

Machine Learning for OpenCV

机器学习

使用OpenCV和Python 进行智能图像处理

[美] 迈克尔·贝耶勒 (Michael Beyeler) 著 王磊 译



机械工业出版社
China Machine Press

■ ■ ■ 智能系统与技术丛书

Machine Learning for OpenCV

机器学习

使用OpenCV和Python 进行智能图像处理

[美] 迈克尔·贝耶勒 (Michael Beyeler) 著 王磊 译



机械工业出版社
China Machine Press

图书在版编目 (CIP) 数据

机器学习：使用 OpenCV 和 Python 进行智能图像处理 / (美) 迈克尔·贝耶勒 (Michael Beyeler) 著；王磊译．—北京：机械工业出版社，2018.11

(智能系统与技术丛书)

书名原文：Machine Learning for OpenCV

ISBN 978-7-111-61151-6

I. 机… II. ①迈… ②王… III. ①机器学习 ②图像处理软件—程序设计 IV. ① TP181
② TP391.413

中国版本图书馆 CIP 数据核字 (2018) 第 231315 号

本书版权登记号：图字 01-2017-7519

Michael Beyeler: Machine Learning for OpenCV (ISBN: 9781783980284).

Copyright © 2017 Packt Publishing. First published in the English language under the title "Machine Learning for OpenCV".

All rights reserved.

Chinese simplified language edition published by China Machine Press.

Copyright © 2019 by China Machine Press.

本书中文简体字版由 Packt Publishing 授权机械工业出版社独家出版。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。

机器学习：使用 OpenCV 和 Python 进行智能图像处理

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：刘 锋

责任校对：殷 虹

印 刷：北京诚信伟业印刷有限公司

版 次：2019 年 1 月第 1 版第 1 次印刷

开 本：186mm×240mm 1/16

印 张：16.5

书 号：ISBN 978-7-111-61151-6

定 价：69.00 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) 88379426 88361066

投稿热线：(010) 88379604

购书热线：(010) 68326294 88379649 68995259

读者信箱：hzit@hzbook.com

版权所有·侵权必究

封底无防伪标均为盗版

本书法律顾问：北京大成律师事务所 韩光 / 邹晓东

华章 IT
HZBOOKS | Information Technology



THE TRANSLATOR'S WORDS

译者序

翻译一直是我的业余爱好之一，而机器学习和图像处理也是我工作的主要内容，因此非常荣幸可以成为本书的译者。这几年随着深度学习的火热，越来越多的人意识到了在非常多的应用场景下，机器学习具有非常高的应用价值。从 AlphaGo 大杀四方，到所谓的“刷脸”支付，再到最近谷歌使用机器学习来替代 B 树索引，可以看到，我们现在正处在被机器学习全面改造的社会变革中！本书介绍了机器学习的基本概念和算法，使用 Python 这一 AI 领域最常使用的编程语言，结合 OpenCV 和 scikit-learn 库，向我们揭开了机器学习的神秘面纱。OpenCV 和 scikit-learn 都是非常著名的开源库，尤其是 scikit-learn 的文档，非常完善而且专业，是学习机器学习相关知识的绝佳资料。本书内容更偏实战，力图通过具体的例子讲解机器学习算法的基本知识和使用方法。因本人学识有限，书中难免存在谬误，还望读者指正！

序

在过去的几年中，我们的机器已经逐渐学会了如何观察。现在我们认为相机能检测到照片中的人脸是理所当然的事，那些社交软件甚至可以从我们上传的照片中识别出我们和我们的朋友。而在未来几年中，我们将会体验到更为剧烈的变化。不久以后，车辆可以自动驾驶，手机可以为我们阅读信息并翻译为任意语言，我们的 X 光片和其他的医疗图像可以被强大的算法读取并分析，从而做出准确的医疗诊断，甚至可以推荐高效的治疗方案。

这些变化的驱动力来自于不断提高的计算力、大量的图像数据以及大量来自于数学、统计学和计算机科学的聪明想法。这些快速发展的变化都汇聚在已经开始腾飞的机器学习技术中，不断影响着我们与世界、与他人的日常交流。在计算机视觉中，当前机器学习模式转变的一个明显特征，在于开源社区中有大量的志愿者、爱好者、科学家和工程师，他们免费开发和提供了大量的软件工具。这意味着，从根本上说，进入这个领域的门槛是有史以来最低的：任何感兴趣的人都可以在图像处理中使用机器学习。

