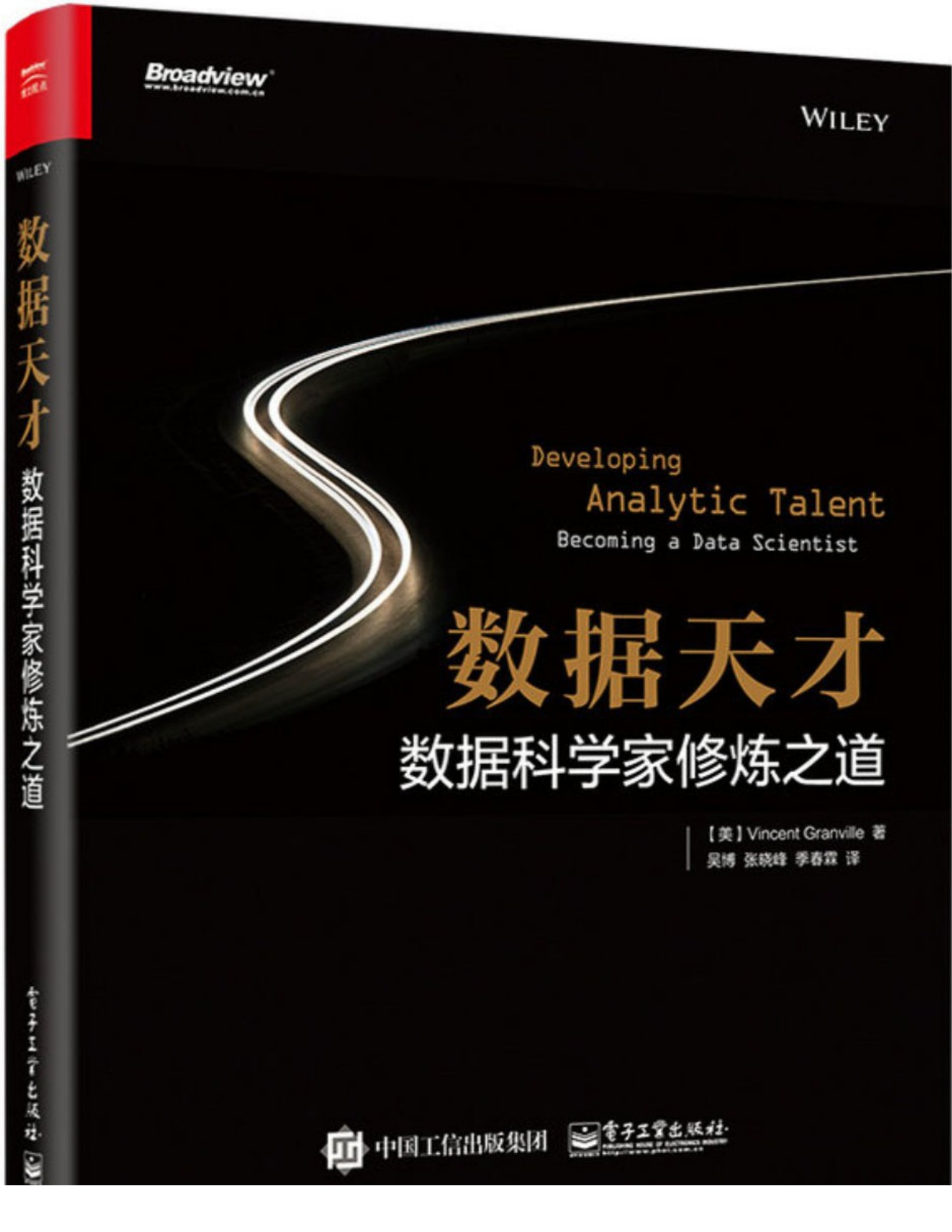
Broadview
WILEY
Developing
Analytic Talent
Becoming a Data Scientist
数据天才
数据科学家修炼之道
【美】Vincent Granville 著
吴博 张晓峰 季春霖 译
中国工信出版集团
电子工业出版社

Developing Analytic Talent

Becoming a Data Scientist

# 数据天才

## 数据科学家修炼之道

【美】Vincent Granville 著

吴博 张晓峰 季春霖 译

電子工業出版社

Publishing House of Electronics Industry

北京•BEIJING

# 目录

内容简介
专家推荐
译者序
关于作者
关于技术编辑
前言
第1章 数据科学是什么
真伪数据科学对比
伪数据科学的两个例子
新大学的面貌
数据科学家
数据科学家与数据工程师
数据科学家与统计学家
数据科学家与业务分析师
13个真实世界情景中的数据科学应用
情景1：国家对烈性酒销售的垄断结束后，DUI（酒后驾驶）逮捕量减少
情景2：数据科学与直觉
情景3：数据故障将数据变成乱码
情景4：异常空间的回归
情景5：分析与诱导在提升销量上有何不同价值
情景6：关于隐藏数据
情景7：汽油中的铅会导致高犯罪率。真的吗
情景8：波音787（梦幻客机）问题
情景9：NLP的7个棘手句子
情景10：数据科学家决定着我们所吃的食品
情景11：用较好的相关性增加亚马逊的销售量
情景12：检测Facebook上的假档案或假“喜欢”数
情景13：餐厅的分析
数据科学的历史、开拓者和现代趋势
统计学将会复兴
历史与开拓者
现代的趋势
最近的问答讨论

总结
第2章 大数据的独特性
两个大数据的问题
大数据“诅咒”
数据快速流动问题
大数据技术示例
大数据问题是数据科学所面临挑战的缩影
大规模数据集的聚类和分类
1亿行的Excel
MapReduce不能做什么
问题
3种解决方案
结论：何时使用MapReduce
沟通问题
数据科学：统计学的终结
8种最差的预测建模技术
把计算机科学、统计学和行业专业知识结合在一起
大数据生态系统
总结
第3章 成为一名数据科学家
数据科学家的主要特征
数据科学家的职能
横向与纵向数据科学家
数据科学家的类型
伪数据科学家
自学成才的数据科学家
业余数据科学家
极限数据科学家
数据科学家人群特征
数据科学方面的培训
大学课程
公司和协会培训项目
免费培训项目
数据科学家职业道路
独立顾问
创业者
总结

第4章 数据科学的技术（I）
新型指标
优化数字营销活动的指标
欺诈检测的指标
选择合适的分析工具
分析软件
可视化工具
实时产品
编程语言
可视化
用R生成数据视频
更复杂的视频
无模型的统计建模
无模型的统计建模是什么
该算法是如何工作的
源代码生成数据集
三类指标：中心性、波动性、颠簸性
中心性、波动性和颠簸性之间的关系
定义颠簸性
在Excel中计算颠簸性
使用颠簸系数
大数据的统计聚类
大数据的相关性和拟合度
一系列新的秩相关性
渐近分布与归一化
计算复杂度
计算q（n）
理论上的解决方案
结构系数
确定簇的数量
方法
例子
网络拓扑映射
安全通信：数据加密
总结
第5章 数据科学的技术（II）
数据字典

什么是数据字典
建立数据字典
隐性决策树
实现方法
示例：互联网流量打分
结论
与模型无关的置信区间
方法
分析桥第一定理
应用
源代码
随机数
解决问题的4个办法
拥有超强直觉能力的业务分析师的直观法
软件工程师的蒙特卡洛模拟法
统计学家的统计建模方法
计算机科学家的大数据方法
因果关系和相关性
怎样检测因果关系
数据科学项目的生命周期
预测模型的错误
逻辑相关回归
变量之间的相互作用
一阶近似
二阶近似
用Excel进行回归分析
实验设计
有趣的指标
把患者分成不同的人群进行治疗
私人定制的治疗
分析即服务和应用程序接口
工作原理
实施案例
关键词相关的API的源代码
其他主题
当数据库改变时，保存好数值
优化网络爬虫

哈希连接
用于模拟簇的简单源代码
Hadoop和大数据的新型合成方差
Hadoop和MapReduce的介绍
综合指标
Hadoop、数值的和统计的稳定性
方差的抽象概念
一个新的大数据定理
平移不变性的度量标准
实现：通信和计算成本
最终意见
总结
第6章 数据科学应用案例研究
股票市场
使回报率提高500%的模式
优化统计交易策略
股票交易的API：统计模型
股票交易的API：具体实现
股票市场模拟
些许数学知识
新趋势
加密
数据科学应用：隐写术
好的电子邮件加密
验证码破解
欺诈检测
点击欺诈
连续点击评分与二进制欺诈/非欺诈
数学模型与基准
虚假转化产生的偏差
一些误解
统计面临的挑战
点击评分优化关键词出价
组合优化自动快速的特征选择
特征的预测能力：交叉验证
勾连检测和僵尸网络的关联规则检测
模式检测的极值理论

数字分析
在线广告：到达率和频率的计算公式
电子邮件营销：提高300%的性能
在7天内优化关键词广告宣传活动
自动新闻提要优化
用bit.ly进行竞争情报分析
测量Twitter哈希标签（hashtag）的收益
用3个修补方法提升谷歌搜索
改进相关性的算法
广告循环问题
杂项
简单模型会获得更好的销售预测
更好的医疗欺诈检测
归因模型
预测陨石撞击
在路口停车场收集数据
数据科学的其他应用
总结
第7章 踏上你的数据科学职业之路
面试问题
关于工作经验的问题
技术问题
一般性问题
关于数据科学项目的问题
测试你自己的视觉和分析思维
通过肉眼的检测模式
识别偏差
误导性的时间序列和随机游走
从统计学家到数据科学家
数据科学家也是统计从业人员
谁应该给数据科学家教统计学
雇佣问题
数据科学家与数据架构师密切合作
谁应该参与战略思考
两种类型的统计学家
大数据与取样
数据科学家的分类

数据科学最流行的技能集合
LinkedIn上的顶级数据科学家
400个数据科学家职位头衔
薪酬调查
根据技能和位置的薪酬分类
创建自己的薪酬调查表
总结
第8章 数据科学资源
专业资源
数据集
书籍
会议与组织
网站
概念定义
职业建设资源
招聘数据科学家的公司
数据科学招聘广告的样本
简历样本
总结

Developing Analytic Talent: Becoming a Data Scientist, 978-1118810088, Vincent Granville

Copyright © 2014 by John Wiley & Sons, Inc.Indianapolis, Indiana

All rights reserved. This translation published under license.

图书在版编目（**CIP**）数据

数据天才：数据科学家修炼之道　　/（美）文森特·格兰维尔（Vincent Granville）著；吴博，张晓峰，季春霖译.　—北京：电子工业出版社，2017.5

书名原文：Developing Analytic Talent: Becoming a Data Scientist

ISBN 978-7-121-30883-3

Ⅰ.　①数…　Ⅱ.　①文…②吴…③张…④季…　Ⅲ.　①数据管理　Ⅳ.　①TP274

中国版本图书馆CIP数据核字（2017）第021480号

责任编辑：付　睿

印　　刷：

装　　订：

出版发行：电子工业出版社　北京市海淀区万寿路173信箱　邮编：100036

开　　本：787×980　1/16

印　　张：22.25

字　　数：385千字

版　　次：2017年5月第1版

印　　次：2017年5月第1次印刷

>>更多新书朋友圈免费分享微信**xueba987**

# 内容简介

这是一本跟数据科学和数据科学家有关的“手册”，它还包含传统统计学、编程或计算机科学教科书中所没有的信息。

本书有3个组成部分：一是多层次地讨论数据科学是什么，以及数据科学涉及哪些其他学科；二是数据科学的技术应用层面，包括教程和案例研究；三是给正在从业和有抱负的数据科学家介绍一些职业资源。本书中有很多职业和培训相关资源（如数据集、网络爬虫源代码、数据视频和如何编写API），所以借助本书，你现在就可以开始数据科学实践，并快速地提升你的职业水平。

本书是写给数据科学家和相关专业人士的（如业务分析师、计算机科学家、软件工程师、数据工程师和统计学家），也适合有兴趣转投大数据科学事业的人阅读。

# 专家推荐

数据科学家是商业分析、统计学和计算机科学等领域的通才，成为数据科学家正成为年轻人的新梦想。光启高等理工研究院季春霖副院长推荐我读这本他参与翻译的书之后，我一开始期望这是一本数学味、计算机味很浓的书籍。后来，完全出乎意料，这本书行文竟然如此清晰明白，原理与观点娓娓道来，并剖析了大量实际而有趣的案例，读起来丝毫没有教科书式的刻板感。通过本书，你可以了解一个数据科学家所需的完善的知识体系，包括商业分析、数据库、统计模型、蒙特卡洛模拟、机器学习、Hadoop、MapReduce、哈希连接等。本书显然为有梦想的你在通往数据科学家的道路上铺就了阶梯，我相信你只要拾阶而上，到达目的地就是确定无疑的。

——王磊　国家统计局高级统计师
中国国际经济交流中心金融学博士后
北京大学肿瘤医院核医学科客座教授

2017年大数据行业已经从上半场开始挺进下半场，数据在不知不觉中影响着我们的生产、生活、娱乐等方方面面。我们深耕在行业，深知目前国内从行业角度真正缺乏的是有着商业精神的数据科学家，本书从场景出发给我们展示了如何成为数据天才。我与吴博、晓峰、春霖交流很多，他们有深厚的学术素养，但仍实实在在地做着数据商业，恰恰这一点也是目前国内缺乏的，我一直认为在中国不缺数据技术人才，但缺乏的是真正懂商业的数据天才、数据科学家。希望大家能从本书中汲取知识，真正走向数据科学的商业之路。

——汪祥斌　DataEye创始人、CEO

数据科学家是“21世纪最性感的职位”，全球到2018年对数据科学家有上千万的职位空缺，仅中国就稀缺上百万这样的人才。这本《数据天才：数据科学家修炼之道》是成为数据科学家的必备宝典。书中对数据科学有着翔实的介绍，并针对数据科学家日常工作中所需的技能进行了

深度的剖析，辅以大量的实用案例分析，有助于快速提升大家对数据科学的理解和应用。本书势必会成为继维克托·迈尔-舍恩伯格的《大数据时代》后的又一经典大作！

——刘金玲　中国大数据产业第一媒体“36大数据”创始人

大数据是近年来媒体的热点话题，大数据时代在科学领域里的表现就是数据科学的兴起。那么人们不禁会问：什么是数据科学以及如何成为数据科学家？作者通过本书及时地为读者用一种全景式的方式给出了答案。本书以通俗易懂的语言风格和众多的真实案例，讲活了大数据与数据科学，全面而又深入浅出地阐明了数据科学的实质与内涵，揭示了数据科学家的修炼秘笈。相信不同读者一定都能从书中得到启发，了解价值，找到灵感，更好地以全新的视角审视自己的专业领域以及汲取更多的新理念、新思想。

——谌东宇　教授　深圳云数通科技有限公司总裁
前海云游数据运营（深圳）有限公司首席数据官
西南交通大学数学学院客座教授

人生的关键决策只有几个，择业就是其中之一。良好的职业决定和素质准备来自于对未来的场景有清晰而且正确的认知。吴博的这本译著，不仅能够帮助我们认识未来几十年社会、商业和技术场景中的数据行业，数据科学家的是和不是，更重要的是提供了修炼自己的宝鉴。本书横跨中美视野、结合生活事件的描述，使得我们带着轻松、开心的心情完成对数据科学的认知、体悟，让人有一种跃跃欲试和大展宏图的感觉。实在是4.0时代必备的一本书！

——郑立新　德摩资本董事长
2017年3月14日于深圳

献给我挚爱的妻子Paris、可爱的女儿Capri和儿子Zinal，感谢他们长久以来的支持。献给我怀念的父亲Roger，感谢他在我孩提时代引领我接触到数学。

# 译者序

本书最适合有志于在大数据与数据科学领域从业的人学习。格拉德威尔在《异类》一书中强调，“若要成为行业专家，离不开十万小时的刻意学习（deliberate learning）”，这跟中国俗语里“板凳要坐十年冷”有些类似。但要实现刻意学习，就不能一味依赖通识科普书籍。在大数据与数据科学领域，市面上已不缺通识性的科普书籍，唯缺这类烧脑、有专业性、适合进行刻意学习的数据科学书籍。

本书不失专业性，但也不是令人生畏的大学教材。它处处体现理论与实践的结合，还兼顾技术与商业的平衡。这要归功于原作者Vincent是学术、技术、商业三栖高手。比如书中对于星空双星的估算、陨石撞地球的建模推算，让作者在数学奥赛方面的天分展现得淋漓尽致；在垃圾邮件、水印加密、点击欺诈等案例中，作者又分享了诸多为大公司实施数据项目的经验；在方案选择、股市预测等场景中，作者更侧重商业视角，帮读者提升对数据科学方法投入/产出比及适用性的敏感度。

本书虽然专业度高，但也因为案例翔实、讲求实际，适合其他行业或领域的人士阅读。特别建议业务跟数据息息相关的企业负责人或高管，或者对数据相关项目感兴趣的投资者品读。毕竟数据科学家这一高层职位，跟企业负责人及高管的对接较多。虽说好的数据科学家，应具备与非技术人士沟通的能力，但作为数据科学家的领导，一旦多懂一些数据科学的思考模式及流程，便会对数据科学家有更多理解，也会对数据化的决策有更深的认识。

本书也传递出对行业热词的审慎态度。比如本书就对“大数据”的缘起、演变、更替、历史、迷思和幻象，着墨不少。就像书中所说，大数据领域许多看似新的方法，可以追溯到二三十年前，如今的不少创新，实乃新瓶旧酒。想必读者从Gartner的成熟度曲线里，可以看到大数据一词已渡过巅峰、渐趋理性，与之相随的，是跟数据科学息息相关的人工智能（AI）重新崛起。若理解本书的立场和价值取向，就知道人工智能

60多年来几起几落，不少如今大放异彩的方法，也可找到前身。透过现象看本质，人工智能多少因为数据体量更大、数据分析更细、计算能力更强，才成为行业焦点。忽视基础理论盲目追随人工智能热点无异于舍本逐末，认真和刻意学习数据科学及人工智能的基础理论和实践，方是正途。

正因为这本书内容如此之好，能满足读者所需，于是我痛快答应电子工业出版社付睿编辑的邀约来翻译本书。但这个小想法变成最终成品，却耗费不少人的时间和精力，对他们的感谢和亏欠不能尽录。我最要鸣谢翻译合作者光启研究院的副院长季春霖博士，还有在哈工大深圳研究生院任教的张晓峰博士，两位的研究和管理任务都很繁重，面对译书这种流程漫长、成效滞后的工作，他们展现了学界出身的坚韧素养，而在翻译校对本书的过程中，又处处体现出手不凡的专业功力。同时，也要感谢配合翻译校对本书的助手和出版社工作人员，他们对我有莫大的包容和支持。本书准备期间，也正是我的一对小孩——泰学和雅学——从孕育到出生的过程，所以要感谢我的太太熊瑛，容许我为本书挤出不少本来可以陪伴家人的时间。

最后，我还要代表季春霖博士感谢广东省自然科学杰出青年基金项目（No.S20120011253）和深圳市数据科学与建模技术重点实验室的资助。也要感谢我所在的宜远智能团队，他们在将本书中许多数据科学方法实践到医疗健康领域时，提出了诸多宝贵的翻译修正补充建议。当然，对专业内容的翻译，难在对作者见识的理解和原意的把握，所以总有力有不逮、不甚精确之处，请各位读者和专家对此海涵，提出宝贵的建议。

本书译者　吴博

# 关于作者

**Vincent Granville** 博士是一名富有远见的数据科学家，有15年大数据、预测建模、数字分析和业务分析的经验。Vincent在评分技术、欺诈检测和网络流量优化及增长等领域，是举世公认的权威专家。在过去的10年中，他曾与Visa一起研究实时信用卡欺诈检测，与CNET一起研究广告组合优化，与Microsoft（微软公司）一起研究“改变点检测”，与Wells Fargo（富国银行）一起研究在线用户体验，与InfoSpace一起研究搜索智能，与eBay一起研究自动竞价，与各大搜索引擎、广告网络和大型广告客户一起研究点击欺诈检测。Vincent也管理着LinkedIn上最大的“大数据及分析数据科学家”小组，该小组拥有超过100 000名成员。

最近，Vincent推出了数据科学中心（Data Science Center）这个大数据、业务分析和数据科学界的领先社区。Vincent曾是剑桥大学和美国国家统计科学学院的博士后。他曾入围沃顿商业计划竞赛和比利时数学奥林匹克的决赛。Vincent已经在统计期刊上发表了40篇论文，并且是许多国际会议的受邀演讲嘉宾。他还开发了一种新的数据挖掘技术，被称为隐性决策树，他还拥有多项专利，是发表数据科学书籍的第一人，并筹集了600万美元的创业启动资金。根据福布斯的排名，Vincent是大数据领域前20位有影响力的人物之一，被VentureBeat、MarketWatch和美国有线新闻网（CNN）专门报道。Vincent的Twitter账号为`@Analyticbridge`。

# 关于技术编辑

**Joni Ngai** 是一位数字化传播者，在财富500强企业任职高管，负责数字视觉开发，以及利用技术和数据来吸引大量当今互联世界里的客户。她拥有带领机构和客户在数字业务、客户关系管理（CRM）、网络媒体、分析和技术发展领域进行新尝试的丰富经验。从2000年开始，Joni在纽约的Razorfish公司开始了她的数据咨询职业生涯。此后，她在许多顶尖的数字化机构里工作过，如MRM Worldwide和Havas Digital。还有亚太地区的一些全球品牌，如Intel（英特尔公司）、Microsoft（微软公司）和P&G（宝洁公司）。她曾是I-COM China的副主席，这是一个由工业界所支持的全球数字度量论坛，该论坛通过促进在线指标度量标准化的发展，帮助相关行业成长。

Joni毕业于滑铁卢大学，主修电气工程，并辅修管理学。她还获得了西北大学凯洛格管理学院和香港科技大学的行政工商管理硕士学位。她也在香港中文大学传授新媒体理科硕士课程。

# 前言

这是一本跟数据科学和数据科学家有关的“手册”，它还包含传统统计学、编程或计算机科学教科书中所没有的信息。凭借作者在数据科学领域20多年的领导者地位，他在本书中收集了他认为对从事数据科学职业最重要的一些信息。在过去3年里，本书中的很多内容首先被发表在Data Science Central官网上，被数百万的网站用户所阅读。本书介绍了数据科学与其他相关领域的差异，以及使用大数据能给组织带来的价值。

本书有3个组成部分：一是多层次地讨论数据科学是什么，以及数据科学涉及哪些其他学科；二是数据科学的技术应用层面，包括教程和案例研究；三是给正在从业和有抱负的数据科学家介绍一些职业资源。本书中有很多职业和培训相关资源（如数据集、网络爬虫源代码、数据视频和如何编写API），所以借助本书，你现在就可以开始数据科学实践，并快速地提升你的职业水平。如果你是一位决策者，你会在本书中找到一些信息，来帮助你建立更好的分析团队，以及决定是否需要及何时需要专业的解决方案，以及哪些方案最为恰当。

## 这本书是写给谁的

这本书是写给数据科学家和相关专业人士的（如业务分析师、计算机科学家、软件工程师、数据工程师和统计学家），以及有兴趣转投大数据科学事业的人。本书也是为学习定量课程、想成为数据科学家的大学生所准备的。最后，本书也可供数据科学家的上级领导、想创建数据科学初创公司开展业务或提供数据科学咨询的人阅读。

这些读者将在本书中找到有价值的信息，特别是在以下几章中。

- 第2、4、5、6章对数据科学工作者　特别有价值，因为它们包含大数据技术内容（如聚类和分类技术），以及前沿数据科学技术，如组合特征选择、隐性决策树、分析类API、判断MapReduce何时有用等。这些章节里很多案例研究（如欺诈检测、数字分析、股票市场策略和其他更多）的说明非常详细，详细到可以让读者在实际工作中面临类似数据时，能沿用这些案例的分析方法。然而，它们的文字描述都很简单，高层管理人员不用花太多时间在细节、代码或公式上，也能阅读下来。
- 修读计算机科学、数据科学或工商管理硕士课程的学生　，会在第2、4、5、6章中找到对他们有用的信息。特别是在第2、4、5章，他们能从中找到进阶内容，如实际的数据科学方法和原则，这些在一般的教科书或典型的大学课程里都没有。第6章还介绍了现实生活应用和案例研究，并包含更深入的技术细节。
- 求职者　将会在第3章中找到有关数据科学的培训和课程资源。第7、8章为求职者提供了大量的资源，包括面试问题、简历模板、招聘广告样板，经常招聘数据科学家的公司的清单，以及薪资调查等。
- 对于想要创建一个数据科学创业公司或顾问公司的企业家　，在第3章中会找到商业计划书样板、创业公司点子和针对顾问职位的薪酬调查。同时，在本书中，数据顾问会了解如何提高数据科学工作沟通效率，掌握数据科学项目的生命周期，并得到相关书籍、会议参考和许多其他资源。
- 对于试图评估数据科学的价值和它们对企业项目的益处，以及评估MapReduce架构何时有用的高管们　　，会在第1、2、6（案例部分）、8章（招聘广告样板、简历、薪金调查）中找到有价值的信息。这些章节的重点通常不是技术。顶多会在第2章和第6章介绍一些新的分析技术。

# 这本书涵盖了什么

本书的技术部分包括数据科学的核心内容，比如：

- 将大数据和传统的算法应用到大数据时的挑战（例如在进行大数据聚类或分类时的解决方案）。

- 一种统计科学上新颖、简化、对数据科学友好的方法，重点在于它是一种健壮的无模型方法。
- 顶尖的机器学习方法（隐性决策树和组合特征选择）。
- 新型数据的新指标（综合指标、预测能力、波动系数）。
- 创建快速算法所需的计算机科学要素。
- MapReduce和Hadoop，以及Hadoop进行计算时的数值稳定性。

重点还是最新的技术。在本书中你不会找到关于旧技术的资料介绍，如线性回归（除非在引文里涉及），因为这些在经典书籍里已经讨论了很多。在本书中，对逻辑回归类的知识讨论不多。我们只是将逻辑回归与其他分类器混合，提出一种数值稳定的近似算法（近似的解决方案往往和精确模型一样有效，毕竟没有任何数据完全符合理论模型）。

除了技术，本书还提供了有用的工作资源，包括工作面试的相关问题、简历模板和招聘广告样板。本书的另一个重要组成部分是案例研究。本书的案例研究，有些带有统计或机器学习的意味，有些则跟商业或决策科学或运筹学有关，有些则关乎数据工程。大多数时候，我喜欢Data Science Central（这是个数据科学家的领先社区）上最新发表和非常热门的主题，而不是我特别重视的话题。

# 本书是如何架构的

本书由三大主题构成。

- 数据科学和大数据是什么和不是什么，以及与其他学科的区别（第1、2、3章）。
- 职业和培训资源（第3章和第8章）。
- 用作教程的技术材料（第4章和第5章，以及第2章中关于大规模数据集聚类和分类的内容，第8章中关于Hadoop的新变化和大数据的内容），以及案例研究（第6章和第7章）。

本书为潜在的和现有的数据科学家和相关专业人员（以及他们的管理者和老板）提供了宝贵的职业资源。宽泛而言，本书适用于所有处理更大、更复杂、更新、频率更快的数据的专业人士。本书还提供一些数据科学的秘诀、技巧、概念（其中许多是原创和首次公开的）、带实施

方法和技术的案例研究，以及已经在不同领域，不论是手动还是自动，能成功分析现代数据的技术。

## 阅读本书你需要什么知识

这本书包含了少量的R或Perl示例代码。你可以在`http://www.activestate.com/activeperl/downloads` 下载Perl，在`http://cran.r-project.org/bin/windows/base/` 下载R。如果你使用Windows计算机，首先需要安装一个Linux式环境：Cygwin。你可以在`http://cygwin.com/install.html` 上下载Cygwin软件。Python也是开源的，且有一个有用的、被称为Pandas的库。

如果你有一两年大学基本定量课程的知识基础，就足以理解书中大多数内容。本书不需要微积分或高等数学的相关知识——事实上，它几乎不包含任何数学公式或符号。

然而，本书也包含一些高度概括性的进阶材料。本书中的一些技术讲义，是针对那些对数学更有倾向和有兴趣深入挖掘的读者。有两年大学微积分、统计学和矩阵理论知识的读者，将能更好地理解这些技术细节。本书提供了一些源代码（R、Perl）和数据集，但本书的重点不是编码。

本书通过多种技术水平混合的介绍方式，让你不用具备高级数学知识，也有机会深度探索数据科学（这有点像Carl Sagan向主流公众介绍天文学的方式）。

## 惯例标记

为了帮助你从本书中学到最多的东西，而不是一头雾水，我们将在本书中使用惯例标记。

注意 本书中的注意、提示、交叉参考，以及对当前讨论的辅助说明，将像这个注意的方式显示。

至于文本的样式标记如下。

- 当我们介绍术语和重要的词时，我们会用楷体突出它们。
- 快捷键用这种方式表示：Ctrl+A。
- 我们在书中显示文件名、链接和代码的格式如下。
  `persistence.properties`
- 我们介绍代码的格式如下。
  对于大多数代码，我们使用`Courier New`字体，不加粗。

# 致谢

我要感谢来自Wiley的Chris Haviland和Carol Long，他们对本书的出版有很大的贡献，承担了不少风险，他们把我很多有价值、分散未经组织的在线文章，整合成一本连贯、全面和有用的书。从许多方面来看，这个复杂的过程类似于将非结构化数据转化为结构化数据，这是许多数据科学家经常面对的常规挑战，而这本书也正好提供了将非结构化数据转化为结构化数据的解决方案。同时，我要感谢我的商业伙伴和共同创始人Tim Matteson，他帮助Data Science Central这个网站成为数据科学社区的领导者，还变成了一个现代的、专注于产生价值的创业项目。最后，我要感谢我们社区的所有成员，感谢他们的评论和支持。如果没有他们的帮忙，本书也无法出版。

# 读者服务

轻松注册成为博文视点社区用户（www.broadview.com.cn），扫码直达本书页面。

- 提交勘误 ：您对书中内容的修改意见可在提交勘误 处提交，若被采纳，将获赠博文视点社区积分（在您购买电子书时，积分可用来抵扣相应金额）。
- 交流互动 ：在页面下方读者评论 处留下您的疑问或观点，与我们和其他读者一同学习交流。

页面入口：http://www.broadview.com.cn/30883

# 第1章
# 数据科学是什么

要理解某样东西是 什么，有时需要清楚地了解它不是 什么。理解数据科学也不例外。正因为数据科学这个词已被滥用，围绕大数据和数据科学又有诸多炒作，因此本章开篇要探讨什么不是数据科学。首先，我们要思考真数据科学跟伪数据科学的差异。其次，我们要了解数据科学训练是怎样从大学课程进化而来的。再次，我们会检视一些现实世界场景中应用现代数据科学的例子。

最后，我们回顾下数据科学的历史，以及它如何从计算机科学、业务优化及统计学演化成现代数据科学的过程。本章结尾是我近期参与讨论而提炼出的问答环节，描绘了数据科学家、数据架构师和业务分析师的分歧和矛盾之处。

本章提出的问题比解答的问题多，更多的解答细节将在后续章节中给出。这一做法是为了让你学会数据科学家的思维方式，抓住如今大数据行业的重点，摒弃那些过时的手法，从而明白如果对数据科学这一职业感兴趣，哪些内容不需要深究。举例来说，你需要知道统计学、计算机科学和机器学习，但并不需要对这些领域无所不知。你没必要纠结排序算法复杂度的细节（只需要了解排序算法复杂度的一般性结论），也不用知晓如何计算矩阵的广义逆，甚至连矩阵广义逆是什么（这可是统计学理论的一大核心）都不必知道，除非你对数据科学数值层面的东西很在行。

## 技术讲义

对数学和技术知识掌握很少的读者也能阅读本章。你们可以选

择跳过列在“技术讲义”里的高深内容。

交叉参考　本书提到的大部分术语可以在第8章中找到定义。

# 真伪数据科学对比

数据科学书籍、认证和文凭，如雨后春笋般层出不穷。不幸的是，许多仅仅是镜花水月：许多人钻了这一新名词的空子，将旧酒（比如统计学和R编程）放在“数据科学”这个新瓶里。

补充说一下R语言编程跟伪数据科学为何扯上了关系：R是一种有20多年历史的开源统计编程语言及编译环境，是商业化产品S+的后继者。R一直以来都局限于内存数据处理，在统计圈子里非常流行，并因其出色的可视化效果为人称道。一些新型的开发环境通过创建R程序包或者将其扩展到分布式架构里（比如将R与Hadoop结合的RHadoop），将R（限于在内存里处理数据）的能力扩大。其他程序语言当然也存在跟伪数据科学沾边的情况，比如说SAS，但不及R这么流行。说到SAS，它价格高昂，在政府机构或者实体企业的应用更为广泛。但在过去10年数据快速增长的领域（如搜索引擎、社交媒体、移动数据、协同过滤推荐等）运用不多。R跟C、Perl或者Python的语法不一样（后三者语法根源一样），其简易性使得写R的程序员比较广泛。R还有很多程序包和不错的用户界面，SAS却难学很多。

更混淆的事情是，管理决策层在搭建其数据科学家团队时，有时也不是很清楚他们想要的到底是什么。他们往往最终招募的是很纯粹的技术极客、计算机科学家，或者缺乏恰当大数据经验的人。人力资源部门对于数据科学的了解也不会好到哪里，因此导致更为严重的问题，他们给出的招聘广告就是不断重复类似的关键词：Java、Python、MapReduce、R、Hadoop和NoSQL。但数据科学真的就是这些技能的混合吗？>>更多新书朋友圈免费分享微信xueba987

MapReduce只是一个将数据分解为子集，在不同机器上分开处理，并把所有结果集合起来，从而处理大数据的泛化框架。因此它涉及的是

处理大数据的分布式框架，用到的这些服务器和设备则组成云 （*Cloud* ）。

Hadoop是MapReduce的一种实现，就像C++是面向对象编程的实现一样。NoSQL意味着“Not Only SQL（意为不单只是SQL）”，是指能更新颖、更高效地访问（比如MapReduce）数据的数据库或数据库管理系统，有时它是作为SQL（标准数据库查询语言）之下的隐藏层而存在的。

交叉参考 参见第2章，更多地了解MapReduce所不能做到的事情。

除了MapReduce以外，还有其他框架——例如，图形数据库和环境，它们依赖于节点和边这类概念，来管理和访问数据（通常是空间数据）。这些概念并不一定是新的。在谷歌存在之前，分布式体系结构已被应用到搜索技术环境中了。15年前，我写Perl脚本进行哈希连接 （一种NoSQL连接，用来连接或合并数据库中的两个表）。然而，现在一些数据库厂商提供的哈希连接，是SQL连接的一个快速替代品。哈希连接在本书后面还会讨论到。它们使用哈希表，并依赖于名称-值对 的形式。我想说的结论是，有时MapReduce、NoSQL、Hadoop、Python（一种优秀的脚本语言，往往用于处理文本和非结构化数据）会被认为是Perl的后继者，但它们根源于几十年前就开发出的系统和技术，只是在过去的10年里变得更成熟而已，但数据科学并不只是这些。

事实上，你可以成为一名真正的数据科学家，且不需要掌握这些技能。NoSQL和MapReduce不是新概念——在这些关键词被创建之前，就有很多人接触到它们。但要成为一名数据科学家，你需要以下能力。

- 敏锐的商业头脑。
- 真正的大数据专业知识（例如，可以在几个小时内快速地处理一个5 000万行的数据集）。
- 认知数据的能力。
- 对模型具有猜疑精神。
- 了解大数据“诅咒”。
- 有能力沟通并理解管理人员正在试图解决哪些问题。
- 能正确评估付你工资所能带来的回报（ROI）或效益提升（lift）。
- 能够快速地识别一个简单的、健壮的、可扩展的解决方案。

- 能够说服并推动管理人员，即使不情愿，也要为了公司、用户和股东的利益，转到正确的方向上。
- 真正热爱数据分析。
- 成功案例的实际应用经验。
- 数据架构知识。
- 数据收集和清理技能。
- 计算复杂度的基础知识——如何开发健壮的、高效的、可扩展的、可移植的架构。
- 良好的算法知识。

数据科学家在商业分析、统计学和计算机科学等领域也是通才，比如会掌握这些专业知识：健壮性、实验设计、算法复杂度、仪表盘和数据可视化。一些数据科学家也是数据策略师——他们可以开发数据收集策略，并使用数据来发现可操作的、能对商业产生影响的见解。这就要求数据科学家具有创造性，能根据业务要求，制定分析、提出解决方案。

要理解数据科学，所需的基本数学知识包括：

- 代数，如果可能的话，包括基本矩阵理论。
- 微积分入门课程。要掌握的理论不多，只需要理解计算的复杂度和 *O* 标记法即可。了解特殊函数，包括对数、指数、幂函数。微分方程、积分和复数不是必要的。
- 统计与概率的入门课程，要了解随机变量、概率、均值、方差、百分位数、实验设计、交叉验证、拟合度和稳健统计的概念（不需要了解技术细节，而是达到本书介绍的程度即可）。

从技术的角度，要掌握的重要技能和知识有R、Python（或Perl）、Excel、SQL、图形（可视化）、FTP、基本的UNIX命令（sort、grep、head、tail、管道和重定向操作符、cat、cron定时等），以及对如何设计和访问数据库有基本了解。了解分布式系统如何工作和在哪里能发现瓶颈（是在硬盘和内存之间的数据传输，还是在互联网上），这也很重要。最后，要了解网络爬虫基本知识，这有助于获取互联网上能找到的非结构化数据。

# 伪数据科学的两个例子

这里有两个伪数据科学的例子，说明数据科学家要开展工作，为什么需要遵循标准和最佳实践。这里讨论的例子，并不是说它们是不好的产品——实际上，它们有很多的内在价值——但跟真的数据科学无关。原因有以下两点：

- 首先，部分统计学家没有参与到大数据革命中。虽然有些人还写了关于应用数据科学的书，但只是重新包装原来的统计课程。
- 第二，适合大数据集的方法需要变化——按2005年的大数据定义，当有2　000万行数据时，才有资格成为大数据——但2010年后不能再这样划分，因为大数据至少是TB级的。

不少人认为数据科学是统计学的一个新名字，他们把数据科学和伪数据科学，以及2005年的大数据定义与2013年的大数据定义弄混淆了。现代数据也有很大的不同，已经被描述为3个V：速度（velocity）（实时、快速流动）、多样化（variety）（结构化、非结构化，如微博内容）和大数据量（volume）。我会增加真实性（veracity）和价值（value）。要了解详细信息，请阅读第2章对数据流动速度超过处理速度的讨论。

交叉参考　更多关于统计学家与数据科学家的对比，请参阅第4章。

## 例子**1**：某一本数据科学电子书

查看一所著名大学2012年的数据科学训练手册，书的大部分内容是关于旧的统计理论。这本书用R来说明各种概念。当用逻辑回归处理仅仅1万行数据时，并不是大数据科学，它是伪数据科学。这本电子书全部关于小 数据，最后几章例外，在那里你会学到一点SQL（嵌入在R代码中）的知识和如何使用R包从Twitter提取微博内容，且创造出作者所说的词云（它和云计算没关系）。

即使提取Twitter内容的项目也还是小数据，也没有分布式体系结构（例如，MapReduce）。事实上，该书从来没有谈到数据架构。该书的

水平是初级的。每章以简单的英语（适合高中学生阅读）简短介绍大数据和数据科学，跟该书实际涉猎的小数据科学不搭，跟项目和技术展示脱节。

也许作者增加了这些简短的段落，是为了可以将他的“R统计”的电子书，重新改名为“数据科学的介绍”。它是免费的、很好的、写得很棒的书，且会使高中学生对统计和编程感兴趣。但它与数据科学无关。

### 例子**2**：数据科学证书

我们讨论一下美国受人尊敬的公立大学提供的数据科学证书。该证书课程咨询委员会主要是高级技术人员，大多数人有学术职位。数据科学家被介绍为“一种新型的数据分析师”，但我不赞同。当你查看招聘广告时，你会发现数据分析师包括统计人员和平均工资收入低的其他人，主要是因为这些职位不是高级职位。数据科学家不是初级职位。

这所大学的课程有很强的数据架构和计算机科学背景，它的计算机科学课程也非常好。这是数据科学的一个重要组成部分，但它只涵盖了数据科学的三分之一。它也有一些陈旧的统计学内容，以及跟健壮性和其他统计主题相关的课程，但没有涉及对数据科学家很有用的那几个主题（例如，六西格玛、近似解、80/20规则、交叉验证、实验设计、现代模式识别、效益提升力指标、第三方数据、蒙特卡洛模拟或数据科学项目的生命周期）。修读该课程需要Java和Python作为先修课。因此该课程价格昂贵——需要几千美元。

那么数据科学剩下的三分之二内容是什么呢？领域知识（一个或两个领域）占到三分之一。最后的三分之一是应用统计、敏锐的商业头脑、有能力与决策者沟通和决策，以及远见力和领导力的结合。你不需要全盘知晓六西格玛、统计学或运筹学，但熟悉这些领域中一些有用的概念，对你将是有帮助的，并且当这些领域有新问题出现时，你可以快速找到专门的信息。也许有一天你会从事时间序列数据或计量经济学模型方面的工作（在微软时，我意外地也从事了这方面的工作）。现在对时间序列只有一点了解也是可以的，但作为一名数据科学家，你应该能够确定恰当的工具和模型，当接触到新类型数据时，学习过程应该是非常快速的。知道时间序列　是必要的，在面对一个新问题时，可以判定应用时间序列建模是一个好的选择还是糟糕的选择。但是你不需要成为时间序列、六西格玛、蒙特卡洛模拟、计算复杂度或逻辑回归等方面的专家　。即使突然接触到时间序列，你也不需要去学习所有的东西，但

是你必须通过快速的在线搜索，找出解决方法，这是很重要的关键技能，所有的数据科学家都应该具备。在这种情况下（比如遇到时间序列问题时），如果需要的话，则可以了解它的相关图、趋势、变化点、标准化和周期性。在第4章的三类指标那一节描述了这些主题：中心性、波动性、颠簸性。

# 新大学的面貌

请允许我与你们分享两个故事，说明当今有抱负的数据科学家面临的一个重大问题。我最近读到一个副教授的故事，他给一个班级上课的薪酬是2 000美元，但根据该课程的学费和学生的数量，这所大学在该课程上赚了50 000美元。那么其余48 000美元的利润去哪儿了呢？

我的妻子申请了一个为期一年的研究生课程，花费22 000美元。然后她收到一封信，说给她发放35 000美元的贷款，以支付该课程。但是，如果她需要一笔贷款来支付这个课程，她一开始就不会继续该课程了。

我分享这两个故事的原因是要指出，一般的美国研究生和本科课程的高额费用通常是由贷款来支撑的，这就是美国学生面临的债务危机。传统的高校收取这么高的学费，是为了支付同样高的费用，包括工资、设备、运营和达到政府监管要求。因此，传统的大学面临越来越多的竞争者，更现代、更短、在线点播的替代课程，成本要少得多 。

注意 并不是所有国家都和美国一样，有高成本的教育。对于一个美国学生来说，在中国或欧洲修读一门数据科学课程，可以最大限度地降低成本，同时还可以学到高质量的内容，积累国际经验，这对美国学生来说也是有意义的。

既然我们批判数据科学的一些传统课程的授课方式和美国传统大学的教育成本，那么现在让我们看一看数据科学高等教育的未来。

正确的培训是根本，因为这关乎你如何成为一名优秀的、合格的数据科学家。许多新的数据科学课程是在线提供的（如在Coursera.com上的那些课程），或通过企业（而不是大学）类似地共享提供，如在线提

供或按需提供。新的数据科学“大学”，总结而言有以下特点。

新数据科学课程有如下特点。

- 花更少的时间来学习，只要6个月，而不是几年。
- 按需求或在线提供课程和资料。
- 专注于运用现代技术。
- 去掉陈旧内容（微分方程或特征值）。
- 培训材料里包括经验法则、交易技巧、技术、真正的实施方法、实际建议。
- 教学成本很低甚至没有，所以学生没有必要大量借贷。
- 有时是由公司或具有前瞻性思维的大学赞助或组织的（内容应该中立）。
- 不再包括知识孤岛（例如，运筹学、统计学与商业分析相互隔离）。
- 需要参与实际的、真实世界里的项目（鼓励进行合作），而不是为了通过考试。
- 包括高度紧凑的，以及深度总结的培训材料，必要时还部分使用免费的在线资源。
- 用学徒制取代博士课程。
- 提供实质性的帮助，帮助学生找到不错、薪酬高的相关工作（要交费用，并成功完成所需的课程；如果是由公司资助的课程，则不需要费用：因为公司已经聘请你，或即将雇用你）。
- 开放给每个人，不管学生以前的教育、语言、年龄、移民身份、财富或居住的国家。
- 甚至比现有的传统课程更为严格。
- 不是照搬书本内容，能减少作弊或抄袭的问题。
- 课程资料经常更新：不断增加新的发现和新方法。
- 课程材料的架构专注于垂直行业（例如，金融服务、新媒体/社交媒体/广告），因为要识别和理解现实世界的问题，特定的行业知识是重要的，并且在上岗时，能够快速上手新的工作（即没有学习曲线）。

同样，新的数据科学“教授”有以下特点。

- 不是终身的教授，但也不是兼职的。

- 在许多情况下，没有受雇于传统的大学。
- 是跨学科的专家，不断地适应变化，并确实为课程和行业带来了有意义的变化。
- 与行业领导者有着很好的联系。
- 被高度尊重和众所周知。
- 有在企业获得的经验，或自己独立获得的经验（担任顾问，擅长发布现代化的数字内容）。
- 他们在网上发布研究成果和其他资料，这是一种比传统杂志更快地推动科学发展的途径。
- 不必花大部分时间写经费申报书，而是侧重于应用及传授科学。
- 很少面对任何官僚机构。
- 在一些情况下，可以在家里工作，消除了博士夫妇所面临的工作地点不同的问题。
- 研究活动很自由，虽然可能倾向于可以赚到高收入、利润丰厚的项目。
- 积累开放共享的知识而不是专利，并广泛传播这方面的知识。
- 在某些情况下，更直接贴近市场。
- 比传统的终身教授赚更多的钱。
- 可能没有博士学位。

交叉参考　第3章包含数据科学学位和培训课程的具体信息。

# 数据科学家

数据科学家在工业、政府和其他组织中具有独特的作用。这个角色不同于统计人员、业务分析师或数据工程师等岗位。下面的章节将讨论他们的差异。

# 数据科学家与数据工程师

数据科学家和数据工程师的主要区别，可以用ETL和DAD的区别来解释。

- ETL（提取/变换/载入）是数据工程师，还有数据架构师或数据库管理员（DBA）的职责。
- DAD（发现/获取/提炼）是数据科学家的职责。

数据工程师往往把重点放在软件工程、数据库设计、生产环境代码上，并确保数据在来源（在那里它被收集）和目的地之间平稳地流动（在目的地那里通过统计摘要提取和处理，或通过数据科学算法产生结果，并最终转移到源或其他地方）。数据科学家却需要了解数据的流动和如何优化（尤其是使用Hadoop时）。实际上不是优化数据流本身，而是数据处理的步骤，从数据中提取出价值。数据科学家与工程师和商业人员一起工作，定义指标、设计数据收集方案，并确保数据科学流程与企业数据系统（存储、数据流）有效地集成。对于在小公司工作的数据科学家，尤其如此，这也是数据科学家应该能编写可被工程师重复使用的代码的原因。

有时数据工程师也操作DAD，有时数据科学家也操作ETL，但这并不常见，且他们通常是在公司内部才这么做。例如，数据工程师可以做一些统计分析，以优化一些数据库流程，而数据科学家可以做一些数据库管理，以管理一个保存汇总信息的、小型的、本地的私有数据库。

DAD包含以下内容。

- 发现　：识别良好的数据源和指标。有时（与数据工程师和业务分析师一起工作），对应该创建的数据提出需求。
- 获取　：获取数据，有时通过API、网络爬虫、互联网下载或数据库，有时是从内存数据库获取数据。
- 提炼　：从数据中提取信息，做出决策，增加投资回报率，并采取行动（比如，在自动投标系统中，确定最佳的投标价格）。它包括以下内容。

- 通过创建数据字典和进行探索性分析，对数据进行探索。
- 清洗数据杂质。
- 通过数据汇总进一步提炼数据，有时是通过多层汇总或分层汇总来

实现的。

- 对数据进行统计分析（有时会采取像实验设计这样的做法，所以在前面的“获取”阶段也可以进行），自动和手动都可以。可能需要设计统计模型，也可能不需要。
- 在某些自动过程中呈现结果或集成结果。

数据科学是计算机科学、商业工程、统计学、数据挖掘、机器学习、运筹学、六西格玛、自动化和行业知识的交叉点。这些不同的领域，加上业务的愿景和行动，汇集了一系列的技术、流程和方法。数据科学是连接不同组件的过程，有助于业务优化，并消除那些降低业务效率的孤岛。它也有自己独特的核心，（例如）包括以下主题。

- 高级可视化。
- 分析即服务（AaaS）和API服务。
- 大规模数据集的聚类与分类。
- 计算大数据的相关性和R平方。
- 任何数据库、SQL或NoSQL都应该有的11个特征。
- 快速特征选择。
- Hadoop/Map-Reduce。
- 网络拓扑。
- 大数据中关键词的相关性。
- 在常规空间、超平面、球面上或单纯形上的线性回归。
- 无模型置信区间。
- 特征的预测能力。
- 无模型的统计建模。
- 大数据“诅咒”。
- MapReduce不能做的事。

请记住，一些雇主寻找的是具备丰富统计知识的Java或数据库开发人员。这些开发人员是非常罕见的，所以雇主反而有时会试图雇用数据科学家，但希望他在开发生产代码方面很优秀。所以职位到底是需要具备统计知识的Java开发者，还是具有较强的Java技术的统计学家，你应该事先询问雇主（如果可能的话，在电话面试的过程中去询问）。然而，有时招聘经理不知道他真正想要的是什么，如果你告诉他你的专业知识所带来的附加价值，你可能会说服他雇用你，哪怕欠缺他们预期的某些技能。对于雇主，招聘Java软件工程师，再要他们学习统计学，比反过来更容易。

# 数据科学家与统计学家

许多统计学家认为，数据科学是关于分析数据的，但它远不止分析数据。数据科学还涉及实现算法、自动处理数据，并提供自动化的预测和行动，比如：

- 分析美国宇航局的照片，以寻找新的行星或小行星。
- 自动报价系统。
- 自动驾驶（飞机和汽车）。
- 在Amazon.com或Facebook上推荐书和朋友。
- 针对所有酒店客房的客户定制（实时）定价系统。
- 在计算化学中模拟用于癌症治疗的新分子。
- 流行病的早期检测。
- （实时）估计美国所有房屋的价值（Zillow.com）。
- 高频交易。
- 为用户和网页匹配谷歌广告，以最大限度地提高转化率。
- 为所有谷歌搜索返回高度相关的结果。
- 信用卡交易评分（欺诈检测）。
- 税收欺诈检测与恐怖行为检测。
- 天气预报。

所有这些都涉及统计科学和TB级的数据。大多数做这些类型的项目的人，并不称自己为统计学家。他们称自己为数据科学家。

几个世纪以来，统计学家一直在收集数据并进行线性回归分析。300年前、20年前、现在或在2015年由统计学家 进行的DAD，跟如今数据科学家 进行的DAD大相径庭。关键是，最终随着越来越多的统计学家学会这些新技术，而更多的数据科学家学会统计科学知识（采样、实验设计、置信区间——不只是第5章中描述的那些），数据科学家和统计学家之间的边界将变得更加模糊。事实上，我可以看到另一种数据科学家正在兴起：具备深厚统计知识的数据科学家。

数据科学家和计算机科学家的区别是，数据科学家有更强大的统计学背景，特别是在计算统计方面，但有时则是在实验设计、采样和蒙特卡洛模拟方面。

# 数据科学家与业务分析师

业务分析师专注于数据库设计（高层次的数据库建模，包括指标定义、仪表盘设计、检索和生成管理报告，并设计报警系统）、评估各种业务项目和支出的投资回报率及预算问题。一些业务分析师从事营销或财务的规划、优化和风险管理工作。许多业务分析师从事高层次的项目管理工作，直接向公司主管汇报。

有些任务也由数据科学家完成，特别是在较小的公司：指标的创建和定义、高层次的数据库设计（收集什么样的数据以及如何收集），或计算营销，甚至增长黑客>>更多新书朋友圈免费分享微信xueba987（最近创造的一个词，来形容一种角色，它能够快速增加公司互联网访问流量，扩大公司或产品的影响力，这种工作涉及较强的工程和分析技能）。

数据科学家让业务分析师有提升空间，例如，帮助将报表制作过程自动化，使数据提取速度更快。数据科学家还可以教给业务分析师FTP和基础的UNIX指令知识：ls -l、rm -i、head、tail、cat、cp、mv、sort、grep、uniq -c及管道和重定向操作符（|, >）。然后，可以在数据库服务器上编程或安装一段代码（业务分析师通常是通过浏览器或工具，如Toad或Brio访问服务器），以检索数据。所有的业务分析师都必须做的是：

1. 创建SQL查询（甚至要有可视化工具），并保存为SQL文本文件。

2. 将它上传到服务器并运行程序（例如，一个Python脚本，读取SQL文件并执行它，检索数据，并将结果存储在CSV文件中）。

3. 输出结果（CSV文件）到业务分析师的电脑中，做进一步分析。

对于业务分析师和 数据科学家，这样的合作是双赢的。在实践中，这样的合作已经帮助业务分析师提取的数据比他们过去所使用的数据要大100倍，而且提取速度快10倍。

总之，数据科学家并不是业务分析师，但前者可以极大地帮助后

者，包括帮助业务分析师将任务自动化。此外，如果数据科学家能带来额外的价值，具备额外的经验，他更容易找到工作，特别是在只有一个职位预算的公司，当雇主不能确定是雇用业务分析师（开展整体分析和数据任务）还是数据科学家（有商业头脑，可以执行一些通常分配给业务分析师的任务）时。在一般情况下，业务分析师是优先聘请的，但如果数据和算法变得过于复杂，也会雇用数据科学家。如果你创建自己的创业公司，你需要扮演两个角色：数据科学家和业务分析师。

# 13个真实世界情景中的数据科学应用

现在让我们看看13个在真实世界情景下的例子，了解现代数据科学家可以帮助我们做些什么。这些例子将有助于你学习如何专注于一个问题和如何形式化一个问题，以及如何仔细评估所有潜在问题——总之，是学习数据科学家在提出解决方案之前，如何定位问题和进行战略性思考。你也会看到为什么一些广泛使用的技术，如标准回归，可能并不适合所有情况。

注意　本章重点是问题本身，以及如何评估它们。第4章、第5章将讨论这些问题的解决方法。

数据科学家　的思维方式不同于工程师、运筹学专业人士、计算机科学家。虽然运筹学涉及很多分析，但这一领域的焦点是具体业务优化层面，如库存管理和质量控制。运筹学涉及国防、经济、工程、军事等领域。它采用马尔可夫模型、蒙特卡洛模拟、排队论、随机过程，以及（由于历史原因）Matlab和Informatica工具。

交叉参考　参阅第4章，数据科学家与业务分析师、统计学家和数据工程师的对比。

数据科学会遇到两种基本类型的问题。

1. 内部数据科学问题，　如损坏的数据、粗心的分析或使用不恰当的技术。内部问题不是针对业务而言，而是针对数据科学社区。因此，解决办法包括训练数据科学家更好地工作，要他们遵循最佳做法。

2. 业务应用问题 是现实世界中的问题，因此要寻求解决方案，如欺诈检测，或识别一个因素是原因还是结果。这些可能涉及内部或外部（第三方）数据。

# 情景1：国家对烈性酒销售的垄断结束后，DUI（酒后驾驶）逮捕量减少

最近发表在*MyNorthWest* 报纸上的一篇文章，说到一年前在华盛顿州实施的一条新法律，允许杂货店销售烈性酒类。这里的问题是如何评估和解释在法律生效后，酒后驾车者的逮捕量下降。

作为一个数据科学家，你首先需要制定一个可能解释逮捕量下降的原因列表（通过与客户或老板的讨论）。然后设计一个方案，排除其中的一些原因，或为每个原因附上正确的权重，或简化结论，判定这个问题除非有更多的数据或信息可以使用，否则没法回答。

以下是15个可能原因，以及跟酒后驾车逮捕率报告存在悖论的问题。你当然可以想出更多的理由。

- 在数据收集过程中有差错（数据是错误的）。
- 这篇文章的作者，可能与别人有利益冲突、想推销特定的观点，或是出于政治动机。也许只是一个大胆的谎言。
- 之所以酒后驾驶被逮捕的人数变少，也许是因为警察的人手少了。
- 在这一时期内，其他犯罪率也有所下降，这是犯罪率总体下降趋势的一部分。若没有这条新法律，酒后驾驶的逮捕量会不会下降更多？
- 缺乏统计显著性。
- 存在其他更为严厉的刑罚，阻止了酒后驾车行为。
- 年纪大的酒客多，因为他们已经肇事死亡，所以逮捕量下降。
- 总人口增加，但喝酒的人数减少，因为移民率高的是中国人和印度人，但他们中喝酒的人数比例远远低于其他人口群体。
- 酒后驾车逮捕量统计口径是针对华盛顿居民，还是非居民的？
- 这一法律没有效果，在此之前，人们也可以从华盛顿的杂货店买酒

（除了烈性酒）。

- （可能是因为增加了税收），酒价升高，导致酒精消费减少（尽管酒精和烟草消费对价格弹性变化有抵抗力）。
- 人们现在开车到更近的地方就能买到烈性酒，因此，喝烈性酒驾驶而被逮捕的人变少了。
- 所有酒后驾驶逮捕量都下降，还是仅仅喝烈性酒的酒后驾驶逮捕量下降才如此？
- 一般来说，如果人们开车次数减少了，无论对饮酒者还是非饮酒者来说，那么可能的原因就是汽油价格上涨了。
- 要评估这个新法律的影响，有一个更好的指标是华盛顿居民的酒类（特别是烈性酒）的总消费量。

数据科学家必须选择正确的方法来评估新法律的影响，并想出如何获得评估所需数据的方法。实际上，逮捕量减少的真正原因是，喝烈性酒者可以开车在很短的距离内就能买到烈性酒。对华盛顿州来说，问题在于判断该法律能否减少因酒精消费而产生的社会问题代价（是否会增加酒精销售的税收、裁减州内商店的员工，使酒精相关的犯罪率维持适度水平或完全消失等）。

# 情景2：数据科学与直觉

在管理层圈子里，许多决定仍然依靠直觉。然而，正如这个例子所示，数据科学和统计分析结果优于直觉，让你不至于凭直觉而形成错误的结论。

双数据点是指几乎相同的观测值。在许多不同类型的数据集中，这种情况经常出现，而非特例。在任何有300多行数据的二维或三维的数据集中，如果数据是定量的，并均匀地分布在一个有界空间中，你应该会看到大比例（大于15%）的数据点都有一个近邻。

这适用于所有数据集，但这一发现首先是从星系照片里发现的。星系照片里有很多双星，你可能会直觉地认为，有一些内在原因使得星星成对地聚集在一起。但是，如果你知道纯概率，你会发现，即使没有任何外部机制推动，15%的星星也会属于双星系统。

例如，我们可以考虑一个包含500颗星星的星系。计算这些星星构成双星系统的纯概率，结果如下。

- 至少有60颗（12%）星星属于双星系统的概率为85%。
- 至少有80颗（16%）星星属于双星系统的概率为18%。
- 至少有100颗（20%）星星属于双星系统的概率（几乎）为0%。

然而，在现实中，超过80%的星星都属于双星系统。概率统计不支持这个数字；因此，很明显有一些机制使得星星成对聚集。

该问题提供了一个很好的、可以测试你分析直觉的机会。看看图表里的双生观测值，然后目测评估下双生观测值是自然的（随机）还是非自然的（数量太多或太少）。可以写一段代码当作练习（Python、Perl、R代码），进行模拟（包括更复杂的三维情况），检查下理论结果，并比较一下R、Perl、Python的速度差异。

交叉参考　第6章“数据科学应用案例研究”，包含了额外的测试，可以用来评估你的分析直觉。

技术讲义

本节讲义解释如何计算概率。如果你不是一个数学家，可以跳过本节。

假设，有一张10cm×10cm的夜空图，其有大约n=500个可见的星星（数据点），而双星定义是一颗星星和邻近的星星距离是1mm（或更短）。如果恒星是完美随机分布的，则500颗恒星中，位于双星系统中的星星数量将是73颗（平均而言）。这个数字远远高于大多数人所认为的数字。这一比例可以表示为p，因此p=14.5%，在500颗恒星中，预计位于双星系统中的星星数量为n*p=73。

在这种情况下，可以用Poisson过程的理论来计算该概率。如果500个点分布在一个100mm×100mm的区域（一个放大的夜空图），则Poisson过程的强度L是每平方毫米面积内点的数量，即L=500/

（100mm×100mm）=0.05/$mm^2$ 。

在1 mm距离之内一颗星星至少有一个邻近点的概率p为1−Proba（0个邻近点）=1−exp（−L*Pi*$r^2$ ），其中r=1mm，Pi=3.14。这里Pi*$r^2$ 是一个半径为1mm的圆的面积。指数项是一个Poisson过程，在一个给定的集合里（圆、矩形等），点的数目有一个均值为L*Area的Poisson分布。因此p=0.145。

所以是否属于双星，是一个参数p=0.145的伯努利（1/0）变量。V可以表示位于双星系统中的星星数目：V是参数p的n个伯努利变量的总和，因而是参数为n、p的二项式分布，而normal（0,1）分布跟标准化变量Z=（V-np）/SQRT{np（1-p）}的值非常近似。这个事实可以用来计算前面所说的概率。

替代计算： 使用蒙特卡洛模拟，而不是使用理论模型，来计算这些概率，也可以得到相同的结果。这将涉及生成100万个模拟图像（二维数据表），并在每个模拟图像里，计算属于双星系统的星星数目。使用现代计算机、良好的随机数生成器和智能算法，这项任务可以在几分钟内自动化进行。

如果你使用朴素的做法，则该模拟过程可能是缓慢的。但计算n个距离，找最近的星星时，计算复杂度可以做得比O（$n^2$ ）好得多。具体做法是将数据存储在一个间隔为1mm的网格中（即，一个有100×100=10 000个单元格的二维数组）。因此，对于每一颗星星，你只需查看周围的8个像素，来计算不到1mm距离的邻近点的数目。通过使用1万个1比特（代表存在/不存在）存储单元，可以把O（$n^2$ ）的复杂度降低到O（n）。

请注意，选100万或1万是随意的，实际只需足够大的数字，保证所估计的结果足够稳定即可，数字若再大则不会带来更准确的结果。选择合适的样本和样本大小是一个实验设计问题，并且使用无模型的置信区间有利于这项任务的开展，也保证结果稳健。这种蒙特卡洛模拟方法较受运筹学专业人士的青睐，一些数据科学家、计算机科学家、喜欢无模型统计建模的软件工程师们也较钟爱。然而，在这个案例里，理论模型众所周知，也很简单（即使不是很基础），也能快速而简捷地得到答案。所以，除非你必须花很多时间才能理解它的原理，否则还是建议寻找理论解决方案。

附加说明：　在这个例子中，星星显示在二维屏幕上。在现实中，它们存在于三维空间中。从X坐标和Y坐标来看两颗星星可能是邻近的，但在Z轴上它们可能是永久分开的。所以要计算出双星的真实比例期望值，你就得模拟三维空间里的星星（点），投影到矩形内计算出双星的比例。在这种情况下，我不确定该理论模型是否是简单的解决方案，但蒙特卡洛模拟仍然是简单的。在实践中，实际上遥远的恒星不够明亮而不能显示在图片上，所以二维模型确实是对真实的三维问题很好的近似。

此外，在理论模型中，关于星星的位置有一些隐含的独立性的假设（当提到二项式模型时），但这不是实际情况，因为星星周围1mm直径范围内有时会重叠。但这种近似依然不错，并且是保守估计的。而理论上的数字，如果排除重叠，肯定要高于73。

# 情景3：数据故障将数据变成乱码

在数据导入或导出的过程中，有许多出差错的例子。这通常是架构较差的结果，或者使用了错误的工具，或者盲目依赖工具而不进行数据质量检查的原因。数据科学家如何检测和解决这个问题？

Microsoft Access有时不能正确导入或导出数据类型，使数据库表自动甚至是强迫类型转换，输出文本或CSV文件（或其他格式），结果较差，容易导致许多错误。这种情况发生，是因为环境不够灵活，从而不能进行智能的类型检测并进行类型转换。例如，你是否处理过以下情况，一行日期数据被记录为整数，另一行却被记录为mmddyy格式（月、日、年格式，比如2016年10月5日记为100516）？或者是你不得不处理数据字段值内的逗号，因为这会搞砸你的CSV文件？

我收到过许多数据文件，损坏率达到5%以上。有时两种不同类型的值（广告主关键词与用户查询，或来源域名与广告网络域名）存储在同一字段中。有时字段没有正确对齐。有时看起来制作数据的人不是使用正则表达式的专家，或是总是错误地使用分隔符。有时则是因为软件把事情搞糟了。

有时这些差错有不良的副作用，比如你的数据库里有一个非真实的用户ID，成为所有没有ID的用户的垃圾收集器 ，导致分析数据时有缺陷。但在其他时候不是个大问题。

以下几个例子是讲混乱的数据、混乱的格式和不完整的数据整合如何导致严重的错误，以及如何解决这些问题。

- *Wells Fargo* （美国富国银行）： IBM Tealeaf用户的会话被分解成多个会话，因为每个服务器都有自己的网络日志，而这些日志并没有被正确混合。这种情况下主要是架构设计造成的问题。幸运的是，发现和解决这个问题能帮助我们形成对该类问题的洞察力。
- *eBay* （易贝）： 数据传输过程中把法语和德语关键词的特殊字符删除了，造成了数据不一致性。通过建立临时的外键查找表，让正确的和错误的拼写对应，从而可以解决数据不一致的问题。
- 点击欺诈检测： 关键词字段有时代表用户查询（如谷歌查询），有时是一个预先指定的关键词类别，这取决于广告网络联盟公司的种类。如果相关公司的关键词是属于关键词类别的，我们会施加一个较大的错误惩罚值，这是因为（出于设计而非欺诈的原因）这种关键词的分布很差。添加一个指定关键词类型的新字段，能够解决该类问题。

作为练习，我希望你多想想这些类型的问题，并回答以下问题。以下问题都是常见的、有趣的工作面试问题。

- 你如何处理这些问题？
- 如何自动检测这些差错？该问题是质量保证（QA）问题吗？
- 在大数据情景下这类问题是变得更糟了吗？检测问题和解决问题，哪个比较困难？
- 能够容许多少坏数据？能否容许1%的错误？欺诈检测的话能容许的错误甚至更少吗？
- 解决这些问题，你花费了多长时间？
- 如何设计智能的类型转换？

# 情景**4**：异常空间的回归

这个例子说明了在使用传统技术解决新问题之前，需要改进原有技术，如回归。这些技术有些已经超过了100年，是统计分析领域的主要工具。针对小的数据集，已经有简单的数学方案能解决问题。随着大数据和大规模并行计算能力的出现，情况已经不同。

比如，你想通过逆向工程得到可口可乐的配方。基于许多人的品尝结果（根据实验设计），响应值Y衡量你的配方跟可口可乐真正配方的接近程度。事实上，这非常类似于在临床试验中，通过测试混合的原子或化学基（每个组合产生一个独特的分子）来改善药物效果。而自变量是二进制值，每一个代表一个成分，如盐、水、玉米糖浆等。如果所讨论的成分在配方中是存在的，则该值等于1，否则等于0。这与标准的线性或逻辑回归有很大的差异。

技术讲义

回归系数a_k （k = 1,..., m）必须满足以下要求。

- 每个k都是正数（a_k大于或等于0）。
- 这些系数的总和等于1。

总之，你在单纯形空间进行回归，其中a_k代表混合的比例。这种回归有一个有趣的特性，a_k系数的平方和等于由SUM（a_k）=1定义的m-1面体面积的平方，且a_k大于或等于0。（这是毕达哥拉斯定理的一个泛化。）这有点像一个套索回归、岭回归或逻辑（是logic而不是logistic）回归，但放在贝叶斯框架里，也可以利用马尔可夫蒙特卡洛（MCMC）来解决。

如何解决在一个球面上的回归？例如：

- 根据鸟的视线，找出加拿大大雁的迁徙飞行路径。
- 针对一颗在行星上撞击出多个排一起的陨石坑的流星，计算其轨迹和起源。

- 基于火山的位置探测地质断层（两者几乎是一一对应的，比如在喀斯喀特山脉内，或在大西洋中部的海底）。
- 答案是通过将球面映射到平面上，通过平面回归来解决问题。

在这个例子中，目的是创造一个竞争产品，口味和可口可乐一样，用不同的名字称呼它，并以更低的价钱销售。如果味道是相同的，但成分是不同的，那么可口可乐制造商若起诉你复制他们的配方是没法胜诉的。我认为，Virgin（维京）几乎克隆出来了这个味道。当然，百事可乐并不相近：味道是那么不同，二者就像苹果和橘子的差别。

最后说回来，有许多不同的方法来解决回归问题（或任何数据科学问题。如果你使用正确的参数，则不同解决方案在效率方面往往是等效的。例如，你可以用纯优化或纯数学解决这个问题。统计学家会认为，使用这种方法不能为回归参数建立置信区间，或没法测试是否有一些参数等于0。作为替代，我自己设计了一个无须统计模型就可以计算置信区间的方法。这个方法被称为分析桥（Analyticbridge）第一定理，将在这本书更多技术化的章节里得到描述。

# 情景5：分析与诱导在提升销量上有何不同价值

这个例子说明即使是最好的分析，如果你忽略了其他关键的、使业务成功的要素，也不会很有用。总之，对于所有业务问题，分析并不是万能的。

这个情景的背景是提高转化率，如将网站访问者转化为活跃用户，或者是将被动的简报订阅者转化为业务意向（打开简报和点击链接即代表意向）。在此我们讨论简报的转化问题，虽然这个例子可以适用于许多不同的环境。

为了最大限度地提高总的意向数量，你需要使用诱导手段和分析手段。销售量是一个与下列影响因子密切相关的函数。

- 诱导

- 分析
- 产品
- 价格
- 竞争
- 声誉
- 市场营销

注意　如何评估每一个因子的权重超出了本章的范围。

首先，即使是度量“诱导”或“分析”也是很困难的。但是你可以用一个0～10的级别（10表示最好的），诱导值=9代表公司在使用诱导方法上非常努力，而分析值=0代表公司完全忽视分析手段。

在优化简报邮件方面（从而最大限度地提升意向的数量和规模），大多数公司把诱导设置为1，把分析设置为4或5。通常是通过多变量测试、数学分割、用户群定向进行分析的。这种方法来源于营销人员所受的训练——但是，这不是最好的方法。现在，ISP（互联网服务提供商）提供的用户细分，比传统的用户分组技术更关键。未能　送达给Gmail用户，比起分组不当送达给不恰当的Gmail用户，后者更糟糕（例如，年轻用户和年老用户虽然收到不同的信息，但是这些信息并不是有针对性发送的）。

另一个关键的错误是在邮件主题上，重复使用相同的关键词，刚开始这样可能有效，但最终会使你的用户厌烦而不想读你的简报（除非你已经找到弥补用户流失的方法，比如说雇用增长极客来帮助你解决问题）。如果你的竞争对手也使用完全相同的关键词，这个问题就更加复杂了。

丰富多样的、非炒作性的关键词对于分析型、高学历、厌恶垃圾邮件的用户来说非常有效果。例如，一个主题行，如*Weekly digest, July 29*（*digest* 用小写字母*d* ）比*25 Fantastic Articles From Top Big Data Gurus*（*From* 用大写字母*F* ）的效果更好。当然，后者可能在前两次效果很好，但最终还是无效的。此外，你的简报内容必须与主题行紧密相关，否则你失去用户的速度会比你获得新用户的速度快。

相反地，你可以基于诱导而不是分析手段，即，猜测用户现在喜欢什么内容，而不是根据历史的效果数据来设计内容。它也可以自动化完成，并转化为分析规则——例如，检测一个关键热词奏效的次数，例

如“肚皮舞分析”这一有效关键词可以持续多久，等等。过度使用通过分析找到的技巧，最终会使得诱导手段没有存在空间，也会拉低销售。但现在，似乎很多网络营销者忽视了这些规则。诱导比分析提供一个更大的杠杆效应（虽然它们应该混合在一起），特别是长期业务。

# 情景**6**：关于隐藏数据

在该情景下，你的数据就像Gruyere奶酪一样。它有很多、很大的孔。有时候，空白的空间占据着比数据本身更大的容积，就像在宇宙中暗物质比可见物质更丰富一样。这个情景不是关于浅层的或稀疏的数据，而是关于你看不到的数据，你甚至不知道它们的存在，但它们比你的数据仓库里的任何东西，都有更好的可操作价值。

以下是“Gruyere数据”的3种情况，以及每一种情况的补救方法。

## 丢失的或不完整的数据

这是最容易解决的问题。任何有才华的数据科学家都可以使用现代的、无偏的数据补齐技术，来解决这个问题。大多数分析软件也包括处理丢失数据的机制。

## 经过删改的数据

所谓删改，我的意思是从统计学角度看的。下面是一个例子：我们要估计所有至少涉及一次犯罪的枪支比例。我们使用的数据集（枪或犯罪统计）已经被删改了，比如品牌的新枪今天并没有用来杀人而没被统计在内，但可能下个星期就被用来射杀某人。此外，一些罪犯丢弃了他们的枪，在案发后可能追踪不到该枪支。

你如何处理这个问题？任何有才华的数据科学家都能轻松使用正确的统计技术处理这个问题，使用统计分布（通常是指数的）来衡量犯罪时间，并基于删改数据估计其平均值。这样问题就解决了。

## 隐藏数据

处理隐藏数据是一个大问题。首先，你甚至不知道它的存在，因为它是看不见的，至少从你的角度来说是看不见的。领域专业知识和统计学知识（更多技术知识的经验法则），会帮助你意识到潜在的隐藏数据。事实上，数据可能根本就不存在，在这种情况下，首先你必须组装数据。

思考这样一个例子：假设Target公司在试图优化它的收入数字。它分析了销售数据，看看与花园相关的物品什么时候销售量最好。该公司总部在明尼苏达州，2月份没有任何关于花园物品的销售数据，任何人提出这样的分析做法可能被当场解雇，或被怀疑是不是吃错药。然而，在加州，2月份Target的竞争对手们的花园物品销售量不错，一直到6月份都没有留给Target什么市场份额。Target不知道原因，错误地认为在加州花园物品这块没钱可赚。

你如何处理该问题？即使Target可能缺乏2月份的花园物品销售数据，但可以查看竞争对手的数据（例如，浏览和分析每天发送的数百万封垃圾邮件），这是正确方向上良好的第一步。但真正的解决方案是雇用一位有远见的数据科学家　。有才华的数据科学家利用每个人都能看到的数据；有远见的数据科学家 利用没人看到的数据。

# 情景7：汽油中的铅会导致高犯罪率。真的吗

这种典型的分析和观点可能在受尊敬的新闻媒体上出现过，然而这种分析不足以支持作者的观点。根据Rick　Nevin的计量经济学的研究，禁用含铅汽油约20年后，（汽油使用量高的）大城市的犯罪率达到高峰。20年的滞后是婴儿受铅影响到成为20岁罪犯所经历的时间。至少这是一些著名的计量经济学家基于过去大城市历年犯罪率的分析而提出的论点。但这又是一个缺乏适当的实验设计的例子。

那么，如何设计一个更好的研究呢？我们可以得到过去30年间所有特定规模的城市1万人的均衡样本，样本分成两个子集（罪犯和非罪犯），并检查（使用比率）罪犯是否比非罪犯在出生时更容易受铅影响。总之，要跟之前的研究相反，要针对个体而不是城市——也就是看

微观层面，而不是宏观层面——并进行经典的假设检验，使用标准抽样和适当的实验程序设计。

或者，如果你真的想研究原有的宏观时间序列（假设数据粒度细到每月），那可以进行*Granger* 因果关系 检验，这样会考虑原始时间序列转为白噪声后的所有互相关系的残差（类似于时间序列的频谱分析或相关图分析）。然而，如果你有几千个指标（因此会有几千个时间序列和几百万个相关性），你最终会发现非常高的相关性纯属意外。这被称为大数据的“诅咒”，在第2章中会对此详细描述。

相关性不是因果关系。不要轻易声称是因果关系，除非你能证明它。很多时候，多个相互依赖的因素会对同一问题产生影响。当婴儿潮一代（不太守法的一代）20岁时，也许正是犯罪发生高峰期。这可能是一个更可信的原因。

# 情景**8**：波音**787**（梦幻客机）问题

波音公司的梦幻客机推出几个月后不久，由于其电池相关问题，被FAA（联邦航空局）禁飞。主要的问题是其使用了从未在飞机上使用过的一种新型锂电池。这种强大的电池容易过热和着火，导致短时间内多次紧急着陆。

根本原因是设计电池的供应商缺少良好的实验设计。这是一个质量控制问题，而质量控制在很大程度上依赖于分析。以下的一些提问证明了更好的质量控制和实验设计可以解决电池问题。

- 这些电池（就像其他很多可以购买到的产品的电池，如汽车或笔记本电脑的电池）难道不能经过深入的质量控制检测，采用健全的统计技术，确保电池次品率或生命周期内产品的失败风险低于可接受的阈值吗？
- 会不会是质量控制测试没有根据最佳实践进行？
- 过热模拟能代表现实世界飞机起飞时的情况吗？
- “压力”测试电池的时间够不够长？
- 是不是日本的质量控制标准更低？因为这些电池的设计和生产是在

日本进行的。

- 这些电池的可靠性统计报告是错误的吗？

一种可能的解决方案是使用更好的机制来冷却这种类型的电池，但在飞机上从来没有使用过这种方法，不过，过去手机就因为过热而发生过火灾，所以现在所有手机中都使用冷却手段。与手机或笔记本电脑不同的是，在一架飞机上，因为外界温度远低于冰点，所以很容易冷却（甚至是冷冻）。>>更多新书朋友圈 免费分 享微信 xueba987

# 情景9：NLP的7个棘手句子

NLP是指自然语言处理 。当数据科学家分析非结构化的数据时，如原始 （未分类）文本 ，将会面临这类问题，所以我们在此将对该类问题进行说明。这类分析叫作文本挖掘 。

交叉参考 第2章会讨论如何在非结构化数据中创建结构（通常是分类）。

以下7种类型的语言模式，很难用自动化算法进行分析。

- “A land of milk and honey”变成“A land of Milken Honey” （这是针对*The Wall Street Journal* 20世纪80年代以来的语料训练而形成的算法，该语料里Michael Milken提到的次数比milk多）。
- “She threw up her dinner”与“She threw up her hands”对比。
- “I ate a tomato with salt”与“I ate a tomato with my mother”或“I ate a tomato with a fork”对比。
- 以“ing”结尾的单词——例如，“They were entertaining people”。
- “He washed and dried the dishes”与“He drank and smoked cigars”对比（在后面的情况下，他没有“喝”香烟）。
- “The lamb was ready to eat”与“Was the lamb hungry and wanting some grass?”对比。
- 具有多重含义的词，如“bay”，可能是一种颜色，或者窗户类型，或者海湾。

在这些例子中，或者在情感分析中，数据科学家试图猜测用户意图，以便正确地解读数据，得到正确的答案或决策。例如，这可能会在以下情况中发生。

- 在分析谷歌搜索数据时，根据用户查询应该显示哪些相关广告？
- 对于发布在Facebook网页上关于产品或公司的评论，如何分析评估它们是正面的还是负面的？
- 在分析Facebook的帖子时，如何判定一个帖子是不是因为违反了Facebook的政策或法律而应该被自动删除？

用户画像（如果可用）指标，或网页和电子邮件内容（特别是如果网页已经分类了），可以帮助解决歧义。

# 情景**10**：数据科学家决定着我们所吃的食品

影响美国人吃什么的因素有很多，但最大的因素是食品杂货店的利润率。这解释了为什么我们买不到红醋栗或百香果，但可以买到高能量饮料和富含糖分的食物。当然，这有一个反馈循环：美国人喜欢甜食，所以许多公司生产甜的食品，并且由于大规模生产，所以价格便宜，因此食品杂货店可以有效定价，销售量也好。

这也说明超市可以通过更好的分析来增加收入。这一切的背后都是数据科学，数据科学有助于回答以下问题。

- 为了客户利益和投资回报率（ROI），应该尝试去卖哪些新产品？红醋栗派？橘子酒？法国风味的樱桃派？野猪肉？紫薯奶酪？红鸡蛋？球状奶酪？（虽然球形不像平行的方形管道形状，从存储的角度来看不是最优的，但这是另一个数据科学问题。）
- 你如何确定一个新产品的成功或失败？你如何测试一个新产品（实验设计问题）？
- 哪些产品应该被淘汰？（美国的大多数杂货店将不再出售百香果、百香果汁和正宗的意大利香肠。）

- 如何度量提升力（收入增长）？你有考虑营销费用和其他费用等因素吗？
- 如何给一件商品定价？
- 如何进行交叉销售？
- 在营销活动中，如何优化投资回报率？
- 什么时候及在哪里销售哪种产品（根据季节性和地方性趋势）？
- 如何改进库存预测？

