| 光明社科文库 |

R语言在政治学中的应用

吴　江◎著

光明日报出版社

| 光明社科文库 |

R语言在政治学中的应用

吴　江◎著

光明日报出版社

图书在版编目（CIP）数据

R 语言在政治学中的应用 / 吴江著 .-- 北京：光明日报出版社，2018.12

ISBN 978-7-5194-4796-0

Ⅰ. ① R… Ⅱ. ①吴… Ⅲ. ①程序语言—程序设计—应用—政治学—研究 Ⅳ. ① D0-39

中国版本图书馆 CIP 数据核字（2018）第 276668 号

R 语言在政治学中的应用

R YUYAN ZAI ZHENGZHIXUE ZHONGDE YINGYONG

著　　者：吴　江

责任编辑：刘兴华　　责任校对：赵鸣鸣

封面设计：中联学林　　责任印制：曹　净

出版发行：光明日报出版社

地　　址：北京市西城区永安路 106 号，100050

电　　话：010-67078251（咨询），63131930(邮购)

传　　真：010-67078227，67078255

网　　址：http://book.gmw.cn

E - mail：liuxinghua@gmw.cn

法律顾问：北京德恒律师事务所龚柳方律师，电话：010-67019571

印　　刷：三河市华东印刷有限公司

装　　订：三河市华东印刷有限公司

本书如有破损、缺页、装订错误，请与本社联系调换

开　　本：170mm × 240mm

字　　数：269 千字　　印张：16.5

版　　次：2019 年 1 月第 1 版　　印次：2019 年 1 月第 1 次印刷

书　　号：ISBN 978-7-5194-4796-0

定　　价：78.00 元

前 言

本书的目标读者是政治学研究者，以及希望用数据分析工具满足政治领域工作需要的读者。当然，书中讲到的分析方法多是各领域通用的，因此也适用于管理学、社会学、传播学等方面的研究和实践。

R 语言是目前世界上最受欢迎的数据分析软件之一，并且已经在生物学、医学、地理学、经济学、心理学、管理学、政治学等领域得到了广泛应用。

总的来看，R 具有以下优势：

（1）R 具有强大的数据获取、数据清理、统计建模和数据可视化功能。

（2）R 用户能够利用数量庞大且不断更新的 R 包，这些 R 包的功能异常丰富；特别是，在目前已存在的各种算法及其变体（包括各种前沿算法）中，很多都可以通过相应的 R 包加以实现。有的学者在提出新的算法后，就会将其开发成 R 包供他人使用。相比之下，一些商业软件或是非编程数据分析工具的功能相当有限，且更新缓慢。

（3）R 代码灵活，直观，可读性强，使得用户在利用现有函数的同时还可以轻松编写满足各种特殊用途的函数。可以这样比喻，R 语言是一台单反相机，它的各项功能都允许用户自行调整参数并进行搭配组合。现在研究者总是强调 idea 的重要性，而 R 正好就是在数据分析方面帮助研究者实现 idea 的强大工具。

（4）R 在软件包的下载、安装、讲解和上传方面非常规范（典型特征是，所有的 R 包的页面和使用手册都有相同的格式），为学习和使用

提供了极大便利。

（5）学习和使用R的过程有助于加深对数据分析和编程的理解，在此基础上，人们可以学习复杂的算法和其他编程软件（如Python）。

R在政治学方面的应用是指两个方面：首先，这是指那些在多个领域通用且可用R来实现的数据处理方法，也可用于政治领域的研究和实践，如回归模型、文本分析等。其次，在可由R实现的功能中，一些功能相较于其他功能更能满足政治领域的需要。

本书介绍的R语言功能主要涉及：排序和打分、文本分析、综合评价指标、绩效评估、定性比较分析、项目反应理论等。其中一些计算方法，如投票计票、指标权重的设定，不单涉及政治学学术研究，而且在一定程度上偏向政治领域的实践，可直接应用于实际工作。

与其他R语言相关书籍相比，本书具有以下特点：

（1）本书介绍的一些数据分析方法相对来讲在政治学领域尚未得到广泛应用。一提起数据分析在政治学中的应用，人们通常想到的都是线性回归模型、广义线性模型、混合效应模型、结构方程模型等用来进行假设检验的数据分析工具。但实际上，还有很多其他方法可以使用。这些方法要么尚未在政治学研究中得到充分重视；要么虽然近年来逐渐受到人们重视，但将其放在政治学情境中进行详细介绍的资料相对较少。写作本书的目的正是对这些分析方法中的一部分进行介绍，使读者能够借助R语言运用这些方法。

（2）本书无意介绍R语言的安装、基本操作等。这些内容已出现在很多R语言书籍中，本书没有必要进行重复介绍。因此，本书适合那些已初步掌握R语言操作的读者学习。不过本书附录列举了一些基础代码，希望对R还不太熟悉的读者在学习正文中的内容之前先跑一下这些代码。

（3）本书重在讲解数据分析操作的细节，而并不只是提供所谓"干货"。一些现有的R相关书籍只讲解"干货"或"实战"，缺少对细节问题的解释，这使得学习者难以将学到的方法转移到有所变化的新的案

例上去（典型情况是，在使用一个函数时，学习者甚至都不知道自己手头的数据是否符合这个函数对输入数据的要求）。尽管本书并不介绍基本操作，但在讲解过程中亦对必要的描述性统计和作图方法进行了介绍。另外，本书也更多地关注如何设置参数以及如何对输出结果进行解读，而这些恰恰是困扰许多 R 学习者的问题。

在内容的连续性方面：从某种角度讲，本书大部分内容都是围绕政治学和公共管理领域中极其常见的评价工作展开的。比如说，专家对各项政策的科学性进行打分，这是在进行评价；而选举则可看成是选民在对候选人进行评价；词语联想亦可看成是被试对词语之间关联的评价。在数学原理层面，这些内容之间都具有内在的关联，这一点体现了本书的连续性。但从另一个角度讲，人们在日常工作中会更多地结合应用情境来选择分析工具，有鉴于此，笔者从应用情境的角度对各章内容进行了划分。本书各部分内容相对独立，读者可根据需要有选择地学习。

在写作本书的过程中，清华大学的苏毓淞老师为笔者提供了无私的帮助，在此表示衷心感谢！

本书提到的全部代码和配套数据都可从网上下载。除了读取硬盘上的数据和从网上下载 R 包的部分，所有代码都可直接复制粘贴到 R 中并跑出计算结果。

配套数据和代码下载地址：

https://github.com/githubwwwjjj/rpolitics

如果无法正常下载，可在以下页面查看其他下载方式：

https://github.com/githubwwwjjj/votesys

如果仍无法正常下载，可联系作者：

textidea@sina.com

目　录

CONTENTS

第1章 排序和打分

排序（ranking）是指评价者依据某种标准（如科学性、可行性、个人偏好等）为多个选项排出顺序，最优选项被赋值为1，次优选项被赋值为2……以此类推。打分（rating），又可称为定级或等级评估，是指评价者按照某种标准对各选项的类别或所处的水平进行评定；例如，用1至5的数字表示对一项政策的偏好程度，1代表完全赞同，5代表完全反对（或者1代表完全反对，5代表完全赞同），其他数字表示居间的偏好程度。

排序与打分在政治学研究和政治实践中有着广泛的应用。在公共政策领域，决策者会邀请若干专家对各种备选方案进行排序，或对每个方案的各项属性进行打分，并据此挑选最优方案。决策者亦可能开展线上调查，请公众对各类公共服务的质量（如效率、便利程度等）进行评价。不过，对排序与打分最为直接的应用要算是选举投票了。投票实质上就是对公众偏好进行聚合从而选出优胜者。在政治学研究领域，我们有时需要利用专家打分法来确定变量的取值并将其用于后续分析；例如，在衡量一个国家或地区的法治水平时，我们除了使用一些客观数据外，还可能需要邀请专家对其进行主观评估和打分；而当测量对象的维度较多以至于难以用客观数据测量或客观数据不可得时，专家打分法更是成了必不可少的工具。

排序与打分在一定程度上是可以互换的。例如，评价者对四个选项的排序结果为D>A>B>C，这实际上就相当于为A、B、C、D四个项目打分，分值分别为2（A排在第2位）、3（B排在第3位）、4、1，较小的数值代表较高的满意度。波达计数法（Borda method）是典型的排序与打分可互换的例子。

本章分为四个部分：第1部分介绍测量评价者间信度的一般方法；第2部分集中介绍选举计票机制；第3部分介绍拟合LLBT模型、模式模型和

Plackett-Luce 模型的方法；第4部分将对多种排序聚合方法进行介绍。

一、评价者间信度的测量

评价者间信度，是指多个评价者在赋分或分类时的一致性或相符性。

为什么要关注评价者间信度呢？设想，公共部门聘请一位资深专家对本地区数百个社区的基础设施建设水平进行评估打分。这位专家意识到自己不可能前往所有社区进行实地评估，于是在征得公共部门同意后雇佣了三名助手以便分头完成实地评估工作。现在的问题是，这三名助手的打分能否与专家保持一致？也许有人标准过严，因此给分普遍较低，也许有人标准过松，给分偏高，也许有人所使用的标准与专家心中的标准非常不同。但是，公共部门视这位专家的打分标准为黄金标准，因此当然希望其他人的打分能与他保持一致。为检验这种一致性，这位专家带着三名助手前往他已经给出分值的15个社区，让他们每个人为这些社区打分。然后，这位专家要做的就是通过特定的指标来衡量三个助手与他在打分方面的一致性程度。

再举一例，设想某个研究者提出了某种对国家政体进行分类的新方法，并希望将以此方法得到的国家类别作为自变量放入回归模型中进行假设检验。他现在意识到，如果政体的划分只由他个人完成的话，期刊编辑会说他的分类主观性较强；于是，这个研究者又找来另一个研究者，把分类标准告诉他，并请他跟自己一起完成分类。理想的情况是，两个研究者对每个国家的政体都作出了同样的判断；不过两个人的分类结果并不一定完全一致。这时，研究者需要计算并报告信度指标的数值，借此证明两个人的分类具有较高的一致性，然后通过讨论解决分类中的分歧。

接下来我们就将介绍如何用 R 计算各种信度系数。

正如数值可分为名义（nominal）变量或分类变量、定序（ordinal）变量、定距（interval）变量和定比（ratio）变量一样，信度系数也可被分为四类。我们本应依次对这四类系数进行介绍，但鉴于 R 中的一些函数能够计算不止一类系数，因此我们将不按照这四个类别的顺序进行介绍。表1对这些函数进行了总结，读者可从中查找自己需要的函数。

表 1　信度系数计算函数分类

	分类变量	定序变量	定距变量	定比变量
仅限两个评价者	rel::gac rel::spi manual_bp immer::immer_agree2 Delta::Delta	rel::ckap rel::gac rel::spi	rel::gac	rel::gac
两个或多个评价者	rel::ckap rel::kra	rel::kra irr::kendall	rel::kra	rel::kra obs.agree::IBMD

我们首先来看基础信度的计算方法。

```
freqtable=matrix(c(239, 16, 6, 21, 73, 9, 16, 4, 280), nrow=3, dimnames=list(c("type 1", "type 2", "type 3"), c("type 1", "type 2", "type 3")))
freqtable
#          type 1  type 2  type 3
# type 1     239      21      16
# type 2       6      73       4
# type 3       6       9     280
```

两个评价者对 sum(freqtable)=664 个项目进行分类，可选的类别为 type 1、type 2、type 3；对两人的分类结果进行汇总，得到频数表。表中数值含义如下：239 代表同时被两个评价者划分为 type 1 的项目有 239 个，21 代表被第 1 个评价者分到 type 1、被第 2 个评价者分到 type 2 的项目有 21 个……以此类推。这样，对角线的总和就是分类一致的项目的总数，其他单元格则显示的是分类不一致的情况

```
n=sum(freqtable)
the_same=sum(diag(freqtable))
pa=the_same/n
# [1] 0.8915663
n=sum(freqtable)
the_same=sum(diag(freqtable))
pa=the_same/n
# [1] 0.8915663
```

现在我们已经得到了基础的信度系数。不过，这个系数并不令人满意，原因在于，即使两个评价者胡乱猜测，他们给出的类别也会有一些是一致的。这就是说，我们在计算信度系数时，要减去这种偶然产生的一致性。这样，计算公式就变为 (pa-pe)/(1-pe)，其中，pa 就是上文计算出的基础系数，pe 代表因偶然因素而给出一致分类的概率。以此公式来计算的若干种信度系数可统称为概率校正系数（chance corrected coefficients），如 Cohen's Alpha（Cohen, 1960）、Scott's Pi（Scott, 1955）、Gwet's AC（Gwet, 2008）、Brennan-Prediger 系 数（Brennan and Prediger, 1981）、Krippendorff's Alpha（Krippendorff, 2004: 221-243）、Aickin's Alpha（Aickin, 1990）等。这些系数的差异在于计算 pe 的方法不同（但是 Krippendorff's Alpha 计算 pa 的方法亦不同于其他系数）。

```
# dat=read.csv("rbookdata/qog_rating.csv", row.names=1) # 配套资料中的文件
# 数据来自 https://qog.pol.gu.se/data。该数据是原始数据的一部分，记录的是18名打分者对某国的公共部门雇佣职员的公平性（q2_a 至 q2_j）和清廉程度（q8_a 至 q8_g）的评估。题项采用的是评价语句所描述的现象是否常见的形式，典型语句为 " 公共部门通过正式考试系统招聘雇员 "；选项为1至7的数字，1代表很少见，7代表很常见；部分题项为反向题，不过配套资料中的分值已经过调整，所有题项中的较小分值都代表情况令人满意

# install.packages("rel")
library(rel)

# rel 包中的 ckap 函数可以计算两个评价者的定序打分数据的 Kappa 值，或两个及以上评价者（Conger, 1980）给出的判断为分类变量时的 Kappa 值
# 假设我们需要计算第14和第18个评价者之间的一致性
y=ckap(dat[, c(14, 18)], weight="unweighted", std.err="Cohen")
#          Estimate     StdErr     LowerCB   UpperCB
# Const   0.323009   0.154804   -0.005161   0.6512

# ckap 函数中 weight 参数的默认值为 "unweighted"，即不对频数表中的单元格进行加权；还可改为 "linear"，单元格的权重随着单元格与对角线的距离的增加而线性递减；当为
```

```
"quadratic" 时，权重随着单元格与对角线距离的增加而指数递减
# std.err 参数用来指定计算标准误的方法，可以改为 "Cohen"，但默认值为 "Fleiss"。Fleiss 等人（1969）认为 Cohen 法易错误地得出较大的标准误，故对算法进行了调整
# 当 std.err="Fleiss" 时，计算标准误的方法改为 Bootstrap 法，用 R 指定 Bootstrap 的次数
# 另外，conf.level 可用来指定置信区间，默认值为 0.95
y=ckap(dat[, c(14, 18)], weight="unweighted", std.err="Fleiss", R=100)

# 再来看分类变量的例子
# 假设两名评价者被要求将若干个项目分成三类
freqtable=matrix(c(239, 16, 6, 21, 73, 9, 16, 4, 280), nrow=3, dimnames=list(c("type 1", "type 2", "type 3"), c("type 1", "type 2", "type 3")))
# 数据为已汇总好的频数表，但实际上我们现在需要使用原始数据；因此用以下函数进行转化
ta2raw=function(x){
    if (is.table(x)) x=as(x, "matrix")
    stopifnot(nrow(x)==ncol(x))
    ijpair=expand.grid(1: nrow(x), 1: nrow(x))
    repn=as.numeric(x)
    repn=rep(1: nrow(ijpair), repn)
    y=ijpair[repn, ]
    colnames(y)=c("p1", "p2")
    rownames(y)=NULL
    y
}

cat_x=ta2raw(freqtable) # 现在得到的是原始数据
y=ckap(cat_x, R=100)
#          Estimate    StdErr    LowerCB    UpperCB
# Const    0.823444  0.019275   0.785597     0.8613

# rel 包中的 gac 函数用来计算两评价者 Gwet's AC 系数，变量可为分类、定序、定距或定
```

比变量

在 gac 系数中可使用 weight 指定单元格权重，默认为 "unweighted"，可改为 "linear" 或 "quadratic"，解释见以上对 ckap 函数的介绍。当使用这三种方法时，该系数又称为 AC1 系数；如果权重为用户给定的数值矩阵，则称为 AC2 系数。当变量为定比变量时，还可将 weight 设为 "ratio"

当变量为分类变量或定序变量时，用 kat 指定可能出现的类别的数量

```
y=gac(dat[, c(14, 18)], kat=7)
#            Estimate      StdErr    LowerCB     UpperCB
# Const      0.394659    0.141544   0.094599      0.6947
```

如果使用 AC2 法的话，由于本数据中的等级数为 7，频数表为一个 7*7 的矩阵，相应地，权重也应该是一个 7*7 的矩阵，每一个单元格中的权重数值与频数表中的数值相对应。我们在此随机生成一个权重矩阵，仅供示范，实际使用时，应根据需要确定权重

```
set.seed(1)
w=matrix(sample(0: 1, 49, replace=TRUE), nrow=7)
y=gac(dat[, c(14, 18)], kat=7, weight=w)
```

rel 包中的 kra 函数用来为有两个或以上评价者时的分类、定序、定距、定比评分计算 Krippendorff's Alpha 系数；置信区间用 Bootstrap 法计算

必须用参数 metric 指定变量类型，分别为 "nominal"、"ordinal"、"interval"、"ratio"

kra 没有 weight 和 std.err 参数

```
set.seed(1)
y=kra(dat[, 1: 5], metric="ordinal", R=100) # 一次性计算5名评价者之间的信度，并设 Bootstrap 数为 100
#            Estimate    LowerCB    UpperCB
# Const      0.403036    0.062708      0.6277
```

rel 包中的 spi 函数用来计算有两个评价者且变量为分类变量或定序变量时的 Scott's pi 系数

可用 weight 指定每个单元格的权重，默认为 "unweighted"，可改为 "linear" 或 "quadratic"

```
y=spi(dat[, c(14, 18)])
```

```
#          Estimate    StdErr     LowerCB     UpperCB
# Const   0.296552   0.165804   -0.054938       0.648

# 以下函数用于在变量为分类变量且只有两个评价者时计算 Brennan-Prediger 系数
# 在以下函数中，要么用 x 和 y 给出两评价者的打分情况，要么用 m 给出汇总后的频数表
# 当用 x 和 y 输入时，unique_value 须设为分类变量所有可能的取值；当为默认的 NULL 时，会自动设为 unique(c(x, y))
manual_bp=function(x=NULL, y=NULL, m=NULL, unique_value=NULL){
    if (is.null(m)){
        uniquexy=if (is.null(unique_value)) unique(c(x, y)) else unique_value
        x=c(x, uniquexy)
        y=c(y, uniquexy)
        m=table(x, y)
        m=as(m, "matrix")
        diag(m)=diag(m)-1
    }
    if (!is.null(m)) m=if(is.table(m)) as(m, "matrix") else m
        n=sum(m)
        po=sum(diag(m))/n
        pe=1/nrow(m)
        y=(po-pe)/(1-pe)
        y
}

y=manual_bp(m=freqtable)
# [1] 0.8373494

# immer 包中的 immer_agree2 可计算数据为分类变量且有两名评价者时的 Aickin's Alpha 系数
# install.packages("immer")
library(immer)
```

```
# immer_agree2可通过 w 参数为每个评价项目设定权重
# 使用上文用过的的 cat_x 变量，该数据有664行；假如要为每个项目指定权重，需把一个长度为664的向量传给 w；此处我们随意生成一个权重，实际使用时不用设置权重，或根据需要设置权重
w=c(rep(1, 600), rep(2, 64))
y=immer_agree2(cat_x , w=w )
summary(y)
# Raw agreement = 0.901
# Scott's Pi = 0.835
# Cohen's Kappa = 0.835
# Aicken's Alpha = 0.848
# Gwet's AC1 = 0.859
# 结果为包括 Aickin's Alpha 在内的多个系数

# 用 irr 包里的 kendall 函数计算肯德尔系数，适用于变量为定序变量，有两个或以上评价者的情况。肯德尔系数不属于概率校正系数
# install.packages("irr")
library(irr)

# 设 correct=TRUE 的意义是对排序打结的情况进行调整；由于我们使用的是打分数据，肯定存在打结情况，所以务必设为 TRUE
y=kendall(dat[, 1: 5], correct=TRUE)
y$value
# [1] 0.5951645

# obs.agree 包中的 IBMD 函数计算基于熵的不一致性（Costa-Santos, 2010），适用于有多个评价者，变量为定比变量（最佳）、定距变量或定序变量的情况
# 注意，IBMD 求得的是不一致性；将不致性转为一致性的方法有多种，最简单的方法是用1减不一致性
# install.packages("obs.agree")
```

```
library(obs.agree)

set.seed(1) # 置信区间采用 Bootstrap 法计算，故需设随机种子
y=IBMD(dat[, 1: 5])
y$IBMD
#    value      ( 2.5 % – 97.5 % )
# 1  0.3946613   0.3220189    0.4480065

# Delta 包可用来计算 Dellta 系数（Andr é s and Marzo, 2004），该系数的表现优于 Kappa；适用于有两个评价者且变量为分类变量的情况
# Delta 系数不属于概率校正系数
# install.packages("Delta")
library(Delta)

m=matrix(c(80, 10, 10, 0), nrow=2) # 输入数据须为频数表
y=Delta(m)
y$Summary_AE
# 1 Kappa ± S.E. = –0.1111  ±  0.0247
# 2  Delta ± S.E. = 0.5769  ±  0.0801
# 在 m 的边缘分布明显不平衡的情况下，Delta 值为0.5769，明显高于 Kappa 值，并且更符合直观判断

m=matrix(c(10, 0, 5, 0, 20, 5, 0, 0, 30),nr=3) # 两个以上类别
y=Delta(m)
y$Summary
# 1  Chi-squared = 0 (d.f.= 1 ); p= 1
# 2 Kappa ± S.E. = 0.7705  ±  0.0666
# 3 Delta ± S.E. = 0.7383  ±  0.1156
```

接下来要考虑的问题是：该选择使用哪个指标？大量文献都已对此问题进行了探讨。例如，有研究者（例如，Gwet, 2008；Feng, 2012；Gwet, 2014: