大写字母序列的长度之和：CAPTOT。

我们用 1 对 spam 编码，用 0 对 email 编码。随机抽取一个容量为 1536 的检验集，其余 3065 个观测留在训练集中。可以使用每个预测子具有四个自由度的三次光滑样条拟合广义加法模型。即对每一个预测 X_j，选定光滑样条参数 λ_j，使得 $\text{trace}[\mathbf{S}_j(\lambda_j)]-1=4$，其中 $\mathbf{S}_j(\lambda)$是使用观测值 $x_{ij}(i=1,\cdots,N)$构造的光滑样条算子矩阵。这是在如此一个复杂的模型中说明光滑总量的方便方式。

检验误差率显示在表 9.1 中，总体检验误差率是 5.3%。通过对比，线性逻辑斯缔回归的检验误差率是 7.6%。表 9.2 显示预测子，它们在加法模型中是高显著的。

表 9.1　用于拟合垃圾邮件训练数据的加法逻辑斯缔回归模型的检验数据混合(confusion)矩阵。总体检验误差率是 5.3%

真实类	预测类	
	email(0)	spam(1)
email(0)	58.5%	2.5%
spam(1)	2.7%	36.2%

表 9.2　加法模型拟合垃圾邮件训练数据的显著预测子。系数表示 $\hat{f}_j$ 的线性部分，依据的是它们的标准误差和 Z-得分。非线性 P 值是 $\hat{f}_j$ 的非线性检验

名称	编号	df	系数	标准误差	Z-得分	非线性 P 值
			正效应			
our	6	3.9	0.566	0.114	4.970	0.052
over	7	3.9	0.244	0.195	1.249	0.004
remove	8	4.0	0.949	0.183	5.201	0.093
internet	9	4.0	0.524	0.176	2.974	0.028
free	17	3.9	0.507	0.127	4.010	0.065
business	18	3.8	0.779	0.186	4.179	0.194
hpl	27	3.8	0.045	0.250	0.181	0.002
ch!	53	4.0	0.674	0.128	5.283	0.164
ch $	54	3.9	1.419	0.280	5.062	0.354
CAPMAX	57	3.8	0.247	0.228	1.080	0.000
CAPTOT	58	4.0	0.755	0.165	4.566	0.063
			负效应			
hp	26	3.9	−1.404	0.224	−6.262	0.140
george	28	3.7	−5.003	0.744	−6.722	0.045
1999	38	3.8	−0.672	0.191	−3.512	0.011
re	46	3.9	−0.620	0.133	−4.649	0.597
edu	47	4.0	−1.183	0.209	−5.647	0.000

为了解释方便，在表 9.2 中，每个变量的分布被分解成线性部分和非线性部分。上部的预测与垃圾邮件正相关，而下部的预测与之负相关。线性部分是预测上的拟合曲线的加权最小二乘方线性拟合，而非线性部分是残差。估计函数的线性部分按系数、标准误差和 Z-得分汇总；Z-得分是系数除以标准误差，且如果它超过标准正交分布的合适的百分位，则被认为是显著的。标有非线性 P 值的列是估计函数的非线性检验。然而，需要注意的是，每一个预测子的效应完全根据其他预测子的全部效应来调整，而不只是对其线性部分进行调整。显示在表中的预测子至少在水平 $p=0.01$（双测）上的一个检验（线性或非线性的）中表明是显著的。

图 9.1 显示了出现在表 9.2 中的显著预测子的估计函数。许多非线性效应似乎解释了在 0 上的不连续性。例如，随 george 的频率从 0 增加，垃圾邮件的概率明显地下降，但之后变化不大。这表明我们可以用一个从 0 开始的指示器变量替换每个频度预测子，并可以借助于线性逻辑斯缔模型。这将导致 7.4% 的检验误差率；包括频率的线性作用时，误差率降到 6.6%。看来，加法模型中的非线性性具有附加的预测能力。

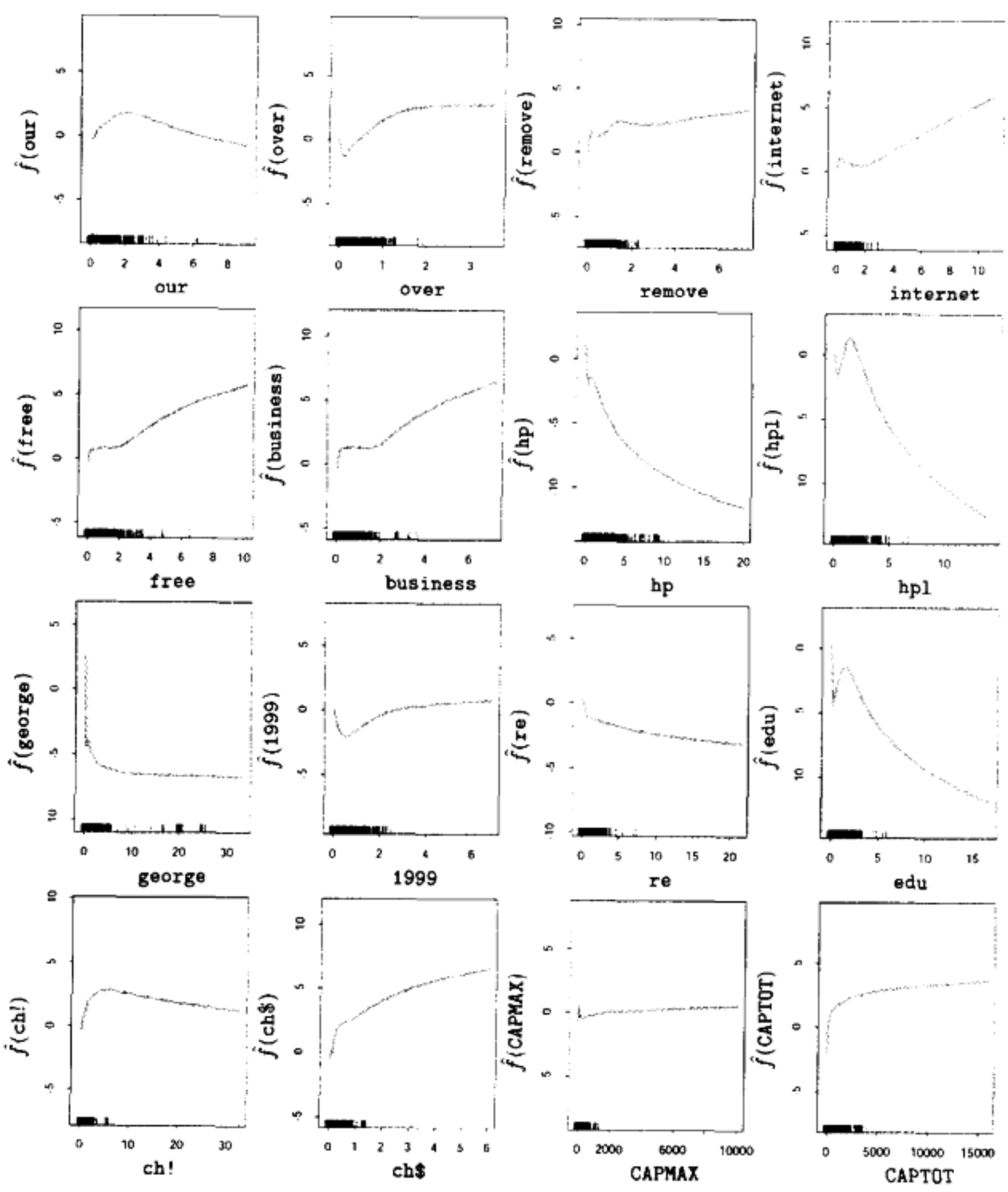


图 9.1 垃圾邮件分析：显著预测子的估计函数。沿着每个框底部的底线图指出相应预测的观测值。对许多预测子来说，非线性特征在0上表现为不连续

把一个真正的 email 处理成 spam 是比较严重的事情，因为这意味一个好的邮件将被过滤掉而永远无法到达用户手里。可以通过改变损失（见第 2.4 节）来改变类误差之间的平衡。如果我们指派将实际类 0 预测为类 1 的损失为 L_{01}，将实际类 1 预测为类 0 的损失为 L_{10}，则当类 1 的概率大于 $L_{01}/(L_{01}+L_{10})$时，估计贝叶斯规则预测类 1。例如，如果我们取 $L_{01}=10$，$L_{10}=1$，则（实际）类 0 和类 1 的误差率就分别变为 0.8% 和 8.7%。

更进一步，通过对类 0 的观测使用权 L_{01}，对类 1 的观测使用权 L_{10}，我们可以让模型更好地拟合类 0 的数据。然后，与上面一样，可以使用估计贝叶斯规则进行预测。这样，（实际）类 0 和类 1 的误差率分别为 1.2% 和 8.0%。下面，我们将在基于树的模型背景下进一步讨论不等损失问题。

拟合一个加法模型之后，我们应该检查某些相互效应是否可以显著地改善这种拟合。这种检查可以通过插入部分或全部显著输入的乘积“手工地”完成，也可以通过 MARS 过程（见第 9.4 节）自动地完成。

该例子以自动方式使用加法模型。作为一种数据分析工具，加法模型常以更交互的方式来使用，通过增加和减少项来确定它们的效应。通过调整 df_j 中的光滑总量，我们可以连续地在线性模型（$df_j=1$）和部分线性模型之间选择，其中有些项可以更灵活地建模。详细内容参见 Hastie 和 Tibshirani（1990）。

9.1.3　小结

加法模型提供了对线性模型有用的扩展，使它们更加灵活，但仍然保持着它们大部分的可解释性。线性模型建模和推理的常用工具也能用于加法模型（见表 9.2 中的例子）。拟合这些模型的反向拟合过程很简单而且有标准组件，且允许我们为每个输入变量选择合适的拟合方法。结果，它们被广泛地应用于统计学领域。

然而，对于大规模数据挖掘应用，加法模型可能有局限性。当数据非常大时，用反向拟合算法拟合所有预测子可能不现实或者说不理想。BRUTO 过程（Hastie 和 Tibshirani，1990，第 9 章）将反向拟合与输入选择结合在一起，但它不是为解决大规模数据挖掘问题而设计的。而对于这些问题，诸如提升（见第 10 章）等方法将更加有效，而且还允许模型中包括交互性。

9.2　基于树的方法

9.2.1　背景

基于树的方法把特征空间划分成一系列的矩形区域，然后在每一个区域中拟合一个简单的模型（如常量）。它们概念上简单但很有效。首先介绍一个流行的基于树的回归和分类的方法 CART，然后将它与一个重要的竞争者 C4.5 比较。

让我们考虑一个具有连续响应 Y 和输入 X_1，X_2 的回归问题，其中 X_1 和 X_2 在单位区间上取值。图 9.2 中左上图显示了与坐标轴平行的直线对特征空间的划分。在每个划分元素，可以用一个不同的常量对 Y 建模。然而，这里有一个问题：尽管每一个划分直线具有像 $X_1=c$ 这样的简单描述，但一些结果区域却很难描述。

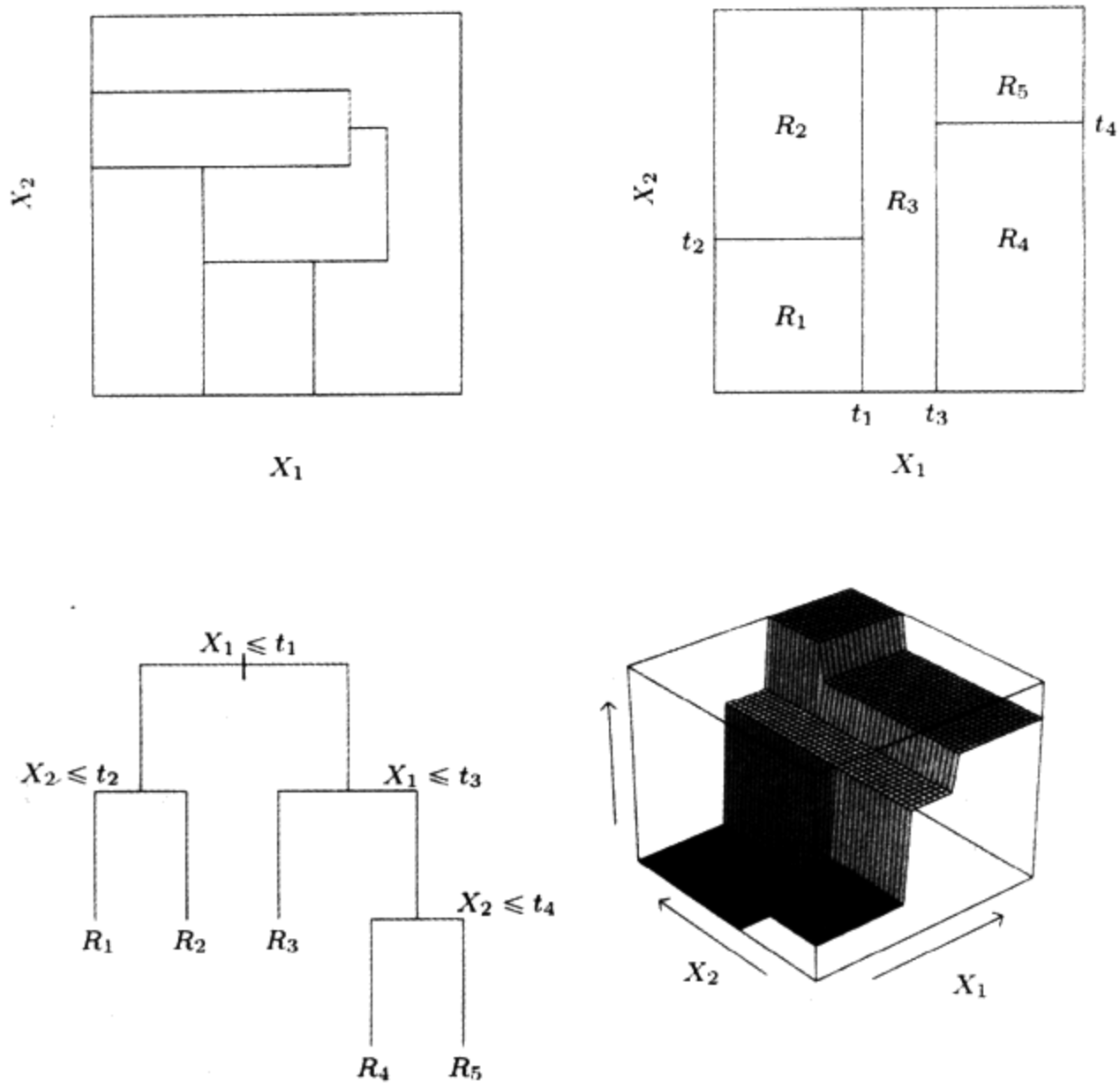


图 9.2 划分和 CART。右上图显示的是递归二叉分裂(如 CART 使用的)对一个二维特征空间的划分,作用于某伪数据上。左上图显示的是一般的划分,它无法从递归二叉分裂得到。左下图显示的树对应于右上图的划分,而预测面的透视图显示在右下图

为了简化问题,我们只关注图 9.2 的右上图所示的递归的二元划分。首先,把空间划分为两个区域,用每一个区域中 Y 的均值对响应建模。我们通过选择变量和分裂点来实现最佳拟合。然后,把这些区域中的一个或者两个进一步分裂成两个区域,这个过程继续下去,直到满足一些停止规则。例如,在图 9.2 的右上图中,我们首先在 $X_1 = t_1$ 处分裂。然后将区域 $X_1 \leqslant t_1$ 在 $X_2 = t_2$ 处分裂,而区域 $X_1 > t_1$ 在 $X_1 = t_3$ 处分裂。最后,区域 $X_1 > t_3$ 在 $X_2 = t_4$ 处分裂。该过程的结果是将整个区域分裂成图中显示的五个区域 $R_1, R_2, \cdots, R_5$。对应的回归模型用区域 R_m 中的常量 c_m 预测 Y,即:

$$\hat{f}(X) = \sum_{m=1}^{5} c_m I\{(X_1, X_2) \in R_m\} \tag{9.9}$$

相同的模型可以用图 9.2 的左下图所示的二叉树表示。整个数据集位于树的顶部。在每个交叉点满足条件的观测被赋给左分支,而其余的赋给右分支。树的端节点或叶节点对应于区域 $R_1, R_2, \cdots, R_5$。图 9.2 的右下图所示的是这个模型的回归面的透视图。为了说明问题,我们选择节点均值 $c_1 = -5, c_2 = -7, c_3 = 0, c_4 = 2, c_5 = 4$ 制作了该图。

递归二叉树的主要优点是它的可解释性。特征空间的划分可以用一棵树充分刻画出来。

当输入多于两个时，如图 9.2 的右上图所示的划分就很难描绘，但二叉树仍然可以用相同的方式表示。这种表示在医学科学家中很流行，可能是因为它与医生的思维方式较吻合。基于患者的特性，这种树将总体划分为高低输出层。

9.2.2　回归树

现在，我们要讨论的问题是如何逐步生成回归树。我们的数据包括 p 个输入和一个响应，以及 N 个观测：即(x_i, y_i)，其中 $i = 1, 2, \cdots, N$，$x_i = (x_{i1}, x_{i2}, \cdots, x_{ip})$。算法需要自动确定分裂变量和分裂点，以及树应该有什么样的拓扑结构(形状)。首先，假设我们已经将空间划分成 M 个区域 R_1，R_2，$\cdots$，R_M，并且在每个区域内用常量 c_m 对响应建模：

$$f(x) = \sum_{m=1}^{M} c_m I(x \in R_m) \tag{9.10}$$

如果采用平方和 $\sum (y_i - f(x_i))^2$ 极小化作为我们的准则，则容易看到最好的 $\hat{c}_m$ 恰好是 y_i 在区域 R_m 的平均值：

$$\hat{c}_m = \text{ave}(y_i | x_i \in R_m) \tag{9.11}$$

根据平方和的极小值来发现最好的二叉划分通常在计算上是不可行的。因此，我们用贪心算法处理。从所有的数据开始，考虑一个分裂变量 j 和分裂点 s，并定义一对半平面：

$$R_1(j, s) = \{X | X_j \leqslant s\} \quad 和 \quad R_2(j, s) = \{X | X_j > s\} \tag{9.12}$$

然后搜索分裂变量 j 和分裂点 s，它求解：

$$\min_{j, s} \left[\min_{c_1} \sum_{x_i \in R_1(j,s)} (y_i - c_1)^2 + \min_{c_2} \sum_{x_i \in R_2(j,s)} (y_i - c_2)^2 \right] \tag{9.13}$$

对任意选择 j 和 s，内部极小化可以用下式求解：

$$\hat{c}_1 = \text{ave}(y_i | x_i \in R_1(j, s)) \quad 和 \quad \hat{c}_2 = \text{ave}(y_i | x_i \in R_2(j, s)) \tag{9.14}$$

对每个分裂变量，分裂点 s 的确定可以很快完成，因而通过扫描全部输入，确定最好的对偶(j, s)是可行的。

找到了最好的分裂，我们把数据划分成两个结果区域，并对每个区域重复分裂过程。然后对所有的结果区域重复这一过程。

我们将使这棵树增长到多大？很明显，一棵很大的树可能过分拟合数据，而较小的树可能无法捕获重要的结构。树的大小是控制模型复杂性的调整参数，树的最佳大小应该由数据自适应地选择。一种方法是：仅当分裂使平方和的降低超过某个阈值时，才分裂树节点。然而，这种策略过于短见，表面看来不值得的分裂却很可能导致在它之下的一个非常好的分裂。

一种可取的策略是增长一棵较大树的 T_0，仅当达到最小节点大小(比如 5)时才停止分裂过程。然后，利用现在就要介绍的代价复杂性剪枝(策略)来修剪这棵较大的树。

我们定义子树 $T \subset T_0$ 是通过修剪 T_0 得到的任意树，即通过坍缩任意数量的内部(非端)节点得到的树。用 m 表示端节点，且节点 m 代表区域 Rm。用 $|T|$ 代表树 T 中端节点的个数。令：

$$\hat{c}_m = \frac{1}{N_m}\sum_{x_i \in R_m} y_i$$
$$Q_m(T) = \frac{1}{N_m}\sum_{x_i \in R_m}(y_i - \hat{c}_m)^2 \tag{9.15}$$

定义代价复杂性准则：

$$C_\alpha(T) = \sum_{m=1}^{|T|} N_m Q_m(T) + \alpha|T| \tag{9.16}$$

其意图是对每个α找到子树 $T_\alpha \subseteq T_0$，使 $C_\alpha(T)$极小化。调整参数$\alpha \geqslant 0$来控制树的大小与它对数据的拟合程度之间的折中。较大的α值导致较小的树 T_α，对较小的α值则情况相反。正如记法所暗示的，当$\alpha = 0$时，结果是整棵树 T_0。下面，我们讨论如何自适应地选择α。

对每个α，可以证明存在惟一的最小子树 T_α，它最小化 $C_\alpha(T)$。为了找出 T_α，我们使用最弱链接剪枝(weakest link pruning)：相继地坍缩在 $\sum_m N_m Q_m(T)$ 上产生最小增长的内部节点，并继续进行该过程，直到产生一棵单节点(根)树。这就给出了一个(有限的)子树序列，并且可以证明该序列一定包括 T_α。详细内容参考 Breiman 等人(1984)或 Ripley(1996)的论文。α的估计通过 5 或 10 折交叉验证来实现：我们选择值$\hat{\alpha}$来极小化交叉验证平方和。最终的树是 T_α。

9.2.3 分类树

如果目标是对取值 $1,2,\cdots,K$ 的输出分类，那么，仅需要对树算法的分裂节点和修剪树准则进行修改。对于回归，我们使用式(9.15)中定义的平方误差节点非纯度量 $Q_m(T)$，但它并不适合于分类。在代表具有 N_m 个观测区域 R_m 的节点 m，令：

$$\hat{p}_{mk} = \frac{1}{N_m}\sum_{x_i \in R_m} I(y_i = k)$$

表示节点 m 中类 k 的观测比例。我们将节点 m 中的观测分到类 $k(m) = \arg\max_k \hat{p}_{mk}$ 中，它是节点 m 上的多数类。不同的节点非纯度量 $Q_m(T)$包括如下几种：:

$$\text{误分类误差：} \frac{1}{N_m}\sum_{i \in R_m} I(y_i \neq k(m)) = 1 - \hat{p}_{mk(m)}$$
$$\text{Gini 索引：} \sum_{k \neq k'} \hat{p}_{mk}\hat{p}_{mk'} = \sum_{k=1}^{K} \hat{p}_{mk}(1 - \hat{p}_{mk})$$
$$\text{互熵或散离：} -\sum_{k=1}^{K} \hat{p}_{mk}\log\hat{p}_{mk} \tag{9.17}$$

对于两个类，如果 p 是在第二个类中的比例，则这三种度量分别是 $1 - \max(p, 1-p)$，$2p(1-p)$和 $-p\log p - (1-p)\log(1-p)$。如图 9.3 所示。这三种度量相似，但互熵和 Gini 索引是可微的，更适合于数值优化。

另外，互熵和 Gini 索引对节点中概率的改变比误分类率更加敏感。例如，在每个类包含 400 个观测的二类问题中[记做(400,400)]，假设一个分裂产生了节点(300,100)和(100,300)，而另外一个分裂产生了节点(200,400)和(200,0)。两种分裂产生的误分类率都是 0.25，但第二种分裂产生了一个纯节点，而且可能是更可取的。对于第二种分裂，Gini 索引和互熵较低。基于这个原因，树增长应当使用 Gini 索引或互熵度量。为指导代价复杂性剪枝，三种测量都

可以使用，但最典型的还是误分类率。

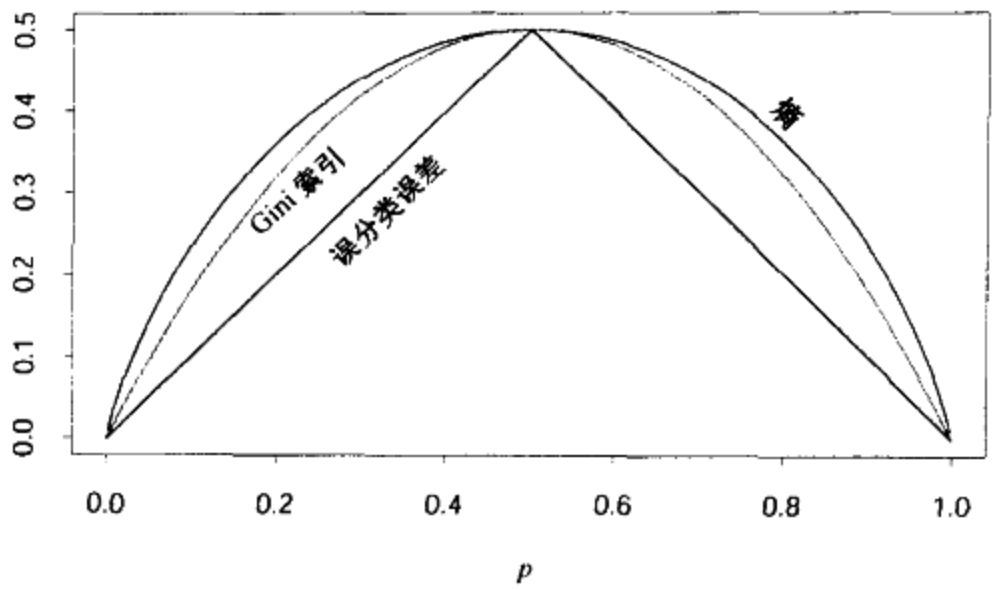


图 9.3　2-类分类的节点非纯度量，是类 2 中的比例 p 的函数。互熵经过点(0.5,0.5)(见彩页)

可以用两种有趣的方式解释 Gini 索引。我们不把观测分类到节点的多数类中，而是将它们分类到概率为 $\hat{p}_{mk}$ 的类 k 中。则该节点上的训练误差是 $\sum_{k \neq k'} \hat{p}_{mk}\hat{p}_{mk'}$ ——Gini 索引。类似地，如果我们将类 k 的每个观测用 1 编码，其他观测用 0 编码，则该 0-1 响应在节点上的方差是 $\hat{p}_{mk}(1-\hat{p}_{mk})$。在类 k 上求和又得到 Gini 索引。

9.2.4　其他问题

分类的预测子

分裂一个具有 q 个无序值的预测子时，把 q 个值分成两组的可能划分多达 $2^{q-1}-1$ 种，而对于较大的 q，计算量更是惊人的。然而，使用 0-1 输出，计算则可以简化。根据落入输出类 1 的比例对预测子类排序。然后，对预测子进行分裂，就像它是一个有序的预测子一样。可以证明，在所有可能的 $2^{q-1}-1$ 种分裂中，该方法在平方误差或 Gini 索引意义下将产生最佳分裂。对于定量输出，该结论也成立——诸类将按其输出均值的增加来排序(Breiman 等人，1984)。

损失矩阵

在分类问题中，观测的误分类后果对于某些类要比其他类更为严重。例如，预测一个人不会患心脏病而实际上他/她患了心脏病可能比反过来更糟糕。为了解决该问题，我们定义一个 $K \times K$ 的损失矩阵 $\mathbf{L}$，其中 $L_{kk'}$ 是由将类 k 观测分类为类 k' 而导致的损失。正确的分类不会导致损失，即对任意 k，$L_{kk}=0$。为了把损失合并到建模过程中，我们可以把 Gini 索引修改成 $\sum_{k \neq k'} L_{kk'}\hat{p}_{mk}\hat{p}_{mk'}$；这是由随机规则带来的期望损失。这对多类情况有效，而对两类情况则无效，原因是 $\hat{p}_{mk}\hat{p}_{mk'}$ 的系数是 $L_{kk'}+L_{k'k}$。对两类情况，一个较好的方法是在类 k 中用 $L_{kk'}$ 对观测加权。仅当作为 k 的函数 $L_{kk'}$ 不依赖于 k' 时，这种方法才能用于多类情况。观测加权也可以与散离一起使用，观测加权的作用是改变类的先验概率。在端节点，经验贝叶斯规则提示我们分类到类 $k(m) = \arg\min_k \sum_\ell L_{\ell k}\hat{p}_{m\ell}$ 中。

遗漏预测子值

假设我们的数据中某些或全部变量存在一些遗漏的预测子值。我们可以丢弃任何具有遗

漏值的观测，但是这会引起训练集的严重减少。替换地，可以尝试填补遗漏值，比如说，用预测子在完整观测上的均值来填补遗漏值。对于基于树的模型，有两种较好的方法。第一种可用于分类的预测子：简单地把"遗漏"看做一个新类。由此可以发现，对某些度量，具有遗漏值的观测与没有遗漏值的观测的行为是不同的。第二种更一般的方法是构造代理变量。当考虑用一个预测子分裂时，我们仅使用该预测子没有遗漏的观测。选择了最好的(初始的)预测子和分裂点之后，构造一个代理预测和分裂点的表。第一个代理是能够最好地模拟初始分裂处理训练数据的预测子和相应的分裂点。第二个代理是做得次好的预测子和相应的分裂点，以此类推。当在训练阶段或预测期间沿着树向下传递观测时，如果初始的分裂预测子遗漏，我们就按照顺序使用代理分裂。代理分裂利用预测之间的相关性试图缓解遗漏数据的影响。具有遗漏值的预测子和其他预测子之间的相关性越高，由于遗漏值引起的信息损失就越小。遗漏数据的一般性问题在第 9.6 节讨论。

为什么使用二叉分裂

在每个阶段，我们可以考虑对每个节点进行多路分裂来分成多个组，而不是把每个节点都分裂成两组(如上所述)。尽管这样做有时是有用的，但它却不是一个好的一般性策略。问题是多路分裂会很快地把数据分裂成碎片，导致下一层上的数据不足。所以，仅在必要时我们才希望使用这种分裂。由于多路分裂可由一系列的二叉分裂实现，所以后者更可取。

其他树构建过程

上述讨论集中在树的 CART(分类和回归树)实现上。其他流行的方法是 ID3 和它的后继版本 C4.5 和 C5.0(Quinlan，1993)。程序的早期版本仅限于分类的预测子，并使用一种不带剪枝的自上而下规则。利用更新的研究，C5.0 已经变得与 CART 非常相近。C5.0 独有的显著特征是产生规则集的方案。树构造好之后，定义端节点的分裂规则有时可以简化：即在不改变属于该节点的观测子集的情况下，可以去掉一个或多个条件。最后，我们得到定义每个端节点的简化规则集；它们将不再遵循树的结构，但简洁性可能对用户更有吸引力。

线性组合分裂

我们可以按照 $\sum \alpha_j X_j \leqslant s$ 形式的线性组合进行分裂，而不是将分裂局限于 $x_j \leqslant s$ 这种形式。对权 α_j 和分裂点 s 进行优化，以便极小化相关准则(如 Gini 索引)。尽管这样做可能增强树的预测能力，但也可能破坏其可解释性。在计算方面，分裂点搜索的离散性阻碍了权值光滑优化的使用。一种体现线性组合分裂较好的方法是采用专家分级混合(HME)模型，这是第 9.5 节的主题。

树的不稳定性

树的主要问题是它具有较高的方差。通常，数据的一个较小变化将导致一系列完全不同的分裂，使得解释有些不稳定。造成这种不稳定的主要原因是过程的分层本性：顶层分裂中的错误影响被传播到下面的所有分裂。通过尝试使用较稳定的分裂准则，我们可以在某种程度上缓解这种影响，但无法消除这种固有的不稳定性。这是从数据估计一个简单的、基于树的结构所付出的代价。装袋(见第 8.7 节)通过对许多树求平均来降低方差。

缺乏光滑性

树的另一个局限性是预测面缺乏光滑性，正如我们在图 9.2 右下图中看到的。在具有 0/1 损失的分类中，这不会造成太大损害，因为类概率估计中偏倚的影响有限。然而，它可能降低回归处理的性能；对于回归，我们通常期望回归函数是光滑的。第 9.4 节介绍的 MARS 过程可以看做是 CART 的修订，旨在缓解这种光滑性的缺乏。

捕获加法结构的困难

树的另一个问题是很难对加法结构建模。在回归中，假设 $Y = c_1 I(X_1 < t_1) + c_2 I(X_2 < t_2) + \varepsilon$，其中 ε 是 0 均值噪声。那么，二叉树可能在接近 t_1 的 X_1 上进行第一次分裂。为了捕获加法结构，下一层必须在接近 t_2 的 X_2 上分裂两个节点。这种情况可能在数据充足的情况下发生，但是模型并没有给出特别的支持来发现这种结构。如果有十个而不是两个加法效应，则必须采取许多偶然的分裂来重建这种结构，而数据分析者将很难在估计树中识别它。这里的“责任”还应该归咎于既有优点又有缺点的二叉树结构。为了获取加法结构，MARS 方法也放弃了树结构(见第 9.4 节)。

9.2.5　例：垃圾邮件(续)

我们把分类树方法应用于前面介绍的垃圾邮件例子。使用散离度量来增长树，并使用误分类率对它修剪。图 9.4 显示了 10 折交叉验证的误差率，误差率是被剪枝树规模的函数，有 ±2 倍均值的标准误差，来自 10 次重复。检验误差曲线用红色来显示。注意，交叉验证误差率由一系列α值，而不是由树的大小来标引。对于用不同的折增长的树，一个α值可能意味不同的大小。显示在图底部树的大小是指剪枝后原始树的大小 $|T_\alpha|$。

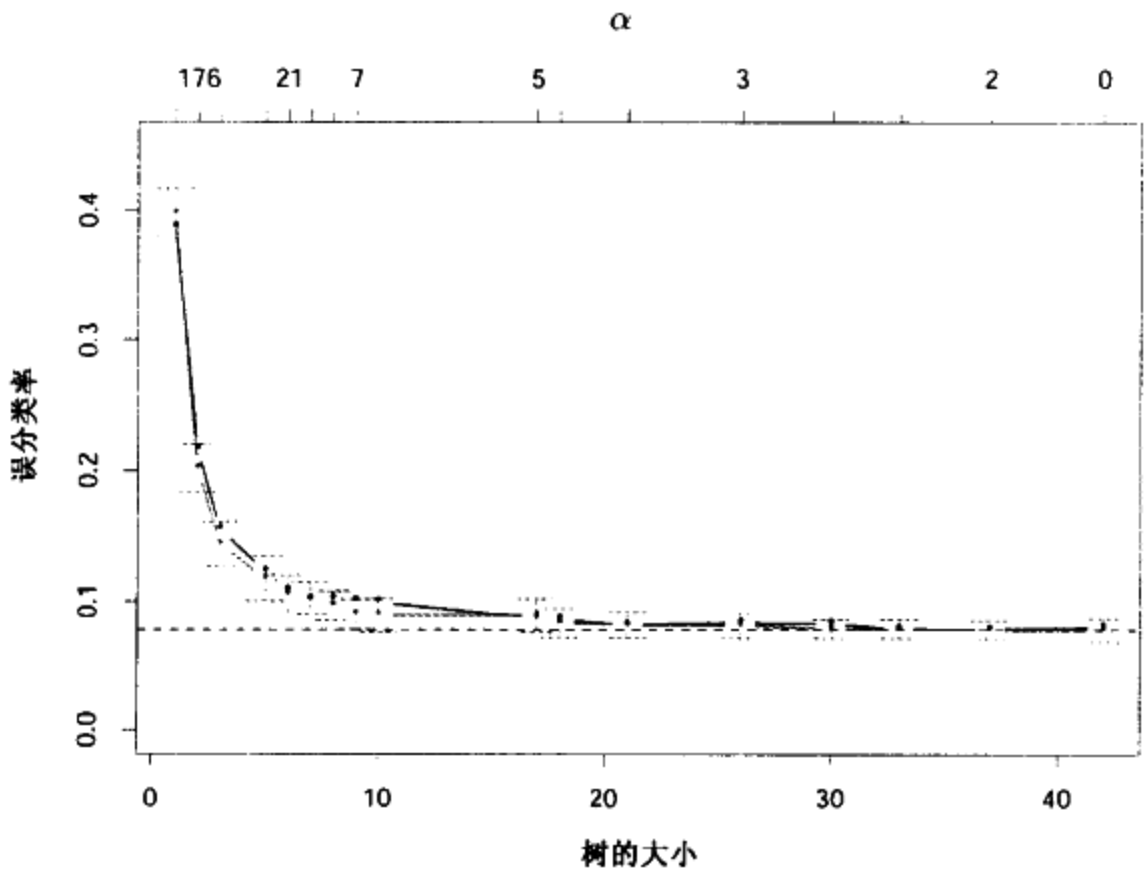


图 9.4　垃圾邮件例子的结果。绿色曲线是误分类率的10折交叉验证估计，误分类率是树规模的函数，有 ±2倍标准误差条。极小值出现在大小约有17个端节点的树上。红色曲线是检验误差，它与 CV 误差非常接近。交叉验证被α标引，在图中上方显示。显示在图底部树的大小是指剪枝后原始树的大小 $|T_\alpha|$(见彩页)

误差曲线在大约有 17 个端节点时变得平坦，图 9.5 中给出了剪枝树。在树选定的 13 个不同的特征中，11 个特征与加法模型中的 16 个显著特征吻合(见表 9.2)。表 9.3 中显示的总体误差率大约比表 9.1 中的加法模型高出 50%。

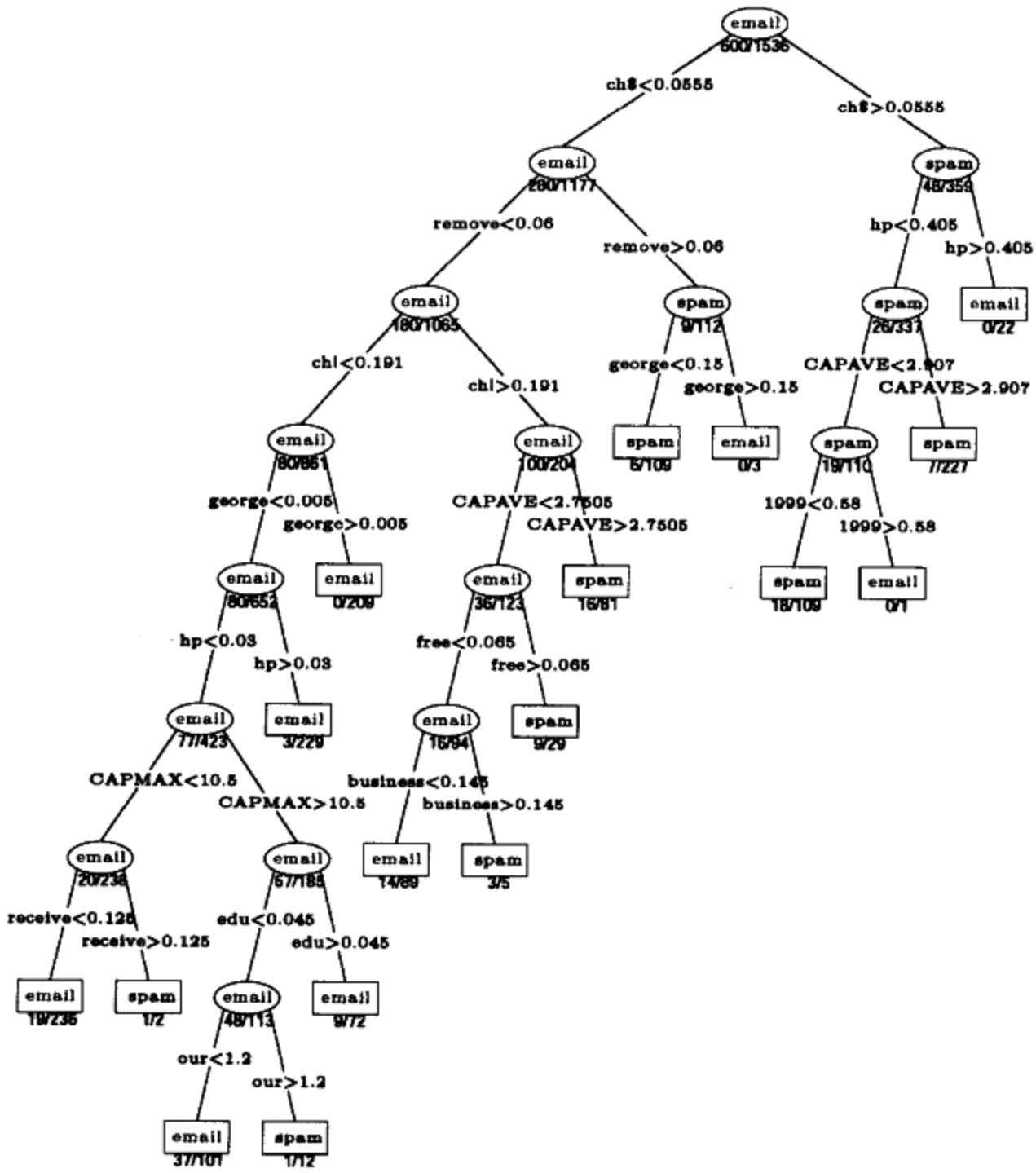


图 9.5 垃圾邮件例子的剪枝树。分裂变量在分枝上显示，而分类在每个节点上显示。端节点下面的数据给出了检验数据上的误分类率

表 9.3 垃圾邮件数据：17 个节点(由交叉验证挑选)的树在检验数据上的误差率。总体误差率是 9.3%

	预测的	
真	email	spam
email	57.3%	4.0%
spam	5.3%	33.4%

考虑树的最右分支。如果超过 5.5% 的字符是 $，我们将进入带有 spam 警告的右分支。然而，如果又有短语 hp 出现频繁，那么这就和公司的事务很像，我们就按 email 分类，则满足这些准则的检验集中的全部 22 个实例都被正确分类了。如果不满足第二个条件，重复大写字母的

平均长度 CAPAVE 又超过 2.9,那么就按 spam 分类。227 个检验实例只有 7 个被误分类了。

在医学分类问题中,术语敏感性(sensitivity)和特效性(specificity)用来刻画规则。它们的定义如下:

敏感性:真实状态是患病而预测为患病的可能性。

特效性:真实状态是没有患病而预测为没有患病的可能性。

如果分别把 spam 和 email 看做患病和没有病,那么,由表 9.3 有:

$$\text{敏感性} = 100 \times \frac{33.4}{33.4 + 5.3} = 86.3\%$$

$$\text{特效性} = 100 \times \frac{57.3}{57.3 + 4.0} = 93.4\%$$

在这个分析中,我们使用的是等损失。和前面一样,用 $L_{kk'}$ 表示把类 k 的对象预测为类 k' 的损失。通过改变损失 L_{01} 和 L_{10} 的相对大小,我们增加规则的敏感性而降低规则的特效性,或相反。在这个例子中,我们希望避免把正常 email 标记成 spam,因此希望特效性非常高。比如说,可以通过置 $L_{01} > 1, L_{10} = 1$ 来做到这一点。如果 spam 的比例是大于或等于 $L_{10}/(L_{10} - L_{01})$,则在每个端节点,贝叶斯规则就分类到类 1(spam),否则就分类到类 0(email)。接收子操作特征曲线(ROC)通常用于概括敏感性和特效性之间的权衡的评估。随着我们改变分类规则的参数,它用于描绘敏感性与特效性。在 0.1 和 10 之间改变损失 L_{01},并对图 9.4 中选择的 17 个节点的树应用贝叶斯规则,产生的 ROC 曲线显示在图 9.6 中。我们看到,为了使特效性接近 100%,敏感性就不得不降至大约 50%。曲线下面的面积是常用的定量汇总;在每个方向上线性地扩展曲线,使它定义在[0,100]上,该面积接近于 0.95。为了对比,我们已经包括了拟合第 9.2 节中那些数据的 GAM 模型的 ROC 曲线;对于任意损失,它都将给出一个较好的分类规则,面积为 0.98。

更好的方法不是仅在节点中修改贝叶斯规则,而是在树增长过程中考虑不相等损失,就像我们在第 9.2 节所做的。对于两个类 0 和 1,通过对类 k 中的观测使用权值 $L_{k,1-k}$,可以把损失合并到树增长过程中。这里,选择 $L_{01} = 5, L_{10} = 1$ 并拟合与以前同样大小($|T_\alpha| = 17$)的树。在较高的特效性值上,与原来的树相比,该树具有较高的敏感性,但是在其他极端情形却更差一些。它的顶部有少量分裂和原来的树一样,然后,就从此处分开了。对于这种应用,使用 $L_{01} = 5$ 增长的树明显比最初的树要好。

9.3　PRIM——凸点搜索

基于树的模型(用于回归)把特性空间划分成箱形区域,目的是使每个箱(box)的响应平均值尽可能不同。定义箱的分裂规则通过二叉树相互关联,有利于它们的解释。

忍耐规则归纳方法(PRIM)也是在特性空间中寻找箱,但搜索的是响应平均值高的箱。因此,它搜索目标函数的极大值,称为凸点搜索(bump hunting)。如果希望搜索极小值而不是极大值,可以简单地用负响应值来处理。

PRIM 也不同于基于树的划分方法,箱的定义不使用二叉树描述。这使得规则集的解释更加困难;然而,通过去掉二叉树限制,单个规则却常常比较简单。

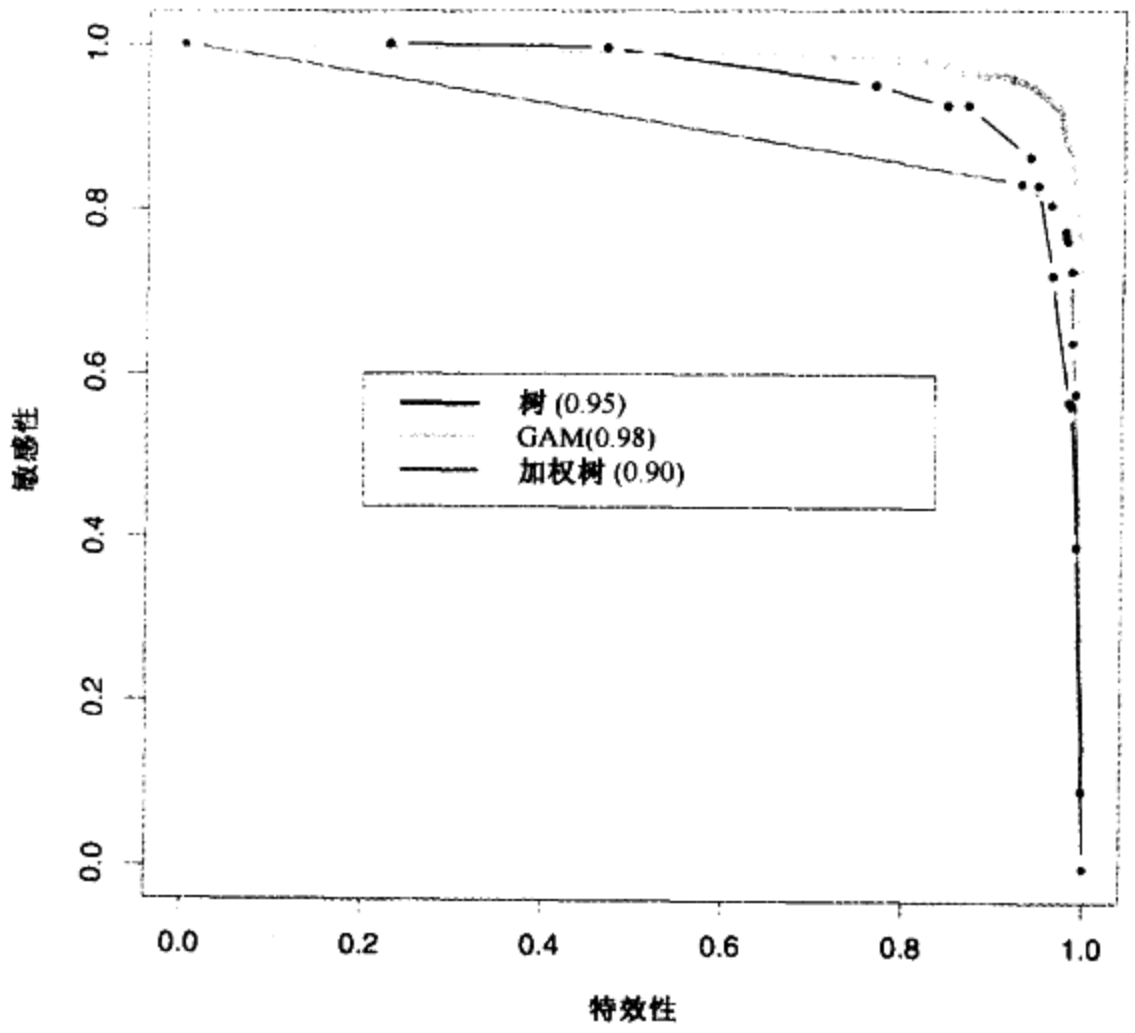


图 9.6 拟合垃圾邮件数据的分类规则的 ROC 曲线。靠近右上角的曲线展示了较好的分类器。在此情况下，GAM分类器优于树。对于较高的特效性，加权树比不加权树能获得较好的敏感性。插图中的数字表示曲线下面的面积(见彩页)

PRIM 中主要的箱构造过程是自上而下进行的，从包含全部数据的箱开始。沿着一个面对箱压缩一个较小的量，落到箱外的观测就被剥掉。压缩完成之后，被压缩选择的面是导致最大箱均值的面。然后，重复这个过程，直到当前箱包含数据点的某个最小个数为止。

图 9.7 中显示了这一过程。200 个数据点均匀分布在单位正方形上。彩码图表明，当 $0.5 < X_1 < 0.8$ 且 $0.4 < X_2 < 0.6$ 时，响应 Y 的取值为 1(红色)，否则为 0(蓝色)。这些图显示了自上而下剥除过程相继发现的箱，每次剥除剩余数据点的比例为 $\alpha = 0.1$。

图 9.8 显示了箱被压缩时箱中响应值的均值。

计算出自上而下的序列后，PRIM 就反转这一过程；如果一个扩展能够增加箱均值，就沿着一条边进行扩展。这个过程称做粘贴(pasting)。由于自上而下过程在每一步上都是贪心的，所以这样的扩展通常是可能的。

这些步骤的执行结果是一个箱序列，每个箱中的观测数量都不同。交叉验证结合数据分析的判断，用来选择最佳箱容量。

用 B_1 表示步骤 1 发现的箱中观测的下标。PRIM 过程从训练集中删除 B_1 中的观测，并在余下的数据集上重复两步过程——自上而下剥除，然后自下而上粘贴。全部过程重复若干次，产生一个箱序列 $B_1, B_2, \cdots, B_k$。每个箱由涉及预测子子集的规则集定义，如

$$(a_1 \leqslant X_1 \leqslant b_1) \text{和} (b_1 \leqslant X_3 \leqslant b_2)$$

算法 9.3 给出了 PRIM 过程。

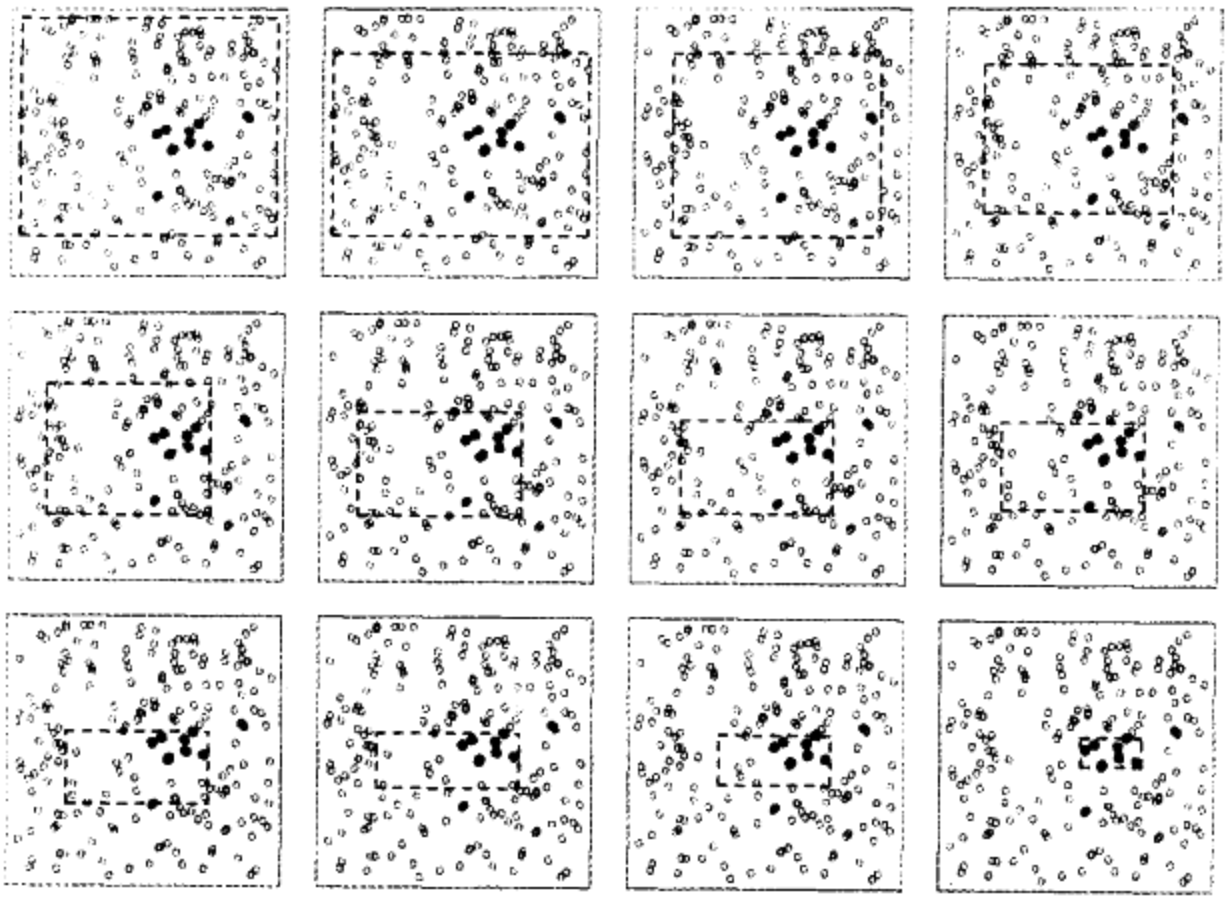


图 9.7　PRIM 算法图解。这里有两个类，分别用蓝点(类 0)和红点(类 1)指示。过程从包围所有数据的矩形(黑色虚线)开始，然后，按预先指定的量沿一条边剥除点，使得留在箱中的点的均值极大化。从左上图开始，显示剥除序列，直到最右下图纯红色区域被隔离为止(见彩页)

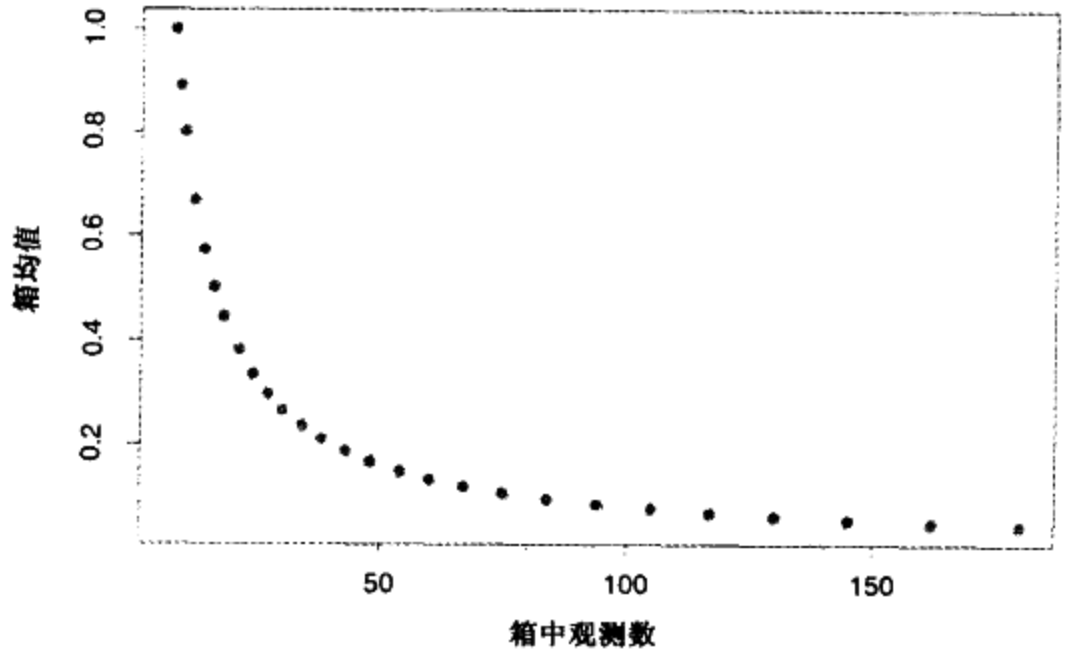


图 9.8　箱均值作为箱中观测数量的函数

算法 9.3　忍耐规则归纳方法

1. 从全部训练数据和包括全部数据的最大箱开始
2. 考虑通过压缩一个面对箱进行收缩，以便能剥除比例为α的观测；这些观测在某预测子 X_j 上具有最大值或最小值。选择这样的剥除，它可以在留在箱中的观测上产生最高响应均值(典型地，$\alpha = 0.05$ 或 0.10)
3. 重复步骤 2，直到某最小个数(比如 10 个)的观测留在箱中
4. 沿着任意一个面对箱进行扩展，只要结果箱均值增加
5. 步骤 1 到步骤 4 给出一个箱序列，每个箱中的观测数量都不同。使用交叉验证从序列中选择一个箱，称该箱为 B_1
6. 从数据集中删除箱 B_1 中的数据，并重复步骤 2 到步骤 5 获得第二个箱；继续下去，获得所需要的一些箱

像 CART 一样，PRIM 也可以通过考虑预测子的全部划分来处理分类预测子。遗漏值也用类似于 CART 的方式来处理。PRIM 是为回归(定量响应变量)设计的；一个 2-类输出可以简单地通过用 0 和 1 对其编码来处理。没有简单的方法同时处理 $k>2$ 个类。一种方法是对每个类相对一个基准类分别运行 PRIM。

相对于 CART，PRIM 的优点是其忍耐性。由于二叉分裂，CART 快速地将数据分割成碎片。假设采用等尺寸分裂，对 N 个观测，在用完数据之前它可以只做 $\log_2(N)-1$ 次分裂。如果在每个阶段，PRIM 按比例α剥除训练点，则在用完数据之前，它大约要做 $-\log(N)/\log(1-\alpha)$次剥除。例如，如果 $N=128$，且$\alpha=0.10$，那么 $\log_2(N)-1=6$，而 $-\log(N)/\log(1-\alpha)\approx 46$。考虑到在每个阶段，观测的数量一定是整数，实际上，PRIM 只能做 29 次剥除。在任何情况下，PRIM 较强的忍耐性将帮助自上而下贪心算法发现较好的解。

9.3.1 例：垃圾邮件(续)

我们把 PRIM 应用于垃圾邮件数据，对于 spam，响应的编码是 1，email 的编码是 0。

由 PRIM 发现的前两个箱概括如下：

规则 1	全局均值	箱均值	箱支持度
训练	0.3931	0.9607	0.1413
检验	0.3958	1.0000	0.1536

$$\text{规则 1}\begin{cases} \text{ch!} > 0.029 \\ \text{CAPAVE} > 2.331 \\ \text{your} > 0.705 \\ 1999 < 0.040 \\ \text{CAPTOT} > 79.50 \\ \text{edu} < 0.070 \\ \text{re} < 0.535 \\ \text{ch;} < 0.030 \end{cases}$$

规则 2	全局均值	箱均值	箱支持度
训练	0.2998	0.9560	0.1043
检验	0.2862	0.9264	0.1061

$$\text{规则 2}\begin{cases} \text{remove} > 0.010 \\ \text{george} < 0.110 \end{cases}$$

箱支持度是落在箱中的观测所占的比例。第一个箱是纯 spam，包括测试数据的大约 15%。第二个箱包括测试观测的 10.6%，其中的 92.6% 是 spam。两个箱总共包括数据的 26%，并且是大约 97% 的 spam。下面的几个箱(没有显示)非常小，包括大约 3% 的数据。

预测子按照重要性次序列出。有趣的是在 CART 树(见图 9.5)顶部的分裂变量并没有出现在第一个箱中。

9.4　MARS：多元自适应回归样条

MARS 是一个自适应的回归过程，很适合高维问题（即大量的输入）。我们可以把它看做逐步线性回归的泛化，或者看做是对 CART 方法的修改，以改善其在回归处理中的性能。我们在第一种观点下引进 MARS，后者建立与 CART 的联系。

MARS 使用形如 $(x-t)_+$ 和 $(t-x)_+$ 的分段线性基函数展开式。“+”表示正的部分，所以

$$(x-t)_+ = \begin{cases} x-t & \text{如果 } x>t \\ 0 & \text{否则} \end{cases} \quad \text{并且} \quad (t-x)_+ = \begin{cases} t-x & \text{如果 } x<t \\ 0 & \text{否则} \end{cases}$$

作为一个例子，函数 $(x-0.5)_+$ 和 $(0.5-x)_+$ 显示在图 9.9 中。

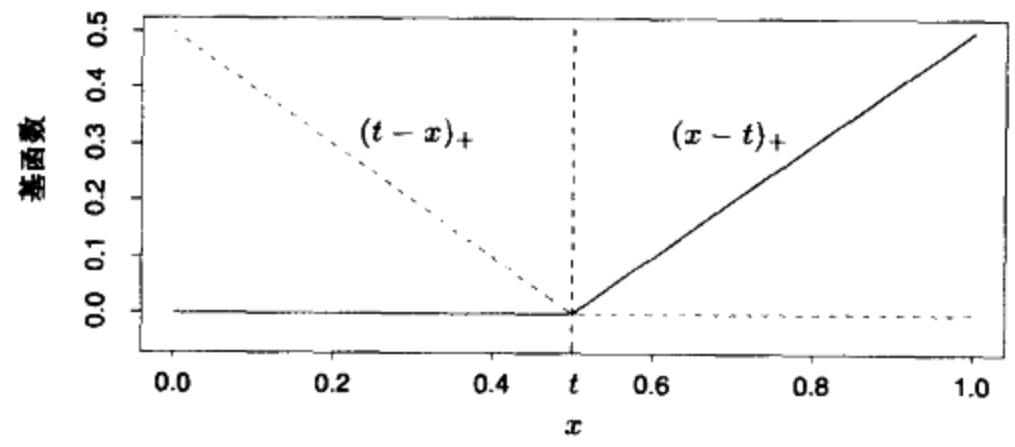


图 9.9　MARS 使用的基函数 $(x-t)_+$（红色实线）和 $(t-x)_+$（绿色虚线）（见彩页）

每个函数是分段线性的，纽结在值 t 上。用第 5 章的术语，这些是线性样条。在下面的讨论中，我们称这两个函数为反演对（reflected pair）。其思想是，对于在输入的每个观测值 x_{ij} 处具有纽结的每个输入 X_j，形成反演对。因此，基函数的集合是：

$$C = \{(X_j - t)_+,\ (t - X_j)_+\}_{\substack{t \in \{x_{1j}, x_{2j}, \cdots, x_{Nj}\} \\ j = 1, 2, \cdots, p}} \tag{9.18}$$

如果所有的输入值都不相同，则总共有 $2Np$ 个基函数。注意，尽管每个基函数只依赖于单个 X_j，例如，$h(X) = (X_j - t)_+$，但仍可把它看做整个输入空间 $\mathbb{R}^p$ 上的函数。

模型构造策略类似于前向逐步线性回归，但我们允许使用集合 C 中的函数和它们的积，而不是使用原始输入。这样，模型具有如下形式：

$$f(X) = \beta_0 + \sum_{m=1}^{M} \beta_m h_m(X) \tag{9.19}$$

其中，每个 $h_m(X)$ 是 C 中的函数，或是两个或多个这种函数的乘积。

给定 h_m 的一个选择，系数 β_m 通过极小化残差的平方和来估计；即通过标准的线性回归来估计。然而，真正的技巧在于函数 $h_m(x)$ 的构造上。开始，模型中仅包含常量函数 $h_0(X) = 1$，并且集合 C 中的函数都是候选；如图 9.10 所示。

在每个阶段，考虑模型集 $\mathcal{M}$ 中的一个函数 h_m 与 C 中反演对中一个函数的积，将所有这样的积看做是一个新的基函数对。我们将如下形式的项添加到模型 $\mathcal{M}$ 中：

$$\hat{\beta}_{M+1} h_\ell(X) \cdot (X_j - t)_+ + \hat{\beta}_{M+2} h_\ell(X) \cdot (t - X_j)_+,\ h_\ell \in \mathcal{M}$$

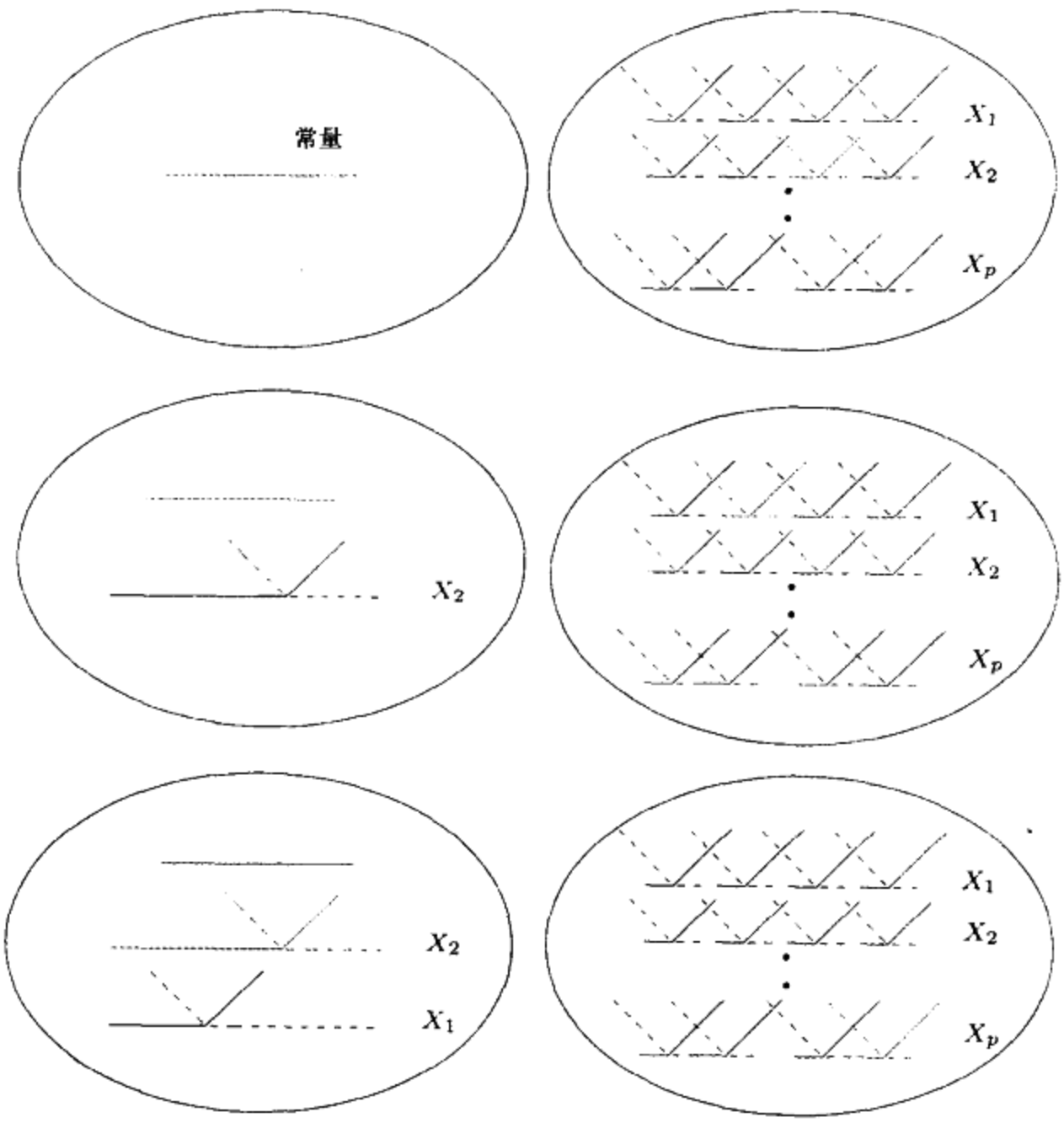


图 9.10 MARS 前向模型建立过程示意图。左侧是当前模型中的基函数：初始时，它是常量函数 $h(X)=1$。右侧是在构造模型时需要考虑的全部候选基函数。它们是图9.9中所示的分段线性函数对，纽结 t 在每个预测 X_j 的全部惟一观测值 x_{ij} 上。在每一步，我们考虑候选对与模型中基函数的所有积。并将最大程度降低残差的积添加到当前模型中。上面我们描述了过程的前三个步骤，所选择的函数用红色表示（见彩页）

它能最大限度降低训练误差。这里，$\hat{\beta}_{M+1}$ 和 $\hat{\beta}_{M+2}$ 是系数，它们与模型的其他 $M+1$ 个系数一起，用最小二乘方估计。获胜的积添加到模型中，并且继续该过程，直到模型集 $\mathcal{M}$ 中项的个数达到预先设定的最大个数。

例如，在第一步，因为一个函数与常量函数的乘积结果是函数本身，我们考虑把形式为 $\beta_1(X_j-t)_+ + \beta_2(t-X_j)_+; t\in\{x_{ij}\}$ 的函数添加到模型中。假设最好的选择是 $\hat{\beta}_1(X_2-x_{72})_+ + \hat{\beta}_2(x_{72}-X_2)_+$，则这个基函数对被增加到模型集 $\mathcal{M}$ 中，而且在下一个阶段，我们考虑包括如下形式的乘积对：

$$h_m(X)\cdot(X_j-t)_+ \text{ 和 } h_m(X)\cdot(t-X_j)_+, t\in\{x_{ij}\}$$

其中，对于 h_m，我们有如下选择：

$$h_0(X) = 1$$
$$h_1(X) = (X_2 - x_{72})_+$$
$$h_2(X) = (x_{72} - X_2)_+$$

第三种选择产生如$(X_1 - x_{51})_+ \cdot (x_{72} - X_2)_+$的函数，如图 9.11 所示。

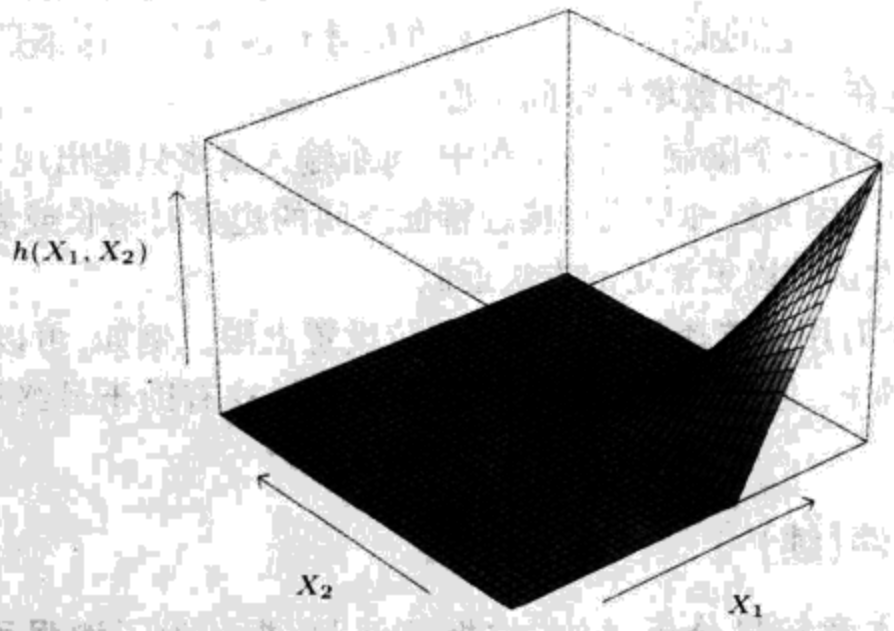


图 9.11　函数 $h(X_1, X_2) = (X_1 - x_{51})_+ \cdot (x_{72} - X_2)_+$，由两个分段线性的 MARS 基函数的乘积产生

该过程结束时，我们得到一个形如式(9.19)的大模型。典型地，该模型过分拟合数据，为此使用一个后向删除过程。在每一步，从模型中删除引起残差平方和增长最少的项，产生大小为 λ(项的数)的估计最佳模型 $\hat{f}_\lambda$。可以用交叉验证估计 λ 的最佳值，但为了节省计算开销，MARS 改为使用广义交叉验证。这个准则定义为：

$$\mathrm{GCV}(\lambda) = \frac{\sum_{i=1}^{N}(y_i - \hat{f}_\lambda(x_i))^2}{(1 - M(\lambda)/N)^2} \tag{9.20}$$

值 $M(\lambda)$是模型中有效的参数个数：它是模型中项的个数，加上用于选择纽结的最佳位置的参数的个数。一些数学和模拟结果表明，在分段线性回归中要选择一个纽结应该付出三个参数的代价。

这样，如果模型中有 r 个线性独立的基函数，并且在前向过程中选择了 K 个纽结，则该公式是 $M(\lambda) = r + cK$，其中 $c = 3$(当模型被限制为加法模型时，使用 $c = 2$ 的罚；细节在下面给出)。这样，我们根据极小化 GCV(λ)的后向序列来选择模型。

为什么使用分段线性基函数，为什么使用特殊模型策略？图 9.9 的函数的关键特性是它们的局部操作能力；它们在其变程的一部分上取值 0。当把它们乘在一起时，如图 9.11 所示，只在其特征空间的一小部分上结果非 0；在这一小部分中，两个分量函数均非 0。结果，仅在需要它们的地方局部地使用非 0 分量，回归面极度俭省地建立起来了。这很重要，在高维中我们应该谨慎地“消耗”参数，因为它们可能很快用完。使用其他基函数，如多项式，将会产生处处非 0 的积，并且也行不通。

分段线性基函数的第二个重要优点与计算有关。考虑$\mathcal{M}$中的一个函数与输入 X_j 上的每个反演对(共计 N 个反演对)的积。这似乎需要拟合 N 个单输入线性回归模型，每个需要 $O(N)$次操作，总共需要 $O(N^2)$次操作。然而，我们可以利用分段线性函数的简单形式。首

先,拟合具有最右结的反演对。当这个结在某时刻被连续地移到左部的一个位置上,这些基函数就由于在域的左部取0、在右部取一个常量而不同。因此,在每一次移动之后,我们可以用$O(1)$次操作更新拟合。这样,仅用$O(N)$次操作就可以尝试每一个纽结。

MARS中的前向建模策略在如下意义下是分层的:多路积用已经包含在模型中项的积来构造。例如,如果一个三路积的分量已经在模型中,一个四路积才可以添加到模型中。其道理是,一个高阶交叉项多半在它的低阶"足迹"也存在时才可能存在。这不是绝对的,但却是合理的假设,而且可以避免在一个指数增长空间上搜索。

对模型中项的形式有一个限制:在一个积中,每个输入最多只能出现一次。这样可以防止一个输入的高阶幂形式,因为这种形式在接近特征空间的边界时增长或者降低特别快。这种幂形式可以用分段线性函数以更稳定的方法近似。

MARS过程中一个有用的选择是对交叉积的阶设置上限。例如,可以设置上限为2,允许分段线性函数的两两乘积,但不允许三路或更多路乘积。这有助于最终模型的解释。上限1将导致产生加法模型。

9.4.1 例:垃圾邮件(续)

把MARS应用于本章较早分析过的"垃圾邮件"数据。为了增强可解释性,我们限制MARS为二阶交叉积。尽管目标是一个2-类变量,但我们仍然使用平方误差损失函数(见第9.4.3节)。图9.12显示了检验误差误分类率,作为模型秩(独立的基函数个数)的函数。误差率水平大约在5.5%附近,稍高于前面讨论过的广义加法模型(5.3%)。GCV选择一个大小为60的模型,大体上是给出最优性能的最小模型。由MARS建立的最主要的交互作用涉及到输入(ch $, remove)、(ch $, free)和(hp, CAPTOT)。然而,这些交互作用并没有改进广义加法模型的性能。

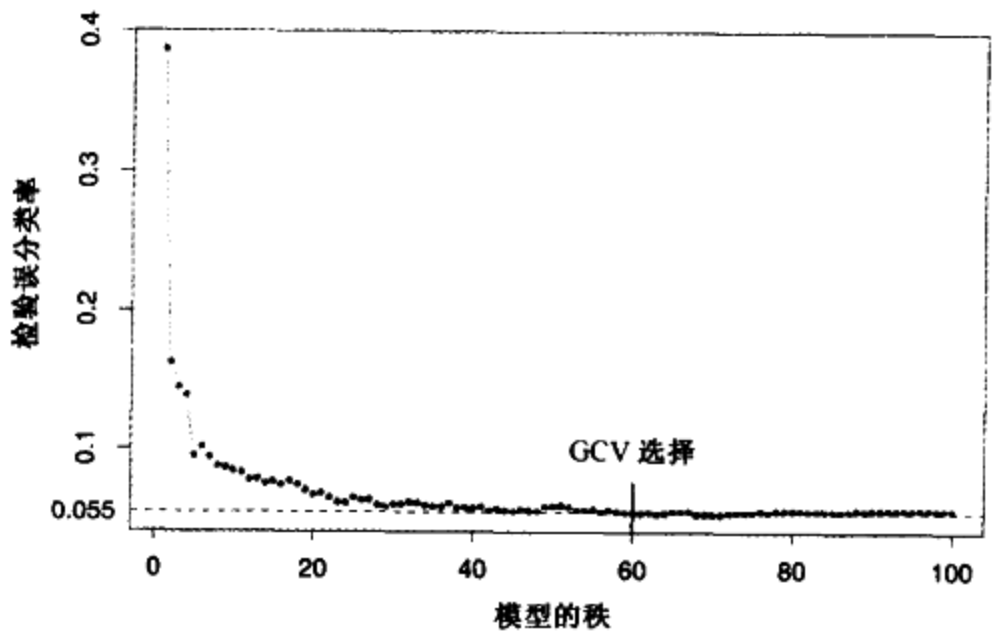


图9.12 垃圾邮件数据:MARS过程的检验误差误分类率,作为模型秩(独立基函数的个数)的函数

9.4.2 例(模拟数据)

这里,我们用三种对比方案考察MARS的性能。我们有$N=100$个观测和预测子$X_1, X_2, \cdots, X_p$,并且误差ϵ服从独立的标准正态分布。

方案 1　数据产生模型是：

$$Y = (X_1 - 1)_+ + (X_1 - 1)_+ \cdot (X_2 - .8)_+ + 0.12 \cdot \varepsilon \tag{9.21}$$

噪声标准偏差选为 0.12，使得信噪比大约为 5。我们称此为张量积方案；乘积项给出的面看上去类似于图 9.11 中的面。

方案 2　和方案 1 相同，但取总数 $p = 20$ 个预测子；即有 18 个独立于响应的输入。

方案 3　具有神经网络结构：

$$\begin{aligned} \ell_1 &= X_1 + X_2 + X_3 + X_4 + X_5 \\ \ell_2 &= X_6 - X_7 + X_8 - X_9 + X_{10} \\ \sigma(t) &= 1/(1 + e^{-t}) \\ Y &= \sigma(\ell_1) + \sigma(\ell_2) + 0.12 \cdot \varepsilon \end{aligned} \tag{9.22}$$

方案 1 和方案 2 对于 MARS 很理想，而方案 3 包含高阶交互作用，并且可能很难用 MARS 近似。我们对每个模型进行了 5 次模拟，并记录了结果。

在方案 1 中，MARS 近乎完美地揭示了正确的模型。在方案 2 中，MARS 建立了正确的结构，也发现了几个涉及其他预测子的额外项。

令 $\mu(x)$ 是 Y 的真实均值，并令

$$\begin{aligned} \text{MSE}_0 &= \text{ave}_{x \in \text{Test}} (\bar{y} - \mu(x))^2, \\ \text{MSE} &= \text{ave}_{x \in \text{Test}} (\hat{f}(x) - \mu(x))^2 \end{aligned} \tag{9.23}$$

它们表示常量模型和拟合的 MARS 模型的均方误差，通过取 x 的 1000 个测试值的平均值进行估计。表 9.4 显示了每种方案下，模型中的模型误差或 R^2 的相应降低：

$$R^2 = \frac{\text{MSE}_0 - \text{MSE}}{\text{MSE}_0} \tag{9.24}$$

显示的值是 5 种模拟上的均值和标准误差。MARS 的性能由于方案 2 中包含无用输入略有降低，在方案 3 中明显更差一些。

表 9.4　MARS 应用于三种不同的方案时，在模型误差（R^2）中比率降低

方案	均值(S.E)
1：张量积 $p = 2$	0.97(0.01)
2：张量积 $p = 20$	0.96(0.01)
3：神经网络	0.79(0.01)

9.4.3　其他问题

用于分类的 MARS

可以通过扩充 MARS 方法和算法来处理分类问题。已经提出了一些策略。

对于两个类，可以用 0/1 对输出编码，并把问题作为一个回归问题来处理；我们对垃圾邮件例就是这样做的。对多于两个类的分类，可以使用第 4.2 节介绍的指示器响应方法。我们通过 0/1 指示变量对 K 个响应类编码，然后执行一个多元响应 MARS 回归。对于后者，我们对所有响应变量使用一个公共的基函数集。将样本分到具有极大预测响应值的类。然而，这种

方法也存在第 4.2 节讨论的潜在问题。一种更一般的较好方法是“最优得分”方法，将在第 12.5 节讨论。

Stone 等人(1997)开发了一种 MARS 的混合方法，叫做 PolyMARS，专门用于处理分类问题。它使用第 4.4 节讨论的多元逻辑斯缔框架。像 MARS 一样，它以一种逐步前向的方式增长模型，但在每一步，它使用对多项式对数似然的二次逼近搜索下一个基函数对。一旦找到，就用极大似然拟合扩大的模型，并重复这一过程。

MARS 与 CART 的联系

尽管看起来非常不同，但实际上 MARS 和 CART 策略还是有很强的相似性。假设我们对 MARS 过程做如下改变：

- 用阶梯函数 $I(x-t>0)$和 $I(x-t\leqslant 0)$代替分段线性基函数。
- 当一个模型项被候选项包含在乘积中，它将被交叉项取代，因此在以后的交叉项中不再使用。

对于这些改变，MARS 的前向过程与 CART 的树增长算法相同。一个阶梯函数乘以一个反演阶梯函数等价于在该步分裂一个节点。第二个限制意味着一个节点不会多次分裂，并引出极有吸引力的 CART 模型的二叉树表示。另一方面，正是该限制使得 CART 很难对加法结构建模。MARS 放弃了树型结构，但获得了捕获加法作用的能力。

混合输入

MARS 能够以自然的、与 CART 非常相似的方式处理“混合”预测子——定量和定性的。MARS 考虑将一个定性预测的范畴分割成两组的所有可能的二路划分，每个这样的划分都产生一个分段常量基函数对——两个类集合的指示子函数。对这个基函数对的处理和其他基函数一样，并与其他已经在模型中的基函数一起形成张量积。

9.5 分层专家混合

分层专家混合(hierarchical mixtures of experts，HME)过程可以看做是树方法的一个变形。其主要不同在于树的分裂不是硬性决定的，而是基于相当软的概率方法。在每个节点，一个观测根据它的输入值的概率决定向左还是向右。这种方法具有一些计算方面的优点，因为结果参数优化问题是光滑的，不同于基于树方法的离散分裂点的搜索。软分裂对预测准确性也有所帮助，并能提供有用数据的替代描述。

HME 和 CART 的树实现之间还存在着其他差别。对于 HME 方法，在每一个端节点上用线性(或逻辑斯缔回归)模型拟合，而不是像 CART 那样用一个常量拟合。分裂可以是多路的，而不只是二路；而且分裂是输入的线性组合的概率函数，而不是 CART 标准使用的单输入。然而，这些选择的相对优缺点还不是很清楚，大部分已经在第 9.2 节的末尾处讨论。

一个简单的两层 HME 模型显示在图 9.13 中。可以把它看做是在每个非端节点具有软分裂的树。但是，该方法的发明者使用了一种不同的术语。端节点称做专家(expert)，非端节点称为门控网络(gating networks)。其想法是，每个专家提供一个关于响应的观点(预测)，并且由门控网络将它们组合到一起。如同我们将要看到的，这个模型是一个形式上的混合模型，而且

图中的两层模型可以被扩展到多层，因此取名分层专家混合。

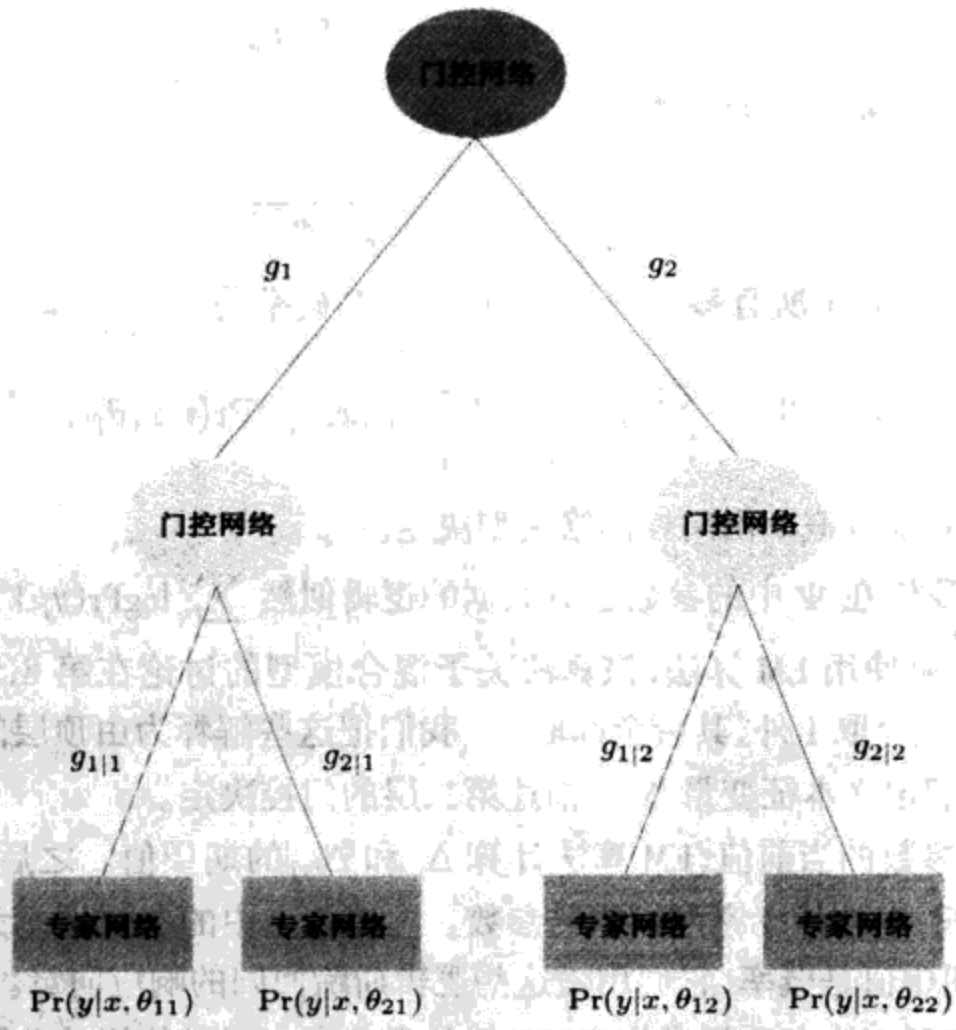


图 9.13　两层分层专家混合(HME)模型

考虑本章前面讨论过的回归或者分类问题。数据形式是(x_i,y_i)，$i=1,2,\cdots,N$，其中y_i是连续或二值的响应，x_i是向量值输入。为了简化记号，我们假设x_i的第一个元素是 1，表示截距。

这里说明如何定义一个 HME。顶端门控网络有如下输出：

$$g_j(x,\gamma_j)=\frac{e^{\gamma_j^T x}}{\sum_{k=1}^{K} e^{\gamma_k^T x}},\ j=1,2,\cdots,K \tag{9.25}$$

其中，每个γ_j是一个未知参数向量。这表示一个软 K 路分裂(在图 9.13 中，$K=2$)。每个$g_j(x,\gamma_j)$是把具有特征向量 x 的观测分配到第j 个分支的概率。注意，对于 $K=2$，如果取 x 的一个元素的系数为 $+\infty$，则得到一个具有无限斜率的对数曲线。在这种情况下，门控概率或者为 0，或者为 1，对应于那个输入上的一个硬分裂。

在第二层，门控网络具有类似的形式：

$$g_{\ell|j}(x,\gamma_{j\ell})=\frac{e^{\gamma_{j\ell}^T x}}{\sum_{k=1}^{K} e^{\gamma_{jk}^T x}},\ \ell=1,2,\cdots,K \tag{9.26}$$

这是在上一层已分配第 j 个分支的情况下，分配到第 ℓ 个分支的概率。

在每个专家(端节点)上，我们有如下形式的关于响应变量的模型：

$$Y\sim\Pr(y|x,\theta_{j\ell}) \tag{9.27}$$

该公式因问题而异。

回归：以 $\theta_{j\ell} = (\beta_{j\ell}, \sigma^2_{j\ell})$，使用高斯线性回归模型：

$$Y = \beta^T_{j\ell} x + \varepsilon \quad 且 \quad \varepsilon \sim N(0, \sigma^2_{j\ell}) \tag{9.28}$$

分类：使用线性逻辑斯缔回归模型：

$$\Pr(Y = 1|x, \theta_{j\ell}) = \frac{1}{1 + e^{-\theta^T_{j\ell} x}} \tag{9.29}$$

用 $\Psi = \{\gamma_j, \gamma_{j\ell}, \theta_{j\ell}\}$ 表示所有参数的集合，$Y = y$ 的总概率是：

$$\Pr(y|x, \Psi) = \sum_{j=1}^{K} g_j(x, \gamma_j) \sum_{\ell=1}^{K} g_{\ell|j}(x, \gamma_{j\ell}) \Pr(y|x, \theta_{j\ell}) \tag{9.30}$$

这是一个混合模型，其混合概率由门控网络模型决定。

为了估计参数，我们在 Ψ 中的参数上对数据的逻辑似然 $\sum_i \log \Pr(y_i | x_i, \Psi)$ 极大化。做这件事最方便的方法是使用 EM 算法，该算法关于混合模型的讨论在第 8.5 节。我们定义本征变量 Δ_j，它们除了一个取 1 外，其余全部取 0。我们把这些解释为由顶层门控网络所做的分支决定。类似地，我们定义本征变量 $\Delta_{\ell|j}$，描述第二层的门控决定。

在 E 步中，给定参数的当前值，EM 算法计算 Δ_j 和 $\Delta_{\ell|j}$ 的期望值。之后，这些期望值用做过程的 M 步的观测权，以估计专家网络中的参数。内部节点中的参数用多元逻辑斯缔回归来估计。Δ_j 或 $\Delta_{\ell|j}$ 的期望值是概率，它们用做这些逻辑斯缔回归的响应向量。

分层专家混合模型方法有希望成为 CART 树的竞争者。通过使用软分裂而不是硬决定规则，可以很好地处理响应从低到高渐变的情况。对数似然是未知权值的光滑函数，因此易于进行数值优化。该模型类似于使用线性组合分裂的 CART，但后者的优化比较困难。另一方面，就我们所知，对于 HME 模型，还没有找到一个像 CART 那样好的树拓扑结构。典型的方法是，使用一定深度的固定的树，它可能是 CART 过程的输出。HME 研究的重点一直是预测，而不是最终模型的解释。HME 的一个近亲是本征类模型（Lin 等人，2000）。该模型通常只有一层，其中，节点或本征类被解释为显示相似响应行为的主题的组。

9.6 遗漏数据

对于一个或多个输入特征，观测中出现遗漏值的现象非常普遍。通常的解决方法是以某种方式填补遗漏的值。

然而，处理遗漏值的首要问题是确定遗漏数据是否使观测数据失真。粗略地说，如果导致遗漏的机制独立于它的（没观测到的）值，则数据遗漏是随机的。一个更精确的定义由 Little 和 Rubin(1987)给出。假设 $\mathbf{y}$ 是响应向量，$\mathbf{X}$ 是 $N \times p$ 的输入矩阵（其中有些数据遗漏了）。用 $\mathbf{X}_{obs}$ 表示 $\mathbf{X}$ 中观测到的项值，并令 $\mathbf{Z} = (\mathbf{y}, \mathbf{X})$，$\mathbf{Z}_{obs} = (\mathbf{y}, \mathbf{X}_{obs})$。最后，如果 $\mathbf{R}$ 是一个指示子矩阵，当 x_{ij} 遗漏时，其第 ij 个元素的值取 1，其余情况取 0。数据是随机遗漏的（missing at random，MAR），如果 $\mathbf{R}$ 的分布只通过 $\mathbf{Z}_{obs}$ 依赖数据 $\mathbf{Z}$：

$$\Pr(\mathbf{R}|\mathbf{Z}, \theta) = \Pr(\mathbf{R}|\mathbf{Z}_{obs}, \theta) \tag{9.31}$$

这里，θ 是 $\mathbf{R}$ 的分布中的任意参数。数据是完全随机遗漏的（missing completely at random，

MCAR)，如果 **R** 的分布不依赖观测到的或遗漏的数据：

$$\Pr(\mathbf{R}|\mathbf{Z},\theta)=\Pr(\mathbf{R}|\theta) \tag{9.32}$$

MCAR 的假设比 MAR 更强：为了有效性，大多数估计方法依赖于 MCAR。

例如，由于医生认为病人太虚弱而没对病人做检查，那么该观测既不是 MAR 的也不是 MCAR 的。在这种情况下，遗漏数据导致我们的观测训练数据产生真实总体的失真描述，并且在此情况下，数据估计是很危险的。通常，一个特征是否是 MCAR 的必须通过数据收集过程的有关信息来进行判断。对于分类的特征，处理该问题的一种方法是把遗漏数据用一个附加的类编码。然后对训练数据拟合我们的模型，并观察遗漏类是否是响应的预测。