上次我去一家杂货店，购买全脂无糖酸奶。我在商店里找了10分钟，只找到最后的一盒（品牌是Dannon）。我已经准备多花点钱购买这种酸奶（世界上数十亿人消费了几千年的产品），而不是另两种选择：低脂肪，或原味带甜的酸奶。（讽刺的是，这种“低脂肪”版本含有180卡路里，而以前普通的酸奶只含有150卡路里。这是因为低脂肪的产品中有玉米淀粉。）

随着时间的推移，我发现杂货铺供应的产品数量正在减少。很多旧产品被淘汰，引入了一些新产品。显然，这些被淘汰的产品市场较小，如百香果。但是，数据科学能不能更好地帮助食品杂货零售商决定上架的商品，决定何时及在哪里销售，并以多少比例和什么价格销售呢？

答案是肯定的。一个好的解决方案是，借助具有较高预测能力的模型，在预测销售量和（每种产品）收入时，颗粒度更精细，方差更小。在这种情况下，虽然很多人尽量避免有脂酸奶，但在西部和东部沿海地区有很多较瘦的人不介意喝原味酸奶。因此，在西雅图或波士顿可以销售原味酸奶（也许仅仅是几十盒高价的产品，放置在低价低脂品牌旁），但在堪萨斯市则销售不了。

这也为专用杂货店创造了新的机会，如在美国西北部的PCC自然品市场，卖的正是其他超市已经停止出售的产品（只要是畅销的）。总之，是销售能产生利润但其他超市已经不再销售的产品。

这个例子也说明了，对于数据科学家，沟通技巧是很重要的：提出了一种新方法，并说服高管使用这种方法，可以同时优化利润，且给客户带来长远的价值。当然，这样的做法将是一个长期的战略投资，在短期内可能无法满足财务上的数字（公司的首席执行官需要和股东说明）。

# 情景11：用较好的相关性增加亚马逊的销售量

亚马逊是如何通过重新定义相关性来增加销售量的？答案：通过改进其搜索和相关性引擎，把物品价格当作主要的相关因素。这里所描述的优化和提升投资回报率的做法，适用于所有的数字化产品类别，虽然在本书中我们只针对书籍这个类目进行了讨论。

## 搜索引擎

当你在亚马逊网站上的购书栏搜索关键词时，返回的搜索结果页面，有10本推荐的书与你的关键词匹配。这是由搜索引擎执行的。搜索引擎会以某种排序显示图书。该排序是基于价格或关键词相似度的。

## 相关性引擎

如果你搜索一个特定的书名，亚马逊还会基于其他用户的购买记录，显示你可能感兴趣的其他的书。这项任务是由相关性引擎执行的。

## 技术讲义

相关性引擎的工作原理是这样的：如果有m（A，B）个用户在过去30天内都购买了A书（是你想要购买的书）和B书，而k（A）代表购买A书的用户数，k（B）代表购买B书的用户数，那将A和B之间的关联（从交叉销售的角度，这些书的关联程度）定义为R（A，B）=m（A，B）/SQRT{k（A）*k（B）}。所推荐的书籍排序完全由函数R（A,*）决定。

## 更好的排序标准

昂贵的图书将产生很少的销售量，但有巨大的利润。廉价的图书利

润较少，但销售数量能弥补每本书的利润。总之，如果显示的书都跟用户的相关性评分相同，那么在#1位置显示的书的定价应使得预期总收益最优。图1-1显示了一个假想的最优图书定价是21美元。

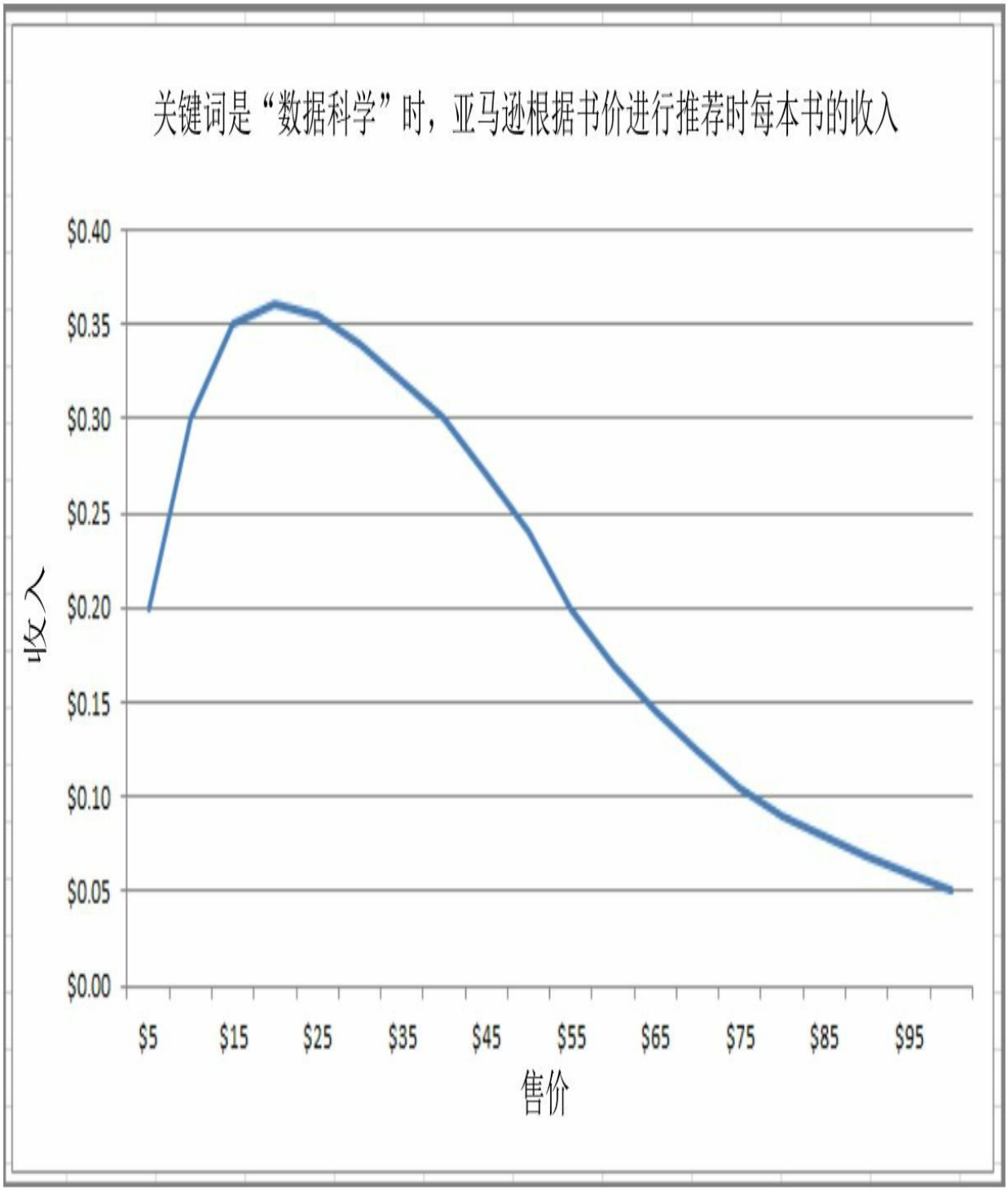


图1-1　最优图书定价

此图基于模拟的数字，假设书售出的机会是图书定价的指数递减函数。也就是：

$$P（sale \mid price） = a * exp（-b*price）$$

其中sale为销售量，price为售价。一个更一般的模型是：

$$P（sale \mid price, relevancy\ score） = a * exp（-b*price） * f（relevancy\ score）$$

其中relevency score为相关性评分。

另一种能进一步增加收入的方式是，将用户数据也包含在公式中。一些用户有能力购买一本昂贵的书。对于经常购买更昂贵图书的用户，应该在他们的搜索结果中，显示更昂贵的书籍。

## 把所有东西放在一起

当销售产生时，我们怎么知道，是因为把正确定价的书籍显示在了顶部，还是因为完美的相关性？例如，“数据科学”和“大数据”之间的关联性很强，但“数据科学”和“云计算”之间的关联性不是很强。如果你的目标是利润最大化的话，那么是把一本昂贵的“云计算”书推荐给一位对“数据科学”感兴趣的富人，或者给他推荐一本不太贵的与“大数据”相关的书，哪种做法明智？这也取决于你如何定义收入优化：是长期（关联性更重要）或短期，还是短期和长期的混合。正如你所看到的，从价格因素中分离出关联性影响是不容易的。

当关键词或类别的相关性是基于“小数据”的（指专门的用户查询或销售量小的书籍），价格因素特别有效。此外，检测到底是什么带来特定的转化或销售，是一个复杂的问题，这被称为归因　。在复杂的情形下，宏观经济（长期，总量的）指标，混合了非常细粒度的实时属性指标，可用于营销组合优化 。使用价格弹性模型 和效率曲线 也可进行价格分析。

另一个有意思的领域是自定义定价，实时地根据客户、客户的购买记录（如果可用）、销售该书的网站（根据网站的统计数据）和书本身来确定书价。一些研究表明，一个固定的价格是最好的，否则精明的用户会尝试多次购买相同的项目，直到他们得到可能最低的价格（虽然他

们没有办法知道什么时候价格最低）。然而，向那些不介意购买昂贵图书的用户销售高价图书（根据他们的购买历史），而不是推荐用户已经购买过的书，是改善投资回报率最容易的方法。然而，在一些国家，这种定价策略已经被禁止，在其他一些地方也有争议。特别要注意，如果定制化价格是悄悄执行，没有被用户了解和同意，还采用不道德的数据挖掘的做法并侵犯隐私的话，可能会导致用户对你信任下降，并最终导致客户流失、法律诉讼和收入下降。最终，用户会注意到价格是定制的。

另一种最大限度地提高亚马逊的利润的方法，是按需求　　印刷书籍，而不是管理库存、每月预测每本书的销量。数据科学还有助于确定仓库位置，以优化交付效果（在速度和成本方面），以及销售税的优化，从而更好地与其他书商竞争。

# 情景12：检测Facebook上的假档案或假“喜欢”数

某些网站出售假Twitter粉丝（1 000个粉丝10美金）、假的Facebook账号，或甚至假的雅虎邮件账号，用来传播垃圾邮件，提升个人或零售商人气，据说还可以用来促进销售。他们预先告诉买家，这些粉丝是假冒的。他们也在网上卖在Facebook点的“喜欢”数，并且他们声称“喜欢”数是来自“真正的”Facebook用户。

自动检测假“喜欢”数最简单的方法是看相关评论数：如果一个Facebook（或任何）帖子后有5000个“喜欢”数，但没有评论或只有20条评论，并且评论内容都类似于“这是个很棒的帖子”，那就可以确定“喜欢”数和评论是机器人或水军造假的。

数据科学算法就是根据这样的指标，以及社交网络图、变化速度、近期账户创建和发布情况、自然语言处理和僵尸网络/欺诈检测规则来处理问题的。一些造假可以实时检测到，而一些会稍后检测和消除掉，比如通过临时分析（数据分析师执行的反应式分析）或一天结束后的算法自动分析（惯例分析）。

# 情景13：餐厅的分析

对于小企业，特别是餐馆，分析可以有很多种形式。这些企业需要解决的问题包括定价优化、库存管理、销售预测、获取竞争对手的信息、客户对你的建议（比如需要使用一些自动化软件定期检查客户对餐厅的评论）、空间优化（餐厅最多可以放多少张桌子）、葡萄酒和奶酪的选择、菜单变化的频率、每天和每月的开业和歇业时间、维持的厨师数量（包括决定最佳薪酬）。

可能最容易解决的问题之一是桌子的布局。最佳的解决方案是让2个座位的桌子靠在墙边，4个座位的桌子在房间的中央。这些桌子应该是圆形的还是正方形的，平行于墙壁或形成一个45° 角？很容易找到一个数学解决方案，或者至少做一些模拟，找出哪种最有效。正方形的桌子比圆形的好，可以轻松地组合起来，适应更大的派对。假设餐厅总是被预订满并赢利，如果你增加20%的座位容量，那么你的利润也应该增加20%。

# 数据科学的历史、开拓者和现代趋势

随着数据科学的发展，其他相关专业出现了下降趋势，如统计学和计算机科学。谷歌的公开数据（见`http://bit.ly/1aF8D5T` ）如下。

- 自2010年以来，数据分析师的数量不断增加。
- 自2005年以来，统计学家的数量有所减少。
- 自2005年以来，计算机科学家的数量有所减少。
- 自2012年以来，数据科学家的数量激增。

你可以在LinkedIn或Indeed上找到其他的公开数据（每个招聘广告的申请人数），不过这个数据跟招聘市场相关。

其他类似的数据显示，所有传统领域的人数都在减少，比如六西格玛、数据挖掘、统计学、运筹学等。谷歌数据表明，在2011年左右大数据开始出现并呈指数增长趋势，到2013年大数据比数据挖掘和六西格玛

更流行了。同样也是根据谷歌数据，尽管大数据的崛起引人注目，但开始于2006年对“分析 ”这个关键词的搜索量更为壮观。在2012年搜它的人数逐渐减少，但仍然比大数据 高出6倍。

当然，在2000年，许多专业人士（包括我），做了统计学、运筹学、计算机科学、六西格玛、精算学、生物统计学，或一些其他的狭义定义的“分析类”活动，积累了丰富的经验、领导能力、广泛的知识、敏锐的商业头脑、跨越许多领域的专业知识。因此他们的职位头衔经历了演变，但他们都有共同点：数据分析行业从业者。同时，数据和现代数据架构的发展，如MapReduce，影响了所有行业，成为许多专业人士的共同特性和必备。

注意　　数据科学家 比数据挖掘 更广泛，包括数据集成、数据采集、数据可视化（包括仪表盘）和数据架构。数据科学家　还须度量数据科学活动的投资回报率。

# 统计学将会复兴

很多人都说统计学会消亡，有些主要的统计学家自己也说统计学会消亡。我相信统计科学最终会复兴，但它会更多地应用于大数据，并适应于大数据，且由更少的模型驱动。它将与计算机科学、预测模型、数据挖掘、机器学习、运筹学和六西格玛的某些方面，以及数据库架构结合在一起，被归纳称为数据科学、业务分析、决策科学、数据情报、分析学，或其他一些尚未被创建或重复使用的术语。我们现在正处于分析学革命的中间阶段。

特别是，像我这样的人，虽然有一个新的职位头衔——数据科学家，但仍然做统计学兼职，有时甚至涉足前沿的理论统计学。对我而言，我已从业25年之久，一直使用那些在1750年被认为太复杂、计算能力不够而被抛弃的，但是足够健壮的技术。在1750年，由于当时计算能力的缺乏，导致了1800年左右一批新颖的、对数学友好的技术出现，它们具有简单的公式、方程形式，如最小二乘回归。这个框架一直延续至今，可能是导致传统统计学家人数减少的原因，现在大数据的健壮性比以往任何时候都更重要，当在分布式系统中（在有MapReduce的云中）几分钟就可以处理10亿字节的数据时，计算的复杂度将不再是问题了。

同时，大多数现代科学家、地理学家、医生、计量经济学家、运筹学专业人士、工程师等都具备较好的应用统计知识。然而，软件工程师和计算机科学家有时会像记者一样忽略或误用统计科学，如开发的系统里（例如，推荐引擎）有大量未被发现的虚假评论和欺诈行为。最终，统计科学将开始踏足这些领域。

有人说，大多数数据分析师都是统计学家。在我看来，数据分析师是一个初级头衔，通常是一个有工学学士学位或文学学士学位的人。统计学家有更多的理论背景，在大数据出现之前就使用以前开发好的数据模型，并且拥有硕士或博士学位。每天编写SQL查询和报告的人是数据分析师。

我不当统计学家的部分原因是因为美国统计协会：它改变了统计学家　这个关键词的意义，这限制了统计学家的发展前景，使其狭窄且单一，只与医药行业、政府（调查、人口普查、政治事务）、小数据（统计学家的大部分收入来源）相关。在过去15年里，该协会一般不参与或跟随基于大数据的新的统计革命。作为一名比利时公民，我可以对比利时统计协会说同样的话。所以这一趋势不仅限于美国，而且也不仅限于（美式）英语国家，还包括法语和荷兰语国家，以及其他语种国家。

统计学家应该非常熟悉计算机科学、大数据和软件——1万变量的10亿行数据，对真正的统计学家应该是小菜一碟。在云中（甚至在笔记本电脑上的流数据），这个数据量都能够快速处理。第一步是缩减数据，即使你必须保存所有的观测值和变量，仍然可以缩减数据。一个优秀的计算机科学家也能生成置信区间：你不需要因此而成为一名统计学家，只需会使用本书后面讨论的分析桥第一定理即可。计算机科学家和统计学家之间的区别变得越来越小和模糊。但无须担心，虽然你在学校没有学到这些知识（统计学的课程），你仍然可以在网上学习。

# 历史与开拓者

现在，让我们来看看数据科学的历史，以及一些在分析学和数据科学领域堪称先驱的公司。首先，看一看从20世纪80年代末开始流行的关键词，和对2022年的一个预测。

- 1988年的关键词：人工智能。另外还有：计算统计学、数据分析、模式识别、规则系统。
- 1995年的关键词：网络分析。另外还有：机器学习、商业智能、数据挖掘、投资回报率、分布式架构、数据架构、量化、决策科学、知识管理、信息科学。
- 2003年的关键词：业务分析。另外还有：文本挖掘、非结构化数据、语义网、自然语言处理（NLP）、关键绩效指标（KPI）、预测模型、云计算、升力、收益率、NoSQL、商业智能（BI）、实时分析、协同过滤、推荐引擎和移动分析。
- 2012年的关键词：数据科学。另外还有：大数据、分析学、软件即服务（SaaS）、按需分析、数字分析、Hadoop、NewSQL、内存数据分析、机器对机器（M2M）、传感器数据、医疗保健分析、效用分析、数据治理、列数据库。
- 2022年的关键词：数据工程。另外还有：分析工程、数据管理、数据整形、优化的艺术、优化科学、优化工程、业务优化、数据智能。

这些里程碑让我们对如何利用数据有更通用、更全局、更全面的理解。大约开始于1995年的大数据运动，谷歌是最重要的贡献者之一。谷歌通过引入谷歌文件系统、MapReduce，解决了传统的分布式系统/数据库管理系统（DBMS）的数据库存储容量限制。（人们经常在2003年至2006年期间的行业会议上讨论谷歌的Bigtable大表方案。）然后是HBase和Hadoop分布式文件系统（HDFS）。除了谷歌，雅虎和Facebook也对Hadoop和开源社区做出了重大贡献，推动了技术进步。

专注于分析的先驱公司，有以下26家，可以在它们各自的维基百科页面，找到很多有趣的历史资料。（警告：这些网页，比如维基百科主题页面，包含商业内容，编写者是有利益冲突的人、内部人士或与这些公司利益有关的人。）

- Datawatch （`http://en.wikipedia.org/wiki/Monarch` ）
- Excel （`http://en.wikipedia.org/wiki/Microsoft_Excel` ）
- FICO （`http://en.wikipedia.org/wiki/FICO` ）
- Greenplum （`http://en.wikipedia.org/wiki/Greenplum` ）
- IMSL（`http://en.wikipedia.org/wiki/IMSL_Numerical_Libraries` ）
- Informatica （`http://en.wikipedia.org/wiki/Informatica` ）

- KNIME （http://en.wikipedia.org/wiki/KNIME ）
- KXEN （http://en.wikipedia.org/wiki/KXEN_Inc. ）
- Lavastorm （http://en.wikipedia.org/wiki/Lavastorm ）
- MapReduce （http://en.wikipedia.org/wiki/MapReduce ）
- Mathematica （http://en.wikipedia.org/wiki/Mathematica ）
- Matlab （http://en.wikipedia.org/wiki/Matlab ）
- Netezza （http://en.wikipedia.org/wiki/Netez za ）
- NoSQL （http://en.wikipedia.org/wiki/NoSQL ）
- Oracle （http://en.wikipedia.org/wiki/Oracle_Database ）
- PMML （http:/en.wikipedia.org/wiki/Predictive_Model_Markup_Languag ）
- R编程语言（http://en.wikipedia.org/wiki/R_ ）
- RapidMiner （http://en.wikipedia.org/wiki/RapidMiner ）
- S-PLUS （TIBCO） （http://en.wikipedia.org/wiki/S-PLUS ）
- SAS （http://en.wikipedia.org/wiki/SAS_Institute ）
- Splunk （http://en.wikipedia.org/wiki/Splunk ）
- SPSS （http://en.wikipedia.org/wiki/SPSS ）
- Statistica （http://en.wikipedia.org/wiki/STATISTICA ）
- Sybase （http://en.wikipedia.org/wiki/Sybase ）
- Tableau （http://en.wikipedia.org/wiki/Tableau_Software ）
- Teradata （http://en.wikipedia.org/wiki/Teradata ）

此列表重点关注已经有一定历史的大型数据分析公司。在过去几年，无数的参与者如雨后春笋般涌现，特别是在MapReduce生态系统行业。

交叉参考　关于大数据生态系统的更多信息，参阅第2章。

# 现代的趋势

在新技术方面，很多当今技术都跟大型分布式数据库的集成分析有关。要让数据架构师、工程师和数据科学家互相沟通，并做二选一的转变——以前是“数据到分析”（传统数据科学方法），现在是“分析到数据”（是受数据架构师和数据库公司青睐的更为现代的方法，因为它的

速度更快）。分析到数据，意味着分析是在数据库或数据存储环境里就近进行的，而不是把数据下载到本地分析它。这减少了多个数据库用户的情况下大量冗余下载的必要。在本章的最后一节中可以读到更多相关的细节。

当然，这归结为在数据库里或就近位置，运用正确而先进的分析工具（不只是提取（Extract）、转换（Transform）和加载（Load）方面，简写为ETL）。当分析是在数据库内部进行时，有时被称为内存数据分析。它是一种更为强大的“分析到数据”的形式，是将分析集成和嵌入到数据库架构中，且主要在快速存储器（内存）中进行，有时还会使用多个服务器。但其中一个问题是，现代数据库处理涉及更复杂的编程语言，而现在大多数人还在使用SQL，要他们改变旧习惯很难。因此，像Pivotal这样的先驱公司发明了一种名为Fast SQL的系统，使那些习惯SQL的程序员不需要学习新的、更加复杂的语言，并且代码优化是在Hadoop（一种分布式架构）下运行的。

其他现代趋势包括自动化的机器到机器实时通信，在高频交易策略或大规模竞价系统里都有使用。一个例子是eBay基于谷歌的点击付费关键词过去的效果（当历史记录可用时），或对没有历史记录的新关键词分析计算其得分，每天实时更新1 000万个关键词出价。这些机器到机器的通信都是通过API或AaaS（分析即服务，按需求提供）。API调用无非是HTTP请求（实际上，大多数情况下都是HTTPS请求），通过查询字符串参数（用XML格式编码的键/值对），指定所需的服务类型（如关键词的价格和数量的预测）。

此外，现代数据如此不同和独特是因为它的多样性（有时是Twitter推文的非结构化格式，有时是来自移动设备或传感器的结构化良好的数据）、产生速度和数量，这使得传统的统计分析并不总是那么合适。

概括说，下面这些都是数据科学现代趋势的特征。

- 内存数据分析。
- MapReduce和Hadoop。
- NoSQL、NewSQL和图形数据库。
- Python和R。
- 数据集成：混合非结构化和结构化的数据（如数据存储和安全性、收集数据时的隐私问题，以及数据的合规性）。

- 可视化。
- 分析即服务，简写为AaaS。
- 文本分类或标签和分类系统，以便于从原文本中获取洞见力，给非结构化数据加上结构。

最后提一下Perl，它是伟大的编程语言，5年前比较流行，却是现代编程环境的牺牲品。它已经被Python及它的分析和图形库取代。虽然Perl非常灵活，且无须专注于代码的编写，但Perl程序维护困难且成本高。Perl没能在现代灵活而精简的环境中存活下来。也有些人说，最优秀的分析工具Excel也会消亡，但我并不这么认为。现代版本的Excel使用云服务，从互联网上获取数据，并将其存储在多维数据表里。

# 最近的问答讨论

最近我与不同社区的数据架构师进行了以下讨论，特别是（但不仅局限于）同LinkedIn上数据仓库研究所（TDWI）的小组成员进行讨论。他们阐述了一些在新的分析革命完成之前，仍然需要解决的挑战。以下是数据架构师和数据库管理员询问的几个问题，以及我的答复。这些问答跟在SQL查询中优化连接操作有关，或干脆跟SQL无关。一些现代数据库提供了许多功能，包括哈希表连接和可以由最终用户进行微调的查询优化程序。该讨论说明了数据科学家、数据架构师和业务分析师之间的冲突。它还涉及许多创新的概念。

问题： 你说SQL的瓶颈之一是用户写查询是通过3个连接实现的，这些查询何时可以分成两个查询，每个查询两个连接？你能详细说明吗？

答案： 通常，我写SQL代码的方式是将它嵌入到一种编程语言里，如Python，且在内存中储存我需要的所有查找表，如哈希表。所以我很少有做连接操作，当我这样做时，它顶多只是两个表格。

在一些（罕见的）情况下，查找表太大，而无法放入内存中，所以我使用抽样的方法，且使用子集和聚合规则。一个典型的例子是，你的数据集（如网页日志文件）的一个字段是一个用户代理（即浏览器，简称UA）。如果你有更独特的UA数据，在内存中装不下，但只要你保留

1 000万个最受欢迎的UA，并把2 000万个罕见的UA 分到几百万类别里（基于UA字符串），在大多数应用中能得到很好的结果。>>更多新书朋友圈免 费分 享 微信 xue ba987

作为一个算法专家（不是SQL专家），我花了几分钟通过Python中的哈希表（使用自己的脚本模板），做了一个高效的四表连接。我所做的绝大部分都是进阶分析，而非数据库聚合：它们涉及先进的算法，但在Python中编码很简单，如隐性决策树。总之，对于非专家级SQL用户，如业务分析师，是培养他们写出更好的SQL代码，包括复杂的连接，还是培养他们学习将Python跟SQL代码混合编程，哪种方法更简单、更有效？

具体而言，我心目中的系统是不需要很频繁地从系统下载查找表（也许一周一次），但需要比较频繁地访问主表（事实数据表）。如果你必须非常频繁地上传查找表，那么Python方法将失效，且你的同事会不高兴，因为你经常下载，从而降低了整个系统的速度。

问题： 像你这样（运行Python或Perl脚本访问数据库）的人会是数据库管理员（DBA）的噩梦。你不觉得你是给DBA带来问题的根源吗？

答案： 因为我更擅长Python和Perl而不是SQL，我的Python或Perl代码是无错误、易于阅读、易于维护、最优化、稳健和可重复使用的。如果我用SQL编码，则它的效率是很低的。我所做的大部分工作是算法和分析（机器学习的东西），而不是查询数据库。我只是偶尔下载查找表到我的本地机器（保存为哈希表和存储为文本文件），因为大多数情况下每周变化不大。当我需要更新时，我只需提取上次更新后添加的新行（基于时间戳）。在运行处理数据量很大的SQL脚本之前，我会进行一些测试，以了解它会消耗多少时间和资源，看看我能否优化。我是一个SQL用户，就像所有的统计学家或业务分析师一样，不是一个SQL的开发者。

但我同意，我们需要找到一个平衡，以尽量减少数据传输和处理，就近运用更好的分析工具。至少，我们需要能轻松地在非生产数据表和系统里部署Python代码的能力，并且需要合适的磁盘空间（也许200 GB）和内存（至少几GB）。

问题： 你偏好的编程语言是什么？

答案： 有些人更喜欢使用脚本语言，而不是SQL。人们认为SQL可能不够灵活且易于出错，产生错误后没人注意到，因为连接操作中有Bug。

你可以写简单的Perl代码，易于阅读和维护。Perl可以让你把精力集中于算法，而不是代码和语法。不幸的是，许多Perl程序员写的代码晦涩难懂，这使得Perl有了坏名声（在代码的维护和可移植性方面）。但是，不一定都是这种情况。

你可以使用多个SQL语句和视图，把一个复杂的连接分解成几个较小的连接。你可能认为，数据库引擎会帮你整理这种分解过程，把不那么高效的SQL代码变得更加有效。至少你可以测试这种方法，看看它是否和一个有许多连接的、单一的复杂查询的运行速度一样快。分解多个连接成几个简单的语句，让业务分析师编写简单的SQL代码，也便于其他程序员重用或维持。

看到一些软件自动修正SQL语法错误（不是SQL逻辑错误）是件有趣的事情。这将为很多像我这样的、非专家级的SQL程序员，节省很多时间，因为经常重复发生的拼写错误，可以自动修正了。同时，你可以使用图形用户界面（GUI）生成合适的SQL代码，利用大多数数据库厂商或开放源码提供的工具，如适用于Oracle的Toad。

问题： 你为什么说这些内置的SQL组合优化器，对于终端用户是黑盒技术呢？你认为参数不能由终端用户调整吗？

答案： 我总喜欢对我做的事情有所控制，虽然不一定是全部控制。例如，我比较满意Perl处理哈希表和内存分配的方式。我宁愿用Perl黑盒内存分配或哈希表管理系统，而不是像C语言那样要自己从头开始，或更糟糕的是，要编写一个编译器。我只是有点担心黑盒优化——我看到过非专家级用户鲁莽地使用黑盒统计软件而造成的危害。如果我至少有一点控制，我会感觉更舒服，甚至只是简单地给DBA发送电子邮件，让她帮助改善我的查询语句，让她帮助解决我的疑问，也许只是微调组合优化器，这对所在组织和其他用户是有必要和有益的。

问题： 你不认为你的方法已经过时20年了吗？

答案： 结果比方法更重要，只要该过程是可重复的。无论我用什么工具，只要我能打败竞争对手（或者帮助我的客户打败竞争对手），

正如有人曾说过，“如果没有坏掉，就不要修理它。”有时我使用API（例如，谷歌的API），有时我用网络爬虫收集外部数据，有时使用Excel或优秀的数据库，有时（不使用任何数据）依靠肉眼、敏锐分析和直觉，效果都很好。有时我使用统计模型，有时非常现代的架构也是必要的。很多时候，我使用这些技术的组合。

交叉参考　我有几个“浅分析”的例子，比复杂的架构效果更好，包括一些电子邮件营销优化和股票交易的情况。这些将在第6章中讨论。

问题：为什么你要问“数据到分析”方法是否有意义？

答案：　我问这个问题的原因是，因为有些事情一直困扰着我：基于不算过时的观察（不超过三四年），我提到的一些做法在数据分析领域里已经根深蒂固（分析是指机器学习、统计学和数据挖掘，而不是ETL）。这也是一个尝试，看看是否有可能在数据科学家和数据架构师这两个非常不同的社区之间建立一个更好的桥梁。数据库搭建者经常（但并不总是）需要数据科学家对组织好的数据提炼出见解和价值。而数据科学家经常（但并不总是）需要数据架构师建立优秀的、快速的、高效的数据处理系统，这样他们就可以更好地专注于分析了。

问题：　所以本质上你维护缓存系统，对查找表的本地副本定期进行小的更新。两个像你一样的用户，做着同样的事情，过一段时间后，会得到两个不同的副本。你如何处理该问题？

答案：　你是正确的，两个用户有两个不同的查找表副本（缓存），造成了问题。虽然对我来说，我倾向于与其他人分享我的缓存，因此它不像5人在研究5个不同版本的查找表。虽然我是一个高级数据科学家，也是一个设计师或架构师，但不是一个数据库的设计师或架构师，所以我倾向于有我自己的、与团队分享的本地架构。有时我的架构存储在一个本地的小数据库里，偶尔会存在生产数据库里，但很多时候是作为有组织的普通文件或哈希表存储在本地驱动器上的，或保存在数据库服务器以外的云里的，但如果数据量大，那么数据库到云的距离不会太远。很多时候，我最重要的“表格”是总结摘要表——要么是很容易用纯SQL生成的简单聚集表，要么是复杂的交易得分（按客户、日期、商户或更细的颗粒度评分），所用的算法太复杂了，以至于使用SQL不能有效地编码。

我的“缓存”系统的好处是尽量减少耗时，减少对系统内其他人的影响。其缺点是，我需要维护它，并且本质上我是在复制数据库系统本身的进程。

最后，对于研究数据的统计学家来说，只要数据是几乎正确的（是指查找表不是最新版本的，而是“缓存”系统中存储的没有频繁更新的），甚至说数据只是抽样，这都不是问题。通常，收集的数据是我们试图采集和度量的信号的近似——所以数据难免混乱。对预测模型来说，不管是从数量适量的数据集（我的“缓存”），还是原始、最新的数据集，或者是认为注入5%噪声的数据，所能得到的投资回报率——在大多数情况下几乎相同。

问题： 你可以对代码的维护和可读性做些评论吗？

答案： 如果某人写了晦涩的SQL，在他离开公司后，或更糟的是，当SQL移植到不同的环境（不是SQL环境）时，这对于要了解原始代码的新程序员来说，是一场噩梦。如果可读性好的SQL（也许包含更多语句，更少的高维连接）运行速度和复杂的SQL语句一样快，这得益于内部对用户透明的查询优化程序，那么为什么不使用易读的代码代替呢？毕竟，优化程序应该使这两种方法是等价的，对吗？换句话说，如果两段代码（一个短但晦涩难懂；一个较长，易于阅读、维护和移植）具有相同的效率，既然它们本质上会被优化器变成相同的伪代码，那我喜欢较长的版本，因为它的编写、调试、维护等需要更少的时间。

那些能将难看、晦涩但高效的代码变成好看、易于阅读的SQL的工具，如“SQL美化器”，可能有市场。当将代码移植到不同的平台时，它是很有用的。虽然在一定程度上，这已经存在了，但你可以很容易地对数据库系统所有的查询或查询集进行图表化和可视化。SQL美化器在某些方面类似于一个能把汇编语言转化成C++的程序——简而言之，是一个反编译器或解释器。

# 总结

本章首先讨论数据科学不是　什么，包括传统学历课程需要如何适应企业和政府的发展。这种适应涉及处理数据的新方法，以及数据的新

类型，包括大型、非结构化和高速流数据，有时是由传感器生成的数据或交易数据。

接着本章讨论了现实生活中可以使用数据科学并且确实奏效的许多情景。最后，我们讨论了数据科学的历史，包括该领域的一些先驱公司和现代趋势。

本书第2章将讨论什么使得数据科学成为一门新的学科，并说明了大数据与10年前所看到的数据之间的差异，解释了将标准统计学技术盲目地应用到大数据时为何会失败、问题到底是什么、应该如何检查和解决。

# 第2章
# 大数据的独特性

在第1章中，我们讨论了什么是以及什么不是数据科学，认识到数据科学的范畴不仅仅局限于数据分析、计算机科学或统计学。本章将进一步探讨数据科学这门新的学科。

本章首先讨论与大数据相关的最重要的两个问题。然后，分析一些现实生活中使用大数据技术的例子，并讨论高效的大数据团队氛围中的沟通交流问题。最后，讨论了统计学在目前和将来如何成为数据科学的一部分，并谈及大数据生态系统的组成部分。

## 两个大数据的问题

关于大数据，有两个问题必须讨论和理解：大数据“诅咒”和快速数据流动的问题。下面将讨论这两个问题。

## 大数据“诅咒”

虽然一些数据科学技术在小型、中型和较大型数据集上运作良好，但却不一定适合大数据。在大数据环境下，如果轻率地使用这些数据科学技术，可能会带来危险，这就是大数据“诅咒”。这个问题在大数据交易的缺陷中（本章提出了解决方案）阐述得比较清楚。

简言之，大数据“诅咒”就是，当你在包含几十亿或几万亿数据点和几千个指标的大型数据集里搜寻模式时，注定会发现存在一些模式但并

没有任何预测能力。更糟的是，这些强烈的模式可能会是：

- 完全出于偶然（像中彩票一样）。
- 不可复制。
- 没有预测能力。
- 包含模糊的、易被忽略的，但具有较强预测能力的弱模式。

所以问题就是，在海量数据中该如何区分真实的信号和偶然的信号？让我们来看一个例子：在时间序列中识别强相关性或关联性。

## 例子

如果你有1 000个指标（时间序列），可以计算出499,500=1000*999/2个相关系数。如果把时间差形成的交叉相关性也考虑在内（例如，IBM今天的股价和谷歌两天前的股价），我们需要处理数百万的相关系数。在所有这些相关系数中，有一些系数碰巧取得极高的数值，如果用这样的系数来做预测性建模，就会遭遇失败。切记，统计学家在开始一个项目时首先要做的步骤之一，就是用所有指标来分析交叉相关性——这是探索性分析步骤的一部分。然而，对规范化的时间序列进行频谱分析（而不是相关分析）能够提供更健壮的机制来识别真正的关联关系。

为了进一步说明这个问题，假设给定k个时间序列，每个序列分别包含n个观测值，比如在长度为n天的同一时期内，具有不同时间差的k个不同股票代码的价格deltas系数（价格升高或降低）。你希望发现类似这样的模式，“如果谷歌股价上涨，1天后Facebook股价会下跌。”要发现这种可以带来赢利的模式，你必须计算成千上万只股票及其不同时间差的交叉相关性，这些时间差可能是一两天，也可能是一两秒，取决于进行的是日间交易，还是快速高频的日内交易。

通常，你会采用较少数量的观测数据（比如，n=10天或n=10mm），因为这些模式会迅速消失。（如果你的竞争对手发现了这些模式，它们就不能再用来进行赢利了。）换句话说，你可以假设n=10或n=20。而在一些以月数据来进行计算的情况下（环境统计或新型疾病的出现），可能需要设定n=48，对应2年内所收集的月数据。在某些情况下，n可能会大得多，此时大数据“诅咒”反而消失了。当n小于200，而k相对较大时，比如k=500，大数据“诅咒”的影响会较为突出。然而，

n很大（>1　000）同时k也较大（>5　000）时，较少能观察到大数据“诅咒”的影响。

并非所有大数据都需要计算数百万的交叉相关。而这里描述的概念适用于一般的交易数据，或者那些需要被划分到数十亿数据桶（通常较小）之后，再被打分计算的数据。同时，大数据的范围可以从10GB到PB，并涉及上亿的观测值——通常每年几十亿的观测值，或者有时每天几十亿，或者每秒成千上万。需要注意的是，在有些情况下大数据可能是稀疏的，或者高度冗余的，所包含的信息远低于人们的想象。社交网络数据、视频流，以及一般来说任何一个能被有效压缩的数据集，均属这种情况。

交叉参考　在分析大数据，以及创建数据词典时，交叉相关分析是前期步骤之一（探索性数据分析的一部分）。在本书随后章节中，你会读到更多关于数据处理步骤的内容。

- 怎样创建数据字典请参阅第5章。
- 数据科学项目的生命周期请参阅第5章。
- 定义请参阅第8章。

现在，我们用数学方法来估算所检测到的高度相关性可能出错的概率。你可以使用蒙特卡洛模拟的方法来计算该概率，但本文将使用传统的统计建模方法。

假设有一个新的参数m，代表从k个时间序列中选出的、成对（二元变量）的独立时间序列数目。现在需要计算这m对时间序列的相关性。此处有一个理论问题：假定有m个独立配对时间序列，每一个都包含n个由随机数字生成器生成的数字（观测值可以是，比如，两支不同股票在指定时间的、规范化的模拟价格），在m个相关系数中，至少有一个高于0.80的概率有多大？

在这个设计中，理论相关系数（相对于估计的相关性）是0。要回答这个问题，假定（不失普遍性）这些时间序列（在简单的规范化后）是高斯白噪声，那么估算的相关系数，设为r，是（渐近地——即当n取值不是很小的时候）正态分布，均值为0，方差为1/（n–1）。r比指定的某个数a大的概率（比如a=0.80，意味着强相关）是p=P（r>a），P代表一个正态分布，均值为0，方差为1/（n–1）。在这m个二元（配对）的

时间序列中，至少有一个的相关系数超过a=0.80的概率就等于1–[（1–p）$^{m}$]。

以下面为例：

- 如果n=20且m=10　000（10　000个配对时间序列，每对有20个观测值），那么结论是错误的概率（即a=0.80）是90.93%。
- 如果n=20而m=100　000（仍然是一个相对较小的值），那么结论是非常 错误的概率（即a=0.90）是98.17%。

现在，它的实际操作如下：给定k个指标或变量，用每个指标来计算一个以不同时间间隔n计算出来的时间序列。我们可以计算所有的交叉相关系数，即m=k*（k–1）/2。然而，m对时间序列间的独立性假设现在被违反了，因此相关性将进一步远离比如像a=0.90这样的高值。此外，你的数据不是随机的数字——即不是白噪声。因此，理论上的相关值远高于0，也许当n=20时，这个值接近于0.15。进一步，m会远高于比如10 000或100 000，即使在你只有k=1 000个时间序列时（例如，一个时间序列对应一个股价）。这3个因素（非独立性，理论上r不等于0，和很大的m）相互平衡，使前述的计算方法在应用于典型的大数据问题时，能得到相当准确的结果。注意，概率p=P（r>a）是使用在线计算工具*stattrek*　　获得的。这个在线工具能直接提供答案——你不需要运行SAS、Python或R程序来获得数值——在你搭乘飞机时用iPad就能算出来。

## 结论

正如这个问题所表明的，类似这样的大数据项目需要合适的数据科学家来推进。你不需要有很高的技术，但是你必须用类似前述论证的方式来思考，在开发一个统计或计算机科学算法前，要先确定可能导致模型失败的原因。统计学家的知识对此会有所帮助，不过你并不需要高级的统计知识。计算机科学家的知识也有助于衡量算法的有效性，并使它们简洁、高效。MBA分析师的知识也有助于理解需要解决的问题。同时具备这3方面当然更好——这样的人确实存在，而且并不罕见。

真正的数据科学家在面对待处理的大数据集时，首先要明白这个事实，即在观察几十亿数据所蕴含的模式时，有些看起来很有价值的模式其实只是偶然现象。像这里所说的，真正的数据科学家更倾向于健壮的

指标，而不是这里列出标准的指标，尤其是在观察、比较和给几十亿的数据桶打分时。他将通过模拟来评估一个模式是否真的像起初想象的那样有价值，并且根据实际情况，使用一些能够对整体分布（根据大小调整）进行比较的指标，而不是体现狭窄、单一结论的指标，比如平均值或相关系数。

最后，我们可以通过询问人们是否同意以下3个句子来识别伪数据科学家，如果他们对其中之一或所有3个回答“是”，那么他们不是真正的数据科学家。

- 算法复杂度（用O符号表示）比数据传输所花费的时间更重要。（错：使用Hadoop与分布式集群架构时，优化数据传输和内存使用往往比算法复杂度更重要。）
- 一个人不能同时掌握统计学、计算机科学和机器学习的知识。（错：一个数据科学家应该对这几门学科都有足够的了解，同时在其中一些领域具有专业知识。）
- 商业专业知识并不重要。不管是什么商业模式，数据集就是数据集。（错：这是狭隘的看法。现在的商业模式远比过去复杂，行业专业知识是必需的。例如，如果你开发了一个验证码系统，却不知道垃圾邮件发送者已经开发出能完全绕过该验证码的系统，那么你的系统几乎没有任何价值。）

技术讲义

下面是一个例子，阐明了本节讨论的复杂相关结构，以及它们并非彼此独立的事实。假设你有3个随机变量X、Y和Z，同时corr（X, Y）=0.70、corr（X, Z）=0.80。那么corr（Y, Z）的最小值是多少？这个相关值可以为负吗？

答案是0.13，是一个正数，它必须为正。该证明是基于这一事实：相关系数矩阵是半正定的，因此它的行列式值必须大于0（详细内容参见`http://bit.ly/1evFjBJ`）。

# 数据快速流动问题

大数据的第二个挑战是，当数据流动变得越来越快，数量越来越多，并超过了数据处理能力时，该怎样应对。通常，这种挑战以两种形式出现：数据以目前的技术无法 处理，或者数据能够 被处理，但由于其规模太大而难以处理。

## 应对无法处理的数据

在某些情况下，不论使用什么处理方法和算法，由于大量数据以极快的速度不断累积，你需要删除一定量（可能每天需要删除很大一部分）的数据以后，才有可能对它进行观测。

例如，用来检测新的行星、小行星等的天文数据。数据不断快速地、大量地积累，以至于无法在云端使用大规模并行处理。也许好的采样是一种解决方案——在开始观测数据之前，先确定哪些数据可以忽略，然后仔细选择要分析的数据。

另一种可能的方案是开发更好的压缩算法，这样当你具有更强计算能力时，就可以分析以前收集、压缩和存储过的所有数据。比如，在2030年，也许你将能研究一颗遥远的超新星，它从2010年开始，连续20年内，每小时的演变情况。但在2010年，它还没有被发现，因为相关数据都被存放在一个休眠的服务器上。

一般来说，解决方案包括采样和/或压缩（在有意义的情况下）。数据科学家应该熟悉健壮的采样技术、不平衡样本和实验设计，从而设计预测模型。由于缺乏适当的统计培训，常见处理方法多是纯计算机科学的方法（使用整个数据集）。也有一种观点认为，在统计如网站每月的特定用户数量这样的数据时，采样法也是行不通的。出现这种情况，原因是缺乏统计知识，以及现实中未经训练的分析师使用样本进行此类计算时很容易出错。比如，他可能会从30天中抽取3天数据作为样本（其中包括一个低流量的星期日），然后乘以10，得到一个总数。这种解决方案对于统计页面浏览量是有效的（假定工作日和周末造成的偏差校正已经处理过），但不适用于特定用户。对于特定用户，必须仔细估计乘数，它应该远低于10，除非你只有一次性用户。

采样法已被广泛应用于较大型数据处理——例如，为人口普查局所用。压缩可以采取两种形式：对于很少访问的数据使用文本或图像压缩；对于陈旧数据，仅存储简洁的统计摘要（报表），而不必保存整个数据集。

交叉参考　如需了解在哪些情况下采样法不可行，请参阅第7章“大数据与取样”一节。

### 可处理但是处理难度大的数据

这种情况指的是数据快速（有时是实时）、大量地累积，不过所有数据都可以 用现代的、快速的、分布式的、基于MapReduce的算法或其他技术来处理。出现的问题是，数据量如此巨大，积累速度如此之快，有时数据还是非结构化的，故只能用简单算法来处理，这通常会导致不良的副作用。或者至少你的CEO会这样认为。那么你该如何处理这种情况？

我们来看几个例子，特别涉及CEO对这一问题的认知，这和业务最为相关。考虑以下的数据情况。

- 信用卡交易实时评分。
- 垃圾邮件检测。
- 定价/折扣实时优化（零售）。
- 反导弹技术（数值优化问题）。
- 高频交易。
- 依靠实时3-D数据流（视频和声音）的系统，如自动驾驶仪（大飞机靠自动驾驶仪在低海拔的、拥挤的城市上空飞行）。
- Facebook的“喜欢”、Twitter消息、Yelp的评论，内容与用户匹配（谷歌），等等。

将简单算法应用于这些数据集时，可能导致以下结果。

- 太多的误报或漏报和未检测出的虚假评论。例如，假的Twitter消息会对股票市场造成影响，关于白宫遇袭的假新闻将会导致美国国内广泛恐慌，然而Twitter可能无法实时检测到这些假消息。这种情况可以通过恰当的数据管理政策来处理，同时网站公司自己也可以处

理。（http://www.vancouversun.com/technology/personal-tech/Yelp+sues+North+Vancouver+reviewer+bogus+reviews/8958494）谷歌也是这样处理的——惩罚黑帽（国际黑客组织的名称）的搜索引擎优化（SEO），同时改进自己的算法。

- Facebook上数十亿未被检测出的虚假“喜欢”，给广告客户造成了混乱，并最终导致广告支出的下降。这对用户也有不良的影响，会使用户最终忽略这些“喜欢”，因此进一步减少已经很低的点进率，导致Facebook收入的降低。Facebook将被迫开发出新功能来替代“喜欢”，最终新功能又会再次被滥用，使用机器学习算法或其他方法无法将其检测出来，最终形成了一个微型的泡沫循环，即不断使用新的方法使其永久保持活跃，从而维持收入流。

我说的简单算法是指朴素贝叶斯分类器，或那些对离群值敏感的算法，比如巨大的决策树或线性回归用于非线性（扭曲的）数据。更复杂的算法包括多分类器组合，或者如Bagged型分类器或Boosted型分类器中的重复采样算法。其中一些可以用MapReduce快速方便地实现。

交叉参考　关于隐性决策树的讨论请参阅第5章。

有一个应考虑的问题是确定哪些数据可以被丢弃，哪些数据很少被访问，哪些领域可以忽略。（我见过被用户代理字段占据了80%空间的日志文件。为什么不使用一个用户代理查找表来减少数据量？）此外，当你开始收集大量的非结构化数据（例如文本数据）或来自多源的混合数据时，如Hadoop这样的MapReduce环境就变得很吸引人。最后，你可能会倾向于使用如本书中讨论的分类方法，对非结构化数据进行分类并添加结构；或在创建文本时加上标签；或使用存储在文件头中的元数据，然后就比较容易在传统的数据库中改造和处理数据了。但目前的趋势是厂商开发解决方案，供分析师使用SQL来访问Hadoop文件管理系统中的底层数据，而不是相反的做法。

## 解决方案

在很多情况下，人们太过依赖众包而不愿使用采样技术，这主要是因为会从大数据中获取优良样本的专家太少。在某些情况下，你仍然需要做出粒度预测（不是摘要）——例如，每幢房子的价格（Zillow）或每个邮区的天气预报。然而，即使在这些情况下，优秀的采样算法也会有所助益。

许多“评论”、“喜欢”、“推文”或“垃圾邮件”的标识是由具有不良意图、企图大规模地利用系统的用户（有时是僵尸网络运营者，有时是业务竞争对手）制造的。贪婪也是导致问题的一部分原因。我们来考虑下面一个虚拟的场景：如果虚假的“喜欢”为社交网络A带来收入，而没有被广告客户发现，社交网络A会给广告客户引入更多的虚假“喜欢”，因为社交网络认为它没有足够的相关流量来供给所有　广告客户，所以要把优质流量分配给大客户（付出更高广告费用的客户）。当客户注意到这种情况时，社交网络A或者停止这种做法，或者等着被集体诉讼——9　000万美元对于现在庞大的社交网络来说不算什么，如果谷歌和其他网站由于提供假流量而遭到集体诉讼，他们会心甘情愿地接受这一罚款数目。

然而，有一种惠及所有人的解决方案——用户，谷歌、Amazon.com、Netflix、Facebook、Twitter这样的公司，以及客户，那就是更好地利用数据科学。这并不涉及开发复杂、昂贵的统计技术，只需采用下面这些做法就可以了。

- 更好的指标。
- 不同的权重（例如，减少对众包数据的侧重）。
- 通过更好的链接分析和关联规则，检测勾结、僵尸网络和大规模的低频欺诈者。
- 更好地、更频繁地更新查找表（IP地址白名单）。

所有这些都可以在不减慢现有算法的情况下完成。下面是一个社交网络数据的例子：选用下面的做法，而不是计算“喜欢”的数目（点击“喜欢”的情况是各异的）。

- 注意那些每天点几百次“喜欢”的用户，以及被Project Honeypot（蜜罐项目）或Stop Forum Spam（阻止论坛垃圾帖）标记了的IP地址带来的“喜欢”点击。但也不要完全信任这两个网站——它们也至少部分地依赖众包（用户报告垃圾邮件），因此也可能有误报和滥用。
- 识别这样的情况：比如有200个“喜欢”却没有一条评论，或只有50个具有同样意思的“很好”或“好帖”评论——显然这是虚假“喜欢”的案例。广告客户不应为此付费，同时应查明并终止这些流量来源。
- 还需注意这样的高流量：流量很大一部分来自频繁回访、每天点两次“喜欢”的低质量用户，有红色旗标（如没有参考域，或很多模糊

的域，IP地址经常不能确定域，流量来自某些网盟的子机构等）的低质量用户。>>更多新书朋友 圈免费分享微 信xue ba987

- 使用对唯一用户更健壮的指标（更难假冒），而不是页面浏览量（但你仍然需要检测虚假用户）。
- 使用简单、健壮、快速、数据驱动（而不是模型驱动）的算法，比如隐性决策树，来给“喜欢”和评论打分。你甚至可以在不使用任何统计模型的情况下计算得分的置信区间。
- 用简单的分类概念创建或改进广告相关性算法。这适用于很多重视相关性的应用，不仅仅是广告投放。这也意味着算法能更好地检测有影响力的人（比如，你必须是PeerIndex的成员才能得到高分）、更好地检测剽窃、更好地发现朋友、更好地给求职者打分、改善归因分析系统（广告组合优化，包括提高统计模型中的长期影响因子）等，好处不胜枚举。

检测Facebook“喜欢”的问题是一大类商业技术中的一部分，它包括给点击、关键词、会员、页面印象等互联网流量打分。其目标有两个：检测欺诈（假用户、假点击、重复用户）和给每个被测量的要素（点击、关键词、用户）分配一个质量得分（代表转化率或价值）。

交叉参考　关于欺诈检测的更多介绍请参阅第6章。

当然，在Facebook广告的案例中，精明的广告主使用其他标准来衡量广告花费的投资回报率。他们使用“顾客有效期”评估和/或对自己网站的参与度。例如，喜达屋酒店（Starwood Hotels）在一个全球会议上说他们公司在量化“参与指数”，其中包括粉丝怎样分享酒店内容（传播和粉丝获得）、对酒店的光顾，以及在他们自己网站上的参与度。随着该行业日益成熟（广告主、出版社，以及用户），黑市上卖Facebook“喜欢”的将越来越少，但坏人还是会找到新的方法来利用Facebook或任何出版商推出的新功能搞不正当交易。

交叉参考　第6章的“归因模型”一节介绍了用“宏观经济”的方法来衡量广告花费的投资回报率。这种宏观经济方法在数据存储、跟踪和处理方面都便宜得多，因为你只需要观测更大的聚合数据。

注意，流量评分对实时竞价，或者对购买大量没有历史转化数据的新关键词（eBay每天都在这样做）特别有用。更好的算法和数据管理（创建像IAB（互联网广告局 ）那样根据点击和用户指标的标准）将有助于根除这些问题，例如定义哪些“喜欢”可以算数，哪些“喜欢”应该被

丢弃。

### 当数据见解流动太快时

如果有能力的话，你应该在终端用户（管理人员和决策者）接收和理解数据之前，对数据进行分析（通常使用实时的自动算法）并快速提取出见解。但在把这些见解提交给终端用户之前 ，你需要对它们做些处理——如果决策者被一大堆没有划分优先顺序的报告淹没，那可不是一件好事。

解决的办法是为报告或警报分出优先等级（使用dashboard系统）、确定谁收到什么、提供良好的可视化效果，并允许用户取消电子邮件发送的自动报告的订阅。还有一个好办法是在电子邮件里只提供摘要和简明的图表，然后附上一个完整报告的链接。一些CEO抱怨他们的电子邮箱定期被很多大型日常报表塞满。

交叉参考 关于可视化工具的信息请参阅第4章。

这种情况也会出现在机器和机器通信时，例如，eBay每天自动定价几百万个竞价关键词，并通过谷歌API（谷歌应用程序接口）把这些定价自动传送给谷歌Ad Words（谷歌在线广告服务）。太多工程师每天收到关于这种活动的详细报告，其实一份摘要就足够了。

# 大数据技术示例

在讨论了这两个大数据问题后，现在我们来看看数据科学家使用哪些技术和工具来处理数据。如需详细了解数据科学项目如何执行以及包括哪些步骤，请参阅第5章“数据科学项目的生命周期”一节。

# 大数据问题是数据科学所面临挑战的缩影

这里讨论的示例问题是创建更好的搜索工具。这是大数据和数据科学发源的问题。

你在谷歌上搜索过*mining data*（采矿数据）吗？它返回的结果会和搜索*data mining*（数据挖掘）相同。然而这是两个不同的关键词：*mining data* 通常指关于采矿的数据（如煤炭、黄金等）。如果你搜索*data about mining*（关于采矿的数据），将得到相同的结果。

然而谷歌已经是拥有最好搜索算法的搜索引擎了。可以试想一个销售产品的电子商店：它允许用户通过一个带有搜索功能的目录寻找商品，但是20%的情况下返回不相关的结果。这将会损失很多钱！事实上，如果你是一个投资者，想在Amazon.com上购买一份关于mining data的报告，结果搜索到的都是关于data mining的书，于是你什么也不会买，这种情况可能会导致Amazon.com相当大的损失。这种情况一年重复几百万次，造成的机会成本将有数十亿美元。

下面讨论这类问题的一个可行的解决方案，并简单解释怎样帮助任何企业和分析团队轻松解决这样的问题。

用户输入的任何搜索查询都将通过以下步骤进行处理。

1. 清理用户查询内容：改正错别字。

2. 然后经过修剪：复数换成单数，动词（ing形式）换成名词（因此mining变成mine），等等。

3. 去掉停止词：the、or、and、of、from、about、it、with等。例如，*IT jobs*（IT工作）变成了*jobs*（工作），*data about mining* 变成了*data mining*，然后变成*data mine*（由于步骤2）。

4. 查询内容规范化：搜索词按字母顺序重新排列（*mine data* 变成*data mine*）。

5. 使用算法和查找表来确定查询中是否包含城市名称、书籍、歌曲、产品或电影。

6. 使用其他算法检测两个词是否必须连在一起。例如，*San Francisco* 是一个词组，而不是两个，尽管它看起来像两个。

7. 检测同义词。例如，*automobile* 可能会被替换成*car* ，以减少索引的大小。

8. 对于一些跨多类别的查询，使用算法来检测用户的意图（有时需要结合用户个人资料）。

如果你搜索IT jobs（IT工作），得到的结果会是所有工作。以我为例，当我搜索“IT jobs（IT工作）”时，得到的结果是在我居住的华盛顿州伊瑟阔市的所有工作职位。谷歌能够正确地识别出我住在伊瑟阔市（通过我的IP地址），但错误地假设我对本地就业机会感兴趣，同时忘记了我只对IT工作感兴趣。我点击的第一份工作是保险代理人。

练习

问题： 去掉停止词是否应该放在步骤1，而不是步骤3？

回答： 你有没有想过这一事实，即*mine* 和*yours* 也可以是停止词？因此，在处理不当的情况下，*data mining* 先被修剪为*data mine* ，然后变成*data* ，这可能会带来与最流行的数据关键词*data entry*（数据条目）相关的搜索结果。在实践中，你先去掉停止词，然后再修剪。所以步骤3的确应该成为步骤1。

如果你使用谷歌搜索的精确匹配 来搜索“IT jobs（IT工作）”、“mining data”和“data about mining”，得到的结果稍好一点。所以谷歌似乎有一个解决方案。我认为真正的解决方法是把精确匹配而不是广泛匹配 设为默认选项。有多少人知道精确匹配？很可能不到1%。不过，这个问题的商业性要多于技术性。

我能想象谷歌如何从不佳的搜索结果中获益（将广泛匹配设置为默认选项）。事实上，这使用户更有可能点击谷歌广告，从而增加谷歌的收入。但是对于Amazon.com就行不通了。也许Amazon.com的搜索框是由谷歌支持的。如果是这样，那么Amazon.com应该把精确匹配设为默认搜索设置，而不是广泛匹配。但是我试着在Amazon.com上用精确匹配搜索“data about mining”，搜索结果并没有改善（事实上甚至更糟

了）。

显然，不论从技术角度还是商业角度来看，Amazon.com的问题更严重。我认为Amazon.com的问题比谷歌更严重是因为，Amazon.com提供的搜索结果页要少得多，因为Amazon.com的搜索范围主要是商品，比如书籍，而谷歌的搜索范围是整个互联网。

# 大规模数据集的聚类和分类

本节讨论了能以极快的速度对大数据集进行聚类的一些算法，以及这种复杂聚类结构的图形化表示。我说的“极快”是指O（n）级别的计算复杂度，有时甚至更快，比如O（n/logn）。这比优良的层次聚类算法还要快得多（参见http://en.wikipedia.org/wiki/Rg/wiki/Hierarchical_clustering），它通常是O（n2logn）。对于大数据，这意味着几百万甚至可能10亿个观测值。

聚类算法速度缓慢，而且它使用的矩阵会占据大量内存空间（观测值数量的平方——这种情况下会超过千万亿个元素）。没有内存大到足以存储这样的矩阵。然而这些矩阵是极其稀疏的。在本节中，我们将讨论解决这些问题的方法：用MapReduce提高速度、减少内存使用及优化内存。聚类是一种最基础的机器学习、统计学和大数据技术。对于非结构化文本数据，它的输出被称为分类体系 。

这项技术被用来产生代表“用户意向”的关键词。把表示相同意图的关键词（例如，在加州购买汽车保险）融入同一个聚类簇，使广告主可以购买这些预先计算好的簇，而不是单个关键词。这种做法的优点是，通过关注用户意图而得到更好的客户倾向，通过在簇中融入所有相关的关键词而扩大了搜索结果。

## 应用前景

可以创建一个关键词分类体系，对英语中所有有价值的关键词进行清理（标准化）和分类。这包括由1个、2个或3个单词 组成的约1 000万个关键词——大概是英语词典中能找到的关键词数量的300倍。这样做的目的可能是，为eBay和Amazon.com可能在谷歌上购买的所有竞价关

键词（点击付费广告活动）分类，从而更好地为它们定价。对这种应用的讨论包括以下方面。

- 聚类数以百万计的文件（例如，Amazon.com上的书）。
- 聚类各种网页，甚至整个互联网，其中包括近1亿的顶级网站和数十亿的网页。
- 确定这样的大规模聚类在什么时候是有意义的，以及怎样利用MapReduce。

注意　可能你不熟悉这个术语，单词（*token*）指用户查询词中的一个项。关键词*car insurance AIabama* 有3个单词。n-grams方法是这些单词的不同组合——例如，*insurance AIabama car* 和*AIabama insurance car* 。

## 第一部分：建立关键词分类体系

这里讲述的解决方案包括两个步骤：预处理和聚类。

### 第一步：预处理

你可以使用网络爬虫（抓取维基百科或DMOZ目录）在互联网上收集大量关键词，计算每个关键词的频率和每个关键词对的频率。一个关键词对　是指在同一个网页上找到的两个关键词，或者在同一个网页上距离很近的两个关键词。一个关键词指类似“California insurance”这样的词；一个关键词通常包含一个以上的单词，但很少超过3个。使用所有你能处理的频率，创建一个像下面这样的表（即使在关键词被清理后，通常也包含数以百万计的关键词），每个条目是1对关键词和3个数字。

A = California insurance

B = home insurance

x = 543

y = 998

z = 11

其中，

- x是关键词A在你抓取过的所有网页中出现的次数。
- y是关键词B在你抓取过的所有网页中出现的次数。
- z是A和B形成一对出现的次数（例如，它们在同一页面内被找到）。

使用MapReduce可以高效方便地建立这个“关键词对”表。注意，绝大多数关键词A和B不构成关键词对；换句话说，z=0。因此，忽略这些空条目后，你的“关键词对”表仍然是可管理的，可能只包含1亿个条目。

注意　步骤1是很多有趣的应用程序的最后一步。例如，它被用于搜索引擎技术，用来识别或推荐与其他一些关键词相关的关键词。第5章中“关键词相关的API的源代码”一节介绍了这种应用程序的例子。有趣的是，几年前，Ask曾授权一些搜索引擎使用类似的应用程序，每年收费50万美元。

第二步：聚类

要创建一个分类体系，你希望把识别出的关键词放进类似的簇。实现的方法之一是计算关键词A、B之间的差异性d（A, B）。例如，d（A, B）=z/SQRT（x*y），虽然其他的选择也是可能的。注意，上述公式里分母可以防止极受欢迎的关键词（例如，“free”）与所有其他的关键词距离太近，并支配整个关键词关系结构。事实上，它有利于更好的关键词结合，如*lemon* 与*law* 或*pie* 结合，而不是*lemon* 与*free* 结合。

d（A, B）的值越大，关键词A和B之间的距离越近。现在面临的挑战是使用任何类型的差异性度量，对上述“关键词对”表执行某种聚类——例如，层级聚类。第二部分中提出了解决方案和供参考的替代方案。

## 第二部分：使用快速聚类算法

虽然下面讲述的算法应用于关键词聚类，它也可以直接适用于其他情况。假设你有n=10 000 000个特定关键词，以及m=100 000 000个关键词对{A, B}，同时d（A, B）>0。也就是说，每个关键词有平均r=10个相

关的关键词。

该算法以增量模式逐步运行几轮（5、6轮），如下。

- 初始化（**0**轮）：小数据（或种子）步骤。　选择1万个种子关键词，并且创建比如100个类别及1个哈希表`$hash` ，其中哈希表的键是1万个种子关键词，值是该关键词分配到的类别的列表。

```
“$hash{"cheap car insurance"} = {"automotive","finance"}
```

最初1万个种子关键词的选择是非常重要的。在这一主题得到更多研究之前，我建议根据关联的数目来选择前1万个关键词——也就是说，带有很多B的关键词A，同时d（A, B）>0。这将加快算法的收敛速度。

- 第**1**回合：大数据步骤。 从头至尾浏览m个关键词对的表。当你在一对{A, B}中，发现比如$hash{A}存在而$hash{B}不存在时，执行以下操作。

- $hash{B} = $hash{A}
- $weight{B} = d（A, B）

当你找到一对{A, B}，其中A和B都存在`$hash` 中，执行以下操作。

- 如果$d（A, B） > $weight（B），那么{ $hash{B} = $hash{A}; $weight{B} = $d（A,} ; # B的类别名设置成B的类别名
- —If $d（A, B） > $weight（A），那么{ $hash{A} = $hash{B}; $weight{A} =$d（A,} ; # A的类别名设置成B的类别名

- 第**2**轮： 重复第1轮。（`$hash` 和`$weight` 被保存在内存中，并在随后的每一轮中继续增长。）
- 第**3**轮： 重复第1轮。

- 第**4**轮： 重复第1轮。
- 第**5**轮： 重复第1轮。

计算复杂度是q*m=O（n），q代表轮数。这样比常规的好的聚类算法还要快n=10 000 000倍。然而，由于`$hash` 和`$weight` 在随后的每一个回合中都会增长，对这些哈希表的访问会使速度略微减慢至O（nlogn）。

m个词对的大表，对其预排序是否会有所帮助（根据d（A, B）排序）？这样就不需要再分类，可以大大减少哈希表访问次数，但排序过程的复杂度也是O（nlogn），所以你不会得到更多收益。注意，可以使用MapReduce高效地执行排序。在这种情况下，Reduce步骤包括合并很多已排序的小表。

这种聚类算法似乎（初看起来）很容易使用MapReduce实现。不过，因为这个大表只有1亿个条目，它可能也适合用在RAM中。

你可以将最重要的m/logn个条目（基于总量和d（A, B））保留在大表中，删除其余的条目，从而改善计算复杂度。在实践中，删除大表的65%（仅那些从关键词分布角度来看极长尾的，而不是一般长尾的）对性能几乎没有影响——你会有大量未分类关键词，但在总数上，这些关键词可能只占不到0.1%。

替代算法

或者，你可以使用Tarjan强连通分量算法进行聚类。执行方法是，首先把距离二进制化：如果d（A, B）高于预定阈值，则将其设为1；否则设为0。这是一个图论算法，其中每个关键词代表一个节点，每对d（A, B）=1的关键词代表一个边。该算法的计算复杂度是O（n+m），其中n是关键词的数目，m是对（边）的数目。要利用该算法，你可能需要把“关键词对”大表存入一个图形数据库（一种NoSQL数据库）。

其他注意事项

可视化： 怎样用一个很好的图来展现这些关键词及由d（A, B）决定的簇结构？一个3 000×3 000像素的图能放下1 000万个关键词。如果你对图形表示法感兴趣，可以看看Fruchterman和Reingold算法，它被广泛应用于展示簇结构。注意它的计算复杂度是O（$n^3$ ），所以你需要极

大地改进它，才能用于这个关键词聚类应用程序——包括图形展示。图形化表示形式可以是一个由几百万像素点组成的栅格图像，类似于热点图，其中颜色代表类别，当你指向一个像素，就会显示一个关键词值（而不是有几十个节点的矢量图像）。相邻像素代表强关联的关键词。

需求：　你真的需要这种聚类算法吗？大多数时候，需要解决的问题简单得多（例如，哪些关键词与关键词A相关），或者你已经有一个分类体系，想扩展或改进它。在这些情况下，完全聚类是不必要的。不过知道有高效简单的算法存在，在必要时可以选用是件好事。

科学类型：这些问题是统计科学、计算机科学，还是数据科学？

数据集：　我自己的关键词对大表，包括d（A, B）值可供大家下载，将在下一节介绍。另外，在`https://ak.quantcast.com/quantcast-top-million.zip` 网站可以免费下载DMOZ数据（100万分类网址），这是一个好的起点，可以从数据集本身，或者用分布式架构从所有网址和相关链接里提取数以百万计的关键词和类别。Quantcast也提供一个免费的含有100万个顶级域名的列表。最后，搜索引擎的查询日志是一个好的关键词数据来源。这些查询日志对于前面讨论的第一步的预处理非常有用。

# 1亿行的Excel

和很多数据科学家一样，我在职业生涯中经常用到Excel，它的确有一些强大的功能。最优秀的功能大概就是它能够帮助我们设计、测试和实时更新模型——只要改变几个核心参数值，所有单元格（成千上万个）和表格立即会被更新。

注意　如果想看只需点击一次就可更新所有单元格和图表的交互式电子表格实例，可以在此下载电子表格：`http://www.analyticbridge.com/profiles/blogs/three-classes-of-metrics-centrality-volatility-and-bumpiness` 。

Excel其他一些不错的功能包括能够通过互联网连接到数据库或微软的API（例如，连接到Bing　API），从中提取有用的信息，并把它们

总结为多维数据集和数据透视表。虽然多维数据集和数据透视表有种很老式的SQL关系数据库的感觉，但在很多情况下它们仍然是有用的。

Excel最主要的缺点是缓慢。它缓慢的方式令人意外。如果你对同一列中的50万个观测值进行排序，速度实际上会很快。但如果你想同时给A和B两列排序，其中B是log（A）的值。因此B的每一行都包含一个公式。这就会大大减慢排序。如果单独给A排序，B保持原状，速度会快得多。B中的公式将会迅速、自动并且正确地更新所有值。

一般而言，任何涉及重复计算的操作，例如，跨多个单元格并由依赖关系关联的20万+行，在以下情况下会执行得很缓慢。

- 其中一列的每个单元格中包含类似VLOOKUP的函数。
- SORT或其他次线性进程是必需的（次线性指O（nlogn）或者更差的进程）。

有一个简单高效（但不太好看）的方式可以避开这个问题，但奇怪的是Excel未把它作为内置功能让用户可以方便地使用。

- 用硬编码值替换VLOOKUP公式，对硬编码值执行更新，然后还原公式。
- 仅对最小数目的必要列执行SORT操作，然后更新包含公式的列中单元格。

还有一个奇怪之处是VLOOKUP的缓慢。在Perl中，我常用它来连接一个比如300万行的数据集和一个20万条目的查找表，速度很快。在Excel中，这样的数据集和查找表将存储在同一电子表格里的两个单独的工作表里，要执行这一“连接”（如果可能的话）会花费几天的时间。（不可能实现。）可能是因为Excel间接执行了一个完全连接，因此指数级地减慢了操作。但几乎没人会对大型数据集进行完全连接。这是一个可以显著改进的地方。

最后，利用云将进一步加快计算及处理远大于100万行的数据集。微软可以允许用户将数据导出到某个云，操作方式可以是一次简单的点击（比如，只需单击“导出到云”按钮）。然后它会模拟你电脑上本地版Excel耗时的操作，相当于将本地Excel电子表格作为虚拟电子表格使

用。完成后，单击“检索云”，然后获得更新的电子表格。“检索云”将执行以下操作。

1. 通过互联网把你的Excel公式发送到云。

2. 把公式应用到你的数据云版本，必要时利用MapReduce。

3. 将处理后的数据返回到你电脑上的本地电子表格里。

另一个缓慢而痛苦的过程是，当你需要将公式应用于包含50万个单元格的列时。幸运的是，有一个窍门，假设你想把产品A和B存入C列，可以执行以下操作。

1. 选择整个C列。

2. 输入公式：=A1*B1。

3. 同时按下Ctrl键和Enter键。

我希望有更容易的办法，就像==A1*A1（双等于号表示该公式应用于整列，而不是仅一个单元格）。这是Excel不够用户友好的另一个例子。

很多时候，有一些隐藏的方法可以有效地解决问题。在第5章“用Excel进行回归分析”一节中你会看到另一个例子，教你如何编写一个返回多个单元格的公式——尤其是使用LINEST，它会返回与一个线性回归相关的回归系数。是的，这可以在基本的Excel中完成，无须借助插件。

如果你有兴趣创建一个有100万个值的列（例如，测试Excel的某些计算速度），下面是操作方法（这会是一个很好的工作面试问题）。

1. 从200个单元格开始。

2. 复制这些单元格，得到400个单元格。

3. 复制这些单元格，得到800个单元格。

这个过程再迭代10次，你会得到80万个单元格。这比原始方法（用

鼠标向下拖）快得多，而如果你初始的200个数字包含公式=RAND()，最后你会得到100万个伪随机数，虽然质量差。

最后，可以使用Perl、Python、R、C、Sort、Grep等其他语言进行数据汇总，然后再把资料输入Excel。但如果Excel具有刚才讨论的功能，在它内部就能完成更多操作。

### 如何升级**Excel**将其用于大数据

你应该使用Microsoft的PowerPivot插件来处理大型数据集。如果你有Excel 2010，你可以从`powerpivot.com` 获得和安装这一插件。如果你有Office 2013，PowerPivot就包含在Office Professional Plus版里（它默认是禁用的，你需要启用它）。如果你想用Excel做预测分析，你应该看看Predixion Insight （`https://www.predixionsoftware.com/` ）。它是Microsoft做数据挖掘插件的同一团队开发的。Predixion Insight的开发者版本是免费的。Predixion Insight能和PowerPivot很好地集成。

最新版本的PowerPivot可以扩展外部数据连接器，比如OData、Azure marketplace等。PowerPivot就像在Excel内部有一个SQL Server分析服务器的概念证明版本。

同时，1010data（`https://www.1010data.com/` ）开发了Trillion Row Spreadsheet，使企业可以使用电子表格界面，通过一个基于云的GUI来执行数据发现和分析数万亿行数据。除了基本的电子表格计算，Trillion Row Spreadsheet还能够运行先进的统计分析和创建机器学习算法，全部使用常用的电子表格范式。最棒的是，完成这些不需要编写MapReduce代码。

另一个解决方案是使用谷歌的电子表格和BigQuery。参见`https://developers.google.com/apps-script/advanced/bigquery-tutorial` 的教程。

## **MapReduce**不能做什么

现在我们来考虑一大类不能使用MapReduce的大数据问题——至少不能直接使用——以及一个相当简单的、使用分析和统计方法的解决方

案。

MapReduce是这样一种技术，它将大数据集拆分成许多较小的部分，在不同的服务器或电脑上分别（但同时）处理每个小数据集，然后收集和汇总所有子过程的结果，产生最终的答案。如果你使用这种分布式架构来处理大数据集，使用1 000个服务器并把主过程分裂为1 000个子过程，会比传统的（非分布式的）设计快1 000倍。

当变量或观测值被一个接一个处理时，MapReduce工作得很好。例如，你要分析1TB文本数据，想计算出其中可找到的所有关键词的频率。你可以把1TB分成1 000个数据集，每个1GB。然后生成1 000个关键词频率表（每个子集一个），再汇总它们，产生最终的表。

然而，当你需要处理联合的变量或数据集时（如2×2或3×3联合变量），MapReduce并不比非分布式架构好。你必须找到一个更高明的解决方案。

# 问题

假设你的数据集由n个观测值和k个变量组成。例如，k个变量代表k个不同股票代码或索引（假设k=10 000），观测值代表在n个不同时间点观测到的股票价格信号（升/降）。你想找到高相关性（理想情况下具有时间差，可以从中赢利）——例如，如果谷歌股价今天上涨，Facebook明天也会上涨。

要解决这个问题，你必须计算k*（k–1）/2个相关值，尽管你只有k=10 000个股票代码。你不能把这1万个股票代码分成1 000个簇，每个簇包含10个股票代码，然后使用MapReduce。你要计算的相关值在大多数情况下是两个股票代码分处于两个不同的簇（因为要计算的相关值远多于簇）。在这种情况下，这些跨簇计算使MapReduce无法施展。如果你把*correlation*（相关性）这个词替换成任何其他函数，比如f，然后计算两个变量，而不是一个，会出现同样的问题。这就是为什么我要说这里讨论的是一大类MapReduce派不上用场的问题。

# 3种解决方案

这个问题的3种可能解决方案是采样、分箱和经典的数据减少技术。

## 解决方案一：采样

不需要计算所有的交叉相关值，只计算其中的一小部分。选择m对随机变量——例如，m=0.001 * k *（k−1）/2——只计算这m对的相关值。一个明智的策略是，从所有可能组合中的一小部分开始入手，然后增加变量对的数量，直到最高（最显著）的相关值几乎不再增长。或者你可以在计算了1 000个随机选择的种子变量对的相关性后，使用模拟退火方法来决定保留哪些变量，以及添加哪些来形成新的配对。

交叉参考 第6章“组合优化自动快速的特征选择”一节中讨论了半组合策略，它不仅能够处理2×2组合（如本节讨论的相关性问题），还能处理3×3、4×4等组合，以找到高质量的多变量向量（根据预测能力），可以用于统计评分或欺诈检测方面。

## 解决方案二：分箱

如果你能以合理的方式将你的变量分箱，并且如果n很小（比如，5或更小），那么你可以预先计算所有可能的相关值，把它们存入一个查找表。在我们的例子中，变量已经被分箱：我们处理的是信号（上涨或下跌），而不是像价格deltas那样实际的、连续的指标。如果n=5，则有最多512个可能的成对的时间序列。举个成对的例子，比如{（升、升、降、升、降），（升、升、升、降、降）}，其中前5个值对应一只特定的股票，后5个值对应另一只股票。因此，可以容易地预先计算所有512个相关值。你仍需浏览所有k*（l–1）/2对股票来解决你的问题，但现在速度快得多，因为你是从查找表里得到每对的相关值——不需要计算，只需从哈希表或者包含512个单元格的数组里获得一个值。

注意对于二元变量，计算相关值的数学公式会显著简化，使用简化公式计算相关系数可能比使用查找表来访问512个预先计算的二元变量相关系数更快。然而，不论是计算相关系数还是更复杂的函数f，这个原理都是起作用的。

### 解决方案三：经典的数据减少技术

也可以使用传统的数据减少技术，例如向前或向后的步进方式，（依次）在一个时间点（或可能两个）添加或移除一个变量。选择添加的变量应能最大限度地提高所产生的熵。熵可以用各种方式计算。简言之，如果你有两个来自相同大数据的数据子集，比如下面这些。

- 数据集A有100个变量，压缩后是1.23 GB。
- 数据集B有500个变量，包括来自数据集A的100个变量，压缩后是1.25 GB。

那么你可以说数据集B中其他400个变量（例如，股票代码）不带来任何额外的预测能力，因此可被忽略。换句话说，由数据集B获得的提升是如此小，甚至可能小于这些股票价格信号的固有噪声。

注意　结合使用这3个策略会是一个有趣的解决方案。还要确保找到的高相关性并不是前面讨论过的大数据“诅咒”所造成的假象。

# 结论：何时使用**MapReduce**

像交易数据或日志数据这样按顺序产生的数据是好的候选对象——不需要计算涉及大量远距离点间的相互作用（例如，依靠局部天气模式产生的全局天气模式就不可用）。评分算法是极好的候选案例。处理大量的非结构化文本数据、协同过滤、网络爬虫和搜索引擎是另外一些好的候选案例。

在涉及处理$n^2$ （或$n^3$ ）个点，其中n是10亿级的，同时确实需要进行$n^2$ 计算的情况下，使用MapReduce的难度比较大。这个$n^2$ 问题也出现在本章前一节“大规模数据集的聚类和分类”中，但那里使用的数据结构以某种方式设法摆脱了$n^2$ 矩阵。尽管这个问题从并行计算的角度初看显得比较棘手，但可能因为大数据的稀疏，使MapReduce仍然可用。对于谷歌的页面排名算法来说也是这样，需要计算$n^2$ 矩阵的逆矩阵，n等于互联网上的网页的数量（n大于1 000亿）。这可以通过巧妙设计的算法结合MapReduce来完成。

# 沟通问题

本章前面部分“大数据问题是数据科学所面临挑战的缩影”一节中讨论的例子，说明了这个问题有时并不是技术上的困难，而是由于企业各自为营、沟通不良，以及团队合作不佳造成的。

有几个方面使得匹配搜索结果和用户查询这个搜索引擎问题难以修正。对于决策者、CTO（首席技术官）或CEO（首席执行官）来说，注意并理解这个问题，以及评估其机会成本并不难（只需运行200个高频词随机搜索查询，然后发现并理解看有多少返回了不相关的结果），然而发现并理解分析团队和商业人员之间沟通不良的问题确较难，尤其是去发现哪些地方仍存在问题。

有时，多个分析团队各自为政地有时多个分析团队在各自为营地工作着——计算机科学家、统计学家、工程师——在意见不同时积极捍卫各自的领域。决策者最终听到的是很多噪声和技术细节，而不知道该怎样着手解决这个问题、情况有多复杂、解决问题需要花多少钱，以及由谁来解决它。

