TURING 图灵程序设计丛书

Apress®

Python Data Analytics

Python 数据分析实战

了解数据分析全貌，全面掌握用Python语言及专业的库进行数据分析，驾驭大数据

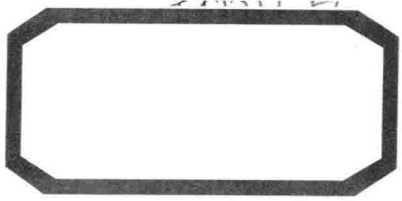
【意】 Fabio Nelli 著

杜春晓 译

中国工信出版集团

人民邮电出版社 POSTS & TELECOM PRESS

TURING 图灵程序设计丛书



Python Data Analytics

Python 数据分析实战

【意】 Fabio Nelli 著
杜春晓 译

人民邮电出版社
北 京

图书在版编目（CIP）数据

Python数据分析实战 / （意）内利（Fabio Nelli）著 ; 杜春晓译. -- 北京 : 人民邮电出版社, 2016.8
（图灵程序设计丛书）
ISBN 978-7-115-43220-9

Ⅰ. ①P… Ⅱ. ①内… ②杜… Ⅲ. ①软件工具－程序设计 Ⅳ. ①TP311.56

中国版本图书馆CIP数据核字(2016)第180326号

内 容 提 要

Python简单易学，拥有丰富的库，并且具有极强的包容性。本书展示了如何利用Python语言的强大功能，以最小的编程代价进行数据的提取、处理和分析，主要内容包括：数据分析和Python的基本介绍，NumPy库，pandas库，如何使用pandas读写和提取数据，用matplotlib库和scikit-learn库分别实现数据可视化和机器学习，以实例演示如何从原始数据获得信息、D3库嵌入和手写体数字的识别。

本书适合数据分析师等所有需要进行数据采集分析的工作人员。

◆ 著 [意] Fabio Nelli
译 杜春晓
责任编辑 朱 巍
执行编辑 贺子娟 李 敏
责任印制 彭志环

◆ 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
北京市昌平百善印刷厂印刷

◆ 开本：800×1000 1/16
印张：18.75
字数：443千字 2016年8月第1版
印数：1－3 500册 2016年8月北京第1次印刷
著作权合同登记号 图字：01-2016-5330号

定价：59.00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：京东工商广字第8052号

版权声明

致　　谢

感谢我的友人，尤其要感谢Alberto、Daniele、Roberto和Alex。本书写作历时一年，其间困难重重，挫折不断，感谢他们包容我，并给予我宝贵的精神支持。

向我的母亲致以最深的谢意。

译 者 序

不知不觉，结识Python已有七个年头，掰着指头数完，不禁惊恐于光阴之易逝，叹人生之不能加长，更兼想起宇宙之无穷，免不了慨叹生命之短暂而微茫，此时更觉Python之哲学观“人生短暂，我用Python”虽朴实无华，却更楚楚动人。七年前，国内Python书籍寥寥无几，七年后已能排满一个小书架。见Python这样的好东西为世人所接受，颇感欣慰。Python在Web开发、网络编程、自然语言处理、图像处理以及本书所讲的数据分析等诸多领域都有着广泛的应用。Python简单易学，新手见它大可不必发怵；Python拥有丰富的库，开发者不必重造轮子；Python有极强的包容性，可整合C、C++等语言的代码，弥补其性能上的不足；再加上近年来计算机效率的提高，Python的优势越发凸显出来。即使是其他领域的从业者，为满足业务的需要，若在性能上没有追求极致的嗜好，Python很可能是最适合的编程语言，对于数据分析亦是如此。

互联网的迅猛发展带来了数据量的指数级增长，而数据增长速度仍在加快。无论从规模上还是结构上讲，数据分析工作面对的对象较以往发生了质的改变。各行各业所产生的数据，在过去也许只是被视为副产品，如今也有可能显露它们的价值。但数据规模之大，结构之复杂，纯人工难以胜任，求诸工具是必然。为了满足这一需求，Python数据分析的羽翼随之不停地生长，如今已丰满有力。NumPy、pandas和matplotlib等库提供了矩阵运算、数据读写和处理及绘图等一揽子解决方案。就拿数据可视化来讲，3D、等值线图、地区分布图和玫瑰图等统统不在话下，用IPython Notebook分析完数据，可直接生成各种图表，你甚至还可以拿它来做汇报，连PPT都省了。