假设特征是完全随机遗漏的，这里有几种处理方法：

1. 丢弃具有遗漏值的任何观测。
2. 依赖学习算法在训练阶段处理遗漏值。
3. 在训练之前估计所有的遗漏值。

如果遗漏数据量相对较小，方法(1)可以使用，否则不能用。至于方法(2)，CART 是一种通过代理分裂(surrogate split)有效处理遗漏值的学习算法。MARS和PRIM使用类似的方法。在广义加法模型中，当反向拟合算法对某输入特征的部分余差进行光滑时，该输入特征上的遗漏值的所有观测都将被忽略，并且把它们的拟合值置成 0。由于拟合曲线有均值 0(当模型包含截距时)，所以，这相当于把平均拟合值赋给遗漏的观测。

对于大多数学习方法，估计方法(3)是必要的。最简单的策略是用该特征的非遗漏值的均值或中值估计遗漏值(注意，上面的广义加法模型的过程与此相似)。

如果特征总是存在一定程度的依赖性，则我们可以做得更好些。对于每个特征，给定其他特征，可以为该特征建立一个预测模型，然后用该模型的预测来估计每个遗漏值。我们必须记住，在为估计特征选择学习方法时，选择的方法不同于由 **X** 预测 **y** 的方法。这样，即使最终目标是在 **X** 上求 **y** 的线性回归，灵活的自适应方法通常也是可取的。另外，如果训练集中有许多遗漏的特征值，那么，学习方法本身就必须能够处理遗漏的特征值。因此，对于这种“引擎”，CART 是最理想的选择。

在估计遗漏值之后，通常把遗漏数据看做实际观测数据处理。这忽略了由于遗漏数据估计而引起的不确定性，这种不确定性本身也会把附加的不确定性从响应模型带到估计和预测中。我们可以通过多元估算并为此建立许多不同的训练集来度量这种附加的不确定性。**y** 的预测模型可以拟合每个训练集，并且可以评估训练集上的变差。如果 CART 用于估算引擎，则可以通过从对应端节点的值中抽样做多元估算。

9.7　计算考虑

对于 N 个观测和 p 个预测子，加法模型拟合需要使用一维光滑或回归方法 mp 次。反拟合算法需要的循环次数 m 一般小于 20，甚至小于 10，这取决于输入的相关程度。例如，对于三次光滑样条，初始分类需要 $N\log N$ 次操作，而样条拟合需要 N 次操作。因而加法模型拟合的总操作次数是 $pN\log N+mpN$。

树方法需要 $pN\log N$ 次操作来对每个预测子初始分类，另外 $pN\log N$ 次操作用于分裂计

算。如果分裂发生在预测区域的边缘，则操作次数可能上升到 N^2p。

MARS需要 $Nm^2 + pmN$ 次操作，从 p 个预测子的集合中，把一个基函数添加到已经有 m 个项的模型中。因此，构造一个具有 M 个项的模型需要 $NM^3 + pM^2N$ 次计算。如果 M 是 N 的一个合理部分，那么，这种计算可能会是非常惊人的。

通常，在每个 M 步上拟合一个 HME 分量的开销不大：该回归需要 Np^2 次操作，K 类逻辑斯缔回归需要 Np^2K^2 次操作。然而，EM 算法收敛却需要很长的时间，大型 HME 模型拟合的代价也是相当大的。

文献注释

广义加法模型最全面的资料是Hastie和Tibshirani(1990)的著作。这项工作在医疗问题中的不同应用在Hastie 等人(1989)、Hastie和Herman(1990)的论文中进行了讨论，使用Splus的软件实现在Chambers和Hastie(1991)中介绍。Green和Silverman(1994)讨论了多种框架下的罚和样条模型。Efron 和Tibshirani(1991)为非数学背景的读者介绍了统计学(包括广义加法模型)的现代进展。分类和回归树的提出至少可以追溯到 Morgan 和 Sonquist(1963)。我们使用了Breiman等人(1984)和 Quinlan(1993)的现代方法。PRIM 方法由Friedman和 Fisher(1999)提出，而 MARS 方法由Friedman(1991)提出，加法模型的先驱出现在 Friedman 和Silverman(1989)中。分层专家混合模型在 Jordan 和Jacobs(1994)中提出；还可以在 Jacobs 等人(1991)的论文中看到。

习题

9.1 证明：y_i 对 x_i 的光滑样条拟合保留了拟合的线性部分。换言之，如果$y_i = \hat{y}_i + r_i$，其中，$\hat{y}_i$ 表示线性回归拟合，$\mathbf{S}$ 是光滑矩阵，则 $\mathbf{Sy} = \hat{\mathbf{y}} + \mathbf{Sr}$。证明：上述结论对局部线性回归(见第 6.1.1 节)也成立。由此讨论算法 9.1 中步骤 2 第二行的调整步骤是不必要的。

9.2 设 $\mathbf{A}$ 是一个已知的 $k \times k$ 矩阵，$\mathbf{b}$ 是已知的 k 向量，$\mathbf{z}$ 是未知的 k 向量。高斯－塞德尔算法求解线性方程组 $\mathbf{Az} = \mathbf{b}$ 的方法如下：相继固定当前推测中其他的 z_j，求解第 j 个方程中的元素 z_j。对 $j = 1,2,\cdots,k,1,2,\cdots,k,\cdots$，重复该过程，直至收敛为止(Golub 和 Van Loan，1983)。

(a) 考虑有 N 个观测和 p 个项的加法模型，第 j 个项被线性光滑子 $\mathbf{S}_j$ 拟合。考虑如下方程组：

$$\begin{pmatrix} \mathbf{I} & \mathbf{S}_1 & \mathbf{S}_1 & \cdots & \mathbf{S}_1 \\ \mathbf{S}_2 & \mathbf{I} & \mathbf{S}_2 & \cdots & \mathbf{S}_2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{S}_p & \mathbf{S}_p & \mathbf{S}_p & \cdots & \mathbf{I} \end{pmatrix} \begin{pmatrix} \mathbf{f}_1 \\ \mathbf{f}_2 \\ \vdots \\ \mathbf{f}_p \end{pmatrix} = \begin{pmatrix} \mathbf{S}_1\mathbf{y} \\ \mathbf{S}_2\mathbf{y} \\ \vdots \\ \mathbf{S}_p\mathbf{y} \end{pmatrix} \tag{9.33}$$

这里，$\mathbf{f}_j$ 是数据点上第 j 个函数的求值 N 向量，$\mathbf{y}$ 是响应值的 N 向量。证明：反拟合是求解该方程组的分块高斯－塞德尔算法。

(b) 如果矩阵 $\mathbf{A}$ 是正定的，则高斯－塞德尔过程收敛。例如，对于 $p = 2$ 的简单情形，假设每个 $\mathbf{S}_j$ 是对称的，其本征值在$[0,1)$上，证明反拟合算法收敛(下一个习题要求直

接证明)。

9.3 两个项的反拟合。设 $\mathbf{S}_1$ 和 $\mathbf{S}_2$ 是对称光滑算子(矩阵),其本征值在[0, 1)上。考虑响应向量 $\mathbf{y}$ 和光滑子 $\mathbf{S}_1$ 和 $\mathbf{S}_2$ 的反拟合算法。证明:对任意初值,算法收敛并产生关于最终迭代的公式。

9.4 反拟合方程。考虑一个正交投影的反拟合过程,令 $\mathbf{D}$ 是总回归矩阵,它的列生成 $V = \mathcal{L}_{\text{col}}(\mathbf{S}_1) \oplus \mathcal{L}_{\text{col}}(S_2) \oplus \cdots, \oplus \mathcal{L}_{\text{col}}(\mathbf{S}_p)$,其中 $\mathcal{L}_{\text{col}}(\mathbf{S})$ 表示矩阵 S 的列空间。证明估计方程:

$$\begin{pmatrix} \mathbf{I} & \mathbf{S}_1 & \mathbf{S}_1 & \cdots & \mathbf{S}_1 \\ \mathbf{S}_2 & \mathbf{I} & \mathbf{S}_2 & \cdots & \mathbf{S}_2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{S}_p & \mathbf{S}_p & \mathbf{S}_p & \cdots & \mathbf{I} \end{pmatrix} \begin{pmatrix} \mathbf{f}_1 \\ \mathbf{f}_2 \\ \vdots \\ \mathbf{f}_p \end{pmatrix} = \begin{pmatrix} \mathbf{S}_1\mathbf{y} \\ \mathbf{S}_2\mathbf{y} \\ \vdots \\ \mathbf{S}_p\mathbf{y} \end{pmatrix}$$

等价于最小二乘方正规方程 $\mathbf{D}^T\mathbf{D}\beta = \mathbf{D}^T\mathbf{y}$,其中 β 是系数向量。

9.5 假设相同的光滑子 $\mathbf{S}$ 用于估计二项加法模型中的两个项(即两个变量都是恒等的)。假定 $\mathbf{S}$ 是对称的,其本征值在[0,1)上。证明反拟合残差收敛于 $(\mathbf{I} + \mathbf{S})^{-1}(\mathbf{I} - \mathbf{S})\mathbf{y}$,并且残差平方和向上收敛。在结构化较差的情形下,残差平方和能够向上收敛吗?该拟合与 $\mathbf{S}$ 拟合单个项的拟合相比,情况如何?(提示:使用 $\mathbf{S}$ 的本征分解帮助比较。)

9.6 树的自由度。给定具有均值 $f(x_i)$ 和方差 σ^2 的数据 y_j,及拟合算子 $\mathbf{y} \to \hat{\mathbf{y}}$,定义拟合的自由度为 $\sum_i \text{Var}(\hat{y}_i)/\sigma^2$。

考虑一个由回归树估计的拟合 $\hat{\mathbf{y}}$,对一个预测子的集合 $X_1, X_2, \cdots, X_p$ 进行拟合。

(a) 用端节点的个数 m,给出拟合自由度的粗略公式。

(b) 产生 100 个以 $X_1, X_2, \cdots, X_{10}$ 为预测子的观测作为独立的标准高斯变量并固定这些值。

(c) 产生响应值也作为标准高斯响应($\sigma^2 = 1$),它们独立于预测子。用 1,5 和 10 个端节点的回归树拟合数据,并据此估计每个拟合的自由度。(做 10 次响应模拟并对结果求平均,以获得自由度较好的估计。)

(d) 比较在(a)和(c)中对自由度的估计,并讨论之。

(e) 如果回归树拟合是线性操作,如对某矩阵 $\mathbf{S}$,我们有 $\hat{\mathbf{y}} = \mathbf{S}\mathbf{y}$。那么自由度将是 $(1/\sigma^2)\text{tr}(\mathbf{S})$。提出一种方法,计算回归树的逼进 $\mathbf{S}$ 矩阵,计算它并将结果自由度与(a)和(c)中的结果做比较。

9.7 考虑图 6.9 中的臭氧数据。

(a) 用加法模型拟合臭氧浓度的三次方根,它是温度、风速和放射性的函数。将你的结果与图 6.9 显示的结果进行比较。

(b) 用树、MARS 和 PRIM 拟合相同的数据,并将其结果与(a)和图 6.9 中的结果进行比较。

第 10 章　提升和加法树

10.1　提升方法

提升是近十年来提出的最有效的学习思想之一。它最初是为分类问题而设计的，但正像在本章中将要看到的，也可以对它进行扩充以解决回归问题。提升方法的动机是合并许多“弱”分类器的输出以产生有效“委员会”的过程。从这一角度看，提升与装袋以及其他基于委员会的方法（见第 8.8 节）具有相似之处。然而，我们将看到联系是极其表面的，提升在本质上是不同的。

我们从最流行的提升算法开始，介绍由 Freund 和 Schapire（1997）提出的名为“AdaBoost.M1”的算法。考虑一个 2-类问题，输出变量编码为 $Y\in\{-1, 1\}$。给定预测子变量的向量 X，分类器 $G(X)$产生在$\{-1, 1\}$中取值的预测。训练样本上的误差率是：

$$\overline{\text{err}} = \frac{1}{N}\sum_{i=1}^{N} I(y_i \neq G(x_i))$$

而未来预测上的期望误差率是 $E_{XY}I(Y\neq G(X))$。

一个弱分类器的误差率只比随机猜测略好一些。提升的目的就是连续对反复修改的数据应用弱分类算法，由此产生一个弱分类器序列 $G_m(x)$，$m=1, 2, \cdots, M$。然后，通过一个加权的多数表决来合并全部预测，以产生最终预测：

$$G(x) = \text{sign}\left(\sum_{m=1}^{M}\alpha_m G_m(x)\right) \tag{10.1}$$

这里，$\alpha_1, \alpha_2, \cdots, \alpha_M$ 由提升算法来计算，并对每个 $G_m(x)$的贡献加权。它们的作用是对序列中较精确的分类器给予较高的影响。图 10.1 给出了 AdaBoost 过程的示意图。

在每个提升步，数据修改就是对每一训练观测$(x_i, y_i)$$(i=1,2,\cdots,N)$实施加权 $w_1, w_2, \cdots, w_N$。开始，所有权都置成 $w_i = 1/N$，以便第一步能够简单地用通常的方式在数据上训练分类器。对每一个相继的迭代 $m=2, 3, \cdots, M$，观测的权值被分别修改，并且分类算法被再次应用于加权观测。在第 m 步，那些被前一步导出的分类器 $G_{m-1}(x)$误分类的观测权值提高，而被正确分类的观测权值降低。这样，随着迭代的进行，那些很难正确分类的观测受到了不断增长的影响。因此，每个后继分类器被强制关注被前面的分类器误分类的训练观测。

算法 10.1 给出了 AdaBoost.M1 算法的细节。步骤 2(a)在加权观测上导出当前分类器 $G_m(x)$。在步骤 2(b)，计算结果加权误差率。步骤 2(c)计算权值α_m，该权值在产生最终分类器$G(x)$时(步骤 3)赋予 $G_m(x)$。在步骤 2(d)，为下一步迭代更新每个观测的权值。被 $G_m(x)$误分类的观测的权值被放大一个因子 $\exp(\alpha_m)$，增加了对导出序列中下一个分类器 $G_{m+1}(x)$的相对影响。

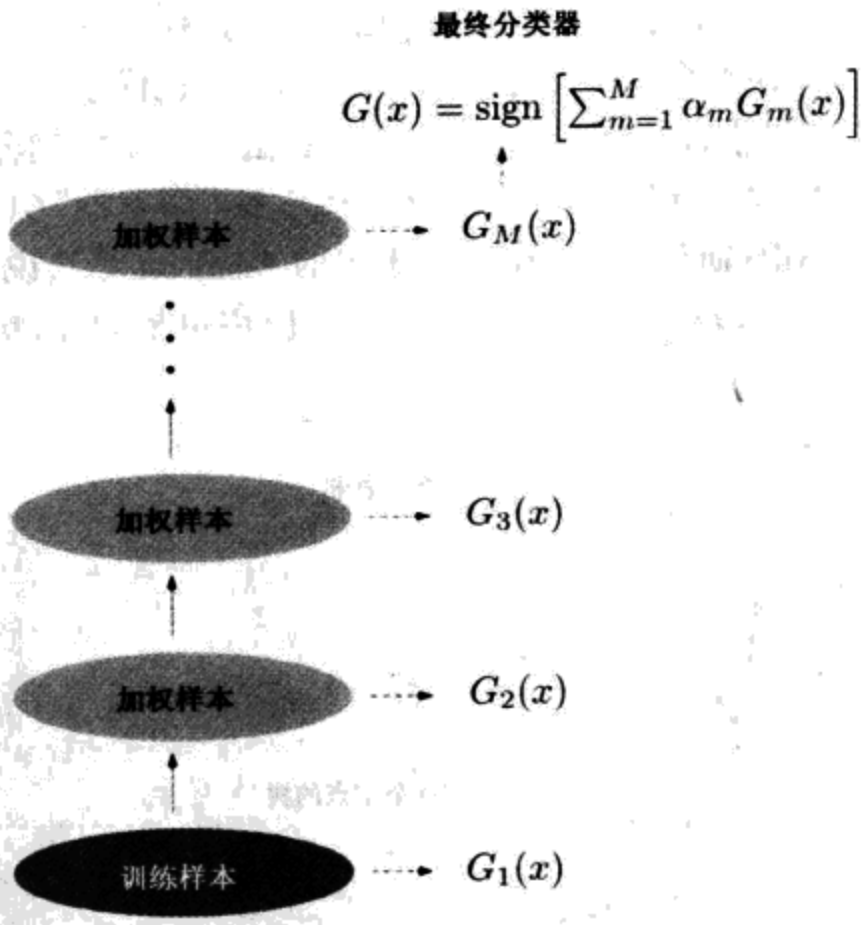


图 10.1　AdaBoost 示意图。在数据集的加权版本上训练分类器，然后合并分类器以产生最终预测

算法 10.1　AdaBoost.M1

1. 初始化观测权值 $w_i = 1/N, i = 1, 2, \cdots, N$
2. 对于 $m = 1$ 到 M：
   (a) 使用权值 w_i，用分类器 $G_m(x)$ 拟合训练数据
   (b) 计算
   $$\text{err}_m = \frac{\sum_{i=1}^{N} w_i I(y_i \neq G_m(x_i))}{\sum_{i=1}^{N} w_i}$$
   (c) 计算 $\alpha_m = \log((1 - \text{err}_m)/\text{err}_m)$
   (d) 置 $w_i \leftarrow w_i \cdot \exp[\alpha_m \cdot I(y_i \neq G_m(x_i))], i = 1, 2, \cdots, N$
3. 输出 $G(x) = \text{sign}\left[\sum_{m=1}^{M} \alpha_m G_m(x)\right]$

在 Friedman 等人(2000)的论文中，AdaBoost.M1 算法称做“离散的 AdaBoost”，因为基本分类器 $G_m(x)$返回离散的类标号。如果基本分类器改为返回实数值预测(例如，映射到区间$[-1, 1]$的概率)，则可以对 AdaBoost 算法做恰当修改[参见 Friedman 等(2000)中的“Real AdaBoost”]。

AdaBoost 算法能够显著提升很弱的分类器的性能，如图 10.2 所示。特征 $X_1, X_2, \cdots, X_{10}$ 是标准的独立高斯分布，而确定性的目标 Y 由下式定义：

$$Y = \begin{cases} 1 & \text{如果} \sum X_j^2 > \chi_{10}^2(0.5) \\ -1 & \text{否则} \end{cases} \tag{10.2}$$

这里 $\chi_{10}^2(0.5) = 9.34$ 是具有 10 个自由度的 χ^2 随机变量的中值(10 个标准高斯的平方和)。这里有 2000 个训练实例，每个类中大约 1000 个实例，以及 10 000 个检验观测。这里的弱分类器仅仅是个“树桩”：两个端节点的分类树。对训练数据集单独应用这个分类器产生非常差的检验集误差率 46%；相比之下，随机猜测的误差率为 50%。然而，随着提升迭代过程的进行，

误差率平稳降低，在 400 次迭代之后误差率降到 12.2%。这样，通过提升这个简单而且非常弱的分类器可以将它的预测误差率大约降低到原来的 1/4。它也优于单一的大型分类树(误差率是 26%)。自从它被引入以来，许多资料对 AdaBoost 算法在产生精确分类方面的成功给出了详细解释。这方面的工作大部分集中在把分类树用做“基本学习器”$G(x)$，其中的改进是很神奇的。事实上，Breiman(NIPS 研讨会，1996)把基于树的 AdaBoost 算法说成是“世界上最好的现货供应分类器”[见 Breiman(1998)]。在数据挖掘中的应用更是如此，更详细的讨论将在本章后面的第 10.7 节中进行。

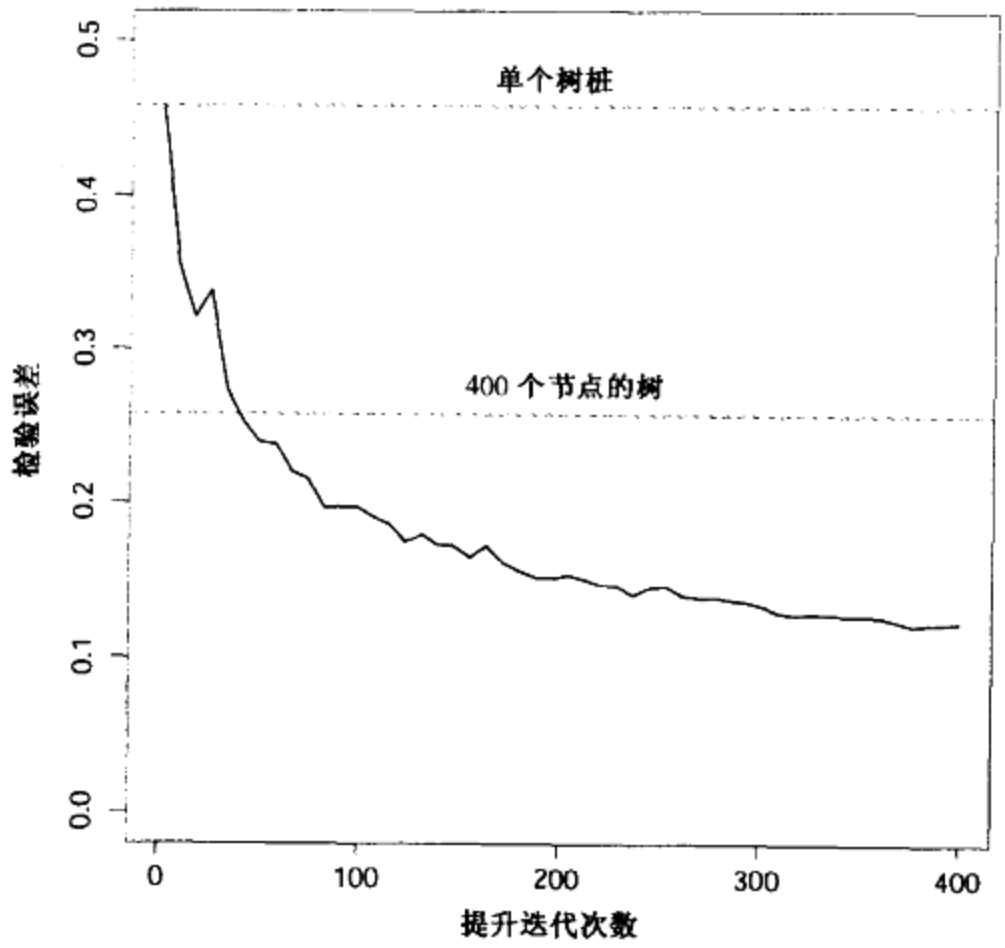


图 10.2 模拟数据(10.2)：树桩的提升检验误差率是迭代次数的函数。也显示了一个单树桩和400个节点的分类树的检验误差率

10.1.1 本章概述

下面是本章的概述：

- 展示 AdaBoost 拟合一个基本学习器的加法模型，优化一个新颖的指数损失函数。该损失函数与(负的)二项式对数似然非常相似(见第 10.2 节到第 10.4 节)。
- 证明指数损失函数的总体极小是类概率的对数几率(见第 10.5 节)。
- 介绍比平方误差或者指数损失具有更强健壮性的回归和分类损失函数(见第 10.6 节)。
- 证明对于提升算法在数据挖掘中的应用来说，决策树是一个理想的基本学习器(见第 10.7 节和第 10.9 节)。
- 使用梯度方法，为具有任意损失函数的提升树开发一类技术(“MART”)(见第 10.10 节)。
- 强调“慢学习”的重要性，并通过收缩进入模型的每个新项实现它(见第 10.12 节)。
- 描绘前向分步收缩与模型参数的 L_1 罚(“套索”)之间的联系。列举理由说明 L_1 罚可能比支持向量机模型使用的 L_2 罚更优越(见第 10.12.2 节)。
- 描述拟合模型的解释工具(见第 10.13 节)。

10.2　提升拟合加法模型

提升算法的成功其实并不很神秘。它的关键在于式(10.1)。提升是一种拟合基本"基"函数集上的加法展开式的方法。这里，每个基函数都是分类器 $G_m(x) \in \{-1, 1\}$。更一般地，基函数展开式取如下形式：

$$f(x) = \sum_{m=1}^{M} \beta_m b(x; \gamma_m) \tag{10.3}$$

其中，$\beta_m (m = 1, 2, \cdots, M)$是展开式系数，$b(x; \gamma) \in \mathrm{IR}$ 通常是多元变量 x 的简单函数，由参数 γ 的集合刻画。我们曾在第 5 章详细讨论过基展开。

像这样的加法展开式构成了本书涵盖的许多学习技术的基础：

- 在单隐藏层神经网络(见第 11 章)中，$b(x; \gamma) = \sigma(\gamma_0 + \gamma_1^{\mathrm{t}} x)$，其中 $\sigma(\cdot)$是一个 S 型函数，而 γ 对输入变量的一个线性组合参数化。
- 在信号处理过程中，小波(见第 5.9.1 节)是一种流行的选择，γ 对"母"小波的位置和标度位移参数化。
- MARS(见第 9.4 节)使用截尾幂样条基函数，其中 γ 对变量和纽结的值参数化。
- 对于树，γ 对分裂变量、内部节点上的分裂点和端节点的预测进行参数化。

通常，通过极小化对训练数据取平均的损失函数来拟合这些模型，如平方误差或者基于似然损失函数，

$$\min_{\{\beta_m, \gamma_m\}_1^M} \sum_{i=1}^{N} L\left(y_i, \sum_{m=1}^{M} \beta_m b(x_i; \gamma_m)\right) \tag{10.4}$$

对许多损失函数 $L(y, f(x))$或基函数 $b(x; \gamma)$，都需要计算密集的数值优化技术。然而，当仅拟合单个基函数的子问题的快速求解可行时，通常能够找到一个较简单的方案，

$$\min_{\beta, \gamma} \sum_{i=1}^{N} L(y_i, \beta b(x_i; \gamma)) \tag{10.5}$$

10.3　前向分步加法建模

通过相继添加新的基函数到展开式，而不调整已添加的参数和系数，前向分步加法建模逼近解式(10.4)。前向分步加法建模在算法 10.2 中给出。在第 m 次迭代，我们求解最优基函数 $b(x; \gamma_m)$和相应的系数 β_m，并添加到当前展开式 $f_{m-1}(x)$中。这样可以产生 $f_m(x)$，并重复这个过程。先前添加的项并不改变。

对于平方误差损失：

$$L(y, f(x)) = (y - f(x))^2 \tag{10.6}$$

我们有：

$$\begin{aligned} L(y_i, f_{m-1}(x_i) + \beta b(x_i; \gamma)) &= (y_i - f_{m-1}(x_i) - \beta b(x_i; \gamma))^2 \\ &= (r_{im} - \beta b(x_i; \gamma))^2 \end{aligned} \tag{10.7}$$

其中，$r_{im} = y_i - f_{m-1}(x_i)$是当前模型在第 i 个观测上的误差(残差)。这样，对于平方误差损失，每一步把对当前残差拟合最好的项 $\beta_m b(x;\gamma_m)$加到展开式中。这种思想是第 10.10.2 节中讨论的最小二乘方回归提升的基础。然而，正如我们在下一节将要看到的，平方误差损失对于分类通常不是一个好的选择，因此需要考虑其他的损失标准。

算法 10.2 前向分步加法建模

1. 初始化 $f_0(x) = 0$
2. 对于 $m = 1$ 到 M:
   (a) 计算

$$(\beta_m, \gamma_m) = \arg\min_{\beta,\gamma} \sum_{i=1}^{N} L(y_i, f_{m-1}(x_i) + \beta b(x_i;\gamma))$$

   (b) 置 $f_m(x) = f_{m-1}(x) + \beta_m b(x;\gamma_m)$

10.4 指数损失函数和 AdaBoost

现在，我们证明 AdaBoost.M1(见算法 10.1)等价于使用如下损失函数的前向分步加法建模(见算法 10.2)：

$$L(y, f(x)) = \exp(-y f(x)) \tag{10.8}$$

该标准的恰当性将在下一节阐述。

对于 AdaBoost，基函数是单个分类器 $G_m(x) \in \{-1, 1\}$。使用指数损失函数，必须求解：

$$(\beta_m, G_m) = \arg\min_{\beta,G} \sum_{i=1}^{N} \exp[-y_i(f_{m-1}(x_i) + \beta G(x_i))]$$

得到每步要添加的分类器 G_m 和相应的系数 β_m。这可以表示为：

$$(\beta_m, G_m) = \arg\min_{\beta,G} \sum_{i=1}^{N} w_i^{(m)} \exp(-\beta y_i G(x_i)) \tag{10.9}$$

其中 $w_i^{(m)} = \exp(-y_i f_{m-1}(x_i))$。因为每个 $w_i^{(m)}$ 既不依赖 β 也不依赖 $G(x)$，它可以看做是应用于每个观测的权值。该权值依赖于 $f_{m-1}(x_i)$，所以，个体权值随每次迭代 m 而改变。

式(10.9)的解可以通过两步获得。第一步，对任意的 $\beta > 0$，关于 $G_m(x)$，式(10.9)的解是：

$$G_m = \arg\min_{G} \sum_{i=1}^{N} w_i^{(m)} I(y_i \neq G(x_i)) \tag{10.10}$$

它是预测 y 时极小化加权误差率的分类器。将式(10.9)的标准用下式表示，这一点容易明白：

$$e^{-\beta} \cdot \sum_{y_i = G(x_i)} w_i^{(m)} + e^{\beta} \cdot \sum_{y_i \neq G(x_i)} w_i^{(m)}$$

接着，可以把它写做：

$$(e^{\beta} - e^{-\beta}) \cdot \sum_{i=1}^{N} w_i^{(m)} I(y_i \neq G(x_i)) + e^{-\beta} \cdot \sum_{i=1}^{N} w_i^{(m)} \tag{10.11}$$

将这个 G_m 插入式(10.9),并且对 β 求解,可以得到:

$$\beta_m = \frac{1}{2}\log\frac{1-\mathrm{err}_m}{\mathrm{err}_m} \tag{10.12}$$

其中 err_m 是极小加权误差率:

$$\mathrm{err}_m = \frac{\sum_{i=1}^{N} w_i^{(m)} I(y_i \neq G_m(x_i))}{\sum_{i=1}^{N} w_i^{(m)}} \tag{10.13}$$

然后,逼近被更新为:

$$f_m(x) = f_{m-1}(x) + \beta_m G_m(x)$$

它导致下一次迭代的权值是:

$$w_i^{(m+1)} = w_i^{(m)} \cdot e^{-\beta_m y_i G_m(x_i)} \tag{10.14}$$

应用事实 $-y_i G_m(x_i) = 2\cdot I(y_i \neq G_m(x_i)) - 1$,式(10.14)变为:

$$w_i^{(m+1)} = w_i^{(m)} \cdot e^{\alpha_m I(y_i \neq G_m(x_i))} \cdot e^{-\beta_m} \tag{10.15}$$

其中,$\alpha_m = 2\beta_m$ 是 AdaBoost.M1 算法(见算法 10.1)步骤 2(c)中定义的量。式(10.15)中的因子 $e^{-\beta_m}$ 用相同的值乘以所有的权,因此没有作用。这样,式(10.15)等价于算法 10.1 步骤 2(d)。还可以把算法 10.1 的步骤 2(a)看做是求解式(10.10)中极小化问题的方法。因此,我们得出结论:AdaBoost.M1 算法通过前向分步加法建模方法使指数损失标准(10.8)极小化。

对于图 10.2 模拟数据问题(10.2),图 10.3 显示训练集误分类率和平均指数损失。训练集误分类率大约在 250 次迭代后平稳下来,但是指数损失保持递减,因为它对估计类概率的变化更敏感。

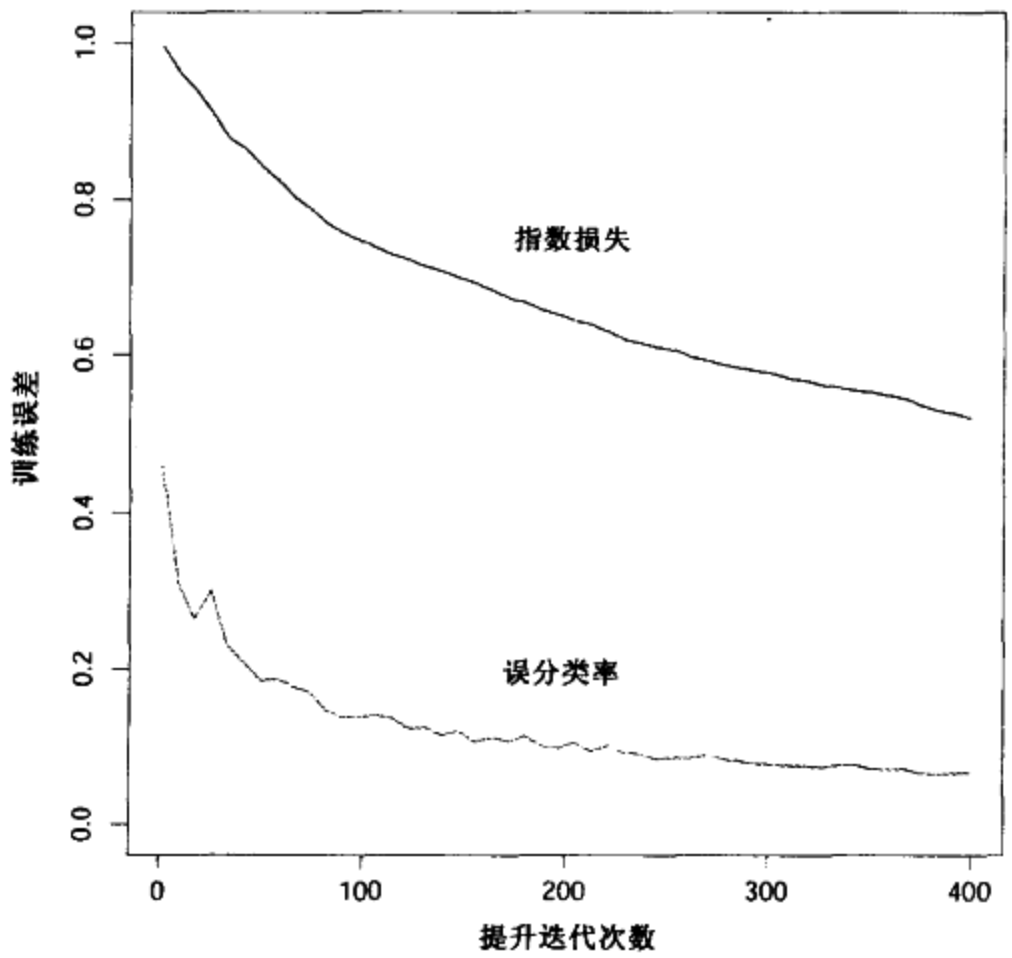


图 10.3　模拟数据,用树桩提升:训练集上的误分类误差和平均指数损失:$(1/N)\sum_{i=1}^{N}\exp(-y_i f(x_i))$

10.5 为什么使用指数损失

AdaBoost.M1 算法最初的动机来自于一个与前一节非常不同的观点。它与基于指数损失的前向分步加法建模方法的等价性只是最近才被发现。通过研究指数损失标准的特性，我们可以洞察过程并发现改进它的方法。

对于加法建模，指数损失的主要吸引力在于计算。它引出简单的模再加权 AdaBoost 算法。然而，了解它的统计特征是有趣的。它估计了什么？它估计得怎么样？第一个问题可以通过寻找它的总体极小来回答。

容易证明(Friedman 等，2000)：

$$f^*(x) = \arg\min_{f(x)} \mathrm{E}_{Y|x}(e^{-Yf(x)}) = \frac{1}{2}\log\frac{\Pr(Y=1|x)}{\Pr(Y=-1|x)} \tag{10.16}$$

或等价地：

$$\Pr(Y=1|x) = \frac{1}{1+e^{-2f^*(x)}}$$

这样，由 AdaBoost 产生的加法展开式对 $P(Y=1|x)$ 的对数几率的一半进行估计。这证实了可以在式(10.1)中使用它的记号作为分类规则。

另一种具有相同总体极小的损失标准是二项式负的对数似然或散离(也称互熵)，它把 f 解释成分对数变换。令

$$p(x) = \Pr(Y=1|x) = \frac{e^{f(x)}}{e^{-f(x)}+e^{f(x)}} = \frac{1}{1+e^{-2f(x)}} \tag{10.17}$$

并且定义 $Y' = (Y+1)/2 \in \{0,1\}$。则二项式对数似然损失函数是：

$$l(Y,p(x)) = Y'\log p(x) + (1-Y')\log(1-p(x))$$

或者等价的散离是：

$$-l(Y,f(x)) = \log\left(1+e^{-2Yf(x)}\right) \tag{10.18}$$

由于对数似然的总体极大是在真实概率 $p(x)=\Pr(Y=1|x)$ 上，从式(10.17)可以看到 $\mathrm{E}_{Y|x}[-l(Y,f(x))]$ 和 $\mathrm{E}_{Y|x}[e^{-Yf(x)}]$ 的总体极小是相同的。这样，使用两个标准中的任意一个在总体级都导致相同的解。注意，e^{-yf} 本身不是一个恰当的对数似然，因为它不是 $Y\in\{-1,1\}$ 的任何概率质量函数的对数。

10.6 损失函数和健壮性

本节，我们更深入考察分类和回归的不同损失函数，并且根据它们对极端数据的健壮性来刻画它们。

分类的健壮损失函数

尽管应用于总体联合分布时指数(10.8)和二项式散离(10.18)产生相同的解，但对于有穷

数据集就不相同了。两个标准都是“边缘”$yf(x)$的单调减函数。在分类(响应是 -1/1)中,边缘起着与回归中的残差$y-f(x)$相似的作用。分类规则 $G(x)=\text{sign}[f(x)]$意味着具有正边缘 $y_i f(x_i)>0$ 的观测被正确地分类,而具有负边缘$y_i f(x_i)<0$ 的观测被误分类。判定边界用$f(x)=0$ 定义。分类算法的目标是尽可能频繁地产生正边缘。任何用于分类的损失标准都应当更多地惩罚负边缘,因为正边缘的观测已被正确分类了。

图 10.4 显示了指数损失(10.8)和二项式散离标准,作为边缘$y\cdot f(x)$的函数。图中还显示了误分类损失 $L(y,f(x))=I(y\cdot f(x)<0)$,它对负边缘值赋予单位罚,而对正边缘值根本不赋予罚。指数和散离损失函数可以看做是对误分类损失的单调连续逼近。它们连续地惩罚递增的负边缘值要比奖励递增的正边缘值多一些。它们之间的区别在于惩罚的程度。对于递增的负边缘值,二项式散离相关的罚呈线性增长,而指数标准对这种观测的影响呈指数增长。

在训练过程中,指数标准主要影响具有大的负边缘值的观测。二项式散离对这样的观测的影响相对较小,并更均匀地对所有数据散布这种影响。因此,在噪声处理中,它的健壮性更强,其中的贝叶斯误差率不接近于 0,特别是在对训练数据中的类标号被误说明的情形中。我们凭经验已经观测到 AdaBoost 的性能在这种情形下有明显的退化。

图中也显示了平方误差损失。总体上相应风险的极小是:

$$f^*(x)=\arg\min_{f(x)}\mathrm{E}_{Y|x}(Y-f(x))^2=\mathrm{E}(Y\mid x)=2\cdot\Pr(Y=1\mid x)-1$$

如前所述,分类规则是 $G(x)=\text{sign}[f(x)]$。对分类误差来说,平方误差损失不是一个好的代用品。正如在图 10.4 中看到的那样,它不是递增边缘$yf(x)$的单调减函数。当边缘值$y_i f(x_i)>1$ 时,它是二次地增加,因此将递增的影响(误差)加到以递增的确定性被正确分类的观测上,从而降低了对被错误分类$y_i f(x_i)<0$ 的观测的相对影响。这样,如果目标是类指派,则单调递减标准就可以作为较好的损失函数。

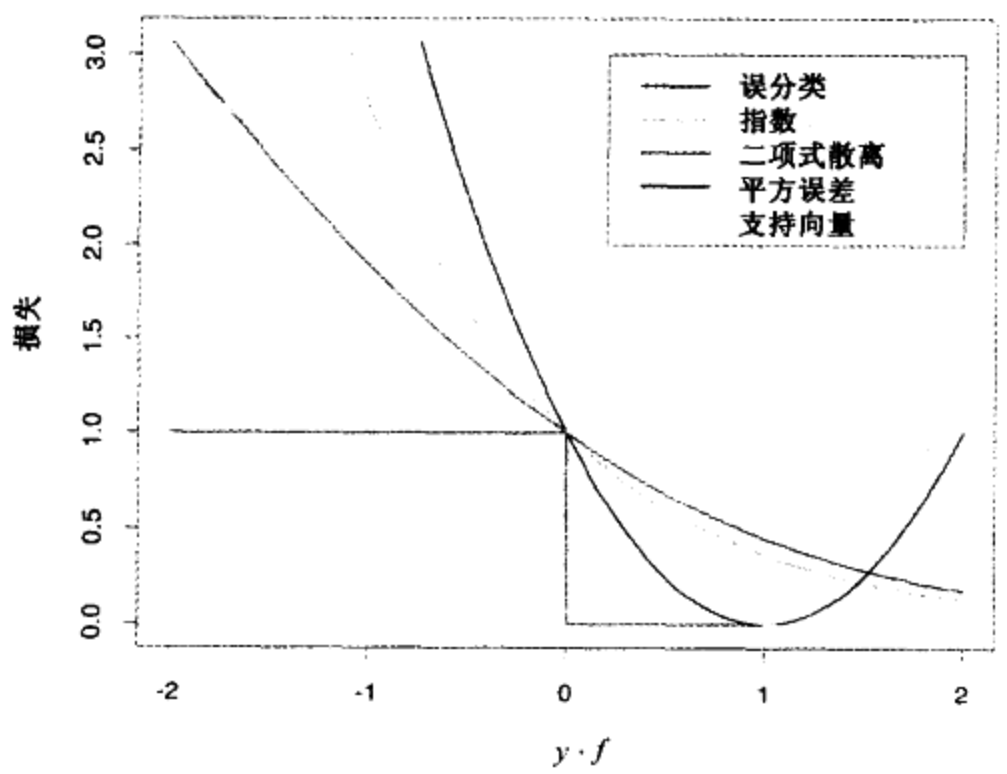


图 10.4　2 - 类分类的损失函数。响应是$y=\pm 1$;预测是f,类预测是$\text{sign}(f)$。损失是误分类:$I(\text{sign}(f)\neq y)$;指数:$\exp(-yf)$;二项式散离:$\log(1+\exp(-2yf))$;平方误差:$(y-f)^2$;支持向量:$(1-yf)\cdot I(yf>1)$(参见第12.3节)。每个函数已被缩放以便经过点(0,1)(见彩页)

对于 K-类分类，响应 Y 在无序集$\mathcal{G}=(\mathcal{G}_1,\cdots,\mathcal{G}_k)$中取值(见第 2.2 节和第 4.4 节)。现在，我们寻找一个在$\mathcal{G}$中取值的分类器 $G(x)$。知道类条件概率 $p_k(x)=\Pr(Y=\mathcal{G}_k\mid x)(k=1,2,\cdots,K)$就足够了，那么贝叶斯分类器是：

$$G(x)=\mathcal{G}_k \quad \text{其中} \quad k=\arg\max_{\ell} p_\ell(x) \tag{10.19}$$

虽然原则上除最大的一个之外，我们不需要学习 $p_k(x)$，但是在数据挖掘应用中，其兴趣通常更多地在于类概率 $p_\ell(x),\ell=1,\cdots,K$ 本身，而不是类指派。如同第 4.4 节，逻辑斯缔模型自然地拓广到 K 类：

$$p_k(x)=\frac{e^{f_k(x)}}{\sum_{l=1}^{K} e^{f_l(x)}} \tag{10.20}$$

它确保$0\leqslant p_k(x)\leqslant 1$，且诸 p_k 的和为 1。注意，我们有 K 个不同的函数，每个类一个。函数 $f_k(x)$ 中有冗余，因为添加任意 $h(x)$到每个叶子中，模型都不变。传统上，将它们中的一个设成 0，例如，和(式 4.17)中一样，$f_K(x)=0$。这里，我们宁愿保留对称性，并且强制约束 $\sum_{k=1}^{K} f_k(x)=0$。二项式散离自然可扩充到 K 类多项式的散离损失函数：

$$\begin{aligned} L(y,p(x)) &= -\sum_{k=1}^{K} I(y=\mathcal{G}_k)\log p_k(x) \\ &= -\sum_{k=1}^{K} I(y=\mathcal{G}_k) f_k(x)+\log\left(\sum_{\ell=1}^{K} e^{f_\ell(x)}\right) \end{aligned} \tag{10.21}$$

与两类情况一样，标准(10.21)根据它们的错误程度线性地惩罚错误预测。我们知道指数标准没有自然的 K 类泛化。

回归的健壮损失函数

类似于指数损失与二项式对数似然之间的联系，对于回归问题，可以考虑平方误差损失 $L(y,f(x))=(y-f(x))^2$ 和绝对损失 $L(y,f(x))=|y-f(x)|$之间的联系。平方误差损失的的总体解是 $f(x)=\mathrm{E}(Y\mid x)$，绝对值损失的解是 $f(x)=\mathrm{median}(Y\mid x)$；对于对称的误差分布，它们相同。然而，在拟合过程中，有限样本上的平方误差损失更重视具有较大绝对残差 $|y_i-f(x_i)|$的观测。这样，它极其缺乏健壮性，而且对于长尾误差分布，特别是对于严重的误度量y值("孤立点")，它的性能会大幅度下降。其他健壮性更强的标准，如绝对损失，在这种情况下性能要好得多。在统计学关于健壮性的文献中，提出了许多不同的回归损失标准，它们对孤立点有很强的抵抗力(如果没有绝对免疫性的话)，而其效果接近高斯误差的最小二乘方。它们常常比具有中度长尾的误差分布好一些。这样的标准之一是用于 M 回归的 Huber 损失标准(Huber，1964)：

$$L(y,f(x))=\begin{cases} [y-f(x)]^2, & \text{对于}|y-f(x)|\leqslant\delta \\ \delta(|y-f(x)|-\delta/2) & \text{其他} \end{cases} \tag{10.22}$$

图 10.5 对这三种损失函数进行了比较。

这样，回归中的绝对损失与分类中的二项式散离相似：对于极端的边缘，它呈线性增长。指数损失比平方误差损失更严重，补偿是指数的而不是二次的。

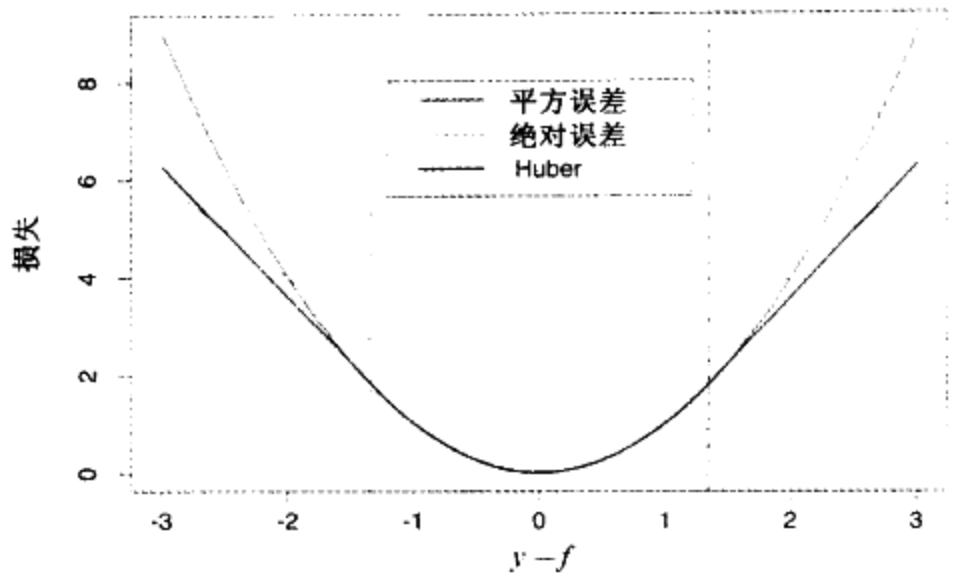


图 10.5　回归所用的三种损失函数的比较，绘制的曲线是$y-f$的函数。Huber 损失函数结合了在 0 附近的平方误差损失和在$|y-f|$较大时的绝对误差损失的好性质(见彩页)

这些考虑表明，当需要关注健壮性时，如数据挖掘应用(见第 10.7 节)，从统计学观点，回归的平方误差损失和分类的指数损失不是最好的标准。然而，它们在前向分步加法建模中都引出优秀的提升算法。对于平方误差损失，我们可以简单地在每一步用基本学习器拟合当前模型的残差$y_i - f_{m-1}(x_i)$。对于指数损失，可以用权值 $w_i = \exp(-y_i f_{m-1}(x_i))$对输出值$y_i$ 进行基本学习器的加权拟合。直接使用其他健壮性更强的标准对于简单可行的提升算法没有帮助。然而，在第 10.10.2 节，我们将展示如何基于任意可微的损失标准导出简单而优秀的提升算法，从而为数据挖掘产生高健壮性的提升过程。

10.7　数据挖掘的“现货”过程

预测学习是数据挖掘的一个重要方面。正如可以在本书中看到的，人们已经为从数据中预测学习开发了大量各种各样的方法。每种特定的方法都有适合于它的特定情况，而其他方法在这些数据上与最好的方法相比性能较差。在讨论每种方法时，我们都试图界定适合它的情况。然而，对任意给定的问题，很难预先知道哪一个过程最好或比较好。表 10.1 归纳了若干学习方法的某些特性。

表 10.1　不同学习方法的某些特性：▲ = 好，■ = 中等，● = 差

特性	神经网络	SVM	树	MARS	k-NN 核
混合类型数据的自然处理	●	●	▲	▲	●
遗漏值的处理	●	●	▲	▲	▲
对输入空间中孤立点的健壮性	●	●	▲	●	▲
对输入的单调转换的不敏感性	●	●	▲	●	●
计算的可伸缩性(大 N)	●	●	▲	▲	●
处理不相关输入的能力	●	●	▲	▲	●
提取特征的线性组合的能力	▲	▲	●	●	■
可解释性	●	●	■	▲	●
预测能力	▲	▲	●	■	▲

从对学习过程的需求来看,工业和商业的数据挖掘应用可能是一个挑战。就观测的个数和对每个观测度量的变量个数来说,数据集通常非常大。这样,计算方面的考虑起着重要的作用。数据通常还是"凌乱的":输入倾向于是定量的、二元的和分类(组)变量的混合,后者又常常分很多级。通常会有多个遗漏值,完整的观测很少。数值预测和响应变量的分布通常是长尾的并且高度倾斜。另外,它们通常还包含着大量的误度量(孤立点)。预测变量常以不同的标度来度量。

在数据挖掘应用中,包含在分析中的大量预测子变量仅有一小部分与预测是实际关联的。同时,与模式识别等诸多应用不同,这里很少有可靠的领域知识帮助建立特别相关的特征或者过滤掉不相关的特征,其结果是严重降低了许多方法的性能。

此外,数据挖掘应用通常需要可解释的模型。简单地产生预测是不够的,还需要提供输入变量的联合值与产生的预测响应值之间的联系的定性理解信息。像神经网络这样的黑盒方法,在模式识别的纯预测处理中是非常有用的,而对数据挖掘就没有多大用处。

对速度、可解释性的要求和数据凌乱的特性严重限制了大量学习过程,使之无法作为数据挖掘的现货方法。"现货(off-the-shelf)"方法是一种可以直接应用于数据,而不需要花费太多时间进行数据预处理或对学习过程小心进行调整的方法。

在大量著名的学习算法中,决策树最符合作为数据挖掘现货过程的要求。决策树可以相对快地构造并产生可解释的模型。正如第 9.2 节讨论过的那样,它们能很自然地合并数值的和分类的预测子变量的混合与遗漏值。在个体预测子(严格单调)的变换下,它们是不变的。结果,缩放或更一般的转换不是一种问题,并且它们不受预测子孤立点的影响。它们把进行内部特征选择作为过程的一个组成部分。因此,如果不是完全免疫,它们也能抵御包括许多不相关的预测变量的影响。决策树的这些特征是它们成为数据挖掘中最流行的学习方法的最大理由。

有一个因素使决策树不能成为理想的预测学习工具,即不精确性。与那些最好的处理手头数据的方法相比较,它们难以提供精确的预测。如在第 10.1 节所看到的那样,提升决策树提高了它们的精度,有时是引人注目的。同时,它还维持了数据挖掘所需要的特征。由于提升算法而牺牲掉的某些优点是速度、可解释性;而对 AdaBoost 来说,还牺牲了抵御重叠类分布和训练数据的误标号的健壮性。一个多元加法回归树(MART)是试图缓解这些问题的树提升算法的拓广,从而为数据挖掘产生一个精确而有效的现货过程。

10.8 例:垃圾邮件数据

在详细讨论 MART 之前,我们来证实一下它在 2-类分类问题上的能力。垃圾邮件数据是在第 1 章引进的,并被第 9 章的许多过程用做例子(见第 9.1.2 节、第 9.2.5 节、第 9.3.1 节和第 9.4.1 节)。

使用与第 9.1.2 节中相同的检验集,应用 MART 产生的检验误差率为 4.0%。相比之下,加法逻辑斯缔回归的检验误差率是 5.3%,完全地生长并用交叉验证剪枝的 CART 树为 8.7%,而 MARS 是 5.5%,这些估计的标准误差大约是 0.6%。

在下面的第 10.13 节,我们将为每个预测开发一种相对重要性度量,以及描述预测子对拟合模型贡献的偏依赖图。现在为垃圾数据来解释这些方法。

图 10.6 显示了全部 57 个预测变量的相对重要性频谱。显然，对于区分 spam 和 email，某些预测子比其他预测子更重要。字符串!、$、hp 和 remove 频率被估计为 4 个最相关的预测子变量。在频谱的另一端，字符串 857、415、table 和 3d 实质上没有相关性。

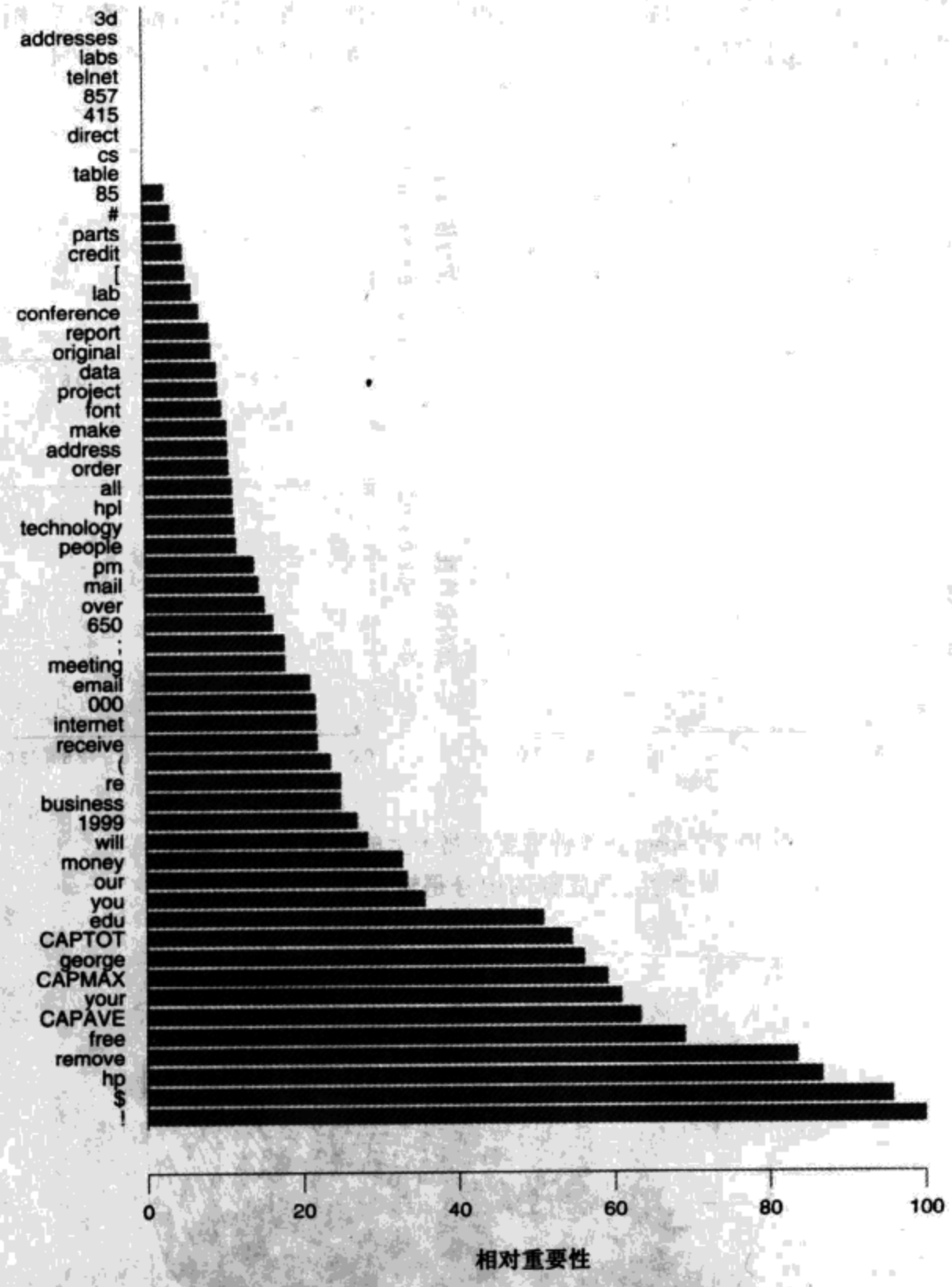


图 10.6　垃圾邮件数据的预测子变量重要性频谱，变量名写在垂直轴上

这里，被建模的量是 spam 和 email 的对数几率

$$f(x) = \log \frac{\Pr(\texttt{spam}|x)}{\Pr(\texttt{email}|x)} \tag{10.23}$$

(见第 10.13 节)。图 10.7 显示了选定的重要预测子的对数几率的偏依赖性，两个(! 和 remove)与 spam 正关联，两个(edu 和 hp)负关联。这些偏依赖性看上去是单调的。这与加法逻辑斯缔回归模型发现的对应函数大体上一致(见图 9.1)。

与完全 MART 模型(有 6 个端节点的树)产生的 4.0% 的误差对比,使用 $J=2$ 个端节点的树,在这些数据上运行 MART 产生了一个纯对数几率的加法(主要效果)模型,相应的误差率是 4.6%。因为检验集中有 1536 个观测,这些比率的标准误差接近于 $\sqrt{0.04(1-0.04)/1536}=0.005$ 或 0.5%。这个略高的误差表明在某些重要的预测变量之间可能存在着相互作用。这可以通过两变量偏依赖图来诊断。图 10.8 显示了具有强相互作用的一个图例。

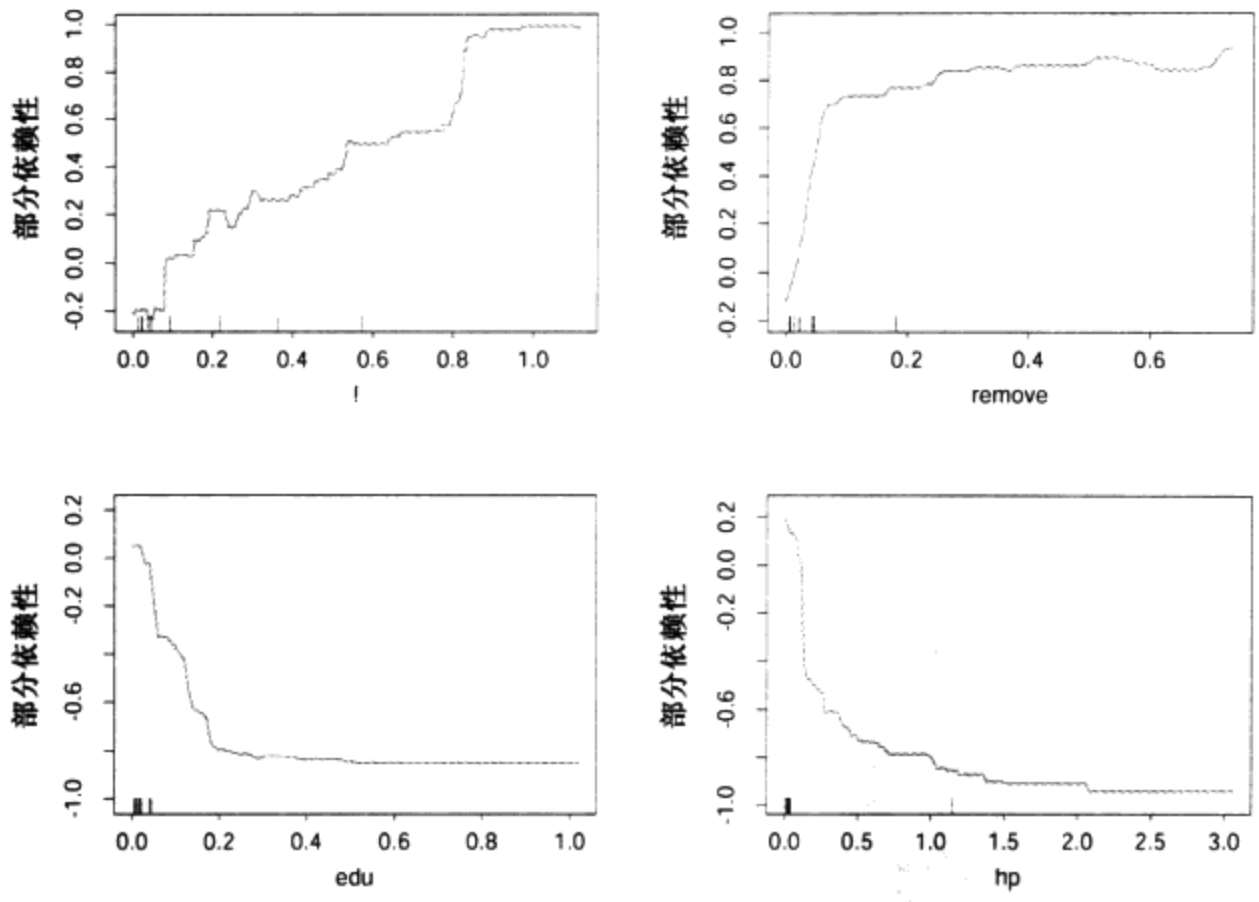


图 10.7 spam 在 4 种重要预测子上的对数几率的偏依赖性。图底部的记号是输入变量的十分位数

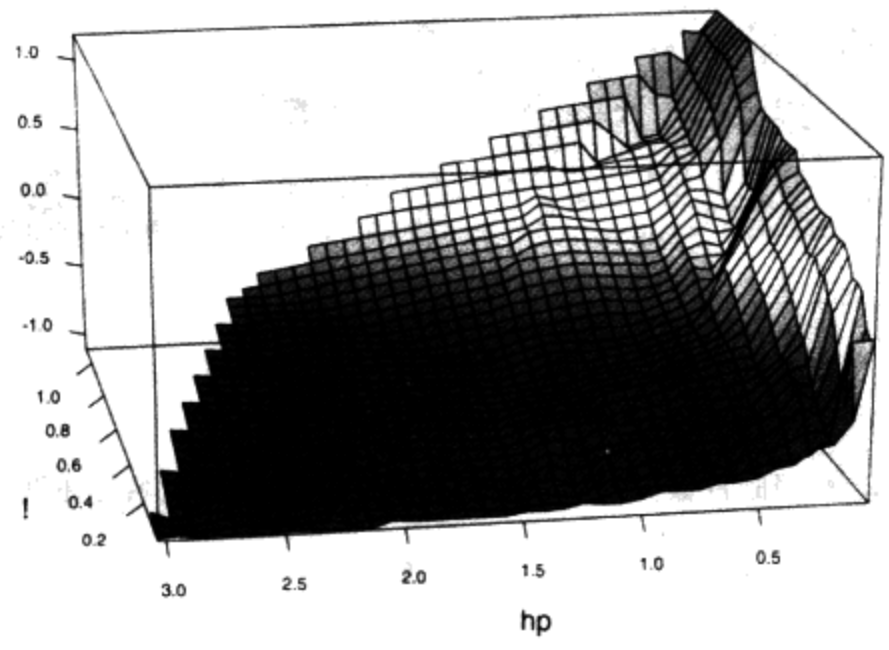


图 10.8 作为 hp 和! 联合的频率函数,spam 对 email 的对数几率的偏依赖(见彩页)

可以看到对 hp 非常低的频率,spam 的对数几率有较大增加。对 hp 的高频率,spam 的对数几率倾向于很低,并且它作为! 的函数粗略地看是一个常数。随着 hp 频率的降低,与! 的函数联系有所加强。

10.9　提升树

第 9.2 节详细讨论过回归和分类树。它们把所有联合预测变量值空间划分成不相交的区域 $R_j, j=1,2,\cdots,J$，像树的终端节点所表示的那样。把常量 γ_j 赋予每个这样的区域，预测规则是：

$$x \in R_j \Rightarrow f(x) = \gamma_j$$

这样，一棵树可以形式地表示为：

$$T(x;\Theta) = \sum_{j=1}^{J} \gamma_j I(x \in R_j) \tag{10.24}$$

其中，参数 $\Theta = \{R_j, \gamma_j\}_1^J$。$J$ 通常处理成一个元参数。该参数通过极小化如下经验风险发现：

$$\hat{\Theta} = \arg\min_{\Theta} \sum_{j=1}^{J} \sum_{x_i \in R_j} L(y_i, \gamma_j) \tag{10.25}$$

这是一个棘手的组合优化问题，我们通常接受近似的次优解。把优化问题分解成两部分是有用的：

给定 R_j 求 γ_j：给定 R_j，γ_j 的估计是平凡的，并且通常有 $\hat{\gamma}_j = \bar{y}_j$，落入区域 R_j 中的 y_i 的均值。对于误分类损失，$\hat{\gamma}_j$ 是落入 R_j 区域内观测的众数组。

求 R_j：这是困难的部分，为此要求近似解。注意，求 R_j 也必须估计 γ_j。一种典型的策略是使用贪心的自上而下的递归划分算法求 R_j。另外，有时也是需要用一个光滑子和较方便的优化 R_j 的标准来逼近式(10.25)：

$$\tilde{\Theta} = \arg\min_{\Theta} \sum_{i=1}^{N} \tilde{L}(y_i, T(x_i, \Theta)) \tag{10.26}$$

然后，给定 $\hat{R}_j = \tilde{R}_j$，使用原来的准则可以更精确地估计 γ_j。

在第 9.2 节，我们对分类树描述了这样的策略。Gini 索引代替了树增长中的误分类损失(识别 R_j)。

提升树模型是这样的树之和：

$$f_M(x) = \sum_{m=1}^{M} T(x;\Theta_m) \tag{10.27}$$

以逐步前向的方式(见算法 10.2)导出。在逐步前向过程的每一步，给定当前模型 $f_{m-1}(x)$，我们必须求解：

$$\hat{\Theta}_m = \arg\min_{\Theta_m} \sum_{i=1}^{N} L\left(y_i, f_{m-1}(x_i) + T(x_i;\Theta_m)\right) \tag{10.28}$$

其结果是关于下一棵树的区域集和常量 $\Theta_m = \{R_{jm}, \gamma_{jm}\}_1^{J_m}$。

给定区域 R_{jm}，在每个区域中求最佳常量 γ_{jm} 是直接的：

$$\hat{\gamma}_{jm} = \arg\min_{\gamma_{jm}} \sum_{x_i \in R_{jm}} L(y_i, f_{m-1}(x_i) + \gamma_{jm}) \tag{10.29}$$

求区域是困难的，甚至比求单棵树还难。对于几种特殊的情形，该问题可以简化。

对平方误差损失，求式(10.28)的解不比单棵树更困难。它仅仅是最好地预测当前残差 $y_i - f_{m-1}(x_i)$ 的回归树，而 $\hat{\gamma}_{jm}$ 是每个相应区域中这些残差的均值。

对 2-类分类和指数损失，这种分步方法导致提升分类树的 AdaBoost 方法(见算法 10.1)。特殊地，如果树 $T(x;\Theta_m)$ 被限制为定标的分类树，则我们在第 10.4 节中已经表明式(10.28)的解是极小化加权误差率 $\sum_{i=1}^{N} w_i^{(m)} I(y_i \neq T(x_i;\Theta_m))$ 的树，其中权值为 $w_i^{(m)} = e^{-y_i f_{m-1}(x_i)}$。定标的分类树是指 $\beta_m T(x;\Theta_m)$，其中，限制为 $\gamma_{jm} \in \{-1,1\}$。

没有这个限制，对于指数损失，式(10.28)仍然可以简化为对新树的加权指数准则：

$$\hat{\Theta}_m = \arg\min_{\Theta_m} \sum_{i=1}^{N} w_i^{(m)} \exp[-y_i T(x_i;\Theta_m)] \tag{10.30}$$

使用这个加权指数损失作为分裂标准，实现一个贪心的递归划分算法是直截了当的。注意，给定 R_{jm}，式(10.29)的解是每个对应区域上的加权对数几率：

$$\hat{\gamma}_{jm} = \log \frac{\sum_{x_i \in R_{jm}} w_i^{(m)} I(y_i = 1)}{\sum_{x_i \in R_{jm}} w_i^{(m)} I(y_i = -1)} \tag{10.31}$$

这需要一个特殊的树增长算法；在实践中，我们采用的近似方法是下面将要介绍的加权最小二乘方回归树。

使用诸如绝对误差和 Huber 损失(10.22) 等损失准则代替回归的平方误差损失，用散离(10.21)代替分类的指数损失将有助于提高分类树的健壮性。遗憾的是，与它们的非健壮性对应的算法不同，这些健壮性准则并不能导致简单快速的提升算法。

对于更一般的损失标准，给定 R_{jm}，式(10.29)的解通常是直截了当的，因为它是一个简单的“定位”估计。对绝对损失，它恰好是各区域残差的中值。对其他准则，存在求解式(10.29)的快速迭代算法，而且通常它们的快速“单步”逼近是可行的。问题是树的归纳。关于这些更一般的损失准则，求解式(10.28)的简单、快速算法并不存在，而像式(10.26) 的近似就成为必不可少的了。

10.10 数值优化

使用任意可微的损失准则求解式(10.28)的快速逼近算法可以通过模拟数值优化得到。在训练数据上使用 $f(x)$ 预测 y 的损失是：

$$L(f) = \sum_{i=1}^{N} L(y_i, f(x_i)) \tag{10.32}$$

其目标是关于 f 对 $L(f)$ 进行极小化，其中 $f(x)$ 被限制为树之和(10.27)。忽略这种限制，对式(10.32) 进行极小化就可以看做是一种数值优化：

$$\hat{\mathbf{f}} = \arg\min_{\mathbf{f}} L(\mathbf{f}) \tag{10.33}$$

其中，“参数”$\mathbf{f}\in \mathbb{R}^N$ 是逼近函数 $f(x_i)$ 在 N 个数据点 x_i 上的函数值：

$$\mathbf{f} = \{f(x_1), f(x_2), \cdots, f(x_N)\}$$

数值优化过程函数求解式 (10.33) 作为分量向量的和：

$$\mathbf{f}_M = \sum_{m=0}^{M} \mathbf{h}_m, \quad \mathbf{h}_m \in \mathbb{R}^N$$

其中 $\mathbf{f}_0 = \mathbf{h}_0$ 是初始猜测，每个后继的 $\mathbf{f}_m$ 都是根据当前参数向量 $\mathbf{f}_{m-1}$ 得到的，而 $\mathbf{f}_{m-1}$ 是前面导出的向量之和。数值优化方法在计算每个增量向量 $\mathbf{h}_m$（“步”）上是不同的。

10.10.1　最速下降

最速下降选择的是 $\mathbf{h}_m = -\rho_m \mathbf{g}_m$，其中 ρ_m 是一个标量，而 $\mathbf{g}_m \in \mathbb{R}^N$ 是在 $\mathbf{f} = \mathbf{f}_{m-1}$ 处计算的 $L(\mathbf{f})$ 的梯度。梯度 $\mathbf{g}_m$ 的分量是：

$$g_{im} = \left[\frac{\partial L(y_i, f(x_i))}{\partial f(x_i)}\right]_{f(x_i)=f_{m-1}(x_i)} \tag{10.34}$$

步长 ρ_m 是下式的解：

$$\rho_m = \arg\min_{\rho} L(\mathbf{f}_{m-1} - \rho \mathbf{g}_m) \tag{10.35}$$

接着，当前解被更新为：

$$\mathbf{f}_m = \mathbf{f}_{m-1} - \rho_m \mathbf{g}_m$$

并在下一次迭代中重复该过程。最速下降可看做是一个非常贪心的策略，因为 $-\mathbf{g}_m$ 是 $\mathbb{R}^N$ 中的一个局部方向，在此方向，$L(\mathbf{f})$ 在 $\mathbf{f} = \mathbf{f}_{m-1}$ 上下降最快。

10.10.2　梯度提升

逐步前向提升（见算法 10.2）也是一种非常贪心的策略。在每一步，给定当前模型 f_{m-1} 和它的拟合 $f_{m-1}(x_i)$，解树是对式 (10.28) 减少最多的那棵树。这样，树预测 $T(x_i;\Theta_m)$ 类似于负梯度 (10.34) 的分量。两者之间的主要不同是树分量 $\mathbf{t}_m = (T(x_1;\Theta_m), \cdots, T(x_N;\Theta_m))$ 不是独立的。它们被限制为一个 J_m 端点决策树的预测，而负梯度是无约束的最大下降方向。

在分步方法中，式 (10.29) 的解类似于最速下降中的线性搜索 (10.35)。其不同是式 (10.29) 分别线性搜索那些与每个终端区域 $\{T(x_i;\Theta_m)\}_{x_i\in R_{jm}}$ 对应 $\mathbf{t}_m$ 的分量。

如果极小化训练数据上的损失 (10.32) 是惟一目的，则最速下降将是最可取的策略。对于任意可微的损失函数 $L(y, f(x))$，梯度 (10.34) 的计算是平凡的，而对于第 10.6 节讨论的健壮性准则，求解式 (10.28) 是困难的。遗憾的是，梯度 (10.34) 仅定义在训练数据点 x_i 上，而最终目标是将 $f_M(x)$ 拓广到不在训练集中出现的新数据。

对这种困境的可能解决方法是在第 m 次迭代引进一棵树 $T(x;\Theta_m)$，它的预测 $\mathbf{t}_m$ 与负梯度尽可能接近。使用平方误差来度量接近程度，导致：

$$\tilde{\Theta}_m = \arg\min_{\Theta} \sum_{i=1}^{N} (-g_{im} - T(x_i;\Theta))^2 \tag{10.36}$$

即，我们通过最小二乘方将树 T 拟合到负梯度值(10.34)。正如在第 10.9 节提到的那样，对于最小二乘方决策树归纳，存在着快速算法。尽管式(10.36)的解区域 $\tilde{R}_{jm}$ 与求解(10.28)的区域 R_{jm} 不等价，但它们足够相似，可以用于相同的目的。在任何情况下，逐步前向提升过程和自上而下的决策树归纳本身就是近似过程。构造树(10.36)之后，在每个区域中的对应常量由式(10.29)给出。

表 10.2 概括了常用损失函数的梯度。对于平方损失误差，负梯度只是普通的残差 $-g_{im}=y_i-f_{m-1}(x_i)$，从而式(10.36)本身等价于标准最小二乘方提升。对于绝对误差损失，负梯度是残差的符号函数，因而在每次迭代，通过最小二乘方，式(10.36)将树拟合于当前残差的符号。关于 Huber M 回归，负梯度是这两种方法的折中(见表 10.2)。

表 10.2 常用损失函数的梯度

处理	损失函数	$-\partial L(y_i, f(x_i))/\partial f(x_i)$
回归	$1/2[y_i-f(x_i)]^2$	$y_i-f(x_i)$
回归	$\lvert y_i-f(x_i)\rvert$	$\text{sign}[y_i-f(x_i)]$
回归	Huber	$y_i-f(x_i)$，如果 $\lvert y_i-f(x_i)\rvert\leqslant\delta_m$ $\delta_m\text{sign}[y_i-f(x_i)]$，如果 $\lvert y_i-f(x_i)\rvert>\delta_m$ 其中，$\delta_m=$第 α 个分位数 $\{\lvert y_i-f(x_i)\rvert\}$
分类	散离	第 k 个分量：$I(y_i=\mathcal{G}_k)-p_k(x_i)$

对于分类的损失函数是多项式散离(10.21)，而 K 个最小二乘方树在每次迭代中构造。每个树 T_{km} 分别拟合它的负梯度向量 $\mathbf{g}_{km}$，

$$-g_{ikm}=\frac{\partial L(y_i,f_{1m}(x_i),\cdots,f_{1m}(x_i))}{\partial f_{km}(x_i)}=I(y_i=\mathcal{G}_k)-p_k(x_i) \tag{10.37}$$

其中，$p_k(x)$由式(10.20)给出。尽管 K 个分离的树在每次迭代中构造，但它们通过式(10.20)联系在一起。

10.10.3 MART

算法 10.3 给出回归的一般梯度树提升算法。通过插入不同的损失标准 $L(y,f(x))$，可以得到特定的算法。这个方法称为“多重加法回归树”(MART)。算法的第一行初始化最优常量模型，它是仅有一个端节点的树。步骤 2(a)计算的负梯度，称做广义或伪残差 r。常用损失函数的梯度汇总在表 10.2 中。

算法 10.3 多重加法回归树的梯度提升

1. 初始化 $f_0(x)=\arg\min_\gamma\sum_{i=1}^N L(y_i,\gamma)$
2. 对于 $m=1$ 到 M：
   (a) 对于 $i=1,2,\cdots,N$ 计算
   $$r_{im}=-\left[\frac{\partial L(y_i,f(x_i))}{\partial f(x_i)}\right]_{f=f_{m-1}}$$
   (b) 拟合回归树到目标 r_{im}，给出终端区域 R_{jm}，$j=1,2,\cdots,J_m$
   (c) 对于 $j=1,2,\cdots,J_m$，计算
   $$\gamma_{jm}=\arg\min_\gamma\sum_{x_i\in R_{jm}}L(y_i,f_{m-1}(x_i)+\gamma)$$
   (d) 更新 $f_m(x)=f_{m-1}(x)+\sum_{j=1}^{J_m}\gamma_{jm}I(x\in R_{jm})$
3. 输出 $\hat{f}(x)=f_M(x)$

关于分类,算法是类似的,步骤 2(a) ~ 2(d)在每次迭代 m 重复 K 次,对每个类应用式(10.37)一次。步骤 3 的结果是 K 个不同的(耦合的)树展开式 $f_{kM}(x), k=1,2,\cdots,K$。这些将通过式(10.20)产生概率,或像式(10.19)那样进行分类(见习题 10.5)。

与 MART.过程相关的调整参数是迭代次数 M 和每个组成树 $J_m(m=1,2,\cdots,M)$的大小。

10.11　提升适当大小的树

历史上,提升被视为一种模型组合技术,这里的模型是树。这样,树构造算法就看做是一种基本操作,它产生将被提升过程组合的模型。在这种情况下,每棵树的最佳大小在构造时以通常的方式分别进行估计(见第 9.2 节)。首先归纳出一棵非常大的树,然后自下而上对它进行修剪,得到被估计的终端节点的最佳个数。这种方法隐含地假设每棵树都是展开式(10.27)的最后一个。也许除了真正最后一棵树之外,该假设很明显是非常差的。其结果是树变得非常大,特别是在早期的迭代中。这无疑会降低性能而增加计算开销。

避免该问题的最简单策略就是限制所有的树都一样大,$J_m=J, \forall m$。在每次迭代中,导出一棵 J 端节点的回归树。这样,J 变成整个提升过程的元参数。调整它使手头数据的估计性能极大化。

为了获得 J 的有用值,考虑如下"目标"函数的性质:

$$\eta=\arg\min_f \mathrm{E}_{XY}L(Y,f(X)) \tag{10.38}$$

这里,期望值是总体上(X,Y)的联合分布。目标函数 $\eta(x)$在未来数据上具有极小预测风险。这正是我们试图去逼近的函数。

$\eta(X)$的一个相关特征是坐标变量 $X=(X_1, X_2,\cdots, X_p)$之间的交互作用程度。这个可以由它的 ANOVA(方差分析)展开得到:

$$\eta(X)=\sum_j \eta_j(X_j)+\sum_{jk}\eta_{jk}(X_j,X_k)+\sum_{jkl}\eta_{jkl}(X_j,X_k,X_l)+\cdots \tag{10.39}$$

式(10.39)的第一个和是对只有一个预测子变量 X_j 的函数求和。其中,函数 $\eta_j(X_j)$是这样一些函数,依据所使用的损失标准,它们联合地对 $\eta(X)$进行最佳逼近。每个这样的 $\eta_j(X_j)$称为 X_j 的"主效应"。第二个和在双变量函数上求和,当添加到主效应上时,最好地拟合了 $\eta(X)$。它们称为各个变量对(X_j, X_k)的二阶交互效应。第三个和表示三阶交互效应,以此类推。对于很多实际当中遇到的问题,低阶交互效应趋于占支配地位。当遇到这样的情况,模型产生很强的高阶交互效应时(如大型决策树),将要面临精度问题。

基于树逼近的交互效应阶受树大小 J 的限制。即不可能有阶大于 $J-1$ 的交互效应。由于提升的模型在树(10.27)中是加法的,因此,该限制也延伸到它们当中。置 $J=2$(单分裂"决策树桩")产生一个只有"主效应"的提升模型,不允许交互效应。取 $J=3$,两个变量的交互效应是允许的,以此类推。这表明 J 值的选择应当反映支配 $\eta(x)$的交互效应的阶。当然,这通常是未知的,但多数情况下它都很低。图 10.9 显示了模拟例子(10.2)上的交互效应的阶(J 的选择)。生成的函数是加法的(二次单项式之和),这样提升模型在 $J>2$ 时将导致不必要的方差,并引起较高的检验误差。图 10.10 将提升树桩发现的坐标函数与真实函数进行了比较。

尽管在许多应用中 $J=2$ 不够充分，但也不大可能需要 $J>10$。迄今为止的经验表明，$4\leqslant J\leqslant 8$很适合于提升算法，结果对这个范围中的特定选择很不敏感。可以这样调整 J 的值：通过试用几个不同的值，并选择一个在确认样本上产生最低风险的 J 值。但是，它对使用 $J\simeq 6$，很难有较明显的改进。

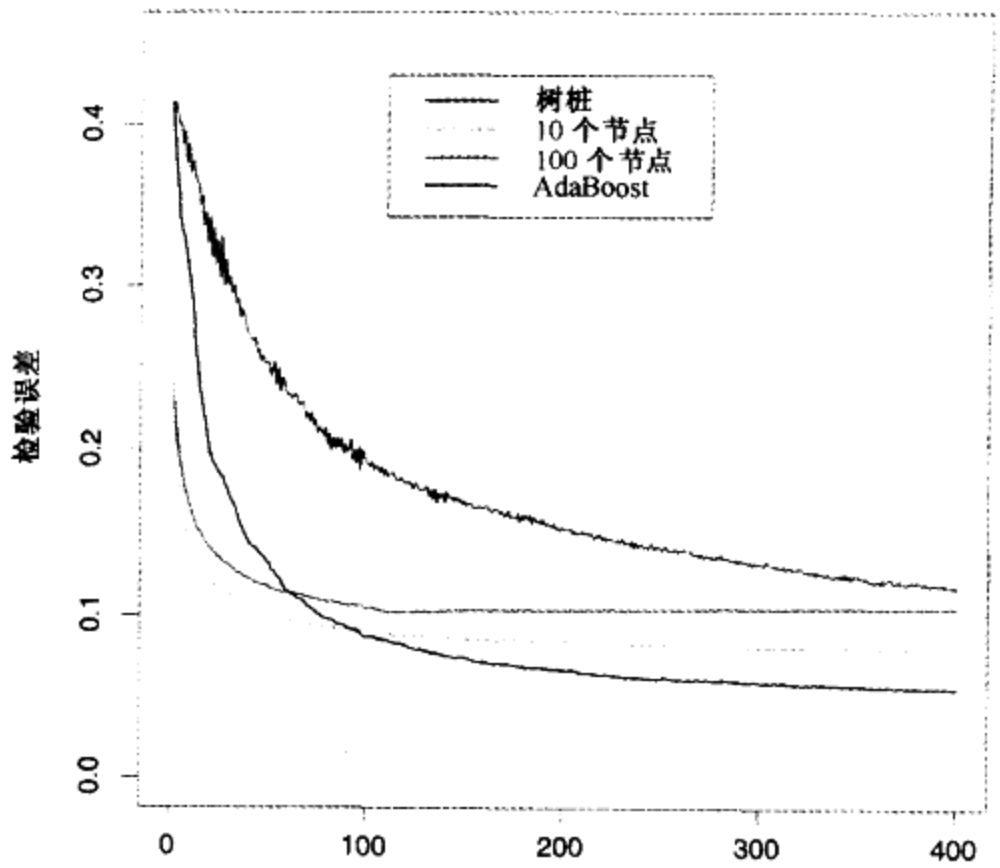


图 10.9 提升不同大小的树，用于图 10.2 使用的例子(10.2)。由于生成的模型是加法的，所以树桩性能最好。提升算法使用算法10.3中的二项式散离损失；为了比较，还显示了AdaBoost算法10.1(见彩页)

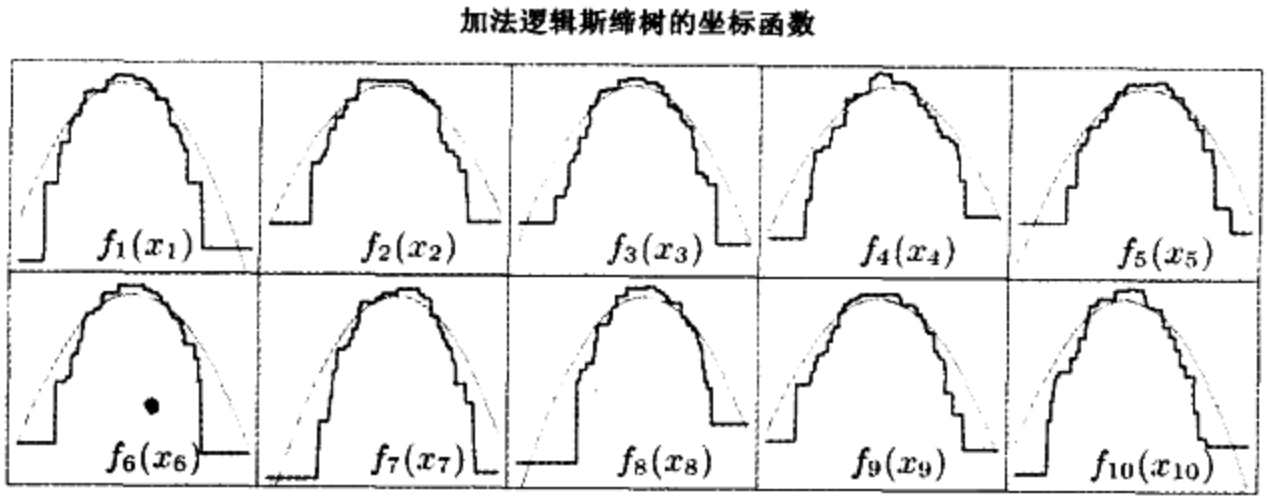


图 10.10 对于图 10.9 使用的模拟例子，提升树桩估计的坐标函数。为了比较，还显示了真实的二次函数

10.12 正则化

除成员树的大小 J 之外，MART 过程的其他元参数是提升迭代次数 M。通常，每次迭代都会减少训练风险 $L(f_M)$，使得 M 足够大时该风险可以任意小。然而，对训练数据拟合太好可能导致过分拟合，低估未来预测的风险。这样，存在一个最优的 M^*，极小化依赖于应用的未

来风险。估计 M^* 较方便的方法是:把预测风险作为 M 的函数在验证样本上检验。取极小化该风险的 M 值作为 M^* 的估计。这类似于神经网络经常使用的及早停止策略(early stopping strategy)(见第 11.4 节)。

10.12.1　收缩

控制 M 的值不是惟一可行的正则化策略。与岭回归和神经网络一样,也可以使用收缩技术(见第 3.4.3 节和第 11.5 节)。在提升背景下,收缩的最简单实现是当每棵树添加到当前逼近时,将树的贡献用因子 $0<\nu<1$ 缩放。这样,算法 10.3 的 2(d)行用下式替换:

$$f_m(x) = f_{m-1}(x) + \nu \cdot \sum_{j=1}^{J} \gamma_{jm} I(x \in R_{jm}) \tag{10.40}$$

参数ν可以看做控制提升过程的学习率。对于相同的迭代次数 M,较小的ν值(收缩更多)导致较大的训练风险。这样,ν和 M 共同控制训练数据上的预测风险。然而,这些参数的作用不是独立的。对于相同的训练风险,较小的ν值导致较大的 M 值,因此需要在它们之间做出权衡。

经验表明(Friedman,2001):较小的ν值有助于产生较好的检验误差,同时也相应地需要较大的 M 值。实际上,最好的策略似乎是先使ν的值非常小($\nu<0.1$),然后,使用及早停止策略选择 M。对于回归和概率估计,这产生了引人注目的改进(与无收缩$\nu=1$ 相比)。虽然对由式(10.19)表示的误分类风险,相应的改进较少,但改进仍然是显著的。为这种改进所付出的代价是计算开销:较小的ν值导致较大的 M 值,而计算量正比于后者。然而,正如下面将要看到的,即使对于大型数据集,许多迭代在计算上通常也是可行的。部分原因在于每一步产生的树都较小,无须剪枝。

图 10.11 显示了图 10.2 中模拟例子(10.2)的检验误差曲线。使用树桩或 6 个终端节点的树,用二项式散离,使用或不使用收缩技术训练 MART。收缩的好处是明显的,特别是对二项式散离。使用收缩,每条检验误差曲线可以达到较低的值,且对多次迭代一直保持着较低的值。

10.12.2　罚回归

收缩策略(10.40)的成功可以通过提取与使用大型基展开式的罚线性回归的相似性加以验证。考虑所有可能的 J 终端节点的回归树的集合$\tau = \{T_k\}$,它们可以作为 $\mathbb{R}^p$ 中的基函数在训练数据上实现。该线性模型是:

$$f(x) = \sum_{k=1}^{K} \alpha_k T_k(x) \tag{10.41}$$

其中,$K = \mathrm{card}(\tau)$。假设系数用最小二乘方估计。由于这种树的个数很可能比最大的训练数据集还大得多,所以需要罚最小二乘方:

$$\hat{\alpha}(\lambda) = \arg\min_{\alpha} \left\{ \sum_{i=1}^{N} \left(y_i - \sum_k \alpha_k T_k(x_i) \right)^2 + \lambda \cdot J(\alpha) \right\} \tag{10.42}$$

其中,α是参数向量,而 $J(\alpha)$是系数的函数,通常罚较大的值。例子有:

$$J(\alpha) = \sum_{k=1}^{K} \alpha_k^2 \qquad \text{岭回归} \tag{10.43}$$

$$J(\alpha) = \sum_{k=1}^{K} |\alpha_k| \qquad \text{套索(见第 3.4.3 节)} \tag{10.44}$$

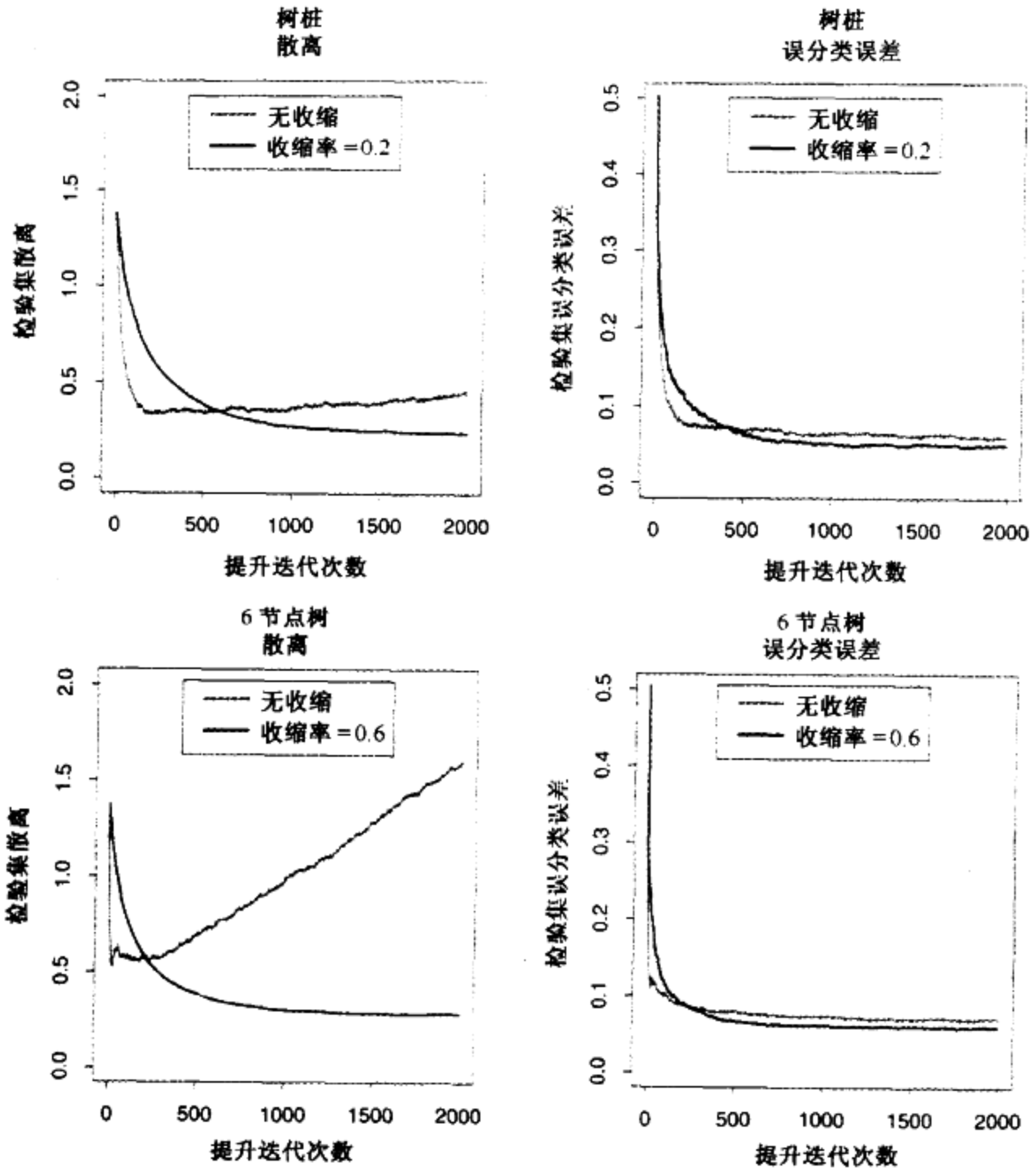


图 10.11 使用 MART,图 10.9 模拟例子(10.2)的检验误差曲线。使用二项式散离、树桩或 6 节点树训练模型,有或没有收缩的情形。左侧报告检验散离,而右侧显示误分类误差。收缩的有利效果在所有情况中都能看得到,特别是对于左边显示的散离(见彩页)

