

Python机器学习 (原书第2版)



[Python机器学习 \(原书第2版\) 下载链接1](#)

著者:[美] 塞巴斯蒂安·拉施卡 (Sebastian Raschka)

出版者:机械工业出版社

出版时间:2018-11

装帧:

isbn:9787111611509

本书自第1版出版以来，备受广大读者欢迎。与同类书相比，本书除了介绍如何用Python和基于Python的机器学习软件库进行实践外，还对机器学习概念的必要细节进行讨论，同时对机器学习算法的工作原理、使用方法以及如何避免掉入常见的陷阱提供直观且翔实的解释，是Python机器学习入门必读之作。

本书将带领你进入预测分析的世界，并展示为什么Python会成为数据科学领域首屈一指的计算机语言。如果你想更好地从数据中得到问题的答案，或者想要提升并扩展现有机器学习系统的性能，那么这本基于数据科学实践的书籍非常值得一读。它的内容涵盖了众多高效Python库，包括scikit-learn、Keras和TensorFlow等，系统性地梳理和分析了各种经典算法，并通过Python语言以具体代码示例的方式深入浅出地介绍了各种算法的应用，还给出了从情感分析到神经网络的一些实践技巧，这些内容能使你快速解决你和你的团队面临的一些重要问题。

不管你是学习数据科学的初学者，还是想进一步拓展对数据科学领域的认知，本书都是一个重要且不可错过的资源，它能帮助你了解如何使用Python解决数据中的关键问题。

本书自第1版出版以来，备受广大读者欢迎。与同类书相比，本书除了介绍如何用Python和基于Python的机器学习软件库进行实践外，还对机器学习概念的必要细节进行讨论，同时对机器学习算法的工作原理、使用方法以及如何避免掉入常见的陷阱提供直观且翔实的解释，是Python机器学习入门必读之作。

本书将带领你进入预测分析的世界，并展示为什么Python会成为数据科学领域首屈一指的计算机语言。如果你想更好地从数据中得到问题的答案，或者想要提升并扩展现有机器学习系统的性能，那么这本基于数据科学实践的书籍非常值得一读。它的内容涵盖了众多高效Python库，包括scikit-learn、Keras和TensorFlow等，系统性地梳理和分析了各种经典算法，并通过Python语言以具体代码示例的方式深入浅出地介绍了各种算法的应用，还给出了从情感分析到神经网络的一些实践技巧，这些内容能使你快速解决你和你的团队面临的一些重要问题。

不管你是学习数据科学的初学者，还是想进一步拓展对数据科学领域的认知，本书都是一个重要且不可错过的资源，它能帮助你了解如何使用Python解决数据中的关键问题。

本书将机器学习背后的基本理论与应用实践联系起来，通过这种方式让你聚焦于如何正确地提出问题、解决问题。书中讲解了如何使用Python的核心元素以及强大的机器学习库，同时还展示了如何正确使用一系列统计模型。

在本书第1版的基础上，作者对第2版进行了大量更新和扩展，纳入最近的开源技术，包括scikit-learn、Keras和TensorFlow，提供了使用Python构建高效的机器学习与深度学习应用的必要知识与技术。

通过阅读本书，你将学到：

探索并理解数据科学、机器学习与深度学习的主要框架

通过机器学习模型与神经网络对数据提出新的疑问

在机器学习中使用新的Python开源库的强大功能

掌握如何使用TensorFlow库来实现深度神经网络

在可访问的Web应用中嵌入机器学习模型

使用回归分析预测连续目标的结果

使用聚类发现数据中的隐藏模式与结构

使用深度学习技术分析图片

使用情感分析深入研究文本与社交媒体数据

作者简介:

塞巴斯蒂安·拉施卡 (Sebastian Raschka)

密歇根州立大学博士，他在计算生物学领域提出了几种新的计算方法，还被科技博客Analytics

Vidhya评为GitHub上最具影响力的数据科学家。他在Python编程方面积累了丰富经验，曾为如何实际应用数据科学、机器学习和深度学习做过数次讲座，包括在SciPy（重要的Python科学计算会议）上做的机器学习教程。正是因为Sebastian在数据科学、机器学习以及Python等领域拥有丰富的演讲和写作经验，他才有动力完成本书的撰写，

