

Some Tips and Examples for Using SAS® PROC TABULATE

Pei Lee Tien, ASG Inc., Stamford, CT

Tsung-Hua Lin, Ph. D., Sandoz Pharmaceuticals Corporation, East Hanover, NJ

Mike McGranaghan, Contino Associates Inc., North Caldwell, NJ

ABSTRACT

This paper has been prepared to promote the use of PROC TABULATE for producing statistical tables, as an alternative to using PROC REPORT or FILE PRINT. The techniques presented combine basic TABULATE statements with DATA steps and other traditional SAS procedures (FORMAT, FREQ and UNIVARIATE) to produce report-ready output. This approach avoids many of the more complex TABULATE methods described in SAS publications, such as using angled brackets to specify denominator definitions. Through examples, PROC TABULATE is shown to be an efficient report writer, capable of displaying a variety of statistics not ordinarily associated with this procedure (e.g. median, 95% confidence intervals, and p-values).

In general the techniques discussed in this paper can be used for all operating systems. However, the FORMCHAR statements are system-dependent, and apply to IBM/CMS only.

INTRODUCTION

It is authors experience that PROC TABULATE, although very efficient and powerful, has been under-utilized by SAS users. We believe this is partly due to the fact that its SAS manual fails to show the real potential of PROC TABULATE. For example, users often complain about the frustration in finding an appropriate denominator for calculating the percentage statistic within PROC TABULATE. In fact, this problem can be easily circumvented by adding some simple data steps. Another reason of under usage is that the manual leaves the impression that the procedure can produce statistics available only within PROC TABULATE. This becomes an obvious restriction

for users who need, for example, MEDIAN, P-VALUE, or even P-VALUE with significance flags. Again, PROC TABULATE is capable of performing all these tasks, by adding appropriate data steps and format statements.

This paper uses four examples to show readers how to put summary statistics, confidence intervals and p-values in a nice table format. Although useful, the summary statistics N, MEAN, SD, MIN, and MAX provided within PROC TABULATE are rarely used because MEDIAN is not available. Instead, PROC UNIVARIATE is generally used to pre-process the data for the desired statistics which are then fed into PROC TABULATE for a final presentation. The principle of calculating statistics outside of PROC TABULATE applies also to providing confidence intervals and p-values in a table.

Finally, there is an exercise involving counts, percentages and summary statistics. This is an opportunity for readers to apply a couple of the basic techniques shown in this paper so far.

Sample Data Set DS_DEMO:

Data set DS_DEMO contains 20 observations with 5 variables as:

TMT (1=A, 2=B)
SUBJ (1, 2, ..., 20 randomly assigned to two treatment groups with 10 subjects each)
AGE (years)
SEX (1=Male, 2=Female)
WEIGHT (kgs)

This same sample data set is used in all the three examples and the final exercise.

OBS	TMT	SUBJ	AGE	SEX	WT
1	1	4	16	2	71
2	1	6	73	1	85
3	1	9	39	2	85
4	1	10	55	1	69
5	1	11	72	2	72
6	1	12	23	1	63
7	1	15	54	2	74
8	1	16	46	2	65
9	1	17	45	1	79
10	1	20	22	2	76
11	2	1	17	2	75
12	2	2	29	2	75
13	2	3	79	2	54
14	2	5	66	1	67
15	2	7	38	2	41

OBS	TMT	SUBJ	AGE	SEX	WT
16	2	8	25	2	59
17	2	13	18	1	61
18	2	14	33	2	68
19	2	18	35	2	66
20	2	19	44	1	75

```

FORMAT tmt tmtfmt.;
TABLES (age='Age (years)' wt='Weight')*
(N='N'*F=6.0 MEAN='Mean'*F=7.1
STD='SD'*F=8.2 MIN='Min'*F=6.0
MAX='Max'*F=6.0),
tmt='Treatment'
/ RTS=30;
RUN;