诚如作者在书中所言，用Python做数据分析根本不用羡慕其他语言的数据分析工具集。然而，好工具摆在那里，但没有明白人教也是白搭，不过，你手中所托之书刚好能充当这方面的良师益友。它不仅能带你一览数据分析全貌，更力求一招一式地教会你数据分析的十八般武艺。本书示例颇丰，在学习过程中，若能打开IPython Notebook，一点点跟着作者比划，想必新人也能出师，而有一定水平的开发者则可将其作为案头常备的参考书，以便节省不少查阅文档的时间。本书最后，作者举了三个实例，以加深读者对数据分析全过程的理解。第一个例子，用数据分析方法探索海洋对气候的影响，你也许能从中得出足以令当年地理老师心服口服的结论。第二个例子讲解了地区分布图的制作，你会发现原来我们还可以在Python环境中使用JavaScript！第三个例子则是用机器学习方法解决经典的图像识别问题。说了这么多，你是不是像我一样觉得这本书很有趣？在翻译本书的过程中，我安装了Anaconda，从此迷上了IPython Notebook的交互性和直观生动，想必你也会为之所动。

经冬历春，译文乃成，其中半数章节的第一稿是在寒假期间完成的。在家那段时间，有时只

有我跟宝宝在屋里，我很想哄她开心，可为了保证进度，实在脱不开身。她要是哭，我就敲敲桌子，抑或是招招手，她见还有人在就不哭了，可怜的娃。岳父岳母一家人帮着妻子照看孩子不辞辛苦。我在翻译时，他们还会悄悄端过一杯茶来。地脏了，他们就轻轻把地拖了，生怕打扰我。到了饭点，岳母又会张罗出一桌可口的饭菜，如此种种，甚是难忘。没有他们，我哪有闲心码字。

感谢作者Fabio Nelli给我们带来了一顿数据分析的饕餮大餐。感谢图灵公司的朱巍编辑等诸位朋友，本书中文版的顺利出版离不开你们幕后的辛勤付出。此外，邵有生阅读了第1章和第2章译稿，范明武阅读了第3章，研究生同学黄毅阅读了第7章。他们发现了几处错误，并提出了很多非常有价值的修改意见，在此对他们表示诚挚的谢意。感谢在翻译过程中给予我鼓励、支持和帮助的诸位老师、同事和朋友，他们是路本福、都帮森、蔡波、蔡颖、陈健锁、韩旭、李玲玲、秦敏、王海霞、辛欣和王晶，我曾向他们中的几位请教过某些专业问题。感谢我初中时代的地理老师朱怡峰老先生，他激发了我对地理学科的兴趣，以至于多少年后，我在翻译本书第9章气象数据分析时会感到趣味盎然。最后感谢我的父亲和姐姐，他们以我翻译本书为荣。

