HZ BOOKS
华章教育

计 算 机 科 学 丛 书

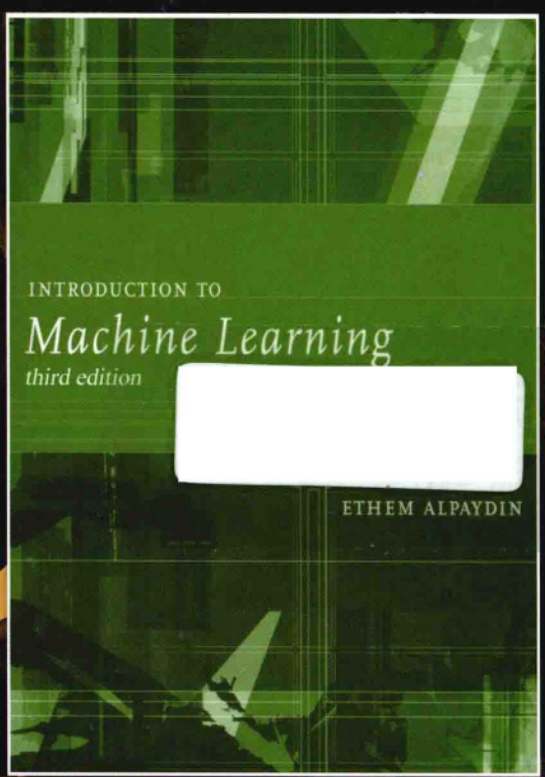
原书第3版

机器学习导论

[土耳其] 埃塞姆·阿培丁（Ethem Alpaydin） 著
范明 译

Introduction to Machine Learning

Third Edition



机械工业出版社
China Machine Press

机器学习导论 原书第3版

Introduction to Machine Learning Third Edition

本书把机器学习的热门话题（如Tom Mitchell）与概率论基础（如Christopher Bishop）很好地融合在一起。第3版向这个重要和迅速发展领域中的学生和研究者介绍了机器学习的一些最新和最重要的课题（例如，谱方法、深度学习和学习排名）。

—— John W. Sheppard 蒙大拿州立大学计算机科学教授

我已经在机器学习的研究生课程中使用本书多年。这本书很好地平衡了理论和实践，并且在第3版中扩充了许多新的先进算法。我期待在我的下一次机器学习课程中使用它。

—— Larry Holder 华盛顿州立大学电子工程和计算机科学教授

对于机器学习而言，这是一本完整、易读的机器学习导论，是这个快速演变学科的“瑞士军刀”。尽管本书旨在作为导论，但是它不仅对于学生，而且对于寻求这一领域综合教程的专家也是有用的。新人会从中找到清晰解释的概念，专家会从中发现新的参考和灵感。

—— Hilario Gómez-Moreno IEEE高级会员

机器学习的目标是对计算机编程，以便使用样本数据或以往的经验来解决给定的问题。已经有许多机器学习的成功应用，包括分析以往销售数据来预测客户行为，优化机器人的行为以便使用最少的资源来完成任务，以及从生物信息数据中提取知识的各种系统。本书是关于机器学习的内容全面的教科书，其中有些内容在一般的在机器学习导论书中很少介绍。主要内容包括监督学习，贝叶斯决策理论，参数、半参数和非参数方法，多元分析，隐马尔可夫模型，增强学习，核机器，图模型，贝叶斯估计和统计检验。

机器学习正在迅速成为计算机科学专业的学生必须掌握的一门技能。本书第3版反映了这种变化，增加了对初学者的支持，包括给出了部分习题的参考答案和补充了实例数据集（提供在线代码）。其他显著的变化包括离群点检测的讨论、感知器和支持向量机的排名算法、矩阵分解和谱方法、距离估计、新的核算法、多层感知器的深度学习和非参数贝叶斯方法。书中对所有学习算法都进行了解释，以便读者可以很容易地将书中的公式转变为计算机程序。本书可以用作高年级本科生和硕士研究生的教材，也可供研究机器学习方法的技术人员参考。

作者简介

埃塞姆·阿培丁（Ethem Alpaydin） 土耳其伊斯坦布尔博阿齐奇大学计算机工程系的教授。于1990年在洛桑联邦理工学院获博士学位，先后在美国麻省理工学院和伯克利大学工作和进行博士后研究。Ethem博士主要从事机器学习方面的研究，是剑桥大学《The Computer Journal》杂志编委和Elsevier《Pattern Recognition》杂志的副主编。2001年和2002年，Ethem博士先后获得土耳其科学院青年科学家奖和土耳其科学与技术研究委员会科学奖。

上架指导：计算机\人工智能\机器学习

ISBN 978-7-111-52194-5

9 787111 521945 >

定价：79.00元

投稿热线：(010) 88379604
客服热线：(010) 88378991 88361066
购书热线：(010) 68326294 88379649 68995259

华章网站：www.hzbook.com
网上购书：www.china-pub.com
数字阅读：www.hzmedia.com.cn

图书在版编目（CIP）数据

机器学习导论（原书第3版）/（土）阿培丁（Alpaydin, E.）著；范明译．—北京：机械工业出版社，2015.11
（计算机科学丛书）
书名原文：Introduction to Machine Learning, Third Edition

ISBN 978-7-111-52194-5

I. 机… II. ①阿… ②范… III. 机器学习－研究 IV. TP181

中国版本图书馆CIP数据核字（2015）第282166号

本书版权登记号：图字：01-2015-1269

本书全面介绍了机器学习的相关内容，涵盖了监督学习、贝叶斯决策理论、参数方法、多元方法、归约、聚类、非参数方法、决策树、线性判别式、多层感知器、局部模型、核机器、图方法、隐马尔可夫模型、贝叶斯估计、组合多学习器、增强学习以及机器学习实验的设计与分析等，反映了快速发展的机器学习领域的最新进展。

本书可以用作高年级本科生和硕士研究生的教材，也可供研究机器学习方法的技术人员参考。

出版发行：机械工业出版社（北京市西城区百万庄大街22号　邮政编码：100037）
责任编辑：盛思源　　责任校对：董纪丽
印　　刷：中国电影出版社印刷厂　　版　　次：2016年1月第1版第1次印刷
开　　本：185mm×260mm　1/16　　印　　张：23.25
书　　号：ISBN 978-7-111-52194-5　　定　　价：79.00元

凡购本书，如有缺页、倒页、脱页，由本社发行部调换
客服热线：（010）88378991　88361066　　投稿热线：（010）88379604
购书热线：（010）68326294　88379649　68995259　　读者信箱：hzjsj@hzbook.com

引　言

1.1　什么是机器学习

这是一个“大数据”时代。过去，只有公司才拥有数据。那时，有一些计算中心，数据在那里存储和处理。先是个人计算机的出现，而后是无线通信的广泛使用，使得我们都成了数据的生产者。每当我们购买一件商品、租借一部电影、访问一个网页、书写一个博客或在社交媒体上发帖子时，甚至当我们散步或开车闲逛时，我们都在产生数据。

我们每个人不仅是数据的生产者，而且也是数据的消费者。我们想要适合的产品和服务，希望我们的需要能被理解，我们的兴趣能被预测到。

以一家连锁超市为例，它通过遍布全国的数百家实体商店或通过网上的虚拟商店向数百万顾客销售数千种商品。每笔交易的细节，包括交易日期、顾客ID、购买的商品和数量、付款金额等都存储在计算机中。这意味每天都有大量的数据。连锁超市希望能够预测哪位顾客可能会购买哪种商品，以便能够使销售和利润最大化。类似地，每位顾客都希望找到最适合他们需要的商品。

这一任务并非显而易见。我们并不确切地知道哪些人比较倾向于购买这种口味的冰激凌，这位作家的下一本书是什么，也不知道谁喜欢看这部新电影、访问这座城市，或点击这一链接。顾客的行为随时间和地点而变化。但是，我们知道这不是完全随机的。人们去超市并不是随机购买商品。当他们买啤酒时，也会买薯片；夏天买冰激凌，而冬天为Glühwein[⊖]买香料。数据中存在确定的模式。

为了在计算机上解决问题，我们需要算法。算法是指令的序列，它把输入变换成输出。例如，我们可以为排序设计一个算法，输入是数的集合，而输出是它们的有序列表。对于相同的任务，可能存在不同的算法，而我们感兴趣的是找到需要的指令、内存最少，或者二者都最少的最有效算法。

然而，对于某些任务，我们没有算法。预测顾客的行为就是一个例子，另一个例子是区分垃圾邮件和正常邮件。我们知道输入是邮件文档，在最简单的情况下是一个字符文件。我们还知道输出应该是指出消息是否为垃圾邮件的“是”或“否”。但是我们不知道如何把这种输入变换成输出。所谓的垃圾邮件随时间而变，因人而异。

我们缺乏的是知识，作为补偿我们有数据。我们可以很容易地编辑数以千计的实例消息，其中一些我们知道是垃圾邮件，而我们要做的是希望从中“学习”垃圾邮件的结构。换言之，我们希望计算机(机器)自动地为这一任务提取算法。不需要学习如何将数排序，因为我们已经有这样的算法。但是，对于许多应用而言，我们确实没有算法，而是有实例数据。

我们也许不能够完全识别该过程，但是我们相信，我们能够构造一个好的并且有用的近似。尽管这样的近似还不可能解释一切，但其仍然可以解释数据的某些部分。我们相

⊖ Glühwein是一种温热、有点甜味儿、加香料的葡萄酒。圣诞节期间，在欧洲很受欢迎。——译者注

信，尽管识别整个过程也许是不可能的，但是我们仍然能够发现某些模式或规律。这正是机器学习的定位。这些模式可以帮助我们理解该过程，或者我们可以使用这些模式进行预测：假定将来(至少是不远的将来)情况不会与收集样本数据时有很大的不同，则未来的预测也将有望是正确的。

机器学习方法在大型数据库中的应用称为*数据挖掘*(data mining)。类似的情况如大量的金属氧化物以及原料从矿山中开采出来，处理后产生少量非常珍贵的物质。类似地，在数据挖掘中，需要处理大量的数据以构建有使用价值的简单模型，例如具有高准确率的预
2 测模型。数据挖掘的应用领域非常广泛：除零售业以外，在金融业，银行分析历史数据，构建用于信用分析、诈骗检测、股票市场等方面的应用模型；在制造业，学习模型可以用于优化、控制以及故障检测等；在医学领域，学习程序可以用于医疗诊断等；在电信领域，通话模式的分析可用于网络优化和提高服务质量；在科学研究领域，比如物理学、天文学以及生物学的大量数据只有使用计算机才可能得到足够快的分析。万维网是巨大的，并且在不断增长，因此在万维网上检索相关信息不可能依靠人工完成。

然而，机器学习不仅仅是数据库方面的问题，它也是人工智能的组成部分。为了智能化，处于变化环境中的系统必须具备学习能力。如果系统能够学习并且适应这些变化，那么系统的设计者就不必预见所有的情况并为它们提供解决方案了。

机器学习还可以帮助我们解决视觉、语音识别以及机器人方面的许多问题。以人脸识别问题为例。我们做这件事毫不费力。即使姿势、光线、发型等不同，我们每天还是可以通过观察真实的面孔或照片来认出家人和朋友。但是我们做这件事是无意识的，而且无法解释我们是如何做的。因为我们不能够解释我们所具备的这种技能，所以我们也就不可能编写相应的计算机程序。但是我们知道，脸部图像并非只是像素点的随机组合；人脸是有结构的、对称的。脸上有眼睛、鼻子和嘴巴，并且它们都位于脸的特定部位。每个人的脸都有各自的眼睛、鼻子和嘴巴的特定组合模式。通过分析一个人的脸部图像的多个样本，学习程序可以捕捉到那个人特有的模式，然后在所给的图像中检测这种模式，从而进行辨认。这就是*模式识别*(pattern recognition)的一个例子。

机器学习使用实例数据或过去的经验训练计算机来优化某种性能标准。我们有依赖于某些参数的模型，而学习就是执行计算机程序，利用训练数据或以往经验来优化该模型的参数。模型可以是*预测性的*(predictive)，用于未来的预测；或者是*描述性的*(descriptive)，用于从数据中获取知识；也可以二者兼备。

机器学习在构建数学模型时利用了统计学理论，因为其核心任务就是由样本推理。计
3 算机科学的角色是双重的：第一，在训练时，我们需要求解优化问题以及存储和处理通常所面对的海量数据的高效算法。第二，一旦学习得到了一个模型，它的表示和用于推理的算法解也必须是高效的。在特定的应用中，学习或推理算法的效率，即它的空间复杂度和时间复杂度，可能与其预测的准确率同样重要。