```

Example 1: Use Summary Statistics Within PROC TABULATE

Table Shell for Example 1:

		Treatment	
		A (N=)	B (N=)
Age (years)	N	99	99
	Mean	99.9	99.9
	SD	99.99	99.99
	Min	99	99
	Max	99	99
Weight	N	99	99
	Mean	99.9	99.9
	SD	99.99	99.99
	Min	99	99
	Max	99	99

Output:

Table Example 1
Use N, MEAN, STD, MIN, and MAX Statistics in PROC TABULATE

		Treatment	
		A (N=10)	B (N=10)
Age (years)	N	10	10
	Mean	44.5	38.4
	SD	19.95	20.07
	Min	16	17
	Max	73	79
Weight	N	10	10
	Mean	73.9	64.1
	SD	7.53	10.83
	Min	63	41
	Max	85	75

The following SAS code demonstrate an application of PROC TABULATE to produce all the required statistics, but MEDIAN.

SAS Code:

```

PROC FORMAT;
VALUE tmt
1 = 'A'
2 = 'B'
;
RUN;

/* Combine treatment name (A/B) with treatment total (N= ) */
/* The following code are used to create treatment format
TMTFMT for all the following examples. However, they are
not included in the other examples. */

PROC SORT DATA=ds_demo; BY tmt subj;
RUN;

DATA tmtf (KEEP=START LABEL FMTNAME);
SET ds_demo; BY tmt subj;
LENGTH LABEL $8;
IF FIRST.tmt THEN tmtot=0;
tmtot+1;
IF LAST.tmt THEN DO;
START=tmt;
LABEL = PUT(tmt,tmt.) ||
(N=' ' || TRIM(LEFT(tmtot)) || ');
FMTNAME='tmtfmt';
OUTPUT;
END;
RUN;

/* Create treatment format TMTFMT */

PROC FORMAT CNTLIN=tmtf;
RUN;

TITLE1 'Table Example 1';
TITLE2 'Use N, MEAN, STD, MIN, and MAX Statistics';
TITLE3 'in PROC TABULATE';

PROC TABULATE DATA=ds_demo FORMCHAR='FABFACCCBCEB8FECACBBB'X;
/* The above formchar statement produces solid lines for the table.
The formchar value is system-specific. For more information,
consult SAS manual, 2nd ed., page 134. */
CLASS tmt;
VAR age wt;

```

Example 2: Use PROC UNIVARIATE for Pre-Processing to Calculate Summary Statistics

Table Shell for Example 2:

		Treatment	
		A (N=)	B (N=)
Age (years)	N	99	99
	Mean	99.9	99.9
	SD	99.99	99.99
	Min	99	99
	Max	99	99
Weight	N	99	99
	Mean	99.9	99.9
	SD	99.99	99.99
	Min	99	99
	Max	99	99

The following SAS code demonstrate the use of PROC UNIVARIATE so that the statistic MEDIAN is shown in the table. Further, this example also adds a few data steps and PROC FORMAT to line up the statistics by the decimal points, improving the table appearance.