由于本人学识有限，且时间仓促，书中翻译错误、不当和疏漏之处在所难免，还望读者批评指正。

杜春晓

2016年4月26日

目　录

第 1 章　数据分析简介 …… 1
1.1　数据分析 …… 1
1.2　数据分析师的知识范畴 …… 2
1.2.1　计算机科学 …… 2
1.2.2　数学和统计学 …… 3
1.2.3　机器学习和人工智能 …… 3
1.2.4　数据来源领域 …… 3
1.3　理解数据的性质 …… 4
1.3.1　数据到信息的转变 …… 4
1.3.2　信息到知识的转变 …… 4
1.3.3　数据的类型 …… 4
1.4　数据分析过程 …… 4
1.4.1　问题定义 …… 5
1.4.2　数据抽取 …… 6
1.4.3　数据准备 …… 6
1.4.4　数据探索和可视化 …… 7
1.4.5　预测模型 …… 7
1.4.6　模型评估 …… 8
1.4.7　部署 …… 8
1.5　定量和定性数据分析 …… 9
1.6　开放数据 …… 9
1.7　Python 和数据分析 …… 11
1.8　结论 …… 11
第 2 章　Python 世界简介 …… 12
2.1　Python——编程语言 …… 12
2.2　Python——解释器 …… 13
2.2.1　Cython …… 14
2.2.2　Jython …… 14
2.2.3　PyPy …… 14
2.3　Python 2 和 Python 3 …… 14
2.4　安装 Python …… 15
2.5　Python 发行版 …… 15
2.5.1　Anaconda …… 15
2.5.2　Enthought Canopy …… 16
2.5.3　Python(x,y) …… 17
2.6　使用 Python …… 17
2.6.1　Python shell …… 17
2.6.2　运行完整的 Python 程序 …… 17
2.6.3　使用 IDE 编写代码 …… 18
2.6.4　跟 Python 交互 …… 18
2.7　编写 Python 代码 …… 18
2.7.1　数学运算 …… 18
2.7.2　导入新的库和函数 …… 19
2.7.3　函数式编程 …… 21
2.7.4　缩进 …… 22
2.8　IPython …… 23
2.8.1　IPython shell …… 23
2.8.2　IPython Qt-Console …… 24
2.9　PyPI 仓库——Python 包索引 …… 25
2.10　多种 Python IDE …… 26
2.10.1　IDLE …… 26
2.10.2　Spyder …… 27
2.10.3　Eclipse（pyDev） …… 27
2.10.4　Sublime …… 28
2.10.5　Liclipse …… 29
2.10.6　NinjaIDE …… 29
2.10.7　Komodo IDE …… 29
2.11　SciPy …… 30
2.11.1　NumPy …… 30
2.11.2　pandas …… 30
2.11.3　matplotlib …… 31

2.12 小结 …… 31

第 3 章 NumPy 库 …… 32

3.1 NumPy 简史 …… 32
3.2 NumPy 安装 …… 32
3.3 ndarray：NumPy 库的心脏 …… 33
3.3.1 创建数组 …… 34
3.3.2 数据类型 …… 34
3.3.3 dtype 选项 …… 35
3.3.4 自带的数组创建方法 …… 36
3.4 基本操作 …… 37
3.4.1 算术运算符 …… 37
3.4.2 矩阵积 …… 38
3.4.3 自增和自减运算符 …… 39
3.4.4 通用函数 …… 40
3.4.5 聚合函数 …… 40
3.5 索引机制、切片和迭代方法 …… 41
3.5.1 索引机制 …… 41
3.5.2 切片操作 …… 42
3.5.3 数组迭代 …… 43
3.6 条件和布尔数组 …… 45
3.7 形状变换 …… 45
3.8 数组操作 …… 46
3.8.1 连接数组 …… 46
3.8.2 数组切分 …… 47
3.9 常用概念 …… 49
3.9.1 对象的副本或视图 …… 49
3.9.2 向量化 …… 50
3.9.3 广播机制 …… 50
3.10 结构化数组 …… 52
3.11 数组数据文件的读写 …… 53
3.11.1 二进制文件的读写 …… 54
3.11.2 读取文件中的列表形式数据 …… 54
3.12 小结 …… 55

第 4 章 pandas 库简介 …… 56

4.1 pandas：Python 数据分析库 …… 56
4.2 安装 …… 57
4.2.1 用 Anaconda 安装 …… 57
4.2.2 用 PyPI 安装 …… 58
4.2.3 在 Linux 系统的安装方法 …… 58
4.2.4 用源代码安装 …… 58
4.2.5 Windows 模块仓库 …… 59
4.3 测试 pandas 是否安装成功 …… 59
4.4 开始 pandas 之旅 …… 59
4.5 pandas 数据结构简介 …… 60
4.5.1 Series 对象 …… 60
4.5.2 DataFrame 对象 …… 66
4.5.3 Index 对象 …… 72
4.6 索引对象的其他功能 …… 74
4.6.1 更换索引 …… 74
4.6.2 删除 …… 75
4.6.3 算术和数据对齐 …… 77
4.7 数据结构之间的运算 …… 78
4.7.1 灵活的算术运算方法 …… 78
4.7.2 DataFrame 和 Series 对象之间的运算 …… 78
4.8 函数应用和映射 …… 79
4.8.1 操作元素的函数 …… 79
4.8.2 按行或列执行操作的函数 …… 80
4.8.3 统计函数 …… 81
4.9 排序和排位次 …… 81
4.10 相关性和协方差 …… 84
4.11 NaN 数据 …… 85
4.11.1 为元素赋 NaN 值 …… 85
4.11.2 过滤 NaN …… 86
4.11.3 为 NaN 元素填充其他值 …… 86
4.12 等级索引和分级 …… 87
4.12.1 重新调整顺序和为层级排序 …… 89
4.12.2 按层级统计数据 …… 89
4.13 小结 …… 90

