

# SAS语言教程 及其应用

张应山 张晓琴 李顺勇 著

山西出版集团  
山西人民出版社

# SAS语言教程及其应用

张应山 张晓琴 李顺勇 著

## 图书在版编目 (C I P) 数据

SAS 语言教程及其应用 / 张应山著. — 太原: 山西人民出版社, 2011.8

ISBN 978-7-203-07394-9

I. ①S… II. ①张… III. ①统计分析—应用软件, SAS—教材 IV. ①C812

中国版本图书馆 CIP 数据核字 (2011) 第 158769 号

### SAS 语言教程及其应用

---

著 者: 张应山 张晓琴 李顺勇

责任编辑: 贾 娟

装帧设计: 王聚金

---

出 版 者: 山西出版集团·山西人民出版社

地 址: 太原市建设南路 21 号

邮 编: 030012

发行营销: 0351-4922220 4955996 4956039

0351-4922127 (传真) 4956038 (邮购)

E - mail : sxskcb@163.com 发行部

sxskcb@126.com 总编室

网 址: www.sxskcb.com

---

经 销 者: 山西出版集团·山西人民出版社

承 印 者: 山西嘉祥印刷包装有限公司

---

开 本: 889mm × 1194mm 1/16

印 张: 20.25

字 数: 500 千字

印 数: 1-500 册

版 次: 2011 年 8 月 第 1 版

印 次: 2011 年 8 月 第 1 次印刷

书 号: ISBN 978-7-203-07394-9

定 价: 60.00 元

---

如有印装质量问题请与本社联系调换

---

# 目录

<b>第一章 SAS编程的基础语法概念及自动化建模</b>	<b>1</b>
§ 1.1 自动化建模基本概念和程序 .....	1
§ 1.2 基于SAS语言的自动化建模程序 .....	2
§ 1.3 程序的基本含义 .....	4
§ 1.3.1 数据步 .....	4
§ 1.3.2 过程步 .....	6
§ 1.3.3 全程和宏步 .....	7
§ 1.4 输出结果及其含义 .....	8
§ 1.4.1 Output窗口的输出结果及其含义 .....	8
§ 1.4.2 Graph窗口的图形输出结果及其含义 .....	35
§ 1.5 自动化建模思想的推广应用 .....	36
§ 1.5.1 自动化建模的模型 .....	36
§ 1.5.2 模型的分析 .....	36
§ 1.6 SAS编程的基础语法概念 .....	40
§ 1.6.1 数据集 .....	40
§ 1.6.2 变量 .....	40
§ 1.6.3 常量 .....	42
§ 1.6.4 SAS函数 .....	46
§ 1.6.5 SAS操作符 .....	50
§ 1.6.6 SAS表达式 .....	51
§ 1.6.7 SAS语言的一般结构 .....	52
<b>第二章 DATA步基本语句和模拟与仿真程序</b>	<b>54</b>
§ 2.1 模拟与仿真的基本概念 .....	54
§ 2.2 模拟与仿真程序 .....	57
§ 2.2.1 模拟数据程序 .....	57
§ 2.2.2 模拟分布程序 .....	61
§ 2.2.3 仿真程序 .....	62
§ 2.3 程序的基本含义 .....	66