SAS Code:

```

PROC FORMAT;
value collfmt

```

```

1 = 'Age (Years)'
2 = 'Weight' ;
value col2fmt
1 = 'N'
2 = 'Mean'
3 = 'SD'
4 = 'Median'
5 = 'Min'
6 = 'Max' ;
VALUE tmt
1 = 'A'
2 = 'B' ;
RUN;

/* See Example 1 for how to create format TMTFMT */

%MACRO univs (var=);
/* Calculate UNIVARIATE statistics */
PROC UNIVARIATE DATA=ds_demo NOPRINT;
BY tmt;
VAR &var;
OUTPUT OUT=univ N=n MEAN=mean STD=std
MEDIAN=median MIN=min MAX=max;
RUN;

/* Create variable LABEL to line up decimal points */
DATA &var (KEEP=col2 tmt label); SET univ;
LENGTH LABEL $8 ;
ARRAY summary n mean std median min max;
DO OVER summary;
col2 = _I_;
IF col2 in (1, 5, 6) THEN LABEL=PUT(summary,5.0); * N, Min,
Max;
ELSE IF col2 in (2, 4) THEN
LABEL=PUT(summary,7.1); * Mean, Median;
ELSE IF col2=3 then LABEL=PUT(summary,8.2); * SD;
OUTPUT;
END;
RUN;
%MEND univs;

%univs(var=age)
%univs(var=wt)

DATA final; SET age (IN=a) wt (IN=b) ;
IF a THEN coll=1; * Age (Years);
ELSE IF b THEN coll=2; * Weight;
START+1;
FMTNAME='valuef';
RUN;

/* Create format VALUEF */
PROC FORMAT CNTLIN=final (keep=START LABEL FMTNAME); RUN;

PROC SORT DATA=final; BY tmt coll col2; RUN;

TITLE1 'Table Example 2';
TITLE2 'Use PROC UNIVARIATE for Pre-Processing';
TITLE3 'to Calculate N, MEAN, STD, MEDIAN, MIN, and MAX';

PROC TABULATE DATA=final FORMCHAR='FABFACCCBCEB8FECABCBB'B'X;
CLASS coll col2 tmt;
VAR start;
FORMAT coll collfmt. col2 col2fmt. tmt tmtfmt.;
TABLES coll=' '*col2=' ',
tmt='Treatment'*start=' '*SUM=' '*F=valuef10.
/ RTS=32;
RUN;

```

Important note: Readers are asked to carefully review the statements involving the variable START and its format VALUEF in the above SAS codes. Also, please note the format VALUEF10 is used to display format VALUEF using 10 character spaces

Data Set Used in PROC TABULATE:

OBS	TMT	COL1	COL2	START	LABEL
1	1	1	1	1	10
2	1	1	2	2	44.5
3	1	1	3	3	19.95
4	1	1	4	4	45.5
5	1	1	5	5	16
6	1	1	6	6	73
7	1	2	1	13	10
8	1	2	2	14	73.9
9	1	2	3	15	7.53
10	1	2	4	16	73.0
11	1	2	5	17	63
12	1	2	6	18	85
13	2	1	1	7	10
14	2	1	2	8	38.4
15	2	1	3	9	20.07
16	2	1	4	10	34.0
17	2	1	5	11	17
18	2	1	6	12	79
19	2	2	1	19	10
20	2	2	2	20	64.1
21	2	2	3	21	10.83
22	2	2	4	22	66.5
23	2	2	5	23	41
24	2	2	6	24	75

Output:

Table Example 2
Use PROC UNIVARIATE for Pre-Processing
to Calculate N, MEAN, STD, MEDIAN, MIN, and MAX

		Treatment	
		A (N=10)	B (N=10)
Age (Years)	N	10	10
	Mean	44.5	38.4
	SD	19.95	20.07
	Median	45.5	34.0
	Min	16	17
	Max	73	79
Weight	N	10	10
	Mean	73.9	64.1
	SD	7.53	10.83
	Median	73.0	66.5
	Min	63	41
	Max	85	75

Example 3: Use PROC UNIVARIATE and PROC FREQ for Pre-Processing to Calculate Summary Statistics, Including Confidence Intervals and P-Values

Table Shell for Example 3:

		Treatment		
		A (N=)	B (N=)	
Age (Years)	N	99	99	
	Mean	99.9	99.9	
	SD	99.99	99.99	
	95% CI	(99.9, 99.9)	(99.9, 99.9)	
	Median	99.9	99.9	
	Min	99	99	
	Max	99	99	
	P-Value #	0.999	0.999	
	P-Value \$	0.999		
	Weight	N	99	99
		Mean	99.9	99.9
SD		9.99	99.99	
95% CI		(99.9, 99.9)	(99.9, 99.9)	
Median		99.9	99.9	
Min		99	99	
Max		99	99	
P-Value #		0.999	0.999	
P-Value \$		0.999		

p-value within treatment group.
\$ p-value between treatment groups.

