

TURING

图灵程序设计丛书

[PACKT]
PUBLISHING

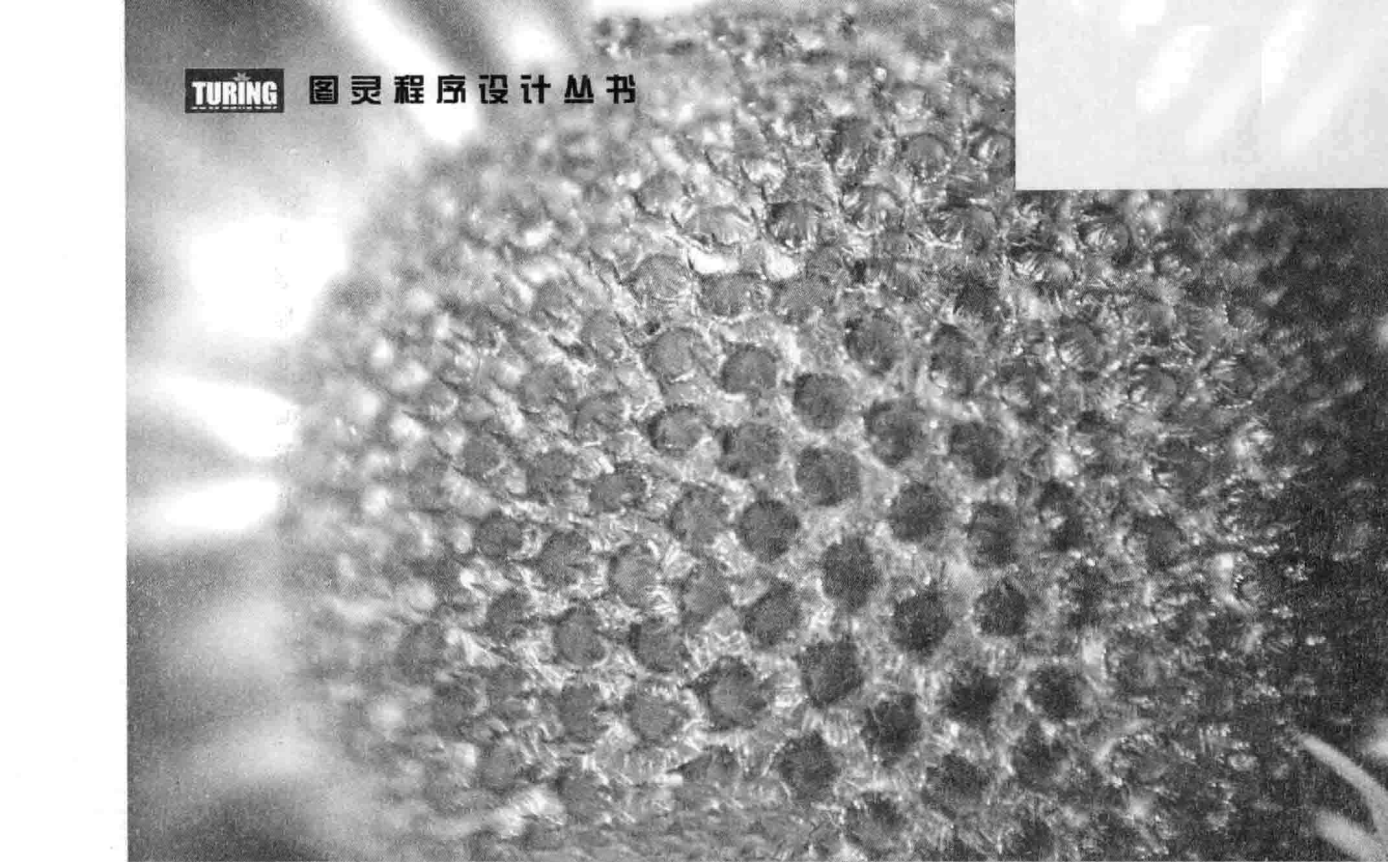
[美] Willi Richert Luis Pedro Coelho 著 刘峰 译