现在，让我们更详细地讨论一些应用领域的例子，以便进一步深入了解机器学习的类型和用途。

1.2 机器学习的应用实例

1.2.1 学习关联性

在零售业，例如超市连锁店，机器学习的一个应用是*购物篮分析*(basket analysis)。它的任务是发现顾客所购商品之间的关联性：如果购买商品 X 的人通常也购买商品 Y，而

一位顾客购买了商品 X 却未购买商品 Y，则他就是商品 Y 的潜在顾客。一旦我们发现这类顾客，我们就能针对他们实施交叉销售策略。

为了发现关联规则(association rule)，我们对学习形如 $P(Y|X)$ 的条件概率感兴趣，其中 X 是我们知道的顾客已经购买的商品或商品集，Y 表示在条件 X 下可能购买的商品。

假定考察已有的数据，计算得到 $P(\text{chips}|\text{beer})=0.7$，那么我们就可以定义规则：

购买啤酒(beer)的顾客中有70%的人也买了薯片(chip)

我们也许想要区分不同的顾客。针对这个问题，我们需要估计 $P(Y|X, D)$，其中 D 是顾客的一组属性，如性别、年龄、婚姻状况等，这里假定我们已经得到了这些属性信息。如果考虑书店而不是超市销售问题，商品就可能是书或作者等。对于 Web 门户网站入口问题，商品对应于到 Web 网页的链接，而我们可以估计用户可能点击的链接，并利用这些信息预先下载这些网页，以便取得更快的网页访问速度。

4

1.2.2 分类

信贷是金融机构(例如银行)借出的一笔钱，需要连本带息偿还，通常分期偿还。对银行来说，重要的是能够提前预测贷款风险。这种风险是客户不履行义务和不全额还款的可能性。既要确保银行获利，又要确保不会因提供超出客户财力的贷款而给客户带来不便。

在资信评分(credit scoring)(Hand 1998)中，银行计算在给定信贷额度和客户信息情况下的风险。客户信息包括我们已经获取的数据以及与计算客户财力相关的数据，即收入、存款、担保、职业、年龄、以往经济记录等。银行有以往贷款的记录，包括客户数据以及贷款是否偿还。通过这类特定的申请数据，可以推断出表示客户属性及其风险关联性的一般规则。也就是说，机器学习系统用一个模型来拟合过去的数据，以便能够对新的申请计算风险，从而决定接受或拒绝该项申请。

这是分类(classification)问题的一个例子，这里有两个类：低风险客户和高风险客户。客户信息作为分类器的输入(input)，分类器的任务是将输入指派到其中的一个类。

利用以往数据进行训练后，学习得到的规则可能具有如下形式：

IF income $>\theta_1$ AND savings $>\theta_2$

THEN low-risk ELSE high-risk

其中 θ_1 和 θ_2 是合适的值(参见图 1-1)。这是判别式(discriminant)的一个例子，判别式是将不同类的样本分开的函数。

有了这样的规则，主要用途就是预测(prediction)：一旦我们拥有拟合以往数据的规则，如果未来与过去类似，那么我们就能够对新的实例做出正确的预测。如果给定一个具有特定收入(income)和存款(savings)的新申请，则我们就可以容易地判断出它是低风险(low-risk)还是高风险(high-risk)。

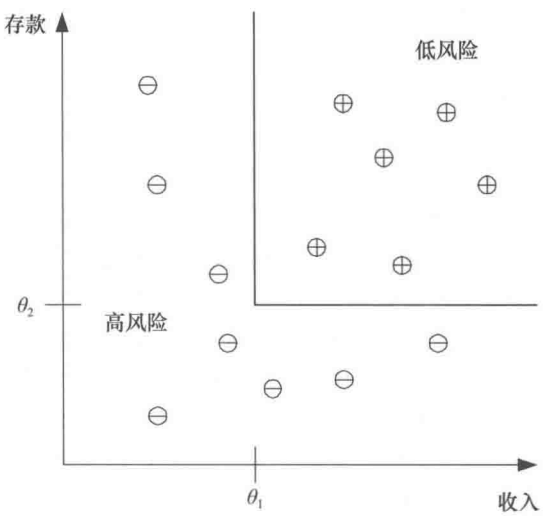


图 1-1　训练数据集例子，其中每个圆圈对应于一个数据实例，输入值在对应的坐标上，符号指示类别。为了简单起见，输入只包括客户的收入和存款两种属性，两个类分别为低风险("+")和高风险("−")。图中还显示了分隔两类样本的判别式的例子

在某些情况下，我们可能不是希望做0/1(低风险/高风险)类型的判断，而是希望计算一个概率值$P(Y|X)$，其中X是顾客属性，Y是0或1，分别表示低风险和高风险。从这个角度来看，我们可以将分类看作学习从X到Y的关联性。于是，给定$X=x$，如果有$P(Y=1|X=x)=0.8$，则我们就说该客户为高风险的可能性有80%，或者等价地说，该客户为低风险的可能性有20%。然后，我们可以根据可能的收益和损失来决定接受还是拒绝这笔贷款业务。

机器学习在模式识别(pattern recognition)方面有很多应用。其中之一是光学字符识别(Optical Character Recognition，OCR)，即从字符图像识别字符编码。这是多类问题的一个例子，类与我们想要识别的字符一样多。特别有趣的是手写体字符的识别问题。人们有不同的书写风格，字体有大有小，倾斜角度不同，还有用钢笔或用铅笔之别，所以同一个字符可能会有许多种可能的图像。尽管书写是人类的发明创造，但是还没有像人类读者一样准确的系统。我们没有字符“A”的形式化描述，涵盖所有“A”而不涵盖任何非“A”。没有这种形式化描述，我们就要从书写者那里取样，从这些实例中学习关于“A”的定义。然而，尽管我们不知道是什么因素使得一个图像被识别为“A”，但是我们确信所有这些不同的“A”的图像都具有某些共同的特征，这正是我们希望从实例中提取的。我们知道，字符图像不只是随机点的集合。它是笔画的集合，并且是有规律的，通过学习程序我们能够捕获这些规律。

5～6

阅读文本时，我们能够利用的一个因素是人类语言的冗余性。词是字符的序列，并且相继的符号不是独立的，而是被语言的词所约束。这有好处，即便有一个符号不能识别，我们仍可以读出词t?e[⊖]。根据语言的语法和语义，这种上下文的依赖性还可能出现在词和句子之间等较高的层次上。目前存在用于学习序列和对这种依赖性建模的机器学习算法。

对于人脸识别(face recognition)，输入是人脸图像，而类是需要识别的人，并且学习程序应当学习人脸图像与身份之间的关联性。这个问题比光学字符识别更困难，因为人脸会有更多的类，输入图像也更大一些，并且人脸是三维的，不同的姿势和光线等都会导致图像的显著变化。另外，对于特定人脸的输入也会出现问题，比如说眼镜可能会把眼睛和眉毛遮住，胡子可能会把下巴盖住等。

在医学诊断(medical diagnosis)中，输入是关于患者的信息，而类是疾病。输入包括患者的年龄、性别、既往病史、目前症状等。当然，患者可能还没有做过某些检查，因此这些输入将会缺失。检查需要时间，还可能要花很多钱，而且也许还会给患者带来不便。因此，除非我们确信检查将提供有价值的信息，否则我们不对患者进行检查。在医学诊断的情况下，错误的诊断结果可能会导致错误的治疗或根本不治疗。在不能确信诊断结果的情况下，分类器最好还是放弃判定，而等待医学专家来决断。

在语音识别(speech recognition)中，输入是语音，类是可以读出的词汇。这里要学习的是从语音信号到某种语言的词汇的关联性。由于年龄、性别或口音方面的差异，不同的人对于相同词汇的读音不同，这使得语音识别相当困难。语音识别的另一个特点是其输入信号是时态的(temporal)，词汇作为音素的序列实时读出，而且有些词汇的读音会比其他词汇长一些。

7

语音信息的作用有限，并且与光学字符识别一样，在语音识别中，“语言模型”的集成是至关重要的，而且提供语言模型的最好方法仍然是从实例数据的大型语料库中学习。机

⊖ 这里，“?”表示不能识别的符号。——译者注

器学习在自然语言处理(natural language processing)方面的应用与日俱增。垃圾邮件过滤就是一种应用，那里垃圾邮件的制造者为一方，过滤者为另一方，它一直都在寻找越来越精巧的方法来超越对方。大型文档汇总是另一个有趣的例子；还有一个例子是分析博客或社交网站上的帖子，以便提取"流行"主题或决定做什么广告。也许最吸引人的是机器翻译(machine translation)。经历了数十年手工编写翻译规则的研究之后，最近人们认识到最有希望的方法是提供大量两种语言文本的实例对，让程序自动地揣摩把一种语言映射到另一种语言的规则。

生物测定学(biometrics)使用人的生理和行为特征来识别或认证人的身份，它需要集成来自不同形态的输入。生理特征的例子有面部图像、指纹、虹膜和手掌；行为特征的例子有签字的力度、嗓音、步态和击键。与通常的鉴别过程(照片、印刷签名或口令)相反，会有许多不同的(不相关的)输入，伪造(欺骗)更困难，并且系统更准确，有望不会对用户太不方便。机器学习既用于对这些不同形态构建不同的识别器，也考虑这些不同数据源的可靠性，用于组合它们的决策，以便得到接受或拒绝的总体决断。

从数据中学习规则也为知识抽取(knowledge extraction)提供了可能性。规则是一种解释数据的简单模型，而观察该模型我们就能得到潜在数据处理的解释。例如，一旦我们学习得到区分低风险客户和高风险客户的判别式，我们就拥有了关于低风险客户特性的知识。然后，我们就能够利用这些信息，通过广告等方式，更有效地争取那些潜在的低风险客户。机器学习还可以进行压缩(compression)。用规则拟合数据，我们得到比数据更简 8
单的解释，需要的存储空间更少，处理所需要的计算更少。例如，一旦掌握了加法规则，就不必记忆每对可能数的和是多少。

机器学习的另一种用途是离群点检测(outlier detection)，即发现那些不遵守规则和例外的实例。基本思想是，典型的实例具有一些可以简单陈述的特征，而不具备这些特征的实例都是非典型的。在这种情况下，我们感兴趣的是找到一个尽可能简单并且覆盖尽可能多的典型实例的规则。落在外面的实例都是例外，它们可能是提示我们需要注意的异常(如诈骗)，也可能是新颖的、先前未曾见过但又合理的情况。因此，离群点检测又称为新颖性检测(novelty detection)。

1.2.3　回归

假设我们想要一个能够预测二手车价格的系统。该系统的输入是我们认为会影响车价的属性信息：品牌、车龄、发动机排量、里程以及其他信息。输出是车的价格。这种输出为数值的问题是回归(regression)问题。

设 X 表示车的属性，Y 表示车的价格。调查以往的交易情况，我们能够收集训练数据，而机器学习程序用一个函数拟合这些数据来学习 X 的函数 Y。图 1-2 给出了一个例子，其中对于 w 和 w_0 的合适值，拟合函数具有以下形式：

$$y=wx+w_0$$

回归和分类均为监督学习(supervised learning)问题，其中给定输入 X 和输出 Y，任务是学习从输入到输出的映射。机器学习的方法是，先假定依赖于一组参数的模型：

$$y=g(x|\theta)$$

其中，$g(\cdot)$是模型，而 θ 是模型的参数。对于回归，Y 是数值；对于分类，Y 是类编码(如 0/1)。$g(\cdot)$为回归函数，或者(对于分类)是将不同类的实例分开的判别式函数。机器学习程序优化参数 θ，使近似误差最小，也就是说，我们的估计要尽可能地接近训练集

