Modern Applied Biostatistical Methods

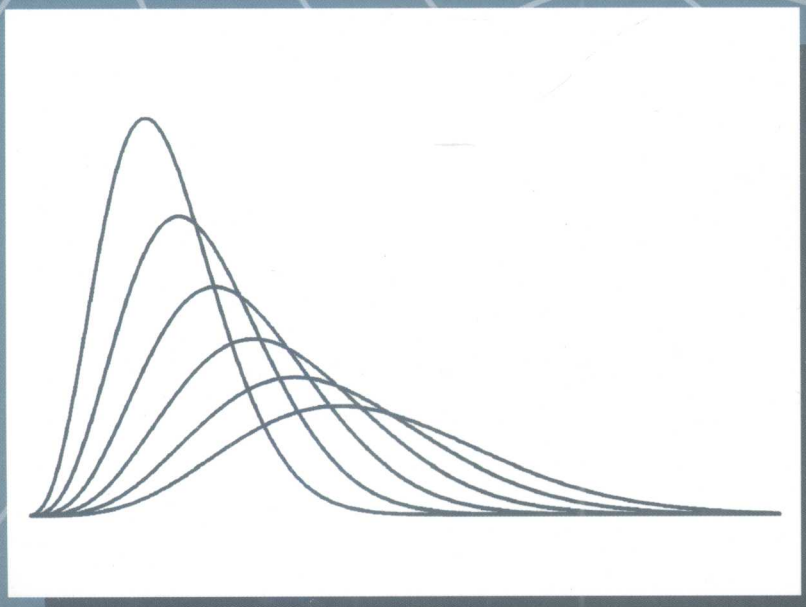
Using S-Plus

现代应用生物统计方法

S-Plus的使用

Steve Selvin 著

吕旌乔 译



北京大学医学出版社

现代应用生物统计方法

——S-Plus 的使用

Modern Applied Biostatistical Methods

Using S-Plus

Steve Selvin 著

吕旌乔 译

北京大学医学出版社

北京市版权局著作权合同登记号：图字：01-2008-0950

Modern Applied Biostatistical Methods：Using S-Plus
Steve Selvin

© 1998 by Oxford University Press

Simplified Chinese Translation Copyright © 2008 by Peking University Medical Press. All rights reserved.

Modern Applied Biostatistical Methods Using Splus was originally published in English in 1998. This translation is published by arrangement with Oxford University Press.

《现代应用生物统计方法——Splus 的使用》英文版于 1998 年首次出版。本书经牛津大学出版社许可出版。

图书在版编目（CIP）数据

现代应用生物统计方法——S-Plus 的使用/吕旌乔译．
—北京：北京大学医学出版社，2008.3
书名原文：Modern Applied Biostatistical Methods：Using S-Plus
ISBN 978-7-81071-992-6

Ⅰ．现… Ⅱ．吕… Ⅲ．计算机应用—生物统计 Ⅳ．Q-332

中国版本图书馆 CIP 数据核字（2008）第 016793 号

现代应用生物统计方法——S-Plus 的使用

译　　：吕旌乔
出版发行：北京大学医学出版社（电话：010-82802230）
地　　址：（100083）北京市海淀区学院路 38 号 北京大学医学部院内
网　　址：http://www.pumpress.com.cn
E - mail：booksale@bjmu.edu.cn
印　　刷：北京佳信达艺术印刷有限公司
经　　销：新华书店
责任编辑：邱　阳　　**责任校对：**杜　悦　　**责任印制：**郭桂兰
开　　本：787mm×1092mm　1/16　　**印张：**27　　**字数：**632 千字
版　　次：2008 年 4 月第 1 版　2008 年 4 月第 1 次印刷
书　　号：ISBN 978-7-81071-992-6
定　　价：89.50 元