在之前的例子中，步骤1～8涉及的人都属于不同的团队，或者是独立工作的人。决策者甚至不知道该和谁　谈论这个问题。有时候，使用数据科学术语与非数据科学人士交流可能使问题变得更糟。例如，数据科学家会使用n-gram这样的术语来讨论步骤4，并提到计算复杂度>>更多新书朋友 圈免费分 享 微 信xueba987 是妨碍步骤4被改进的瓶颈。他们会告诉决策者，某个查询包含3个单词并有6个n-gram，另一个查询包含4个单词，有24个n-gram，还有一个查询包含6个单词，并有720个n-gram，而如果他们需要考虑关键词索引里的所有单词组合，索引将会爆炸！然后决策者可能会说：“好，我们多买一些服务器，再弄一个集群架构（Hadoop）。”

但是当科学家声称关键词索引会爆炸时，他们是错误的。因为他们的思考方式是科学家的，而不是商人的。在实践中，有多少个查询会包含4个以上的单词？很少。同时，对于包含3个或4个单词的查询，只有很少几个n-gram有像样的流量。绝大多数可以用主导的n-gram取代（根据流量）。例如，人们有时会搜索“*car insurance* （汽车保险）”，但很少会搜索“*insurance car* （保险汽车）”。所以对这两个查询只设置一个

条目是可行的。但是对于“*data mining* （数据挖掘）”和“*mining data* （采矿数据）”，有必要保留两个单独的条目。所以实践中并不会出现一个具有1 000个系数且呈爆炸性规模的表格，事实上我们会看到大约50%的规模增长，这完全是可管理的。当用户搜索“*mining data* （采矿数据）”时，算法首先应检查这个特定的n-gram是否有一个对应的条目（应该设置一个），如果没有，再根据字母顺序重新排列单词——也就是说，在关键词索引中寻找“*data mining* （数据挖掘）”。

小提示 关键词索引里的关键词长度应该不超过4个单词。

另一个解决方法和步骤2有关。有时，一些像“*mining* ”或“*booking* ”这样的词是不宜被修剪的（把它们替换成*mine* 或*book* ）。你需要有一个查找表，包括前1万个以“ing”结尾不宜被修剪的词。

这同样适用于步骤3：创建一个查找表，包括前1万个含有不宜删除的停止词的组合。例如，该查找表中包括“*IT job* （IT工作）”，附属于停止词*it* 。

小提示 作为一条经验法则，遇到只有两个单词的查询词时，不要删除其中的停止词。

要实现有效沟通，挑战之一是如何创建一个最佳的团队结构。也许最重要的是要有能跨多个团队工作的高级专业人士，有点像管理顾问。这个人应该对问题有全局的把握，具有一定技术和 商业导向（比如具有较强分析技术的MBA），能够架起多个数据科学团体或个人之间的桥梁。

交叉参考 第5章“数据科学项目的生命周期”一节，解释了一个数据科学项目的典型过程。

这个示例问题和解决方案正是数据科学地位得以形成的原因：它能架起统计学家、商业人士、计算机科学家和工程师之间的桥梁，并经常被用来处理像该例中的大数据问题。注意，要开发像本节中介绍的解决方案，数据科学家必须有扎实的行业知识——在本例中，即指搜索技术。

这一问题集中体现了什么是数据科学的真谛及其面临的挑战。像这样的案例研究是典型的，同时该问题并不只与搜索引擎有关。仅仅雇用

合适的统计学家或数据挖掘专家并不能解决该问题。事实上，如果你读了这个解决方案，就不需要雇用任何人，也许反而需要解雇一些人。这是一个人事/组织的问题。

# 数据科学：统计学的终结

本节首先列举了一些传统统计学被滥用并导致错误结论的例子，特别是在应用于现代数据时。现代数据的特点通常是更加的非结构化、更大规模，需要合并来自各种源的不完全兼容的数据集。然后介绍现代统计学怎样帮助数据科学更好地发展。

# **8**种最差的预测建模技术

以下技术大多数已经发展了较长时间（在过去10年中），其中大部分缺点已经得到弥补，因此更新后的技术已经远不同于其原始版本，性能也大为提高。但通常情况下，这些有弊端的技术仍然被广泛使用。

1. 线性回归　　依靠一般标准、异方差性和其他假设，不能捕获高度非线性的混沌模式。它倾向于过度拟合、参数难以解读，并且在独立变量高度相关时非常不稳定。修正方法包括减少变量、进行变量变换，以及使用约束回归（例如，岭回归或Lasso回归）。

2. 传统决策树　　大而不稳定，无法解读，而且容易过度拟合。修正方法包括使用多个小决策树，而不是使用一个大决策树。

3. 线性判别分析法　　用于监督聚类。这是一个很差的技术，因为它假定簇没有重叠并且被超平面完全分开。在实践中从来没有这样的情况。应改用密度估计技术。

4. **K**-均值聚类　　倾向于产生环形簇，不容易处理不符合高斯混合分布的数据点。

5. 神经网络　不容易解读，不稳定，容易过度拟合。

6. 最大似然估计　　要求你的数据符合预先规定的概率分布。它不是数据驱动的，很多时候预先指定的高斯分布和你的数据很不适合。

7. 高维密度估计　　常受到维度的影响。修正方法之一是使用非参数核密度估计与自适应的带宽。

8. 朴素贝叶斯　　用于如欺诈检测、垃圾邮件检测和评分。它们假定变量是独立的，但如果不是，就会惨遭失败。在进行欺诈检测和垃圾邮件检测时，变量（有时被称为规则）是高度相关的。修正方法之一是将变量分为独立的变量簇，每个簇包含高度相关的变量。然后将朴素贝叶斯应用于簇，或者使用数据减少技术。不好的文本挖掘技术（例如，垃圾邮件检测中的基本“单词”规则）和朴素贝叶斯结合会产生非常可怕的结果，带来很多误报和漏报。

小提示　　永远记住，在测试模型时要使用可靠的交叉验证技术。

这些不好的模型仍然被广泛使用的原因如下。

- 很多大学课程使用过时的教材，因此很多学生没有接触过更好的数据科学技术。
- 人们使用黑箱统计软件，不知道其局限性和缺点，或者不会正确调整参数和优化各种节点，或者不了解该软件实际生成什么结果。
- 政府强制监管行业（制药业和银行业——见银行的Basel　III规定）为统计合规而使用30年前的SAS程序。例如，虽然SAS有更好的评分方法可用于信用评分，却被当局武断地拒绝使用。提交给FDA的临床试验分析也是一样，SAS是强制使用的合规软件，使FDA可以从制药公司复制分析和结果。
- 现代数据集比当初开发这些技术时使用的数据集复杂得多，也极不相同。简而言之，这些技术不是为现代数据集开发的。
- 没有完美的适用于所有数据集的统计技术，但有很多差的技术。

此外，由于不良的交叉验证高估了未来数据的预期提升值、实际的准确度，或训练集之外的真实投资回报率，从而使差的模型也能达标通过。良好的交叉验证包括以下方面。

- 将训练集拆分成多个子集（测试和控制子集）。

- 在控制集中放入不同类型的客户和比测试集更新的数据。
- 检查控制集中预测值的质量。
- 计算个别错误（错误定义为，比如真实值减去预测值　）的置信区间，以确保错误足够小而且波动不大（在所有控制集上有小方差）。

# 把计算机科学、统计学和行业专业知识结合在一起

现在我们用一个现实生活中的问题来说明如何将统计学、计算机科学和行业专业知识融合起来，为一个现代大数据问题提供数据科学解决方案。

考虑阅读一篇由几位科学家撰写的文章，内容是关于如何根据用户查询和广告文本预测广告点击数的。文章参阅网址`https://static.google-usercontent.com/external_content/untrusted_dlcp/research.google.c`，这篇文章由一个科学家团队撰写，主要讨论该模型（几十亿特征）中的大量指标、逻辑回归（一种统计技术）的应用和用于处理逻辑回归（找到最佳回归系数）的优化技术（数值分析梯度法）。如你所料，他们详细讨论了特征空间的稀疏及怎样利用稀疏性来设计一个高效的算法。

这一切听起来很棒，而它当然是机器学习方面一个正确、优秀、有趣、教科书般的应用实例。事实上，这是计算机科学。不过，对于我读过的几乎所有“纯科学”的东西，我有两条批评意见，意思是相同的。它就像用核武器打一只苍蝇而同时没有意识到自己的做法，而行业专家设计的基本方法也没有什么提升——本案例中指同时在广告技术、商业管理和统计学方面有深厚专业知识的专家。虽然谷歌及其算法比很多公司做得好，但如果他们减少对科学方面的侧重而增加对行业专业知识的重视，则会更轻松且做得更好。这正是数据科学努力要弥合的沟堑。

以下是谷歌开发的广告预测技术的一个例子。

## 要实现的目标是什么

这是所有数据科学家都应该想到的第一个问题。在这个案例中，谷歌的目标可能是通过为广告主提供最大化的点击量而增加自己的收入。大概这篇研究论文的目标是为了在谷歌内部使用。或者目标也可能是为了帮助广告主——只要能转化成销量，他们对点击量的需求就永无止境。

如果论文是为了谷歌内部使用，它应讨论通过提高点进率（CTR）来增加谷歌收入只在短期内有效这一事实。对于很多广告主，过度提高CTR（由出版商提供，本例中即谷歌）最终会导致投资回报率降低。这种情况我已经经历过无数次。文章里至少应讨论一下长期目标（提高转化率）和短期目标（提高CTR）。在解决任何业务优化问题时，这两者都是必需的，不能分开考虑。

如果此论文是服务于广告主的，则其忽略了关键的一点：大多数广告主（那些对真人产生的真实流量感兴趣的广告主）感兴趣的是转化率。对广告主而言，通过修改广告措辞和添加一些关于其活动的关键词来产生大量点击数是轻而易举的事——但这会降低投资回报率。例外的情况是，广告主是出版商，采用按曝光收费模式（来自谷歌的一次点击便是其客户的一次曝光）向其下游广告客户收费——简言之，一切以点击量为准。

注意　此处讨论的例子可能是你能遇到的最极端情况之一。但这里要说明的关键理念是，统计学方法不应被低估。在许多体系中，解决方案是使用具有数十亿条规则的系统，这些规则能够涵盖所有可能包含的内容并处理整个大数据集。重点是存在成本更低的替代方案。

## 需要用核武器打苍蝇吗

如果一个体系使用了几十亿个特征，但其中大多数几乎从未触发过，这完全没有意义。如何处理这些特征的相互依赖性？在5　000亿观测值（点击）中，有99.9%的特征触发不超过3次，它们能有什么统计意义呢？当然，你可以进行一些特征组合和聚类（从计算角度来说，这是一种成本相当高的技术），但这种特征聚合问题在论文中并未提及。

此外，这些特征绝大多数可能是通过一种特征生成算法自动创建的。这是该系统目前最吸引人的部分，但论文中并未加以论述。这是一

个组合优化问题，着眼于一组基本指标（如日志文件字段）中的所有关系（比率、产品、对数变换，以及IP分类等其他映射），旨在发现具有预测能力的特征。一些特征也可由观测这些数据的分析人员批量创建。这种数十亿个特征的集合可能会遗漏两三个可使算法优越得多的核心（但不明显）特征。论文中没有提到谷歌算法中使用的任何特征。

只需使用该领域专家精心选择的几个特征（一个特征是一个变量），即可解决这个广告点击量预测问题。以下是一些不能通过自动特征生成算法提炼，但在预测广告点击量时建议使用的特征。

- 关键词类别：将关键词类别和为文本广告分配的类别匹配起来。这意味着你有算法可以匹配用户查询关键词和文本广告分配的类别。你用另一种算法来实现用户查询标准化以及区分，比如mining data和data mining。这也意味着你有一个包含500个类别、10万个子类别及300万个子子类别的列表，足以涵盖99.99%的商业用户查询（广告主为此竞价）。注意，关键词可以包含两三个词语（例如“*car insurance Alabama* ”（汽车保险阿拉巴马州））和两个类别（例如“保险 ”和“区域 ”）。
- 特殊关键词：从一个文本广告中创建或获取一列特殊关键词（例如*2013* 、*new* 、*free* 、*best* ）。
- 罕见子类别：识别具有同一罕见子类别的文本广告和用户查询。这将增加用户点击的概率。
- 广告主类型：检查是否存在高CTR（来自未转化的点击量）。高CTR表示这是虚假广告主，因为真实广告主的CTR通常较低。
- 广告浏览次数：识别一个用户浏览广告的次数（第一次、第二次还是第三次？）。优秀的广告最初会发挥良好作用，但如果文本广告的内容始终保持一成不变，它就会失去作用，CTR也会下降。
- 热门事件：识别与同一热门事件相关的广告和用户查询。
- 值得信任的域名：编制或获得值得信任和受推崇的域名清单。你需要有对域名评分并按类别细分的算法。
- 特殊字符：编制在广告中出现的特殊字符和大写字母清单。

当然，最佳解决方案是将这些特征与自动化机器学习算法检测和分析出的重要特征混合起来。

## 智能统计方法的适用情况

注意到特征空间的高稀疏度后，我设计了隐性决策树来解决这类问题。你是否需要使用梯度算法的逻辑回归？当数据杂乱无章时，你是否需要精确的解决方案？我敢说，你只需使用20个精心挑选的特征，就可以做出高质量的预测，这也是数据科学家可以发挥作用的地方，即运用统计学知识创建一个运行速度快1　000倍的系统，使用的计算机资源更少，但可提供类似或更好的结果。你不需要使用如健壮逻辑回归等标准方法。我在很长时间里一直使用无模型统计方法，并取得了非常满意的结果；同样，我还计算了无模型置信区间，这将在第5章中讨论。

统计学可以发挥作用的另一个方面是——如果你喜欢处理几十亿个特征——尝试识别具有预测能力的特征。谷歌使用的10亿特征中大多数都没有预测能力；实际上，前述的论文中根本没有讨论预测能力。有时，将单独使用时并没有预测能力的两个特征结合在一起便会产生预测能力。例如，将国家（美国与英国）和时间结合在一起时，便会产生大得多的预测能力。统计科学有助于界定预测能力及评估什么是有意义的情况。

最后，如果你有几十亿个特征，你会发现一些看似具有预测能力的特征实际上并不具备预测能力。更糟的是，这些杂乱的特征可能掩盖了那些真正具有预测能力的特征，使你的系统容易发生错误，从而导致混乱的预测。这个问题应使用特定的统计分析来解决，而不应使用目前大学课程所教授的那种统计方法。

注意　此问题的原因（及解决方法）在一篇关于大数据“诅咒”的文章中有详细说明，下载网址是`http://www.analyticbridge.com/profiles/blogs/the-curse-of-big-data`。

# 大数据生态系统

一本关于大数据的书至少应该简要介绍大数据生态系统的组成部分，否则就是不完整的。大数据生态系统由以下产品和服务（其部分示例也在下文列出）组成。

- 硬件提供商。
- 云服务提供商，包括公共、私有及混合云。例如EMC、Cloudera及Amazon Web Service。安全公司也纳入此类别中，以保护你的数据。
- 数据集成供应商。
- MapReduce/Hadoop环境。
- NoSQL、NewSQL图形、内存分析及数据库方面的数据库供应商。例如Teradata和Pivotal。
- 商业智能和仪表盘，例如BIRT。
- 虚拟化工具，例如Tableau和TIBCO。
- 数据科学及分析工具，包括SAS、StatSoft、Revolution Analytics、R及Python数据分析库。

这些产品和服务既可以是开源的，也可以是企业解决方案。有关大数据生态系统及相关主题的更多信息，请参阅：

- `http://www.bigdatanews.com/profiles/blogs/big-data-ecosystem`
- `http://www.bigdatanews.com/profiles/blogs/42-big-data-startup`
- `http://www.bigdatanews.com/profiles/blogs/big-data-vendor-revenue-and-market-forecast-2012-2017`

# 总结

本章继续论述了使数据科学成为一门新学科的依据，说明了大数据与10年前所看到的数据之间的差异，解释了盲目地对此类数据运用标准的统计学方法为何会失败，并指出一些滥用统计学和使用统计学不够的问题，对此应该如何调查和解决。

本章讨论了大数据“诅咒”和快速流动的数据问题，然后列举并评估了一些实际生活中使用大数据技术的例子。此外论述了大数据环境中的沟通问题，以及在大数据环境中如何及为何要用到统计学。最后简介了大数据生态环境的组成部分。

第3章“成为一名数据科学家”将讨论什么是数据科学家，数据科学家可以选择的不同道路，以及如何获得适当的培训，以便找到一份数据科学家的工作。

# 第3章
# 成为一名数据科学家

大数据和数据科学已经形成一门学科并将持续发展。本章将探讨数据科学家的主要特征、不同类型的数据科学家，以及怎样成为一名数据科学家，并介绍现有的相关培训项目，以及不同类型的数据科学家职业道路。

## 数据科学家的主要特征

你可能已经注意到数据科学家应有的几个主要特征。本节将讨论这些特征，以及他们应该具有或获取的专业知识，并讨论横向知识的重要性。最后附上对数据科学家群体的统计数据。

## 数据科学家的职能

数据科学家不是统计学家，不是数据分析师，不是计算机科学家，不是软件工程师，也不是业务分析师。他们对这些领域都有一定的了解，但同时还具有这些领域之外的知识。

在过去的15年里，统计学家和数据科学家之间的距离越来越远的原因之一是，统计学家，即发表理论性文章（有时不是建立在数据分析基础上的）并且培训统计学家的那些人……已不再是统计学家。此外，很多统计学家认为数据科学是关于分析数据的学科，但事实上它的范畴不止于此。数据科学还包括实现数据自动处理的算法，从而提供对数据自动化的预测和操作，例如下面这些领域。

- 自动竞价系统。
- （实时）评估美国所有房屋的价值（Zillow.com）。
- 高频交易。
- 将一条谷歌广告与一个用户及一个网页匹配起来，使转化率最大化。
- 使每条谷歌搜索都能返回高相关性的结果。
- Amazon.com或者Facebook上的书籍和朋友推荐。
- 税务欺诈检测和恐怖主义检测。
- 为所有信用卡交易评分（欺诈检测）。
- 计算化学，模拟新分子用于癌症治疗。
- 传染病的早期检测。
- 分析NASA（美国太空总署）图像，寻找新的行星或小行星。
- 天气预报。
- 自动驾驶（飞机和汽车）。
- 用于所有旅馆房间的客户定制化（实时）定价系统。

所有这些领域皆涉及统计科学和百万兆的数据。从事这方面工作的人通常不把自己称为统计学家，他们称自己为数据科学家。随着时间的推移，统计学家也会逐渐开始使用大数据和现代应用程序，这时数据科学和统计学之间的差距就会越来越小。

同时，数据科学家不是统计学家，因为统计学家懂得很多数据科学家工作中不需要的知识：广义逆矩阵、特征值、随机微分方程，等等。当数据科学家需要用到基于这类数学知识的技术时，通常他们会借助高级的工具（软件），而其中这些数学知识被封装成了黑盒——这就好像软件工程师不再使用汇编语言来编写代码一样。反过来，数据科学家需要新的统计知识，特别是数据驱动、模型无关、稳定可靠的技术，可以用于现代、大规模、快速变化、时常是非结构化的数据。此外数据科学家还要将非结构化数据进行结构化处理和精通分类学、NLP（自然语言处理，也被称为文本挖掘）和标签管理系统。

数据科学家也不是计算机科学家——首先，因为他们不需要具备计算机科学家那样完整的计算机理论知识体系，其次，数据科学家需要更好地了解随机过程、实验设计及抽样，这些通常是统计学家擅长的领域。但是，数据科学家需要熟悉计算复杂度、算法设计、分布式架构和编程（R、SQL、NoSQL、Python、Java和C++）。独立的数据科学家可

以使用Perl来代替Python，虽然Python已经成为标准的脚本语言。数据科学家需要熟悉软件开发生命周期和精简架构以进行代码开发和团队协作。

数据科学家还应成为一两个应用领域内的专家（例如，本书的作者是数字分析和欺诈检测两个领域的专家）、有可以分享的成功经验（以及量化成功的指标）、有商业洞察力，并能评估数据科学解决方案给客户或者老板带来的投资回报率。如果你在某个横向领域已经具有几年的行业经验和培训（运筹学、面向大规模数据集的应用统计学、计算机科学、工程学或者侧重分析学科的MBA学位），以上这些技能大多数可以在短期内获得。

数据科学家还需善于沟通，能够理解，很多时候还需猜测，客户、老板或者管理层试图解决的问题。尤其重要的是能够把高层次的英语语言转换成简单、高效、可扩展、可复制、健壮、灵活、平台独立的解决方案。

最后，分析师　这个名称适合于一个较初级的角色，一般指分析数据的人，但其不涉及创建系统或架构来自动分析处理数据并基于自动检测到的模式或理解执行自动的操作。

总结起来就是，数据科学=一些（计算机科学）+一些（统计科学）+一些（业务管理）+一些（软件工程）+专业领域知识+新的（统计科学），其中

- 一些（）指该领域并非完全从属于数据科学。
- 新的（）指需要该领域的新知识。

# 横向与纵向数据科学家

纵向数据科学家　在某个细分的领域有深厚的技术知识。例如，他们可能是下面这些人中的一种。

- 计算机科学家，熟悉所有排序算法的计算复杂度。

- 统计学家，精通特征值、奇异值分解及其数值稳定性和最大伪似然估计的渐近收敛性。
- 软件工程师，在编写Python代码（包括图形库）并将其应用于API开发和网络爬虫技术方面有多年的经验。
- 数据库专家，在数据建模、数据仓库、图形数据库、Hadoop和NoSQL方面专业知识过硬。
- 预测建模师，具有贝叶斯网络、SAS软件、支持向量机方面的专业知识。

这里的关键是，我所说的“纵向数据科学家”指那些技能领域较窄的人，比如精通所有Lasso相关的回归分析，但对时间序列所知有限，而对计算机科学则了解更少的人。然而，深厚的行业专业知识对数据科学家的成功是绝对必要的。在一些领域，比如股票市场套利、网络广告套利、临床试验，如果你不了解其复杂的商业模式、生态系统、周期和泡沫，以及繁杂的行规，就注定会失败。如在欺诈检测领域，你要跟得上犯罪分子们玩的新花招（详细内容请参阅第6章关于欺诈检测的相关章节）。

纵向数据科学家是大学教育系统的副产品，即大学可以把一个人培养成一位计算机科学家、一位统计学家、一位运筹学者，或者一位工商管理硕士——而不是这四者的综合。

相反，横向数据科学家　是业务分析师、统计学家、计算机科学家和领域专家的混合体。他们既有远见卓识又有专业知识。在特征值、广义线性模型和其他快过时的统计技术方面他们可能不是专家，但他们了解用更现代的基于数据驱动的技术去处理非结构化的、流式的大数据，比如用简单实用的分析桥第一定理来建立置信区间。他们能够设计稳定、高效、简明、可重复、可扩展的代码和算法。

因此我所说的“横向数据科学家”，意思是你需要有跨学科的知识，包括计算机科学、统计学、数据库、机器学习、Python方面的知识，当然还要有行业专业知识。这是一般统计学家通常缺少的典型技术知识。

横向数据科学家还具有以下特点。

- 他们比较熟悉六西格玛概念（特别是80/20规则），即使并不清楚术语的叫法。从本质上说，对于这些分析从业者，速度比完美更重

要。

- 他们在处理大型、复杂、混乱的数据集方面有成功的经验，以及衡量成功的标准。
- 在识别真正需要解决的问题、确定所需的数据集（外部的和内部的）、所需的数据库结构和指标方面，他们都有经验。他们并非被动地使用第三方生产或收集的数据集，这些第三方常常缺乏收集和创建正确数据的技能。
- 他们知道实践经验法则和需要避开的陷阱，而不仅有理论知识。不过对以下这些方面，他们掌握的要比基础知识更加深入：计算复杂度、良好的采样与实验设计、健壮统计学与交叉验证、现代数据库设计，以及编程语言（R、脚本语言、MapReduce概念、SQL）。
- 他们掌握高级Excel功能和可视化技术。
- 他们能创造实用的信息板（人们每天会用它们来做各种决定），或者一些替代工具，人们可以用它们来交流数据方面的发现（就是说，通过E-mail，或者自动的和某些实时的端对端模式）。
- 他们具有创新的思维方式。例如，在创建一个推荐引擎时，他们知道会受到垃圾邮件发送者和竞争用户的干扰，因此会设计一个有效的机制来检测虚假评论。
- 他们是创新者，创造真正有用的东西。讽刺的是，这可能会吓跑潜在的雇主。尽管这些雇主声称他们需要这样的人才，但出于明显的原因，他们更喜欢服从命令的好士兵，而不是破坏性的创造者。

数据科学家的另一部分工作是参与数据库设计和数据收集过程，识别那些对价值发现最有用的指标和外部数据源。这包括辨别任何典型的大数据集里都会存在的深深埋藏的信号和海量噪声。更详细的信息请参阅本书第2章“大数据‘诅咒’”一节，其中介绍了在比较数百万的时间序列时，分别使用相关图匹配和相关比较来检测真实信号。还可参阅第6章关于预测能力的章节，其中讨论了一种新型的熵（无序状态测量法），用来识别哪些功能能够捕获信号，而哪些功能是“信号盲”。

数据科学家还需要精通健壮的交叉验证、置信区间和方差缩减技术，从而提高精确度或至少能衡量和量化不确定性。甚至数值分析也可能有用，尤其对于那些参与数据科学研究的人。以上所述以及业务专业知识的重要性，会在第6章关于股票交易的章节中得到更清楚的阐述。

最后，数据科学家会用到3种主要的分析学方法：描述性分析、预测性分析和规范性分析。每一种会以不同的方式给机构带来商业价值。

我想给这个列表再添上“自动化分析”和“根源分析”，虽然它们在某些方面和预测性分析及规范性分析有重合。

真正的横向数据科学家需要具备宽广和有深度的知识结构，这也成为招聘者难以找到这样人选的原因之一（所以他们找到并雇用的大部分是纵向数据科学家）。用人企业还不太熟悉怎样识别横向数据科学家——在各种职业分析人员中，他们是真正能为公司赚钱和提高投资回报率的人。造成这种情况的原因是双重的，如下。

- 未经训练的招聘者很快注意到横向数据科学家缺少一些真正的计算机科学家、统计学家，或者MBA必须具有的传统知识——于是将横向数据科学家从应聘人选中排除出去。招聘者需要对软件工程师、业务分析师、统计学家和计算机科学家都比较了解，才能辨认出简历关键词里概括不了的特质，以及哪些技能是关键的，哪些是可以忽略的，从而发现真正的人才。
- 横向数据科学家，虽然具有能真正显著提升投资回报率的知识，却面临着工作职位稀少的前景，最后只好自己独立创业，有时甚至和一些需要真正的数据科学家（这样的人才比较少有）的公司直接竞争。在不止一次向Microsoft、eBay、Amazon.com，或谷歌投递简历而得不到面试机会后，他们不再向其投递简历，这更减少了合格的应聘人选。

练习

你能说出几位横向数据科学家的名字吗？D. J. Patil，曾在LinkedIn担任首席数据科学家，是一个横向数据科学家的例子。看看你能不能通过调查研究发现其他一些横向数据科学家。

# 数据科学家的类型

数据科学家有几种不同的类型，包括伪数据科学家、自学成才的数据科学家、业余数据科学家和极限数据科学家。下面的章节里分别讨论了这些不同类型的数据科学家，并介绍了一些业余数据科学和极限数据科学的例子。

# 伪数据科学家

本书前文已讨论过伪数据科学，以及什么是伪数据科学家——一位具有狭窄专业知识的职业人士，可能因为她还处在职业生涯的起点，所以不应被称为数据科学家。前一节谈到，真正的纵向科学家不能算作真正的数据科学家。同样，如果一位统计学家虽具有R和SQL编程技能，但从未处理过1　000万行的数据集，那么他也只是统计学家，不能算作数据科学家。有一些书的书名强调数据科学，然而只不过是给统计学书籍重新贴了个标签。

# 自学成才的数据科学家

自学成才的数据科学家指没有接受过正规教育培训（没有数据科学方面的学位），但是具有合格的数据科学家所需的所有知识和技能的人。目前，大部分数据科学家都是自学成才的，因为培养真正的数据科学家的培训项目非常少。然而随着时间推移，这种类型的数据科学家会变成少数群体，甚至会消失。本书的作者是一位自学成才的数据科学家，曾在计算统计学专业获得博士学位。就像企业管理人员的存在远早于MBA课程的出现，数据科学家的存在也必然会早于正式教学项目的出现。他们中的一些人从事以下的专业工作。

- 为数据科学铺设基础条件。
- 帮助企业定义他们的数据科学需求和策略。
- 帮助大学开设和调整教学项目。
- 开展创业项目，解决数据科学问题。
- 开发软件和服务（提供数据科学或分析学服务）。
- 开发数据科学方面的独立培训项目、资格认证或者实习训练（包括

私人资助的数据科学研究）。

# 业余数据科学家

另一类值得一提的数据科学家是业余数据科学家　。业余数据科学家就像业余天文学家：装备齐全，拥有正确的工具（比如R、SAS软件，或RapidMiner），能接触到大数据集，受过教育，有实际专长。她通过参加数据科学竞赛赚钱，比如Kaggle、CrowdANALYTIX和Data Science Central发布的比赛，或者做相关兼职。

## 业余数据科学示例

由于到处都有可以免费使用的大量数据及大量的开放工具，因此出现了一种新的分析人士：业余数据科学家。像业余天文学家一样，业余数据科学家能够为行业的发展和完善做出很大贡献，并解决一些终极难题。如果有成千上万的业余数据科学家使用开放源代码工具分析公开的数据（图像、视频、Twitter消息等），像波士顿马拉松爆炸嫌疑犯那样的人能不能被找到？毕竟，业余天文学家曾经检测到过天体行星及很多其他发现。

同时，就像业余天文学家只需要一样昂贵的工具（一架具有数据记录功能的高性能望远镜），业余数据科学家也只需要一样昂贵的工具（一台高性能的笔记本电脑，以及可能需要注册一些云端存储/计算服务）。

业余数据科学家可能通过以下活动来赚钱：在Kaggle竞赛中获胜、研究怎样识别僵尸网络、解释股市的闪电崩盘、击败谷歌网页排名算法、帮助找到新的复杂分子来对抗癌症（分析化学），以及预测太阳耀斑及其强度。对成为业余数据科学家感兴趣吗？下一节会为你提供一个可供初步尝试的项目。

## 业余数据科学的一个项目实例

问题：大流星会导致多个小陨石坑还是一个大陨石坑？如果流星通常分裂成多个碎块，或者在接近太阳系时已经分解成几块，其危险性可

能就不像整个大流星一次冲击那么大。这就是要探讨的问题。虽然我不确定这个假设是否正确，但即使相反的假设是正确的，关于二次撞击坑的频率问题还是值得探讨的。

最后，如果能弄清楚陨石是以碎块形式，而不是以一个整体形式到达地球，可能会改变政府的政策及其优先级，也许我们就不必在检测和击毁流星的项目上花太多钱了（或者情况也可能相反）。

那么，你该怎样着手评估一颗大流星（撞击地球时）造成多个小撞击的概率，以及平均会有多少个小撞击？察看月球的陨石坑是一个思路，可以看看有多少陨石坑是规律分布的。然而，流星在撞击地球前爆炸的原因（因此造成多个陨石坑）是地球厚厚的大气层，因此月球可能提供不了好的参考数据。而地壳的地质活动非常活跃，以至于所有的陨石痕迹在几百万年后就会消失。也许金星会是一个好的数据来源？不，甚至比地球更差。也许火星？不，它和月球一样。也许木星或土星的一些卫星会是好的备选对象。

在确定了数据源并且回答了上面的问题后，就可以进一步探究更深层的问题，比如，如果你看到一个二次撞击坑（两个坑，由同一颗流星造成），那么两个坑之间的平均距离是多少？　这项研究也有助于更好地评估人口风险，以及NASA应该在流星追踪项目上花费多少亿美元。

总之，作为一个新手，我做了一些研究，在网站 http://www.stecf.org/~ralbrech/amico/intabs/koeberlc.html ，以及一幅标示了地球上陨石坑的地图上找到了数据。

肉眼所见的情况像是多次撞击（例如，二次陨石坑），而且陨石坑对齐排列是常态而不是例外。但是人的大脑在检测概率方面并不擅长，因此需要进行统计数据分析。值得注意的是，第一步处理图像，从中检测陨石坑和提取坐标，你可以使用一些软件或者自己编写代码。但这些对一个好的业余数据科学家不难做到，你可以在互联网上找到适用的工具。

# 极限数据科学家

另一种有趣的数据科学家的类型是极限数据科学家　：就像极限登山或极限编程，极限数据科学意味着在不借助任何数据模型的情况下开发出强大、稳定的预测解决方案。

## 极限数据科学示例

极限数据科学是一种使用数据科学快速创造高价值的方式。将它和极限登山相比可能最为贴切，那些登山者设法用最少的时间，以小团队方式和少量的预算挑战无人攀登过的山峰，有时成功地使用有争议的攀登路线或技术。例如，Reinhold　Messner曾独自一人在冬天攀登珠峰，只用了3天，没有氧气，没有夏尔巴人向导——是的，他真的做到了！

极限数据科学借助精简的六西格玛方法、敏捷的开发和分析、极限编程，很多时候在没有统计模型的情况下对大数据进行高级的统计分析，然而却提供了卓越的投资回报率（部分原因是由于节省了雇用高收费的大批统计人员所需的资金）。

就像极限登山，这不是胆怯者所能做到的：几乎没有统计人员能在不需要数学建模和类似广义线性模型这样强大工具的情况下设计出可靠的预测技术和置信区间。

唯有那些具有卓越直觉、判断力和远见的数据科学家才能胜任极限数据科学工作。他们将深度分析技术、先进的统计知识、一贯正确的直觉和快速深入问题本质的能力结合起来。他们能用1个月的时间为商业问题做出一个更好的解决方案，而一群依靠模型的“正规”数据科学家在同样的项目上可能要花费6个月。更好的解决方案还意味着更简明、更具可扩展性和更加健壮。这些数据科学家们往往是“万能型”的，有时他们的成功来自于将比较普通的任务广泛自动化。

现今的一个问题是，许多所谓的数据科学家在做着极限数据科学工作而不自知。然而他们没有所谓的胜任这种工作的能力，结局也总是不尽如人意，因为他们在设计和实施算法时，每一步都引入了偏差和错误。求职面试问题并不测试极限数据科学家所需的能力。结果如何呢？比如你有50个得到认可的专业登山者，在冬天攀登珠峰，没有夏尔巴人，没有氧气瓶，钱也很少，结果48个失败了（没有到达顶峰）。然而有25个会到达几个里程碑处（其中5个很接近顶峰），5个丧生，10个拒绝做这件事，8个惨遭失败，2个会成功登顶。极限数据科学的情况和这差不多。

那些成功登顶者会比任何一个配备了几十个夏尔巴人、几百万美元、各种闹钟和哨子的人做的还要好。注意这点因为（按本意）极限数据科学家以小团队方式工作，所以他们必须是“万能型”的——就像Reinhold Messner在他的极限登山探险中一样，他需要同时是登山者、医生、摄影师、气象学家等。

# 数据科学家人群特征

本节讨论的信息来源于最受欢迎的数据科学网站上的流量统计数据和人群统计数据、Quantcast.com数据，以及内部流量统计数据和其他一些资源。

数据科学网站上的访客人数以美国居首，通常印度访客的访问率排名第二，但大约是美国访客访问率的四到五分之一。美国的访客往往从事较资深的或者管理的职务，而印度的访客通常是初级的、入门级的分析师，管理人员较少。最近在一些地方，包括爱尔兰、新加坡和伦敦，有经验的数据科学专业人士的数量有所增长。

能提供这类数据的较好的来源有：

- Quantcast.com。利用邮政编码可以收集关于数据科学家性别、收入、教育程度、种族等的人群统计数据。邮政编码信息可以通过IP地址获得。
- 用户信息则来自ISP（互联网服务供应商）。几家大的互联网服务供应商将用户账户信息按IP地址分类，出售给Quantcast。然后Quantcast的数据科学引擎基于这些ISP样本数据进行统计推断。
- 此外调查活动或其他来源——例如，在你的浏览器上安装一些工具栏，允许你的网页浏览活动被监测，可以得到概括性的统计数据。（这种方法有很大的偏见，因为安装这种工具栏的人和那些不安装的人本来就属于不同的两类）。这是一段时间之前，Alexa.com最喜欢的产生流量统计数据的方式。

数据科学网站吸引了高学历、富裕的男性，他们主要是亚裔，大多数生活在美国。

# 数据科学方面的培训

不多的一些大学和组织已经开始提供数据科学学位、培训和证书。下面的章节对此做一些介绍。

# 大学课程

下面是一些美国大学开设的数据科学和数据科学家教学项目，以及它们网站上的项目介绍。

- 华盛顿大学（西雅图，华盛顿州）：　培养进入数据科学领域所需的计算机科学、数学知识和分析技能。使用数据科学技术从极大的数据集或者大数据中分析和提取其内涵。熟悉关系型和非关系型数据库，将统计学、机器学习、文本检索及自然语言处理技术应用于数据分析，并解释其结果。学习将数据科学应用于市场营销、商业智能、科学研究等领域。（http://www.pce.uw.edu/certificates/data-science.html ）
- 西北大学（埃文斯顿，伊利诺伊州）：　随着企业寻求最大限度地利用现有的、海量的、新数据，西北大学开设了预测分析科学硕士学位，培养学生满足各行业日益增长的对数据驱动及解决问题能力的需求。先进的数据分析、预测建模、基于计算机的数据挖掘、市场营销、网络、文本和风险分析只是该项目提供的研究内容的一部分。（http://www.scs.northwestern.edu/info/predictive-analytics.php ）
- 加州大学伯克利分校（伯克利，加州）：　加州大学伯克利分校信息学院独家提供完全在线授课的专业信息和数据科学学位（MIDS）。这个振奋人心的项目包括以下内容。

- 多学科的课程，培养学生使用复杂和非结构化的数据解决现实世界问题的能力。
- 基于网络的学习环境，将面对面的实时教学与互动的在线授课式课

程融为一体，由加州大学伯克利分校的教师设计课程并进行讲授。

- 基于项目的学习方式，此学习方式整合了最新的工具和方法，可以使用它们从数据中识别模式、获得见解并高效交流这些发现。
- 访问I School社区的完全权限，包括个性化的技术和学术支持。
- 通过加州大学伯克利分校校友网络，拥有在海湾地区（数据科学革命的心脏地区）建立人脉的机会。

（http://requestinfo.datascience.berkeley.edu/index.html ）

- 纽约城市大学（纽约市，纽约州）：　数据分析专业的在线硕士学位（科学硕士），培养学生通过综合及挖掘大数据来分析现实世界现象和日常活动的能力，并从中发现模式、关联和趋势。大数据已越来越成为重大商业决策背后的驱动力。通过本项目的学习，学生在收集、存储及处理来自通常互不关联的源的不同种类数据方面的能力会大为提高，能够用前所未有的新方式回答复杂的、数据驱动的问题。（http://sps.cuny.edu/programs/ms_dataanalytics ）
- 纽约大学（纽约市，纽约州）：　整个纽约大学都积极重视数据科学，因为它已经开始对学校的大部分学术活动产生革命性的影响，对世界范围内的学术活动也是如此。这场革命才刚刚开始。数据科学正在成为回答我们这个时代一些重大科学问题和技术挑战的必要工具：大脑如何工作？怎样制造智能机器？怎样更好地找到治愈疾病的方法？

  数据科学与纽约大学的多门传统学科有交叉，比如数学（理论数学和应用数学）、统计学、计算机科学，以及更多其他应用领域。它还会对很多领域产生影响——从医疗保健，到企业，到政府——那些与纽约大学的院系有合作的机构。

  （http://datascience.nyu.edu/ ）
- 哥伦比亚大学（纽约市，纽约州）：　哥伦比亚大学数据科学与工程学院致力于成为泛数据科学这一新兴领域全球最领先的研究和教育机构，涵盖理论和实践方面。另一同样重要的使命是，支持和鼓励在研究项目中产生的创业企业。为了实现这一目标，学院积极促进与其他院系教师的紧密关系，聘请新的教师，以吸引对大数据相关问题感兴趣的各学科学生，并和业界合作伙伴建立牢固的互惠关系。学院努力吸引联邦政府和业界的外部投资，用以支持研究和教育项目。（http://idse.columbia.edu/ ）
- 斯坦福大学（帕罗奥图，加州）：　随着用户与网络交互技术和网

络互联技术的崛起，以及处理能力和存储能力方面的技术进步，对高效、先进的知识发现技术的需求已呈指数级增长。企业需要把大量的信息转化为商业智能，从而做出明智的商业决策。
获得“大规模数据集挖掘”证书后，意味着你掌握了高效强大的技术和算法，能够从网络、社交网络图和大型文档库等大数据集中提取信息。这些技术能使你的公司获得竞争优势，使你的事业更上一层楼。
数据挖掘和应用学位涵盖了数据挖掘和机器学习方面很多重要的新思想，并以统计学框架对它们进行了解释，同时介绍了它们在商业、科学和技术上的一些应用。
（http://scpd.stanford.edu/public/category/course-CategoryCertificateProfile.do?method=load&certificateId=10555807 ）
（http://scpd.stanford.edu/public/category/courseCategory-CertificateProfile.do?method=load&certificateId=1209602 ）

- 北卡罗来纳州立大学（罗利，北卡罗来纳州） ：如果你有数学和统计编程的天赋，以及用数据来解决挑战性问题的热情，这个项目就是为你而设的。MSA（测量系统分析）专为像你这样的人精心设计，通过10个月的学习，你就能够驾驭大量多样化的数据，从中得到具有实操性的深刻见解。>>更多新书朋友圈免费分享微信xueba987（http://analytics.ncsu.edu/?page_id=1799 ）

许多其他大学也有很强的分析学课程，包括卡耐基-梅隆大学、哈佛大学、麻省理工学院、佐治亚理工和沃顿商学院（客户分析计划）。要了解各个学术性数据科学教学项目的信息，可以访问以下网站。

- http://www.analyticbridge.com/group/analyticscourses/forum/topics/science-university-programs
- http://whatsthebigdata.com/2012/08/09/graduate-programs-in-big-data-and-data-science/

类似上面所列的大学教学项目通常提供以下课程。

- 数据导论（数据类型、数据变换、术语等）。
- 存储和并发处理初步。
- 文件和基于文件的数据系统。

- 关系型数据库管理系统。
- Hadoop导论。
- NoSQL——MapReduce和并行数据库。
- 搜索和文本分析。
- 实体识别。
- 推断统计学。
- 高斯分布、其他分布和中心极限定理。
- 测试和实验设计。
- 贝叶斯与经典统计学。
- 概率表达的线性回归和最大似然估计。
- 图算法。
- 原始数据到推理模型。
- 机器学习的动机与应用。
- 有监督学习。
- 线性和非线性学习模型。
- 分类、聚类和降维。
- 高级非线性模型。
- 协同过滤和推荐系统。
- 健壮的模型。
- 数据科学与文本和语言。
- 数据科学与定位。
- 社交网络分析。

# 公司和协会培训项目

一些其他类型的组织——既有私人公司和专业组织——也提供证书或培训。下面是其中一些。

- INFORMS（运筹研究协会）：分析学证书。
- 数字分析协会：证书。
- TDWI（数据仓库研究所）：课程，重点是数据库架构。
- 美国统计协会：统计师许可证。
- 数据科学中心：数据科学实习训练。
- 国际分析协会：更像一个智囊团。由著名的Tom Davenport创立

（哈佛客座教授及数据科学奠基人之一）。
- Statistics.com：统计学课程。

收费范围从100美元以下的无考试认证，到几千美元的完整课程。

在许多专业会议上，也有可能得到关于分析学、大数据或数据科学方面的培训，比如下面这些。

- Predictive Analytics World（预测分析学世界）。
- GoPivotal Data Science（数据科学之路）。
- SAS数据挖掘与分析。
- ACM——计算机协会。
- IEEE——电气和电子工程师协会的分析/大数据/数据科学会议。
- IE Group——创新型企业集团。
- Text Analytics News（文本分析学新闻）。
- IQPC——国际质量与生产力中心。
- Whitehall Media（白厅媒体）。

像EMC、SAS和Teradata这样的供应商也提供有价值的培训。另外如kaggle.com等网站提供的数据科学竞赛的机会，你可以访问真实的数据，有时获奖者还会被Facebook或Netflix这样的公司雇用。

# 免费培训项目

你可以不用抵押房产，就能获得一些数据科学培训。下面是一些最好的免费培训网站和培训项目。

### Coursera.com

Coursera.com提供免费的在线培训。教师是来自世界各地的大学教授，所以教材有时会有点学术性。以下是本书写作时该网站提供的一些课程。

- 机器学习（斯坦福大学）。

- 数据结构与算法（北京大学）。
- 网络智能和大数据（印度理工学院）。
- 数据科学导论（华盛顿大学）。
- 地图和地理空间革命（宾夕法尼亚州立大学）。
- 数据库简介（斯坦福大学，自学课程）。
- 数据分析计算（约翰·霍普金斯大学）。
- 初级编程（洛桑联邦理工学院，瑞士，法语）。
- 统计学一（普林斯顿大学）。

### 数据科学中心

`https://www.datasciencecentral.com` 网站提供在线、按需、自助、免费的数据科学实习训练。课程包括以下部分。

- 培训基础。
- 教程。
- 数据集。
- 实践项目。
- 样本源代码。

该实习训练包括以下基本科目。

- 怎样下载Python、Perl、Java和R，获取样本程序，开始编写一个高效的网络爬虫，并开始使用Linux、Cygwin和Excel（包括逻辑回归）。
- Hadoop、MapReduce、NoSQL，它们的局限性和更现代的技术。
- 怎样在网络上寻找数据集或免费下载大型数据集。
- 怎样分析数据，从理解业务需求到维护生产模式下的自动化（机器与机器通信）网络/数据库应用程序：过程有12个步骤。
- 怎样开发你的第一个“分析即服务”应用程序并且扩展它。
- 大数据算法，以及怎样使它们更高效、更健壮（应用于计算优化：怎样有效测试几万亿个多元向量，从而设计好的评分）。
- 统计学基础知识、蒙特卡洛算法、交叉验证、健壮性、采样及实验设计。
- 分析人员的创业点子。
- Perl、Python、实时分析、分布式架构基础知识，以及一般编程实

践。
- 数据可视化、信息板，以及如何以管理顾问的方式进行沟通。
- 给未来顾问的一些建议。
- 给未来企业家的一些建议。
- 经验法则、最佳操作、技术秘密，以及为什么数据科学也是一门艺术。
- 其他在线资源。
- Lift和其他衡量成功的指标、指标选择、外部数据应用、通过模糊融合和统计技术使数据孤岛间建立通信。

还有一些特别教程，介绍多种数据科学主题，如大数据、统计学、可视化、机器学习及计算机科学。提供一些样本数据集，用于和教程配合。

学生使用在实习训练中学到的技能完成几项实践项目。以下是一些实习训练中完成的项目例子。

- 黑客和逆向工程项目。
- 使用网络爬虫技术浏览Facebook或Twitter网页，然后提取分析获得的内容，估算Facebook上无效或重复账户的比例，或对Twitter消息进行分类。
- 创建分类或改进现有的分类。
- 谷歌关键词的最优定价。
- 开发网络应用程序，用来提供（实时的）高于平均水平的交易信号。
- 在海量数据中，识别低频和僵尸网络欺诈案例。
- 以数据科学为先导，在计算市场营销领域实践。
- 自动化抄袭检测。
- 使用网络爬虫，评估谷歌是否将自己的产品和服务（如谷歌分析）设置得优于竞争对手产品。

实习训练课程提供样本源代码，以及外部代码的链接，如Perl、Python、R和C++，有时会有XML、JavaScript和其他语言。甚至包括用Excel来执行逻辑回归和预测建模的“代码”，这是一种开发自动化信息流的系统。

# 数据科学家职业道路

本节讨论个人创业的职业道路。想为大小企业雇主和其他组织工作的专业人士请参阅第8章“数据科学资源”，其中有相关资源介绍。

# 独立顾问

数据科学家的职业选择之一是成为一名独立顾问，全职、兼职均可，用咨询收入来补充传统收入（工资）。顾问通常有多年的经验和深厚的专业知识，使用软件和编程语言，并具有销售和演讲技巧。他们知道如何提供价值，如何衡量提升，如何规划一个咨询项目。第一步是写一个计划书。独立顾问可能会花费50%的时间寻找客户和进行市场营销，尤其是在他们的收费远远高于100美元/小时的情况下。

找到客户

参加会议、与前上司和同事保持联系，在LinkedIn上建立一个强大的人际网（推荐，互相认证职业技术，拥有几千个相关联系人）是至关重要的，可能需要几个月的时间来完成。为扩大你的网络社交圈，可以在特定网站上发布相关的非商业性文章，或者为网站论坛做贡献（回答问题）。

- Analyticbridge.com。
- BigDataNews.com。
- DataScienceCentral.com。
- LinkedIn群组。
- 谷歌+大数据团体。
- Quora.com。

例如，LinkedIn小组“高级商业分析、数据挖掘和预测建模”有超过10万名成员。它是引领性的数据科学小组，每个月增长3000名成员，有很多下属小组，比如大数据、可视化、运筹学、分析管理人员、培训小组等。（参见www.linkedin.com/groups/Advanced-Business-Analytics-

Data-Mining-35222 。）

通过参加Kaggle竞赛，你可以赚取金钱和名声（以及新客户）（或者参加我们自己的比赛——例子参见http://bit.ly/1arClAn ），或在Kaggle上被列为顾问。它的“最佳”顾问每小时收费300美元（参见http://bit.ly/12Qqqnm ）。然而，你要和来自世界各地的成千上万的其他数据科学家竞争，这些项目往往是相当小的（每个几千美元），而且Kaggle要从你的收入中抽走一笔不小的佣金。

## 管理你的财务

我起步的时候，没有商业计划书（15年后仍然没有），没有律师，没有办公室（现在还是没有），没有员工，我妻子负责我们的网络联系和会计工作。现在我们有一个记账员（每月40小时）和一个会计，他们处理发票和收款事宜，每年需付给他们大约1万美元，这占我们总收入的不到百分之一，花得很值。

你确实可以用很少量的钱开始创业。所有创业者都是冒险家，而一点点风险意味着未来更好的回报，因为在昂贵的服务上节省的钱可以投入到业务中，用来推动业务增长。在注册成立LLC in 2000时，我们使用了MyCorporation.com的服务（现在它仍然存在），它只收了我们几百美元，就把一切都办好了。直到今天，我们还是没有律师，不过我们订购了一种法律保险，费用是每月50美元。我们的商业模式本身就不大可能产生昂贵的诉讼费用：我们远离那些可能导致诉讼的情况。这同样也适用于培训：你可以免费学到很多东西，例如通过互联网，或者通过阅读本书，或者参加Coursera.com的免费课程。

营销和广告是一大笔开销。降低营销成本的方法之一是创建自己的社区，聚拢一大批对你的服务感兴趣的人，从本质上说就是“拥有”你的市场。建立自己人际网的一种方法是，在LinkedIn小组里发布相关讨论及创建自己的LinkedIn小组。事实上，我们自己建的社团后来变得如此之大——成了领先的数据科学家社团——于是我们便不再致力于扩大它。现在我们赚钱的方式是，接受广告和为赞助商发送eBlast及消息，它们被混在优质、新鲜的非商业性内容中提供给我们的订阅者。这也表明，如果数据科学家对非传统职业道路感兴趣，做一个数字出版商是一个可行的选择。

最后在与员工共事方面谈一点：你可能会和罗马尼亚人或印度人远

程合作，只需支付相当于欧洲或美国雇员工资一小部分的费用，不必受员工法制约，并体验双赢的关系。事实上，你的顾问业务会增长，但你每个月能用于工作的时间是固定的，扩大规模的方法之一是接受更多的项目，可把一部分可能比较平常的业务外包到国外。另一种实现增长的方法是把很多任务自动化，比如报表制作、分析、数据清理和探索性分析。如果你能设计一个脚本，在你睡觉的时候运行它，让它一晚上做5个项目，产出等同于5个顾问的工作量，那么你就找到了让年收入增加3～5倍的办法。然后，你可以把你的收费结构削减50%，这样仍比你的竞争对手赚得更多。

## 薪酬调查

许多网站提供薪酬方面的信息，但通常适用于工薪雇员而不是顾问。LinkedIn上的招聘广告通常会提供一个薪酬范围。Indeed.com和PayScale.com也是好的信息来源。Glassdoor.com在员工报告中列出了实际工资。数据科学中心定期发布薪酬调查，包括顾问的薪酬，参见 `http://bit.ly/19va0n1` 。

据说Kaggle对数据科学项目的收费是300美元/小时，但通常美国的收费是介于60美元/小时和250美元/小时之间，根据项目的复杂度和所需的专业知识水平而定。统计诉讼领域通常收费更高，但你需要是知名的数据科学家，才能揽到这样的活。

下面是2013年statistics.com上发布的一些统计顾问广告里的小时费率。

- Michael Chernick：150美元。
- Joseph Hilbe：200美元。
- Robert A. LaBudde：150美元。
- Bryan Manly：130美元。
- James Rutledge：200美元。
- Tom Ryan：125美元。
- Randall E. Schumacker：200美元。

## 样本提案

在开始做任何一个3　000美元以上的项目前，要先做一个提案，概

述项目的范围、所要执行的分析、数据来源、可交付成果、时间期限和费用。下面是两个样本提案。

样本提案 **#1**

这个项目是关于点击欺诈检测的。注意以下信息。

- 我收取多少费用及收费方式（我在伊瑟阔市，华盛顿州）。
- 我怎样将项目拆分成多个阶段。
- 怎样展示可交付成果和可选择分析，以及项目里程碑。

提案包括下列工作项目。

1. 概念验证（5天的工作）

- 处理测试数据集：最近7或14天的点击数据；包括领域有，用户KW（关键词）、推荐ID（或者更好的情况，推荐域）、识别码、代销商ID、CPC（每点击成本）、会话ID（如果可获得）、UA（用户代理）、时间、广告主ID或登录页面，以及IP地址（域待定）。
- 通过我的基本评分系统识别测试数据集中的欺诈案例或无效点击。
- 将我的得分与你的内部得分或指标进行比较。
- 解释差异：两种方法中的误报/漏报。我的方法是否带来了性能的提升？
- 成本：4 000美元。

2. 创建规则体系

- 每周创建4个规则，共4周；首先创建最紧急/最强的规则。
- 和产品团队协作，以保证正确执行（QA测试）。
- 成本：200美元/规则。

3. 创建评分系统

- 从第一天开始，让你的团队为生产代码和数据库中的每次点击添加

一个关联的标记向量。

- 每次点击的标记向量都被实时计算，并且/或者每天或每小时更新。
- 点击得分将是标记向量的一个简单函数。
- 讨论数据库架构的基本组成部分，包括查找表。

- 标记向量为每次点击存储以下信息。

- —触发了哪些规则。
- —每个规则关联的值（如果不是二元的）。
- —规则是否活跃。

- 创建基于标记向量的得分：即给每个规则和标记向量分配权重。
- 按规则集群将规则分组，进一步细化系统（可选，需额外成本）。
- 成本：5 000美元（主要用于创建评分系统）。

4. 维护

- 机器学习：3个月后再测试系统，以发现新规则，微调规则参数，或学习如何自动更新/发现新规则。
- 评估查找表（例如，坏区域表）更新的频率（每3个月）。
- 培训数据分析人员执行点对点分析，以检测欺诈案例/误报。
- 执行聚类分析，给每个欺诈段分配一个标签（可选；如果实施的话，将为每一个不好的点击提供一个理由代码）。
- 印象文件：我们是否应该创建基于印象数据的新规则（例如，0印象点击）？
- 确保评分对所有分支长期一致。
- 信息板/高级汇总/报告能力。
- 和金融引擎整合。

- 如果可能，整合转化数据，并检测伪转化。
- 成本：待定。

该数据挖掘方法将主要依靠健壮、高效、简明的隐性决策树方法，很容易通过Perl或Python实现。

样本提案 **#2**

这个项目是关于网络分析、关键词竞价、复杂A/B测试和关键词评分的。提案包括下列组成部分。

1. 目标：本项目的目标是预测每个点击付费程序的转化率（对于很少或没有历史数据的关键词）。使用文本挖掘技术、规则引擎和预测建模（逻辑回归、决策树、朴素贝叶斯分类器或混合模型），为每位用户建立一个关键词得分。同时，将实施一个简单的参数化关键词竞价算法。该竞价算法将基于关键词的分数。

2. 方法：首先对来自一个客户的包含25万个关键词的训练集进行计分。为每个竞价关键词/匹配类型/源（搜索或内容）提供以下的域：点击、转化、转化类型和每转化收入（如果可得），收集时间为过去8周。

该竞价算法将使用一个混合公式，对有稳定转化率的关键词赋予转化率高权重，对没有历史数据的关键词赋予高权重。如果客户已经使用了一个基于转化率的竞价算法，我们将直接使用客户的算法，对于几乎没有历史数据的关键词，将转化率替换为一个适当的得分函数。该得分代表其转化率。

如果客户没有使用竞价算法，我们将提供一个参数化竞价算法，基于一个参数a（可能是多变量的）和为每个关键词随机选取的值，最终检测到最佳的a。例如，该算法的简化版本可能是：如果预测的投资回报率（对于一个指定的关键词）高于某个阈值，将竞价乘以a，a是一个介于0.85和1.25之间的随机数。在这种类型的多变量测试中，受控案例对应于a=1。

预测的投资回报率是当前竞价、得分（按设计，反过来是转换率的一个预测指标）和每次转化收入这几项的一个简单函数。

3. 可交付成果和时间安排

- 阶段1：为25万个关键词评分（包括交叉验证）。提供为新关键词评分的算法和源代码。提供公式或报表，用于计算作为得分函数的预测转化率。4周的工作4000美元。每月更新以调整评分（机器学习）：500美元/月。
- 阶段2：与客户一起实施竞价算法，4周的工作4 000美元。
- 付款方式：每一阶段预付50%，完成后付50%。

**CRM**工具

你需要客户关系管理（CRM）软件，用于跟踪线索（潜在客户）并记录你与他们的所有互动（包括联系信息，通常每个公司不止一次联系）。日志条目形式类似如下这样。

- 我的合伙人在2013年6月5日的INFORMS会议上遇到工程副总裁，讨论了反向聚类。这是一个有前景的热点线索，非常有兴趣。
- 6月8日我发了一封后续邮件。
- 安排7月23日与IT主管通电话。
- 需要索取与xyz项目关联的数据样本。
- 讨论了对模拟数据执行概念证明。等待数据。

另一个有用的工具是FreshBooks.com，这是一个在线记账系统，可以用来发送发票（并且跟踪发票是否被打开），在测量每月资金流方面功能很不错。我把它用作金融方面的客户关系管理工具，它的一些报表有助于制定预算，并使你对自己的财务状况有一个全面的了解。

最后，办理信用卡及准备一些信用卡授权表格也很有用。

注意　本章中“管理你的财务”一节也适用于创业者职业道路。

# 创业者

创业比管理咨询业务更为复杂，因为你想要扩大规模，争取更高的

收入，并且要有一个退出战略。20世纪90年代末，风险投资（VC）很流行，但是现在已经不太容易获得，而且也没有必要。花在寻找投资上的时间会相当巨大，可能会相当于6个月的全职工作，而没有任何成功的保证。我们在2006年筹到了600万美元，创建了一个公司，业务是给互联网点击、关键词或印象打分，以检测虚假点击和不良发行商，并且帮助广告主决定对没有历史数据或历史数据很少（长尾效应）的谷歌关键词投标的价格。它持续了几年，很有趣。我们犯的唯一错误是找到的投资者不团结，他们由于一个问题而开始彼此对抗。

现在风险投资已不是必需的，因为你可以用很少的费用创办一家新公司，尤其是如果精心选择一个能使用精简方法，方便扩张和部署的商业模式。例如，我们的Data Science Central平台依托Ning.com。我们没有工程师、网页开发人员、系统管理员和安全专家——所有这些功能都外包给Ning。我们没有购买任何服务器：我们的网站由Ning共同托管，每月花费不到200美元。缺点是，由于这些技术障碍被消除，你的竞争者会更多。

如果你的确需要钱来开展跟你的业务相关的小项目，众筹可能是一个解决方案。我们以前曾尝试用众筹筹集资金，来支付为数据科学中心实习培训编写教材的教授的稿酬，但没有和众筹平台签约成功。现在如果我们要通过众筹筹集资金，完全靠自己就行，不用再找中间商，通过我们的邮寄名单发布项目即可。

关于筹集资金的另一个问题是，你必须与投资者和他们的代理（分析师会尽职地研究你的IP（知识财产））分享你的创意想法。你的知识财产在这个过程中反而可能会被偷走。保护你的知识财产不被盗走的一个办法是把它高度公开，从而使它不能再申请专利。我把这称为开源的专利　。如果你选择这样做（它肯定符合小成本精简企业的理念），你需要确保控制你的市场；否则，资金更雄厚的竞争对手会卖跟你同样的产品，而且比你营销得更好。如果你拥有自己的市场——也就是说，你拥有最优质的、最可能加入的潜在客户邮寄名单——任何想宣传该产品的人都必须通过你。这时你就很容易甩掉竞争对手。拥有你的市场也是一个精简企业的重要组成部分，因为它将营销成本几乎降低为零。

最后，你可以用类似MyCorporation.com这样便宜的服务来注册成立公司（这也是一条重要的精简经营原则），不需要商业计划（另一个精简经营理念），只要你高度专注于一个明确的目标，不被各种新的想法和项目干扰就行。如果你是擅长创新的人，就找一个务实的合作伙伴。

然后每天仅需要做底层的思考。

## 我的故事：数据科学出版商

以我而言，作为数据科学家，我为市场营销人员提供营销线索。一条好的营销线索价值40美元。挖掘一条线索所需花费的成本是10美元。这需要数据科学家高效地寻找出大量高度相关的线索（通过检测和购买合适的流量，优化内在增长，以及使用其他分析技术）。如果我每年不能产出至少1万条线索，由于产量低，不会有人来买。如果我的线索不能转化为实际收入，为客户产生投资回报率，也没有人会买。

另外，感谢数据科学，我可以用比竞争对手更低的价格销售线索——远低于40美元。例如，我们推送的简报打开率在8%到24%之间，这大大提高了收入并降低了成本。我们也把客户流失率降低到了一个点，因而事实上我们一直在增长，所有这一切都是拜数据科学所赐。用来实现这些成果所用的技术举例如下。

- 提高对用户、客户和内容的细分程度。
- 外包及在国外招聘适合的人才。
- 使用自动化。
- 用多个供应商并行测试（A/B测试）。
- 利用竞争情报。
- 使用真正的计算营销策略。
- 从工程学的角度优化交付率。
- 清理无效成员。
- 检测并阻止垃圾邮件。
- 优化一个极其多元化的简报邮件指标组合（标题里的关键词、HTML代码、内容融合、商业内容和系统内容比率、避免耗尽的关键词变化度、每条消息的第一句话、提高转发数的手段、口碑和极速传播等），从而增加总的点击率、潜在客户及交付给客户的转化数。此外，你需要预测销售和收入——数据科学的另一种运用。

## 数据科学家的创业点子

下面有一些想法，提供给想在数据科学方面创业的人。希望它们还能激发你的其他想法。

浏览器里的**R**

创建一个平台，允许用户通过一个内嵌浏览器的控制台，在Chrome或Firefox中输入R命令。想想在iPad的浏览器里运行R有多轻松？我不太确定该如何导入数据文件，但我想R提供了可能性，它可能允许你打开一个Web或FTP服务器上的文件，而不是只能打开电脑上的本地文件。或者不是这样？同样，在浏览器里使用Python也会很酷。

参阅例子`http://www.datasciencecentral.com/profiles/blogs/r-in-your-browser` 。

一种新型的武器级的安全电子邮件

这对于数学家和数据科学家来说可能是一个很好的机会：创建一个公司，提供政府或任何实体都不能破译的加密邮件，为那些不希望自己的秘密被竞争对手、罪犯或政府“偷走”的企业提供安全解决方案。

下面是一个电子邮件平台的例子。

- 它的形式是一个Web应用程序，只能用于100KB以内的文本信息。你将文本复制粘贴在某个网络服务器（简称A）上的某种网络表格里。你还要创建一个检索密码，可能使用另外一个不同的应用程序来创建随机的、安全的长密码。当你点击提交后，该文本即被加密，并设定为在另外一个网络服务器（简称B）上打开。一个缩短的URL（网址）会显示在你的屏幕上，你或收件人可以点击它来读取加密的文本。
- 你给收件人打电话（或发传真），可能通过一个公用电话，告诉他这个缩短的URL及打开和解密文本所需的密码。
- 收件人随后点击这个缩短的URL，输入密码，就可以在线阅读加密的消息（在服务器B上）。收件人阅读后，该加密文本即被删除，或者按文本创建的时间顺序在48小时后自动删除。

技术讲义

加密算法在加密你的消息之前会给消息添加半随机文本。该算

法有一个加密的时间戳，同时如果没有首先添加半随机文本，它就不会工作。于是，情况会是这样：（i） 48小时后消息无法再被破译（如果加密版本被截取），因为加密信息和可执行文件本身都嵌入了一个自我销毁机制；同时，（ii）如果你对同一个消息进行两次加密（即使是一个空消息或只有一个字符），两个加密版本也截然不同（长度随机，大小至少1KB），这使得逆向工程几乎不可能实现。

- 用来执行加密的可执行文件也可以每三四天变换一次，以提高安全性并确保以前加密的消息不能再被破译。（旧版本和新版本将同时在B上保留仅仅48小时。）
- 在任何加密操作开始之前，可执行文件（在A上）会测试它是否位于正确的IP地址，以防它在别处运行，比如，政府的服务器上。此功能被加密编入可执行代码中。用来解密B端消息的可执行文件中也加入了同样的功能。
- 加密算法里嵌入了犯罪检测系统，通过检测和拒绝加密可疑的消息（儿童色情、恐怖主义、欺诈、仇恨言论等）来防止犯罪分子使用该系统。
- 该平台通过广告主和杀毒软件的付费广告赢利。
- 与B关联的URL可以随时随地改变，或者基于用户提供的密码并且位于美国之外。
- 而与A关联的URL需要更加静态。这是一个弱点，因为政府可以把它关闭。不过有一个绕开的办法，可以给这个应用程序设定一些特定关键词，比如“装甲邮件 ”，这样如果A被关闭，在别的地方会出现一个基于相同关键词的新网站，以保证服务不中断（用户需要通过谷歌搜索“装甲邮件 ”来找到一个可用的网站——即A的镜像）。
- 最后，未加密的文本不会被存储。

不可否认，政府也可以创建一个这样的应用程序，并把它伪装成私有企业产品：在这种情况下，它就成了一个诱捕用的蜜罐应用。有人担心政府会监控所有人，因此，如果政府使用的算法觉得你发布的内容，比如对压力锅的评论，是极度危险的，你就可能有麻烦了（网络连接被关闭，银行账户被冻结）。然而我认为这种看法有点夸张。

有人对创建这样的加密应用程序感兴趣吗？注意，没有一个系统是

完全安全的。如果你背后有一个看不见的摄像头，把你在电脑上做的一切都拍下来，那么我的系统不能为你提供保护——但对于收件人来说仍然是安全的，除非也有一个摄像头监控他所有的电脑活动。但是，你和收件人之间的关联（你们两人有联系的事实）对任何第三方将是不可见的。如果你在一台匿名的计算机上使用这个网络应用程序——也许某个酒店大厅里的公用电脑——安全性会更高。

求文本的平均值，而不是数字的平均值

求数字的平均值很容易，但是求文本的平均值呢？我问这个问题是因为牛津词典最近增加了一个新名词：大数据。

那么你如何定义大数据呢？找几个专家（怎样确定人选）请他们定义？那如果咨询成千上万的大数据从业者（众包），然后设法得出他们回复的平均值？你怎样用自动化的方式平均他们的意见？可能先要把他们的回复分类成几组？看上去需要用到下面这些。

- 使用分类法。
- 标准化、清理和巧妙地修剪关键词（使用停止词表和例外列表——例如，*booking*（预订）和*book*（书）不一样）。
- 使用n-grams算法，但是要谨慎。（有些关键词的形式转变不是等效的——列出这些例外。例如，*data mining*（数据挖掘）和 *mining data*（采矿数据）不一样。）
- 把回复中出现的，并且已经清理好/规范化的关键词序列与分类条目匹配起来，给回复分配类别，进一步简化文本平均过程。
- 也许会用到一个与分类相关联的规则集，比如求平均（输入A，输入C）=xyz。
- 形成一部同义词词典。

另外，你有没有听说过能自动总结文本的软件？如果有的话，将是一个神奇的工具！

用生物特征代替输入密码

想象一下，在登录你喜欢的网站时，不是用传统的方式输入密码，而是有一个网络应用程序请你说点什么，或者你用便携式电脑的摄像头拍张你自己的照片，或者通过电脑触摸屏检查你的指纹，感觉会怎么

样？然后，程序会使用几个指标的组合对你的1秒语音信息或指纹图像进行编码，并存储在数据库里，用于以后的登录验证。

这个设想是为了用更安全的技术取代传统的登录/密码（不是一种连接到网站的安全方式），只要安装了这种ID检验应用程序，就可以在任何地方，任何台式机、笔记本电脑、手机，或其他设备上使用这种技术。

同时，旧式的登录/密码系统仍然可以作为一个选项共存。这种新系统的目的是让用户可以在任何地点/设备上一键登录任意网站，而不需要记住一长串（往往是）弱密码。这种方法需要大的带宽和大量计算机资源。

进行民意测验和实时显示投票结果的**Web**应用程序

它可以有10个问题，每个问题都有多项选择，调查结果用颜色实时显示在世界地图上。每个回答者的地理位置都通过IP地址实时检测。显示的颜色（对于每个受访者）将代表所选的答案。例如，男性=红色，女性=黄色，其他=绿色。我以前为Vizu工作过，他们有一个民意调查的网络小工具，免费提供所有这些功能，但现在已经没有了。如果这类调查工具能够处理2　　000个实际的回复，并且地图能够放大，那就太好了。

针对大量邮件的收件箱投递和管理系统

这里有一个新想法，能让谷歌省下每天处理过滤数十亿条消息所花的成本，还可从中赚钱。这同时还是一个消除垃圾邮件的解决方案，而且不会像现在的系统那样有很多误报。这个解决方案就是谷歌创建自己的通信管理系统。或者至少，谷歌可以和主要供应商（VerticalResponse、Constant Contact、iContact、MailChimp等）合作，让他们的客户（每天发送数十亿条消息的公司，比如LinkedIn）根据发送量支付一定的费用。付费用户的邮件只要符合谷歌的政策，就不会沦落到Gmail的垃圾邮件箱里。更好的情况是，谷歌直接把这种通信管理服务提供给那些愿意在谷歌的控制和管理下更有效利用Gmail的用户。

我相信谷歌现在已能提供这种服务，因为目前超过50%新建的个人电邮账户是Gmail，而它们会比任何公司的电邮账户都用得久。（当你丢掉工作时，你也不会弃用自己的Gmail账户。）谷歌可以合理地对发

送到Gmail账户的每2万封邮件收取100美元的费用：潜在收入是巨大的。

如果谷歌内部提供这种服务（而不是通过Constant Contact这样的第三方），它会赚更多的钱，有更多控制权，消除垃圾邮件的任务会更容易、成本更低。

目前，因为谷歌没有提供这些服务，你将不得不面对以下问题。

- Gmail的反垃圾邮件技术很大一部分是通过协同过滤算法。如果有不少用户抱怨你的邮件，没有打开它，或没有点击它，你的邮件在开始发送的几毫秒后就会进入垃圾邮件箱。
- 因此，骗子可以创建成千上万的假Gmail账户来提高“打开”和“点击”率，使他们的垃圾邮件顺利通过，反而使协同过滤对他们有利。
- 骗子也可以使用成千上万的假Gmail账户大规模欺诈性地把它们收到的真实公司或竞争对手的邮件标记为欺诈邮件。

简报的传送速度太快：通信管理公司通常5分钟发送10万条消息。如果Gmail通过自己的系统（比如说，Gmail Contact）传送这些简报，那么它可以放慢传送速度，从而更好地控制垃圾邮件，也不会产生大量的误报。

同时，需要定期向大量订阅者发送简报的公司可以执行以下解决方案。

- 给所有的Gmail账户创建一个特别的区段，并且较少使用该区段。以我们为例，事实上我们的Gmail区段在低流失率、打开率、点击率方面是最好的（在我们所有的区段中）——如果我们不会过于频繁地使用它，并且把我们最好的讯息留给它。
- 请你的通信管理提供商使用一个指定的IP地址发送消息。
- 每3个月，删除那些从未打开过简报的订阅者，甚至可以删除那些没有点击相关内容的人。（尽管你会失去一些设置了拒收电子邮件图片的好的订阅者。）
- 创建SPF（发件人策略框架）记录。

医疗程序的定价优化

问题：

美国的医疗保健费用极高且变化无常，甚至一些标准程序的价格波动也很大。如果你想知道你的医院对一个通常5000或以上美元的医疗程序会收取多少费用，通过明确地问他们对之前5名病人收费多少是没法得到准确答案的（似乎因为他们没有做财务记录）。

试想如果汽车经销商在你购车2个月之后才知道该收你多少钱，你会买这辆车吗？如果你买了，然后得知你的邻居用一半的价格从另一个经销商那里买到了一模一样的车，你会做什么？在任何其他类型的行业里，如果经营者使用这种业务运作方式，将会迅速破产。

原因：

- 尽管医院雇用了最聪明的人（外科医生和医师），但他们缺乏基本的分析人才来进行预算、定价和预测工作。他们不会创建包括收入（保险支付等）和开支（客户不支付、人力、药品开支、设备、诉讼等）的统计金融模型。
- 造成成本的部分原因是患者和医院都不太愿意承担风险（担心诉讼）和要求做一些不必要的检查。然而，几乎没有患者会说他们对某个医疗程序的预算是多少（比如2 000美元），并且不会为此支付更多，不然就取消治疗或者必要的话找别的供应商。
- 价格肯定会波动，但必须得到控制。如果在我向客户提供咨询服务时，客户（就像病人）有一个有限的预算，而我（就像医院）也受一些限制并有需要考虑的多个方面（我需要购买好的软件，或者数据可能比预期的多很多等混乱情况）。

解决方案：

一家互联网创业公司，提供20种顶级医疗程序的价格，包括5 000家医院和20个病人区段。价格信息将来自病人填写的网络表格或者他们分享的发票细节，以及其他合作伙伴，比如希望吸引新病人的经济型医院。每个月可能只需基于8 000个数据点，通过数据推断来评估这20×20×500=200 000个价格（和它们的波动）。统计引擎将类似于Zillow（根据有限的销售数据评估所有房屋的价值），但也包括一个患

者评论组件（像Yelp那样），以及虚假评论检测。

收入将来自合作医院、把数据卖给相关机构、制药公司广告及会员费（高级会员能访问更详细的数据）。

通过电子邮件发送的支票

通过电子邮件发送的支票，代替邮局的“蜗牛”邮递——该设想方案具体如下。

- 你（付款人）要把支票寄给，比如说约翰（收款人）。
- 你登录到这个新的创业网站，输入你的银行信息（账户名称、账户号码和路由编号）和你的签名照片（第一次使用该服务时），然后输入每张支票的金额、付款原因、收款人姓名和电子邮件地址。
- 收款人收到一封电子邮件通知，请他访问某个URL来处理给他的支票。
- 他可以像打印登机牌一样，从这个URL上打印出支票（由该创业公司运营）。
- 收款人去银行兑现支票。

注意，该创业公司可以每张支票收取收款人20美分费用（比一张邮票便宜一半多），投递成功率轻松击败“蜗牛”邮件。

这种商业模式在分析方面要做的工作是开发安全组件，以防止欺诈行为的发生。其次涉及分析方面的是，在最小程度上要保证电子邮件至少99.99%的时间能成功发送给收款人。信息必须通过互联网安全地传送，尤其是发送给收款人的电子邮件里用于打印支票的URL必须包含加密信息。>>更多新书朋友圈免费分享微信xueba987

PayPal已经提供了这种功能（在线支付），但是没有打印服务。对于一些人或企业，实体支票仍然是必需的。我相信支票最终会消失，所以不能确定这样的初创公司是否有很好的前景。

用于物物交换的匿名数字货币

已经有几种数字货币存在，并在使用（想想PayPal和Bitcoin），但据我所知，它们不是匿名的。

匿名的数字货币形式将是加密的数字，只能使用一次，不捆绑任何纸币或电子邮件地址。起初，它可以用于对物物交换有兴趣的合作伙伴间。（比如，我在你的网站上购买广告，换取你帮我设计网页——中间没有美元现金交换。）这种做法很大的优点如下。

- 它不是杠杆货币——没有“银行挤兑”、通货膨胀等风险。
- 通过这种货币进行交易不需要交税。

设计这种货币有哪些挑战？最大的挑战如下。

- 让很多人信任和使用它。
- 很难折算成美元。
- 管理这个货币的实体必须以某种方式赚钱。通过什么方式？能不能对所有交易收取小额费用？交易费用必须远小于使用非匿名货币（不论是不是数字货币）进行交易时的税收处罚。
- 安全：作为一名反欺诈专家，我知道如何使它很安全。我也知道如何使它完全匿名。

在骗局传播开之前把它们检测出来

基于Twitter和博客，在新的骗局广泛传播开之前识别出它们。

- 搜索类似这样的关键词：骗局、被骗、盗窃、敲诈、欺诈 等。
- 给每个感兴趣的帖子分配一个日期标记、地理位置和类别。
- 识别并删掉编造欺诈故事的虚假帖子。
- 根据骗局的严重性和帖子的可信度为每个帖子打分。
- 创建一个分类系统，把每个帖子归类。
- 创建一个世界地图，每小时更新诈骗警报。给每个骗局标明类别、强度和时间性。

是否已经存在这样的公共系统？的确，联邦调查局肯定有，但它不提供给公众共享。另外，这种骗局警报系统很像基于众包方式来检测疾病在世界范围内传播情况的系统。两种情况的传播机制是类似的：骗局和病原体都使用病毒来传播和蔓延。骗局使用的方式是，用计算机病毒感染僵尸网络里的用户，把这些电脑转变为垃圾邮件发送机器（僵

尸），向数百万的收件人发送欺诈电子邮件。

游乐园手机应用程序

分析学能帮助游乐园在21世纪生存吗？最近在加州迪士尼乐园游玩后，我想到了很多能提高游乐园收入和改善用户体验的小办法。以下是几个可以先着手的点。

- 优化排队。考虑对那些需要花5分钟以上时间购票的游客，每超出1分钟多收费1美元。
- 提供网上售票和其他解决方案（不需要人际交流）以避免多售票点瓶颈。
- 使门票更难伪造，评估和打击假门票欺诈（门票看起来比较容易伪造）。
- 提高价格和找到最佳价格。（很明显，价格必须比现在高，由于大量的、密集的人群前来参观这些公园，造成了排队长龙，带来的其他危害也无处不在——从人满为患的餐馆和供不应求的厕所，到人山人海的景点。）
- 让游客可以在手机上查看每一个景点的等待时间，优化用户体验。（使用户能根据等待时间选择景点。我去过一些景点，游玩时间只有5分钟，排队却花了2小时，还带来其他问题，比如排了90分钟队后很多人需要去上厕所。）
- 更多餐厅，至少有一家高档餐厅。
- 创建更多可以随时进入的景点，比如像缆车和摩天轮景点这样，游客可以在它们运行时进入。（简单说就是，景点不需要每5分钟停下来一次，让所有人出来，再让新来的人进去，而是允许游客持续进出。这样当然会减少等待时间。）
- 根据历史趋势，向用户推荐最佳参观日期，以避开人潮（并相应地调整票价——在“低潮日”降低票价）。

实时优化酒店房间价格的软件

如果你在酒店预订5个晚上，通常每晚（对于同一房间）的价格都不一样。价格的变化基于很多因素，包括以下。

- 季节。

- 一周中的哪一天。
- 竞争对手定价。
- 可用库存。

我在想定价模型里还可以包括哪些其他指标。例如，是否可以包括天气预报、特殊事件、价格弹性？价格是否可以根据不同用户而变化（例如，根据IP地址），或者根据购买方式（在线购买、通过电话、使用美国运通卡或Visa卡），根据提前多长时间购买，等等？

例如，提前很长时间购买会得到较低的价格，但这是因为库存量高。所以你可能只需要使用可用库存即可。

最后，这些定价模型多久更新一次？以及如何评估模型性能？价格是在实时变化吗？

预测税务审计风险的网络应用程序

你被审计的概率有多大？数据科学如何帮助我们每个人回答这个问题？

确定一些因素会增加被审计的概率，包括：

- 高收入。
- 个体经营与有限责任公司的合伙人。
- 归档早。
- 既有W2s的收入又有自我雇用的收入。
- 连续4年经营亏损。
- 在旅行和餐厅类别扣除的费用比例>40%。
- 大额捐赠。
- 家庭办公高额扣费。

Deloitte和KPMG这样的公司可能会为他们的大客户相当准确地计算出税务审计风险（包括可能的审计处罚），因为他们能访问大型内部数据库，接触到被审计和未被审计的客户资料。

但对你我来说，数据科学如何帮助我们预测审计风险？是否有可用的公开数据能用来建立一个预测模型？建立一个网站来帮助人们计算税

务审计风险是个好的创业想法。其风险评分引擎将基于以下3个部分。

- 用户可以匿名回答关于他们纳税申报单的问题，以及以前的审计情况（如果有的话），或提交一份匿名的纳税申报单。
- 收集和分析用户数据，给用户提供关于他们审计风险的实时答案。
- 随着越来越多的人使用该系统，预测会更准确，评估审计概率的置信区间会更小。