§ 2.3.1 数据步 .....	66
§ 2.3.2 过程步 .....	67
§ 2.3.3 全程和宏步 .....	69
§ 2.4 输出结果及其含义 .....	70
§ 2.4.1 Output窗口的输出结果及其含义 .....	71
§ 2.4.2 Graph窗口的图形输出结果及其含义 .....	75
§ 2.5 DATA步基本语句的语法 .....	77
§ 2.5.1 DATA语句 .....	77
§ 2.5.2 INPUT语句和PUT语句 .....	79
§ 2.5.3 FILE语句和INFILE语句 .....	92
§ 2.5.4 FORMAT语句和INFORMAT语句 .....	96
§ 2.5.5 CARDS   CARDS4   DATALINES   DATALINES4语句 .....	99
§ 2.5.6 信息语句 .....	100
§ 2.5.7 ATTRIB 语句(属性语句) .....	106
§ 2.5.8 观测值的删除和增加 .....	107
§ 2.5.9 赋值语句 .....	117
§ 2.5.10 累加求和语句(SUM语句) .....	118
§ 2.5.11 条件语句 .....	119
§ 2.5.12 SAS语句的转折 .....	131
§ 2.5.13 RETURN语句(返回语句) .....	134
§ 2.5.14 LINK语句(连接语句) .....	136
§ 2.5.15 循环语句 .....	137
§ 2.5.16 ARRAY语句(数组语句) .....	142
§ 2.5.17 对多个数据集的操作语句 .....	148
§ 2.5.18 其它可执行语句 .....	158
<b>第三章 过程步基本语句和自动化控制</b> .....	<b>164</b>
§ 3.1 工业自动化控制问题的提出和算法思想 .....	164
§ 3.2 基于SAS语言的求稳定中心的程序 .....	167
§ 3.3 程序的基本含义 .....	171
§ 3.3.1 数据步 .....	172
§ 3.3.2 过程步 .....	175
§ 3.3.3 全程和宏步 .....	176

§ 3.4 输出结果及其含义 .....	176
§ 3.4.1 Output窗口的输出结果及其含义 .....	176
§ 3.4.2 Graph窗口的输出图形及其含义 .....	178
§ 3.5 PROC步基本语句的语法 .....	179
§ 3.5.1 PROC 语句 .....	179
§ 3.5.2 VAR 语句 .....	180
§ 3.5.3 BY 语句 .....	181
§ 3.5.4 CLASS 语句 .....	182
§ 3.5.5 MODEL 语句、PLOT 语句、TABLES 语句和TABLE 语句 .....	184
§ 3.5.6 WEIGHT 语句 .....	186
§ 3.5.7 FREQ 语句 .....	190
§ 3.5.8 ID 语句 .....	191
§ 3.5.9 WHERE 语句 .....	192
§ 3.5.10 OUTPUT 语句 .....	193
§ 3.5.11 LABEL 语句 .....	193
§ 3.5.12 FORMAT 语句 .....	195
§ 3.5.13 ATTRIB 语句 .....	195
§ 3.5.14 制图基本语句 .....	196
§ 3.5.15 SAS/IML模块语言 .....	201
§ 3.5.16 RUN语句和QUIT语句(过程退出语句) .....	215
<b>第四章 全程和宏步基本语句和自动化预测</b> .....	<b>217</b>
§ 4.1 自动化预测的基本概念和程序 .....	217
§ 4.2 基于SAS语言的自动化预测程序 .....	221
§ 4.3 程序的基本含义 .....	228
§ 4.3.1 数据步 .....	228
§ 4.3.2 过程步 .....	228
§ 4.3.3 全程和宏步 .....	229
§ 4.4 输出结果及其含义 .....	231
§ 4.5 全程和宏步基本语句的语法 .....	233
§ 4.5.1 注释语句 .....	233
§ 4.5.2 TITLE 语句 .....	234
§ 4.5.3 FOOTNOTE 语句 .....	236