版权所有　不得翻印　违者必究
本书如有质量问题请与教材供应部门联系

译序

由美国加州大学柏克莱分校 Steve Selvin 教授编写、北京大学吕旌乔博士翻译的《现代应用生物统计方法——S-Plus 的使用》是一本特色鲜明的书。Steve Selvin 是我的老朋友，一位非常友善和睿智的学者。据我所知，他是学校里一位非常受学生欢迎的教授，他开设的生物统计学基础课程在校内知名度很高，是选课人数最多、长期得到好评的一门精品课程。他的教学特点是把深奥难懂的统计课程讲得深入浅出，结合实例和统计分析工具，边讲解理论，边指导实践，学生能学懂，学了会用。《现代应用生物统计方法——S-Plus 的使用》一书原先是 Steve Selvin 教授的教案，经过多年改进完善最终成为一本专著。正如前言所述，该书的主要适用对象是从事流行病学和生物统计学的研究生，只要在高中数学知识的基础上略具微积分和矩阵代数知识，就可以满足学习的要求。对国内医学生而言，这是一个福音。任何一位对生物统计和 S-Plus 感兴趣的医学生，都可以通过这本书进入生物统计和 S-Plus 的世界，在实际操作和验证理论的过程中逐步体会什么是生物统计、统计软件如何使用，以及怎样将统计思想和统计软件融为一体。从事生物统计教学的专业人员，则可以通过该书了解美国生物统计教学的一个侧面，为改进国内生物统计和医学统计教学提供新的思路。

一本书不可能解决所有的问题，但一本书可以解决某些问题。该书是帮助读者了解生物统计和 S-Plus 的敲门砖，简单、实用、易学是其特色。如果你对生物统计和 S-Plus 有所兴趣，不妨拾起这本书慢慢阅读。再找一台计算机装上 S-Plus 软件，跟着 Steve Selvin 教授设计的思维轨迹，发现在精巧的 S-Plus 程序运行之后出现的“奇迹”，体会这些现象背后的统计学原理。当你感受到其中的乐趣，发出会心的微笑时，你的收获就是对作者和译者辛勤劳动最好的回报。你对这本书有兴趣吗？不妨试一下。

赵一鸣

2007 年 6 月 18 日于北京

前　言

常见的统计学书籍或偏重理论，或讨论统计应用，又或是某种计算机语言的使用手册。典型的手册型书籍重点在于说明分析性计算机语言的细节，往往忽视对统计推理过程的讲解，而统计理论或应用统计学著作则极少涉及统计方法如何实现的问题。本书将这些相互关联的元素整合成一体加以展现，通过统计型计算机语言 S-Plus[①] 和数据实例相结合的方式，阐述一些中初级生物统计方法。统计理论介绍（剔除复杂的数学部分）、详尽的应用方法和针对相应计算机/图形分析过程的细致描述，构成了本书各项主题的基本要素，从而将理论、数据、实现过程、统计图形和结果解读结合在一起，展现了生物统计分析的完整过程。

本书各章节的开始部分，都会简要地叙述特定统计方法的内在推理过程。紧接着将针对相应数据集进行统计分析，通过 S-Plus 命令、S 函数和计算机指令生成数值结果以及相对应的图形输出——图形是数据分析过程中容易被忽视，但却非常关键的部分。在最后会给出统计推断及结论，完成对统计方法的描述。

书中涉及的方法和例子，都来自作者向生物统计学专业、高级流行病学专业一年级研究生讲授的一门 1 学期课程。书中涉及的数学知识为初等水平，需要一些大学一年级的微积分和矩阵代数知识（可以忽略，仅对阅读连续性有轻微的影响）。读者最好具备在 UNIX 或 Windows 环境下使用 S-Plus 的条件。尽管这一点并非严格必需，但在计算机上验证示例分析或尝试其他方法（所有例子的数据都是完整的）显然非常重要。书中各章后的习题都应该用 S 语言工具完成。

本书在正文中给出了大量的 S 语言示例，例中详细阐明了分析过程的原理，并通过具体的数学项式对原理加以表述。抽象符号和数学推导是统计分析过程中在基本概念和问题之间建立联系的基本方式。书中给出了大量有关 S 语言的注解，目的是为了让读者更容易理解统计计算的细节，以便从另一个角度来观察统计学如何识别被收集数据的特征以及为什么要如此识别。当然，学习统计分析的计算机实现过程有一个额外收获，就是读者将掌握 S 这一现代统计分析系统的语言。本书中关于 S 语言的示例特意写得尽量简单，代价是放弃了程序的通用性。要执行同样或类似的任务，基本上 S 系统都有某种更通用的、“用户友好”的替代选择。

有人会提出一个很好的问题：为什么要使用 S-Plus？S-Plus 并不是一个典型的统计分析系统，事实上，它是一种便于进行统计分析的计算机语言。S-Plus 包含了许多非常有用的组件，使用 S-Plus 最明显的优势，就在于利用这些组件，可以在很大的范围