中给定的正确值。例如，图 1-2 所示的模型是线性的，w 和 w_0 是为最佳拟合训练数据优化后的参数。在线性模型限制过强的情况下，我们可以利用二次函数

$$y=w_2x^2+w_1x+w_0$$

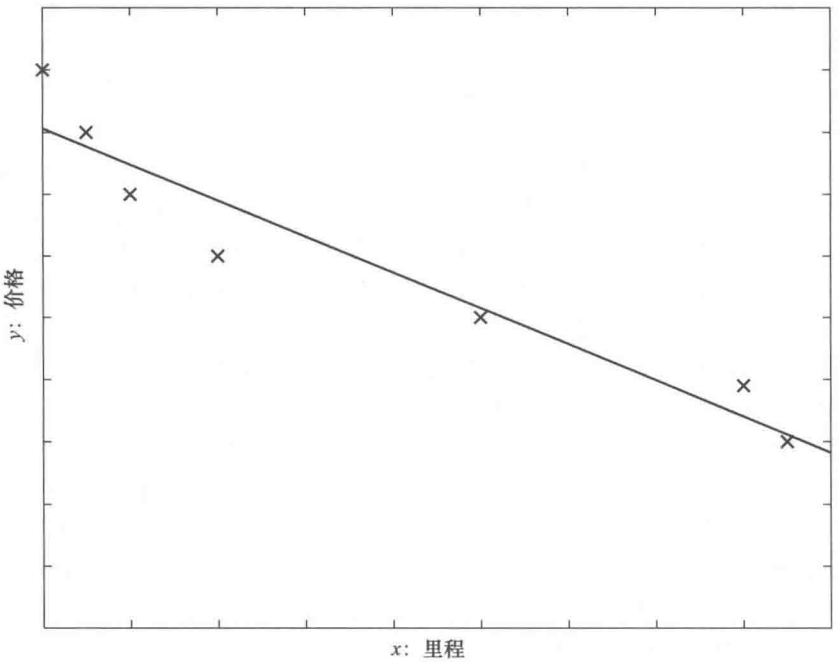


图 1-2 二手车的训练数据及其拟合函数。为简单起见，这里采用线性模型，输入属性也只有里程

或更高阶的多项式，或其他非线性函数，为最佳拟合优化它们的参数。

回归的另一个例子是移动机器人导航，例如，自动汽车导航，其中输出是每次转动车轮的角度，使汽车前进而不会撞到障碍物或偏离车道。在这种情况下，输入由汽车上的传感器(如视频相机、GPS 等)提供。训练数据可以通过监视和记录驾驶员的动作来收集。

我们可以想象回归的其他应用，这里我们试图优化一个函数[⊖]。假设我们想要制造一个焙炒咖啡的机器。该机器有多个影响咖啡品质的输入：温度、时间、咖啡豆种类等。我们针对不同的输入配置进行大量试验，并估量咖啡的品质。例如，根据消费者的满意度测量咖啡的品质。为找到最优配置，我们拟合一个联系这些输入和咖啡品质的回归模型，并在当前模型的最优样本附近选择一些新的点，以便寻找更好的配置。我们抽取这些点，检测咖啡的品质，将它们加入训练数据，并拟合新的模型。这通常称为*响应面设计*(response surface design)。

有时，我们希望能够学习一个相对位置，而不是估计一个绝对数值。例如，在电影*推荐系统*(recommendation system)中，我们希望产生一张表，按照用户的喜欢程度将电影排序。根据电影的体裁、演员等属性，并使用用户对他们所看过电影的评级，我们希望能够学习一个*排名*(ranking)函数，然后可以使用它选择新电影。

9 ~ 10

1.2.4 非监督学习

在监督学习中，我们的目标是学习从输入到输出的映射关系，其中输出的正确值已经

⊖ 感谢 Michael Jordon 提供这个例子。

由指导者提供。然而，在非监督学习中却没有这样的指导者，只有输入数据。我们的目标是发现输入数据中的规律。输入空间存在着某种结构，使得特定的模式比其他模式更常出现，而我们希望知道哪些经常发生，哪些不经常发生。在统计学中，这称为密度估计(density estimation)。

密度估计的一种方法是聚类(clustering)，其目标是发现输入数据的簇或分组。对于拥有老客户数据的公司，客户数据包括客户的个人统计信息及其以前与公司的交易，而公司也许想知道其客户的分布，搞清楚什么类型的客户会频繁出现。这种情况下，聚类模型会将属性相似的客户分派到相同的分组，为公司提供其客户的自然分组，这称作客户划分 11
(customer segmentation)。一旦找出了这样的分组，公司也许会做出一些决策，比如对不同分组的客户提供特别的服务和产品等，这称作客户关系管理(customer relationship management)。这样的分组也可以用于识别“离群点”，即那些不同于其他客户的客户，这可能意味新的市场商机，公司可以进一步开发。

聚类的一个有趣的应用是图像压缩(image compression)。在这种情况下，输入实例是由 RGB 值表示的图像像素。聚类程序将颜色近似的像素分到相同的分组，而这样的分组对应于图像中频繁出现的颜色。如果图像中只有少数几种颜色，并且属于同一分组的像素用一种颜色(例如，颜色的平均值)进行编码，则图像被量化。假设像素是 24 位，表示 1600 万种颜色，但是如果只有 64 种主色调，那么对于每个像素，只需要 6 位而不是 24 位。例如，如果景象在图像的不同部分有多种不同的蓝色色调，并且采用它们的平均值来表示所有这些蓝色，那么就丢失了图像的细节，但是赢得了图像的存储空间和传送时间。在理想情况下，人们希望通过分析重复的图像模式(如纹理、对象等)来识别更高层次的规律性。这为更高层次、更简单、更有用地描述景象提供了可能，并且实现了比像素级更好的压缩。如果我们扫描了文档页，则我们得到的不是随机的有/无像素，而是一些字符的位图。这样的数据是有结构的，并且我们利用这些冗余信息，找出数据的较短描述：“A”的 16×16 的位图占 32 字节，其 ASCII 码只占 1 字节。

在文档聚类(document clustering)中，目标是把相似的文档分组。例如，新闻报道可以进一步划分为政治、体育、时尚、艺术等子组。通常，文档用词袋(bag of words)表示，即预先定义 N 个词的词典，并且每个文档都是一个 N 维二元向量，如果第 i 个词出现在该文档中，则其第 i 个分量取 1。删除后缀“-s”和“-ing”等，以避免重复，并且不用“of”、“and”等不包含什么信息的词。然后，文档根据它们包含的相同词的个数分组。当然，如何选取词典是至关重要的。

机器学习方法还应用于生物信息学(bioinformatics)。在我们的基因组中，DNA 是“生命的蓝图”，也是碱基(即 A、G、C 和 T)序列。RNA 由 DNA 转录而来，而蛋白质由 12
RNA 转换而来。蛋白质就是生命体和生命体的产物。正如 DNA 是碱基序列，蛋白质则是氨基酸(由碱基定义)序列。计算机科学在分子生物学的应用领域之一就是比对(alignment)，即将一个序列与另一个匹配。这是一个困难的串匹配问题，因为序列可能相当长，有很多模板串要进行匹配，并且还可能会删除、插入和置换。聚类用于学习基序(motifs)，这是蛋白质结构中反复出现的氨基酸序列。基序之所以令人感兴趣，是因为它们可能对应于它们所表征的序列内部的结构或功能要素。比方说，如果氨基酸是字母，蛋白质是句子，那么基序就像单词，即具有特别意义、频繁出现在不同句子中的一串字母。

1.2.5　增强学习

在某些应用中，系统的输出是动作(action)的序列。在这种情况下，单个的动作并不

重要，重要的是策略(policy)，即达到目标的正确动作的序列。不存在中间状态中最好动作这种概念。如果一个动作是好的策略的组成部分，那么该动作就是好的。在这种情况下，机器学习程序就应当能够评估策略的好坏程度，并从以往好的动作序列中学习，以便能够产生策略。这种学习方法称为增强学习(reinforcement learning)算法。

游戏(game playing)是一个很好的例子。在游戏中，单个移动本身并不重要，正确的移动序列才是重要的。如果一个移动是一个好的游戏策略的一部分，则它就是好的。游戏是人工智能和机器学习的一个重要研究领域。这是因为游戏容易描述，但又很难玩好。像国际象棋这样的游戏，其规则只有少量的几条，但是它非常复杂，因为在每种状态下都有大量可行的移动，并且每局又都包含大量的移动。一旦有了能够学习如何玩好游戏的好算法，我们也可以将这些算法用在具有更显著经济效益的领域。

在某种环境下搜寻目标位置的机器人导航是增强学习的另一个应用领域。在任何时
13 候，机器人都能够朝着多个方向之一移动。经过多次试运行，机器人应当学到正确的动作序列，尽可能快地从某一初始状态到达目标状态，并且不会撞到任何障碍物。

使增强学习更困难的一个因素是系统具有不可靠和不完整的感知信息。例如，装备视频照相机的机器人就得不到完整的信息，因此该机器人总是处于部分可观测状态(partially observable state)，并且在决定其动作时应当将这种不确定性考虑在内。例如，机器人可能不知道它在房间的准确位置，而只知道其左边有一道墙。一个任务还可能需要多智能主体(multiple agents)的并行操作，这些智能主体将相互作用并协同操作，以便完成一个共同的目标。机器人足球是这种情况的例子之一。

1.3 注释

进化是形成我们的身体形状和我们内在本能的主要力量。我们还需要终生学习，以改变我们的行为。这有助于我们适应进化论还不能预测的环境变化。在合适的环境中，具有短暂寿命的生物体可能具备它们所有天生的行为能力，而上苍并未赋予我们应对在有限生命中可能遇见的所有状况的能力。但是，进化赋予我们大脑和学习机制，使得我们可以根据经验实现自我更新，从而适应各种环境。当我们在特定情境下学习最好的策略时，知识就存储在我们的大脑里。当情境再现时，当我们再认知("认知"意味认出)情境时，我们就能够回忆起合适的策略并采取相应的动作。

不过，学习有其局限性。就我们大脑的有限容量来说，也许有些东西我们永远都不可能学会，正像我们永远不可能"学会"长出第三只手臂或在脑袋后面长眼睛，即使它们是有用的我们也学不会。注意，与心理学、认知科学以及神经系统科学不同，机器学习的目标并不是理解人类和动物学习的过程，而是像任何工程领域一样，机器学习旨在构建有用的系统。

几乎所有的科学领域都在用模型拟合数据。科学家设计实验、进行观测并收集数据。然后，通过寻找解释所观测数据的简单模型，尝试抽取知识。该过程称为归纳(induc-
14 tion)，它是从一组特别的示例中提取通用规则的过程。

现在，这样的数据分析已经不能依赖人工完成了，原因有二：一是数据量巨大；二是能够做这种分析的人非常短缺且人工分析又很昂贵。因此，对于能够分析数据且自动从中提取信息的计算机模型，也就是说对于学习，人们的兴趣正在不断地增长。

在下面的章节中，我们要讨论的方法源于不同的科学领域。有时，相同的算法会在多个领域中沿着各自不同的历史轨迹被独立地发现。

在统计学中，从特殊观测到一般描述称为推断(inference)，而学习称为估计(estimation)。分类在统计学中称为判别式分析(discriminant analysis)(McLachlan 1992；Hastie，Tibshirani 和 Friedman 2001)。在计算机价格低廉且数量充足以前，统计学家只能处理小样本。作为数学家，统计学家主要使用能够精确分析的简单参数模型。在工程学中，分类称为模式识别(pattern recognition)，方法是非参数的，并且更大程度是凭借经验的(Duda，Hart 和 Stork 2001；Webb 1999)。

机器学习还与人工智能(artificial intelligence)有关(Russell 和 Norvig 1995)，因为智能系统应当能够适应其环境的变化。像视觉、语音和机器人这样的应用领域都是从样本数据中学习。在电子工程领域，信号处理(signal processing)的研究导致自适应计算机视觉和语音程序出现。其中，隐马尔科夫模型(Hidden Markov Model，HMM)的发展对于语音识别尤其重要。

20 世纪 80 年代后期，随着 VLSI 技术的发展和制造包含数千个处理器并行硬件的可能性出现，基于多处理单元的分布式计算理论的可行性使得人工神经网络(artificial neural network)研究领域获得重生(Bishop，1995)。随着时间的推进，人们认识到在神经网络研究领域中，大多数神经网络学习算法都具有统计学的基础(例如，多层感知器就是另一类的非参估计)，因此模拟人脑计算的说法开始逐渐淡出。

近年来，基于核的算法(如支持向量机)日趋流行。借助于使用核函数，支持向量机适用于各种应用，尤其适合生物信息学和自然语言处理方面的应用。如今，人们已经广泛认识到，对于学习而言，好的数据表示至关重要，而核函数是一种引进这种专家知识的好 15
方法。

另一种新方法是使用生成模型(generative model)，它通过一组隐藏因子的相互影响来解释观测数据。一般而言，图模型(graphical model)用来对这些因子和数据的相互影响进行可视化，而贝叶斯形式化机制(Bayesian formalism)使我们既可以定义隐藏因子和模型上的先验信息，又能推导模型的参数。

最近，随着存储和连接费用的降低，在因特网上使用非常大的数据库已经成为可能，再加上廉价的计算，已经使得在大量数据上运行学习算法成为可能。在过去的几十年中，人们一般相信，对于人工智能而言，我们需要新的范型、新的思维、新的计算模型或一些全新的算法。