§ 4.5.4 MISSING语句 .....	237
§ 4.5.5 LIBNAME 语句 .....	238
§ 4.5.6 FILENAME 语句 .....	239
§ 4.5.7 OPTIONS语句 .....	242
§ 4.5.8 %INCLUDE(程序语句的调用)语句和INCLUDE语句 .....	243
§ 4.5.9 %LET(宏变量赋值)-&name语句 .....	244
§ 4.5.10 %MACRO-%MEND(宏函数)语句 .....	248
§ 4.5.11 %IF-%THEN(宏条件)语句 .....	249
§ 4.5.12 %GOTO-%label(宏转折)语句 .....	250
§ 4.5.13 %DO-%END(宏循环)语句 .....	251
§ 4.5.14 %WINDOW-%DISPLAY (宏窗口)语句.....	253
§ 4.5.15 其它全程和宏步的语句 .....	255
<b>第五章 基本SAS过程和自动化工程计算</b> .....	<b>260</b>
§ 5.1 二分之一逼近解方程 .....	260
§ 5.2 三分之二逼近求极小值 .....	261
§ 5.3 求积分 .....	262
§ 5.4 求偏导数 .....	263
§ 5.5 PRINT 过程.....	266
§ 5.5.1 过程功能与格式 .....	266
§ 5.5.2 过程说明 .....	267
§ 5.5.3 应用举例 .....	268
§ 5.6 SORT 过程.....	269
§ 5.6.1 过程说明 .....	270
§ 5.6.2 应用举例 .....	270
§ 5.7 FORMAT 过程 .....	271
§ 5.7.1 过程说明 .....	272
§ 5.7.2 应用举例 .....	273
§ 5.8 TRANSPOSE 过程 .....	274
§ 5.8.1 过程功能与格式 .....	274
§ 5.8.2 过程说明 .....	275
§ 5.8.3 应用举例 .....	275
§ 5.9 CONTENTS 过程 .....	276

§ 5.9.1 过程功能与格式 .....	276
§ 5.9.2 过程说明 .....	277
§ 5.9.3 应用举例 .....	277
§ 5.10 TABULATE 过程 .....	279
§ 5.10.1 过程功能与格式 .....	279
§ 5.10.2 过程说明 .....	279
§ 5.10.3 应用举例 .....	281
§ 5.11 FREQ 过程 .....	282
§ 5.11.1 过程功能与格式 .....	282
§ 5.11.2 过程说明 .....	283
§ 5.11.3 应用举例 .....	284
§ 5.12 GPLOT 过程 .....	287
§ 5.12.1 过程功能与格式 .....	287
§ 5.12.2 过程说明 .....	287
§ 5.12.3 应用举例 .....	288
§ 5.13 MEANS 过程 .....	289
§ 5.13.1 过程功能与格式 .....	289
§ 5.13.2 过程说明 .....	289
§ 5.13.3 应用举例 .....	290
§ 5.14 UNIVARIATE 过程 .....	292
§ 5.14.1 过程功能与格式 .....	292
§ 5.14.2 过程说明 .....	293
§ 5.14.3 应用举例 .....	295
§ 5.15 TTEST 过程 .....	295
§ 5.15.1 过程功能与格式 .....	295
§ 5.15.2 过程说明 .....	296
§ 5.15.3 应用举例 .....	296
§ 5.16 REG 过程 .....	297
§ 5.16.1 过程功能与格式 .....	297
§ 5.16.2 过程说明 .....	298
§ 5.16.3 应用举例 .....	298
§ 5.17 ANOVA 过程 .....	299

§ 5.17.1 过程功能与格式.....	299
§ 5.17.2 过程说明.....	299
§ 5.17.3 应用举例.....	300
§ 5.18 FACTOR 过程.....	301
§ 5.18.1 过程功能与格式.....	301
§ 5.18.2 过程说明.....	302
§ 5.18.3 应用举例.....	302
§ 5.19 FORECAST 过程.....	304
§ 5.19.1 过程功能与格式.....	304
§ 5.19.2 过程说明.....	304
§ 5.19.3 应用举例.....	305
§ 5.20 判别归类过程.....	307
§ 5.20.1 过程功能与格式.....	307
§ 5.20.2 过程说明.....	308
§ 5.20.3 应用举例.....	309
§ 5.21 聚类过程.....	310
§ 5.21.1 过程功能与格式.....	310
§ 5.21.2 过程说明.....	311
§ 5.21.3 应用举例.....	312

# 第一章 SAS编程的基础语法概念及自动化建模

本章给出一种基于SAS语言的计算机自动化建模程序, 结合程序介绍SAS语言的基本结构和含义, 并进行了推广应用. 结合这些应用程序, 介绍一些SAS语言的整体基本概念.