正如第 3.4.3 节的讨论,对于适度大的 λ,套索问题的解趋于稀疏;许多$\hat{\alpha}_k(\lambda) = 0$。即所有可能的树只有一小部分进入模型(10.41)。这看上去是合理的,因为在逼近任意特定的目标函数时,所有可能的树中通常只有一小部分是相关的。然而,对于不同的目标,相关的子集可能不同。没有设置为 0 的那些系数被套索收缩,因为它们的绝对值小于对应的最小二乘方值:$|\hat{\alpha}_k(\lambda)| < |\hat{\alpha}_k(0)|$。随着 λ 的增加,所有系数都被收缩,且每个系数最终都变成 0。

由于基函数 T_k 的个数非常大,用套索罚(10.44)直接解决式(10.42)是不可能的。然而,存在一种可行的逐步前向策略,它与套索的作用非常近似,而且与提升和逐步前向算法 10.2 非常相似。算法 10.4 给出了细节。尽管使用树基函数 T_k 来表达,该算法仍然可以与任意基

函数集一起使用。初始时，所有系数都为 0(步骤 1)；这对应于式(10.42)中 $\lambda = \infty$。在每个后继步，2(a)行都选择树 T_{k^*}，它最好地拟合当前残差。然后，其对应的系数α_{k^*}在 2(b)行增加或减少一个微量，而其他系数α_k，$(k \neq k^*)$仍保持不变。原则上，该过程可以重复，直到所有残差都为 0，或者 $\beta^* = 0$。如果 $K < N$，后一种情况可能会发生，而那时系数值代表最小二乘方解。这对应于式(10.42) 中 $\lambda = 0$ 的情形。

算法 10.4　逐步前向线性回归

1. 初始化$\hat{\alpha}_k = 0, k = 1, \cdots, K$。置 $\varepsilon > 0$ 为某个小常量，M 为某个大常数
2. 对于 $m = 1$ 到 M：
   (a) $(\beta^*, k^*) = \arg\min_{\beta,k} \sum_{i=1}^{N} \left(y_i - \sum_{l=1}^{K} \alpha_l T_l(x_i) - \beta T_k(x_i)\right)^2$
   (b) $\alpha_{k^*} \leftarrow \alpha_{k^*} + \varepsilon \cdot \text{sign}(\beta^*)$
3. 输出 $f(x) = \sum_{k=1}^{K} \alpha_k T_k(x)$

使用算法 10.4，执行 $M < \infty$ 次迭代之后，许多系数都将为 0，即那些没有增大的系数将为 0。其他系数的绝对值将趋向于小于它们对应的最小二乘方解，即 $|\hat{\alpha}_k(M)| < |\hat{\alpha}_k(0)|$。因此，该 M 迭代解和套索定性相似，M 与 λ 逆相关。

图 10.12 展示了一个例子，使用的是第 3 章的前列腺数据。这里不使用树 $T_k(X)$作为基函数，而是使用原来的变量 X_k 本身；即一个多元线性回归模型。左图显示了对于不同约束参数 $t = \sum_k |\alpha_k|$，由套索估计的系数曲线。右图显示了逐步前向算法 10.4 的结果，其中，$M = 250, \varepsilon = 0.01$。两个图之间的相似性是显而易见的。

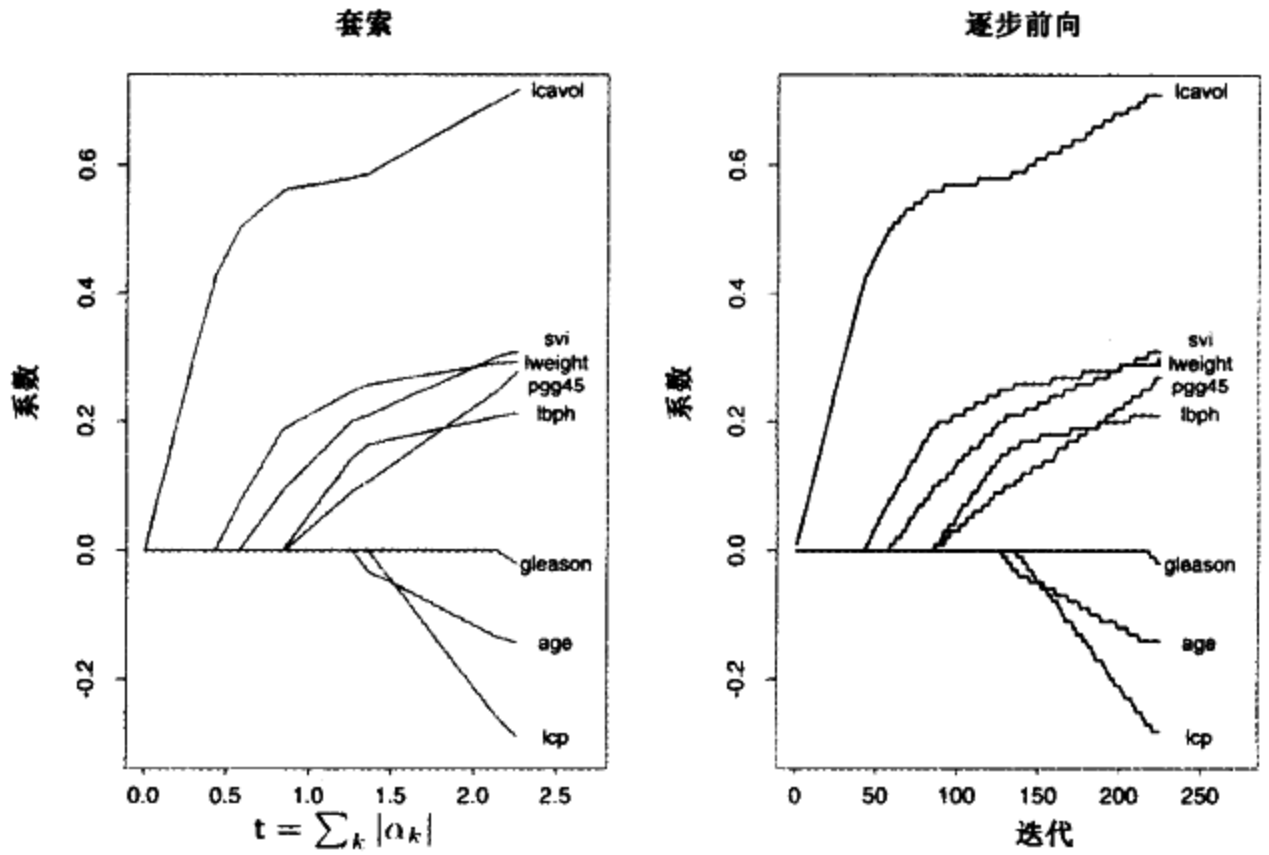


图 10.12　对于第 3 章中的前列腺数据，由线性回归估计的系数曲线。左图显示对不同约束参数 $t = \sum_k |\alpha_k|$ 套索的结果。右图显示逐步线性回归算法10.4的结果，使用 $M = 250$，相继步长 $\varepsilon = 0.01$

在某些情况下，相似性不仅是定性的。例如，如果全部基函数 T_k 互不相关，那么随$\varepsilon \to 0$，算法 10.4(对于 $0 < M < \infty$)恰恰产生与套索(对于$(\infty \geqslant \lambda \geqslant 0)$相同的解集。当然，基于树的回

归子并非不相关。但是，如果系数$\alpha_k(\lambda)$都是λ的单调函数，那么解集也完全相同。这是很常见的情形。当$\alpha_k(\lambda)$不是λ的单调函数时，则解集不相同，但通常很接近。算法 10.4 的解集随正则化参数值的改变比套索慢。

使用收缩(10.40)的树提升(见算法 10.3)与算法 10.4 非常相似，其学习率参数ν对应于ϵ。对于平方误差损失，惟一的区别是每次迭代选择的最优树 T_k· 由标准自上而下贪心的归纳算法来逼近。对于其他损失准则，收缩与具有特殊罚的罚回归不是严格可比的。然而，定性地说，可以期望存在类似于最小二乘方的对应。这样，可以将收缩的树提升看做所有可能(J 端节点)树上的回归，用套索罚(10.44)作为正则化子。

不使用收缩[在式(10.40)中，$\nu = 1$]的树提升类似于罚非零系数 $J(\alpha) = \sum_k |\alpha_k|^0$ 的个数的子集选择。对于预测，众所周知子集选择过分贪心(Copas, 1983)，与诸如套索或岭回归等保守方法相比，产生较差的结果。在提升的背景下，由于收缩导致的引人注目的改进再次证实了这种方法的优势。

10.12.3 L_1 罚(套索) 优于 L_2

如上一节所示，使用收缩提升的逐步前向策略可以近似地极小化与套索型 L_1 罚相同的损失函数。模型逐步建立起来，穿越“模型空间”并增加重要预测子的收缩函数。相比之下，L_2 罚就非常容易处理，如第 12.3.6 节所示。选定匹配一个特殊正定核的基函数和 L_2 罚，我们可以求解对应的极小化问题，而不必显式地搜索各个基函数。

然而，提升在诸如支持向量机等过程上的卓越性能在很大程度上可能是由于隐式地使用了罚 L_1 而不是罚 L_2。由罚 L_1 而引起的收缩能更好地适应稀疏解，那里非 0 权值的基函数(在所有可能的选择中) 很少。对于小波基的特定情况，Donoho 等人(1995)给出了支持该断言的一些结果。直接对罚 L_1 问题进行极小化比 L_2 要困难得多，但是提升的逐步前向算法提供了一种能实际解决该问题的近似策略。

10.13 可解释性

单棵决策树是可解释的。整个模型完全可以用可视化的二维图(二叉树)表示。树的线性组合(10.27)失去了这一重要特征，因此必须用不同的方式解释。

10.13.1 预测子变量的相对重要性

在数据挖掘应用中，输入预测子变量很少等同相关。通常，只有少数预测子对响应有实质性的影响；而绝大多数是不相关的，可以不必包含在内。了解每个输入变量在预测响应中的相对重要性或贡献常常是有用的。

对于单棵决策树 T，Breiman 等人(1984)提出：

$$\mathcal{I}_\ell^2(T) = \sum_{t=1}^{J-1} \hat{\imath}_t^2 I(v(t) = \ell) \tag{10.45}$$

可以作为每个预测子变量 X_ℓ 相关性的一种度量。求和在树的 $J-1$ 个内部节点上进行。在

每个内部节点 t 上，使用一个输入变量 $X_{v(t)}$ 把与该节点相关联的区域分割成两个子区域；在每个子区域，用一个常量拟合响应值。选取的特定变量是，它对在常量拟合整个区域上平方误差风险的估计改进 i_t^2 极大。变量 X_ℓ 的平方相对重要性是它被选择作为分裂变量所有内部节点上的这种改进的平方和。

这个重要性度量可以很容易推广到加法树展开式(10.27)上，它简单地对树求平均值：

$$\mathcal{I}_\ell^2 = \frac{1}{M}\sum_{m=1}^{M}\mathcal{I}_\ell^2(T_m) \tag{10.46}$$

由于求平均的稳定效果，该度量比它在单棵树上的对应度量(10.45)更加可靠。由于收缩(见第 10.12.1 节)，重要变量被其他高度相关的变量屏蔽也不是什么问题。注意，式(10.45)和式(10.46)涉及到平方相关性，实际的相关性是它们的平方根。由于这些度量是相对的，通常的惯例是令极大值为 100，然后相应调整其他值。图 10.6 显示了在预测 spam 和 email 中 57 个输入的相对重要性。

对于 K-类分类，引进 K 个模型 $f_k(x)$，$k = 1, 2, \cdots, K$，每个都包含树的和：

$$f_k(x) = \sum_{m=1}^{M} T_{km}(x) \tag{10.47}$$

在这种情况下，式(10.46)拓广为：

$$\mathcal{I}_{\ell k}^2 = \frac{1}{M}\sum_{m=1}^{M}\mathcal{I}_\ell^2(T_{km}) \tag{10.48}$$

这里，$\mathcal{I}_{\ell k}$ 是 X_ℓ 把类 k 的观测与其他类分开的相关性。X_ℓ 的总相关性可以通过在所有类上求平均值得到：

$$\mathcal{I}_\ell^2 = \frac{1}{K}\sum_{k=1}^{K}\mathcal{I}_{\ell k}^2 \tag{10.49}$$

图 10.19 和图 10.20 显示了这些求平均和各自的相对重要性的使用。

$\mathcal{I}_{\ell k}$ 本身就很有用。我们可以以不同的方式概括这些相关性值的 $p \times K$ 矩阵。每一列 $\mathcal{I}_k$ 给出变量分离类 k 的相对重要性。每一行 $\mathcal{I}_\ell$ 揭示 X_ℓ 分离各个类的影响。我们可以在选定的类子集上求矩阵元素(10.48)的平均值，以确定该子集变量的相关性。类似地，可以在变量的子集上求平均值，以确定所选定的变量子集对分离哪些类的影响最大。

10.13.2　偏依赖图

在大部分相关变量确定之后，下一步就是试图理解逼近 $f(X)$ 对它们的联合值的依赖特点。可视化是最有效的工具之一。作为自变量的函数，$f(X)$ 的透视图提供了它对输入变量联合值依赖的全面概括。

遗憾的是，这种可视化只对低维视图适用。我们可以很容易地用各种不同的方式显示--两个自变量的或者连续或不连续(或者混合的)的函数；本书到处都是这种显示。维数稍高的函数可以通过限制于一两个自变量上特定值的集合来绘制，产生格子图(trellis)(Becker 等人，1996)。

对于多于两个或三个变量的情况，观察高维自变量对应的函数就比较困难。一种有用的替代方法是观察一组图，每幅图显示在选定输入变量的一个较小子集上逼近 $f(X)$ 的偏依赖性。尽管这样一组图不能提供逼近的全面描述，但它通常能提供有用的线索，特别是当 $f(X)$ 被式(10.39)的低阶交互作用左右时更是如此。

考虑输入预测子变量 $X=(X_1, X_2, \cdots, X_p)$ 的 $\ell < p$ 子向量 X_S，这里下标集 $S \subset \{1,2,\cdots, p\}$。令 C 表示补集，$S \cup C = \{1, 2, \cdots, p\}$。一般函数 $f(X)$ 原则上将依赖于全部输入变量：$f(X)=f(X_S, X_C)$。一种定义 $f(X)$ 在 X_S 上的平均或偏依赖的方法是：

$$f_S(X_S) = \mathrm{E}_{X_C} f(X_S, X_C) \tag{10.50}$$

这是 f 的边缘平均，而且当 X_S 中的变量与 X_C 中的变量没有很强的交互作用时，它可以作为选定子集在 $f(X)$ 上作用的有用描述。

偏依赖函数可以用于解释任何“黑盒”学习方法的结果。它们可以由下式估计：

$$\bar{f}_S(X_S) = \frac{1}{N}\sum_{i=1}^{N} f(X_S, x_{iC}) \tag{10.51}$$

其中，$\{x_{1C}, x_{2C}, \cdots, x_{NC}\}$ 是出现在训练数据中的 X_C 的值。这需要为 X_S 每个联合值的集合传递数据，以计算 $\bar{f}_S(X_S)$。这可能增加计算强度，甚至对中等大小的数据集也是一样。所幸的是对于决策树 $\bar{f}_S(X_S)$ 式(10.51)可以迅速地计算，无须引用数据(Friedman, 2001)。对于加法树模型(10.27)，其结果是在成分树上求平均。

重要的是需要注意，式(10.50)中定义的偏依赖函数表示考虑了其他变量 X_C 对 $f(X)$ 的(平均)影响之后 X_S 对 $f(X)$ 的影响。它们不是忽略了 X_C 作用之后 X_S 对 $f(X)$ 的影响。后者由下面的条件期望给出：

$$\tilde{f}_S(X_S) = \mathrm{E}(f(X_S, X_C)|X_S) \tag{10.52}$$

它是仅用 X_S 的函数对 $f(X)$ 的最佳最小二乘方逼近。量 $\tilde{f}_S(X_S)$ 和 $\bar{f}_S(X_S)$ 只有在全部预测子变量间的完全独立这种不太可能的情况下才是相同的。例如，如果选定的变量子集的作用是纯粹加法的，

$$f(X) = h_1(X_S) + h_2(X_C) \tag{10.53}$$

那么式(10.50)产生 $h_1(X_S)$，相差一个加法常量。如果作用是纯粹乘法的，

$$f(X) = h_1(X_S) \cdot h_2(X_C) \tag{10.54}$$

那么式(10.50)产生 $h_1(X_S)$，相差一个乘法常量因子。另一方面，式(10.52)在两种情况下都不会产生 $h_1(X_S)$。事实上，式(10.52)对 $f(X)$ 完全不依赖的变量子集可能会产生很强的作用。

观察提升树逼近式(10.27)在选定变量子集上的偏依赖性有助于提供其特性的定性描述。例证在第 10.8 节和第 10.14 节给出。由于计算机图形和人们感知的限制，子集 X_S 必须很小($l \approx 1,2,3$)。当然，这样的子集有很多，但是，只有从非常小的且高度相关的预测子中选择的子集才可能有丰富的信息。此外，对 $f(X)$ 的作用是近似加法的(10.53)或乘法的(10.54)那些子集将最有启发性。诊断这种情况所出现的程度可以通过如下方法得到：对于式(10.53)，计算 $f(X)$ 与 $\bar{f}_S(X_S)$ 和 $\bar{f}_C(X_C)$ 的多重相关系数；而对于式(10.54)，计算 $f(X)$ 与 $\bar{f}_S(X_S)\cdot \bar{f}_C(X_C)$ 的简单相关。

对 K-类分类，有 K 个分离的模型(10.47)，每个模型对应一个类。每个模型通过下式与各自的概率相关联：

$$f_k(X) = \log p_k(X) - \frac{1}{K}\sum_{l=1}^{K} \log p_l(X) \tag{10.55}$$

这样，每个 $f_k(X)$是各自概率对数标度上的单调递增函数。每个 $f_k(X)$ (10.47) 在其最相关的预测子(10.48)上的偏依赖图有助于揭示该类的对数几率如何依赖各自的输入变量。

10.14　实例

本节，我们用两个较大的公共领域数据集进一步讲解 MART 过程。在这两个例子中，成员树的大小(见第 10.11 节)取 $J = 6$ 个端节点，而学习率(10.40)置成$\nu = 0.1$。Huber 损失标准用于预测数值响应(回归)和分类的多项式散离。每个数据集 20% 的随机样本作为评估性能的检验数据集，而在剩余的 80%数据上训练模型。

10.14.1　加利福尼亚住房

该数据集(Pace 和 Barry, 1997)取自 Carnegie-Mellon StatLib 仓库[①]。它包括加利福尼亚的 20 460个小区(1990 人口普查组)的聚集数据。响应变量 Y 是每个小区的房价中值，以 $100 000 为单位度量。预测子变量是人口统计量，如收入中值 MedInc，住房密度由房子数量 House 反映，每个住宅的平均居住率用 AveOccup 表示。预测子还包括每个小区的位置(经度 longitude 和纬度 latitude)，以及几个反映小区房屋特征的量：平均房间数 AveRooms 和卧室数 AveBedrms。共有 8 个预测子，都是数值型的。

图 10.13 显示了平均绝对误差：

$$AAE = \mathrm{E}\,|y - \hat{f}_M(x)| \tag{10.56}$$

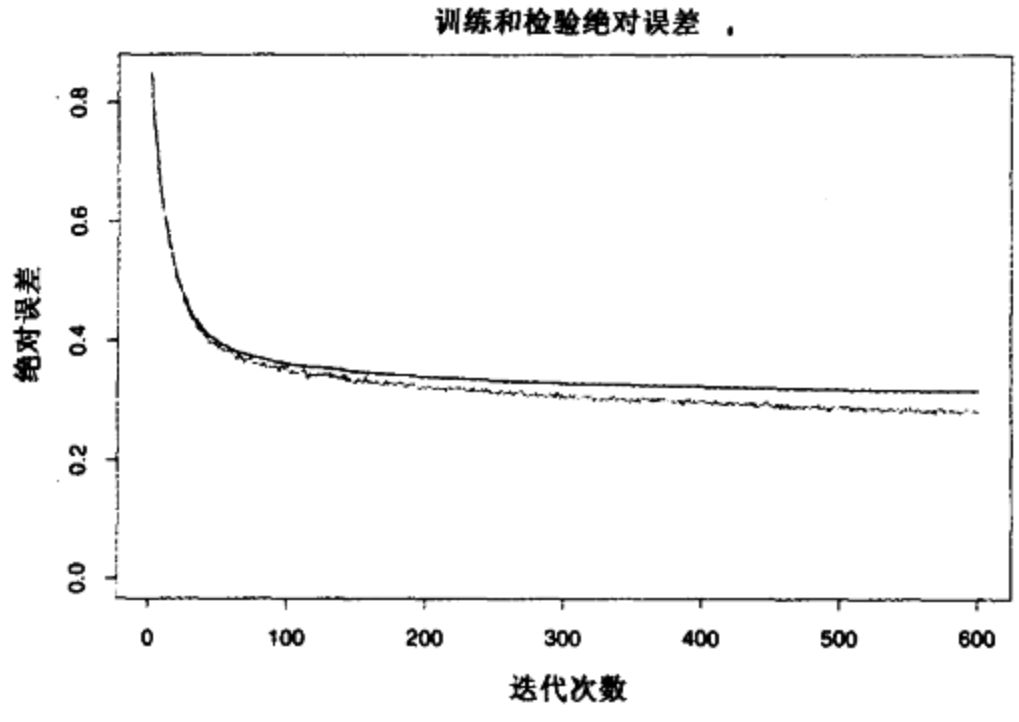


图 10.13　对于加利福尼亚住房数据，平均绝对误差是迭代次数的函数

① http://lib.stat.cmu.edu。

它是训练数据(较低的曲线)和检验数据(较高的曲线)上迭代次数 M 的函数。由于 Friedman (1999)中描述的算法具有随机性,所以训练误差曲线的外形粗糙并且有摆动。检验误差看上去随 M 的增长而单调递减,在早期阶段下降很快,之后随着迭代次数的增加而稳定下来并接近于常量。这样,M 特定值的选取就不大重要,只要它不是特别小就可以。几乎在所有的应用中都是这种情况。收缩策略(10.40)通常排除了过分拟合问题,特别是对大型数据集。

600 次迭代后的 AAE 值是 0.31。这可以与最优常量预测子的中值 $\{y_i\}$ 相比较,后者为 0.89。使用较熟悉的量,该模型的平方多元相关系数是 $R^2 = 0.84$。Pace 和 Barry(1997)使用了复杂的空间自动回归过程,其中对每个小区的预测都是根据附近小区的房价中值,使用其他预测子作为协变量。用变换来做实验,它们预测 log Y 时达到 $R^2 = 0.85$。用 log Y 作响应,MART 的相应值是 $R^2 = 0.86$。

图 10.14 显示了 8 个预测子变量的相对重要性。毫不奇怪,小区收入中值是最相关的预测子。经度、纬度和平均居住面积的相关性大约都是收入的一半,而其他预测子的影响则较小。

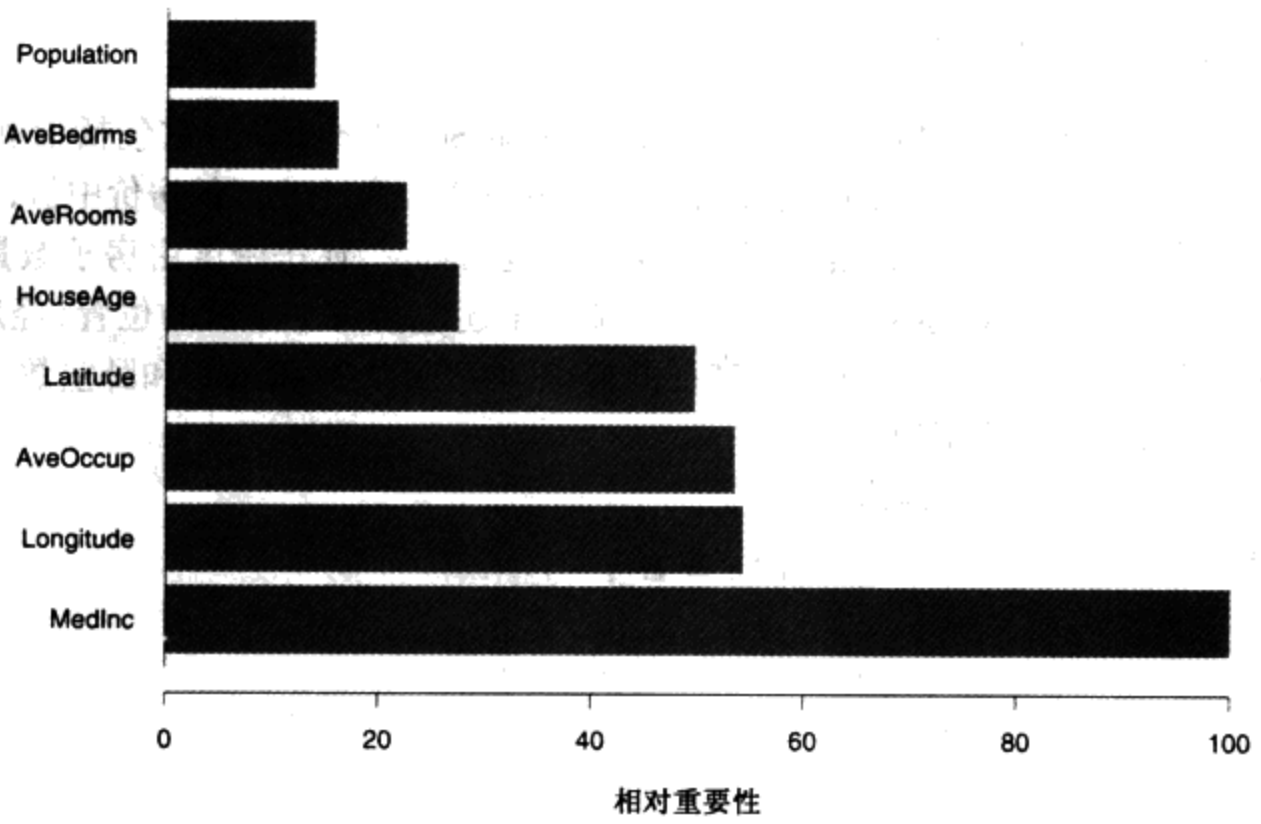


图 10.14 加利福尼亚住房数据的预测子的相对重要性

图 10.15 显示了单变量依赖于最相关的非定位预测子的偏依赖图。注意,这些图不是严格光滑的。这是使用基于树模型的结果。决策树产生了非连续分段常量模型(10.24)。继续在树(10.27)上求和,当然有更多的段。与本书中讨论的大部分方法不同,这里并没有对结果强加光滑性限制。任意陡峭的非连续性都可以建模。这些曲线通常都展示光滑趋势,其原因是所估计的是对问题的响应的最佳预测。通常都是这种情形。

每幅图底部的散列标记描绘了相应变量数据分布的十分位。注意,这里的数据靠近边界密度较低,特别是对较大的值。这就导致了在这些区域中曲线多少有些不确定。图的垂直刻度是相同的,并给出了不同变量相对重要性的可视化比较。

房价中值对收入中值的偏依赖是单调增的,在数据的主体上接近线性。房价通常随着平均居住率的增加而单调递减,可能的例外是平均居住率低于 1 的情况。房价中值对平均房间

数的偏依赖是非单调的。它在接近 3 个房间时取极小值，而且对较小和较大的值都是递增的。

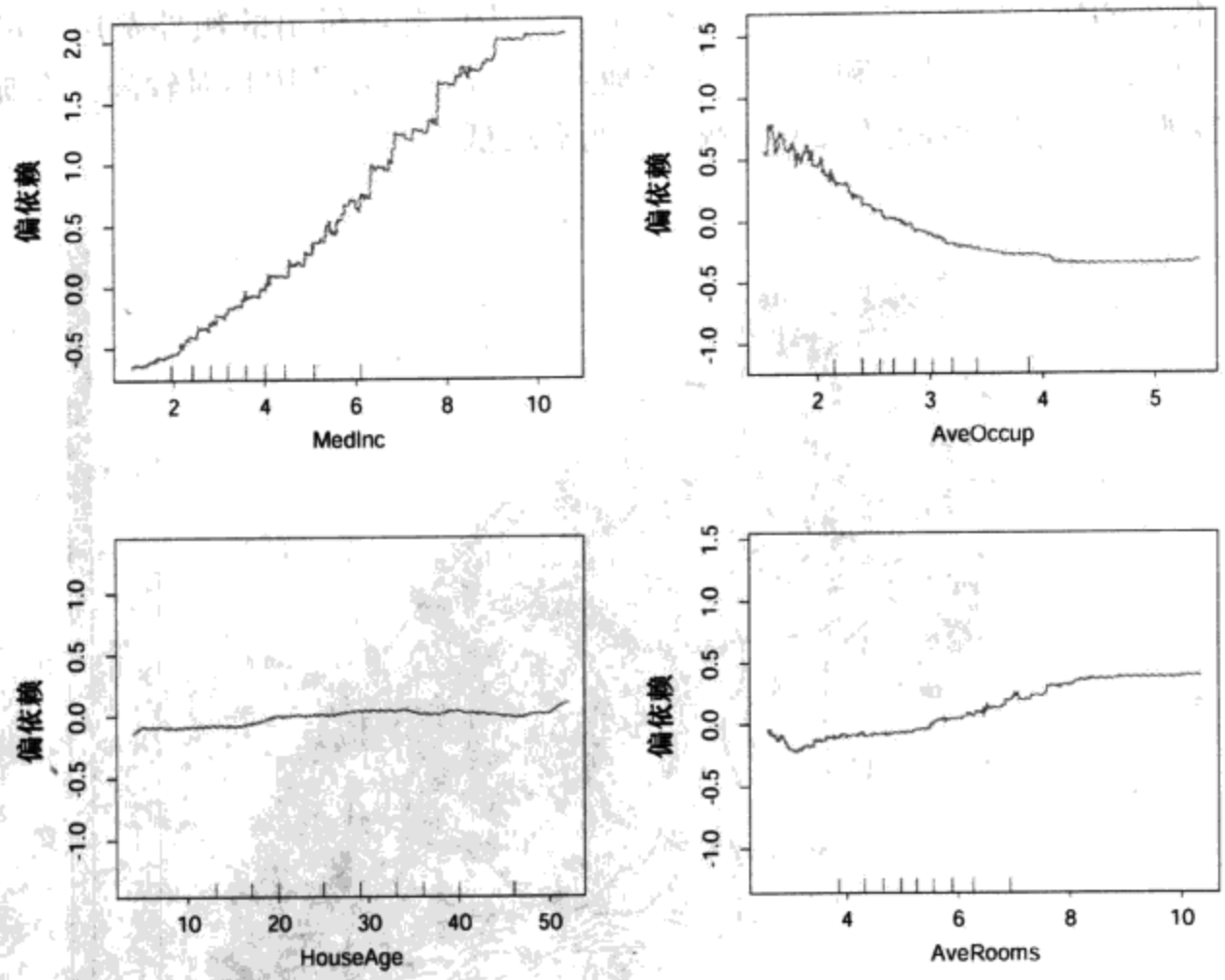


图 10.15　加利福尼亚住房数据房价对非定位变量的偏依赖

可以看出房价中值对房龄的偏依赖非常弱，这与房龄的重要性秩不一致（见图 10.14）。这表明，这个弱主效应可能会掩盖与其他变量较强的交互作用。图 10.16 显示了房价对两个变量房龄中值和平均居住率的联合值的偏依赖。这些两变量间的交互作用是明显的。对于平均居住率大于 2 的情况，房价几乎独立于房龄中值，而对于值小于 2 的情况，对房龄却有很强的依赖性。

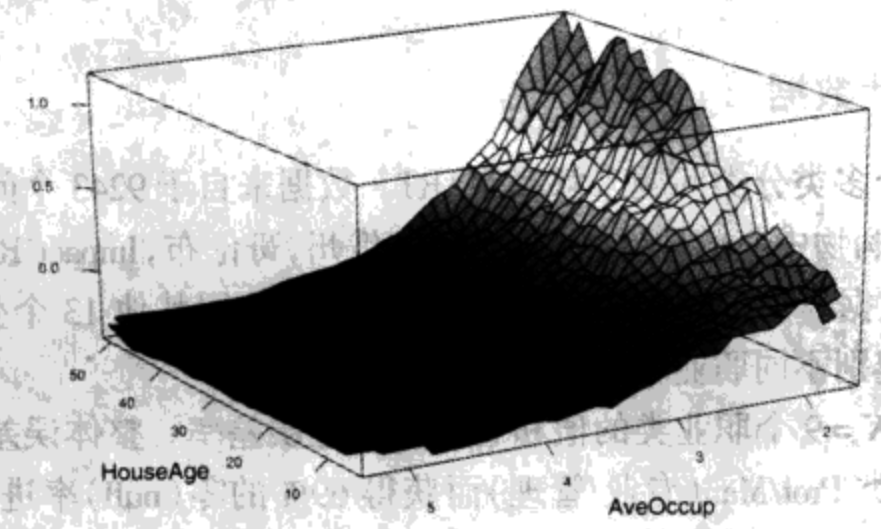


图 10.16　在中等房龄和平均面积上房价的偏依赖。在这些两变量间似乎存在着很强的交互（见彩页）

图 10.17 显示了双变量对经度、纬度的联合值偏依赖的等值线图。显然，房价中值对小区在加利福尼亚的位置有非常强的依赖性。注意，图 10.17 不是忽视其他预测子作用后（10.52）

房价与房屋位置的函数关系图。与所有偏依赖图一样，它表现了在考虑其他小区和房子属性的影响之后位置的作用(10.50)。它可以看做是个人为位置所付出的额外费用。可以看到在太平洋沿岸，特别是在海湾地区和洛杉矶－圣地亚哥地区，额外费用相对较高。在加利福尼亚北部、中部谷地和东南部沙漠地区，位置费用则明显较低。

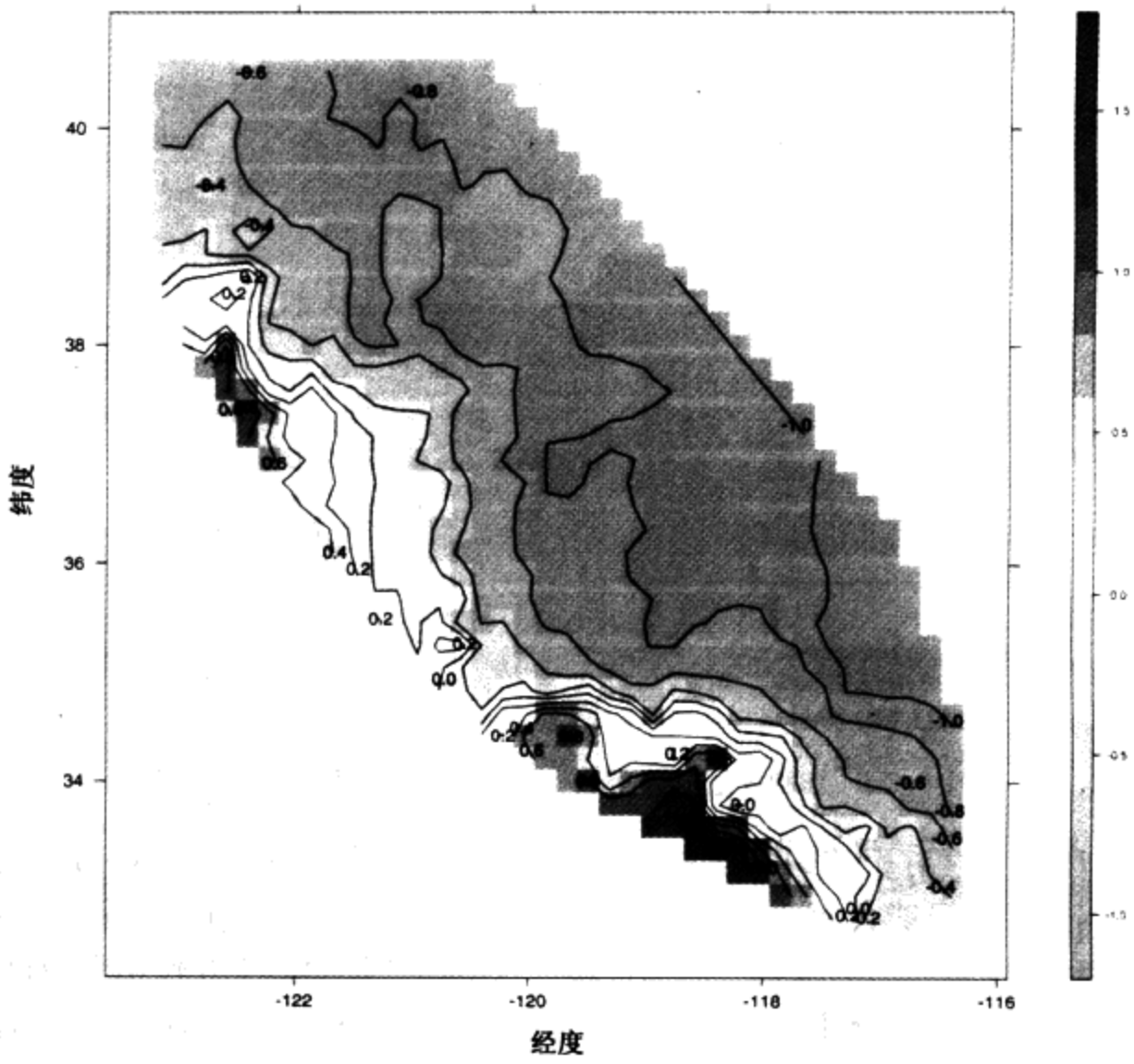


图 10.17 房价中值对加利福尼亚位置的偏依赖(见彩页)

10.14.2 人口统计数据

本节，我们就一个多类分类问题来解释 MART。数据来自于 9243 个问卷调查表，这些表格是由旧金山海湾地区购物中心的顾客填写的(俄亥俄州，哥伦布，Impact Resources 公司)。这些问题中有 14 项统计数据。为了解释清楚，我们的目标是使用其他 13 个变量作为预测子来预测职业，并由此确定辨别不同职业类的人口统计变量。

图 10.18 显示了 $K=9$ 个职业类的值和它们的相应误差率。整体误差率是 42.5%，可以将它与预测人数最多的类 Prof/Man(专业/管理)而获得 69% 的零(null)率进行比较。可以看到 4 个最好的预测类是：Retired(退休者)、Student(学生)、Prof/Man 和 Homemaker(家庭主妇)。

图 10.19 显示了作为所有类上的平均，预测子变量的相对重要性(10.49)。图 10.20 显示了对于 4 种最好预测的类的个体相对重要性的分布(10.48)。可以看到，对于不同的类，最相关的预测子通常是不同的。一个例外是 age(年龄)，对于预测 Retired、Student 和 Prof/Man，它是最相关的变量。

对于这三个类，图 10.21 显示了对 age 的对数几率偏依赖(10.55)。在每个相等的年龄区间上，横坐标值是按序编码的。可以看出，在考虑其他变量的贡献之后，对老年人 Retired 的几率较高，而对于学生情况则正相反。对于中年人是专业人员/管理人员的可能性最高。当然，这些结果并不令人吃惊。它们说明对每个类分别检查偏依赖性可以导致很实用的结果。

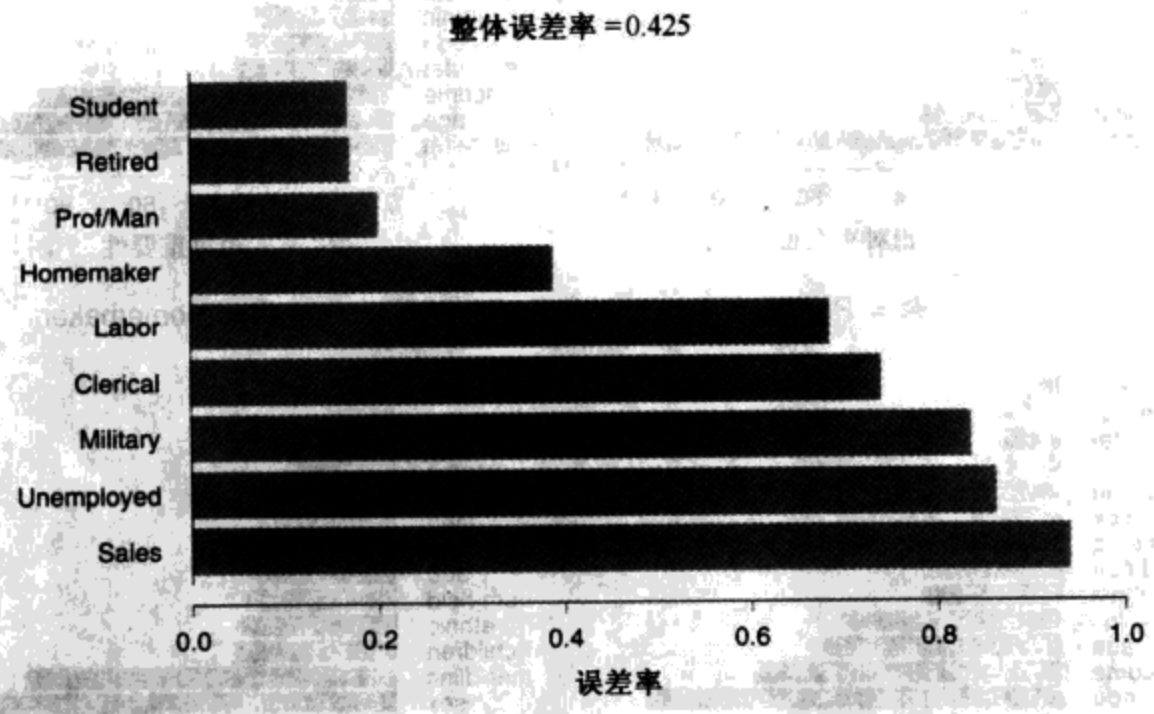


图 10.18　人口统计数据中每种职业的误差率

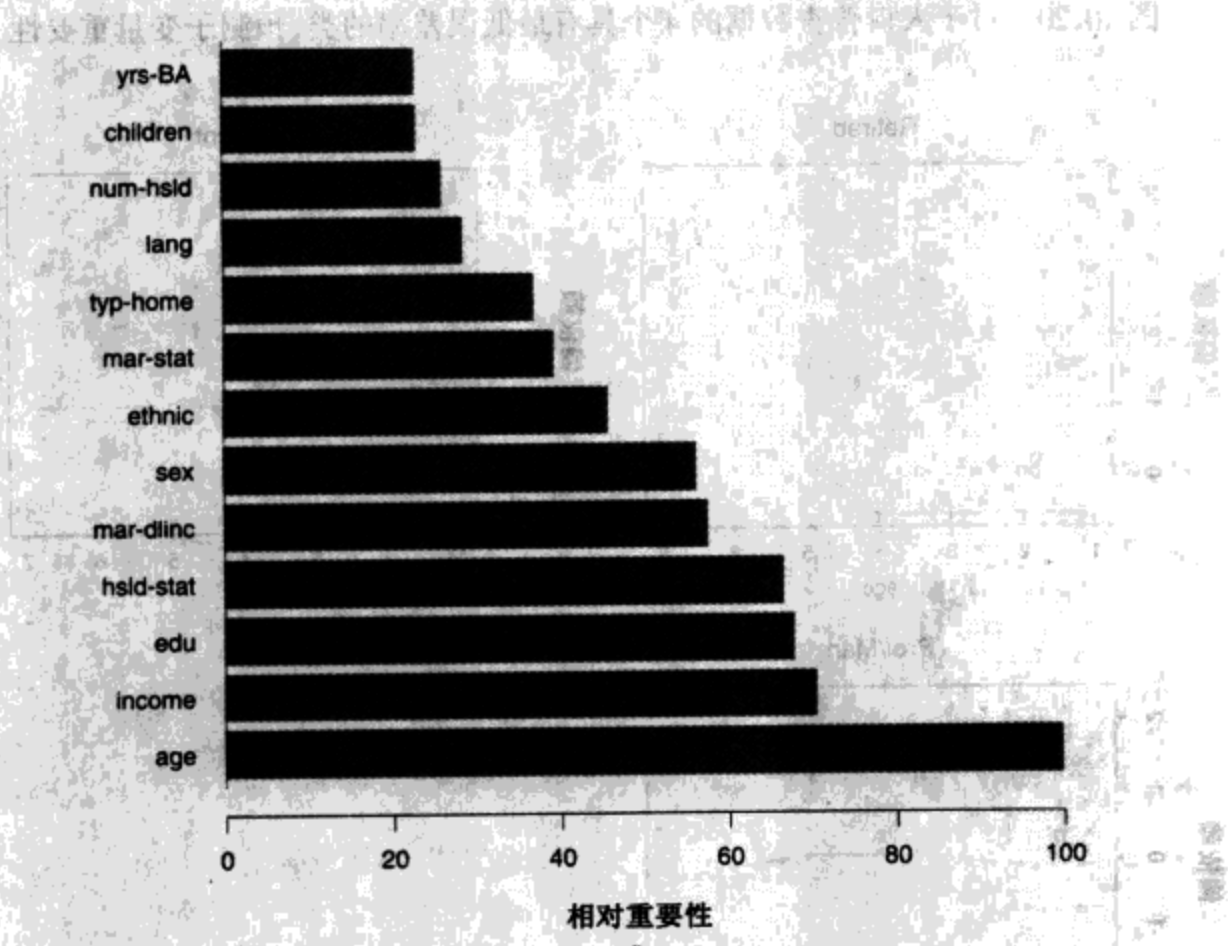


图 10.19　对于人口统计数据，作为所有类上的平均，预测子的相对重要性

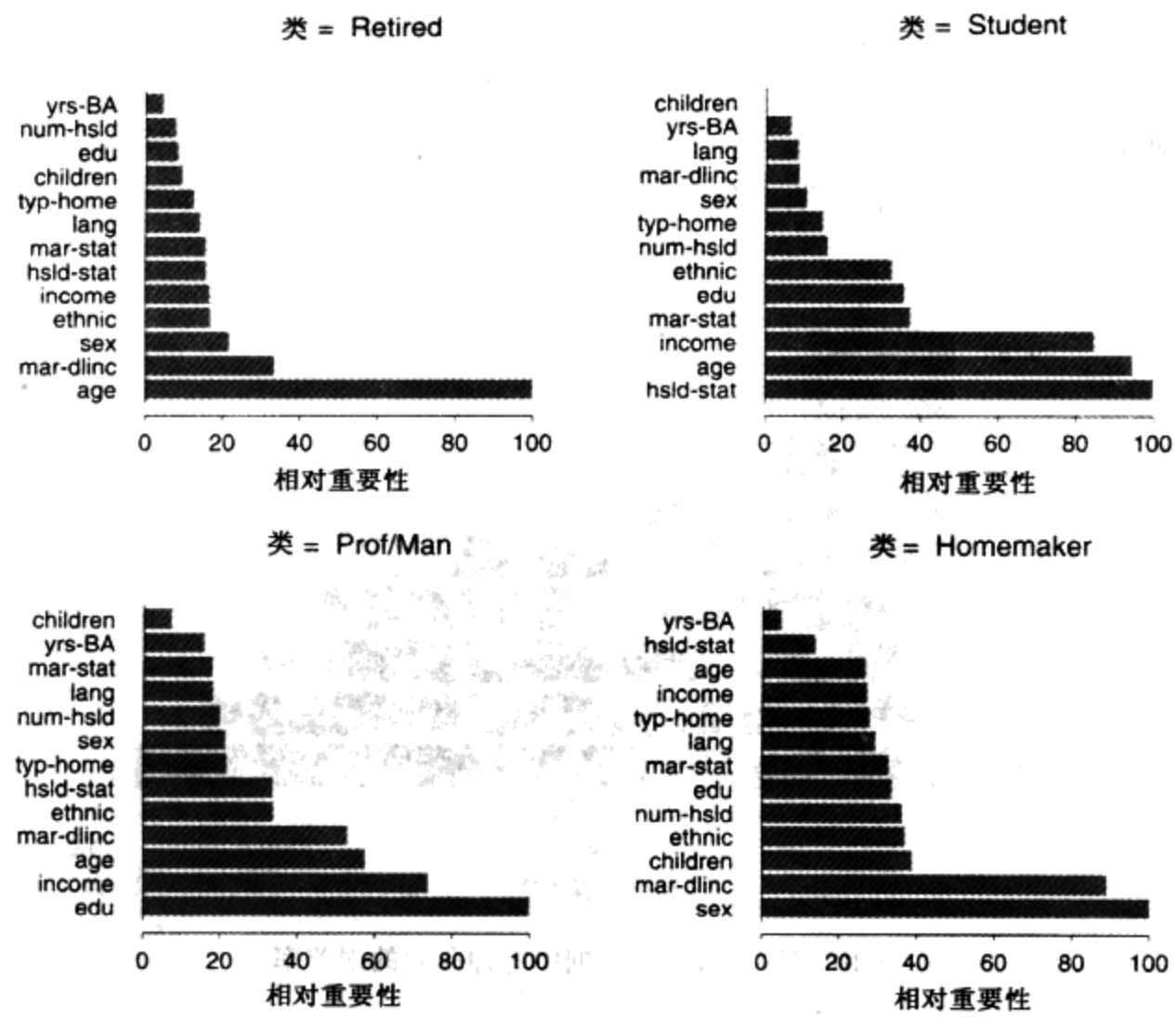


图 10.20 对于人口普查数据的 4 个具有最低误差率的类，预测子变量重要性

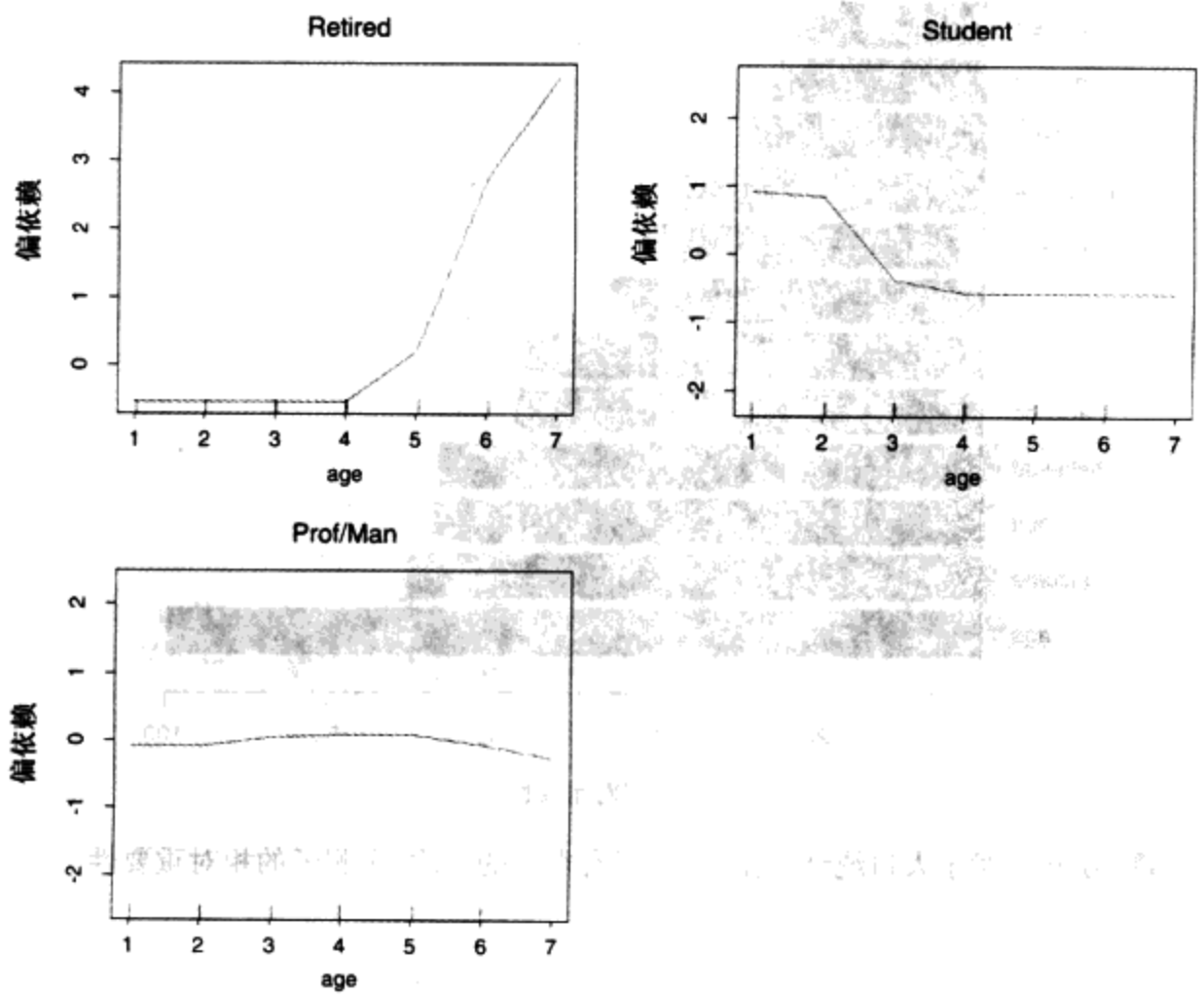


图 10.21 对人口普查数据，三种不同职业的几率对 age 的偏依赖

文献注释

Schapire(1990)在 PAC 学习框架下(Valiant,1984、Kearns 和 Vazirani,1994)开发了第一个简单的提升过程。Schapire 指出,在过滤后的输入数据流上,通过训练两个附加的分类器总能使弱学习器的性能有所改善。弱学习器是一个产生两个类的分类器算法,其性能可以(以较高的可能性)显著地好于抛硬币方法。在最初的 N 个训练点上学习一个初始的分类器 G_1 之后,

- G_2 在一个 N 个点的新样本上学习,N 个点的一半被 G_1 误分类。
- G_3 在 N 个点上学习,对于这 N 个点,G_1 和 G_2 不一致。
- 提升的分类器是 G_B = 多数表决(G_1, G_2, G_3)。

Schapire 的"弱学习能力的强壮性"理论证明了 G_B 的性能比 G_1 有所提高。

Freund(1995)提出了一种"多数提升"的变型,同时合并许多弱学习方法,并改进了 Schapire 的简单提升算法。支持这两种算法的理论需要弱学习器产生一个具有固定误差率的分类器。这导致了更加自适应且实用的 AdaBoost 算法(Freund 和 Schapire,1996a)及其派生形式,其中这种假设已被删去。

Freund 和 Schapire(1996a)、Schapire 和 Singer(1998)以泛化误差上界的形式提供一些理论以支持他们的算法。这一理论已由计算学习界进一步发展,起初是基于 PAC 学习概念的。其他试图解释提升算法的理论源自于游戏理论(Freund 和 Schapire, 1996b、Breiman, 1999、Breiman, 1998)和 VC 理论(Schapire 等人,1998)。这种与 AdaBoost 算法相关的上界和理论很有吸引力,但太宽松,没有实际价值。在实践中,提升产生的结果给人的印象比这些上界所蕴涵的更加深刻。Friedman 等(2000)和 Friedman(2001)形成了本章内容的基础。Friedman 等人(2000)从统计学角度分析 AdaBoost,导出了指数标准,并证明了它估计类概率的对数几率。他们提出加法器模型、适当大小的树和第 10.11 节的 ANOVA 表示,以及多类分对数公式。Friedman(2001)为分类和回归提出了梯度提升和收缩,而 Friedman(1999)探索了提升算法的随机变型。正如 Friedman 等人(2000)讨论所示的,目前仍然存在着有关提升算法的"怎样"和"为什么"的争论。

习题

10.1 推导 AdaBoost 中关于更新参数的式(10.12)。

10.2 证明结论(10.16),即 AdaBoost 标准的总体形式的极小值是对数几率的一半。

10.3 证明边缘平均(10.50)能够恢复加法和乘法函数(10.53)和(10.54),而条件期望(10.52)则不能。

10.4 (a) 编写一个程序,用树实现 AdaBoost 算法。

(b) 重新计算图 10.2 的例子,绘图表示训练误差和检验误差,并讨论它的行为。

(c) 考察使检验误差最终开始升高所需要的迭代次数。

(d) 修改这个例子的结构如下:定义两个类,在类 1 中,特征是 $X_1, X_2, \cdots, X_{10}$,它们都是标准的独立高斯变量。在类 2 中,其特征也是 $X_1, X_2, \cdots, X_{10}$,它们也是标准的独立

高斯变量，但限制条件是 $\sum_j X_j^2 > 12$。现在，两个类在特征空间中有明显的重叠。像在图 10.2 中一样，重复 AdaBoost 实验并讨论结果。

10.5 考虑 K-类问题，其中，如果观测 i 在类 k 中，则目标 y_{ik} 编码为 1，否则为 0。使用对称逻辑斯缔变换(10.55)作为损失函数，使用导致 MART 算法 10.3 的理由，导出算法 10.5。

算法 10.5 关于 K-类分类的 MART

1. 初始化 $f_{k0}(x)=0,\ k=1,2,\cdots,K$
2. 对于 $m=1$ 到 M：
   (a) 置

$$p_k(x)=\frac{e^{f_k(x)}}{\sum_{\ell=1}^{K} e^{f_\ell(x)}},\ k=1,2,\cdots,K$$

   (b) 对于 $k=1$ 到 K：
      i. 计算 $r_{ikm}=y_{ik}-p_k(x_i),\ i=1,2,\cdots,N$
      ii. 对目标 r_{ikm} 拟合一个回归树，$i=1,2,\cdots,N$，产生端区域 $R_{jkm},\ j=1,2,\cdots,J_m$
      iii. 计算

$$\gamma_{jkm}=\frac{K-1}{K}\frac{\sum_{x_i\in R_{jkm}} r_{ikm}}{\sum_{x_i\in R_{jkm}} |r_{ikm}|(1-|r_{ikm}|)},\ j=1,2,\cdots,J_m$$

      iv. 更新 $f_{km}(x)=f_{k,m-1}(x)+\sum_{j=1}^{J_m}\gamma_{jkm}I(x\in R_{jkm})$
3. 输出 $\hat{f}_k(x)=f_{kM}(x),\ k=1,2,\cdots,K$

第11章　神经网络

11.1　引言

本章，我们讲述一类学习方法，它们成长于不同的领域——统计学和人工智能，但都基于本质上相同的模型。这类方法的中心思想是：提取输入的线性组合作为导出特征，然后将目标作为这些特征的非线性函数建模。其结果是强有力的学习方法，被广泛地应用于许多领域。我们首先讨论投影寻踪模型，它是在半参数统计和光滑领域发展起来的。本章的其余部分将讨论神经网络模型。

11.2　投影寻踪回归

像一般的有指导学习问题一样，假定我们有一个含 p 个分量的输入向量 X 和一个目标 Y。令 $\omega_m, m=1,2,\cdots,M$ 是未知参数的单位 p 向量。投影寻踪回归(projection pursuit regression, PPR)模型具有如下形式：

$$f(X)=\sum_{m=1}^{M} g_m(\omega_m^T X) \tag{11.1}$$

这是一个加法模型，但它是导出特征 $V_m=\omega_m^T X$ 上而不是输入本身上的加法模型。函数 g_m 并没有指定，而是利用一些灵活的光滑方法与方向 ω_m 一起估计(见下面)。

函数 $g_m(\omega_m^T X)$ 称为 $\mathbb{R}^p$ 上的岭函数(ridge function)，它仅在由向量 ω_m 定义的方向上变化。标量变量 $V_m=\omega_m^T X$ 是 X 到单位向量 ω_m 上的投影，并且我们寻找 ω_m 使得模型能很好地拟合，因此称之为“投影寻踪”。图 11.1 显示了一些岭函数的例子。在左图的例子中，$\omega=(1/\sqrt{2})(1,1)^T$，它使得函数只在方向 X_1+X_2 上变化。在右图的例子中，$\omega=(1,0)$。

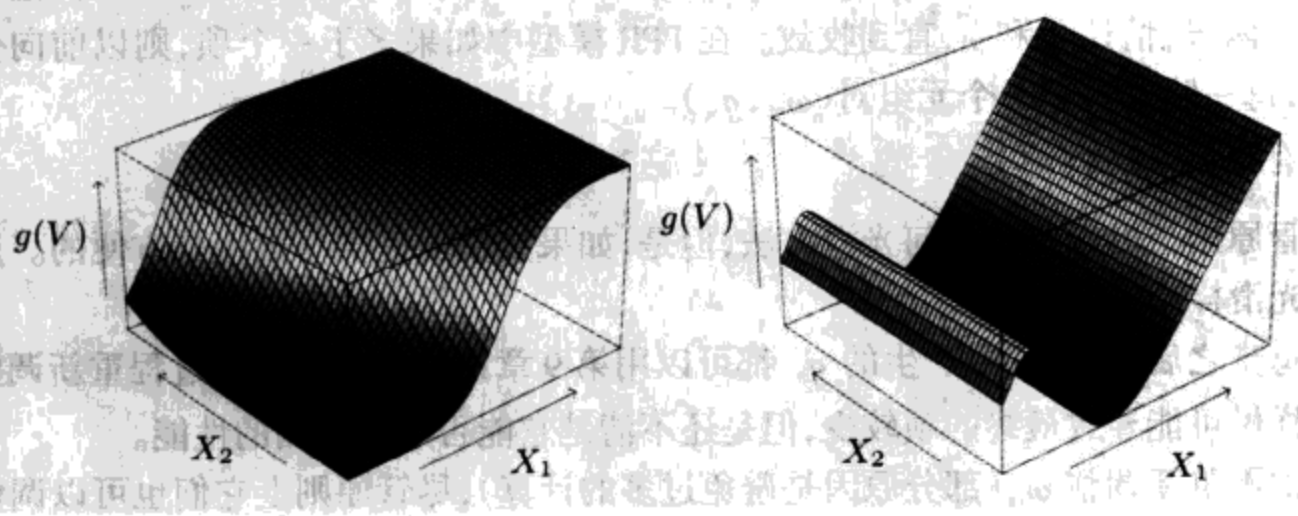


图 11.1　两个岭函数的透视图。左：$g(V)=1/[1+\exp(-5(V-0.5))]$，其中 $V=(X_1+X_2)/\sqrt{2}$。右：$g(V)=(V+0.1)\sin(1/(V/3+0.1))$，其中 $V=X_1$(见彩页)

PPR 模型(11.1)非常一般,因为形成线性组合的非线性函数的操作产生了使人吃惊的一大类模型。例如,积 $X_1 \cdot X_2$ 可以写成$[(X_1 + X_2)^2 - (X_1 - X_2)^2]/4$,并且高阶积也可以类似地表示。

事实上,只要 M 任意大,对于适当选择的 g_m,PPR 模型就可以任意好地逼近 $\mathbb{R}^p$ 中的任何连续函数。这类模型称为普适逼近子(universal approximator)。然而,这种通用性有一定的代价。拟合模型的解释通常很困难,因为每个输入都是以复杂多面的方式进入模型的。这样,PPR 模型对预测非常有用,而对产生易理解模型就不是很有用。$M = 1$ 模型是一个例外,它在计量经济学中称为单指标模型(single index model)。它比线性回归模型略微一般一些,并提供类似的解释。

给定训练数据(x_i, y_i), $i = 1, 2, \cdots, N$,怎样拟合一个 PPR 模型呢?我们在函数 g_m 和方向向量 ω_m, $m = 1, 2, \cdots, M$ 上寻找误差函数

$$\sum_{i=1}^{N}\left[y_i - \sum_{m=1}^{M} g_m(\omega_m^T x_i)\right]^2 \tag{11.2}$$

的近似极小值。与其他光滑问题一样,我们需要显式或隐式地对 g_m 施加一些复杂约束,以避免过分拟合。

考虑只有一个项($M = 1$,并且省略下标)的情形。给定方向向量 ω,我们形成导出变量 $v_i = \omega^T x_i$。于是,得到一个一维光滑问题,并且可以使用任意散点图光滑子,如光滑样条,以获得 g 的一个估计。

另一方面,给定 g,我们希望在 ω 上对式(11.2)极小化。对于该任务,高斯 - 牛顿搜索是方便的。这是一个拟牛顿方法,其中涉及 g 的二阶导数的 Hessian 部分被舍弃。它可以按如下方法简单地导出。令 ω_{old}是 ω 的当前估计,我们有:

$$g(\omega^T x_i) \approx g(\omega_{\text{old}}^T x_i) + g'(\omega_{\text{old}}^T x_i)(\omega - \omega_{\text{old}})^T x_i \tag{11.3}$$

给出:

$$\sum_{i=1}^{N}\left[y_i - g(\omega^T x_i)\right]^2 \approx \sum_{i=1}^{N} g'(\omega_{\text{old}}^T x_i)^2\left[\left(\omega_{\text{old}}^T x_i + \frac{y_i - g(\omega_{\text{old}}^T x_i)}{g'(\omega_{\text{old}}^T x_i)}\right) - \omega^T x_i\right]^2 \tag{11.4}$$

为了极小化右端,我们执行最小二乘方回归,输入 x_i 上的目标是$\omega_{\text{old}}^T x_i + (y_i - g(\omega_{\text{old}}^T x_i))/g'(\omega_{\text{old}}^T x_i)$,权是 $g'(\omega_{\text{old}}^T x_i)^2$ 并且没有截距(偏倚)项。它产生了更新的系数向量 ω_{new}。

重复这两步,估计 g 和 ω,直到收敛。在 PPR 模型中如果多于一个项,则以前向分步方式构建模型,每一阶段增加一个元组对(ω_m, g_m)。

这里有许多实现的细节:

- 尽管原则上可以使用任何光滑方法,但是,如果方法能提供导数将是方便的。局部回归和光滑样条也很方便。
- 在每步之后,前面步骤产生的 g_m 都可以用第 9 章讨论的反向拟合过程重新调整。虽然这样做可能导致最终的项较少,但是还不清楚它能否改进预测的性能。
- 通常不重新调整 ω_m(部分原因是避免过多的计算),尽管原则上它们也可以调整。
- 项数 M 通常作为前向分步策略的一部分来估计。当下一项不能明显改进模型的拟合时,模型构造过程将停止。交叉验证也可以用于确定 M。

还有许多其他应用，如密度估计（Friedman 等人，1984、Friedman，1987），可以使用投影寻踪思想。特别地，可以参考第 14.6 节 ICA 的讨论和它与试探性投影寻踪之间的联系。然而，投影寻踪回归模型并没有广泛应用于统计学领域，或许因为在它出现的时候（1981 年），它对计算的要求已经超出当时计算机的能力。但它确实代表了重要的智能进展，这已在神经网络领域得到进一步发展。神经网络是本章余下部分的主题。

11.3　神经网络

术语神经网络（neural network）已经逐步演变，包括一大类模型和学习方法。这里我们讲述一种最广泛使用的“香草”（vanilla）神经网络，有时称单隐藏层后向传播网络，或单层感知器（perceptron）。围绕神经网络有很多虚假宣传，使得它们看起来神奇而不可思议。本节，随着我们揭开它的神秘面纱，就会发现它们只不过是非线性统计模型，与上面讨论过的投影寻踪模型非常相似。

神经网络就是一个两阶段回归或分类模型，通常用如图 11.2 所示的网络图（network diagram）表示。网络应用于回归或者分类。对于回归，通常 $K=1$，而且只在顶端有一个输出单元 Y_1。然而，这些网络可以无缝的方式处理多元定量响应，所以我们将处理一般的情形。

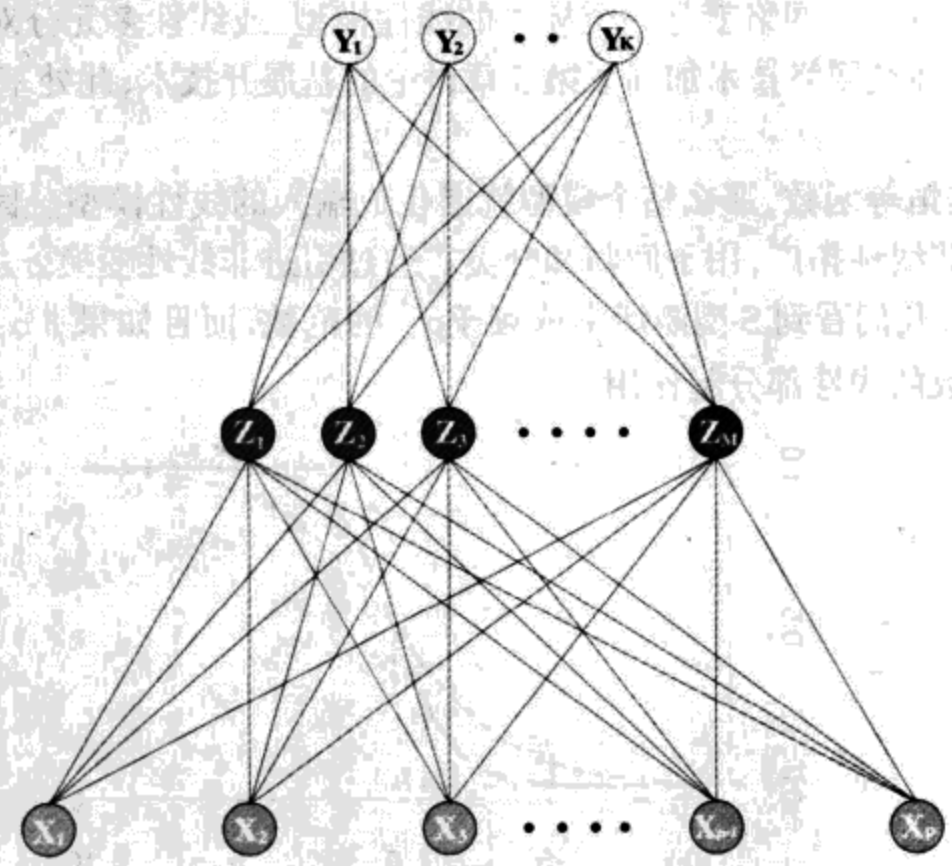


图 11.2　单隐藏层、前馈神经网络示意图

对于 K-类分类，顶端有 K 个单元，第 k 个单元对类 k 的概率建模。有 K 个目标度量 Y_k，$k=1,\cdots,K$，每个被编码为第 k 个类的 0-1 变量。

导出特征 Z_m 由输入的线性组合创建，然后，目标 Y_k 作为 Z_m 的线性组合函数建模，

$$\begin{aligned} Z_m &= \sigma(\alpha_{0m} + \alpha_m^T X),\ m = 1,\cdots,M \\ T_k &= \beta_{0k} + \beta_k^T Z,\ k = 1,\cdots,K \\ f_k(X) &= g_k(T),\ k = 1,\cdots,K \end{aligned} \tag{11.5}$$

其中 $Z=(Z_1,Z_2,\cdots,Z_M)$，且 $T=(T_1,T_2,\cdots,T_K)$。

激活函数 $\sigma(v)$通常选取 S 型函数 $\sigma(v)=1/(1+e^{-v})$，$1/(1+e^{-v})$的图形参见第 11.3 节。有时，高斯径向基函数(见第 6 章)用于 $\sigma(v)$，产生径向基函数网络(radial basis function network)。

有时，如图 11.2 所示的神经网络图与附加的、馈入隐藏层和输出层每个单元的偏置单元一起绘制。考虑将常量"1"作为一个附加的输入特征，在模型(11.5)中，该偏置单元捕获截距 α_{0m}和 β_{0k}。

输出函数 $g_k(T)$允许输出向量 T 的一个最终变换。对于回归，我们通常选取恒等函数 $g_k(T)=T_k$。早期的 K-类分类也使用恒等函数，但后来放弃了，而使用 softmax 函数：

$$g_k(T)=\frac{e^{T_k}}{\sum_{\ell=1}^{K}e^{T_\ell}} \tag{11.6}$$

当然，这正是用于多元分对数模型(见第 4.4 节)中的变换，并产生正和为 1 的正估计。在第 4.2 节我们讨论了线性激活函数的其他问题，特别是潜在而严重的屏蔽作用。

在网络中间的单元，计算导出特征 Z_m，称为隐藏单元(hidden unit)，因为值 Z_m 不是直接观测到的。一般地，可以有多个隐藏层，如同在本章末尾例子中讨论的那样。可以把 Z_m 看做初始输入 X 的基展开，则神经网络就是一个标准的线性模型，或线性多元分对数模型，将这些变换作为输入。然而，神经网络技术加强了第 5 章讨论的基展开技术；此处基函数的参数从数据中学习。

注意，如果 σ 是恒等函数，那么整个模型就退化成输入的线性模型。因此，神经网络可以看做是线性模型的非线性拓广，用于回归和分类。通过引进非线性变换 σ，极大扩充了线性模型类。在图 11.3 中，我们看到 S 型激活率依赖于α_m 的范数，而且如果 $\|\alpha_m\|$ 很小，则单元实际将会在其激活函数的线性部分起作用。

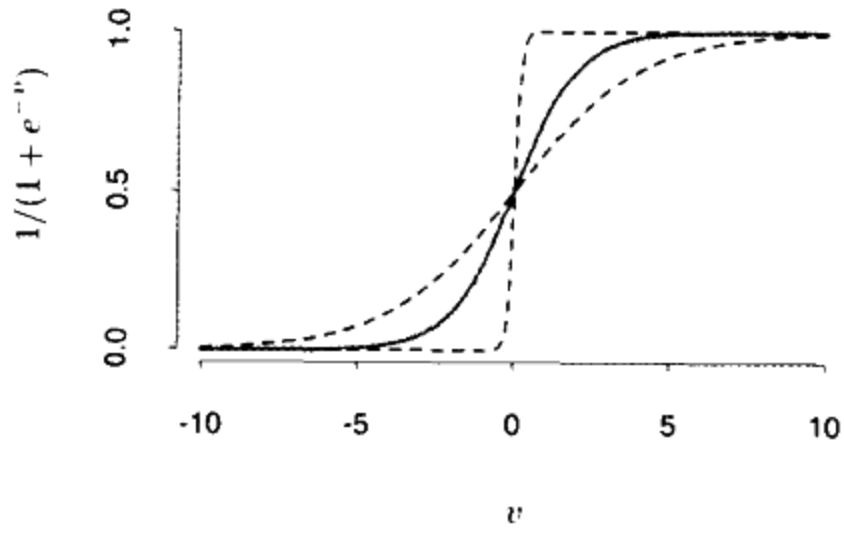


图 11.3 S 型函数 $\sigma(v)=1/(1+\exp(-v))$(红色曲线)的平面图。通常用于神经网络的隐藏层，所包含的是 $s=\frac{1}{2}$时的$\sigma(sv)$(蓝色曲线)和$s=10$时的$\sigma(sv)$(紫色曲线)。缩放参数s控制激活率，我们可以看到大的s相当于在$v=0$处的硬激活。注意到$\sigma(s(v-v_0))$将激活阈值从0移动到v_0(见彩页)

还要注意，具有一个隐藏层的神经网络模型与上面讨论的投影寻踪模型具有完全相同的形式。其不同点在于 PPR 模型使用了非参数函数 $g_m(v)$，而神经网络使用了基于 $\sigma(v)$的简单

得多的函数，其变元具有三个自由参数。详细地说，把神经网络模型看做 PPR 模型，我们有：

$$\begin{aligned} g_m(\omega_m^T X) &= \beta_m \sigma(\alpha_{0m} + \alpha_m^T X) \\ &= \beta_m \sigma(\alpha_{0m} + \|\alpha_m\|(\omega_m^T X)) \end{aligned} \tag{11.7}$$

其中，$\omega_m = \alpha_m / \|\alpha_m\|$ 是第 m 个单元向量。由于 $\sigma_{\beta,\alpha_0,s}(v) = \beta\sigma(\alpha_0 + sv)$ 更一般的非参数函数 $g(v)$ 的复杂度低，因此神经网络可能使用 20 或 100 个这样的函数就不足为奇了，而 PPR 模型典型地使用较少的项（例如，$M = 5$ 或 10）。

最后，我们注意到名称“神经网络”源自于最初开发的人脑模型，每个单元表示一个神经元，而连接（见图 11.2 中的链）表示神经元的突触。在早期模型中，当传到单元的总信号超过一定的阈值时神经元被激活。在上述模型中，这相当于对 $\sigma(Z)$ 和 $g_m(T)$ 使用阶梯函数。后来，人们意识到神经网络是一种非线性统计建模的有用工具，而对于这一目的，阶梯函数不够光滑，难以优化。因此，阶梯函数被光滑阈函数——图 11.3 中的 S 型函数取代。

11.4　拟合神经网络

神经网络模型具有未知参数，通常称为权（weight），而我们寻找它们的值，使得模型能很好地拟合训练数据。用 θ 表示权的全集，它包括：

$$\begin{aligned} &\{\alpha_{0m}, \alpha_m;\ m = 1, 2, \cdots, M\}\ \ M(p+1)\ \text{权} \\ &\{\beta_{0k}, \beta_k;\ k = 1, 2, \cdots, K\}\ \ K(M+1)\ \text{权} \end{aligned} \tag{11.8}$$

对于回归，我们使用误差的平方和作为拟合（误差函数）度量：

$$R(\theta) = \sum_{k=1}^{K}\sum_{i=1}^{N}(y_{ik} - f_k(x_i))^2 \tag{11.9}$$

对于分类，使用平方误差或互熵（散离）：

$$R(\theta) = -\sum_{i=1}^{N}\sum_{k=1}^{K} y_{ik} \log f_k(x_i) \tag{11.10}$$

并且相应的分类器是 $G(x) = \text{argmax}_k f_k(x)$。使用 softmax 激活函数和互熵误差函数，神经网络模型在隐藏层确实是一个线性逻辑斯缔回归模型，而且全部参数都是用极大似然来估计的。

通常，我们不需要 $R(\theta)$ 的全局极小化，因为这很可能是一个过分拟合解。替换地，需要某种正则化：这可以通过罚项直接实现，或者通过提前停止而间接实现。细节将在下一节给出。

极小化 $R(\theta)$ 的一般方法是梯度下降，在此情况下称做反向传播（back-propagation）。由于模型的复合形式，梯度可以很容易使用微分法的链规则导出。这可以用对网络的前向和后向搜索来计算，只需要维护局部于每个单元的量。

下面介绍是平方误差损失的反向传播细节。令 $z_{mi} = \sigma(\alpha_{0m} + \alpha_m^T x_i)$，由式（11.5）令 $z_i = (z_{1i}, z_{2i}, \cdots, z_{Mi})$。那么有：

$$\begin{aligned} R(\theta) &\equiv \sum_{i=1}^{N} R_i \\ &= \sum_{i=1}^{N}\sum_{k=1}^{K}(y_{ik} - f_k(x_i))^2 \end{aligned} \tag{11.11}$$

具有导函数：

$$\begin{aligned}\frac{\partial R_i}{\partial \beta_{km}} &= -2(y_{ik} - f_k(x_i))g_k'(\beta_k^T z_i)z_{mi} \\ \frac{\partial R_i}{\partial \alpha_{m\ell}} &= -\sum_{k=1}^{K} 2(y_{ik} - f_k(x_i))g_k'(\beta_k^T z_i)\beta_{km}\sigma'(\alpha_m^T x_i)x_{i\ell}\end{aligned} \tag{11.12}$$

给定这些导函数，梯度下降更新在第$(r+1)$次迭代具有如下形式：

$$\begin{aligned}\beta_{km}^{(r+1)} &= \beta_{km}^{(r)} - \gamma_r \sum_{i=1}^{N} \frac{\partial R_i}{\partial \beta_{km}^{(r)}} \\ \alpha_{m\ell}^{(r+1)} &= \alpha_{m\ell}^{(r)} - \gamma_r \sum_{i=1}^{N} \frac{\partial R_i}{\partial \alpha_{m\ell}^{(r)}}\end{aligned} \tag{11.13}$$

其中，γ_r 是学习率，在下面讨论。

现在，将式(11.12)写成：

$$\begin{aligned}\frac{\partial R_i}{\partial \beta_{km}} &= \delta_{ki} z_{mi} \\ \frac{\partial R_i}{\partial \alpha_{m\ell}} &= s_{mi} x_{i\ell}\end{aligned} \tag{11.14}$$

量 δ_{ki} 和 s_{mi} 分别是当前模型的输出层和隐藏层的“误差”。由它们的定义，这些误差满足：

$$s_{mi} = \sigma'(\alpha_m^T x_i)\sum_{k=1}^{K} \beta_{km}\delta_{ki} \tag{11.15}$$

称做反向传播方程(back-propagation equation)。使用它，式(11.13)中的更新可以用一个两次传递算法实现。在前向传递(forward pass)时，固定当前权值，预测值 $\hat{f}_k(x_i)$ 用式(11.5)计算。在后向传递(backward pass)时，计算误差 δ_{ki}，然后由式(11.5)后向传播得到误差 s_{mi}。然后，使用两个误差集，通过式(11.14)计算式(11.13)中更新的梯度。

这个两次传递过程就是反向传播，也称 δ 规则(Widrow 和 Hoff, 1960)。互熵的计算分量与平方和误差函数具有相同的形式，在习题 11.3 中推导。

反向传播的优点在于它的简单性和局部特性。在反向传播算法中，每个隐藏单元只向或从共享连接的单元传送或接收信息。因此，它可以在并行体系结构计算机上有效地实现。

式(11.13)中的更新就是一种批学习(batch learning)，其参数更新是全部训练实例上的更新之和。学习也可以在线执行——一次处理一个观测，在每个训练实例之后更新梯度，并循环处理训练实例多次。在这种情况下，式(11.13)中的和被一个简单被加数所取代。一个训练周期(training epoch)是指对整个训练集的一次扫描。在线训练允许网络处理很大的训练集，并随新观测的到来而更新权值。

对于批学习，学习率 γ_r 通常取常数，也可以在每次更新时通过极小化误差函数的线搜索来优化。使用在线学习，γ_r 应随迭代次数 $r \to \infty$ 而递减到零。这种学习是一种随机逼近(stochastic approximation)形式(Robbins 和 Munro, 1951)；如果 $\gamma_r \to 0$，$\sum_r \gamma_r = \infty$，且 $\sum_r \gamma_r^2 < \infty$(例如，被 $\gamma_r = 1/r$ 满足)，该领域的结果保证其收敛性。

反向传播可能很慢，并且由于这个原因它不是通常的选择。诸如牛顿方法这样的二阶技术在这里就不大吸引人，因为 R 的二阶导数矩阵(Hessian 矩阵)可能很大。关于拟合较好的方法包括共轭梯度和变量度量方法。这些方法避免了二阶导数矩阵的显式计算，且仍然保持快速收敛。

11.5　训练神经网络的一些问题

训练神经网络确实是一门艺术。模型通常是过分参数化的，而且如果不遵循某些指导性准则，优化问题则是非凸的和不稳定的。本节，我们概述一些重要问题。

11.5.1　初始值

注意，如果权值接近于 0，则 S 型函数(见图 11.3)的运算部分大致是线性的，从而神经网络退化为近似线性的模型(见习题 11.2)。通常，初始权值取接近于 0 的随机值。因此，开始模型接近于线性的，并且随权值的增加而变成非线性的。每个单元在需要的地方对方向局部化，并引进非线性特征。使用恰为 0 的权值导致 0 导数和良好的对称性，且算法永远不会前进。相反，以大权值开始常常导致很差的解。

11.5.2　过分拟合

神经网络常常有太多的权值，而且在 R 全局极小值处将过分拟合数据。在神经网络发展初期，无论是由于设计还是偶然的原因，一种提前终止规则可以用来避免过分拟合。在这里，我们只是短暂地训练模型，并且在到达全局极小值之前完全停下来。由于权值开始于一个高度正则化的线性解，这有将最终模型向线性模型收缩的作用。验证数据集对确定何时停止是有用的，因为我们期望停止时验证误差开始增加。

一种更直接的正则化方法是权衰减(weight decay)，类似于用于线性模型的岭回归(见第 3.4.3 节)。我们将一个罚添加到误差函数 $R(\theta)+\lambda J(\theta)$ 中，其中：

$$J(\theta)=\sum_{km}\beta_{km}^2+\sum_{m\ell}\alpha_{m\ell}^2 \tag{11.16}$$

而 $\lambda\geqslant 0$ 是调整参数。较大的 λ 值趋向于将权值收缩到 0；典型地，交叉验证用于估计 λ。罚的作用是简单地将项 $2\beta_{km}$ 和 $2\alpha_{m\ell}$ 添加到各自的梯度表达式(11.13)中。其他形式的罚已被提出来，例如：

$$J(\theta)=\sum_{km}\frac{\beta_{km}^2}{1+\beta_{km}^2}+\sum_{m\ell}\frac{\alpha_{m\ell}^2}{1+\alpha_{m\ell}^2} \tag{11.17}$$

是权重消除(weight elimination)罚。与式(11.16)相比，它对较小的权值收缩得更多。

图 11.4 显示了对第 2 章的混合例子训练一个具有 10 个隐藏单元的神经网络的结果，包括无权衰减(上图)和有权衰减(下图)的情况。权衰减明显提高了预测性能。图 11.5 显示训练估计权值的热度图(其灰度版本叫做 Hinton 图)。我们看到权衰变已经在两个层上降低了权值：结果权完全均匀地分布在 10 个隐藏的单元上。

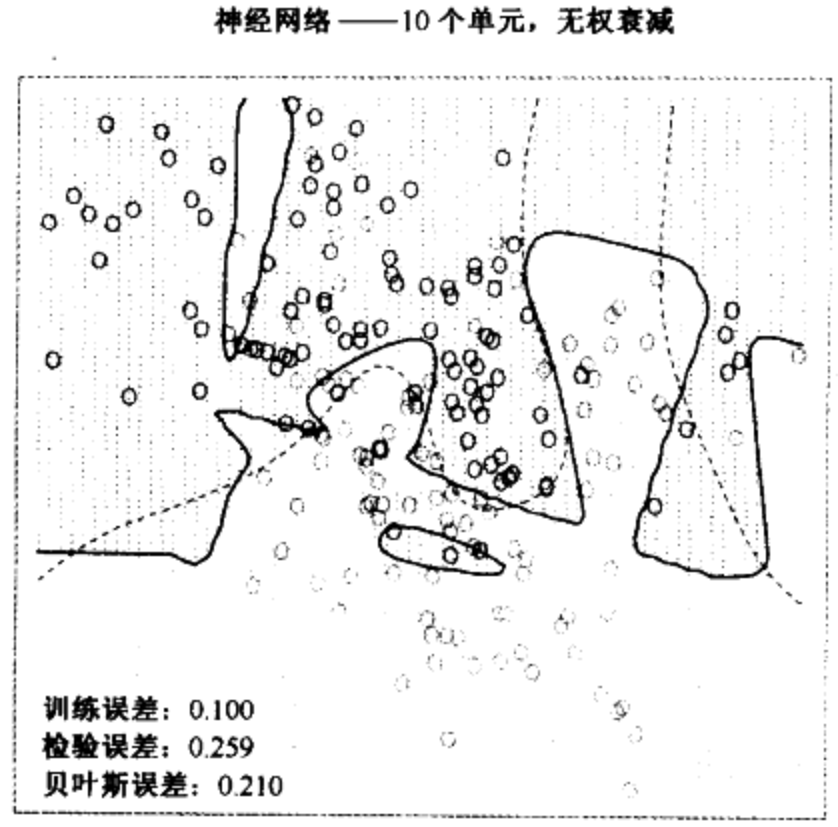


图 11.4 第 2 章混合例子上的神经网络。上图没有使用权衰减，并过分拟合训练数据。下图使用权衰减，接近于贝叶斯误差率(紫色虚边界)。两者皆使用了softmax激活函数和互熵误差(见彩页)

11.5.3 输入的定标

由于输入的定标决定底层权值的有效定标，它可能对最终解的质量有很大影响。开始，最好对所有输入进行标准化，使之具有均值 0 和标准差 1。这可以保证所有输入在正则化过程中被平等地处理，而且允许为随机初始权值选择一个有意义的值域。使用标准化输入，通常在值域$[-0.7, +0.7]$上随机地取均匀的权值。

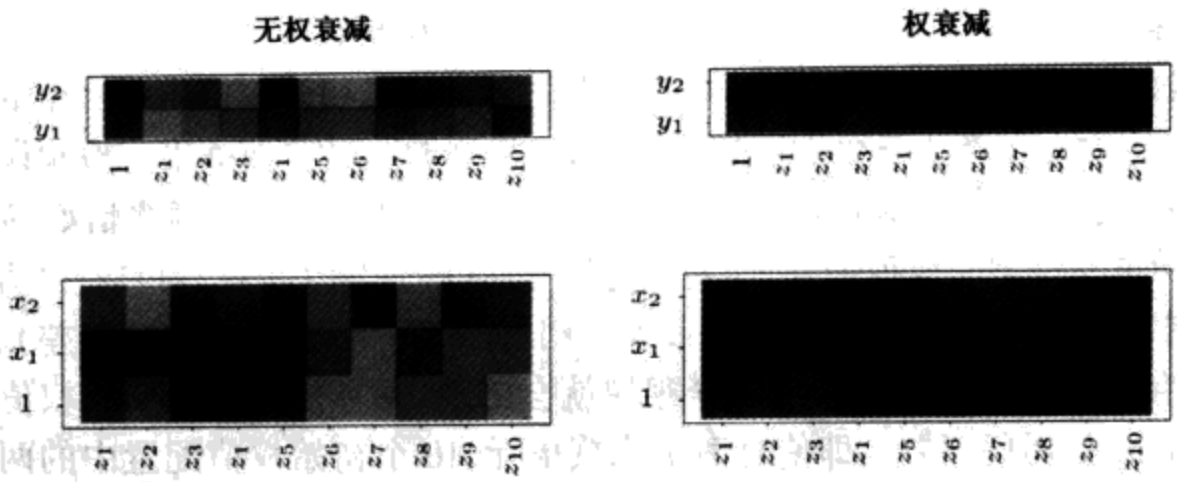


图 11.5　图 11.4 神经网络训练估计权值的热度图。显示的范围由鲜绿色(负的)到鲜红色(正的)(见彩页)

11.5.4　隐藏单元和层的数目

一般来说,隐藏单元过多比过少好。隐藏单元过少,模型可能不具有足够的灵活性来捕获数据中非线性特征。隐藏单元过多,如果进行合适的规范化,则额外的权可以收缩到 0。典型地,隐藏单元的数量一般在 5 到 100 之间,随输入的数量和训练实例的数量而增加。最普通的方法就是取合理大的单元数量,并用正则化训练它们。有些研究者用交叉验证来估计最优数量,但如果用交叉验证来估计正则化参数,似乎不必要。隐藏层数量的选择由背景知识和实验来指导。每层抽取回归或分类的输入特征。多个隐藏层的使用允许在解的不同层上有分层特征结构。多层有效使用的例子在第 11.6 节中给出。

11.5.5　多极小值

误差函数 $R(\theta)$是非凸的,拥有许多局部极小值。这样,得到的最终解在一定程度上依赖于初始权值的选择。我们至少需要试一定数量的随机初始配置,并选择产生最低(罚)误差的解。或许更好的方法是取网络集的平均预测作为最终预测(Ripley 1996)。对权求平均会更合适,因为模型的非线性特征意味着平均解可能是很差的。另一种方法是通过装袋方法,它对训练数据的随机扰动版本训练的网络预测求平均。第 8.7 节讨论过这些内容。

11.6　例:模拟数据

由两个加法误差模型产生数据 $Y = f(X) + \varepsilon$:

$$\text{S 型函数之和}: Y = \sigma(a_1^T X) + \sigma(a_2^T X) + \varepsilon_1$$

$$\text{径向模型}: Y = \prod_{m=1}^{10} \phi(X_m) + \varepsilon_2$$

这里,$X = (X_1, X_2, \cdots, X_p)$,每个 X_j 是一个标准高斯变量,第一个模型中,$p = 2$;在第二个模型中,$p = 10$。