机器学习系统设计

Building Machine Learning
Systems with Python



人民邮电出版社
POSTS & TELECOM PRESS



TURING

图灵程序设计丛书

[美] Willi Richert Luís Pedro Coelho 著 刘峰 译

机器学习系统设计

Building Machine Learning
Systems with Python

人民邮电出版社
北 京

图书在版编目 (CIP) 数据

机器学习系统设计 / (美) 里彻特 (Richert, W.),
(美) 科埃略 (Coelho, L. P.) 著; 刘峰 译. — 北京:
人民邮电出版社, 2014. 7
(图灵程序设计丛书)
ISBN 978-7-115-35682-6

I. ①机… II. ①里… ②科… ③刘… III. ①机器学习—系统设计 IV. ①TP181

中国版本图书馆CIP数据核字(2014)第104264号

内 容 提 要

本书是实用的 Python 机器学习教程, 结合大量案例, 介绍了机器学习的各方面知识。本书不仅告诉你“怎么做”, 还会分析“为什么”, 力求帮助读者掌握多种多样的机器学习 Python 库, 学习构建基于 Python 的机器学习系统, 并亲身实践和体验机器学习系统的功能。

本书适合需要机器学习技术的 Python 开发人员、计算机科学研究人员、数据科学家、人工智能程序员, 以及统计程序员阅读参考。

-
- ◆ 著 [美] Willi Richert Luis Pedro Coelho
译 刘 峰
责任编辑 李松峰 毛倩倩
执行编辑 姜力心
责任印制 焦志伟

- ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路11号
邮编 100164 电子邮件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京鑫正大印刷有限公司印刷

- ◆ 开本: 800×1000 1/16

印张: 14

字数: 334千字

印数: 1-4 000册

2014年7月第1版

2014年7月北京第1次印刷

著作权合同登记号 图字: 01-2013-6322号

定价: 49.00元

读者服务热线: (010)51095186转600 印装质量热线: (010)81055316

反盗版热线: (010)81055315

广告经营许可证: 京崇工商广字第 0021 号



版权声明

Copyright © 2013 Packt Publishing. First published in the English language under the title *Building Machine Learning Systems with Python*.

Simplified Chinese-language edition copyright © 2014 by Posts & Telecom Press. All rights reserved.

本书中文简体字版由Packt Publishing授权人民邮电出版社独家出版。未经出版者书面许可，不得以任何方式复制或抄袭本书内容。

版权所有，侵权必究。

译者序

在眼花缭乱的互联网产品背后，你会发现总有一些东西在沙子下面闪闪发光，它们本身并不是产品，却能把令人惊艳的产品带到你我面前。机器学习技术就是这样一种宝贝。

如果在十年前，你不知道机器学习，那么可以理解，因为它还是一个科研实验室的玩具；如果在十年后的今天，作为IT从业人员的你，还没有听说过机器学习，那么你真是“奥特曼”了。

对于产品来说，机器学习技术的应用，可以给产品带来质的飞跃，提高产品的核心竞争力；对于IT从业人员来说，机器学习技术已经成为了一种必备的技能，掌握了它，可以在各大IT公司游刃有余，个人价值徒增。

《机器学习系统设计》就是一本带你机器学习海洋中遨游的书。如果你只想学习基础理论，那么这本书或许并不适合你。它并没有深入机器学习背后的数学细节，而是通过Python这样一种广泛应用的脚本语言，从数据处理，到特征工程，再到模型选择，把机器学习解决实际问题的过程一一呈现在你的面前。这本书的最大特点在于：易上手、实践性强、贴近应用。它可以让你在很短的时间内了解机器学习的基本原理，掌握机器学习工具，然后去解决实际问题。从文字、声音到图像，从主题模型、情感分析到推荐技术，本书所教给你的都是最实际的技术，让你从一个新手迅速成长为大咖。