目录: 译者序

关于作者

关于审校人员

前言

第1章 赋予计算机从数据中学习的能力 1

1.1 构建把数据转换为知识的智能机器 1

1.2 三种不同类型的机器学习 1

1.2.1 用有监督学习预测未来 2

1.2.2 用强化学习解决交互问题 3

1.2.3 用无监督学习发现隐藏结构 4

1.3 基本术语与符号 4

1.4 构建机器学习系统的路线图 6

1.4.1 预处理—整理数据 6

1.4.2 训练和选择预测模型 7

1.4.3 评估模型和预测新样本数据 7

1.5 用Python进行机器学习 7

1.5.1 从Python包索引安装Python和其他包 8

1.5.2 采用Anaconda Python和软件包管理器 8

1.5.3 科学计算、数据科学和机器学习软件包 8

1.6 小结 9

第2章 训练简单的机器学习分类算法 10

2.1 人工神经元—机器学习早期历史一瞥 10

2.1.1 人工神经元的正式定义 11

2.1.2 感知器学习规则 12

2.2 在Python中实现感知器学习算法 14

2.2.1 面向对象的感知器API 14

2.2.2 在鸢尾花数据集上训练感知器模型 16

2.3 自适应神经元和学习收敛 20

2.3.1 梯度下降为最小代价函数 21

2.3.2 用Python实现Adaline 22

2.3.3 通过调整特征大小改善梯度下降 25

2.3.4 大规模机器学习与随机梯度下降 27

2.4 小结 30

第3章 scikit-learn机器学习分类器一览	32
3.1 选择分类算法	32
3.2 了解scikit-learn软件库的第一步—训练感知器	32
3.3 基于逻辑回归的分类概率建模	37
3.3.1 逻辑回归的直觉与条件概率	37
3.3.2 学习逻辑代价函数的权重	39
3.3.3 把转换的Adaline用于逻辑回归算法	41
3.3.4 用scikit-learn训练逻辑回归模型	44
3.3.5 通过正则化解决过拟合问题	45
3.4 支持向量机的最大余量分类	47
3.4.1 最大边际的直觉	48
3.4.2 用松弛变量处理非线性可分	48
3.4.3 其他的scikit-learn 实现	50
3.5 用核支持向量机求解非线性问题	50
3.5.1 处理线性不可分数据的核方法	50
3.5.2 利用核技巧，发现高维空间的分离超平面	52
3.6 决策树学习	55
3.6.1 最大限度地获取信息—获得最大收益	55
3.6.2 构建决策树	58
3.6.3 通过随机森林组合多个决策树	61
3.7 K-近邻—一种懒惰的学习算法	63
3.8 小结	65
第4章 构建良好的训练集—预处理	66
4.1 处理缺失数据	66
4.1.1 识别数据中的缺失数值	66
4.1.2 删除缺失的数据	67
4.1.3 填补缺失的数据	68
4.1.4 了解scikit-learn评估器API	68
4.2 处理分类数据	69
4.2.1 名词特征和序数特征	69
4.2.2 映射序数特征	70
4.2.3 分类标签编码	70
4.2.4 为名词特征做热编码	71
4.3 分裂数据集为独立的训练集和测试集	73
4.4 把特征保持在同一尺度上	75
4.5 选择有意义的特征	76
4.5.1 L1和L2正则化对模型复杂度的惩罚	76
4.5.2 L2正则化的几何解释	77
4.5.3 L1正则化的稀疏解决方案	78
4.5.4 为序数特征选择算法	80
4.6 用随机森林评估特征的重要性	84
4.7 小结	87
第5章 通过降维压缩数据	88
5.1 用主成分分析实现无监督降维	88
5.1.1 主成分分析的主要步骤	88
5.1.2 逐步提取主成分	89
5.1.3 总方差和解释方差	91
5.1.4 特征变换	92
5.1.5 scikit-learn的主成分分析	93
5.2 基于线性判别分析的有监督数据压缩	96
5.2.1 主成分分析与线性判别分析	96
5.2.2 线性判别分析的内部逻辑	97
5.2.3 计算散布矩阵	97
5.2.4 在新的特征子空间选择线性判别式	99