SAS Code:

```

PROC FORMAT;
VALUE collfmt
1 = 'Age (Years)'
2 = 'Weight'
;
VALUE col2fmt
1 = 'N'
2 = 'Mean'
3 = 'SD'

```

Data Set Used in PROC TABULATE:

```

4 = '95% CI'
5 = 'Median'
6 = 'Min'
7 = 'Max'
8 = 'P-Value #'
9 = 'P-Value $'
;
PICTURE pvalue (ROUND)
    . = ' ' (NOEDIT)
    LOW-<.001 = '<0.001' (NOEDIT)
    .001-HIGH = '09.999'
;
VALUE tmt
    1 = 'A'
    2 = 'B'
;
RUN;

/* See Example 1 for how to create format TMTFMT */
%macro univs (var=);
PROC UNIVARIATE DATA=ds_demo NOPRINT;
    BY tmt;
    VAR &var;
    OUTPUT OUT=univ N=n MEAN=mean STD=std MEDIAN=median
        MIN=min MAX=max PROBS=pval;
RUN;

/* Get the 95% Confident Interval of population mean */
DATA univ1; SET univ;
    ci_high=mean+1.96*(std/SQRT(n));
    ci_low=mean-1.96*(std/sqrt(n));
RUN;

/* Get between treatment group p-value with the van Elteren test */
PROC FREQ DATA=ds_demo NOPRINT;
    TABLES tmt*&var / CMH SCORES=MODRIDIT;
    OUTPUT OUT=van (KEEP=P_CMHRMS RENAME=(p_cmhrms=cmh)) CMHRMS;
RUN;

DATA van1; SET van;
    tmt=1; * Place all between group p-values under treatment
        group A column;
RUN;

DATA all; MERGE univ1 van1; BY tmt;
RUN;

/* Create variable LABEL to line up decimal points */
DATA &var (KEEP=col2 tmt label); SET all;
LENGTH LABEL $14 ci 8;
ARRAY summary n mean std ci median min max pval cmh;
ci_label='(' || PUT(ci_low,5.1) || ' ' ||
    ' ' || PUT(ci_high,5.1) || ')';
DO OVER summary;
    col2 = _I_;
    IF col2 IN (1, 6, 7) THEN LABEL=PUT(summary,7.);
    * N, Min, Max;
    ELSE IF col2 IN (2, 5) THEN LABEL=PUT(summary,9.1);
    * Mean, Median;
    ELSE IF col2=3 THEN LABEL=PUT(summary,10.2); * SD;
    ELSE IF col2=4 THEN LABEL=ci_label; * 95% CI;
    ELSE IF col2 IN (8,9) THEN
        LABEL=PUT(summary,pvalue1.);
    * p_value within and between groups;
OUTPUT;
END;
RUN;

%mend univs;

%univs(var=age)
%univs(var=wt)

DATA final; SET age (IN=a) wt (IN=b);
    IF a THEN coll=1; * Age (Years);
    ELSE IF b THEN coll=2; * Weight;
    START+1;
    FMTNAME='valuef';
RUN;

PROC FORMAT CNTLIN=final (keep=START LABEL FMTNAME);
PROC SORT DATA=final; BY tmt coll col2;
RUN;

TITLE1 'Table Example';
TITLE3 'Use PROC UNIVARIATE and PROC FREQ for Pre-Processing';
TITLE4 'to Calculate Summary Statistics and P-Values';
FOOTNOTE1 '# p-value within treatment group.';
FOOTNOTE2 '$ p-value between treatment groups.';

PROC TABULATE DATA=final FORMCHAR='FABFACCCBCEB8FECABCBBB'X;
    CLASS coll col2 tmt;
    VAR start;
    FORMAT coll collfmt. col2 col2fmt. tmt tmtfmt.;
    TABLES coll=' '*col2=' ',
        tmt='Treatment'*start=' *SUM=' '*P=valuef14.
    / RTS=32;
RUN;

```

OBS	TMT	COL1	COL2	START	LABEL
1	1	1	1	1	10
2	1	1	2	2	44.5
3	1	1	3	3	19.95
4	1	1	4	4	(32.1, 56.9)
5	1	1	5	5	45.5
6	1	1	6	6	16
7	1	1	7	7	73
8	1	1	8	8	0.002
9	1	1	9	9	0.406
10	1	2	1	19	10
11	1	2	2	20	73.9
12	1	2	3	21	7.53
13	1	2	4	22	(69.2, 78.6)
14	1	2	5	23	73.0
15	1	2	6	24	63
16	1	2	7	25	85
17	1	2	8	26	0.002
18	1	2	9	27	0.049
19	2	1	1	10	10
20	2	1	2	11	38.4
21	2	1	3	12	20.07
22	2	1	4	13	(26.0, 50.8)
23	2	1	5	14	34.0
24	2	1	6	15	17
25	2	1	7	16	79
26	2	1	8	17	0.002
27	2	1	9	18	
28	2	2	1	28	10
29	2	2	2	29	64.1
30	2	2	3	30	10.83
31	2	2	4	31	(57.4, 70.8)
32	2	2	5	32	66.5
33	2	2	6	33	41
34	2	2	7	34	75
35	2	2	8	35	0.002
36	2	2	9	36	

Output:

Table Example 3

Use PROC UNIVARIATE and PROC FREQ for Pre-Processing to Calculate Summary Statistics and P-Values

		Treatment	
		A (N=10)	B (N=10)
Age (Years)	N	10	10
	Mean	44.5	38.4
	SD	19.95	20.07
	95% CI	(32.1, 56.9)	(26.0, 50.8)
	Median	45.5	34.0
	Min	16	17
	Max	73	79
	P-Value #	0.002	0.002
	P-Value \$	0.406	
Weight	N	10	10
	Mean	73.9	64.1
	SD	7.53	10.83
	95% CI	(69.2, 78.6)	(57.4, 70.8)
	Median	73.0	66.5
	Min	63	41
	Max	85	75
	P-Value #	0.002	0.002
	P-Value \$	0.049	

p-value within treatment group.
\$ p-value between treatment groups.

Review

For tables involving only summary statistics, PROC UNIVARIATE is recommended if the median is required. Otherwise, the statistics within PROC TABULATE would suffice (see Examples 1 and 2).

To display other statistics (e.g. confidence intervals, p-values, etc) or to improve table appearance, appropriate format statements should be used (see Examples 2 and 3).

Exercise

Table Shell for Exercise:

		Treatment	
		A (N=)	B (N=)
Age (Years)	N	99	99
	< 18	9 (99%)	9 (99%)
	18-64	9 (99%)	9 (99%)
	>64	9 (99%)	9 (99%)
	Mean	99.9	99.9
	SD	99.99	99.99
	Median	99.9	99.9
	Min	99	99
Sex	N	99	99
	Male	99 (99%)	99 (99%)
	Female	99 (99%)	99 (99%)
Weight	N	99	99
	Mean	99.9	99.9
	SD	99.99	99.99
	Median	99.9	99.9
	Min	99	99
	Max	99	99

Note that the table now includes counts and percentages. This requires some careful format statements.