鉴于译者水平有限，书中难免有错误疏漏之处，欢迎读者批评指正。微博：@飞旋的世界。电子邮箱：gnefuil@gmail.com。

作者致谢

感谢我的妻子Natalie和我儿子Linus及Moritz，没有家人的支持，本书将不会写就。感谢我的现任及前任经理Andras Bode、Clemens Marschner、Hongyan Zhou和Eric Crestan，感谢他们与我进行富有成效的讨论。感谢我的同事和朋友Tomasz Marchniak、Cristian Eigel、Oliver Niehoerster和Philipp Adelt，本书很多有趣的想法大都来自于他们。如果你发现本书中的错误，记得联系我，它们都归咎于我。

——Willi Richert

我要感谢我妻子Rita的爱心和支持，还要感谢我的女儿Anna，她是我生命中最美好的存在。

——Luis Pedro Coelho

关于审校者

Matthieu Brucher在法国高等电力学院读取了工程学学位（专业是信息、信号、测量），并获得了法国斯特拉斯堡大学非监督流形学习方向的博士学位。他目前在一家石油公司担任高性能计算（HPC）软件开发员，正致力于下一代油藏模拟软件的开发。

Mike Driscoll从2006年春季开始从事Python编程，经常在博客 <http://www.blog.pythonlibrary.org/>上发表关于Python的文章，偶尔也为Python软件基金会、i-Programmer和开发者论坛（Developer Zone）撰写文章。他喜爱摄影和阅读。Mike曾多次参与Packt图书的审校工作，这些图书包括：*Python 3 Object Oriented Programming*、*Python 2.6 Graphics Cookbook*和*Python Web Development Beginner's Guide*。

我要感谢我的妻子Evangeline，感谢她一直以来无尽的支持。我也要感谢我的朋友及家人，感谢他们对我的帮助。感谢耶稣对我的拯救。

Maurice HT Ling在墨尔本大学获得了分子与细胞生物学学士学位（优等），以及生物信息学博士学位。他目前在新加坡南洋理工大学担任研究员，同时还是墨尔本大学的荣誉研究员。他是*The Python Papers Anthology*的联合主编，也是新加坡Python用户组的联合创始人（自2010年起担任副主席一职）。他的研究兴趣在于生命——生物生命、人工生命以及人工智能——将计算机科学和统计学作为工具来理解生命以及它的诸多方面。个人网站：<http://maurice.vodien.com>。

前 言

如果你手里（或者你的电子阅读器里）有这本书，可以说，这是一个幸运的巧合。毕竟，每年有几百万册图书印刷出来，供数百万读者阅读，而你恰好选择了这一本。可以说，正是机器学习算法引领你来阅读这本书（或者说是把这本书引领到你面前）。而我们作为本书的作者，很高兴看到你愿意了解更多的“怎么做”和“为什么”。

本书大部分内容都将涉及“怎么做”。例如，怎么处理数据才能让机器学习算法最大限度地利用它们？怎么选择正确的算法来解决手头的问题？

我们偶尔也会涉及“为什么”。例如，为什么正确评估很重要？为什么在特定情形下一个算法比另一个算法的效果更好？

我们知道，要成为该领域的专家还有很多知识要学。毕竟，本书只介绍了一些“怎么做”和极小一部分“为什么”。但在最后，我们希望这些内容可以帮你“启航”，然后快速前行。

本书内容

第1章通过一个非常简单的例子介绍机器学习的基本概念。尽管很简单，但也可能会有过拟合的风险，这对我们提出了挑战。

第2章讲解了使用真实数据解决分类问题的方法，在这里我们对计算机进行训练，使它能够区分不同类型的花朵。

第3章讲解了词袋方法的威力，我们可以在没有真正理解帖子内容的情况下，用它来寻找相似的帖子。

第4章让我们超越将每个帖子分配给单个簇的方式。由于真实的文本可以处理多个主题，我们可以看到如何把帖子分配到几个主题上。

第5章讲解了如何用逻辑回归判定用户的答案是好还是坏。在这个情景的背后，我们将学会用偏差-方差的折中调试机器学习模型。

第6章介绍了朴素贝叶斯的工作原理，以及如何用它对推文进行分类，来判断推文中的情感

是正面的还是负面的。

第7章讨论了一个处理数据的经典课题，但它在今天仍然有意义。我们用它构建了一个推荐系统，这个系统根据用户所输入的喜欢和不喜欢的信息，为用户推荐新的商品。

第8章同时使用多种方法改进推荐效果。我们还可以看到如何只根据购物信息构建推荐系统，而不需要用户的评分数据（用户并不总会提供这一信息）。

第9章举例说明，如果有人把我们收集而成的庞大音乐库弄乱了，那么为歌曲建立次序的唯一希望就是让机器来对歌曲分类。你会发现，有时信任别人的专长比我们自己构建特征更好。