事实上，这相当于对美国国税局的算法进行逆向工程。

评估税务审计风险时，应使用哪些指标？预测的准确程度如何？这样的系统能达到99.5%的准确率吗？即只有0.5%的纳税人会得到错误的预测吗？目前，如果你告诉某人“你今年不会被审计”，只有98%的时候是正确的。更有趣的是，对于高风险的纳税人可以达到什么程度的准确率呢？

最后，如果你的模型能够预测审计风险和相应的审计处罚，你就可以更明智地决定哪些风险可以承担，哪些应该规避。这是避免被国税局扣费的纯决策科学，不是利用税收法律的漏洞（那样做很危险），而是通过诚实、公平和聪明的分析——赢了美国国税局的数据科学算法，而不是赢了税收法律。

# 总结

本章探讨了如何成为一名数据科学家，介绍了目前已有的大学学位、证书培训课程和由数据科学中心提供的在线实习培训。

此外，还讨论了不同类型的数据科学职业道路，包括企业家、咨询顾问、个人从业者和领导者，并提供了一些数据科学创业公司的设想，供想成为企业家的人参考。

第4章选择了一些数据科学技术进行详细介绍，但并没有引入不必要的数学知识。内容包括可视化和统计技术方面的信息，以及大数据和新型数据相关的度量指标。

# 第4章
# 数据科学的技术（I）

这是本书最具技术性的章节。本章先讨论一些指标选择的细节，再讨论用新的可视化技术表示复杂的、随着时间推移而不断变化的空间过程。然后，深入到一系列主题的技术方面。

本章介绍了一些最近开发的、能处理现代商业数据的有趣技术。本章比前几章更具技术性，但如果你有些统计学、数学或计算机科学的知识，阅读本章会很容易。本章选择的主题是从我发表在Data Science Central的各类文章，根据分析人员所认为的有用程度排名而确定的。本章尽可能广泛地涵盖了许多不同的技术和概念。

高级统计，特别是交叉验证、稳健统计、实验设计，是数据科学的一部分。同样，适于处理现代、大量、非结构化流数据集的算法本身的计算复杂度，它也是数据科学的一部分。因此，本章在一个较高的水平上讨论了许多计算机科学技术和统计技术。

注意　统计教材里通常能找到的内容（如通用线性模型），以及普通老旧的计算机科学概念（如排序算法），本书不予涉及。这些主题的介绍请查阅第8章“数据科学资源”。

## 新型指标

识别强大的指标，即有着强大预测能力的指标，是数据科学家的基本任务。下面的章节中提出了一系列的、用于优化数字营销活动和欺诈检测的指标。在第6章的“特征的预测能力：交叉验证”一节中定义了预测能力。衡量一个商业模式的业务成果的指标被称为KPI（关键绩效指

标）。用来预测趋势的指标被称为领先指标。

一些不是KPI的指标仍然需要确认和追踪。例如，如果你开展电子邮件活动推广以增加会员数量，两个KPI是每月新增会员数量（基于电子邮件发送的数量）和会员流失数量。但你仍然需要追踪打开率，甚至要按照ISP、分段和消息发送细化打开率。因为虽然它们不是KPI，但这些指标是了解新成员的变化和提升新成员的状态/减少流失的关键。例如，分析打开率时会发现在星期六晚上发送电子邮件的有效率比在星期三的早上发送的有效率低很多；或者没有一个Gmail邮箱用户会打开你的信息——这是一个相对容易解决的问题，但必须先检测到这个问题。如果你不能检测到它，则不能解决它；如果你不通过ISP（互联网服务供应商）追踪打开率，则会检测不到它。

大数据和快速集群式计算机已经可以追踪和计算更复杂的指标，且可以一次性测试成千上万的指标（以确定最佳的预测因素）和生成复杂的复合指标：其他指标的函数的指标，例如IP地址的类型。 这些复合指标有时是简单的转换（如对数转换），有时是复杂的组合和多个基本指标的比率，有时是使用复杂的聚类算法，由基本指标衍生的指标，如前面的一个例子：IP地址的类型　（将在本章的“网络拓扑映射”一节中讨论）。

复合指标和基本指标的关系就如分子和原子的关系。就像7个原子（氧、氢、氦、碳、钠、氯、硫）能产生数万亿的分子和化合物（对于分析和计算化学家设计分子以治疗癌症是一个挑战）。同样地，当从基本的指标变到复合的指标时，会发生组合爆炸。基本指标是数据集中发掘的原始领域。

下面的指标列表（主要是复合指标）是不全面的。你可以以它为起点思考如何创造新的指标。

# 优化数字营销活动的指标

如果开展在线新闻简讯推广，这里有很多需要追踪的指标。

- 打开率：订阅人数与打开你所发送的时事通讯的人数比例。低于10％即很差，除非你的e-CPM很低。
- 打开次数：一些用户会多次打开你的邮件。
- 打开2次以上的用户：这些人是潜在的领导者或者竞争对手。如果打开邮件超过一次的用户很少，那么你的内容是无趣的，或者在你的时事通讯里只有一个可点击的链接。
- 点击率：每次打开的平均点击次数。如果不到0.5次，你的主题可能很好，但与内容（主体）无关。
- 每个邮箱客户端细分的打开率（如雅虎、Mail、Gmail、Hotmail等）：如果Hotmail用户的打开率较低，你应该考虑清除你的邮件列表中的Hotmail用户，因为他们会影响你的整个列表。
- 细分到每个用户段的打开率和点击率。
- 趋势：打开率、点击率，以及它们在每个分段随着时间的推移是上升还是下降？确定最佳执行分段。向不同的分段发送不同的时事简讯。
- 退订和流失率：什么样的主题或内容会增加退订或投诉率？
- 垃圾邮件投诉：应当做到发送1 000条信息，投诉不到1条。确定产生高投诉率的分段和客户（例如，Hotmail），并删除它们。
- 地理位置：你有过在印度增长，但在美国却萎缩的情况吗？在尼日利亚的打开率更高吗？这不是一个好迹象，即使总体趋势看起来不错。
- 语言：你有很多西班牙裔的订阅者吗？如果有的话，你能给这些人发西班牙语的时事简讯吗？你能识别讲西班牙语的人吗？（如果你在用户注册时，问一个关于语言的问题，你就可以确认。）
- 一周中每天和每个时间段的打开率：确定发送时事简讯的最佳时间。
- 用户分段：向新订阅者提出许多问题，例如，他们的兴趣。使这些问题可选。这将让你能更好地瞄准你的订阅者。
- 增长加速：你达到饱和点了吗？如果达到了，你需要找到新的用户来源或降低你的邮件群发频率（可能找到更少但更好、更相关的广告主来优化e-CPM）。
- 低打开率：是图像导致了低打开率吗？是重定向（用于追踪点击）导致了低打开率吗？有些URL虽然很有用，如bit.ly，但可能导致低打开率，或人们由于存在计算机感染的风险而不点击链接。
- 关键词追踪：你有没有通过追踪主题中那些效果好或差的关键词以增加打开率？
- 寄信人字段：你有没有试着改变你的寄信人 字段，看看哪种效果

最好？A/B测试可以帮助你获得更好的结果。
- 信息的大小：如果太大，这可能会导致性能问题。
- 格式：文本或HTML（超文本标记语言格式）？做一些A/B测试能找到最佳格式吗？

你应该注意到，*e-CPM* 是每千次曝光所产生的收入。这是你最重要的指标，与用户流失同等重要。

# 欺诈检测的指标

很多数据科学家致力于欺诈检测。自1995年以来，处理欺诈检测的指标已被细化，所以现在有强大的新指标可用。现代欺诈检测指标的讨论重点在网络日志数据上。而这里介绍的指标可以作为规则使用。如果你使用这些指标来构建报警系统，则必须触发多个规则才能启动报警系统。你可能使用一个系统，如隐藏决策树，为每一个规则分配一个特定的权重。

- 极端事件：用蒙特卡洛模拟来检测极端事件。例如，一个非代理IP地址的大簇，每天有8次点击，日复一日。这一事件自然发生的概率是多少？
- 黑名单和白名单：判断一个IP地址或推荐人是否属于黑名单或白名单。例如，将IP地址空间划分成主要的簇，如静态IP、匿名代理、公司代理（白名单）、教育代理（高风险）、高回收的IP（更高风险）等。
- 推荐人统计：按时收集以下统计数据——负荷及其方差（根据3次测量结果），页面大小及其方差（根据3次测量结果），或在网页上找到的文本字符串（在HTML或JavaScript代码中）。例如，创建一个可疑条款（*Viagra* 、网上赌场 等）的列表，或创建一个可疑JavaScript标签或代码的列表，但是使用推荐人的白名单（例如，顶级出版商）来消除误报。
- 分析域名模式：例如，一串拥有完全相同的欺诈评分的域名，都是xxx-and-yyy.com的形式，且它们的网页都有相同的大小（1字符）。
- 关联分析：例如，占很大比例（大于30%）的流量，但会话时间非

常短（小于15秒），且有着2个或更多未知的来源（即来源除了Facebook、谷歌、雅虎或者其他前500的域名）。聚合这些会话中所有神秘的来源，有机会发现他们都是同一个僵尸网络的一部分（例如，用于点击欺诈的僵尸网络）。

- 信用卡字段不匹配：例如，电话号码在一个国家，代理域名的电子邮件或IP地址为另一个国家的人所拥有，物理地址仍然在别的地方，名称（例如，Amy）和电子邮件地址（例如，joy431232@hotmail.com）看起来不同，谷歌搜索电子邮件地址，显示了来自同一账户以前的诈骗操作，或什么都不显示。
- 无用数据：例如，一个连接到付费点击的查询网页或搜索关键词，包含无用数据或由键盘上相邻字母组成的文本字符串，如*fgdfrffrft*。
- 电子邮件地址异常：例如，电子邮件地址包含数字，除了区域代码、年份（例如，73），或邮政编码（除非邮件来自印度人或中国人）之外的数字。
- 交易时间：例如，如果是注册后的第一次交易，则交易时间短。
- 异常采购模式：例如，在星期日凌晨2点，一名顾客在目标顾客是美国人的B2B电子商店购买最贵的产品，但其IP地址是来自美国以外的地区。
- 重复的小额采购：例如，当多个9.99美元商品的购买操作是来自相同商品类别的多个其他商家时，且每个持卡人有1个或2个交易。

这里的关键点是，欺诈检测指标变得更加复杂和多样化，因为犯罪分子不断尝试通过逆向工程欺诈检测算法以获取竞争优势。于是，从事欺诈检测的数据科学家将永远有工作要做。欺诈检测也包括和非罪犯分子斗争的指标——例如，营销者要与搜索引擎博弈，以使他们的网站在谷歌上排名第一。同理，剽窃者和垃圾邮件发送者也是一样。

# 选择合适的分析工具

作为一名数据科学家，你会经常使用现有的工具而不是重新发明做无用功。所以，在深入到技术性工作之前，重要的是你要为工作确认并选择合适的工具。当寻找一个供应商购买分析软件、可视化工具、实时产品和编程语言时，以下是一些你需要询问的重要问题和需要牢记的准则。

# 分析软件

这里有几个很好的准则和问题要考虑，将有助于你选择适当的分析软件。那些我认为非常重要的，用星号标记了。

- 它是一个新的产品或公司，还是已经颇具规模了？
- 它是开放源码的吗？如果不是，可以免费试用吗？
- 它可以处理大数据吗？*
- 价格。*
- 和其他产品兼容。
- 容易使用和提供GUI（图形用户接口）。*
- 可以在批量处理或可编程的模式下工作。*
- 提供API（应用程序接口）。
- 能够在网络上获取数据或从数据库（支持SQL（结构化查询语言））获取数据。
- 漂亮的图形功能（例如，可视化决策树）。*
- 计算速度和使用内存方式的效率。*
- 良好的技术支持、培训或可用文档。*
- 本地公司。
- 提供全面的现代化程序集。
- 通过额外收费，提供附加组件或模块（这可能是重要的，因为你的需求在增加）。*
- 平台独立。*
- 技术专业型（例如，时间序列、空间数据、评分技术），还是通用型。
- 领域专业化（例如，网络日志数据、医疗数据和财务）。
- 与你的客户端兼容（例如，SAS，因为你使用的客户端是SAS）。*
- 相同分析领域的从业者也使用（定量、计量经济学、六西格玛、运筹学、计算机科学等）。
- 处理丢失的数据、数据清理、审计功能和交叉验证。
- 支持诸如C++、Python、Java、Perl等编程语言，而不是内部的、特设的语言（尽管特设的语言可能更有效）。*
- 学习曲线不是太陡。*
- 容易升级。*
- 能在云上运作，或可以使用MapReduce和NoSQL的特性。

- 实时特征（可以集成在一个实时系统里，如拍卖竞价）。

在某些情况下，可能需要3个软件产品：一个是可视化，一个是具有大量的处理功能（神经网络、决策树、约束的逻辑回归、时间序列和单纯形）和一个可以为你进行海量高强度分析提供算力的设备。

你应该询问的相关问题还包括：

- 你打算什么时间为现有的产品升级或购买额外的模块，例如，在SAS基础上选用SAS/GRAPH或SAS/ACCESS模块？
- 在你的组织中谁负责对选择分析软件做出最后的决定？终端用户？统计学家？商业人员？首席技术官（CTO）？

# 可视化工具

问自己以下问题，以帮助你为每一个项目选择适当的可视化工具。

- 你如何定义和衡量图表的质量？
- 你使用哪些工具能产生交互图表或地图？
- 你推荐使用哪种大数据可视化的工具？
- 哪些可视化工具可以通过API在批处理模式下访问（例如，要每5分钟更新地震图或每秒钟更新股票价格）？
- 你是如何看待Excle的？Python或Perl图形库，你又是如何看待的？R呢？
- 是否有任何工具，可以让你轻松地制作数据视频（例如，显示欺诈案件或疾病是如何随着时间的推移而蔓延的）？
- 在Excel中更新你的数据，模型和图表也会随着自动更新。是否有任何Excel的替代品，提供相同的功能，但有更好的数据建模能力？
- 如何制作漂亮的图表结构——例如，形象化地显示Facebook联系人？
- 热力图是什么？什么时候使用它才合乎情理？
- 你如何绘制力导向图？

- 这个工具对于光栅图像、矢量图像、图表、决策树、分形、时间序列、股票价格、地图、空间数据等，是否是一个不错的选择？
- 如何整合R与其他图形包？
- 在一个简单的二维图里，如何表示5个维度（例如，时间、数量、类别、价格和位置）？如果你的目标只是传达信息给主管，那么表示更少的维度是否更好？
- 为什么数学家和运筹学从业者使用的可视化工具（例如，Matlab）不同于数据科学家使用的工具呢？是因为数据类型，还是只是历史原因呢？

# 实时产品

这些问题将有助于你更容易地使用和选择实时产品。

- 除了交易评分（信用卡交易和在线广告投放系统），还有其他的应用需要真正实时吗？
- 什么类型的应用能准实时更好地运作，而不是真正实时？
- 如何提高真正实时评分的性能？例如，通过预载入、持久、内存操作、小的查找表（每小时更新）或其他机制？解释为什么。
- 由于高峰时间的服务器负载是其他时期负载的10倍，比如凌晨2点，应该如何处理？在凌晨2点，你是否使用空闲的服务器进行算法定期（每小时或每天）改进或调整实时评分？
- 数据科学家选择和整合实时算法到成果里，主要是基于它们可以在分布式环境中很容易地部署吗？
- 你如何解决移动的观察点的实时三维流视频处理这样的困难问题（以实现在拥挤的天空中大飞机的低海拔自动飞行）？
- 你认为终端用户（例如，决策者）需要的是得到真正实时更新的显示面板，还是提供延迟5分钟的统计数据给终端用户，哪种会更好？实时在哪些应用中对终端用户更好？
- 实时仅限于机器生成数据吗？
- 什么是机器生成的数据？一个基于最近的甚至刚更新的Twitter消息或Facebook帖子的实时交易系统怎样工作？你是否把大数据、机器数据、机器和机器的对话等都称为实时的？
- 真正实时信号相对于5分钟延迟信号的获益是什么？这些益处（增

加准确性）是否超过额外的成本？（在华尔街上，以上问题的答案通常是肯定的。但就关键词相关的竞价算法而言，延迟的反应可以吗？）

- 是否有就具体应用而言的最佳延迟时间（当无法实施实时操作时）的判定规则呢？例如，对于互联网流量监测，15分钟是合适的，因为它涵盖了大多数用户的会话。
- 什么样的编程环境较适合实时大数据？（SQL不适合，C++更适合，那Hadoop怎么样？你会用什么技术？）
- 什么样的应用程序较适合实时大数据？
- 什么类型的指标（例如，最后交易的时间）容易或大量地在实时中使用？
- 为实现平稳的反馈点击数和即时的欺诈捕捉，你认为对谷歌来说唯一的解决方案是真正实时监测点击数（付费广告）吗？
- 你认为Facebook会使用真正实时的广告定向投放吗？如何提高非常低的曝光点击率？尽管事实上它知道很多关于其用户的情况，为什么这个比率还是如此低？技术能帮助它提高吗？
- 在未来10年，实时会是什么样的？什么将成为实时？对于以每天结束一次运行的系统来说，还有哪些仍会按小时更新？
- 是不是所有的实时系统实际上是混合的，依赖于每小时、每天甚至每年（有季节性）的组件以提高其精度？实时预测是如何预测高度稀疏的粒度数据的？例如，为任何的广告主实时预测任何广告关键词的收益率。（答案：将稀疏的数据分类到更大的桶里，对整个桶进行预测。）

# 编程语言

Python和R已经非常流行了：它们是开源的，所以你可以从网络下载它们，获取示例脚本，并开始编程。Python提供了分析库，被称为*Pandas* 。R也非常流行。Java、Perl和C++已经失去了一些人气，但仍有人用C编程（在金融领域常见），因为它快速，且有大量的可供分析及可视化和数据库访问的库。Perl非常好用而且灵活度高，尤其适用于那些具有独立性、创造性且喜欢编写代码的数据科学家，还无须担心诸如句法、代码版本控制、团队协作或者客户如何整合你的代码等问题。Perl还配备了图形库。

当然，你还需要了解SQL。Hadoop已经成为MapReduce编程的标准。SAS已经在很多环境中普遍应用，特别是政府和临床试验。我有时会使用R绘制先进的图形（包括视频），因为它能与调色板和其他图形功能一起很好地运作。R曾经因为计算机的可用内存受到限制，但利用云可以解决（RHadoop）。

# 可视化

本节不是可视化工具或准则的回顾。这样的回顾超出了这本书的范围，并且可视化的回顾将会自成一书。读者可以查阅本书第8章的部分，以供参考。你可以在`http://www.datasciencecentral.com/group/data-science-apprenticeship/forum/topics/additional-topics-for-the-data-science-apprenticeship` 找到更多关于如何选择可视化工具的信息。

本节将教你用R如何制作高级图表，如何制作关于数据或数据视频信息的视频。它还包括编程生成任何你想要的颜色图案，因为*rgb* （红/绿/蓝）的R函数，嵌入在一个R指令 *plot* 里。通过创建视频，而不是图片，你可以为可视化自动添加更多的维度——该维度通常是（但不限于）时间维度，形成一个时间序列。所以，本节是教你如何用R生成数据视频的教程。最后，你可以将数据视频上传到YouTube上。在这里选择讨论R，是因为它非常容易使用，而且它是当今数据科学家使用的最流行的工具之一。

# 用**R**生成数据视频

下面是逐步制作视频的过程。前3节会讲解如何去操作。然后，随后的章节提供了额外细节，对你可能会很有帮助。

视频是很重要的，因为它们可以让你随着时间的改变（时空数据）表达空间信息，或显示一个额外维度，否则就是静态的图像。它们也可以用来查看数据科学算法的收敛性（或缺乏收敛性）和收敛速度。Song-Chun Zhu是制作“数据视频”的开拓者之一（见于

`http://www.stat.ucla.edu/~sczhu/publication.html` ）。

视频的另一个有用的方面，是显示可旋转的三维图表的能力，这样用户可以从各个角度看图表，并发现传统的投影（静态图像）无法显示的模式（除非你查看几十个不同的投影）。基本原则是：

- 决定你将用哪些指标作为你的时间变量（我们称之为“时间”）。
- 框架移动速度多快。
- 粒度时间是多少（你来决定）。
- 生成数据框（使用R、Python或Perl，每次生成一个）。
- 显示框架（使用R很容易），且/或把它们嵌入到视频文件（使用开放源码的ActivePresenter）。

有两个重要的问题，一是如何把数据可视化，二是应该由谁做出这一决定——数据科学家、业务分析师或商务用户。通常是商务用户决定，但是数据科学家也应该提出新的可视化方案给商务用户（取决于分析的数据类型和公司使用的关键绩效指标），且让商务用户根据它能从一个可视化中提取多少价值，以及提取该值需要多少时间，而做出最终的决定。

最好的可视化可以将商业价值连接到关键绩效指标，或解决一个昂贵的问题，或者显示一个巨大的创收契机，而且能在30秒或更少的时间内被大家理解。最糟糕的图表有一个或多个以下特征：（1）数据混乱，（2）数据尺度比例不当，（3）不重要的数据掩盖或隐藏的有价值的信息，以及（4）显示的指标是不可比较的。

## 生成你想要可视化的数据

使用Python、R、Perl、Excel、SAS或任何其他工具，生成一个文本文件，称为`rfile.txt` ，有4列。

- k：框架号。
- x：x坐标。
- y：y坐标。
- z：颜色，与（x, y）相关联。

或者，如果你想完全复制我的实验，你可以在该 http://www.datashaping.com/first.r.txt 网站下载我的rfile.txt （样本数据可视化）。

## 运行下面的**R**脚本

请注意，R脚本的第一变量是iter 。它与一个迭代算法相关联。迭代算法在每次的迭代中，生成500个（x, y）的可观测和更新的数据集。第4个字段称为new （新的），它表示点（x, y）对于给定的数据点和给定的迭代是否是新的。新的点出现在红色区域，旧的点出现在黑色区域。

```
vv<-read.table("c:/vincentg/rfile.txt", header=TRUE);
iter<-vv$iter;
for (n in 0:199) {
  x<-vv$x[iter == n];
  y<-vv$y[iter == n];
  z<-vv$new[iter == n];
  plot(x, y, xlim=c(0,1), ylim=c(0,1), pch=20, col=1+z,
xlab="", ylab="", axes=FALSE);
dev.copy(png, filename=paste("c:/vincentg/Zorm_", n,".png", sep="
  dev.off ();
}
```

## 制作视频

制作视频有4种不同的方法。当运行R脚本时，将发生下列情况。

- 在大约30秒内生成200幅图像（散点图），按顺序显示在R图形窗口上。
- 200幅图像（循环或散点图）以Zorm0.png 、Zorm1.png 等的格式保存在目标目录里，直到Zorm199。

然后，有3个选项来制作视频，如下。

1. Caveman类型：用手机抓拍R图形框架序列形成视频文件。

2. 半Caveman类型：使用屏幕抓图工具（例如，ActivePresenter）来捕捉在R图形窗口中显示的流动点。

3. 使用Adobe或其他软件自动组合由R生成的200个Zorm*.png 图像。

你可以在http://stackoverflow.com/questions/1298100/creating-a-movie-from-a-series-of-plots-in-r 找到其他可能的解决方案（如开源ffmpeg或ImageMagick库）。你可以看到《统计软件杂志》在2013年4月发表的动画“用R程序包创建动画和演示统计方法”。

制作出这些录像后一个很有意思的事情是你可以将它们上传到YouTube。你可以在下面这些网址查看我的视频。

- https://www.youtube.com/watch?v=DL4cYD2X9PM
- https://www.youtube.com/watch?v=fb0yevGSAxY
- https://www.youtube.com/watch?v=QBQIHbQdptk
- https://www.youtube.com/watch?v=4oIsB0EdnmQ

## **ActivePresenter**软件

你可以使用ActivePresenter（http://atomisystems.com/ ）屏幕抓图软件（免费版），如下。

- 让200幅图像自动快速地显示在R图形窗口中。
- 使用ActivePresenter选择你要抓取的区域（在R图形窗口的一部分中，就像屏幕截图，不同的是这里抓取的是流动的内容而不是静态的图像）。
- 完成后点击停止，并将其导出为.wmv 格式，然后上传到网络服务器上，供其他人访问。

我最近使用ActivePresenter创建了两个高质量的视频，它们在数据集上增加了两列内容，所用的R脚本文件如下（你可以在http://www.datashaping.com/rfile2.txt 下载7MB文本文件数据）。

```
vv<-read.table("c:/vincentg/rfile.txt", header=TRUE);
iter<-vv$iter;
for (n in 0:199) {
  x<-vv$x[iter == n];
  y<-vv$y[iter == n];
  z<-vv$new[iter == n];
```

```
  u<-vv$d2init[iter == n];
  v<-vv$d2last[iter == n];
  plot(x, y, xlim=c(0,1), ylim=c(0,1), pch=20+z, cex=3*u,
col=rgb(z/2,0, u/2), xlab="", ylab="", axes=TRUE);
  Sys.sleep(0.05); # sleep 0.05 second between each iteration
}
```

你也可以尝试改变plot 指令，如：

```
plot(x, y, xlim=c(0,1), ylim=c(0,1), pch=20, cex=5, col=rgb(z,0,0
xlab="", ylab="", axes=TRUE);
```

第2次的R脚本与第1次的不同之处如下。

- dev.copy 和dev.off 调用被移除，故不再在硬盘驱动器上生成PNG图像。（因为你使用了屏幕抓图，所以不再需要这些PNG文件）。生成PNG文件拖慢了整个过程并且产生了视频闪烁——上述这一步消除了大部分的视频闪烁。
- Sys.sleep 的功能是使每个框架之间有短暂的停顿，使得视频更流畅。
- rgb 用在绘图指令里，给每幅图像指定一种颜色：（x, y） 在每次迭代时，都会获得指定的颜色，是z 和u 的函数。
- 现在，在绘图指令中点的大小（cex ）取决于变量u ：这就是为什么你看到各种规模的泡泡在变大或缩小。

请注意，d2init （用于制作视频，在rfile2.txt 的第4列输入数据）是当前迭代（x, y） 的位置和它第一次迭代的位置之间的距离。此外，d2last （在第5列）是当前和先前迭代每个点之间的距离。如果点在当前和先前的迭代之间进行大移动，则该点将被着色为深蓝色。

rgb 函数接受3个值在0和1之间的参数，分别代表红色、绿色和蓝色的频率强度。例如，rgb（0,0,1） 是蓝色，rgb（1,1,0） 是黄色，rgb（0.5,0.5,0.5） 是灰色，rgb（1,0,1） 是紫色等。确保0≤rgb≤1，否则它会崩溃。

# 更复杂的视频

还可以创建点和箭头（不只是点）连续地从一个帧移动到另一个帧的视频。箭头附着点，并指示在任何给定的时间里点朝向哪个簇移动或远离。从帧到帧的转换是随着时间而加速的，且也运用了颜色梯度。

我最近制作了几个这样的视频。你可以在以下网址找到更多视频的信息。

- 在http://www.datashaping.com/rfile3.txt 下载数据集。
- 在http://www.analyticbridge.com/profiles/blogs/shooting-stars 可观看视频。

下面是该版本中使用的R源代码。

```
vv<-read.table("c:/vincentg/rfile3.txt", header=TRUE);
iter<-vv$iter;
for (n in 1:199) {
  x<-vv$x[iter == n];
  y<-vv$y[iter == n];
  z<-vv$new[iter == n];
  u<-vv$d2init[iter == n];
  v<-vv$d2last[iter == n];
  p<-vv$x[iter == n-1];
  q<-vv$y[iter == n-1];
  u[u>1]<-1;
  v[v>0.10]<-0.10;
  s=1/sqrt(1+n);
  if (n==1) {
    plot(p, q, xlim=c(-0.08,1.08), ylim=c(-0.08,1.09), pch=20, ce
col=rgb(1,1,0), xlab="", ylab="", axes=TRUE  );
  }
  points(p, q, col=rgb(1-s,1-s,1-s), pch=20, cex=1);
  segments(p, q, x, y, col=rgb(0,0,1));
  points(x, y, col=rgb(z,0,0), pch=20, cex=1);
  Sys.sleep(5*s);
  segments(p, q, x, y, col=rgb(1,1,1));
}
segments(p, q, x, y, col=rgb(0,0,1)); # arrows segments
points(x, y, col=rgb(z,0,0), pch=20, cex=1);
```

# 无模型的统计建模

统计学的“无模型建模”是一种不使用统计分布或模型做推理，甚至预测或计算置信区间的方法。有时对系统物理模型的了解（例如，股票市场）相比搭建可行的交易系统来说并没那么重要。有时（如在股票市场），你可以尝试仿真这个系统（使用蒙特卡洛仿真来模拟股市价格），即使在不知道系统的物理机理和理论的情况下，也可以通过数据来仿真这个系统，只要它能运行且能符合条件或环境（例如，市场状况）的变化。

你可能想知道如何获得所需的数据，用以制作先前讨论过的视频。本节介绍了用于制作视频数据的算法细节。我会用射击明星视频为例，这意味着你需要了解一点该项目的历史。

该项目始于个人兴趣和研究结果，包括以下：

- 对天文学、可视化方面和物理模型应用于商业问题方面的兴趣。
- 对聚类系统（以及簇生成和消亡的过程）的兴趣，以及对用簇信息可视化显示而不是图表显示的兴趣。
- 如何用万有引力定律对城市增长建模的研究。
- 对使用数据同步生成有声音和图像的视频的兴趣。

视频背后的统计模型是簇生成和消亡的过程——“引力”产生了簇的生成和消亡——点从一个簇移到另一个簇的过程，和点从远方被吸引过来的过程。最终，它成了无模型建模 ，是数据科学的流行趋势。

# 无模型的统计建模是什么

用来创建视频数据的算法背后通常有一个通用的数学模型，但在首次创建这一算法时可能尚未顾及其数学模型。这表明了数据科学的一个新趋势：越来越少关注建模且越来越多关注结果。

我的算法有一堆参数和特征，可以通过微调来产生你想要的任何东西——可以是一个Neyman-Scott聚类过程的模拟或一些没有名字的随机过程的模拟。这有点类似于现代登山的演变：从过去专注于攀登名山，如珠穆朗玛峰，到如今探索更深的荒野和攀登没有名字的高峰（他们自

己的挑战），且未来有可能到火星爬山。

你可以微调参数：

- 使用传统的拟合优度检验和敏感性分析，实现了模拟数据与实际业务（或其他数据）之间的最佳拟合。请注意，该模拟数据代表了一个时空随机过程的实现（面向对象的实例）。
- 在校准参数后，进行预测（统计学家的语言）或外推（数学家的语言）。

# 该算法是如何工作的

该算法起始时有m个移动点在[0,1]×[0,1]的正方形窗口随机分布。这些点彼此吸引（靠得越近，吸引力越强），从而随着时间的推移，它们集合成聚类。

该算法有以下几个组成部分。

- 在[–0.5, 1.5]×[–0.5, 1.5]窗口内，生成n（n=100）个随机固定点。此窗口是包含移动点的窗口的4倍大，以消除影响移动点的边缘效应。这些固定点（它们从不移动）也表现为某种暗物质：它们是不可见的，它们没有在视频中体现出来，但它们是防止整个系统崩溃和防止系统聚到单一点的黏合剂。
- 在[0,1]×[0,1]窗口内，创建m（m=100）个随机移动点。
- 主循环（200次迭代）。在每次迭代中，计算每一个移动点（x, y）与它的m−1个移动邻居点和n个固定邻居点之间的距离。权重w是d的函数，对于点（x, y）本身有特别的权重。然后，通过对所有点的加权求和更新点（x, y）。在每次迭代中，对于每一个点（x, y）都需要这样做。权重的限制条件是所有点的权重和为1。换句话说，用一个凸的、包含所有点的线性组合来替换每一个点。

该算法的一些特殊特征，包括以下几个。

- 如果在一个给定的迭代过程中点（x, y）（正在更新）的权重十分高，那么（x, y）将几乎没有移动。
- 当测试负权重值（特别对于正在更新的点）时，结果会更好。一个微妙数量的负权重值，也可以防止系统崩溃，并引入一些可取的混沌效应。
- 有时候，一个点会被一个全新的随机点所代替，而不是用邻近点的加权总和来更新。这被称为“生成”。它发生的概率不到所有点更新的1%，并且在开始阶段发生更频繁。当然，你可以尝试自己调整这些参数。

如Perl的源代码所示，点$k 的生成过程可以简单地编码为：

```
if (rand()<0.1/(1+$iteration)) { # birth and death
  $tmp_x[$k]=rand();
  $tmp_y[$k]=rand();
  $rebirth[$k]=1;
}
```

在源代码中，在对$k 的内循环中，需要更新的点（$x,$y） 用$k 表示，即（$x, $y） = （$moving_x[$k], $moving_y[$k]） 。此外，更深一层，在对$l 的循环中，点（$p,$q） 用$l 表示，是用来更新（$x,$y） 的加权平均公式中用到的邻居点。使用distance 函数计算距离d，该函数接受4个参数（$x 、$y 、$p 和$q ），返回权重w的值$weight 。

总之，用于研究现实机理的仿真模拟可以帮助你理解和在实验室复现这些机理，但这些机理本身并没有被充分理解。在这种情况下，这些模拟（视频中很直观）可以帮助你了解多个天体的各种尺寸、速度和起源（或任何与万有引力相似的力量——可能是在市场上的竞争力），包括它们如何合并、生成及消亡。反过来，通过拟合数据来模拟，即使在不是很了解这些机理的情况下，你仍可以进行预测。

# 源代码生成数据集

Perl和R都被用来生成制作视频的数据集。你可以在http://www.analyticbridge.com/group/codesnippets/forum/topics/fchaos-to-clusters-simulation-of-stochastic-processes 下载源代码。

Perl代码运行速度比R代码要快得多，不是因为Perl本身更快，而是因为在程序中使用的算法架构：Perl一次性重新计算了所有的框架，并加载到内存中，而R只是按需生成每一个框架。R版本的优点是，它可以完成过程中的各方面内容，包括生成数据和显示视频帧。

# 三类指标：中心性、波动性、颠簸性

本节标志着进入一个更深层次地研究更多技术性数据科学问题的开始。让我们从一个简单的概念开始。统计教材注重中心性（中位数、平均值（average）或算术平均值（mean））和波动性（方差）。它们不提及第3个基本指标：颠簸性。本节首先考虑中心性、波动性和颠簸性之间的关系。然后，论述颠簸性的概念，以及如何使用。最后，你将看到一个新的概念是如何发展起来的，一个强大的现代定义将如何实行，以及如何在先前学科的基础上创建一个更有意义的定义，且和先前的科学兼容。

# 中心性、波动性和颠簸性之间的关系

两个不同的数据集可以有相同的中心性和波动性，但有不同的颠簸性。颠簸性与数据点如何排序相关，而中心性和波动性完全忽略了顺序。所以颠簸对有顺序的数据集是有用的——尤其是，时间序列。此外，颠簸性整合数据点之间的依赖概念，而中心性和方差则没有。值得注意的是一个时间序列可以具有较高的波动性（高方差）但低颠簸性。反过来也如此。

检测颠簸里的变化是很重要的。股票市场策略是一个经典的例子：一个算法暂时效果很好，然后颠簸性变化使它失效了。检测颠簸性变化发生，可以帮助你更容易地调整算法，有希望减少损失。颠簸性低与更强大、更准确的预测能力密切相关。如果可能的话，使用颠簸性低的数据，或使用颠簸缩减技术。

# 定义颠簸性

给定一个时间序列，一个直观的、数据科学的、尺度相关的、健壮的指标是所有顶点形成的平均锐角。该指标是有限制的，如下。

- 最大值是Pi，平稳的时间序列（直线）可以达到最大值。
- 最小值是0，当时间序列每个间隔都有极值且无限振荡时可以达到最小值。

该指标对异常值不敏感。它完全是一个现代的指标。然而，我并不想重新造轮子，因此我使用典型的“统计”（相对于“数据科学”）的指标来定义颠簸性。该指标具有与老式的平均值（测量中心性）或方差（测量波动性）类似的数学和理论上的优缺点。

颠簸性可以定义为滞后一个间隔的自相关函数。图4-1显示了3个时间序列，具有相同的算术平均值、方差和值，但颠簸性不同。

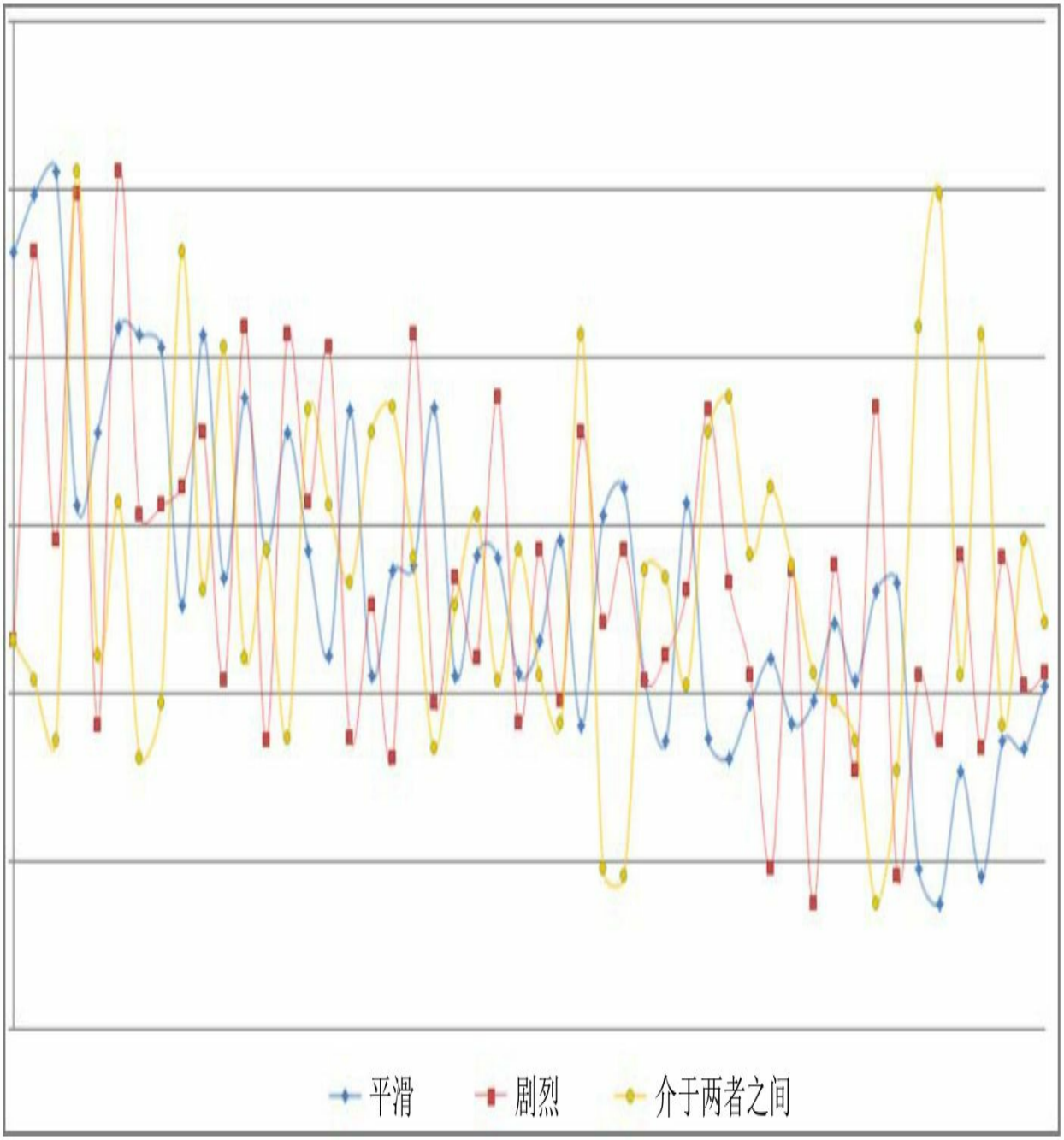


图4-1　3个时间序列，具有相同的算术平均值、方差和值，但颠簸性不同

请注意，滞后一个间隔的自相关函数的绝对值是所有自相关函数中最大的。因此，它是时间序列的自相关结构的唯一一个最佳指标。它的值总是在−1和1之间。对于平稳时间序列，它接近1；对于纯噪声，它接近0；对于周期时间序列，它是负数；对于具有巨大振荡的时间序列，它接近−1。通过如下伪随机序列你可以生成接近–1的r: x（1）, x（n）, x（2）, x（n–1）, x（3）, x（n–2）...其中x（k） [k = 1, ..., n]表示一组n

个点的次序统计，x（1）=最小值，x（n）=最大值。

一个更好但更复杂的定义，涉及一个所有自相关系数的权重逐次衰减的加权和。当该值为0时表示数据点完全独立，就大多数实际应用来说这是很有意义的。

# 在**Excel**中计算颠簸性

现在你考虑做一个Excel电子表格，显示各种时间序列的颠簸系数r的计算。另一个可能感兴趣的点是，借此你可以学到Excel的新概念，如用*RAND* 生成随机数字，*INDIRECT* 、*RANK* 和*LARGE* 的间接引用，以及其他功能强大但不知名的Excel函数。最后，这是一个由双核心参数驱动的、全互动的电子表格的案例。

你可以在`http://api.ning.com/files/Jx8KmtZz6StunetM0XVMvbi2IbLZ*2mhLOHEk`下载电子表格。它包含一个基础的（平滑，r>0）时间序列在第G列和其他4个由这个基础时间序列衍生出的时间序列。

- 颠簸在H列（r<0）。
- 中性在I列（r和0没有统计差异）。
- 极大值（r = 1）在K列。
- 极小值（r =–1）在M列。

两个核心参数可以微调：单元格N1和O1。请注意，即使时间序列是下降趋势的，r也可能是正的：r不代表趋势。相反，一个衡量趋势的指标是时间相关性（也在电子表格中计算）。

基于一组给定的数据点，可以通过对原始值（G列）的随机洗牌来获得一个中性时间序列（r = 0），同时保持原序列的平均值、方差等数值。它基于在B列中的伪随机排序，它本身的创建使用了Excel里面的*RAND* 和*RANK* 公式。它的理论框架基于分析桥第二定理。

分析桥第二定理

一组非独立数的随机排列构成一个独立的数列。然而，本质上这不是一个真正的定理，尽管它是一个相当直观且易于解释的基本概念。总之，无论原来的数据是什么，只要数据点越多，洗牌后的序列（使用随机排列）就越像随机数序列（一个预先设定的、通常是非均匀的统计分布）。通过计算一批模拟洗牌后数据的统计量，可以很容易地验证这个定理：所有这些统计量（例如，自相关）一致性地支持这样一个事实，洗牌后的统计量数值是（渐进地）相互独立的。

注意 你可以在http://www.analyticbridge.com/profiles/blogs/how-to-build-simple-accurate-data-driven-model-free-confidence-in 查看分析桥第一定理。

Excel有一些问题。特别是，它的随机数生成器一直被批判，每次更新电子表格都会重新计算随机数的值，但这个结果不能复制（除非你“冻结”B列中的值）。

# 使用颠簸系数

研究经济时间序列，常需要将序列按照高低颠簸性进行分段处理，同时研究其产生颠簸的机理并在第一时间检测出颠簸。在某些情况下，颠簸可能太小而肉眼看不到，但统计工具可以检测到。

另一个应用是高频交易。具有很大负颠簸性的股票（在很短的时间窗上）是统计交易的理想候选对象，因为它们提供了可控制的、可利用的波动性——不像颠簸性接近于零，相当于不受控制的波动性（纯噪声）。当然，很多正颠簸性的股票已经不存在了。它们在30年前就开始存在了：它们是投资者的面包和黄油，投资者永远持有股票或指数，看它会不会自动逐年增长。

问题

你如何推广这个定义到更高的维度——例如，到空间过程？你

可以有一个方向性颠簸的概念（南北或东西）。

潜在的应用：实时飞行路径的优化，以避免严重的颠簸气流（即较大的负风速和风向颠簸）。

# 大数据的统计聚类

聚类技术已经在第2章中探讨了。在这里，将描述一个额外的技术，来帮助大数据集聚类。它可以添加到任何聚类算法里。它由样本第一次创建的核心聚类组成，而不需要用到整个数据集。

比如说，你必须聚集1　000万个点——例如，关键词。假设你有一个相异性的函数，呈现为有1亿个条目的文本文件，每个条目由3个数据点组成。

- 关键词A。
- 关键词B。
- A和B之间的距离，记为d（A, B）。

简而言之，你可以执行k-NN，（K近邻算法）聚类或一些其他类型的聚类，从计算复杂度的角度来看，复杂度通常是O（$n^2$）或更复杂。

我们的办法是先抽取1亿个条目的1%（或更少），并针对这些关键词对进行聚类，形成“种子”或“基线”聚类结构。

下一步是按顺序浏览你的1　000万个关键词，对每个关键词从基线聚类结构里找到最接近的簇。如果有1　000个基线聚类，你必须执行100亿（10　000　000×1　000）次计算（哈希表访问），这是一个相当大的数目，但它用一个分布式架构（Hadoop）可以很容易地计算。

你可以重复整个过程5次（用5个不同的样本），然后将所得的5个不同的聚类结构混合在一起。最终输出可能是一个有3　000万个条目的表格，其中每个条目包括一对关键词A和B，以及A、B属于同一个簇

（基于5个不同样本的5次迭代）的次数（如果大于0）。最后的聚类检测算法包括从这3　　000万个条目里提取的连接组件。从图论的观点来看，这些条目被称为脊，它连接着两个节点A和B。（这里节点是关键词。）

这样的算法运行的条件是什么？你可以假设每一个关键词A都有2～50个相邻的关键词B，例如d（A, B）>0。这样的邻近点的数目通常和2更接近，而不是50。因此，最终如果对所有关键词进行分类，会有10%或20%的关键词被孤立（形成单点聚类）。

当然，如果你选择50个样本进行上述迭代而不是5个，或选择的样本容量是整体数据的25%而不是1%，则你可以解决上述问题。但这是一个时间上难以实现的方法，是一个糟糕且无效率的统计抽样方法。优化这些参数的方法是通过测试：如果算法运行得很快，需要很少的内存，但留下太多重要的、孤立的关键词，那么你必须增加样本容量或样本数，或两者的组合。相反，如果算法运行得很慢，则停止程序，减少样本容量或样本数。当你运行一个算法时，在屏幕上（或在日志文件中）显示程序运行状态是个好主意，如显示哈希表规模的增长（这样你会知道什么时候算法已用尽所有的内存，方便解决这个问题），以及程序所有步骤已经执行完毕等。简而言之，显示程序进展情况。例如：

```
已处理 10 000 个关键词和已创建545 个聚类;
已处理 20 000 个关键词和已创建690 个聚类;
已处理 30 000 个关键词和已创建738 个聚类
```

当程序开始运行，21秒后，第1行将显示在屏幕上；45秒后，第2行将显示在第1行下面；61秒后，第3行将显示；如此继续。窗口控制台显示符号>，表示程序正在运行的地方。有时，不是>而是$或像 `$c://home/vincentg/myprograms>` 这样显示。

在每一步结束后，通过调用时间函数，可以计算完成每一步所需的时间（以确定瓶颈）和测量所需时间的差异（详情见于 `http://stackoverflow.com/questions/415511/how-to-get-current-time-in-python` ）。

关键词聚类的一个有趣的应用如下：传统黄页有几百个类别和子类别。餐厅类别有子类别，通常和菜肴的类型相关：意大利餐、美国餐、寿司、海鲜等。通过分析和聚类在线用户的问题，你可以确定一个更丰富餐厅类别的子类别：菜肴的类型、餐厅的类型（酒吧、高档餐厅、酒

馆、家庭餐厅、风景餐厅、浪漫餐厅、民族餐厅、夜宵餐厅）、位置的类型（市区、河边、郊区、沙滩、山区），或与食物本身不相关的东西（餐厅工作、食谱、菜单、餐厅家具）。

注意　你应该优先考虑具有节点密度高的区域，然后假设这些区域被采样到的概率更高。

# 大数据的相关性和拟合度

在大数据情况下，你有时需要计算涉及成千上万个数据桶里的配对观测数据或时间序列数据的相关性。例如，数据桶可以对应一个决策树中的节点、一个客户分区或一个具有相同多变量特征的观测子集。特别有趣的问题包括多变量特征的选择（一个组合问题）或最佳预测指标集的辨别。

在大数据集，一些数据桶包含异常值或无意义的数据，桶的大小可能也不尽相同。你需要有比传统统计工具更好的工具。特别是，你需要一个相关性的指标，满足以下条件。

1. 与样本容量无关，允许比较不同大小的数据桶：例如，0.6的相关性须对应于在所有n个潜在配对序列（X, Y）的第74个百分位数，不管n的值是多少。

2. 与传统相关性有相同的边界以保证后向兼容性：它必须介于−1和+1之间，通过极值、奇异值数据集达到−1和+1，0意味着没有相关性。

3. 比传统的相关性更具普适性：它衡量两个变量X和Y之间的单调性程度（当Y增加时，X也增加吗？）而不是线性度（Y = a + b*X +噪声，其中a、b是使得噪声最小化的数值）。然而，它不应该和距离的相关性（当且仅当X和Y是独立时，等于0）或本章后面将会定义的结构性系数一样普通。

4. 对异常值不敏感，健壮性。

5. 更直观、更符合人类大脑感知相关性的方式。

请注意，拟合度是一个比较不同模型的模型效率的拟合优度检测指标，通常是观测值和预测值之间相关系数的平方，通过交叉验证技术在训练集上度量所得。与传统的相关性一样，它具有相同的缺点和益处。所以在这里，我将重点讲解相关性。

为了说明第一个条件（对n的依赖），考虑用下面的一对变量（或时间序列）制作出来的数据集：

X Y

0 0

1 1

2 0

3 1

4 0

5 1

6 0

7 1

在这里，n = 8，r（传统的相关性）= 0.22。如果n = 7（删除最后一个观测值），然后r = 0。如果n = 6，则r = 0.29。我观察到当n是偶数时，相关性很高。即使当n增大时，它们会慢慢地减少收敛到0。如果你将Y列向下移一个单元格（这里假设X和Y都是无限的时间序列），这时相关性变成了负数！然而，这组（X, Y）本来是用于模拟一个完美的0相关性的数据集的。对于较小的n而言，传统的相关性系数未能捕捉到这种模式（相关性不变）。

在大数据处理中有这样一个问题，你可能有很多的观测值（3亿个房屋的每月房屋价格和面积），但你以某个小数据桶里计算的相关性（例如，2013年某特定邮政地区出售的所有指定类型房屋）来为不在市场上出售的房子估计房屋价值，即使考虑到了本地波动性，这种做法也是不可取的。特别是，不同大小的数据桶之间的比较变得毫无意义。

我们提高标准相关性的出发点是Spearman的秩相关性。它是传统意义上的相关性，但是在排序后的变量上进行度量。下一节介绍的这一族相关性都是基于秩统计量的。基于排序变量，我们就可以满足条件#3和#4（使用新的相关性，#4将进一步改善）。现在用s表示Spearman的相关性系数。

容易证明：

$$s = 1 - S(X, Y)/q(n)$$

其中，

- $S(X, Y) = SUM\{ |x(j) - y(j)|^2 \}$
- x（j）、y（j）表示序列X和Y排序后的第j个观测值。
- $q(n) = n*(n^2 - 1)/6$

通过以上构造，s可以满足条件#2。一个重要且有趣的事实，也是一个值得关注的问题是，$q(n) = O(n^3)$。简而言之，q（n）增长过快。

# 一系列新的秩相关性

不失普遍性，从现在开始你可以假设X是有序的，因此x（j） = j。新的相关性将被表示为t或t（c），以强调它是一族受参数c影响的相关性。贝叶斯统计学家可能认为c是先验分布。

新相关性t计算过程如下。

步骤**A**

- 计算$u = SUM\{ |x(j) - y(j)|^c \}$。
- 计算$v = SUM\{ |x(j) - z(j)|^c \}$，其中z（j） = n–1–y（j）。
- T = T（c） = min（u, v）。
- 如果u<v，则Sign=+1；如果u>v，则Sign=–1；如果u=v，则

Sign=0。

步骤**B**

- t= t（c） = Sign * { 1–T（c）/q（n） }。
- 对于有n个观测值的所有（X, Y），q（n）是T（c）可以取得的最大值。

## 阐述与讨论

下面是一些关于这种新相关性的属性。

- 这种新的相关性仍然是对称的。（即使互换X和Y，t也不会改变。）此外，如果你对调了Y的顺序，则相关性只有正负号改变。（但s也满足这个属性，请把证明它作为练习。）
- 当c> 0时，它恒满足条件#2。此外，当c>0，当且仅当s =+1时，t=+1；当且仅当s=−1时，t=−1。
- 当c = 2时，t和s（Spearman的）相似，然而c = 2，t对异常值过于敏感。此外，c = 2是人为设置的值，可以使传统的计算和建模变容易。但上述参数设置没有考虑到实际应用（除非选择c=1）。总之，它是基于理论和历史等因素而建立的，类似于老式的拟合度，有不少缺点且常被批判。
- 我猜想q（n） = O（$n^{\{c+1\}}$ ）。当c很大时，它为T（c）的分布制造了一个人工长尾。应该选用较小的c；否则，罕见、非常高的T（c）值，将人为地提高q（n），并且使得所产生的t很难平衡。
- 这里我们将重点关注c = 1的情况。
- 计算X和Y（u表示）之间的秩的距离，同时计算X和Z（v表示）之间的秩的距离，其中Z是Y的倒序，然后选择u和v中较小的值。这一步也决定了相关性的正负号。
- 一般来说，当值接近–1或+1时，s和t是非常相似的，但当值接近0时——即，当你想测试是否相关性为0时，它们有着显著的差异。图4-2介绍了一个例子，显示了所有n！=364 880潜在的相关性组合。图4-3显示了一个例子，其中s和t是截然不相同的。

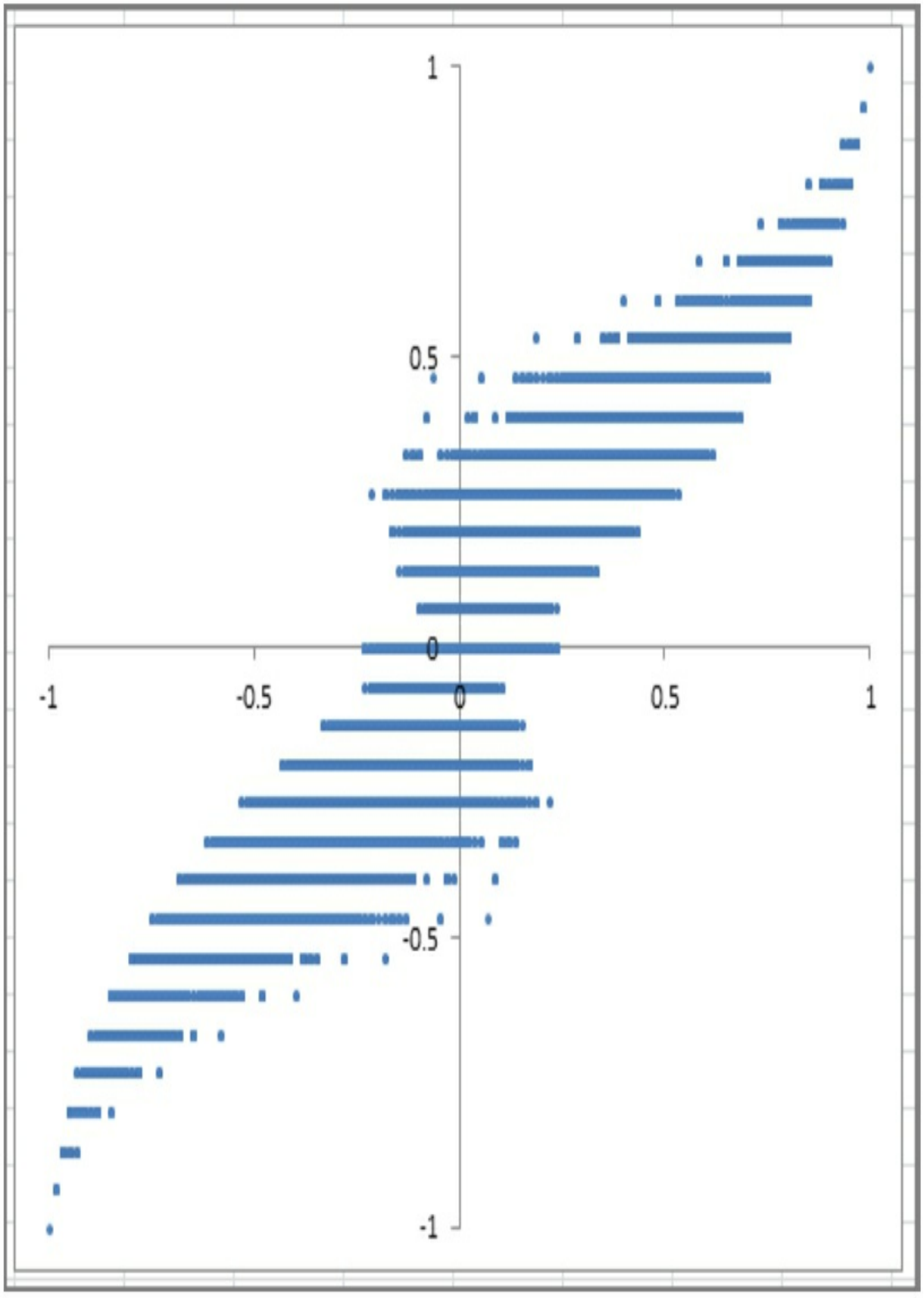
1
0.5
0
-0.5
-1
-1
-0.5
0
0.5
1

图4-2　当n= 9时，相关性向量（s, t）的分布

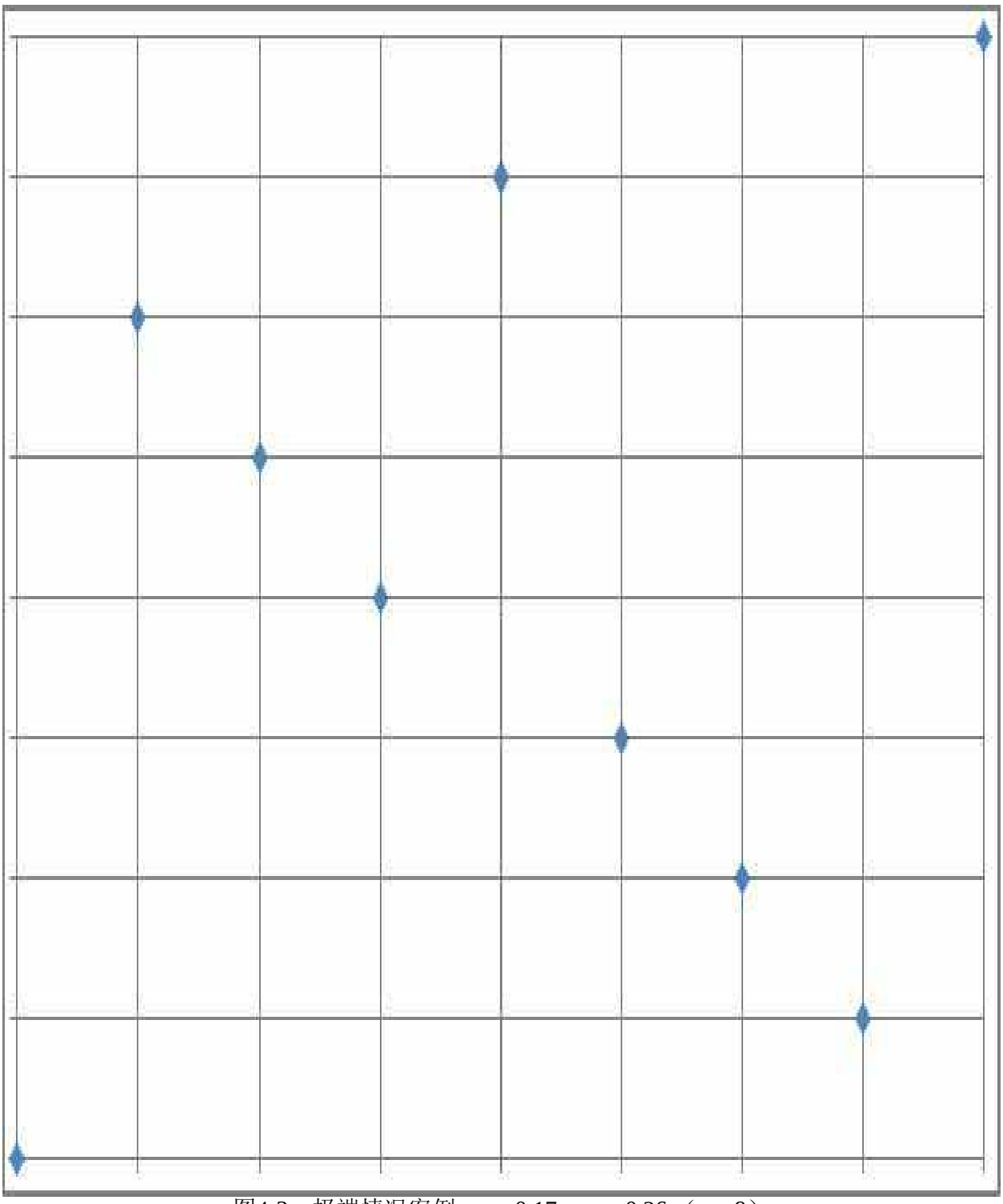

图4-3　极端情况案例，s = 0.17，t = –0.26 （n = 9）

注意　　问题：在图 4-3 中，哪个相关性最有意义，s还是t?

答案：哪个和零值都没有统计上的差异。然而，t更有意义，由于右上角和左下角的两个点看起来像异常值，它使s为正（但t不是，t对异常值不太敏感）。

# 渐近分布与归一化

再次，我假设使用c = 1来定义t。在模拟中，X = （0, 1... n–1），Y是n个元素的随机排列（由首先创建的n个随机数获得，然后提取它们的秩），结果表明。

- t收敛到0的速度比s要快：在一个实验中，n=10，对于30多次模拟，t的平均值是0.02，s的平均值是0.04。
- t的平均数也小于s的平均数。
- 你还没有计算p值，但它们是不相同的，其他的一切参数（n、c、测试级别、相关性是0的零假设）是相等的。

最后，当t和s完全不同时，通常t值看起来更自然，如图4-3所示；t<0但s>0，因为t能更好地消除两个异常值的影响（右上角和左下角）。当你计算几千个各种规模数据桶的相关性时，这是至关重要的：你必然会发现一些桶中存在异常值。如果你寻找极端的相关性，你的结论可能会有严重的偏差。（阅读第2章中“大数据‘诅咒’”一节，以了解更多的信息。）

图4-4显示了当n=10时，T的分布。当n=10时，97.2%的t的值都在−0.5和+0.5之间。当n=7时，92.0%的t的值都在−0.5和+0.5之间。因此，t的分布取决于n。在理论上，当c接近0时，会发生什么？

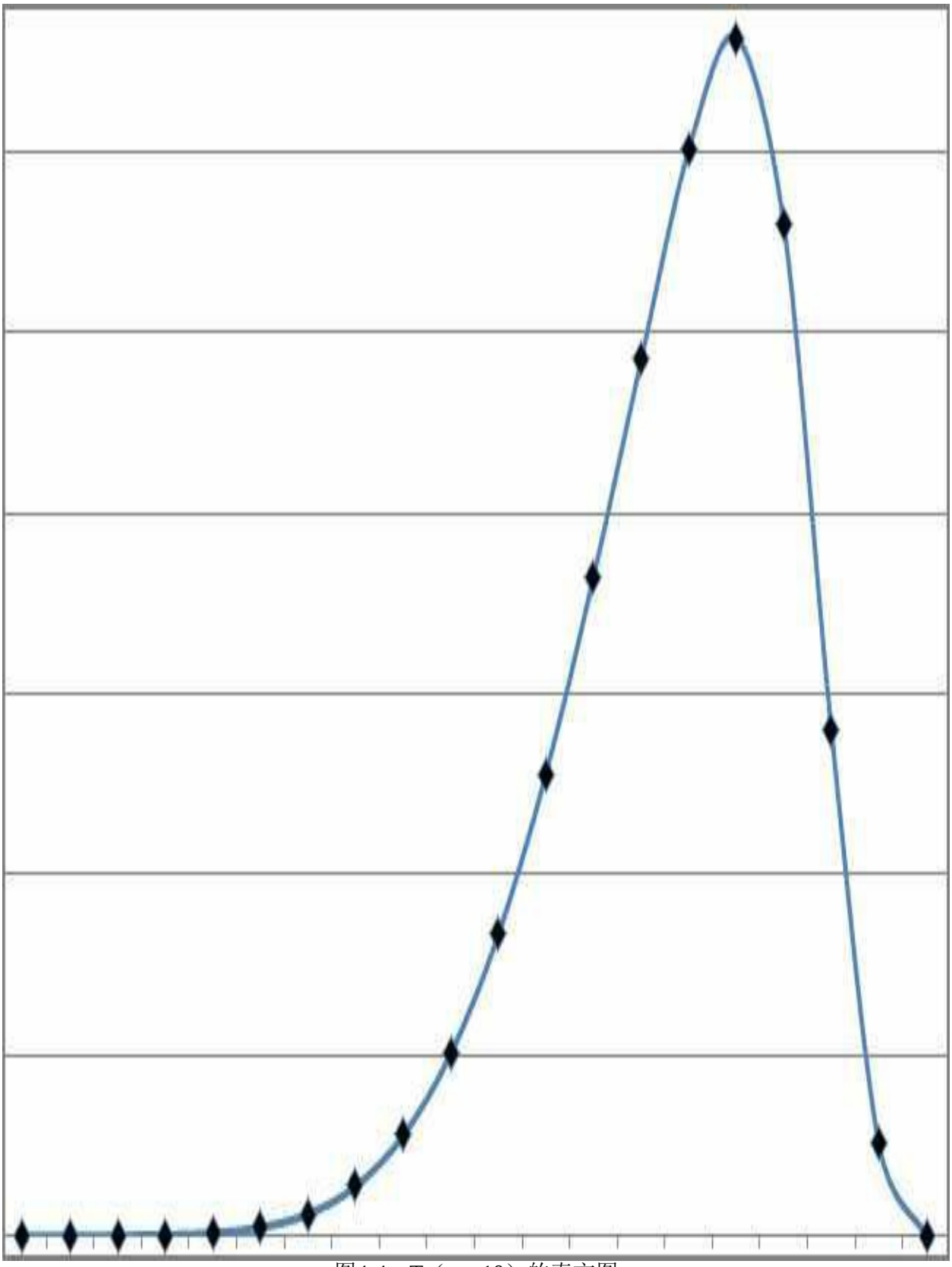

图4-4　T（n = 10）的直方图

注意　一个有趣的事实是，t和s大部分时间都是一致的。t和s之间的经典的相关性r，通过计算观测值n=9时所有n！= 362880个潜在的（X, Y）向量，等于0.96。

要估计t的渐近分布（当n变大时），你需要计算q（n）。有没有一个确切的公式来计算q（n）呢？我举行过一个比赛，看谁能够想出一个公式，或证明公式不存在，或至少找到一个渐进公式。阅读下一节找出细节。让我们现在就说吧！有两位获胜者——他们都是数学优化专家——获得了1 500美元的奖励。

## 归一化

因为相关性的值都在[−1,1]内，它是有界限的。例如，对于任何大小的n，我想相关性0.5总代表相同的百分位数t。有不同的方法可以实现这个性质，但它需要涉及转化t。我们仍需要做一些研究。对于r或s，通常可以使用Fisher变换。但它对t没有效果，再加上它把一个有边界的指标转化为一个无边界的指标，这使其很难解释。

## 排列

这里大多数的研究和仿真涉及生成很多关于n的排列。本节讨论如何生成这些排列。

一种生成所有排列的方法：你可以在`http://www.analyticbridge.com/group/codesnippets/forum/topics/scode-to-compute-all-permutations-of-n-elements`　找到源代码和操作指南。当n大于2时，它变得缓慢；然而，很容易使用MapReduce将任务分配到多个虚拟机并行处理。另一个策略是抽取足够多的随机排列，从而得到对t的分布的近似，以及q（n）。在n个元素的排列和1到n的整数之间，有一对一的映射。一个n元排列可以唯一地表示一个整数序列，反之亦然。另外查看一下维基百科有关排列的文章中“编号排列”一节，详细信息：`http://en.wikipedia.org/wiki/Permutation#Numbering_permutatio`。它是缓慢的：将一个数字转换为一个排列的复杂度是O（$n^2$　）。所以，我用了一个基本的方法（你可以在`http://api.ning.com/files/AMgcP5snh*Bu4RUyaEA*dZy4PbWgxQyxMNdEhQ/permut3simul.pl.txt` 下载源代码），仅仅是为了速度更快。

生成随机排列的算法如下：

（p（0）, p（1）, ..., p（n–1））

对k = 0到n–1，执行

步骤 1：生成p（k）作为一个随机数{0, ..., n–1}。

步骤 2：重复步骤1，直到p（k）和p（0）, p（1）, ..., p（k–1）各不相同。

## 最后的评论

下面的评论更深入地洞察了相关性t和s两者之间的差异。

- 如果你有兴趣使用相关性来评估你的预测值的准确性，可以把你的数据集分成两组进行交叉验证：测试组和控制组。在测试组数据（也称为训练集）中微调你的模型，使用t来测量其预测能力。但实际的预测能力是通过在控制组数据上计算t来测量的。`http://www.analyticbridge.com/profiles/blogs/use-press-not-r-squared-to-judge-predictive-power-of-regression` 上的文章“使用PRESS而不是拟合度来评估回归的预测能力”提出了留一法（leaving-one-out）策略，来达成一个类似的目标。
- 注意，t（不同于s），是一个双峰分布并在0附近有一个微降。这是有道理的，因为0相关性意味着没有模式被发现（属罕见事件），然而非0（但远离–1或+1）意味着有模式被发现（任何类型的模式，可能很微弱），是常发生的事件。事实上，我认为这种双峰分布是t超越s的一种优势。
- t可用于异常值检测：当|t−s|较大时，或|t−r|较大时，说明我的数据集有异常值了。
- 我的t是不多的几个由算法定义的而不是公式定义的指标。这是数据科学的一个新趋势，使用算法来定义概念。
- 事实上，s使用平方距离，而不是距离（如t），这使得它不够健壮，因为异常值会在s里占很大比重。虽然s和t两者都是基于秩统计量以消除异常值的影响的，但当n较大时，s仍然对异常值敏感。有了大数据，我们会遇到更多n值较大的案例。
- 你可以花上几年的时间来研究t和随机排列这一主题，并就此主题

写一本书或博士论文。在这里我做了最少量的研究，以深刻理解该问题和获得问题的解决方案。与进行缓慢的综合性学术研究相比，这种数据科学研究就是如上这种方式。

# 计算复杂度

本节有两个目的。首先，要解决前面章节讨论过的问题——即用适合处理大量数据的算法（模拟），找到一个明确的公式来计算q（n）。这可以帮助你熟悉很多与计算复杂度相关的计算机科学概念。其次，要展示一个精确的数学解决方案，证明有时数学建模可以击败最强大的集群计算机系统。虽然通常情况下，两者是携手工作的。

函数q（n）是为处理大数据提出的新型统计指标的核心：用以测量新类型的相关性或拟合优度的一个非参数的、健壮的指标。这个指标使已经用了几个世纪的传统指标更通用化，更适合于大型有序数据，如秩数据。虽然基于秩统计量，但它对异常值的敏感性要远小于当前基于秩统计量（Spearman的秩相关性）的指标，尽管当前这些指标被设计为适用于比较小的n，且像Spearman秩相关性本身已经有较好的健壮性了。

# 计算**q（n）**

从一个原始方法开始，然后经过几轮的改进。

## 暴力算法

暴力算法是通过访问n个元素的所有n！排列，来寻找q（n）的最大值。因此，计算的复杂度是O（n!）——这非常缓慢。

这里的代码，迭代地生成存储在数组中的n个元素的所有n！（n的阶乘）排列。它可以让你很容易地计算与这些排列相关的统计数字和计算聚焦在所有或特定的排列。对于n>10，它可能很缓慢，一些MapReduce架构会对此有帮助。如果n>15，你可能对抽样感兴趣，而不是访问所有排列来计算摘要统计量，后文有更多信息。因为q（n）是计

算n个数的全排列中的最大值，所以上述采样方法可以帮助你的计算。

这个代码对于计算关键词的n-grams模型也有用，特别在非参数统计的情况下，可能更有价值。在Stackoverflow.com（`http://stackoverflow.com/questions/2390954/how-would-you-calculate-all-possible-permutations-of-0-through-n-iteratively` ）中，你可以找到各种现代语言的源代码，包括Python。用谷歌搜索**calculate all permutations** （计算所有的排列）可以得到很多有趣的链接。下面是Perl的代码。

```
use List::Permutor;
$n=5;
@array=();
for ($k=0; $k<$n; $k++) { $array[$k]=$k; }
  $permutIter=0;
  my $permutor = new List::Permutor(@array);
  while ( my @permutation = $permutor->next() ) {
   for ($k=0; $k<$n; $k++) { print "$permutation[$k] "; }
  print "\n";
  $permutIter++;
}
```

它再容易不过了。你需要Perl的库Permutor。（在其他语言中不需要库，因为提供了完整的代码。）以下是安装库最简单的方法，这个方法适用于任何Perl库。

不要相信那些声称可以用ppm或CPAN自动一键下载并安装库的人。这是迄今为止最困难的方法。相反，使用原始安装（据说是最难的，但其实是最简单的解决方案），操作如下。

1. 用谷歌搜索**Permutor.pm** 。

2. 在Permutor CPAN页（谷歌搜索结果页面的第一个链接），单击SOURCE（在页面的顶部）。这个.pm文件是一个文本文件（在这里pm是文件扩展名，如.doc、.txt、.xls）。

3. 复制源代码，粘贴到记事本上，并命名为**Permutor.pm** 保存在`Perl64/lib/List/` 文件夹下（或你的计算机上的任何其他文件夹名）。

这就是它！注意，如果你没有在你的系统上安装List库（Permutor

的母库），你必须首先按相同过程下载并安装List库，但现在每个人的计算机上都有List库了，因为它是Perl包默认设置的一部分。

第一次改进：抽样排列以估计q（n）。而不是访问所有的排列，通过访问精心挑选的小样本，可以节省大量时间。如先前提到的，该算法生成随机排列（p（0）, p（1）, ..., p（n–1））：

```
For k=0 to n-1, do:
Step 1. Generate p(k) as a random number on {0, ... , n-1}
Step 2. Repeat Step 1 until p(k) is different
from p(0), p(1), ... , p(k-1)
End
```

你还应该考虑该算法的计算复杂度，因为它要与两个备选算法进行比较。

- 备选算法**I**： 在[0,1]中创建n个随机数字，然后p（k）是第k个偏离量的秩（k = 0,..., n−1）。从计算复杂度的角度来看，这相当于排序算法，因此它的复杂度是O（nlogn）。
- 备选算法**II**： 针对n的排列和1~n的数字建立一对一映射。然而，将一个数字转换成一个排列，使用阶乘数表示，计算复杂度是O（$n^2$）。

你可能会认为最基本的算法是极其慢的。事实上，它可能永远不会结束，将无限期地运行，甚至仅仅只是生成p（1），更不用说p（n-1）了，这是相当难生成的。然而，事实证明基本算法的复杂度也是O（nlogn）。（证明这一事实，实际上是一个很好的工作面试问题。）下面是推导概要。

对于正数k严格小于n，你可以用如下方法来创建p（k）：

- 第1步，产生概率L（k,1） = 1 – k/n
- 第2步，产生概率L（k,2） = （k/n） * （1 – k/n）
- 第3步，产生概率L（k,3） = （k/n）$^2$ * （1 – k/n）
- 第4步，产生概率L（k,4） = （k/n）$^3$ * （1 – k/n）
- 等等

请注意，所需的步骤次数可能是无限的（概率为0）。因此，平均而言，你需要M（k） = SUM{ j * L（k, j） } = n/（n–k）次来创建p（k），其中，求和（sum）是针对正整数j = 0, 1, 2...的。因此，总的操作数（平均）是SUM{M（k）} = n * {1 +1/2 + 1/3 + ...+ 1/n}，其中求和（sum）是对k = 0,1,..., n–1的。这样计算复杂度就是O（nlogn）。请注意，在数据结构方面，你需要一个大小为n的辅助数组A，初始化为0。当创建p（k）（在最后一次，成功达到迭代k）时，更新A：A[p（k）] = 1。这个数组是用来核查p（k）是否不同于p（0）, p（1）, ..., p（k–1）。或换句话说，是否A[p（k）] = 0。

O（nlogn）是你能获得的最好的算法吗？不是的，下一节的算法更好，计算复杂度为O（n）。

### 第二次改进：改进排列取样器

Fisher–Yates洗牌算法是实现排列的最佳算法。其工作原理如下：

```
p(k)←k, k=0,..., n-1
For k=n-1 down to 1, do:
j←random integer with 0≤j≤k
exchange p(j) and p(k)
End
```

在计算复杂度方面，这是随机排列生成器能实现的最好算法（即，它是最快的算法）。

# 理论上的解决方案

有时，处理海量数据（十亿或万亿的随机排列）不是最好的办法。使用一点点数学知识可以简单而优雅地解决你的问题。

2013年，Jean-Francois Puget博士，一名杰出的工程师，在IBM从事行业解决方案的分析和优化，找到了准确的公式，并想出了该如何证明该公式。因为该解决方案，他获得了由Data Science Central颁发的1 000美元奖金，他是第一届理论数据科学竞赛的获胜者。如果你对应用数据科学竞赛感兴趣（涉及真实数据集），你可以访问

http://www.Kaggle.com 。

下面仔细思考Puget博士的结果。

- 假设m是商，r是n除以4的余数：$n = 4m + r$，$0 \le r < 4$。
- 假设$p(n) = 6m^2 + 3mr + r(r-1)/2$。
- 然后：

- 如果$p(n)$是偶数，则$q(n) = p(n)$。
- 如果$p(n)$是奇数，则$q(n) = p(n) - 1$。

你可以在www.datashaping.com/Puget-Proof.pdf 中找到关于此的更长、更复杂的证明。

# 结构系数

本节讨论测量一个数据集是否具有某种结构的指标。该指标可以测量数据集中的结构或模式存在或不存在。目的是测量两变量之间的关联强度，从如下几个方面拓宽在“大数据的相关性和拟合度”一节讨论过的现代相关性系数。

- 它适用于非数值数据，例如，关键词对的列表，每对都附上编号，测量两个关键词之间的紧密程度。
- 它检测的并非函数性的关系（例如，点分布在一个不寻常的域，如一个有孔的球体，并且孔包含较小的球体，也是域的一部分）。
- 可以用于传统的、数值的、二元的观测值。

结构系数表示为w。我在如下的框架中操作。

- 我有一个包含n个点的数据集。为了简单起见，现在考虑这n个点为n个向量$(x, y)$，其中x、y是实数。

- 对于每对点{（x, y），（x', y'）}，计算两点之间的距离d。在更一般的情况下，它可以测量两个关键词之间距离的近似量。
- 我整理所有的距离d，且根据这n个点，计算距离的分布。
- 留一法（Leaving-one-out）每次删除一个点，并基于n−1个点，计算n个新的距离分布。
- 然后比较用n个点计算的n个分布和用n−1个点计算的n个分布。
- 重复该迭代，但这次是有n−2个点，然后是n−3个点，n−4个点，等等。
- 假设不存在模式，这些距离分布（对于连续的值n）将有某种行为，不存在唯一的特征结构。通过模拟，你可以识别行为。有任何偏离该行为的情况，都会表明有结构存在。无模式行为将独立于基本点的分布或域，这是一个重要的点。当然，所有这些点都必须被确定或测试。
- 测试该指标是否可以识别模式会很有趣，如分形分布、分形维数。

这种类型的结构系数，没有假设基础域的形状，其中n个点位于这里。这些领域可以是平滑的或颠簸的，由线组成或由双点组成，有孔等。这种类型的结构系数可以应用于非数值数据（例如，数据包括关键词）、时间序列、空间数据，或在更高维度的数据。它比目前用来测量结构的指标更为通用，而且完全是由数据驱动的。

# 确定簇的数量

聚类算法在前几章及本章中已经讨论过很多了。在这里，你可以找到一个经验法则，可以自动地、不需要由人们视觉检查地确定与任何聚类算法相关的簇的最佳数量。

注意，簇的概念是一个模糊的概念。你如何定义簇？然而，在许多应用中，都有一个明确的簇的最佳数量。这里所描述的方法将解决所有容易和较容易的情况，且为模棱两可案例提供一个所谓“不知道”的答案。

# 方法

所描述的方法，有以下3个步骤。

1. 创建一个包含以下行的二维表：簇的编号在#1行，簇对方差的解释比率在#2行。

2. 计算第3个差异。

3. 计算第3个差异（如果远高于其他值）的最大值，以确定簇的数量。

基于以下事实，簇的个数 和由簇解释的方差的百分比 之间的分段线性图，是有肘点 （*elbow point* ）的凸函数（见图4-5）。肘点确定簇的最佳数目。如果分段线性函数近似于平滑曲线，最佳的是这个近似平滑曲线的二阶导数为零的点。这一点对应于所考虑曲线上的拐点。对一个离散的框架（簇的数目是离散数字）来说，这种方法是“肘点检测”技术的简单应用。