Answer:

SAS Code:

```
PROC FORMAT;
  VALUE col1fmt
    1 = 'Age (Years)'
    2 = 'Sex'
    3 = 'Weight';
  VALUE col2fmt
    1 = 'N'
    2 = '< 18'
    3 = '18-64'
    4 = '>64'
    5 = 'Male'
    6 = 'Female'
    11 = 'Mean'
    12 = 'SD'
    13 = 'Median'
    14 = 'Min'
    15 = 'Max' ;
  VALUE agegrp
```

```

    0 -< 18 = 2
    18 - 64 = 3
    64 <- high = 4;
  VALUE sex
    1 = 5
    2 = 6 ;
  VALUE tmt
    1 = 'A'
    2 = 'B';
RUN;

/* See Example 1 for how to create format TMTFMT */
PROC FREQ DATA=ds_demo NOPRINT;
  FORMAT age, agegrp, sex sex.;
  TABLE tmt * age / OUT = agetot (DROP=PERCENT);
  TABLE tmt * sex / OUT = sextot (DROP=PERCENT);
  TABLE tmt / OUT = tmtot (DROP=PERCENT);
RUN;

DATA age_grp; MERGE agetot tmtot (RENAME=(COUNT=denom));
  BY tmt;
  LENGTH LABEL $11;
  LABEL = PUT(count,4.0) || ' (' ||
    PUT((count/denom)*100,3.0) || '%)';
  coll1=1; * Age (Years);
  coll2=INFUT(PUT(age,agegrp.),1.0);
RUN;

/* Get the counts of male/female by treatment */
DATA sex_m_f; MERGE sextot tmtot (RENAME=(COUNT=denom));
  BY tmt;
  LENGTH LABEL $11;
  LABEL = PUT(count,4.0) || ' (' ||
    PUT((count/denom)*100,3.0) || '%)';
  coll1=2; * Sex;
  coll2=INFUT(PUT(sex,sex.),1.0); * Male/Female;
RUN;

PROC FREQ DATA=ds_demo (WHERE=(SEX IS NOT MISSING)) NOPRINT;
  TABLE tmt / OUT = sexn (DROP=PERCENT);
RUN;

/* Get total number of male and female by treatment */
DATA sex_tot; SET sexn;
  LENGTH LABEL $11;
  LABEL = put(count,4.0);
  coll1=2; *Sex;
  coll2=1; * N;
RUN;

/* Calculate UNIVARIATE statistics */
%MACRO univ (var=);
PROC UNIVARIATE DATA=ds_demo NOPRINT;
  BY tmt;
  VAR &var;
  OUTPUT OUT=univ N=n MEAN=mean STD=std MEDIAN=median
    MIN=min MAX=max;
RUN;

DATA &var (KEEP=coll2 tmt label); SET univ;
  LENGTH LABEL $11;
  ARRAY summary n mean std median min max;
  DO OVER summary;
    IF _i_ = 1 then coll2=1; * N;
    ELSE coll2 = _i_+9; * Mean, SD, Median, Min, Max;
    IF coll2 IN (1, 14, 15) THEN LABEL=PUT(summary,4.0);
    * N, Min, Max;
    ELSE IF coll2 IN (11, 13) THEN LABEL=PUT(summary,6.1);
    * Mean, Median;
    ELSE IF coll2=12 THEN LABEL=PUT(summary,7.2);* SD;
  OUTPUT;
  END;
RUN;

%MEND univ;

%univ(var=age)
%univ(var=wt)

DATA univ; SET age (IN=a) wt (IN=b) ;
  IF a THEN coll1=1; * Age (Years);
  ELSE IF b THEN coll1=3; * Weight;
RUN;

DATA final; SET age_grp sex_tot sex_m_f univ;
  START+1;
  FMTNAME='valuef';
RUN;

PROC FORMAT CNTLIN=final (KEEP= START LABEL FMTNAME);
PROC SORT DATA=final; BY tmt coll1 coll2; RUN;

TITLE1 'Table Exercise';
TITLE3 'Use PROC UNIVARIATE and PROC FREQ for Pre-Processing ';
TITLE4 'to Calculate Summary Statistics, Counts and Percentages';

PROC TABULATE DATA=final FORMCHAR='FABFACCCBCEB8FECABCBBB'X;
  CLASS coll1 coll2 tmt;
  VAR start;
  FORMAT coll1 col1fmt. coll2 col2fmt. tmt tmtfmt.;
  TABLES coll1= ' '*coll2= ' ',
    tmt='Treatment'*start= '*SUM=' '*F=valuef11.
  / RTS=32;
RUN;
```