考虑到机器学习最近在各领域的成功，也许可以说，我们需要的不是新算法，而是大量数据实例和在这些数据上运行算法的充足计算能力。例如，支持向量机源于势函数(potential function)、线性分类和基于最近邻的方法，这些都是 20 世纪 50 或 60 年代提出的，那时，我们只是没有适合这些算法的快速计算机或大型存储器，不能完全展示它们的潜力。可以推测，机器翻译甚至规划这样的任务都可以用这种相对简单的算法来解决，但需要在大量实例数据上训练或通过长时间试错运行。“深度学习”最近取得的成功支持了这种说法。智能看来不像源于某些稀奇古怪的公式，而是源于简单、直截了当的算法的耐心和近乎蛮力的使用。

数据挖掘(data mining)的命名来源于机器学习算法在商界海量数据上的应用(Witten 和 Frank 2011；Han 和 Kamber 2011)。在计算机科学领域中，数据挖掘也称为数据库中知识发现(Knowledge Discovery in Databases，KDD)。

在统计学、模式识别、神经网络、信号处理、控制、人工智能以及数据挖掘等不同领域中，研究工作遵循着各自的途径，并有其各自的侧重点。本书的目标是结合所有这些研

16 究重点，以便给出统一的处理问题方法和建议的解决方案。

1.4 相关资源

机器学习的最新研究成果发表在不同领域的会议和期刊上。机器学习专门的期刊有《Machine Learning》(机器学习)和《Journal of Machine Learning Research》(机器学习研究)。像《Neural Computation》(神经计算)、《Neural Networks》(神经网络)以及《IEEE Transactions on Neural Networksand Learning Systems》(IEEE 神经网络和学习系统汇刊)这样的期刊也发表了有关大量机器学习的论文。统计学方面的期刊，如《Annals of Statistics》(统计学年鉴)和《Journal of the American Statistical Association》(美国统计学会杂志)也会发表一些机器学习方面的文章，并且许多《IEEE Trousactions》，如《Pattern Analysis and Machine Intelligence》(IEEE 模式分析与机器智能汇刊)、《Systems，Man，and Cybernetics》(系统、人和控制论)、《Image Processing》(IEEE 图像处理汇刊)和《Signal Processing》(IEEE 信号处理汇刊)都有一些涉及机器学习的理论和它应用的有趣论文。

关于人工智能、模式识别和信号处理方面的期刊也包含机器学习方面的文章。以数据挖掘为主的期刊有《Data Mining and Knowledge Discovery》(数据挖掘与知识发现)、《IEEE Transactions on Knowledge and Data Engineering》(IEEE 知识与数据工程汇刊)以及《ACM Special Interest Group on Knowledge Discovery and Data Mining Explorations Journal》(ACM 知识发现和数据挖掘特别兴趣组期刊)。

关于机器学习方面的主要会议有"*Neural Information Processing Systems*(NIPS)"、"*Uncertainty in Artificial Intelligence*(UAI)"、"*International Conference on Machine Learning*(ICML)"、"*European Conference on Machine Learning*(ECML)"以及"*Computational Learning Theory*(COLT)"。模式识别、神经网络、人工智能、模糊逻辑和遗传算法方面的会议，以及关于计算机视觉、语音技术、机器人和数据挖掘等应用方面的会议，也会有针对机器学习的专题。

网站 http://www.ics.uci.edu/~mlearn/MLRepository.html 上的 UCI Repasitory 包含大量数据集，致力于机器学习的研究者经常把它们作为性能评价基准。另一个资源是网站 http://lib.stat.cmu.edu 上的 Statlib。此外，还有一些针对特定应用的数据库，例如，针对计算生物学、人脸识别、语音识别等。

新的、更大的数据集不断地添加到这些库中。但是，有些研究者仍然相信这些库的范围有限，不能反映实际数据的全部特征，因此在这些库中的数据集上的准确性并不说明问
17 题。甚至可以说，当反复使用固定库中的数据集并量身打造新算法时，我们正在产生针对这些数据集的一组新的"UCI 算法"。这就像仅通过解决一组实例问题来学习一门课程的学生。正如我们将在后面的章节中所看到的，不同的算法在不同的任务上会好一些，因此最好是针对一种应用，为该应用抽取一个或一些大型数据集，并针对特定的任务，在这些数据集上进行算法比较。

机器学习研究者近期的大多数文章都可以从因特网上找到，大部分作者还在网站上提供了他们的程序和数据。机器学习会议和暑期班上的辅导讲座也多半可以获取。还有一些实现各种机器学习算法的免费工具箱和软件包，其中 http://www.cs.waikato.ac.nz/ml/weak/上的 Weka 特别值得关注。

1.5 习题

1. 设想你有两种选择：可以扫描并传送图像；或者先使用光学字符阅读器(OCR)，然后再传送相应的文本文件。用对比方式讨论这两种方法的优缺点。在什么时候一种方法比另一种方法更可取？
2. 假定我们正在构建一个 OCR，并且对于每一字符，我们都存储该字符的位图作为与逐个像素读取的字符进行匹配的模板。请解释什么时候这样的系统会失败。为什么条码阅读器目前仍在使用？

   解：在这种系统中，每个字符只能有一个模板，并且不能识别来自多种字体的字符。存在 OCR-A 和 OCR-B 这样的标准字体(通常在我们购买的资料包装上看到的字体)，它们与 OCR 软件一起使用(这些字体的字符被稍加改变，以便使得它们之间的相似性最小)。条码阅读器仍然在使用，因为与阅读任意字体、字号和样式的字符相比，它仍然更好(更便宜、更可靠、更可用)。
3. 假定我们的既定目标是构建识别垃圾邮件的系统。请问是垃圾邮件中的什么特征使我们能够确认它为垃圾邮件？计算机如何通过语法分析来发现垃圾邮件？如果发现了垃圾邮件，你希望计算机如何处理它：自动删除？转到另一个文件夹？还是仅仅在屏幕上标亮显示？

   解：通常，基于文本的垃圾邮件过滤器检查邮件中是否有某些词或符号。像“机会”(opportunity)、“伟哥”(viagra)、“美元”(dollar)这样的词，以及像“$”和“!”这样的字符提高了邮件是垃圾邮件的概率。这些概率从用户先前已经标记为垃圾邮件的过去邮件样例的训练集中学习。在后面的章节中，我们会看到许多这样的算法。 18

   垃圾邮件过滤器没有 100%的可靠性，可能在分类时出错。如果有一个垃圾邮件没有被过滤掉，那么不太好，但是总比把好邮件当作垃圾邮件过滤掉好。稍后我们将讨论如何考虑这种假正和假负的相对代价。因此，不应该自动删除系统认为是垃圾邮件的信息，而是应该把它们放在一旁，使得如果用户愿意的话用户可以看到它们，特别是在使用垃圾邮件过滤器的早期阶段，系统训练尚不充分时尤其如此。垃圾邮件过滤可能是机器学习的最好应用领域之一，学习系统可以自动地适应垃圾邮件信息产生方式的变化。
4. 假设给定的任务是制造自动出租车，请定义约束。输入是什么？输出是什么？如何与乘客沟通？需要与其他的自动出租车沟通，即需要某种语言吗？
5. 在购物篮分析中，我们希望找出产品 X 和 Y 二者之间的依赖关系。对于给定的顾客交易数据库，如何能够发现这些数据之间的依赖关系？如何将依赖关系发现算法推广到多于两个的产品之间？
6. 在你的日报中，为政治、体育和艺术类各找出 5 个新闻报道样例。阅读这些报道，找出每类报道频繁使用的词，这些词可能帮助我们区别不同的类别。例如，政治方面的新闻报道多半会包含“政府”、“经济衰退”、“国会”等词，而在艺术类的新闻报道中可能包括“专辑”、“油画”或“剧院”等词。还有一些词(如“目标”)是模棱两可的。
7. 如果人脸图像是 100×100 的图像，按行写出，则它是一个 10 000 维向量。如果我们把图像向右移动一个像素，则将得到 10 000 维空间中的一个很不相同的向量。如何构造一个对于这种扰动具有鲁棒性人脸识别器？

   解：通常，人脸识别系统都有一个用于输入标准化的预处理阶段，在识别之前，

将输入中间对齐，并且可能调整大小。一般通过先找出眼睛，然后相应地变换图像来实现。还有一些识别程序不把图像看作像素，而是从图像中提取结构特征。例如，提取两眼间距离与整张脸的大小之比。对于变换和尺寸变化，这种特征具有不变性。

8. 取一个词，例如“machine”。写 10 次，请一位朋友也写 10 次。分析这 20 个图像，试找出区分你与朋友手书的特征、笔画类型、曲度、圆和如何画点等。
19
9. 在估计二手车的价格时，估计它相对于原价的折旧率，而不是估计它的绝对价格更有意义。为什么？

1.6 参考文献

Bishop, C. M. 1995. *Neural Networks for Pattern Recognition.* Oxford: Oxford University Press.

Duda, R. O., P. E. Hart, and D. G. Stork. 2001. *Pattern Classification,* 2nd ed. New York: Wiley.

Han, J., and M. Kamber. 2011. *Data Mining: Concepts and Techniques,* 3rd ed. San Francisco: Morgan Kaufmann.

Hand, D. J. 1998. “Consumer Credit and Statistics.” In *Statistics in Finance,* ed. D. J. Hand and S. D. Jacka, 69–81. London: Arnold.

Hastie, T., R. Tibshirani, and J. Friedman. 2011. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction,* 2nd ed. New York: Springer.

McLachlan, G. J. 1992. *Discriminant Analysis and Statistical Pattern Recognition.* New York: Wiley.

Russell, S., and P. Norvig. 2009. *Artificial Intelligence: A Modern Approach,* 3rd ed. New York: Prentice Hall.

Webb, A., and K. D. Copsey. 2011. *Statistical Pattern Recognition,* 3rd ed. New York: Wiley.

Witten, I. H., and E. Frank. 2005. *Data Mining: Practical Machine Learning Tools and Techniques,* 2nd ed. San Francisco: Morgan Kaufmann.

20

监督学习

我们从最简单的情况开始来讨论监督学习，首先从正例和负例的集合中学习类别，继而推广并讨论多类的情况，然后再讨论输出为连续值的回归。

2.1 由实例学习类

假设我们要学习“家用汽车”类 C。现在有一组汽车实例和一组看过这些汽车的被调查的人。被调查的人观察汽车并标记它们，将他们认为的家用汽车标为*正例*(positive example)，其他标为*负例*(negative example)。类学习就是寻找一个涵盖所有的正例而不涵盖任何负例的描述。这样做，我们可以做预测：给定一辆我们以前从未见过的汽车，检查学习得到的描述，我们就可以判断这辆汽车是否为家用汽车。我们还可以进行知识提取。这种研究可能由汽车公司赞助，目的可以是为了理解人们对家用汽车的期望。

经过与该领域专家沟通，假定我们得到了一个结论：在我们所掌握的汽车的所有特征中，区别家用汽车与其他汽车的特征是价格和发动机功率。这两个属性就是类识别器的*输入*(input)。注意，当我们决定采用这种特殊的*输入表示*(input representation)时，我们忽略其他属性，将它们看作不相关的。尽管有人可能认为座位数量、车身颜色等属性对于辨别车型也很重要，但是这里为了简单起见，我们只考虑价格和发动机功率。

我们假设价格为第一个输入属性 x_1(比如以美元计算)，发动机功率为第二个输入属性 x_2(比如以立方厘米计发动机排量)。这样，每辆汽车就可以用两个数值来表示，

$$\boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \tag{2-1}$$

而它的标号表示汽车的类型

$$r = \begin{cases} 1 & \text{如果 } \boldsymbol{x} \text{ 是正例} \\ 0 & \text{如果 } \boldsymbol{x} \text{ 是负例} \end{cases} \tag{2-2}$$

每辆汽车都用一个这种有序对($\boldsymbol{x}$，r)来表示，而训练集中包括 N 个这样的实例，

$$\mathcal{X} = \{\boldsymbol{x}^t, r^t\}_{t=1}^{N} \tag{2-3}$$

其中，t 用于标记训练集中的各个汽车实例，它不表示时间或任何类似的序。

现在，我们的训练数据可以绘制在二维空间(x_1，x_2)上，其中每个实例 t 是一个数据点，坐标为(x_1^t，x_2^t)，其类型(即正或负)由 r^t 给定(参见图 2-1)。

通过进一步与专家讨论和分析数据，我们有理由相信，对于家用汽车，其价格和发动机功率应当是在某个确定的范围内：

$$(p_1 \leqslant \text{价格} \leqslant p_2) \quad \text{AND} \quad (e_1 \leqslant \text{发动机功率} \leqslant e_2) \tag{2-4}$$

