

Advanced Machine Learning with Python

Python 高级机器学习

[英] 约翰·哈蒂 著 王大鹏 管晨 郭鹏程

■ 用Python语言详细解读前沿机器学习技术，助你高效解决数据科学难题





图灵程序设计丛书

Advanced Machine Learning with Python

Python 高级机器学习

[英] 约翰·哈蒂 著

王大鹏 管晨 郭鹏程 译

人民邮电出版社

北 京

图书在版编目 (CIP) 数据

Python高级机器学习 / (英) 约翰·哈蒂
(John Hearty) 著 ; 王大鹏, 管晨, 郭鹏程译. -- 北京:
人民邮电出版社, 2020.1
(图灵程序设计丛书)
ISBN 978-7-115-52968-8

I. ①P… II. ①约… ②王… ③管… ④郭… III. ①
软件工具—程序设计 IV. ①TP311.561

中国版本图书馆CIP数据核字(2019)第290132号

内 容 提 要

Python 是杰出的脚本语言, 在速度、安全性和可扩展性等方面表现出众。在机器学习领域, 越来越多的从业者也选择使用易于上手、功能强大的 Python 语言。本书依次介绍无监督机器学习、深度信念网络、堆叠式降噪自编码器、卷积神经网络、半监督学习技术、文本特征工程, 模型集成方法, 以及其他一些实用的 Python 机器学习工具。除了理论知识与代码实现, 本书还细致讲解了分析和解决问题的思路, 极具启发性。

本书适合 Python 开发人员、分析师和数据科学家阅读。

-
- ◆ 著 [英] 约翰·哈蒂
译 王大鹏 管晨 郭鹏程
责任编辑 岳新欣
责任印制 周昇亮
 - ◆ 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: 12
字数: 283千字 2020年1月第1版
印数: 1—3 500册 2020年1月北京第1次印刷
著作权合同登记号 图字: 01-2017-9035号
-

定价: 59.00元

读者服务热线: (010)51095183转600 印装质量热线: (010)81055316

反盗版热线: (010)81055315

广告经营许可证: 京东工商广登字 20170147 号

版 权 声 明

Copyright © 2016 Packt Publishing. First published in the English language under the title *Advanced Machine Learning with Python*.

Simplified Chinese-language edition copyright © 2020 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。

前言

你好！欢迎阅读这本基于 Python 的高级机器学习指南。你很可能是因为最初萌发的兴趣而翻开本书，但并不确定能从中获得什么。简单地说，学习和运用机器学习技术从未像现在这样激动人心，而且，在机器学习领域中工作会使你长期受益。要想了解更高级的数据建模技术的最新知识或积累应用这些技术解决难题的经验，本书便是你的绝佳选择！

什么是高级机器学习

受计算能力持续高速增长（根据摩尔定律）的影响，机器学习已经从曾经的研究型学科逐渐演化为活跃在各个商业领域的技术。这导致了新应用和新的（或重新发掘的）技术大量涌现，也使得原本不为大众所熟知的数据科学、人工智能和机器学习等概念迅速走进人们的视野，并成为全球企业战略规划的一部分。

机器学习应用的飞速发展得益于人们对持续创新的不断追求和大量研究资源的投入。这些先驱开发的技术开拓了新的应用领域，并让大众进一步了解了这些技术。人工智能和机器学习在个别领域的应用革新仍然很难实现，但有些则已经成为现实，如自动驾驶汽车、复杂图像的识别和修正、基因研究的更大胆尝试，而最为普遍的可能是在线商店、电子邮件收件箱和网络生活中逐渐成熟的个性化内容推荐。

面对众多可能性，在数据科学家专注的努力下，这个行业正经历着虽有些坎坷但飞速的成长。不但数据科学家和人工智能从业者的数量比几年前（2014 年初）多了很多，而且高端机器学习研究的解决方案也变得更开放和更容易获取。

谷歌和 Facebook 的研究团队逐渐开始共享其架构、语言、模型和工具，以期它们能得到更多数据科学家的应用和改进。

机器学习社区已经足够成熟，能够不断创造或重新挖掘许多先进算法，而且这种趋势已经日益明显。更准确地说，曾经主要属于学术界的研究方向如今已经得到了业界的高度重视。成果之一是，现在出现了很多跨越业界和学术界的机器学习专家。另一个成果是，人们逐渐认识到一些高级算法可以用于攻克当今的一些前沿难题。随着时间的推移，我们也目睹了不断取得的新进展，其效果越来越好，前沿边界也在逐步推进。

综上所述,如果你想进入数据科学领域并培养机器学习技能,那么现在就是最好的时机。有关入门算法(包括聚类、回归模型和神经网络架构)及其工具的内容可以在众多在线课程和各类博文中找到。虽然掌握最前沿的数据科学技术(包括深度学习、半监督算法和集成模型)依然有难度,但这些技术本身已经可以在多种编程语言的软件库中使用了。你需要做的只是学习如何将理论知识与实际指导相结合,从而正确应用各种模型,这正是本书被赋予的使命。

你将从本书中得到什么

本书致力于解读近年来出现的一些高级建模技术。无论你是经验丰富的数据科学家,还是想充分利用现有技能进入新领域的开发人员,都可以通过本书学习这些算法。

我的首要目标是确保你能够理解书中所探讨的算法。其中一些算法非常棘手,而且它们与统计学和机器学习中的其他概念紧密相关。

强烈建议新手先大致了解以下重要概念。

- 包括 MLP 架构在内的神经网络架构
- 包括梯度下降法和反向传播算法在内的学习算法
- 网络模型的性能评估指标,如均方根误差
- k 均值聚类

有时,本书对所涉及的主题探讨得不够充分。因为本书的内容范围较广,所以进度会略快。在每章末尾,我都会推荐一些书或在线文章作为补充阅读材料,以便你就相关知识建立更广泛的基础。在阅读本书的过程中,如果遇到任何陌生概念,建议你进行一些额外的阅读,因为机器学习的知识之间具有协同的关联性;掌握得越多,就越容易在积累新的知识与技能时理解新概念。

持续积累新的知识与技能这一理念对于本书要达成的目标至关重要。每章都会介绍一种或多种算法,以期达成以下目标:

- 从高层次上解释某算法的功能、适合解决的问题及其应用方法
- 浏览算法的关键部分,包括拓扑结构、学习方法和性能评估
- 通过评估模型输出确定如何提升性能

除了传授知识和技能,本书还希望达到一个更重要的目的:讨论和传达出资深机器学习从业者常见的一些品质。其中一种品质是创造力,主要体现在定义复杂结构和处理特定问题中。另一种重要品质则是严谨,本书通篇强调了这种品质,特别是在有意义的目标的性能测定和早期工作的严格评估方面。

最后,本书无意掩盖挑战数据难题时的现实:早期试验好坏参半的结果、大量的反复计算和频繁出现的绝境。与此同时,我会用一系列示例、对专业方法的剖析,以及更符合现实的难题

（直到本书末尾）来展示如何用具有创造性、可靠和严谨的方法破除这些障碍，从而取得有意义的结果。

希望你在阅读过程中一切顺利，也希望你能在解决为你准备的问题、将所学知识应用于新领域或新数据的过程中获得乐趣。

我们现在就开始吧！

本书导读

第 1 章，无监督机器学习，介绍了如何用无监督学习技术来识别数据集的内在模式与结构。

第 2 章，深度信念网络，解释了受限玻尔兹曼机算法和深度信念网络算法的工作原理。通过学习这一章，你将了解如何应用这些算法，且更有把握通过评估输出结果来实现算法模型的改进与优化。

第 3 章，堆叠式降噪自编码器，通过用堆叠式降噪自编码器学习高维输入数据的特征表示，进一步提高你对深度架构的掌握程度。

第 4 章，卷积神经网络，展示了如何应用卷积神经网络（Convnet）。

第 5 章，半监督学习，阐释了如何应用多种半监督学习技术，如 CPLE、自主学习和 S3VM。

第 6 章，文本特征工程，探讨了一些数据预处理技巧，用于显著提高前述模型的性能和有效性。

第 7 章，特征工程 II，展示了如何清除或缓解数据质量问题来实现数据治理，以及如何针对具体的算法进行数据转换，并介绍了增强数据的一些技巧。

第 8 章，集成方法，探究了如何构建更复杂的集成模型，并介绍了提升模型结果稳健性的方法。

第 9 章，其他 Python 机器学习工具，回顾了近期数据科学家使用的最佳工具，阐述了各工具的优势，并以贯穿全书的工作过程讨论了如何配合前文介绍的工具和技术来应用这些工具。

附录，代码运行要求，讨论了本书所需要的工具，且分别指明了每一章所需要的库。

阅读前提

本书所使用的数据和代码都是公开的，包括开源 Python 库和框架。虽然每一章的示例代码都附有一个 README 文件，用于指明运行该章附带脚本中的代码需要安装的库，但我还是会将文件内容整理出来，以便你查阅。

建议你在阅读前几章时就安装好所需要的库，以便运行后续各章中的代码时使用。最好能提早安装第 1 章所需要的库，以便后文使用。

读者对象

本书面向渴望了解近期数据科学领域中最新的进展，以提高自身技能的 Python 开发人员、分析师和数据科学家。例如，如果你想构造自己的图片或文本标注解决方案，或者想要参加 Kaggle 大赛，那么本书就是为你而写！

如果具备一定的 Python 功底并了解机器学习的一些核心概念，那么你将可以更轻松地阅读本书。

排版约定

本书采用多种文本样式，以表示不同类型的信息。以下是一些样式的例子及其含义。

正文中的代码、数据库表名、用户输入将按如下样式显示：“通过使用下列代码，我们将 PCA 应用到手写 digits 数据集。”

代码块的格式如下所示：

```
import numpy as np
from sklearn.datasets import load_digits
import matplotlib.pyplot as plt
from sklearn.decomposition import PCA
from sklearn.preprocessing import scale
from sklearn.lda import LDA
import matplotlib.cm as cm
```

```
digits = load_digits()
data = digits.data
```

```
n_samples, n_features = data.shape
n_digits = len(np.unique(digits.target))
labels = digits.target
```

命令行输入或输出的格式如下所示：

```
[ 0.39276606  0.49571292  0.43933243  0.53573558  0.42459285
  0.55686854  0.4573401   0.49876358  0.50281585  0.4689295 ]

0.4772857426
```



此图标表示警告或重要注释。



此图标表示提示或技巧。

读者反馈

欢迎你随时提供阅读反馈，告诉我们你对本书的看法。读者反馈对我们很重要，它可以帮助我们开发更多读者能够真正充分利用的图书。

如有常规问题的反馈，请发送邮件至 feedback@packtpub.com，并在主题处注明书名。

如果你擅长某个话题，而且愿意撰写相关文章或投稿，请登录 www.packtpub.com/authors 阅读我们的作者指南。

用户支持

现在你已经成为 Packt 出版社作品的拥有者，因此我们为你准备了一系列内容，以帮助你充分发挥所购图书的作用。

下载示例代码

如果你是从 <http://www.packtpub.com> 网站购买的图书，登录自己的账号后就可以下载所有已购图书的示例代码。如果你是从其他地方购买的图书，请访问 <http://www.packtpub.com/support> 网站并注册，我们会将代码文件直接发送到你的电子邮箱。

你也可以通过以下步骤下载代码文件。

- (1) 使用你的电子邮箱地址和密码在我们的网址上登录或注册。
- (2) 将鼠标移到页面上方的 SUPPORT 标签。
- (3) 点击 Code Downloads & Errata。
- (4) 在 Search 框中输入书名。
- (5) 选择你需要下载代码文件的图书。
- (6) 在下拉菜单中选择你购买本书的途径。
- (7) 点击 Code Download。

下载文件后，确保使用以下工具的最新版本来解压或提取文件夹：

- WinRAR / 7-Zip (Windows)

- Zipeg / iZip / UnRarX (Mac)
- 7-Zip / PeaZip (Linux)

本书代码也托管在 GitHub 上, 访问 <https://github.com/bonzanini/Book-SocialMediaMiningPython> 即可获得。Packt 拥有丰富的图书和视频资源, GitHub 仓库 <https://github.com/PacktPublishing/> 提供了这些资源的相关代码。欢迎查阅!

下载本书的彩色图片

我们也为你提供了一份 PDF 文件, 其中包含了本书中的截屏和图表等彩色图片。这些彩色图片可以帮助你更好地理解输出的变化。下载地址: https://www.packtpub.com/sites/default/files/downloads/AdvancedMachineLearningwithPython_ColorImages.pdf。

勘误

虽然我们竭力确保图书内容的正确性, 但错误在所难免。如果你在我们出版的任何一本图书中发现了文本或代码错误, 希望你能告知我们, 我们将非常感激。你的善举足以减少其他读者在阅读出错内容时的纠结和不快, 并帮助我们在后续版本中更正错误。如果你发现任何错误, 请访问 <http://www.packtpub.com/submit-errata>, 选择相应图书, 点击 Errata Submission Form 链接, 并输入有关错误的信息。^①提交的错误得到验证后, 我们就会接受你的建议, 并将该处错误信息上传到我们的网站或添加到已有勘误表的相应位置。