第10章讲解了如何在处理图像这个特定情景下应用分类方法。这个领域又叫做模式识别。

第11章告诉我们还有其他什么方法可以帮我们精简数据，使机器学习算法能够处理它们。

第12章讲解了不断膨胀的数据规模，以及这为何会为数据分析造成难题。在本章中，我们利用多核或计算集群，探索了一些更大规模数据的处理方法。另外，我们还介绍了云计算（将亚马逊的Web服务当做云计算提供商）。

附录A罗列了一系列机器学习的优质资源。

阅读需知

本书假定读者了解Python，并且知道如何利用easy_install或pip安装库文件。我们并不依赖于任何高等数学知识，如微积分或矩阵代数。

总体而言，本书将使用以下版本的软件，不过如果你使用任何新近版本，也没有问题。

- ☐ Python 2.7
- ☐ NumPy 1.6.2
- ☐ SciPy 0.11
- ☐ Scikit-learn 0.13

读者对象

本书适合想通过开源库来学习机器学习的Python程序员阅读参考。我们会通过示例概述机器学习的基本模式。

本书也适用于想用Python构建机器学习系统的初学者。Python是一个能够快速构建原型系统的灵活语言，它背后的算法都是由优化过的C或C++编写而成。因此，它的代码运行快捷，并且十分稳健，完全可以用在实际产品中。

排版约定

当你阅读本书时，会发现书中有各式各样的文本，它们用来区分不同类型的信息。下面是这些样式文本的示例以及相应说明。

正文中的代码是这样的：“我们可以通过使用include命令将其他内容包含进来。”

代码段采用如下形式：

```
def nn_movie(movie_likeness, reviews, uid, mid):
    likes = movie_likeness[mid].argsort()
    # 逆序排列，使最受喜爱的电影排在前面
    likes = likes[::-1]
    # 返回最相似电影的打分
    for ell in likes:
        if reviews[u,ell] > 0:
            return reviews[u,ell]
```

如果我们想让你注意代码段的特定部分，就会用粗体表示相应代码行或条目：

```
def nn_movie(movie_likeness, reviews, uid, mid):
    likes = movie_likeness[mid].argsort()
    # 逆序排列，使最受喜爱的电影排在前面
    likes = likes[::-1]
    # 返回最相似电影的打分
    for ell in likes:
        if reviews[u,ell] > 0:
            return reviews[u,ell]
```

新的术语以及重要文字采用楷体字。你在屏幕（如菜单或者对话框）中见到的文字这样出现在正文中：“点击**Next**按钮以进入下一界面。”



这里给出重要的注意事项。



提示和技巧则会在这里出现。

读者反馈

我们一贯欢迎读者的反馈意见。请告诉我们你对本书的看法，喜欢哪些部分，不喜欢哪些部分。这些反馈对于协助我们创作出真正对读者有所裨益的内容至关重要。

如果给我们反馈一般性信息，你可以发送电子邮件到feedback@packtpub.com，并在邮件标题

中注明书名。如果你是某一方面的专家并愿意参与撰稿，请访问www.packtpub.com/authors参阅我们的作者指南。

客户支持

现在你已经拥有了某本由Packt出版的书，为了让你的付出得到最大的回报，我们还为你提供其他许多方面的服务，请注意以下信息。

下载代码

如果你是通过<http://www.packtpub.com>的注册账户购买的图书，可以从该账户中下载相应Packt图书的示例代码^①。如果你是从其他地方购买的本书，可以访问<http://www.packtpub.com/support>并进行注册，我们将会为你发送一封附有示例代码文件的电子邮件。