然而，就如同置身于一个充满岔路的公园一样，面对丰富的工具和想法以及快速开发这些想法的方法，都需要一个向导为你引路，指引正确的方向。对此，我有一个好消息：通过浏览本书，你可以把我的同事和合作者 Michael Beyeler 博士当作你的向导。Michael 知识渊博，既是一个冷静的工程师、计算机科学家和神经系统科学家，也是多产的开源软件开发开发者。他不仅教会机器如何在复杂环境中观察和导航，让计算机模拟人脑活动，还经常指导人们如何使用编程来解决各种不同的机器学习和图像处理的难题。这意味着，你不仅可以从他丰富的专业知识和经验中获益，也会享受他在教授本书时的睿智和幽默。

另一个好消息是，这将会是一段令人兴奋的旅程。基于代码和数据像完成拼图一样，解决计算机视觉和机器学习中的问题，这种喜悦无以复加。如 Richard Feynman 所说的：“我所不能创造的东西，就是我没有理解的东西。”那么，亲自实践本书的代码示例（开源！），使自己变得更有创造性，理解也就随之而来了。

Ariel Rokem

数据科学家，华盛顿大学 eScience 研究所

前 言

你能看到这里，我非常高兴，是时候来谈谈机器学习了。

机器学习已经不再是一个时髦的词汇，它在我们周围随处可见：保护你的电子邮件、自动在图片上标记朋友、预测你喜欢的电影。作为数据科学的一个子领域，机器学习让电脑可以通过经验来学习：通过收集历史数据来对未来进行预测。

而要被分析的数据是无穷无尽的！目前每天产生的数据量达到了 2.5 艾字节（约为 10 亿 GB）。你能相信吗？这些数据足够塞满 1000 万张蓝光光盘，或者相当于能够持续播放 90 年的高清视频。为了处理如此庞大的数据，诸如谷歌、亚马逊、微软和脸书这样的公司，投入了大量的人力物力到让我们可以随时随地从机器学习中获益——从你的手机应用扩展到连接着云端的超级计算机的数据平台开发上。

换句话说：现在是时候对机器学习进行投资了。如果你也希望成为机器学习从业人员，那么本书非常适合你！

不过先别急：你的应用并不需要像上面的例子一样规模巨大或者有影响力，才能从机器学习中获益。不积跬步，无以至千里。因此，本书第一步会通过简单直接的例子，向你介绍统计学习的基础概念，比如分类和回归。如果你已经详细学习了机器学习，本书将教会你如何学以致用。对了，如果你对这个领域一无所知也没关系——只要你好学即可。

当本书介绍完了所有的基础概念后，将会开始探索各种算法，比如决策树、支持向量和贝叶斯网络，以及如何把它们与其他 OpenCV 功能结合。在这个过程中，你将会学习如何通过理解数据来理解任务，以及如何构建具有完整功能的机器学习管道。

你的机器学习技能将会随着本书的深入而提高，直到你准备接触这个领域最热的话题：深度学习。结合如何针对任务选择正确工具的训练技巧，我们将确保你可以掌握所有的机器学习基础知识。

在本书的最后部分，你将会准备好面对你自己的机器学习问题，要么基于现有的源代码构建，要么从零开始构建你自己的算法！

本书所涉及的内容

第 1 章将会简要介绍机器学习不同的子领域，并讲解如何安装 OpenCV 和 Python

Anaconda 版本下的其他必要工具。

第 2 章将展示经典的机器学习处理流程，以及载入处理数据的时机。将会解释训练数据和测试数据之间的区别，以及如何使用 OpenCV 和 Python 载入、存储、处理数据并进行可视化。

第 3 章将会通过回顾一些核心概念来介绍监督学习的内容，比如分类和回归。你将会学到如何使用 OpenCV 实现一个简单的机器学习算法，如何对数据进行预测，以及如何评估模型。