其中 p_1，p_2，e_1和 e_2为适当的值。式(2-4)假定类C是价格-发动机功率空间中的矩形(参见图 2-2)。

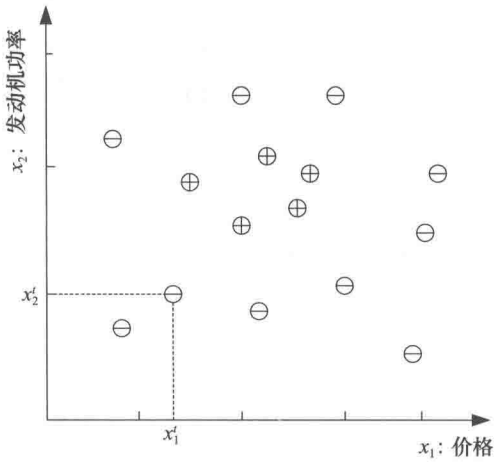


图 2-1 “家用汽车”类的训练集。其中每个点代表一个汽车实例，点的坐标值分别表示汽车的价格和发动机功率。“+”表示正例(家用汽车)，“−”表示负例(非家用汽车)，即其他类型的汽车

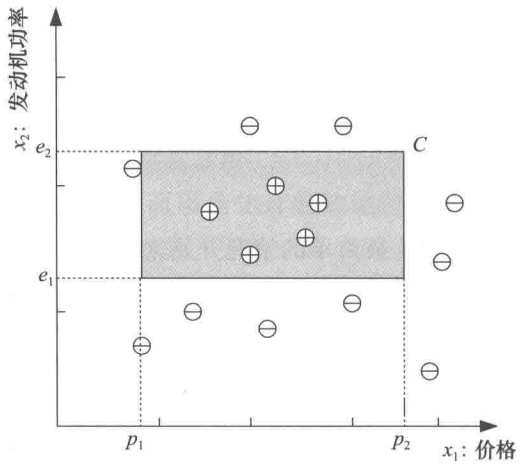


图 2-2 假设类的实例。家用汽车类是价格和发动机功率空间中的矩形

式(2-4)确定了假设类(hypothesis class)$\mathcal{H}$(即矩形的集合)，我们相信C是从中抽取的。学习算法应当找到一个由特定 4 元组(p_1^h，p_2^h，e_1^h，e_2^h)指定的特定的假设(hypothesis)$h\in\mathcal{H}$，尽可能地逼近C。

尽管专家定义了假设类，但是参数值是未知的。换句话说，尽管我们选定了$\mathcal{H}$，但是我们不知道哪个特定的$h\in\mathcal{H}$等于或最接近C。然而，一旦我们把注意力局限于这个假设类，学习类就归结为较简单的问题——找出定义 h 的 4 个参数。

我们的目标是找出$h\in\mathcal{H}$，它与C尽可能类似。假设 h 对实例 $\boldsymbol{x}$ 进行预测，使得

$$h(\boldsymbol{x})=\begin{cases}1 & \text{如果 } h \text{ 将 } \boldsymbol{x} \text{ 分类为正例}\\ 0 & \text{如果 } h \text{ 将 } \boldsymbol{x} \text{ 分类为负例}\end{cases} \tag{2-5}$$

实际上，我们并不知道$C(\boldsymbol{x})$，因此无法评估$h(\boldsymbol{x})$与$C(\boldsymbol{x})$的匹配程度。我们所拥有的是训练集$\mathcal{X}$，它是所有可能的 $\boldsymbol{x}$ 的一个小子集。经验误差(empirical error)是 h 的预测值(prediction)不同于$\mathcal{X}$中给定的预期值(required value)的训练实例所占的比例。对于给定的训练集$\mathcal{X}$，假设 h 的误差是

$$E(h\mid\mathcal{X})=\sum_{t=1}^{N}1(h(\boldsymbol{x}^t)\neq r^t) \tag{2-6}$$

其中，当$a\neq b$时$1(a\neq b)$为 1，当$a=b$时该值为 0(参见图 2-3)。

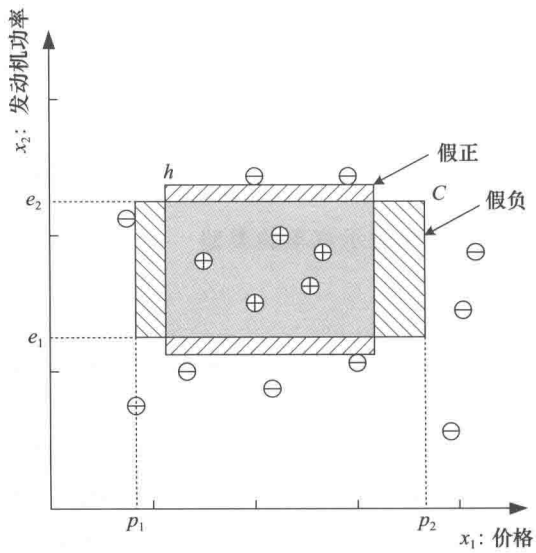


图 2-3 C是实际的类，h 是我们的诱导假设。C为 1 而 h 为 0 的点为假负，C为 0 而 h 为 1 的点为假正。其他点，即真正和真负，都被正确地分类

在我们的例子中，假设类$\mathcal{H}$是所有可能矩形的集合。每个四元组(p_1^h，p_2^h，e_1^h，e_2^h)都

定义了$\mathcal{H}$中的一个假设h，而我们需要选择其中最好的一个。换句话说，给定训练集，我们需要找出这4个参数的值，使它涵盖所有的正例而不包括任何负例。注意，如果x_1和x_2都是实数值，则存在无穷多个h满足上述条件，也就是说，对于这些h，误差E为0。但是，给定一个接近于正例和负例边界的某个未来实例，不同的候选假设可能做出不同的预测。这是泛化(generalization)问题，即我们的假设对不在训练集中的未来实例分类准确率如何。

一种可能的策略是找出最特殊的假设(most specific hypothesis)S，它是涵盖所有正例而不包括任何负例的最紧凑的矩形(参见图2-4)。这样就得出一个假设$h=S$作为诱导类(induced class)。注意，实际的类C可能比S更大但绝不会更小。最一般的假设(most general hypothesis)G是涵盖所有正例而不包括任何负例的最大矩形(参见图2-4)。对于任何介于S和G之间的$h\in\mathcal{H}$，h为无误差的有效假设，称作与训练集相容(consistent)，并且这样的h形成解空间(version space)。给定另一个训练集，S、G、解空间、参数和因此学习得到的假设h可能不同。

实际上，依赖于训练集$\mathcal{X}$和假设类$\mathcal{H}$，可能存在多个S_i和G_j，它们分别形成S集和G集。S集中的每个假设都与所有的实例相容，并且不存在更特殊的相容假设。类似地，G集中的每假设都与所有的实例相容，并且不存在更一般的相容假设。这两个集合形成边界集，它们之间的任何假设都是相容的，并且是解空间的一部分。存在一个称作候选删除的算法，随着逐个看到训练实例，它增量地更新S集和G集，见Mitchell 1997。我们假定$\mathcal{X}$足够大，存在唯一的S和G。

给定$\mathcal{X}$，我们可以找到S或G，或解空间中的任意h，并将它作为假设h。直观地，h应该选取S与G的中间，这将增大边缘(margin)，而边缘是边界和与它最近的实例之间的距离(参见图2-5)。为了使误差函数在具有最大边缘的h上最小化，应该选择这样的误差(损失)函数：它不仅检查实例是否在边界的正确一侧，而且还要指出实例离边界多远。也就是说，取代返回0/1的$h(\boldsymbol{x})$，我们需要一个返回携带$\boldsymbol{x}$到边界距离度量值的假设，并且需要一个使用该值的不同于检查相等性$1(\cdot)$的损失函数。

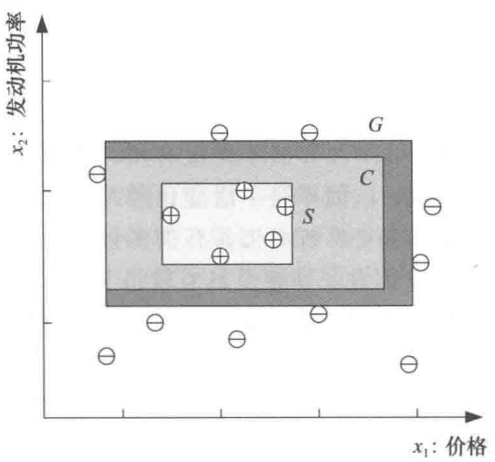


图2-4 S是最特殊的假设，G是最一般的假设

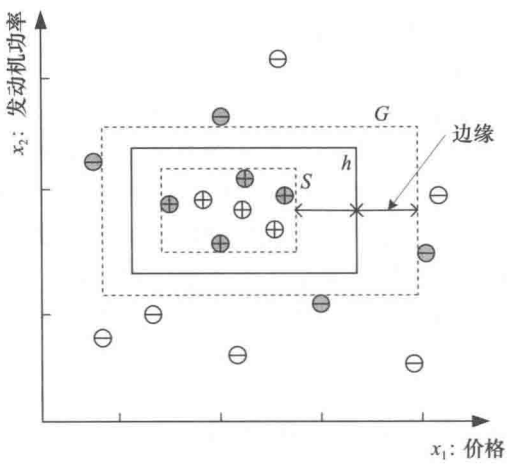


图2-5 为了获得最佳分离，我们选择具有最大边缘的假设。带阴影的实例是定义(支撑)边缘的实例。可以删除其他实例，而不会影响h

在某些应用中，错误的决策可能代价很高，并且落在S和G之间的实例都是不确定的

(doubt)实例，由于缺乏数据支持，这些不确定实例无法被确定地标注。在这种情况下，系统将拒绝(reject)考虑这些实例，并留待人类专家来判定。

这里，我们假定$\mathcal{H}$包含C，即存在$h\in\mathcal{H}$，使得$E(h|X)$为0。给定假设类$\mathcal{H}$，可能存在不能学习C的情况，即不存在$h\in\mathcal{H}$，使得误差为0。因此，对于任何应用，我们需要确保$\mathcal{H}$有足够的柔性，或$\mathcal{H}$具有足够的“能力”学习C。

2.2 VC维

假定有一个包含N个点的数据集。这N个点可以用2^N种方法标记为正例和负例。因此，N个数据点可以定义2^N种不同的学习问题。如果对于这些问题中的任何一个，都能够找到一个假设$h\in\mathcal{H}$将正例和负例分开，那么就称$\mathcal{H}$散列(shatter)N个点。也就是说，由N个点定义的任何学习问题都能用一个从$\mathcal{H}$抽取的假设无误差地学习。可以被$\mathcal{H}$散列的点的最大数量称为$\mathcal{H}$的VC维(Vapnik-Chervonenkis dimension)，记为VC($\mathcal{H}$)，它度量假设类$\mathcal{H}$的学习能力。

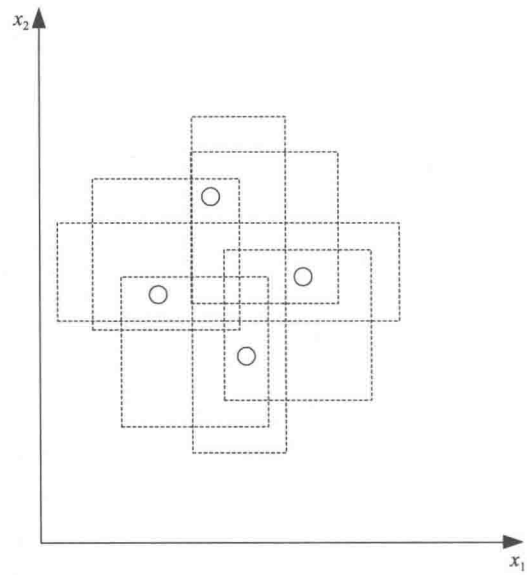


图2-6 轴平行的矩形能够散列4个点，其中只显示了覆盖两个点的矩形

在图2-6中，我们可以看到轴平行的矩形能够散列二维空间中的4个点。当$\mathcal{H}$为二维空间中轴平行的矩形的假设类时，VC($\mathcal{H}$)等于4。在计算VC维时，找到4个被散列的点就够了，没有必要散列二维空间中的任意4个点。例如，位于同一条直线上的4个点不能被矩形散列。然而，我们无法在二维空间的任何位置设置5个点，使得对于所有可能的标记，一个矩形能够分开正例和负例。

也许VC维看起来比较悲观，它告诉我们使用矩形作为假设类，我们只能学习包括但不多于4个点的数据集。能够学习含有4个点的数据集的学习算法不是很有用。然而，这是因为VC维独立于数据实例的概率分布。在实际生活中，世界是平滑变化的，在大多数时候相近的实例具有相同的标记，我们并不需要担心所有可能的标记。有很多包含远不止4个点的数据集都可以通过假设类来学习(参见图2-1)。因此，即便是具有较小VC维的假设类也是有应用价值的，并且比那些较大VC维的假设类(例如，具有无穷VC维的查找表)更可取。

22
~
28

2.3 概率近似正确学习

使用最紧凑的矩形S作为假设，希望找出需要多少实例。我们希望假设是近似正确的，即误差概率不超过某个值。我们还要对假设有信心，因为我们想知道假设在大多数时候都是正确的(如果并非总是正确的话)。因此，我们希望假设很可能(以我们可能指定的概率)是正确的。

在概率近似正确(Probably Approximately Correct，PAC)学习中，给定类C和从未知但具有固定概率分布 $p(x)$中抽取的样本，我们希望找出样本数 N，使得对于任意的 $\delta \leqslant 1/2$ 和 $\varepsilon > 0$，假设 h 的误差最多为 ε 的概率至少为 $1-\delta$

$$P\{C \Delta h \leqslant \varepsilon\} \geqslant 1-\delta$$

其中$C \Delta h$ 是C与 h 不同的区域。

在这种情况下，因为 S 是最紧凑的可能的矩形，所以C与 $h=S$ 之间的误差区域是 4 个矩形条带之和(参见图 2-7)。我们希望确保正例落在该区域(导致错误)的概率最多为 ε。对于任何这样的条带，如果我们能够确保该概率的上界为 $\varepsilon/4$，则误差最多为 $4(\varepsilon/4)=\varepsilon$。注意，我们将矩形角部的重叠部分计算了两次，并且这种情况下总的实际误差小于 $4(\varepsilon/4)$。随机抽取的样本不在此条带中的概率是 $1-\varepsilon/4$。所有 N 个独立抽取的样本不在该条带中的概率为 $(1-\varepsilon/4)^N$，而所有 N 个独立抽取的样本不在这 4 个矩形条带中的概率最多为 $4(1-\varepsilon/4)^N$，我们希望它最多为 δ。有不等式

$$(1-x) \leqslant \exp[-x]$$

因此，如果选定 N 和 δ 满足

$$4\exp[-\varepsilon N/4] \leqslant \delta$$

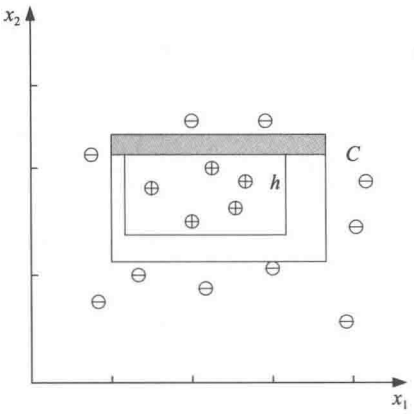


图 2-7 h 与C之差是 4 个矩形条带之和，其中一个用阴影显示

则有 $4(1-\varepsilon/4)^N \leqslant \delta$。不等式两边同时除以 4，再取(自然)对数，并重新排列各项，得到

$$N \geqslant (4/\varepsilon)\log(4/\delta) \tag{2-7}$$

因此，只要我们至少从C中取$(4/\varepsilon)\log(4/\delta)$个独立样本，并使用紧凑矩形作为假设 h，则在置信概率(confidence probability)至少为 $1-\delta$ 的情况下，一个给定点被误分类的错误概率(error probability)最多为 ε。减少 δ，可以有任意大的置信度；而减少 ε，可以有任意小的误差，并且我们在式(2-7)中看到，样本的数量是分别随 $1/\varepsilon$ 和 $1/\delta$ 呈线性和对数缓慢增长的函数。

2.4 噪声

噪声(noise)是数据中有害的异常。由于噪声的存在，类的学习可能更加困难，并且使用简单的假设可能做不到零误差(参见图 2-8)。噪声有以下几种解释：

- 记录的输入属性可能不准确，这可能导致数据点在输入空间中移动。
- 标记的数据点可能有错误，可能将正例标记为负的，或相反。这种情况有时称为指导噪声(teacher noise)。
- 可能存在没有考虑到的附加属性，而它们会影响实例的标记。这些附加属性可能是隐藏的(hidden)或潜在的(latent)，因此可能是不可观测的。这些被忽略的属性所造成的影响作为随机成分建模，是“噪声”的一部分。

如图 2-8 所示，当有噪声时，在正例与负实例之间不存在简单的边界，并且为了将它们分开，需要对应于具有更大能力的假设类的复杂假设。矩形可以用 4 个数定义，但为了定义更复杂的形状，需要具有大量参数的更复杂的模型。利用这些复杂模型，可以更好地拟合数据，得到零误差(参见图 2-8 中的曲线图形)。另一种可行的方法是保持模

29
⌇
30

型的简单性并允许一些误差存在(参见图 2-8中的矩形)。

使用简单矩形(除非其训练误差很大)更有意义，原因如下：

1) 矩形是一种容易使用的简单模型。容易检查一个点是在矩形内还是在矩形外，并且对于未来的数据实例，可以容易地检查它是正例还是负例。

2) 矩形是一种容易训练的简单模型，并且具有较少的参数。相对任意图形的控制点来说，比较容易找到矩形的拐角值。利用小的训练集，当训练实例有少许差异时，我们预料简单模型比复杂模型变化小一些：简单模型具有更小的*方差*(variance)。另一方面，太简单的模型假设更多、更严格，并且如果基础类(underling class)并非那么简单，模型预测就可能失败：较简单的模型具有较大的*偏倚*(bias)。求解最优模型相当于最小化偏倚和方差。

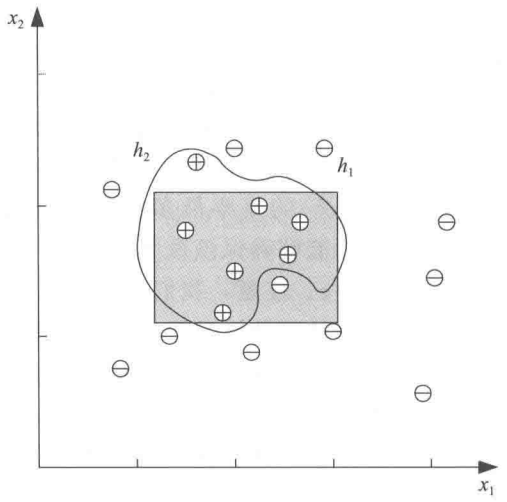


图 2-8 当有噪声时，在正例和负例之间不存在一个简单的边界，使用简单假设也许不可能达到零误差的分类结果。矩形是具有 4 个定义拐角的参数的简单假设。使用大量控制点的分段函数能够导出任意的闭合图形

3) 矩形是一种容易解释的简单模型。矩形简单地对应于两个属性上定义的区间。通过学习简单模型，能够从给定训练集的原始数据中提取信息。

4) 如果输入数据中确实存在错误标记的实例或噪声，并且实际的类确实就是矩形这样的简单模型，那么由于矩形具有较小的方差，并且较少地被单个实例所影响，所以尽管简单矩形可能导致训练集上较大的误差，但是它也是比曲线图形更好的分类器。给定类似的经验误差，我们说简单(但不是太简单的)模型比复杂模型泛化能力更好。该原则就是著名的*奥克姆剃刀*(Occam's razor)，它说*较简单的解释看上去更可信*，并且任何不必要的复杂性都应该被摒弃。

2.5 学习多类

31
~
32

在学习家用汽车的例子中，我们有属于家用汽车类的正例和属于其他所有汽车类的负例。这是一个*两类*(two-class)问题。通常情况下，有 K 个类，记为 $C_i(i=1, \cdots, K)$，并且每个输入实例都严格地属于其中一个类。训练集形如

$$\mathcal{X} = \{\boldsymbol{x}^t, \boldsymbol{r}^t\}_{t=1}^N$$

其中 $\boldsymbol{r}$ 是 K 维的，并且

$$r_i^t = \begin{cases} 1 & \text{如果 } \boldsymbol{x}^t \in C_i \\ 0 & \text{如果 } \boldsymbol{x}^t \in C_j, j \neq i \end{cases} \tag{2-8}$$

一个例子在图 2-9 中给出，其中实例来自 3 个类：家用汽车、运动汽车和豪华轿车。

在用于分类的机器学习中，我们希望学习将一个类与所有其他类分开的边界。这样，我们把 K 类的分类问题看作 K 个两类问题。属于 C_i 类的训练实例是假设 h_i 的正例，属于所有其他类的训练实例是假设 h_i 的负例。因此，在 K 类问题中，我们要学习 K 个假设，使得

$$h_i(\boldsymbol{x}^t)=\begin{cases}1 & \text{如果 } \boldsymbol{x}^t \in C_i \\ 0 & \text{如果 } \boldsymbol{x}^t \in C_j, j \neq i\end{cases} \tag{2-9}$$

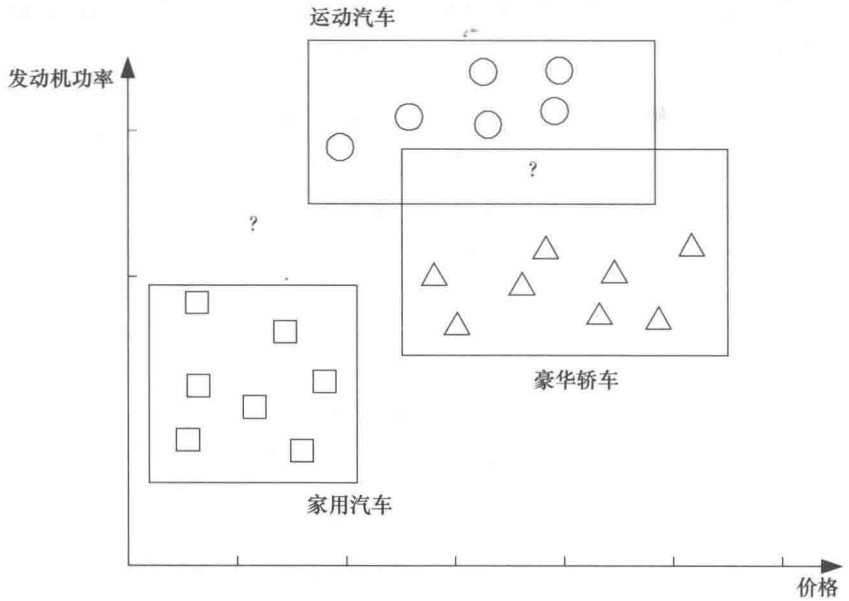


图 2-9 有 3 类：家用汽车、运动汽车和豪华轿车。有 3 个归纳的假设，每个假设覆盖一个类的实例而不包括另外两个类的实例。“?”为拒绝区域，其中没有类或有多个类被选中

整体经验误差对所有类在所有实例上的预测上取和：

$$E(\{h_i\}_{i=1}^K \mid \mathcal{X})=\sum_{t=1}^{N}\sum_{i=1}^{K}1(h_i(\boldsymbol{x}^t)\neq r_i^t) \tag{2-10}$$

在理想情况下，对于给定的 $\boldsymbol{x}$，只有其中一个假设 $h_i(\boldsymbol{x})(i=1, \cdots, K)$为 1，并且我们能够选定一个类。但是，当没有或者有两个或者更多的 $h_i(\boldsymbol{x})$为 1 时，我们就无法选定一个类，这是不确定(doubt)情况并且分类器拒绝这种情况。

在学习家用汽车的例子中，只用了一个假设，并且只对正例建模。任何未包括在其中的负例都不是家用汽车。作为另一种选择，有时我们可能更倾向于构建两个假设，一个是对正例，另一个是对负例。这也为被另一个假设所覆盖的负例假定一个结构。将家用汽车与运动汽车分开就是一种这样的问题，每个类都有其自己的结构。这种处理的优点在于，如果输入的是一个豪华轿车，我们就能够通过两个假设来判定其为负例并拒绝该输入。

如果我们预料数据集中所有类的结构(在输入空间中的形状)都类似，则可以对所有类使用相同的假设类。例如，在手写数字识别数据集中，我们预料所有数字都具有类似的分布。但是，在医疗诊断数据集中，有病人和健康人两个类，这两个类可能具有完全不同的分布。一个人是病人可能有不同原因，反映在输入中的不同：所有健康的人都是相似的，而每个病人都有他们自己的病情。

2.6 回归

在分类问题中，给定一个输入，所产生的输出是一个布尔值，这是一个是/否类型的答案。当输出是数值时，我们希望学习的不是一个类$C(\boldsymbol{x})\in\{0, 1\}$，而是一个数值函数。

在机器学习中，该函数是未知的，但我们有从其中抽取样本的训练集

$$\mathcal{X} = \{\boldsymbol{x}^t, r^t\}_{t=1}^N$$

其中$r^t \in \Re$。如果不存在噪声，则任务是插值(interpolation)。我们希望找到通过这些点的函数$f(\boldsymbol{x})$，使得

$$r^t = f(\boldsymbol{x}^t)$$

在多项式插值(polynomial interpolation)中，给定N个点，找出可以用来预测$\boldsymbol{x}$的任意输出的$(N-1)$阶多项式。如果$\boldsymbol{x}$落在训练集中$\boldsymbol{x}^t$的值域之外，则该方法称为外插或外推(extrapolation)。例如，在时间序列预测中，我们拥有最新的数据，希望预测未来的值。在回归(regression)分析中，噪声添加到未知函数的输出

$$r^t = f(\boldsymbol{x}^t) + \varepsilon \tag{2-11}$$

其中，$f(\boldsymbol{x}) \in \Re$是未知函数，而$\varepsilon$是随机噪声。对噪声的解释是，存在我们无法观察到的额外隐藏(hidden)变量

$$r^t = f^*(\boldsymbol{x}^t, \boldsymbol{z}^t) \tag{2-12}$$

其中$\boldsymbol{z}^t$表示这些隐藏变量。我们希望通过模型$g(\boldsymbol{x})$来逼近输出。训练集$\mathcal{X}$上的经验误差是

$$E(g \mid \mathcal{X}) = \frac{1}{N}\sum_{t=1}^{N}[r^t - g(\boldsymbol{x}^t)]^2 \tag{2-13}$$

因为r和$g(\boldsymbol{x})$是数值量(例如，属于$\Re$)，所以存在定义在它们值域上的序，而且我们可以定义值之间的距离(distance)作为差的平方。相对于分类使用的相等或不等来说，距离为我们提供了更多的信息。差的平方是一种可以使用的误差(损失)函数。另一种误差函数是差的绝对值。在后续章节中，我们将看到其他的例子。

我们的目标是找到最小化经验误差的$g(\cdot)$。而且我们的方法是相同的。我们对$g(\cdot)$假定一个具有少量参数的假设类。如果假定$g(\boldsymbol{x})$是线性的，则有

35

$$g(\boldsymbol{x}) = w_1 x_1 + \cdots + w_d x_d + w_0 = \sum_{j=1}^{d} w_j x_j + w_0 \tag{2-14}$$

现在，再回到1.2.3节的例子，那里我们估计二手车的价格。当时我们使用单个输入的线性模型

$$g(x) = w_1 x + w_0 \tag{2-15}$$

其中，w_1和w_0是需要从数据中学习的参数。w_1和w_0的值应该使下式最小化

$$E(w_1, w_0 \mid \mathcal{X}) = \frac{1}{N}\sum_{t=1}^{N}[r^t - (w_1 x^t + w_0)]^2 \tag{2-16}$$

它可最小点可以通过求E关于w_1和w_0的偏导数，令偏导数为0，并求解这两个未知量来计算：

$$w_1 = \frac{\sum_t x^t r^t - \bar{x}\bar{r}N}{\sum_t (x^t)^2 - N\bar{x}^2}$$

$$w_0 = \bar{r} - w_1 \bar{x} \tag{2-17}$$

其中，$\bar{x} = \sum_t x^t / N$、$\bar{r} = \sum_t r^t / N$。找到的直线如图1-2所示。

如果线性模型过于简单，则它就会太受限制，导致大的近似误差，在这种情况下，输出可以取输入的较高阶的函数，如二次函数

$$g(x) = w_2 x^2 + w_1 x + w_0 \tag{2-18}$$

这里类似地，我们有参数的解析解。当多项式的阶增加时，训练数据上的误差将会降低。但是高阶多项式关注个体样本，而不是捕获数据的一般趋势(参见图 2-10 中的六次多项式)。这意味奥克姆剃刀也适用于回归，并且当精确调整的模型复杂度达到基础数据的函数的复杂度时，我们应该谨慎行事。

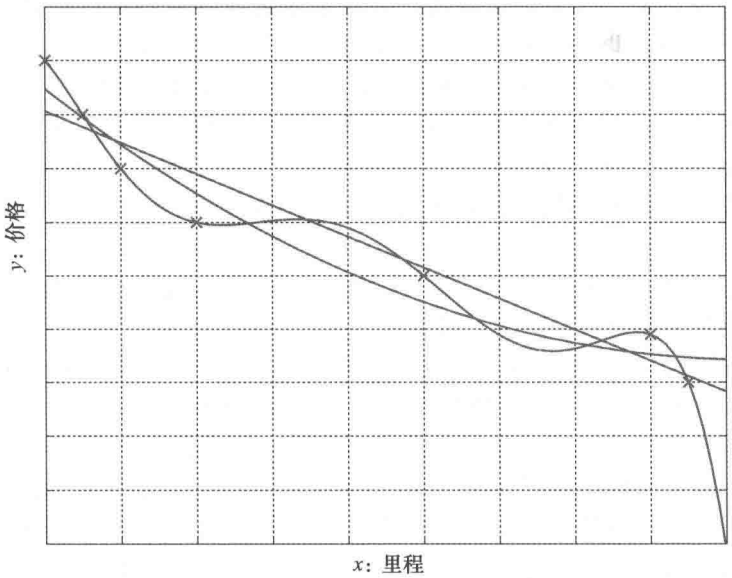


图 2-10 拟合相同数据点集的线性、二次和六次多项式。最高阶的多项式给出了正确的拟合，但是给定更多数据，真实的曲线很可能不是这种形状。二次多项式看起来比线性拟合好，它捕获了训练数据的走势

2.7 模型选择与泛化

我们用从实例学习布尔函数作为开始。在布尔函数中，所有的输入和输出均为二元的。d 个二元值有 2^d 种可能的写法。因此，对于 d 个输入，训练集最多有 2^d 个样本。如表 2-1所示，其中的每一位都能标记为 0 或 1，因而对于 d 个输入，有 2^{2^d} 个可能的布尔函数。

表 2-1 2 个输入存在 4 种可能的情况和 16 种可能的布尔函数

x_1	x_2	h_1	h_2	h_3	h_4	h_5	h_6	h_7	h_8	h_9	h_{10}	h_{11}	h_{12}	h_{13}	h_{14}	h_{15}	h_{16}
0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

每一个不同的训练样本都会去掉一半的假设，即去掉那些猜测是错的假设。例如，假定有 $x_1=0$，$x_2=1$，而输出为 0，这种情况就去掉了假设 h_5、h_6、h_7、h_8、h_{13}、h_{14}、h_{15} 和 h_{16}。这是解释学习的一种途径：随着我们看到更多的训练样例，我们逐步去掉那些与训练数据不一致的假设。在布尔函数的情况下，为了最终得到单个假设，我们需要看到所有的 2^d 个训练样本。如果给定的训练集只包含所有可能实例的一个小的子集(通常情况就是如此)，也就是说，如果我们仅对少数情况知道输出应该是什么，则解不是唯一的。看到 N 个样本后，还有 2^{2^d-N} 个可能的函数。这是一个不适定问题(ill-posed problem)，其中

36
~
37

仅依靠数据本身不足以找到唯一解。

在其他的学习应用中，在分类、回归中也存在同样的问题。随着我们看到更多的训练样本，我们对基础函数的了解就更多，并且我们从假设类去掉更多不一致的假设，但是我们还剩下许多一致的假设。

这样，由于学习是一个不适定问题，并且单靠数据本身不足以找到解，所以我们应该做一些特别的假设，以便得到已有数据的唯一解。我们所做的为了使得学习成为可能的假设集称为学习算法的*归纳偏倚*(inductive bias)。引入归纳偏倚的一种途径是假定一个假设类$\mathcal{H}$。在学习家用汽车类时，存在着无限种将正例与负例分开的方法。假定矩形形状是一种归纳偏倚，那么具有最大边缘的矩形就是另一种归纳偏倚。在线性回归中，假定线性函数是一种归纳偏倚，而在所有的直线中选择最小化平方误差的直线是另一种归纳偏倚。

然而，我们知道，每个假设类都有一定的能力，并且只能学习某个函数。使用具有更大能力、包含更复杂假设的假设类，可以扩充可学习的函数类。例如，假设类"两个矩形的并"具有更大的能力，但其假设也更复杂。类似地，在回归分析中，随着多项式阶的增大，其能力和复杂度也不断增加。现在的问题是决定在哪里停止。

因此，如果没有归纳偏倚，学习将是不可能的，而现在的问题是如何选择正确的偏倚。该问题称作*模型选择*(model selection)，即在可能的模型$\mathcal{H}$之间选择。对于这种问题的解答，我们应当记住机器学习的目标很少是复制训练数据，而是预测新情况。也就是
38 说，我们希望对于训练集之外的输入(其正确的输出并没有在训练集中给出)能够产生正确的输出。训练集上训练的模型如何能够对新的实例预测出正确的输出称为*泛化*(generalization)。

对于最好的泛化来说，我们应当使假设类$\mathcal{H}$的复杂度与基础数据的函数的复杂度相匹配。如果$\mathcal{H}$没有函数复杂，例如，当试图用直线拟合从三次多项式抽取的数据时，则是*欠拟合*(underfitting)。在这种情况下，随着复杂度的提高，训练误差降低。但是，如果$\mathcal{H}$太过复杂，数据不足以约束该假设，则我们最后可能得到不好的假设$h\in\mathcal{H}$。例如，当用两个矩形拟合从一个矩形抽取的数据时，这种情况就会发生。或者，如果存在噪声，则过分于复杂的假设可能不仅学习基础函数，而且也学习数据中的噪声，导致很差的拟合。例如，用六次多项式拟合从三次多项式抽样的噪声数据时，这种情况就会发生。这称为*过拟合*(overfitting)。在这种情况下，拥有更多的训练数据是有帮助的，但是只能在某种程度上有帮助。给定训练集和$\mathcal{H}$，可以找到最小化训练误差的$h\in\mathcal{H}$。但是，如果$\mathcal{H}$选择不好，则无论选择哪个$h\in\mathcal{H}$都得不到好的泛化。

我们可以引用*三元权衡*(triple trade-off)(Dietterich 2003)来总结我们的讨论。在所有的由样本数据训练的学习算法中，存在以下3种因素之间的平衡：

- 拟合数据假设的复杂度，即假设类的能力。
- 训练数据的总量。
- 在新的样本上的泛化误差。

随着训练数据量的增加，泛化误差降低。随着模型类$\mathcal{H}$的复杂度的增加，泛化误差先降低，然后开始增加。过于复杂的$\mathcal{H}$的泛化误差可以通过增加训练数据的总量来控制，但是只能达到一定程度。如果数据从直线抽样并且拟合高阶多项式，那么如果周围有训练数据的地方，则拟合将被限制在该直线附近；而在没有训练数据的地方，高阶多项式的行为可能难以预测。

如果我们访问训练集以外的数据，则我们就能够度量假设的泛化能力，即它的归纳偏倚的质量。我们通过将已有的训练集划分为两部分来模拟这一过程。我们使用一部分来做训练(即拟合一个假设)，而剩下的部分称作验证集(validation set)，它用来检验假设的泛化能力。也就是说，给定可能的假设类的集合$\mathcal{H}_i$，对于每一个集合我们在训练集上拟合最佳的$h_i \in \mathcal{H}_i$。假定训练集和验证集都足够大，则在验证集上最准确的假设就是最好的假设(即具有最佳归纳偏倚的假设)。这一过程称为交叉验证(cross-validation)。例如，为了找出多项式回归的正确的阶，给定多个不同阶的候选多项式，其中不同阶的多项式对应于不同的$\mathcal{H}_i$，我们在训练集上求出它们系数，在验证集上计算它们误差，并取具有最小验证误差的多项式作为最佳多项式。 39

注意，如果需要报告最佳模型的期望误差，就不应该使用验证误差。我们已经使用验证集来选择最佳模型，并且它实际上已经成为训练集的一部分。我们需要第三个数据集——检验集(test set)。检验集有时也称为发布集(publication set)，它包含在训练或验证阶段未使用过的数据。现实生活中也有类似的情况，例如我们选修一门课程：老师在讲授一门课程时，课堂上求解的例题构成了训练集，考试题目就是验证集，而我们在职业生涯中解决的问题则是检验集。

我们也不能一直使用相同的训练和验证集划分，因为一旦使用一次，验证集就实际上成为训练数据的一部分。这就像老师每年都使用相同的考试题一样。精明的学生会意识到不必听课，仅仅记住这些问题的答案即可。