- 5.2.5 将样本投影到新的特征空间 101
- 5.2.6 用scikit-learn实现的LDA 101
- 5.3 非线性映射的核主成分分析 102
 - 5.3.1 核函数与核技巧 103
 - 5.3.2 用Python实现核主成分分析 106
 - 5.3.3 投影新的数据点 111
 - 5.3.4 scikit-learn的核主成分分析 113
- 5.4 小结 114
- 第6章 模型评估和超参数调优的最佳实践 115
 - 6.1 用管道方法简化工作流 115
 - 6.1.1 加载威斯康星乳腺癌数据集 115
 - 6.1.2 集成管道中的转换器和评估器 116
 - 6.2 使用k折交叉验证评估模型的性能 118
 - 6.2.1 抵抗方法 118
 - 6.2.2 k折交叉验证 119
 - 6.3 用学习和验证曲线调试算法 122
 - 6.3.1 用学习曲线诊断偏差和方差问题 122
 - 6.3.2 用验证曲线解决过拟合和欠拟合问题 124
 - 6.4 通过网格搜索为机器学习模型调优 126
 - 6.4.1 通过网格搜索为超参数调优 126
 - 6.4.2 以嵌套式交叉验证来选择算法 127
 - 6.5 比较不同的性能评估指标 128
 - 6.5.1 含混矩阵分析 128
 - 6.5.2 优化分类模型的准确度和召回率 129
 - 6.5.3 绘制受试者操作特性图 130
 - 6.5.4 多元分类评分指标 133
 - 6.6 处理类的不平衡问题 133
 - 6.7 小结 135
- 第7章 综合不同模型的组合学习 136
 - 7.1 集成学习 136
 - 7.2 采用多数票机制的集成分类器 139
 - 7.2.1 实现基于多数票的简单分类器 139
 - 7.2.2 用多数票原则进行预测 143
 - 7.2.3 评估和优化集成分类器 145
 - 7.3 套袋—基于导引样本构建分类器集成 149
 - 7.3.1 套袋简介 150
 - 7.3.2 应用套袋技术对葡萄酒数据集中的样本分类 151
 - 7.4 通过自适应增强来利用弱学习者 153
 - 7.4.1 增强是如何实现的 154
 - 7.4.2 用scikit-learn实现AdaBoost 156
 - 7.5 小结 158
- 第8章 应用机器学习于情感分析 159
 - 8.1 为文本处理预备好IMDb电影评论数据 159
 - 8.1.1 获取电影评论数据集 159
 - 8.1.2 把电影评论数据预处理成更方便格式的数据 160
 - 8.2 词袋模型介绍 161
 - 8.2.1 把词转换成特征向量 161
 - 8.2.2 通过词频逆反文档频率评估单词相关性 162
 - 8.2.3 清洗文本数据 164
 - 8.2.4 把文档处理为令牌 165
 - 8.3 训练文档分类的逻辑回归模型 166
 - 8.4 处理更大的数据集—在线算法和核心学习 168
 - 8.5 具有潜在狄氏分配的主题建模 171
 - 8.5.1 使用LDA分解文本文档 171

- 8.5.2 LDA与scikit-learn 172
- 8.6 小结 174
- 第9章 将机器学习模型嵌入网络应用 175
 - 9.1 序列化拟合scikit-learn评估器 175
 - 9.2 搭建SQLite数据库存储数据 177
 - 9.3 用Flask开发网络应用 179
 - 9.3.1 第一个Flask网络应用 179
 - 9.3.2 表单验证与渲染 181
 - 9.4 将电影评论分类器转换为网络应用 184
 - 9.4.1 文件与文件夹—研究目录树 185
 - 9.4.2 实现主应用app.py 186
 - 9.4.3 建立评论表单 188
 - 9.4.4 创建一个结果页面的模板 189
 - 9.5 在面向公众的服务器上部署网络应用 190
 - 9.5.1 创建PythonAnywhere账户 190
 - 9.5.2 上传电影分类应用 191
 - 9.5.3 更新电影分类器 191
 - 9.6 小结 193
- 第10章 用回归分析预测连续目标变量 194
 - 10.1 线性回归简介 194
 - 10.1.1 简单线性回归 194
 - 10.1.2 多元线性回归 195
 - 10.2 探索住房数据集 196
 - 10.2.1 加载住房数据 196
 - 10.2.2 可视化数据集的重要特点 197
 - 10.2.3 用关联矩阵查看关系 198
 - 10.3 普通最小二乘线性回归模型的实现 200
 - 10.3.1 用梯度下降方法求解回归参数 200
 - 10.3.2 通过scikit-learn估计回归模型的系数 203
 - 10.4 利用RANSAC拟合稳健的回归模型 205
 - 10.5 评估线性回归模型的性能 206
 - 10.6 用正则化方法进行回归 209
 - 10.7 将线性回归模型转换为曲线—多项式回归 210
 - 10.7.1 用scikit-learn增加多项式的项 210
 - 10.7.2 为住房数据集中的非线性关系建模 211
 - 10.8 用随机森林处理非线性关系 214
 - 10.8.1 决策树回归 214
 - 10.8.2 随机森林回归 215
 - 10.9 小结 217
- 第11章 用聚类分析处理无标签数据 218
 - 11.1 用k-均值进行相似性分组 218
 - 11.1.1 scikit-learn的k-均值聚类 218
 - 11.1.2 k-均值++—更聪明地设置初始聚类中心的方法 221
 - 11.1.3 硬聚类与软聚类 222
 - 11.1.4 用肘法求解最佳聚类数 223
 - 11.1.5 通过轮廓图量化聚类质量 224
 - 11.2 把集群组织成有层次的树 228
 - 11.2.1 以自下而上的方式聚类 228
 - 11.2.2 在距离矩阵上进行层次聚类 229
 - 11.2.3 热度图附加树状图 232
 - 11.2.4 scikit-learn凝聚聚类方法 233
 - 11.3 通过DBSCAN定位高密度区域 233
 - 11.4 小结 237
- 第12章 从零开始实现多层人工神经网络 238

- 12.1 用人工神经网络为复杂函数建模 238
 - 12.1.1 单层神经网络扼要重述 239
 - 12.1.2 介绍多层神经网络体系 240
 - 12.1.3 利用正向传播激活神经网络 242
- 12.2 识别手写数字 243
 - 12.2.1 获取MNIST数据集 243
 - 12.2.2 实现一个多层感知器 247
- 12.3 训练人工神经网络 256
 - 12.3.1 逻辑成本函数的计算 256
 - 12.3.2 开发反向传播的直觉 257
 - 12.3.3 通过反向传播训练神经网络 258
- 12.4 关于神经网络的收敛性 260
- 12.5 关于神经网络实现的最后几句话 261
- 12.6 小结 261
- 第13章 用TensorFlow并行训练神经网络 262
 - 13.1 TensorFlow与模型训练的性能 262
 - 13.1.1 什么是TensorFlow 263
 - 13.1.2 如何学习TensorFlow 264
 - 13.1.3 学习TensorFlow的第一步 264
 - 13.1.4 使用阵列结构 266
 - 13.1.5 用TensorFlow的底层API开发简单的模型 267
 - 13.2 用TensorFlow的高级API高效率地训练神经网络 270
 - 13.2.1 用TensorFlow的Layers API构建多层神经网络 270
 - 13.2.2 用Keras研发多层神经网络 274
 - 13.3 多层网络激活函数的选择 277
 - 13.3.1 逻辑函数回顾 278
 - 13.3.2 在多元分类中调用softmax函数评估类别概率 279
 - 13.3.3 利用双曲正切拓宽输出范围 280
 - 13.3.4 修正线性单元激活函数 281
 - 13.4 小结 282
- 第14章 深入探讨TensorFlow的工作原理 283
 - 14.1 TensorFlow的主要功能 283
 - 14.2 TensorFlow的排序与张量 284
 - 14.3 了解TensorFlow的计算图 285
 - 14.4 TensorFlow中的占位符 287
 - 14.4.1 定义占位符 287
 - 14.4.2 为占位符提供数据 287
 - 14.4.3 用batchsizes为数据阵列定义占位符 288
 - 14.5 TensorFlow中的变量 289
 - 14.5.1 定义变量 289
 - 14.5.2 初始化变量 290
 - 14.5.3 变量范围 291
 - 14.5.4 变量复用 292
 - 14.6 建立回归模型 295
 - 14.7 在TensorFlow计算图中用张量名执行对象 297
 - 14.8 在TensorFlow中存储和恢复模型 298
 - 14.9 把张量转换成多维数据阵列 300
 - 14.10 利用控制流构图 303
 - 14.11 用TensorBoard可视化图 305
 - 14.12 小结 308
- 第15章 深度卷积神经网络图像识别 309
 - 15.1 构建卷积神经网络的模块 309
 - 15.1.1 理解CNN与学习特征的层次 309
 - 15.1.2 执行离散卷积 310

- 15.1.3 子采样 316
- 15.2 拼装构建CNN 317
 - 15.2.1 处理多个输入或者彩色频道 317
 - 15.2.2 通过淘汰正则化神经网络 319
- 15.3 用TensorFlow实现深度卷积神经网络 321
 - 15.3.1 多层CNN体系结构 321
 - 15.3.2 加载和预处理数据 322
 - 15.3.3 用TensorFlow的低级API实现CNN模型 323
 - 15.3.4 用TensorFlow 的Layers API实现CNN 332
- 15.4 小结 336
- 第16章 用递归神经网络为序列数据建模 338
 - 16.1 序列数据 338
 - 16.1.1 序列数据建模—顺序很重要 338
 - 16.1.2 表示序列 339
 - 16.1.3 不同类别的序列建模 339
 - 16.2 用于序列建模的RNN 340
 - 16.2.1 理解RNN的结构和数据流 340
 - 16.2.2 在RNN中计算激活值 341
 - 16.2.3 长期交互学习的挑战 343
 - 16.2.4 LSTM单元 343
 - 16.3 用TensorFlow实现多层RNN序列建模 345
 - 16.4 项目一：利用多层RNN对IMDb电影评论进行情感分析 345
 - 16.4.1 准备数据 345
 - 16.4.2 嵌入式 348
 - 16.4.3 构建一个RNN模型 350
 - 16.4.4 情感RNN类构造器 350
 - 16.4.5 build方法 351
 - 16.4.6 train方法 353
 - 16.4.7 predict方法 354
 - 16.4.8 创建SentimentRNN类的实例 355
 - 16.4.9 训练与优化情感分析RNN模型 355
 - 16.5 项目二：用TensorFlow实现字符级 RNN语言建模 356
 - 16.5.1 准备数据 356
 - 16.5.2 构建字符级RNN语言模型 359
 - 16.5.3 构造器 359
 - 16.5.4 build方法 360
 - 16.5.5 train方法 362
 - 16.5.6 sample方法 362
 - 16.5.7 创建和训练CharRNN模型 364
 - 16.5.8 处于取样状态的CharRNN模型 364
 - 16.6 总结 365
 - • • • • (收起)

[Python机器学习 \(原书第2版\) 下载链接1](#)

标签

Python

机器学习

人工智能

ML

计算机科学

计算机

数据科学

数据分析

评论

[Python机器学习 \(原书第2版\) 下载链接1](#)

书评

充其量不过是几个常用python ML包 (scikit NumPy SciPy matplotlib pandas) 的 cookbook 罢了。
基本上每节的流程就是先告诉你一个ML概念大概是怎么回事, 真的很大概, 不过好处是至少会告诉你为什么要这么做。然后用一段示例代码告诉你这个东西在Python ML包里要调用哪几个接口...

中文翻译 (非官方) [<https://www.gitbook.com/book/ljalpha/python-/details>]
=====

但是是有前提的： 1. 基础的线性代数知识需要大家温故知新一下； 2. 对于python中的numpy和pandas的一些基本操作需要熟悉； 3. 抽象能力，最好能把代数方程在大脑里映射出一个几何图形（最多三维）； 只要有了以上的前提，读这本书还是挺靠谱的。

[Python机器学习（原书第2版）下载链接1](#)