第 4 章将会教你如何切身体会一些常见的、著名的机器学习数据集，以及如何从原始数据中提取感兴趣的内容。

第 5 章将会展示如何使用 OpenCV 构建决策树，以及如何在不同的分类问题和回归问题上使用它。

第 6 章将会解释如何使用 OpenCV 构建支持向量机，以及如何把它们应用于检测图像中的行人。

第 7 章将会介绍概率论，并展示如何使用贝叶斯推断来判断邮件是否为垃圾邮件。

第 8 章将会讨论一些非监督学习算法，比如 k 均值聚类算法和期望最大化算法，并展示如何使用它们从简单、无标签的数据集中提取隐藏的结构信息。

第 9 章将会把你带到令人激动的深度学习领域。从感知器和多层感知器开始，你将会学习如何构建深度神经网络来对 MNIST 数据集中的手写数字进行识别。

第 10 章将展示如何高效地集成多个算法，以克服单个学习器的弱点，让预测结果更加准确可靠。

第 11 章将会介绍模型选择的概念，帮助你针对手上的任务，从不同的机器学习算法中，选择合适的工具。

第 12 章将会总结本书，并对如何进一步学习机器学习，以及从哪里可以找到更高级话题信息，给出了一些有用的建议。

阅读本书你需要什么

你将需要一台电脑、Python Anaconda 以及学习的激情，非常多的激情。你可能会问：为什么使用 Python？答案很简单：由于它有大量的开源库和可以用于数据处理的交互的工具，这让它最终成为数据科学的专属语言。

其中一个工具是 Anaconda Python 发行版，它可以提供所需的所有科学计算库，比如 NumPy、SciPy、Matplotlib、scikit-Learn 和 Pandas。此外，安装 OpenCV 仅仅只需一行代码。并不需要设置 cc 编译的选项开关或者从源码进行编译！我们将会在第 1 章中介绍如何安装 Python Anaconda。

如果你经常结合 C++ 使用 OpenCV，那也没有问题。但是，至少在本书的目标中，还

是强烈建议你切换到 Python。C++ 主要应用于开发高性能代码或者构建实时应用。但在学习一项新的技能时，我相信 Python 从根本上说会是更好的语言选择，因为编写的代码少，你就有时间做更多的事情。相对于纠结 C++ 的语法细节，或者花费大量时间把数据从一种格式转换为另一种格式，Python 可以让你更加集中精力在手头的任务上：在机器学习领域变成专家。

样例代码下载

可以从 GitHub 上下载本书最新版本的示例代码文件：<http://github.com/mbeyeler/opencv-machine-learning>。所有的代码以 MIT 软件协议发布，因此可以按自己的需要自由地使用、修改和分享。同时也可以在不同的 Jupyter 笔记本中搜索里面的源代码。

如果对源代码有任何疑问，欢迎在我们的论坛上提出来：<https://groups.google.com/d/forum/machine-learning-for-opencv>。很有可能的是，已经有人分享了这个问题的解决方案。

读者也可以访问华章图书官网 www.hzbook.com，通过注册并登录个人账号，下载本书的源代码和彩图。

审校者简介

Vipul Sharma 是印度班加罗尔一家创业公司的软件工程师。他在贾巴尔普尔工程大学的信息科技学院（2016年）学习工程学。他是 Python 的粉丝，在业余时间乐于构建计算机视觉的项目。他也是开源爱好者，为感兴趣的项目贡献力量。他乐于学习，争取让自己成为一名更优秀的开发者。他在 <http://vipul.xyz> 上写关于自己项目的博客，也会将代码发布到 <http://github.com/vipul-sharma20>。

Rahul Kavi 是硅谷的一名研究科学家。他拥有西弗吉尼亚大学计算机科学专业的硕士和博士学位。Rahul 的工作是为大量的平台和应用程序研究和优化计算机视觉应用。他也为 OpenCV 贡献过机器学习模块。在 NASA 2015 年和 2016 年举行的百年纪念挑战赛中，由他编写计算机视觉和机器学习软件的机器人获得了机器人组的第一名。Rahul 的研究曾发表在多个会议论文和期刊上。