---

① S是一种计算机语言。S-Plus 是一个商业化的 S 语言平台，由 Insightful 公司开发，目前已经发展到了 8.0 版。请注意，R 是另一种符合 S 语言规范的计算机语言。R 是自由软件，安装包可通过 Internet 下载。除了极少数的例外，本书中的 S 程序都可以在 R 语言中正常运行。（译者注）

试读结束，需要全本PDF请购买 www.ertongbook.com

内自由地创建数据分析方案，而不会受限于典型“批处理”型统计软件中的预定格式。S-Plus 的组件可以连接在一起，精确地建立理想的分析策略。而且，S 程序在交互式环境中编写，得到的结果持续可用，并且允许用户根据中间结果对 S 程序进行修改和订正。一种在交互式环境中特别有用的工具，就是持续地创建图形显示，S 图形的生成既快又容易。S 语言的另一个优点是能够生成具有特定统计特征的随机数值，这些随机数值在模拟和估计过程中非常重要。为统计分析量身定做的灵活的计算机语言、交互式的应用环境、创建图形显示和生成随机数值的能力，这一切结合在一起，将 S-Plus 系统打造成了数据分析人员手中的有力工具。

本书各章主题如下：

第 1 章介绍一批经过精选、对于统计应用特别有用的 S 命令。这些命令只是所有可用命令中的一小部分，但却组成了一个“最小”集，构成后文中统计应用的基础。这些 S 工具将通过大量例子进行阐述。正如大多数语言的学习过程，模仿已有的方式，是达到整体理解的开始。

第 2 章包括两类基本要素：描述性统计量和 S 图形。本章的目的之一是讨论数据描述技术，例如直方图、茎叶图、箱图、QQ 图、数据平滑策略和聚类分析。统计描述方法的讲解过程，在相当大的程度上从各种图形显示方法中得到了加强。展示生成这些图形的 S 工具，是本章的另一个主要目的。例如，进行密度函数估计、形成估计密度函数图形所必需的 S 语句等等，都将成为讨论主题。

第 3 章致力于随机变量的生成、检验和应用。由计算机生成、具有特定统计学特征的模拟数据，在探讨分析技术特性的过程中是一个重要角色。对于统计估计和统计检验，随机数值也发挥了非常重要的作用。在现代生物统计分析中，统计推理的知识和生成随机数值的方法同样必不可少。

线性模型的应用对于鉴别、认识多元测量之间关系而言是一种非常实用的手段。第 4 章包括了 3 种常用分析模型——线性模型、logistic 模型和泊松模型的理论、实现和结果解读。每种模型都被应用于研究数据集、生成数值结果，然后通过统计图形分析加以补充。同样，叙述中包含了统计方法原理和计算机 S 语言程序，对应用线性模型探索多元数据内部关系的过程进行综合描述。

第 5 章介绍 3 种统计估计方法——最大似然，靴袢和最小二乘及非线性最小二乘估计。同样，文中展示了原理、S 程序、图形分析和一系列应用过程。同前面的章节一样，对这 3 种基本估计方法的研究着重于数据汇总统计量的估计，并对完整的统计过程进行深入的展示。

以表格显示的离散数据，无疑在许多数据集中处于中心地位。第 6 章介绍性地讨论了多元离散数据分析。对这一宽泛主题的讨论从 2×2 表格开始，继之以 r×c 列联表，最后介绍对数线性模型在多元离散列表数据分析中的应用。同样，通过 S 程序和具体的数据示例对有关概念和实现过程加以阐述。

第 7 章汇集了一些颇有价值的统计学方法（方差分析、模型选择方法、主成分分析和典则相关分析）。对于这四个主题，重点在于应用和实现过程。

第 8 章开始于率和寿命表，结束于 Cox 比例风险模型。展示了研究几个来自生存

数据的基本问题的理论、图形和分析手段。例如，利用S语言工具分析一项白血病临床试验的随访数据，用S图形展示数据，并从不同角度对结果加以解释。

Steve Selvin
加州大学柏克莱分校，1997

目　录

试读结束，需要全本PDF请购买 www.ertongbook.com

1 S语言

起步

要启动S语言，只需敲入 *Splus*，按下 Enter 键[①]。计算机将做出以下回应：

```
S-Plus: Copyright (c) 1988, 1993 Statistical Sciences.
S: Copyright AT&T.
Version 3.2 Release 1 for Sun Sparc, SunOS 4.x : 1993
Working data will be in /h/name/.Data
>
```