访问 <https://www.packtpub.com/books/content/support> 并在搜索框中输入书名, 即可查看该图书已有的勘误信息。这部分信息会在 Errata 部分显示。

反盗版

任何媒体都会面临版权内容在互联网上的盗版问题, Packt 也不例外。Packt 非常重视版权保护。如果你发现我们的作品在互联网上被非法复制, 不管以什么形式, 都请立即为我们提供相关网址或网站名称, 以便我们进行补救。

请将疑似盗版材料的网址发送到 copyright@packtpub.com。

你的反盗版行动就是在保护作者和出版社, 只有这样, 我们才能继续以优质内容回馈像你这样的热心读者。

^① 本书中文版勘误请到 <https://www.it-ebooks.com.cn/book/1941> 查看和提交。——编者注

问题

如果对本书存有任何方面的疑问，可以通过 questions@packtpub.com 邮箱联系我们，我们将尽力为你答疑解惑。

致谢

我要感谢许多人，尤其是我的父母……感谢他们的耐心。还要感谢 Tyler Lowe 真挚的友情，感谢 Mark Huntley 对准确性的再三强调，感谢我之前在 Lionhead 工作室的团队。我还非常感谢 Jared Huffman 和 Packt 出版社勤勉的编辑团队所做的出色工作，在本书的创作过程中，他们一直给予我极大的信心和支持。

最后，我想把这本书献给你，我的读者。若想掌握本书的主题，没有比现在更好的时机了。如今的世界充满新机遇，而你可以用创造性的头脑和恰当模型来抓住机会。希望你在寻求解决方案时，总是能够取得成功。

电子书

扫描如下二维码，即可购买本书电子版。



目 录

第 1 章 无监督机器学习	1	第 3 章 堆叠式降噪自编码器	41
1.1 主成分分析	1	3.1 自编码器	41
1.1.1 主成分分析入门	2	3.1.1 自编码器简介	41
1.1.2 应用主成分分析	3	3.1.2 降噪自编码器	43
1.2 k 均值聚类	5	3.1.3 应用降噪自编码器	44
1.2.1 聚类入门	5	3.2 堆叠式降噪自编码器	47
1.2.2 开始聚类分析	6	3.2.1 应用堆叠式降噪自编码器	48
1.2.3 调整聚类参数	10	3.2.2 评估堆叠式降噪自编码机的性能	53
1.3 自组织映射	13	3.3 扩展阅读	54
1.3.1 自组织映射入门	13	3.4 小结	54
1.3.2 部署自组织映射	14	第 4 章 卷积神经网络	55
1.4 扩展阅读	17	4.1 CNN 介绍	55
1.5 小结	18	4.1.1 CNN 拓扑结构	56
第 2 章 深度信念网络	19	4.1.2 应用 CNN	66
2.1 神经网络入门	19	4.2 扩展阅读	71
2.1.1 神经网络的组成	20	4.3 小结	71
2.1.2 网络拓扑结构	20	第 5 章 半监督学习	72
2.2 受限玻尔兹曼机	23	5.1 简介	72
2.2.1 受限玻尔兹曼机简介	23	5.2 何为半监督学习	72
2.2.2 受限玻尔兹曼机的应用	26	5.3 半监督算法实战	73
2.2.3 受限玻尔兹曼机的扩展应用	35	5.3.1 自训练	73
2.3 深度信念网络	35	5.3.2 对比悲观似然估计	81
2.3.1 训练深度信念网络	36	5.4 扩展阅读	89
2.3.2 应用深度信念网络	36	5.5 小结	90
2.3.3 验证深度信念网络	39		
2.4 扩展阅读	40		
2.5 小结	40		

第 6 章 文本特征工程	91	8.1 集成简介	143
6.1 介绍	91	8.1.1 理解平均集成	144
6.2 文本特征工程	92	8.1.2 应用提升法	148
6.2.1 清洗文本数据	92	8.1.3 使用堆叠集成	153
6.2.2 根据文本数据构造特征	99	8.2 在动态应用中使用模型	157
6.2.3 测试准备好的数据	103	8.2.1 理解模型稳健性	158
6.3 扩展阅读	108	8.2.2 控制模型稳健性的策略	163
6.4 小结	109	8.3 扩展阅读	166
第 7 章 特征工程 II	110	8.4 小结	166
7.1 介绍	110	第 9 章 其他 Python 机器学习工具	167
7.2 创建特征集	110	9.1 可选的开发工具	167
7.2.1 为机器学习应用构建特征	111	9.1.1 Lasagne 简介	167
7.2.2 运用特征选择技术	117	9.1.2 TensorFlow 简介	169
7.3 特征工程实战	123	9.1.3 何时使用这些库	173
7.4 扩展阅读	141	9.2 扩展阅读	174
7.5 小结	142	9.3 小结	175
第 8 章 集成方法	143	附录 代码运行要求	176

第 1 章

无监督机器学习



本章将介绍如何用无监督学习技术来识别数据集中的模式和结构。

无监督学习是用于探索性分析的一系列很有价值的技术。它们能够挖掘出隐藏在数据集中的模式和结构，获取有用的信息或为进一步分析提供指导。拥有一套可靠的无监督学习工具集至关重要，你可以应用它们将陌生或复杂的数据集分解为可用信息。

我们将先从主成分分析（principal component analysis, PCA）这个基本的数据操作技术开始，并结合一系列的降维应用。接着探讨一项应用广泛且简单易用的无监督学习技术—— k 均值聚类。随后会讨论一种能将复杂数据集映射到二维的拓扑聚类方法——Kohonen 提出的自组织映射（self-organizing map, SOM）。

本章会用一些篇幅来探讨如何有效应用这些技术将高维数据集整合为易于处理的形式。我们将使用 UCI 手写数字数据集来展示每种算法的技术应用。在探讨和应用每种技术的过程中，我们将回顾一些实际应用和方法论的问题，尤其是如何校准和验证每种技术，以及哪些性能评测方法是有效的。综上，我们将按顺序讨论以下主题：

- 主成分分析
- k 均值聚类
- 自组织映射

1.1 主成分分析

为了有效处理高维数据集，我们必须有一套技术能将维度降至可处理的水平。降维的优势包括：可以在二维空间中对多维数据进行可视化、用最少的特征描述数据集所包含的信息，以及在某些情况下识别模型向量的共线性。



在机器学习的背景下，所谓的共线性是指模型特征共有的接近线性的关系。这些特征一般用处不大，原因很明显，相关特征不大可能比它们独立时提供更多的信息量。而且，共线特征可能会突出局部极小值或给出其他错误的线索。

当今应用最广泛的降维技术可能就是主成分分析了。本书将在多处使用主成分分析，因此最好先回顾一下该技术、理解其背后的理论，并编写能够有效应用主成分分析的 Python 代码。

1.1.1 主成分分析入门

主成分分析是一种强大的分解方法，它能够将变量极多的数据集分解为一系列正交向量。取足够多的正交向量，就能够解释数据集中近乎全部的方差了。从本质上来说，集合中的这些正交向量就是数据集的精简版描述。主成分分析用途广泛，很值得我们花时间学习。



这里稍微提醒一下，当分解后的向量维度小于原始数据集的变量个数时，往往会损失一些原始数据集中的信息。若向量数目足够多，这种损失通常很小，但如果将变量极多的数据集分解成很少的向量，那么就会损失“惨重”。因此，在应用主成分分析时，必须考虑对数据集有效建模所需要的向量数目。

主成分分析通过逐一辨别数据集中方差最大的方向（主成分）来提取向量，步骤如下所示：

- (1) 找出数据集的中心点；
- (2) 计算数据的协方差矩阵；
- (3) 计算协方差矩阵的特征向量；
- (4) 将特征向量标准正交化；
- (5) 计算每个特征向量表示的方差比例。

我们来简要介绍一下这些概念。

- **协方差**：多维数据的方差，两个或多个变量之间的方差。一维或一个变量的方差可由一个值来描述，两个变量之间的方差需要用一个 2×2 的矩阵来描述，三个变量之间的方差需要用一个 3×3 的矩阵来描述，以此类推。因此，主成分分析的第一步就是计算协方差矩阵。
- **特征向量**：一种针对数据集或线性变换的向量。具体来说，这种向量的方向在转换前后不会发生变化。为便于理解，假设你的两手之间有一根直的橡皮筋，你伸展橡皮筋，直到它绷紧为止。特征向量正是伸展前和伸展过程中不会发生方向变化的向量，也就是本例中直接穿过橡皮筋中心从一只手到另一只手的向量。
- **正交化**：寻找两个相互正交（成直角）的向量的过程。在 n 维数据空间中，正交化的过程就是选取一系列向量，然后转化为一组正交向量。
- **标准正交化**：对积进行标准化的正交化过程。
- **特征值**：（大致相当于特征向量的长度）用于计算由每个特征向量表示的方差比例，该比例由每个特征向量的特征值除以所有特征向量的特征值之和得到。

总的来说,协方差矩阵用于计算特征向量,标准正交过程则是根据特征向量得出正交的、标准化向量。特征值最大的特征向量是第一个主成分,其他成分的特征值都相对较小。这样一来,主成分分析算法就能够将数据集转化为用新的低维坐标系来表示。

1.1.2 应用主成分分析

概略地回顾了主成分分析算法后,接下来我们将其应用于一个重要的 Python 数据集,即 UCI 手写数字数据集 (digits),它是作为 scikit-learn 的一部分发布的。

该数据集包含了来自 44 位作者的 1797 个手写数字。对这些作者的手写数字的笔压和位置信息使用 8×8 的点阵,经过两次重复采样后,得到了图 1-1 中的映射。

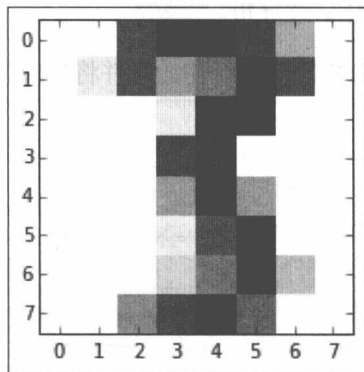


图 1-1

这些映射可以转化为维度为 64 的特征向量,从而作为分析输入项。对于含有 64 个特征的输入数据集,我们往往会立即想到使用诸如主成分分析的技术,从而将变量减少到可处理的数量。目前来看,我们无法用可视化探索来有效地分析数据集。

先对 digits 数据集应用主成分分析技术,代码如下所示:

```
import numpy as np
from sklearn.datasets import load_digits
import matplotlib.pyplot as plt
from sklearn.decomposition import PCA
from sklearn.preprocessing import scale
from sklearn.lda import LDA
import matplotlib.cm as cm

digits = load_digits()
data = digits.data

n_samples, n_features = data.shape
n_digits = len(np.unique(digits.target))
labels = digits.target
```

这段代码完成了以下工作。

首先，它加载了一系列必要的库，包括 `numpy` 库、`scikit-learn` 库和用于绘图的 `matplotlib` 库。其中 `scikit-learn` 库主要包括 `digits` 数据集、主成分分析算法和数据标准化函数等一系列相关数据、算法和函数。

接着，开始准备 `digits` 数据集。步骤如下所示。

- ❑ 首先，在构造有用的变量前加载数据集。
- ❑ 创建 `data` 变量用于表示数据集，并根据 `digits` 数据集中的目标向量计算出因变量的类别数（因为数字取值范围为 0~9，所以 `n_digits = 10`）并保存为变量，以供后续分析使用。
- ❑ 将 `target` 向量存为变量，并命名为 `labels`，以便后续使用。
- ❑ 以上变量构造均是为了简化后续分析。

准备好数据集后，对主成分分析算法进行初始化并将其应用于数据集。

```
pca = PCA(n_components=10)
data_r = pca.fit(data).transform(data)

print('explained variance ratio (first two components): %s' %
      str(pca.explained_variance_ratio_))
print('sum of explained variance (first two components): %s' %
      str(sum(pca.explained_variance_ratio_)))
```

以上代码会输出由前 10 个主成分解释的方差，按解释能力由高到低排序。

本例中的 10 个主成分共同解释了 0.589 的总方差。这结果不算太差，毕竟我们将 64 个变量减少到 10 个主成分，但也的确反映出了主成分分析带来的潜在损失。然而，关键问题是，精简后的变量集能否便于后续的分析或分类，也就是说，降维向量保留的方差信息是否会影响分类预测效果。

用 `pca` 处理 `digits` 数据集，并将输出结果保存到新变量 `data_r` 中后，可以对输出进行可视化。为此，首先需要构建 `colors` 向量来给类上色，然后便可以绘制散点图：

```
X = np.arange(10)
ys = [i+x+(i*x)**2 for i in range(10)]

plt.figure()
colors = cm.rainbow(np.linspace(0, 1, len(ys)))
for c, i, target_name in zip(colors, [1,2,3,4,5,6,7,8,9,10], labels):
    plt.scatter(data_r[labels == i, 0], data_r[labels == i, 1],
               c=c, alpha = 0.4)
plt.legend()
plt.title('Scatterplot of Points plotted in first \'10\' Principal Components')
plt.show()
```