目 录

译者序	
序	
前言	
审校者简介	
第 1 章 品味机器学习	1
1.1 初步了解机器学习	1
1.2 机器学习可以解决的事情	3
1.3 初步了解 Python	4
1.4 初步了解 OpenCV	4
1.5 安装	5
1.5.1 获取本书最新的代码	5
1.5.2 掌握 Python Anaconda	6
1.5.3 在 conda 环境中安装 OpenCV	8
1.5.4 验证安装结果	9
1.5.5 一睹 OpenCV ML 模块	11
1.6 总结	11
第 2 章 使用 OpenCV 和 Python 处理数据	12
2.1 理解机器学习流程	12
2.2 使用 OpenCV 和 Python 处理数据	14
2.2.1 创建一个新的 IPython 或 Jupyter 会话	15
2.2.2 使用 Python 的 NumPy 包处理数据	16
2.2.3 在 Python 中载入外部 数据集	20
2.2.4 使用 Matplotlib 进行 数据可视化	21
2.2.5 使用 C++ 中 OpenCV 的 TrainData 容器处理数据	26
2.3 总结	27
第 3 章 监督学习的第一步	28
3.1 理解监督学习	28
3.1.1 了解 OpenCV 中的 监督学习	29
3.1.2 使用评分函数评估 模型性能	30
3.2 使用分类模型预测类别	35
3.2.1 理解 k-NN 算法	37
3.2.2 使用 OpenCV 实现 k-NN	37
3.3 使用回归模型预测连续 结果	43
3.3.1 理解线性回归	43
3.3.2 使用线性回归预测 波士顿房价	44
3.3.3 应用 Lasso 回归和 ridge 回归	48

3.4	使用逻辑回归对鸢尾花种类进行分类	48	5.1.6	控制决策树的复杂度	90
3.5	总结	53	5.2	使用决策树进行乳腺癌的诊断	90
第 4 章	数据表示与特征工程	54	5.2.1	载入数据集	91
4.1	理解特征工程	54	5.2.2	构建决策树	92
4.2	数据预处理	55	5.3	使用决策树进行回归	96
4.2.1	特征标准化	56	5.4	总结	99
4.2.2	特征归一化	57	第 6 章	使用支持向量机检测行人 ...	100
4.2.3	特征缩放到一定的范围	57	6.1	理解线性支持向量机	100
4.2.4	特征二值化	58	6.1.1	学习最优决策边界	101
4.2.5	缺失数据处理	58	6.1.2	实现我们的第一个支持向量机	102
4.3	理解降维	59	6.2	处理非线性决策边界	107
4.3.1	在 OpenCV 中实现主成分分析	61	6.2.1	理解核机制	108
4.3.2	实现独立成分分析	64	6.2.2	认识我们的核	109
4.3.3	实现非负矩阵分解	65	6.2.3	实现非线性支持向量机 ...	109
4.4	类别变量表示	66	6.3	自然环境下的行人检测	110
4.5	文本特征表示	68	6.3.1	获取数据集	111
4.6	图像表示	69	6.3.2	初窥方向梯度直方图	113
4.6.1	使用色彩空间	69	6.3.3	生成负样本	114
4.6.2	图像角点检测	71	6.3.4	实现支持向量机	116
4.6.3	使用尺度不变特征变换	72	6.3.5	模型自举	116
4.6.4	使用加速健壮特征	74	6.3.6	在更大的图像中检测行人	118
4.7	总结	75	6.3.7	进一步优化模型	120
第 5 章	使用决策树进行医疗诊断	76	6.4	总结	121
5.1	理解决策树	76	第 7 章	使用贝叶斯学习实现垃圾邮件过滤	122
5.1.1	构建第一个决策树	79	7.1	理解贝叶斯推断	122
5.1.2	可视化训练得到的决策树	85	7.1.1	概率论的短暂之旅	123
5.1.3	深入了解决策树的内部工作机制	87	7.1.2	理解贝叶斯定理	124
5.1.4	特征重要性评分	88	7.1.3	理解朴素贝叶斯分类器 ...	126
5.1.5	理解决策规则	89	7.2	实现第一个贝叶斯分类器	127
			7.2.1	创建一个练习数据集	127

7.2.2	使用一个正态贝叶斯分类器对数据分类	128	8.6.2	实现凝聚层次聚类	162
7.2.3	使用一个朴素贝叶斯分类器对数据分类	131	8.7	总结	163
7.2.4	条件概率的可视化	132	第 9 章 使用深度学习对手写数字分类		164
7.3	使用朴素贝叶斯分类器对邮件分类	134	9.1	理解 McCulloch-Pitts 神经元	164
7.3.1	载入数据集	134	9.2	理解感知器	167
7.3.2	使用 Pandas 构建数据矩阵	136	9.3	实现第一个感知器	169
7.3.3	数据预处理	137	9.3.1	生成练习数据集	170
7.3.4	训练正态贝叶斯分类器	138	9.3.2	使用数据拟合感知器	171
7.3.5	使用完整的数据集进行训练	139	9.3.3	评估感知器分类器	171
7.3.6	使用 n-gram 提升结果	139	9.3.4	把感知器应用到线性不可分的数据上	173
7.3.7	使用 TD-IDF 提升结果	140	9.4	理解多层感知器	174
7.4	总结	141	9.4.1	理解梯度下降	175
第 8 章 使用非监督学习发现隐藏结构		142	9.4.2	使用反向传播训练多层感知器	178
8.1	理解非监督学习	142	9.4.3	在 OpenCV 中实现多层感知器	179
8.2	理解 k 均值聚类	143	9.5	了解深度学习	183
8.3	理解期望最大化	145	9.6	手写数字分类	186
8.3.1	实现期望最大化解决方案	146	9.6.1	载入 MNIST 数据集	187
8.3.2	了解期望最大化的局限	148	9.6.2	MNIST 数据集预处理	188
8.4	使用 k 均值压缩色彩空间	154	9.6.3	使用 OpenCV 训练一个 MLP	189
8.4.1	真彩色调色板的可视化	154	9.6.4	使用 Keras 训练一个深度神经网络	190
8.4.2	使用 k 均值减少调色板	157	9.7	总结	192
8.5	使用 k 均值对手写数字分类	159	第 10 章 组合不同算法为一个整体		193
8.5.1	载入数据集	159	10.1	理解集成方法	193
8.5.2	运行 k 均值	159	10.1.1	理解平均集成	195
8.6	把聚类组织成层次树	161	10.1.2	理解提升集成	197
8.6.1	理解层次聚类	161	10.1.3	理解堆叠集成	200
			10.2	组合决策树为随机森林	200

10.2.1	理解决策树的不足	200
10.2.2	实现第一个随机森林	204
10.2.3	使用 scikit-learn 实现 一个随机森林	205
10.2.4	实现极端随机树	206
10.3	使用随机森林进行人脸识别	208
10.3.1	载入数据集	208
10.3.2	预处理数据集	209
10.3.3	训练和测试随机森林	210
10.4	实现 AdaBoost	212
10.4.1	使用 OpenCV 实现 AdaBoost	212
10.4.2	使用 scikit-learn 实现 AdaBoost	213
10.5	组合不同模型为一个投票 分类器	214
10.5.1	理解不同的投票机制	214
10.5.2	实现一个投票分类器	215
10.6	总结	217
第 11 章 通过超参数调优选择 合适的模型		
11.1	评估一个模型	218
11.1.1	评估模型错误的方法	219
11.1.2	评估模型正确的方法	220
11.1.3	选择最好的模型	221
11.2	理解交叉验证	223
11.2.1	使用 OpenCV 手动 实现交叉验证	225
11.2.2	使用 scikit-learn 进行 k 折交叉验证	226
11.2.3	实现留一法交叉验证	227
11.3	使用自举评估鲁棒性	228
11.4	评估结果的重要性	230
11.4.1	实现 T 检验	230
11.4.2	实现配对卡方检验	232
11.5	使用网格搜索进行超 参数调优	233
11.5.1	实现一个简单的 网格搜索	234
11.5.2	理解验证集的价值	235
11.5.3	网格搜索结合交叉 验证	236
11.5.4	网格搜索结合嵌套交叉 验证	238
11.6	使用不同评估指标来对 模型评分	239
11.6.1	选择正确的分类指标	239
11.6.2	选择正确的回归指标	240
11.7	链接算法形成一个管道	240
11.7.1	用 scikit-learn 实现 管道	241
11.7.2	在网格搜索中使用 管道	242
11.8	总结	243
第 12 章 综合		
12.1	着手处理一个机器学习 问题	244
12.2	构建自己的估计器	245
12.2.1	使用 C++ 编写自己 的基于 OpenCV 的 分类器	245
12.2.2	使用 Python 编写自己 的基于 scikit-learn 的 分类器	247
12.3	今后的方向	249
12.4	总结	251

品味机器学习

所以，你已经决定进入机器学习领域。太棒了！

现在，机器学习随处可见——无论是保护我们的电子邮件，还是自动在图片上标记我们的朋友，还有预测我们喜欢的电影是什么。作为人工智能的形式之一，机器学习让电脑可以通过经验进行学习：通过收集过去的数据对未来进行预测。除此之外，计算机视觉是机器学习最令人激动的应用领域之一，深度学习和卷积神经网络不断驱使着诸如无人驾驶汽车和谷歌 DeepMind 这样的创新系统的发展。