第 5 章 pandas：数据读写 …… 91

5.1 I/O API 工具 …… 91
5.2 CSV 和文本文件 …… 92
5.3 读取 CSV 或文本文件中的数据 …… 92
5.3.1 用 RegExp 解析 TXT 文件 …… 94
5.3.2 从 TXT 文件读取部分数据 …… 96
5.3.3 往 CSV 文件写入数据 …… 97
5.4 读写 HTML 文件 …… 98

5.4.1　写入数据到 HTML 文件 ……99
5.4.2　从 HTML 文件读取数据 ……100
5.5　从 XML 读取数据 ……101
5.6　读写 Microsoft Excel 文件 ……103
5.7　JSON 数据 ……105
5.8　HDF5 格式 ……107
5.9　pickle——Python 对象序列化 ……108
5.9.1　用 cPickle 实现 Python 对象序列化 ……109
5.9.2　用 pandas 实现对象序列化 ……109
5.10　对接数据库 ……110
5.10.1　SQLite3 数据读写 ……111
5.10.2　PostgreSQL 数据读写 ……112
5.11　NoSQL 数据库 MongoDB 数据读写 …114
5.12　小结 ……116

第 6 章　深入 pandas：数据处理 ……117

6.1　数据准备 ……117
6.2　拼接 ……122
6.2.1　组合 ……124
6.2.2　轴向旋转 ……125
6.2.3　删除 ……127
6.3　数据转换 ……128
6.3.1　删除重复元素 ……128
6.3.2　映射 ……129
6.4　离散化和面元划分 ……132
6.5　排序 ……136
6.6　字符串处理 ……137
6.6.1　内置的字符串处理方法 ……137
6.6.2　正则表达式 ……139
6.7　数据聚合 ……140
6.7.1　GroupBy ……141
6.7.2　实例 ……141
6.7.3　等级分组 ……142
6.8　组迭代 ……143
6.8.1　链式转换 ……144
6.8.2　分组函数 ……145
6.9　高级数据聚合 ……145
6.10　小结 ……148

第 7 章　用 matplotlib 实现数据可视化 …149

7.1　matplotlib 库 ……149
7.2　安装 ……150
7.3　IPython 和 IPython QtConsole ……150
7.4　matplotlib 架构 ……151
7.4.1　Backend 层 ……152
7.4.2　Artist 层 ……152
7.4.3　Scripting 层（pyplot） ……153
7.4.4　pylab 和 pyplot ……153
7.5　pyplot ……154
7.5.1　生成一幅简单的交互式图表 ……154
7.5.2　设置图形的属性 ……156
7.5.3　matplotlib 和 NumPy ……158
7.6　使用 kwargs ……160
7.7　为图表添加更多元素 ……162
7.7.1　添加文本 ……162
7.7.2　添加网格 ……165
7.7.3　添加图例 ……166
7.8　保存图表 ……168
7.8.1　保存代码 ……169
7.8.2　将会话转换为 HTML 文件 ……170
7.8.3　将图表直接保存为图片 ……171
7.9　处理日期值 ……171
7.10　图表类型 ……173
7.11　线性图 ……173
7.12　直方图 ……180
7.13　条状图 ……181
7.13.1　水平条状图 ……183
7.13.2　多序列条状图 ……184
7.13.3　为 pandas DataFrame 生成多序列条状图 ……185
7.13.4　多序列堆积条状图 ……186
7.13.5　为 pandas DataFrame 绘制堆积条状图 ……189
7.13.6　其他条状图 ……190
7.14　饼图 ……190
7.15　高级图表 ……193
7.15.1　等值线图 ……193
7.15.2　极区图 ……195