## § 1.1 自动化建模基本概念和程序

建模问题在统计学中是一个重要的概念, 所谓建模是根据自变量和因变量的试验数据, 建立一个数学分析模型, 用于以后的统计分析, 建模的好坏的一个标准是对已知的数据要拟合的比较好. 对模型类型的选择, 通常是根据专家的经验选择的.

但是统计学家不可能是各个方面的专家, 对模型的某些假设可能是不合适的. 统计学家研究问题, 可以不从假设开始, 而可以从数据开始. 让计算机按一定的程序根据已知数据自己建立分析模型的方法, 称作**自动化建模**.

本章介绍一个计算机自动建模的思想方法. 这个方法来源于中国古代的哲学思想: 八卦的算法. 中国古代的八卦算法要求对研究的问题, 尽可能的收集或者变换出各个方面的信息, 然后把各种信息分成相互正交类, 在同等条件下(不考虑显著因素)综合平衡各个类的信息, 得到分析结果. 这类似于多元统计分析理论中的因子分析. 此因子分析对应SAS的factor过程, 可参见[3].

因此, 在建模时, 根据数据的已知的自变量, 用函数变换的方法, 把自变量变得尽可能的多, 然后用因子分析的方法把所有自变量首先中心标准化(减去均值除以标准差, 此时自变量的变化范围基本在 $[-1, 1]$ 之间)后, 根据方差阵提出重要的正交因子(对应的方差阵的特征值之和占总特征值之和的比例比较大, 比如大于95%), 并再进行正交旋转, 把各个重要的正交因子的贡献率(或者特征值变为1)变得相等, 并使得各个重要的正交因子的变化范围也基本在 $[-1, 1]$ 之间, 然后用逐步回归分析建模. 此逐步回归分析建模对应SAS的reg过程的method=stepwise选项, 可参见[3].

在用逐步回归分析建模时, 由于各个重要的正交因子的变化范围也基本在 $[-1, 1]$ 之间, 相互之间正交, 因此用这些重要的正交因子进行二次多项式回归, 应可以达到应有的精确度. 这种方法甚至可以使得对已知的数据的拟合优度接近于0.

上述方法中, 要估计的因变量(实验数据)的形式 $fun$ , 重要正交因子的个数 $n$ 和自变量的个数 $m$ , 都是在建模以前需要知道的量, 这些量的不同, 将对应不同的模型. 但在这些量确定以后, 所用的建模程序是一样的. 需要建立一个可以对任何上述量, 都可以进行的自动化建模程序. 在所有计算机语言中, SAS语言是解决这类问题最好的语言. 如下是用SAS语言根据上述思想建立的分析程序的模板. 对任何数据, 用类似下述程序可以实现自动化建模.

在应用中, 通过调整要估计的因变量的形式 $fun$ , 重要正交因子的个数 $n$ 和自变量的个数 $m$ , 可以获得所需要的拟合优度.

## § 1.2 基于SAS语言的自动化建模程序

```
%let fun=y2/y1;
%let n=7;
%let m=8;
data a0;
input x1-x7 y1 y2;
cards;
30 5 30 0 0 8 45 42.3 0.9
30 10 40 10 0.1 8.5 55 8.27 1.98
30 20 50 20 0.2 9 62 213 4.25
40 5 30 10 0.1 9 62 50.3 2.25
40 10 40 20 0.2 8 45 64.6 2.80
40 20 50 0 0 8.5 55 95.7 3.93
50 5 40 0 0.2 8.5 62 29.7 2.40
50 10 50 10 0 9 45 48.9 2.30
50 20 30 20 0.1 8 55 47.1 6.90
30 5 50 20 0.1 8.5 45 24.0 1.81
30 10 30 0 0.2 9 55 38.5 2.21
30 20 40 10 0 8 62 101 3.80
40 5 40 20 0 9 55 52.7 0.44
40 10 50 0 0.1 8 62 23.6 4.01
40 20 30 10 0.2 8.5 45 32.6 5.75
50 5 50 10 0.2 8 55 33.4 2.45
50 10 30 20 0 8.5 62 92.1 3.61
50 20 40 0 0.1 9 45 17.4 4.81
43 21 36 0.05 0.1 9.1 43 a b
;
data aa;
set a0;
x8=log(x7);
```

```
x9=sin(x7);  
proc factor data=aa nfactors=&n out=bb;  
var x1-x&m;  
data a;  
set bb;  
array mfactor{&n} factor1-factor&n;  
array mf&n f1-f&n;  
do i=1 to &n;  
mf{i}=mfactor{i};  
end;  
proc reg ;  
model f1-f&n=x1-x&m;  
output out=cc p=ff1-ff&n;  
proc print;  
%macro ff;  
%do ii=1 %to &n;  
%do jj=1 %to &n;  
f&ii.f&jj  
%end;  
%end;  
%mend;  
data b;  
set a;  
array mf{&n} f1-f&n;  
array mff{&n,&n} %ff;  
do i=1 to &n;  
do j=1 to &n;  
mff{i,j}=mf{i}*mf{j};  
end;  
end;
```

```
t=_n_;
z=&fun ;
proc reg data=b;
model z=f1-f&n %ff / method=stepwise;
output out=bb p=zz;
proc gplot;
symbol1 i=spline v=star c=green;
symbol2 i=spline v=circle c=magenta;
plot (z zz)*t/overlay;
title 'z=' &fun;
data www;
merge bb a0;
proc print;
var x1-x&m z zz;
run;
```

### § 1.3 程序的基本含义

SAS软件一般分三个窗口: " Program Editor ", " Log " 和 " Output " 窗口. " Program Editor " 窗口用于编程, " Log " 窗口用于检验程序, " Output " 窗口用于输出文本数据形式的计算结果.

SAS语言结构一般分成三部分. 第一部分是对数据的基本处理, 称为**数据步**, 重点是控制对数据的输入和输出, 建立数据集; 第二部分是对SAS过程的调用和过程内部的运算处理, 称为**过程步**, 重点是对数据集进行特殊的运算并输出运算结果, 输出运算结果经常也是数据集; 第三部分是对全程变量和宏变量的编程和处理, 称为**全程和宏步**, 作用是对数据步和过程步综合进行处理和嵌套. 目的是在不知某些 " 宏变量 " 和 " 宏参数 " 时, 实现计算机自动化编程.

SAS程序语言的每一个语句中间的几个语词之间用空格间隔, 每一个语句后面用 ";" 结束, 最后用 " run; " 执行所有程序.

数据步和过程步的变量的默认值为数值型变量, 而宏步的变量的默认值为字符型变量.

#### § 1.3.1 数据步

数据步的主要作用之一是建立数据集. 此数据步以SAS语句 " data " 表示数据步开始.

## 1. data a;

建立一个以a命名的虚拟数据集. 这个数据集在计算机关机以后自动消失.

如果想建立在计算机硬盘 " D: " 的路径为 " D : \mydata " 下保存的实体数据集, 那么可用语句:

```
libname ZZZ'D : \mydata';
```

```
data ZZZ.a;
```

其中 " ZZZ " 和 " a " 可以任意改动.

## 2. input x1-x7 y1 y2;

```
cards;
```

```
30 5 30 0 0 8 45 42.3 0.9
```

```
30 10 40 10 0.1 8.5 55 8.27 1.98
```

```
;
```

这是一个以计算机数据卡片进行输入数据的一种数据输入方式. 建立的SAS数据集以列向量形式放置. " input " 语句是对相应的数据集的各个列向量命名, 这些列向量称为变量. 所有列向量组成的数据集是一个矩阵.

" cards; " 后面是实际数据, 同行数据中间用空格间隔, 不同行数据之间不需要换行符.

"; " 也是一个语句, 称为空语句, 单独使用的含义是不进行任何运算, 然后继续进行后面的运算. 在这里和 " cards; " 语句联合使用, 表示数据行结束.