然而，不用担心。要从机器学习中受益，你的应用并不需要像前面的例子那样规模巨大或者可以改变世界。在本章中，将讨论机器学习为何会变得如此流行，并讨论它可以解决的问题。接下来，为了使用 OpenCV 解决机器学习问题，将会介绍我们需要的工具。在本书的范畴内，假定你已经拥有 OpenCV 和 Python 的基础知识，但总会有需要学习的地方。

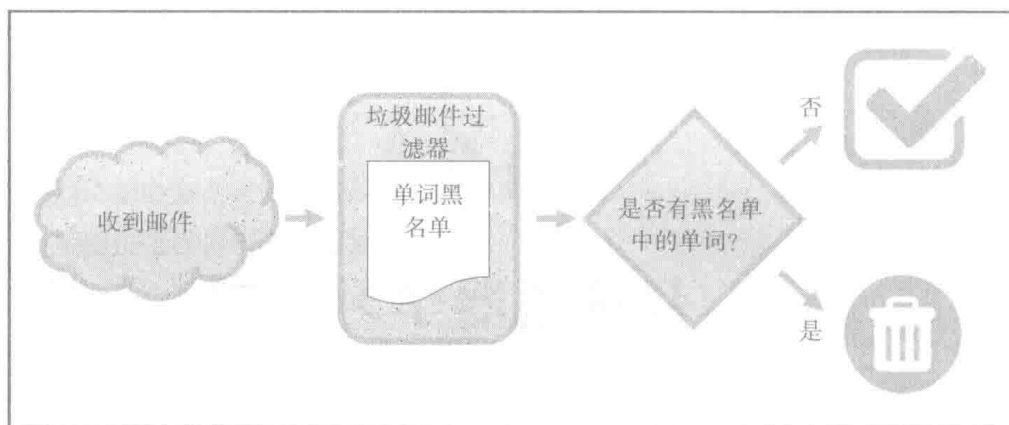
那么你准备好了吗？让我们开始学习吧！

1.1 初步了解机器学习

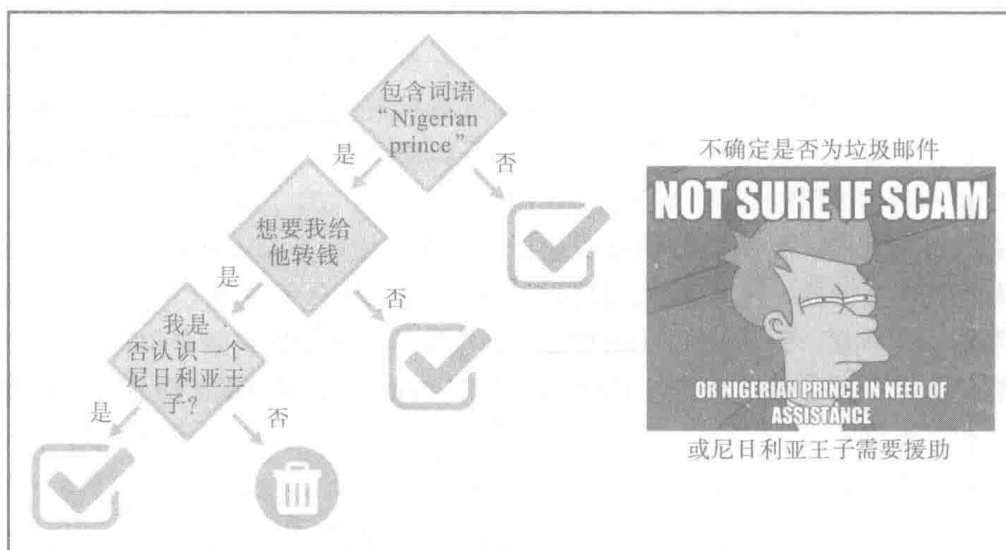
机器学习至少存在六十多年了。早期的机器学习系统诞生于人工智能的探索之路，使用 `if...else` 这样的人工编码规则语句来处理数据并做出决策。例如一个垃圾邮件过滤器，解析收到的电子邮件，把不想要的邮件移动到垃圾文件夹中。