勘误

虽然我们会全力确保本书内容的准确性，但错误仍在所难免。如果你发现了本书中的错误（包括文字和代码错误），而且愿意向我们提交这些错误，我们感激不尽。这样一来，不仅可以减少其他读者的疑虑，也有助于改进本书后续版本。要提交你发现的错误，请访问<http://www.packtpub.com/submit-errata>，选择相应图书，点击**errata submission form**（提交勘误表^②），登记你的勘误详情。勘误通过验证之后将上传到Packt网站，或添加到已有的勘误列表中。任何图书当前的勘误都可以通过<http://www.packtpub.com/support>来查看。

举报盗版

对所有媒体来说，互联网盗版都是一个棘手的问题。Packt很重视版权保护。如果你在互联网上发现我们公司出版物的任何非法复制品，请及时告知我们相关网址或网站名称，以便我们采取补救措施。

如果发现可疑盗版材料，请通过copyright@packtpub.com联系我们。

对你帮助我们保护作者权益、确保我们持续提供高品质图书的行为表示敬意。

疑难解答

如果你就本书存有疑问，请发送电子邮件到questions@packtpub.com，我们会尽力解决。

① 读者还可免费注册iTuring.cn，至本书页面下载。——编者注

② 中文版的勘误请注册iTuring.cn，至本书页面提交。——编者注

目 录

第 1 章 Python 机器学习入门	1	第 3 章 聚类：寻找相关的帖子	35
1.1 梦之队：机器学习与 Python	1	3.1 评估帖子的关联性	35
1.2 这本书将教给你什么（以及不会教什么）	2	3.1.1 不应该怎样	36
1.3 遇到困难的时候怎么办	3	3.1.2 应该怎样	36
1.4 开始	4	3.2 预处理：用相近的公共词语个数来衡量相似性	37
1.4.1 NumPy、SciPy 和 Matplotlib 简介	4	3.2.1 将原始文本转化为词袋	37
1.4.2 安装 Python	5	3.2.2 统计词语	38
1.4.3 使用 NumPy 和 SciPy 智能高效地处理数据	5	3.2.3 词语频次向量的归一化	40
1.4.4 学习 NumPy	5	3.2.4 删除不重要的词语	41
1.4.5 学习 SciPy	9	3.2.5 词干处理	42
1.5 我们第一个（极小的）机器学习应用	10	3.2.6 停用词兴奋剂	44
1.5.1 读取数据	10	3.2.7 我们的成果和目标	45
1.5.2 预处理和清洗数据	11	3.3 聚类	46
1.5.3 选择正确的模型和学习算法	12	3.3.1 K 均值	46
1.6 小结	20	3.3.2 让测试数据评估我们的想法	49
第 2 章 如何对真实样本分类	22	3.3.3 对帖子聚类	50
2.1 Iris 数据集	22	3.4 解决我们最初的难题	51
2.1.1 第一步是可视化	23	3.5 调整参数	54
2.1.2 构建第一个分类模型	24	3.6 小结	54
2.2 构建更复杂的分类器	28	第 4 章 主题模型	55
2.3 更复杂的数据集和更复杂的分类器	29	4.1 潜在狄利克雷分配（LDA）	55
2.3.1 从 Seeds 数据集中学习	29	4.2 在主题空间比较相似度	59
2.3.2 特征和特征工程	30	4.3 选择主题个数	64
2.3.3 最邻近分类	30	4.4 小结	65
2.4 二分类和多分类	33	第 5 章 分类：检测劣质答案	67
2.5 小结	34	5.1 路线图概述	67
		5.2 学习如何区分出优秀的答案	68
		5.2.1 调整样本	68