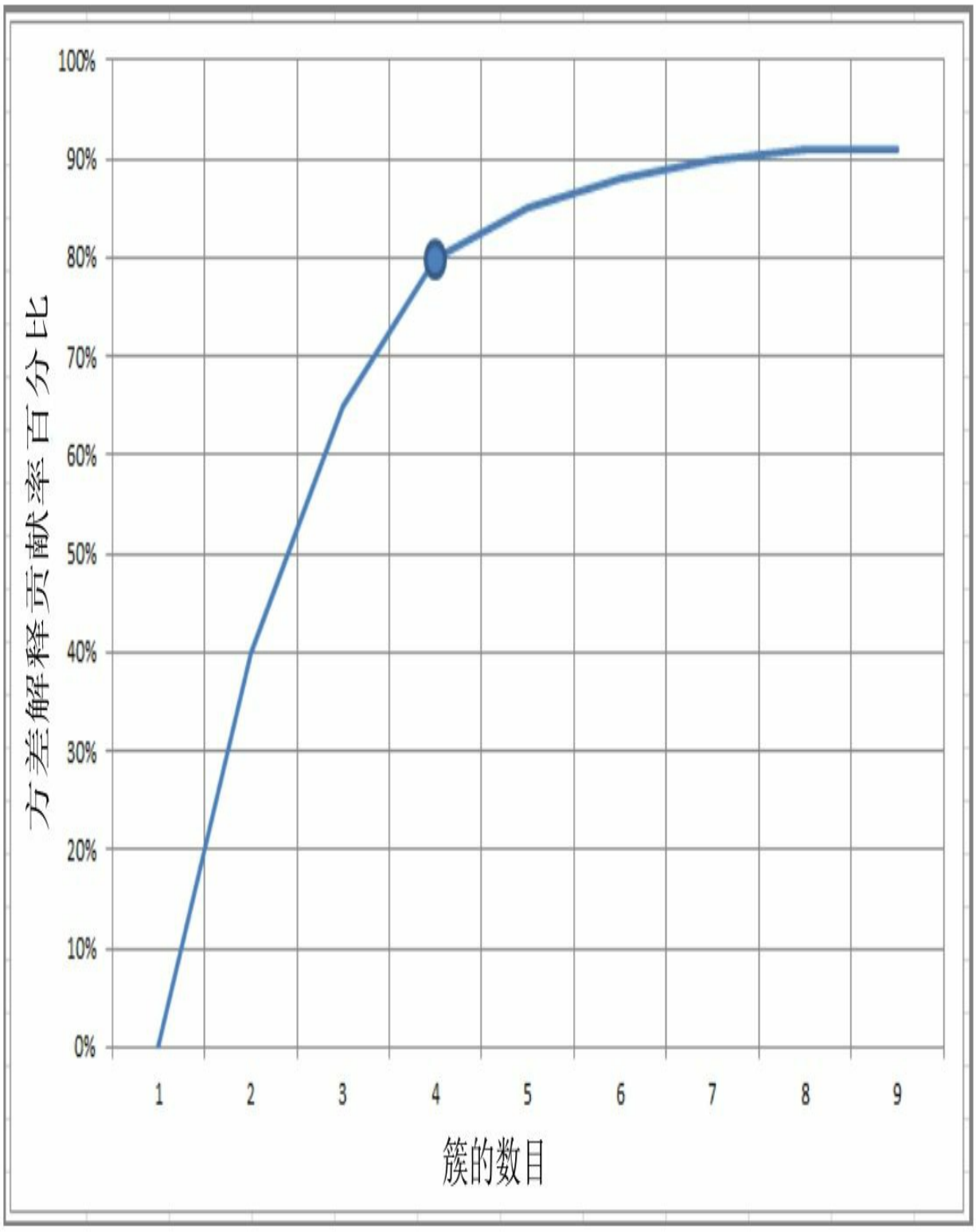


图4-5　肘点（标记为圆圈的）用来检测簇的最佳数目

# 例子

在图4-5中，行#1代表簇的数目（X轴），行#2代表由簇解释的方差的百分比（Y轴）。然后，第3、第4和第5行是和上面每行的差异（有时称为差量 （*deltas* ））。例如，在#3行的第2个数字是15 = 80−65。

- 1　2　3　4　5　6　7　8　9　==>簇的数目（X轴）
- 40　65　80　85　88　90　91　91　==>由簇解释的方差的百分比（Y轴）
- 25　15　5　3　2　1　0　==>在Y轴的第1个差异
- −10　−10　−2　−1　−1　−1　==>在Y轴的第2个差异
- 0　8　1　0　0　==>在Y轴的第3个差异

在这个例子中，簇的最佳数目为4，对应于第3个差异中的最大值=8。

注意　　如果你在第2个差异中已经有了一个很强的最小值（该例子没有这种情况），你不需要继续第3个差异：停止在第2层即可。

# 网络拓扑映射

本节讨论一个往往缺少却对很多系统有价值的组件：算法和架构用于处理在线或移动数据，如交易评分、欺诈检测、在线营销、营销组合和广告优化、在线搜索、剽窃和垃圾邮件检测。

我将称它为网络拓扑映射。它可能无法作为传统的数据库存储。（它可能是一个图形数据库、一个文件系统，或一组查找表。）它必须是预生成的（例如，作为查找表并定期进行更新），以有效地使用。

从本质上讲，网络拓扑映射是一种系统，它用于匹配IP地址（互联网或移动网络）与域名（ISP）。当你在产品模式中处理实时交易时（例如，在线信用卡交易，决定接受或拒绝），你的系统只有几毫秒的时间来评分交易并做出决定。总之，你只有几毫秒的时间去调用和运行

一个算法（子进程），分别为每个信用卡交易决定接受或拒绝。如果算法涉及匹配IP地址和ISP域名（这个过程叫作网络管理的工具向导（*nslookup* ）），那就没办法运行了：直接的nslookup过程需要花费几百到几千毫秒，它们会大大地拖慢系统。

正因为如此，大部分系统都缺少网络拓扑映射。然而，有一个简单的方法来建立它。

1. 在数据库中查看所有IP地址。会发现这样一种情况，即使是谷歌，99.9%的流量是由不到1亿个IP地址产生的。事实上，全部IP地址的数目（整个宇宙）少于$256^4$ = 4 294 967 296个。大约40亿，在大数据中这不是一个很大的数字。此外，许多IP地址是成组的：很多IP地址开始（比如）是120.176.231，可能是同一个域的一部分。总之，你需要存储查找表，可能只是2 000万个记录（IP范围/域映射），以解决你交易的99.9%的nslookup问题。对于剩下的0.1%，你可以指定“未知领域”（不推荐，因为实际上相当多的IP地址都有一个未知的领域），或指定“丢失”（更好），或运行原始缓慢的nslookup过程。

2. 创建一个查找表，对于99.9%的流量，该表映射IP地址到域名。

当处理交易时，访问在最后一个步骤中创建的查找表（存储在内存中，或在内存中至少有一些高速缓存可用）来检测该域名。现在你可以使用一个包含域名的规则系统。

基于域名的规则和指标的例子如下。

- 域名扩展（.com、.cc等）。
- 域名长度。
- 域名标记为好的（白名单）或坏的（黑名单）。
- 域名中发现的模式（例如，数字、日期、标记的关键词和特殊字符，如破折号）。
- 域名中发现的特定关键词。
- 域名的所有者（所有者是否还拥有其他的域名？特别是坏的域名）。
- 域名的创建日期。
- 对所考虑的IP地址运行nslookup所需的时间（0.5秒或4秒？）。
- 当创建IP地址或域名表时，需要多个nslookup来找到属于该IP地址

的域名吗？

- 没有发现域名。
- 这是一个子域名吗？

这是网络拓扑映射的第一个组成部分 。第二个组成部分 是一个聚类结构。简而言之，一个结构（文本文件也是），其中聚类被分配给每个IP地址或IP范围。聚类的例子，包括：

- IP地址对应于一个匿名代理。
- 教育代理。
- 公司代理（大型的有许多IP地址或小型的有少量但常用的IP地址，用一个前5万家公司或网站的列表来帮助检测公司域名）。
- 政府代理。
- 静态、个人IP。
- 不稳定、可回收的IP（例如，AOL、Comcast和移动IP地址）。
- 隔离的IP地址与属于IP地址的一个簇或多个簇（无论小的或大的）的IP地址。
- 知名的ISP（互联网服务供应商）：Comcast等。
- 移动IP地址。
- IP地址是AWS（亚马逊网页服务和云服务）或类似的服务。
- IP地址是一个邮件/网页/FTP或API服务器。

有了这些组件（IP地址/域映射+IP地址簇结构，又名网络拓扑结构），你现在可以开发更好的算法：实时、后台（back-end）或一天即结束（end-of-day）的算法。你需要IP地址、域映射来创建簇结构。如果你雇用了一个数据科学家，他应该能很容易地创建网络拓扑结构映射和识别使用它的巨大好处。

创建该产品唯一的问题（假设它包含2 000千万个IP地址范围且定期更新）是花费大量时间去做数百万缓慢原始的nslookup（每个需要0.5秒）。现在，有名的域名范围都预留给美国在线（AOL）和其他大的互联网服务供应商，所以你可能只需做1 000万次nslookup。鉴于1%的nslookup过程会失败（2秒后超时），你必须在某些IP地址运行两次nslookup，所以简而言之，你要运行1 000万次nslookup，每次平均1秒。这大约需要2 777个小时，或115天。

通过利用分布式架构，你可以使用MapReduce环境，很容易地减少

20倍所用时间。如果你同时运行25个nslookup脚本，即使只有一台机器，你也应该能让它运行快4倍——即，它会在不到1个月的时间内完成。这就是为什么我说，一个小家伙单独在他的车库里，几个星期或更少时间内，就可以创建网络拓扑映射。输入的数据集（例如，1亿个IP地址）将需要不到20GB的存储空间，当压缩时，甚至更少，相当小。

最后，这是一个Perl脚本，在IP地址的输入文件`ips.txt` 中自动执行nslookup且结果存储在`outip.txt` 中。它在我的Windows笔记本电脑上就可以运行。你需要一个互联网连接，使它能运行，同时你还应该添加一个错误的管理系统，如果你的电脑没电或互联网连接突然中断，也可以很好地恢复。

```
open(IN,"<ips.txt");
open (OUT,">outip.txt");
while ($lu=<IN>) {
  $ip=$lu;
  $n++;
  $ip=~s/\n//g;
  if ($ip eq "") { $ip="na"; }
  `nslookup $ip | grep Name > titi.txt`;
  open(TMP,"<titi.txt");
  $domainName="n/a";
  while ($i=<TMP>) {
    $i=~s/\n//g;
    $i=~s/Name\://g;
    $domainName=$i;
  }
  close(TMP);
  print OUT "$ip\t$domainName\n";
  print "$n> $ip | $domainName\n";
}
close(OUT);
close(IN);
```

总之，采用这里所描述的步骤就可以创建一个网络拓扑映射。它是一个有用的工具，可以帮助解决若干商业问题，包括垃圾邮件检测、欺诈检测、归因建模、数字营销（包括电子邮件营销）、出版（基于IP地址分类，自定义内容）和广告（优化相关性）。传统算法使用IP地址或IP范围的黑名单或白名单（网络拓扑映射的一个组成部分），但据我所知，没有使用簇算法生成IP簇的。我见过在某些特定指标（流量、欺诈金额等）下显示出的前100名IP地址，但这么做是非常糟糕的实践，因为IP地址非常分散，（所以前100名不能表达出很多内在的东西），且

一些IP地址由大量用户共享（尤其是移动IP地址代理）。而根据来自网络拓扑映射得出的IP分类显示，对于决策者将会更有用。

我也见到过，在决策树中使用IP地址作为指标，这也是很糟糕的。IP分类或IP簇（基于网络拓扑映射的）应该在决策树中使用，而不是用高度分散的IP地址做指标。举一个例子，可将网络拓扑映射用于制定潜在的欺诈检测规则，它包括“企业的IP地址（除非被黑客入侵）应该在白名单里”和“与很多学生电脑连接（共享代理）的大学IP地址，是高风险的（除了他们其中一些系统管理员，他们做了优秀的工作，让这些网络更安全）。”难点不是想出这两条直观的规则，而是将一个IP地址分类为企业的代理或大学的代理——网络拓扑映射可以帮你解决该问题。

# 安全通信：数据加密

本节很重要，因为它阐述了如何利用JavaScript代码生成一个实用的应用程序，如何创建一个任何用户通过他们的浏览器都可以使用而无须安装R的工具，以及如何下载和购买软件或编写代码。这里提供的JavaScript代码可以在浏览器中运行，无论是在线或离线状态。我相信在未来，将有更多基于浏览器的数据科学工具，且比API更轻便（我更倾向于让程序在你的浏览器离线运行）。该代码还提供了另一个生成随机数的例子。

与数据相关的一个大问题是如何安全地传输大数据。这里，有一些简单的JavaScript代码加密数据，如信用卡号码、由数字组成的密码、电话号码、社会安全号码、日期（如20131014）等。

下面是一个应用程序例子，任何人都可以以文本的格式下载并保存为HTML文件，然后在本地电脑的浏览器中运行（不需要网络连接）。因为与移动 、网页 或社交 （Facebook）应用程序完全不同，所以我把它称为离线应用程序 。与大多数应用程序一样，你不需要任何编程技巧来使用此程序。它最初的意义是在不需要互联网连接的情况下仍然正常运作。

它的运作原理如下。

1. 在一个单独的浏览器标签页中打开Web应用程序（`datashaping.com/encode.html`）。

2. 在网页上显示的框中输入数字，进行编码或解码。

3. 选择加密/解密。

4. 用E-mail发送加密的数字（它应该以e开头）给你的联系人。

5. 你的联系人使用相同的方式，输入加密的数字，选择加密/解密，然后原来的数字立即恢复。

此代码简单（不是强加密），没有uuencode那么复杂。但使用uuencode要精通高深加密知识，而我们的应用程序对于任何非技术人员都很容易使用。该编码的值也是文本字符串，因此在任何电子邮件客户端都可以复制和粘贴。编码值具有一定的随机性，因此两次编码相同的值会导致两个不同的编码值。最后，如果你不告诉任何人（除了电话以外）解码器在哪里，这种方法其实比你想象得更安全。

创建一个带有参数的版本，可以使它更加安全。这是JavaScript/HTML代码（这是源代码的网页，我们的应用程序托管于此）。你可以将它保存到本地计算机的文件夹中（例如，`C://Webpages`），用文件名（例如，`encode.html`）保存为一个HTML文件，然后打开并在你的本地机器上的浏览器中运行。如果你使用的是Chrome浏览器，则在你的本地网页上的URL将显示为`\\/C:/Webpages/encode.html`。

```
<html>
<script language="Javascript">
<!--
function encrypt2() {
  var form=document.forms[0]
  if (form.encrypt.checked) {
    form.cardnumber.value=crypt(form.cardnumber.value)
  } else {
    form.cardnumber.value=decrypt(form.cardnumber.value)
  }
}
function crypt(string) {
  var len=string.length
  var intCarlu
```

```
  var carlu
  var newString="e"
  if ((string.charCodeAt(i)!=101)&&(len>0)) {
    for (var i=0; i<len; i++) {
      intCarlu=string.charCodeAt(i)
      rnd=Math.floor(Math.random()*7)
      newIntCarlu=30+10*rnd+intCarlu+i-48
      if (newIntCarlu<48) { newIntCarlu+=50 }
      if (newIntCarlu>=58 && newIntCarlu<=64) { newIntCarlu+=10 }
      if (newIntCarlu>=90 && newIntCarlu<=96) { newIntCarlu+=10 }
      carlu=String.fromCharCode(newIntCarlu)
      newString=newString.concat(carlu)
    }
    return newString
  } else {
    return string
  }
}
function decrypt(string) {
  var len=string.length
  var intCarlu
  var carlu
  var newString=""
  if (string.charCodeAt(i)==101) {
    for (var i=1; i<len; i++) {
      intCarlu=string.charCodeAt(i)
      carlu=String.fromCharCode(48+(intCarlu-i+1)%10)
      newString=newString.concat(carlu)
    }
    return newString
  } else {
    return string
  }
}
// -->
</script>

<form>
Enter Number <input type=text name=cardnumber size=19><p>
Encrypt / Decrypt <input type=checkbox name=encrypt onClick="encr
</form>
</html>
```

# 总结

本章讨论了许多独创的新技术和方法，数据科学家可将这些用于处理现代数据，包括大数据。通过深入细节的介绍，人们可以在需要时复制应用它们，而不需要涉及太多技术细节、术语和数学公式，这样商务人员就能快速提取其精华。

第5章讨论其他技术，也是核心数据科学的一部分。虽然第4章和第5章包含本书中最具技术性的资料，但需要通过学习第2章引入的一些技术资料（见“大规模数据集的聚类和分类”一节），使过渡尽可能顺利。

# 第5章
# 数据科学的技术（II）

第4章讲解了数据科学的一些技术和诀窍，比如用数据视频进行数据可视化、新的指标类型、与计算机科学有关的话题、在挑选供应商时该问的问题，以及数据科学家、统计学家与数据工程师的异同。

本章内容较少关注指标，而更多关注具体应用。这包括如何创建数据字典、隐性决策树、NoSQL数据库里的哈希连接，以及分析桥第一定理，该定理提供了一种无须统计理论知识就可以用简单的、不依赖模型的、非参数的方法，用来计算置信区间。

相比第4章，本章不太侧重于统计理论。本章讨论的主题，可以看成数据分析，而不是统计分析或计算机分析。本章大部分材料之前尚未公开发表过。第6章则会讨论案例、应用及成功的故事。

本章讨论的隐性决策树、数据字典和哈希连接等，对于数据科学家来说是很重要的主题，因为这些是统计学与计算机科学的交叉技术，能够用于处理大数据。传统的统计学家通常不使用或不涉猎这些技术，但数据科学家恰恰需要掌握这些技能。

## 数据字典

进行探索分析时，一个最有价值的工具是数据字典。建立数据字典带来如下优势。

- 确定高维数据集里稀疏和密集的部分。

- 确定异常点及数据小差错。
- 对数据包含哪些内容有更好的了解，知道后续数据挖掘时哪里值得（或不值得）花时间。

# 什么是数据字典

数据字典 是一个有3列或4列的表。第1列是标签，指一个变量名，或者多个变量（最多3个变量）组合对应的名称。第2列是跟第1列标签对应的值：第1列和第2列实际上构成了名-值对。第3列是频率统计：衡量该值（跟第1列标签对应）在数据集里出现的次数。你还可以添加第4列，注明该标签对应的维度（1代表一个变量，2代表两个变量，以此类推）。

数据字典通常包含所有维度是1和2并且频率大于某个阈值（比如阈值=5）的标签。通常不包含维度是3的标签，除非它们有比较高的频率。应该先建立起维度是1和2的标签集，然后仔细探究维度是2的高频名-值对，从而得到维度是3的标签。

这里有个数据字典记录例子：类别~关键词 旅游~东京756 2。这个例子的标签维度是2（如第4列数字2所示），756则表示数据集中（旅游、东京）同时组合出现了756次。

对于数据字典，首先要做的就是按照这样的三维索引进行排序：先按第4列排序，然后是第1列，接着是第3列。随后就可以分析数据，并从中寻找模式。

# 建立数据字典

按顺序查看数据集。将所有1维和2维的名-值对作为哈希表的键保存下来，并将这些标签/值对应的频率加1。用Perl编程的话，这个操作可以通过执行代码如`$hash{“$label\ t$value”}++` 来实现。

如果该哈希表变得很大。停止运行该程序，将哈希表保存到文件

中，在内存中清除该哈希表变量。然后创建一个新哈希表从上次停止的地方继续。最后，忽略那些频率太小的记录，将哈希表合并。

交叉参考　　建立数据字典后，下一步的工作是探索式数据分析（EDA）。请参见本章“数据科学项目的生命周期”一节。

# 隐性决策树

隐性决策树 （*HDT, Hidden Decision Tree* ）是我发明并申请专利的技术（`http://www.freshpatents.com/Method-and-system-for-scoring-quality-of-traffic-to-network-sites-dt20070816ptan20070192190.php`），用来给海量交易数据打分。隐性决策树混合了健壮逻辑回归和成百上千个小决策树（例如，每个决策树代表一类欺诈交易），但隐性决策树比逻辑回归和决策树有明显优势，包括健壮性、易解释、无须剪枝算法和无须节点分裂准则。这些优势使得该方法强大、容易实现。哪怕是没有统计背景的用户，都很容易上手。

HDT属于统计与数据挖掘方法（就像逻辑回归、支持向量机、神经网络或者决策树），可用于处理数据量很大、非线性、变量强相关的问题。我发现大数据通常很稀疏（也很多噪声），有价值的信息（信号）集中在少量数据集上，因此发明了HDT方法。这种现象就好比星星和物质只占宇宙非常小的比重。HDT相比传统的大规模决策树，减少了复杂度、不稳定性、参数随意性及可解读性不足的问题。HDT通过混合一些决策树回归方法，解决了传统逻辑回归难以应付的高度非线性结构识别的问题。HDT率先用于基于信用卡记录打分的欺诈检测中的最优特征选取问题。这门技术用任何程序语言都容易编写实现。它比决策树和逻辑回归更健壮，能帮助探测真正的终端节点。它通常严重地依赖庞大、小颗粒度的哈希表来实现。

实际上我们并没有真的构建一个决策树（隐性决策树这一方法因此而得名），隐性决策树的最终输出来自于不交叠的小决策树构成的几百个节点。而这些（隐藏的）父决策树，举例来说，对应欺诈检测模型里的某个特定类别的欺诈行为。相比传统决策树，解读因此变得直观。

该方法首先是2003年在信用卡欺诈检测的应用中发明的。目前任何

统计包都还没实现该方法。隐性决策树通常和逻辑回归合成混合的打分算法，80%的交易通过隐性决策树打分，余下20%则使用兼容的逻辑回归式打分方法打分。

隐性决策树得益于给大量交易打分（比如用于欺诈检测）时常能观察到的大量多变量特征结构。这个技术跟隐性马尔可夫场没有联系。

隐性决策树的潜在应用领域如下。

- 欺诈及垃圾信息检测。
- 网络分析。

- 关键词打分/竞价（广告网络及付费搜索）。
- 交易打分（点击、曝光、转化及行动）。
- 点击欺诈检测。
- 网站评分、广告评分、载入页或者广告打分。
- 协同过滤（社交网络分析）。
- 相关性算法。

- 文本挖掘。

- 打分及排序算法。
- 侵权及垃圾信息检测。
- 情感分析。

将隐性决策树运用于上述问题的具体方法如下：在应用某一典型的机器学习算法时，用规则集方法或者关联规则　方法来解决这些问题。举个例子，如果一封邮件包含“乳腺”或者其他属于关键词黑名单的词，但没有白名单里的“乳腺癌”这样的关键词，那么垃圾信息可能性分数会升高。如果触发了许多规则，那么成为垃圾信息的可能性分数会非常高。隐性决策树给不同的规则自动赋予权重，并且从数百万（甚至数十亿）规则中自动侦测出哪些规则需要保留或合并。

自从我发明隐性决策树后，我用了10年时间进一步改良这个模型，如下。

- 2003：第一个版本应用到信用卡欺诈检测。
- 2006：应用到点击打分及点击欺诈检测。
- 2008：发展出更高级的版本，可以处理小颗粒度并且大规模的数据集，如下。

- 隐性决策森林　——包含多个隐性决策树，每个都运用于一簇关联规则。
- 分层隐性决策树　——用隐性决策树对顶层结构，而不仅是对规则集进行建模。
- 非二值规则 ——朴素贝叶斯跟隐性决策树混合。

# 实现方法

此处陈述的模型以在点击打分中的应用为例。任务目标是给出预测分数，而分数 =f（响应 ），也就是说，分数是响应的函数。响应值有时是指统计和预测模型里的因变量。 响应值的例子包括：

- 转化率（网络流量数据——硬转化率或软转化率）。
- CR（转化率）。
- 属于欺诈交易的概率。

独立变量被称为特征 或规则。它们高度相关。

跟隐性决策树做比较的传统方法包括逻辑回归、决策树和朴素贝叶斯方法。

隐性决策树在分数和多元特征之间建立起一对一映射。多元特征（*multivariate feature* ）是跟某个特定交易绑定的规则组合（它是一个向量，标明哪个规则被触发，哪个没有），有时这被称为标记向量　或

者节点 。

针对通常的数据集，隐性决策树的基本思路如下。

- 如果你使用40个二元规则，那就有2的40次方个多元特征。
- 如果训练集有10MM条交易，显然最多观测到10MM个多元特征，这个数字远小于2的40次方。
- 10MM特征中有500个特征涵盖了80%的交易。
- 故此排在前500的多元特征有很强的预测能力。
- 同时需要另外的算法来对余下的20%的交易进行分类。
- 用相近的重要的多元特征来给余下的20%的交易打分会导致偏差（Bias），因为稀有的多元特征（有时不能从训练集中找到）对应的是低于平均数的观测值，分数也低（因为它们触发了许多欺诈检测规则）。

## 实现细节

每个顶级节点（或者多元特征）是隐性决策树的一个终节点。没必要采用剪枝或者分裂算法和准则：HDT直观而快速，能够充分利用高效的哈希表（键=特征，值=分数）。前500个节点用于对80%的交易分类（即打分），它们来自多个隐性决策树。称其为隐性决策树是因为我们不需要用决策树算法去生成它们。

余下的20%交易则是用别的方法打分的（通常是逻辑回归）。因此HDT其实是一个混合算法，将多个、小型、容易解读、不可见的决策树（只有终节点）和逻辑回归混合而成。

注意，对于逻辑回归，可以采用有限定条件的回归系数。这些系数依赖于两到三个顶级参数，并且与其所表示的规则和响应值或分数之间的相关系数的正负符号相同。这使得逻辑回归对“独立变量（规则）”之间很高的交叉相关系数不再敏感，其实这种情况下的独立变量并不独立。这种方法有点类似于岭回归 （*ridge regression* ）、逻辑回归（l*ogic regression* ），或者套索回归 （*Lasso regression* ）。这种回归方法用于对跟回归系数关联的顶级参数的调优。近似解（我们就是在运用近似逻辑回归），只要精心处理，跟精确解几乎一样准确，但要健壮得多。

我们处理下列这些类型的分数。

- 前500个节点为80%的交易产生分数$S_1$。
- 逻辑回归则为100%的交易产生分数$S_2$。

合并分数，需要进行如下操作。

- 对于80%具备两个分数（$S_1$ 和$S_2$ ）的交易将其$S_2$ 按比例调整 。调整比例是指通过线性变换，使得两类分数都有相同的均值和方差。假设$S_3$ 是$S_2$ 进行比例调整之后的结果。
- 不能用$S_1$ 打分的交易，则用$S_3$ 打分。

HDT节点为数据提供了另外一种分割　方法。其中，较大或中等的分数值的分割对应于中立型的交易（即不会触发欺诈规则）。分数很低的分割对应具体的欺诈案例。每一个分割里所有交易分数相同。HDT相比主成分分析 （*PCA* ）和其他分析方法，生成的是不同的细分结果。

# 示例：互联网流量打分

图5-1展示了一个基于20个规则系统生成的分数分布，每个规则都有一个低触发频率。该分布具备下列特点。

- 翻转钟形曲线。
- 低于425的分数代表不会付款的交易。
- 底部和顶部的尖峰用来调整分数尺度。
- 50%的交易分数良好。
- 打分卡参数。

- 下降50个点意味着转化率下降50%。
- 平均分是650。

- 模型改进程度：从翻转钟形曲线变成钟形曲线。

- 交易质量与欺诈检测。
- 添加逆向规则，将分数平滑化（并移除分数结块）。

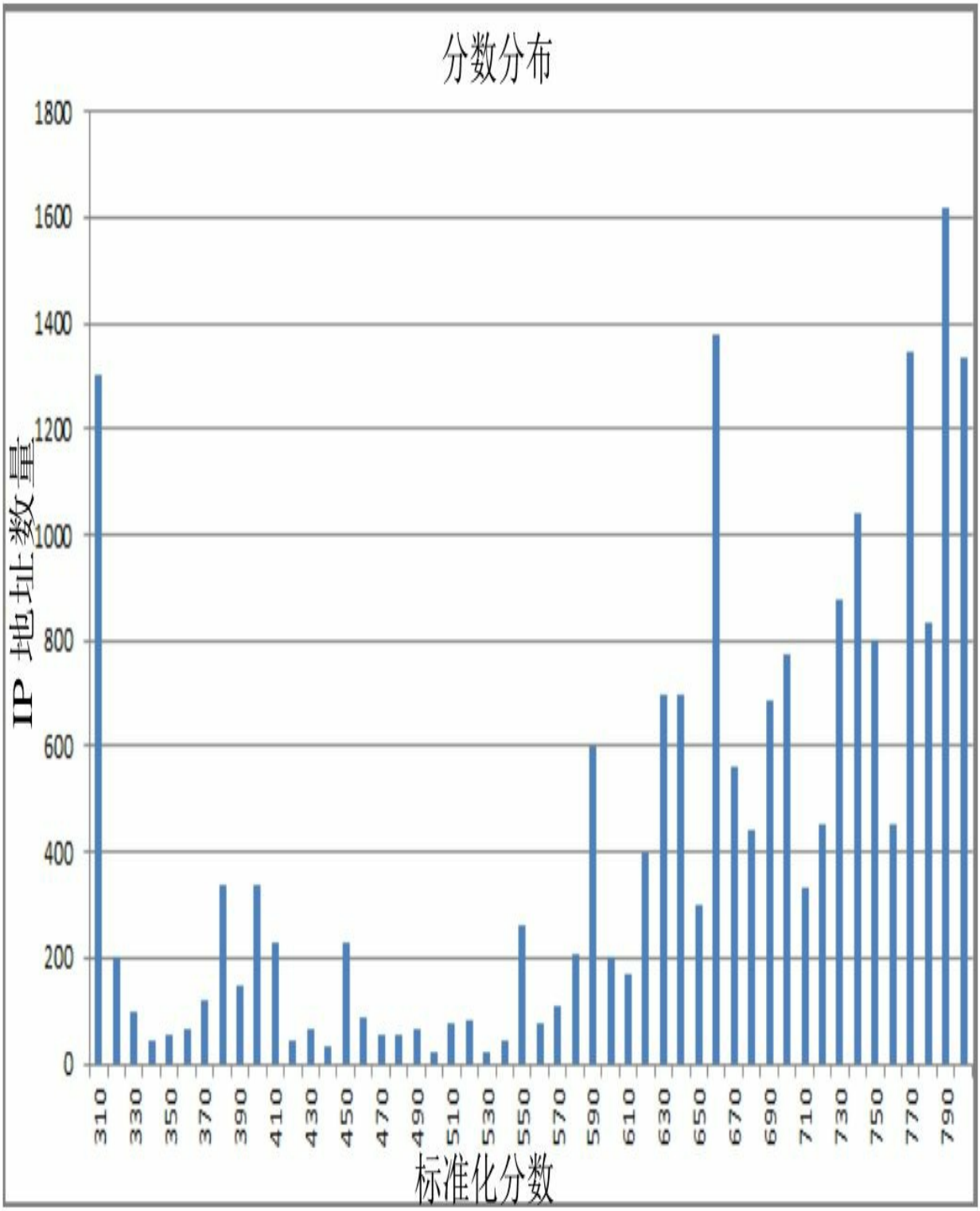


图5-1　HDT的分数分布示例

图5-2展示了转化率（CR）和分数的对比。将HDT运用到互联网数据上，对点击进行打分。其中一个分数用于预测转化率（转化代表着购

买、点击或者在载入页注册）。整体而言，会得到相当好的拟合结果。

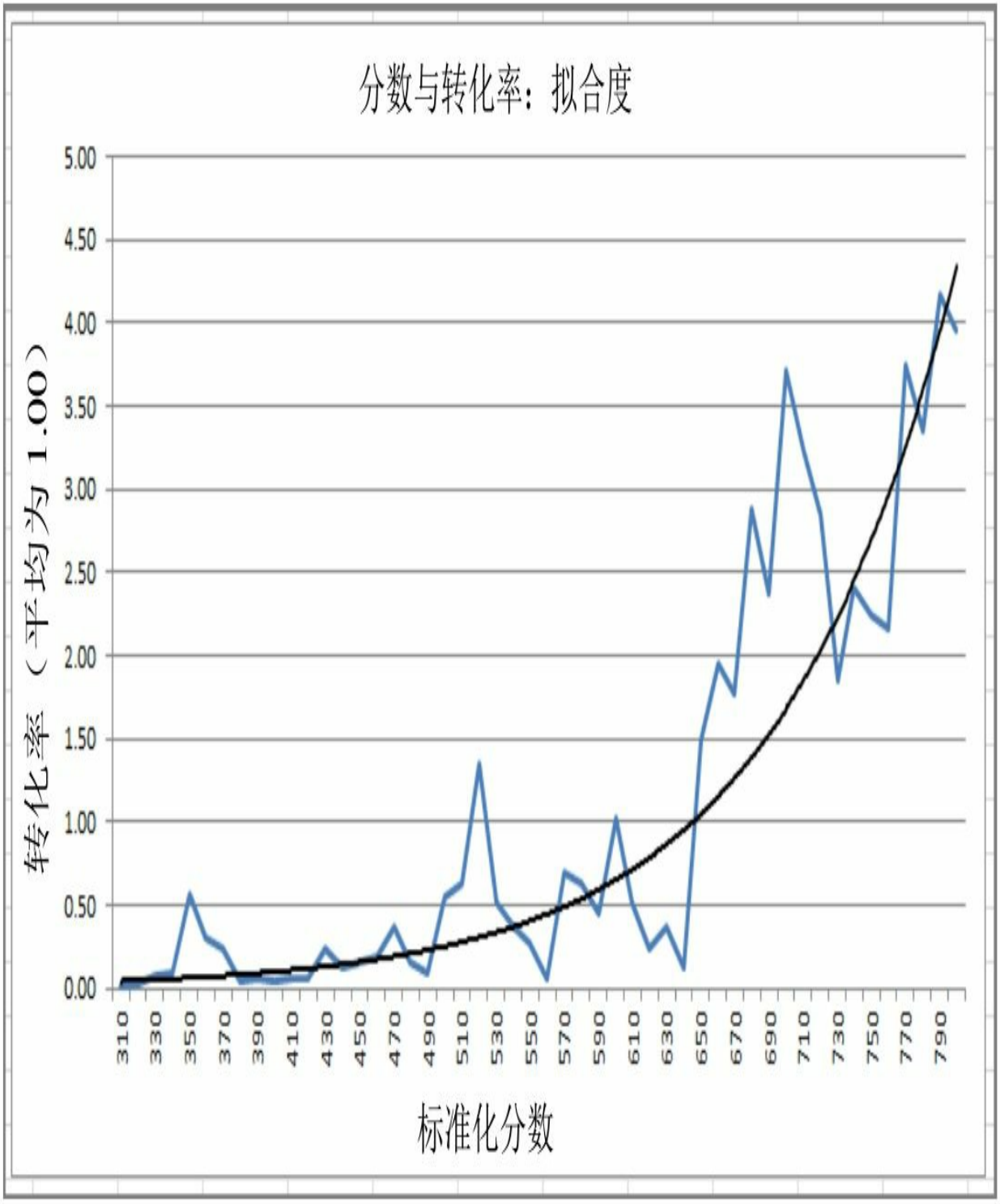


图5-2　用于预测转化率的HDT分数

图5-2中的峰值意味着：

- 错误的转化（如果转化仅仅意味着点击时经常出现错误转化）。
- 残留噪声。
- 模型需要改进（需要添加逆向规则）。

而谷底代表：

- 未检测到的转化（cookie问题，转化时间异常高）。
- 残留噪声。
- 模型需要改进。

# 结论

HDT是个快速算法，容易实施，能有效地处理大数据集，输出也容易解读。这是个非参数的方法，很健壮。过拟合　的风险小，只要不超过前500个节点，并采用交叉验证技术来去掉不稳定的节点。它还提供了一个内建而简单的机制来计算分数的置信区间（见下一节）。

HDT是可以检测多种类型结构的算法：通过回归算法检测线性结构，通过前面的节点检测非线性结构。HDT未来的发展方向包括能处理细颗粒度数据的隐性决策森林方法，以及层级HDT方法。

# 与模型无关的置信区间

本节你将找到容易计算、与分布无关、精确到小数点的置信区间值。如果给某个特定实验（比如对1　000万信用卡交易计算的分数）得到的观测值赋予随机的ID（标记为1,⋯ ，N），那么通过使用接下来描述的分析桥定理，很容易计算k个随机ID的任意百分比或是分数的置信区间。

# 方法

比如说你想预测某个参数p（比例、均值都行），那么你将如下这样做。

1. 将观测值划分为N个随机区间。

2. 为每个随机区间计算一个估计值。

3. 对估计值排序，从p_1（最小值）一直到p_N（最大值）。

4. 假设p_k是参数p的置信区间最低值，k小于N/2。

5. 假设p_（N−k+1）是参数p的置信区间最大值。

那么结果如下。

- [p_k, p_（N−k+1）]是p的非参数置信区间。
- 置信度是2k/（N+1）。

注意，通过尝试多个k值（以及N值，但尝试N值计算量更大），可以插值出任何置信水平的置信区间值。

最终，你会希望将N保持在尽可能低的水平（理想情况下k=1），以得到理想的置信区间水平。比方说，对于90%的置信区间，N=19和k=1能奏效并且是优化的。

# 分析桥第一定理

该定理的证明依赖于复杂的组合参数和对贝塔函数的运用。值得注意的是，最后的结果并不依赖于你数据的分布。简而言之，你的数据不需要是高斯分布，或者任何既有的统计分布，才能使得置信区间有效。你可以从E.J. Gumbel的*Statistics of Extremes* 一书中找到该定理的证明。

可以选择分析桥定理里的参数，以得到期望的精度，比如95%、99%或者99.5%的置信区间。该定理也告诉你要达到某个预定的精度水平需要什么样的样本尺寸。该定理属于基础性的成果，在许多场景下可用来计算简单、细到每个分割区、数据驱动、跟模型无关的置信区间，特别是为逻辑回归、岭回归、决策树或隐性决策树（例如用于欺诈检测或信用打分时）的结果生成预测分数时。

分析桥第一定理如图5-3所示。

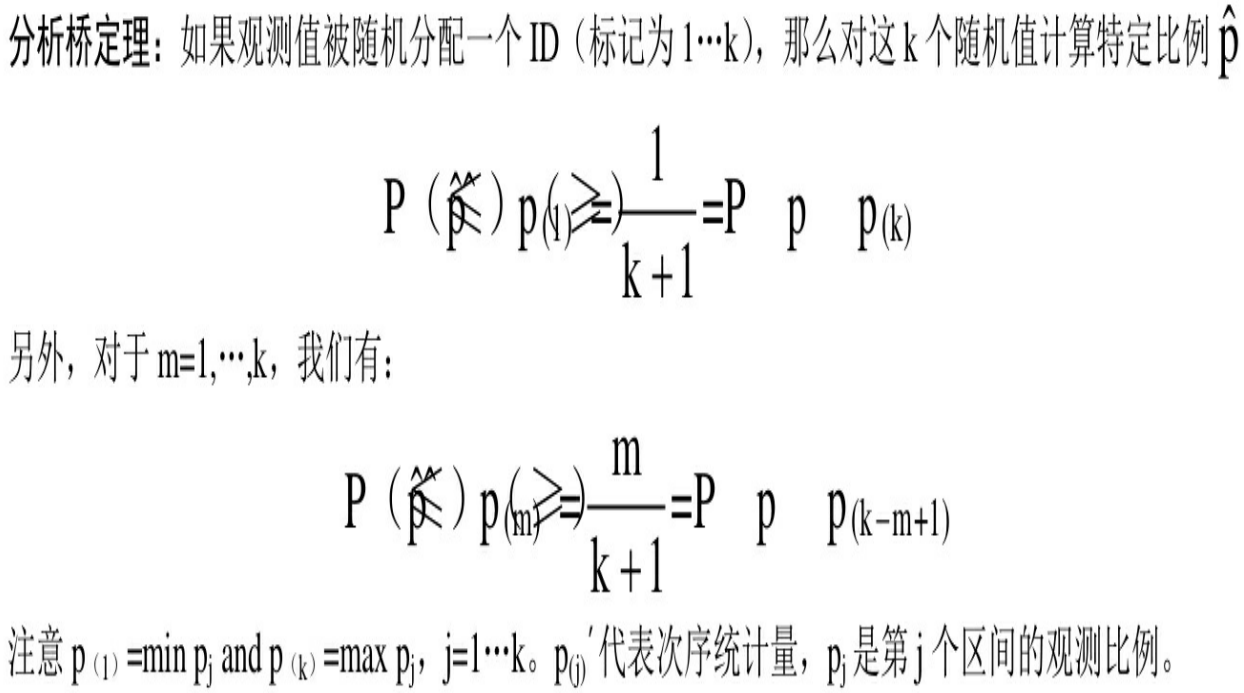


图5-3　分析桥第一定理

如果观测值被随机分配一个ID（标记为1…k），那么对这k个随机值计算任何比例的估计值p满足Proba[p < p（1）]=1/（k+1）=Proba[p > p（k）]。另外，对于m=1, . . . , k，会得到Proba[p < p（m）]=1/（k+1）=Proba[p> p（k−m+1）]。注意p（1）=min p（j）及p（k）=max p（j），j=1 . . . k。p（j）’代表次序统计量，p（j）是第j个区间的观测比例。

你可以从http://bit.ly/1a2hWi7 找到定理证明。

# 应用

判断顾客贷款违约的打分系统通常基于一套规则集。平均而言，每个顾客违约的比例是5%。有100万观测值（顾客）和信用分数、债务数量、薪水等指标的数据集，如果随机选择99个区间，每个区间包含1 000名顾客，那么违约率的98%置信区间（每区间1 000名顾客）是[4.41%或5.53%]。这是根据分析桥定理k=99和m=1计算得到的结果。（阅读该理论来理解k和m的含义，实际上很容易理解这些参数的意义。）

现在，查看一个非随机区间，该区间有1 000名信用分数<650、年龄<26的顾客的观测值，你会发现违约率是6.73%。因此可以推断出这样的规则：信用分<650，年龄<26是很好的违约判断规则，因为6.73%的违约率远高于[4.41%或5.53%]的置信区间。

实际上，你可以系统性地、自动地测试成百上千个规则，轻易地确定具有很高预测度的规则。这只需要查看（对于特定规则）观测到的违约率跟标准的置信区间偏差多远。这让你立刻排除噪声、流程及大量规则排序的影响（把违约率超出置信区间上限的程度作为预测能力指标）。

# 源代码

一名Data Science Central会员写了如下的R代码来展示该定理。他还创建了一个YouTube视频，地址是`https://www.youtube.com/watch?v=72uhdRrf6gM`。

```
set.seed(3.1416)
x <- rnorm(1000000,10,2)

analytic_theorem <- function (N, x, med, desv)
{
    length(x)
    meansamples <- rep(NA, N)
    for(i in 1:N){
      meansamples[i]  <- mean(sample(x, length(x)/N, replace=TRUE
    }
    meansamples <- sort(meansamples)
    confidence <- rep(NA, N/2)
    p_lower <- rep(NA, N/2)
```

```
    p_upper <- rep(NA, N/2)

    for(k in 1:N/2){
        confidence[k] <- 2*k/(N+1)
        p_lower[k] <- meansamples[k]
        p_upper[k] <- meansamples[N-k+1]
    }

    mean_t <- (meansamples[k] + meansamples[N-k+1])/2
    x_a <- seq(1:(N/2))
    par(mar=c(5,4,4,5)+.1)
            plot(x_a,   p_upper,   type="l",   ylim=range(c(med-
1*desv, med+1*desv)),
col="blue", xlab="K", ylab="Values")
    lines(x_a, p_lower, type="l", col="red", xaxt="n", yaxt="n",
ylab="")
    par(new=TRUE)
    plot(confidence*(N/2), rep(mean_t, N/2),
ylim=range(c(med-1*desv, med+1*desv)),
type="l", col="black", xaxt="n",
yaxt="n", ylab='', xlab='', lty=2)
   axis(3, seq(from=0, to=1, by=.25)*(N/2), las=0,
at=seq(from=0, to=1, by=.25)*(N/2),
labels=c(1,0.75,0.50,0.25,0))
    mtext("Confidence", side=3, line=3)
    legend("bottomright", col=c("red","blue"), lty=1,
legend=c("Lower bound","Upper Bound"))
    legend("topright", legend=c(paste("N=", as.character(N))))
}

for(i in seq(1000,10000,100)) analytic_theorem(i, x,10,2)
```

# 随机数

这里我们可以用一个简单且先进的技术来模拟基于某个数（比如π）的小数部分生成随机数。随机数广泛应用于许多统计应用和蒙特卡洛模拟算法之中。但许多人依赖于有缺陷的技术来生成随机数，典型的是调用`rand()`这样的黑盒子函数，而不知道其随机数生成的好坏。无数的分析师用Excel来创造随机数。但是Excel随机数生成器的缺陷是众所周知的，当它用于以高质量的随机性为核心要求的应用程序中时，会惹

出大麻烦。你可以找到更多的关于如何定义和测量随机性的例子，前面章节中提到的结构性系数便是其中之一。

本节讨论的是拥有无限周期的随机数生成器，换句话说，它没有循环。这个周期无穷无尽的原因在于它以无理数为基础。

π是数一数二的随机数字集合，大批的科学家对其进行成千上万、各类规模、广泛的随机性检测。当前算法通常使用递归的同余关系或随机排列合成和周期性展示，尽管π的随机性优于它们，但π还是面临着两大挑战，这两个挑战使其无法成为有用的随机数生成器：

- 假设所有人都知道π的小数部分被用于许多高安全性的加密算法以生成不可预测的随机性，然后你猜会发生什么？它会失去这个伟大的“不可预测性”。
- 计算π数以百万计的小数部分非常困难，它需要比传统的随机数生成器多得多的时间。

下面是我应对这两大挑战的答案及能解决这两个棘手难题的新随机数生成器的提案。

- 关于速度，你现在有极快的算法来计算π的小数部分（如下面的公式所示）。
- 关于使用π，你应该选择不那么流行的数字，可以通过类似的计算公式去进行计算，这样既保持速度，也使逆向工程在加密算法中变得不可能。这里有一个关于奇妙的随机数字生成器的例子：对于使用公式定义的数字，做如下这样的变化：用3, 1,−2,−2代替4,−2,−1,−1。你也许懂了——使用第一个变形公式就可以开发出数以万亿计的随机生成器。

尝试用以下两个快速公式来计算π。第一个公式是：

$Pi=Sum\{（A+B+C+D）/16^k\}$，其中Sum是对k=0, 1, 2 …的求和，A=4/（8k+1）、
B=−2/（8k+4）、C=−1/（8k+5）及D=−1/（8k+6）。

这个公式非常值得关注的一点是，在不经过前述步骤的情况下，人

们就能随意提取任意一个π的十六进制或二进制数字。

Chudnovsky兄弟于1987年发现了π的第二个公式，如下：

426880 * SQRT（10005） / Pi=SUM{ （6k）!（13591409 + 545140134*k）/[（3k）!（k!）^3（-640320）^（3k）] }，其中SUM是对k=0, 1, 2 …的求和。

似乎你可以得到一个基于第（k−1）位数计算第k位数字（十六进制）的递归公式。

测试模拟数字的随机性，要把π的数字作为一个时间序列，同时要计算自相关函数（或相关图），并且要看看它是否在统计上不同于0。要执行这个测试，你需要做如下步骤。

1. 在前十亿位数字上，计算自相关函数c（k），k＝1,2,…,100。

2. 计算m=max|c（k）|。

3. 对观察到的m值与一个在真正的随机时间序列下且以π为基础进行计算的理论值进行比较，看看它是否足够小。（可以用蒙特卡洛模拟来计算理论值。）

通过用任意的a, b, c, d来代替数字4,−2,−1,−1，研究第一个公式F（a, b, c, d），会非常有趣。当然我们知道，F（4,−2,−1,−1）=π。

对于参数a, b, c, d，这个有趣的函数F（a, b, c, d）是否拥有上述的两个属性呢?e=2.71这个数字（或者是其他非常漂亮的无理数）会成为F的典例吗？对于π，我们有：

a + b + c + d=0
|a| + |b| + |c| + |d|=8

注意　　另一个方法是利用连续分数去近似π。第n个近似值用p（n）表示，则：p（n）=A（n）/B（n），其中A（n）和B（n）是由简单递推关系定义的整数。稍微修改一下这些递推关系，那么：你会创建出一个新的有趣的数字，而这个数字有着随机数位（希望如此）。

或者构造一个简单的连续分数，如：

x=1/（1+1/（1+2/（1+3/（1+4/（1+5/（1+6/1+..））））））

尽管x的结构过程有很强的模式，但其数字显示出良好的随机属性。

# 解决问题的4个办法

对于数据科学家而言总有几个方法可以解决任何问题，所以在采取某个策略之前应该考虑多种选择。以下部分展示了解决这个问题的4种不同的方法：你想通过在某些谷歌论坛发布更多内容提高你的存在感，但是论坛太多了，你必须缩小其范围，这样才能找到在你的目标领域最流行和活跃的那一个。

所以，你每天首先要在目标群体列表中找出最受欢迎的谷歌论坛。每个论坛的唯一、快速、可用信息就是最新的发帖时间。很直观，最后发帖时间越新，这个论坛就越活跃。但是这个假设也不是万试万灵，有些论坛的帖子全部来自某个用户，甚至是机器人发布的，或者有些论坛只用于发布招聘广告。你应该把这类论坛从你的名单中淘汰。

一旦你确定了想要参与的论坛，就要确保其他论坛成员能够阅读你的帖子。为此，你的消息和标题是关键因素（和选择目标论坛同等重要），它能使你发布的帖子得到阅读并且转化为某种价值。这个例子假设你的内容优化问题（创新部分）已经单独使用其他数据科学技术处理过（例如，在不考虑其他影响因素的情况下，通过使用*A/B* 测试 并观察看客的反应，用统计显著性来鉴别哪些关键词在主题栏更吃香）。所以一个后续的假设是，你在撰写内容方面很有经验，并且现在对在保持相关性的同时提高帖子的阅读量非常感兴趣。

最后，我们假设你的博客获得了高分，所以你的帖子是大家普遍接受的，而没有被标记为垃圾邮件（这本身就是一个优化问题）。事实上，现代数据科学应用程序的一个共同缺陷是，一个问题常常需要由不同的团队进行多级优化。这就是为什么获得专业知识和商业智慧，以及识别并跟项目所涉及的各个团队对话，对数据科学家来说如此重要。

所以关键问题是：你怎么通过最后发布时间来估计某个论坛的活跃

性？我们可以使用以下4个方法来解决这个问题。

# 拥有超强直觉能力的业务分析师的直观法

每个单位时间的帖子数量约为最新帖子的时间后所流逝时间的2倍。如果你对数字有良好的判断力，即使你不做分析，也会感受到以上规律。实际上，这里有一个简单的经验解释。可能很少有人有这种一贯正确的直觉，也许你的员工里都没有。如果是这样的话，不能用这个方法（这是4个方法中成本最低的）。

注意，你并不需要估计确切的访客数量，而只需对不同的论坛进行排列。所以访客的相对数量（非绝对数量）就已经足够了。通过使用最新发布时间后所流逝的时间就可以得到这个相对数量。

# 软件工程师的蒙特卡洛模拟法

任何优秀的工程师，就算没有或几乎没有接受过系统的统计培训，在并没有真实发布任何东西的情况下，可以在论坛上模拟发布随机帖子，测试不同的发布频率，然后在每个测试中，选取一个随机模拟时间并计算出最新的发帖时间。然后，例如以20 000个论坛的发帖模拟作为基础，你可以计算出（实际上是重建）一个能够反映从最新发帖时间到发帖数量的图表。要警告的是：工程师必须使用良好的随机码生成器并且要评估图表的准确性，或者用前面章节介绍的分析桥定理构建置信区间。这即使对非统计学家们而言也是一个简单的技术。

# 统计学家的统计建模方法

基于随机过程理论（泊松过程）及厄兰分布，每单位时间的帖子数

量约为最新发布的时间后所流逝时间的2倍。这个理论也可以给你这个估计量（可以是无限的）的方差并告诉你使用前面的第2、3、4个帖子发布的时间更稳健，因为它们的方差已知且有限。现在，如果该论坛不怎么活跃，之前发帖时间本身可以是无限的，但实际上这不是一个问题。注意，这就违反了泊松假设。这个理论也告诉我们如何结合前面3次的发帖时间，以获得更好的估计量。欲知详情，请阅读我的论文 *Estimation of the intensity of a Poisson point process by means of nearest neighbor distances* （`http://bit.ly/HFyYse` ）。比起每天只对每个论坛进行一次测试，如果我们能够进行多个测试并取其平均值，这样就会得到更加精确的估计值。

# 计算机科学家的大数据方法

我们每天都逛不同的论坛，并仔细计算所有论坛上的帖子，并不是走马观花式的概要统计。重点在于使用一个分布式结构来快速浏览和处理数据，而不是研究一个小数据样本的有效采样机制。

# 因果关系和相关性

众所周知，统计分析善于发现相关性，但是相关性并不意味着因果关系。在一些项目中（如数据挖掘），若已经留意到外部因素引起的虚假相关性，区分相关性和因果关系并不是很重要。

很多情况可以用黑箱模式检测处理，而其所得的模式并不需要深入理解。对于许多应用程序，如高频培训来说，只要你的算法可行，就不需关心因果关系。但在其他应用中，如根除癌症的根源分析，如果要获得更大的成功，我们就要更深入地研究因果关系（虽然一些药物确有疗效，但是没有人知道其所以然）。在所有问题中，识别解释原因的因素并予以赋权，都会使我们得到更准确的预测结果，特别是如果我们使用了好的模型选择、模型拟合和交叉验证方法。但是获得先进的模型需要给统计学家们付高额的年薪，也许这样会使得投资回报率变为负值，而简单模式匹配算法则就投资回报率而言会更好。在这两种情况下，无论

是否关心因果关系，它仍属于科学范畴。实际上，它是数据科学，并且包含判断何时及是否需要更深层次的统计科学。同时，它经常涉及交叉验证和实验设计。只有统计理论建模方面可以忽略。其他方面，如可扩展性和处理速度，也必须加以考虑，这也是科学：数据和计算机科学。

在特殊情况下，如根源分析　，你想要确定因果关系，而不是相关性。例如，做更多的运动与降低肥胖率有很大的相关性。所以有人断定孩子们在学校应该多上体育课以预防和抑制肥胖。然而，缺乏体育锻炼并不是肥胖的根源，不良的饮食习惯才是真正的元凶，要解决肥胖这个问题，要从饮食习惯着手。然后，一旦饮食习惯得以改变，就可以用数学课、外语课代替体育课，这样不仅使孩子们减重，同时还能使他们接受更多的教育。

在任何情况下，使用直接因果关系的预测有助于减少模型中的方差和得到更有效的解决办法。

# 怎样检测因果关系

如果你处理超过200个变量，你会发现一些相关性在统计意义上是很显著的，但实际上这是由大量的变量造成的假象。例如，对10　000个变量模拟出50个观察值。每个变量中的每个值是一个模拟出来的随机数。尽管我们已经把10　000个变量中任意两个变量之间真正的相关性设计为零，但还是会从中发现两个高度相关的变量。再举一个与现实生活关系比较密切的例子，你可以设计一个函数以某个人日常消费香烟或其电费单来预测其在未来5年内患肺癌的概率。两者都是不错的预测因子，不过很显然，第一个是原因，而第二个不是，但它与年龄有关。随着年龄的增长，由于通货膨胀，商品越加昂贵，这是一种非因果因素，是肺癌的间接原因。欲知详情，你可以回头看看第2章中讲述“大数据‘诅咒’”的部分内容。

本节总结了几个各种背景下已经被用于确定因果关系的技术。

- David Freedman是一位著名作家，他的代表作有*Statistical Models*：*Theory and Practice*，在书中他与读者探讨了因果关系的问题。这

是一本讨论模型假设问题方面很独到的统计学书籍，我强烈推荐这本书。这本书声称是介绍性的，但我认为，如果没有一年半载学习数学统计知识的基础，你无法通读此书。

- 在时间序列的问题中，你可以尝试运行VAR模型，然后做Granger因果关系测试，看看一个变量是否会产生另一变量，这个定义来自于Granger（在任何一本Granger的著作上，我们都可以看到Granger因果关系的技术定义）。编程语言R有一个很好的工具包称为vars，用它可以非常容易地构建VAR模型和做测试验证。
- 虽然相关性常常伴随着因果关系而存在，但这也不是必然的，相关性并不意味着因果关系。我们还可以通过学习贝叶斯网的数据进行因果关系分析。下面的网站上有David Heckerman制作的一流教程——贝叶斯网络，详情可见 `http://research.microsoft.com/apps/pubs/?id=69588` 。
- Stefan Conrady负责管理贝叶斯网络公司（`www.bayesia.com` ），他专注于确定因果关系。
- 关于时间序列，请参阅`http://en.wikipedia.org/wiki/Granger_causality/` 。
- 区分相关性和巧合的唯一方法是对照实验。通过设计一个实验我们可以检测出每一个参数是否与其他参数独立。
- Judea Pearl写的*Causality* 充分阐述了这个主题。他用许多方法详细探讨了这个话题，其中包括直接、间接、混淆和反设事实等。在这里值得我们关注的是，因果模型着重于描述作用机制，也就是某事发生的原因，通常比相关分析更容易获得对事物本质上的理解。其中一种常用的技术可以描述如下。

1. 从因果模型图出发做出假设。（你可以使用结构方程模型图（SEM）。实际上，我更偏向于使用一种名为系统动态方程的技术。）

2. 接下来，找到数据来证实或者反驳你模型所做的一系列假设。对于不存在的数据，我们需要做重点实验。

3. 基于数据、相关性及sem分析，完善我们的因果关系理解。

4. 基于模型做仿真模拟，看看结果是否合理，然后改进。

5. 开始使用你的模型，随着新证据的出现不断完善它。

一个众所周知的发现因果关系的例子是肺癌，在这个例子中，吸烟

被认为是引起肺癌的真正原因而非干扰因素　。例如，在美国，肺癌与贫穷和吸烟息息相关。但事实上，贫穷并不是肺癌的原因　，它仅仅是一个干扰因素。事实是，许多过着潦倒生活的人是烟民，但使他们患上肺癌的真正原因是吸烟。你可能认为，贫穷让他们吸烟，这反过来会增加他们患肺癌的风险。但从来没有人证明这是一个因果关系，据我所知，它仍然是一个间接的联系。

另一个有趣的例子发生在印度，关于追查疟疾扩散的原因所做出的前期努力。第一个发现的原因是“污浊空气”吸引受感染的蚊子，因此实施专注于改善空气质量的解决方案。结果表明真正的原因不在于污浊空气，而在于携带疟疾的蚊子幼虫生活在滞水中。

为了找到适当的和有效的解决办法，急需找到问题真正的原因。任何时候当数据科学家被要求做根源分析时，它往往会涉及真正　原因的探索，而不仅仅是分析一些干扰因素。

# 数据科学项目的生命周期

生命周期这一关键词通常适于描述大多数大中型项目的通用任务分解。一个项目的生命周期也可以描述为其工作流程。项目的工作流程设计，包括设计软件工程、产品开发、统计分析，或数据科学，是项目成功的关键因素。它能帮助我们井然有序地组织规划任务，提供了一个项目的全局蓝图，并迅速确定优先级、问题、选择和里程碑。同时，它也是最佳实践 的一部分。

下面描述一个项目生命周期的主要步骤。

1. 辨识问题所在

- 确定你遇到的问题的类型（例如原型设计、概念验证、根源分析、预测分析、规范分析、机器到机器的实施等）。
- 寻找你的团队内外可以帮忙解决这个问题的核心人物。
- 明确权衡成功解决问题的方案指标。
- 选择适当的人选分别做出规格、需求、项目细节、时间线、优先级

和预算等策划。
- 确定所需的精度水平。
- 确定是否需要所有可用的数据。
- 判断是选择和构建一个内部解决方案，还是使用供应商的解决方案。
- 进行供应商比较（通常是进行分析，但有时也用BI或小公司数据库里的解决方案），选取项目推进评估和供应商比较所需的标准。
- 写提案或回复提案申请。

2. 确定可用数据源

- 使用合理的样本采样技术提取（或获得）和检查样品数据，并对字段进行讨论以确保你充分理解数据。
- 对数据使用EDA分析（探索性分析、数据字典）。
- 评估数据的质量和价值。
- 识别数据中的差错并找到解决办法。
- 确保数据质量足够好，确保字段的数据随时间有一致性。
- 确定一些字段是不是“混合物”（例如，关键词字段有时是用户查询关键词，有时是广告关键词——除非跟统计分析或与业务人员交谈，否则我们无从得知）。
- 确定如何提高数据质量。
- 确定你是否需要创建迷你汇总表或数据库来帮助你进行项目中的一些任务，比如统计分析。
- 确定你需要哪些工具（如R、Excel、Tableau、Python、Perl、SAS等）。

3. 如果需要，要确定额外的数据来源。

- 确定应该截取哪些字段。
- 确定数据的颗粒度。
- 选择所需的历史数据。
- 评估是否需要实时数据。
- 确定你将如何存储或访问数据（如NoSQL、MapReduce等）。
- 评估实验设计的需求。

4. 进行统计分析。

- 根据需要使用填补方法。
- 检测和删除极端值。
- 实施变量选择（变量缩减）。
- 确定数据是否是截断数据（例如生存分析和犯罪时间之类的隐藏数据）。
- 执行互相关分析。
- 根据需要进行模型选择，优选简单模型。
- 进行灵敏度分析。
- 执行交叉验证和模型拟合。
- 测量数据的准确性，提供置信区间。

5. 确保正确的实施和开发。

- 确保解决方案符合FSSRR原则，即快速的、简单的、可伸缩的、健壮的、可重复使用的。
- 确定多久需要更新查找表、白名单和数据上传等。
- 确定是否需要调试。
- 评估是否需要创建一个应用程序接口与其他应用程序进行通信。

6. 成果交流。

- 确定是把结果集中在一个仪表盘上，还是创建一个电子邮件警报系统。
- 决定与业务用户沟通的仪表盘架构。
- 评估可视化选项。
- 讨论潜在的可改进之处（同时需要做成本估算）。
- 作为项目的成果，为用户提供包括产品或解决方案在内的培训。
- 为用户提供代码注释和撰写技术报告之类的支持，解释如何使用你的解决方案，如何调整参数和解释结果。

7. 维护系统。

- 使用压力测试检测模型或实现的系统。
- 定期更新系统。

- 一旦解决方案的稳定性通过测试，就可以把最终的系统外包业务交给公司的工程和业务人员。
- 如果需要，把解决方案移植到一个新平台或供应商上。

你可以使用如上这个“生命周期”模板并根据你的项目进行定制。它可以用作你为客户编写的任何建议的骨架。请注意，并非所有项目都需要这些步骤。有些项目是内部的，不需要建议书。在另外一些情况下，你的客户（或雇主）已经有了数据库、BI和分析工具，所以没有必要为你的项目寻找供应商的解决方案，或是你的客户或经理想保持你项目的延续性而不想使用供应商。

# 预测模型的错误

当你按照前面的步骤进行分析时，还需要避免一些陷阱。这些错误在快速浏览博客中曾经被提及。下面的列表给出了更多实例。你可以在`http://bit.ly/19vrkeZ` 阅读更详细的版本。

数据预处理阶段的错误包括：

- 将ID字段作为自变量。
- 使用非同步的变量。
- 允许重复的记录。
- 在过小的群体上建模。
- 未考虑异常值和缺失值。
- 将两个编码稍不同的表格关联到一个域。
- 使用混合字段，例如一个关键词有时是用户查询关键词，有时是一个分类范畴。
- 可视化程度太低，没有将数据充分表达清楚。

建模错误包括：

- 没有考虑足够多的变量。
- 没有手工添加一些额外的变量。
- 选择了错误的Y因变量。

- 没有足够的Y因变量响应。
- 在错误的群体上建立模型。
- 只使用一个衡量方法去判断一个模型的质量。

为了避免数据预处理中的错误，你应该花更多的时间跟参与这项工作的业务人员进行沟通，以确保你对所有的字段含义了如指掌。创建一个数据字典并对你的数据进行测试，将进一步减少发生错误的概率。SQL查询时间过长，或返回的数据远远超过预期，是存在坏连接的一项指标。

最后，从尽可能多的变量开始，然后把范围缩小到一个有效的子集，是一条有效的途径。每当更改模型或删除/添加变量时，应该测量它新的性能或提升（减少了错误或残差，或增加了ROI）：预测的波动越小，模型越好。但切忌过度拟合！把数据集分割为测试和控制两个子集，方便进行模型验证。当训练集确定后，在测试子集上训练你的模型，同时在控制子集上测量模型的性能。

# 逻辑相关回归

这里你将看到一些传统逻辑回归的现代变体，它是统计科学的核心利器之一，适用于二进制或概率的响应值的问题。逻辑回归在临床试验、评分模型和欺诈检测中十分受欢迎。

# 变量之间的相互作用

在信用评估的问题中，我们需要开发一个使用回归公式的预测模型，如$Y=\sum w_i R_i$，其中Y是比值比（虚假与真实）的对数。另外一种相关但略有不同的情况是，处理Y是二进制值的逻辑回归，例如Y=1表示错误交易，Y=0表示无错误交易。自变量$R_i$，定义为欺诈规则，是一个二值的标记量，例如：

- 高金额的交易。
- 高危国家。
- 高风险的行业。

这是一阶模型。二阶模型包括交叉乘积项$R_i * R_j$，用于表达交互规则。如何最好地计算回归系数$w_i$（也称为规则赋权）。存在的问题是很多规则相互重叠，使得这种回归方法非常不稳定。有一种方法是约束权重，迫使它们变成二进制（0/1）或者让它们与规则自变量和相关因变量Y之间的相关系数同符号。这种方法与岭回归 相关。

注意，当权重是二进制值时，这就是一个典型的组合优化问题。当权重被约束为线性无关的一组整数，那么每个$\sum w_i R_i$（有时称为非变形分数）对应于一个独特的组合规则。同时它也唯一地代表着有规则定义的决策树的最后一个节点。另一种方法是逻辑回归 ，或隐性决策树，在本章前面已有所述。

# 一阶近似

现在来看看线性回归的一个近似解。逻辑回归实际上是在一个变形的响应变量上定义的一个线性回归，下面将会讲到逻辑回归与Excel的一个章节。所以解决线性回归问题是很有用的。在本节中，你至少需要对矩阵理论有些基本理解，不擅长数学的人可以跳过本节。

对于一个回归方程Y=AX，其中Y是响应变量（向量），X是输入变量，A是回归系数向量。思考下面的迭代算法（k代表迭代次数）：

$$A_{k+1} = cYU + A_k \ （I - cXU）$$

其中，

- c是任意常数。
- U是任意矩阵使得YU具有和A相同的维度。例如，U=转置（X）。
- $A_0$ 是A的初始估计值。例如，$A_0$ 是独立变量和响应变量之间的相关向量。

其他几个问题

算法收敛的条件是什么？是不是只在矩阵的最大特征值（绝对值）小于1时算法才会收敛？在收敛的情况下，它会收敛于回归问题的解吗？例如，若c=0，该算法会收敛，但并不是问题的解。在这种情况下，它收敛于$A_0$。

其中的参数如下。

n是独立变量的个数

m是观察值的个数

矩阵维度如下。

A:（1, n）（1行，n列）

I:（n, n）

X:（n, m）

U:（m, n）

Y:（1, m）

那么，问题出现了，为什么要使用迭代算法而不是传统的解法？有几种原因，如下。

- 我们正在处理的是病态问题，其中大多数独立变量是高度相关的。
- 许多解法（只要回归系数是正数）都可以得到不错的拟合效果，全局最优解比一个所有的回归系数等于1的解好不了太多。
- 这里的方案是使用迭代算法，从第一步迭代的一个性能不错的近似解开始，迭代一步提高一点性能，直到算法停止。

# 二阶近似

如果你不熟悉矩阵理论的基本概念，你可以跳过这部分内容。在这里我们需要通过如下公式理解回归问题的一阶解：

$$Y=\sum w_i R_i$$

其中，Y是响应变量，$w_i$ 回归系数，$R_i$ 则是独立变量或规则。假设变量的数量很大，并且独立变量是高度相关的。

我们希望通过二阶回归的形式来改善模型，如：

$$Y=\sum w_i R_i +\sum w_{ij} C_{ij} m_{ij} R_i R_j$$

这里，

$c_{ij}$ 是$R_i$ 与$R_j$ 的相关系数。

$w_{ij} =|w_i w_j |^{0.5}$ sign（$w_i w_j$ ）。

$m_{ij}$ 是任意常数。

在实践中，一些$R_i$ 是高度相关的，可以被分为规则簇。这些簇可以用$c_{ij}$ 上的聚类算法加以识别。下面我们来看一下有两个簇A和B的模型，例如：

$$Y=\sum w_i R_i + m_A \sum_A w_{ij} c_{ij} R_i R_j +m_B \sum_B w_{ij} c_{ij} R_i R_j$$

其中，

$\sum_A$ （或$\sum_B$ ）是对属于簇A（或簇B）的所有i和j（且i<j）的求和；若i和j属于簇A，则$m_{ij} =m_A$ （常数）；若i和j属于簇B，则$m_{ij} =m_B$ （常数）。

当簇的结构足够强时，一个有趣的情况就会发生，结果如下：

若i和j属于同一个簇（A或B），$c_{ij}$ 的绝对值等于1，相反，$c_{ij}$ 等于0。

这种特殊的情况导致如下结果：

$$m_A = 4/[1+(1+8k_A)^{0.5}]$$

$$m_B = 4/[1+(1+8k_B)^{0.5}]$$

其中：

$$k_A = \sum_A |c_{ij}| 和 k_B = \sum_B |c_{ij}|$$

注意　　如果簇的结构强度适中，且相关系数$c_{ij}$ 接近1、-1或0，那么前面涉及$k_A$ 和$k_B$ 的公式准确性如何呢？这里我们假设$w_i$ 是已知值或近似值。通常，$w_i$ 是一个常数或Y和$R_i$ 之间的相关性的简单函数。

## 替代方法

现在，来看一个简化的只有一个簇的模型，其中$m_{ij}$ =常数=m，这个简化模型可替代本节开头所述的模型。例如，这个唯一的簇可以包含所有$|c_{ij}| > 0.70$的变量i和j。模型可以表示为：

$$Y = \sum w_i R_i + m\sum w_{ij} c_{ij} R_i R_j$$

就残差而言，你想知道最能改善一阶模型性能的m的取值。一阶模型对应于m=0。现在看看以下的表示法：

$W = \sum w_{ij} c_{ij} R_i R_j$，

$V = W - u$，其中u=average（W）。（V是W的中心，均值为0）

$S = \sum w_i R_i$。（则S的平均值等于Y的平均值）

在不失一般性的情况下，看看稍稍修改过的模型：

$$Y = S + mV$$

其中m等于：

m=[Transposed（V）*（Y−S）]/[Transposed（V）*V]

其中Y、S、V是n行的向量，n是观测值的数量。

### 进一步的优化

用于信用评估的迭代算法的一个替代措施是在每一步都增加一个新的簇。所以，每一步我们都要对m进行重新计算，优化Y=S + mV的残差。然而，此次S包含了之前所进行步骤中发现的所有簇，而V则包含了新加入模型中的簇。

# 用**Excel**进行回归分析

值得一提的是，即使不使用数据分析插件，你也可以用Excel进行逻辑回归分析。若你的响应值是在0和1之间的比率r，使用分对数 变换s=ln{r/（1-r）}，然后在s上执行线性回归，接着使用逆变换r=exp（s）/{1+exp（s）}从s切换回r。

多元线性回归在Excel中也有，定义为函数*linest* 。输入是一个矩形数据集，第1列是响应值，余下各列是独立的变量。输出是一个有着所有回归系数的向量。因为输出并不是单个数据，而是一个数组（linest是Excel中的一个数组公式）。你需要指定输出的数组形式数据在电子表格的存储位置，可以使用Ctrl+Shift+Enter键来选择数据。你可以通过以下网址获得更多的信息：`http://office.microsoft.com/en-us/excel-help/linest-function-HP010342653.aspx` 。

# 实验设计

实验设计有时称DOE（设计实验）、A/B测试、多元测试（网络分析），或是田口方法（质量控制），它是统计分析中的一个基本步骤，以确保设计出合适的调查或分析，以及能够得到可用的数据和消除无法

控制的偏离。这里有一个类似于临床试验的例子：测试戒除酒瘾的策略。

# 有趣的指标

对于监控和保证戒酒成功来说，有3个依赖酗酒者本身的指标至关重要。

- 喝酒的时间间隔 ：对于大部分戒酒者来说，5天是最小的间隔，而且在恢复喝酒前的30天必须节制。
- 喝酒的持续时间 ：例如，周五和周六晚上，共两天。
- 喝酒的强度 ：例如，喝酒精含量低于50%到70%的酒。

康复中心根据这些指标判定最有可能恢复的患者，甚至可以基于患者对这些问题的答案来决定是否收治他们。

# 把患者分成不同的人群进行治疗

我们可以粗略地将患者分为4类。

- 生活正常的酗酒者。
- 生活紊乱的酗酒者。
- 生活正常的饮酒障碍者。
- 生活紊乱的饮酒障碍者。

现在我们可以引入一些新的分类，分别是想喝醉者和不想喝醉者。两种类型的喝酒量不相上下，对身体的影响也差不多，但第一种似乎是永久性醉酒者，而第二种貌似从来不会喝醉。这在某些方面与生活正常与紊乱的二分法有所联系。假设它是不同的，这就产生了8个分类。

- 想喝醉的生活正常的酗酒者。
- 想喝醉的生活紊乱的酗酒者。
- 想喝醉的生活正常的饮酒障碍者。
- 想喝醉的生活紊乱的饮酒障碍者。
- 不想喝醉的生活正常的酗酒者。
- 不想喝醉的生活紊乱的酗酒者。
- 不想喝醉的生活正常的饮酒障碍者。
- 不想喝醉的生活紊乱的饮酒障碍者。

# 私人定制的治疗

基于这个分类，可以进行两种有趣的研究，假设你有一个很好的方式可以给想治愈的人进行分类，同时假设他们不需要担心个人隐私。（酗酒者可能不愿接受治疗，甚至不愿意别人把他们归类，因为他们担心酒瘾会反扑。）

1. 对于这8类中的每一类，什么是最佳的基于分类的治疗方案呢？

2. 使用马尔可夫链模型来创建一个8×8转换概率矩阵，评估在未来6个月中，患者从其中一类转变为另一类的概率。然后使用该模型来评估风险和决定应该优先治疗哪类人以及如何治疗。

我们可以在马尔可夫链模型中添加治愈和死亡这两项重要的状态。当然，治愈指的并不是永久性的，因为还是有复发的可能。此外，转移概率与很多因素相关，如年龄、人们处于当前状态的时间长度（前面已提到过的8种状态）和复发的次数等。所以转换（随机）矩阵不是8×8，而是10×10。除了死亡状态外，人们可以从任何一个状态转变为另一个状态。

这种分析的其中一个目标是创建一种新型康复中心，这个新型中心可以识别并服务少数最有可能康复的患者。5年后，此类康复中心的成功率能达到70%。与大多数项目还不到20%的成功率相比，它极具市场营销效应，而且因为在患者还没有出现任何症状的情况下就可以停止治疗，其费用低于常规治疗的20 000美元。因此，他们需要的医疗看护很少，甚至大部分时间并不需要。我的猜测是，这些中心只是想用其特有

的办法去迎合不想喝醉的生活正常的酗酒者这一类患者的治疗需要。

请注意，这个框架可以应用于任何类型的化学成瘾，也许甚至对性、赌博或者挥霍成瘾都有疗效，当然，要先对患者进行测试。

实验设计是统计学的一个重要课题。它先做出假设和设计实验，然后进行测试和优化。实验设计中常见的错误包括：欠采样、没有检测和消除混淆变量、错误地假设观测符合某特定的分布（仅在以推断为目的的情况）、使用过小的数据桶进行统计推断（比如以一个小数据集估计3×3的变量相互的影响作用）、在检测中被忽略或不存在但在检测之外存在的数据的相互影响，以及在测试集中选择的观测值有偏差。比如测试一种新的药物时，如果你给参与者金钱奖励，那么样本数据就无法代表普罗大众。

# 分析即服务和应用程序接口

现在越来越多的个人和公司使用现代网络平台提供分析解决方案：如Data-Applied、DataSpora、AnalyticBridge等。作为我们浏览器里的应用程序的一个例子，`www.RonCloud.com` 使得我们可以在iPad（甚至手机里）运行复杂的R程序。在RonCloud（`http://bit.ly/1cJsvHm` ）上可以找到相关的探讨。

虽然10年前分析即服务这个概念就已经出现，但直到现在，由于廉价的网络服务器才可以处理很大的带宽，甚至在不用云的情况下可以在数秒处理百万字节的数据，此外，因为互联网用户宽带连接的速度很快，所以才能开发出在线、按需、实时的处理数以百万计的观测值的分析应用程序，并都用API或on-the-fly形式传送结果。在某些情况下，结果是由处理过的数据集构成的，它被当成一列加入到输入文件中，使得有时候输出结果相当大：例如，这新加入的一列（输出的结果）是观测值的得分，对应于每一个观测值。这个解决方案的实现需要随机地选用某些统计技术并用隐性决策树有效地处理数据，需要使用较小的内存和高效的数据结构，从而允许用户在线处理实时的大数据集，这是R或其他统计软件包甚至在桌面电脑上都无法处理的。实际上，R和Salford系统等传统统计软件包需要将全部数据都存储在内存中，尽管有像RHadoop那样的大数据管理系统，当输入文件的观察值超过500 000个

时，这些软件包通常还是会崩溃。

有趣的是，这种新型的分析服务可以依赖于某些流行的统计软件包，如SAS，或者可以使用Perl（包括有图形处理函数库的图表产品）、Python、C、C#、Java编写的任意的算法。基于SAS的版本可能被称为SAS网页服务器（外联网和内联网）。

# 工作原理

其所涉及的步骤如下：

1. 向安装了SAS的外联网发出API调用指令，API调用中的参数描述了所需的分析类型（逻辑回归等）。

2. Perl/CGI脚本处理HTTP请求、提取参数，并自动撰写与用户相关的SAS程序。

3. SAS程序在命令行模式的Perl脚本中运行，并生成一个输出文件，如一个图表、XML或数据文件。

4. Perl脚本读取图表，在浏览器中展示出来（如果正在使用Web浏览器的是人类），并且给用户提供一个网址，可以获得表格（假设用户是执行API调用命令的网络机器人）。

# 实施案例

在使用收费服务之前，你必须购买一定的交易额度并定期往账户里添钱。一次交易就是一个对API的调用。我们可以通过HTTP来访问这项服务，如：

```
http://www.datashaping.com/AnalyticsAPI?clientID=xxx&
dataSource=yyy&service=zzz&parameters=abc
```

在执行请求时，就会出现下列情况。

1. 脚本会检测用户的账户里是否有足够的美元。

2. 如果足够，它会在Web服务器的客户端获取数据：数据源的链接地址是yyy。

3. 脚本检查数据源是否可用，或客户端服务器是否无法连接。

4. 然后它就会执行服务zzz，通常是预估评分的算法。

5. 参数字段会告诉你，你是在训练数据集上训练预测器模型，还是在训练数据集以外的数据上执行评分预测。

6. 在训练阶段，它能够快速处理数据，只需数秒就可以处理1000亿个观测值。

7. 完成后会通过电子邮件将生成结果的数据服务器地址一并发送到客户端。（地址也可以作为一个API的调用，当成一个额外的字段，同时采用一种防止文件冲突发生机制。）

8. 客户的预算也随之更新。

以上所述步骤的实施不需要人工干预。可以用网络机器人检索分析得到的结果数据，然后再自动集成到客户的数据库。训练评分比对训练数据集以外的观察评分的收费要更高。给一个观察值评分算作一次交易，有可能仅收取0.0025美元。

这一软件架构适合于按日或按小时进行的处理，但如果没有把参数设置为“训练”，它也可以用于实时处理。然而，当初在设计软件架构时，我的想法是处理大批量的交易，也许是每次1亿个。

总之，AaaS（分析即服务）和API是数据科学家们应该了解的重要话题。很多时候，我们可以通过API访问数据，而一些数据库也会提供API访问。另外，自动数据科学（和机器到机器之间的信息传递）在很大程度上依赖于API和AaaS。设计API是工程学和数据科学之间的界线。

# 关键词相关的**API**的源代码

下面是第2章描述的“大规模数据集的聚类和分类”的一个应用。如前所述，这个应用程序被所有搜索引擎用于识别与特定关键词相关的关键词。本节拥有一切你从零开始创建第一个API所需要的东西。阅读和下载以下内容，可以让你忙上一段时间！

- 托管应用程序的HTML页面。特别地，它包含了API使用的包括所有参数的网页表单。地址是：`http://bit.ly/1de7qoY`。
- 关键词同频表格先前已经描述过了。它保存为27个文件；例如kwsum2s_SRCH_CLICKS_p.txt对应于以字母p开头的关键词，而kwsum2s_SRCH_CLICKS_0.txt则对应于不以字母开头的关键词，如2013、98029（邮编），或1040（如“国税局纳税申请表1040”）。它是一个压缩的zip文件（7 MB），你可以在`http://.bit.ly/1de7qoY` 处下载。未压缩的版本包含27个文件，可以使程序运行速度快27倍。
- 你可以在`http://.bit.ly/1aJZBdj` 处下载Perl脚本kw8x3.pl。根据所给的预先计算好频率的关键词同频表格，它可以计算出关键词之间的相关性。

