The TABULATE Procedure

Alan Eaton and Ken Howell
SAS Institute

1. Purpose.

The TABULATE procedure has been developed by Ken Howell and myself to produce hierarchically organized tables of descriptive statistics. The hierarchy is defined using discrete class variables. Statistics are computed on continuous variables. TABULATE produces the kind of table shown in figure 1.

The user is required to specify only labelling and formatting information in conjunction with a TABLE statement. The set of tables produced is a superset of the familiar "banner and stub" class of tables. Tables of one, two, or three dimensions can be produced. TABULATE automatically associates the correct titling with the pages, rows, and columns, and automatically fits the table to the size of the physical page. A one dimensional table has only column headings. A two dimensional table has row titles and column headings. A three dimensional table has page, row, and column titles. "Page" in this context refers to a logical dimension of the table rather than to the physical representation of the table.

2. SAS statements required to produce tables

Four statements are required to produce a table: the procedure statement, a CLASSES statement, a VARIABLES statement, and a TABLE statement.

```
PROC TABULATE;
   CLASSES <class variable list>;
   VARIABLES <continuous variable list>;
   TABLE <page expression>, <row expression>,
         <column expression>;
```

The CLASSES statement specifies the set of categories that may be used in the table hierarchy. The VARIABLES statement specifies the set of continuous variables upon which descriptive statistics may be computed. Statistics are requested by including their names in the appropriate page, row, or column expression of the TABLE statement. The statistics which may be requested are the same statistics available from PROC SUMMARY.

N	number of observations in the subgroup
NMISS	the number of observations in the subgroup having missing values for the variable
MEAN	mean
STD	standard deviation
MIN	minimum value
MAX	maximum value
RANGE	range
SUM	sum
USS	uncorrected sum of squares
CSS	corrected sum of squares
STDERR	standard error of the mean
CV	coefficient of variation
T	Student's t value for testing the hypothesis that the population mean is 0

FIGURE 1

```
TABLE PRODUCT, BRANCH, SALES*(N*F=9 SUM MEAN);

PRODUCT= 4005 - WIDGETS
```

	SALES FOR JANUARY 1982		
	N	SUM	MEAN
BRANCH OFFICE=			
RALEIGH	13	201902.85	15530.99
WASHINGTON	9	117575.43	13063.94
CHICAGO	12	181248.49	15104.04
NEW YORK	14	226345.79	16167.56

PRT	probability of a greater absolute value for the Student's t value
VAR	variance

If a variable but no statistic is specified the statistic computed will be SUM. If neither a variable nor a statistic is specified, the statistic computed will be N.

3. The TABLE statement.

The TABLE statement consists of up to three dimension expressions separated by commas. The general form of the statement is

```
TABLE <page expression>, <row expression>,
<column expression>/<options>;.
```

Three examples follow accompanied by the tables they produce. The first example includes only a column expression; the second example includes row and column expressions; the third example includes page, row, and column expressions.

```
TABLE INCOME*(N MEAN RANGE) HOUSING*(MEAN RANGE);
```

(See figure 2.)

```
TABLE STATE*POPGRP,
   INCOME*(N MEAN RANGE) HOUSING*(MEAN RANGE);
```

(See figure 3.)

```
TABLE REGION, STATE*POPGRP,
   INCOME*(N MEAN RANGE) HOUSING*(MEAN RANGE);
```

(See figure 4.)

FIGURE 2

```
TABLE (INCOME*(N MEAN RANGE) HOUSING*(MEAN RANGE));
```

PER CAPITA INCOME			VALUE OF SINGLE FAMILY OWNED HOUSING	
N	MEAN	RANGE	MEAN	RANGE
383.00	4811.08	5615.00	18997.49	33513.00

FIGURE 3

```
TABLE STATE*POPGRP,
      (INCOME*(N MEAN RANGE) HOUSING*(MEAN RANGE));
```

		PER CAPITA INCOME			VALUE OF SINGLE FAMILY OWNED HOUSING	
		N	MEAN	RANGE	MEAN	RANGE
STATE=	SMSA POPULATION AS OF 1975=					
AK	OVER 100000	1.00	6886.00	0.00	33113.00	0.00
AL	50000 TO 75000	1.00	3840.00	0.00	11781.00	0.00
	OVER 100000	4.00	4349.50	904.00	15801.00	6487.00
AR	50000 TO 75000	3.00	4074.33	832.00	12774.33	2888.00
	OVER 100000	1.00	4687.00	0.00	16042.00	0.00
AZ	50000 TO 75000	1.00	4465.00	0.00	16756.00	0.00
	75000 TO 100000	3.00	5231.00	1119.00	20601.00	5243.00
	OVER 100000	2.00	4663.50	557.00	16275.00	588.00
CA	50000 TO 75000	29.00	5261.14	5192.00	24285.83	31603.00
	75000 TO 100000	21.00	5049.10	4071.00	23845.48	19341.00
	OVER 100000	21.00	5223.33	2188.00	23705.76	14897.00
CO	50000 TO 75000	2.00	4744.50	865.00	20121.00	2252.00

(CONTINUED)

FIGURE 4

```
TABLE REGION,
      STATE*POPGRP,
      (INCOME*(N MEAN RANGE) HOUSING*(MEAN RANGE));
```

REGION= NORTH CENTRAL

		PER CAPITA INCOME			VALUE OF SINGLE FAMILY OWNED HOUSING	
		N	MEAN	RANGE	MEAN	RANGE
STATE=	SMSA POPULATION AS OF 1975=					
IA	50000 TO 75000	2.00	4309.00	340.00	15405.00	5268.00
	75000 TO 100000	3.00	4748.67	425.00	15803.33	4365.00
	OVER 100000	2.00	4938.00	74.00	16448.50	3297.00
IL	50000 TO 75000	11.00	5591.09	5230.00	23998.00	27604.00
	75000 TO 100000	4.00	5527.00	1710.00	20610.25	19111.00
	OVER 100000	3.00	5029.33	681.00	19002.00	4886.00
IN	50000 TO 75000	3.00	4350.00	657.00	11791.67	4075.00
	75000 TO 100000	1.00	4136.00	0.00	11555.00	0.00
	OVER 100000	6.00	4570.33	884.00	14022.67	4540.00
KS	75000 TO 100000	1.00	6759.00	0.00	24495.00	0.00
	OVER 100000	3.00	4684.00	731.00	13392.33	2388.00
MI	50000 TO 75000	4.00	5147.25	2353.00	23081.00	15359.00

(CONTINUED)

4. Dimension expressions.

An expression may contain class variable names, continuous variable names, and statistic names. Names may be nested using the nesting or crossing operator, '*', and concatenated using juxtaposition.

```
      Nesting:  INCOME*RANGE
Concatenation:  INCOME HOUSING
```

Concatenations may be nested as a unit by parenthesizing the concatenation.

```
INCOME*(N MEAN RANGE)
```

Dimension expressions obey rules of precedence analogous to the rules of precedence for arithmetic expressions. Nesting corresponds to multiplication, concatenation to addition. Associations within parentheses are evaluated first.

The nesting operator between two class names, as in REGION*POPGRP, indicates that all levels of the right hand class variable are nested within each level of the left hand class value. For example, if REGION has the values 'NORTH', 'SOUTH', 'EAST', 'WEST', and POPGRP has the formatted values '50000 TO 75000', '75000 TO 100000', and 'OVER 100000', the hierarchy produced is

```
NORTH   50000  TO  75000
        75000  TO 100000
        OVER  100000
SOUTH   50000  TO  75000
        75000  TO 100000
        OVER  100000
EAST    50000  TO  75000
        75000  TO 100000
        OVER  100000
WEST    50000  TO  75000
        75000  TO 100000
        OVER  100000
```

The nesting operator may also be used between class names and variable or statistic names to indicate what values should be produced at each leaf in the hierarchical tree. (In the following example, INCOME would have appeared in a VARIABLE statement.)

```
REGION*POPGRP*INCOME

NORTH 50000 TO <income value for NORTH region
      75000     population group 50000 to
                75000>

      75000 TO <income value for NORTH region
      100000    population group 75000 to
                100000>

      OVER      <income value for NORTH region
      100000     population group over 100000>
                .
                .
                .
(and so on for SOUTH, EAST, and WEST)
```

The page, row, and column dimensions are themselves nested one within the other (column within row, row within page) to determine the value associated with each table cell.

Concatenation is accomplished by juxtaposing terms. The terms may be class names, variable names, statistic names, or a nesting. The statement

```
TABLE COUNTY*SEASON COUNTY*DECADE,
   LOTEMP*(MIN MAX RANGE);
```

illustrates the concatenation of nestings and the nesting of a concatenation. (See figure 5.)

Nesting may be arbitrarily deep and concatenations may be arbitrarily wide. It is only necessary that all information for a logical page fit in memory at the same time. It should be noted that logical pages may or may not coincide with physical pages. A concatenation in the column dimension may extend the table to a width greater than the width of the physical page. TABULATE will automatically back up along the column dimension to an appropriate break point, continue printing all rows for the columns that will fit, then begin at row one on the remainder of the columns. Likewise, rows may extend across a page boundary. TABULATE will back up along the row dimension to a point which prevents row titles being broken across the boundary and will begin a new page with complete titles.

Continuous variables and statistics may appear in any dimension, separately or together. A continuous variable may not be nested with a continuous variable and a statistic may not be nested with a statistic. For example, the following statements are not valid if INCOME and HOUSING are continuous variables.

```
TABLE INCOME*(N MEAN), HOUSING;
TABLE INCOME*(N MEAN), (VAR STD);
```

5. Other SAS statements

Other SAS statements that may be used with TABULATE are LABEL, FORMAT, FREQ, TITLE, and BY. Use the LABEL statement to replace class or continuous variable names with descriptive text. Use PROC FORMAT and FORMAT statements to format class values. TABULATE will use these supplied values as text in the page, row, and column titles. The text will be broken into multiple lines to fit the space available.

```
PROC FORMAT;
   VALUE SEASNFMT 1 = SPRING
                  2 = SUMMER
                  3 = FALL
                  4 = WINTER;
   VALUE DECFMT  60-69 = SIXTIES
                 70-79 = SEVENTIES;


PROC TABULATE;
   VARIABLE TEMP;
   CLASSES COUNTY SEASON DECADE;
   LABEL COUNTY = 'COUNTIES OF NORTH CAROLINA';
   LABEL SEASON = 'SEASONS OF THE YEAR';
   FORMAT SEASON SEASNFMT.;
   FORMAT DECADE DECFMT.;
   TABLE COUNTY*SEASON COUNTY*DECADE,
      LOTEMP*(MIN MAX RANGE);
```

(See figure 5.)

FIGURE 5

```
TABLE COUNTY*SEASON COUNTY*DECADE,
      LOTEMP*F=5 *(MIN MAX RANGE);
```

		LOTEMP		
		MIN	MAX	RANGE
COUNTIES OF NORTH CAROLINA=	SEASONS OF THE YEAR=			
ALAMANCE	WINTER	-8	42	50
	SPRING	35	55	20
	SUMMER	55	80	25
	FALL	42	75	33
COUNTIES OF NORTH CAROLINA=	DECADE=			
ALAMANCE	SIXTIES	0	80	80
	SEVENTIES	-8	76	84

6. Universal class variable.

The universal class variable is a special class which has only one level value. It is specified by using the name ALL anywhere that a class name may be used. The effect of specifying the universal class variable is to construct a subgroup which includes all sub-levels in the hierarchy. The following examples illustrate its use.

```
TABLE COUNTY*SEASON ALL, LOTEMP*(MIN MAX RANGE);
```

(See figure 6.)

```
TABLE COUNTY*(SEASON ALL), LOTEMP*(MIN MAX RANGE);
```

(See figure 7.)

FIGURE 6

```
TABLE COUNTY*SEASON ALL,LOTEMP*F=5 *(MIN MAX RANGE);
```

		LOTEMP		
		MIN	MAX	RANGE
COUNTIES OF NORTH CAROLINA=	SEASONS OF THE YEAR=			
ALAMANCE	WINTER	0	42	42
	SPRING	40	55	15
	SUMMER	60	80	20
	FALL	42	75	33
ALEXANDER	WINTER	-8	36	44
	SPRING	35	50	15
	SUMMER	55	76	21
	FALL	42	72	30
ALL		-8	80	88

FIGURE 7

```
TABLE COUNTY*(SEASON ALL),LOTEMP*(MIN MAX RANGE)
```

		LOTEMP		
		MIN	MAX	RANGE
COUNTIES OF NORTH CAROLINA=	SEASONS OF THE YEAR=			
ALAMANCE	WINTER	0	42	42
	SPRING	40	55	15
	SUMMER	60	80	20
	FALL	42	75	33
ALEXANDER	WINTER	-8	36	44
	SPRING	35	50	15
	SUMMER	55	76	21
	FALL	42	72	30
COUNTIES OF NORTH CAROLINA=	ALL			
ALAMANCE		0	80	80
ALEXANDER		-8	76	84

7. Formatting table values.

Formatting is limited to w.d numeric formats. Formats are specified by nesting a format item at the appropriate location in an expression.

```
TABLE REGION*POPGRP,
   INCOME*(N*F=5 MEAN*F=8.2 MIN*F=8.2);
```

(See figure 8.)

Format items may occur in any dimension and may be nested with any type of name, class, continuous variable, or statistic. If a format is nested with a class name the format will apply to all variables and statistics that become nested with that class (directly or indirectly). In the event that two formats become associated with the same table cell, e.g.,

```
TABLE REGION*POPGRP*F=10.2, INCOME*(N*F=5 MEAN MIN);
```

FIGURE 8

```
TABLE REGION*POPGRP,
      INCOME*(N*F=5 MEAN*F=8.2 MIN*F=8.2);
```

		PER CAPITA INCOME		
		N	MEAN	MIN
REGION=	SMSA POPULATION AS OF 1975=			
NORTH CENTRAL	50000 TO 75000	38	5124.00	2593.00
	75000 TO 100000	29	5115.69	3885.00
	OVER 100000	39	4684.56	3925.00
NORTHEAST	50000 TO 75000	29	4574.14	3695.00
	75000 TO 100000	14	4588.00	3207.00
	OVER 100000	27	4401.48	3348.00
SOUTH	50000 TO 75000	33	4367.73	2196.00
	75000 TO 100000	12	4241.67	2279.00
	OVER 100000	59	4628.17	3479.00
WEST	50000 TO 75000	37	5142.27	3226.00
	75000 TO 100000	29	5050.62	2839.00
	OVER 100000	37	5146.92	4128.00

the format with the least scope will override any others. (See figure 9.) In both of the above examples, N will have the format F=5. The effect of this scheme is that column formats override row formats when multiple formats are associated with the same table cell.

Column width is taken to be the widest format width associated with any cell in the column. The formats of cells in a column may differ between row concatenations. For example,

```
TABLE REGION*(INCOME*F=8.2 HOUSING*F=12.2),
   N*F=5 MEAN RANGE;
```

(See figure 10.)

FIGURE 9

```
TABLE REGION*POPGRP*F=10.2
      INCOME*(N*F=5 MEAN MIN);
```

		PER CAPITA INCOME		
		N	MEAN	MIN
REGION=	SMSA POPULATION AS OF 1975=			
NORTH CENTRAL	50000 TO 75000	38	5124.00	2593.00
	75000 TO 100000	29	5115.69	3885.00
	OVER 100000	39	4684.56	3925.00
NORTHEAST	50000 TO 75000	29	4574.14	3695.00
	75000 TO 100000	14	4588.00	3207.00
	OVER 100000	27	4401.48	3348.00
SOUTH	50000 TO 75000	33	4367.73	2196.00
	75000 TO 100000	12	4241.67	2279.00
	OVER 100000	59	4628.17	3479.00
WEST	50000 TO 75000	37	5142.27	3226.00
	75000 TO 100000	29	5050.62	2839.00
	OVER 100000	37	5146.92	4128.00

FIGURE 10

```
TABLE REGION*(INCOME*F=8.2 HOUSING*F=12.2),
      N*F=5 MEAN RANGE;
```

		N	MEAN	RANGE
REGION=				
NORTH CENTRAL	PER CAPITA INCOME	106	4960.05	5886.00
	VALUE OF SINGLE FAMILY OWNED HOUSING	106	18567.50	27604.00
NORTHEAST	PER CAPITA INCOME	70	4510.31	3922.00
	VALUE OF SINGLE FAMILY OWNED HOUSING	70	19042.16	36164.00
SOUTH	PER CAPITA INCOME	104	4500.93	5062.00
	VALUE OF SINGLE FAMILY OWNED HOUSING	104	15127.09	31333.00
WEST	PER CAPITA INCOME	103	5118.14	6002.00
	VALUE OF SINGLE FAMILY OWNED HOUSING	103	22678.10	36011.00

8. Options and parameters on TABLE statement.

NOZERO — Indicates that within a logical page any row which has all zero or missing values should not be printed. Default action is to print all rows.

FUZZ=.nnn — Specifies a number to be used in testing values for 0. If a value is within FUZZ of 0 the value is set to 0 for calculations and printing.

ZERO='text' — Specifies that 'text' will appear in any table cell whose value is 0 in lieu of the zero value.

MISSING — Indicates that missing class values should be treated as a classification level. Default action will be to bypass observations in which there is a missing class value.

VCHAR='character'
HCHAR='character'
CCHAR='character'

Specify the vertical, horizontal, and corner characters to be used in delimiting title text and table cells. Default values are

```
VCHAR='|'
HCHAR='-'
CCHAR='+'
```

RTSPACE=<integer>
Specifies the total width (including delimiters) allotted to row titles. Default value is 1/4 of the line size.

9. Relevant system options.

```
LINESIZE
PAGESIZE
CENTER
```

TABULATE will use these system options to establish the number of print positions in a line, the number of lines on a page, and the horizontal position of the table on the page.

INFORMATION SYSTEMS

SAS Interfaces: Data Bases and Data Base Management Systems