5.2.2 调整分类器	68	6.5 清洗推文	104
5.3 获取数据	68	6.6 将词语类型考虑进去	106
5.3.1 将数据消减到可处理的程度	69	6.6.1 确定词语的类型	106
5.3.2 对属性进行预选择和处理	70	6.6.2 用 SentiWordNet 成功地作弊	108
5.3.3 定义什么是优质答案	71	6.6.3 我们第一个估算器	110
5.4 创建第一个分类器	71	6.6.4 把所有东西融合在一起	111
5.4.1 从 k 邻近 (kNN) 算法开始	71	6.7 小结	112
5.4.2 特征工程	72	第 7 章 回归: 推荐	113
5.4.3 训练分类器	73	7.1 用回归预测房价	113
5.4.4 评估分类器的性能	74	7.1.1 多维回归	116
5.4.5 设计更多的特征	74	7.1.2 回归里的交叉验证	116
5.5 决定怎样提升效果	77	7.2 惩罚式回归	117
5.5.1 偏差-方差及其折中	77	7.2.1 L1 和 L2 惩罚	117
5.5.2 解决高偏差	78	7.2.2 在 Scikit-learn 中使用 Lasso 或 弹性网	118
5.5.3 解决高方差	78	7.3 P 大于 N 的情形	119
5.5.4 高偏差或低偏差	78	7.3.1 基于文本的例子	120
5.6 采用逻辑回归	81	7.3.2 巧妙地设置超参数 (hyperparameter)	121
5.6.1 一点数学和一个小例子	81	7.3.3 评分预测和推荐	122
5.6.2 在帖子分类问题上应用逻辑 回归	83	7.4 小结	126
5.7 观察正确率的背后: 准确率和召回 率	84	第 8 章 回归: 改进的推荐	127
5.8 为分类器瘦身	87	8.1 改进的推荐	127
5.9 出货	88	8.1.1 使用二值推荐矩阵	127
5.10 小结	88	8.1.2 审视电影的近邻	129
第 6 章 分类 II: 情感分析	89	8.1.3 组合多种方法	130
6.1 路线图概述	89	8.2 购物篮分析	132
6.2 获取推特 (Twitter) 数据	89	8.2.1 获取有用的预测	133
6.3 朴素贝叶斯分类器介绍	90	8.2.2 分析超市购物篮	134
6.3.1 了解贝叶斯定理	90	8.2.3 关联规则挖掘	136
6.3.2 朴素	91	8.2.4 更多购物篮分析的高级话题	137
6.3.3 使用朴素贝叶斯进行分类	92	8.3 小结	138
6.3.4 考虑未出现的词语和其他古怪 情况	94	第 9 章 分类 III: 音乐体裁分类	139
6.3.5 考虑算术下溢	95	9.1 路线图概述	139
6.4 创建第一个分类器并调优	97	9.2 获取音乐数据	139
6.4.1 先解决一个简单问题	97	9.3 观察音乐	140
6.4.2 使用所有的类	99	9.4 用 FFT 构建第一个分类器	143
6.4.3 对分类器的参数进行调优	101	9.4.1 增加实验敏捷性	143

9.4.2 训练分类器	144	11.4.2 PCA 的局限性以及 LDA 会有什么帮助	183
9.4.3 在多分类问题中用混淆矩阵 评估正确率	144	11.5 多维标度法 (MDS)	184
9.4.4 另一种方式评估分类器 效果: 受试者工作特征 曲线 (ROC)	146	11.6 小结	187
9.5 用梅尔倒频谱系数 (MFCC) 提升 分类效果	148	第 12 章 大数据	188
9.6 小结	152	12.1 了解大数据	188
第 10 章 计算机视觉: 模式识别	154	12.2 用 Jug 程序包把你的处理流程分解 成几个任务	189
10.1 图像处理简介	154	12.2.1 关于任务	189
10.2 读取和显示图像	155	12.2.2 复用部分结果	191
10.2.1 图像处理基础	156	12.2.3 幕后的工作原理	192
10.2.2 加入椒盐噪声	161	12.2.4 用 Jug 分析数据	192
10.2.3 模式识别	163	12.3 使用亚马逊 Web 服务 (AWS)	194
10.2.4 计算图像特征	163	12.3.1 构建你的第一台机器	195
10.2.5 设计你自己的特征	164	12.3.2 用 starcluster 自动创建集 群	199
10.3 在更难的数据集上分类	166	12.4 小结	202
10.4 局部特征表示	167	附录 A 更多机器学习知识	203
10.5 小结	170	A.1 在线资源	203
第 11 章 降维	171	A.2 参考书	203
11.1 路线图	171	A.2.1 问答网站	203
11.2 选择特征	172	A.2.2 博客	204
11.2.1 用筛选器检测冗余特征	172	A.2.3 数据资源	205
11.2.2 用封装器让模型选择特征	178	A.2.4 竞争日益加剧	205
11.3 其他特征选择方法	180	A.3 还剩下什么	205
11.4 特征抽取	181	A.4 小结	206
11.4.1 主成分分析 (PCA)	181	索引	207