一定要记住，我们使用的训练数据是一个随机样本。也就是说，对于相同的应用，如果我们多次收集数据，则我们将得到稍微不同的数据集，拟合的h也稍微不同，并且具有稍微不同的验证误差。或者，如果我们把固定的数据集划分成训练、验证和检验集，则依赖于如何划分，我们会有不同的误差。这些稍微的不同使我们可以估计多大的差别可以看作显著的(significant)而非偶然的。也就是说，在假设类$\mathcal{H}_i$和$\mathcal{H}_j$之间进行选择时，我们将在大量训练集和验证集上多次使用它们，并且检查h_i与h_j的平均误差之差是否大于多个h_i之间的平均差。在第19章，我们将讨论如何设计机器学习实验，利用有限的数据来回答我们的问题(例如，“最好的假设类是哪个?”)，以及如何分析实验结果，使得我们可以得到受随机性影响最少、统计显著的结论。 40

2.8 监督机器学习算法的维

现在，让我们来总结和归纳上述要点。我们有样本

$$\mathcal{X} = \{x^t, r^t\}_{t=1}^N \tag{2-19}$$

该样本是独立同分布的(independent and identically distributed，idd)；次序并不重要，所有的实例都取自相同的联合分布$p(x, r)$。t指示N个实例中的一个，x^t是任意维的输入，而r^t是相关联的预期输出。对于两类学习，r^t是0/1；对于$K(K>2)$类分类，r^t是一个K维二元向量(其中恰有一维为1，其他各维均为0)；在回归分析中，r^t是一个实数值。

我们的目标是使用模型$g(x^t|\theta)$来构建一个r^t的好的、有用的近似。为了达到预期目标，我们必须做出3个决定：

1) 学习所使用的模型(Model)，记作

$$g(x|\theta)$$

其中，$g(\cdot)$是模型，x是输入，θ是参数。

$g(\cdot)$定义假设类$\mathcal{H}$，而θ的特定值实例化一个假设$h \in \mathcal{H}$。例如，在类的学习中，我们把矩形当作模型，其4个坐标值构成了θ。在线性回归中，模型是输入的线性函数，其斜率和截距是从数据中学习的参数。模型(归纳偏倚)或$\mathcal{H}$由机器学习系统的设计者根据其应用知识背景决定，而假设h由学习算法利用取样于$p(x, r)$的训练集进行选择(调整参数)。

2) 损失函数(loss function)$L(\cdot)$计算期望输出r^t与θ近似值$g(x^t|Q)$之间的差(给定
41 参数θ的当前值)。近似误差(approximation error)或损失(loss)是各个实例的损失之和

$$E(\theta|\mathcal{X}) = \sum_t L(r^t, g(x^t|\theta)) \tag{2-20}$$

在输出为0/1的类学习中，$L(\cdot)$检测相等或不相等；在回归分析中，由于输出是数值，所以我们有关于距离的序信息，而一种可能性是使用差的平方。

3) 最优化过程(optimization procedure)求解最小化总误差的θ^*

$$\theta^* = \arg\min_\theta E(\theta|\mathcal{X}) \tag{2-21}$$

其中，arg min返回使E最小化的参数值。在多项式回归中，我们能够分析地求解最优化问题，但并不总是这种情况。使用其他模型和误差函数，最优化问题的复杂度就变得非常重要。我们特别感兴趣的是，无论它是否有对应于全局最优解的单个最小，还是有对应于局部最优解的多个最小。

为了做好上述工作，必须满足以下条件：第一，$g(\cdot)$的假设类应当足够大，即要有足够的能力，能够包含以噪声形式产生的$\mathcal{X}$表示的数据的未知函数。第二，必须有足够的训练数据，使得我们能够从假设类中识别正确(或足够好)的假设。第三，给定训练数据，我们应当有好的优化方法，以便找出正确的假设。

不同机器学习方法之间的区别或者在于它们假设的模型(假设类/归纳偏倚)不同，或者在于它们所使用的损失度量不同，或者在于它们所使用的最优化过程不同。我们将在后续的章节中看到更多的例子。

2.9 注释

Mitchell提出了解空间和候选排除算法，使得当样本实例逐一给出时，可以增量地构
42 建S和G；近期的评述可参见Mitchell 1997。矩形学习取自Mitchell 1997的习题2.4。Hirsh(1990)讨论了当实例受到少量噪声影响时，如何处理解空间。

有关机器学习最早的研究工作之一是Winston(1975)提出的“几乎错过”(near miss)思想。几乎错过是一个与正例非常相似的负例。用我们的术语，几乎错过就是可能落在S与G之间灰色区域的实例，该实例将会影响边缘，因而相对于普通的正例和负例来说，它们对学习可能更有用。靠近边界的实例是定义(或支撑)边界的实例，删除那些被许多具有相同标号包围的实例不会影响边界。

与此相关的思想是*主动学习*(active learning)，其中学习算法能够自己生成实例，并请求标记它们，而不是被动地被给定(Angluin 1988)(参见习题4)。

VC维早在20世纪70年代就已经由Vapnik和Chervonenkis提出，新近的相关资源是Vapnik 1995，其中他指出“没有什么比好的理论更实用”。像在其他科学领域一样，这在机器学习领域也得到了证实。你不必急于使用计算机。你可以通过思考，使用纸张、铅笔，也许还需要橡皮擦之类的东西，节省自己的时间，避免无用的编程。

PAC模型由Valiant(1984)提出，学习矩形的PAC分析来自Blumer等(1989)。一本涵盖PAC学习和VC维的计算学习理论的好教材是Kearns和Vazirani 1994。

近年来，求解模型拟合的优化问题的定义正在变得非常重要。曾经我们对从某随机状态开始收敛于最近似好解的局部下降方法感到相当满意，现在我们对证明问题是凸的(存在单个全局解)感兴趣(Boyd 和 Vandenberghe 2004)。随着数据集规模的增大，模型变得越来越复杂，我们还对优化过程收敛于解的速度感兴趣。

2.10 习题

1. 假定假设类是圆而不是矩形。参数是什么？这种情况下，如何计算圆假设的参数？如果是椭圆又如何？为什么用椭圆代替圆会更有意义？ 43

   解： 在假设类是圆的情况下，参数是圆心和半径(参见图 2-11)。然后，我们需要找出 S 和 G，其中 S 是包含所有正例的最紧凑的圆，而 G 是包含所有正例而不包含负例的最大的圆。在它们之间的任何圆都是相容的假设。

   使用椭圆比圆更有意义，因为两个轴不必有相同的尺度，并且椭圆有两个参数，表示两个轴上的宽度，而不是一个半径。实际上，价格与发动机功率是正相关的。汽车的价格趋向于随发动机功率的增加而增加，因此使用倾斜的椭圆更有意义。我们将在第 5 章看到这样的模型。

2. 设想假设类不是一个矩形而是两个(或 $m>1$ 个)矩形的联合，请问这种假设类的优点是什么？说明使用足够大的 m，任何类都能够由这种假设类表示。

   解： 当只有一个矩形时，所有的正例都应来自单个分组；使用多个矩形，例如两个矩形(参见图 2-12)，正例可以在输入空间形成两个可能不相交的簇。注意，每个矩形对应于两个输入属性上的合取，而有多个矩形对应于它们的析取。任何逻辑公式都可以写成合取的析取。在最坏情况下($m=N$)下，每个正例都有单独的矩形。

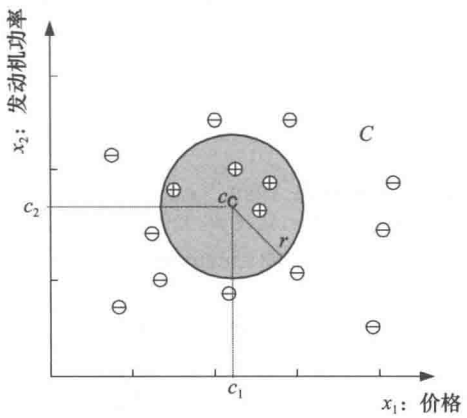


图 2-11 假设类是圆，有两个参数：圆心坐标和半径

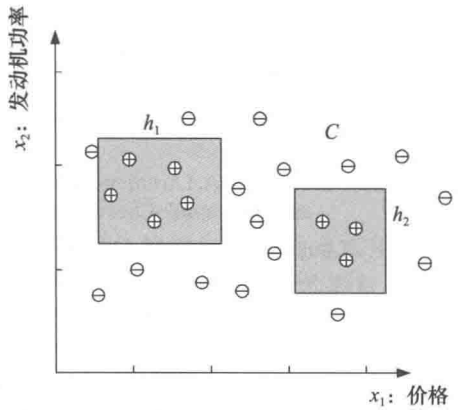


图 2-12 假设类是两个矩形的并

3. 在许多应用中，错误的决策(即假正和假负)都有资金成本，并且两种错误的成本可能不同。S 和 G 之间的 h 的位置与这两者的相对成本之间有什么联系？

   解： 可以看到，S 不会导致假正，而只会导致假负。类似地，G 不会导致假负，而只会导致假正。因此，如果假正与假负同样不好，则我们希望 h 在 S 和 G 的中间；如果假正的成本更大，则 h 应该更靠近 S；如果假负的成本更大，则 h 应该更靠近 G。

4. 大部分学习算法的复杂度都是训练集的函数。你能提出一个发现冗余实例的过滤算法吗？

   解： 影响假设的实例是那些处于具有不同标号的实例附近的实例。各个方向都被

许多正例包围的正例是不必要的；各个方向都被许多负例包围的负例也是不必要的。在第 8 章，我们将讨论这种基于近邻的方法。

5. 如果我们有能够给任何实例 $\boldsymbol{x}$ 提供标记的指导者，那么我们应当在哪里选择 $\boldsymbol{x}$，以便用较少的询问来进行学习？

44 ~ 45

解： 模糊区域是 S 和 G 之间的区域。最好在这里提问，使得我们可以缩小这种不确定的区域。如果给定的实例为正，则我们可以扩大 S 到该实例；如果它为负，则我们可以缩小 G 到该实例

6. 在式(2-13)中，我们对实际值与估计值的差的平方求和。该误差函数是一种使用最频繁的误差函数，但它只是多个可行的误差函数之一。由于它对差的平方求和，所以它对于离群点不是鲁棒的。为了实现鲁棒回归(robust regression)，更好的误差函数是什么？
7. 请推导式(2-17)。
8. 假定假设类是直线的集合，并且利用直线来隔开正例和与负例，而不是用矩形来界定正例，并将负例留在矩形外(参见图 2-13)。证明直线的 VC 维为 3。
9. 证明在二维空间中，三角形假设类的 VC 维为 7。(提示：为了最佳隔开，最好在某个圆上设置 7 个等距离的点)

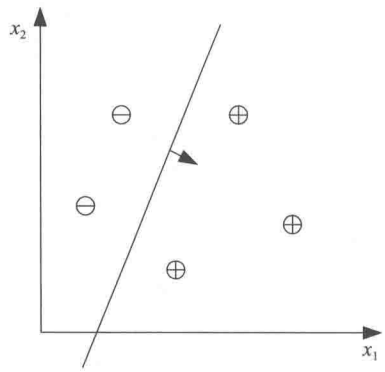


图 2-13 直线隔开正例与负例

10. 假定像习题 8 那样，假设类是直线的集合。写一个误差函数，它不仅最小化误分类数，而且也最大化边缘。

46

11. 噪声的一个来源是标号错误。你能提出一种方法，找出很可能是误标记的数据点吗？

2.11 参考文献

Angluin, D. 1988. "Queries and Concept Learning." *Machine Learning* 2:319-342.

Blumer, A., A. Ehrenfeucht, D. Haussler, and M. K. Warmuth. 1989. "Learnability and the Vapnik-Chervonenkis Dimension." *Journal of the ACM* 36:929-965.

Boyd, S., and L. Vandenberge. 2004. *Convex Optimization.* Cambridge, UK: Cambridge University Press.

Dietterich, T. G. 2003. "Machine Learning." In *Nature Encyclopedia of Cognitive Science.* London: Macmillan.

Hirsh, H. 1990. *Incremental Version Space Merging: A General Framework for Concept Learning.* Boston: Kluwer.

Kearns, M. J., and U. V. Vazirani. 1994. *An Introduction to Computational Learning Theory.* Cambridge, MA: MIT Press.

Mitchell, T. 1997. *Machine Learning.* New York: McGraw-Hill.

Valiant, L. 1984. "A Theory of the Learnable." *Communications of the ACM* 27:1134-1142.

Vapnik, V. N. 1995. *The Nature of Statistical Learning Theory.* New York: Springer.

Winston, P. H. 1975. "Learning Structural Descriptions from Examples." In *The Psychology of Computer Vision*, ed. P. H. Winston, 157-209. New York: McGraw-Hill.

47 ~ 48