7.16 mplot3d ····· 197
7.16.1 3D 曲面 ····· 197
7.16.2 3D 散点图 ····· 198
7.16.3 3D 条状图 ····· 199
7.17 多面板图形 ····· 200
7.17.1 在其他子图中显示子图 ····· 200
7.17.2 子图网格 ····· 202
7.18 小结 ····· 204

第 8 章 用 scikit-learn 库实现机器学习 ····· 205

8.1 scikit-learn 库 ····· 205
8.2 机器学习 ····· 205
8.2.1 有监督和无监督学习 ····· 205
8.2.2 训练集和测试集 ····· 206
8.3 用 scikit-learn 实现有监督学习 ····· 206
8.4 Iris 数据集 ····· 206
8.5 K-近邻分类器 ····· 211
8.6 Diabetes 数据集 ····· 214
8.7 线性回归：最小平方回归 ····· 215
8.8 支持向量机 ····· 219
8.8.1 支持向量分类 ····· 219
8.8.2 非线性 SVC ····· 223
8.8.3 绘制 SVM 分类器对 Iris 数据集的分类效果图 ····· 225
8.8.4 支持向量回归 ····· 227
8.9 小结 ····· 229

第 9 章 数据分析实例——气象数据 ····· 230

9.1 待检验的假设：靠海对气候的影响 ····· 230
9.2 数据源 ····· 233
9.3 用 IPython Notebook 做数据分析 ····· 234
9.4 风向频率玫瑰图 ····· 246
9.5 小结 ····· 251

第 10 章 IPython Notebook 内嵌 JavaScript 库 D3 ····· 252

10.1 开放的人口数据源 ····· 252
10.2 JavaScript 库 D3 ····· 255
10.3 绘制簇状条状图 ····· 259
10.4 地区分布图 ····· 262
10.5 2014 年美国人口地区分布图 ····· 266
10.6 小结 ····· 270

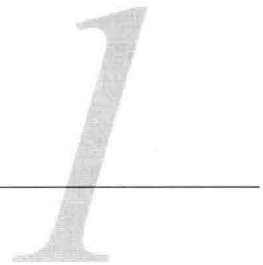
第 11 章 识别手写体数字 ····· 271

11.1 手写体识别 ····· 271
11.2 用 scikit-learn 识别手写体数字 ····· 271
11.3 Digits 数据集 ····· 272
11.4 学习和预测 ····· 274
11.5 小结 ····· 276

附录 A 用 LaTeX 编写数学表达式 ····· 277

附录 B 开放数据源 ····· 287

第1章 数据分析简介



欢迎来到数据分析世界。作为后续章节的铺垫，本章介绍数据分析的主要概念和流程。完成本章的学习，你就能在数据分析的世界中迈出坚实的一步。其余章节会陆续介绍如何借助Python库，把在这里学到的概念和流程转化为Python代码。

1.1 数据分析

当今世界对信息技术的依赖程度日渐加深，每天都会产生和存储海量的数据。数据的来源多种多样——自动检测系统、传感器和科学仪器等。不知你有没有意识到，你每次从银行取钱、买东西、写博客、发微博也会产生新的数据。

什么是数据呢？数据实际上不同于信息，至少在形式上不一样。对于没有任何形式可言的字节流，除了其数量、用词和发送的时间外，其他一无所知，一眼看上去，很难理解其本质。信息实际上是对数据集进行处理，从中提炼出可用于其他场合的结论，也就是说，它是对数据进行处理后得到的结果。从原始数据中抽取信息的这个过程叫作*数据分析*。

数据分析的目的正是抽取不易推断的信息，而一旦理解了这些信息，就能够对产生数据的系统的运行机制进行研究，从而对系统可能的响应和演变作出预测。

数据分析最初用作数据保护，现已发展成为*数据建模*的方法论，从而完成了到一门真正学科的蜕变。模型实际上是指将所研究的系统转化为数学形式。一旦建立数学或逻辑模型，对系统的响应能作出不同精度的预测，我们就可以预测在给定输入的情况下，系统会给出怎样的输出。这样看来，数据分析的目标不止于建模，更重要的是其*预测能力*。

模型的预测能力不仅取决于建模技术的质量，还取决于选择供分析用的优质数据集的能力。因此*数据搜寻*、*数据提取*和*数据准备*等预处理工作也属于数据分析的范畴，它们对最终结果有重要影响。

到现在为止，我们一直在讲数据、数据的准备及数据处理。在数据分析的各个阶段，还有各种各样的*数据可视化*方法。无论是孤立地看数据，还是将其放到整个数据集来看，理解数据的最好方法莫过于将其做成可视化图形，从而传达出数字中蕴含（有时是隐藏着）的信息。到目前为止，已经有很多可视化模式：类型多样的图表。

数据分析的产出为模型和图形化展示，据此可预测所研究系统的响应；随后进入测试阶段，

用已知输出结果的一个数据集对模型进行测试。这些数据不是用来生成模型的，而是用来检验系统能否重现实际观察到的输出，从而掌握模型的误差，了解其有效性和局限。

拿新模型的测试结果与既有模型进行对比便可知优劣。如新模型胜出，即可进行数据分析的最后一步：部署。部署阶段需要根据模型给出的预测结果，实现相应的决策，同时还要防范模型预测到的潜在风险。

很多工作都离不开数据分析。了解数据分析及实际操作方法，对工作中做出可靠决策大有裨益。有了它，人们可以检验假说，加深对系统的理解。

1.2　数据分析师的知识范畴

数据分析学科研究的问题面很广。数据分析过程要用到多种工具和方法，它们对计算、数学和统计思维要求较高。

因此，一名优秀的数据分析师必须具备多个学科的知识和实际应用能力。这些学科中有的是数据分析方法的基础，熟练掌握它们很有必要。根据应用领域、研究项目的不同，数据分析师可能还需要掌握其他相关学科的知识。总的来说，这些知识可以帮助分析师更好地理解研究对象以及需要什么样的数据。

一般而言，对于大的数据分析项目，最好组建一个由各个相关领域的专家组成的团队，他们要能在各自擅长的领域发挥出最大作用。对于小点的项目，一名优秀的分析师就能胜任，但是他必须善于识别数据分析过程中遇到的问题，知道解决问题需要哪些学科的知识和技能，并能及时学习这些学科，有时甚至需要向相关领域的专家请教。简言之，分析师不仅要知道怎么搜寻数据，更应该懂得怎么寻找处理数据的方法。

1.2.1　计算机科学

不论从事什么领域的数据分析工作，掌握计算机科学知识对分析师来说都是最基本的要求。只有具备良好的计算机科学知识及实际应用经验，才能熟练掌握数据分析必备工具。事实上，数据分析的各个步骤都离不开计算机技术，比如用于计算的软件（IDL、Matlab等）和编程语言（C++、Java、Python等）。

要高效地处理随信息技术迅猛发展而产生的海量数据，就必须用到特定的技能。数据研究和抽取，要求分析师掌握各种常见格式的处理技巧。数据通常以某种结构组织在一起，存储于文件或数据库表中，格式多样。常见的数据存储格式有XML、JSON、XLS、CSV等。很多应用都能处理这些格式的数据文件。从数据库中获取数据要稍微麻烦些，需要掌握SQL数据库查询语言，或使用专门为从某种数据库抽取数据而开发的软件。

此外，一些特定类型的数据研究任务中，分析师所能拿到的不是立刻就能用的干净数据，而是文本文件（文档、日志）或网页。需要的数据则来自这些文件中的图表、测量值、访客量或者HTML表格，而解析文件、抽取数据（数据抓取）需要专业知识。

因此，学习信息技术知识很有必要，只有这样才能掌握在当代计算机科学基础上发展起来的

各种工具，比如软件和编程语言。数据分析和可视化离不开它们。

本书尽可能全面地介绍用Python编程语言及专业的库进行数据分析所需的全部知识。针对数据分析的各个阶段，从数据研究、数据挖掘到预测模型研究结果的部署，Python都有专门的库。

1.2.2 数学和统计学

数据分析涉及大量数学知识，本书全篇都少不了它们的身影。数据处理和分析过程涉及的数学知识可能会很复杂。因此具备扎实的数学功底显得尤为重要，至少要能够理解正在做的事。熟悉常用的统计学概念也很有必要，因为所有对数据进行的分析和解释都以这些概念为基础。如果说计算机科学提供的是数据分析工具，那么统计学提供的就是基础概念。