这个应用程序是用简单的Perl语言编写成的，但可以很容易地转化为Python编程语言。而且不需要特殊的Perl库。以后你会遇到API的另一个例子，在这个例子中需要下载特殊的库（连同网络爬虫的源代码和说明）。

要使API运行，首先要在你的电脑或服务器上安装小型的UNIX模拟环境（可在`http://cygwin.com/install.html` 处下载），然后安装Perl（可在`http://www.activestate.com/activeperl/downloads` 处下载）。如果你想要API作为Web应用程序工作（像在`www.frenchlane.com/kw8.html` 上一样），就必须把它安装到Web服务器上。使用UNIX命令chmod 755，将Perl脚本（所有扩展名为.pl的文件）编译为可执行文件（通常在/cgi-bin/目录中）——在这种情况下是chmod 755 kw8x3.pl。可以在`www.frenchlane.com/kw8.html` 测试应用程序。

## **API**调用的两个例子

以下两个例子关于如何使用API来找到与关键词数据相关的关键词。单击URL，复制结果。注意，在第一个例子中，参数模式被设为静

默，而且没有指定相关系数。在第二个例子中，模式被设为详细打印，另外，正如上面建议的相关系数设为$n12/sqrt（$n1*$n2）（其中，n1=x、n2=y、n12=z）。

例**1**

- API调用的地址为：`http://www.frenchlane.com/cgi-bin/kw8x3.pl?query=data&ndisplay=10&sort=Descending&mode=Silent&filter=Yes&boost=&correl=`
- 简化地址：`http://bit.ly/1biu1mc`
- 返回结果：

```
data recovery
data sheet
data base
data cable
data management
recovery
data entry
data protection
data from
data storage
```

例**2**

- API调用的地址为：`http://www.frenchlane.com/cgi-bin/kw8x3.pl?query=data&ndisplay=10&sort=Descending&mode=Verbose&filter=Ye correl=$n12/sqrt（$n1*$n2）`
- 简化地址：`http://bit.ly/15td3Oi`
- 返回结果：

```
0.282 : data =data recovery= 2143:171:0.245
0.167 : data =data sheet= 2143:60:1.066
0.139 : data =data base= 2143:42:0.571
0.138 : data =data cable= 2143:41:1.414
0.134 : data =data management= 2143:39:0.512
0.121 : data =recovery= 2143:928:0.637
0.116 : data =data entry= 2143:29:1.068
```

```
0.112 : data =data protection= 2143:27:1.074
0.105 : data =data from= 2143:24:1
0.103 : data =data storage= 2143:23:1.217
```

## 结果

结果可以用你的浏览器从Web应用程序中手动恢复，例如当你点击提供的链接时，或用网络爬虫批处理或实时处理时。注意，这个示例中使用的相关公式$n12/sqrt（$n1*$n2）与第2章“大规模数据集的聚类和分类”一节中所描述的公式是一致的。唯一的区别在于，在第2章中，$n1、$n2和$n12分别被称为x、y、z。>>更多新书朋友圈免费分享微信xueba987

在例2中，返回的结果是$n1=2143、$n2=928和关键词对{数据，恢复}的correlation=0.121。注意，$n_1$ =2143是你下载的同频表格中关键词数据出现的次数。$n_2$ =928是关键词恢复出现的次数，而$n_{12}$ 则是数据和恢复同时出现的次数（例如，在同一个网页或用户查询中），如同频表格中所示。

这个API的一个小技巧是，当API调用的控制参数为空时，它可以接受用户提供的公式以$n1、$n2和$n3为基础来计算关键词相关性。正因为如此，API从kw8x3.pl内创建出一个辅助Perl脚本，称为formula.pl，它和父脚本（kw8x3.pl）位于同一个目录。然后父脚本会调用存储在formula.pl中的getRho子程序来计算相关性。下面是formula.pl的默认代码：

```
sub getRho{
  my $rho;
  $rho=$n12/sqrt($n1*$n2);
  return($rho)
}
1;
```

formula.pl的存储路径是`/home/cluster1/data/d/x/a1168268/cgi-bin/` 。在你的服务器上安装应用时要相应地更改存储路径。同样你也可以使用一个包含*from* 、*the* 和*how* 等的停止词表来改善这个API。

最后，请记住，这只是一个起点。如果你想使它成为一个高质量的应用程序，你还需要增添一些亮点。特别是，你必须使用不能被分解成

单独单词的关键词查找表，如“New York”、“San Francisco”等。你还需要使用一个关键词停止词表，做一些关键词清洗。（你可以把“traveling”规范为“travel”，但不要把“booking”归为“book”。）你用来创建同频表格的数据也很重要，它必须包含数以百万计的关键词。如果使用得太少，你的结果看起来便不够充分。如果使用得太多，结果则会显得充满噪声。在这个应用程序中，使用的是多种数据的结合。

- DMOZ中约100万个类别和网站描述（公共数据）。
- 数百万来自搜索引擎的用户查询（私人数据）。
- 从数百万网页中用网络爬虫提取的文本（公共数据）。

# 其他主题

本节讨论一些更专业的主题：数据保存（这点很重要，当你把来自不同客户、不同来源或不同时期的数据融合起来，随着新数据文件的出现，将不再与旧版本兼容）、网络爬虫的优化（数据科学家应该熟悉网络爬虫的使用，因为它可以爬取大量充足的非结构化数据）、快速数据库连接，改善预测模型的技巧和模拟簇的源代码。

# 当数据库改变时，保存好数值

数据的变化来自多个数据源：例如，突然修改互联网流量数据中的访客的定义，导致访客数量突然下降或暴涨。网络日志文件定期改变，例如如果不再记录完整的代理字符串，流量的质量评分将受到影响。或者一个客户有数据字段与其他客户不同或仅仅是部分重叠。

你怎么处理这个问题？答案很简单：当检测到数据的变化（无论是你的评分算法或数据发生了变化）时，使用变化前至少两周的数据。比较这两周数据的新旧评分分数，然后使用这两周中有价值的数据修订新的分数，使它们保持一致性（例如，相同的中位数、相同的方差）。

如果不是时间的问题，而是不同的客户有不同的数据集，那么就使

用两个数据集的一个子集，子集中的数据字段是兼容的，而且这个简化数据集可以为不同用户计算评分值（可以与用完整的数据集计算出来的评分值进行比较）。这4个数据（包括两个客户、简化数据和完整数据）将用于模型校准。

在使用这个方法时，要注意以下事项。

利用突变点、趋势反转或斜率变化检测算法去检测数据的突变。然而，你找到的突变通常是无理可循的，并且绝对肉眼可见的，甚至在非统计学人士看来也很明显（通常，不幸的是，他是你的一个客户）。

当你提高评分算法，如果它使A上的分值提高，但是B上的分值降低，那么就用B的旧值和A的新值构造一个混合的分值。

# 优化网络爬虫

众所周知，网页爬取的速度很慢，因为它要用2秒才能完成一个HTTP请求。当你爬取数十亿网页，便需要数十亿秒。假设你要搜索的网页是精心挑选的，如此一来，缓存、浏览和索引表已经充分优化。

以下步骤实现80 000倍的速度提升。

1. 使用云计算：把你爬取的范围扩散至8 000个服务器。速度提升：8 000倍。

2. 在每个服务器上，运行20个网络爬虫的副本，并使之并行（在服务器级别上称为子任务并行化）。你可以预期（根据我的经验，假设每个服务器专门用于爬取项目）提高的速度达不到20倍，但也许达到5倍。到目前，速度提升：8 000×5=40 000。

3. 把超时阈值（与每个HTTP请求相关联）从2秒变成0.5秒。这样可以把速度提高3倍（并不是下载所有的网页都需要2秒），但是你也需要再访问更多因为这0.5秒而下载失败的页面。正因为如此，没有3倍的增益，只有2倍。注意，你应该尝试用不同的值作为阈值，直到找到一个最优值为止。到目前，速度提升：8 000×5×2=80 000。

4. 除了改变超时阈值，你还可以改变数据最大值的阈值，如果页面超过24KB，只下载24KB，并跳过其余部分。当然，尽管这样可以提高速度性能，但会造成信息丢失。

5. 你也应该有一个你不想爬取资料的网站或网页的黑名单，因为它们一直加载缓慢或引起其他速度问题（多个重定向等）。

6. 不要反复访问同一页面，会有陷入无限循环的风险。使用一种不会再次访问之前访问过的页面的机制，例如，时间间隔为30天。

下面是一个爬取网页的Perl代码：它访问大数据上的Tw$_i$ tter页面，然后在本地机器上建立副本`test.html`。网页的内容存储在变量`$page`（文本）中。

```
#!/usr/bin/perl

require LWP::UserAgent;

my $ua = LWP::UserAgent->new;
$ua->timeout(2);
$ua->env_proxy;
$ua-
>agent("Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 5.1;)");
$ua->max_size(200000);

$url="http://twitter.com/search?q=%23BigData&src=typd";

my $response = $ua->get($url);
if ($response->is_success) {
  $page=$response->content;
  $score=0;
  $success++;
} else {
  $error=$response->status_line;
  print "$error\n";
  $page="-";
}

open(OUT,">test.html");
print OUT "$page\n";
close(OUT);
```

# 哈希连接

哈希连接 ，是一些数据库供应商提供的一个选择，效率很高，尤其在处理稀疏数据时。连接通常是典型的Cartesian产品，而且在许多数据库系统中速度很慢。

举个例子，如果你连接A和B这两个表，每个表都有100万行，`where` 条件把表A中的行数缩减至1万，把表B中的行数缩减至50万，一个有效的建立连接的解决方法如下。

1. 识别表A中的1万行。

2. 把这1万行放进哈希表中。

3. 使用快速查找表浏览表B的100万条记录，为表B的每500 000行信息检查哈希表，满足与表B相关的`where` 标准。

这远比一个完整的Cartesian积更有效，因为它要包括10 000亿行。显然，最终的解决方案（除非它是一个外部连接 ）最多只能包含10 000行。同时，在一个脚本语言（如Python或Perl）中，可以很容易地写出两三行执行这种类型连接的代码。

# 用于模拟簇的简单源代码

下面是一个简短又简单的用于模拟簇结构（存储在名为`cluster.txt` 的文本文件中）和测试簇检测算法的源代码。这个版本是用Perl编写的，代码很简单，易于改编为任何语言：R、C或Python。在`http://bit.ly/V5FVI2` 上，你可以看到由下面的脚本得出的模拟结果。

```
`rm cluster.txt`;
$cluster=0;
&seed(1,1);
$cluster++;
&seed(-5,-5);
```

```
$cluster++;
&seed(0,0);

#-------------------------------------
sub seed {

local($x,$y)=@_;

$kmax=rand(300);
$x=rand($x)-0.5;
$y=rand($y)-0.5;

#print "==$x == $y\n";

for ($k=0; $k<$kmax; $k++) {
  $x=$x+rand($1)-0.5;
  $y=$y+rand($1)-0.5;
  $px[$k]=$x;
  $py[$k]=$y;
}
open(OUT,">>cluster.txt");
for ($k=0; $k<$kmax; $k++) {
  print OUT "$cluster\t$px[$k]\t$py[$k]\n";
}
close(OUT);
}
```

# **Hadoop**和大数据的新型合成方差

本节总结了许多书中介绍的概念，包括Data Science Central的统计研究实验室得出的其他合成指标和应用数据科学定理。两个技术注释（你可以跳过）可以帮助懂数学的读者深入了解有趣的数据科学研究课题，包括矩阵理论（包括双重随机正交矩阵）、凸函数、$L_p$ 范数、不变式和高维几何等。这里以原始的方式引进和推广方差的概念。传统的方差（对应于$L_2$ 并在所有参考书上都有提及）是唯一在现有架构描述时无法保持尺度不变的量。尺度不变的方差是检测簇结构、孔和不均匀点分布极好的指标。

Andrew Peterson给出了很好的评论：Hadoop是先加载再组织，而数据库是先组织再加载（`http://linkd.in/1eH7bmb` ）。

# Hadoop和MapReduce的介绍

MapReduce是一个能够有效地处理大量数据的框架。把原始问题映射到较小的问题（其本身可能成为“原始”问题），然后并行处理这些较小的问题。较小问题的处理结果再进行合并或“缩减”。它适用于易于被并行化处理的过程，比如交易数据处理（评分）、网页信息爬取、处理日志数据（社交网络数据），甚至类似排序的任务（可通过一个融合步骤分解为排序数据子集），或操纵巨大但非常稀疏的矩阵，即使用精心设计的算法来利用数据的稀疏性。在处理许多数据点之间有重要的交互和依赖项的数据集时效率稍低，例如大量文本的聚类。即使在这种情况下，也可以在完全忽略占关键词关联99.9%以上的非常弱的关联后，在大数据集中使用MapReduce（如谷歌关键词索引的数据集）。当强烈的相互作用（短期和长期）存在时，如某些空间数据，我们最好另选图形数据库。

Hadoop是MapReduce的一个实现，用于分割和融合数据，并以分布式方式处理，它可以在单机上运行，针对处理数据太大内存放不下的问题（也可以是每次数据存储的尺寸小，但哈希表太大），或是在集群上运行。如果你先安装一个Linux环境，你甚至可以将它安装在一个Windows笔记本电脑上。尽管一些SQL层建于Hadoop之上，但它本身更像是一个文件管理系统而不是传统的数据库框架。可以在`http://bit.ly/1dQVjwf` 上找到更多关于Hadoop的资料。

# 综合指标

这里介绍的新的方差修正了传统方差在大数据应用中的两个问题，并介绍基于Hadoop的计算方式（使用一个数值不稳定的公式）。这个新指标是合成的：它不是从数学计算自然派生而来的，不同于任何统计学101课程里所讲授的方差或是目前在Hadoop中实现的方差。对于合成 ，我的意思是，它是为解决大数据（异常值）的问题和适合现在许多计算大数据的方式而建立的：MapReduce是一个框架，Hadoop是一个实现。这是一个不同的指标设计方法——从数据到理论，而非从理论到数据。这里的理论，我指的是统计模型，如可推导的度量（统计人员称之为统

计量）。例如，在高斯分布的假设下求解最大似然方程。

这些新的度量可以像用于商业目的的标准指标一样使用。在大多数时候，它们实际可以作为更通用化的指标，因此可以应用于更多的问题中，而标准指标的用处则相对有限（例如，如果用标准指标，你必须假设数据符合高斯分布，或没有太多的异常值）。

在我们实验室中设计的其他合成指标还有如下几种。

- 预测能力指标，它与熵有关（也就是信息的量化指标），通常被用于大数据框架，例如为评分算法确定最优特征组合。（详情请看第6章“数据科学应用案例研究”。）
- 大数据的相关性：由一个算法定义且与这里讨论的最佳方差指标密切相关。（详情请看第4章：“数据科学的技术（I）”）
- 结构性系数：详见第4章。
- 颠簸系数：详见第4章。

# Hadoop、数值的和统计的稳定性

在Hadoop中用于计算方差的公式有两个问题。首先，使用的公式，即$\mathrm{Var}(x_1,\ldots,x_n)=\{\mathrm{SUM}(x_i^2)/n\}-\{\mathrm{SUM}(x_i)/n\}^2$，是出了名的不稳定。对于大值n，虽然公式内两个部分抵消了一些，但单独拿出来任何一个都可以是一个巨大的数值，因为这是数十亿观察值的平方级的运算。它导致了数值不准确，甚至负的方差的出现。欲知详情，请看本人文章《大数据的诅咒》的评论：`http://bit.ly/104jiqU` 。此外，还有一些变形公式不需要对整个数据集进行两轮运算，数据集数值上是稳定的。

当然，目前使用的公式在MapReduce环境下（如Hadoop）是非常容易实现的。首先把你的数据分散到200个桶中。在200个数据桶中分别计算两个求和（每个计算都是在一个单独的服务器上同时进行的，以此减少200倍计算时间），然后进行200个数据的求和并最终计算一个平方就可以得到正确的结果了。

我主要担心的不是数值不稳定，而是在大型数据集，异常值是一个

大问题，它会扰乱你的方差。这个方差公式将跨数据桶的方差比较变得毫无意义并出现很大的数据波动，如果数据用于预测建模，还会导致预测不准确。解决方案是使用一个对异常值不敏感且易于用Hadoop计算的不同类型的方差。

# 方差的抽象概念

在引进一个新的方差定义之前，让我们首先讨论方差指标应满足什么条件。以下是根据其重要性排列的有用属性：#1、#2和#3是最基本的，而#5、#6和#7则可有可无。传统方差可以满足属性#4的要求，但当放松它的条件后，可以引出一些新的有趣的方差类型（本节稍后会讨论到）。

1. 方差是正数。当且仅当$x_1 =…=x_n$ 时，方差值为0。

2. 方差（用V表示）是对称的。任意排列的（$x_1$ ,…, $x_n$ ）具有相同的方差V（$x_1$ ,…, $x_n$ ）。

3. 离$x_1 =…=x_n$ 越远，方差越大。

4. 方差具有平移不变性：对任意常数c，有V（$x_1$ +c, ..., $x_n$ +c）=V（$x_1$ , ..., $x_n$ ）。

5. 方差具有尺度不变性：对任意常数c，有V（$cx_1$ , ..., $cx_n$ ）=V（$x_1$ , ..., $x_n$ ）。首先看一下后面的技术注释。

6. 在大量的矩阵变换下，如转置或反转的情况下，方差不变。

7. 方差是有界的，例如，V（$x_1$ ,…, $x_n$ ）< 1。

属性#1和属性#2是强制型属性。属性#5是非必要的。从任意方差V，很容易得出一个尺度不变的方差W：定义W（（$x_1$ , ..., $x_n$ ）为V（$x_1$ , ..., $x_n$ ）除以V（$x_i$ , $x_j$ ）的均值，其中V（$x_i$ , $x_j$ ）是{$x_1$ , ..., $x_n$ }的所有子集{$x_i$ , $x_j$ } （i<j）的方差。属性#7是非必要的。使用W=V /

（1+V），你可以把任何无界方差V变成有界方差。无界方差的缺点是一个异常值可以使方差任意大。

属性#4是非必要的。如下，你可以从V推出一个平移不变的方差W：W（$x_1$ , ..., $x_n$ ）=max V（$x_1$ +c, ..., $x_n$ +c）,这个最大值是基于所有c计算得来的。仅当c使得V的最大值为定值时，无须考虑（$x_1$ , ..., $x_n$ ），此公式成立。

我想重新定义属性#3。这里描述的所有方差都有类似于#3的属性。但是严格来说，属性#3意味着（$x_1$ ，..., $x_n$ ）和n维空间中包含点（0,...,0）和（1,...,1）的线的距离d越大，方差越大。d是传统的用（$x_1$ ,…$x_n$ ）计算出来的标准差，因此，只有满足属性#3的方差才能成为传统方差的单调函数。

这里介绍的新的方差满足属性#1和#2。定义如下：假设g是任意的严格凸函数，例如g（x）=$|x|^a$ （其中a>1）。然后：

$$V(x_1, ..., x_n) = \{SUM(g(x_i))/n\} - g(SUM(x_i)/n)$$

如果你使用g（x）=$|x|^a$ （a=2），然后公式符合传统的方差定义。我建议用一个更低的值，例如a=1.25，来解决数值不稳定和异常值问题。我希望可以用a=1，但这样一来，g就不是严格凸函数了，所以行不通。有趣的是，当通过变换使得V（使用g（x）=$|x|^a$ 来定义）具有平移不变性时，我们得到一个新的方差W，就算a=1它仍然能很好地工作，同时使得计算速度大大加快：计算$|x|^{1.25}$ 比计算$|x|^1$ 需要更多的时间。

当g（x）=$|x|^a$ 并且1 < a < 2时，最大化V（$x_1$ +c, ..., $x_n$ +c）的c为c=−（$x_1$ +1...+ $x_n$ ）/n。我尚未证明这一结果，我邀请读者您给出一个正式的证明。但是我在几个数据集对其进行过测试。显然，当a=1或a=2时，结果不成立；如果（当且仅当）a=2时，方差V和它的平移不变的版本W相等。所以经过这样的转化后，我们最后得到的方差是：

$$W(x_1, ..., x_n) = SUM\{g(x_i - u)/n\}$$，其中u=（$x_1$ + ...+ $x_n$ ）/n，

g（x）=$|x|^a$ 并且a=1。

注意　　在La范数条件下，W满足属性#4（平移不变）。如果使用

之前描述的方法，把W变换成尺度不变的方差，最终将得到一个擅长检测聚类和有洞的数据结构的度量标准。当数据中有较强聚类模式存在时，结果指标较高；相反，当数据点分布更均匀时，指标值较低。这适用于1<a<2，以及a=1。这不适用于a=2（标准方差）的情况，因为此时产生的度量标准只取决于n：如果a=2，那么无论（$x_1$ , ..., $x_n$ ）的值是多少，其值都恒等于（n-1）/（2n）。所以，传统的方差（参数a在1到2之间）是唯一一个不能使用上面描述的方法来使其尺度不变的方差。

作为练习，在变换W（方差V的平移不变版本）使其尺度不变后，计算a=1时基于4个n=10数据点的数据集的方差。

- 第1个数据集是 {0, 1, 2, 3, 4, 5, 6, 7, 8, 9}。
- 第2个数据集是 {0, 0, 10/3, 9/2, 9/2, 9/2, 9/2, 9, 9, 9}。
- 第3个数据集是 {0, 0, 0, 0, 0, 9, 9, 9, 9, 9}。
- 第4个数据集是 {0, 0, 0, 0, 0, 1, 1, 1, 1, 1}。

这可以帮助你理解什么是尺度不变的方差。因为尺度不变的性质，所以基于第3和第4的数据集计算的值是完全相同的。就计算复杂度而言，最后我们的公式获得尺度不变的方差的计算复杂度是O（$n^2$ ），可见这个过程是非常耗时的。这里讨论的其他算法的计算复杂度都是O（n）。那怎么改善这个O（$n^2$ ）？

# 一个新的大数据定理

Data Science Central研究实验室时不时就提出一些新的定理，不管数据集大小，这些新的定理都可以更有效地处理数据。例如，当g是严格凸函数时，公式V（$x_1$ , ..., $x_n$ ）={SUM（g（$x_i$ ））/n} – g（SUM（$x_i$ ）/n）满足属性#1和属性#2，可用于计算方差，

另外，我称为“分析桥第四定理”的原理——是*Jensen* 不等式 的直接推论。感兴趣的读者可以了解一下我们之前的3个定理。

- 分析桥第一定理 ：用于构建模式无关的置信区间。（第5章）

- 分析桥第二定理　：用在随机排列问题。一个随机排列的非独立数据构成一个独立数据序列。（第4章）
- 分析桥第三定理　：用于证明一个新定义的相关性（在大数据背景下使用）的值在−1和1之间。2013年，由Jean-Francois　Puget在第一届数据科学竞赛中所证明。（第4章）

已知X=（$x_1$ , ..., $x_n$ ）和Y=（$y_1$ , ..., $y_n$ ）是两个向量，如果g（x）=$|x|^a$ 且a=2，那么Cov（X, Y）={V（X+Y）−[V（X）+V（Y）]}/2就定义了传统的协方差。一个潜在的研究课题是改进1<a<2情况下的协方差公式，并把这个定义推广到两个向量以上。

# 平移不变性的度量标准

很容易证明，当a=2时（传统方差），V满足属性#4，也就是平移不变性：给所有的数据点$x_1$ ,…, $x_n$ 加上一个常数不会改变V的值。当1<a<2时，这个结论不成立。例如，n=3时：

- 当a=2且$x_1$ =3、$x_2$ =5、$x_3$ =10，可得V=8.67。
- 当a=2且$x_1$ =10 003、$x_2$ =10 005，$x_3$ =10 010，可得V=8.67（unchanged）。
- 当a=1.25且$x_1$ =3、$x_2$ =5、$x_3$ =10，可得V=0.34。
- 当a=1.25和$x_1$ =10 003、$x_2$ =10 005、$x_3$ =10 010，可得V=0.00。

这是我不喜欢平移不变性的原因之一，与传统方差a=2相比，我更喜欢a=1.25。

可以很容易地证明，要满足平移不变性质，函数g必须满足g（x）−2g（x+1）+g（x+2）=常数。唯一满足这个约束的g函数是二次型（例如当a=2时），并且因此与传统的方差连接起来。同样，为了满足尺度不变性，函数g必须满足方程g（2x）−2 g（x）=常数。没有严格的凸函数能够满足这个约束，所以我们没有尺度不变的V。但是你也可以用之前描述的转换算法来建立尺度不变的方差W。

注意　a=2定义的传统方差在正交变换（反射或者旋转）时具有保证均值不变的属性——也就是说，在正交且随机的线性运算下，传统方差具有不变性。用矩阵表示，上述线性运算即为一个n x n的矩阵A乘以列向量X=（$x_1$ ,…$x_n$ ）'，并且A的每一列的和等于1。这里用'表示向量的转置（从行到列）。简而言之，如果A是一个这样的矩阵且a=2，则V（AX）=V（X）。

一个很有趣的情况是，当A同时是双随机和正交矩阵时，则A是置换矩阵。使用以下迭代算法，即可生成矩阵A：从任何双随机矩阵$A_0$ 开始；对于k>0，$A_{k+1} = (A_k - s\,I)(I - s\,A_k)^{-1}$。这里我们假设0 < s < 1时收敛，符号I表示n x n的单位矩阵。

# 实现：通信和计算成本

Hadoop和MapReduce的一个问题是，当将一个任务分配到在各自服务器上运行的并行流程时，在数据传输上所花费的时间。它把一个计算复杂度的问题从算法复杂度转移到通信或带宽成本上。我指的通信是，从一个集中的服务器（数据库或文件管理系统）或分布式系统内存或硬盘移动数据所需的时间。如何优化这些数据传输？如何优化内存计算？如果不正确实现，你不能依赖一个内置函数，如方差，如这里所示。

实际上，方差计算是一个有趣的问题，说明了一些Hadoop面临的技术挑战。如果没有有效的内置算法可用，有必要研究自制的计算，这种计算可以利用MapReduce的架构，同时使局域网内的数据流动尽可能少（尤其不会一次性下载所有数据，用以计算均值，然后在计算方差时再次重复加载）。已知一个用于Hadoop的高度不稳定的方差公式（a=2），其中一个解决方案是通过在计算之前删除一个较大的$x_i$ 的值来使其稳定。毕竟，a=2时，方差是平移不变的，那么我们为什么不利用这个属性呢？使用a=1.25比异常值更可靠。或者更好的是用平移不变的W，且a=1。但是这时，你必须自己写代码，或者使用某些函数库。要找到更好的解决方案，详见 http://webmail.cs.yale.edu/publications/techreports/tr222.pdf 。

# 最终意见

以上大部分的讨论集中在一个特定参数a上，其潜在值处于1和2之间。在贝叶斯架构中，我们可以为这个参数分配一个先验分布。

最后，这形成一个有趣的统计学教学方法：从讲述方差　（最具有可视性的度量）开始，然后转到介绍协方差　，最后用转换的非独立性概念介绍均值　。同时可以看出这样的方法不适用于解释传统方差理论（a=2）。它仅适用于a<2的情况。

# 总结

本章展示的内容没有过多关注度量标准，而是更专注于应用。内容包括：关于如何创建一个数据字典的讨论（数据探索的第一步）、构造隐性决策树（一个由我创立的健壮的数据科学技术）、在NoSQL数据库背景下的哈希连接，以及分析桥第一定理——为无须统计理论和知识就能计算置信区间提供了一个简单的、模式无关的、非参数的方法。最后一节主要介绍的是为Hadoop设计的新式合成方差，帮助解决数值（不稳定）和统计（异常值敏感性）的问题。

第6章主要涉及案例研究、实际的应用程序和各类项目的成功故事，从基于数据科学的股票市场技术到广告组合优化、欺诈和侵犯版权检测、搜索引擎优化（搜索公司）、天文学（预测陨石撞击）、自动化新闻管理、电子邮件营销和相关性问题（在线广告），一应俱全。

# 第6章
# 数据科学应用案例研究

第5章讨论的材料，应该成为你数据科学训练的一部分。这些材料不太注重度量标准，而是更注重应用。本章讨论案例研究、现实生活中的应用和一些成功范例。其中涵盖了各种各样的项目，包括基于数据科学的股票买卖技术、广告组合优化、欺诈检测、搜索引擎优化、天文学、自动化新闻反馈管理、数据加密、电子邮件营销，以及（在线广告）相关度（Relevancy）问题。

## 股票市场

以下展示了一个基于数据科学的简单策略，我最近一次使用它是在2013年，用于选择和交易标普500的股票，带来了持续不断的高回报。本节还讨论了其他策略、新潮趋势和API——可用来给技术分析型的专业交易员提供股票交易信号。

## 使回报率提高500%的模式

这种模式是最近在500只股票价格波动里发现的，这些股票是标准普尔500指数的一部分。该模式带来了5倍的回报。从4月24日到5月23日，通过搜索和比较每一天与昨天的收盘价，找出这500家公司中股价回报最极端的一些公司。

接下来（通过比较每天的收盘价）看排名第1到第500的公司每天的回报表现。排名第1的公司，（平均而言）在第二天股价也会上升。对

于股价上涨7.5%（或以上）的公司，次日股价的增长幅度更大。这种次日回报提升在统计意义上相当显著。事实上，与标准普尔500指数4.5%的整体回报率相比，把每日回报排名及次日股价当作预测信号，30天（非复合的）总回报率高达20%。

以下3个特点使它更具有统计学意义。

- 它发生在整个时间段（不只是在仅仅几天中）。
- 它没有受到异常数据的影响（比如某天一只股票有巨幅回报，而其他时间都是小损失）。
- 它适用于很多不同的公司（而不只是某三四家公司）。

这些公司的次日回报里20次中有15次是正数。

## 股票交易的建议和注意事项

这些数字是在短时间段内计算的，且发生在增长期（牛市）。但是，还有以下两点应当注意。

- 这些模式从未持续超过几周，因为它们会被多个交易者检测到，然后就不复存在了。
- 交易算法利用颠簸系数检测到的模式获利，并在模式的增长阶段、下降阶段、平缓且稳定阶段、平缓但不稳定阶段，持续得利。

有时，当模式由于过度使用而不再奏效时，则需要扭转策略（例如，一周5天都是坏结果，那么就要交换买入和卖出），这样可以使你继续享受几个星期更好的回报，直到你需要再次逆转策略，以创造连续的、周期的策略或反向策略循环。事实上，当你读这本书时，这种模式可能就不会再有效了，因为每个人都会尝试利用它。但也许反向策略会有效！

你可以考虑在更小的时间窗口内买入或卖出，以进一步增加收益：比如盘中，甚至是高频买卖，而不是每天一次的买/卖周期。然而，较小时间窗口的回报通常较少，并且利润会更容易因为以下因素耗散。

- 买卖（交易）收费。
- 差价：买入价与卖出价之间的差额（对于流动性好的股票，像标普500，价差通常是小的、合理的）。
- 税收。
- 交易失误。
- 用于预测的数据出错。
- 数值逼近。

注意，专注于标普500，可以避过大部分股票市场的波动。还可以专注于流动性高的股票，这样能够降低你成为股价操控受害者的风险，且能够保证你的买卖交易可以拿到想要的价格。你可以专注于价格10美元以上的股票，从而使投资处在更稳健、可预测的环境中。你还可以专注于每个业务类别的极端日收益，看是否能增加回报。标准普尔500数据可分解成大约10个主要类别。

在数据中引入一些干扰信息，做一些模拟以检查你的算法对价格变化的敏感度，这也是一个不错的主意。此外，你不应该仅仅依靠回测，还应该向前来检查策略是否在未来也有效。这等于在使用策略之前执行明智的交叉验证。

## 如何获取股票价格数据

你可以在`http://www.stockhistoricaldata.com/daily-download` 这个网站上找到标普500每只股票每月每天的价格。你需要提供这500只股票的列表。每个标准普尔500公司的股票代码和业务类别，可以在`http://api.ning.com/files/4*5ByqO6p6dnq6rxVCe60W1NrodvAiWSr1y7wI4P9Y2umALk-uIKb1b0OZaSx2wvOjhT6T60uoU-jfY7UjoCY*xYVvypG/sp500.xlsx` 上找到，在维基百科中也可以找到。

另外需要注意，你能向股票交易者出售股票价格预测结果吗？要如何给这些预测结果定价？当预测失败时，你会如何处理不开心的客户的强烈退款要求？

结论是，除了单纯的统计知识，你还需要深厚的专业领域知识，来为精明的客户设计专业交易信号。在这种情况下，处理客户的问题（而不是开发纯粹的理论模型），会引入潜在的责任和诉讼风险，如果你向没有交易经验的人出售交易信号则更容易如此。

# 优化统计交易策略

优化统计交易策略时常见的错误之一，就是将问题过度参数化并计算全局最优解。众所周知，这种错误做法在历史数据上有极高的回报，但在实践中却行不通。本节将会探讨这个问题，帮助你了解如何规避它。此外还将介绍如何建立一个有效的只用到6个参数的策略。

这个错误实际上涉及许多统计学和数学，又可以称为过度参数化或过度拟合。可以通过一个简单的例子说明这种方法为什么会失败。想象一下你用包含30个参数的模型拟合数据。如果有30个数据点（即参数的数目等于观察值的数量），那么你可以得到一个对数据集完美且完全优化的拟合。然而，任何未来的数据点（如明天的股票价格）都可能与模型不相匹配，从而导致巨大的损失。为什么呢？因为你的参数与数据点数量相同。因此，每个模型的估计参数值，最多对应一个数据点。

从统计的角度来看，这就好像要估计全美国人工资的中位数，但只调查了一个人。即使每个样本一个参数的拟合是完美的，但模型估计的结果很有可能会很差。实际上，50％的概率是，受访者的工资要么非常低，要么非常高。

大致来说，这就是把模型过度参数化后产生的后果。你显然会得益于参数数量的减少。然而，如果处理得当，这个参数过多的缺点实际上可以变成优点。事实上，你可以建立一个多参数的模型，（在回报率方面）这样的模型比参数更少的简单模型有更稳健和更高效的预测能力。这该如何实现？答案在于你测试策略的方式。当你使用的模型包含3个参数以上时，在历史数据上产生最高回报的策略并非是最好的。你需要使用更复杂的优化标准。

一个解决方案是给问题增加边界条件，从而进行约束优化。寻找至少满足一个基本约束条件——可靠性——的策略。也就是说，你想排除所有对细小变化过于敏感的策略。因此，你把重点放在了小部分对各种干扰信息都健壮的参数空间上。在该情况下，干扰信息可以是在历史股票价格或在参数集上的交易误差、差额或细小的变化。

从实践的角度来看，该解决方案关键在于尝试在许多不同市场条件下都有效的数百万个策略。通常，它依赖于几个月的数据，以包含不同

的市场模式和一些统计显著性。然后，对于每一种策略，必须以数百万种不同的方式引入干扰信息，并观察其影响。随之舍弃所有被干扰信息严重影响的策略，只保留极小部分稳健的策略。

这计算起来相当复杂，因为这等同于测试数百万的策略，但这是值得的。最终会得到一个可靠的策略，它可以通过稍微改变参数，做到随时间调整。我自己的策略其实就是这样设计的。它们最多与6个参数相关联，如下。

- 4个参数用来追踪股票是如何移动的（上涨、中性或下跌）。
- 1个参数用来设置买入价。
- 1个参数用来设置卖出价。

本来可能是通过在参数中强加对称性（例如，让买入和卖出价的参数相同），来减少问题的维数，而这里的方法相反，结合了低维度（也就是可靠性）和高回报的优点，而且回报明显高于平常的保守估计。

最后，当回测交易系统时，要使用1个月前的历史数据来优化策略。然后检查上个月获得的真实回报（不包含在回测历史数据时间窗口里）是否令人满意。如果你的系统通过了这个测试，那么可以使用最新的数据来优化并使用该策略。否则，不要在现实情形下使用你的交易系统。

## 在短期策略基础上提高长期回报

本节将介绍如何回测短期策略，以评估其长期收益率的分布。本节侧重于需要经常更新的策略，即自适应策略。如果遇到长期振荡且平均回报为零这种不受期望的长期特性，则可以利用回报函数的周期性特性，设计一个真正赢利的系统。

当一个策略依赖于参数，需要经常更新时，你必须设计出合适的回测工具。我建议将参数的数目限制为6，并使用稳健的统计方法和约束优化。本节你将学会如何改进回测技术。为简单起见，假设被测系统提供每日信号，并需要每月更新。测试这种自适应系统的正确方法是根据历史数据，每个月回测一次，如以下算法所示。

测试期内的每个月，需要完成以下步骤。

步骤 1：回测。收集过去6个月的历史数据。回测这6个月的系统，以估计模型参数。

步骤 2：向前推进。把从步骤1获得的参数应用于当月的交易系统。计算每天的收益。

整个测试周期至少要18个月。因此，你需要收集和处理24个月的历史数据（18个月，再加额外6个月的回测）。在步骤2中（每个月1次），依次得到样本外（OUT OF SAMPLE）每月回报，记录下来，以做进一步调查。你可能会注意到以下模式。

- 有许多月份的收益表现很不错。
- 也有许多月份的收益表现很糟糕。
- 平均来说，回报为零。
- 收益好的月份经常紧接收益好的月份。
- 收益差的月份经常紧接收益差的月份。

现在，要建立一个长期可靠的系统所需的元素已经俱备了。因为建立在原有系统基础上，所以它是一个元策略系统，且其工作原理为：如果上个月的回报呈正数，则本月使用跟上月相同的策略；否则利用反向策略，即交换买入和卖出价格。

# 股票交易的**API**：统计模型

本节详细描述了一个基于预测模型的股票价格信号（买入/卖出信号）系统，这个系统曾经是通过API提供给付费客户的。简而言之，这个网络应用程序是一个可实时获取股票买入/卖出信号的股票预测平台。本节首先考虑预测模型，然后在下一节考虑（通过API）基于互联网的实现。

这个应用程序依赖于一个原始的、提供每日指数和股票趋势信号的系统。这里描述的非参数统计技术有以下几个优点。

- 简单： 不涉及高等数学，只有基本的代数。算法无须复杂的编程

技术，只需要依靠容易得到的数据。

- 高效： 在测试中，每日预测的正确率高达60%。
- 方便： 非参数系统无须参数估计。它可以自动适应新的市场条件。此外，该算法对计算的要求较低，即使在运转缓慢的机器上也能快速提供预测。
- 普适性： 只要没有重大事件影响价格，该系统可以在任何特定的时间支持交易量大的股票和指数。同样的算法适用于所有股票和指数。

## 算法

对于一只特定的股票或指数，该算法计算明天的收盘价将比明天的开盘价高某个特定百分比的概率。该算法也容易修改为比较今天的收盘价与明天的收盘价。所估计的概率，是根据过去最多100天股票（或指数）的历史数据来进行估计的。

第一步是选择一些平均值为1的价格交叉比率。比率中的变量可以用来优化预测。在一个应用程序中，选择了以下3个交叉比率。

- 比例A=（今日最高价/今日最低价）/（昨日最高价/昨日最低价）
- 比例B=（今日收盘价/今日开盘价）/（昨日收盘价/昨日开盘价）
- 比例C=（今日成交量/昨日成交量）

随后，历史数据集中的每一天都会被分配到8个可能的价格配置中。配置的定义如下。

- 比例A>1，比例B>1，比例C>1。
- 比例A>1，比例B>1，比例C≤1。
- 比例A>1，比例B≤1，比例C>1。
- 比例A>1，比例B≤1，比例C≤1。
- 比例A≤1，比例B>1，比例C>1。
- 比例A≤1，比例B>1，比例C≤1。
- 比例A≤1，比例B≤1，比例C>1。
- 比例A≤1，比例B≤1，比例C≤1。

现在，为了计算出明天收盘价比开盘价至少高1.25%的概率，首先

需要计算今天的价格配置；然后，检查在历史数据集中满足上述配置的所有日期，并计算天数。用N代替这些天数，然后，再用M来代替满足下述情况的天数：

次日的收盘价至少比开盘价高1.25%。

你要计算的概率仅仅是M/N，即次日收盘价至少比开盘价高1.25%的概率（基于过去数据得出）。当然，1.25这个数字可以用任意百分比替换。

## 效果评估

要评估股票走势预测指标的效果，有几种不同的方式。以下是其中两种方法。

- 使用一个临界值0%，而不是1.25%，在至少200个交易日以上的一段时间中，计算每日预测成功的比例。
- 对于开盘时买入，收盘时卖出这个策略，使用预测到的趋势（设置0%的临界值）；也可以根据预测使用其他策略。

测试表明，在预测纳斯达克指数趋势方面，上述算法的成功率在54%和65%之间。当预测趋势有56％的成功率，在许多情况下，长期（非复合）年回报率在40％以上。像许多交易策略那样，系统回报有时会出现振荡。

使用前述的元策略，很有可能可以大幅减轻这些振荡。最简单的方法是，如果今天它有效，明天继续使用；如果系统未能正确预测今天的趋势，则明天使用逆向策略。

更多优化模型的通用流程包括如下几项。

- 通过检测和排除那些价格或成交量等低过平均水平的股票或者低价股票，来自动优化投资组合。
- 使用指数，而不是股票。
- 检测应避免交易的时间段。
- 将事件纳入你的模型中，如企业季度报告、工作报告和其他经济指

标。
- 计算任何模式下的每只股票的强度值，强度值即置信区间。强烈的信号意味着可靠，如果你是一位统计学家，强信号意味着小置信区间（可以提高你预测的准确度）。

# 股票交易的**API**：具体实现

该系统作为一个网络应用程序（网络表单），在线提供给付费客户，也可实时通过API自动访问。当用户访问网络表单时，必须提供一个股票符号、一个键（Key）以及一些参数，举例如下。

- 只显示强烈的信号：如果信号不是足够强（假定用户坚持只要统计显著信号），那么买入/卖出信号将被N/A替换。其缺点是用户交易将变得不频繁，回报可能会减少，但也限制了损失。
- 用于产生买入/卖出信号的历史数据的天数上限为180。

键是用来验证用户是否按时付款的。活跃的键意味着用户有权运行该网络应用程序。该Web应用程序返回明天是买入还是卖出信号的网页，以及如下几项信息。

- 所关注股票的历史每日价格。
- 过去90天所关注的股票每日买入/卖出的预测信号。
- 过去90天所关注的股票预测器的每日表现。

当用户点击提交时，Web服务器会运行一个Perl脚本（如`GetSignals.pl`）以快速计算历史数据（最多过去180天的股价）并显示结果。每次调用计算指令所需时间均小于0.5秒。另外，机器可通过URL调用应用程序（API则通过网络机器人调用），如`www.DataShaping.com/GetSignals.pl?stock=GE&key=95396745&strength=weak&history=60`。

结果保存在网页上，可以立即查看（实时、动态创建），且可以通过网络机器人（机器对机器通信）自动查看，如`www.DataShaping.com/Results-GE-95396745-weak-60.txt`。

### 技术细节

Web应用程序由以下部分组成。

- 嵌入到`GetSignals.pl` 里的统计模型，详见前一节。
- 嵌入到`GetSignals.pl` （Perl脚本）里的HTTP请求：为了从雅虎财经获得过去180天的股票历史数据。（所请求的数据是免费提供的。另外，也可以使用API调用——这里指的是雅虎——来自动访问，然后解析所下载的页面。）你可以在第5章的优化Web爬虫一节，查看自动HTTP请求的示例代码。
- 高速缓存系统：　今天请求过的股票历史数据会存储在服务器上，以避免同一天中多次调用雅虎财经数据。
- 避免冲突：　根据每个键（即每个客户端）和每一组参数每天生成不同的文本文件输出。
- 用户或计算机界面：　指通过用户或一台机器来访问应用程序（网络表单或API）的外部层。

无须使用数据库系统，但可以说使用了NoSQL系统——实际上就是一个文件管理系统，可以通过HTTP请求（若不在缓存中）或文件查找（若在缓存中）实时访问股票价格。

# 股票市场模拟

虽然可以随时查看股票市场数据，但是自己拥有模拟器还是非常有益的，尤其是当想自己测试当日策略或模拟股市涨跌的时候。

这里所使用的统计过程是具有无限状态空间的马尔可夫链，以模仿股票价格的对数（记为y）。在每次迭代中，y都增加一个数量x（可能为正数也可能为负数）。以下源代码中，对转换采用如下的表达。

- x=pval[0]与概率ptab[0]。
- x=pval[1]与概率ptab[1]。
- x=pval[6]与概率ptab[6]。

根据pval和ptab中的值，潜在的过程可能会显示出增长的趋势、下降的趋势或者是静止不动的状态。此外，还应增加某些干扰信息使得该过程更加切合实际。模拟数据存储在datashap.txt 中。

为了方便，假设唯一可能的转换是–1、0或+1。

- 如果ptab[2]=ptab[4]，则为无上升或下降的趋势。
- 如果ptab[4] > ptab[2]，则市场呈上升的趋势。
- 如果ptab[4] < ptab[2]，则市场呈下降的趋势。
- 如果ptab[3]是最大值，则市场呈缓慢移动趋势。

转换概率实际上与ptab值成比例。另外，请注意，为了模拟真实的市场情况，即长期的整体上涨趋势（日收益率+0.02％），在源代码中，ptab[4]要比ptab[2]稍大。

你可以在http://www.analyticbridge.com/group/codesnippets/forum/topics/scode-to-produce-realistic-stock-market-simulations 中找到对完全中性市场的模拟。X轴代表以美元为单位的股票价格；Y轴代表时间，基本的时间单位为1天。结果出人意料，且恰恰与直觉相反，居然没有出现增长——如果运行这个实验时间足够长，则会导致结果无限次在起点上下波动。这样的无记忆过程称为随机游走。基于本节前面介绍过的相同符号，下面的代码可用来模拟这种类型的随机游走。

```
#include <stdlib.h>
#include <stdio.h>
#include <time.h>
#include <math.h>

#define N 7

int idx, l;
long k, niter=50000L;
double u, aux;
double ptab[N];
double pval[N];
double aux, p_distrib, x, y=0;
FILE *out;

int main(void)
{
```

```
ptab[0]=0.0;
ptab[1]=0.0;
ptab[2]=0.5;
ptab[3]=0.5;
ptab[4]=0.51;
ptab[5]=0.0;
ptab[6]=0.0;

for (l=0; l<N; l++) { aux+=ptab[l]; }
for (l=0; l<N; l++) { ptab[l]/=aux; }

pval[0]=-3;
pval[1]=-2;
pval[2]=-1;
pval[3]=0;
pval[4]=1;
pval[5]=2;
pval[6]=3;

randomize();
out=fopen("datashap.txt","wt");
fprintf(out,"simulated stock prices - www.datashaping.com \n");

for (k=0L; k<niter; k++) {
        u=((double)rand())/RAND_MAX;
        idx=0;
        p_distrib=ptab[0];
        while (u>p_distrib) {
               idx++;
               p_distrib+=ptab[idx];
        }
        x=pval[idx];
        u=(double)rand()/RAND_MAX;
        x+=0.5*(u-0.5);
        y=y+x;
        fprintf(out,"%lf\n", exp(y/50.0));

}
fclose(out);
return 0;
}
```

# 些许数学知识

对于那些对数学更感兴趣的人，本节提出了一些更深入的研究结果，但没有深入到技术细节。你可以在这里了解到一些与华尔街相关的数学概念。在第2章“大数据的独特性”中的“大数据‘诅咒’”一节，也讨论了股票交易的统计数据。

以下理论与股票市场和交易策略有关，根源涉及鞅理论、随机游走过程、博弈论和神经网络。你会看到一些使股民们好奇心大发的最令人惊讶且深入的数学结果。

## 洛伦兹曲线

假设你有90%的交易利润来自5%的成功交易，则可表示为h（0.05）=0.90。函数h被称为洛伦兹曲线。如果利润由一个带有cdf F的随机变量X和期望值E[X]表示，那么：

$$h(x) = \left(\int_{[0,x]} F^{-1}(v)\,dv\right)/E[X],\ 0 \le x \le 1$$

为了避免在几个交易中获得太多集中的利润，要避免使用太尖锐洛伦兹曲线的策略。这个概念也同样适用于损失的情况。相关的关键词有库存管理、六西格玛、基尼指数、帕累托分布 和极值理论 。

## **Black-Scholes**期权定价理论

Black-Scholes公式涉及5个跟期权价格相关的输入：过期时间、成交价、标的证券、标的证券的隐含波动率和无风险利率。技术细节可以在`www.hoadley.net` 上核查。你还可以阅读Sidney Resnik所著的*A Probability Path*（Birkhauser，1998）。

该公式由布朗运动的理论衍生。它依赖于一个事实，即股市价格可以通过几何布朗过程进行建模。该模型假定过程的方差不随时间而变化，且该漂移是一个时间的线性函数。然而，这两项假设在实践中都可能失效。

## 判别分析

判别分析方法可用来优化选股策略。根据多年的历史股票价格，在任何特定的时间里，所有股票可被分为3类——坏的、中性的、好的。分类规则必须与特定的交易策略相关联，比如在今天收盘时买入，并在

7天后收盘时卖出。

## 广义极值理论

每日最高价/最低价比值的参数分布是怎样的？52周的最高价/最低价比值参数分布呢？另外，要如何估计某一特定股票的参数？股价时间序列之间的互相依赖性使我们很难计算出确切的理论分布。上述比值如图6-1所示。

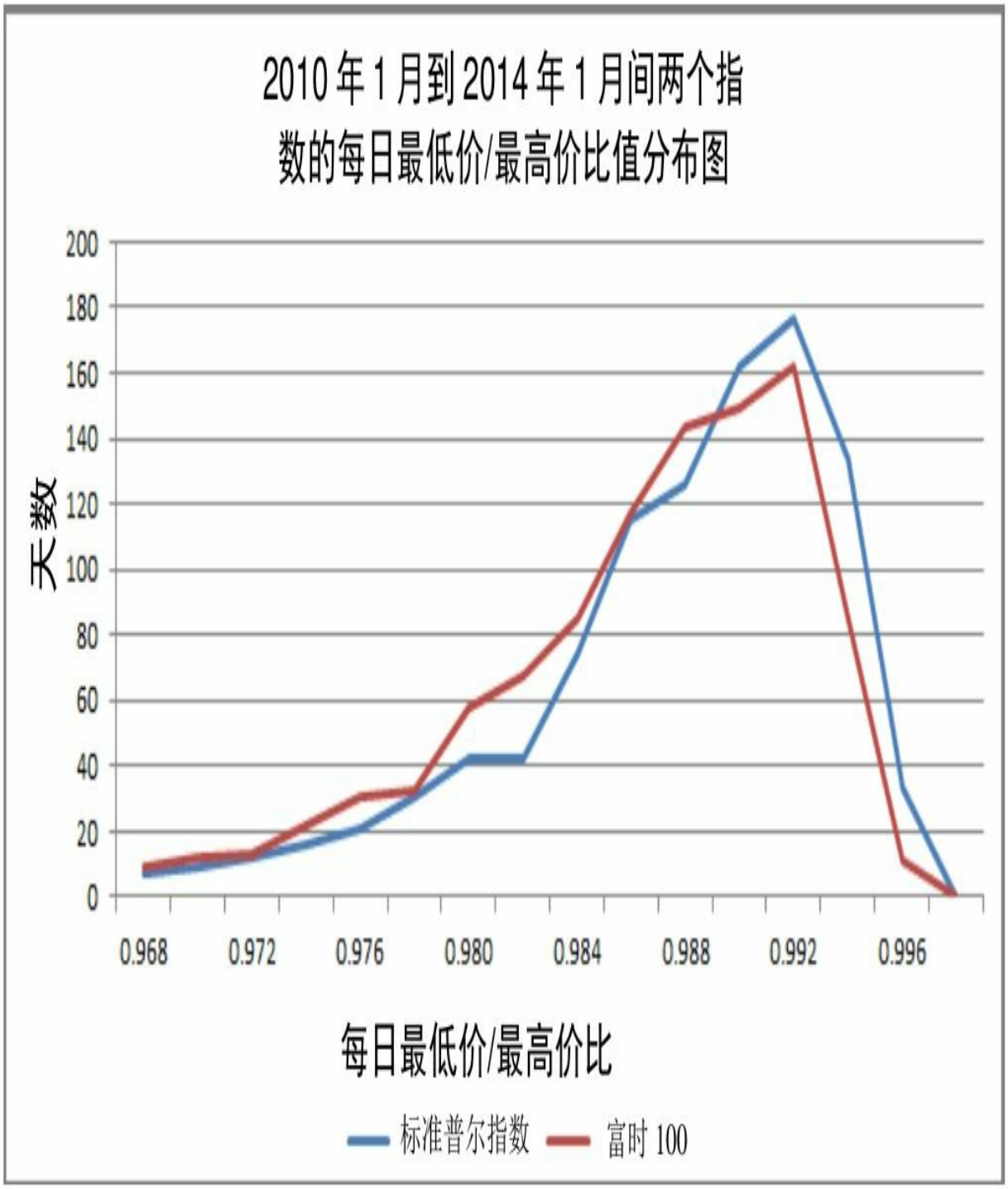


图6-1　道琼斯与纳斯达克每日指数比值

其分布的特点由两个参数来表征：众数（mode）和四分位距（interquartile）。虽然两个分布平均值相同，但富时100（国际指数）的左侧分布要比纳斯达克的更重，这使得它对日间交易者更具吸引力。从经验来看，一个左侧是肥尾分布的指数，因为具有更高的波动性和回

报而广受好评，但往往风险也更高。

### 随机游走和**Wald**恒等式

对于Z（正、负整数）的随机游走，转移概率是P（k为k + 1）=p, P（k）=q, P （k为k-1）=r, p + q + r=1，如果p小于r，要移动到超越任何给定开始点，期望步骤数是无穷的。否则它就等于1/（p–r）。

将这一结果应用到股票市场，意味着在固定条件（p=r）下，买入并持有的投资股票策略可能无法赢利，甚至在经过极长的时间后也是如此。

### 反正弦定律

该结果解释了为什么有50%的人一直在赔钱，而另外50%的人始终在赢利。比较一下股票交易和投掷硬币（反面=损失，正面=获利）。那么：

- 在一系列投掷硬币的计数中，正面的次数超过反面的次数，用中心极限定理可以估计这一概率，投掷次数越多概率越接近1。
- 根据长久领先定律，更确切地说是反正弦定律，在掷硬币的游戏中，往往存在一个令人惊讶的结果——很大比例的抽样结果显示其中一个玩家几乎一路领先游戏，只有在极少数情况下两者才会交替领先，而后者才是我们事先所预期的掷硬币游戏的结果。
- 关于随机游走的几何解释，游走路径很少跨越X轴，且随着游走时间的增长，跨越的频率逐步下降，而波形曲线的长度在轴的一侧不断增加。

# 新趋势

现代交易策略包括：爬取Twitter帖子来提取细分行业的市场情绪、事件建模和分类信息（基于关键词统计模型分析公告、季度报告、分析报告、工作报告、经济新闻的影响）以及检测以下模式。

- 轧空：　一只股票以极快的速度突然大量做空，有时成交量不高，但会快速恢复。如果错过了下跌，那么你依然会在上升中赚到钱。
- 时间滞后的交叉相关性：　例如，“如果今天谷歌股价上升，那么明天Facebook股价将下降。”
- 市场时机：　如果亚洲市场在早上呈上升趋势，那么纽约证券交易所将在几个小时后呈上升趋势。
- 余震（好比地震后）：　在经历剧烈事件（雷曼兄弟公司破产后，股市迅速崩盘）后的几日内，股市会经历剧烈的上下波动。提前模拟这些波动是可以做到的，并会在随后几天内获得可观的投资回报。事实上，有一类策略是，持币观望，在剧烈事件发生后的几天内进行大量交易，获得巨大回报，随后再休眠几年。
- 时间序列技术：　可观测宏观趋势（对改变点、股市价格的斜率变化进行观测）。

这些模式都运用了数据科学——更确切地说，是模式检测——这是数据科学家们都应熟练掌握的技术。这里的关键信息是，现实世界处于一个几乎持续的变化状态。变化可能或缓慢，或渐进，或突然。数据科学家们应该具备检测重大变化发生的能力，并根据实际调整他们的技术。

# 加密

加密也跟数据科学有关，体现在两个方面：它需要大量的统计专业知识，有助于使数据传输更为安全。第4章“数据科学的技术（I）”中讨论了信用卡编码，其中谈到了JavaScript代码的特定编码技术，因为给编码数字添加了随机模糊，所以就算是同一条消息也从来不会出现完全相同的编码。本节讨论现代隐写术：在多媒体文件、电子邮件加密和验证码技术中隐藏信息的艺术和科学。隐写术的讨论可以帮助你熟悉图像处理技术。

# 数据科学应用：隐写术

隐写术涉及数据加密和安全。想象一下，你需要通过互联网发送一项专利的细节或发起机密的金融交易，那么需要考虑以下3个关键问题。

- 确保消息不会被第三方获取和解密。
- 确保第三方无法确定是谁发送的消息。
- 确保第三方无法识别收件人。

消息加密属于第一步，但它可能无法保证高安全性。隐写术主要涉及运用某些机制来隐藏以图像、视频、可执行文件或其他格式存在的机密信息（如合同这类扫描文件）。如果与加密技术结合，这会是一个在不引起怀疑的情况下传送机密或保密文件的有效途径。

现在，我们来考虑这样一项统计技术，利用24位的图像（如Windows BMP图像这样的位图）来实现这一目标。隐写术在低位数字图像里隐藏信息，这种方式已经沿用了30多年。而更为先进的统计技术，应该使得隐写分析（对隐写进行逆向工程）更加困难，也就是说，对于用户更安全。

虽然这里专注于应用广泛的BMP图像格式，但你也可以使用拥有其他无损图像格式的技术。只要信息损失最小，它甚至可以和压缩图像一起运行。

## 一些逆向工程科学的知识

微软创建的BMP图像格式是用于隐写术的最佳格式之一。该格式是开源的，公共源代码可用于生成BMP图像。尽管存在如此多的变量和参数，但对这种格式进行逆向工程，可能要比花几个小时读上百页的文档来了解它是如何工作的更加容易。简单来说，这种24位格式是最容易操作的：它由一个54位的文件头以及后面的位图所组成。每个像素有4个组成部分：RGB（红色、绿色和蓝色通道）值以及alpha通道（可以忽略）。因此，存储每个像素需要4字节。

转到`http://www.siggraph.org/education/materials/HyperVis/asp_data/c`查看关于24位BMP图像的详细C语言代码。要想对这种格式进行逆向工程，其中一种方法是，生成一个空白图像，添加一个像素（如紫色），

它由50％的红色、50％的蓝色和0％的绿色组成。先改变像素的颜色，再改变像素的位置，来观察BMP二进制编码是如何变化的。用这种方法，此代码的作者计算出256色BMP格式（也被称为8位BMP格式）是如何工作的。而24位不仅更易于理解，而且用于隐写术也更加灵活、更加有用。

要隐藏秘密代码、图像或信息，首先要选择一个原始（目标）图像。有一些原始图像是隐写术的最佳选择；有一些效果则不好，甚至可能导致资料泄露。应该避免选择颜色昏暗，或存在均匀区域的图像。反之，颜色丰富且没有均匀区域的图像是合适的选择。所以，一个好的隐写算法，首先要考虑的就是图像选择的机制，好判断图像是否适合隐藏秘密消息。

## 基于一对多字母编码的新技术

现在，你发现了一个适合隐藏信息的图像，接下来就是操作的过程了。假设你想隐藏的信息来自80个字符集里的字母（26个小写字母、26个大写字母、10个数字和一些特殊字符，如圆括号）。现在假设你的秘密消息为300KB大小（30万个1字节字符），你打算把它隐藏在600×600像素×24位的图像里（即1 440KB的图像；1 440KB=600×600×（3+1），3代表3个RGB通道，1代表1个alpha通道。简而言之，就是每个像素需要4个字节的存储空间）。该算法包括以下步骤。

步骤 1：创建一个（一对多）表格，在字母表中的80个字符，其中的每个字母都与广泛分布在RGB空间的1 000 RGB颜色相关，并且无碰撞和重叠（RGB成分最多与一个字符相关）。所以你需要80 000条记录的查找表，每个记录3字节大小。（因此这个查找表的大小为240 KB。）从某种程度上说，此表相当于加密系统里的密钥。

步骤 2：在目标图像中嵌入你的信息。

- 2.1. 预处理。在目标图像中，用邻近的颜色来替换那些查找表中与80 000个颜色之一匹配的像素。例如，如果像素颜色R=231、G=134、B=098既存在于目标图像中，又存在于80 000条记录的查找表中，则用（比如说）R=230、G=134、B=099替换此目标图像中的颜色。
- 2.2. 在600×600目标图像中随机选择30万个像素的位置。这就是30

万个字符信息存储的地方。

- 2.3. 将30万个位置，从80 000条记录的查找表中找到的最近邻居，来替换RGB颜色。

信息被嵌入到目标（原始）图像后，图像和之前看起来完全相同。

## 如何解码图像

寻找在80 000 RGB颜色查找表中发现的RGB颜色像素，并与它们所代表的字符相匹配。这个过程相对简单，因为这个查找表有两个字段：字符（在你的字母表中有80个独有的字符）和RGB表示法（每个字符有1 000种不同的RGB表示）。

## 如何发布你的信息

已经将信息安全编码且隐藏在一个图像中，你会认为只需要用电子邮件把图像发送给收件人就可以了，他也可以很容易地提取已编码的信息。这是一个危险的行动，因为即使加密的消息不能被解码，但如果你的电子邮件账户或收件人的电子邮件账户被劫持（例如，被美国国家安全局劫持），那么劫持者至少可以找出是谁发出的信息，或是发给谁的。

要传递消息还有一个更好的机制，就是将你的图像匿名发表在Facebook或其他公共论坛上。要做到匿名必须非常小心。使用假的Facebook个人资料发布高度机密的内容（已经利用隐写技术隐藏在图像中）以及240KB的查找表，还不能暴露你的IP地址。

## 最后的评论

隐藏在图像里的信息不应该包含标识符，因为如果被捕获和解码，劫持者可能会找出是谁发送的，或发送给谁的。

大多数现代图像包含元标记。当用户使用谷歌做关键词搜索时，元标记可帮助用户轻松分类和检索图像。然而，这些元标记存在安全性风险：它们可能包含关于创造的图像的用户及其IP地址、时间戳和机器ID的信息。因此，它可能导致劫持者发现你的身份。这就是你应该修改或删除这些元标记的原因。建议使用来自网络的图像（不是由你自己创建的）为载体图像，或者自己编写54字节大小的BMP标头。元标记不应该

像伪造过的，因为这可能使得图像被标记为可疑图像。重新使用从外部获取的现有图像（不是在你计算机上生成的图像）作为载体图像是一个好办法。

一个提高安全性的方法就是使用查找表（那个240KB的表）的双重系统。假设你有10 000张嵌入了编码消息的图像存储在云端。查找表被称为密钥，并且也可以嵌入到图像中。为了提高安全性，你可以在其中混入另外10 000张没有编码内容的伪造（诱饵）图像和两个键：A和B。因此，你的存储库中一共有20 002张图像。你需要用A键解码B键，再用B键解码其他图像。因为所有的图像文件（包括键）看起来都一样，除非知道A键和B键对应的文件名，才能成功解码图像。所以即使你的云泄露了，你的编码消息也不可能被劫持者成功解码。

注意 如果你查阅Jessica Friedrich所著的*Steganography in Digital Media*（Cambridge University Press，2010年出版）一书，其中并未提及基于字母查找表的算法。（虽然字母查找表在书中没有提到，但在作者的核心算法中有提到。）当然，这并不意味着作者的算法就是新的，也不意味着它比现有的算法更好。该书更侧重讲解隐写分析（解密图像），而不是隐写术（编码技术）。

# 好的电子邮件加密

有没有考虑成立一家创业公司，专门提供任何政府或企业都无法解密的加密电子邮件，以及向那些不希望被竞争对手、罪犯或者政府窃取机密的企业提供安全的解决方案？

例如如下这样一个电子邮件平台。

- 它是一个Web应用程序，只兼容大小上限为100KB的纯文本消息。一些网页托管在某些Web服务器（简称A）上，把其文本复制并粘贴到某个网页表单里。再创建一个提取密码——可能是用另一个应用程序创建的比较长、随机的、高强度的密码。当点击提交的时候，系统对文本加密，且可在某个网络服务器（简称B）上访问到。屏幕上会显示一个缩短的URL，这是你或收件人能读取加密文

本的地方。

- 打电话（或发传真）给收件人，可能在公用电话亭拨打并接听电话，向他提供那个缩短的URL以及取回和解密该消息所需的密码。
- 收件人访问缩短的网址，输入密码，可以在线（在服务器B）读取未加密的消息。无论是收件人阅读后，还是加密邮件创建48小时后，该加密文本都会被删除。
- 加密算法是这样的（在加密前添加半随机文本信息和加密时间戳，但如果半随机的文本不是先加入的，加密将无效），48小时后（如果加密版本被截获）消息将不能被解密，因为加密消息和可执行文件中嵌入了自我毁灭机制。如果你两次加密同样的（甚至是空的或只包含一个字符的）信息，那么两次加密的版本会有很大的不同，长度随机，大小差别至少1KB，使得进行逆向工程几乎不可能。也许每三四天，可执行文件的加密就会改变一次，以提高安全性，并确保先前加密的消息不能被解密。（服务器B上新旧版本同时可用，但只限48小时。）
- （在服务器A上的）执行程序在做任何加密之前，测试IP地址是否正确，比如防止它在政府服务器设备上运行。此功能是在加密执行代码中。在服务器B上，用于解密消息的可执行文件中也会做IP检测。
- 在加密算法中嵌有犯罪检测系统，通过检测似乎可疑的消息（儿童色情、恐怖主义、诈骗、煽动仇恨的言论等），对他们不提供加密服务，防止罪犯利用该邮件系统。
- 该平台吸引一些广告主，如防病毒软件（如Symantec）或谷歌Adwords，从而通过付费广告货币化。
- 与B关联的URL可以在任何地方、任何时间改变，基于用户提供的密码访问，可以链接到美国以外的国家。
- 与A相关的URL必须是较为静态的。这属于弱点，因为会被政府击破。然而，还是有变通的办法。可以为该应用程序使用一些特定的关键词，如ArmuredMail。如此，即使A被击破，一个基于相同关键词的新网站也将会在其他地方出现，保持服务不间断。（用户在谷歌上搜索ArmuredMail以找到A的镜像网站。）
- 最后，不在任何一个地方存储没有加密的文本。

事实上，政府可以创建这样一个应用程序，并伪装成一家民营企业，这种情况倒是相当有吸引力。

注意，没有绝对安全的系统。如果有人给你的电脑装了秘密摄像

头，记录下了你所有的操作，那么该系统便失去了保护作用，不过对于收件人来说仍然是安全的，除非他也有一个摄像头跟踪他所有电脑的操作。但是，你和收件人之间的联系（也就是说你们存在联系这个事实）是对任何第三方都不可见的。此外，在匿名的计算机——比如某酒店大堂的公共计算机上，使用该Web应用程序，也能够提高安全性。

为了进一步提高安全性，该系统可以提供一个电子邮件工具，如 `www.Black-HoleMail.com`，其工作原理如下。

- Bob（bob@nsa.com）想发信息给John（john@ge.com）。
- Bob加密了john@ge.com，使之变成x4ekh8vngalkgt。
- Bob将电子邮件发送到x4ekh8vngalkgt@BlackHoleMail.com。
- BlackHoleMail.com在解密了收件人的邮件地址后，把信息转发给john@ge.com。
- John不知道是谁发的消息。他只知道该消息来自于x4ekh8vngalkgt@BlackHole Mail.com，这就是他所知道的关于发件人的所有信息。

为了防止执法机构闯入系统，加密电子邮件地址48小时有效。这样一来，他们仅可以重建在过去48小时期间使用的电子邮件地址。总之，这个系统使得美国国家安全局和类似机构更难识别到底是谁和谁在相互联系。

双重加密系统会更加安全。

- 用C来加密你的邮件。（加密的版本是文本格式。）
- 用A来加密需要加密的文本。
- 收件人使用B来解密消息，解密出来的消息仍然是一个加密消息。
- 然后，收件人使用C来完全解密这个经双重加密的信息。
- 你甚至可以使用2层以上的加密。

# 验证码破解

设计一种正确率达90%的验证码识别算法，并提出改善验证码系统

的建议：对于未来的数据科学家来说，这会是一个十分有趣的项目。

对验证码系统进行逆向工程需要6个步骤。

1. 收集大量的图像，每个字符至少要有20种表现形式。这大概需要收集1 000多张验证码图片。

2. 对每个图像使用一个简单过滤器以过滤噪音，工作原理如下：（a）用邻域像素里的中值颜色取代原来的每个像素；（b）颜色深度由24位减少到8位。通常情况下，你会使用过滤器去除单独的斑点，以提高亮度和对比度。

3. 对图像进行分割，以确定轮廓，把图像二进制化（将深度从8位减少至1位，即黑色和白色）、矢量化，再通过重新连接被分割的各部分（边缘）将矢量结构（一系列保存为图像结构的节点列表和边缘）简化。

4. 执行无监督聚类，其中每个有关联的组成部分均由先前的分割抽取出。它们中的每一个都应该表示为一个字符。如果顺利的话，你已经收集了1 000多个样本字符，且字母表中的每个字符均带有多个版本。人为地给每个有关联的组成部分添加标签来代表字符。每个人为添加到字符上的标签就是一个字母。现在，训练集里的所有验证码已经被成功解码，但愿有90%以上的成功破解率。

5. 通过每天收集验证码进行机器学习，应用前面的步骤，再将每个字符的新版本（经由每日收集验证码获取）添加至训练集。长此以往，你的训练集将不断变得更大、更好。找出训练集中那些难以区分的成对字符，并删除易引起混淆的样本字符。

6. 应用步骤2和步骤3，使用验证码解码器提取出每个验证码中的字符。然后基于你的训练集，执行监督聚类 来确定它们所代表的字符。此操作对于每个验证码应该不超过1分钟。

这个通用验证码解码器可能擅长处理某些类型的验证码（模糊的字母），而对于处理其他类型的验证码（字母被随机线网纵横交错）也许效果就一般了。

请注意，一些黑客已经设计出了可以完全绕过验证码的技术。他们

设计的系统甚至可以在不“读取”验证码的情况下，每次都能获取正确答案。这些系统进入更深一层的服务器，读取正确答案，再把验证码的正确答案填入Web表单中。垃圾邮件技术可以绕过注册表格最具挑战性的问题，如在1秒内分解两个大素数（至少2 000位数）的乘积。当然，他们不提取主要因素；相反，会直接从被入侵的服务器、JavaScript代码或网页中读取正确答案。

无论如何，这个有趣的练习会教给你一些关于图像处理和聚类的知识。最后，你应该能够察觉到某些可以使验证码更有效的特性。

- 使用破碎的字母，例如将一个单独的字母C分成三四个单独的部分。
- 使用多个验证码算法，每天更改算法。
- 在验证码中使用特殊字符（圆括号和逗号）。
- 在字母中间挖洞。
- 编码成两个字母的组合（例如，ab、ac、ba等），而非单独一个字母。那么，攻击者将必须解码数百个可能的符号，而不仅仅是26个或36个，并且将需要一个大很多的样本。

# 欺诈检测

本节主要讨论特定类型的欺诈行为：点击欺诈，也就是直接或间接由网络广告联盟生成假点击以从广告主那里骗钱的行为。你会了解到一种方法，用来识别每年偷取超过1亿美元的大量僵尸网络（Botnets），你还将了解如何在数万亿的特征组合中选择特征；（特征选择是一个组合数学问题）你还会发现用于度量一个特征或一组特征预测能力的指标。

交叉参考 更多信息请参阅第4章的互联网技术和欺诈检测的指标，以及第5章的映射和隐藏决策树。

# 点击欺诈

一般来说，根据定义，点击欺诈通常指在对目标网址明显不感兴趣的情况下，仍故意打开点击付费（Pay-Per-Click）活动中的广告的行为。主要涉及以下两种类型。

- 广告主点击竞争对手的广告，来消耗他们的广告支出预算，这类欺诈通常发生在清晨且通过多个分发合作伙伴进行：AOL、Ask.com、MSN、谷歌、雅虎等。
- 恶意的分发合作伙伴，为增加其收入，而使用点击机器人（点击僵尸网络）或付钱给他人点击产生流量，看起来像真正的点击一样。

虽然这两个属于非转化流量的重要来源，但还有许多其他流量较差的来源。其中有些被称为无效点击，而非点击欺诈，但从广告主或发布者的角度看，两者之间并无差别。无论是否导致欺诈行为的发生，还是行为本身是否有诈骗意图，以及诈骗意图是否因金钱动机产生，结合所有类型的非收费或部分收费的流量考虑的话，这些不良流量来源包括以下类型。

- 意外欺诈：　并非专为点击欺诈而制造的机器人，可能会因为一个设计缺陷，而运行不慎，失去控制，导致其点击所有链接。如垃圾邮件发送者为了收集电子邮件地址而使用的机器人，这种机器人不是专为点击欺诈的目的而设计，但最终仍会花费广告主的钱。
- 政治积极分子：　这些人并非出于金钱动机，而是出于仇恨。在集体诉讼中，他们用恶意点击来反对企业招聘员工，从而带来虚假点击和转化。这是一种有害的欺诈点击，受害者以为其点击付费活动产生了许多转化意向，然而实际上大多数转化意向（比如留下的电子邮件地址）都是伪造的。
- 心怀不满的人：　可能是在点击付费广告公司或搜索引擎公司工作过，而且最近被开除的雇员。或者可能是某位认为自己被不公平禁言的发布者。
- 点击付费社区中不道德的人：　小型搜索引擎试图通过制造不合格的点击，或股东欺诈，来败坏其竞争对手的名声。
- 有组织的罪犯：　垃圾邮件发送者和其他互联网侵犯版权者常常用到机器人和病毒，这些设备能够通过编程来生成点击欺诈。恐怖主义筹资属于这一类。联邦调查局（FBI）和美国证券交易委员会（SEC）负责调查该犯罪行径。
- 黑客：　现在很多人都可以使用自制的网络机器人。虽然用机器人

伪造流量很简单，但模仿合法流量更加复杂，因为它需要欺骗成千上万的普通IP地址，业余者未必能做得很好。有些人制造高品质的模拟流量，仅仅只是为了挑战，没有金钱动机。

下面讨论其他问题的来源，通常不被认为是点击欺诈，而是有时被称为无效点击、非收费点击或低质量点击，包括以下类型。

- 曝光欺诈：　曝光次数和点击次数应该关联起来考虑，而不是单独处理。对于搜索引擎来说，这就需要把大型数据库关联起来，将用户、曝光与点击匹配。在一些场合中，曝光欺诈是为了让对手的点进率（click-through rate）看起来比较低。更先进点的做法是使用好的代理服务器，从而不露痕迹。当点进率降到足够低时，则不再显示竞争对手的广告。该做法常与自点击（即广告主通过代理服务器点击自家广告提高其排名，从而提高其在搜索结果页的位置的做法）相互配合。该方案针对付费和自然流量。
- 多次点击：　虽然多次点击不一定属于欺诈，但最终不是花费了广告主的钱（当广告主被全价收费的时候），就是花费了发布者和搜索引擎的钱（如果是首次点击收费的话）。还有一个问题是如何准确地确定两次点击——如5分钟的间隔——来自同一用户。
- 虚拟欺诈：　出现点击欺诈，但永远不会被收费。这些点击是被不道德的点击欺诈公司编造出来，或者是测试活动的结果，被称为点击干扰。谷歌机器人就是一个典型的例子。虽然谷歌从来不收取源自其谷歌机器人点击产生的费用，但是其他没有谷歌机器人最新IP地址列表的搜索引擎，便可能会意外地因为这些点击而收费。

# 连续点击评分与二进制欺诈/非欺诈

网络流量不是非黑即白的，而是从低质量到高质量涵盖整个范围。此外，不带来转化的流量，其质量也不一定就差，在许多情况下，实际上可能质量不错。缺少转化可能是因为广告本身质量不佳或针对性差。下面两点值得提出来。

- 流量评分：　任何流量源高达5%的流量很快能确认为完全不可结算

和没有转化机会的类型，此类流量比例高过5%的话，则是普适的质量问题，而非具体广告主的问题。对有意给分发合作伙伴进行排名的搜索引擎，以及想改进广告活动的广告主来说，流量评分方法（点击评分或曝光评分）是解决这个问题更加可执的机制。

- 一般而通用的评分方法，让没有投资回报率（ROI）指标或者指标有限的广告主，能够测试新的流量源，从而提早知道，若抛开转化率不谈，普遍不错的流量在哪里。这可以帮助广告主大幅提高影响力，开拓新的流量来源，而不是从A/B测试中小小地提高投资回报率。点击评分对一些线下广告主、虚假转化的受害者，以及对品牌推广感兴趣的广告主来说，价值最大。

计分方法可以帮助搜索引擎确定多次点击的最优价格（如真正用户产生多次点击，而非技术故障产生的双击）。通过把分数并入到搜索引擎的智能定价算法中，借助“每个广告每个用户每天一次点击”这个简单的业务规则，来减少损失。

搜索引擎、广告发布者和广告主都可以获益，因为可以接受质量差的发布者，但会对投放广告做出正确的定价，使得广告主的投资回报率为正。优秀的发布者若质量下降，可以根据点击的分数减少佣金，而不是彻底停掉。当它的流量变得更好时，其佣金会随着分数相应增加。

为了使搜索引擎靠谱，评分系统需要尽可能通用。通用的转化情况下，点击分数应该设计成跟转化率分布匹配，同时考虑到虚假转换，并基于归因分析 来给带正确用户标示的点击进行转化评分。一个IP可以属于多个用户，而且单个用户也可以在2分钟的时间内拥有多个IP地址。日志（特别是服务器日志，其次是重定向日志）也存在饱受诟病的缺陷，因此处理广告主服务器日志数据时，不应该仅仅依赖于cookie。

我亲自设计了基于点击日志的评分系统，比如是依靠网络拓扑结构的指标。评分系统是基于广告主服务器日志设计的，也依赖于网络拓扑结构指标（分发合作伙伴、每个IP集群独特的浏览器等），甚至还依赖于“曝光—点击率”指标和其他搜索引擎指标。这是因为服务器日志与搜索引擎报告合并时，可以获得最准确的信息。使用搜索引擎指标为广告主的流量评分，能够为搜索引擎数据设计好的评分系统，而反过来，搜索引擎分数也与真实的转化率密切相关。

在处理广告主服务器日志的时候，无论何时尽可能使用恰当的标签（例如，谷歌的gclid），可以排除浏览器中产生的虚假点击数。

广告主评分系统旨在成为转化率的优秀指示器。搜索引擎的分数使用基于专家知识和广告主数据的权重组合。评分应当跟信用卡评分系统方法相同，从而顺利地进行且得以标准化。最好的质量评估系统依靠实时和颗粒度小的评分，比如以天为单位。

采用基于可靠指标的平滑评分系统，能极大地减少误报。如果（点击）触发一个规则，或者就算触发两个规则，则可能会对点击进行惩罚。此外，如果一条规则被点击触发太多，或者和真正的转化不相关，则会被忽略。例如“双点击”（在两次点击之间有足够长的时间）规则，就是一个很好的转化指示器，相关性为正时，它从规则变成反规则。没有外部来源但正常的点击，在评分标准化后，不会被惩罚。

# 数学模型与基准

我所研制的评分方法非常先进，基于的是最近30年对实时和历史数据审计、统计和欺诈检测的经验。它结合了复杂的交叉验证、实验设计、连接聚类和用于寻找新规则的无监督聚类、机器学习、评分领域曾使用的最先进模型，以及能立即产生大量重叠较少的决策树的快速、稳健、并行化算法。该聚类算法是独特决策树技术和一种新类型的PLS逻辑逐步回归的混合，可以处理数以万计的高度冗余指标。它可在短时间内计算出有意义的回归系数，且可有效地处理规则之间的相互作用。

该方法在一些方面的表现与脊回归、树bagging、树boosting稍有相似之处（见 https://www.cs.rit.edu/~rlaz/prec20092/slides/Bagging_and_Boostin）。现在你可以比较不同系统的效率，以在高度逼真的模拟数据中检测点击欺诈了。比较的标准是均方误差，此为测量点击和转化之间相符程度的指标。

- 具有相同权重的评分系统： 相比二进制（欺诈/非欺诈）方法提高了60％。
- 一阶**PLS**回归（偏最小二乘回归）： 相比二进制的方法提高了113％。
- 全部标准回归（不推荐，因为其结果高度不稳定且不可解释）：相

比二进制的方法提高了157％。

- 二阶**PLS**回归（偏最小二乘回归）：　　相比二进制的方法提高了197％，这是一种有简单解释的、强大的、几乎无参数的技术。
- 大量额外的改进：　能在添加决策树组成部分时获得。实际数据的改善率是相似的。

# 虚假转化产生的偏差

因为虚假转化的影响比大多数人想象得更加严重，故在此详述其产生原因。如果不谨慎对待，这些转化会导致欺诈检测系统有严重偏差。依靠售前或非销售式（软性）转化——如报名表格——的搜索引擎，评估流量时可能会误以为一些流量质量较好，但其实是糟糕的。