视操作系统不同，计算机回应的细节会有所差别。差异同时也取决于 S-Plus 的版本，以及其他一些琐碎的一般因素。最重要的角色是符号“>”，它提示 S-程序已经就位，准备对用户的指令做出反应。在计算机终端 “>”之后键入的任何字符，都会被 S-Plus 程序当作一条命令，只要按下 Enter 键，S-Plus 就会尝试去执行。

q( )是一条仅次于 *Splus* 的重要命令，可以结束 S 进程并返回到计算机操作系统界面。在 S 语言中，括号是用于识别输入参量的约定用语，在后文中可以看出这一点。如果只键入 *q* 和“Enter”键（*q* 后没有括号），S-Plus 将在屏幕显示命令 *q* 所调用的计算机程序，但并不执行命令。这是所有 S 函数的共同特征。

随之而来的一个问题是：有效的命令是什么样的？回答这一问题并不容易，可能的答案有数百种，本书将描述其中的一部分。作为起步，表 1.1 列出了多种可能对统计分析有用的符号和函数。要了解这些命令做什么以及如何使用这些命令，S 语言提供了一些“在线”描述。要调出相应说明，可键入 *help*( )并在括号中填入要查询的特定命令，比如表 1.1 中列出的各种函数。例如，命令 *help*(*q*)将在终端上显示：

```
Quit From S-PLUS
DESCRIPTION:
      Terminates the current S-PLUS session
USAGE:
     q(n=0)
```

---

① 此处为 unix 下的启动方法，在 windows 环境下通过“开始程序”菜单或快捷方式启动程序。（译者注）

表 1.1　部分 S 函数和符号列表（详细说明见 S-帮助系统）

!	all	df	kappa	phyper	replace
!=	anova	dgamma	kronecker	pi	resid
$	any	dgeom	kruskal.test	pie	residuals
%%	aperm	dget	labels	plnorm	rexp
%*%	apply	dhyper	lag	plogis	rf
%c%	approx	diag	lapply	plot	rgamma
%m%	array	diff	leaps	pnbinom	rgeom
%o%	arrows	dim	legend	pnorm	rhyper
&	asin	dimnames	length	pnrange	rlnorm
&&	asinh	discr	lgamma	points	rlogis
*	assign	dist	lines	poisson	rm
**	atan	dlnorm	list	poly	rnbinom
+	atanh	dlogis	lm	postscript	rnorm
-	attach	dnbinom	lo	ppois	round
->	attr	dnorm	loess	prcomp	row
:	audit.file	dotchart	log	pretty	rpois
<	axes	double	log10	prod	rt
<-	axis	dpois	logical	prompt	runif
<<-	backsolve	dput	loglin	pscript	rweibull
<=	barplot	drop	lowess	pt	rwilcox
==	binomial	dt	lpr	punif	sample
>	boxplot	dump	ls	pweibull	sapply
>=	break	dunif	lsfit	pwilcox	save
?	c	duplicated	mahalanobis	q	scale
AUDIT	cancor	dweibull	mantelhaen.test	qbeta	scan
BATCH	cat	dwilcox	match	qbinom	search
Beta	category	eigen	matpoints	qcauchy	segments
Binomial	cbind	else	matrix	qchisq	seq
Cauchy	ceiling	encode	max	qexp	show
Chisquare	chisq.test	end	mcnemar.test	qf	sign
Exponential	chol	exp	mean	qgamma	signif
F	coef	faces	median	qgeom	sin
For	col	factor	min	qhyper	single
GAMMA	compare	family	month	qlnorm	sinh
Gamma	contour	find	motif	qlogis	sink
Geometric	coplot	fix	na	qnbinom	smatrix
Hypergeometric	cor	floor	na.omit	qnorm	smooth
Logistic	cor.test	for	ncol	qpois	solve
Lognormal	cos	format	next	qqline	sort
NegBinomial	cosh	formula	nrow	qqnorm	source
Normal	coxreg	frequency	ns	quantile	spin
T	crossprod	friedman.test	null	qunif	spline
TRUNC_AUDIT	crt	function	order	qweibull	sqrt
Uniform	cstr	gaussian	outer	qwilcox	stars
Weibull	cumprod	get	pairs	random	stepwise
Wilcoxon	cumsum	glm	par	randomize	sum
X11	cut	hclust	paste	Range	summary
abline	date	help	pattern	rank	sweep
abs	dbeta	hist	pbeta	rbeta	t
ace	dbinom	hist2d	pbinom	rbind	t.test
acf	dcauchy	history	pcauchy	rbinom	table
acos	dchisq	identify	pchisq	rcauchy	tabulate
acosh	density	if	persp	rchisq	tan
again	derive	ifelse	pf	remove	tanh
aggregate	deviance	integer	pgamma	rep	tapply
alias	dexp	integrate	pgeom	repeat	terms

```
OPTIONAL ARGUMENTS:

      n: integer value, to be used, modulo 256, as the
         exit status of the S-PLUS process. (In Shell
         scripts, a non-zero exit status is conventionally
         an indication of an error.)

SIDE EFFECTS:
         Causes termination of the S-PLUS session and
         return to the operating system. If one or
         more graphics devices are active, a
         device-dependent wrap-up routine will be
         executed for each active device. The
         function or expression .Last, if it exists,
         will be called or evaluated before quitting.
                              ⋮
```

