Paper 236-28

An Automated Reporting Macro to Create Cell Index – An Enhanced Revisit

Shi-Tao Yeh, GlaxoSmithKline, King of Prussia, PA

ABSTRACT

When generating tables from SAS® PROC TABULATE or PROC REPORT to summarize data, sometimes it is necessary to produce a cell index as part of the data display. A cell index is simply a data listing that reports the summarized count of each cell shown in the table.

This topic was presented and discussed at NESUG'01 Conference [7]. Several enhancements have since been made resulting in a major modification of the original code. It provides an automated SAS reporting macro to generate a cell index with several new features. The design goal for this program is to provide an efficient, flexible and easy-to-use macro.

The SAS product used in this paper is SAS BASE, with no limitation of operating systems.

INTRODUCTION

SAS System provides several useful tools for the production of tables and tabulations that summarize data. The procedures, PROC REPORT, and PROC TABULATE are very popular.

It is sometimes required to generate a cell index as part of the data display after producing a summarized table. The clinical study reviewer or SAS programmer uses a cell index for program validation purposes. The difference between a data listing and a cell index is that a cell index displays only categorical variables that appear in the table. It displays data observations to support the summary count shown in each cell of the table. The last column of a cell index usually displays the subject variable. It is also a preferred design to have the column before the last as a numeric count of the sub-categorical grouping of the subject variable.

The features and macro arguments of this macro are:

FEATURE	DESCRIPTION
dsin	a dataset name from which the cell index is to be produced
sortby	a categorical variable list that appeared in the table
idvar	a subject variable name
num_var	a variable that provides additional information for the cell index
freqsort	a variable name that is used for descending sort order
noprintv	a dummy variable that provides the correct sort order but not showing on cell index output
pageby	a variable list for output separated by page by variable list

The new and enhanced features are: **num_var**, **freqsort**, **noprintv**, and **pageby**.

This paper provides a step-by-step approach to developing a SAS reporting module for cell index production.

This paper is comprised of five parts. Part 1 includes the abstract and the introduction. Part 2 describes the sample data used throughout the paper. Part 3 is devoted to the program design and SAS code. Part 4 involves the SAS macro invocation and application. Part 5 concludes the paper.

SAS MACRO DESIGN SPECIFICATION FOR CELL INDEX

The program requirements for this macro are efficient, flexible and easy-to-use. This design goal can be achieved by minimizing the macro arguments. The framework of the program flow is described as follows

The main code consists of six steps:

STEP 1:

This step accepts the user's input macro argument information and prepares the dataset for the cell index output.

There are seven macro arguments, namely, ***dsin***, ***sortb***y, **num_var**, **freqsort**, **noprintv**, and ***idvar***.

STEP 2:

This step performs three counts: one for number of variables in the ***sortby*** list, the other for number of subject observations in each sub-category grouping, the last one for the space width available for listing of all variables in the ***sortby*** list.

STEP 3:

This step performs specific features for variables **freqsort**, **noprintv**, **pageby** and **num_var**.

STEP 4:

This step creates all SAS statements related to the procedure PROC REPORT. Then the SAS macro statements create a temporary SAS macro file named ***ctemp.sas***.

STEP 5:

SAS file ***ctemp.sas*** is run to create a cell index.

STEP 6:

All data files and temporary SAS macro files are deleted after the execution of the SAS macro ***get_cell.sas.***

Note: The X statement used in this program is for the UNIX platform. You need to issue your operating system command to delete the temporary file ctemp.sas

Note: The statement
`%let dwidth=%eval(%eval(94/%eval(&j-1)));`
is for reserved space width for all sortby variables. You can make further adjustments to fit your needs.