我们可以想到的是，创建一个单词黑名单，每当邮件中出现里面的单词时，就把这封邮件标记为垃圾邮件。这就是一个简单的人工编码的专家系统的例子。（我们将会在第 7 章构建一个更聪明的过滤器。）

如果允许把这些专家决策规则联合嵌套到决策树中（第 5 章），我们可以让它们变成任意复杂度的规则。那么，它将使我们可以做出包含一系列决策步骤的、更加精确的决策成为可能，正如下图所示：



垃圾邮件过滤器



在简单的垃圾邮件过滤中的决策步骤

人工编码这些决策规则是切实可行的，但有以下两个主要的缺点：

- ❑ 做决策所需应用的逻辑仅针对单一领域的特定任务。比如，我们没法使用这个垃圾邮件过滤器去标记照片上的朋友。即使我们想要使用这个垃圾邮件分类器去做一些稍有不同事情，比如从正常邮件中过滤出钓鱼欺诈的邮件，我们将需要重新设计整个决策规则。
- ❑ 手动设计规则需要对问题有深刻的理解。我们需要明确知道哪种类型的邮件构成垃圾邮件，包括所有可能的例外情况。这可不像是看起来那么简单。否则，我们就需要经常检查垃圾邮件文件夹，看是否有被偶然过滤掉的重要信息。对于其他领域的问题，手动设计规则可能行不通。

这些问题就需要机器学习出马了。有时定义的任务可能并不明确——比如少了一些可

能情况——而我们希望机器可以感知到这种情况并自动解决任务。其他时候，有可能大量的数据中隐含了重要的关系和相关性，而我们人类可能难以发现（参见第8章）。在这些情况中，机器学习常常被用来提取这些隐藏的关系（也被称作数据挖掘）。

人工专家系统在图像中检测人脸失败就是一个很好的例子。很傻，不是吗？现在，每个智能手机都可以检测照片中的人脸。然而，20年前，这个问题还无法解决。造成这种情况的原因是人们认为人脸的组成方式对机器来说并没有用。作为人类，我们不会以像素的粒度去思考。如果让我们来检测人脸，我们可能仅会去寻找人脸的特定特征，比如眼睛、鼻子、嘴等。但当所有的机器只知道图像是由像素组成，像素拥有某种灰度时，我们要怎么告诉机器去寻找什么呢？在相当长的一段时间，图像表示的这种差异，基本上导致人们无法想出一套很好的决策规则来让机器从图像中检测人脸。我们将会在第4章中讨论针对这个问题的不同方法。

然而，随着卷积神经网络和深度学习（第9章）的出现，机器在识别人脸这个问题上已经变得和我们一样出色。我们所要做的全部，仅仅是为机器准备大量的人脸图像。接着，机器就可以从中发现一组特征用于识别人脸，而无须像我们以前做的那样来解决这个问题。这是机器学习真正的能力。

1.2 机器学习可以解决的事情

大多数的机器学习问题属于以下三个类别之一：

- 监督学习：每个数据点都被标记或关联为一个类别或者分值（第3章）。类别标签的例子是给一张图片分配一个猫或者狗的字段。分值标签的例子是为一辆二手车设置一个出售价格。监督学习的目标就是学习大量类似这样的样本（叫作训练数据），从而对未来的数据点做出预测（叫作测试数据）。预测结果分为两个情况，比如从新图片中识别正确的动物（叫作分类问题），或者为其他二手车分配一个准确的出售价格（叫作回归问题）。不用担心现在无法理解——我们整本书会详细阐述这些内容的细节。
- 非监督学习：数据点没有相关的标记（见第8章）。与此相反，非监督学习算法的目标是以某种方式来组织数据或者描述数据的结构。这就意味着把这些数据进行聚类或者发现其他不同的方式来观察数据，从而让这些数据看起来更简单些。
- 强化学习：算法会根据各个数据点选择动作进行响应。这是机器人科学中比较常见的算法，一组感知器在某个时间点读取的数据就是一个数据点，这时算法需要为机器人的下一个动作做出选择。这也是物联网应用的本质需求，学习算法在未来的某一小段时间内接收到奖励信号，这个信号可以表明这个决策的好坏。基于此，算法会不断地修改策略以得到最高的奖励。

下图演示了这三种主要的类别：