调用 *help* 命令的快捷方式，是键入由“问号”引导的函数名。例如，*?q* 和 *help*(*q*)的效果完全一样。此外，S-Plus 还有一个详尽的在线帮助系统，可以通过菜单使用。S 命令 *help.start*(*gui*=“*motif*”)可以显示一个菜单，允许用户先在众多类别中进行选择，进而挑选各个类别下的特定 S 函数。此处“*gui*”的含义为图形用户界面。若需打印 S 帮助“页面”，可使用命令 *help*(*function.name*, *offline*=*T*)，则所要求的“帮助页面”将直接发往打印机。本书始终都鼓励读者使用 S-Plus 的“在线”帮助系统，这是一个非常有用的工具。此外，目前已经出版了数本详尽的手册，从用户和程序员的角度全面地描述了 S 语言。

退出命令可以停止 S-Plus 程序。不过，如果在某些情况下要中断执行中的 S 命令，可以使用 Ctrl+c（同时按下“Ctrl”键和字母“c”键）停止当前正在执行的命令，系统将回到 S 命令行状态，但并不退回操作系统。如果同时按下“Ctrl”键和“\”键，则会中止运行中的 S-Plus 程序并退回操作系统。

3 种数据类型及一些输入规则

S 语言中有 3 种主要的数据类型：单一值、向量和数组。3 种数据都可以用单独的符号加以命名，S-Plus 会自行追踪数据的类型。某个单一值，如数值 10，可以用一个字符如 *x* 标识，即

```
> x<- 10
```

由“<”和“-”组成的“箭头”表示数值 10 被赋给了符号 *x*。这一语句还可以写成：

```
> x_10
```

即使用一个“下划线”[①]将 x 的值指定为 10。两种方式的结果完全一样，本书今后将采用“箭头”语法。“箭头”方式有一个小小的好处，即赋值可以是双向的：

```
> 10 ->x
```

是合法的语句，而

```
> 10_x
```

则不然。如果仅仅键入 S 变量名，随即按下“Enter”键，计算机屏幕上将显示该变量的值。例如：

```
> x
[1] 10
```

S 标号“[1]”的意义，将随着更广泛的变量被讨论而更加清晰。通过运算操作，可以从 S 变量产生新的数值。例如：

```
> x+100
[1] 110
```

或者：

```
> x^2
[1] 100
```

其中 x^2 表示数值 x 的平方（x^2）。

几乎所有字母、数字和字符的组合，都可以用来命名 S 变量。极少数的组合会和 S 语言中已经存在的函数名冲突，如 *for*，*if*，*c* 和 *t*（参见表 1.1）。当一个变量的名称与已有 S 保留命名相同时，会出现一个警告或错误提示。某些情况下用户指定的变量仍然可以被定义，但不管怎样，最好完全避免命名冲突，这一点很容易做到。顺便提一下，S-Plus 中的变量名对大小写敏感，也就是说，命名时大、小写字母会造成差别（如 x 不同于 X，fx 不同于 Fx）。

数值向量同样可以用某一单独字符如 x 标识。要将 x 定义为一个包含了数值 10，20，30 和 40 的向量，可使用 S 函数 *c*( )，“*c*”代表了 *combine*——连接。*c*( )将括号中一系列由逗号分隔的数值，连接成一个单一的变量表达式，即：

```
> x<- c(10,20,30,40)
> x
[1] 10 20 30 40
```

与单值变量类似，对向量型变量也可以进行数学运算以形成新的变量，例如：

```
> x+100
[1] 110 120 130 140
> x^2
[1] 100 400 900 1600
```

请注意，此时运算符（+和^）被应用于向量内的各个元素。*c*( )可以执行多种多样的连接操作。例如：

[①] 在 R 语言中，命令 x_10 不成立。（译者注）

```
> ctemp<- c(1,2,3,4,5,6,7,8,9)
> c(ctemp,ctemp,ctemp)
[1] 1 2 3 4 5 6 7 8 9 1 2 3 4 5 6 7 8 9 1 2 3 4 5 6 7 8 9
> c(0,ctemp,10)
[1] 0 1 2 3 4 5 6 7 8 9 10
```

与单值型、向量型变量类似，对数值数组也可以赋予一个变量名。对于两个向量，以下命令：

```
> c1<- c(10,20,30,40)
> c2<- c(5,10,15,20)
```

和 *cbind*( )命令（“*cbind*” 表示多列绑定）联用，可以形成一个二维数值数组，即：

```
> x<- cbind(c1,c2)
> x
     c1  c2
[1,]  10   5
[2,]  20  10
[3,]  30  15
[4,]  40  20
```

命令 *cbind*( )将多个指定向量（作为参数写在括号内并以逗号分隔）联合在一起，各个向量成为数组的列。与前面两种情况一样，该数组也可以通过一个单独字符 *x* 进行运算，例如：

```
> x+100
     c1   c2
[1,] 110  105
[2,] 120  110
[3,] 130  115
[4,] 140  120
```

以及：

```
> x^2
      c1   c2
[1,]  100   25
[2,]  400  100
[3,]  900  225
[4,] 1600  400
```

还有一个与 *cbind*( )并列的命令 *rbind*( )，“*rbind*” 代表多行合并，可以将多个向量作为数组的行组合在一起，例如：

```
> rbind(c1,c2)
    [,1]  [,2]  [,3]  [,4]
c1    10    20    30    40
c2     5    10    15    20
```

一个上升或下降的整数数列，可通过一条简练的 S 语句生成，即起始值：结束值，例如：

```
> 0:10
[1] 0 1 2 3 4 5 6 7 8 9 10
> temp<- 1:9
> temp
[1] 1 2 3 4 5 6 7 8 9
> 20:8
[1] 20 19 18 17 16 15 14 13 12 11 10 9 8
> c(1:5,5:1,1:5)
[1] 1 2 3 4 5 5 4 3 2 1 1 2 3 4 5
> rbind(1:6,11:16,111:116)
      [,1]  [,2]  [,3]  [,4]  [,5]  [,6]
[1,]     1     2     3     4     5     6
[2,]    11    12    13    14    15    16
[3,]   111   112   113   114   115   116
```

二维的数值数组也可以通过 S 函数 *matrix*( )创建，其一般形式为：

matrix (数值或数值向量,行的数目,列的数目)

例如，一个名为 *darray* 的数组可由以下语句创建：

```
>darray<- matrix(1:12,3,4)
>darray
      [,1]  [,2]  [,3]  [,4]
[1,]     1     4     7    10
[2,]     2     5     8    11
[3,]     3     6     9    12
```

某些情况下最后一个参数可以省略，例如：

```
>array0<- matrix(1:12,6)
>array0
      [,1]  [,2]
[1,]     1     7
[2,]     2     8
[3,]     3     9
[4,]     4    10
[5,]     5    11
[6,]     6    12
```

产生一个 6 行，且列数必然为 2 的数组。

数组内的特定数值可以用数组名+“方括号”取出。例如，命令 *darray*[*i*,*j*]将从数组 *darray* 中抽出一个单独的数值，其中 *i* 代表数值所处行，*j* 代表所处列。例如：

试读结束，需要全本PDF请购买 www.ertongbook.com