对于 S 型模型,$a_1 = (3,3)$,$a_2 = (3,-3)$;对于径向模型,$\phi(t) = (1/2\pi)^{1/2}\exp(-t^2/2)$。$\varepsilon_1$ 和 ε_2 都是高斯误差,选择方差使得信噪比

$$\frac{\text{Var}(\text{E}(Y|X))}{\text{Var}(Y - \text{E}(Y|X))} = \frac{\text{Var}(f(X))}{\text{Var}(\varepsilon)} \tag{11.18}$$

在两个模型中都为 4。取大小为 100 的训练样本和大小为 10 000 的检验样本。我们拟合具有权衰减和不同数量的隐藏单元的神经网络，并记录 10 个随机初始权值中每一个的平均检验误差 $E_{Test}(Y-\hat{f}(X))^2$。仅产生一个训练集，但结果是典型的"平均"训练集。检验误差如图 11.6 所示。注意，0 个隐藏单元的模型涉及到线性最小二乘方回归。神经网络恰好适合于 S 型模型之和，并且两单元模型完成得最好，误差接近贝叶斯(误差)率(注意，对具有平方误差的回归来说，贝叶斯率就是误差方差；图中，我们报告了相对于贝叶斯误差的检验误差)。然而，需要注意的是，使用更多的隐藏单元，过分拟合很快就会出现，并且使用某些初始权值的模型比线性模型(0 隐藏单元)做得更差。即使对两个隐藏单元，10 个初始权值配置中的两个所产生的结果也不比线性模型好，这证实了多个初始值的重要性。

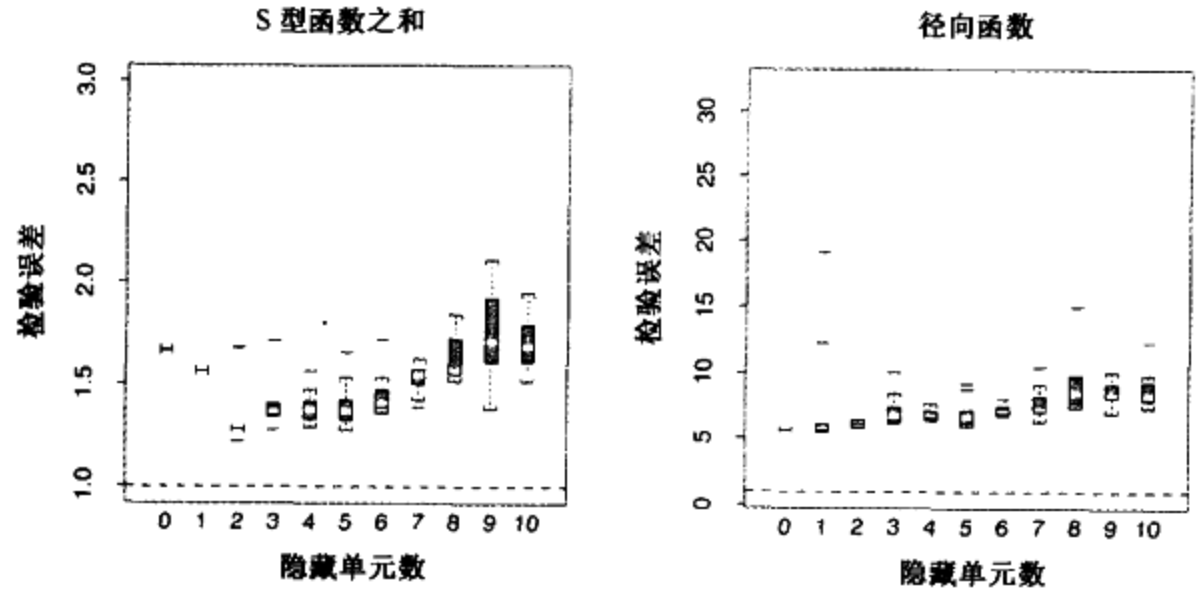


图 11.6 关于模拟数据例子，相对于贝叶斯误差(虚的水平线)的检验误差的盒图。左图，真实函数是两个S型函数之和；右图，径向函数。对于10种不同的初始权值和一个具有指定单元数目的单隐藏层神经网络，图中显示了检验误差

在某种意义上，径向函数最难用神经网络处理，因为它是球形对称而且没有首选的方向。在图 11.6 的右图中，我们看到它在该情况下性能很差，检验误差一直高于贝叶斯误差(注意垂直刻度不同于左图)。事实上，由于常量拟合(如样本平均)达到的相对误差是 5(当 SNR 是 4 时)，所以我们看到神经网络的表现比均值越来越差。

在这个例子中，我们使用了一个固定的权衰减参数 0.0005，表示一个适度的正则化量。图 11.6的左图结果表明具有更多的隐藏单元需要更多的正则化。

在图 11.7 中，我们重复了 S 型模型之和的试验，左图没有权衰减，右图有更强的权衰减($\lambda=0.1$)。没有权衰减时，如果有大量的隐藏单元，过分拟合会变得更加严重。权衰减值 $\lambda=0.1$时，对于所有的隐藏单元数都会产生好的结果，而且随着单元数目的增加没有出现过分拟合。最后，图 11.8 显示了 10 个隐藏单元网络的检验误差，权衰减参数在一个大范围上变化。值 0.1 是近似最优的。

概括地说，有两个自由参数可以选择：权衰减 λ 和隐藏单元数 M。作为一种学习策略；我们可以固定任何一个参数，对于极小约束模型，以确保模型足够丰富，并且使用交叉验证来选择另一个参数。这里，极小限制值是 0 权衰减和 10 个隐藏单元。比较图 11.7 的左图和图 11.8，我们看到检验误差对权衰减参数的值不是很敏感，因此，该参数的交叉验证将是可取的。

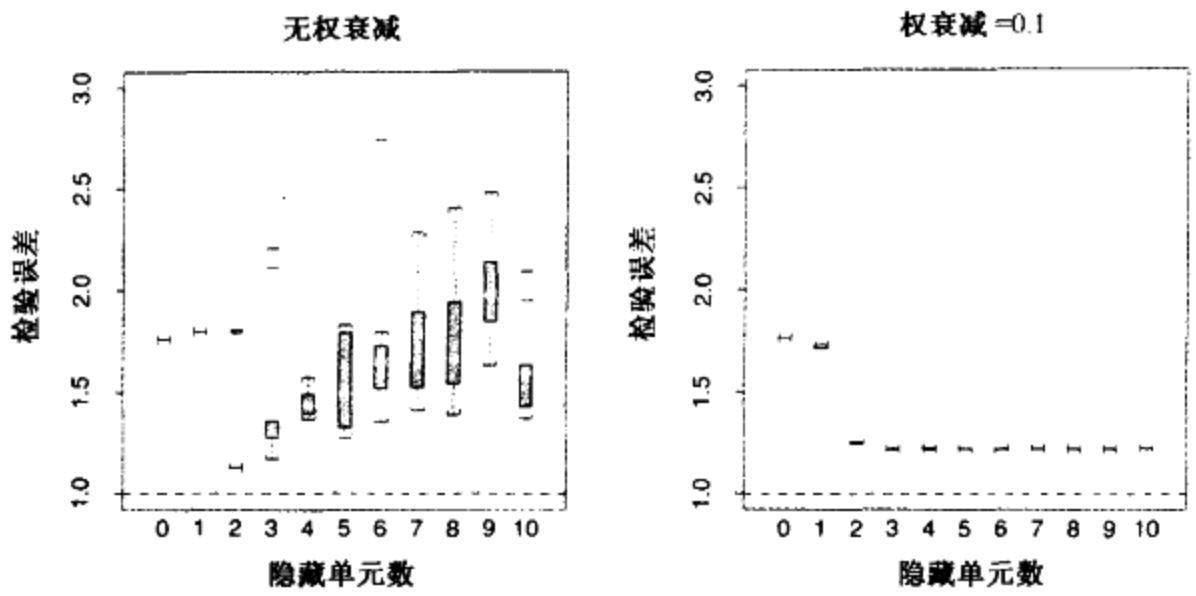


图 11.7　模拟数据例子相对于贝叶斯误差的检验误差盒图。真实函数是两个 S 型函数之和。对于10种不同的初始权值和一个具有指定单元数目的单隐藏层神经网络，图中显示了检验误差。两幅图分别表示了无权衰减(左)和具有强权衰减λ = 0.1(右)

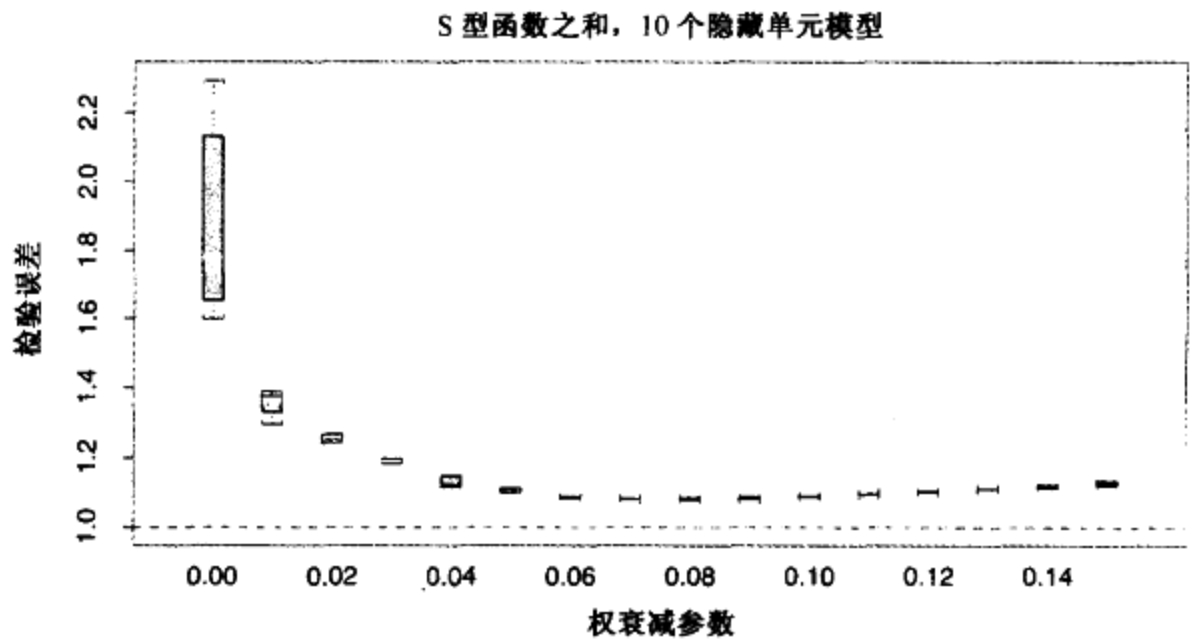


图 11.8　模拟数据例子的检验误差盒图。真实函数是两个 S 型函数之和。对于 10 种不同的初始权值以及一个具有10个隐藏单元的单隐藏层和指定权衰减参数值的神经网络，图中显示了检验误差

11.7　例：ZIP 编码数据

该例子是一个字符识别任务：手写数字的分类。多年来，该问题吸引了机器学习和神经网络界的关注，而且一直是该领域的基准问题。图 11.9 显示了一些规范化手写数字的例子，这些数字是由美国邮政局从信封上自动扫描下来的。最初扫描的数字是二进制的，有不同的大小和倾斜度；这里所显示的图像已被归一化，并对大小进行了标准化，结果用 16 × 16 灰度图像显示(Le Cun 等人，1990)。这 256 个像素值用做神经网络分类器的输入。

对于这种模式识别任务，黑箱(black box)神经网络并不理想，部分原因是图像的像素表示缺乏某种不变性(如图像较小的旋转)。因而，神经网络在该问题的不同实例上的早期尝试产生的误分类率大约是 4.5%。本节将展示一些手工构造神经网络，以克服某些不足的开拓性工作

(Le Cun,1989),这些努力最终导致神经网络性能的技艺状态(Le Cun 等人,1998)[①]。

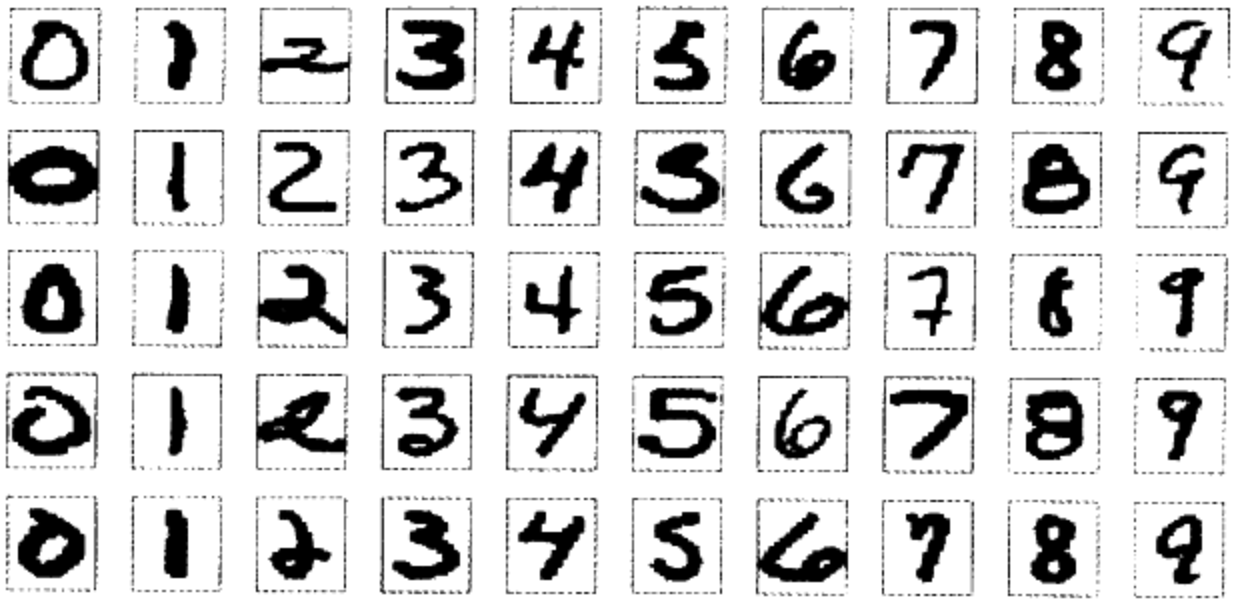


图 11.9 ZIP 编码数据的训练实例。每幅图像是手写数字的 16×16 的 8 位灰度表示

尽管目前的数字数据集中有数万个训练和检验的实例,但这里的样本大小是适度的,旨在强调效果。这些例子是通过扫描一些手写数字获得的,然后通过随机平移产生附加图像。细节可以在 Le Cun(1989)中找到。训练集中有 320 个数字,而检验集中有 160 个数字。

5 种不同的网络用于这些数据拟合:

网络 1:无隐藏层,等价于多项式逻辑斯缔回归。

网络 2:一个隐藏层,12 个隐藏单元,全连接。

网络 3:两个隐藏层,局部连接。

网络 4:两个隐藏层,局部连接,权值共享。

网络 5:两个隐藏层,局部连接,两个权值共享层。

上述 5 种网络在图 11.10 中描绘。例如,网络 1 有 256 个输入,对应于 16×16 个输入像素,10 个输出单元,对应于数字 0~9。预测值 $\hat{f}_k(x)$ 表示图像 x 属于数字类 $k(k=0,1,\cdots,9)$ 的估计概率。

网络都有 S 型输出单元,而且都用平方和误差函数来拟合。第一个网络没有隐藏层,因此几乎等价于线性多项式回归模型(见习题 11.4)。网络 2 是一个具有 12 个隐藏单元的单隐藏层神经网络,其种类如上所述。

所有网络的训练集误差都是 0%,因为在所有情况下,参数都多于训练观察。训练阶段的检验误差演变显示在图 11.11 中。线性网络(网络 1)过分拟合开始得相当快,而其他网络的检验性能在后继的较好值上稳定下来。

其他三种网络还有别的特征,这些特征显示了神经网络的能力和灵活性。它们在网络上引入约束,对于手头上的问题是自然的,这允许更加复杂的连通性和较少的参数。

网络 3 使用局部连通性:这意味着每个隐藏单元仅与下层的一小片单元相连接。在第一隐藏层(一个 8×8 的数组),每个单元从输入层的一个 3×3 的小片取得输入;对于第一个隐藏层中相隔一个单元的单元,它们的接收区域被一行或一列所覆盖,因此相隔两个像素。在第二个隐藏层,输入取自一个 5×5 的小片,相隔一个单元的单元的接收区域也相隔两个单元。其

① 本例的图表根据 Le Cun(1989)重新制作。

他连接的权值都置为 0。局部连通性使得每个单元负责从下一层提取局部特征，并大量减少权值的总数。与具有更多隐藏单元的网络 2 相比，网络 3 有较少的连接和权值(1226 对 3214)，并获得相似的性能。

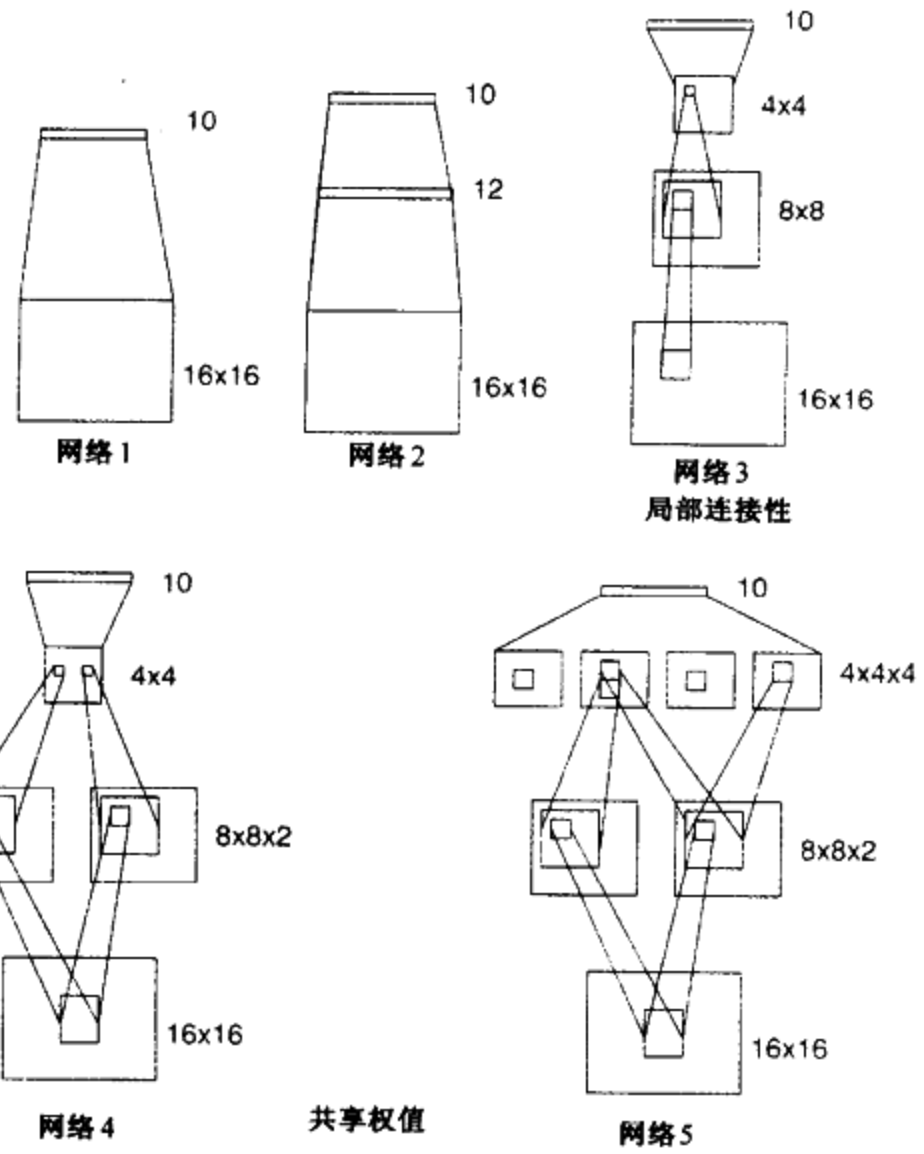


图 11.10　用于 ZIP 编码例子的 5 种神经网络的结构

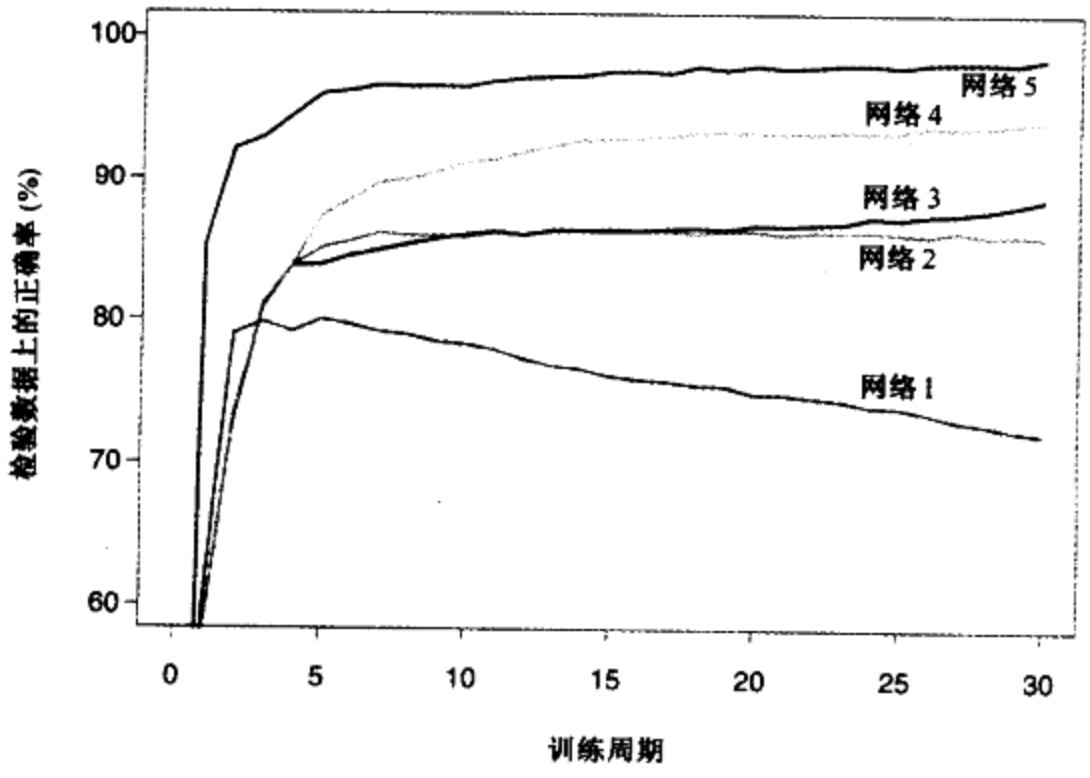


图 11.11　表 11.1 中的 5 种网络用于 ZIP 编码数据的检验性能曲线，是训练周期的函数(Le Cun, 1989)

网络 4 和网络 5 具有局部连通性,共享权值。在一个局部特征映射中的所有单元在图像的不同部分执行相同操作,这些操作通过共享相同的权值来完成。网络 4 的第一个隐藏层有两个 8×8 的数组。正如网络 3,每个单元从一个 3×3 小片中取得输入。然而,在每个 8×8 特征映射中,每个单元共享相同的 9 个权值的集合(但有各自的偏置参数)。这就迫使在图像的不同部分抽取的特征可以用相同的线性泛函来计算,因此,这些网络有时称为卷积网络(convolutional network)。网络 4 的第二个隐藏层没有权共享,与网络 3 一样。误差函数 R 关于共享权值的梯度是 R 关于被考虑的权控制的每个连接的梯度之和。

表 11.1 给出了每个网络的链的数目、权的数目和最优检验性能。我们看到网络 4 比网络 3 有更多的链但有更少的权,也具有更好的检验性能。网络 5 在第二个隐藏层有 4 个 4×4 特征映射,每个单元与下一层的一个 5×5 局部区域相连接。权在这些特征映射的每一个中被共享。我们看到网络 5 做得最好,误差仅为 1.6%。相比之下,"香草"网络 2 误差为 13%。手写风格可以出现在一个数字的多个部分,受这种特点启发,网络 5 的巧妙设计是许多人多年经验的结果。这种类似的网络对邮政编码问题给出了比当时(上世纪 90 年代早期)的任何其他学习方法都好的性能。正像宣传的那样,该例子也表明神经网络不是一种完全自动的工具。像使用所有统计模型一样,主题知识可以并且应该用于提高它们的性能。

表 11.1 5 种不同神经网络在手写数字分类例上的检验集性能(Le Cun,1989)

网络结构	链	权值	正确率(%)
网络 1:单层网络	2570	2570	80.0%
网络 2:两层网络	3214	3214	87.0%
网络 3:局部连接	1226	1226	88.5%
网络 4:被约束的网络 1	2266	1132	94.0%
网络 5:被约束的网络 2	5194	1060	98.4%

这个网络后来被第 13.3.3 节讨论的正切距离方法(Simard 等人,1993)胜出。正切距离方法显式地结合了自然仿射的不变性。从那时起,数字识别数据集变成各种新学习过程的实验台,而研究者们努力降低误差。截止到本书编写时,在一个大型数据库(60 000 个训练,10 000 个检验观测,由标准 NIST[①] 数据库导出)上,最好误差率有如下报道(Le Cun 等人,1998):

- 1.1%:正切距离,使用 1 - 最近邻分类法(见第 13.3.3 节);
- 0.8%:9 次多项式 SVM(见第 12.3 节);
- 0.8%:LeNet-5,这里介绍的卷积网络更复杂的版本;
- 0.7%:提升的 LeNet-4。提升方法在第 8 章介绍过,LeNet-4 是 LeNet-5 的前驱。

Le Cun 等人(1998)给出了一张相当大的性能结果表。显然,许多研究小组一直在努力工作,以降低检验误差率。他们报道了误差估计的 0.1% 的标准误差,是基于二项式平均的,其 $N=10\,000$ 而 $p\approx0.01$。这意味相互之间 0.1% 到 0.2% 的误差率在统计上是等价的。实际上,标准误差甚至更高,因为检验数据隐式地用于各种过程的调整。

① 国家标准和技术委员会维护了一些大型数据库,包括手写体字符数据库;http://www.nist.gov/srd/。

11.8　讨论

投影寻踪回归和神经网络都采用了输入线性组合("导出特性")的非线性函数。对于回归和分类,这是一种非常有效而一般的方法。业已表明对于很多问题,它能与最好的学习方法媲美。

这些工具对于高信噪比的问题和以预测为目标而不需要解释的情况特别有效。而对于那些旨在描述产生数据的物理过程和每个输入作用的问题,它们就不大有效。每个输入以非线性方式在不同地方进入模型。某些作者(Hinton,1989)绘制了进入每个隐藏单元的估计权值的图,试图理解每个单元正在抽取的特征。然而,这受到参数向量α_m($m = 1,\cdots,M$)缺乏同一性的限制。通常,存在一些解,α_m 生成的线性空间与训练阶段所发现的相同,给出大致相同的预测值。有些作者建议对这些权进行主成分分析,以便发现一个可解释的解。通常,解释这些模型的困难限制了它们在一些领域(如医学领域)的使用;那里,模型的可解释性是非常重要的。

有大量神经网络模型在训练方面的研究。与 CART 及 MARS 不同,神经网络是实值参数的光滑函数,这就有利于这些模型在贝叶斯推理方面的发展。一些参考文献在下面的文献注释中给出。

11.9　计算考虑

对于 N 个观察、p 个预测、M 个隐藏单元和 L 个训练周期,神经网络拟合一般需要 $O(NpML)$次操作。许多软件包可以用于拟合神经网络,可能远远多于已存在的主流统计学方法。由于这些可用的软件在质量上有很大不同,神经网络的学习对输入定标这样一类问题是很敏感的,这样的软件应谨慎地选择和测试。

文献注释

投影寻踪是由 Friedman 和 Tukey(1974)提出的,Friedman 和 Stuetzle(1981)将其专门用于回归。Huber(1985)给出了学术性的综述,Roosen 和 Hastie(1994)提供了使用光滑样条的系统而确切的陈述。神经网络的动机可追溯到 McCulloch 和 Pitts(1943)、Widrow 和 Hoff(1960)[重印于 Anderson 和 Rosenfeld(1988)],以及 Rosenblatt(1962)。Hebb(1949)对学习算法的发展影响很大。20 世纪 80 年代中期神经网络的复苏应归功于 Werbos(1974)、Parker(1985)和 Rumelhart 等人(1986),他们提出了反向传播算法。现在,有很多关于该主题的书,适用于很大范围的读者。对于本书的读者来说,Hertz 等人(1991)、Bishop(1995)和 Ripley(1996) 的资料可能是最详实的。关于神经网络的贝叶斯学习在 Neal(1996)中讲述。ZIP 编码例子取自 Le Cun(1989),还可以参见 Le Cun 等人(1990)和 Le Cun 等人(1998)的著作。

我们没有讨论诸如神经网络的逼近特性等理论问题,例如,Barron(1993)、Girosi 等人(1995)和 Jones(1992)的工作。Ripley(1996)对这些结果进行了概括。

习题

11.1 建立投影寻踪回归模型(11.1)和神经网络(11.5)之间准确的对应关系。特别地,证明:单层回归网络等价于具有 $g_m(\omega_m^T x) = \beta_m \sigma(\alpha_{0m} + s_m(\omega_m^T x))$的 PPR 模型,其中 ω_m 是第 m 个单元向量。为分类网络建立一个类似的等价性。

11.2 考虑一个有定量输出的神经网络,像式(11.5)那样,使用平方误差损失和恒等输出函数 $g_k(t) = t$。假设从输入到隐藏层的权α_m 接近于 0。证明结果模型在输入上接近于线性的。

11.3 导出互熵损失函数的正向和反向传播方程式。

11.4 为一个 K-类输出,考虑一个使用互熵损失的神经网络。如果网络没有隐藏层,证明模型等价于第 4 章讨论的多项式逻辑斯缔模型。

11.5 (a) 编写一个程序,通过反向传输和权衰变拟合一个单隐藏层神经网络(10 个隐藏单元)。

(b) 对来自下面模型的 100 个观察应用它:

$$Y = \sigma(a_1^T X) + (a_2^T X)^2 + 0.30 \cdot Z$$

其中 σ 是 S 型函数,Z 是标准正态的,$X = (X_1, X_2)$,每个 X_j 是独立、标准正态的,$a_1 = (3,3), a_2 = (3,-3)$。产生一个大小为 1000 的检验样本,并对于权衰减参数的不同值,作为训练周期的函数,绘制训练和检验误差曲线。

(c) 从 1 到 10 改变网络中隐藏单元的数量,并确定能很好地完成此任务的最小数目。

11.6 编写一个程序,使用具有固定自由度的三次光滑样条实现投影寻踪回归。对不同的光滑参数值和模型项数,用它拟合前面习题中的数据。找出模型能很好完成任务所需模型项的最小数目,并将其与前一习题中的隐藏单元数目进行比较。

11.7 拟合一个神经网络到第 9.1.2 节的 spam 数据,并将该结果与第 9 章的加法模型结果做比较。比较最终模型的分类性能和可解释性。

第 12 章　支持向量机和柔性判别

12.1　引言

本章将讲述关于分类的线性判定边界的推广。对于两个类线性可分的情况，我们在第 4 章介绍了最佳分离超平面。这里考虑把它扩展到不可分的(类重叠)情况。然后，将这些技术推广到支持向量机(support vector machine)。通过在一个大的、变换后的特征空间中构造线性边界，支持向量机产生非线性边界。第二种方法集是对费希尔线性判别分析(LDA) 的推广，包括柔性判别分析(flexible discriminant analysis)，它有助于非线性边界以类似于支持向量机的方式构造；用于诸如信号和图像分类的罚判别分析(penalized discriminant analysis)，那里大量的特征是高度相关的；以及用于不规则形状类的混合判别分析(mixture discriminant analysis)。

12.2　支持向量分类器

在第 4 章，我们讨论了在两个完全可分的类之间构造最佳分离超平面的技术。这里略做回顾，并将它推广到不可分的情况，其中类也许不能被线性边界分开。

我们的训练数据包括 N 个对 $(x_1\ y_1)$，(x_2, y_2)，$\cdots$，(x_N, y_N)，其中 $x_i \in \mathbb{R}^p$，而 $y_i \in \{-1, 1\}$。超平面由下式定义：

$$\{x:\ f(x) = x^T\beta + \beta_0 = 0\} \tag{12.1}$$

其中，β 是一个单位向量：$\|\beta\| = 1$。由 $f(x)$ 导出的分类规则是：

$$G(x) = \text{sign}[x^T\beta + \beta_0] \tag{12.2}$$

超平面的几何形状参见第 4.5 节。在那里，我们看到式(12.1)中 $f(x)$ 给出了从点 x 到超平面 $f(x) = x^T\beta + \beta_0 = 0$ 的有符号距离。由于类是可分的，我们可以找到函数 $f(x) = x^T\beta + \beta_0$，它满足对于任意 i，$y_i f(x_i) > 0$。因此，我们能够找到超平面，在类 1 和类 -1 的训练点之间产生最大的边缘 (见图 12.1)。这对应于如下最优化问题：

$$\begin{aligned} &\max_{\beta, \beta_0, \|\beta\|=1} C \\ &\text{受限于 } y_i(x_i^T\beta + \beta_0) \geqslant C,\ i = 1, \cdots, N \end{aligned} \tag{12.3}$$

图中的带在超平面的两侧距超平面 C 个单位，因此宽度为 $2C$。它被称为边缘(margin)。

我们已经表明该问题可以更方便地表示为：

$$\begin{aligned} &\min_{\beta, \beta_0} \|\beta\| \\ &\text{受限于 } y_i(x_i^T\beta + \beta_0) \geqslant 1,\ i = 1, \ldots, N \end{aligned} \tag{12.4}$$

其中，我们略去了对 β 的范数的限制。注意，$C = 1/\|\beta\|$。式(12.4)是描述分离数据的支持向量机准则的常用方式。这是一个凸优化问题(二次准则，线性不等式约束)，解曾在第 4.5.2 节描述过。

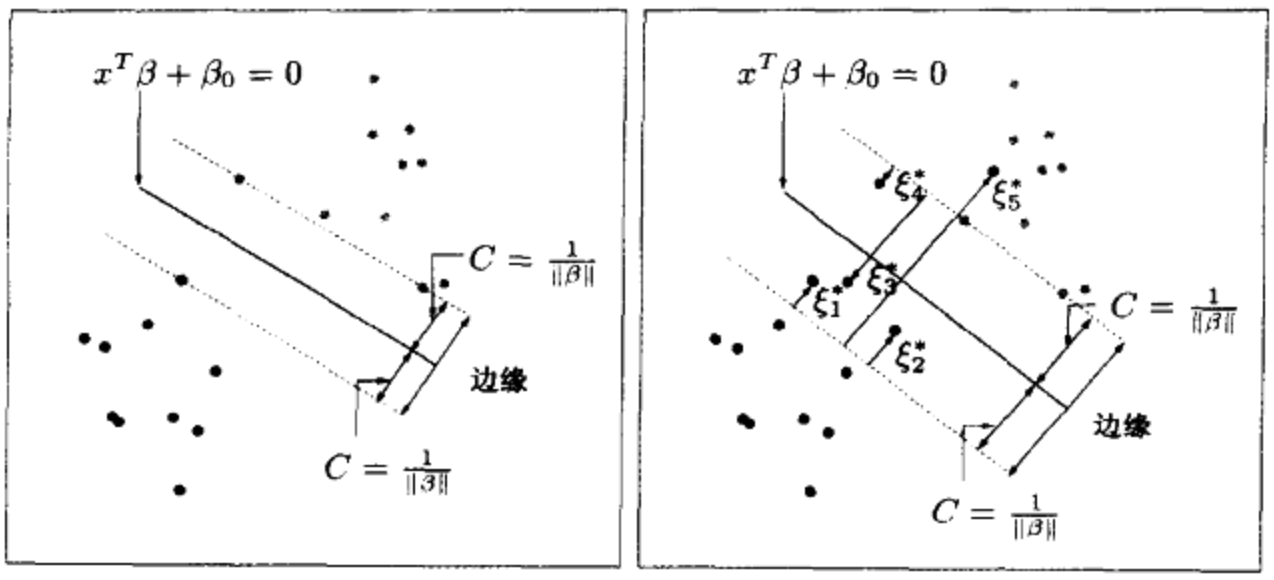


图 12.1 支持向量分类器。左图显示可分情况。判定边界是实线，而虚线界定宽度为 $2C=2/\|\beta\|$ 的阴影的最大边缘。右图显示不可分的(重叠)情况，标有 ξ_j^* 的点位于其边缘的错误侧，相差量 $\xi_j^*=C\xi_j$；在正确侧的点都有 $\xi_j^*=0$。边缘被极大化，服从 $\sum\xi_i\leqslant$常量。因此，$\sum\xi_j^*$ 是在其边缘错误侧的点的总距离(见彩页)

现在，假设类在特征空间中有重叠。处理重叠的一种方法仍然是极大化 C，但允许某些点出现在边缘的错误侧。定义松弛变量 $\xi=(\xi_1,\xi_2,\cdots,\xi_N)$。有两种自然方式修改式(12.3)中的约束：

$$y_i(x_i^T\beta+\beta_0).\quad\geqslant\quad C-\xi_i \tag{12.5}$$

或

$$y_i(x_i^T\beta+\beta_0).\quad\geqslant\quad C(1-\xi_i) \tag{12.6}$$

$\forall i,\xi_i\geqslant 0$，$\sum_{i=1}^N\xi_i\leqslant$常量。两种选择导致不同的解。虽然两种方法都很自然，但第二种选择能导致“标准的”支持向量分类器，因此我们使用它。

这里给出形式化思想。约束 $y_i(x_i^T\beta+\beta_0)\geqslant C(1-\xi_i)$ 中的值 ξ_i 是使预测 $f(x_i)=x_i^T\beta+\beta_0$ 出现在其边缘错误侧的比例。因此，通过约束 $\sum\xi_i$，就可以限制使预测落在它们的边缘错误侧的总比例。误分类发生在 $\xi_i>1$ 时，所以约束 $\sum\xi_i$ 在值 K 就限制训练误分类的总数于 K。

和在第 4.5.2 节中的式(4.44) 一样，我们可以忽略对 β 的范数的限制。定义 $C=1/\|\beta\|$，则式(12.4)可以写成等价的形式：

$$\min\|\beta\|\ \text{受限于}\begin{cases} y_i(x_i^T\beta+\beta_0)\geqslant 1-\xi_i\ \forall i \\ \xi_i\geqslant 0,\ \sum\xi_i\leqslant\text{常量}\end{cases} \tag{12.7}$$

对于不可分的情况，这是定义支持向量分类器的常用方式。然而，我们发现在约束 $y_i(x_i^T\beta+\beta_0)\geqslant 1-\xi_i$ 中固定标量“1”的出现容易误解，而宁愿由式(12.6)开始。图 12.1 的右图刻画了这种重叠情况。

根据准则(12.7)的特征，我们看到在其类边界以内的点对边界形成所起的作用不大。这似乎是一个吸引人的特性，也正是它区别于线性判别分析(见第 4.3 节)的一个特性。在 LDA 中，判定界限由类分布的协方差和类质心的位置来确定。我们将在第 12.3.3 节看到，在这个意义上，逻辑斯缔回归更接近于支持向量分类器。

12.2.1　计算支持向量分类器

问题(12.7)是二次的,具有线性不等式约束,因此是一个凸优化问题。我们使用拉格朗日乘子描述一个二次规划解。在计算上,用如下等价式形式地表示式(12.7)是方便的:

$$\min_{\beta,\beta_0} \frac{1}{2}\|\beta\|^2 + \gamma\sum_{i=1}^{N}\xi_i \tag{12.8}$$
$$\text{受限于 } \xi_i \geqslant 0,\ y_i(x_i^T\beta+\beta_0) \geqslant 1-\xi_i\ \forall i$$

其中 γ 代替式(12.7)中的常量;可分情况相当于 $\gamma=\infty$。

拉格朗日(原始的)函数是:

$$L_P = \frac{1}{2}\|\beta\|^2 + \gamma\sum_{i=1}^{N}\xi_i - \sum_{i=1}^{N}\alpha_i[y_i(x_i^T\beta+\beta_0)-(1-\xi_i)] - \sum_{i=1}^{N}\mu_i\xi_i \tag{12.9}$$

关于 β, β_0 和 ξ_i,我们对其极小化。令各自的导数为 0,可以得到:

$$\beta = \sum_{i=1}^{N}\alpha_i y_i x_i \tag{12.10}$$

$$0 = \sum_{i=1}^{N}\alpha_i y_i \tag{12.11}$$

$$\alpha_i = \gamma - \mu_i,\ \forall i \tag{12.12}$$

和正约束 $\alpha_i, \mu_i, \xi_i \geqslant 0,\ \forall i$。把式(12.10)到式(12.12)替换到式(12.9)中,得到拉格朗日(Wolfe)对偶目标函数:

$$L_D = \sum_{i=1}^{N}\alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{i'=1}^{N}\alpha_i\alpha_{i'}y_i y_{i'}x_i^T x_{i'} \tag{12.13}$$

对任意可行的点,它给出了目标函数(12.8)的一个下界。在 $0\leqslant\alpha_i\leqslant\gamma$ 和 $\sum_{i=1}^{N}\alpha_i y_i = 0$ 约束下,我们对 L_D 极大化。除式(12.10)到式(12.12)之外,Karush-Kuhn-Tucker 条件包括约束:

$$\alpha_i[y_i(x_i^T\beta+\beta_0)-(1-\xi_i)] = 0 \tag{12.14}$$

$$\mu_i\xi_i = 0 \tag{12.15}$$

$$y_i(x_i^T\beta+\beta_0)-(1-\xi_i) \geqslant 0 \tag{12.16}$$

其中,$i=1,\cdots,N$。式(12.10)~式(12.16)一起惟一刻画了原问题和对偶问题的解。

由式(12.10),我们看到 β 的解具有如下形式:

$$\hat{\beta} = \sum_{i=1}^{N}\hat{\alpha}_i y_i x_i \tag{12.17}$$

其中,仅对满足式(12.16)中约束[由于有式(12.14)]的观测 i 有非零系数 $\hat{\alpha}_i$。这些观测称为支持向量(support vector),因为 $\hat{\beta}$ 仅用它们表示。在这些支持点中,有些将位于边缘的边上($\hat{\xi}_i=0$),因此由式(12.15)和式(12.12),它们将由 $0<\hat{\alpha}_i<\gamma$ 来刻画;其余的($\hat{\xi}_i>0$)有 $\hat{\alpha}_i=\gamma$。从

式(12.14)可以看到任何一个边缘点($0<\hat{\alpha}_i, \hat{\xi}_i=0$)都可以用于求解 β_0,而为了数值的稳定性,我们可以利用全部解的平均值。

极大化对偶(12.13)是一个比原问题(12.9)更简单的凸二次规划问题。可以用标准技术求解(例如,Murray 等人的著作,1981)。

给定解 $\hat{\beta}_0$ 和 $\hat{\beta}$,则判定函数可以写做:

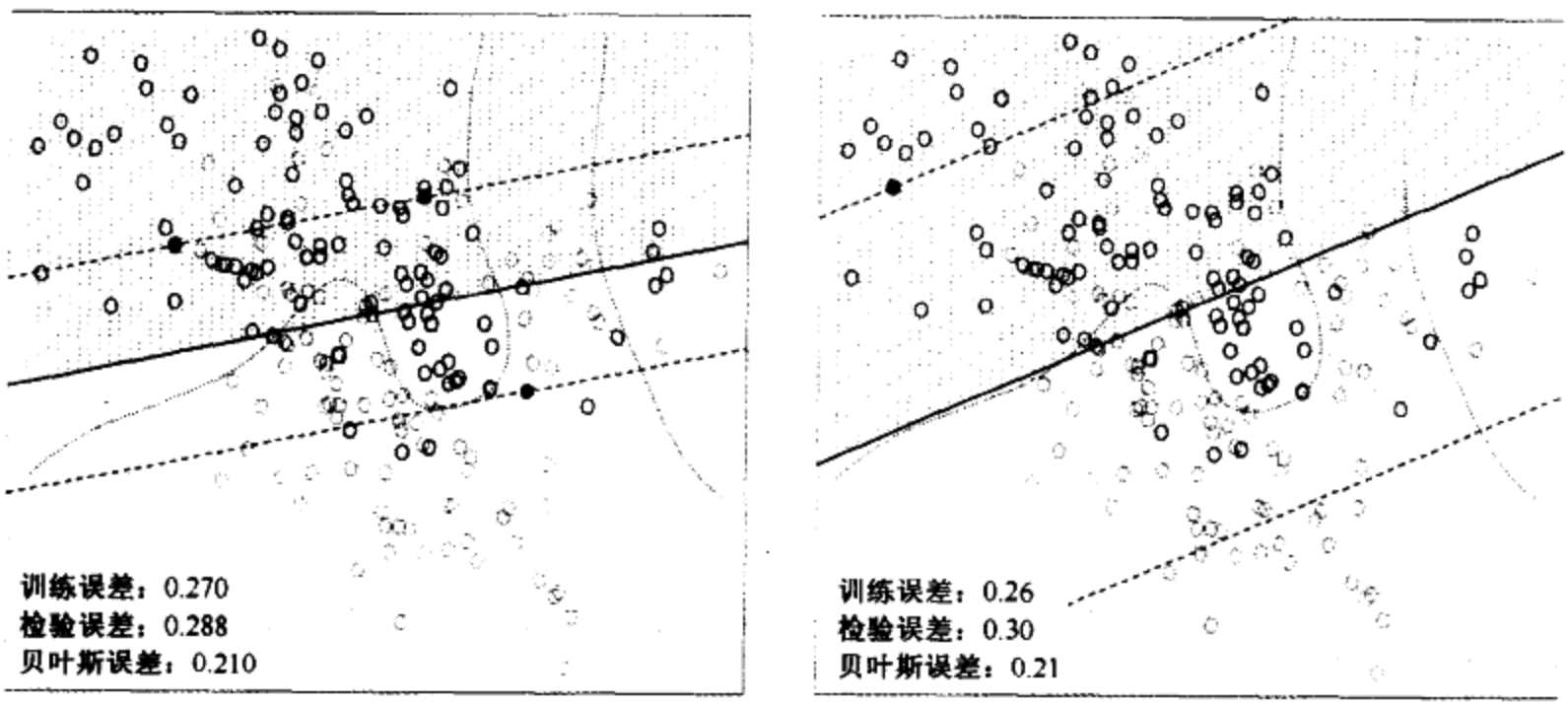
$$\begin{aligned}\hat{G}(x) &= \text{sign}[\hat{f}(x)] \\ &= \text{sign}[x^T\hat{\beta}+\hat{\beta}_0]\end{aligned} \tag{12.18}$$

这个过程的调整参数是 γ。

12.2.2 混合例子(续)

图 12.2 显示了图 2.5 的混合示例(有两个重叠类)的支持向量边界,调整参数 γ 取了两个不同的值。这些分类器在性能上非常相似。在边界错误侧上的点是支持向量。另外,在边界正确侧但很接边界(在边缘内)的点也是支持向量。$\gamma=0.01$ 的边缘比 $\gamma=10\,000$ 的边缘大。因此,γ 值越大,越关注靠近判定边界(被正确分类)的点,而较小的值则涉及较远的数据。两种方式,不管离开多远,误分类的点都给定了权值。在这个例子中,由于线性边界的严格性,过程对于 γ 的选择并不是特别灵敏。

γ 的最优值可以用交叉验证来估计,如第 7 章的讨论。有趣的是,留一交叉验证误差可以用数据中支持点的比例来约束。理由是忽略一个不是支持向量的观测将不会改变解。因此,这些被原始边界正确分类的观测在交叉验证过程中将被正确分类。然而,这个约束太强,对选择 γ 通常是没有用的(在我们的例子中,分别是 62% 和 85%)。



$\gamma=10\,000$ $\gamma=0.01$

图 12.2 对于两个不同的 γ 值,有两个重叠类的混合数据示例的线性支持向量边界。虚线指明了边缘,其中 $f(x)=\pm 1$。支持点($\alpha_i>0$)是在边缘错误侧上的全部点。黑实点是恰好落在边缘($\xi_i=0, \alpha_i>0$)上的支持点。在左图中 62% 的观测是支持点,在右图中 85% 的观测是支持点,背景上的紫色虚线是贝叶斯判定边界(见彩页)

12.3　支持向量机

迄今为止讨论的支持向量分类器发现了输入特征空间中的线性边界。如同对其他线性方法一样，我们可以通过使用基展开，如多项式或样条来扩大特征空间，使过程更加灵活（见第 5 章）。通常，在扩大的特征空间中的线性边界能较好地实现训练类的分离，并变换成原始空间中的非线性边界。一旦选择了基函数 $h_m(x)$，$m=1,\cdots,M$，过程就和以前的相同。我们使用输入特征 $h(x_i)=(h_1(x_i), h_2(x_i), \cdots, h_M(x_i))$，$i=1, 2, \cdots, N$，拟合 SV 分类器，并产生（非线性）函数 $\hat{f}(x)=h(x)^T\hat{\beta}+\hat{\beta}_0$。和以前一样，分类器是 $\hat{G}(x)=\text{sign}(\hat{f}(x))$。

支持向量机分类器是这种思想的扩展，允许扩大的空间维数非常大，在某些情况下可能无穷大。看上去计算量可能变得让人望而生畏。也许使用充足的基函数，数据将是可分的，但可能出现过分拟合。我们首先展示 SVM 技术如何处理这些问题。然后，我们会看到，事实上，SVM 分类器使用特殊准则和正则化形式解决函数拟合问题，是包括第 5 章的光滑样条在内的更大的一类问题的一部分。读者可以查阅第 5.8 节，那里提供了背景资料并与下面两节多少有些重叠。

12.3.1　计算分类的 SVM

可以用一种特殊方式表示最优化问题（12.9）和它的解，这种方式只通过内积涉及输入特征。我们直接对变换后的特征向量 $h(x_i)$来做。然后，我们会看到对于 h 的特定选择，这些内积可以毫不费力气地计算出来。

拉格朗日对偶函数（12.13）有如下形式：

$$L_D=\sum_{i=1}^{N}\alpha_i-\frac{1}{2}\sum_{i=1}^{N}\sum_{i'=1}^{N}\alpha_i\alpha_{i'}y_iy_{i'}\langle h(x_i),h(x_{i'})\rangle \tag{12.19}$$

由式（12.10），我们看到解函数 $f(x)$可以写成：

$$\begin{aligned} f(x) &= h(x)^T\beta+\beta_0 \\ &= \sum_{i=1}^{N}\alpha_iy_i\langle h(x),h(x_i)\rangle+\beta_0 \end{aligned} \tag{12.20}$$

与前面一样，给定的α_i 和β_0 可以通过对任意（或全部）满足 $0<\alpha_i<\gamma$ 的 x_i，在式（12.20）中求解$y_if(x_i)=1$ 来确定。

式（12.19）和式（12.20）都仅通过内积涉及 $h(x)$。事实上，我们根本不需要指定变换 $h(x)$，而只是需要核函数的知识：

$$K(x,x')=\langle h(x),h(x')\rangle \tag{12.21}$$

它在变换后的空间中计算内积。K 应该是一个对称正定（半正定）函数；参见第 5.8.1 节。

在 SVM 文献中，对 K 的三种流行选择是：

$$\begin{aligned} d\text{次多项式}&: K(x,x')=(1+\langle x,x'\rangle)^d \\ \text{径向基}&: K(x,x')=\exp(-\|x-x'\|^2/c) \\ \text{神经网络}&: K(x,x')=\tanh(\kappa_1\langle x,x'\rangle+\kappa_2) \end{aligned} \tag{12.22}$$

例如，考虑有两个输入 X_1 和 X_2 的特征空间和一个 2 次多项式核。则

$$\begin{aligned} K(X, X') &= (1 + \langle X, X' \rangle)^2 \\ &= (1 + X_1 X_1' + X_2 X_2')^2 \\ &= 1 + 2X_1 X_1' + 2X_2 X_2' + (X_1 X_1')^2 + (X_2 X_2')^2 + 2X_1 X_1' X_2 X_2' \end{aligned} \tag{12.23}$$

那么 $M = 6$，并且如果选择 $h_1(X) = 1$，$h_2(X) = \sqrt{2}X_1$，$h_3(X) = \sqrt{2}X_2$，$h_4(X) = X_1^2$，$h_5(X) = X_2^2$，而 $h_6(X) = \sqrt{2}X_1 X_2$，则 $K(X, X') = \langle h(X), h(X') \rangle$。由式(12.20)，解可以写成：

$$\hat{f}(x) = \sum_{i=1}^{N} \hat{\alpha}_i y_i K(x, x_i) + \hat{\beta}_0 \tag{12.24}$$

在扩大的特征空间中，参数 γ 的作用很明显，因为在那里完全分离通常是可以实现的。一个大的 γ 值将阻止任何正的 ξ_i；并导致在原特征空间中过分拟合的摆动边界；小的 γ 值鼓励小的 $\|\beta\|$ 值，它依次导致 $f(x)$ 以及边界较为光滑。图 12.3 显示了两个应用于第 2 章混合例子的非线性支持向量机。两种情况都选择了正则化参数，得到了较好的检验误差。对于该例，径向基核产生的边界与贝叶斯最优边界非常相似；比较图 2.5。

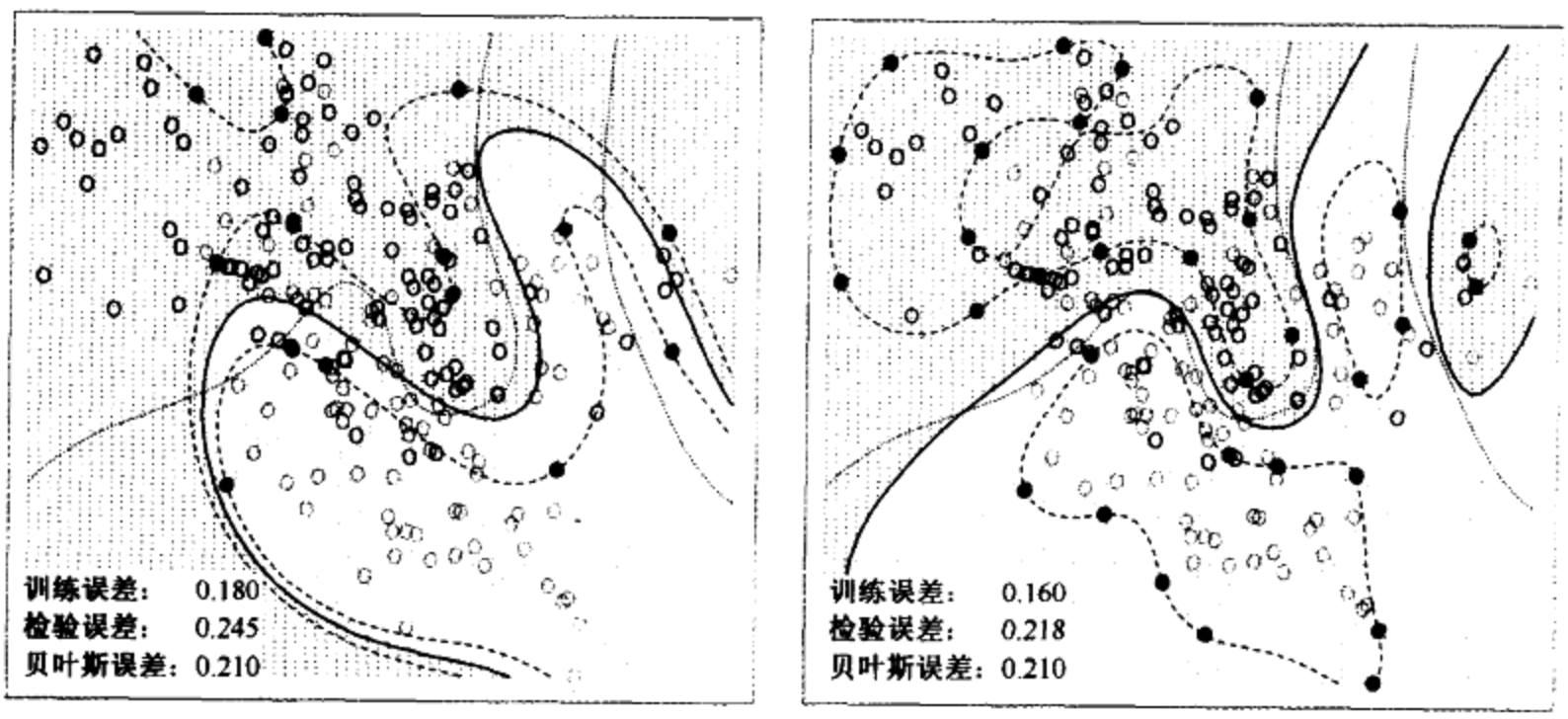


图 12.3 混合数据的两个非线性 SVM。左图使用一个 4 次多项式核，右图使用径向基核。在每种情况下，调整 γ 以近似地实现最好检验误差性能，且 $\gamma = 1$ 时，两种情况做得都很好。径向核实现得最好(接近贝叶斯最优解)；给定由高斯混合产生的数据，与期望的结果一样。背景上的紫色虚线是贝叶斯判定边界(见彩页)

在支持向量的早期文献中，有人称支持向量机的核性质是它的惟一特性，并允许我们巧妙地解决维灾难问题。这些说法都不正确，我们将在下面三节中讨论这两种观点。

12.3.2 作为罚方法的 SVM

对 $f(x) = h(x)^T\beta + \beta_0$，考虑最优化问题：

$$\min_{\beta_0,\beta}\sum_{i=1}^{N}[1-y_i f(x_i)]_+ + \lambda\|\beta\|^2 \tag{12.25}$$

其中，下标“+”指出正的部分。它的形式是“损失 + 罚”，这是函数估计中的常见形式。容易证明(见习题 12.1)：取 $\lambda = 1/(2\gamma)$，式(12.25)的解与式(12.8)相同。

对损失函数 $L(y,f) = [1-yf]_+$ 的研究表明，与其他更传统的损失函数相比，它对 2-类分类是合理的。图 12.4 将它与逻辑斯缔回归的对数似然损失以及平方误差损失做了比较。(负的)对数似然和 SVM 损失有相似的尾，对在其边缘内的点给予 0 罚，对错误侧和远离的点给予线性罚。另一方面，平方误差给出二次罚，并且在自己边缘内的点也对模型有很强的影响。

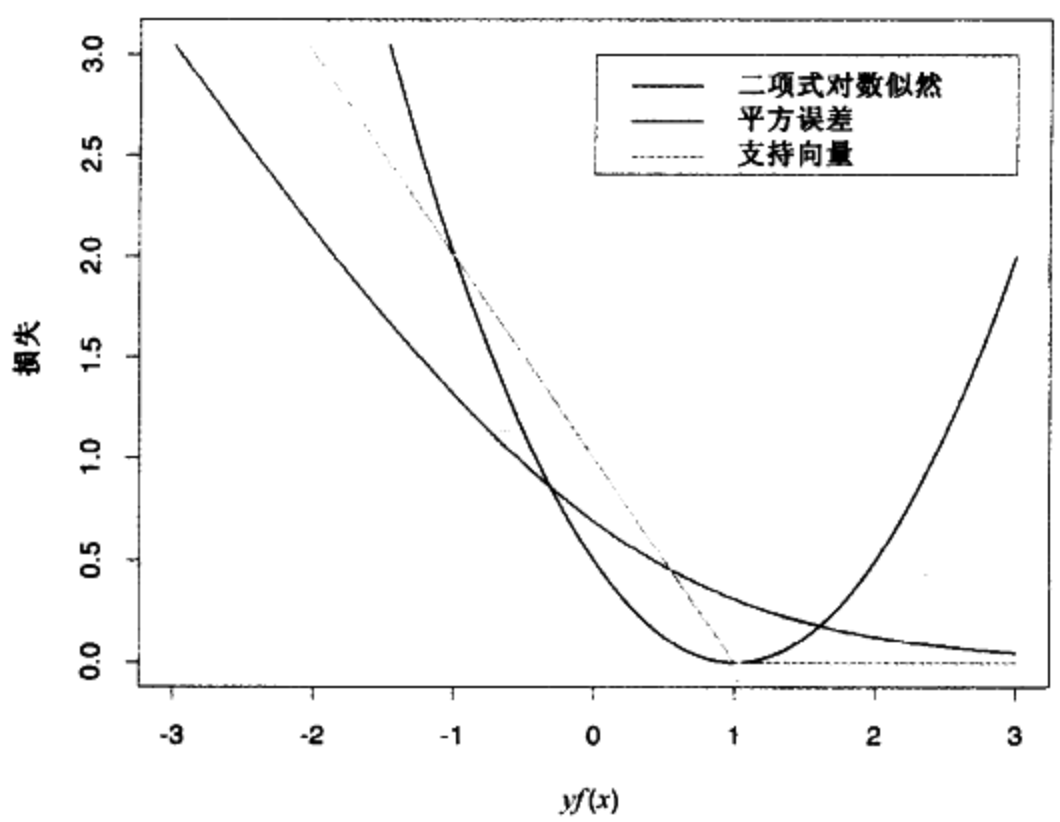


图 12.4　支持向量损失函数，与逻辑斯缔回归的(负的)对数似然损失和平方误差损失比较。所显示的都是 yf 而不是 f 的函数，因为在 $y = +1$ 和 $y = -1$ 之间三条曲线是对称的。对数似然与SVM有相同的渐近线，但是在内部是圆形的(见彩页)

我们可以根据它们在总体级的估计来刻画这三种损失函数的特点。考虑极小化 $EL(Y, f(X))$。表 12.1 汇总了这些结果。

这将 SVM 作为正则化的函数估计问题，其中线性展开式 $f(x) = \beta_0 + h(x)^T\beta$ 的系数向 0 收缩(除常量外)。如果 $h(x)$ 表示有一些有序结构(如按粗糙度有序)的分层基，那么当粗糙元素 h_j 在向量 h 中有较小的范数时，均匀收缩就更有意义。

表 12.1　三种不同损失函数的总体极小值。逻辑斯缔回归使用二项式对数似然。线性判别分析[见习题(4.51)]使用平方误差损失。SVM 损失估计后验类概率的众数

损失函数	$L(Y, f(X))$	极小化函数
(-)二项式对数似然	$\log(1+e^{-Yf(X)})$	$f(X) = \log\frac{\Pr(Y=+1\mid X)}{\Pr(Y=-1\mid X)}$
平方误差	$(Y-f(X))^2$	$f(X) = \Pr(Y=+1\mid X) - \Pr(Y=-1\mid X)$
支持向量机	$[1-Yf(X)]_+$	$f(X) = \begin{cases} +1, \text{如果 } \Pr(Y=+1\mid X) \geqslant \frac{1}{2} \\ -1, \text{其他} \end{cases}$

12.3.3 函数估计与再生核

这里,我们以再生核希尔伯特(Hilbert)空间中的函数估计来解释 SVM,其中核性质很丰富。这些内容在第 5.8 节有过详细的讨论。它提供了支持向量分类器的另一种视角,并且有助于阐明它是怎样工作的。

假设基 h 由一个正定核 K 的(可能是有穷的)本征展开式产生:

$$K(x,x')=\sum_{m=1}^{\infty}\phi_m(x)\phi_m(x')\delta_m \tag{12.26}$$

且 $h_m(x)=\sqrt{\delta_m}\phi_m(x)$。那么令 $\theta_m=\sqrt{\delta_m}\beta_m$,我们可以把式(12.25)写成:

$$\min_{\beta_0,\theta}\sum_{i=1}^{N}\Big[1-y_i(\beta_0+\sum_{m=1}^{\infty}\theta_m\phi_m(x_i))\Big]_+ +\lambda\sum_{m=1}^{\infty}\frac{\theta_m^2}{\delta_m} \tag{12.27}$$

现在,式(12.27)在形式上与第 5.8 节中的式(5.49)完全一样,并且在那里介绍的再生核希尔伯特空间理论确保存在如下形式的有限维解:

$$f(x)=\beta_0+\sum_{i=1}^{N}\alpha_iK(x,x_i) \tag{12.28}$$

特别地,在那里我们看到了最优化准则(12.19)的等价形式[见第 5.8.2 节的式(5.66),参见 Wahba等人(2000)的著作],

$$\min_{\alpha_0,\alpha}\sum_{i=1}^{N}(1-y_if(x_i))_+ +\lambda\alpha^T\mathbf{K}\alpha \tag{12.29}$$

其中,$\mathbf{K}$ 是对所有训练特征对的核求值的 $N\times N$ 矩阵(见习题 12.2)。

这些模型非常一般,例如,包括在第 5 章和第 9 章讨论的,并在 Wahba(1990)及 Hastie和 Tibshirani(1990)更详细阐述的整个光滑样条族、加法和交互样条模型。它们可以更一般地表示为:

$$\min_{f\in\mathcal{H}}\sum_{i=1}^{N}[1-y_if(x_i)]_+ +\lambda J(f) \tag{12.30}$$

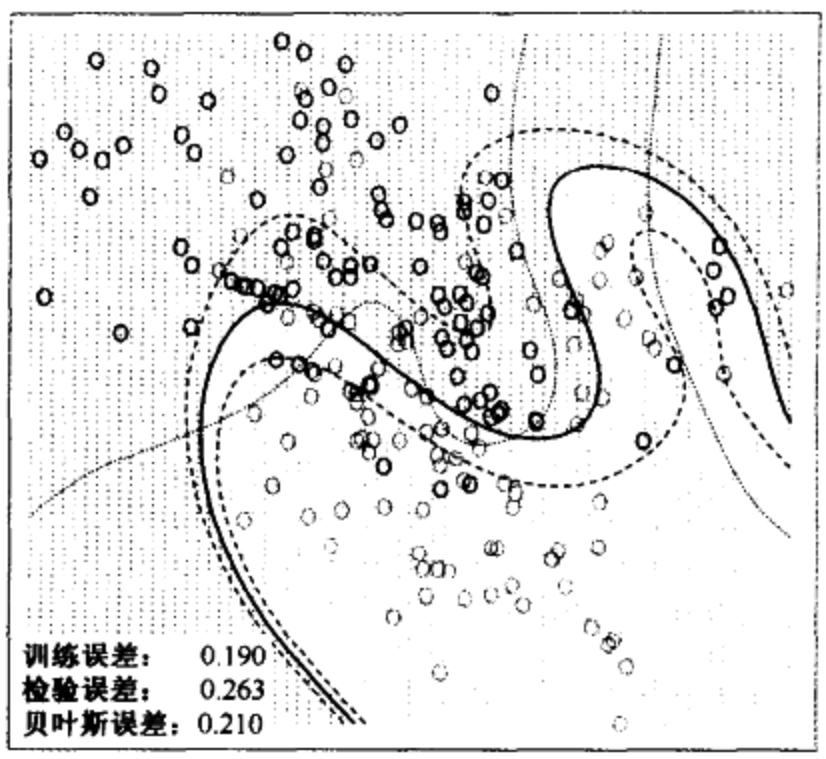
其中,$\mathcal{H}$ 是结构化的函数空间,$J(f)$ 是该空间上适当的正则化子。例如,假设 $\mathcal{H}$ 是加法函数 $f(x)=\sum_{j=1}^{p}f_j(x_j)$ 空间,而 $J(f)=\sum_j\int|f_j''(x_j)|^2\mathrm{d}x_j$,那么式(12.30)的解是一个加法三次样条,并具有核表示(12.28),其中 $K(x,x')=\sum_{j=1}^{p}K_j(x_j,x_j')$。每个 K_j 是 x_j 上的一元光滑样条的核(Wahba,1990)。

反过来,这里的讨论也表明,如上面式(12.22)讨论的任意核可以与任意凸损失函数一起使用,也将导致形如式(12.28)的有限维表示。除使用二项式对数似然做损失函数外,图 12.5 使用了与图 12.3 中相同的核函数[①]。因此,拟合函数是对数几率估计:

① Ji Zhu 帮助准备这些例子。

$$\begin{aligned}\hat{f}(x) &= \log\frac{\hat{\Pr}(Y=+1|x)}{\hat{\Pr}(Y=-1|x)} \\ &= \hat{\beta}_0+\sum_{i=1}^{N}\hat{\alpha}_i K(x,x_i)\end{aligned} \tag{12.31}$$

LR-特征空间上的4次多项式



LR-特征空间上的径向核

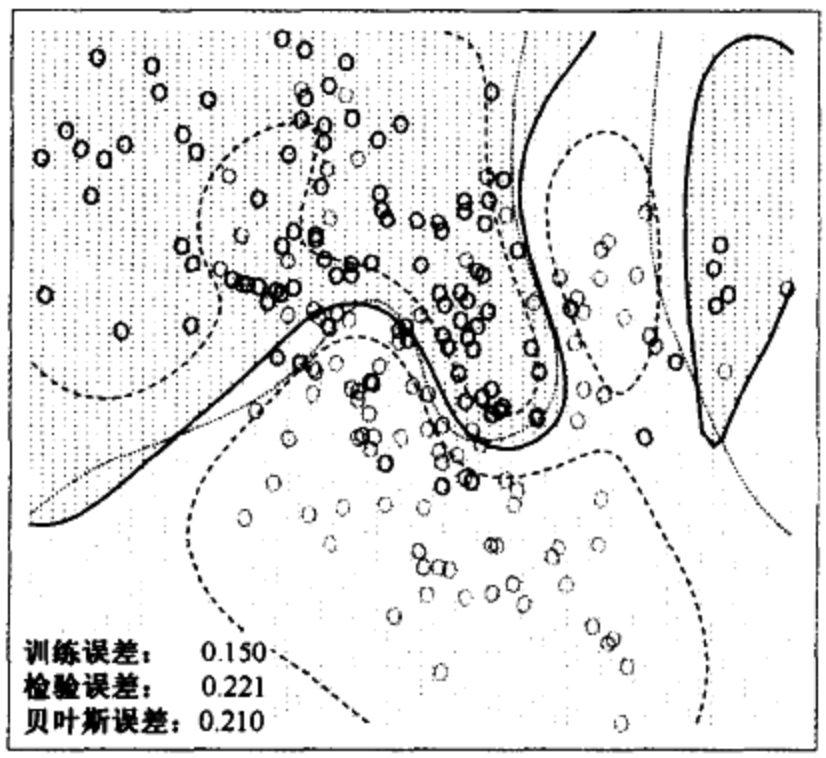


图 12.5　图 12.3 的 SVM 模型的逻辑斯缔回归版本，使用了同样的核，因此有相同的罚，但使用了对数似然损失函数，而不是SVM损失函数。两个虚线轮廓相当于 + 1类的0.75和0.25后验概率。背景上的紫色虚线是贝叶斯判定边界（见彩页）

或相反，我们得到类概率的一个估计：

$$\hat{\Pr}(Y=+1|x)=\frac{1}{1+e^{-\hat{\beta}_0-\sum_{i=1}^{N}\hat{\alpha}_i K(x,x_i)}} \tag{12.32}$$

这些拟合模型在外观和性能上很相似。例子和更详细的描述已经在第 5.8 节给出过。

对于 SVM，在 N 个α_i 值中，确实有相当大的部分可能为 0(非支持点)。在图 12.3 的两个例子中，取 0 的α_i 分别占 42% 和 45%。这是准则(12.25)第一部分的分段线性性质的推论。(在训练数据上)类重叠率越低，这一部分就越大。通常，缩小 λ 可以减少重叠(允许更灵活的 f)。减少支持点的数量意味着可以更快地计算 $\hat{f}(x)$，这对查找时间很重要。当然，过多地减少重叠会导致较差的泛化。

12.3.4 SVM 和维灾难

本节，我们讨论 SVM 是否面临维灾难问题。注意，在(12.23)中，不允许幂和积空间上完全一般的内积。例如，所有形如 $2X_jX'_j$ 的项赋予相同的权值，并且核不能自适应以便集中在子空间上。如果特征的数目 p 很大，但类分离仅发生在由 X_1 和 X_2 生成的线性子空间中，这个核将很难发现该结构，并可能不得不在很多维上搜索。我们必须把关于该子空间的知识构建到核中，即告诉它忽略除前两个输入之外的全部输入。如果有这样的先验知识，则大部分统计学习就会变得容易得多。自适应方法的主要目标就是发现这种结构。

我们讲解一个例子来支持这些陈述。在两个类中各产生 100 个观测。第一个类有 4 个标准正态的独立特征 X_1，X_2，X_3，X_4；第二个类也有 4 个标准正态的独立特征，但受条件 $9 \leqslant \sum X_j^2 \leqslant 16$ 的约束。这是一个相对简单的问题。作为一个稍难的问题，我们用 6 个标准高斯噪声特征增广这些特征。因此，在 4 维子空间中，第二个类几乎完全包围了第一个类，如同包围橘子的皮。该问题的贝叶斯误差是 0.029(不考虑维)。我们产生 1000 个检验观测以比较不同的过程。有和没有噪声特征的 50 次模拟的平均检验误差显示在表 12.2 中。

表 12.2 橘子皮：所显示的是 50 次模拟上的检验误差均值(均值的标准误差)。BRUTO 自适应地拟合一个加法样条模型，而 MARS 自适应地拟合一个低阶交互模型

方法	检验误差(SE)	
	无噪声特征	6 个噪声特征
1 SV 分类器	0.450(0.003)	0.472(0.003)
2 SVM/2 维多项式核	0.078(0.003)	0.152(0.004)
3 SVM/5 维多项式核	0.180(0.004)	0.370(0.004)
4 SVM/10 维多项式核	0.230(0.003)	0.434(0.002)
5 BRUTO	0.084(0.003)	0.090(0.003)
6 MARS	0.156(0.004)	0.173(0.005)
贝叶斯	0.029	0.029

第 1 行在原特征空间中使用支持向量分类器。第 2～4 行涉及到具有 2、5 和 10 维多项式核的支持向量机。对所有支持向量过程，我们选择调整参数 C 来极小化检验误差，尽可能对方法公平。第 5 行使用最小二乘方，用加法样条模型拟合(-1，$+1$)响应，使用的是在 Hastie 和 Tibshirani(1990)中描述的加法模型算法 BRUTO。第 6 行使用了 MARS(多元自适应回归样条)，允许在所有的阶交互，如第 9 章所示；这样它可以与 SVM/10 维多项式比较。BRUTO 和 MARS 都有忽略冗余变量的能力。在第 5 行或第 6 行中，没有使用检验误差来选择光滑参数。

在原特征空间中，超平面不能分离诸类，并且支持向量分类器（第 1 行）也不能很好地分离类。多项式支持向量机对检验误差率做出了实质性的改进，但它受到了 6 个噪声特征的不利影响。它对核的选择也很敏感：二次多项式核（第 2 行）做得最好，因为真实的判定边界是一个二次多项式。然而，高次多项式核（第 3 行和第 4 行）做得特别差。BRUTO 的性能很好，因为边界是加法的。BRUTO 和 MARS 能够很好地自适应，它们的性能并不因为噪声的出现而恶化。

12.3.5　回归的支持向量机

本节，我们展示如何以继承 SVM 分类器的某些特性的方式，将 SVM 用于具有一个定量响应的回归。首先，讨论线性回归模型：

$$f(x) = x^T\beta + \beta_0 \tag{12.33}$$

然后，处理它的非线性推广。为了估计 β，我们考虑极小化：

$$H(\beta, \beta_0) = \sum_{i=1}^{N} V(y_i - f(x_i)) + \frac{\lambda}{2}\|\beta\|^2 \tag{12.34}$$

其中：

$$V_\epsilon(t) = \begin{cases} 0 & \text{如果} |t| < \epsilon \\ |r| - \epsilon & \text{其他} \end{cases} \tag{12.35}$$

这是一种“ϵ不敏感”误差度量，忽略小于ϵ的误差（见图 12.6 左图）。这与支持向量分类大致相似，其中，在判定边界的正确侧的点和远离它的点在优化过程中被忽略。对于回归，这些“低误差”点就是一些具有较小残差的点。

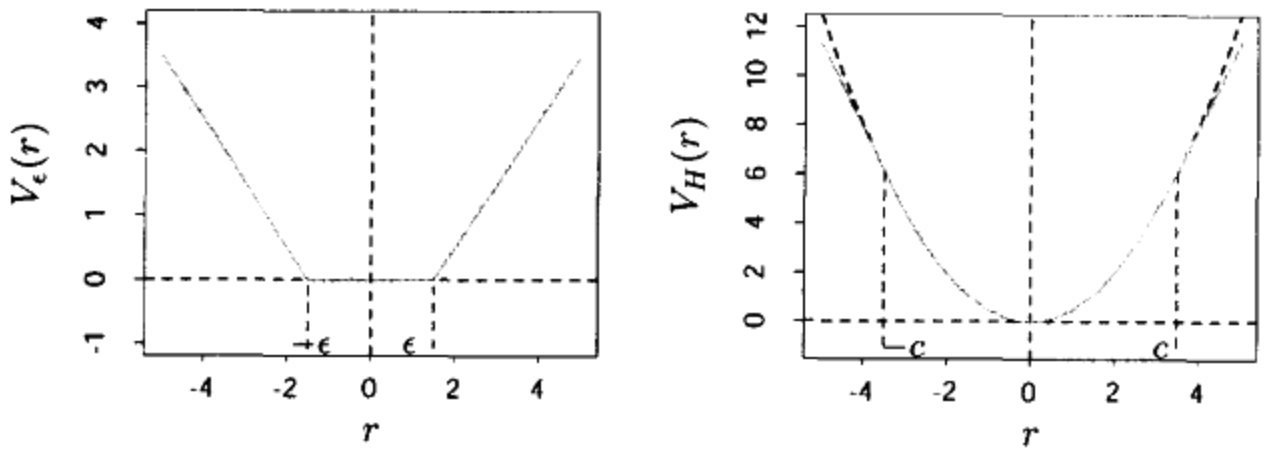


图 12.6　左图显示支持向量回归机使用的 ϵ 不敏感误差函数。右图显示 Huber 的健壮回归使用的误差函数（曲线）。在 $|c|$ 之外，函数由二次的转变为线性的

将此与统计学健壮回归使用的误差度量进行对比会很有趣。统计学最流行误差度量源自 Huber（1964），具有如下形式：

$$V_H(r) = \begin{cases} r^2/2 & \text{如果} |r| \leqslant c \\ c|r| - c^2/2 & |r| > c \end{cases} \tag{12.36}$$

显示在图 12.6 的右图中。这个函数将绝对残差大于预先选定常量 c 的观测的贡献由二次降到线性。这使得拟合对孤立点不太敏感。支持向量误差度量（12.35）也有线性尾（超出ϵ），但它还使具有较小残差的那些实例的贡献变得平坦。

如果 $\hat{\beta},\hat{\beta}_0$ 是 H 的极小化,可以证明解函数具有如下形式:

$$\hat{\beta} = \sum_{i=1}^{N}(\hat{\alpha}_i^* - \hat{\alpha}_i)x_i \tag{12.37}$$

$$\hat{f}(x) = \sum_{i=1}^{N}(\hat{\alpha}_i^* - \hat{\alpha}_i)\langle x, x_i\rangle + \beta_0 \tag{12.38}$$

其中,$\hat{\alpha}_i,\hat{\alpha}_i^*$ 是正的,且求解二次规划问题:

$$\min_{\alpha_i,\alpha_i^*} \epsilon\sum_{i=1}^{N}(\alpha_i^* + \alpha_i) - \sum_{i=1}^{N} y_i(\alpha_i^* - \alpha_i) + \frac{1}{2}\sum_{i,i'=1}^{N}(\alpha_i^* - \alpha_i)(\alpha_{i'}^* - \alpha_{i'})\langle x_i, x_{i'}\rangle$$

受限于约束:

$$\begin{gathered} 0 \leqslant \alpha_i,\ \alpha_i^* \leqslant 1/\lambda \\ \sum_{i=1}^{N}(\alpha_i^* - \alpha_i) = 0 \\ \alpha_i\alpha_i^* = 0 \end{gathered} \tag{12.39}$$

由于这些约束的特性,解值$(\hat{\alpha}_i{}^* - \hat{\alpha}_i)$只有一个子集是非零的,而相关联的数据值称为支持向量。与分类情况相同,解只通过内积$\langle x_i, x_{i'}\rangle$依赖于输入值。这样,通过定义适当的内积,如式(12.22)中定义的那样,可以将方法推广到更丰富的空间。

注意,参数 ϵ 和 λ 与准则(12.34)相关联。它们似乎起不同的作用,ϵ 是损失函数 V_ϵ 的参数,正如 c 是 V_H 的参数一样。注意,V_ϵ 和 V_H 都依赖y的标度,因此也依赖 r 的标度。如果我们缩放响应[因此改为使用 $V_H(r/\sigma)$和 $V_\epsilon(r/\sigma)$],则可能考虑使用 c 和ϵ 的预置值(值$c = 1.345$达到高斯效率的 95%)。量 λ 是一个较传统的正则化参数,例如,可以用交叉验证来估计。

12.3.6 回归和核

如第 12.3.3 节的讨论,对支持向量机来说这种核性质并非惟一的。假设我们考虑用基函数集$\{h_m(x)\}(m = 1,2,\cdots,M)$表示回归函数的逼近,

$$f(x) = \sum_{m=1}^{M}\beta_m h_m(x) + \beta_0 \tag{12.40}$$

为了估计 β 和β_0,对于某种一般误差度量 $V(r)$,极小化:

$$H(\beta,\beta_0) = \sum_{i=1}^{N} V(y_i - f(x_i)) + \frac{\lambda}{2}\sum\beta_m^2 \tag{12.41}$$

对任意选取的 $V(r)$,解 $\hat{f}(x) = \sum\hat{\beta}_m h_m(x) + \hat{\beta}_0$ 具有如下形式:

$$\hat{f}(x) = \sum_{i=1}^{N}\hat{a}_i K(x, x_i) \tag{12.42}$$

其中,$K(x,y) = \sum_{m=1}^{M} h_m(x)h_m(y)$。注意,这与第 5 章和第 6 章讨论的径向基函数展开式和

正则化估计具有相同的形式。

为了更具体，让我们考虑 $V(r) = r^2$ 的情况。令 $\mathbf{H}$ 是 $N \times M$ 基矩阵，第 im 个元素为 $h_m(x_i)$，并假设 $M > N$ 比较大。为简单起见，我们假设 $\beta_0 = 0$，或 β_0 为常量被 h 吸收；另外的情况参见习题 12.3。

我们通过极小化罚最小二乘方标准来估计 β：

$$H(\beta) = (\mathbf{y} - \mathbf{H}\beta)^T(\mathbf{y} - \mathbf{H}\beta) + \lambda\|\beta\|^2 \tag{12.43}$$

解是：

$$\hat{\mathbf{y}} = \mathbf{H}\hat{\beta} \tag{12.44}$$

$\hat{\beta}$ 由下式确定：

$$-\mathbf{H}^T(\mathbf{y} - \mathbf{H}\hat{\beta}) + \lambda\hat{\beta} = 0 \tag{12.45}$$

由此看来，我们需要在变换空间中计算 $M \times M$ 内积矩阵。然而，可以用 $\mathbf{H}$ 左乘上式，得到：

$$\mathbf{H}\hat{\beta} = (\mathbf{H}\mathbf{H}^T + \lambda\mathbf{I})^{-1}\mathbf{H}\mathbf{H}^T\mathbf{y} \tag{12.46}$$

$N \times N$ 矩阵 $\mathbf{H}\mathbf{H}^T$ 由观测对 i，i' 之间的内积组成，即内积核 $\{\mathbf{H}\mathbf{H}^T\}_{i,i'} = K(x_i, x_{i'})$ 的求值。在此情况下容易直接证明，式(12.42)在任意 x 上的预测值满足：

$$\begin{aligned} \hat{f}(x) &= h(x)^T\hat{\beta} \\ &= \sum_{i=1}^{N}\hat{\alpha}_i K(x, x_i) \end{aligned} \tag{12.47}$$

其中 $\hat{\alpha} = (\mathbf{H}\mathbf{H}^T + \lambda\mathbf{I})^{-1}\mathbf{y}$。与支持向量机一样，我们不需要指定或计算一个很大的函数集 $h_1(x)$，$h_2(x)$，…，$h_M(x)$。只需要在 N 个训练点的每对 i，i' 和预测点 x 上计算内积核 $K(x_i, x_{i'})$。谨慎选择 h_m（如，特别的本征函数，易求值的核 K）意味着计算 $\mathbf{H}\mathbf{H}^T$ 以 K 的 $N^2/2$ 次求值为代价，而非直接代价 N^2M。

然而需要注意，该性质依赖于罚中平方范数 $\|\beta\|^2$ 的选择。例如，对于 L_1 范数 $|\beta|$，上面的结果并不成立；L_1 范数可能导致更好的模型。

12.3.7　讨论

支持向量机可以扩展到多类问题，本质上是通过求解多个 2-类问题。为每两个类构造一个分类器，而最终的分类器是最有优势的那一个（Kressel，1999、Friedman 1996、Hastie 和 Tibshirani，1998）。替换地，我们可以像第 12.3.3 中一样，将多项式损失函数与合适的核一起使用。SVM 在许多其他有指导和无指导学习问题中具有广泛的应用。经验表明它在许多实际学习问题中表现得很好。

最后，我们提一下支持向量机和结构风险极小化 (7.9)之间的联系。假设训练点（或它们的基展开）包含在一个半径为 R 的球形中，如同在式(12.2)中那样，令 $G(x) = \text{sign}[f(x)] = \text{sign}[\beta^T x + \beta_0]$。则可以证明函数类 $\{G(x), \|\beta\| \leqslant A\}$ 具有满足下式的 VC 维 h：

$$h \leqslant R^2A^2 \tag{12.48}$$

如果 $f(x)$ 分离训练数据，对 $\|\beta\| \leqslant A$ 来说是最优的，则在训练集上至少以概率 $1 - \eta$ 有（Vap-

nik，1996)：

$$\text{Error}_{\text{Test}} \leqslant 4\frac{h(\log(2N/h)+1)-\log(\eta/4)}{N} \tag{12.49}$$

支持向量分类器是最实用的学习过程之一，可以得到它在 VC 维上有用的上界，因此可以执行 SRM 程序。然而，在推导过程中，球置于数据点周围——一个依赖于特征观测值的过程。因此，严格地讲，类的 VC 复杂性在看到特征之前，不是固定的先验。

正则化参数 γ 控制分类器的 VC 维的上界。按照 SRM 范例，我们可以通过极小化检验误差上界来选择 γ，在式(12.49)中已给出。然而，与通过交叉验证选择 γ 相比，这样做有什么优点还不清楚。

12.4 线性判别分析的推广

在第 4.3 节，我们讨论了线性判别分析(LDA)，它是分类的基础性工具。本章余下部分讨论一类技术，通过直接推广 LDA 来产生比 LDA 更好的分类器。

LDA 的一些优点如下：

- 它是一个简单的原型分类器，它将新观测分类到具有最近中心点的类。一个小技巧就是距离用 Mahalanobis 度量，使用合并的协方差估计。
- 如果在每一个类中观测都是多元高斯的，且具有一个公共的协方差矩阵，则 LDA 是估计的贝叶斯分类器。由于这种假设未必为真，所以这一点看来并不是多大的优点。
- LDA 建立的判定边界是线性的，导致决策规则易于描述和实现。
- LDA 提供了数据自然的低维视图。例如，图 12.10 是 256 维数据的富含信息的二维视图，数据具有 10 个类。
- 基于 LDA 的简单性和低方差，它通常能产生最好的分类结果。对于 STATLOG 项目中研究的 22 个数据集中的 11 个，LDA 是三个最好的分类器之一(Michie 等人，1994)[①]。

遗憾的是，LDA 的简单性也使它在一些情况下无效：

- 通常，线性判定边界并不足以分离类。当 N 较大时，可能要估计较复杂的判定边界。这时，二次判别分析(QDA)通常是有用的，它允许二次判定边界。更一般地，我们希望能够对不规则的判定边界进行建模。
- 前面提到 LDA 的不足通常可以解释为：每个类仅一个单一原型是不够充分的。LDA 使用单一原型(类中心点)加上一个公共协方差矩阵描述每个类的数据分布。在许多情况下，多个原型更合适一些。
- 在这个谱系的另一端，在某些情况下(例如，数字化模拟信号和图像处理)我们可能有太多(相关的)预测子。在这种情况下，LDA 使用了太多的参数，以较高的方差估计这些参数，而影响了它的性能。在这样一些情况下，我们需要更进一步限制或正则化 LDA。

在本章的剩余部分，我们将阐述一类技术，旨在解决因推广 LDA 模型带来的所有问题，这主要通过三种不同想法来实现。

① 该研究在 SVM 出现之前。

第一种想法是把 LDA 问题重新改造成线性回归问题。许多技术都是为了把线性回归推广成更加灵活的非参数回归形式。这样导致了判别分析更加灵活的形式，我们称之为 FDA。在大部分感兴趣的情况下，可以看到回归过程是通过基展开来识别一个扩大的预测子的集合。在这个扩大的空间上，FDA 等同于 LDA，与 SVM 使用相同的范例。

在预测子过多的情况下（如数字化图像的像素），我们不想扩大这个集合，因为它已经太大了。第二种想法是拟合一个 LDA 模型，但惩罚它的系数使其在空间域中成为光滑或粘合的；即，如同一个图像。我们称这个过程为罚判别分析（penalized discriminant analysis，PDA）。对 FDA 本身，扩展的基集通常太大以至于也需要正则化（又与 SVM 一样）。这两种想法都可以在 FDA 模型的背景下通过适当正则化的回归来实现。

第三种想法是通过混合有不同中心点的两个或多个高斯来对每个类进行建模，但每个分量高斯（类内或类间）共享相同的协方差矩阵。这允许更复杂的判定边界，也允许像在 LDA 中那样约化子空间。我们称这种扩展为混合判定分析（mixture discriminant analysis，MDA）。

这三种泛化都使用了一个共同的框架，即利用了它们与 LDA 的联系。

12.5　柔性判别分析

本节，我们介绍一种在导出响应上使用线性回归执行 LDA 的方法。这导致 LDA 的非参数的、灵活的替代方法。与第 4 章一样，假设观测具有定量响应 G，落入 K 个类 $\mathcal{G}=\{1,\cdots,K\}$ 中的一个，每个观测都具有度量特征 X。假设 $\theta:\mathcal{G}\to\mathbb{R}^1$ 是一个函数，它将得分赋予这些类，使得变换后的类标号被 X 上的线性回归最优地预测：如果我们的训练样本形式是 (g_i,x_i)，$i=1,2,\cdots,N$，则我们求解：

$$\min_{\beta,\theta}\sum_{i=1}^{N}\left(\theta(g_i)-x_i^T\beta\right)^2 \tag{12.50}$$

使用 θ 上的限制以避免平凡解（训练数据上的均值 0 和单位方差）。这将在类之间产生一个一维分离。

更一般地，我们可以对 $L\leqslant K-1$，为类标号找 L 个独立的评分集 $\theta_1,\theta_2,\cdots,\theta_L$，并选取 L 个对应的线性映射 $\eta_\ell(X)=X^T\beta_\ell$，$\ell=1,\cdots,L$，对于 $\mathbb{R}^p$ 上的多元回归，它们是最优的。选择评分 $\theta_\ell(g)$ 和映射 β_ℓ 使得平均平方残差极小，

$$ASR=\frac{1}{N}\sum_{\ell=1}^{L}\left[\sum_{i=1}^{N}\left(\theta_\ell(g_i)-x_i^T\beta_\ell\right)^2\right] \tag{12.51}$$

假定关于适当的内积，得分集是相互正交和规格化的，以避免产生平凡的零解。

我们为什么走这条路呢？可以证明第 4.3.3 节中导出的判别（典范）向量 ν_ℓ 序列与序列 β_ℓ 是相同的，相差一个常量（Mardia 等人，1979、Hastie 等人，1995）。另外，一个检验点 x 和第 k 个类中心 $\hat{\mu}_k$ 的 Mahalanobis 距离由下式给出：

$$\delta_J(x,\hat{\mu}_k)=\sum_{\ell=1}^{K-1}w_\ell(\hat{\eta}_\ell(x)-\bar{\eta}_\ell^k)^2+D(x) \tag{12.52}$$

其中，$\bar{\eta}_\ell^k$ 是第 k 个类中 $\hat{\eta}_\ell(x_i)$ 的均值，而 $D(x)$ 不依赖于 k。这里，w_ℓ 是坐标权值，它用第 ℓ

个最优评分拟合的均方残差 r_ℓ^2 定义：

$$w_\ell = \frac{1}{r_\ell^2(1-r_\ell^2)} \tag{12.53}$$

在第 4.3.2 节，我们看到对于每个类具有相同的协方差高斯分布，这些典范距离正是分类需要的。简言之：

> LDA 可以通过一个线性回归序列，后随到拟合空间最近的类中心的分类来进行。模拟既可以应用于降秩的情况，又可以应用于满秩的情况（$L = K-1$）。

这个结果的实际能力在于它引起的推广。我们可以用更加灵活、非参数拟合替换线性回归拟合 $\eta_\ell(x) = x^T\beta_\ell$，并且通过模拟实现比 LDA 更灵活的分类器。我们已经有广义的加法拟合、样条函数、MARS 模型等。在这种更加一般的形式下，回归问题可以通过以下准则来定义：

$$ASR(\{\theta_\ell, \eta_\ell\}_{\ell=1}^L) = \frac{1}{N}\sum_{\ell=1}^{L}\left[\sum_{i=1}^{N}(\theta_\ell(g_i) - \eta_\ell(x_i))^2 + \lambda J(\eta_\ell)\right] \tag{12.54}$$

其中，J 是一个正则化子，适用于某种形式的非参数回归，如光滑样条、加法样条和低阶ANOVA样条模型。还包括由核产生的函数类和相关罚，如第 12.3.3 节中所述。

在我们描述这种推广涉及的计算之前，先考虑一个简单的例子。假设我们对每个 η_ℓ 使用二次多项式回归。由式(12.52)给出的判定边界将是二次曲面，因为每个拟合函数是二次的，并且如同在 LDA 中一样，当比较距离时，约去它们的平方。实际上也可以用更常规的方式得到等价的二次边界，用预测子的平方和叉积增广原预测子。在扩大的空间中，我们可以执行 LDA，并将扩大的空间中的线性边界映射到原空间中的二次边界。一个经典的例子是中心在原点的多元高斯对，一个具有协方差矩阵 I，而另一个为 $cI(c>1)$，如图 12.7 所示。贝叶斯判定边界是球 $\|x\| = \frac{pc\log c}{2(c-1)}$，它是扩大空间中的线性边界。

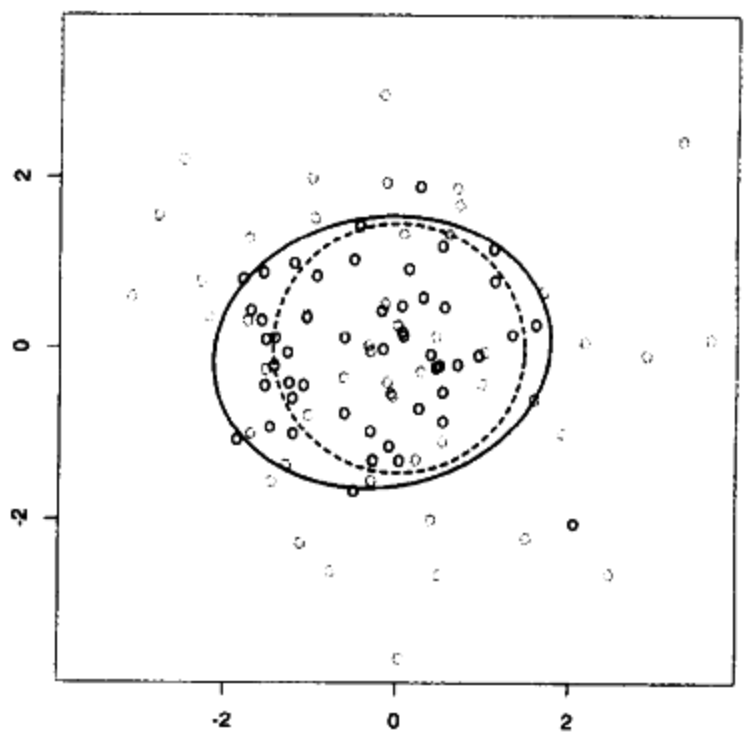


图 12.7 数据由 50 个点组成，每个数据点由 $N(0,I)$ 和 $N\left(0,\frac{9}{4}I\right)$ 产生。黑色实线椭圆是使用二次多项式回归的FDA发现的判定边界。紫色虚线圆是贝叶斯判定边界（见彩页）

许多非参数回归过程都是通过产生导出变量的基展开，然后在扩大的空间中进行线性回归来实现的。MARS 过程(见第 9 章)正是这种形式。光滑样条和加法样条模型产生一个非常大的基集(加法样条有 $N \times p$ 个基函数)，而之后在扩大的空间中进行罚回归拟合。SVM 也是如此，见第 12.3.6 节基于核回归的例子。在这种情况下，FDA 可以看做在扩大的空间中进行罚线性判别分析。在第 12.6 节中我们将详尽叙述。扩大空间中的线性边界可以映射到缩小空间中的非线性边界。这恰好和支持向量机所使用的范例是相同的(见第 12.3 节)。

我们用第 4 章的语音识别例子来解释 FDA，使用 $K = 11$ 个类和 $p = 10$ 个预测子。这些类对应 11 个元音，每个元音包含在 11 个不同的单词中。这里给出这些单词，由代表它们的符号做先导：

元音	单词	元音	单词	元音	单词	元音	单词
i:	heed	O	hod	I	hid	C:	hoard
E	head	U	hood	A	had	u:	who'd
a:	hard	3:	heard	Y	hud		

在训练集中，8 个人每人读每个单词 6 次；同样，在检验集中的 7 个人也照此办理。10 个预测子以非常复杂的方式从数字化语音中获得，但在语言识别领域是标准的。有 528 个这样的训练观测，462 个检验观测。图 12.8 显示了由 LDA 和 FDA 产生的二维投影。FDA 模型使用自适应加法样条回归函数模拟 $\eta_\ell(x)$，并且画在右侧的点以 $\hat{\eta}_1(x_i)$和 $\hat{\eta}_2(x_i)$为坐标。使用的 S-PLUS 中的例程叫做 bruto，因此它出现在图的标题和表 12.3 中。我们看到在这种情况下，灵活的建模有助于分离类。表 12.3 显示了一些分类技术的训练和检验误差率。FDA/MARS 引用 Friedman 的多元自适应回归样条。次 = 2 意味着两两积是许可的。注意，对于 FDA/MARS，最好的分类结果是在降秩子空间中得到的。

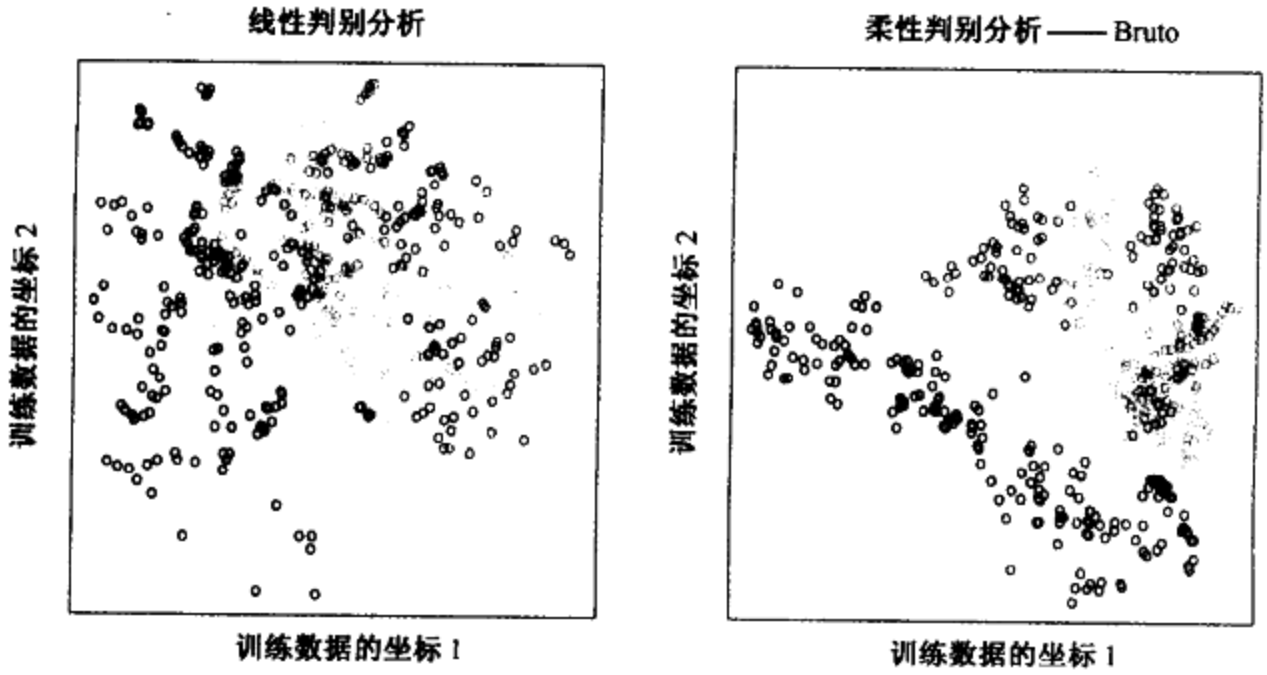


图 12.8　左图显示元音训练数据的前两个 LDA 正规变量。右图显示当 FDA/BRUTO用于拟合模型时相应的投影；所描绘的是拟合回归函数 $\hat{\eta}_1(x_i)$和 $\hat{\eta}_2(x_i)$。注意改进的分离。字母标记元音(见彩页)

表 12.3 元音识别数据执行结果。在更大的集合中，神经网络的结果是最好的，其结果取自神经网络档案。记号 FDA/BRUTO 指出与 FDA 一起使用的回归方法

	技术	误差率	
		训练	检验
(1)	LDA	0.32	0.56
	Softmax	0.48	0.67
(2)	QDA	0.01	0.53
(3)	CART	0.05	0.56
(4)	CART(线性组合分裂)	0.05	0.54
(5)	单层感知器		0.67
(6)	多层感知器(88 个隐藏单元)		0.49
(7)	高斯节点网络(528 个隐藏单元)		0.45
(8)	最近邻		0.44
(9)	FDA/BRUTO	0.06	0.44
	Softmax	0.11	0.50
(10)	FDA/MARS(次 = 1)	0.09	0.45
	最佳约化维(= 2)	0.18	0.42
	Softmax	0.14	0.48
(11)	FDA/MARS(次 = 2)	0.02	0.42
	最佳约化维(= 6)	0.13	0.39
	Softmax	0.10	0.50

12.5.1 计算 FDA 估计

FDA 坐标的计算在许多重要情况下可以简化，特别是当非参数回归过程可以表示为一个线性算子时。我们将用 $\mathbf{S}_\lambda$ 表示该算子；即 $\hat{\mathbf{y}} = \mathbf{S}_\lambda \mathbf{y}$，其中 $\mathbf{y}$ 是响应向量，$\hat{\mathbf{y}}$ 是拟合向量。如果光滑参数是固定的，加法样条就具有该性质，与基函数被选定时的 MARS 一样。下标 λ 表示光滑参数的全集。在这种情况中，最优评分等价于一个标准相关问题，而且解可以通过一个单一的本征分解来计算。该问题在习题 12.6 中将继续讨论，而结果算法在这里给出。

我们从响应 g_i 创建一个 $N \times K$ 指示子响应矩阵 $\mathbf{Y}$，使得如果 $g_i = k$，则 $y_{ik} = 1$，否则 $y_{ik} = 0$。对于一个 5-类问题，$\mathbf{Y}$ 看上去可以有如下形式：

$$\begin{array}{c} \\ g_1 = 2 \\ g_2 = 1 \\ g_3 = 1 \\ g_4 = 5 \\ g_5 = 4 \\ \vdots \\ g_N = 3 \end{array} \begin{array}{c} \begin{array}{ccccc} C_1 & C_2 & C_3 & C_4 & C_5 \end{array} \\ \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ & & \vdots & & \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix} \end{array}$$

这里给出计算步骤：

1. 多元非参数回归。在 $\mathbf{X}$ 上拟合一个 $\mathbf{Y}$ 的多响应、自适应非参数回归，给出拟合值 $\hat{\mathbf{Y}}$。令 $\mathbf{S}_\lambda$ 是拟合最终选取的模型的线性算子，$\eta^*(x)$是拟合回归函数的向量。
2. 最优得分。计算 $\mathbf{Y}^T\hat{\mathbf{Y}} = \mathbf{Y}^T\mathbf{S}_\lambda\mathbf{Y}$ 的本征分解，其中本征向量 Θ 被规范化：$\Theta^T\mathbf{D}_\pi\Theta = \mathbf{I}$。这里，$\mathbf{D}_\pi = \mathbf{Y}^T\mathbf{Y}/N$ 是估计类先验概率的对角矩阵。
3. 使用最佳评分，更新步骤 1 得到的模型：$\eta(x) = \Theta^T\eta^*(x)$。

$\eta(x)$中，K 个函数中的第一个是常量函数——平凡解；余下的 $K-1$ 个函数是判别函数。该常量函数和规范化一起使得余下的全部函数中心化。

$\mathbf{S}_\lambda$ 还可以对应任何回归方法。当 $\mathbf{S}_\lambda = \mathbf{H}_X$ 时，它是线性回归投影算子，FDA 是线性判别分析。我们在本章的“计算考虑”中提到的软件很好地利用了这种模块性；fda 函数有一个自变量 method =，允许提供任意回归函数，只要它符合一些自然约定。我们提供的回归函数可以是多项式回归、自适应加法模型和 MARS。它们都能有效地处理多元响应，所以步骤(1)是对回归子程序的一次调用。步骤(2)的本征分解同时计算所有最佳评分函数。

在第 4.2 节，我们讨论了在指示器响应矩阵上使用线性回归作为分类方法的缺陷。特别地，对三个或更多的类会发生严重的屏蔽。FDA 在步骤(1)使用了这样的回归拟合，但是，之后对它们进一步变换以产生没有这种缺陷的、有用的判别函数。习题 12.9 取这种现象的另外一种观点。

12.6　罚判别分析

尽管 FDA 是被推广最优评分推动的，但是也可以直接把它视为一种正则化判别分析形式。假设用于 FDA 的回归过程相当于在一个基展开 $h(X)$上的线性回归，在系数上具有二次罚：

$$ASR(\{\theta_\ell, \beta_\ell\}_{\ell=1}^L) = \frac{1}{N}\sum_{\ell=1}^{L}\left[\sum_{i=1}^{N}(\theta_\ell(g_i) - h^T(x_i)\beta_\ell)^2 + \lambda\beta_\ell^T\Omega\beta_\ell\right] \tag{12.55}$$

Ω 的选择依赖于问题。如果 $\eta_\ell(x) = h(x)\beta_\ell$ 是样条基函数上的展开式，那么 Ω 可能要限制 η_ℓ 在 $\mathbb{R}^p$ 上是光滑的。在加法样条情况下，每个坐标有 N 个样条基函数，导致 $h(x)$中总共有 Np 个基函数；在这种情况下，Ω 是 $Np \times Np$ 块对角矩阵。

FDA 中的步骤也可以看做是 LDA 的一种拓广形式，我们称它为罚判别分析(penalized discriminant analysis)，PDA：

- 通过基展开 $h(X)$扩大预测子的集合 X。
- 在扩大的空间中使用(罚)LDA，其中罚 Mahalanobis 距离由下式给出：

$$D(x, \mu) = (h(x) - h(\mu))^T(\Sigma_W + \lambda\Omega)^{-1}(h(x) - h(\mu)) \tag{12.56}$$

  其中，Σ_W 是导出变量 $h(x_i)$的类内协方差矩阵。
- 使用如下罚度量来分解分类子空间：

$$\max u^T\Sigma_{\text{Bet}}u, \text{ 受限于 } u^T(\Sigma_W + \lambda\Omega)u = 1$$

宽松地说，罚 Mahalanobis 距离对“粗糙的”坐标赋予较少的权，而对于“光滑的”坐标赋予较大的权；由于罚不是对角的，所以，它在粗糙或光滑的线性组合上使用相同的做法。

对一些问题类，涉及基展开的第一步是不需要的；我们已经有相当多的（相关的）预测子。一个重要的例子是当被分类的对象是数字化模拟信号时：

- 语音片断对数 - 周期图，在一个 256 个频率的集合中抽样；见图 5.5。
- 一个手写数字的数字化图像中的灰度像素值。

在这些情况下，为什么需要正则化直观上是清楚的。以数字化图像为例，邻近的像素值趋向是相关的，常常几乎相同。这表明这些像素对应的 LDA 系数对可能是完全不同的，并且符号相反，从而在应用于相似的像素值时会抵消掉。正相关预测子导致噪声、负相关系数估计，而且这种噪声会导致不希望的样本方差。一个合理的策略是对系数正则化，使其在空间域上是光滑的，就像处理图像一样。这正是 PDA 所做的。除使用适当的罚回归方法外，计算过程与 FDA 一样。这里，$h^T(X)\beta_\ell = X\beta_\ell$，选择 Ω 以便看做图像时 $\beta_\ell^T\Omega\beta_\ell$ 能够补偿 β_ℓ 中的粗糙性。图 1.2显示了一些手写数字的例子。图 12.9 显示了使用 LDA 和 PDA 的判别变量。LDA 产生的图像看上去就像是盐加胡椒粉，而由 PDA 产生的却是光滑的图像。第一个光滑图像可以看做是线性对比函数的系数，该函数将具有黑色中间垂直带（1，或许是 7）的图像与中心为空的图像（0，有时是 4）分离开。图 12.10 支持这种解释，而且对于更多的困难允许第二维坐标的解释。本例和其他例子的详细讨论在 Hastie 等人（1995）的著作中可以找到。Hastie 等人还表明正则化改进了 LDA 在独立检验集上的分类性能，他们的实验表明大约提高了 25%。

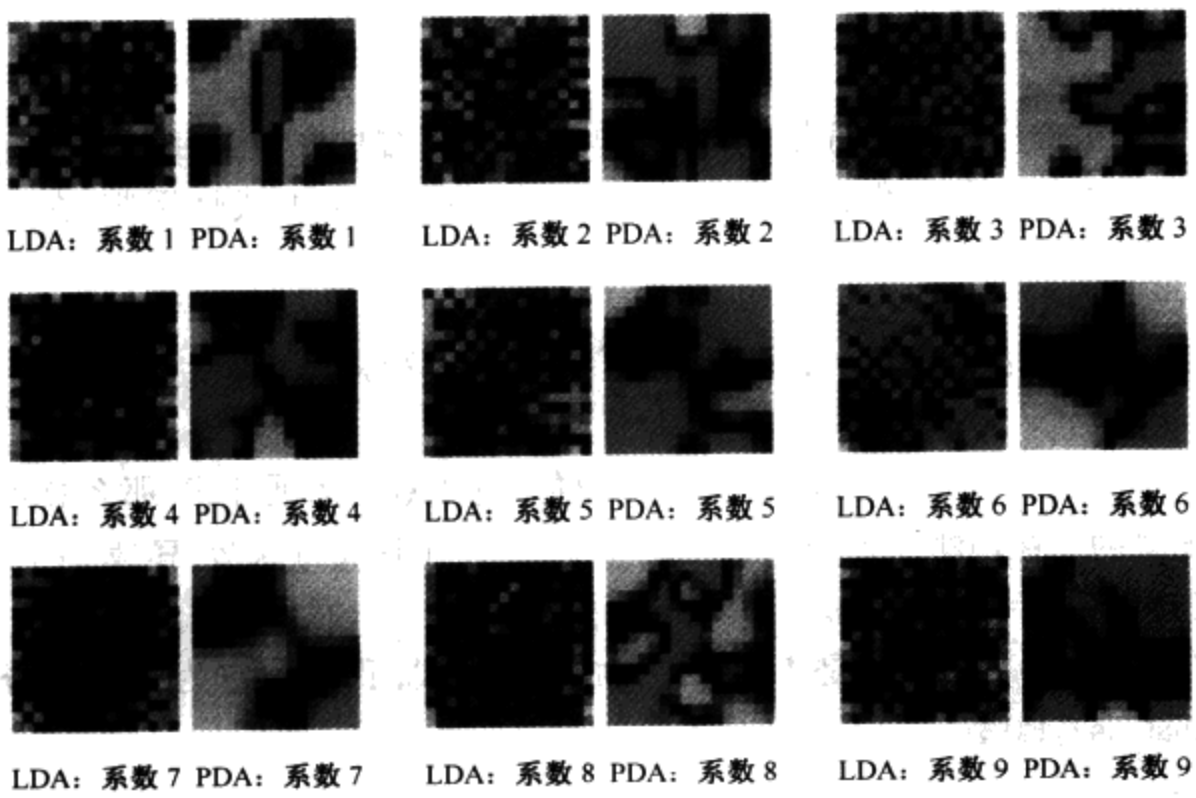


图 12.9 图像成对出现，表示数字识别问题中 9 个判别系数函数。每对中左边成员是 LDA 系数，而右边是 PDA 系数，已经正则化，以加强空间光滑性（见彩页）

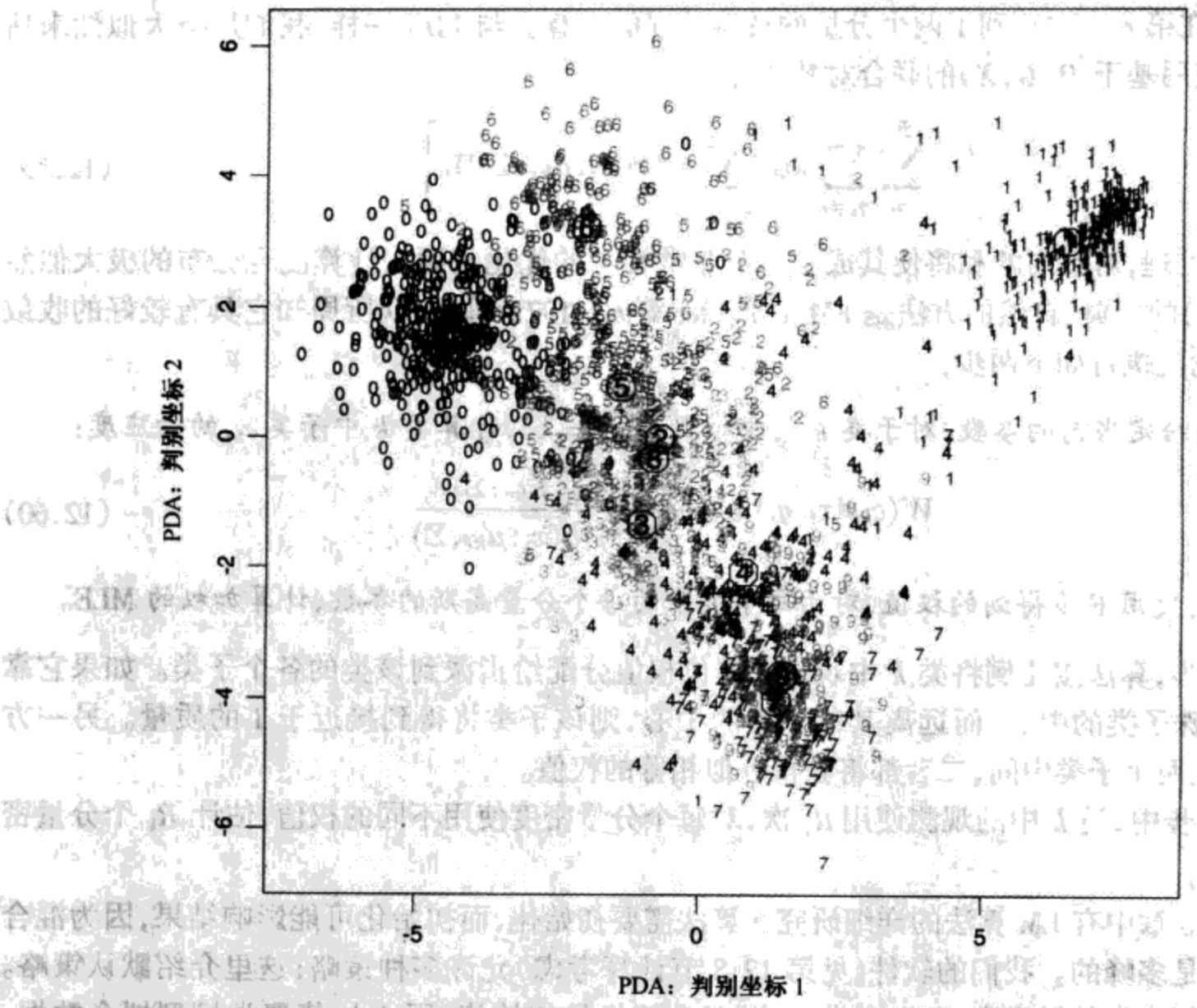


图 12.10　前两个处罚规范变量，是对检验数据求的值。圆圈指明了类的中心点。第一个坐标主要对比0和1，而第二个对比6和7/9(见彩页)

12.7　混合判别分析

线性判别分析可以看做原型(prototype)分类器。每个类用它的中心点表示，并且我们使用适当的度量分类到最近的中心点。在许多情况下，单个原型对表示不同质的类是不充分的，而混合模型则更合适些。本节，我们回顾一下高斯混合模型，并说明怎样通过前面讨论过的 FDA 和 PDA 对它们进行泛化。第 k 个类的高斯混合模型有密度：

$$P(X|G=k)=\sum_{r=1}^{R_k}\pi_{kr}\phi(X;\mu_{kr},\Sigma) \tag{12.57}$$

其中混合比例 π_{kr} 之和为 1。对于第 k 个类，有 R_k 个原型，并在我们的说明中，相同的协方矩阵 Σ 自始至终被用做度量。对每个类给定这样一个模型，则类的后验概率由下式给出：

$$P(G=k|X=x)=\frac{\sum_{r=1}^{R_k}\pi_{kr}\phi(X;\mu_{kr},\Sigma)\Pi_k}{\sum_{\ell=1}^{K}\sum_{r=1}^{R_\ell}\pi_{\ell r}\phi(X;\mu_{\ell r},\Sigma)\Pi_\ell} \tag{12.58}$$

其中，Π_k 表示类先验概率。

我们在第 8 章中看到了两个分量的特殊情况的计算。与 LDA 一样，我们用极大似然来估计参数，使用基于 $P(G,X)$ 的联合对数似然：

$$\sum_{k=1}^{K}\sum_{g_i=k}\log\left[\sum_{r=1}^{R_k}\pi_{kr}\phi(x_i;\mu_{kr},\Sigma)\Pi_k\right] \tag{12.59}$$

如果直接处理，对数内的和将使其成为一个非常棘手的优化问题。计算混合分布的极大似然估计(MLE)的经典、自然的方法是 EM(Dempster 等人，1977)算法，众所周知它具有较好的收敛性。EM 轮流执行如下两步：

E-步：给定当前的参数，对于类 k 的每个观测($g_i=k$)，计算类 k 中子类 c_{kr} 的响应度：

$$W(c_{kr}|x_i,g_i)=\frac{\pi_{kr}\phi(x_i;\mu_{kr},\Sigma)}{\sum_{\ell=1}^{R_k}\pi_{k\ell}\phi(x_i;\mu_{k\ell},\Sigma)} \tag{12.60}$$

M-步：使用 E 步得到的权值，对于每个类中的每个分量高斯的参数，计算加权的 MLE。

在 E 步，算法按比例将类 k 中观测的单位权值分配给指派到该类的各个子类。如果它靠近一个特殊子类的中心，而远离其他子类的中心，则该子类将得到接近于 1 的质量。另一方面，观测在两个子类中间，二者都将获得近似相等的权值。

在 M 步中，类 k 中的观测使用 R_k 次，对每个分量密度使用不同的权值，估计 R_k 个分量密度的参数。

在第 8 章中有 EM 算法的详细研究。算法需要初始化，而初始化可能影响结果，因为混合似然通常是多峰的。我们的软件(见第 12.8 节计算考虑)允许多种策略；这里介绍默认策略。用户提供每个类的子类数 R_k。在类 k，以多个随机的初始值，用 k 均值聚类模型拟合数据。这样就把观测划分成 R_k 个互不相交的组，由它们创建一个由 0 和 1 组成的初始矩阵。

始终使用相等的分量协方差矩阵 Σ 换取了附加的简洁性；与 LDA 一样，可以将秩限制合并到混合公式中。为理解这一点，我们回顾一下关于 LDA 的不太被人了解的事实。秩 L LDA 拟合(见第 4.3.3 节)等价于高斯模型的极大似然拟合，其中，每个类的不同均值向量被限制到 $\mathbb{R}^p$ 的一个秩 L 子空间(见习题 4.8)。我们可以继承混合模型的这种性质，并在如下条件下极大化对数似然(12.59)：所有 $\sum_k R_k$ 个中心上的秩约束为 $\text{rank}\{\mu_{kl}\}=L$。

EM 算法仍是可用的，M 步结果是 LDA 的一个加权形式，具有 $R=\sum_{k=1}^{K}R_k$ 个“类”。进一步，可以像以前一样使用最佳评分，求解加权的 LDA 问题，这允许我们在这个阶段使用 FDA 或 PDA 的加权形式。除了“类”的数量增长之外，我们期望第 k 类的“观测”数量也相似地增长一个因子 R_k。如果线性算子用于最优评分回归，则结果并非如此。在这种情况下，扩大的指示子矩阵 $\mathbf{Y}$ 坍缩成一个模糊的响应矩阵 $\mathbf{Z}$，直观上这是令人满意的。例如，假设有 $K=3$ 个类，而且每个类有 $R_k=3$ 个子类，那么 $\mathbf{Z}$ 可能是：

$$
\begin{array}{c}
 \\
g_1 = 2 \\
g_2 = 1 \\
g_3 = 1 \\
g_4 = 3 \\
g_5 = 2 \\
\vdots \\
g_N = 3
\end{array}
\begin{array}{c}
\begin{array}{ccccccccc}
c_{11} & c_{12} & c_{13} & c_{21} & c_{22} & c_{23} & c_{31} & c_{32} & c_{33}
\end{array} \\
\begin{pmatrix}
0 & 0 & 0 & 0.3 & 0.5 & 0.2 & 0 & 0 & 0 \\
0.9 & 0.1 & 0.0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0.1 & 0.8 & 0.1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0.4 & 0.1 \\
0 & 0 & 0 & 0.7 & 0.1 & 0.2 & 0 & 0 & 0 \\
 & & & \vdots & & & & & \\
0 & 0 & 0 & 0 & 0 & 0 & 0.1 & 0.1 & 0.8
\end{pmatrix}
\end{array}
\tag{12.61}
$$

其中，类 k 行上的值对应于 $W(c_{kr} \mid x, g_i)$。

剩下的步骤是相同的：

$$
\left.
\begin{array}{l}
\hat{\mathbf{Z}} = \mathbf{SZ} \\
\mathbf{Z}^T\hat{\mathbf{Z}} = \Theta\mathbf{D}\Theta^T \\
\text{更新 } \pi\text{s 和 } \Pi\text{s}
\end{array}
\right\} \text{MDA 的M步}
$$

这些简单的修改为混合模型增加了相当大的灵活性：

- 在 LDA、FDA 或者 PDA 中的降维步受类个数的限制；特别地，对于 $K=2$ 个类，不可能降维。MDA 替换类的子类，然后，允许我们考虑由这些子类中心生成的子空间的低维视图。通常，该子空间对于判别来说很重要。
- 通过在 M 步中使用 FDA 或 PDA，我们甚至可以适应更特殊的情况。例如，可以利用内置的光滑性约束，用 MDA 模型拟合数字化的模拟信号和图像。

图 12.11 在混合例子上对 FDA 和 MDA 做了对比。

12.7.1　例：波形数据

现在就一个流行的模拟例子来解释这些思想；该例子取自 Breiman(1984, pp.49 ~ 55)，并在 Hastie 和 Tibshirani(1996b)及其他地方使用。它是一个有 21 个变量的 3-类问题，并认为是一个困难的模式识别问题。预测子由下式定义：

$$
\begin{array}{rcll}
X_j &=& Uh_1(j) + (1-U)h_2(j) + \epsilon_j & \text{类 1} \\
X_j &=& Uh_1(j) + (1-U)h_3(j) + \epsilon_j & \text{类 2} \\
X_j &=& Uh_2(j) + (1-U)h_3(j) + \epsilon_j & \text{类 3}
\end{array}
\tag{12.62}
$$

其中，$j = 1,2,\cdots, 21$，U 在(0,1)上是均匀的，ϵ_j 是标准规范变量，h_ℓ 是移位的三角波形：$h_1(j) = \max(6 - |j-11|, 0)$，$h_2(j) = h_1(j-4)$且 $h_3(j) = h_1(j+4)$。图 12.12 显示了来自每个类的一些波形示例。

表 12.4 显示了应用于波形数据的 MDA 的结果，以及本章和其他章节中的一些其他方法的结果。每个训练样本有 300 个观测。而且使用相等的先验，所以在各个类中大致有 100 个观测。我们使用了大小为 500 的检验样本。两种 MDA 模型在相应的标题中描述。

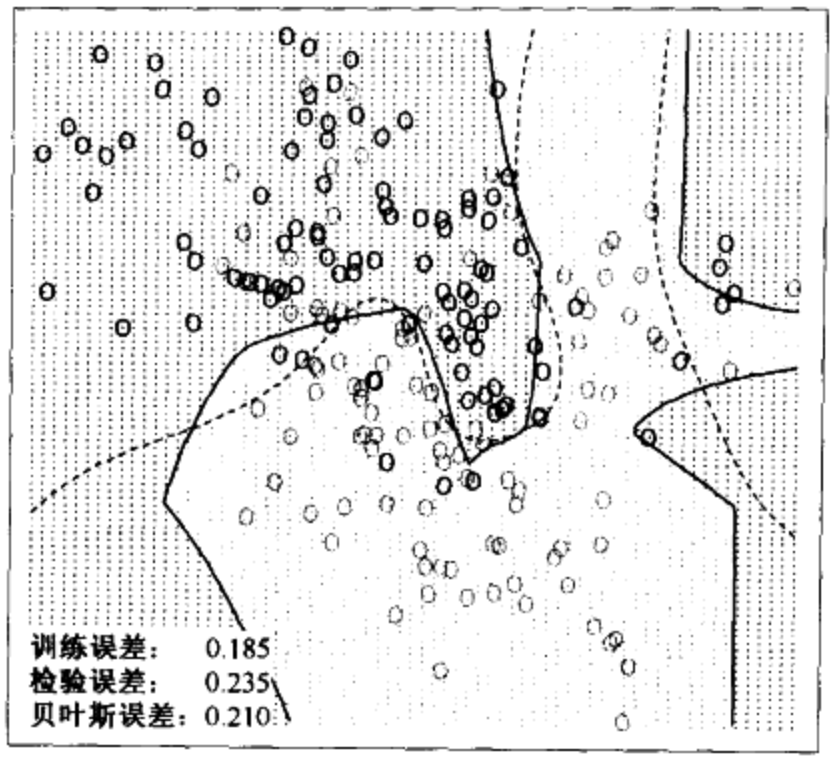


图 12.11 混合数据上的 FDA 和 MDA。上图使用 FDA，以 MARS 作为回归过程。下图使用 MDA，每个类有5个混合中心（已指出）。MDA的解接近于贝叶斯最优解，它是由给定高斯混合数据可能期望的结果。背景上的紫色虚线是贝叶斯判定边界（见彩页）

表 12.4 波形数据的结果。值是对 10 个模拟数据求的平均值，圆括号中是其标准平均误差。直线上面的五行取自Hastie等人(1994)的著作。直线下面的第一个模型是MDA，每个类拥有三个子类。除了通过对有效 的4 df做一个粗糙性罚来补偿判断式系数之外，下一行是相同的。第三行是对应的罚LDA或PDA模型

技术	误差率	
	训练	检验
LDA	0.121(0.006)	0.191(0.006)

（续表）

技术	误差率	
	训练	检验
QDA	0.039(0.004)	0.205(0.006)
CART	0.072(0.003)	0.289(0.004)
FDA/MARS(1 次)	0.100(0.006)	0.191(0.006)
FDA/MARS(2 次)	0.068(0.004)	0.215(0.002)
MDA(3 个子类)	0.087(0.005)	0.169(0.006)
MDA(3 个子类，罚 4 df)	0.137(0.006)	0.157(0.005)
PDA(罚 4 df)	0.150(0.005)	0.171(0.005)
贝叶斯		0.140

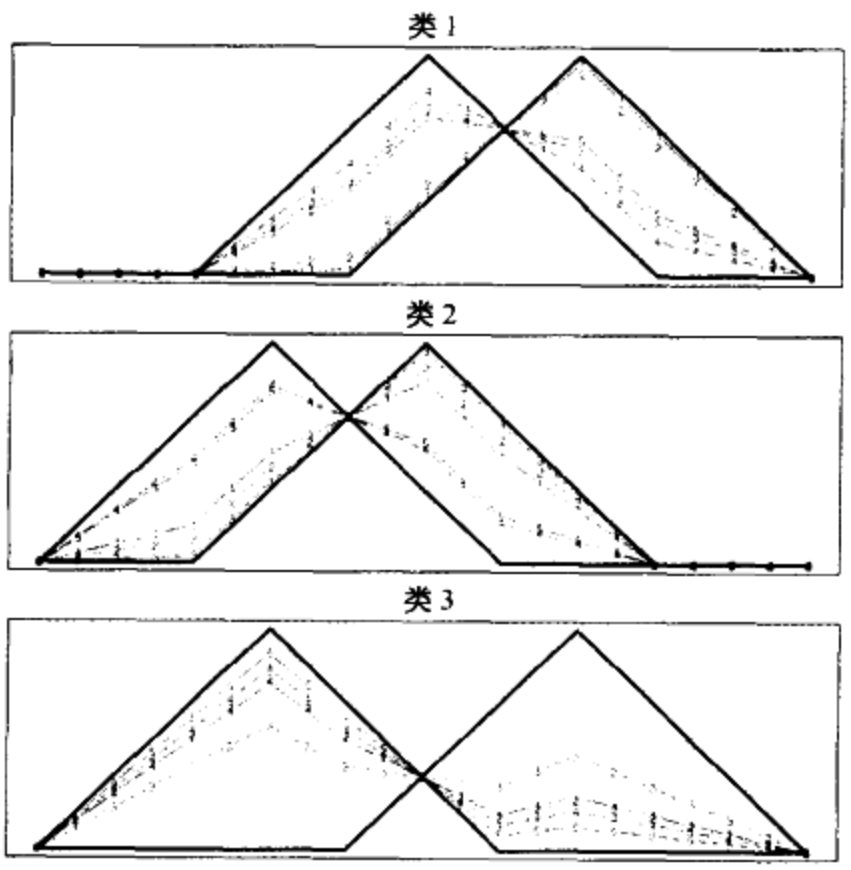


图 12.12　在高斯噪声加入前，一些由模型(12.62)产生的波形例子(见彩页)

图 12.13 显示了罚 MDA 模型的主要标准变量，在检验数据上求值。正像我们可能已经猜到的那样，类看上去位于三角形的边缘上。这是因为 $h_j(i)$是用 21 维空间中的三个点表示的，因此形成三角形的顶点，并且每个类都表示为一对顶点的凸组合，因此处于边缘上。所有的信息在前两个维上也是清晰可见的；前两个坐标导致的方差百分率是 99.8%，而且在那里即使截断解也不会失去什么。估计该问题的贝叶斯风险大约为 0.14(Breiman 等人，1984)。MDA 接近于最优风险率；这不奇怪，因为 MDA 模型的结构类似于生成的模型。

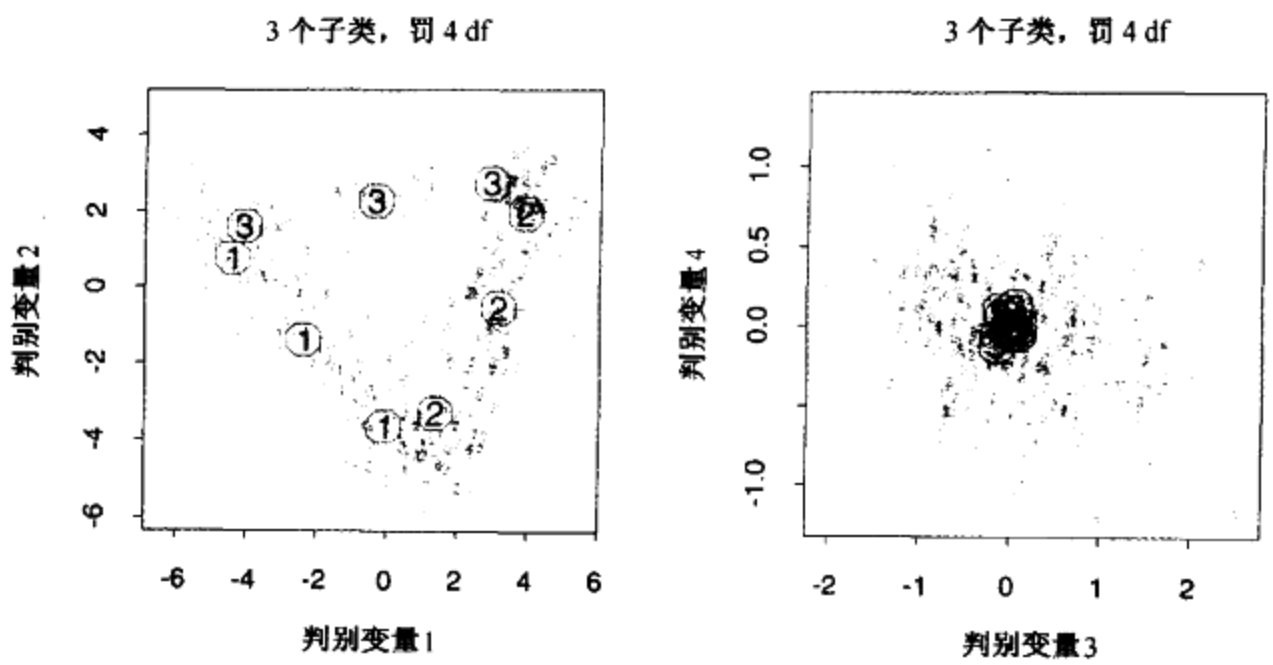


图 12.13 拟合波形模型样本的 MDA 模型的二维视图。点是独立的检验数据，投影到两个主标准坐标上(左图)，以及第三个和第四个坐标上(右图)。图中指出了子类的中心(见彩页)

12.8 计算考虑

对于 N 个训练实例，p 个预测子和 m 个支持向量，假设 $m \approx N$，支持向量机需要 $m^3 + mN + mpN$ 次操作。尽管有计算上的捷径可走，但它们对 N 不能很好地伸缩(Platt, 1999)。由于这方面的工作发展迅速，读者必须通过 Web 去查找最新的技术。

和 PDA 一样，LDA 需要 $Np^2 + p^3$ 次操作。FDA 的复杂性依赖于使用的回归方法。许多技术在 N 上是线性的，如加法模型和 MARS。一般的样条和基于核的回归模型，通常都需要 N^3 次操作。

用 S-PLUS 编写的拟合 FDA、PDA 和 MDA 模型的软件公开发布在网站 http://lib/stat.cmu/edu/S/mda 上。

文献注释

支持向量机的理论背景源于 Vapnik，并在 Vapnik(1996)中给予了介绍。有大量关于 SVM 的文献；一种在线文献由 Alex Smola 和 Bernhard Schölkopf 建立并维护，在以下网站可以找到：

http://www.kernel-machines.org/publications.html

我们的处理是基于 Wahba 等人(2000)和 Evgeniou 等人(2001)的著作，而指南由 Burges (Burges, 1998)给出。

线性判别分析源于 Fisher(1936)和 Rao(1973)。与最优评分的联系至少应追溯到 Breiman 和 Ihaka(1984)，简单的形式应追溯到 Fisher(1936)。线性判别分析与对应分析有很强的联系(Greenacre, 1984)。灵活的、罚的和混合判别分析取自 Hastie 等人(1994)的著作。Hastie 等人(1995)的著作及 Hastie 和 Tibshirani(1996b)，所有三种方法的总结在 Hastie 等人(1998)的著作中可以找到，也可参考 Ripley(1996)。

习题

12.1 证明准则(12.25)和式(12.8)是等价的。

12.2 证明式(12.25)与式(12.29)的解相同。

12.3 考虑对式(12.41)的修改,其中不惩罚常量。形式化该问题,并说明解的特点。

12.4 假设你进行 K 群问题的约化 - 子空间的线性判别分析。计算由 $z = \mathbf{U}^T x$ 给定的维 $L \leqslant K - 1$ 的标准变量,其中 U 是判别式系数的 $p \times L$ 矩阵,且 $p > K$ 是 x 的维。

(a) 如果 $L = K - 1$,证明:

$$\|z - \bar{z}_k\|^2 - \|z - \bar{z}_{k'}\|^2 = \|x - \bar{x}_k\|_W^2 - \|x - \bar{x}_{k'}\|_W^2$$

其中 $\|\cdot\|_W$ 表示关于协方差 $\mathbf{W}$ 的 Mahalanobis 距离。

(b) 如果 $L < K - 1$,证明对于投影到 $\mathbf{U}$ 生成的子空间的分布,左边相同的表达式可以度量 Mahalanobis 平方距离中的差。

12.5 在 phoneme.subset 中的数据从本书的站点可以找到:

http://www-stat.stanford.edu/ElemStatLearn

该数据包括 60 名讲话者发出的音素的数字化对数 - 周期图,每个讲话者为 5 个类中的每个类产生音素。作为频率 0 - 255 的函数,绘图表示 256 个"特征"的每个向量是合适的。

(a) 对每个类,作为频率的函数,分别产生所有音素曲线图。

(b) 你打算使用最近邻原型分类方案将曲线分类为音素类。特别地,你将在每个类中使用 K-均值聚类算法(S-PLUS 的 kmeans()),然后,把观测分类到离簇中心最近的簇中。曲线是高维的,且有一个相当小的样本大小与变量的比率。你决定将全部原型限制成频率的光滑函数。特别地,你决定把每个原型 m 表示成 $m = B\theta$,其中 B 是一个 $256 \times J$ 的自然样条基函数矩阵,J 个纽结在(0,255)中均匀选择,而边界纽结在 0 和 255 上。说明处理如何解析地进行,特别是如何避免高代价的高维拟合过程。(提示:将 B 限制成正交的会有帮助。)

(c) 在音素数据上实现你的过程,并用它进行试验。把数据分成一个训练集和一个检验集(50 - 50),确保说话者不会跨越不同的集合(为什么?)。使用每个类有 $K = 1, 3, 5, 7$ 个中心,且对其每一个使用 $J = 5, 10, 15$ 个纽结(注意,对 J 的每个值,以相同的初值开始 K-均值过程),并比较这些结果。

12.6 假设在 FDA(见第 12.5.1 节)中使用的回归过程是基函数 $h_m(x)(m = 1, \cdots, M)$ 的线性展开式。令 $\mathbf{D}_\pi = \mathbf{Y}^T\mathbf{Y}/N$ 是类比例对角矩阵。

(a) 证明最优评分问题(12.50)可以用向量记法写成:

$$\min_{\theta,\beta} \|\mathbf{Y}\theta - \mathbf{H}\beta\|^2 \tag{12.63}$$

其中,θ 是 K 个实数的一个向量,而 $\mathbf{H}$ 是对 $h_j(x_i)$ 求值的 $N \times M$ 矩阵。

(b) 假设 θ 上的规范化是 $\theta^T\mathbf{D}_\pi 1 = 0$ 且 $\theta^T\mathbf{D}_\pi\theta = 1$。根据初始评分 $\theta(g_i)$ 来解释这些规范化。

(c) 利用这种规范化，证明式(12.63)关于 β 可以部分地优化，并导致

$$\max_{\theta} \theta^T \mathbf{S} \theta \tag{12.64}$$

满足规范化限制，其中 $\mathbf{S}$ 是对应于基矩阵 H 的投影操作子。

(d) 假设 h_j 包括常量函数，证明 $\mathbf{S}$ 的极大本征值是 1。

(e) 令 Θ 是一个 $K \times K$ 评分矩阵(按列)，并假设规范化是 $\Theta^T \mathbf{D}_\pi \Theta = \mathbf{I}$。证明式(12.51)的解是由 $\mathbf{S}$ 的本征向量全集给出的；第一个本征向量是平凡的，并为评分的中心进行定位。而其余的用于刻画最优评分解的特征。

12.7 导出罚最佳评分问题(12.55)的解。

12.8 证明最佳评分求出的系数 β_ℓ 与线性判别分析求出的判别方向 ν_ℓ 成比例。

12.9 设 $\hat{\mathbf{Y}} = \mathbf{X}\hat{\mathbf{B}}$ 是在 $N \times p$ 矩阵 $\mathbf{X}$ 上线性回归之后被拟合的 $N \times K$ 指示子响应矩阵，其中 $p > K$。考虑约化特征 $x_i^* = \hat{\mathbf{B}}^T x_i$。证明使用 x_i^* 的 LDA 与原空间中的 LDA 等价。

12.10 核与线性判别分析。假设你希望使用输入变量 $h(x)$ 的变换向量来实现线性判别分析(两个类)。由于 $h(x)$ 是高维的，你将使用一个正则化的类内协方差矩阵 $\mathbf{W}_h + \gamma\mathbf{I}$。证明可以仅使用内积 $K(x_i, x_{i'}) = \langle h(x_i), h(x_{i'}) \rangle$ 就可以评估模型。因此，支持向量机的核性质也可以被正则化的线性判别分析所共享。

12.11 MDA 过程将每个类建模成为混合高斯。因此，每个混合中心属于且仅属于一个类。一个更一般的模型允许每个混合中心被所有类共享。我们将标号和特征的联合密度取做

$$P(G, X) = \sum_{r=1}^{R} \pi_r P_r(G, X) \tag{12.65}$$

混合联合密度。进一步，假设：

$$P_r(G, X) = P_r(G)\phi(X; \mu_r, \Sigma) \tag{12.66}$$

这个模型由中心在 μ_r 的区域组成，且对于每个区域有一个类轮廓 $P_r(G)$。后验类分布由下式给出：

$$P(G = k | X = x) = \frac{\sum_{r=1}^{R} \pi_r P_r(G = k)\phi(x; \mu_r, \Sigma)}{\sum_{r=1}^{R} \pi_r \phi(x; \mu_r, \Sigma)} \tag{12.67}$$

其中，分母是边缘分布 $P(X)$。

(a) 说明这个模型(称为 MDA2)可以看做是 MDA 的一个泛化，因为：

$$P(X | G = k) = \frac{\sum_{r=1}^{R} \pi_r P_r(G = k)\phi(x; \mu_r, \Sigma)}{\sum_{r=1}^{R} \pi_r P_r(G = k)} \tag{12.68}$$

其中，$\pi_{rk} = \pi_r P_r(G = k) / \sum_{r=1}^{R} \pi_r P_r(G = k)$ 是对应于第 k 个类的混合比例。

(b) 为 MDA2 导出 EM 算法。

(c) 说明如果初始权值矩阵像 MDA 那样构造，且涉及每个类的 k – 均值聚类，则 MDA2 算法与原 MDA 过程是一样的。

第 13 章　原型方法和最近邻

13.1　引言

本章，我们将讨论一些简单而基本的无模型分类和模式识别方法。由于它们是高度非结构化的，所以对于理解特征和类结果之间联系的本质一般不大有用。然而，作为黑盒预测引擎，它们可能是非常有效的，而且对于实际的数据问题，它们通常是性能最好的方法之一。最近邻技术也可以用于回归；这在第 2 章中已经提及，并对低维问题相当有效。然而，对于高维特征，偏倚－方差权衡对于最近邻回归却没有像对分类那样顺利。

13.2　原型方法

贯穿本章，我们的训练数据由 N 个数对 $(x_1, g_1), \cdots, (x_N, g_N)$ 组成，其中，g_i 是在 $\{1, 2, \ldots, K\}$ 中取值的类标号。原型方法用特征空间中的点集表示训练数据。除了后面要讨论的 1-最近邻分类外，通常这些原型都不是训练样本中的例子。

每个原型都有一个相关联的类标号，查询点 x 被分类到最近原型的类。将每个特征标准化，使之在训练样本上具有均值 0 和方差 1，之后“最近”通常用特征空间中的欧几里德距离来定义。欧几里德距离适合于定量的特征。我们将在第 14 章讨论定性特征和其他类型特征之间的距离度量。

如果原型被恰当定位以捕获每个类的分布，那么这些方法可能是非常有效的。不规则的类边界可以用特征空间中正确位置上足够的原型来表示。我们面临的主要挑战是要解决使用多少个原型以及把它们放在哪里。根据原型选择的数量和方式，这些方法各有不同。

13.2.1　K-均值聚类

K-均值聚类是一种在无类标号数据中发现簇和簇中心的方法。选择期望的簇中心数，比如 R，K-均值过程反复移动簇中心以极小化整个簇内方差[①]。给定一个初始中心集，K-均值算法交替执行如下两步：

- 对每个中心，我们识别离该中心比离任何其他中心都近的训练点的子集（它的簇）；
- 计算每个簇中数据点的每个特征的均值，并且该均值向量成为该簇新的中心。

重复这两步，直到收敛。典型的初始中心是从训练数据中随机选择的 R 个观测。K-均值过程的细节，以及允许不同变量类型和更一般的距离度量的推广将在第 14 章给出。

使用 K-均值聚类对有标号数据进行分类，其步骤是：

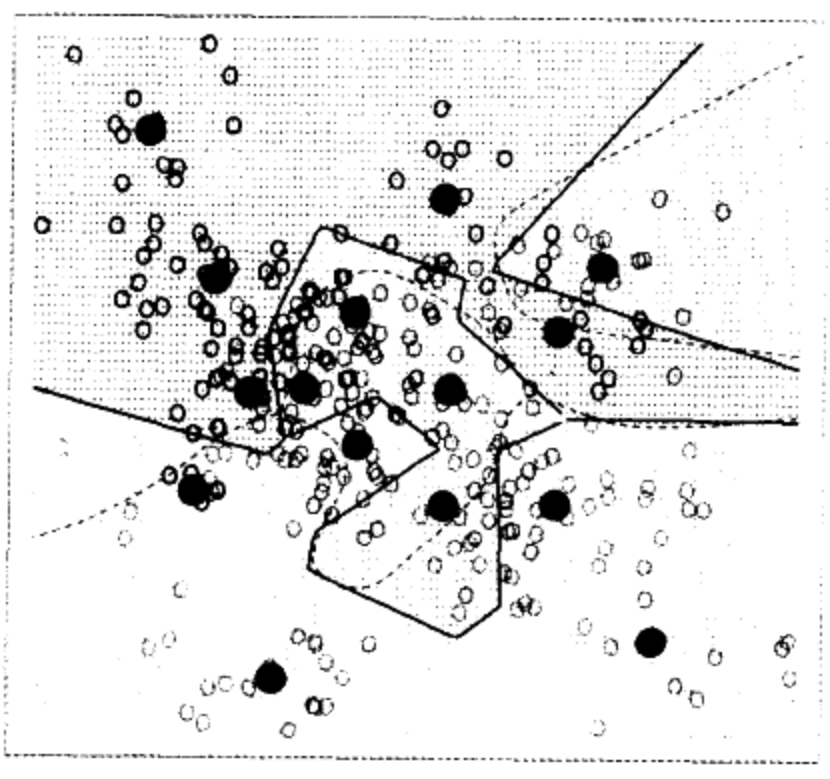
- 对每个类中的训练数据分别使用 K-均值聚类，每个类使用 R 个原型。

---

① K-均值中的“K”是指簇中心的个数。由于我们已经用 K 专指示类的个数，所以用 R 表示簇的个数。

- 对 $K \times R$ 个原型中的每个原型赋一个类标号。
- 将新特征 x 分到最近原型的类。

图 13.1(上图)显示了具有三个类和两个特征的模拟例子。对每个类五们使用 $R = 5$ 个原型,并显示了分类区域和判定边界。注意,一些原型靠近类边界,导致靠近这些边界的点可能误分类。这源于这种方法的一个明显的缺点:对每个类,其他类关于该类的原型定位没有发言权。下面讨论一种较好的方法,它使用所有数据对全部原型定位。



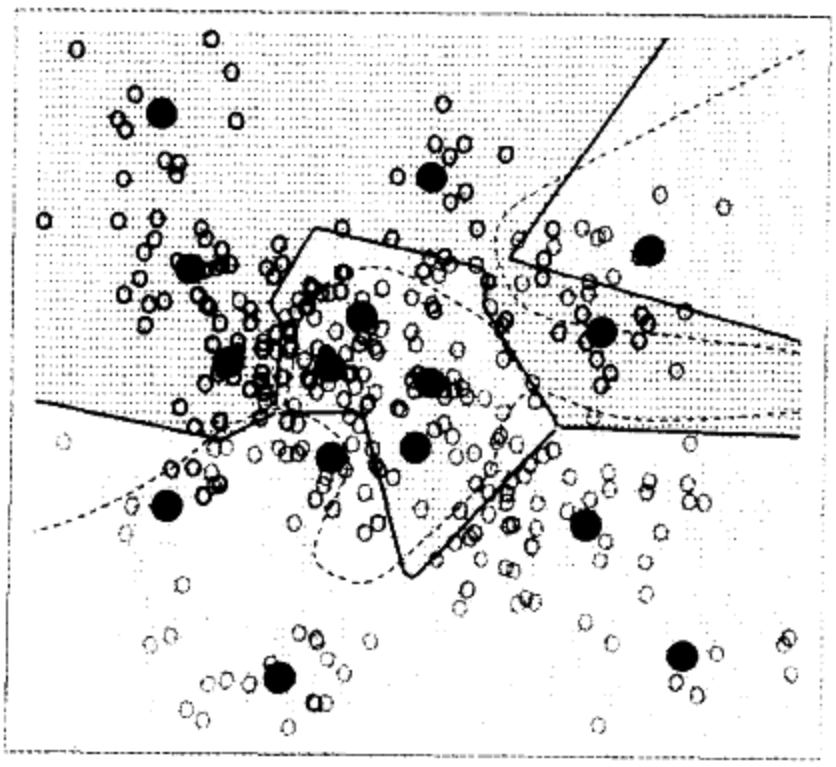


图 13.1 模拟例子,具有三个类,每个类 5 个原型。每个类中的数据由一个高斯混合产生。上图,通过在每个类中分别使用 K-均值聚类算法找出原型。下图,LVQ算法(从一个 K-均值解开始)将原型从判定边界移开。背景上的紫色虚线是贝叶斯判定边界(见彩页)

13.2.2　学习向量量化

在这种出自 Kohonen(1989)的技术中，关于判定边界，原型被以一种特别方式策略地安置。学习向量量化(Learning Vector Quantization，LVQ)是一种在线算法——观测被逐个处理。

其思想是训练点吸引正确类的原型，而排斥其他原型。当迭代结束时，原型将靠近它们所在类的训练点。按照随机逼近学习率准则(见第 11.4 节)，学习率 ϵ 随每次迭代而降至 0。

图 13.1(下图)显示了 LVQ 的结果，它使用 K-均值解作为开始值。原型已趋于从判定边界移开，且远离竞争类的原型。

刚介绍过的过程实际上叫 LVQ1。其改进(LVQ2，LVQ3 等)已被提出来，它们在某些情况下可以提高性能。学习向量量化方法的不足在于它们是由算法而不是由某些固定准则的优化标准定义的。这使得理解它们的特征比较困难。

算法 13.1　学习向量量化

1. 为每个类选择 R 个初始原型 $m_1(k), m_2(k), \cdots, m_R(k), k = 1, 2, \cdots, K$，例如，从每个类中随机抽取 R 个训练点

2. 随机抽取一个训练点 x_i(有放回)，令 (j, k) 指示是离 x_i 最近的原型 $m_j(k)$

(a) 如果 $g_i = k$(即它们在同一个类中)，将原型移向训练点：

$$m_j(k) \leftarrow m_j(k) + \epsilon(x_i - m_j(k))$$

其中 ϵ 是学习率。

(b) 如果 $g_i \neq k$(即它们在不同的类中)，将原型从训练点移开：

$$m_j(k) \leftarrow m_j(k) - \epsilon(x_i - m_j(k))$$

3. 重复步骤 2，随迭代次数的增加学习降低率 ϵ 直到 0。

13.2.3　高斯混合

高斯混合模型也可以认为是原型方法，其思想与 K-均值和 LVQ 相似。我们在第 6.8 节、第 8.5 节和第 12.7 节略微详细地讨论过高斯混合。每个簇用高斯密度描述，它具有一个中心(和 K-均值一样)和一个协方差矩阵。如果我们限制分量高斯有一个标量协方差矩阵，则比较就变得较为明显(见习题 13.1)。交替的 EM 算法的两步与 K-均值的两步非常相像：

- E 步，基于每个观测对应的高斯似然，对每个簇，赋予每个观测一个响应度或权。靠近一个簇中心的观测对于该簇最可能获得权值 1，而对于其他簇则获得权值 0。界于两个簇之间的观测将它们的权相应地分开。
- 在 M 步，每个观测用于调整每个聚类的加权均值(和协方差)。

结果，高斯混合模型通常作为一个软聚类方法，而 K-均值是硬聚类方法。

类似地，当高斯混合模型用于表示每个类中的特征密度时，为了分类 x，它产生光滑的后验概率 $\hat{p}(x) = \{\hat{p}_1(x), \cdots, \hat{p}_K(x)\}$[见式(12.58)]。通常将它解释成软分类，而实际的分类规则是 $\hat{G}(x) = \arg\max_k \hat{p}_k(x)$。图 13.2 就第 2 章的模拟混合问题，对 K-均值和高斯混合的结果进行了对比。我们看到，尽管判定边界大致相似，但对于混合模型来说判定边界还是光滑些(尽管原型在近似相同的位置上)。我们也看到尽管两种过程在西北角的一个区域(不正确地)产生了一个绿色原型，但高斯混合分类器可能最后会忽略这个区域，而 K-均值不会。关于这个例子，LVQ 给出与 K-均值非常相似的结果，图中没有显示。

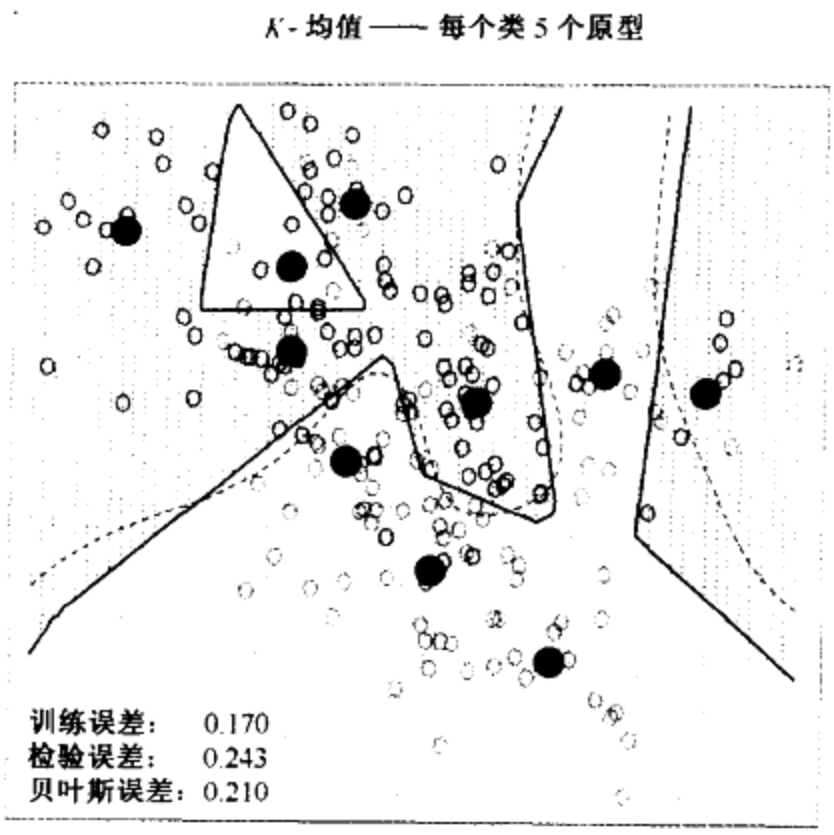


图 13.2 上图显示对混合数据例子应用的 K-均值分类器。判定边界是分段线性的。下图显示高斯混合模型，所有分量高斯具有公共协方差。混合模型的EM算法开始于一个 K-均值解。背景上的紫色虚线是贝叶斯判定边界（见彩页）

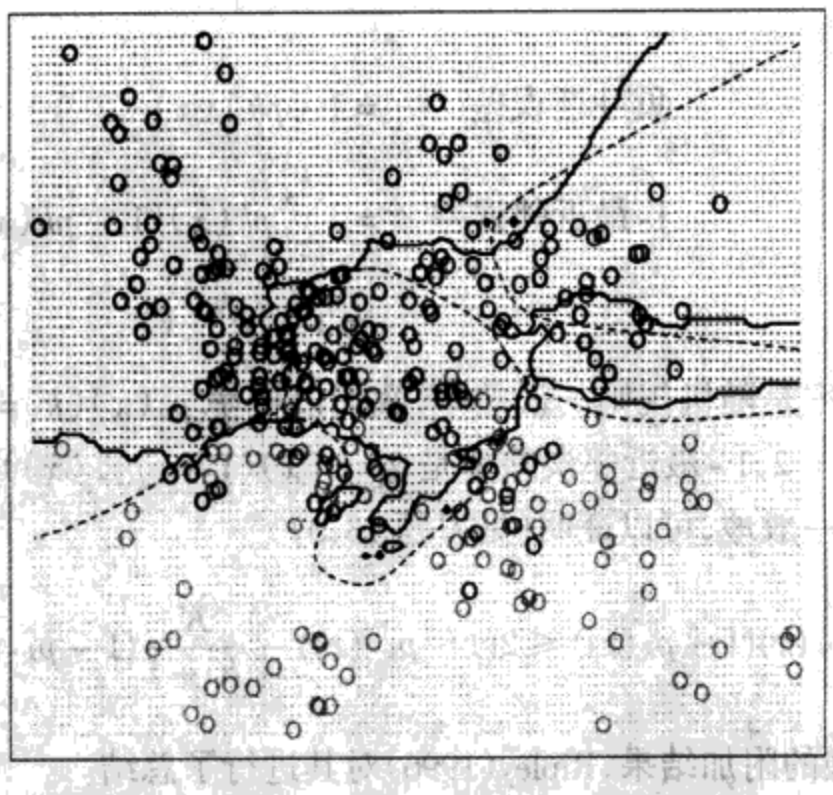
13.3 k-最近邻分类器

这些分类器是基于内存的，不需要拟合模型。给定查询点 x_0，寻找在距离上与 x_0 最近的 k 个训练点 $x_{(r)}$，$r = 1, \cdots, k$，然后在 k 个近邻中使用多数表决分类。平局被随机地打破。为了简单起见，我们假设特征是实数值的，并且在特征空间中使用欧氏距离：

$$d_{(i)} = \|x_{(i)} - x_0\| \tag{13.1}$$

通常，首先对每个特征标准化，使之均值为 0、方差为 1，因为它们可能以不同的单位度量。在第 14 章，我们将讨论适合于定性和序数特征的距离度量，以及对于混合数据怎样合并它们。自适应地选择距离度量将在本章后面讨论。

尽管简单，k -最近邻方法仍在大量分类问题中是成功的，包括手写数字、卫星图像和 EKG 模式等。当每个类有许多可能的原型且判定边界非常不规则时，它通常是成功的。图 13.3（上图）显示了 15 -最近邻分类器应用于 3-类模拟数据的判定边界。与下图比较，判定边界相当光滑，在下图使用的是 1 -最近邻分类器。在最近邻和原型方法之间有着密切的联系：在 1 -最近邻分类中，每个训练点都是一个原型。



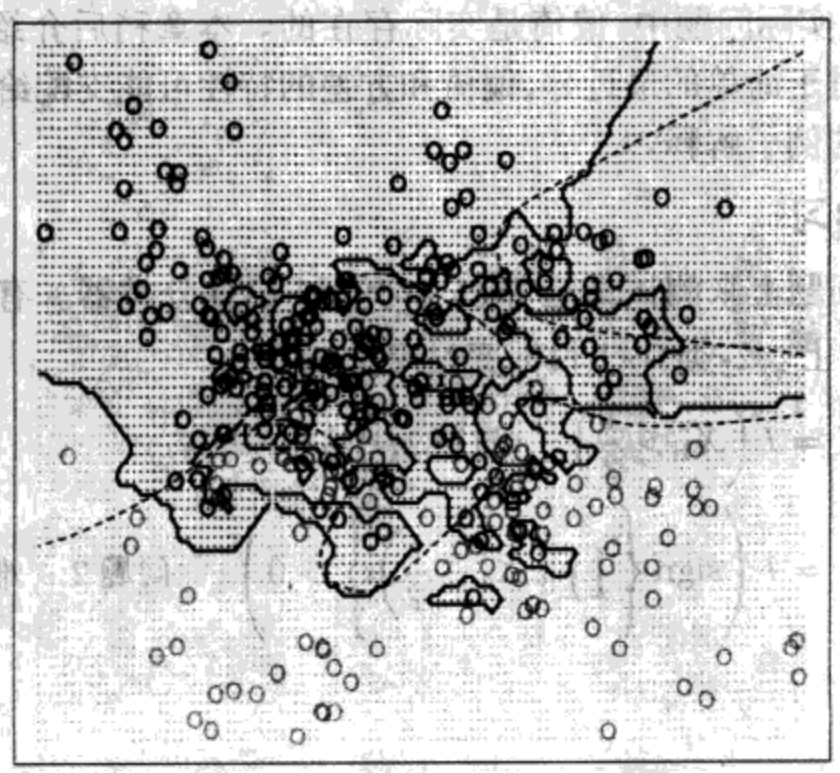


图 13.3　k -最近邻分类器应用于图 13.1 的模拟数据。背景上的紫色虚线是贝叶斯判定边界（见彩页）

图 13.4 显示了 2-类混合问题的训练、检验和 10 折交叉验证误差，作为近邻个数的函数。由于 10 折 CV 误差是 10 个数的平均，所以我们可以估计一个标准误差。

由于它们仅使用了离查询点最近的训练点，所以 1 -最近邻估计的偏倚通常很低，但方差很高。Cover 和 Hart(1967)的一个著名结果表明：近似地 1 -最近邻分类器的误差永远不会高于贝叶斯误差率的二倍。证明的大致思想如下(使用平方误差损失)：假设查询点与训练点之一一致，则偏倚为 0。如果特征空间的维数是固定的，且训练数据以密集的方式充满空间，这便渐近地为真。那么，贝叶斯规则的误差恰恰是伯努利随机变量的方差(在查询点的目标)，而 1 -最近邻规则的误差是伯努利随机变量方差的二倍，为训练和查询目标各贡献一份。

现在我们对于误分类损失给予更详细的说明。在 x 处令 k^* 是支配类，且 $p_k(x)$是类 k 的真条件概率。那么

$$\text{贝叶斯误差} \quad = 1 - pk^*(x) \tag{13.2}$$

$$\text{1-最近邻误差} \quad = \sum_{k=1}^{K} pk(x)(1 - pk(x)) \tag{13.3}$$

$$\geqslant 1 - pk^*(x) \tag{13.4}$$

渐近的 1 -最近邻误差率是随机规则的误差率，我们以概率 $p_k(x)(k = 1, \cdots, K)$随机挑选分类和检验点。对于 $K = 2$，1 -最近邻误差率是 $2p_{k*}(x)(1 - p_{k*}(x)) \leqslant 2(1 - p_{k^*}(x))$(二倍于贝叶斯误差率)。更一般地，可以证明(见习题 13.3)

$$\sum_{k=1}^{K} p_k(x)(1 - p_k(x)) \leqslant 2(1 - p_{k*}(x)) - \frac{K}{K-1}(1 - p_{k*}(x))^2 \tag{13.5}$$

已经导出许多这种类型的附加结果；Ripley(1996)对其进行了总结。

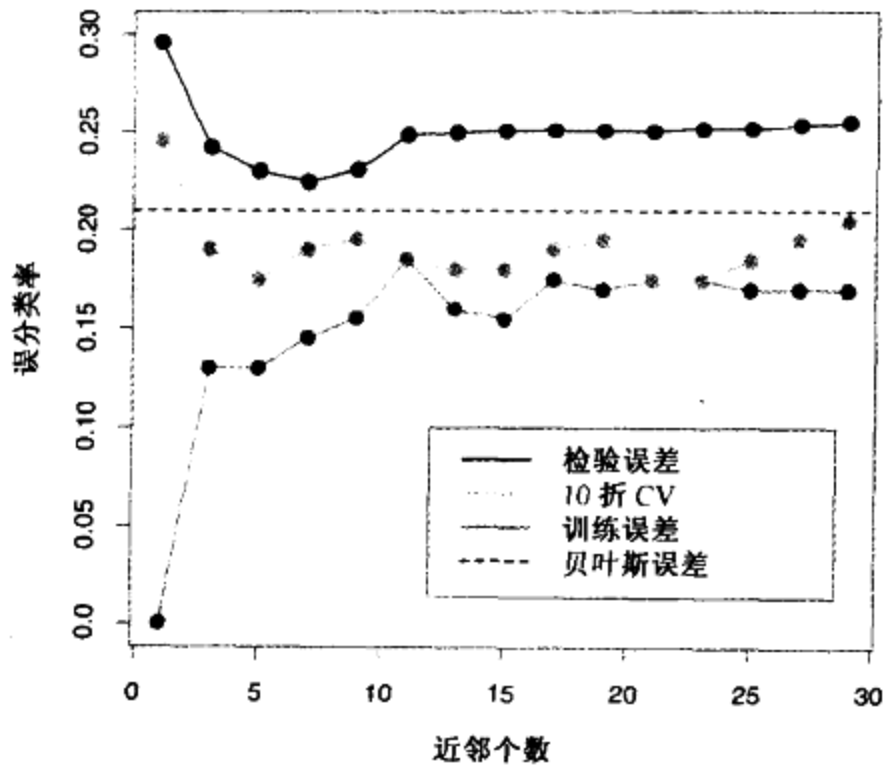
该结果可以粗略地表明在一个给定问题中可能达到的最好性能。例如，如果 1 -最近邻规则具有 10% 的误差率，则渐近贝叶斯误差率至少是 5% 。这里，难点是渐近部分，它假设最近邻规则的偏倚是 0。在实际问题中，偏倚是实际存在的。本章稍后介绍的自适应最近邻规则试图缓解这种偏倚。对于简单的最近邻，偏倚和方差的特性可能支配给定问题的最佳近邻数。这一点可以通过下面的例子解释。

13.3.1 例：比较学习

我们在两个模拟问题上来测试最近邻、K-均值和 LVQ 分类器。有 10 个独立特征 X_j，每个均匀地分布在[0, 1]上。2-类 0 -1 目标变量定义如下：

$$\begin{aligned} &Y = I\left(X_1 > \frac{1}{2}\right); \quad \text{问题 1: “易”}, \\ &Y = I\left(\text{sign}\left\{\prod_{j=1}^{3}\left(X_j - \frac{1}{2}\right)\right\} > 0\right): \quad \text{问题 2: “难”} \end{aligned} \tag{13.6}$$

因此，在第一个问题中，两个类被超平面 $X_1 = 1/2$ 所分离；在第二个问题中，两个类在由前三个特征定义的超立方体中形成了棋盘模式。在这两个问题中，贝叶斯误差率都是 0。这里有 100 个训练观测和 1000 个检验观测。



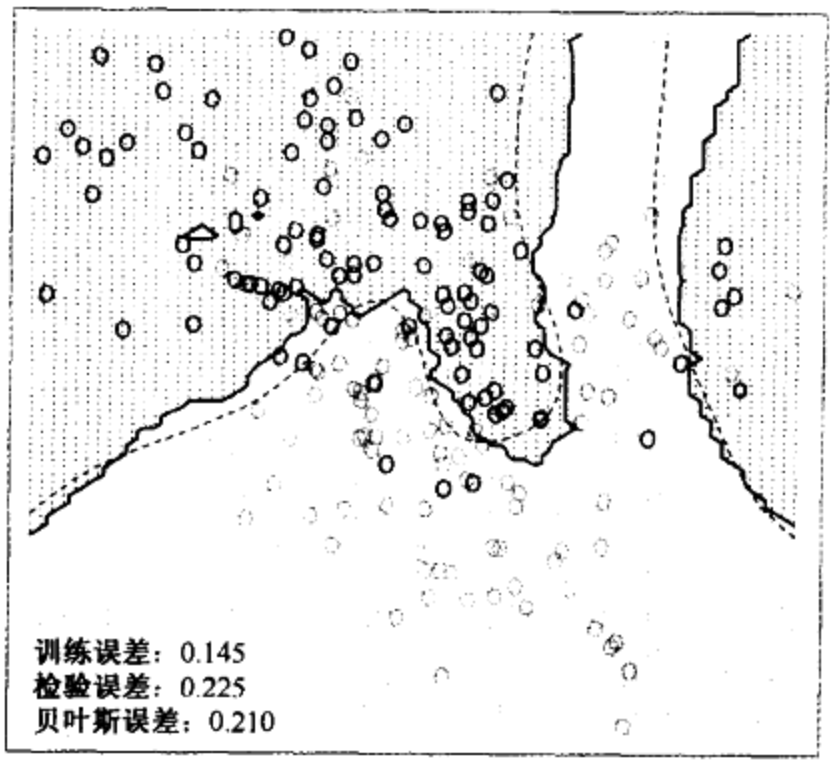


图 13.4　2 - 类混合数据上的 k - 最近邻。上图显示误分类率，作为邻域大小的函数。下图显示 7 - 最近邻的判定边界，关于极小化检验误差看上去它是最优的。背景中的紫色虚线是贝叶斯判定边界（见彩页）

图 13.5 显示了随调整参数的变化，最近邻、K - 均值和 LVQ 在 10 次实现中的误分类的均值和标准误差。我们看到，K - 均值和 LVQ 产生几乎相同的结果。关于它们的调整参数的最佳选择，K - 均值和 LVQ 在第一个问题中做得比最近邻好，而对于第二个问题，它们很相似。注意，每个调整参数的最佳值明显依赖于问题。例如，在第一个问题中，25 - 最近邻的误分类率是 1 - 最近邻的 70%；而在第二个问题中，1 - 最近邻的误分类率是 25 - 最近邻的 18%。这些

结果表明使用客观的、基于数据的方法(如,交叉验证)估计调整参数最佳值的重要性(见图 13.4和第 7 章)。

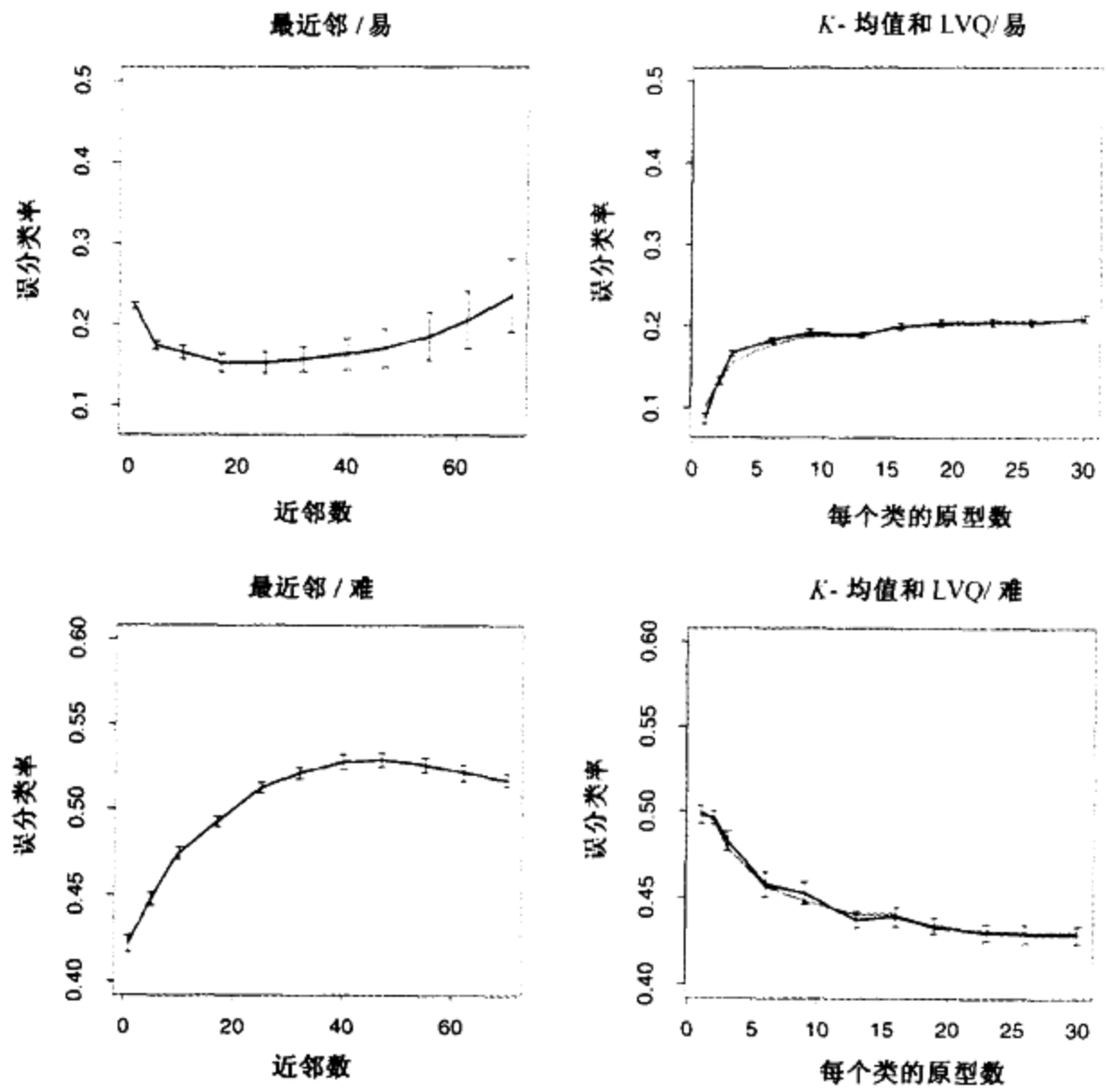


图 13.5 关于文中描述的两个模拟问题:"易"和"难"问题,10 次实现上的最近邻、K-均值(蓝色)和LVQ(红色)的误分类均值 ± 一个标准误差(见彩页)

13.3.2 例:k-最近邻和图像场景分类

STATLOG 项目(Michie 等人,1994)使用部分 LANDSAT 图像作为分类(82 × 100 像素)的性能标准。图 13.6 显示了澳大利亚某农业区域的 4 幅热度图图像,两幅以可见光谱成像,两幅以红外光谱成像。每个像素有一个类标号,取自 7 个元素的集合$\mathcal{G}$ = {红土壤,棉状,植被茬,混合,灰土壤,潮湿灰土壤,极度潮湿灰土壤},由勘察该区域的研究助手人工确定。中下图显示了实际的土地使用情况,用不同颜色指出这些类。目标是基于 4 个谱带中的信息,在像素上对土地使用情况进行分类。

5 -最近邻产生一个预测地图(右下图),并按以下方法计算。对于每个像素,我们抽取 8 -近邻特征映射——像素本身和它的 8 个直接近邻(见图 13.7)。在 4 个谱带上分别做,每个像素产生$(1+8)\times 4 = 36$ 个输入特征。然后,在这个 36 维空间上进行 5 -最近邻分类。结果检验误差率大约是 9.5%(见图 13.8)。在 STATLOG 项目中使用的全部方法中,包括 *LVQ*、*CART*、神经网络、线形判定分析和许多其他方法,k-最近邻在该任务上的性能最好。因此,IR[36]中的判

定边界很可能是相当不规则的。

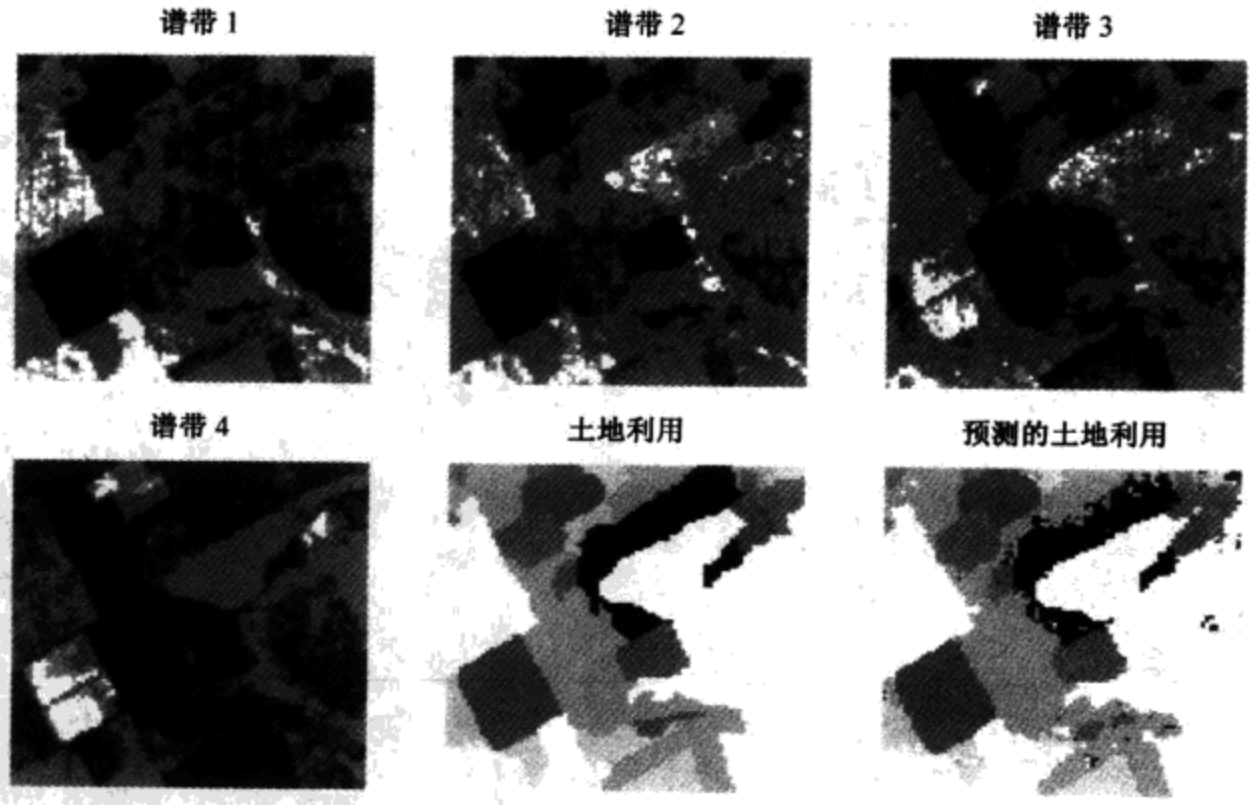


图 13.6　前 4 幅图是某农业区域 4 个谱带下的 LANDSAT 图像，用热度图描绘。其余两幅图给出实际的土地使用情况（彩色编码）和使用文中描述的5-最近邻规则预测的土地使用情况（见彩页）

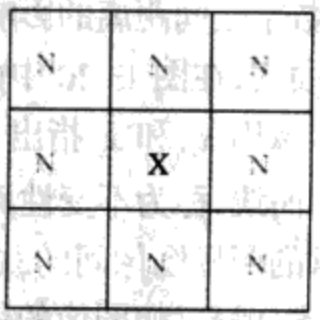


图 13.7　一个像素和它的 8 - 近邻特征映射

13.3.3　不变度量与切距离

对于某些问题，在一定的自然变换下，训练特征是不变的。最近邻分类器能够利用这些不变性，它把这样的不变性合并到用于度量两个对象之间距离的估价中。这里给出一个例子，它成功地使用了这种思想，并且结果分类器优于与它同时开发的其他方法（Simard 等人，1993）。

问题仍然是第 1 章和第 11.7 节中讨论过的手写数字识别问题。输入是 $16 \times 16 = 256$ 像素的灰度图像，一些例子显示在图 13.9 中。在图 13.10 的顶部显示了“3”，以它的实际方向（中间）和向左、右各旋转 7.5°和 15°显示。这种旋转常发生在实际书写中，并且在我们看来轻微旋转后的“3”显然仍然是“3”。因此，我们希望最近邻分类器认为这两个“3”非常接近（相似）。但是，旋转的“3”的 256 灰度像素值看起来与原图像的灰度像素值非常不同，因此，这两个对象在 IR^{256} 中的欧几里德距离可能相差很多。

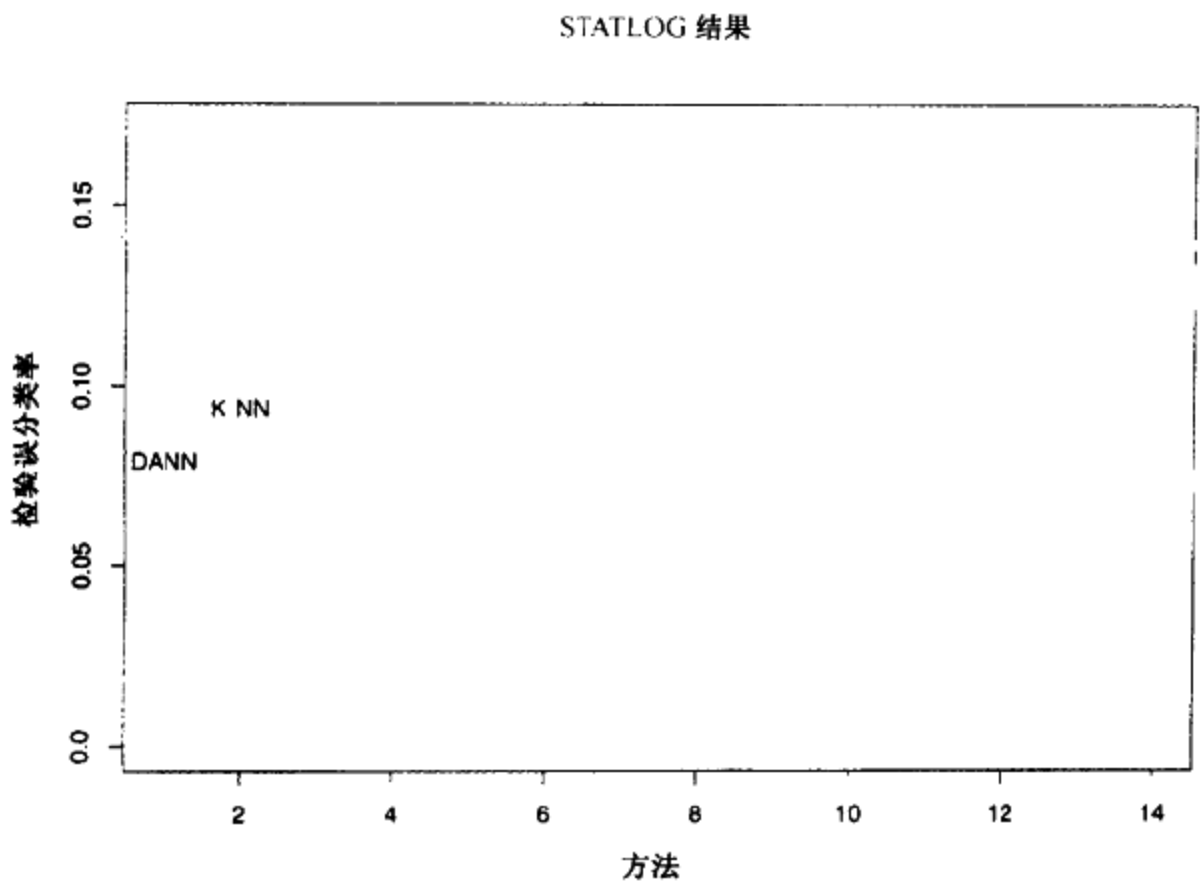


图 13.8 由 STATLOG 项目报告的一些分类器的检验误差性能。其中 DANN 是k-最近邻的一种变型，使用一种自适应度量(见第13.4.2节)

我们希望消除旋转对度量两个同类数字之间距离的影响。考虑包括原始的"3"及其旋转形式的像素值的集合。这是一个 $\mathbb{R}^{256}$ 中的 1 维曲线，在图 13.10 中由经过"3"的绿色曲线描绘。图 13.11显示了 $\mathbb{R}^{256}$ 中的一种风格化的形式，两幅图像用 x_i 和 $x_{i'}$ 指出。例如，这可能是两个不同的"3"。通过每幅图像，我们绘制了该图像的旋转版本曲线，称为不变性流形(invariance manifolds)。现在，我们使用两条曲线间的最短距离，而不使用通常的两幅图像间的欧几里德距离。换句话说，两幅图像间的距离取为第一幅图像的任意一个旋转版本和第二幅图像的任意一个旋转版本间的最短欧几里德距离。这种距离叫做不变度量(invariant metric)。

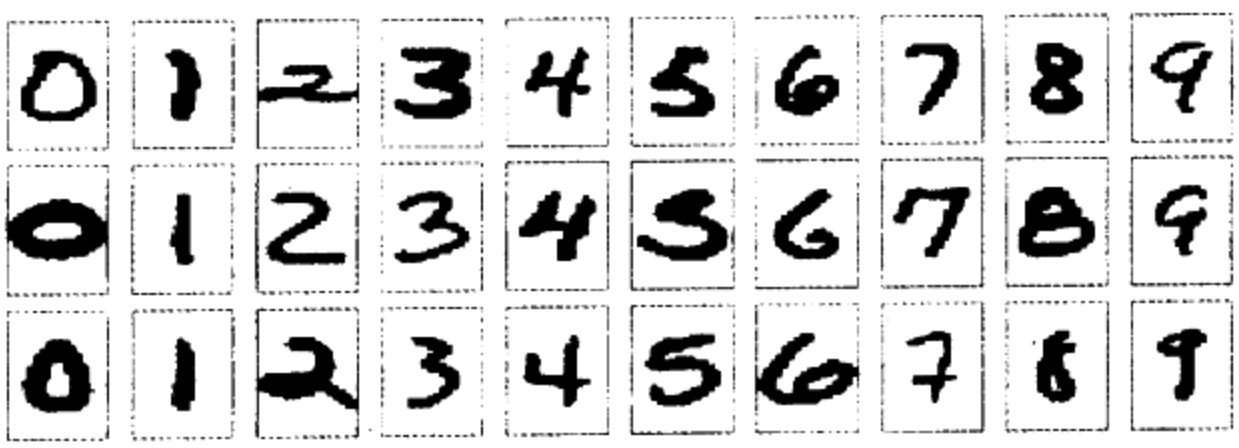


图 13.9 手写数字的灰度图像示例

原则上，我们可以使用这种不变度量实现 1 -最近邻分类。然而，这里有两个问题。第一，计算实际图像非常困难。第二，它允许大变换，可能导致很差的性能。例如，旋转 180°后，"6"将被看成近似的"9"。我们需要将注意力限制在较小的旋转上。

切距离(tangent distance)的使用解决了这两个问题。如图 13.10 所示，我们可以在其中用原图像的切线逼近图像"3"的不变性流形。这个切线可以通过估计图像的小旋转方向向量，或

通过更复杂的空间光滑方法(见习题 13.4)来计算。对于大旋转,切图像看上去不再像“3”。所以,使用大变换的问题会得到缓解。

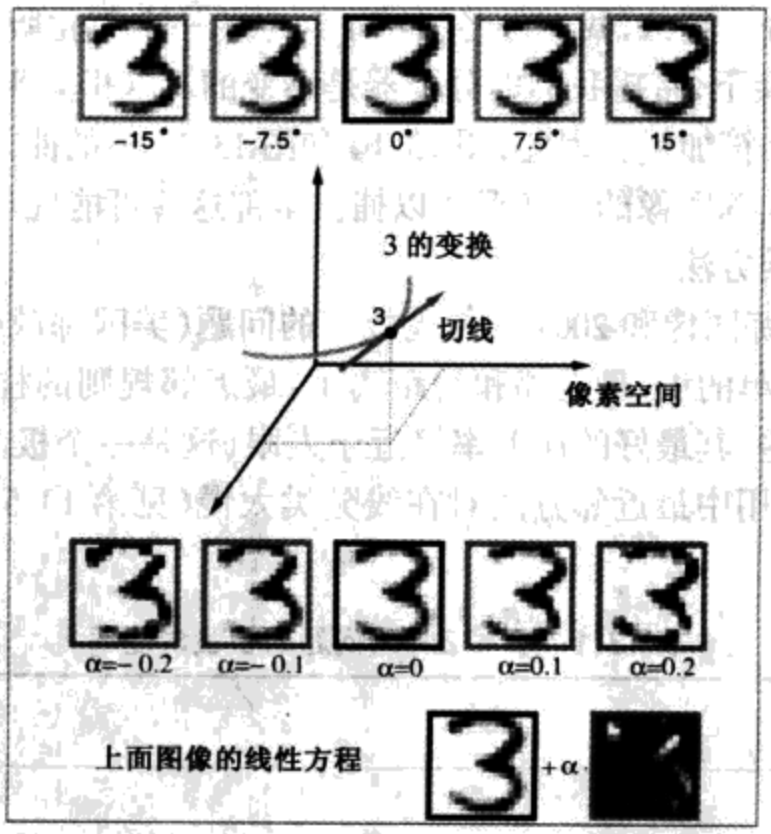


图 13.10　顶行显示原方向上的“3”(中间)及其旋转版本。图中部的曲线描绘了256维空间中旋转的“3”的集合,直线是曲线在原始图像上的切线,有一些“3”在该切线上,而曲线的方程显示在图的底部

这样,基本思想就是计算每幅训练图像的不变切线。对于一幅将要分类的查询图像,我们计算它的不变切线,并且在训练集的直线中发现离它最近的直线。对应于这条最近直线的类(数字)是我们对查询图像的预测类。在图 13.11 中,两条切线相交,但这仅仅是因为我们强行绘制了实际的 256 维情形的 2 维表示。在 $\mathbb{R}^{256}$ 中两条这样的线相交的概率实际是 0。

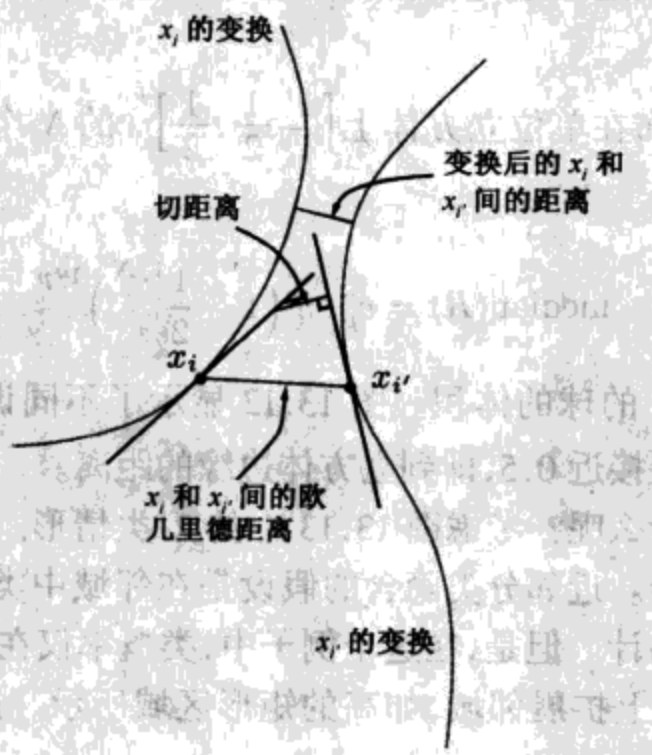


图 13.11　两幅图像 x_i 和 $x_{i'}$ 的切距离计算。我们使用了两条切线间的最短距离,而不是使用 x_i 和 $x_{i'}$ 间的欧几里德距离,或两曲线间的最短距离

现在，一种实现这种不变性较简单的方法是将每种训练图像的一些旋转形式加到训练集中，然后使用一个标准最近邻分类器。这种思想在 Abu-Mostafa(1995)中称为“暗示”，当不变空间较小时工作得非常好。到此，我们已经展示了问题的一个简化版本。除了旋转，还有其他6种变换类型，在这些变换下，希望我们的分类器是不变的。这里有平移(两个方向)、缩放(两个方向)、偏航(sheer)和字符加粗。因此，图 13.10 和图 13.11 中的曲线和切线实际上是 7 维流形和超平面。增加每个训练图像的变换形式以捕获全部这些可能性是不可行的。切流形提供了一种捕获不变性的上乘方法。

对于具有 7291 个训练图像和 2007 个检验数字的问题(美国邮政数据库)，表 13.1 显示了精心构造的神经网络、简单的 1-最近邻和切距离 1-最近邻规则的检验误分类误差。切距离最近邻分类器明显好一些，其最好的误差率接近于人眼(这是一个极其困难的检验集)。在实际中，已证明了在这种应用中最近邻方法对在线分类太慢(见第 13.5 节)，而神经网络分类器是为模仿它而开发的。

表 13.1 手写 ZIP 码问题的检验误差率

方法	误差率
神经网络	0.049
1-最近邻/欧几里德距离	0.055
1-最近邻/切距离	0.026

13.4 自适应的最近邻方法

在高维特征空间中执行最近邻分类时，点的最近邻可能非常远，会引起偏倚并降低规则的性能。

为量化它，考虑均匀分布在单位立方体上$\left[-\frac{1}{2},\frac{1}{2}\right]^p$的 N 个数据点。令 R 是中心在原点的 1-最近邻半径。则

$$\text{median}(R)=v_p^{-1/p}\left(1-\frac{1}{2}^{1/N}\right)^{1/p} \tag{13.7}$$

其中，$v_p r^p$ 是 p 维中半径为 r 的球的体积。图 13.12 显示了不同训练样本容量和维的中值半径。我们看到中值半径很快接近 0.5，即到立方体边缘的距离。

对于这个问题能做些什么呢？考虑图 13.13 中的 2-类情形。这里有两个特征，而查询点的最近邻域由圆形区域描绘。近邻分类隐含的假设是在邻域中类概率大致是常量，因此简单求平均就能够给出较好的估计。但是，在这个例子中，类概率仅在水平方向上不同。如果知道这一点，我们将在垂直方向上扩展邻域，如高的矩形区域所示。这将降低估计偏倚而方差不变。

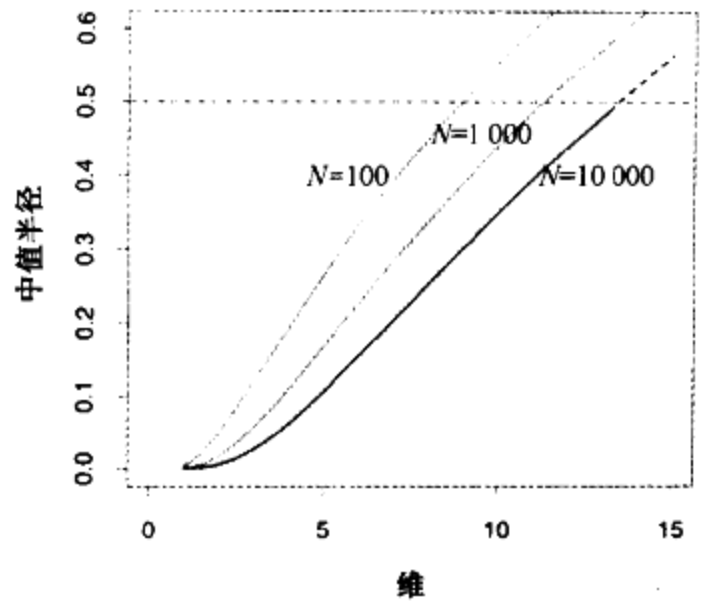


图 13.12　p 维中 N 个观测的均匀数据的 1 -最近邻域的中值半径

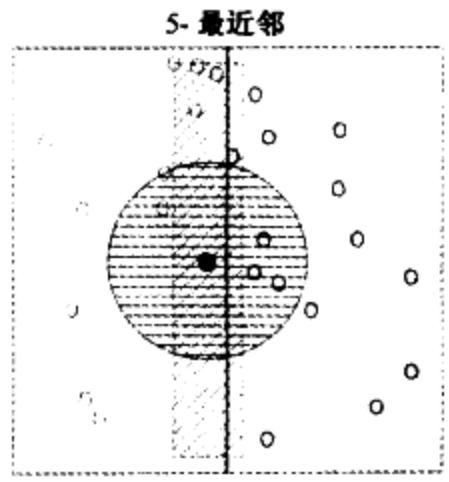


图 13.13　立方体上均匀分布的点，垂线将红色类和绿色类分隔开。垂直的带代表仅使用水平坐标发现目标点(实点)的最近邻的5-最近邻区域。球显示了使用两个坐标的5-最近邻区域，我们看到在这种情况下它已经伸展到红色类区域(而且该实例被错误类控制)(见彩页)

一般地说，这要求调整最近邻分类使用的度量，使得结果邻域在类概率变化不大的方向上伸展。在高维特征空间中，类概率可能仅在一个低维子空间改变，因此自适应度量可能有相当大的好处。

Friedman(1994a)提出一种方法，通过相继截开包含训练数据的盒子的边缘，自适应地发现矩形邻域。这里将介绍 Hastie 和 Tibshirani(1996a)的判别自适应最近邻(DANN)规则。早些时候，相关的提议出现在 Short 和 Fukunaga(1981)及 Myles 和 Hand(1990)的论文中。

在每个查询点上形成一个邻域，例如，包含 50 个点的邻域；使用这些点上的类分布决定怎样改变邻域的形状，即调整度量。然后，自适应的度量用于查询点上的最近邻规则。这样，在每个查询点上使用一种潜在不同的度量。

在图 13.13 中，邻域显然应当在与类中心连线正交的方向上伸展。该方向也与线性判别边界一致，并且类概率在该方向上改变最少。通常，最大改变的方向将不与连接类中心的线正交(见图 4.9)。假设有一个局部判别模型，包含在局部类内和类间协方差矩阵中的信息正是决定邻域最优外形需要的全部信息。

查询点 x_0 的判别自适应最近邻(DANN)度量由下式定义：

$$D(x, x_0) = (x - x_0)^T \mathbf{\Sigma} (x - x_0) \tag{13.8}$$

其中：

$$\begin{aligned}\Sigma &= \mathbf{W}^{-1/2}[\mathbf{W}^{-1/2}\mathbf{B}\mathbf{W}^{-1/2}+\epsilon\mathbf{I}]\mathbf{W}^{-1/2}\\ &= \mathbf{W}^{-1/2}[\mathbf{B}^{*}+\epsilon\mathbf{I}]\mathbf{W}^{-1/2}\end{aligned} \tag{13.9}$$

这里，$\mathbf{W}$ 是合并的类内协方差矩阵 $\sum_{k=1}^{K}\pi_k\mathbf{W}_k$，而 $\mathbf{B}$ 是类间协方差矩阵 $\sum_{k=1}^{K}\pi_k(\bar{x}_k-\bar{x})(\bar{x}_k-\bar{x})^T$，$\mathbf{W}$ 和 $\mathbf{B}$ 仅使用 x_0 周围 50 个最近邻进行计算。在计算度量之后，它被用于 x_0 上的最近邻规则中。

这个复杂公式的计算实际上非常简单。首先关于 $\mathbf{W}$ 将数据球形化，然后在 $\mathbf{B}^{*}$（球形化数据的类间矩阵）的 0 - 本征值方向上扩展邻域。这很有意义，因为局部地，观测到的类均值在这些方向上都相同。参数 ϵ 围绕着邻域，从一条无限带到一个椭圆体，以避免使用离查询点较远的点。$\epsilon = 1$ 在通常情况下就工作得很好。图 13.14 显示了由类形成的两个同心圆问题的结果邻域。需要注意的是，当两个类出现在邻域中时，邻域是怎样垂直伸展到判定边界的。在只有一个类的纯区域中，邻域保持圆形；在这些情形下，式(13.8)中的类间矩阵 $\mathbf{B}=0$，且 $\mathbf{\Sigma}$ 是恒等矩阵。

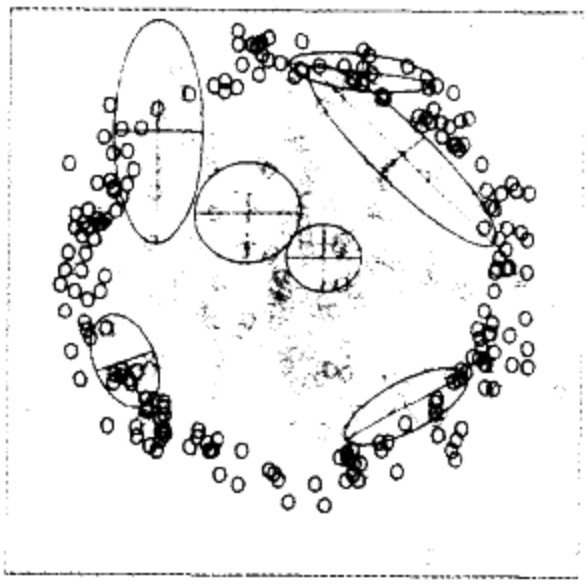


图 13.14 在不同的查询点(十字形中心)，由 DANN 过程发现的邻域。数据中有两个类，其中一个类包围另一个类。用50个最近邻估计局部度量。显示的是用于形成15-最近邻的结果度量

13.4.1 例

这里，我们产生 10 维空间中的 2-类数据，与图 13.14 中的 2 维例子类似。类 1 中的全部 10 个预测子是独立标准正态的，条件是半径大于 22.4 而小于 40；类 2 中的预测子是独立标准正态的，无限制条件。每个类中有 250 个观测。因此，在整个 10 维空间中第一个类几乎完全包围了第二个类。

在这个例子中，没有纯噪声变量，即没有最近邻子集选择规则可以剔除的一类变量。在特征空间中的任意给定点上，类辨别仅发生在一个方向上。然而，当我们在整个特征空间中移动时这个方向会改变，并且所有变量将在空间中的某个地方是重要的。

图 13.15 显示了标准 5 - 最近邻、LVQ 和判别自适应 5 - 最近邻的 10 次实现的检验误差率的盒图。对 LVQ 使用每类 50 个原型，使它可与 5 - 最近邻比较(因为 250/5 = 50)。与 LVQ 或标

准最近邻比较，自适应度量的误差率明显降低了。

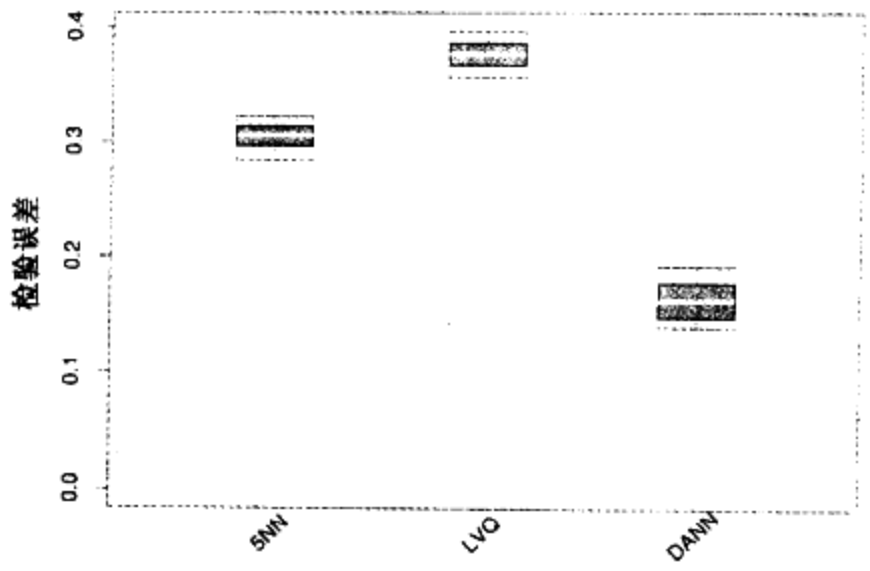


图 13.15　10 维模拟例：标准 5 - 最近邻、有 50 个中心的 LVQ 和判别自适应5- 最近邻的10次实现的检验误差率的盒图

13.4.2　最近邻的全局维归约

判别自适应最近邻方法实现了局部维归约，即在每个查询点上分别进行维归约。在许多问题中，我们可以从全局维归约中获益。即在原特征空间中选出的某优化子空间中应用最近邻规则。例如，假设两个类在特征空间的 4 维中形成两个嵌套的球，并且有 6 个附加的噪声特征，其分布独立于类。那么，我们希望发现这个重要的 4 维子空间，并在该归约的子空间中执行最近邻分类。为此，Hastie 和 Tibshirani(1996a)讨论了判别自适应最近邻方法的变种。在每个训练点 x_i，计算平方矩阵 $\mathbf{B}_i$ 的中心间的和，然后在所有训练点上对这些矩阵求平均：

$$\bar{\mathbf{B}} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{B}_i \tag{13.10}$$

令 $e_1, e_2, \cdots, e_p$ 是矩阵 $\bar{\mathbf{B}}$ 的本征向量，按本征值 θ_k 从大到小排列的。则这些本征向量生成全局子空间归约的最佳子空间。其由来基于如下事实：对 $\bar{B}$ 的最佳 L 秩逼近（其中，$\bar{\mathbf{B}}_{[L]} \sum_{\ell=1}^{L}\theta_\ell e_\ell e_\ell^T$）解决了最小二乘方问题

$$\min_{\text{rank}(\mathbf{M})=L}\sum_{i=1}^{N}\text{trace}[(\mathbf{B}_i-\mathbf{M})^2] \tag{13.11}$$

由于每个 $\mathbf{B}_i$ 包含局部判定子空间和该子空间的判别强度信息，所以式(13.11)可以看做是通过加权的最小二乘方对一系列 N 个子空间发现 L 维的最佳逼近子空间的方法(见习题 13.5)。

在上面提到并在 Hastie 和 Tibshirani(1996a)中考察过的 4 维空间例子中，4 本征值 θ_ℓ 中的 4 个是大的(其本征向量几乎生成感兴趣的子空间)，而余下的 6 个本征值接近于 0。操作上，我们把数据投影到主要的 4 维子空间中，然后执行最近邻分类。在第 13.3.2 节的卫星图像分类例中，图 13.8 中标有 DANN 的技术在一个全局归约子空间中使用了 5 - 最近邻方法。这种技术与 Duan 和 Li(1991)的分片(sliced)逆回归建议也有联系。在回归处理中，这些作者使用了相似的思想，但进行全局而非局部地计算。他们假定并利用了特征分布的球形对称性以估计令人感兴趣的子空间。

13.5 计算考虑

通常，最近邻规则的一个缺点是发现最近邻和存储整个训练集的计算负荷。对于 N 个观测和 p 个预测子，最近邻分类需要 Np 次操作以发现每个查询点的最近邻。有一些用于发现最近邻的快速算法(Friedman 等，1975、Friedman 等，1977)，该算法在某种程度上可以降低这种负荷。Hastie 和 Simard(1998)通过在不同度量的背景下开发 K-均值聚类的相似物来降低切距离的计算量。

减少存储的需求更困难，已经提出各种各样的编辑和压缩过程。其思想是隔离一个满足最近邻预测的训练集的子集，抛弃剩余的训练数据。直观地，保留靠近判定边界且在这些边界正确侧的训练点看来是重要的，而远离边界的一些点可以舍弃。

Devijver 和 Kittler(1982)的多编辑(multi -edit)算法循环地将数据分为训练集和检验集，在训练集上计算最近邻规则并删除误分类的检验点。这种思想是试图保持训练观测的同质聚类。

Hart(1968)的压缩过程则更进一步，它试图仅保留这些聚类重要的外部点。以单个随机选择的观测作为训练集开始，每个附加数据项一次处理一个，仅当它被当前训练集上计算的最近邻规则误分类时，才把它加到训练集中。

这些过程的全面综述在 Dasarathy(1991)和 Ripley(1996)中。它们也可以应用于除最近邻方法以外的其他学习过程。尽管这样的方法在某些时候是有用的，但我们既没有太多关于它们的实际经验，也没有在文献中发现任何有关它们性能的较系统的比较。

文献注释

最近邻方法至少可以追溯到 Fix 和 Hodges(1951)。关于这个主题更详尽的文献由 Dasarathy(1991)给出。Ripley(1996)的第 6 章包含着一个好的总结。K-均值聚类源于 Lloyd(1957)和 MacQueen(1967)。Kohonen(1989)引进了学习向量量化。切距离方法源自于Simard等人(1993)的著作。Hastie 和 Tibshirani(1996a)提出了判别自适应最近邻技术。

习题

13.1 考虑一个高斯混合模型，其中假设协方差矩阵是标量：$\Sigma_r = \sigma \mathbf{I}$，$\forall\, r = 1, \cdots, R$ 且 σ 是一个固定参数。详细讨论拟合这个混合模型时，K-均值聚类算法和 EM 算法之间的相似性。证明在 $\sigma \to 0$ 时两个方法是一致的。

13.2 导出 1 -最近邻域的中值半径公式(13.7)。

13.3 令 E^* 是 K-类问题贝叶斯规则的误差率，其中真实类概率由 $p_k(x)$，$k = 1, \cdots, K$ 给出。假设检验点和训练点有相同的特征 x，证明式(13.5)：

$$\sum_{k=1}^{K} p_k(x)(1 - p_k(x)) \leqslant 2(1 - p_{k^*}(x)) - \frac{K}{K-1}(1 - p_{k^*}(x))^2$$

其中，$k^* = \arg\max_k p_k(x)$。因此随着训练集容量的增加，1 -最近邻规则的误差率在 L_1 中收敛值 E_1，上方受限于：

$$E^*\left(2 - E^*\frac{K}{K-1}\right) \tag{13.12}$$

[Cover 和 Hart(1967)定理的这种陈述取自 Ripley(1996)第 6 章，那里也给出了简短的证明。]

13.4 把一个图像看做二维空间(纸坐标)上的函数 $F(x):\mathbb{R}^2 \mapsto \mathbb{R}^1$。那么 $F(c + x_0 + \mathbf{A}(x - x_0))$ 表示一个图像 F 的仿射变换，其中 $\mathbf{A}$ 是一个 2×2 矩阵。

1. 以这样一种方式分解 $\mathbf{A}$(通过 Q -R)，使得识别 4 种仿射变换(两种缩放、修剪和旋转)能被明显地标识。
2. 使用链规则，证明 $F(c + x_0 + \mathbf{A}(x - x_0))$关于每个这种参数的导数可以用 F 的两个空间导数表示。
3. 使用 2 维核光滑子(见第 6 章)，说明当图像被量化为 16×16 像素时，怎样实现这个过程。

13.5 令 $\mathbf{B}_i$，$i = 1, 2, \cdots, N$ 是 $p \times p$ 半正定矩阵，并令 $\overline{\mathbf{B}} = (1/N)\sum \mathbf{B}_i$。将 $\overline{B}$ 的本征分解写做 $\sum_{\ell=1}^{p}\theta_\ell e_\ell e_\ell^T$，其中 $\theta_\ell \geqslant \theta_{\ell-1} \geqslant \cdots \geqslant \theta_1$。证明 $\mathbf{B}_i$ 的最佳 L 秩逼近，

$$\min_{\text{rank}(\mathbf{M})=L}\sum_{i=1}^{N}\text{trace}[(\mathbf{B}_i - \mathbf{M})^2]$$

由 $\overline{\mathbf{B}}_{[L]} = \sum_{\ell=1}^{L}\theta_\ell e_\ell e_\ell^T$ 给出。(提示：把 $\sum_{i=1}^{N}\text{trace}[(\mathbf{B}_i - \mathbf{M})^2]$ 写成：

$$\sum_{i=1}^{N}\text{trace}[(\mathbf{B}_i - \bar{\mathbf{B}})^2] + \sum_{i=1}^{N}\text{trace}[(\mathbf{M} - \bar{\mathbf{B}})^2])$$

13.6 这里考虑外形平均(shape averaging)问题。特别地，$\mathbf{L}_i(i = 1, \cdots, M)$是 $\mathbb{R}^2$ 中点的 $N \times 2$ 矩阵，每个都是从手写(草写的)字母的相应位置抽取的样本。我们寻找一个仿射不变量平均 $\mathbf{V}$，也是 $N \times 2$ 矩阵，$\mathbf{V}^T\mathbf{V} = I$，具有如下性质：$\mathbf{V}$ 极小化

$$\sum_{j=1}^{M}\min_{\mathbf{A}_j}\|\mathbf{L}_j - \mathbf{V}\mathbf{A}_j\|^2$$

描述它的解。

如果某些字母较大且控制平均值，则该解会受到损害。一个替代的方法是极小化：

$$\sum_{j=1}^{M}\min_{\mathbf{A}_j}\|\mathbf{L}_j\mathbf{A}_j^* - \mathbf{V}\|^2$$

导出这个问题的解。准则有何不同？使用 $\mathbf{L}_j$ 的 SVD，可以简化两种方法的比较。

13.7 对于图 13.5 左图中的“易”和“难”问题，考虑最近邻方法的应用。

1. 重复图 13.5 左图的结果。
2. 使用 5 折交叉验证估计误分类误差，并将它们的误差率曲线与 1 中的做比较。
3. 考虑训练集误分类误差的“类 AIC”罚。特别地，对训练集误分类误差增加 $2t/N$，其

中 t 是参数 N/r 的近似数，r 是最近邻数。将结果罚误分类误差图与 1 和 2 中的比较。哪种方法给出了最优最近邻数的较好估计，交叉验证还是 AIC?

13.8 在两个类中产生数据，有两个特征。这些特征都是独立高斯变量，具有标准差 1。它们的均值向量在类 1 中是 $(-1,-1)$，在类 2 中是 $(1,1)$。对每个特征向量随机旋转一个角度 θ，θ 从 0 到 2π 中均匀地选取。从每个类中产生 50 个观测形成训练集，在每个类中产生 500 个观测作为检验集。应用以下 4 种不同的分类子：

1. 最近邻。
2. 有提示的最近邻：在应用最近邻之前，将每个数据点的 10 个随机旋转形式增加到训练集中。
3. 不变度量最近邻，使用对原点旋转的欧几里德距离不变式。
4. 切距离最近邻。

在每种情况下，通过 10 折交叉验证选择近邻数目。比较这些结果。

第 14 章　无指导学习

14.1　引言

前几章涉及给定输入集合，预测一个或多个输出值；或给定预测子变量 $X = (X_1, \cdots, X_p)$，预测其响应变量 $Y = (Y_1, \cdots, Y_m)$。记第 i 个训练实例的输入为 $x_i = (x_{i1}, \cdots, x_{ip})$，令 y_i 为响应度。预测是基于预先已求解的训练样本 $(x_1, y_1), \cdots (x_N, y_N)$ 而进行的，其中所有变量的联合值是已知的。我们把这种预测称为有指导学习(supervised learning)或“有导师的学习”。这可以比喻为“学生”为每个训练样本 x_i 提供一个结果 $\hat{y}_i$，而指导者或“导师”提供正确结果和(或)学生结果的误差。通常，这用某种损失函数 $L(y, \hat{y})$ 来表征，例如，用 $L(y, \hat{y}) = (y - \hat{y})^2$ 来表征。

如果假定 (X, Y) 为由某联合概率密度 $\Pr(X, Y)$ 表示的随机变量，那么有指导学习就可以形式地刻画为一个密度估计问题，其中涉及到条件密度 $\Pr(Y \mid X)$ 的决定特性。通常情况下，感兴趣的特性是那些“位置”参数 μ，它们使得每个 x 上的期望误差最小，

$$\mu(x) = \underset{\theta}{\operatorname{argmin}} E_{Y|X} L(Y, \theta) \tag{14.1}$$

据条件，我们有：

$$\Pr(X, Y) = \Pr(Y|X) \cdot \Pr(X)$$

其中，$\Pr(X)$ 只是 X 值的联合边缘密度。在有指导学习中，一般不直接涉及 $\Pr(X)$，主要关心的是条件密度 $\Pr(Y \mid X)$ 的特性。由于 Y 经常是低维的(通常为 1 维)，并且我们只关心其位置 $\mu(x)$，这样问题就大大简化了。正如在前几章中所论述的，在不同的背景下有多种方法能够很好地处理有指导的学习。

本章我们来讲解无指导学习(unsupervised learning)或“无导师的学习”。在这种情况下，对于一个联合密度为 $\Pr(X)$ 的随机 p 维向量 X，已知其 N 个观测的集合 $(x_1, x_2, \cdots, x_N)$。目标是在没有指导者或导师为每个观测提供正确结果或误差度的情况下，直接推断出概率密度的特性。有时，X 的维数会比有指导学习的维数高得多，并且我们感兴趣的特性常常比简单的位置估计要复杂很多。由于令 X 表示所有涉及到的变量，这些因素的影响将会有一些减轻；我们没有必要在依赖于另一组变量变化的条件下，推断 $\Pr(X)$ 变化的特性。

在低维问题中(比如 $p \leqslant 3$)，有多种有效的非参数方法可以对所有 X 值直接估计密度 $\Pr(X)$ 本身，并可以利用图形来表示(例如 Silverman, 1986)。由于维灾难问题，这些方法在高维中将会失败。我们必须勉强接受相当粗糙的全局模型的估计，比如混合的高斯分布或刻画 $\Pr(X)$ 的各种简单的描述性统计数据。

一般地，这些描述性统计数据试图刻画 X 值或者 X 值的集合，在这些 X 上 $\Pr(X)$ 相对较

大。例如,主成分、多维定标、自组织映射以及主曲线等,都试图识别 X 空间上的高数据密度的低维流形。无论是否可以把它们看做较小的"本征"变量集合上的函数,这都将提供有关变量间的关联信息。聚类分析试图寻求包含 $\Pr(X)$众数的 X 空间的多元凸区域,它将断定 $\Pr(X)$能否由一个混合的密度函数来表示,其中组成这些混合的是一些代表不同类型或不同类别的观测的较为简单的密度函数。混合建模也有类似的目的。关联规则试图构建简单的描述(合取规则),用以刻画非常高维的二值数据特例的高密度区域。

在有指导学习中,我们对成功或失败有一个明确的度量。该度量能用于判断模型在特殊情况下的适应性,还能用于比较各种情况下不同方法的有效性。是否成功可以直接通过联合分布 $\Pr(X,Y)$的期望损失来度量。这可以通过包括交叉验证在内的多种方法来估计。在无指导学习的背景下就没有成功的直接度量。对于大多数无指导学习的算法来说,人们难于从它输出的结果确定其推理的有效性。这时我们必须借助于启发式论据,不仅要用于诱导算法(像有指导学习中的通常情况那样),而且还要用于判断结果的质量。由于有效性是一个主观评判而不能直接验证,所以这种不利状况导致了可选方法大量增加。

本章将介绍实践中最通用的无指导学习技术,另外还介绍了作者偏爱的几种技术。

14.2 关联规则

关联规则分析已经成为挖掘商业数据库的一个流行工具。其目标是寻求数据库中最频繁出现的变量 $X=(X_1,X_2,\cdots,X_p)$的联合值。该技术最常用于二值数据 $X_j\in\{0,1\}$,在那里它称为"购物篮"分析。在此背景下,观测是销售事务,诸如商场收银台的事务等。变量代表商场销售的所有商品。对于观测 i,每个变量 X_j 被赋予二值之一;如果第 j 种商品在交易中售出,则 $x_{ij}=1$,反之 $x_{ij}=0$。频繁地具有联合值 1 的那些变量代表频繁地同时购买的商品。这些信息对货架库存、促销货品搭配、价目表设计,以及以根据购买模式对消费者分类等都非常有用。

更一般地,关联规则分析的基本目标是为特征向量 X 求解一个原型 X 值的集合$\nu_1,\cdots,\nu_L$,使得在这些值上计算的概率密度 $\Pr(\nu_l)$相对较大。在这个通用的构架下,该问题可以看做是"众数发现"或"凸点搜索"。至于用公式表示,这个问题将非常困难。对每个 $\Pr(\nu_l)$的一个自然估计是 $X=\nu_l$ 的观测所占的比值。对于涉及的变量较多,且每个变量可能具有多个值的问题,$X=\nu_l$ 的观测的个数对于可靠的估计来说几乎总是太小。因此,为了得到一个易于处理的问题,分析的目标和用于分析的数据的一般性都必须大为简化。

首要的简化是修改目标。我们要寻找的是相对于样本量或支持度具有高概率的 X 空间区域(region),而不是寻找 $\Pr(x)$大的 x 值。令s_j 表示第j 个变量所有可能值的集合(支集),并且令 $s_j\subseteq s_j$ 为这些值的子集。修改后的目标为试图求解变量值的子集 $s_1,\cdots,s_p$,使得每个变量同时在其对应的子集中取值的概率,

$$\Pr\left[\bigcap_{j=1}^{p}(X_j\in s_j)\right] \tag{14.2}$$

相对较大。子集的交集$\bigcap_{j=1}^{p}(X_j\in s_j)$称为合取规则(conjunctive rule)。对于数量型变量,子集

s_j 是一些相邻的区间；对于分类型变量，子集给出了明确的描述。注意经常有这种情况：如果子集 s_j 实际上是 X_j 值的全集，即 $s_j = s_j$，则称变量 X_j 不在规则(14.2)中出现。

14.2.1　购物篮分析

求解式(14.2)的通用方法将在第 14.2.5 节中讨论。它们在许多应用中都非常有用，但是对于大规模($p \approx 10^4$，$N \approx 10^8$)的商业数据库却不可行，这时常常用到的是购物篮分析。式(14.2)需要再做一些简化。首先，只考虑两种类型的子集：s_j 仅包括 X_j 的一个值，$s_j = v_{0j}$；或者 s_j 包括 X_j 设定取值的整个集合，$s_j = s_j$。这将问题(14.2)简化为求整数 $\mathcal{J} \subset \{1, \ldots, p\}$ 的子集和对应值 v_{0j}，$j \in \mathcal{J}$，使得

$$\Pr\left[\bigcap_{j\in\mathcal{J}}(X_j = v_{0j})\right] \tag{14.3}$$

比较大。图 14.1 显示了这种假设。

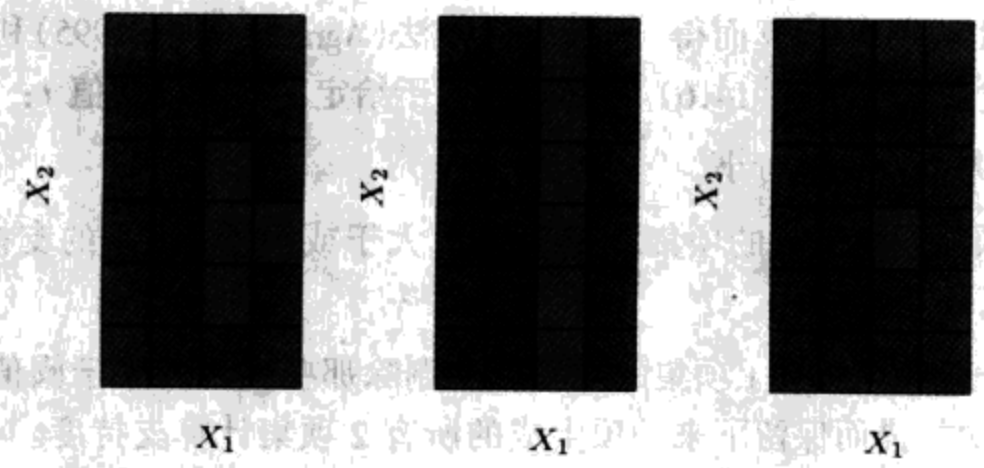


图 14.1　关联规则的简化。有两个输入 X_1 和 X_2，分别取 4 个和 6 个不同值。红色方块表示高密度区域。为了简化计算，我们假定导出的子集对应输入的一个值或所有值。在这个假定下，能够求得中间或右边的模式，但得不到左边的模式(见彩页)

我们可以应用哑变量(dummy variables)技术将式(14.3)变换成只涉及二值变量的问题。这里假定对每个变量 X_j 的支集样本数 S_j 是有限的。特别地，创建一个新的变量集合 $Z_1, \cdots, Z_K$，每个变量代表一个可以由原始变量 $X_1, \cdots, X_p$ 获得的值 v_{ij}。哑变量的个数 K 是：

$$K = \sum_{j=1}^{p} |S_j|$$

其中，$|S_j|$ 是由 X_j 得到的不同值的个数。对于每个哑变量 Z_K，如果与之关联的变量取指派给 Z_k 的对应值，则 $Z_k = 1$，否则赋值 $Z_k = 0$。这样就将式(14.3)转换为求整数 $\mathcal{K} \subset \{1, \ldots, K\}$ 的子集，使得

$$\Pr\left[\bigcap_{k\in\mathcal{K}}(Z_k = 1)\right] = \Pr\left[\prod_{k\in\mathcal{K}} Z_k = 1\right] \tag{14.4}$$

比较大。这是购物篮问题的标准公式。集合 $\mathcal{K}$ 称做一个“项集”。(译注：严格地说，引入哑变量 $Z_1, \cdots, Z_K$ 后，$\mathcal{K}$ 是下标集，而 $\{Z_k \mid k \in \mathcal{K}\}$ 是项集。为简化记号，文中简单地用 $\mathcal{K}$ 表示项集。)

项集中变量 Z_k 的个数称为项集的“规模”。(注意规模不会大于 p)。式(14.4)的估计值为数据库中(14.4)合取值为真的观测的比值:

$$\widehat{\Pr}\left[\prod_{k\in\mathcal{K}}(Z_k=1)\right]=\frac{1}{N}\sum_{i=1}^{N}\prod_{k\in\mathcal{K}}z_{ik} \tag{14.5}$$

这里,z_{ik}是第 i 种情况的 Z_k 值,它称为项集$\mathcal{K}$的“支持度”或“流行度”$T(\mathcal{K})$。我们称 $\prod_{k\in\mathcal{K}}z_{ik}=1$ 的观测 i“包含”项集$\mathcal{K}$ 。

在关联规则挖掘中,指定一个支持度下界 t,并且寻求变量 $Z_1,\cdots,Z_K$ 能够生成的所有项集$\mathcal{K}_l$,它们在数据库中支持度大于该下界 t

$$\{\mathcal{K}_l\mid T(\mathcal{K}_l)>t\} \tag{14.6}$$

14.2.2 Apriori 算法

调整阈值 t 使得式(14.6)只包括全部 2^K 个可能项集中的一小部分,对于大型数据库,问题(14.6)的解可以由可行的计算而得。“Apriori”算法(Agrawal 等人,1995)利用维灾难的几个特点,通过少数几次数据扫描式(14.6)。特殊地,对于给定的支持度阈值 t:

- 基数||$\mathcal{K}\mid T(\mathcal{K})>t$||相对小。
- 任何由$\mathcal{K}$中项的子集组成的项集$\mathcal{L}$,必须具有大于或等于项集$\mathcal{K}$的支持度,即$\mathcal{L}\subseteq\mathcal{K}\Rightarrow T(L)\geqslant T(\mathcal{K})$。

第一遍扫描数据,计算所有 1 项集的支持度。删除那些支持度小于阈值的项。第二次扫描对可以通过第一次扫描而保留下来的项生成的所有 2 项集计算支持度。也就是说,要生成所有满足$|\mathcal{K}|=m$ 的频繁项集,我们仅需要考虑这样的 m 项集候选,它们的 m 个规模为 $m-1$ 的祖先项集都是频繁的。删除那些支持度小于阈值的 2 项集。每一遍后继数据扫描只考虑能由前一遍扫描保留项集与第一遍扫描留下的项组合所生成的项集。继续扫描数据,直到由上一步得到的所有候选项集的支持度都小于指定的阈值。对于$|\mathcal{K}|$的每个值,Apriori 算法只需要扫描一次数据。这是至关重要的,因为我们假定所有数据不能全部装入计算机内存。如果数据足够稀疏(或者,如果阈值 t 足够大),即使对巨大的数据集,该过程也能在合理的时间内终止。

还有一些其他的技巧,可以用做加速运算和收敛的策略(Agrawal 等人, 1995)。Apriori 算法代表数据挖掘技术的主要进展之一。

由 Apriori 算法得到的每个形如式的(14.6)的高支持度项集$\mathcal{K}$(14.6)产生一组“关联规则”。项 $Z_k,k\in\mathcal{K}$被划分为两个不相交的子集,$A\cup B=\mathcal{K}$,并记为:

$$A\Rightarrow B \tag{14.7}$$

第一个子项集 A 称做“前件”,第二个子项集 B 称做“后件”。关联规则具有一些特性,这些特性基于前件和后件项集在数据库中的频繁出现。规则的“支持度”$T(A\Rightarrow B)$是前件和后件并集中观测的比例,这正是由其推导出规则的项集$\mathcal{K}$的支持度。它可以看做是在随机选择的购物篮中同时观测到两个项集的概率 $\Pr(A,B)$的一个估计(14.5)。规则的“置信度”或“预测度”$C(A\Rightarrow B)$是规则的支持度除以其前件的支持度:

$$C(A \Rightarrow B) = \frac{T(A \Rightarrow B)}{T(A)} \tag{14.8}$$

它可以看做是 $\Pr(B|A)$的一个估计。记号 $\Pr(A)$，即项集 A 出现在同一个购物篮中的概率，是 $\Pr(\prod_{k\in A} Z_k = 1)$ 的缩写。“期望置信度”定义为后件支持度 $T(B)$，它是无条件概率 $\Pr(B)$的一个估计。最后，规则的“提升度 ”定义为置信度除以期望置信度：

$$L(A \Rightarrow B) = \frac{C(A \Rightarrow B)}{T(B)}$$

这是关联度量 $\Pr(A, B)/\Pr(A)\Pr(B)$的一个估计。

举例来说，假定项集为$\mathcal{K}$ = {peanut, butter, jelly, bread}，并考虑规则{peanut, butter, jelly} ⇒{bread}。该规则的支持度为 0.03，意味 peanut、butter、jelly 和 bread 同时出现在购物篮中的可能性为 3%。该规则的置信度为 0.82，意味着当 peanut、butter 和 jelly 被订购时，bread 也有 82% 的机会被订购。如果 bread 出现在 43% 的购物篮中时，规则{peanut, butter, jelly}⇒{bread}将有 1.95 的提升度 。

该分析的目标是产生支持度和置信度(14.8)二者同时都较高的关联规则(14.7)。Apriori 算法返回所有高支持度[如通过式(14.6)的支持度阈值 t 定义]的项集。设定一个置信度阈值 c，将得到由项集(14.6)形成的置信度大于该值的所有规则：

$$\{A \Rightarrow B \,|\, C(A \Rightarrow B) > c\} \tag{14.9}$$

对于每个规模为$|\mathcal{K}|$的项集$\mathcal{K}$，存在 $2^{|\mathcal{K}|-1} - 1$ 个形如 $A \Rightarrow (\mathcal{K} - A)$的规则，其中 $A \subset \mathcal{K}$。Agrawal 等人(1995)提出 Apriori 算法的一个改进版，该算法可以快速地判断在所有可能从项集(14.6)中导出的规则中哪些规则可以通过置信度的阈值(14.9)。

整个分析的输出是满足以下约束条件的关联规则(14.7)集合：

$$T(A \Rightarrow B) > t \quad \text{和} \quad C(A \Rightarrow B) > c$$

通常，这些存储在数据库中，可以由用户查询。典型的请求可能是按置信度、提升度或支持度排序显示规则。更特殊的请求可能是以前件甚至后件中某些特定项为条件的列表。例如，下面就是一个可能的请求：

显示置信度大于 80%、支持度大于 2%，并且以溜冰鞋为后件的所有交易。

这可能为那些预测溜冰鞋销售的项(前件)提供信息。关注特定后件将把问题引到有指导学习的体系中。

关联规则已经成为分析有关于购物篮等大型商用数据库的一个流行工具。即数据可以映射为多维列联表的形式。输出是易于理解和解释的合取规则(14.4)的形式。Apriori 算法允许该分析应用于巨型数据库，这大大超越了其他类型的分析对数据库规模的限制。关联规则是数据挖掘方面最大的成功。

除了对所应用数据形式上的限制外，关联规则还有其他的限制。计算可行性的关键是支持度的阈值(14.6)。解项集的数量、规模以及扫描数据次数随支持度下界的减小而呈指数级增长。因此，置信度或提升度较高，但支持度较低的规则将不被发现。例如，由于后件 caviar 的销售量少，一个形如 vodka⇒caviar 的高置信度规则将不被发现。

14.2.3 例:购物篮分析

我们来举例说明 Apriori 算法在中等规模统计数据库中的应用。该数据集合包括由旧金山海湾地区大型购物中心的客户所填写的 $N = 9409$ 份问卷(俄亥俄州哥伦布市 Impact 资源有限公司,1987)。为举例说明,这里使用与人口统计有关的前 14 个问题的答卷。这些问题如表 14.1 所列。可以看出,数据包括序数型和(无序的)分类型变量的混合,后者中有些具有较多的值。数据中还有一些遗漏值。

表 14.1 人口统计数据的输入

特征	统计量	值个数	类型
1	性别(sex)	2	分类的
2	婚姻状况(marital status)	5	分类的
3	年龄(age)	7	序数的
4	学历(education)	6	序数的
5	职业(occupation)	9	分类的
6	收入(income)	9	序数的
7	海湾居住年限(years in Bay Area)	5	序数的
8	双收入(dual income)	3	分类的
9	家庭人口(number in household)	9	序数的
10	子女数(number of children)	9	序数的
11	户主状态(householder status)	3	分类的
12	住宅类型(type of home)	5	分类的
13	民族(ethnic classification)	8	分类的
14	家庭语言(language in home)	3	分类的

使用 Christian Borgelt[①] 提供的 Apriori 算法的一个免费软件。删除具有遗漏值的观测之后,我们将每个序数型预测子在其中值处截开,并用两个哑变量来编码;对每个具有 k 个值的分类预测子,使用 k 个哑变量来编码。对于 6876 个观测、50 个哑变量,我们将得到一个 6876

① 参看 http://fuzzy.cs.uni-magdeburg.de/~borgelt。

×50 的矩阵。

该算法总计发现 6288 条关联规则，涉及少于 5 个的预测子，支持度最低为 10%。理解规则的大型集合本身就是一个富有挑战性的数据分析问题。这里不做这方面的尝试，只是解释图 14.2 中数据的哑变量的相对频率(上)和关联规则的相对频率(下)。主流的类型倾向于更经常出现在规则中，例如语种的第一类(英语)；而对于诸如职业等其他类型则表述不足，但第一个和第五个职业例外。

下面是由 Apriori 算法发现关联规则的三个例子：

关联规则 1：支持度 25%，置信度 99.7%，提升度 1.03

$$\{\text{家庭人口} = 1, \text{子女数} = 0\} \Rightarrow \{\text{家庭语言} = \text{英语}\}$$

关联规则 2：支持度 13.4%，置信度 80.8%，提升度 2.13

$$\{\text{家庭语言} = \text{英语}, \text{户主状态} = \text{拥有}, \text{职业} = \{\text{专业/管理人员}\}\} \Rightarrow \{\text{收入} \geqslant \$40\,000\}$$

关联规则 3：支持度 26.5%，置信度 82.8%，提升度 2.15

$$\{\text{家庭语言} = \text{英语}, \text{收入} < \$40\,000, \text{婚姻状况} = \text{未婚}, \text{子女数} = 0\} \Rightarrow \{\text{学历} \notin \{\text{研究生}, \text{在校研究生}\}\}$$

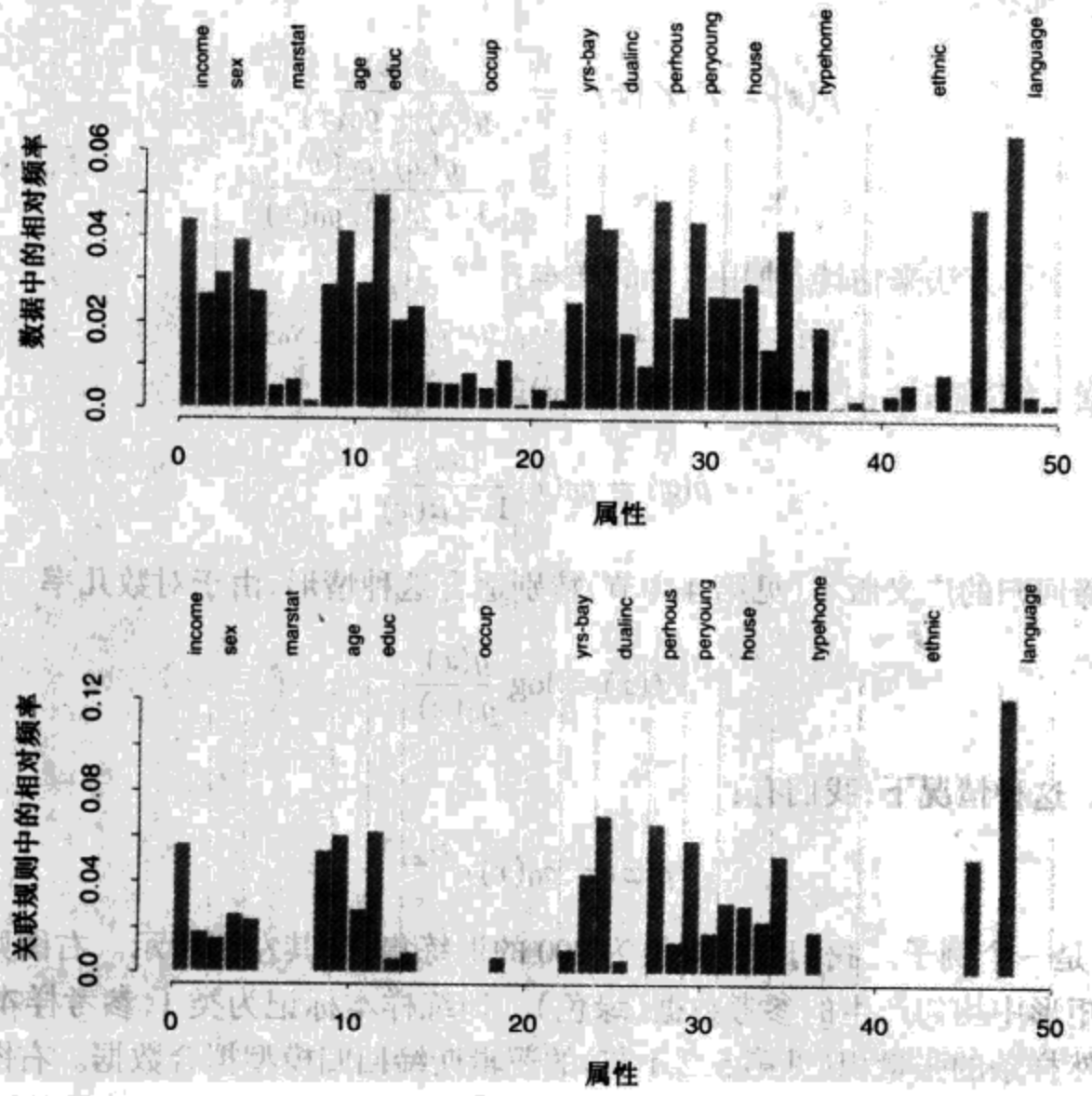


图 14.2　购物篮分析：数据中每个哑变量(编码一个输入类)的相对频率(上)，以及由 Apriori 算法发现的关联规则的相对频率(下)

基于高支持度，我们选择第一个和第三个规则。第二个规则是以高收入为后件的规则，可以用来确定高收入个体。

如上所述，我们为输入预测子的每个类创建了哑变量，例如，为低于和高于中值的收入分别创建：$Z_1 = I(\text{收入} < \$40\,000)$和 $Z_2 = I(\text{收入} \geqslant \$40\,000)$。如果我们感兴趣的是寻求有关高收入的关联规则，就可以只包括 Z_2而不包括 Z_1。实际的购物篮问题经常就是这种情况，我们只对寻求相对稀少的商品出现的关联感兴趣，而不是它不出现的关联。

14.2.4 作为有指导学习的无指导学习

这里，我们讨论把密度估计问题转换为一种有指导的函数逼近的技术。该技术构成了将在下一节讨论的广义关联规则的基础。

设$g(x)$为待估计的未知数据的概率密度，$g_0(x)$为用于推理的指定的概率密度函数。举例来说，$g_0(x)$可以是变量值域上均匀的密度，其他可能将在下面讨论。假定数据集 $x_1, x_2, \cdots, x_N$ 是抽自$g(x)$的一个独立同分布的随机样本。一个规模为 N_0 的样本可以用蒙特卡罗方法由$g_0(x)$抽取。将这两个数据集合并，并将质量 $w = N_0/(N + N_0)$赋予从$g(x)$得到的样本，将质量 $w_0 = N/(N + N_0)$ 赋予从$g_0(x)$得到的样本，导致一个从混合密度 $(g(x) + g_0(x))/2$ 抽样的随机样本。如果把 $Y = 1$ 赋予从$g(x)$抽取的每个样本点，把 $Y = 0$ 赋予从$g_0(x)$抽取的每个样本点，那么：

$$\begin{aligned} \mu(x) = E(Y \mid x) &= \frac{g(x)}{g(x) + g_0(x)} \\ &= \frac{g(x)/g_0(x)}{1 + g(x)/g_0(x)} \end{aligned} \tag{14.10}$$

可以用有指导学习方法来估计，使用组合的样本：

$$(y_1, x_1), (y_2, x_2), \cdots, (y_{N+N_0}, x_{N+N_0}) \tag{14.11}$$

作为训练数据。结果估计 $\hat{\mu}(x)$可以转换为$g(x)$的一个估计：

$$\hat{g}(x) = g_0(x)\frac{\hat{\mu}(x)}{1 - \hat{\mu}(x)} \tag{14.12}$$

逻辑斯缔回归的广义版本(见第 4.4 节)特别适合这种情形，由于对数几率

$$f(x) = \log \frac{g(x)}{g_0(x)} \tag{14.13}$$

被直接估计。这种情况下，我们有：

$$\hat{g}(x) = g_0(x)\, e^{\hat{f}(x)} \tag{14.14}$$

图 14.3 是一个例子。我们产生规模为 200 的训练集，如其左图所示。右图则显示在包含训练数据的矩形中均匀产生的参考数据(绿色)。训练样本标记为类 1，参考样本标记为类 0，并且使用自然样条的张量积(见第 5.2.1 节)的逻辑斯缔回归模型拟合数据。右图显示了$\hat{\mu}(x)$的一些概率等高线；它们也是密度估计$\hat{g}(x)$的等高线，因为$\hat{g}(x) = \hat{\mu}(x)/(1 - \hat{\mu}(x))$是单调函数。等高线粗略地反应了数据密度。

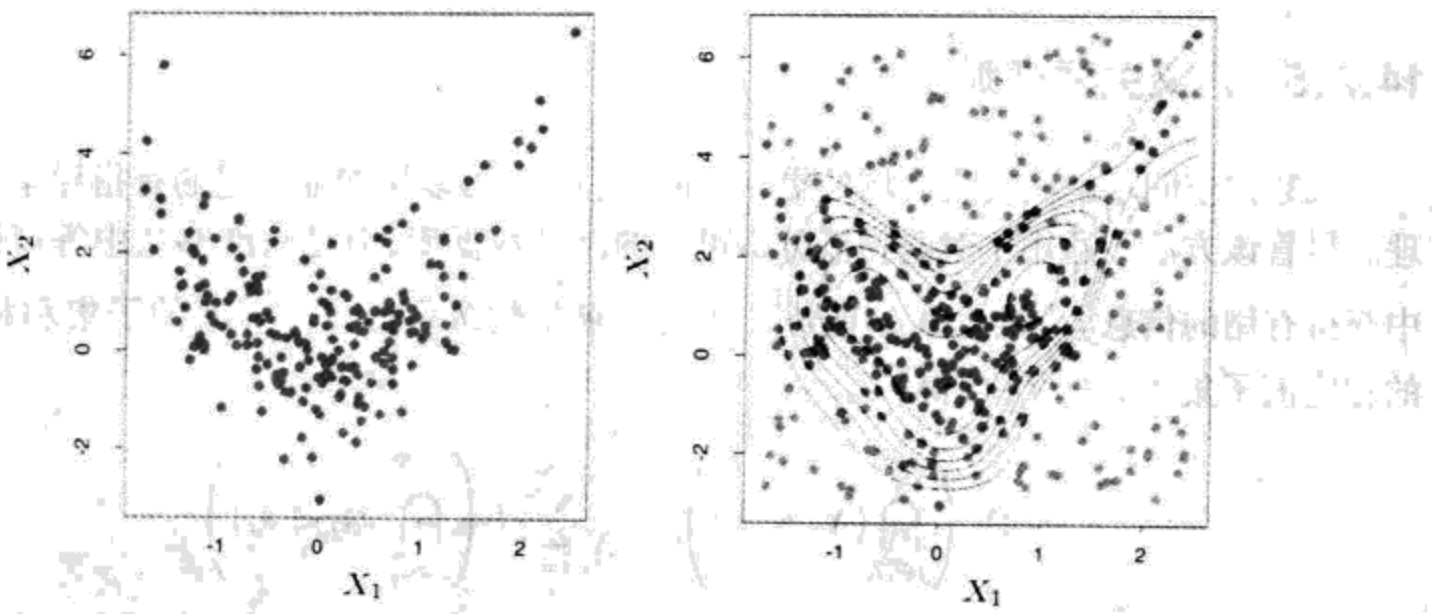


图 14.3　通过分类的密度估计。左图:200 个数据点的训练集。右图:训练集加上 200 个参考数据点,它们在包含训练数据的矩形框内均匀生成。训练样本标记为类1,参考样本标记为类0,并且用半参数化的逻辑斯缔回归模型拟合数据。图中显示了$\hat{g}(x)$的一些等高线(见彩页)

原则上,任何参考的密度都可以用于式(14.14)的$g_0(x)$。但是实际上估计$g(x)$的精度非常依赖于特定参考密度的选择。好的选择取决于数据密度$g(x)$和用于估计式(14.10)或式(14.13)的算法。如果目标是精确性,就应当选择这样的$g_0(x)$,它使得通过所采用的方法可以容易逼近结果函数$\mu(x)$或$f(x)$。但是,精确性常常不是主要的目标。$\mu(x)$和$f(x)$都是密度比率$g(x)/g_0(x)$的单调函数,因此它们可以看做提供有关数据密度$g(x)$与选择参考密度$g_0(x)$相差异的"对比"统计数据。所以在数据分析的环境下,可以根据我们认为在上下文中最感兴趣的特殊问题的差异类型来选择$g_0(x)$。例如,如果感兴趣的是与均匀性的差异,则$g_0(x)$可以是变量值域上的均匀密度。如果感兴趣的是与联合正态性的差异,好的$g_0(x)$选择应当是与数据具有相同均值向量和协方差矩阵的高斯分布;我们将在第 14.6.4 节的独立成分分析(Independent Components Analysis)中进一步探讨该问题。与独立性的差异可通过使用下式研究:

$$g_0(x)=\prod_{j=1}^{p}g_j(x_j) \tag{14.15}$$

其中,$g_j(x_j)$是 X 的第j 个坐标 X_j 的边缘数据密度。来自这些独立密度(14.15)的样本易于通过对每个变量数据值的不同随机数列由数据本身生成。

如上所述,无指导学习关心的是揭示数据密度$g(x)$的性质。每种技术都关注于某个特定的性质或性质的集合。将问题转化为一种有指导学习方法的式(14.10)~式(14.14)似乎在统计学界一度流传。尽管它具有把成熟的有指导学习方法引入无指导学习问题的潜在能力,但是并没有显示出多大的影响力。一个原因可能是用蒙特卡罗技术生成模拟数据集,问题就必须放大化。由于数据集规模应当至少和数据样本 $N_0(N_0\geqslant N)$相当,估计过程的运算和存储要求至少增加了一倍。此外,产生蒙特卡罗样本本身可能还需要可观的计算量。尽管这些在过去是一种很大的障碍,但随着可用资源不断增加,这些增加计算的需求已经不再是多大的负担。下一节我们阐述有指导学习方法在无指导学习中的应用。

14.2.5 广义关联规则

在数据空间求解高密度区域的更一般问题(14.2)可以用如前所述的有指导学习方法来处理。尽管该方法不适用于对购物篮分析可行的巨型数据库,但是它能够从中等规模的数据集中获得有用的信息。问题(14.2)能够形式化为求整数$\mathcal{J}\subset\{1,2,\cdots,p\}$的子集和相应变量$X_j$的相应值子集$s_j, j\in\mathcal{J}$,使得:

$$\widehat{\Pr}\left(\bigcap_{j\in\mathcal{J}}(X_j\in s_j)\right)=\frac{1}{N}\sum_{i=1}^{N}I\left(\bigcap_{j\in\mathcal{J}}(x_{ij}\in s_j)\right) \tag{14.16}$$

较大。沿用关联规则分析的术语,$\{(X_j\in s_j)\}_{j\in\mathcal{J}}$将被称为"广义"项集。与定量型变量相对应的子集$s_j$被取做其值域内的相邻区间,而分类型变量的子集可能涉及多个值。该公式的特性是排除对支持度(14.16)高于指定最小阈值的所有广义项集进行全面搜索,该情况在限制更严格的购物篮分析环境中是可能出现的。这里必须使用启发式搜索方法,并且我们最希望寻求广义项集的有用集合。

购物篮分析(14.5)和广义公式(14.16)都隐含地涉及到均匀概率分布。我们寻找这样的项集,假定所有的联合数据值$(x_1, x_2,\cdots, x_N)$都是均匀分布的,它们比期望的更频繁。这将有利于发现其边缘要素$(X_j\in s_j)$是个体(individually)频繁的项集,也就是说,量

$$\frac{1}{N}\sum_{i=1}^{N}I(x_{ij}\in s_j) \tag{14.17}$$

的值较大。相对于边缘不太频繁子集的合取,频繁子集(14.17)的合取更倾向于在高支持度项集中经常出现。这就是尽管规则 vodka⇒caviar 是高关联(提升)的却很可能不被发现的原因;每个项的边缘支持度都不高,因此它们的联合支持度就特别小。关于均匀分布,可能导致组成成分低关联的高频繁项集,它左右着最高支持度项集的集合。

高频繁子集s_j作为最频繁的X_j值的析取而形成。取变量边缘数据密度(14.15)的乘积作为参考分布,排除对在已发现项集中个别变量的高频繁值的优先考虑。这是因为如果变量间没有关联(完全独立),不管单个变量值的频率分布如何,密度比率$g(x)/g_0(x)$都是均匀的。诸如 vodka⇒caviar 的规则将有机会出现。然而,如何把除均匀分布之外的参考分布与 Apriori 算法相结合,目前仍然不清楚。正如在第 14.2.4 节中所解释的,给定原始数据集,直接从积密度(14.15)生成样本是直截了当的。

在选择参考分布,并且从中抽取样本之后[如式(14.11)],便获得一个带二值输出变量$Y\in\{0,1\}$的有指导学习问题。目标是利用训练数据求区域:

$$R=\bigcap_{j\in\mathcal{J}}(X_j\in s_j) \tag{14.18}$$

对该区域,目标函数$\mu(x)=E(Y\mid x)$的值相对较大。另外,我们也许还希望这些区域的数据支持度

$$T(R) = \int_{x \in R} g(x)\,\mathrm{d}x \tag{14.19}$$

不是太小。

14.2.6　有指导学习方法的选择

区域(14.18)由合取规则定义。因此学习这类规则的有指导学习方法将最适用于这种情况。CART 决策树的终端节点由符合式(14.18)形式的规则精确定义。把 CART 应用于合并的数据(14.11),将产生一棵决策树,它试图通过区域的不相交集合(终端节点)在整个数据空间建立目标(14.10)的模型。每个区域由形如式(14.18)的规则定义。那些具有高平均y值

$$\bar{y}_t = \mathrm{ave}(y_i \mid x_i \in t)$$

的终端节点 t 是高支持度广义项集(14.16)的候选。实际(数据)的支持度由下式给定:

$$T(R) = \bar{y}_t \cdot \frac{N_t}{N + N_0}$$

其中,N_t 是终端节点表示的区域内(合并后)观测的个数。通过检查结果决策树,我们可以发现支持度相对高的有趣广义项集。在对高置信度/或高提升度的广义关联规则的搜索中,该广义项集可能被划分为前件和后件。

对于同样的目标,另一个自然的学习方法是在第 9.3 节中介绍的忍耐规则归纳方法 PRIM。PRIM 也精确地生成符合式(14.18)形式的规则,但它是为求解具有最大平均目标(14.10)值的高支持度区域特别设计的,而不是试图在整个数据空间为目标函数建模。它还为在支持度/平均目标值阈值上的权衡提供更多的控制。

14.2.7　例:购物篮分析(续)

我们用表 14.1 中人口统计的数据来说明 PRIM 的使用。

如下所示是由 PRIM 分析形成的三个高支持度的广义项集:

项集 1: 支持度 = 24%

{婚姻状况 = 已婚,户主状态 = 拥有,住宅类型 ≠ 公寓}

项集 2: 支持度 = 24%

{年龄 ≤ 24,婚姻状况 ∈ {未婚同居,单身},职业 ∉ {专业人员,家庭主妇,退休},
户主状态 ∈ {租房,全家同住}}

项集 3: 支持度 = 15%

{户主状态 = 租房,住宅类型 ≠ 独门独院,家庭人口 ≤ 2,子女数 = 0,
职业 ∉ {家庭主妇,学生,失业},收入 ∈ [$20 000,$150 000]}

下面是从这些项集推导而来的置信度(14.8)大于 95% 的广义关联规则:

关联规则 1: 支持度 25%,置信度 99.7%,提升度 1.35

$$\{\text{婚姻状况} = \text{已婚}, \text{户主状态} = \text{拥有}\} \Rightarrow \{\text{住宅类型} \neq \text{公寓}\}$$

关联规则 2：支持度 25%，置信度 98.7%，提升度 1.97

$$\{\text{年龄} \leqslant 24, \text{职业} \notin \{\text{专业人员}, \text{家庭主妇}, \text{退休}\}, \text{户主状态} \in \{\text{租房}, \text{全家同住}\}\}$$

$$\Rightarrow \{\text{婚姻状况} \in \{\text{未婚同居}, \text{单身}\}\}$$

关联规则 3：支持度 25%，置信度 95.9%，提升度 2.61

$$\{\text{户主状态} = \text{拥有}, \text{住宅类型} \neq \text{公寓}\} \Rightarrow \{\text{婚姻状况} = \text{已知}\}$$

关联规则 4：支持度 15%，置信度 95.4%，提升度 1.50

$$\{\text{户主状态} = \text{租房}, \text{住宅类型} \neq \text{独门独院}, \text{家庭人口} \leqslant 2, \text{职业} \notin \{\text{家庭主妇}, \text{学生}, \text{失业}\},$$

$$\text{收入} \in [\$20\,000, \$150\,000]\} \Rightarrow \{\text{子女数} = 0\}$$

这些特定的规则中没有大的意外，它们在很大程度上验证了我们的直觉。在其他先验信息较少的情况下，意外结果会有更多的机会出现。这些结果确实阐明广义关联规则能提供的信息类型，以及有指导学习方法与诸如 CART 或 PRIM 的规则归纳方法结合能够发现组成成分之间高关联性的项集。

这些广义关联规则如何与先前由 Apriori 算法发现的规则相比较呢？由于 Apriori 过程产生数以千计的规则，很难对它们进行比较。然而，可以得到一些通用的立论。Apriori 算法是穷举的——它找出支持度大于某个规定量的所有规则。相比之下，PRIM 是一种贪心算法，并不保证给出规则的“最优”集。另一方面，Apriori 算法只能处理哑变量，因而不能发现上面的某些规则。例如，由于住宅类型是类别型输入，每个水平一个哑变量，Apriori 算法不能够发现涉及集合

$$\text{住宅类型} \neq \text{公寓}$$

的规则。为发现这样的集合，必须将与公寓相对的住宅类型的其他类别也用哑变量编码。对所有可能感兴趣的比较都预先编码一般是不可行的。

14.3 聚类分析

聚类分析也称数据分割，具有多种目标，但都涉及把一个对象集合分组或分割为子集或“簇”（也称“聚类”），使得每个簇内部的对象之间的相关性比与其他簇中对象之间的相关性更紧密。对象可以用一组度量，或用它与其他对象的关系来刻画。另外，有时目标是把簇整理为自然的层次结构。这涉及到逐步将簇本身分组，使得在每一层，组内聚类对象之间比不同组的对象之间更为相似。

聚类分析也用于形成描述性的统计数据，以确认数据是否包括几个不同的子组，且每个组代表具有实质上不同特征的对象。后一个目标要求评估各簇对象之间的差异程度。

聚类分析所有目标的核心是待聚类的个体对象之间的相似度（或相异度）概念。聚类方法根据所给的相似度定义来尝试对对象进行分组，这取决于所关心事务的主题。该情况有些类似于预测问题中（有指导学习）指定损失函数或代价函数。与不正确预测相关的代价取决于数据之外的因素。

如图 14.4 所示，通过流行的 K-均值算法把一些模拟数据聚成了三类。该例中有两个类没有很好分开，因此“分割”比“聚类”能更精确地描述该处理的作用。K-均值聚类以对三个簇中心的猜测开始。然后循环执行下列步骤，直到收敛：

- 对每个数据点，识别最近的簇中心（基于欧氏距离）；
- 每个簇中心用离它最近的所有数据点的坐标平均值替换。

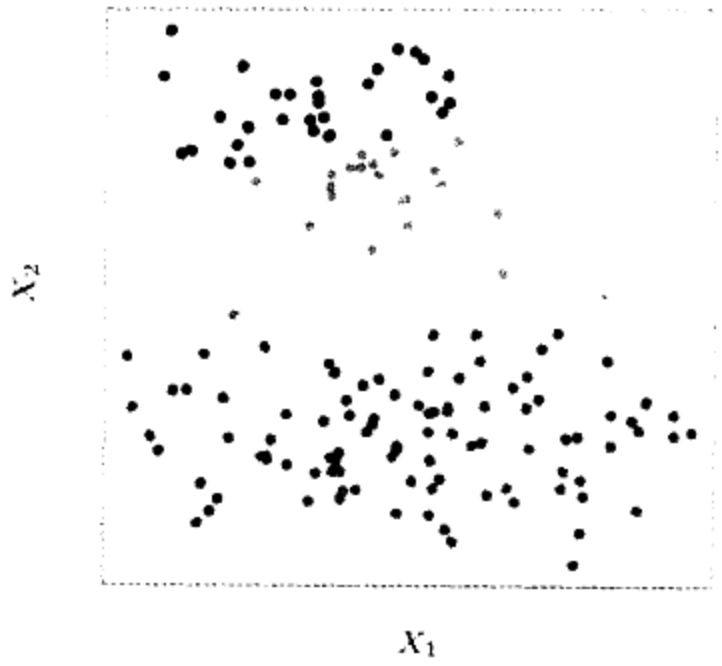


图 14.4　图中的模拟数据由 K-均值聚类算法聚类为三类（由红色、蓝色和绿色表示）（见彩页）

后面我们会更详细地讲解 K-均值聚类方法，包括如何选择簇的个数（该例中取 3）问题。K-均值聚类是一种自上而下的过程，而我们要讨论的其他聚类方法都是自下而上的。所有聚类技术的根本问题是两个对象间距离或相异度度量的选择。在讨论不同聚类算法之前，首先来讨论距离的度量。

14.3.1　邻近矩阵

有时数据直接以每对对象间的近似性（相似性或类似性）来表示。这可以是相似度（similarity）或相异度（dissimilarity）（或者称差异度或非类似度）。例如，在社会科学的实验中，要求参与者判断特定对象与其他对象有多大差别。这种数据类型可以表示成一个 $N \times N$ 的矩阵 $\mathbf{D}$，其中 N 是对象的个数，每个元素 $d_{ii'}$ 记录了第 i 个对象和第 i' 个对象之间的邻近程度。该矩阵可以提供作为聚类算法的输入。

大多数算法都假定相异度矩阵的元素非负并且对角线上元素为零：$d_{ii} = 0, i = 1,2,\cdots, N$。如果原始数据是以相似度收集的，则可以用一个单调递减函数将它们转换为相异度。大多数算法还假设相异度矩阵是对称的，所以如果原始矩阵 $\mathbf{D}$ 不对称，则必须用 $(\mathbf{D} + \mathbf{D}^T)/2$ 替换它。主观判断的相异度很少是严格意义下的距离，因为三角不等式 $d_{ii'} \leqslant d_{ik} + d_{i'k}$ 并非对所有的 $k \in \{1,\cdots,N\}$ 都成立。因此，用到距离的一些算法不能使用这样的数据。

14.3.2　基于属性的相异度

在多数情况下，我们有关于变量的度量 x_{ij}（也称属性），其中 $i = 1, 2,\cdots, N$，$j = 1, 2, \cdots, p$。由于大多数流行的聚类算法将相异度矩阵作为输入，我们就必须首先构建每对观测之

间的差异度。通常大多数情况下，定义第 j 个属性值之间差异度为 $d_j(x_{ij}, x_{i'j})$，然后定义

$$D(x_i, x_{i'}) = \sum_{j=1}^{p} d_j(x_{ij}, x_{i'j}) \tag{14.20}$$

为对象 i 和对象 i' 之间的差异度。到目前为止，最常用的选择是平方距离：

$$d_j(x_{ij}, x_{i'j}) = (x_{ij} - x_{i'j})^2 \tag{14.21}$$

然而，其他选择也是可能的，并且可能导致不同的结果。对于非定量的属性（比如，分类型数据），平方距离可能不适用。另外，有时还希望对属性赋予不同的权值，而不是像式(14.20)中那样给所有的属性赋同样的权值。

首先，我们按照属性类型讨论备选方案：

- 定量型变量(quantitative variables)。这种类型的变量或属性的度量由连续的实数值来表示。自然地，定义二者之间的“误差”为它们的差的绝对值的单调递增函数：

  $$d(x_i, x_{i'}) = l(|x_i - x_{i'}|)$$

  除了平方－误差损失 $(x_i - x_{i'})^2$ 外，另一个常见的选择是同一性（绝对误差）。前者较多强调了差异较大而不是差异较小的对象。作为选择，聚类可以基于相关性：

  $$\rho(x_i, x_{i'}) = \frac{\sum_j (x_{ij} - \bar{x}_i)(x_{i'j} - \bar{x}_{i'})}{\sqrt{\sum_j (x_{ij} - \bar{x}_i)^2 \sum_j (x_{i'j} - \bar{x}_{i'})^2}} \tag{14.22}$$

  其中，$\bar{x}_i = \sum_j x_{ij}/p$。注意，这是对变量而非对观测取平均。如果输入数据预先被标准化了，那么 $\sum_j (x_{ij} - x_{i'j})^2 \propto 2(1 - \rho(x_i, x_{i'}))$。因此，基于相关（相似度）的聚类等价于基于平方距离（相异度）的聚类。
- 序数型变量(ordinal variables)。这种类型变量的值经常表示为连续的整数，并且可实现的值将被看做一个有序的集合。例如学校的成绩（A, B, C, D, F）、偏爱的程度（不能忍受、不喜欢、不错、喜欢、特别喜欢）。秩数据是一种特殊的序数型数据。对序数型变量的误差度量一般通过用

  $$\frac{i - 1/2}{M},\ i = 1, \cdots, M \tag{14.23}$$

  以指定的原始值的顺序来替换它们的 M 个原始值。在这种标度下它们就可以被当做定量型变量处理。
- 分类型变量(categorical variables)。对于无序的分类型（也称标称型）变量，必须明确地描述每对值之间的差异程度。如果变量取 M 个不同值，这些可以通过对称的 $M \times M$ 矩阵安排，其中 $L_{rr'} = L_{r'r}$，$L_{rr} = 0$，$L_{rr'} \geqslant 0$。对所有的 $r \neq r'$，最通常的选择是 $L_{rr'} = 1$，而不相等的损失可用来强调某些误差比其他误差更重要。

14.3.3 对象相异度

下面我们定义一个过程，将 p 个单独的属性相异度 $d_j(x_{ij}, x_{i'j})$（其中 $j = 1, 2, \cdots, p$）组合为一个在两对象之间或各属性值的观测 $(x_i, x_{i'})$ 上的总体相异度 $D(x_i, x_{i'})$。该过程几乎总

是通过一种加权平均(凸组合)来实现：

$$D(x_i, x_{i'}) = \sum_{j=1}^{p} w_j \cdot d_j(x_{ij}, x_{i'j}); \quad \sum_{j=1}^{p} w_j = 1 \tag{14.24}$$

其中，w_j 是赋予第 j 个属性的权值，用以在确定对象间总体相异度时，调节变量的相对影响。该选择应该基于主题内容考虑。

重要的是应该认识到，为每个变量设置同样的权值 w_j(比如，对于任意 j，$w_j = 1$)并不是必然导致所有属性具有相同的影响。第 j 个属性 X_j 对于对象相异度 $D(x_i, x_{i'})$(14.24)的影响取决于它对数据集中所有观测对平均对象相异性度量的相对贡献

$$\bar{D} = \frac{1}{N^2} \sum_{i=1}^{N} \sum_{i'=1}^{N} D(x_i, x_{i'}) = \sum_{j=1}^{p} w_j \cdot \bar{d}_j$$

其中

$$\bar{d}_j = \frac{1}{N^2} \sum_{i=1}^{N} \sum_{i'=1}^{N} d_j(x_{ij}, x_{i'j}) \tag{14.25}$$

是第 j 个属性上的平均相异度。因此，第 j 个变量的相对影响是 $w_j \cdot \bar{d}_j$，并且在刻画对象间总体相异度时，设置 $w_j \sim 1/\bar{d}_j$ 将对所有属性赋予相等的影响。例如，如果有 p 个定量型变量，并且对每个坐标使用平方误差距离，则式(14.24)就成为 IR^p 中一对点间的(加权)平方欧氏距离：

$$D_I(x_i, x_{i'}) = \sum_{j=1}^{p} w_j \cdot (x_{ij} - x_{i'j})^2 \tag{14.26}$$

定量型变量为坐标轴。在此情况下，式(14.25)就变成：

$$\bar{d}_j = \frac{1}{N^2} \sum_{i=1}^{N} \sum_{i'=1}^{N} (x_{ij} - x_{i'j})^2 = 2 \cdot \mathrm{var}_j \tag{14.27}$$

其中 var_j 是 $\mathrm{Var}(X_j)$的样本估计。这样，每个变量的相对重要性与它在数据集上的方差成正比。一般来说，如果不考虑类型因素，对所有的属性设置，令 $w_j = 1/\bar{d}_j$ 将导致每个属性对所有对象对$(x_i, x_{i'})$间总体相异度产生同样的影响。尽管这样也许是合理的，并且经常推荐使用，但它很可能达不到预期目标。如果目标是将数据分割成相似对象的组，所有属性在对象间的相异度的概念(依赖于问题)上可能具有不同的贡献。在问题域的背景下，某些属性值差异可能更反映实际对象的差异。

如果目标是揭示数据的自然分组，某些属性可能比其他属性更能展示分组的趋势。在定义对象的相异度时，应该为在分组中较为相关的变量值赋予较高的影响因子。在这种情况下，对所有属性赋予相等的影响，将会掩盖分组而使聚类算法不能揭示它们。图 14.5 显示了一个例子。

尽管选择个别属性相异度 $d_j(x_{ij}, x_{i'j})$ 及其权值 w_j 的规则简单通用、令人鼓舞，但是它不能代替对每个个别问题背景下的仔细考虑。就是否能获得聚类的成效而言，指定适当的相异度度量比选择聚类算法更为重要。由于相异度的指定依赖于专业领域的知识，而不太适合一般研究，因此与聚类算法本身相比，这个问题在聚类的文献很少被强调。

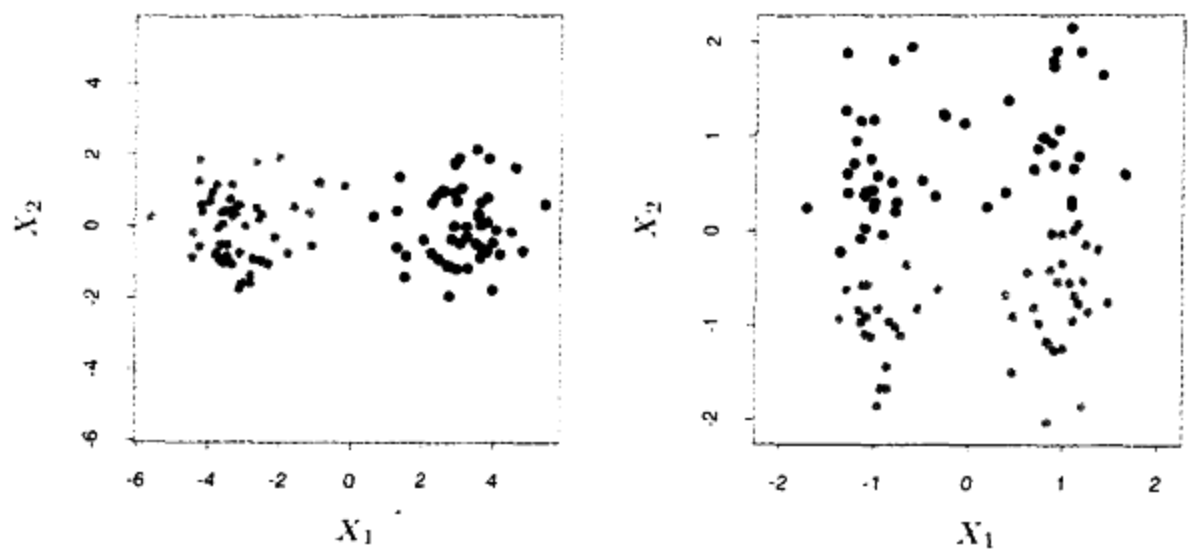


图 14.5 模拟数据:左图,对原始数据应用 K-均值聚类方法($K=2$)。用两种颜色表明簇的成员。右图,在聚类前首先将特征做了标准化。这等价于使用特征权值$1/[2\cdot \mathrm{var}(X_j)]$。标准化使原来分割不错的组间界限模糊。注意,每幅图的横坐标和纵坐标使用相同的坐标单位(见彩页)

最后,观测在一个或多个属性上经常有遗漏值(missing values)。在相异度计算(14.24)中,处理遗漏值最通用的方法是在计算观测 x_i 和 $x_{i'}$ 之间的相异度时,忽略至少有一个遗漏值的观测对 $x_{ij}, x_{i'j}$。但是当两个观测没有共同度量值时,这种方法可能失败。此时,两个观测都可能从分析中删除。另一种可选方法是,可以用每个属性未遗漏数据的均值或中值填补遗漏值。对于分类型变量,如果它们都在同一变量上有遗漏值,将两个对象当做相似对象考虑是合理的,我们可以将遗漏值当做另一种类别值来考虑。

14.3.4 聚类算法

聚类分析的目的是将观测分割成组("簇"或"聚类"),使分到同一簇的每对观测间的相异度趋向于要比不同簇中观测的相异度小。聚类算法分为三种类型——组合算法、混合建模和众数搜索。

组合算法(combinatorial algorithms)直接在观测上处理,不直接涉及潜在的概率模型。混合建模(mixture modeling)假定数据来自某一个概率密度函数描述的总体的独立同分布样本。该密度函数由带参数的模型刻画,可以看做各支密度函数的混合,每个支密度函数描述其中一个簇。这个模型通过极大似然或对应的贝叶斯方法拟合数据。众数搜索(mode seekers)("凸点搜索")以一种非参数的观点,尝试直接估计概率密度函数的不同众数。距离各个众数"最近"的观测定义一个单独的簇。

混合建模已在第 6.8 节中讨论。在第 9.3 节和第 14.2.5 节中讨论的 PRIM 算法是众数搜索或"凸点搜索"的一个例子。下面,我们来讨论组合算法。

14.3.5 组合算法

最流行的聚类算法直接将每个观测指派到一个组或簇,而不考虑描述数据的概率模型。每个观测由整数 $i\in\{1,\cdots, N\}$ 惟一标记。假定预先设定簇数 $K<N$,并且每个簇由整数 $k\in\{1,\cdots,K\}$ 标记。每个观测被指派到一个且仅指派到一个簇。这种指派工作可以用一个多对一的映射,或者用编码器 $k=C(i)$ 来刻画表示将第 i 个观测指派到第 k 个簇。基于每对观测间的相异度 $d(x_i, x_{i'})$,我们来求解实现要求目标的特定的编码器 $C^*(i)$(详述见下)。如上

所述，相异度由用户指定。通常，通过为每个观测 i 给定其相应的值(簇指派)明确地刻画编码器 $C(i)$。这样，过程的“参数”是对 N 个观测中每一个观测的簇指派。这些参数将被调整，使得表征聚类目标未达到程度的“损失”函数被极小化。

一种方法是直接指定一种数学损失函数，并且尝试通过某种组合最优算法将它极小化。由于目标是将相近的点指派到相同的簇，一种自然的损失(或“能量”)函数将是：

$$W(C)=\frac{1}{2}\sum_{k=1}^{K}\sum_{C(i)=k}\sum_{C(i')=k}d(x_i,x_{i'}) \tag{14.28}$$

这个准则表征了被赋予相同类的观测趋向相互间接近的程度。有时这被称做“簇内”点散布，由于

$$T=\frac{1}{2}\sum_{i=1}^{N}\sum_{i'=1}^{N}d_{ii'}=\frac{1}{2}\sum_{k=1}^{K}\sum_{C(i)=k}\left(\sum_{C(i')=k}d_{ii'}+\sum_{C(i')\neq k}d_{ii'}\right)$$

或

$$T=W(C)+B(C)$$

其中，$d_{ii'}=d(x_i,x_{i'})$。这里，T 是总的点散布，它是给定数据的一个常量，不依赖于簇的指派。量

$$B(C)=\frac{1}{2}\sum_{k=1}^{K}\sum_{C(i)=k}\sum_{C(i')\neq k}d_{ii'} \tag{14.29}$$

是簇间点散布。当观测被指派到相离很远的不同簇时，它趋向于比较大。因此，我们有：

$$W(C)=T-B(C)$$

并且极小化 $W(C)$等价于极大化 $B(C)$。

通过组合优化的聚类分析原则上是直截了当的。在 N 个数据点到 K 个簇的所有可能的指派上，我们可以简单极小化 W 或等价地极大化 B。遗憾的是，这种通过完全枚举的优化只对非常小的数据集适用。不同指派的个数是(Jain 和 Dubes，1988)

$$S(N,K)=\frac{1}{K!}\sum_{k=1}^{K}(-1)^{K-k}\binom{K}{k}k^N \tag{14.30}$$

例如，$S(10,4)=34\ 105$，这是完全可行的。但是，随着参数值的增加，$S(N,K)$增长得很快。$S(19,4)\approx 10^{10}$，并且大多数聚类问题涉及比 $N=19$ 更大的数据集。由于这种原因，实际可行的聚类算法只能检查所有可能编码器 $k=C(i)$中的一小部分。目标是识别一个小的子集，它可能包括最优，或者至少是一个好的局部最优划分。

这种可行策略基于迭代的贪心下降。指定一个初始划分。在每次迭代，以这样的方法更新簇的指派，使得准则的值是其先前值的改进。这种类型的聚类算法因每次迭代更改簇指派的规定不同而相异。当其规定不能够提供改进时，算法终止，并将当前的簇指派作为算法的解。由于在任何一步迭代中观测的类指派是一个对上一步迭代的变动，因此只检查所有可能指派中非常小的一部分(14.30)。然而，这些算法只收敛到局部最优，与全局最优相比它很可能是局部最优的。

14.3.6 K-均值

K-均值算法是一种主流的迭代下降聚类方法。它专门用于所有变量都是定量类型的情况，并采用平方欧氏距离

$$d(x_i, x_{i'}) = \sum_{j=1}^{p}(x_{ij} - x_{i'j})^2 = ||x_i - x_{i'}||^2$$

作为相似度度量。注意，可以通过重新定义 x_{ij} 值使用加权的欧氏距离(见习题 14.1)。

点内散布(14.28)可以写做：

$$\begin{aligned} W(C) &= \frac{1}{2}\sum_{k=1}^{K}\sum_{C(i)=k}\sum_{C(i')=k}||x_i - x_{i'}||^2 \\ &= \sum_{k=1}^{K}N_k\sum_{C(i)=k}||x_i - \bar{x}_k||^2 \end{aligned} \tag{14.31}$$

其中 $\bar{x}_k = (\bar{x}_1 k, \cdots, \bar{x}_p k)$ 是与第 k 个簇相关联的均值向量。并且 $N_k = \sum_{i=1}^{N} I(C(i) = k)$。这样，该准则的极小化通过如下方法实现：将 N 个观测指派到 K 个簇，使得在每个簇内，观测到簇均值(由该簇中的点定义)的平均相异度最小。

求解下式的一个迭代下降算法：

$$C^* = \min_{C}\sum_{k=1}^{K}N_k\sum_{C(i)=k}||x_i - \bar{x}_k||^2$$

可以通过如下步骤得到。注意，对于任意的观测集合 S

$$\bar{x}_S = \underset{m}{\operatorname{argmin}}\sum_{i\in S}||x_i - m||^2 \tag{14.32}$$

因此，我们可以通过求解放大的优化问题

$$\min_{C,\{m_k\}_1^K}\sum_{k=1}^{K}N_k\sum_{C(i)=k}||x_i - m_k||^2 \tag{14.33}$$

来获得 C^*。该极小化可以通过算法 14.1 中所给的迭代过程来实现。

算法 14.1 K-均值聚类

1. 对给定的簇指派 C，关于 $\{m_1, \cdots, m_K\}$ 对总的簇方差(14.33)极小化，产生当前指派簇的均值(14.32)
2. 给定均值的当前集合 $\{m_1, \cdots, m_K\}$，通过将每个观测指派到(当前)最近的簇均值，对(14.33)极小化。即：

$$C(i) = \underset{1\leqslant k\leqslant K}{\operatorname{argmin}}||x_i - m_k||^2 \tag{14.34}$$

3. 重复执行步骤 1 和步骤 2，直到指派不再改变

每次执行步骤 1 和步骤 2 都缩小准则(14.33)的值，因此算法一定收敛。然而，结果可能代表一个次最优的局部极小值。Hartigan 和 Wong(1979)的算法则更进了一步，并确保不存在

降低目标的单个观测从一个组到另一个组的转换。另外，我们必须用一些随机选择的不同初始均值执行算法，并且选择使目标函数取最小值的解。

图 14.6 所示为图 14.4 的模拟数据的一些 K-均值迭代步骤。形心用“O”表示。直线所示为点的划分，每一个区域是距离各形心最近的点的集合。这些划分叫做 Voronoi 格(Voronoi tessellation)。过程在 20 次迭代后收敛。

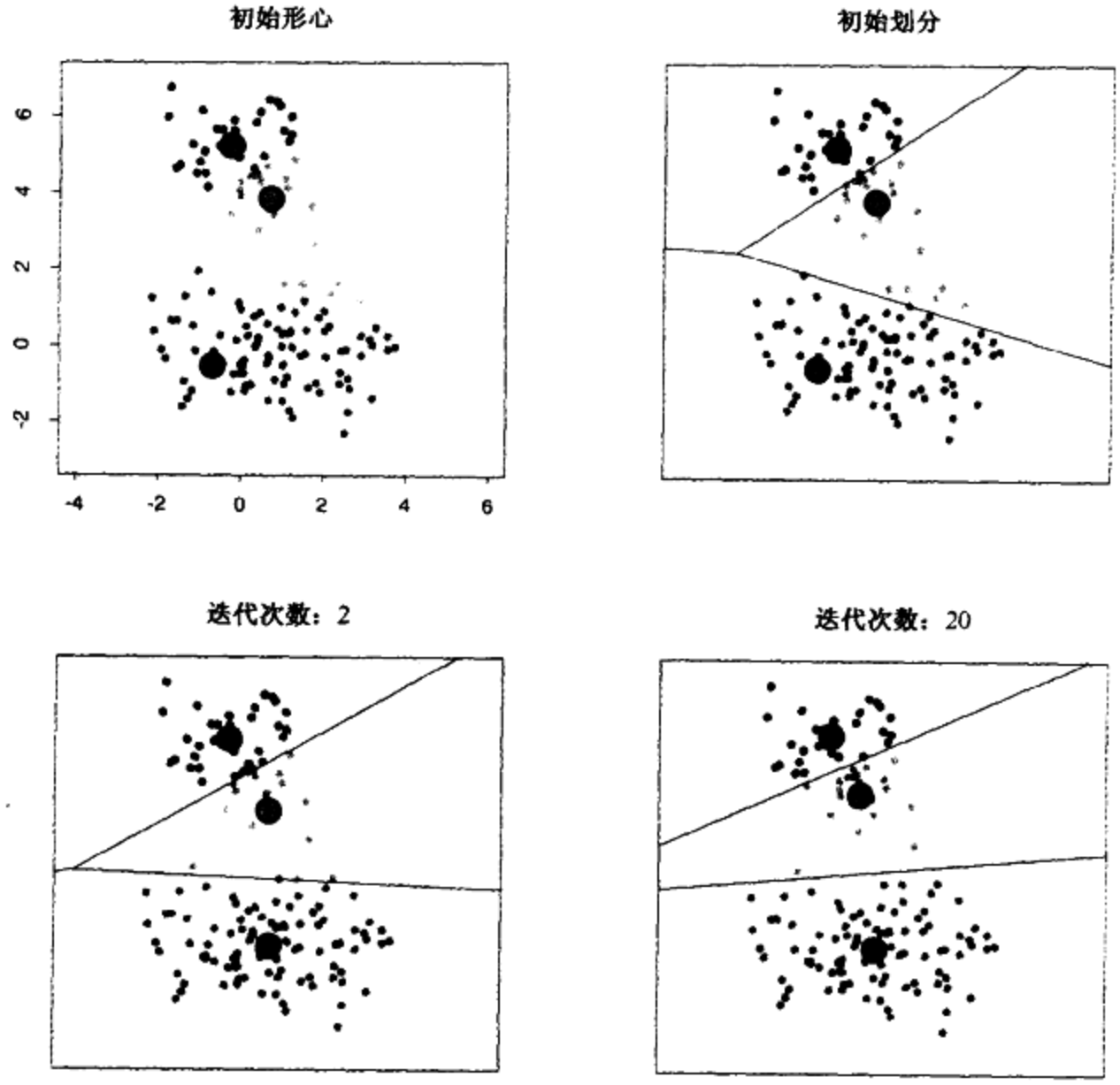


图 14.6　对图 14.4 模拟数据，K-均值聚类算法的相继迭代(见彩页)

14.3.7　作为软 K-均值聚类的高斯混合模型

K-均值聚类过程与估计特定高斯混合模型的 EM 算法相关甚密(见第 6.8 节和第 8.5.1 节)。EM 算法的 E 步根据每个数据点在每个混合分量下的相对密度为其指派“响应度”，而 M 步则根据当前的响应度重新计算支密度参数。假定我们指定 K 个混合分量，而且每个分量都是具有标量协方差矩阵为 $\sigma^2\mathbf{I}$ 的高斯密度。那么，每个分量下的相对密度是数据点与混合中心之间欧氏距离的单调函数。因此，在这样的设置中 EM 是一种 K-均值聚类的“软化”版本，它做的是点到簇中心的概率(而非确定)指派。当协方差 $\sigma^2 \to 0$ 时，这些概率取 0 和 1，两种方法一致。细节在习题 14.2 给出。图 14.7 对实线上的两个簇显示了该结果。

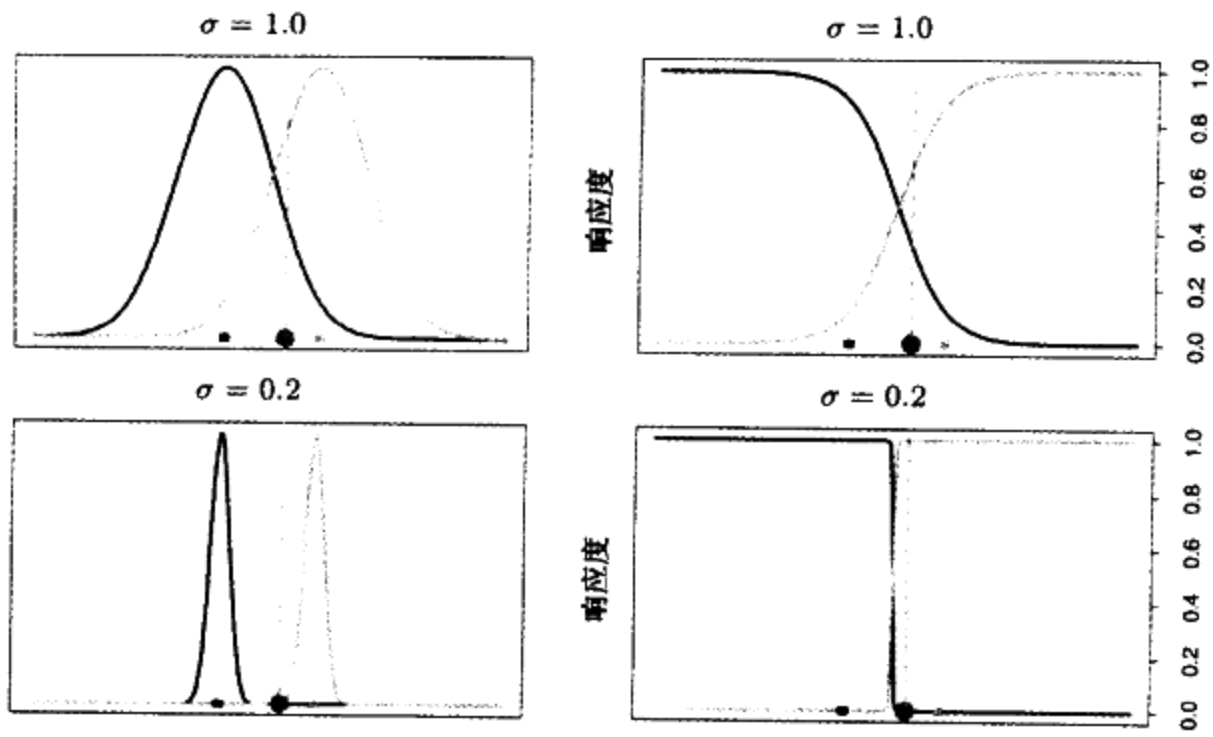


图 14.7 左图:实线是两个高斯密度$g_0(x)$和$g_1(x)$(蓝色和橙色),$x=0.5$处为单个数据点(绿色点)。彩色方块绘制在 $x=-1.0$ 和 $x=1.0$ 上,即在每个密度的均值上。右图:相对密度$g_0(x)/(g_0(x)+g_1(x))$和$g_1(x)/(g_0(x)+g_1(x))$,称做该数据点对每个簇的"响应度"。上图中高斯标准差$\sigma=1.0$,下图中$\sigma=0.2$。EM算法使用这些"响应度"做每个数据点到两个簇中每一个的"软"指派。当 σ 相当大时,响应度可能接近 0.5(右上角的图中响应度为 0.36 和 0.64)。当 $\sigma\to 0$ 时,对于离目标点最近的簇中心,响应度→1,对于其他簇,响应度为 0。右下图中所示为"硬"指派(见彩页)

14.3.8 例:人体肿瘤的微阵列数据

我们把 K-均值聚类应用于第 1 章讲过的人体肿瘤微阵列数据。这是一个高维聚类的实例。数据是一个 6830×64 的实数矩阵,每个元素描述一个基因(行)和一个样本(列)的表达度量。这里,聚类这些样本,每个样本是一个长度为 6830 的向量,对应 6830 个基因的表达值。每个样本有一个标记,如 breast(代表乳腺癌)、melanoma 等;聚类时我们不用这些标记,但聚类后会检查哪个标记落入哪个簇中。

以 K 从 1 到 10 来应用 K-均值聚类,计算每次聚类的簇内平方和,如图 14.8 所示。典型地,我们在平方和(或其对数)曲线中找一个纽结,以定位聚类的最优化个数(见第 14.3.11 节)。这里没有清晰的标志:为举例说明我们取 $K=3$,给定 3 个簇,如表 14.2 所示。

我们看到该过程成功聚集了相同癌症的样本。实际上,后来发现第二个簇中的两例乳腺癌是误诊,是维亚稳型的黑素瘤。然而 K-均值聚类在该应用中也有缺点。例如,它没有给出簇内对象的线性排序:以上只简单地按字母序排列。还有,随着簇个数 K 的变化,簇成员以随机的方式改变。也就是说,如果有 4 个簇,簇不一定嵌套在上面的 3 个簇中。因此,对这种应用来说,分层聚类算法(将在后面讲解)也许更可取。

表 14.2　人体肿瘤数据:在 K-均值聚类得到的 3 个簇中,每个类型癌症实例的个数

簇	Breast	CNS	Colon	K562	Leukemia	MCF7
1	3	5	0	0	0	0
2	2	0	0	2	6	2
3	2	0	7	0	0	0
簇	Melanoma	NSCLC	Ovarian	Prostate	Renal	Unknown
1	1	7	6	2	9	1
2	7	2	0	0	0	0
3	0	0	0	0	0	0

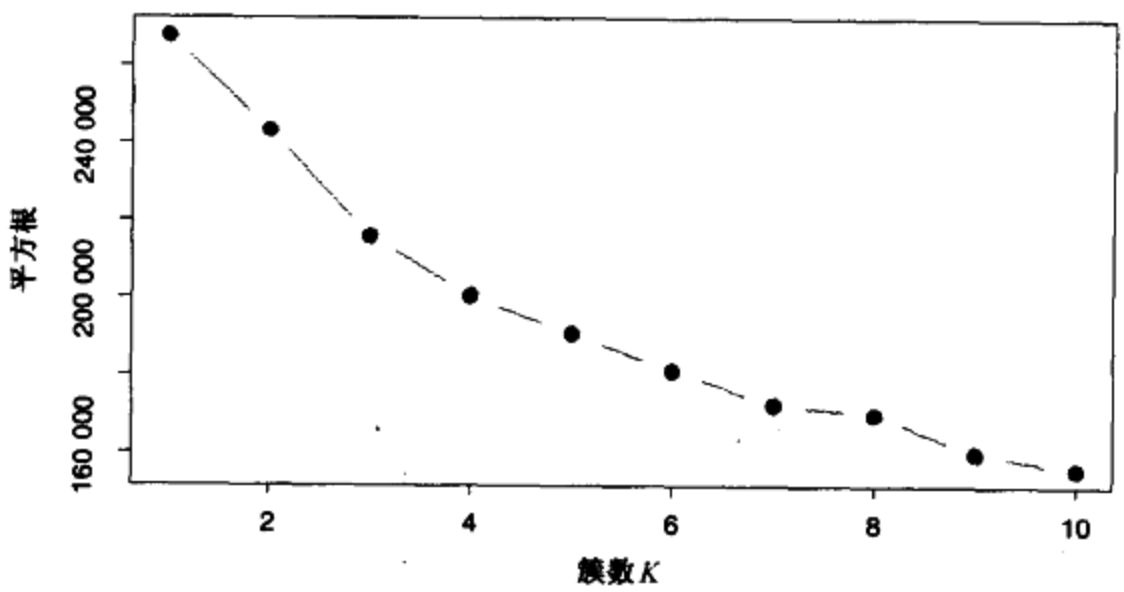


图 14.8　对人体肿瘤微阵列数据应用 K-均值聚类的簇内平方和

14.3.9　向量量化

K-均值聚类算法为与此明显不相关的图像和信号压缩领域提供了一个重要工具,特别是在向量量化(vector quantization)或 VQ(Gersho 和 Gray, 1992)方面。在图 14.9[①] 中,左图是著名统计学家 Ronald Fisher 先生的一张经数字化处理的照片。该图片由 1024 × 1024 个像素点构成,其中每个像素点的灰度值范围是 0 到 255,因此每个像素点需要 8 位存储。整幅图片占 1 兆的存储空间。中间的图片是左边图片的一个 VQ 压缩版,需要 0.239 兆存储(质量上有一些损失)。右图是更进一步的压缩,只需要 0.0625 兆存储(质量上有相当大的损失)。

这里实现的 VQ 方案首先将图片分割为小块,该例中每块有 2×2 个像素点。512 × 512 个 4 位数的块,每一个被视为 $\mathbb{R}^4$ 中的一个向量。K-均值聚类算法(在此情况下也称 Lloyd 算法)在该空间上运行。中图取 $K = 200$,右图取 $K = 4$。512 × 512 像素块(或点)的每一个用离它最近的簇形心(称

① 该例由 Maya Gupta 制作。

图 14.9 Ronald A. Fisher 先生(1890～1962)是现代统计学的奠基人之一,我们把极大似然、充分性和其他许多基础概念归功于他。左图是一张1024×1024的灰度图,每个像素点占8位。中图是2×2块VQ的结果,使用200个编码向量,压缩率为1.9位每像素。右图只用了4个编码向量,压缩率为0.50位每像素

为编码字)近似。聚类过程称做编码(encoding)步,而形心的集合称做编码本(codebook)。

为表示近似的图像,需要为每个块提供近似它的编码本条目标识。这要求每块 $\log_2(K)$ 位。还应该提供编码本本身,它是 $K=4$ 个实数(典型地可以忽略)。大体上,压缩图像的存储总量为原图像的 $\log_2(K)/(4\times 8)$($K=200$ 时为 0.239,$K=4$ 时为 0.063)。典型地,用位每像素点表示压缩率:$\log_2(K)/4$,则分别为 1.91 和 0.50。从形心构建近似图像的过程称做解码(decoding)步。

我们为什么期望 VQ 有成效呢? 原因是对于普通的日常图片(比如照片),许多小块看起来是一样的。在这种情况下,有许多几乎是纯白色的块,还有不同阴影的相似的纯灰色块。每种相似的块只需要用一个块来表示,然后用多个指针指向它即可。

由于图片是原始图像的退化版本,我们讨论的方法通常也称有损(lossy)压缩。退化或变形一般以均方误差来度量。该例中,$K=200$ 时,$D=0.89$,而 $K=4$ 时,$D=16.95$。更一般地,比率/变形曲线将用于评估权衡。也可以用块聚类实现无损(lossless)压缩,并且仍然利用重复的模式。如果取原始图片并对它进行无损压缩,最好使每个像素点占 4.48 个位。

上面,我们提出编码本中 K 个编码字,每个需要 $\log_2(K)$ 位来识别。这用的是定长编码,如果图像中一些编码字要比其他编码字出现的次数多,这种方法就不是有效的。利用香农编码理论,我们知道通常变长编码更好一些,这样比率变为 $-\sum_{\ell=1}^{K} p_\ell \log_2(p_\ell)/4$。分子中的项是图像中编码字分布 p_ℓ 的熵。使用变长编码,编码率分别下降到 1.42 和 0.39。目前已经开发了许多 VQ 的推广版本;例如,树结构 VQ 使用自上而下的 2-均值风格算法找形心,将在第 14.3.12 节提到。这允许对压缩逐步求精。进一步的细节可以在 Gersho 和 Gray 的著作(1992)中找到。

14.3.10 K-中心点

如上所述,当相异度采用平方欧氏距离 $D(x_i, x_{i'})$(14.74)时,K-均值算法是适用的。这要求所有的变量均为定量类型。另外,使用平方欧氏距离对最远的距离施加了最高的影响。这会导致过程对产生很大距离的孤立点的处理缺少健壮性。该限制可以取消,代价是增加计算开销。

在 K-均值算法中,仅有的采用平方欧氏距离的部分是极小化步骤(14.32);在式(14.33)

中簇代表$\{m_1,\cdots,m_K\}$被当做当前指派簇的均值。用显式优化式(14.33)中的$\{m_1,\cdots,m_K\}$替换这一步,可以推广算法,使用任意定义的相异度$D(x_i,x_{i'})$。在最常见的形式中,每个簇中心限制为所指派类中的一个观测,如算法 14.2 中的概述。该算法假设属性数据,但是它也只能应用于以邻近矩阵(见第 14.3.1 节)描述的数据。我们没有必要显式地计算簇中心,而只需要一直记录下标i_k^*。

算法 14.2　K-中心点聚类

1. 对给定的簇指派 C,找出簇中的观测,它到该簇其他点的总距离最小:

$$i_k^* = \underset{\{i:C(i)=k\}}{\operatorname{argmin}} \sum_{C(i')=k} D(x_i,x_{i'}) \tag{14.35}$$

则$m_k = x_{i_k^*}$,$k=1,2,\cdots,K$是簇中心的当前估计

2. 给定簇中心的当前集合$\{m_1,\cdots,m_K\}$,通过将每个观测指派到(当前)最近的簇中心

$$C(i) = \underset{1\leqslant k\leqslant K}{\operatorname{argmin}} D(x_i,m_k) \tag{14.36}$$

来极小化总体误差

3. 重复执行步骤 1 和步骤 2,直到指派不再改变

对每一个临时的簇 k,求解(14.32)所需的计算量与指派给它的观测个数成正例,然而为求解(14.35),计算量增至$O(N_k^2)$。和以前一样,给定簇"中心"的一个集合$\{i_1,\cdots,i_K\}$,得到新的指派:

$$C(i) = \underset{1\leqslant k\leqslant K}{\operatorname{argmin}} d_{ii_k^*} \tag{14.37}$$

所需要的计算量与$K\cdot N$成比例。因此,K-中心点比 K-均值的计算密集很多。

交替式(14.35)和式(14.37)代表着一种特别的启发式搜索策略,为求解:

$$\min_{C,\{i_k\}_1^K} \sum_{k=1}^{K} \sum_{C(i)=k} d_{ii_k} \tag{14.38}$$

Kaufman 和 Rousseeuw(1990)提出另一个可选的直接求解策略:对于每个中心i_k,用非当前中心的观测与之交换,选取使得准则(14.38)减少最大的交换。重复该过程,直至找不到更有优势的交换。Massart 等人(1983)推出一种分支 - 绑定(branch -and -bound)求解式(14.38)全局极小的组合方法,它仅对非常小的数据集是可行的。

例:国家相异度

本例取自 Kaufman 和 Rousseeuw(1990)的著作,源自要求政治学专业的学生提供 12 个国家每对之间相异度的研究。这 12 个国家是比利时、巴西、智利、古巴、埃及、法国、印度、美国、以色列、前苏联、南斯拉夫和扎伊尔。表 14.3 给出了相异度的平均值。我们应用 3 - 中心点来聚类这些相异度。注意,K - 均值聚类在此不适用,因为这里只有距离而没有原始观测。在图 14.10 中,左图显示依据 3 - 中心点聚类重新排序和分组的相异度;右图是一个二维的多维标量图,由不同的颜色表示 3 - 中心点聚类的指派(多维标量在第 14.7 节讨论)。两幅图都显示了簇的良好分割,但是 MDS 指出"埃及"落入两个簇的中间。

表 14.3 源自政治学调查的数据:值为每对国家相异度的平均,来自对政治学专业学生的有关调查问卷

	BEL	BRA	CHI	CUB	EGY	FRA	IND	ISR	USA	USS	YUG
BRA	5.58										
CHI	7.00	6.50									
CUB	7.08	7.00	3.83								
EGY	4.83	5.08	8.17	5.83							
FRA	2.17	5.75	6.67	6.92	4.92						
IND	6.42	5.00	5.58	6.00	4.67	6.42					
ISR	3.42	5.50	6.42	6.42	5.00	3.92	6.17				
USA	2.50	4.92	6.25	7.33	4.50	2.25	6.33	2.75			
USS	6.08	6.67	4.25	2.67	6.00	6.17	6.17	6.92	6.17		
YUG	5.25	6.83	4.50	3.75	5.75	5.42	6.08	5.83	6.67	3.67	
ZAI	4.75	3.00	6.08	6.67	5.00	5.58	4.83	6.17	5.67	6.50	6.92

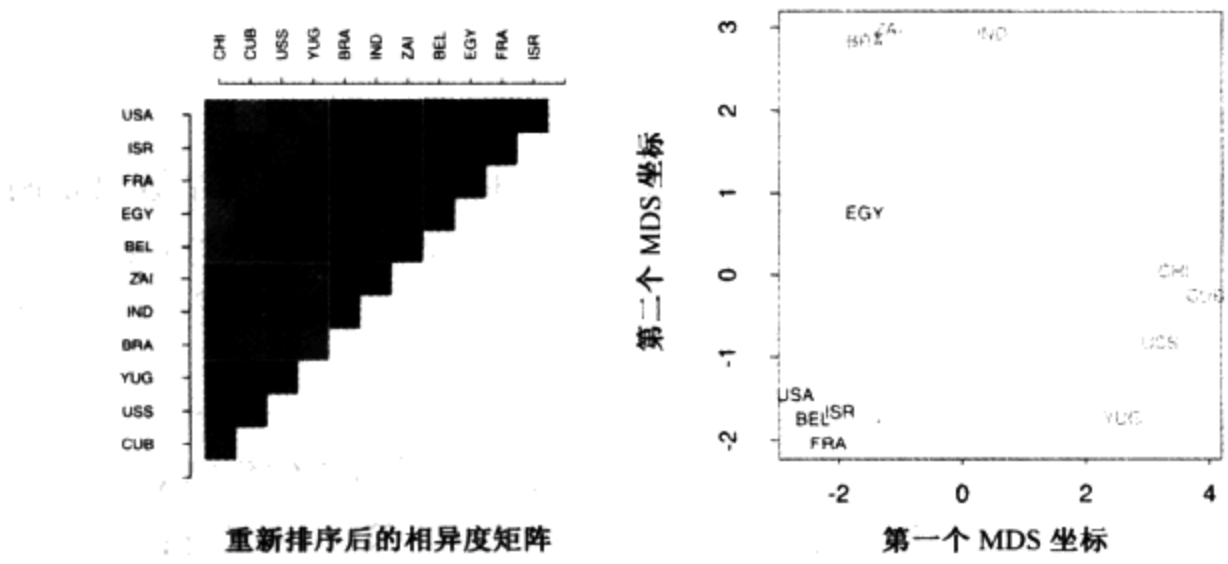


图 14.10 国家相异度的调查。左图:根据 3 - 中心点聚类排序并分组的相异度。热度图由最相似(深红色)到最不相似(浅红色)进行编码。右图:二维的多维定标图,3种颜色表示3- 中心点聚类的簇(见彩页)

14.3.11 实践问题

为了应用K-均值或K-中心点,必须选择簇的个数 K^* 并做初始化。后者可以通过指定中心$\{m_1,\cdots,m_K\}$或$\{i_1,\cdots,i_K\}$的初始集合或者初始编码器 $C(i)$来定义。通常,指定中心更方便些。建议包括从简单的随机选择到基于逐步前向指派的专门策略。给定先前步骤选取的中心 $i_1,\cdots,i_{k-1}$,每一步选择一个新的中心 i_k,它极小化准则(14.33)或(14.38)。该过程进行 K 步,产生 K 个初始中心,由此开始优化算法。

簇个数 K 的选择取决于目标,因为数据分割的个数 K 一般是问题定义的一部分。例如,一家公司雇用了 K 个销售人员,目标是要将客户数据库划分为 K 个部分,每一部分分给一个销售人员,使指派给一个销售人员的客户尽可能地相似。然而,聚类分析经常用于提供描述性统计数据,以确定构成数据库的观测属于不同自然组的程度。这里组的个数 K^* 是未知的,而且正如分组本身一样,用户要求从数据中估计。

典型地，估计 K^* 的基于数据的方法把类内相异度 W_K 看做簇个数 K 的函数。对于 $K\in\{1,2,\cdots,K_{max}\}$ 分别得到解。相应的值 $\{W_1,W_2,\cdots,W_{Kmax}\}$ 通常随着 K 的增加而减少。当准则在独立的检验集上评估时情况就是这样，这是由于大量簇中心将倾向于紧凑地充斥特征空间，因而离所有数据点都很近。这样，在有指导学习中对模型选择非常有用的交叉校验技术在这里却不能用。

这种方法的直观依据是，如果实际上观测有 K^* 个不同分组（根据相异度定义），则对于 $K<K^*$，由算法得到的每个簇都将包含基本组的一个子集。也就是说，该解决方案不会把出现在同一自然组中的观测分派到不同的估计簇中。就该例来说，随着指定簇数不断增加，解的准则值将趋向于显著地降低，$W_{K+1}\ll W_K$，因为自然组被不断分派到分割的簇中。对于 $K>K^*$，则至少有一个估计簇将一个自然组分成两个子组。这样，随着 K 的进一步增加，准则值的增加将有减少的趋势。与适当地划分两个良好分割的组的并集相比，分隔一个组内的观测相当近的自然组对标准的降低较少。

就该情况的实现来说，相继的不同准则值之差 W_K-W_{K+1} 将在 $K=K^*$ 上锐减。即 $\{W_K-W_{K+1}\mid K<K^*\}\ll\{W_K-W_{K+1}\mid K\geqslant K^*\}$。通过在 W_K 的图（作为 K 的函数）中识别“纽结”，可以得到对 K^* 的一个估计 $\hat{K}^*$。与聚类过程的其他方面一样，该方法也具有一定的启发式特点。

最近提出的间隙统计（Gap statistic）（Tibshirani 等人，2001）比较了曲线 $\log W_K$ 与由矩形区域上均匀分布的数据中得到的曲线。它估计簇的最优个数，使两曲线的间隙最大。本质上，这是一种自动定位上述“纽结”的方法。当数据分成单独一个簇时该方法也能工作得相当好，并且在这种情况下，它倾向于将最优的簇数估计为 1。这正是令大多数其他有竞争方法失效的情况。

图 14.11 所示的是对图 14.4 的模拟数据应用间隙统计的结果。左图对于 $K=1,2,\cdots,8$ 个簇显示 $\log W_K$（绿色曲线）和均匀分布的 20 个模拟数据上 $\log W_K$ 的期望值（蓝色曲线）。右图所示为间隙曲线，它是期望曲线与观测曲线之差。图中所示还有半宽度 $s'_K=s_K\sqrt{1+1/20}$ 的误差棒，其中 s_K 是 $\log W_K$ 在 20 个模拟数据上的标准差。当 $K=2$ 个簇时，间隙曲线是极小化的。如果 $G(K)$ 是 K 个簇的间隙曲线，估计 K^* 的形式化规则是：

$$K^*=\underset{K}{\operatorname{argmin}}\{K\mid G(K)\geqslant G(K+1)-s'_{K+1}\} \tag{14.39}$$

这里给定 $K^*=2$，从图 14.4 来看它是合理的。

14.3.12　分层聚类

应用 K-均值或 K-中心点聚类算法的结果取决于要搜索的簇数的选择和初始格局的设定。相比之下，分层聚类方法不要求这些规定。作为替代，它们要求用户根据两组间（不相交）每对观测的相异度，指定观测组之间相异度的度量。顾名思义，分层聚类产生分层表示，每一层的簇通过合并其次低层的簇而创建。在最低层，每个簇包括一个观测。在最高一层，只有一个簇，它包括所有的数据。

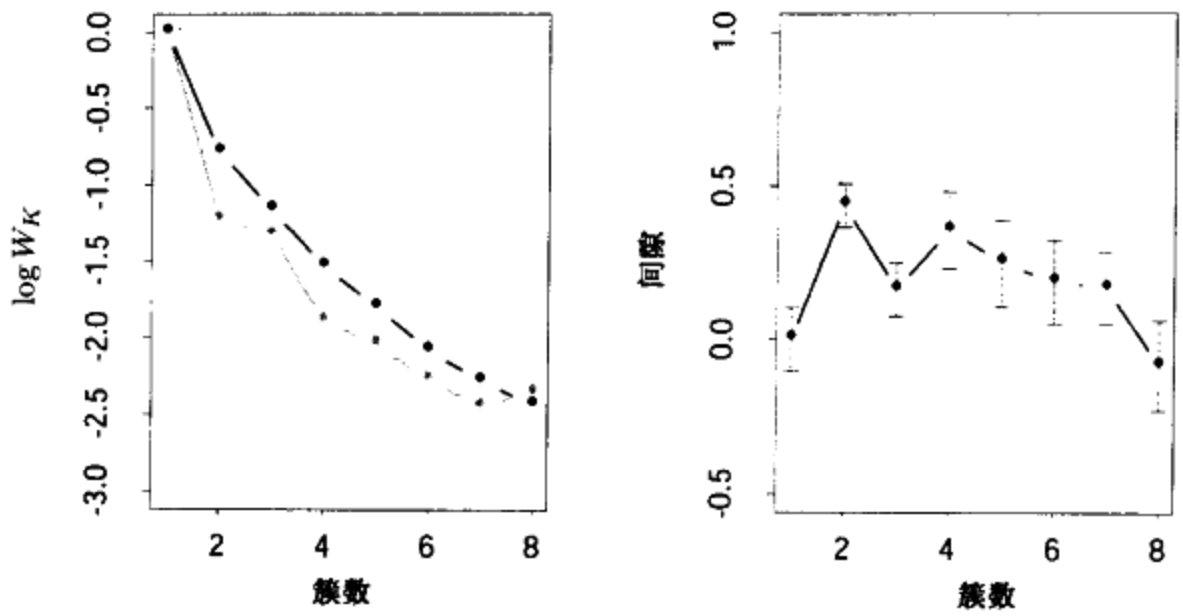


图 14.11 左图:对于图 14.4 中模拟数据,$\log W_K$ 的观测值(绿色)和期望值(蓝色)。两条曲线在1个簇时均等于0。右图:间隙曲线,等于 $\log W_K$ 的观测值和期望值之差。间隙估计 K^* 是在最大值的一个标准差内产生间隙的最小 K 值;这里 $K^* = 2$(见彩页)

分层聚类的策略有两种基本类型:凝聚的(agglomerative)(自下而上)和分裂的(divisive)(自上而下)。凝聚策略从底部开始,在每一层递归地将两个选定的簇合并为一个簇。该过程在下一较高层产生一个少一个簇的分组。选取的合并组对由具有最小组间相异度的两个组构成。分裂的策略从顶部开始,在每一层递归地将当前层中的一个簇分裂为两个新簇。选取的分裂产生具有最大组间相异度的两个新组。用这两种方法,分层结构均有 $N-1$ 层。

每一层表示把数据划分为观测不相交的簇的特定分组。整个分层结构表示分组的一个有序序列。哪一层(如果有的话)实际代表一个"自然"聚类是由用户决定的,自然聚类就是每一组内的观测比同一层中被指派到不同组的观测具有更大的相似性。前面讲的间隙统计可用于这种目的。

递归的二叉分裂/凝聚策略可以用一棵二叉树表示。树的节点表示组,根节点表示整个数据集合。N 个终端节点中,每个都表示一个单独的观测(单元素簇)。每个非终端节点("父节点")有两个子节点。对于分裂聚类来说,两个子节点表示从父节点分裂而来的两个组;对于凝聚聚类,子节点表示被合并为父节点的两个组。

所有的凝聚方法和某些分裂方法(当由自下而上的观点来看时)都具有单调性。也就是说,合并的两个类之间的相异度随着合并的层次单调递增。因此,可以画出二叉树使每个节点的高度与其两个子节点的组间相异度值成正例。终端节点表示单独的观测,高度为 0。该类型的图示称为树状图(dendrogram)。

树状图以图的形式提供分层聚类完整的高度可解释的描述。这也是分层聚类方法之所以受欢迎的主要原因之一。

对于微阵列数据,图14.12所示为使用平均连接的凝聚聚类结果的树状图;凝聚聚类和该例在本章后续部分将有更详细的讨论。在某个特定高度对树状图水平剪枝划分数据到分离的簇,由与之相交的垂直线表示。当最优组间相异度超过剪枝阈值时终止过程,所产生的就是这些簇。在高值处合并的组,相对于树中被其包含的低层子组的合并值,是自然簇的候选。注意,这可能在几个层次出现,以指出一个聚类层次结构,即嵌套在簇中的簇。

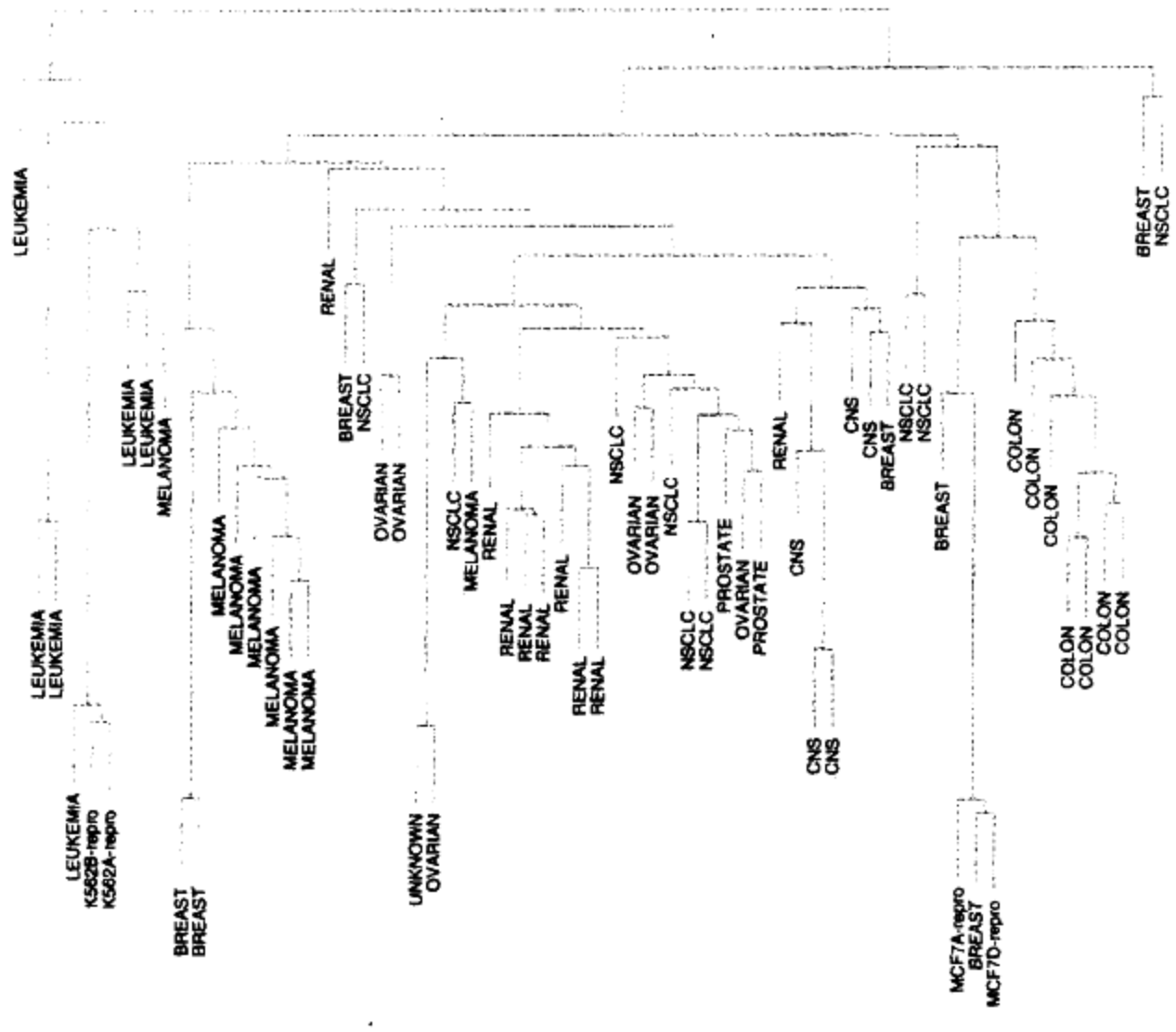


图14.12　人体肿瘤微阵列数据平均关联的凝聚聚类树状图

这样的树状图经常被看做数据本身的图解概括，而不是算法结果的描述。然而，应当谨慎对待这样的解释。首先，不同的分层方法（见下述）以及数据的微小变化，都能导致完全不同的树状图。此外，图解概括只对具有由算法产生的分层结构的逐对观测相异度区域有效。无论实际数据中是否存在分层结构，分层的方法均利用分层结构。

树状图产生的分层结构实际代表数据本身的程度可用共分类相关系数（cophenetic correlation coefficient）来判断。它是输入到算法的 $N(N-1)/2$ 对观测的相异度 $d_{ii'}$ 和由树状图得到的相关共分类相异度 $C_{ii'}$ 之间的相关性。两个观测 (i, i') 之间的共分类相异度 $C_{ii'}$ 是观测 i 和 i' 首次合并入同一组时的组间相异度。

共分类相异度是一种非常严格的相异度度量。首先，观测上的 $C_{ii'}$ 必须包含很多关联，因为 $N(N-1)/2$ 值中只有 $N-1$ 个值可能不同。而且对任何三个观测 (i, i', k)，这些相异度服从超度量不等式（ultrametric inequality）：

$$C_{ii'} \leqslant \max\{C_{ik}, C_{i'k}\} \tag{14.40}$$

作为一个几何学例子，假设数据表示为欧氏坐标空间中的点。为了使数据集中点之间距离的集合服从式（14.40），所有由三个点组成的三角形必须具有下列性质的等腰三角形——不等边长度短于两个等长边的长度（Jain 和 Dubes，1988）。因此，期望任意数据集上的一般相异度非常类似于由树状图计算而得的与其相关的共分类相异度是不现实的，特别是在不存在很多关联值的情况下。正如特定算法所做的一样，树状图应当主要看做数据聚类结构的一种描述。

凝聚聚类

凝聚聚类算法以每个观测表示一个单独的簇为开始。在 $N-1$ 步的每一步，两个最近的(相异度最小)簇合并为一个簇，在较高一层产生减少一个簇的聚类。因此，必须定义两个簇(观测组)间的相异度度量。

令 G 和 H 表示两个这样的组：G 和 H 之间的相异度 $d(G,H)$ 由每对观测的相异度 $d_{ii'}$ 集合来计算，其中一个是来自 G 中的 i，另一个是来自 H 中的 i'。单连接(single linkage，SL)凝聚聚类将组间的相异度定义为最近(最小相异)对的相异度：

$$d_{SL}(G,H)=\min_{\substack{i\in G\\ i'\in H}} d_{ii'} \tag{14.41}$$

这也经常被称做最近邻(nearest -neighbor)技术。完全连接(complete linkage，CL)凝聚聚类(最远近邻技术)，将组类间的相异度定义为最远(最大相异)对的相异度：

$$d_{CL}(G,H)=\max_{\substack{i\in G\\ i'\in H}} d_{ii'} \tag{14.42}$$

组平均(group average，GA)聚类使用组间的平均相异度：

$$d_{GA}(G,H)=\frac{1}{N_G N_H}\sum_{i\in G}\sum_{i'\in H} d_{ii'} \tag{14.43}$$

其中，N_G 和 N_H 分别是每个组中观测的个数。在凝聚聚类算法背景下，尽管已经有许多定义组间相异度的其他提案，但上述三种最常用。图 14.13 所示的是这三种定义的实例。

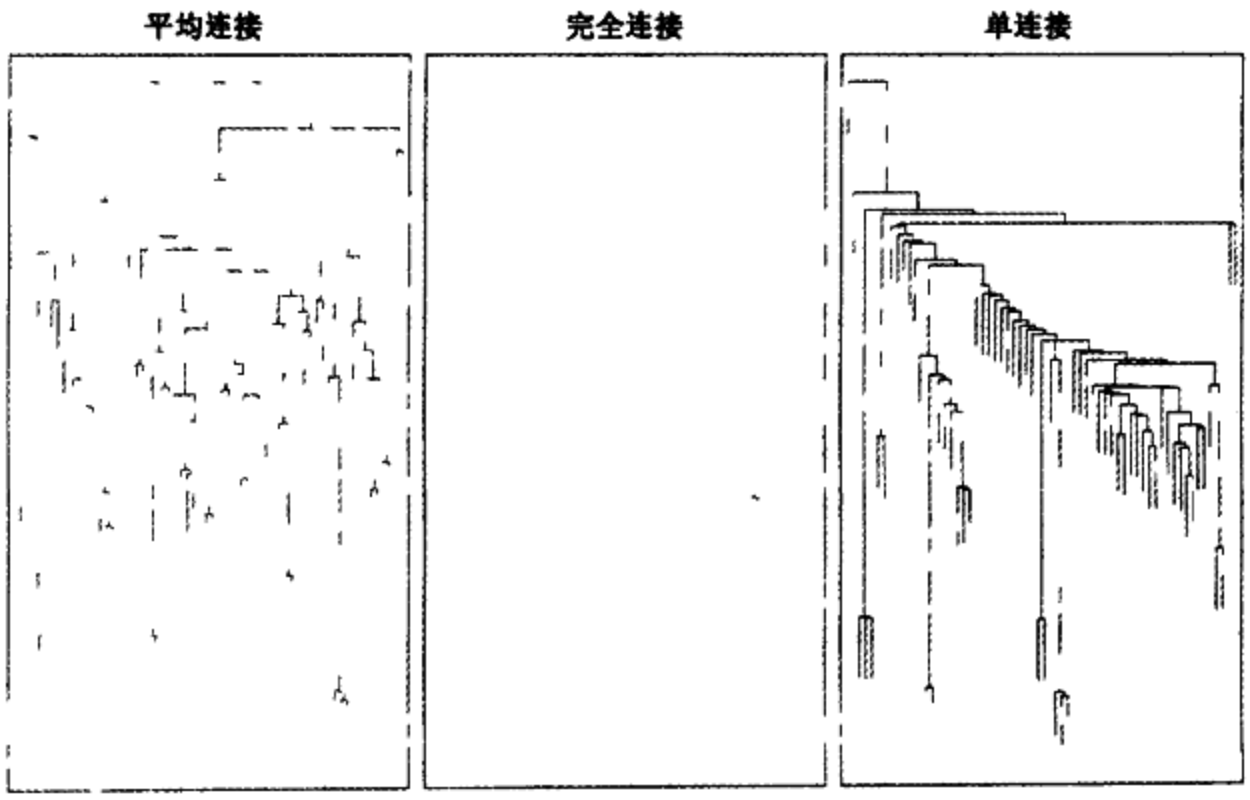


图 14.13 人体肿瘤微阵列数据的凝聚分层聚类树状图

如果数据的相异度 $\{d_{ii'}\}$ 展示强聚类趋势，每个簇是紧凑的并与其他簇完好分割，那么三种方法将产生近似的结果。簇是紧凑的，如果与不同簇观测相比，簇内所有观测都相对紧密靠拢(相异度小)。如果不是这种情况，三种方法的结果将不相同。

对于单连接(14.41)，只要单个相异度 $d_{ii'}$ (其中 $i\in G, i'\in H$)小，就认为组 G 和组 H 是

紧密靠拢的，而不管组间其他观测的相异度如何。因此，在相对低的阈值下，这倾向于合并由一系列靠近的观测连接的观测。这种现象称为链条(chainning)，它常常被认为是该方法的不足之处。由单连接产生的聚类可能破坏"紧凑性"，该特性是指在所给观测的相异度 $|d_{ii'}|$，同一簇中的所有观测比其他簇中的观测更相似。如果定义一组观测的直径 D_G 为其成员间最大的相异度：

$$D_G = \max_{\substack{i\in G \\ i'\in G}} d_{ii'} \tag{14.44}$$

则单连接可以产生具有很大直径的簇。

完全连接(14.42)代表另一个极端。两个组类 G 和 H 只有当它们的并集中的所有观测都相对近似时才被认为是靠近的。这将倾向于产生具有小直径(14.44)的紧凑簇。然而，它可能产生违背"闭合性"(closeness)的簇。也就是说，指派到一个簇的观测距其他簇成员可能比距本簇的某些成员更近。

组平均聚类(14.43)代表单连接和完全连接这两个极端的折中。它试图产生相对紧凑的簇，这些簇又相对远离。然而，它的结果取决于度量观测相异度 $d_{ii'}$ 的数值标度。应用一个严格单调增函数 $h(\cdot)$转换 $d_{ii'}$，即 $h_{ii'} = h(d_{ii'})$，则可以改变由式(14.43)产生的结果。相比之下，式(14.41)和式(14.42)只依赖于 $d_{ii'}$ 的序，因此在这样的单调变换下是不变的。该不变性经常被用做支持单连接或完全连接方法优于组平均方法的一个理由。

Kelly 和 Rice(1990)认为组平均聚类具有统计相容性，而单连接或完全连接违背这种相容性。假设我们有属性值数据 $X = (X_1, \cdots, X_p)$，并且每个簇 k 是一个来自总体联合密度 $p_k(x)$ 的随机样本。完整的数据集是来自 K 个密度混合的随机样本。组平均相异度 $d_{GA}(G, H)$ (14.43)是下式的一个估计：

$$\int\int d(x,x')\,p_G(x)\,p_H(x')\,dx\,dx' \tag{14.45}$$

其中，$d(x, x')$是属性值空间中点 x 和 x' 之间的相异度。随样本容量 N 趋向于无穷大，$d_{GA}(G, H)$(14.43)趋向于式(14.45)，这是两个密度 $p_G(x)$和 $p_H(x)$之间关系的一个特征。对于单连接，随 $N\to\infty$，$d_{SL}(G, H)$(14.41)趋向于 0，独立于 $p_G(x)$和 $p_H(x)$。对于完全连接，随 $N\to\infty$，$d_{CL}(G, H)$(14.42)变成无穷大，也独立于那两个密度。因此，并不清楚是总体分布的哪些方面被 $d_{SL}(G, H)$和 $d_{CL}(G, H)$估计。

例：人体肿瘤微阵列数据(续)

图 14.13 左图所示为微阵列数据样本(列)的平均连接凝聚聚类算法结果的树状图。中图和右图所示为完全连接和单连接的结果。平均和完全连接的结果类似，而单连接导致长而细的不平衡组。我们现在来关注平均连接聚类。

像 K 均值聚类一样，分层聚类成功地将简单肿瘤聚类。然而，它还有其他好的特性。通过在不同高度对树状图剪枝，出现不同的簇个数，并且簇的集合是嵌套的。另外，它还给出关于样本的偏序信息。在图 14.14 中，我们按照从分层聚类中导出的顺序整理了表达矩阵的基因(行)和样本(列)。

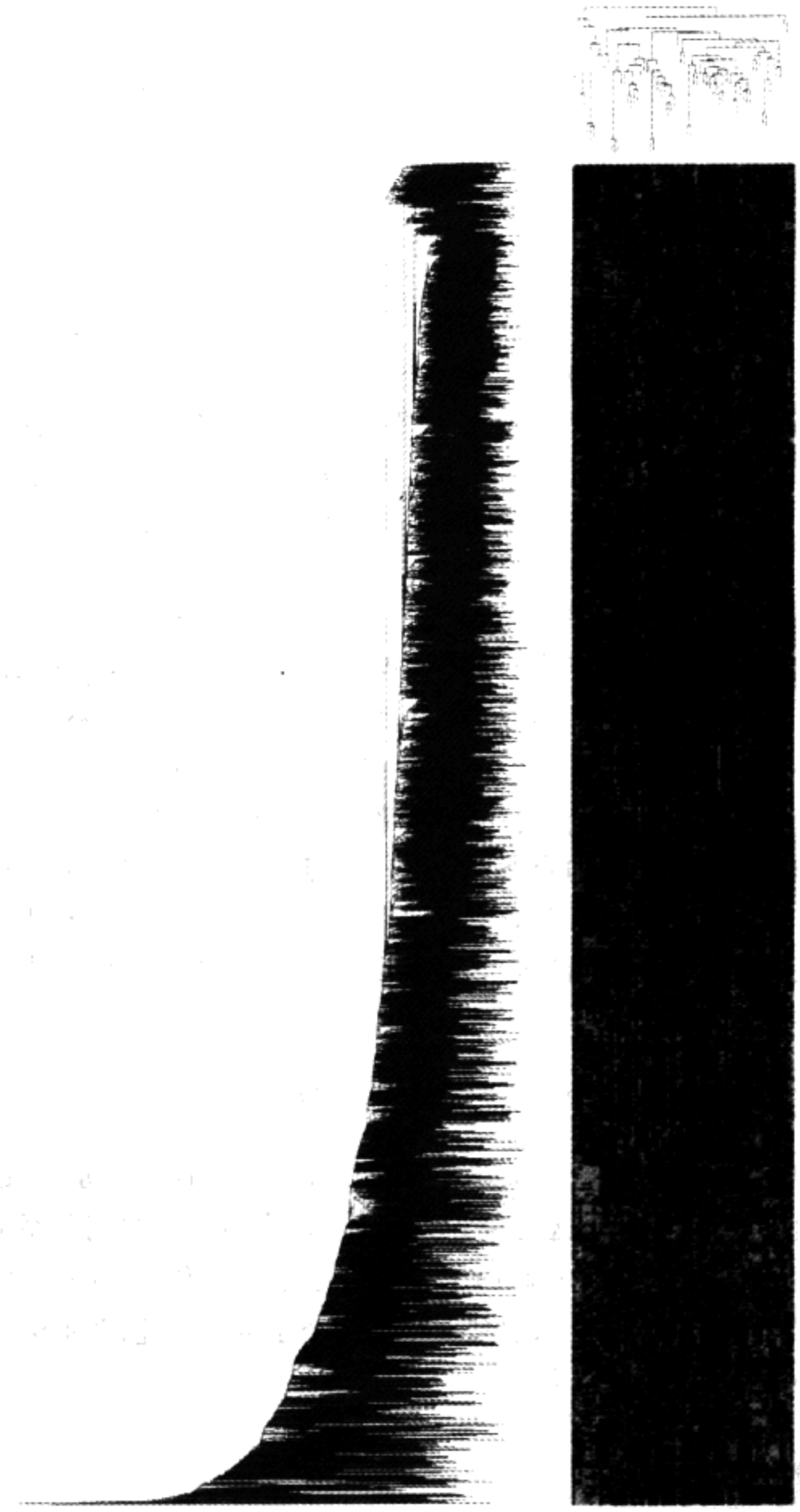


图 14.14 DNA 微阵列数据：平均连接分层聚类独立应用于行(基因)和列(样本)，以确定行和列的顺序。颜色由浅绿(阴性，低显性)变为浅红(阳性，高显性)(见彩页)

注意，如果在任一次合并中稍稍改动树状图分支的方向，结果树状图将仍然与分层聚类操作的序列一致。因此，为确定叶子的顺序，我们必须增加一个限制。为产生图 14.14 的行序，我们利用了 S-PLUS 中默认的规则：每一次合并中，具有更紧凑簇的子树置于左边(旋转的树状图中的底部)。个体基因是可能的最紧密的簇，合并涉及到两个个体基因，按照它们观测量的顺序排放。对列也用同样的规则。许多其他规则也是可能的——例如，按照基因的多维定

标排序，见第 14.7 节。

图 14.14 的两路重排产生一个基因和样本的信息图。该图比第 1 章图 1.3 中行和列的随机排序信息更丰富。此外，树状图本身也很有用，比如生物学家能够用生物学处理的方式来解释基因簇。

分裂聚类

分裂聚类算法以全部数据作为一个簇为开始，并且以自上而下的方式进行，在每次迭代中递归地将现有的一个簇分为两个子簇。在聚类算法的文献中，该方法没有像凝聚方法那样被广泛研究。在有关压缩的工程文献(Gersho 和 Gray，1992)中对该方法有过一些探讨。在聚类设置中，当我们对将数据划分为相对少的簇感兴趣时，分裂方法的潜在优势(与凝聚方法相比)就会显现出来。

可以通过递归地应用任何组合方法使用分裂方法。例如，令 $K = 2$，使用 K-均值(见第 14.3.6 节)或 K-中心点(见第 14.3.10 节)，在每次迭代中执行分裂。然而，这样的方法将依赖于每一步指定的初试格局。另外，它也不一定产生一个分裂序列，使得它具有树状图表示所要求的单调性。

Macnaughton Smith 等人(1965)提出了一种避免这些问题的分裂算法。开始时，该算法把所有的观测放入一个单独的簇 G。然后选择这样的观测，它与其他观测平均相异度最大。该观测构成第二个簇 H 的第一个成员。在每个后继步，将 G 中观测到 H 中观测的平均距离减去与 G 中其他观测的平均距离，并选择差值最大的观测将其移到 H。重复执行该过程，直到相应差值的平均值变成负数。也就是说，G 中不再有任何这样的观测，它在平均意义上更接近于 H 中的观测。结果原来的簇分成了两个子簇：转到 H 的观测和留在 G 中的观测。这两个簇代表层次结构的第二层。每个后继层是对前一层中的一个簇应用该分裂过程的处理结果。Kaufman 和 Rousseeuw(1990)提出选择每一层中具有最大直径(14.44)的簇进行分裂。另一种可选的方法是选择具有最大平均相异度的簇：

$$\bar{d}_G = \frac{1}{N_G}\sum_{i\in G}\sum_{i'\in G} d_{ii'}$$

继续递归分裂，直到所有的簇都成为单元素簇，或者每个簇中所有成员之间的相异度均为 0。

14.4　自组织映射

自组织映射(SOM)可以看做 K-均值的约束版本；这里，原型被置于特征空间的一维或二维流形中。结果的流形也称约束拓扑映射(constrained topological map)，因为原始的高维观测可以映射到二维坐标系中。最初的 SOM 算法是在线的，一次处理一个观测，后来又提出了批处理版本。该技术与主曲线和曲面(principal curves and surface)也有密切的关系，将在下一节讨论。

考虑 SOM 和 K 个原型 $m_j \in \mathbb{R}^p$ 的二维矩形网格(也可以用其他选择，如六边形网格)。每个原型都关于一对整数坐标 $\ell_j \in \mathcal{Q}_1 \times \mathcal{Q}_2$ 参数化。这里，$\mathcal{Q}_1 = \{1,2,\cdots,q_1\}$，$\mathcal{Q}_2$ 也类似，并且 $K = q_1 \cdot q_2$。初始化 m_j，例如，将其放在数据的二维主分量平面(见下一节)。我们可以把原型

想像为“纽扣”，被“缝合”在正则模式的主分量平面上。SOM 过程试图将该平面弯曲使纽扣尽可能地近似数据点。一旦模型拟合，观测就可以映射到二维网格上。

一次处理一个观测 x_i。寻找离 x_i 最近的原型 m_j（“最近”用 $\mathbb{R}^p$ 中的欧氏距离度量）；然后，对于 m_j 的所有近邻 m_k，通过更新

$$m_k \leftarrow m_k + \alpha(x_i - m_k) \tag{14.46}$$

将 m_k 移向 x_i。m_j 的“近邻”定义为使 ℓ_j 和 ℓ_k 之间距离较小的所有 m_k。最简单的方法使用欧氏距离，“较小”的概念通过一个阈值 r 来确定。近邻常常包括最近的原型 m_j 本身。

注意，距离定义在原型的整数拓扑坐标系空间 $\mathcal{Q}_1 \times \mathcal{Q}_2$ 中，而不是定义在特征空间 $\mathbb{R}^p$ 中。更新(14.46)的作用是把原型向数据移近，但同时也保持了原型之间光滑的二维空间联系。

SOM 算法的性能依赖于学习率 α 和距离阈值 r。典型地，在数千次迭代（每个观测迭代一次）中，α 从 1.0 递减到 0.0。类似地，r 也线性地递减，在数千次迭代中从初值 R 降低到 1。下面举例说明选择 R 的方法。

我们已经讨论过 SOM 最简单的版本。更复杂的版本使用距离

$$m_k \leftarrow m_k + \alpha h(\|\ell_j - \ell_k\|)(x_i - m_k) \tag{14.47}$$

修改更新步骤。其中 h 为邻域函数（neighborhood function），指标 ℓ_k 离 ℓ_j 越近，h 对原型 m_k 给予的权重就越大。

如果距离 r 取得足够小，使得每个邻域只包含一个点，则将失去原型之间的空间联系。在这种情况下，可以证明 SOM 算法是 K-均值聚类的一个在线版本，并且最终稳定在一个由 K-均值求得的局部极小值上。由于 SOM 是 K-均值聚类的一个约束版本，所以必须检查约束对任意给定的问题是否合理。该检查可以通过计算重构误差 $\|x - m_j\|^2$ 来实施，对于两种方法均在观测上求和。对于 K-均值方法，该值必然较小；但是，如果 SOM 是一个合理的近似，该值就不应该太小。

作为一个说明性例子，我们在三维空间产生 90 个数据点，接近半径为 1 半球面。这些点分属三个簇——红、绿和蓝——位于(0,1,0)、(0,0,1)和(1,0,0)附近。数据如图 14.15 所示。

按照设计，红色簇比绿色或蓝色簇更紧密（数据生成的细节在习题 14.5 中给出）。使用 5×5的原型网格，初始网格的尺寸为 $R = 2$；这意味着每个邻域开始大约有三分之一的原型。我们对 90 个观测的数据集做了 40 遍处理，并且在这 3600 次迭代中令 r 和 α 线性递减。

在图 14.16 中，原型用圆圈表示，投影到每个原型的点随机绘制在相应的圆圈中。左图所示为初始格局，右图为最终的结果。算法成功地分割了这些簇；但是，红色簇的分割表明流形其本身折回了（见图 14.17）。由于在二维图示中没有用距离，所以在 SOM 投影中没有多少迹象表明红色簇比其他两个簇更紧密。

图 14.18 所示为重构误差，它等于在其原型周围每个数据点的平方和。为了比较，我们执行 25 个中心点的一个 K-均值聚类，并且通过图中的水平线表示其重构误差。我们看到，SOM 显著地减少误差，接近 K-均值聚类的水平。这也说明 SOM 使用的二维约束对于这种特定的数据是合理的。

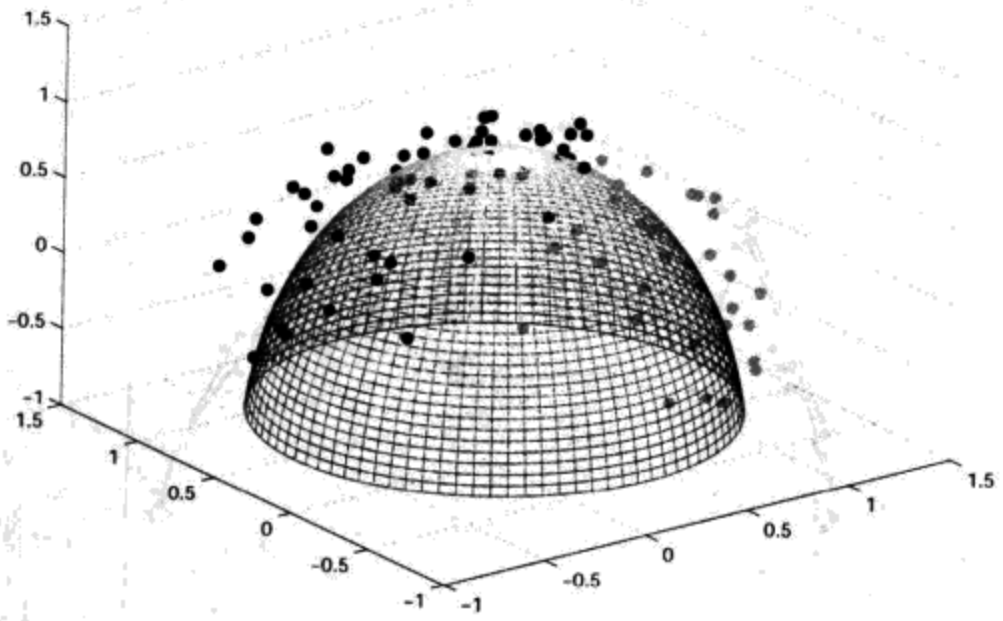


图 14.15　聚为三个类的模拟数据，接近一个半球面(见彩页)

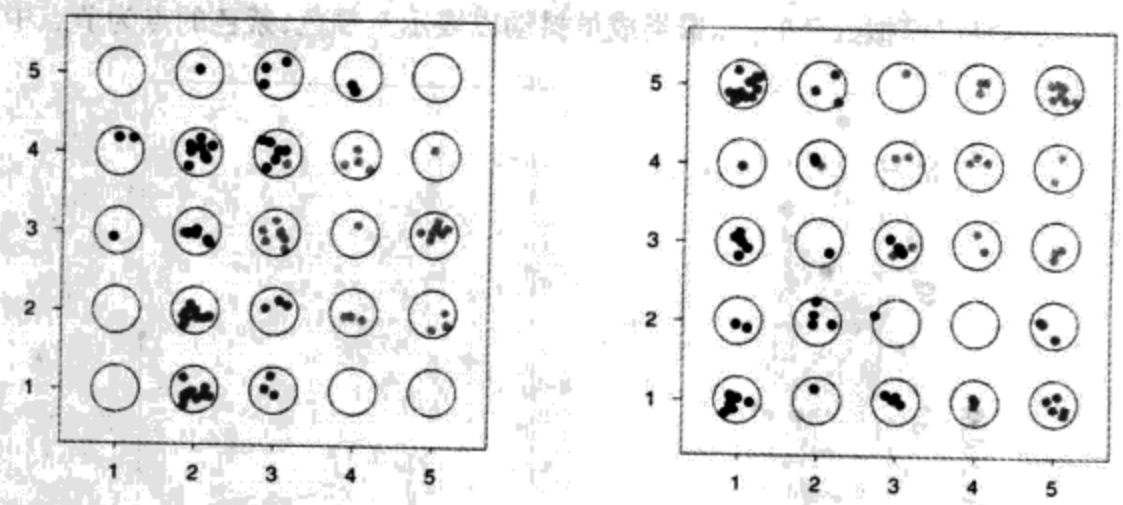


图 14.16　自组织映射用于半球面数据的例子。左图是初始格局，右图是最终结果。5×5 的原型网格由圆圈表示，投影到每个原型的点随机绘制在相应的圆圈内(见彩页)

在 SOM 的批处理版本中，我们通过下式更新每个 m_j：

$$m_j = \frac{\sum w_k x_k}{\sum w_k} \tag{14.48}$$

这里，我们对映射到(即靠近) m_j 的近邻 m_k 的点 x_k 求和。权值函数可以是矩形的，即对 m_k 的近邻等于 1，或者像以前一样随距离 $\| \ell_k - \ell_j \|$ 而平缓递减。如果邻域选得足够小，以至于它只包含 m_k，使用矩形权值，它将退化为前面讲的 K-均值聚类过程，也可以看做是主曲线和主曲面(见第 14.5 节)的一个离散版本。

例：文档的组织和检索

随着 Internet 和 Web 的迅速发展，文档检索得到了重视，已经证明，SOM 对于大型语料的组织和标引是有用的。该例取自 WEBSOM 的主页：http://websom.hut.fi/websom/(Kohonen 等人，2000)。图 14.19 表示对 12 088 个新闻组 comp.ai.neural-nets 文章的一个 SOM 拟合。标记由 WEBSOM 软件自动生成，并且对节点的代表性内容提供指导。

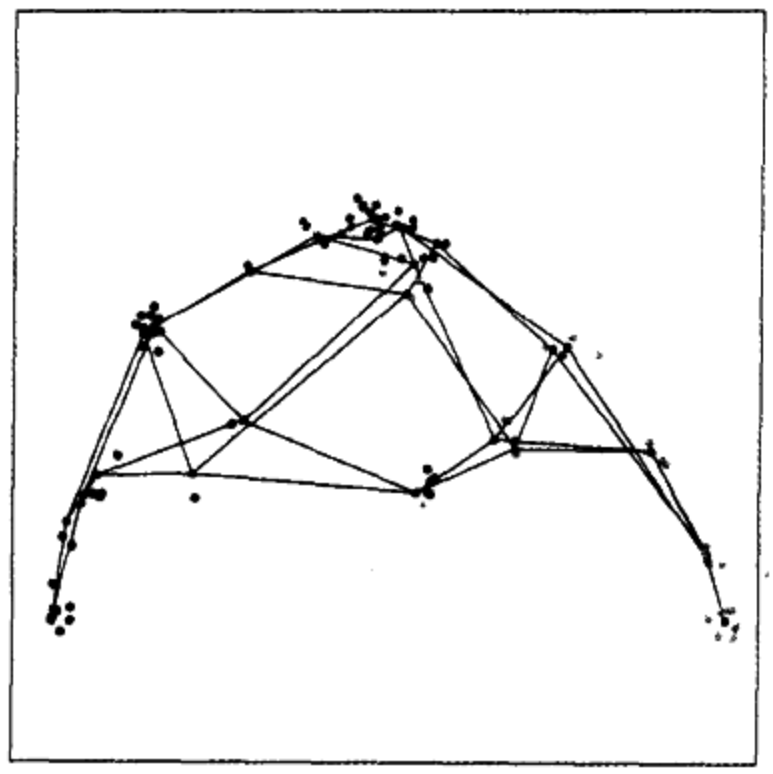


图 14.17 $I\!R^3$ 中拟合 SOM 模型的线网表示。其中直线表示拓扑网格的水平边和垂直边，双线表示曲面对角地折回自身以模拟红色的点。聚类成员抖动以表示其颜色，紫色的点为节点中心(见彩页)

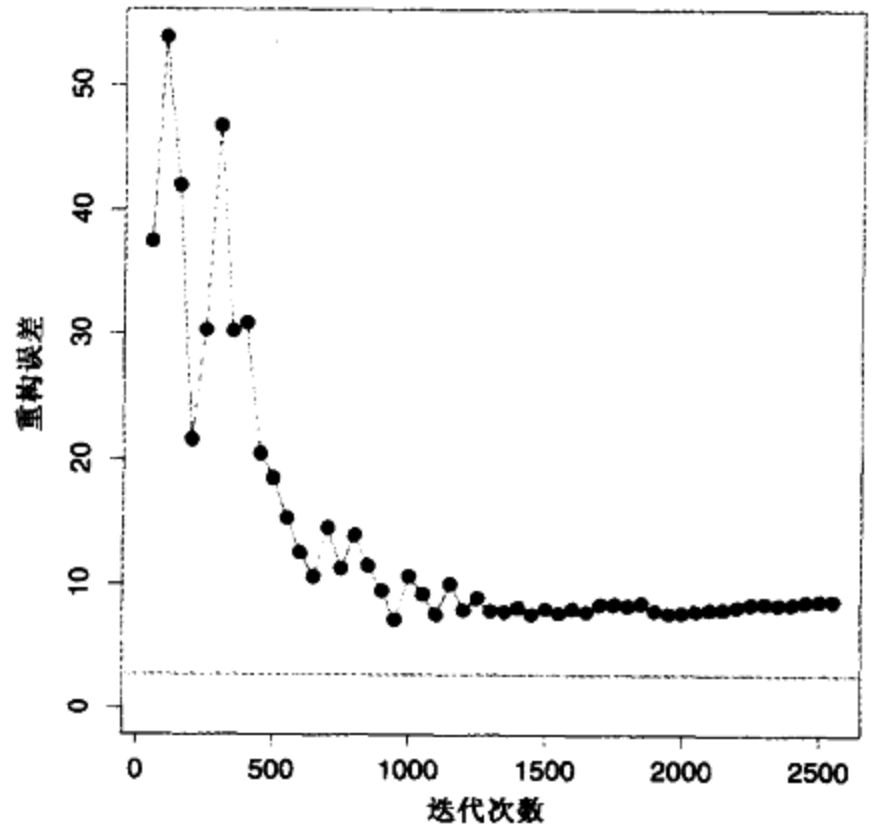


图 14.18 半球面数据：SOM 重构误差，作为迭代次数的函数。K-均值聚类的误差由水平线表示

在诸如此类的应用中，需要处理文档并创建特征向量。我们创建术语 - 文本矩阵，其中每行代表一个单独的文档。每行的元素是某文档对预先规定的术语集合的相对频率。这些术语可能是一个大的词典词条集合(50 000 个单词)，甚至是一个二元语法的集合(词对)，或者是它们的子集。一般地，这些矩阵都非常稀疏，因此经常对它们做一些预处理以减少特征(列)的个数。有时利用 SVD(见下一节)来压缩矩阵，Kohonen 等人(2000)使用了它的一个随机变异版本。然后将这些压缩的向量作为 SOM 输入。

在该应用领域，作者开发了一种“缩放”特征，允许用户与映射交互，以得到更详细的信息。最终一级放大检索实际可读的新闻文章。

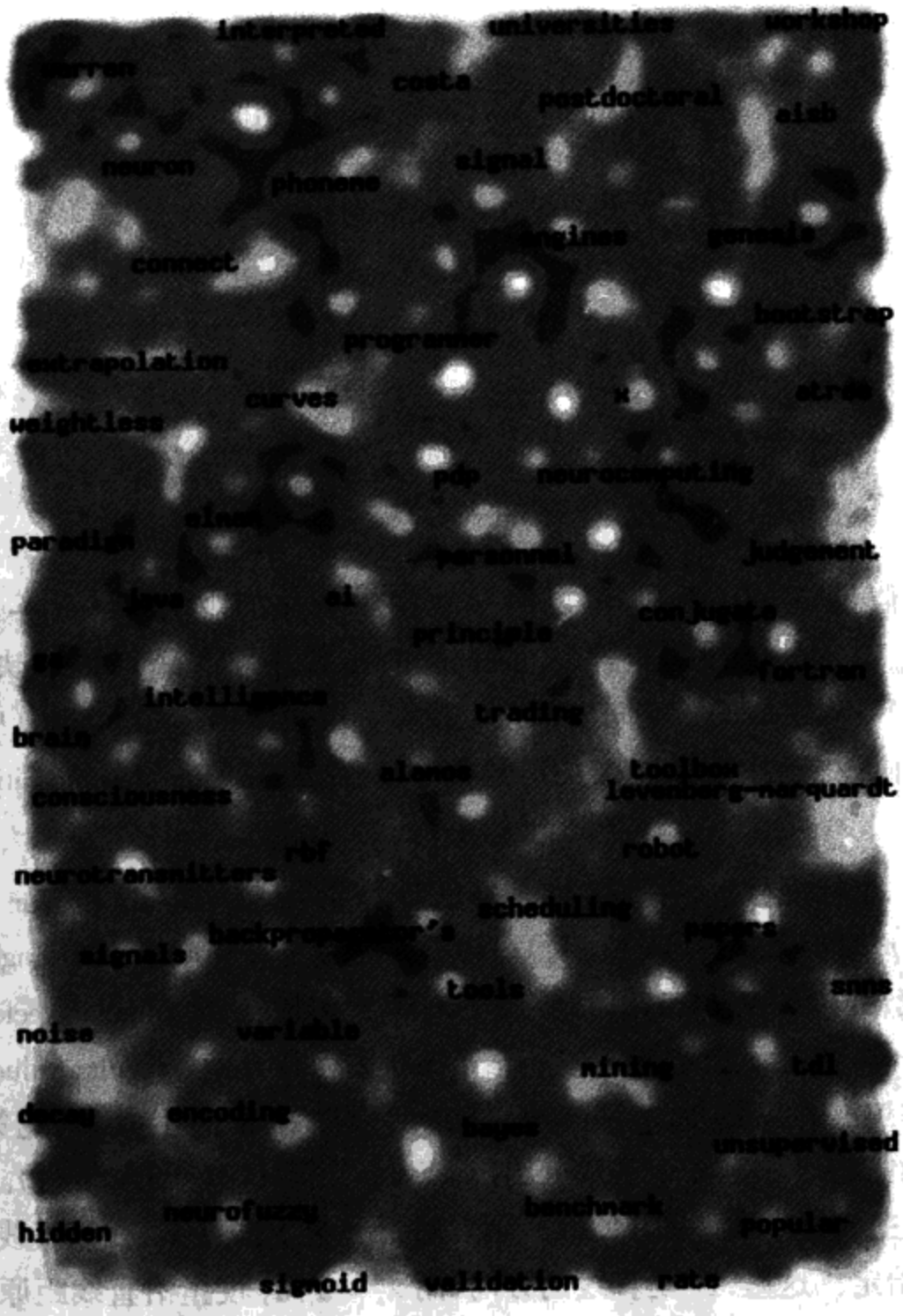


图 14.19　SOM 模型拟合 comp.ai.neural-nets 捐助(免费的 WEBSOM 主页)的 12 088 个新闻组语料的热度图表示。浅色区域表示高密度区域。其中的点是根据有代表性的内容自动标记

14.5　主成分、曲线和曲面

14.5.1　主成分

$\mathbb{R}^p$ 中数据集合的主成分提供了对所有秩 $q \leqslant p$ 的数据的一系列最佳线性逼近。

记观测为 $x_1, x_2, \cdots, x_N$，考虑表示它们的秩 q 线性模型：

$$f(\lambda) = \mu + \mathbf{V}_q \lambda \tag{14.49}$$

其中，μ 是 $\mathbb{R}^p$ 中的定位向量，$\mathbf{V}_q$ 是其 q 个列为正交单位向量的 $p \times q$ 矩阵，λ 是参数的 q 向量。这是秩 q 的一个仿射超平面的参数表示。图 14.20 和图 14.21 分别是对 $q = 1$ 和 $q = 2$ 的图解说明。通过最小二乘方对数据拟合这样一个模型相当于极小化重构误差(reconstruction

error)：

$$\min_{\mu,\{\lambda_i\},\mathbf{V}_q} \sum_{i=1}^{N} \|x_i - \mu - \mathbf{V}_q\lambda_i\|^2 \tag{14.50}$$

我们可以对 μ 和 λ_i 分别进行优化(见习题 14.7)，得到：

$$\hat{\mu} = \bar{x} \tag{14.51}$$

$$\hat{\lambda}_i = \mathbf{V}_q^T(x_i - \bar{x}) \tag{14.52}$$

剩下的任务是求解正交矩阵 $\mathbf{V}_q$：

$$\min_{\mathbf{V}_q} \sum_{i=1}^{N} \|(x_i - \bar{x}) - \mathbf{V}_q\mathbf{V}_q^T(x_i - \bar{x})\|^2 \tag{14.53}$$

为方便起见，我们假设 $\bar{x} = 0$(否则，可以简单地用中心化的观测 $\tilde{x}_i = x_i - \bar{x}$ 取代原观测)。$p \times p$矩阵 $\mathbf{H}_q = \mathbf{V}_q\mathbf{V}_q^T$ 是一个投影矩阵(projection matrix)，并且将每个点 x_i 映射到它的秩 q 重构 $\mathbf{H}_q x_i$——x_i 到 $\mathbf{V}_q$ 列生成的子空间上的正交投影。解可以描述如下。将(中心化的)观测放入一个 $N \times p$ 矩阵 $\mathbf{X}$ 的行中。构造 $\mathbf{X}$ 的奇异值分解(singular value decomposition)：

$$\mathbf{X} = \mathbf{U}\mathbf{D}\mathbf{V}^T \tag{14.54}$$

这是数值分析的一个标准分解，并且有许多算法计算它(例如，Golub 和 Van Loan，1983)。这里，$\mathbf{U}$ 是一个 $N \times p$ 的正交矩阵($\mathbf{U}^T\mathbf{U} = \mathbf{I}_p$)，其列 $\mathbf{u}_j$ 称为左奇异向量(left singular vectors)；$\mathbf{V}$ 是一个 $p \times p$ 的正交矩阵($\mathbf{V}^T\mathbf{V} = \mathbf{I}_p$)，其列 v_j 称为右奇异向量(right singular vectors)；而 $\mathbf{D}$ 是一个 $p \times p$ 的对角矩阵，其对角元素 $d_1 \geqslant d_2 \geqslant \cdots \geqslant d_p \geqslant 0$ 称为奇异值(singular values)。对于每个秩 q，式(14.53)的解 $\mathbf{V}q$ 由 $\mathbf{V}$ 的前 q 列组成。$\mathbf{UD}$ 的列称做 $\mathbf{X}$ 的主成分(见第 3.4.4 节)。式(14.52)中 N 个最优的 $\hat{\lambda}_i$ 由前 q 个主成分给出($N \times q$ 矩阵 $\mathbf{U}_q\mathbf{D}_q$ 的 N 个行)。

IR^2 中的一维主成分直线如图 14.20 所示。对每一个数据点 x_i，在该直线上存在一个最近的点，由 $u_{i1}d_1v_1$ 给定。这里，v_1 是直线的方向，而 $\hat{\lambda}_i = u_{i1}d_1$ 度量沿直线距原点的距离。类似地，图 14.21 所示为二维主成分的面拟合半球面数据(左图)。右图显示数据到前两个主成分上的投影。该投影是前面所讲的 SOM 方法初始格局的基础。该过程相当成功地分割了邻近的各个簇。由于半球面是非线性的，因此非线性投影将会做得更好，而这是下一节的主题。

主成分还有许多其他好的性质，例如，在所有特征的线性组合中，$\mathbf{X}v_1$ 具有最高的方差；在所有满足 v_2 与 v_1 正交的线性组合中，$\mathbf{X}v_2$ 具有最高的方差，如此等等。

例：手写体数字

对于维归约和压缩，主成分分析是一个有用的工具。现在以第 1 章介绍的手写体数字为例解释该特性。图 14.22 所示为 130 个手写体“3”的样本，每个都是一个数字化的 16×16 点阵灰度图，样本来自 658 个同类样本的总体。我们看到在书写风格、字体粗细和字体方向上各图像间有相当大的偏倚。把这些图像看做 IR^{256} 中的点 x_i，并且通过 SVD(14.54)计算它们的主成分。

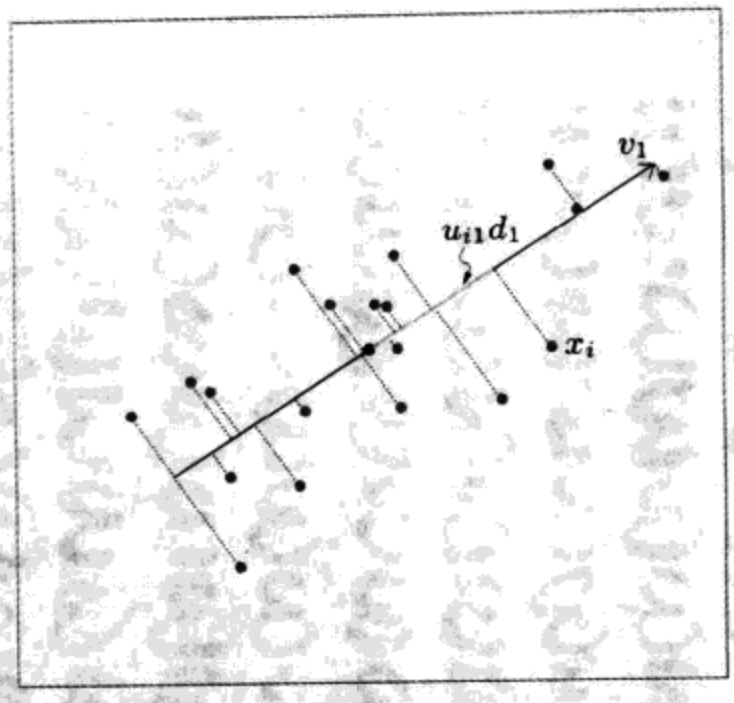


图 14.20　数据集的第一个线性主成分。该直线极小化每个点与其在该直线的正交投影的距离平方和(见彩页)

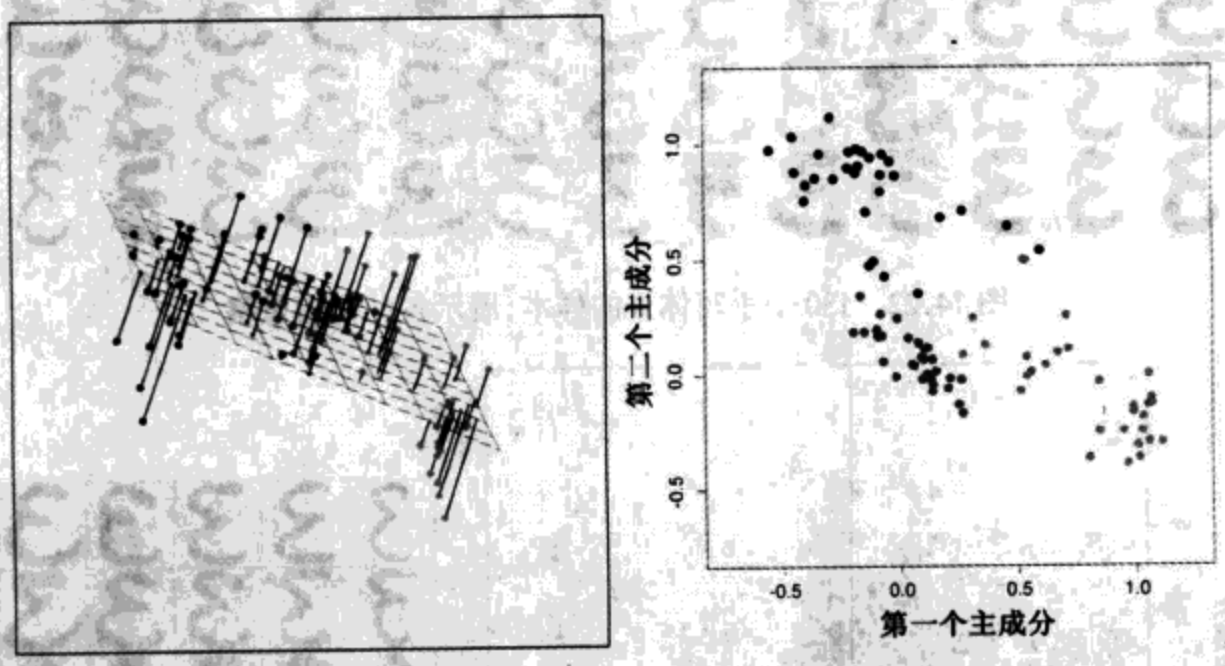


图 14.21　对半球面数据最好的秩 2 线性近似。右图显示投影点,其坐标由数据的前两个主成分 $\mathbf{U}_2\mathbf{D}_2$ 给出(见彩页)

图 14.23 所示为这些数据的前两个主成分。对每一个这样的主成分 u_{i1} 和 u_{i2},计算其 5%、25%、50%、75% 和 95% 分位点,并用它们定义叠加在图上的矩形网格。圆圈中的点指示靠近网格顶点的图像,其中距离度量主要集中在它们的投影坐标上,但也对正交子空间中的分量赋予一定权值。右图所示为在左图中被圈起来的点的对应图像。这使前两个主成分性质可视化。我们看到,v_1(水平移动)主要说明 3 的下面尾巴的长度,而 v_2(垂直移动)主要说明 3 的字体粗度。根据参数模型(14.49),该二成分模型有如下形式:

$$\begin{aligned}\hat{f}(\lambda) &= \bar{x} + \lambda_1 v_1 + \lambda_2 v_2 \\ &= \text{[3]} + \lambda_1 \cdot \text{[v1]} + \lambda_2 \cdot \text{[v2]}\end{aligned}$$

这里,我们用图像表现前两个主成分 v_1 和 v_2。尽管有 256 个可能的主成分,但是大约 50 个导致图像 90% 的变化,12 个导致图像 63% 的变化。图 14.24 比较了这些奇异值和从等价的不相关数据得到的奇异值,这些不相关数据通过随机拼凑 $\mathbf{X}$ 的每列得到。数字化图像中的像素是自然相关的,并且由于它们是同样的数字,相关性就更强。一个相对小的主成分子集可以用做

表示高维数据非常好的较低维特征。

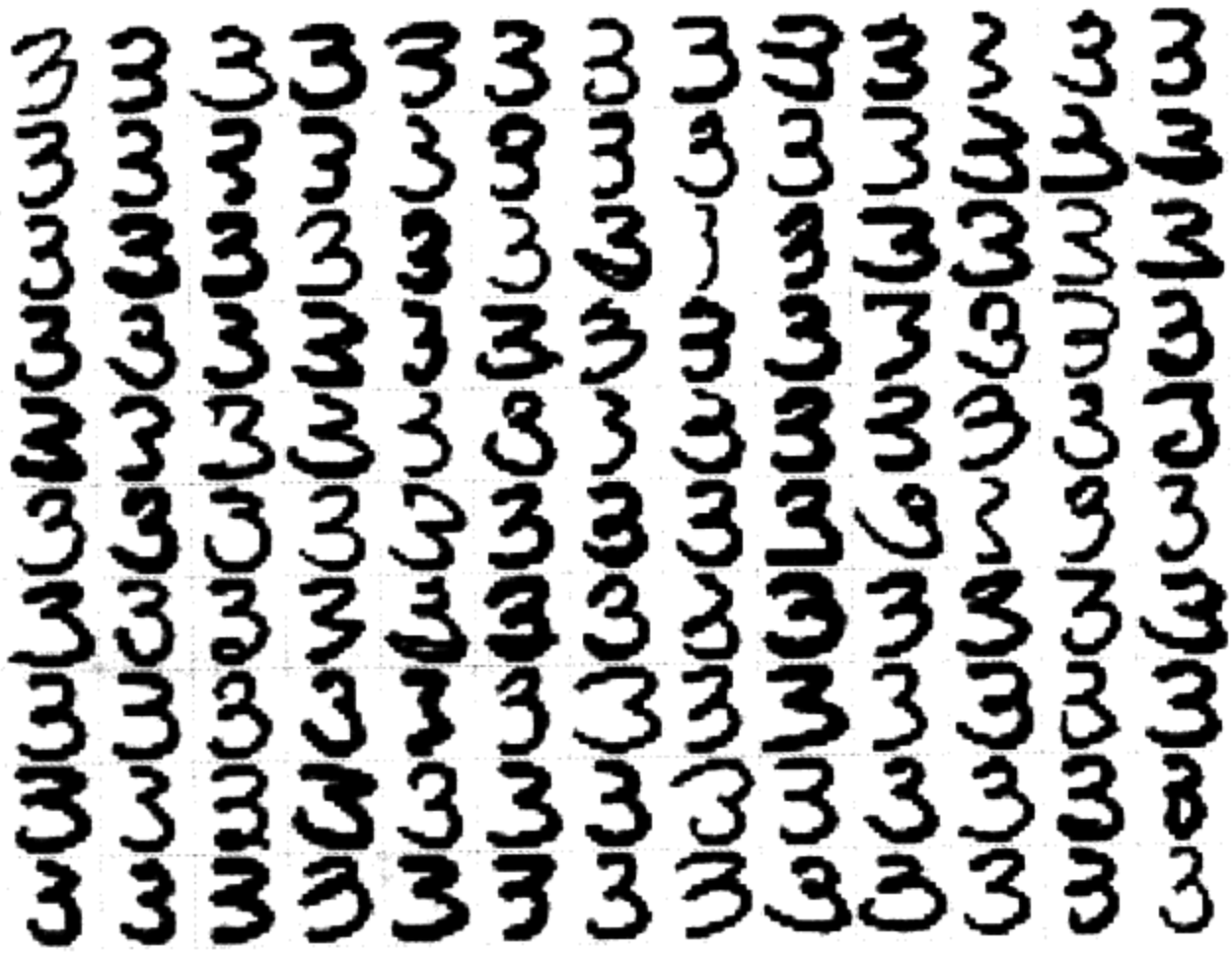


图 14.22 130 个手写体 3 的样本，展示各种书写风格

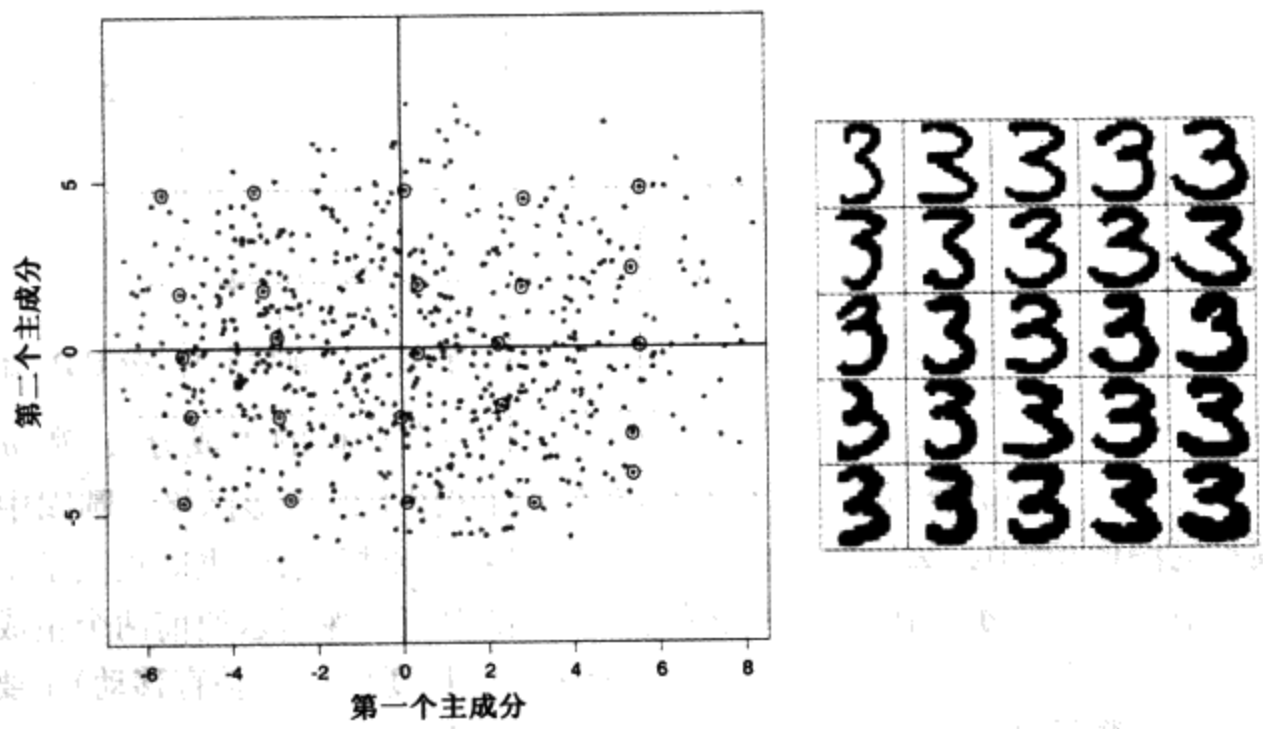


图 14.23 左图：手写体 3 的前两个主成分。被圈起来的点是到网格顶点最近投影的图像，距离由主成分的边缘分位数定义。右图：对应于圈起来的点的图像。它们显示了前两个主成分的本质

14.5.2 主曲线和曲面

主曲线拓广了主成分直线，提供了 $\mathbb{R}^p$ 中数据点集合光滑的一维曲线近似。主曲面则更一般，提供了二维或更高维的曲流形近似。

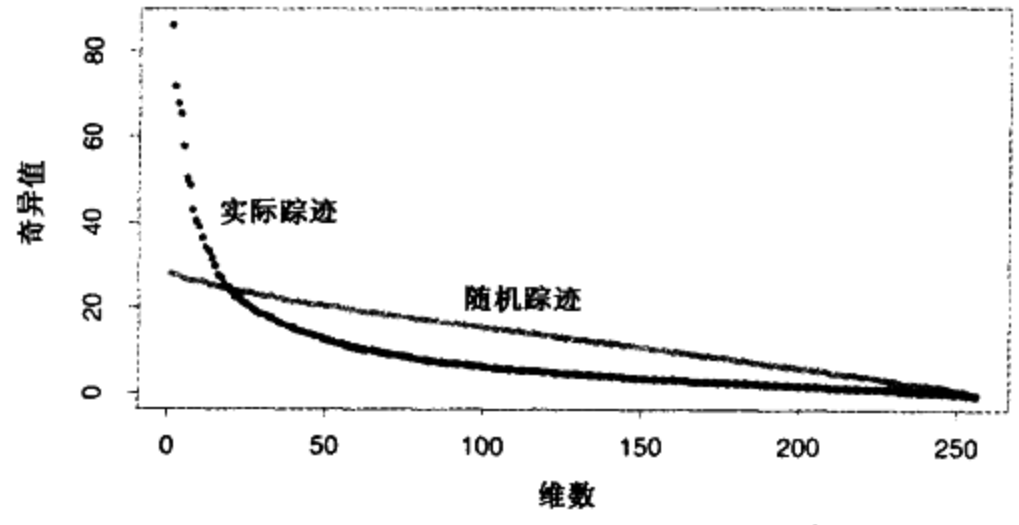


图 14.24　数字化 3 的 256 个奇异值,与数据随机视图的对比(X 的每个列是乱序的)

首先为随机变量 $X \in \mathbb{R}^p$ 定义主曲线,然后转向有限数据的情况。令 $f(\lambda)$为 $\mathbb{R}^p$ 中参数化的光滑曲线。因此 $f(\lambda)$是具有 p 个坐标的向量函数,每一个为单个参数 λ 的光滑函数。参数 λ 是可选择的,例如,它可以是自某固定原点的曲线弧长。对于每一个数据值 x,令 $\lambda_f(t)$定义曲线上离 x 最近的点。如果

$$f(\lambda) = \mathrm{E}(X|\lambda_f(X) = \lambda) \tag{14.55}$$

则称 $f(\lambda)$为随机向量 X 分布的主曲线。这表示 $f(\lambda)$是所有投影于它的数据点的平均值,即对它"响应"的点。这也称做自相容性(self-consistency property)。尽管在实际上,连续的多元分布有无穷多的主曲线(Duchamp 和 Stuetzle, 1996),我们仍然主要对光滑的主曲线感兴趣。图 14.25 所示为一个主曲线的例子。

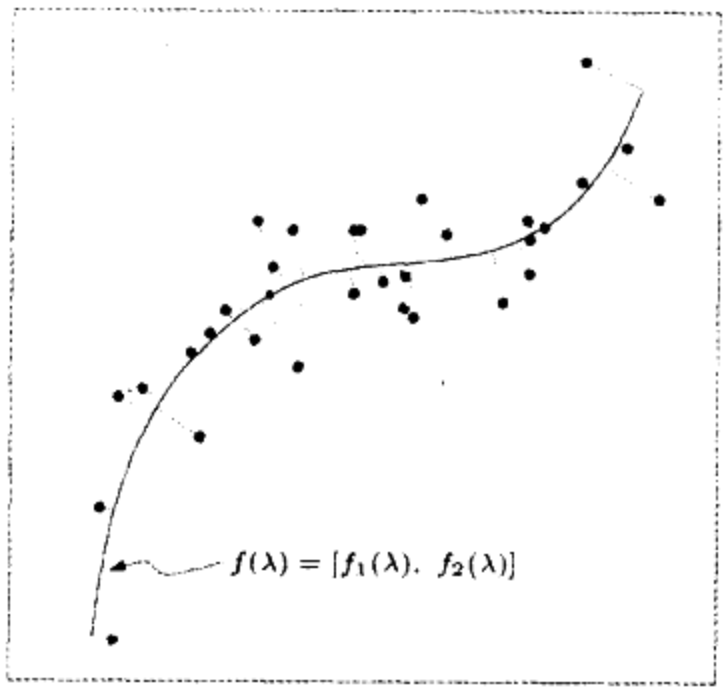


图 14.25　一个数据集合的主曲线。曲线上的每个点是投影到那里的所有数据点的平均值

主点(principal point)是一个有趣的相关概念。考虑 k 个原型的集合,对于分布的支集中的每个点 x,识别最近的原型,即对它响应的原型。这导致将特征空间划分为 Voronoi 区域。极小化 X 到其原型期望距离的 k 个点的集合称为分布的主点。每一个主点是自相容的,因为它等于 X 在其 Voronoi 区域的均值。例如,当 $k=1$ 时,圆正态分布的主点是均值向量;当 $k=2$ 时,它们是一对点,对称地置于一条穿过平均向量的射线上。主点是由 K-均值聚类所发现的中心点的分布模拟。主

曲线可以看做是 $k=\infty$ 个主点，但被约束在光滑曲线上；同样，SOM 约束 K-均值聚类中心落入光滑的流形上。

为求得一个分布的主曲线 $f(\lambda)$，考虑其坐标函数 $f(\lambda)=[f_1(\lambda),f_2(\lambda),\cdots,f_p(\lambda)]$，并且令 $X=(X_1,X_2,\cdots,X_p)$。考虑以下的交替步骤：

$$\begin{array}{lll} \text{(a)} & \hat{f}_j(\lambda) \leftarrow & \mathrm{E}(X_j|\lambda(X)=\lambda);\ j=1,2,\cdots,p \\ \text{(b)} & \hat{\lambda}_f(x) \leftarrow & \mathrm{argmin}_{\lambda'}\|x-\hat{f}(\lambda')\|^2 \end{array} \tag{14.56}$$

第一个方程固定 λ，加强自相容性的要求(14.55)。第二个方程固定曲线，求曲线上到每个数据点的最近的点。对于有限数据，主曲线算法以线性主成分开始，并迭代执行式(14.56)中的两步直到收敛。通过光滑每个 X_j[作为一个弧长 $\hat{\lambda}(X)$的函数]，利用散点图光滑子估计步骤(a)中的条件期望，并对每个观测数据点执行步骤(b)的投影。一般证明收敛是困难的，但可以证明：如果对散点图光滑使用线性最小二乘方拟合，则过程将收敛到第一个线性主成分，并且等同于求解矩阵的最大本征向量的幂方法(power method)。

主曲面与主曲线有着完全相同的形式，但是维数更高。最常用的是二维主曲面，具有坐标函数：

$$f(\lambda_1,\lambda_2)=[f_1(\lambda_1,\lambda_2),\cdots,f_p(\lambda_1,\lambda_2)]$$

上述步骤(a)中的估计由二维曲面光滑子得到。很少使用高于二维的主曲面，这是因为其可视化特性不够好，在高维中的光滑也是同样。

图 14.26 所示的是主曲面拟合半球面数据的结果。绘制的数据点是估计的非线性坐标 $\hat{\lambda}_1(x_i),\hat{\lambda}_2(x_i)$的函数。图中类的分割是明显的。

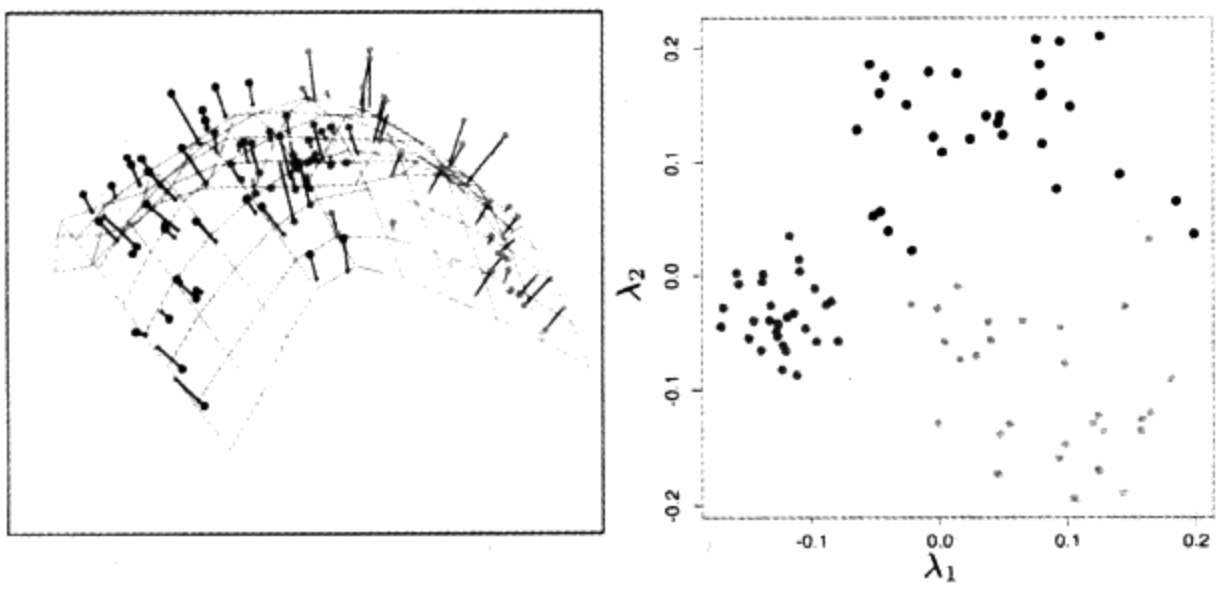


图 14.26 主曲面拟合半球面数据。左图：拟合的二维曲面。右图：数据点在曲面上的投影，结果在坐标上$\hat{\lambda}_1$，$\hat{\lambda}_2$(见彩页)

主曲面非常类似于自组织映射。如果我们使用核曲面光滑子估计每个坐标函数 $f_j(\lambda_1,\lambda_2)$，就与 SOM 的批处理版本(14.48)具有同样的形式。SOM 的权值 w_k 正好就是核中的权值。然而，它们也有区别：对于每个数据点 x_i，主曲面估计一个原型 $f(\lambda_1(x_1),\lambda_2(x_i))$，而 SOM 对所有的数据点共享较少的原型。这样，只有当 SOM 原型的个数增长很大时，SOM 和主曲面才取得一致。

另外，二者之间还有概念上的不同。主曲面以其坐标函数的方式提供整个流形的光滑参数，而 SOM 是离散的，只为近似的数据产生估计原型。主曲面的光滑参数局部地保持距离：图 14.26 示出

红色的簇比绿色或蓝色的簇更紧凑。在简单的例子中，估计坐标函数本身也能提供有用的信息，见习题 14.9。

14.6　独立成分分析和探测性投影寻踪

多元数据常被看做是来自对基础数据的多个间接观测，它们一般不能被直接度量。实例包括：

- 教育或心理方面的测试利用问卷来度量被测者的基本智力和其他智力方面的能力。
- EEG 脑扫描通过置于头部不同位置的传感器所记录的电磁信号，间接度量大脑不同部分的神经活动。
- 股票交易的价格随时间不断地变化，并反映着多种不可测量的因素，诸如市场信心、外部影响和其他不易识别或测量的驱动力。

因子分析是一种经典的技术，它在以识别这些本征源为目标的统计学文献中有详细的论述。典型地，因子分析模型与高斯分布结合在一起，这在某种程度上妨碍了它们的应用。最近出现的独立成分分析方法，成为因子分析的一个有力竞争者，而且正如我们将要看到的，其成功之处在于潜在源的非高斯特性。

14.6.1　本征变量和因子分析

奇异值分解 $\mathbf{X}=\mathbf{UDV}^T$(14.54)有一个本征变量表示。记 $\mathbf{S}=\sqrt{N}\mathbf{U}$ 和 $\mathbf{A}^T=\mathbf{DV}^T/\sqrt{N}$，我们有 $\mathbf{X}=\mathbf{SA}^T$，因此 $\mathbf{X}$ 的每一列是 $\mathbf{S}$ 的列的线性组合。由于 $\mathbf{U}$ 是正交的，并且像以前一样假设 $\mathbf{X}$ 的每一列(因此 $\mathbf{U}$ 也同样)具有均值 0。这意味 $\mathbf{S}$ 的列具有均值 0，是不相关的且具有单位方差。用随机变量的术语，可以将 SVD 或相关的主成分分析(PCA)解释为本征变量模型：

$$\begin{aligned} X_1 &= a_{11}S_1 + a_{12}S_2 + \cdots + a_{1p}S_p \\ X_2 &= a_{21}S_1 + s_{22}S_2 + \cdots + a_{2p}S_p \\ &\vdots \qquad\qquad \vdots \\ X_p &= a_{p1}S_1 + s_{p2}S_2 + \cdots + a_{pp}S_p \end{aligned} \tag{14.57}$$

或简单地，$X=\mathbf{A}S$ 的一个估计。每个相关变量 X_j 表示为一个非相关的单位方差变量 S_i 的线性展开式。但是这不太令人满意，因为给定任意 $p\times p$ 的正交矩阵 $\mathbf{R}$，我们都可以记：

$$\begin{aligned} X &= \mathbf{A}S \\ &= \mathbf{A}\mathbf{R}^T\mathbf{R}S \\ &= \mathbf{A}^*S^* \end{aligned} \tag{14.58}$$

并且 $\mathrm{Cov}(S^*)=\mathbf{R}\mathrm{Cov}(S)\mathbf{R}^T=\mathbf{I}$。因此，存在许多这样的分解，从而识别任意一个特定的本征变量为惟一潜在源是不可能的。SVD 分解确实有这样的特性：任意秩 $q<p$ 的截断分解都以最优的方式逼近 $\mathbf{X}$。

典型的因子分析(factor analysis)模型最初是由心理测量学研究者发展起来的，它在某种程度上缓解了这些问题；例如，参见 Mardia 等人(1979)的著作。当 $q<p$ 时，因子分析模型有下列形式：

$$\begin{aligned} X_1 &= a_{11}S_1 + \cdots + a_{1q}S_q + \varepsilon_1 \\ X_2 &= a_{21}S_1 + \cdots + a_{2q}S_q + \varepsilon_2 \\ &\vdots \qquad\qquad \vdots \\ X_p &= a_{p1}S_1 + \cdots + a_{pq}S_q + \varepsilon_p \end{aligned} \tag{14.59}$$

或 $X = \mathbf{A}S + \varepsilon$。这里，$S$ 是 $q < p$ 个基本本征变量或因子的向量，$\mathbf{A}$ 是 $p \times q$ 的因子负荷(loadings)矩阵，ε_j 是不相关的 0 - 均值干扰。基本思想是——本征变量是 X_j 中的公共变化源，并且是它们相关结构的原因所在，而不相关的 ε_j 对每个 X_j 是惟一的，并收集剩余无法解释的变化。典型地，S_j 和 ε_j 被模型化为高斯随机变量，而模型用极大似然拟合。参数在如下协方差矩阵中：

$$\mathbf{\Sigma} = \mathbf{A}\mathbf{A}^T + \mathbf{D}_\varepsilon \tag{14.60}$$

其中，$\mathbf{D}_\varepsilon = \text{diag}[\text{Var}(\varepsilon_1), \cdots, \text{Var}(\varepsilon_p)]$。$S_j$ 是高斯的并且是不相关的，这使得它们成为统计上独立的随机变量。因此，一组智力测验分数将被认为是由独立的基本因素(如智力、本能冲动等)驱动的。矩阵 $\mathbf{A}$ 的列称做因子负荷(factor loadings)，它用来命名和解释因子。

遗憾的是这里仍然存在可辨别性问题(14.58)，因为对于任意 $q \times q$ 的正交矩阵 $\mathbf{R}$，$\mathbf{A}$ 和 $\mathbf{AR}^T$ 在式(14.60)中都是等价的。这就允许因子分析的使用有一定的主观性，因为用户能够探求因子的旋转视图，使它们更容易解释。在这方面的问题导致了许多统计学家对因子分析产生怀疑，这也许是它在当代统计学中不够流行的原因。尽管我们不打算在这里讨论细节，但是 SVD 对式(14.60)的估计起着关键的作用。举例来说，如果假设所有的 $\text{Var}(\varepsilon_j)$ 相等，则 SVD 最重要的 q 个成分识别由矩阵 $\mathbf{A}$ 确定的子空间。

由于 ε_j 是对每个 X_j 的独立干扰，因子分析可以视为对 X_j 的相关结构而不是协方差结构建模。这一点可以容易地通过标准化(14.60)中的协方差结构看到(见习题 14.10)。这是因子分析与 PCA 之间的一个重要差别，尽管它不是这里讨论的中心。习题 14.11 讨论一个简单的例子；由于这种差别，导致因子分析的解和 PCA 的解显著不同。

14.6.2 独立成分分析

除了假设 S_i 为统计独立(statistically independent)而非不相关以外，独立成分分析(independent component analysis，ICA)模型和式(14.57)有着极为相似的形式。直观地，缺乏相关性决定多元分布的二阶交叉矩(协方差)，通常，统计独立性决定所有的交叉矩。这些额外的矩条件允许我们惟一识别 $\mathbf{A}$ 的元素。由于多元高斯分布仅由其二阶矩决定(这是一个例外)，并且和先前一样，任何高斯独立成分仅由旋度决定。因此，如果假设 S_i 是独立且非高斯的(non - Gaussian)，就可以避开式(14.57)和式(14.59)的识别问题。

现在我们来讨论如式(14.57)中的完整 p 成分模型，其中 S_ℓ 是独立的，具有单位方差；因子分析模型(14.59)的 ICA 版本仍存在。我们的处理基于 Hyvärinen 和 Oja(2000)的综述文章。

我们希望求解 $X = \mathbf{A}S$ 中的混合矩阵 $\mathbf{A}$。不失一般性，可以假设 X 已经被漂白(whitened)并有 $\text{Cov}(X) = \mathbf{I}$；典型地，这可以通过上述的 SVD 来实现。这意味着 $\mathbf{A}$ 是正交的，因为 $\mathbf{S}$ 也有协方差 $\mathbf{I}$。因此，解决 ICA 问题相当于求一个正交矩阵 $\mathbf{A}$，使向量随机变量 $S = \mathbf{A}^T X$ 的分量是独立的(并且是非高斯的)。

图 14.27 显示 ICA 在分离两个混合信号方面的能力。这是经典的鸡尾酒会问题(*cocktail party problem*)的一个例子，其中不同的麦克风 X_j 接收不同独立声源 S_ℓ(音乐、不同人的话音等)的混合。通过使用初始源的独立性和非高斯性，ICA 能够实现盲源分离(blind source separation)。

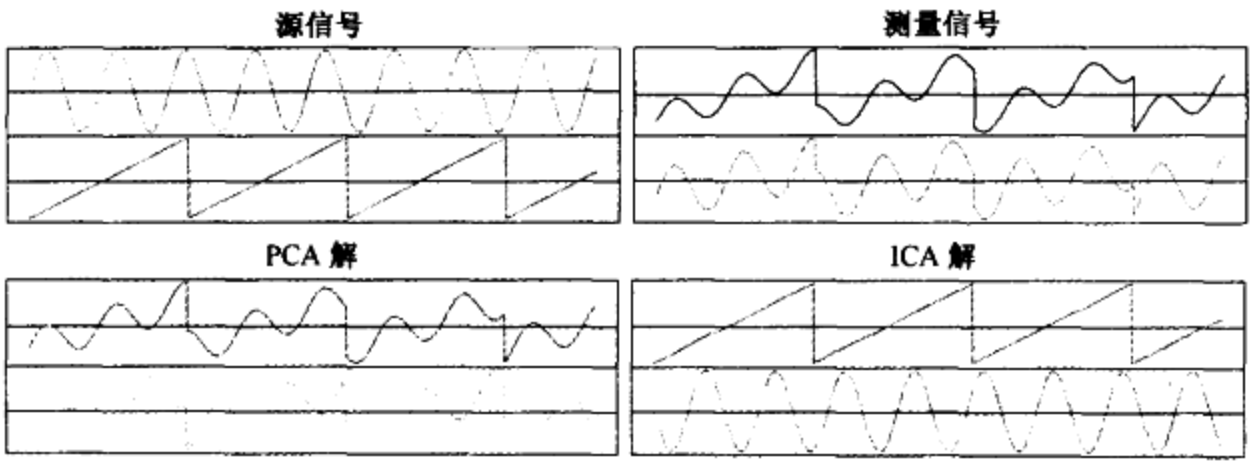


图 14.27　基于人工时间序列数据的 ICA 和 PCA 的比较示例。左上图所示为两个源信号，在 1000 个均匀分布时间点上测量而得。右上图所示为观测的混合信号。下面的两个图为主成分的解和独立成分的解

许多流行的 ICA 方法都是基于熵的。密度为$g(y)$的随机变量 Y 的微分熵 H 由下式给出：

$$H(Y) = -\int g(y)\log g(y)dy \tag{14.61}$$

信息论中有一个众所周知的结果——在具有相同方差的所有随机变量中，高斯变量的熵最大。最后，随机向量 Y 每两个分量之间的互信息 $I(Y)$是依赖性的一个自然度量：

$$I(Y) = \sum_{j=1}^{p} H(Y_j) - H(Y) \tag{14.62}$$

$I(Y)$称做 Y 的密度$g(y)$与其独立版本 $\prod_{j=1}^{p} g_j(y_j)$ 之间的 Kullback -Leibler 距离，其中$g_j(y_j)$是 Y_j 的边缘密度。既然 X 有协方差 $\mathbf{I}$，同时 $Y = \mathbf{A}^T X$，且 $\mathbf{A}$ 为正交的，那么就容易看出：

$$I(Y) = \sum_{j=1}^{p} H(Y_j) - H(X) - \log|\det \mathbf{A}| \tag{14.63}$$

$$= \sum_{j=1}^{p} H(Y_j) - H(X) \tag{14.64}$$

求解使 $I(Y) = I(\mathbf{A}^T X)$极小化的 $\mathbf{A}$ 即为求解一个正交变换，该正交变换导致其分量之间的最大独立性。根据式(14.63)，这等价于极小化 Y 的各分量熵的总和，相当于最大化背离高斯性。

为方便起见，还可以不用熵 $H(Y_j)$，Hyvärinen 和 Oja(2000)使用负熵(negentropy)度量 $J(Y_j)$，并定义：

$$J(Y_j) = H(Z_j) - H(Y_j) \tag{14.65}$$

其中 Z_j 是一个与 Y_j 有相同方差的高斯随机变量。负熵是非负的，度量 Y_j 对高斯性的背离。他们在文章中还提出对负熵的简单近似，该方法能在数据上计算并优化。图 14.27 和图 14.28 所示为 ICA 利用近似求解的结果：

$$J(Y_j) \approx [\mathrm{E}G(Y_j) - \mathrm{E}G(Z_j)]^2 \tag{14.66}$$

其中，对于 $1 \leqslant a \leqslant 2$, $G(u) = \frac{1}{a}\log\cosh(au)$。当应用于 x_i 的一个样本时，期望由数据的平均

值代替。更经典的(并且健壮性较弱)度量是基于四阶矩,因此通过峰度求解对高斯性的背离。更详细的资料和简单的牛顿算法用于求解最优的方向,参见 Hyvärinen 和 Oja(2000)。

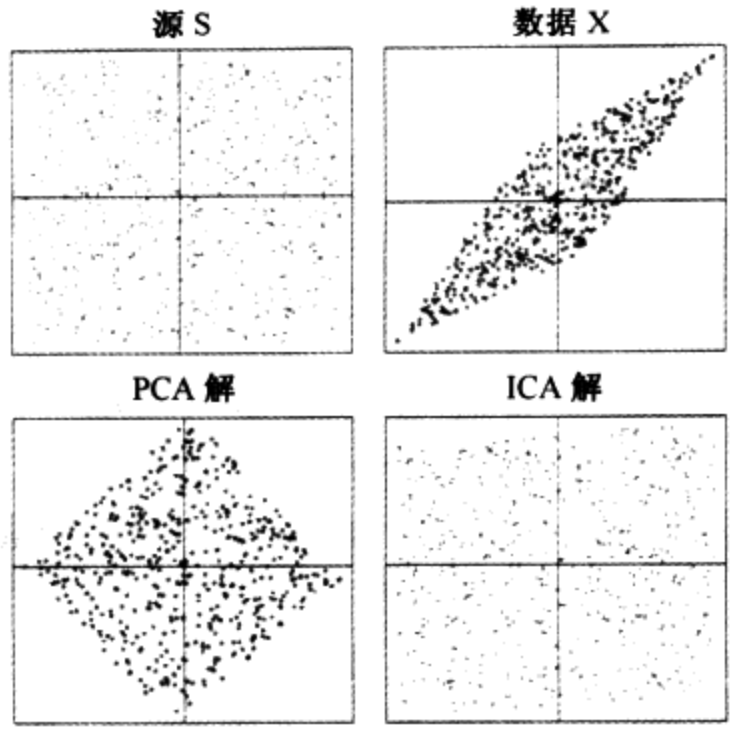


图 14.28 独立均匀分布的随机变量的混合。左上图所示为来自两个独立均匀分布数据源的500个样本,右上图是它们的混合视图。下面两幅图分别是PCA和ICA的解

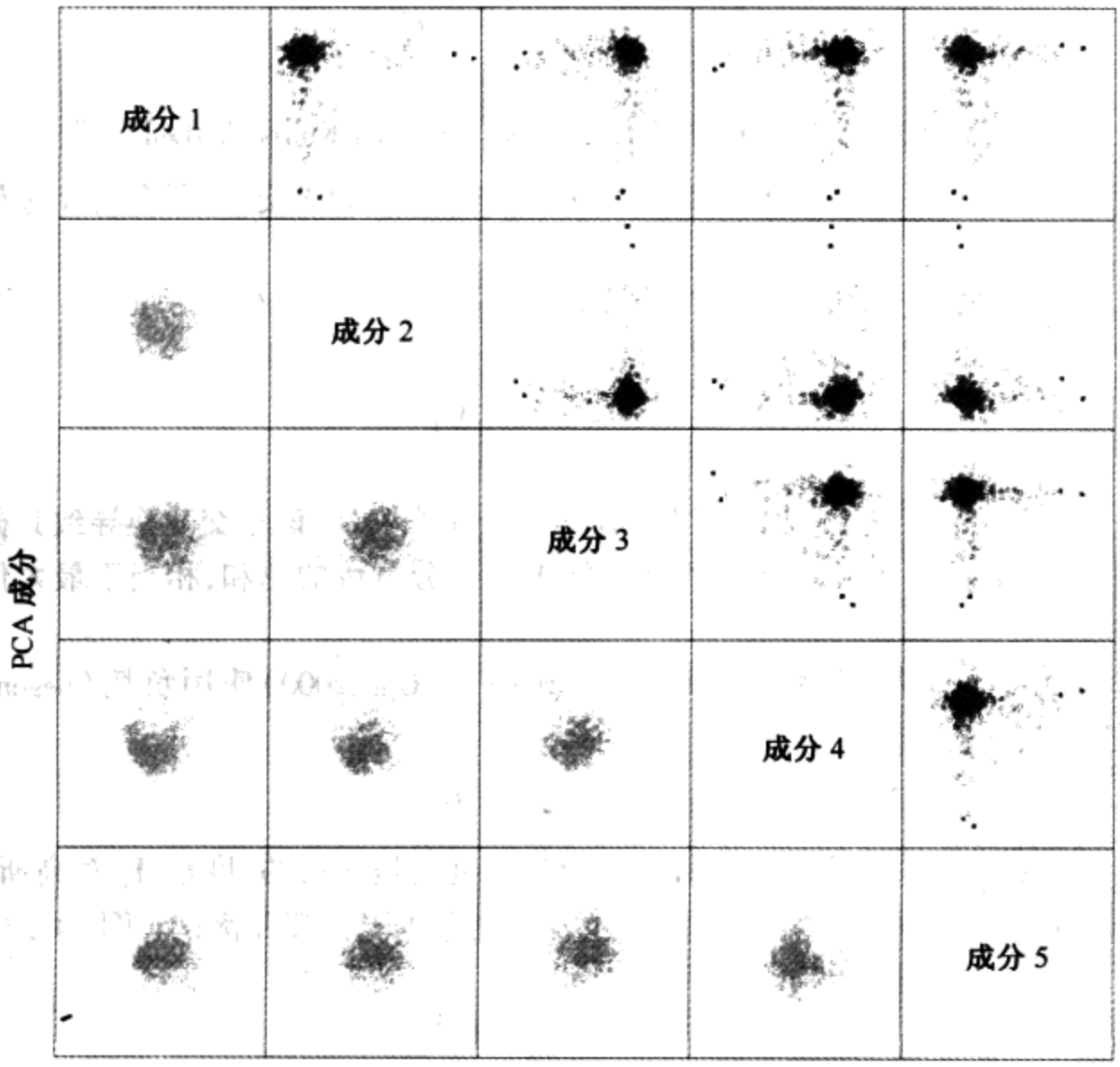


图 14.29 比较前 5 个 ICA 成分(对角线上面)和前 5 个 PCA 成分(对角线下面)。每个成分都被标准化,具有单位方差

总之，ICA 应用于多元数据求正交投影的一个序列，使得投影的数据看起来尽可能背离高斯性。利用预漂白的数据，这相当于寻找尽可能独立的成分。

ICA 基本上是从因子分析的解开始，寻找能产生独立成分的旋度。从这个角度看，ICA 正是另外一种因子旋转方法，与传统的“varimax”和“quartimax”一起用于心理测验。

例：手写体数字

我们再来看一下在第 14.5.1 节中由 PCA 分析的手写体 3。图 14.29 用前 5 个 ICA 成分比较了前 5 个(标准化了的)主成分，所有的显示都在同一个标准单位下。注意每一幅图都是从 256 维空间到二维空间的投影。PCA 的成分均呈现联合高斯分布，而 ICA 的成分具有长尾分布。这并不太奇怪，因为 PCA 关注方差，而 ICA 专门寻求非高斯分布。所有的成分都已被标准化，所以看不到主成分的递减方差。如图 14.30 所示，对于每一个 ICA 成分，我们加亮了两个极端数字和两个中心数字。该例说明了每个成分的特性。例如，ICA 成分 5 获取了有长长的弯曲尾巴的 3。

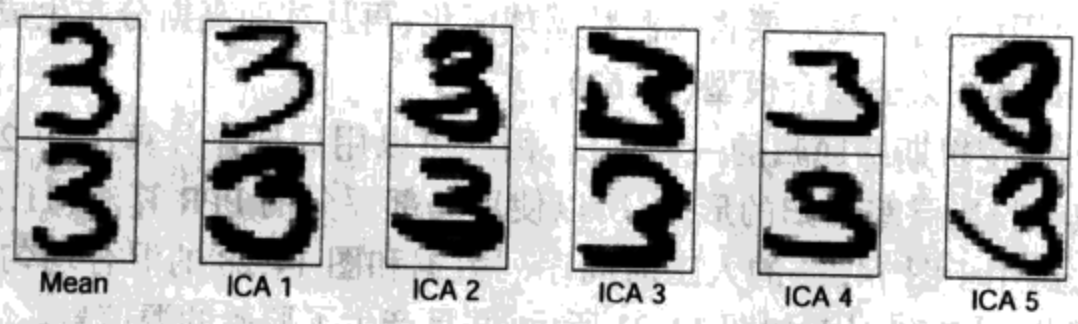


图 14.30　源自图 14.29 的加亮的数字。通过与均值数字 3 的比较，可以看到 ICA 成分的特性

14.6.3　探测性投影寻踪

Friedman 和 Tukey(1974)提出了探测性投影寻踪(exploratory projection pursuit)，这是一项可视化高维数据的图形探测技术。他们的观点是大多数高维数据的低维(一维或二维)投影看起来像高斯的。诸如聚类或长尾这样有趣的结构，将由非高斯投影揭示。他们为优化提出了许多投影指标(projection indices)，每个聚焦于一种对高斯性的背离。自最初的提议以来，又出现了多种改进(Huber，1985；Friedman，1987)，并且在交互图形包 Xgobi(Swayne 等人，1991)中实现了包括熵在内的多种指标。这些投影指标与上面的 $J(Y_j)$ 有着极为相似的形式，其中 $Y_j = a_j^T X$，是正规化的 X 分量的线性组合。实际上，对互熵的某些逼近和替代与所提出的投影寻踪的指标是一致的。典型地，对于投影研究，并不约束方向 a_j 为正交的。Friedman(1987)转换数据，以便在选择的投影中观察高斯曲线，而后搜索随后的方向。尽管起源不同，ICA 和探测性投影寻踪是相当形似的，至少在这里描述的表示中二者极为相似。

14.6.4　一种不同的 ICA 方法

根据定义，独立成分有一个联合积密度，所以为求解它们，可以估计它们的积密度。利用第 14.2.4 节中介绍的技巧，通过把密度估计任务看做一个 2-类的分类问题，将问题简化。观测的数据点被指派为类 $G = 1$，背景样本由一个密度 $g_0(x)$ 生成，并指派为类 $G = 0$。为举例说明，考虑一个双变量问题 $X = (X_1, X_2)$，以及如下形式的二项模型：

$$\log \frac{\Pr(G=1)}{1-\Pr(G=1)} = f_1(a_1^T X) + f_2(a_2^T X) \tag{14.67}$$

根据第 14.2.4 节，该分对数的加法模型给出如下形式的数据密度：

$$\begin{aligned} g(X) &= g_0(X) \cdot \exp\{f_1(a_1^T X) + f_2(a_2^T X)\} \\ &= g_0(X) \cdot \exp\{f_1(a_1^T X)\} \cdot \exp\{f_2(a_2^T X)\} \end{aligned} \tag{14.68}$$

我们寻求独立的成分 $a_1^T X$ 和 $a_2^T X$，因此有联合密度，其分解式为：

$$h(a_1^T X, a_2^T X) = h_1(a_1^T X) \cdot h_2(a_2^T X) \tag{14.69}$$

从式(14.68)到式(14.69)需要改变变量，并且容易看出必须把$g_0(a_1^T X, a_2^T X)$分解为一个乘积。这只有在如下情况下才出现：g_0 是多元高斯分布的密度，并且在 X 的协方差矩阵 $\mathbf{\Sigma}$ 的逆的度量下，a_1 正交于 a_2。与前面相同，首先变换数据形式，使它们具有单位协方差，这允许我们对 g_0 使用球形高斯分布。

因此，为了使用该过程，我们要将观测数据球面化，再从球面高斯分布生成背景数据，然后利用 a_1 正交于 a_2 的约束去拟合模型(14.67)。

模型(14.67)是逻辑斯缔回归的一种拓广，因此可以用局部评分算法(9.2)来拟合。该模型等式的右边具有投影寻踪回归的形式，所以使用局部评分的 PPR 算法(11.2)中步骤 2(c)。我们把这种方法试用于图 14.27 的人工时间序列数据和图 14.28 的混合均匀分布数据，每一种情况的结果成分 $\hat{a}_1^T X$ 和 $\hat{a}_2^T X$ 如图 14.31 所示，并且产生了原始的源信号。

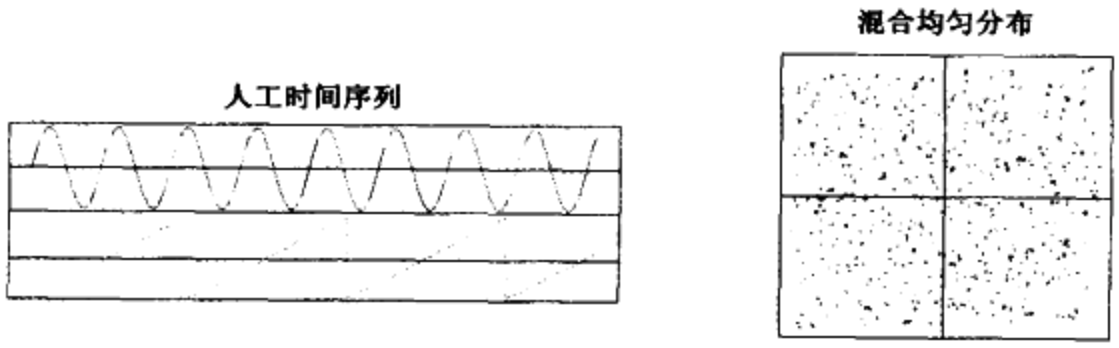


图 14.31 图 14.27 人工时间序列问题的投影 - 寻求解(左图)，以及图 14.28 的混合均匀分布(右图)。这些解是利用广义逻辑斯缔回归模型(14.67)得到的

像 ICA 一样，上述过程通过发现数据的正交、非高斯投影而发现独立成分。它有明显的优点：不需要选择非正态指标，而是利用背景高斯数据判断投影的非正态性。然而，给定投影时，在度量原始数据与背景数据之间分割的二项式散离损失函数中，指标是隐含的(见习题 14.12)。该过程对于此处所给的例子效果不错，但在本书编写时它还没有做大规模测试。

14.7 多维定标

自组织映射和主曲线、主曲面都是将 IR^p 中的数据点映射到一个低维的流形中。多维定标(*multidimensional scaling*, MDS)也有类似的目标，但却用了稍微不同的方式来解决问题。

从观测 $x_1, x_2, \cdots, x_N \in \mathrm{IR}^p$ 开始，并令 d_{ij} 为观测 i 和 j 之间的距离。通常，我们选择欧氏距离 $d_{ij} = \| x_i - x_j \|$，但是也可以用其他距离。更进一步，在某些应用中我们甚至没有可利

用的数据点 x_i，只有某种相异度 d_{ij}（见第 14.3.10 节）。例如，在品酒实验中，d_{ij} 可能是判断被测的酒 i 和酒 j 有多大不同的度量，品酒人需要为每一对酒 i 和 j 提供这样一个度量值。MDS 只要求相异度 d_{ij}，相比之下，SOM 和主曲线、主曲面则需要数据点 x_i。

多维定标寻求值 $z_1, z_2, \cdots, z_N \in \mathbb{R}^k$，以极小化应力函数（stress function）：

$$S_D(z_1, z_2, \cdots, z_N) = \left[\sum_{i \neq i'} (d_{ii'} - ||z_i - z_{i'}||)^2\right]^{1/2} \tag{14.70}$$

这就是众所周知的最小二乘方（least squares）或 Kruskal -Shephard定标。主要思想是找数据的一个低维近似，以尽可能地保持每对观测之间的距离。注意，该近似是根据距离而不是距离的平方，外面的平方根只是一个习惯表示。利用梯度下降算法极小化 S_D。

最小二乘方定标的一个变种即Sammon 映射（Sammon mapping），它极小化：

$$\sum_{i \neq i'} \frac{(d_{ii'} - ||z_i - z_{i'}||)^2}{d_{ii'}} \tag{14.71}$$

这里更强调了要保持较小的逐对距离。

在经典定标（classical scaling）中，我们是从相似度 $s_{ii'}$ 开始的：通常是利用中心化的内积 $s_{ii'} = \langle x_i - \bar{x}, x'_i - \bar{x} \rangle$。这样问题就成为在 $z_1, z_2, \cdots, z_N \in \mathbb{R}^k$ 上极小化：

$$\sum_{i \neq i'} (s_{ii'} - \langle z_i - \bar{z}_i, z'_i - \bar{z}'_i \rangle)^2 \tag{14.72}$$

这一点非常诱人，因为存在一个用本征向量表示的显式解（见习题 14.8）。经典定标不等价于最小二乘方定标，因为内积依赖于原点的选择，而逐对点的距离不依赖原点的选择。内积的集合决定逐对数据点间距离的集合，但反之则不然。

最小二乘方定标和经典定标在以下意义下称做度量定标（metric scaling）方法：它们都产生实际相异度或相似度的近似。Shephard -Kruskal 非度量定标（nonmetric scaling）只有效地使用秩。非度量定标试图在 $d_{ii'}$ 和一个任意增函数 $\theta(\cdot)$ 上极小化应力函数：

$$\frac{\sum_{i,i'} [\theta(||z_i - z_{i'}||) - d_{ii'}]^2}{\sum_{i,i'} d_{ii'}^2} \tag{14.73}$$

固定 $\theta(\cdot)$，我们通过梯度下降在 $d_{ii'}$ 上极小化应力函数。固定 $d_{ii'}$，保序回归（isotonic regression）方法用于求解最佳单调逼近。迭代地执行这些步骤，直到解稳定。

与自组织映射一样，多维定标将数据投影到较低维的流形上，但并不给出流形的参数，主曲面也是同样。在主曲面和*SOM*中，原始特征空间中紧凑的点被紧凑映射到流形上，但是特征空间中远距的点也可能映射到一起。而在多维定标中多半不会这样，因为它明确地试图保持所有的逐对距离。

图 14.32 显示的是来自经典定标对半球面例子的前两个*MDS*坐标。簇间有清晰的分割，并且红色簇的紧凑特性是显而易见的。

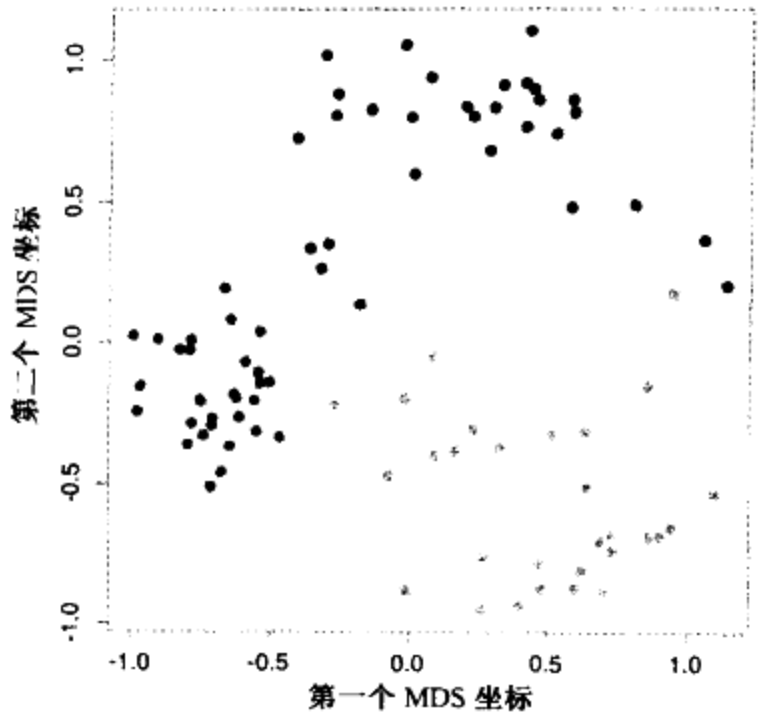


图 14.32 来自经典定标对半球面数据的前两个坐标(见彩页)

文献注释

关于聚类方面有很多书籍,包括 Hartigan(1975)、Gordon(1999)以及 Kaufman 和 Rousseeuw(1990)。K-均值聚类至少追溯到 Lloyd(1957)、Forgy(1965)、Jancey(1966)和 MacQueen(1967)。在工程应用方面,特别是在通过向量量化的图像压缩方面的内容,可以在 Gersho 和 Gray(1992)中找到。k-中心点过程在 Kaufman 和 Rousseeuw(1990)中讲述。关联规则在 Agrawal 等人(1995)的论文中有概述。自组织映射由 Kohonen(1989)和 Kohonen(1990)提出,Kohonen 等人(2000)有较新的解释。主成分分析和多维定标在有关多元分析的权威书籍中都有介绍,例如 Mardia 等人(1979)的著作。Buja 等人(1999)实现了一个称为 XGvis 的功能强大的环境,用于多维定标,而且其用户手册中包括了一个清晰的主题综述。图 14.17、图 14.21(左图)和图 14.26(左图)均由 XGobi 产生,它是同一组作者开发的一个多维数据可视化程序包。主曲线和曲面在 Hastie(1984)以及 Hastie 和 Stuetzle(1989)中提出,主点的思想在 Flury(1990)中被形式化,在 Tarpey 和 Flury(1996)中给出了自相容性通用概念的阐述。独立成分分析由 Comon(1994)提出,并且由 Bell 和 Sejnowski(1995)继续研究;第 14.6 节中的处理方法是基于 Hyvärinen 和 Oja(2000)。投影寻踪由 Friedman 和 Tukey(1974)提出,其在 Huber(1985)中有详细论述。XGobi 系统实现了一种动态投影寻踪算法。

习题

14.1 聚类加权(*weights for clustering*)。证明加权的欧氏距离

$$d_e^{(w)}(x_i, x_{i'}) = \frac{\sum_{l=1}^{p} w_l (x_{il} - x_{i'l})^2}{\sum_{l=1}^{p} w_l}$$

满足:

$$d_e^{(w)}(x_i, x_{i'}) = d_e(z_i, z_{i'}) = \sum_{l=1}^{p} (z_{il} - z_{i'l})^2 \tag{14.74}$$

其中：

$$z_{il} = x_{il} \cdot \left(\frac{w_l}{\sum_{l=1}^{p} w_l} \right)^{1/2} \tag{14.75}$$

因此，x 上的加权欧氏距离等价于 z 上的未加权的欧氏距离。

14.2 考虑 p 维特征空间中的混合模型密度：

$$g(x) = \sum_{k=1}^{K} \pi_k g_k(x) \tag{14.76}$$

其中，$g_k = N(\mu_k, \mathbf{I} \cdot \sigma^2)$，并且对于任意 k，$\pi_k \geqslant 0$，$\sum_k \pi_k = 1$。这里，$\{\mu_k, \pi_k\}$ $(k = 1, \cdots, K)$，σ^2 是未知参数。

假设我们有数据 $x_1, x_2, \cdots, x_N \sim g(x)$，并且希望拟合该混合模型。

1. 写出数据的对数似然。
2. 导出计算极大似然估计的 EM 算法(见第 8.1 节)。
3. 证明：如果在混合模型中 σ 的值已知，并且令 $\sigma \to 0$，那么在某种意义上该 EM 算法与 K-均值聚类算法是一致的。

14.3 说明在什么情况下，K-均值过程可以看做应用于混合高斯密度模型的 EM 算法(见第 8 章)的一个特例。

14.4 用分类树对表 14.1 中的人口统计数据聚类。特殊地，通过随机排列每个特征中的值，生成一个与训练集相同规模的参考样本。构建训练样本(类 1)和参考样本(类 0)的一个分类树，并且描述对估计类 1 概率最高的终端节点。对于相同的数据，将该结果分别与表 14.1 附近的 PRIM 的结果以及 K-均值聚类的结果进行比较。

14.5 生成数据，要求有三个特征，并在如下列出三个类的每个类中含有 30 个数据点

$$\begin{aligned}
\theta_1 &= U(-\pi/8, \pi/8) \\
\phi_1 &= U(0, 2\pi) \\
x_1 &= \sin(\theta_1)\cos(\phi_1) + W_{11} \\
y_1 &= \sin(\theta_1)\sin(\phi_1) + W_{12} \\
z_1 &= \cos(\theta_1) + W_{13} \\
\\
\theta_2 &= U(\pi/2 - \pi/4, \pi/2 + \pi/4) \\
\phi_2 &= U(-\pi/4, \pi/4) \\
x_2 &= \sin(\theta_2)\cos(\phi_2) + W_{21} \\
y_2 &= \sin(\theta_2)\sin(\phi_2) + W_{22} \\
z_2 &= \cos(\theta_2) + W_{23} \\
\\
\theta_3 &= U(\pi/2 - \pi/4, \pi/2 + \pi/4) \\
\phi_3 &= U(\pi/2 - \pi/4, \pi/2 + \pi/4) \\
x_3 &= \sin(\theta_3)\cos(\phi_3) + W_{31} \\
y_3 &= \sin(\theta_3)\sin(\phi_3) + W_{32} \\
z_3 &= \cos(\theta_3) + W_{33}
\end{aligned}$$

这里，$U(a, b)$ 表示 $[a, b]$ 范围内一个均匀分布的随机变量，W_{jk} 是独立的正态随机变量，其标准差是 0.6。因此数据在接近球形曲面处分别以(1,0,0)、(0,1,0)和(0,0,1)为中心聚为三个簇。

利用本章中给出的学习率编写程序，用 SOM 拟合这些数据。再对相同数据执行 K-均

值聚类，并把这些结果与文中的结果进行比较。

14.6 利用基于二维网格的原型，编写程序实现 K-均值聚类和自组织映射(SOM)。把两个程序应用于人体肿瘤微阵列数据的聚类，这里使用形心 $K=2,5,10,20$。并论证随着 SOM 邻域规模越来越小，SOM 的解越来越接近 K-均值的解。

14.7 推导第 14.5.1 节中的式(14.51)和式(14.52)。证明 $\hat{\mu}$ 是不惟一的，并描述等价解的族特征。

14.8 经典多维定标。令 $\mathbf{S}$ 是元素为$\langle x_i-\bar{x}, x_j-\bar{x}\rangle$的中心化的内积矩阵。令 $\lambda_1>\lambda_2\cdots>\lambda_k$ 为 $\mathbf{S}$ 的 k 个最大本征值，并有相关联的本征向量 $\mathbf{E}_k=(e_1,e_2,\cdots,e_k)$。令 $\mathbf{D}_k$ 是一个对角矩阵，对角元素为$\sqrt{\lambda_1},\sqrt{\lambda_2},\cdots,\sqrt{\lambda_k}$。

证明经典定标问题(14.72)的解 z_i 是矩阵 $\mathbf{E}_k\mathbf{D}_k$ 的行。

14.9 生成 200 个数据点，具有三个特征，位置邻近于一个螺旋线(helix)。详细的要求为：定义 $X_1=\cos(s)+0.1\cdot Z_1$，$X_2=\sin(s)+0.1\cdot Z_2$，$X_3=s+0.1\cdot Z_3$，其中 s 在 0 和 2π 之间等距地取 200 个值，Z_1、Z_2、Z_3 是独立的并具有标准高斯分布。

(a) 用主曲线拟合数据，并绘制估计的坐标函数。将它们与基本函数 $\cos(s)$、$\sin(s)$以及 s 进行比较。

(b) 用一个自组织映射拟合相同数据，并观察是否能够发现原始点云的螺旋形状。

14.10 用一个包含 X_j 的逆方差的对角矩阵前乘、后乘式(14.60)，由此获得相关矩阵的一个等价分解，在某种意义上是把一个简单的定标应用于矩阵 $\mathbf{A}$。

14.11 根据下式的描述生成三个随机变量 X_1、X_2、X_3 的 200 个观测点：

$$\begin{aligned} X_1 &\sim Z_1 \\ X_2 &= X_1+.001\cdot Z_2 \\ X_3 &= 10\cdot Z_3 \end{aligned} \tag{14.77}$$

其中 Z_1、Z_2、Z_3 是独立的标准正态随机变量。计算第一主成分和因子分析方向，并由此证明第一主成分以最大方差方向 X_3 排列，而第一因子却基本上忽略不相关成分 X_3，并获得相关成分 X_2+X_1[Geoffrey Hinton，个人通讯]。

14.12 ICA 和投影寻踪(见第 14.6.4 节)

(a) 假定在点 X 的真实密度是$f(X)$，并记高斯密度为$g(X)$。证明随机选择一个点来自 f 概率是：

$$p(X)=\Pr(Y=1|X)=\frac{f(X)}{f(X)+g(X)}$$

并且其分对数是：

$$\text{logit}(p(X))=\log(f(X))-\log(g(X))$$

(b) 证明模型 $\tilde{p}(X)$的期望二项式对数似然为：

$$E_X\left[p(X)\log(\tilde{p}(X))+(1-p(X))\log(1-\tilde{p}(X))\right]$$

(c) 假定 $\tilde{p}$ 被非参数地建模，并且接近真实的 p，我们则用 p 替代它。

因此有非正态的指标：

$$\kappa(p) = E_X p(X)\log(p(X)) + (1 - p(X))\log(1 - p(X))$$

并且我们寻求投影方向 a 以极小化 $\kappa(p_a)$，其中 $p_a(X) = p(a^T X)$。

既然当 p_a 是 0 或 1(极值)时 $\kappa(p_a)$ 最大，所以 $\text{logit}(p_a(X)) = \log(f_a(X)) - \log(g_a(X))$ 是绝对值大的。因此，通过关于 a 极大化该似然，我们能够找到一个方向，使得以二项式熵来度量，对该方向投影的数据尽可能地背离高斯密度。

在泰勒级数 $p_a = 0.5$ 附近扩展 κ，以证明：

$$\kappa(p_a) \approx E_X \frac{f_a(X)g_a(X)}{f_a(X) + g_a(X)} [\log(f_a(X)) - \log(g_a(X))]^2$$

比较上式右边非正态的度量与在第 14.6.2 节讨论的非正态的度量。

术 语 表

A

AIC, see Akaike information criterion	AIC, 见 Akaike 信息准则
activation function	激活函数
adaptive method	自适应方法
adaptive nearest neighbor method	自适应最近邻方法
adaptive wavelet filtering	自适应小波过滤
additive model	加法模型
adjusted response	调整的响应
affine invariant average	仿射不变量平均
affine set	仿射集
Akaike information criterion (AIC)	Akaike 信息准则
analysis of deviance	散离分析
application	应用
aorta	动脉
bone	骨质
California housing	加利福尼亚住房
country	国家
document	文档
galaxy	星系
heart attack	心脏病
marketing	销售
microarray	微阵列
nuclear magnetic resonance	核磁共振
ozone	臭氧
prostate cancer	前列腺癌
satellite image	卫星图像
spam	垃圾邮件
vowel	元音
waveform	波形
ZIP code	邮政编码
association rule	关联规则
automatic selection of smoothing parameters	光滑参数的自动选择

B

BIC, see Bayesian Information Criterion	BIC, 见贝叶斯信息准则
B -Spline	B 样条
back -propagation	反向传播
backfitting procedure	反向拟合过程
backward pass	后向传递
backward stepwise selection	逐步后向选择
bagging	装袋
basis expansions and regularization	基展开和正则化
basis functions	基函数
batch learning	批学习
Baum -Welch algorithm	Baum -Welch 算法
Bayes	贝叶斯
classifier	贝叶斯分类器
factor	贝叶斯因子
methods	贝叶斯方法
rate	贝叶斯率
Bayesian information criterion (BIC)	贝叶斯信息准则
between -class covariance matrix	类间协方差矩阵
bias	偏倚, 偏置
bias -variance decomposition	偏倚 – 方差分解
bias -variance tradeoff	偏倚 – 方差折中/权衡
boosting	提升
bootstrap	自助法
relationship to maximum likelihood method	自助法与极大似然的关系
relationship to Bayesian method	自助法与贝叶斯方法的关系
bottom -up clustering	自下而上聚类
bump hunting, see patient rule induction method (PRIM)	凸点搜索, 见忍耐规则归纳方法
bumping	冲击

C

CART, see classification and regression tree	CART , 见分类和回归树
canonical variable	标准/典范变量
categorical predictor	分类的预测子
classical multidimensional scaling	经典多维定标
classification and regression trees(CART)	分类和回归树
clustering	聚类
agglomerative	凝聚的聚类
hierarchical	分层, 分级聚类

K -means	*K* -均值聚类
codebook	编码本
combinatorial algorithm	组合算法
combining model	组合模型
committee method	委员会方法
complete data	完全数据
complexity parameter	复杂性参数
comparison of learning methods	学习方法比较
condensing procedure	凝聚过程
conditional likelihood	条件似然
conjugate gradients	共轭梯度
confusion matrix	混淆矩阵
convolutional networks	卷积网络
cost complexity pruning	代价复杂性剪枝
Cp statistic	*Cp* 统计量
cross -entropy	互熵
cross -validation	交叉验证
cubic smoothing spline	三次光滑样条
cubic spline	三次样条
curse of dimensionality	维灾难

D

data augmentation	数据增广
Daubechies symmlet -8 wavelets	Daubechies symmlet -8 小波
decision boundary	判定边界
decision tree	决策树
decoding step	解码步
degrees of freedom	自由度
in ridge regression	岭回归的自由度
of smoother matrices	光滑矩阵的自由度
of a tree	树的自由度
in an additive model	加法模型的自由度
Delta rule	Delta 规则
Demmler -Reinsch basis for splines	样条函数的 Demmler -Reinsch 基
density estimation	密度估计
deviance	散离
discrete variables	离散变量
discriminant adaptive nearest neighbor (DANN) classifier	判别自适应最近邻分类器/法
discriminant	判别式
analysis	判别分析

coordinate	判别坐标
function	判别函数
dissimilarity measure	相异性度量
dummy variables	哑变量

E

early stopping	提前停止
effective degrees of freedom	有效自由度
effective number of parameters	有效的参数个数
eigenvalues of a smoother matrix	光滑矩阵本征值
expectation -maximization algorithm, see EM algorithm	期望极大化算法,见 EM 算法
EM algorithm	EM 算法
for two component Gaussian mixture	二分量高斯混合 EM 算法
as a maximization -maximization procedure	EM 算法作为极大化 – 极大化过程
encoder	编码器
entropy	熵
equivalent kernel	等价核
error rate	误差率
estimates of in -sample prediction error	样本内预测误差估计
exponential loss and AdaBoost	指数损失和 AdaBoost
extra -sample error	样本外误差

F

features	特征
feature extraction	特征提取
feed -forward neural networks	前馈神经网络
Fisher's linear discriminant	Fisher 线性判别
flexible discriminant analysis	柔性判别分析
forward selection	前向选择
forward stagewise additive modeling	前向分步加法建模
Fourier transform	傅里叶变换
frequentist methods	频率论方法
function approximation	函数逼近

G

GCV, see Generalized cross -validation	GCV,见广义交叉验证
GEM (generalized EM)	GEM(广义 EM)
gap statistic	间隙统计
gating network	门控网络
Gaussian (normal) distribution	高斯(正态)分布
Gauss -Markov theorem	高斯 – 马尔可夫定理

Gauss -Newton method	高斯 - 牛顿方法
Gaussian mixture	高斯混合
Gaussian radial basis function	高斯径向基函数
generalization	泛化,一般化
error	泛化误差
performance	泛化性能
generalized additive model	广义加法模型
generalized association rules	广义关联规则
generalized cross -validation	广义交叉验证
generalized linear models	广义线性模型
generalizing linear discriminant analysis	广义线性判别分析
Gibbs sampler	Gibbs 抽样法
Gibbs sampler for mixtures	混合 Gibbs 抽样法
Gini index	Gini 索引
global dimension reduction for nearest neighbors	最近邻整体维归约
gradient boosting	梯度提升
gradient descent	梯度下降

H

Haar basis function	Haar 基函数
hat matrix	帽矩阵
Hessian matrix	Hessian 矩阵
helix	螺旋线
hidden units	隐藏单元
hierarchical clustering	分层/级聚类
hierarchical mixtures of experts	分层/级专家混合
hints	线索,提示
hyperplane	超平面
separating	分离超平面

I

ICA, see independent components analysis	ICA,见独立成分分析
IRLS, see iteratively reweighted least squares	IRLS,见迭代加权最小二乘方
in -sample prediction error	样本内预测误差
incomplete data	非完全数据
independent variables	独立变量
independent components analysis	独立成分分析
indicator response matrix	指示器/子响应矩阵
inference	推理
information	信息

Fisher	费希尔信息
observed	观测信息
information theory	信息论
inputs	输入
instability of trees	树的不稳定性
intercept	截距
invariance manifold	不变流形
invariant metric	不变度量
inverse wavelet transform	逆小波变换
irreducible error	不可约误差
iteratively reweighted least squares (IRLS)	迭代加权最小二乘方

J

Jensen 不等式	Jensen's inequality

K

K -means clustering	K -均值聚类
k -medoid clustering	k -中心点聚类
k -nearest neighbor classifiers	k -最近邻分类法/器
Karhunen -Loeve transformation (principal components)	Karhunen -Loeve 变换(主成分)
kernel density classification	核密度分类
kernel density estimation	核密度估计
kernel function	核函数
kernel method	核方法
knot	纽结
Kriging	克瑞精
Kruskal -Shephard scaling	Kruskal -Shephard 定标
Kullback -Leibler distance	Kullback -Leibler 距离
Karush -Kuhn -Tucker conditions	Karush -Kuhn -Tucker 条件

L

LVQ, see Learning Vector Quantization	LVQ,见学习向量量化
Lagrange multipliers	拉格朗日乘子
Laplacian distribution	拉普拉斯分布
lasso	套索
learning	学习
learning rate	学习率
Learning Vector Quantization	学习向量量化
least squares	最小二乘方
leave -one -out cross -validation	留一交叉验证

left singular vector	左奇异向量
LeNet	LeNet
likelihood function	似然函数
linear basis expansion	线性基展开式
linear combination splits	线性组合分裂
linear discriminant function	线性判别函数
linear methods	线性方法
for classification	分类的线性方法
for regression	回归的线性方法
linear models and least squares	线性模型与最小二乘方
linear regression of an indicator matrix	指示矩阵的线性回归
linear separability	线性可分性
linear smoother	线性光滑法/子
link function	链接函数
local likelihood	局部似然
local methods in high dimensions	高维的局部方法
local minima	局部最小值
local polynomial regression	局部多项式回归
local regression	局部回归
localization in time and in frequency	时间和频率的局部性
Loess (local regression)	局部回归
log-odds ratio (logit)	对数几率(分对数)
logistic (sigmoid) function	逻辑斯缔(S 型)函数
logistic regression	逻辑斯缔回归
loss (log-odds ratio)	损失(对数几率)
loss function	损失函数
loss matrix	损失矩阵
lossless compression	无损压缩
lossy compression	有损压缩

M

MAP (maximum aposteriori) estimate	MAP(最大后验)估计
MARS, see multivariate adaptive regression splines	MARS,见多元自适应回归样条
MART, see Multiple additive regression trees	MART,见多元加法回归树
MCMC, see Markov Chain Monte Carlo methods	MCMC,见马尔可夫链蒙特卡罗方法
MDL, see Minimum description length	MDL,见最小描述长度
Mahalanobis distance	Mahalanobis 距离
majority vote	多数表决
margin	边缘
market basket analysis	购物篮分析

maximum likelihood estimation	极大似然估计
maximum likelihood inference	极大似然推理
Markov chain Monte Carlo (MCMC) methods	马尔可夫链蒙特卡罗方法
mean squared error	均方误差
memory -based method	基于内存的方法
Metropolis -Hasting algorithm	Metropolis -Hasting 算法
minimum description length (MDL)	最小描述长度
misclassification error	误分类误差
missing predictor values	遗漏预测值
missing data	遗漏数据
mixing proportions	混合比例
mixture discriminant analysis	混合判定分析
mixture modeling	混合建模
mixture of experts	专家混合
mixture and the EM algorithm	混合与 EM 算法
mode seekers	众数搜索
model averaging and stacking	模型平均和堆栈
model combination	模型组合
model complexity	模型复杂性
model selection	模型选择
Monte Carlo method	蒙特卡罗方法
mother wavelet	母小波
multi -dimensional splines	多维样条函数
multi -edit algorithm	多编辑算法
multi -resolution analysis	多分辨率分析
multidimensional scaling	多维定标
multi -layer perceptron	多层感知器
multinomial distribution	多项式分布
multiple minima	多极小值
multiple outcome shrinkage and selection	多结果收缩与选择
multiple outputs	多输出
multiple regression from simple univariate regression	从简单的一元回归到多元回归
multivariate adaptive regression splines (MARS)	多元加法回归样条
multiple additive regression trees(MART)	多元自适应回归树
multivariate nonparametric regression	多元非参数回归

N

Nadaraya -Watson estimate	Nadaraya -Watson 估计
naive Bayes classifier	朴素贝叶斯分类
natural cubic splines	自然三次样条函数

nearest neighbor methods	最近邻方法
network diagram	网络图
neural network	神经网络
Newton's method (Newton -Raphson procedure)	牛顿方法(牛顿 - 拉斐桑过程)
nonparametric logistic regression	非参数逻辑斯缔回归
normal (Gaussian) distribution	正态(高斯)分布
normal equations	法方程,正规方程
numerical optimization	数值优化

O

object dissimilarity	对象相异性
online algorithm	联机算法,在线算法
optimal scoring	最优得分
optimal separating hyperplane	最佳分离超平面
optimism of the training error rate	训练误差率的最优化
ordered categorical (ordinal) predictor	有序的分类预测子
orthogonal predictors	正交预测子
overfitting	过分拟合

P

PRIM, see patient rule induction method	PRIM,见忍耐规则归纳方法
parametric bootstrap	参数自助法
partial dependence plots	部分/偏依赖图
partial least squares	部分最小二乘方
Parzen window	Parzen 窗口
pasting	粘贴
patient rule induction method (PRIM)	忍耐规则归纳方法
peeling	删除,去除
penalization, see regularization	惩罚,见正则化
penalized discriminant analysis	罚判别分析
penalized polynomial regression	罚多项式回归
penalized regression	罚回归
penalty matrix	罚矩阵
perceptron	感知器
piercewise polynomials and splines	分段多项式和样条
posterior	后验(的)
distribution	后验分布
probability	后验概率
prediction accuracy	预测精度
predictive distribution	预测分布

prediction error	预测误差
principal components	主成分
principal components regression	主成分回归
principal curves and surfaces	主曲线和曲面
principal points	主点
prior distribution	先验分布
projection pursuit	投影寻踪
projection pursuit regression	投影寻踪回归
prototype classifier	原型分类法/器
prototype method	原型方法
proximity matrices	临近矩阵
pruning	剪枝

Q

QR decomposition	QR 分解
quadratic approximations and inference	二次逼近和推理
quadratic discriminant function	二次判别函数

R

radial basis function	径向基函数
radial basis function(RBF)network	径向基函数网络
Rao score test	Rao 得分检验
Rayleigh quotient	Rayleigh 商
receiver operating characteristic (ROC) curve	接收机工作特征曲线
reduced -rank linear discriminant analysis	降秩线性判别分析
regression	回归
regression spline	回归样条
regularization	正则化
regularized discriminant analysis	正规化判别分析
representer of evaluation	估值表示
reproducing kernel Hilbert space	再生核希尔伯特空间
reproducing property	再生性
responsibilities	响应度
ridge regression	岭回归
risk factor	风险因素/子
robust fitting	健壮拟合
Rosenblatt's perceptron learning algorithm	Rosenblatt 感知器学习算法
rug plot	底线图

S

SOM, see self -organizing map	SOM,见自组织映射

SRM, see structural risk minimization	SRM, 见结构风险最小化
SURE shrinkage method	SURE 收缩方法
SVD, see singular value decomposition	SVD, 见奇异值分解
Sammon mapping	Sammon 映射
scaling of the inputs	输入定标
Schwartz's criterion	Schwartz 准则
score equations	得分方程
self-consistency property	自相容性质
self-organizing map (SOM)	自组织映射
sensitivity of a test	检验的敏感性
separating hyperplanes	分离超平面
shape averaging	外形平均
shrinkage methods	收缩方法
Sigmoid	S 型
similarity measure, see dissimilarity measure	相似性度量, 见相异性度量
single index model	单索引模型
singular value decomposition (SVD)	奇异值分解
singular values	奇异值
skin of the orange example	橘子皮例子
sliced inverse regression	切片逆回归
smoother	光滑法, 光滑器
smoother matrix	光滑矩阵
smoothing parameter	光滑参数
smoothing spline	光滑样条
soft clustering	软聚类
Softmax function	Softmax 函数
sparseness	稀疏性
specificity of a test	检验的特效性
spline	样条
additive	加法样条
cubic smoothing	三次光滑样条
cubic	三次样条
interaction	交互样条
regression	回归样条
smoothing	光滑样条
thin plate	薄板样条
squared error loss	平方误差损失
stacking (stacked generalization)	堆栈(堆栈泛化)
starting values	初始值

statistical decision theory	统计判决理论
statistical model	统计模型
steepest descent	最速下降
stochastic approximation	随机逼近
stochastic search (bumping)	随机搜索(冲击)
stress function	应力函数
structural risk minimization (SRM)	结构风险最小化
subset selection	子集选择
supervised learning	有指导学习
support vector classifier	支持向量分类法/器
support vector machine	支持向量机
Symmlet basis	Symmlet 基

T

tangent distance	切距离
Tanh activation function	Tanh 激活函数
target variable	目标变量
tensor product basis	张量积基
test set	检验集
test error	检验误差
thin plate spline	薄板样条
thinning strategy	稀释策略
trace of a matrix	矩阵的迹
training epoch	训练周期
training error	训练误差
training set	训练集
tree -based methods	基于树的方法
trees for classification	分类树
tree for regression	回归树
Trellis display	Trellis 显示

U

universal approximator	普适逼近子
unsupervised learning	无指导学习
unsupervised learning as supervised learning	作为有指导学习的无指导学习

V

VC dimension, see Vapnik -Chernovenkis dimension	VC 维,见 Vapnik - Chernovenkis 维
validation set	验证集
Vapnik -Chernovenkis (VC) dimension	Vapnik -Chernovenkis 维
variable types and terminology	变量类型和术语

variance	方差
between	类间方差
within	类内方差
varying coefficient models	变系数模型
vector quantization	向量量化
Voronoi regions	Voronoi 区域

W

Wald test	Wald 检验
wavelet basis function	小波基函数
wavelet smoothing	小波光滑
wavelet transform	小波变换
weak learner	弱学习器
weakest link pruning	最弱链接剪枝
weights in a neural network	神经网络中的权
weight decay	权衰减
weight elimination	权重消除
within-class covariance matrix	类内协方差矩阵

参考文献

Abu-Mostafa, Y. (1995). Hints, *Neural Computation* **7**: 639–671.

Agrawal, R., Mannila, H., Srikant, R., Toivonen, H. and Verkamo, A. I. (1995). Fast discovery of association rules, *Advances in Knowledge Discovery and Data Mining*, AAAI/MIT Press, Cambridge, MA.

Akaike, H. (1973). Information theory and an extension of the maximum likelihood principle, *Second International Symposium on Information Theory*, pp. 267–281.

Allen, D. (1977). The relationship between variable selection and data augmentation and a method of prediction, *Technometrics* **16**: 125–7.

Anderson, J. and Rosenfeld, E. (eds) (1988). *Neurocomputing: Foundations of Research*, MIT Press, Cambridge, MA.

Barron, A. (1993). Universal approximation bounds for superpositions of a sigmoid function, *IEEE transactions on Information Theory* **39**: 930–945.

Becker, R., Cleveland, W. and Shyu, M. (1996). The visual design and control of trellis display, *Journal of Computational and Graphical Statistics* **5**: 123–155.

Bell, A. and Sejnowski, T. (1995). An information-maximization approach to blind separation and blind deconvolution, *Neural Computation* **7**: 1129–1159.

Bellman, R. E. (1961). *Adaptive Control Processes*, Princeton University Press.

Bishop, C. (1995). *Neural Networks for Pattern Recognition*, Clarendon Press, Oxford.

Breiman, L. (1992). The little bootstrap and other methods for dimensionality selection in regression: X-fixed prediction error, *J. Amer. Statist. Assoc* **87**: 738–754.

Breiman, L. (1996a). Bagging predictors, *Machine Learning* **26**: 123–140.

Breiman, L. (1996b). Stacked regressions, *Machine Learning* **24**: 51–64.

Breiman, L. (1998). Arcing classifiers (with discussion), *Annals of Statistics* **26**: 801–849.

Breiman, L. (1999). Prediction games and arcing algorithms, *Neural Computation* pp. 1493–1517.

Breiman, L. and Friedman, J. (1997). Predicting multivariate responses in multiple linear regression (with discussion), *J. Roy. Statist. Soc. B.* **59**: 3–37.

Breiman, L., Friedman, J., Olshen, R. and Stone, C. (1984). *Classification and Regression Trees*, Wadsworth.

Breiman, L. and Ihaka, R. (1984). Nonlinear discriminant analysis via scaling and ACE, *Technical report*, Univ. of California, Berkeley.

Breiman, L. and Spector, P. (1992). Submodel selection and evaluation in regression: the X-random case, *Intern. Statist. Rev* **60**: 291–319.

Bruce, A. and Gao, H. (1996). *Applied Wavelet Analysis with S-PLUS*, Springer.

Buja, A., Hastie, T. and Tibshirani, R. (1989). Linear smoothers and additive models (with discussion), *Annals of Statistics* **17**: 453–555.

Buja, A., Swayne, D., Littman, M. and Dean, N. (1999). Xgvis: A system for multidimensional scaling and graph layout in any dimension, *Technical report*, AT&T Laboratories.

Burges, C. J. C. (1998). A Tutorial on Support Vector Machines for Pattern Recognition, *Knowledge Discovery and Data Mining* **2**(2): 121–167.

Chambers, J. and Hastie, T. (1991). *Statistical Models in S*, Wadsworth/Brooks Cole, Pacific Grove, CA.

Cherkassky, V. and Mulier, F. (1998). *Learning from Data*, Wiley, New York.

Chui, C. (1992). *An Introduction to Wavelets*, Academic Press, London.

Comon, P. (1994). Independent component analysis — a new concept?, *Signal Processing* **36**: 287–314.

Copas, J. B. (1983). Regression, prediction and shrinkage (with discussion), *Journal of the Royal Statistical Society, Series B, Methodological* **45**: 311–354.

Cover, T. and Hart, P. (1967). Nearest neighbor pattern classification, *Proc. IEEE Trans. Inform. Theory* **IT-11**: 21–27.

Cover, T. and Thomas, J. (1991). *Elements of Information Theory*, Wiley, New York.

Cox, D. and Hinkley, D. (1974). *Theoretical Statistics*, Chapman and Hall, London.

Cressie, N. A. C. (1993). *Statistics for Spatial Data (Revised Edition)*, Wiley-Interscience, New York.

Csiszar, I. and Tusnády, G. (1984). Information geometry and alternating minimization procedures, *Statistics & Decisions Supplement Issue* **1**: 205–237.

Dasarathy, B. (1991). *Nearest Neighbor Pattern Classification Techniques*, IEEE Computer Society Press.

Daubechies, I. (1992). *Ten Lectures in Wavelets*, Society for Industrial and Applied Mathematics, Philadelphia, PA.

de Boor, C. (1978). *A Practical Guide to Splines*, Springer-Verlag, New York.

Dempster, A., Laird, N. and Rubin, D. (1977). Maximum likelihood from incomplete data via the EM algorithm (with discussion), *J. R. Statist. Soc. B.* **39**: 1–38.

Devijver, P. and Kittler, J. (1982). *Pattern Recognition: a Statistical Approach*, Prentice-Hall, Englewood Cliffs, N.J.

Donoho, D. and Johnstone, I. (1994). Ideal spatial adaptation by wavelet shrinkage, *Biometrika* **81**: 425–455.

Donoho, D., Johnstone, I., Kerkyachairan, G. and Picard, D. (1995). Wavelet shrinkage; asymptopia? (with discussion), *J. Royal. Statist. Soc.* **57**: 201–337.

Duan, N. and Li, K.-C. (1991). Slicing regression: a link-free regression method, *Annals of Statistics* **19**: 505–530.

Duchamp, T. and Stuetzle, W. (1996). Extremal properties of principal curves in the plane, *The Annals of Statistics* **24**: 1511–1520.

Duda, R., Hart, P. and Stork, D. (2000). *Pattern Classification (Second Edition)*, Wiley, New York.

Efron, B. (1975). The efficiency of logistic regression compared to normal discriminant analysis, *J. Amer. Statist. Assoc.* **70**: 892–898.

Efron, B. (1979). Bootstrap methods: another look at the jackknife, *Annals of Statistics* **7**: 1–26.

Efron, B. (1983). Estimating the error rate of a prediction rule: some improvements on cross-validation, *J. Amer. Statist. Assoc.* **78**: 316–331.

Efron, B. (1986). How biased is the apparent error rate of a prediction rule?, *J. Amer. Statist. Assoc.* **81**: 461–70.

Efron, B. and Tibshirani, R. (1991). Statistical analysis in the computer age, *Science* **253**: 390–395.

Efron, B. and Tibshirani, R. (1993). *An Introduction to the Bootstrap*, Chapman and Hall, London.

Efron, B. and Tibshirani, R. (1997). Improvements on cross-validation: the 632+ bootstrap: method, *J. Amer. Statist. Assoc.* **92**: 548–560.

Evgeniou, T., Pontil, M. and Poggio, T. (2000). Regularization networks

and support vector machines, *Advances in Computational Mathematics* **13**: 1–50.

Fan, J. and Gijbels, I. (1996). *Local Polynomial Modelling and Its Applications*, Chapman and Hall, London.

Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems, *Eugen.* **7**: 179–188.

Fix, E. and Hodges, J. (1951). Discriminatory analysis- nonparametric discrimination: Consistency properties, *Technical Report 21-49-004,4*, US Air Force, School of Aviation Medicine, Randolph Field, TX.

Flury, B. (1990). Principal points, *Biometrika* **77**: 33–41.

Forgy, E. (1965). Cluster analysis of multivariate data: Efficiency vs. interpretability of classifications (abstract), *Biometrics* **21**: 768–769.

Frank, I. and Friedman, J. (1993). A statistical view of some chemometrics regression tools (with discussion), *Technometrics* **35**(2): 109–148.

Freund, Y. (1995). Boosting a weak learning algorithm by majority, *Information and Computation* **121**(2): 256–285.

Freund, Y. and Schapire, R. (1996a). Experiments with a new boosting algorithm, *Machine Learning: Proceedings of the Thirteenth International Conference*, Morgan Kauffman, San Francisco, pp. 148–156.

Freund, Y. and Schapire, R. (1996b). Game theory, on-line prediction and boosting, *Proceedings of the Ninth Annual Conference on Computational Learning Theory*, pp. 325–332.

Freund, Y. and Schapire, R. (1997). A decision-theoretic generalization of online learning and an application to boosting, *Journal of Computer and System Sciences* **55**: 119–139.

Friedman, J. (1987). Exploratory projection pursuit, *Journal of the American Statistical Association* **82**: 249–266.

Friedman, J. (1989). Regularized discriminant analysis, *Journal of the American Statistical Association* **84**: 165–175.

Friedman, J. (1991). Multivariate adaptive regression splines (with discussion), *Annals of Statistics* **19**(1): 1–141.

Friedman, J. (1994a). Flexible metric nearest-neighbor classification, *Technical report*, Stanford University.

Friedman, J. (1994b). An overview of predictive learning and function approximation, *in* V. Cherkassky, J. Friedman and H. Wechsler (eds), *From Statistics to Neural Networks*, Vol. 136 of *NATO ISI Series F*, Springer Verlag, New York.

Friedman, J. (1996). Another approach to polychotomous classification, *Technical report*, Stanford University.

Friedman, J. (1997). On bias, variance, 0-1 loss and the curse of dimensionality, *J. Data Mining and Knowledge Discovery* **1**: 55–77.

Friedman, J. (1999). Stochastic gradient boosting, *Technical report*, Stanford University.

Friedman, J. (2001). Greedy function approximation: a gradient boosting machine, *Annals of Statistics* **29**(5).

Friedman, J., Baskett, F. and Shustek, L. (1975). An algorithm for finding nearest neighbors, *IEEE Transactions on Computers* **24**: 1000–1006.

Friedman, J., Bentley, J. and Finkel, R. (1977). An algorithm for finding best matches in logarithmic expected time, *ACM Transactions on Mathematical Software* **3**: 209–226.

Friedman, J. and Fisher, N. (1999). Bump hunting in high dimensional data, *Statistics and Computing* **9**: 123–143.

Friedman, J., Hastie, T. and Tibshirani, R. (2000). Additive logistic regression: a statistical view of boosting (with discussion), *Annals of Statistics* **28**: 337–407.

Friedman, J. and Silverman, B. (1989). Flexible parsimonious smoothing and additive modelling (with discussion), *Technometrics* **31**: 3–39.

Friedman, J. and Stuetzle, W. (1981). Projection pursuit regression, *Journal of the American Statistical Association* **76**: 817–823.

Friedman, J., Stuetzle, W. and Schroeder, A. (1984). Projection pursuit density estimation, *Journal of the American Statistical Association* **79**: 599–608.

Friedman, J. and Tukey, J. (1974). A projection pursuit algorithm for exploratory data analysis, *IEEE trans. on computers, Ser. C* **23**: 881–889.

Furnival, G. and Wilson, R. (1974). Regression by leaps and bounds, *Technometrics* **16**: 499–511.

Gelfand, A. and Smith, A. (1990). Sampling based approaches to calculating marginal densities, *J. Amer. Statist. Assoc.* **85**: 398–409.

Gelman, A., Carlin, J., Stern, H. and Rubin, D. (1995). *Bayesian Data Analysis*, CRC Press, Boca Raton, FL.

Geman, S. and Geman, D. (1984). Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **6**: 721–741.

Gersho, A. and Gray, R. (1992). *Vector Quantization and Signal Compression*, Kluwer Academic Publishers, Boston, MA.

Girosi, F., Jones, M. and Poggio, T. (1995). Regularization theory and neural network architectures, *Neural Computation* **7**: 219–269.

Golub, G., Heath, M. and Wahba, G. (1979). Generalized cross-validation as a method for choosing a good ridge parameter, *Technometrics* **21**: 215–224.

Golub, G. and Van Loan, C. (1983). *Matrix Computations*, Johns Hopkins University Press, Baltimore.

Gordon, A. (1999). *Classification (2nd edition)*, Chapman and Hall/CRC press, London.

Green, P. and Silverman, B. (1994). *Nonparametric Regression and Generalized Linear Models: A Roughness Penalty Approach*, Chapman and Hall, London.

Greenacre, M. (1984). *Theory and Applications of Correspondence Analysis*, Academic Press, New York.

Hall, P. (1992). *The Bootstrap and Edgeworth Expansion*, Springer-Verlag, New York.

Hand, D. (1981). *Discrimination and Classification*, Wiley, Chichester.

Hart, P. (1968). The condensed nearest-neighbor rule, *IEEE Trans. Inform. Theory* **14**: 515–516.

Hartigan, J. A. (1975). *Clustering Algorithms*, Wiley, New York.

Hartigan, J. A. and Wong, M. A. (1979). [(Algorithm AS 136] A k-means clustering algorithm (AS R39: 81v30 p355-356), *Applied Statistics* **28**: 100–108.

Hastie, T. (1984). Principal curves and surfaces, *Technical report*, Stanford University.

Hastie, T., Botha, J. and Schnitzler, C. (1989). Regression with an ordered categorical response, *Statistics in Medicine* **43**: 884–889.

Hastie, T., Buja, A. and Tibshirani, R. (1995). Penalized discriminant analysis, *Annals of Statistics* **23**: 73–102.

Hastie, T. and Herman, A. (1990). An analysis of gestational age, neonatal size and neonatal death using nonparametric logistic regression, *Journal of Clinical Epidemiology* **43**: 1179–90.

Hastie, T. and Simard, P. (1998). Models and metrics for handwritten digit recognition, *Statistical Science* **13**: 54–65.

Hastie, T. and Stuetzle, W. (1989). Principal curves, *Journal of the American Statistical Association* **84**(406): 502–516.

Hastie, T. and Tibshirani, R. (1987). Nonparametric logistic and proportional odds regression, *Applied Statistics* **36**: 260–276.

Hastie, T. and Tibshirani, R. (1990). *Generalized Additive Models*, Chapman and Hall, London.

Hastie, T. and Tibshirani, R. (1996a). Discriminant adaptive nearest-neighbor classification, *IEEE Pattern Recognition and Machine Intelligence* **18**: 607–616.

Hastie, T. and Tibshirani, R. (1996b). Discriminant analysis by Gaussian mixtures, *J. Royal. Statist. Soc. B.* **58**: 155–176.

Hastie, T. and Tibshirani, R. (1998). Classification by pairwise coupling, *Annals of Statistics* **26**(2).

Hastie, T., Tibshirani, R. and Buja, A. (1994). Flexible discriminant analysis by optimal scoring, *J. Amer. Statist. Assoc.* **89**: 1255–1270.

Hastie, T., Tibshirani, R. and Buja, A. (1998). Flexible discriminant and mixture models, *in* J. Kay and M. Titterington (eds), *Statistics and Artificial Neural Networks*, Oxford University Press.

Hathaway, R. J. (1986). Another interpretation of the EM algorithm for mixture distributions, *Statistics & Probability Letters* **4**: 53–56.

Hebb, D. (1949). *The Organization of Behavior*, Wiley, New York.

Hertz, J., Krogh, A. and Palmer, R. (1991). *Introduction to the Theory of Neural Computation*, Addison Wesley, Redwood City, CA.

Hinton, G. (1989). Connectionist learning procedures, *Artificial Intelligence* **40**: 185–234.

Hoerl, A. E. and Kennard, R. (1970). Ridge regression: Biased estimation for nonorthogonal problems, *Technometrics* **12**: 55–67.

Huber, P. (1964). Robust estimation of a location parameter, *Annals of Math. Stat.* **53**: 73–101.

Huber, P. (1985). Projection pursuit, *Annals of Statistics* **13**: 435–475.

Hyvärinen, A. and Oja, E. (2000). Independent component analysis: Algorithms and applications, *Neural Networks* **13**: 411–430.

Izenman, A. (1975). Reduced-rank regression for the multivariate linear model, *Journal of Multivariate Analysis* **5**: 248–264.

Jacobs, R., Jordan, M., Nowlan, S. and Hinton, G. (1991). Adaptive mixtures of local experts, *Neural computation* **3**: 79–87.

Jain, A. and Dubes, R. (1988). *Algorithms for Clustering Data*, Prentice-Hall.

Jancey, R. (1966). Multidimensional group analysis, *Austral. J. Botany* **14**: 127–130.

Jones, L. (1992). A simple lemma on greedy approximation in Hilbert space and convergence rates for projection pursuit regression and neural network training, *Ann. Stat.* **20**: 608–613.

Jordan, M. and Jacobs, R. (1994). Hierachical mixtures of experts and the

EM algorithm, *Neural Computation* **6**: 181–214.

Kaufman, L. and Rousseeuw, P. (1990). *Finding Groups in Data: An Introduction to Cluster Analysis*, Wiley, New York.

Kearns, M. and Vazirani, U. (1994). *An Introduction to Computational Learning Theory*, MIT Press.

Kelly, C. and Rice, J. (1990). Monotone smoothing with application to dose-response curves and the assessment of synergism, *Biometrics* **46**: 1071–1085.

Kohonen, T. (1989). *Self-Organization and Associative Memory (3rd edition)*, Springer-Verlag, Berlin.

Kohonen, T. (1990). The self-organizing map, *Proc. of IEEE* **78**: 1464–1479.

Kohonen, T., Kaski, S., Lagus, K., Salojärvi, J., Paatero, A. and Saarela, A. (2000). Self organization of a massive document collection, *IEEE Transactions on Neural Networks* **11**(3): 574–585. Special Issue on Neural Networks for Data Mining and Knowledge Discovery.

Kressel, U. (1999). Pairwise classification and support vector machines, *in* B. Sch olkopf, C. Burges and A. Smola (eds), *Advances in Kernel Methods - Support Vector Learning*, MIT Press, Cambridge, MA., pp. 255–268.

Lawson, C. and Hansen, R. (1974). *Solving Least Squares Problems*, Prentice-Hall, Englewood Cliffs, NJ.

Le Cun, Y. (1989). Generalization and network design strategies, *Technical Report CRG-TR-89-4*, Dept. of Comp. Sci,. Univ. of Toronto.

Le Cun, Y., Boser, B., Denker, J., Henderson, D., Howard, R., Hubbard, W. and Jackel, L. (1990). Handwritten digit recognition with a back-propogation network, *in* D. Touretzky (ed.), *Advances in Neural Information Processing Systems*, Vol. 2, Morgan Kaufman, Denver, CO.

Le Cun, Y., Bottou, L., Bengio, Y. and Haffner, P. (1998). Gradient-based learning applied to document recognition, *Proceedings of the IEEE* **86**(11): 2278–2324.

Leblanc, M. and Tibshirani, R. (1996). Combining estimates in regression and classification, *J. Amer. Statist. Assoc.* **91**: 1641–1650.

Lin, H., McCulloch, C., Turnbull, B., Slate, E. and Clark, L. (2000). A latent class mixed model for analyzing biomarker trajectories in longitudinal data with irregularly scheduled observations., *Statistics in Medicine* **19**: 1303–1318.

Little, R. and Rubin, D. (1987). *Statistical Analysis with Missing Data*,

Wiley, New York.

Lloyd, S. (1957). Least squares quantization in PCM., *Technical report*, Bell Laboratories. Published in 1982 in IEEE Trans. Inf. Theory **28** 128-137.

Loader, C. (1999). *Local Regression and Likelihood*, Springer–Verlag.

Macnaughton Smith, P., Williams, W., Dale, M. and Mockett, L. (1965). Dissimilarity analysis: a new technique of hierarchical subdivision, *Nature* **202**: 1034–1035.

MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations, *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, eds. L.M. LeCam and J. Neyman*, Univ. of California Press, pp. 281–297.

Madigan, D. and Raftery, A. (1994). Model selection and accounting for model uncertainty using occam's window., *J. Amer. Statist. Assoc.* **89**: 1535–46.

Mardia, K., Kent, J. and Bibby, J. (1979). *Multivariate Analysis*, Academic Press.

Massart, D., Plastria, F. and Kaufman, L. (1983). Non-hierarchical clustering with MASLOC, *The Journal of the Pattern Recognition Society* **16**: 507–516.

McCulloch, W. and Pitts, W. (1943). A logical calculus of the ideas immenent in nervous activity, *Bull. Math. Biophys.* **5**: 115–133. pp 96-104; Reprinted in Andersen and Rosenfeld (1988).

McLachlan, G. J. (1992). *Discriminant Analysis and Statistical Pattern Recognition*, Wiley, New York.

Michie, D., Spiegelhalter, D. and Taylor, C. (eds) (1994). *Machine Learning, Neural and Statistical Classification*, Ellis Horwood Series in Artificial Intelligence, Ellis Horwood.

Morgan, J. N. and Sonquist, J. A. (1963). Problems in the analysis of survey data, and a proposal, *Journal of the American Statistical Association* pp. 415–434.

Murray, W., Gill, P., and Wright, M. (1981). *Practical Optimization*, Academic Press.

Myles, J. and Hand, D. (1990). The multiclass metric problem in nearest neighbor clasification, *Pattern Recognition* **23**: 1291–1297.

Neal, R. (1996). *Bayesian Learning for Neural Networks*, Springer-Verlag, New York.

Neal, R. and Hinton, G. (1998). *A view of the EM algorithm that justifies incremental, sparse, and other variants; in Learning in Graphical*

Models, M. Jordan (ed.),, Dordrecht: Kluwer Academic Publishers, Boston, MA., pp. 355–368.

Pace, R. K. and Barry, R. (1997). Sparse spatial autoregressions, *Statistics & Probability Letters* **33**: 291–297.

Parker, D. (1985). Learning logic, *Technical Report TR-87*, Cambridge MA: MIT Center for Reserach in Computational Economics and Management Science.

Platt, J. (1999). Fast training of support vector machines using sequential minimal optimization, *in* B. Schölkopf, C. J. C. Burges and A. J. Smola (eds), *Advances in Kernel Methods — Support Vector Learning*, MIT Press, Cambridge, MA., pp. 185–208.

Quinlan, R. (1993). *C4.5: Programs for Machine Learning*, Morgan Kaufmann, San Mateo.

Ramsay, J. and Silverman, B. (1997). *Functional Data Analysis*, Springer Verlag.

Rao, C. R. (1973). *Linear Statistical Inference and Its Applications*, Wiley, New York..

Ripley, B. D. (1996). *Pattern Recognition and Neural Networks*, Cambridge University Press.

Rissanen, J. (1983). A universal prior for integers and estimation by minimum description length, *Annals of Statistics* **11**: 416–431.

Robbins, H. and Munro, S. (1951). A stochastic approximation method, *Ann. Math. Stat.* **22**: 400–407.

Roosen, C. and Hastie, T. (1994). Automatic smoothing spline projection pursuit, *Journal of Computational and Graphical Statistics* **3**: 235–248.

Rosenblatt, F. (1958). The perceptron: a probabilistic model for information storage and organization in the brain, *Psychological Review* **65**: 386–408.

Rosenblatt, F. (1962). *Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms*, Spartan, Washington, D.C.

Rousseauw, J., du Plessis, J., Benade, A., Jordaan, P., Kotze, J., Jooste, P. and Ferreira, J. (1983). Coronary risk factor screening in three rural communities, *South African Medical Journal* **64**: 430–436.

Rumelhart, D., Hinton, G. and Williams, R. (1986). *Learning internal representations by error propagation in Parallel Distributed Processing: Explorations in the Microstructure of Cognition (Rumelhart, D.E. and McClelland, J. L eds.)*, The MIT Press, Cambridge, MA., pp. 318–362.

Schapire, R. (1990). The strength of weak learnability, *Machine Learning* **5**(2): 197–227.

Schapire, R., Freund, Y., Bartlett, P. and Lee, W. (1998). Boosting the

margin: a new explanation for the effectiveness of voting methods, *Annals of Statistics* **26**(5): 1651–1686.

Schapire, R. and Singer, Y. (1998). Improved boosting algorithms using confidence-rated predictions, *Proceedings of the Eleventh Annual Conference on Computational Learning Theory.*

Schwartz, G. (1979). Estimating the dimension of a model, *Annals of Statistics* **6**: 461–464.

Scott, D. (1992). *Multivariate Density Estimation: Theory, Practice, and Visualization*, Wiley, New York.

Seber, G. (1984). *Multivariate Observations*, Wiley, New York.

Shao, J. (1996). Bootstrap model selection, *J. Amer. Statist. Assoc.* **91**: 655–665.

Short, R. and Fukunaga, K. (1981). The optimal distance measure for nearest neighbor classification, *IEEE Transactions on Information Theory* **27**: 622–627.

Silverman, B. (1986). *Density Estimation for Statistics and Data Analysis*, Chapman and Hall, London.

Silvey, S. (1975). *Statistical Inference*, Halsted.

Simard, P., Le Cun, Y. and Denker, J. (1993). Efficient pattern recognition using a new transformation distance, *Advances in Neural Information Processing Systems*, Morgan Kaufman, San Mateo, CA, pp. 50–58.

Spiegelhalter, D., Best, N., Gilks, W. and Inskip, H. (1996). Hepatitis B: a case study in MCMC methods, *in* W. Gilks, S. Richardson and D. Spegelhalter (eds), *Markov Chain Monte Carlo in Practice*, Interdisciplinary Statistics, Chapman and Hall, London.

Stamey, T., Kabalin, J., McNeal, J., Johnstone, I., Freiha, F., Redwine, E. and Yang, N. (1989). Prostate specific antigen in the diagnosis and treatment of adenocarcinoma of the prostate II. radical prostatectomy treated patients, *Journal of Urology* **16**: 1076–1083.

Stone, C., Hansen, M., Kooperberg, C. and Truong, Y. (1997). Polynomial splines and their tensor products (with discussion), *Annals of Statistics* **25**(4): 1371–1470.

Stone, M. (1974). Cross-validatory choice and assessment of statistical predictions, *J. Roy. Statist. Soc.* **36**: 111–147.

Stone, M. (1977). An asymptotic equivalence of choice of model by cross-validation and Akaike's criterion, *J. Roy. Statist. Soc.* **39**: 44–7.

Stone, M. and Brooks, R. J. (1990). Continuum regression: Cross-validated sequentially constructed prediction embracing ordinary least squares, partial least squares and principal components regression (Corr: V54 p906-907), *Journal of the Royal Statistical Society, Series B, Methodological* **52**: 237–269.

Swayne, D., Cook, D. and Buja, A. (1991). Xgobi: Interactive dynamic graphics in the X window system with a link to S, *ASA Proceedings of Section on Statistical Graphics*, pp. 1–8.

Tanner, M. and Wong, W. (1987). The calculation of posterior distributions by data augmentation (with discussion), *J. Amer. Statist. Assoc.* **82**: 528–550.

Tarpey, T. and Flury, B. (1996). Self-consistency: A fundamental concept in statistics, *Statistical Science* **11**: 229–243.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso, *J. Royal. Statist. Soc. B.* **58**: 267–288.

Tibshirani, R. and Knight, K. (1999). Model search and inference by bootstrap "bumping, *J. Comp. and Graph. Stat.* **8**: 671–686.

Tibshirani, R., Walther, G. and Hastie, T. (2001). Estimating the number of clusters in a dataset via the gap statistic, *J. Royal. Statist. Soc. B.* **32**(2): 411–423.

Valiant, L. G. (1984). A theory of the learnable, *Communications of the ACM* **27**: 1134–1142.

van der Merwe, A. and Zidek, J. (1980). Multivariate regression analysis and canonical variates, *The Canadian Journal of Statistics* **8**: 27–39.

Vapnik, V. (1996). *The Nature of Statistical Learning Theory*, Springer-Verlag, New York.

Vidakovic, B. (1999). *Statistical Modeling by Wavelets*, Wiley, New York.

Wahba, G. (1980). Spline bases, regularization, and generalized cross-validation for solving approximation problems with large quantities of noisy data, *Proceedings of the International Conference on Approximation theory in Honour of George Lorenz*, Academic Press, Austin, Texas.

Wahba, G. (1990). *Spline Models for Observational Data*, SIAM, Philadelphia.

Wahba, G., Lin, Y. and Zhang, H. (2000). GACV for support vector machines, *in* A. Smola, P. Bartlett, B. Schölkopf and D. Schuurmans (eds), *Advances in Large Margin Classifiers*, MIT Press, Cambridge, MA., pp. 297–311.

Weisberg, S. (1980). *Applied Linear Regression*, Wiley, New York.

Werbos, P. (1974). *Beyond regression*, PhD thesis, Harvard University.

Wickerhauser, M. (1994). *Adapted Wavelet Analysis from Theory to Software*, A.K. Peters Ltd, Natick, MA.

Widrow, B. and Hoff, M. (1960). Adaptive switching curcuits, Vol. 4, IRE *WESCON* Convention record. pp 96-104; Reprinted in Andersen and Rosenfeld (1988).

Wold, H. (1975). Soft modelling by latent variables: The nonlinear iterative partial least squares (NIPALS) approach, *Perspectives in Probability and Statistics, In Honor of M. S. Bartlett*, pp. 117– 144.

Wolpert, D. (1992). Stacked generalization, *Neural Networks* **5**: 241–259.

Yee, T. and Wild, C. (1996). Vector generalized additive models, *Journal of the Royal Statistical Society, Series B.* **58**: 481–493.

Zhang, P. (1993). Model selection via multifold cross-validation, *Ann. Statist.* **21**: 299–311.

[General Information]
书名=统计学习基础　　数据挖掘、推理与预测_11210738
SS号=12431023