Data Set Used in PROC TABULATE:

OBS	TMT	COL1	COL2	START	LABEL
1	1	1	1	13	10
2	1	1	2	1	1 (10%)
3	1	1	3	2	7 (70%)
4	1	1	4	3	2 (20%)
5	1	1	11	14	44.5
6	1	1	12	15	19.95
7	1	1	13	16	45.5
8	1	1	14	17	16
9	1	1	15	18	73
10	1	2	1	7	10
11	1	2	5	9	4 (40%)
12	1	2	6	10	6 (60%)
13	1	3	1	25	10
14	1	3	11	26	73.9
15	1	3	12	27	7.53
16	1	3	13	28	73.0
17	1	3	14	29	63
18	1	3	15	30	85
19	2	1	1	19	10
20	2	1	2	4	1 (10%)
21	2	1	3	5	7 (70%)
22	2	1	4	6	2 (20%)
23	2	1	11	20	38.4
24	2	1	12	21	20.07
25	2	1	13	22	34.0
26	2	1	14	23	17
27	2	1	15	24	79
28	2	2	1	8	10
29	2	2	5	11	3 (30%)
30	2	2	6	12	7 (70%)
31	2	3	1	31	10
32	2	3	11	32	64.1
33	2	3	12	33	10.83
34	2	3	13	34	66.5
35	2	3	14	35	41
36	2	3	15	36	75

Output:

Table Exercise

Use PROC UNIVARIATE and PROC FREQ for Pre-Processing to Calculate Summary Statistics, Counts and Percentages

		Treatment	
		A (N=10)	B (N=10)
Age (Years)	N	10	10
	< 18	1 (10%)	1 (10%)
	18-64	7 (70%)	7 (70%)
	>64	2 (20%)	2 (20%)
	Mean	44.5	38.4
	SD	19.95	20.07
	Median	45.5	34.0
	Min	16	17
Sex	N	10	10
	Male	4 (40%)	3 (30%)
	Female	6 (60%)	7 (70%)
Weight	N	10	10
	Mean	73.9	64.1
	SD	7.53	10.83
	Median	73.0	66.5
	Min	63	41
	Max	85	75

CONCLUSION

In the past, it seems that programmers often resort to PROC REPORT or FILE PRINT for table generation, because PROC TABULATE met most, but not all of their needs. However, through the examples presented in this paper, the authors have shown that PROC TABULATE can be an

efficient report writer, capable of producing a variety of displays. Statistics not ordinarily associated with this procedure are easily incorporated, without the use of many of the traditionally complex TABULATE methods.

This paper does not replace the official SAS manual for PROC TABULATE. It is an aid for users who possess basic understanding of the procedure. Syntax presented in this paper includes: PROC TABULATE <DATA, FORMCHAR, MISSING>, CLASS, VAR, TABLE</MISSTEXT, RTS>. Users are encouraged to familiarize themselves with these TABULATE statements.

ACKNOWLEDGEMENT

The authors would like to express their appreciation to Isabelle Bourdeix, Cornelia Dellenbaugh and Nai-Huei Chen for their contribution to the ideas and techniques presented in this paper.

REFERENCE

SAS Institute Inc. (1990), SAS® Guide to TABULATE Processing, Second Edition, Cary, NC: SAS Institute Inc.

AUTHORS

Pei Lee Tien
 Phone: (201) 503-5603
 Fax: (201) 503-6463
 PEI.TIEN@SANDOZ.COM

Mike McGranaghan
 Phone: (201) 403-1264
 Fax: (201) 226-3461
 MICHAEL.MCGRANAGHAN@ROCHE.COM