通常，广告主不愿意向搜索引擎提供太多的信息，因此转化计算一般是由于广告主在目标转换页面上放置了一些JavaScript代码或透明的gif图像（信标）的结果。然后搜索引擎可以跟踪这些网页上的转换。然而，搜索引擎无法控制广告主想要跟踪哪一个“转化页面”。此外，搜索引擎无法知道点击和转化之间具体发生了什么，以及转化后的情况。如果搜索引擎只拥有预售数据的访问权限，那么虚假转化的可能性很高。虚假转化显著增的时候，可能会出现一些特定的流量段。

虚假转化的另一个问题是，当广告主（例如，广告代理人）购买上游流量时，搜索引擎会把下游流量分散给其他广告主，这个商业模式相当普遍。如果上游流量是人为的，带来许多虚假的转化——转化是一个点击或跳转——广告经纪人不会看到投资回报率下降，事实上还可能看到投资回报率上升，只有下游的广告主开始抱怨。当发现问题时，可能已经太晚了，也许已经导致广告经纪人丢失客户。

此商业缺陷可能会被那些操纵分发合作网络的犯罪分子所利用。狡猾的犯罪分子会加大对这种类型“广告代理”的广告主的攻击力度。犯罪分子可以生成虚假的点击来赚钱，只要他们生成足够数量的虚假转化，受害者也能够赚钱，并且可能不会注意到这个阴谋。

# 一些误解

有人认为，点击欺诈的受害者是优秀的发布商，而不是广告主，因为广告主可以自动调整出价。然而，这对缺乏良好转化率指标的广告主不适用（例如转化是离线发生的），也不适用于不能实时更新出价和关键词的小广告主。当造假问题是在线上进行时，它实际上可能导致广告主永久丢掉整个流量段，以及失去良好的投资回报率。在一些二线网络中，曝光诈骗可能导致广告主迟早被踢出局且永无东山再起之日。在这种情况下，搜索引擎和广告主双双一败涂地，获胜的是在网络上刊登俗气、不相关广告的坏家伙。网站用户也属于输家，因为所有好广告都被不相关信息代替了。

最后，许多检测欺诈的系统基本上仍然是基于异常值检测和检测平均偏差。但大多数骗子会忽略过于昂贵或便宜的点击，努力使其点击价格看起来尽可能平均，并通过用户代理进行恰当的分发，摒除掉通过大型代理的点击，每天在每个受感染的计算机上生成一些零碎的随机点击。揭露这种类型的欺诈需要一个真正多元的方法，同时看着数十亿细心挑选的变量组合，从数十亿点击的极小部分里寻找统计证据，以揭露更多影响大量点击的复杂欺诈案件。

# 统计面临的挑战

在一定程度上，打击点击欺诈的技术类似于银行使用的打击信用卡诈骗的技术。最好的系统是基于统计评分技术建立的，因为交易——该语境下也就是点击——通常来说非好即坏。

基于IP分数、点击分数和度量指标混合（特征）优化的多重评分系统是基本的要素。由于包含大量的数据，以及一个良好评分系统中潜在的数百万度量指标，使用如马尔可夫链蒙特卡洛（Markov Chain Monte Carlo）算法或模拟退火算法来进行组合优化是必需的。

虽然由于因变量是转化指标，给广告主打分可以看作一个回归问题，给搜索引擎打分更具挑战性，因为转化数据并不容易获得。即使在

处理广告主的数据时，也有几个问题需要解决。首先，分数需要标准化。如果登录页面不同，则两个相同的广告活动也可能会有不同的表现。评分系统需要解决这个问题。

此外，虽然评分可以看作回归问题，但也是有一定难度的回归问题。所涉及的指标通常是高度相关的，使得回归问题从数学的角度来看是病态的（ill-conditioned）。指标可能比观察到的点击数还要多（也因此有更多的回归系数），故回归方法变得高度不稳定。最后，回归系数——也称为权重——必须加以限制，只能取几个潜在的值。因变量是二进制的，这就变成了一个复杂的岭型逻辑回归问题。

最好的技术实际上是依赖于能够处理反向配置的混合系统，如“时间早于凌晨4点”是坏的，“国家不是美国”是坏的，但“时间早于凌晨4点且国家是英国”是好的。良好的交叉验证对于消除无统计学意义或不太健壮的配置和指标也非常关键。把指标（特征）仔细分段，以及快速的分布式特征优化算法是很重要的。

最后还有至关重要的几点，那就是创建测试活动的实验设计——一些欺诈比例高，一些没有欺诈——这要用到通用转化指标以及恰当的用户识别。记住，没有剔除虚假转化会导致系统片面且带有许多误报。在实际中，转换率超过10%的大量流量和转化率低于2%的大量流量，应分别处理。

# 点击评分优化关键词出价

点击评分可以做很多事情，包括：

- 确定与点击相关的最优定价。
- 确定具有潜在转化流量的新来源。
- 在无转化或存在虚假转化的情况下，测量流量的质量。
- 预测被添加至点击付费活动的新关键词（没有历史数据）的转化机会。
- 评估分发网络合作伙伴的质量。

这些仅仅只是点击评分的几个应用范围之一。还要注意的是，评分不仅限于点击，也可以涉及曝光和指标，如每个曝光的点击数。

从广告主的角度来看，点击评分的一个重要应用是检测流量的新来源以提高总收入。而这一点通过A/B/C测试、传统的投资回报率优化或者搜索引擎优化（SEO）的方式均无法实现。这个想法在于接入精心选择过的新流量来源，而不是改进现有的。

现在考虑下面两种类型的评分框架。

- 评分Ⅰ：　使用一群广告主，很可能几十个同一类别的广告主计算一个通用的评分。
- 评分Ⅱ：针对特定的广告主定制的评分系统。

当你把这两者结合起来的时候，你能得到什么？解决方案如下。

- 评分Ⅰ和评分Ⅱ都很好。这通常是广告主考虑的两个流量段。通常情况下，广告主着力于搜索引擎优化或A/B测试，以进一步完善质量并获得一些优势。
- 评分Ⅰ好，评分Ⅱ差。此流量通常遭到拒绝。不设法理解为什么良好的流量未进行转换。广告主拒绝这些流量可能会失去收入的主要来源。
- 评分Ⅰ差，评分Ⅱ好。这是广告主考虑的另一个流量段。不幸的是，虽然在这种情况下，表面上广告主会因为得到转换而高兴，然而，这也是一个红色标志，表示转换可能是伪造的。当转换包括填充Web表单的时候，这种情况经常发生。任何试图提高转换的行为（例如，通过搜索引擎优化）都会适得其反。相反，应该认真研究调查流量。
- 评分Ⅰ和评分Ⅱ都很差。在这里，大多数情况下，对策仅仅是使流量来源完全、永久下降。这也是一个糟糕的方法。通过使用一个基于点击评分的策略来减少流量，可显著降低遭遇不良流量的机会，且当流量质量提高时也不会错过机会。

这里得出一个结论，在点击付费活动中的关键词竞价（如谷歌的AdWords）中，对于几乎没有或没有历史记录的关键词汇，汇总关键词得到的点击分数预测转化率十分有效。这些关键词占所有购买关键词的

比例高达90%，且超过总广告费的20%。这些分数在实时竞价中也很有用。

# 组合优化自动快速的特征选择

特征选择讲述的是一种在几十或数百个特征（也被称为变量或规则）里检测最佳子集特征的方法。“最佳”指预测能力最强，下面一节定义了其概念。总而言之，你的目标是剔除那些重复的特征，简化（几个特征之间）相互关联的结构，剔除没有价值的特征，如那些因呈现随机值而缺乏预测能力的特征，以及几乎从来没有被触发过的特征（规则）（除非被触发时是完美的欺诈指标）。

该问题符合组合数学的性质。举个例子，你想要从一组500个特征中选出一个便于管理的小型特征集（如20个特征），来运行隐性决策树（或其他一些分类/评分技术），并且从统计角度上看足够稳健。但要在500个特征中挑选出20个，一共有$2.7\times10^{35}$ 种组合，你需要把所有的组合计算一遍，以找出最具预测能力的那一个。这个问题是难以计算的，你需要找到替代的解决方案。好消息是你不需要找到那个绝对的最大值，而只需要找到20个足够满足条件的子集就可以了。

这里有一个操作方法。首先计算每个特征的预测能力。然后，在子集中每次添加一个特征（从0个特征开始），直到你满足了下面的任何一个条件。

- 得到（你要求的）20个特征。
- 添加一个新的特征已不能显著提高子集的整体预测能力。（简而言之，已经实现了收敛性。）

在每次迭代中，从两个剩余的具有最高预测能力的特征中，选择要添加的特征。（从待挑选的子集中）选择（两个特征中间）那个最能增加整体预测能力的特征。现在，你的计算量已经从$2.7\times10^{35}$ 减少到40，也就是$2\times20$了。

注意　一个提高预测能力的额外的步骤：每次在子集中删除一个

特征，并从（外部子集中）剩余的特征中随机选取一个取代它。如果这个新特征能增加子集的整体预测能力，则保留它；否则，放回到旧的子集。重复此步骤10000次，或直到不能获得更高的预测能力为止（以先到者为准）。

最后，你可以一次添加2个或3个特征，而不是一次1个。有时，组合特征比单一特征具备更好的预测能力。例如，如果特征A=国家，值落在{美国，英国}，特征B=一天中的小时，值落在{“白天——太平洋时间”，“夜晚——太平洋时间”}，如果分开来，这两个特征几乎不存在任何预测能力。但如果把两者结合，你将拥有一个更强大的特征：英国/夜晚是好的，美国/夜晚是糟糕的，英国/白天是糟糕的，美国/白天是好的。使用这种混合特征同时降低了误报或漏报的风险。

此外，为了避免过于复杂的特征，使用列表是一个好办法。也就是说，避免使用特征A=国家（有200个潜在值），特征B=IP地址（有数十亿的潜在值），转而使用：

- 特征A=国家小组，有3个国家列表（高风险、低风险、中等风险）。这些小组会随时间而改变。
- 特征B=IP地址类型（有六七个类型，如其中之一是“IP地址在一些白名单里”）。详细信息可查看IP拓扑结构映射一节。

# 特征的预测能力：交叉验证

本节基于两种特征的子集来做验证，从更常规的角度阐述了预测能力的概念。假设你有两个二进制特征A和B，取两个可能值——0或1。另外，在欺诈检测的背景下，假设在训练集合中的每次观测结果要么是Good好的（无欺诈），要么是Bad坏的（欺骗）。欺诈状态（G或B）在统计学中被称为因变量 或应变量 ，特征A和B被称为规则或自变量。

交叉验证

首先，把你的训练集（数据已知，响应B或G）分成两个部分：控制部分和测试部分。确保这两个部分数据足够丰富：如果测试集比较大

（数百万的观察结果），但（在200个中）只包含一两个客户端，这样的数据太贫乏了。当处理训练集之外的数据时，统计推断将会受到负面影响（导致低健壮性）。使用两个不同的时间段来控制和测试是一个好主意。你将计算控制数据的预测能力（包括规则选择）。当最终决定最佳子集的特征后，你就可以计算测试数据的预测能力了。如果测试数据里的预测能力显著下降（与控制数据中的相比），这说明你的分析存在错误，那么就发现问题，解决问题，并重新开始。你可以使用多个控制和测试集，以便了解预测能力在不同控制集之间是如何变化的。过多的差异是一个应该处理的问题。

预测能力

使用前面的例子，二进制的特征（A, B）取两个值（0,1），你可以将控制数据中的观察结果分成8类。

- A=0，B=0，响应=G。
- A=0，B=1，响应=G。
- A=1，B=0，响应=G。
- A=1，B=1，响应=G。
- A=0，B=0，响应=B。
- A=0，B=1，响应=B。
- A=1，B=0，响应=B。
- A=1，B=1，响应=B。

现在将8类中每类观测结果的数量表示为$n_1$ , $n_2$ ，...$n_8$ ，并引进以下量：

$$P_{00} = n_5 /（n_1 + n_5 ）, P_{01} = n_6 /（n_2 + n_6 ）, P_{10} = n_7 /（n_3 + n_7 ）, P_{11} = n_8 /（n_4 + n_8 ）$$

$$p=（n_5 + n_6 + n_7 + n_8 ）/（n_1 + n_2 + … + n_8 ）$$

假设测量欺诈行为的总体比例为p，且p低于50%（即p<0.5，否则你可以在欺诈和非欺诈之间互换）。对于任何介于0和1之间的r，基于参数**a**（0<**a** <1，我建议**a** =0.5−p）定义W函数（形状像W），如下：

如果0 < r < p，则W（r）=1 –（r/p）；

如果$p < r < 0.5$，则W（r）=**a** *（r–p）/（0.5–p）；

如果$0.5 < r < 1-p$，则W（r）=**a** *（r–1+p）/（p–0.5）；

如果$1-p < r < 1$，则W（r）=（r–1+p）/ p。

通常情况下，r=$P_{00}$、$P_{01}$、$P_{10}$ 或$P_{11}$。W函数具有以下属性：

- 当r=p或r=1–p时，也就是，当r不提供任何有关欺诈/非欺诈的信息时，它小于或等于0。
- 当r=1或r=0时，也就是，当在给定的桶里，你能完美地分辨欺诈和非欺诈时，它大于或等于1。
- 它是对称的：W（r）=W（1–r）且0<r<1。所以如果你互换了好和坏（G和B），它仍然提供了相同的预测能力。

现在给预测能力下定义，如下：

$$H=P_{00}\,W(P_{00}) + P_{01}\,W(P_{01}) + P_{10}\,W(P_{10}) + P_{11}\,W(P_{11})$$

函数H跟特征子集{A, B}的预测能力相关，具有4个桶——00、01、10和11，分别对应（A=0, B=0）、（A=0, B=1）、（A=1, B=0）以及（A=1, B=1）。虽然H似乎与熵 不甚相关，H的设计旨在满足良好的特性并受参数驱动（比如上述参数**a** ）。不像熵，H并非基于物理概念或模型，它实际上是一个合成的（虽然有用）指标。

请注意，在H中的$P_{00}$ ...$P_{11}$ 权重保证了频率较低（即低触发率）的桶对H的低影响。事实上，建议将W（r）设置为0，表示任何少于20个观测结果的桶。例如，桶00的触发率是（$n_1 + n_5$）/（$n_1 + \ldots + n_8$），该桶的观察数是$n_1 + n_5$，且r=$P_{00}$ =$n_5$ /（$n_1 + n_5$）。如果$n_1 + n_5$ =0，则设置$P_{00}$ 为0且W（$P_{00}$）也为0。其实，不仅是当$n_1 + n_5$ =0时推荐这样做，无论何时只要$n_1 + n_5 < 20$，尤其是当p较小时，我也推荐这样做。（如果p非常小，如p<0.01，那么当构建训练集以及对计数进行相应权重时，你需要对坏的交易进行过抽样。）当然，同样的规则也适用于$P_{01}$、$P_{10}$ 和$P_{11}$。请注意，你应该避免观察结果有很大比例是空桶的特征子集，以及产生大量空桶的特征子集。训练集以外的观察结果很可能为空桶或几乎为空桶，并导致预测结果容易波动。为了避免这种弊端，

应该坚持使用二元特征，选择多达20个特征，并使用（混合的）隐性决策树方法来进行评分交易。最后，因为i,　j=0,1，所以$P_{ij}$　是P（A=i, B=j）的朴素估计概率。

H的预测能力有一些有趣的特性如下。

- 如果特征子集没有预测能力，则它始终在0和1之间或等于0；如果特征子集具有最大的预测能力，则它等于1。
- 可以通过设置p=0.5来创建H（不取决于p）的一般版本。那么W函数的形状则将不再和W相像，而是与V相像。

## 数据结构和计算方法

你可以为顶尖的20个特征（即拥有最高预测能力）预先计算所有桶的数量结果$n_i$ ，并把它们存储在一个最多有$2\times2^{20}$ 条目的小哈希表内，（约200万；系数是2，因为每个桶需要两个量度：B的数量和G的数量）。此哈希表中的条目如下。

$Hash {01101001010110100100_G}=56

这意味着，桶#　01101001010110100100有56个良好（G）的观测结果。

哈希表通过按顺序依次解析训练集而产生：每次观察时，计算标志向量（标志向量是指触发了哪条规则，比如01101001010110100100就是标志向量），检查它是好还是坏，并根据下面的指令相应更新（每次增加计数1）关联的哈希表条目。

$Hash{01101001010110100100_G}++

这样的话，每当你需要测量这20个特征子集的预测能力时，就不需要再次分析大数据集（可能高达数十亿个观测结果）了，相反，只需访问这个小哈希表即可。此表包含了所有构建的标志向量，以及计算任意排名前20的特征组合的分数所需要的条件。

你甚至可以做得比前20名更好，甚至前30名。虽然这将创建一个有2亿个条目的哈希表，但其中大部分的条目跟空桶对应，因此不会存在

于哈希表中。你的哈希表可能只包含20亿个条目，但也许太大而无法放入内存，所以需要借助于MapReduce或Hadoop来实现。

更好的是，为排名前40的特征构建哈希表。它可以完全解决前面所提到的特征选择问题。然而，现在这个哈希表可能有多达2万亿个条目。但如果数据集只有1　000亿个观测结果，那么哈希表中当然不会有超过1 000亿个条目。在这种情况下，建议创建一个有2 000万观测结果的训练集，这样哈希表最多将只有2　000万个条目（排除空桶后加起来可能不到1 000万），这样就可以存入内存了。

通过分析前一步骤中40个特征哈希表输入，你可以计算出较多（例如，100）特征子集的预测能力。然后为每个哈希表输入中的标志矢量和G/B条目，遍历这100个目标特征子集来更新这100个特征子集计数（所有$n_i$ ）。在哈希表输出中存储或更新这些计数。哈希表输出中的键由两部分组成：特征ID和标志向量。随后循环输出哈希表以计算每个特征子集的预测能力。但这一步有待进一步优化。

# 勾连检测和僵尸网络的关联规则检测

一种勾连　的情况是多个分支机构（或子发布商）在同一诈骗计划中相互联系且共享相同的资源。通常情况下，这是个僵尸网络，即一组计算机的IP地址被病毒感染，每台计算机每天（最多）自动点击一次。

关联规则是通过外部指标试图把两个欺诈者关联的规则。例如，当一个子公司A被认定为欺诈，查看连接到它的所有IP地址，如果这些IP地址有50%或以上仅仅与另一子公司B（而没有其他地方）相关联，那么很有可能B也是欺诈计划的一部分。更明显的是，现在查看所有连接到B的IP地址，发现又一个公司C，它共享其中10%的IP地址，那么你就找到了欺诈计划的另一个参与者。我已经使用这种技术成功地确定了与一个欺诈者相关的几个其他来源。

同一类中的另一个规则是短访问（小于20秒）的比例，涉及两个规模小且不知名的子公司或来源域名的两个点击数。当这一比例超过一定阈值（例如，20％的特定子公司的所有访问者）时，那么就很可能涉及诈骗。通过两个点击访问，发现所有子公司的名单，然后查看其他还涉

及这些子公司以及其他关联公司的访问，你可以检测到部分不良分支机构。

最后，用于点击欺诈的僵尸网络还通常涉及其他的犯罪活动，且由一个操作员控制。因此，检测这些进行点击欺诈的僵尸网络的算法，也可以帮助检测其他犯罪行为：垃圾邮件、网络钓鱼、ID盗窃等。

# 模式检测的极值理论

从统计的角度来看，因为犯罪分子没有投入太多精力来创造看起来像正常流量一样的虚假流量，所以他们的流量充满了极端的事件，使之更容易被检测出来。这些事件的概率可以通过蒙特卡洛模拟（在正常的流量里）估计。当发现一个事件（在可疑流量里）的发生率低于一百万分之一的时候，这就是一个危险信号。当然，如果你在追踪数十亿的事件，那将无法避免误报。阅读“大数据‘诅咒’”（第2章）一节，看看如何解决这个问题。已成功用于检测欺诈的罕见事件如下。

- 长期来自同一个推荐源，且每天都恰好是50%转化率的访问。
- 长期每天正好产生8个点击的这类IP地址的数量激增。（当为一个特定的子公司或关键词绘制每IP点击分布图时，你会看到一个尖峰值8。）
- 每周有3天以上表现为活跃但每天流量很少（最多一两个点击）的IP地址。如果你遇上符合上述条件的子公司占到40%，这是极为罕见的（对于大多数网站来说），也意味着危险信号。
- 当使用你的评分算法时，大量流量（如5%）的点击分数完全相同。这就非常可疑了。
- 在发生点击的网页上找到特定的关键词：“Viagra”或一些JavaScript标签，这些会经常在欺诈性网站上见到，或者甚至是空白页！
- 域名规则，如一个使用数十个域名的子公司，绝大多数用的名字都是xxx-and-yyy.com这样的模式（例如，chocolate-and-car.com）。

以上所有规则均已成功用于检测欺诈。要确认欺诈是否发生必须满足多条规则。例如，在一种情况下，不良的子公司同时满足域名条件、评分相同条件和空白网页条件，那么它会因为其所有点击的分数都相同

而被发现。通过进一步分析数据，我发现并创建了另外两条规则：域名规则和空页规则。

# 数字分析

数据科学与其他领域之间的界限往往不甚明确，商业智能（业务分析师从事的工作）更是如此。本节讨论的几个项目类型，在小型公司、创业公司和非传统的公司（聘请了很大一部分数据科学家）里，都由数据科学家一个人处理——数据科学家偶尔还会兼任业务分析师的角色，具备多种价值（polyvalent），可以索取更高的薪水并能够增加工作牢固性。

专业的领域知识是数据分析成功的关键。本节会分享一些专业领域知识，重点会分享经验法则和一些原理，即使数字和广告技术行业随着时间的推移不断变化，但这些法则和原理仍然有效（如可靠的搜索引擎优化（SEO）原则、电子邮件营销优化、广告匹配等）。

一些在本节中讨论的技术，通常由供应商（Comscore、Rocket及其他供应商）、搜索引擎（诸如Bing、谷歌、雅虎）、广告网络（例如，AdKnowledge或精明的发布商）实现，而不是通过广告主自己实现。尽管像eBay或Cars.com这样大的广告主，除了供应商的工具之外还开发和集成了自己专有的工具，但是广告主还是需要依靠供应商。重要的是，你要通过供应商和搜索引擎，来学习数据科学工作内容（数字分析）的类型，因为他们聘用的数据科学家，比所有的广告主全体加起来聘用的数据科学家都多。

交叉参考　想要知道更多有关数字分析或传统数据科学的讨论内容，请阅读第2章“大数据问题是数据科学所面临挑战的缩影”一节。

# 在线广告：到达率和频率的计算公式

对于广告主来说，怎样有效计算广告的到达率和频率呢？这里讨论的公式可帮助在线广告主计算到达率和频率，以帮助他们确定为了实现

目标要购买多少次曝光。下面这个例子说明：有时花两个小时重新发明轮子，并尝试用数学的方法解决问题和获取高效的解决方案，比计算机科学家传统的做法（如花时间试图在文献中找到解决方案或盲目运行昂贵的模拟或计算）更容易。

在一个网站上，广告主需要购买多少广告曝光才能实现特定的到达率？我研究了这个问题，并找到了一个简单的数学解决方案：

$$Rcach=唯一用户数–SUM\{U_k * （1 – P）^k \}$$

SUM求和是针对所有正整数（k=1，2,…）进行的。而$U_k$ 是在给定时间周期内访问k个页面的用户数目。所以，你需要计算在目标时间段内某个特定用户的页面访问量的分布，来计算$U_k$ 。P为已购买曝光页面数量除以总页面数的比例。

在给定时间里，看到过一个广告n次的唯一用户数量，由以下参数给定：

$$SUM\{ U_k * C（k, n） * P^n （1 – P）^{k-n} \}$$

SUM求和是针对所有大于等于n的正整数k。而C（k， n）=k!/ [ n!（k–n）!]是二项式系数。

在这种情况下，“唯一用户数”指的是用户不管在所计算的时间段内进行过多少次访问，均被视为一个用户。用户应该是人，因此应该过滤机器人（网络爬虫）。IAB（互联网广告局）就如何区分唯一用户及过滤机器人，给出了定义、指令和最佳实践。用户通常通过cookie来追踪。在过去，许多计数方法导致一个用户被多次计数，因而用“唯一用户数”一词来区分用户数和访问次数。（定义“唯一用户”这样的概念，也是数据科学家的工作内容之一。）

# 电子邮件营销：提高**300%**的性能

基于数据分析策略，并使用如下方法，使Data Science Central上的营销邮件打开率提高了300％，并显著提高了总点击数和点进率，方法

如下。

- 移除过去8年内没有打开过简讯的订阅者。这将大幅提高打开率，也会显著降低我们的“垃圾邮件评分”，因为简讯被置于垃圾箱或被当成垃圾邮件过滤掉的可能性几乎减少到0。
- 针对更精准的目标成员，进行订阅者细分，例如给位于欧洲的会员发送英国会议公告，但不发给亚洲或美国会员。
- 检测哪些消息打开率低，不再发送。
- 收集注册会员的信息（如职业、经验、行业、所在地区等）来进行更好的细分。
- 通过如下这些方法发展新用户。

- 只面向订阅者免费开放优质产品，如电子书，如果他们想下载这本书，就要求他们订阅你的简讯。
- 要求新的LinkedIn群组成员成为你社区中的一员。（IEEE组织也使用了这个策略。）

- 当群发电子邮件时，使用实时A/B测试：先给3　000个订阅者发送3个不同的标题行，然后将那个点进率最高的标题行发送给其余的订阅用户。
- 确定哪些模式在标题行能起到良好效果，如下。

- 行业标杆企业的研究报告。
- 案例研究，成功经历。
- 关于最近在知名新闻媒体发表的十大文章公告。
- 大型企业的招聘广告和薪资调查，来自顶尖大学的数据科学课程。
- 不反复使用相同的、显著的关键词，因为它没有效果。（每次都换一个不佳的主题标题，比滥用好的标题效果更好。）

最终，邮件列表管理供应商告诉我们，打开率远高于平均水平，这些供应商既有小客户也有百万级别邮件列表的客户，客户数目多达60 000！

# 在7天内优化关键词广告宣传活动

SEO（搜索引擎优化）与SEM（搜索引擎营销）公司会定期聘请数据科学家，所以你需要掌握如何建立起可靠的、通用的SEM原则。这里有一个基于谷歌分析的关键词表现详细分析的应用示例，如下。

1. 根据业务范围，确定访问量大、针对性佳的前10名关键词。这些就是你的种子关键词。

2. 建立投标关键词的大列表：使用谷歌工具来识别与你的种子关键词相关的“相关关键词”。

3. 继续寻找与上一步骤中得到的关键词有关联的关键词，以发现更多关键词。

4. 使用谷歌AdWords工具查找网页上的关键词，最重要的是，要查找竞争对手网站页面里的关键词。当你管理广告组时，看一下“查找关键词”和“添加关键词”下的网站选集。只选择中高量级的关键词，并放弃那些竞争激烈的关键词。（谷歌会在同一页面上提供竞争统计和量级统计。）

5. 创建一个活动，然后为这2 000多个关键词添加一个广告组。在活动设置里选择目标市场（例如，“北美+欧洲”以及“谷歌搜索+合作伙伴搜索”）。设置一个是你意向价格三四倍的最高出价。

6. 创建四五个不同的广告，这样其中至少会有一个拥有高点进率，且会提高点击量。

7. 4小时后，检查流量报告，暂停所有CTR<0.20%以及曝光次数>400的关键词。这样会提高点进率，进一步提高广告活动效率，并最终提高投资回报率。之后，把最高出价降低10％，因为你可能会太快花完你的预算。

8. 第2天：再次检查统计数据，删除低点进率的关键词，并把最高出价减少10%。

9. 第3天：再次检查统计数据，删除低点进率的关键词，并把最高

出价减少10%。然后把点击量大跟点击率高的关键词分离出来，并把它们放在不同的广告活动或广告组中。如果你每天的预算仍花费太快，可以通过排除一些国家、取消显示广告或其他流量质量较低的来源，以提高针对性。

10. 第4天到第7天：按照第三天继续进行，但不要过多减少你的最高出价，如果你每次点击费用（CPC）下降过快过多，那么你的活动就会失效。

11. 到第8天，你应该有几个广告组了，每组可能有50个关键词，它们表现稳定，且非常适合你的网站，另外，还有一个巨大的广告组（第1天到第7天剩下的），将继续稳定运行，但需要不定期监测。

# 自动新闻提要优化

这是数据科学的另一个应用领域。每年投入到在线广告活动（展示广告和点击付费广告）的费用达数十亿美元，以吸引网站访客。那些网站之所以能吸引大量游客，靠的就是优质的内容。

制作内容优质的网站既费时又费钱，所以几乎每个网站发布者都希望其网站的内容创作过程可以自动化。此外，在正确的时间显示正确的内容给正确的用户，将会显著增加广告收入。分析技术有助于找到相关且针对性强的内容。

例如，在Data Science Central上，我们已经实现了自动化内容创作的第一步。我们的新闻提要系统会从关键词相关的谷歌新闻、指定的Twitter和博客，以及我们自己的原创内容中收集新闻。创建这样的新闻聚合系统无须编写任何代码，靠各种小工具、Twitter feed以及Feedburner就可以完成。我们把分析新闻刊登在AnalyticBridge、LinkedIn以及许多其他地方并实时更新，这样给我们的网站带来了不错的流量。

较大的公司正在招聘分析专家来优化这些流程。对于网站发布商以及利用在线广告赚取营收的人来说，内容就是王道。我的第一个实验已经证明，要想增加广告收入，优化内容及新闻要比优化广告活动作用更

大。增加收入有如下两个来源。

- 更多的流量和更多的老用户，也就意味着更高的页面浏览量和更多的转化。
- 当想要制作新闻和内容吸引和留住对广告主的产品和服务感兴趣的用户时，最重要的还是要有更相关的流量、更高质量的流量和更高的转化率。

事实上，Data Science Central是一个社交网络，但这点没有在整个系统动态中发挥主要作用（假设垃圾邮件控制一直在有效进行）。关键还是内容质量和动态内容，因此我们专注于实时新闻和一小批备受尊重的数据科学家的推文。有了这个战略，收入自然而然会增加，社交网络也可以变成营销网站。

# 用**bit.ly**进行竞争情报分析

竞争情报分析是商业智能的一部分，是市场研究的一个子领域。bit.ly可以用于查看自己或竞争对手所发布网页的受欢迎程度。它还可以让你看到（你自己的网页，或作为你营销活动一部分所推广的外部网页）对一个网页贡献的流量，并跟其他人或竞争对手所贡献的流量作对比。

bit.ly是一个域名缩短服务（和tinyurl.com一样），在LinkedIn的电子邮件群发中，bit.ly作为标准工具来缩短带多个标记的长域名。（原因在于LinkedIn群发邮件只允许纯文本。）然而，bit.ly提供的服务远远不只是重定向服务，它也具备有趣的报告功能。

如果你了解它简单而强大的仪表盘，就能明白它的报告功能是大数据应用程序的极佳例子。这也是一个典型的大数据案例，但除非你知道它的特点以及如何去利用它，否则它是无用的。

以下是一个例子。如图6-2所示，这是一个bit.ly仪表盘的截图。

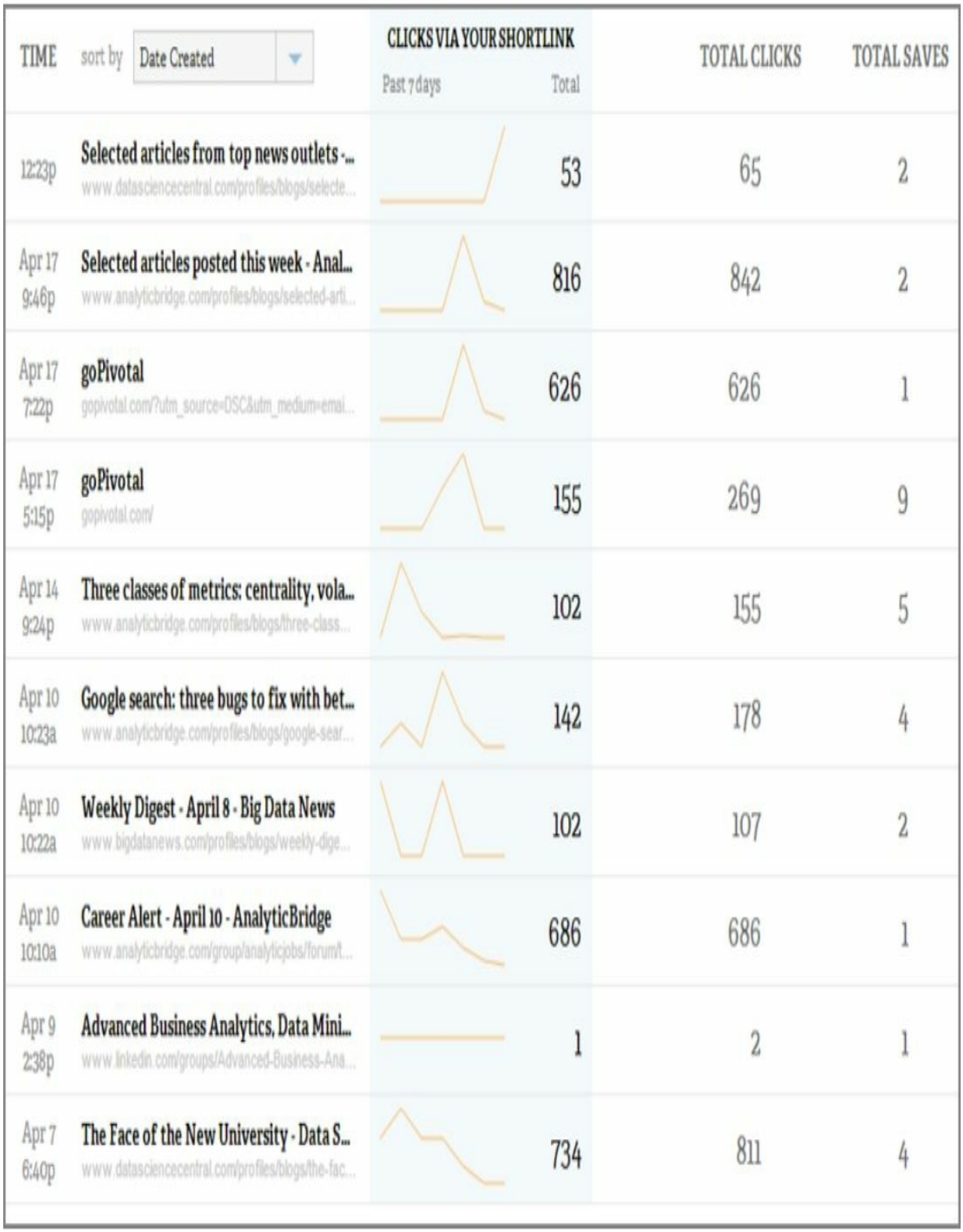


图6-2　bit.ly仪表盘

你可以从CLICKS VIA YOUR SHORTLINK（短链接点击量）找到bit.ly重定向去的流量总量（页面浏览量）。请记住，同一个链接可能被其他人用短链接缩短了。这就是有一个额外标记为TOTAL CLICKS（总点击量）的列的原因，用来告诉你重定向到原始URL的总流量，包括你带去的，以及其他人使用bit.ly缩短相同URL带去的流量。

注意　其他重要的竞争情报工具，除了bit.ly以外，还包括Quantcast.com、Compete.com和Alexa.com（但Alexa是最不可靠的）。

## 通过来源比较来评估流量

对于网站发布者或媒体公司来说，通过比较来评估流量很有用。比如下午5:15创建的GoPivotal链接（见图6-2），155次点击来自自己，另外114次点击来自竞争对手。这意味着我给这个内容引流的效果比竞争对手好。当然，不是所有人在他们的营销活动里都使用bit.ly，但很多人会使用它，因为bit.ly能向客户提供按推荐人、国家和每天进行分类的流量统计。具体做法是只需简单地添加一个+符号在缩短的URL里（如`https://bit.ly/11xTDUD+`）。

注意，在下午7:22创建的goPivotal链接包含特殊的专有标签，因此要分别进行跟踪。（尽管指向了同一页面，但bit.ly的URL是不同的。）这就是第三方的点击数为0的原因。

## 评估共享你链接的人的流量

继续看图6-2，“Three classes of metrics（三类指标）”的链接是Data Science Central网站内部的，所以我知道有额外的53次点击（53=155–102）来自那些使用bit.ly分享文章的人。在bit.ly仪表盘上点击“Three classes of metrics”，你就会知道是谁分享的链接，以及他们产生了多少流量（在这种情况下，是53次点击）。此信息让你能够追踪谁可以帮助你的网站产生有效流量。

## 实时**bit.ly**

实时bit.ly仍然是一个测试项目（你可以访问`http://rt.ly/`查找更多内容），但如果你使用实时功能，它会允许你通过A/B测试实时微调邮件推广活动。例如，如果你想发送一封电子邮件给10万个订阅者，你会像如下这么做。

1. 将A版本发送给2 500个联系人，将B版本发送给另一组2 500个联系人，均为随机选择。

2. 等待10分钟（有可能更短），看看哪一组表现得更好（比如B表现更好）。

3. 将B版本发送给剩余的95 000个联系人，以开展整个活动。

例如，A与B之间的区别是主题行，且成功的指标是打开率或点击率。例如，A的主题行=“每周文摘，4月22日”，B的主题行=“DSC最好的文章”。请注意，通常在进行邮件营销活动优化时，你并非只会选择优化某一个活动，而是会优化你所有的活动。如果只需要优化一个广告活动，那么B的主题行会更好些。对于全球性的长期优化，A则更胜一筹。

# 测量**Twitter**哈希标签（**hashtag**）的收益

拥有大量Twitter粉丝，并且明智地使用带#号的哈希标签，是增加流量的法宝，并可以从社交媒体中获益。关于这个主题的一个很好的参考是Mashable的一篇文章如何从Twitter的哈希标签中获益最大（*How to Get the Most Out of Twitter #Hashtags*，）可以在`http://mashable.com/2009/05/17/twitter-hashtags/` 上找到该文章。

但实际情况非常不同。首先，你上一次阅读推文是什么时候？可能是很久以前了。此外，哈希标签（#标签）往往不利于人们点击你的链接。在实际情况中，即使有6 000个真正的粉丝，对于一篇推文来说，最坏的情况是0次点击，最好的情况是20次点击。如果你只计算来自你自己所发推文的点击数，而忽略间接流量和下文描述的乘数效应，那么实际情况就会是上述那样。

下面链接指向的是一个有两个#标签推文的例子，如果你没有见过可以自行查看：

naXys专注于复杂系统的研究：http://ning.it/12EFrsc #bigdata #datascience

## 它真的有那么糟糕吗

不是，实际上它是有效的。首先，如果你的推文被转发，它就很容易获得大量的点击，这可以归因于Twitter（以及更多其他的来源），使得像数据科学一样专业的学科内容也可以得到大量点击。Twitter可让推文产生高达3倍的阅读量。

例如，在我们Data Science Central的每周文摘上发表了一篇好文章，eBlast将其推送给60 000多人后直接获取了800次点击。12个月后，这篇文章被浏览了12 000次。分析数据时，你会发现总访问量中有2 400可以归因于Twitter（人们通过Twitter转载和分享你的链接）。

也有一些其他积极的副作用。一些发布商运行Web爬虫以发现、提取信息和对信息分类，#标签就是其检测并对新闻分类的工具之一。Data Science Central也算是其中一员——网络爬虫和运行网络爬虫的人。同样的发布商也会发布顶级数据科学家的名单。这些名单是基于在特定的#标签下受欢迎的那些数据科学家，而不管所考虑的推文是自动生成（内容同步）还是人为生成的。

当一条推文都是特殊字符时，对一般的互联网用户会具有威慑作用，而对于极客们则很少有威慑作用。（如果你读了这本书，很有可能你就是极客，或者说多少算是一个极客。）但是，用#标签时仍然必须遵守以下规则。

- 每条推文不超过两个#标签。
- 两个#标签是相关的。
- 50%的推文不带#标签。

否则，没有人愿意再看你的推文，你要面临的挑战就是让新粉丝的增加快过老粉丝的流失。

#标签特别适用于网络讨论和评论，因为它有直播的性质：INFORMS可以使用#INFORMS2013的#标签供人们评论其年会。#标签也有助于扩散或讨论某些事件。例如，我们创建了#abdsc来标示和感谢

在推文里提到我们的博主。

不过，发推文时，相比上到全民排行 （*All People*） 榜，它不会那么容易就上热门（*TOP*）榜 。如果你的推文不在热门中，那你的#标签就没有价值。此外，你可能成为#标签劫持的受害者：#标签的恶意用户滥用流行的#标签来推广自己的东西。

图6-3通过bit.ly跟踪Data Science Central上一些最近的帖子，从中可以看出一篇博客文章很容易有3倍流量统计（最右边的一列），这要感谢其他喜欢你所写的文章的人，以及把你的文章重新转发到他们博客上的人。可以注意到，内部点击越少，乘数越大：当我们并没有怎么推广自己的帖子时，其他博主会替我们推广它。图6-3中最极端的例子是一个内部点击乘数达到53。

| TIME | sort by Date Created | CLICKS VIA YOUR BITLY LINK | | TOTAL CLICKS |
|---|---|---|---|---|
| | | Past 7 days | Total | |
| May 26 11:42p | New pattern to predict stock prices. mul... www.analyticbridge.com/profiles/blogs/new-pattern... |  | 75 | 183 |
| May 26 1:23p | Reverse clustering for tactical warfare - ... www.analyticbridge.com/profiles/blogs/reverse-clu... |  | 47 | 70 |
| May 26 12:29a | La solitude du thésard de fond - Héloise ... www.sciencesdumanoir.com/la-solitude-du-thesar... |  | 4 | 93 |
| May 24 9:56p | Why is it so hard to count this way? - An... www.analyticbridge.com/forum/topics/why-is-it-so-... |  | 216 | 216 |
| May 23 9:26a | Big Analytics 2013 Atlanta - Confirm Ide... www.event.com/events/big-analytics-2013-atlanta/... |  | 472 | 472 |
| May 22 10:01p | Data Scientist Demographics - Data Scie... www.datasciencecentral.com/profiles/blogs/data-s... |  | 69 | 257 |
| May 22 5:54p | Data Science eBook by Analyticbridge - ... www.datasciencecentral.com/profiles/blogs/data-s... |  | 304 | 387 |
| May 22 2:12p | Visual analytics for everyone \| Tableau S... www.tableausoftware.com/trial/tableau-software?... |  | 1 | 53 |
| May 22 12:48p | Now, choose a right career for your chil... www.business-standard.com/article/careers/no... |  | 8 | 40 |
| May 18 4:50p | How to choose an analytic tool? - Data S... www.datasciencecentral.com/forum/topics/how-to-... |  | 27 | 69 |

图6-3　通过社交媒体实现点击量成倍增长

## #标签对你有用吗？

检查#标签是否有用的一个简单方法是进行一些A/B测试。你可以使用以下协议。

1. 在超过14天的时间段里，每天推送两条链接：中午一条，下午6点一条。

2. 第一天，早上把标签# datascience加到推文里，傍晚把那条不加#标签的内容加到推文中。

3. 第二天反过来：中午的推文不加#标签，下午6点那条加#datascience标签。

4. 保持中午和下午6点交替，以确定何时合适添加标签。

5. 附加特定的查询字符串给各个链接，例如，在推文的网址上附加`?date=0528&hashtag=yes&time=6pm` 到你推文里（在生成该网址的缩短链接之前）中，这样可以起到追踪的作用。

6. 使用**bit.ly** 来缩短你的URL。

例如，你所发布的推文（带#标签的版本）可能像如下这样。

测试Twitter#标签的性能
`https://bit.ly/11wpSXi#datascience#hashtagABtest`

请注意，当你点击`https://bit.ly/11wpSXi` 的时候，查询字符串会显示在你浏览器中的原始URL上（在重定向之后）。更重要的是，通过查看`https://bit.ly/11wpSXi+` （相同的bit.ly URL，只是末尾添加了+），你可以监视自己获得了多少次点击。

20天后进行统计分析，来验证以下数据。

- #标签可以增加直接点击数。
- #标签会提高之前讨论过的乘数效应。

如果点击变化能通过时间、周末/工作日、链接、#标签的积极促进作用等因素解释，则可以检测并量化这方面的贡献。

### 实时检测有病毒的推文

*Micro-meme* （微模因）指的是因为创建#标签而催生、在流行一些时日后会消失的话题。那么，如何检测和利用它们呢？内容撰写者和内容发布商对此特别感兴趣。

TrendSpottr和Top-Hashtags都是提供这类型服务的网站。但是，它们靠的是样本数据，主要的缺点是难发现新的、罕见的突发事件。由于这些模因（memes）使用一些之前并不存在的新创出来的关键词，如#abdsc或# hashtagABtest，所以使检测变得特别困难。

此外，我上次检查发现，Top-Hashtag平台上的内容跟Instagram关联大，跟Twitter的关联小。#标签在谷歌+上也有效。对于数据科学而言，最有趣的大型社交网络包括LinkedIn（更专业）、谷歌+、Quora（更偏极客）以及Twitter。因此，实时地识别正出现在这些网络上的新趋势，是很重要的。

有一种解决方案，就是关注你所在领域中最有影响力的人，每次当他们的#标签下获得10条以上推文时，你将会得到特别提醒。如果一些普通人创建的新#标签开始制造大量的热度，会被这些有影响力的人（至少一部分人）很快注意到，你也会因为关注这些人而注意到。你需要选择一批正确的关键词和#标签以确定最有影响力的人：数据科学、Hadoop、大数据、商业分析、可视化、预测建模等。最有影响力的人物列表是公开的。

以上的讨论包含两个关键信息：1）目前，从干扰信息中检测信号以及对推文分类的良好算法仍有待开发，这些算法涉及数据科学和自然语言处理；2）这样的算法对网站发布者、内容管理者和内容撰写者特别有用。

# 用**3**个修补方法提升谷歌搜索

下面这些大数据的问题可能对很多搜索引擎有影响，也就是说新创业的公司还有机会创造更好的搜索引擎。数据科学家们有足够的能力来识别这些问题，并定量衡量它们是大问题还是小问题，进而区分这些问题的优先次序。这些问题可以通过改进的分析方法和数据科学来解决。以下是各个问题及其解决方案。

## 过时的搜索结果

在搜索结果中显示最新的网页或最近更新的网页，谷歌在这块做得一般。当然，最新的网页不一定意味着其质量就高，谷歌的算法倾向于将排名靠前的旧页面放于搜索结果靠前的位置——也许是因为新网页的排名缺乏可信度或历史记录较少。（这就是我创建统计分数，以便给没有历史记录的网页进行排名的原因。）为了解决这个问题，用户可以在谷歌中搜索的时候加上最新年份（比如2013），也可以通过默认设置添加。例如，把查询数据科学 与查询数据科学 2013的搜索结果相比较。你更喜欢哪一个？如果能做得更好，谷歌应当允许你在搜索时选择“最近”或“永久”的搜索结果。

问题在于要给网页附上正确的日期，这并不简单，因为网站管理员可以使用假时间戳来欺骗谷歌。但又因为谷歌每隔几天就会给大多数页面编索引，所以可以创建一个谷歌时间戳，为每一个（静态）网页保留两个日期：首次编入索引日期和最近修改日期，这样做还是比较简单的。另外，你还需要为每个网页保留一个128位的签名（除相关关键词以外），以方便检测其修改时间。对于动态创建的网页，该问题就更加困难了。

## 错误的文章归属

你在博客上写了一篇文章。然后这篇文章被另一个媒体，如《纽约时报》收录。谷歌会在顶部显示《纽约时报》版本。有时即使使用完全匹配的关键词搜索文章标题，原始版本也根本不会出现在搜索结果中。你可能会认为《纽约时报》比你那名不见经传的小博客更值得信赖，或者是你博客的网页排名太低了。但这个问题有两层潜在含义：它带来了糟糕的用户体验；另外，对于谷歌用户而言，互联网看起来要比实际小得多。

网站管理员可以使用有失公平的策略来打败谷歌：用智能程序和正确的关键词来制造大量有效的谷歌查询，但只给自己的链接产生点击。

这将降低竞争对手的实际点进率，提高自身点进率。这种“攻击”有望解决你的内容归属问题。

还有一种更粗野的点进率攻击方式，利用谷歌的有效点击（谷歌重定向）打开问题网站但不允许谷歌广告曝光。这样做可以欺骗谷歌的算法，使之以为该网站在操纵点进率，涉及点进率欺诈，从而导致谷歌删除对这些网站的收录。同时，广告主看到的是糟糕的“曝光到相关载入页转化（点出）”比例，对广告显示造成特别严重的影响，从而使得广告主放弃该发布商。总之，这种策略可以同时借助两种手段来自动处理侵权的行为：点进和点出。它必须依次地（以避免被检测到）使用违规手段，直到页面被取消收录。如果执行恰当，没有人会知道攻击者是谁，甚至不会有人察觉发生过攻击。

要解决这个问题，谷歌可以使用一个简单的方法，有如先前段落所述，需要正确识别文章的首个版本。

## 支持不相关的网页

谷歌每个星期都为各个网站带来大量的搜索结果曝光机会，而且这个数字是非常稳定的。该数字可能源于网页数目、关键词和所考虑的网页人气（页面排名），以及很多其他指标（加载时间、原创内容的比例、专用还是通用网站等）得出。如果每个星期谷歌都给你的网站展示10 000次，那么谷歌会倾向于哪种类型的页面或关键词匹配呢？

按理说，它应该会偏向跳出率低的页面。但在实践中，它却正好相反。

为什么呢？也许如果用户不觉得你的网站有趣，他会执行更多的谷歌搜索，那么点击付费谷歌广告的概率也就增加了。（你的载入页）用户体验如此糟糕，但谷歌却因此增加了收入。

对此你该如何做出修正？如果大多数用户在某个网页上花费很少的时间（谷歌可以很容易地计算出所花费的时间），该网页（或更好一点，该网页或关键词的组合）在谷歌的索引便应受到惩罚，出现得更少。很多发布商也使用谷歌分析功能（GA），向谷歌提供了有关跳出率和页面级用户兴趣的额外宝贵信息。

然而，你可能会认为，如果跳出率很高，也许用户通过访问你的首

页，会很快找到搜索的答案，因此用户体验实际上应该很好。对此我并不同意，因为在Data Science Central上，每个页面都带有链接显示类似文章，通常带来额外的页面访问量。事实上，最糟糕的跳出率是与谷歌搜索相关的。更让人不懂的是，谷歌的跳出率变得越来越差了（虽然所有其他的流量来源都变得更好了），可能因为谷歌算法好像缺乏机器学习能力，又或者非常不擅长处理那些每天新增的页面。在将来，把长的文章分解成两页或三页，将很有可能改善谷歌搜索（和其他来源）跳出率问题。

# 改进相关性的算法

搜索引擎广告等在线广告主要关注于广告的相关度问题。当你在搜索结果页面或其他地方看到广告时，显示在你的眼前的广告（应该）是精心选择的，以此来最大限度地提高你替发布商或搜索引擎带来转化及广告收入的机会。

如果你认为看到的是不相关的广告，那么要么是广告出价便宜，要么就是谷歌的广告相关性算法运行效果不佳。有时是由于广告费太少，有时是由于评分方法太差，你可能会认为，有时是因为广告主在关键词选择和针对性方面做了很糟糕的工作。

广告评分算法曾经非常简单，分数是广告主最高出价和点进率相关的函数。这导致了滥用：广告主可以生成虚假的曝光来稀释竞争对手的点进率，或点击自己的广告来提高自己的点进率，或两者兼有。他们通常使用代理服务器或僵尸网络来掩护，相比搜索引擎在点进率上获得不公平竞争优势。要注意的是，搜索引擎是使用流量质量进行评分的。事实上，我作为数据科学家曾经受聘对其中一些流量质量算法进行逆向工程并执行压力测试，进而探索这些评分算法失败的原因。

最近，除了点进率和最高出价，广告网络把广告相关性加到他们广告评分组合算法中（该算法用来确定你将会看到什么广告和广告怎么排序）。总之，广告网络不想显示会让用户感到沮丧的广告——它会改进用户体验并减少用户流失，以增加长期利润。那么，广告相关性评分是如何进行的呢？

在这个组合算法中有3个组成部分，如下。

- 用户访问一个会显示广告的网页。
- 一个会显示广告的网页。
- 广告本身，包括它的文本或视觉内容，或其他指标，如大小等。

第4个重要的组成部分是载入页，但该部分在这个简短的讨论中不考虑。（好的发布商会利用广告主的载入页来检查文本广告和载入页之间的匹配，去掉不良广告软件）。

## 解决方案

你将需要创建3个分类法，如下。

- 分类法A：根据已注册用户的兴趣、会员资料或网络搜索历史，对用户进行分类。
- 分类法B：根据内容或发布商提供的关键词标签，对显示广告的网页进行分类。
- 分类法C：根据文本广告内容，或广告主提供的描述广告的关键词（如在点击付费的广告竞价关键词或广告标题）对广告进行分类。

除非广告显示在通用的网页上，这时分类法A比分类法B更重要，否则分类法B和C更重要。因此我们先忽略分类法A。目标是对分类法 *B* 跟分类法 *C* 进行类目匹配 。分类法可能有相同的类别，也可能没有，所以一般会进行模糊匹配。例如，在分类法B中，显示广告的页面附加到金融或证券市场类别；而在分类法C中，广告会附加到投资或金融类别。所以你需要有一个系统，以测量属于两个不同的分类法的类别之间的距离。

## 如何建立一个分类法

市场上有许多供应商和开源解决方案，但是如果你想从零开始建立你自己的分类法，这有一个办法，步骤如下。

1. 搜罗网站（DMOZ目录有数百万提前分类好可免费下载的网页，这是一个很好的起点），在同一网页上摘要一些文本片段，并创建一个

测量邻近文本片段之间的距离。

2. 清洗关键词并抽取词干。

3. 利用搜索查询日志数据：在同一会话中找到比任意查询更接近彼此的两个查询（基于前面提到的距离）。

4. 创建一个包含所有摘要的“文本片段”的表格，且文本片段是相关联的（例如，在相同的网页上或同一用户会话中找到的）。表格中至少需要有1亿对这样的文本片段。

假设（X, Y）就是这样的一对，那么计算$n_1$ =表格中X的次数，$n_2$ = 表格中Y的次数，再计算$n_{12}$ =X和Y的关联次数（例如，在相同的网页上的关联次数）。衡量X和Y间关联程度的指标是$R=n_{12}$ /SQRT（$n_1 \times n_2$ ）。有了这个不相似度的指标（在例如 `http://www.frenchlane.com/kw8.html` 使用），你可以通过层次聚类对关键词分卒，并最终建立一个分类，这无非是一种手动分配标签给（假如）前20个簇的一般关键词的无监督的簇，每一个簇代表一个类别。

# 广告循环问题

例如，当广告主使用相同的登录页和相同的广告词关键词组（在谷歌购买的关键词）的5个版本时，应该如何使5个广告频繁地出现呢？从谷歌的角度看，最佳效果是通过广告提供达到每个印记的最高收入，无论广告主是按被曝光次数（CPM模型）收费，还是按点击次数（点击付费）收费。对于广告主来说，每次点击成本（CPC）最低的广告有时是最佳的选择，虽然它可能只产生很少的点击。

然而，谷歌不能只简单地在100%的时间显示每个展示收入最高的广告。这个指标是随时间而变化的，在2周之内，最佳广告可能会改变，因为用户会因为总看到相同的广告而感到无聊，或者广告主购买了新的关键词。所以谷歌将半随机地显示5个广告。最佳广告显示会更频繁，最差的广告则非常罕见。

但是，你是如何定义“更频繁”和“非常罕见”的呢？当然，这取决于

展示击率和每次点击成本（CPC）的乘积。有趣的是，对谷歌来说，广告显示频率和其CTR（展示点击率）之间是对数关系。在谷歌的模型中忽略了CPC，至少在这个案例中忽略了。

你对谷歌选择的对数分布有何想法？它是：

- 随机的？
- 巧合的？（更多测试将表明，发现的这个关系是错误的，只是在特定的广告活动上偶然出现。）
- 谷歌数据科学家的深度调查结果？

为什么没用上CPC？也许是为了避免欺诈和广告主的操纵。

如果你在`http://www.analyticbridge.com/group/webanalytics/forum/topics/a serving-optimization` 上核查数据，你将看到最佳广告在76.29 %的时间里都有出现，第二名的时间占比是10.36%，尽管两者都有相同的CTR和CPC。实际上，这其中没有考虑CPC，只用到CTR；涉及CTR的模型，其$R^2$ （拟合优度）比较糟糕。对数模型以外的模型中，$R^2$ 大都比较糟糕。

# 杂项

本节讨论在各种其他情况下一些数据科学的实际应用。这些应用在现今的市场中会经常遇到。

# 简单模型会获得更好的销售预测

一个普遍存在的挑战是如何做出更准确的销售预测。解决方案是利用外部数据，简化预测模型。2000年，我在GE公司的分析团队工作，以提高NBC网络（美国全国广播公司旗下的门户网站）的销售预测准确度。销售或财务人员利用一个基于前1个月销售的初级公式来预测下个

月的销售量。在GE工作期间，我开始开发更复杂的模型，其中包括时间序列技术（ARMA——自动回归模型），考虑了季节效应，但仍然完全基于内部的销售数据。

至今，许多公司仍然没有使用正确的方法来得到准确的销售预测。这对于正在成长或衰退行业的公司，或者因宏观经济、结构因素而波动性高的公司来说，尤是如此。事实上，我为GE选择更复杂模型的措施是错误的：它确实提供了小的升力，但未能产生巨大的且有望上正轨的升力。

## 正确的方法

大多数拥有超过200名员工的公司，使用独立的竖井式软件来存储和利用数据。财务、广告/市场营销以及经营/库存管理/产品部门的数据或多或少都是独立的，很少合并在一起以提高投资回报率。更糟糕的是，这样做完全忽略了外部的数据源。

每个部门有它们自己的数据竖井：在商业智能部门内，数据的付费搜索、有效搜索和其他广告（印刷物、电视等）均由每个数据分析师单独处理，他们之间没有互相交流。从商业的角度来看，付费搜索（由SEM人员管理）、有机搜索（由SEO人员管理）以及其他广告，其提升指标显然是彼此高度依赖的，但实际操作中，这些板块彼此之间并无互动，不同的渠道采取单独优化措施，而非共同优化。

如果销售结果来自于3个月前看到的电视广告，以及上个月在谷歌上的两次广告，或因为好的SEO和优化，那么将无法准确地把销售金额归功到各个参与销售的管理人员身上。更糟的是，销售预测没有使用外部数据和计量经济模型。

对于新兴的创业公司（或推出新产品的老公司）来说，同样重要的是要采用自回归时间序列模型准确地评估销售增长，该模型要考虑到广告花费和时间衰减函数。在NBC网络示例中，发现电视广告的影响约为6个月。有一个简单但效果不错的模型，如下。

$$\text{Sales}(t) = g\{f(\text{sales}(t-1, t-2, ..., t-6), a_1 * \text{SQRT}[\text{AdSpend}(t-1)] + ... + a_6 * \text{SQRT}[\text{AdSpend}(t-6)]\}$$ [1]

其中时间单元是1个月（实际上28天更合适），另外g和f是函数，需要通过交叉验证和模型拟合技术来确定（f函数与前面提到的ARMA模型相一致）。

以下是影响销售的一系列因素。

- 任何基本的销售测试都应该包含定价优化（包括销售预测模型基本价格的弹性部分）、客户反馈、新产品上市以及客户流失。
- 销售预测应当与外部数据整合，特别是以下几部分。

- 市场份额的趋势：市场份额在上升还是下降？
- 行业预测（你所处的行业呈增长还是下降趋势）。
- 基于竞争对手的数据，你所在行业的总销售量。（所考虑的数据可以很容易地在销售竞争情报的公司购买到。）
- 主要竞争对手的相同产品的价格，特别是价格比率（你的产品价格对比竞争对手的产品价格）。
- 经济预测：一些公司出售这种类型的数据，但他们的统计模型有缺陷。他们要么使用过时的数据，要么使用最新数据很慢地对模型进行更新。

## 一个简单的模型

在我建议的那些指标里，确定能推动销售的前4个指标，（不要忽略外部数据来源，包括产品的情绪分析指数，也就是说客户在Twitter上写的关于你的产品的意见），并建立一个简单的回归模型。你可以使用Excel来建立该模型（使用数据分析插件或线性函数），并获得更好的预测结果，而不是只基于自己很多数据竖井中的内部数据，去用更复杂的模型实现。

## 如何招聘一个优秀的销售预测员

你需要招聘分析敏锐型的管理顾问，这个人需要与公司的所有部门进行互动，收集、合并和分析来自大部分竖井的所有数据来源，并能与管理人员以及公司里负责数据竖井的每个人进行沟通。他将会推荐较好的解决方案。交流的内容应该包括数据质量问题、哪个指标你应该在未

来继续追踪，以及如何为管理人员创建有用的仪表盘。

# 更好的医疗欺诈检测

华盛顿教育协会（WEA，在华盛顿州）与验证公司Aon Hewitts（在伊利诺伊州）一起合作，以消除特定类型的医疗保险欺诈：老师把不合格的人（比如没有健康保险且失业的朋友）申报为自己的亲属。“好”人（如教师）使用该欺诈手段，把没有保险的人申报为配偶或子女，为他们提供医疗保险。

有趣的是，我看到了发送给所有华盛顿州老师的信。信中要求他们填写大量的文字信息，并提供多个身份证明（税单、出生证明、结婚证等），类似于在公司上班所需要的ID文件（I9表单）。

这些必须要邮寄给核查公司的纸质文件是很容易作假的（例如，伪造假的出生证明或干脆声称没有此证明等）。此外，要求人们填这么多的文字信息浪费时间和资源（树木用于生产纸），且会导致大量的错误、隐私问题和ID盗窃的风险，会浪费华盛顿州很多钱。

那么他们为什么不使用现代方法来检测欺诈呢？例如，用数据挖掘技术来检测可疑的社保号码（SSN）、基于IRS（美国国税局）的数据来确定那些被多个家庭申报为亲属的社保号、没有出现在任何税单里却提交到了国税局的社保号、匹配失败的家庭地址等。请注意，虽然5天大的婴儿在国税局数据库中可能没有记录，但如果出于税收或健康保险的目的，他是有资格成为亲属的。

为什么不使用数据挖掘技术代替纸张获取信息的方式，来获得数据挖掘带来的所有优势呢？纸张的优势在哪呢？我没有发现纸张有任何优势。

# 归因模型

我曾应邀为NBC进行一项研究。它利用不同时间段电视节目每周的

组合，播放广告，以宣传其互联网网站。我曾被要求去确定哪些电视节目能够最有效地推动新用户去注册网站。每个星期，它使用的是编号10到12的组合，播放20个不同的节目。收集6个多月的GRT数据（在特定的时间看特定电视节目的人数），以及每周新用户的数量。A/B测试是不可能的，所以问题相当复杂。

下面是我解决这个问题的过程。我计算了所有双周的距离{A，B}，用到的数据是几周的时间里播放了哪个电视节目，以及每个星期的总GRT数据（高GRT数据会带来许多新增用户，除非电视广告真的很差）。然后我查看了双周的数据，它们非常相似（在GRT数据和节目组合方面）而且会很快关闭（不超过2个月）。我研究了双周，A周比B周多一个节目，比如节目S。B周没有S，但另外有一个不和A周共享的节目，比如节目T。实际上，比这种情况还稍微复杂些。可以不止一个非共享节目，而是两个不同的非共享节目。但为了简单起见，依然以此为例：节目S在A周，而不在B周；节目T在B周，而不在A周。如果在A周有更多的新用户，我会调整节目S和节目T的性能指标，如下：P（S）=P（S） + 1, P（T）=P（T） – 1。与B周相比，A周有较少的新用户的情况则相反。

要确定最佳的表演节目，剩下所要做的是按其性能指标进行排名。《法律与秩序》是赢家，并经常在电视节目的组合中使用，以便对网站进行推广。

# 预测陨石撞击

这个例子可以作为培养数据科学家课程的一部分，用于培训。本节讲述了一个典型的真实分析所涉及的各个步骤：从理解问题（如何预测陨石撞击）到提供答案。

结论是基于人工估计的，但在培训的背景下是讲得通的。本节还描述了顾问专家为这个分析会要价多少，证明数据科学家不一定昂贵。

统计分析得出的结论和你最近在媒体上读到的是不同的：根据记者们的报道，俄罗斯流星每40年发生一次。

以下是统计分析的8个步骤。

1. 确定分析的范围。这是一个小项目，完成至多需要10个小时，收费为100美元/小时。提供每年每类尺寸的陨石撞击的风险分析。

2. 确定数据和注意事项。你可以在`http://osm2.cartodb.com/tables/2320/public#/map` 上找到并下载数据集。这项分析不包括那些已经确定的陨石——不会撞击地球但会偶尔飞过的陨石数据。确定和预测这类陨石的数量属于另一个类似项目的范畴。

3. 数据清理：数据看起来全面，实则比较混乱。数据库的更新似乎需要花费数年时间（2000年到2010年的陨石数量比过去10年少很多），某些年份的陨石数据很少。陨石的数量在1974年左右开始激增，可能是因为那年开始收集数据的缘故。数据中有5个关键词段：陨石的类型、尺寸、年份、陨石坠落时间、陨石发现时间。这里没有考虑的字段是位置。并丢弃1900年之前的数据。

4. 探索性分析：可以发现一些明显的模式，尽管存在以下几点不足。

- 混乱的数据。
- 近年来存在观测结果缺失或数据误差。
- 巨大的陨石是非常罕见并且非常大的，每年的吨数估计有极大的起伏。
- 事实上，大多数陨石是在坠落后被发现的（而不是正在坠落时被看见），所以“发现”和“坠落”之间的时间间隔未知。“发现的陨石”有时只是一个碎片，而不是整个岩石。

分析出的模式如下。

- 因为越来越多的地方有了人烟，加上仪器的进步，所以现在发现了更小的陨石。
- 陨石落到地面的频率，随尺寸变大成指数下降（在数据中以克衡量）。
- 按对数刻度对尺寸分组，按10年为单位进行分组，都是有道理的。

5. 实际分析：你可以在以下网址中查看含有数据和公式的Excel电子表格，http://api.ning.com/files/IJeLHwAUIk-Wn2e2drRoYuImASMAh2eXuk*ru-II-bgsv8nykgecq*x-j7oe*2v-HJBPVP0A4zTtH6x-YZE3lFuKI*UXYD yJ/meteors4.xlsx 。它有助于将“发现的陨石”的类别包含在分析中（尽管其精度较低）；否则数据集就太小了。数据汇总（按10年和大小分组）在数据集电子表格的分析标签中，使用一个基本的Excel函数，如Concatenate、Countif、Averageif、log和int。

有9组的尺寸至少有一个观察结果：从尺寸0，也就是最小的尺寸，到尺寸8，代表从9吨到66吨重——比如在俄罗斯上空爆炸的那个陨石。由数据集可得，在过去的40年里没有发现尺寸为8的陨石，这也许就是为什么人们声称每40年发生一次2013年的那次俄罗斯陨石爆炸的原因。但在1900年到1950年间记录到5次尺寸为8的陨石，但不知道这些旧数据有多可靠。在这里，用INT（0.5*log m）来计算陨石的尺寸，其中m是陨石的质量，单位为克。

另外是大小比率，是指将尺寸为k的陨石数量，除以尺寸为k+1的陨的数量（比如一个给定的10年时间），基于大小比率建立预测模型。分析这些比率如何变化，以及固定的尺寸k几十年来的比率变化。

针对Y=比率（1950—2010年的汇总数据）和X=尺寸进行快速回归，可得到一个完美的线性拟合（$R^2 > 0.98$），如http://www.analyticbridge.com/profiles/blogs/great-statistical-analysis-forecasting-meteorite-hits 。在图中，我已经排除了尺寸0（它本身有问题），还有罕见的尺寸8和7（在所考虑的时间期间，尺寸8只出现过一次）。所有尺寸的陨石总数为31 690。

我尝试为每个10年的数据进行回归分析（针对大小与比例），试图找到某种模式。我的确找到了一个模式，但用大脑比用电脑更容易检测到这个模式：（非常粗略且平均来看）回归的斜率随时间的推移而下降，这表明最近发现了更多的小陨石，而大陨石（尺寸5以上的陨石）的数量随着时间的推移并没有增加太多。

6. 模型选择：有两个10年段显示出了较好的模式稳定性和时效性，2000—2010年及1990—2000年。同样，尺寸5和4显示出模式稳定性（随着时间的推移，尺寸4增加，而尺寸5相对平缓）。我使用这两个时间段和尺寸进行初步预测，估算斜率，并最终为撞击概率（上限和下限）提

供两个预测。正如所预期的那样，跨度（或置信区间）随尺寸呈指数增长。

在电子表格中白色字体/红色背景的比率，代表对最稳定模式进行插值得到的人工估计（有时覆盖实际数据）。没有进行交叉验证。很多验证工作集中在对比率汇总表中单元格Y1:AG8里的稳定岛进行确定。

7. 准备开始预测，如图6-4所示。

| Meteorite Size | Meteorite Weight | Unit | Occur every... (in years) | |
|---|---|---|---|---|
| 0 | 0 | grams | Upper bound (1) | Lower bound (2) |
| 1 | 6 | grams | | |
| 2 | 54 | grams | | |
| 3 | 402 | grams | | |
| 4 | 3 | kilograms | | |
| 5 | 22 | kilograms | 0.2 | 0.2 |
| 6 | 163 | kilograms | 1.3 | 1.3 |
| 7 | 1,203 | kilograms | 10.3 | 8.7 |
| 8 | 9 | tons | 99.5 | 65.4 |
| 9 | 66 | tons | 1,114 | 556 |
| 10 | 485 | tons | 14,153 | 5,281 |
| 11 | 3,585 | tons | 200,971 | 55,448 |
| 12 | 26,489 | tons | 3,155,244 | 637,651 |
| 13 | 195,730 | tons | 54,270,201 | 7,970,639 |
| 14 | 1,446,257 | tons | 1,014,852,765 | 107,603,626 |
| 15 | 10,686,475 | tons | 20,500,025,853 | 1,560,252,571 |
| 16 | 78,962,960 | tons | 444,850,561,020 | 24,183,914,855 |
| 17 | 583,461,743 | tons | ############### | 399,034,595,112 |
| | | | | |

(1) based on 2000-2009 data

(2) based on 1990-1999 data

9 tons means: from 9 to 66 tons.