SAS CODE FOR CELL INDEX

The whole SAS macro code are shown as follows:

```
%macro get_cell(dsin=, sortby= , idvar=,
num_var=, freqsort=N, noprintv=, pageby=);
***********************************************;
* read in dataset ;
***********************************************;
data dsin;
     set &dsin;
     count=1;
proc sort;
     by &sortby;
%if &pageby ne  %then %do;
 proc sort data=dsin nodupkey;
    by &pageby &sortby &idvar;
%end;
run;
***********************************************;
* using &sortby for grouping to count number of
* objects in each cell
***********************************************;
proc summary data= dsin;
     by &pageby &sortby;
     var count;
     output out=pxyz1 sum=n;
data dsin;
     merge dsin pxyz1;
     by &pageby &sortby;
run;

%if %upcase(&freqsort) eq Y %then %do;

   proc sort data=dsin;by descending n ;

   data dsinn(keep=orderv n);
     set dsin;
     length orderv $5.;
     orderv = substr(left(_n_),1,5);
 run;
   proc sort data=dsin;by n;
   proc sort data=dsinn nodupkey;by n;
   data dsin;
    merge dsin dsinn;;
    by n;
 run;
%end;
%if %upcase(&pageby) ne %then %do;
   proc sort data=dsin;by &pageby;
%end;
***********************************************;
* count number of variables in &sortby list
***********************************************;
%macro varn;
     %local j;
     %let j=1;
     %let varn=%scan(&sortby,&j);
     %do %while ("&varn" ne "");
        %let j=%eval(&j+1);
        %let varn=%scan(&sortby,&j);
     %end;
     %global sortbyn dwidth;
     %let sortbyn=%eval(&j-1);
   %if &num_var ne %then %do;
     %let dwidth=%eval(%eval(81/%eval(&j-1)));
     %if &noprintv ne  %then %do;
        %let dwidth=%eval(%eval(81/%eval(&j-
2)));
     %end;
  %end;
  %else %do;
     %let dwidth=%eval(%eval(93/%eval(&j-1)));
     %if &noprintv ne  %then %do;
        %let dwidth=%eval(%eval(93/%eval(&j-
2)));
     %end;
  %end;
%mend varn;

%varn;
run;
***********************************************;
* covert &sortby list to a column that contains
* each variable in &sortby list
* as a record with one &sortby variable
***********************************************;
data f1;
     length &sortby 4.;
     %do i=1 %to &sortbyn;
         %scan(&sortby,&i)=1;
     %end;
run;
proc transpose data=f1 out=f2(keep=_name_);
     var &sortby;
data f2;
  set f2;
  if _name_ = "%upcase(&noprintv)" then delete;
run;
***********************************************;
* The following 3 datasets f1, f2, and f3,
* construct a temp SAS macro to perform
* proc report and produce a cell index by
* constructing a dataset with define statement
* for &sortby argument
***********************************************;
data f2;
     set f2 end=eof;
     var1 = 'define ' || _name_  || "  / order
left width=&dwidth flow     ; ";
     output;

     if eof then do;
        %if &noprintv ne  %then %do;
            var1 = 'define ' || "&noprintv"  ||
"  / order noprint   ; ";
            output;
        %end;
     end;
run;

***********************************************;
* construct a dataset with define statement for
* variable n and  &idvar;
***********************************************;
data f3;
     length var1 $90.;
     var1='define n / order width=5 "N";';
     output;
     var1="define &idvar / display width=15 flow
;";
     output;
     %if &num_var ne  %then %do;
         var1="define &num_var / display right
width=13   ;";
         output;
     %end;
     var1="break after %scan(&sortby, &sortbyn )
/skip;";
     output;
     var1='run;';
     output;
     var1=' %mend ctemp; ';
     output;
run;
***********************************************;
* construct a dataset with macro statement and
* proc report statement
***********************************************;
data f1;
     length var1 $90.;
     var1=' %macro ctemp; ';
     output;
     var1="proc report data=dsin headline
headskip split='~';";
     output;
     %if %upcase(&pageby) ne  %then %do;
      var1="by &pageby;";
     output;
     %end;
     %if %upcase(&freqsort) eq Y %then %do;
     var1="column  ('==' orderv &sortby n
&num_var &idvar);";
     output;
     var1="define orderv / order noprint   ;";
     output;
     %end;
     %else %if %upcase(&freqsort) ^= Y %then
%do;
     var1="column  ('=='  &sortby n &num_var
&idvar);";
     output;
     %end;
run;
data f2(keep=var1);
     set f1 f2 f3; run;
***********************************************;
* macro call ctemp.sas to produce cell index
***********************************************;
data _null_;
     set f2;
     file 'ctemp.sas';
     put var1;
run;
%inc 'ctemp.sas'; run;
%ctemp; run;
***********************************************;
* remove temp SAS macro ctemp.sas and clean up
* datasets
***********************************************;
x 'rm ctemp.sas'; run;
proc datasets nolist;
     delete pxyz1 f1 f2 f3 dsin dsinn;
run;
%mend get_cell;
```

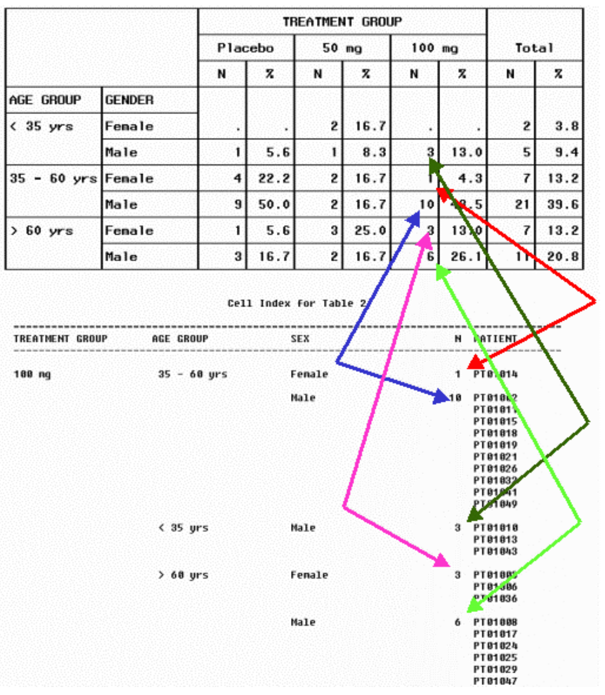
SAS MACRO INVOCATION AND APPLICATION

Before this macro invocation, it is important that you use the DATA STEP to prepare the SAS data set and label for variables to appeare in the cell index. In this section, individual an invocation example with code and a figure showing the relationship between table and cell index are provided. The sample macro invocation example 1 is as follows:

```
%get_cell(dsin = final,
          sortby = trt agec sex,
          idvar = patient);
```

Note: This macro accepts categorical variables in the sortby argument. If you have numeric variables in the sortby list, you need to convert them to character variables before invoking the macro.

A summary table displays the treatment variable as the across variable with categorical variables of age group and sex. The output of summary table and cell index from this example 1 is shown in Figure 1.



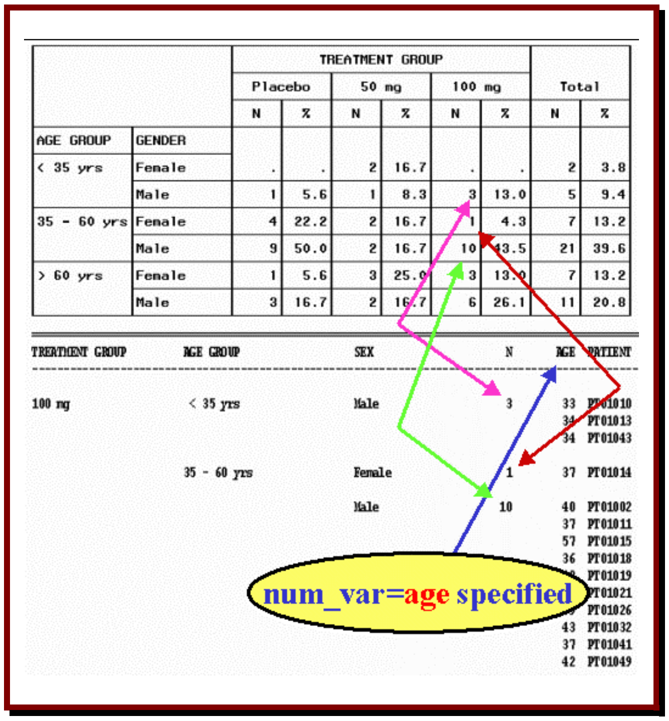
		TREATMENT GROUP						Total	
		Placebo		50 mg		100 mg			
		N	%	N	%	N	%	N	%
AGE GROUP	GENDER								
< 35 yrs	Female	.	.	2	16.7	.	.	2	3.8
	Male	1	5.6	1	8.3	3	13.0	5	9.4
35 - 60 yrs	Female	4	22.2	2	16.7	1	4.3	7	13.2
	Male	9	50.0	2	16.7	10	43.5	21	39.6
> 60 yrs	Female	1	5.6	3	25.0	3	13.0	7	13.2
	Male	3	16.7	2	16.7	6	26.1	11	20.8

```
                          Cell Index For Table 2
-----------------------------------------------------------------------------------
TREATMENT GROUP      AGE GROUP            SEX                       N  PATIENT
-----------------------------------------------------------------------------------
100 mg                35 - 60 yrs         Female                    1  PT01014
                                          Male                     10  PT01002
                                                                       PT01011
                                                                       PT01015
                                                                       PT01018
                                                                       PT01019
                                                                       PT01021
                                                                       PT01026
                                                                       PT01032
                                                                       PT01041
                                                                       PT01049
                      < 35 yrs            Male                      3  PT01010
                                                                       PT01013
                                                                       PT01043
                      > 60 yrs            Female                    3  PT01005
                                                                       PT01006
                                                                       PT01036
                                          Male                      6  PT01008
                                                                       PT01017
                                                                       PT01024
                                                                       PT01025
                                                                       PT01029
                                                                       PT01047
```

Figure 1 Output of Summary Table and Cell Index from Example 1

Example 2 uses the feature ***num_var***, to add additional variable to the cell index, either numeric or character variable, and specifies it as age. The output from this example is shown in Figure 2.

```
%get_cell(dsin = final,
        sortby = trt  agec sex,
        num_var = age,
        idvar = patient);
```



		TREATMENT GROUP						Total	
		Placebo		50 mg		100 mg			
		N	%	N	%	N	%	N	%
AGE GROUP	GENDER								
< 35 yrs	Female	.	.	2	16.7	.	.	2	3.8
	Male	1	5.6	1	8.3	3	13.0	5	9.4
35 - 60 yrs	Female	4	22.2	2	16.7	1	4.3	7	13.2
	Male	9	50.0	2	16.7	10	43.5	21	39.6
> 60 yrs	Female	1	5.6	3	25.0	3	13.0	7	13.2
	Male	3	16.7	2	16.7	6	26.1	11	20.8

```
TREATMENT GROUP     AGE GROUP          SEX              N     AGE  PATIENT
---------------------------------------------------------------------------

100 mg              < 35 yrs           Male             3      33  PT01010
                                                               34  PT01013
                                                               34  PT01043

                    35 - 60 yrs        Female           1      37  PT01014

                                       Male            10      40  PT01002
                                                               37  PT01011
                                                               57  PT01015
                                                               36  PT01018
                                                [illegible]  PT01019
                                                [illegible]  PT01021
                                                [illegible]  PT01026
                                                               43  PT01032
                                                               37  PT01041
                                                               42  PT01049
```

Figure 2 Cell Index with Additional Variable num_var Specified

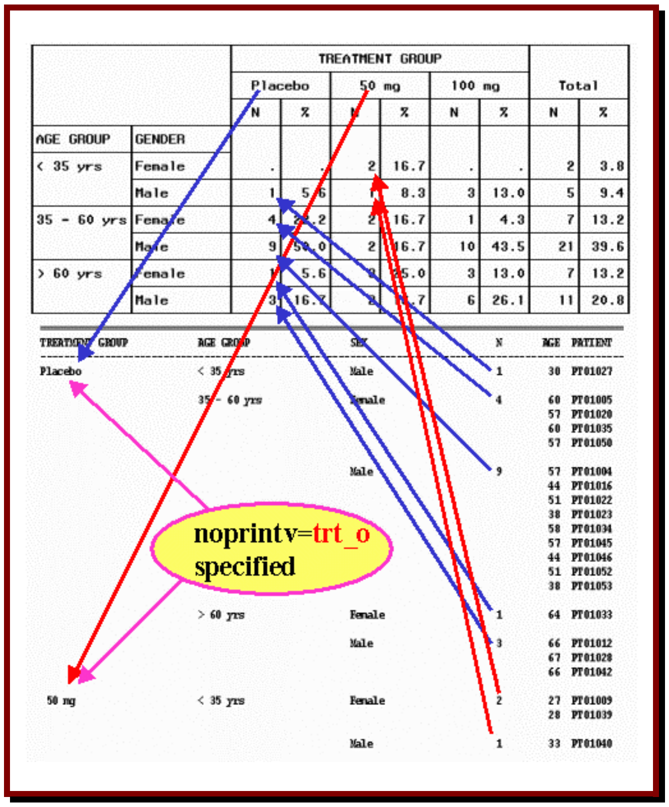
Sometimes the output table contains a certain sort order for the key variable. It is desired that the cell index contains the sort order for the same variable. You can create a 'no print' variable that controls the desired sort order without printing it on the output. Example 3 is a sample of this type of output.

```
%get_cell(dsin = final,
        sortby = trt_o trt  agec sex,
        num_var = age,
      noprintv = trt_o,
        idvar = patient);
```

Sometimes the output table program contains a by statement that output is breaking by this 'pageby' variable. It is desired that the cell index contains the same feature to break the output by 'pageby' variable.. Example 4 is a sample of this type of output.

```
%get_cell(dsin = lab,
        sortby = visit range,
        num_var = labhl,
        pageby = labtest trt,
```

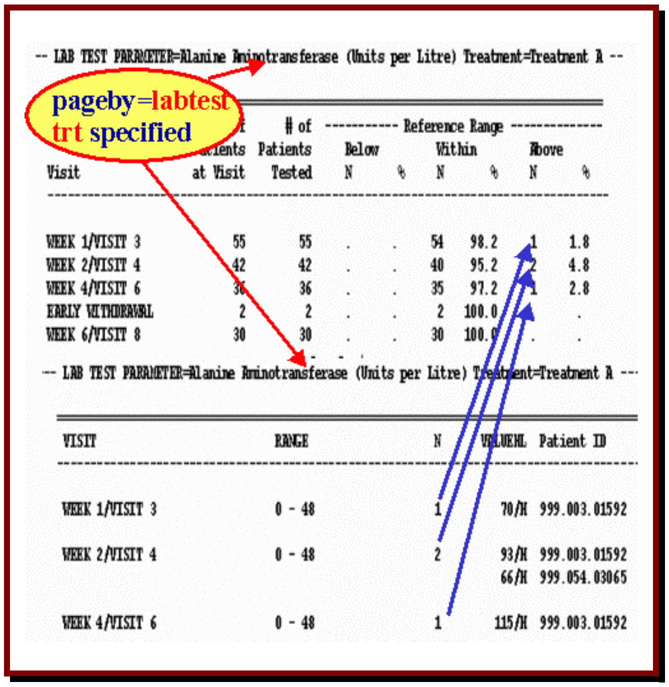
```
        idvar = pid);
```



		TREATMENT GROUP						Total	
		Placebo		50 mg		100 mg			
		N	%	N	%	N	%	N	%
AGE GROUP	GENDER								
< 35 yrs	Female	.	.	2	16.7	.	.	2	3.8
	Male	1	5.6	1	8.3	3	13.0	5	9.4
35 - 60 yrs	Female	4	22.2	2	16.7	1	4.3	7	13.2
	Male	9	50.0	2	16.7	10	43.5	21	39.6
> 60 yrs	Female	1	5.6	3	25.0	3	13.0	7	13.2
	Male	3	16.7	2	16.7	6	26.1	11	20.8

```
TREATMENT GROUP     AGE GROUP          SEX              N     AGE  PATIENT
---------------------------------------------------------------------------
Placebo             < 35 yrs           Male             1      30  PT01027

                    35 - 60 yrs        Female           4      60  PT01005
                                                               57  PT01020
                                                               60  PT01035
                                                               57  PT01050

                                       Male             9      57  PT01004
                                                               44  PT01016
                                                               51  PT01022
                                                               38  PT01023
                                                               58  PT01034
                                                               57  PT01045
                                                               44  PT01046
                                                               51  PT01052
                                                               38  PT01053

                    > 60 yrs           Female           1      64  PT01033

                                       Male             3      66  PT01012
                                                               67  PT01028
                                                               66  PT01042

 50 mg              < 35 yrs           Female           2      27  PT01009
                                                               28  PT01039

                                       Male             1      33  PT01040
```

Figure 3 Cell Index with noprintv Specified

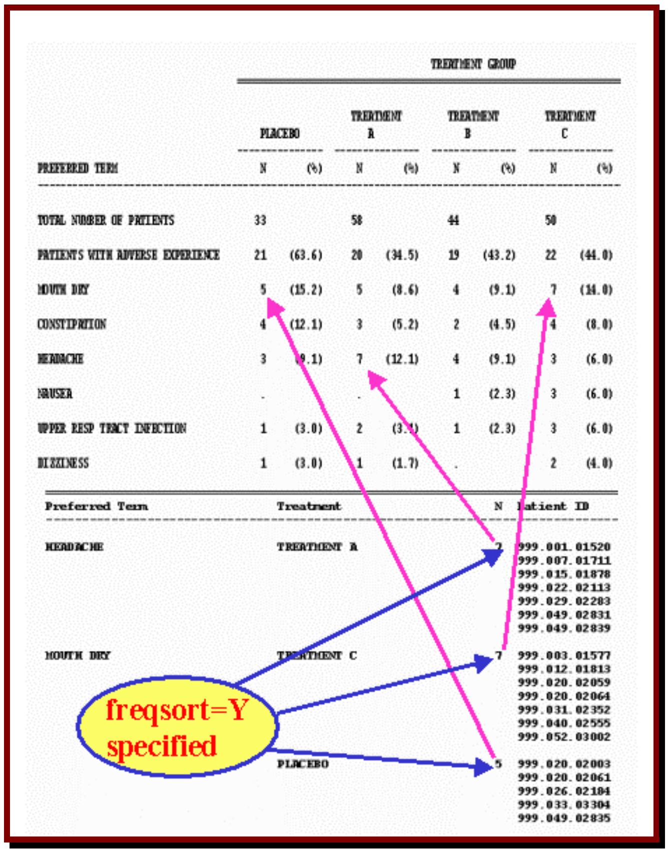


```
-- LAB TEST PARAMETER=Alanine Aminotransferase (Units per Litre) Treatment=Treatment A --

                 # of       # of  ----------- Reference Range -------------
              Patients   Patients      Below         Within          Above
Visit         at Visit     Tested    N      %      N       %      N       %
----------------------------------------------------------------------------


WEEK 1/VISIT 3      55         55    .      .     54    98.2      1     1.8
WEEK 2/VISIT 4      42         42    .      .     40    95.2      2     4.8
WEEK 4/VISIT 6      36         36    .      .     35    97.2      1     2.8
EARLY WITHDRAWAL     2          2    .      .      2   100.0      .       .
WEEK 6/VISIT 8      30         30    .      .     30   100.0      .       .

-- LAB TEST PARAMETER=Alanine Aminotransferase (Units per Litre) Treatment=Treatment A ---


 VISIT                     RANGE                N   VALUEHL  Patient ID
 --------------------------------------------------------------------------


 WEEK 1/VISIT 3            0 - 48               1      70/H  999.003.01592

 WEEK 2/VISIT 4            0 - 48               2      93/H  999.003.01592
                                                       66/H  999.054.03065

 WEEK 4/VISIT 6            0 - 48               1     115/H  999.003.01592
```

Figure 4. Cell Index with pageby Specified

Sometimes the output table contains a certain descending sort order for the key variable. It is desired that the cell index contains the descending sort order for the frequency count N. Example 5 is a sample of this type of output.

```
%get_cell(dsin = aesub,
          sortby = pref trt ,
          freqsort = Y,
          idvar = pid);
```



```
                                              TREATMENT GROUP
                                 ==========================================
                                             TREATMENT   TREATMENT   TREATMENT
                                   PLACEBO       A           B           C
                                 ----------  ----------  ----------  ----------
PREFERRED TERM                    N    (%)    N    (%)    N    (%)    N    (%)
-------------------------------------------------------------------------------

TOTAL NUMBER OF PATIENTS         33          58          44          50

PATIENTS WITH ADVERSE EXPERIENCE 21  (63.6)  20  (34.5)  19  (43.2)  22  (44.0)

MOUTH DRY                         5  (15.2)   5   (8.6)   4   (9.1)   7  (14.0)

CONSTIPATION                      4  (12.1)   3   (5.2)   2   (4.5)   4   (8.0)

HEADACHE                          3   (9.1)   7  (12.1)   4   (9.1)   3   (6.0)

NAUSEA                            .           .           1   (2.3)   3   (6.0)

UPPER RESP TRACT INFECTION        1   (3.0)   2   (3.4)   1   (2.3)   3   (6.0)

DIZZINESS                         1   (3.0)   1   (1.7)   .           2   (4.0)

  ==============================================================================
  Preferred Term                  Treatment                      N  Patient ID
  ------------------------------------------------------------------------------

  HEADACHE                        TREATMENT A                    7  999.001.01520
                                                                    999.007.01711
                                                                    999.015.01878
                                                                    999.022.02113
                                                                    999.029.02283
                                                                    999.049.02831
                                                                    999.049.02839

  MOUTH DRY                       TREATMENT C                    7  999.003.01577
                                                                    999.012.01813
                                                                    999.020.02059
                                                                    999.020.02064
                                                                    999.031.02352
                                                                    999.040.02555
                                                                    999.052.03002

                                  PLACEBO                        5  999.020.02003
                                                                    999.020.02061
                                                                    999.026.02184
                                                                    999.033.03304
                                                                    999.049.02835
```

CONCLUSION

This paper takes a comprehensive approach in designing a reporting module for creation of a cell index. SAS code and the examples of how to invocate the SAS macro are provided in this paper. This paper achieves the following goals:

* Provides an easy-to-use macro with only seven macro arguments
* Provides enhanced features to fit the different needs
* Provides a compact, efficient, and simple code for cell index production.

ACKNOWLEGMENTS

The author would like to thank Ellen Forman, Shu-Mei Huang, Biostatistics & Data Management, GlaxoSmithKline, and Ju Zhang from EDP Contract Services, for their review, validation and enhancement suggestions of this macro.

REFERENCES

[1] Alissa S. Bernholc: "*A Tip for Using the Tabulate Procedure: Generating Tables with Percentages in Only One Category*" Observations, online article, June1998, SAS Institute Inc., Cary, NC, USA

[2] Stephen M. Noga, Jeffery M. Abolafia: "*The Tabulate Procedure: One Step Beyond the Final Chapter*" Proceedings of the Twenty-Third Annual SAS Users Group International Conference, pp. 839-844, March 1998

[3] Ron Coleman: "*The Building Blocks of PROC TABULATE"* Proceedings of the Twenty-Third Annual SAS Users Group International Conference, pp. 34-39, March 1998

[4] Dan Bruns: "*The Utter Simplicity of the TABULATE Procedure – The Final Chapter*", Proceedings of the Sixteenth Annual SAS Users Group International Conference. 1997

[5] Dan Bruns: "*The Utter Simplicity of the TABULATE Procedure – The Sequel*", Proceedings of the Sixteenth Annual SAS Users Group International Conference. 1996

[6] Shi-Tao Yeh ; "*SAS® Macros for Grouping Count and Its Application to Enhance Your Reports*", NESUG '99 Annual Conference Proceedings, pp. 583-587, October 1999

[7] ___________ "*An Automated Reporting Macro to Create Cell Index*", NESUG 2001 Annual Conference Proceedings, pp. 711-715, September, 2001

Author
Shi-Tao Yeh, Ph. D.
(610)917-5883(W)
E-mail: shi-tao_yeh-1@gsk.com