统计学为分析师提供了很多工具和方法，全部掌握它们需要多年的磨练。数据分析领域最常用到的统计技术有：

- 贝叶斯方法
- 回归
- 聚类

用到这些方法时，会发现其中数学和统计学知识紧密结合，且对两者都有很高的要求。但是在本书中所讲述的Python库的帮助下，读者将有能力驾驭它们。

1.2.3 机器学习和人工智能

数据分析领域最先进的工具之一就是机器学习方法。实际上，尽管数据可视化以及聚类和回归等技术对分析师发现有价值的信息有很大帮助，但在数据分析过程中，分析师经常需要查询数据集中的各种模式，这些步骤专业性很强。

机器学习这门学科所研究的正是如何把一系列步骤和算法结合起来，分析数据，识别数据中存在的模式，找出不同的簇，发现趋势，从数据中抽取有用信息用于数据分析，并实现整个过程的自动化。

机器学习日渐成为数据分析的基础工具，因此了解它（至少也要知道个大概）对数据分析工作的重要性不言而喻。

1.2.4 数据来源领域

数据来源领域（生物、物理、金融、材料试验和人口统计等）的知识也是非常重要的一块。事实上，分析师虽然受过统计学的专业训练，但是他也必须深入到应用领域，记录原始数据，以便更好地理解数据生成过程。此外，数据不仅仅是干巴巴的字符串或数字，还是实际观测参数的表达式，更确切地说是其度量值。因此，对数据来源领域有深入的理解，能够提升解释数据的能力。当然，即使是对乐意学习的分析师来说，学习特定领域的知识也是要下一番工夫的。因此最好能找到相关领域的专家，以便有问题时及时咨询。

1.3　理解数据的性质

数据分析所研究的对象自然是数据。在数据分析的各个阶段，数据都是主要关注对象。要分析、处理的原材料由数据构成。经过处理、分析数据后，最终可能会从中得到有用的信息。这些信息能够增加对研究对象，也就是产生原始数据的系统理解。

1.3.1　数据到信息的转变

数据是对世界万物的记录。任何可以被测量或是分类的事物都能用数据来表示。采集完数据后，可以对其进行研究和分析，以理解事物的性质。人们也常常借助它们进行预测，或者即使做不到预测，至少也能让推测更加有根据。

1.3.2　信息到知识的转变

当信息转化为一组有助于更好地理解特定机制的规则时，就说信息已转化为知识，我们也因而可以用这些知识预测事件的演变。

1.3.3　数据的类型

数据可以分为两个不同的类别：

- 类别型
  - 定类
  - 定序
- 数值型
  - 离散
  - 连续

*类别型数据*指可以被分成不同组或类别的值或观察结果。有两种类别型数据：*定类*（nominal）和*定序*（ordinal）。定类型变量的各类别没有内在的顺序，而定序型变量有预先指定的顺序。

*数值型数据*指通过测量得到的数值或观察结果。有两种不同的数值型数据：*离散型*和*连续型*。离散值的个数是可数的，每个值都与其他值区别开来。相反，连续值产生于结果属于某一确定范围的测量或观察。

1.4　数据分析过程

数据分析过程可以用以下几步来描述：转换和处理原始数据，以可视化方式呈现数据，建模做预测。因此，数据分析无外乎由几步组成，其中每一步所起的作用对后面几步而言都至关重要。因此数据分析几乎可以概括为由以下几个阶段组成的过程链：

- 问题定义

- 数据抽取
- 数据清洗
- 数据转换
- 数据探索
- 预测模型
- 模型评估/测试
- 结果可视化和阐释
- 解决方案部署

图1-1为数据分析各步骤的示意图。

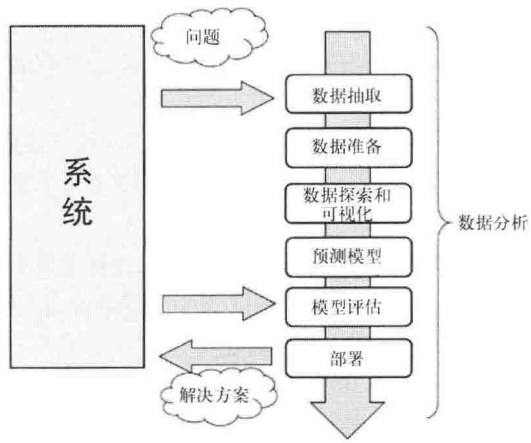


图1-1 数据分析过程

1.4.1 问题定义

采集原始数据前，数据分析过程实际上早已开始。事实上，数据分析总是始于要解决的问题，而这个问题需要事先定义。

只有深入探究作为研究对象的系统之后，才有可能准确定义问题：这个系统可能是一种机制、应用或是一般意义上的过程。一般而言，研究工作是为了更好地理解系统的运行方法，尤其是为了理解其运行规则，因为这些规则有助于我们作出预测或选择（在知情的基础上进行选择）。

问题定义这一步及产生的相关文档（可交付成果），无论是对于科研还是商业问题都很重要，因为这两项能严格保证分析过程是朝着目标结果前进的。事实上，对系统进行全面或详尽的研究有时会很复杂，一开始可能没有足够的信息。因此问题的定义，尤其是问题的规划，将唯一决定整个数据分析项目所遵循的指导方针。

定义好问题并形成文档后，接下来就可以进入数据分析的项目规划环节。该环节要弄清楚高效完成数据分析项目需要哪些专业人士和资源。因此就得考虑解决方案相关领域的一些事项。你

需要寻找各个领域的专家，安装数据分析软件。

因此，在项目规划过程中，应组建起高效的数据分析团队。一般而言，这个团队应该是跨学科的，因为从不同角度研究数据有助于解决问题。因此，一个优秀的团队必然是成功完成数据分析工作的关键因素之一。

1.4.2　数据抽取

问题定义步骤完成之后，在分析数据前，首先要做的就是获取数据。数据的选取一定要本着创建预测模型的目的，数据选取对数据分析的成功起着至关重要的作用。所采集的样本数据必须尽可能多地反映实际情况，也就是能够描述系统对来自现实刺激的反应。事实上，如果原始数据采集不当，即使数据量很大，这些数据描述的情境往往也是与现实相左或存在偏差。

因此，如果对选取不当的数据，或是对不能很好地代表系统的数据集进行数据分析，得到的模型将会偏离作为研究对象的系统。

数据的查找和检索往往要凭借一种直觉，超乎单纯的技术研究和数据抽取。它还要求对数据的内在特点和形式有细致入微的理解，而只有对问题的来源领域有丰富的经验和知识，才能做到这一点。

除了所需数据的质量和数量，另一个问题是查找和正确选择数据源（data source）。

如果工作室环境为（技术或科学）实验室，数据源生成的数据是用来做实验的。这种情况下就很容易鉴别数据源的优劣，这时唯一要注意的就是实验过程的设置。

无论是对于哪一个领域的应用，都不可能采用严格的实验方法来重建数据源所属的系统。很多领域的应用需要从周边环境搜寻数据，往往依赖于外部实验数据，甚至常通过采访或调查来收集数据。这种情况下，寻找包含数据分析所需全部信息的数据源难度很大。这时往往需要从多种数据源搜集信息，以弥补缺陷，识别矛盾之处，使数据集尽可能具有普遍性。

当你想找些数据来用时，Web是个不错的起点。但Web中的大多数数据获取起来具有一定难度。事实上，不是所有的数据都是以文件或是数据库形式存在的，有些数据以这样或那样的格式存在于HTML页面中；有的内容很明确，有的则不然。为了获取网页中的内容，人们研究出了Web抓取（Web scraping）方法，通过识别网页中特定的HTML标签采集数据。有些软件就是专门用来抓取网页的。它们找到符合条件的标签，从中抽取目标数据。查找、抽取完成后，就得到了用于数据分析的数据。

1.4.3　数据准备

在数据分析的所有步骤中，数据准备虽然看上去不太可能出问题，但事实上，这一步需要投入更多的资源和时间才能完成。数据往往来自不同的数据源，有着不同的表现形式和格式。因此，在分析数据之前，所有这些不同的数据都要处理成可用的形式。

数据准备阶段关注的是数据获取、清洗和规范化处理，以及把数据转换为优化过的，也就是准备好的形式，通常为表格形式，以便使用在规划阶段就定好的分析方法处理这些数据。