图6-4　根据陨石尺寸预测撞击机会

很容易得到如下每年陨石撞击概率的计算公式，这个概率是陨石实际重量的函数。

年撞击概率（重量）=1/（A + B* log（重量））

给你一个练习：基于图6-4估计A和B的值。

根据图6-4中的数据，“每40年”发生一次2013年俄罗斯爆炸的断言看起来是合理的。因为角度、组成成分或速度的因素，俄罗斯的陨石发生爆炸的声响比预期的大。又或者，这个陨石的重量是基于其爆炸强度估计的，而不是周围的其他因素。

在数据集里没有尺寸9、10、11或12的陨石，而且很少有尺寸8的，因此所对应的（这些大尺寸陨石的）撞击风险，纯粹基于内插法计算。然而，它们也构成了这项研究中最有趣和最有价值的部分。

8. 后续工作：详细的分析包括针对陨石类型（铁和水）、角度和速度进行细分预测。此外，撞击风险分析要评估人口增长的影响。在这个小分析中，它实际上是被隐含地考虑在内了。另外，你可以做一些更传统的交叉验证，比如为2013年做一个预测，看看你针对尺寸0～4的陨石的预测偏差有多大。与客户讨论这个问题可以带来更多的顾问工作，且为客户提供更多的价值。

这是一个小分析，并且没有一个真正的指标来衡量这个项目模型的表现情况或投资回报率。有许多项目也是这样，并非这一个项目。对于统计学家和数据科学家来说，做出聪明的猜测和人工预测也是不同寻常的。尽管，这在工程上是很常见的事。

# 在路口停车场收集数据

本部分将表明数据科学适用于许多不同的环境。本部分内容还侧重于实验设计和取样，是统计科学的一个领域，所以应包括在数据科学里。最后，它表明“无数据”并不意味着你不能做任何事情。当数据最初不存在时，数据科学家应该能够告诉你如何（从有助于数据科学的角

度）先创建数据，然后跟踪数据。

路口的停车场因犯罪活动而臭名昭著。这些偏远、毫无警戒的位置，隐藏在茂密的森林中，也吸引了各种坏人。一种比较原始的犯罪行为是小偷盗窃徒步旅行游客的停车费。用户（游客）在一个信封里插入5美元的钞票，放在路口已上锁的信箱中。这些木箱很容易被破坏也很容易被修复，因此，小偷很容易窃取木箱里的5美元钞票。那些聪明的小偷会在每周晚上去10个人比较多的路口停车场，一次偷50%的信封。（如果偷100%会引起警察的注意。）如此下来一年收入可达50 000美元，工作量少，风险也低——因为有很多机构涉及收取费用和收集这些费用。分摊下来，10个不同的机构有10个不同的受欢迎的徒步点，一个聪明的小偷每年只需从每个机构偷取5 000美元。

我并不是说这种犯罪行为正在发生，但它确实有可能发生。另一种类型的犯罪行为是正式员工收集这些费用并偷偷留下一小部分。

如何才能查明这种犯罪行为呢？在这些偏远地区安装摄像头可能费用太高，因为没有网络，也没有手机服务。该如何防止这种犯罪行为的发生呢？而且你如何确保每个徒步游客都支付了费用——通常情况是每天5美元？下面回答一部分问题。

我们邀请了Data Science Central的读者回答这个问题：如何发现犯罪行为，并利用数据科学解决之？这些问题的答案有真实的数据科学解决方案做支持，包括所运用的实验设计、数据创建、追踪员工接收的钞票上的编号、员工轮班的记录等。（如果你想了解更多详细答案，可以访问`http://www.analyticbridge.com/profiles/blogs/how-would-you-detect-and-stop-this-type-of-crime` 中的评论部分。）

# 数据科学的其他应用

数据科学可应用于许多其他领域和产业中，举例如下。

- 石油工业： 使用统计模型在尽可能少钻井的情况下探测油田。如果你能钻10个井，5个产油，5个不产油，并假设产油区域是一个凸

多边形，那么问题就在于根据5个产油区外的点和5个产油区内的点，估计（产油区）边界，使得钻井发现石油的概率更高。

- 信用评分：　其中一个阻碍新模型创建和投入部署的问题就是法规监管。比较大型的银行都配有研究贷款信用模型的统计学家，而这些模型必须得到监管部门的批准，如FDIC（美国联邦存款保险公司）（详情请访问`http://1.usa.gov/1aLJ9aa` ）John Elder就开创了一种基于集成方法 （*ensemble method* ）的新方法，可实现更好的评分。请注意，临床试验领域也因为对统计建模的规管，导致数据科学创新能力被扼杀。

数据科学的应用涵盖了大部分领域：传感器数据、移动数据、智能电网（优化电气网络和电力定价）、计量经济学、临床试验、运筹学（优化业务流程）、天文学、剽窃和版权侵权检测，甚至计算化学（设计对抗癌症的新分子）。

# 总结

本章讨论了与数据科学有关的案例研究、现实生活应用及一些成功例子。涵盖了各个方面的创新，从股票市场技术到广告组合优化、欺诈检测、搜索引擎优化（针对搜索公司的）、天文学（预测陨石撞击）、自动新闻反馈管理、数据加密、电子邮件营销，以及各种相关问题（在线广告）。

第7章的重点是帮助你应聘数据科学家职位。内容包括简历样本、90多个面试问题、经常雇用数据科学家的公司，以及数据科学家的典型职位名称。

---

[1] SQRT是开平方。——译者注

# 第7章
# 踏上你的数据科学职业之路

第6章讨论了各类数据科学首创的案例研究、现实生活应用和成功的例子。本章的重点在于帮助你应聘数据科学家职位。首先列出了90道面试问题（包括如何解决小而真实的数据科学问题）；然后是一些检验你视觉分析技巧的练习题，从统计学家向数据科学家转变的方法，以及知名的数据科学家和他们所拥有的技能；最后是通常使用的职位名称（实际被称为数据科学家的并不多，不过这些职位名称也在随着时代而变化）和薪酬调查。

## 面试问题

这些问题大多数是开放式的。雇主通过问这些问题来评估如总监等高级职位资深候选人技术层面知识的跨度。当然，也可以用来面试初级候选人。这些关键问题的答案可以在`http://www.datasciencecentral.com/m/group/discussion?id=6448529%3ATopic%3A133140`中找到。

## 关于工作经验的问题

- 你处理过的最大数据集是怎样的？是如何处理的？结果如何？
- 说出两个你做过的取得成功的分析项目或计算机科学项目。你是如何衡量提升效果（或成功程度）的？
- 曾参与过数据库设计和数据建模的项目吗？

- 有没有创建过仪表盘以及参与度量指标的选择？你怎样看待商业智能及报表工具（BIRT，Business Intelligence and Reporting Tools）？
- 你喜欢Teradata的哪些特点？
- 如何处理缺失数据？有哪些缺失数据填补技术可以推荐？
- 你处理过白名单吗？（在欺诈检测或垃圾邮件检测时）设置过正面规则吗？有没有优化过代码或算法，以提高SQL、Perl、C++和Python等语言的速度呢？是如何优化的？且优化了多少呢？
- 花费5天的时间得出一个90%正确率的解决方案，和花费10天的时间得出一个100%正确率的解决方案，哪一个好？这与所处的应用背景相关吗？
- 在没有做任何事情的情况下，如何证明你的算法提高了速度呢？你熟悉A/B测试吗？真正成功进行过吗？（举例说明）
- 除了PCA，你还知道或者使用过其他数据缩减技术吗？你对逐步回归有何看法？你熟悉什么类型的逐步回归技术？在什么情况下，完整的数据比部分数据或抽样数据好？
- 要做罕见事件概率估计的话，你熟悉极值理论、蒙特卡洛模拟、数理统计（或其他相关知识）吗？
- 你有使用APs、编程API、谷歌或亚马逊的API，或者AaaS（分析即服务）的经验吗？
- 在什么情况下，自己写代码要比使用数据科学软件包好？
- 你通常使用哪种工具来进行可视化？怎样看待Tableau、R和SAS？你知道如何有效地在图表或视频中展示5个维度吗？
- 你曾与哪些类型的客户一起工作过：内部客户、外部客户、销售/财务/营销或者技术人员？是否有咨询经验？和供应商打过交道吗（包括供应商选择和测试）？
- 你熟悉软件的生命周期吗？追踪过IT项目的生命周期吗（包括从收集请求到维护的整个过程）？（详情可阅读第5章“数据科学项目的生命周期”一节）
- 你是独立工作的编码人员吗？还是开发人员（开发者）？或者属于软件设计师（架构师）？
- 你熟悉定价优化、价格弹性、库存管理、竞争情报吗？请举例说明。
- 你曾经使用过时间序列模型、时间滞后的交叉相关、相关图、频谱分析、信号处理和滤波技术吗？是在哪种情况下使用的？
- 你最欣赏哪位数据科学家？哪些数据创业公司？
- 你是如何对数据科学产生兴趣的？

- 你今天都做了些什么事？或者这周或上周做了什么？
- 你读过哪些最新的数据挖掘的书或文章？什么时候读的？你参加过哪些最新的数据挖掘会议、在线研讨会、专题研讨会和培训班？什么时候参加的？你学到的最新的编程技能是什么？什么时候学到的？
- 你最喜欢哪些数据科学网站？在数据科学社群里，你最崇拜的是谁？为什么？你最欣赏哪家公司？
- 你最近一次写的数据科学博客文章是关于什么的？什么时候写的？在哪里发表的？
- 你招募过最好的员工是谁？他们现在在哪里工作？

# 技术问题

- lift、KPI、健壮性、模型拟合、实验设计和80/20规则各是指什么？
- 协同过滤、n-grams、MapReduce和余弦距离各是指什么？
- 什么是概率合并（又称模糊合并）？SQL相比其他语言，会不会更容易？你会选择哪种语言来进行半结构化的文本数据一致性处理呢？
- Toad、Brio或其他类似的客户端在查询Oracle数据库时相当低效，这是为什么呢？请说明理由。怎样才能把速度提高10倍，并能够处理更大的输出？
- 哈希表冲突是什么？如何避免？冲突多久发生一次？
- 如何确保MapReduce应用程序具有良好的负载均衡？负载均衡是什么？
- 就访问速度来看，内存中有100个小哈希表或1个大哈希表，哪一个更好（假设二者都和RAM匹配）？如何看待数据库内分析（in-database）？
- 为什么朴素贝叶斯算法问题很大？如何改善基于朴素贝叶斯算法的垃圾邮件检测算法？
- 星型架构是什么？查找表是什么？
- 你能用Excel进行逻辑回归计算吗？如果可以，结果够不够好？
- 如何定义质量保证、六西格玛以及实验设计。请举例说明好的和糟糕的实验设计。
- 一般线性模型的缺点是什么？你熟悉线性模型的替代品（如

Lasso、ridge regression和boosted trees）吗？

- 你认为50个小决策树比1个大决策树更好吗？为什么？请说明理由。
- 举一个不是高斯分布或对数正态分布的例子，以及一个混沌分布的例子。
- 为什么均方误差是很糟糕的模型性能指标？对此，你有什么建议？
- cron任务是什么？
- 效率曲线是什么？它的缺点是什么？如何克服？
- 推荐引擎是什么？它是如何运作的？
- 什么是精准测试？如果不使用精准测试，什么情况下进行模拟测试能起到辅助作用？且如何进行测试？
- 一个好的、快速的聚类算法的计算复杂度是多少？怎样才称得上是一个好的聚类算法？如何确定聚类的数量？假设有1　000万个数据点，每一个数据点都包含两个关键词，以及衡量这两个关键词相似度的指标，那么如何针对100万个独特的关键词进行聚类计算？如何创建这个含有1 000万个数据点的数据表？
- 在第2章“大数据问题是数据科学所面临挑战的缩影”一节描述的搜索引擎算法中，删除停止词是第1步还是第3步？

# 一般性问题

- 用户点击数据是否应该实时处理？为什么？请说明理由。在哪种情况下应该实时处理？
- 良好的数据和良好的模型，哪一个更好？且如何定义这个“好”？是否存在一个通用的好模型？是否存在一些必定有缺陷的模型？
- 如何处理缺失数据？请推荐一些缺失数据填补技术？
- 请比较一下SAS、R、Python和Perl。
- 大数据“诅咒”是什么？（见第2章中“大数据‘诅咒’”一节。）
- 在什么样的例子中，MapReduce不起作用？在什么样的例子中，它运作良好？云计算中涉及哪些安全问题？EMC的解决方案提供了一种混合的方法，通过私有和公有云来降低风险并提供其他优势，你怎样看待这种解决方案？（在第2章“大数据生态系统”一节，尤其是该节提供的在线参考中，你可以找到这个行业内更多参与者的信息。）

- 灵敏度分析是什么？灵敏度低（即更健壮）但预测能力弱是不是比灵敏度高但预测能力强要好？还是相反？如何进行有效的交叉验证？如何看待注入干扰信息到数据集中以测试模型灵敏度的想法？
- 请比较决策树、神经网络和逻辑回归。在过去的15年中，这些技术有多大的发展？
- 精算是统计学的一个分支吗（即生存分析）？如果不是，请说明理由。（可以在http://en.wikipedia.org/wiki/Actuarial_science和http://www.dwsimpson.com/ 上找到更多关于精算的信息。）
- 根源分析是什么？如何区别因果与相关性？请举例说明。（可以在第5章“数据科学的技术（II）”中找到这些信息。）
- 如何定义和衡量一个度量指标的预测能力？
- 误报率高好还是漏报率高好？
- 有没有想过成立创业公司？围绕哪一方面的想法或概念？
- 是否认为键入式的登录账号或密码有朝一日会不再流行？它们如何被取而代之？
- 如何才能成为一名优秀的数据科学家？
- 数据科学是艺术还是科学？
- 请举出一些数据科学最佳实践的例子。
- 什么因素会造成图表的误读、难以阅读或解读？有用的图表具有哪些特征？
- 你知道统计或计算机科学中的一些经验法则吗？或者在商业分析中有什么经验法则？
- 对未来20年，你的五大预测是什么？
- 当一篇文章（例如，在报纸上）公布一些统计数字时，如何迅速地知道它是错误的，或只是为支持作者的观点出现的，而不是具体、正确、全面的事实信息？例如，如何看待在新闻中定期讨论的官方月度失业统计数字？如何使得它们更准确？
- 根据你的观点，你认为什么是数据科学？什么是机器学习？什么是数据挖掘？

# 关于数据科学项目的问题

- 如何检测出多个用户共享一个支付账户？这对于出版商、数字化报纸出版商和提供API接入的软件开发商、（存在文件共享问题的）

音乐和电影业、收取包月费让单个用户查看或下载内容的公司来说，是个重要问题。

- 如何优化网络爬虫使系统运行得更快、提取更好的信息和更好的总结数据，以便得到更清洁的数据库？
- 有没有一个识别剽窃的好的解决方案？
- 如果要发送100万封电子邮件（进行营销活动），你将如何优化发送程序和回复程序？能否单独优化两个程序？
- 如何将非结构化数据转化为结构化数据？这是不是必需之举？是否可以将数据存储为纯文本文件，而不是使用SQL驱动的RDBMS存储数据？
- 如何创建针对打分数据的非参数置信区间？
- 如何得到欺诈检测评分技术的最好规则集？如何处理规则冗余、规则发现以及问题的组合学性质（从而寻找到最具预测能力的最佳规则集）？规则集问题可以用近似解吗？如何找到一个可行的近似解？如何判断它足够好，并停止寻找另一个更好的？
- 如何创建关键词分类？
- Betnet是什么？如何检测它？
- Zillow算法是如何估计美国任意一所房屋价值的？
- 如何检测目的不良的虚假评论或伪造的Facebook账户？
- 如何创建一种新的（重点是安全的）匿名数字货币？如何使之免受网络盗版的侵害？要怎样做才能使商店更易于接受它？（例如，每次交易都有一个独一无二的ID，每个ID只能使用一次且数天后即过期，另外独特ID的总量空间是非常大的，以防止黑客碰巧创建一个合法的ID。）
- 你设计了一个稳健的非参数统计（度量指标）取代相关性或R-Squared（拟合度），它独立于样本大小，处在−1和+1之间，并基于秩统计量，那么如何对样本大小进行正则化？请写出一个算法，计算n个元素的所有排列。如果n数值很大，如何对排列进行抽样（即产生大量的随机排列），以估计新创建度量的渐近分布呢？你可以使用这个渐近分布规范度量指标。你认为是否可能存在一个确切的理论分布，可以让你找到它并加以利用，以免在通过模拟估计该渐近分布上浪费时间？
- 接下来这个问题要比上一个问题难度更高、更具技术性。n个元素的排列，和1到n的阶乘之间，有明显的一对一的对应关系。设计这样一个算法，将小于n阶乘的整数编码成n元素的排列。怎样反过来把排列解码成一个数字？提示：　中间有一个步骤是使用阶乘数字系统表示整数。你可以在网上查参考资料来回答这个问题。更好的

办法是浏览网页找到问题的详细答案。（这将测试应聘者是否拥有快速在线搜索并找到解决方法的能力，而不是白费时间做无用功。）

- Yelp评论会收到多少“有用”的票呢？
- 基于亚马逊的公共数据，你能估计预测出任意一本书的销售情况吗？提示： 参见`http://www.fonerbooks.com/surfing.htm` 。
- 这是一个与乐高积木有关的实验设计问题（也属于计算机科学类问题）。假设你买了两套乐高积木（一组用于组装一辆车，另外一组用于组装另一辆车）。现在假设这两套积木有很多重复的相同部件。你有3种不同的方法来组装这两辆车。第一步都是通过大小及颜色来对部件进行分类。3种组装方式如下。
  1. 按顺序：一次组装一辆车，这是常规做法。
  2. 准并行系统：同时给两套积木分类，这时两套积木是混合在一起的。红色那堆部件中有的部件属于汽车A，有的部件属于汽车B。然后按照随附的说明书中的说明，依次组装这两辆车。
  3. 并行：同时给这两套积木分类，然后同时根据两组指令组装两部车。
  哪种方式最有效率？

注意　　为了检验这些假设，并帮助你熟悉分布式体系结构的概念，可以让你的孩子先同时组装两辆乐高玩具车A和B，然后再依次组装另外两辆车C和D。如果A和B之间的重复（A和B部件相同的比例）较少，则按顺序的组装方式效果最好。另一个可以引出的概念是，组装一辆有80个部件的车所需要的时间要远多于组装一辆有40个部件的车所需时间的两倍。为什么？请说明理由。（该规则同样也适用于拼图游戏。）

- 如何设计一个算法来估算你所有同事在LinkedIn上的联系人数目？如果一级联系人的数目不超过500，那LinkedIn会直接显示这个数字。但是，如果这个数字大于500，LinkedIn就只会显示“500+”。另外，你和LinkedIn联系人间共同联系人数目可以随时查看。
  对于这个问题，在此提供以下解决方案，以供学习。首先，设置如下这些代号。

- A代表你。

- B是你的一个联系人。
- y是你的联系人数量（容易获得）。
- x是B的联系人数量。（如果x大于500，就不知道确切的数量。）
- z是A和B的共同联系人数量（容易获得）。
- N是LinkedIn云的规模（会员数）或者说你所属的大网络（需要估计）。
- 基本公式：P（C是共同联系人）=P（C链接到你）× P（C链接到B）=（y/N）×（x/N）=（x×y）/（N×N）。因此x=（z×N）/y，或者N=（x×y）/z。

步骤**1**：计算**N**

建立如图7-1所示的表格，我选取了我的一部分联系人，这些人的联系人均少于500个，以确定x和z。我的联系人数量是y=9 670。

| x, for some of my connections | z, for some of my connections |
|---|---|
| 476 | 6 |
| 448 | 1 |
| 171 | 0 |
| 134 | 0 |
| 345 | 7 |
| 490 | 12 |
| 230 | 3 |
| 56 | 0 |
| 138 | 1 |
| 411 | 3 |
| 366 | 2 |
| 333 | 9 |
| 88 | 1 |
| 464 | 11 |

图7-1　样本x和z的参数表格

初步近似（只动用我的大脑而不使用其他任何工具，这就是可视化分析！）

得到N=（x×y）/z=（约）500×9670/5=（约）100万。所以我的N是100万，你的可能不同。注意，LinkedIn拥有1亿以上的用户量。

步骤**2**：为特定的联系人计算**x**

使用公式x=（z×N）/y，如果某个LinkedIn用户和我有200个共同联系人，使用y=10 000作为我联系人的近似值而不是9 670，那么他可能有大约20 000位联系人。如果他和我只有一个共同联系人，则根据预计他有100个联系人。通过看上面表中的变化，首先计算置信区间N，从而计算置信区间x。你还可以通过使用一个跟职位名称或地理位置相关的变量N提高准确性。

- 你将如何对流行的BMP图像格式进行逆向工程，使用编程语言一个字节一个字节地创建一幅图像吗？

# 测试你自己的视觉和分析思维

以下是一些可以做的练习，将有助于你在求职面试时准备得更好。这些练习旨在评估视觉和分析判断能力。

# 通过肉眼的检测模式

看看这3个图表（A、B和C），如图7-2所示。其中的两个显示出模式。哪两个？你知道这些图表被称为散点图 吗？是否有其他方式来直观地表示这种类型的数据？

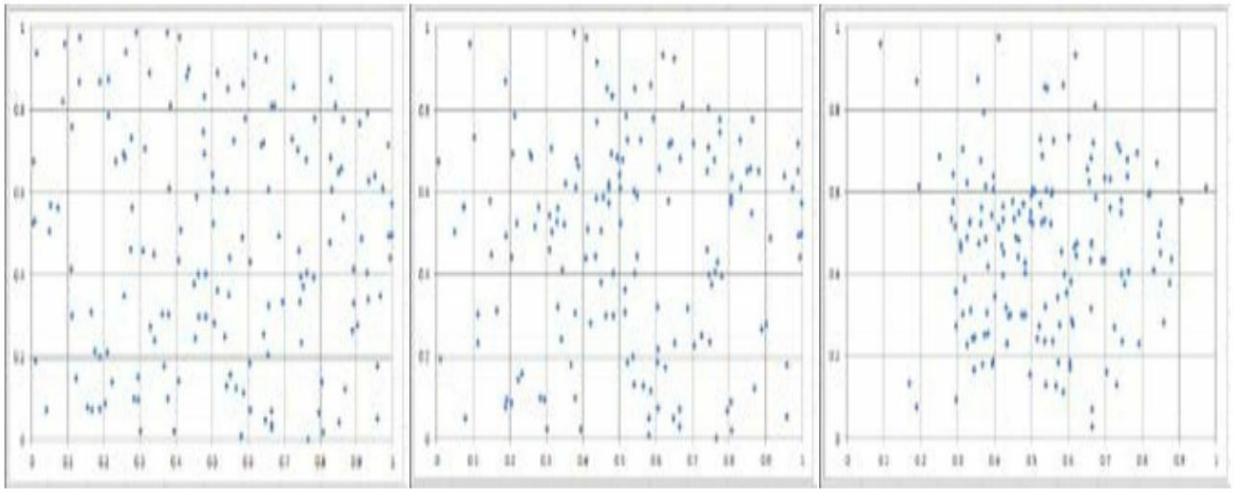


图7-2　哪个图表没有表现出模式？

很明显图表C表现出很强的聚类模式，除非要处理的数据点是未知领域里需要估计边界的随机分布。所以，最大的问题是：图表A和B之间哪一个是随机的？仔细看一下这些图表，60秒后做一次猜测，再接着往下阅读。请注意，3个图表包含相同数量的点，所以这里没有涉及缩放问题。

现在，假设你正在处理整个二维空间上数据点的空间分布，但只能通过一个小正方形窗口看到观测值。例如，数据点（观测结果）可以是从望远镜里拍摄的照片上的星星。

第1个问题是，数据被删减了。如果观察最近邻距离的分布并得出结论，那么必须考虑边界附近的点有较少的邻居，因为它们有一些邻近点在边界之外。要消除偏差，可以通过以下几种方式。

- 将该窗口复制到周围，构造数学意义上的镶嵌图。
- 将正方形观察窗映射到一个圆环面的表面。
- 应用统计偏差校正技术。
- 如果数据是真随机的话，可以用蒙特卡洛模拟来估计真正的分布（并计算置信区间）。

第2个问题是，需要使用更好的可视化工具来查看模式。使用“+”符号而不是圆点符号来表示数据点，因为有些数据点很接近时，用圆点符

号表示两个接近的数据点，肉眼看不会太明显。（在本例中，双数据点对应双星系统，即这些小尺度上点的相互作用使得上面两个图的分布非随机。）但是你可以做得更好：可以测量数量指标（平均值、标准偏差、x和y的相关性、在每一个正方形中点的数量、密度估计等），并确定其他指标，能证明所处理的不是纯粹的随机分布。

在如图7-2所示的3个图表中，如果是纯随机性的情况，x或y的标准偏差应为0.290±0.005。若用这种随机性测试，3个图表中只有一个能通过。

第3个问题是，即使多个统计测试表明数据为真正随机，也并不意味着实际上就是如此。例如，3个图表都表明x和y之间的相关性为零，并且x和y的均值接近0.50（0.50的均值是这种情况下随机分布的必要条件）。然而，只有一个图表显示随机性。

第4个问题是，需要一个数学框架来定义和检查随机性。真正的随机性是通过Poisson随机过程实现的，需要使用能唯一描述Poisson过程的指标，来检查数据点的分布是不是真正随机。这些衡量指标可以是：

- 点与点之间的距离分布。
- 在小格子里观察到的数据点数。（每个小格子里数据点的数量应该是均匀分布的，可以用chi-square检验，但是，在这些图表中，每个小格子里没有足够多的点，因此没法得到有效的检测结果。）

第5个问题是，一些好用的指标（如点k它和最近的邻近点之间的距离）可能不存在一个简单的数学公式。但是，你可以使用蒙特卡洛模拟来解决这个问题：模拟一个随机过程，根据成千上万次的模拟，（用置信区间）计算距离的分布，并且将其与你基于数据集计算的距离结果做比较。如果在数据集上计算的距离分布与模拟结果可以相匹配，那么你是正确的，它意味着数据可能是随机的。然而，你必须确保距离分布就是Poisson过程，没有任何非随机过程可以产生跟它相同的距离分布。这项练习被称为“拟合度测试”：可以尝试看看你的数据是否支持特定的随机性假设。

第6个问题是，如果你有100万个点（在高维空间中，由于维度诅咒，需要的点的数量还要超过100万），那么你将有10　000亿个距离需要计算。没有电脑能在1　000年内完成这些计算，即使在云计算中也不

能，所以你需要随机选择10 000个点计算距离，并与基于模拟数据的等效计算进行比较。还需要重复1 000次模拟来得到置信区间，不过这个计算量是可行的。

以下展示了如何创建图表A、B、C中的数据。

- 生成158个随机点[a（n）, b（n）]，n=1,...,158。
- 生成158个随机的偏差u（n），v（n），n=1,...,158。
- 对于n>1，定义x（n）如下：如果u（n）<r，那么x（n）=a（n），否则x（n）=s×v（n）× a（n）+[1–s×v（n）]×x（n–1），其中x（1）=a（1）。
- 对于n>1，定义y（n）如下：如果u（n）<r，那么y（n）=b（n），否则y（n）=s×v（n）× b（n）+[1–s×v（n）]×y（n–1），其中y（1）=b（1）。
- 图表A：x（n）=a（n），y（n）=b（n）。
- 图表B：r=0.5，s=0.3。
- 图表C：r=0.2，s=0.8。

唯一呈现随机性的图表是图表A，图表B的x和y有明显太低的标准差，且有很少的点接近边界，很多点的距离太近。此外，图表A（随机分布）表现出一些聚类效果，以及一些点的对齐效果。然而，这符合随机分布的完美预期。如果在每个小格子里的点数目相同，分布就不是随机的，反倒符合有外力使得数据点尽可能远离彼此的情况。如果有2个点（因此没法做检测），你会如何测试随机性？3个点呢？10个点又如何测试？

最后，一旦检测到一个模式（例如，邻近点之间的距离异常小），则应该解释或尊重这种模式。也就是说，如果框架是股票交易，那么会推导出正投资回报率的交易规则；如果框架是天文学，那根据图表B可以得出双星确实存在的结论。

# 识别偏差

假设图7-3是我画的，然后我告诉你，这就是生活在距地球10亿英

里远的土星上的人看到的地球和月球。地球的半径约为4　000英里，地球与月球之间的距离约为20万英里。你的反应是什么？答案无所谓正确还是错误，只是用来简单地检查你的思维过程。

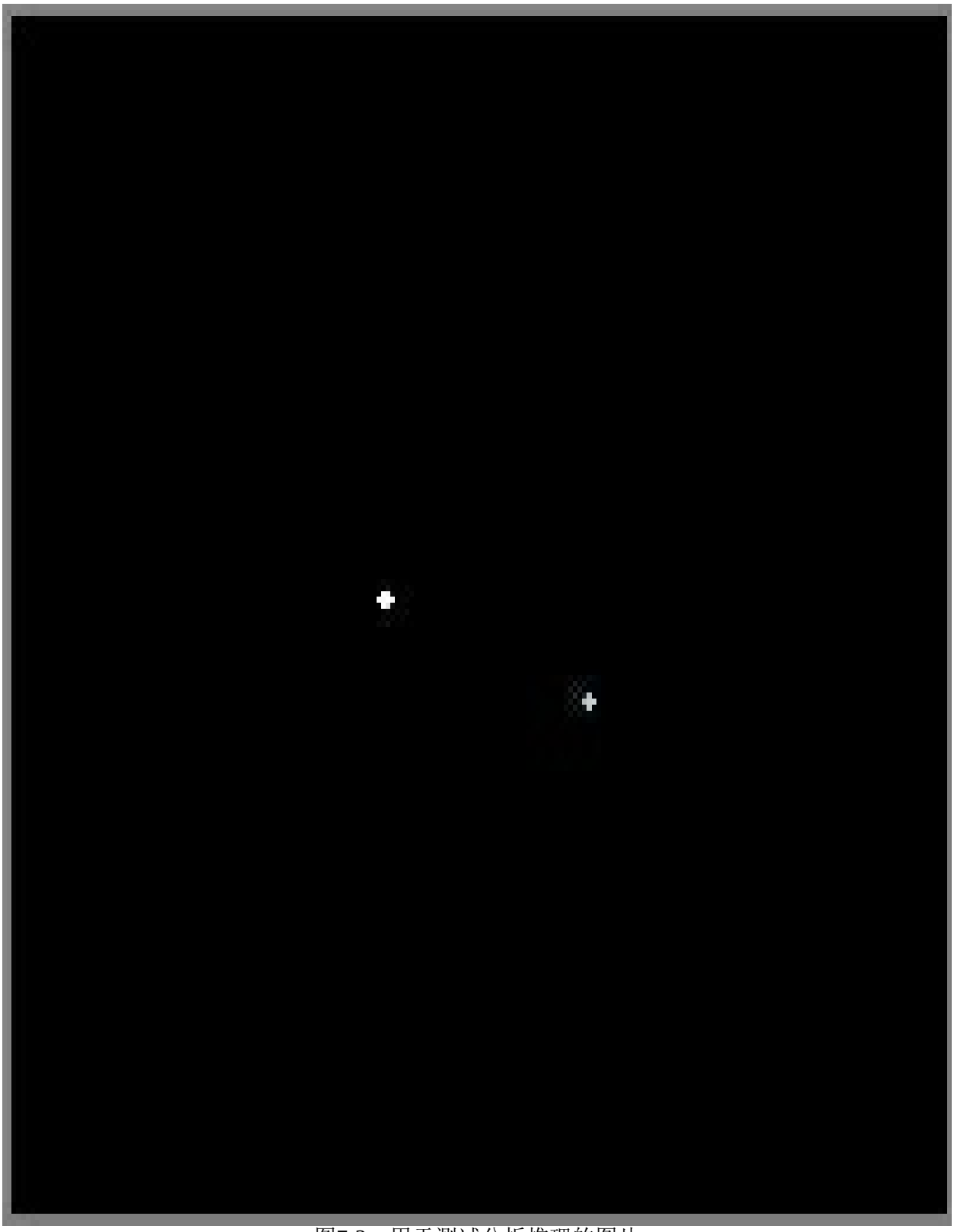

图7-3　用于测试分析推理的图片

可能的答案，包括：

- 图像被放大了。地球太小，在火星上用肉眼是看不见的，除非被太阳照得很明亮；这是一个虚假的彩色图像。
- 在土星上观看，地球和月亮之间的距离似乎太大；在土星上，肉眼甚至是看不见地球的；从土星上看到的太阳，才和照片上的地球一样大一样亮。
- 与地球相比，月球的体积太大了。
- 为什么月球和地球一样明亮？为什么地球比其他的恒星亮？
- 其他恒星都在哪里？

# 误导性的时间序列和随机游走

许多时间序列图表似乎表现出一种模式：上升趋势明显的周期，或者是无记忆的随机过程等。如图7-4所示的图片代表股票价格的模拟，X轴代表时间（天），Y轴代表模拟的股价。

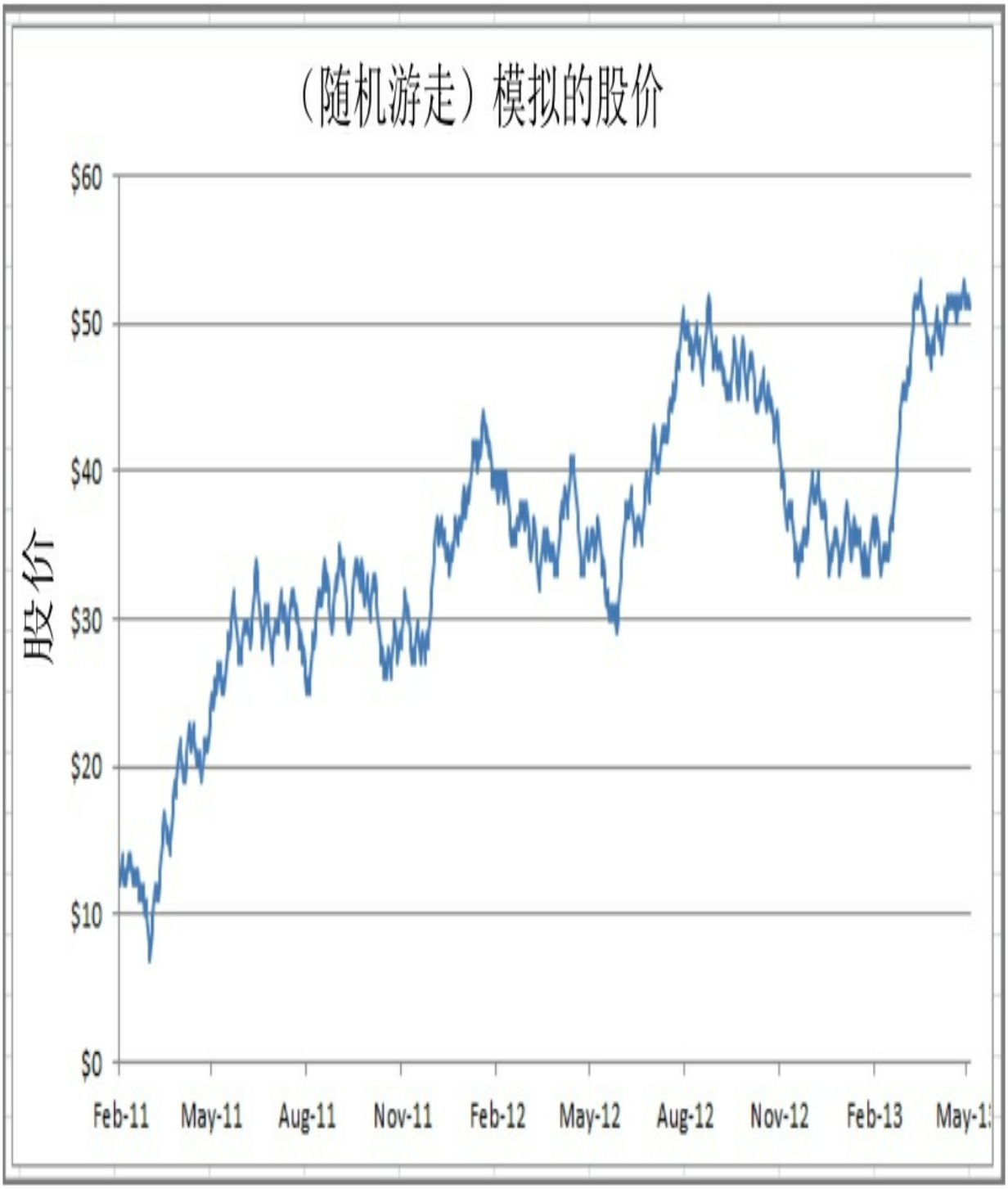


图7-4　股价模拟

你认为数据有上升趋势吗？其实从长远来看，并不存在上升趋势，它是纯粹随机游走的体现。在任何时候，股价上升或下降一定数值的概率稳定在50%。然而，对于短周期，观察到起伏是完全正常的。这并不

会使得这个时间序列可预测：可以尝试设计基于前1　000个观测值训练出的预测模型，然后测试剩下的1　000个观测值。你会发现，这个（用于预测上下走势的）模型还比不上随机掷骰子。

要了解随机性和缺乏随机性的不同表现，还可以做另一个测试：模拟自回归时间序列。其中最基本的过程之一就是X（t）=a×X（t–1）+b×X（t–2）+E+c，t是时间，c是常数（–0.25<c<+0.25），E是[–1,+1]之间均匀分布的偏离误差项（白噪声，许多频率不同但强度相等的混合杂音）。例如，X（t）表示时间处于t时的股价。你可以从X（0）=X（1）=0开始。如果a+b=1，则该过程是稳定平衡的。为什么？请说明理由。如果c<0，则存在下降的趋势。如果c>0，则存在上升的趋势。如果c=0，则既不上升也不下降（平衡过程）。如果a=b=0，那这个过程是无记忆的，里面没有随机模式，它只是纯粹的噪声。试着用不同的a、b、c值来制作一些图表，然后看你是否在视觉上能注意到这种模式，并能正确地解释它。在图7-4中，c=0，b=0，a=1。当然，a、b、c取某些值时，视觉上看有明显的模式。但如果使c和a+b都接近零，那么这是视觉上更困难的练习，你可能要看很长的时间，大脑才能识别正确的模式。试着用一个预测算法来检测这个模式，看看大脑能否击败你的预测算法模型？如果能，什么时候能击败？

# 从统计学家到数据科学家

最近LinkedIn上有一篇长文章，其中谈到为何统计学家不属于大数据范畴，以及如何改善这种情况。你可以在http://www.datasciencecentral.com/m/blogpost?id=6448529%3ABlogPost%3A123227 上找到全部的讨论。在撰写本书时，我已经从业界领头的数据科学家和统计学家那儿收集到160条评论。我认为这些评论最能帮助你（尤其当你是统计学家时）了解能做些什么来让自己更受市场欢迎。

# 数据科学家也是统计从业人员

统计学科对跟大数据相关的应用计算层面处理得不够多，或忽视了从数据中获取商业价值的层面。统计人员应具备敏锐的商业触觉，尤其是在小公司或科技初创公司工作时，能够执行原本由业务分析师执行的任务。（在第6章“数字分析”一节中有这方面的例子）。这也将有助于统计学家获得更多工程和计算机科学的知识，包括APIs、Python、MapReduce、NoSQL，甚至Hadoop架构，至少也能了解这些技术和语言，以及它们是如何影响数据访问、数据存储、算法效率和数据处理的。

如果要转变为数据科学家，统计人员往往具有很大的优势，因为他们通常拥有许多数据科学家所缺乏的关键技能，比如实验设计、抽样和R编程。

如果认为大数据和数据科学仅仅属于纯粹的工程或计算机科学领域，跟统计学无关，或者统计的用途不大，那你现在读的这本书，将终结这些谬论。在本书中，你会发现数据科学包括统计和统计研究这部分核心内容。例如，在第2章“大数据的独特性”中，计算数十亿或数万亿的大数据集的相关性时，必然会找到虚假的相关性。这种虚假相关性掩盖了难以发现的真实相关性，所以此时应该比较相关图，而不是看相关性。如果两个时间序列相似，那么相关性没有作用，相关图却有用。另外，还需要学会对尺寸进行正则化。你不必成为一名统计专家，才能确定这些问题和偏差并纠正它们。一名数据科学家也应该知道这些事情，以及其他的事情，比如实验设计、极值理论和蒙特卡洛模拟的用法、无须统计模型创建置信区间（如分析桥第一定理）、识别非随机性等。

你可以同时是一名数据科学家和一名统计学家，就像你可以同时是一名数据科学家和一名企业家一样——实际上，这也是必然要求。这样的混合角色完全可以存在。你必须意识到，在ASA出版物图片或招聘广告上的统计学家的官方形象，并不代表许多统计学家的真实情况（至少目前是这样的）。

数据科学跟统计仍然存在不同的地方。许多数据科学书籍给你的印象是，它们是千篇一律的，但那是因为作者只是重复使用了旧的信息（甚至不属于数据科学范畴），加了一点R和Python的东西，并提出一个新的名字。我称这个为假数据科学　。同样，没有统计基础的数据科学（或对统计原理的轻率应用）也不属于真正的数据科学。

最终来说，关于数据科学、统计学、计算机科学、商业智能和创业

是什么，每个人都有自己的想法。你决定想要称自己是什么角色。就我而言，我很清楚自己不再是一名统计学家，而是一名数据科学家。（虽然我个人以计算机统计学家的身份出道。）我的知识和专业有别于那些统计学家。（或许跟计算机科学更接近。）虽然我具备良好的实验设计、蒙特卡洛模拟、抽样等知识，但大部分不是我在学校或训练项目中学到的。这些知识在互联网上是免费提供的，无须参加统计课程，任何人，无论是律师、政治家还是地理学家都可以获取这些知识。

注意　Data Science Central的数据科学学徒计划，就是要使更广泛的群体能获得这一知识。我的直觉是，无模型置信区间的分析桥定理是一个例子，就算没有数学前提和适用于大数据环境下的大量数据，也可被一个12岁的孩子理解的“统计”工具。

# 谁应该给数据科学家教统计学

数据科学方面的统计文化，可以来源于数据科学家，也可以来自外部（那些不想被称为数据科学家的人，比如说宁愿称自己为统计学家的人）。我就是一个作为数据科学领域内部人员创建数据科学相关统计文化的例子。有些人喜欢外部统计学家的贡献，而有些人喜欢内部人员的贡献。因为内部人员熟悉数据科学，了解统计和数据科学之间的差异，与商界人士也更接近。

# 雇佣问题

许多政府机构和大型非政府组织提供了各种有趣的职位，但富有创意的数据科学家在创业公司中可能会发展得更好，尽管可能面临一些挑战。其中一些挑战（同时也是阻碍雇主吸引应聘者的因素）包括：

- 官僚政策往往阻碍了人们申请职位。例如，美国的安全许可要求（不接受绿卡持有者和来自国外的申请者，但如果你是美国公民，将会给你提供更多的职位选择）。
- 工作地点不吸引人或不灵活。如果你申请的职位是其他申请人不怎

么愿意去的地方，那么你得到该工作的可能性会增加。如今，一些雇主提供远程办公职位，以增加吸引优秀人才的机会。（雇主需要注意，当雇用一个数据科学家时，你可能需要考虑远程办公这个选择。）

# 数据科学家与数据架构师密切合作

统计学家应该多少了解如何设计数据库：如指标的使用（加以精确界定）、怎样采集数据，以及当保留旧数据的时候，需要多大的颗粒度。这些都是数据科学的一部分。这涉及需要与数据库工程师和数据架构师密切合作。数据科学家必须有一些数据架构师和商业人士的知识，以确切了解他们在做什么，他们将要获得什么。数据科学家也应该参与仪表板设计。通常情况下，这些东西都是许多统计学家不想做的，他们可能觉得这不是分内职责。

# 谁应该参与战略思考

无论任何项目，讨论问题是第一个关键而基础的步骤。如果没有讨论，就像是在建没有地基的摩天大厦。但应允许灵活性思考。如果效果不如预期，那么你将如何重建或改变该项目？要如何适应变化？你的设计是否足够灵活，容易适应变化？谁负责确定项目及其主要功能？

当业务分析师（在较大的公司会有业务分析师）和高管们最终要定义KPI和仪表板的设计时，数据科学家必须参与进去而且要和他们沟通（特别是雇员少于500人的公司），而不是孤立地工作。数据科学家还应该与数据架构师、产品经理、工程师或软件工程师进行沟通。

这种沟通之所以至关重要，有几个原因：帮助数据科学家理解商业问题，并获得商业敏锐度和必要领域的知识，使技术实现过程更平滑，通过建议影响公司领导，帮助他们做出可靠的决策，使得分析敏锐度更高。这种双向沟通对于优化业务流程十分有用。

# 两种类型的统计学家

在我看来，统计学家有两种类型：一类属于ASA（美国统计协会），另一类则不属于。同样，有两种类型的大数据从业者：

- 一类对统计学知之甚少，但对大数据其他基本领域知识的了解远远超过统计学家。
- 另一类则像很多世界上最优秀的统计学家一样资深，拥有面向数据科学的统计知识。

我建议统计人员申请去不具备较强统计知识的团队工作。（可以通过在面试过程中向对方提问来评估团队的能力，比如他们做的是什么样的统计分析。）如果你能够通过展示抽样、数学建模、实验设计、数据补齐技术、识别和处理不良数据、生存分析和置信区间的价值来说服他们，那将更能展示自己的实力（展示真实案例是加分项）。

同时，当面试一个数据科学家职位时（或当面试官不是统计学家时），面试中不要使用过多的统计学术语，可以只是讨论简单模型的好处。例如，p值 的统计概念很少用于数据科学，并不是因为数据科学家知识匮乏，而是因为数据科学会使用不同的术语和不同的度量标准，能起到类似的作用。另外，在数据科学背景下，很多时候找不到基础模型。所以主要做无模型的推理，如预测、置信区间等，这些没有统计模型，而基于数据驱动。因此，数据科学家面试官会更容易接受简单的模型讨论。

注意　建议使用其他的表示方法来代替p值，如在第5章谈及的无模型置信区间。非统计学家很容易理解这种等效的方法。在另外的情形下，我使用预测能力（在第6章讨论了这个容易被非统计学家理解的指标）来给p值做不同的、更简单的、无模型的解释。此外，和客户交谈时，需要说他们能够明白的词，而不是p值这类行业术语。

数据科学家也做EDA（探索性数据分析），这不完全是由统计学家做的事。（见第5章“数据字典”一节。）作为一名统计学家，应该很容易理解数据字典，在处理大数据集时，你可能已经在使用它了。事实上，EDA过程很大程度上可以自动化，创建数据字典是这个自动化过程

的第一步。

# 大数据与取样

有时候需要整个数据集来做分析。当创建一个估计美国每所房屋价值的系统时，你可能希望将整个数据集作为分析的一部分。每天有数以百万计的人在了解数百万的房屋和邻近的房屋的信息，进行比较和购买。在这种情况下，使用整个数据集是有意义的。如果数据库中只有10%的历史价格数据，那么，虽然会错过很多的局部模式，但依然可以做一些推理。不过还是有90%的机会，当用户输入地址以获得价格估计时，还必须补充房屋面积、历史销售记录、浴室的数目、学校排名等信息。这种情况会导致（Zillow.com）这个应用程序无用。

不允许采样的观测数据的例子如下。

- 信用卡处理：每一笔交易都必须进行审批。
- 在亚马逊网站上推荐的书：每一本书和每个用户都必须是数据集的一部分。
- 每所房子的估价。
- 高频交易（上万亿的微小数据块，每块的数据越多越好）。
- 社交网络上的朋友和其他推荐结果。
- 邮件分类：垃圾邮件、非垃圾邮件。每一封邮件都必须经过处理。
- 传感器数据：较高的分辨率和频率会提供更高的预测能力。
- 可以按每个客户网上预订房间的请求来自定义房价。
- 关键词竞价：需要对10亿关键词全部进行实时定价。
- 关键词相关性：找到与特定关键词相关的所有关键词。搜索引擎技术或关键词分类中会用到，因为每天要进行数十亿次搜索。
- 广告相关性：针对每个页面请求，将用户与网页和广告进行匹配，每天次数达到数十亿。
- 新闻提要聚合器：检测、分类、管理数以百万计的微博帖子，向新闻联盟的合作伙伴提供高质量的新闻。每个帖子都要计算在内。

# 数据科学家的分类

了解数据科学家的分类，可以帮助确定你对数据科学职业的哪些领域存在潜在兴趣。这个讨论是我最近做的一个数据科学项目的结果，它比上一节略有更多的技术性。你可以跳过技术性问题，重点关注主要的结论。但因为这是一本数据科学的书，所以我觉得最好也将我过去常用来得出结论的方法细节在此介绍一下。

这个讨论的关键在于，基于你的背景帮助你确定想要专门从事的技术领域。基于LinkedIn上的公开数据，通过附加权重给不同的领域，你能（有统计显著性地）识别数据科学相关的主要技术子领域（机器学习、数据挖掘、大数据和分析等）。

另外，提供的名单中，列出了一些知名的、成功的数据科学家。你可以进一步查看他们的个人资料，以更好地理解成功的数据科学家实际上做了什么。最后，对各领域之间的相互关系进行了解释，展示了它们实际上是有相当多重叠的。

# 数据科学最流行的技能集合

这是我对LinkedIn的一项研究结果，根据自己LinkedIn账户里8 000多位数据科学家联系人，来确定使用最频繁的数据科学技能是哪一项，并对杰出的数据科学家进行定位。通过搜索LinkedIn获得10个以上数据科学技能背书支持的顶级数据科学家，并分析他们个人资料中的五大技能，得出如图7-5所示的列表。

对数据科学相关技能的统计有效性强，然而对顶级数据科学家统计的效力往往会比较弱。原因是，你至少需要10个LinkedIn数据科学技能背书，才能成为列表中的顶级数据科学家。这个列表未列出一些开创性的数据科学家，因为他们没在LinkedIn的个人资料上添加数据科学技能。这一点容易理解，如果你不在内容中使用Twitter的#标签，或根本不发推文，是完全不可能跻身于Twitter#标签统计出的顶尖大数据人列表中的。

图7-5展示了数据科学相关的技术（DS代表数据科学）。简而言之，你可以把数据科学方程式表示为：

数据科学=0.24*数据挖掘+0.15*机器学习+0.14*分析+0.11*大数据+…

| 技术 | 人数 | 与数据科学的关联度 | 百分比 |
|---|---|---|---|
| 数据挖掘 | 10 | 139.54 | 27.3% |
| 机器学习 | 4 | 58.79 | 11.5% |
| 分析 | 6 | 69.69 | 13.6% |
| 大数据 | 7 | 53.82 | 10.5% |
| 预测分析 | 3 | 33.27 | 6.5% |
| 数据分析 | 4 | 29.86 | 5.8% |
| 预测建模 | 2 | 24.13 | 4.7% |
| Hadoop | 2 | 14.56 | 2.8% |
| 文本挖掘 | 1 | 14.33 | 2.8% |
| 统计 | 1 | 13.09 | 2.6% |
| 自然语言处理 | 1 | 11.67 | 2.3% |
| 创业公司 | 1 | 10.43 | 2.0% |
| 算法 | 1 | 8.18 | 1.6% |
| 分布式系统 | 1 | 6.50 | 1.3% |
| Map Reduce | 1 | 5.56 | 1.1% |
| 数据仓库 | 1 | 4.89 | 1.0% |
| 商业智能 | 1 | 4.29 | 0.8% |
| SQL | 1 | 3.18 | 0.6% |
| R | 1 | 2.83 | 0.6% |
| 可扩展性 | 1 | 2.72 | 0.5% |

图7-5　顶尖数据科学技术及相关领域

请注意，出人意料的是可视化这个技能并没有显现在顶部。这是因为可视化，就像UNIX，被看作一种工具（例如Tableau），或有时被看作一项软技能，而不是一项硬技能、一门技术或一个领域，因此虽然它经常在LinkedIn资料中被提到，但仍然不存在于大数据科学技能的前5名里。计算机科学也没有被提及，或许是因为它是一个过于宽泛的领域，人们反而会（在个人资料中）列出较窄的领域，如数据挖掘和机器学习。

图7-6（从定量的角度）显示了按技能和人员分类的数据科学相关技能。（用于计算如图7-5所示的汇总表。）例如，第1行中的信息是这样的：Monica Rogati在她的LinkedIn资料里把数据科学列为第3项技能；有61位数据科学相关人员和106位机器学习相关人员为她背书。机器学习对数据科学的贡献，可以通过计算106/3得出，她的贡献值是35.33。根据LinkedIn的数据，图7-6列出了顶级数据科学家中的先锋人物。

| 技术 | Name | 认可度 | 数据科学技术排名 | 数据科学认可度 | 数据科学技术关联度 |
|---|---|---|---|---|---|
| 机器学习 | Monica Rogati | 106 | 3 | 61 | 35.33 |
| 分析 | Vincent Granville | 474 | 16 | 44 | 29.63 |
| 数据挖掘 | Vincent Granville | 421 | 16 | 44 | 26.31 |
| 数据挖掘 | Monica Rogati | 77 | 3 | 61 | 25.67 |
| 分析 | DJ Patil | 141 | 7 | 56 | 20.14 |
| 预测建模 | Vincent Granville | 292 | 16 | 44 | 18.25 |
| 数据挖掘 | DJ Patil | 125 | 7 | 56 | 17.86 |
| 数据挖掘 | Kirk Borne | 176 | 11 | 46 | 16.00 |
| 预测分析 | Vincent Granville | 252 | 16 | 44 | 15.75 |
| 数据挖掘 | DJ Patil | 108 | 7 | 56 | 15.43 |
| 数据挖掘 | Gregory Piatetsky-Shapiro | 137 | 9 | 24 | 15.22 |
| 预测分析 | DJ Patil | 102 | 7 | 56 | 14.57 |
| 文本挖掘 | Monica Rogati | 43 | 3 | 61 | 14.33 |
| 大数据 | Monica Rogati | 42 | 3 | 61 | 14.00 |
| 大数据 | Vincent Granville | 223 | 16 | 44 | 13.94 |
| 机器学习 | Kirk Borne | 152 | 11 | 46 | 13.82 |
| 统计 | Kirk Borne | 144 | 11 | 46 | 13.09 |
| 自然语言处理 | Monica Rogati | 35 | 3 | 61 | 11.67 |
| Hadoop | Milind Bhandarkar | 194 | 18 | 10 | 10.78 |
| 创业公司 | DJ Patil | 73 | 7 | 56 | 10.43 |

图7-6　数据科学技术领域的贡献者

在LinkedIn上被认可的前10位数据科学家的完整列表，可以在`http://bit.ly/LiRJXQC` 上下载。

## 替代公式

对于每个技能，如果不是针对图7-6中的10人按Endorsements（person,　skill）/DS_Skill_Rank（person）求和，而是按SQRT{ Endorsements（person, skill） * DS_ Endorsements （person） }求和，将得到下面这个不同的结果：

数据科学=0.22*数据挖掘+0.12*机器学习+0.13*分析+0.11*大数据+…

无论是用第一个还是第二个公式，你都要处理3个参数：n、m和k。

- 只看到每个人的前n项技能（n=5，并且“数据科学”这个技能排除在外）。
- 忽略那些得到的数据科学背书数目m小于10的人。
- 只能看到k=10位顶尖人士。

这个做法最复杂的问题是至少要用10个背书，来确定在LinkedIn上的所有数据科学专业人士。其实可以把100位知名数据科学家作为种子用户，然后查找这些种子用户下的第一级、第二级和第三级相关人员来解决这个问题。“查找相关人员”是指访问LinkedIn的功能，该功能会告诉你“同时访问了X和Y资料的人”，并通过网络爬虫，为每个人提取背书的数目和技能。

## 哪一个最好

从方法论角度来看，你觉得哪一个公式是最好的？第一个，还是第二个？毫不奇怪，它们都得出相似的结果。我比较喜欢第二个。

如何找到数据挖掘、大数据、机器学习等学科类似的公式，这可以当作很好的练习题。重要的是，要记住一个人可以不仅仅是数据科学家——比如可以同时是数据科学家和音乐家。这解释了为什么对于任何人的技能等级来说，数据科学排第一的情况很少：我获得更多的是跟数据挖掘和数据分析有关的背书，这是因为数据科学是相对较新的领域。

### 技能的互相影响

机器学习是数据挖掘的一部分（至少对某些人来说）。数据挖掘和机器学习都涉及分析、大数据和数据科学。大数据包括分析、数据挖掘、机器学习、数据科学等。那么如何处理技能之间的相互影响呢？应该用多个方程吗？一个数据科学方程、一个数据挖掘方程、一个大数据方程，或尝试解决线性方程系统？每个方程可以使用跟数据科学方程相同的方法。这个问题留给你作为练习。

Hilary Mason（在本书写作的时候）把数据科学技能排第四，获得了49个背书，此外，她还有以下技能。

- 机器学习（104个背书）。
- 数据挖掘（64个背书）。
- Python（54个背书）。
- 大数据（48个背书）。
- 自然语言处理（28个背书）。

虽然她是第二级联系人，使用我的LinkedIn账户搜索数据科学时不会出现，但她绝对应该进入前10名。如果将她添加到列表中，那么Python会以低权值上升成数据科学相关技能，排名高于R或SQL。

## **LinkedIn**上的顶级数据科学家

本节将讨论一些世界顶级数据科学家和他们感兴趣的领域。在图7-7中对此进行了总结。

| | 分析 | 大数据 | 数据挖掘 | 数据挖掘 |
|---|---|---|---|---|
| DJ Patil | 0.34 | 0.09 | 0.30 | 0.26 |
| Dean Abbott | 0.24 | 0.02 | 0.46 | 0.27 |
| Eric Colson | 0.28 | 0.38 | 0.27 | 0.08 |
| Gregory Piatetsky-Shapiro | 0.21 | 0.24 | 0.40 | 0.16 |
| Kirk Borne (1) | 0.00 | 0.15 | 0.45 | 0.39 |
| Marck Vaisman | 0.28 | 0.17 | 0.42 | 0.13 |
| Milind Bhandarkar | 0.09 | 0.54 | 0.12 | 0.25 |
| Monica Rogati | 0.09 | 0.17 | 0.31 | 0.43 |
| Simon (Ximeng) Zhang (2) | 0.53 | 0.14 | 0.32 | 0.00 |
| Vincent Granville | 0.38 | 0.18 | 0.34 | 0.10 |

图7-7　LinkedIn上的顶级数据科学家及技能

根据LinkedIn的个人资料，此处列出的10位开拓性数据科学家被评为顶级数据科学家。我计算了LinkedIn上他们在四大数据科学相关技能——分析、大数据、数据挖掘和机器学习上——获得的背书数量（如在上一节中讨论过的，这些技能被认定为跟数据科学联系最为密切）。然后对这些数量进行归一化，每个数量都变成0和1之间的比例，每个人的这四项之和是百分之百。这种方法使得分类更加容易。

请注意，机器学习和分析的相关性结果是负数（–0.82）。同样，大数据与数据挖掘之间的相关性结果也是负数（−0.80）。所有其他交叉技能的相关性可以忽略不计。其他值得注意的项目包括以下内容。

- Kirk Borne的技能是高度分散的。没有列出“分析”技能，但他有许多分析相关的技能。
- 完成这一分析的第二部分时，Simon Zhang的分析背书量出现峰值。峰值可能代表数据故障。

在6个散点图中，大数据和机器学习表现出最强的聚类结构。Milind Bhandarkar（Pivotal的首席科学家）是离群的，Eric Colson（Netflix的前数据科学与工程部门副总裁）离群程度更小点，但两人的大数据知识都很丰富。

- Kirk Borne（乔治梅森大学计算机科学教授）和Monica Rogati（Jawbome的数据科学副总裁）因强大的机器学习识别形成一个聚类。
- DJ Patil（前LinkedIn的首席科学家）和Dean Abbott（Abbott Analytics的总裁）属于中间集群，具有良好的机器学习能力，但不知道大数据能力如何。
- Simon Zhang（LinkedIn总监，此前在eBay，不算纯机器学习背景），以及至少和图表密切相关的Gregory Piatetski-Shapiro（KDnuggets的创始人）、Vincent Granville（Data Science Central的联合创始人，本书作者）和Marck Vaisman（EMC），构成了剩余的聚类。

这是大数据分析吗

可以说是也可以说不是。之所以说是，因为从LinkedIn TB级的数据中抽取需要的东西，利用专业知识尽量减少工作量和要处理的数据。之所以说不是，因为它不涉及大量的数据传输——信息组织得很好，易于高效访问。可以说这是微小的数据，毕竟只有10个观察值和4个变量。但是，这个10×4的表是一个汇总表。计算LinkedIn上获得最多背书的数据科学家不是易事，除非你有这个领域的专家技能。

我用的是“手动分群”。可以把我的分析看成是轻分析。使用重分析能否做得更好（或更糟）？比如从LinkedIn中提取更多的数据

（用10个度量从5　　000人中选出200人），并应用真正的（不是手动）聚类算法会怎么样？用什么度量来评估重分析和轻分析谁的提升效果大？

## 谁是“最典型的”数据科学家

针对数据科学技能组合，我用一个公式：数据科学=0.24*数据挖掘+0.15*机器学习+0.14*分析+　0.11*大数据，比较了10个数据科学家的技能。简单说，我计算了10个相关性（每个数据科学家一个），以确定谁最能代表数据科学。结果如图7-8所示。

| | 数据科学典型程度 |
|---|---|
| DJ Patil | 0.55 |
| Dean Abbott | 0.93 |
| Eric Colson | -0.18 |
| Gregory Piatetsky-Shapiro | 0.83 |
| Kirk Borne | 0.70 |
| Marck Vaisman | 0.84 |
| Milind Bhandarkar | -0.63 |
| Monica Rogati | 0.41 |
| Simon (Ximeng) Zhang | 0.20 |
| Vincent Granville | 0.43 |

图7-8　LinkedIn上“最典型的”数据科学家

Dean Abbott是最接近“平均”的（我定义为“最典型的”），而（擅长大数据、分布式计算的）Milind Bhandarkar则离中心最远。尽管（我自己和其他人）一再声称，我是一个纯粹的数据科学家，但我只得了0.43

分。（当然，我也算是产品经理/营销人员/财务人员/企业家，而不仅仅是一个数据科学家，但这些额外的技能从这个计算实验中分离出去了。）令人惊讶的是，Kirk Borne是一个天体物理学家，但却在数据科学典型指数中得了高分。Gregory Piatetsky-Shapiro是一位数据挖掘工作者，但也得到了高分。

# 400个数据科学家职位头衔

我查看了自己LinkedIn联系人网络里10 000多个数据科学家的资料，发现大多数数据科学家（包括我）不使用“数据科学家”这个头衔来描述自己的工作。那么对于数据科学家，最受欢迎的职位头衔是什么？下面是我从中找到的“总监”和“首席”级别的职位名称的列表。在一个团队中，每一个总监下面有多个初级和中级的数据科学家，这些人有类似的职位头衔（只是没有“总监”或“首席”二字）。还有许多数据科学家的身份是顾问、负责人、创始人、首席执行官、教授、学生、软件工程师等。

首席：

- 地理空间分析与建模精算师。
- 分析员。
- 分析和算法人员。
- 信贷和分析人员。
- 数据和分析人员。
- 研究和分析人员。
- 科学家、全球分析主管。
- 分析副总裁。
- 技术官、企业信息管理者和分析师。
- 客户总监、业务分析员。

总监：

- 分析。

- 分析科学家。
- 分析与交付。
- BI和分析。
- 虚假分析和研发人员。
- 预测分析。
- 分析师与创新策略。
- 市场营销分析。
- 数字分析。
- 先进分析。
- 分析咨询、产品或数据忠诚度分析。
- 分析解决方案。
- 数据分析和广告平台。
- 数字分析和客户洞察。
- 健康分析。
- 创新、大数据分析。
- 产品分析。
- 风险分析与政策。
- 企业营销管理（EMM）科学和分析。
- 网络分析与优化。
- 先进战略分析。
- 分析战略。
- 分析服务。
- 大数据分析和细分。
- 商业分析。
- 商业分析和决策管理战略。
- 商业智能和分析。
- 商业规划和分析。
- 临床分析。
- 客户分析。
- 客户分析和定价。
- 客户洞察和商业分析。
- 数据分析。
- 数据科学与分析实践。
- 数据仓库和分析。
- 数据库营销和分析（市场营销）。
- DVD BI和分析。
- 游戏化分析平台、信息分析与创新。

- 全球数字营销分析。
- 群组分析。
- 法医数据分析。
- 市场营销分析。
- 预测分析应用。
- 报告/分析。
- 风险与分析。
- 风险与商业分析。
- 统计建模与分析。
- 统计及项目分析/高级分析顾问。
- 战略分析。
- 网页分析。
- 完整的列表可以在http://www.datasciencecentral.com/m/blogpost?id=6448529%3ABlogPost%3A74980 上找到。

# 薪酬调查

本节提供了全球几个主要城市中大数据职位的数量及薪酬数据。

交叉参考　第3章“成为一名数据科学家”简要介绍了薪酬调查并给出了综合信息链接（http://www.analyticbridge.com/m/group?id=2004291%3AGroup%3A140438 ）。在第8章“数据科学资源”中，提供了岗位描述和简历范例，以及招聘数据科学家的公司信息。

下面的数据是2013年12月从求职网站Indeed.com上获得的。（你可以在http://www.indeed.com/m/jobs?q=hadoop&l=san+francisco%2C+ca 上找到搜索结果。）我把在许多求职广告中列出的、快速增长的数据科学技能——Hadoop，跟其他技能相比较，包括大数据、Python、Java、SQL和其他程序语言开发。请注意，在这些搜索结果中，引号代表精确匹配（例如“数据科学”），宽泛匹配不包含引号。带引号的“数据科学”和数据科学的搜索结果，差异巨大，因为如在前面部分提到过的，数据科学家可以有数百个不同的职位头衔。带引号的“大数据”的搜索结果比大数据的搜索结果小15倍。另外要注意的是，旧金山是大数据、Hadoop和数据科学等大数据类职位最多的地方。排第二位的可能是伦

敦，那里的工作机会更多，薪水比纽约市更高。

# 根据技能和位置的薪酬分类

从以下给出的信息可以看出，什么工具可以给你带来大量就业机会；哪里有大量的就业机会，以及不同的数据科学技能的大致薪酬水平。括号中的数字代表每个薪酬级别对应的空缺数目。

Hadoop——旧金山，加州

- $60 000+ （2 196）
- $80 000+ （1 669）
- $100 000+ （865）
- $120 000+ （422）
- $140 000+ （210）

Hadoop——西雅图，华盛顿州

- $50 000+ （349）
- $70 000+ （248）
- $90 000+ （122）
- $110 000+ （47）
- $130 000+ （9）

Hadoop——纽约，纽约州

- $60 000+ （781）
- $80 000+ （557）
- $100 000+ （358）
- $120 000+ （204）
- $140 000+ （121）

Hadoop——芝加哥，伊利诺伊州

- $50 000+ （194）
- $70 000+ （150）
- $90 000+ （84）
- $110 000+ （40）
- $130 000+ （24）

Hadoop——洛杉矶，加州

- $50 000+ （320）
- $70 000+ （238）
- $90 000+ （113）
- $110 000+ （65）
- $130 000+ （29）

Hadoop——伦敦，英国

- £40 000+ （619）
- £60 000+ （351）
- £80 000+ （170）
- £100 000+ （97）
- £120 000+ （37）

大数据——旧金山，加州

- $60 000+ （4 388）
- $80 000+ （3 268）
- $100 000+ （1 735）
- $120 000+ （867）
- $140 000+ （433）

“大数据”——旧金山，加州

- $60 000+ （2 630）

- $80 000+ （1 990）
- $100 000+ （1 075）
- $120 000+ （581）
- $140 000+ （296）

数据科学——旧金山，加州

- $50 000+ （10 000）
- $70 000+ （7 739）
- $90 000+ （4 535）
- $110 000+ （2 060）
- $130 000+ （965）

“数据科学”——旧金山，加州

- $50 000+ （320）
- $70 000+ （219）
- $90 000+ （152）
- $110 000+ （68）
- $130 000+ （24）

Python——旧金山，加州

- $60 000+ （5 880）
- $80 000+ （4 024）
- $100 000+ （1 757）
- $120 000+ （761）
- $140 000+ （296）

SQL——旧金山，加州

- $60 000+ （7 455）
- $80 000+ （5 206）
- $100 000+ （2 348）
- $120 000+ （1 040）

- $140 000+ （420）

Java——旧金山，加州

- $60 000+ （10 266）
- $80 000+ （7 675）
- $100 000+ （3 527）
- $120 000+ （1 547）
- $140 000+ （649）

Perl——旧金山，加州

- $60 000+ （3 850）
- $80 000+ （2 528）
- $100 000+ （1 119）
- $120 000+ （445）
- $140 000+ （187）

统计学家——旧金山，加州

- $50 000+ （163）
- $70 000+ （111）
- $90 000+ （65）
- $110 000+ （33）
- $130 000+ （20）

“数据挖掘”——旧金山，加州

- $50 000+ （1 063）
- $70 000+ （772）
- $90 000+ （453）
- $110 000+ （257）
- $130 000+ （121）

“数据挖掘工程师”——旧金山，加州

- 没有可用的数据

`http://www.analyticbridge.com/m/group?id=2004291%3AGroup%3A140438` 这个链接是薪酬调查的一个重要参考，有些还会根据经验水平进行分类。

# 创建自己的薪酬调查表

可以使用和Indeed.com类似的网站，在搜索框中修改搜索关键词和位置，以便找到任何区域与数据科学相关的职位的数目和薪酬细分。（请参阅前几节，以确定潜在的职位名称。）许多职位都可以在LinkedIn和AnalyticTalent上找到，但以下公司的职位竞争激烈（几百个人申请一个职位）：谷歌、Twitter、Netflix，PayPal、eBay、LinkedIn、Facebook、雅虎、苹果——它们自数据科学开创以来就是典型的数据科学公司，还有Intel、Pivotal、IBM、微软、亚马逊和一些其他公司。不知名或坐落在不太受欢迎的城市的公司（例如，在俄亥俄州的银行或保险公司）吸引的申请人比较少，一个职位不到50个申请人。创业公司也是一个很好的工作渠道。

其他的职业信息来源包括Glassdoor.com和Wetfeet.com。

# 总结

本章的重点在于帮助你应聘数据科学家职位。本章讨论了90多个面试问题，来测试你的分析和视觉技能，对从统计学家到数据科学家的过渡进行了指导，还介绍了顶级数据科学家和数据科学的相关技能，以及数据科学家的职位名称和薪酬调查。另外还包括了一些小型数据科学项目的例子。

在接下来的最后一章，有抱负的在职数据科学家们，可以从中获取一些有用的资源：会议列表、简历范例和求职广告、数据科学书籍、术

语定义、数据集、组织、热门网站和云集许多数据科学家的公司等。

# 第8章
# 数据科学资源

第7章的重点是帮助你应聘数据科学家的职位。在本章你会发现很多资源，对你的专业生涯及职业发展有价值。

## 专业资源

本章提供了各类资源信息，包括数据集、书籍、会议、组织、网站及重要的概念，能在你推进数据科学职业发展时帮到你。

交叉参考　第3章“成为一名数据科学家”涵盖了训练计划、课程及认证。

参见第1章“历史与开拓者”一节，以及第2章“大数据生态系统”一节，可以了解供应商详情。

参见第7章“数据科学家的分类”一节可以了解当今著名的数据科学家。

## 数据集

Data Science Central包含一些数据集，放在`http://www.datasciencecentral.com/group/data-science-apprenticeship/forum/topics/update-about-our-data-science-apprenticeship` 中，如下。

- 大数据关键词相关性API所需的源代码及数据（还请参见第5章“关键词相关的API的源代码”一节）。
- 很棒的统计分析案例：预测陨石撞击（请参见第6章“预测陨石撞击”一节）。
- 用于基准测试的535亿个点击数据。
- 超过500万条金融、经济和社交媒体数据。
- 用于预测股价的新模式，可以将回报提升5倍（基于标普500的股市数据，参见第6章“股票市场”一节）。
- 35亿个网页：该网页链接图谱是从Common Crawl 2012网页语料库中提取出来的，包含35亿个网页和1280亿个超链接。
- 另外一个大数据集是：2.5亿个数据点。该数据集是1979年1月1日到2013年3月31日期间全部的GDELT事件数据，包含每个事件记录的所有字段。
- 还有125年的公开健康数据供下载。

此外，还可以从哈佛大学数据科学网站 `http://cs109.org/resources.php` 上找到一些额外的数据集。我对 `http://linkeddata.org/` 上的链接数据特别感兴趣。哈佛数据科学课程所包含的资源可以在`http://vcg.seas.harvard.edu/classes/computer-science-109-data-science-new-course` 上找到。

KDnuggets也是一个很棒的资源站，其获取地址为`http://www.kdnuggets.com/datasets/index.html` 。额外的资源还有英国的`http://data.gov.uk/` ，以及美国类似的行动计划（参见`http://onforb.es/1m0W8cU`）。

# 书籍

对数据科学家有用的书籍能分成几类：可视化、大数据/Hadoop、统计学/机器学习、纯数据科学、商业分析、决策者该掌握的数据科学（比如招聘和项目管理），以及其他类别。

Data Science Central上列出了100多本书，

在http://www.datasciencecentral.com/profiles/blogs/more-than-100-data-science-analytics-big-data-visualization-books 和 http://www.analyticbridge.com/group/books 上可以看到。

下面是上述两个网页中的一些书名和参考资料。这些书在数据科学圈（比如KDnuggets）受到推荐和赞赏，在Data Science Central社区里也很受欢迎。这两参考网页上有书的各种细节（内容简介、作者、出版日期等），还有购书网站的链接（通常是亚马逊的），让你可以快捷地找到并购买。有些标题对应的资料是期刊，有些是丛书（比如其中包含5本关于数据可视化的书），也有一些是免费的PDF文档。

这些书都是通识类型的，而不是特定行业或特定问题的专著。上述网页中包含了更全面的清单，其中有不少专著。

商业类别

- *Data Science for Business* , O'Reilly, 2013
- *Big Data Computing* , CRC Press, 2013
- *Past, Present, and Future of Statistical Science,* CRC Press, 2014
- *The Field Guide to Data Science* ，免费PDF文档，出自Booz Allen Hamilton, 2013
- *Implementing Analytics* , Elsevier, 2013
- *Automate This: How Algorithms Came to Rule Our World* , Portfolio Hardcover, 2012
- *Analyzing the Analyzers* , O'Reilly, 2013
- *Business Analytics: A Practitioner's Guide* , Springer, 2013
- *A Practitioner's Guide to Business Analytics* , McGraw-Hill, 2013
- *Delivering Business Analytics: Practical Guidelines for Best Practice* , Wiley, 2013
- *Building Data Science Teams* , O'Reilly, 2011

技术类别

- *Practical Text Mining and Statistical Analysis for Non-structured Text Data Applications* , Academic Press, 2012
- *Data Mining* , Wiley-IEEE Press, 2011

- *Encyclopedia of Machine Learning* , Springer, 2010
- *Analyzing Data from Facebook, Twitter, LinkedIn, and Other Social Media* , O'Reilly, 2013
- *Ensemble Methods in Data Mining: Improving Accuracy Through Combining Predictions* , Morgan & Claypool出版社, 2010
- *Causality: Models, Reasoning, and Inference* , Cambridge University Press, 2000
- Mining of Massive Datasets，免费版本的下载地址是 http://infolab.stanford.edu/~ullman/mmds/booka.pdf.stanford.e
- Applied Data Science,哥伦比亚大学课程
- *Forecasting: principles and practice* ，2013，免费版本的下载地址是https://www.otexts.org/fpp/
- *Alternative Methods of Regression* , John Wiley & Sons, 1993

丛书

- 8本关于数据分析的计算统计丛书，CRC出版社出版，参见 http://www.analyticbridge.com/group/books/forum/topics/new-computational-statistics-books-for-analyzing-your-data
- 两本关于程序编写面试的书，参见 http://www.analyticbridge.com/group/books/forum/topics/two-books-on-programming-interviews
- 60本统计教材，参见http://bit.ly/1hnYZKa
- 5本关于数据可视化的书，参见 http://www.analyticbridge.com/group/books/forum/topics/5-books-on-data-visua-ization

程序编写工具

- *Big Data Analytics with R and Hadoop* , Packt Publishing, 2013
- *Practical Data Science with R* ，书正在筹备中，参见 http://www.manning.com/zumel/.
- *Predictive Analytics: Microsoft Excel* , Que Publishing, 2012
- *Data Analysis with Open Source Tools* , O'Reilly, 2010
- *Data Mining: Discovering and Visualizing Patterns with Python* ，免费下载地址是http://refcardz.dzone.com/refcardz/data-mining-

discovering-and

- *Hadoop in Practice* , Manning Publications, 2012
- *Hadoop: the Definitive Guide* , O'Reilly, 2012

可视化

- *Visualizing Data* , O'Reilly, 2007

杂志

- *Big Data*，参见http://www.liebertpub.com/big
- *EPJ Data Science*，参见http://www.epjdatascience.com/
- *Journal of Data Science*，参见http://www.jds-online.com/
- *Data Science Journal*，参见http://www.codata.org/dsj/index.html
- *Decision Analytics* , Springer，参见http://decisionanalyticsjournal.com/
- *The R Journal*，参见http://journal.r-project.org/
- *Information Visualization*，参见http://ivi.sagepub.com/

# 会议与组织

专业会议的参与者基本上有3类组织：供应商、专业协会以及组织会议的公司。下面是一些关键的会议和相关的组织。

供应商

SAS会组织每年的数据挖掘大会（Data Mining Conference）。Pivotal和VentureBeat以及Data Science Central会一起组织数据科学峰会（Data Science Summit，http://venturebeat.com/events/databeat2013/ ）。像Teradata、Hadoop（http://hadoopsummit.org/ ）、Cloudera、Alpine Labs和Hortonworks等其他供应商也都会举办他们自己的会议。

专业协会

基本上有两类数据科学的专业协会。

- IEEE或Direct Marketing Association（DMA，参见 http://thedma.org ），其专注领域比数据科学更广泛，近期开始组织数据分析、大数据和数据科学活动。
- 其他协会则更专注于数据分析，比如INFORMS（运筹学）、美国统计协会、数字分析协会和国际分析研究所。

### 会议组织者

其中几个最活跃的会议组织者如下。

- IE Group在全球组织了许多大数据、Hadoop及商业分析会议。（http://theinnovationenterprise.com/summits/calendar ）
- O'Reilly负责组织“Strata”会议。（http://strataconf.com/ ）
- Rising Media组织了很多“Predictive Analytics World” 会议。（http://www.predictiveanalyticsworld.com/ ）
- FC Business Intelligence则因为文本挖掘等会议而闻名。（http://www.datadrivenbiz.com/ ）
- IQPC则组织“Big Data for Government & Defense（政府及防卫大数据）”会议（http://www.iqpc.com/ ）
- White Hall Media组织了“Big Data Analytics”和其他会议。（http://whitehallmedia.co.uk/wmconferences/ ）

# 网站

Data Science Central根据成员的建议，已经汇集了跟数据分析、数据科学或大数据有关的网站列表。（这些排名高的域名，至少都被4位成员推荐过。）这个列表包括供应商、出版社、大学、组织，以及知名数据科学家的个人博客。其中有些是纯数据科学网站，而其他的则比较通识，但技术倾向性也很明显，很大比重都在强调数据议题，或者定期涉及数据科学内容。

由于这个列表在持续更新，所以可以访问 http://www.datasciencecentral.com/profiles/blogs/175-analytic-and-data-science-web-sites 以保证得到最新的列表。你肯定也想给我们提供建议对吧！

# 概念定义

本节有几个精选的名词，你有必要理解，可能在职业场合中用得到。可以访问 http://www.analyticbridge.com/profiles/blogs/2004291:BlogPost:223 了解这些名词的细节，以及这些概念的贡献者。

- 调整后的 $R^2$ （R-Squared）：是统计学家偏好的方法，用来决定把哪个变量添加到模型里。它是$R^2$ 的修改版，$R^2$ 是根据每个变量的投入程度，对该变量给予的惩罚。因为$R^2$ 的特点，在添加新变量时，它会不断上升，从而导致模型过度拟合数据，预测能力不佳。可以在更保守的模型里用调整后的$R^2$ ，仅当拟合提升程度比惩罚大时，才添加新变量，这样能提升“样本外（out-of-sample）预测”这个终极目标。
- 聚类分析： 聚类分析是将一组对象分配到不同组别的方法。这些组别称为聚类，一个聚类里的对象彼此的相似度高于跟其他类的对象的相似度。知名的聚类分析算法有层级聚类、k-means、模糊聚类和监督聚类等。
- 交叉验证： 交叉验证是一种通用的高计算量方法，用来估计统计模型的准确度。该验证方法是将数据分为N个子集，将一个子集放在一边，从余下的N-1个子集里估计模型参数，并使用预留的那个子集来估计模型错误率。这个流程会重复N次，N个子集里的每个子集都会被作为一次交叉验证集使用。然后这N次实验获得的错误率合并起来，就能得到模型错误率的最终估计值。
- 决策树： 根据单个数据向量里的值，决策树可以为终端用户输出结论。经典的例子就是根据特定病人的病症进行医疗诊断。数据科学里一个普遍的问题是基于跟已知结论相关的大量数据自动或者半自动地生成决策树。这类算法包括CART和ID3。
- 实验设计（**Design of experiments**）：英文有时还称作experimental

design。这是一套通过抽样、分组观测、测试统计模型来探测根源或有影响力的预测因素的方法。

- 探索式数据分析： 也称为EDA（Exploratory Data Analysis）。在所有的统计分析中，这是数据收集后的第一步：观察相互作用、可视化、异常检测，用数据字典总结数据。
- 因子分析： 用于减少变量的技术，来识别聚类所关联的变量组。
- 特征选择： 特征是一个变量，特征选择是在万亿潜在的特征组合中，检测出那些有着好的预测能力且健壮的特征。
- 一般线性模型： 一般（或广义的）线性模型（GLM），相比线性模型来说，允许你在一个因变量和N个自变量之间，描述附加性的和非附加性的关系。一般线性模型中的自变量可以是连续的，也可以是离散的。（因变量一般被叫作反应变量 ，自变量被叫作因子或者协变量 ，这取决于它们是否被控制。）
- 拟合度： 交叉验证测试中，模型预测值把错误最小化的程度。然而，数据过拟合是比较危险的，因为它将会导致模型对新数据没有预测能力。真正的拟合度是由一个模型如何适合于新数据来决定的，比如这个模型的预测能力。
- Hadoop：开源框架，支持大规模数据分析，通过将问题分解成独立执行的离散块，来靠近数据切片，并最终重新整合出问题的答案。相对于传统的数据库框架来说，这更像是一个文件管理系统，尽管有一些SQL层已经建立在它上面。
- 隐性决策树： 交易数据评分的方法。融合了线性和非线性分类器，隐含地建立和混合多种小决策树（非线性分类器），并最终合并，重新校准两个分数，生成唯一的得分。它快速且有效，但是要求有特征选择的专业知识，尽管这个过程可以使用本书中描述的快速特征选择算法自动实现。
- **K-means** ：一种流行的聚类算法，给定（假定的）一个K，通过反复移动聚类的中心位置，调节聚类分配来找到K个聚类。
- 逻辑回归： 基于二进制数据，对某个特定结果发生概率进行建模的回归方法。也可用于响应值是概率时的回归。
- 机器学习： 这套技术通常描述为“算法”，基于训练数据集进行数据分类。训练集构成了学习部分，针对它的讨论更多是设计自动或半自动的学习系统。
- **Mahout** ：Mahout是Apache项目，实现免费的分布式或以其他可扩展的机器学习算法，主要集中在协同过滤、聚类和分类，通常会利用Hadoop平台，但也不限于Hadoop平台。
- **MapReduce** ：用于高效处理大量数据的模型。将原始问题“映

射”到较小的问题（小问题再变成“原始”问题）。接下来并行处理这些小的问题。组合较小问题的结果，或者“归总”出原始问题的解答。

- 蒙特卡洛（**Monte Carlo**）模拟： 用许多随机采样的值计算随机现象模型的期望值和概率。类似于通过反复放置和反复计算成功率来计算，计算赢得赌博轮盘的概率（比如是黑）。它对有不确定性特征的复杂模型来说是有用的。
- 自然语言处理（**NLP**） ：指自动处理文本来提取信息，比如情感分析或者自动生成分类树或摘要的技术。已经从词统计（词袋）演变到更精细的文本挖掘技术。
- **NoSQL**：“Not only SQL”是一组数据库管理系统。数据不像关系型数据库那样存储在表中，也不是基于表和表之间的数学关系。这是一种快速检索非结构化数据的存储方式。
- 多维尺度： 通过将尺寸为$N^2$ （N=观察数）的相似矩阵投影成二维的视觉表达来降低空间维度。一个典型的例子是要生成一个城市的地图，但唯一可用的数据是任意一对城市之间的旅行时间。
- 朴素贝叶斯： 朴素贝叶斯是一种基于贝叶斯理论、适用于类别式数据的简单分类方法。它快速、容易实现，但是它假设变量（也叫作特征）是独立的。在实践中，情况并非如此，基于朴素贝叶斯的算法表现不一定好（比如垃圾邮件检测）。
- 非参数统计： 用这套统计技术处理数据时，不对有关的统计观测值进行假设。相对于模型驱动统计来说，它也被称作数据驱动统计。
- Pig：Pig是一种Hadoop的脚本接口，它使你在即使缺乏MapReduce编程经验的情况下，也不会让Hadoop拖你的后腿。它也因处理大量各种不同的数据类型而被大家知晓。
- 预测模型： 基于统计模型来进行预测（股票市场、欺诈检测、用户转化成销售的概率）的一组技术，通常和预测值的置信区间一起。
- 灵敏度分析：确定预测模型对噪声、缺失值、离群值和其他异常等模型预测因素的敏感性的过程。
- 六西格玛： 一组流程改进的工具和策略，最初是由摩托罗拉于1985年提出的。六西格玛旨在通过识别和消除缺陷（错误）的原因，并在生产和业务流程中最大限度地减少可变性，提高流程产出的质量。
- 逐步回归： 多变量回归的变量选择过程。在向前逐步选择的过程中，选择一个种子变量，把每个额外变量输入到模型，但是只保留

显著改善拟合度（如测量$R^2$ 的增量）的变量。向后选择从所有变量开始，一个接一个地移除，直到移除多一个的变量会使$R^2$ 减少较多为止。这个方法的两个缺点是：选择的种子会不成比例地影响被保留的变量，变量的去留是由$R^2$ 决定的，而不是由调整后的$R^2$ 决定的。

- 监督聚类： 也被称为判别分析。已有一个预分类观测值的集合（训练集），再对新的数据点进行分类，也就是说观测值的类标是已知的。
- 时间序列： 一组（t, x）值，其中x通常是标量（尽管也可能是向量），t值通常以规则间隔采样（尽管有些时间序列不规则采样）。在规则采样时间序列的情形下，实际数据会忽略t，而使用$t_0$（开始时间）及增量t应用于整个时间序列。

# 职业建设资源

在本节中，你可以找到不同公司雇用和按例行情况聘请数据科学家的信息。本节还包含了简历样本和招聘广告，对识别现在在数据科学界随处可见的热门关键词和技能（例如，R、Python、Hadoop、NoSQL、SQL、预测模型、机器学习等）来说，算是一个宝藏。

# 招聘数据科学家的公司

有多种机构会定期聘请数据科学家，尽管可能在招聘广告中使用不同的职位名称。你可能想不到有一小部分公司也会在这个列表中，包括沃尔玛（Walmart）、普华永道（PwC）、电子艺界（Electronic Arts）、波音（Boeing）和星巴克（Starbucks）。在列表顶部的公司（IBM、微软），其员工中的数据科学家占比要比其他像Facebook、谷歌、FICO、eBay或者领英这些公司的数据科学家占比要少。传统工业，如制造业，比较趋向于用“运筹学分析师”来称呼该职位。

很多工作——特别是高级角色——仍然集中在美国（纽约和旧金山），但是有些地区也已经在紧追了，比如新加坡、西班牙、爱尔兰和

伦敦。下面是招聘大量数据科学家的公司列表。这个列表是基于我自己10 000多LinkedIn的链接，根据公司进行细分并根据链接数进行排序的。前20个公司占据数据科学家岗位招聘的10%左右，但是该分布有很长的异常厚重的尾巴。你可以在`http://www.datasciencecentral.com/profiles/blogs/6000-companies-hiring-data-scientists`中找到更全面的6 000多家公司。

- 微软
- IBM
- 亚马逊
- SAS
- 谷歌
- 埃森哲
- Oracle
- 领英
- FICO
- 美国银行
- 花旗
- 塔塔咨询服务公司
- Facebook
- 高知特信息技术公司
- 富国银行
- 凯捷
- eBay
- 苹果
- 惠普
- EMC
- Pivotal

# 数据科学招聘广告的样本

如果没有当下公司对数据科学领域要求的信息和其他有用的信息，那么本章就不能称之为完整。以`http://www.analyticbridge.com/profiles/blogs/selected-job-ads-`

for-readers-of-our-data-science-book 中的招聘广告样本为例，这也是最新而真实的样本，里面频繁提到的技术包括：Python、Linux、UNIX、MySQL、MapReduce、Hadoop、Matlab、SAS、Java、R、SPSS、Hive、Pig、Scala、Ruby、Cassandra、SQL Server和NoSQL。

交叉参考 参见第7章“踏上你的数据科学职业之路”，有更多关于如何以及在哪里搜索不同的职位、级别和技能的信息。

# 简历样本

下面的简历样本摘录，来自一位同意在本书中引用该内容的实际数据科学工作者。为了能让这些专家删除或更新他们的简历，我将这些简历放在了http://www.analyticbridge.com/profiles/blogs/selected-resumes-for-readers-of-our-data-science-book 中。你可以通过在LinkedIn搜索数据科学的关键词，或者在浏览器上搜索Data Science Central会员的简介来找到更多的简历和简介。

包括在此名单中的人，有不同的语言环境和背景，以便覆盖数据科学的不同层面。这份专业数据分析人士的名单，重点是提呈现一个比较衡的组合——既包括初级和高级人才，也包括有大公司经验、创业经验或者两者兼备的人才，还有顶级明星和简历常规的人（有时是最忠实的员工），以及企业顾问或学术界相关人士。这份名单旨在帮你找到一个或者更多与你相关类似的专业人士。我也把自己的简历加了进去，我是一个拥有专利和经典的大数据科学经验的参考范例（比如具备信用卡欺诈检测和数字分析经验）。

这些简历中提到的典型的技术有：R程序语言、Python、Matlab、MongoDB、SQL、MySQL、统计/机器学习（KNN、决策树、神经网络、线性/逻辑回归），最后还有Java、JavaScript、Tableau、Excel、推荐引擎和谷歌分析。当然，没有简历会包含所有被罗列出来的技术，但是在50%的简历中最经常被列出的技术是R语言和Python。

你应该查看一下这些简历，看看这些人（横向或纵向）的职业发展、学位、所参加过的培训以及证书。

通过比较这些简历和先前所提的招聘广告，人力资源部门有时似乎在寻找独角兽——一个不可能存在的、混合了所有技能的专家。我鼓励招聘者寻找并聘用有着强大潜能的人并且训练他们，而不是寻找稀少且价值不菲的独角兽，他们也通常不那么适合（可能只愿意开心地做自己的事业）。

注意　在数据科学行业不断有很多讨论，比如数据科学项目的理想团队是有一个或两个万能的“独角兽”好，还是一个多元化的团队好。在现实中，多元化的团队结构是最常见的，因为很难找到一个人拥有所有必需的技能。但是如果可能的话，人力资源需要尝试聘用某些有着很深专业知识、商业头脑、代码经历的人（生产代码，除非该职位是做原型算法），具备分析背景（在统计、大数据和工程/计算机方面有真正的经验），然后让他学习统计、Hadoop（作为用户而不是架构师）、核心数据科学或者计算机科学技术。如果一家公司能在一个人身上找到所有想要的条件（可以不具备数据科学核心技术，因为合格的人可以轻松地学会），就能避免竞争和人才缺陷。但是要注意，这些有技术、多价值的人可能不会像想象中的那样待那么长时间。

# 总结

本章提到了一些数据科学的有用资源。训练数据科学家的资源包括数据集、数据科学书籍和期刊、文献、一些组织、流行的网站和专业术语。职业发展的资源包括有着很多数据科学家的公司的信息、招聘广告和简历范例。