第1章

Python机器学习入门

机器学习（ML）就是教机器自己来完成任务，就这么简单。复杂性源于细节，而这很可能就是你要读这本书的原因。

也许你现在拥有过多的数据，却对这些数据缺少理解，你希望机器学习算法可以帮助解决这个难题。于是你随机找了一些算法开始钻研，但过了一段时间就感到困惑了：在无数的算法中应该选择哪一个呢？

或许你笼统地对机器学习感兴趣，也阅读过相关的博客和文章。机器学习中的任何东西看起来都那么不可思议、那么酷，所以你开始进行探索，把一些简单的数据放入一个决策树或者一个支持向量机。但是，成功将它应用到一些其他数据之后，你又心生疑惑：所有的设置都正确吗？你得到最优的结果了吗？怎么知道有没有更好的算法？或者，你的数据是否就是“正确的”？

欢迎加入机器学习的行列！我们作为本书的作者，也曾处在这个阶段，寻找过机器学习理论教材背后的真实故事。我们发现，很多东西都是标准教材中通常不会讲到的“魔术”。所以，从某种意义上说，我们在把这本书写给年轻的自己。它不仅是机器学习的快速入门书，而且还会把我们积累的经验教训传授给你。我们希望它还可以让你更顺畅地走进计算机科学中最令人兴奋的一个领域。

1.1 梦之队：机器学习与 Python

机器学习的目标就是通过若干示例（怎样做或不做一个任务）让机器（软件）学会完成任务。假设每天早上当你打开电脑，都会做同样的事情：移动电子邮件，把属于某一特定主题的邮件放入同一个文件夹。过了一段时间，你感到厌烦了，开始琢磨是否可以让这种琐事自动完成。一种方法是分析你的大脑，将整理电子邮件时大脑思考过程中的规则记录下来。然而，这种方式相当麻烦，而且总不完美。你会漏掉一些规则，同时又会对另一些规则细致过头。另一种更好的、更加面向未来的方法是把这个过程自动化，即选择一组电子邮件元数据信息和邮件正文/文件夹名对，让算法据此选出最好的规则集。这些数据对就是你的训练数据，而生成的规则集（也叫做模型）以后能够应用到新的电子邮件上。这就是最简单的机器学习。

当然，机器学习（也常称作数据挖掘或预测分析）本身并不是一个全新的领域。正相反，这些年来来的成功可以归因于务实地采用了已经验证了的坚实技术，以及借鉴其他成功领域的真知灼见，例如统计学。统计学的目的是通过学习更多的潜在模式和关联关系，来帮助人类深入理解数据。对机器学习的成功应用了解得越多（你已经查看过kaggle.com了吧？），越会发现应用统计学是机器学习专家经常研究的一个领域。

本书后面将会介绍，构想出一个合适的机器学习（ML）方法，从来都不是一个瀑布式的过程。相反，你需要反复分析，在形形色色的机器学习算法中尝试不同版本的输入数据。这种探索方式非常适合Python。作为一门解释性高级编程语言，Python似乎就是专为尝试不同事物而设计的。更重要的是，用它进行这些尝试非常迅捷。无疑，它比C语言或其他类似的静态类型编程语言要慢一点。然而，它有着大量易用的库，而这些库往往是用C语言编写的，因此你不必为了敏捷性而牺牲速度。

1.2 这本书将教给你什么（以及不会教什么）

本书将全面展示不同应用领域正在使用的各种机器学习算法，以及使用它们时应当注意什么。然而，根据亲身经验，我们知道做这些很“酷”的事——使用和调整机器学习算法，比如支持向量机（SVM）、最邻近搜索（NNS），或者同时支持两者——其实只需要耗费一位优秀机器学习专家的一点儿时间。看看下面这个典型的工作流程，你就会发现绝大部分时间将花费在一些相当平凡的任务上：

- (1) 读取和清洗数据；
- (2) 探索和理解输入数据；
- (3) 分析如何最好地将数据呈现给学习算法；
- (4) 选择正确的模型和学习算法；
- (5) 正确地评估性能。

在探索和理解输入数据的时候，我们需要一点统计学和基础数学知识。但当这样做的时候，你会发现，这些数学课上似乎十分枯燥的知识，用来处理有趣的数据时，其实真的很令人兴奋。

解读数据标志着旅程的开始。你面对诸如无效值或缺失值的问题时，会发现这更像是一种技艺而非一门精确的科学。这是一种非常有益的技艺，因为如果这部分做得正确，那么你的数据就能够适应更多的机器学习算法，从而成功的可能性大大提高。

数据在程序的数据结构中就绪之后，你要清楚自己正在跟何方神圣打交道。你有足够的数据来回答自己的问题吗？如果没有，也许应当考虑通过额外的途径来获取一些。或许你的数据过多？那么你可能要考虑怎样最有效地从中抽取样本。

你通常不会直接将数据输入机器学习算法，而是在训练前对部分数据进行提炼。很多时候，使用机器学习算法会让你得到性能提升的回报。一个简单算法在提炼后数据上的表现，甚至能够超过一个非常复杂的算法在原始数据上的效果。这部分机器学习流程叫做特征工程 (feature engineering)，通常是一个非常令人兴奋的有意思的挑战。你有创意和智慧，便会立即看到结果。

选择正确的学习算法并不只是尝试一下工具箱中的三四个算法那么简单(工具箱中会有很多的算法)。它更需要的是深思熟虑，来权衡性能和功能的不同需求。你是否会为了快速得到结果而牺牲质量，还是愿意投入更多的时间来得到最好的结果？你是否对未来的数据有一个清晰的认知，还是应该在这方面更保守一点？

最后，性能评估是怀有远大抱负的机器学习初学者最常犯错误的地方。有一些简单的错误，比如使用了与训练相同的数据来测试你的方法。但还有一些比较难的，例如，你使用了不平衡的训练数据。再说一次，数据决定了你的任务是成功还是失败。

我们看到，只有第(4)点是关于那些花哨的算法的。虽然如此，希望这本书可以使你相信，另外4个任务并不是简单的杂务，它们同等重要，或许还更加令人兴奋。我们希望读过本书之后，你可以真正爱上数据，而非学到的算法。

最后，我们并不想让机器学习算法的理论把你压垮，因为这方面已经有很多优秀的著作了(可以在附录A中找到我们的推荐)。相反，我们会在各节中直观地介绍各种基础方法——这对于你大致理解其中的思想已经足够了，并且能够确保你走好第一步。因此，这本书并不是机器学习“权威指南”，而更像是初学者的工具。我们希望它能够激发你的好奇心，并足以让你保持渴望，不断探索这个有趣的领域。

在本章的余下部分，我们将着手介绍Python的基础库NumPy和SciPy，并且使用Scikit-learn进行第一个机器学习训练。同时我们将介绍基本的ML概念，它们稍后将贯穿于全书。本书余下的各章会详细讲述之前介绍的5个步骤，同时突出介绍使用Python的机器学习方法在各种应用场景中的不同方面。

1.3 遇到困难的时候怎么办

本书中，我们会试图讲清楚每一个必要的想法，保证你能重现各个步骤。虽然如此，你仍然可能会遇到困难。其原因可能是软件包版本的古怪组合，可能是简单的拼写错误，也可能是理解上的问题。

在这种情况下，可以通过很多不同的途径来获取帮助。很有可能，你想问的问题早已有人提出，而且下面这些优质的问答网站已经给出了答案。