如果想在计算机硬盘 " D: " 的路径为 " D : \mydata " 下读取的实体文本数据集 " z.txt " , 那么可用语句:

```
data a;
```

```
infile 'D:\ mydata \ z.txt';
```

```
input x1-x7 y1 y2;
```

## 3. data aa; set a0;

建立一个以aa命名的虚拟数据集. " set a0; " 的含义是将以a0命名的虚拟数据集后接在以aa命名的虚拟数据集之中.

" data aa; set a0 b0; " 的含义是将以a0 和b0 命名的虚拟数据集依次纵向连接在以aa 命名的虚拟数据集之中. 如果虚拟数据集a0 和b0 的变量同名, 那么将相应的变量纵向连接在一起.

## 4. data www; merge bb a0;

建立一个以www命名的虚拟数据集. " merge bb a0; " 的含义是将以bb 和a0 命名的虚拟数据集依次横向并列在以aa 命名的虚拟数据集之中. 如果后面虚拟数据集a0和前面虚拟数据集bb 的变量同名, 那么用后面虚拟数据集a0 的变量替换前面虚拟数据集bb 的相应变量.

如果想在计算机硬盘 "D:" 的路径为 "D : \mydata" 下存入的实体文本数据集 "xxx.txt", 那么可用语句:

```
data www; merge bb a0;
t=_n.; p='&'; pp='\ \';
file 'D:\ mydata \ xxx.txt';
put t p x1 comma.2 p x2 comma.2 p x3 comma.2 p x4 comma.2 p x5 comma.2 p x6
comma.2 p x7 comma.2 p z comma.4 p zz comma.4 pp;
```

其中 "p='&';" 和 "pp='\ \';" 为字符变量的输入形式; "comma.2" 和 "comma.4" 指相应的变量保留2位和4位小数, 而整数部分每隔三位可以加一个逗号 ", ". " \_n\_ " 为计算机默认变量, 其含义是数据的序号.

```
5. x8=log(x7); x9=sin(x7);
```

对数据向量x7施行函数变换, 新的数据向量记为 "x8" 和 "x9". 用函数变换和运算可以得到任意多个变量. 例如 "f1f2=f1\*f2;" 是将变量f1和f2 的对应元素相乘, 得到新的变量f1f2.

### § 1.3.2 过程步

调用 SAS 过程用 "proc" 语句, proc 的含义是 proceeding. 过程的输入和输出基本都是数据集.

```
1. proc factor data=aa nfactors=7 out=bb; var x1-x8;
```

调用因子分析 "factor" 过程程序, 其中 "data=aa" 和 "out=bb" 含义为因子分析 "factor" 过程的输入和输出数据集分别为虚拟数据集aa和bb; "nfactors=7" 的含义是因子分析 "factor" 过程提取 7 个正交公共因子, 并进行标准化处理; "var x1-x8;" 的含义是要从变量 "x1, x2, ..., x8" 中提取正交公共因子.

```
2. proc reg ; model f1-f7=x1-x9; output out=cc p=ff1-ff7;
```

以x1-x9 为自变量, 对f1-f7 中的每一个因变量进行回归分析. "p=ff1-ff7" 的含义是对相应因变量的预测向量记作ff1-ff7, 输入数据集没有声明, 计算机默认为当前数据集. 而 "out=cc" 的含义是输出虚拟数据集为cc.

```
3. proc reg data=b;
```

```
model z=f1-f7 f1f1-f1f7 f2f1-f2f7 f3f1-f3f7 f4f1-f4f7 f5f1-f5f7 f6f1-f6f7 f7f1-f7f7
/method=stepwise;
output out=bb p=zz;
```

以 f1-f7 f1f1-f1f7 f2f1-f2f7 f3f1-f3f7 f4f1-f4f7 f5f1-f5f7 f6f1-f6f7 f7f1-f7f7 为自变量, z为因

变量进行逐步回归分析. " p=zz " 的含义是对相应因变量 z 的预测向量记作zz, " data=b " 的含义是输入虚拟数据集为b; 而 " out=bb " 的含义是输出虚拟数据集为bb.

#### 4. proc print;

在 " Output " 窗口输出当前数据集的所有变量.

如果只在 " Output " 窗口输出数据集www的部分变量, 那么可以用语句:

```
proc print data=www; var x1-x7 z zz;
```

#### 5. %let fun=y2/y1;

```
proc gplot;
```

```
symbol1 i=spline v=star c=green;
```

```
symbol2 i=spline v=circle c=magenta;
```

```
plot (z zz)*t/overlay;
```

```
title 'z=' &fun;
```

此为调用制图过程. " plot(z zz)\*t/overlay;" 的含义是以t 为横坐标,而z 或者zz 为纵坐标制图, 并把两张图叠合在一起. " symbol1 i=spline v=star c=green;" 和 " symbol2 i=spline v=circle c=magenta;" 的含义是叠合图中两条曲线都用连线, 数值分别取 " star " (星形)和 " circle " (空心圆圈), 颜色分别取 " green " 和 " magenta "; " %let fun=y2/y1 " 和 " title 'z=' &fun;" 的含义是输出图形的标题为 " z=y2/y1 " .

### § 1.3.3 全程和宏步

" 宏语句 " 的作用是它相应的语句可以跨越数据步和过程步, 在全程或者 " 宏函数 " 内部运行. " 宏语句 " 一般要定义 " 宏变量 " 和 " 宏函数 ", " 宏函数 " 的自变量也称为 " 宏参数 ", 各种 " 宏函数 " 可以嵌套和并列, " 宏变量 " 可以全程调用, 而 " 宏参数 " 只能在 " 宏函数 " 内部调用. " % " 是常用 " 宏 " 语句的标记, 有时也作为 " 宏函数 " 的调用符号. " & " 是 " 宏变量 " 和 " 宏参数 " 调用符号.

#### 1. %let fun=y2/y1;

这是 " 宏 " 赋值语句. 含义是把字符 " y2/y1 " 赋予宏参数 " fun ". 这一个语句可以用于任何场合.

#### 2. title 'z=' &fun;

这是一个在标题语句中, 采用了 " 宏 " 替换语句的程序. 含义是以 " z=y2/y1 " 为标题. 这里 " 'z=' &fun;" 的含义是把字符'z='和宏参数 " fun " 表示的字符'y2/y1'加在一起.

#### 3. %let fun=y2/y1;

```
proc gplot;
```

```
symbol1 i=spline v=star c=green;
symbol2 i=spline v=circle c=magenta;
plot (z zz)*t/overlay;
title 'z=' &fun;
```

上述语句形成一个宏语句. 由 " %let fun=y2/y1; " 定义宏参数, 而 " title 'z=' &fun; " 调用宏参数.

4. % let n=7; %let m=8;

上述语句形成两个宏语句. 在程序中, " &n " 和 " &m " 的含义都分别是7和8.

5. %macro ff;

```
%do ii=1 %to &n;
%do jj=1 %to &n;
f&ii.f&jj
%end;
%end;
%mend;
```

上述语句形成一个宏函数ff. 根据这个函数ff, 以后简单书写程序 " %ff " , 说明其为对宏函数ff的调用. 在&n=7时, 就相当于书写 " f1f1-f1f7 f2f1-f2f7 f3f1-f3f7 f4f1-f4f7 f5f1-f5f7 f6f1-f6f7 f7f1-f7f7 " . 一般在不知&n的值时, 不能按穷举的方法书写程序, 而宏语句可以在不知&n的值时, 定义程序的书写形式, 让计算机在运行时自己书写具体的程序, 从而实现自动化功能.

## § 1.4 输出结果及其含义

### § 1.4.1 Output窗口的输出结果及其含义

本计算机自动化建模程序, 主要包含三个过程的输出: 因子分析过程的输出、回归分析过程的输出、打印过程的输出.

#### 1. 因子分析过程的输出

主要包含自变量方差阵的特征值和特征向量.

特征值表: